12d Model Reference Manual

Version 11

May 2015

12D SOLUTIONS PTY LTD

ACN 101 351 991

PO Box 351 Narrabeen NSW Australia 2101

Australia Telephone (02) 9970 7117 Fax (02) 9970 7118

International Telephone 61 2 9970 7117 Fax 61 2 9970 7118

Email support@12d.com Web www.12d.com.

12d Model Reference Manual

This book is the reference manual for the software product 12d Model.

First Release Dates of Software and Manual

V2.0	July 1990
V3.0	October 1995
V4.0	April 1999
V5.0	December 2000
V6.0	September 2002
V7.0	January 2005
V8.0	March 2007
V9.0	May 2009
V10.0	April 2012
V11.0	October 2014

Disclaimer

12d Model is supplied without any express or implied warranties whatsoever.

No warranty of fitness for a particular purpose is offered.

No liabilities in respect of engineering details and quantities produced by 12d Model are accepted.

Every effort has been taken to ensure that the advice given in this manual and the program 12d Model is correct, however, no warranty is expressed or implied by 12d Solutions Pty Ltd.

Copyright

1 Preface

1.1 Introduction

12d Model is an object oriented, interactive graphics program designed to process survey data, quickly build terrain, conceptual and detail design models. It is used in constructing the design, doing quality check and collecting as-constructed data.

So ***12d Model*** is used in all stages of the civil planning, design and construction.

Data is easily read in, triangulated and contoured to build an initial terrain model. Roads, platforms, channels or other design features can be added interactively and a merged model containing the initial terrain and the new design features formed to produce conceptual design models.

All Models can be examined in plan, section or perspective views. The number and type of views displayed on the screen is totally user defined.

By using a mouse and flexible on-screen menus,***12d Model*** is easy to use and requires a minimum of training.

This document is the ***12d Model Reference*** manual.

1.2 Reference Manual in PDF Form

12d Model 11 has a Help available from within ***12d Model*** for most panels and menus.

However, because Microsoft's Help system only allows individual topics to be printed, the entire 12d Model Reference manual has also been supplied as a PDF file. The PDF file can be used to print out large sections of the manual. Adobe's PDF format can be read by Adobe Acrobat or the free Acrobat Reader

The PDF file for the ***12d Model*** *Reference* manual is called

12dm_ref.pdf

and is in the folder on the ***12d Model*** *Installation* DVD called

Documentation\Reference_Manual

If you do not have an *Acrobat* Reader installed, it is available on the ***12d Model*** *Installation* DVD under **Install**.

1.3 Getting Started Manuals and Data

12d Model is supplied with a very comprehensive on -line Reference manual which describes the function of each menu option in detail. However it is a Reference manual and makes no attempt to describe how to use 12d for production surveying and civil engineering work.

The ***12d Model*** *installation* DVD also contains two (2) Training manuals:

Getting Started for Design manual

Getting Started for Surveying manual

The ***Getting Started for Design*** manual is available as a printed manual and as a PDF file on the ***12d Model*** *Installation* DVD.

The ***Getting Started for Surveying*** has the first seven chapters in common with the *Getting Started for Design* manual (installing ***12d Model***, on-line help and basic modelling) but then diverts to cover topics from the direction of a Surveying whereas the *Getting Started for Design* manual continues on with alignment design techniques.

The *Getting Started for Surveying* manual is available as a printed manual and as a PDF file on the ***12d Model*** *Installation DVD*.

As well as the ***Getting Starte***d manuals, there are videos on a DVD which also work through all the material in the ***Getting Started*** manuals.

The ***Getting Started*** tutorials assumes that a series of files are already on your hard disk. These tutorial files are automatically installed from the DVD during installation of the ***12d Model*** software

1.3.1 Training Courses

There is a wide range of ***12d Model*** *Training* Courses given by official ***12d Model*** training centres.

Each training course usually has its own set of training files, manual in pdf form and videos going through the training course. This material is distributed on DVDs.

1.3.2 Using the Practise and Small Versions of 12d Model

The Practise version of ***12d Model*** is limited to a maximum of 5,000 points. Following the procedures as stated in the training manuals may create projects with more than 5,000 points.

Where appropriate, the text will suggest how to vary the input for each instruction so that the example feature can be completed within the limits of the ***12d Model*** Practise version.

The number of points used at any time in the Practise and small release versions can be displayed by the option

Projects => Check points

The easiest way to reduce the current point count is to delete any unwanted models with

Models => Delete

The installed icon on your desktop for running the practise version of 12d with these training files is labelled ***12d 11 Practise Training***.

Please Note: Projects created by Practise versions of ***12d Model*** cannot be accessed by Release versions of ***12d Model*** and vice-versa.

2 Installation of 12d Model 11 Release Version

The *12d Model 11 Installation DVD* can be used to install the *Release* and *Practise* versions of ***12d Model 11***.

The *Practise* version is limited to a maximum of 5,000 points and creates projects that cannot be accessed by the *Release* versions of *12d Model* and vice-versa. However the Practise version can be used free of charge by *12d Solutions* customers and registered Practise Users.

These notes are for installing the ***Release*** *version* of *12d Model 11*. There are separate notes for installing the Practise version.

For a new installation of the Release version of *12d Model 11*, the user is provided with

(a) one *12d Model* dongle

(b) one *12d Model 11 Installation DVD*

(c) an email with the *12d Model 11* authorization file *nodes.12d11n* attached, or a folder wit the *12d Model 11* authorization file *nodes.12d11n* or *nodes.4d* in it.

Please check that you have all three items before commencing the installation.

For existing *12d Model 11* users, the user is provided with

(a) one *12d Model 11 Installation DVD*

(b) an email with the *12d Model 11* authorization file *nodes.12d11n* attached, or a folder with the *12d Model 11* authorization file *nodes.12d11n* or *nodes.4d* in it.

For existing users, the dongle you already have for *12d Model 11* will work with the new version of *12d Model 11 once you have the new nodes file for 12d Model 11*. If *12d Model 11* is **already** running on your computer, please **uninstall** it before installing a new version of *12d Model 11*.

Important Note on *Nodes.12d11n* and *Nodes.4d*

The *12d Model 11* ***nodes.12d11n*** file will normally be emailed to you.

During the installation of *12d Model 11*, an association is created for files ending in ***.12d11n*** so after the installation, opening the ***nodes.12d11n*** will automatically copy it to the required area (normally *C:\12d\11.00*) and rename it to ***nodes.4d***.

If you do not have the ***nodes.12d11n*** attached to an email but instead have it in a folder, clicking on the ***nodes.12d11n*** file after installing *12d Model 11* will also automatically copy it to the required area (normally *C:\12d\11.00*) and rename it to ***nodes.4d***. Or if the *12d Model 11* icon is clicked on, the **Project Selection** panel will be brought up and clicking on the **Nodes** button will allow the ***nodes.12d11n*** or ***nodes.4d*** file to be selected and then processed.

Please **do not change** the **name** of the created ***nodes.4d*** file. ***12d Model*** will only search for an authorization file called *nodes.4d.*

Extra Notes for All Installations

1. A three-button mouse is essential.
2. The *12d Model Installation DVD* contains extra information other than just the installation version of 12d Model. For example, documentation, source to macros and plot parameter files.

   The extra data can be copied from the *12d Model Installation DVD* but the copied files may only have a “read only” attribute set. This means that the files can not be edited or modified in any way.

   To change the attribute so that a file can be modified, select the file in Explorer, bring up the *Properties* sheet and under the *General* tab sheet change the “read only” box so that it is not ticked on.

   Selecting ***OK*** or ***Apply*** will then modify the attribute of the file.

2.1 12d Model 11 - 32 bit or 64 bit Version ?

Microsoft has a 32 bit Windows and a 64 bit Windows operating system.

So when installing **12d Model 11**, you need to select either the 64 bit 12d installation or the 32 bit installation.

In 64 bit Windows, Microsoft have allowed for both 32 bit and 64 bit versions of the same software to coexist on the same computer, so you could install both the 32 bit and the 64 bit version of **12d Model 11**.

The 64 bit **12d Model 11** only runs on 64 bit Windows and supports any of the **12d Mode**l point sizes from 5K through to 5M and 250M.

The 32 bit **12d Model 11** runs on both 32-bit and 64 bit Windows and supports any of the **12d Mode**l point sizes from 5K through to 5M.

So if you require **12d Model 11** for 250M then you must run 64 bit **12d Model 11** on 64 bit Windows.

2.2 Installing the Release Version of 12d Model 11

These notes are for installing the Release version of *12d Model 11.* There are separate notes for installing the Practise version.

A. Disk Space required for installing 12d Model 11

Approximately 2 Gigabytes of disc space will be required for the installation to succeed. Before installing from the DVD

For Windows 2000, XP, Vista, Windows 7:

It is usually best to reboot the PC **before** installing **12d Model** from the DVD and have no other applications running.
Your login must have Administrator privileges.

B. Do not attach the 12d dongle before installing dongle drivers

USB dongles must **NOT be attached** to the computer before the dongle drivers are installed.

C. Using the email with the *nodes.12d11n* file attached to it

The *12d Model 11* ***nodes.12d11n*** file will normally be emailed to you.

During the installation of *12d Model 11*, an association is created for files ending in ***.12d11n*** so after the installation, opening the ***nodes.12d11n*** will automatically copy it to the required area (normally *C:\12d\11.00*) and rename it to ***nodes.4d***.

If you do not have the ***nodes.12d11n*** attached to an email but instead have it in a folder, clicking on the ***nodes.12d11n*** file after installing *12d Model 11* will also automatically copy it to the required area (normally *C:\12d\11.00*) and rename it to ***nodes.4d***. Or if the *12d Model 11* icon is clicked on, the **Project Selection** panel will be brought up and clicking on the **Nodes** button will allow the ***nodes.12d11n*** or ***nodes.4d*** file to be selected and then processed.

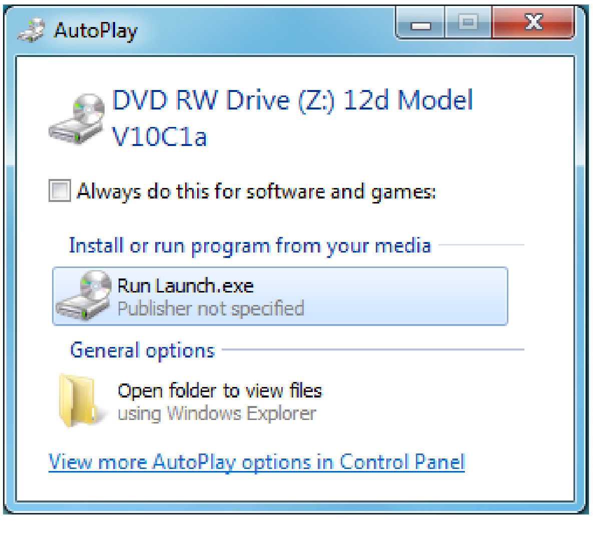
D. Starting the Installation of *12d Model 11*

Insert the ***12d Model 11 Installation DVD*** into the DVD drive.

On inserting the DVD, the ***12d Model Installation*** program automatically begins.

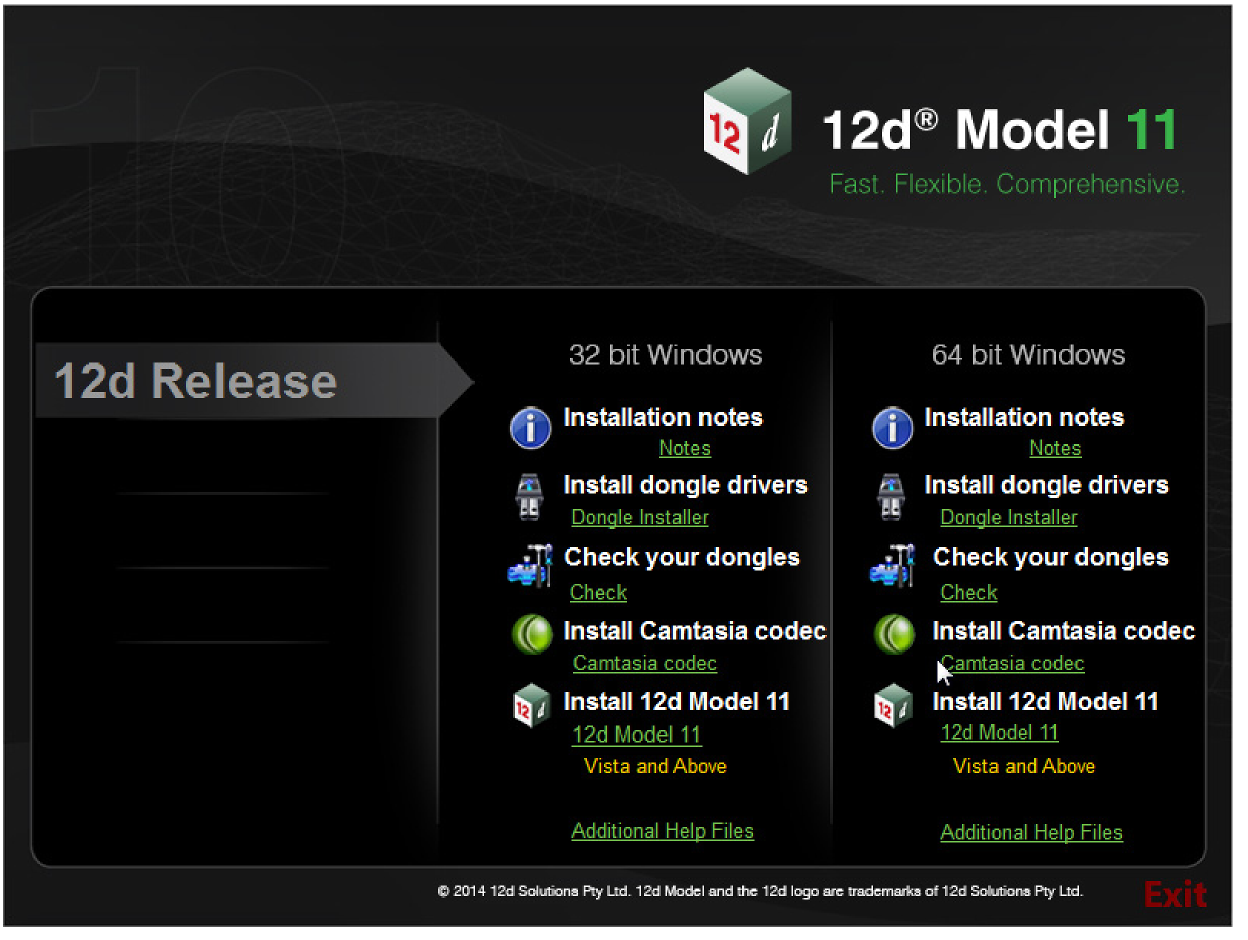
If it doesn't, simply double click on the program ***Launch.exe*** on the DVD.

Depending on the Windows operating system and your settings, you may be asked to allow *Launch.exe* to run. If so, click on ***Run Lauch.exe***.



E. Installing

The *12d Model Release* screen will appear.



The Steps on the ***12d Release*** tab of the *12d Model Installation DVD* will lead you through the installation of the Release version of *12d Model*. The sequence on the left is for installing on 32 bit Windows, and the sequence on the right is for installing on 64 bit Windows.

For running 12d Model 11 on 32 bit Windows:

If you are running 12d Model on 32 bit Windows, then you need to install the *12d Model* 32 bit 12d.exe and 32 bit dongle drivers. So go down the options on the left hand side of the screen.

Note- the 32 bit 12d.exe will run point versions **up to 5M**. The 250M version of *12d Model* is only available as the 64 bit *12d Model* and it only runs on a Windows 64 bit operating system.

For running 12d Model 11 on 64 bit Windows:

If you are running 12d Model on 64 bit Windows, then you need to install the *12d Model* 64 bit 12d.exe and 64 bit dongle drivers. So go down the options on the right hand side of the screen.

The 64 bit 12d.exe runs all point versions up to and including 250M.

For each installation, ***Installation Notes*** in pdf format can be viewed by clicking on ***Notes*** under the ***Installation notes*** heading. If ***Videos*** also appears then a video to guide you through the installation process can be viewed by clicking on ***Videos*** under the ***Installation notes*** heading.

NOTE - If the correct dongle drivers and the Camtasia codec are already installed on your computer then you can proceed to Step 4. Installing 12d Model 11 Release Program files

Step 1. Install the Wibu or Codemeter Drivers

You need to have System Administrator rights to install the drivers.

Installing Dongle Drivers:

On the *12d Model Release* screen, under *Install Dongle Drivers*, click on ***Drivers*** to bring up the **Dongle Drivers** panel:

Depending on the Windows operating system and your privileges (you need Administrator rights to install the drivers), the **User Account Control** panel may appear and you will be asked *Do you want the following program to make changes to your computer? If so, click on **Yes**.*

The **Dongle Drivers** panel will then appear.

If you only have one type of dongle Wibu (green USB, translucent green USB) or Codemeter (variety of colours and sizes) then you only need to install the drivers for those dongles.

If you have both *Codemeter* and *Wibu* dongles then both sets of drivers ***must*** be installed.

(a) If you have both **Wibu** and **Codemeter** ticked on then clicking on **Install** will first install the Wibu drivers and then continue on to installing the Codemeter drivers.

Go to Installing the Wibu Drivers:.

(b) If you have only **Wibu** ticked on then clicking on **Install** will install the Wibu drivers.

Go to Installing the Wibu Drivers:.

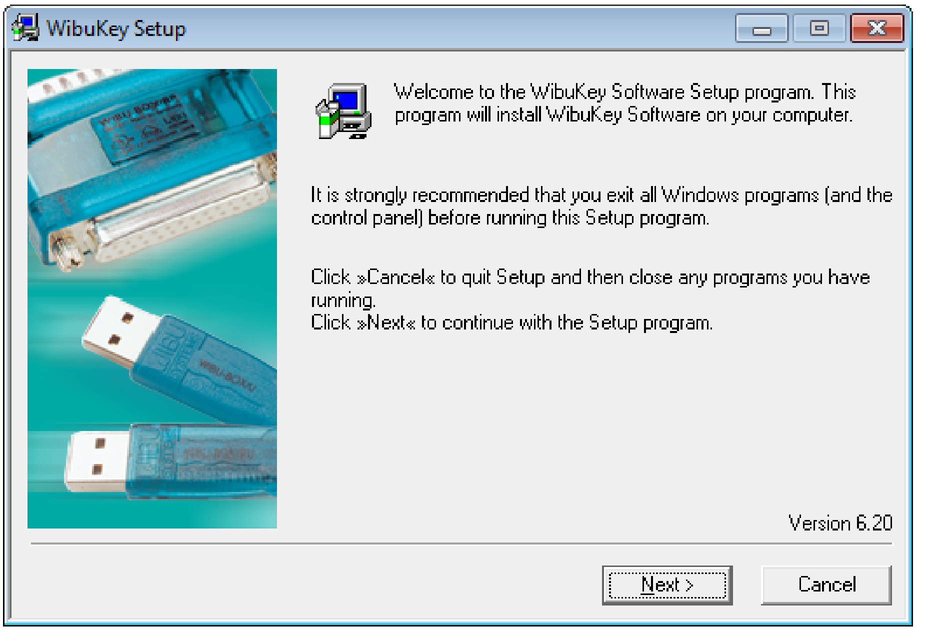
(c) If you have only **Codemeter** ticked on then clicking on **Install** will install the Codemeter drivers.

Go to Installing the Codemeter Drivers:

Installing the Wibu Drivers:

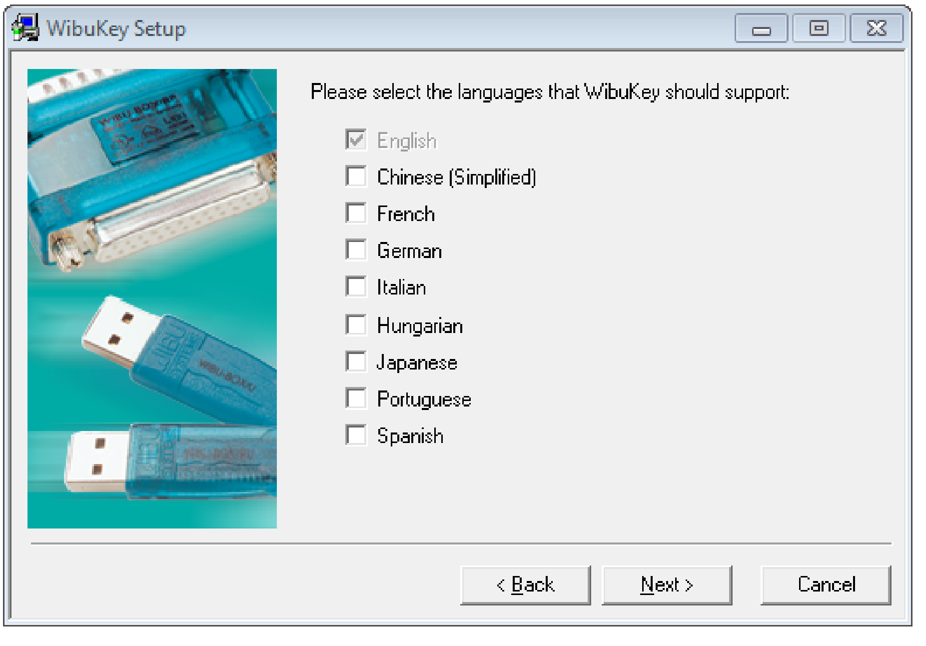
The script to install the **Wibu** dongle drivers begins.

WIBU-KEY Setup *Welcome to WIBU-KEY Software Setup*



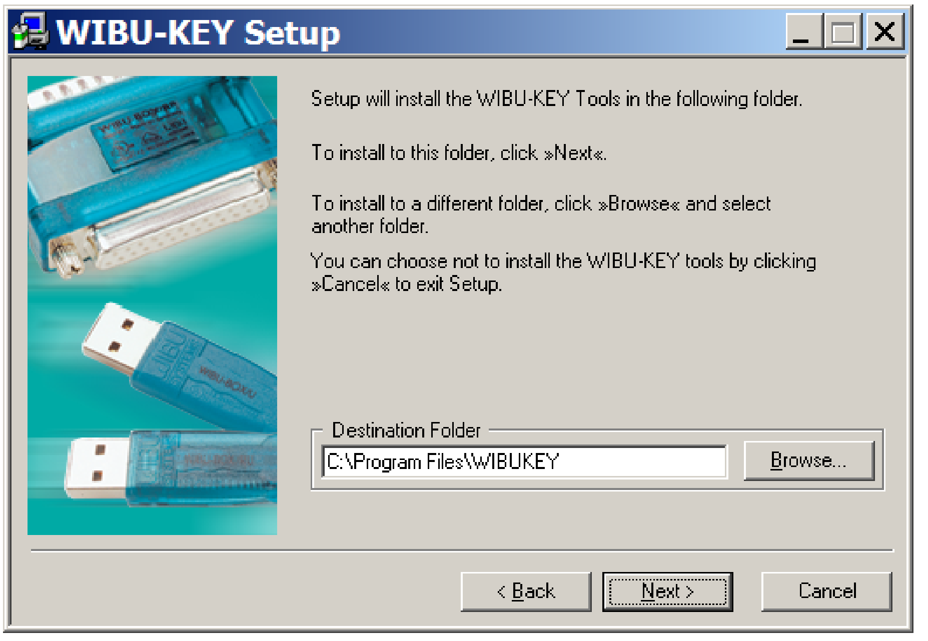
select ***Next*** to continue

WIBU-KEY Setup *Language Selection*



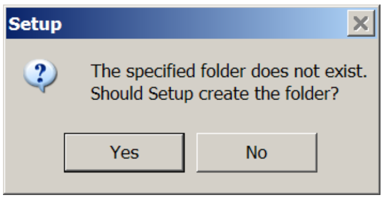
tick your language and then select ***Next*** to continue

WIBU-KEY Setup *Installation folder*



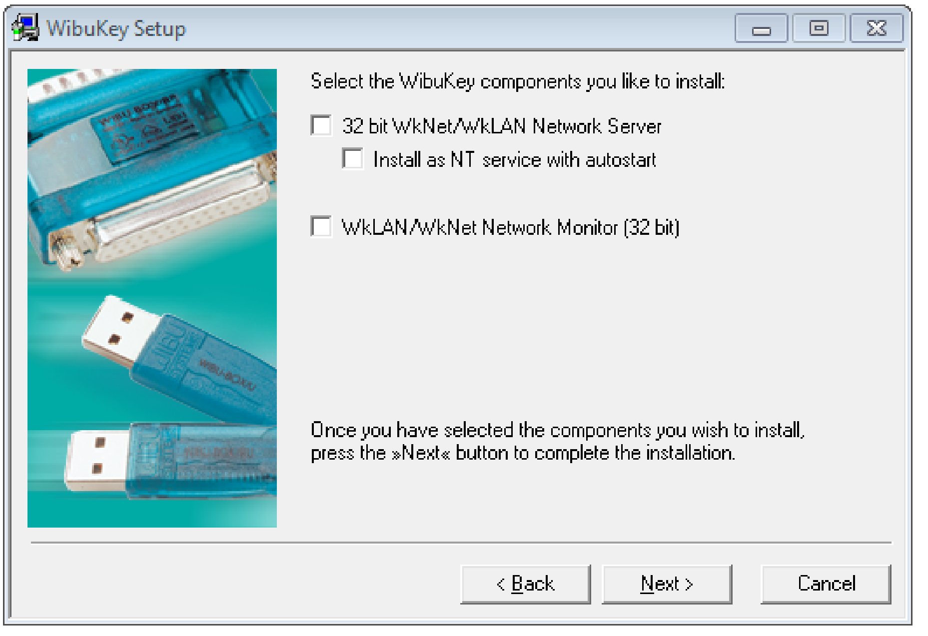
select ***Next*** to continue

WIBU-KEY Setup *Installation folder doesn't exist*



click Yes to continue

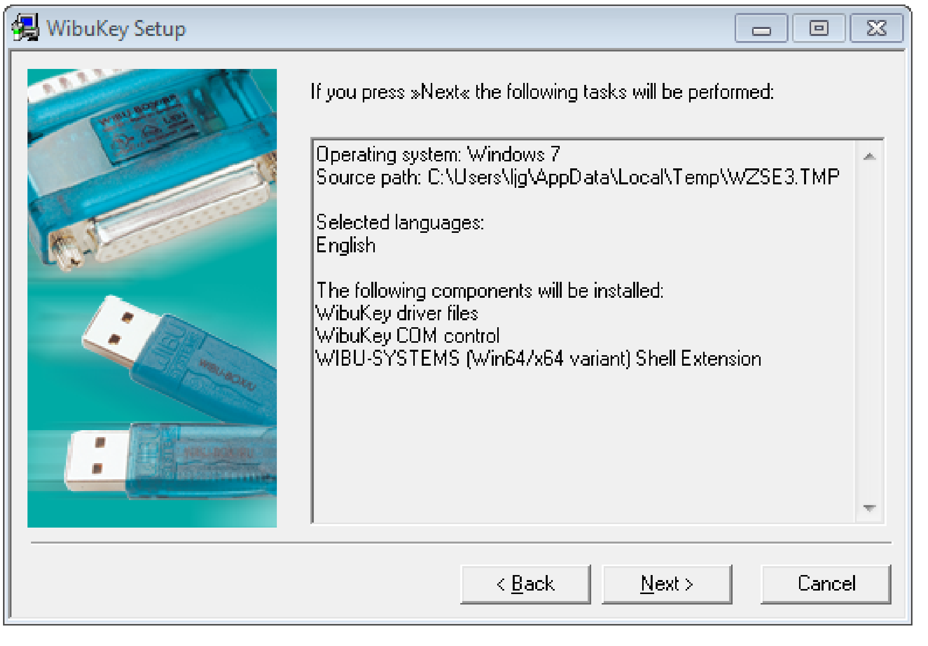
WIBU-KEY Setup *Component Selection*



For a standard install, nothing needs to be ticked on (the WibuKey components are for monitoring a Wibu network dongle).

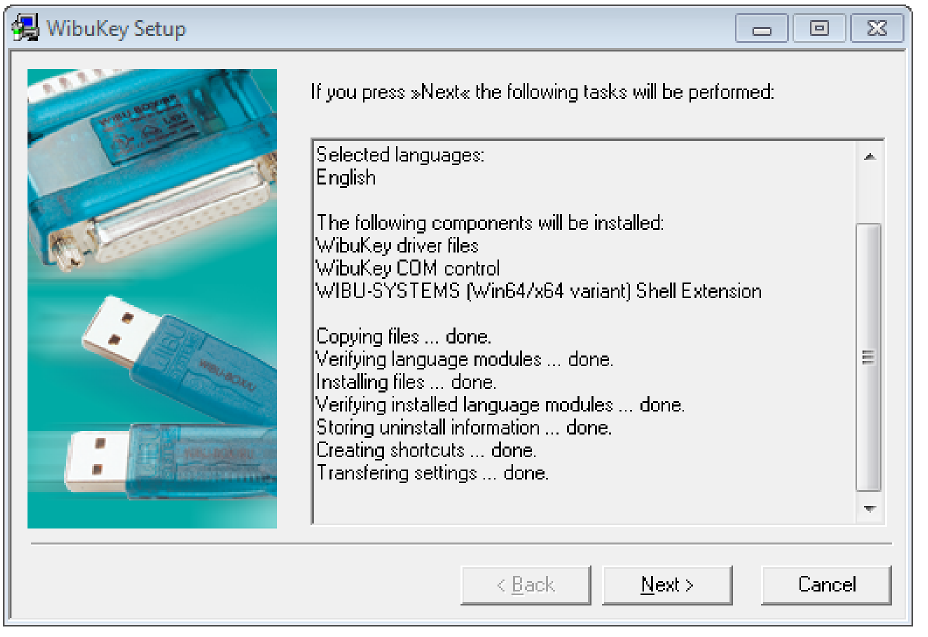
make sure nothing is ticked and then select ***Next*** to continue

WIBU-KEY Setup *Tasks to be Performed*



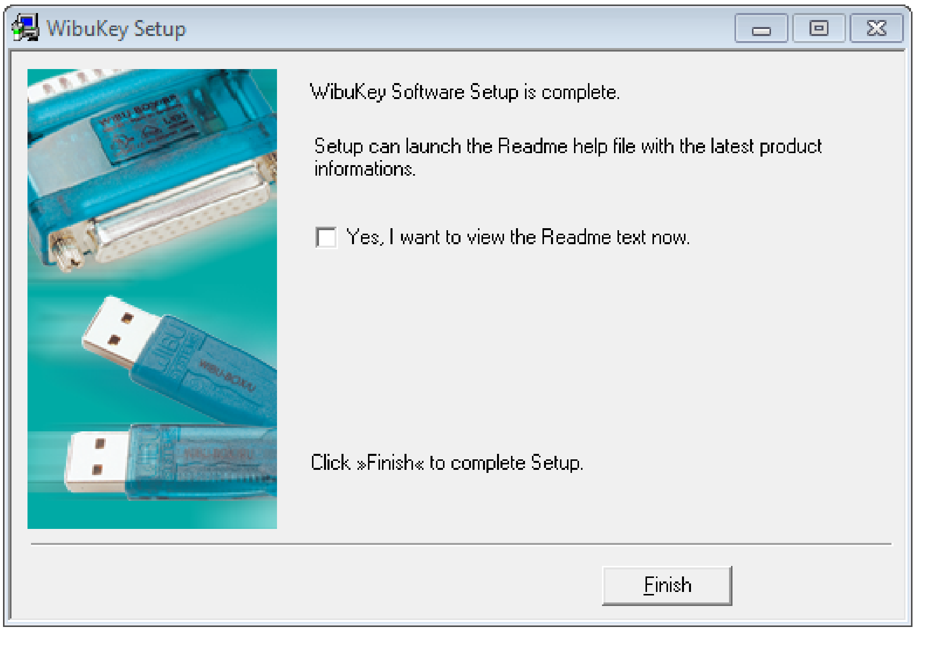
select ***Next*** to continue

WIBU-KEY Setup *Tasks done*



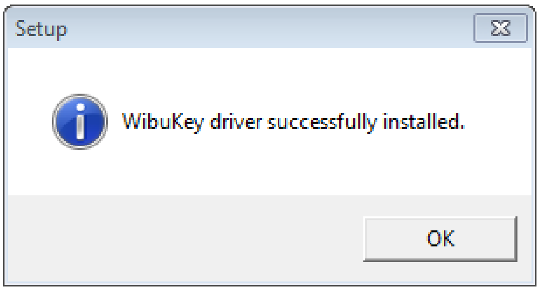
select ***Next*** to continue

WIBU-KEY Setup *Setup Complete*



leave **Yes, I want to view the Readme text now** unticked and select ***Finish***

Setup OK *Setup OK*

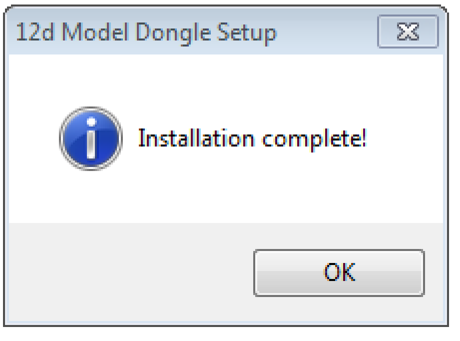


click ***OK***

The WIBU dongle drivers have now been installed

If both **Wibu** and **Codemeter** were ticked on then the installation of the Codemeter drivers will then begin. See Installing the Codemeter Drivers:,

Otherwise the **12d Model Dongle Setup** screen will appear.



click ***OK***

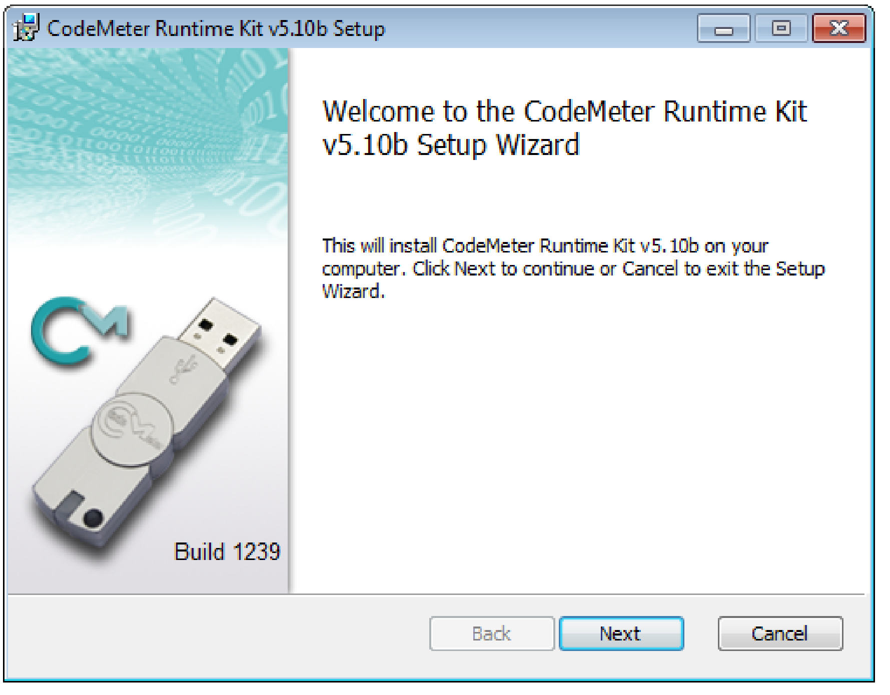
The *12d Model Release* screen will then reappear.

Continue to Step 2. Check the dongle.

Installing the Codemeter Drivers:

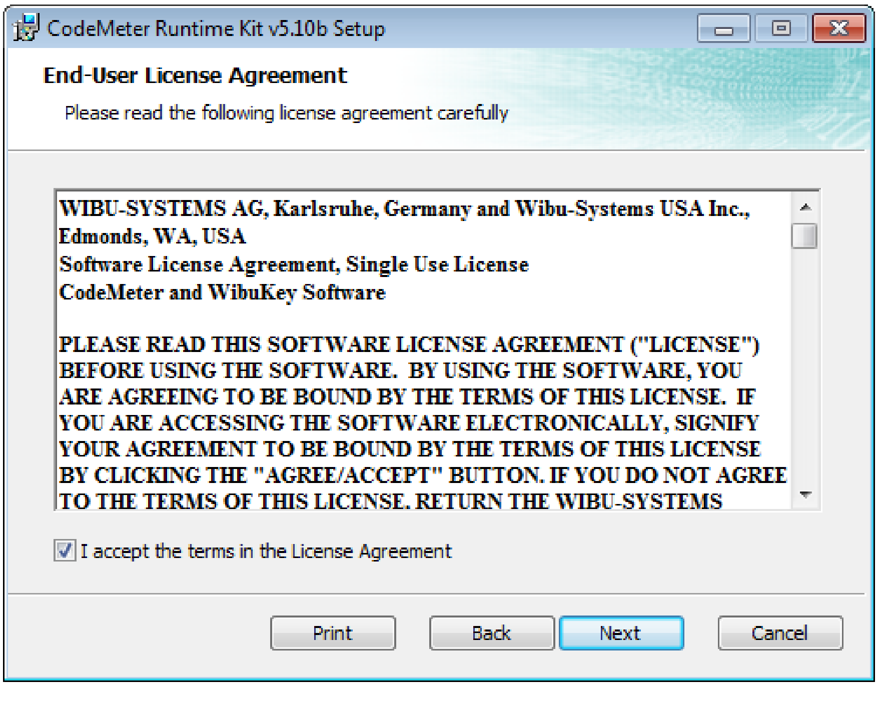
The script to install the **Codemeter** dongle drivers begins.

Codemeter Setup *Codemeter Runtime Kit Setup*



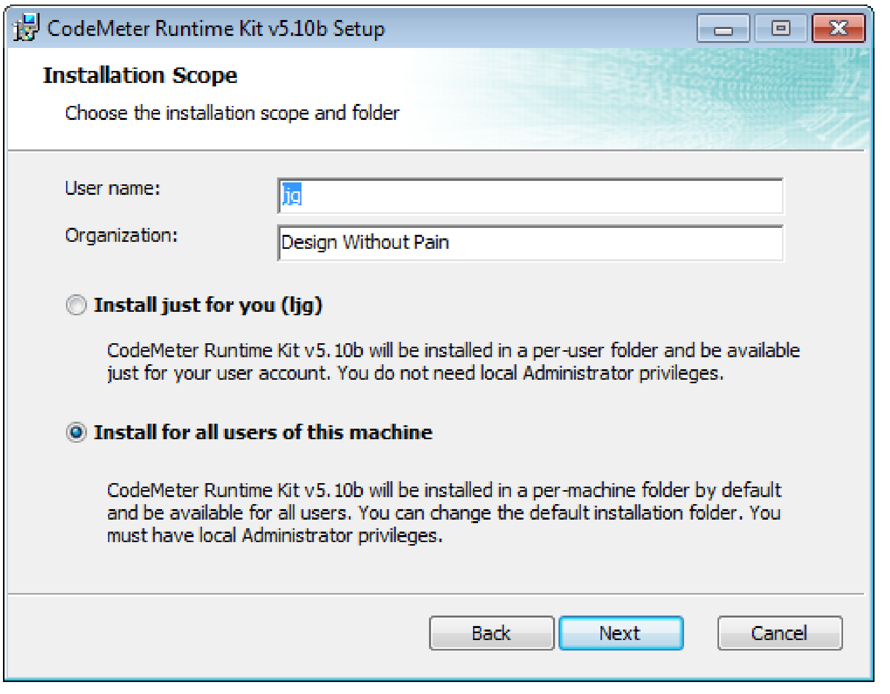
Select ***Next*** to continue

End-User License Agreement *Codemeter Runtime Kit Setup*



Read the License Agreement and if you are happy with it,
tick I accept the terms in the License Agreement and then select ***Next*** to continue

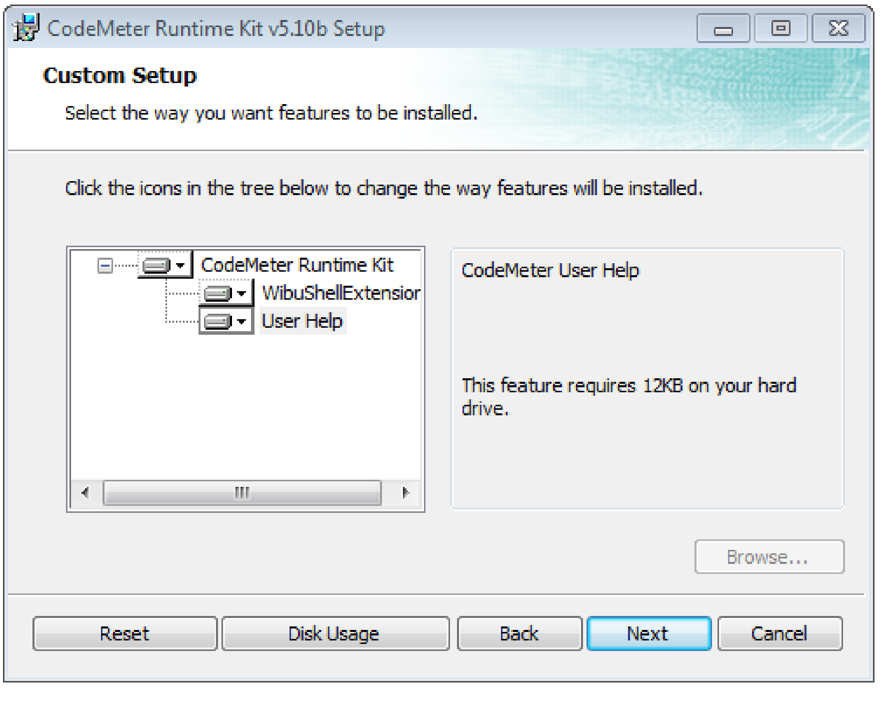
Installation Scope *Codemeter Runtime Kit Setup*



Enter your **User name** and **Organisation**.

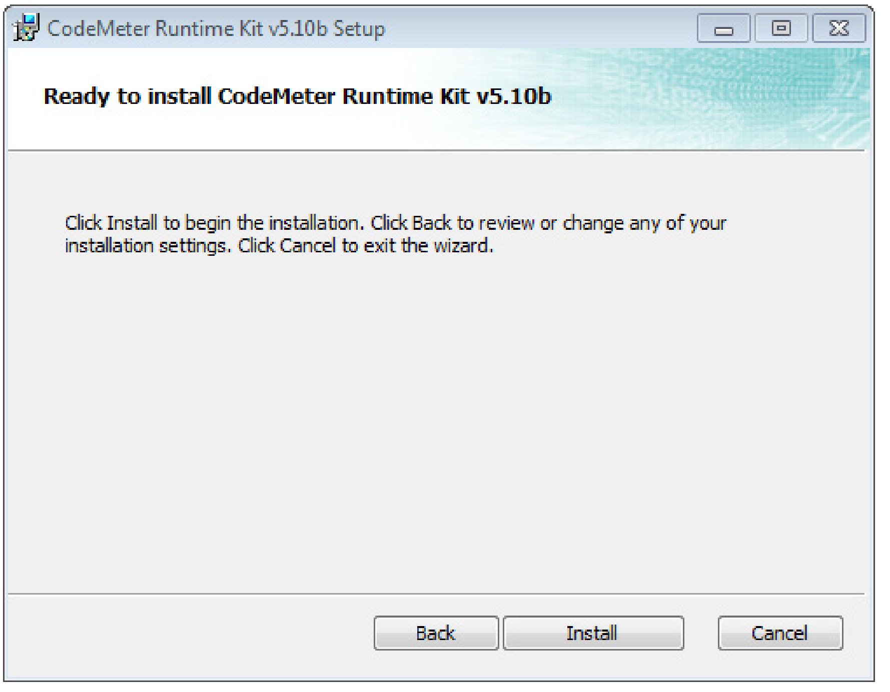
tick either **Install just for you** or **Install for all users of this machine** and then select ***Next*** to continue

Custom Setup *Codemeter Runtime Kit Setup*



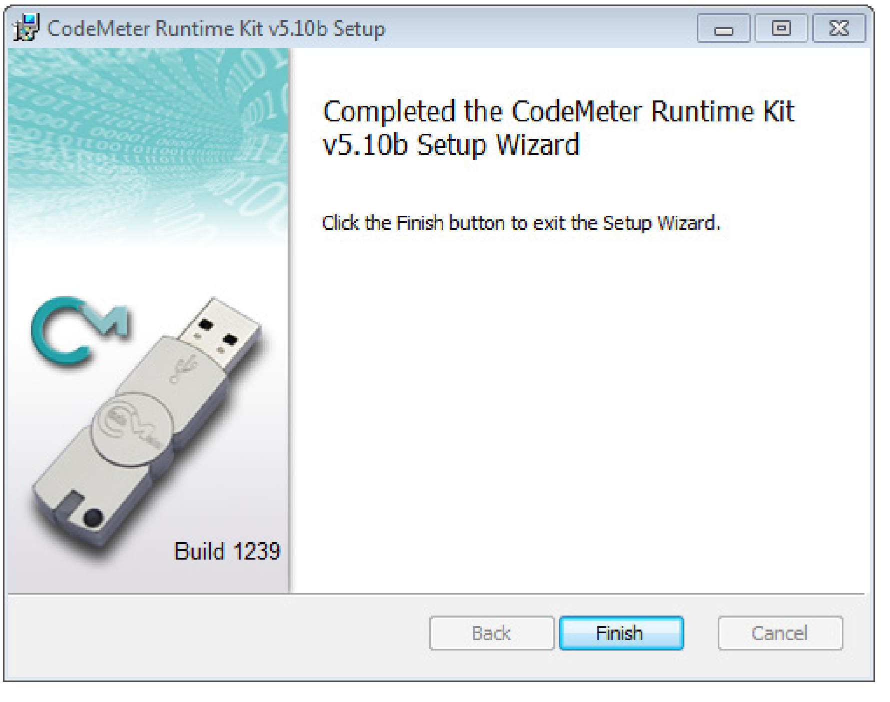
Select ***Next*** to continue

Ready to Install *Codemeter Runtime Kit Setup*

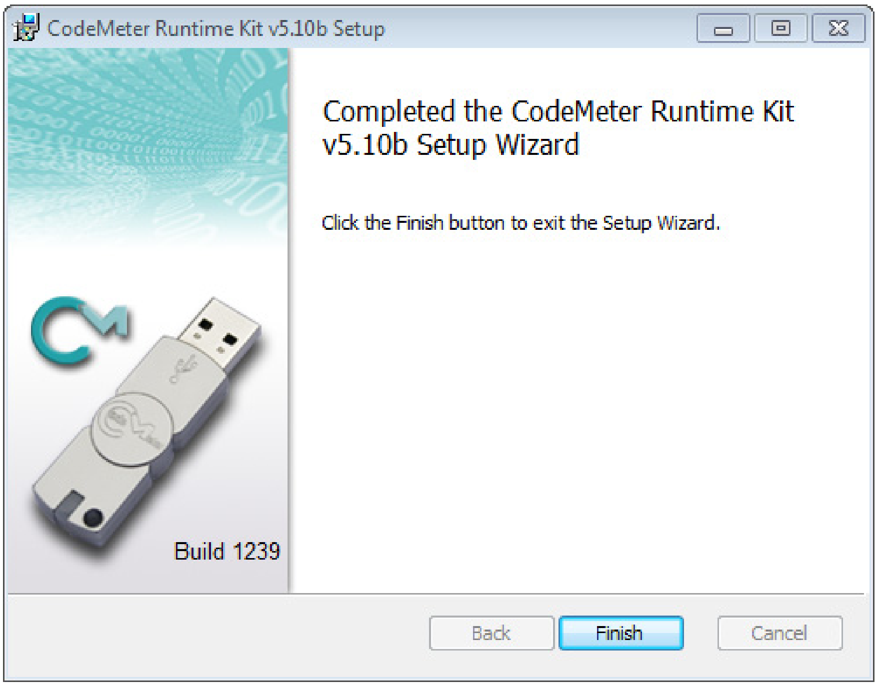


Select ***Install*** to continue

Installing *Codemeter Runtime Kit Setup*

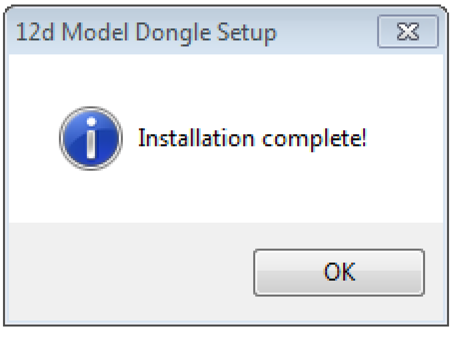


Completed *Codemeter Runtime Kit Setup*



Select ***Finish*** to end

The **Codemeter** dongle drivers have now been installed and the **12d Model Dongle Setup** screen will appear.



click ***OK***

The *12d Model Release* screen will then reappear.

Continue to Step 2. Check the dongle.

Step 2. Check the dongle

NOT YET UPDATED FOR V11

The 12d Model dongle (Wibu or Codemeter) can now be attached to the computer.

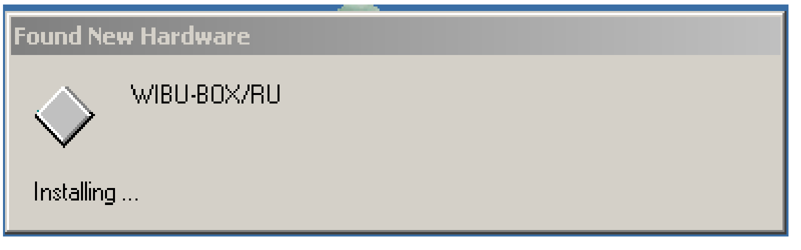
For USB dongles:

Warning: you must have System Administrations rights the first time you attach a USB dongle to any USB port:

When a USB dongle is attached for the first time to any USB port, Windows will detect that it is new hardware and needs to load the dongle drivers for that USB port - this will require System Administration rights.

Since the dongle drivers have already been loaded onto your computer, when the USB dongle is attached to a new USB port, it will be recognised as new hardware

and Windows should automatically install the correct dongle driver.



IMPORTANT NOTE IF THE WIBU DONGLE IS NOT RECOGNISED:

If the WIBU dongle is not recognised and you get the **Found New Hardware** screen

please go to the section on how to find the Wibu drivers

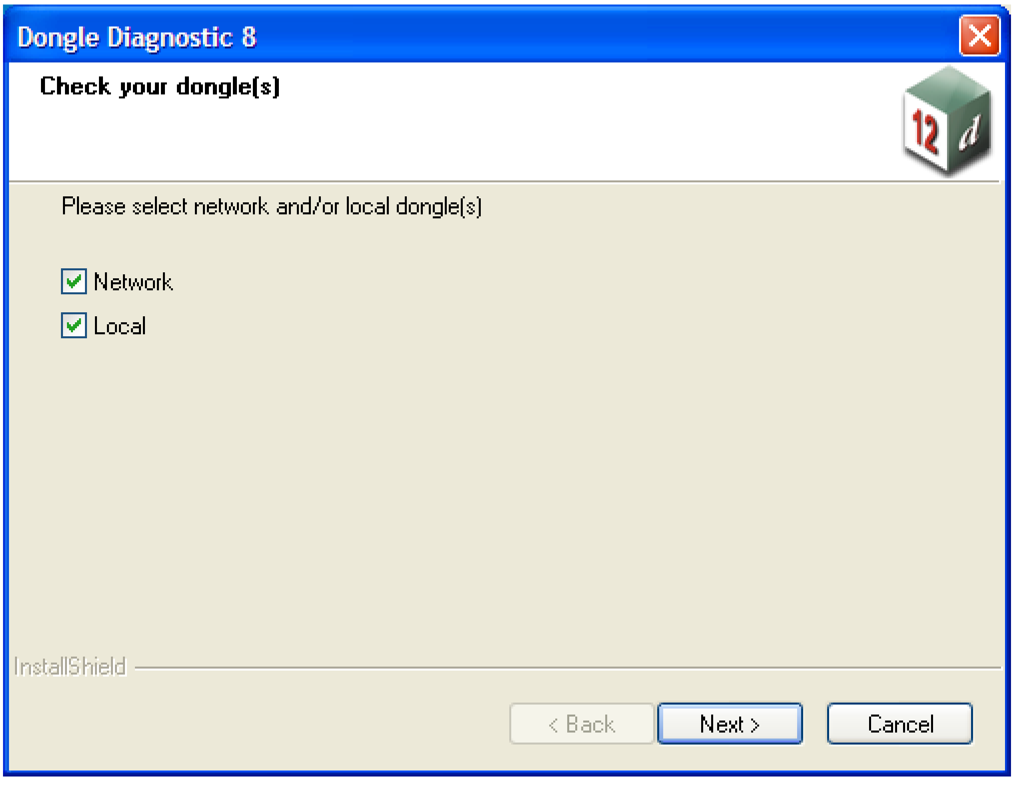
NOTE - even though your dongle is working on one USB port, if you try and attach the dongle to a new USB port, the dongle drivers will need to be installed for the new USB port. You will need System Administration rights to install the dongle drivers on the ***new*** USB port.

The dongle must be attached to the USB port at all times, otherwise 12d Model will stop running.

On the *12d Model Release* screen, click on *Check*:

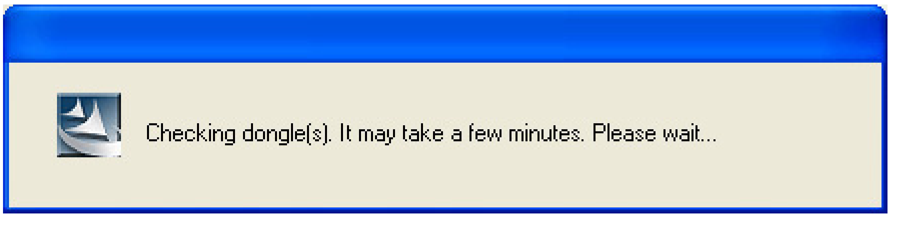
The dongle checking program begins.

Installation *Installation message*

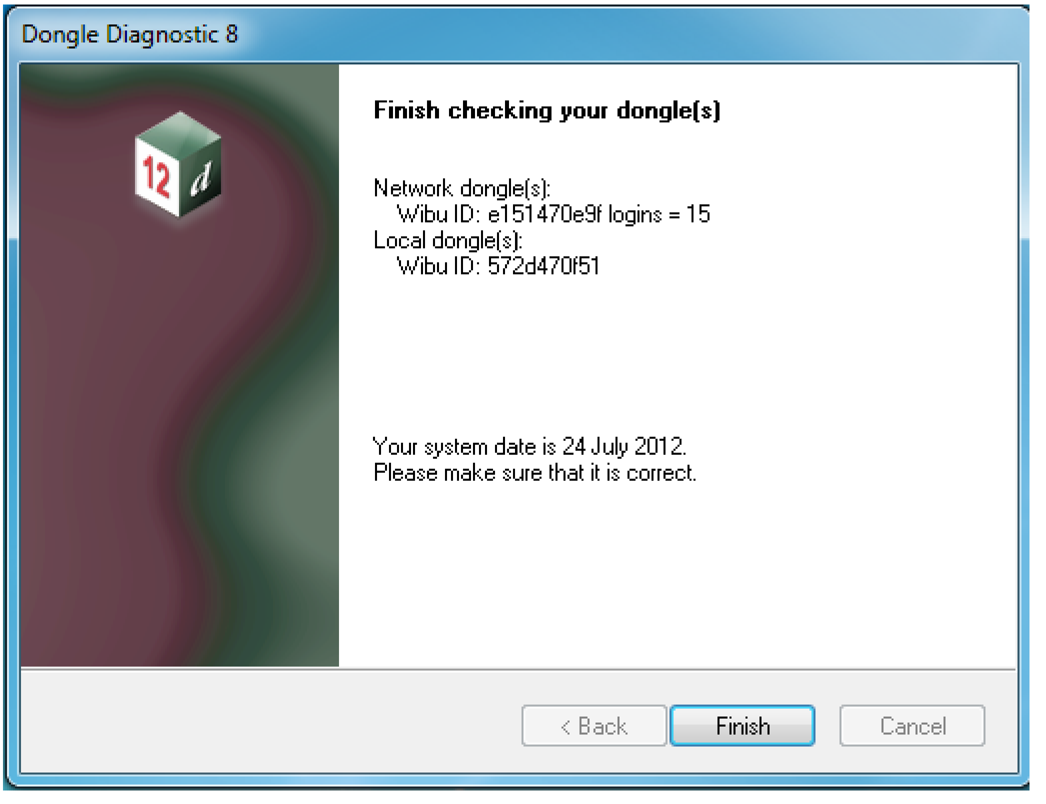


Tick **Local** and select ***Next***

The computer/network will be checked for *12d dongles* and also for your *System Date*.



Check Report



The correct dongle number should be displayed and also a check on the date in the computer is correct.

select ***Finish***

This completes the dongle *Check*.

The *12d Model Release* screen will then appear.

Continue to Step 3. Install Camtasia Codec.

Step 3. Install Camtasia Codec

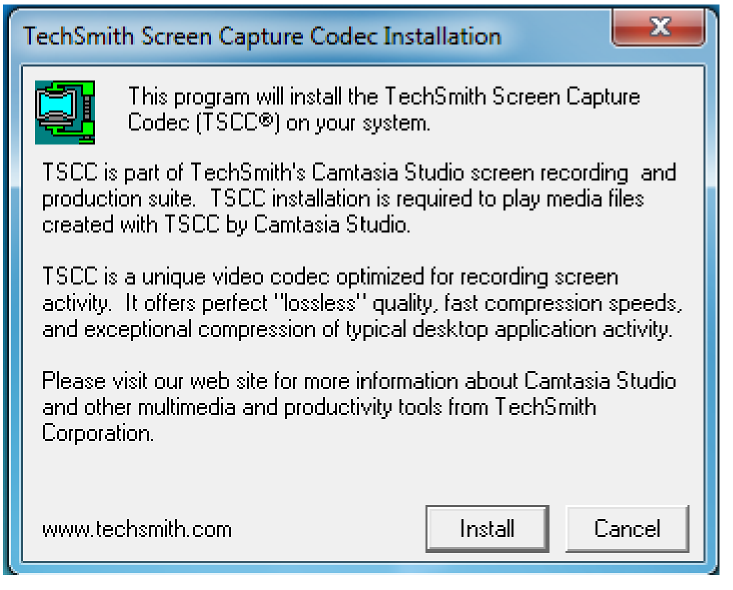
The Camtasia Codec is used for displaying the images on the *12d Model Training* CDs and DVDs.

You need to have System Administrator rights to install the Camtasia Codec.

On the *12d Model Release* screen, under *Install Camtasia codec*, click on *Camtasia Codec*:

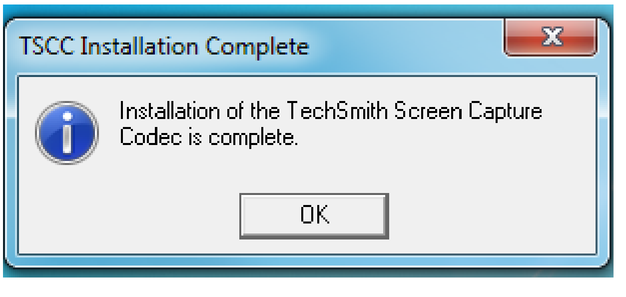
The Camtasia Codec installation begins.

Installation *Installation message*



Select ***Install*** to continue with the installation

Installation Complete



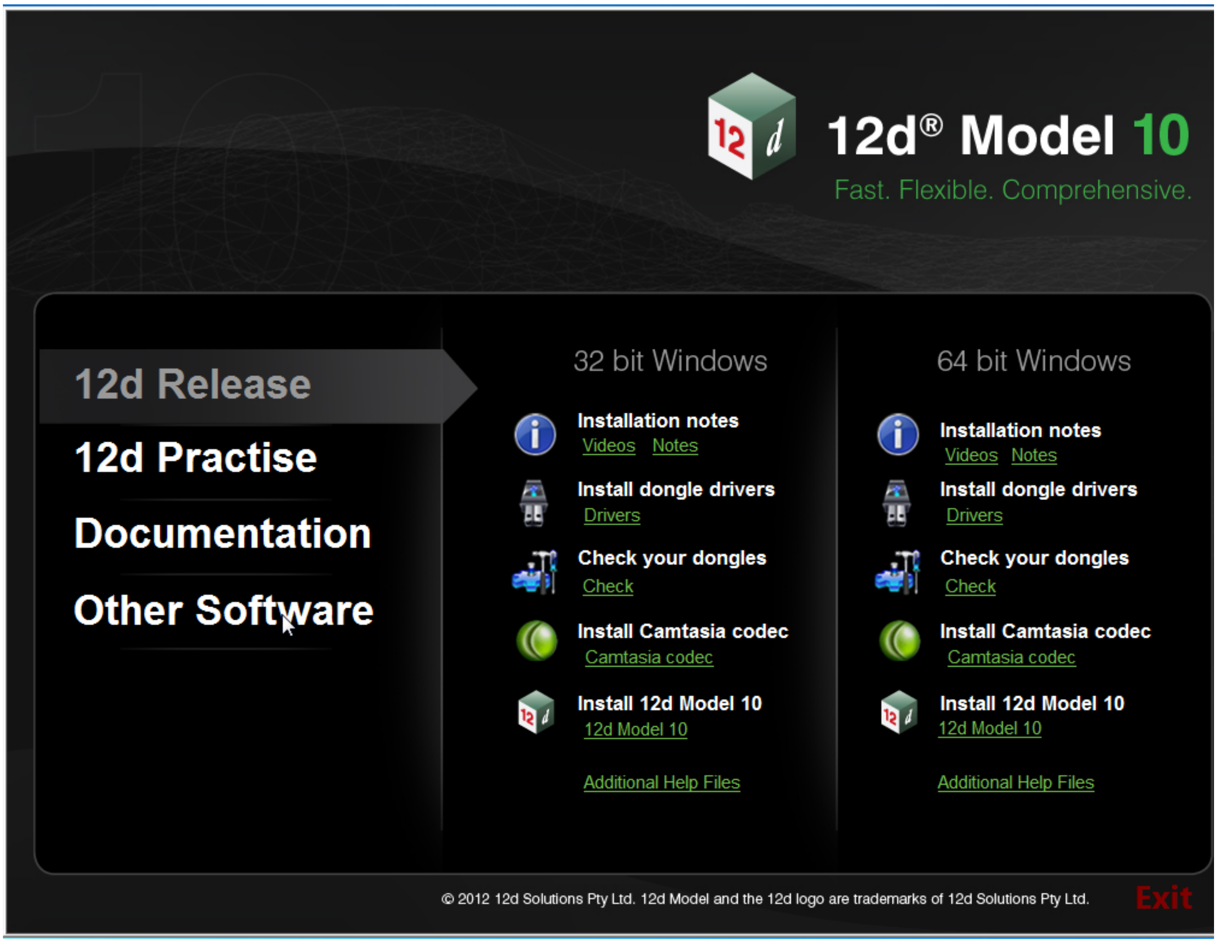
Select ***OK***.

This completes the installation of the *Camtasia Codec*.

The *12d Model Release* screen will then appear.

Continue to Step 4. Installing 12d Model 11 Release Program files.

Step 4. Installing 12d Model 11 Release Program files



For running 12d Model 11 on 32 bit Windows:

If you are running 12d Model on 32 bit Windows, then you need to install the *12d Model* 32 bit 12d.exe and 32 bit dongle drivers. So go down the options on the left hand side of the screen.

Note- the 32 bit 12d.exe will run point versions **up to 5M**. The 250M version of *12d Model* is only available as the 64 bit *12d Model* and it only runs on a Windows 64 bit operating system.

For running 12d Model 11 on 64 bit Windows:

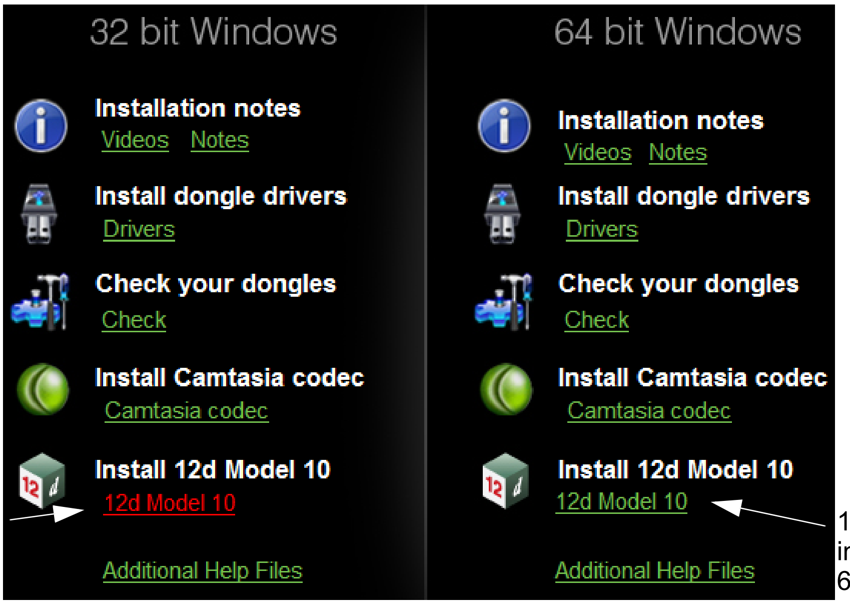
If you are running 12d Model on 64 bit Windows, then you need to install the *12d Model* 64 bit 12d.exe and 64 bit dongle drivers. So go down the options on the right hand side of the screen.

The 64 bit 12d.exe runs all point versions up to and including 250M.

So if you are installing on 32 bit Windows, then select the ***12d Model 11*** under ***Installing 12d Model 11*** under the ***left*** hand column.

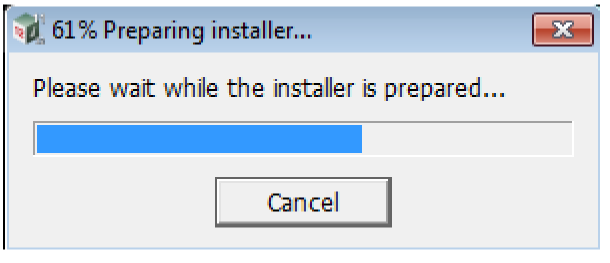
If you are installing on 64 bit Windows, then select the ***12d Model 11*** under ***Installing 12d Model 11*** under the ***right*** hand column.

12d Model 11
install for
32 bit Windows



12d Model 11
install for
64 bit Windows

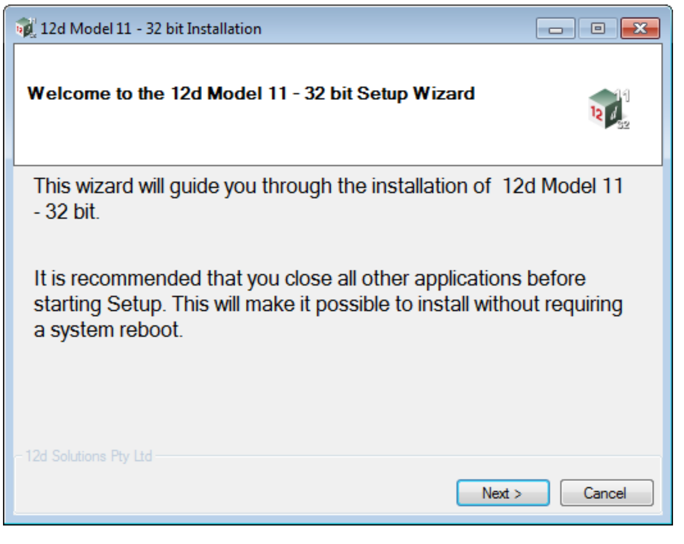
The ***12d Model 11*** installation begins.



Note - the following screens are for the 64 bit install but the 32 bit install is identical except the words **64 bit** are replaced by **32 bit**.

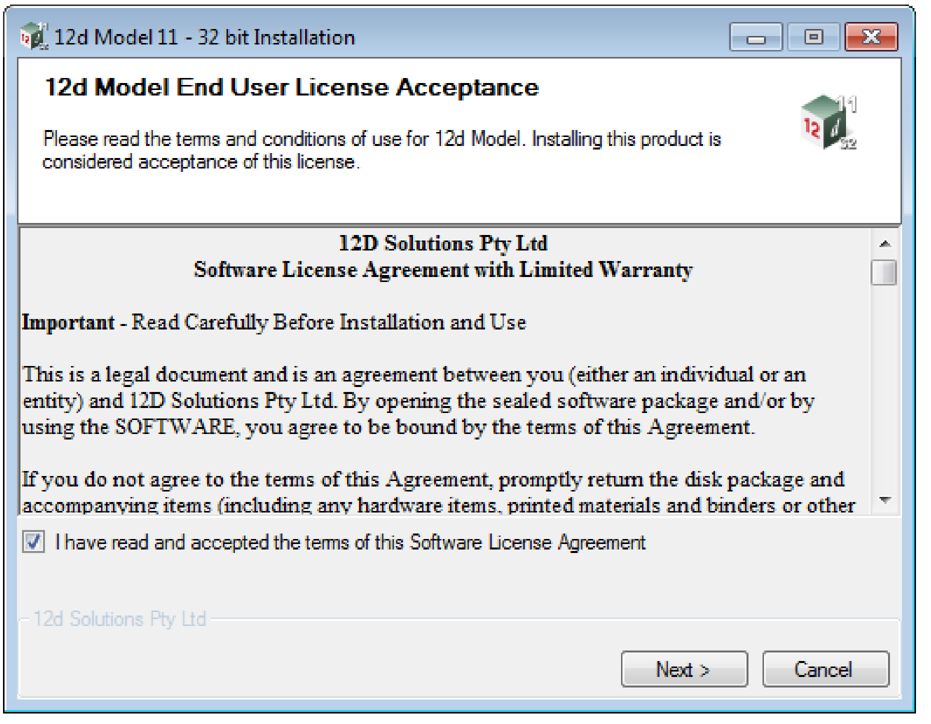
Welcome

Welcome message



Select ***Next*** to continue with the installation

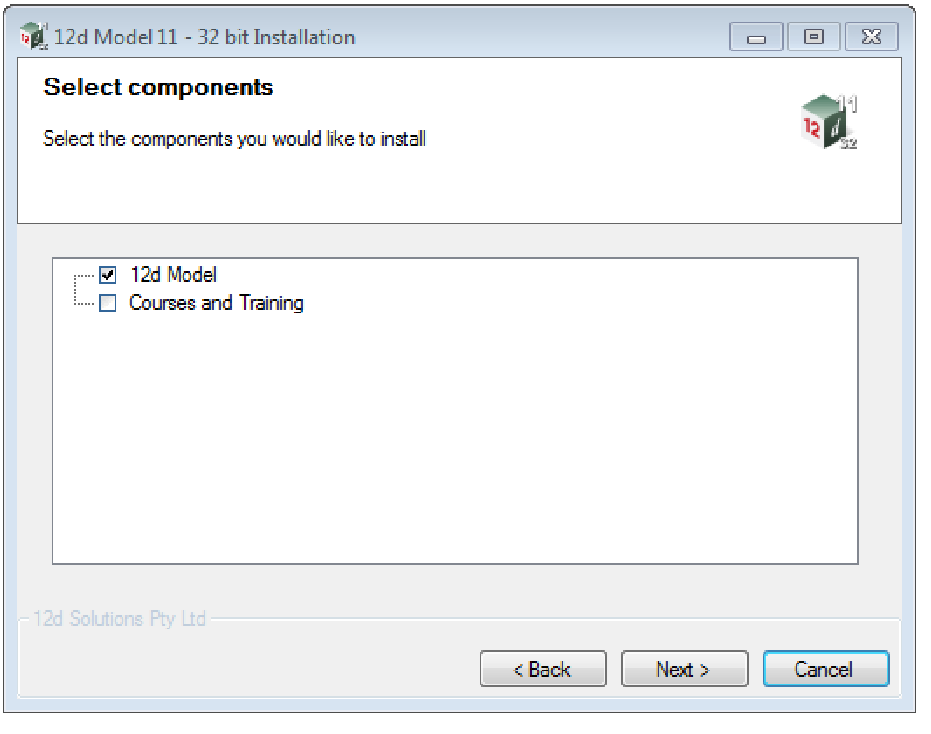
Software License Agreement *12D Solutions license agreement*



If you agree with the License conditions, click on *I have read and accept the terms of the Software License Agreement.*

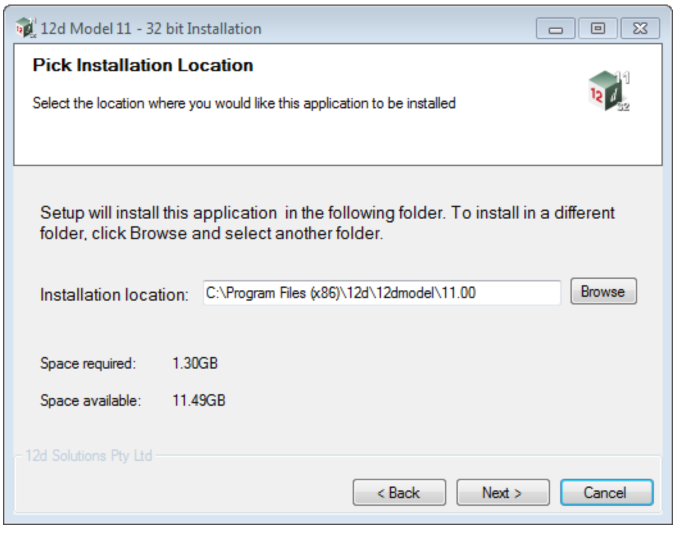
Select ***Next*** to continue with the installation

Select Components



select ***Next*** to continue with the installation

Installation Location



Continue with the default installation folder for the software:

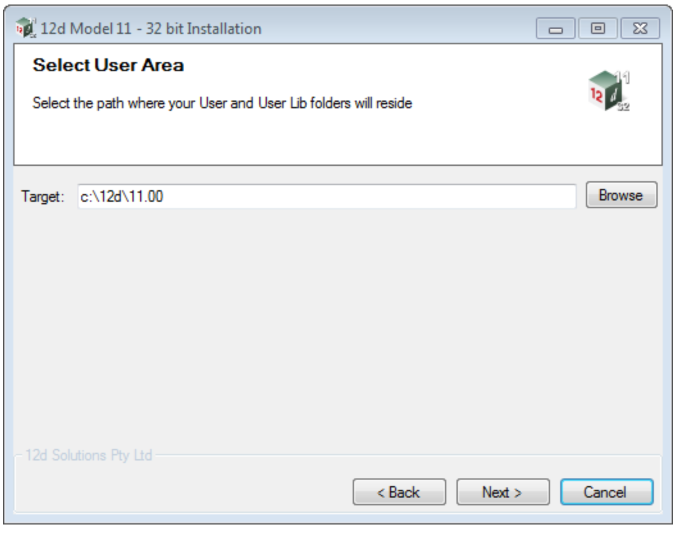
for 64 bit version: c:\Program Files\12d\12dmodel\11.0

for 32 bit version: c:\Program Files (x86)\12d\12dmodel\11.0

or click on ***Browse*** to browse to another folder for the installation

Select ***Next*** to continue with the installation

User Area



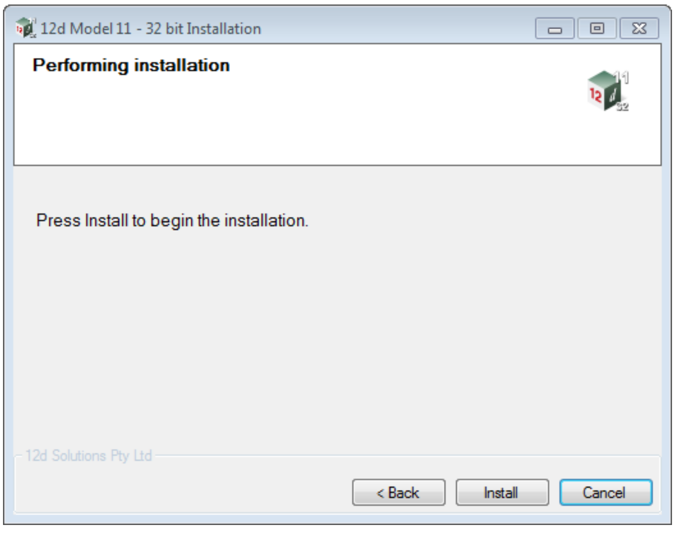
Continue with the default folder for the User Area for the software:

c:\12d\11.0

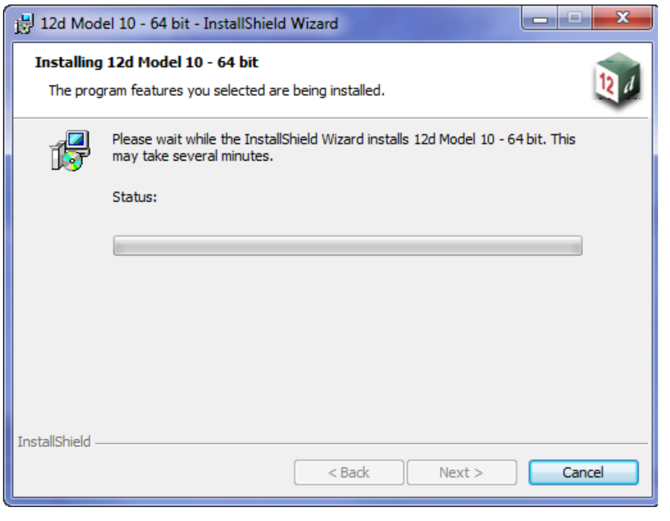
or click on ***Browse*** to browse to another folder for the User Area.

Select ***Next*** to continue with the installation

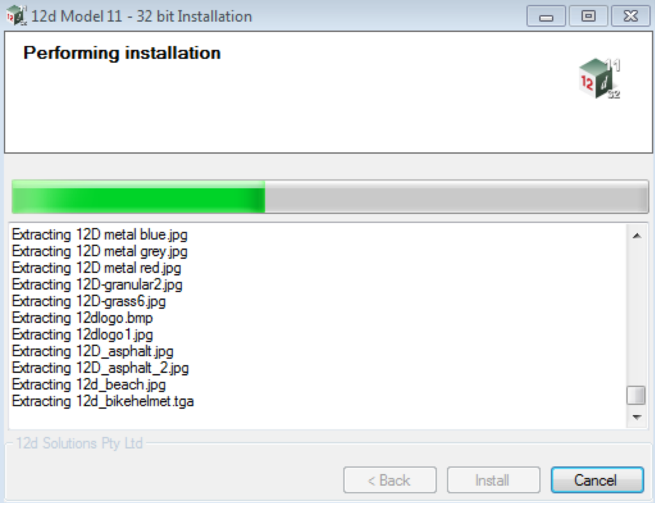
Ready to Install



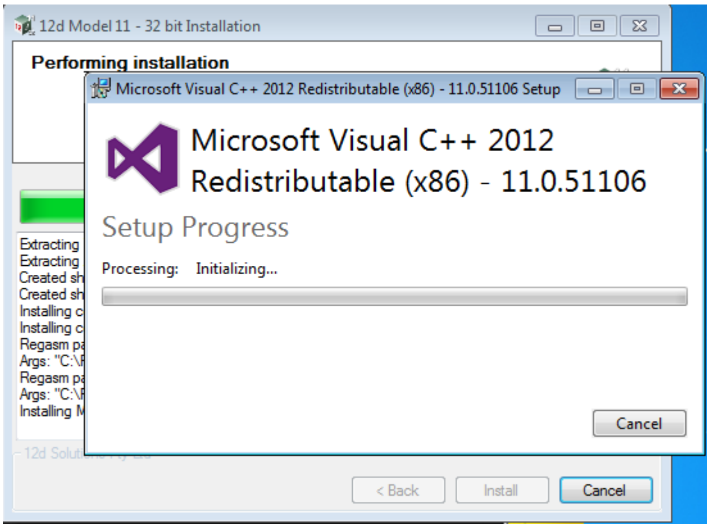
Select ***Install*** to begin the actual installation



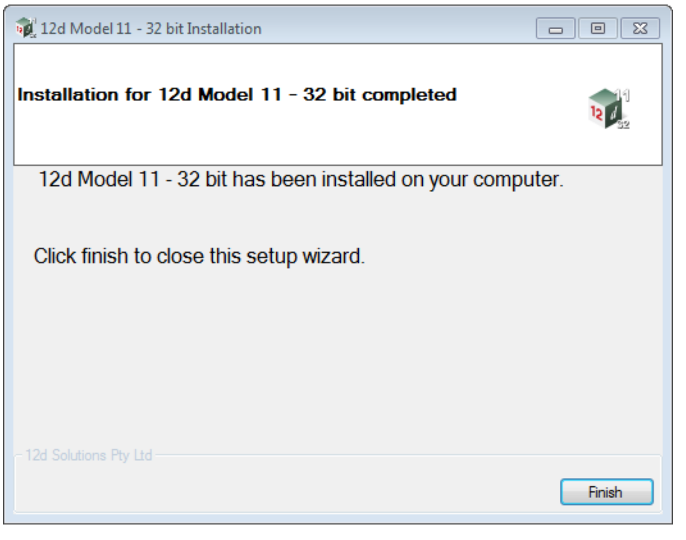
The software will be copied and installed onto the computer.



The Microsoft C++ Redistributables will be installed (32 and 64 for 64-bit install)

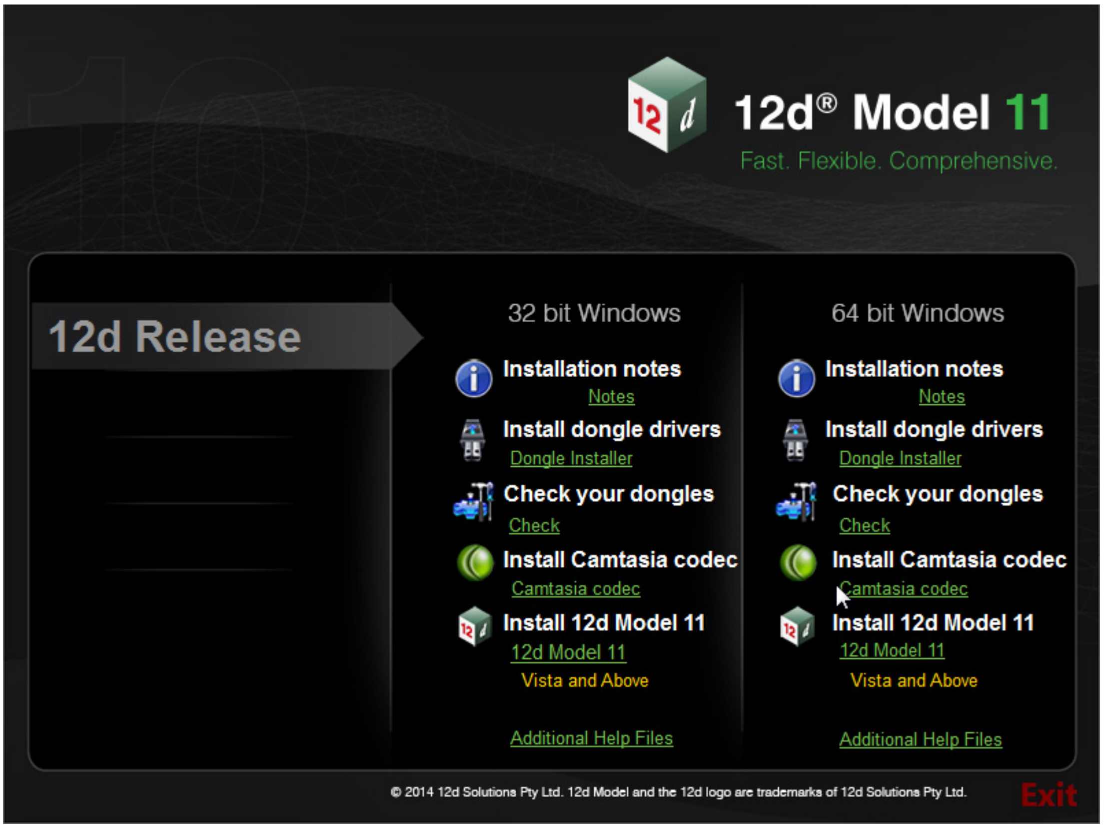


Setup Complete



End of installation. Select ***Finish*** to complete the installation

The *12d Model Release* screen will then appear.



Select **Exit** at the bottom right hand corner of the screen to end the installation.

This completes the installation of the ***12d Model*** software.

12d Model will not function without an authorisation file called ***nodes.4d***.

So the next step is to install and test the ***nodes*** file.

See

Folders Created by the Installation
Icons Created by the Installation,

Or go to Step 5. Installing the Nodes.4d file.

Folders Created by the Installation

(a) For 32 bit *12d Model* on Windows 32 bit operating systems

The *12d Model* installation loads the ***12d Model software*** into the folder

C:\Program Files\12d\12dmodel\11.00

OR

(b) For *64 bit 12d Model* on Windows 64 bit operating systems:

The *12d Model* installation loads the ***12d Model software*** into the folder

C:\Program Files\12d\12dmodel\11.00

OR

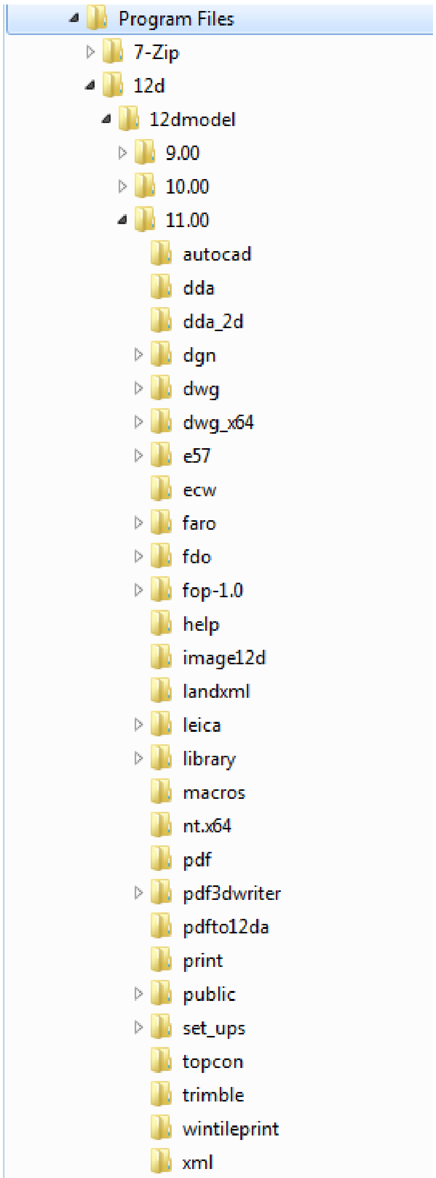
(c) For 32 bit *12d Model* 32 bit (for versions up to 5M) on Windows 64 bit operating systems:

The *12d Model* installation loads the ***12d Model software*** into the folder

C:\Program Files (x86)\12d\12dmodel\11.00

12d Model users normally only have ***read*** access to the *Program Files* and/or *Program Files (x86) areas*.

12d Model software area



under Program Files

The *12d Model* installation on either 32 bit or 64 bit WIndows, also creates an area

C:\12d\11.00

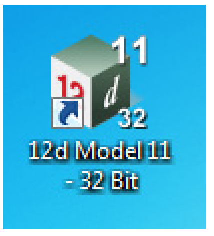
which during the installation is given read/write access for the user.

The folder 12d\11.00 contains the training data used with the *Getting Started for Design and Getting Started for Surveying* manuals *and the subfolders* ***User*** and ***User_Lib*** are created for future user customisations of *12d Model*

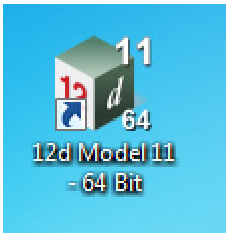
12d Model user areas

Icons Created by the Installation

The installation loads the appropriate components and creates the ***12d Model 11*** icon



icon for 12d Model 32 bit exe



icon for 12d Model 64 bit exe

The ***12d Model 11 - 32*** *and* ***12d Model 11 - 64*** icons fire up ***12d Model*** and attach to the folder *12d\11.00*.

Continue to Step 5. Installing the Nodes.4d file.

Step 5. Installing the ***Nodes.4d*** file

12d Model will not function without an authorisation file called ***nodes.4d***.

The information inside ***nodes.4d*** controls

(a) ***which dongles*** are authorised to run ***12d Model***

(b) what ***version*** of ***12d Model*** will run for a dongle

(c) what ***modules*** are authorised to run for a dongle

To authorise your ***12d Model***, two files ***nodes.12d11n*** and ***nodes.4d*** will have been emailed to you by your local ***12d Model*** Reseller.

Subject: 12D Auth file for 12D Solutions - Lee; Product 12d Model - 5c2d47013c -

Attachments: nodes.12d11n (11 KB)

Nodes file for : 12D Solutions - Lee
- 5c2d47013c -

The authorization file for 12d Model - is attached to this message.

The self-install procedure which follows is for this one workstation and requires 12dmodel installed.
After 12d Model version 11 is installed, please double click on the file attachment [nodes.12d11n] & to open it and 12d will guide you through.

If this succeeds you do not need to read further.

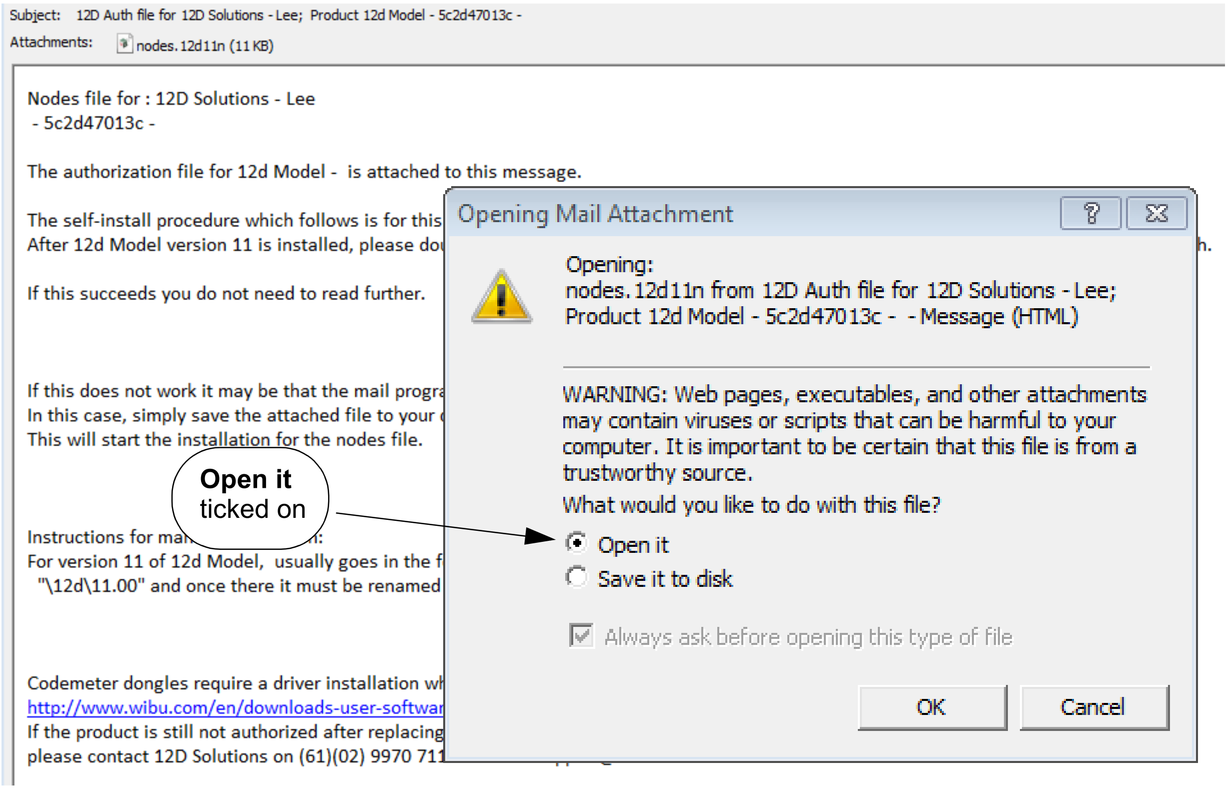
If this does not work it may be that the mail programme does not allow you to run attachments.
In this case, simply save the attached file to your desktop and open (from the right-mouse menu or by double-click).
This will start the installation for the nodes file.

Instructions for manual Installation:
For version 11 of 12d Model, usually goes in the folder
"\12d\11.00" and once there it must be renamed to "nodes.4d"

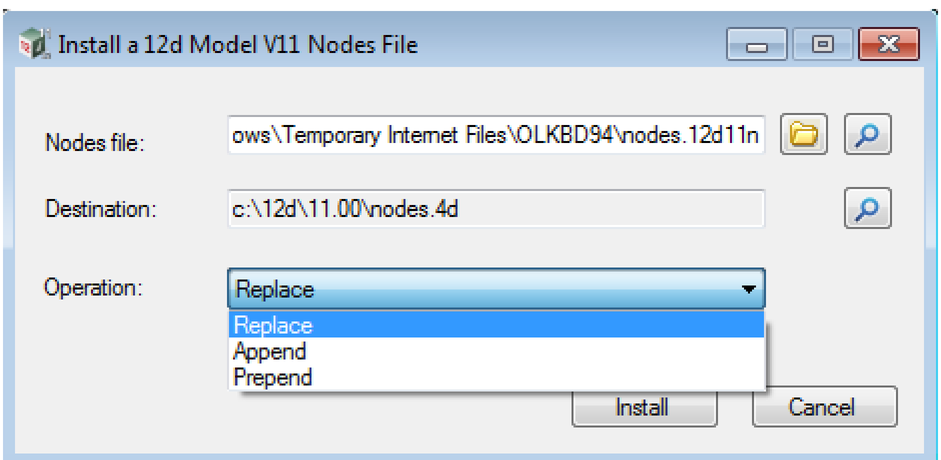
Codemeter dongles require a driver installation which can be obtained here
http://www.wibu.com/en/downloads-user-software.html
If the product is still not authorized after replacing or updating "nodes.4d"
please contact 12D Solutions on (61)(02) 9970 7117 or email support@12d.com

Installing ***12d Model*** sets up a *Windows* file association so that a ***nodes.12d11n*** file is recognised by ***12d Model*** and automatically installs a ***nodes.4d*** file in the correct location.

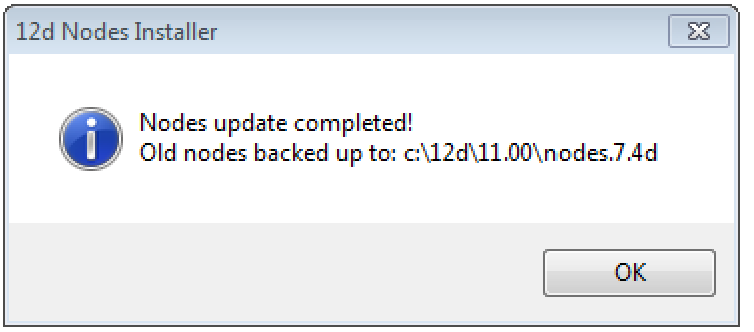
In the email, double click on the files ***nodes.12d11n***, click on ***Open it*** and then click on ***OK***



The **Accept Nodes** panel then appears



Select ***Replace***, ***Append*** *or* ***Prepend*** and then click **Install**.
12d Model saves the ***nodes.4d*** file to the folder ***C:\12d\11.00***.



Note - if you don't have one or both of the files ***nodes.4d*** or ***nodes.12d11n*** attached to an email but instead have either of the files in a folder accessible from your computer, then you can do the following:

If there is a ***nodes.12d11n*** file, double click on that file and the **Accept Nodes** panel should appear and you continue with the instructions above.

If that doesn't work or you only have a ***nodes.4d*** file, start up ***12d Model*** and click on the **Nodes** button at the bottom of the **Project Selection** panel. This will also bring up the **Accept Nodes** panel and you can browse for the ***nodes.4d*** or ***nodes.12d11n*** file in the New nodes file panel field. You can then continue with the instructions above.

Continue to Step 6. Installing 12d Model Context Sensitive Help for Vista and Win 7.

Step 6. Installing 12d Model Context Sensitive Help for Vista and Win 7

12d Model has a context sensitive reference help system which can be accessed directly from most menus and panels:

(a) For most menus in ***12d Model***, when the menu is on the screen and has the Windows focus, simply press F1 and the ***12d Help*** will open automatically at that menu

(b) For most panels in ***12d Model***, when the panel is on the screen and has the Windows focus, simply press F1 and the ***12d Help*** will open automatically at that panel

(c) Most panels in ***12d Model*** have a **Help** button on them and clicking on the **Help** button opens the ***12d Help*** at that panel

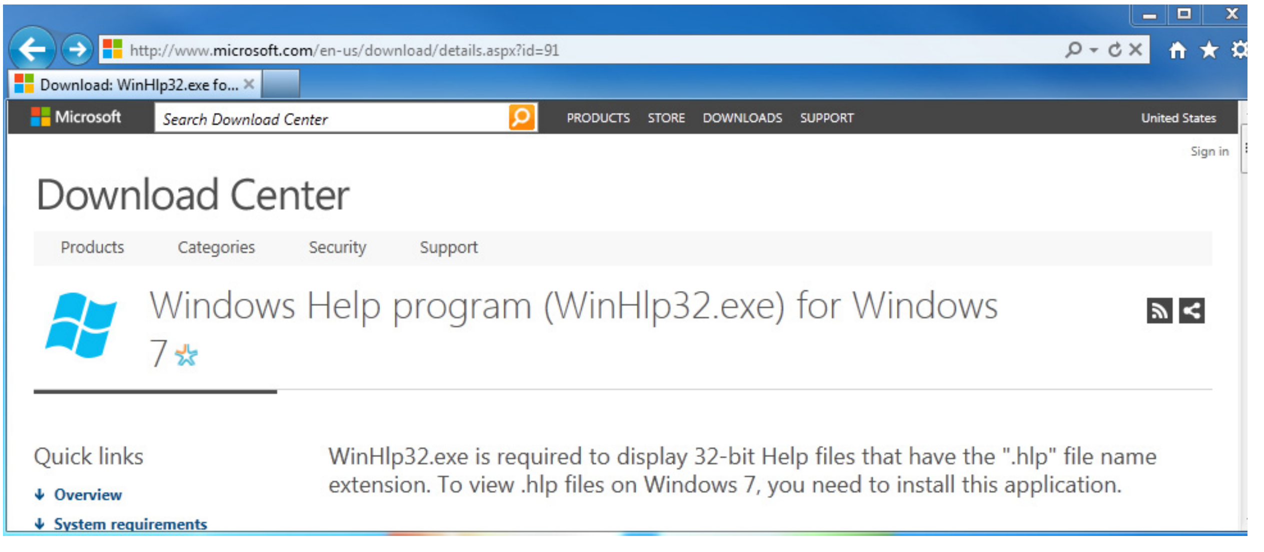
(d) By clicking on the option ***Help =>12d Model***

The ***12d Model*** context sensitive help uses Microsoft's ***WinHlp***.

For *Vista* and *Win 7*, Microsoft no longer ships the WinHlp.exe executable and Microsoft will no longer allow it to be installed by the 12d Model installation DVD.

So if you are running Vista or Win 7, you need to download and install WinHlp.exe from Microsoft's website to access the ***12d Model*** *Help* files. The link to the area for the download is

www.microsoft.com/en-us/download/details.aspx?id=91



Or if you have 12d Model up and running, the link is available on the menu

Help =>Microsoft 7

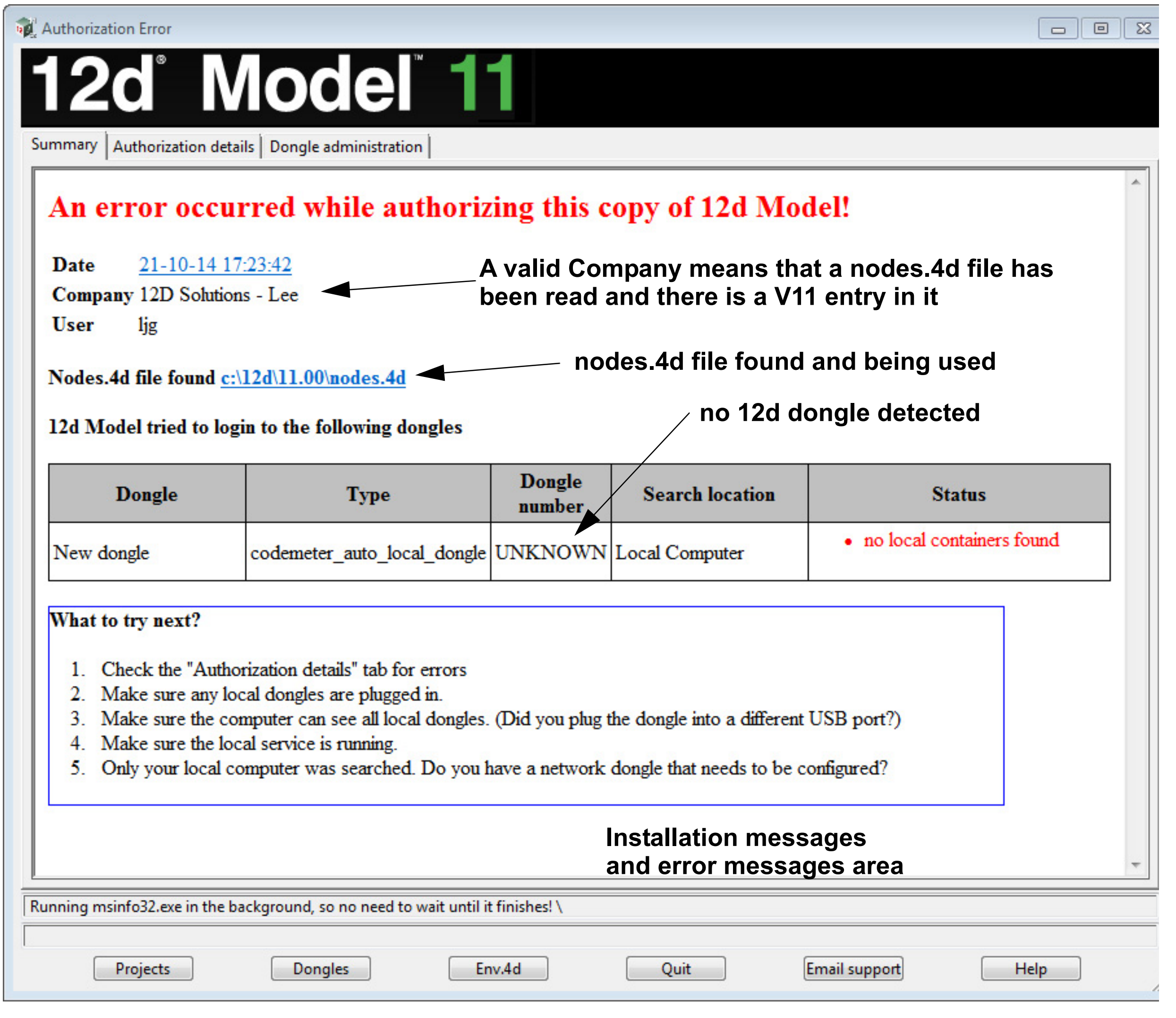
Note: the entire ***12d Model Reference*** manual is available in pdf format on the ***Documentation*** tab of the ***12d Model 11 Installation*** DVD.

CONGRATULATIONS - 12d Model has now been successfully installed and you have finished with these notes.

If ***12d Model*** did ***not*** install correctly, please continue to the next section, 2.3 12d Model 11 Not Authorising, of these notes.

2.3 12d Model 11 Not Authorising

If there is an ***error*** with the installation, then the **Error Authorizing Release Version** panel appears with possibly error message in the panel message area.



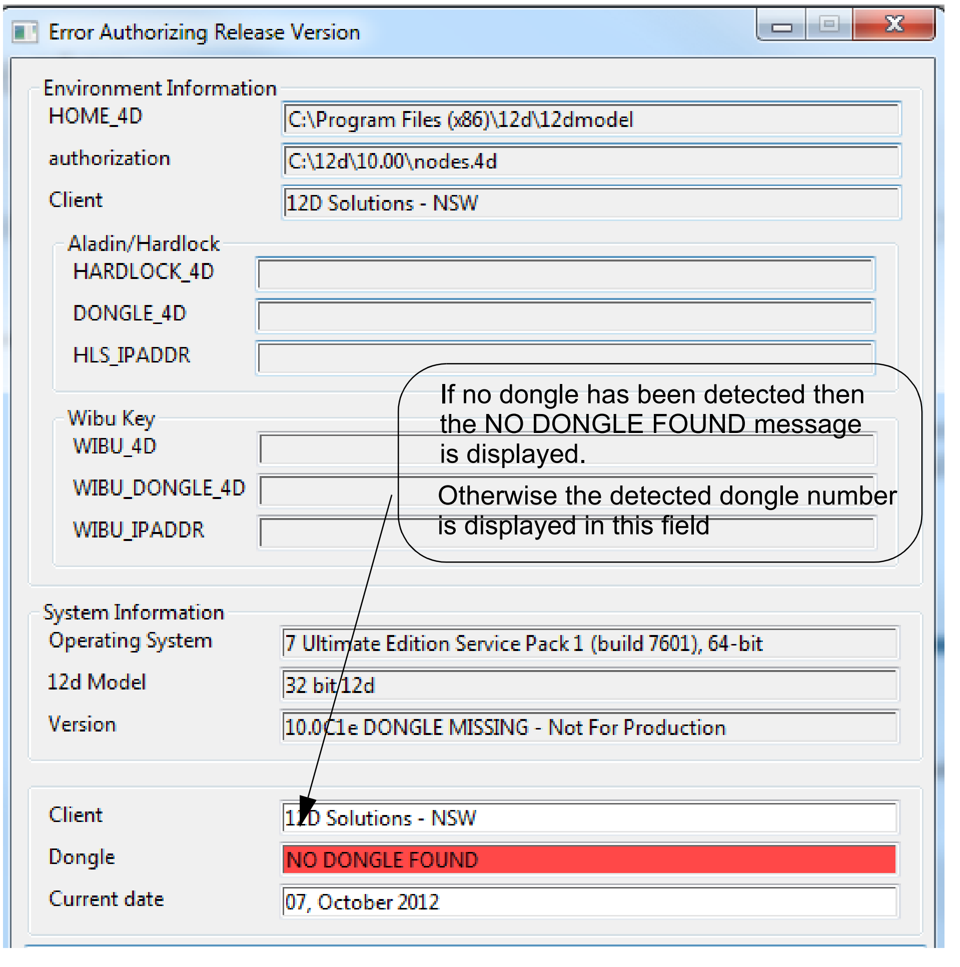
If there is no error message on the panel or it is still not obvious what the problem is, please check the list of possible errors in the next section 2.4 Possible Problems When Authorizing the Release Version.

For example, in the above panel, the DONGLE UNKNOWN shows that no ***12d*** dongle has been detected.

2.4 Possible Problems When Authorizing the Release Version

NOT YET UPDATED FOR V11

1. The ***12d dongle number*** does not come up in the ***Dongle*** field of the **Error Authorizing Release Version** panel.



Please check that

(a) the ***12d*** dongle is firmly attached to the computer.

If not, please attach the dongle and retry starting up ***12d Model*** and accessing a project.

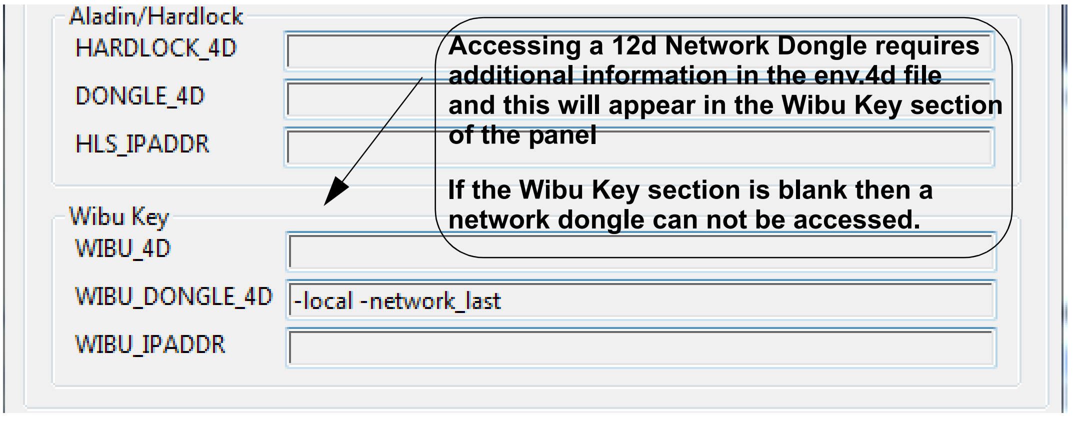
(b) if it is a parallel dongle, the dongle is before any Rainbow dongles. For example, before any AutoCAD dongle.

(c) your computer has the latest dongle drivers installed. These can be obtained from *Updates* section of the web site www.12d.com.

If the dongle number still does not come up in the **Error Authorizing Release Version** panel, please click on the **Email info** button to bring up the **Email Information on 12d Model to 12D Solutions** panel. Fill in the panel and click on **Email** to send the information to 12D Solutions and then contact 12d Solutions or your local distributor.

Special Note If Using a Network Dongle

If you are using a network dongle, then your ***env.4d*** needs to be modified so that ***12d Model*** will correctly ***search*** for the Network dongle. In that case there will be information in the ***Wibu Key*** section of the **Error Authorizing Release Version** panel.

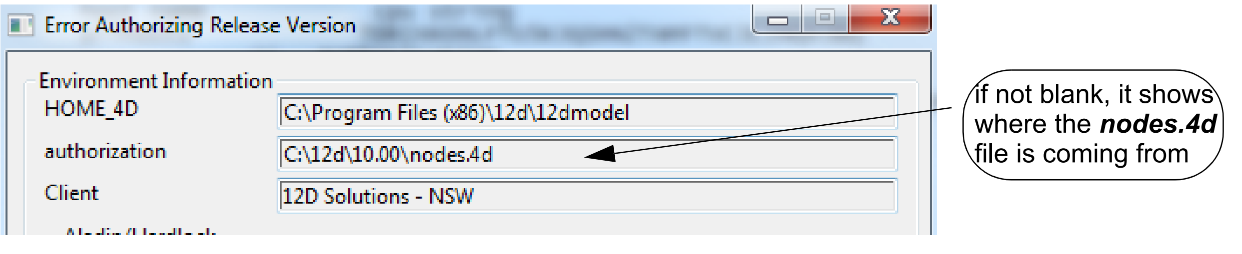


If the Wibu Key section is blank then a network dongle will not be searched for and so cannot be accessed.

Please see the notes on *Installing a Network Dongle* under the ***Documentation*** tab of the ***12d Model 11 Installation*** DVD on how to set up the env.4d file correctly for accessing a network dongle.

2. Check if there is a recent ***nodes.4d*** or ***nodes.12d11n*** file in the correct area on your computer, which by default is in the folder *C:\12d\11.00.*

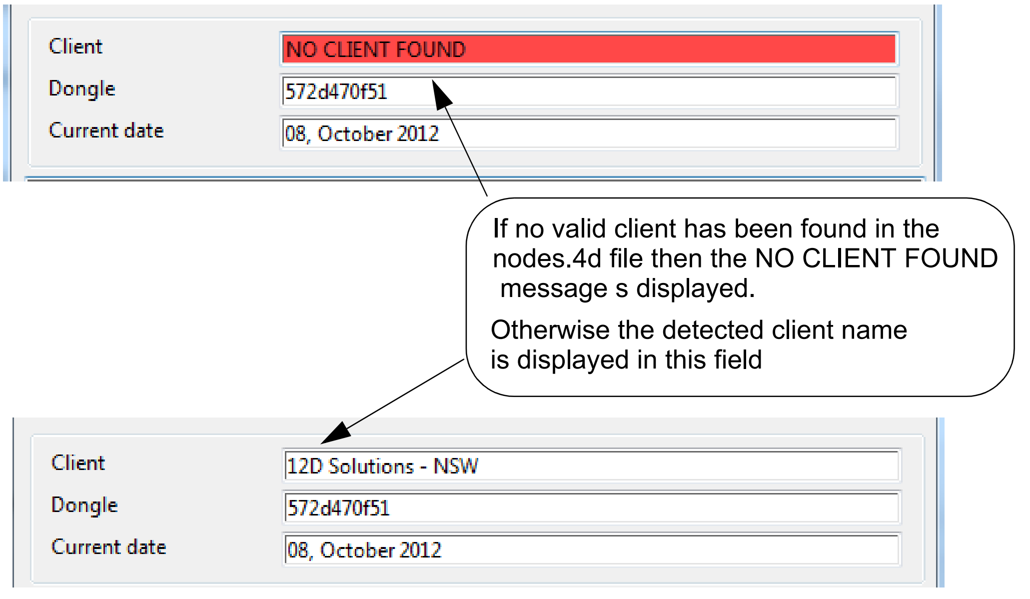
   The nodes.4d file found is displayed in the **Authorization** field.



   If it is blank then no nodes.4d has been found.

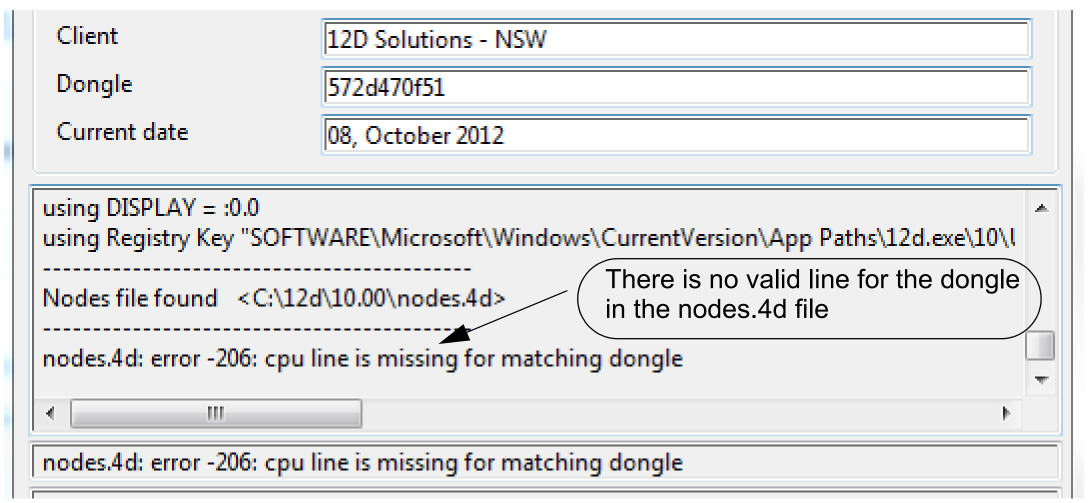
   If you do not have a ***nodes.4d*** or ***nodes.12d11n*** file, contact your local distributor.

3. The ***Client*** field in the **Error Authorizing Release Version** panel has NO CLIENT FOUND.



If the ***Client*** field has a valid *Client Name* in it then that indicates that a 1***2d Model 11 nodes.4d*** file is being read and has a valid Client line in it.

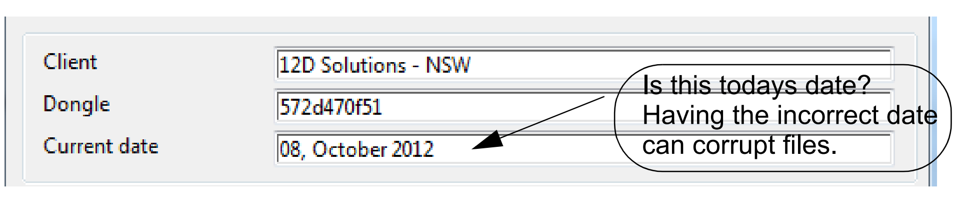
4. Client and Dongle fields look correct but you now get an **error -206**



This error now indicates that there is **no** line, or no valid line, in the ***nodes.4d*** file for the dongle number given in the **Dongle** field.

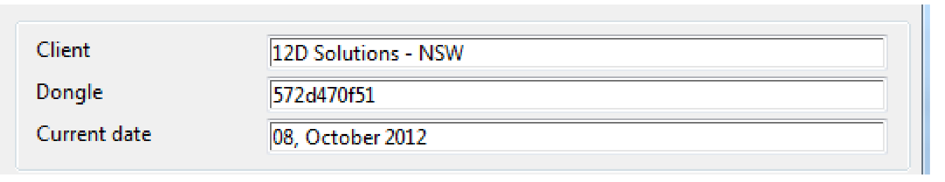
If you do not have the correct *nodes.4d* file, contact your local distributor.

5. Check that the date on your computer is correct.



If the date is not correct then be very careful. Having an incorrect date in your computer can corrupt computer files and ***12d Model*** projects.

6. If the date on your computer is correct then you need to check that it is between the *start* and *end dates* given in the line in the *nodes.4d* file for your dongle.



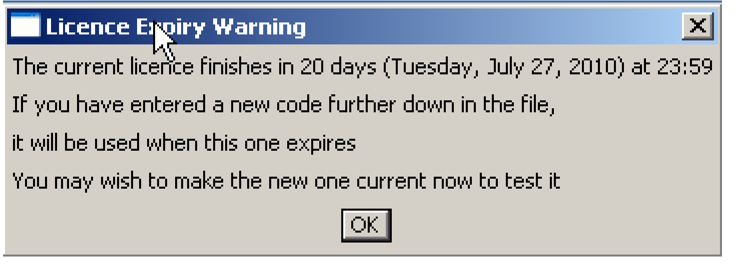
   If there is no valid date line in your nodes.4d file for your dongle, then contact your local distributor for a new nodes.4d file.

7. ***12d Model*** still can't access a project.

   Please contact your local Reseller.

2.5 Licence Expiry Warning

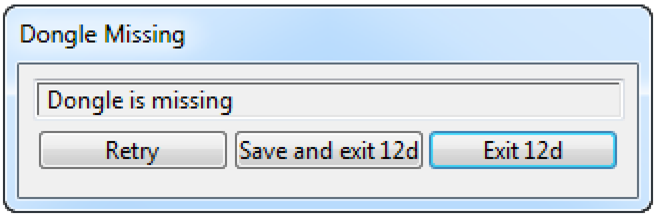
The ***nodes.4d*** file that controls *12d Model*, only validates *12d Model* to run for a certain period of time.
If when running ***12d Model***, the **Licence Expiry Warning** panel appears:



it simply means that the ***nodes.4d*** file needs to be updated.
Please contract your ***12d Model*** *Distributor* to obtain a new *nodes.4d*.

2.6 Dongle Missing

If when running ***12d Model***, the **Dongle Missing** panel appears:



it simply means that the ***12d*** dongle can no longer be seen by ***12d Model***.

Please check that your ***12d*** dongle is still attached to your computer, or if you are using a network dongle, that your network is still active.

Even if you can not see the dongle, ***12d Model*** will allow to save your current work do nothing is lost.

3 Installation of 12d Model 10 Practice Version

The *Practise* version is limited to a maximum of 5,000 points and creates projects that cannot be accessed by the *Release* versions of ***12d Model*** and vice-versa. However the *Practise* version can be used free of charge by 12d Solutions customers and registered *Practise* Users.

These notes are for installing the ***Practise version*** of ***12d Model 10.*** There are separate notes for installing the *Release* version.

The *Practise* version must be Registered with 12d Solutions once it is installed on a computer. A new Registration is required for each computer that the Practise version is run on.

The *12d Model 10 Installation CD* can be used to install the *Practise* versions of ***12d Model 10***.

Or

The file *12dModel 10 Practise.exe* can be down loaded from the web site www.12d.com. The file *12dModel 10 Practise.exe* is a self-extracting file for installing the *Practise* version of ***12d Model***.

Extra Notes for All Installations

1. A three-button mouse is essential.
2. The *12d Model Installation CD* contains extra information other than just the installation version of *12d Model*. For example, extra documentation, course notes and source code to macros.

   The extra data can be copied from the *12d Model Installation CD* but the copied files may only have a “read only” attribute set. This means that the files can not be edited or modified in any way.

   To change the attribute so that a file can be modified, select the file in Explorer, bring up the Properties sheet and under the General tab sheet change the “read only” box so that it is not ticked on.

   Selecting ***OK*** or ***Apply*** will then modify the attribute of the file.

3.1 Installing the Practise Version of 12d Model 10

These notes are for installing the **Practise** version of 12d Model 10. There are separate notes for installing the **Release** version.

A. Disk Space required for installing 12d Model 10

Approximately 200 megabytes of disc space will be required for the installation to succeed. After installation this can be reduced to a minimum of 60 megabytes.

B. Before installing from the DVD or the Web

For Windows 2000, XP, Vista, Windows 7:

Reboot the PC **before** installing 12d Model from the DVD or the Web.

Your login must have Administrator privileges.

C. Installing 12d Model from the Web

The self-extracting file *12dModel 10 Practise.exe* can be down loaded from the web site www.12d.com. *and* used to install the *Practise* version of *12d Model.*

After down loading *12dModel 10 Practise.exe*, double-click on *12dModel 10 Practise.exe* and the installation software will be automatically extracted from the file and fires up the ***Install 12d Model Practise*** screen.

Please go to section Step 3. Dialogues from the 12d Model 10 Practise InstallShield Wizard:

D. Installing Using the 12d Model 10 Installation DVD

Insert the ***12d Model 10 Installation DVD*** into the CD drive.

On inserting the CD, the ***12d Model*** Setup program automatically begins.

If it doesn’t, simply double click on the program ***Launch.exe*** from the DVD.

The ***12d Model Release*** front screen will appear.

Click on the ***12d Practise*** tab

The ***12d Model Practise*** screen will appear.

E. 12d Model Practise Screen

The ***12d Model Practise*** screen:

Note - when installing from the web, the *tabs* with *12d Release*,*12d Practise*, *Other Software* and *Documentation* will not appear. The extra tabs only appear if you are installing from the *12d Model Installation DVD*.

The Steps on the ***12d Practise*** *tab* of the*12d Model Installation DVD* will lead you through the installation of the *Practise* version of *12d Model.*

A ***video*** to guide you through the installation process can be viewed by clicking on ***Videos*** under the ***Installation notes*** heading. Or the ***Installation Notes*** in pdf format can be viewed by clicking on ***Notes*** under the ***Installation notes*** heading.

NOTE - If the *Camtasia* codec is already installed on your computer then you can proceed to ***Install 12d Model 10 Practise.***

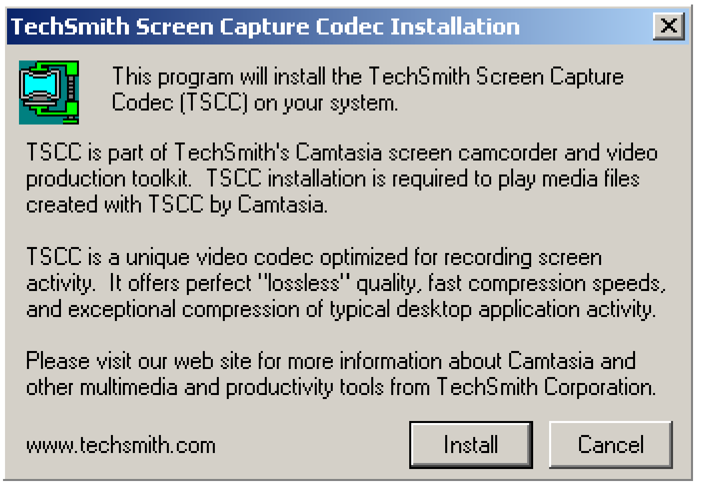
Step 1. Install Camtasia Codec

Install the Camtasia Codec that is used for displaying some of the videos on the ***12d Model Training CD***s and DVDs.

Note - you need Administrator rights to install the Camtasia Codec.

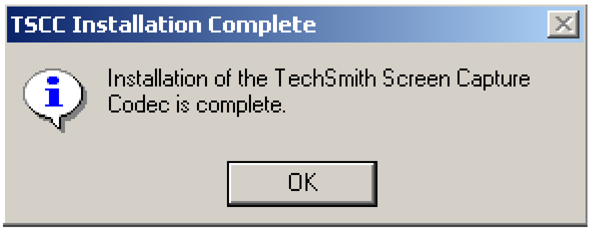
Click on *Camtasia codec*:

Installation *Installation message*



Select **Install** to continue with the installation

Installation Complete



Select **OK**.

This completes the installation of the *Camtasia Codec*.

The ***12d Model Practise*** screen will then appear.

Step 2. Install 12d Model 10 Practise From DVD

Installing the 12d Model Practise software and Training area:

Note - you need Administrator rights to install ***12d Model 10 Practise***.

Click on *12d Model 10 Practise*:

The installation will then start.

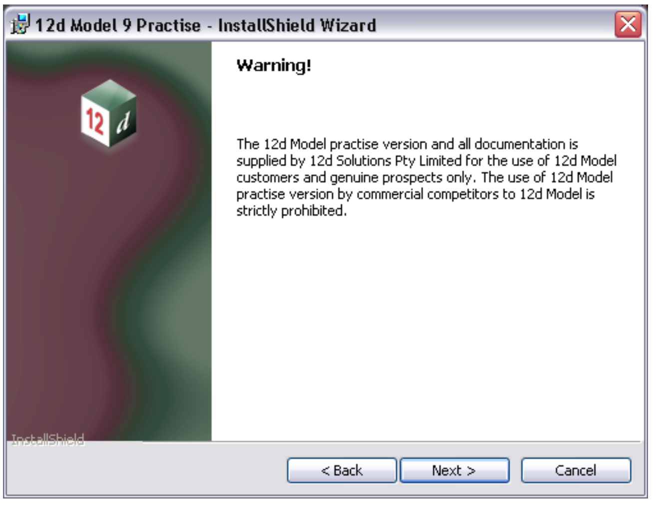
Step 3. Dialogues from the 12d Model 10 Practise InstallShield Wizard:

Welcome *Welcome message*

Select **Next** to continue with the installation

Warning for commercial competitors

12d Solutions warning



The ***12d Model Practise*** version and all documentation are supplied by *12d Solutions Pty Limited* for the use of ***12d Mode***l Customers and genuine prospects only. The use of the ***12d Model*** practise versions by commercial competitors to ***12d Model*** is strictly prohibited.

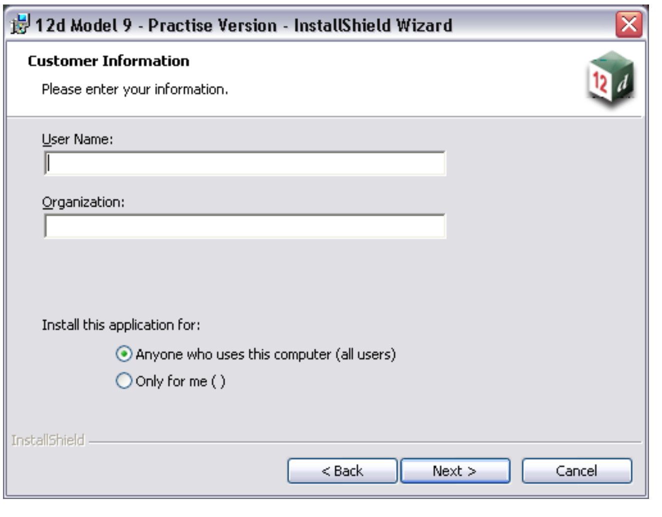
Select **Next** to continue if you are a ***12d Model*** Customers or a genuine prospect

Software License Agreement *12d Solutions license agreement*

If you agree with the License conditions, click on *I accept the terms in the license agreement*

Select **Next** to continue with the installation

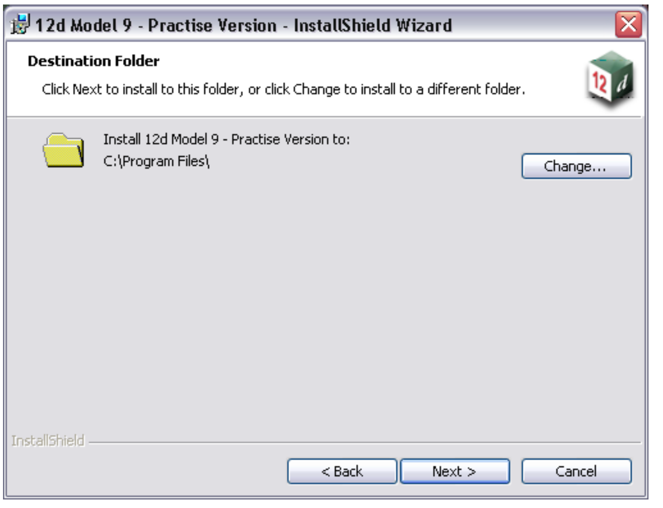
Customer Information



fill in *User Name* and *Organisation*, tick who can use the computer

select ***Next*** to continue with the installation

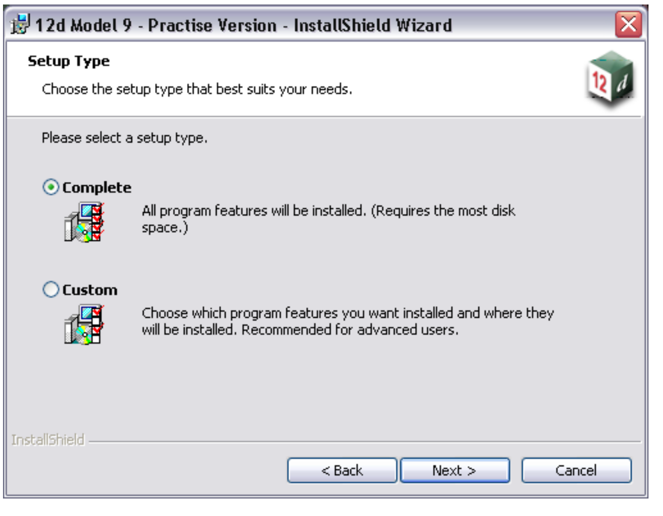
Destination Folder



continue with the default installation area for the software *c:\Program Files* or click on *Change* to browse to another area for installation

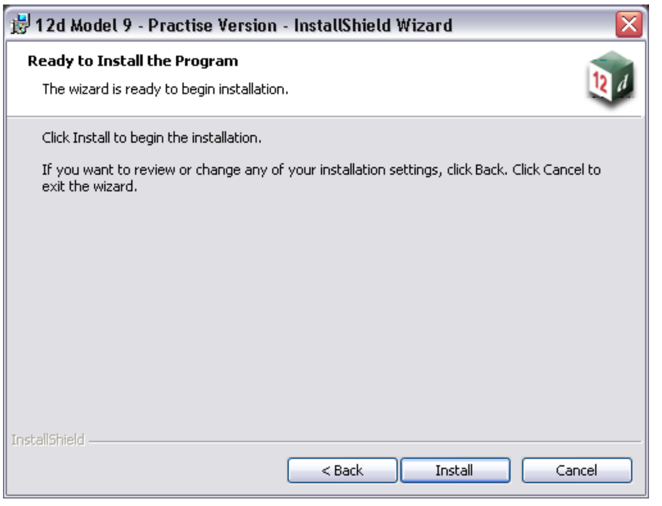
select ***Next*** to continue with the installation

Setup Type



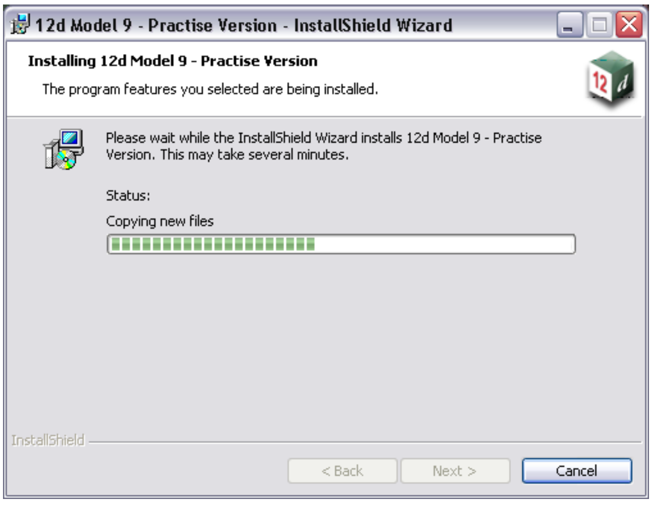
select **Complete** select **Next** to continue with the installation

Ready to Install *Begin the installation*

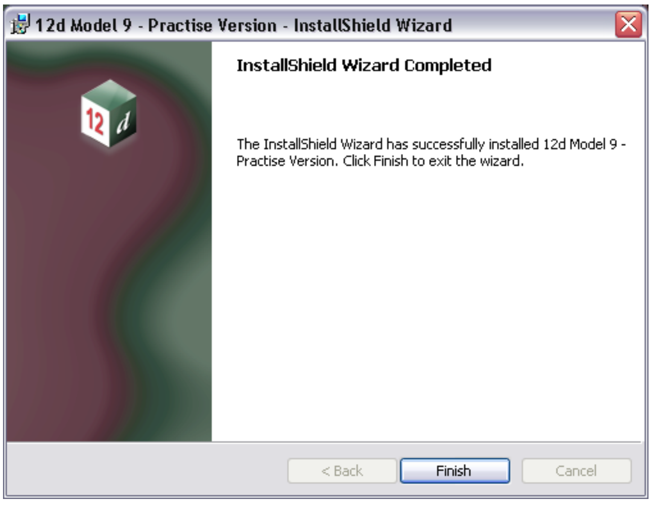


select **Install** to continue the installation

The software will be copied onto the computer and installed.



Setup Complete *End of installation*



Select **Finish** to complete the installation

The ***12d Model Practise*** screen will then appear.

Select **Exit** at the bottom right hand corner of the screen to end the installation.

For ***Practise*** versions, the software needs to be authorised by emailing information about your computer to 12d Solutions.

The form for the required information is automatically created when an unauthorised *12d Model Practise version* is ***started up***. The form can be emailed to *12d Solutions* from the *12d Model Practise* software.

3.2 Folders and Icons Created by Installing 12d Model Practise

Folders Created by the Installation

(a) For *12d Model Practise* on Windows 32-bit operating systems:

the *12d Model Practise* installation loads the ***12d Model Practise software*** into the folder

C:\Program Files\12d Practise\12dmodel\10.00

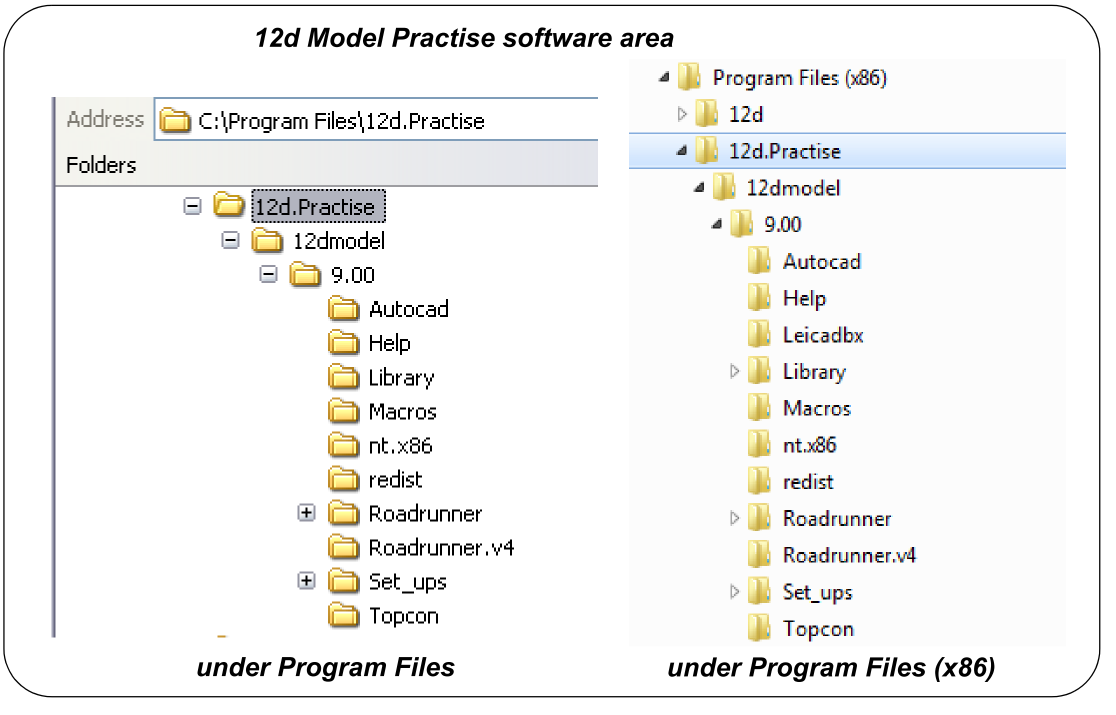
OR

(b) For *12d Model Practise* on Windows 64-bit operating systems:

the *12d Model Practise* installation loads the ***12d Model Practise software*** into the folder

C:\Program Files (x86)\12d Practise\12dmodel\10.00

12d Model Practise users normally only have ***read*** access to the *Program Files* and/or *Program Files (x86) areas.*

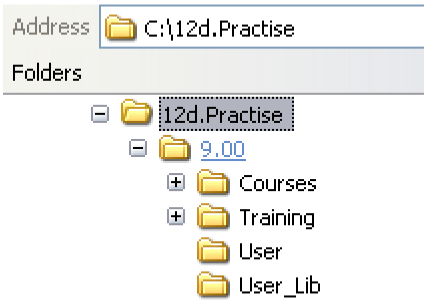


The *12d Model* installation also creates an area

C;\12d Practise \10.00

which during the installation is given read/write access for the user.

The folder 12d Practise\10.00 contains the training data used with the *Getting Started for Design and Getting Started for Surveying* manuals *and the subfolders* ***User*** and ***User_Lib*** are created for future user customisations of *12d Model*



12d Model Practise user areas

Icon ~~s~~Created by the Installation

The installation loads the appropriate components and creates the ***12d Model 10 Practise*** icon

The *12d Model 10 Practise* icon fires up ***12d Model Practise*** and attaches to the folder *12d Practise\10.00*.

3.3 Authorizing the Practise Version of 12d Model 10

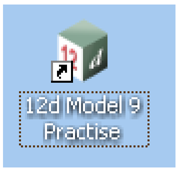
Warning

The *12d Model Practise* version and all documentation is supplied by *12d Solutions Pty Ltd* for the use of 12d Model Customers and genuine prospects only. The use of the *12d Model Practise* version by commercial competitors to 1***2d Model*** is strictly prohibited.

The Practise version is limited to a maximum of 5,000 points and creates projects that cannot be accessed by the Release versions of 12d Model and vice-versa. However the Practise version can be used free of charge by *12d Solutions* customers and registered Practise Users.

The Practise version needs to be Registered with *12d Solutions* once it is installed on a computer and a new Registration is required for each computer the Practise version is run on.

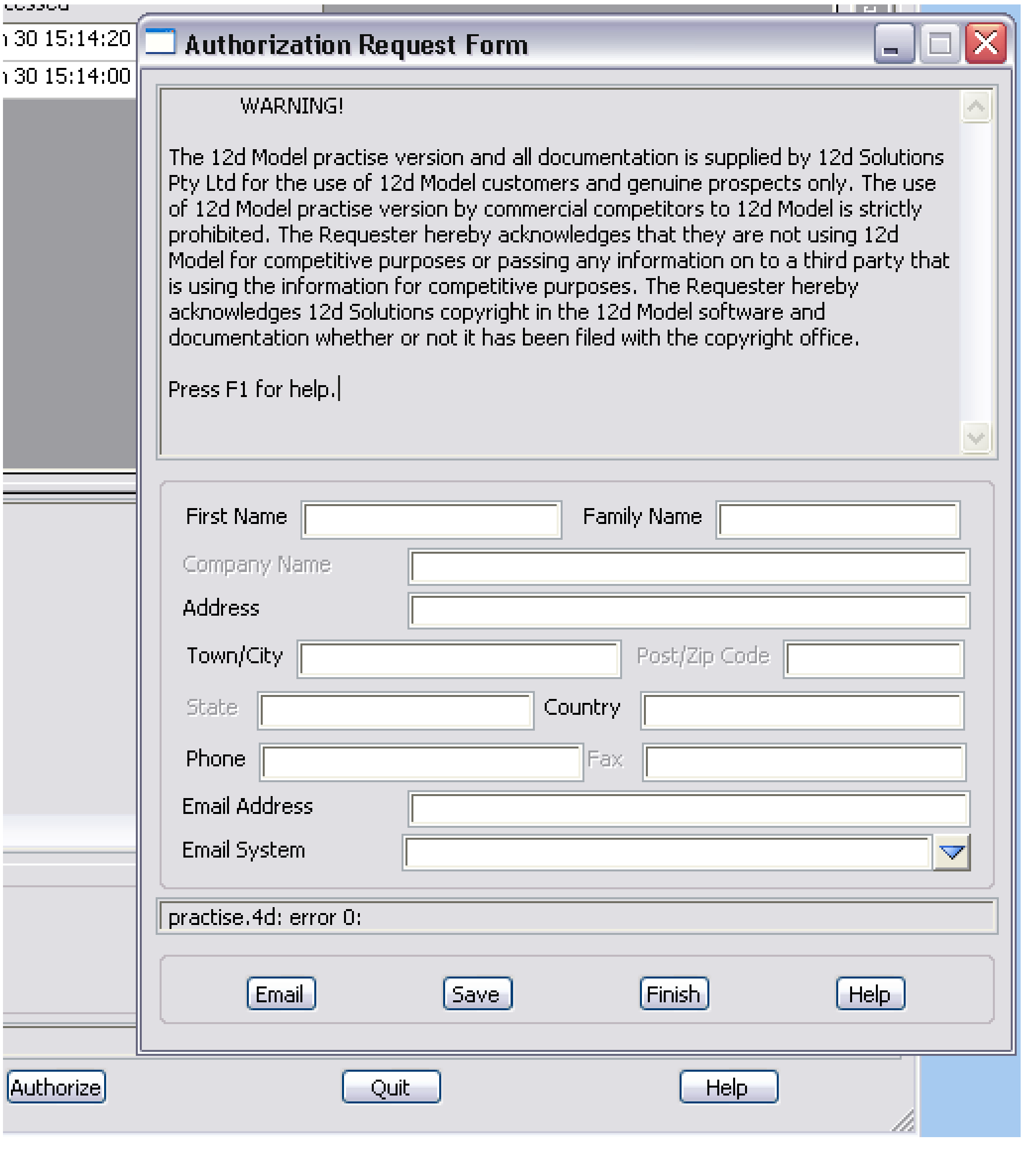
The installation creates the ***12d Model 10 Practise*** icon:



Next fire up ***12d Model Practise*** by double clicking on the *12d Model 10 Practise* icon.

This will bring up the Project Selection panel with an **Authorize** button it.

Click on **Authorize** to display the **Authorize Request Form** panel.



Fill in the details (all those in black type must be filled in) and then either

(a) click on the **Email** button to send the information to 12d Solutions Pty Ltd

or if this fails

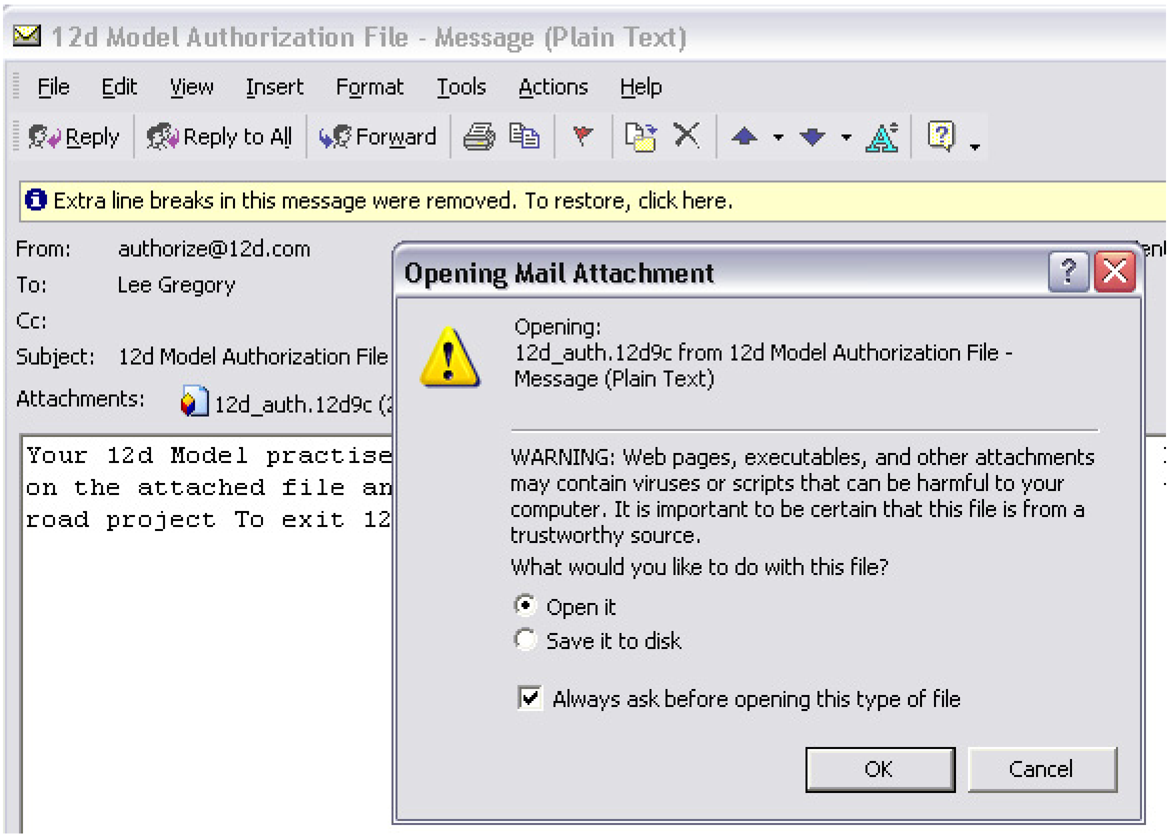
(b) click on the **Save** button. A file called *12d_auth.12d10r* is then written out to the folder ***My Documents***. Please email this file (as an attachment) to *authorize@12d.com*

Then click on the **Finish** to exit *12d Mode*l.

On receiving your request, 12d Solutions will generate an authorization code and send an email back to you with an attachment called ***12d_auth.12d10c***

When you receive the email with the file *12d_auth.12d10c* attached to it, double click on the attached file *12d_auth.12d10c,* click on ***Open it*** and finally click on ***OK***.

.



12d Model Practise will then start up and

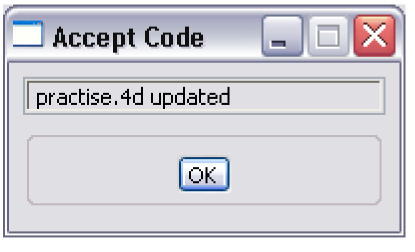
validate the ***12d_auth.12d10c*** file, and if it is valid

save the authorization code away as the file ***practise.4d*** in the folder

c:\12d Practise\10.00

and then bring up the ***Accept Code*** panel with the *practise.4d updated* message.

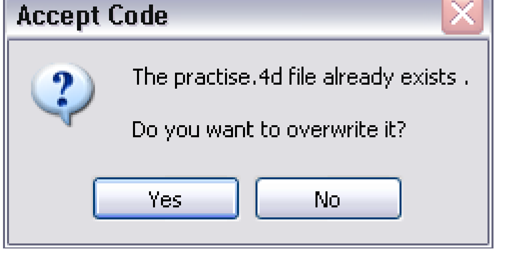
.



Click on ***OK*** and *12d Model Practise* will open and display the project *Road*.

Note: if you already have a pratise.4d file in the folder then the ***Accept Code*** panel will be brought up, stating that the *practise.4d* file already exists, and asking to overwrite it.

.

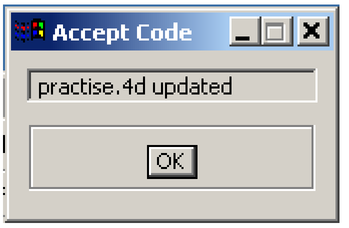


Click on ***Yes*** and the new authorization code will be saved away as the file ***practise.4d*** in the folder

c:\12d Practise\10.00

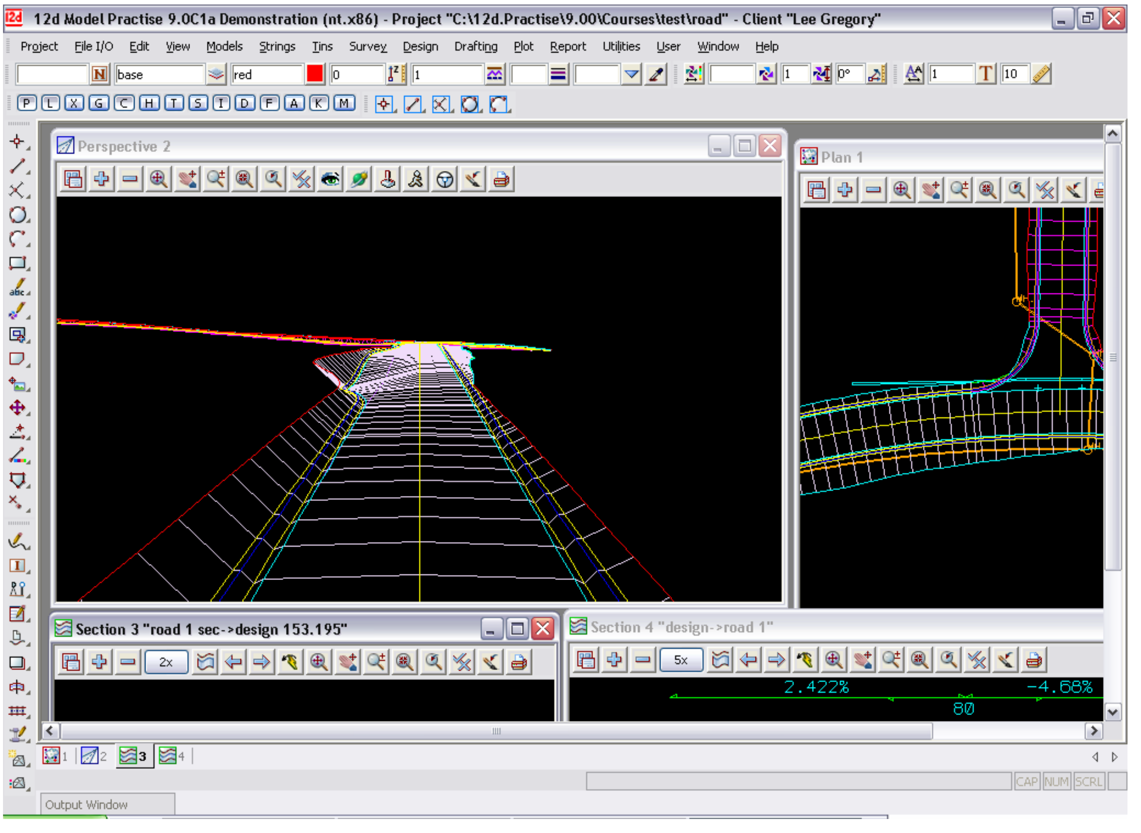
The ***Accept Code*** panel then comes up with the *practise.4d updated* message.

.



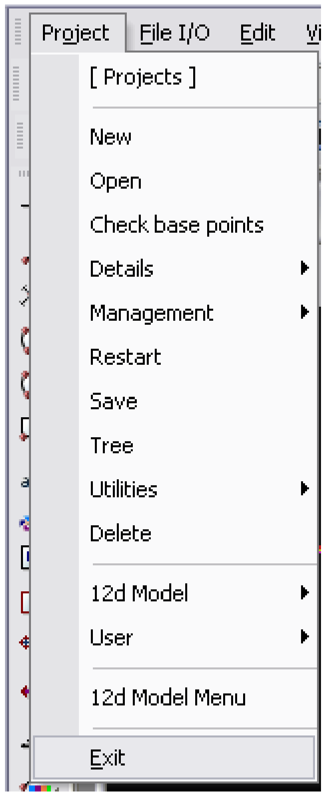
Click on ***OK*** and *12d Model Practise* will open and display the project *Road*.

.



To exit *12d Model Practise*, click on **Project =>Exit**

.



If *12d Model Practise* does not start up correctly, please go to the next section 3.4 Possible Problems When Authorizing the Practise Version

3.4 Possible Problems When Authorizing the Practise Version

1. The date on your computer is not the correct date.
2. The computer your are authorizing is not the same one that the *Authorisation Request Form* was generated for.

   The ***12d_auth.12d10c*** and the generated ***practise.4d*** file are only valid for the computer that the *Authorisation Request Form* was generated on. If you want the Practise version to run on another computer, you need to generate a new *Authorisation Request Form* on that computer and send it to 12d Solutions.
3. If you change your network card after generating the *Authorisation Request Form* then the authorization will stop working. A new *Authorisation Request Form* needs to sent to 12d Solutions.
4. If the file ***12d_auth.12d10c*** does not appear as an attachment then your email system can't handle MIME attachment. Please contact 12d Solutions at *authorize@12d.com*

4 Tools and Concepts

The chapter contains information about the definitions and behaviour of general items used in ***12d Model*** such use of the mouse, keyboard, how information is displayed on the screen, picking items, *etc.*

See

For documentation on the first item, continue to the next section 4.1 The Mouse.

4.1 The Mouse

The mouse is used extensively in ***12d Model***.

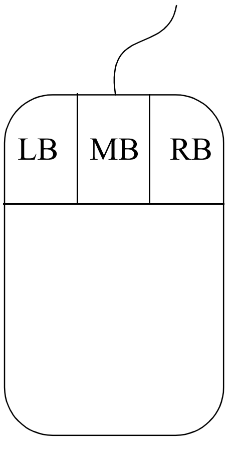
12d Model can be operated with either a two or a three button mouse but a three button mouse is preferred.

In this manual the buttons will be denoted by

LB = the left button

MB = the middle button

RB = the right-button



12d Model monitors the mouse being pushed down and when it is subsequently released as separate events. Unless otherwise specified in the manual, **clicking** a button will mean **pressing the button down and releasing it again**. The **position of the mouse** is normally taken as being when the **button is released**.

In screen messages, the effect of pressing each button on the mouse is shown by enclosing the effect for each button in square brackets ([]) in left-to-right button order. That is

[left button effect] [middle button effect] [right button effect]

Empty brackets, [], indicate that pressing the button has no effect at that time.

NOTE: If the Middle button is also a wheel, then the wheel can be used in some ***12d Model*** operations such as zoom.

Please continue to the next section 4.2 The Keyboard or return to 4 Tools and Concepts.

4.2 The Keyboard

For clarity, the characters and special keys on the keyboard will be enclosed in the angle brackets **< >**. For example, the delete key is <del>.

When two or more keys are to pressed down together, they will be shown in angle brackets separated by a plus sign (**+**).For example, **<ctrl> + <d>** means that the control key and d are pressed down together.

In ***12d Model***, the *escape* key (normally labelled **Esc** on the keyboard and denoted by <esc> in this manual) is used to **stop drawing** in a view or to **break out** of computer intensive options (*escape* or *abort* the option) but still remain in ***12d Model***. Options which can be terminated by <esc> are noted in the manual.

Please continue to the next section 4.3 Screen Layout or return to 4 Tools and Concepts.

4.3 Screen Layout

Inside the ***12d Model*** screen are six distinct areas (main menu, top Toolbar area, side toolbar area, view area, output window and status bar) which create and control nine associated objects (panels, toolbars, control bars, floating menus, Plan views, Section views, Perspective views and Perspective OpenGL views displayed on the screen.

The main areas and their purposes are:

(a) Main Menu

The **Main Menu** is at the top of the ***12d Model*** screen and is a standard Microsoft type menu. Options are selected in the standard Microsoft way and bring up ***12d Model*** **panels** and **floating menus**. See 4.3.1 Main Menu.

(b) Views Area

The **Views Area** displays the ***12d Model*** drawing views for seeing and examining the data. There is no limit to the number of **Plan**, **Section** or **perspective, Perspective OpenGL** and views that can be created, overlapped or iconised (the **Perspective Hide** view has been superseded by the Perspective OpenGL view). See 4.3.7 Views.

(c) Output Window

The **Output Window** displays ***12d Model*** system and error messages, and intelligent log lines. The Output Window can be turned on/off from the Main menu. See 4.3.9 Output Window.

(d) Background Tasks Window

The **Background Tasks** window displays messages from tasks started by ***12d Model*** as background tasks. The Background Tasks window can be turned on/off from the Main menu. See 4.3.10 Background Tasks Window.

(e) Status Bar

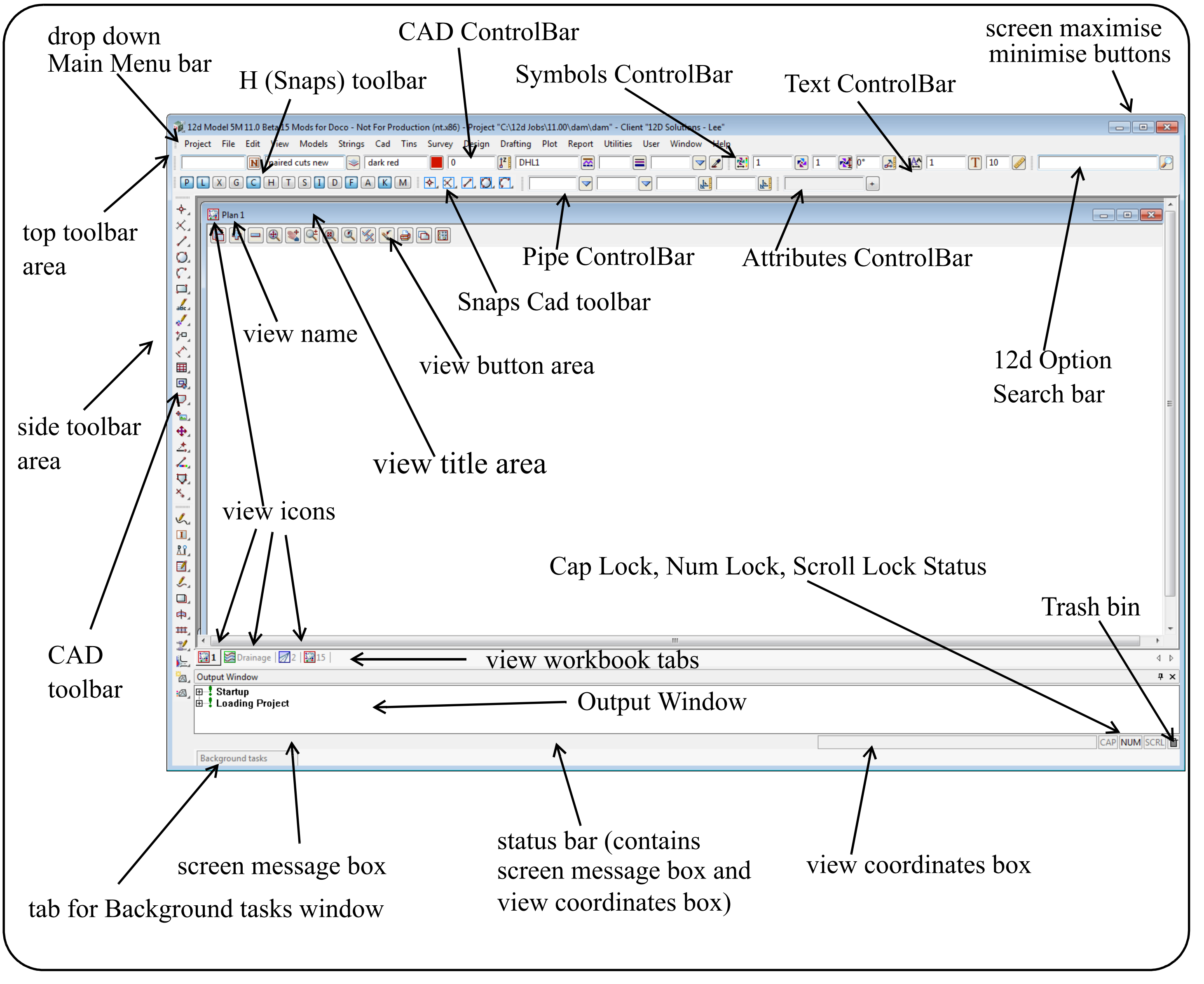
The **Status Bar** displays message prompts and the x, y and z coordinates of the cursor when it is in a drawing view. The Status Bar can be turned on/off from the Main menu. See 4.3.8 Status Bar.

(f) Top Toolbar area

The **Top Toolbar area** is below the Main Menu and contains the CAD control bar, Text control bar and the snaps toolbar.

(g) Side Toolbar area

The **Side Toolbar area** is on the far left side and the CAD Toolbar and other Toolbars defined in *Toolbars.4d*

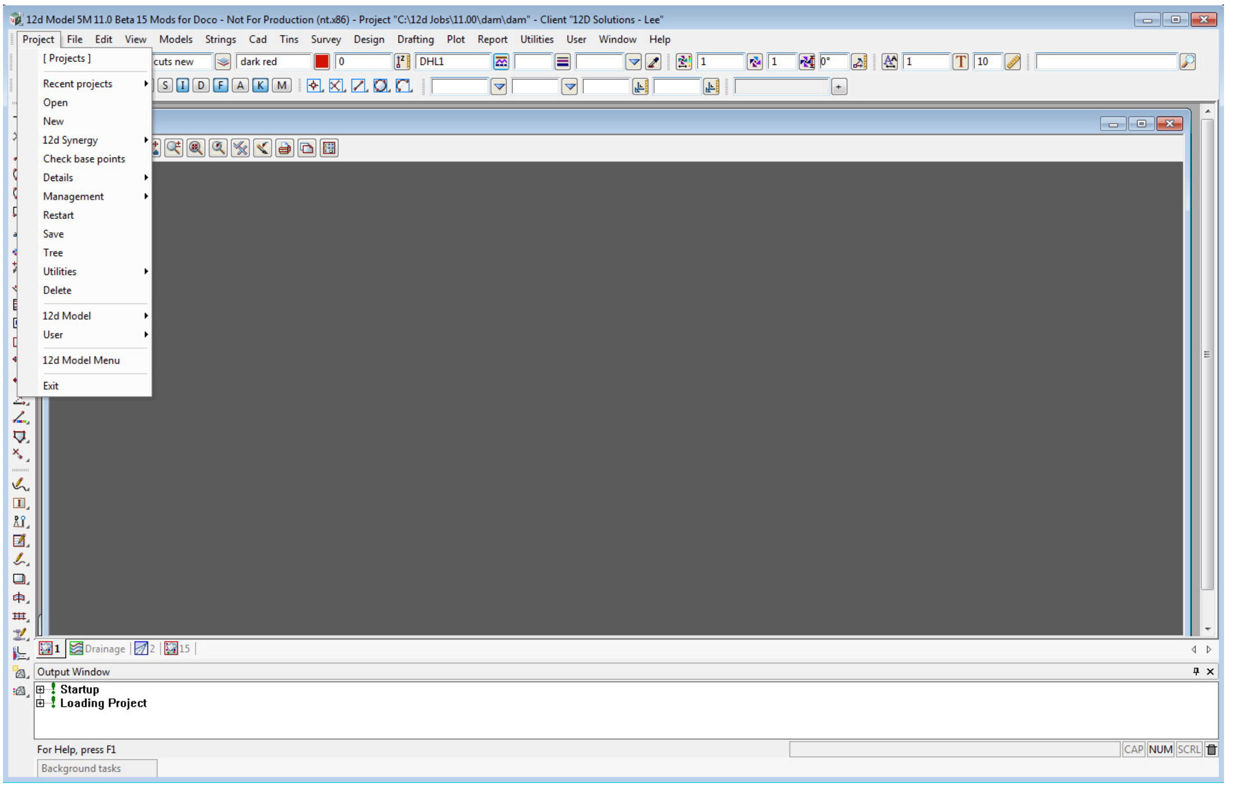


Please continue to the next section 4.3.1 Main Menu.

4.3.1 Main Menu

At the top of the ***12d Model*** window is the **Main** menu, a standard Microsoft type menu.

.



Options can be selected in the standard Microsoft way and bring up ***12d Model*** panels (dialogues) or create floating menus.

The **Main** menu can be moved around, or docked on any of the four sides of the window.

If the Window is not wide enough, the **Main Menu** will automatically wrap around onto two or more lines.

Options can be selected from the **Main Menu** or from the walk rights on option on the **Main Menu**.

Or most menus on the **Main Menu** can be torn off to create floating menus that can be placed anywhere on the screen,

Continue to the next section 4.3.2 Creating Floating Menus from the Main Menu.

4.3.2 Creating Floating Menus from the Main Menu

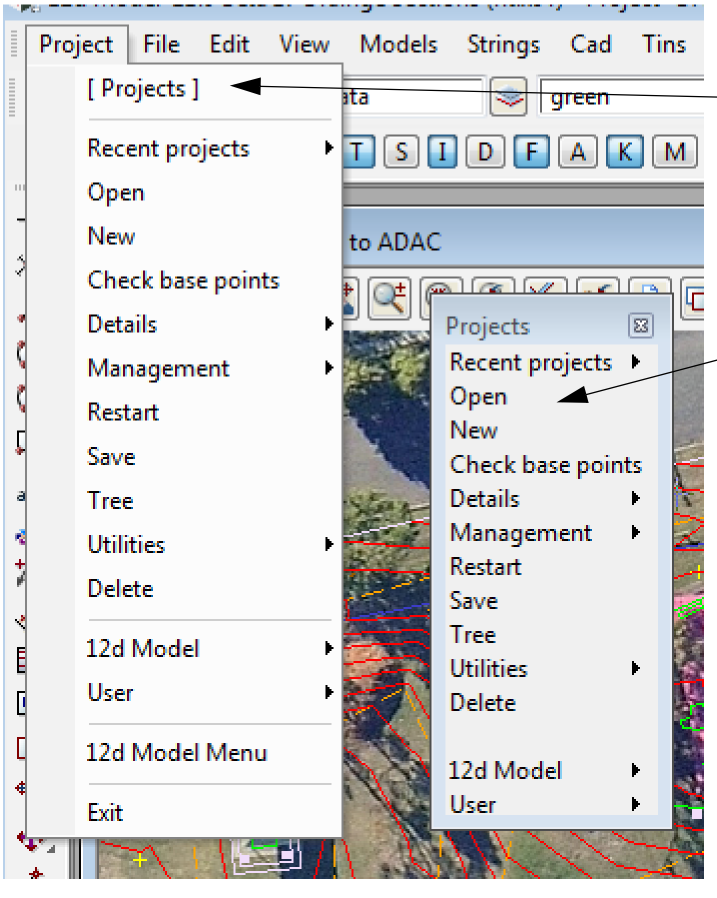
As well as the standard Microsoft type menu, ***12d Model*** has the concept of **floating (tear away) menus**.

The presence of a **floating menu** on the **Main Menu** is indicated by a menu item enclosed in square brackets **[ ]**. Selecting a menu item in square brackets will create a **floating menu** of that name.

For example, selecting **[Project]** from the top menu **Project** creates the floating menu called **Projects.**

A floating menu can be moved around the screen, even outside the ***12d Model*** window, and will stay up until the **[X]** button is selected on the top right hand side of the floating menu.

A floating menu created from the **Main Menu** usually contains the same items and walk-right menus as the **Main Menu** but has the distinct advantage that it **doesn't disappear** when the cursor is moved to select other menus from the **Main Menu.** It stays on the screen as a menu in its own right.



[Projects] to indicate that a floating menu (tear away menu) can be created by clicking on the **[Projects]**

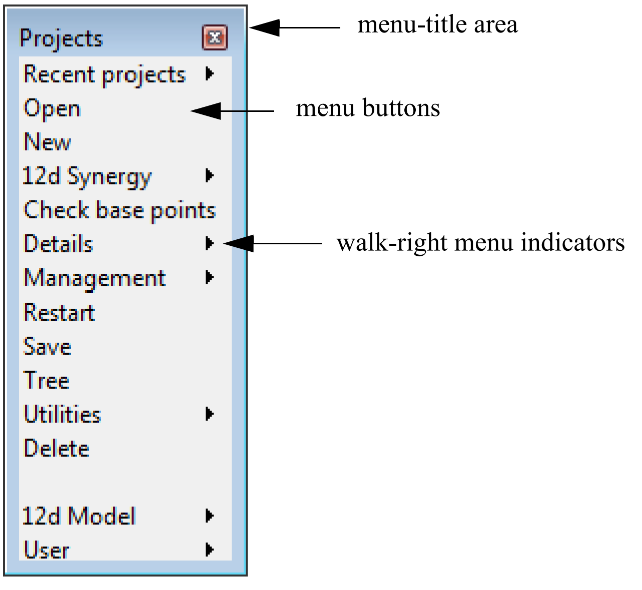
the **Projects** floating or tear away menu

Continue to the next section 4.3.2 Creating Floating Menus from the Main Menu.

4.3.3 Floating Menus

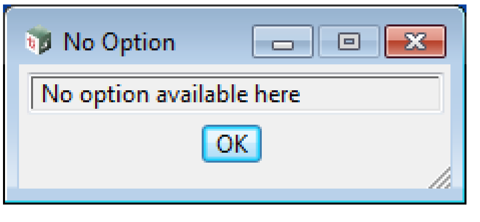
Floating menus or **tear away** menus consist of a **menu title area** and a series of options, called menu buttons, lined up under the menu title.

For example, the **Projects** menu looks like



An option on the menu is invoked (or selected) by clicking LB whilst over the option.

If no option exists, a **No Option** panel is placed on the screen. The panel will remain until the **OK** or **[X]** button is selected.



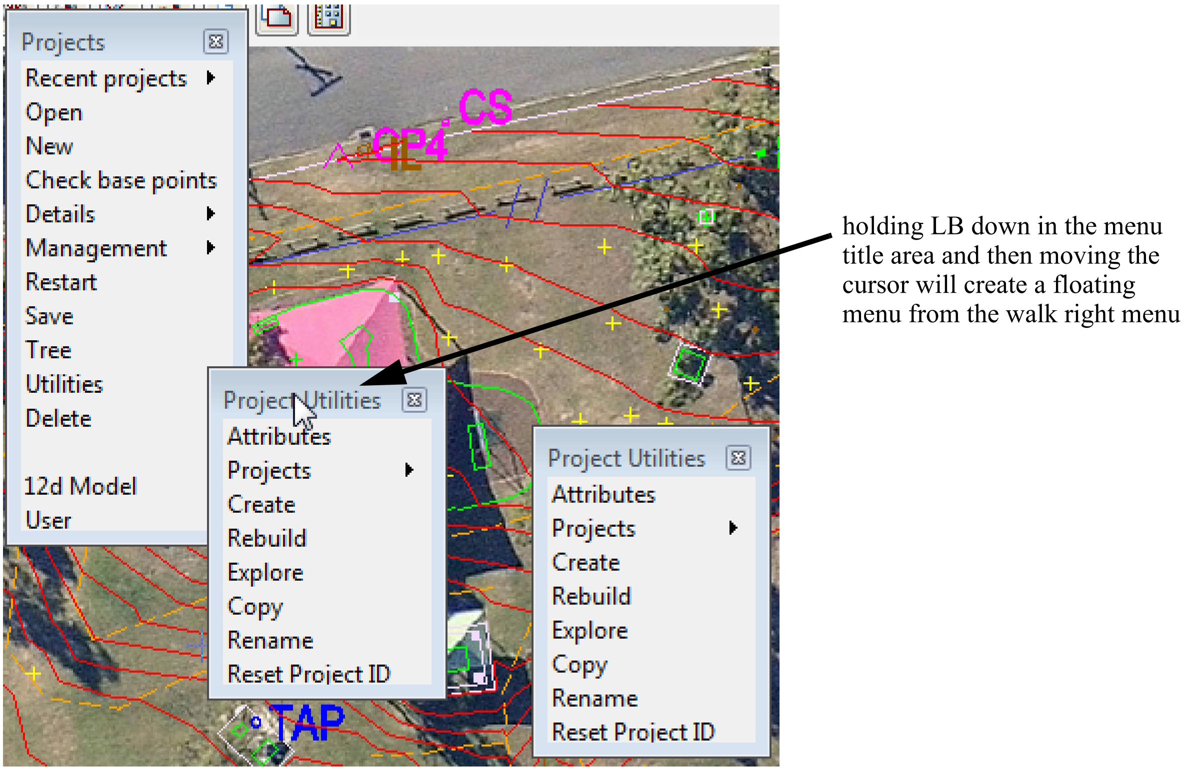
The icon ▸ on a menu-option indicates that there is a **walk-right** menu attached to that button.

The walk-right menu only appears as the cursor **moves over** the ▸ .

The walk-right menu can consist of further menu options and walk-rights.

If a walk-right menu is showing and the user wanders back into the menu where the walk-right originated, the walk-right menu will collapse back to the ▸ .

When a walk right menu is showing on a **floating menu**, holding LB down in the menu title area and then moving the cursor will tear the sub menu from the main menu and create a **new floating menu** of the **walk right menu**.



For the description on:

Moving a menu go to	4.3.3.1 Moving a Menu
Dumping a menu	4.3.3.2 Dumping a Menu or Panel
Deleting a menu	4.3.3.3 Deleting a Menu

4.3.3.1 Moving a Menu

A menu can be moved (dragged) by **holding down LB** anywhere in the menu title area (except on the **[X]** button) and then moving the cursor with **LB** still depressed. The menu moves with the cursor to indicate where the new menu position will be. The menu is finally positioned when **LB** is released.

When an expanded walk-right menu is moved, a **copy** of the walk-right menu will be moved and placed on the screen as a new floating menu. The original menu still contains the walk-right menu.

Continue to the next section 4.3.3.2 Dumping a Menu or Panel or go back to 4.3.2 Creating Floating Menus from the Main Menu.

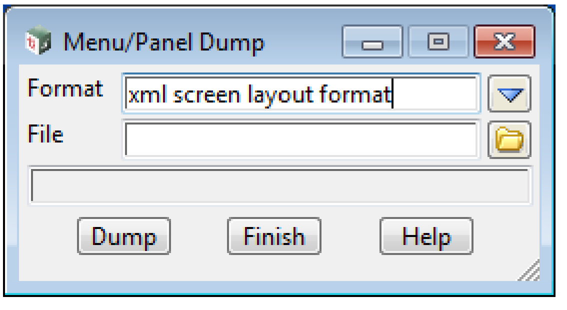
4.3.3.2 Dumping a Menu or Panel

The menu or panel can be written out to disk in a variety of images formats, as a screen layout file (SLX), and for a panel, as a defaults file (DDX). This is called dumping the menu or panel.

The **Dump** option is invoked by clicking the middle mouse button (MB) anywhere in the menu/panel title area. The **Menu/Panel Dump** panel is then displayed.

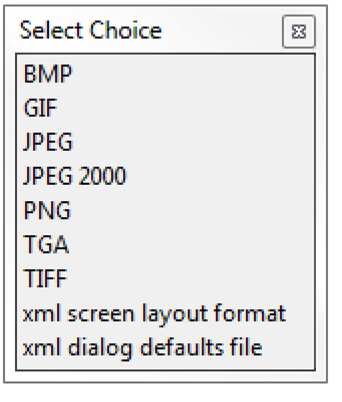
Note: If a screen layout file is created and added to the *layout.4d* file (by selecting *layout.4d* as the file name and selecting **Append**), then the menu/panel will appear whenever a project is opened. See layout.4d

.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Format	choice box	screen layout file	



format for writing the menu/panel to disk.

File file box

name of the disk file to write the image or screen layout information to.

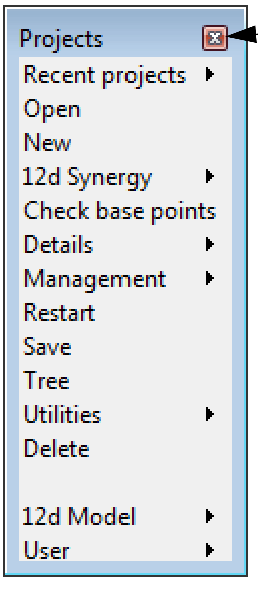
Dump button

dump the menu information to disk.

Continue to the next section 4.3.3.3 Deleting a Menu or go back to 4.3.2 Creating Floating Menus from the Main Menu.

4.3.3.3 Deleting a Menu

A floating menu is deleted by selecting the **[X]** button in the menu-title area.



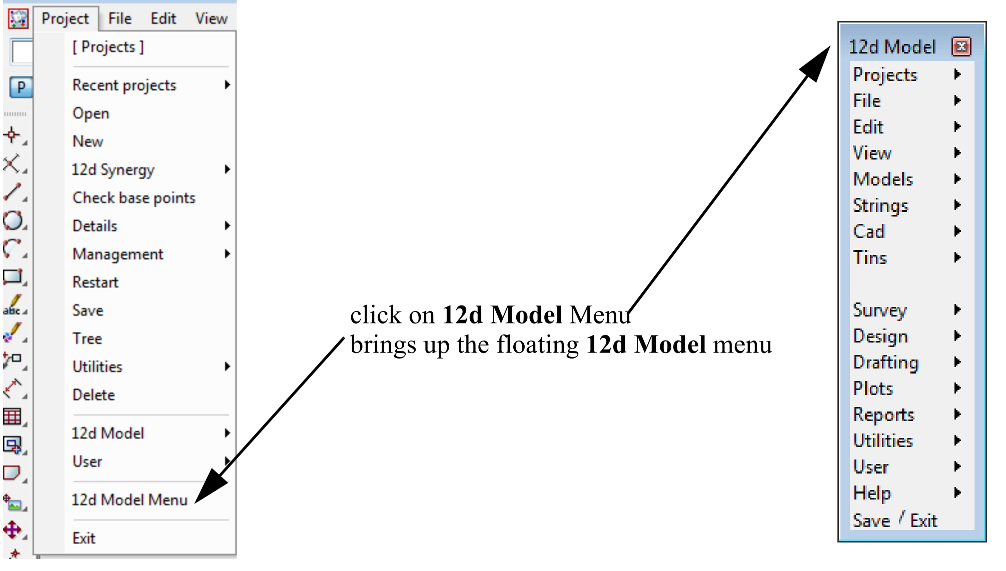
click on [X] to delete the menu and remove it from the screen

Please continue to the next section 4.3.4 12d Model Menu.

4.3.4 12d Model Menu

A floating ***12d Model*** menu is available by selected ***12d Model* Menu** on the **Projects** menu.

Selecting **12d Model Menu** again will *remove* the **12d Model** menu.



All of the options going across the **Main Menu** are available by going down the **12d Model** menu.

Please continue to the next section 4.3.5 Toolbars and Controlbars.

4.3.5 Toolbars and Controlbars

See

4.3.5.1 Toolbars
4.3.5.2 ControlBars
4.3.5.3 Snaps (H) Toolbar

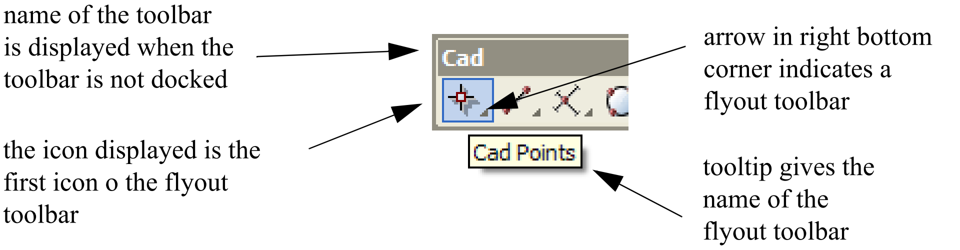
4.3.5.1 Toolbars

Options can be grouped together on *toolbars*. T*oolbars* are made up entirely of *icons* and each icon can have a ***12d Model*** option attached to it.

Toolbars have unique names which are displayed on the top of the toolbar when it is not docked.

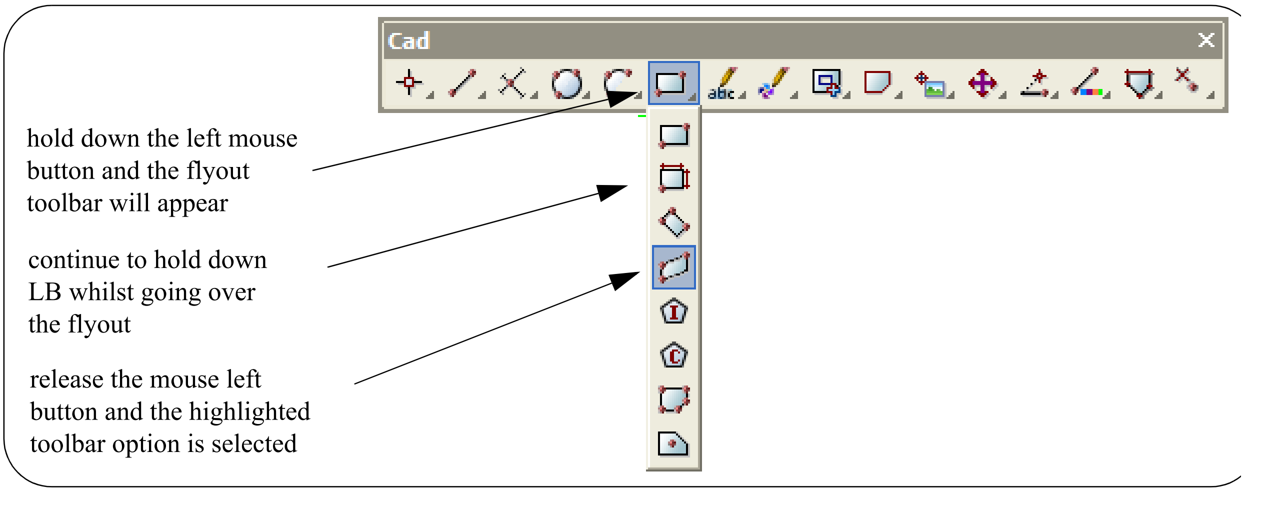
Toolbars can also have one level of **Flyouts** of other toolbars and when a flyout exists, there is a small arrow on the bottom right of the icon to indicate a flyout exist. The *tool tip* for the icon gives the *name* of the flyout toolbar. The actual icon itself is the icon of the f*irst item* on the *flyout* toolbar.

For example, the *CAD toolbar* is entirely made up of flyout toolbars.



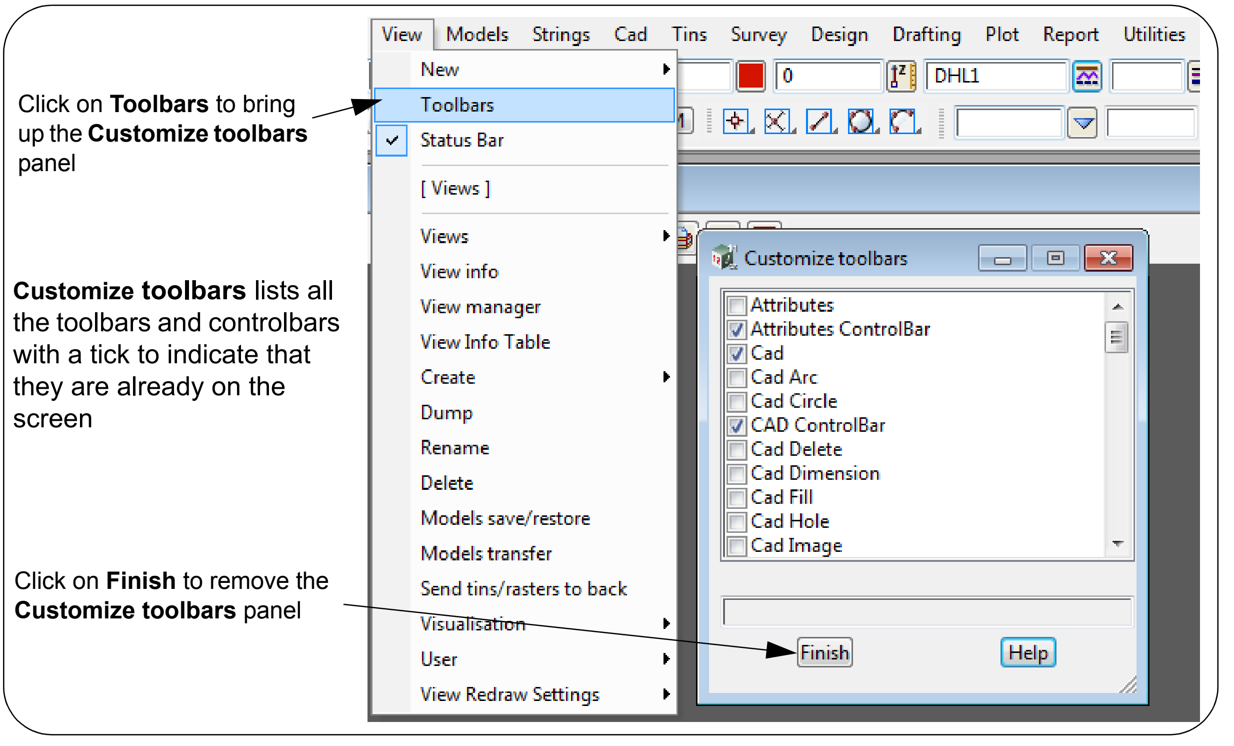
To display and select from a flyout toolbar, hold down LB whilst over the icon on a toolbar with a flyout menu and the flyout menu will appear. Whilst still holding down LB, move along the flyout toolbar to the appropriate option and then release LB. The option attached to the highlighted icon will then be selected.

Note - if LB is released whilst a icon with a flyout toolbar is highlighted, the first option on the flyout Toolbar is selected.



All the toolbars in ***12d Model*** are defined in a file with the default name *toolbars.4d or user_toolbars.4d.* For the definition of toolbars, see the section 41.3 User Defined Toolbarsin the Appendix 41 Functions Keys, Menus, Toolbars.

The list of all defined toolbars is given by selecting **View =>Toolbars** from the main menu which brings up the **Customize toolbars** panel. A toolbar is shown with a tick if it is already on the screen.



Ticking **on** a toolbar in the list brings up the toolbar in the top left hand corner of the screen. The toolbar can then be moved or docked.

Ticking **off** a toolbar in the list, or clicking on **X** in the top right hand corner when the toolbar is not docked, removes the toolbar from the screen.

Important Note - the toolbars are not active whilst the **Customize toolbars** panel is on the screen. Close the **Customize toolbars** panel as soon as you finished with it.

What toolbars are on the screen and their position is recorded with the project so that when a project is exited and later started again, the final position of the toolbars is restored.

Also at any time, a workspace file can be read in and define a new setup and position of toolbars. For more information on using workspace files, please go to the section 7.6.9 Project Workspace.

For links to the options on each *Toolbar, go to* 4.28 Options on Toolbars.

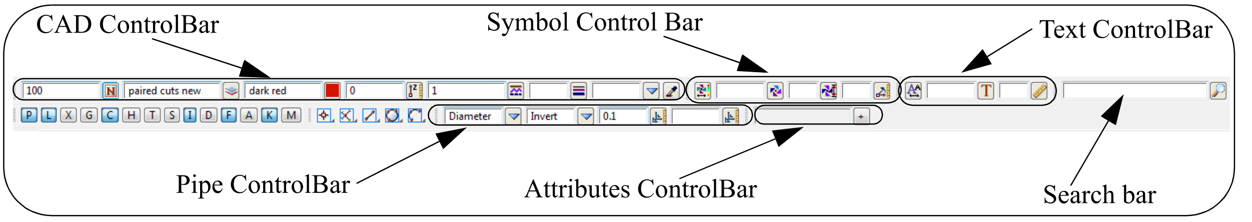
Continue to the next section 4.3.5.2 ControlBars or return to 4.3.5 Toolbars and Controlbars.

4.3.5.2 ControlBars

ControlBars can have icons on them but they also have controls such as a model box on them.

ControlBars are not user defined and there are six of them in ***12d Model*** - the **CAD ControlBar**, **Symbol ControlBar**, **Text ControlBar**, **Pipe ControlBar**, **Attributes ControlBar** and **Search bar.**

This default position of the *controlbars* is on the top left hand side of the screen under the main menu.



Although not toolbars, the *Controlbars* do appear in the *Toolbars* list in the **Customize toolbars** panel and can be turned on/off and docked just like toolbars.

Important Note - the toolbars are not active whilst the **Customize** panel is on the screen. Close the **Customize toolbars** panel as soon as you finished with it.

The *ControlBars* CAD, Symbol, Text, Pipe and Attributes are used by *CAD toolbars* and the *Strings Create* options. *Strings* created by the *CAD toolbars* and *Strings Create* options use values in the *ControlBars* to define string properties such as name, model, colour, pipe justification and size.

Like toolbars, the type and position on the screen of ControlBars is recorded with the Project so that when a Project is exited and then started later, the final position of the *ControlBars* is restored.

Also at any time, a workspace file can be read in and define a new setup and position of the *ControlBars*. For more information on using workspace files, please go to the section 7.6.9 Project Workspace.

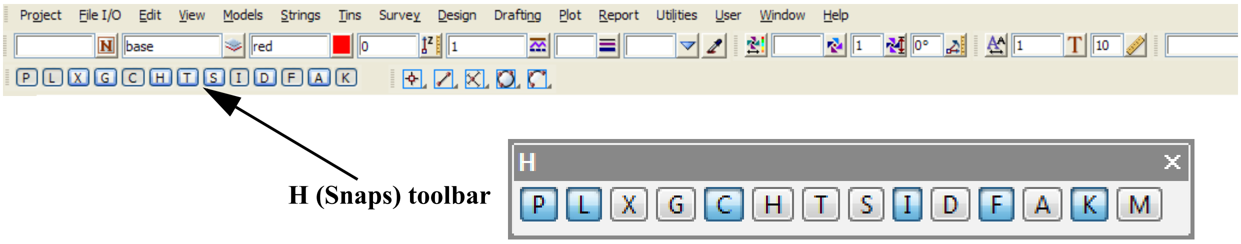
The Search bar is also a ControlBar and allows quick access to any option in ***12d Model***.

The *ControlBars* **CAD**, **Symbol**, **Text**, **Pipe**, **Attributes** and **Search bar** will be described in detail in the section 15.1 Controlbars.

Continue to the next section 4.3.5.3 Snaps (H) Toolbar or return to 4.3.5 Toolbars and Controlbars.

4.3.5.3 Snaps (H) Toolbar

The ***H Toolbar*** (the ***Snaps*** toolbar) which has the default position at the top left hand side of the ***12d Model*** screen, under the CAD ControlBar.



Snaps are documented in the section 4.13 Snaps or return to 4.3.5 Toolbars and Controlbars.

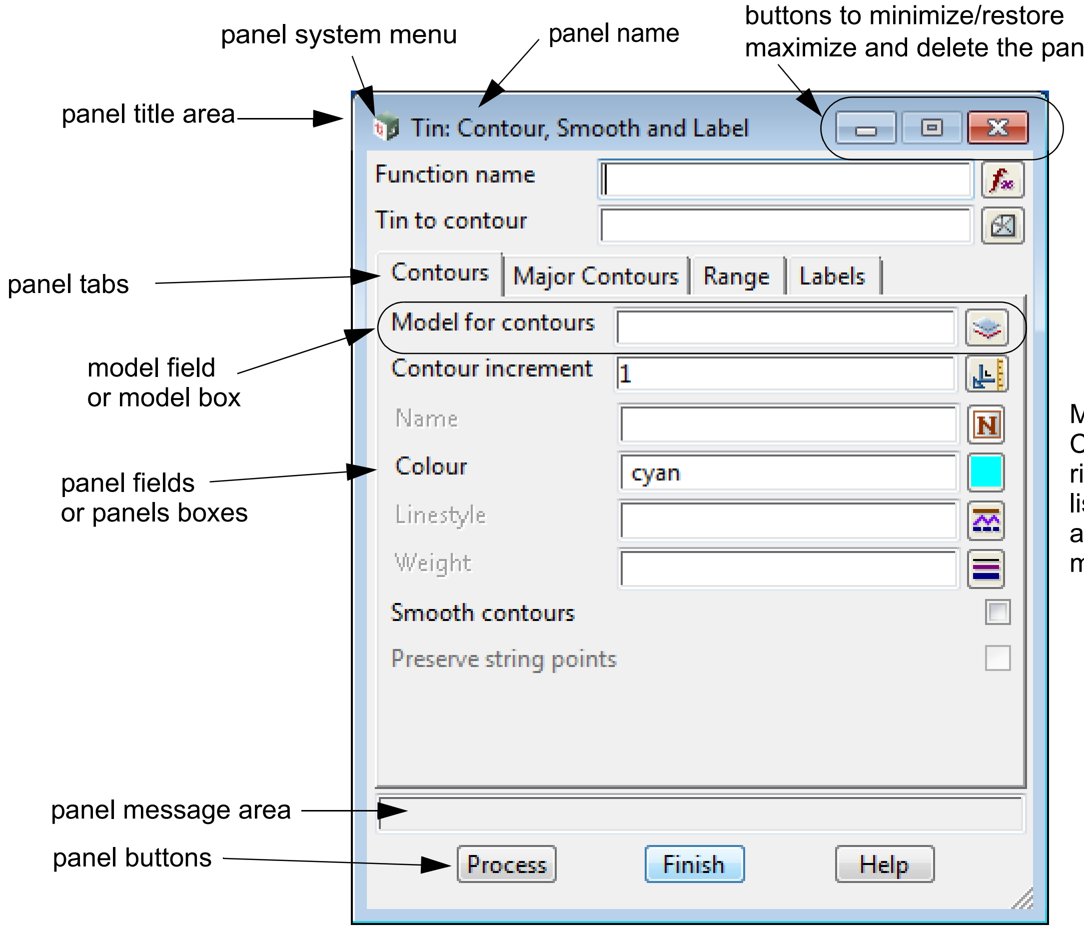
4.3.6 Panels

After a menu option has a been selected, extra information is often required before the operation represented by the menu option can begin.

For example, before the contour option can proceed, the required contour interval must be supplied.

In ***12d Model***, an object called a **panel** is used to collect and validate any extra information required to run the option.

For example, the **Model Information** panel is



A Panel consists of various types of areas.

Panel title area

Contains the title (name) of the panel, the buttons to minimize and restore the panel, and the **[X]** button. In the example above, the panel title is **Tin: Contour, Smooth and Label**.

Panel fields (also know as Panel Boxes)

Depending on the panel, these area for entering and/or displaying information for the panel.

The user can type answers into some fields or the program may display special information in the fields (output only fields).

Often the areas are for collecting/displaying special ***12d Model*** information. There is often an icon at the far right of the panel field/panel box which may contain a pop-up list of values to choose from.

In the example, **Model for contours** is an input field and is a Model box.

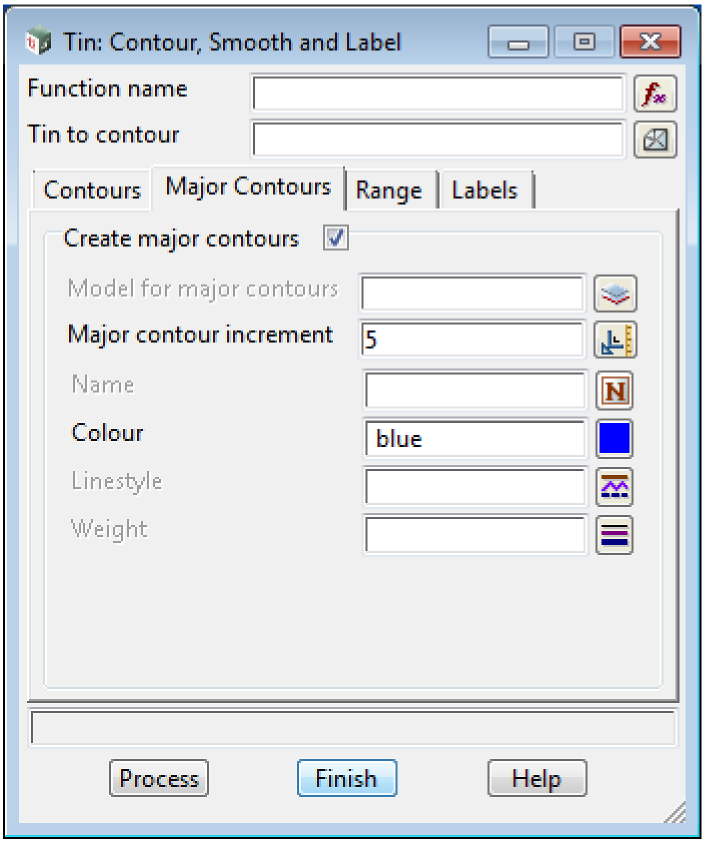
If the panel field is *optional*, then the description text is greyed out - entering data into the field will cause the text to be redrawn in black.

Panel Tabs

Clicking on a **Panel Tab** displays a whole new page of panel fields.

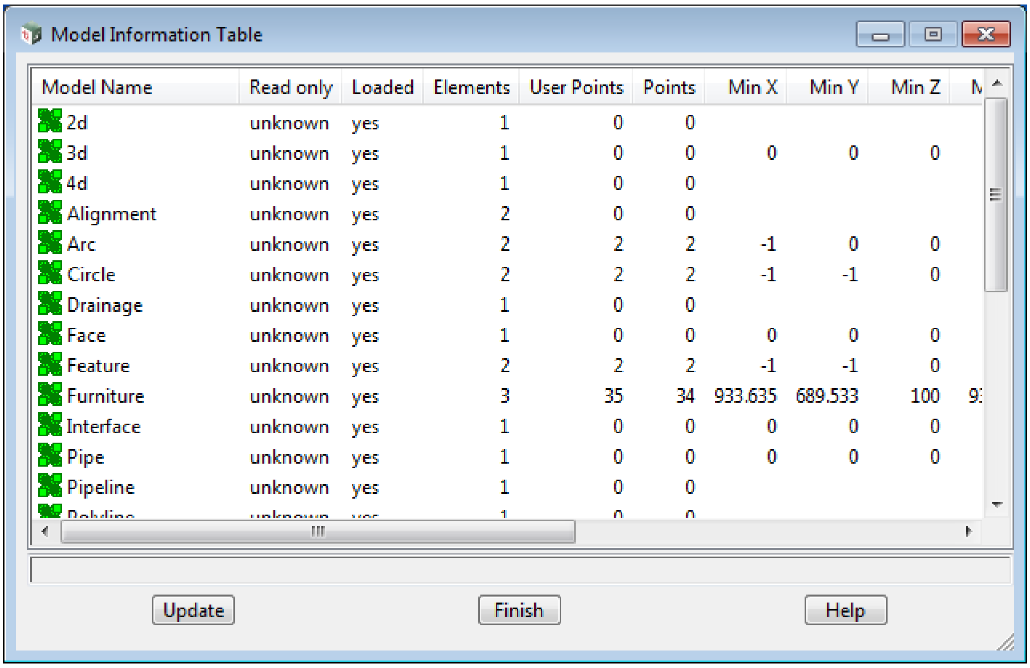
Panels tabs are often needed because there are too many panel fields to fit on the panel at the same time, or to group some fields together.

For example, clicking on Major Contours in the **Tin: Contour, Smooth and Label** panel will display panel fields to collect information to define the **Major Contours**.



Panel Grids

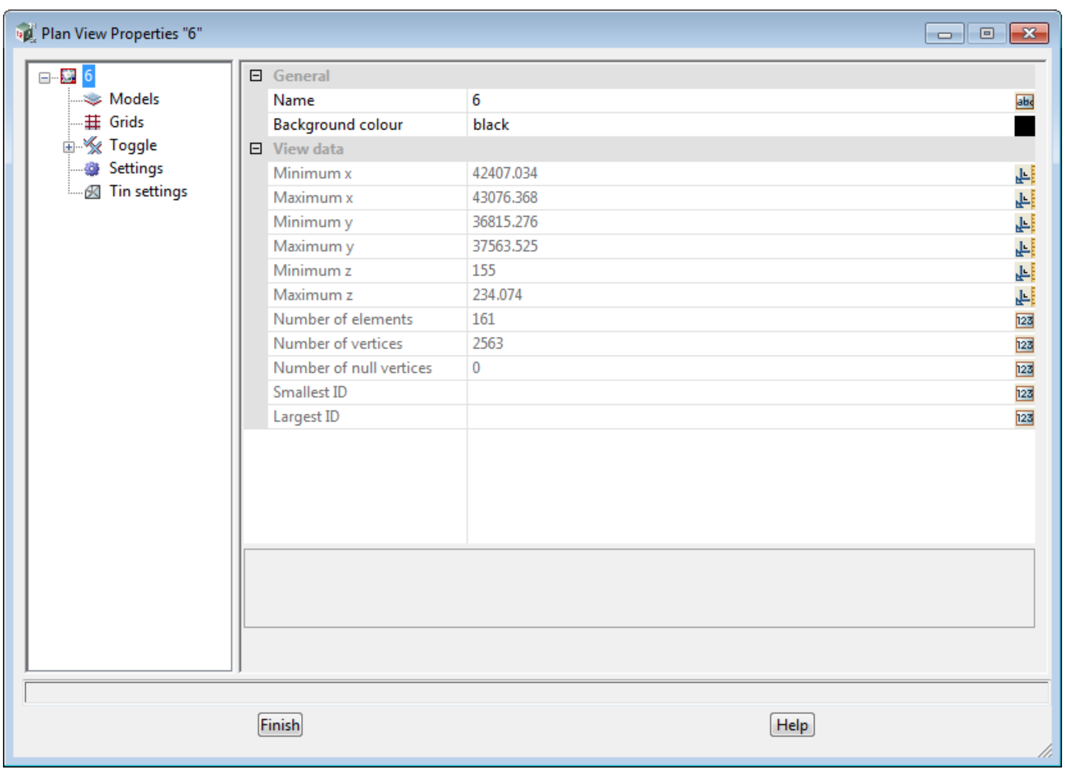
Many panels have grids to collect information that is best displayed and entered in columns.



For information on grids in panels, see 4.19.6 Grids in Panels.

Panel Trees

Panels have trees to better order the information that is to be displayed and collected.



Message area

Each panel has its own area where ***12d Model*** displays messages for the option.

Messages are used for a variety of purposes including

- indicating the **next step** in a complex option
- reporting **errors**
- giving **progress** in time consuming operations
- informing that an option has **completed** successfully.

Most messages are sent to the panel message area but a few are displayed in the Status Bar.

Panel buttons

Buttons are used on the panel to select things or control the processing of the panel. Almost every panel has minimize and restore button, and a **Finish** and **[X]** buttons which when selected will end the option and remove the panel from the screen. All panels also has a **Help** button which brings up on-line help about the panel.

The minimize, restore, **Finish**, **[X]** and **Help** buttons will not be documented for each panel.

Continue to the next section 4.3.6.1 Data Entry in a Panel or return to 4.3.6 Panels.

4.3.6.1 Data Entry in a Panel

To help save the user time, many panel fields have default answers.

However, any panel field value can be replaced by typing in new information (**typed input**) or when available, by selecting an answer from the **panel field pop-up** or using special name completion characters.

All panel fields are validated before the option runs and any error messages displayed in the ***panel message area.***

Typed input

To type information into a panel field, move the cursor to the position for the information, left click to get focus in the panel field and then start typing. The information will go into the field starting at the input-position indicator (a upright bar in the panel field).

The input-position indicator can be moved by

- clicking LB when the cursor is at the new position for input
- using the keys ->, <-, home and end

The **backspace** key, **<backspace>**, will delete one character before the input-position indicator and the **delete** key, **<del>**, will delete one character after the input-position indicator.

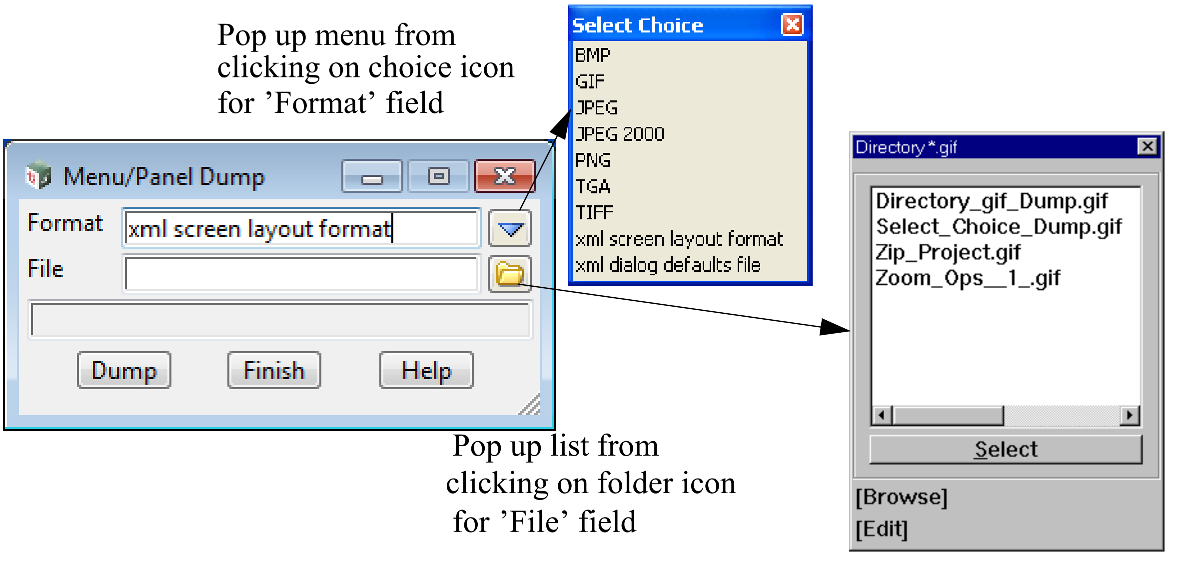
The user can move to any field in the panel by using the mouse. The **<tab>** key will move the cursor to the next panel field or button and **<shift>+< tab>** will move the cursor to the previous panel field or button.

Note - the entire field is used as the answer, not just up to the input-position indicator. The input-position indicator indicates where the characters will go when typing, not the end of the data.

Pop-Up Lists and Menus

The **panel field pop- up list** or **menu**, or panel field pop-up, is raised by clicking LB on the **[+]** or other choice icons at the right of the panel field.

A panel field pop-up consists of a list of choices which may be displayed as either a *menu* or a *list*.



For a pop-up list, an answer is chosen from the list by **double** clicking LB over the required answer.This answer is then displayed in the panel field and the pop-up list disappears.

For a pop-up menu, an answer is chosen from the pop-up menu by clicking LB over the required answer.This answer is then displayed in the panel field and the pop-up menu disappears.

The pop-up list or menu can also be removed without a selection by clicking LB on the **[X]** on the pop-up list or menu, by clicking LB again on the icon or **[+]** for the panel field, or by simply typing into any visible part of the panel field that the pop-up is for (some of the field may be obscured by the pop-up itself).

The pop-up lists and menus and the choice icons are described in more detail in the section on *Special Panel Fields*.

Name completion

To help speed data entry a process called **name completion** is available in most panel fields. **<ctrl> + <d>** is the special key combination used for name completion and how it works will now be described.

If one or more characters have been typed into a panel field and the **<ctrl> + <d>** combination is pressed, then ***12d Model*** checks to see how many answers in the available pop-up start with the same typed characters. If a unique match exists, then it will be placed into the panel field. If more that one match exists, a pop-up menu with all the matches will be presented for the user to select from.

Name mapping

When typing a string name into a **Name Box** panel field, a **name mapping file** can be used to fill out information such as colour, model *etc.* for given string names.

The name mapping works in two ways. After typing part or all of a string name,

(a) if <enter> is entered, the name map file is searched for a match in the first column of the Basic node (the **Key** column).

If a match is found, the name, colour, model, style *etc.* from the other columns in the **Basic** node are used to fill out the panel fields.

The key can contain wild cards and/or characters.

Other nodes of the name mapping file may also fill out other information such as pipe type and pipe diameter (**Pipe** node).

(b) if **<ctrl> + <d>** is entered, the **second** column of the Basic node of name mapping file (the **Name** column) is searched for a list of completions which is written to the field if it is unique or displayed in a pop-up if there is more than one match. If a * is found in the **Name** column, the first column is used for matching. When an entry is selected from the completion list, the name, colour, model, style *etc.* from the columns in the **Basic** node of the name mapping file are used to fill out the panel fields.

If a name mapping file exists, then the N for the *name* panel field will include the entries from the first column of the **Basic** tab of the name mapping file. If an item is selected from the list, then the name, colour, model, style *etc.* from the columns in the name mapping file are used to fill out the panel fields

The name mapping file is pointed to by the environment variable

NAME_MAPPINGS_4D

or is if the environment variable is not set, the default name is **names.4d**.

During a ***12d Model*** session, the name mapping file can be changed by the *Name Settings* tab on the option **Utilities=>Defaults**. Note that this new name mapping file is not saved with the project.

Validation

After the panel field information is entered, an <enter> key requests that the panel field information be validated. Any error message will be displayed in the panel's message area. If there is no error, the cursor will move onto the next panel field. If an error occurs, the cursor will remain in the invalid panel field.

New panels or menus can be fired up before completing panels or menus already on the screen. This gives the user full control over the work flow, rather than being locked in by fixed sequences in a program.

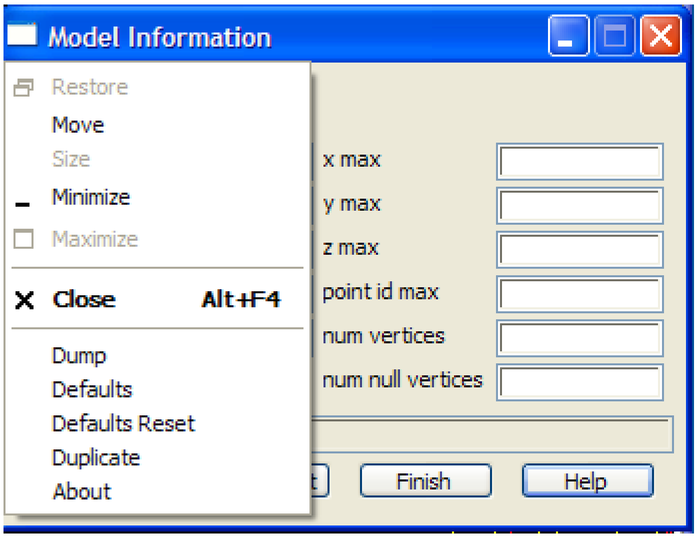
Continue to the next section 4.3.6.2 Panel System Menu or return to 4.3.6 Panels.

4.3.6.2 Panel System Menu

The **Panel System** menu is brought up by clicking LB on the windows icon on the left hand corner of the panel title area.

The **Panel System** menu has options to move, minimize, close, dump (write out an image of the panel), create defaults (ddx) file (ddx), reset a defaults (ddx) file and duplicate the panel.

.



For the option
Move, go to 4.3.6.3 Moving a Panel
Minimize/Maximize 4.3.6.4 Minimizing and Maximizing a Panel
Dump 4.3.6.5 Dumping a Panel, Creating a Screen Layout File or Default File
Close 4.3.6.10 Deleting a Panel
Defaults 4.3.6.6 Panel Defaults - ddx Files
Defaults Reset 4.3.6.7 Defaults Reset - Deleting a ddx File
Duplicate 4.3.6.8 Duplicating a Panel
Also for
*Special panel field*s 4.19.2 Pre*Postfix in Panel Fields
Expressions in panel fields 4.19.1 Expressions in Panel Fields
Resizing a panel 4.3.6.9 Resizing of Panels
OK panel 4.3.6.11 OK Panel
Yes-No panel 4.3.6.12 Yes-No Panel
Yes-No-cancel panel 4.3.6.13 Yes-No-Cancel Panel
Yes-No-All-cancel panel 4.3.6.14 Yes-No-All-Cancel Panel

TODO: Replace-Cancel
TODO: Append-Replace-Cancel

Continue to the next section 4.3.6.3 Moving a Panel or return to 4.3.6 Panels.

4.3.6.3 Moving a Panel

A panel can be moved (dragged) by **holding down LB** anywhere in the menu title area (except over the windows icon on the left and the minimize, restore and **[X]** buttons on the right) and then moving the cursor with the LB still depressed.

The panel (or panel outline) moves with the cursor to indicate where the new panel position will be, and the panel is finally positioned when LB is released.

Move can also be started by selecting **Move** from the **Panel System** menu and then holding down LB in the menu title area and continuing as described above.

Continue to the next section 4.3.6.4 Minimizing and Maximizing a Panel or return to 4.3.6 Panels.

4.3.6.4 Minimizing and Maximizing a Panel

The **panel** can be minimized (iconized)/restored and maximized using the standard Windows minimize/restore and maximize buttons on the top right hand corner of the panel.

A panel can also be minimized by selecting **Minimize** from the **Panel System** menu.

Continue to the next section 4.3.6.5 Dumping a Panel, Creating a Screen Layout File or Default File or return to 4.3.6 Panels.

4.3.6.5 Dumping a Panel, Creating a Screen Layout File or Default File

The panel can be written out to disk in a variety of images formats, or as a screen layout (slx) file or a default (ddx) file. This is called dumping the panel.

The **Dump** option is selected by clicking MB in the panel title area, or by selecting **Dump** from the **Panel System** menu which is invoked by clicking the left mouse button (LB) on the Windows icon on the left of the panel title area.

The **Menu/Panel Dump** panel is then displayed and for its description, go to 4.3.3.2 Dumping a Menu or Panel.

Continue to the next section 4.3.6.6 Panel Defaults - ddx Files or return to 4.3.6 Panels.

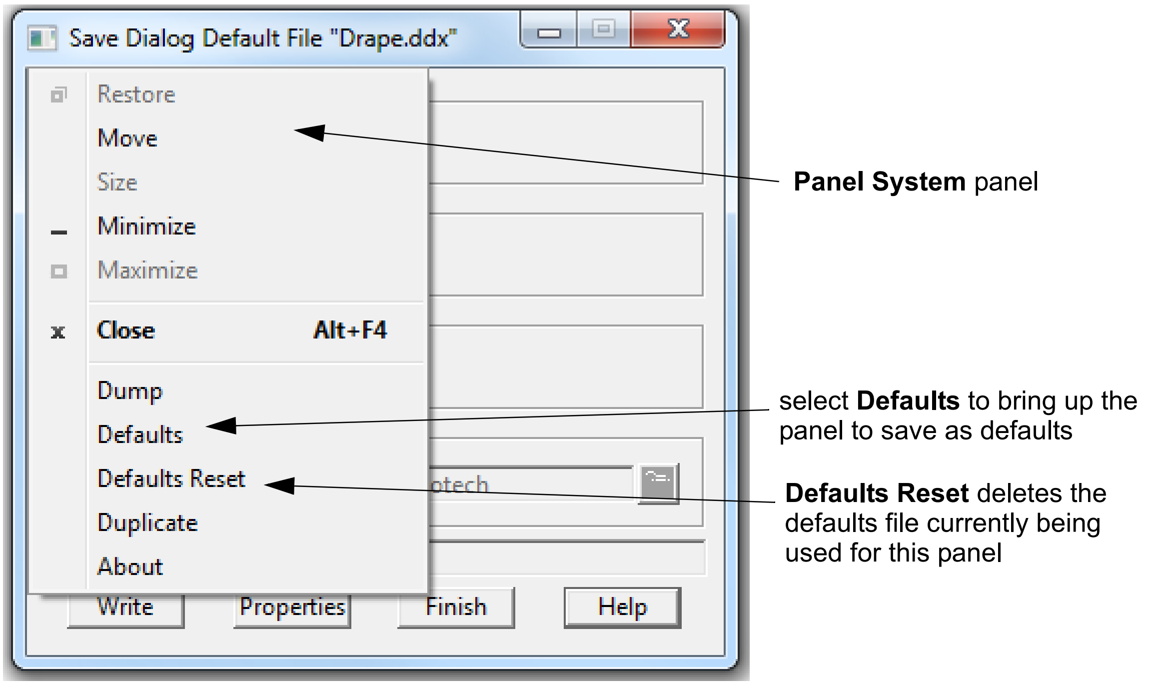
4.3.6.6 Panel Defaults - ddx Files

When a panel is opened, ***12d Model*** may set some default values.

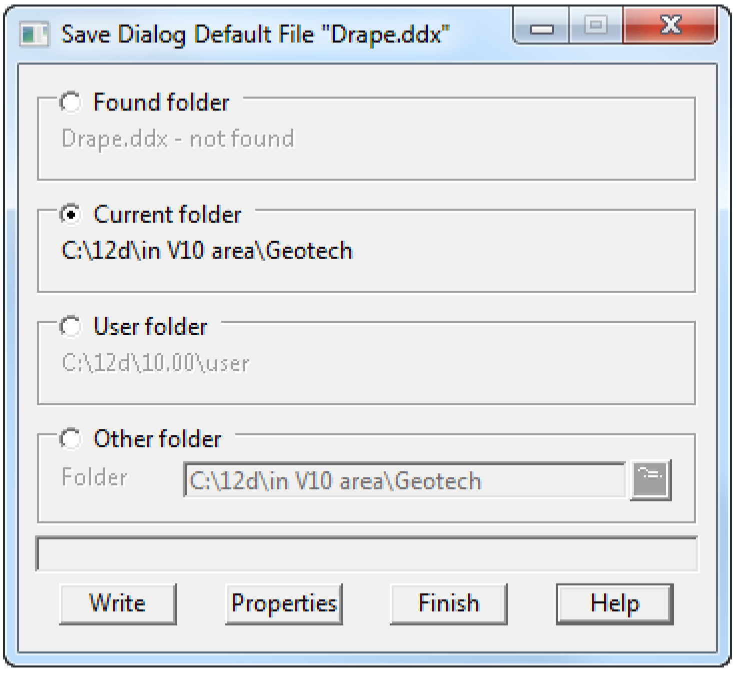
However for most panels ***the user can supply the values*** to be used ***for*** all the ***panel fields*** each time a panel is opened. This information is stored in a defaults file for a panel. The defaults file has the file ending .ddx (or a pre- ***12d Model 11*** format, .ddf).

If no defaults file exists for a panel, then the ***12d Model*** defaults are used for the panel.

To set user defined defaults for a panel, simply open the panel, fill in the default values that are required, and then bring up the **Panel System** menu (by clicking LB in the top left had corner of the panel) and select **Defaults**.



A panel comes up asking what folder to write the ***ddx*** file for the panel to.



Clicking on **Write** writes out the ddx file for the panel.

When a ***12d Model*** panel starts up, the standard paths are searched for a defaults file to use to set the default values for the panel fields.

If no defaults file is found for the panel, the ***12d Model*** system defaults are used.

Note - for ***12d Model 10*** and above, the defaults file has an xml structure and is written to a file ending in **.ddx**. An earlier format ended in **.ddf**. If both a .ddx and a .ddf file exists for the panel, then the .ddx file is used.

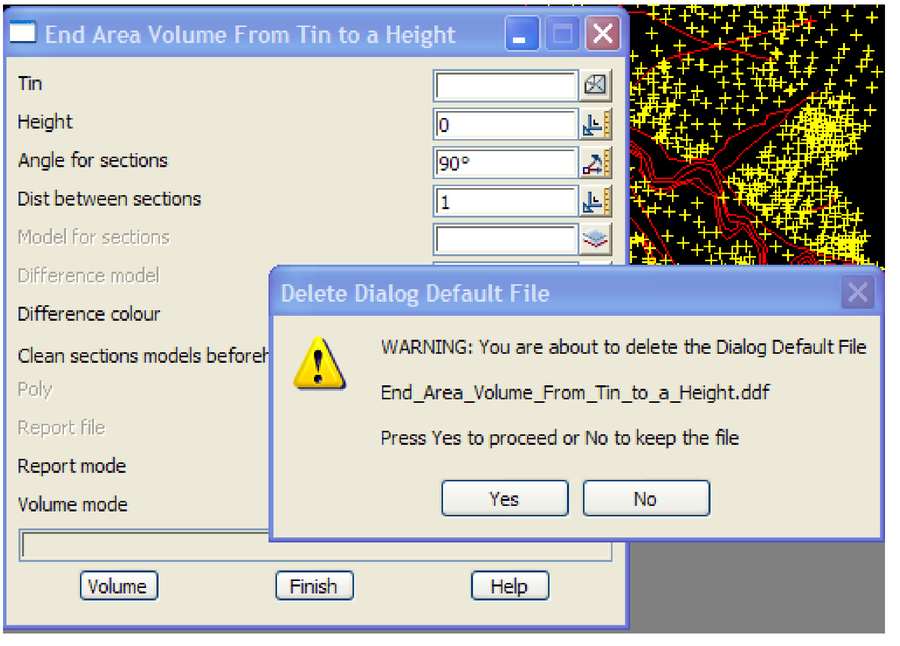
Continue to the next section 4.3.6.7 Defaults Reset - Deleting a ddx File or return to 4.3.6 Panels.

4.3.6.7 Defaults Reset - Deleting a ddx File

If user defined defaults are no longer required for a panel, simply bring up the panel and then the **Panel System** menu by clicking LB in the top left had corner of the panel.

Select **Defaults Reset** and the default ***ddx*** file for the panel is deleted.

.



Continue to the next section 4.3.6.8 Duplicating a Panel or return to 4.3.6 Panels.

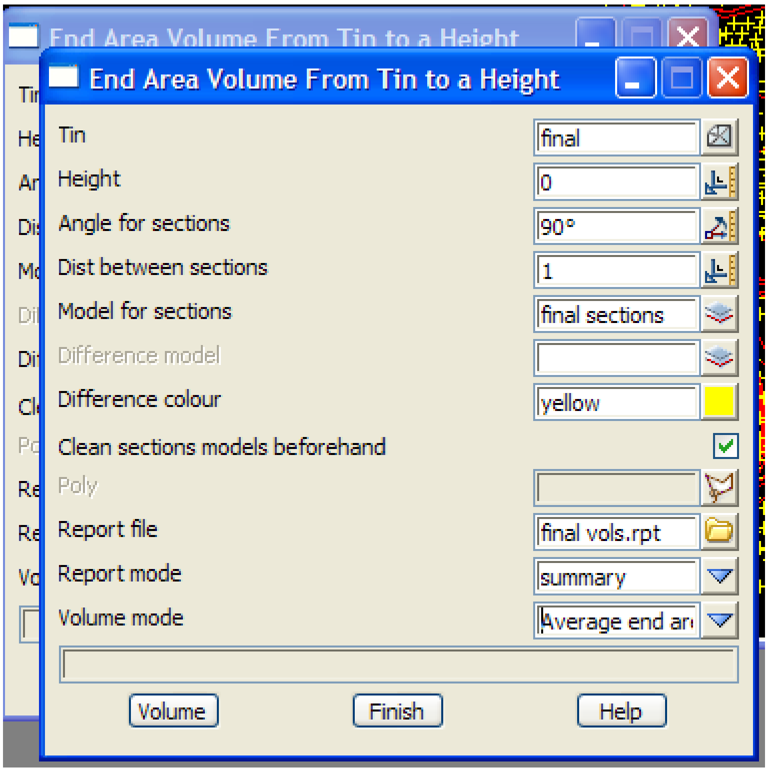
4.3.6.8 Duplicating a Panel

After filling in all the fields of a panel, it is often useful to have a copy of the panel with almost all the same panel field values.

For example, when calculating the volume from a tin to a height using end area, it is normal to run the option with a number of different angles for the sections, and maybe different distances between the sections.

To create a new panel with all the same values in the panel fields, bring up the **Panel System** menu for the panel and select **Duplicate.**

.

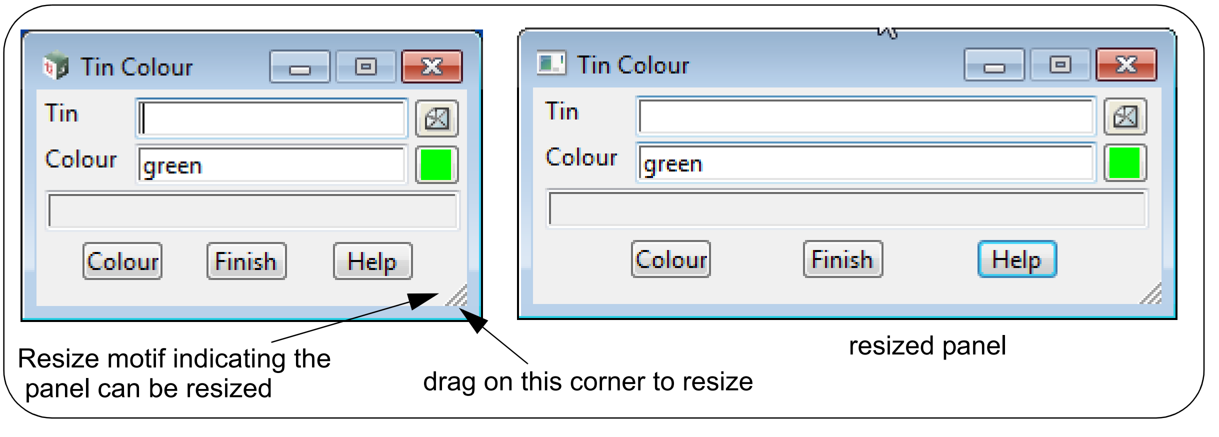


Continue to the next section 4.3.6.9 Resizing of Panels or return to 4.3.6 Panels.

4.3.6.9 Resizing of Panels

Most panels can be resized.

If a panel can be resized then there is an *Resize motif* (three lines) in the bottom right corner and dragging from that corner will resize the panel.



How the panel can be resized will depend on what items are in the panel. For example, the **Tin Colour** panel can only be made wider but cannot be made smaller or taller.

When a screen layout file (SLX) or a panel defaults file (DDX) is written out for a resized panel, or a resized panel recorded in a Chain, the resized panel size is recorded.

Continue to the next section 4.3.6.10 Deleting a Panel or return to 4.3.6 Panels.

4.3.6.10 Deleting a Panel

There are three methods are available for deleting a panel. Picking the **[X]** button in the panel title area, using the **finish** button or by selecting **Close** from the **Panel System** menu.

Continue to the next section 4.3.6.11 OK Panel or return to 4.3.6 Panels.

4.3.6.11 OK Panel

The **OK** panel requires the selecting of the button **OK**.

Continue to the next section 4.3.6.12 Yes-No Panel or return to 4.3.6 Panels.

4.3.6.12 Yes-No Panel

The **yes no** panel requires the selecting of the button **yes** or **no**.

Continue to the next section 4.3.6.13 Yes-No-Cancel Panel or return to 4.3.6 Panels.

4.3.6.13 Yes-No-Cancel Panel

The **yes no cancel** panel requires the selecting of the button **yes, no** or **cancel**.

Continue to the next section 4.3.6.14 Yes-No-All-Cancel Panel or return to 4.3.6 Panels.

4.3.6.14 Yes-No-All-Cancel Panel

The **yes no all cancel** panel requires the selecting of the button **yes, no, all** or **cancel**.

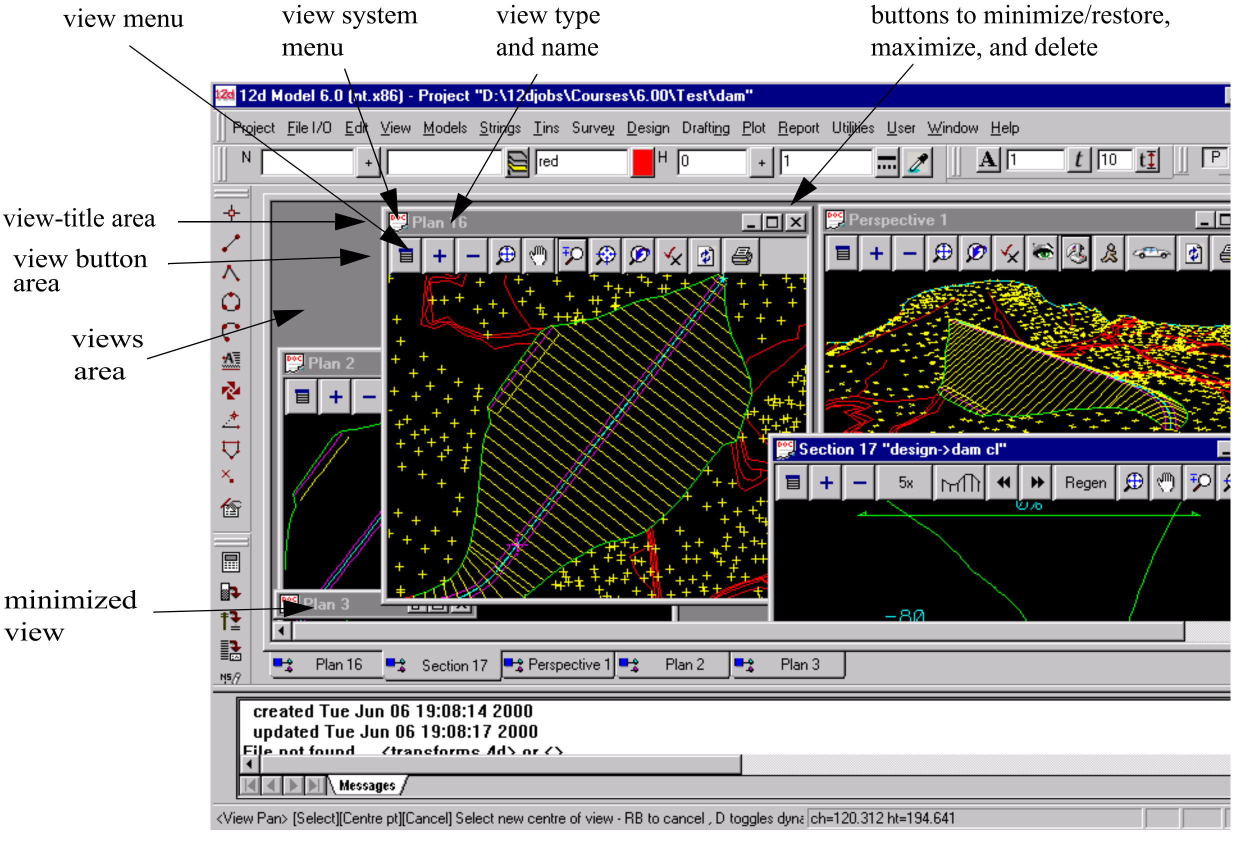
The next major section 4.3.7 Views or return to 4.3.6 Panels.

4.3.7 Views

Views are the screen drawing areas for ***12d Model*** and come in four flavours - plan, section, perspective and opengl perspective.

Views can be created and deleted as required by the user and there is no limit to the number of views on the screen. Views can be overlapped and minimized.

Each view has a unique name of up to two hundred characters. The ***view type*** and ***name*** are displayed on the top left corner of the view in what is called the **view-title area**.



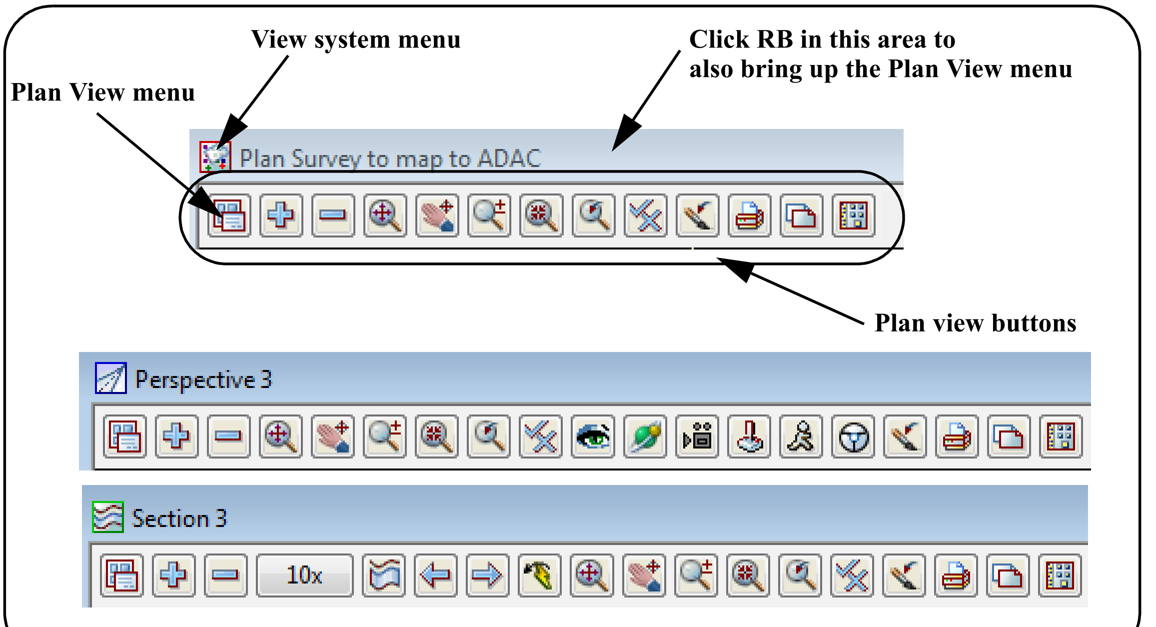
For information on:

View buttons, go to [4.3.7.1 View Buttons]
View menu [4.3.7.2 Plan/Section/Perspective View Menu]
View system menu [4.3.7.3 View System Menu]
View dump [4.3.7.4 Dumping a View]
Moving a view [4.3.7.5 Moving a View]
Resizing a view [4.3.7.6 Resizing a View]
Minimising/maximizing a view [4.3.7.7 Minimizing and Maximizing a View]
Deleting a view [4.3.7.8 Deleting a View]

4.3.7.1 View Buttons

On the row under the view type and name, are a number of options called **view buttons**.

The view buttons act like menu items and are activated by clicking LB when the cursor is above the button.



View Buttons

For the function of each View Button see

- 9.5.1 Plan View Buttons
- 9.5.2 Perspective View Buttons
- 9.5.3 Section View Buttons

Continue to the next section 4.3.7.2 Plan/Section/Perspective View Menu or return to 4.3.7 Views.

4.3.7.2 Plan/Section/Perspective View Menu

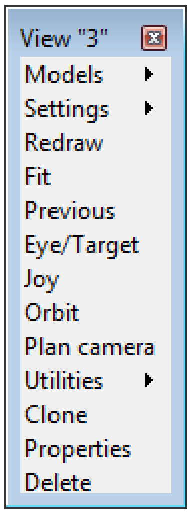
If LB is clicked on the **Menu** icon in the view buttons area, or RB is clicked in the *View Title area* or the *View Buttons area*, a new menu called the **Plan/Section/Perspective View** menu appears (or just **View** menu for short).

To remove the **Plan/Section/Perspective View** menu, select **[X]** on the **View** menu or click RB again in the *View Title area* or the *View Button area*.

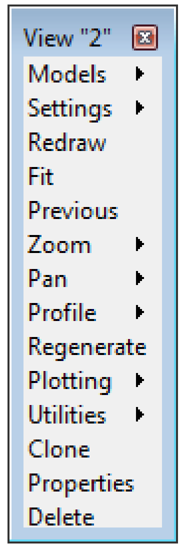
Because of the differences between plan, perspective and section views, the options on the **View** menu vary for each view type.

Plan View Menu

Perspective View Menu



Section View Menu



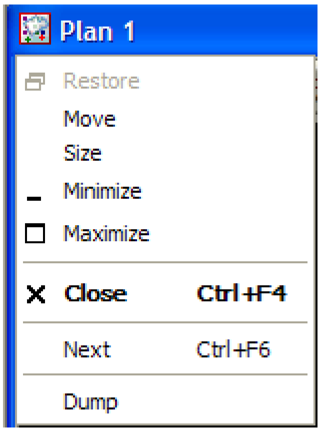
The **View** menus are discussed in the section 9 Menus on Views.

Continue to the next section 4.3.7.3 View System Menu or return to 4.3.7 Views.

4.3.7.3 View System Menu

The **view system** menu is brought up by clicking LB on the windows icon on the left hand corner of the *view title area*.

.



The **View system** menu has options to move, resize, minimize, maximize, close (delete) and dump the view.

For information on *Moving a view*, go to	4.3.7.5 Moving a View
Resizing a view	4.3.7.6 Resizing a View
Minimizing/maximizing a view	4.3.7.7 Minimizing and Maximizing a View
Closing a view	4.3.7.8 Deleting a View
Dumping a view image	4.3.7.4 Dumping a View

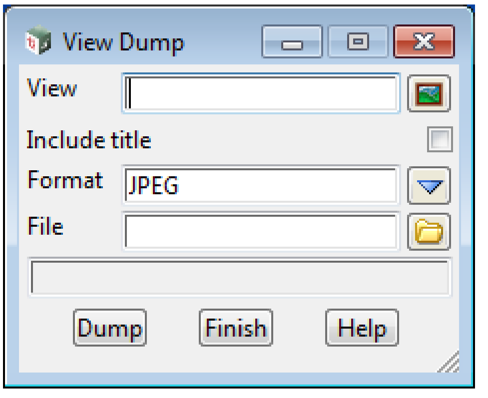
Continue to the next section 4.3.7.4 Dumping a View or return to 4.3.7 Views.

4.3.7.4 Dumping a View

An image of the view can be written out to disk in either bmp, tif, postscript format or as a screen layout (slx) file. This is called dumping the view.

The **Dump** option is selected from the **View System** menu which is invoked by clicking the left mouse button (LB) on the Windows icon on the left of the view-title area. The **View Dump** panel is then displayed

.



The fields and buttons have the following functions.

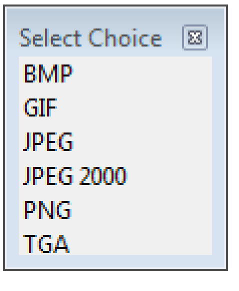
Field Description	Type	Defaults	Pop-Up
View	view box	current view picked from	available views

the view to dump out in the specified format.

Include title	tick box		

*if **ticked**, the view title area is included in the dump.*

Format	choice box	JPEG	BMP, GIF, JPEG, JPEG 2000, PNG, TGA



the format to dump the view out in.

File	file box	*.jp*g	

the file to dump the view images out to

Dump	button		

dump in the given format the image of the view given in the **View** *field to the file given in the* **File** *field.*

Continue to the next section 4.3.7.5 Moving a View or return to 4.3.7 Views.

4.3.7.5 Moving a View

A View can be moved (dragged) by **holding down LB** anywhere in the view-title area (except over the windows icon on the left and the minimize, restore and **[X]** buttons on the right) and then moving the cursor with the LB still depressed. Or **Move** can be selected from the **View System** menu.

The View moves with the cursor to indicate where the new view position will be. The view is finally positioned when LB is released.

Continue to the next section 4.3.7.2 Plan/Section/Perspective View Menu or return to 4.3.7 Views.

4.3.7.6 Resizing a View

A **View** can be resized using the standard Windows resize methods of holding down LB on the view border and moving the mouse to the new position for the view border. and then releasing LB.

Continue to the next section 4.3.7.7 Minimizing and Maximizing a View or return to 4.3.7 Views.

4.3.7.7 Minimizing and Maximizing a View

A **View** can be minimized (iconised), restored and maximized using the standard Windows minimize/restore and maximize buttons on the top right hand corner of the view, or by selecting **Minimize** or **Maximize** from the **View System** menu.

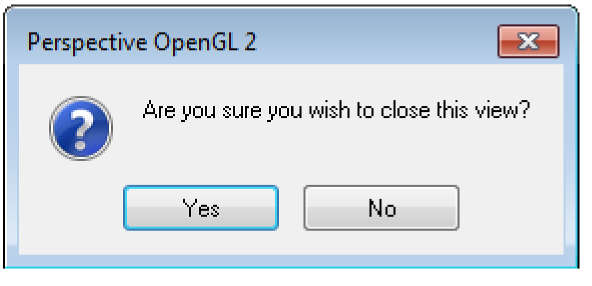
Continue to the next section 4.3.7.8 Deleting a View or return to 4.3.7 Views.

4.3.7.8 Deleting a View

Picking the **[X]** button in the *view-title area* or selecting **Close** from the **View System** menu will delete the view.

If the environment variable ***Prompt on close*** is set on, then a prompt will be displayed to confirm that the view is to be closed./deleted

.



Please continue to the next major section 4.3.8 Status Bar or return to 4.3.7 Views.

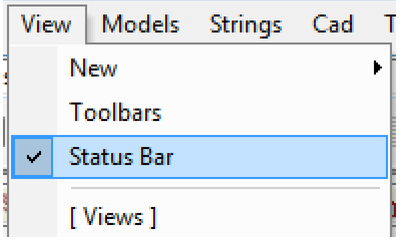
4.3.8 Status Bar

The Status Bar is used to display many of the messages generated by ***12d Model*** options (each panel also has its own special message area) and the dynamic coordinate position of the cursor as it moves around in any view on the screen.

The messages are displayed on the left hand side of the Status Bar and the view coordinates on the right hand side of the Status Bar.

The coordinate values displayed depends on the view type. For example, the world coordinate (x,y) position of the cursor is displayed when inside plan views, and the (chainage,height,x,y) position when in a section view.

The Status Bar can be toggled on/off by the option **View** => **Status Bar** on the Main menu.

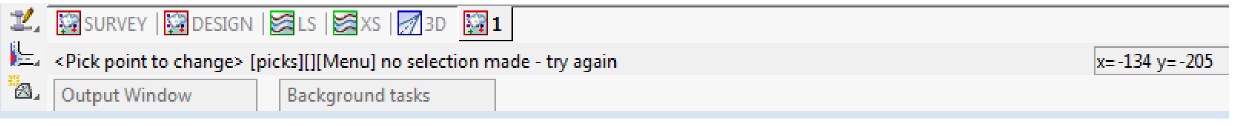


Please continue to the next section 4.3.9 Output Window.

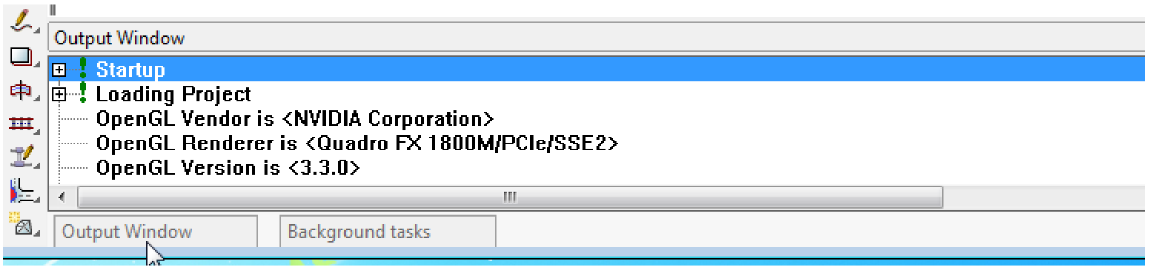
4.3.9 Output Window

The **Output Window** displays ***12d Model*** system and error messages.

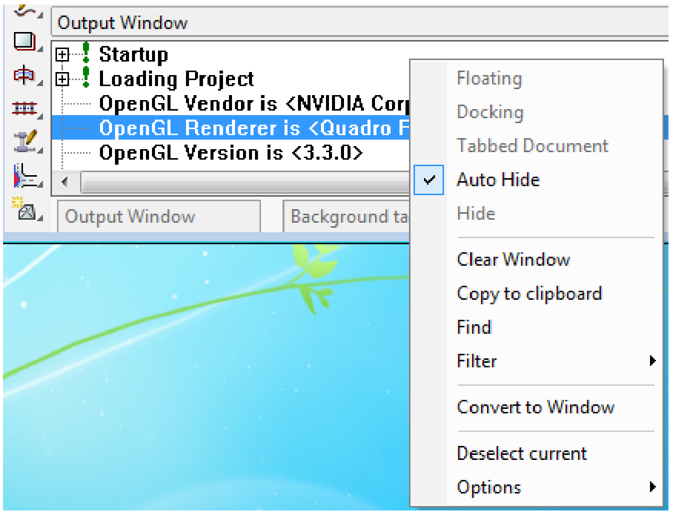
By default the **Output Window** is in **Auto-Hide** mode and appears as a tab at the bottom left of the screen and flashes if there are any messages that need to be reviewed.



When in *Auto-Hide* mode, as you move your cursor over the Output Window tab the Output Window appears.



Auto-Hide mode can be turned off by moving over the Output Window area and pressing RB to bring up the **Output Window** menu. Click on **Auto-Hide** to remove the tick and *Auto-Hide* is no longer on.



When **Auto-Hide** is turned **off**, the Output Window stays open.

The **Output Window** menu includes the options:

Clear - clears the Output Window,

Copy to clipboard - copy any selected text in the Output Window to the Clipboard.

Hide - removes the Output Window.

Float - makes the Output Window a floating window that can be docked on any of the sides of the 12d Model screen.

Convert to window - turns the Output Window into a normal Window which then appears on your desktop. It may be moved by clicking LB in the Output Window heading area, then dragging the cursor to another part of your desktop and releasing the LB to pin it down.

When the Output Window is a normal Window or a floating window, clicking on **x** at the top right of the window will remove the Output Window.

The Output Window can be made taller or shorter by moving the bar at the top of the Output Window.

The Output Window can be removed by **Hide** but also unticking the Output Window on the Window Main Menu will remove the Output Window.

If an **error message** is sent to the Output Window and the Output Window it is removed (**turned off**) then the Output Window is **automatically toggled on** to indicate an error has occurred.

Once the Output Window is removed and no error messages have been sent to it, the only way to turn it back on is to click on **Output Window** on the **Window** Main Menu.

Message lines which are intelligent log lines are begun with a green exclamation mark!

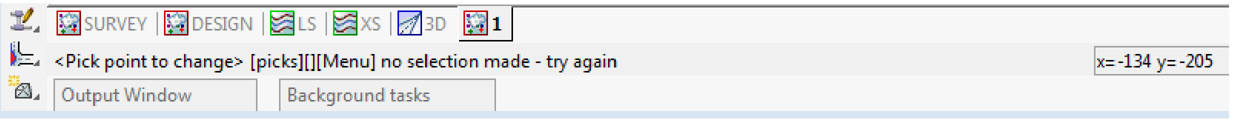
Please continue to the next section 4.3.10 Background Tasks Window.

4.3.10 Background Tasks Window

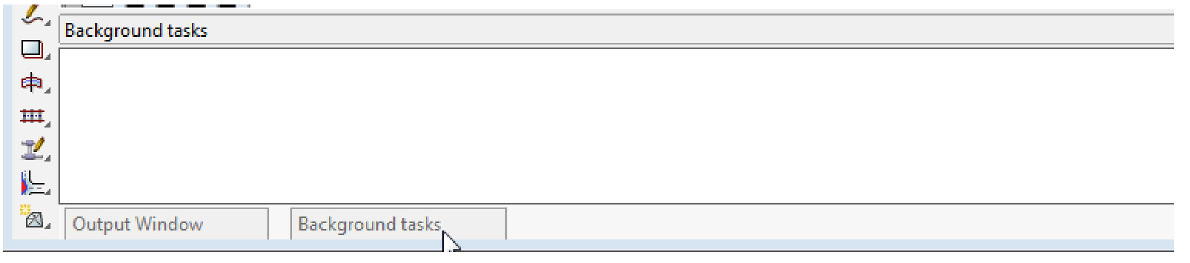
12d Model uses multi-cores and background processes to make things faster for the user.

The **Background Task** window displays messages from tasks running in the background.

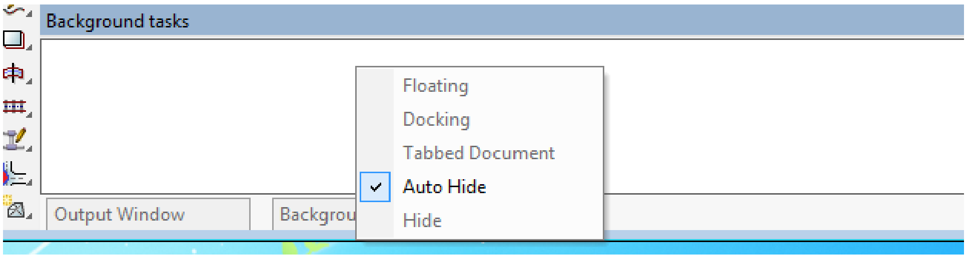
By default the **Background tasks** window is in **Auto-Hide** mode and appears as a tab at the bottom left of the screen and flashes if there are any messages from a background task that need to be reviewed.



When in *Auto-Hide* mode, as you move your cursor over the **Background Task** tab the **Background tasks** window appears.



Auto-Hide mode can be turned off by moving over the Background Tasks window area and pressing RB to bring up the **Background Tasks** menu. Click on **Auto-Hide** to remove the tick and *Auto-Hide* is no longer on.



When **Auto-Hide** is turned **off**, the Background Tasks window stays open.

The **Background Tasks** menu includes the options:

Hide - removes the Background Tasks window.

Float - makes the Background Tasks window a floating window that can be docked on any of the sides of the 12d Model screen.

When the **Background Tasks** window is a normal window or a floating window, clicking on **x** at the top right of the window will remove the Background Tasks window.

The Background Tasks window can be made taller or shorter by moving the bar at the top of the Background Tasks window

The Background Tasks window can be removed by **Hide** but also unticking the Background Tasks on the **Window** Main Menu will remove the **Background Tasks** window.

If a message is sent to the Background Tasks window and the Background Tasks window is removed (**turned off**) then the Background Tasks window is **automatically toggled on** to indicate that there is a background task message.

Once the Background Tasks window is removed and no messages have been sent to it, the only way to turn it back on is to click on **Background Tasks** window on the **Window** Main Menu.

Message lines which are intelligent log lines are begun with a green exclamation mark!

Please continue to the next section 4.4 Ascii, Ansi and Unicode.

4.4 Ascii, Ansi and Unicode

From ***12d Model 10*** onwards, text is stored in the ***12d Model*** database as Unicode (UTF-16 Unicode) and the default format for all output files produced by ***12d Model*** is for them to be Unicode files.

But what does that mean?

Computers can only understand numbers (only zeros and ones actually), so a common code is needed for the numerical representation of characters such as 'a' or '1' or some action such as TAB and a number of common codes have evolved over time.

The common code is not only needed for text in a file or text on a Web page, but also for the names of the files and folders on a computer disc or an internet site.

See 4.4.1 ASCII Character Set
4.4.2 ANSI Character Set
4.4.3 Unicode Character Set
4.4.4 Unicode Encoding: UTF-8
4.4.5 Unicode Encoding: UTF-16
4.4.6 Endian and BOM
4.4.7 Writing out Files from 12d Model

4.4.1 ASCII Character Set

The ASCII (American Standard Code for Information Exchange) was first published in 1963 and was adopted by the American National Standards Institute (ANSI) during the 1960s and has been in common use since then.

The ASCII definition used 7 bits to define characters and some non character codes such as tab, back space and line feed (new line). The seven bits means that only a maximum of 127 codes are allowed.

An examples of the ASCII codes are:

2 is the ASCII code for start of text (STX)

8 is the ASCII code for back space (BS)

9 is the ASCII code for horizontal tab (TAB)

10 is the ASCII code for line feed, new line (NL)

27 is the ASCII code for escape (ESC)

32 is the ASCII code for a space (" ")

36 is the ASCII code for a dollar sign **$**

40 is the ASCII code for a left parenthesis **(**

41 is the ASCII code for a right parenthesis **)**

48 is the ASCII code for the digit zero **0**

49 is the ASCII code for the digit zero **1**

65 is the ASCII code for the Latin capital letter A **A**

97 is the ASCII code for the Latin small letter a **a**

126 is the ASCII code for a tilde **~**

127 is not used

Even with the newer standards, the 7-bit ASCII table continues to be the backbone of modern computing and data storage. Is is so ubiquitous that the terms "text file" and "ascii file" have come to mean the same thing for most computer users.

The ASCII standard was good, as long as you were only working in US English.

Go to the next section 4.4.2 ANSI Character Set or back to 4.4 Ascii, Ansi and Unicode.

4.4.2 ANSI Character Set

The ANSI standard extended the ASCII character set. In the ANSI standard, the first 128 characters where the same as for ASCII but from character 128 onwards, there were different ways depending on where you lived. These different ways were called ***code pages***.

For example, in Israel DOS used a codepage called 862 while Greek users used code page 737.

The ANSI set of 218 characters (also know as Windows-1252) was the standard for core fonts supplied with US versions of Microsoft Windows up to and including Windows 95 and Windows NT 4 (character 218 was the euro currency symbol was added during this time).

ANSI characters 32 to 127 correspond to those in the 7-bit ASCII character set.

Some of the extra ANSI codes are:

163 is the ANSI code for a currency Pound sign

165 is the ANSI code for a currency Yen sign

If you use a version of Windows that is designed for a non-Latin alphabet such Arabic, Cyrillic, Greek or Thai to view a document that has been typed using the ANSI character set, then in the codepage for the characters from these languages may replace some of those in the 128-255 range and so the document will look different.

There are similar problems when transferring ANSI documents to DOS or Macintosh computers, because DOS and MacRoman arrange characters differently in the 128-255 range.

Go to the next section 4.4.3 Unicode Character Set or back to 4.4 Ascii, Ansi and Unicode.

4.4.3 Unicode Character Set

Today people want to transfer information around the world in emails and on Web sites but the ASCII and ANSII character sets can not work with a variety of Latin and non-Latin alphabets in the one document.

The solution is to move to a system that assigns a unique number to each character in each of the major languages of the world. Such as system has been developed and is known as **Unicode** and it is intended to be used on all computer systems, not just Windows.

The Unicode Standard covers more than 110,000 characters covering 100 scripts, a set of code charts for visual reference, an encoding methodology and set of standard character encodings, an enumeration of character properties such as upper and lower case, a set of reference data computer files, and a number of related items such as character properties, rules for normalisation, decomposition, collation rendering and bidirectional display order (for the correct display of text containing both right-to-left scripts such as Arabic and Hebrew and left-to-right scripts such as English). As of 2012, the most recent version is ***Unicode 6.1***

Unicode's success at unifying character sets has led to its widespread use in computer software and the standard has been implemented in XML, Java, Microsoft .NET Framework and modern operating systems.

To make it Unicode compatible with ASCII, the first 128 characters where the same as for ASCII but from character 128 onwards they are totally different.

All the Unicode characters can be covered with 32 bits but to use a 32-bit representation in a file means that a standard ASCII file would be four times as large when written out in Unicode.

So to save on disk space, and the size of files for emailing etc, there are a number of different mapping methods, or character encodings, for writing Unicode characters to a file.

The Unicode standard defines two mapping methods: the Unicode Transformation Format (UTF) encodings, and the Universal Characters Set (UCS) encodings. An encoding maps the range of Unicode characters (or possibly a subset) to sequences of values in some fixed-size range.

Note: Even though software stores Unicode characters, the computer system still needs the graphics for the character sets to be able to correctly display the Unicode characters.

Go to the next section 4.4.4 Unicode Encoding: UTF-8 or back to 4.4 Ascii, Ansi and Unicode.

4.4.4 Unicode Encoding: UTF-8

One of the most common character encodings is UTF-8.

In UTF-8 encoding, only 8-bits are used for any ASCII characters from 0 to 127. For characters 128 and above, it uses between 16, 24 and up to 48 bits.

And because the representation of the first 128 characters are the same in Unicode and ASCII, US English text looks exactly the same in UTF-8 as it did in ASCII.

So why can't a standard ASCII text editor, or a program requiring plain ASCII text have problems with a Unicode file just containing ASCII characters?

The main reason is that in many Unicode files, a special character called a BOM (see 4.4.6 Endian and BOM) is often placed at the beginning of the file, and the BOM would not be recognised by a program only expecting ASCII and would generate an error, or show up as blank spaces or strange-looking characters.

Go to the next section 4.4.5 Unicode Encoding: UTF-16 or back to 4.4 Ascii, Ansi and Unicode.

4.4.5 Unicode Encoding: UTF-16

In UTF-16 encoding, 16-bits are the basic unit and depending on the Unicode character, UTF-16 encoding may require one or two 16-bit code units. Using the two 16-bit code units, UTF-16 is capable of encoding up to 1,112,064 numbers.

The basic unit of computers is a byte which consists of 8-bits. Because the UFT-16 encoding uses 16-bit and so is made up of two bytes and the order of the bytes may depend on the endianness (byte order) of the computer architecture.

To assist in recognizing the byte order of code units, UTF-16 allows a Byte Order Mark (BOM - see 4.4.6 Endian and BOM), a code with a special value to precede the first actual coded value.

Because the fundamental unit in UFT-16 is 16 bits, storing a text file only containing ASCII text will take twice as much disk space as the ASCII version.

Microsoft has used UTF-16 for internal storage for Windows NT and its descendents including Windows 2000, WIndows XP, Windows Vista and Windows 7.

Go to the next section 4.4.6 Endian and BOM or back to 4.4 Ascii, Ansi and Unicode.

4.4.6 Endian and BOM

From early computing, the fundamental unit of storage was a byte consisting of 8-bits (a bit is a one or a zero). When computers started using 16-bits, this could be stored as two bytes but there was a choice of the order of storing the two bytes. Two different approaches arose and are referred to the endian or endianness.

Big endian stores the most significant byte first and the least significant byte second. Similar to a number written on paper. ***Little endian*** stores the least significant byte first and the most significant byte second.

The ***byte order mark*** (BOM) is a Unicode character used to signal endianness (byte order) of a text file or character stream.

A BOM is essential when the basic unit of an encoding consists of two bytes such as in UTF-16.

Beyond its specific use as a byte-order indicator, the BOM character may also indicate which of the Unicode encoding has been used because the values of the bits in the BOM will be different for the different Unicode encodings.

So although a BOM is not strictly necessary for UTF-8 when it only contains ASCII data, it still alerts the software that it is UTF-8.

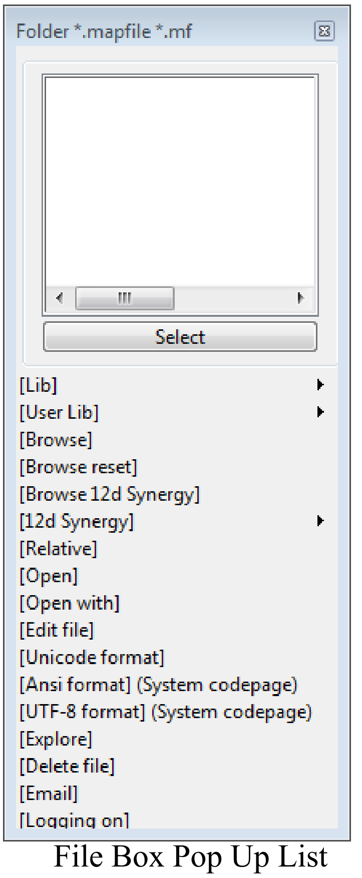
Some common programs from Microsoft, such as Notepad and Visual C++, add BOMs to UTF-8 files by Default. Google Docs adds a BOM when a Microsoft Word document is downloaded as a .txt file.

When a BOM is used, it should appear at the **start** of the text.

Go to the next section 4.4.7 Writing out Files from 12d Model or back to 4.4 Ascii, Ansi and Unicode.

4.4.7 Writing out Files from 12d Model

When a file box is used in a ***12d Model*** panel, clicking **LB** of the ***Folder*** icon will bring up the folder pop-up and on the pop-up is the choice for writing out the file in ANSI, or UTF-8 rather that the default of Unicode (UTF-16).



File Box Pop Up List

For more information on the **File Box**, see 4.19.8 File Box.

Continue to the next section 4.5 Data Types (or return to 4.4 Ascii, Ansi and Unicode).

4.5 Data Types

12d Model was designed from the ground up as an object based system and programmed in C++, the most widely used Object Oriented programming language.

As objects, strings know what type of string they are and behave accordingly without user intervention. As a simple example, when selecting a string for editing, the string knows what type of string it is and brings up the appropriate editor.

A more complicated example is a super alignment with computations. Such a string can be constructed from references to other string, automatic fillets, automatic draping on tins *etc.* and if any of this auxiliary information changes, the super alignment will change as well.

See

4.5.1 Strings
4.5.2 General String Properties
4.5.3 Models
4.5.4 Tins
4.5.5 Trimeshes
4.5.6 Templates
4.5.7 Projects
4.5.8 Attributes or Meta Data

4.5.1 Strings

12d Model uses the string object as one of its basic modelling elements.

In its simplest form, a ***12d Model*** string is an *ordered* series of vertices or points, joined by a segment.

However a ***12d Model*** string can also be much more complex including referencing to other objects and complicated construction methods. The string object can also have an almost unlimited number of user defined attributes for the whole string, for each vertex and for each segment joining vertices.

However, as an object, the ***12d Model*** string shields the user from its full complexity and only displays the relevant information and properties as required for a particular context.

So starting with the basics, a ***12d Model*** string is an *ordered* series of vertices or points.

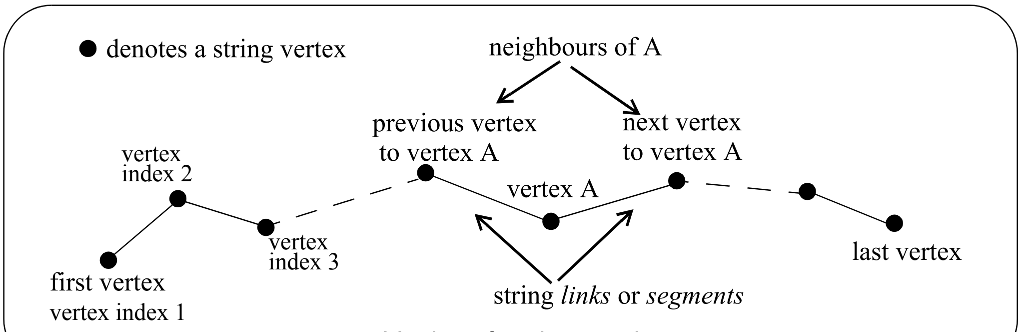
Apart from the first and last vertex in a string, each vertex in a string has a unique ***previous vertex*** (predecessor) and a unique ***next vertex*** (successor). The previous and next vertices for a vertex are called its ***string neighbours***.

The objects joining a vertex with its neighbours are called string links or ***segments***. Segments may be straight lines or plan arcs, or transition elements such as spirals. The segments can be *visible* or *invisible*.

The string has an implied **direction** starting from the first vertex of the string and then proceeding to the successor vertices in the string.

The ordering along a strings allows each vertex to be given a ***vertex index*** where the first vertex has vertex index **1**. Note that if additional vertices are inserted or vertices deleted then the vertex indices will change. Also note that for super strings, a vertex may have a ***point id*** (also called a ***vertex id*** *and sometimes a* ***point number***) which is ***not*** the same thing as the *vertex index*.

A string which has the same first and last vertex is called a ***closed*** string otherwise a string is said to be ***open***.



Vertices forming a string

Strings are very useful in the modelling of terrain and design surfaces and when using strings, for some applications they need to a particular properties.

For example, a ***contour*** displayed on maps is actually a string with the special property that all the vertices have exactly the same height (z value). And if the height of one vertex of the contour is modified, then all the vertices of the contour string must also be modified to have exactly the same height or the string is no longer a contour. So a ***contour*** string is a special type of string that ***knows*** how to behave when the z-value of the string is modified. This is why strings are ***objects*** - they intrinsically know how to behave.

To make it easier to refer to particular sets of properties that some strings must have, ***12d Model*** defines a number of different **types** of strings. Using the string ***type*** is a a convenient short hand for saying the string has certain well defined properties.

See Super string
2d or contour super string
3d super string
4d super string
alignment string
super alignment string
arc string
circle string
control station
drainage and sewer string
feature string
interface string
pipe super string
pipeline string
polyline super string
text string
4.5.2 General String Properties

Super string

A super string is a general purpose string.

Each vertex can have tinability, a symbol with its own size and rotation, z-value, text and an unlimited number of user definable attributes.

Segments can be lines, arcs or transitions (for example spirals). Each segment (link) has visibility, colour and breakline flags (segment tinability) and values for diameter or width and height (box culvert), radius, text and an unlimited number of user definable attributes.

2d, 3d, 4d, polyline and pipe strings are special cases of the super string.

2d or contour super string

The z-value of a 2d super string is the same at every vertex in the string.

Consequently the entire string has a height (the ***string height***) rather than having to refer to the height at each vertex. By changing the string height, the heights of all the vertices are modified. The height of a single vertex of a *2d* string ***can not*** be changed to a height that is different to the string height.

So if a ***2d*** string is selected by the string editor, as an object the ***2d*** string can signal the object editor that it has a ***string height*** and that the editors height command can only work on the string height and won't allow the individual vertex heights to be modified.

3d super string

The z-value can vary for each vertex in the string.

So if a ***3d*** super string is selected by the string editor, as an object the ***3d*** string can signal the object editor that each vertex has an individual height, and that the editors height command must select an individual vertex and modify that vertex's height.

4d super string

A ***4d*** super string has (x,y,z) values at each vertex ***plus*** vertex text defined at each string vertex. Useful when a description is needed at each vertex. For example, design sections are created as 4d strings and the vertex text records the name of the string that goes through that vertex.

alignment string

An ***alignment*** string is a more complicated string object which defines the string by specifying the horizontal and the vertical geometry for the string.

The horizontal geometry of the string is a a series of horizontal intersection points (HIP's) with plan arcs and leading (left) and trailing (right) transitions defined for each HIP. This defines the plan geometry of the alignment string.

The vertical geometry of the alignment string is defined as a series of vertical intersection points (VIP's) defined by horizontal chainage along the string and height. Each VIP can have a parabolic or circular curve on it.

Alignment strings are used for defining entities such as the centre line or a simple road or railway.

The alignment string has been superseded by the super alignment string which not only has IP methods but also fixed and floating definitions of elements.

super alignment string

A ***super alignment*** string is also specified by defining the horizontal and the vertical geometry for the string but neither of the geometries is restricted to just intersection (IP) points. The horizontal geometry can be made up of almost any combination of straights, arcs, transitions, partial transitions, horizontal IP's. Similarly the vertical geometry is made up of combinations of straights, arcs and parabolas.

In the super alignment object, the elements of the horizontal and vertical geometries can be defined by referencing to other strings, offsets, intersections *etc.* and if these items change, the super alignment object knows how to re-calculate itself.

arc string

An arc string is an arc in a plan projection but with a linearly varying z-value on the circumference. So the arc string is actually a 3d helix. The centre point of the arc is for display purposes only.

circle string

A circle string is a plan circle a constant z-value around the circumference. Hence the circle string is always parallel to the (x,y) plane. The centre point of the circle is for display purposes only.

control station

control stations are used in the survey reduction option where a name of a station can be given in the 12d survey field file and the coordinates of the station can given by the control station of that name in the specified model of control points. In ***12d Model***, a Control station is represented by a one vertex 4d or super string and the name of the control station is the name of the one vertex string. Usually the point id for the vertex of the super string, or the text for the 4d string, is also the Control station name.

drainage and sewer string

Special string made up of straights, arcs and manholes.

feature string

A feature string is a plan circle with a z-value at the centre point but only null values on the circumference.

If a feature string is given a ***world*** line style, then the style is centred on the centre point of the feature string and scaled up to the radius of the feature string.

If a feature string is given a ***screen*** or ***paper*** line style, then the style is wrapped around the circumference of the feature string.

interface string

An interface string is a special string constructed from a string and a surface. Each vertex of the interface string records whether the corresponding vertex on the original string was above or below the surface.

pipe super string

A pipe string is a 3d string with a diameter.

pipeline string

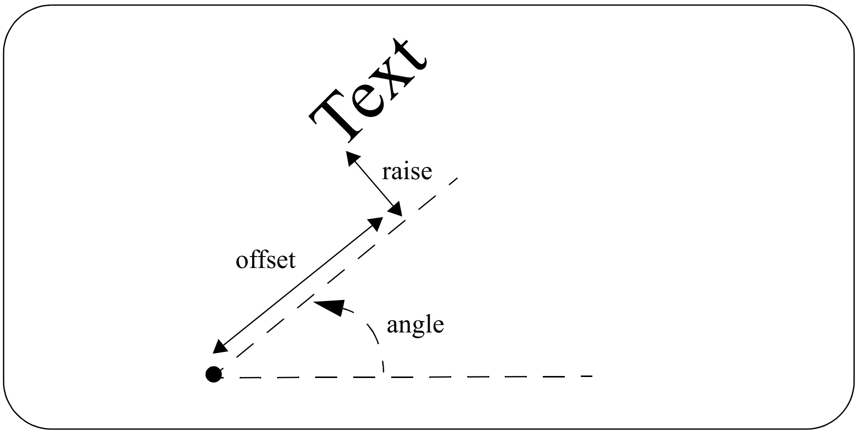
A pipeline string is an alignment string with a diameter.

polyline super string

A polyline string is similar to a 3d string except that there can be straights or arcs joining the string vertices.

text string

a text string has an (x,y) position and contains characters and information about how the characters are displayed (textstyle, units, height, offset, raise, justification, angle, slant and x factor). For more information on angle, offset and raise, go to the section 4.6 Text Definitions.



Super strings can also have text defined on its vertices and segments, and 4d strings can have text defined on each vertex,

Go to the next section 4.5.2 General String Properties or back to 4.5 Data Types.

4.5.2 General String Properties

See Name and Model
Breakline Type
Tinable
Linestyle and Colour
String Chainage

Name and Model

Strings have a text name of up to two hundred alphanumeric characters, spaces and dots (**.**). The string name does not have to be unique and can be blank. When strings are created, they are stored in **models**. A string is in one and only one model. Models are discussed in the next section.

Breakline Type

When triangulating data, if all the vertices and all the segments of a string are to be preserved as the sides of triangles and hence are part of the triangulation the string is called a **breakline** or **line** string. **Note** - all the segments of all the breaklines can only be preserved with there are no crossing segments amongst any of the breakline strings

If all the vertices are included in the triangulation but none of the segments are, then the string is called a point string.

For triangulation purposes, each of the super string types 2d, 3d, 4d and polyline can be used as **either** breaklines (line strings) or point strings.

Line strings are useful in describing terrain features such as ridge lines and creek beds or design features such as the edge of a building platform. Point strings can represent information such as spot heights.

Alignments, arcs, circles, pipelines, interfaces, drainage and sewer strings can only be breakline strings. Feature strings can only be point strings with a z-value at the centre of the circle.

Tinable

For a ***super string***, the concept of breakline has been **extended** to a property called **tinable** which can be set *independently* for each vertex and each segment of the super string.

If a **vertex** is tinable, then the vertex is included in triangulations. If the vertex is not tinable, then the vertex is ignored when triangulating.

If a **segment** is tinable, then the segment is used as a side of a triangle during triangulation. That is, the segment is used as a breakline. This may not be possible if there are *crossing* tinable segments.

Note that for a segment to be used as a side of a triangle, then its end vertices must also be tinable.

So a ***breakline string*** is one where all the vertices and all the segments are tinable.

A **point** string is one where all the vertices are tinable and all the segments are not tinable.

Linestyle and Colour

All strings (except text) can be given a user defined **linestyle** or simply **style**.The style describes how the string is drawn on the screen and on plots. The default style is **1**.

A style can also be continuous or just at the vertices of a string, regardless of the breakline type of the string. A breakline string (line string) with default style **1** will be drawn with solid lines between the nodes, and a non breakline string (point string) with default style will be drawn with crosses at the string nodes with no visible lines between the nodes.

Strings have a default colour but how that colour is used depends on the breakline type and style of the string. For example, a line string with default style **1** is drawn with the string's links in the string colour and for a point string with default style **1**, crosses are drawn at each string point in the string colour. However, styles can have their own colours which override the default colour of the string. Interface strings have two colours (red and green) which are used to represent cut and fill information.

Hence super strings with **default styles** appear on the screen as

point string (non breakline string)

each string link is considered to be an **invisible** line. String values are not defined along the invisible string links.

line string (breakline string)

each string link is considered to be a **solid** line. String values are defined along the string links by linear interpolation between the end points of the string link.

String Chainage

Every vertex along a string has a **chainage** value. This chainage is calculated by taking the start chainage defined for the first vertex of the string, and adding to it the **plan** distance along the string from the start vertex to the selected vertex on the string.

Go to the next section 4.5.3 Models or back to 4.5 Data Types.

4.5.3 Models

Within a ***12d Model*** project, information is collected in units called **MODELS**. Models contain strings, tins, super tins, grid tins, trimeshes and plot frames. (see the next section).

Each model has a unique user-defined text name of up to two hundred alphanumeric characters and spaces.

For convenience, model names should reflect the nature of the information in the model. For example, a model containing terrain data could be called **terrain**. **design** could be another model containing design data.

Each model has minimum and maximum x, y and z values which define a bounding box which encloses all the data in the model (the model bounding box). The size of the model bounding box is automatically updated as new data is added to the model, but not adjusted as data is deleted. A re-calculation of the model bounding box is performed by the **calc extents** buttons on the model and view information panels.

There is an option to list all the models available in a ***12d Model*** session (the model list). The model list is also used in various pop-ups and walk-right menus.

Displaying Models

The screen display areas in ***12d Model*** are the views. Models are displayed in views by “adding” the model to the view. Similarly, when a model is “removed” from a view, it is no longer displayed in that view.

There are options in the ***12d Model*** menu and on each view to add and remove models from views.

Go to the next section 4.5.4 Tins or back to 4.5 Data Types.

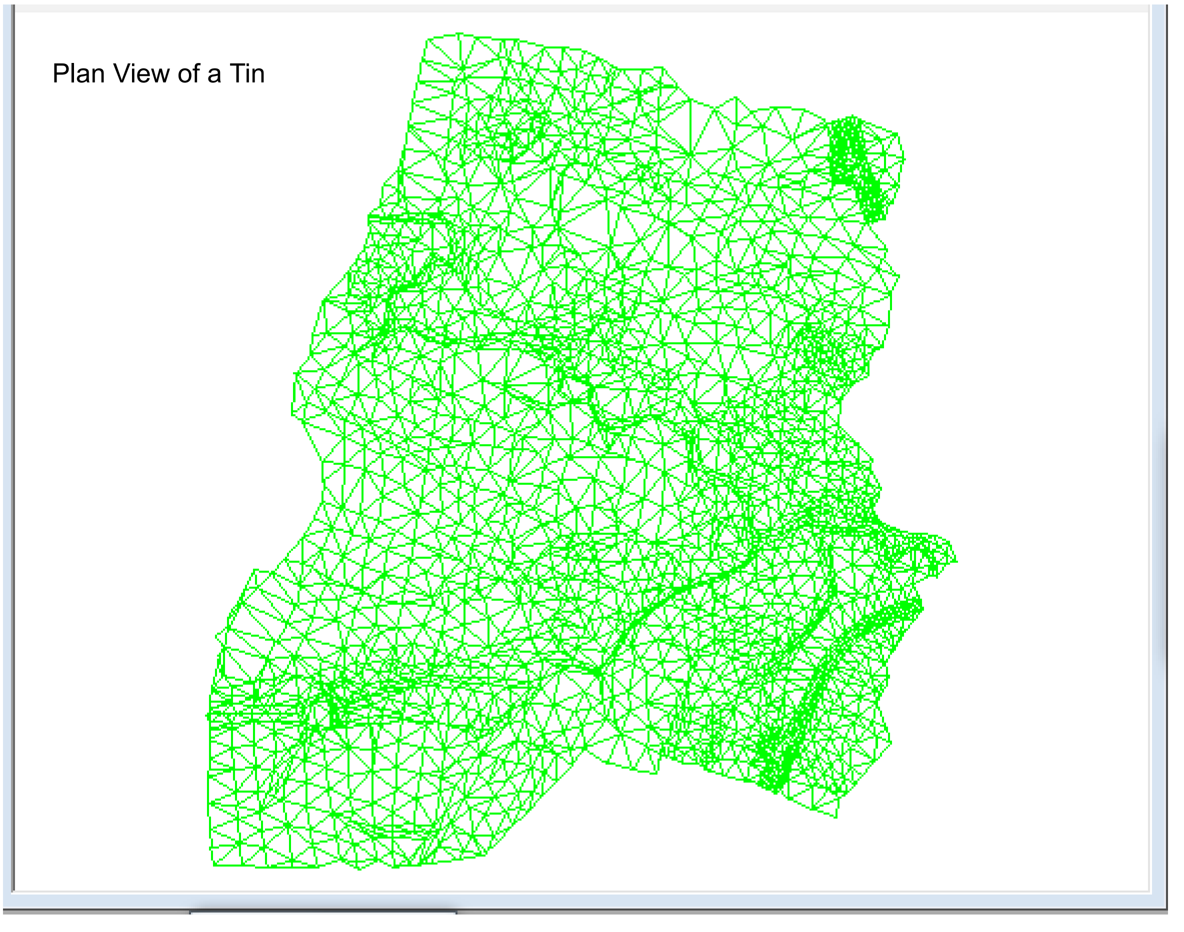
4.5.4 Tins

To form a continuous 2.5D surface representing the data in a model, a process called triangulation is used.

Note: a 2.5D surface has only one z-value for any (x,y) value in the surface.)

Triangulation creates in 2D, a web of connected, non-overlapping triangles, whose nodes are the (x,y) vertices from ***12d Model*** strings. Because the data points are normally irregularly spaced in plan, the triangulation is referred to as a **TIN** - a **T**riangulated **I**rregular **N**etwork.

If more than two vertices have the same (x,y) location, then only one of the z-values can be used for that triangle node.



For the interface strings and the old 2d, 3d, 4d strings, all non null points are included in the triangulation. For super strings, only vertices that are **tinable** are included in a triangulation.

To allow the triangulation to accurately represent features such as ridge lines and creek beds, the segments of breakline strings (line strings) are preserved as edges of triangles in the tin.

For a **super string,** only **segments** that are **tinable** and whose end vertices are also tinable, are included as break lines.

Each tin is given a unique user-defined name of up to two hundred alphanumeric characters and spaces.

There is a default colour for the triangles (faces) in a tin but individual faces can be also be given a different colour to the default colour.

Individual triangles in a tin can also be **nulled**. That is, turned off so they are effectively not there. However the nulled triangles are not deleted and are still stored in the tin and they can be un-nulled at any time.

A tin has a **surface area** which is the sum of the area of each of the triangles in the tin.

Unlike strings, tins can be in more than one model, or even no model at all. However, to be **displayed** in a view or used for **profiling** on a section view, tins need to be in at least one model. It is suggested that each tin be in its own model called "**tin tin_name"**. This makes it easy to know the model a tin is in, and also to see which models contain tins and obtain lists of all tins when using <ctrl> + <d> for name completion.

There is an option to list all the tins available in a ***12d Model*** session (the tin list) which also displays the models the tin is in as a walk-right. The tin list is also used in various pop-ups and walk-right menus.

Tins can be "added" and "removed" from models with options in the ***12d Model*** **Triangles** menu. If a tin is to displayed in a view, it must be "added" to a model that is being displayed in that view.

For safety, they are **not** deleted when any model containing them is cleaned or deleted.

Tins can only be **deleted** by using the **Triangles=>Delete** options.

WARNING

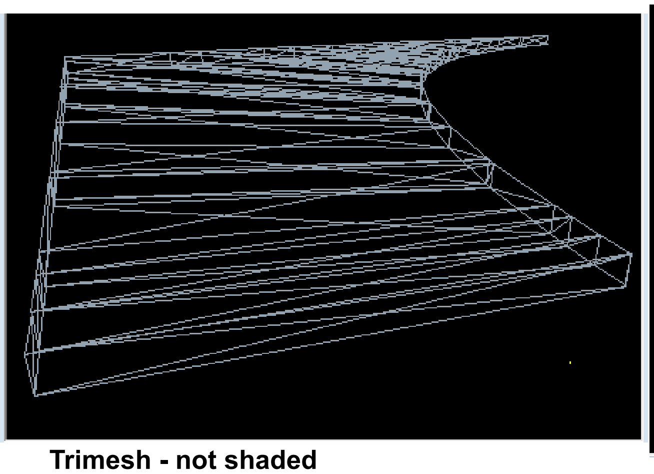
Tins are stored with **copies** of the points that were triangulated so if the original points are modified, the triangulation will not reflect this change until a **Recalc** of the triangulation is done.

Go to the next section 4.5.5 Trimeshes or back to 4.5 Data Types.

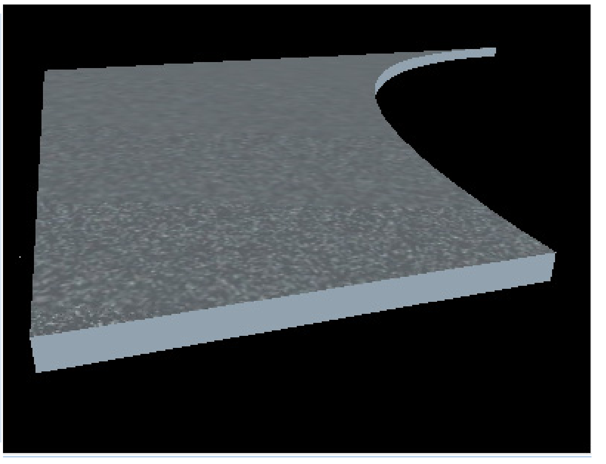
4.5.5 Trimeshes

To model a continuous 3D surface, trimeshes are used.

A trimesh is a web of connected 3D triangles. where none of the triangles intersect each other.



Trimesh - not shaded



Trimesh - shaded

A trimesh can have a name of up to two hundred alphanumeric characters and spaces. The trimesh name can be blank.

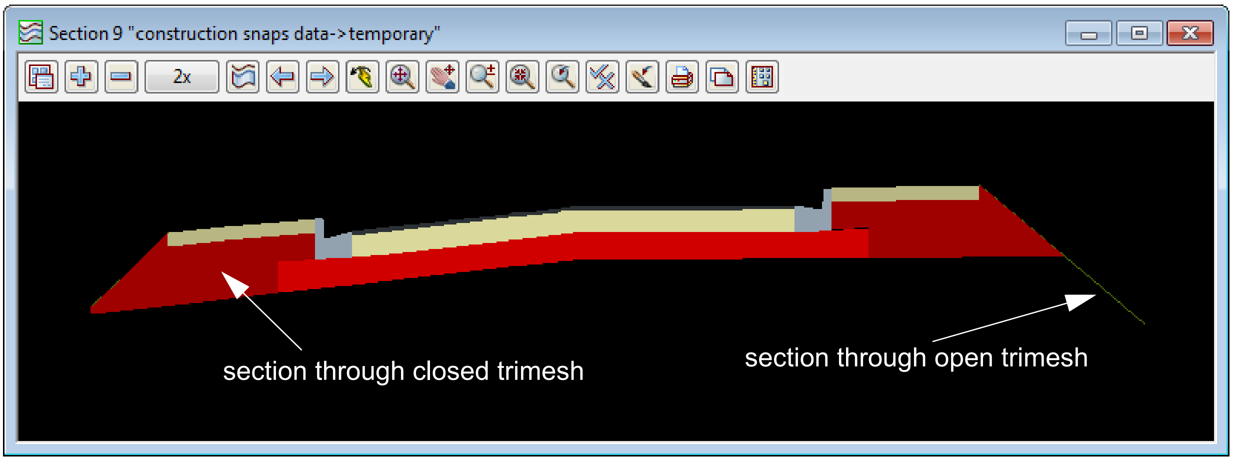
There is a default colour for the triangles (faces) in a trimesh but individual faces can be also be given a different colour to the default colour.

A trimesh has a **surface area** which is the sum of the area of each of the triangles in the trimesh.

A trimesh is **closed** if for each side of a triangle in the trimesh, there is another triangle in the trimesh butting up to it. A **closed** trimesh can have a **volume**.

If a trimesh is not **closed** then it is said to be **open**. An **open** trimesh **does not** have a **volume**.

When taking a section through a trimesh, the **outline** of the trimesh is shown and if the trimesh is **closed**, then the trimesh is **colour filled**.



A trimesh is a generalisation of a tin and unlike a tin, a trimesh can fold under itself.

There are numberous tools in ***12d Model*** for generating ***trimeshes*** for representing pavement layers, gutters, tunnels etc as 3D objects.

Go to the next section 4.5.6 Templates or back to 4.5 Data Types.

4.5.6 Templates

Templates are used as a quick and easy method for defining design details along a string for use in conceptual and detail designs, and visualizations. Templates are stored with each project but can be written out and read in using a readable file format.

Go to the next section 4.5.7 Projects or back to 4.5 Data Types.

4.5.7 Projects

In ***12d Model***, information is organised into **projects**.

When ***12d Model*** begins, the user specifies the name of the project to work on. A project name can be up to two hundred alphanumeric characters and spaces.

All the information created for that project is kept together in a special folder called the **project area**.

The project area contains all the models, templates and tins associated with the project. Copies of models and tins created in other projects can be added to another project from within ***12d Model***.

External data files can be read into the project (imported) using special input options. Similarly, output options are provided to write data out for use in other software packages (exported).

Go to the next section 4.5.8 Attributes or Meta Data or back to 4.5 Data Types.

4.5.8 Attributes or Meta Data

Extra data can be attached to the Project, Models, Strings, Tins and other Elements as **attributes**, including **user defined attributes**. Attributes are also called **meta data**.

Super strings can not only have attributes defined for the entire string (**string attributes**) but can also have attributes defined for each vertex (**vertex attributes**) and for each segment (**segment attributes**).

An attribute has a **name** and can store either an **integer** (32 bit), a **real** (64-bit double precision), a **text**, a **64-bit integer**, an **uid** (which is just a number), or it can be a special attribute called a **group** attribute that can store other attributes.

Hence attributes in ***12d Model*** can be grouped into a ***hierarchy*** or ***tree.***

That is, there are attributes at the ***Top*** level (the first level), and amongst the top level attributes can be group attributes that contain zero or more other attributes. Attributes in a first level group attribute are said to be in the **second** level.

Similarly an attribute in the second level can be a group attribute and hence contain other attributes which are said to be in the **third** level.

So attributes form a tree structure much like a folder structure for files.

Within a group, all the attributes must have unique names.

See Attribute Pathname
Accessing Attribute Values
Attribute Data Panel
Comparing Attribute Data
Accessing Attribute Values

Attribute Pathname

The **pathname** of an attribute is the list of attribute names that you must go through, starting at the top level, until you get to the attribute. That is, the path name is the path through the attribute structure that you must go through until you reach the attribute.

The attribute names that you pass through at each level are unique so the pathname to an attribute is unique is a unique and is written by listing the attribute names you pass through at each level separating each name by a **/**:

first_level_att_name / second_level_att_name/ **...**/attribute_name

For example, if the first level contains the attributes

sun, moon and earth   top or first level

and **earth** is a group attribute containing the attributes

asia, africa, lee, america europe and australia   second level

and **australia** is a group attribute containing the attributes

brisbane, melbourne, perth and sydney   third level

and **sydney** is a group attribute containing

fred, joe, mary, sam and lee   fourth level

then the path name to **lee** in the group **sydney** is

earth/australia/sydney/lee

Note that there is also a **lee** in the second level but there is only one **lee** at each level so there is no problem. The pathname to the first **lee** is:

earth/lee

In***12d Model*** options where attribute names are required in an input field, the full ***pathname*** of the attribute must be given to uniquely identify the attribute.

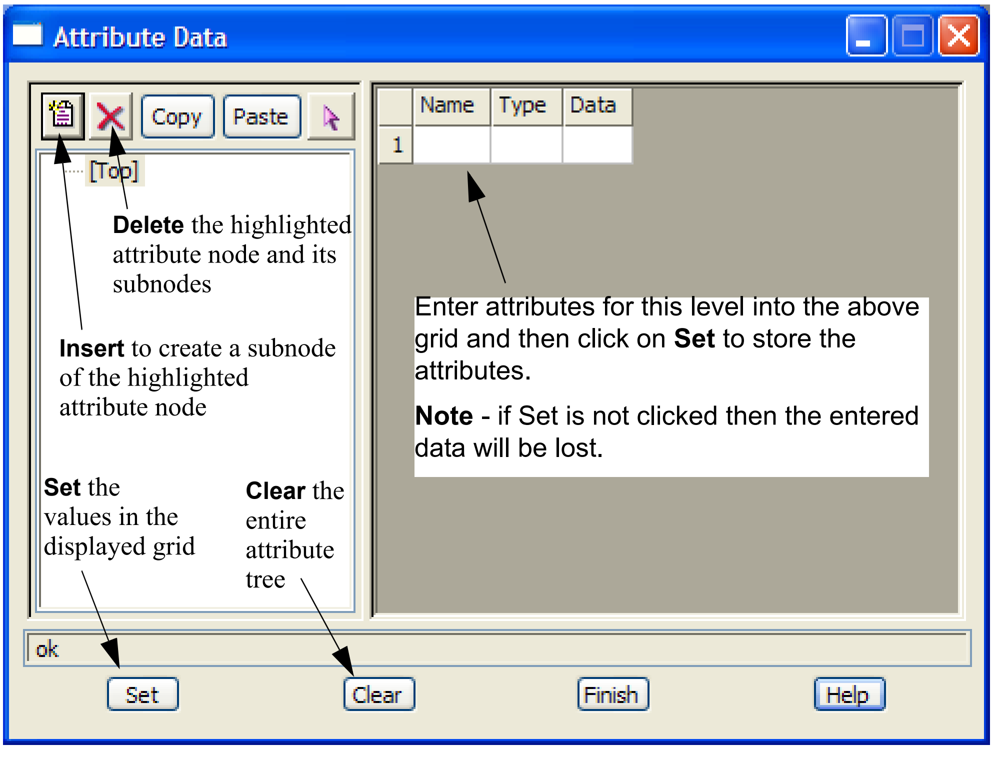
Attribute Data Panel

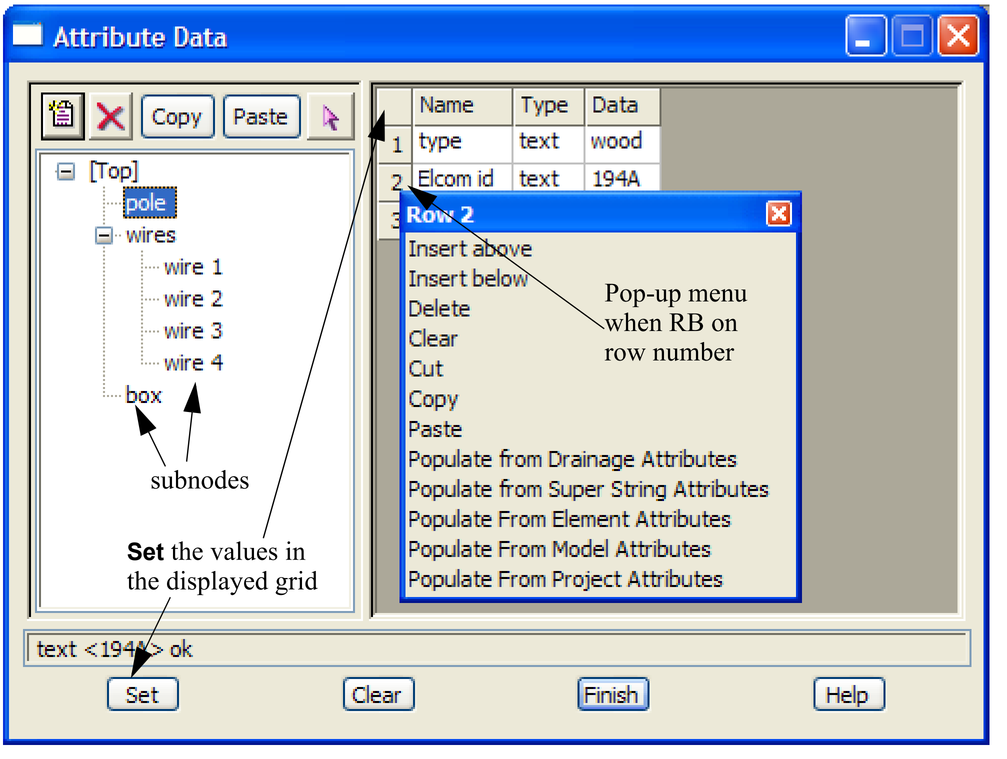
In many options when attributes are to be defined, the **Attribute Data** panel is displayed.

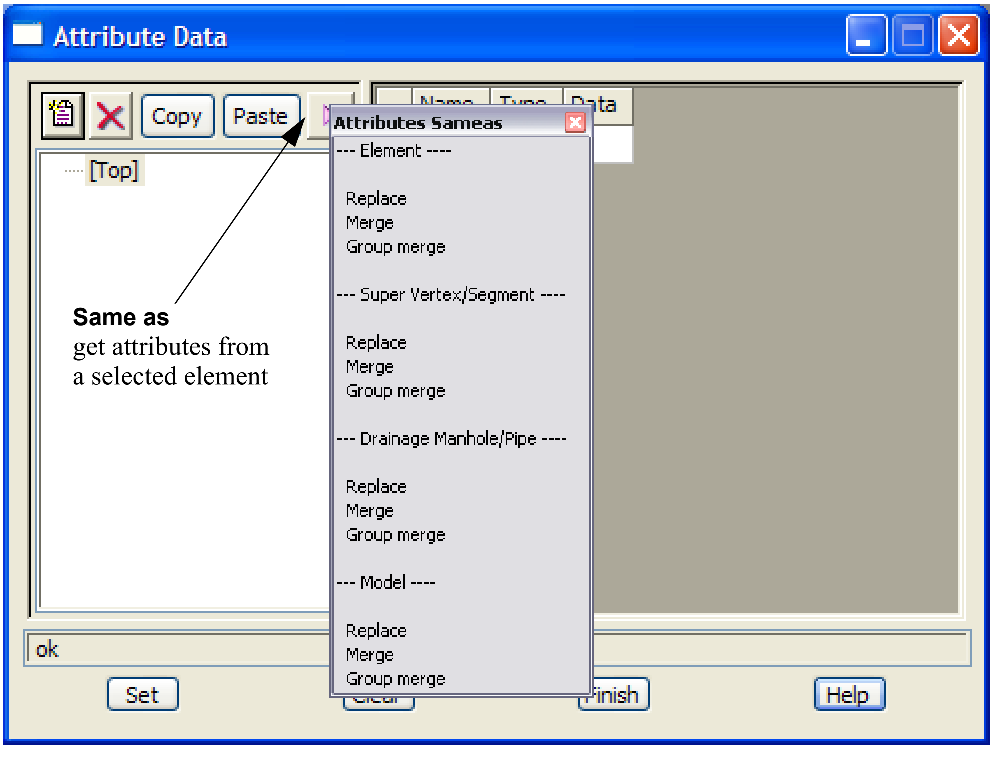
The **Attribute Data** panel can create and edit attributes.

When no attributes exist for an object, the **Attribute Data** panel just shows the **Top** of the attribute tree. New attributes for the top level can be added in the right hand side grid.

Sub nodes (the lower levels of the attribute tree) of an attribute node are created by simply highlighting the attribute node and then clicking on the **Insert** icon. Attributes for the subnode are then entered into the right hand grid and the **Set** button clicked to save the subnode and its attributes.







The buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Insert	button		

create a new attribute node (subnode) beneath the highlighted attribute node

Delete button

delete the highlighted attribute node

Copy button

copy the highlighted attribute node into the attribute buffer

Paste button

paste the attribute buffer to the highlighted attribute node

Same As button

paste the attribute buffer to the highlighted attribute node

Set button

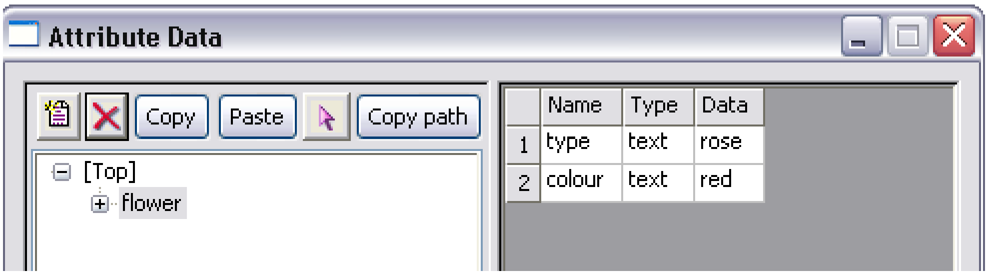
set the values given in the current grid in the attribute tree

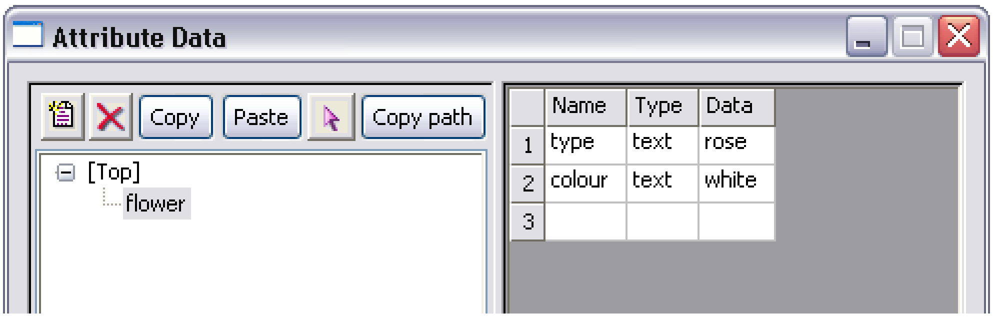
Clear button

clears out the entire attribute tree

Comparing Attribute Data

Two Attribute Datas are the same if each Attribute Data has exactly the same groups, and each group has exactly the same attributes in them, and each identical attribute has ***exactly the same value***.





For example, the two attribute data shown above have exactly the same attribute group "flower" and the group "flower" has exactly the same attributes "type" and "colour", the attribute data are not the same because the value of "colour" is "red" for one attribute data, and "white" for the other attribute data.

Accessing Attribute Values

In some options, in particular in the Map File, the actual ***value*** of a given attribute (the column called Data in the **Attribute Data** panel***)***, is specified by putting a **$** in front of the attribute path name.

That is

$earth/australia/sydney/lee

refers to the actual value of the attribute given by the path name *earth/australia/sydney/lee.*

For example, see 8.8.3 Map File Substitution by Attributes.

Please continue to the next section 4.6 Text Definitions.

4.6 Text Definitions

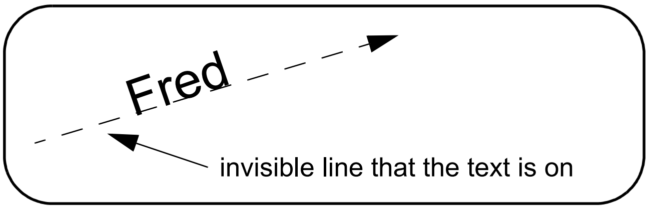
Text can occur as a text string, or on vertices and segments of a super string.

The Text itself consists of one or more lines of text but the display position of the text is not as straight forward as simply placing a vertex.

The ***text display position*** of is defined by

(a) an initial start position - usually a vertex or a segment

(b) the ***direction*** for the text - the invisible line that the text runs along



(c) the ***text justification point*** and the ***offset***, ***raise*** and ***angle*** defined with respect to the justification point. Note that the definition of the justification point and the offset, raise and angle depend on whether the text is vertex text or segment text.

(d) the justification (top left etc) for the text about the justification point

Apart from text display positioning, text also has a

(e) textstyle

(f) text units for text height, offset and raise value

(g) height (size)

(h) width factor (w or x factor)

(i) slant

(j) colour

(k) weight

(l) underline, strikeout and italic

(m) border and border type

(n) whiteout

There are also special ways of defining characters which are not available on the keyboard. For example the degree character and the squared and cube characters.

Instead of having separate variables on each panel for all of these text definition values, a ***Textstyle Data*** has been introduced to hold all the Text variables.

Apart from saving a massive amount of room on panels, another major benefit of the ***Textstyle Data*** is that in the future, extra variables can be added to the ***Textstyle Data*** structure and the variables are then available everywhere a ***Textstyle Data*** is used.

The **Textstyle Data** or **Textstyle Info** panels are usually brought up to enter Text setup data. For information on those panels, see 4.6.12 Textstyle Data and Textstyle Info.

For definitions on textstyle data values, see

4.6.1 Justification Point and Offset, Raise, Angle
4.6.2 Text Justification
4.6.3 Textstyle
4.6.4 Text Units

4.6.1 Justification Point and Offset, Raise, Angle

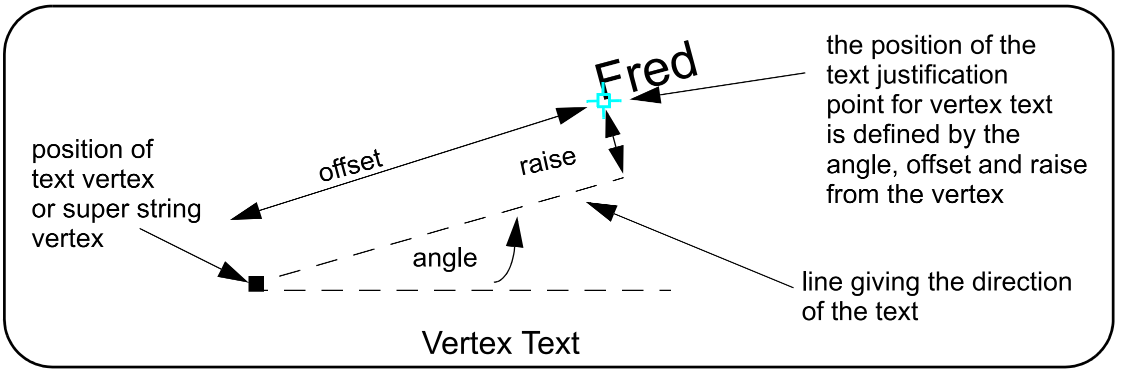
Vertex Text

Vertex text refers to the text of a *text string* or text at a super string vertex (vertices are displayed by toggling on ***Vertices*** for a plan view).

For vertex text, the text ***justification point*** and the ***direction of the text*** are defined by:

(a) the *direction of the text* is given as a *counter clockwise* ***angle*** *of rotation* (measured from the x-axis) about the vert*ex*

(b) the *justification point* is given as an ***offset*** from the vertex *along the line through the vertex with the direction of the text*, and a perpendicular distance (called the ***raise***) from that offset point to the justification point.



The vertex and justification point only coincide if the offset and raise values are both zero.

All text on a 4d string must have the same height, colour, angle, offset and raise.

What parts of the text on a super string vertex can be independently modified depends on the settings for the super string.

Note that these definitions are relative to the vertex and if the vertex moves, then the text moves with it.

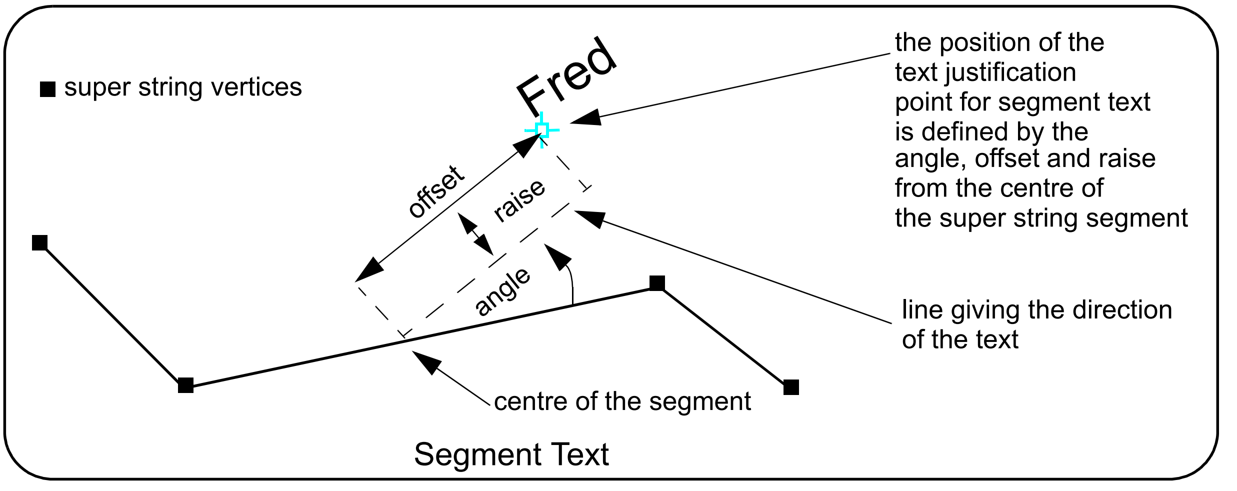
Segment Text

Segment text is a special type of text that can only be placed on the *segment* of a super string. Unlike text at a vertex, the segment for segment text has a direction and mostly the text is required to be parallel, or related to the segment direction.

For segment text, the text ***justification point*** and the ***direction of the text*** are defined by:

(a) the *direction of the text* is given as a *counter clockwise* ***angle*** *of rotation,* measured from the segment, about the centre of the segment

(b) the *justification point* is given as an ***offset*** from the centre of the segment *along the line through the centre of the segment with the direction of the text*, and a perpendicular distance (called the ***raise***) from that offset point to the justification point.



The direction of the text is parallel to the segment if the angle is zero.

Note that these definitions are relative to the segment and if the vertex segment in any way, then the text also moves with it.

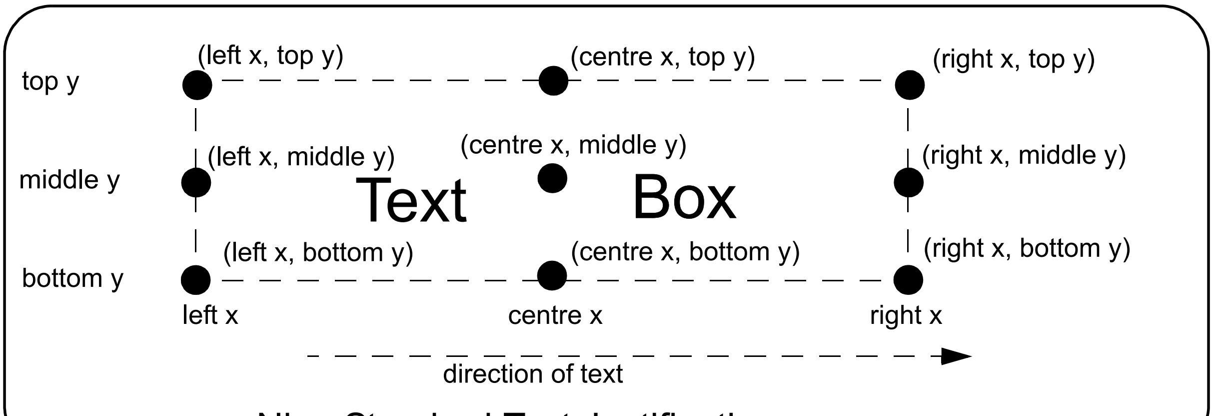
Please continue to the next section 4.6.2 Text Justification or return to 4.6 Text Definitions.

4.6.2 Text Justification

There are nine ways that any text can be positioned relative to the ***justification point***, and sixteen ways for numbers with a decimal point.

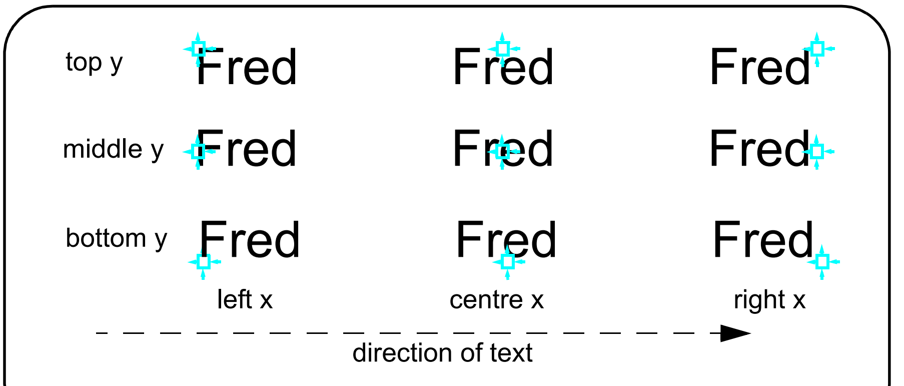
For standard text, the justification is defined in terms of the rectangular box surrounding the text, and this box depends on the height of the text, the text font, and the actual characters in the text (especially for proportional fonts).

In term of the text box, the nine justifications are formed by dividing the bottom and sides of the box in half and calling bottom points left x, centre x and right x, and the side points bottom y, middle y and top y. The nine justifications are



Nine Standard Text Justifications

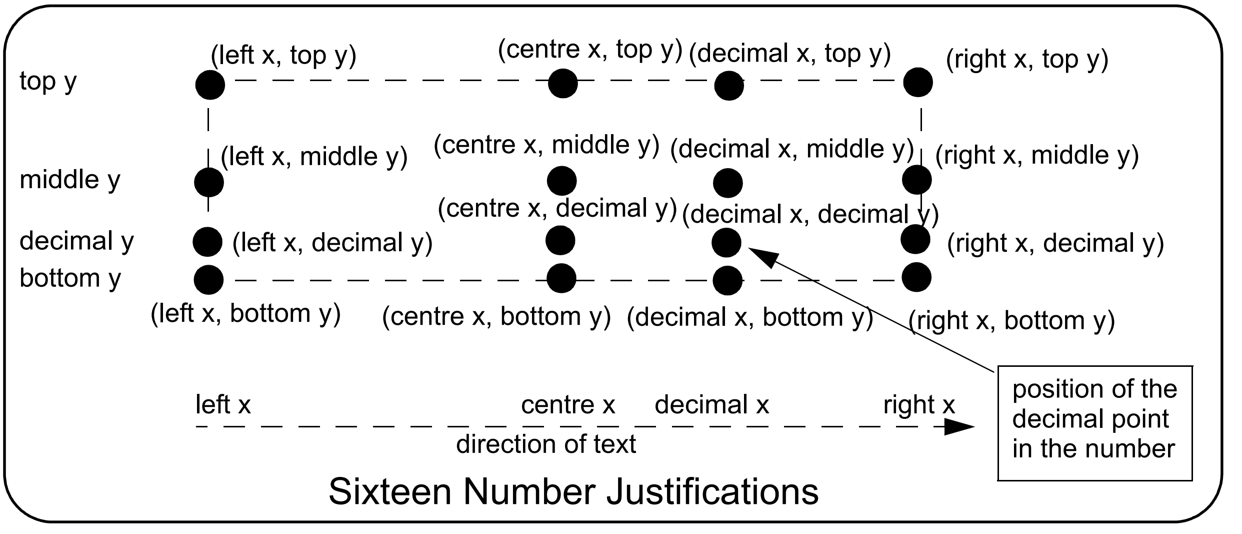
As an example, for the text **Fred**



Text Justifications

For numbers with a decimal point, the position of the decimal point gives an addition point on the bottom called decimal x and on the side called decimal y.

So there are sixteen possible justification for numbers.

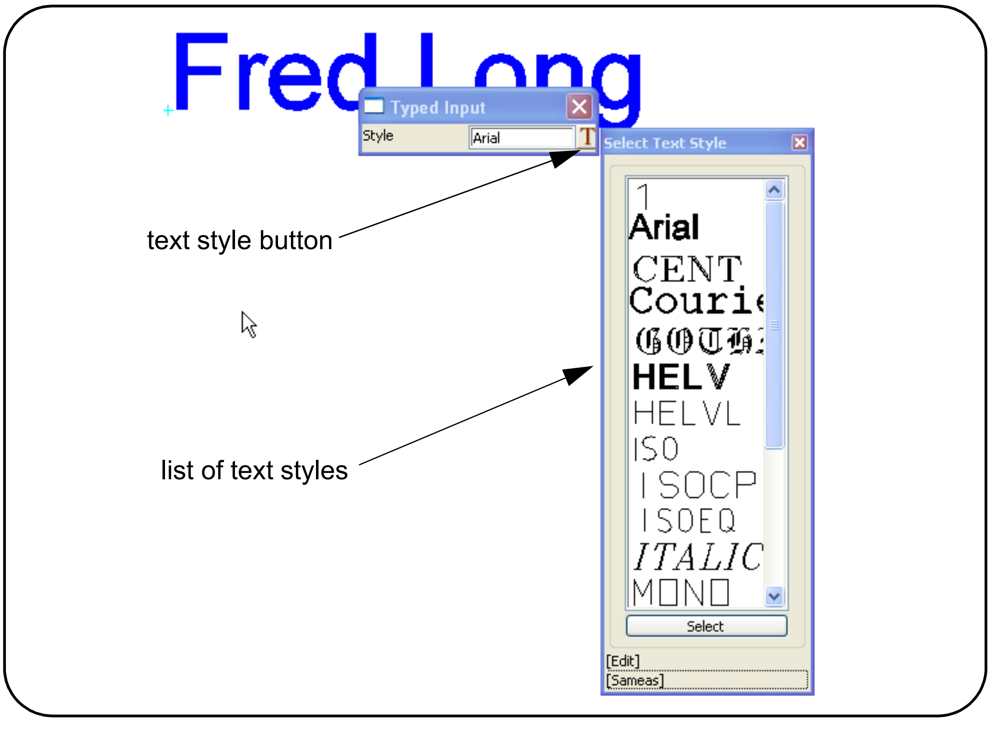


Please continue to the next section 4.6.3 Textstyle or return to 4.6 Text Definitions.

4.6.3 Textstyle

The available text styles for a ***12d Model*** project are defined by textstyles definition file (see the section 40.3 Textstyles and Fonts in the Appendix 40 Linestyles, Symbols, Textstyles &Patterns).

The text style is selected from the pop-up list displayed when clicking on the **Text style** button.



Please continue to the next section 4.6.4 Text Units or return to 4.6 Text Definitions.

4.6.4 Text Units

Text occurs in 12d Model in three ways -

- user defined **text strings**
- automatic text such as grid values, x-section and long-section plot annotation.
- text within linestyles.

The most difficult thing about text is that because of the different uses of text, there needs to be more than one systems of **units** to define text heights.

The height of text for a given textstyle is defined to be the height of a capital A. However, in 12d Model, there are three methods of defining the units for measuring this height.

- world units - the units used for data
- screen units - pixels (the screen is 1000 pixels wide)
- plot paper units - millimetres.

World Units

World units are the units of user data. For most users, the base unit for user data is metres. However 12d Model is a dimensionless system and the base unit is totally dependent on the user.

The height of world text when displayed in a view depends upon on the text height and the scale of the view.

When plotted, the height that world text appears on a plot sheet is the same as for any data defined in world units - the height depends on the **scale** used for the plot.

Text heights that are only given in world units have **(w)** after them.

For some text, the choice of units is either world or pixels. The text parameters then have a **(u)** after them.

Screen Units - pixels

When screen units (pixels) are used, the text is a fixed height on the screen. If the user zooms in on text given in pixels, the text remains the same height.

To have a height on a plot, screen unit text needs a height defined in millimetres.

For some screen text, both a pixel and a millimetre height is supplied when the text is defined.

For text with only a pixel height, there is a plotting multiplication parameter called **pixels-to-millimetres** which is used to convert pixel heights to plot paper heights. The value of pixels-to-millimetres is set using the **plots=>pixels to mm** option and is stored for the project.

Text heights that are only given in pixels have a **(pix)** or **(p)** after them.

For some text, the choice of units is either world or pixels. These text parameters then have a **(u)** after them.

Plot Paper Units - millimetres

Text defined in plot paper units (millimetres) has a well defined height on a plot sheet. When a view scale is set for a plan view, then paper text will draw at the correct size for that scale.

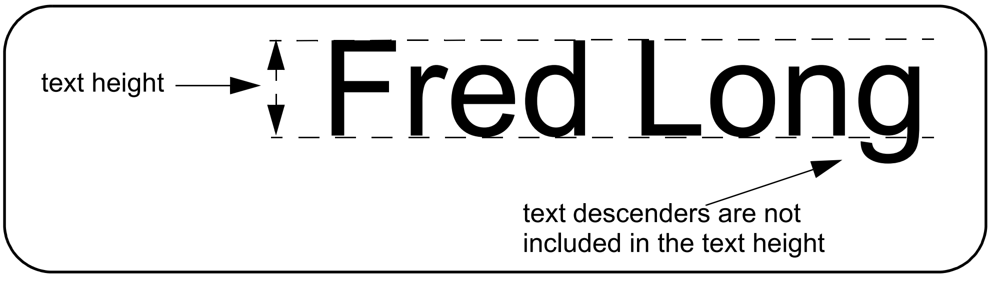
Text heights that are only given in millimetres have **(mm)** after them.

Please continue to the next section 4.6.5 Text Height or return to 4.6 Text Definitions.

4.6.5 Text Height

The height of text is defined to be the height measured from the line that the text sits on to the top of the upper case characters such as F.

Text descenders are not included in the definition of height.



Please continue to the next section 4.6.6 Text Width Factor or X Factor or return to 4.6 Text Definitions.

4.6.6 Text Width Factor or X Factor

In a text font, each character has a defined width which may depend on the character.

For a ***non-proportional font*** such as `Courier New`, all the characters in the font have the same width.

For a ***proportional font*** such as Arial, the width varies for each character.

The character width is actually defined as part of the font and is not available to 12d text but the character widths are multiplied by a *width factor (x factor)* which is set for each text string.

Proportion Font - Arial

x factor = 1 Fred Long

x factor = 2 Fred Long

Non-Proportion Font - `Courier New`

x factor = 1 `Fred Long`

x factor = 2 `Fred Long`

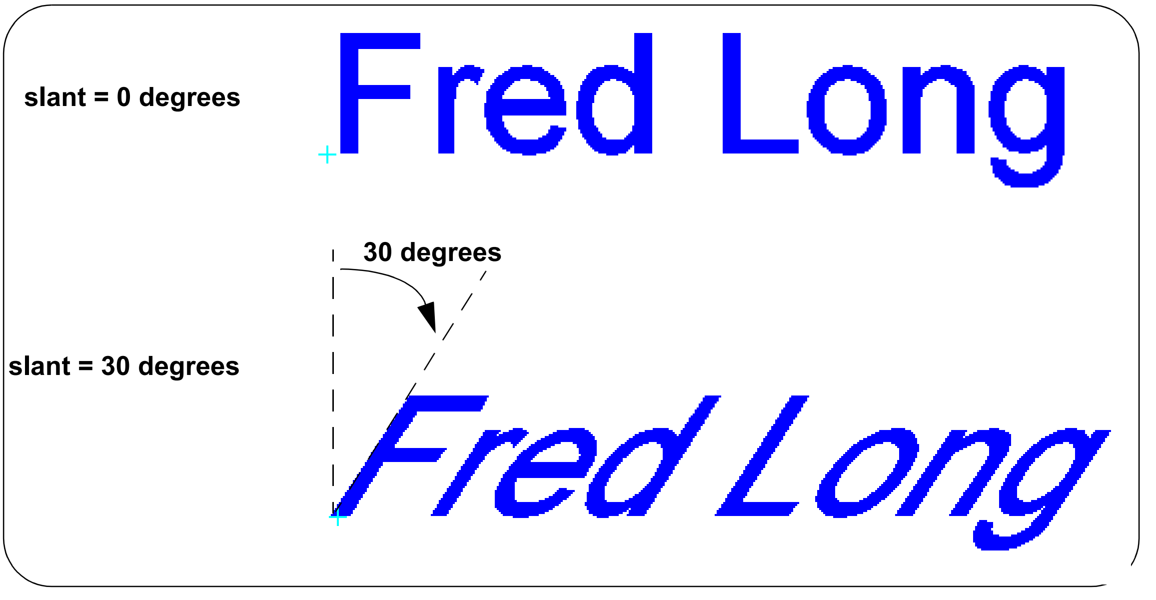
Please continue to the next section 4.6.7 Text Slant or return to 4.6 Text Definitions.

4.6.7 Text Slant

A ***slant*** angle of between 0 and 45 degrees can be defined for a text string.

For non-zero slant angles, the text characters are slanted by the given angle measured in a clockwise direction from the vertical (a bearing).

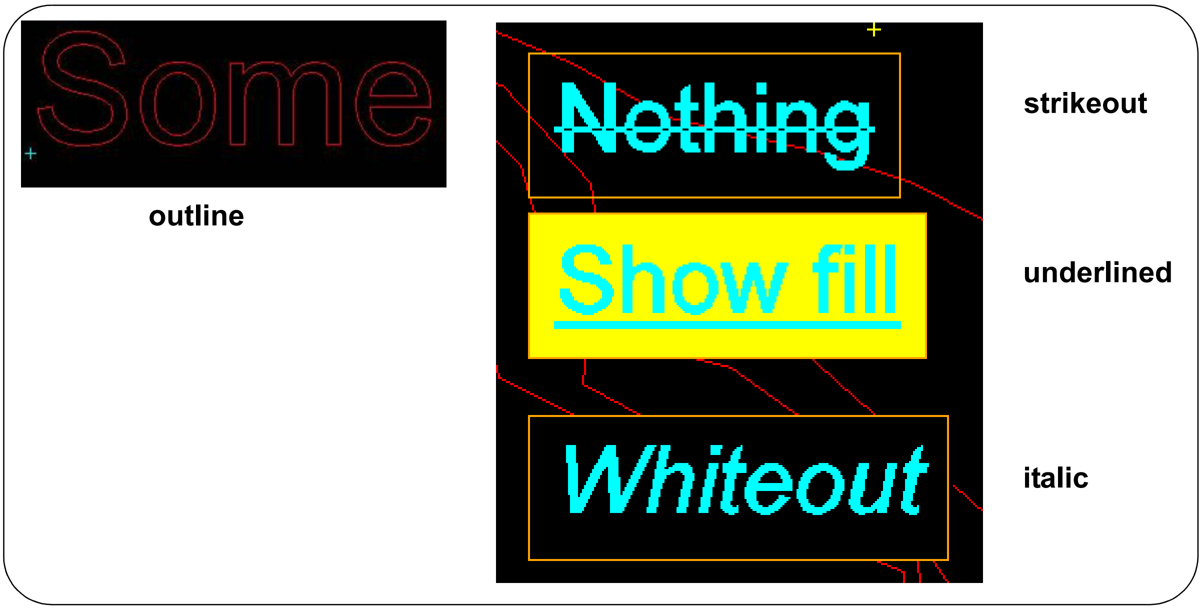
Currently the text *slant* angle can only be set by the ***Slant*** field in the **Textstyle Data** panel field. The value is in degrees in the format ddd.mmssfff.



Please continue to the next section 4.6.8 Weight, Underline, Strikeout, Italic, Outline or return to 4.6 Text Definitions.

4.6.8 Weight, Underline, Strikeout, Italic, Outline

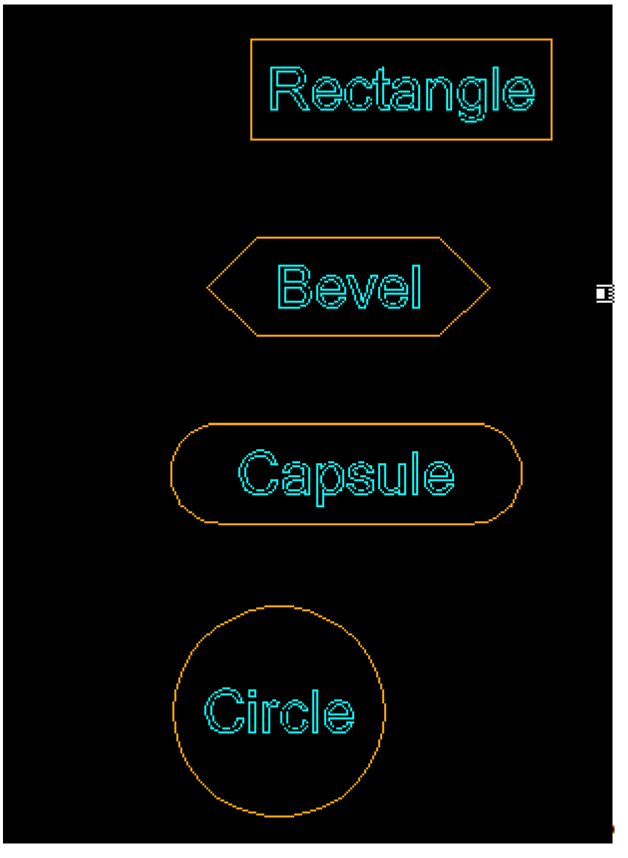
For ***True Type Fonts*** only:



Please continue to the next section 4.6.9 Border and Border Style or return to 4.6 Text Definitions.

4.6.9 Border and Border Style

Text can have a border around the text with one of the border styles Rectangle, Bevel, Capsule and Circle.



Please continue to the next section 4.6.10 Whiteout or return to 4.6 Text Definitions.

4.6.10 Whiteout

The border around the text an be colour filled before the text is drawn. This is called Whiteout.

In the **Select Colour** pop up there is one special colour on the bottom of the menu called **[View colour]**.

If **[View colour]** is selected, then the fill colour automatically changes to whatever the background colour of the view that the text is on.

Although the whiteout colour fills the border around the text, the border does not have to be drawn.

Please continue to the next section 4.6.11 Special Text Characters or return to 4.6 Text Definitions.

4.6.11 Special Text Characters

There are special text characters that do not appear on the standard keyboard but can still be entered into text strings, vertex text and segment text with textstyles using ***true type fonts***.

The special characters are entered by holding down the Alt key and typing in certain numbers on the *number pad* whilst the Alt key is still being held down). The character will appear when the Alt key is released.

Some commonly used special characters and their Alt codes are

Squared character	²	Alt 0178
Cubed character	³	Alt 0179
Middle dot character	·	Alt 0183
Large diameter character	Ø	Alt 0216
Small diameter character	ø	Alt 0248
Degree character	°	Alt 0176
Copyright character	©	Alt 0169
Registered character	®	Alt 0174
British Pound character	£	Alt 0163
Japanese Yen character	¥	Alt 0165

The Alt values come from the character map for Windows. This can be quickly viewed by

run charmap.

NOTE - these values may depend on the true typed font used in the textstyle.

Please continue to the next section 4.6.12 Textstyle Data and Textstyle Info or return to 4.6 Text Definitions.

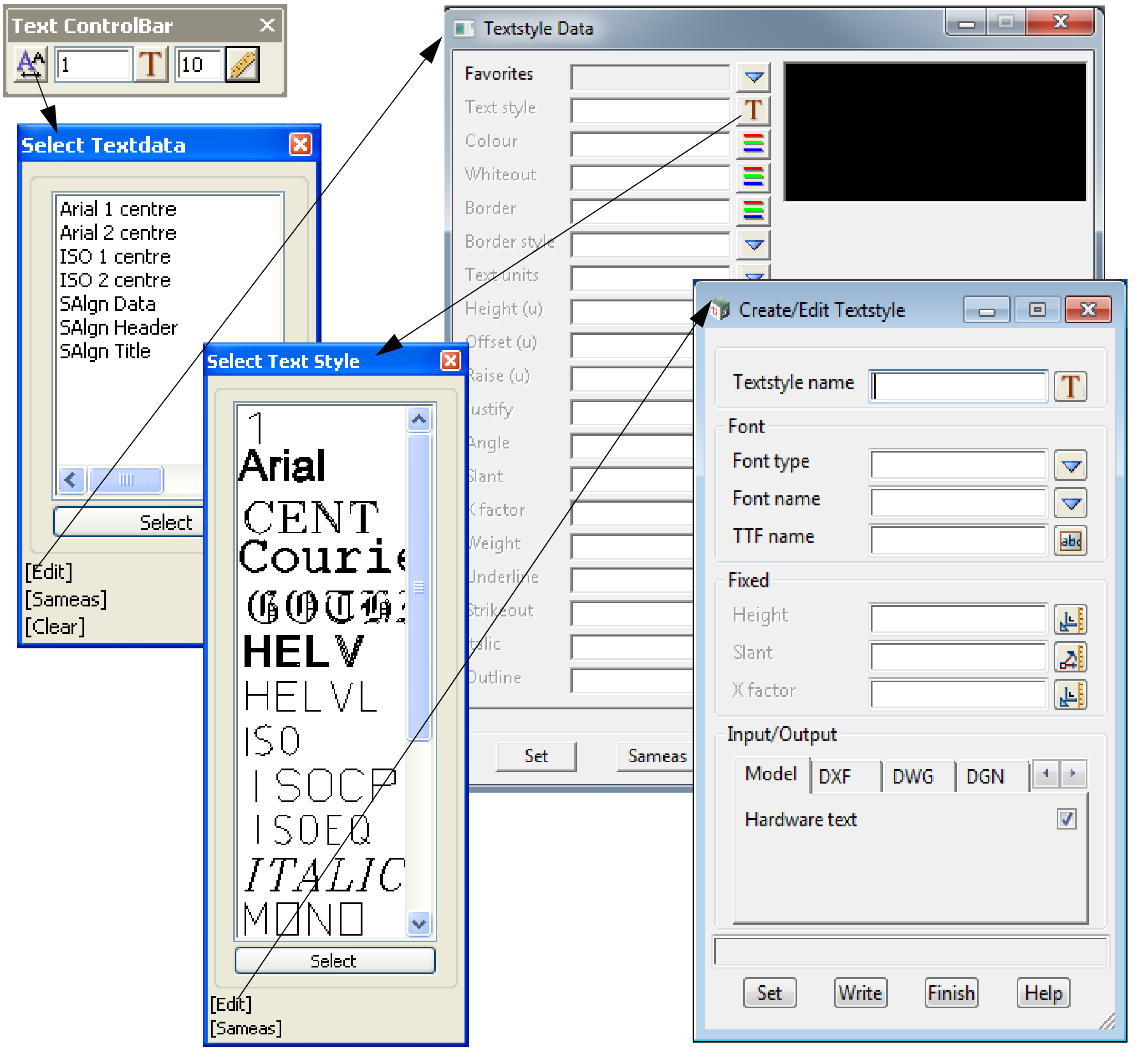
4.6.12 Textstyle Data and Textstyle Info

Text_info

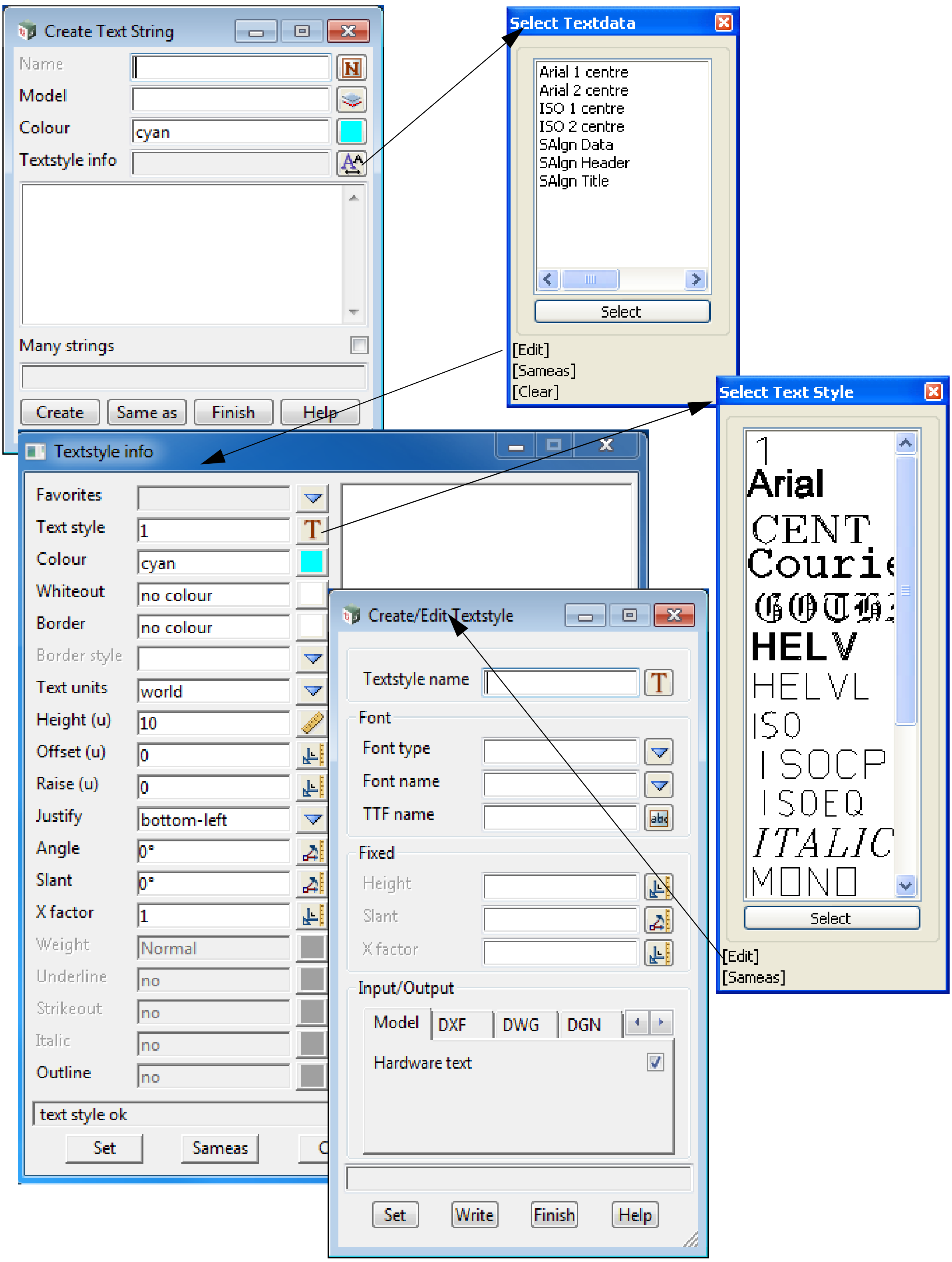
There is a **textstyle info** panel field for defining the text parameters.
Clicking on the textstyle info icon on the Text Controlbar or on a panel, brings up the **Select Textdata** pop-up menu which lists the textstyle favourites. By clicking on **[Edit]**, the **Textstyle Info** panel is brought up and all the definitions for the text style can be modified and written to the **textstyle info** panel field by clicking on **Set**.

Textstyle favourites are defined in the Browse option **Projects=>Browse=>Textstyle data favourites** and documented in the section 7.9.7 Textstyle Data Favourites.

Note that there is a **Textstyle Data** panel that is identical to the **Textstyle Info** panel except that it doesn't have the **Colour** field. The Textstyle Data panel is brought up by clicking on the A on the Text Controlbar.



For information about the **Create/Edit Textstyle** panel, see 7.9.6 Textstyles.



For information about the **Create/Edit Textstyle** panel, see 7.9.6 Textstyles.

For information on the definitions of the fields in the **Textstyle Data** and **Textstyle Info** panels, go to the section 4.6 Text Definitions

Please continue to the next section 4.7 Symbol Definitions.

4.7 Symbol Definitions

Symbols are often on a one vertex super string but they can be placed on any vertex of a super string.

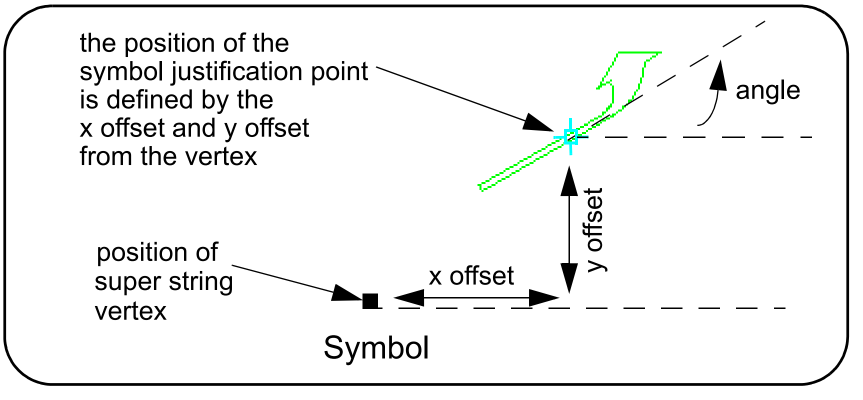
The displayed symbol is defined by

(a) an initial start position

(b) the actual symbol

(c) the **x offset** and **y offset** of the ***symbol justification point*** from coordinates of the vertex. The vertex and justification point only coincide if the x offset and y offset values are both zero.

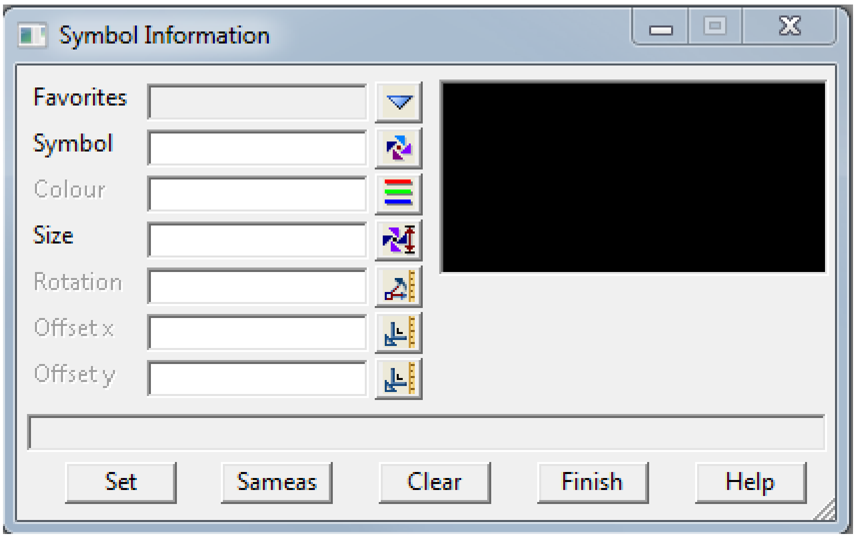
(d) the rotation *angle of the symbol* is given as a *counter clockwise **angle** of rotation* (measured from the positive x-axis) about the symbol justification point.



Note that these definitions are relative to the vertex and if the vertex moves, then the symbol moves with it.

What parts of the symbols on the vertices of the same super string can be independently modified depends on the settings for the super string. For example, all symbol angles may have to be the same.

The **Symbol Information** panel is usually brought up to enter symbol information (see 4.7.1 Symbol Data).



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Favourites

not yet implemented.

Symbol	**Symbol** box		available symbols

*the name of the **symbol**. This field can not be blank.*

Colour — **select colour** menu

the colour of the symbol

Size — **measures** menu

the size of the symbol. This field can not be blank.

Rotation — **measures** menu

the angle of rotation of the symbol. The angle is measured in a counterclockwise direction from the positive x-axis. The units for angle are degrees minutes and seconds and it is entered in 4.17.1 HP Notation.
*If **blank** then the rotation is 0.*

Offset x/y — **measures** menu

the x/y-distance to offset the symbol from each vertex of the string.
*If **blank** the value is 0.*

Set — button

set the values to be those in the panel.

Sameas — button

anther string can be selected and the information about it automatically put in the panel fields.

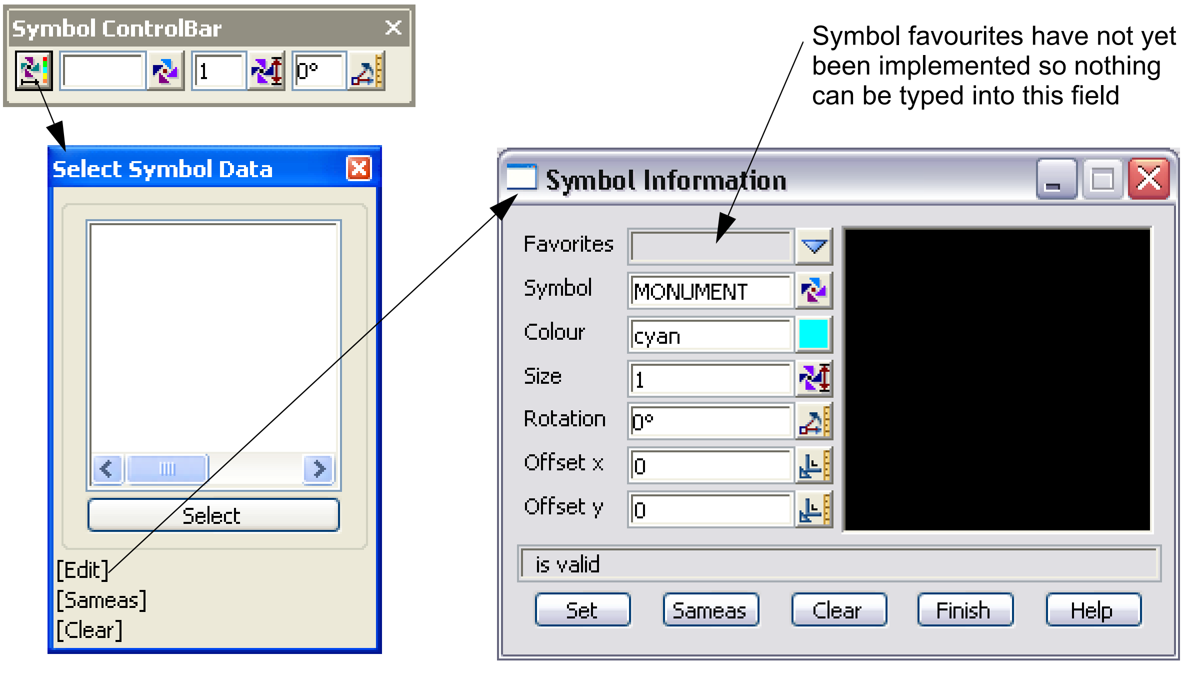
Clear — button

clear all the values in the panel fields.

4.7.1 Symbol Data

There is a **Symbol Information** panel for defining symbol parameters.

Clicking on the **Symbol info** icon on the **Symbol Controlbar** or on a panel, brings up the **Select Symbol Data** pop-up menu. By clicking on **[Edit]**, the **Symbol Information** panel is brought up and all the definitions for the symbol can be modified and written to the **Symbol Info** panel field by clicking on **Set**.



For information on the **Symbol Information** panel and the definitions for *symbols.* go to 4.7 Symbol Definitions

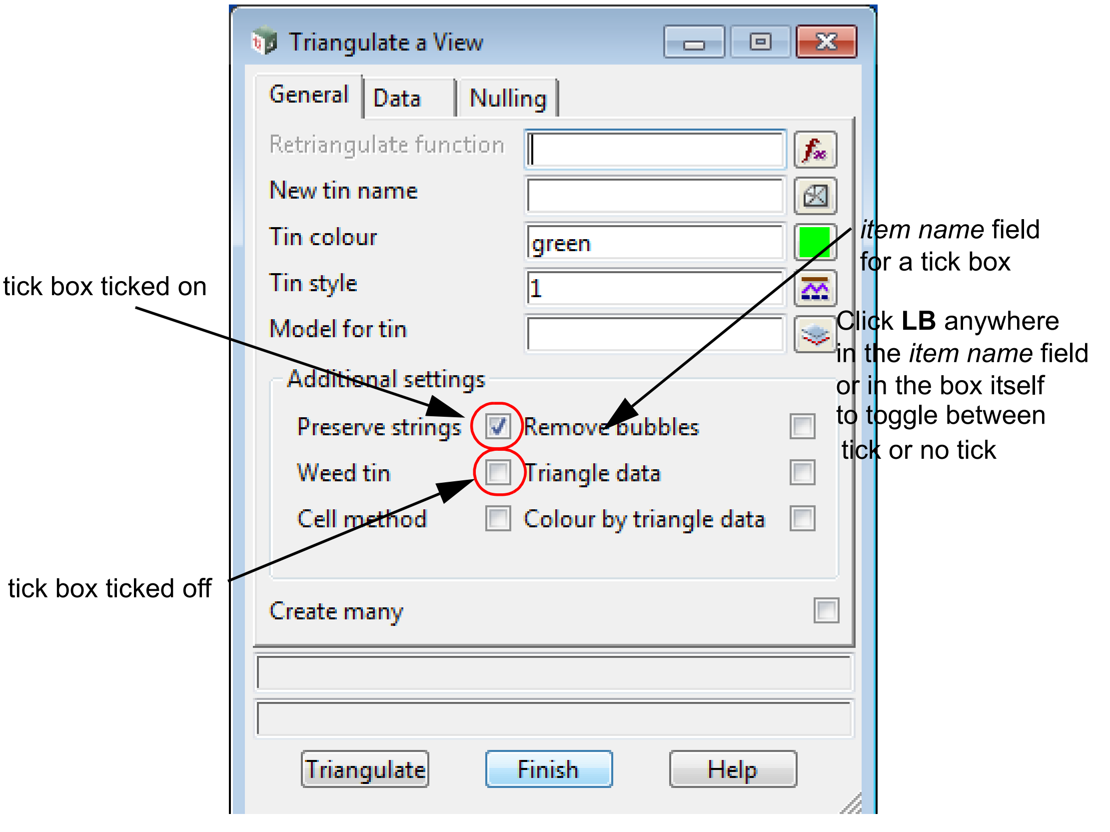
Please continue to the next section 4.8 Tick Box.

4.8 Tick Box

Many items require a mode to be either on or off.

In ***12d Model***, there is a special box with a **tick** to indicate that the setting is **on** and **nothing** to indicate that the setting is **off**.

In a **tick** box, the tick is changed to the nothing state by clicking LB in the box surrounding the tick or clicking LB on the **item name** field for the tick box.



Please continue to the next section 4.9 Picking Strings.

4.9 Picking Strings

In many ***12d Model*** options, the user is required to “pick” the string to be used in the option, or to get information about a string.

After any option requiring a pick is selected, a message regarding the function of the mouse buttons is written to the screen message area.

<option> [picks][fast][menu]

There are three picking method available in ***12d Model***:

(a) fast pick where the pick and the accept occur as one operation

(b) fast accept where the if there is only one item satisfying the snap conditions then it is automatically accepted but if there is more than one, the tentative pick mechanism is used

and

(c) tentative picks with a separate pick and accept mechanism

Fast pick will be documented first, followed by ***fast accept*** and ***tentative pick*** (pick and accept).

See 4.9.1 Fast Pick
4.9.2 Fast Accept
4.9.3 Tentative Pick

Go to the next section 4.9.1 Fast Pick or back to 4.9 Picking Strings.

4.9.1 Fast Pick

To **fast pick** a string, simply move the cursor near the string and **click MB** or type <Enter>. The nearest string to the cursor satisfying the snap conditions is selected.

Fast pick is used when the **F** snap is on.

Various snap modes can be set for the fast pick. The available snaps are point (vertex), line, grid, tin and cursor. One or more snaps can be set simultaneously. The snaps are described in more detail in the section 4.13 Snaps.

Go to the next section 4.9.2 Fast Accept or back to 4.9 Picking Strings.

4.9.2 Fast Accept

To **fast accept** a string, simply move the cursor near the string and **click LB** or type <Enter>. If there is only one string satisfying the snap conditions, then it is automatically accepted.

If there is more than one string satisfying the snap conditions, the nearest string to the cursor satisfying the snap conditions highlighted and an information menu containing information about the selected string is displayed.

If the correct string has been tentatively picked, **click MB** or **type <enter>** accepts the string. If the incorrect string is tentatively picked, click LB and the next closest string satisfying the snap conditions will be highlighted. See the section 4.9.3 Tentative Pick for more information on tentative picking.

Fast accept is used when the **A** snap is on.

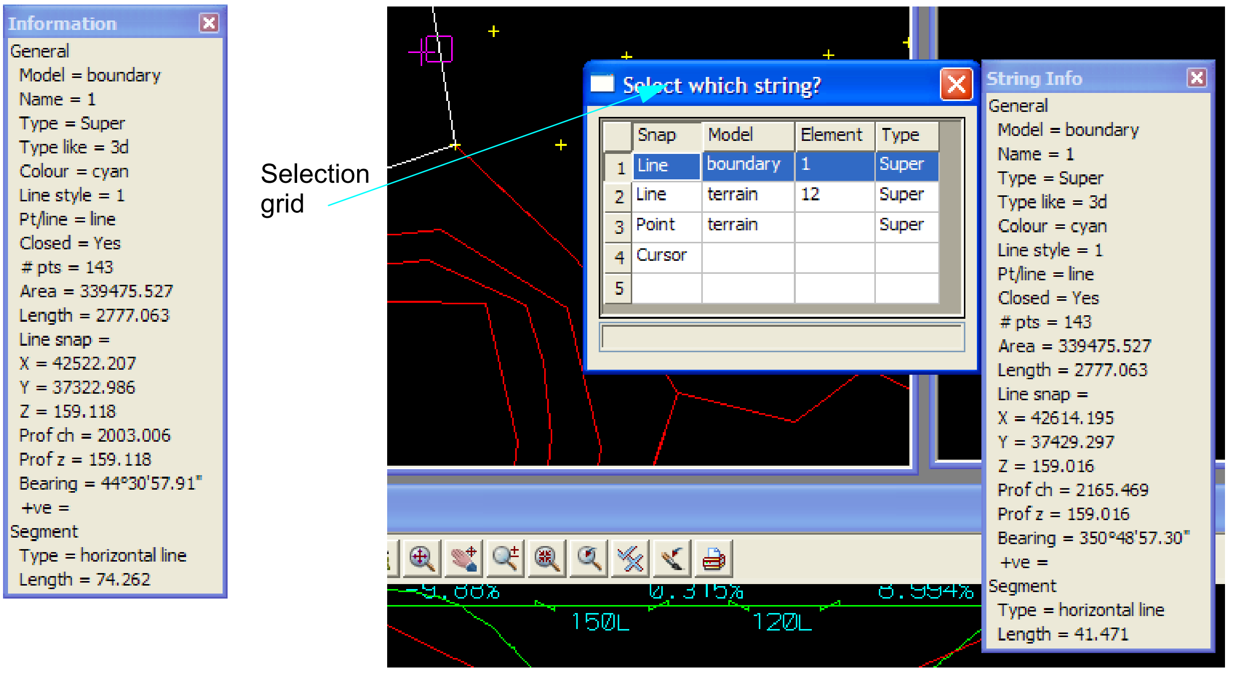
Various snap modes can be set for the fast pick. The available snaps are point (vertex), line, grid, tin and cursor. One or more snaps can be set simultaneously. The snaps are described in more detail in the next major section 4.13 Snaps.

Go to the next section 4.9.3 Tentative Pick or back to 4.9 Picking Strings.

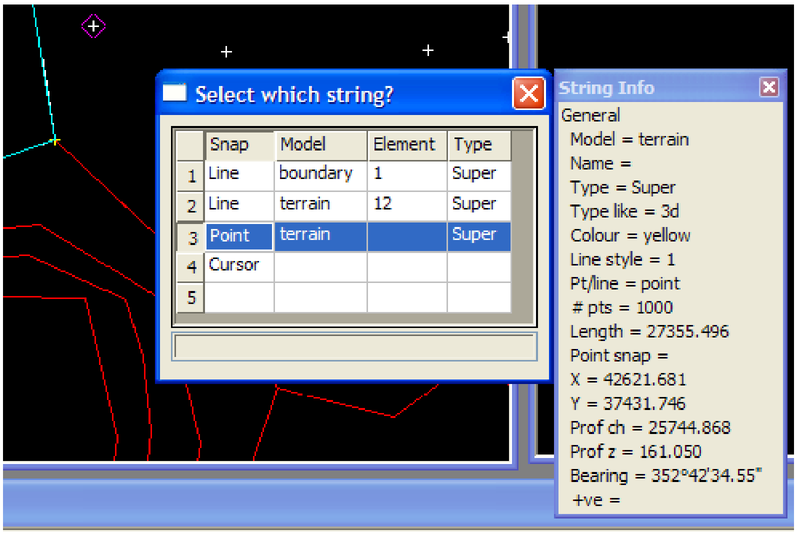
4.9.3 Tentative Pick

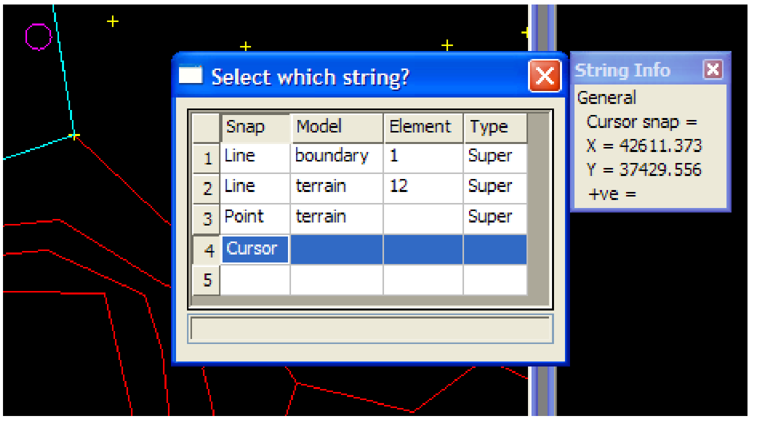
To **tentatively pick** (or tentatively select) a string, move the cursor near the string and **click LB**. The nearest string to the cursor satisfying the snap conditions (see the next section on Snaps) is highlighted and an information menu containing information about the selected string is displayed (the information varies for each string type- see the section 14.11.1 String Information).

If ***M*** snap is on, all the strings satisfying the search criteria are listed in a grid (the selection grid) as well as the Information panel.



When the selection grid is displayed, any other string in the grid can be made the considered string by simply clicking on the row containing the string in the grid. The new string will highlight and the snap position displayed.





The information menu disappears if the cursor is moved slightly.

The string name, the string model and the mouse button functions are also displayed in the Status Bar when the picked string is highlighted:

<option> [selects][accepts][menu] “model->string name”

If the correct string has been tentatively picked, **click MB** or **type <enter>** to accept the string. If the information menu was still up, it disappears when the string is accepted.

Note - if the information menu does not appear when a string is highlighted, then the *information snap* may be set to off. See the section 4.13 Snaps.

Tentative Picking and Repicking

If ***M*** snap is ***off***, and one string is tentatively picked (and highlighted) but another string was intended to be picked, click ***LB*** again without moving the cursor and the next nearest string to the current cursor position will be selected, highlighted and an information menu put up for it. Any strings already rejected during the pick will be ignored. Continuing to click ***LB*** again without moving the cursor will sequentially pick the next nearest string and ignore the earlier rejected strings.

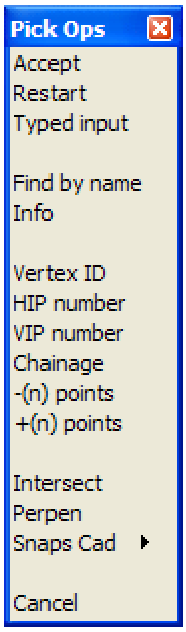
To allow **all** strings to be eligible for picking again, simply move the cursor a small distance (the reset distance - default five pixels) and start picking again. This will automatically **reset** the rejection list.

When ***M*** snap is ***on***, all strings are shown in the selection grid and any one can be picked without the need to keep clicking ***LB***.

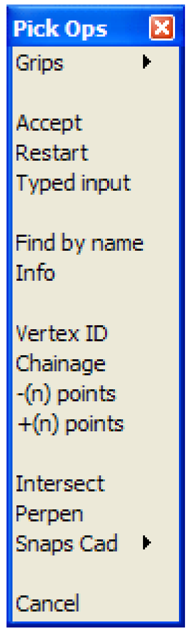
Pick Ops Menu

Options to allow all strings to be eligible for picking again, to cancel the pick operation and to accept a picked string are all available from the **Pick Ops** menu.

The **Pick Ops** menu appears whenever RB is clicked whilst in the pick option. The **Pick Ops** menu is



Alignment and Pipeline strings



For most other string types

To allow all strings to be eligible for picking again, select **Restart** from the **Pick Ops** menu. This has the same effect as moving the cursor the reset distance.

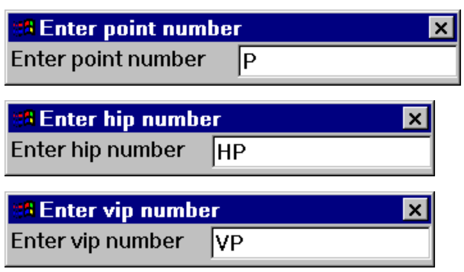
To cancel the pick without selecting any string, select **Cancel** from the **Pick Ops** menu.

Selecting **Accept** from the **Pick Ops** menu, accepts the current picked string. This is the same as clicking MB without calling up the **Pick Ops** menu.

The **Find by name** option allows the user to specify a string name to restrict the pick by. This options is also available in the snap option and will be described there.

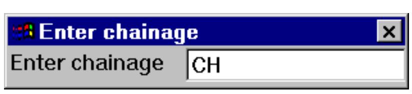
Selecting **Info** will redisplay the information menu for the string.

If **Vertex/HIP/VIP number,** is selected, the **enter vertex/hip/vip number** box is displayed.



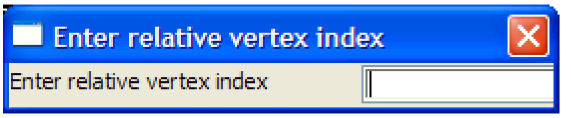
Typing the number into the box terminated by <enter> will move the pick position to that vertex or IP point.

If **Chainage** is selected, the **Enter Chainage** box is displayed.



Typing a chainage into the box terminated by <enter> will move the pick position to that chainage on the tentatively selected string.

If **-(n) vertices** or **+(n) vertices** is selected, the **Enter relative vertex index** box is displayed.



Typing a positive/negative number into the box terminated by <enter> will move the pick vertex to that many vertices after/before the selected position on the tentatively selected string.

The left and right arrow keys (**<-**, **->**) can also be used to move the pick position to the previous or following vertex respectively.

When selecting **Snaps Cad**, the **Snaps Cad Menu** appears on the screen. The **Snaps Cad Menu** can also be accessed by walking right on **Snaps Cad**. For more information please go to 4.28.4 Snaps Cad

To remove the **Pick Ops** menu without selecting an option, either select the **[X]** button from the **Pick Ops** menu title area or simply click RB again whilst the cursor is in a view.

Various snap modes can be set for the pick. The available snaps are point, line, grid, tin and cursor. One or more snaps can be set simultaneously. The snaps are described in more detail in the next section.

Go to the next section 4.9.4 Picking with Direction or back to 4.9 Picking Strings.

4.9.4 Picking with Direction

Picking a string is one operation where **12d** ***Model*** is monitoring the cursor position when button LB (or MB for fast pick) is pushed down and also the cursor position when button LB (MB) is subsequently **released**.

The **pick position** is taken as the cursor position when button LB (MB) is released, but the difference between the down and up positions for button LB (MB) defines a **direction** vector or **sense** to the pick. This direction is called the **picking direction** or **picking sense**.

The **picking direction** is used in a number of **12d** ***Model*** options.

For example, in the information panel displayed for any picking operation, if the picking direction is the same as the string direction, a **+ve** is displayed at the bottom of the panel. If the directions are opposite, then a **-ve** is displayed in the panel.

Important Note
If a string was picked with direction, then to automatically reset the rejection list, the cursor must be move fifty (50) pixels rather than the five required for a non-directional pick.

Go to the next section 4.9.5 Summarising the 12d Model Picking Mechanisms or back to 4.9 Picking Strings.

4.9.5 Summarising the 12d Model Picking Mechanisms

Tentative picking - pick and accept as separate operations

LB - left button — select the next nearest string.
moving the cursor more than five pixels resets the rejection list for a non-directional pick.
moving the cursor more than fifty (50) pixels resets the rejection list for a directional pick.

MB - middle button — accept the current selected (highlighted) string - ends the pick.

RB - right button — brings up the **Pick Ops** menu.

Typing **<Enter>** — same as MB: accepts the current selected (highlighted) string - ends the pick

Fast accepting - pick and accept in one operation if there is only one item satisfies the snap conditions or a tentative pick if more than one item.

LB - left button — select the next nearest string. If there is only one item possible then it is automatically accepted. Otherwise it is the same as for a tentative pick.
moving the cursor more than five pixels resets the rejection list for a non-directional pick.
moving the cursor more than fifty (50) pixels resets the rejection list for a directional pick.

MB - middle button — accept the current selected (highlighted) string - ends the pick.

RB - right button — brings up the **Pick Ops** menu.

Typing **<Enter>** — same as MB: accepts the current selected (highlighted) string - ends the pick

Fast picking - pick and accept in one operation

MB - middle button — pick and accepts the nearest string.
or

Typing **<Enter>** — same as MB: pick and accepts the nearest string - ends the pick.

Please continue to the next section 4.10 X Y Z and Ch Ht Typed Input Box (or back to 4.9 Picking Strings).

4.10 X Y Z and Ch Ht Typed Input Box

Many **12d** ***Model*** options use a *cursor select* (picking) in a view to pick strings, obtain information about a string, get an x, y and z coordinates for a new vertex *etc.*

In most cases, **typed input** can also be used to enter the information instead of using the cursor select. For example, an exact values for the x, y (and possibly z) co-ordinates could be typed in or a chainage and a height typed in rather than picking a position on a view.

Typed input is available in ***12d Model*** whenever a *cursor select* is used.

To use *typed input*, simply begin typing the values. As soon as a character is typed, a special *typed-input* box appears on the screen and the typed characters are automatically placed in it.

Alternatively, instead of clicking **LB** to select a string, click **RB** to bring up the **Pick Ops** menu and then select **Typed Input** from the **Pick Ops.** This will also bring up the *typed input* box.

Depending on the view type that the cursor is over, the *typed input* box will be either an *Enter X Y Z* or an E*nter Ch Ht* box. If the focus is on a Plan or Perspective view, then an **Enter X Y Z** box comes up. If the focus is on a Section view, then an **Enter Ch Ht** box appears. These are also called the *XYZ typed input* box and the *Chainage Height typed input* box respectively.

The x, y (and possibly z) co-ordinate values or chainage height required by the box are simply typed in, each value being separated by **one or more spaces**.

An <enter> terminates the typed input and sends the entered values to the option. The typed-input box then disappears.

If the user wishes to **abort** the typed input and return to mouse input, simply select the **[X]** button on the typed input box, or type <enter> with no values in the typed input box.

The *Enter X Y Z/Ch Ht* box can be **moved** (dragged) to a new position on the screen in the standard way.

The default mode for the **Enter X Y Z** box is to receive absolute *x*, *y* and *z* values.

However, by typing special character codes before the values, the **Enter X Y Z** box can be used to enter data in a variety of formats including relative co-ordinates and bearing-distance combinations.

The full list of typed input codes now follows (the text will be given in lower case, but it can be typed in either upper or lower case).

s *actual x* *actual y* *actual z*

x y z

or

a x y z

s *relative x* *relative y* *relative z*

r x y z

s bearing in degrees-minutes-seconds distance actual z

bd ddd.mmssfff distance

where ddd.mmssfff is the bearing expressed in degrees, minutes and seconds as

ddd whole degrees

. separator between degrees and minutes

mm whole minutes

ss whole seconds

fff fractions of seconds (as many as required)

- bearing in decimal degrees distance actual z

  dbd ddd.dd distance z

- angle in degrees-minutes-seconds distance actual z

  ad ddd.mmss distance z

- angle in decimal degrees distance actual z

  dad ddd.dd distance z

The default mode for the **Enter Ch Ht** box is to receive *chainage* and *height* values.

However, by typing special character codes before the values, the **Enter Ch Ht** box can be used to enter data in a variety of formats including grade distance and slope distance.

The full list of typed input codes now follows (the text will be given in lower case, but it can be typed in either upper or lower case).

- actual *chainage* actual *height*

  chainage *height*

  or

  a *chainage* *height*

- relative *chainage* relative *height*

  r *chainage* *height*

- *slope* and *distance*

  sd *slope* *distance*

  where slope is **1 vertical in slope horizontal** units

- *grade* and *distance*

  gd grade distance

  where grade is 1/slope as a percentage

Please continue to the next section 4.11 Tentative Typed Inputs.

4.11 Tentative Typed Inputs

If the string has been ***tentatively*** selected (by clicking LB and highlighting the string but not yet accepting it), then further typed inputs are available which then work for the ***highlighted*** string only.

The ***Tentative typed inputs*** are:

CH*chainage*	go to the given chainage
V*vertex-index*	go to the vertex number *vertex-index*
P*point-id*	go to vertex with the given point-id (vertex-id)
+	go to the next vertex
-> (right arrow)	go to the next vertex
+*number*	go forward *number* vertices
-	go back to the previous vertex
<- (left arrow	go back to the previous vertex
-*number*	go backward *number* vertices
HP*number*	go to horizontal IP number ***number***
VP*number*	go to vertical IP number ***number***

When a string is *tentatively* selected, the above *typed input* options are also available as *menu items* from the **Pick-Ops** menu. This has been documented in the section Pick Ops Menu.

Warning - If ***M*** snap is on and the selection grid is displayed, the tentative typed inputs will not work. Either turn ***M*** snap off, or move the cursor so the selection grid disappears before typing, or use ***RB*** to bring up the **Pick Ops** menu and select the required command from there.

Please continue to the next section 4.12 Picking Point Ids (Point Numbers, Vertex ids).

4.12 Picking Point Ids (Point Numbers, Vertex ids)

When a **12d** ***Model*** option uses a *cursor select* (picking) in a plan view to pick strings, then *typed input* can be used to select a point id (point number, vertex id) which exists in any model on the view (note that a point id can be numeric or alphanumeric).

When over the plan view, instead of clicking ***LB*** to select a string, simply type:

number to select a *numeric* point id

or

P*text* to select an *alphanumeric* point id of name *text.*

This will bring up the ***Enter X Y Z*** box and the typed information will go into the box. Type <enter> and the view will then be searched for the given point number.

If the ***point id*** exists, then a large cross will be drawn over the vertex on the view and the string containing the point id will be highlighted.

Note that if a string is *tentatively* selected (i.e. highlight but not yet accepted), then

Vn will select the nth vertex of the tentatively selected string (vertex index of n)

Please continue to the next section 4.13 Snaps.

4.13 Snaps

In most ***12d Model*** options, the mouse is used to pick data from a string displayed on the screen.

Unfortunately different options usually require different data so an easy method is needed to help select from all the strings on the screen, a specific string or part of a string.

In ***12d Model***, ***Snaps*** are used as part of the string picking mechanism (discussed in the previous sections) to help filter out what data is interactively selected on a view.

Using the **12d** ***snaps*** it is possible to

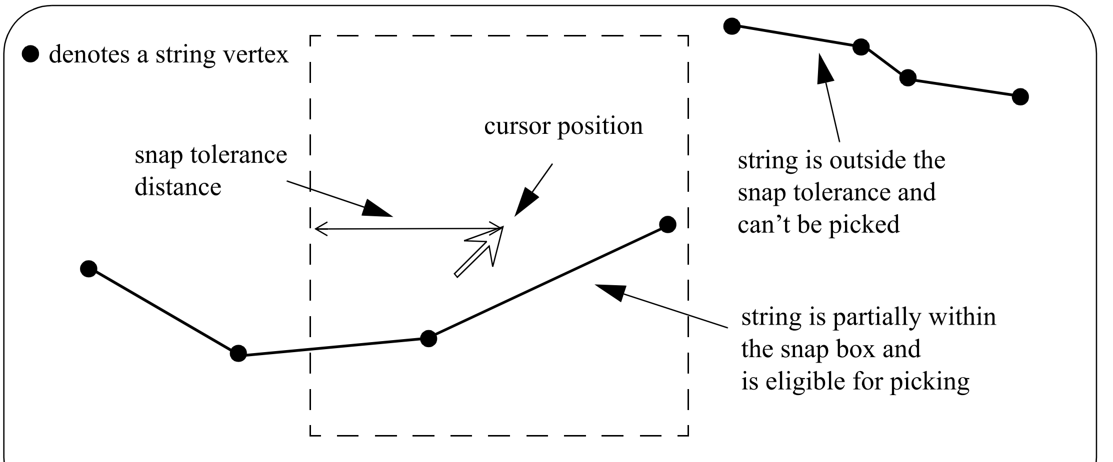
- restrict any string picks to selected string names and/or models
- set modes which are used to determine what parts of strings are used for picking
- set the search distance for picking vertices (point snap tolerance)
- set the search distance for picking (snap tolerance)
- give the name of a tin to be used to set z-values at a (x,y) position
- set whether the user is prompted for a new height (z-value) after every point edit
- turn on/off the information menu when picking
- turn on/off the ***selection*** grid when picking
-

Go to the next section 4.13.1 Point and Line Snaps or back to 4.13 Snaps.

4.13.1 Point and Line Snaps

When trying to interactively pick a string in ***12d Model***, the user moves the cursor near the string in a view and clicks LB (or the MB for a fast pick). ***12d Model*** then makes a search of all the strings on the view to make a selection.

The help restrict the search, only strings that are partially or totally within an x and y distance called the ***snap tolerance*** of the cursor are eligible for picking. That is, only strings that have part, or all, of them within the *snap box* can be selected.

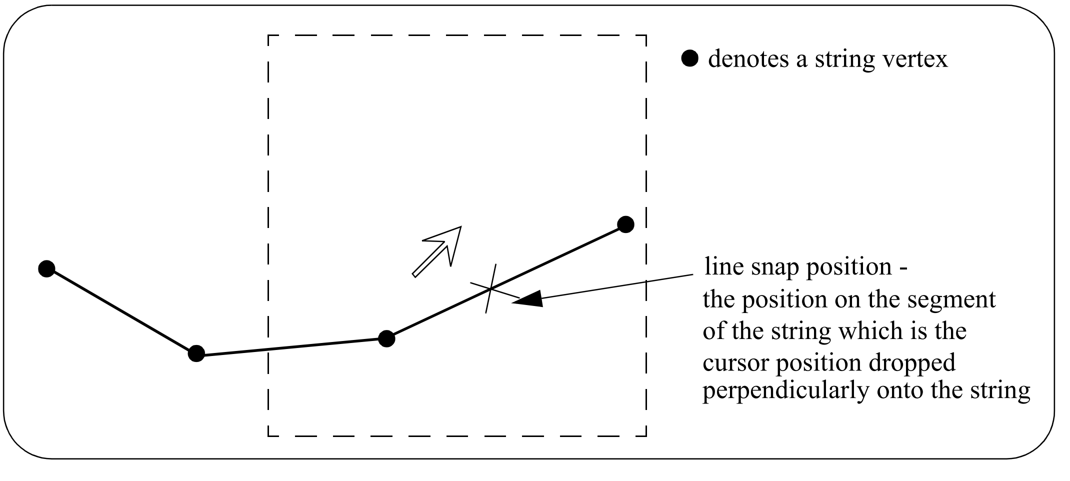


Snap Box

For strings inside the snap box, ***12d Model*** has two snap modes, *point* and *line*, to specify which parts of the strings can be selected.

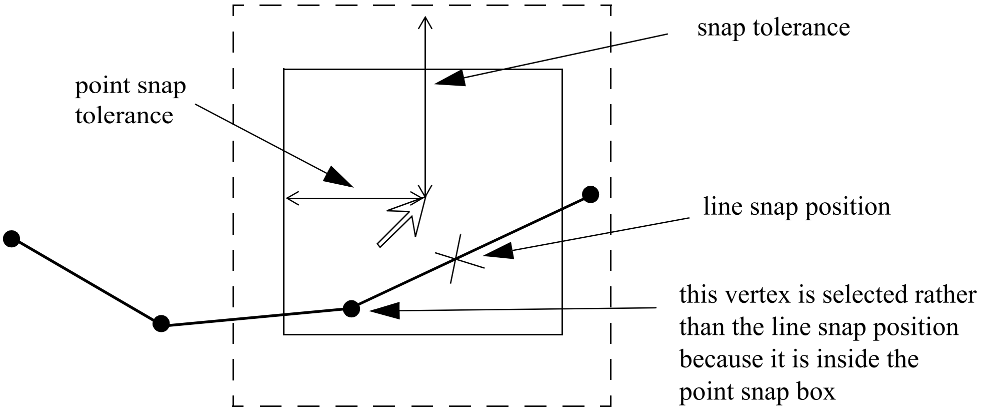
If ***point*** snap is set on, the vertices (points) of a string can be selected. If only *point* snap is on, then the string containing the ***closest vertex*** to the cursor is selected.

If ***line*** snap is set on, then the position on the string that is determined by dropping the cursor perpendicularly onto the string, can be selected. Note that there is usually no string vertex at the perpendicular position (line snap position). If only *line* snap is on, then the string containing the closest perpendicular position to the cursor is selected.



Since in general it is rare for a vertex of a string to be closer to the cursor than the line snap position, then with only the above definitions and *point* and *line* snap both on, the line snap position usually wins. Hence to pick a vertex, line snap would have to be turned off. Since this can lead to line snap being regularly toggled on and off, another distance called the ***point snap tolerance*** has been introduced to give vertices ***priority*** over line snaps when both snaps are on.

When *point snap* is set on, any vertex of a string that is within an x and y distance called the ***point snap tolerance*** of the cursor when LB is clicked, is considered for selection ***before*** any other type of snap is considered. That is, any vertices in the *point snap box* are selected before any line snap positions.



Point Snap Box and Snap Box

In the area between the point snap box and the snap box, vertices and line snap positions are treated equally and the closest one to the cursor is selected.

The other types of snaps are not as complicated and will be discussed in the following sections.

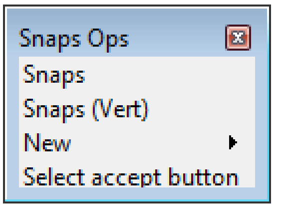
Go to the next section 4.13.2 Setting Snaps or back to 4.13 Snaps.

4.13.2 Setting Snaps

Snaps can be toggled on and off from the **H** toolbar (also know as the **Snaps** toolbar), or the snap settings can be set from the **Snaps** menu or the **V** menu under *Utilities* on the Main menu.

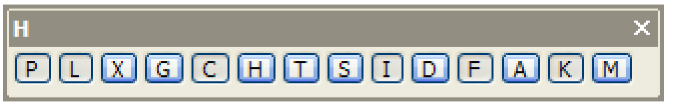
Snaps toolbar

The menu **Utilities =>Snaps** is:



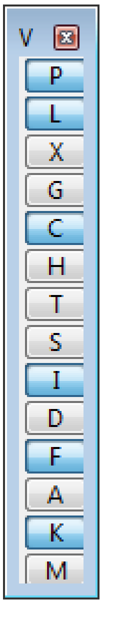
The *Snaps (Vert)* option brings up a Snaps toggle menu similar to the ***H*** (**Snaps**) toolbar except that the *Snaps (Vert)* runs vertically rather than horizontally. Either can be used to toggle the snaps on and off.

H toolbar



The Snaps (Vertical) menu

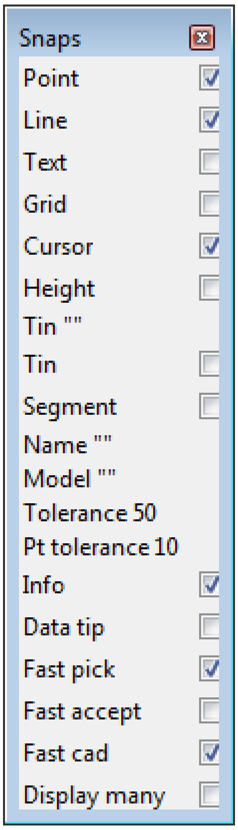
Point
Line
Te**X**t
Grid
Cursor
Height
Tin
Segment
Info
Data tip
Fast pick
fast **A**ccept
Konstruction
Many strings in select



The option **Utilities=> Snaps =>Snaps** option brings up a menu that provides tick boxes to turn the various snap modes on or off plus it is used to set the snap tolerance value, the point snap tolerance value, the tin for tin snaps and string and/or model names for restricting the snaps.

Any combination of snap modes and names can be set.

The **Snaps** menu is



point snap on/off
line snap on/off
text snap on/off
grid snap on/off
allow cursor point on/off
ask for z value at each point in edits
name of tin to snap to
tin snap on/off
turn segment snap on/off
restriction on string names
restriction on model names
set snap tolerance
set point snap tolerance
bring up info panel on/off
turns data tips on/off
turns fast pick snap on/off
turns fast accept snap on/off
turns fast construction snap on/off
display the selection grid (many strings)

The snaps point, line, grid, cursor, tin and info are all set on or off using the tick boxes.

The *snap tolerance* is displayed on the **Tolerance** line of the menu.

The *point snap tolerance* is displayed on the **Pt tolerance** line of the menu.

When a select with snap is made, the type of snap is given in the **Information** panel and also graphically displayed by **changing** the **shape** of the snap cursor. The shape of the snap cursor is

a **diamond** for a **point** snap
square for a **line** snap
circle for a **cursor** snap
circle for a **grid** snap
squiggles under a line for a **tin** snap.

Each of these snaps/settings will now be discussed.

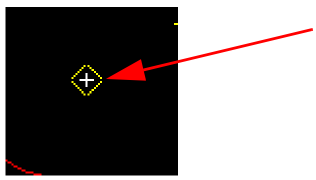
See Point Snap - P and Diamond
Line Snap - L and Square
Text Snap - X and Diamond
Grid Snap - G and Circle
Cursor Snap - C and Circle
Height - H and Enter Height Box
Tin Snap - T and Squiggles under a Line
Name - Snap Name Panel
Model - Snap Model Panel
Snap Tolerance
Point Snap Tolerance
Info Snap - I
Data Tip or Data Tool Tip - D
Fast Pick Snap - F
Fast Accept - A

Display Many Snap - M

Point Snap - P and Diamond

When **Point** (or vertex) snap is set on, any vertex of a string within the point snap box around the cursor when LB is clicked, is considered for selection before any other type of snap is considered. Centres of circles, centres of arcs and arc end points are considered to be vertices.

If the selection is made by a **Point** snap, then the cursor is shown as a **diamond** and **Point snap =** is displayed on the **Information** panel. Note that the entire string is highlighted, not just the selected vertex.



Diamond cursor for **Point** snap

Note that if there are no vertices within the point snap box then any vertices within the snap box of the cursor are considered for selecting.If the tick box is **on** (a tick), snapping to vertices is used.

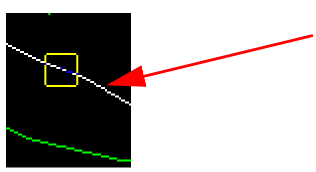
If the **P** is highlighted in the **Snaps** toolbar then **Point** snap is on. Otherwise it is off.

If the **Point** tick box is **on** (a tick) in the **Snaps** menu, snapping to vertices is used.
If the tick box is **off** (a cross or nothing), no **Point** snaps are used.

Line Snap - L and Square

When **Line** snap is set on, the cursor only needs to be within the tolerance distance of any visible segment of a string when LB is clicked, and that string is considered for selection. Also arcs and circles are considered for selection.

If the selection is made by a **Line** snap, then the cursor is shown as a **square** and **Line snap =** is written on the **Information** panel. The entire string is highlighted.



Square cursor for **Line** snap

Note that if a non-zero point snap tolerance is given and point snap is set on, then any vertices that are closer than the point snap tolerance to the cursor will be considered for snapping before any line snap is considered.

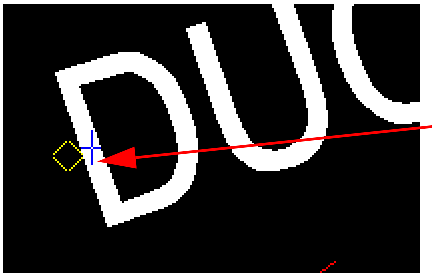
If the **L** is highlighted in the **Snaps** toolbar then **Line** snap is on. Otherwise it is off.

If the **Line** tick box is **on** (a tick) in the **Snaps** menu, dropping perpendicular to strings is used.
If the tick box is **off** (a cross or nothing), dropping perpendicular to strings is not used.

Text Snap - X and Diamond

When **Text** snap is set on, the cursor only needs to be within the tolerance distance of the bounding box of a text string when LB is clicked, and that text string is considered for selection.

If the selection is made by a **Text** snap, then the cursor is shown as a **diamond** and **Text snap =** is written on the **Information** panel. The text string is highlighted.



Diamond cursor for **Text** snap and the text is highlighted

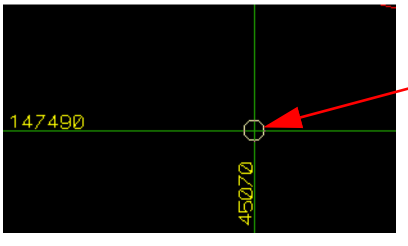
If the **X** is highlighted in the **Snaps** toolbar then **Text** snap is on. Otherwise it is off.

If the **Text** tick box is **on** (a tick) in the **Snaps** menu, **Text** snap is used.
If the tick box is **off** (a cross or nothing), **Text** snap is not used.

Grid Snap - G and Circle

When **Grid** snap is set on, the cursor will snap to the intersection of any grid lines that are displayed in a view.

If the selection is made by a **Grid** snap, then the cursor is shown as a **circle** and **Grid snap =** is displayed on the **Information** panel.



Circle cursor for **Grid** snap

Grid snap is normally used when creating new vertices, or moving vertices so they have only coordinates that are grid intersection points.

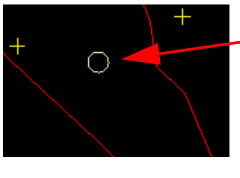
If the **G** is highlighted in the **Snaps** toolbar then **Grid** snap is on. Otherwise it is off.

If the **Grid** tick box is **on** (a tick) in the **Snaps** menu, snapping to grid intersections is used.
If the tick box is **off** (a cross or nothing), no grid intersection snaps are used.

Cursor Snap - C and Circle

If **Cursor** snap is set on and the other snaps are either not set on or have failed, the cursor position is used for the (x,y) and possibly the z value of the pick.

If the selection is made by a **Cursor** snap, then the cursor is shown as a **circle** (like grid snap) and **Cursor snap =** is displayed on the **Information** panel.



Circle cursor for **Cursor** snap.
Nothing else is highlighted because nothing except the cursor position is selected.

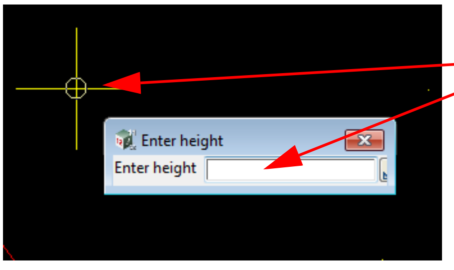
If the **C** is highlighted in the **Snaps** toolbar then **Cursor** snap is on. Otherwise it is off.

If the **Cursor** tick box is **on** (a tick) in the **Snaps** menu, snapping to the cursor position is used.
If the tick box is **off** (a cross or nothing), no **Cursor** snaps are used.

Height - H and Enter Height Box

If **Height** snap is set, then when creating and/or editing strings, the z-value for the snap position is displayed in an **Enter Height** box.

The z-value may be the z-value from a selected vertex when a **Point** snap, or a dropped position on a string when a **Line** snap, or a value from a tin when **Tin** snap is on, or nothing if it is a **Cursor** snap.



Circle cursor for **Cursor** snap and then the **Enter Height** box has no value in it

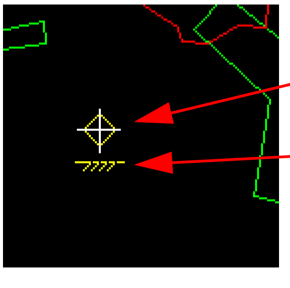
If the **H** is highlighted in the **Snaps** toolbar then **Height** snap is on. Otherwise it is off.

If the **Height** tick box is **on** (a tick) in the **Snaps** menu, the z-value for the snap position is used.
If the tick box is **off** (a cross or nothing), no **Height** snaps are used.

Tin Snap - T and Squiggles under a Line

Tin snap means that when an (x,y) point is selected, the z-value for the point is taken to be the z-value on the tin at the same (x,y) position.

Tin snap is used in conjunction with the other snaps - the other snaps control what the (x,y) location, and the z-value is then taken from the tin for that (x,y) location. So for example, the (x,y) location may be via a Point snap (and hence a Diamond cursor) and then the Tin snap squiggles is show below the Diamond cursor.



Diamond for Point snap

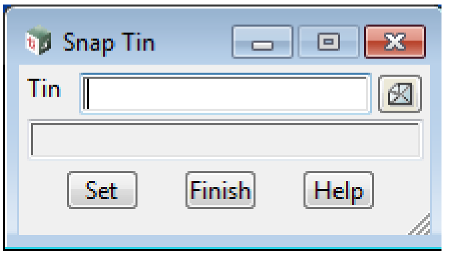
Symbol indicating z-value taken from Tin.snap

To use **Tin** snap, the name of the tin to snap to must first be given. Then, snapping to that tin is controlled by the state of the **Tin** snap.

If the **T** is highlighted in the **Snaps** toolbar then **Tin** snap is on. Otherwise it is off.

If the **Tin** tick box is **on** (a tick) in the **Snaps** menu, snapping to the tin is used.
If the tick box is **off** (a cross or nothing), no Tin snap is used.

The name of the tin to snap to is set by selecting the **Tin " "** item on the **Snaps** menu to bring up the **Snap Tin** panel and selecting the **Set** button.



When a Tin name is set by the panel, **Tin** snap is automatically set to on.

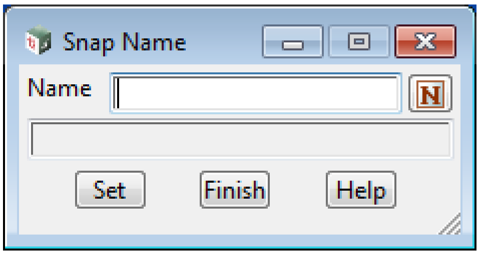
Name - Snap Name Panel

The user can restrict the snap to only strings of a specific name. The name of the strings to restrict the snap to is set by selecting the **Name " "** item on the **Snaps** menu to bring up the **Snap Name** panel.

When ***12d Model*** creates a new project, **line** and **point** snap are set to **on** and all other snaps set off.

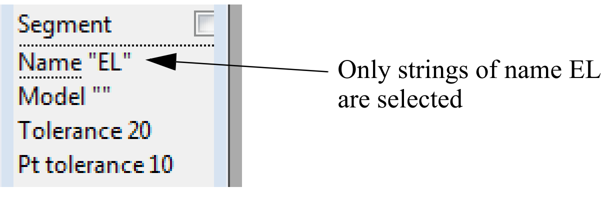
The user can restrict the snap to only strings of a specific name. The name of the string to restrict the snap to is set by selecting the **Name " "** item on the **Snaps** menu to bring up the **Snap Name** panel.

The user can restrict the snap to strings that only have a given name by selecting the **Name** option on the **Snaps** menu. This brings up the **Snap Name** panel.



The string name used to restrict the snap is entered into the **Name** field. The value are then set by selecting the **Set** button.

The string **Name** is then written to the **Snaps** menu.



Note - if there is no string name displayed in the **Snaps** menu then there is no restriction on the string names.

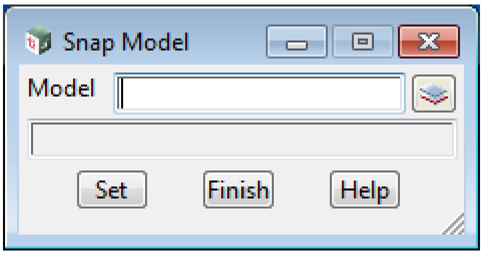
WARNING - if selecting does not appear to be working, check that there is no string name set by mistake.

Model - Snap Model Panel

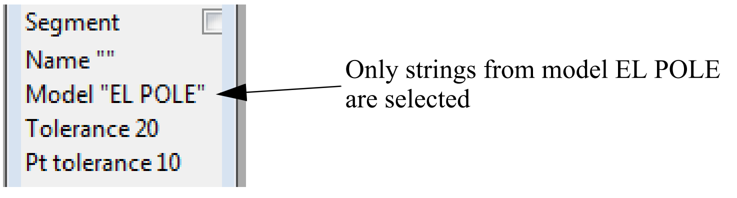
The user can restrict the snap to only strings from a specific model. The name of the model to restrict the snap to is set by selecting the **Model " "** item on the **Snaps** menu to bring up the **Snap**

Model panel.

The user can restrict the snap to strings that are only in a given model by selecting the **Model** option on the **Snaps** menu. This brings up the **Snap Model** panel. The model name used to restrict the snap is entered into the **Model** field. The values are then set by selecting the **Set** button.



The **Model** name is then written to the **Snaps** menu.



Note - if there is no model name displayed in the **Snaps** menu then there is no restriction on the models searched.

WARNING - if selecting does not appear to be working, check that there is no model set by mistake.

Snap Tolerance

Snap Tolerance is not a toggle/tick box but an item on the **Snaps** menu.

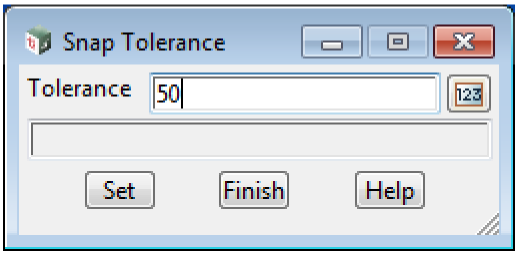
The *Snap Tolerance* value is the distance to be used for considering data when snaps are on.

The snap tolerance distance is given in screen units (pixels). There are about one thousand pixels per screen width, hence a snap tolerance of say 50 means that the snap distance about a point, line *etc.* is one twentieth of the screen width.

Snap tolerance is given in terms of screen units rather than world units because it is a distance on the screen, independent of any coordinate system being used in a view.

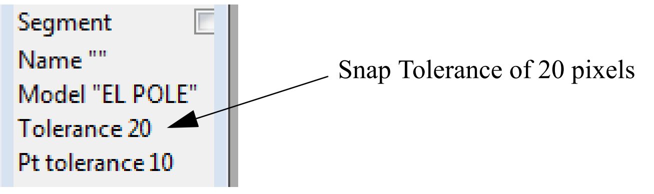
The current snap tolerance value is displayed as ***Tolerance*** on the **Snap** menu.

The *Snap tolerance* is modified by selecting the **Tolerance** item on the **Snaps** menu to bring up the **Snap Tolerance** panel.



To change the **Snap Tolerance**, type a new value into the **Tolerance** panel field and then click LB on the **Set** button.

The new **Snap Tolerance** is then written to the **Snaps** menu.



Point Snap Tolerance

Point Snap Tolerance is not a toggle/tick box but an item on the **Snaps** menu.

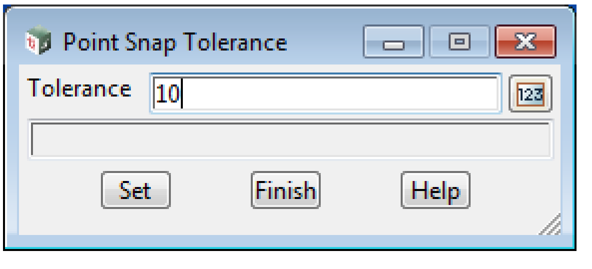
The *Point Snap Tolerance* value is the distance to be used for considering vertices over anything else when **Point** snap is on.

The Point Snap Tolerance distance is given in screen units (pixels). There are about one thousand pixels per screen width, hence a point snap tolerance of say 10 means that the point snap distance about a vertex is one hundredth of the screen width.

The *Point Snap Tolerance* is given in terms of screen units rather than world units because it is a distance on the screen, independent of any coordinate system being used in a view.

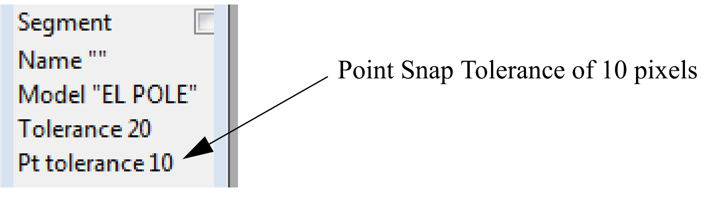
The current point snap tolerance value is displayed as ***Pt tolerance*** on the *Snap* menu.

The *Point snap tolerance* is modified by selecting the **Pt tolerance** item on the **Snaps** menu to bring up the **Point Snap Tolerance** panel.



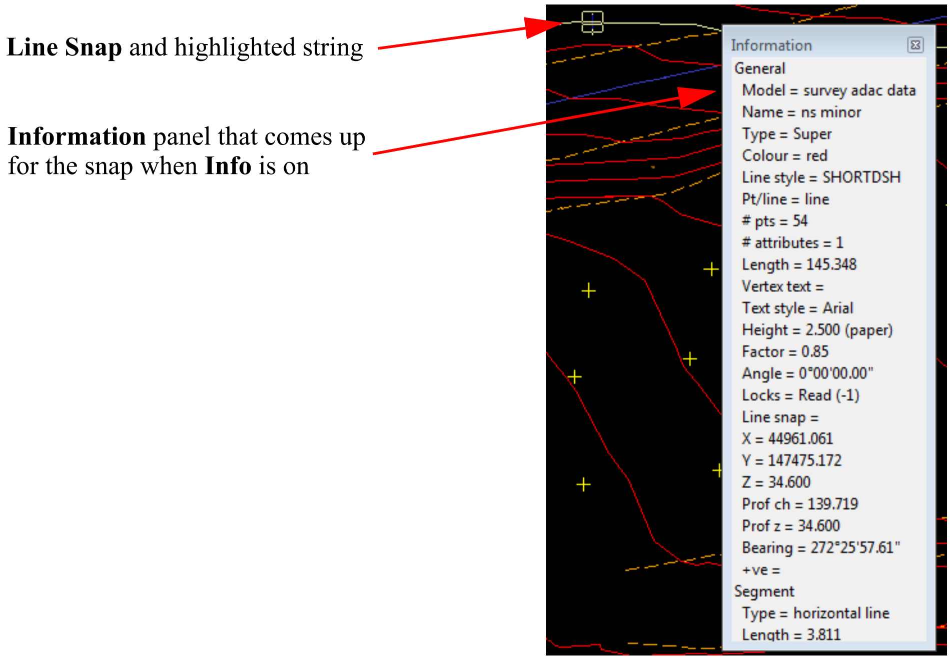
To change the **Point Snap Tolerance**, type a new value into the **Tolerance** panel field and then click LB on the **Set** button.

The new **Point Snap Tolerance** to be used to restrict the point snap is then written to the **Snaps** menu



Info Snap - I

If **Info Snap** is set, then the **Information** panel comes up whenever a string is picked.



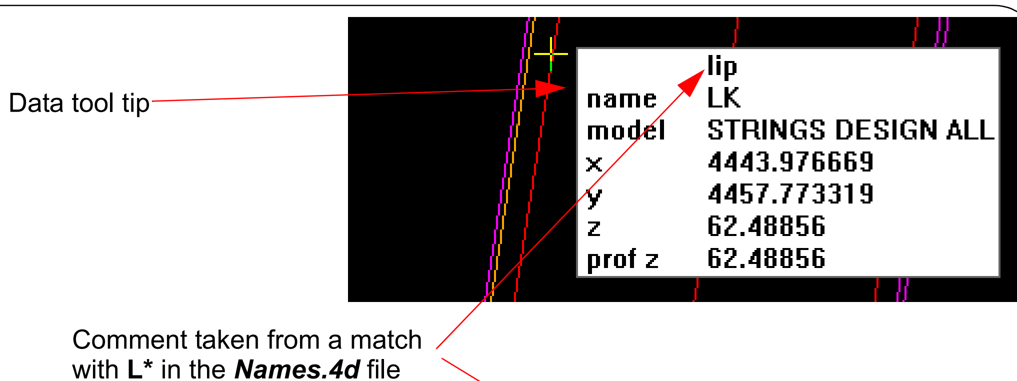
If the **I** is highlighted in the **Snaps** toolbar then **Info** snap is on. Otherwise it is off.

If the **Info** tick box is **on** (a tick) in the **Snaps** menu, then **Info** snap is on.
If the tick box is **off** (a cross or nothing), **Info** snap is off.

Data Tip or Data Tool Tip - D

If **Data Tip Snap** is set, then the Data Tipping for strings occurs. That is when the cursor is near a string, information about the string is displayed beside the cursor.

When ***Data tooltips*** are turned on (**D** snap) and a string matches a row in the ***Names.4d*** file with a comment, then the comment is displayed at the top of the Data Tool tip.



part of ***Names.4d*** file

213	J*	optic	option:	cadast	green		line	1	optional	boundary join line
214	K*	optic	option:	roads	yellow		line	1	optional	top of kerb
215	L*	optic	option:	roads	yellow		line	1	optional	lip
216	N*	optic	option:	surv	red		line	1	optional	ridge breaklines

If the **D** is highlighted in the **Snaps** toolbar then **Data Tips** snap is on. Otherwise it is off.

If the **Data tip** tick box is **on** (a tick) in the **Snaps** menu, then **Data tool tips** are displayed.

If the tick box is **off** (a cross or nothing), **Data tool tips** are not displayed

Fast Pick Snap - F

If **Fast Pick Snap** is set, then **fast picking** is allowed. That is, clicking MB will pick the nearest string to the cursor that satisfies the snap conditions. See 4.9.1 Fast Pick.

If the **F** is highlighted in the **Snaps** toolbar then **Fast Pick Snap** is on and hence Fast Picking is enabled. Otherwise it is off.

If the **Fast pick** tick box is **on** (a tick) in the **Snaps** menu, then **Fast Snap** is on.
If the tick box is **off** (a cross or nothing), **Fast Pick Snap**, and hence Fast Picking, is off

Fast Accept - A

If **Fast Accept Snap** is set, then **fast accepting** is allowed. That is, if when picking and there is only one string that satisfies the snap conditions then that string is automatically accepted. See 4.9.2 Fast Accept.

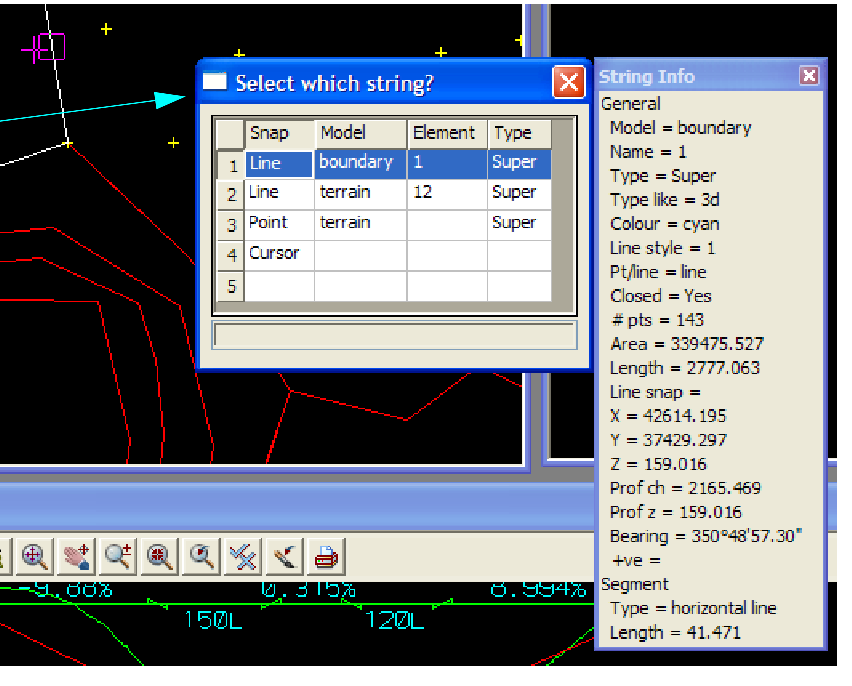
If the **A** is highlighted in the **Snaps** toolbar then **Fast Accept Snap** is on and hence Fast Accepting is enabled. Otherwise it is off.

If the **Fast accept** tick box is **on** (a tick) in the **Snaps** menu, then **Fast Accept Snap** is on.
If the tick box is **off** (a cross or nothing), **Fast Accept Snap**, and hence Fast Accepting, is off

Display Many Snap - M

If **Display Many Snap** is set, then when selecting a string, all the strings satisfying the search criteria are listed in a grid (the selection grid - **Many** strings listed, not just the one). If not set, the selection grid is not displayed. See 4.9.3 Tentative Pick.

Selection grid



If the **M** is highlighted in the **Snaps** toolbar then **Display Many Snap** is on and hence all strings satisfying the selection criteria are displayed. Otherwise it is off.

If the **Display many** tick box is **on** (a tick) in the **Snaps** menu, then **Display Many Snap** is on.
If the tick box is **off** (a cross or nothing), **Display Many Snap** is off.

NOTE: **Segment** snap (**S**) is currently under development.

When 12d **Model** creates a new project, **line** and **point** snap are set to **on** and all other snaps set off. Snap tolerance is set to 50 and Point Snap Tolerance set to 10.

When an existing project is saved, the snap settings, snap tolerance and positions of any **snaps** menus on the screen, are also saved.

Please continue to the next section 4.14 Text Grips (or back to 4.13 Snaps).

4.14 Text Grips

When text is selected for editing, ***grips*** are displayed for quick editing. Vertex text has four grips and segment text three grips.

Note - see the section 4.6 Text Definitions for the definitions of vertex and segment text, text justification point, text height, text angle, text offset and text raise.

Go to the next section 4.14.1 Vertex Text or back to 4.14 Text Grips.

4.14.1 Vertex Text

When vertex text is selected for editing, four ***grips*** are displayed for

(a) modifying the text height (diamond)

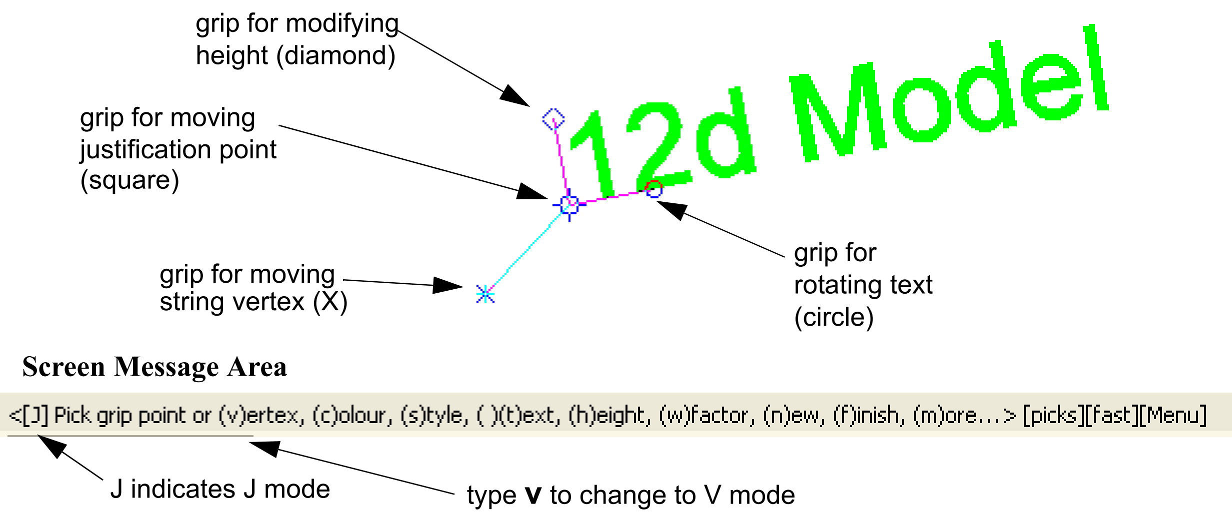
(b) rotating the text around the text justification point (circle)

(c) moving the text justification point (square)

(d) moving the string vertex (**X**)

There are two modes for the grips:

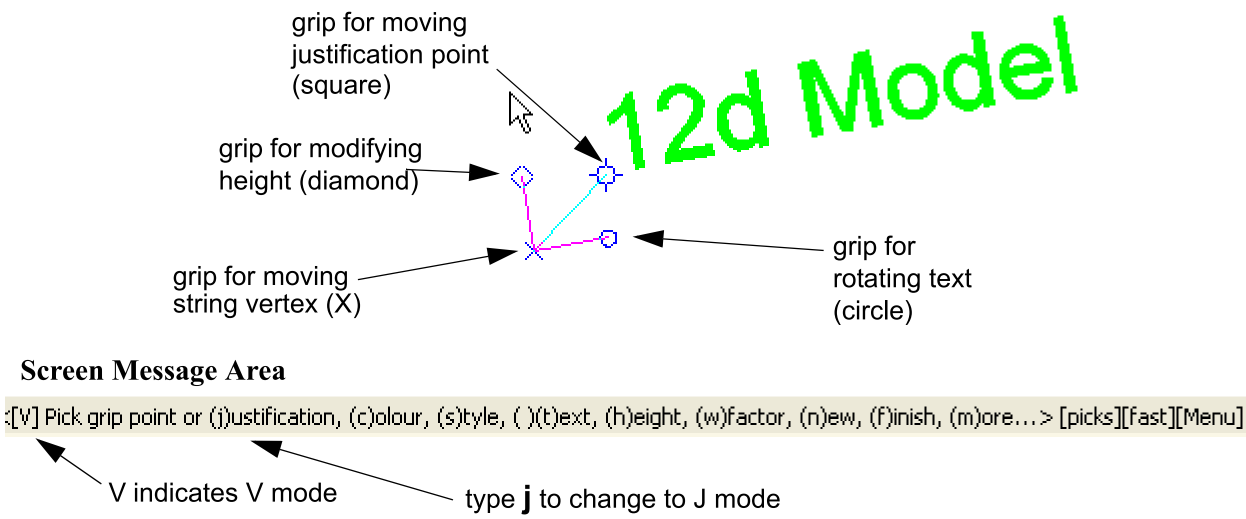
The grips for modifying text height (diamond) and rotating the text (circle) are joined to the justification point to form a right angle (J mode).



Grips on the Justification Point for Vertex Text - J Mode

OR

The grips for modifying text height (diamond) and rotating the text (circle) are joined to the text vertex to form a right angle (V mode).



Grips on the Vertex for Vertex Text - V Mode

The main difference between the J and the M modes is that in J mode, the rotation is about the text justification point, and in V mode the rotation is about the text vertex.

The appropriate grip is then selected for quick editing.

Go to the next section 4.14.2 Segment Text or back to 4.14 Text Grips.

4.14.2 Segment Text

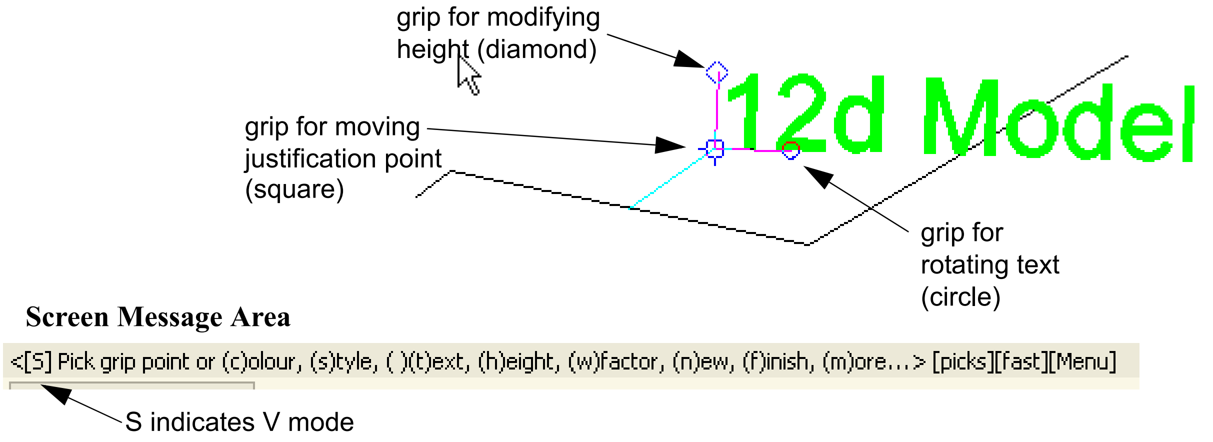
When segment text is selected for editing, three ***grips*** are displayed for

(a) modifying the text height (diamond)

(b) rotating the text around the text justification point (circle)

(c) moving the text justification point (square)

For segment text, the grips for modifying text height (diamond) and rotating the text (circle) are joined to the justification point to form a right angle (S mode).



Grips on the Justification Point for Segment Text - S Mode

The appropriate grip is then selected for quick editing.

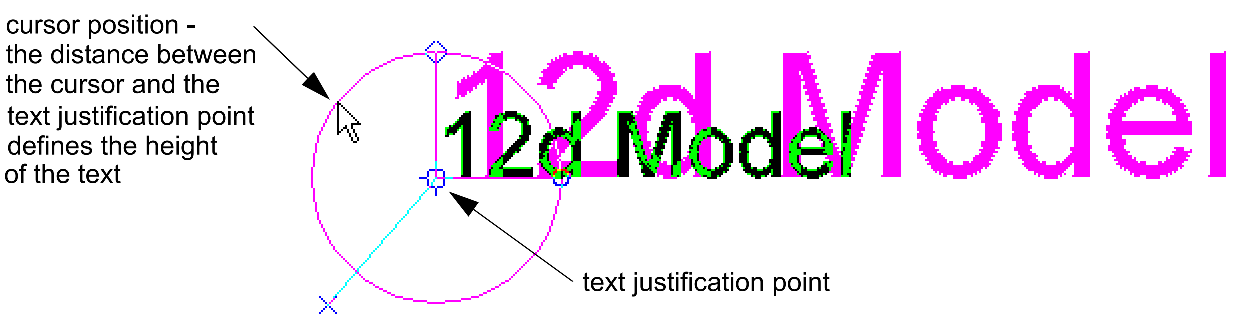
Note - there is no equivalent of the Vertex grip for segment text because the segment text is locked to the centre point of the segment rather than a vertex.

See the section 4.6 Text Definitions for the definitions of vertex and segment text, text justification point, text height, text angle, text offset and text raise.

4.14.3 **Height or Size Grip (diamond):**

J Mode

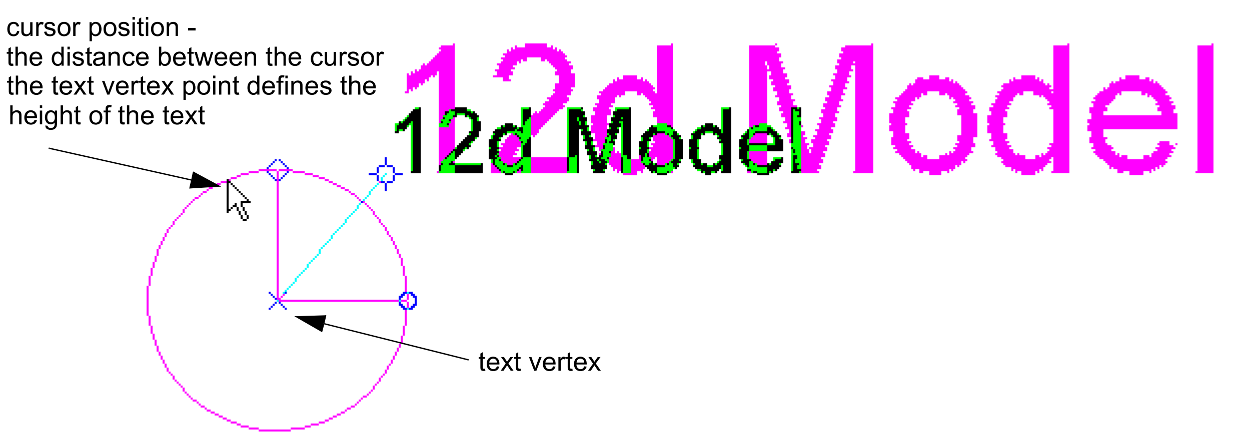
In J mode: if the ***Height* grip** is selected, a circle is drawn from the text justification point to the Height grip and this represents the height of the text. As the cursor is moved, the distance from the cursor to the text justification point is taken as the circle radius and the text height.



Modifying Height - J Mode

V Mode

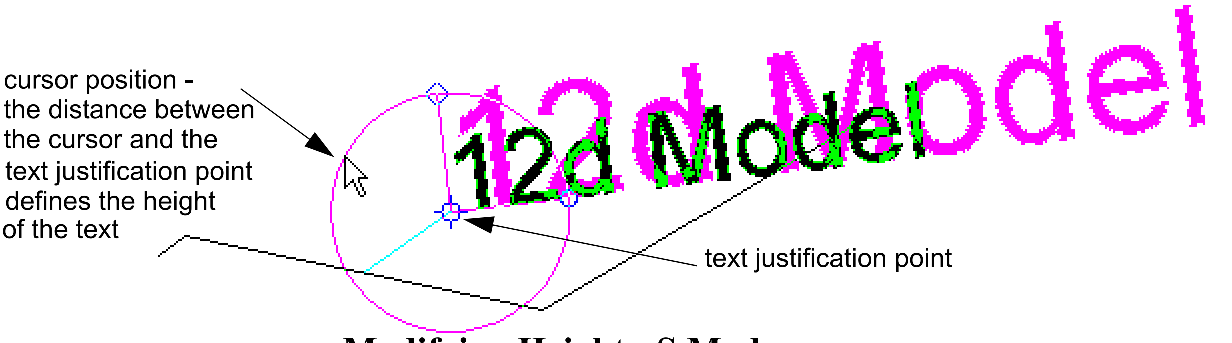
In V mode: if the ***Height* grip** is selected, a circle is drawn from the text vertex to the Height grip and this represents the height of the text. As the cursor is moved, the distance from the cursor to the text vertex is taken as the circle radius and the text height.



Modifying Height - V Mode

S Mode

In S mode: if the ***Height* grip** is selected, a circle is drawn from the text justification point to the Height grip and this represents the height of the text. As the cursor is moved, the distance from the cursor to the text justification point is taken as the circle radius and the text height.

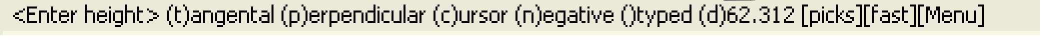


Modifying Height - S Mode

In J, V and S mode, the text height is dynamically displayed as part of the *Enter height* prompt in the screen message area

Screen Message Area

dynamic height value



The text height is set when the cursor position is accepted.

When the text is being dynamically sized, pressing the **d** key brings up the **Enter distance** typed input box with the current **dynamic** text height in it.

An ***exact*** text height can be typed into the ***Enter distance*** **Typed Input** box and accepted by pressing <Enter>.

If instead of typing **d,** the <space bar> is pressed, the ***Enter distance*** **Typed Input** box is displayed with a space in it. The required height can be typed into the box and accepted by pressing <Enter>.

If instead of typing **d,** any character other than **t**, **p**, **c** or **n** is typed (for example the beginning of the exact height required), the ***Enter distance*** **Typed Input** box is displayed with the typed characters in it. The height is accepted by pressing <Enter>.

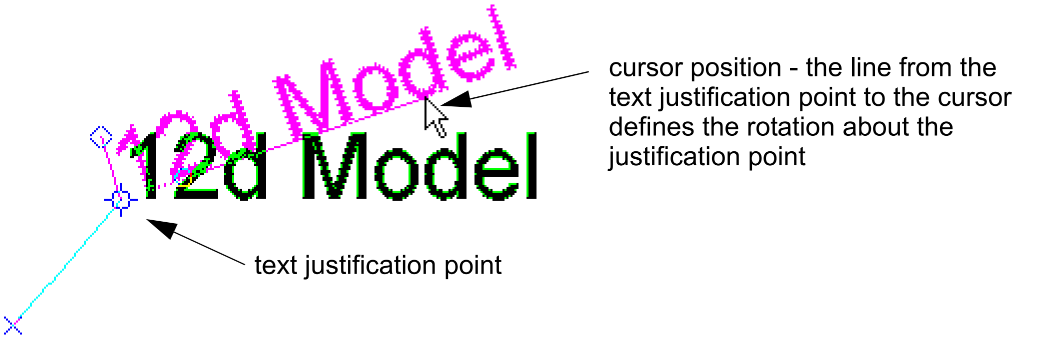
For more information on typing **t**, **p**, **c** or **n**, go to the section 4.14.7 Typing t, p, c or n

To return to dynamic sizing without entering a distance into the ***Enter distance*** input box, simply select the **X** on the top of the input box. The box will disappear and dynamic sizing will resume.

4.14.4 **Rotate Grip (circle)**:

J Mode

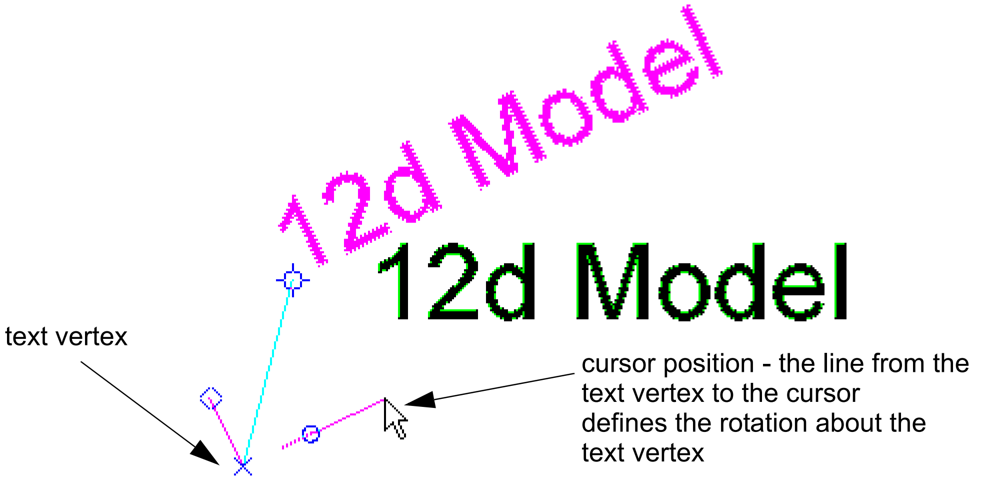
In J mode: if the **Rotate grip** is selected, a line from the text justification point to the cursor position is taken as the text angle. As the cursor is moved, the angle from the cursor to the text justification point defines the text angle.



Modifying Angle - J Mode

V Mode

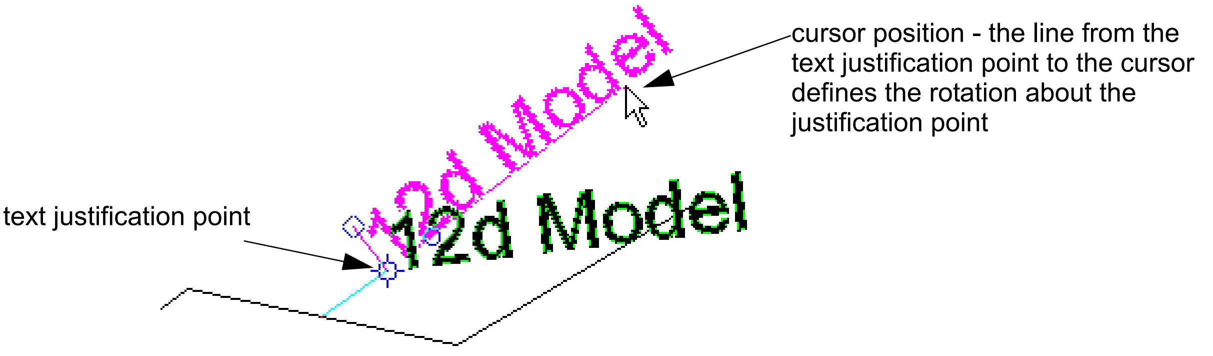
In V mode: if the **Rotate grip** is selected, a line from the text vertex to the cursor position is taken as the text angle. As the cursor is moved, the angle from the cursor to the text vertex defines the text angle.



Modifying Angle - V Mode

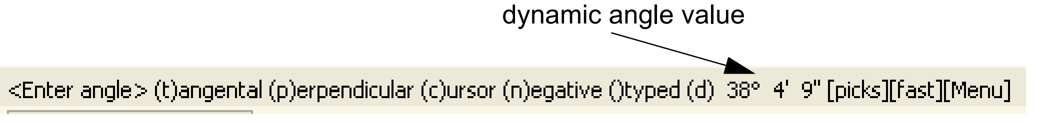
S Mode

In S mode: if the **Rotate grip** is selected, a line from the text justification point to the cursor position is taken as the text angle. As the cursor is moved, the angle from the cursor to the text justification point defines the text angle.



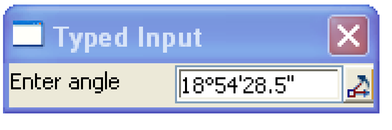
Modifying Angle - S Mode

In J, V and S mode, the angle of the text is displayed as part of the *Enter angle* prompt in the screen message area.



The text angle is set when the cursor position is accepted.

When the angle is being dynamically changed, pressing the **d** key brings up the **Enter angle** typed input box with the current **dynamic** text angle in it.



An ***exact*** text angle can be typed into the ***Enter angle*** **Typed Input** box and accepted by pressing the <enter key>.

If instead of typing **d,** the <space bar> is pressed (shown as **()**), the ***Enter angle*** **Typed Input** box is displayed with a space in it. The required angle can be typed into the box and accepted by pressing the <enter key>.

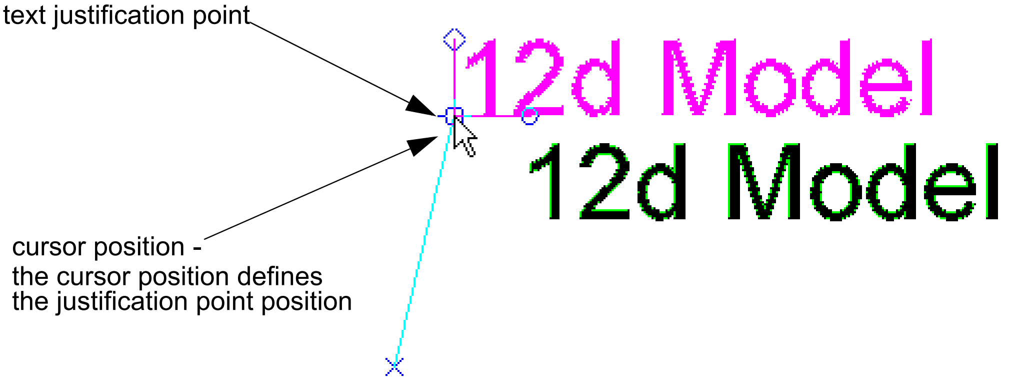
If instead of typing **d,** any character other than **t**, **p**, **c** or **n** is typed (for example the beginning of the exact angle required), the ***Enter angle*** **Typed Input** box is displayed with the typed characters in it. The angle is accepted by pressing the <enter key>.

For more information on typing **t**, **p**, **c** or **n**, go to the section 4.14.7 Typing t, p, c or n

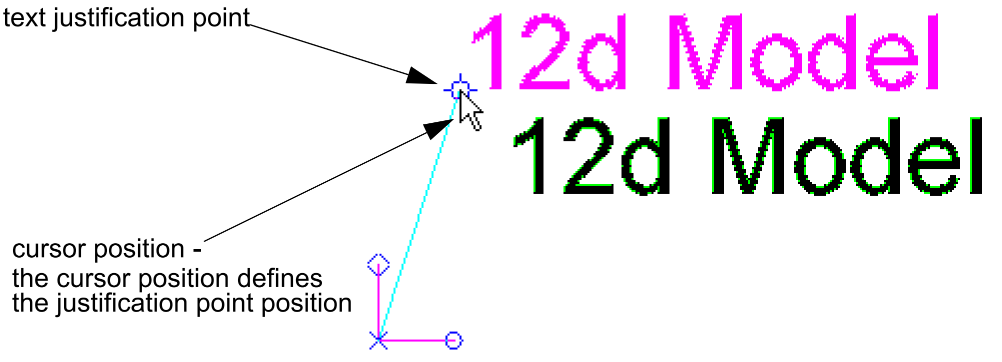
To return to dynamic angling without entering an angle into the ***Enter angle*** input box, simply select the **X** on the top of the input box. The box will disappear and dynamic angling will resume.

4.14.5 **Justification Grip (square)**:

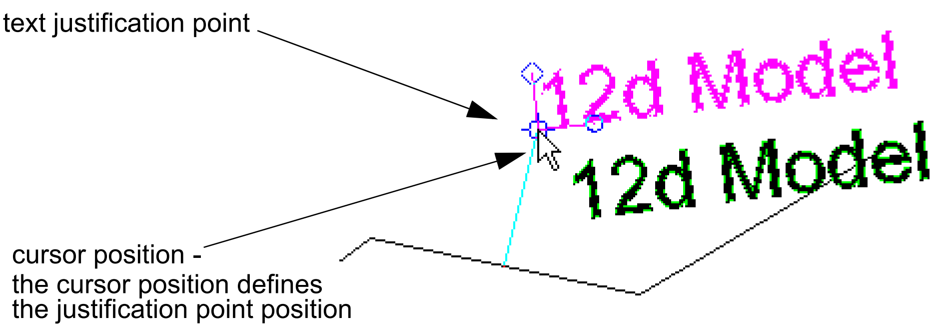
In J, V and S Mode: if the **Justification grip** is selected, the cursor position is taken as the text justification point. As the cursor is moved, the text justification point moves with it.



Moving Text Justification Point - J Mode

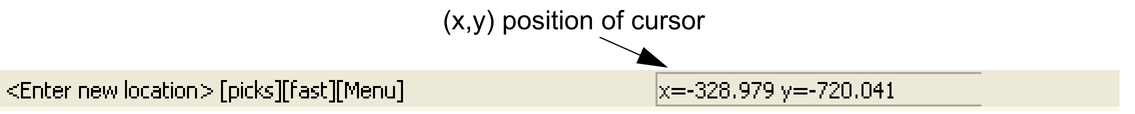


Moving Text Justification Point - V Mode



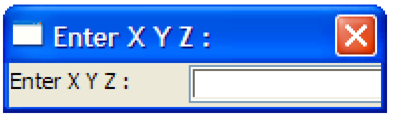
Moving Text Justification Point - S Mode

In J, V and S mode, the (x,y) position of the text is displayed in the screen message area.



The text justification point is set when the cursor position is accepted.

An ***exact*** coordinate can be entered by hitting the <space bar> or by starting to type a coordinate. This brings up the **Enter X Y Z** typed input box. The *x and y coordinates* are typed in, separated by a space, and the <enter> key pressed.



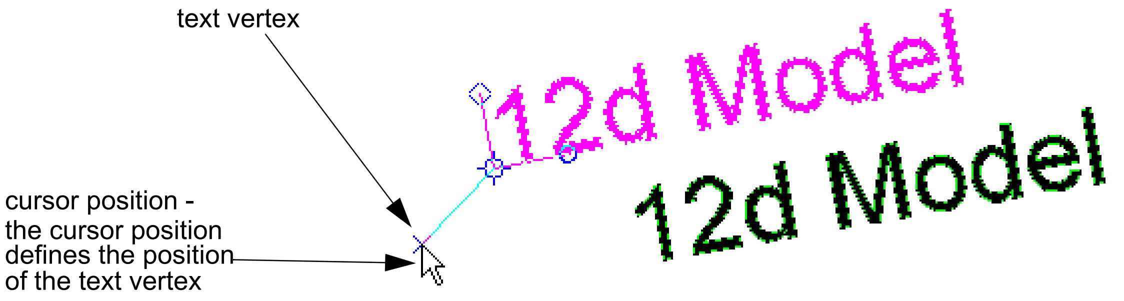
To return to dynamic moving without entering a coordinate, simply select the **X** on the top of the input box. The box will disappear and dynamic moving will resume.

Go to the next section 4.14.6 Vertex Grip (X): - for Vertex Text Only or back to 4.14 Text Grips.

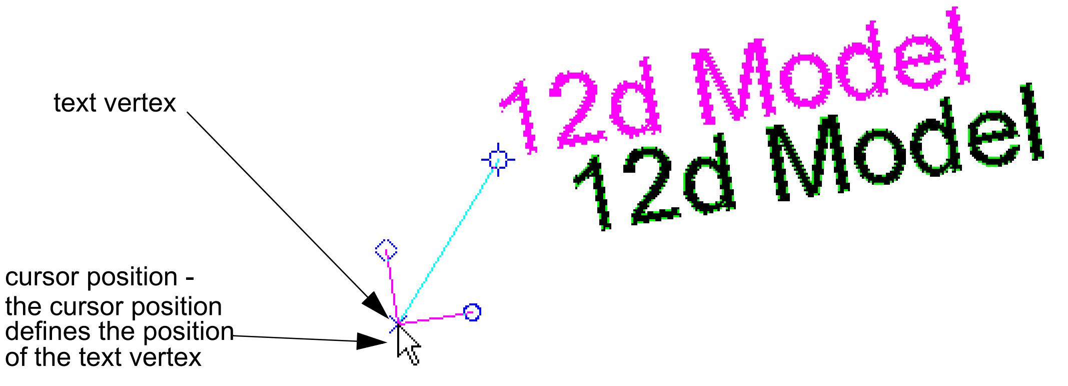
4.14.6 Vertex Grip (X): - for Vertex Text Only

In J and V mode: if the **Vertex grip** is selected, the cursor position is taken as the text vertex, As the cursor is moved, the text vertex moves with it.

Note - the Vertex can not be selected when it is in ***locked*** mode.

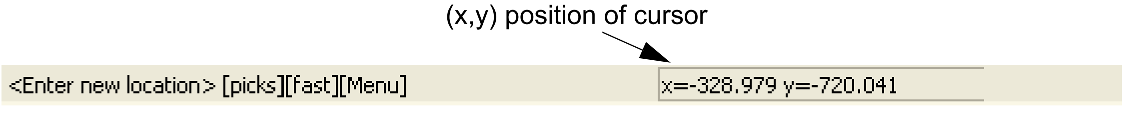


Moving Text Vertex - J Mode



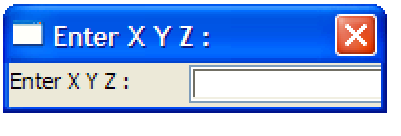
Moving Text Vertex - V Mode

In both J and V mode, the (x,y) position of the text is displayed in the screen message area.



The text vertex is set when the cursor position is accepted.

An ***exact*** coordinate can be entered by hitting the <space bar> or by starting to type a coordinate. This brings up the **Enter X Y Z** typed input box. The *x and y coordinates* are typed in, separated by a space, and the <enter> key pressed.



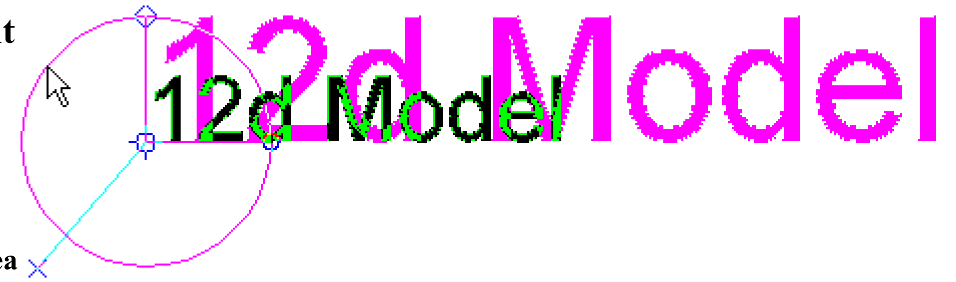
To return to dynamic moving without entering a coordinate, simply select the **X** on the top of the input box. The box will disappear and dynamic moving will resume.

Go to the next section 4.14.7 Typing t, p, c or n or back to 4.14 Text Grips.

4.14.7 Typing t, p, c or n

After selecting the ***Height*** grip (diamond) or the ***Rotate*** grip (circle), the screen message area shows that there are the typed option **t**, **p**, **c**, **n**, **()** and **d** available.

Modifying Height



Screen Message Area

<Enter height> (t)angental (p)erpendicular (c)ursor (n)egative ()typed (d)35.467 [picks][fast][Menu]

Modifying Angle

Screen Message Area

<Enter angle> (t)angental (p)erpendicular (c)ursor (n)egative ()typed (d) 38° 4' 9" [picks][fast][Menu]

Type **t**: snap cursor tangential to a selected string - go to Snap Tangential - Typing t

Type **p**: snap cursor perpendicular to a selected string - go to Snap Perpendicular - Typing p

Type **c**: return to cursor movement - go to Cursor Mode - Typing c

Type **n**: return to cursor movement - go to Negative Angle - Typing n

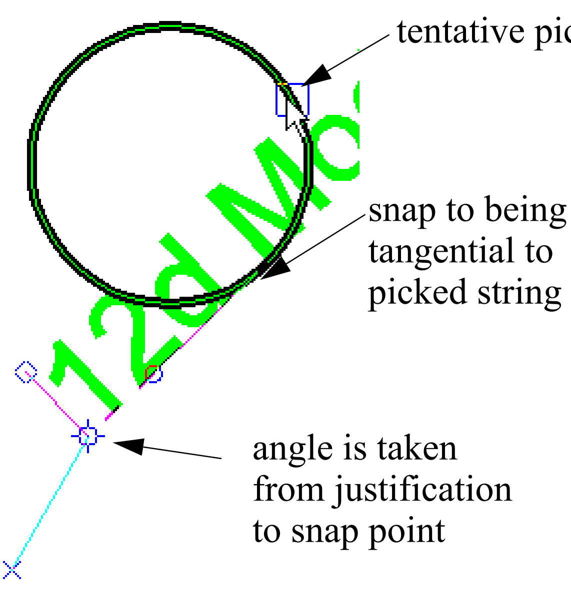
Type **<space bar>**: bring up typed input box for value - go to Hitting Space Bar

Type **d**: bring up dynamic value - go to Display Dynamic Value - Typing d

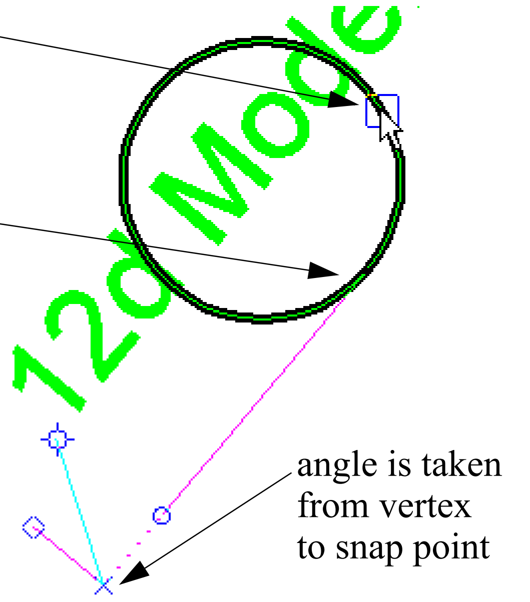
Snap Tangential - Typing t

Snap Tangential - Rotation Grip (Circle)

After selecting the ***Rotation*** grip (circle), typing **t** puts the string select into tangential mode and when a string is tentatively picked, the snap point moves to make the line from the snap point to the text justification point in J mode (or vertex in V-mode) tangential to the selected string.



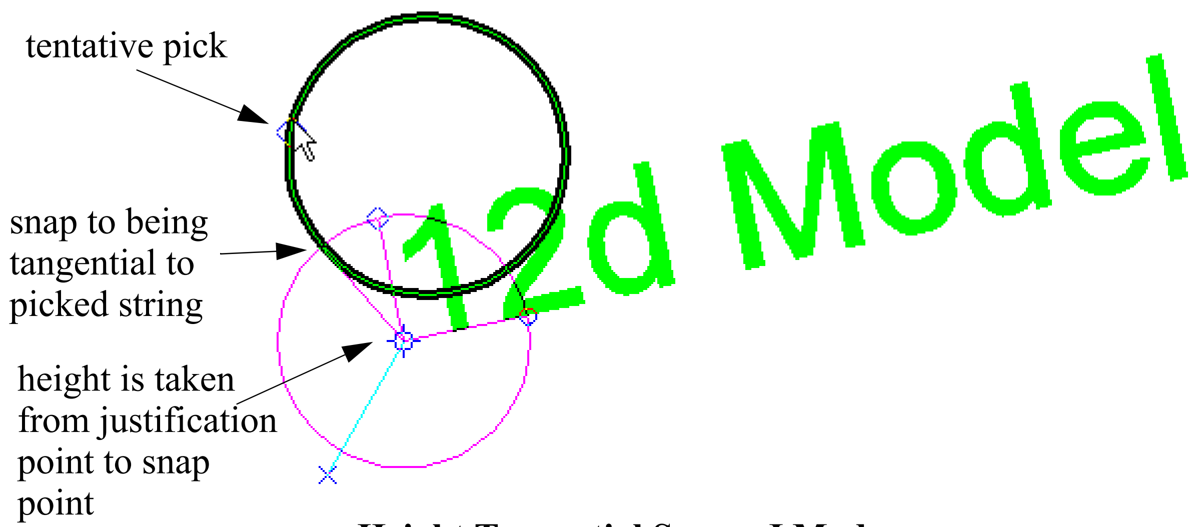
Rotation Tangential Snap - J Mode



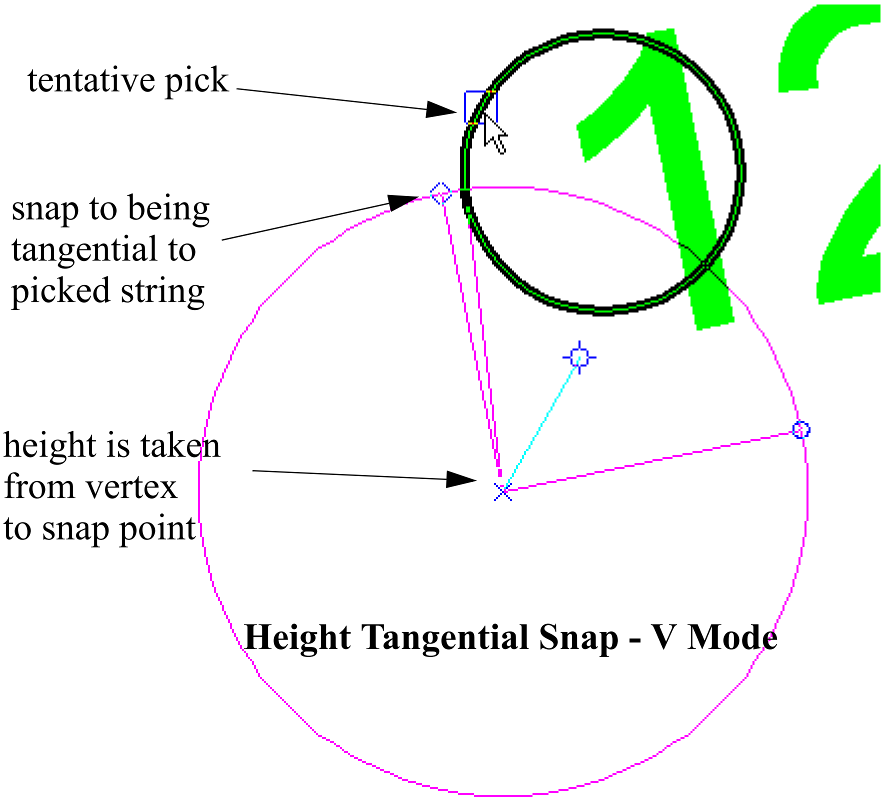
Rotation Tangential Snap - V Mode

Snap Tangential - Height Grip (Square)

After selecting the ***Height*** grip (diamond), typing **t** puts the string select into tangential mode and when a string is tentatively picked, the snap point moves to make the line from the snap point to the text justification point in J mode (or vertex in V-mode) tangential to the selected string.



Height Tangential Snap - J Mode



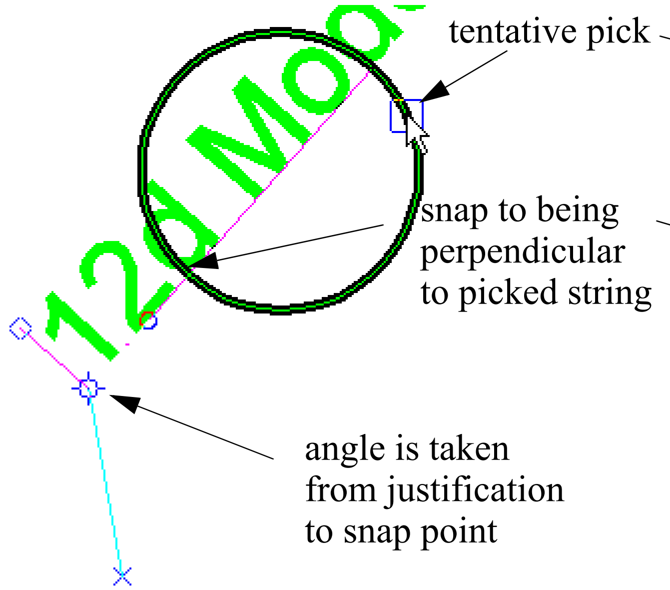
Height Tangential Snap - V Mode

Before the accept button is selected, **p** can be typed to toggle to perpendicular snap mode, **c** to return to using the cursor position or **d** to bring up the dynamic value.

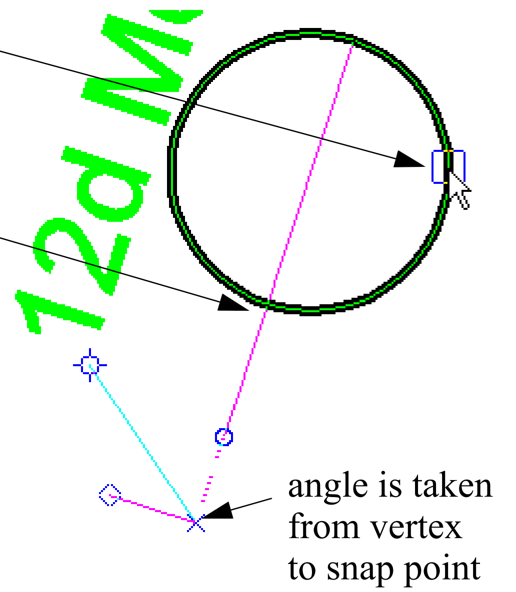
Snap Perpendicular - Typing p

Snap Perpendicular - Rotation grip (Circle)

After selecting the ***Rotation*** grip (circle), typing **p** puts the string select into perpendicular mode and when a string is tentatively picked, the snap point moves to make the line from the snap point to the text justification point in J mode (or vertex in V-mode) perpendicular to the selected string.



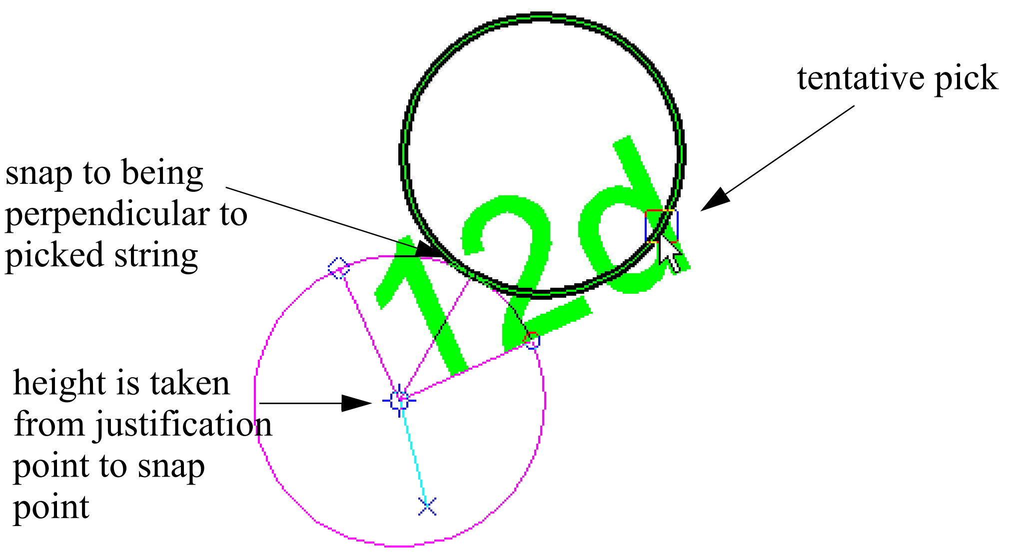
Rotation Perpendicular Snap - J Mode



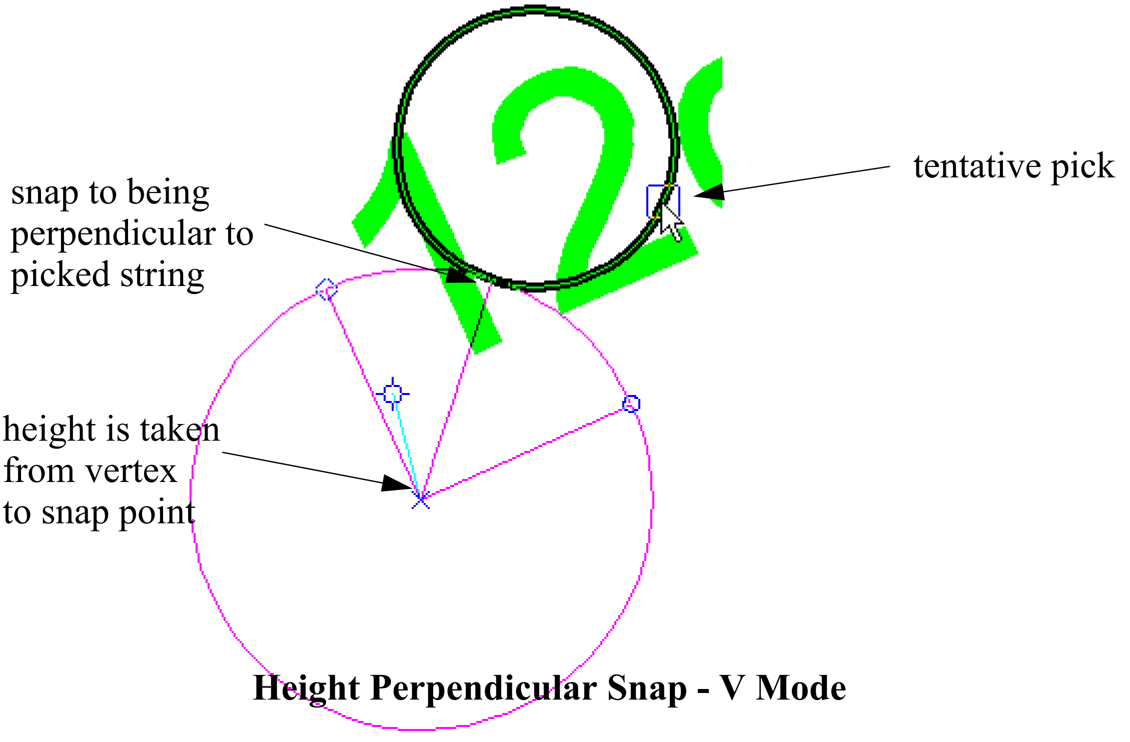
Rotation Perpendicular Snap - V Mode

Snap Perpendicular - Height grip (Square)

After selecting the ***Height*** grip (diamond), typing **p** puts the string select into perpendicular mode and when a string is tentatively picked, the snap point moves to make the line from the snap point to the text justification point in J mode (or vertex in V-mode) perpendicular to the selected string.



Height Perpendicular Snap - J Mode



Height Perpendicular Snap - V Mode

Before the accept button is selected, **t** can be typed to toggle to tangential snap mode, **c** to return to using the cursor position or **d** to bring up the dynamic value.

Cursor Mode - Typing c

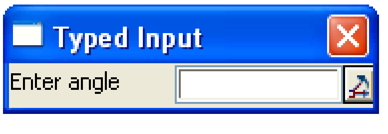
If in either Tangential or Perpendicular mode, typing **c** return to cursor mode.n

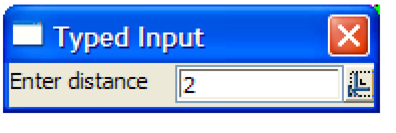
Negative Angle - Typing n

Adds 180 degrees to the current angle.

Hitting Space Bar

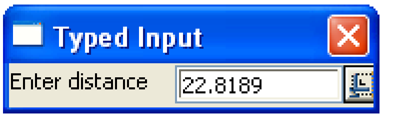
Brings up the **Enter angle** (modifying angle) or **Enter distance** (modifying height) Typed input box with no value in it.





Display Dynamic Value - Typing d

Brings up the **Enter angle** (modifying angle) or **Enter distance** (modifying height) Typed input box with the *dynamic* value in it.



Please continue to the next section 4.15 Symbol Grips.

4.15 Symbol Grips

When a symbol is selected for editing, four ***grips*** are displayed for quick editing.

Note - see the section 4.7 Symbol Definitions for the definitions of symbol justification point, symbol angle, x offset and y offset.

When a symbol is selected for editing, four ***grips*** are displayed for

(a) modifying the symbol height (diamond)

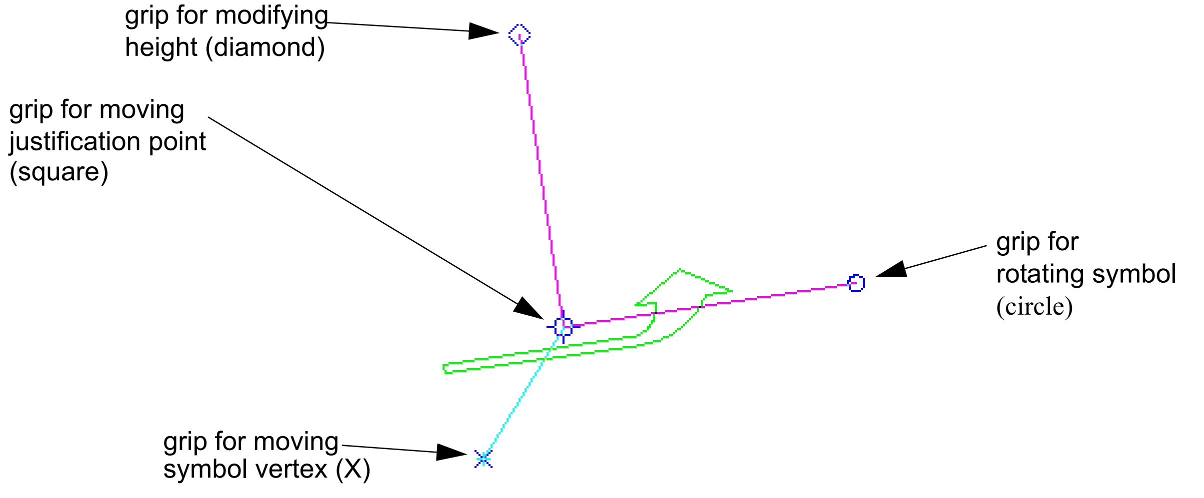
(b) rotating the symbol around the symbol justification point (circle)

(c) moving the symbol justification point (square)

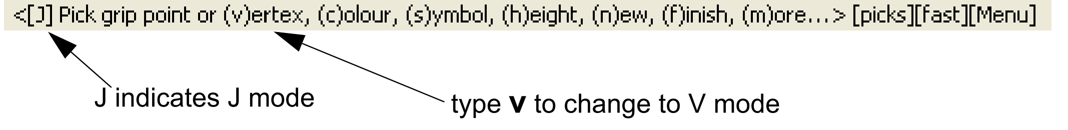
(d) moving the string vertex (**X**)

There are two modes for the grips:

The grips for modifying symbol height (diamond) and rotating the symbol (circle) are joined to the justification point to form a right angle (J mode).



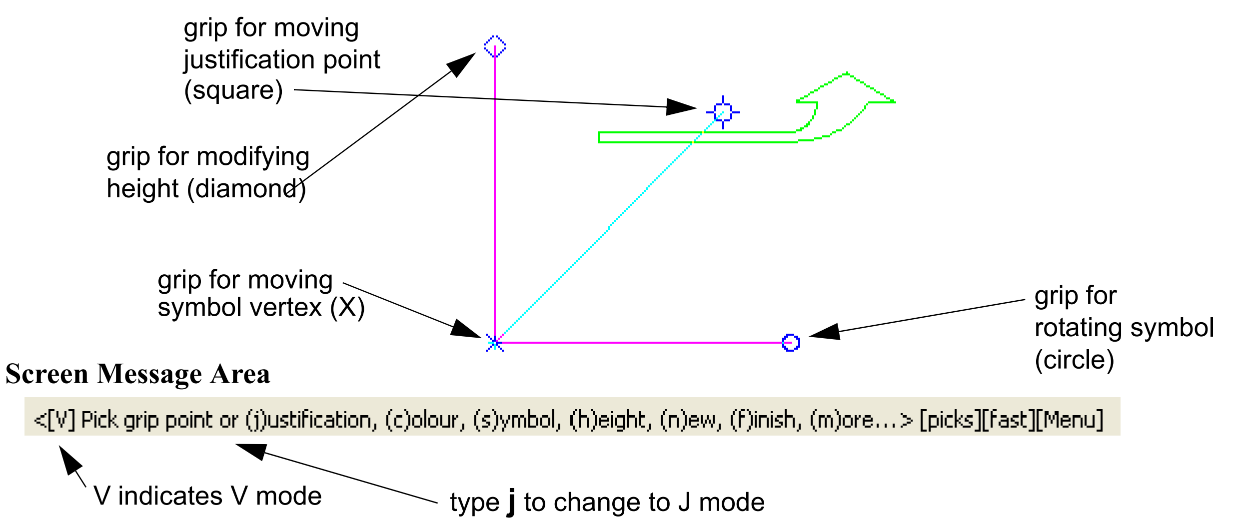
Screen Message Area



Grips on the Justification Point for a Symbol- J Mode

OR

The grips for modifying symbol height (diamond) and rotating the symbol (circle) are joined to the symbol vertex to form a right angle (V mode).



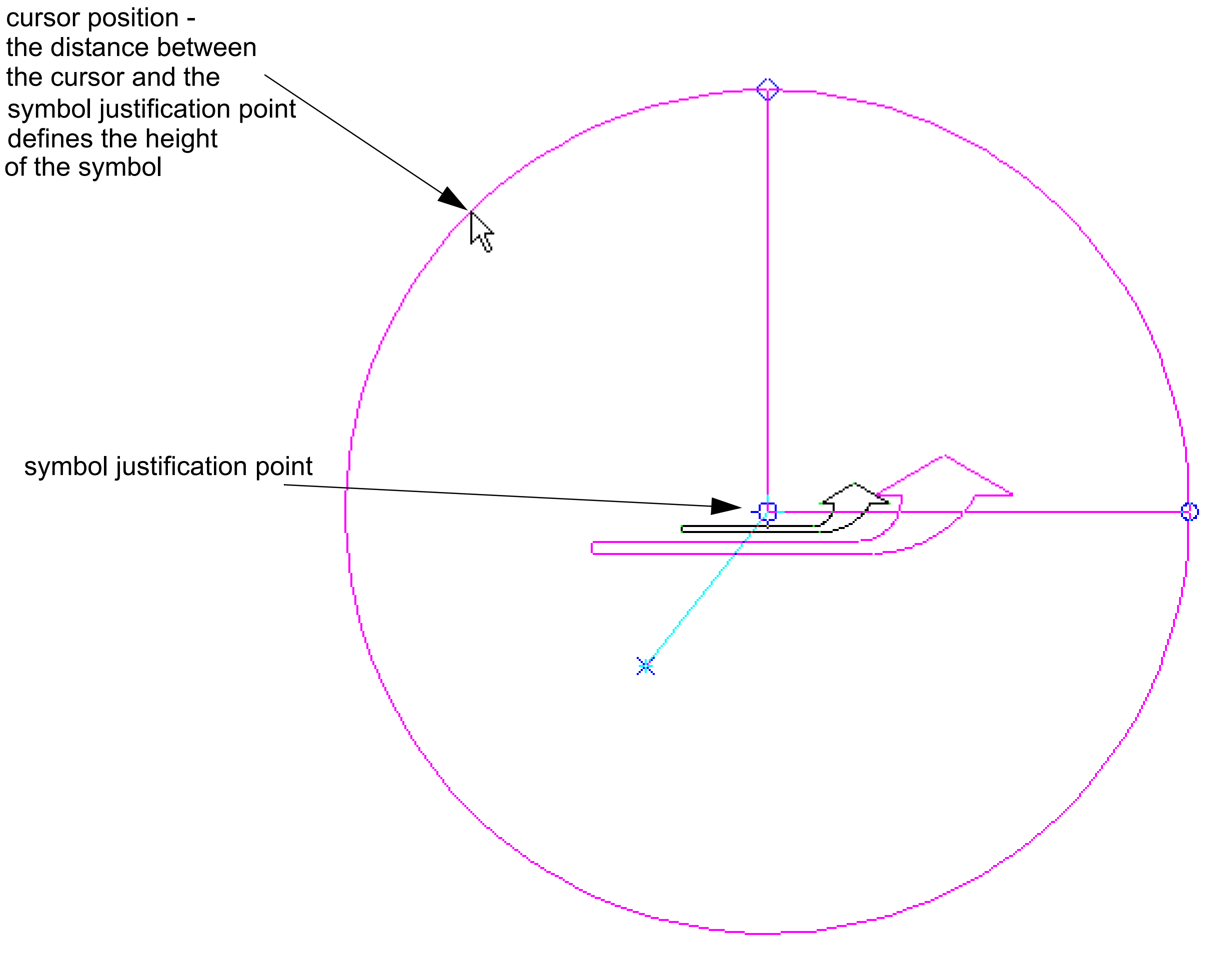
Grips on the Vertex for a Symbol - V Mode

The appropriate grip is then selected for quick editing.

4.15.1 **Height or Size Grip (diamond)**:

J Mode

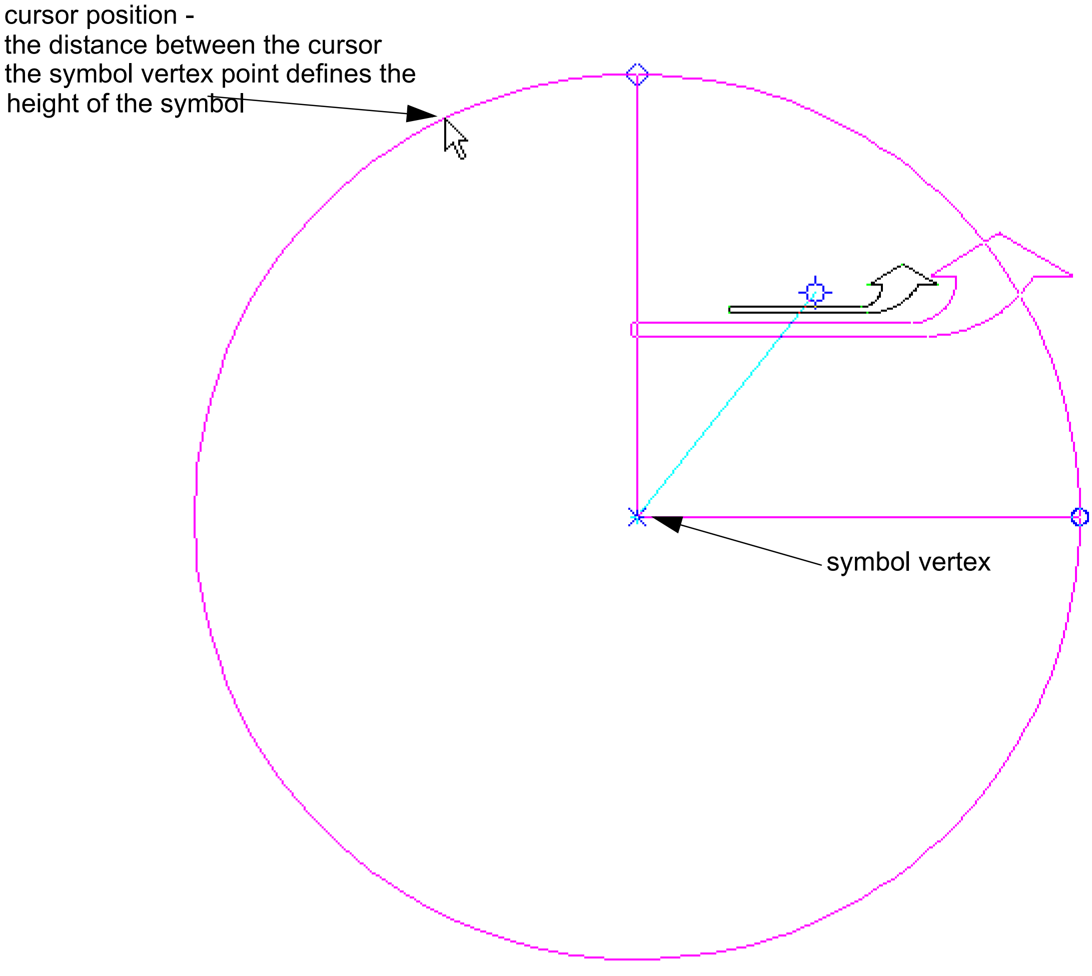
In J mode: if the ***Height* grip** is selected, a circle is drawn from the symbol justification point to the Height grip and this represents the height of the symbol. As the cursor is moved, the distance from the cursor to the symbol justification point is taken as the circle radius and the symbol height.



Modifying Height - J Mode

V Mode

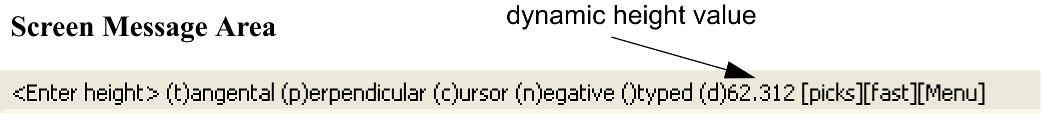
In V mode: if the ***Height* grip** is selected, a circle is drawn from the symbol vertex to the Height grip and this represents the height of the symbol. As the cursor is moved, the distance from the cursor to the symbol vertex is taken as the circle radius and the symbol height.



Modifying Height - V Mode

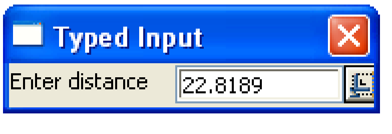
In J and V mode, the symbol height is dynamically displayed as part of the *Enter height* prompt in the screen message area

Screen Message Area



The symbol height is set when the cursor position is accepted.

When the symbol is being dynamically sized, pressing the **d** key brings up the **Enter distance** typed input box with the current **dynamic** symbol height in it.



An ***exact*** symbol height can be typed into the ***Enter distance*** **Typed Input** box and accepted by pressing the <enter key>.

If instead of typing **d,** the <space bar> is pressed, the ***Enter distance*** **Typed Input** box is displayed with a space in it. The required height can be typed into the box and accepted by pressing the <enter key>.

If instead of typing **d,** any character other than **t**, **p**, **c** or **n** is typed (for example the beginning of the exact height required), the ***Enter distance*** **Typed Input** box is displayed with the typed characters in it. The height is accepted by pressing the <enter key>.

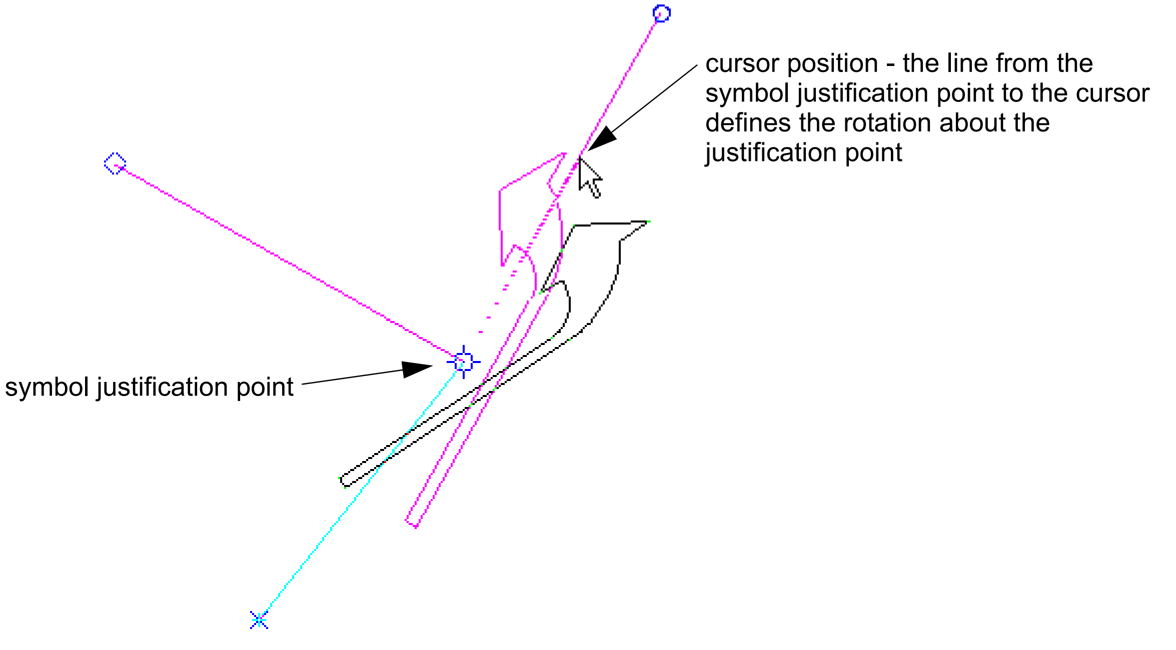
For more information on typing **t**, **p**, **c** or **n**, go to the section 4.14.7 Typing t, p, c or n

To return to dynamic sizing without entering a distance into the ***Enter distance*** input box, simply select the **X** on the top of the input box. The box will disappear and dynamic sizing will resume.

4.15.2 **Rotate Grip (circle)**:

J Mode

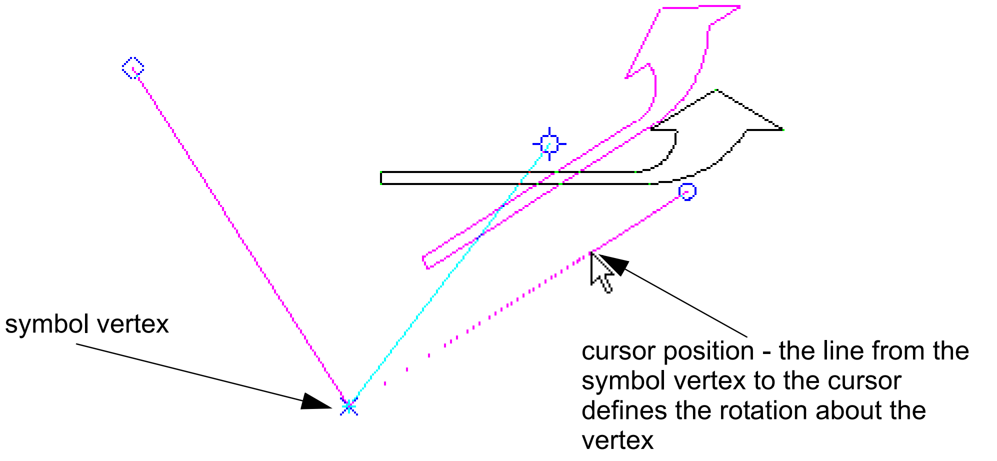
In J mode: if the **Rotate grip** is selected, a line from the symbol justification point to the cursor position is taken as the symbol angle. As the cursor is moved, the angle from the cursor to the symbol justification point defines the symbol angle.



Modifying Angle - J Mode

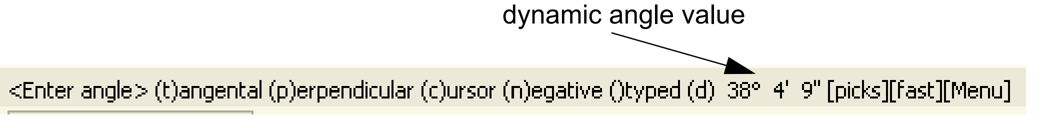
V Mode

In V mode: if the **Rotate grip** is selected, a line from the symbol vertex to the cursor position is taken as the symbol angle. As the cursor is moved, the angle from the cursor to the symbol vertex defines the symbol angle.



Modifying Angle - V Mode

In J and V mode, the angle of the symbol is displayed as part of the *Enter angle* prompt in the screen message area.



The symbol angle is set when the cursor position is accepted.

When the angle is being dynamically changed, pressing the **d** key brings up the **Enter angle** typed input box with the current **dynamic** symbol angle in it.

An ***exact*** symbol angle can be typed into the ***Enter angle*** **Typed Input** box and accepted by pressing the <enter key>.

If instead of typing **d,** the <space bar> is pressed (shown as **()**), the ***Enter angle*** **Typed Input** box is displayed with a space in it. The required angle can be typed into the box and accepted by pressing the <enter key>.

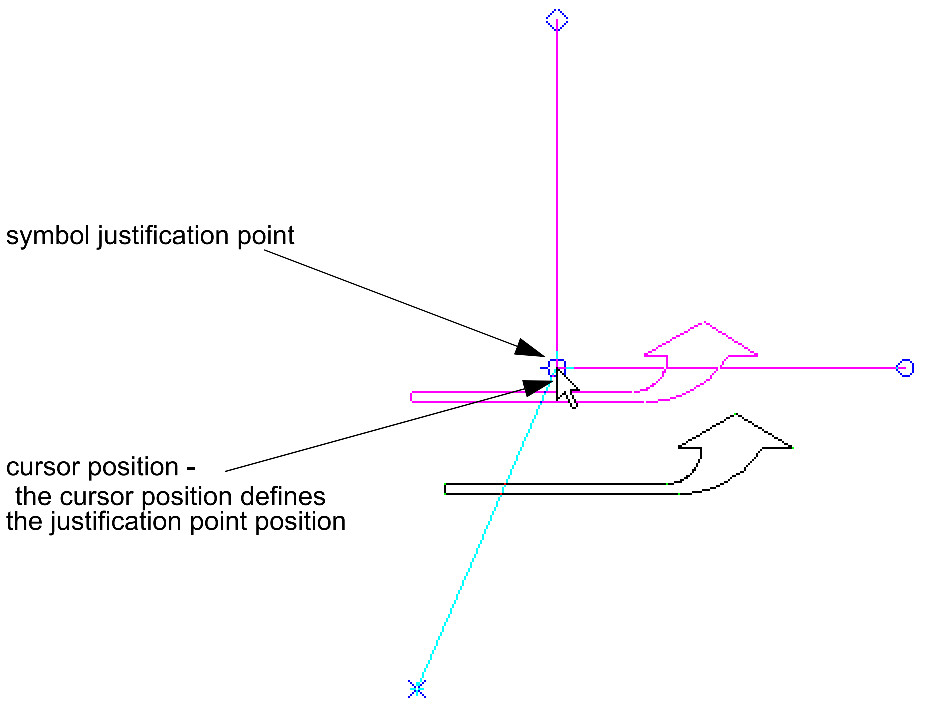
If instead of typing **d,** any character other than **t**, **p**, **c** or **n** is typed (for example the beginning of the exact angle required), the ***Enter angle*** **Typed Input** box is displayed with the typed characters in it. The angle is accepted by pressing the <enter key>.

For more information on typing **t**, **p**, **c** or **n**, go to the section 4.14.7 Typing t, p, c or n

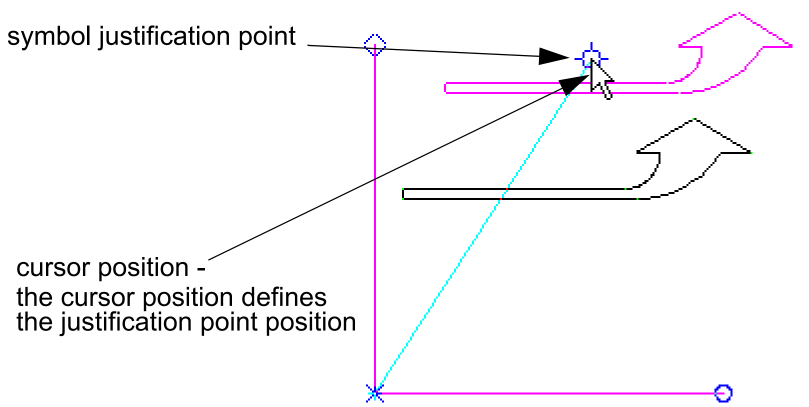
To return to dynamic angling without entering an angle into the ***Enter angle*** input box, simply select the **X** on the top of the input box. The box will disappear and dynamic angling will resume.

4.15.3 **Justification Grip (square)**:

In J and V Mode: if the **Justification grip** is selected, the cursor position is taken as the symbol justification point. As the cursor is moved, the symbol justification point moves with it.

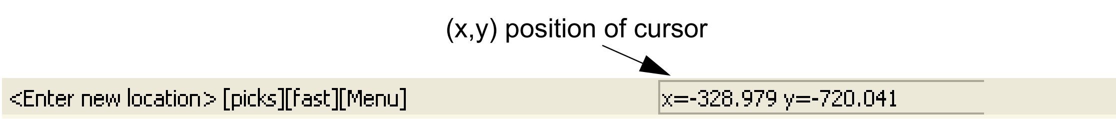


Moving Symbol Justification Point - J Mode



Moving Symbol Justification Point - V Mode

In J and V mode, the (x,y) position of the symbol is displayed in the screen message area.



The symbol justification point is set when the cursor position is accepted.

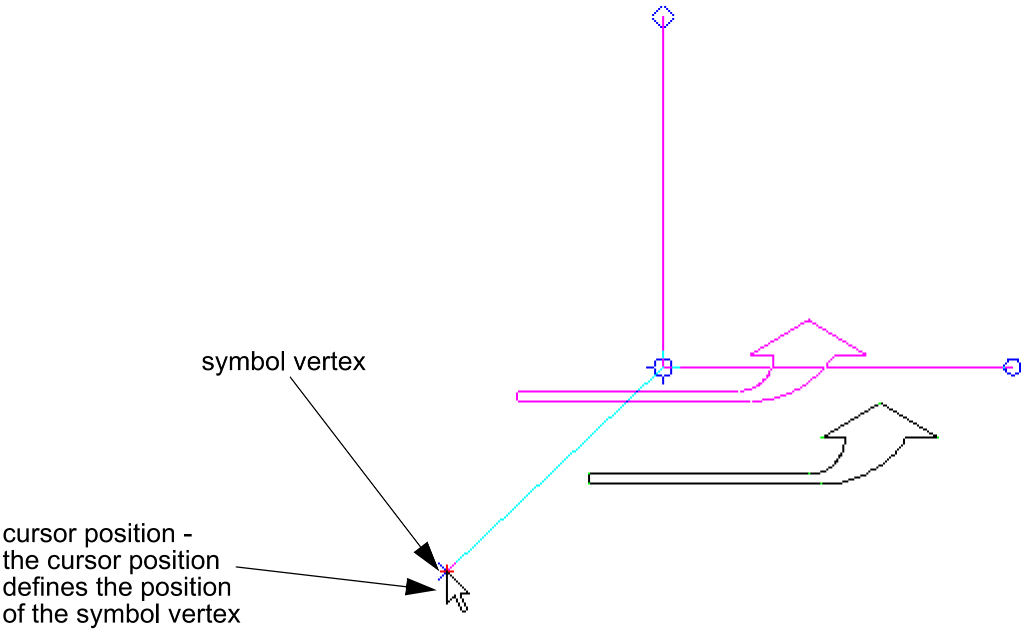
An ***exact*** coordinate can be entered by hitting the <space bar> or by starting to type a coordinate. This brings up the **Enter X Y Z** typed input box. The *x and y coordinates* are typed in, separated by a space, and the <enter> key pressed.

To return to dynamic moving without entering a coordinate, simply select the **X** on the top of the input box. The box will disappear and dynamic moving will resume.

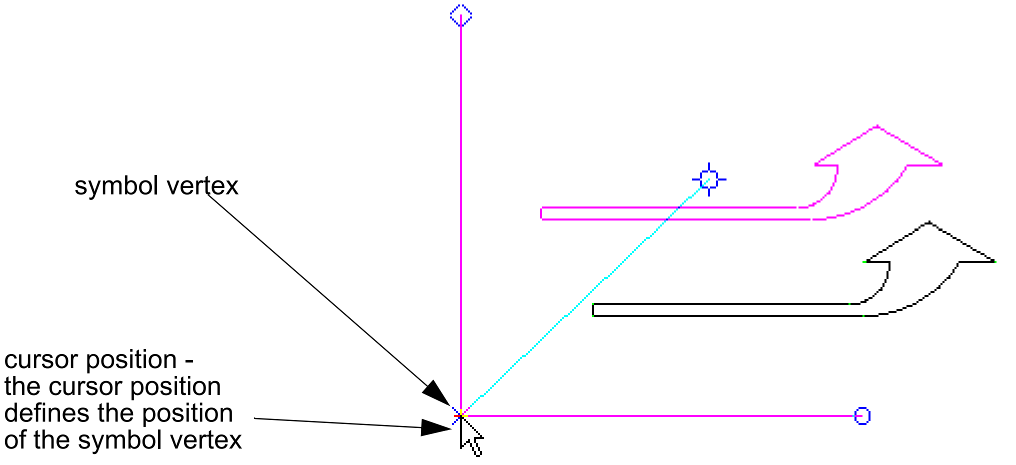
4.15.4 Vertex Grip (X):

In J and V mode: if the **Vertex grip** is selected, the cursor position is taken as the symbol vertex, As the cursor is moved, the symbol vertex moves with it.

Note - the Vertex can **not** be selected when it is in ***locked*** mode.

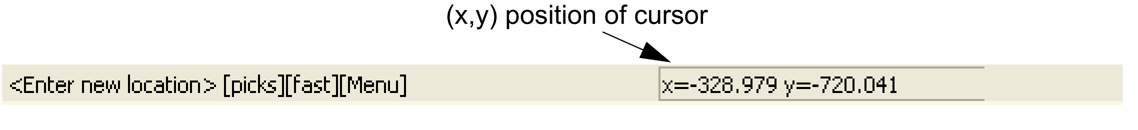


Moving Symbol Vertex - J Mode



Moving Symbol Vertex - V Mode

In both J and V mode, the (x,y) position of the symbol is displayed in the screen message area.



The symbol vertex is set when the cursor position is accepted.

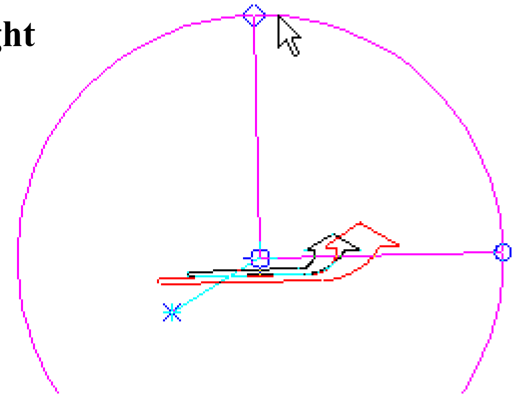
An ***exact*** coordinate can be entered by hitting the <space bar> or by starting to type a coordinate. This brings up the **Enter X Y Z** typed input box. The *x and y coordinates* are typed in, separated by a space, and the <enter> key pressed.

To return to dynamic moving without entering a coordinate, simply select the **X** on the top of the input box. The box will disappear and dynamic moving will resume.

4.15.5 Typing t, p, c or n

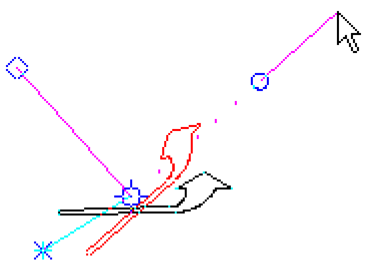
After selecting the ***Height*** grip (diamond) or the ***Rotate*** grip (circle), the screen message area shows that there are the typed option **t**, **p**, **c**, **n**, **()** and **d** available.

Modifying Height



Screen Message Area

<Enter height> (t)angental (p)erpendicular (c)ursor (n)egative ()typed (d)35.467 [picks][fast][Menu]



Modifying Angle

Screen Message Area

<Enter angle> (t)angental (p)erpendicular (c)ursor (n)egative ()typed (d) 38° 4' 9" [picks][fast][Menu]

Type **t**: snap cursor tangential to a selected string - go to Snap Tangential - Typing t

Type **p**: snap cursor perpendicular to a selected string - go to Snap Perpendicular - Typing p

Type **c**: return to cursor movement - go to Cursor Mode - Typing c

Type **n**: return to cursor movement - go to Negative Angle - Typing n

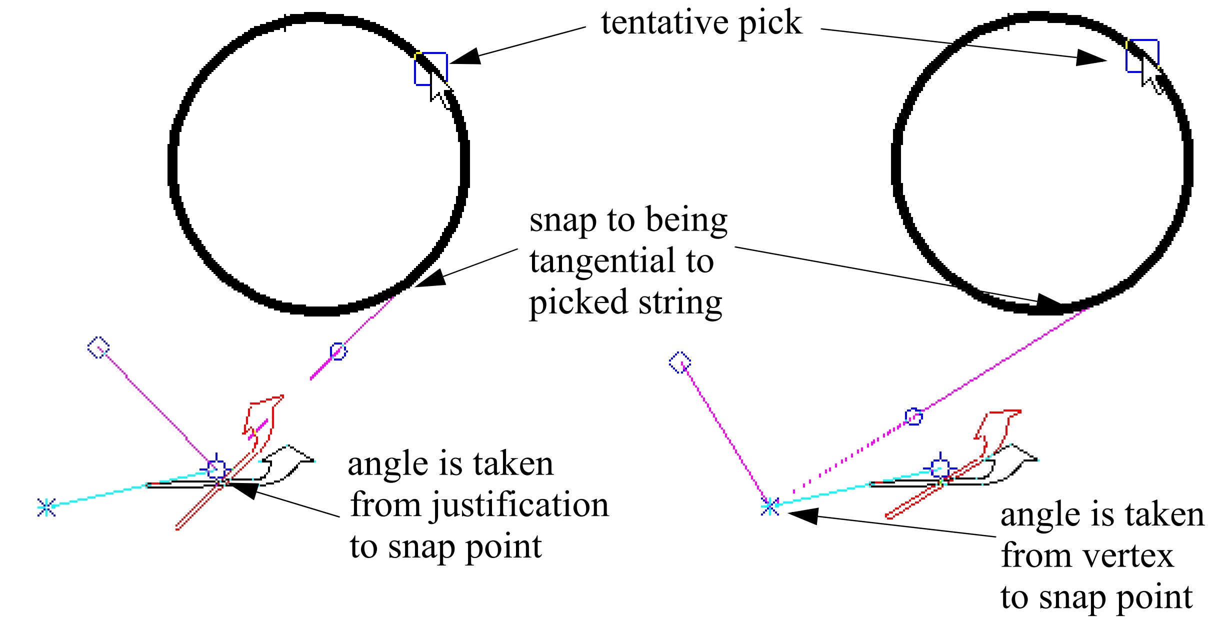
Type <space bar>: bring up typed input box for value - go to Hitting Space Bar

Type **d**: bring up dynamic value - go to Display Dynamic Value - Typing d

Snap Tangential - Typing t

Snap Tangential - Rotation Grip (Circle)

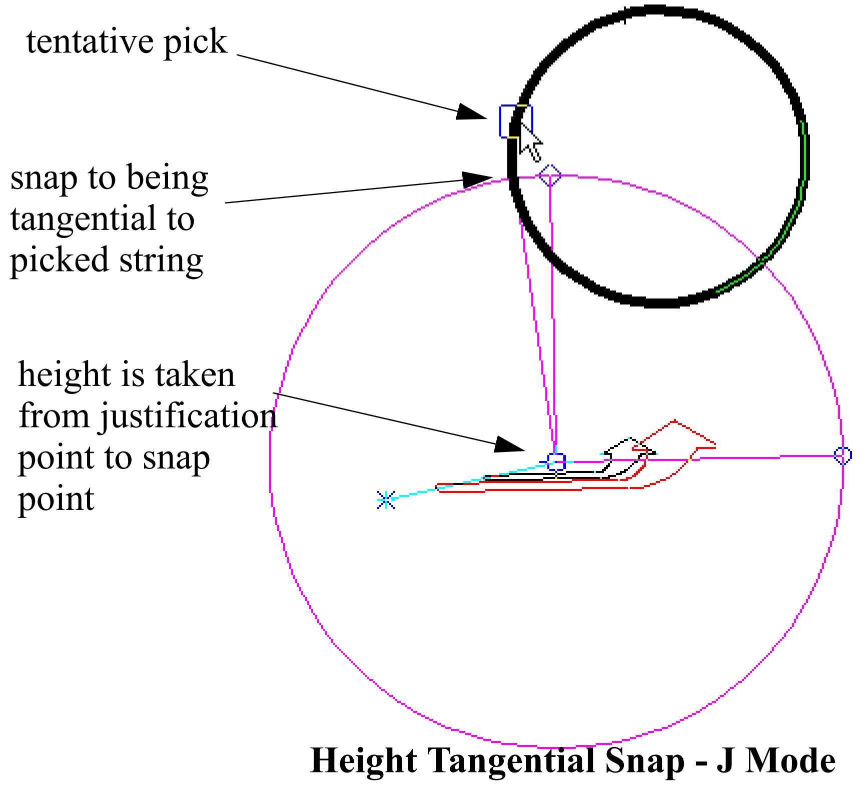
After selecting the ***Rotation*** grip (circle), typing **t** puts the string select into tangential mode and when a string is tentatively picked, the snap point moves to make the line from the snap point to the symbol justification point in J mode (or to the vertex in V-mode) ***tangential*** to the selected string.



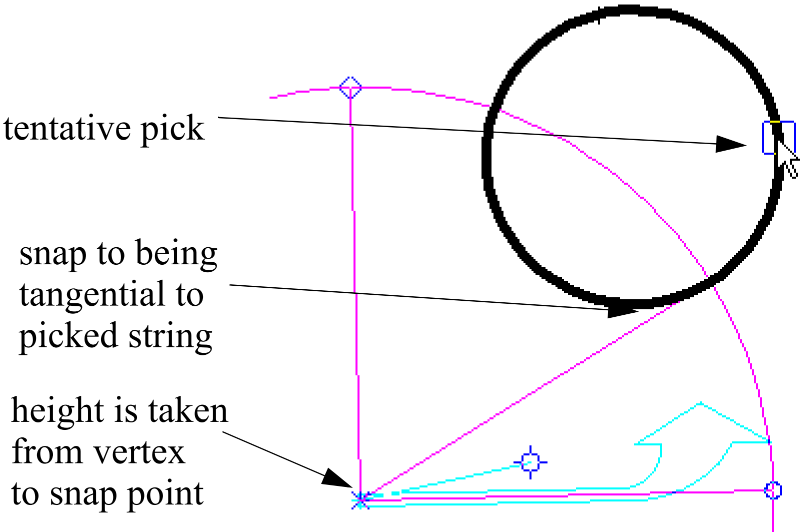
Rotation Tangential Snap - J Mode **Rotation Tangential Snap - V Mode**

Snap Tangential - Height Grip (Square)

After selecting the ***Height*** grip (diamond), typing **t** puts the string select into tangential mode and when a string is tentatively picked, the snap point moves to make the line from the snap point to the symbol justification point (J mode)/vertex (V-mode) tangential to the selected string.



Height Tangential Snap - J Mode



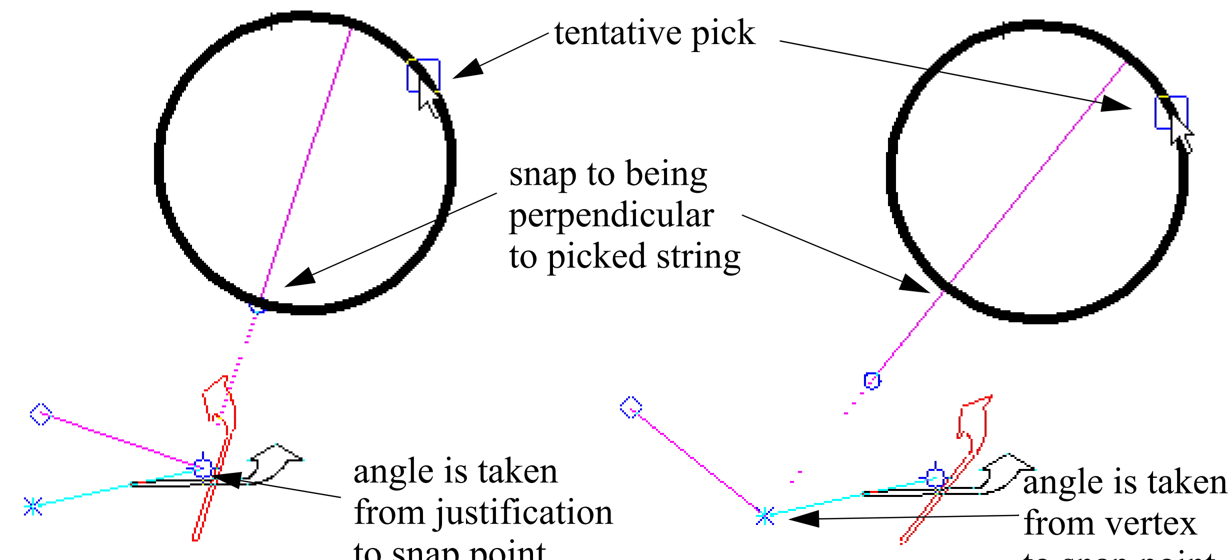
Height Tangential Snap - V Mode

Before the accept button is selected, **p** can be typed to toggle to perpendicular snap mode, **c** to return to using the cursor position or **d** to bring up the dynamic value.

Snap Perpendicular - Typing p

Snap Perpendicular - Rotation grip (Circle)

After selecting the ***Rotation*** grip (circle), typing **p** puts the string select into perpendicular mode and when a string is tentatively picked, the snap point moves to make the line from the snap point to the symbol justification point in J mode (or the vertex in V-mode) perpendicular to the selected string.

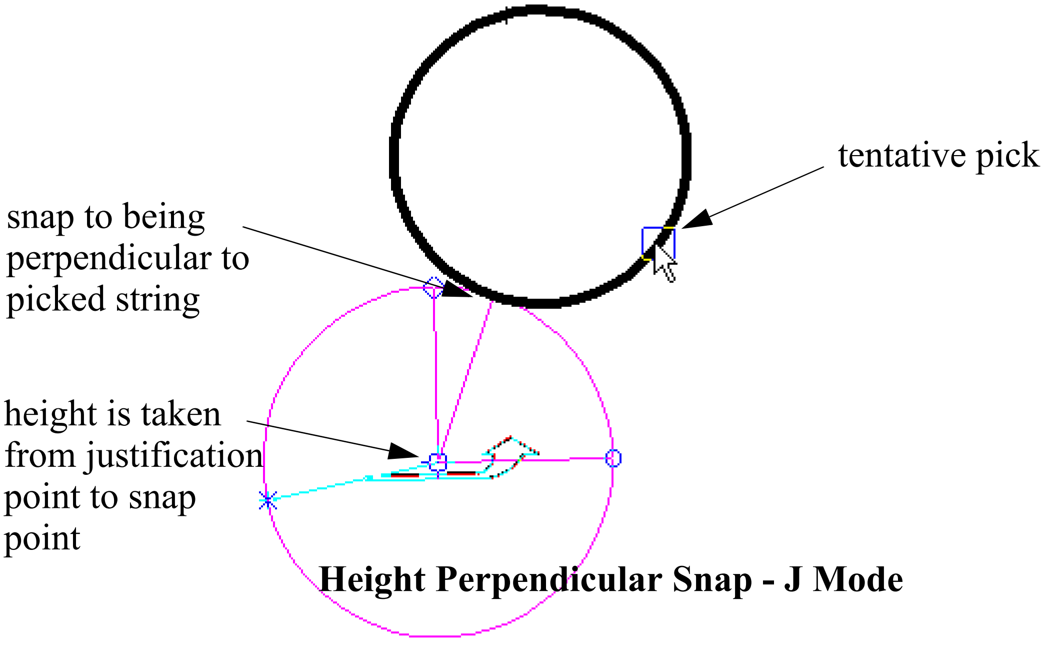


Rotation Perpendicular Snap J Mode

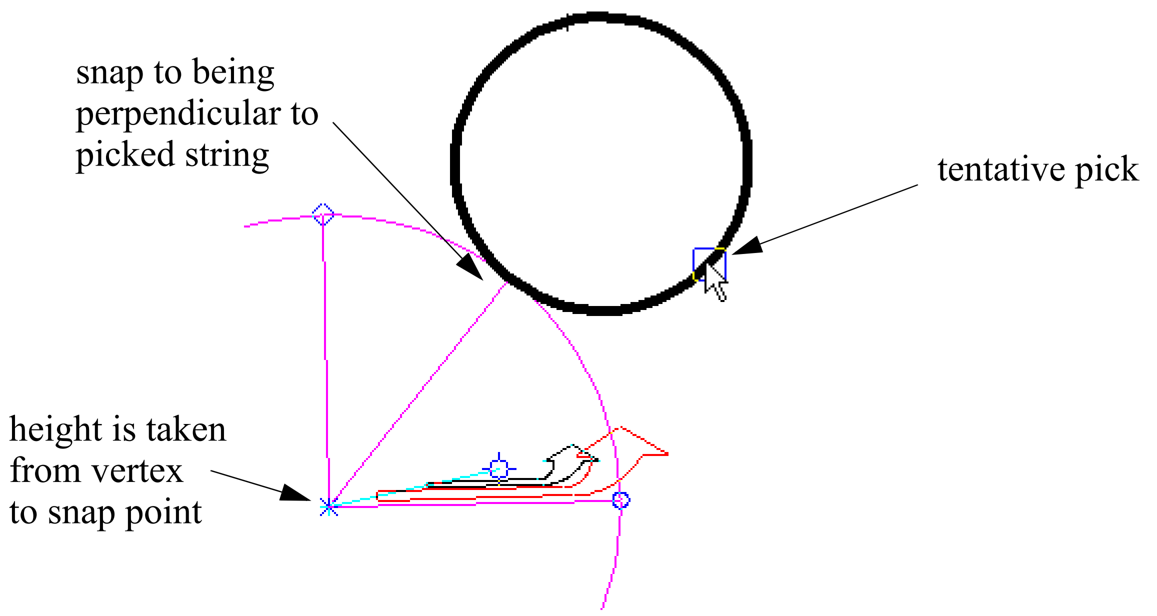
Rotation Perpendicular Snap V Mode

Snap Perpendicular - Height Grip (Square)

After selecting the ***Height*** grip (diamond), typing **p** puts the string select into perpendicular mode and when a string is tentatively picked, the snap point moves to make the line from the snap point to the symbol justification point in J mode (or vertex in V-mode) perpendicular to the selected string.



Height Perpendicular Snap - J Mode



Height Perpendicular Snap - V Mode

Before the accept button is selected, **t** can be typed to toggle to tangential snap mode, **c** to return to using the cursor position or **d** to bring up the dynamic value.

Cursor Mode - Typing c

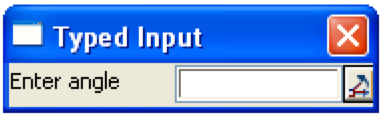
If in either Tangential or Perpendicular mode, typing **c** return to cursor mode.

Negative Angle - Typing n

Adds 180 degrees to the current angle.

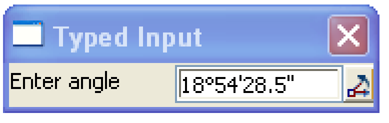
Hitting Space Bar

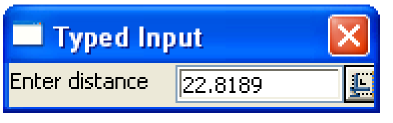
Brings up the **Enter angle** (modifying angle) or **Enter distance** (modifying height) **Typed input** box with no value in it.



Display Dynamic Value - Typing d

Brings up the **Enter angle** (modifying angle) or **Enter distance** (modifying height) **Typed input** box with the *dynamic* value in it.





Please continue to the next section 4.19.1 Expressions in Panel Fields.

4.16 Last Expression

Brings up the last expression or real value typed into the box. For example, if 3*4 is typed and <Enter> pressed, 12 will be displayed. The **Last Expression** pop-up will bring back the 3*4.

Please continue to the next section 4.17 Bearings and Angles.

4.17 Bearings and Angles

4.17.1 HP Notation

Many panels and options take angles or bearings as input. The 12d Model default for most angles or bearings is in degrees, minutes and second (dms).

To save typing, this is written in ***HP Notation*** in the form

ddd.mmssfff

where **ddd.mmssfff** is the angle or bearing expressed in degrees, minutes and seconds as

ddd	whole degrees
.	separator between degrees and minutes
mm	whole minutes
ss	whole seconds
fff	fractions of seconds (as many as required)

Notes

1. The decimal point **.** indicates where the degrees stops and the minutes begin.
2. If there is only whole degrees, the **.mmssfff** can be omitted. However, if there are any minutes, seconds or fractions of seconds, there must be **two digits** of whole minutes.
3. If there are no seconds or fractions of seconds, then the **ssfff** can be omitted. However, if there are any seconds or fractions of seconds, there must be **two digits** of whole seconds.
4. There are as many digits of fractions of seconds as required (possibly none).

For example

35	is 35 degrees
35.09	is 35 degrees and nine minutes
35.0901	is 35 degrees, nine minutes and one second
35.090107	is 35 degrees, nine minutes and 1.07 seconds

4.17.2 Decimal Degrees for Angles

At times, angles or bearings are given as decimal degrees rather than in 4.17.1 HP Notation.

For decimal degrees, positions after the decimal point represent decimal degrees.

Please continue to the next section 4.18 Precision.

4.18 Precision

Double precision variables are used throughout 12d Model for all coordinate values and calculations.

Although this increases the time taken for calculations, it is more than offset by the increase in the accuracy of the data and the results of any calculations.

Because of the accuracy of double precision variables (fourteen significant figures), there is no practical restriction on the coordinate area covered by projects.

The default number of decimal places for displaying values in the *information menu* is three (3).

This number of decimal places is controlled by the **display precision** field in the **System Settings** tab of the **Defaults** panel (option **Utilities=>Defaults**).

The default number of decimal places for displaying values in *boxes* and *panels* is four(4).

This number of decimal places is controlled by the **box precision** field in the **System Settings** tab of the **Defaults** panel (option **Utilities=>Defaults**).

Go to the next section 4.19 Panel Fields.

4.19 Panel Fields

For general information about Panels, *see* 4.3.6 Panels.

For more particular information, see

4.19.1 Expressions in Panel Fields

See

4.19.1.1 Expressions in Real Value Fields
4.19.1.2 Expressions in Bearing, Angle Panel Fields
4.19.1.3 Expressions in Integer Value Fields

4.19.1.1 Expressions in Real Value Fields

See

4.19.1.1.1 Mathematical Expressions in Real Value Fields
4.19.1.1.2 If Else in Real Value Fields

4.19.1.1.1 Mathematical Expressions in Real Value Fields

A **Real** panel field is a panel field that can only take real numbers. That is, numbers that have a decimal place. For example 0.1, 1 and -2.3.

Whenever a ***real value*** such as height or width is required in a **Real** panel field other **than an Angle or Bearing panel field**, mathematical expressions can be typed in and then evaluated by ***12d Model*** when <enter> is pressed. See the section 4.19.1.2 Expressions in Bearing, Angle Panel Fields for what is allowed in angle/bearing panel fields.

Expressions can be made up from the operators

*	multiply
/	divide
+	addition
-	subtraction

where * and / take precedence over + and -.

Nested brackets "(" and ")" are supported to any level.

The following functions are also supported

sin
cos
tan
sqrt
square
null

Note that for the trigonometric functions, the angle is in **degrees, minutes and seconds** (see section on Angles and Bearings) and is either a cartesian angle or a bearing depending on the setting of the **Angle mode** in the *System Settings* tab of **Utilities=>Defaults.**

For example, cos(90) is 0 for a Cartesian angle of 90 or 1 for a bearing of 90 degrees.

Examples of expressions are:

10.0 + 19.7

sin(90.30)

(10 + sin(45))/3.0

After the expression has been evaluated, it is still possible to bring back the last expressions, make modifications to the expression and then re-evaluate it.

To get the last expression, simply click on the **[+]** button at the end of the field to bring up the **measures** menu and select **Last expression** from it. The last expression is then returned to the panel field ready for modifications.

Warning

Only limited expressions are supported for bearing/angle boxes. See the section 4.19.1.2

Expressions in Bearing, Angle Panel Fields for what is allowed in the Angle and Bearing panel fields.

Go to the next section 4.19.1.2 Expressions in Bearing, Angle Panel Fields or return to 4.19.1 Expressions in Panel Fields.

4.19.1.1.2 If Else in Real Value Fields

There is an **if else** conditional expression that can be used in **Real** fields:

if *(logical_expression) value_1* **else** *value_2*

which has the meaning

if *logical_expression* is true then use *value_1* otherwise use *value_2*.

where ***value_1*** and ***value_2*** are expressions that evaluate to a real value.

One restriction is that the l***ogical_expression*** must be enclosed in round brackets.

For example,

if (3*4<13) 3 else 4

This may not seem to be of much use except that it works nicely with Snippets.

For example, you can then have in a Real field

if ($VAL1< 0) -1*VAL1 else $VAL1

where VAL1 is a snippet parameter.

For more information on snippets, see 21.5 Defining and Using Snippets.

Important Note

Note that *value_1* or *value_2* can be an **if else** conditional expressions so you can have **nested conditional expressions**. That is

if *(logical_expression) conditional_expression_1* **else** *conditional_expression_2*

For example

if ($VAL1 < 0) if (3*4<13) 3 else 4 else $VAL1

will be par sed as

if ($VAL1 < 0) (if (3*4<13) 3 else 4) else $VAL1

Hint: if something doesn’t seem to work, put more round brackets in.

Go to the next section 4.19.1.2 Expressions in Bearing, Angle Panel Fields or return to 4.19.1.1 Expressions in Real Value Fields or 4.19.1 Expressions in Panel Fields.

4.19.1.2 Expressions in Bearing, Angle Panel Fields

Bearing and **Angle** panel fields are panel field that can only take angles.

Because bearings (angles) and numbers **can not** be used interchangeably in mathematical expressions, only a limited number of expressions can be supported in a **Bearing** or **Angle** panel field.

Bearings (angles) can be added or subtracted from other bearings (angles) but a number can not be added or subtracted from a bearing (angle). So the addition and subtraction of bearings (angles) is supported but addition/subtraction of a bearing (angle) and a number is invalid.

Bearings (angles) can not be multiplied or divided by another bearing (angle) but a bearing (angle) can be multiplied or divided by a number. However because the input of a bearing (angle) in degrees, minutes and seconds looks like a decimal number (see the previous section 4.17 Bearings and Angles), a rule must be made to distinguish between bearings (angles) and numbers for multiplication and division. In ***12d Model***, the bearing (angle) is on the left of the number it is being multiple or divided by.

So in a bearing (angle) panel field and grid, the following is supported:

(a) add or subtract bearings (angles) i.e. bearing + bearing or bearing - bearing is allowed

(b) type bearing*number + bearing *number - note that the number is on the right of the *

(c) type bearing *number - bearing *number - note that the number is on the right of the *

(d) type bearing /number + bearing *number - note that the number is on the right side of the /

(e) type bearing /number + bearing *number - note that the number is on the right side of the /

Because a user often wishes to add or subtract 90 degrees from a bearing or angle, the Page Down/Page Up keys have a special meaning when the focus in on the bearing (angle) panel field. For a bearing (or angle) panel field (but ***not for a grid***):

(f)	Page up	- subtracts 90 degrees to the bearing/angle
(g)	Ctrl + page up	- subtracts 15 degrees to the bearing/angle
(h)	Shift + page up	- subtracts 10 degrees to the bearing/angle
(i)	Shift + Ctrl + page up	- subtracts 6 degrees to the bearing/angle
(j)	Page down	- adds 90 degrees to the bearing/angle
(k)	Ctrl + page down	- adds 15 degrees to the bearing/angle
(l)	Shift + page down	- adds 10 degrees to the bearing/angle
(m)	Shift + Ctrl + page down	- adds 6 degrees to the bearing/angle

Go to the next section 4.19.1.3 Expressions in Integer Value Fields or return to 4.19.1 Expressions in Panel Fields.

4.19.1.3 Expressions in Integer Value Fields

An **Integer** panel field is a panel field that can only take and integer number. For example 0, 1 and -2.

The arithmetic operations plus (+), subtraction (-), divide (/) and multiplication (*) are now allowed in an **Integer** panel field.

The calculations are all done as real arithmetic and the result is rounded to an integer value at the **end** of all the calculations and not along the way.

Go to the next section 4.19.2 Pre*Postfix in Panel Fields or return to 4.19.1 Expressions in Panel Fields.

4.19.2 Pre*Postfix in Panel Fields

In many options in ***12d Model***, text needs to added to the beginning (prefix) of name and also added to the end (postfix) of a name. For example, the

Instead of needing a panel field for the prefixed text and another for the postfixed text, ***12d Model*** often used a special shorthand notation called pre*postfix to combine the two into one text field.

When the word ***pre*postfix*** is used on a panel field, it has the special meaning that for any text type into the panel field, any text before the * is considered to be text for prefixing, and any text after the ***** is taken to be text for postfixing. Note that spaces are significant.

Hence "E ***** m" means that "E " is prefix text and " m" is postfix text.

If prefix text only is required, just give the text since the * is not required at the end of the text. For example "E " or "E *" will prefix the text "E ".

If postfix text only is required then the text must be preceded by a *. For example "* m" will postfix the text " m".

For example, if a panel field had

Pre*postfix for models	new * data

then "new " would be added to the beginning of the model names and " data" would be added to the end of the model names.

Go to the next section 4.19.3 Data Source or return to 4.19 Panel Fields.

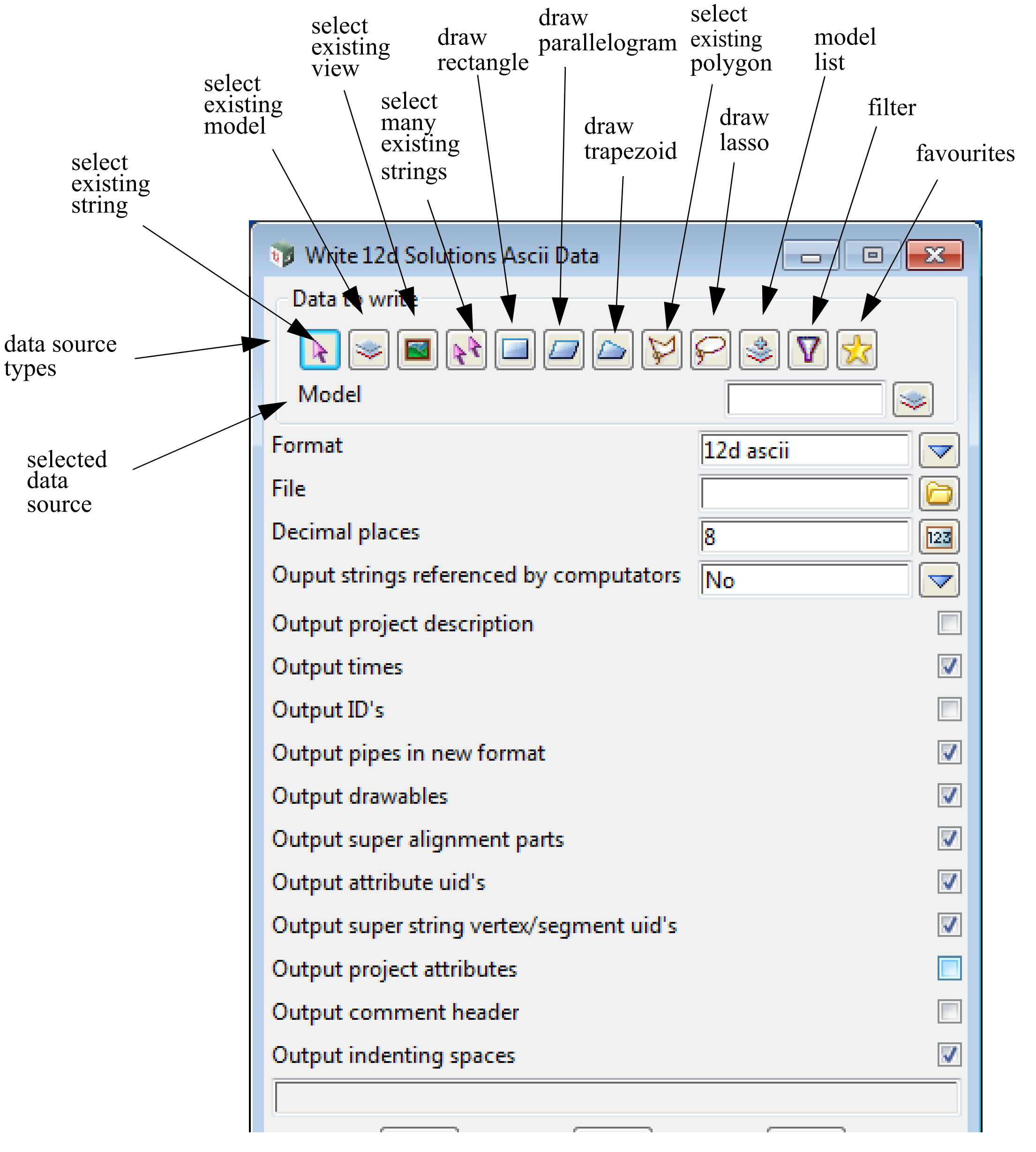
4.19.3 Data Source

Many options contain the panel field **Data source** which may have one or more of the selection choices:

string, model, view, multi-pick strings, rectangle, parallelogram, polygon, lasso, model list, filter, favourites

Depending on the Data Source choice, the panel fields after the Data Source icons will be changed to suite the Data Source choice.

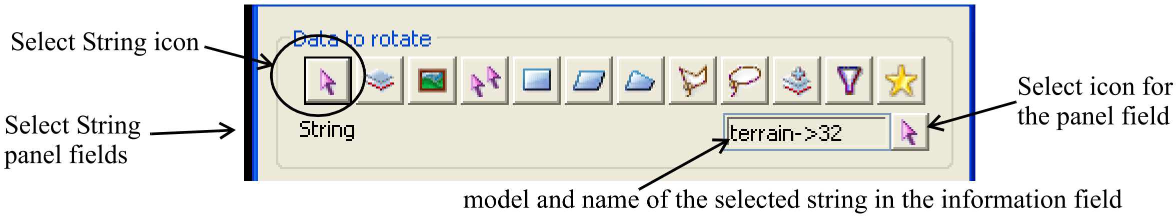
For example, for the choice **Model**, the next field will be **Model**.



See 4.19.3.1 Select Existing String
See 4.19.3.2 Select Existing Model
See 4.19.3.3 Select Existing View
See 4.19.3.4 Select Many Strings
See 4.19.3.5 Select Using a User Drawn Rectangle
See 4.19.3.6 Select Using a User Drawn Parallelogram

See 4.19.3.7 Select Using a User Drawn Trapezoid
See 4.19.3.8 Select an Existing Polygon
See 4.19.3.9 Select Using a User Drawn Lasso
See 4.19.3.10 Select many models
See 4.19.3.11 Select Filter
See 4.19.3.12 Favourites

4.19.3.1 Select Existing String



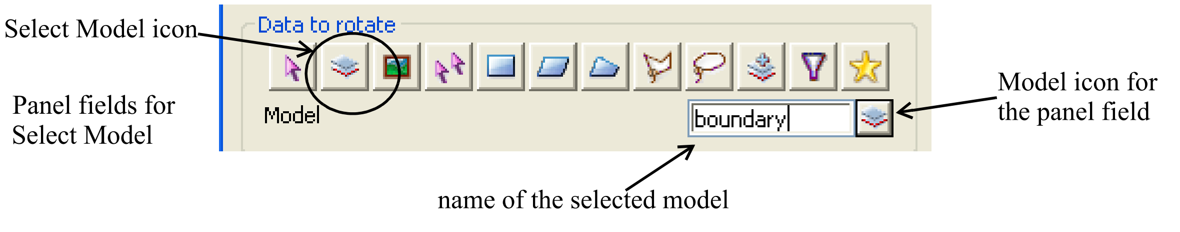
*After clicking on the **Select String** icon, a String Select is started and the Select String panel field called **String** is placed under the Data Source icons.*

When a string is selected, its model and name of the string is displayed in the String Select information field.

To selected a different string, the Select icon for the String Select panel field (to the right of the string name field) must be used as the Data Source Select String icon is now inactive.

Continue to 4.19.3.2 Select Existing Model or return to 4.19.3 Data Source

4.19.3.2 Select Existing Model



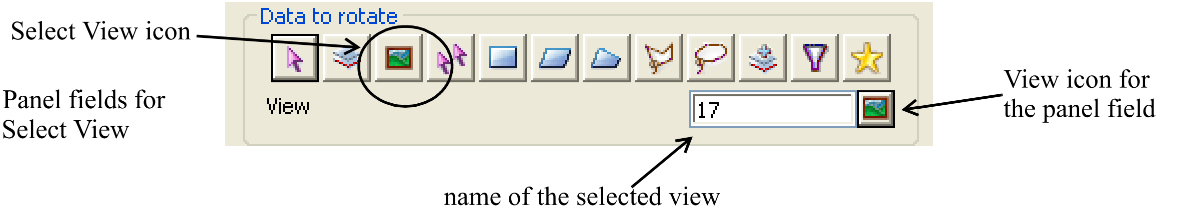
*After clicking on the **Select Model** icon, a Model panel field is placed under the Data Source icons.*

A model name can be typed into the panel field or a model selected from the Model icon from the Model panel field (to the right of the model name field).

To select a different model, the Model icon for the Model panel field must be used as the Data Source Select Model icon is now inactive.

Continue to 4.19.3.3 Select Existing View or return to 4.19.3 Data Source

4.19.3.3 Select Existing View



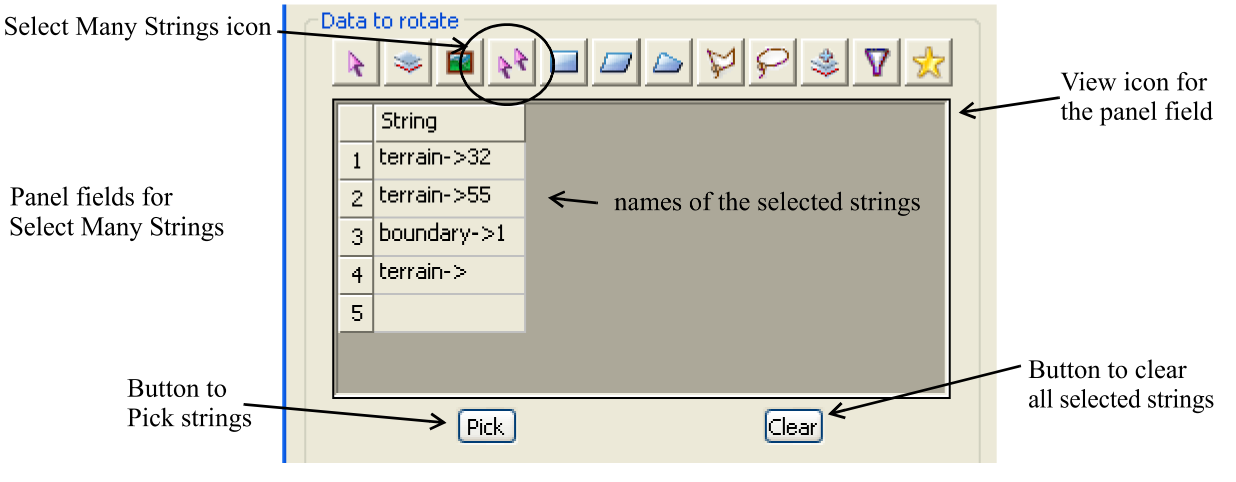
After clicking on the ***Select View*** *icon, a View panel field is placed under the Data Source icons.*

A view name can be typed into the panel field or a view selected from the View icon from the View panel field (to the right of the view name field).

To select a different view, the View icon for the View panel field must be used as the Data Source Select View icon is now inactive.

Continue to 4.19.3.4 Select Many Strings or return to 4.19.3 Data Source

4.19.3.4 Select Many Strings



After clicking on the ***Select Many Strings*** *icon, the panel fields for the Select Many Strings are placed under the Data Source icons.*

The Select Many Strings panel fields consists of a grid to display the model and string name of selected strings, a Pick button to start selecting strings and a Clear button to deselect the currently selected strings.

To begin selecting strings, click on the ***Pick*** *button. Strings are then selected and accepted and this continues until RB is pushed and* ***Cancel*** *selected from the* ***Pick Ops*** *menu. As each string is selected it is highlighted and the selected strings stay highlighted after they are selected.*

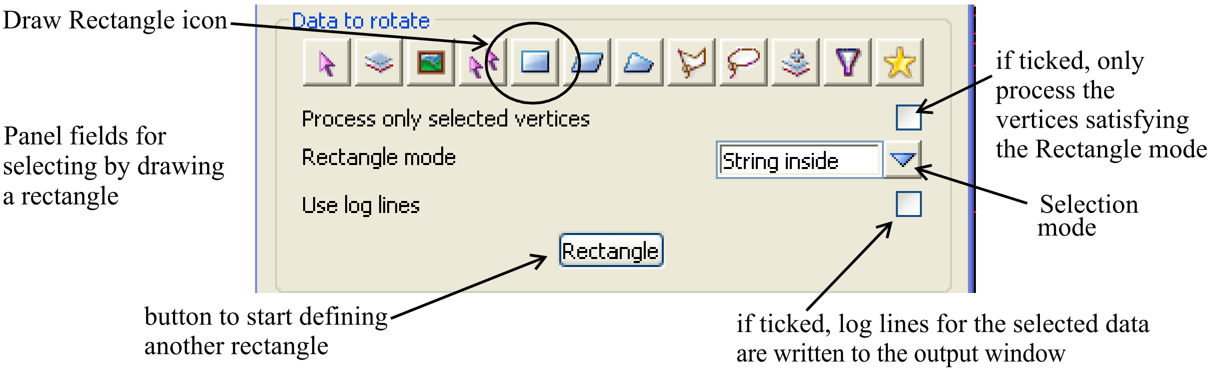
To pick ***more*** *strings, simply click on* ***Pick*** *and start selecting again. The new strings are added to the bottom of the String name grid and all the new strings also remain highlighted.*

To clear all the selected string, simply click on ***Clear.***

The Data Source Select Many Strings icon is made inactive after the option is first selected.

Continue to 4.19.3.5 Select Using a User Drawn Rectangle or return to 4.19.3 Data Source

4.19.3.5 Select Using a User Drawn Rectangle



*After clicking on the **Draw Rectangle** icon, the panels fields for selecting by drawing a rectangle are placed under the Data Source icons and the rectangle drawing process is automatically started.*

First the position of one corner of the rectangle is selected by clicking LB on a plan view and then a rectangle parallel to the x and y axis is drawn to the current cursor position. The rectangle is completed by clicking LB at a second position.

Strings are selected by using the rectangle and the Rectangle mode which is:

*if **Process only selected vertices** is ticked on, **Rectangle mode** choices are:*

Vertices inside or Vertices outside

*if **Process only selected vertices** is not ticked, **Rectangle mode** choices are:*

String inside, String outside, String crossing, String inside/crossing, String outside/crossing

*Any strings/vertices satisfying the **Rectangle mode** will be highlighted when the rectangle is completed.*

*If **Process only selected vertices** or **Rectangle mode** is modified, the new strings/vertices satisfying he new parameters are highlighted.*

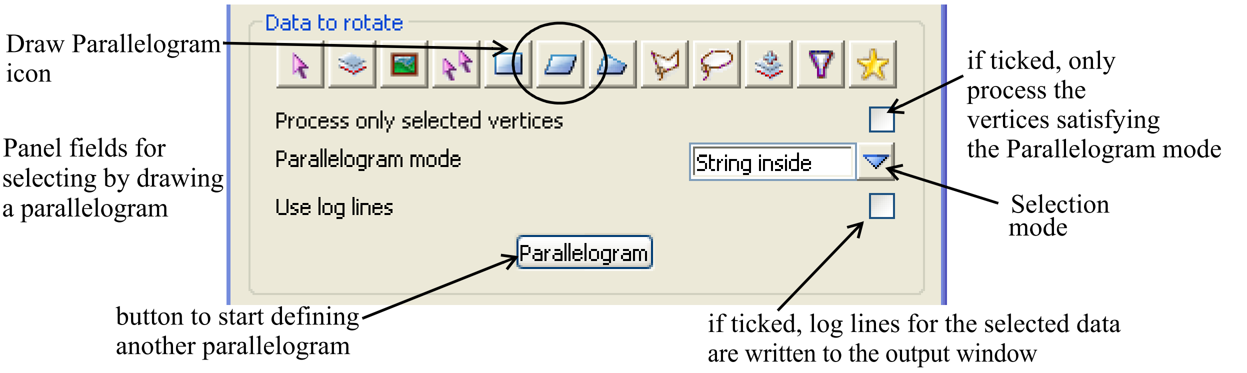
*If **Use log lines** is ticked, log lines for the selected data are written to the output window.*

*To draw a new rectangle, click on the **Rectangle** button and draw a new rectangle.*

The Data Source Draw Rectangle icon is made inactive after the option is first selected.

Continue to 4.19.3.6 Select Using a User Drawn Parallelogram or return to 4.19.3 Data Source

4.19.3.6 Select Using a User Drawn Parallelogram



*After clicking on the **Draw Parallelogram** icon, the panels fields for selecting by drawing a parallelogram are placed under the Data Source icons and the parallelogram drawing process is automatically started.*

First two positions defining ne side of the parallelogram by clicking LB at each position. A parallelogram is then drawn to the current cursor position. The parallelogram is completed by clicking LB at a third position.

Strings are selected by using the parallelogram and the Parallelogram mode which is:

*if **Process only selected vertices** is ticked on, **Parallelogram mode** choices are:*

Vertices inside or Vertices outside

*if **Process only selected vertices** is not ticked, **Parallelogram mode** choices are:*

String inside, String outside, String crossing, String inside/crossing, String outside/crossing

*Any strings/vertices satisfying the **Parallelogram mode** will be highlighted when the parallelogram is completed.*

*If **Process only selected vertices** or **Parallelogram mode** is modified, the new strings/vertices satisfying he new parameters are highlighted.*

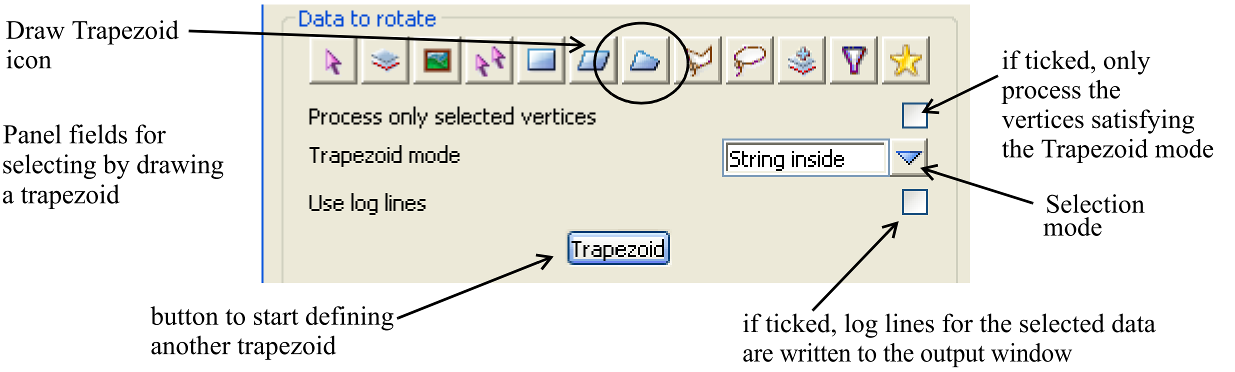
*If **Use log lines** is ticked, log lines for the selected data are written to the output window.*

*To draw a new parallelogram, click on the **Parallelogram** button and draw a new parallelogram.*

The Data Source Draw Parallelogram icon is made inactive after the option is first selected.

Continue to 4.19.3.7 Select Using a User Drawn Trapezoid or return to 4.19.3 Data Source

4.19.3.7 Select Using a User Drawn Trapezoid



*After clicking on the **Draw Trapezoid** icon, the panel fields for selecting by drawing a trapezoid are placed under the Data Source icons and the trapezoid drawing process is automatically started.*

Four positions representing the four vertices of the trapezoid by clicking LB at each of the positions. The trapezoid is completed after the fourth position is selected.

Strings are selected by using the trapezoid and the Trapezoid mode which is:

*if **Process only selected vertices** is ticked on, **Trapezoid mode** choices are:*

Vertices inside or Vertices outside

*if **Process only selected vertices** is not ticked, **Trapezoid mode** choices are:*

String inside, String outside, String crossing, String inside/crossing, String outside/crossing

*Any strings/vertices satisfying the **Trapezoid mode** will be highlighted when the trapezoid is completed.*

*If **Process only selected vertices** or **Trapezoid mode** is modified, the new strings/vertices satisfying he new parameters are highlighted.*

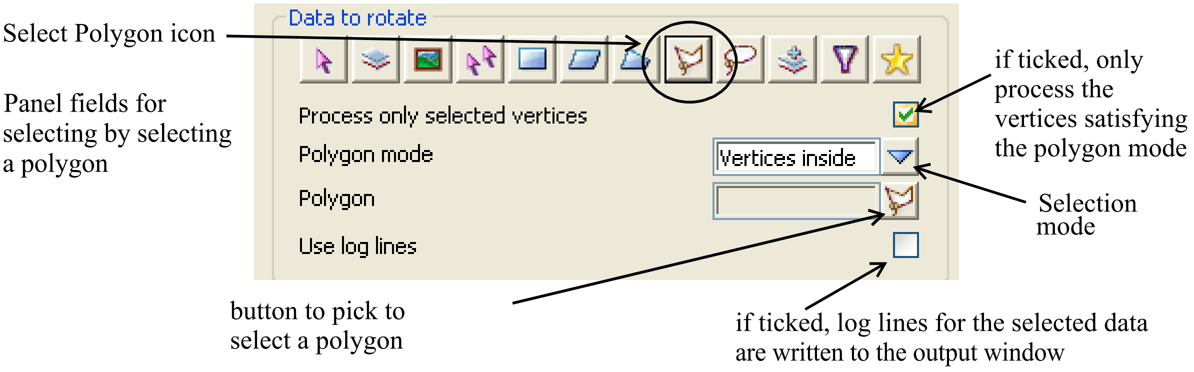
*If **Use log lines** is ticked, log lines for the selected data are written to the output window.*

*To draw a new trapezoid, click on the **Trapezoid** button and draw a new trapezoid.*

The Data Source Draw Trapezoid icon is made inactive after the option is first selected.

Continue to 4.19.3.8 Select an Existing Polygon or return to 4.19.3 Data Source

4.19.3.8 Select an Existing Polygon



After clicking on the ***Select Polygon*** *icon, the panel fields for selecting by selecting a polygon are placed under the Data Source icons.*

A polygon is then selected by clicking on the Select polygon button on the right of the Polygon panel field, and then selecting a polygon.

Note*: a polygon is just a closed string. If an open string is selected then a closed string is automatically created by joining the first and the last vertices of the string.*

if ***Process only selected vertices*** *is ticked on,* ***Polygon mode*** *choices are:*

Vertices inside or Vertices outside

if ***Process only selected vertices*** *is not ticked,* ***Polygon mode*** *choices are:*

String inside, String outside, String crossing, String inside/crossing, String outside/crossing

Any strings/vertices satisfying the ***Polygon mode*** *will be highlighted when the polygon is selected.*

If ***Process only selected vertices*** *or* ***Polygon mode*** *is modified, the new strings/vertices satisfying he new parameters are highlighted.*

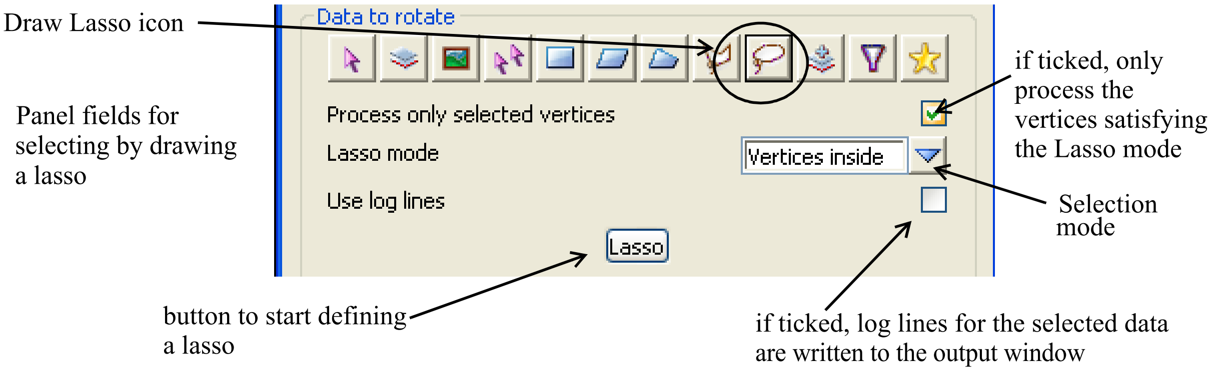
If ***Use log lines*** *is ticked, log lines for the selected data are written to the output window.*

To select a new polygon, click on the Select polygon button on the right of the Polygon panel field and select a new polygon.

The Data Source Select Polygon icon is made inactive after the option is first selected.

Continue to 4.19.3.9 Select Using a User Drawn Lasso or return to 4.19.3 Data Source

4.19.3.9 Select Using a User Drawn Lasso



*After clicking on the **Draw Lasso** icon, the panel fields for selecting by drawing a lasso are placed under the Data Source icons.*

*The lasso drawing process is started by clicking on the **Lasso** button.*

A lasso is then defined by clicking down on the LB and whilst holding the LB down, moving the cursor around to define a freehand lasso. The lasso is completed by letting the LB up and then clicking and releasing MB.

Strings are selected by using the lasso and the Lasso mode which is:

*if **Process only selected vertices** is ticked on, **Lasso mode** choices are:*

Vertices inside or Vertices outside

*if **Process only selected vertices** is not ticked, **Lasso mode** choices are:*

String inside, String outside, String crossing, String inside/crossing, String outside/crossing

*Any strings/vertices satisfying the **Lasso mode** will be highlighted when the lasso is completed.*

*If **Process only selected vertices** or **Lasso mode** is modified, the new strings/vertices satisfying the new parameters are highlighted.*

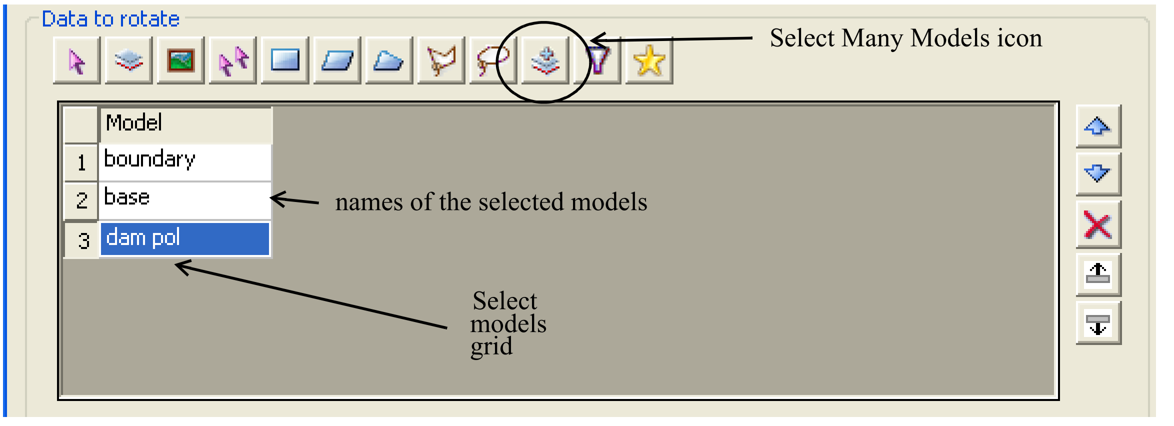
*If **Use log lines** is ticked, log lines for the selected data are written to the output window.*

*To draw a new lasso, click on the **Lasso** button and draw a new lasso.*

The Data Source Draw Lasso icon is made inactive after the option is first selected.

Continue to 4.19.3.10 Select many models or return to 4.19.3 Data Source

4.19.3.10 Select many models



*After clicking on the **Select Many Models** icon, the Select Many Models grid is placed under the Data Source icons.*

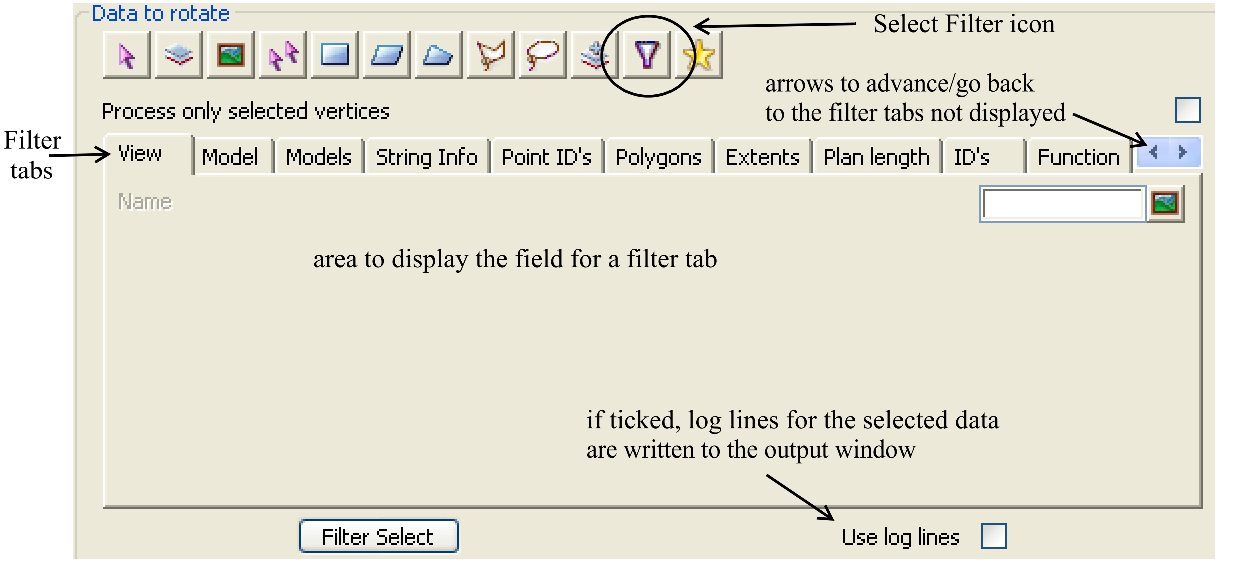
To begin selecting models, type the model name into the a field of the grid, or click RB in the grid and select from the Select Model pop-up (if a pop-up with Browse comes up, select Browse to bring up the Select Model pop-up).

Pressing the <Enter> key will add another blank row to the grid. Or clicking RB on the numbers on the left of the grid will bring up a menu to insert, delete, clear, cut, copy and paste rows. The buttons on the right hand side of the grid will also allow for rows to be moved up or down, delete rows and insert blank rows.

The Data Source Select Many Model icon is made inactive after the option is first selected.

Continue to 4.19.3.11 Select Filter or return to 4.19.3 Data Source

4.19.3.11 Select Filter



*After clicking on the **Select Filter** icon, the **Process only selected vertices** tick box, the **Select Filter tabs**, **Filter Select** button and **User log lines** tick box are placed under the Data Source icons.*

The option works by selecting the data that satisfies EVERY filter tab. That is, the data satisfies ALL the values in ALL the filter tabs.

*So **all** the filter tabs work together to define the selected data.*

For each filter tab required, the user clicks on the tab and fills in the required fields in the filter tabs that are to be satisfied by the selected strings.

*When all the required filter tabs filled in, click on the **Filter Select** button to select the data that satisfies all the filter tabs.*

*If **Process only selected vertices** is ticked, only the vertices of the strings that satisfy the filters are used.*

are selected are

*If **Use log lines** is ticked, log lines for the selected data are written to the output window.*

The full set of filter tabs are:

See 4.19.3.11.1 View Filter 4.19.3.11.2 Model Filter 4.19.3.11.4 String Info Filter
See 4.19.3.11.5 Point ID's Filter4.19.3.11.6 Polygons Filter4.19.3.11.7 Extents Filter
See 4.19.3.11.7 Extents Filter4.19.3.11.8 Plan Length Filter4.19.3.11.9 String ID's Filter
See 4.19.3.11.10 Function Filter4.19.3.11.11 Time Filter4.19.3.11.12 Model Attributes Filter
See 4.19.3.11.13 Element Attributes Filter 4.19.3.11.14 Vertex Attributes Filter 4.19.3.11.15 Segment Attributes Filter
See 4.19.3.11.16 Vertex Count Filter4.19.3.11.17 Tags Filter4.19.3.11.18 Name Masks Filter
See 4.19.3.11.19 Vertex UID's Filter4.19.3.11.20 Segment UID's Filter

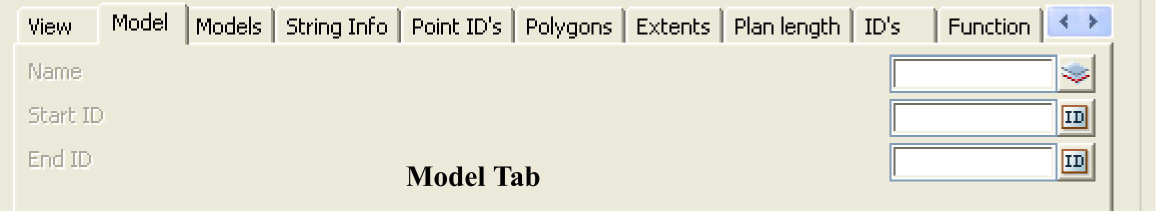
4.19.3.11.1 View Filter

View Tab

A view name can be typed into the panel field or a view selected from the View icon from the View panel field (to the right of the view name field).

Continue to 4.19.3.11.2 Model Filter or return to 4.19.3.11 Select Filter or 4.19.3 Data Source

4.19.3.11.2 Model Filter

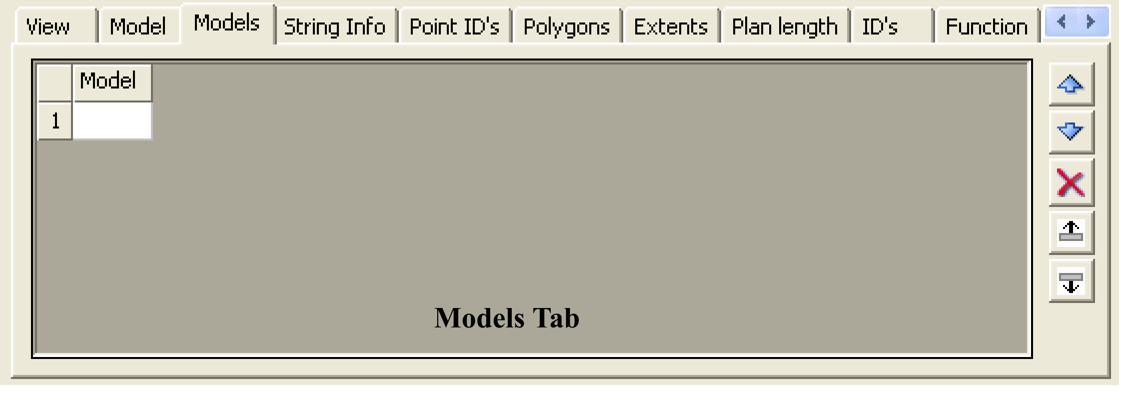


Model Tab

A model name can be typed into the panel field or a model selected from the Model icon from the Model panel field (to the right of the model name field).

Continue to 4.19.3.11.3 Models Filter or return to 4.19.3.11 Select Filter or 4.19.3 Data Source

4.19.3.11.3 Models Filter

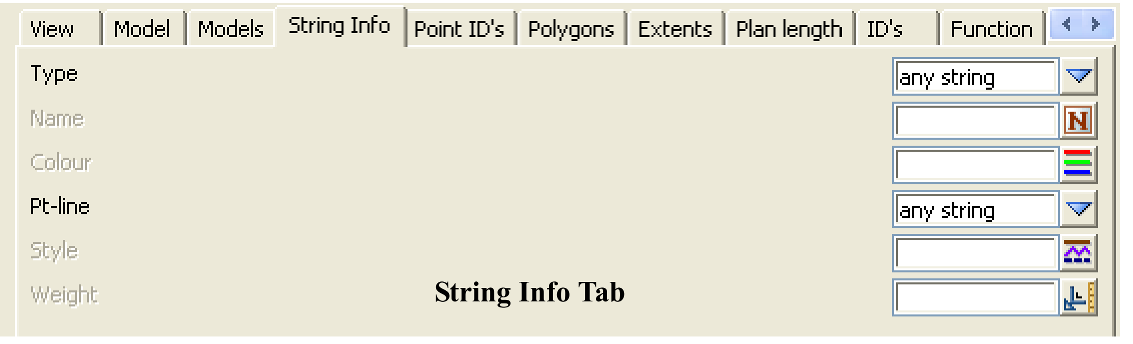


Models Tab

The names of one or models can be typed into the ***Model*** *grid, or by clicking RB in the grid and selecting from the Select Model pop-up (if a pop-up with Browse comes up, select Browse to bring up the Select Model pop-up).*

Continue to 4.19.3.11.4 String Info Filter or return to 4.19.3.11 Select Filter or 4.19.3 Data Source

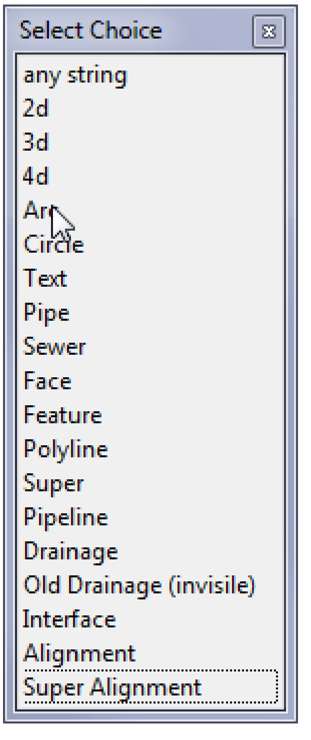
4.19.3.11.4 String Info Filter



String Info Tab

Strings can be selected by setting on or more properties in the String Info tab.

Type - the type of string



Types

Name - the name of the string

Colour - the colour of the string

Pt-line - the breakline type of the string

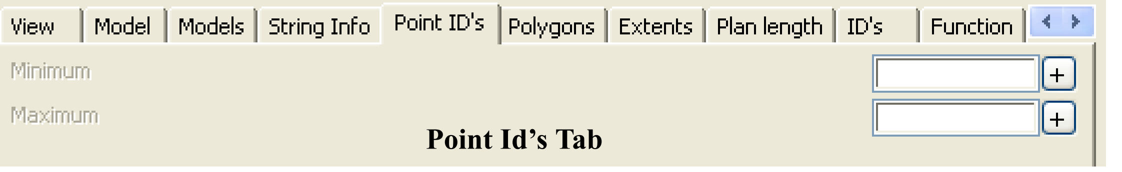
Style - the linestyle of the string

Weight - the weight of the string

If a field is left blank, then it is not used in the selection process.

Continue to 4.19.3.11.5 Point ID's Filter or return to 4.19.3.11 Select Filter or 4.19.3 Data Source

4.19.3.11.5 Point ID's Filter



Point Id's Tab

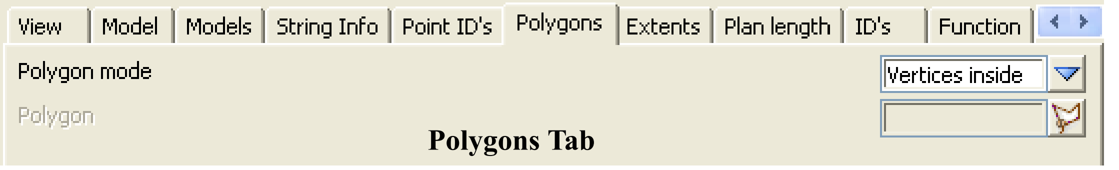
Minimum - if non-blank, then if a string vertex has a Point Id, then this is the minimum value that it can be. The Point id can be text so the ascii sort sequence is used for comparisons.

Maximum - if non-blank, then if a string vertex has a Point Id, then this is the maximum value that it can be. The Point id can be text so the ascii sort sequence is used to comparisons.

If a field is left blank, then it is not used in the selection process.

Continue to 4.19.3.11.6 Polygons Filter or return to 4.19.3.11 Select Filter or 4.19.3 Data Source

4.19.3.11.6 Polygons Filter



Polygons Tab

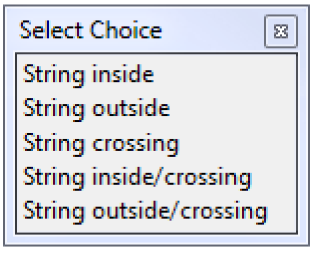
A polygon is selected/created to restrict the selection of strings by the choice given in the **Polygon mode** *field.*

Clicking LB on the Select Polygon icon on the right hand side of the **Polygon** *field allows the user to select a polygon.*

Clicking RB on the Select Polygon icon on the right hand side of the **Polygon** *field brings up the Polygon Choice Box for the user to select a method of creating/selecting a polygon.*

Clicking MB does nothing.

The strings selected are then restricted to those using the polygon and satisfying the ***Polygon mode****:*



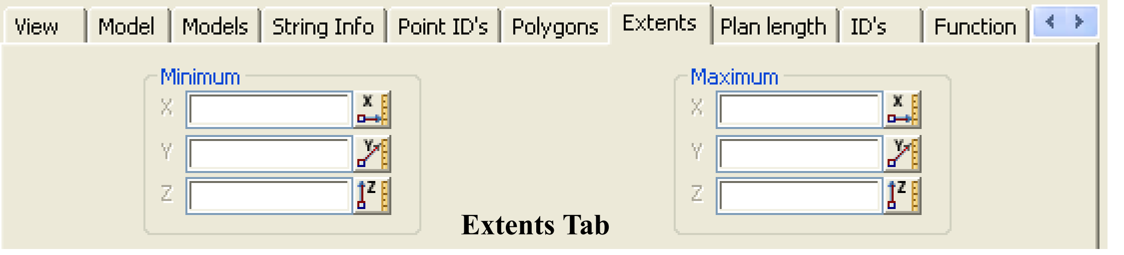
String inside - strings totally inside the polygon
String outside - strings totally outside the polygon
String crossing - strings crossing the polygon
String inside/crossing - strings totally inside or crossing the polygon
String outside/crossing - strings totally outside or crossing the polygon

Polygon Modes

Clicking MB does nothing.

Continue to 4.19.3.11.7 Extents Filter or return to 4.19.3.11 Select Filter or 4.19.3 Data Source

4.19.3.11.7 Extents Filter



Extents Tab

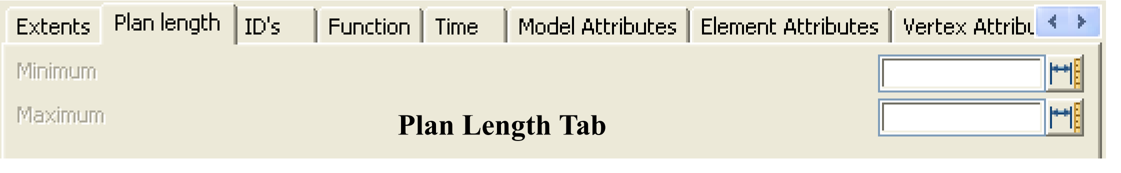
Minimum X - if non-blank, all the string's X coordinates must be at least this value.
Minimum Y - if non-blank, all the string's Y coordinates must be at least this value.
Minimum Z - if non-blank, all the string's Z coordinates must be at least this value.

Maximum X - if non-blank, none of the string's X coordinates can be greater than this value.
Maximum Y - if non-blank, none of the string's Y coordinates can be greater than this value.
Maximum Z - if non-blank, none of the string's Z coordinates can be greater than this value.

If a field is left blank, then it is not used in the selection process.

Continue to 4.19.3.11.8 Plan Length Filter or return to 4.19.3.11 Select Filter or 4.19.3 Data Source

4.19.3.11.8 Plan Length Filter



Plan Length Tab

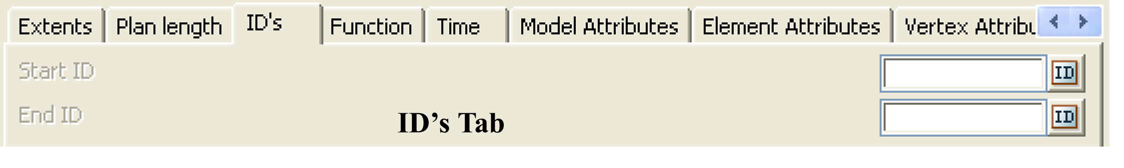
Minimum - if non-blank, the string's plan length (2d length) must be at least this value.

Maximum - if non-blank, the string's plan length (2d length) can't be greater than this value.

If a field is left blank, then it is not used in the selection process.

Continue to 4.19.3.11.9 String ID's Filter or return to 4.19.3.11 Select Filter or 4.19.3 Data Source

4.19.3.11.9 String ID's Filter



ID's Tab

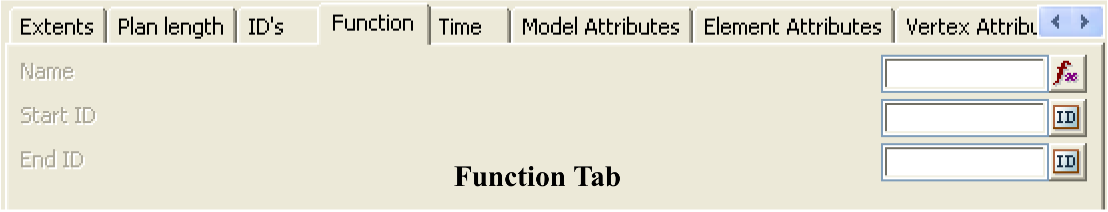
Start ID - if non-blank, the string's ID must be at least this value.

End ID - if non-blank, the string's ID can't be greater than this value.

If a field is left blank, then it is not used in the selection process.

Continue to 4.19.3.11.10 Function Filter or return to 4.19.3.11 Select Filter or 4.19.3 Data Source

4.19.3.11.10 Function Filter



Function Tab

Name - if non-blank, the string must be part of a function with this name.

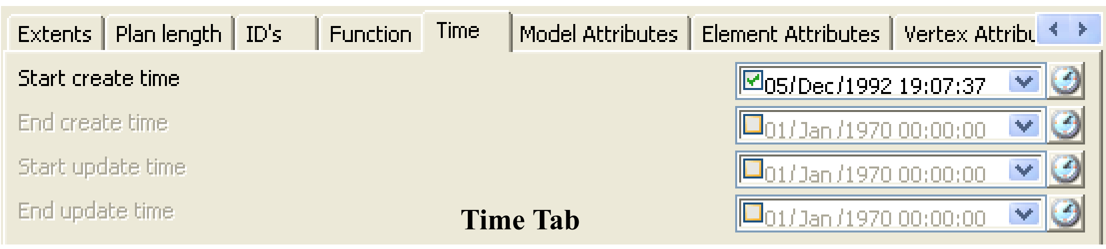
Start ID - if non-blank, the string's ID must be at least this value.

End ID - if non-blank, the string's ID can't be greater than this value.

If a field is left blank, then it is not used in the selection process.

Continue to 4.19.3.11.11 Time Filter or return to 4.19.3.11 Select Filter or 4.19.3 Data Source

4.19.3.11.11 Time Filter



Time Tab

Start create time - if non-blank, the string's time of creation must be at least this time.

End create time - if non-blank, the string's time of creation can't be greater than this time.

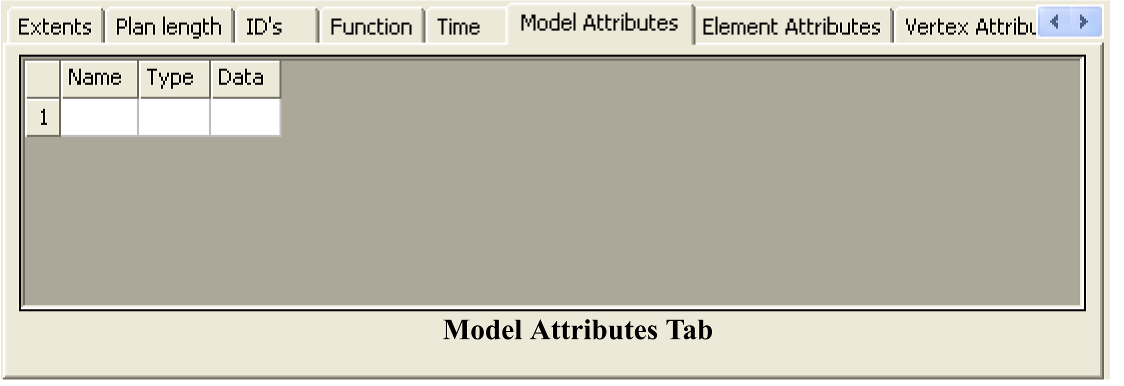
Start update time - if non-blank, the time the string was last updated must be at least this time.

End update time - if non-blank, the time the string was last updated can't be greater than this time.

If a field is left blank, then it is not used in the selection process.

Continue to 4.19.3.11.12 Model Attributes Filter or return to 4.19.3.11 Select Filter or 4.19.3 Data Source

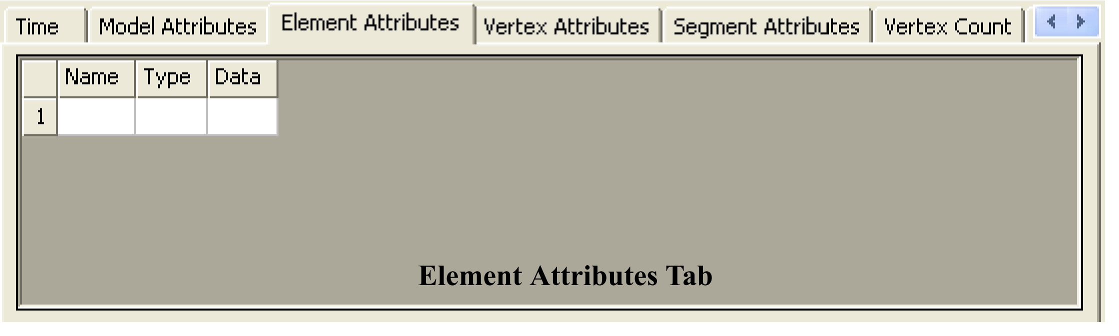
4.19.3.11.12 Model Attributes Filter



Model Attributes Tab

Continue to 4.19.3.11.13 Element Attributes Filter or return to 4.19.3.11 Select Filter or 4.19.3 Data Source

4.19.3.11.13 Element Attributes Filter



Element Attributes Tab

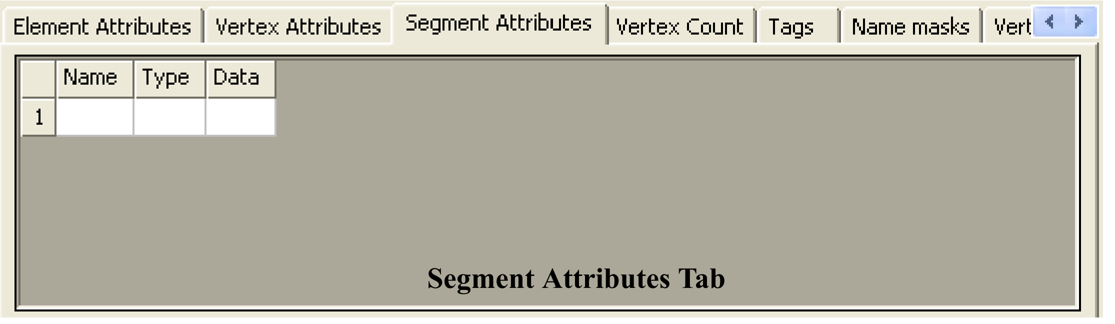
Continue to 4.19.3.11.14 Vertex Attributes Filter or return to 4.19.3.11 Select Filter or 4.19.3 Data Source

4.19.3.11.14 Vertex Attributes Filter

Vertex Attributes Tab

Continue to 4.19.3.11.15 Segment Attributes Filter or return to 4.19.3.11 Select Filter or 4.19.3 Data Source

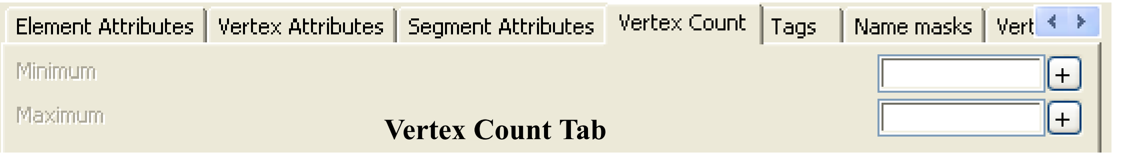
4.19.3.11.15 Segment Attributes Filter



Segment Attributes Tab

Continue to 4.19.3.11.16 Vertex Count Filter or return to 4.19.3.11 Select Filter or 4.19.3 Data Source

4.19.3.11.16 Vertex Count Filter



Vertex Count Tab

Minimum - if non-blank and the string have vertices, then the number of vertices must be at least this value.

Maximum - if non-blank and the string have vertices, then the number of vertices can't be greater than this value.

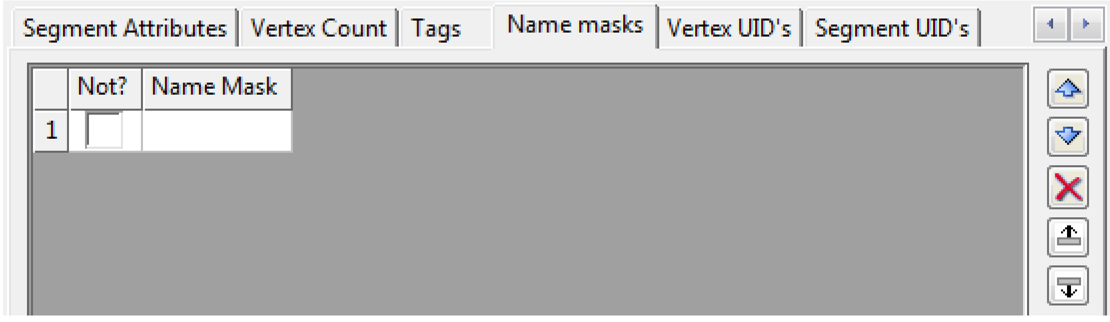
Continue to 4.19.3.11.17 Tags Filter or return to 4.19.3.11 Select Filter or 4.19.3 Data Source

4.19.3.11.17 Tags Filter

Tags Tab

Continue to 4.19.3.11.18 Name Masks Filter or return to 4.19.3.11 Select Filter or 4.19.3 Data Source

4.19.3.11.18 Name Masks Filter



Name Masks Tab

The Name Masks are used to restrict the strings by comparing the string name with the masks in the Name Mask column.

The order of the Names Masks is important.

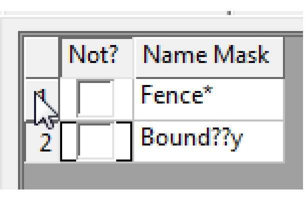
The string name is first compared to the first Name Mask and if the string satisfies the Name Mask, then the string satisfies the Name Masks Tab and no more tests are done for that string.

If the string does not satisfy the first Name Mask, then the string is then compared to the second Name Mask.

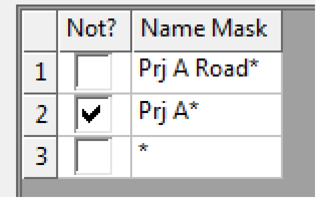
This process is repeated until either the string matches a Name Mask and is selected, or the string matches no Name Mask and so is not selected.

The Name Masks can include the wild card character "" which stands for zero or characters, and the wild character "?" which stands for just one character.*

*If **Not** is ticked, then strings are selected if they **don't satisfy** that Name Mask.*



include all strings starting with "Fence"
include all strings starting with "Bound" then any two characters and then "y"
So only include the "Fence" strings and "Bound??y" strings



include all strings starting with "Prj A Road"
exclude all other strings starting with "Prj A"
include all other strings
So only include the "Prj A Road" strings from "Prj A",*
and all other strings not starting with "Prj A"

Continue to 4.19.3.11.19 Vertex UID's Filter or return to 4.19.3.11 Select Filter or 4.19.3 Data Source

4.19.3.11.19 Vertex UID’s Filter

Vertex UIDS’s Tab

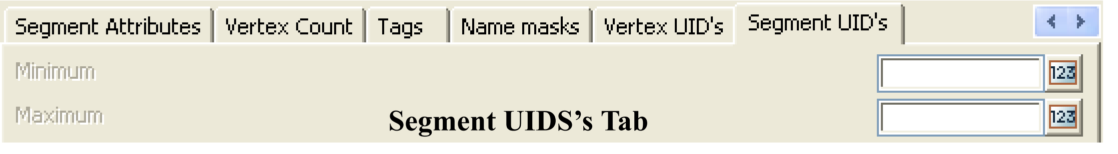
Minimum - if non-blank, then if a string has vertices, then this is the minimum value that the UID of the vertex can be.

Maximum - if non-blank, then if a string has vertices, then this is the maximum value that the UID of the vertex can be.

If a field is left blank, then it is not used in the selection process.

Continue to 4.19.3.11.20 Segment UID’s Filter or return to 4.19.3.11 Select Filter or 4.19.3 Data Source

4.19.3.11.20 Segment UID’s Filter



Segment UIDS’s Tab

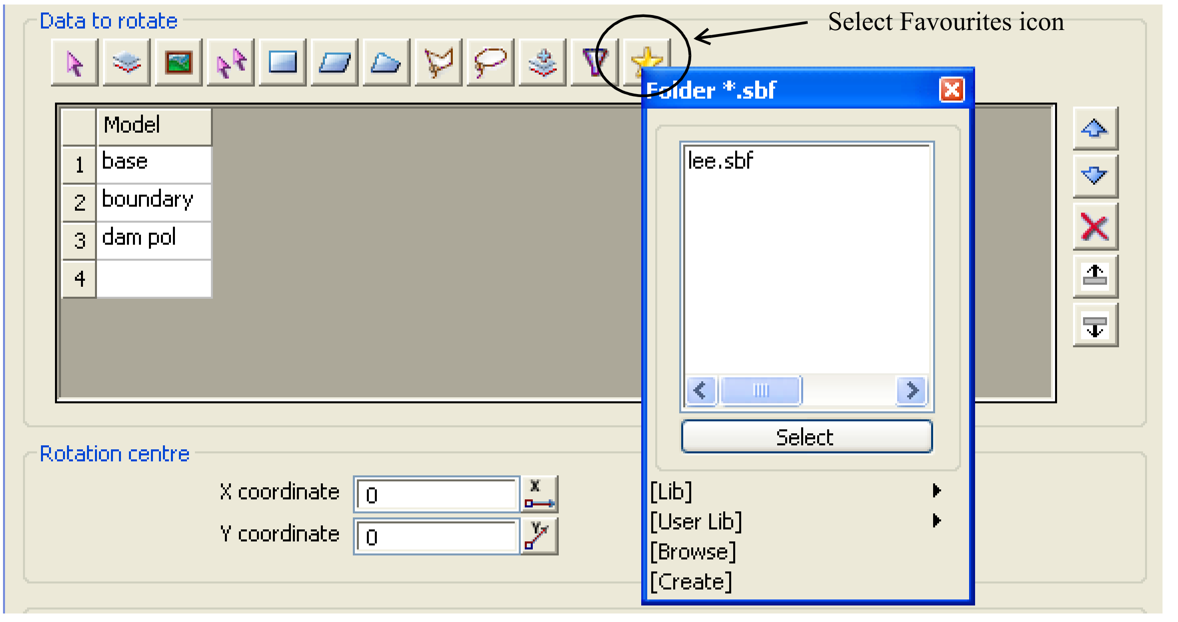
Minimum - if non-blank, then if a string has segments, then this is the minimum value that the UID of the segment can be.

Maximum - if non-blank, then if a string has segments, then this is the maximum value that the UID of the segment can be.

If a field is left blank, then it is not used in the selection process.

Continue to 4.19.3.12 Favourites or return to 4.19.3.11 Select Filter or 4.19.3 Data Source

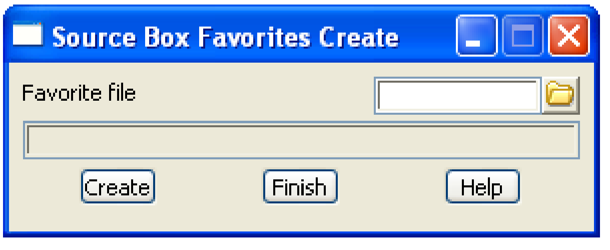
4.19.3.12 Favourites



*After clicking on the **Select Favourites** icon, the Folder *.sbf pop-up is raised showing any local source box favourites files (*.sbf) and with options to look in Lib, User Lib, Customer Lib (if it exists) or to Browse for a sbf file.*

If a sbf file is selected, then the values in it will be used to set up the Data Source Selection.

To write out an sbf file that saves all the parameters for the current Source Box Selection, click on [Create]. This will bring up the Source Box Favourites Create panel.



*To create a sbf file, simply type the name for the sbf file into the Favourite file panel field and then click on **Create**.*

In the documentation for a panel with a data source, only the panel for **Data Source type** *Model* will be shown.

Go to the next section 4.19.4 Data Target or return to 4.19.3 Data Source or 4.19 Panel Fields.

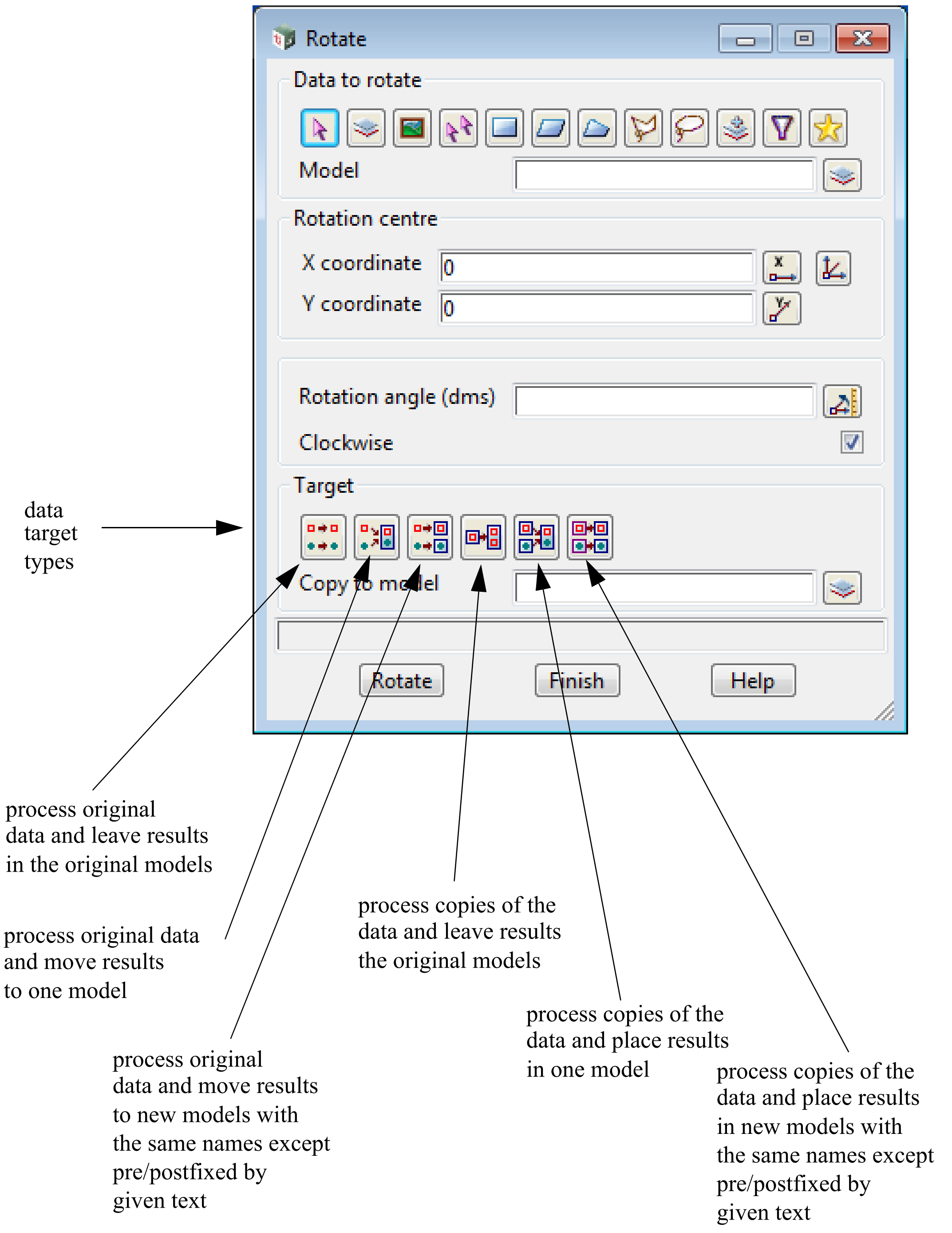
4.19.4 Data Target

Many options contain the panel field **Data target** which may have one or more of the selection choices:

```
move to original model(s)/replace      // process the original data and leave in models
move to one model                      // move the processed data to one model
move to many models                    // move processed data to models with the same
                                       // names as the original models except the model
                                       // names are pre/postfixed by given text.

copy to original model(s)/replace      // process copies of the data and leave in models
copy to one model                      // process copies of the data and place in one model
copy to many models                    // process copies of the data and place in models with
                                       // the same names as the original models except the
                                       // model names are pre/postfixed by given text.
```

Depending on the choice, the next panel field will be changed to suite the choice of target.

For example, for the choice **Copy to one model**, the next field will be **Copy to model**.



In the documentation for a panel with a data target, only the panel for **Data Target type** *Copy to Model* will be shown.

Go to the next section 4.19.5 Pop-Up Lists and Menus or return to 4.19 Panel Fields.

4.19.5 Pop-Up Lists and Menus

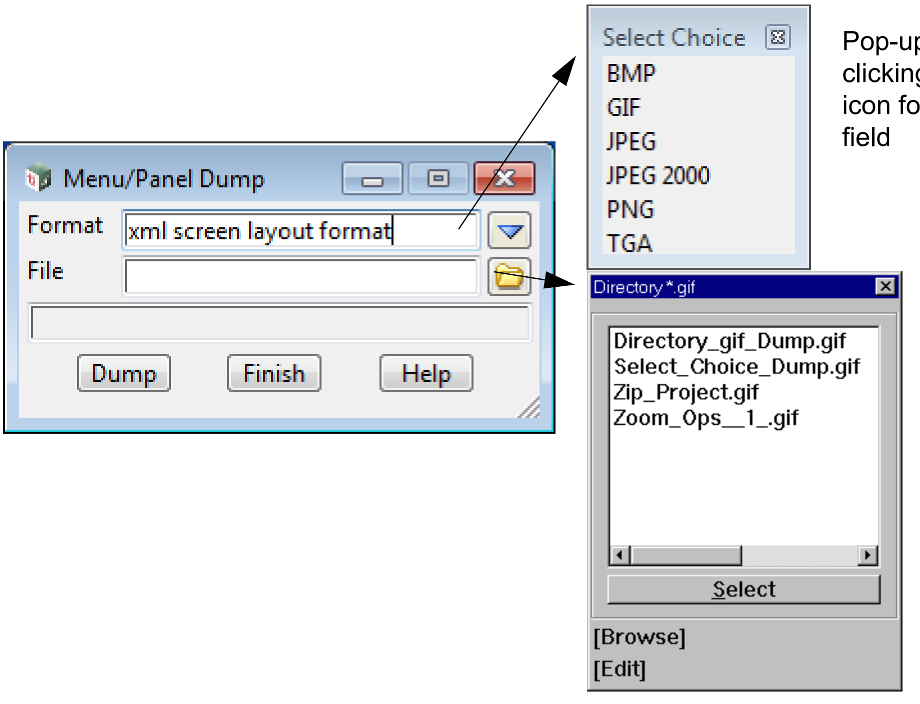
If there are choices available to select from for the panel field, there is a [+] or another special icon displayed at the right hand side of the panel field.

For example, icons that may be used in place of the **+** are:

file/folder	tin	textstyle info
model	choice	line weight
colour when none selected	selected colour	view
line style	polygon	symbol
project	XYZ	X
Y	Z	chainage
string name	select	angle
date/time	function	typed input
ID	plotter/printer	same as

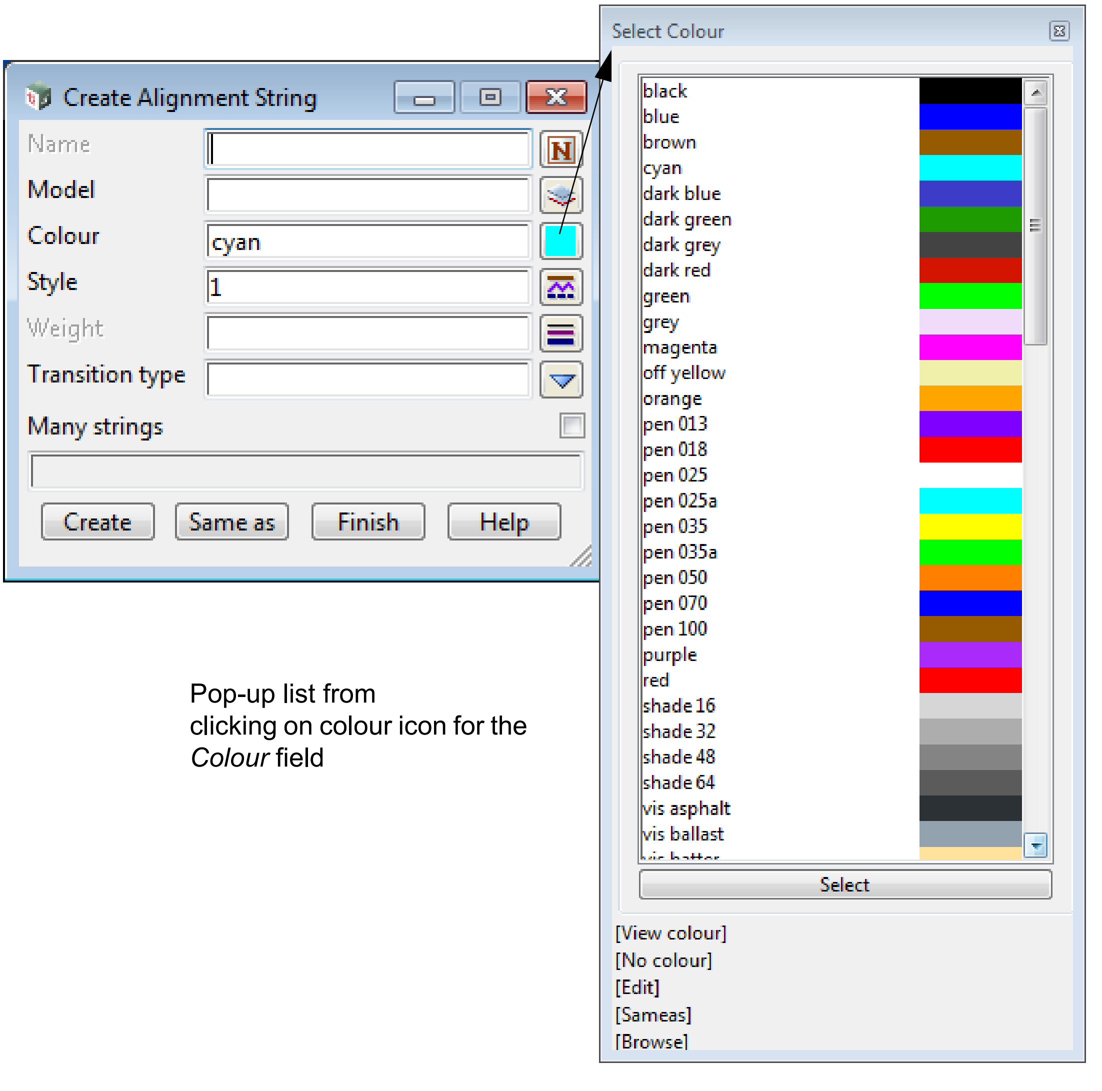
Clicking LB on the one of the above icons or a **[+]** brings up a the **panel field pop-up list** or **pop up menu**.

A panel field pop-up consists of a list of choices which may be displayed as either a *menu* or a *list*



Pop-up menu from clicking on the choice icon for the *Format* field

Pop-up list from clicking on the folder icon for the *File* field



Pop-up list from clicking on colour icon for the *Colour* field

For a *pop-up list*, an answer is chosen from the list by **double** clicking LB over the required answer.This answer is then displayed in the panel field and the pop-up list disappears.

At the bottom of some *pop-up lists* is a:

(a) **[Sameas]** button. If **[Sameas]** is selected then an object with the require property is selected using the mouse and the value from the selected item is written to the panel field.

(b) **[Browse** button. The meaning of **[Browse]** is different for each pop up list.

For some options, more than one selection from a list can be made. In that case, simply use the standard Microsoft methods for multiple selections and then click on **Select**.

For a *pop-up menu*, an answer is chosen from the pop-up menu by clicking LB over the required answer.This answer is then displayed in the panel field and the pop-up menu disappears.

If there are more items to be displayed in a pop-up menu than the number **Popup length** in the panel **Defaults**, the list of choices will have a scroll bar.

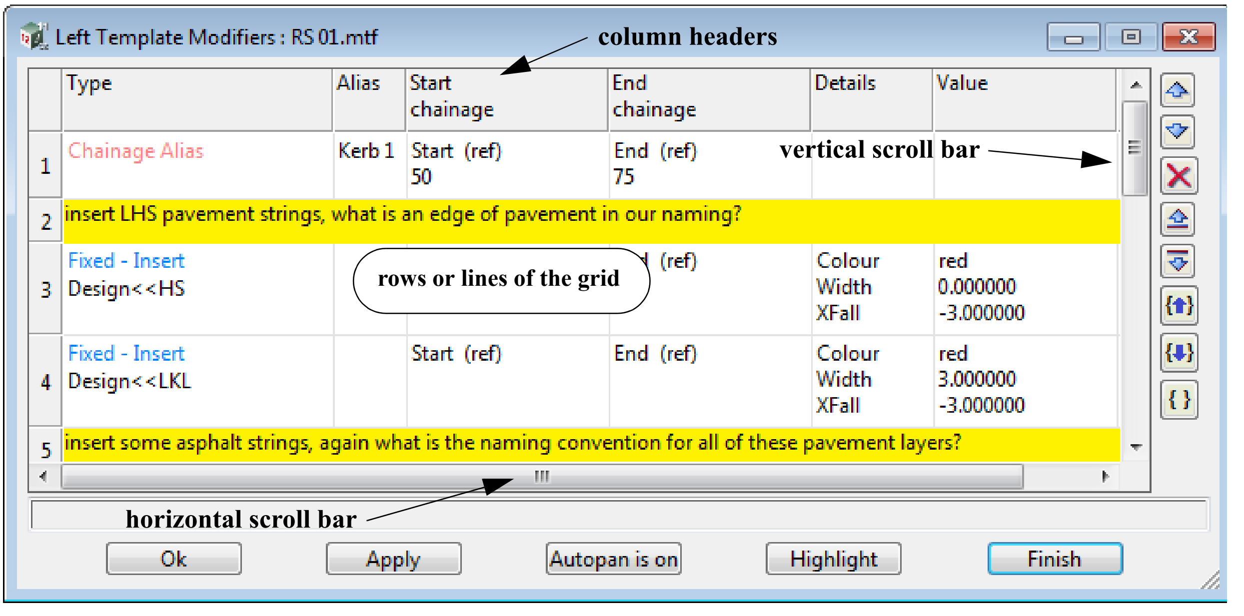
The pop-up list or menu can also be removed without a selection by clicking LB on the **[X]** on the pop-up list or menu, by clicking LB again on the choice icon or **[+]** for the panel field, or by simply typing into any visible part of the panel field that the pop-up is for (some of the field may be obscured by the pop-up itself).

Go to the next section 4.19.6 Grids in Panels or return to 4.19 Panel Fields.

4.19.6 Grids in Panels

Many **12d Model** panels include grids or lists of information which hold an unlimited number of lines or **rows**. The grids may have more than one **column** but the number of columns is usually fixed for that grid.

Each column of the grid normally has a **column header** such as **Type** and **Alias** in the example of the **Left MTF MTF Modifiers** panel.



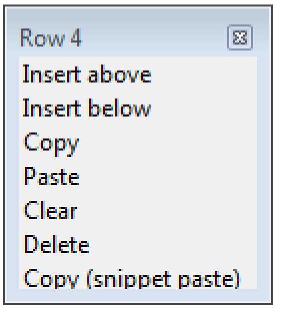
The grid in a panel has only a limited size and so can only display a certain number of rows of the grid. and once the number of rows in the panel exceeds that number, a vertical scroll bar is created on the right hand side of the grid so you can scroll the rows up or down to see the extra rows.

Similarly the grid in a panel can only display a certain number of columns of the grid and once the number of columns in the panel exceeds that number, a horizontal scroll bar is created on the bottom of the grid so you can scroll the columns left and right to see the extra columns.

The **numbers** on the left hand side of the grid indicate the **row** (line) number of the information being displayed and clicking LB on the row number will highlight the row. Additional rows can be highlight at the same time by using the standard Microsoft key combinations on the **row numbers**. For example, <Ctrl>+LB on a row number to add that row to the highlighted rows.

The row numbers are also buttons and when clicked with RB, the **Edit Row n** panel is displayed.

For example



The row that the **Edit Row n** panel was raised on is called the **current** row.

The options on the **Edit Row n** panel have the following functions:

Insert above insert a blank line above the row the current row. The current row and all following rows are pushed down

Insert below insert a blank line below the current row. All the rows below the inserted blank row are pushed down

clear clear the highlighted rows

copy copy the current line into the next line; all following lines are pushed down

delete delete the highlighted rows

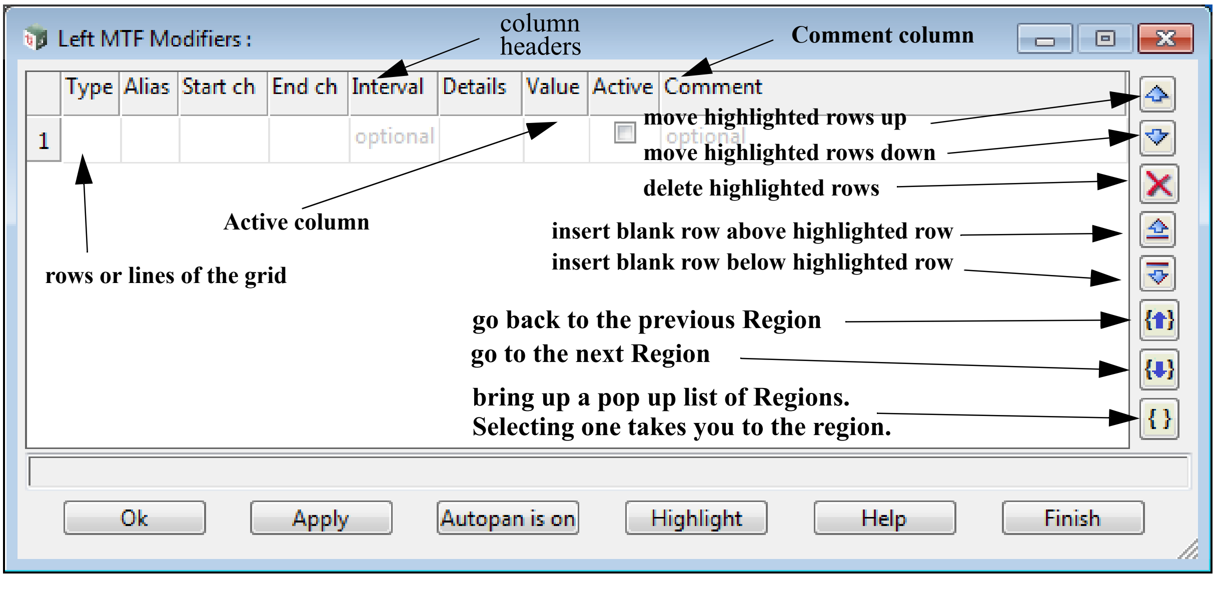
up move each of the highlighted rows up one row.

down move each of the highlighted rows down one row

Also, when rows are highlighted, pressing <Delete> will **delete** the highlighted rows.

There are also icons on the right hand side of the panel that can move highlighted rows up and down, delete highlighted rows, and insert blank lines above and below a selected row.

Plus for some grids, Regions can be defined and their are icons on the right hand side of the grid to go back to the previous region, go forward to the next region, and to go to a region selected from a region pop up list. See 4.19.6.0.1 Regions in Some Grids



Some grids have an **Active** column, which may contain **Tick** boxes, or **Yes/No Choice** boxes.

If the **Active** column for a command is ticked **on**, or **yes**, then the row is used.

If the **Active** column for a command is **not ticked**, or **no**, then the row is not used.

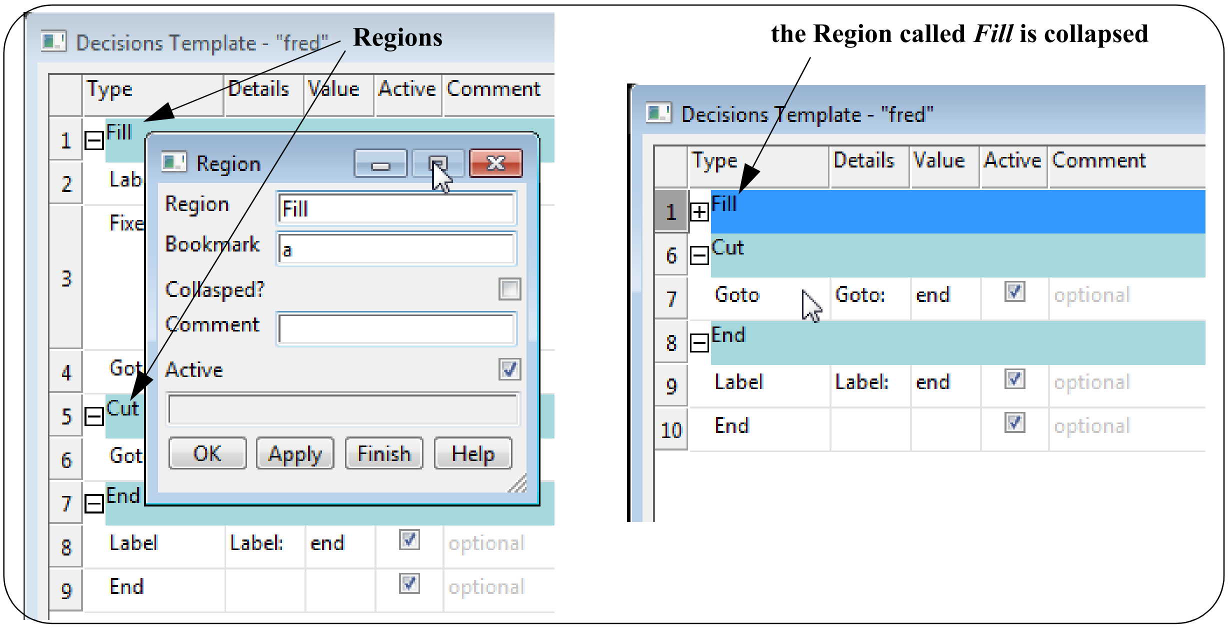
Some grids have a **Comment** column, which may contain text.

4.19.6.0.1 Regions in Some Grids

In some grids there is a **Region** command (or rule).

A **Region** in a grid is a special command that has the property that a Region can be **collapsed**. And *collapsing* a Region means that all the rows after the Region command until the next Region command, are hidden in the grid. The row of the **Region** command is coloured light blue.

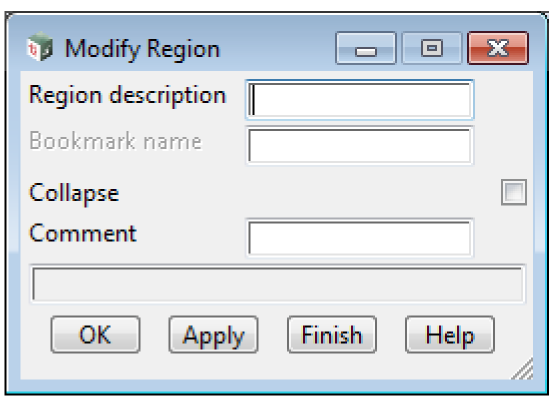
Regions must have a unique name within the grid.



A **Region** can have a secondary name called a **Bookmark**. If no Bookmark is given then when the grid is saved, the **Bookmark** is given the same name as the Region.

All grids that have *Regions* also have a command in the Grid for ***creating a Region***.

For example, for the **Left/Right MTF Modifiers** Grid, the panel to create a Region is **Modify Region**



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Region description	text box		

name for the Region.

Bookmark name	text box		

*if **not blank**, this is a secondary name, often a shorter name, that is used in the **Region** pop up rather than using the **Region description.***

*If **blank**, then when the grid is saved, the* **Bookmark name** *is set to be the same as the* **Region description**.

Collapse tick box

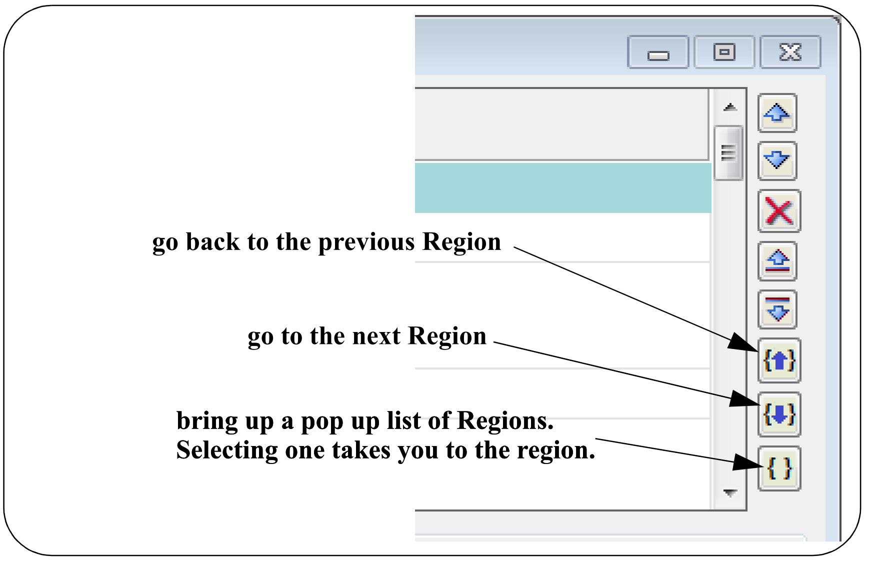
*if **ticked**, all the commands in the grid until the next **Region** command are collapsed into this Region command.*
*If **not ticked**, the commands in the grid until the next **Region** command are not collapsed.*

Once one or more *Regions* have been created in the grid, there are icons **Regions**, **Previous region** and **Next region**, on the right hand side of the grid that can then be used.

Clicking on the **Regions** icon will bring up a list of all the **Bookmarks** in the grid, and clicking on a **Bookmark** in the list will take you to that Region/Bookmark.

Previous region will take you from the highlighted row to the Region before the highlighted row.

Next region will take you from the highlighted row to the next Region after the highlighted row.



Go to the next section 4.19.7 Scrolling Panel Tables or return to 4.19 Panel Fields.

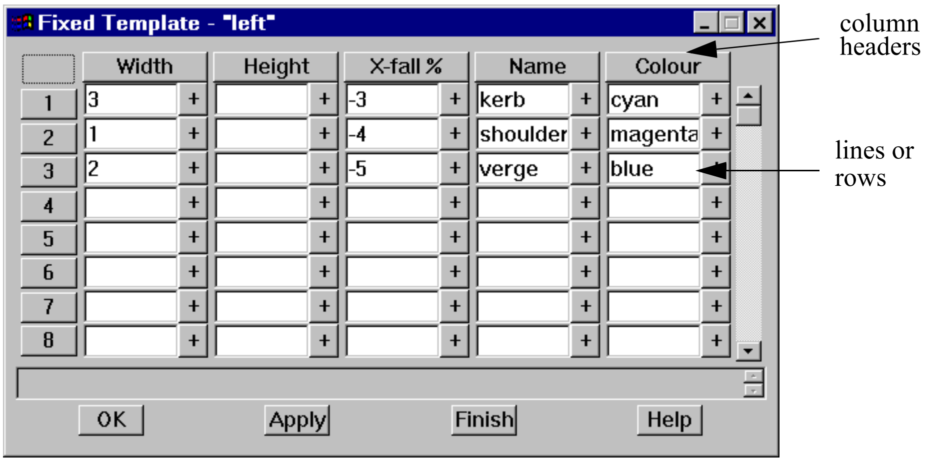
4.19.7 Scrolling Panel Tables

Scrolling Panel Tables have been mainly replaced by 4.19.6 Grids in Panels.

Some **12d Model** panels include tables or lists of information which hold an unlimited number of lines or **rows**. The tables may have more than one **column** but the number of columns is fixed for that table.

Such tables are displayed and edited in a **scrolling panel table**.

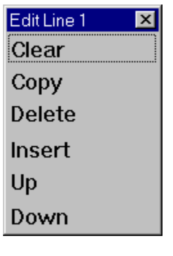
For example, the **Fixed Template** panel can contain an unlimited number of lines (rows) defining fixed links for the template.



Normally a set number of lines of the table are displayed in the **scrolling panel table** and once the number of lines exceeds the set number, the up and down arrows on the right hand side of the scrolling panel table are needed to scroll the lines up or down to see the extra information.

Each column of the panel table normally has a **column header** such as **width** and **height** in the above example of the **Fixed Template** scrolling panel table.

The **numbers** on the left hand side of the scrolling panel table indicate the line (row) numbers of the information being displayed and the line numbers are also buttons which when selected, bring up the **Edit Line n** panels. For example



The options on the **Edit Line n** panel have the following functions:

clear	clear the current line
copy	copy the current line into the next line; all following lines are pushed down
delete	delete the current line
insert	insert a blank line; the current line and all following lines are pushed down
up	swap the current line with the preceding line
down	swap the current line with the following line

Also, when the **number** on the left is selected, typing <del> will **delete** the line
<insert> will **insert** a line.

Go to the next section 4.19.8 File Box or return to 4.19 Panel Fields.

4.19.8 File Box

12d Model panels frequently include a field for entering a **file** name. This type of panel field is called a **file panel** field or box, or simply **file box**.

When RB is clicked on the **[+]** at the right of the panel field to bring up the list of available files, the pop-up menu can contain several different types of items:

(a) list of **local** files satisfying the file ending for the file box

(b) [Lib] button

(c) [User lib] button

(d) [Browse] button

(e) [Open] button for some special files which have an internal 12d Model editor

(f) [Open file]

(g) [Edit file]

(h) [Unicode format]

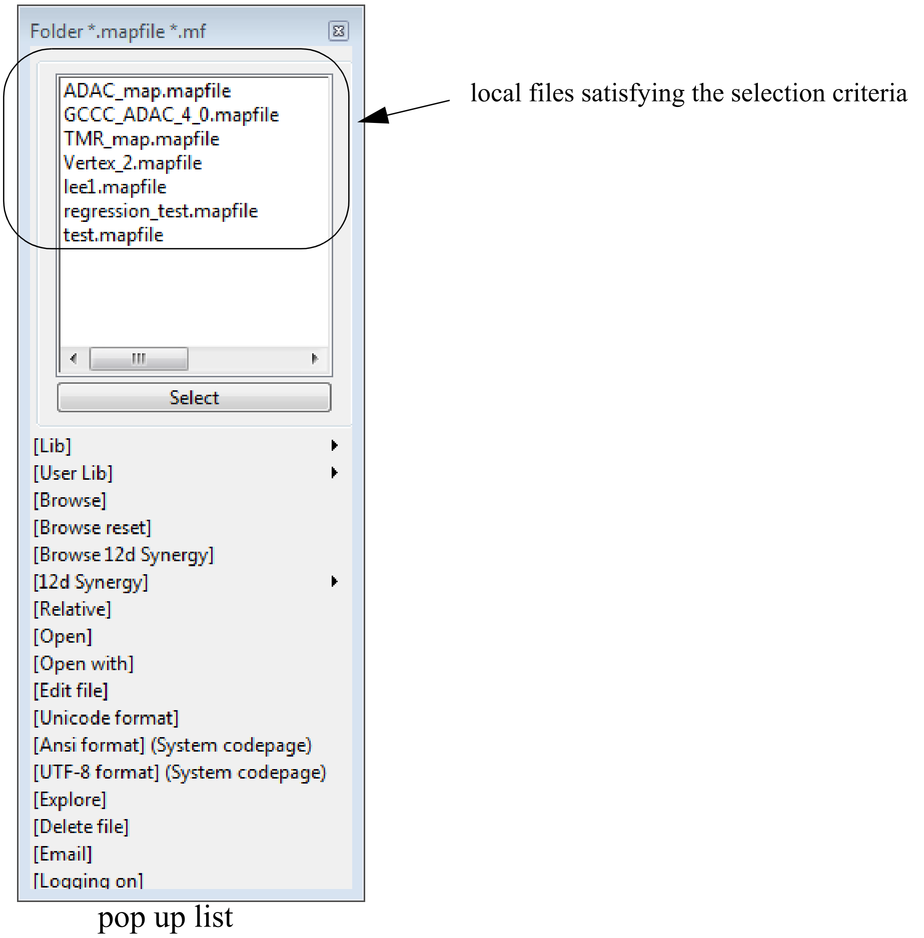
(i) [Ansi format]

(j) [UTF-8 format]

(k) [Explore]

(l) [Delete file]

For example,



pop up list

The special menu items under the list of local files satisfying the selection criteria (usually file endings) have the following meanings:

[Lib]

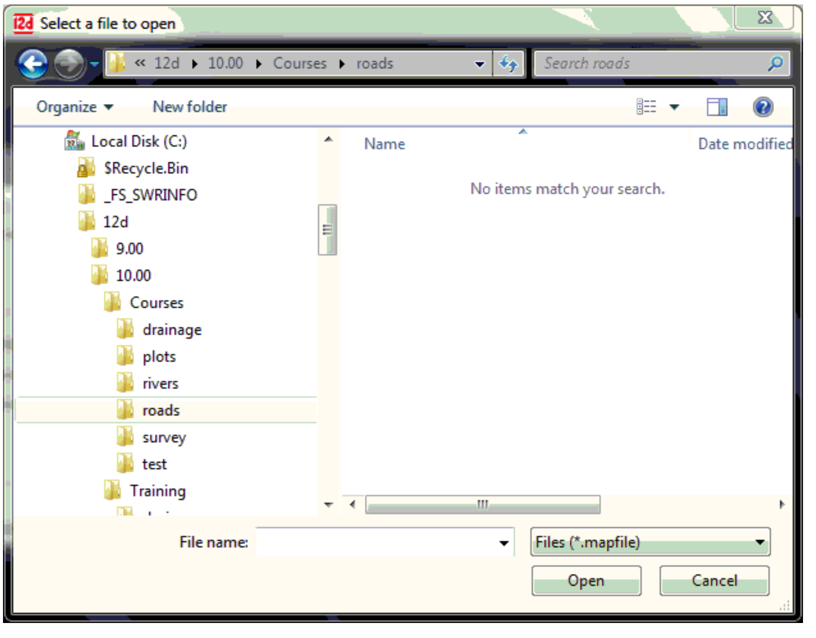
the walk-right lists all files satisfying the file box ending, in the library pointed to by the environment variable LIB_4D (see Appendix A)

[User Lib]

the walk-right lists all the files satisfying the file box ending, in the user library pointed to by the environment variable USER_LIB_4D (see Appendix A)

[Browse]

selecting **Browse** brings up the **Microsoft File Browser** which can be used to search for a file with a specified ending in the local or other folders. Note that the appearance of the Browser dialog depends on the version of Windows you are using. If a file is selected via the Open button in the current 'browsing' folder, the file name is loaded. If the file is selected from another folder, the full path and name for the file is loaded and the 'browsing' folder is updated to the file's folder.



[Browse reset]

selecting **Browse reset** sets the 'browsing' directory back to the working directory of the project, and then starts [Browse]. CAUTION: selecting Cancel will make the current file invalid if the file is not in the working folder or the file does not have a full path specified.

[Browse 12d Synergy]

selecting **Browse 12d Synergy** browses 12d Synergy folders. You need to have 12d Synergy installed for this to appear.

[12d Synergy]

walking right on **12d Synergy** brings up a menu containing options to Browse 12d Synergy for files or Add files to 12d Synergy folders. You need to have 12d Synergy installed for this to appear.

[Relative]

selecting **Relative** will take a full path name in the file box and convert it to a relative pathing. Note that this is only possible if the file is on the same disk as the Project. The path to the file in the file box will be updated so that it is relative to project's working folder.

[Open]

selecting **Open** will bring up the special ***12d Model*** editor for the file. If not a special ***12d Model*** file, the standard Windows File Associations will be used to open the file. So if the file is an Adobe PDF file, and a PDF 'Viewer' is installed, then the PDF file will be opened in that viewer. Note that in earlier versions of ***12d Model***, this was known as [Edit].

[Open with]

selecting **Open with** will bring up the Windows Open with program selector to allow the user to choose which program to open the selected file with. If there is one or more than one program associated with a file type, the dialog gives you the choice. Note that the appearance of the Open dialog depends on the version of Windows you are using.

[Edit file]

selecting **edit file** when there is a file name already in the file box, will edit the file using the editor pointed to by the EDITOR_4D environment variable.

[Unicode format]

selecting **Unicode format** sets the output format of currently selected file to the ***Unicode*** format with UTF-16 encoding.

Unicode allows for most languages of the world to be used concurrently. The resulting file will be marked with the Unicode BOM (byte order mark - see 4.4.6 Endian and BOM) so that other programs know this file is in Unicode format with UTF-16 encoding (see 4.4 Ascii, Ansi and Unicode and 4.4.5 Unicode Encoding: UTF-16).

Note ***12d Model*** is Unicode enabled to the UTF-16 level and allows the use of non-Latin alphabets supported by UTF-16. Not all programs can work with Unicode files so if the file being outputted is to be used by another program, you need to check if UTF-16 files can be read by the intended software.

[Ansi format]

selecting **Ansi format** sets the output format of currently selected file to the ’Ansi’ format (see 4.4 Ascii, Ansi and Unicode). The resulting file will not be marked with any BOM (byte order mark - see 4.4.6 Endian and BOM). Writing Unicode information to an Ansi file may result in the lost of information.

[UTF-8 format]

selecting **UTF-8 format** sets the output format of currently selected file to the ’UTF-8’ format which is a special Unicode format (see 4.4 Ascii, Ansi and Unicode and 4.4.4 Unicode Encoding: UTF-8). The resulting file will also marked with the UTF-8 BOM (byte order mark - see 4.4.6 Endian and BOM) so that other programs know this file is in UTF-8 format.

Note ***12d Model*** is Unicode enabled to the UTF-16 level and allows the use of non-Latin alphabets supported by UTF-16. Not all programs can work with Unicode files so if the file being outputted is to be used by another program, you need to check if UTF-8 files can be read by the intended software.

[Explore]

selecting **Explore** opens a new Windows Explorer window to the currently ’browsed’ folder.

[Delete file]

selecting **Delete file** will prompt the user to send the currently file to the Windows Recycle Bin if it exists for the drive that the file is on. Note this depends on which drive the file resides. Many network drives do not support Recycle Bins and so a deleted file is lost. It is up to the user to implement good backup procedures to avoid permanent loss of files.

[Email]

selecting **Email** emails the file given in the File Box. See .

[Logging off/on]

Toggle between *Logging on* and *Logging off*. This is only used for Map files and when ***Logging on*** is set, then entries are made a log lines in the Output Window that show what tabs in the Map File were used in creating and attributing the string.

Note that certain ’types’ of File Box many have additional [...] fields.

Go to the next section 4.19.9 Model Panel Field or return to 4.19 Panel Fields.

4.19.9 Model Panel Field

See

[4.19.9.1 [Options] on Model Box Pop Up](#)
[4.19.9.2 Adding Model to Views in Model Panel Field](#)

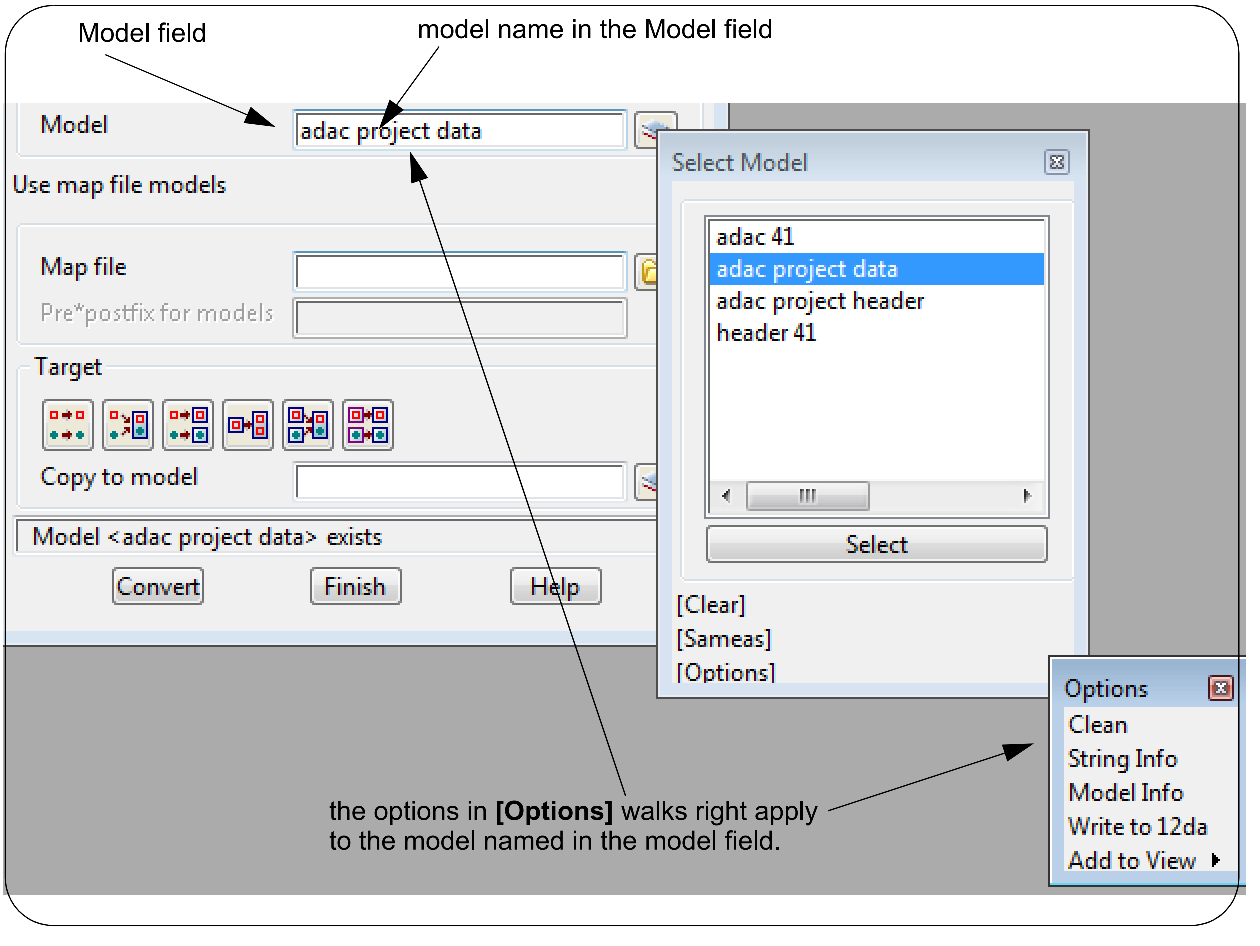
4.19.9.1 [Options] on Model Box Pop Up

When you have a **Model** field with a model name of an existing model in it, the **Select Model** pop up for the **Model** icon at the end of the **Model** field has an **[Options]** at the bottom of the pop up with a walk right list of options that apply to the model in the **Model** field.

Note that the model must exist so if the **Model** field in a panel is one that creates the model, then using **[Options]** will only make sense **after** the panel has been run and the model has been created.

Warning

The options on **[Options]** apply to the model in the **Model** field, **NOT** the model highlighted in the **Select Model** pop up.



Clean - cleans the model given in the **Model** field.

String Info - brings up the **String Information Table** panel and runs it for the model given in the **Model** field.

Model Info - brings up the **Model Information** panel and runs it for the model given in the **Model** field.

Write to 12da - brings up the **Write 12d Solutions Archive Data** panel with the model given in the

Model field set as the *Data Source*.

Add to View - adds the model given in the **Model** field to the view selected from the Add to View walk right list.

Go to the next section 4.19.9.2 Adding Model to Views in Model Panel Field or return to 4.19.9 Model Panel Field or 4.19 Panel Fields.

4.19.9.2 Adding Model to Views in Model Panel Field

12d Model panels frequently include a field for entering the name of a model. These fields are called **model panel** fields or simply model fields.

Because models are often required to be displayed on views and hence, need to be **added** to a view, there is a short-hand method for adding models to views when using a model field.

After the model name is entered into the model field, the name of the view that the model is to be added to is typed in, preceded by a comma. If the model is to be added to more than one view, simply type in each of the view names separated by commas.

For example, to add the model **fred** to the views **1**, **3**, and **5**, type

fred,1,3,5

into the model field.

Alternatively, the view names can be enclosed in round brackets.

For example,

fred (1,3,5)

Go to the next section 4.19.11 String Select Panel Field or return to 4.19.9 Model Panel Field or 4.19 Panel Fields.

4.19.10 Tin Panel Field

See

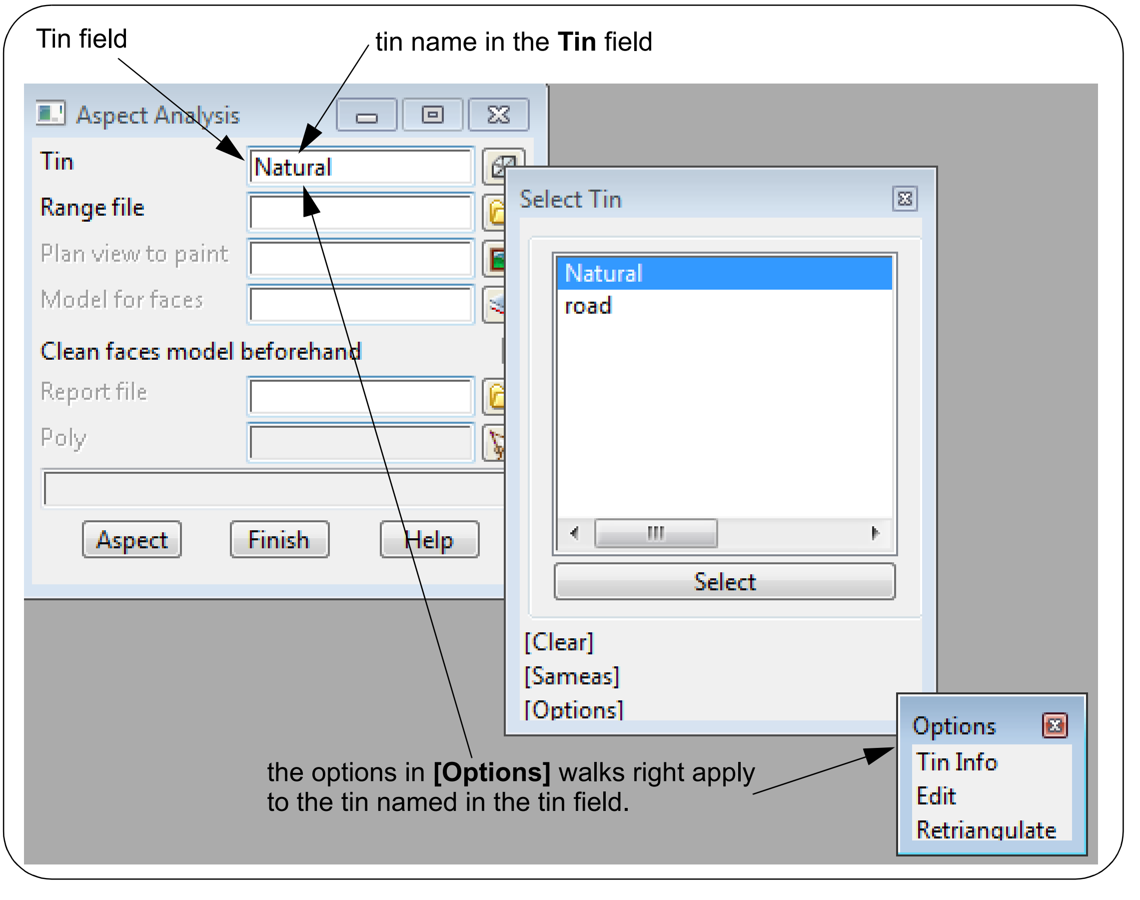
4.19.10.1 [Options] on Tin Box Pop Up

4.19.10.1 [Options] on Tin Box Pop Up

When you have a **Tin** field with a tin name of an existing tin in it, the **Select Tin** pop up for the **Tin** icon at the end of the **Tin** field has an **[Options]** at the bottom of the pop up with a walk right list of options that apply to the tin in the **Tin** field.

Note that the tin must exist so if the Tin field in a panel is one that creates the tin, then using **[Options]** will only make sense **after** the panel has been run and the tin has been created.

Warning
The options on **[Options]** apply to the tin in the **Tin** field, **NOT** the tin highlighted in the **Select Tin** pop up.



Tin Info - brings up the **Tin Information** panel and runs it for the tin given in the **Tin** field.

Retriangulate - runs retriangulate for the tin given in the **Tin** field.

Edit - brings up the **Retriangulate Tin** panel with the tin given in the **Tin** field set as the *Tin* and the information for that tin loaded into the panel.

Return to 4.19.9 Model Panel Field or 4.19 Panel Fields.

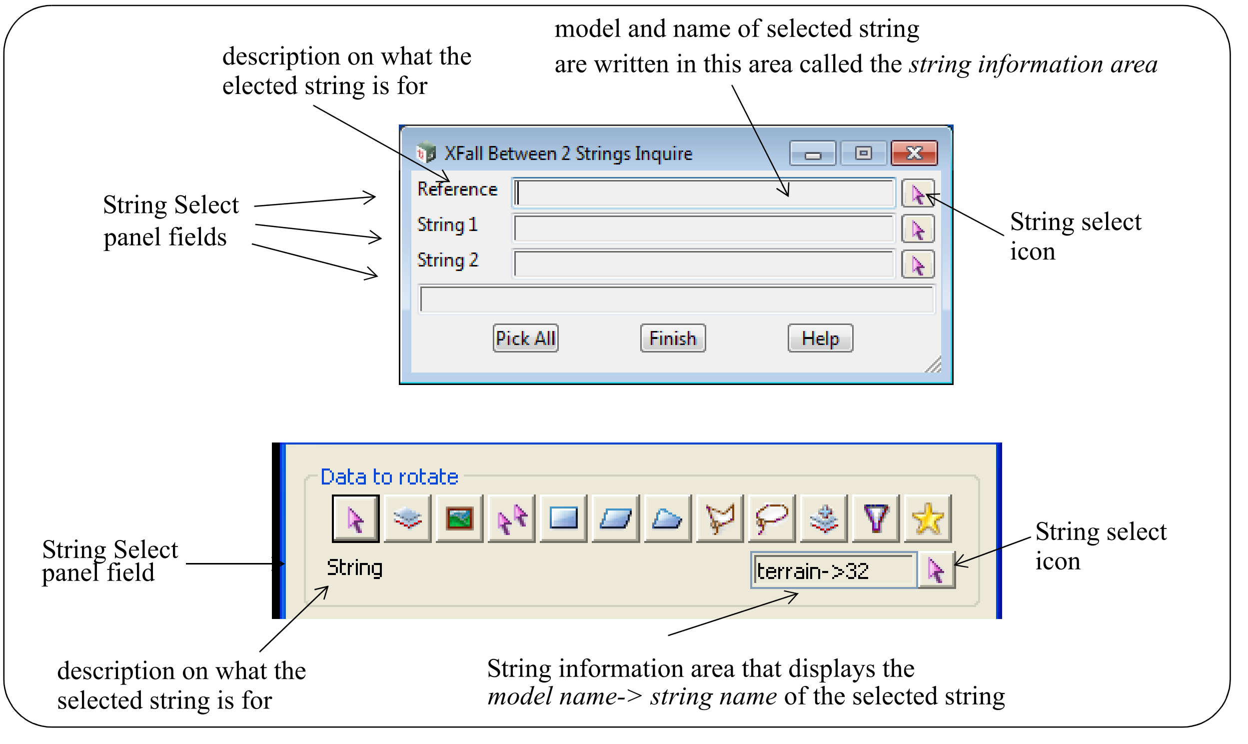
4.19.11 String Select Panel Field

There is a special panel field called a **string select** panel field which is used for **selecting** a string and **recording** the string and model name in the panel field. The string select panel field consists of

(a) a description of what the field is for

(b) n string information area where the model and string name of a selected string are displayed and

(c) the **string select** icon which is clicked on to start selecting a string.



Clicking LB on the *String Select* icon will start the string selection process.

Clicking MB on the *String Select* icon does nothing.

Clicking RB on the *String Select* icon brings up the Choice menu which has options to

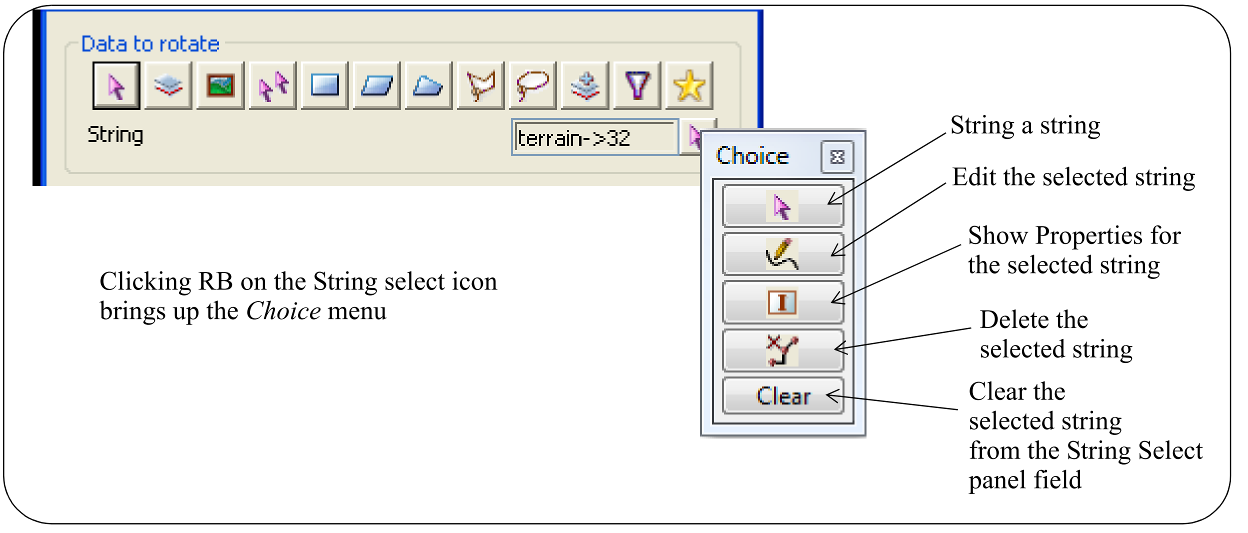
(a) Select a string (same as clicking LB on the *String Select* icon)

(b) Edit the string that has already been selected for the String Select field. The string to be edited will have its model name and string name shown in the string information area.

(c) Show the Properties panel for the string that has already been selected for the String Select field.

(d) Delete the string that has already been selected for the String Select field. The string to be deleted will have its model name and string name shown in the string information area.

(e) Clear the String Select panel of the string that had already been selected.



If a *String Select* operation has been started, it can be cancelled by clicking *RB* to bring up the **Pick Ops** menu and selecting **Cancel** from the menu.

Go to the next section 4.19.12 MB for Same As Pick for Panel Fields or return to 4.19 Panel Fields.

4.19.12 MB for Same As Pick for Panel Fields

For most panel fields/input boxes, clicking the middle mouse button (MB) in the field will activate a **Same As** pick which allows the user to get information from other objects.

(a) For a **View** panel field:

After clicking MB in the **View** panel field, you then select a string, plot frame *etc.* from a view and the **name** of the **View** that the string is on will be written to the **View** field

When a view name is selected this way, <Enter> must be pressed for the **View** field to validate and perform any other actions that the **View** field automatically does if the view name was selected from the **View** icon popup.

(b) For most panel field, other than a **View** panel field:

After clicking MB in the panel field, you then select a string, plot frame *etc.* from a view and the same value as the panel field type will be extracted from the selected item and piped into the panel field.

For example, for a **Model** field, the model that the selected item is in will be piped into the **Model** field.

When a panel field value is selected this way, <Enter> must be pressed for the panel field to validate and perform any other actions that the panel field automatically does when the panel field value was selected from the panel field icon popup.

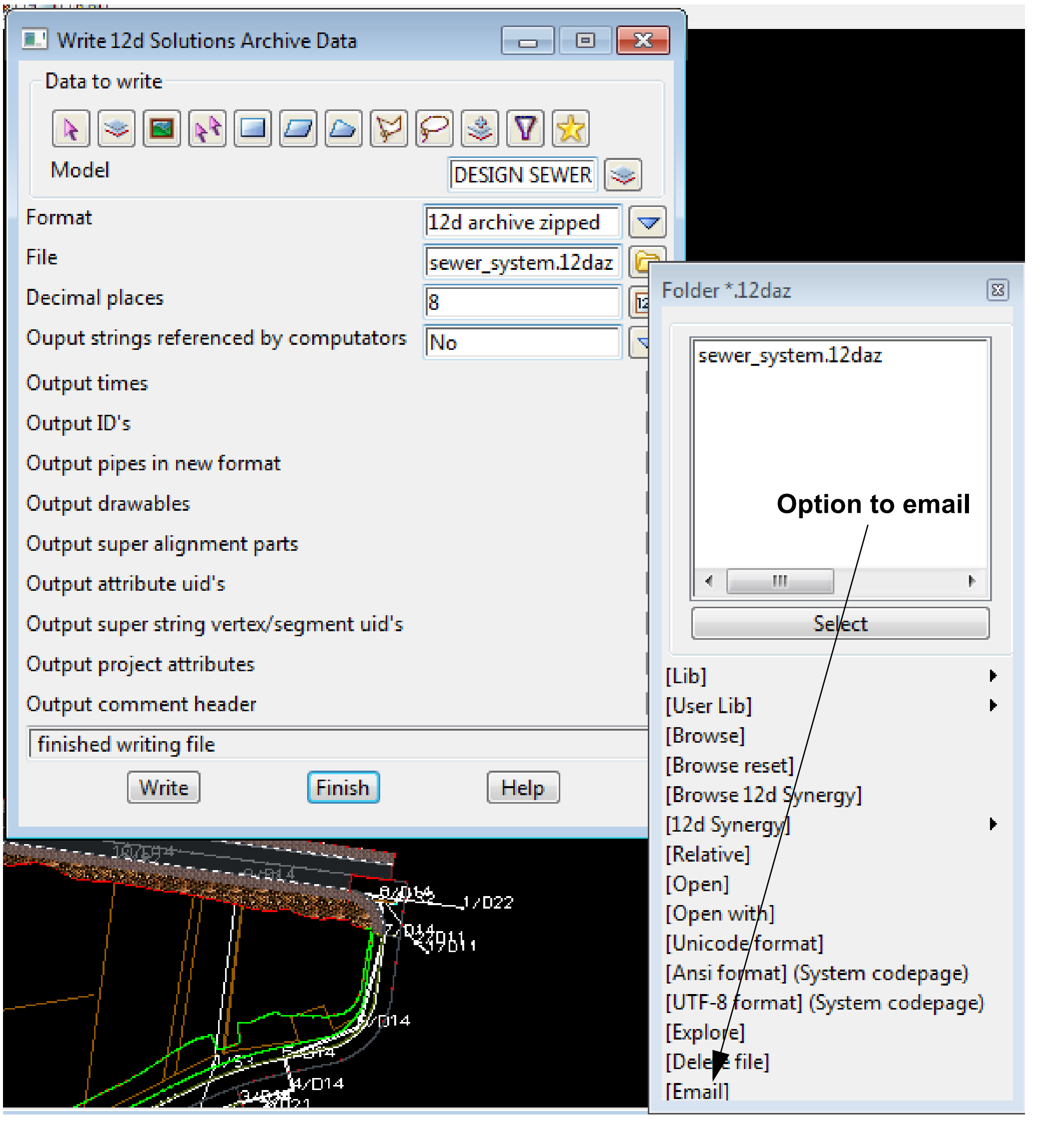
MB **Same As Pick** works for most panel fields including

Views
Models
Colours
Justify
Text styles
Text units
Names
Plotter (from plot frames)
Sheet size (from plot frames)
Line styles

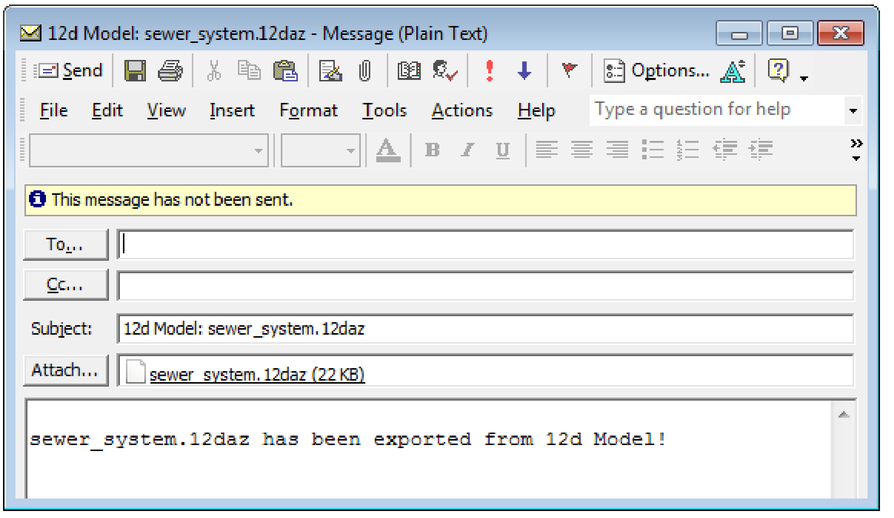
Go to the next section 4.20 Emailing from File Boxes or return to 4.19 Panel Fields.

4.20 Emailing from File Boxes

When a **File box** has a file name in it and the file exists (this usually means the option has to be run first), clicking on the **File** icon at the end of the field will bring up the **Folder** pop-up and at the bottom there is now an **[Email]** choice.



If you have *Outlook 2002* and above (or another **MAPI** registered application), clicking on **[Email]** will create an email with that file as an attachment ready for you to give the email address and any extra information you would like to put in the email.

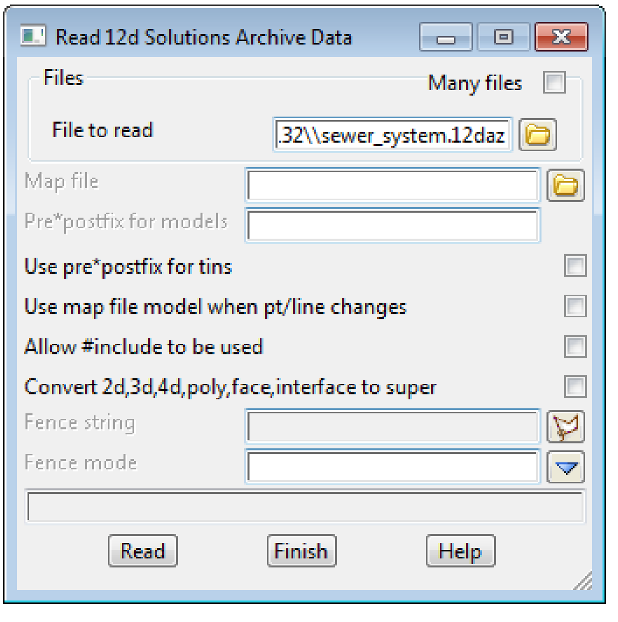


This will work with Outlook 2002 and above, and should also work with other **MAPI** registered applications.

In the above example, the 12d Archive File was written out as a zipped 12da file (a 12daz file).

Zipping the 12da file can result in files a twentieth of the size of the original 12da file (*i.e.* 5% of the original), which makes it much better for emailing around.

Also, when you receive a 12daz or a 12da file as an attachment to an email, you can drag and drop it onto an open 12d project and it will automatically open the **Read 12d Solutions Archive Data** panel with the dropped file name already filled in as the **File to read**.



Important Note

The file must already exist **before** the **[Email]** on the pop up can be used so usually the Panel option must be run first to create the file, and then go back and click on the **File** icon to bring up the pop up for the file to be emailed.

Please continue to the next section 4.21 Measures.

4.21 Measures

Whenever a real value is required in a box (e.g. height, width), the pop-up menu on the **[+]** button includes a **measures** menu.

The **Measures** menu allows the user to pick a selected measure value from existing strings, and the picked value is displayed and piped into the panel field.

For example, the x, y, or z coordinate of a point or the plan distance between two points.

If default values already exist for the pop-up menu, the options on the **measures** menu will be added to the bottom of the default list.

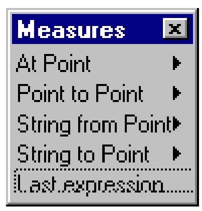
The measures functions support dynamic measures to give the user more visual feedback. This includes the drawing of a rubber line in the case of **Point Point** and **String to Point** measures, or a cross in the case of **At Point.**

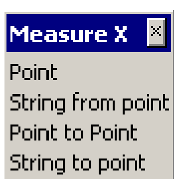
The measures and their dynamic aspect can be easily explored via the option

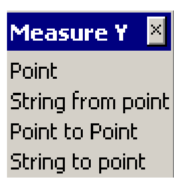
Utilities=>Measure=>Value

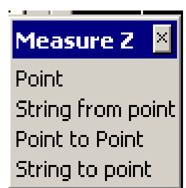
The walk-right **Measure** menus are documented in the next four sections.

The stand alone **Measures** menus is









For the option At point, go to 4.21.1 Measure At Point
Point to point 4.21.2 Measure Point to Point
String from point 4.21.3 Measure String from Point
String to point 4.21.4 Measure String to Point
Last expression 4.16 Last Expression

If information on measures is not required, please continue to the next section 4.16 Last Expression.

4.21.1 Measure At Point

get x co-ordinate of selected point
y co-ordinate
z co-ordinate
chainage at the point
instantaneous angle
instantaneous bearing
cross-fall of selected point
slope at the point
radius (0 if no curve)

The **At Point** measures calculate and display the picked measure at a selected point.

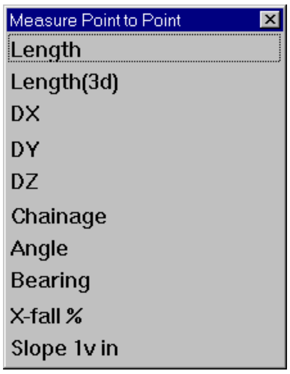
For example, **X** gets the x-coordinate of a selected point.
Bearing gets the instantaneous bearing of a selected point.

The dynamic aspect of the **at point** measures is as follows:

> after the desired **at point** option is picked and the string for the **at point** measure is tentatively selected but not yet accepted, and if the cursor is moved around, then the cursor position is dynamically dropped perpendicularly onto the highlighted string and a cross displayed on the string at the dropped point, and the picked measure value is dynamically displayed in the panel field.

For example, if the **x** measure was picked from the **Measure Point** menu and a string highlighted but not yet accepted, then as the cursor is moved around the screen, the cursor position is dropped perpendicularly onto the highlighted string and displayed as a cross and the x-coordinate of the cross is displayed in the panel field.

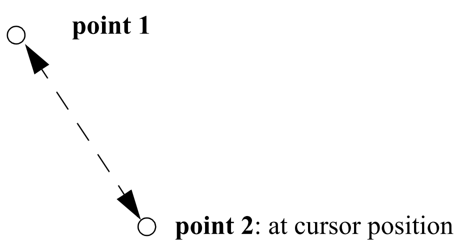
4.21.2 Measure Point to Point



get 2d length between selected points
3d length
delta x
delta y
delta z
delta chainage
angle of line joining selected points
bearing of line
cross-fall of line
slope of line

The **Point to Point** measures calculate and displayed the picked measure between two selected points.

For example,	**Length**	gets the plan distance between two selected points.
	X-fall %	gets the % cross fall of the line joining two selected points.



The **point to point** measures are calculated between point 1 and point 2

The dynamic aspect of the **point to point** measures is as follows:

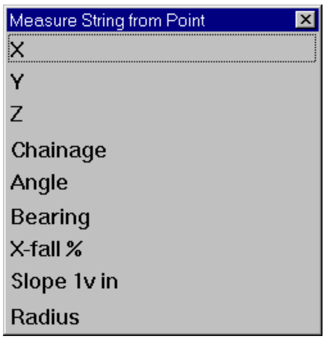
after the desired **point to point** measure is picked, and the first point for the **point to point** measure is selected, then a rubber line is dynamically drawn between the selected point and the current cursor position and the picked measure value is dynamically displayed in the panel field.

Further, if the selected string for the **second** point is tentatively selected but not yet accepted, and if the cursor is moved around, then the cursor position is dynamically dropped perpendicularly onto the highlighted string and the picked measure value to the second string is dynamically displayed in the panel field.

For example, if the **length** measure was picked from the **Measure Point to Point** menu and the first point selected, then a rubber line is dynamically drawn between the selected point and the cursor position and the plan length from the selected point to the current cursor position is dynamically displayed in the panel field.

If a string is selected for the second point but not yet accepted, the cursor position is dynamically dropped onto the highlighted string and the plan distance from the first point to the dropped point is dynamically displayed in the panel field.

4.21.3 Measure String from Point



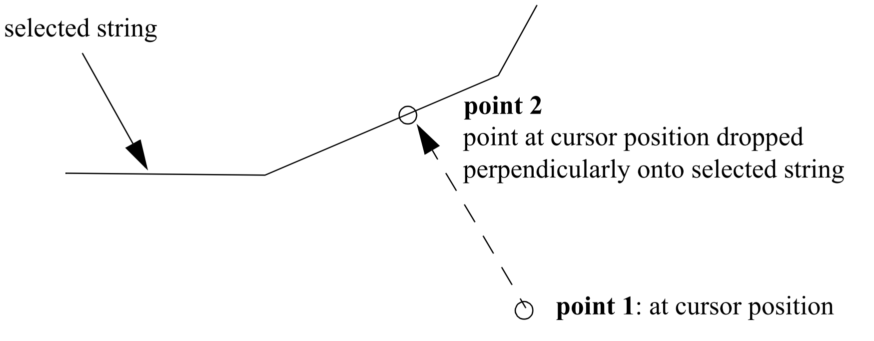
get x co-ordinate of point dropped onto string
y co-ordinate
z co-ordinate
chainage at the point
instantaneous angle
instantaneous bearing
cross-fall of selected point
slope at the point
radius (0 if no curve)

The **String from Point** measures calculate the picked measure *from the selected string* by taking the current cursor position and dropping it perpendicularly onto the selected string and using the dropped point on the string to get the measure value.

For example, **X** gets the x-coordinate from the string of the cursor dropped onto the selected string.

Z gets the z-coordinate from the string of the cursor dropped onto the selected string

Bearing gets the instantaneous bearing from the string of the cursor dropped onto the selected string.



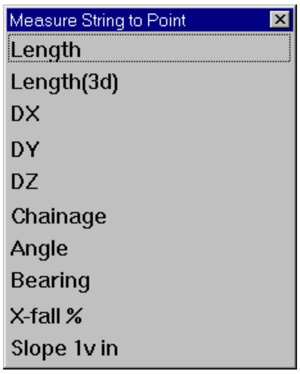
The **string from point** measures are calculated at point 2

The dynamic aspect of the **string from point** measures is as follows:

after the desired **string from point** measure is picked and the string for the **string from point** measure is selected, then as the cursor is moved around, a rubber line is dynamically drawn between the cursor and the cursor position dropped perpendicularly onto the selected string, and the picked measure value of the dropped cursor position is dynamically displayed in the panel field.

For example, if the **z** measure was picked from the **Measure String from Point** menu and a string selected, then as the cursor is moved around the screen, the cursor position is dropped perpendicularly onto the selected string. The perpendicular line is dynamically redrawn and the z-coordinate of the dropped point on the string is dynamically displayed in the panel field.

4.21.4 Measure String to Point

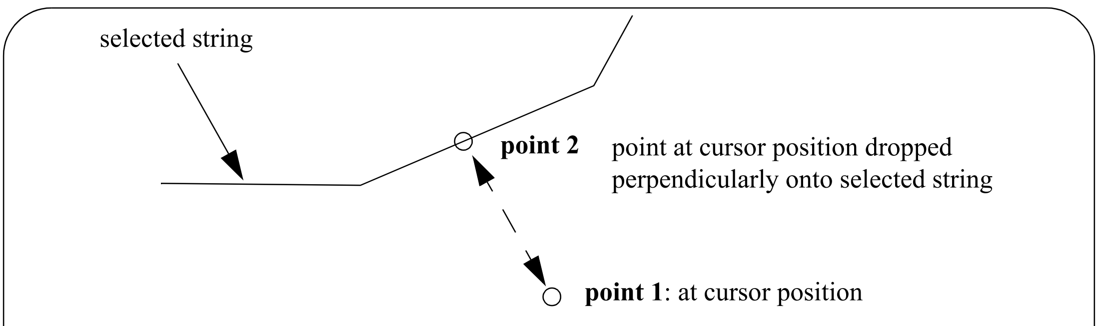


get 2d length between selected points
3d length
delta x
delta y
delta z
delta chainage
angle of line joining selected points
bearing of line
cross-fall of line
slope of line

The **String to Point** measures are similar to the *point to point* measures and calculate the picked measure from a point and the point ***dropped perpendicularly*** onto the selected string.

For example, **Length** gets the plan distance between the point and the point dropped perpendicular on a selected string.

Bearing gets the bearing of line joining the point and the point dropped perpendicularly onto a selected string



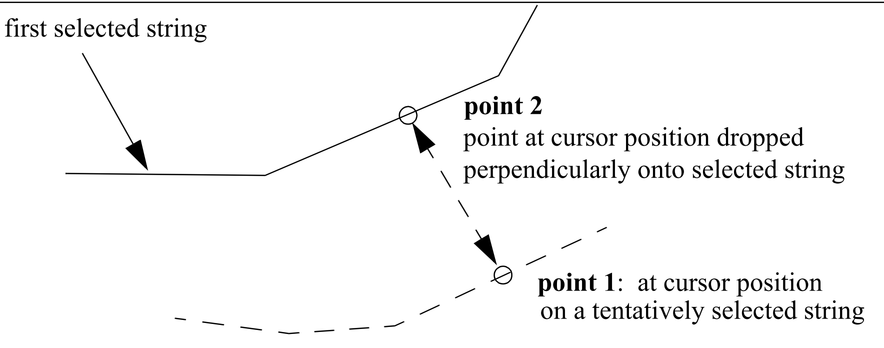
The **string to point** measures are calculated between point 1 and point 2

The dynamic aspect of the **string to point** measures is as follows:

after the desired **string to point** measure is picked, and a *string* selected, then a line is dynamically drawn between the cursor position and the cursor position *dropped perpendicularly* onto the selected string. The **string to point** value is dynamically displayed in the panel field as the cursor is moved around.

Further, if the cursor position is on a string which has been selected but not accepted and the cursor is moved around, then the cursor is *constrained* to remain on the tentatively selected string. It is dynamically dropped perpendicularly onto the original string and the measure value to the dropped point displayed in the panel field.

Hence the selected point is ***constrained*** to remain on the tentatively picked string and the measure dynamically calculated and displayed.



The **string to point** measures where point 1 is constrained to be on a tentatively selected string.

For example, if the **length** measure was picked from the **Measure String to Point** menu and the first string selected, then a line is dynamically drawn perpendicularly between the selected string and the current cursor position and the plan length from the string to the current cursor position is dynamically displayed in the panel field.

If a string is tentatively selected (picked but not yet accepted), the cursor position is dynamically dropped onto the first string and the plan distance measured perpendicularly from the point to the first string is dynamically displayed in the panel field.

Please continue to the next section 4.16 Last Expression.

Please continue to the next section 4.22 Colours.

4.22 Colours

See

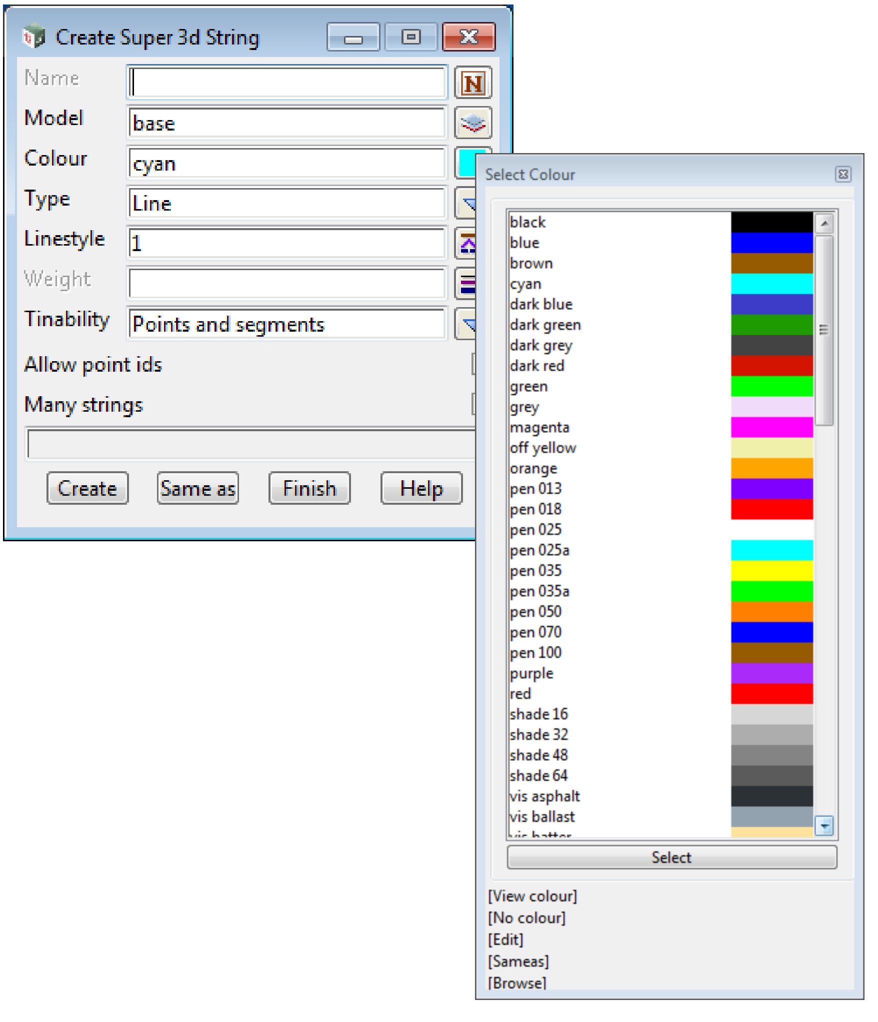
[4.22.1 Colours on the Screen](#)
[4.22.2 Colours on a Plot](#)

4.22.1 Colours on the Screen

For ***12d Model*** elements, there is a palette of up to 10,240 colours that can be used for displaying elements and parts of elements on a view. Each of the colours has a unique number, the colour number, and it is the ***colour numbers*** that are **saved** in the ***12d Model*** database.

How each colour number appears when drawn on a view is defined by giving the values of the intensities of red, green, and blue (rgb) for the colour number.

When a panel requests a colour using a colour box, clicking on the colour icon will bring up the **Select Colour** choice box to choose a colour.



What the rgb is for a colour number, its text name that is shown in the **Select Colour** choice box and other properties are defined in the ***colours.4d*** file.

The *colours.4d* file is read in whenever a project is opened so it is only possible to use a new *colours.4d* file by restating a project. A default *colours.4d* file is distributed with ***12d Model*** and is in the *set_ups* folder.

Note: for viewing, creating and editing a *colours.4d* file, see 4.22.3 Editing Colours.4d and for the actual definition of the *colours.4d* on disk, see 39.2.7.2 Colours File (colours.4d).

Colour Box Panel Field

Either colour numbers or the unique colour name associated with the colour number, can be typed in any Colour box in a panel field. The Colour box converts a colour name to its colour number for storing in the *12d* ***Model*** database.

Limiting the Number of Colour Shown in Select Colours

When a large number of colours have been defined, the **Select Colour** pop-up list can get unwieldy so there is a default parameter, **Display colours** which gives the number of colours to use in a pop-up. **Display colours** is on the **System Settings** tab of the **Defaults** panel which brought up by **Project=>Management=>Defaults**. See 7.6.1 Defaults.

Hence it is possible to have thousands of defined colour numbers but still have only a selected number displayed in the **Colour** pop-ups.

The order that the colours are selected from the *colours.4d* file for the pop-up is ***not*** by the colour number but the ***Pop-up number*** *order* which can be completely different. This means that the user can add extra colours after the standard 2000 ***12d Model*** colours number and have the new colours coming up in the *Colours* pop-up even thought it is restricted on how many colours. The Pop-up number is given in the *colours.4d* file.

Order in the Select Colours Pip-Up

Finally the order that the colours are sorted in the Colours pop-up can be by pop-up number or alphabetically by colour name. See 4.22.3.1 Colours Sort Order in Select Colour Pop-Up.

Note:

In ***12da*** files, colours are written out by their *colour names,* ***not*** by *colour numbers. S*o when reading in a 12da file, the colour name in the 12da file gets mapped to the colour number that matches it in the ***12d Model*** project that the *12da* file is being read into.

For viewing, creating and editing a *colours.4d* file, see 4.22.3 Editing Colours.4d and for the actual definition of the *colours.4d* on disk, see 39.2.7.2 Colours File (colours.4d).

For more information on panel file pop-ups, see 4.19.5 Pop-Up Lists and Menus.

Continue to the next section 4.22.2 Colours on a Plot or return to 4.22 Colours.

4.22.2 Colours on a Plot

How a *colour* is drawn on a view is one thing, but how it appears on a plot can be completely different. And how the colours appears on one plot may be different for another plot. For example, all the strings on a view are in colour and you produce a coloured plot but you may also want to produce a and black and white plot of exactly the same data.

These cases are covered by defining different *12d* ***Model*** plotters where plotter has a unique name, a plot engine to say what type of device it is being plotted to, and colour to pen mappings to specify what the rgb and thickness is used for a pen.

All the*12d* ***Model*** plotters are defined in the *plotters.4d* file.

The *plotters.4d* file is read in whenever a project is opened so it is only possible to use a new *plotters.4d* by restating a project. A default plotters.4d is distributed with ***12d Model*** and is in the *set_ups* folder.

For more information *12d* ***Model*** plotters and plotters.4d, see 43.1 12d Model Plotters.

For more information on creating and editing a *colours.4d* file, see 4.22.3 Editing Colours.4d and for the actual definition of the *colours.4d* on disk, see 39.2.7.2 Colours File (colours.4d).

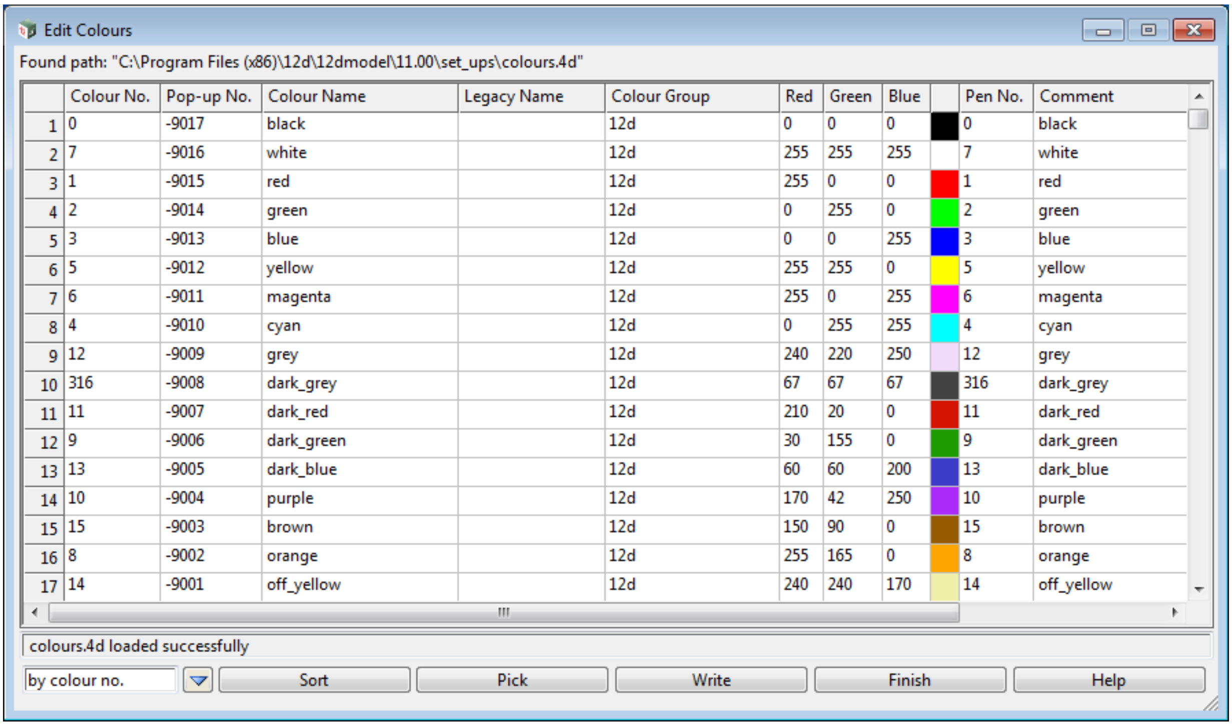
Return to 4.22 Colours.

4.22.3 Editing Colours.4d

For each colour number, the red, green, blue (rgb) for each colour number, an alternate text name, pop-up number and pen number are defined in the *colours.4d file.* Up to 10,240 colours can be defined. See 39.2.7.2 Colours File (colours.4d)

A default file is distributed with ***12d Model*** and resides in the *set_ups* folder.

To view, add or modify the definitions for the colour numbers in *colours.4d*, use the **Edit Colours** panel which is brought up by clicking on the **[Edit]** section of the **Select Colours** pop-up for a colour box, or by the option ***Plots =>Plotting setups =>Pen mapping***.



	Colour No.	Pop-up No.	Colour Name	Legacy Name	Colour Group	Red	Green	Blue		Pen No.	Comment
1	0	-9017	black		12d	0	0	0		0	black
2	7	-9016	white		12d	255	255	255		7	white
3	1	-9015	red		12d	255	0	0		1	red
4	2	-9014	green		12d	0	255	0		2	green
5	3	-9013	blue		12d	0	0	255		3	blue
6	5	-9012	yellow		12d	255	255	0		5	yellow
7	6	-9011	magenta		12d	255	0	255		6	magenta
8	4	-9010	cyan		12d	0	255	255		4	cyan
9	12	-9009	grey		12d	240	220	250		12	grey
10	316	-9008	dark_grey		12d	67	67	67		316	dark_grey
11	11	-9007	dark_red		12d	210	20	0		11	dark_red
12	9	-9006	dark_green		12d	30	155	0		9	dark_green
13	13	-9005	dark_blue		12d	60	60	200		13	dark_blue
14	10	-9004	purple		12d	170	42	250		10	purple
15	15	-9003	brown		12d	150	90	0		15	brown
16	8	-9002	orange		12d	255	165	0		8	orange
17	14	-9001	off_yellow		12d	240	240	170		14	off_yellow

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Grid Cells

Colour No. integer

number of the colour. A unique integer value between 0 and 10,239.

Pop-up No. integer

a unique integer - positive, 0 or negative.

The number of colours in the colours pop-up can be restricted (see 7.6.1 Defaults*) and if this is the case, the colours are selected for display in the pop-up by the pop-up number.*

Also the pop-up number can be used as the sorting order for the colours in the colours pop-up. See 4.22.3.1 Colours Sort Order in Select Colour Pop-Up.

Colour Name text

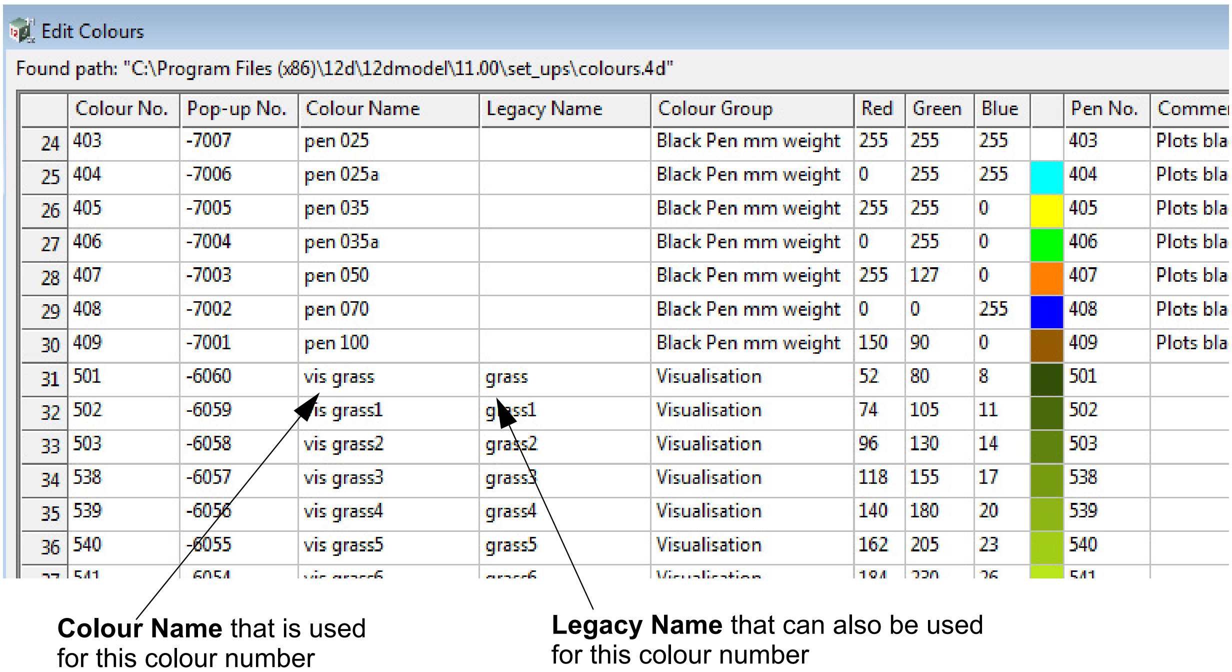
text name for the colour. The colour names must all be unique within the Colours.4d file.

Colour names are restricted to having alphanumeric characters and decimal points (.*). Upper and lower case characters can be used but colour names are case insensitive for uniqueness checks. So Red is considered the same as RED.*

Legacy Name text

*whenever a colour name is searched for, the **Legacy Name** column will also be searched for if there is no match with the **Colour Name** column.*

*The **Legacy Name** was introduced so that some 12d Model colours can be given new names without upsetting the colour number that went with the original name.*


Edit Colours

Found path: "C:\Program Files (x86)\12d\12dmodel\11.00\set_ups\colours.4d"

	Colour No.	Pop-up No.	Colour Name	Legacy Name	Colour Group	Red	Green	Blue		Pen No.	Commer
24	403	-7007	pen 025		Black Pen mm weight	255	255	255		403	Plots bla
25	404	-7006	pen 025a		Black Pen mm weight	0	255	255		404	Plots bla
26	405	-7005	pen 035		Black Pen mm weight	255	255	0		405	Plots bla
27	406	-7004	pen 035a		Black Pen mm weight	0	255	0		406	Plots bla
28	407	-7003	pen 050		Black Pen mm weight	255	127	0		407	Plots bla
29	408	-7002	pen 070		Black Pen mm weight	0	0	255		408	Plots bla
30	409	-7001	pen 100		Black Pen mm weight	150	90	0		409	Plots bla
31	501	-6060	vis grass	grass	Visualisation	52	80	8		501	
32	502	-6059	vis grass1	grass1	Visualisation	74	105	11		502	
33	503	-6058	vis grass2	grass2	Visualisation	96	130	14		503	
34	538	-6057	vis grass3	grass3	Visualisation	118	155	17		538	
35	539	-6056	vis grass4	grass4	Visualisation	140	180	20		539	
36	540	-6055	vis grass5	grass5	Visualisation	162	205	23		540	
37	541	-6054	vis grass6	grass6	Visualisation	184	230	26		541	

Colour Group text

group for the colour. This is not currently used.

Red/Green/Blue integer

amount of red/green/blue in the colour when it is displayed on a view. The value is between 0 and 255.

Pen No. text box

the default pen number to map the colour to.

Comment text box

optional comment.

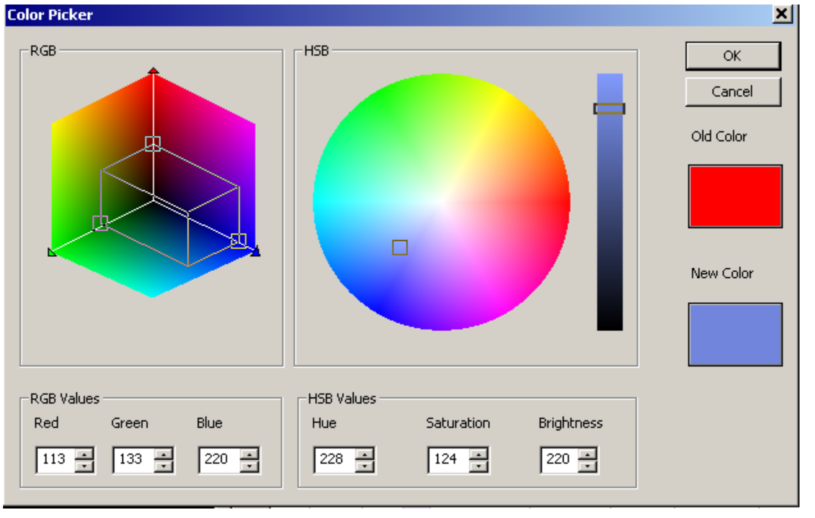
Buttons

Sort button

sort the table with respect to the column given in the adjacent field.

Picker button

if a row of the table is highlighted, selecting **Picker** *brings up the* **Color Picker** *to interactive define the RGB for new colours.*



Write button

write the table out to a colours.4d file.

WARNING

Do not change the colour numbers and definitions for colour numbers 0 to 2000 - they are restricted for ***12d Solutions*** use only.

In the ***12d Model*** database, only the colour number is saved so if the rgb of any colour numbers are changed then the display of colours in existing projects may change.

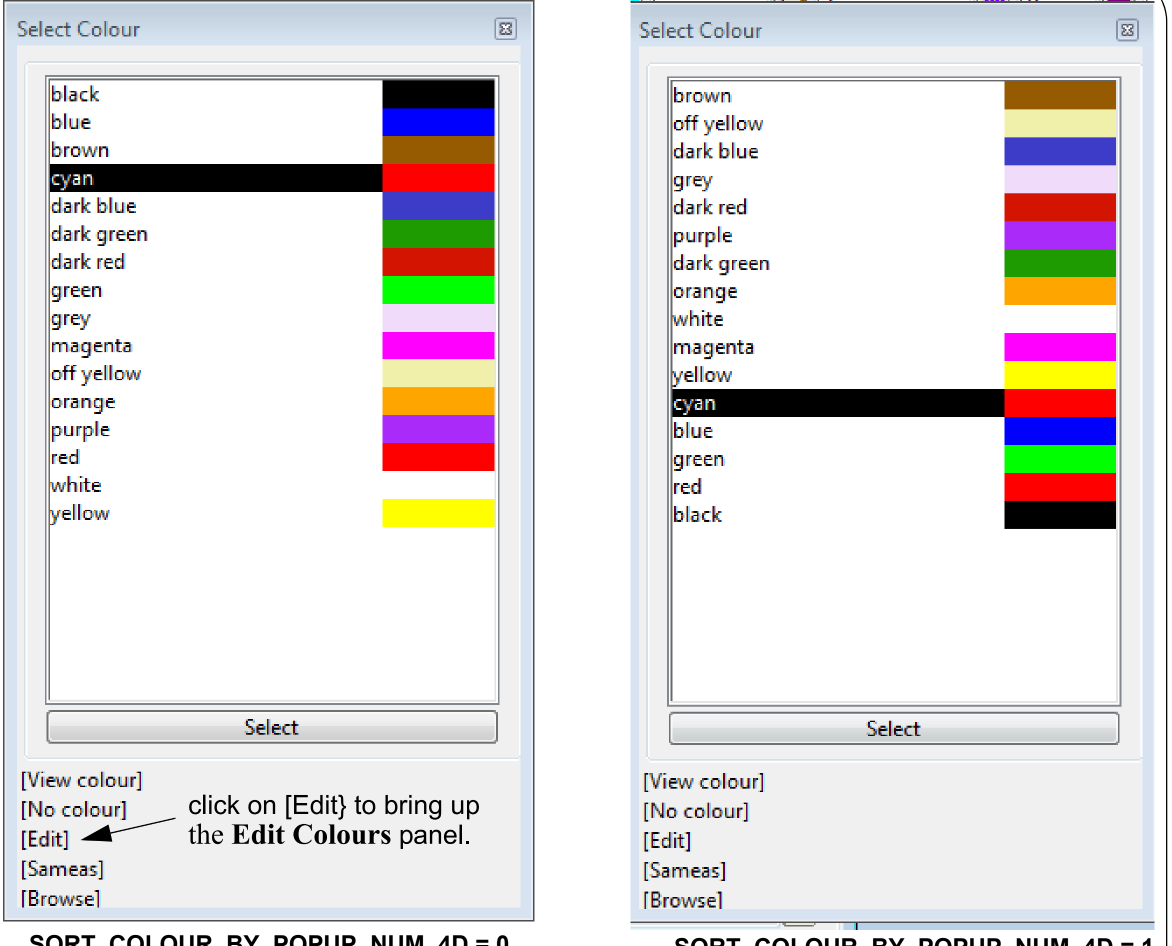
Please continue to the next section 4.22.3.1 Colours Sort Order in Select Colour Pop-Up or return to 4.22 Colours.

4.22.3.1 Colours Sort Order in Select Colour Pop-Up

There is an environment variable SORT_COLOURS_BY_POPUP_NUM_4D that controls the display order of the colours in the **Select Colour** popup that you get when you click on a **Colour** icon:

If **SORT_COLOURS_BY_POPUP_NUM_4D** is **1,** the colours in the **Select Colour** popup are sorted by the Pop-up Number given for the colours in the **Colours.4d** file.

If **SORT_COLOURS_BY_POPUP_NUM_4D** is **0,** the colours in the **Select Colour** popup are sorted alphabetically.

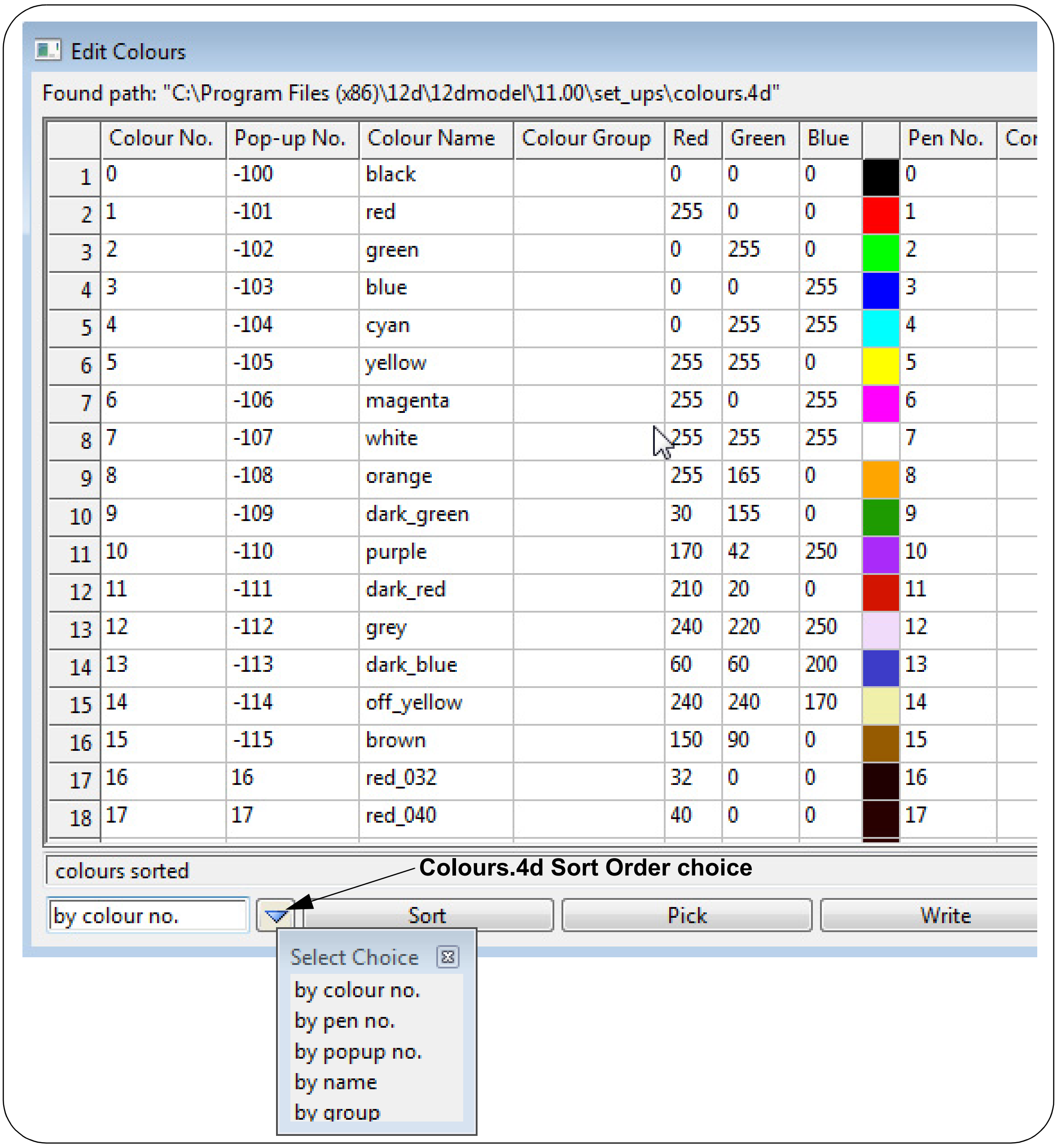


SORT_COLOUR_BY_POPUP_NUM_4D = 0

SORT_COLOUR_BY_POPUP_NUM_4D = 1

Note - Remember that the number of colours shown in the **Select Colour** popup is controlled by the ***Display colours*** value in the **Defaults** panel.

To edit the **Colours.4d** file to edit the popup number, click on **[Edit]** in the **Select Colour** popup. You can then edit the Pop-Up Number column and after any changes are made and write out the new **Colours.4d** file.



	Colour No.	Pop-up No.	Colour Name	Colour Group	Red	Green	Blue		Pen No.	Co
1	0	-100	black		0	0	0		0	
2	1	-101	red		255	0	0		1	
3	2	-102	green		0	255	0		2	
4	3	-103	blue		0	0	255		3	
5	4	-104	cyan		0	255	255		4	
6	5	-105	yellow		255	255	0		5	
7	6	-106	magenta		255	0	255		6	
8	7	-107	white		255	255	255		7	
9	8	-108	orange		255	165	0		8	
10	9	-109	dark_green		30	155	0		9	
11	10	-110	purple		170	42	250		10	
12	11	-111	dark_red		210	20	0		11	
13	12	-112	grey		240	220	250		12	
14	13	-113	dark_blue		60	60	200		13	
15	14	-114	off_yellow		240	240	170		14	
16	15	-115	brown		150	90	0		15	
17	16	16	red_032		32	0	0		16	
18	17	17	red_040		40	0	0		17	

Note - The sort order used when displaying the **Colours.4d** file has no effect on the order in the **Select Colour** popup.

To see the **Colours.4d** file sorted in different ways, click on the **Sort Order** choice pop-up, select the required sort order and then click on the **Sort** button.

You need to do a **Project Restart** for any changes in **Colours.4d** to take effect.

Please continue to the next section 4.23 Plotters or return to 4.22.3 Editing Colours.4d or 4.22 Colours.

4.23 Plotters

When creating a **plot**, 12d Model has a variety of options about where the potting data goes to. The user can:

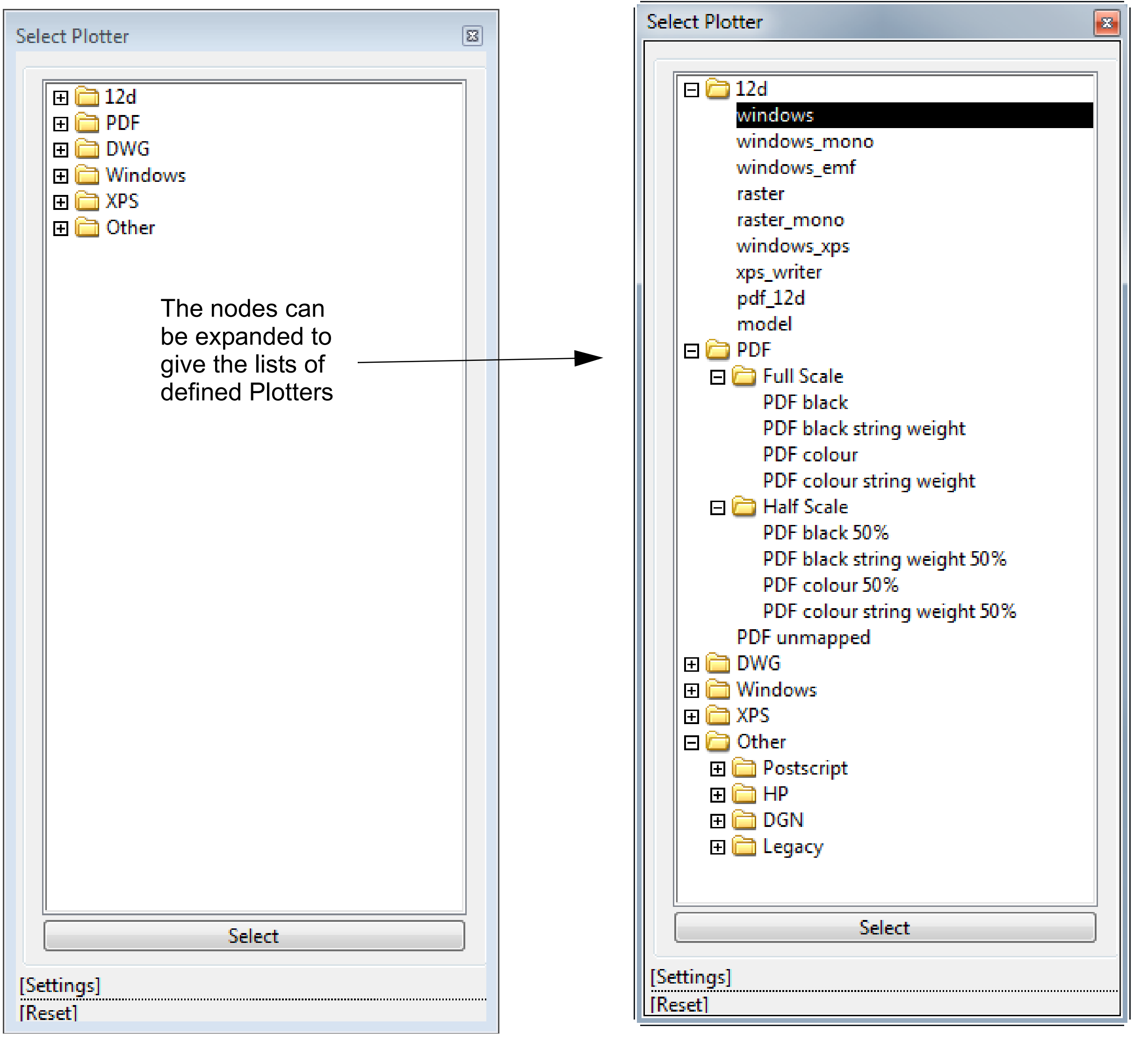
(a) drive the plotters **directly** using Windows printer drivers or raster plotter drivers

(b) plot **indirectly** by creating a computer disk file (the plot file) in a variety of formats containing the relevant plotter instructions for producing the plot. The formats include pdf, XPS, DWG and DGN.

(c) plot **indirectly** to a ***12d Model* model**

In all of these cases, their are **plotter mapping files** that control how the colours appear on the plot and with what weights. For example, for some plots, you may want all the colours to be mapped to black and shades of grey rather than actual colours. There are plotter mapping files that are installed with ***12d Model*** and users can define their own.

The choice of the device to plot to plot to and the plotter mapping file go together to define a **Plotter** and in plotting options, the choice of plotters that is available is shown in the **Select Plotter** pop-up menu. The default ***12d Model*** list is



From the list you will see that there can be a number of Plotters using the same output type (e.g. windows or pdf) but there are different entries in the list depending on a particular plotting mapping file (pmf) is being used.

For example:

Full Scale >PDF black

The 12d standard **colour numbers 1-15** & **316** plot with **0.25mm black** pens.

The "**pen***" (**400 series**) colour numbers plot with **black pens** of **specific weights**.

All "**ppf***" (**900 series**) colour numbers plot with black/grey scale pens of specific weights.

Full Scale >PDF black string weight

If the string has no weight then it is the same as **Full Scale >PDF black** but if the string has a weight then the string weight is used.

So if a non standard weight is required for a particular string then you set the weight for that string. Otherwise do not give the string a weight.

When the plot file is created, it can be passed to a user specified program or batch file which can automatically direct the plot file to a plotter. See 43.4 Sending Plots to a Plotter.

For more information on plotters and defining plotters, see 43 Plotters and Plotting.

The file *plotters.4d* is described in 43.2 Defining Plotters - Plotters.4d.

The *plotter mapping file* is described in 43.3.2 Plotter Mapping File.

Please continue to the next section 4.24 Defaults.

4.24 Defaults

4.24.1 Defaults

Many options in 12d Model require settings which are almost always the same. To save repetitous typing, *12d Model* provides a number of user defined defaults that are used throughout *12d Model*.

For example, the chord to arc tolerance, point cross size and highlight cross size and colour.

Certain default values are read in from a file when a new project is created. These defaults can also be modified inside 12d Model using the **Defaults** panels.

The lay out of the file used to define the default values for a new project is given in the appendix **Set Ups**.

User defaults can also be set for entire panels - see the section 4.3.6.6 Panel Defaults - ddx Files.

Go to the next section 4.24.2 Default Panel Values or return to 4.24 Defaults.

4.24.2 Default Panel Values

When a panel is opened, ***12d Model*** may set some default values.

However for most panels ***the user can supply the values*** to be used ***for*** all the ***panel fields*** each time a panel is opened. This information is stored in a defaults file for a panel.

For more information on the panel default files and how to create them, go to the section 4.3.6.6 Panel Defaults - ddx Files.

Go to the next section 4.24.3 Default File Ending or return to 4.24 Defaults.

4.24.3 Default File Ending

In any panel pop-up requiring a file name, default file endings are used to restrict the names of the files selected from the current folder to be displayed in the pop-up.

The list of special file endings is given in the section 42.1 Default File Ending in the appendix 42 Special File Formats.

Return to 4.24 Defaults or continue to the next section 4.25 Miscellaneous Panels.

4.25 Miscellaneous Panels

The value is entered into the typed-input box, terminated with <enter>.

The **Enter Value** typed-input box then disappears.

The length is entered into the typed-input box, terminated with <enter>.

The **New Length** typed-input box then disappears.

The radius is entered into the typed-input box, terminated with <enter>.

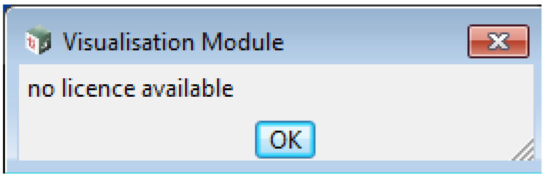
The **New Radius** typed-input box then disappears.

4.26 No Option Available

12d Model consists of a base module of varying point sizes and optional modules.

If a module has not been purchased and any options included in that option are selected then a ***No licence available*** message will be displayed.

The missing module can be purchased by contacting your ***12d Model*** distributor or **12d Solutions Pty Ltd.**



4.27 No Information Available

There is no information available on this panel or menu.

4.28 Options on Toolbars

For general information *Toolbars* and *Controlbars*, go to 4.3.5 Toolbars and Controlbars

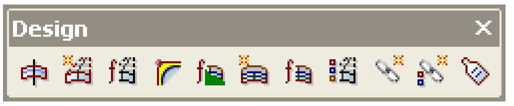
Each *Toolbars* is described in detail in:

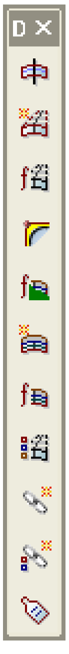
Cad	15.2 CAD Options
Cad Arcs	15.13 CAD Arc
Cad Circles	15.12 CAD Circle
Cad Controlbar	15.1 Controlbars
Cad Delete	15.25 CAD Delete
Cad Fills	15.18 CAD Fill
Cad Holes	15.17 CAD Hole
Cad Images	15.19 CAD Image
Cad Intersect	15.10 CAD Intersection
Cad Lines	15.11 CAD Line
Cad Modify	15.20 CAD Modify
Cad Points	15.9 CAD Point
Cad Polygons	15.14 CAD Polygon
Cad Segment	15.22 CAD Segment
Cad String	15.23 CAD Edit Strings
Cad Symbols	15.16 CAD Symbol
Cad Text	15.15 CAD Text
Design	4.28.1 Design Toolbar
Cad Vertex	15.21 CAD Vertex
H	4.3.5.3 Snaps (H) Toolbar
Measure edits	4.28.2 Measure Edits Toolbar
Menu Bar	can't be turned off - all the options in ***12d Model***
Options	turns off the area for docking toolbars on left hand side of ***12d Model*** window
Road	4.28.3 Road Toolbar
Search bar	15.1.6 Search Bar
Snaps Cad	4.28.4 Snaps Cad
Snaps Cad Arcs	4.28.5.4 Snaps Cad Arc
Snaps Cad Circles	4.28.5.3 Snaps Cad Circle
Snaps Cad Intersect	4.28.5.1 Snaps Cad Intersection
Snaps Cad Lines	4.28.5.2 Snaps Cad Line
Snaps Cad Points	4.28.5 Snaps Cad Point
Strings Edits	4.28.6 String Edits Toolbar
Super alignment tools	4.28.7 Super Alignment Tools Toolbar
Survey reductions	4.28.8 Survey Reductions Toolbar
Symbol Controlbar	15.1.2 Symbol Controlbar
Text Controlbar	15.1.3 Text Controlbar
Tin Utility	4.28.9 Tin Utility Toolbar
Tin Utility	4.28.9 Tin Utility Toolbar
Track	4.28.12 Track Toolbar
Visualisation	4.28.11 Visualisation Toolbar

4.28.1 Design Toolbar

Position of option on menu: various

The Design Toolbar is:





Options on Toolbar

template create/edit

create mtf file

apply templates function

apply kerb return function

interface function

create boxing file

boxing many

apply templates defaults

create/edit chain

chain parameters editor

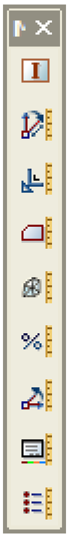
tag tree

For Template Create/Edit, go to	20.2.2 Create/Edit Templates
Create MTF File	20.4.1 Create MTF
Apply Templates Function	20.3.2 Apply MTF Function
Apply Kerb Return Function	20.3.5 Kerb Return Function
Interface Function	20.3.7 Interface Function
Create Boxing File	20.5.1 Create Boxing Definitions
Boxing Many	20.5.4 Boxing Many Function
Apply Templates Defaults	20.3.1 Apply Template Function
Create / Edit Chain	28.3.1 Create/Edit a Chain
Boxing Many	20.5.4 Boxing Many Function
Chain Parameters Editor	28.3.5 Creating/Editing Parameter Value Files
Tag Tree	7.6.10 Tags

4.28.2 Measure Edits Toolbar

Position of option on menu: **various**

The **Measure Edits** toolbar is:



Options on Toolbar

string inquire

measure bearing/distance

measure value

measure plan area

measure surface area

measure xfall

measure angle by 3 points

multi string properties

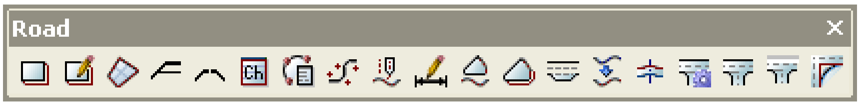
measure menu

For *String inquire,* go to	14.11 Inquire
Measure bearing/distance	28.6.2 Bearing and Distance
Measure value	28.6.5 Value
Measure plan area	28.6.3 Plan Area
Measure surface area	28.6.4 Surface Area
Measure xfall	28.6.8 X Fall by Strings
Measure angle	28.6.1 Angle by 3 Points
Match properties	14.12.1 Multi String Properties
Measure menu	28.6 Measure

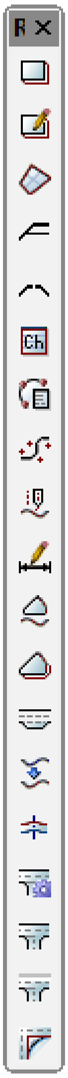
4.28.3 Road Toolbar

Position of option on menu: various

The **Road** toolbar is:



Options on Toolbar



pad create
pad edit
dynamic pad pond
subgrade batter intersection
section segment removal
special chainage file
kret special chainage file
set out points
label chainage/ offset
dimension create/edit
traffic island create
traffic island function
road widening
convert alignment
overlay design
create roads set up
create roads
create roads kret convert
computator kerb return

For *Pad Create,* go to	20.7.3 Allotment Pad Create
Pad Edit	20.7.4 Allotment Pad Edit
Dynamic Pad Pond	20.7.5 Dynamic Pad/Pond Interface
Subgrade Batter Intersection	20.8.12.11 Subgrade Intersect Function
Section Segment Removal	20.8.12.12 X Segment function
Special Chainage File	20.8.12.7 Special Chainage File
Kret Special Chainage File	20.8.12.6 Kerb Special Chainage File
Setout Points	17.18.5 Create Setout Points Using Super String
Label chainage / offset	24.8 Chainage/Offset Label Inquire
Dimension *Create/edit*	24.7 Create/Edit Dimensions

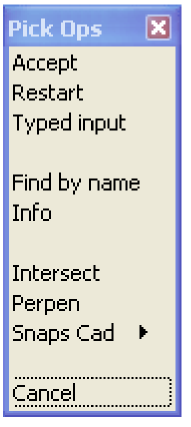
4.28.4 Snaps Cad

Position of option on menu: various

The Snaps Cad Menu can be accessed the following ways:

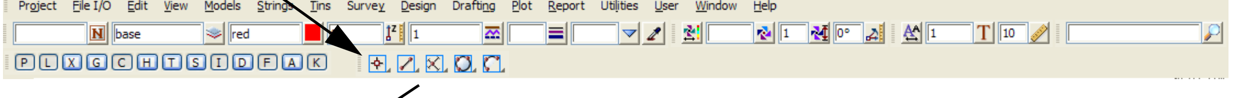
(a) by selecting **Snaps Cad** from the **Pick Ops** menu

(b) by walking right on **Snaps Cad** on the **Pick Ops** menu



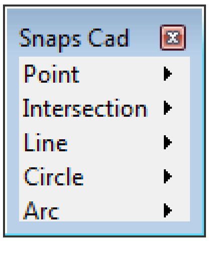
(c) From the toolbar, as long as **Snaps Cad** is ticked in the **Customize toolbars** panel (**View =>Toolbars).**

snaps cad toolbar



When you unpin the **Snaps Cad toolbar**, it will then appear on the screen as a floating toolbar.

The **Snaps Cad Menu** walk- right menu is:

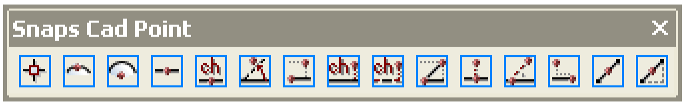


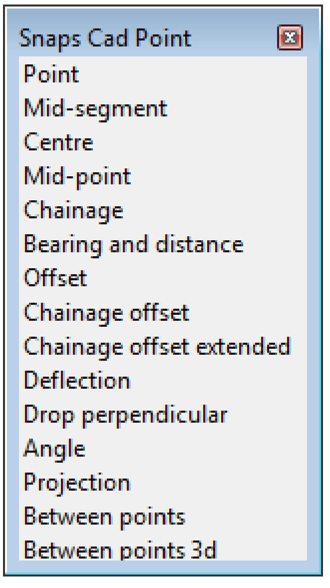
For the option *Point*, go to	4.28.5 Snaps Cad Point
Intersection	4.28.5.1 Snaps Cad Intersection
Line	4.28.5.2 Snaps Cad Line
Circle	4.28.5.3 Snaps Cad Circle
Arc	4.28.5.4 Snaps Cad Arc

Note: Snap Cads work in a very similar pattern to the normal Cad options.However, they create virtual points and segments which can be used by either **Cad** or **Snap Cad** options. **Snap Cad** options don’t create any points or strings.

4.28.5 Snaps Cad Point

This section of documentation is a work in progress and will be updated in subsequent releases.





at a user selected position
at middle of a selected segment
at the centre point of an arc
at the middle of the two selected points
at the position of a chainage along a selected string
a given bearing and distance from a selected position
at the offset position of a point on a selected string
at the offset position of a chainage along a selected string
at the offset position of a chainage along a selected string which can be extended
a given distance at right angles to a selected string
dropped perpendicularly onto a selected string
a given distance at a given angle to a selected string
projected a chainage distance along a selected string

For the option *Point*, go to		Point
	Mid- segment	Mid- segment
	Centre	Centre
	Mid-point	Mid point
	Chainage	Chainage
	Bearing and distance	Bearing and distance
	Offset	Offset
	Chainage offset	Chainage offset
	Chainage offset extended	Chainage offset extended
	Deflection	Deflection
	Drop perpendicular	Drop perpendicular
	Angle	Angle
	Projection	Projection
	Between points	Between Points
	Between points 3d	Between points 3d

Point

On selecting **Point**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

The user can select a position with the mouse and on accepting that point (Middle mouse button or enter) the virtual point is created at the selected position and ready to be used by the calling option.

The snap mode will influence the mouse selection. For example if cursor snap is on, the user can choose a position not yet defined. If point snap is on and the selection snaps to an existing point, the option will place another point at that location.

The user can also activate the selection menu used with the mouse (right button) that allows various positioning options.

Mid- segment

This option creates a virtual point in the middle of a selected line or arc segment.

On selecting **Mid- segment**, the user is prompted to select a segment and a virtual point is created at the mid point of the segment.

Centre

This option creates a virtual point at the centre of a selected segment.

On selecting **Centre** the user is prompted to select a segment and a virtual point is created at the centre of the segment.

Mid point

This option creates a virtual point in the middle of two selected positions.

On selecting **Mid point,** the user is prompted to select the first position and then the second position. A virtual point is created at the mid point of the two selected positions.

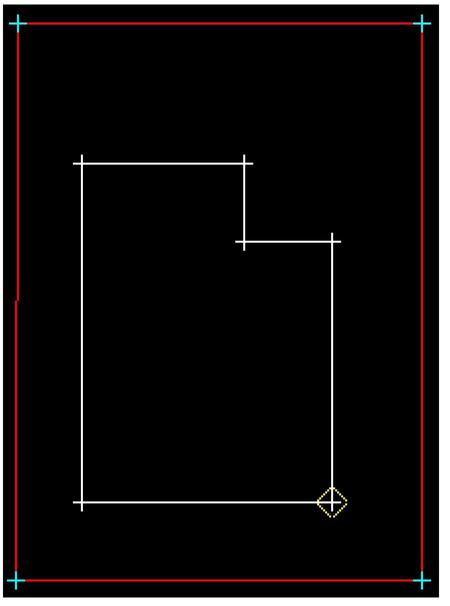
Chainage

This option creates a virtual point at the position of a user specified chainage of an existing string.

On selecting **Chainage**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

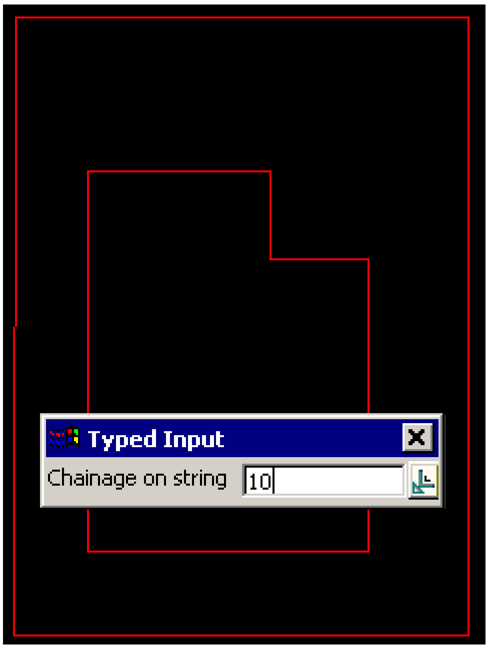
STEP 1:

The user selects a valid string (left mouse button) and accepts that string (Middle mouse button or enter).



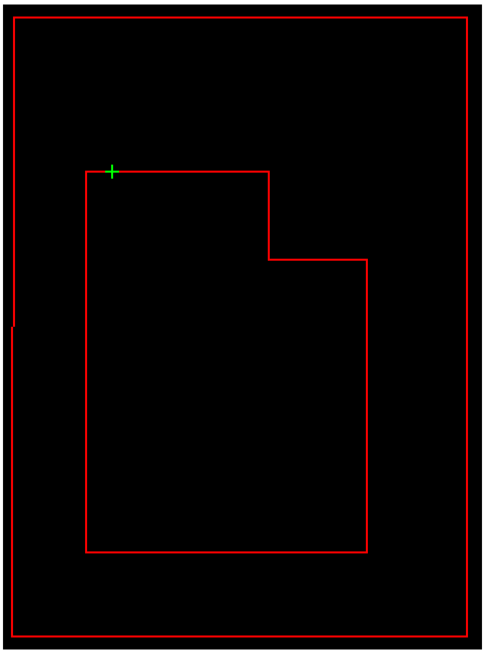
STEP 2:

The user enters a chainage value where a point should be placed followed by the enter key. This chainage is with respect to the selected string.



STEP 3:

A virtual point is created at the positioned on the string at the given chainage.



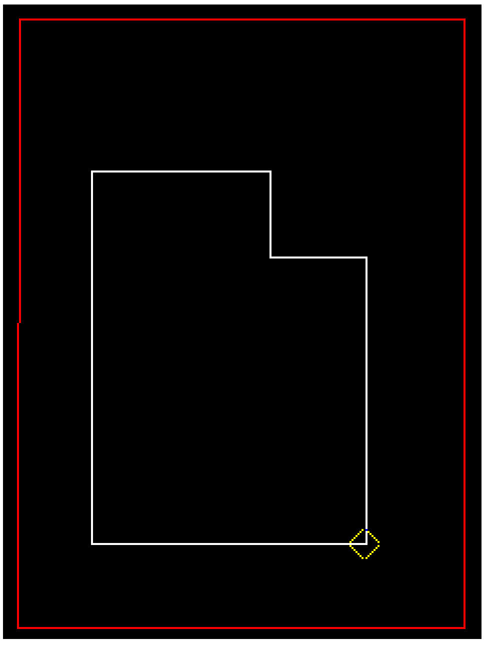
Bearing and distance

This option calculates a virtual point that is located at a given bearing and distance from a start position.

On selecting **Bearing and distance**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

A start position is selected and accepted.

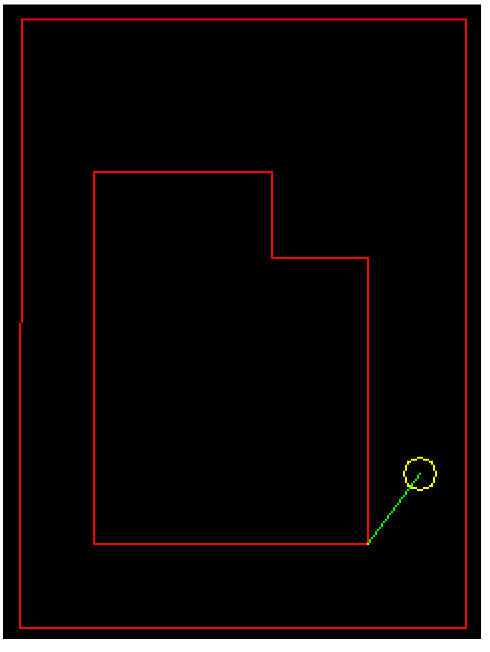


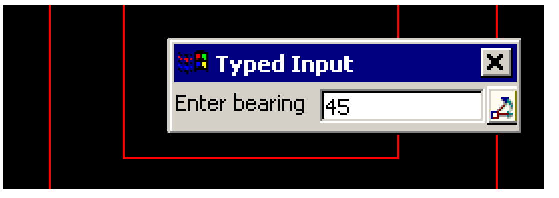
STEP 2:

A bearing is selected with the mouse or entered in via the keyboard. For typed entry, simply start typing or press the space bar to bring up the bearing entry box. The value is entered into the input box followed by the Enter key.

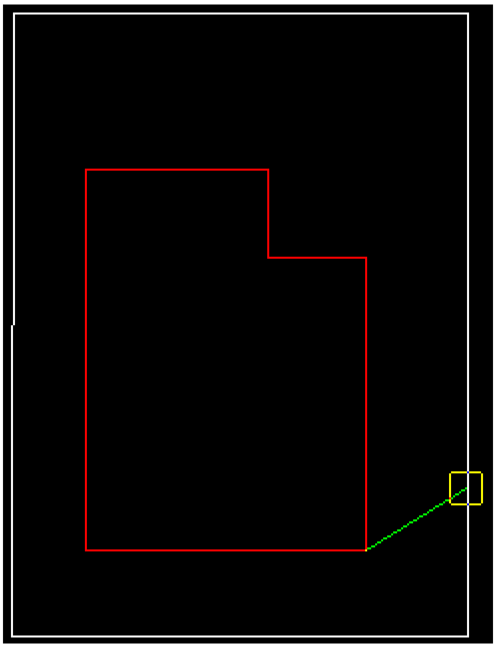
The line drawn represents the bearing value and changes with movement of the mouse. If the user wants to see what the current value of the bearing is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the point or the input box can be closed and the rubber banding (graphically changing) of the bearing continued.

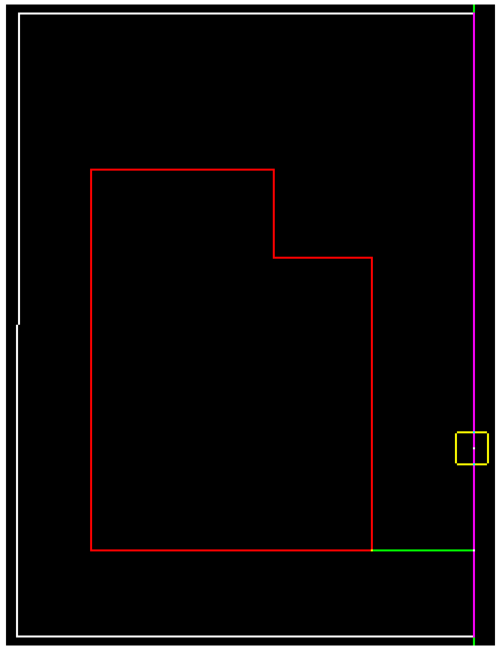
Note: **The Page up and page down keys can be used when the input angle box comes up to add or subtract intervals of 90 degrees.**





This option also allows the definition of the bearing by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next. The example shown below is the perpendicular case.

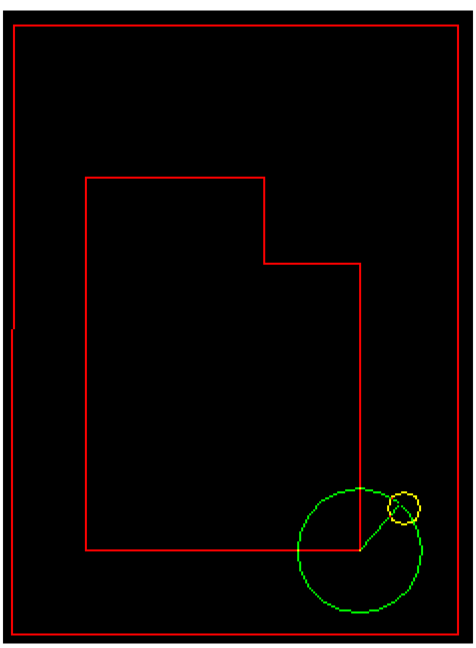


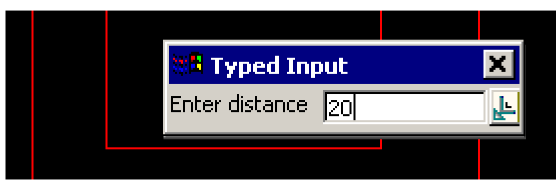


STEP 3:

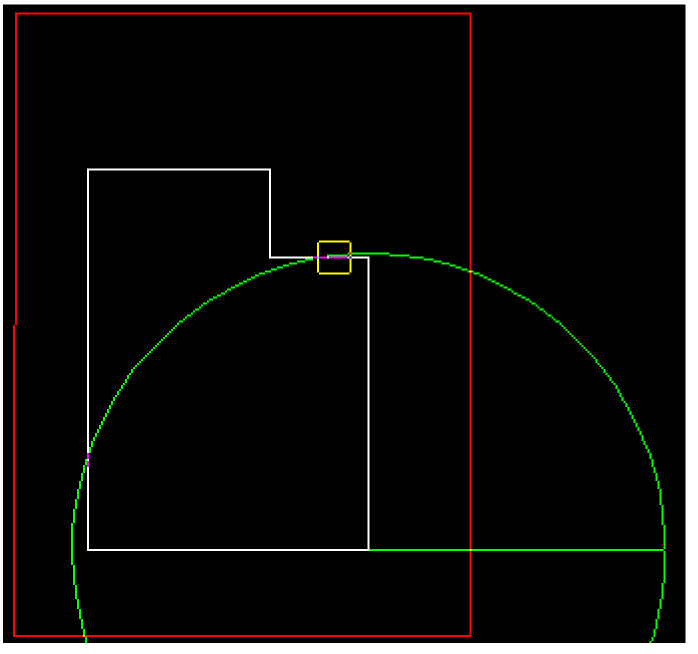
A distance is selected and accepted with the mouse or entered in via the keyboard. For typed entry, simply start typing or press the space bar to bring up the distance entry box. The value is entered into the input box followed by the enter key.

The circle drawn represents the distance value and changes with movement of the mouse. If the user wants to see what the current value of the distance is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the point or the input box can be closed and the rubber banding (graphically changing) of the circle continued.

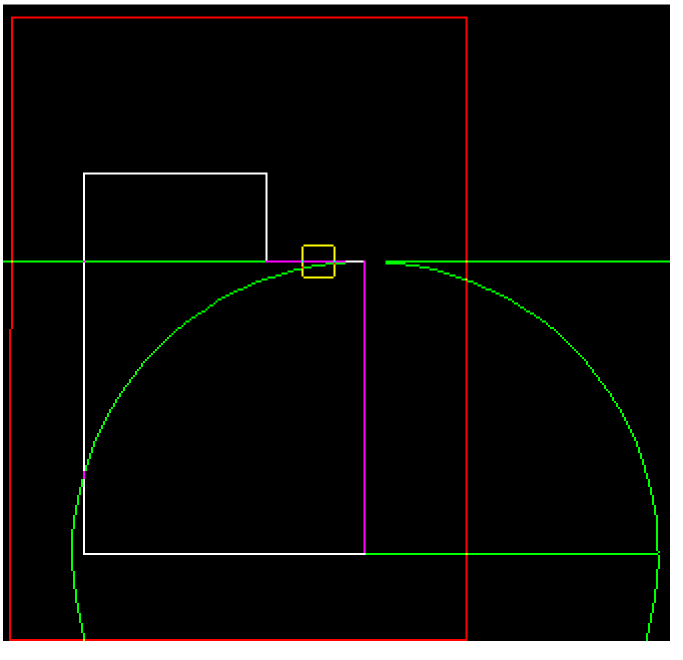




This option also allows the definition of the distance by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next. The example shown below is the perpendicular case.

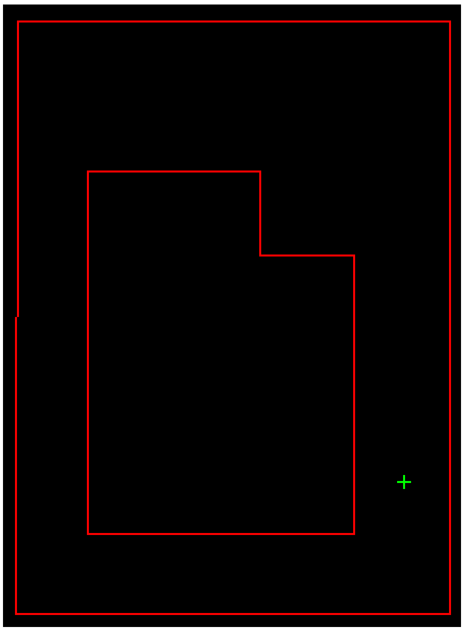


The perpendicular distance shown below by the purple line will be use at the defined bearing to create the point.

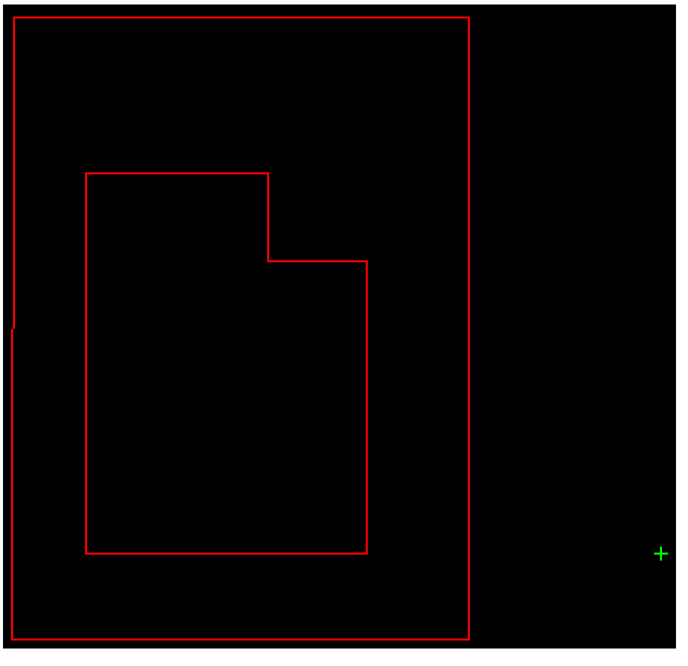


STEP 4:

A virtual point is created using created using the information supplied. By entry into the input boxes:



Or by using the perpendicular/tangential tools:



Offset

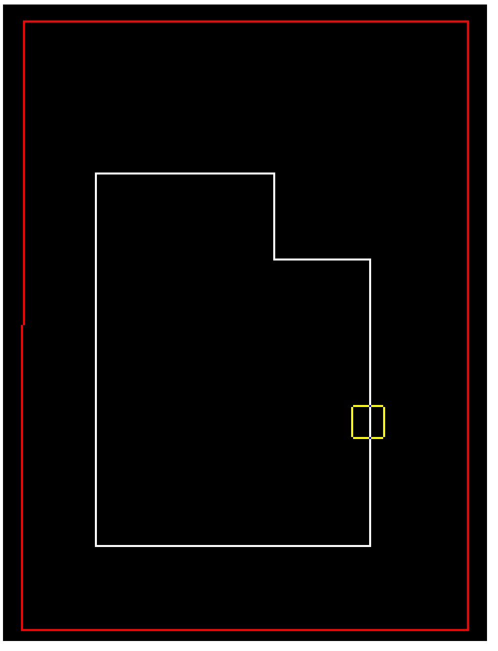
This option creates a virtual point that is located by reference to a string, a control point, a chainage distance along the string from the control point and an offset to the selected string.

After selection and acceptance of a string, a control point is selected and accepted. This point is dropped perpendicular onto the string. The distance along the string is measured from this dropped point. Positive distances are in the direction that the string was picked. Finally a offset to the string can be specified for the placement of the new virtual point.

On selecting **Offset**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

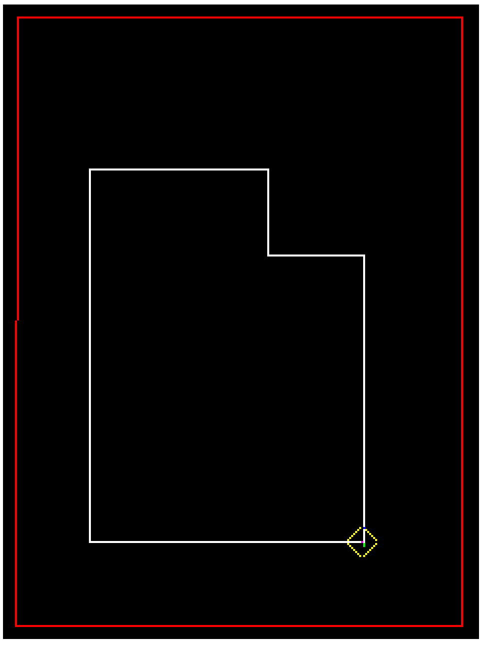
STEP 1:

The user selects and accepts a string with direction. This defines what side the offset applies to.



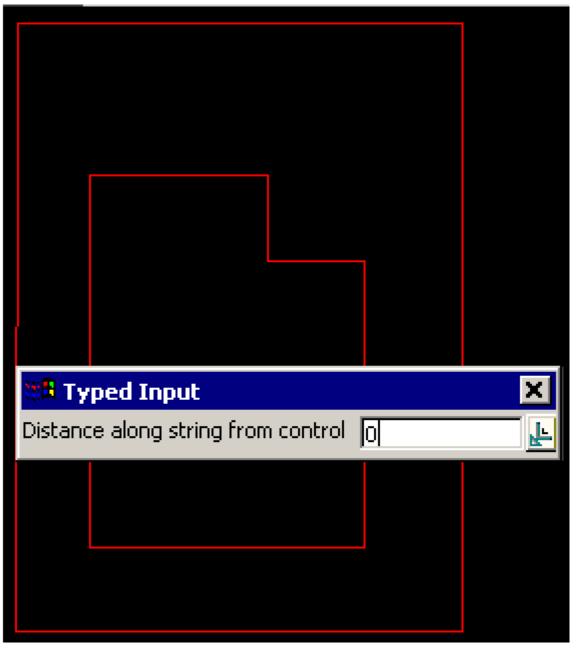
STEP 2:

The user picks and accepts a control point to be dropped onto the selected string.



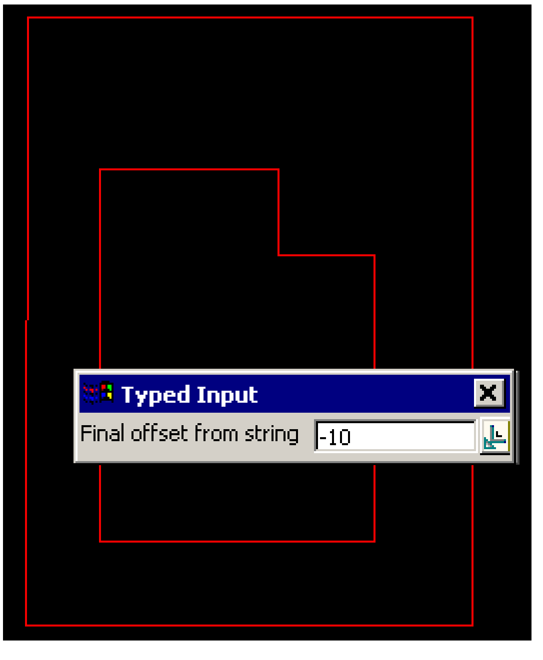
STEP 3:

A distance along from the dropped point is specified. Positive distances are in the direction of the string selection pick. The value is entered into the input box followed by the Enter key.



STEP 4:

An offset relative to the selected string (and direction) is specified in the input box followed by the Enter key.



STEP 5:

A virtual point is created using the information supplied.

Chainage offset

This option creates a virtual point that is located perpendicular to the reference string with a defined offset.

STEP 1:

The user selects the reference string with direction.

STEP 2:

The user specifies a chainage on the reference string.

STEP 3:

The user specifies an offset distance from the reference string.

A virtual point is created using the information supplied.

Chainage offset extended

Is similar to chainage offset except the specified chainage can be extended beyond the start and end chainages of the selected string.

Deflection

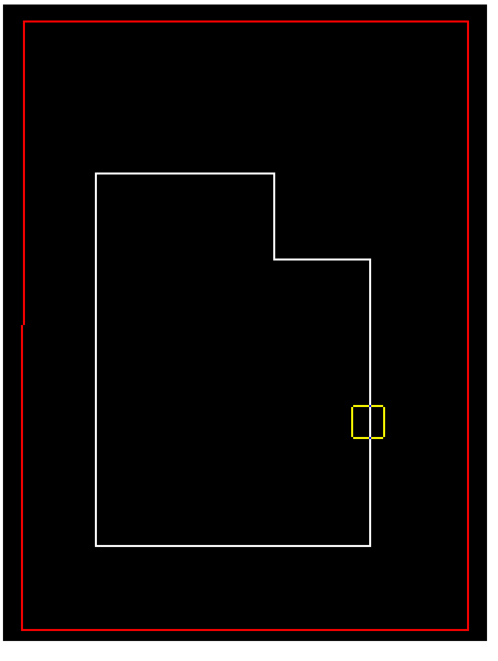
This option creates a virtual point that is located by reference to a string, a control point, a distance along the string from the control point, a deflection angle and deflection distance.

After selection of a string, a control point is selected. This point is dropped perpendicular onto the string. A distance along the string can be entered to move the measure point. Positive distances are in the direction that the string was picked. A deflection angle is specified which is a clockwise angle from the measure point. The deflection distance is the distance from the measure point to the point which is to be created.

On selecting **Deflection**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

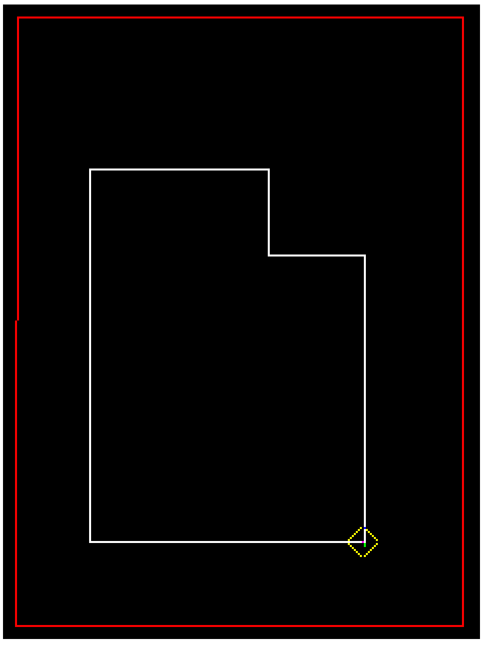
STEP 1:

The user selects and accepts a string with direction.



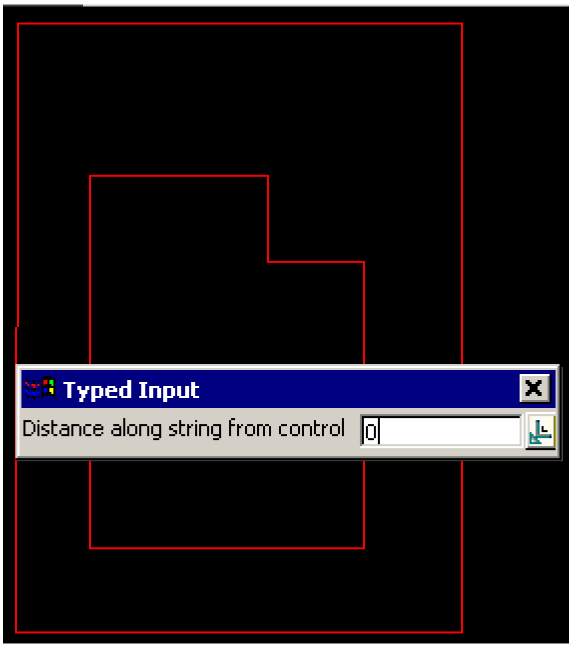
STEP 2:

The user picks and accepts a control point to be dropped onto the selected string.



STEP 3:

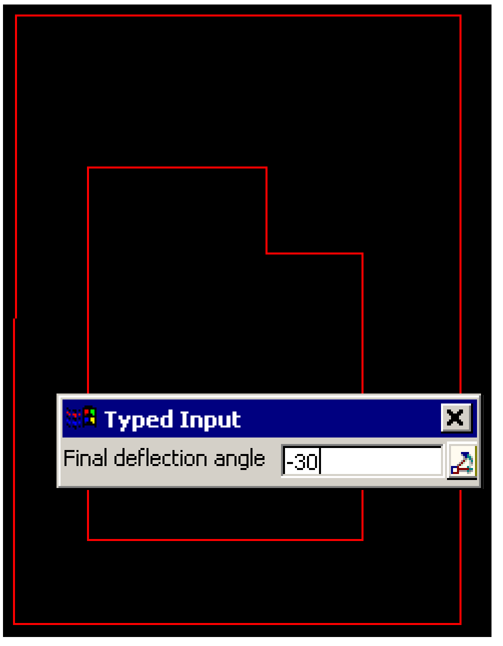
A distance along from the dropped point is specified. Positive distances are in the direction of the string selection pick. The value is entered into the input box followed by the Enter key.



STEP 4:

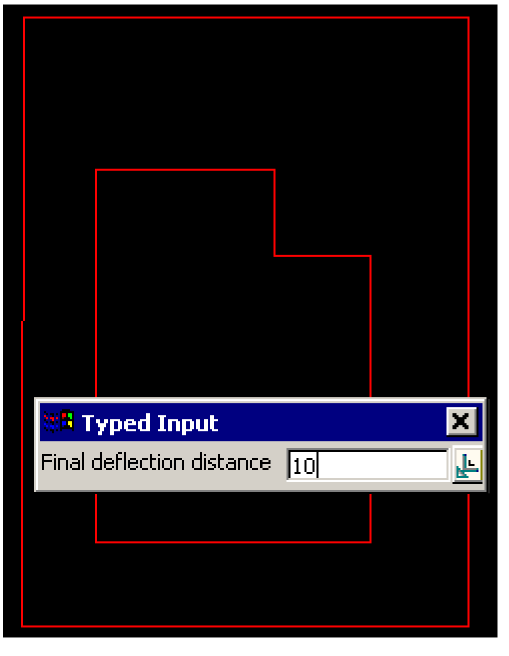
A deflection angle is specified. This angle is clockwise, relative to the direction of the string selection pick. The value is entered into the input box followed by the enter key.

Note: **The Page up and page down keys can be used when the input angle box comes up to add or subtract intervals of 90 degrees.**



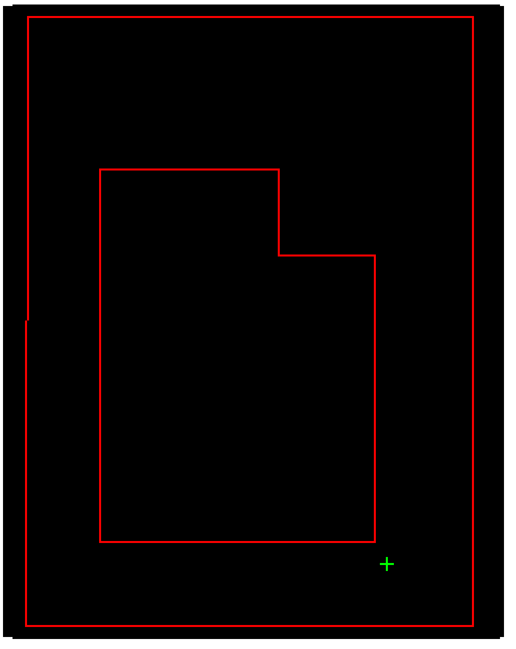
STEP 5:

A deflection distance is supplied. Positive is in the direction of the string selection pick. The value is entered into the input box followed by the Enter key.



STEP 6:

A virtual point is created using the information supplied.



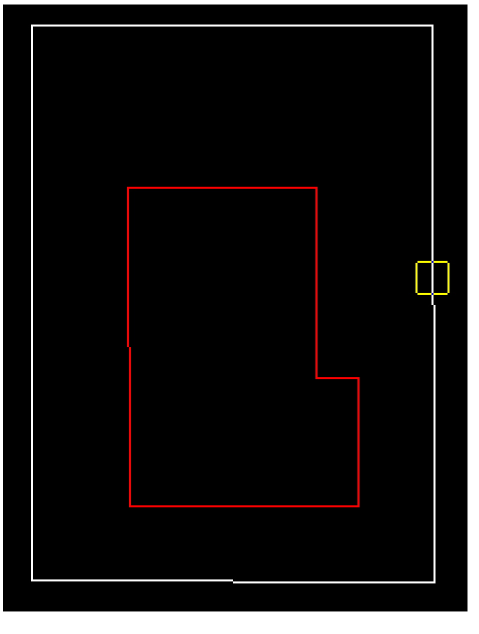
Drop perpendicular

This option creates a virtual point by dropping from a user selected position perpendicularly onto a user selected string.

On selecting **Drop perpendicular**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

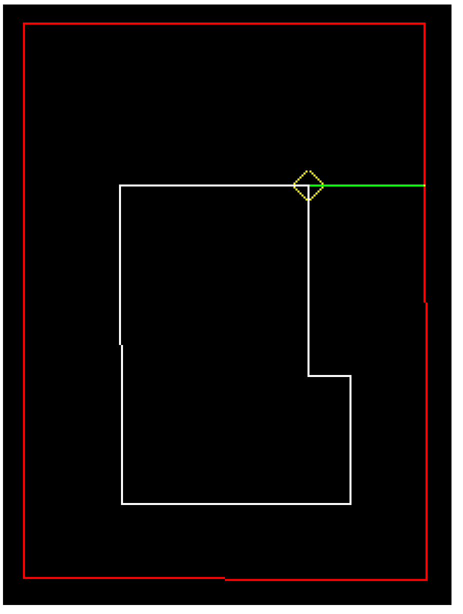
STEP 1:

The user selects a string and accepts (with direction) to have the point dropped onto.



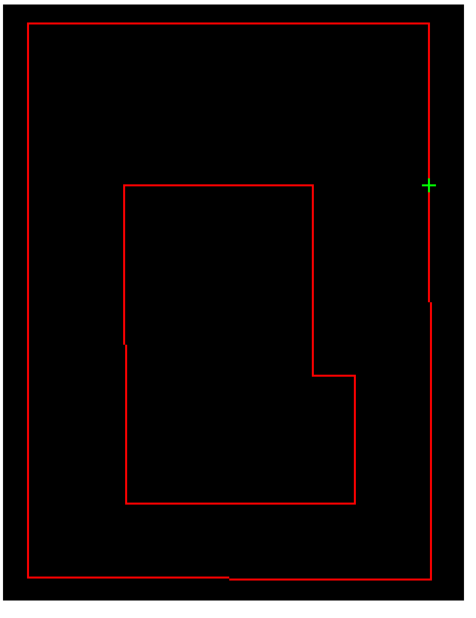
STEP 2:

The user picks and accepts a position to drop onto the nominated string



STEP 3:

A virtual point is created at the perpendicular drop point.



Angle

This option creates a virtual point that is located by projecting a selected position back to a string by a specified angle.

STEP 1:

User selects and accepts a string to have the point projected onto.

STEP 2:

User specifies an angle for the projection.

STEP 3:

User selects a position on the screen to be projected back to the string.

STEP 4:

A virtual point is created, which form the nominated angle with the selected position on **STEP 3.**

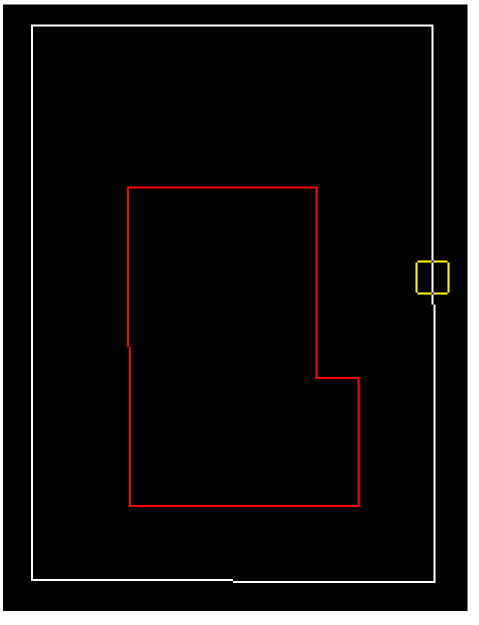
Projection

This option creates a virtual point that is located by firstly dropping a point onto a string and then giving a distance along the string from the dropped point.

On selecting **Projection**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

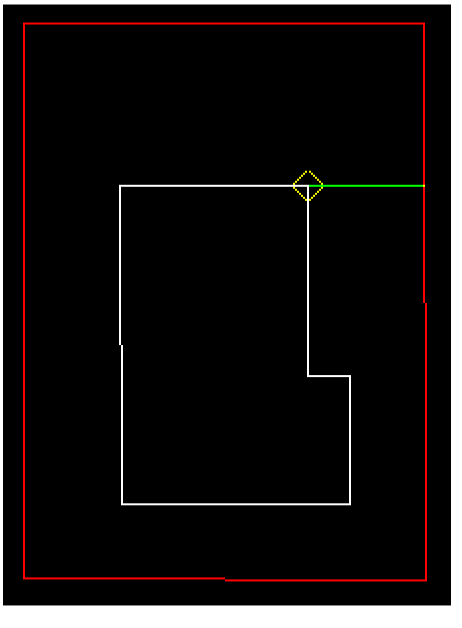
STEP 1:

The user selects and accepts a string (with direction) to have the point dropped onto



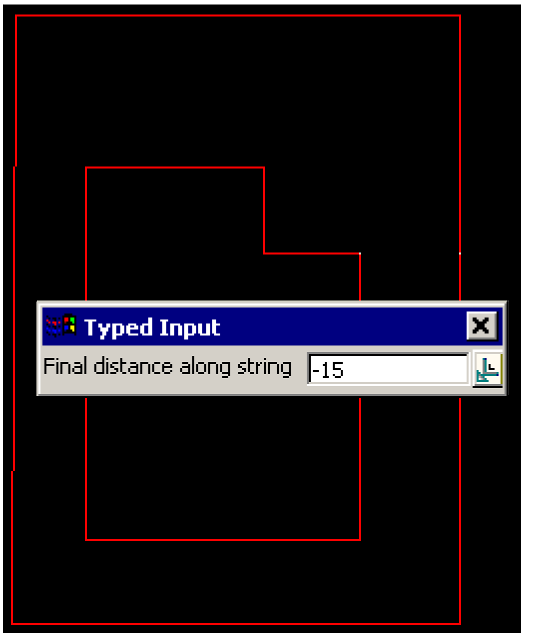
STEP 2:

The user picks and accepts a position to drop onto the nominated string



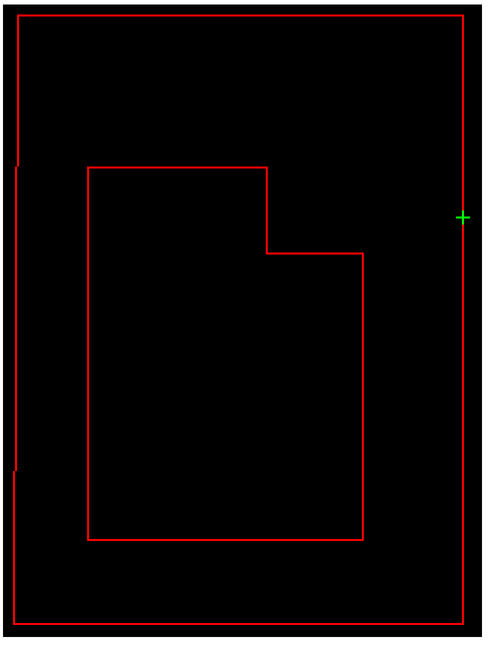
STEP 3:

The user is prompted for the distance along the string. Positive distances are in the direction that the string was picked. A value is entered into the input box followed by the Enter key.



STEP 4:

The virtual point is created the nominated distance along the string from the dropped point.



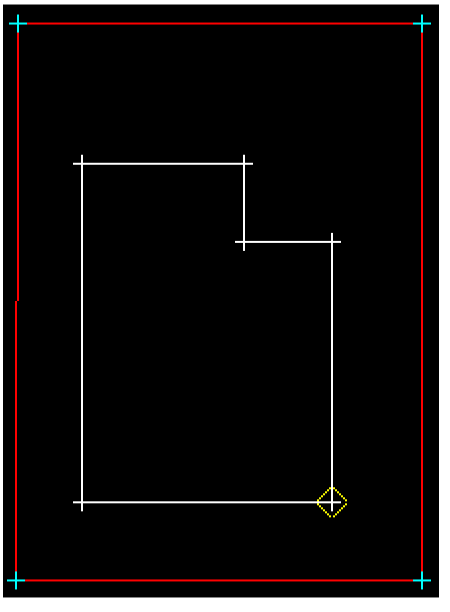
Between Points

This option creates a virtual point that is on the line between two selected positions and a given distance from the first point.

On selecting **Between points**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

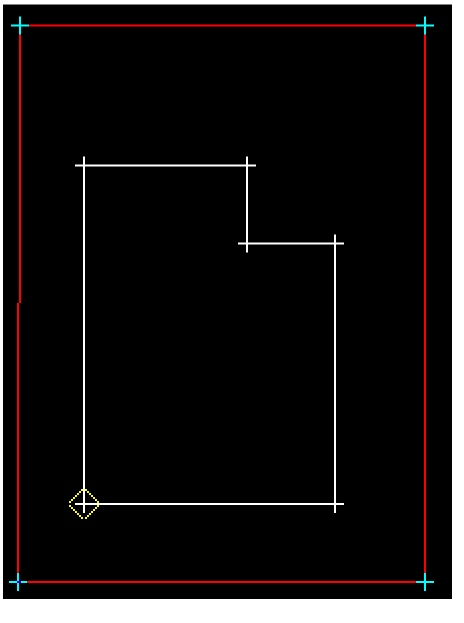
STEP 1:

The user selects a start point. Specification of a position can be done by the direct input of the xyz coordinate of the point by pressing the space bar to bring up the enter XYZ panel or by typing of the value to bring up the XYZ panel. The user can also select a point with the mouse and accepts that point (Middle mouse button or enter).



STEP 2:

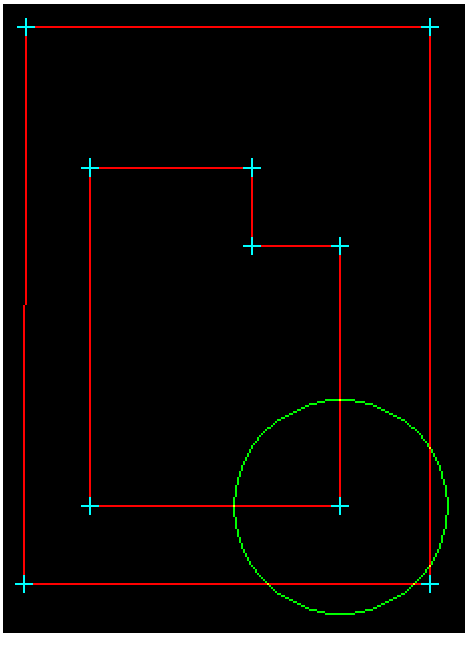
The user picks a 2nd point and accepts that point (Middle mouse button or enter) to define the reference line.

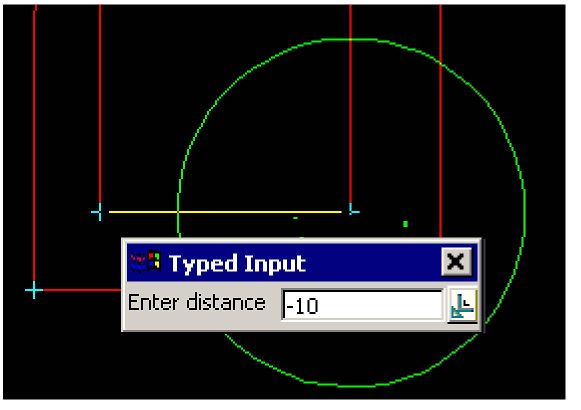


STEP 3:

A distance from the 1st point to create the point is given either by selection with the mouse or by typing a value. To type a value, simply start typing and the input box for the distance will appear. Alternatively you can press the space bar to bring up the input box. Enter the value and then the enter key.

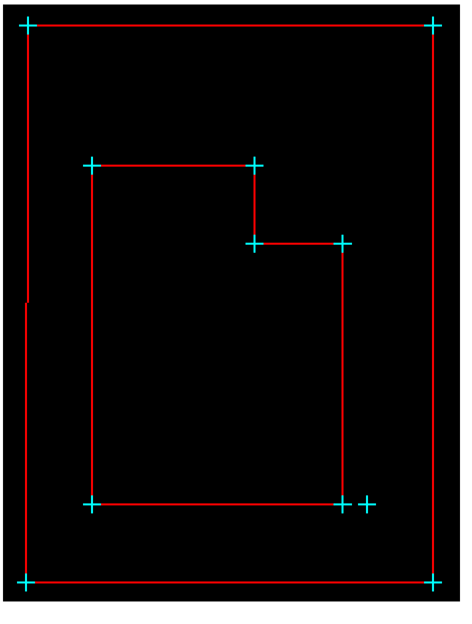
The circle drawn represents the distance value and changes with movement of the mouse. If the user wants to see what the current value of the distance is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the point or the input box can be closed and the rubber banding (graphically changing) of the circle continued.





STEP 4:

A virtual point is created on the segment specified by the selection of the 1st and 2nd points, at the nominated distance. Negative distances can be entered as in this example.



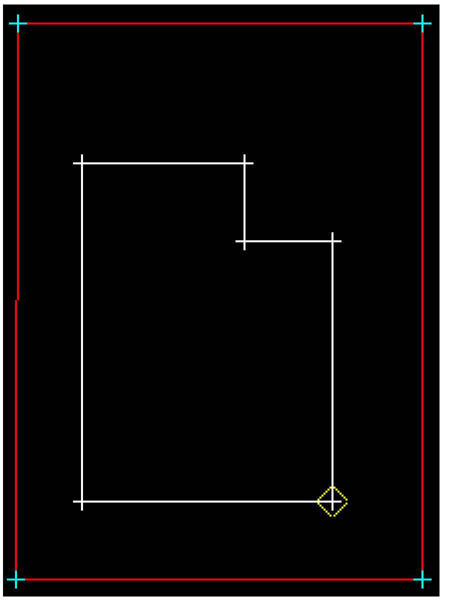
Between points 3d

This option creates a virtual point that is on the line between two selected positions and a given distance from the first point. The z-value of the string is interpolated from the two selected positions.

On selecting **Between points 3d**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

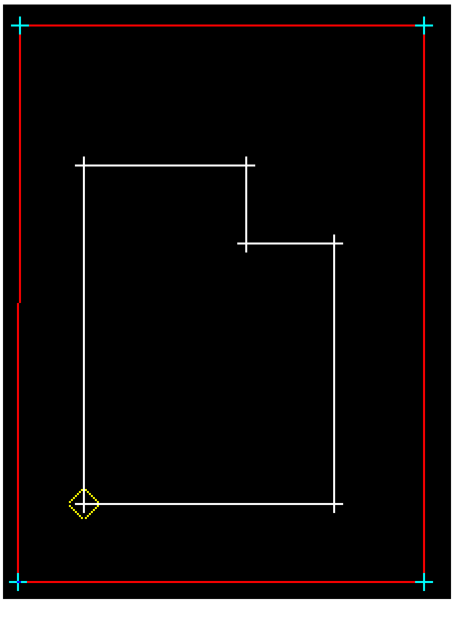
STEP 1:

The user selects a start point. Specification of a position can be done by the direct input of the xyz coordinate of the point by pressing the space bar to bring up the enter XYZ panel or by typing of the value to bring up the XYZ panel. The user can also select a point with the mouse and accepts that point (Middle mouse button or Enter).



STEP 2:

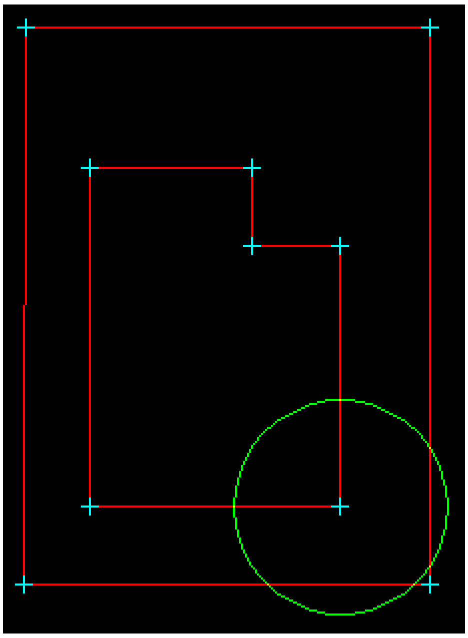
The user picks a 2nd point and accepts that point (Middle mouse button or <Enter>) to define the reference line.

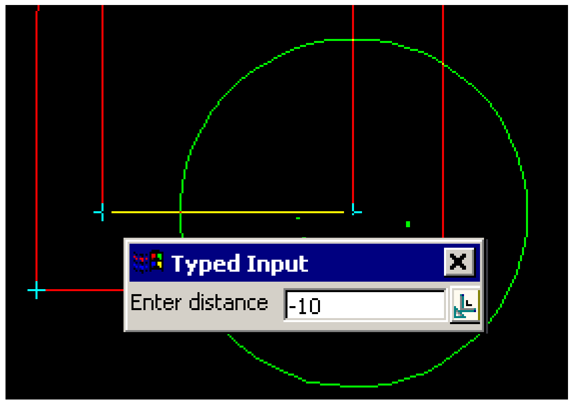


STEP 3:

A distance from the 1st point to create the point is given either by selection with the mouse or by typing a value. To type a value, simply start typing and the input box for the distance will appear. Alternatively you can press the space bar to bring up the input box. Enter the value and then the enter key.

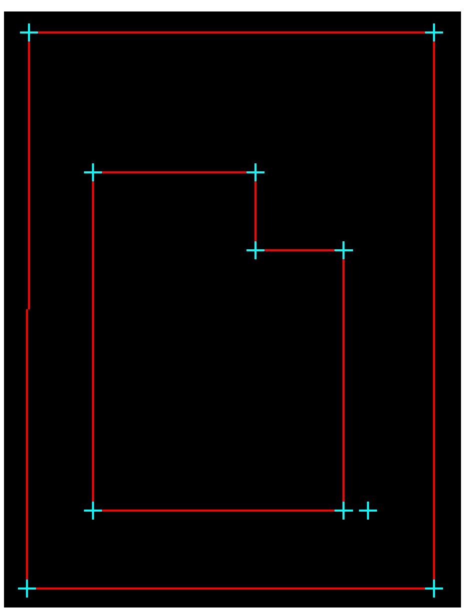
The circle drawn represents the distance value and changes with movement of the mouse. If the user wants to see what the current value of the distance is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the point or the input box can be closed and the rubber banding (graphically changing) of the circle continued.





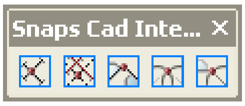
STEP 4:

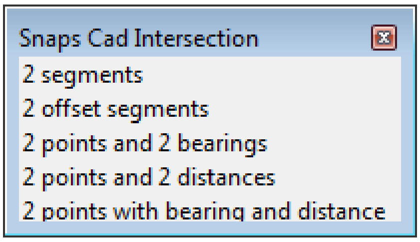
A virtual point is created on the segment specified by the selection of the 1st and 2nd points, at the nominated distance. Negative distances can be entered as in this example.



4.28.5.1 Snaps Cad Intersection

This section of documentation is a work in progress and will be updated in subsequent releases.





for the option 2 segments, go to the section [2 segments](#)
2 offset segments [2 offset segments](#)
2 points and 2 bearings [2 points and 2 bearings](#)
2 points and 2 distances [2 points and 2 distances](#)
2 points with bearing and distance [2 points with bearing and distance](#)

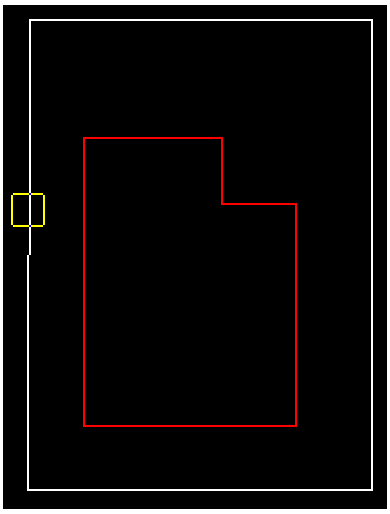
2 segments

This option creates a virtual point at the intersection of the projections of two line or arc segments.

On selecting **2 segments**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

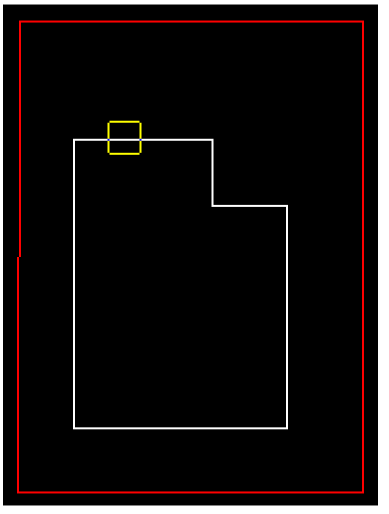
STEP 1:

The 1st segment is selected and accepted.



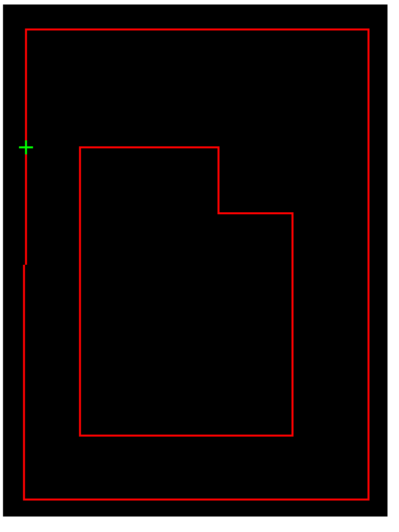
STEP 2:

The 2nd segment is selected and accepted.



STEP 3:

A virtual point is created at the intersection of the two segments (if a solution exists). Note that the 2nd segment in this case has been projected to enable a solution to be calculated.



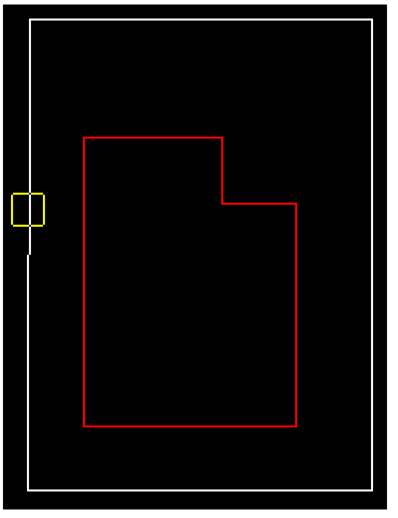
2 offset segments

This option creates a virtual point at the intersection of the offsets of two selected segments.

On selecting **2 offset segments**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

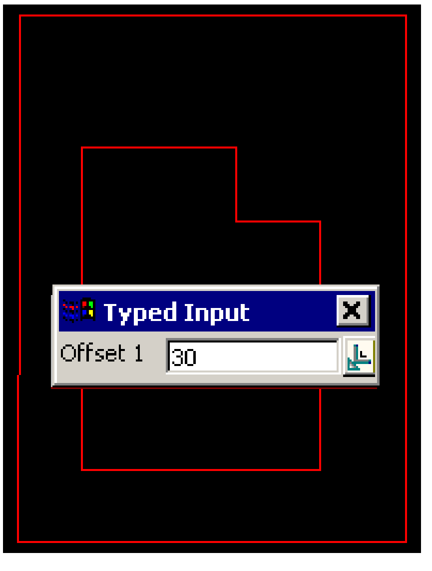
STEP 1:

The 1st segment is selected and accepted with direction. This sets the positive direction of the offset to the right of the direction of pick.



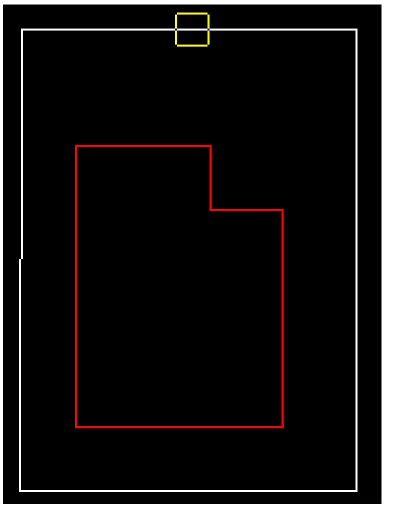
STEP 2:

The offset is given by into an offset input box. The positive direction is at 90 degrees to the direction of pick for the segment. The value is entered into the input box followed by the Enter key.



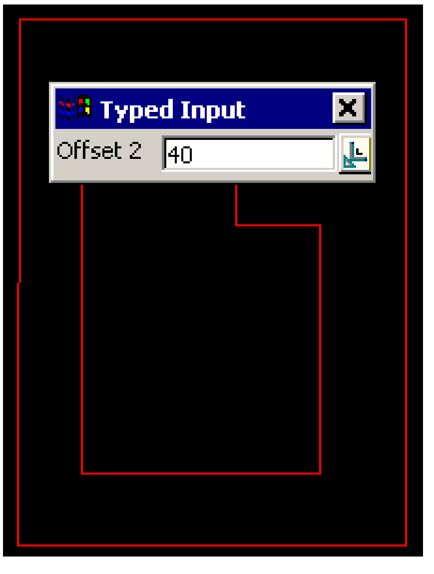
STEP 3:

The 2nd segment is selected with direction and accepted. This sets the positive direction of the offset to the right of the direction of pick.



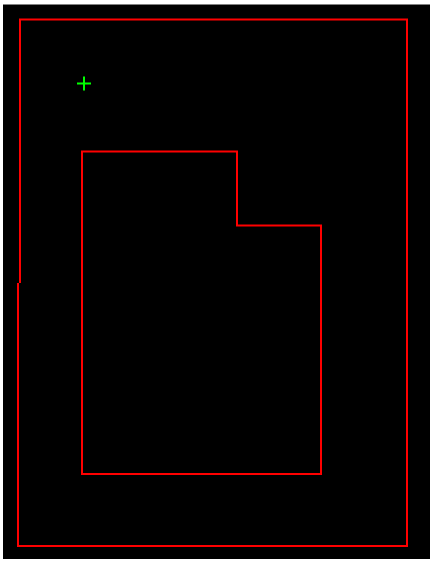
STEP 4:

The offset is given by into an offset input box. The positive direction is at 90 degrees to the direction of pick for the segment. The value is entered into the input box followed by the Enter key.



STEP 5:

A virtual point is created at the intersection of the projected lines offset to the segments (if a solution exists).



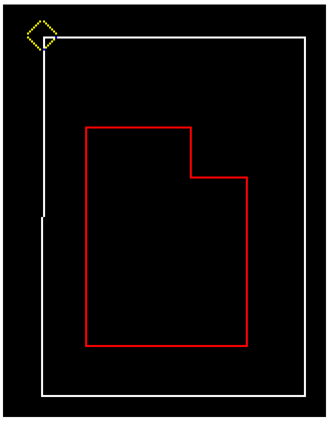
2 points and 2 bearings

This option creates a virtual point by using two points and two bearings.

On selecting **2 points and 2 bearings**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

The 1st point is selected and accepted.

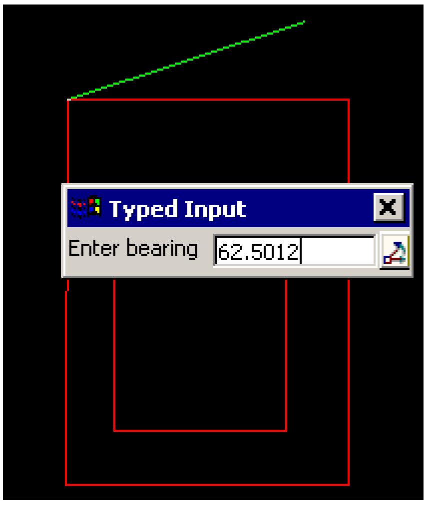


STEP 2:

A bearing from the 1st point to create the point is given either by selection with the mouse or by typing a value. To type a value, simply start typing and the input box for the bearing will appear. Alternatively you can press the space bar to bring up the input box. The value is entered into the input box followed by the Enter key.

The line drawn represents the bearing value and changes with movement of the mouse. If the user wants to see what the current value of the bearing is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the point or the input box can be closed and the rubber banding (graphically changing) of the line continued.

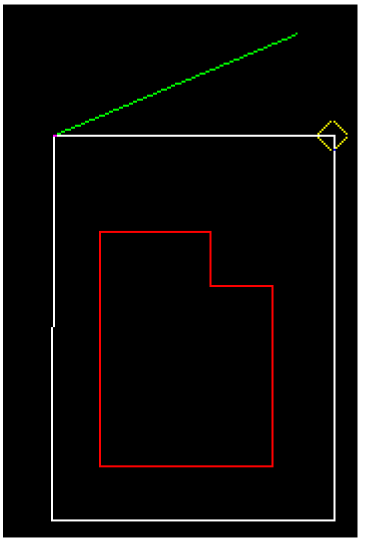
Note: **The Page up and page down keys can be used when the input angle box comes up to add or subtract intervals of 90 degrees.**



This option also allows the definition of the bearing by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next.

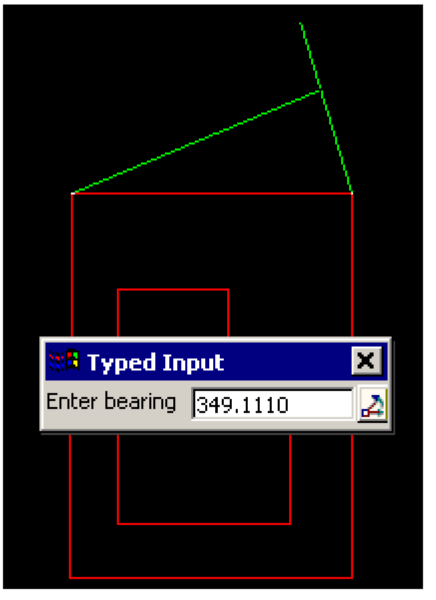
STEP 3:

The 2nd point to create the point from is selected and accepted.



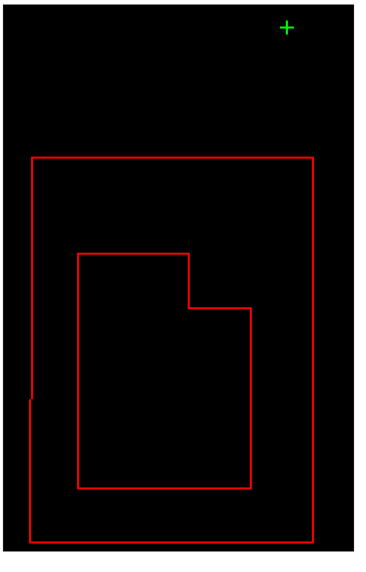
STEP 4:

A bearing from the 2nd point to create the point is given using the optional outlined in **STEP 2** above.



STEP 5:

The virtual point created if there is a valid solution.



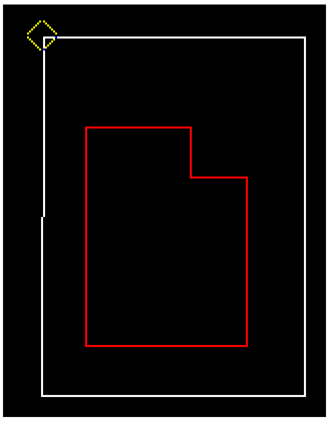
2 points and 2 distances

This option creates a virtual point using two points and two distances.

On selecting **2 points and 2 distances**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

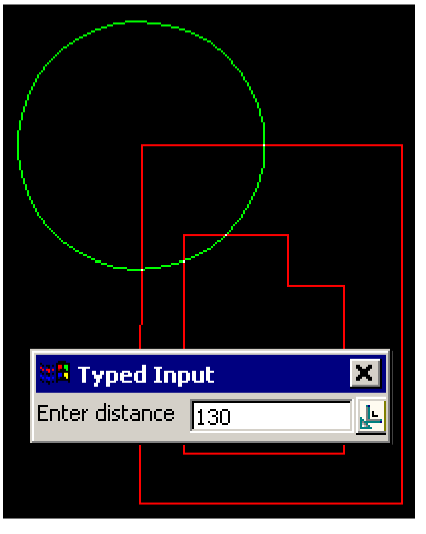
The 1st point is selected and accepted.



STEP 2:

A distance from the 1st point to create the point is given either by selection with the mouse or by typing a value. To type a value, simply start typing and the input box for the distance will appear. Alternatively you can press the space bar to bring up the input box. The value is entered into the input box followed by the enter key.

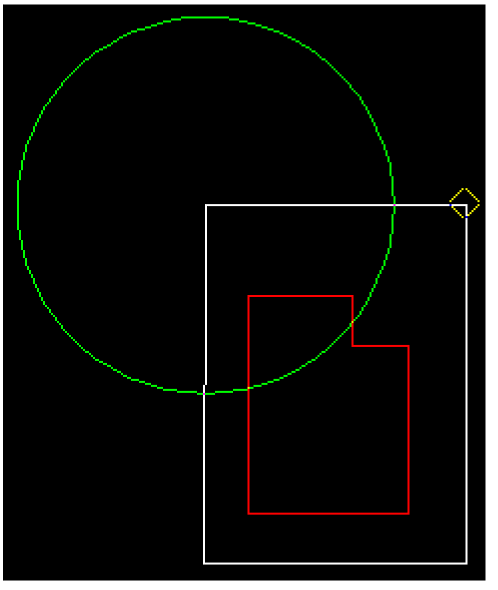
The circle drawn represents the distance value and changes with movement of the mouse. If the user wants to see what the current value of the distance is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the point or the input box can be closed and the rubber banding (graphically changing) of the circle continued.



This option also allows the definition of the distance by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next. The example shown below is the perpendicular case.

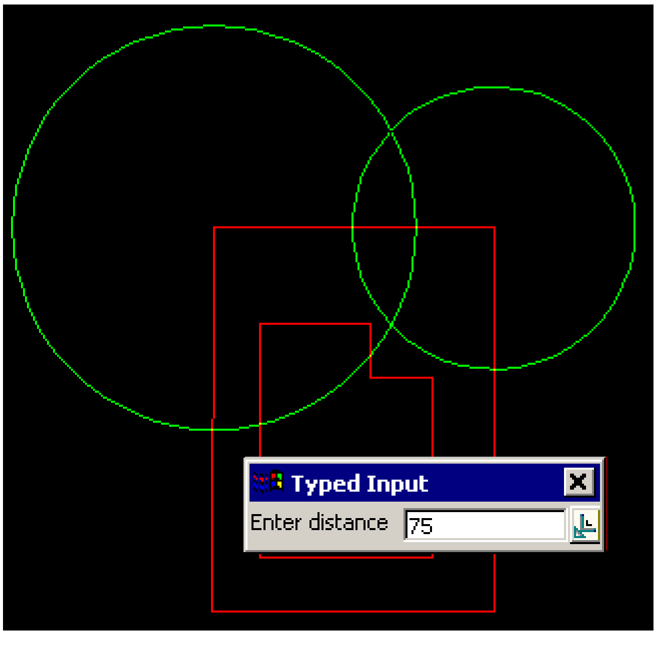
STEP 3:

The 2nd point to create the point from is selected.



STEP 4:

A distance from the 2nd point to create the point is given using the optional outlined in **STEP 2** above.

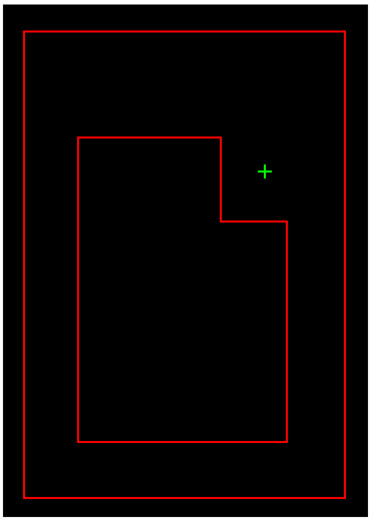


STEP 5:

As there is two solutions, the user can select the correct one depending on the method of construction. This can be done by the direct entry of distances or by use of the mouse.

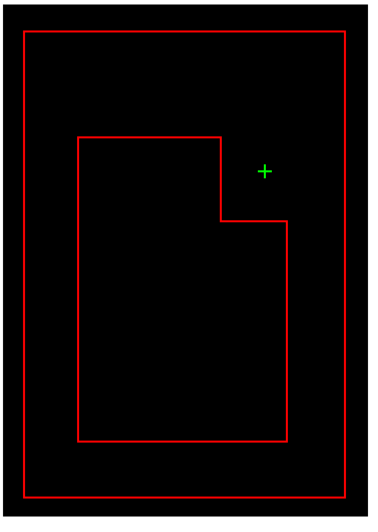
1. Distance entry. After the entry of the 1st distance, the 2nd point is selected. Following the selection of the 2nd point, the 2nd radius is shown (rubber banding). The user can select one of the two solutions by choosing with a **LB** mouse click over the approximate position of the required solution. The solution chosen is the closest solution to the selected point. The final radius can then be entered via the keyboard by simply starting typing which brings up the radius entry panel automatically. This panel can also be activated by pressing the space bar.
2. Use of the mouse. The 1st point is selected and the radius entered by using the mouse or by direct entry from the keyboard. The 2nd point is then selected and the 2nd radius is displayed

(rubber banding). A solution can be chosen by selecting with a **LB** mouse click over the required solution. The final solution will be the closest one to the selection. **MB** to accept the intersection and create the point.



STEP 6:

The virtual point is created if there is a valid solution.



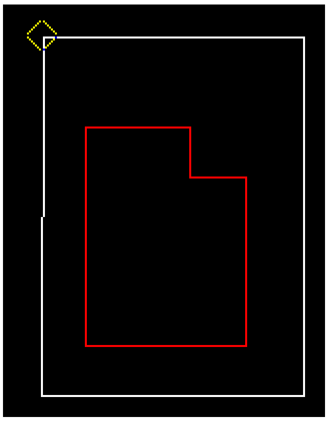
2 points with bearing and distance

This option creates a virtual point from a given point and a bearing, and a second point and a distance.

On selecting **2 points with bearing and distance**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

The 1st point is selected and accepted.

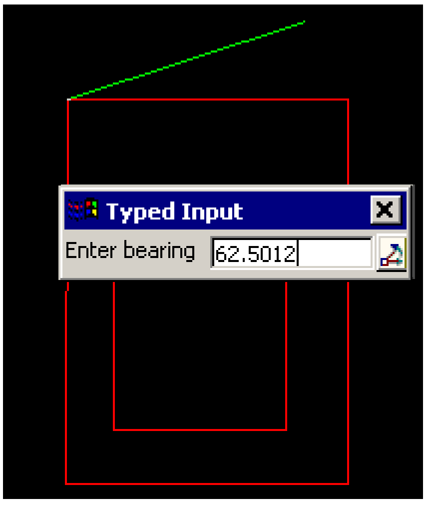


STEP 2:

A bearing from the 1st point to create the point is given either by selection with the mouse or by typing a value. To type a value, simply start typing and the input box for the bearing will appear. Alternatively you can press the space bar to bring up the input box. The value is entered into the input box followed by the Enter key.

The line drawn represents the bearing value and changes with movement of the mouse. If the user wants to see what the current value of the bearing is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the point or the input box can be closed and the rubber banding (graphically changing) of the line continued.

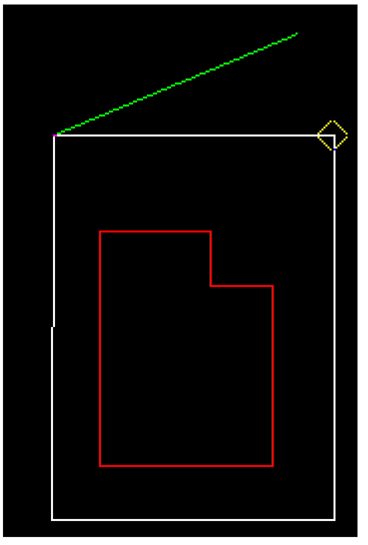
Note: **The Page up and page down keys can be used when the input angle box comes up to add or subtract intervals of 90 degrees.**



This option also allows the definition of the bearing by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next.

STEP 3:

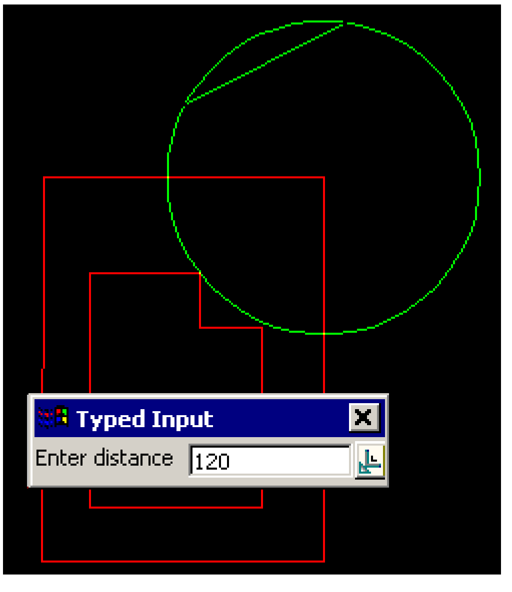
The 2nd point to create the point from is selected and accepted.



STEP 4:

A distance from the 2nd point to create the point is given either by selection with the mouse or by typing a value. To type a value, simply start typing and the input box for the distance will appear. Alternatively you can press the space bar to bring up the input box. The value is entered into the input box followed by the Enter key.

The circle drawn represents the distance value and changes with movement of the mouse. If the user wants to see what the current value of the distance is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the point or the input box can be closed and the rubber banding (graphically changing) of the circle continued.

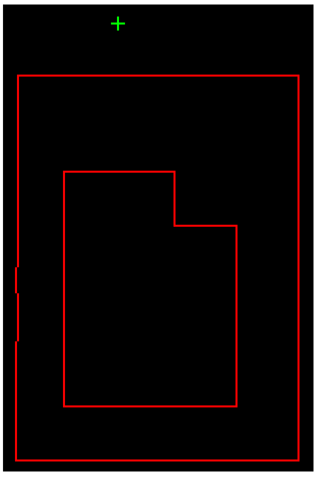


This option also allows the definition of the distance by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next. The example shown below is the perpendicular case.

A line is drawn between the two possible solutions.

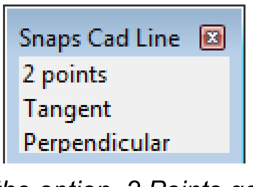
STEP 5:

As there is two solutions, the user can select the correct one with the mouse.



4.28.5.2 Snaps Cad Line

This section of documentation is a work in progress and will be updated in subsequent releases.



For the option, 2 Points go to 2 Points
Tangent Tangent
Perpendicular Perpendicular

2 Points

This option creates a virtual line segment.

On selecting **2 points**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

The user can select a position with the mouse and on accepting that point (Middle mouse button or enter) the point is created at the selected position.

The snap mode will influence the mouse selection. For example if cursor snap is on, the user can choose a position not yet defined. If point snap is on and the selection snaps to an existing point, the option will place another point at that location.

The user can also activate the selection menu used with the mouse (right button) that allows various positioning options.

Specification of a position can also be done by the direct input of the xyz coordinate of the point by pressing the space bar to bring up the enter XYZ panel. **NOTE:** The z value will default to the value entered into the **Cad Control Bar** whether or not it is specified in the XYZ box. If no height value exists in the **Cad Control Bar**

The 2nd virtual point is selected in the same way as the 1st virtual point. The line is created after successful selection and acceptance of the 2nd virtual point.

Tangent

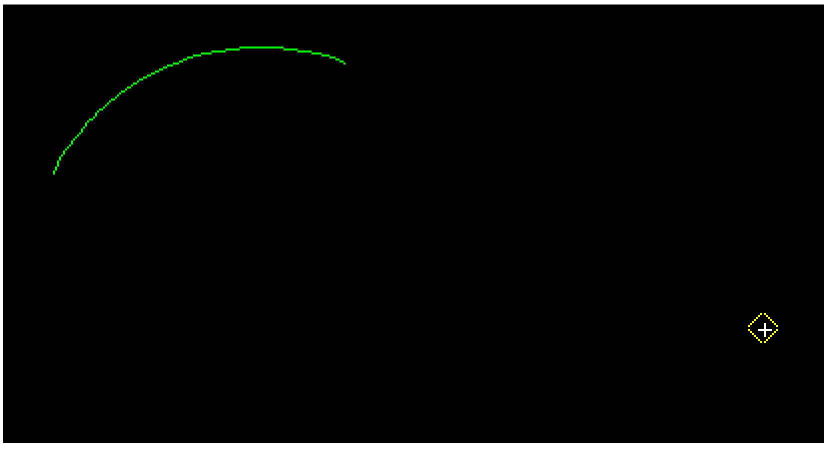
This option creates the virtual tangential line between two elements.

NOTE: When Selecting an arc or circle, the selection must be a line snap with direction. If a point is selected on the arc/circle the line will be draw between the selected points and not the tangent.

On selecting **Tangent**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

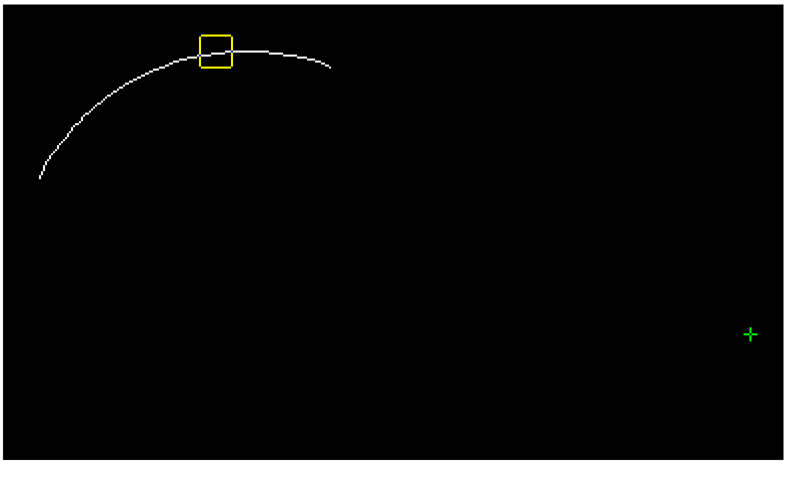
STEP 1:

A start position of the reference line is selected and accepted.



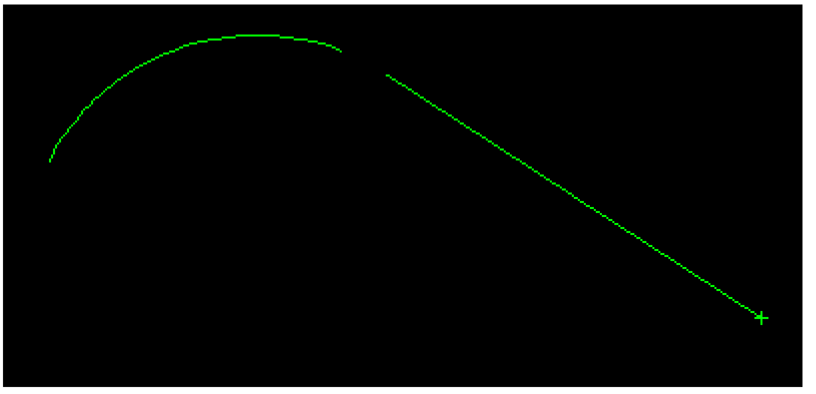
STEP 2:

The user selects and accepts the arc/circle segment with direction. The direction is required because there are two possible solutions. In this case, the direction was anti clockwise.



STEP 3:

After accepting the segment, a line is draw from the 1st selected point to the tangent point. Note that in this example, the arc is produced around so that a solution can be found.



Perpendicular

This option creates a virtual line by selecting a reference string and a reference point the create a line from the reference point perpendicular to the reference string.

On selecting **Perpendicular**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

A reference string is selected and accepted.

STEP 2:

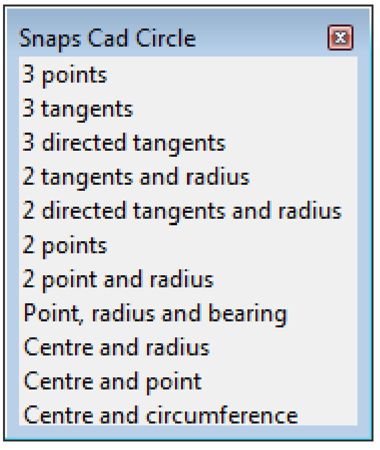
A line is drawn from the current mouse position perpendicular to the reference string

STEP 3:

After the final position is accepted, a virtual line is created which start from the selected position and perpendicular to the reference string.

4.28.5.3 Snaps Cad Circle

This section of documentation is a work in progress and will be updated in subsequent releases.



For the option *3 points, go to*	3 points
3 tangents	3 tangents
3 directed tangents	3 directed tangents
2 tangents and radius	2 tangents and radius
2 directed tangents and radius	2 directed tangents and radius
2 points	2 points
2 point and radius	2 point and radius
Point, radius and bearing	Point, radius and bearing
Centre and radius	Centre and radius
Centre and point	Centre and point
Centre and circumference	Centre and circumference

3 points

This option creates a virtual circle through three selected points.

On selecting **3 points**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

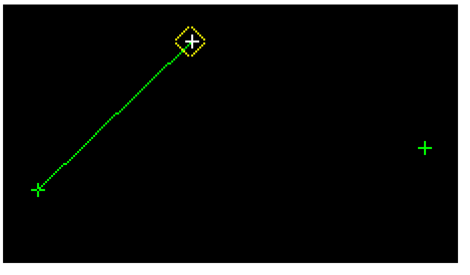
STEP 1:

The 1st point is selected with the mouse or entered in via the keyboard. To specify the 1st point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter the 1st point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the Enter key.

STEP 2:

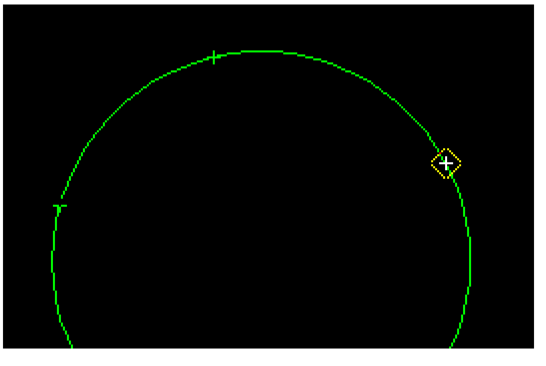
The 2nd point is selected and accepted.

After the 2nd point is accepted a circle will be displayed ‘rubber banding’ to the various solutions according to the position of the cursor. This will continue until the 3rd point is selected and accepted.



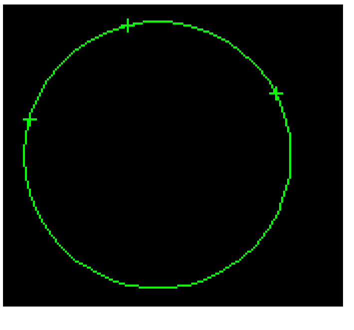
STEP 3:

The 3rd point is selected and accepted.



STEP 4:

A virtual circle is constructed through the three selected points



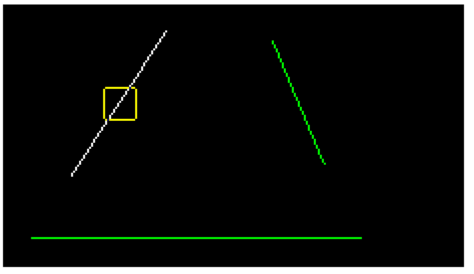
3 tangents

This option creates a virtual circle that is tangential to three selected segments.

On selecting **3 tangents**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

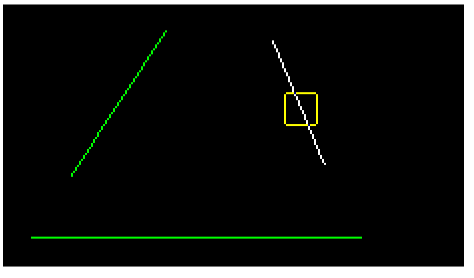
STEP 1:

The 1st tangent is selected with the mouse (Left Button) and accepted (Middle Button).



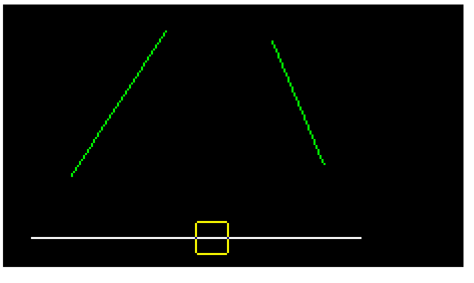
STEP 2:

The 2nd tangent is selected and accepted.



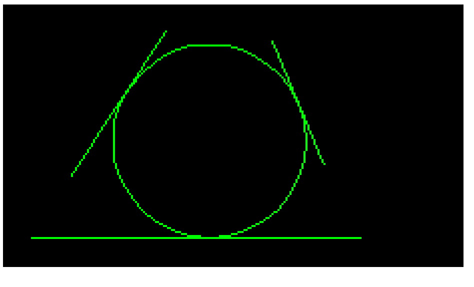
STEP 3:

The 3rd tangent is selected and accepted.



STEP 4:

A virtual circle is constructed that touches each of the three selected tangents.



3 directed tangents

This option creates a virtual circle that is tangential to three selected segments. The segments are selected in order and with direction and the circle is to the right of the direction of the selected segments.

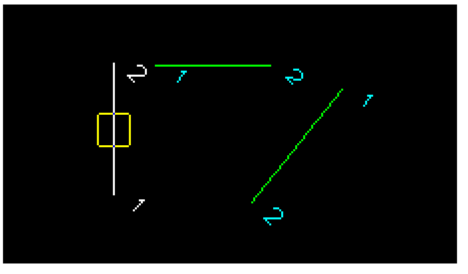
On selecting **3 directed tangents** the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

Select and accept the 1st tangent.

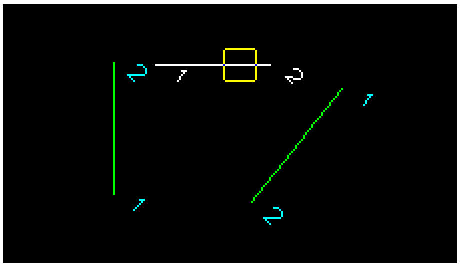
Note: For this option the direction of the selected tangents is important. The circle will be constructed to the right of a tangent. A user may reverse the direction of a tangent by selecting the tangent *with direction*. For further notes on picking tangents with direction, see Picking with Direction.

Note: The Vertex indices can be displayed by toggling the option on the Toggle Menu.



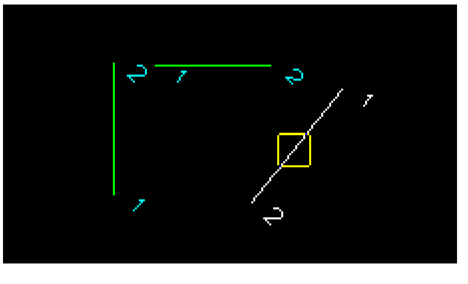
STEP 2:

The 2nd tangent is selected and accepted.



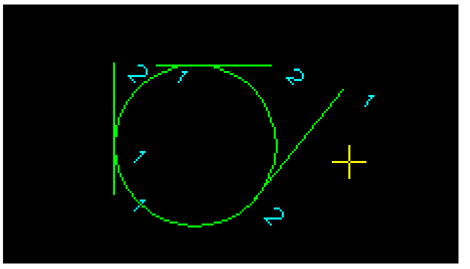
STEP 3:

The 3rd tangent is selected and accepted.



STEP 4:

If a solution exists, a virtual circle is constructed using the given information.



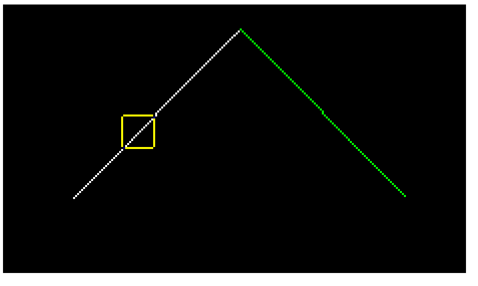
2 tangents and radius

This option creates a virtual circle with a given radius that is tangential to two selecting segments.

On selecting **2 tangents and radius**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

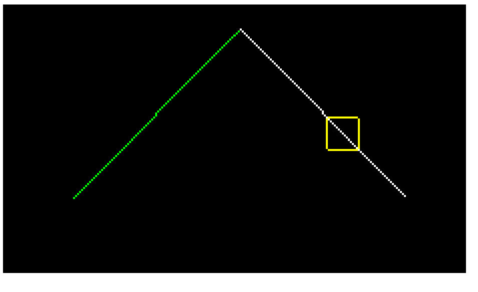
STEP 1:

Select the 1st tangent with the mouse (Left Button) and accept it (Middle Button).



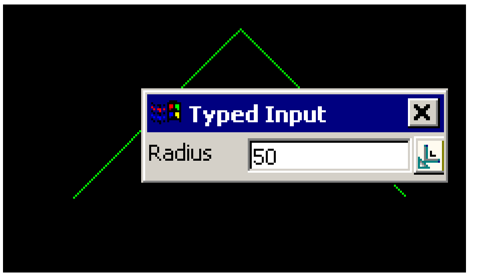
STEP 2:

Select the 2nd tangent and accept it.



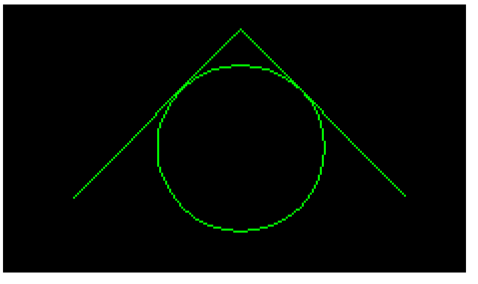
STEP 3:

After the 2nd tangent is accepted, the Radius Input box will appear. Type the radius value into the Input box and press the Enter key. The browse button  on the Input box can be used to define the radius by measuring existing elements.



STEP 4:

If a solution exists, a virtual circle is fitted touching the two selected tangents using the given radius.



2 directed tangents and radius

This option creates a virtual circle with a given radius that is tangential to two selected segments that are picked with direction.

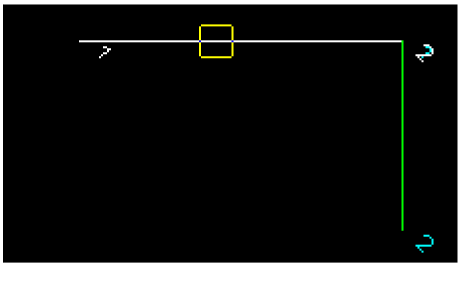
On selecting **2 directed tangents and radius**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

Select and accept the 1st tangent.

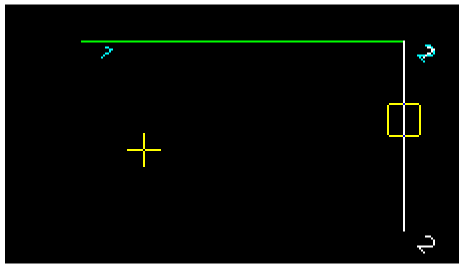
Note: For this option the direction of the selected tangents is important. The circle will be constructed to the right of a tangent. A user may reverse the direction of a tangent by selecting the tangent *with direction*. For further notes on picking tangents with direction, see Picking with Direction.

Note: The Vertex indices can be displayed by toggling the option on the Toggle Menu.



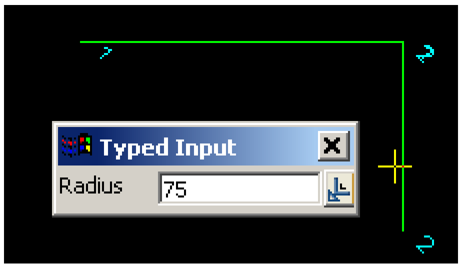
STEP 2:

The 2nd tangent is selected and accepted.



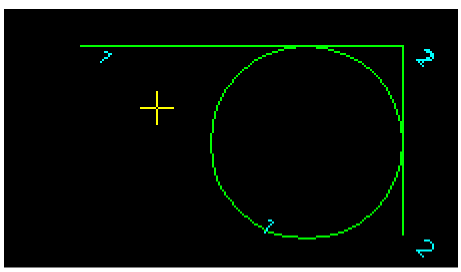
STEP 3:

After the 2nd tangent is accepted, the Radius Input box will appear. Type the radius value into the Input box and press the enter key. The browse button on the Input box can be used to define the arc radius by measuring existing elements.



STEP 4:

If a solution exists, a virtual circle is constructed using the given information.



2 points

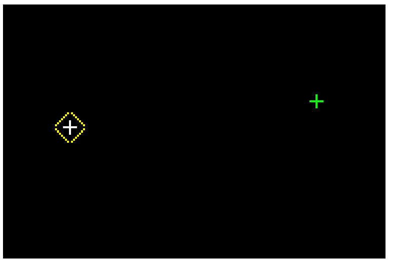
This option creates a virtual circle by selecting two points that define the diameter of the circle.

On selecting **2 points**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

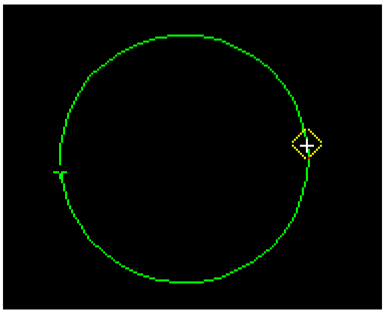
The 1st point is selected with the mouse or entered in via the keyboard. To specify the 1st point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter the 1st point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.

After the 1st point is accepted a circle will be displayed ‘rubber banding’ to the various solutions according to the position of the cursor (cursor position taken as the other end of a diameter). This will continue until the 2nd point is selected and accepted.



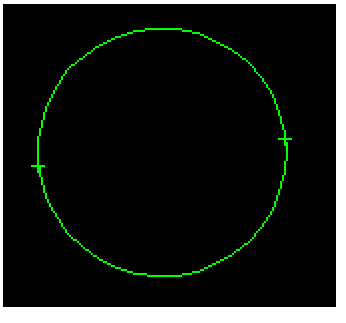
STEP 2:

The 2nd point is selected and accepted.



STEP 3:

The virtual circle is constructed through the two selected points. The two points define the diameter.



2 point and radius

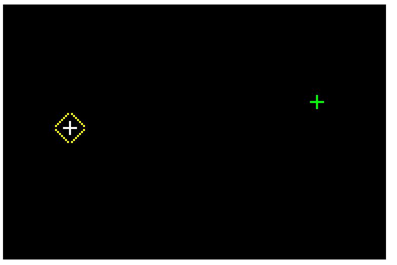
This option creates a virtual circle of a given radius that goes through two selected points.

On selecting **2 point and radius**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

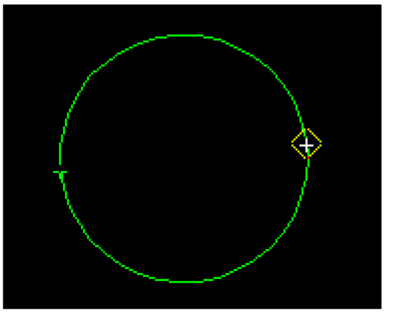
The 1st point is selected with the mouse or entered in via the keyboard. To specify the 1st point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter the 1st point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.

After the 1st point is accepted a circle will be displayed 'rubber banding' to the various solutions according to the position of the cursor. This will continue until the 2nd point is selected and accepted.



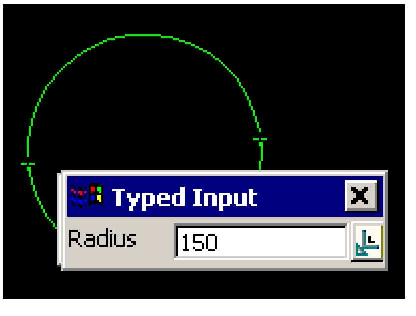
STEP 2:

The 2nd point is selected and accepted.



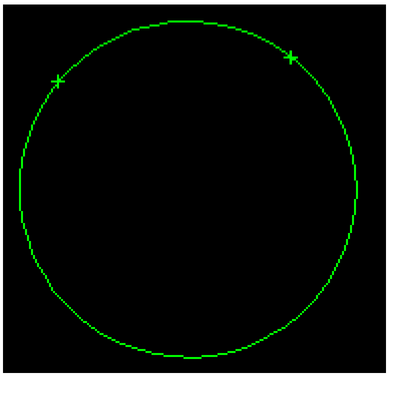
STEP 3:

After the 2nd point is accepted, the Radius Input box will appear. The radius value is entered into the input box followed by the Enter key.



STEP 4:

If a solution exists, the virtual circle is fitted through the two selected points using the given radius.



Point, radius and bearing

This option creates a virtual circle of a given radius, a selected point on the circle and the bearing of the tangent to the circle at that point.

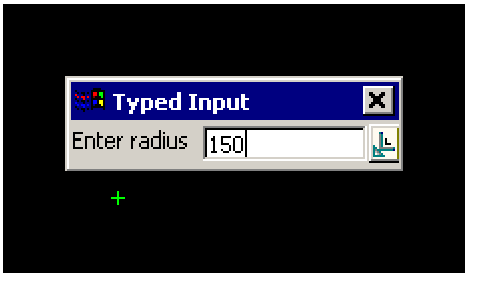
On selecting **Point, radius and bearing**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

The 1st point is selected with the mouse or entered in via the keyboard. To specify the 1st point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter a 1st point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the Enter key.

STEP 2:

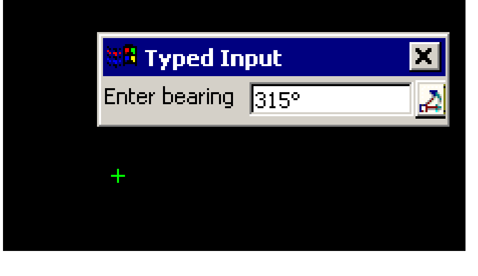
After the 1st point is accepted, the **Enter radius** input box will appear. Type the radius value into the Input box and press the enter key. The browse button on the Input box can be used to define the arc radius by measuring existing elements.



STEP 3:

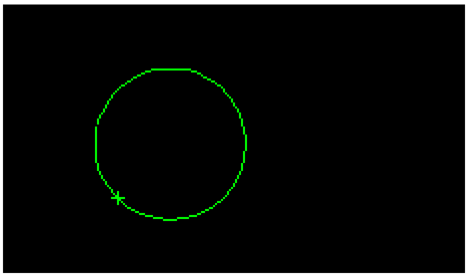
After the radius has been entered, the **Enter bearing** input box will appear. Type the bearing into the Input box and press the enter key. The browse button on the Input box can be used to define the bearing by measuring existing elements.

Note: **The Page Up and Page Down keys can be used when the Enter Bearing input box comes up to add or subtract intervals of 90 degrees.**



STEP 4:

If a solution exists, a virtual circle is fitted from the given bearing at the selected point using the selected radius.



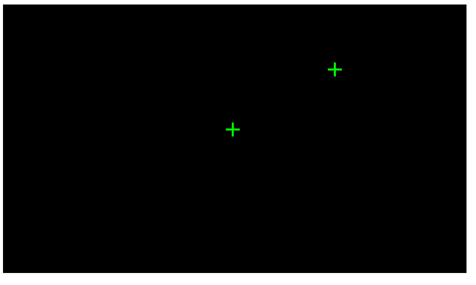
Centre and radius

This option creates a virtual circle of a given radius and a selected centre point

On selecting **Centre and radius**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

A centre point is selected with the mouse or entered in via the keyboard. To specify a centre point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter a centre point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the Enter key.



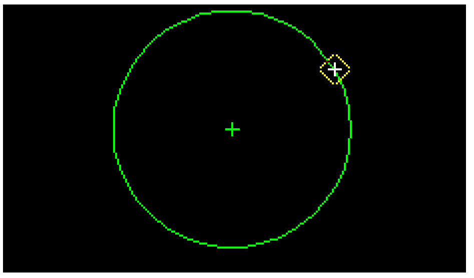
STEP 2:

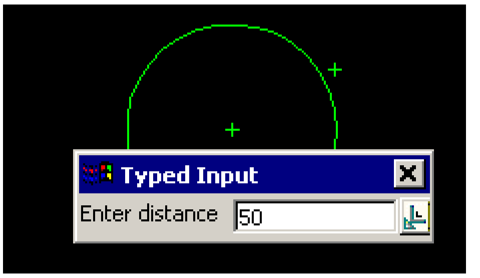
After the centre point is accepted a circle will be displayed ‘rubber banding’ to the various solutions according to the position of the cursor. This will continue until a radius value is entered.

A radius value is selected with the mouse or entered in via the keyboard.

To specify a radius value with the mouse, a point must be selected (Left Button) and accepted (Middle Button). This option also allows the definition of the radius by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next.

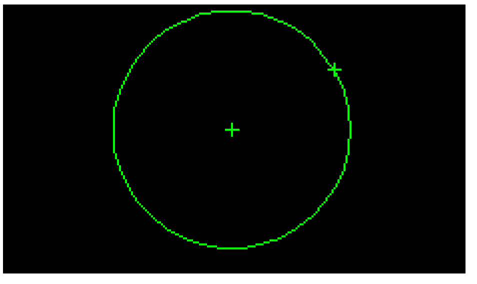
To enter a radius value with the keyboard, simply start typing or press the space bar to bring up the Radius Input box. Type the radius value into the Radius Input box and press the enter key. The browse button on the Input box can be used to define the arc radius by measuring existing elements.





STEP 3:

The virtual circle is constructed through the centre point using the specified radius.



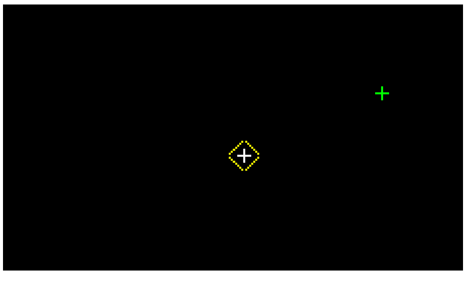
Centre and point

This option creates a virtual circle by selecting a centre point and then using the cursor to select a second point that in on the circle.

On selecting **Centre and point**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

A centre point is selected with the mouse or entered in via the keyboard. To specify a centre point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter a centre point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the Enter key.



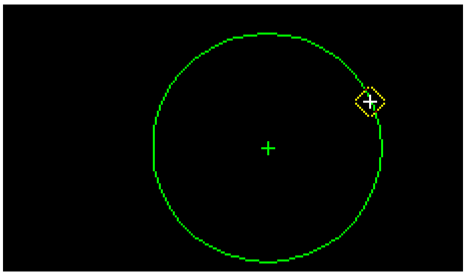
STEP 2:

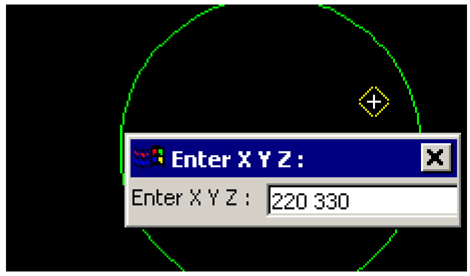
After the centre point is accepted a circle will be displayed 'rubber banding' to the various solutions according to the position of the cursor. This will continue until a radius value is entered.

A radius value is selected with the mouse or entered in via the keyboard.

To specify a radius value with the mouse, a point must be selected (Left Button) and accepted (Middle Button). This option also allows the definition of the radius by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next.

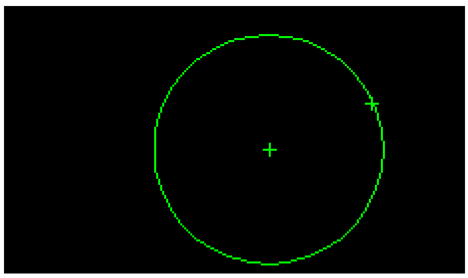
To enter a radius value with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the point into the Input box and press the Enter key.





STEP 3:

A virtual circle is constructed using the two points defined by the user.



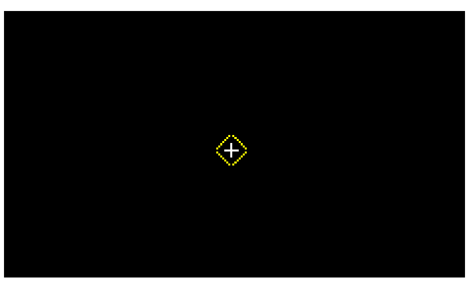
Centre and circumference

This option creates a virtual circle by selecting the centre point and giving a circumference value.

On selecting **Centre and circumference**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

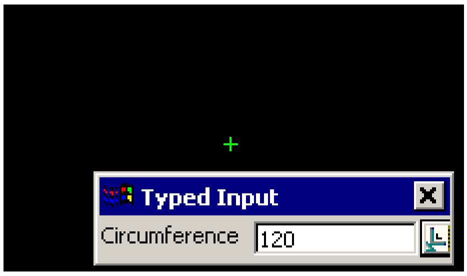
STEP 1:

A centre point is selected with the mouse or entered in via the keyboard. To specify a centre point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter a centre point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the Enter key.



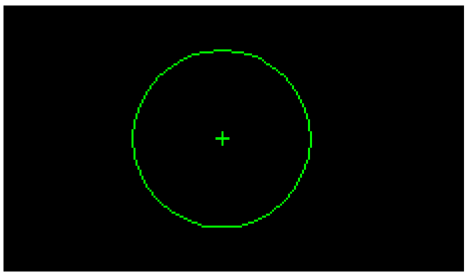
STEP 2:

After the centre point is accepted, the **Circumference** Input box will appear. Type the circumference length into the Input box and press the enter key. The browse button on the Input box can be used to define the arc radius by measuring existing elements.



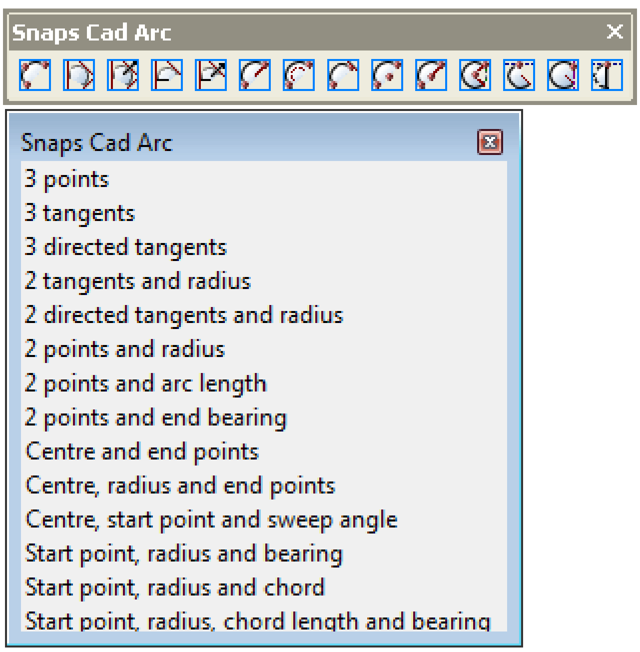
STEP 3:

A virtual circle is constructed using the centre point and the given circumference.



4.28.5.4 Snaps Cad Arc

This section of documentation is a work in progress and will be updated in subsequent releases.



For the option *3 points, go to* 3 Points
3 tangents 3 Tangents
3 directed tangents 3 directed tangents
2 tangents and radius 2 tangents and radius
2 directed tangents and radius 2 directed tangents and radius
2 points and radius 2 points and radius

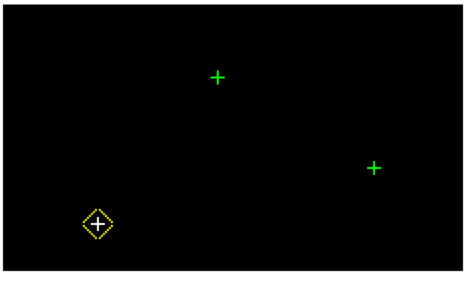
3 Points

This option creates a virtual arc through three selected points and the first and third points are the start and end of the arc.

On selecting **3 points**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

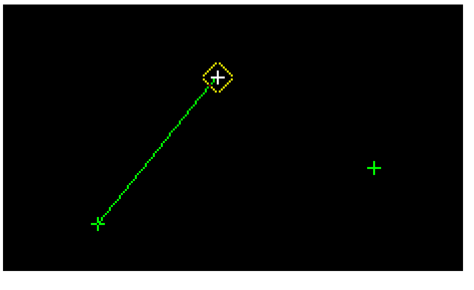
The 1st point is selected with the mouse or entered in via the keyboard. To specify the 1st point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter the 1st point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the Enter key.



STEP 2:

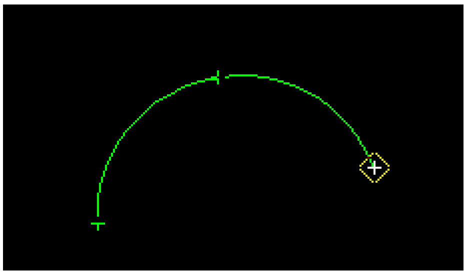
The 2nd point is selected and accepted.

After the 2nd point is accepted an arc will be displayed ‘rubber banding’ to the various solutions according to the position of the cursor. This will continue until the 3rd point is selected and accepted.



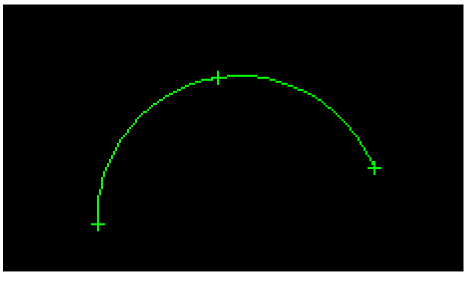
STEP 3:

The 3rd point is selected and accepted.



STEP 4:

A virtual arc is constructed through the three selected points.



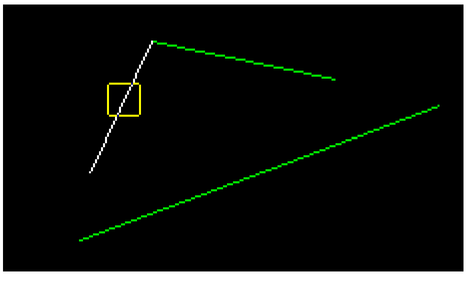
3 Tangents

This option creates a virtual arc that is tangential to three selected segments. The first and third segments are the start and end of the arc.

On selecting **3 tangents**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

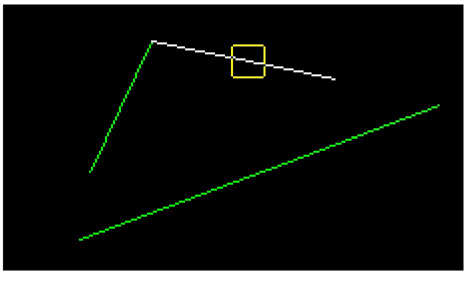
STEP 1:

The 1st tangent is selected with the mouse (Left Button) and accepted (Middle Button).



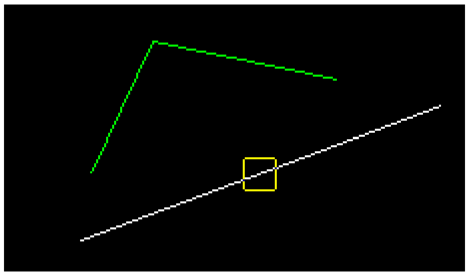
STEP 2:

The 2nd tangent is selected and accepted.



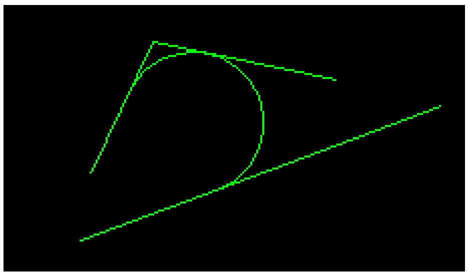
STEP 3:

The 3rd tangent is selected and accepted.



STEP 4:

The virtual arc is constructed touching the three selected tangents.



3 directed tangents

This option creates a virtual arc that is tangential to three selected segments. The segments are selected in order and with direction and the arc is to the right of the direction of the selected segments. The first and third segments are the start and end of the arc.

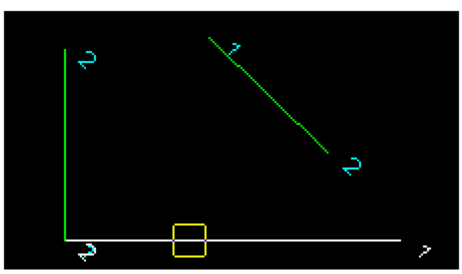
On selecting **3 directed tangents**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

Select and accept the 1st tangent.

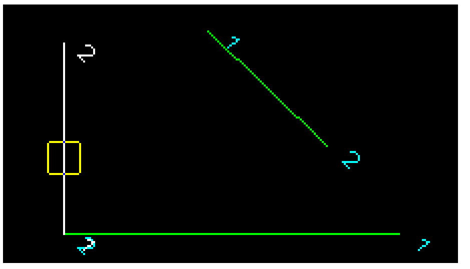
Note: For this option the direction of the selected tangents is important. The arc will be constructed to the right of the tangent. A user may reverse the direction of the tangent by selecting a tangent *with direction*. For further notes on picking tangents with direction, see Picking with Direction.

Note: The Vertex indices can be displayed by toggling the option on the Toggle Menu.



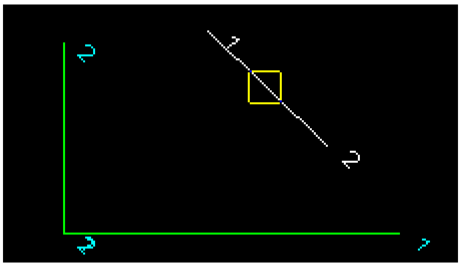
STEP 2:

The 2nd tangent is selected and accepted.



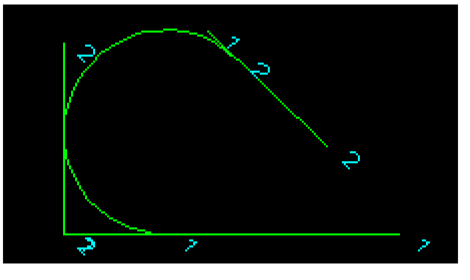
STEP 3:

The 3rd tangent is selected and accepted.



STEP 4:

If a solution exists, a virtual arc is constructed using the given information.



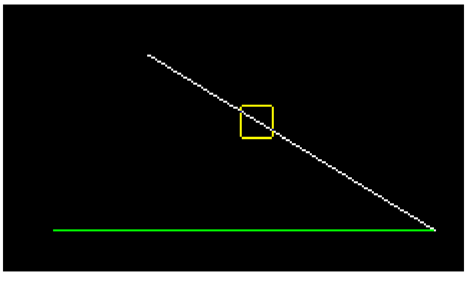
2 tangents and radius

This option creates a virtual arc with a given radius that is tangential to two selecting segments. This is the same as a fillet.

On selecting **2 tangents and radius**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

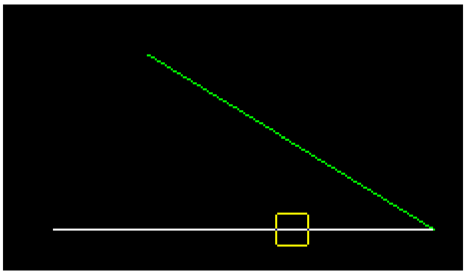
STEP 1:

The 1st tangent is selected with the mouse (Left Button) and accepted (Middle Button).



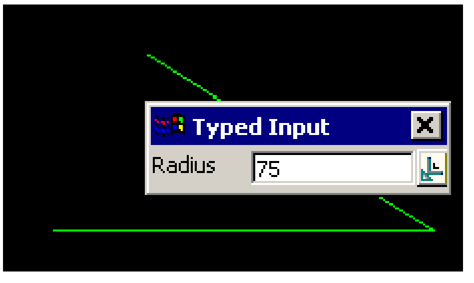
STEP 2:

The 2nd tangent is selected and accepted.



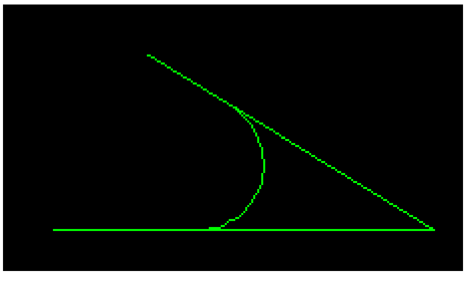
STEP 3:

After the start point is accepted, the Radius Input box will appear. Type the radius value into the Input box and press the enter key. The browse button on the Input box can be used to define the arc radius by measuring existing elements.



STEP 4:

If a solution exists, the virtual arc is fitted through the two selected tangents using the given radius.



2 directed tangents and radius

This option creates a virtual arc with a given radius that is tangential to two selected segments that are picked with direction.

On selecting **2 directed tangents and radius**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

Select and accept the 1st tangent.

Note: For this option the direction of the selected tangents is important. The arc will be constructed to the right of a tangent. A user may reverse the direction of a tangent by selecting the tangent *with direction*. For further notes on picking tangents with direction, see Picking with Direction.

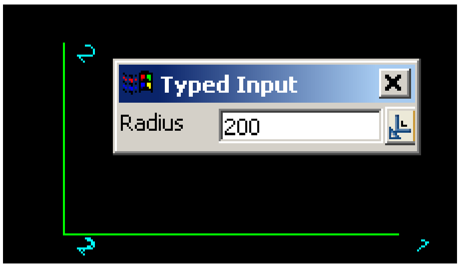
Note: The Vertex indices can be displayed by toggling the option on the Toggle Menu.

STEP 2:

The 2nd tangent is selected and accepted.

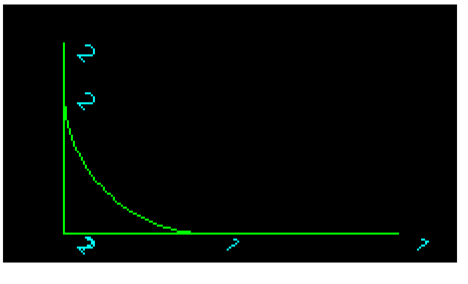
STEP 3:

After the start point is accepted, the Radius Input box will appear. Type the radius value into the Input box and press the Enter key. The browse button  on the Input box can be used to define the arc radius by measuring existing elements.



STEP 4:

If a solution exists, the virtual arc is fitted through the two selected tangents using the given radius.



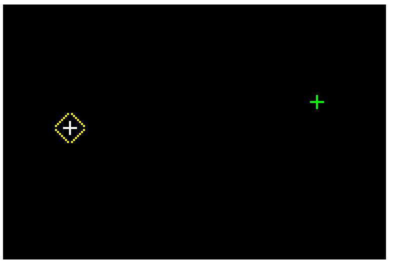
2 points and radius

This option creates a virtual arc of a given radius that starts and ends on two selected points.

On selecting **2 points and radius**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

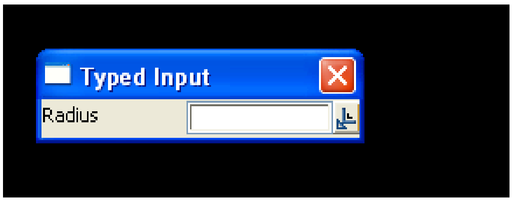
STEP 1:

The 1st point is selected with the mouse or entered in via the keyboard. To specify the 1st point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter the 1st point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the Enter key.



STEP 2:

After the start point is accepted, the Radius Input box will appear. Type the radius value into the Input box and press the enter key. The browse button  on the Input box can be used to define the arc radius by measuring existing elements.

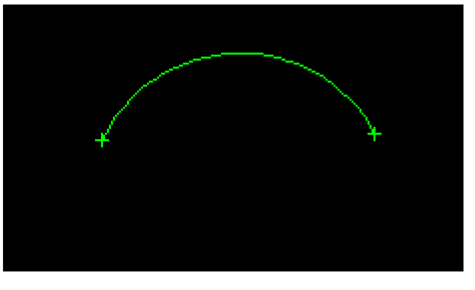


STEP 3:

The 2nd point is selected and accepted.

STEP 4:

If a solution exists, an arc is constructed using the given information.



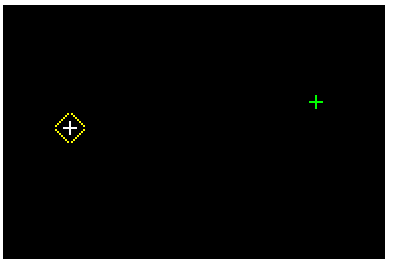
2 points and arc length

This option creates a virtual arc of a given arc length that starts and ends on two selected points.

On selecting **2 points and arc length**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

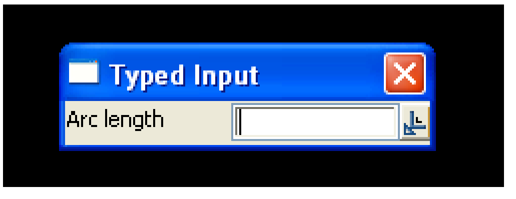
STEP 1:

The 1st point is selected with the mouse or entered in via the keyboard. To specify the 1st point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter the 1st point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the Enter key.



STEP 2:

After the start point is accepted, the Arc length Input box will appear. Type the arc length value into the Input box and press the enter key. The browse button  on the Input box can be used to define the arc length by measuring existing elements.

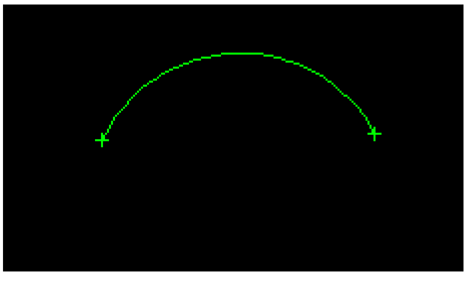


STEP 3:

The 2nd point is selected and accepted.

STEP 4:

If a solution exists, an arc is constructed using the given information.



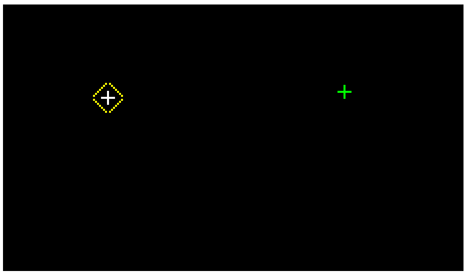
2 points and end bearing

This option creates a virtual arc that starts and end on two selected points and has a given bearing of the tangent at the end point (the end bearing).

On selecting **2 points and end bearing**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

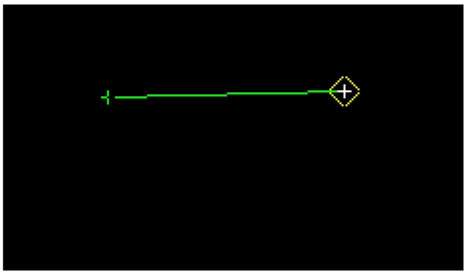
STEP 1:

The 1st point is selected with the mouse or entered in via the keyboard. To specify the 1st point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter the 1st point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the Enter key.



STEP 2:

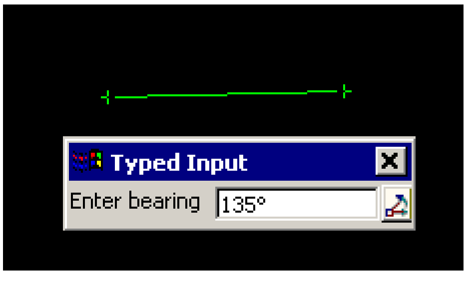
The 2nd point is selected and accepted.



STEP 3:

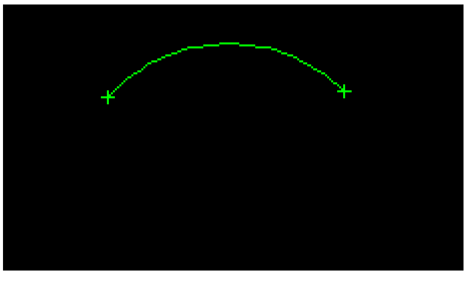
After the 2nd point is accepted, the Enter bearing input box will appear. Type the bearing into the Input box and press the enter key. The browse button on the input box can be used to define the bearing by measuring existing elements.

Note: **The Page Up and Page Down keys can be used when the Enter Bearing input box comes up to add or subtract intervals of 90 degrees.**



STEP 4:

If a solution exists, the arc is fitted through the two selected points with the end point of the given bearing.



Centre and end points

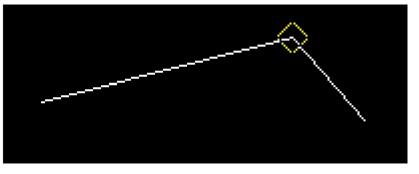
This option creates a virtual arc by selecting in order, the centre point and the start and end points. The radius of the arc is the distance between the centre and the start point.

On selecting **Centre and end points**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

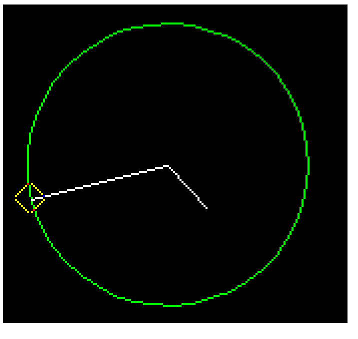
A centre point is selected with the mouse or entered in via the keyboard. To specify a centre point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter a centre point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the Enter key.

After the centre point is accepted a circle will be displayed 'rubber banding' to the various solutions according to the position of the cursor. This will continue until the start point is selected and accepted.



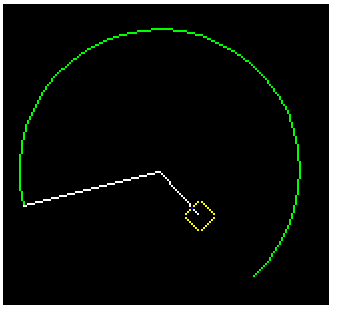
STEP 2:

Select and accept the start point. This point defines the radius and the start of the arc.



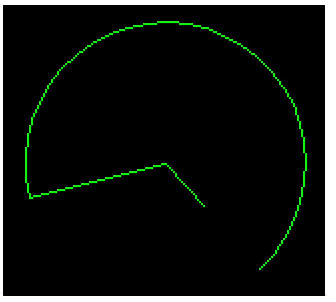
STEP 3:

The end point of the arc is specified.



STEP 4:

The virtual arc is constructed using the given information.



Centre, radius and end points

This option allows the creation of a virtual arc given a centre point, radius, start and end points.

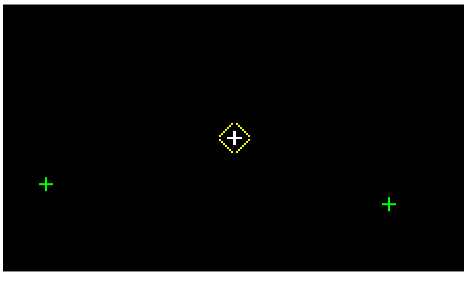
This option creates an arc of a given radius by selecting in the centre point and the start and end positions.

On selecting **Centre, radius and end points** the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

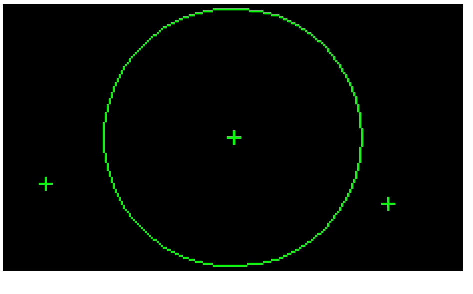
A centre point is selected with the mouse or entered in via the keyboard. To specify a centre point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter a centre point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the Enter key.

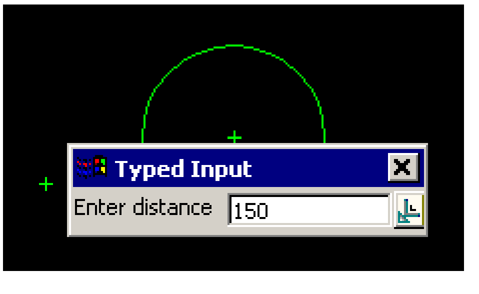
After the centre point is accepted a circle will be displayed 'rubber banding' to the various solutions according to the position of the cursor. This will continue until the start point is selected and accepted.



STEP 2:

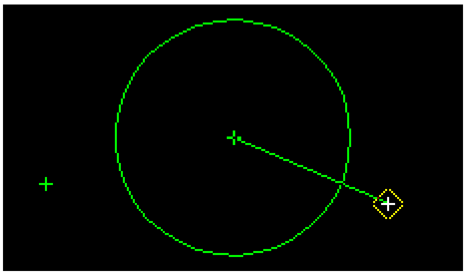
A radius value is selected with the mouse or entered in via the keyboard. To specify a radius value with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter radius value with the keyboard, simply start typing or press the space bar to bring up the Radius Input box. Type the radius value into the Radius Input box and press the Enter key. The browse button on the Input box can be used to define the arc distance by measuring existing elements.

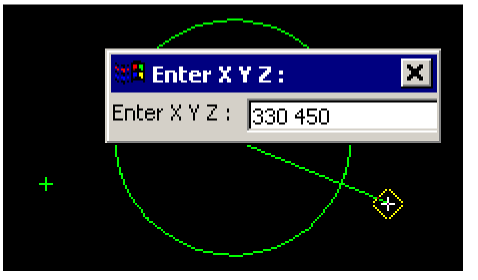




STEP 3:

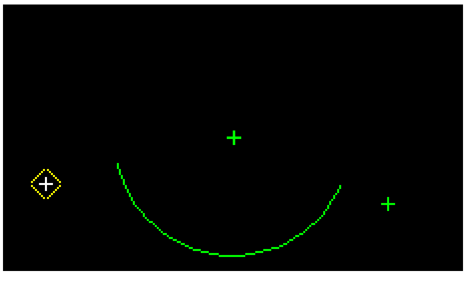
A start point defines the start of the sweep angle to define where to start the arc. It can be selected with the mouse or entered in via the keyboard. To specify a start point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter a start point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the Enter key.

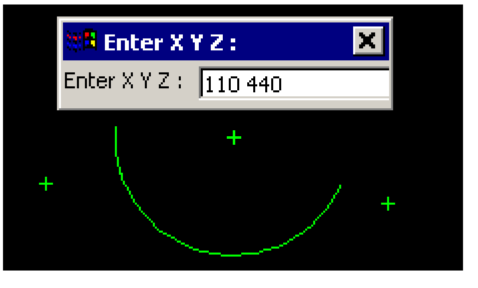




STEP 4:

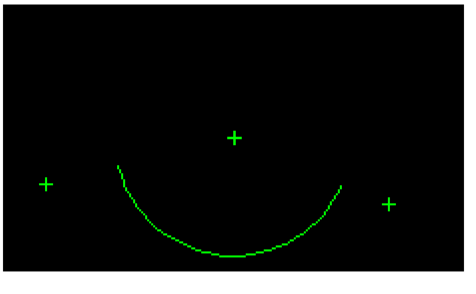
The end point of the arc is specified. The start and end points define the sweep angle to define the arc. The rubber banding of the arc will use the cursor position as the end position until a end point is selected and accepted. It can be selected with the mouse or entered in via the keyboard. To specify a start point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter a start point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key. The end point does not have to be on the arc itself. It is used to define the sweep angle.





STEP 5:

The arc is constructed using the given information.



Centre, start point and sweep angle

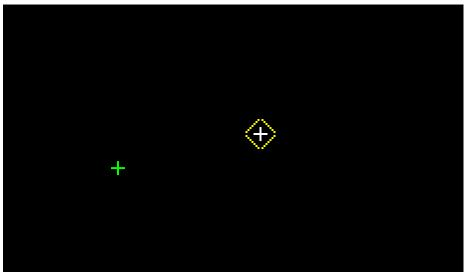
This option creates a virtual arc by selecting a centre point and a start points and giving a sweep angle. The radius of the arc is the distance between the centre and start point.

On selecting **Centre, start point and sweep angle**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

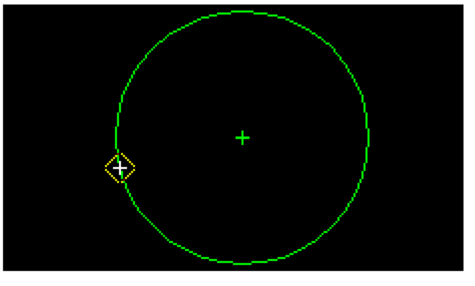
A centre point is selected with the mouse or entered in via the keyboard. To specify a centre point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter a centre point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the Enter key.

After the 1st point is accepted a circle will be displayed 'rubber banding' to the various solutions according to the position of the cursor. This will continue until the 2nd point is selected and accepted.



STEP 2:

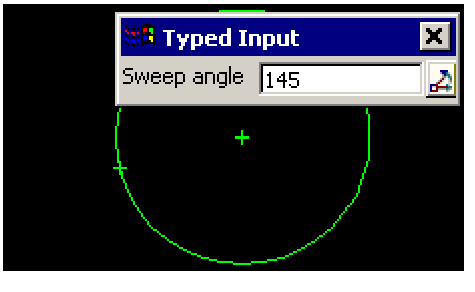
The 2nd point is selected with the mouse or entered in via the keyboard. This point defines the radius as well as the start point of the arc.To specify the 2nd point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter the 2nd point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the Enter key.



STEP 3:

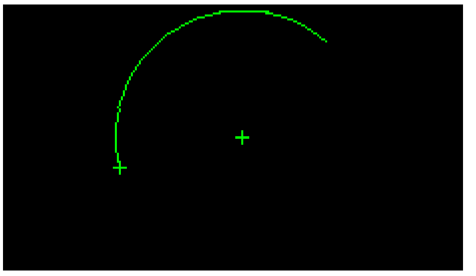
After the 2nd point is accepted, the **Sweep Angle** input box will appear. Type the sweep angle value into the input box and press the enter key. The browse button on the Input box can be used to define the arc radius by measuring existing elements.

Note: **The Page up and page down keys can be used when the input angle box comes up to add or subtract intervals of 90 degrees.**



STEP 4:

The virtual arc is created using the information supplied



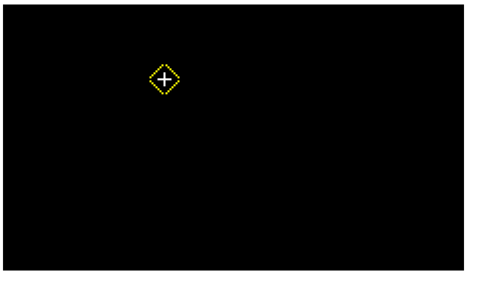
Start point, radius and bearing

This option creates a virtual arc of a given radius, starting at a selected point and bearing of the tangent at the start point and going for a given arc length.

On selecting the **Start point, radius and bearing** option, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

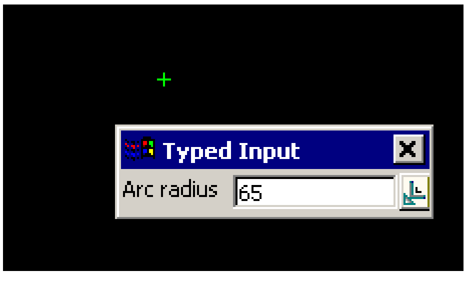
STEP 1:

A start point is selected with the mouse or entered in via the keyboard. To specify a start point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter a start point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the Enter key.



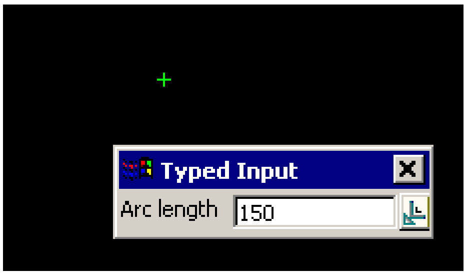
STEP 2:

After the start point is accepted, the Arc Radius Input box will appear. Type the radius value into the Input box and press the enter key. The browse button on the Input box can be used to define the arc radius by measuring existing elements.



STEP 3:

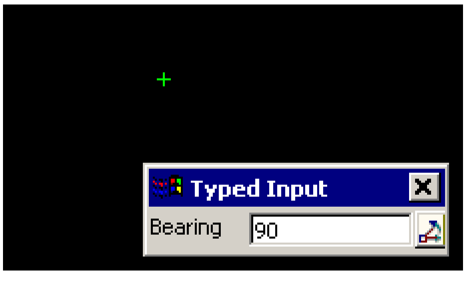
After the radius has been entered, the **Arc Length** Input box will appear. Type the arc length into the Input box and press the enter key. The browse button on the Input box can be used to define the arc distance by measuring existing elements.



STEP 4:

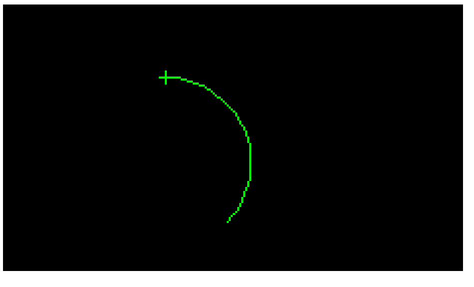
After the arc length is accepted, the Bearing Input box will appear. Type the bearing into the Input box and press the enter key. The browse button  on the Input box can be used to define the bearing by measuring existing elements.

Note: **The Page Up and Page Down keys can be used when the Bearing Input box comes up to add or subtract intervals of 90 degrees.**



STEP5:

The virtual arc is created with the given information.



Start point, radius and chord

This option creates a virtual arc of a given radius, starting from a selected point and with a given chord bearing at the start point and a given arc length.

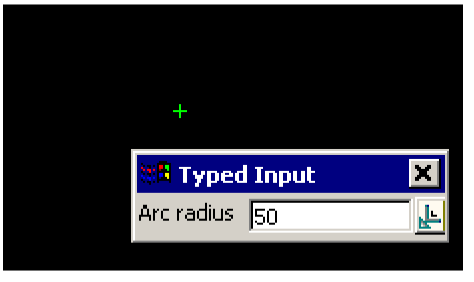
On selecting **Start point, radius and chord**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

The start point is selected with the mouse or entered in via the keyboard. To specify the start point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter the start point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the Enter key.

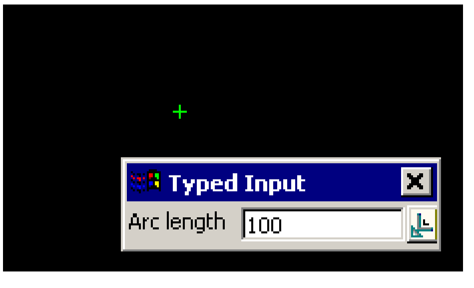
STEP 2:

After the start point is accepted, the **Arc Radius** Input box will appear. Type the radius value into the Input box and press the enter key. The browse button on the Input box can be used to define the arc radius by measuring existing elements.



STEP 3:

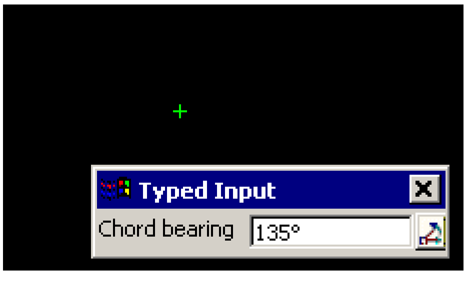
After the arc radius is entered, the **Arc Length** Input box will appear. Type the arc length into the Input box and press the enter key. The browse button on the Input box can be used to define the arc distance by measuring existing elements.



STEP 4:

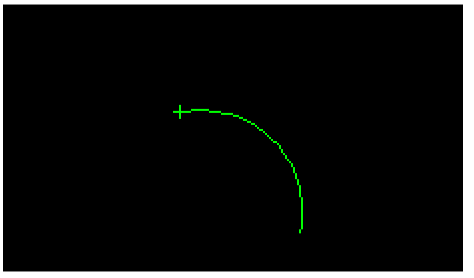
After the arc length is entered, the **Chord Bearing** Input box will appear. Type the bearing into the Input box and press the enter key. The browse button on the Input box can be used to define the bearing by measuring existing elements.

Note: **The Page Up and Page Down keys can be used when the Chord Bearing Input box comes up to add or subtract intervals of 90 degrees.**



STEP 5:

A virtual arc is created with the given information.



Start point, radius, chord length and bearing

This option creates a virtual arc of a given radius, starting from a selected point, with a given bearing of the tangent at the start point and a given chord length from the start point.

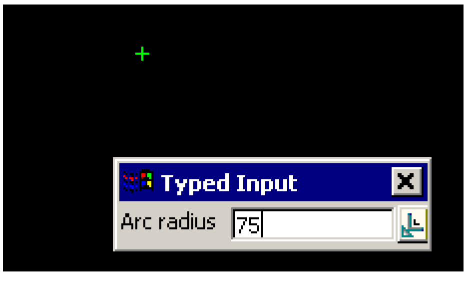
On selecting **Start point, radius, chord length and bearing**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

A start point is selected with the mouse or entered in via the keyboard. To specify a start point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter a start point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.

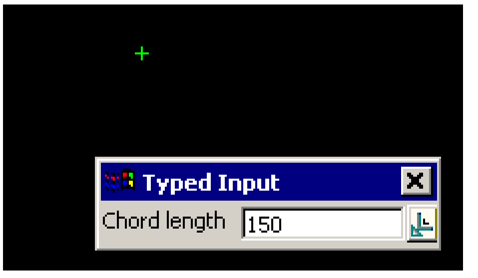
STEP 2:

After the start point is accepted, the **Arc Radius** Input box will appear. Type the radius value into the Input box and press the enter key. The browse button on the Input box can be used to define the arc radius by measuring existing elements.



STEP 3:

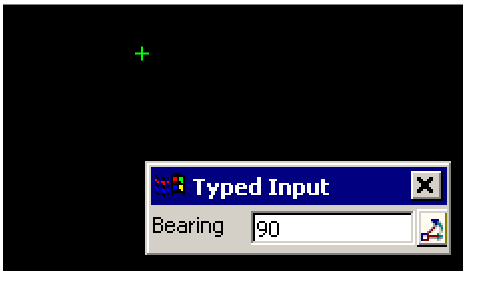
After the radius has been entered, the **Chord Length** Input box will appear. Type the chord length into the Input box and press the enter key. The browse button on the Input box can be used to define the chord length by measuring existing elements.



STEP 4:

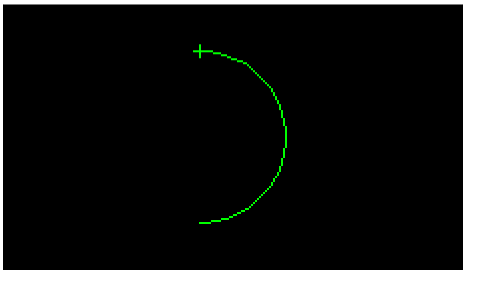
After the chord length has been entered, the **Bearing** Input box will appear. Type the bearing of the start tangent into the Input box and press the enter key. The browse button on the Input box can be used to define the bearing by measuring existing elements.

Note: **The Page Up and Page Down keys can be used when the Bearing Input box comes up to add or subtract intervals of 90 degrees.**



STEP 5:

The virtual arc is created with the given information.



4.28.6 String Edits Toolbar

Position of option on menu: **various**

The **String edits** toolbar is:

Options on Toolbar

edit string

edit vertex

edit segment properties

edit vertex properties

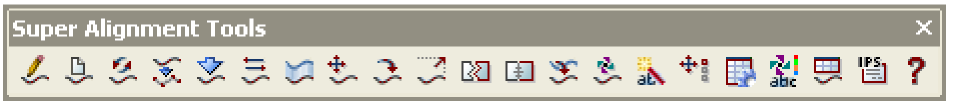
change strings

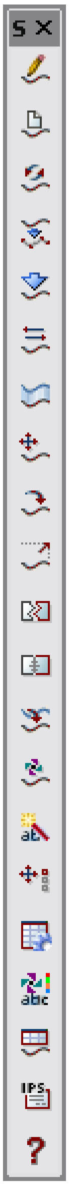
For the option *Edit string,* go to	14.4 Editor
Edit vertex	14.5.11 Edit Vertex
Edit segment properties	14.12.4 Segment
Edit vertex properties	14.12.3 Vertex
Change string	14.6.1 Change

4.28.7 Super Alignment Tools Toolbar

Position of option on menu: various

The **Super Alignment Tool** toolbar is:





Options on Toolbar

edit super alignment
create super alignment
resolve super alignment
fixed/floating to IPs
dereference a super alignment
reverse super alignment
parallel a super alignment
translate super alignment
rotate super alignment
scale super alignment
split a super alignment
join a super alignment
copy vertical super alignment
change super alignment style
explode super alignment labels
move super alignment point settings
define road design parameters
super alignment style create/edit
alignment/super alignment table
super alignment IP tabulation
help

For edit super alignment go to	14.4.11 Edit Super Alignment
create super alignment	14.2.1 Create Super Alignment
resolve super alignment	14.2.2.3 Resolve a Super Alignment
fixed/floating to IP’s	14.2.2.13 Convert Fixed/Floating Elements to IPs
dereference a super alignment	14.2.2.4 Dereference Super Alignments
reverse super alignment	14.2.2.5 Reverse Super Alignment
parallel a super alignment	14.2.2.6 Parallel a Super Alignment
translate super alignment	14.2.2.7 Translate Super Alignment
rotate super alignment	14.2.2.8 Rotate Super Alignment

scale super alignment	14.2.2.9 Scale Super Alignment
split a super alignment	14.2.2.10 Split a Super Alignment
join a super alignment	14.2.2.11 Joining Two Super Alignments
copy vertical super alignment	14.2.2.12 Copy Vertical Geometry
change super alignment style	14.2.2.14 Change the Styles for Selected Super Alignments
explode super alignment labels	14.2.2.15 Explode the Super Alignments Labelling
move super alignment point settings	14.2.2.16 Constrain the Movement of Super Alignment Points
define road design parameters	14.2.8 Define Design Parameters - from V9
super alignment style create/edit	14.2.3 Super Alignment Style
alignment/super alignment table	24.16.14 Alignment & Super Alignment Table
super alignment IP tabulation	14.2.11 Super Alignment IP Tabulation
help	14.2.9 SA Element Parts Viewer

4.28.8 Survey Reductions Toolbar

Position of option on menu: various

The **Survey Reductions** toolbar is:

Options on Toolbar

survey menu

create control stations

bearing/distance entry

traverse adjustment

horizontal least squares adjustment

level nets adjustment

set up data collector

download survey data

convert raw survey fie to 12d field file

survey data reduction function

quick change code

join surveyed strings

insert target height

start new survey strings

auto order survey shots

order by survey point ids

reset order to original order

edit survey data

edit survey field data by string

lock a function

edit a text file

For the option *Survey menu,* go to	17 Survey
Create control stations	14.1.12 Create - Control Stations
Bearing/distance entry	17.14.2 Plane Bearing/Distance Entry
Traverse adjustment	17.9.6 Traverse Adjustment
Horizontal least squares	17.9.3 Least Squares Network
Level network adjustment	17.9.4 Level Network
Set default data collector	17.3 Setup
Download from data collector	17.4 Download Raw

Convert data collector file to field file	17.5 Convert Raw
Create survey data reduction function	17.6 Create Survey Function
Survey code change	17.7.4.2 Quick Change
Join strings	17.7.9.4 Join Two Strings of Same Feature Code
Insert target height	17.7.10.1 Insert Target Height
Start new string	17.7.9.1 New String
Auto order string points	17.7.7.3 Auto Order
Order by points	17.7.7.1 By Points
Remove ordering	17.7.7.4 Remove Order
Edit survey data	17.7.2 Field Data
Edit field data by picking a string	17.7.3 Field Data by String
Lock a function	28.4.2 Lock
Edit a report file	27.1 Edit

4.28.9 Tin Utility Toolbar

Position of option on menu: **various**

The **Tin Utility** toolbar is:

Options on Toolbar

create a tin

quick triangulation selection

null a tin by angle and length

tin to tin depth polygons

contour, smooth and label a tin

intersect two tins

translate/copy a tin

For the option *Create a tin,* go to	16.5.1 Triangulate Data
Quick triangulation selection	16.5.4 Quick Tin
Null by angle and length	16.13.5 Null by Angle and Length
Tin to tin depth polygons	16.11.8 Depth Range Polygons
Contour, smooth and label	16.9.2 Contour, Smooth and Label
Intersect two tins	16.11.3 Intersection
Translate/copy a tin	16.14.13 Translate/Copy

4.28.10 Traverse Spreadsheet Toolbar

Position of option on menu: various

The **Traverse spreadsheet** toolbar is:

Options on Toolbar

edit traverse spreadsheet

create a traverse spreadsheet

traverse spreadsheet utilities

lot check for tss

New Zealand traverse spreadsheet report

standard traverse spreadsheet report

drafting of traverse spreadsheet

drafting of radiation tables

read a Landonline XML file

write a Landonline XML file

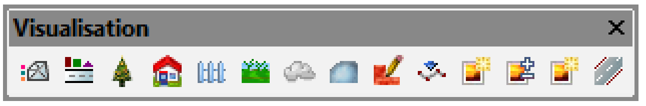
set up parameters for traverse spreadsheet

For the option *Edit traverse spreadsheet*, go to	17.12.3 TSS Edit
Create a traverse spreadsheet	17.12.2 TSS Create
Traverse spreadsheet utilities	17.12.4 TSS Utilities
Lot check for traverse spreadsheet	17.12.5 Lot Check
NZ traverse spreadsheet report	17.12.6 NZ TSS Report
Traverse spreadsheet report	17.12.7 Standard TSS Report
Traverse spreadsheet drafting	17.12.8 TSS Drafting
TSS radiation tables drafting	17.12.9 TSS Radiation Table Drafting
Read Landonline XML	17.12.10 Landonline XML Read
Write Landonline XML	17.12.11 Landonline XML Write
Set up tss parameters	17.12.1 TSS Parameters

4.28.11 Visualisation Toolbar

Position of option on menu: various

The **Visualisation spreadsheet** toolbar is:



Options on Toolbar

tin render settings

create roadside furniture

create trees and shrubs

create houses

create fences

create forest

create clouds

create sky dome

create/edit a texture map file

change polygon drape

create and place billboards

change super string billboards

12d - billboard signs create

new linemarking

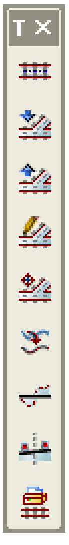
For the option *Tin render settings*, go to 16.6.7 Render Settings

Create roadside furniture	12.13.8.3 Roadside Furniture
Create trees and shrubs	12.13.5.5 Trees/Shrubs as Faces and Billboards
Create houses	12.13.35.4 Houses
Create fences	12.13.35.3 Fences
Create forest	12.13.5.4 Create Forest
Create clouds	12.13.35.1 Clouds
Create sky dome	12.13.35.2 Sky dome
Create/edit texture map file	12.13.4 Texture Map Edit
Change polygon drape	12.13.3 Render Drape
Create and place billboards	12.13.5.1 Create and Place Billboard
Change super string billboards	12.13.5.2 Add and Remove Billboard from Super String
12d - billboard signs create	12.13.5.3 12D - Billboard Signs Create
New linemarking	12.13.35.6 Line Marking (New)

4.28.12 Track Toolbar

Position of option on menu: **various**

The Track toolbar is:



Options on Toolbar

calc cl panel

read turnouts

write turnouts file

turnouts create/edit

turnout place

copy vc

rail slew calculator

cant panel

plot rails panel

For the option calc cl, go to	20.11.1 Calc CL Panel
read turnouts	20.11.2.3 Read Turnouts
write turnouts file	20.11.2.4 Write Turnouts File
turnouts create/edit	20.11.2.2 Turnouts Create/ Edit
turnout place	20.11.2.1 Turnout Place
copy vc	20.11.3 Copy VC
rail slew calculator	20.11.4 Rail Slew Calculator
calculate cant	20.11.5 Calculate Cant
plot rails panel	20.11.7.1 Plot Rails Panel

To return to the beginning of this chapter, click on 4 Tools and Concepts.

4.29 Drag and Drop

Some files can be dragged and dropped onto ***12d Model*** and panels automatically open for them, and some can be dragged and dropped as attachments to Outlook 2002 and above.

.project files can be dragged and dropped to open a new project and files can be dragged and dropped onto a file box and the full path name to the file will be automatically entered into the file box.

The files that can be dragged and dropped include

4.29.1 File Box

If you drag a file from *Windows Explorer* onto a **File Box** on any panel, the full file path name of the file is entered into the **File Box**.

Return to 4.29 Drag and Drop.

4.29.2 .12daz, 12da Files

If you drag a ***.12daz*** or ***.12daz*** file onto an open ***12d Model*** project, it opens up the **Read 12d Solutions Archive Data** panel with the full path name of the ***.12daz*** or ***.12da*** file entered into the **File to read** box.

This will work for files from folders, 12d Synergy, Skype and most other places that allow drag and drop.

It will also work for files that are email attachments in *Outlook 2002 and above.*

If you are dragging and dropping more than one ***12daz*** or ***12da*** file at a time, then the one **Read 12d Solutions Archive Data** panel will be opened and all the dropped files listed in the ***Many Files*** grid.

Return to 4.29 Drag and Drop.

4.29.3 .dat Files

If you drag a ***.dat*** file onto an open ***12d Model*** project, it opens up the **Read x y z s Data** panel with the full path name of the ***.dat*** file entered into the **File to read** box.

This will work for files from folders, Skype and most other places that allow drag and drop.

It will also work for files that are email attachments in *Outlook 2002 and above.*

If you are dragging and dropping more than one .dat file at a time, then the one **Read x y z s Data** panel will be opened and all the dropped files listed in the ***Many Files*** grid.

Return to 4.29 Drag and Drop.

4.29.4 .chain,.rcn Files

If you drag a ***.chain*** or ***.rcn*** file onto an open ***12d Model*** project, it opens up the **Create/Edit Chain** panel with the full path name of the ***.chain*** or ***.rcn*** file entered into the **Chain file** box and the chain file is automatically read in.

This will work for files from folders, 12d Synergy, Skype and most other places that allow drag and drop.

If you are dragging and dropping more than one **chain** file at a time, then a separate **Create/Edit Chain** panel is opened for each chain file.

Return to 4.29 Drag and Drop.

4.29.5 .mtf Files

If you drag a ***.mtf*** file onto an open ***12d Model*** project, it opens up the mtf in the **MTF Edit** panel.

This will work for files from folders, 12d Synergy, Skype and most other places that allow drag and drop.

If you are dragging and dropping more than one **mtf** file at a time, then a separate **MTF Edit** panel is opened for each mtf file.

Return to 4.29 Drag and Drop.

4.29.6 .mapfile Files

If you drag a ***.mapfile*** file onto an open ***12d Model*** project, it opens up the **Map File Create/ Edit** panel with the full path name of the ***.mapfile*** file entered into the **Map file** box and the Map File is automatically read in.

This will work for files from folders, 12d Synergy, Skype and most other places that allow drag and drop.

If you are dragging and dropping more than one **Map File** at a time, then a separate **Map File Create/Edit** panel is opened for each Map File.

Return to 4.29 Drag and Drop.

4.29.7 .slx Files

If you drag an ***.slx*** file onto an open ***12d Model*** project, it opens up all the panels as specified in the **SLX** file.

This will work for files from folders, 12d Synergy, Skype and most other places that allow drag and drop.

If you are dragging and dropping more than one **SLX** at a time, then all the panel in all the SLX files are opened.

Return to 4.29 Drag and Drop.

4.29.8 .4do Files

If you drag a ***.4do*** file onto an open ***12d Model*** project, it runs the macro.

This will work for files from folders, 12d Synergy, Skype and most other places that allow drag and drop.

If you are dragging and dropping more than one **.4do** file at a time, then each macro will be started.

Return to 4.29 Drag and Drop.

4.29.9 .project Files

If you drag a ***.project*** file (the ***12d Model*** project) onto an open ***12d Model*** project, the current open ***12d Model*** project is closed (it will ask whether you want to save) and the dragged ***.project*** is opened in ***12d Model***.

Return to 4.29 Drag and Drop.

4.29.10 .dwg,.dxf Files

If you drag a ***.dwg*** or ***.dxf*** file onto an open ***12d Model*** project, it opens up the read **Read DWG/ DXF Data** panel with the full path name of the ***.dwg/.dxf*** file entered into the **File** box.

This will work for files from folders, 12d Synergy, Skype and most other places that allow drag and drop.

If you are dragging and dropping more than one **DWG/DXF File** at a time, then a separate **Read DWG/DXF Data** panel is opened for each DWG/DXF file.

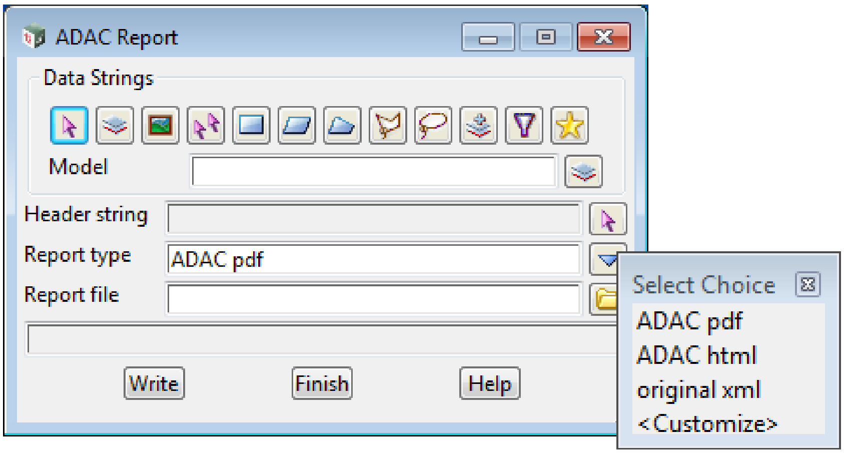
Return to 4.29 Drag and Drop.

4.30 Setting Up XML Reports

Many options now have ***XML reports*** and this allows users to create their own customised reports.

Once you have a customised report for a panel, it can be added to the list of reports given in the **Report type** field for that panel.

For example, the ADAC Report is shipped with 3 report types - ADAC.pdf, ADAC.html and original.xml - but a user may will to add their own customised report.



Hence in ***12d Model***, there is an XML file called **report_templates.xml** that defines which reports, 12d supplied or user customised, are available for a panel.

For each of the reports for a panel, the **report_templates.xml** file contains

(a) the name in the **Report type** pop-up list

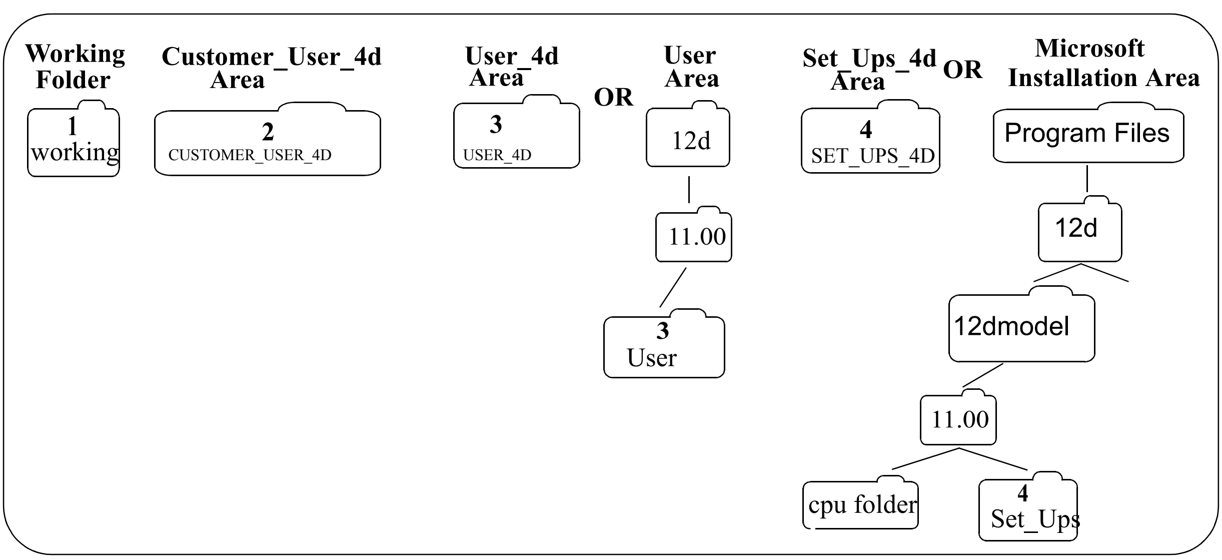
(b) the xslt that generates the report

(c) the file ending for the report file

(d) the number of decimal places in the report

(e) a comment about the report

The **report_templates.xml** are all **Set_Ups** files like *colours.4d files*, and folders are searched in a specific order to find **report_templates.xml** files.



HOWEVER, unlike *colours.4d*, it doesn't matter if the ***report_templates.xml*** file is found in a folder, the search **continues** and **all** the **folders** that **contain the file** are found.

And, again unlike *colours.4d*, the reports used are the **merger** of **all** the **found files**.

The search order is still important because if the **same name for a report for a panel** exists in more than one file in the found folders, the **first** occurrence in the files in the search order is the one that **is used**.

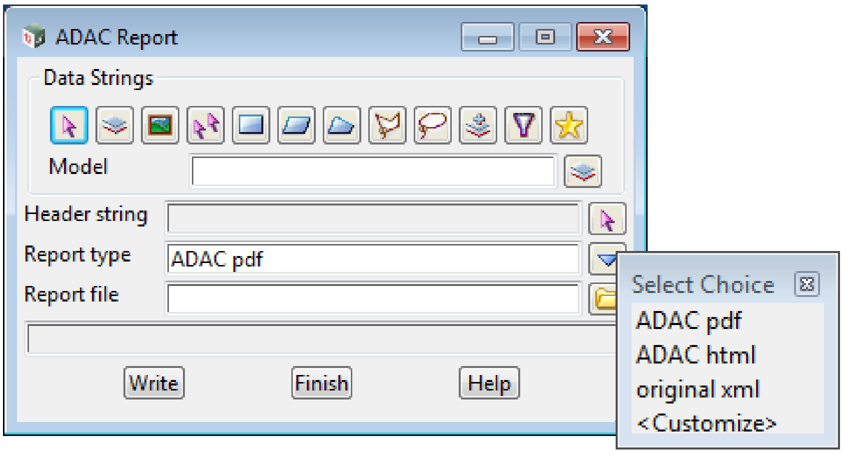
For example, if you had a report called **ADAC pdf** in the file *report_templates.xml* in your working folder, and one called **ADAC pdf** in the file *report_templates.xml* in Programs files\12d\12dmodle\11.0\set_ups, then the **ADAC pdf** in your **working folder** is the one that will be used and seen in the **Report type** pop-ups in panel **ADAC Report**.

To create and/or edit the **Report types** for a panel, see 4.30.1 Create/Edit Report Types

Go to the next section 4.30.1 Create/Edit Report Types or return to 4.30 Setting Up XML Reports.

4.30.1 Create/Edit Report Types

To create and/or edit the **Report types** for a panel, click on the choice icon for the **Report types** field and select **<Customise>**



The **Edit Report_templates.xml** panel is then brought up and it shows the standard areas for looking for the **report_templates.xml** files - Working folder, Customer (User), User, Set_Ups and Other - and displays whether the appropriate xml file:

(a) exists and is available for editing (**Edit**)

If the xml file exists in the folder and you have access to edit it, then the folder name is written in the panel and **[Edit]** is written on the right hand side of the line.

(b) does not exist but can be created (**Create**) by the editor

If the xml file does not exist in the folder and you have access to create it, then the folder name is written in the panel and **[Create]** is written on the right hand side of the line.

In this case a new file is created and written out to the given folder.

(c) is in a folder that has no access for editing and so can only be viewed (**View**)

If there is no access to the folder to create/edit the file, then the folder name is written in the panel and **[View]** is written on the right hand side of the line. Although the styles can’t be edited, they can be **copied**.

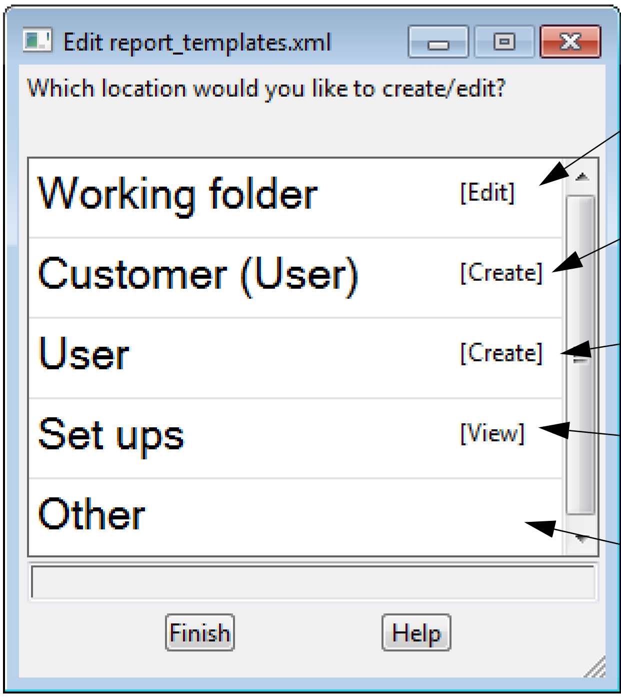
This usually applies to the **Set Ups** folder which normally requires Admin privileges for it to be written to.

(d) is in **Customer (User)**.

This only appears if a *Customer User* area has been defined.

(e) is defined separately by the user and then **Other** is displayed.

In this case, the full path name of the XML file is given by the user.



[Edit] - there is an xml file and it can be edited

[Create] - there is no xml file but one can be created. **Customer (User)** only appears if you have defined a Customer User area

[Create] - there is no xml file but one can be created.

[View] - no xml file can be created because there is no access, just viewed.

nothing means that the user has not defined any additional files to use

See

4.30.1.1 [Create] for Report Types
4.30.1.2 [Edit] for Report Types

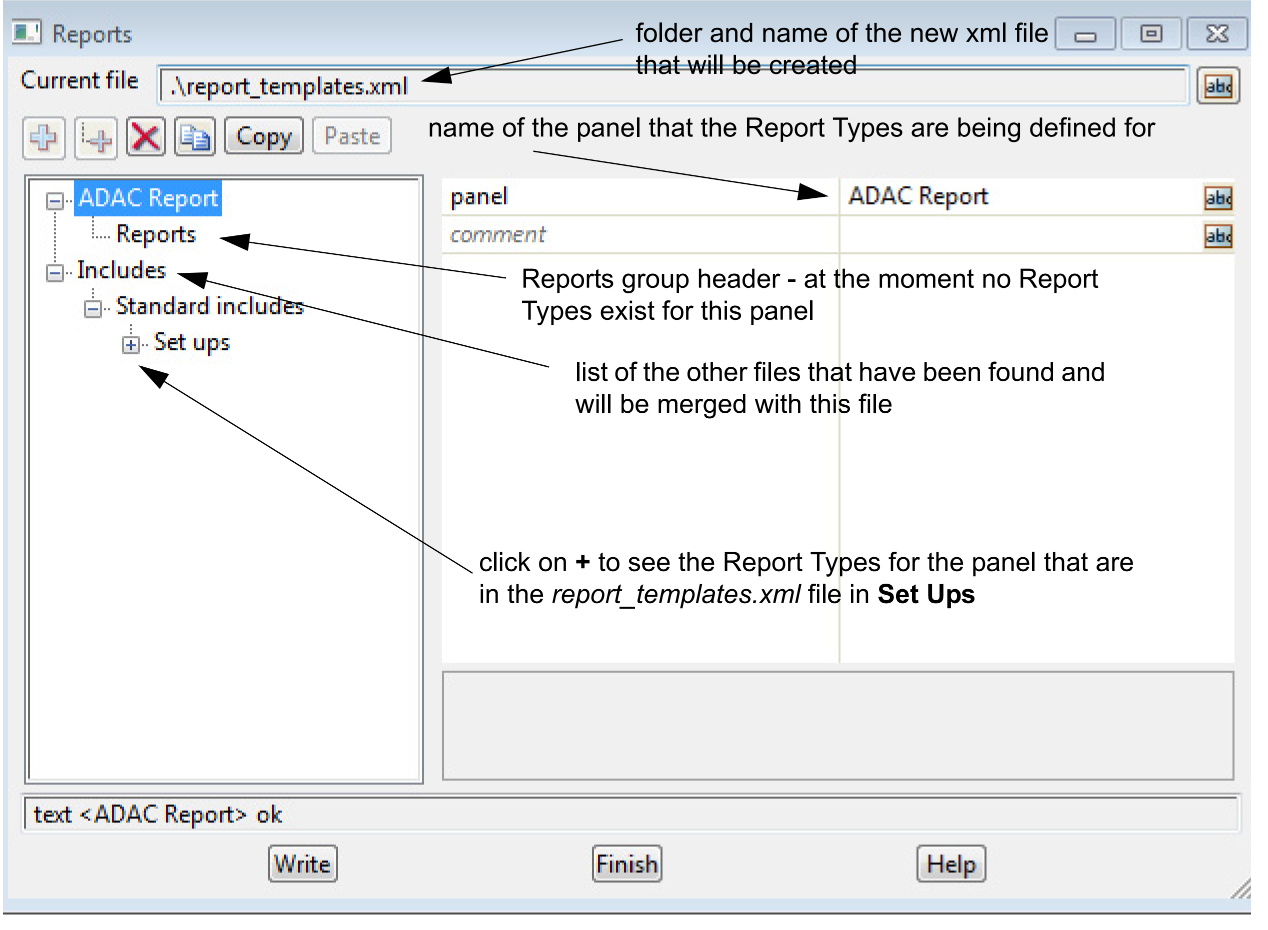
Go to the next section 4.30.1.1 [Create] for Report Types or return to 4.30 Setting Up XML Reports.

4.30.1.1 [Create] for Report Types

When **[Create]** is shown, a new xml file can be created in that folder.

Click on the line with **[Create]** and the **Editor** for a **report_templates.xml** starts up with a **Reports group header** but reports in it.

All the other **report_templates.xml** files that are found are listed in the **Includes** section displayed at the bottom of the tree.



The included files can not be edited whilst create/editing this file BUT you can copy items from them and paste them into items in this file.

New items for the **Report type** pop up can now be created (see 4.30.1.2.1 Creating New Report Types) and a new XML created by clicking on the **Write** button.

Creating and editing of *report_templates.xml* files is the same as for **[Edit]** and so is documented in that section. See 4.30.1.2 [Edit] for Report Types.

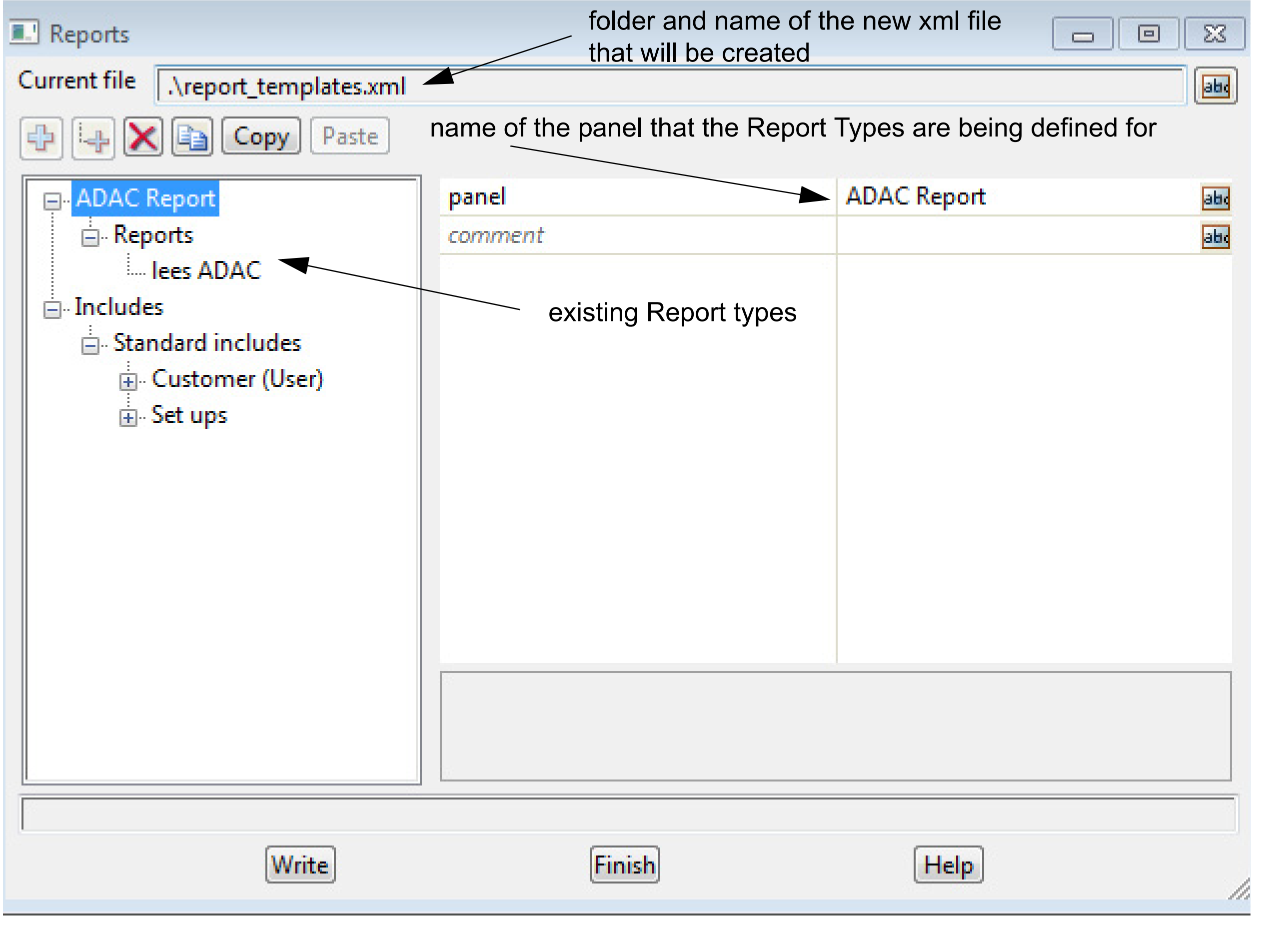
Write - writes out the information in the panel to the files name given in **Current file**.

Go to the next section 4.30.1.2 [Edit] for Report Types or return to 4.30 Setting Up XML Reports.

4.30.1.2 [Edit] for Report Types

When **[Edit]** is shown, an xml file already exists and can be edited.

Click on the line with **[Edit]** and the **Editor** for the *report_templates.xml* starts up and displays from the file, all the existing **Report types** for the panel.



After completing any editing, the XML is saved by clicking on the **Write** button.

As in the **[Create]** case, all the other appropriate files that are found are listed in the Includes section are displayed at the bottom of the tree (see 4.30.1.1 [Create] for Report Types).

The included files can not be edited whilst create/editing this file BUT you and copy items from them and paste them into items in this file.

For editing in the panel, see

4.30.1.2.1 Creating New Report Types

4.30.1.2.2 Copy and Paste

4.30.1.2.3 Deleting a Report Type

4.30.1.2.4 Duplicating a Report Type

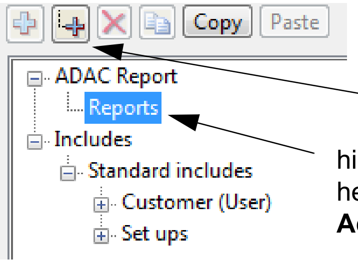
4.30.1.2.5 Editing a Report Type

Go to the next section 4.30.1.2.1 Creating New Report Types or return to 4.30 Setting Up XML Reports.

4.30.1.2.1 Creating New Report Types

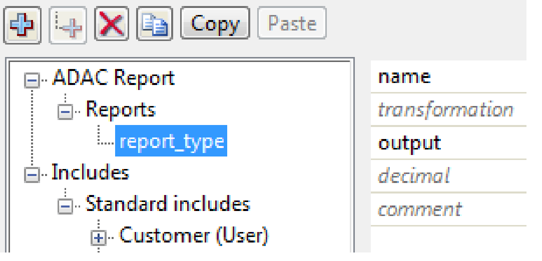
To create a **new Report type** from scratch when there are **no Report types**, highlight the **Reports group header** and click on the **Add Child** icon.

Note - the **Add Child** icon adds a node as a subnode of the highlighted node.

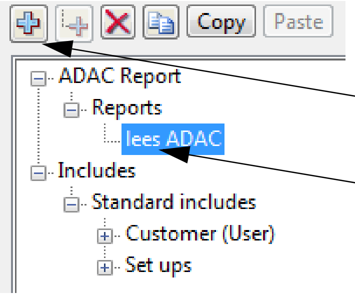


Add Child icon

highlight the Reports group header before clicking the **Add Child** icon



Too create a **new Report type** from scratch when **some Report types already exist**, you can highlight the **Report type name** and click on the **Add** (Add Sibling) icon.

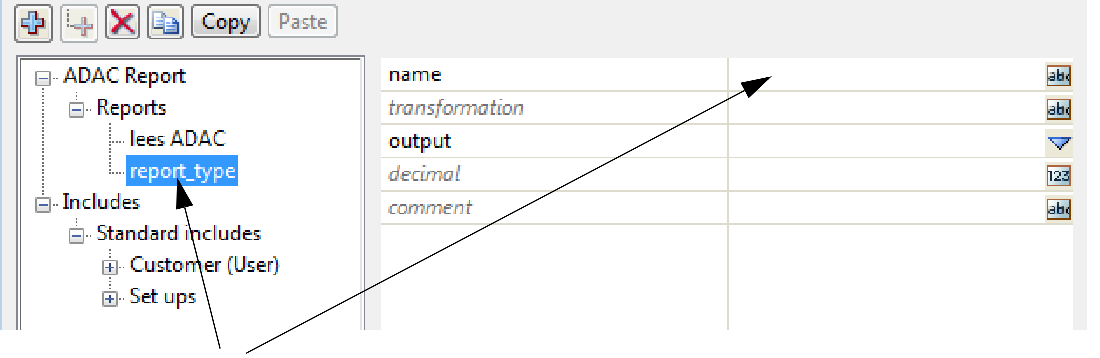


Add sibling icon

highlight the existing Report type name before clicking the **Add** sibling icon

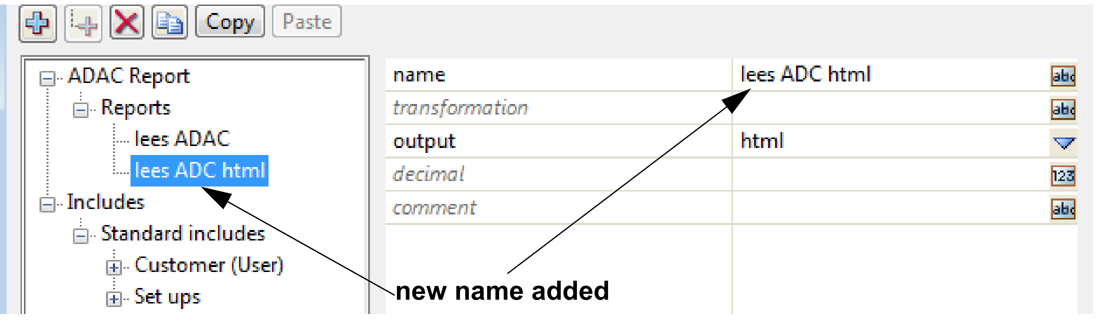
The **Add** icon adds a new item at **the same level** (a sibling) as the highlighted item and that is why a Report type must already exist so that one can be selected.

In both cases a new *Report type* is created with no name and so the general name **report_type** is written in the *Reports* list until the *Report type* is given a name.



new Report type added - no name as yet

Once the new *Report type* is given a name then that name will appear in the tree.



Once the new *Report type* is created then it can be edited by going into each field of the *Report type* and making changes.

Other Methods for Creating New Report Types

Duplicate can be used if there are already existing *Report types* for the panel in the file (4.30.1.2.4 Duplicating a Report Type).

Copy and **Paste** is particularly useful when there are **no** *Report types* for the panel in the file but there are *Report types* for the panel in other files listed in the **Includes** node. Although these *Report types* can't be edited, they can be copied (4.30.1.2.2 Copy and Paste).

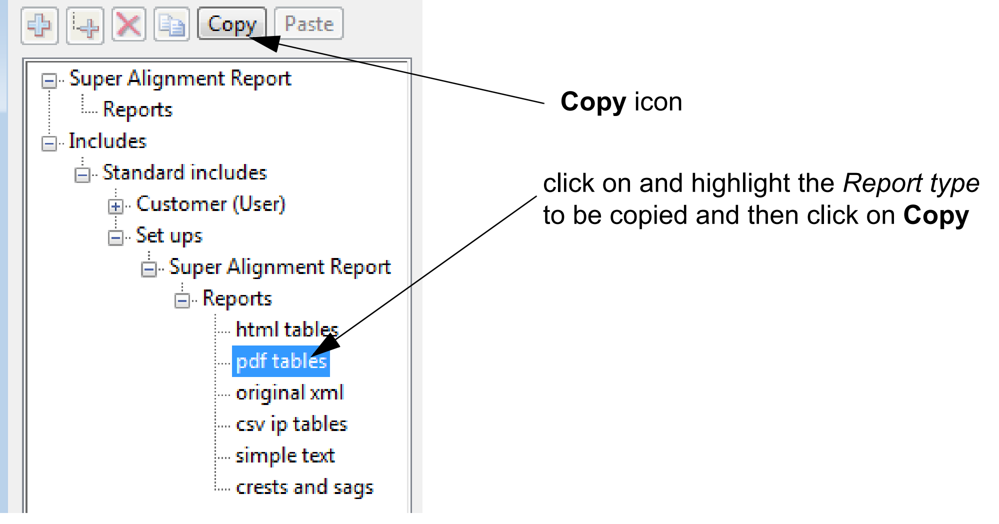
Go to the next section 4.30.1.2.2 Copy and Paste or return to 4.30 Setting Up XML Reports.

4.30.1.2.2 Copy and Paste

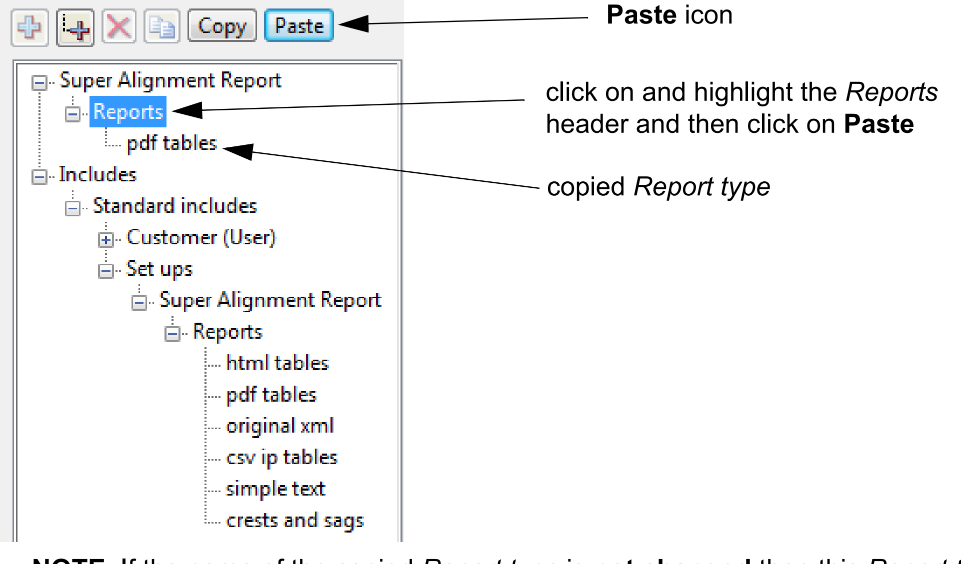
Copy and **Paste** can be used to create a copy of an existing *Report type*, and then the copied *Report type edited*.

Although there may be no *Report types* in the folder you are creating *Report types* for, there will usually be some *Report types* in the XML file in **Set Ups** that can be copied.

In the tree in the Editor, click on the **+** in front of **Set Ups** to expand the tree and then click on the item in the tree that you want to copy. Then click on the **Copy** icon.



Next click on and highlight the *Reports* header and click on **Paste**. The *Report type* will then appear under the *Reports* header with the same name as the original *Report type*. The copied *Report type* can then be edited.

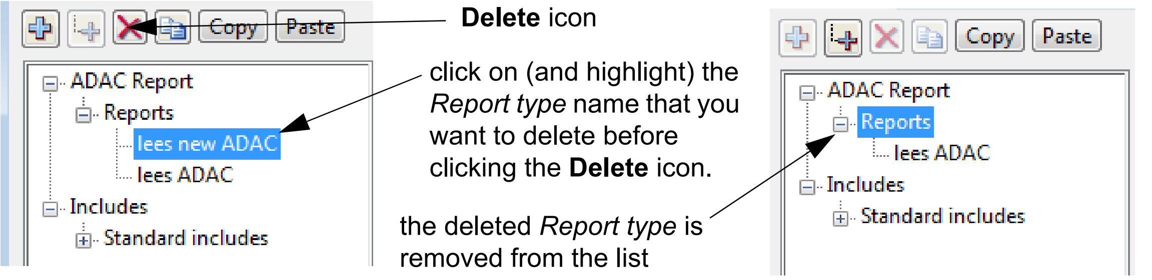


NOTE: If the name of the copied *Report type* is **not changed** then this *Report type* will be used instead of the one in **Set Ups**.

Go to the next section 4.30.1.2.3 Deleting a Report Type or return to 4.30 Setting Up XML Reports.

4.30.1.2.3 Deleting a Report Type

To **delete** a *Report type*, simply click on and highlight the *Report type* **name,** and then click on the **Delete** icon.

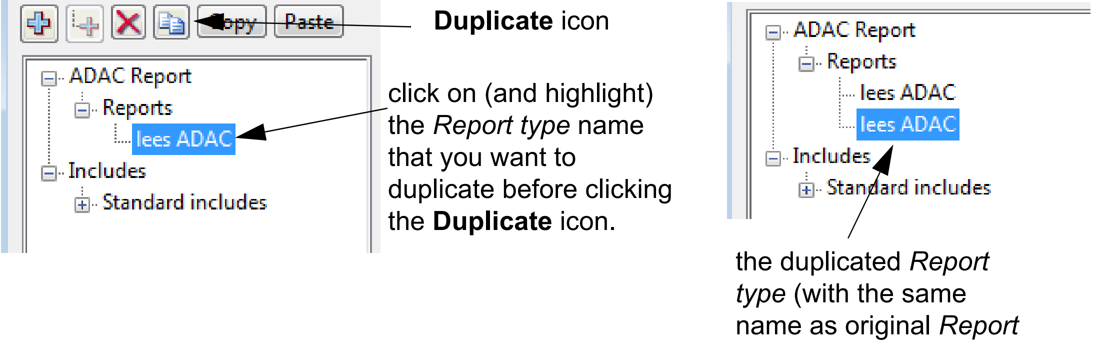


Go to the next section 4.30.1.2.4 Duplicating a Report Type or return to 4.30 Setting Up XML Reports.

4.30.1.2.4 Duplicating a Report Type

To **duplicate** a *Report type*, simply click on and highlight the *Report type* **name,** and then click on the **Duplicate** icon.

A copy of the *Report type*, with exactly the same name, is created and added to the end of the list of *Report type*.

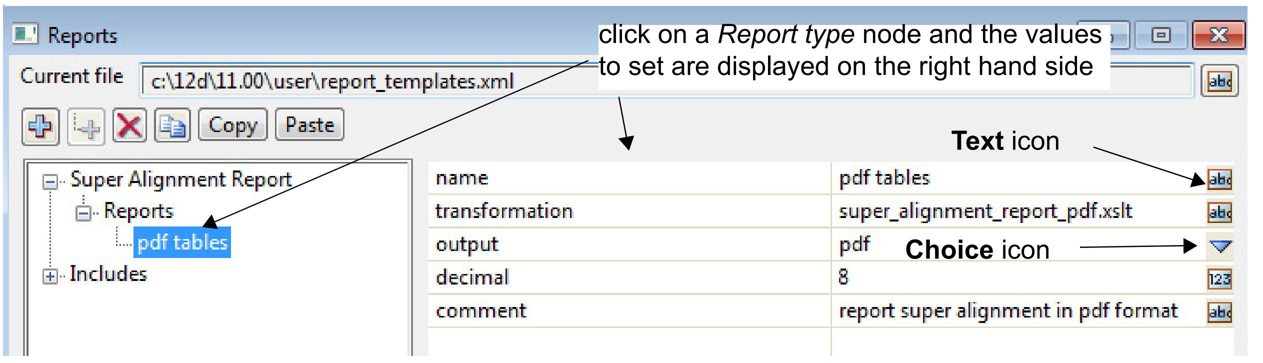


The name of the *Report type* needs to be changed, and any other edits done that are required.

Go to the next section 4.30.1.2.5 Editing a Report Type or return to 4.30 Setting Up XML Reports.

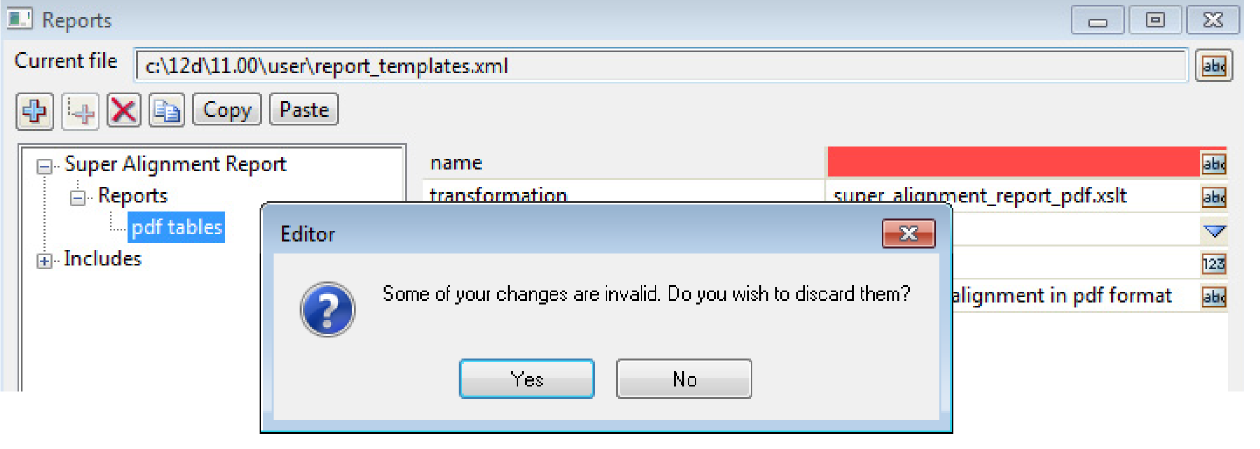
4.30.1.2.5 Editing a Report Type

To edit a **Report type**, click on the *Report type* name and all the values for that *Report type* will be displayed on the right hand side of the panel.



The panel fields on the right hand side of the panel have the same icons as in other ***12d Model*** panels and the standard pop ups are all available. For example, text, number and choice icons.

When clicking to go to another *Report type,* all the fields currently on the right hand side are validated and if there are any problems, the first line with an error is coloured red and an **Editor error** box is displayed.



If **Yes** is selected, then all the **invalid field** values are **discarded**, the **Editor error** panel removed, and the data for the new selected node displayed on the right hand side of the panel.

If **No** is selected, then the right hand side of the panel does not change and can undergo further edits.

Go back to the section 4.30.1.2 [Edit] for Report Types or return to 4.30 Setting Up XML Reports.

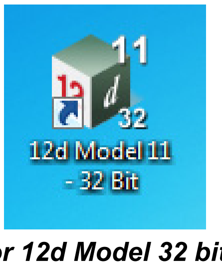
5 Starting Up

This chapter contains information about how to start ***12d Model***, organising ***12d Model*** error logging and how to run chains and macros when starting up ***12d Model***.

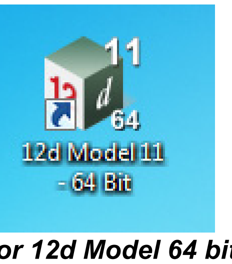
Go to
5.1 Front Panel - Open a Recent Project
5.4 Options for the Highlighted Project
5.5 Organizing Working Areas
5.6 Project Shortcuts
5.7 Creating Project Shortcuts by Hand
5.8 Environment Variables Shortcut
5.9 Running Macros and Chains on Start Up
5.10 Error Logging File

5.1 Front Panel - Open a Recent Project

When ***12d Model*** is installed from the 12d Installation CD, the 12d icon ***12d Model*** **11** is created.



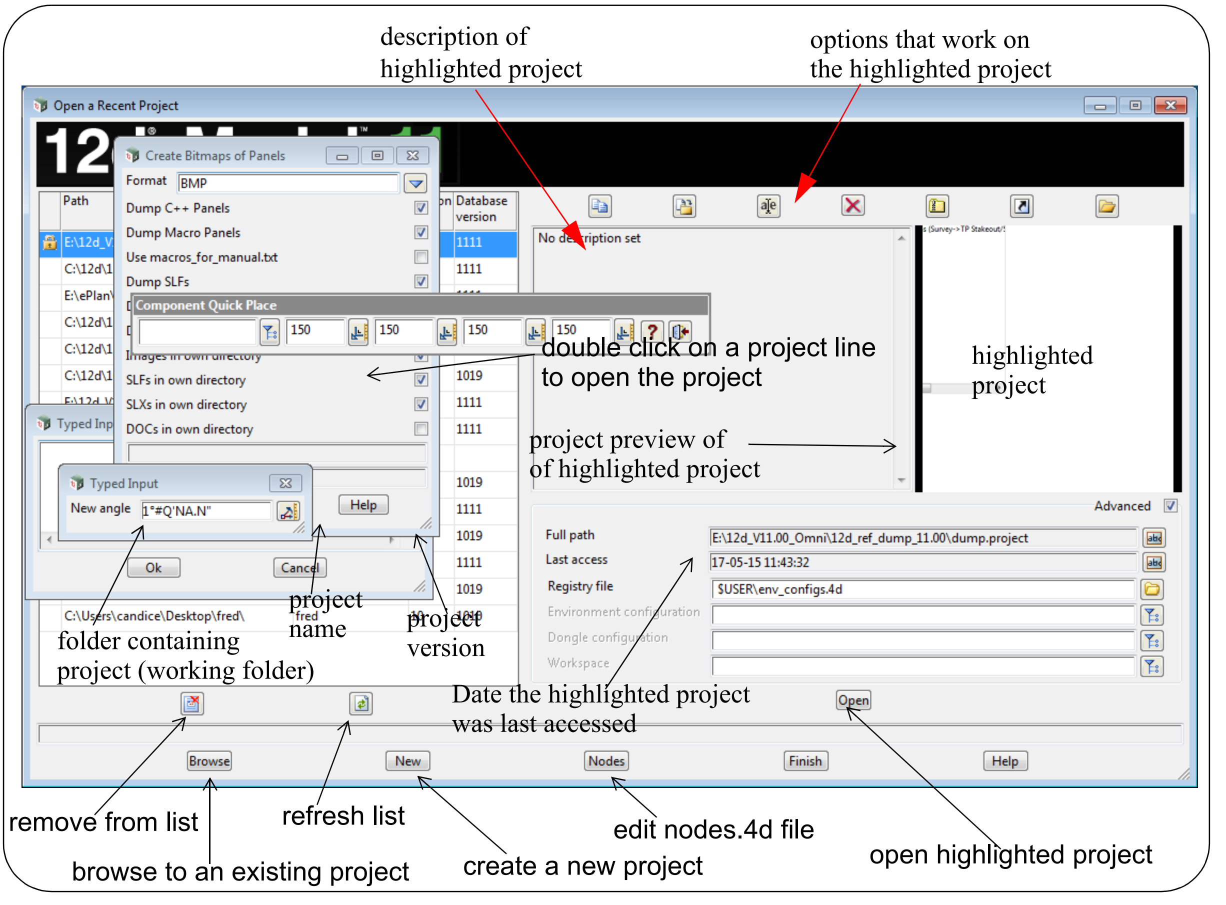
icon for 12d Model 32 bit exe



icon for 12d Model 64 bit exe

The ***12d Model*** icon is a **shortcut** to start ***12d Model*** and then attach to the folder *C:\12d\11.00*.

Clicking the 12d Model icon starts 12d Model. First the 12d Model splash screen will appear and then the initial ***12d Model*** screen with the **Open a Recent Project** panel.



If this does not work, please contact your ***12d Model*** administrator to check the procedure to be used on your system. For a detailed description on how to install ***12d Model***, please refer to the *Installing* ***12d Model*** chapter in the *Getting Started* manuals.

To select a project is the ***Recent projects list***, simply double click on the entry in the list and ***12d Model*** will open the project up.

The description of the functions of the fields and buttons in the **Open a Recent Project** panel are:

Field Description	Type	Defaults	Pop-Up
Recent projects list	output	recent projects	

names and folders of project recently select. Single clicking on an entry in the list displays the projects description in the Project description area. Double clicking on an entry in the list changes to this project

Project description output

description of the project highlighted in the Recent projects list.

Advanced tick box

if tick, environment configurations can be assigned to the project and extra panel fields are displayed to allow for this. For more information, please go to the section 7.2.1 Project Open/Create Panel with Advanced Ticked.
If not tick, no extra panel fields are displayed.

Full path output

name of the full path to the project.

Last accessed output

date that the project was last accessed.

The icon **Remove from list** - removes the project from the *recent projects* list

The icon **Refresh list** - refreshes the *recent projects list*

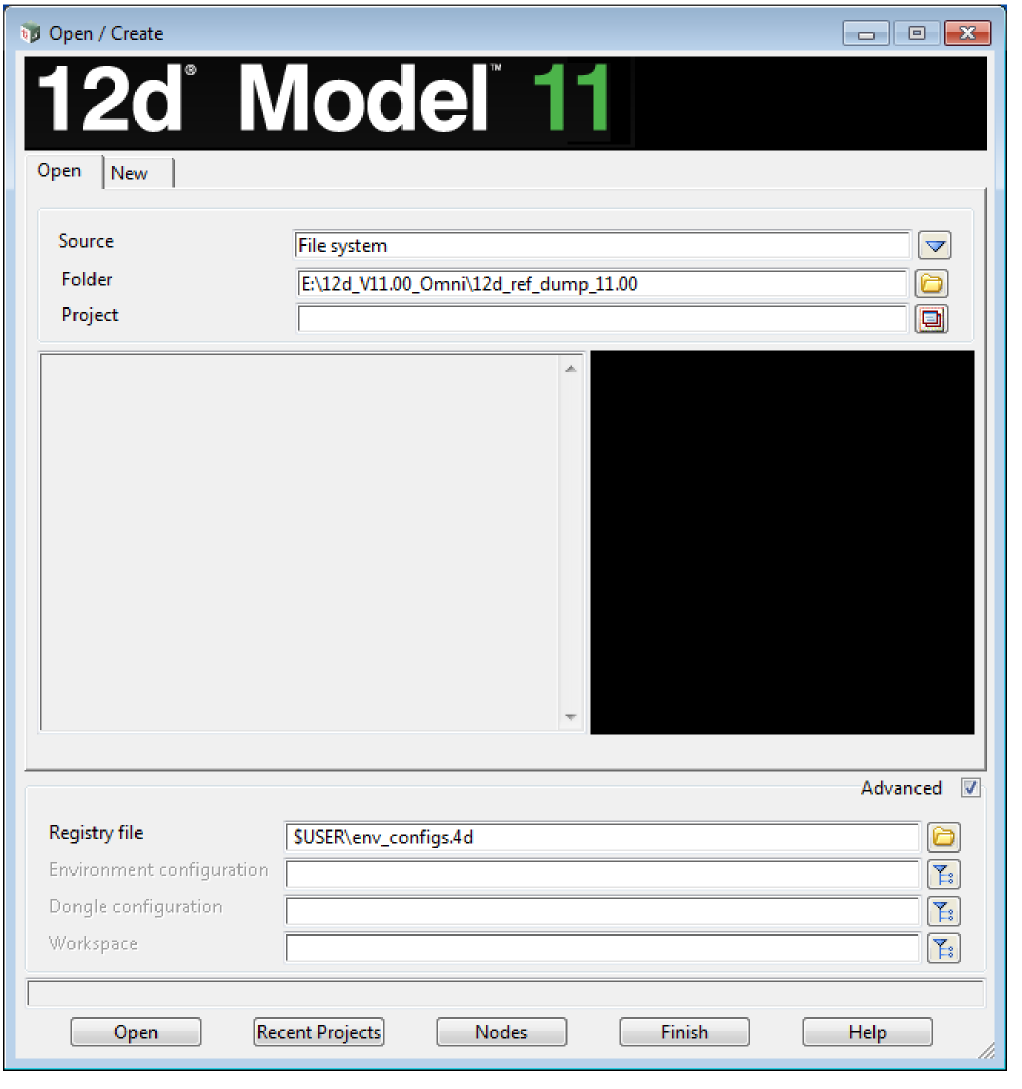
For the icons that work on the highlighted project, go to 5.4 Options for the Highlighted Project

For the **Browse** button, go to 5.2 Browse Button and Open Tab
For the **New** button, go to 5.3 New Button and New Tab
For the **Nodes** button, go to 7.6.5.6 Nodes.4d Editor

5.2 Browse Button and Open Tab

Clicking on the **Browse** button brings up the **Open** tab that allows you to browse and open an **existing** project. This is used when opening a project that is not in the *Recent Projects* list.

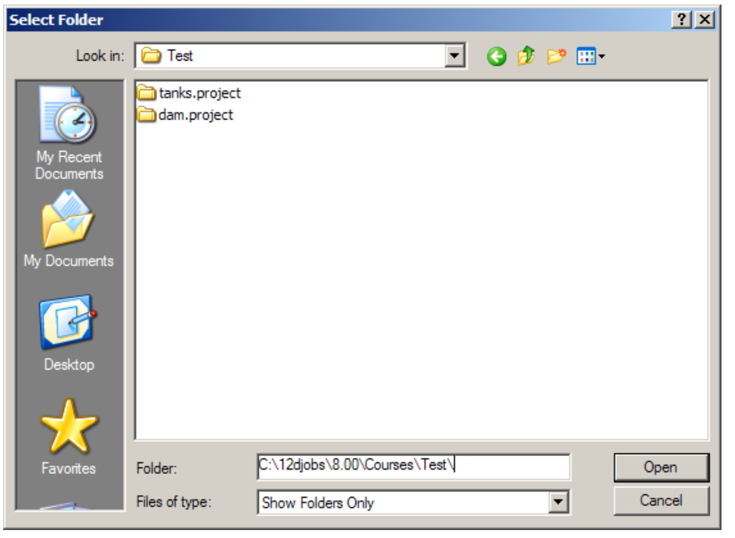
The **Open** button opens the project that has been browsed to.



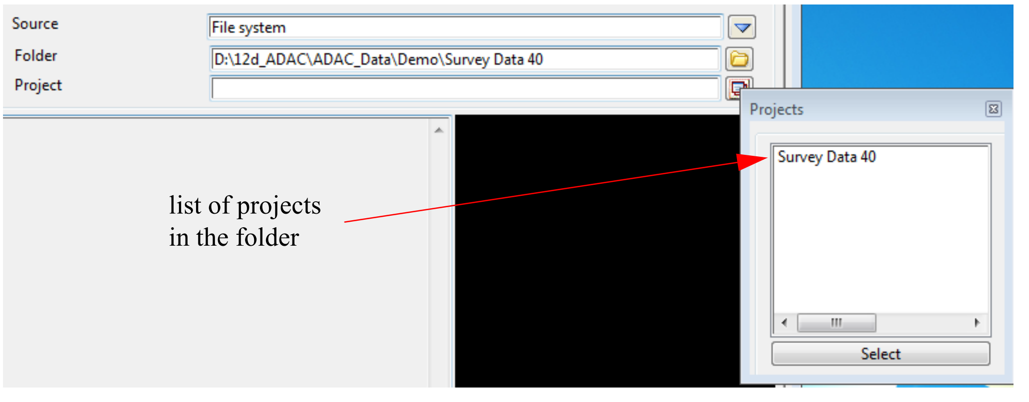
Folder input current folder Microsoft browser

name of the folder to look for the project. This will become the working folder.

If the displayed folder does not contain the required project, clicking LB on the folder icon on the right side of the field will bring up the Microsoft browser which is used to navigate to the folder containing the required project.



If the selected folder contains ***12d Model*** *projects, then a pop-up with a list of projects in the folder is displayed in the* Project name *field and the required project can be selected from the list.*



Project input none projects in folder

name of the project to become the working project.

Open button

after selecting this button, the working folder is changed to the folder specified in the folder *field and the selected project opened. If the project does not exist, and error message appears.*

Recent Projects button

takes you back to the original front screen. See 5.1 Front Panel - Open a Recent Project.

Quit button

exit ***12d Model****.*

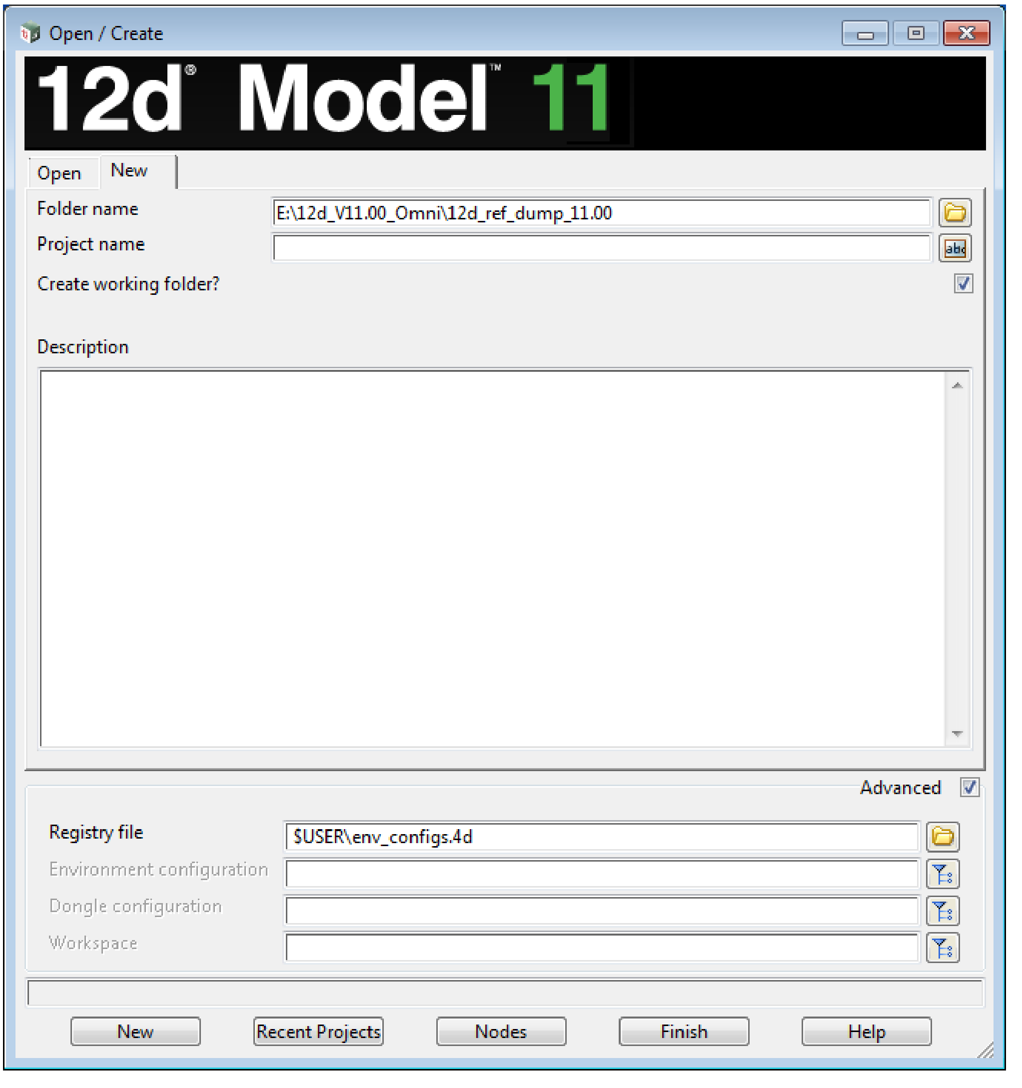
Note

The **New** tab takes you to the tab for creating a new project. See 5.3 New Button and New Tab.

5.3 New Button and New Tab

Clicking on the **New** button brings up the **New** tab that allows you to create a **new** project.

The **New** button creates the new project.



To create a ***new*** project, fill in the Folder name and Project name and click on the **New** button.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Folder	folder box	current folder	select folder panel

if **Create working folder** *is ticked, a folder with the same name as the project is created in the folder given in this panel field, and the project will be created inside this folder with the same name as the project. This is to encourage the habit of creating a folder of the same name of the project and then creating the project inside the folder.*

If **Create working folder** *is not ticked, the project is created in the folder given in this panel field.*

Project name	typed input		

name of the new project to be created.

Create working folder tick box

if tick, a folder of the same name as the project is created in the folder **Folder***, and the project is created inside this folder. The folder with the same name as the project is then the working folder.*

If not tick, the project is created in the folder **Folder** *and this folder is the working folder.*

Description typed input

a description of the project can be typed into this field. This is stored with the project.

Advanced tick box

if tick, environment configurations can be assigned to the project and extra panel fields are displayed to allow for this. For more information, please go to the section 7.2.1 Project Open/ Create Panel with Advanced Ticked*.*
If not tick, no extra panel fields are displayed.

New button

after selecting this button, the new project created in either the **Folder** *or if* **Create working folder** *is ticked, inside a folder created in the* **Folder** *of the same name as the project.*

12d Model *then opens the newly created project.*

Recent Projects button

takes you back to the original front screen (5.1 Front Panel - Open a Recent Project).

Quit button

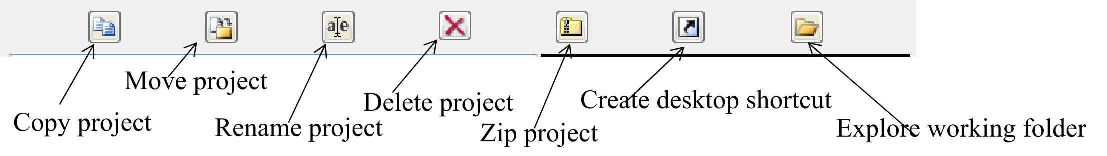
exit out of ***12d Model.***

Note

The **Open** tab takes you to the tab for browsing to open an existing project (5.2 Browse Button and Open Tab).

5.4 Options for the Highlighted Project

When a project is highlighted in the list of **recent projects**, icons for options are activated on the right hand side of the **Project Selection** panel.



Copy project - copy the project and the working folder

Move project - move the project and working folder to a new folder

Rename project - rename the project and the working folder

Delete project - delete the project and the working folder

Zip project - zip up the project and the working folder

Create desktop shortcut - create a shortcut on the desktop that starts ***12d Model*** and opens the project - see 5.6 Project Shortcuts

Explore working folder - open Windows Explorer and show the contents of the working folder

These options are documented in the section 7.2.4 Options Active When Recent Project Highlighted.

Go to the next section 5.5 Organizing Working Areas.

5.5 Organizing Working Areas

For each project,***12d Model*** creates a unique sub-folder of the working folder with the name consisting of the project_name followed by ***.project***.For example, for the project ***Olympic***, and folder called ***Olympic.project*** is created.

This sub-folder (Olympic.project) is called the **project area**. All the internal ***12d Model*** information for that project is kept inside the project area.

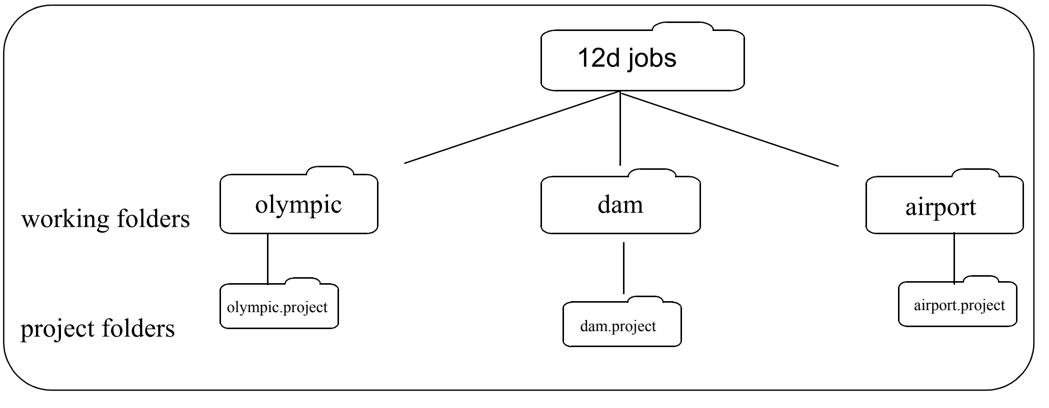
All outputs, reports and plots are written to the working folder (the folder containing the project) and are **not** held inside the project area.

Hence to get a complete backup of the project and all associated information, it is best to ***backup the entire working folder***.

The project name, which can be up to 256 alphanumeric characters and can include spaces, must be unique within the working folder but other folders may include ***12d Model*** projects with the same name - these projects are distinct and are not related in any way.

There is no limit to the number of projects in a particular working folder but because all the outputs, reports and plots for each project in the same working folder would be mixed in together, ***it is recommended to have each project in its own working folder***. That way the inputs files, output files, plots, reports etc. from the separate projects do not end up in the same working folder. Each separate working folder can then be easily backed up.

As an example, if three unrelated projects - olympics, airport, and dam - are to be created in a folder called **12d jobs**, it is suggested that the project ***olympic*** is created in a **sub folders** of **12d jobs** called ***olympic***, ***dam*** is created in a **sub folders** of **12d jobs** called ***dam*** and ***airport*** is created in a **sub folders** of **12d jobs** called ***airport***.



To help users adhere to this convention, when creating new projects, there is a Create working folder tick box which if ticked, will first create a folder of the same name as the project being created and then create the project in that folder.

Go to the next section 5.6 Project Shortcuts.

5.6 Project Shortcuts

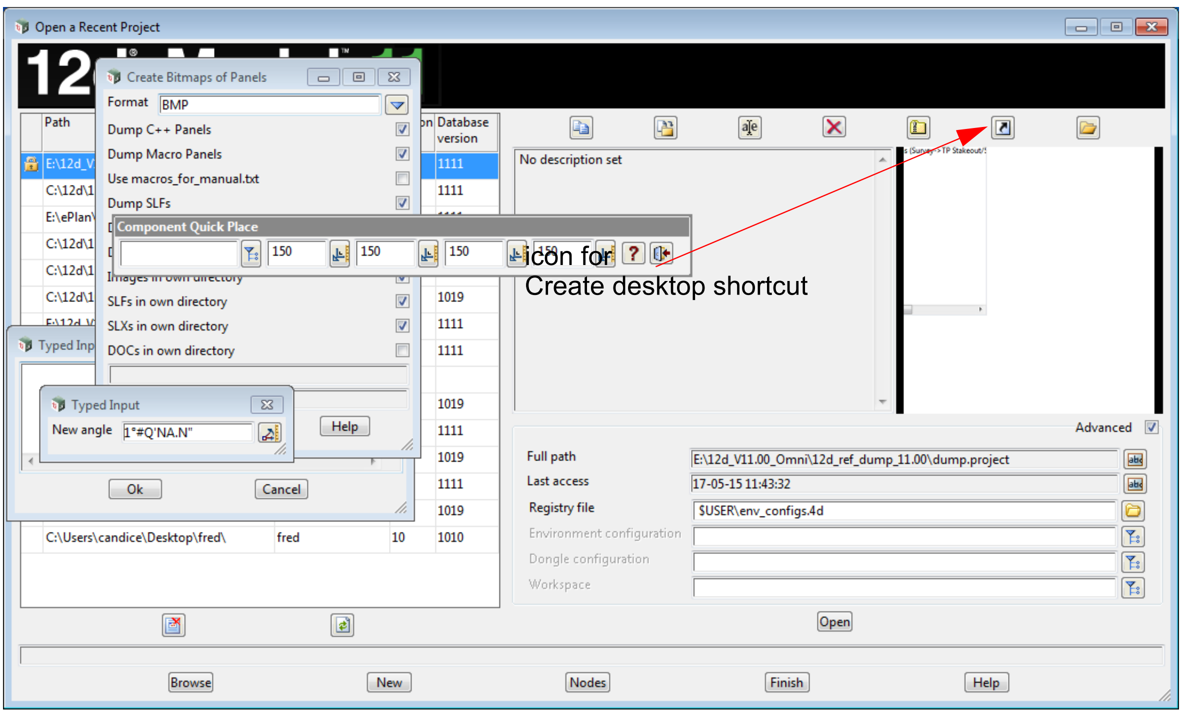
The list of most recent projects makes it very easy for 12d operators to get into their projects.

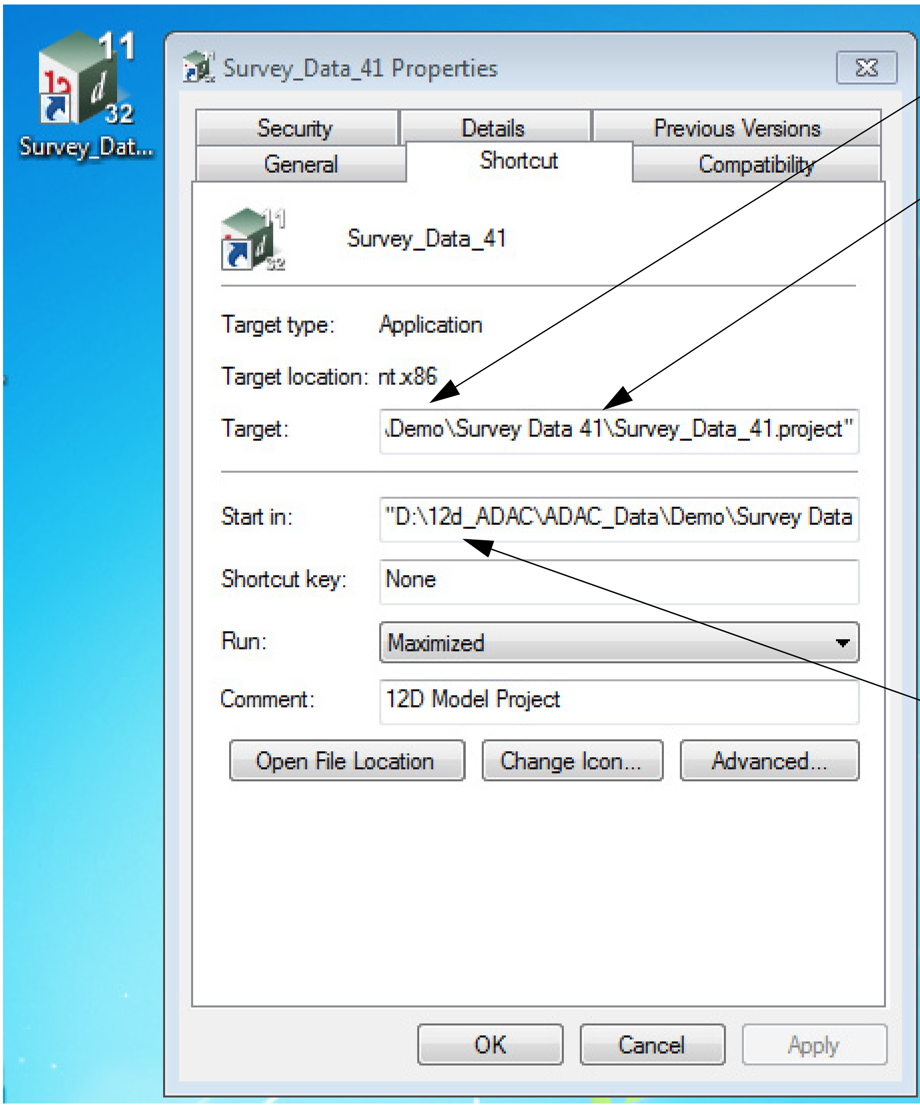
However if there is a folder of projects, or even a particular project, that a user wants to get into by simply double clicking on an Icon on the screen, then a Windows shortcut can be used to do it.

The easiest way to create the shortcut is to first make sure that the project is in the recent projects list. To do this, just open up the project and then exit ***12d Model***.

Then start up ***12d Model*** again by clicking on the ***12d*** icon on the desktop, highlight the project in the ***recent projects list*** and select the icon for Create desktop shortcut on the icons on the right hand side of the project list.

This will create a shortcut on the desktop with the name of the project, and the shortcut is set to open up the project when the icon is double clicked.





Target: has the path to **12d.exe**

This is optional and can be the full path name to an existing project (including the project name) and this is opened when the icon is clicked

OR

just an existing project and this must be in the folder given in **Start in:** and it is opened

Start in: is used as the working folder when no full project path is given in **Target:**

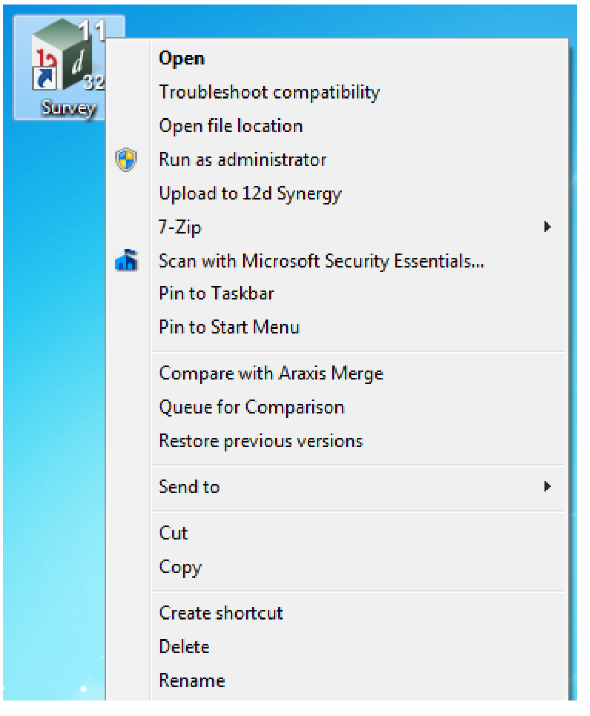
Go to the next section 5.7 Creating Project Shortcuts by Hand.

5.7 Creating Project Shortcuts by Hand

Creating an icon to open up in an existing ***12d Model*** project can also be done by hand by copying a ***12d*** icon that is already on the screen, renaming the copied icon to the name of the project (say *survey).*

The Properties for the icon *survey* is then modified by changing what is in the **Target:** box and maybe the **Start in:** box.

To display and modify the properties of a desktop icon, click RB over the ***12d*** icon and select *Properties* from the menu. Then click on the *Shortcut* tab



The Properties for the icon *survey* can now be modified to:

(a) open an existing project

If an icon is to open a given project, then the full path to the projects (including the project folder name) in given in the **Target:** field after the path to the 12d exe.

For example, if the ***12d Model*** project *survey* was in the folder "12d jobs\survey ", set the **Target:** for the icon ***survey*** to:

"C:\Program Files\12d\12dmodel\11.00\nt.x86\12d.exe" "C:\12d jobs\survey\wurvey.project"

The icon *survey* would then automatically open up the project

C:\12d jobs\survey\survey.project

Note that if the path name for the project in **Target:** contain spaces, then it must be enclosed in double quotes (**"**).

If the *survey* icon was double clicked on, it would now open the project *survey.*

The **Start in:** is ignored.

(b) open an existing project

Another way that an existing project can be opened is that if a project such as *survey* already exists in the folder given by the **Start in:** field, then in the **Target:** field, the project name is added after the path to 12d. exe.

For example, the **Target**:

"C:\Program Files\12d\12dmodel\11.00\nt.x86\12d.exe" survey

would automatically open the project **survey** in the **Start in:** folder of the shortcut.

So **Target:** and **Start in:** are used to obtain the existing project to open whereas in (a), only Target: is used.

Note that if either the path names for the **Target:** or **Start in:**, or the project name contain spaces, then they must be enclosed in double quotes (**"**).

Note

The **Target:** field is actually a command line to start up ***12d Model***. For more information see 39.8 Arguments When Starting 12d Model in Appendix 39 Setting Up & Configuring 12d.

Please continue to the next section 5.8 Environment Variables Shortcut.

5.8 Environment Variables Shortcut

If a file of ***12d Model*** environment variables has been set up by the user (see section 39.4 Environment Variables in Appendix 39 Setting Up & Configuring 12d), then instead of setting the environment variable ENVIRONMENT_4D to point to the file or setting it up with the default name, env.4d, the environment file can be passed to ***12d Model*** using the **Target** of the icon properties.

For example, the **Target**:

"C:\Program Files\12d\12dmodel\11.00\nt.x64\12d.exe" -env F:\12d\env.4d

would fire up the 64-bit ***12d.exe*** using the file of environment variables called F:\12d\env.4d

The **Target**:

"C:\Program Files\12d\12dmodel\11.00\nt.x64\12d.exe" -env F:\12d\env.4d airport

would fire up the 64-bit ***12d.exe*** using the file of environment variables called F:\12d\env.4d **and** also automatically open the project **airport** in the **Start in:** folder of the shortcut.

Again if any of the path names contain spaces, then they must be enclosed in double quotes (**"**).

The **Target:** field is actually a command line to start up ***12d Model***. For more information see 39.8 Arguments When Starting 12d Model in Appendix 39 Setting Up & Configuring 12d.

Please continue to the next section 5.9 Running Macros and Chains on Start Up.

5.9 Running Macros and Chains on Start Up

To allow for tailoring *12d Model* when a new project is created or an existing project opened, *12d Model* runs user supplied files of macros and/or chains.

For ***new*** projects, the default name of the file is

macros.4d

This can be changed to a different file by setting the environment variable

RUN_MACROS_FILE_4D *file_of_macros_to_run_for_new_projects*

For ***existing*** projects, the default name of the file is

project_macros.4d

This can be changed to a different file by setting the environment variable

RUN_PROJECT_MACROS_FILE_4D *file_of_macros_to_run_for_existing_projects*

(the environment variables can be set on the ***Extra A*** tab of the env.4d editor - see 7.6.3 env.4d)

The files consist of macros and/or chains, one per line, where

for macros, just the ***name of the macro*** is needed on the line

and for chains, the command **run_chain** *name_of_the_chain* is needed on the line.

The macros and chains are run in the order that they occur in the file.

Please continue to the next section 5.10 Error Logging File.

5.10 Error Logging File

When ***12d Model*** starts up, it tries to create an **error logging file**, called

log?????.4de

where ????? is a hashed number using your login name, process id & the current time.

When 12d Model terminates, the error log file is deleted if no errors were logged.

The **folder** that the error log file is created in is given by the environment variable LOG_DIR_4D

If LOG_DIR_4D is not used,***12d Model*** tries to create the log file in the current folder, the HOME folder, the TMP folder and the TEMP folder.

If creating a **error log file** fails in all these areas, ***12d Model*** will not start up. This should never happen.

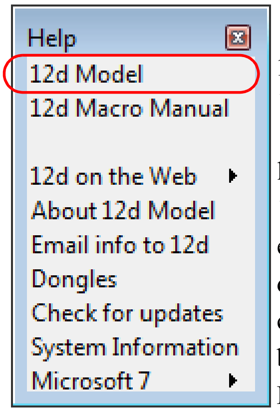
To return to the beginning of this chapter, click on 5 Starting Up.

6 12d Model Help

Position of option on menu: **Help =>12d Model**

All the information in the ***12d Model*** **Reference** manual is also available as electronic **Help** accessed from within ***12d Model*** (also know as the ***12d Model*** **Context Sensitive Help)**.

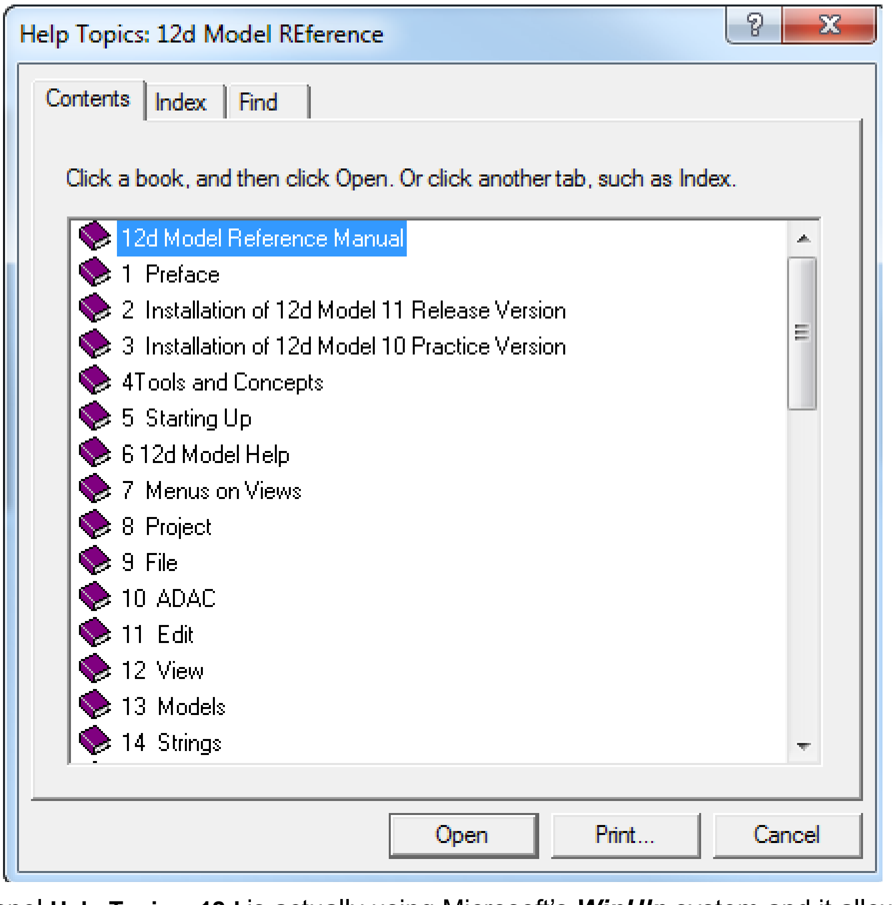
The entire ***12d Model Help*** manual can be accessed by selecting ***12d Model*** on the **Help** menu item on the main ***12d Model*** menu.



12d Model help
12d macro programming language help

links to web site www.12d.com.
12d Model modules authorized, dongle number
email details of your 12d Model to 12d Solutions
dongle testing panel
check for newer versions of 12d Model
brings up Microsoft's System Information panel
For Windows 7 links to the WinHlp32.exe

Clicking on ***12d Model*** brings up **Help Topics: 12d Model Reference**



The panel **Help Topics: 12d** is actually using Microsoft's ***WinHlp*** system and it allows you to look at the overall structure of the ***12d Model*** *Help* and access any part of it. More information on using the tabs **Contents**, **Index** and **Find** will follow in the next section.

Alternatively, individual topics for a panel or menu can be invoked by pressing the F1 key whenever the focus is on the menu or panel, or by clicking on the *Help* button on any ***12d Model*** panel (see F1 Key). This is the *context sensitive* nature of the ***12d Model*** Help.

For some options, there is also additional help files and videos. This is denoted by a * after **Help** on the **Help** button. That is **Help*** (see Extra Help).

It is also possible to have a **Help** button and **F1** key available for ***12d Model*** PLs (macro) programs written by 12d Solutions or by Users. Please see the ***12d Model*** *Programming Language* manual for more information on this feature.

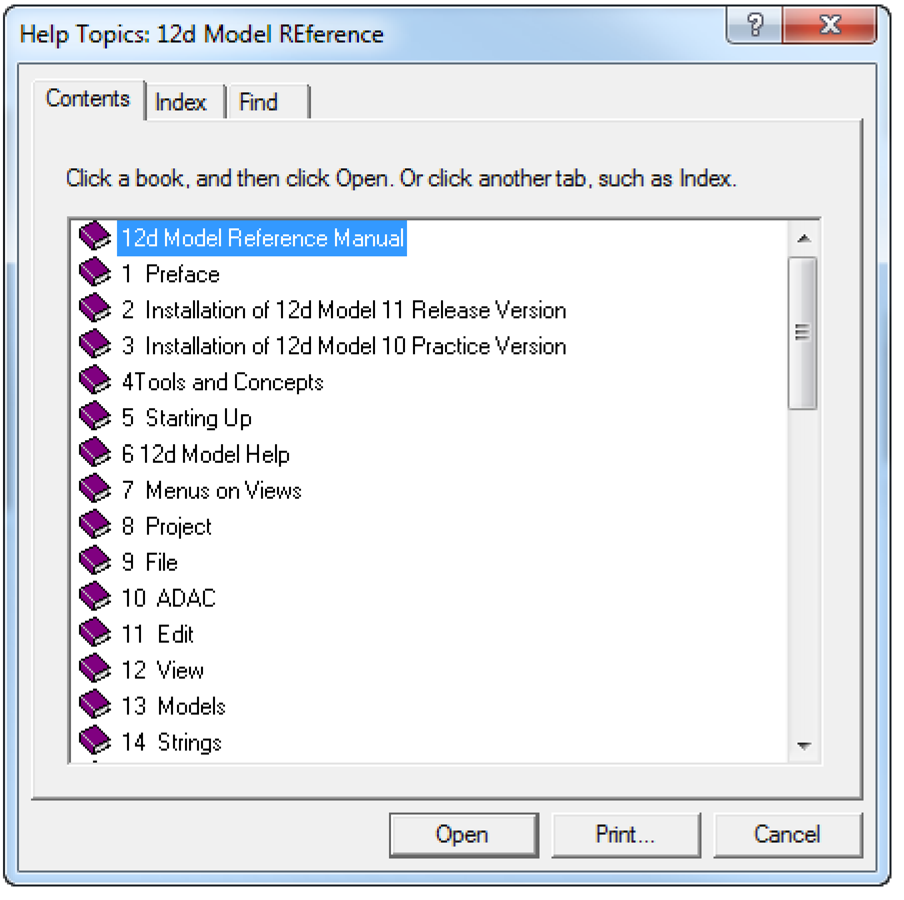
Note: The ***12d Model*** *Reference* manual is available in pdf on the ***12d Model*** installation DVD, or on the 12d web site www.12d.com.

More information on the **Help** system is given in the sections:

Contents
Index
Find
Panel Help Button
F1 Key
Navigating in Help
Extra Help

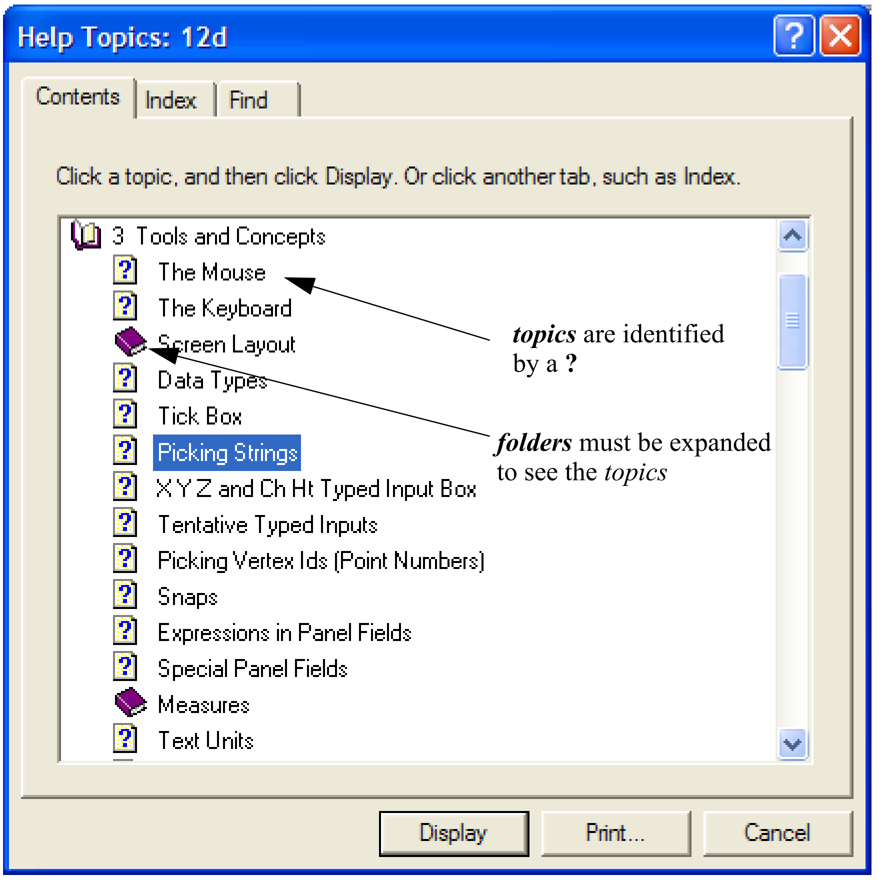
Contents

The **Contents** tab allows you to look at the overall structure of the ***12d Model*** *Help* and access any part of it.



Warning - only *topics* in the *Contents* can be viewed in *Help* so any folders in *Contents* folders must be expanded until *topics* are displayed. *Topics* can be easily identified because they have a question mark beside them indicating that *Help* is available and can be viewed.

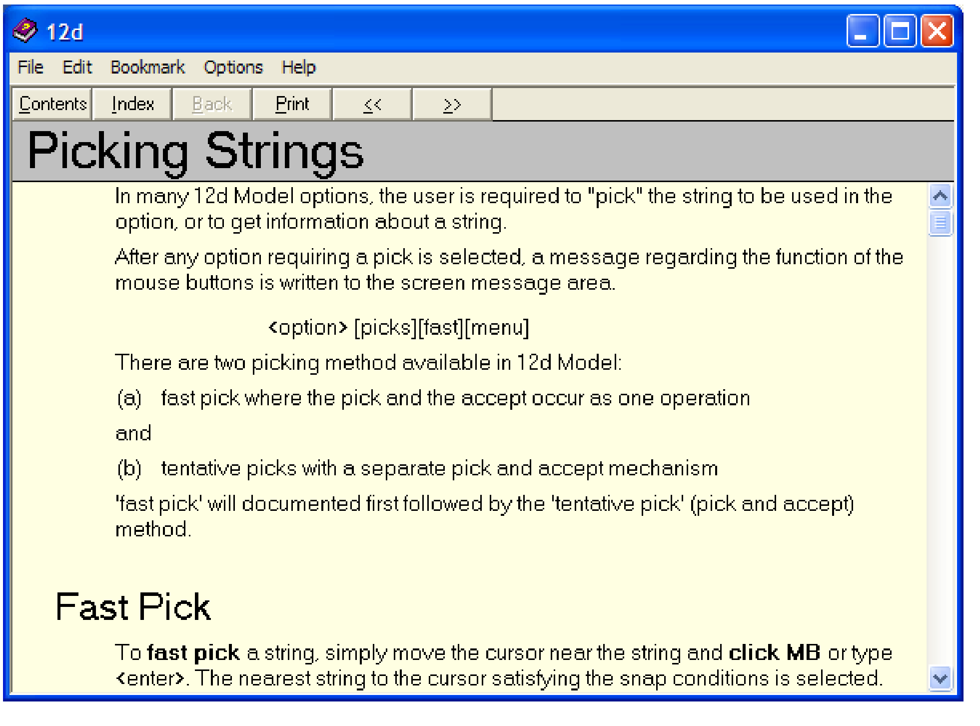
For example, double clicking on ***Tools and Concepts*** expands the next level of ***Tools and Contents***.



and topics are *The Mouse*, *The Keyboard* etc.

Double clicking on the topic ***Picking Strings*** will then display the topic.

The ***Contents*** then disappear leaving *Help* open at the selected topic.



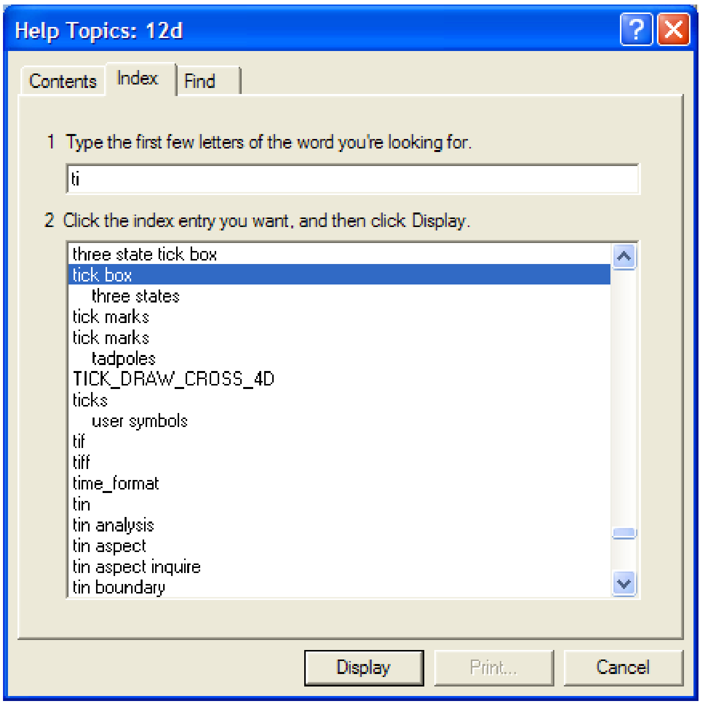
Double clicking on ***Contents*** on the top of the *Help* will bring the Contents listing back up.

Continue to Index or return to 6 12d Model Help.

Index

The *Index* tab searches through all entries in the Index of the Help.

As the first few characters of the required entry are typed in, the matching index entries are displayed.



Double clicking on the displayed entries will go to the topic in the Help containing the selected index entry. If more than one topic includes the index entry, then the list of topics is displayed.

If the index has sub-indices, they can be searched by first typing in the main index followed by a comma, then a space and the first few characters of the sub-index.

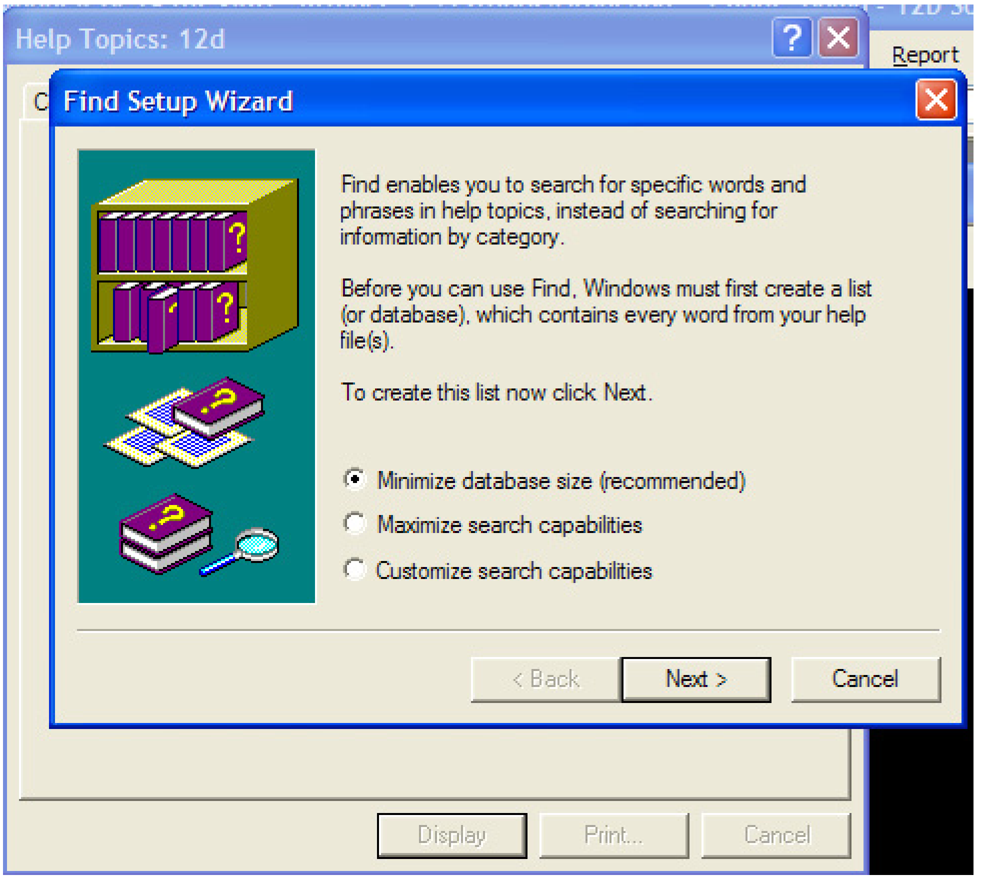
Continue to Find or return to 6 12d Model Help.

Find

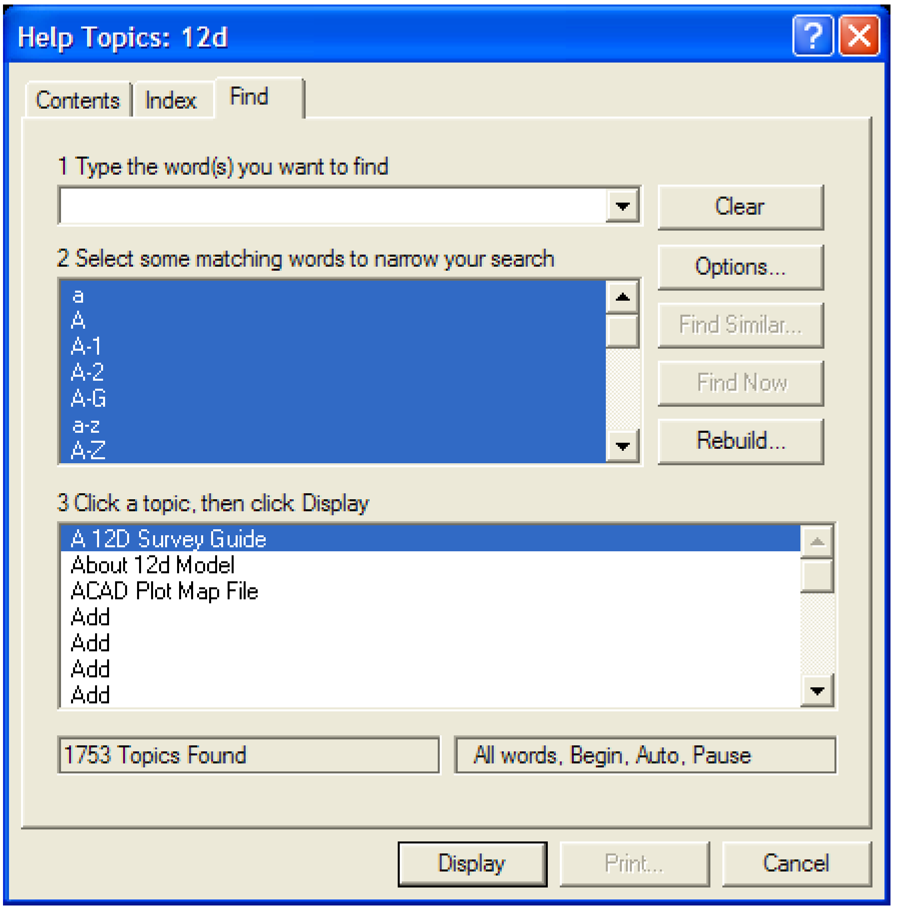
The most powerful searching method for the *Help* system is *Find*.

Simply click on the *Find* tab to search for words or phrases that may be contained in a Help topic.

If *Find* is being invoked for the first time, the ***Find Setup Wizard*** runs to create an index of every word in the Help.



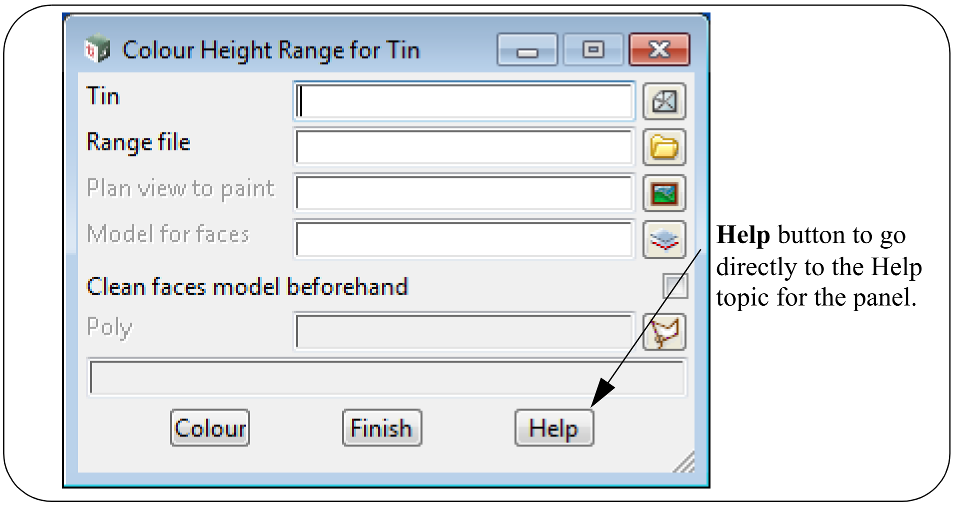
From then on, selecting the *Find* tab goes straight to the *Find* screen.



Continue to Panel Help Button or return to 6 12d Model Help.

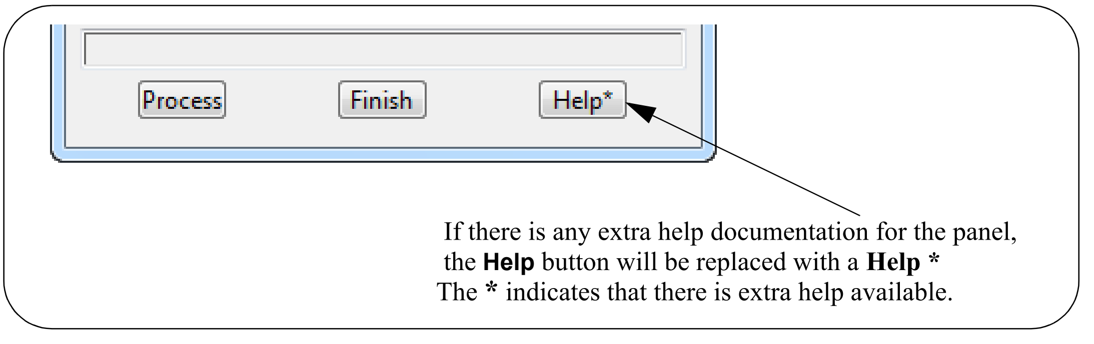
Panel Help Button

Every panel has a **Help** button which when selected goes to the *topic* describing that panel.



The default ***12d Model*** **Help** is all in one *Winhlp* file but a method for displaying additional help information exists so 12d Solutions, 12d Distributors and Users can supply additional (extra) **Help** information.

If there is extra help available for an option, then **Help*** will appear instead of **Help** on the panel button.



Information on how the extra help is set up is given in the section Extra Help.

Continue to F1 Key or return to 6 12d Model Help.

F1 Key

Another method of invoking **Help** is by using the **F1** key as follows:

> when a **menu** or **panel** is on the screen and has focus (the menu or panel title area will be highlighted), or the cursor is over an item on a **toolbar**, pressing **F1** will bring up the *help* for that menu, panel or toolbar item.

Warning - some of the items on the *Strings* menu automatically start up a string select and change the focus from the panel to a View. This means that pressing F1 will bring up the Help for the View and not the Help for the panel.

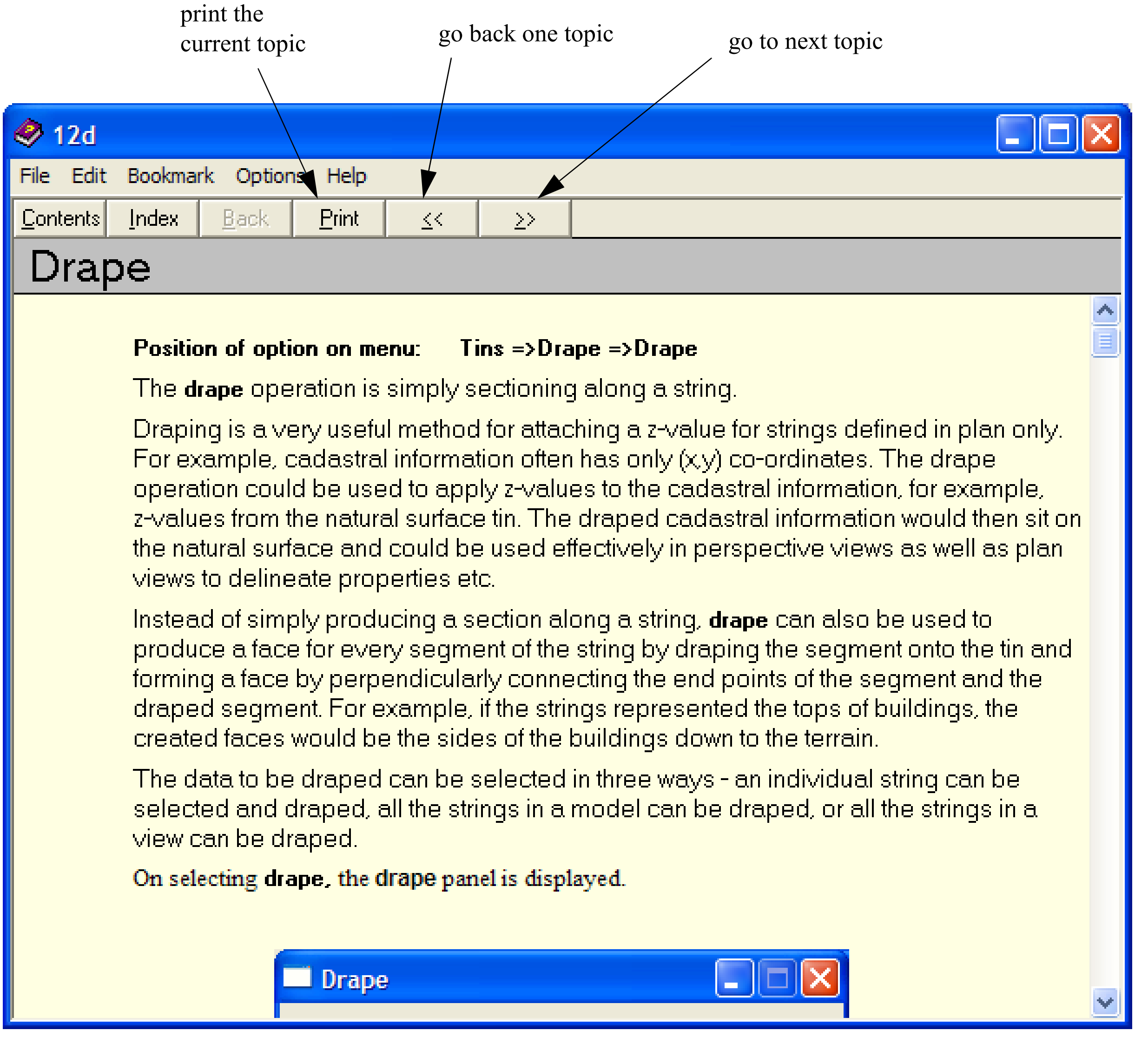
To get **Help** for such a panel, click on the panel to bring the focus back to the panel before pressing F1. The top of the panel will highlight showing that it has focus.

Continue to Navigating in Help or return to 6 12d Model Help.

Navigating in Help

Once at a *topic* in the Help, the **<<** and **>>** buttons at the top of the Help topic will go to the previous and next Help topics respectively.

Individual Help topics can be printed by clicking ***Print*** at the top of the Help page.



Because it is difficult to print large sections of Microsoft's Help system, a PDF file of the entire ***12d Model Reference*** Manual has been created and can be used to print out large sections of the manual.

The ***12d Model Reference*** Manual PDF file is on the ***12d Model*** *11 Installation DVD* in the folder ***Documentation\Reference Manual***.

Continue to Extra Help or return to 6 12d Model Help.

Extra Help

The default context sensitive ***12d Model*** **Help** is all in one help file supplied by *12d Solutions* but a method for displaying additional help information exists so 12d Solutions, 12d Distributors and Users can supply additional (extra) **Help** information. This extra information can also be supplied by ***12d Model*** PLs (macros) written by 12d Solutions or Users.

How to Set Up Extra Help

Any extra help for an inbuilt panel (that is, one not created by a macro) is placed in a folder with the same name as the dump name for the panel without the ending after the "." (to get the dump name, see ***Dumping a Panel, Creating a Screen Layout File or Default File*** in *12d Help* or the ***12d Model*** *Reference manual).*

For macros, created by Users or 12d Solutions, there can only be the same Help button for any panels created by the macro and the extra help for the macro is placed in a folder with the same name as the macro without the ending "4do" after the "." **and** with any blanks or non alphanumeric characters replaced by a underscore ("_"). For example, the extra help files for the macro called "testing help (3) system.4do" go in a folder called testing_help__3__system. Note there is an underscore for the blanks and the "(" and ")" in the macro name.

The extra help files for an inbuilt panel or macro can have *any name* and can be a pdf, wmv, avi. txt etc.

For example, for the panel ***Project Tree*** brought up by selecting **Project =>Tree,** the extra documentation would be in a folder called ***Project_Tree.***

The folder of extra help for a panel, is then placed in any one of the three places:

(a) in the *Help* folder in the ***12d Model*** installation area: For example, for version 11

c:\Program Files\12d\12d Model\11.00\Help

c:\Program Files (x86)\12d\12d Model\11.00\Help

(b) in a folder called *Help* inside the *Set_ups* folder in the ***12d Model*** installation area. For example

c:\Program Files\12d\12d Model\11.00\Set_ups\Help

c:\Program Files (x86)\12d\12d Model\11.00\Set_ups\Help

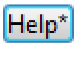
or

(c) in a folder called *Help* inside the *User* folder in the 12d User area. For example

c:\12d\11.00\User\Help

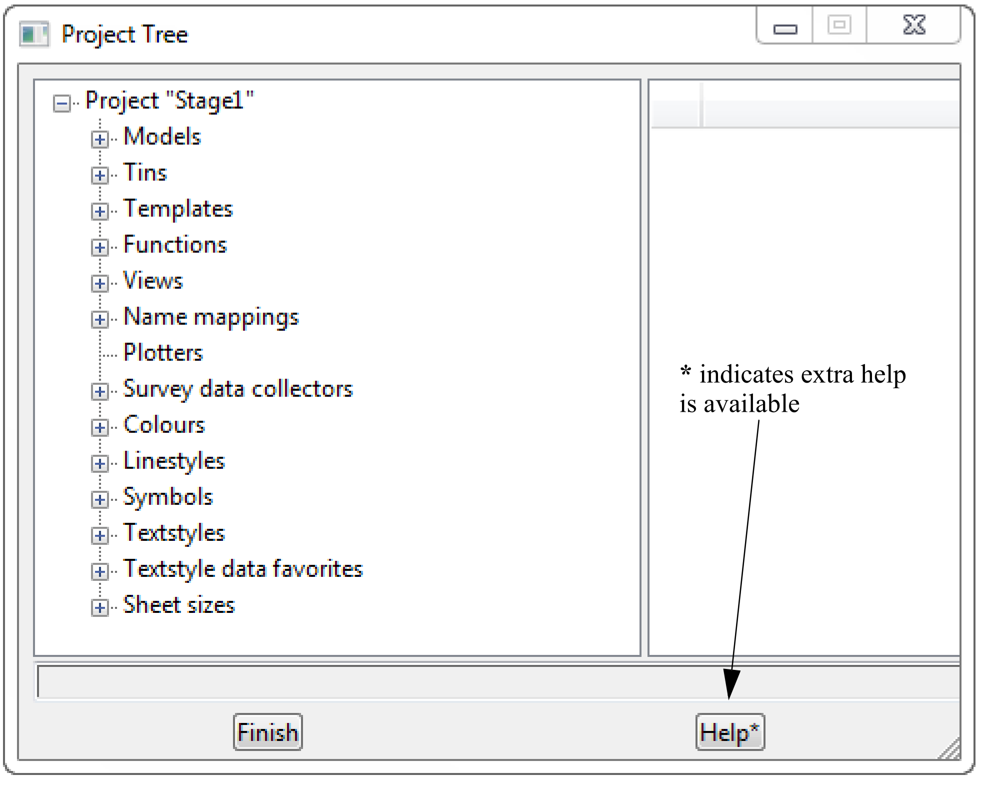
For an inbuilt panel an macro, each of these areas is searched and if any extra help is found, it is listed with the full path to each extra help file.

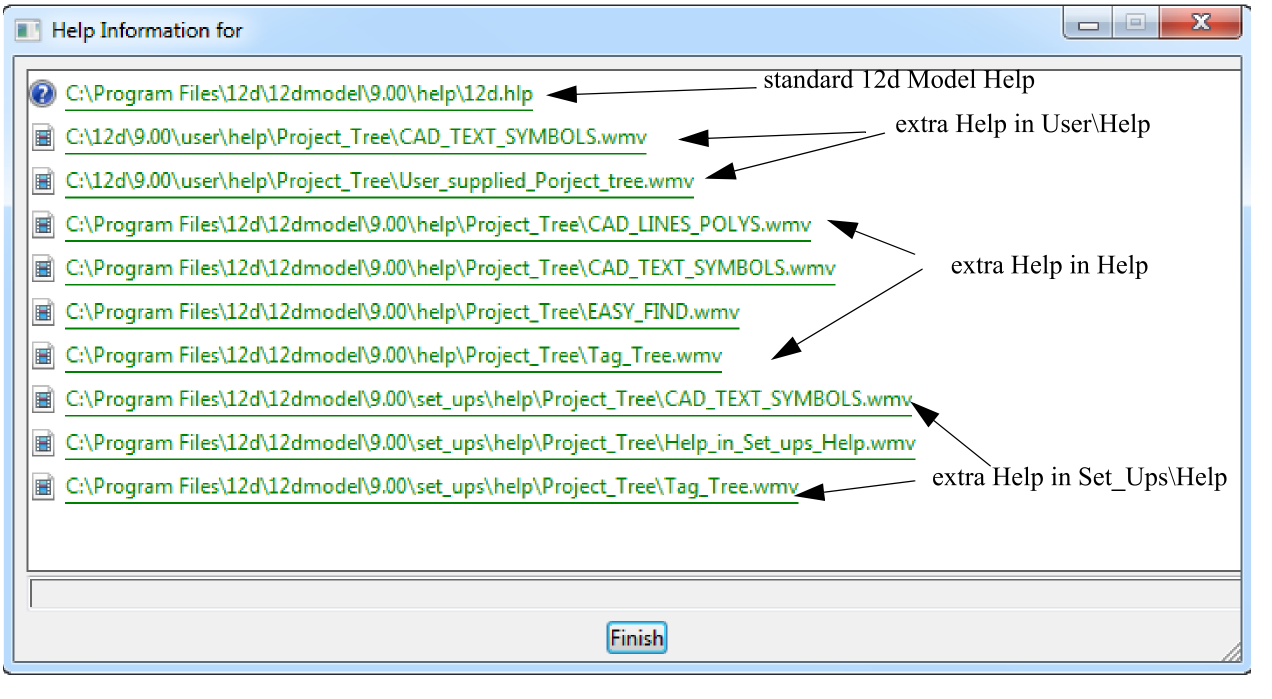
If there is any extra help for a inbuilt panel or macro, the **Help** button on the panel will be replaced with a **Help *** button. The * indicates that there is extra help available.



When you click on the **Help** * button, you will get a list of all the extra help files for that inbuilt panel or macro with the full pathname to the extra help. Clicking on the file name will bring up that extra help.

For example,



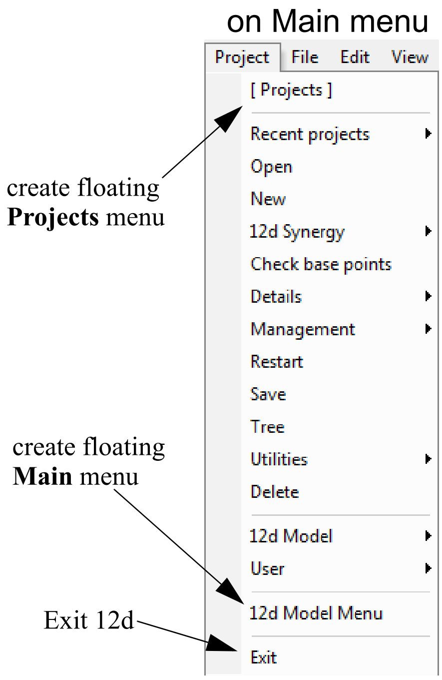


Users Own Extra Help Files

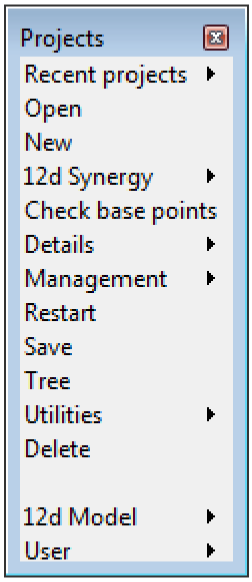
Note that users can also have their own extra help files and the files are simply placed in the correctly named folder under **User\Help**.

7 Project

The **Projects** walk-right menu is:



floating **Projects** menu



lists recent projects
open an existing project
create a new project
only available for 12d Synergy users
number of points for base point count
project description, details and diary
edit project setup/management files
exit and restart in same project
save the working project to disk
project data and set up files in tree form
project utilities
delete a project from disk

information on ***12d Model***
Projects User menu

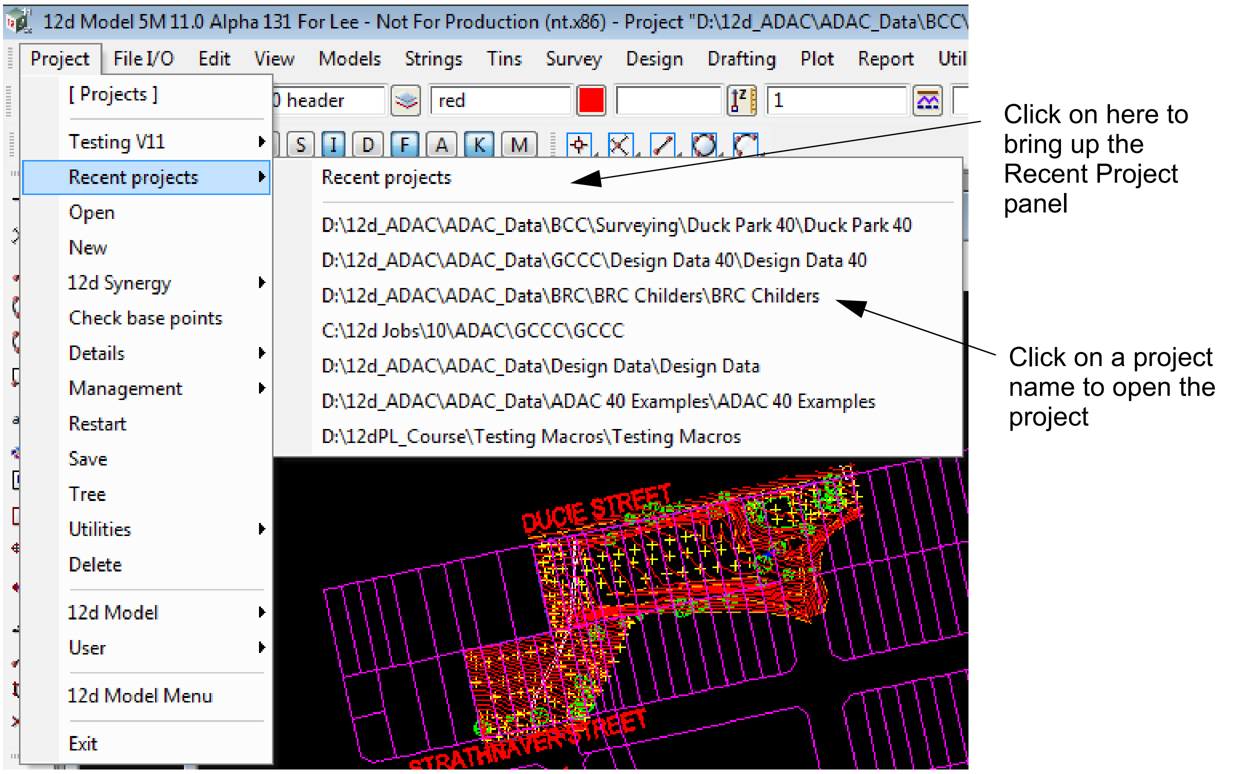
For the options, see:

Recent Projects	7.1 Recent Projects
Open	7.2.2 Open
New	7.2.3 New
12d Synergy	7.3 12d Synergy
Check base points	7.4 Check Base Points
Details	7.5 Details
Management	7.6 Management
Restart	7.7 Restart
Save	7.8 Save
Tree	7.9 Tree
Utilities	7.10 Utilities
Delete	7.11 Delete
Model	7.12 12d Model
Model Menu	7.13 12d Model Menu (only on the **Main Menu**)
Exit	7.14 Exit

7.1 Recent Projects

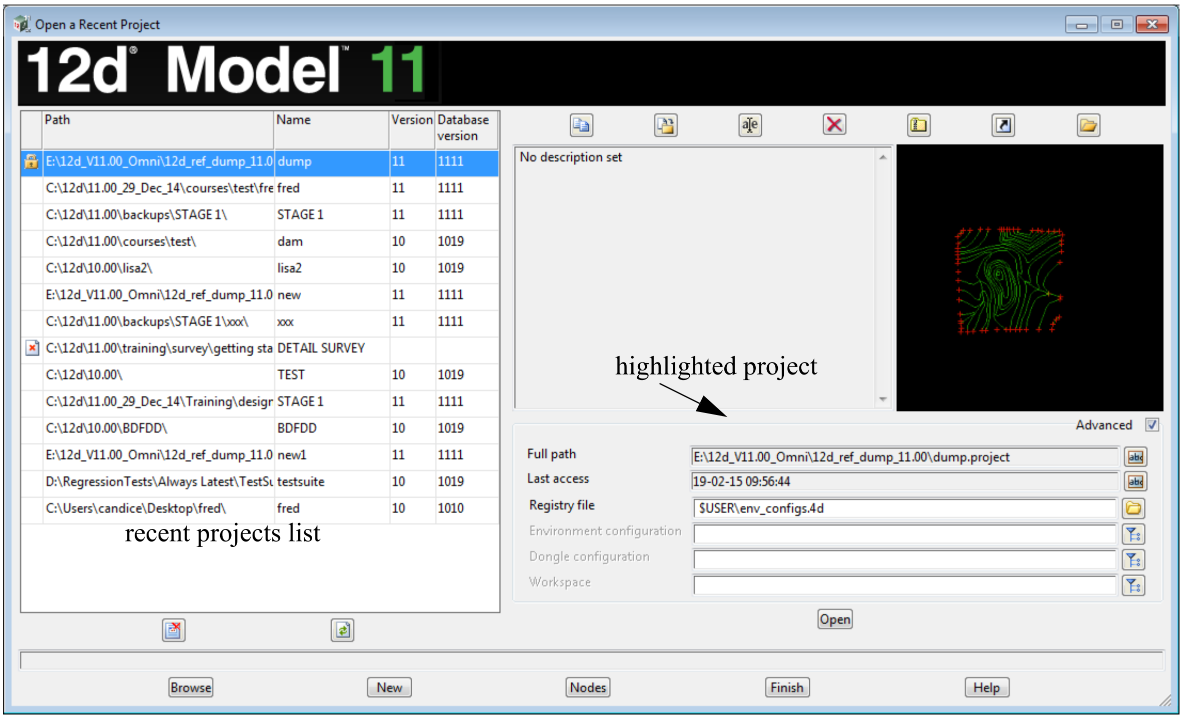
Position of option on menu: **Project =>Recent projects**

Walking right on the ***Project =>Recent projects*** lists the projects recently accessed by ***12d Model***.



Clicking on a project in the list will open the project.

Clicking on the ***Recent Projects*** heading on the ***Main Menu*** or on the **Recent projects** item when the **Projects** menu is pinned up, brings up the **Open a Recent Project** panel which shows the recent project in the front panel (see 5.1 Front Panel - Open a Recent Project).



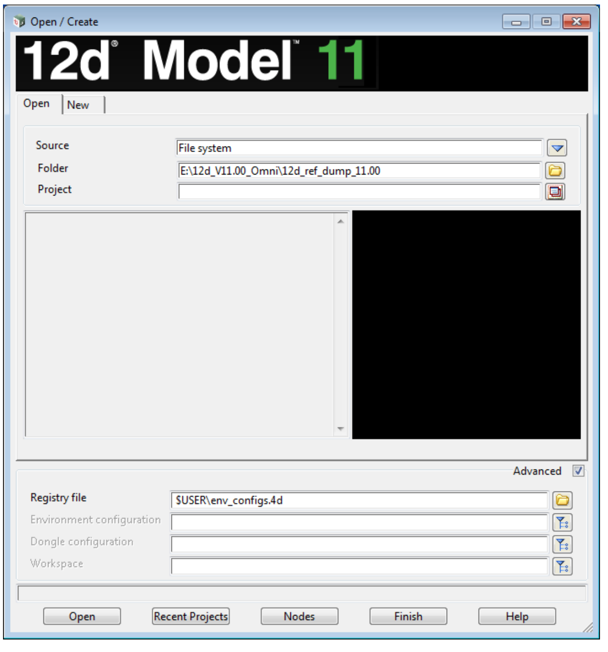
Go to the next section 7.2 Open/Create Project or return to 7 Project.

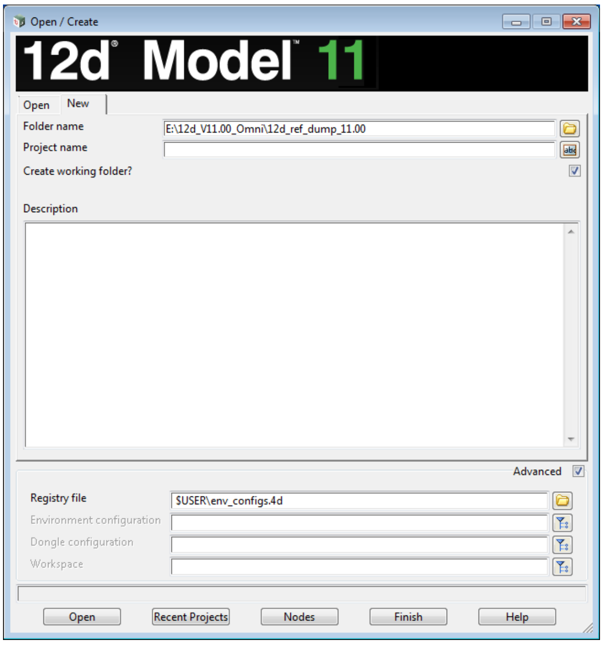
7.2 Open/Create Project

Position of option on menu: **Project =>Open**

Position of option on menu: **Project =>New**

Both **Open** and **New** bring up the **Open/Create** panel with the **Open** or **New** showing.





(a) The **Open** option is used to change the working folder and the working project to an another existing project.

With ***Advanced*** ticked, it is also possible to set or modify one or more of the *Environment configuration*, *Dongle* or *Workspace* set ups for the projects. For more information on Environment configuration settings, go to 7.6.4 Env Configuration or a quick discussion in the section 7.2.1 Project Open/Create Panel with Advanced Ticked.

The **Open/Create** panel with the **Open** tab showing is described in the section 5.2 Browse Button and Open Tab.

(b) **New** creates a new ***12d Model*** project.

A working folder (directory) of the same as the project can also be created with the actual project created inside the folder.

The **Open/Create** panel with the **New** tab showing is described in the section 5.3 New Button and New Tab.

7.2.1 Project Open/Create Panel with Advanced Ticked

To tailor ***12d Model,*** there are a large number of environment variables in a file called ***env.4d***.

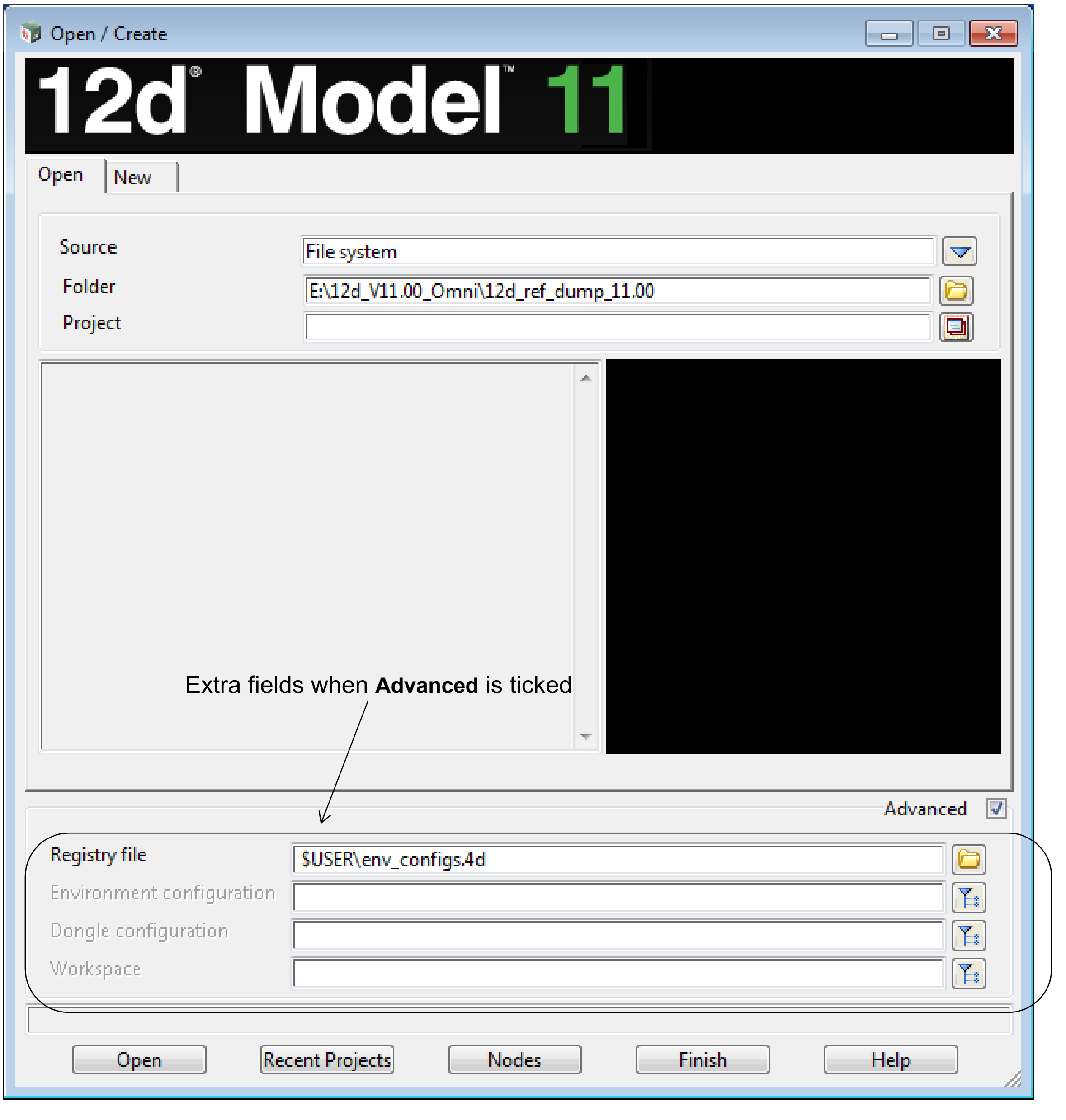
For most users, one ***env.4d*** file provides enough customisation for their site but for users with customers requiring very different set ups, a more flexible system is required.

For example, one project may be for a Main Roads Department which requires its own mapping files, linestyle files, ppf files etc, and another project is for a Local Authority who has totally different standards and requirements.

For this more complex situation, different ***env.4d*** files and environment variable overrides can be defined for each project and the **Advanced** tick box gives access to the advanced tailoring for a project.

For more information on the advanced tailoring, go to 7.6.4 Env Configuration.

With **Advanced** ticked, the **Open/Create and Open a Recent Project** panels display a section for extra configuration options



The extra fields and buttons in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Registry file	registry box		env_configs.4dfolder browse

the registry file used to define the Environments, Dongles and Workspaces definitions.

Environment configuration

if non blank, the given Environment configuration set up is used for this project.

The choices for the Environment configuration pop-up come from the registry file. For more information on defining Registry files, go to 7.6.4 Env Configuration *and for more information on the Environmental configuration section in particular, go to* Creating/Editing an Environments Set Up.

Dongle configuration

if non blank, the given Dongle set up is to be used for this project.

The choices for the Dongles pop-up come from the Registry file. For more information on defining Registry files, go to 7.6.4 Env Configuration *and for more information on the Dongles section in particular, go to* Creating/Editing a Dongles Set Up.

Workspace

if non blank, the given Workspace set up is used and if it includes a workspace file, then that file is used when the project is opened. However once the project is opened, the workspace file is saved locally and any changes saved with the project.

The choices for the Workspace pop-up come from the Registry file. For more information on defining Registry files, go to 7.6.4 Env Configuration *and for more information on the Workspaces section in particular, go to* Creating/Editing a Workspaces Set Up.

Go to the next section 7.2.2 Open or return to 7.2 Open/Create Project.

7.2.2 Open

Position of option on menu: **Project =>Open**

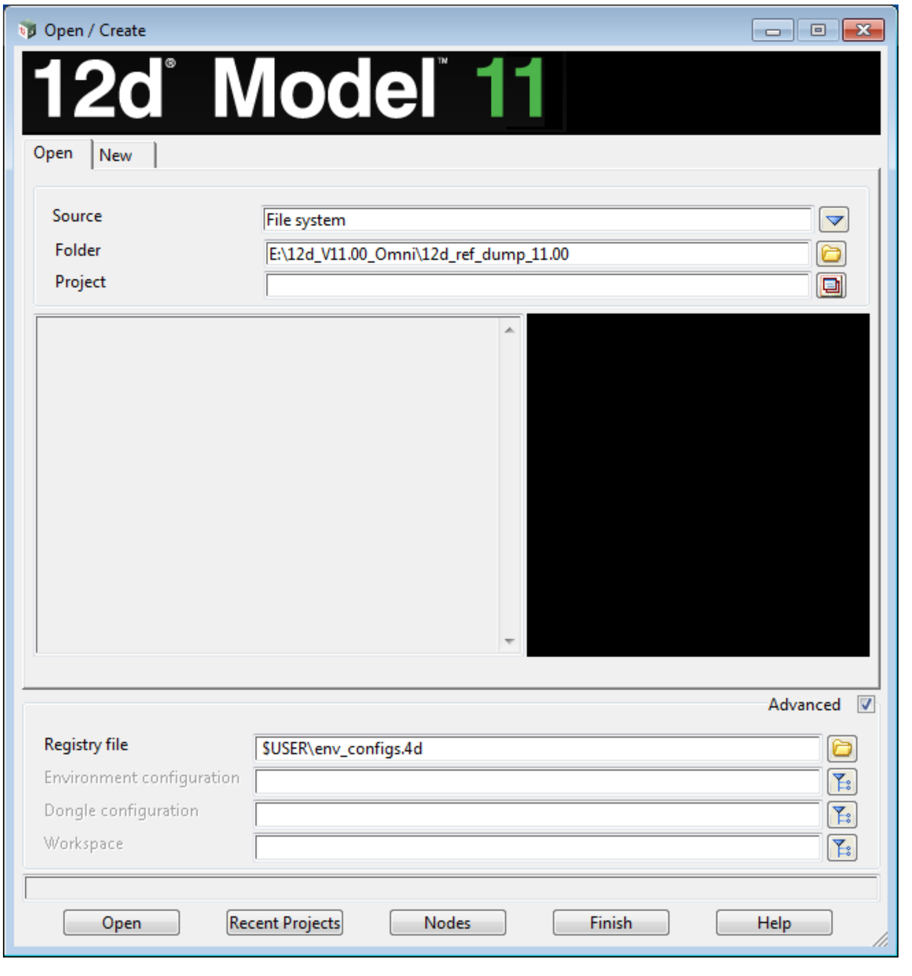
Position of option on menu: **Project =>Utilities =>Projects**

For the **Project =>Utilities =>Project** option, go to the section 7.10.2 Projects

The **Open** option is used to change the working folder and the working project to an another existing project.

With ***Advanced*** ticked, it is also possible to set or modify one or more of the *Environment configuration*, *Dongle* or *Workspace* set ups for the projects. For more information on Environment configuration settings, go to 7.6.4 Env Configuration or a quick discussion in the previous section 7.2.1 Project Open/Create Panel with Advanced Ticked.

Open option brings up the **Open/Create** panel with the **Open** tab showing, and this is described in the section 5.2 Browse Button and Open Tab.



Go to the next section 7.2.3 New or return to 7.2 Open/Create Project.

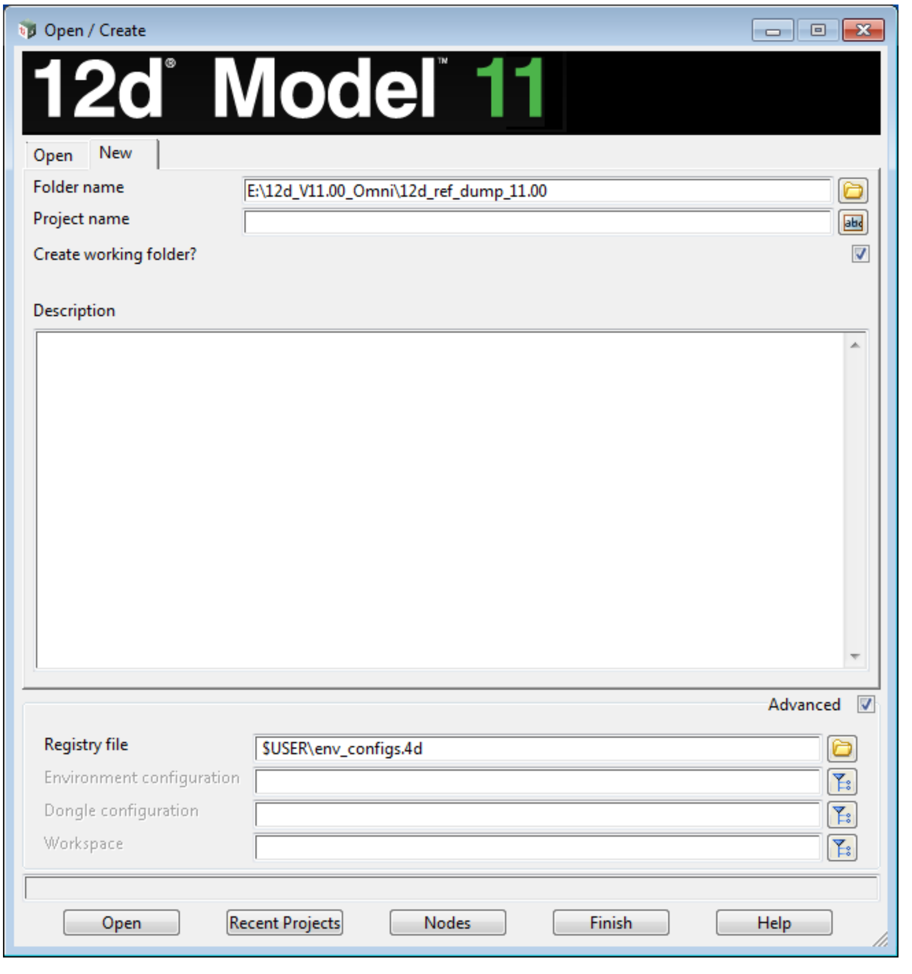
7.2.3 New

Position of option on menu: **Project =>New**

New creates a new ***12d Model*** project.

A working folder (directory) of the same as the project can also be created with the actual project created inside the folder.

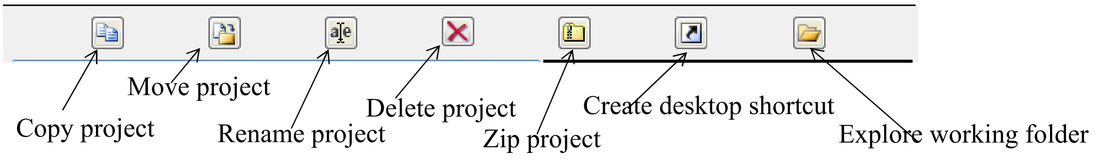
New option brings up the **Open/Create** panel with the **New** tab showing, and this is described in the section 5.3 New Button and New Tab.



Go to the next section 7.2.4 Options Active When Recent Project Highlighted or return to 7.2 Open/Create Project.

7.2.4 Options Active When Recent Project Highlighted

When a project is highlighted in the list of **recent projects** in the **Open a Recent Project** panel, icons for options are activated on the right hand side of the **Project Selection** panel.



Copy Project	see 7.2.5 Copy Project - Icon
Move Project	see 7.2.6 Move Project - Icon
Rename Project	see 7.10.7 Rename
Delete Project	see 7.2.7 Delete Project - Icon
Zip Project	see 7.2.8 Zip Project - Icon
Create Desktop Shortcut	create a shortcut on the desktop that starts ***12d Model*** and opens the project - see 5.6 Project Shortcuts
Explore	open Windows Explorer and show the contents of the working folder
Refresh	refresh the *recent projects list*

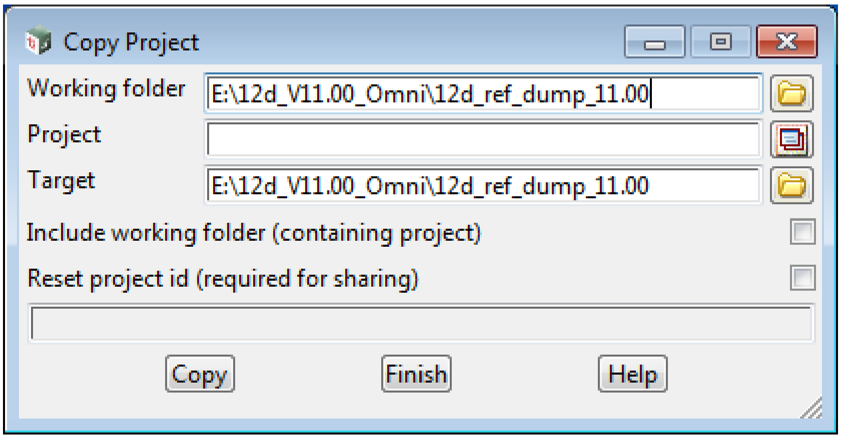
7.2.5 Copy Project - Icon

Position of option on menu: **Project Selection =>Copy Project icon**

Position of option on menu: **Project =>Open=> Copy Project icon**

This panel copies a project to a new location. If you are attempting to copy the project you are currently in, ***12d Model*** will prompt you to save and restart.

Selecting  brings up the **Copy Project** panel:



The fields and buttons used in this panels have the following functions.

Field Description Type Defaults Pop-Up

Working folder file

the folder containing the project

Project

the name of the project to copy

Target file

where the project should copied to

Include working folder (containing project) tick box

if ticked, the folder containing the project (and all sub folders) will also be copied

Reset project id (required for sharing) tick box

if you are intending to share the copied project, you must reset the ID - otherwise sharing will not be allowed from the copied project

Copy button

copies the project

Go to the next section 7.2.6 Move Project - Icon or return to 7.2 Open/Create Project.

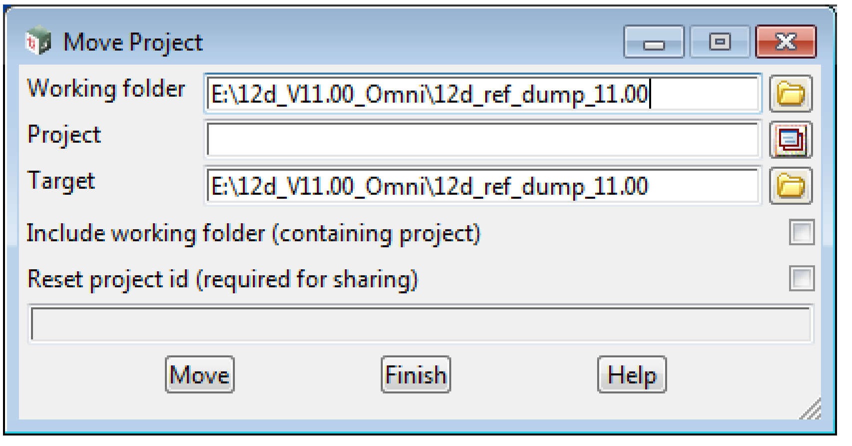
7.2.6 Move Project - Icon

Position of option on menu: **Project Selection=> Move Project icon**

Position of option on menu: **Project =>Open=> Move Project icon**

This panel moves a project to a new location. If you are attempting to move the project you are currently in, ***12d Model*** will prompt you to save and restart.

Selecting brings up the **Move Project** panel:



The fields and buttons used in this panels have the following functions.

Field Description	Type	Defaults	Pop-Up
Working folder	file		

the folder containing the project

Project

the name of the project to move

Target file

where the project should moved to

Include working folder (containing project) tick box

if ticked, the folder containing the project (and all sub folders) will also be moved

Reset project id (required for sharing) tick box

if you are intending to share the moved project, you must reset the id - otherwise sharing will not be allowed from the moved project.

Move button

moves the project

Go to the next section 7.2.7 Delete Project - Icon or return to 7.2 Open/Create Project.

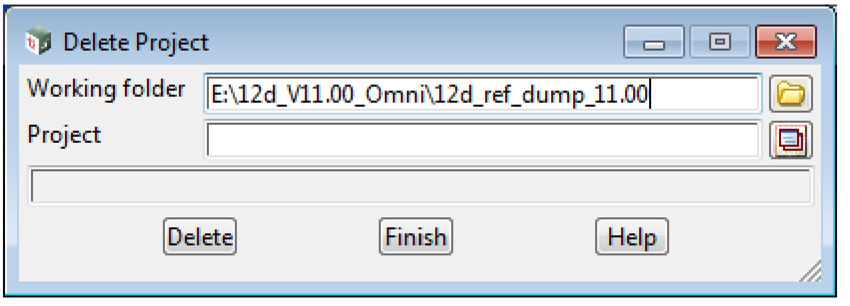
7.2.7 Delete Project - Icon

Position of option on menu: **Project Selection=> Delete Project icon**

Position of option on menu: **Project =>Open=> Delete Project icon**

This will delete a project. This cannot be the current open project.

Selecting  brings up the **Delete Project** panel:



The fields and buttons used in this panels have the following functions.

Field Description	Type	Defaults	Pop-Up
Working folder	file		

the folder containing the project to be deleted

Project

the name of the project to delete

Delete button

delete the project

Go to the next section 7.2.8 Zip Project - Icon or return to 7.2 Open/Create Project.

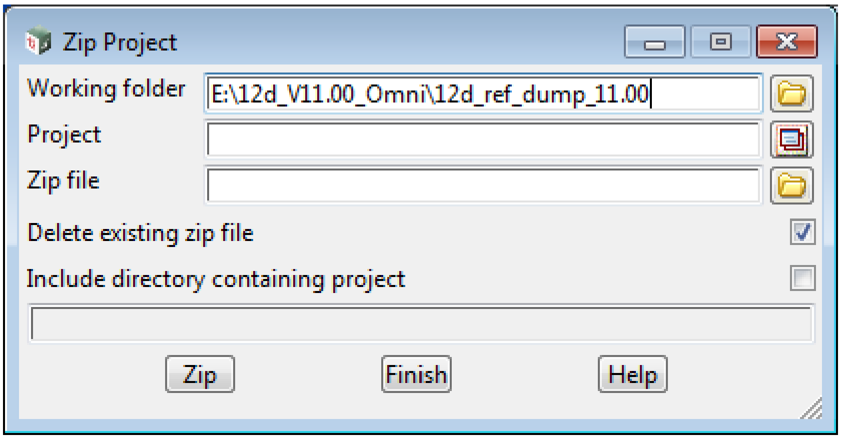
7.2.8 Zip Project - Icon

Position of option on menu: **Project Selection=> Zip Project icon**

Position of option on menu: **Project =>Open=> Zip Project icon**

This panel will zip a project. If you attempt to zip the current project, ***12d Model*** will prompt you to save and restart.

Selecting  brings up the **Zip Project** panel:



The fields and buttons used in this panels have the following functions.

Field Description	Type	Defaults	Pop-Up
Working folder	file box		

the folder where the project resides

Project	project box		

the name of the project you wish to zip

Zip file	file box		

the file to zip into

Delete existing zip file	tick box		

if ticked on, any existing zip file will be deleted

Include directory containing project tick box

if ticked on, the directory containing the project (the working directory) will also be zipped, along with all sub folders

Zip	button		

zips the project

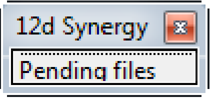
Return to 7.2 Open/Create Project.

7.3 12d Synergy

Position of option on menu: **Project =>12d Synergy**

This section of documentation is a work in progress and will be updated in subsequent releases.

On selecting the **12d Synergy** option,



Go to the next section 7.4 Check Base Points or return to 7 Project.

7.4 Check Base Points

Position of option on menu: **Project =>Check base points**

All ***12d Model*** versions have a Base module that is authorised for a set number of Base Points. For example, the 50K Base allows up to 50,000 base points, the 5M version allows up to 5,000,000 base points and the 250M version allows up to 250,000,000 base points.

The **Check base points** option displays the number of base points used in the current project and the total number of base points authorised for the version of ***12d Model*** being used to open the project.

The items included in the Base Point count are:

1. each vertex in a string counts as a base point except

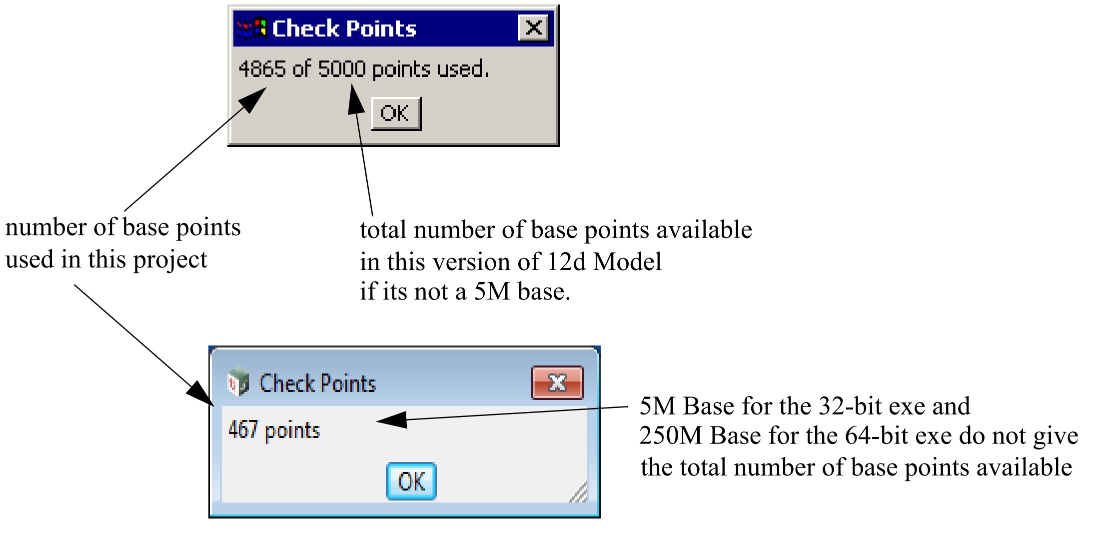
   (a) for a project, the vertices of strings created when plotting to a model inside the project do not count as base points for that project

   (b) for a project, contours generated by ***2d Model*** and written to models inside the project do not count for that project.

2. points in a tin count as base points
3. for an alignment, the number of base points is the number of HIP's plus the number of VIP's
4. items in models and tins shared into a project are included in the base point count.

Important Note:
If any non-counted base point data is written out as 12da and read into a project then the data read in from the 12da ***will*** be included in the base point count in the project that the data is read into.

Selecting **Check base points** displays the **Check Base Points** panel.



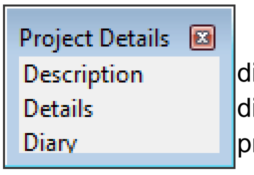
Go to the next section 7.5 Details or return to 7 Project.

7.5 Details

Position of option on menu: **Project =>Details**

A project description and project details can be defined.

The **Project details** walk-right menu contains various project items.



display/modify the project description
display/modify the project details
project diary

For the option *Description*, go to 7.5.1 Description
Details 7.5.2 Edit Details
Diary 7.5.3 Diary

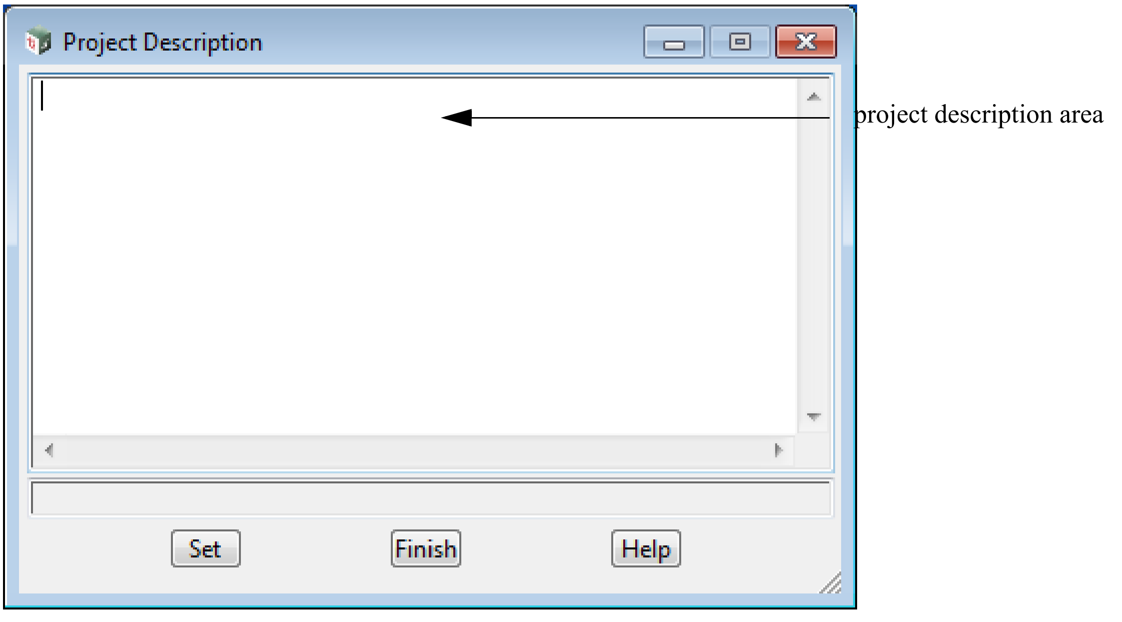
The options in the menu will now be described.

7.5.1 Description

Position of option on menu: **Project =>Details =>Description**

The **description** option is used to create and edit the project description.

Selecting **Description** brings up the **Project Description** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Project description area	input		

the description for the project to be rebuilt. The description can be more than one line long.

Set	button		

set the project description.to be the text in the **project description area**.

Note: This panel is a resizable panel.

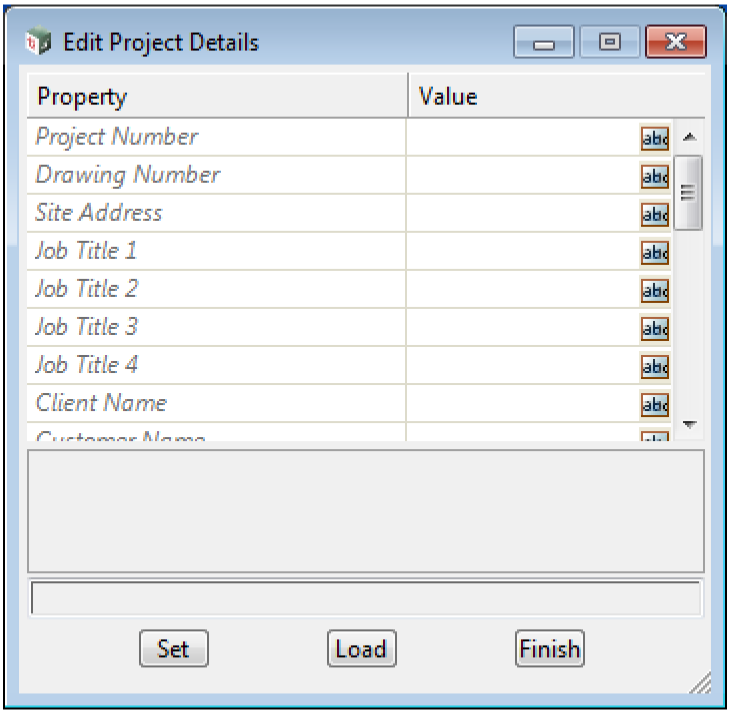
Go to the next section 7.5.2 Edit Details or return to 7.5 Details.

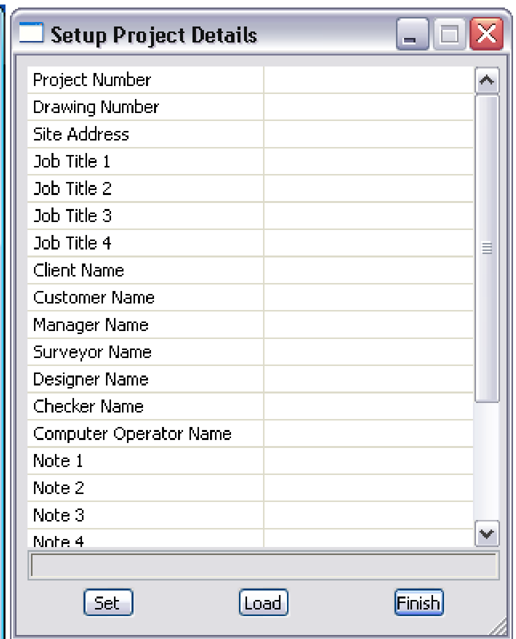
7.5.2 Edit Details

Position of option on menu: **Project =>Details =>Details**

The **Details** option is used to enter project details.

Selecting **Details** opens the **Edit Project Details** panel.





Panel displayed when new project started

What information is to be entered into the **Enter Project Details** panel is defined in the option

Project =>Management =>Details editor - *go to* 7.6.2 Details Editor

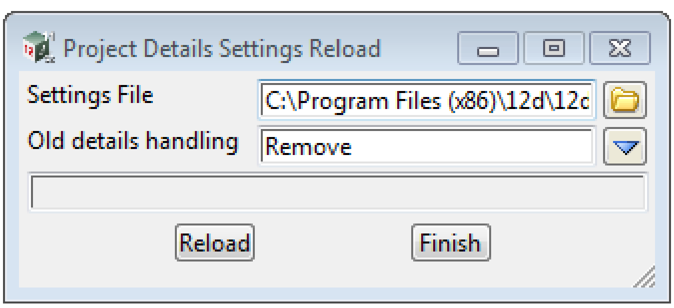
The fields and buttons used in the **Enter Project Details** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Set	button		

set the project details to the values show in the panel. The values are stored as project attributes.

Field Description	Type	Defaults	Pop-Up
Load	button		

brings up the **Projects Details Setting Reload** *panel that is used to load a new definition of project detail attributes.*



A file defining a new set of project attributes of project details can be loaded. How existing project details are handled depends on the value of **Old details handling**

Old details handling

Remove *- all existing project detail attributes are removed and totally replaced by the new set*

Keep (no update) *- any existing project detail are kept but their values* **are not** *updated by the default setting if the same attribute exists in the Settings file.*

Keep (with update) *- any existing project detail are kept and their values* **are updated** *by the default setting if the same attribute exists in the Settings file.*

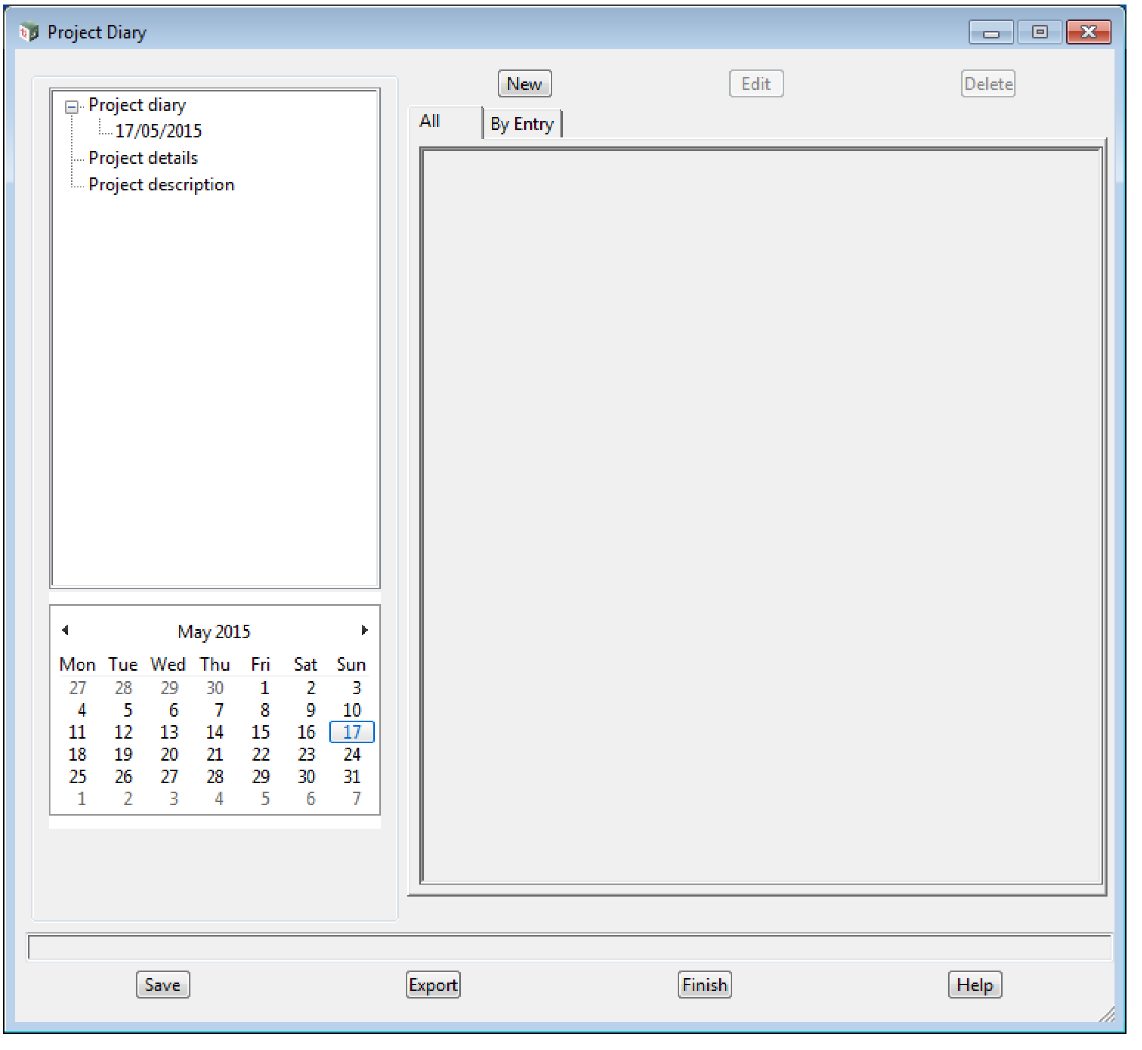
Go to the next section 7.5.3 Diary or return to 7.5 Details.

7.5.3 Diary

Position of option on menu: **Project =>Details =>Diary**

The **Project diary** allows information to be entered for each day. It can also be used to edit project details and the project description.

Selecting **Diary** brings up the **Project Diary** panel.



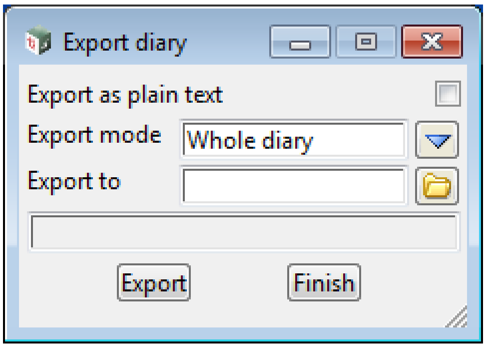
The **Project diary** panel has a grid with nodes for the **Project diary, Project Details** and **Project description**.

Clicking on the **+** on **Project diary** will expand the tree to show all the dates with diary notes as well as a position for entry at today's date. Clicking in the calendar at the bottom of the panel will allow notes to be added for earlier days that do not already have notes. Notes can not be added for future dates.

Clicking on **Project details** will display in read-only format the ***Project details*** information for the project with their current values. To modify any of the values, click on the **Edit** button and the **Edit Project Details** panel is displayed. The information can then be modified/saved as documented in the section 7.5.2 Edit Details.

Clicking on **Project description** display the current description in the right hand side of the panel. The description is simply text and can be modified directly in the panel. Click on the **Write** button to save the modified description.

Clicking on **Export** display the **Export Diary** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Export as plain text	tick box	not ticked	

if ticked, exports the diary as plain text without formatting
if not ticked, exports the diary in RTF, with all formatting retained

Export mode	choice box	Whole diary	Whole diary, Selected month, Selected day

which data should be exported - the whole diary, the selected month or the selected day

Export to	file		

the file to export the diary to

Export

writes the current export diary settings out to the file given in the **Export to** *field.*

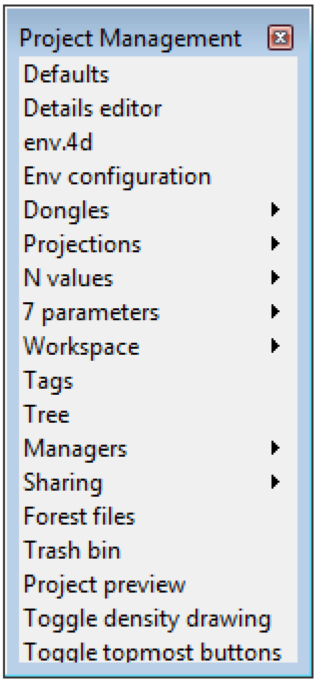
Return to 7.5 Details.

7.6 Management

Position of menu: **Project =>Management**

Options to set up the project

The **Project management** walk-right menu contains various project items.



set default values for the project
edit the details definition file
edit the env.4d file
set up environments with different user, user library folders, dongles
find network dongles, edit dongles.4d and certify CodeMeters
define/use projections
define/use N value parameters
define/use 7 value parameters
define/use workspace setup
create tag tree
edit items on the project tree
managers for project, tin, models etc
set share settings for this session
create/edit forest files
look at what models and tins are in the Trash Bin
set up user defined preview rather than automatic one
toggle density drawing - only for 250M
toggle on buttons to use instead of a keyboard when on tablets

For the option:

Defaults, go to	7.6.1 Defaults
Details editor	7.6.2 Details Editor
env.4d	7.6.3 env.4d
Env configuration	7.6.4 Env Configuration
Dongles	7.6.5 Dongles
Projections	7.6.6 Projections
N Values	7.6.7 N values
7 Parameters	7.6.8 7 Parameters
Workspace	7.6.9 Project Workspace
Tags	7.6.10 Tags
Tree	7.6.11 Tree
Managers	7.6.12 Managers
Sharing	7.6.13.2 Project Sharing
Forest files	7.6.14 Forest File
Trash bin	7.6.15 Trash Bin
Project preview	7.6.16 Project Preview
Toggle density drawing	7.6.17 Toggle Density Drawing
Toggle topmost buttons	7.6.18 Toggle Topmost Buttons

The options in the menu will now be described.

7.6.1 Defaults

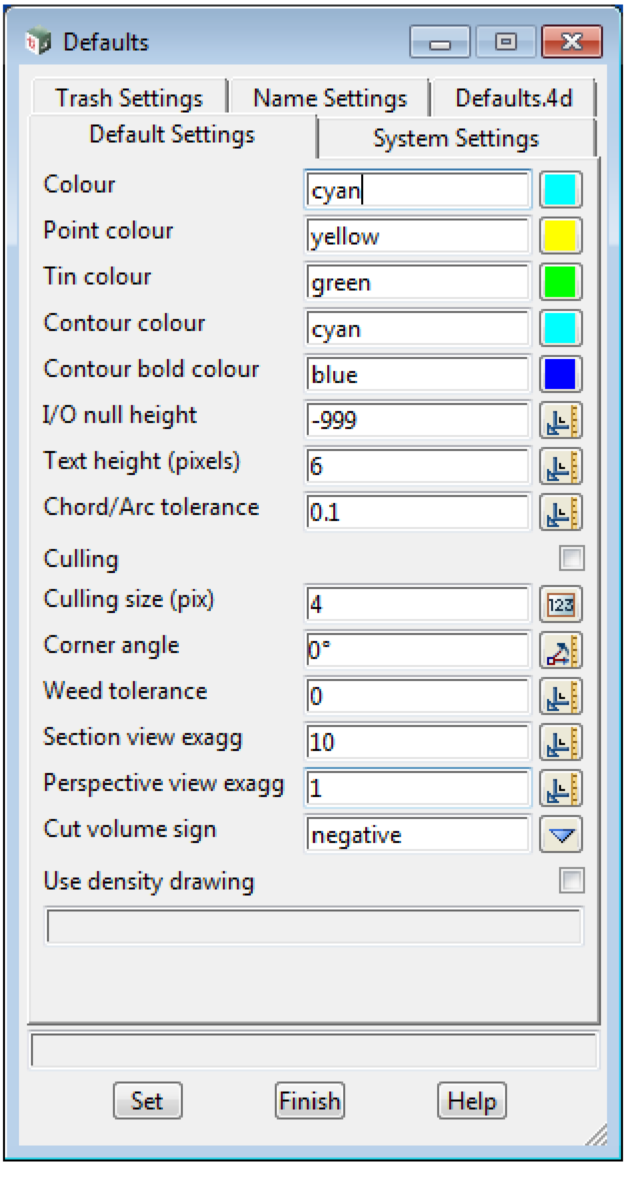
Position of option on menu: **Project =>Management =>Defaults**

The **Defaults** option allows the user to view and modify many of the default values used in the current 12d Model project.

NOTE - The values in the default option panels are initially set for a new project by the values in the **defaults.4d** file. For more information on the defaults.4d file, go to the section 39.2.7.3 Defaults File (defaults.4d) in the Appendix 39 Setting Up & Configuring 12d

Any changes made in the **Defaults** panel are used in the current session for the project when the **Set** button is pressed. The values are only *saved* for the project if a **Project =>Save** is done after the **Set**.

On selecting the **Default Settings** tab, the **Default Settings** are displayed.

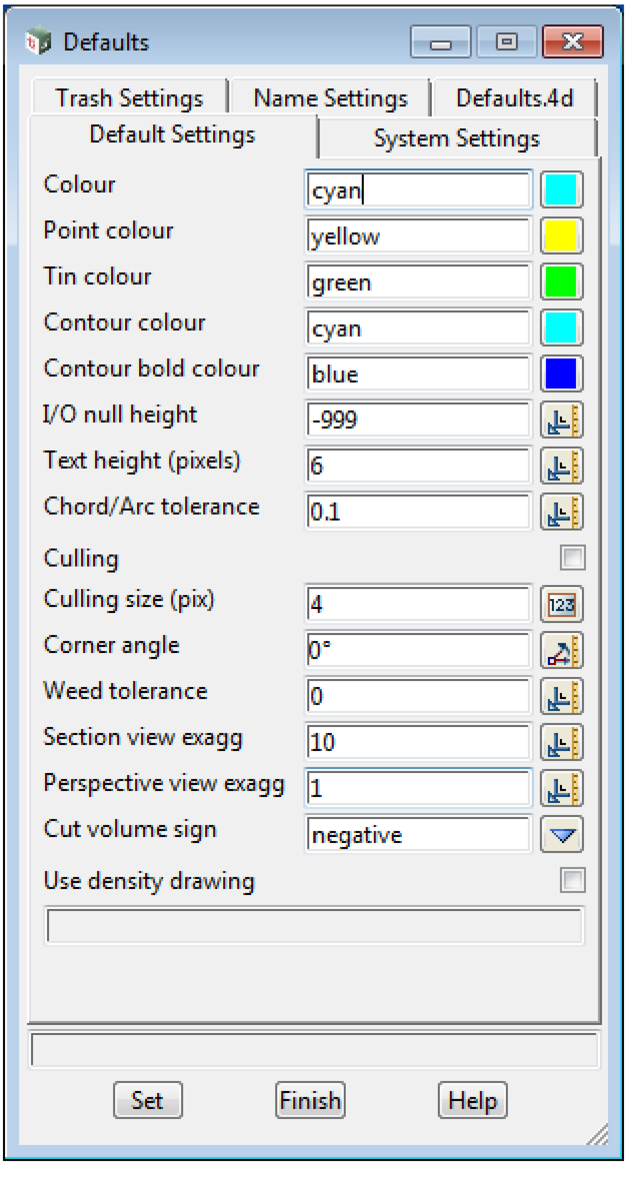


The **Defaults** panel consists of tabs for the groups of settings *Defaults*, *System*, *Trash* and *Name*.

Default Settings tab
System Settings tab
Trash Settings tab

Names Settings tab

Default Settings tab



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Colour	colour box	default colour	available colours

the name of the current default colour used for line strings.

Point colour	colour box	def point colour	available colours

the current default colour used for point strings.

Tin colour	colour box	def tin colour	available colours

the default tin colour used in the triangulate model and view panels.

Contour colour	colour box	default cont colour	available colours

the default contour colour used in the contour panel.

Cont bold colour colour box default cont index colour available colours

the default contour bold colour is used in the contour panel.

I/O null height input -999.0

this field contains the value of the value recognised as a null height when reading in and writing out data.

Text height (pixels) input 8

the default text height used in the sewer option and other panels.

Chord/Arc tolerance input 0.1

the maximum value of a chord to arc distance. If this distance is exceeded when approximating curves by chords, extra points are inserted into the curve so that the chords all have chord/arc distances less than this value. A value of zero disables the chord/arc test.

Culling tick box

the **culling** *setting for any new perspective or plan views.*

Culling size (pix) input 5

the **culling size** *for any new perspective or plan views.*

Corner angle input 15

when applying templates or calculating interfaces along a string, extra sections may be required at string vertices with no horizontal curve on them. If the **corner angle** *is non-zero, extra sections are added in at multiples of the* **corner angle** *value for the plan angle at the vertex.*

Weed tolerance input 0

if two points on a string (with the same bearing) are closer that this distance then the second point is left out. This applies to the extra points added in at chainage points in interfacing and corner angles.

Section view exagg input 10

the **vertical exaggeration** *used for any new section views.*

Perspective view exagg input 1

the **vertical exaggeration** *of any new perspective.*

Cut volume sign choice box negative negative, positive

the sign (positive or negative) used for cut volumes and areas. The sign for fill is the opposite.

Use density drawing tick box

If **Use density drawing** *is ticked, the data density for a model is calculated and if it is too high, a red rectangle is drawn around the model instead of the individual vertices of the strings in the model.*
If the data density of the model is low enough not to replace the entire model by a red rectangle, a data density is calculated for each string in the model and if the data density is too high, the string is replaced by a red rectangle.

Note*: This setting is only applicable to the 250M version of* ***12d* *Model.***

The default for a new project is given by the environment variable USE_DENSITY_CHECKS_4D.

Load button

read the default values from the current default.4d file for the project. The values are not used for the project until the ***Set*** *button is pressed. The values are not saved for the project until a* **Project =>Save** *is done after the* ***Set****.*

Set button

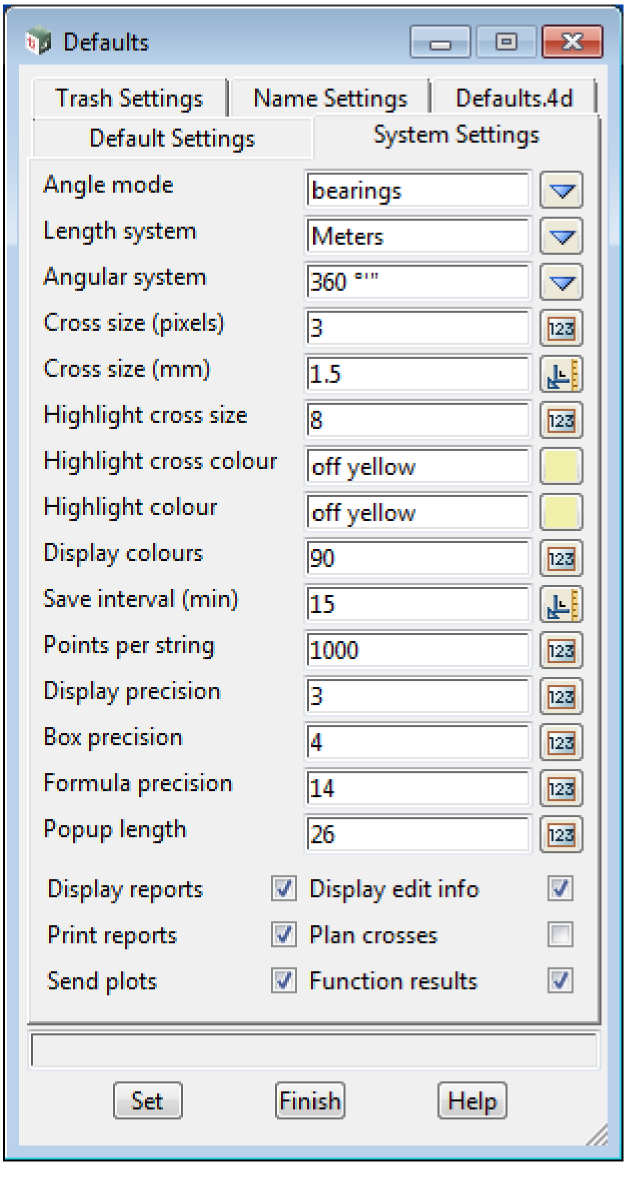
when set is selected, the defaults are set to the values given in the corresponding fields of this panel. These are not saved unless a **Project =>Save** *is done after the* ***Set****.*

Write button

write the values in the panel to the defaults.4d file.
The values are not used for the project until the ***Set*** *button is pressed. The values are not saved for the project itself until a* **Project =>Save** *is done after the* ***Set****. For more information on the* **Write** *button, go to the section* 39.2.6 Writing Out Setup Files *in the Appendix* 39 Setting Up & Configuring 12d.

System Settings tab

On selecting the **System Settings** tab, the **System Settings** are displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Angle mode	choice box	bearings	cartesian, bearings

specifies whether bearings or cartesian angles are used in reporting the instantaneous direction of the selected string in the **information** *menu.*

Cross size (pixels) input/output 2

the size in pixels that crosses (for points etc.) are drawn on the screen.

Cross size (mm) input/output 2.5

the size in millimetres that crosses (for points etc.) are drawn on any plots.

Highlight cross size input/output 8

the size in pixels of the cross used for highlighting objects in views.

Highlight cross colour colour box yellow available colours

the colour of the cross used for highlighting objects in views.

Highlight colour colour box white available colours

the colour used to display objects in views when they are highlighted.

Display colours input/output 0

the number of colours from the colours.4d file that are displayed in a **Select Colour** *pop-up.*
If ***0****, all colours are displayed.*
The order that the colours are selected is given by the Pop-up number in the colours.4d file. See 4.22.3 Editing Colours.4d.

Save interval (min) input/output 5

the number of minutes that elapse after a save before the **save project reminder** *panel comes up. If 0, the panel never comes up.*

Display precision input/output 3

the number of decimal places used for values displayed in the information menus.

Box precision input/output 4

the number of decimal places used for values displayed in boxes and panels.

Popup length input/output 28

this is no longer used - the maximum number of items in a pop-up before breaking the pop-up into walk-rights.

Display reports tick box tick

if ***ticked****, as soon as a report is produced, it will be displayed in the editor defined by the environment variable, EDITOR_4D.*

Display edit info tick box tick

if ***ticked****, the edit info panel is automatically displayed whenever a string is created or edited.*

Print reports tick box tick

if ***ticked****, as soon as a report is produced, it will be passed to the script/program defined by the environment variable, PRINTER_4D.*

Send plots tick box tick

if ***ticked****, as soon as a plot is produced, it will be passed to the script/program defined by the environment variable, PLOTTER_4D.*

Plan crosses tick box

This option is experimental *- at the moment things will look messy when editing in a section view with plan crosses turned on.*
if ***ticked****, when the cursor is in a plan or perspective view, it is projected onto any section views as well.*

Function results tick box tick

if ***ticked****, function results such as volumes will be displayed on the screen every time a recalc is done.*
if ***not ticked****, no function results as displayed on a function recalc.*

Load button

read the default values from the current default.4d file for the project. The values are not used for the project until the ***Set*** *button is pressed. The values are not saved for the project until a* ***Project =>Save*** *is* *done after the **Set**.*

Set button

when set is selected, the defaults are set to the values given in the corresponding fields of this panel. These are not saved unless a **Project =>Save** *is done after the **Set**.*

Write button

write the values in the panel to the defaults.4d file.
*The values are not used for the project until the **Set** button is pressed. The values are not saved for the project itself until a* **Project =>Save** *is done after the **Set**. For more information on the* **Write** *button, go to the section* 39.2.6 Writing Out Setup Files *in the Appendix* 39 Setting Up & Configuring 12d

Trash Settings tab

On selecting the **Trash Settings** tab, the **Trash Settings** are displayed.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Trash mode	choice box	trash string	keep string, trash string, delete string

In many string options, new strings are created from existing strings. What happens to the original strings may be determined by the **trash mode***.*
If set to ***keep string****, the original strings will not be touched*
trash string*, the original strings will be moved to the trash model*
delete string*, the original strings will be deleted.*

Trash model	model box	Trash Model	available models

the model that trashed strings are put into. This model needs to be cleaned or deleted to permanently remove the strings.

Load button

read the default values from the current default.4d file for the project. The values are not used for the project until the ***Set*** *button is pressed. The values are not saved for the project until a* ***Project =>Save*** *is done after the* ***Set****.*

Set button

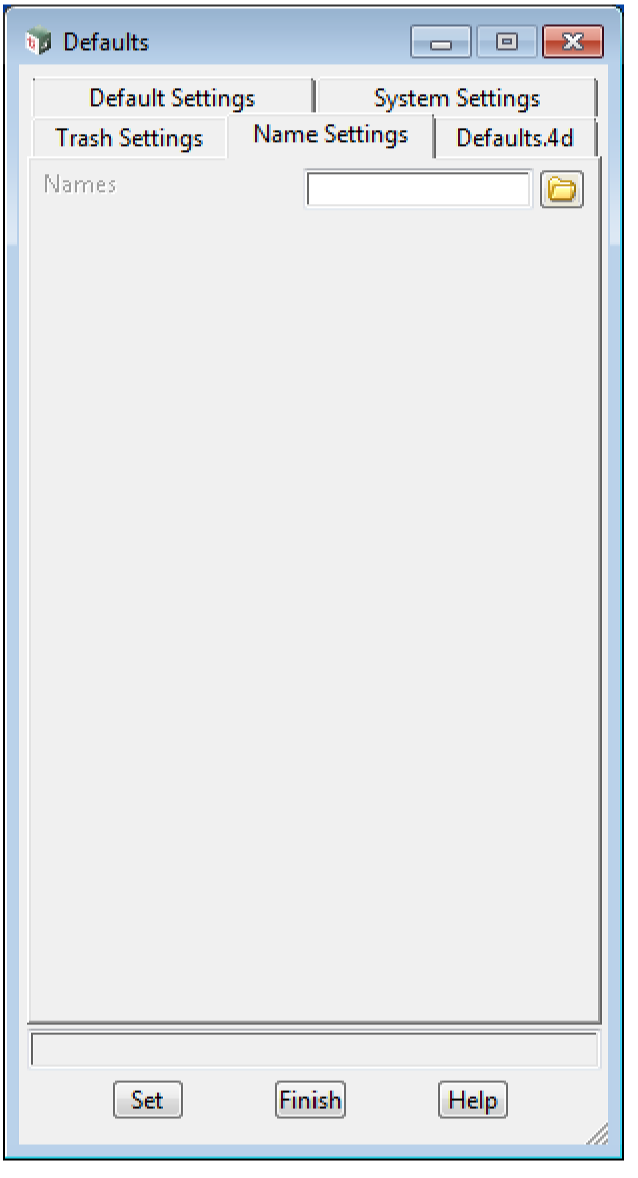
when set is selected, the defaults are set to the values given in the corresponding fields of this panel. These are not saved unless a **Project =>Save** *is done after the* ***Set****.*

Write button

write the values in the panel to the defaults.4d file.
The values are not used for the project until the ***Set*** *button is pressed. The values are not saved for the project itself until a* **Project =>Save** *is done after the* ***Set****. For more information on the* **Write** *button, go to the section* 39.2.6 Writing Out Setup Files *in the Appendix* 39 Setting Up & Configuring 12d

Names Settings tab

On selecting the **Name Settings** tab, the **Name Settings** are displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Names	input		*.mapfile, *.mf files

name of the Map File used as the default name mapping file.

Load button

*read the default values from the current **default.4d** file for the project. The values are not used for the project until the **Set** button is pressed. The values are not saved for the project until a **Project =>Save** is done after the **Set**.*

Set button

when set is selected, the defaults are set to the values given in the corresponding fields of this panel. These are not saved unless a **Project =>Save** *is done after the **Set**.*

Write button

write the values in the panel to the defaults.4d file.
*The values are not used for the project until the **Set** button is pressed. The values are not saved for the project itself until a* **Project =>Save** *is done after the **Set**. For more information on the* **Write** *button, go*

to the section 39.2.6 Writing Out Setup Files *in the Appendix* 39 Setting Up & Configuring 12d.

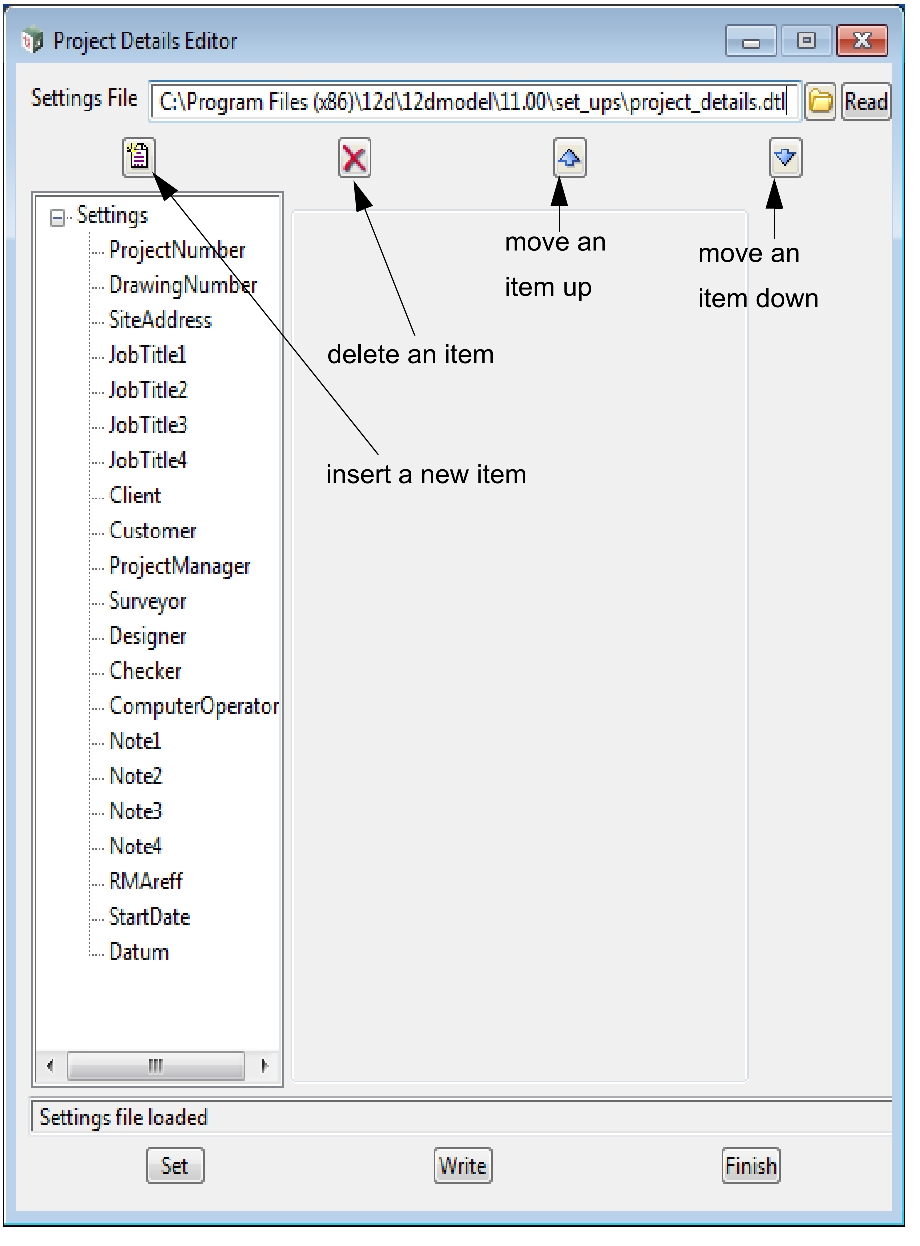
Go to the next section 7.6.2 Details Editor or return to 7.6 Management.

7.6.2 Details Editor

Position of option on menu: **Project =>Details =>Management =>Details editor**

The **Details Editor s** option creates the files used to define the project attributes displayed in the **Project Details** panel.

On selecting the **Details Editor** option, the **Project Details Editor** panel is displayed.



The items defined in the **Project Details Editor** panel can be of type Text (e.g. fred), Integer (e.g. 11) or Real (e.g. 23.15).

The fields and buttons used in the **Enter Project Details** panel have the following functions.

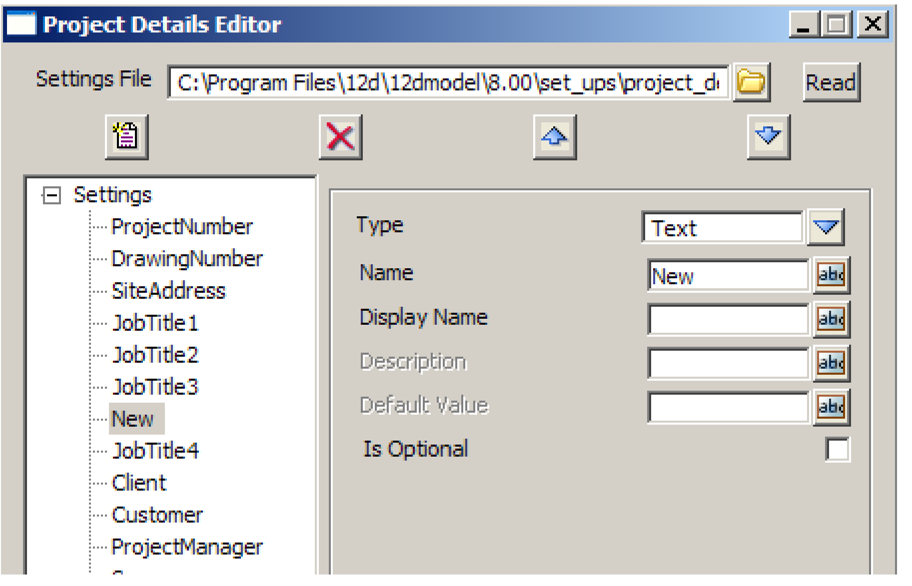
Field Description	Type	Defaults	Pop-Up
Set	button		

the values in the panel are set as the project details setup

Write	button		

writes out a project details file.

When inserting a new ***detail***, the following is displayed



Field Description	Type	Defaults	Pop-Up
Type	choice box		TextText, Integer, Real

the detail can be text, an integer or a real value

Name input

name of the project attribute used to store the detail

Display name input

name to appear in the **Enter Project Details** *panel*

Description input

description that appears when the **Display name** *is clicked in the* **Enter Project Details** *panel*

Default value input

default value for the detail

Precision integer box

only for Type Real - number of decimal places

Is optional tick box tick

*if **ticked**, then the detail does not have to be filled in.*
*If **not ticked,** it is compulsory to fill in the detail in the* **Enter Project Details** *panel before leaving the panel.*

Go to the next section 7.6.3 env.4d or return to 7.6 Management.

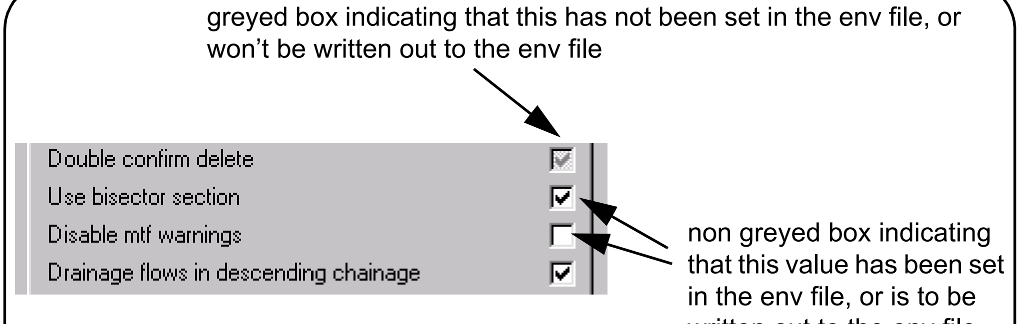
7.6.3 env.4d

Position of option on menu: **Project =>Management =>env.4d**

The **env.4d** option is used to create/update the *env.4d* file of environmental variables.

When the option is selected, it reads in the *current* env.4d file and displays in the panel, the values for any environment variables *in the file*. Hence the panel shows the values for the environment variables *in the file*, not those that are not in the file and have default values set by ***12d Model***.

Hence the tick boxes in the *Edit Environment Variables* panel have three states rather than the standard two states. There is the standard tick or nothing to denote the value of the variable is set from the env file (or is going to be written out to the env file) or a greyed out tick or greyed out blank box to indicate that the value has *not* been set by the environment variable. Clicking on the tick box will toggle between the three states - on, off and no value.

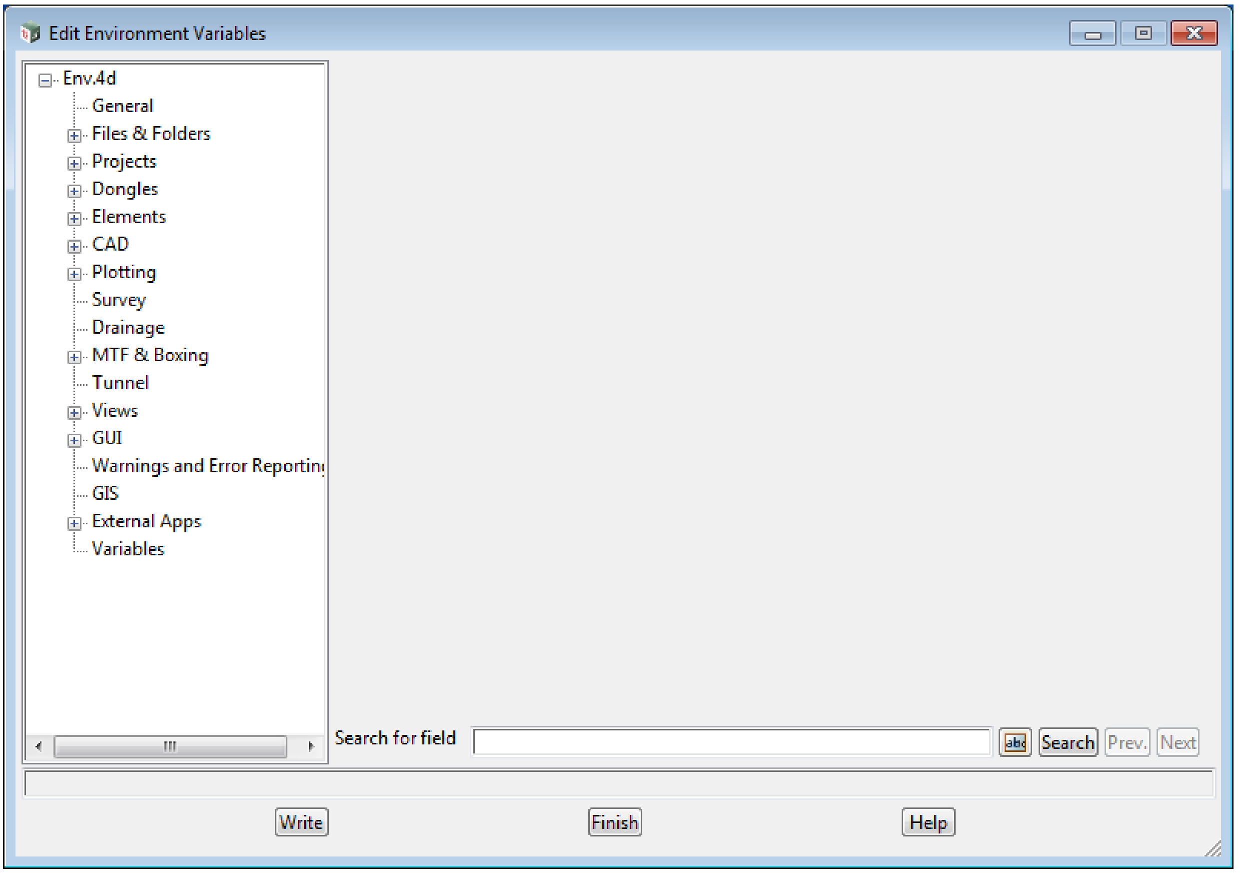


Three State Tick Box

After any modifications are made, the parameters are written out to an env.4d file.

Note that the *env.4d* file is only used when a project is loaded so the parameters written to an *env.4d* file can only take affect for the current project if a project *restart* is done.

Selecting **env.4d** displays the **Edit Environmental Variables** panel.



This panel has a tree on the left hand side and clicking on the **+** will expand the node and **-** will collapse the node.

The fields and buttons at the bottom of the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Search for field	typed input		

text to search for on the env.4d panel (not the env variable name).

Search *button - search for the text in the* **Search for field**. *The search starts from where your cursor is in the panel fields.*

Prev. *button - search for the previous occurrence of the text in the* **Search for field**.

Next *button - search for the next occurrence of the text in the* **Search for field**.

Write	button		

write the values in the panel to a file. For more information on the **Write** *button, go to the section* 39.2.6 Writing Out Setup Files *in the Appendix* 39 Setting Up & Configuring 12d

Note - the *env.4d* file is only used when a project is loaded so the parameters written to the env.4d file will only take affect for the current project if a project *restart* is done.

For information on the variables, see

General	General
Files & Folders	Files & Folders
Projects	Projects
Dongles	Dongles
Elements	Elements

CAD	CAD
Plotting	Plotting
Survey	Survey
Drainage	Drainage
MTF & Boxing	MTF & Boxing
Views	Views
GUI	GUI
Warnings and Error Reporting	Warnings and Error Reporting
GIS	GIS
External Apps	External Apps
Variables	Variables

The environment variables are described in more detail in the section 39.4 Environment Variables of the Appendix 39 Setting Up & Configuring 12d.

Field Description Type Defaults Pop-Up

General

Show full path name tick box

if ticked, when 12d Model fires up the actual file names defined by any environment variables are written to the output window. This is useful for debugging.

Env variable and full documentation: see SHOW_PATHS_4D

Show Vista full path name tick box

if ticked, then for Microsoft Vista, when 12d Model fires up, the actual file names defined by any environment variables are written to the output window. This is useful for debugging due to the fact that Vista may put file sin strange places.

Env variable and full documentation: see SHOW_VISTA_VIRTUAL_STORE_PATHS_4D

Enable undo tick box

if ticked, allow Undos.

Env variable and full documentation: see UNDO_4D

Plan table settings tick box

if ticked, allow the drawing of z-values, vertices, etc. on the plan view to be set by individual models rather than for all models on the view.

Env variable and full documentation: see PLAN_TABLE_SETTINGS_4D

Use names.4d comment field tick box

if ticked,

Environment variable USE_NAMES_COMMENT_4D

Check for updates on startup tick box

if ticked, the 12d web site is checked on startup for any newer updates to 12d Model.

If not ticked, the 12d web site is not checked.

Environment variable CHECK_FOR_UPDATES_4D

Show time taken for options tick box

if ticked, the time taken for an option to run is written to the output window.

If not ticked, the information is not written to the output window.

Environment variable SHOW_OPTION_EXECUTION_TIME_4D

Text editor file box

if non-blank, the script or program that is fired up when a report is created. It usually points to an editor.

Env variable and full documentation: see EDITOR_4D

Typed units mode choice box

Controls the typed input units for feet - international and/or US.

Environment variable TYPED_UNITS_MODE_4D

System names Choice box Long, Short, Short then long

for file name compatibility with DOS 8.3 format, the default names for all set up files can be restricted to just short names (8.3), long names, or short and then long.

Environment variable SYSTEM_NAMES_4D

Web search typed input www.google.com

Web address that is used in 12d option ***Help =>12d on the Web =>Search the web***

Env variable and full documentation: see WEB_SEARCH_4D

Processor affinity typed input

Environment variable PROCESSOR_AFFINITY_4D

Sort colour box by popup number tick box

In ticked the colours in the Select Colour popup are sorted by the Pop-up number.

Env variable and full documentation: see SORT_COLOURS_BY_POPUP_NUM_4D

Files & Folders

Files & Folders > Folders

Use "\Documents and Settings\All Users" for env.4d, users user_lib folders tick box

Env variable and full documentation: see USE_ALL_USERS_PROFILE_4D

Setups folder folder box default set_ups

if non blank, the full path name of the folder to use for the 12D Solutions supplied setup files.

Env variable and full documentation: see SET_UPS_4D

Library folder folder box default library

if non blank, the full path name of the folder to use for the 12D Solutions supplied library files.

Env variable and full documentation: see LIB_4D

User folder folder box default user

if non blank, the full path name of the folder to use for the User supplied setup files.

See User and Customer_User.

Env variable and full documentation: see USER_4D

User library folderfolder box default user_lib

if non blank, the full path name of the folder to use for the User supplied library files.

See 39.3.2 User Library.

Env variable and full documentation: see USER_LIB_4D

Customer user folder folder box no default

if non blank, the full path name of the folder to use for the Customer User supplied setup files.

See User and Customer_User.

Env variable and full documentation: see CUSTOMER_USER_4D

Customer library folder folder box no default

if non blank, the full path name of the folder to use for the Customer supplied library files.

See 39.3.3 Customer Library.

Env variable and full documentation: see CUSTOMER_LIB_4D

Log folder folder box default *working folder*

if non blank, the full path name of the folder to use for the **12d Model** *log files.*

Env variable and full documentation: see LOG_DIR_4D

Help folder folder box default *help*

If non blank, the full path name of the folder containing the **12d Model** *help files.*

Env variable and full documentation: see HELP_4D

Files & Folders > Files

Fonts file file box default fonts.4d

if non blank, the full path name of the file to use for defining text fonts.

Env variable and full documentation: see FONTS_4D

Function keys file file box default userkeys.4d

if non blank, the full path name of the file to use for function keys definitions.

Env variable and full documentation: see FUNCTION_KEYS_4D

Linestyles file file box default linestyl.4d

if non blank, the full path name of the file to use for defining linestyles.

Env variable and full documentation: see LINESTYLES_4D

Name mappings file file box default names.4d

if non blank, the full path name of the file to use for defining names file for the mapping of string names.

Env variable and full documentation: see NAME_MAPPINGS_4D

Textstyles file file box default textstyl.4d

if non blank, the full path name of the file to use for defining the text styles and fonts they use.

Env variable and full documentation: see TEXTSTYLES_4D

Textstyles favourite file file box default textstyle_names.4d

if non blank, the full path name of the file to use for defining the textstyles favourites.

Env variable and full documentation: see TEXTSTYLE_MAPPINGS_4D

Colours.4d file box default colours.4d

If non blank, the full path name of the file to use as the **12d Model** *colour file.*

Env variable and full documentation: see COLOURS_4D

Defaults.4d file box default defaults.4d

If non blank, the full path name of the file to use as the **12d Model** *defaults file.*

Env variable and full documentation: see DEFAULTS_4D

Digitizers.4d file box default digitizers.4d

If non blank, the full path name of the file to use as the **12d Model** *digitizers file.*

Environment variable DIGITIZERS_4D

Symbols.4d file box default symbols.4d

If non blank, the full path name of the file to use as the **12d Model** *symbols file.*

Env variable and full documentation: see SYMBOLS_4D

Patterns.4d file box default patterns.4d

If non blank, the full path name of the file to use as the **12d Model** *patterns file.*

Env variable and full documentation: see PATTERNS_4D

astyles.4d file box default astyles.4d

If non blank, the full path name of the file to use for defining super alignment symbology.

Environment variable SUPER_ALIGNMENT_STYLE_4D

On duplicate toolbar choice box Do nothing, Take First, Take last

This environment variable manages what to do if toolbars with the same name are read from a toolbars.4d file.

Do nothing - accept all toolbars of the same name
Take first - accept only the first instance of the toolbar
Take last - accept only the last instance of the toolbar

Env variable and full documentation: see TOOLBAR_DUPLICATE_MODE_4D

On duplicate linestyle / symbol Do nothing, Take First, Take last

This environment variable manages what to do if symbols or linestyles with the same name are read from the symbols.4d or linstyle.4d file

Do nothing *- accept all instances of symbols / linestyles of the same name*
Take first *- accept only the first instance of the symbol / linestyle*
Take last *- accept only the last instance of the symbol / linestyle*

Env variable and full documentation: see LINESTYLE_DUPLICATE_MODE_4D

Files & Folders > File input

Use anonymous functions for file inputs tick box

if ticked, when a file is read with a File input option, a function is automatically created and named. This function must exist to allow the data to be added to a view after it is read in, to be able to delete all the data read in at a later time, and to re-run the function to reread the data file and replace the data read in last time.

Env variable and full documentation: see ALLOW_ANONYMOUS_FUNCTIONS_4D

Add file input data to a view choice box

if anonymous function are set, the data created by the File input option can be added to a view. The choices are:

Do nothing *- don't add the data read in to any view*

Add to current view *- add the data read in to the current view (the current view is the view that highlighted)*

Add to new view *- automatically create a new view and add the data read in to that view*

Add to named view *- add the data read in to the view given by FILE_READ_ADD_TO_VIEW_NAME_4D (see* **Add file input data to view name).**

Env variable and full documentation: see FILE_READ_ADD_TO_VIEW_4D

Add file input data to view name text box

name of the view to use if FILE_READ_ADD_TO_VIEW_4D is set to ***Add to named view****. (see* **Add file input data to a view** *and* **Use anonymous functions for file inputs**).

If the view does not exist then it is created.

Env variable and full documentation: see FILE_READ_ADD_TO_VIEW_NAME_4D

Genio wildcard typed input *.mos

sets the ending of the files selected for the pop-up list for the **File** *field in the* **Read Genio Data** *panel.*

Env variable and full documentation: see GENIO_WILDCARD_4D

Files & Folders > File backups

Use backups.4d folder tick box default tick

if tick, backup files are placed in the folder backups.4d. in the project working folder (that is, in the folder containing the .project folder).

Env variable and full documentation: see USE_BACKUPS_4D_FOLDER_4D

Files & Folders > Usage Logs

Usage log folder folder box

if non blank, log files of the form

<log file folder>\(<dongle> <user> <computer> <time stamp> <process ID>.log)

will be created in the given folder.

Env variable and full documentation: see USAGE_LOG_4D

Usage logs folder folder box

if non blank, log files of the form

<log file folder>\<dongle>\<user>\<computer>\(<time stamp> <process ID>.log)

will be created in the given folder.

Env variable and full documentation: see USAGE_LOGS_4D

Projects

Projects > General

Short project names tick box

if ticked, use short extension names (3 characters after the .) for all internal files such as models, tins, projects.

Env variable and full documentation: see PROJECT_NAMES_4D

Setups file file box default setups.4d

if non blank, the full path name of the file to use for setting up the initial screen layout for new projects.

Env variable and full documentation: see SETUPS_FILE_4D

Recent projects positive integer box default 20

if non blank, the maximum number of accessed projects displayed in the Project list when ***12d Model*** *first starts up.*

Env variable and full documentation: see RECENT_PROJECTS_4D

Projects > Macros

Macro input mode tick box

controls whether or not the value passed down in the variable to receive the answer for any macro prompt, is actually placed into the console panel as the default answer so that it can be accepted by just typing <enter> into the console panel.

If ticked, put the passed down values into the console panel.

Env variable and full documentation: see MACRO_INPUT_MODE_4D

macros.4d (run on new project file box default macros.4d

if non blank, the full path name of the file of macros that is run when ***12d Model*** *creates a* ***new*** *project.*

Env variable and full documentation: see RUN_MACROS_FILE_4D

project_macros.4d (run on every project file box default project_macros.4d

If non blank, the full path name of the file of macros that is run when ***12d Model*** *opens an existing project.*

Env variable and full documentation: see RUN_PROJECT_MACROS_FILE_4D

Projects > Details

Show project details on new project tick box

If ticked, the Edit Project Details *panel is displayed when a new project is created.*

Env variable and full documentation: see SHOW_PROJECT_DETAILS_4D

Validate project details on startup tick box default tick

If ticked, the Project Details are validated when a project is opened and the user can not continue until all the details validate. See 7.5.2 Edit Details.

Env variable and full documentation: see ALWAYS_VALIDATE_PROJECT_DETAILS_4D

New project details file file box default

If non blank, the full path name of the file of project details file to use for new projects.

Env variable and full documentation: see PROJECT_DETAILS_4D

Projects > Reports

Amount of report header choice box no header, minimal header, full header

controls the amount of header information in reports.

Env variable and full documentation: see REPORT_HEADER_4D

Printer script file box

if not blank, points to a script or program which is fired up whenever a report is generated.

Env variable and full documentation: see PRINTER_4D

Projects > Sharing

Auto sync tins tick box default off

if ticked, the server projects for any shared tins added to this project are checked to see if they have been modified (checked every SHARE_CHECK_INTERVAL seconds). If any tins have been modified, they are re-copied to this project.

Env variable and full documentation: see AUTO_TIN_SYNC_4D

Auto sync models tick box default off

if ticked, the server projects for any shared models added to this project are checked to see if they have been modified (checked every SHARE_CHECK_INTERVAL seconds). If any models have been modified, they are re-copied to this project.

Env variable and full documentation: see AUTO_MODEL_SYNC_4D

Interval to check for updates positive integer default 0

if non zero, the number of seconds between checks to see if any of the shared tins or models added to the project have been modified.

Env variable and full documentation: see SHARE_CHECK_INTERVAL_4D

Share locking folder folder box default *inside project*

if non blank, the full path name of the folder used to keep lock files for shares.

Env variable and full documentation: see SHARE_LOCKS_FOLDER_4D

Colour for shared elements in list boxes colour box default blue

if non blank, the colour to use for showing shared tins/models in a list of tins/models. That is, the tins/models that have been added to the project as shared tins/models are shown in this colour. Setting the colour to black will disable this feature.

Env variable and full documentation: see SHARED_ELEMENT_COLOUR_4D

Colour for sharing elements in list boxes colour boxdefault 255,165,0

if non blank, the colour to use for showing tins/models that are allowed to be shared in a list of tins/models. That is, those tins/models in the project that the user has allowed others to share are shown in this colour. Setting the colour to black will disable this feature.

Env variable and full documentation: see SHARING_ELEMENT_COLOUR_4D

Sharing map file file box default none

if not blank, the full path name of the map file to be applied to shared models.

Env variable and full documentation: see SHARE_MAP_FILE_4D

Sharing cache folder file box default none

if not blank, the full path name of the folder to use for caching models and tins.

Env variable and full documentation: see SHARING_CACHE_4D

Projects > Trash bin

Use trash bin tick box default tick

if ticked, any deleted models or tins, or cleaned models, are placed in the Trash bin.

Env variable and full documentation: see USE_TRASH_BIN_4D

Days before auto purge positive integer default 0 (don't purge)

the number of days before the Trash Bin is automatically purged of tins and models.
*If 0, the Trash Bin is **not** purged.*

Env variable and full documentation: see PURGE_TRASH_DAYS_4D

Maximum trash bin size (Mb) positive integer default 0 (don't limit)

maximum number of Mb that the trash file can be.
*If 0, the Trash Bin is **not** limited in size.*

Env variable and full documentation: see MAXIMUM_TRASH_SIZE_4D

On overflowing trash bin Choice box Auto manage, auto empty default Auto manage

the action to take when the trash bin exceeds the maximum trash bin size.

*If **Auto manage,** the oldest files in the trash bin are deleted until the new item can fit in the trash bin.*
*if **Auto empty,** files in the trash bin are deleted.*
Note - if a large model or tin is deleted and it is bigger than the maximum trash bin size, the user is alerted and asked to decide if they want the element to go in the trash bin anyway, or if they want to permanently delete it.

Env variable and full documentation: see OVERFLOWING_TRASH_MODE_4D

Projects > Workspace

New project workspace file file box default

if non blank, the full path name of the workspace file for new projects.

Env variable and full documentation: see WORKSPACE_FILE_4D

Always show new toolbars tick box default

if ticked, when a project starts up, all the toolbars are checked to see if they are listed in the workspace (visible or invisible) and if the toolbar does not exist, then the toolbar will be displayed. This is to allow any new toolbars added to toolbars.4d to be automatically displayed so that the user knows that it exists.

Env variable and full documentation: see NEW_TOOLBARS_VISIBLE_4D

Dongles

Dongles > General

Time dongle access positive integer

time between searches for a dongle

Env variable and full documentation: see DEBUG_DONGLE_ACCESS_4D

Nodes file file box default nodes.4d *if not blank, the full path name of the* **12d Model** *authorization file.*

Env variable and full documentation: see AUTHORIZATION_4D

Dongles file file box default dongles.4d

if not blank, the full path name of the **12d Model** *dongles file.*

Env variable and full documentation: see DONGLES_4D

Dongles > Wibu (depreciated)

these environment variables for Wibu dongles have been replaced by the dongles.4d file. The dongles.4d file has the information about what Local and Network Wibu dongles and CodeMeter Containers to search for. See 7.6.5.3 Dongles.4d Editor.

Use Wibu dongle tick box

if ***ticked****, search for 12d Wibu dongles*
If ***not ticked****, don't search for Wibu dongles*

Env variable and full documentation: see WIBU_4D

Local choice box local, no local

if ***local****, then a stand alone Wibu dongle is looked for on the computer hat the user is on.*
If ***no local****, then no stand alone Wibu dongle is looked for on the computer*

Env variable and full documentation: see WIBU_DONGLE_4D

Network choice box no network, network first, network last

if ***no network****, then no Wibu network dongle is looked for.*
If ***network first****, then a Wibu network dongle is looked for before a single user local dongle.*
If ***network last****, then a Wibu network dongle is looked for after looking for a single user local dongle.*

Env variable and full documentation: see WIBU_DONGLE_4D

Login retries positive integer

number of retries to find a Wibu dongle
Env variable and full documentation: see WIBU_DONGLE_4D

Login wait positive integer

time in seconds to wait between Wibu dongle retries
Env variable and full documentation: see WIBU_DONGLE_4D

Disable dongle tick box

if ***ticked****, don't search for a Wibu dongle*
if ***not ticked****, search for a Wibu dongle*

Env variable and full documentation: see WIBU_DONGLE_4D

Debug dongle tick box

if ***ticked****,*
if ***not ticked****,*

Env variable and full documentation: see WIBU_DONGLE_4D

Wibu IP/Name addresses

if ***non blank****, a list of IP addresses and/or computer names to search for a 12d Wibu network dongle. The items in the list are separated by commas*
If ***blank****, search the entire network for a 12d Wibu network dongle*

Env variable and full documentation: see WIBU_IPADDR

Elements

Elements > Rasters

Use image server tick box

If ticked

Environment variable USE_IMAGE_SERVER_4D

Use ECW server tick box

If ticked

Environment variable USE_ECW_SERVER_4D

Elements > Strings

Use new string creates tick box

If ticked then the String Creates are for super strings.
If not ticked, the String Creates are the non-super string creates.

Environment variable NEW_STRING_CREATES_4D

Super strings ? tick box default tick

if ticked, super strings are allowed.
If not ticked super strings are not allowed.

Environment variable SUPER_STRINGS_4D

Only use super string tick box default no tick

if ticked, only super strings are created by all options.

Environment variable USE_SUPER_STRINGS_4D

Always user Super Strings advanced mode tick box default no tick

if ticked, the super string editor always comes up in the Advanced mode with all options available.

Environment variable SUPER_ADVANCED_MODE_4D

Use speed tables tick box default tick

if ticked, speed tables are enabled.
If not ticked, speed tables are not allowed.

Environment variable SPEED_TABLES_4D

Alignment VG Corridor fixup tick box default tick

if ticked, alignment corridor calculations introduced in V8 are used.
If not ticked, the V7 alignment corridor calculations are used.

Env variable and full documentation: see ALIGNMENT_CORRIDOR_FIXUP_4D

Use bisector section tick box default no tick

at a HIP with no curve on it, either two sections can be applied at the HIP point or just a single bisector section applied to the bisector of the change of angle through the HIP.

If ticked, the bisector section is used.
If not ticked, two sections will be created at the HIP.

Env variable and full documentation: see BISECTORS_4D

Weed Tolerance positive real

Used in Alignment and Super strings so that when arcs have been chord-to-arc'd, the resulting points are weeded so that no point is closer than the weed tolerance. Is is also used in Apply Template and Apply MTF so that no cross sections are closer than the weed tolerance.

Env variable and full documentation: see WEED_TOLERANCE_4D

Polyline draw choice box don't use speed ups
intermediate speed ups
faster speed ups

Different methods that speed up drawing of polylines.

Env variable and full documentation: see POLYPOLYLINES_4D

Elements > Tins

Tin viewport clip tick box shipped env.4d has this ticked

If ticked then some experimental techniques for speeding up the drawing of tins, fast contours is used.

Env variable and full documentation: see TIN_VIEWPORT_CLIP_4D

Pre*postfix for model for tin text box *default is* ***"tin "***

This environment variable is used to customize the default model for the tin in the panels for creating triangulations. Text can be defined for prepending and/or appending to the tin name to create a default model name from the tin name.

The text for prefixing and postfixing is given in a special form: ***pre-text*post-text***

If pretext only, just give the text. If post text is required, precede it by a ".".*

If the environment variable is not set, the default "tin " is used.

Env variable and full documentation: see MODEL_FOR_TIN_PREFIX_4D

Allow exact calculations for super tins tick box

Options using tins are split into two categories:

(1) options that require sections through tins
(2) options that require the triangles of the tin

if ticked, this allows super tins to be used in options where triangles are required for calculations.
if not ticked, this prevents super tins to be used in options where triangles are required for calculations.

Note: ticking this option requires super tins to be updated when a tin within the super tin changes, so there is a time, storage, and memory cost. It was found that some projects only ever required options that used sections (not triangles), so for these type of projects, there is an efficiency gain by turning off this option.

Env variable and full documentation: see ALLOW_SUPER_TINS_EXACT_CALCS_4D

Elements > Selects

Autopan on selects tick box default tick

if ticked, if you have accepted but not accepted a string and are zoomed in on the strings and type **ch value** *where the position at that chainage is off the view, the view will autopan so that the new selection point (at chainage* value*) is on the view. This applies for all typed selects.*

Env variable and full documentation: see AUTO_PAN_SELECT_4D

Auto hide panel on selects tick box default not tick

if ticked, when the string select icon is picked on a panel, the panel minimises until a string is selected *and accepted.*

Env variable and full documentation: see AUTO_HIDE_PANEL_SELECTS_4D

Pick only on selects tick box

if ticked,

Env variable and full documentation: see PICK_ONLY_ON_SELECTS_4D

Display all strings at snap tick box

if ticked,

Env variable and full documentation: see USE_DISPLAY_MANY_SNAP_4D

Show prettier information tick box

if ticked,

Env variable and full documentation: see USE_NEW_SELECT_4D

CAD

Fast accept tick box

*If ticked, the **Fast Accept** snap (**A** snap) is turned on by default for new projects.*

If Fast Accept (A snap) is on, when an item is picked and there is one item in the selection list, then the item is automatically accepted without clicking MB.

Env variable and full documentation: see FAST_ACCEPT_4D

Fast construction snap tick box

*If ticked, the **Fast Construction** snap (**K** snap) is turned on by default for new projects.*

Env variable and full documentation: see FAST_CONSTRUCTION_SNAPS_4D

Never snap to self tick box default tick

If ticked, the software tries to stop snapping to itself during editing.

Env variable and full documentation: see NEVER_SNAP_ITSELF_4D

Construction snaps model

If non blank, the model to use for objects created during construction snaps.

Env variable and full documentation: see CONSTRUCTION_SNAP_MODEL_4D

Plan height max real value box ‘ default 0

set the default value to use in Plan Settings panels that have a **Height max (w)** *field.*

Env variable and full documentation: see HEIGHT_MAX_DEFAULT_4D

Default CAD Modify mode choice box ‘ Multi pick

Env variable and full documentation: see CAD_START_IN_MULTI_PICK_4D

Plotting

Plotting > General

Plan plot scale 1: real box default blank

if non blank, the default plot scale to be used on any new plan views.

Note that the scale is used in the plan view for displaying text, linestyles and symbols defined in paper *units.*

Env variable and full documentation: see DEFAULT_PLAN_PLOT_SCALE_4D

Show title variables choice box as blank
as $variable
no substitution

this is used for debugging the title block file. Default is ***as blank****.*

If ***as blank****, any $variable not used is left as blank.*
If ***as $variable****,* any *$variable* not used is shown as $variable.
If ***no substitution****,* all *$variable* are plotted with no substitution.

Env variable and full documentation: see SHOW_TITLE_VARIABLES_4D

Symbol file for long and x-sections file box default *plotsymb.4d*

if non blank, the full path name of the plot symbols file.

Env variable and full documentation: see PLOT_SYMBOLS_4D

Plotter script file box

If non blank, points to a script or program which is fired up whenever a plot is generated.

Env variable and full documentation: see PLOTTER_4D

User plotters file box default *plotters.4d*

if non blank, the full path name of the file containing the definitions of plotters.

Env variable and full documentation: see PLOTTERS_4D

Plotter mapping file file box default *pmf.4d*

if non blank, the full path name of the file used as the default plotter mapping file.

Env variable and full documentation: see PLOTTER_MAPPING_4D

Sheets file file box default *sheets.4d*

if non blank, the full path name of the file defining the plot sheet sizes.

Env variable and full documentation: see SHEET_SIZES_4D

DGN plot seed file file box

if non blank, the full path name of the file to use as a seed file for Microstation plots.

Env variable and full documentation: see DGN_PLOT_SEED_FILE_4D

DWT plot template file file box

if non blank, the full path name of the file to use as a template file for AutoCAD plots.

Env variable and full documentation: see DWG_PLOT_SEED_FILE_4D

Acad plot unit choice box English, Metric

Env variable and full documentation: see DWG_PLOT_UNIT_4D

Use title blocks in 12a format tick box default tick

if ticked, the title block file is in 12da format.
If not ticked, the title block is in the pre V7 title block .tf format

Env variable and full documentation: see V7_TITLE_BLOCKS_4D

Use hardware arcs tick box default tick

if ticked, use computer hardware to draw arcs (rather than software).
If not ticked, draw arcs in software

Env variable and full documentation: see WRITE_ALL_PLOT_PARAMETERS_4D is preceded by HARDWARE_ARCS_4D

Write all plot parameters tick box default tick

Only used in old Plot options using (text) ppf files which have been superseded by binary ppf files. The panels have a Plot parameters write *field to write out an text ppf for the plot.*

If ticked, when writing out an (text) ppf files, write out all plot parameters.
If not ticked, only write out those plot parameters that have been used in the ppf file.

Env variable and full documentation: see WRITE_ALL_PLOT_PARAMETERS_4D

Show old plotting options tick box default tick

*If ticked, then a menu **Plot =>Old plotting** is included which has all the old text ppf options.*

Env variable and full documentation: see ALLOW_OLD_PLOTTING_4D

Offset chainages to output window tick box default not tick

If ticked,

Environment variable SPECIAL_OFFSET_CHAINAGES_4D

Plotting > Parameter files > Binary

Binary drainage plan parameter file file box default *none*

if non blank, the full path name of the file used as the default binary plot parameter file (.drainplanppf) for the plan annotation produced by the panel **Drainage Plan Plot PPF Editor**.

Env variable and full documentation: see NEW_DRAINAGE_PLAN_PPF_4D

Binary Melbourne Water parameter file file box default *none*

if non blank, the full path name of the file used as the default binary plot parameter file (.melbppf) for the long section plot produced by the panel **Sewer Plot Melbourne Water PPF Editor**.

Env variable and full documentation: see NEW_DRAINAGE_MELB_PPF_4D

Binary drainage parameter file file box default *none*

if non blank, the full path name of the file used as the default binary plot parameter file (.drainppf) for the long section plot produced by the panel **Drainage Plot PPF Editor**.

Env variable and full documentation: see NEW_DRAINAGE_PPF_4D

Binary pipeline parameter file file box default *none*

if non blank, the full path name of the file used as the default binary plot parameter file (.pipelineppf) for the long section plot produced by the panel **Pipeline Plot PPF Editor**.

Env variable and full documentation: see NEW_PIPELINE_PPF_4D

Binary plot frame parameter file file box default *none*

if non blank, the full path name of the file used as the default binary plot parameter file (.plotframeppf) for the long section plot produced by the panel **Section Long Plot PPF Editor**.

Env variable and full documentation: see NEW_PLOT_FRAME_PPF_4D

Binary long section parameter file file box default *none*

if non blank, the full path name of the file used as the default binary plot parameter file (.lplotppf) for the plan plot produced by the panel **Plot Frame PPF Editor**.

Env variable and full documentation: see NEW_LONG_SECTION_PPF_4D

Binary x-section parameter file file box default *none* *if non blank, the full path name of the file* used as the default binary plot parameter file (.xplotppf) for the cross section plot produced by the panel **Section X Plot PPF Editor**.

Env variable and full documentation: see NEW_X_SECTION_PPF_4D

Plotting > Parameter files > Ascii

X-section parameter file file box default *none*

Note - the text ppfs have been superseded by the binary ppfs.

if non blank, the full path name of the file used as the default text plot parameter file (.ppf) for the cross section plot produced by the panel **Cross Section Plot**.

Env variable and full documentation: see X_SECTION_PPF_4D

Long-section parameter file file box default *none*

Note - the text ppf's have been superseded by the binary ppfs.

if non blank, the full path name of the file used as the default text plot parameter file (.ppf) for the long section plot produced by the panel **Long Section Plot**.

Env variable and full documentation: see LONG_SECTION_PPF_4D

Drainage parameter file file box default *none*

Note - the text ppf's have been superseded by the binary ppfs.

if non blank, the full path name of the file used as the default text plot parameter file (.ppf) for the drainage long section plot produced by the panel **New Plot Drainage Network**.

Env variable and full documentation: see DRAINAGE_PPF_4D

Melbourne Water parameter file file box default *none*

Note - the text ppf's have been superseded by the binary ppfs.

if non blank, the full path name of the file used as the default text plot parameter file (.ppf) for the Melbourne Water long section plot produced by the panel **Melbourne Water Sewer Plot**.

Env variable and full documentation: see SEWER_PPF_4D

Plotting > Printing

Enable Windows printers tick box default tick

If ticked, use Windows printers

Env variable and full documentation: see WINDOWS_PRINTERS_4D

Update document properties before printing tick box

If ticked,

Environment variable WINDOWS_PRINTER_SET_DOCUMENT_PROPERTIES_4D

Windows 2000/XP print dialog tick box

If ticked, PrintD1gEX.
If not ticked, PrintD1g

Env variable and full documentation: see WINDOWS_PRINT_MODE_4D

Use exclusive access to printer tick box

If ticked, force direct printing to the printer (the user may need Printer admin access).

Env variable and full documentation: see WINDOWS_PRINT_MODE_4D

Use intermediate print file tick box

If ticked, print to a file first and then submit the file to the printer.

Env variable and full documentation: see WINDOWS_PRINT_MODE_4D

Raster resolutions dots/inch real value box *default 150.0*

number of dots per inch to use for plotting rasters

Env variable and full documentation: see DEFAULT_RASTER_DPI_4D

Maximum time for PDF995 (seconds) positive integer

the number of seconds to wait for PDF995 to finish producing the current PDF file.

Env variable and full documentation: see PDF995_TIME_LIMIT_4D

Windows printer resolution dots/mm choice box Windows 95/98/Me
Windows 2000/XP

For Windows 95, 98 and ME, the printer resolution can only be 0.04 mm when covering an A0 sheet.

Under Windows NT, 2000 and XP, no such restriction exists and the full resolution of 0.01 mm can be used so the environment variable HIMETRIC_4D allows access to the higher resolution for Windows NT, 2000 and XP.

Env variable and full documentation: see HIMETRIC_4D

Survey

Ignore extra words in .fld files tick box

If ticked, trailing tabs are not considered words in a field file. Mainly for Leica when writing a format file which can't suppress trailing tabs when writing a 12d Field file.

Env variable and full documentation: see FLD_IGNORE_XTRA_WORDS_4D

Allow named point attributes tick box

If ticked,

Environment variable ALLOW_NAMED_POINT_ATTRIBUTES_4D

Display commands in SDR Editor as per V8 tick box

If ticked,

Environment variable SDR_DISPLAY_V8_FORMAT_4D

Data collectors file file box default *survey.4d*

if non blank, the full path name of the file of definitions of available data collectors.

Env variable and full documentation: see DATA_COLLECTORS_4D

Data collector choice box data collectors in *survey.4d*

the data collector that is used if no data collector has been set for a project, and the default data collector that is used in the Survey Data Setup panel.

Env variable and full documentation: see DATA_COLLECTOR_4D

Station prefix text box

the Station prefix to user if no Station prefix has been set for a project, and the default Station prefix that is used in the Survey Data Setup panel.

Env variable and full documentation: see STATION_PREFIX_4D

DMS entry, treat 0.123 as 12 minutes 03 seconds tick box *if ticked, the special case of 0.mms is interpreted as mm minutes and s seconds. That is, 0.123 is interpreted as 12 minutes and 3 seconds.*

If not ticked, the special case of 0.mms is interpreted as mm minutes and 10 x s seconds. That is, 0.123 is interpreted as 12 minutes and 30 seconds.

Env variable and full documentation: see INTERPRET_DMS_INPUT_OLD_4D

Drainage

Drainage flow direction same as string direction tick box

if ticked, the drainage flow direction is the same as the string direction.
If not ticked, the drainage flow direction is in the opposite direction to the string direction.

Env variable and full documentation: see DRAINAGE_FLOW_DIR_4D

Show old grading edit options tick box

if ticked, the grading options used before V8 are still in the Drainage editor.

Environment variable DRAINAGE_EDIT_GRADE_4D

Show full hydraulic report details tick box *default not ticked*

the rational hydraulic report contains hydrology data.

If ticked, the hydrology calculations for each catchment set (1 to 3) are written to the hydraulic report.

Env variable and full documentation: see FULL_HYDRAULIC_REPORT_4D

Label 150 and PVC pipes tick box default ticked

if ticked, the 150 PVC pipes are labelled on the drainage long section.
If not ticked, the 150 PVC pipes are not labelled on the drainage long section.

Env variable and full documentation: see LABEL_PVC_150_PIPES_4D

Drainage.4d file box default *drainage.4d*

if not blank, the full path name of the file to use as the filed of ***12d Model*** *drainage definitions.*

Env variable and full documentation: see DRAINAGE_4D

MTF & Boxing

MTF & Boxing > MTF General

Disable MTF warnings tick box

if ticked, write mtf warnings to the background window.
If not ticked, display the mtf warnings in the text editor.

Env variable and full documentation: see DISABLE_MTF_WARNINGS_4D

MTF extra start/end tick box

if ticked, Extra start/end is turned on by default in the MTF Editor.

Env variable and full documentation: see EXTRA_START_END_MTF_4D

MTF extra start/end value measure box

the default extra start/end value for a new MTF. After a MTF is created, the value can be changed within the MTF, and saved with the MTF.

Env variable and full documentation: see MTF_EXTRA_START_END_VAL_4D

Default name for design layer

the default name for the design layer in a new MTF. After a MTF is created, the value can be changed within the MTF, and saved with the MTF.

Env variable and full documentation: see MTF_DESIGN_LAYER_NAME_4D

Name to access hinge as link

the default name used to refer to the Hinge string as a link in a new MTF. After a MTF is created, the value can be changed within the MTF, and saved with the MTF.

Env variable and full documentation: see MTF_CONSTRUCTION_HINGE_NAME_4D

Snippet temp file, use snippet name tick box

if ticked, the temporary file created after the preprocessing occurs for the snippet is given the same name as the snippet, with the file extension ***.tmp_mtf***

If not ticked, the MTF name is used. Note: If there is more than one snippet, this file will be overwritten.

Env variable and full documentation: see MTF_SNIPPET_TEMP_FILE_USE_SNIPPET_NAME_4D

For more information on snippets, see 21.5 Defining and Using Snippets.

Don't delete snippet temp file tick box

if ticked, the temporary files created after the preprocessing occurs for the snippet are left on the hard drive.

If not ticked, the temporary files are deleted.

NOTE: If the snippet is compiled, the temporary files are always deleted.

Env variable and full documentation: see MTF_SNIPPET_TEMP_FILE_USE_SNIPPET_NAME_KEEP_4D

For more information on snippets, see 21.5 Defining and Using Snippets.

Snippet temp file custom extension *text*

if not blank, the standard ***.tmp_mtf*** *file extension will be replaced with this value.*

If blank, the standard ***.tmp_mtf*** *file extension will be used.*

Env variable and full documentation: see MTF_SNIPPET_TEMP_FILE_EXTENSION_4D

For more information on snippets, see 21.5 Defining and Using Snippets.

MTF Seed files, include MTF files tick box

*if ticked, existing MTF files are included (in addition to files of type **.MTF_Seed**) in the choices of files which can be used as a MTF Seed file.*

*if not ticked, only files of type **.MTF_Seed** are included in the choices of files which can be used as a MTF Seed file.*

Env variable and full documentation: see MTF_SEED_INCLUDE_MTFS_4D

MTF & Boxing > MTF Editor General

Remember panel location/size tick box ticked

if ticked, when a MTF panel is placed on the screen and closed with Finish, its final position and size are recorded and used when the panel is opened again.

Env variable and full documentation: see REMEMBER_MTF_PANEL_SIZE_4D

Default MTF editor zoom buffer (%) *real box* *default 5*

the value is a percentage increase on the distance required to fit the arrows.

if Autopan is on for the **Left/Right MTF Modifiers** *panel, when a command is clicked in the panel and the region to pan is not displayed on the view, then the extent displayed is the required amount increased by the real_value percentage.*

Env variable and full documentation: see NEW_MTF_EDITOR_ZOOM_BUFFER_4D

Default MTF editor autopan on *tick box* *default tick*

*if ticked, the default for Autopan is **on** for the MTF* **Left/Right MTF Modifiers** *panel.*
*if not ticked, the default for Autopan is **off** for the MTF* **Left/Right MTF Modifiers** *panel.*

Env variable and full documentation: see NEW_MTF_EDITOR_AUTOPAN_DEFAULT_4D

Named part text size (pixels) measure box

the size of the text to use when the name of the named part is highlighted.

Env variable and full documentation: see MTF_NAMED_PART_HIGHLIGHT_TEXT_SIZE_4D

Pasted modifier colour colour box

when you cut/paste the modifiers, the background colour of the pasted rows will remain that colour until you edit them, i.e. this option lets you know the pasted rows until you edit them.

Env variable and full documentation: see NEW_MTF_EDITOR_PASTED_MODIFIER_COLOUR_4D

MTF & Boxing > MTF Hinge Colours

Hinge offset colour colour box

the colour of the text in the Type column for the Hinge Offset commands in the **Hinge Modifiers** *panel.*

Env variable and full documentation: see HINGE_OFFSET_ZONE_COLOUR_4D

Hinge height colour colour box

the colour of the text in the Type column for the Hinge Height commands in the **Hinge Modifiers** *panel.*

Env variable and full documentation: see HINGE_HEIGHT_ZONE_COLOUR_4D

Hinge position colour colour box

the colour of the text in the Type column for the Hinge Position commands in the **Hinge Modifiers** *panel.*

Env variable and full documentation: see HINGE_POSITION_ZONE_COLOUR_4D

MTF & Boxing > MTF Fixed/Snippet/Decision Colours

Fixed colour *colour box* *default blue*

the colour of the text in the Type column for the Fixed commands in the **Left/Right MTF Modifiers** *panel.*

Env variable and full documentation: see FIXED_ZONE_COLOUR_4D

Fixed width colour *colour box* *default light blue*

*the colour of the text in the Type column for the Fixed commands involving width, in the Modifiers section of the **Left/Right MTF Modifiers** panel.*

Env variable and full documentation: see FIXED_WIDTH_ZONE_COLOUR_4D

Fixed height colour *colour box* *default dark blue*

*the colour of the text in the Type column for the Fixed commands involving height, in the Modifiers section of the **Left/Right MTF Modifiers** panel.*

Env variable and full documentation: see FIXED_HEIGHT_ZONE_COLOUR_4D

Fixed xfall colour *colour box* *default dark blue*

*the colour of the text in the Type column for the Fixed commands involving xfall, in the Modifiers section of the **Left/Right MTF Modifiers** panel.*

Env variable and full documentation: see FIXED_XFALL_ZONE_COLOUR_4D

Fixed decision colour colour box

the colour of the text in the Type column for the Fixed Decision commands in the **Left/Right MTF Modifiers** *panel.*

Env variable and full documentation: see FIXED_DECISION_ZONE_COLOUR_4D

Snippet colour *colour box* *default RGB(155, 140, 150*

*the colour of the text in the Type column for the Snippet command in the Modifiers section of the **Left/Right MTF Modifiers** panel.*

Env variable and full documentation: see SNIPPET_ZONE_COLOUR_4D

Decision colour colour box *default purple*

*the colour of the text in the Type column for the Decision commands in the Modifiers section of the **Left/Right MTF Modifiers** panel.*

Env variable and full documentation: see DECISION_ZONE_COLOUR_4D

Non zone specific colour colour box

the colour of the text in the Type column for the commands that aren't in the Fixed, Cut, Fill or Final sections of the modifiers in the **Left/Right MTF Modifiers** *panel, e.g. Interval commands.*

Env variable and full documentation: see NOP_ZONE_COLOUR_4D

MTF & Boxing > MTF Cut/Fill Colours

Cut colour *colour box* *default red*

*the colour of the text in the Type column for the Cut commands (other than those involving width, height or slope) in the Modifiers section of the **Left/Right MTF Modifiers** panel.*

Env variable and full documentation: see CUT_ZONE_COLOUR_4D

Cut width colour *colour box* *default light red*

the colour of the text in the Type column for the Cut commands involving width, in the Modifiers section of the ***Left/Right MTF Modifiers*** *panel.*

Env variable and full documentation: see CUT_WIDTH_ZONE_COLOUR_4D

Cut height colour *colour box* *default dark red*

the colour of the text in the Type column for the Cut commands involving height, in the Modifiers section of the ***Left/Right MTF Modifiers*** *panel.*

Env variable and full documentation: see CUT_HEIGHT_ZONE_COLOUR_4D

Cut slope colour *colour box* *default dark red*

the colour of the text in the Type column for the Cut commands involving slope, in the Modifiers section of the ***Left/Right MTF Modifiers*** *panel.*

Env variable and full documentation: see CUT_SLOPE_ZONE_COLOUR_4D

Fill colour *colour box* *default green*

the colour of the text in the Type column for the Fill commands (other than those involving width, height or slope) in the Modifiers section of the ***Left/Right MTF Modifiers*** *panel.*

Env variable and full documentation: see FILL_ZONE_COLOUR_4D

Fill width colour *colour box* *default light green*

the colour of the text in the Type column for the Fill commands involving width, in the Modifiers section of the ***Left/Right MTF Modifiers*** *panel.*

Env variable and full documentation: see FILL_WIDTH_ZONE_COLOUR_4D

Fill height colour *colour box* *default dark green*

the colour of the text in the Type column for the Fill commands involving height, in the Modifiers section of the ***Left/Right MTF Modifiers*** *panel.*

Env variable and full documentation: see FILL_HEIGHT_ZONE_COLOUR_4D

Fill slope colour *colour box* *default dark green*

the colour of the text in the Type column for the Fill commands involving slope, in the Modifiers section of the ***Left/Right MTF Modifiers*** *panel.*

Env variable and full documentation: see FILL_SLOPE_ZONE_COLOUR_4D

MTF & Boxing > MTF Final Colours

Final colour *colour box* *default light brown RGB(147, 87, 28)*

the colour of the text in the Type column for the Final commands (other than those involving width or slope) in the Modifiers section of the ***Left/Right MTF Modifiers*** *panel.*

Env variable and full documentation: see FINAL_ZONE_COLOUR_4D

Final width colour *colour box* *default dark brown*

the colour of the text in the Type column for the Final commands involving width, in the Modifiers section of the ***Left/Right MTF Modifiers*** *panel.*

Env variable and full documentation: see FINAL_WIDTH_ZONE_COLOUR_4D

Final cut colour *colour box* *default light brown*

the colour of the text in the Type column for the Final command for Final Cut slope, in the Modifiers *section of the **Left/Right MTF Modifiers** panel.*

Env variable and full documentation: see FINAL_CUT_SLOPE_ZONE_COLOUR_4D

Final fill colour *colour box* *default light brown*

*the colour of the text in the Type column for the Final command for Final Fill slope, in the Modifiers section of the **Left/Right MTF Modifiers** panel.*

Env variable and full documentation: see FINAL_FILL_SLOPE_ZONE_COLOUR_4D

Final no cut colour *colour box* *default light brown*

*the colour of the text in the Type column for the Final command for No Cut slope, in the Modifiers section of the **Left/Right MTF Modifiers** panel.*

Env variable and full documentation: see FINAL_NO_CUT_SLOPE_ZONE_COLOUR_4D

Final no fill colour *colour box* *default light brown*

*the colour of the text in the Type column for the Final command for No Fill slope, in the Modifiers section of the **Left/Right MTF Modifiers** panel.*

Env variable and full documentation: see FINAL_NO_FILL_SLOPE_ZONE_COLOUR_4D

Final no cut/fill colour *colour box* *default light brown*

*the colour of the text in the Type column for the Final commands for No cut/fill, in the Modifiers section of the **Left/Right MTF Modifiers** panel.*

Env variable and full documentation: see FINAL_NO_CUT_FILL_SLOPE_ZONE_COLOUR_4D

MTF & Boxing > Boxing Editor

Boxing wall offset *measure box* *default 0.000001*

when boxing automatically creates a vertical wall, this is the offset distance between the top and bottom of the vertical wall.

Env variable and full documentation: see BOXING_WALL_OFFSET_4D

Rules colour *colour box* *default blue*

the colour of the text in the Type column in the Boxing Rules panel for all Boxing command other than Comment, Decision, Goto and Label.

Env variable and full documentation: see BOXING_RULES_COLOUR_4D

Comment colour *colour box* *default yellow*

background colour for the Comment boxing command grid row in the Boxing Rules panel. The text in the grid row is black.

Env variable and full documentation: see BOXING_COMMENT_COLOUR_4D

Decision/goto colour *colour box* *default mauve*

the colour of the text in the Type column for the Decision and Goto commands in the Boxing Rules panel.

Env variable and full documentation: see BOXING_DECISION_GOTO_COLOUR_4D

Label colour *colour box* *default light red*

the colour of the text in the Type column for the Label command in the Boxing Rules panel.

Env variable and full documentation: see BOXING_LABEL_COLOUR_4D

Region colour colour box

the colour of the text in the row for the Region command in the Boxing Rules panel.

Env variable and full documentation: see BOXING_REGION_COLOUR_4D

Views

Views > General

Prompt on close tick box

if ticked, when a view is closed/deleted, a prompt will ask for a confirmation of deleting/closing.

Env variable and full documentation: see PROMPT_ON_VIEW_CLOSE_4D

Use density drawing tick box

if ticked, for a new project, the default for Use density drawing *in the* **Defaults** *panel is* ***tick***.

If not ticked, for a new project, the default for Use density drawing *in the* **Defaults** *panel is* ***no tick***.

For more information on ***density drawing****, see* Use density drawing tick box.

Note*: This setting is only applicable to the 250M version of* ***12d Model.***

Env variable and full documentation: see USE_DENSITY_CHECKS_4D

Default pan mode tick box default *tick*

if ticked, standard pan is used for pan/pans on view.
If not ticked, pane delta is used for pan/pans on view.

Env variable and full documentation: see PAN_MODE_4D

Zoom origin dynamic tick box default not tick

controls the origin of the dynamic zoom.

If ticked, then the point selected in the view to indicate which view to dynamically zoom (and to be the zoom-in, zoom-out definition point) becomes the point to dynamically zoom about.

Whilst the dynamic zoom is running, another point can be selected to become the new zoom origin.

Env variable and full documentation: see ZOOM_ORIGIN_DYNAMIC_4D

Zoom pan dynamic tick box default not tick

if ticked, ***dynamic pan*** *is the default for the pan options.*

Env variable and full documentation: see ZOOM_PAN_DYNAMIC_4D

Icons on views tick box default *tick*

if ticked then icons instead of text are used for menu items on the views.
If not ticked, text is used for menu items on the views.

Env variable and full documentation: see VIEW_BITMAP_BUTTONS_4D

Show view buttons tick box default *tick*

if ticked then menu items (view buttons) are displayed on the views (as icons or text).
If not ticked, menu items (view buttons) on not displayed on the views.

Note*: displaying view buttons as icons or text is controlled by* ***Icons on views.***

Env variable and full documentation: see VIEW_BUTTONS_4D

Data tool tips tick box default *tick*

if ticked then data tips can be displayed when the cursor moves over vertices in a plan view. The ***D*** *snap is then operational and toggles the data tips on and off in a 12d session.*

If not ticked, data tips will not be displayed and ***D*** *snap has no effect.*

Env variable and full documentation: see DATA_TIPS_4D

Default view colour colour box default *black*

the default background colour for views. After a view is created, its background colour can then be modified and the new colour is saved for that view.

Env variable and full documentation: see DEFAULT_VIEW_COLOUR_4D

Preview view name text box default *blank*

if non blank, the name of the view whose image is dumped on exiting the project. The image is used as the project preview.
If blank then the last active view is used.

Env variable and full documentation: see PREVIEW_VIEW_4D

Views > OpenGL

View backing store choice box off, on

if ***on****, a backing store is used.*
if ***off****, a backing store is not used.*

For Window Vista/Windows 7, the default is on. Otherwise the default is 0.

Env variable and full documentation: see OPENGL_VIEW_BACKING_STORE_4D

Use offset tick box default tick

if ticked, the coordinates are localised for OpenGL calls. This is to work around problems with some graphics cards that can't handle large coordinates.

Env variable and full documentation: see OPENGL_OFFSET_4D

Cache tins tick box *default not ticked*

if ticked, tins are cached in the graphics card memory for potential speed ups. More memory in the graphics card allows more caching and usually more performance gains.

Env variable and full documentation: see OPENGL_CACHE_TINS_4D

Cache textures and rasters tick box *default ticked*

if ticked, rasters and textures are cached in the graphics card memory for potential speed ups. More memory in the graphics card allows more caching and usually more performance gains.

Env variable and full documentation: see OPENGL_CACHE_4D

Use mipmaps for rasters tick box *default tick*

if ticked, the graphics card down samples for rasters when the image is further away.

Env variable and full documentation: see OPENGL_MIPMAP_4D

Use mipmaps for billboards tick box

if ticked, the graphics card down samples for billboards when the image is further away.

Env variable and full documentation: see OPENGL_MIPMAP_BILLBOARDS_4D

Use mipmaps for plan images tick box default ticked

if ticked, the graphics card down samples plan images when draped onto a tin.

Env variable and full documentation: see OPENGL_MIPMAP_PLAN_IMAGES_4D

Use mipmaps for projector images tick box default ticked *if ticked, the graphics card down samples projector images (a projector is for the "hidden" perspective image data object of a super string).*

Env variable and full documentation: see OPENGL_MIPMAP_PROJECTOR_IMAGES_4D

GUI

GUI > General

Width of edit fields (number of characters) text box *default 10*

if non blank, the number of characters to make the width of an Edit Box on a panel.

Env variable and full documentation: see EDIT_BOX_WIDTH_IN_CHARACTERS_4D

Show help buttons tick box

if ticked, Help buttons are added to panels.

Env variable and full documentation: see HELP_BUTTONS_4D

Use lists for popups tick box

if ticked, the popup lists are scrolling lists.
If not ticked, the popups turn into walk-right menus when too long.

Env variable and full documentation: see LIST_POPUPS_4D

Use trees for linestyles and symbols tick box default tick

if ticked, the linestyle and symbol pop-up lists are in scrolling boxes so they don't run over the bottom of the screen. The list of linestyles/symbols is displayed in a tree structure with the Groups as the nodes of the tree.
If not ticked, the linestyle and symbols lists are one long list with each Group being an item on the list. The list may get too long to fit on the screen.

Env variable and full documentation: see USE_NEW_LINESTYLE_LIST_BOX_4D

Warp cursor hide tick box default *not tick*

controls whether the cursor is hidden before moving - only needed on some computers.

If ticked, don't hide the cursor before moving it.
If not ticked, hides the cursor before moving it.

Env variable and full documentation: see WARP_CURSOR_HIDE_4D

Multi-line text edits tick box default not ticked

if ticked, a Text Edit Box is used instead of the Input Box and more than one line of text can be typed.
Env variable and full documentation: see MULTI_LINE_TEXT_4D

Colour widgets with errors tick box default ticked

*if ticked, when a panel field fails to validate, the panel field is filled with the colour given in the **Colour for failed widgets**.*
Env variable and full documentation: see USE_VALIDATION_COLOURS_4D

Colour for failed widgets colour box

if not blank, the colour to fill the panel field with when there is a validation error for the field. The value is either a colour name, a colour number or RGB(x,y,z).

*This is only used if **Colour widgets with errors** is ticked (or at least not ticked since the default is ticked).*
If blank, the default colour is RGB(255,72,72),

Env variable and full documentation: see VALIDATION_FAIL_COLOUR_4D

Xtra menu file file box default *xtramenu.4d*

if non blank, the full path name of the 12d Solutions supplied file of definitions for some extra menus.

Env variable and full documentation: see EXTRA_OPTIONS_4D

User options file file box default *usermenu.4d*

if non blank, the full path name of the user supplied file of definitions for user defined menus.

Env variable and full documentation: see USER_OPTIONS_4D

Toolbars file file box default *toolbars.4d*

if non blank, the full path name of the toolbar definitions and names.

Env variable and full documentation: see TOOLBARS_4D

Delete walkrights distance positive integer default *32*

the distance in pixels that is used to collapse the cascade of walk-right menus when the cursor moves that distance past the end of the last walk-right menu.

Env variable and full documentation: see AUTO_DELETE_WALKRIGHTS_4D

Drag reset distance positive integer default *50*

the distance in pixels to move the cursor to reset the picking rejection list for a directional pick. If blank then the value 50 is used.

Env variable and full documentation: see AUTO_RESET_SELECT_DRAG_TOLERANCE_4D

Reset distance positive integer default *5*

the distance in pixels to move the cursor to reset the picking rejection list for a non-directional pick. If blank then the value 5 is used.

Env variable and full documentation: see AUTO_RESET_TOLERANCE_4D

Default table width positive integer default *5*

if non blank, gives the number of characters displayed in tables such as decisionals and mtf editor.

Env variable and full documentation: see DEFAULT_TABLE_WIDTH_4D

Linestyle/symbol box width number box

if non blank, the pixel width of the linestyle and symbol scrolling pop-ups.
If blank, the default is 256.
Env variable and full documentation: see LINESTYLE_BOX_WIDTH_4D

Linestyle/symbol box height number box

if non blank, the pixel height of the linestyle and symbol scrolling pop-ups.
If blank, the default is 512.
Env variable and full documentation: see LINESTYLE_BOX_HEIGHT_4D

GUI > OnScreen keyboard

On screen keyboard mode choice box *No keyboard*
Dockable keyboard
Full screen keyboard

Default No keyboard

if ***No keyboard,*** *then no onscreen keyboard comes up when you double click in a panel field.*

if ***Dockable keyboard****, when you double click in a panel field that takes typed input, then a dockable onscreen keyboard come up for the user to type the data for the panel field into.*

*if **Full screen keyboard**, when you double click in a panel field that takes typed input, then a full screen onscreen keyboard come up for the user to type the data for the panel field into.*

Env variable and full documentation: see ONSCREEN_KEYBOARD_4D

On screen keyboard layout choice box *Full keyboard*
Numeric keyboard

Default Full keyboard

*if **Full keyboard,** then when there is an onscreen keyboard, it comes up with a full keyboard.*

*if **Numeric keyboard**, then when there is an onscreen keyboard, it comes up with just a numeric keyboard.*

Env variable and full documentation: see ONSCREEN_KEYBOARD_LAYOUT_4D

On screen keyboard font size number box

if non blank, the pixel size of the font for the onscreen keyboard.

If blank, it defaults to the normal system font size.

Env variable and full documentation: see ONSCREEN_KEYBOARD_FONT_SIZE_4D

GUI > Special characters

Large diameter symbol *integer (base 10)* default *216*

The integer (base 10) value of the character to use as the large diameter symbol.

Env variable and full documentation: see DIAMETER_LARGE_CHARACTER_4D

Cubed symbol *integer (base 10)* default *179*

The integer (base 10) value of the character to use as the cubed symbol.

Env variable and full documentation: see CUBED_CHARACTER_4D

Small diameter symbol *integer (base 10)* default *248*

The integer (base 10) value of the character to use as the small diameter symbol.

Env variable and full documentation: see DIAMETER_SMALL_CHARACTER_4D

Middle dot *integer (base 10)* default *183*

The integer (base 10) value of the character to use as the middle dot symbol.

Env variable and full documentation: see MIDDLE_DOT_CHARACTER_4D

Squared symbol *integer (base 10)* default *178*

The integer (base 10) value of the character to use as the squared symbol.

Env variable and full documentation: see SQUARE_CHARACTER_4D

Degrees character *integer (base 10)* default *176*

The integer (base 10) value of the character to use as the degrees character.

Env variable and full documentation: see DEGREES_CHARACTER_4D

GUI > Chain Editor

Prompt for on deleting a chain command *tick box*

if ticked,

Env variable and full documentation: see ASK_ON_CHAIN_COMMAND_DELETE_4D

GUI > Split drafting message

Split long drafting messages *tick box*

if ticked, for CAD Dimensions, Leaders and Tables, long messages being written to the screen message area are broken up into two shorter lines with the first one being displayed. Both the shorter lines have a (m)ore on them to get to the other line.

Env variable and full documentation: see DRAFTING_SPLIT_MESSAGE_4D

Warnings and Error Reporting

Double confirm delete tick box default not tick

if ticked, the user is asked twice to confirm for deletes and cleans.
If not ticked, the user is asked once to confirm for deletes and cleans.

Env variable and full documentation: see DOUBLE_CONFIRM_DELETE_4D

Crash log level choice box

MiniDumpNormal
MiniDumpWithDataSegs
MiniDumpWithFullMemory
MiniDumpWithHandleData
MiniDumpFilterMemory
MiniDumpScanMemory
MiniDumpWithUnloadedModules
MiniDumpWithIndirectlyReferencedMemory
MiniDumpFiltureModulePaths
MiniDumpWithProcessThreadData
MiniDumpWithPrivateReadWriteMemory
MiniDumpWithoutOptionalData

Controls the amount of information written out if ***12d Model*** *crashes.*

Environment variable MINI_DUMP_LEVEL_4D

GIS

ArcSDE DLL Path file box

the full path name to the ArcSDE DLL. This is only required when using the ArcSDE Server.

Env variable and full documentation: see GIS_ARCSDE_PATH_4D

Oracle DLL Path file box

the full path name to the Oracle DLL. This is only required when using the Oracle Server.

Env variable and full documentation: see GIS_ORACLE_PATH_4D

External Apps

External Apps > ADAC

Adac version text

the ADAC Schema version, e.g. 4.1.0

Env variable and full documentation: see ADAC_VERSION_4D

External Apps > AutoCAD

Autocad template folder folder box

if non blank, the full path name of the folder holding Autocad template files.

Env variable and full documentation: see ACAD_SEEDFILES_4D

Acad.pat file box

If non blank, the full path name of the AutoCAD patterns file.

Env variable and full documentation: see AUTOCAD_PATTERNS_4D

Trimesh output mode choice box Do not write, Faces, Polyface mesh

Env variable and full documentation: see TRIMESH_TO_DWG_MODE_4D

External Apps > CivilCAD

CivilCAD folder folder box

if non blank, the full path name of the folder holding CivilCAD files.

Env variable and full documentation: see CIVILCAD_PATH_4D

External Apps > Microstation

Microstation seed folder folder box

if non blank, the full path name of the folder of Microstation seed files.

Env variable and full documentation: see MS_SEEDFILES_4D

External Apps > TP Stakeout

TP Stakeout folder folder box

if non blank, the full path name of the folder of TP Stakeout files.

Env variable and full documentation: see TP_STAKEOUT_PATH_4D

External Apps > VPath

Vpath Windows folder folder box

if non blank, the full path name of the folder for the Queensland Main Road Windows program Vpath - for vehicle turning paths.

Env variable and full documentation: see WINDOWS_VEHICLE_PATH_4D

External Apps > Winter

Winter data folder folder box

if non blank, the full path name of the *Folder containing the Winter data of N-values for Australia.*

Env variable and full documentation: see WINTER_DATA_4D

Winter faster interpolations tick box *default tick*

if ticked, re-reading the Winter data is avoided and this speeds up the calculation for the Winter interpolations.

Env variable and full documentation: see WINTER_USE_NEW_METHOD_4D

Variables

the Variables grid allows for the entry of 12d environment variables that have not been especially included in the Edit Environment Variables *panel.*

Variable **Value**

*The variable tab consists of a table for specifying environment variable names (**Variable**) and setting their values (**Value**).*

This is mainly used for setting environment variables that are not already in the **Edit Environment Variables** *panel.*

Go to the next section 7.6.4 Env Configuration or return to 7.6 Management.

7.6.4 Env Configuration

Position of menu: **Project =>Management =>Env configuration**

The default ***12d Model*** system uses the 12d Solutions supplied ***set_ups*** and ***library*** folders in ***Program files\12d\12d Model\11.0*** and the user supplied folders ***user*** and ***user_lib*** in *c:\12d\11.00*.

There are ***12d Model*** environment variables in ***env.4d*** that can modify the location of each of these folders and most set up files, dongle to be used and the workspace file for new projects.

For most users this provides enough customisation however for users requiring totally different ***user*** and ***user_lib*** folders for different clients will find the above system too restricting.

For example, one project may be for a Main Roads Department which requires its own mapping files, linestyle files, ppf files etc. Another project is for a Local Authority who has totally different requirements.

Environment Configurations can

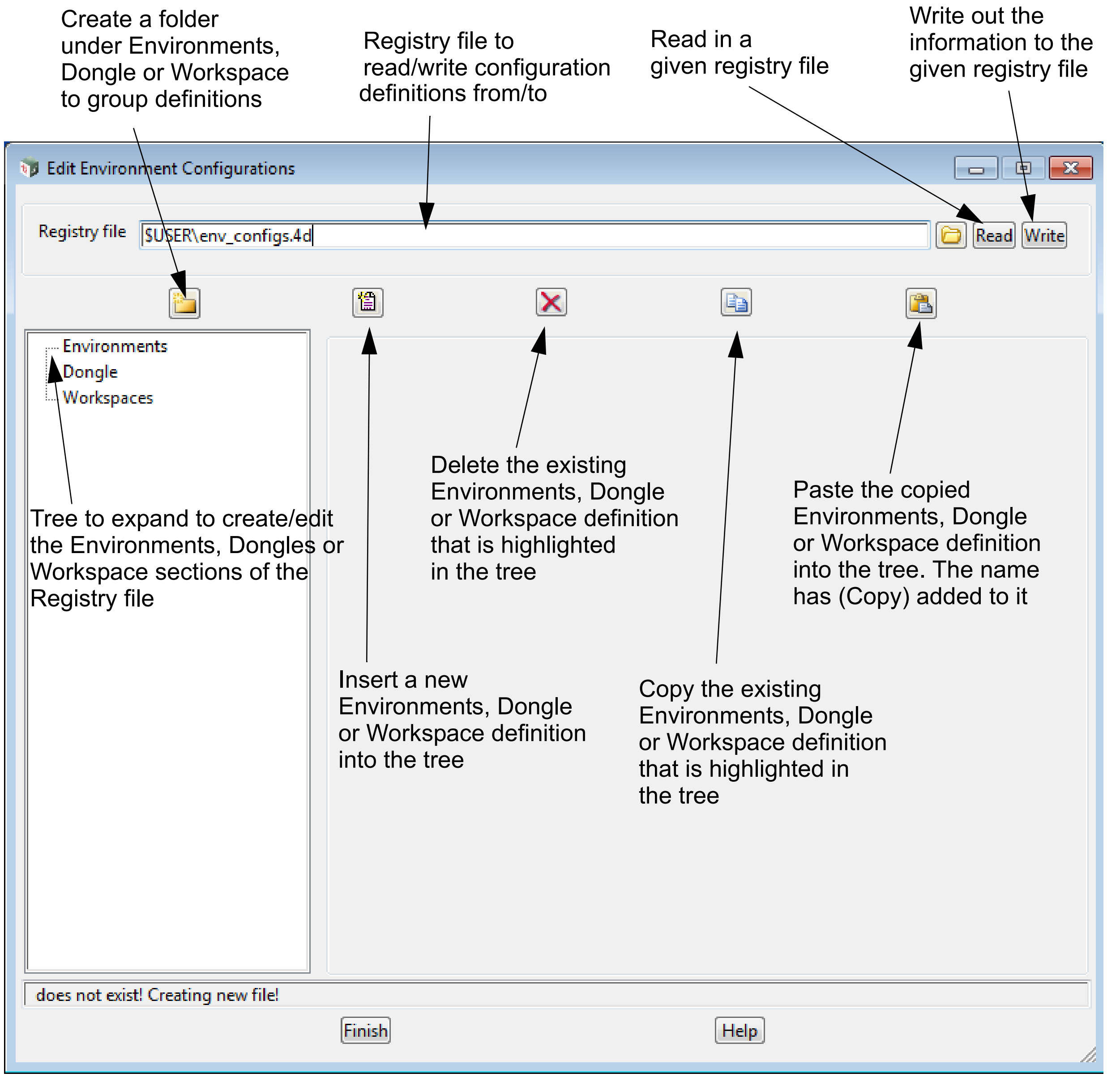
(a) define the environment variable file ***env.4d*** to be used. Hence if different ***env.4d*** files have been set up for different clients, the environment configuration file quickly sets up that binding for a project.

(b) define environment variables that over writes the values set up in the ***env.4d*** file being used for the project. Often a whole new env file is not required but just the replacement of some of the environment variables. For example, just override the location of the folders *user* and *user_lib* from those in the ***env.4d*** file being used.

(c) define the computer with the network dongle to be used with the project. If used, this will override any settings in the env.4d file and values in Environments.

(d) define the workspace file to be used for new projects. If used, this will override any settings in the env.4d file or values in *Environments*.

Selecting **Env configuration** displays the **Edit Environment Configuration** panel which defines the different Environment configurations, Dongle and Workspace settings.



Buttons at Bottom grid

Write button

write out a Registry file

Read button

read in the registry file given in the Registry file field.

Each of the sections Environments, Dongles and Workspaces will now be described in greater detail.

For the section *Registry file*, go to Creating/Editing a Registry File
Environments Creating/Editing an Environments Set Up
Dongles Creating/Editing a Dongles Set Up
Workspaces Creating/Editing a Workspaces Set Up

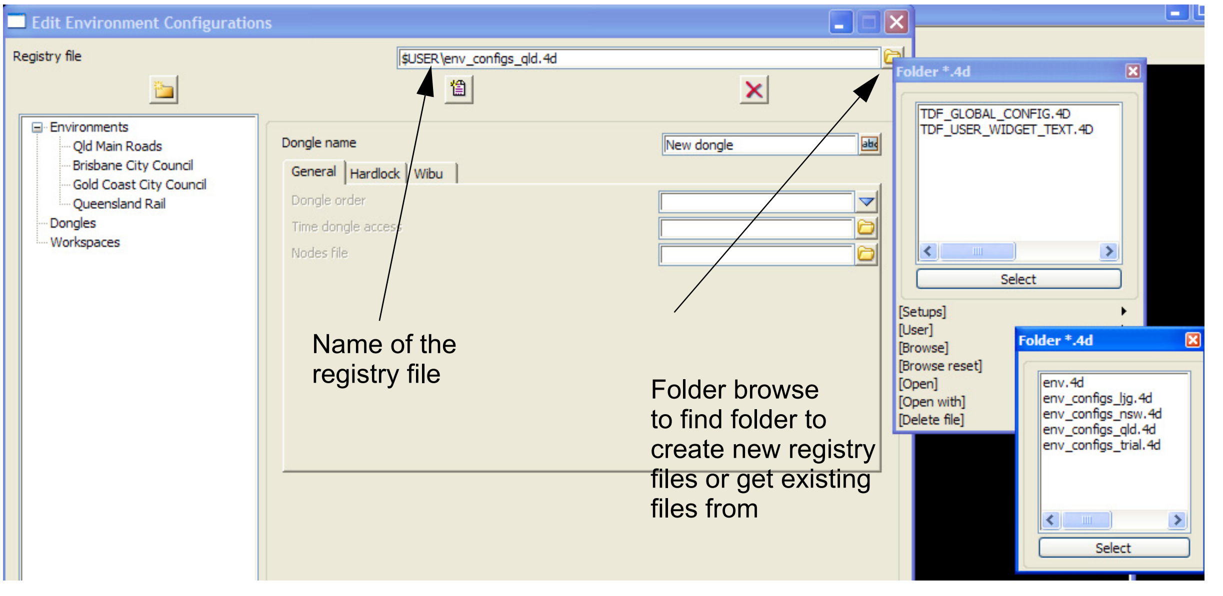
How To Use the Edit Environment Configurations Panel

Creating/Editing a Registry File

When the **Edit Environment Configurations** panel is opened, if no ***env_configs.4d*** file exists in the ***User*** folder for that project, a new blank ***env_configs.4d*** file is automatically created.

To create a ***Registry file*** of a different name, navigate to the folder to write the registry file to, type the registry file name into the ***Registry file box*** and hit <enter>. All the created information will be written away to this file when the **Write** icon is clicked on the **Edit Environment Configurations** panel.

The edit and modify a ***new Registry file***, type in the path name or use the navigator on the ***Registry file box*** and select the required registry file. Clicking on the **Read** icon reads in the information from the selected registry file.



Each of the sections Environments, Dongles and Workspaces will now be described in greater detail.

For the section *Environments*, go to [Creating/Editing an Environments Set Up]

Dongles [Creating/Editing a Dongles Set Up]

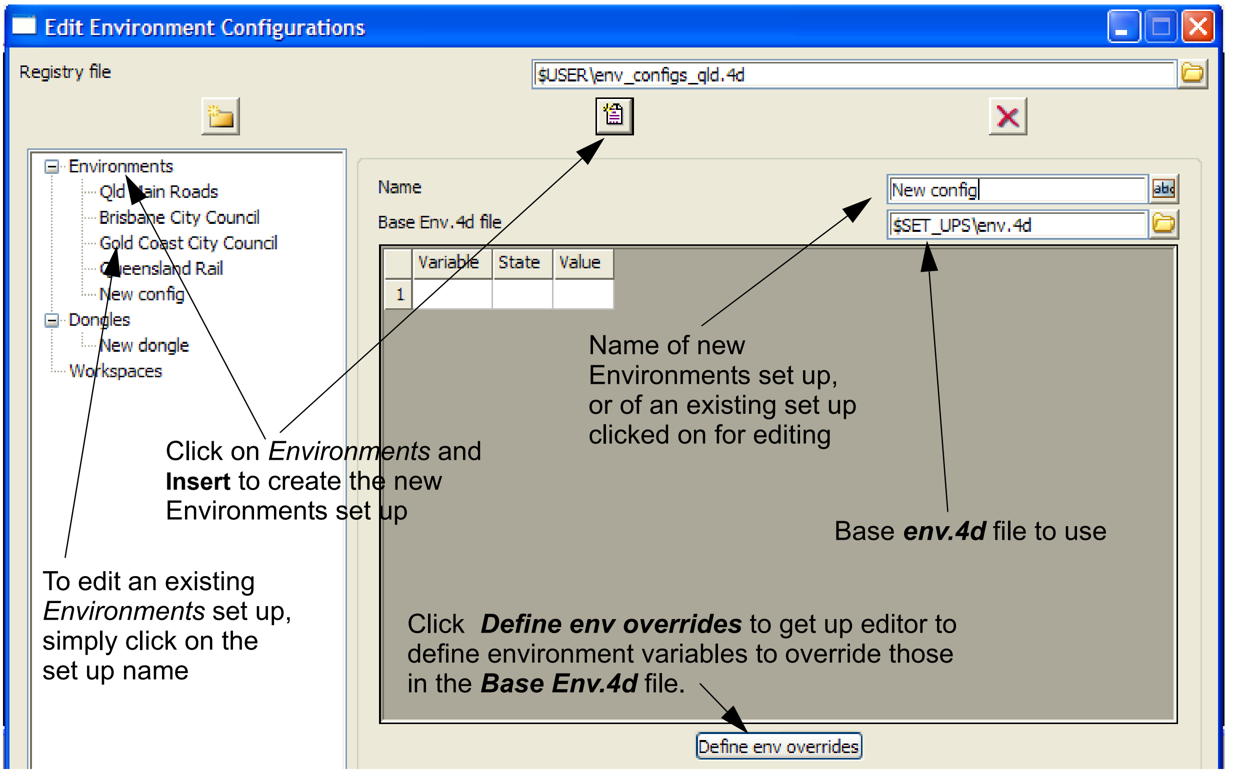
Workspaces [Creating/Editing a Workspaces Set Up]

Creating/Editing an Environments Set Up

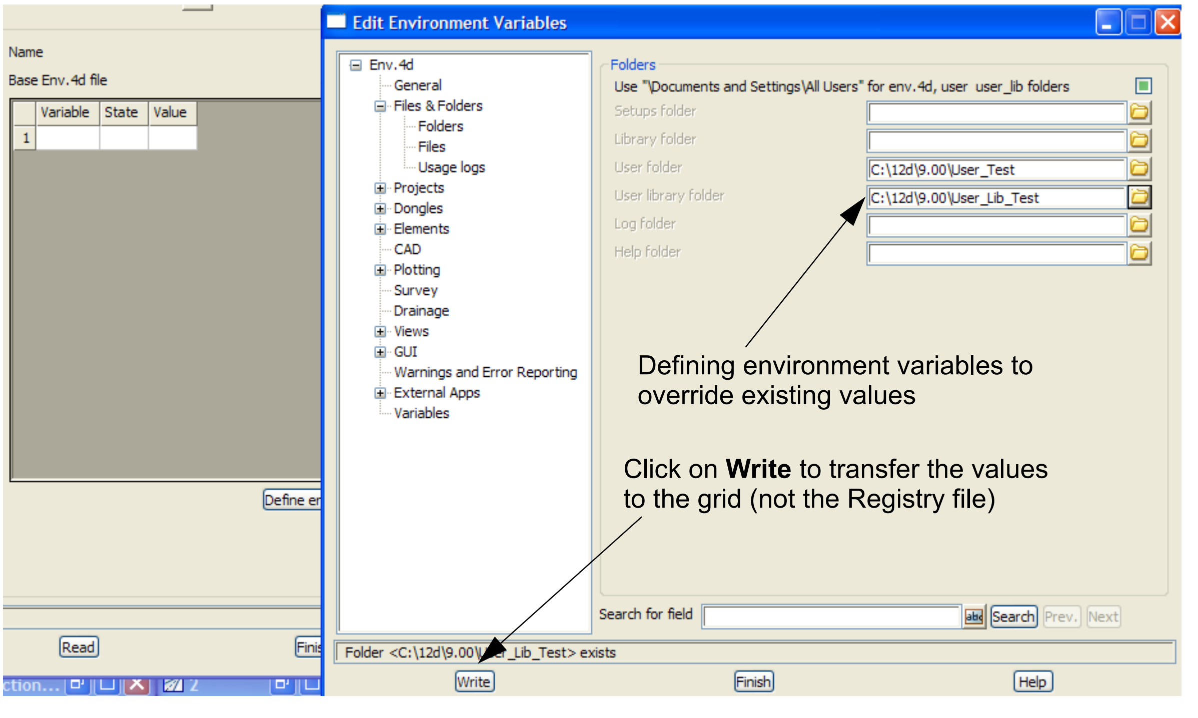
To create a new **Environments** set up, click on the *Environments* node and then on the **Insert** button.

To edit an existing **Environments** set up, simply click on the ***name*** of the set ups to be edited in the *Environments* section

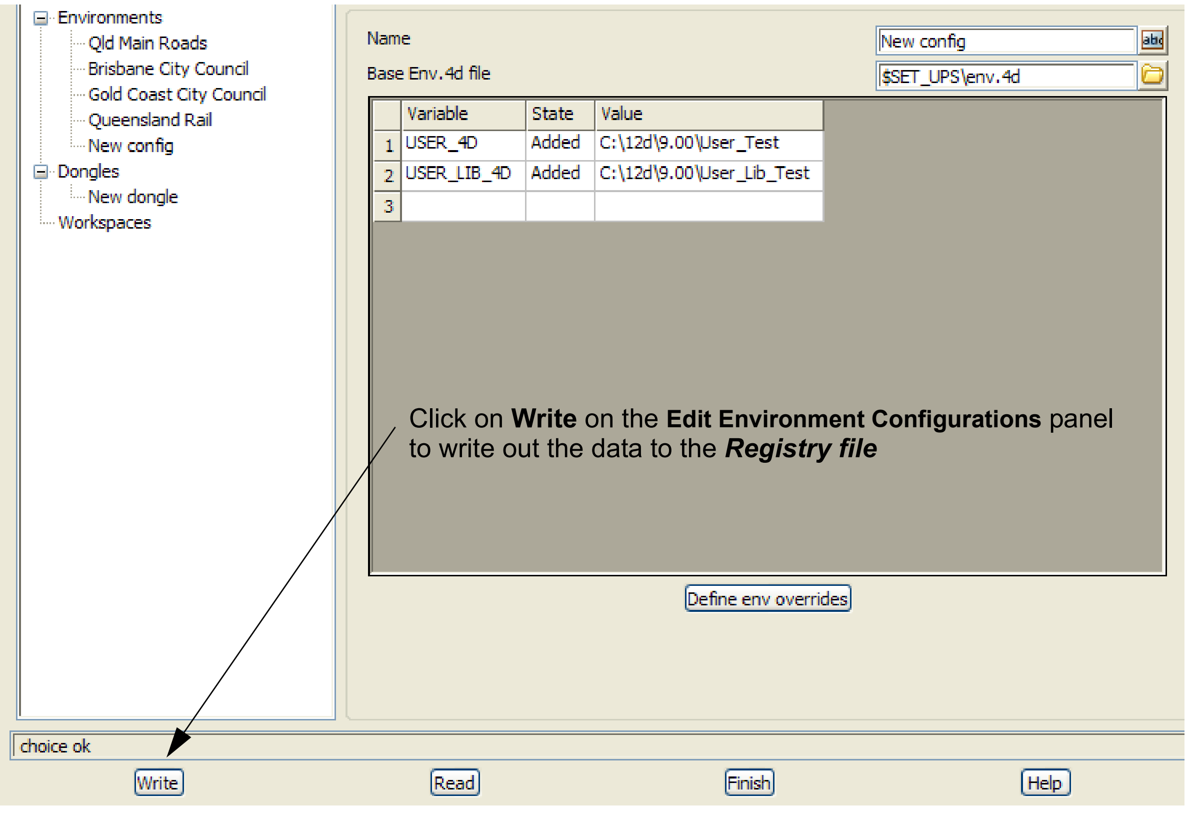
If the ***Base Env.4d file*** field is ***blank***, then the default ***env.4d*** file is used for the project.
If the ***Base Env.4d file*** field is not blank, then it gives the (path) name of the file to use as the ***env.4d*** file.



Any environment variable can be overridden by typing is name and value into the grid, or by bringing up an environment variable editor by clicking on **Define env overrides** button and fill in the required values in the **Edit Environment Variables** panel that is brought up.



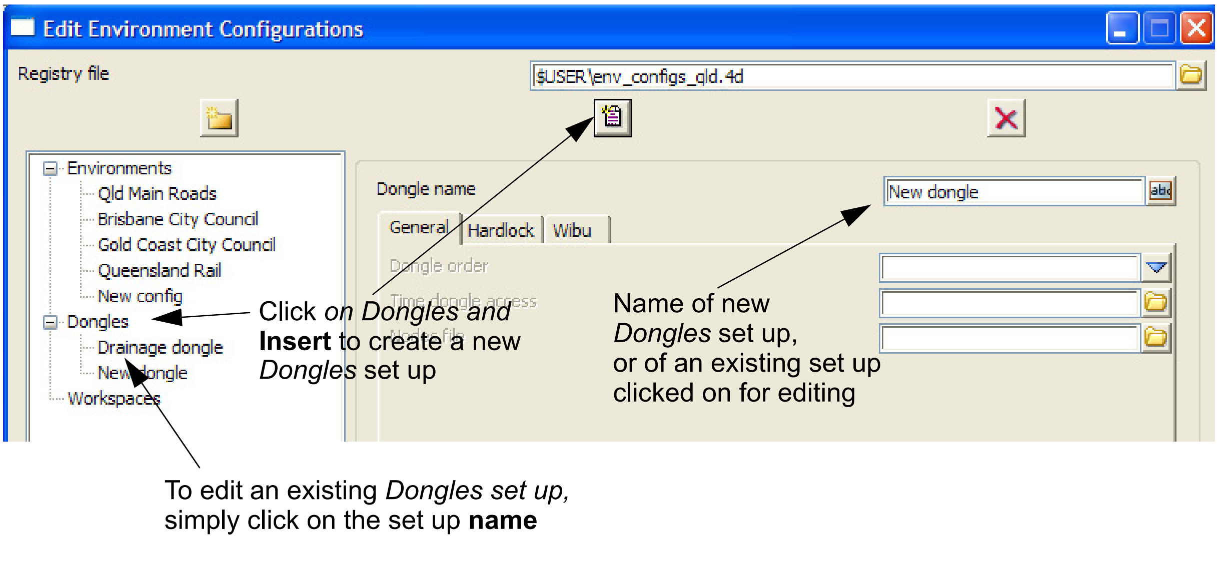
Clicking on **Write** in the **Edit Environment Configuration** panel writes out all the data for the sections *Environments*, *Dongles* and *Workspaces*, to the registry file.



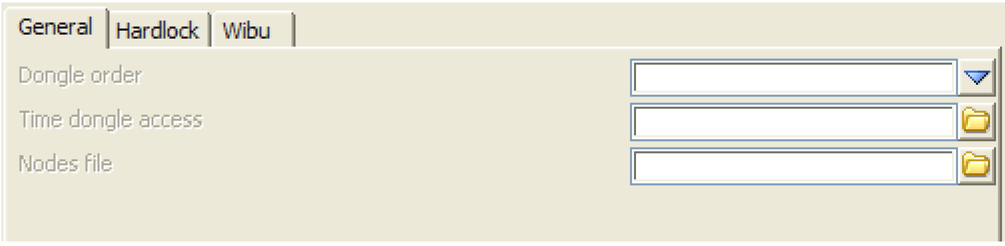
Creating/Editing a Dongles Set Up

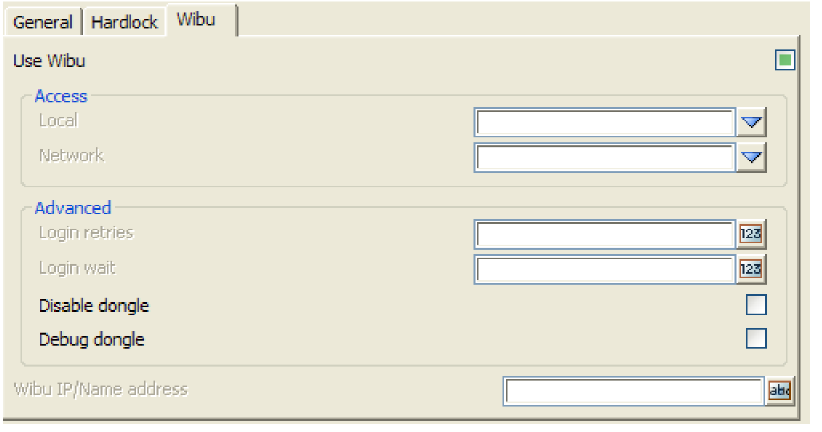
To create a new **Dongle** set up, click on the *Dongles* node and then on the **Insert** button.

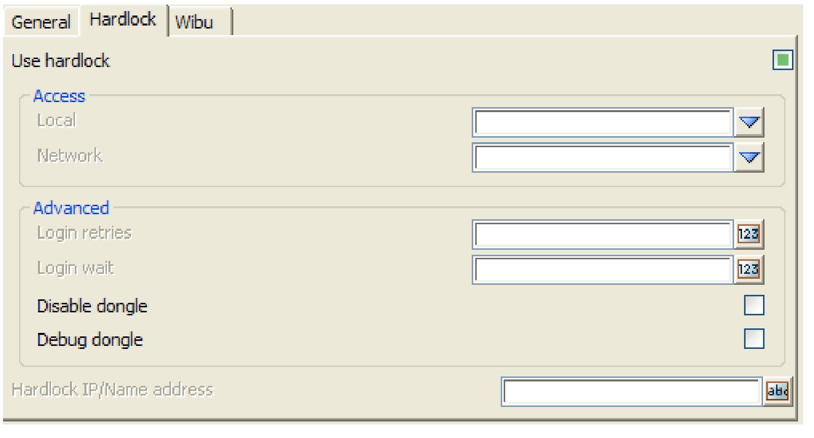
To edit an existing **Dongle** set up, simply click on the name of the **Dongle** set up to be edited.



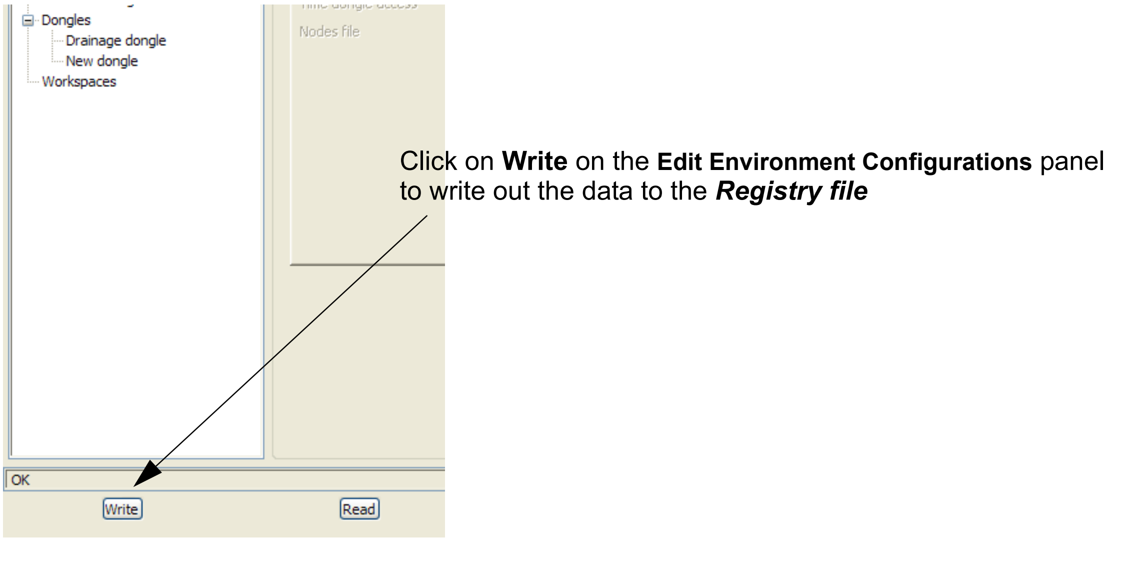
Any environment variable values defined in the *General*, *Hardlock* and *Wibu* tabs will override those in the environment file.







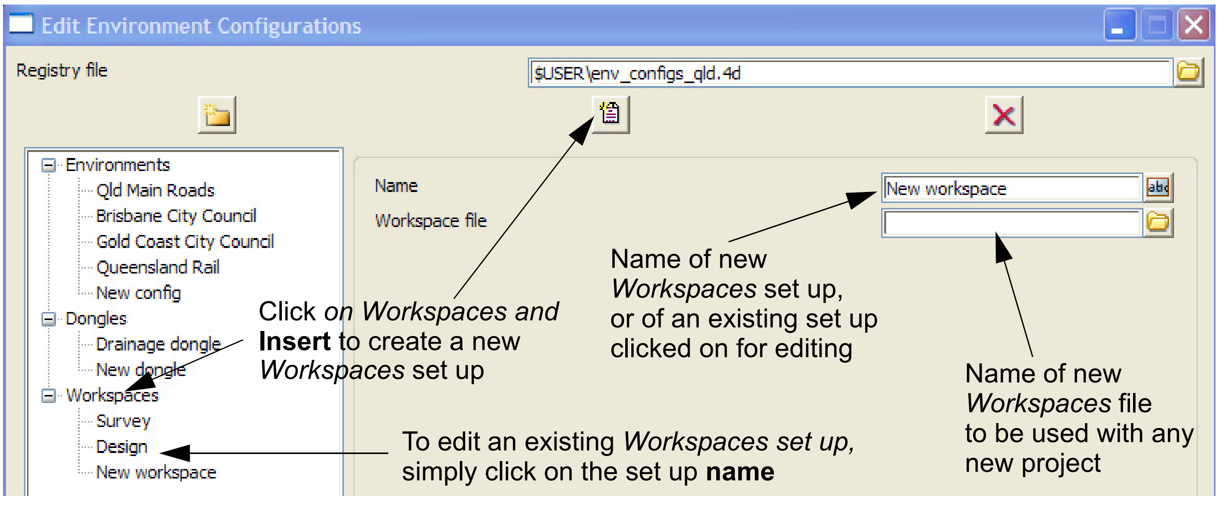
Clicking on **Write** in the **Edit Environment Configuration** panel writes out all the data for the sections *Environments*, *Dongles* and *Workspaces*, to the registry file.



Creating/Editing a Workspaces Set Up

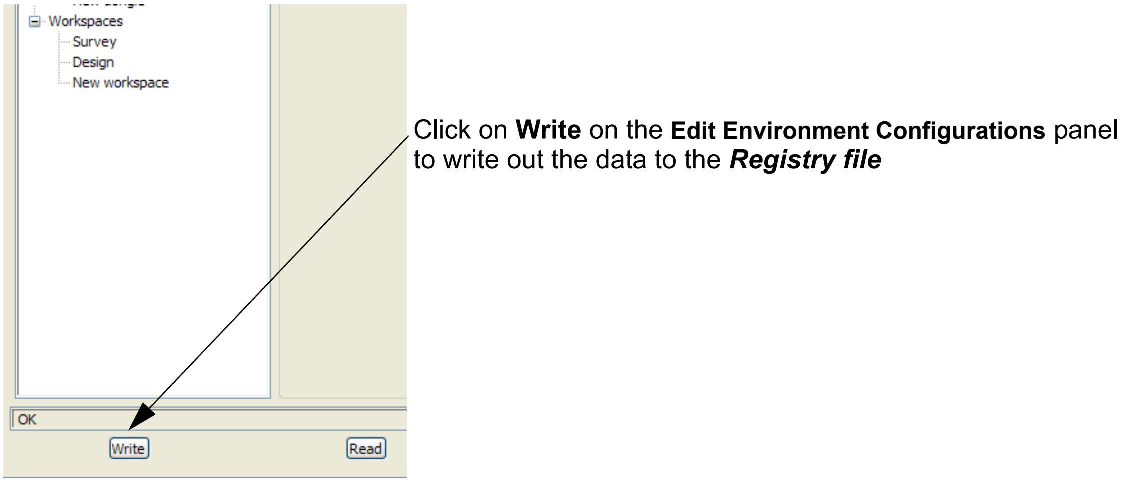
To create a new **Workspaces** set up, click on the *Workspaces* node and then on the **Insert** button.

To edit an existing **Workspaces** set up, simply click on the name of the **workspaces** set up to be edited.



The full path name of the Workspaces file to be used when creating new projects is recorded in the **Workspace file** panel field.

Clicking on **Write** in the **Edit Environment Configuration** panel writes out all the data for the sections *Environments*, *Dongles* and *Workspaces*, to the registry file.

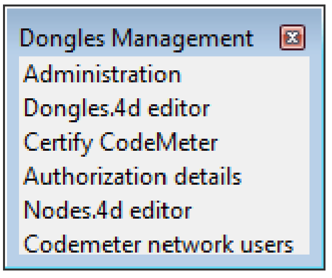


Go to the next section 7.6.6 Projections or return to 7.6 Management.

7.6.5 Dongles

Position of menu: **Project =>Management =>Dongles**

The **Dongles** walk-right menu contains options to discover all the dongles on your subnet, edit the dongles.4d file, certify CodeMeters and show authorization information.



run the dongle administration options
edit the dongles.4d file
certify CodeMeter containers
show Authorization details
edit the nodes.4d file
report on hte users of CodeMeter network dongles

See

- 7.6.5.2 Dongles Administration
- 7.6.5.3 Dongles.4d Editor
- 7.6.5.4 Certify CodeMeter
- 7.6.5.5 Authorization Details
- 7.6.5.6 Nodes.4d Editor
- 7.6.5.7 CodeMeter Network User

Also for information on the **Authorization Error** panel that comes up if there is an error authorising ***12d Model***, see 7.6.5.5.1 Authorization Error.

For more information about what is needed for authorizing ***12d Model***, see 7.6.5.1 12d Dongles, Dongles.4d, Nodes.4d, and Authorizing

7.6.5.1 12d Dongles, Dongles.4d, Nodes.4d, and Authorizing

To control the licensing of ***12d Model*** and specify what modules are used for each ***12d Model*** license, ***12d Model*** uses a combination of a **physical hardware lock** and a **nodes.4d file**.

The **physical hardware lock** is a **Wibu dongle** or a **CodeMeter Container**.

A **Wibu dongle** is either a Wibu USB Standalone (local) dongle (green colour), or a Wibu USB Network dongle (blue colour) that controls a given number of licenses. Each Wibu dongle has a unique ***12d Model*** dongle number. *See* 7.6.5.1.1 Wibu Dongles

A **CodeMeter Container** comes in a variety of physical shapes (USB, SD card etc) but unlike the Wibu dongle, the CodeMeter Container can contain one or more *virtual* ***12d Model*** dongle number. Each virtual ***12d Model*** dongle number is either a Standalone number or a Network number, and in Network case, it also have a given number of licenses. See 7.6.5.1.2 Wibu CodeMeter Containers.

Hence the ***12d Model* dongle number** (from a physical Wibu dongle or within a CodeMeter Container) is either

a Standalone (Local) number

or

a Network number with a given number of licenses

For brevity, the ***12d Model* dongle number** is also referred to as the **12d dongle number** or just the **12d dongle**. But remember that the **12d dongle** may not be a physical item - it may be just one of many 12d dongles stored in a CodeMeter Container.

So the term **12d dongle** will be used to mean:

(a) a physical Wibu dongle (either stand alone or network). This has a label with the 12d dongle number written on it.

(b) a virtual 12d dongle inside a CodeMeter Container. The 12d dongle numbers are obtained by running a CodeMeter Administration tool.

The **dongles.4d** file lists the computers to search for Wibu dongles and CodeMeter Containers to find 12d dongle numbers. It also defines the order to search for both computers and dongle types. See 7.6.5.3 Dongles.4d Editor.

The **nodes.4d** file is supplied by your ***12d Model*** Reseller and contains a list of ***12d Model*** dongle numbers that are valid for you, and for each ***12d Model*** dongle number in the list, the valid modules for that ***12d Model*** dongle number. The nodes.4d file is an XML file and there is an option to examine, modify and report on it. See 7.6.5.3 Dongles.4d Editor.

When ***12d Model*** tries to create a new project, or open an existing project, ***12d Model*** uses

1. the dongles.4d file to obtain a usable ***12d Model*** dongle number from the specified Wibu dongles or CodeMeter Containers

and if a ***12d Model*** dongle number is found,

2. a matching valid entry in the nodes.4d file for that ***12d Model*** dongle number.

If a match is found then the ***12d Model*** project is opened.

Otherwise an **Authorization Error** panel is displayed.

See

7.6.5.1.1 Wibu Dongles
7.6.5.1.2 Wibu CodeMeter Containers

7.6.5.1.1 Wibu Dongles

A **Wibu dongle** can be either a Wibu USB Standalone (local) dongle (a translucent green colour), or a Wibu USB Network dongle (a solid colour) that controls a given number of licenses. Each Wibu dongle has a unique ***12d Model*** dongle number.

The ***12d Model*** dongle numbers for Wibu Standalone (Local) dongles start with 572d.
The ***12d Model*** dongle numbers for Wibu Network dongles start with e151.

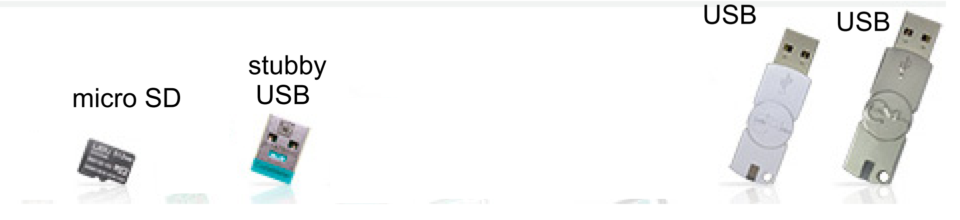
.

Continue to the next section [7.6.5.1.2 Wibu CodeMeter Containers]() or return to [7.6.5.1 12d Dongles, Dongles.4d, Nodes.4d, and Authorizing]() or [7.6.5 Dongles]().

7.6.5.1.2 Wibu CodeMeter Containers

In V11 a new type of physical hardware lock from Wibu, referred to as a **Wibu CodeMeter Container**, has been introduced. We will refer to the *Wibu CodeMeter Container* as a **CodeMeter Container** or a **CodeMeter**.

The **CodeMeters** are similar to the earlier Wibu standalone and network dongles except that they come in a wider variety of hardware shapes that will be more suitable for portable and tablet computers.



Unlike the Wibu dongle, the CodeMeter Container can contain one or more *virtual* ***12d Model*** dongle numbers.

Each virtual ***12d Model*** number may be either a Standalone number or a Network number, and in the Network case, it will also have a given number of licenses.

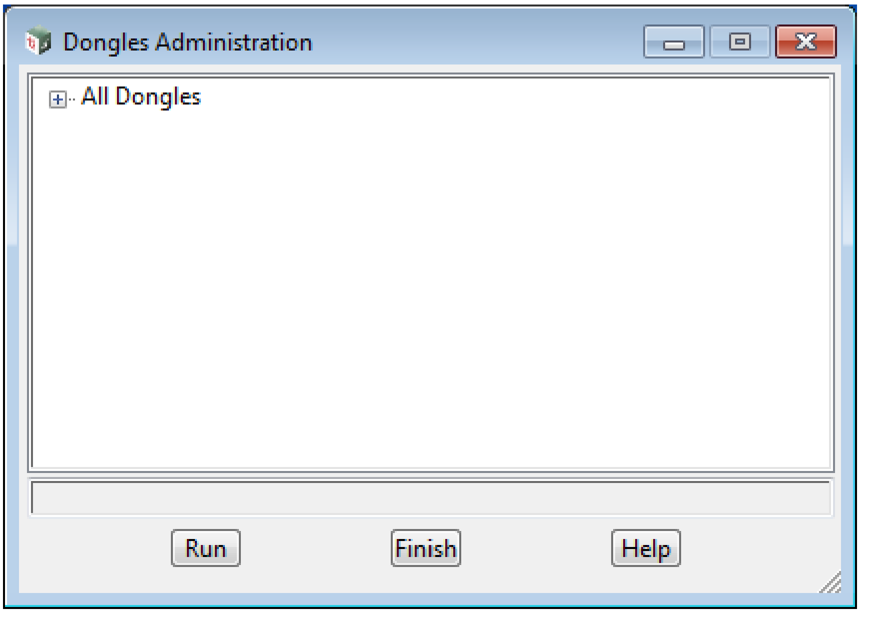
The ***12d Model*** Local (Standalone) dongle numbers in a CodeMeter Container start with 5c2d.
The ***12d Model*** Network dongle numbers in a CodeMeter Container start with ec51.

Return to [7.6.5.1 12d Dongles, Dongles.4d, Nodes.4d, and Authorizing]() or [7.6.5 Dongles]().

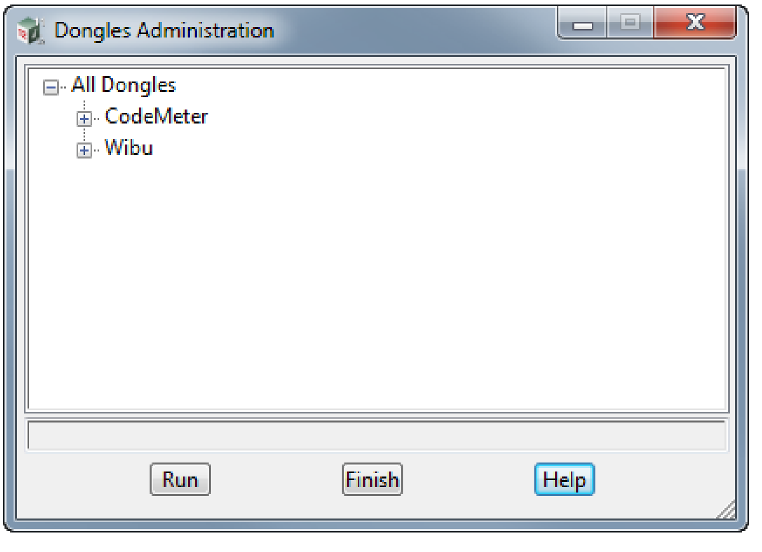
7.6.5.2 Dongles Administration

Position of option on menu: **Project =>Management =>Dongles =>Administration**
Position of option on menu: **Help =>Dongles**

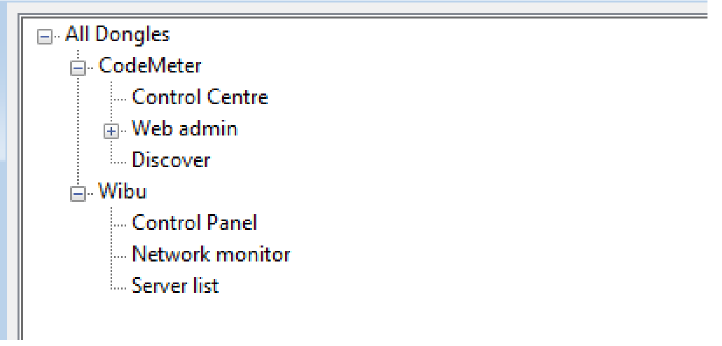
Selecting **Administration** brings up the **Dongles Administration** panel which displays information on ***CodeMeter Containers*** and ***Wibu Dongles*** as nodes in a tree structure.



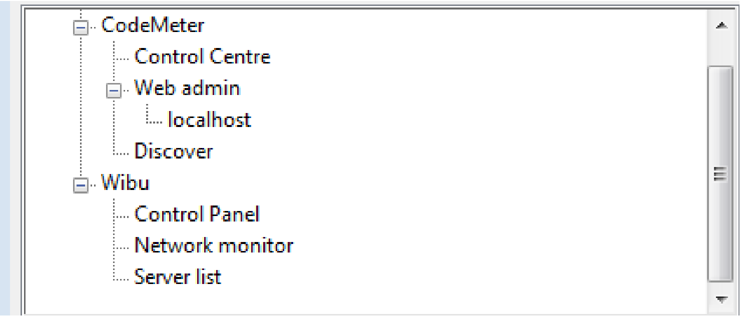
Clicking on the **+** in front of **All Dongles** will list the ***physical dongles*** that could be on your computer. That is, **CodeMeter Containers** and **Wibu Dongles**.



The **CodeMeter** and **Wibu** nodes can also be expanded by clicking on their **+**.



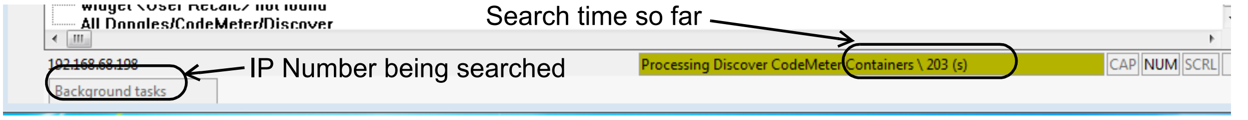
Clicking **+** on **CodeMeter >Web admin** will show the computer you are on as **localhost**.



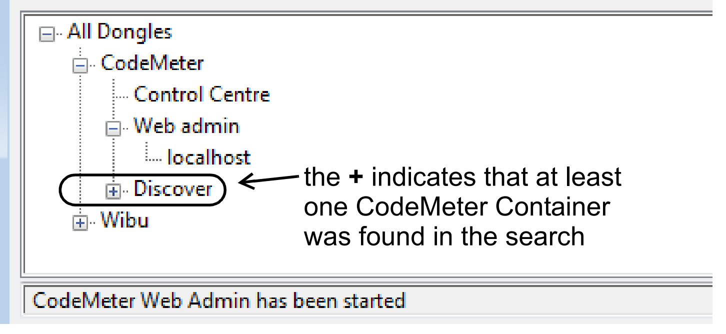
1. Clicking on **CodeMeter >Web admin >localhost** will run the CodeMeter Administration program and display information about the computer you are on in **Home** tab of the **CodeMeter WebAdmin** panel.

2. Clicking on **CodeMeter >Discover** will search the 255 sub nets looking for any *CodeMeter* Containers on each subnet.

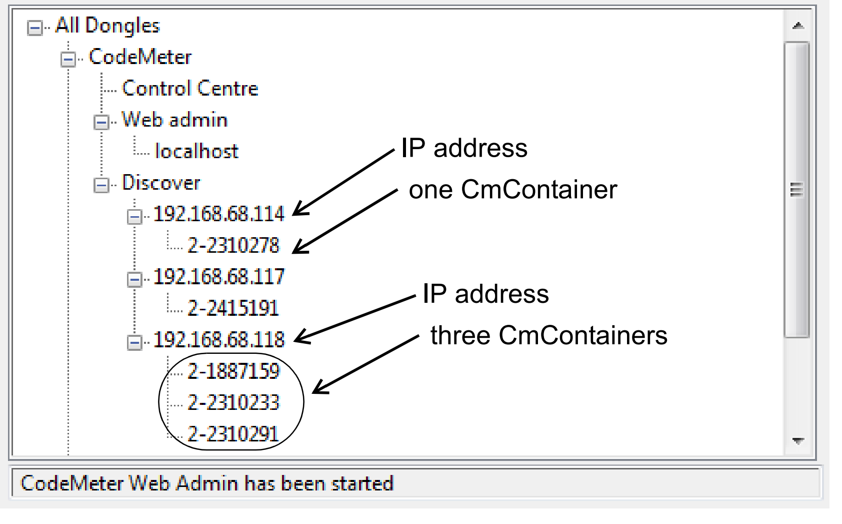
   The **Discovery** may take up to five minutes and you can see the progress of the **Discovery** as it lists the IP addresses being searched in the left had corner of the screen message area, and the right hand side displays how many seconds the **Discovery** has been running.



When **Discovery** has finished searching the 255 sub nets, the IP address of any computer with a CodeMeter Container on it is listed under the **Discover** node. So a **+** will appear in front of **Discover** if at least *one* CodeMeter Container has been found.



Expanding **Discover** will display the IP Addresses with *CodeMeter* Containers (called **CmContainers**) on them. And expanding an IP Address will list all the CmContainers attached to that IP.



Double clicking on a **CmContainer** will bring up the **CodeMeter WebAdmin** panel with the **Content >Licenses** tab on display.

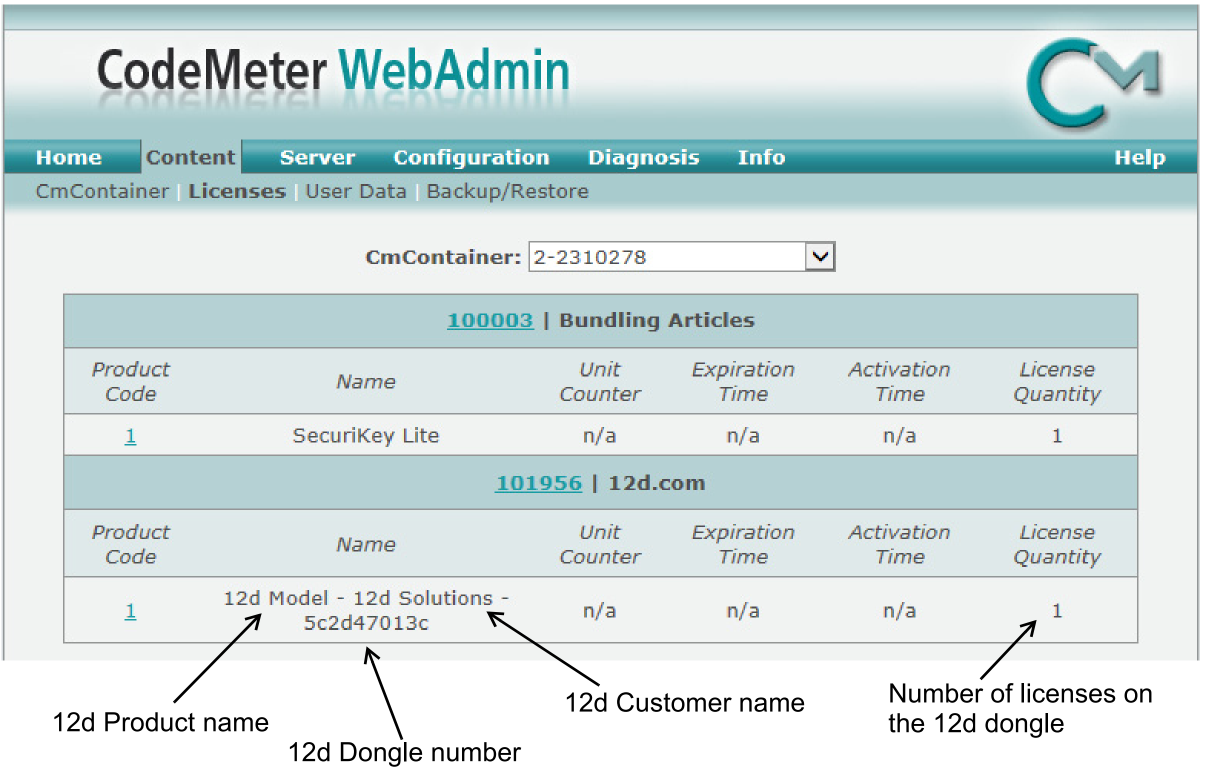
The **Content >Licenses** tab will show the **Products** it contains and for each **12d Product**, it will show the

(a) 12d Product name

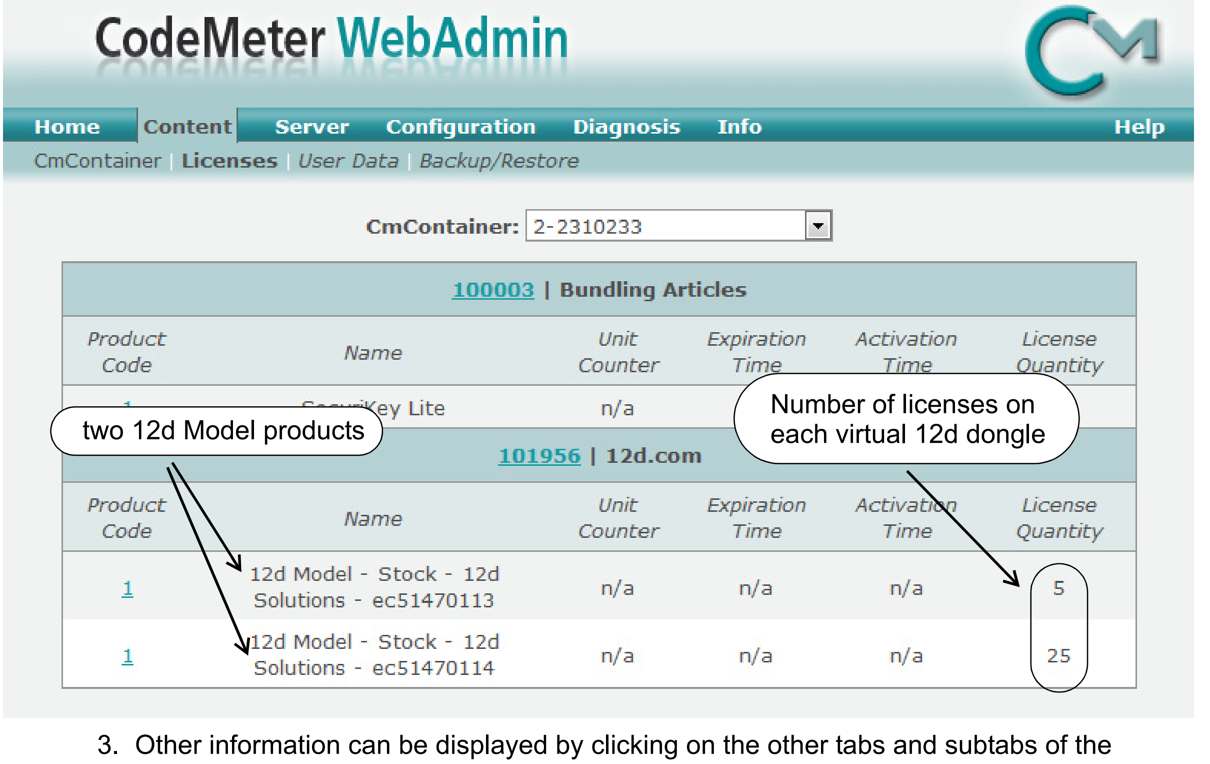
(b) 12d Customer name

(c) ***12d Model*** dongle number (a virtual dongle inside the physical CodeMeter Container)

(d) **number of licenses** for that ***12d Model*** **dongle**.



Note that there may be more than one virtual **12d Dongle** inside the one physical CodeMeter Container, as well as products from other software vendors. For example,



3. Other information can be displayed by clicking on the other tabs and subtabs of the **CodeMeter WebAdmin** panel.

Continue to the next section 7.6.5.3 Dongles.4d Editor or return to 7.6.5 Dongles.

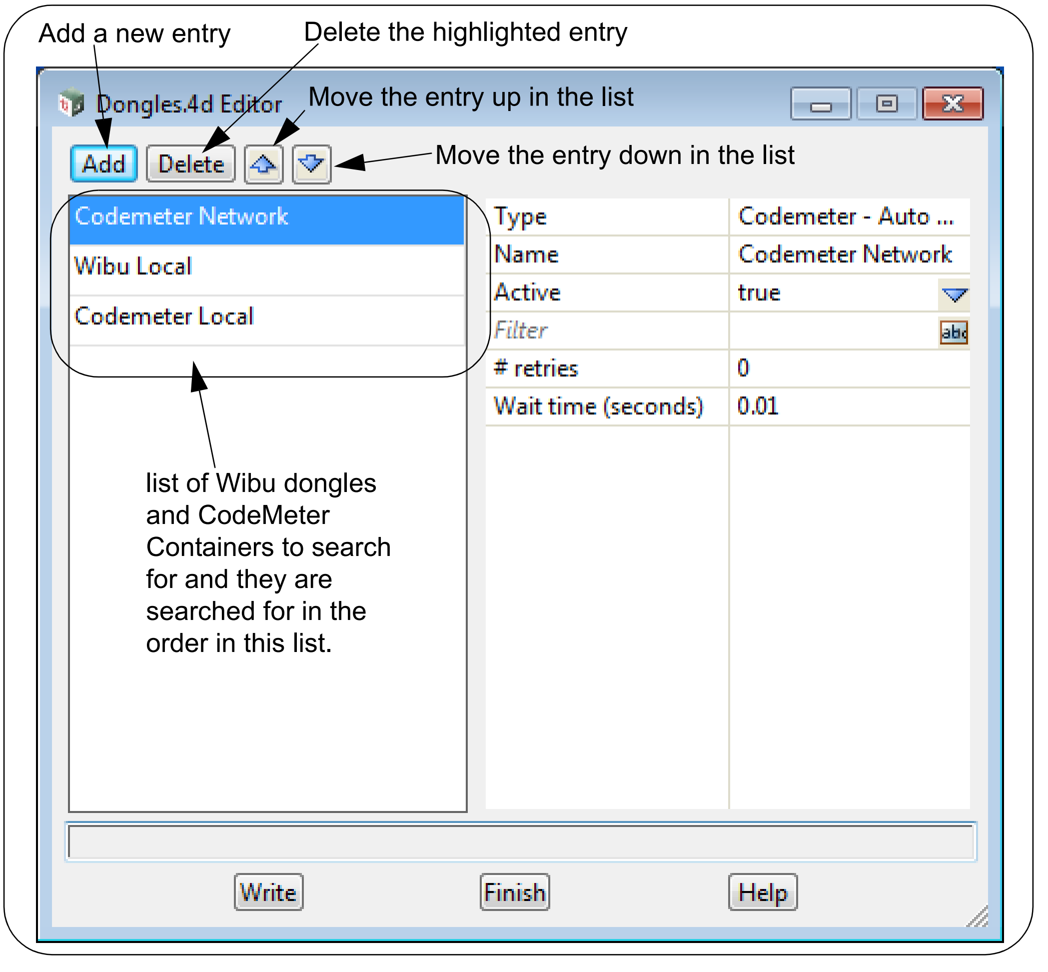
7.6.5.3 Dongles.4d Editor

Position of option on menu: **Project =>Management =>Dongles => Dongles.4d editor**

The **dongles.4d** file lists the computers to search for Wibu dongles (7.6.5.1.1 Wibu Dongles) and CodeMeter Containers (7.6.5.1.2 Wibu CodeMeter Containers) to find 12d dongle numbers. It also defines the order to search for both computers and dongle types.

The **Dongles Editor** option is for editing the ***dongles.4d*** file.

Selecting **Dongles.4d editor** brings up the **Dongles.4d Editor** panel.

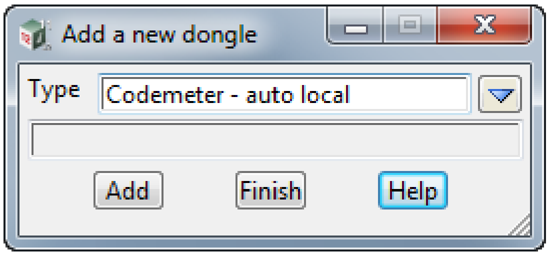


The **Dongles.4d Editor** panel displays the list of Wibu dongles (local and/or network) and CodeMeter containers (local and/or network) to search for to find 12d dongles to potentially use when opening a project. The order of searching is the order that the items occur in this list.

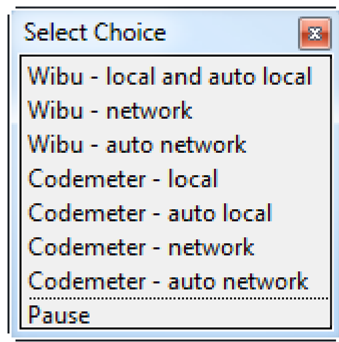
The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Add	button		

*this brings up the **Add a New Dongle** panel*



Type choice box ,



Wibu - local and auto local*: search the local computer for a Wibu Standalone dongle. If there is more than one Standalone Wibu dongle on the computer then the first one found is used. So there should only be one Standalone Wibu dongle on the computer.*

	Type	Wibu - Local and Auto Local
user defined name →	Name	Wibu Auto Local
use entry if Active is true →	Active	true
number of times to retry →	# retries	0
time to wait between retries →	Wait time (seconds)	0.01

Wibu - network*: a server name (computer name) is given as part of the information for this Type. The computer on the network with this name is searched for a Wibu Network dongle. There should only be one Wibu network dongle on the given computer*

	Type	Wibu - Network
user defined name →	Name	New dongle
use entry if Active is true →	Active	true
name of the computer →	Server name	
number of times to retry →	# retries	0
time to wait between retries →	Wait time (seconds)	0.01

Wibu - auto network*: search all computers on the network for Wibu Network dongles. Note there should only be one Wibu network dongle on the one computer*

	Type	Wibu - Auto Network
user defined name →	Name	New dongle
use entry if Active is true →	Active	true
not currently used →	Filter	
number of times to retry →	# retries	0
time to wait between retries →	Wait time (seconds)	0.01

CodeMeter - local: *a Dongle UNC is given as part of the information for this Type. The local computer is searched for a CodeMeter Container of the given Dongle UNC.*

.

	Type	Codemeter - Local
user defined name	Name	New dongle
use entry if Active is true	Active	true
CodeMeter dongle UNC	Dongle UNC	
number of times to retry	# retries	0
time to wait between retries	Wait time (seconds)	0.01

CodeMeter - auto local: *search the local computer for all CodeMeter Containers that have 12d Standalone dongles in them.*

	Type	Codemeter - Auto Local
user defined name	Name	New dongle
use entry if Active is true	Active	true
not currently used	Filter	
number of times to retry	# retries	0
time to wait between retries	Wait time (seconds)	0.01

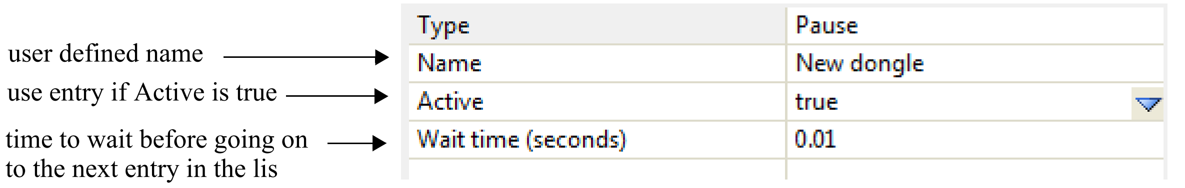
CodeMeter - network: *a computer name is given as part of the information for this Type. The computer on the network with this name is searched for CodeMeter Containers with 12d network dongle numbers in them.*

	Type	Codemeter - Network
user defined name	Name	New dongle
use entry if Active is true	Active	true
name of the computer	Server name	
number of times to retry	# retries	0
time to wait between retries	Wait time (seconds)	0.01

CodeMeter - auto network: *search all computers on the network for CodeMeter Containers that have 12d network dongles in them.*

	Type	Codemeter - Auto Network
user defined name	Name	Codemeter Auto Network
use entry if Active is true	Active	true
not currently used	Filter	
number of times to retry	# retries	0
time to wait between retries	Wait time (seconds)	0.01

Pause: *wait for a given amount of time before going on to the next entry in the list*



Delete button

delete the highlighted entry.

Up Arrow button

move the highlighted entry up in the list.

Down Arrow button

move the highlighted entry down in the list.

Write button

write out the dongles.4d file. For more information on the **Write** *button, go to the section* 39.2.6 Writing Out Setup Files *in the Appendix* 39 Setting Up & Configuring 12d.

For information on the **dongles.4d** file installed with ***12d Model***, see 7.6.5.3.1 Dongles.4d Shipped with 12d Model.

Important Note

If the *dongles.4d* file being used by ***12d Model*** has no entries relating to network dongles, then any existing deprecated WIBU_DONGLE_4D and WIBU_IPADDR environment variables for WIBU network dongles are still used.

Continue to the next section 7.6.5.4 Certify CodeMeter or return to 7.6.5 Dongles.

7.6.5.3.1 Dongles.4d Shipped with 12d Model

When ***12d Model*** is installed, a **dongles.4d** file is included in **set_ups.**

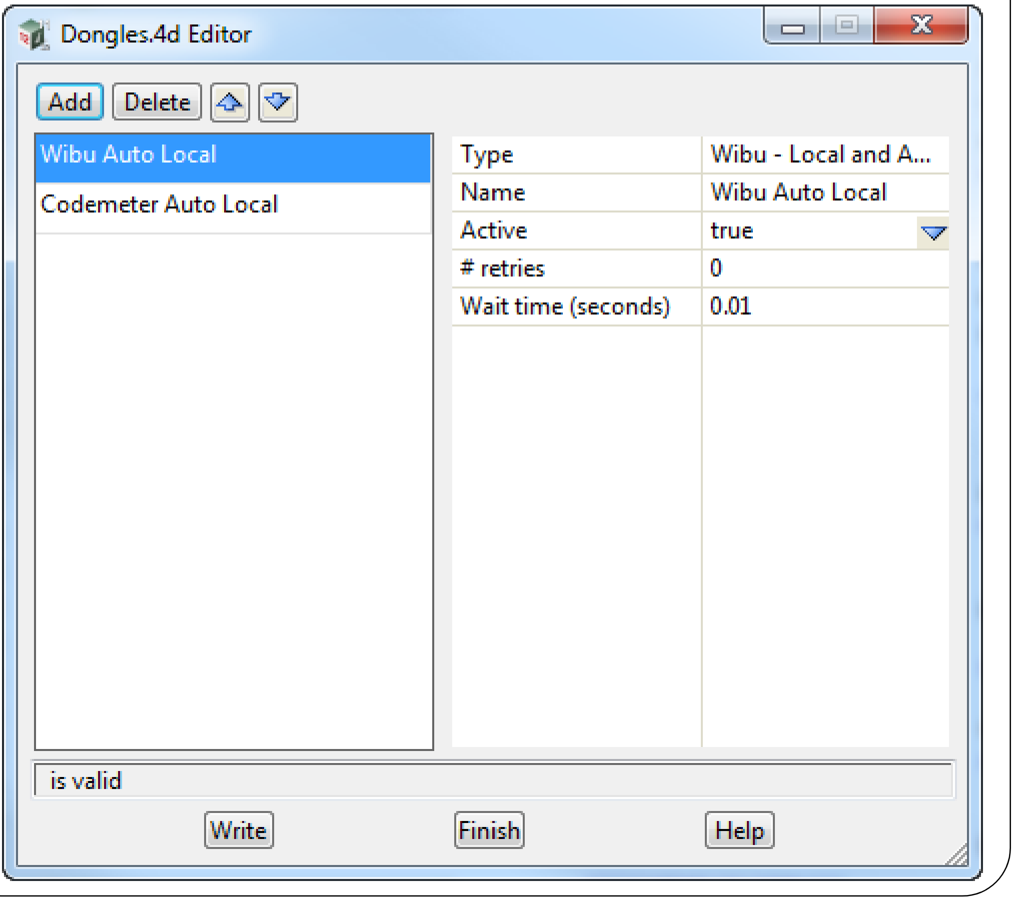
This shipped dongles.4d file only contains entries to as ***12d Model*** to look for any local Wibu dongles and any local Codemeter Containers.

Important Note for Network Dongle Users

There are no entries for network dongles in the shipped dongles.4d file.

So if you are using Wibu network dongles or Codemeter Containers with network 12d dongles in them, then you need to have **your own** dongles.4d file that includes entries for network dongles. Your dongles.4d file is searched for as a Set Up file, or by using the environment variable DONGLES_4D.

Installed dongles.4d File



Return to 7.6.5.3 Dongles.4d Editor or 7.6.5 Dongles.

7.6.5.4 Certify CodeMeter

Position of option on menu: **Project =>Management =>Dongles => Certify CodeMeter**

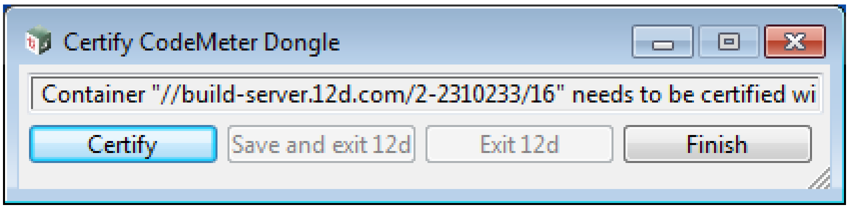
The Wibu **CodeMeter** Container (also called CodeMeter Container or CodeMeter dongle or just CodeMeter) use the Certified Time capability of the dongle and the dongle must communicate over the Internet with a certified Wibu Time Server every two months to check that there is no problem with the ***CodeMeter Container*** that your ***12d Model*** license is coming from.

For Certification to occur, a computer able to access the dongle must also have access to the Internet to run the Certification.

One month before the end of the two month period, ***12d Model*** will bring up the **Certify CodeMeter Dongle** panel which gives the number of days, minutes and seconds remaining until Certification is required.

The **Certify CodeMeter Dongle** panel can also be brought up by the option

Project =>Management =>Dongles =>Certify Codemeter



Clicking on the button **Certify** will attempt to certify the CodeMeter.

Important Note

There is an environment variable AUTO_CERTIFY_DONGLE_4D and if it is set on then when the warning period is active for the CodeMeter Container (dongle) being used by ***12d Model***, ***12d Model*** will automatically attempt to certify the dongle and if successful, no intervention by the user is required. See AUTO_CERTIFY_DONGLE_4D.

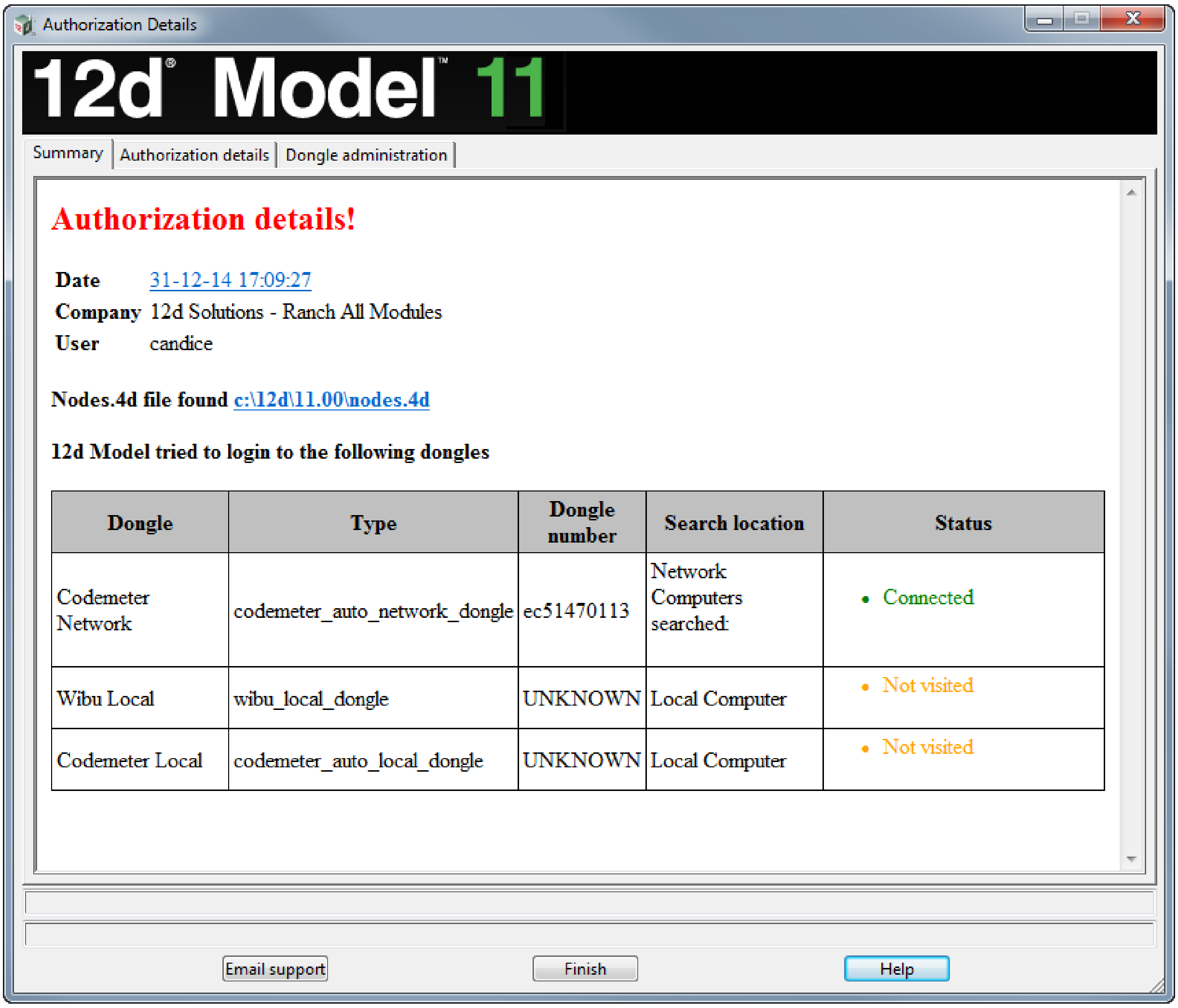
Continue to the next section 7.6.5.5 Authorization Details or return to 7.6.5 Dongles.

7.6.5.5 Authorization Details

Position of option on menu: **Project =>Management =>Dongles => Authorization details**

The **Authorization Details** option shows similar information to the **Authorization Error** panel except in this case there is a match between a found ***12d Model*** dongle and an entry in the nodes.4d file.

Selecting **Authorization details** brings up the **Authorization Details** panel.



The fields and buttons used in this panel have the following functions.

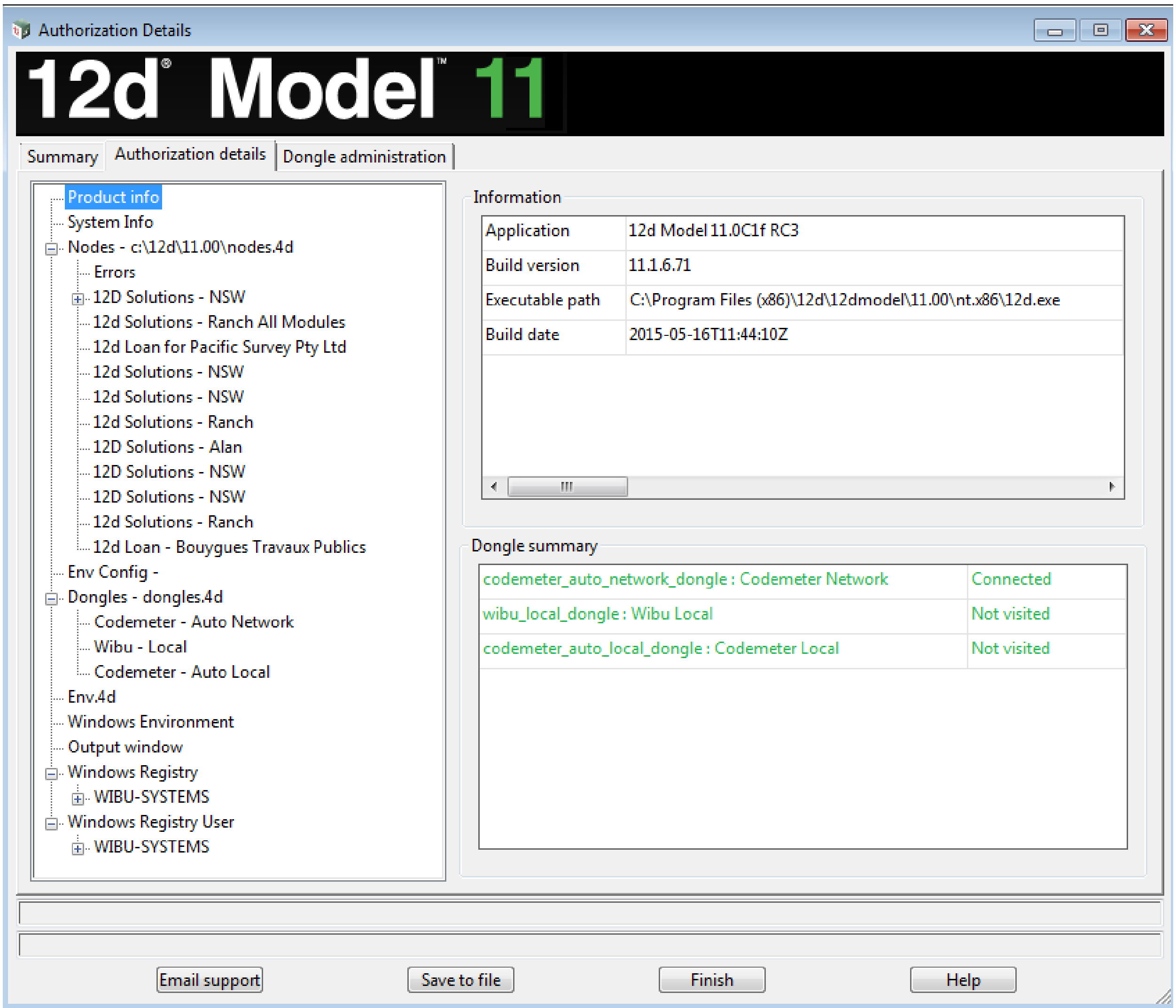
Field Description Type Defaults Pop-Up

Summary tab

*this summarises information from the **Authorization details** tab.*

It shows the 12d Model dongle being used.

Authorization details tab



shows details about the

version of 12d Model

nodes.4d file used

dongles.4d file used

env.4d file used

Windows Environment

What is in the Output Window

Windows Registry

Windows Registry User

Dongle administration tab

same as running the ***Project =>Management =>Dongles =>Administration*** *option. See* 7.6.5.2 Dongles Administration.

Buttons at Bottom

Email support button

creates an email with an attachment of a zipped up copy of all the information on the ***Authorization details*** *tab.*

If there is an error in the authorising of 1***12d Model*** , ***12d Model*** will not open a project but instead bring up a similar panel called the **Authorization Error** panel. See 7.6.5.5.1 Authorization Error.

Continue to the next section 7.6.5.6 Nodes.4d Editor or return to 7.6.5 Dongles.

7.6.5.5.1 Authorization Error

This panel only appears if ***12d Model*** will not authorise for the selected project.

Before any selected ***12d Model*** project will open, a valid authorization is required.

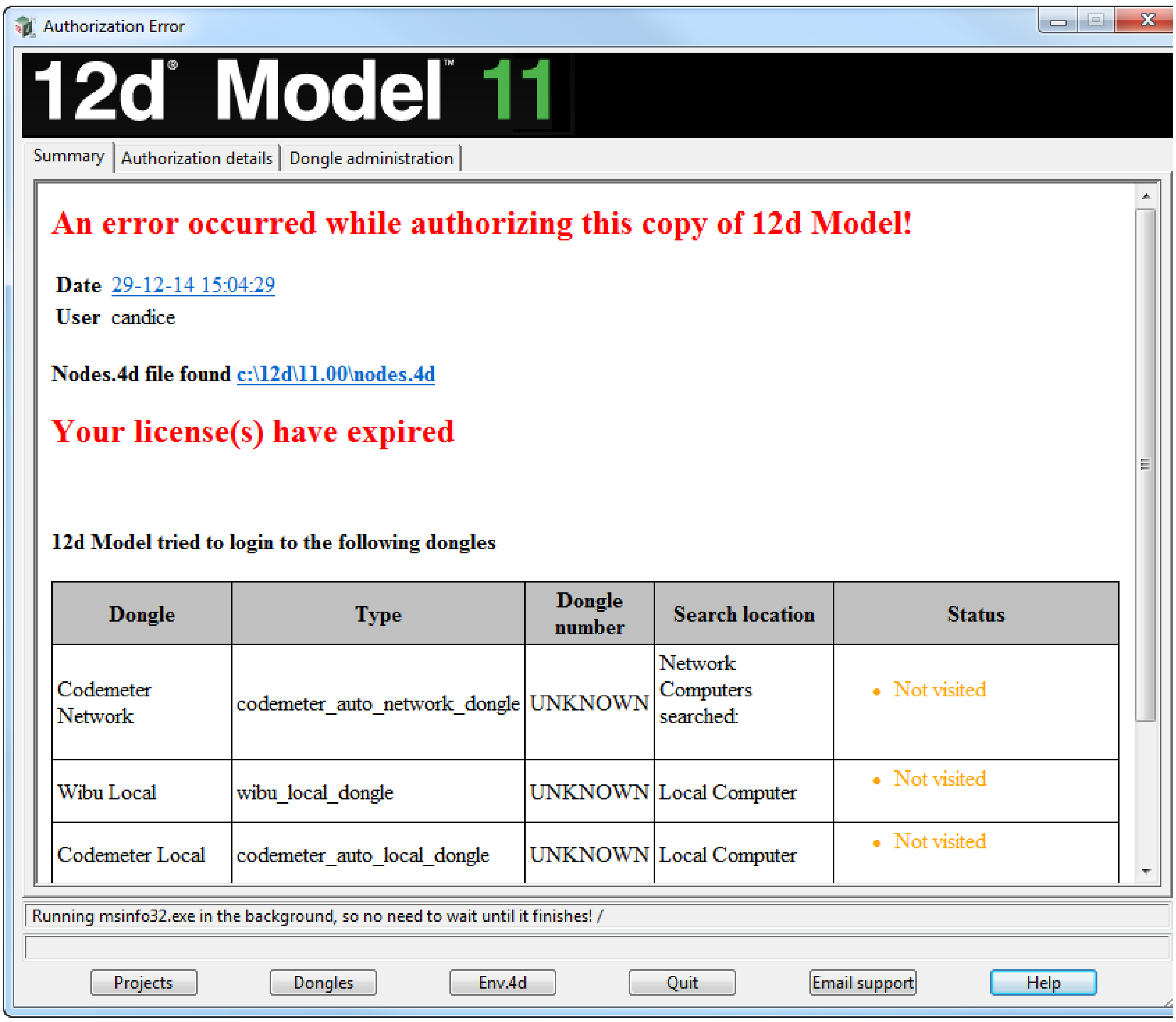
A valid authorization consists of:

1. a ***12d Model*** dongle number from a Wibu dongle or a CodeMeter Container
2. a valid entry in a nodes.4d to match the ***12d Model*** dongle number

If either of the above is missing, then instead of the project being opened, an **Authorization Error** panel is displayed with information to help find the error.

For more information on what is needing for authorising ***12d Model***, see 7.6.5.1 12d Dongles, Dongles.4d, Nodes.4d, and Authorizing.

The **Authorization Error** panel contains information to help find out why there is an authorization error.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Summary tab

*this summarises information from the **Authorization details** tab.*

Authorization details tab

shows details about the

version of 12d Model
nodes.4d file used
dongles.4d file used
env.4d file used
Windows Environment
What is in the Output Window
Windows Registry
Windows Registry User

Dongle administration tab

same as running the ***Project =>Management =>Dongles =>Administration*** *option. See* 7.6.5.2 Dongles Administration.

Buttons at Bottom

Projects button

opens the **12d Model** *front screen so a* **12d Model** *project can be selected or created.*

Dongles.4d button

opens the ***Dongles.4d Editor*** *panel so that the dongles.4d file can be edited. See* 7.6.5.3 Dongles.4d Editor.

Env.4d button

opens the ***Env.4d Editor*** *panel so that the env.4d file can be edited. See* 7.6.3 env.4d.

Email support button

creates an email with an attachment of a zipped up copy of all the information on the ***Authorization details*** *tab. This will provide useful information for someone trying to determine why the project is not authorising.*

Return to 7.6.5.5 Authorization Details or 7.6.5 Dongles.

7.6.5.6 Nodes.4d Editor

Position of option on menu: **Project =>Management =>Dongles => Nodes.4d editor**

The ***nodes.4d*** is an XML format and so can not be easily edited, or even viewed, in its native XML format.

There is a program shipped by ***12d Solutions*** that is used to install the nodes.4d file that can:

(a) Install a ***nodes.4d*** file

but also load an existing XML nodes.4d and

(b) merge in another nodes file

(c) look at all the entries in the file and display information on each entry (start & end dates, modules authorised etc)

(d) move entries up and down

(e) delete entries

(f) create an html report on the entries

(g) create a new ***nodes*** file from selected entries

This program is accessible from inside ***12d Model*** by

(a) the **Nodes** button on the Front screen (the one before you select a project)

(b) the option ***Projects =>Management =>Dongles =>Nodes.4d editor***

and also from outside ***12d Model***

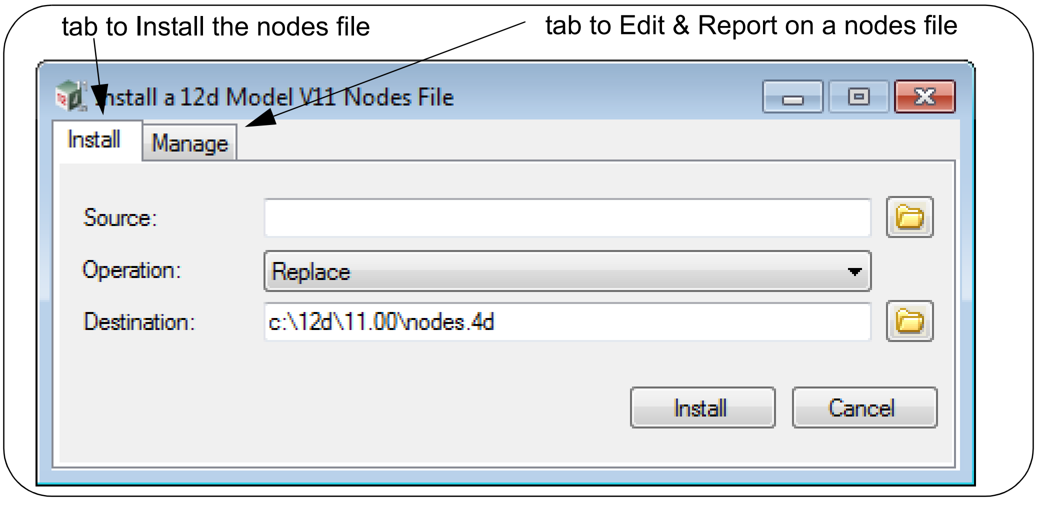
(c) the program is called 12dNodesUtility.exe and is installed as a 32-bit program in

Program Files (x86) \12d\Nodes\11.0

The **Nodes.4d Editor** option is for editing the ***Nodes.4d*** file.

Note that program is actually external to ***12d Model*** and when it is run, it is running independently of ***12d Model***.

Selecting **Nodes.4d editor** brings up the external program and its panel **Installing a *12d Model* 11 Nodes File**:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Install **tab**

this tab is for installing the nodes.4d file.

Source file box file browser

the full path name of the nodes file to install.

Operation choice box Replace Replace, Append, Prepend

*if **Replace**, the Destination file is replaced by the Source file.*

*If **Append**, the Source file is added to the end of the Destination file.*

*If **Prepend**, the Source file is added to the beginning of the Destination file.*

Destination file box file browser

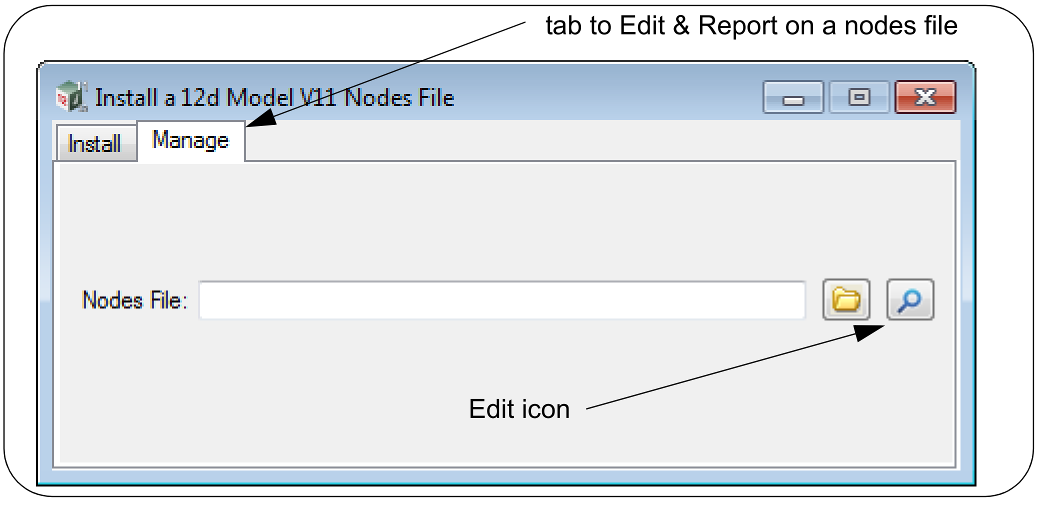
*the full path name of where to install the **nodes.4d** file.*

Install button

install the nodes.4d file.

Manage **tab**

this tab is for editing and reporting a nodes.4d file.



Nodes file file box file browser

the full path name of the nodes file to edit/modify/report.

After the nodes.4d file has been fill in, click on the Edit icon (Looking glass icon) to bring up the **Node Contents** *panel.*

The panel shows all the Client Entries in the nodes.4d file in a tree structure with all the dongle records for a particular Client Entry displayed when the + is clicked on in front of the Client Entry.

***Note:** The Client Entry contains all the dongles records that were produced for a Client at the one time. These dongle records can only stay inside that one Client Entry, even though there may be other Client Entries of the same name.*

Dongle Records can be moved up or down within a Client Entry but can't be moved to a different Client Entry.

A Client Entry can be moved up and down in the Tree.

Highlighted Client Entries can be deleted and all the Dongle Records for the Client Entry will also be deleted.

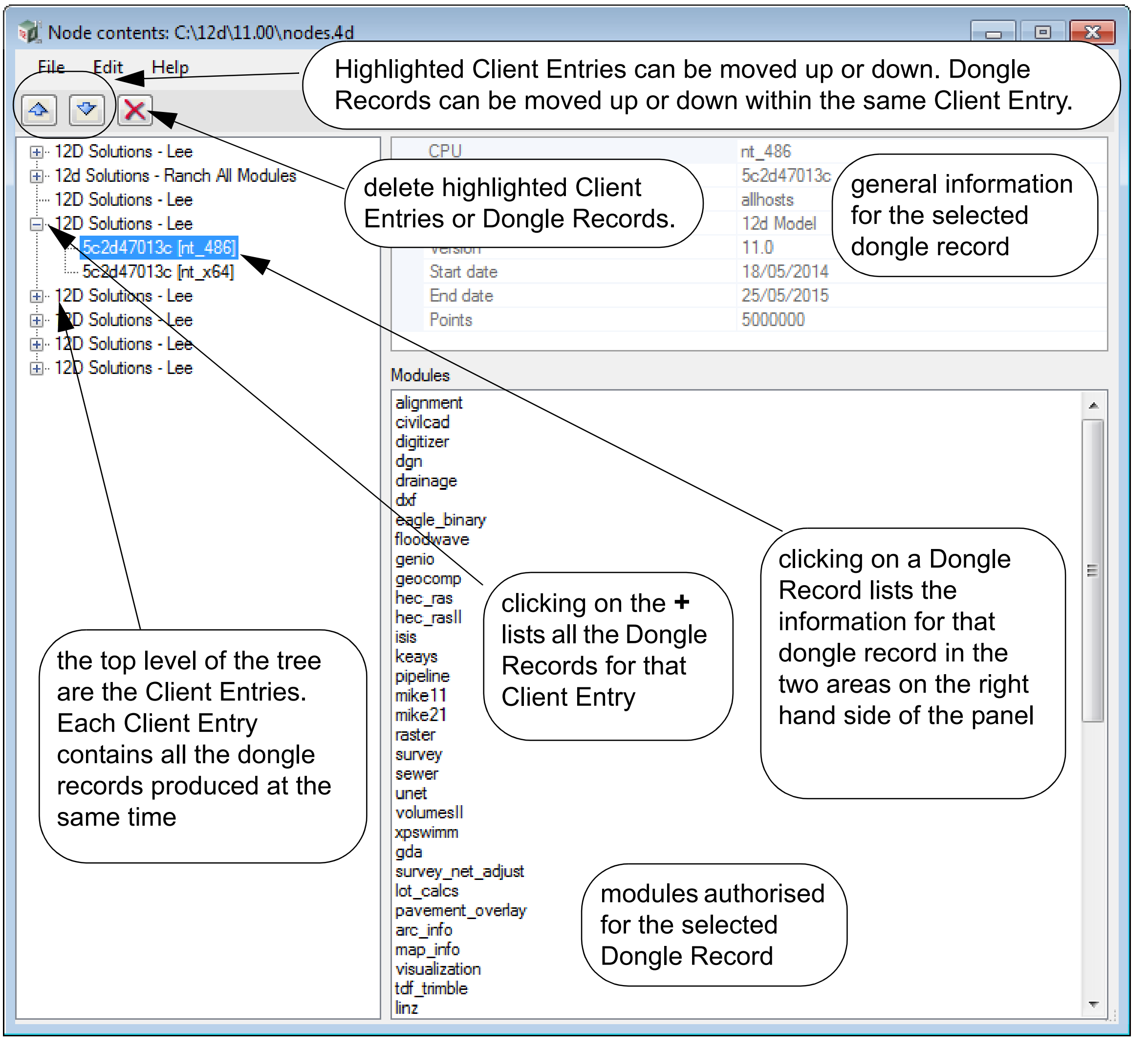
Highlighted Dongle Records can be deleted.

***File** has options to produce a HTML report and write out to a new nodes file sa subset of the Client*

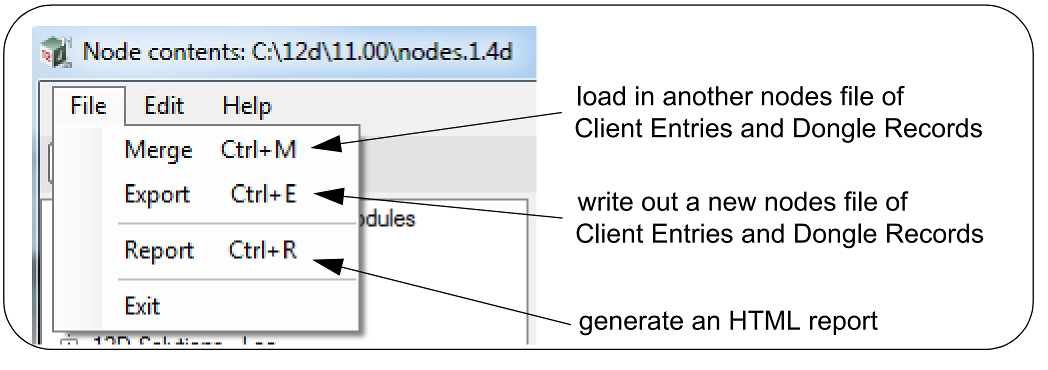
Entries and Dongle Records.

*Clicking on **X** removes the* **Node Contents** *panel*

.

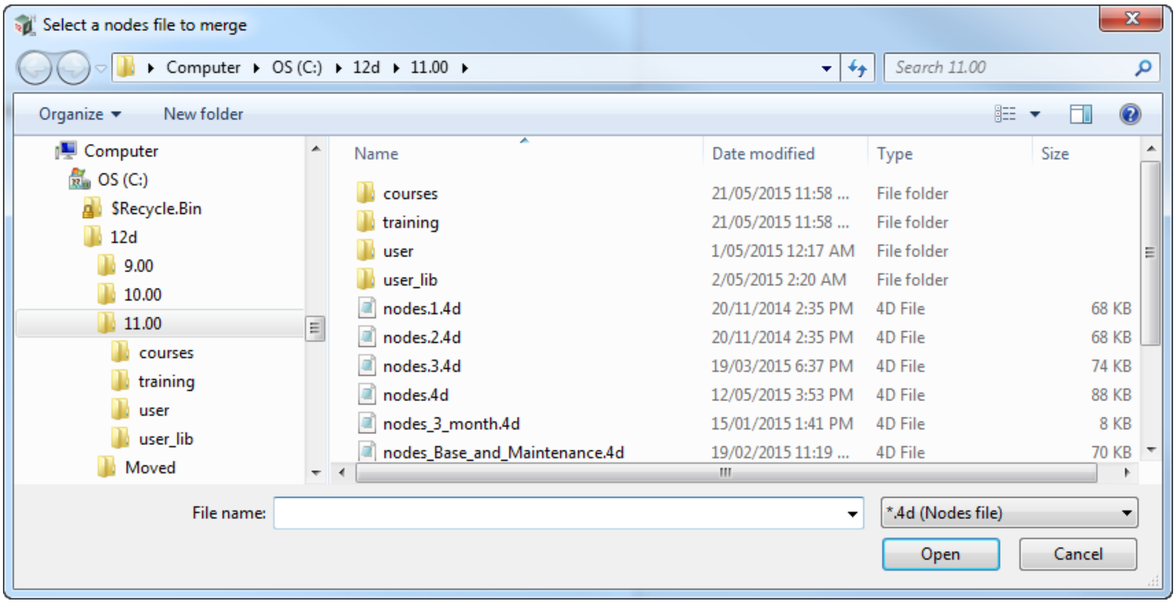


Clicking on **File** brings up the



1. Merge menu item

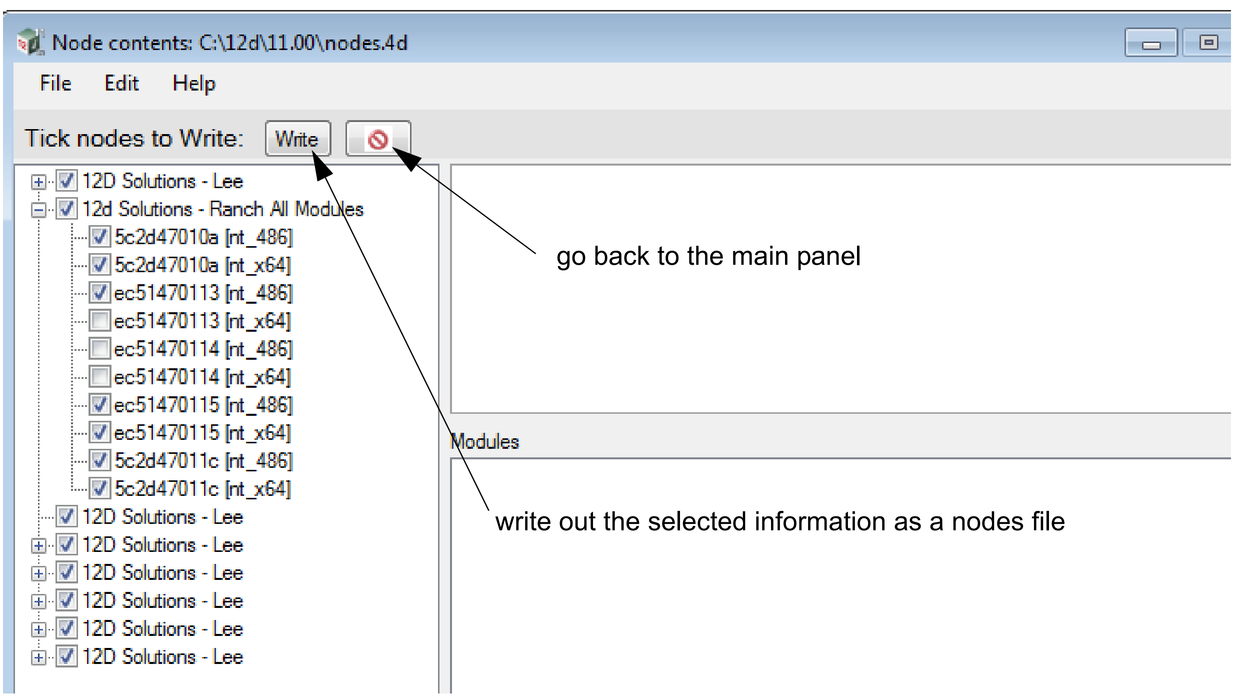
*clicking on **Merge** brings up the Microsoft browser to obtain the another nodes.4d read in.*



The new ***Client Entries*** *and* ***Dongle Records*** *are added to the bottom of the list of existing* ***Client Entries*** *ready for editing.*

2. Export menu item

clicking on ***Export*** *brings a Tree showing all the Client Entries and Dongle Records but with a tick box to indicate if the information is to be written out or not, to the new nodes file.*



Write button

clicking on ***Write*** *brings up the Microsoft* **Save As** *browser to obtain the name of the file to write the selected information out to as a node.4d file.*

Stop button

clicking on ***Stop*** *takes you back to the* **Node Contents** *panel*

3. Report menu item

*clicking on **Report** brings up the Microsoft* **Save As** *browser to obtain the name of the file to write the HTML report to.*

*The report will list all the **Client Entries.** And for each **Client Entry**, the **Dongle Record**s in it showing the modules authorised for that Dongle Record in blue.*

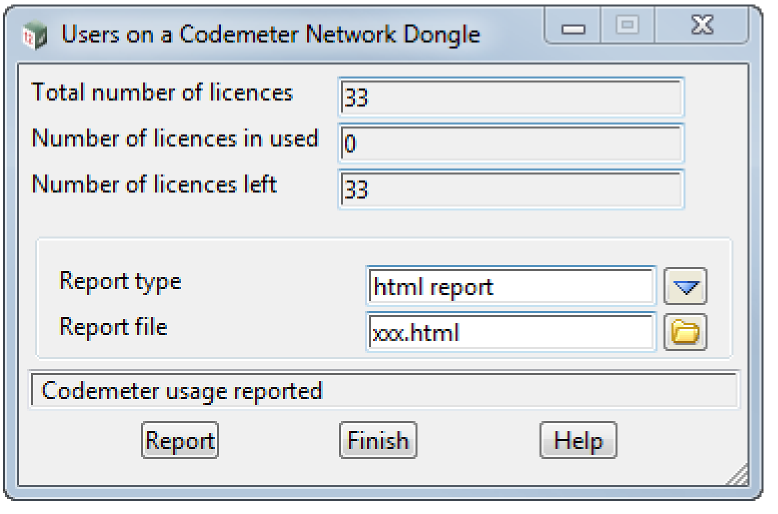
Continue to the next section 7.6.5.7 CodeMeter Network User or return to 7.6.5 Dongles.

7.6.5.7 CodeMeter Network User

Position of option on menu: **Project =>Management =>Dongles => CodeMeter network users**

This option shows the number of ***12d Model*** licenses on the CodeMeter network dongles and how many are used/unused. A report can also be produced showing details for each individual network dongle.

Selecting **CodeMeter network users** brings up s panel **Users on a CodeMeter Network Dongle**:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Total number of licenses	output only		

the total number of 12d Model licenses on the CodeMeter network dongles. This number is updated when **Report** *button is pressed.*

Number of licenses in use output only

the number of 12d Model licenses currently being used.

Number of licenses left output only

the number of 12d Model licenses left to be used.

Report type choice box html report, original xml, <customize>

output format for the report. An XML file will be produced and then if required, converted to the selected report.

For information on setting up custom reports from the generated XML file using xslts, see 4.30 Setting Up XML Reports.

Report file file box

*if **not blank**, an XML file will be created and a report of this name, and of the type given by* **Report type** *will be generated from the XML file.*

*If **blank**, no report is created but the number of licenses information is still written to the appropriate panel fields.*

Report button

check how many licenses are on CodeMEter network dongles and write the values to the panel fields. If Report file is not blank then a report is also produced.

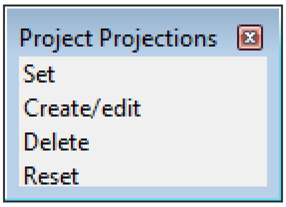
Return to 7.6.5 Dongles.

7.6.6 Projections

Position of option on menu: **Project =>Management =>Projections**

User defined Projections can be defined using the Create/edit option. A user defined projection can then be set for the project (the project projection). Values from the project projection (e.g. scale factor) are used in various options.

The **Project projections** walk-right menu contains various projection items.



set the default projection for the project
create/edit project definitions
delete a project definition
reset the projection set for the project

For the option *Set*, go to the section	7.6.6.1 Set Projection.
Create/edit	7.6.6.2 Create/Edit Projection.
Delete	7.6.6.4 Delete Projection.
Reset	7.6.6.5 Reset Projection.

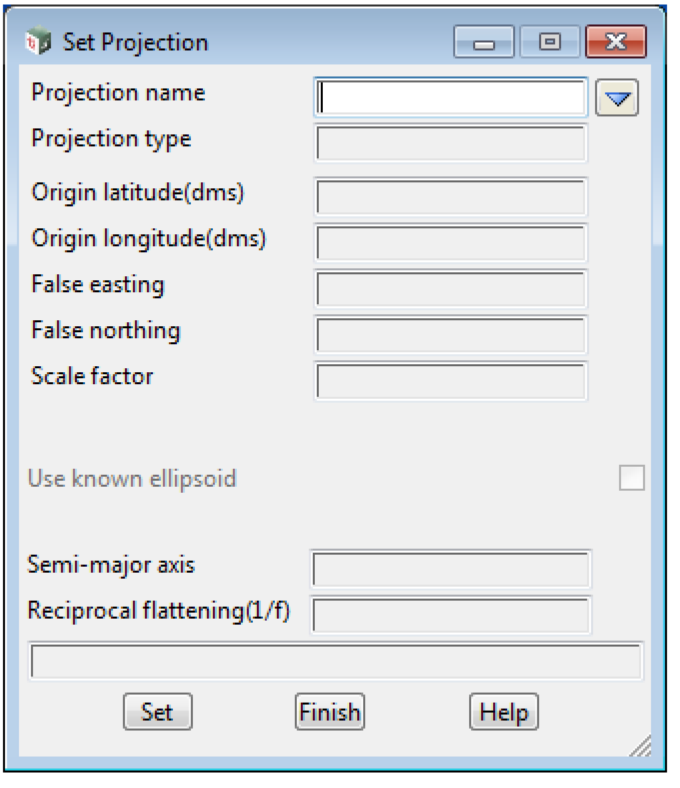
7.6.6.1 Set Projection

Position of option on menu: **Project =>Management =>Projections =>Set**

The **Set** option sets the *Project Projection* to be the one selected from the list defined projections.

To define a new projection, see 7.6.6.2 Create/Edit Projection.

On selecting the **Set** option, the **Set Projection** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Projection name	choice box		all defined projections

name of the projection to be set as the Project projection.

Depending on the projection type, the relevant projection parameters are displayed in the rest of the fields in the panel.

The rest of the fields are for reporting information about the selected projection.
The fields will change depending on the type of the selected Projection.

To define a new projection, see 7.6.6.2 Create/Edit Projection.

Set	button

after selecting this button, the projection for the project is set to that given in the **Project name** *field. The project projection is used in a number of options in* ***12d Model***.

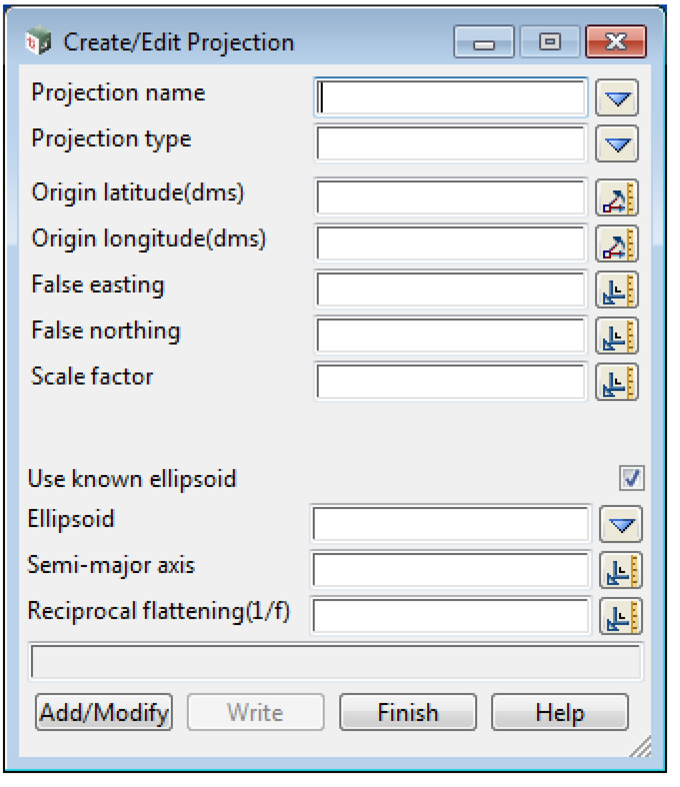
Go to the next section 7.6.6.2 Create/Edit Projection or return to 7.6.6 Projections.

7.6.6.2 Create/Edit Projection

Position of option on menu: **Project =>Management =>Projections =>Create/Edit**

This option defines a new projection, or edits an existing projection in the current list of available projections.

On selecting the **Create/edit** option, the **Create/edit Projection** panel is displayed.



For more information about terminology used in this sections, see the Appendix 38 Geodetics Summary.

Field Description	Type	Defaults	Pop-Up
Projection name	typed input an choice box		all defined projections

name of the projection to be set as the Project projection.

Field Description	Type	Defaults	Pop-Up
Projection Type	choice box		Transverse Mercator, UTM RSO, General

the type of projection to be set as the Project projection.

Depending on the projection type, the relevant projection parameters can be entered.

*For **Transverse Mercator** the fields are:*

Origin latitude (dms); Origin longitude (dms), False Easting, False Northing, Scale factor.

*For **UTM** the fields are:*

Zone number, North/South Hemisphere.

*For **RSO** the fields are:*

Origin latitude (dms); Origin longitude (dms), REctified Skew Azimuth (dms), False Easting, False Northing, Scale factor.

*For **General** the field is Projection Parameter which requires a special line of text defining the projection. For more information on defining a **general** projection, see* 7.6.6.3 Defining a General Cartographic Projection

Use known ellipsoid tick box tick

*if **ticked**, the ellipsoid for the projection has been defined previously and the parameters for the particular ellipsoid can be used. The values of defined ellipsoids are displayed when a selection is made from the ellipsoid pop-up box (e.g. GRS80).*

*if **not ticked** then the ellipsoid parameters can be defined by entering the ellipsoid parameters.*

Add/Modify button

after selecting this button, the user defined projection is added to the current list or the existing projection definition is modified. using the entered values. A number of new projections can be added by simply entering the relevant data and selecting the add/modify button.

Write button

*after selecting this button, the user defined projections defined by the add/modify process are written to a **carto.4d** file which can be written to various locations. This means that the new defined projections can be used for other **12d Model** projects. For more information on the* **Write** *button, go to the section* 39.2.6 Writing Out Setup Files *in the Appendix* 39 Setting Up & Configuring 12d.

Go to the next section 7.6.6.3 Defining a General Cartographic Projection or return to 7.6.6 Projections.

7.6.6.3 Defining a General Cartographic Projection

A general *cartographic projection* is defined by giving a series of parameters starting with the mandatory +proj parameter giving the projection type. For example,

+proj=*projection_type* e.g. +proj=tmerc or +proj=utm

To specify the Earths's elliptical figure used in the projection, two parameters are required.

The first required value is the major semi-axis of the ellipse or equatorial radius,

major semi-axis of the ellipse	+a= *value*

and the second parameter can be any one of the following standard forms:

minor semi-axis of the ellipse	+b= *value*
flattening	+f= *value*
reciprocal flattening (i.e. 1/f)	+rf= *value*
eccentricity)	+e= *value*
eccentricity squared)	+es= *value*

For example, the Australian National Spheroid (ANS) is defined by:

+a=6378160
+rf= 298.25

GRS80 is defined by:

+a=6378137
+rf= 298.257222101

WGS84 is defined by:

+a=6378137
+rf= 298.257223663

A further three parameters are common to most projections:
the central meridian +lon_0=*value*, and the cartesian offsets for the respective x and y axis, +x_0=*value*, +y_0=*value* (often referred to as false easting and northing).

central meridian	+lon_0= *value*	*units in decimal degrees*
x offset- false easting	+x_0=*value*	
y offset - false northing	+y_0=*value*	

A fourth parameter, +lat_0=*value*, is used to designate a central parallel and associated y axis origin for a projection.

Unless a value is specified for the parameters lon_0, lat_0, x_0 and y_0, they are assumed to be zero.

Other parameters will depend on the particular projection used.

See:

7.6.6.3.1 Transverse Mercator Projection +proj=tmerc

7.6.6.3.2 Universal Transverse Mercator Projection +proj=utm

7.6.6.3.3 Australian Map Grid (AMG84)

7.6.6.3.4 Map Grid of Australia (MGA94)

7.6.6.3.5 Lambert Conformal Conic Projection +proj=lcc

Go to the next section 7.6.6.3.1 Transverse Mercator Projection +proj=tmerc or return to 7.6.6 Projections.

7.6.6.3.1 Transverse Mercator Projection +proj=tmerc

project	+proj=tmerc
scale factor	+k=*value*

e.g. Metro-Perth is:
+proj=tmerc +a=6378160 +rf=298.25 +lon_0=115.833333333 +lat_0=0.0 +x_0=54466.561 +y_0=3690893.265 +k=1.0

Perth Coastal Grid 1984 (PCG84) is:
+proj=tmerc +a=6378160 +rf=298.25 +lon_0=115.833333333 +x_0=40000 +y_0=3800000 +k=1.000006

Go to the next section 7.6.6.3.2 Universal Transverse Mercator Projection +proj=utm or return 7.6.6.3 Defining a General Cartographic Projection or 7.6.6 Projections.

7.6.6.3.2 Universal Transverse Mercator Projection +proj=utm

This is a special from of the Transverse Mercator Projection. The central meridian is constrained to 6 degree intervals starting at 3 degrees. An extra parameter exists called +zone where +zone=1 specifies the region from 180 degree W to 174 degrees W (equivalent to +lon_0=177 degrees W) and proceeds easterly until +zone=60 for the region from 174 degree E to 180 degrees E (i.e. +lon_0=177 degrees E). Hence

+zone=N	N = 1, 2,... 60

which automatically defines the central meridian +lon_0= 6 x N - 183 where N = zone number.

The +south option adds a false northing of 10,000,000 m (used for AMG and MGA co-ordinates) for projection in the Southern Hemisphere.

+south	defines +y_0=10000000

In all cases, for a Universal Transverse Mercator projection, a false easting of 500,000 m is used.

+x_0=500000

Go to the next section 7.6.6.3.3 Australian Map Grid (AMG84) or return 7.6.6.3 Defining a General Cartographic Projection or 7.6.6 Projections.

7.6.6.3.3 Australian Map Grid (AMG84)

The Australian Map Grid is a Universal Transverse Mercator Projection with

(a) coordinates are in metres

(b) zones are 6 degrees wide plus overlapping belts of 80 kilometres at each grid junction

(c) AMG zones are numbered from zone 49 with central meridian 111 degrees E to zone 57 with central meridian 159 degrees E.

(d) the origin of each zone is the intersection of the central meridian with the equator

(e) a central scale factor, k, is defined as 0.9996

(f) a false easting of 500,000 and a false northing of 10,000,000 are used

(g) uses the Australian National Spheroid +a=6378160 +rf=298.25

For example, the definition of AMG zone 50 is:

+proj=utm +south +zone=50 +k=0.9996 + a=6378160 +rf=298.25

Go to the next section 7.6.6.3.4 Map Grid of Australia (MGA94) or return 7.6.6.3 Defining a

General Cartographic Projection or 7.6.6 Projections.

7.6.6.3.4 Map Grid of Australia (MGA94)

The Map Grid of Australia is a Universal Transverse Mercator Projection with

(a) coordinates are in metres

(b) zones are 6 degrees wide plus overlapping belts of 80 kilometres at each grid junction

(c) MGA zones are numbered from zone 49 with central meridian 111 degrees E to zone 57 with central meridian 159 degrees E.

(d) the origin of each zone is the intersection of the central meridian with the equator

(e) a central scale factor, k, is defined as 0.9996

(f) a false easting of 500,000 and a false northing of 10,000,000 are used.

(g) uses GRS80 +a=6378137 +rf=298.257222101

For example, the definition of MGA zone 50 is:

+proj=utm +south +zone=50 +k=0.9996 + a=6378137 +rf=298.257222101

Go to the next section 7.6.6.3.5 Lambert Conformal Conic Projection +proj=lcc or return 7.6.6.3 Defining a General Cartographic Projection or 7.6.6 Projections.

7.6.6.3.5 Lambert Conformal Conic Projection +proj=lcc

This is a special from of the Conic Projection.

There are two standard parallels +lat_1 and +lat_2 and the projection is centred on +lat_0 and +lon_0

For example, a Lambert Conformal Conic Conformal with standard parallels of 15 degrees South and 39 degrees South, centred at centred at 27degrees South 134degrees East, with a false origin at that point is 5,000,000m/5,000,000m and using the Australian National Spheroid (i.e. AGD84) is

+proj=lcc +lat_1=-15 +lat_2=-39 +lon_0=134 +lat_0=-27
+x_0=5000000 +y_0=5000000 +a=6378160 +rf=298.25

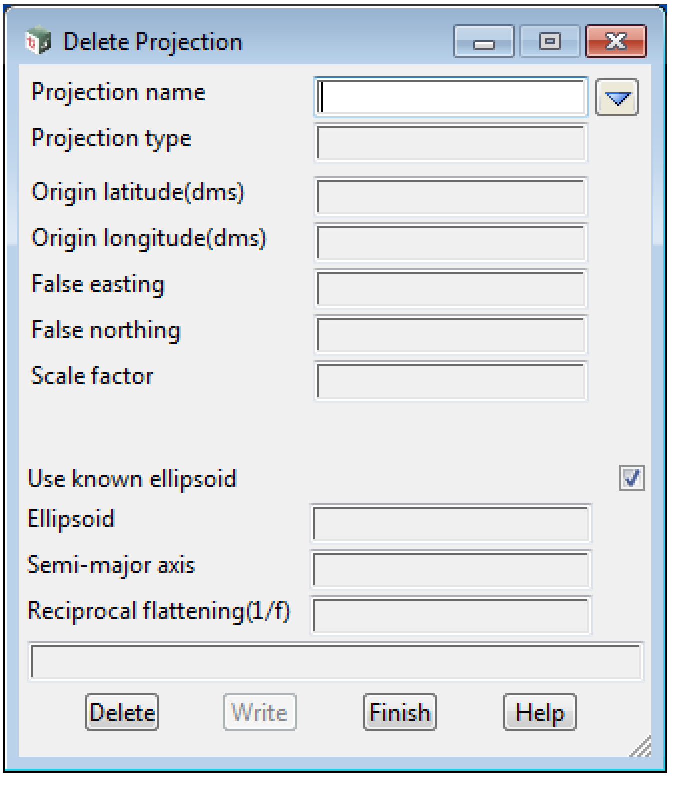
Go to the next section 7.6.6.4 Delete Projection or return 7.6.6.3 Defining a General Cartographic Projection or 7.6.6 Projections.

7.6.6.4 Delete Projection

Position of option on menu: **Project =>Management =>Projections =>Delete**

The **Delete** option deleted a projection from the list of available projections.

On selecting the **Delete Projection** option, the **Delete Projection** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Projection name	input		all defined projections

name of the projection to be deleted from the Project projection list.

The rest of the fields depend upon the projection type and are used to display information about the projection.

Delete	button		

after selecting this button, the selected projection is deleted from the project projection list. This can be done for a number of projections by simply selecting the projection to be deleted and selecting the delete button.

Write	button		

after selecting this button, the list of projections altered by the delete process can be written in various locations. This means that the updated projections list will be used to set current and future projects. For more information on the **Write** *button, go to the section* 39.2.6 Writing Out Setup Files *in the Appendix* 39 Setting Up & Configuring 12d

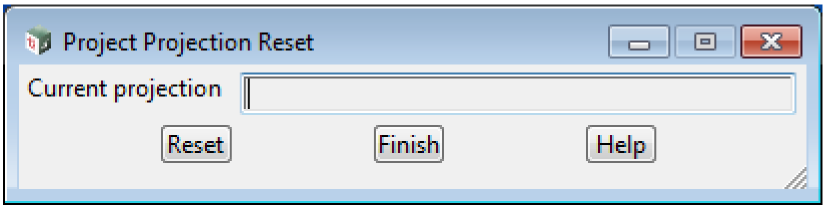
Go to the next section 7.6.6.5 Reset Projection or return 7.6.6.3 Defining a General Cartographic Projection or 7.6.6 Projections.

7.6.6.5 Reset Projection

Position of option on menu: **Project =>Management =>Projections =>Reset**

The **Reset** option is used to remove the projection set for the project.

On selecting the **reset Projection** option, the **Project Projection Reset** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	pop-Up
Current projection	output only		

name of the projection currently set for the project.

Reset	button		

after selecting this button, the projection for the project is set to none. That is, no project has been set for the Project.

Go to the next section 7.6.7 N values or return to 7.6 Management.

7.6.7 N values

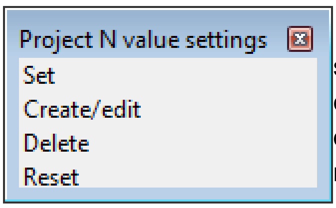
Position of option on menu: **Project =>Management =>N values**

N values are the separation distances between the geoid and the ellipsoid. In GPS surveys, heights are often given in ellipsoid heights whilst most other level datums are based on the geoid. Since most geodetic calculations are based on the ellipsoid, any observations should be reduced onto the ellipsoid. This reduction process takes into the consideration the heights above the ellipsoid. Therefore, it is necessary to convert non ellipsoid heights to ellipsoid values by adding the geoid-ellipsoid separations (n values). There are various methods for determining the N values.

An N value method can be defined using the Create/edit option. Values from the project N value settings are used to determine which method is used to calculate N values. These values are used in various options where the level values are not ellipsoid values and require the N value to be added to get to an ellipsoid height. A user defined N value method can then be set for the project (the project N value setting) which is then the default N Values in any panel.

For more information about terminology used in this section, see the Appendix 38 Geodetics Summary.

The **Project n value** walk-right menu contains various projection items.



set the default N-values for the project
create/edit N-values definitions
delete a N-values definition
reset the N-values set for the project

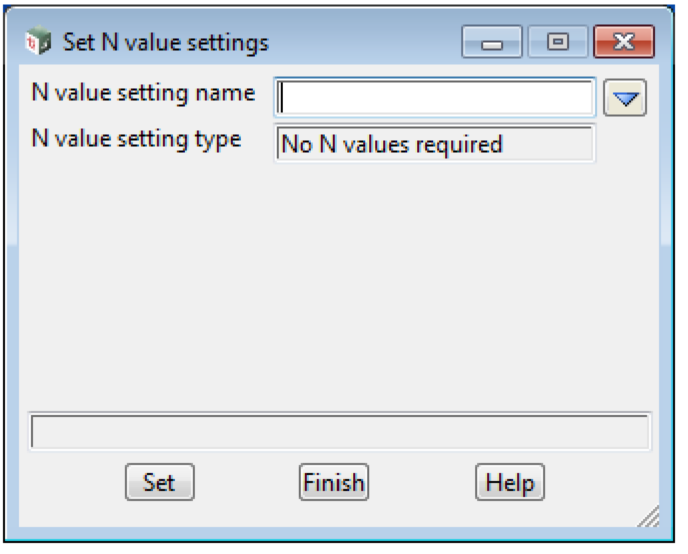
For the option *Set*, go to the section 7.6.7.1 Set N-Values.
Create/edit 7.6.7.2 Create/Edit N-Values.
Delete 7.6.7.3 Delete N-Values.
Reset 7.6.7.4 Reset N-Values.

7.6.7.1 Set N-Values

Position of option on menu: **Project =>Management =>N values =>Set**

The **Set** option sets the Project N Values and these are then used as the default in panels.

On selecting the **Set** option, the **Set N value settings** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
N value setting name	choice box		all defined N value settings

name of the n value setting to be set as the Project N value setting.

N value setting type	input		defined N value types

Depending on the N value interpolation method, the relevant method is displayed.

Set	button		

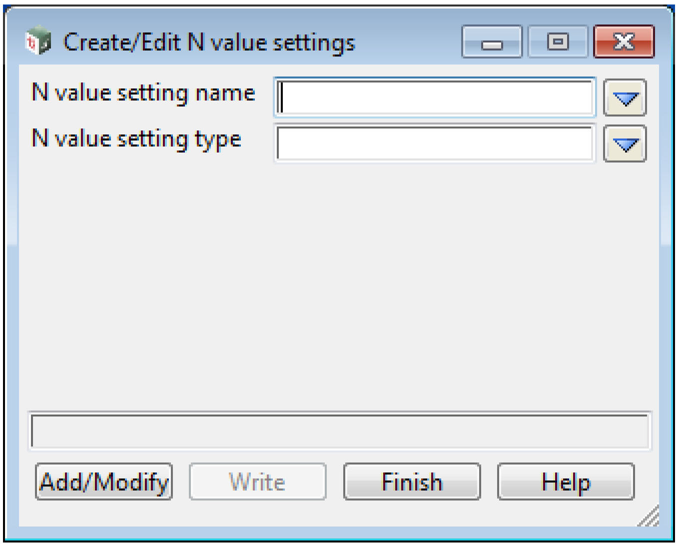
after selecting this button, the current N value setting is set using the displayed values. These values are used as defaults in panels that need N values for calculations

Go to the next section 7.6.7.1 Set N-Values or return to 7.6.7 N values.

7.6.7.2 Create/Edit N-Values

Position of option on menu: **Project =>Management =>N values=>Create/Edit**

On selecting the **Create/edit** option, the **Create/edit N value settings** panel is displayed.

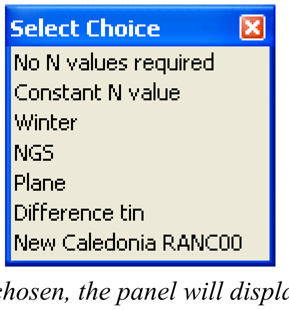


Field Description	Type	Defaults	Pop-Up
N value setting name	choice box		all defined N value settings

name of the n value setting to be set as the Project N value setting.

Field Description	Type	Defaults	Pop-Up
N value setting type	choice box		defined N value types

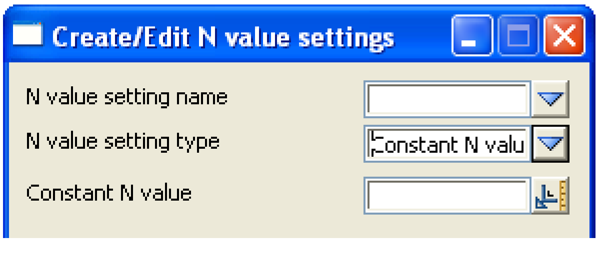
There are various methods for the determination of the N values. These are given in the choice box:



Depending on the setting type chosen, the panel will display the appropriate fields.

No N values required: This applies to data that already has ellipsoid heights and requires no interpolation for N values.

For N value setting type = Constant N value



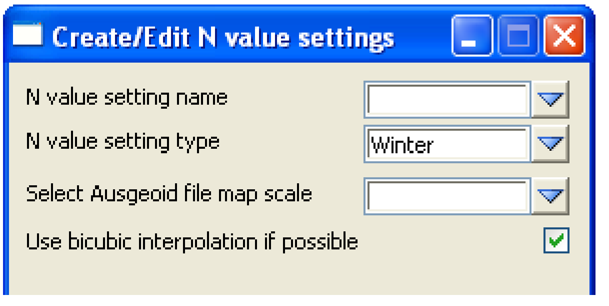
Field Description Type Defaults Pop-Up

Constant N value input box

this N value will be used for the entire n value interpolation process. (i.e. it will not change).

For N value setting type = Winter

This method uses the same method as the Winter interpolation software supplied by Auslig. The data files used should be winter compatible and reside in a folder that is pointed to by the WINTER_DATA_4D parameter in the env.4d file. The data files can be downloaded from the Auslig website.



Field Description	Type	Defaults	Pop-Up
Select Ausgeoid file map scale	choice box		1:100,000
			1:250,000
			1:1,000,000

this value will be dependant on the type of files to be used by winter. ***The files themselves should be in a folder that is pointed to by the WINTER_DATA_4D parameter in the setup area of the env.4d file.***

Use bicubic interpolation if possible tick box ticked

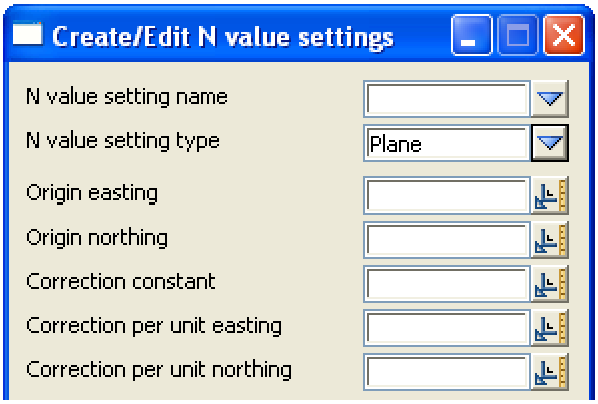
if ticked, the bicubic interpolation method will be used if possible.

For N value setting type = NGS

This method is currently under development.

For N value setting type = Plane

This method allows a plane to be defined allowing N values to be derived from that plane.



Field Description	Type	Defaults	Pop-Up
Origin easting	input box		

the origin easting value.

Origin northing	input box		

the origin easting value.

Corr constant	input box		

the constant z value of the plane.

Corr per unit easting	input box		

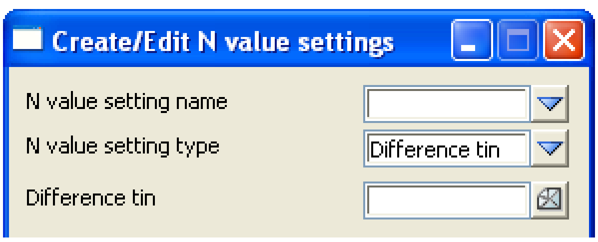
the z value correction per unit easting.

Corr per unit northing	input box		

the z value correction per unit nothing.

For N value setting type = Difference tin

This method allows value to be interpolated from a difference tin. A difference tin is simply a tin of difference values (N values). This tin can be re-triangulated as new points become available, thus introducing more points than a regular grid.



Field Description	Type	Defaults	Pop-Up
Difference tin	tin box		Available tins

the difference tin from which the N values will be interpolated.

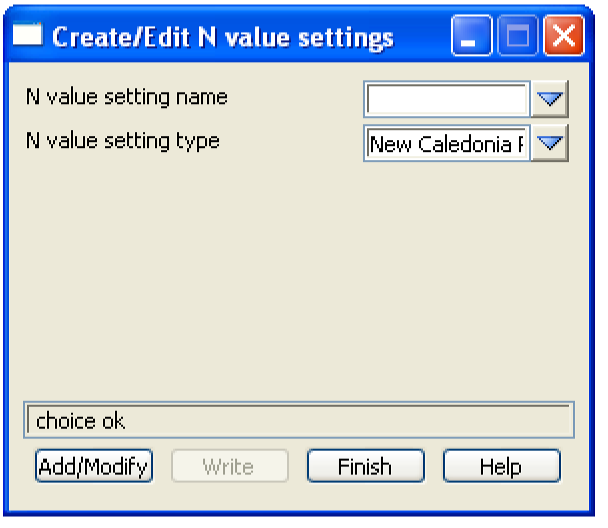
Add/Modify	button		

after selecting this button, the user defined N value settings are added to the current list or the existing N value settings is modified. using the entered values. A number of new settings can be added by simply entering the relevant data and selecting the add/modify button.

Write button

after selecting this button, the user defined N value settings defined by the add/modify process can be written to the file nvalues.4d in various locations. This means that the defined N value settings will be allowed to be set in current and future projects. For more information on the **Write** *button, go to the section* 39.2.6 Writing Out Setup Files *in the Appendix* 39 Setting Up & Configuring 12d

For N value setting type = New Caledonia RANC00

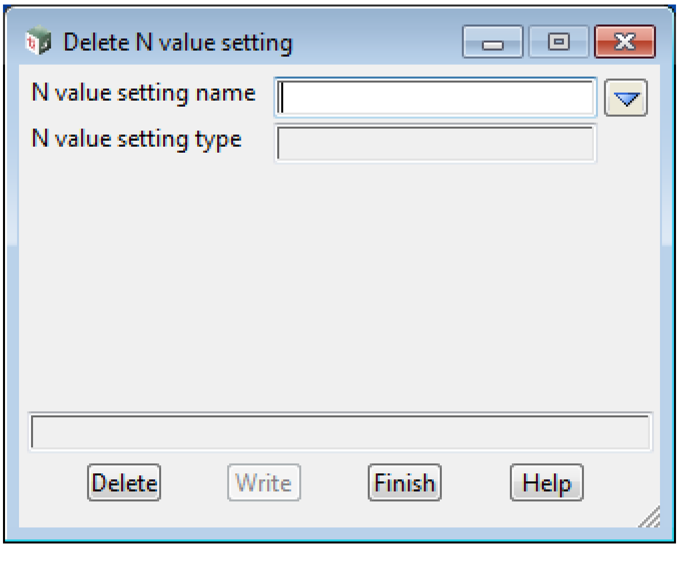


7.6.7.3 Delete N-Values

Position of option on menu: **Project =>Management =>N value =>Delete**

The **Delete** option deleted a projection from the list of available projections.

On selecting the **Delete** option, the **Delete N value setting** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
N value setting name	input		all defined N value settings

name of the setting to be deleted from the N value setting list.
Depending on the setting type, the relevant setting parameters are displayed.

Delete button

after selecting this button, the selected N value setting is deleted from the setting list. This can be done for a number of n value settings by simply selecting the projection to be deleted and selecting the delete button.

Write button

after selecting this button, the list of n value settings altered by the delete process can be written to the nvalues.4d file in various locations. This means that the updated n value settings list will be used to set current and future projects. For more information on the **Write** *button, go to the section* 39.2.6 Writing Out Setup Files *in the Appendix* 39 Setting Up & Configuring 12d.

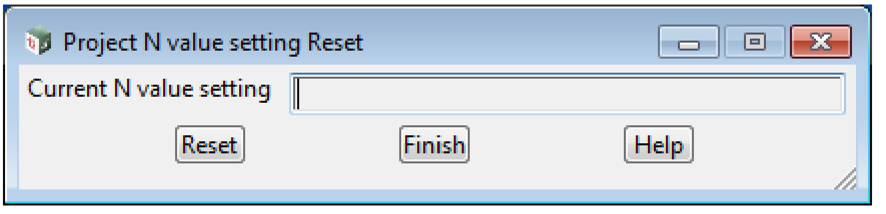
Go to the next section 7.6.7.4 Reset N-Values or return to 7.6.7 N values.

7.6.7.4 Reset N-Values

Position of option on menu: **Project =>Management =>N values =>Reset**

The.**Reset** option is used to remove the projection set for the project.

On selecting the **reset** option, the **Project N value setting reset** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Current N value setting	output only		

name of the n value setting, set for the project.

Reset	button		

after selecting this button, the N value setting for the project is set to none. That is, no N value setting has been set for the Project.

Go to the next section 7.6.8.1 Create/Edit 7 Parameters or return to 7.6.7 N values.

7.6.8 7 Parameters

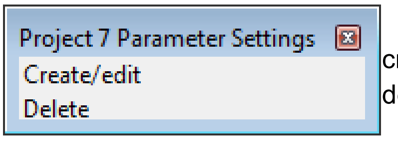
Position of option on menu: **Project =>Management =>7 parameters**

Seven (7) parameter transformations are used to transform data between two ellipsoids. They are also known as 7 parameter similarity transforms, Bursa-Wolf and 7 parameter Helmert transformations.

***Warning*: the definition and hence sign of some terms varies between countries.**

In ***12d Model***, the seven parameter similarity transformations are used in the ***General Transformation*** option **Survey =>Conversions =>General transformations** (see 17.10.6 General Transformations) and the transformation is applied in the Global XYZ system.

The **7 parameters** walk-right menu contains options to create and delete seven parameter transformation setting.



create/edit 7 parameter definitions
delete 7 parameter definitions

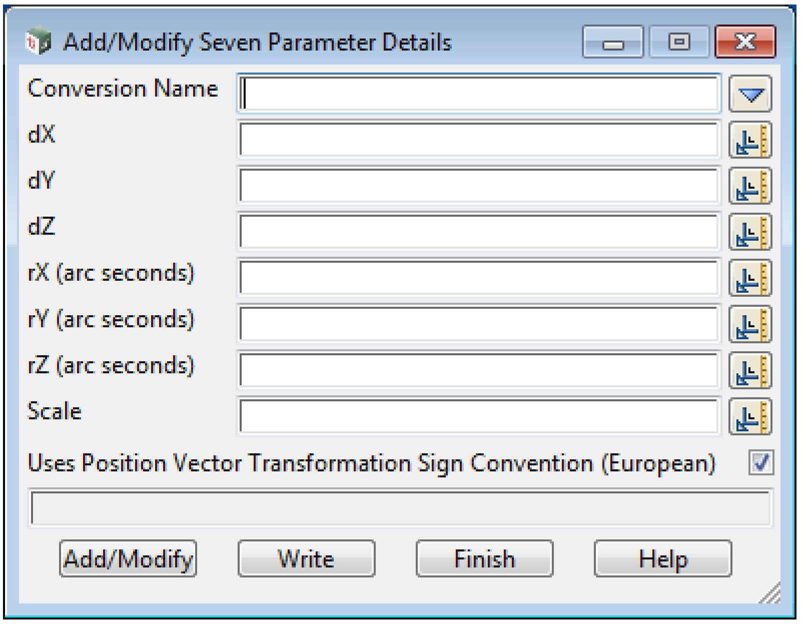
For the option *Create/edit*, go to 7.6.8.1 Create/Edit 7 Parameters
Delete 7.6.8.2 Delete 7 Parameters

The options in the menu will now be described.

7.6.8.1 Create/Edit 7 Parameters

Position of option on menu: **Project =>Management =>7 parameters=>Create/Edit**

On selecting the **Create/edit** option, the **Add/Modify Seven Parameter Details** panel is displayed.



Field Description	Type	Defaults	Pop-Up
Conversion name	choice box		all defined 7 param settings

name of the 7 parameter setting to be created/edited.

DX, DY, DZ input box

the translations for the 7 parameter transformation.

rX (arc seconds), rY (arc seconds, rZ (arc secondsinput box

the rotations, in arc seconds, for the 7 parameter transformation.

Scale input box

the scale for the 7 parameter transformation.

Use Position Vector Transformation sign convention (European) tick box

if ticked, the European Position Vector Transformation convention is used.
If not ticked, PVT sign is not used (mainly not used for USA).

Depending on the sign convention, some of the parameters will have the opposite sign.

Add/Modify button

after selecting this button, either the new 7 parameter set is created or an exiting one modified. To save the setting list, use the **Write** *button.*

Write button

after selecting this button, the user defined 7 parameter settings are written out to the file ***7params.4d*** *in the user selected folder. This means that the 7 parameter settings will be available in the current and future projects. For more information on the* **Write** *button, go to the section* 39.2.6 Writing Out Setup Files *in the Appendix* 39 Setting Up & Configuring 12d.

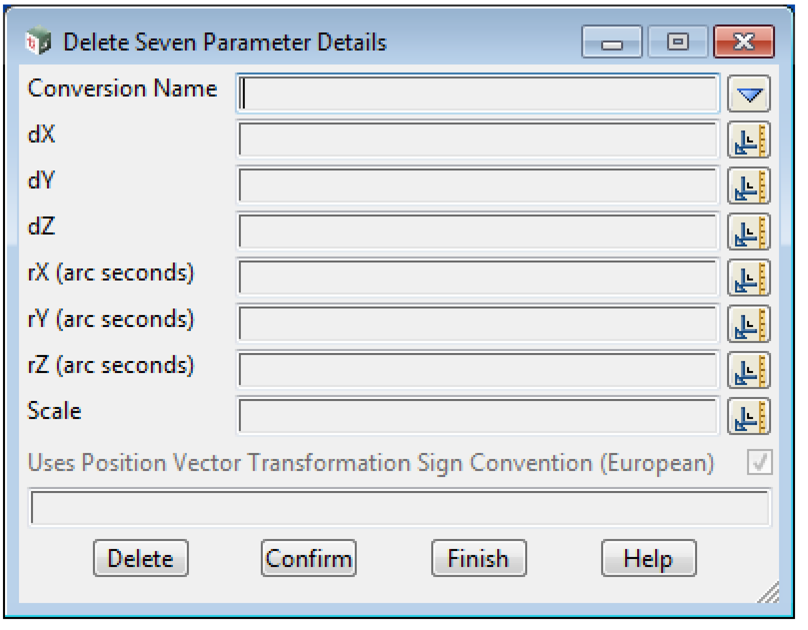
Go to the next section 7.6.8.2 Delete 7 Parameters or return to 7.6.8 7 Parameters.

7.6.8.2 Delete 7 Parameters

Position of option on menu: **Project =>Management =>7 parameters =>Delete**

The **Delete** option deleted a seven parameter set from the list of available seven parameter settings.

On selecting the **Delete** option, the **Delete 7 Parameter Details** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Conversion name	input		all defined 7 param settings

name of the 7 parameters setting to be deleted from the 7 parameter settings list. The relevant setting parameters are displayed.

Delete	button		

after selecting this button, the selected 7 parameter setting is deleted from the setting list. This can be done for a number of 7 parameter sets by simply selecting the name to be deleted and selecting the delete button. To save the modified setting list, use the **Write** *button*

Write	button		

after selecting this button, the user defined 7 parameter settings are written out to the file ***7params.4d*** *in the user selected folder. This means that the 7 parameter settings will be available in the current and future projects. For more information on the* **Write** *button, go to the section* 39.2.6 Writing Out Setup Files *in the Appendix* 39 Setting Up & Configuring 12d

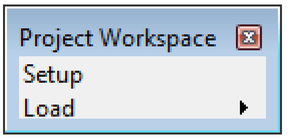
Return to 7.6.8 7 Parameters.

7.6.9 Project Workspace

Position of option on menu: **Project =>Management =>Workspace**

Options to set up the project

The **Project Workspace** walk-right menu contains various projection items.



create/edit workspace files
list and load workspace files

For the option *Setup*, go to 7.6.9.1 Workspace Setup

Load 7.6.9.2 Project Workspace Load

7.6.9.1 Workspace Setup

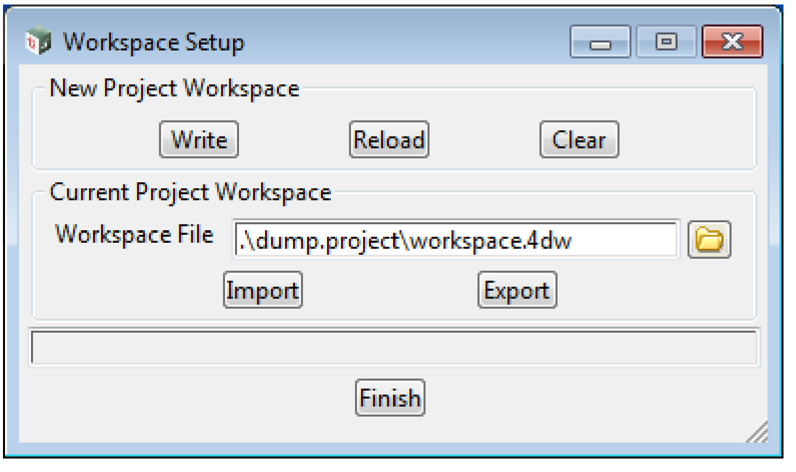
Position of option on menu: **Project =>Management =>Workspace =>Setup**

The workspace file is used to define the position of toolbars and output window for a new projects.

Once inside an existing project, the workspace file is no longer used and the final positions of the toolbars and output window are recorded inside the project so they come up in the same final position as when the project is reopened.

The Workspace Setup option allows the user to define new workspace setup files that can be loaded at any time in an existing project or used as the default workspace file used for all new projects.

On selecting the **Setup** option, the **Workspace Setup** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Write	button		

write out the current toolbar configuration to a .4dw file.

Load	button		

read in the workspace.4dw file in the current project folder.

Clear	button		

remove all the current toolbars and read in the default workspace.4dw file

Workspace file	file box	current workspace file	*.4dw files

name of the workspace file to re imported/exported

Import	button		

read in and use the workspace file given in the **Workspace file** *field.*

Export	button		

write the current toolbar settings out to the workspace file given in the **Workspace file** *field.*

Go to the next section 7.6.9.2 Project Workspace Load or return to 7.6.9 Project Workspace.

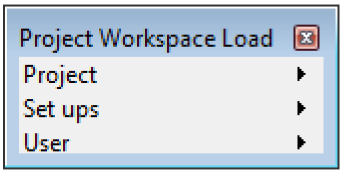
7.6.9.2 Project Workspace Load

Position of option on menu: **Project =>Management =>Workspace >Load**

This option displays and selects a project workspace file to set up the icons etc. on the screen.

The **Project Workspace Load** walk-right menu contains further walk-rights to display the workspace files in the local project, the set ups folder and the user folder.

To load a workspace file, simply click on the name in the walk-right lists.



list and load workspace files in the project
list and load workspace files in the set ups folder
list and load workspace files in the user folder

Return to 7.6.9 Project Workspace.

7.6.10 Tags

For the option *Tag tree* go to	7.6.10.1 Tag Tree
Find new tags	7.6.10.2 Register New Tags

7.6.10.1 Tag Tree

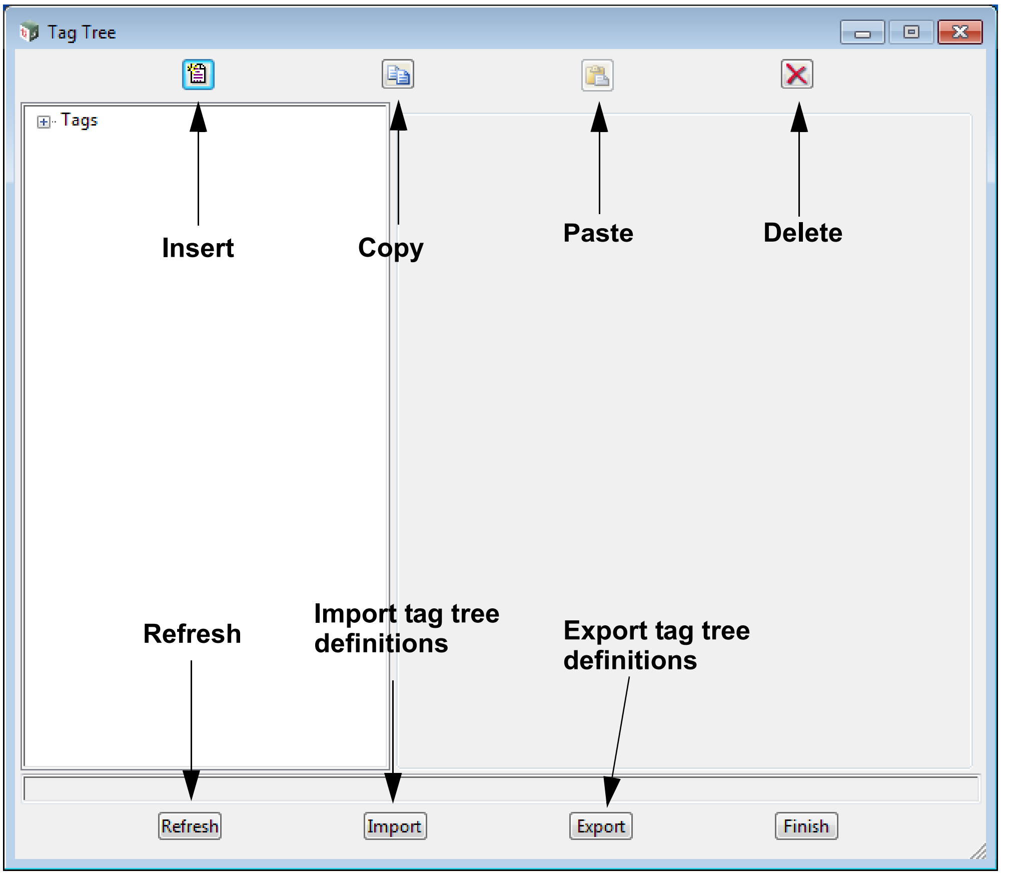
Position of option on menu: **Project =>Management =>Tags =>Tag tree**

The ***12d Model*** objects *models*, *tins* and *strings* can be tagged with an unlimited number of user defined text strings called ***tags.*** For example, a string may be tagged with the two tags, "survey day 1" and "sewer".

Tags names are ***case sensitive*** and must be ***unique***.Tags are used in an increasing number of ***12d Model*** options. For example, on a view, models can be selected for adding/removing by giving a *tag* name.

To further increase their usefulness, tags in a project can also be assigned a ***hierarchy***, or ***tree structure***. However that will be discussed shortly after first describing how to create, apply and removed simple tags (see Defining a Tag Tree (Tag Hierarchy)).

Selecting **Tags** displays the **Tag Tree** panel which is used to defined tags, and also apply and remove tags from objects.



Buttons at Bottom grid

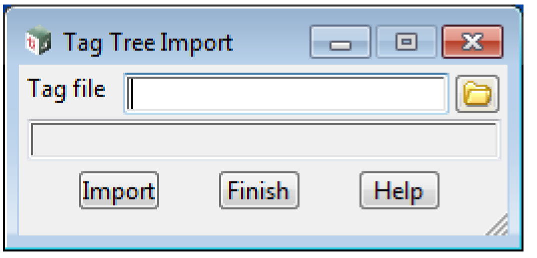
Refresh button

refresh the grid.

Import button

import a tag tree definition file.

Clicking on **Import** *brings up the* **Tag Tree Import** *panel.*



Tag file tags box all .tags files

name of the tag file to read the tag definitions from.

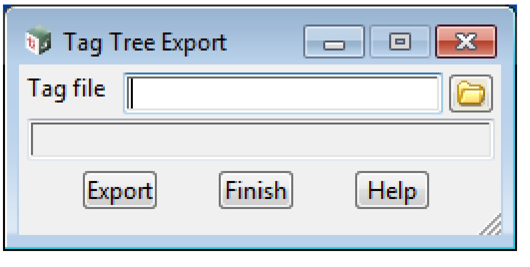
Import button

*read in the tag definitions from the **Tag file** and merge them with the current definitions for the project.*

Export button

export a tag tree definition file.

Clicking on **Export** *brings up the* **Tag Tree Export** *panel.*



Tag file tags box all .tags files

name of the tag file to write the project tag definitions out to.

Export button

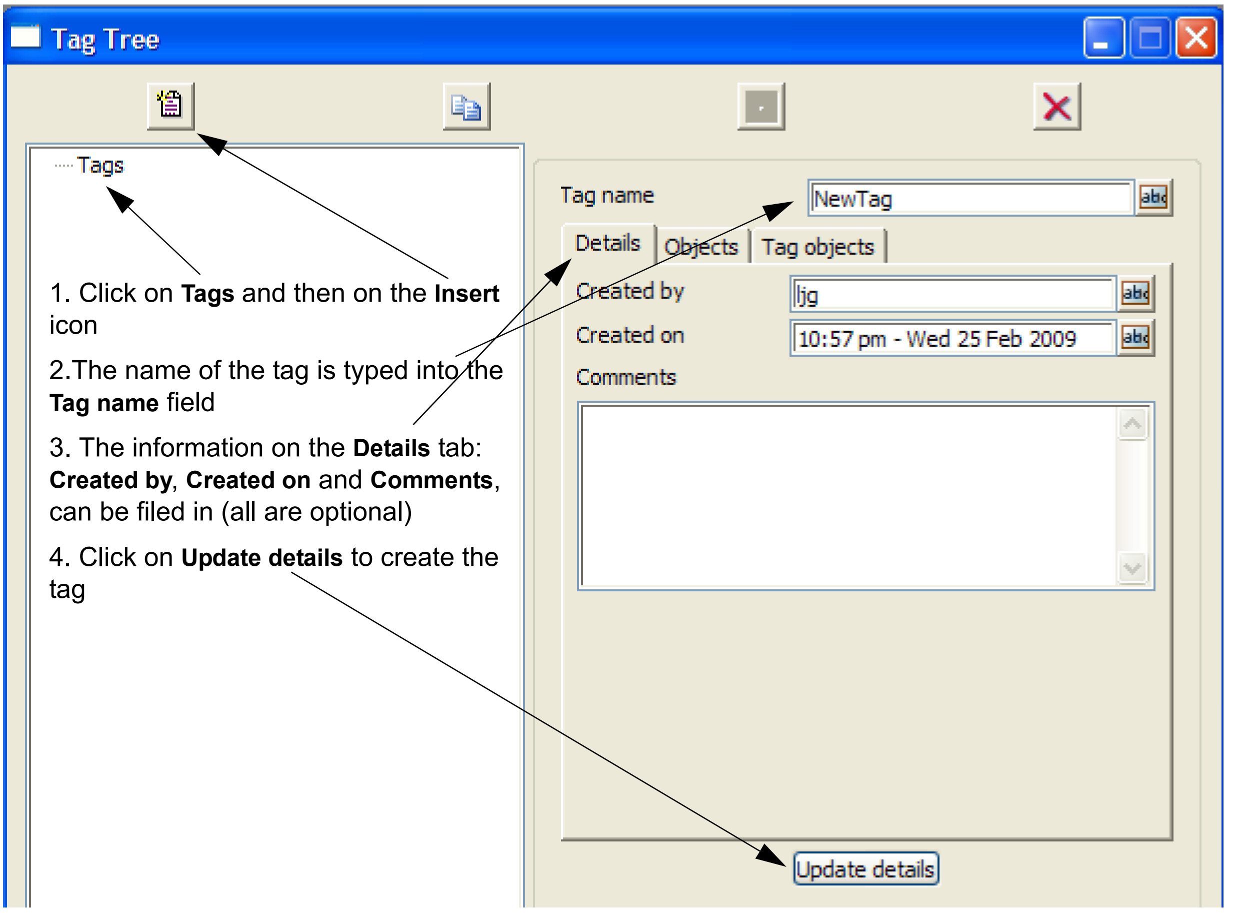
*write out the tag tree definition for this project to the **Tag file**.*

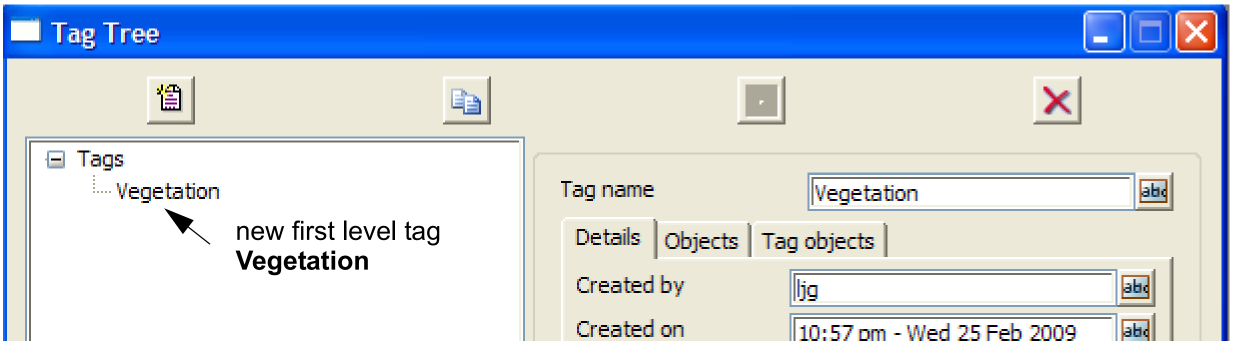
How To Use the Tag Tree Panel

Creating a Tag

To create a ***new first level tag***, click on ***Tags*** and then the **Insert** icon.

The name of the tag is typed into the **Tag name** field, the information on the **Details** tab: **Created by**, **Created on** and **Comments**, can be filled in (all are optional) and then click on **Update details** to create the tag.





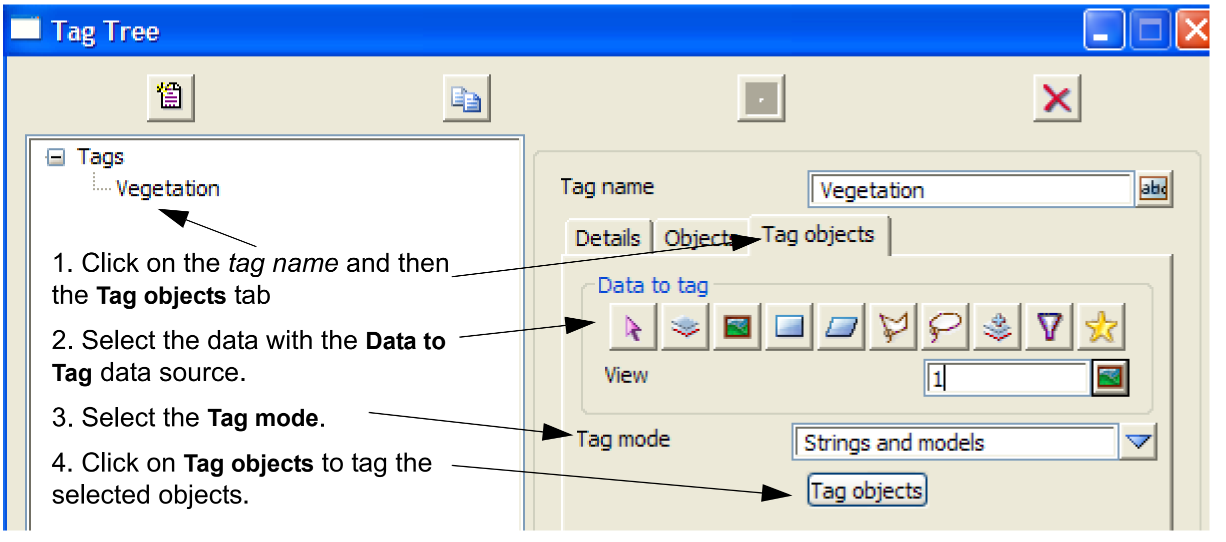
Applying a Tag To Objects

To apply a tag, click on the *tag name* and then the **Tag Objects** tab.

Select the data with the **Data to Tag** data source.

Select the **Tag mode** (choice of *Strings only*, *Models only*, *Strings and models*).

Click on **Tag objects** to tag the selected objects.



Retrieving and Deleting Tagged Objects

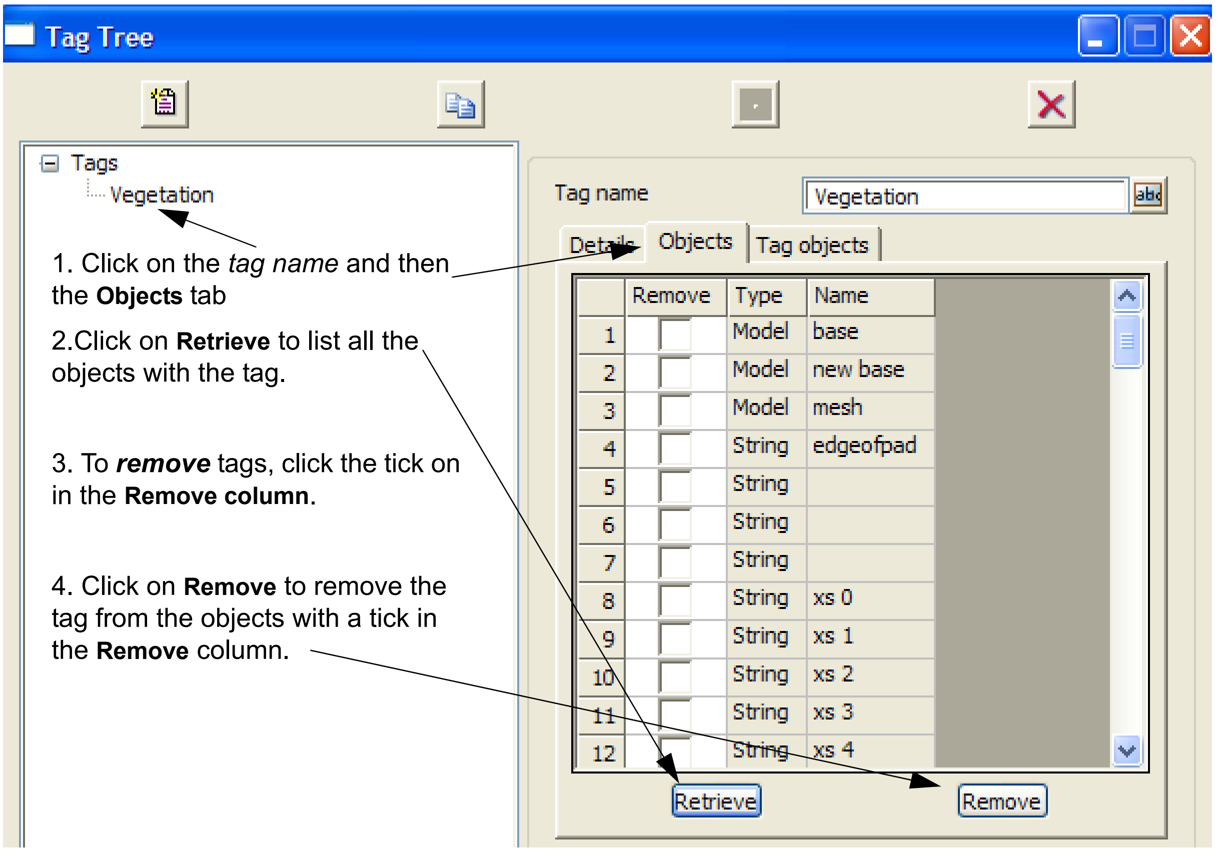
To retrieve a list of data tagged with a certain tag, click on the *tag name* and then the **Objects** tab.

Click on **Retrieve** to list all the objects with the tag.

To remove tags, click the tick on in the **Remove column**.

Click on **Remove** to remove the tag from the ticked objects.

Note - clicking RB on the ***Remove*** at the *top of the column* in the grid brings up a panel with options to set all the ticks on, set all the ticks off or toggle the tick state.



Defining a Tag Tree (Tag Hierarchy)

Tag names are ***case sensitive*** and must be ***unique*** *and* objects can be tagged with **more than one** tag. For example, a string may be tagged with the two tags, "survey day 1" and "sewer"

To further increase their usefulness, tags in a project can also be assigned a ***hierarchy***, or ***tree structure***.

A ***tag tree structure*** can be created for the project by following the rules:

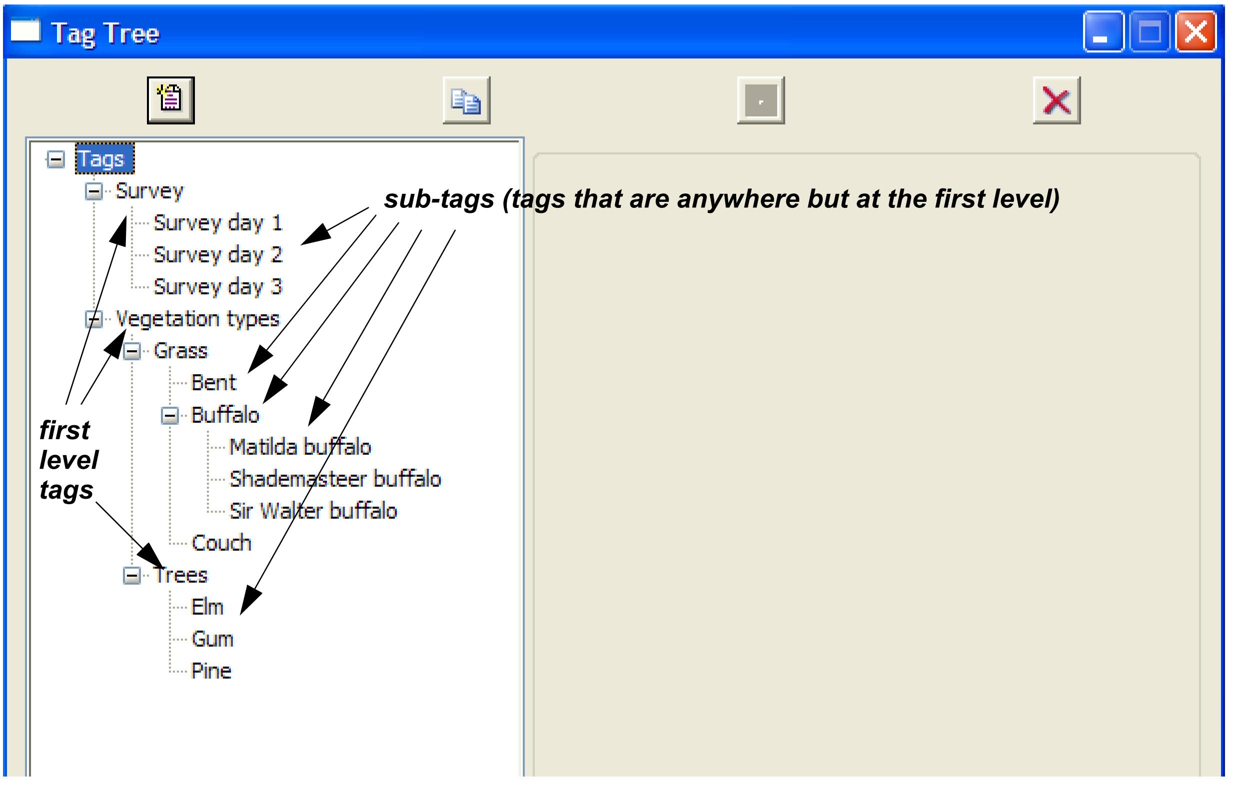
1. Tags can be ***first level tags*** or ***sub-tags***, but not both.
2. ***First level tags*** are tags created at the top level of the tag tree. That is, they are the level directly below the ***Tags*** label in the ***Tag Tree*** panel.
3. ***Sub-tags*** are tags that are created as sub-tags *of a first level tag*, or *sub-tags of other sub-tags*
4. First level tags can not be used as sub-tags and sub-tags can not be a first level tag.
5. ***Tag names*** must be ***unique*** amongst ***all tags - first level tags*** and ***sub-tags***. So you can't have a first level tag with the same name as a sub-tag, and vice-versa.
6. The same subtag name can not appear more than once in any single ***tag-subtag*** sequence.

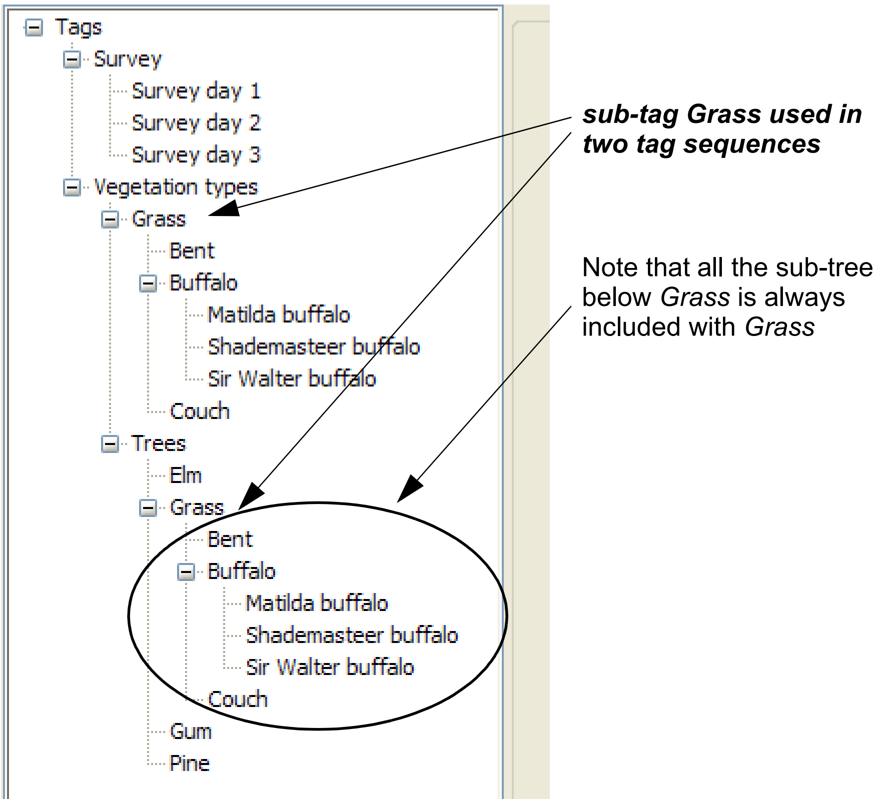
   For example, you can have a tag *Vegetation types* with a subtag of *Grass* and with a sub-subtag of *Bent*. but you can't have a tag *Vegetation types* with a subtag of *Grass,* a sub-subtag of *Bent*, and a sub-sub-subtag of *Grass*.

   That is *Vegetation types > Grass > Bent* is allowed
   but *Vegetation types > Grass >Bent > Grassis **not** allowed*

   Note that because tags are case sensitive

   Vegetation types > Grass >Bent > grass is allowed.
7. A subtag can appear in more than one tag sequence as long as Rule 6 is not violated.





A tag hierarchy can be very useful.

For example, if we have the tags and tag tree as defined in the previous picture, adding the models tagged as *Trees* with the **Add Tagged Models** option will also add the models tagged with *Elm*, Gum, *Pine*, Grass, *Bent, Buffalo, Matilda buffalo*, *Shademasteer buffalo, Sir Walter buffalo* and *Couch* ***even if they are not tagged as Trees.***

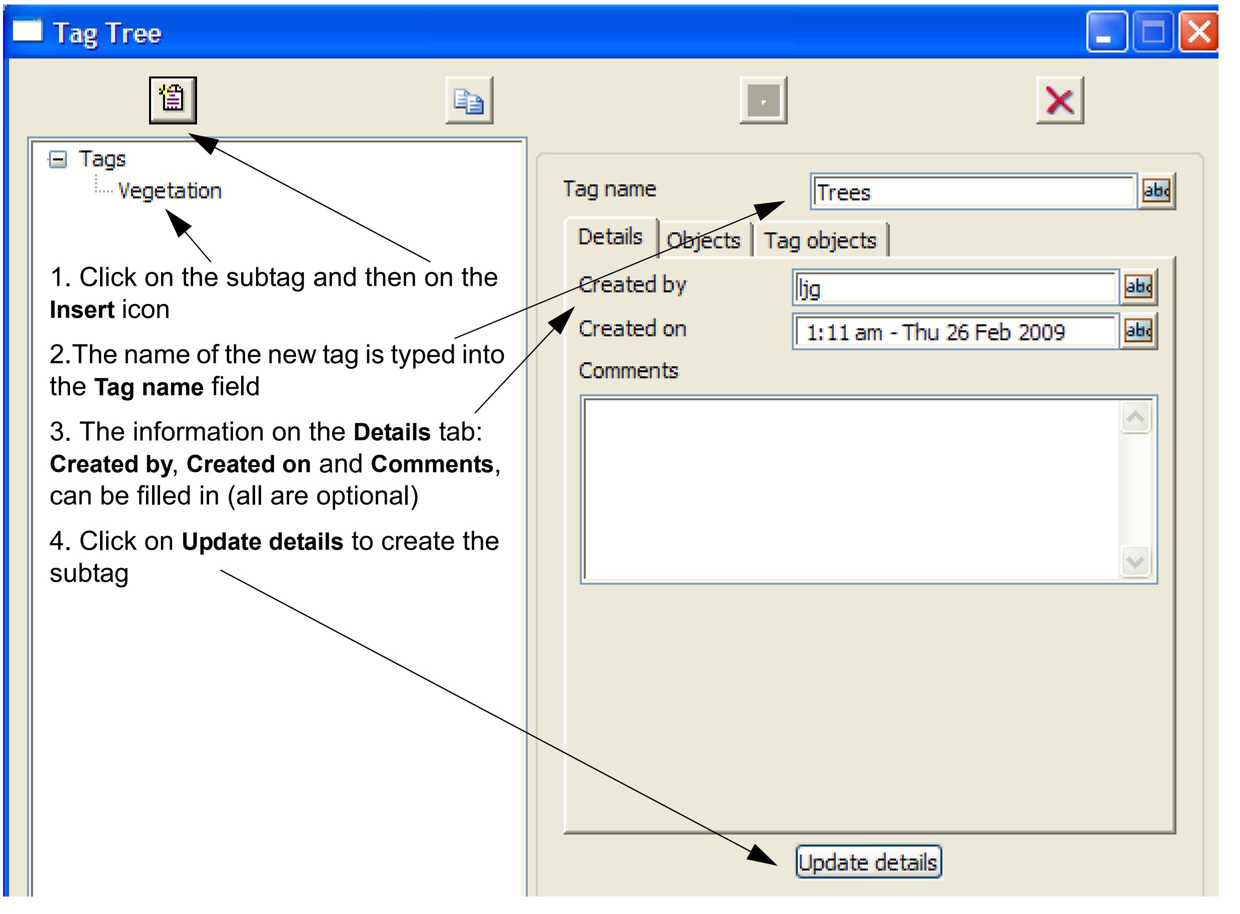
If *Grass* was removed as a subtag of *Trees* then adding *Trees* with the **Add Tagged Models** option will then only add the models tagged with *Trees*, *Elm*, Gum and *Pine.* So no new tagging was needed, just a modification to the tag tree definition.

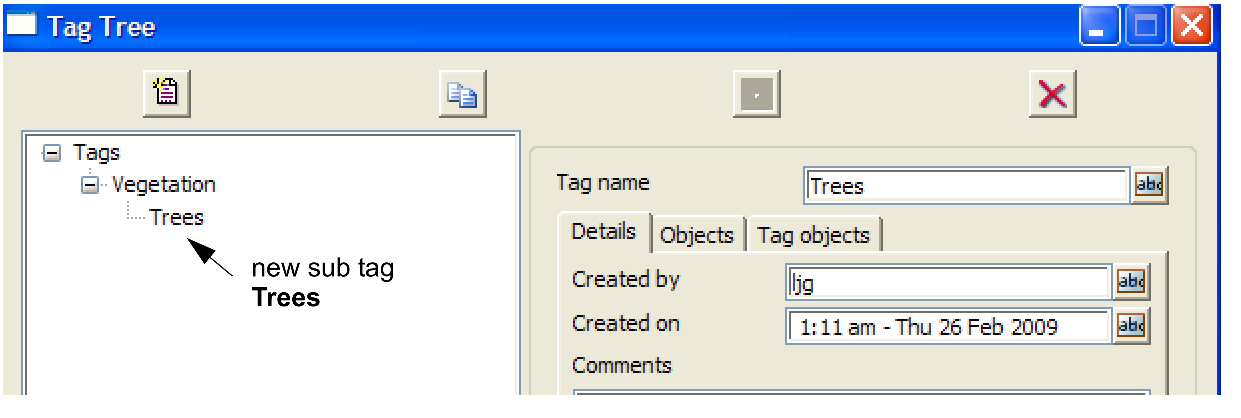
Creating a Tag Tree Structure

To create a subtag of any tag is very similar to creating a first level tag.

To create a ***new sub tag***, click on the tag to have the subtag, and then click on the **Insert** icon.

The name of the subtag is typed into the **Tag name** field, the information on the **Details** tab: **Created by**, **Created on** and **Comments**, can be filled in (all are optional) and then click on **Update details** to create the subtag.


Tag Tree
Tags
Vegetation
1. Click on the subtag and then on the **Insert** icon
2.The name of the new tag is typed into the **Tag name** field
3. The information on the **Details** tab: **Created by**, **Created on** and **Comments**, can be filled in (all are optional)
4. Click on **Update details** to create the subtag
Tag name
Trees
Details
Objects
Tag objects
Created by
ljg
Created on
1:11 am - Thu 26 Feb 2009
Comments
Update details

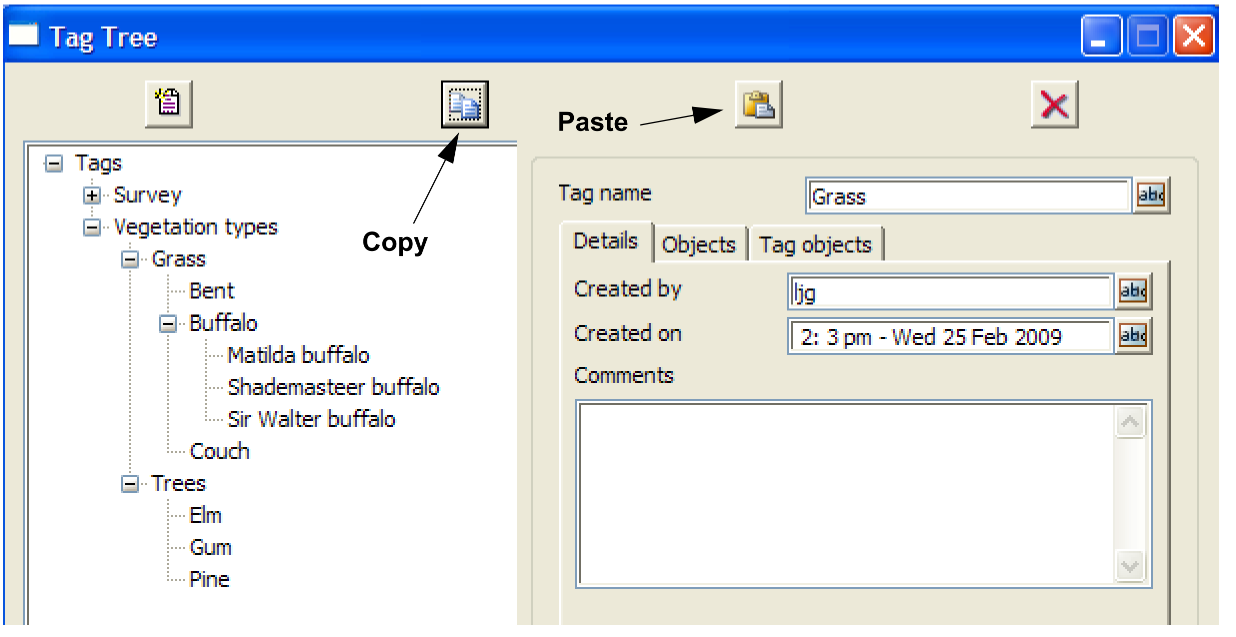

Tag Tree
Tags
Vegetation
Trees
new sub tag **Trees**
Tag name
Trees
Details
Objects
Tag objects
Created by
ljg
Created on
1:11 am - Thu 26 Feb 2009
Comments

Copy and Paste in a Tag Tree Structure

The **Copy** and **Paste** icons are used to copy subtags around the tag tree structure.

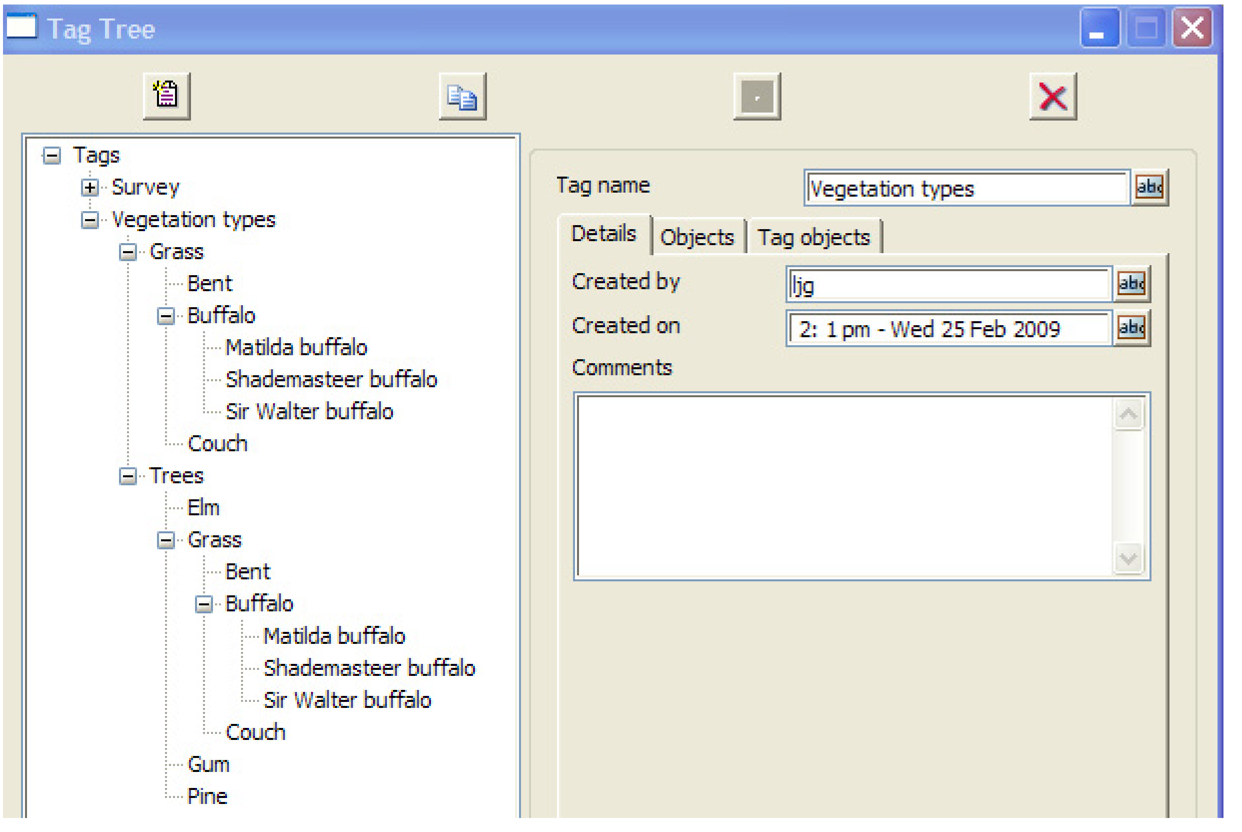
To copy, first click on the subtag you wish to copy (say Grass), and then click on the **Copy** icon.

The **Paste** button will then become active.



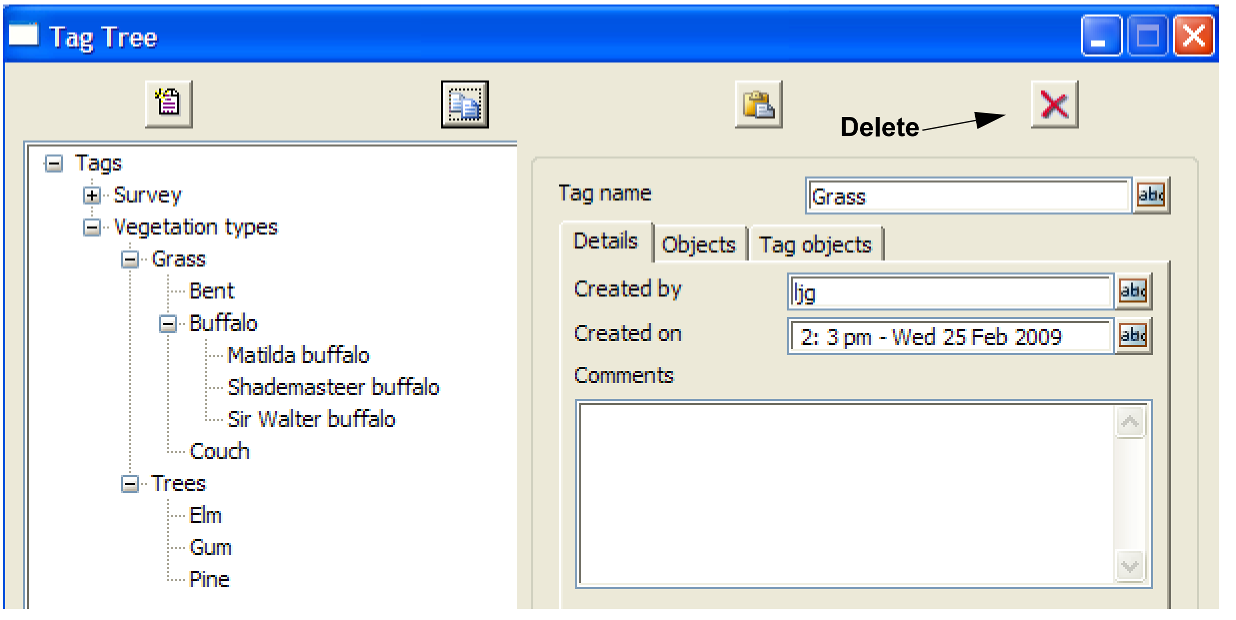
Now click on the first level tag or subtag to copy the subtag to (say *Trees*), and then click on the **Paste** icon.

The subtag (and the tree below it) will now be copied as a subtree of the selected tag.


Tag Tree
Tags
Survey
Vegetation types
Grass
Bent
Buffalo
Matilda buffalo
Shademasteer buffalo
Sir Walter buffalo
Couch
Trees
Elm
Grass
Bent
Buffalo
Matilda buffalo
Shademasteer buffalo
Sir Walter buffalo
Couch
Gum
Pine
Tag name
Vegetation types
Details
Objects
Tag objects
Created by
ljg
Created on
2: 1 pm - Wed 25 Feb 2009
Comments

Deleting a Subtag in Tag Tree Structure

A first level tag or subtag can be deleted from anywhere in the tree by simply clicking on the tag to be deleted and then clicking on the **Delete** icon.



Go to the next section 7.6.10.2 Register New Tags or return to 7.6.10 Tags.

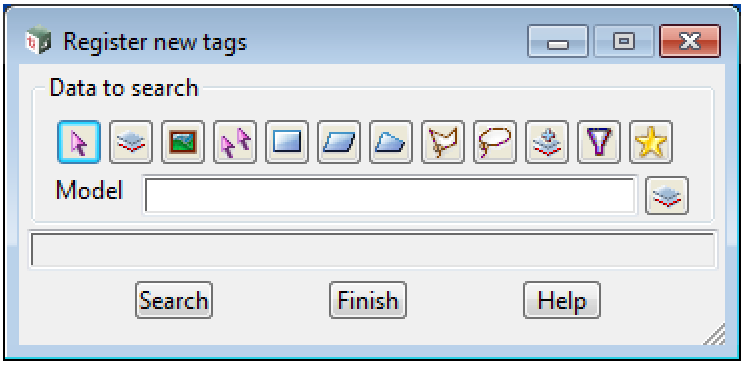
7.6.10.2 Register New Tags

Position of option on menu: **Project =>Management =>Tags => Find new tags**

Register new tags is used to locate any new tags that have been added on models or strings which have not been registered with the project.

Unregistered tags may occur when reading in a 12da that contains models or strings with tags. Until a tag is registered, it can not be used in source boxes or other panels.

Selecting **Find new tags**, brings up the **Register new tags** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data to search	source		

Data to search - for a full description go to 4.19.3 Data Source.

Search	button		

perform the search.

Return to 7.6.10 Tags.

7.6.11 Tree

Position of option on menu: **Project =>Tree**

Position of option on menu: **Project =>Management =>Tree**

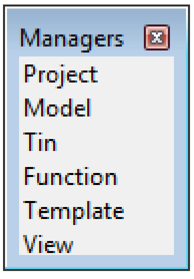
For documentation on this option, go to the section **Project =>Tree** (see 7.9 Tree).

7.6.12 Managers

Position of option on menu: **Project =>Management =>Managers**

The **Managers** display in a tree structure information and settings for all the models, tins, functions and templates both in the project and removed from the project, and for all the views in the project.

The **Managers** walk-right menu contains a manager for the entire project and separate managers for modes, tins, functions templates and views.



display nodes for all models, tins, functions & templates (in & removed from the project), & all views in the project

For the option:

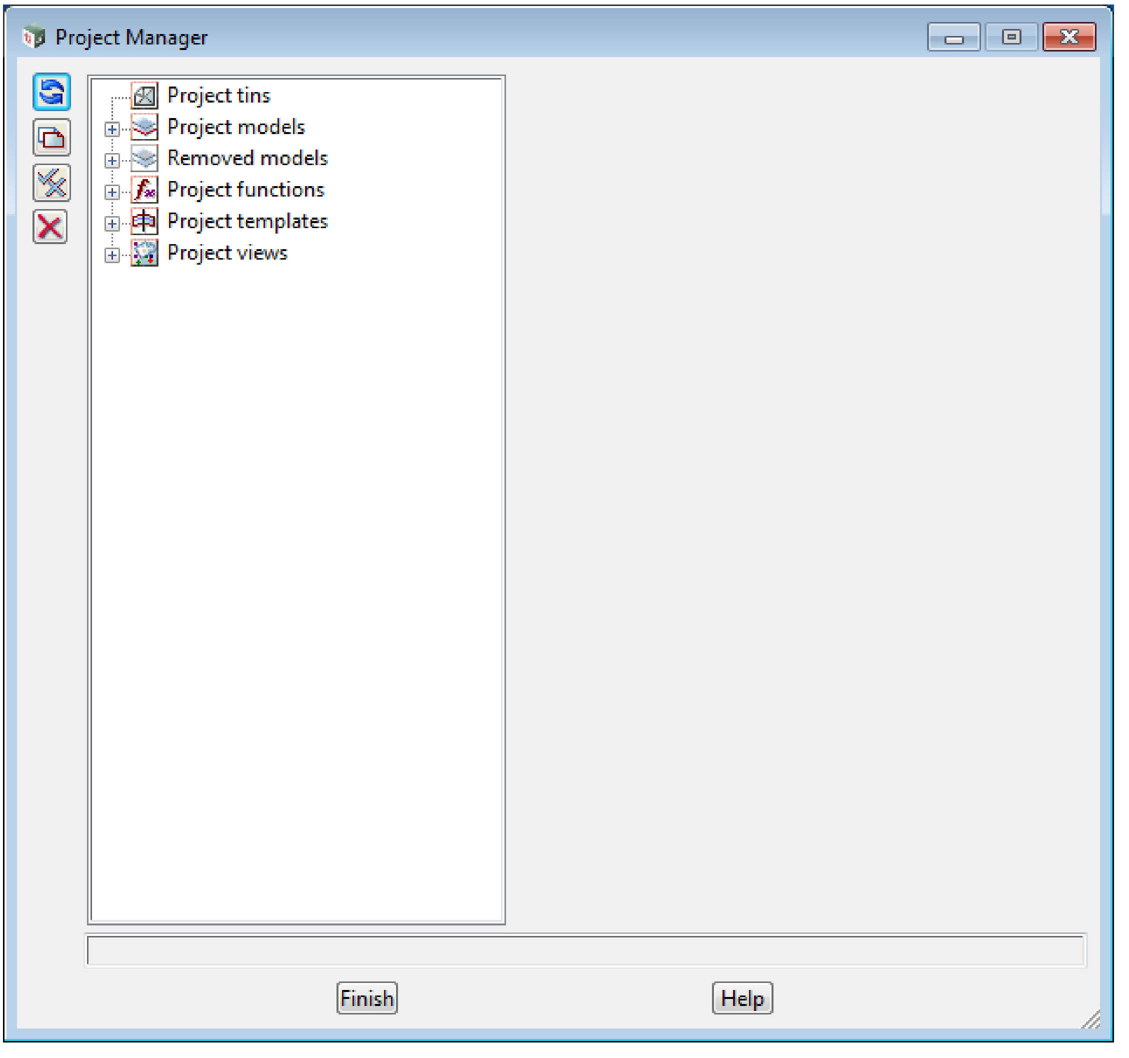
Project, go to 7.6.12.1 Project Manager.
Model 7.6.12.2 Model Manager.
Tin 7.6.12.3 Tin Manager.
Function 7.6.12.4 Function Manager.
Template 7.6.12.5 Template Manager.
View 7.6.12.6 View Manager.

7.6.12.1 Project Manager

Position of option on menu: **Project =>Management =>Managers =>Project**

The **Project Manger** displays in a tree structure nodes for all the models, tins, functions and templates both in the project and removed from the project, and all the views in the project.

Selecting **Project** displays the **Project Manager** panel.



If there are any models/tins/functions/templates in the Project, then there will be a **Project models/tins/functions/templates** node in the tree and that node has sub nodes for all the models/tins/functions/templates in the project. The name of the sub node is the name of the model/tin/function/template.

If there are any models/tins/functions/templates that have been removed from the Project, then there will be a **Removed models/tins/functions/templates/views** node in the tree and that node has sub nodes for all the removed models/tins/functions/templates. The name of the sub node is the name of the removed model/tin/function/template

If there are any views in the Project, then there will be a **Project view** node in the tree with the views as sub nodes. The name of sub node is the name of the view.

The information and setting displayed for these nodes is the same as those in the separate Model/Tin/Function/Template/View Managers and so will be documented in those sections.

See

7.6.12.2 Model Manager

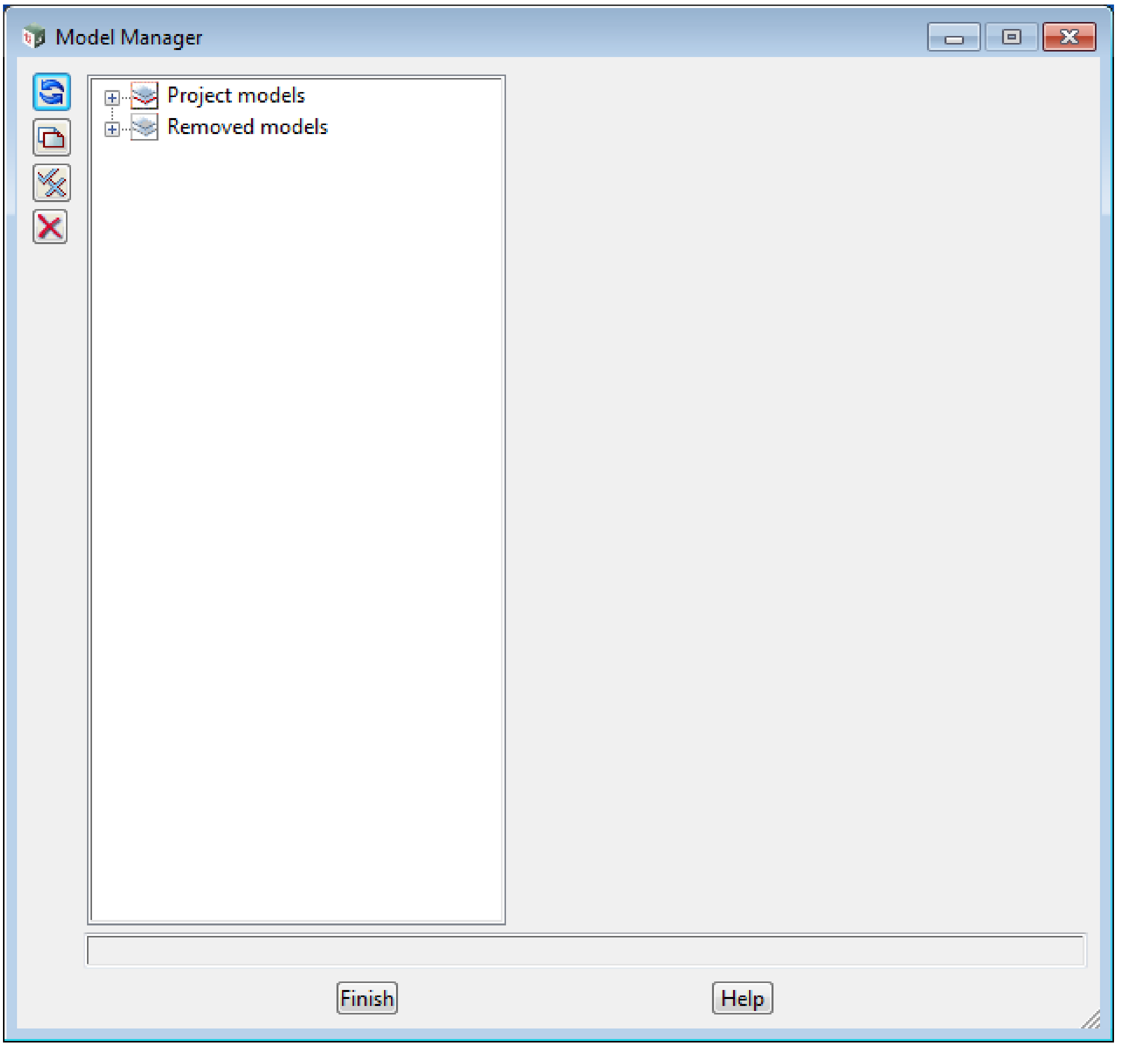
Continue to the next section 7.6.12.2 Model Manager or return to 7.6.12 Managers.

7.6.12.2 Model Manager

Position of option on menu: **Project =>Management =>Managers =>Model**

The **Model Manger** displays in a tree structure all the settings for models in the project, and the names of all the removed models.

Selecting **Model** displays the **Model Manager** panel.



If there are any models in the Project, then there will be a **Project models** node in the tree.

Expanding the **Project models** node lists all the models in the project as sub nodes, and expanding the sub node of a project model displays information and settings for the model on the right hand side of the panel. The name of the sub node is the name of the model.

Many of the settings can then be changed, and the changes take place immediately without having to press any other button.

If there are any models that have been removed from the Project, then there will be a **Removed models** node in the tree.

Expanding the **Removed models** node lists all the models removed from the project as sub nodes but no other information or settings can be displayed for removed models. The name of the sub node is the name of the removed model.

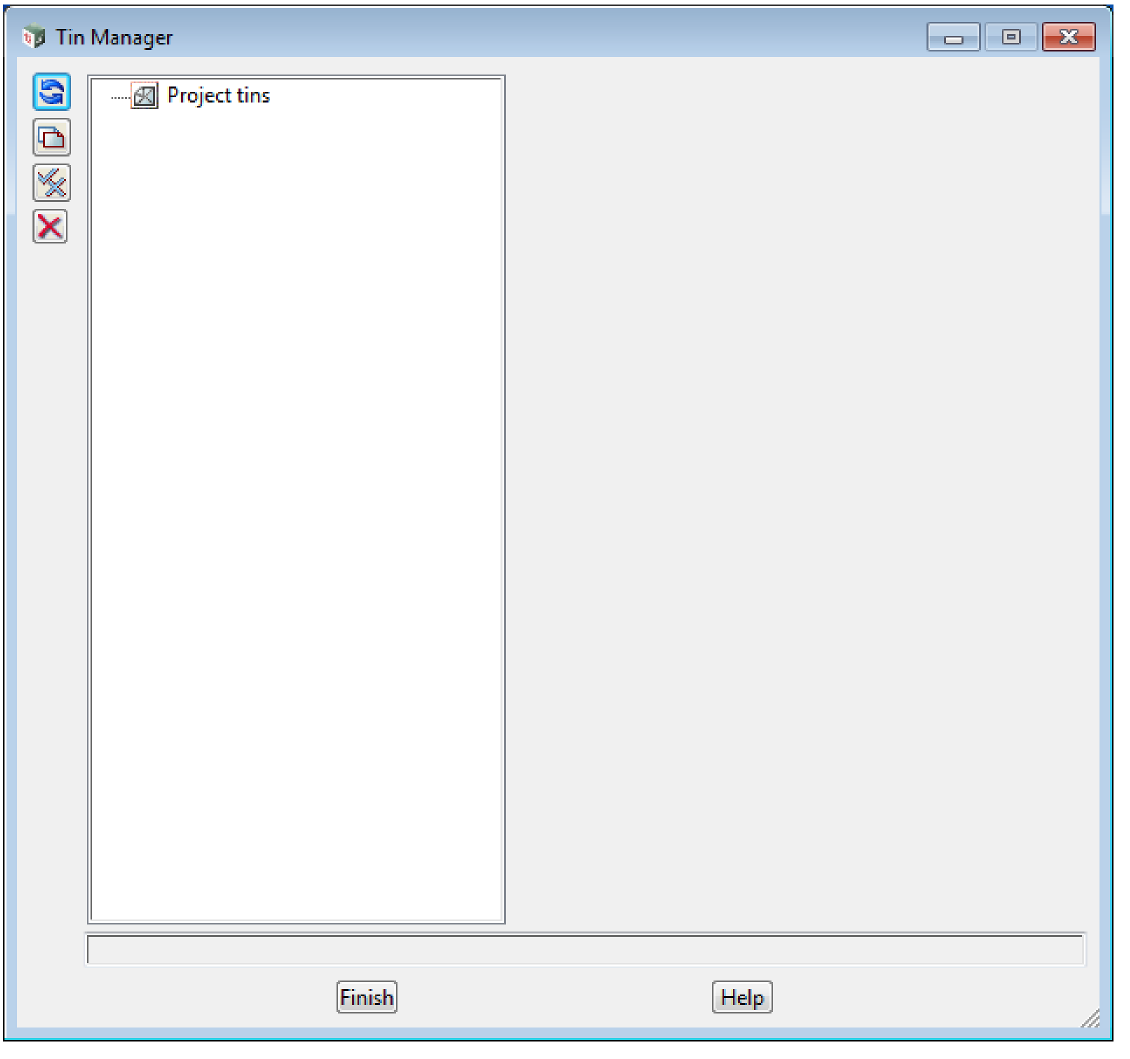
Continue to the next section 7.6.12.3 Tin Manager or return to 7.6.12 Managers.

7.6.12.3 Tin Manager

Position of option on menu: **Project =>Management =>Managers =>Tin**

The **Tin Manger** displays in a tree structure, all the settings for tins in the project, and the names of all the removed tins.

Selecting **Tin** displays the **Tin Manager** panel.



If there are any tins in the Project, then there will be a **Project tins** node in the tree.

Expanding the **Project tins** node lists all the tins in the project as sub nodes, and expanding the sub node of a project tin displays information and settings for the tin on the right hand side of the panel. The name of the sub node is the name of the tin.

Many of the settings can then be changed, and the changes take place immediately without having to press any other button.

If there are any tins that have been removed from the Project, then there will be a **Removed tins** node in the tree.

Expanding the **Removed tins** node lists all the tins removed from the project as sub nodes but no other information or settings can be displayed for removed tins. The name of the sub node is the name of the removed tin.

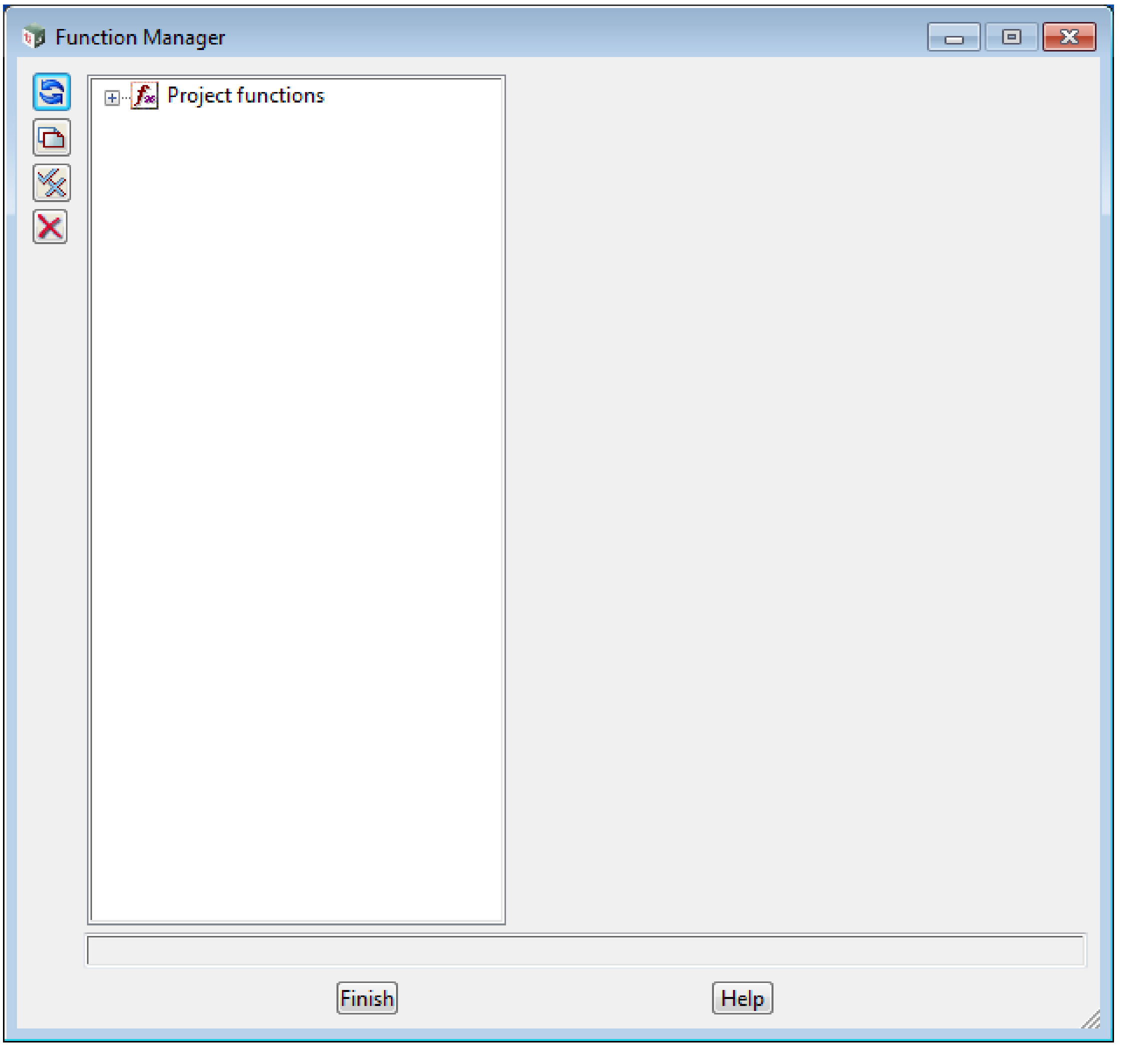
Continue to the next section 7.6.12.4 Function Manager or return to 7.6.12 Managers.

7.6.12.4 Function Manager

Position of option on menu: **Project =>Management =>Managers =>Function**

The **Function Manger** displays in a tree structure, all the settings for functions in the project, and the names of all the removed functions.

Selecting **Function** displays the **Function Manager** panel.



If there are any functions in the Project, then there will be a **Project functions** node in the tree.

Expanding the **Project functions** node lists all the functions in the project as sub nodes, and expanding the sub node of a project function displays information and settings for the function on the right hand side of the panel. The name of the sub node is the name of the function.

Many of the settings can then be changed, and the changes take place immediately without having to press any other button.

If there are any functions that have been removed from the Project, then there will be a **Removed functions** node in the tree.

Expanding the **Removed functions** node lists all the functions removed from the project as sub nodes but no other information or settings can be displayed for removed functions. The name of the sub node is the name of the removed function.

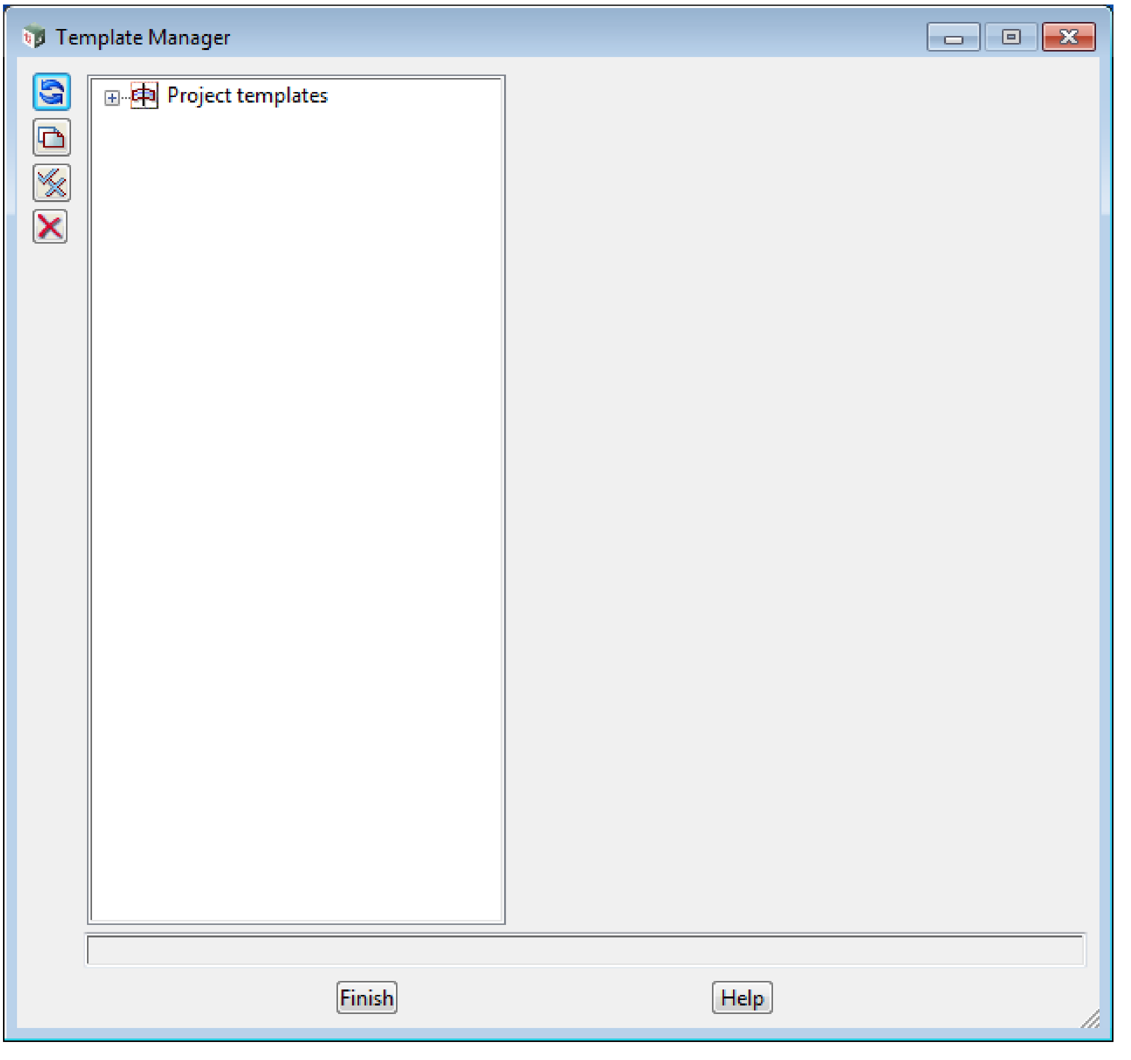
Continue to the next section 7.6.12.5 Template Manager or return to 7.6.12 Managers.

7.6.12.5 Template Manager

Position of option on menu: **Project =>Management =>Managers =>Template**

The **Template Manger** displays in a tree structure, all the settings for templates in the project, and the names of all the removed templates.

Selecting **Template** displays the **Template Manager** panel.



If there are any templates in the Project, then there will be a **Project templates** node in the tree.

Expanding the **Project templates** node lists all the templates in the project as sub nodes, and expanding the sub node of a project template displays information and settings for the template on the right hand side of the panel. The name of the sub node is the name of the template.

Many of the settings can then be changed, and the changes take place immediately without having to press any other button.

If there are any templates that have been removed from the Project, then there will be a **Removed templates** node in the tree.

Expanding the **Removed templates** node lists all the templates removed from the project as sub nodes but no other information or settings can be displayed for removed templates. The name of the sub node is the name of the removed template.

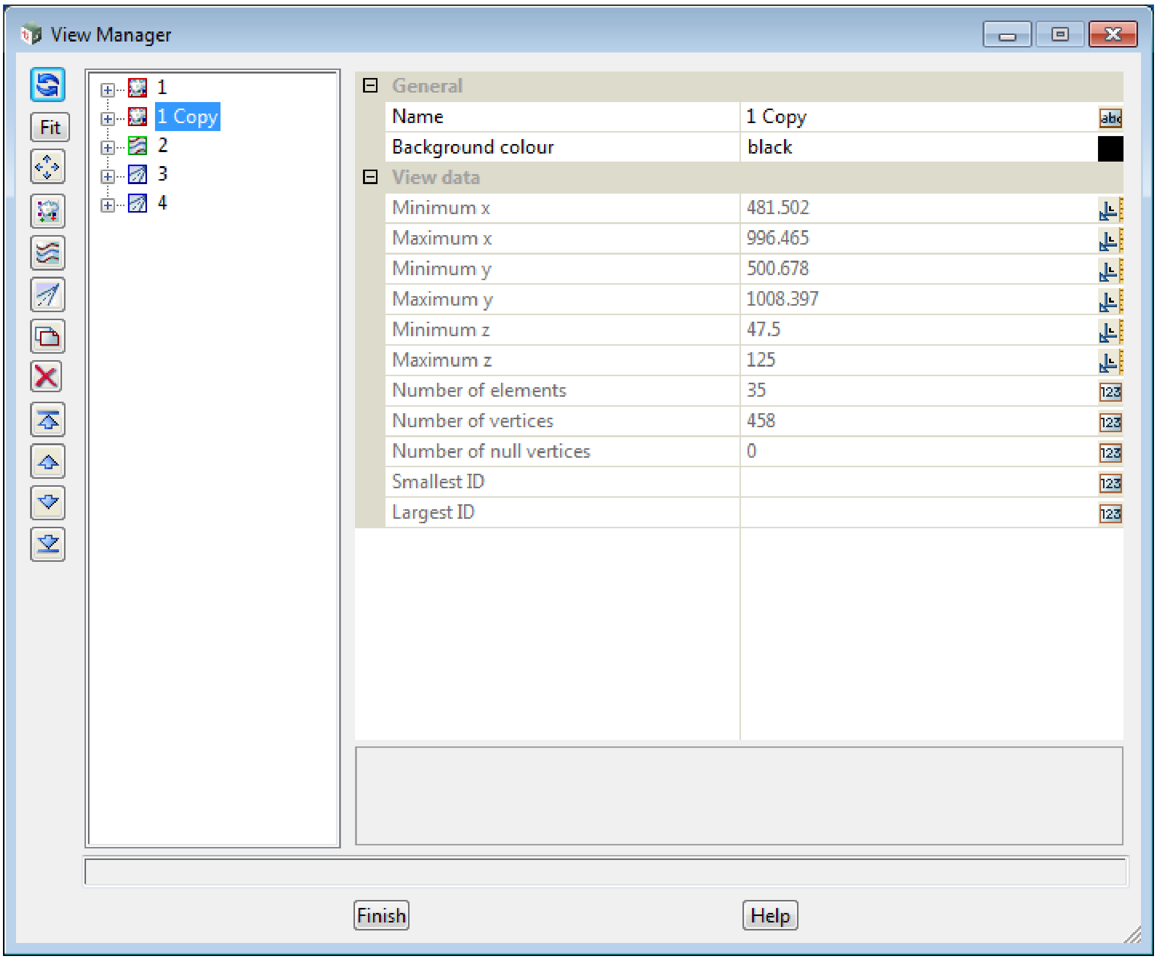
Continue to the next section 7.6.12.6 View Manager or return to 7.6.12 Managers.

7.6.12.6 View Manager

Position of option on menu: **Project =>Management =>Managers =>View**

The **View Manger** displays in a tree structure, all the settings for views in the project.

Selecting **View** displays the **View Manager** panel.



If there are any views in the Project, then there will be a node (**view node**) in the tree with the **name of the view** as the **node name**.

Expanding a **view node** lists information and setting for that view as sub nodes, and displays values for the nodes and sub nodes in the right hand side of the panel.

Once displayed in the right had side, many of the settings can be changed, and the changes will take place immediately without having to press another button.

For each **view node**, the sub nodes, settings and information displayed is the same as for the **View Properties** panels. See 9.4.6 Properties.

Return to 7.6.12 Managers.

7.6.13 Sharing

See

7.6.13.1 Sharing of Models and Tins

7.6.13.1 Sharing of Models and Tins

Models and Tins can be **shared into your project** from other projects. That means that in your project, the models an tins that are shared in from another project are not created in your project, but are copies of the models and tins from the other project.

And when the models and/or tins are modified in the other project, they can be **automatically updated** in your project.

In your own project, you decide which models and tins are available to be shared into other peoples projects. That is, you say which or your models and tins can be **shared out to another project**.

It is possible to have Shares of Shares. That is, you share a model or tin **into your project**, and then that model or tin is **shared out to another project**.

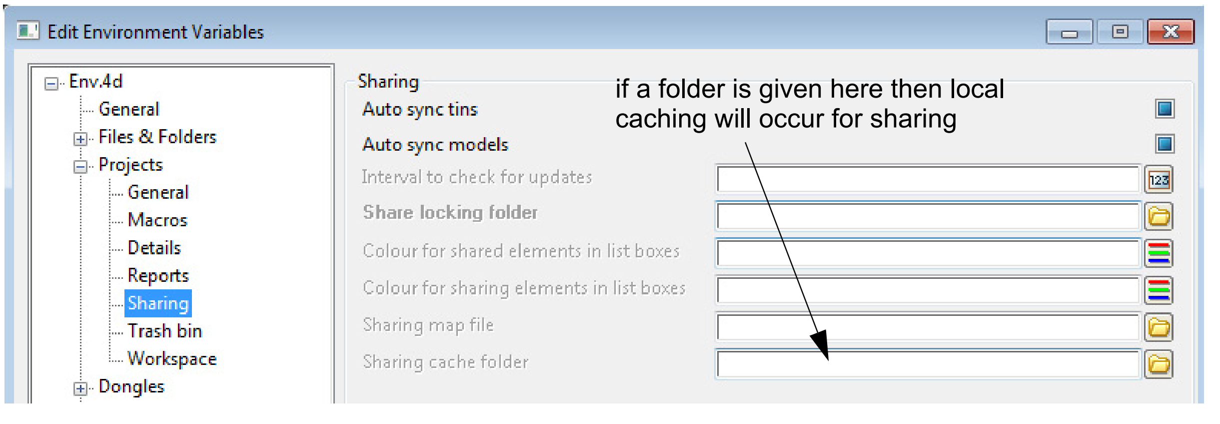
When models and tins are being shared into your project, a copy of the models and tins are copied over into your project. And each time one of the models or tins is modified in the original project, a new copy is made in your project.

However when you exit a project, the next time you start the project the models and tins that are shared into your project may not have changed. Rather than copying the models and tins again, it is possible to have the models and tins saved in a special folder, called a **Sharing Cache Folder**, so they do not have to be copied over again when ever a project is restarted.

There is an environment variable, SHARING_CACHE_4D, which gives the full path name to the local folder to use as the ***Sharing Cache Folder***.

In the ***Edit Environment Variables*** panel, it is the field ***Sharing cache folder*** in

Projects >Sharing.

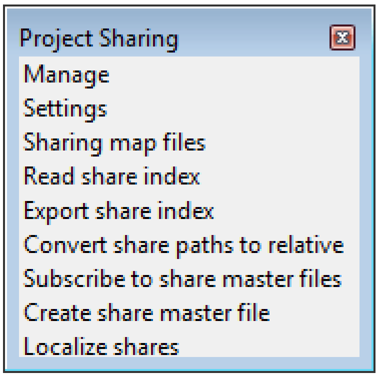


If ***Sharing cache folder*** is left blank then local caching does not occur.

Go to the next section 7.6.13.2 Project Sharing or return to 7.6.13 Sharing.

7.6.13.2 Project Sharing

The **Sharing** walk-right menu is:



For information about sharing, go to 7.6.13.1 Sharing of Models and Tins.

For the options:

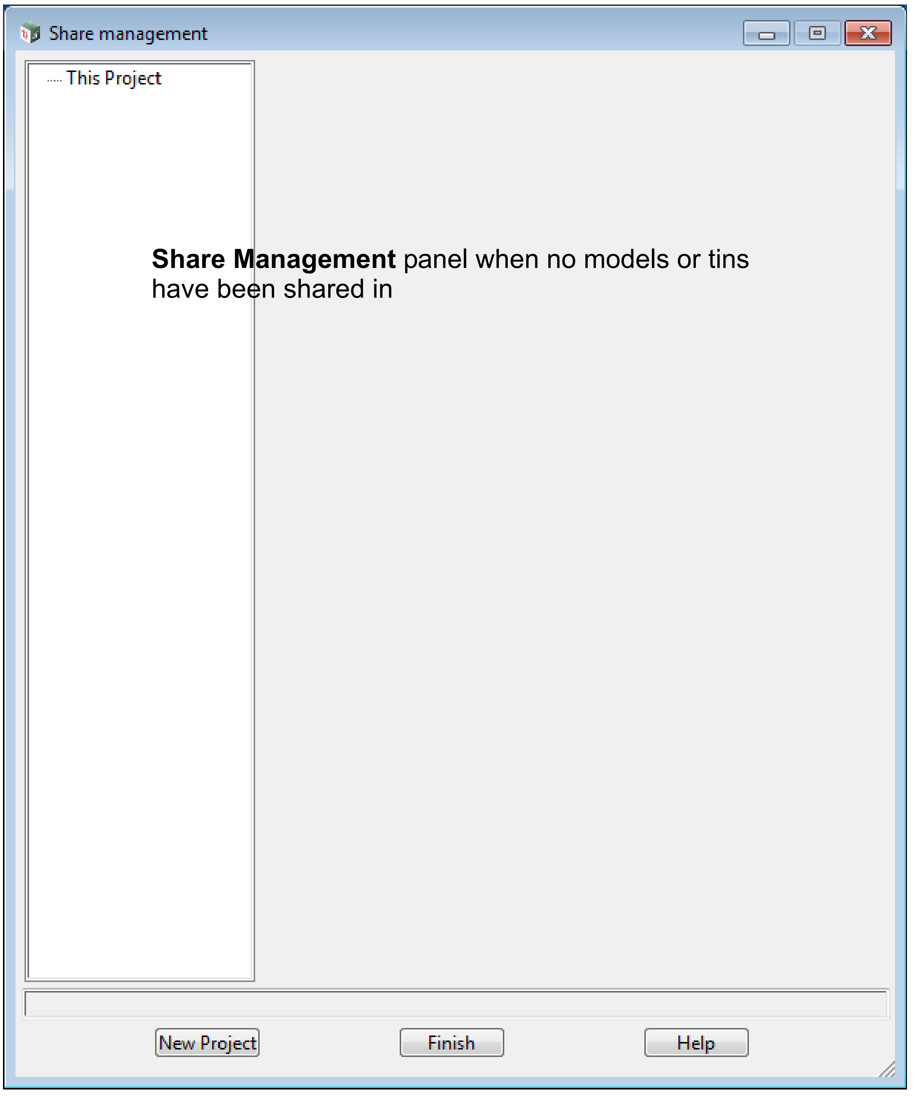
Manage, please go to 7.6.13.3 Share Management
Settings 7.6.13.4 Project Share Settings
Sharing map files 7.6.13.5 Sharing Map Files
Read share index 7.6.13.6 Add Shares By Index File
Export share index 7.6.13.7 Export A Share Index File
Convert share paths to relative 7.6.13.8 Convert Share Paths To Relative
Subscribe to share master files 7.6.13.9 Subscribe to Sharing Master Files
Create share master file 7.6.13.10 Sharing Master File
Localize shares 7.6.13.11 Localize Shares

7.6.13.3 Share Management

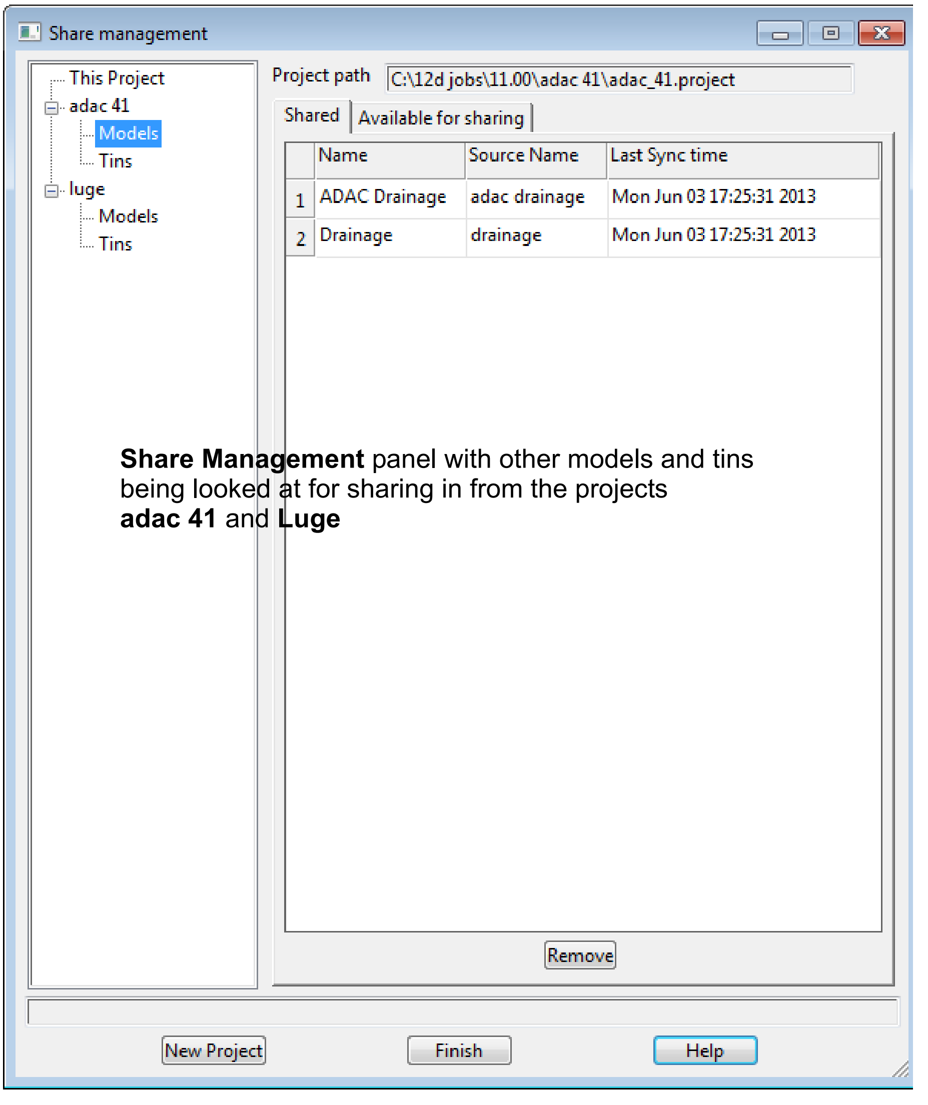
Position of option on menu: **Project =>Management =>Sharing =>Manage**

The **Share Management** tool combines all the **Share**, **Add** and **Remove** options for the sharing of models and tins in and out.

Selecting **Manage** brings up the **Share Management** panel and the panel will list under the **This Project** node, all the models and tins in the project that are available to share out, and as extra as nodes, all the projects that models and tins have been **shared in** from.



Share Management panel when no models or tins have been shared in



Share Management panel with other models and tins being looked at for sharing in from the projects **adac 41** and **Luge**

See

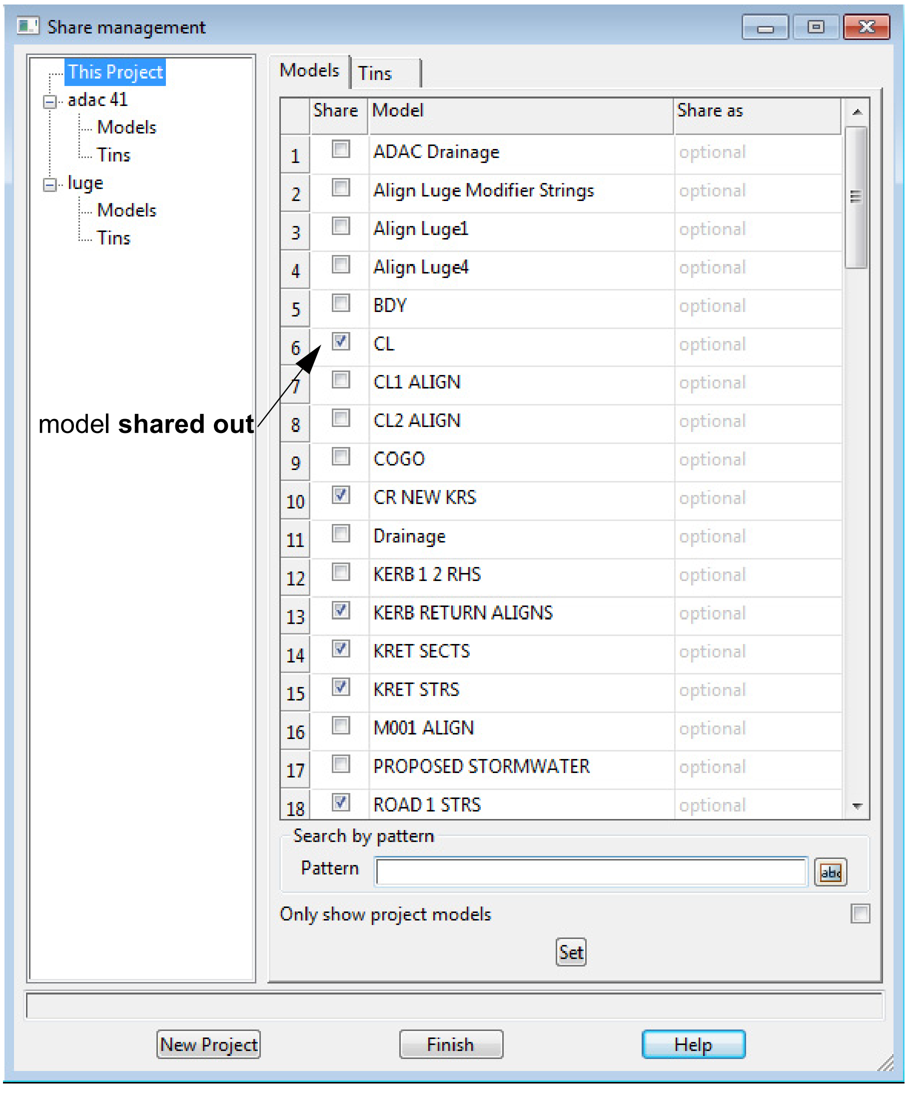
7.6.13.3.1 Making Models or Tins Available for Sharing

7.6.13.3.2 Sharing in Models and Tins

7.6.13.3.1 Making Models or Tins Available for Sharing

Clicking on **This Project** displays all the models in the project on the **Models** tab, and all the tins in the project under the **Tins** tab.

Models or tins that have been made available for sharing, that is are **shared out** so other projects can share in these models and tins from this project, have a tick beside them in the **Set** column.



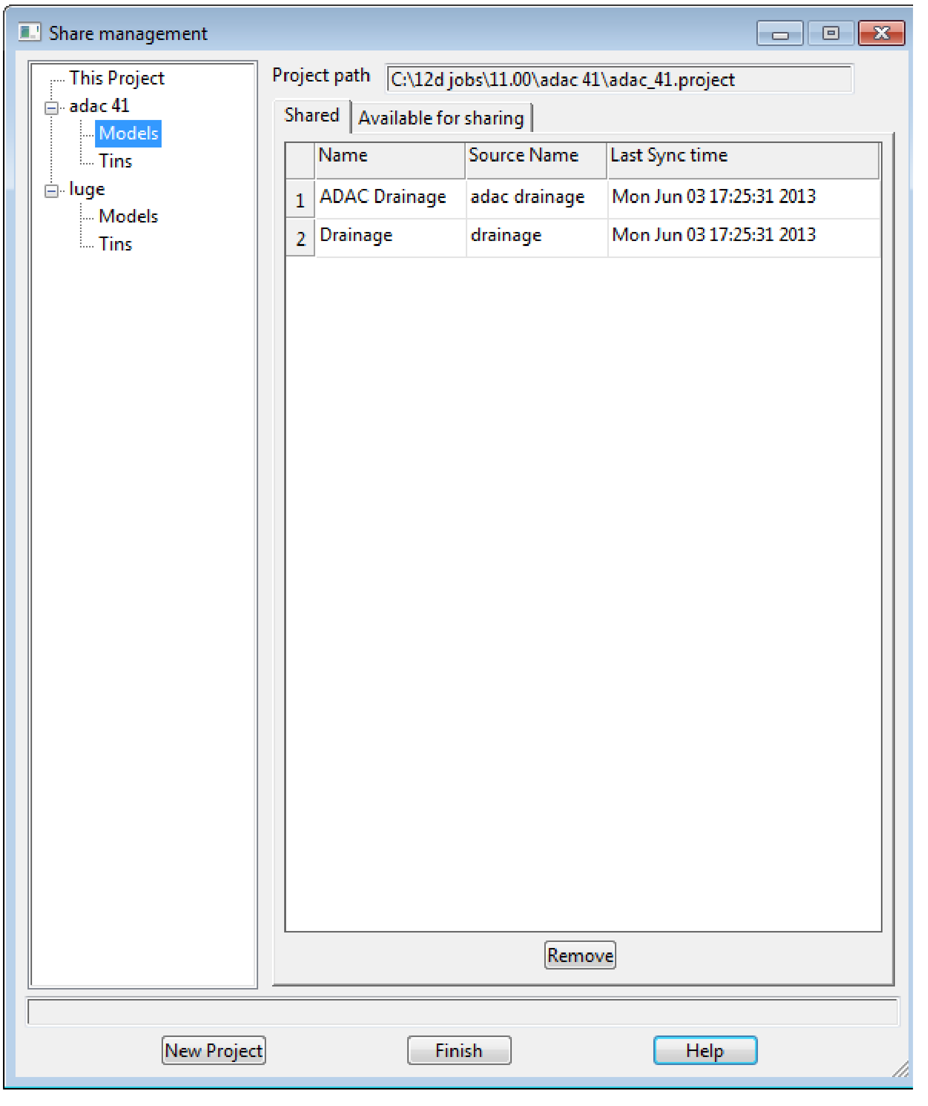
To share models or tins out, simply click the tin on in the **Share Out** column and then click on **Set**.

Similarly if there is no tick in the **Share** column then clicking on **Set** will make the models or tins no longer available to share out.

Go to the next section 7.6.13.3.2 Sharing in Models and Tins or return to 7.6.13.3 Share Management.

7.6.13.3.2 Sharing in Models and Tins

Under each project node, there is a **Models** and **Tins** node.



Models

Clicking on the **Models** node lists all the models that have been shared in on the **Shared** tab, and all the models that are available to be shared in from that project on the **Available for Sharing** tab.

On the **Shared** tab, highlighting a model name and clicking on the **Remove** button will stop the model from being shared in.

On the **Available Sharing** tab, all models available for sharing in are shown and those that have already been shared in are coloured yellow. Models that are not shared in can be shared in by ticking in the **Share ?** column and then clicking the **Add** button.

Tins

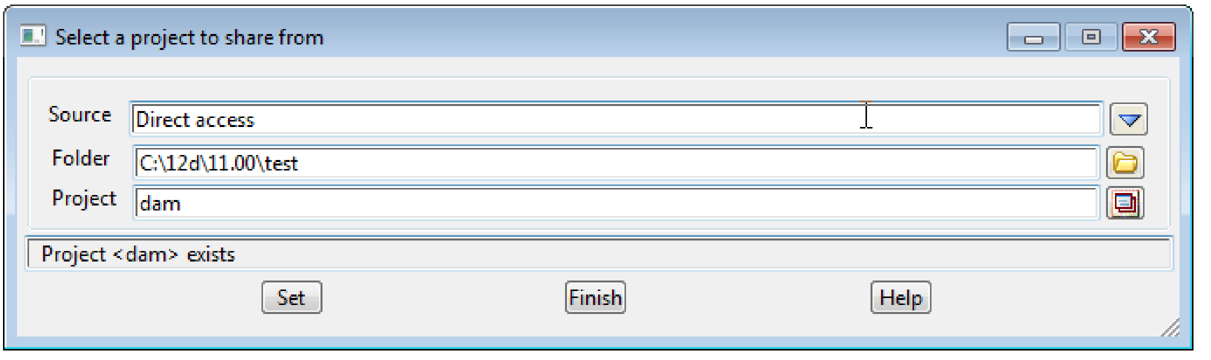
Clicking on the **Tins** node lists all the tins that have been shared in on the **Shared** tab, and all the tins that are available to be shared in from that project on the **Available for Sharing** tab.

On the **Shared** tab, highlighting a tin name and clicking on the **Remove** button will stop the ting from being shared in.

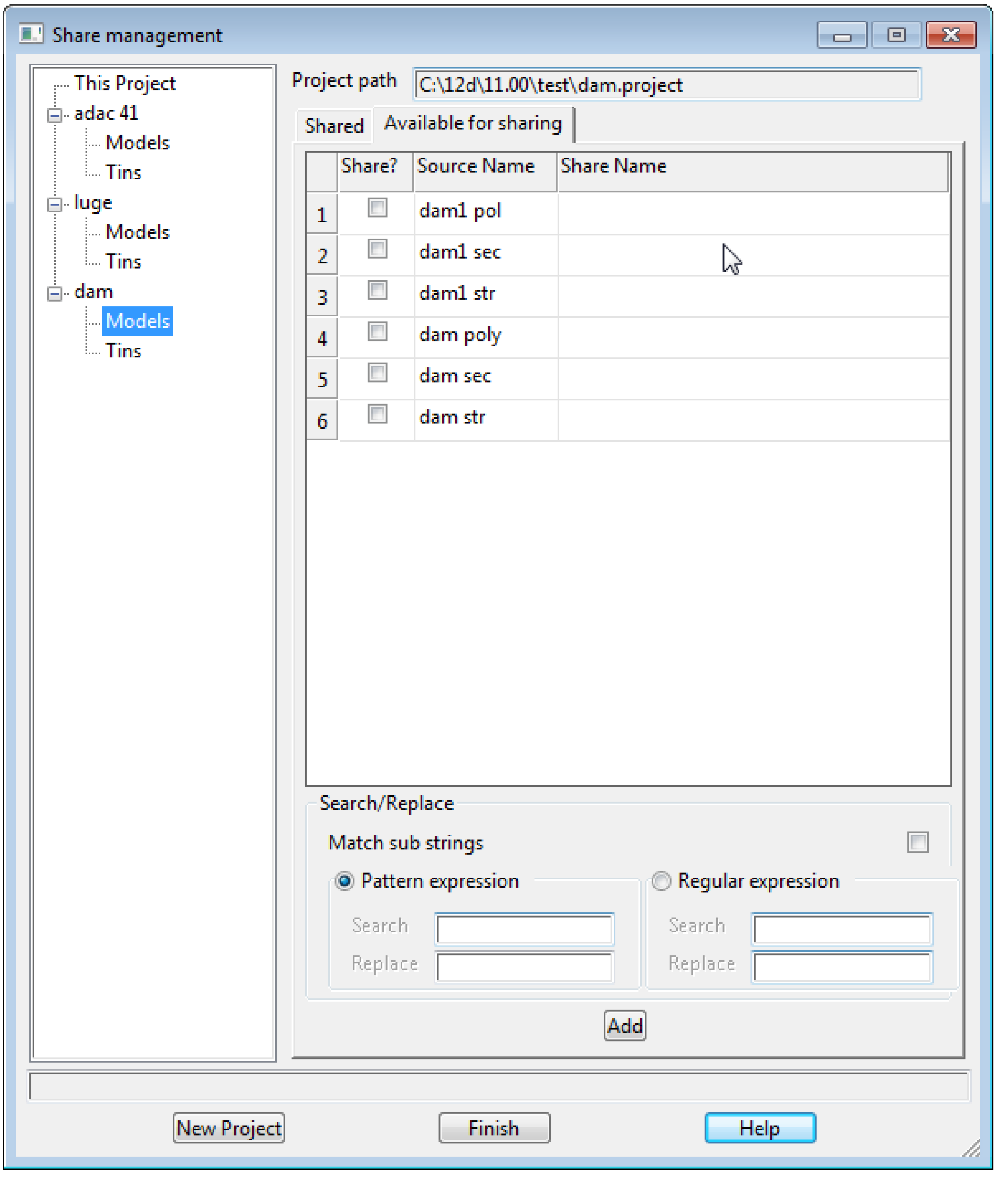
On the **Available Sharing** tab, all tins available for sharing in are shown and those that have already been shared in are coloured yellow. Tins that are not shared in can be shared in by ticking in the **Share ?** column and then clicking the **Add** button.

New Project Button

Clicking on the **New Project** button brings up the **Select a Project to Share From** panel.



Selecting a project and clicking on **Set** adds the given project as a new node in the **Share Management** panel and the **Model** and **Tin** nodes can be used to share in models and tins from that new project.



Go to the next section 7.6.13.4 Project Share Settings or return to 7.6.13.3 Share Management.

7.6.13.4 Project Share Settings

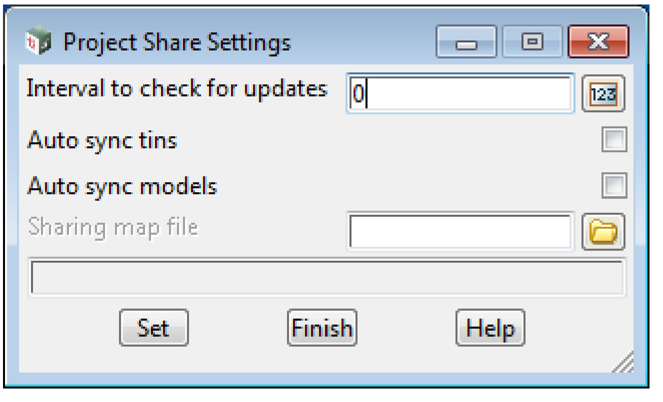
Position of option on menu: **Project =>Management =>Sharing =>Settings**

There are three environment variables that control the auto-synchronizing of any shared models or tins added to this project.

SHARE_CHECK_INTERVAL_4D	number_of_seconds
AUTO_MODEL_SYNC_4D	1 or 0
AUTO_TIN_SYNC_4D	1 or 0

This option will temporarily modify the values for this session. When the project is restarted, the values will revert to those given by the environment variables.

Selecting **Share settings** brings up the **Project Share Settings** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Interval to check for updates input

time in seconds to check if any of the shared models or tins added to the project have been updated in the server projects.

The environment variable controlling this when the project starts up is

SHARE_CHECK_INTERVAL_4D number_of_seconds

The value is only modified for this session. To permanently change the value, please modify the environment variable.

Autosync tins tick box

if ticked, the server projects for any shared tins added to this project are checked to see if they have been modified (checked every SHARE_CHECK_INTERVAL seconds). If any tins have been modified, they are re-copied to this project.

The environment variable controlling this when the project starts up is

AUTO_TIN_SYNC_4D 1 or 0

The value is only modified for this session. To permanently change the value, please modify the environment variable.

Autosync models tick box

if ticked, the server projects for any shared models added to this project are checked to see if they have been modified (checked every SHARE_CHECK_INTERVAL seconds). If any models have been modified, they are re-copied to this project.

The environment variable controlling this when the project starts up is

AUTO_MODEL_SYNC_4D 1 or 0

The value is only modified for this session. To permanently change the value, please modify the environment variable.

Set button

set the values in the panel for this session. The values will revert back to the ones given by the environment variables when the project is restarted.

The values are only modified for this session and will evert back to the ones given by the environment variables when the project is restarted. To permanently change the value, please modify the environment variable.

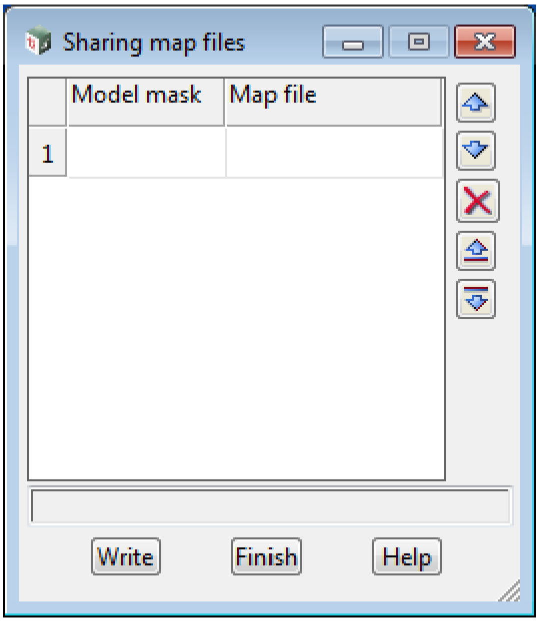
Go to the next section 7.6.13.5 Sharing Map Files or return to 7.6.13 Sharing.

7.6.13.5 Sharing Map Files

Position of option on menu: **Project =>Management =>Sharing =>Sharing map files**

The **Sharing Map Files** option allows different models **shared in** to use different ***Map files***.

Clicking on the option **Sharing map files** brings up the **Sharing Map Files** panel.



Model Mask-Map File grid:

The ***Model Mask Map*** grid in loaded from the file ***sharing_map_files.4d*** which is a ***Set Up*** file that is loaded when the project is opened.

The **Model Mask-Map File** grid consists of rows of text for **Model Mask** and one the same row, and associated **Map File**.

The **Model Mask** is a text string of alphanumeric characters and spaces, and including * for a wild card and **?** as a wild character.

For each **shared in** model, the model name is checked against the **Model Mask** in the first the first row, and if no match is made, is then tested against the next row. This is repeated until a match occurs.

When a match occurs, the **Map File** for the match row is applied to the model and no more tests against **Model Masks** are made.

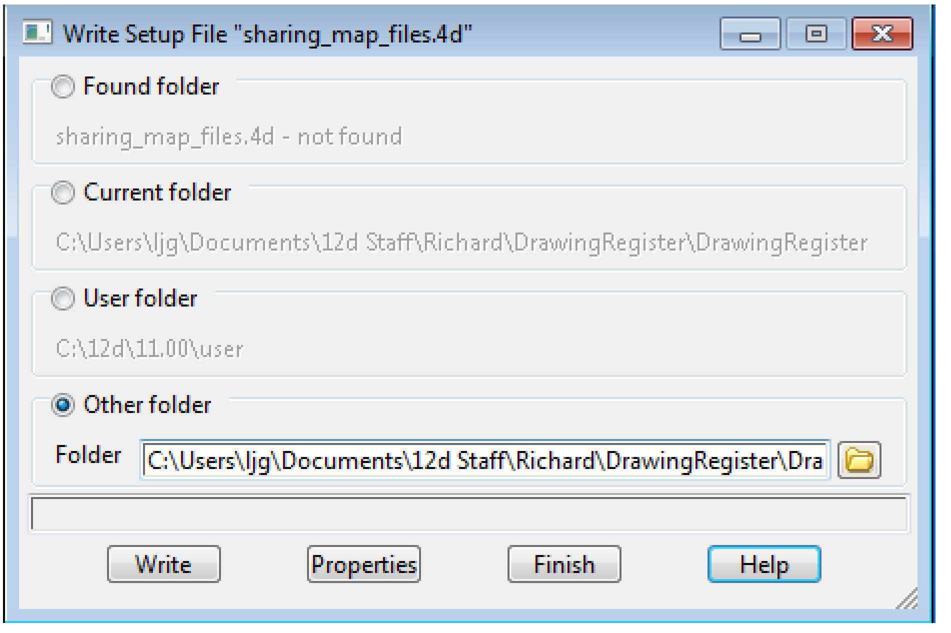
If there is no match but there is a project **Sharing map file** (see the field **Projects >Sharing >Sharing map file** in the **Edit Environment Variables** panel**)**, then the **Sharing map file** is applied to the **shared in** model.

If no match occurs and there is no project **Sharing map file** then the **shared in** model is left alone.

If the data in the ***Model Mask Map*** grid is modified then the modified file is written out with the **Write** button. A project restart is required for the new file to take effect.

Write Button

When the **Write** button is selected, a **Write Setup File** panel comes up to specify where the ***sharing_map_files.4d*** file is to be written out to.



For the choices on the panel, see 39.2.6 Writing Out Setup Files

A project restart is required for the new file to take effect

Note

When applying a **Map File** to a **shared in** model, only the **Basic** tab of the **Map File** is used and it is only used to change the colour of strings in the **shared in** model.

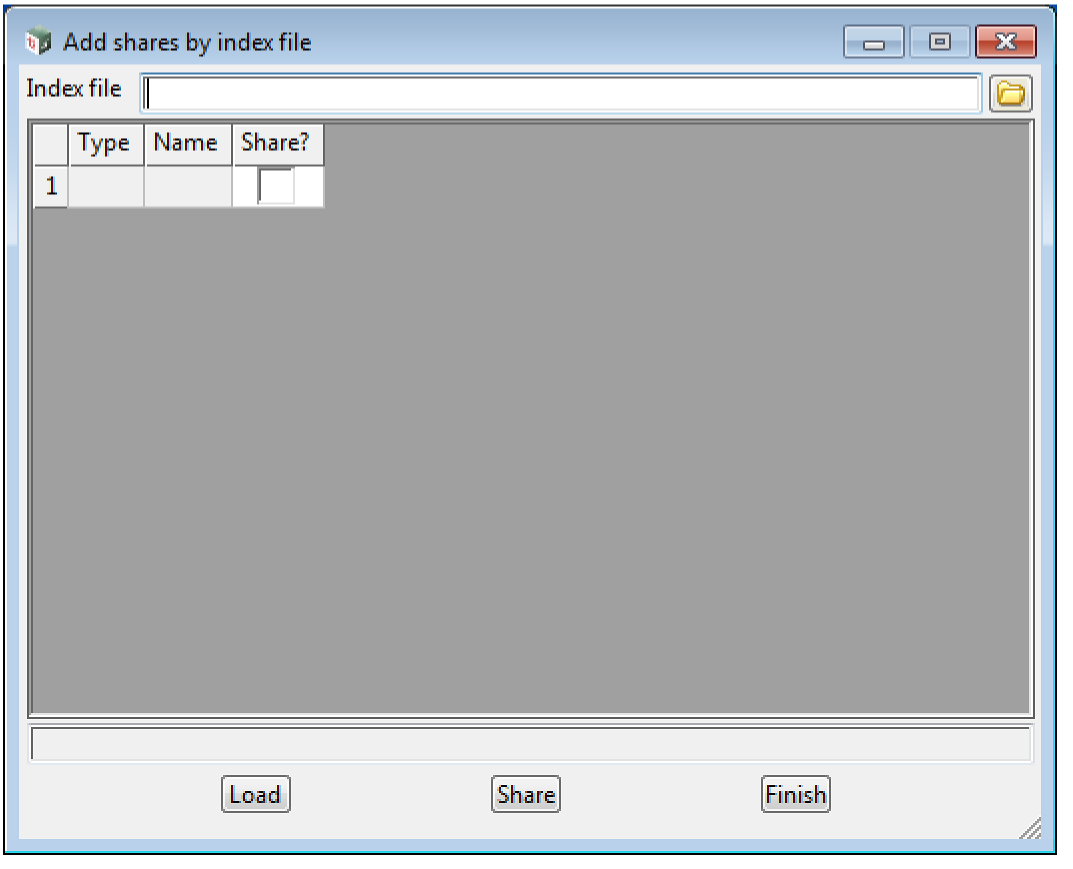
Go to the next section 7.6.13.6 Add Shares By Index File or return to 7.6.13 Sharing.

7.6.13.6 Add Shares By Index File

Position of option on menu: **Project =>Management =>Sharing =>Read share index**

This panel loads shared tin and model details from a share index file, created by 7.6.13.7 Export A Share Index File and allows you to choose which items should be shared into your project.

Selecting **Read share index** brings up the **Add Shares By Index File** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Index file	file		

the sharing index file to read

Type

the type of element (model or tin)

Name

the name of the element

Share?	tick box		

whether or not to share the element in

Load	button		

loads the information from the selected index file

Share	button		

adds the selected shares to the project

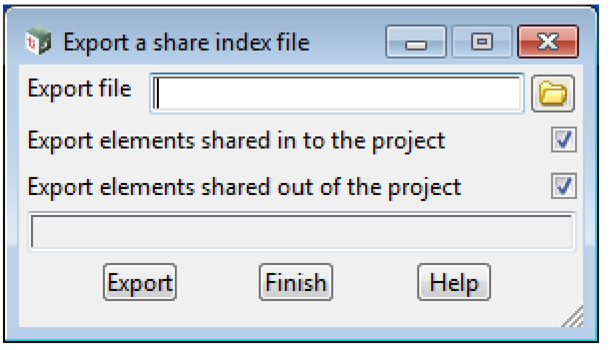
Go to the next section 7.6.13.7 Export A Share Index File or return to 7.6.13 Sharing.

7.6.13.7 Export A Share Index File

Position of option on menu: **Project =>Management =>Sharing =>Export share index**

This panel exports a list of all the shared elements (models and tins) that have been shared into this project. This can then be used to recreate the same sharing setup in another client project.

Selecting **Export share index** brings up the **Export a share index file** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Export file	file		
the file to export details to			
Export	button		
exports the sharing information			

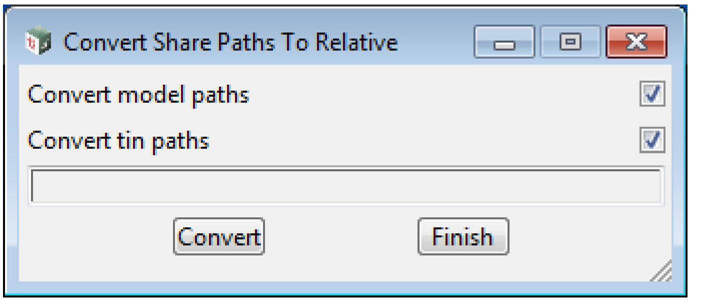
Go to the next section 7.6.13.8 Convert Share Paths To Relative or return to 7.6.13 Sharing.

7.6.13.8 Convert Share Paths To Relative

Position of option on menu: **Project =>Management =>Sharing=>Convert share paths to relative**

This option changes the share paths of models and tins from absolute path names to relative path names. It is mainly used for projects where the share names were recorded as full paths names (e.g. V8 projects).

Selecting **Convert share paths to relative** brings up the **Convert Share Paths to relative** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Convert model paths	tick box		tick

if ticked, the absolute path names of shared models added to this project will be converted to relative paths.

Convert tin paths	tick box		tick

if ticked, the absolute path names of shared tins added to this project will be converted to relative paths.

Convert button

convert absolute paths to relative.

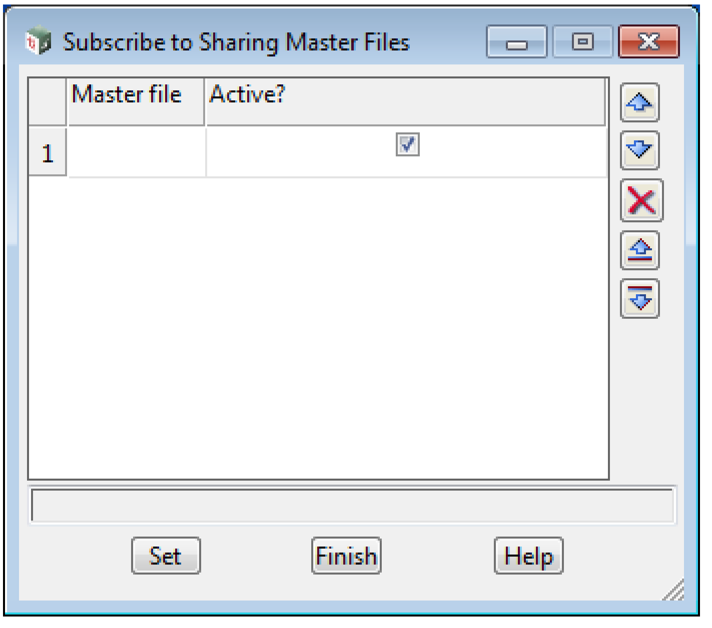
Go to the next section 7.6.13.9 Subscribe to Sharing Master Files or return to 7.6.13 Sharing.

7.6.13.9 Subscribe to Sharing Master Files

Position of option on menu: **Project =>Management =>Sharing =>Subscribe to share master files**

This panel allows you to subscribe to a set of sharing master files. Each master file defines a set of known models and tins, which will be shared into this project.

Selecting **Subscribe to share master files** brings up the **Subscribe to Sharing Master Files** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Set	button		

sets the list of files to which you wish to subscribe

Note that the share check interval must be set, which can be set from the 7.6.13.4 Project Share Settings **panel.**

Go to the next section 7.6.13.10 Sharing Master File or return to 7.6.13 Sharing.

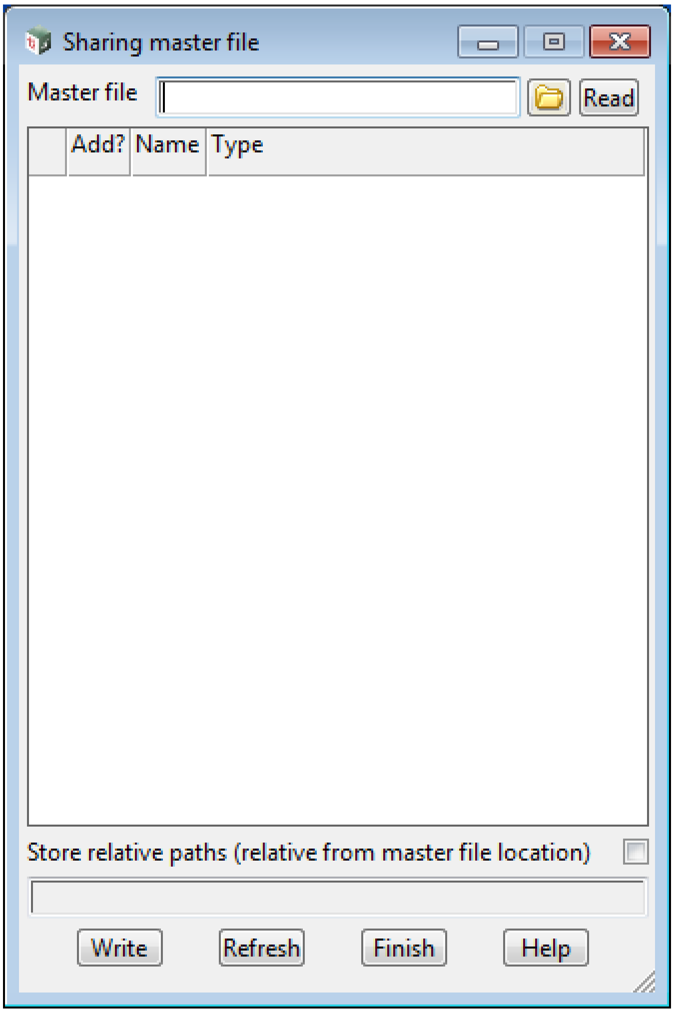
7.6.13.10 Sharing Master File

Position of option on menu: **Project =>Management =>Sharing =>Create share master files**

This panel allows you to define a sharing master file. A sharing master file can be created and subscribed to by a number of clients. This will enforce and define the set of shared models and tins the clients will consume. Updating the sharing master file will likewise update the clients, at a set interval as defined by the client.

This panel will list all models and tins that the user has marked as shareable.

Selecting **Create share master files** brings up the **Sharing master file** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Master file

the file to write to

Add? tick box

whether or not to add the item to the sharing master file

Name

the name of the model or tin

Type

the type of the element (model or tin)

Write button

writes the sharing master file

Refresh button

refreshes the list of shareable models and tins.

Go to the next section 7.6.13.11 Localize Shares or return to 7.6.13 Sharing.

7.6.13.11 Localize Shares

Position of option on menu: **Project =>Management =>Sharing =>Localize shares**

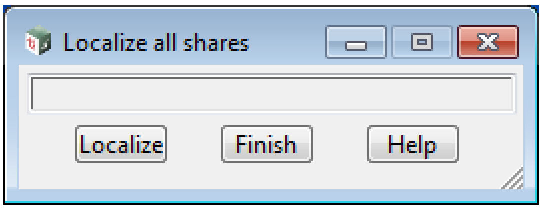
This option copies all the shared models and tins to the local project in a special folder and make them "local shares" so like the original shared models and tins, they still can’t be edited.

This is useful for creating a project that is being sent to another user so that all the shared models and tins are included in the project.

IMPORTANT WARNING

You can’t go back to the original shares so make sure this is only run on a coply of a project and not a working project. If you only wanting copies of your shared models and tins in the project then use the Copy models and tins commands.

Selecting **Subscribe to share master files** brings up the **Subscribe to Sharing Master Files** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Localize	button		

creates "local" shares of all the shared models and tins.

Return to 7.6.13 Sharing.

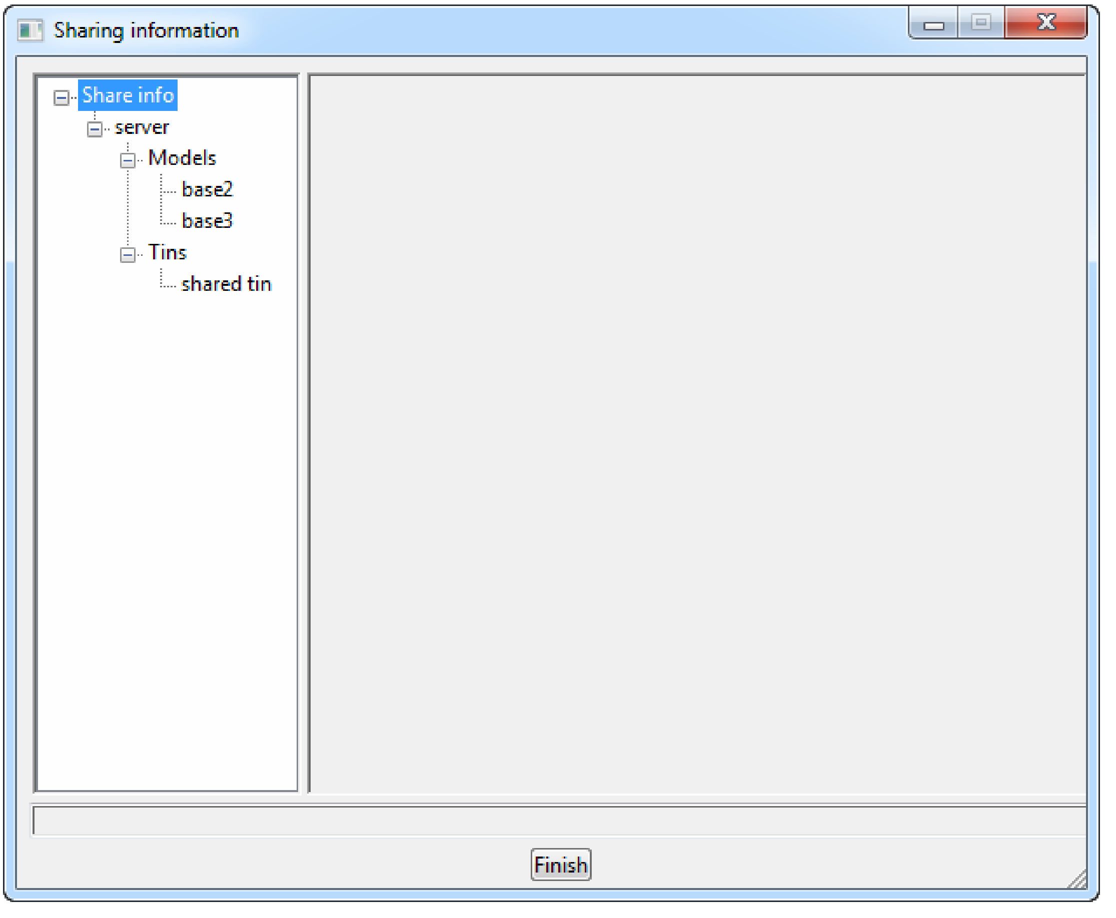
7.6.13.12 Sharing Information

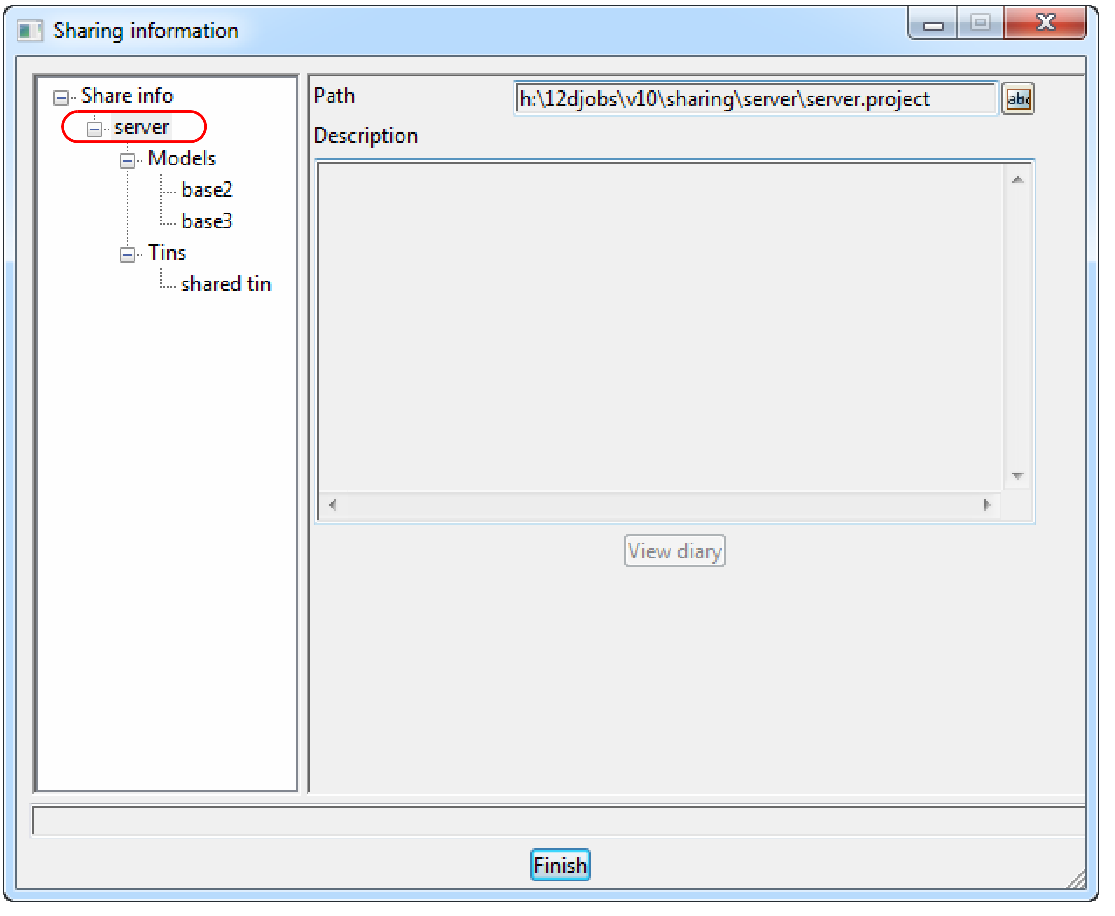
Position of option on menu: **Project =>Management =>Sharing =>Share info**

THIS OPTION HAS BEEN REMOVED FOR V11 AND REPLACED BY7.6.13.3 Share Management**.**

This panel shows information about all models and tins shared into your current project.

Selecting **Share info** brings up the **Sharing information** panel.





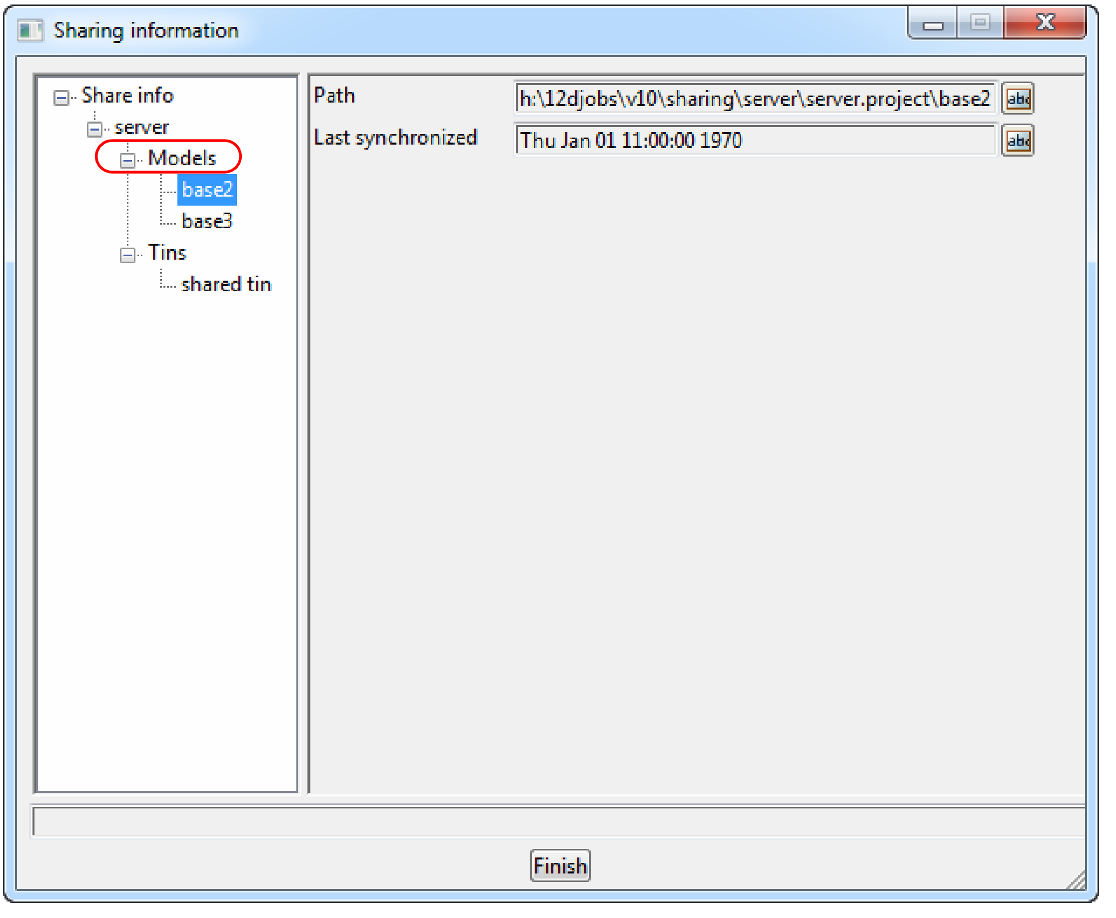
Server Node

Each server you are sharing from is listed in the tree.
Selecting the server node displays the path of the server, the project description and an option to view the diary for that project.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View diary	button		

Views the diary for that project - disabled if no project diary created



Models Node

Each Model Node shows details about the shared model.

The fields and buttons used in this panel have the following functions.

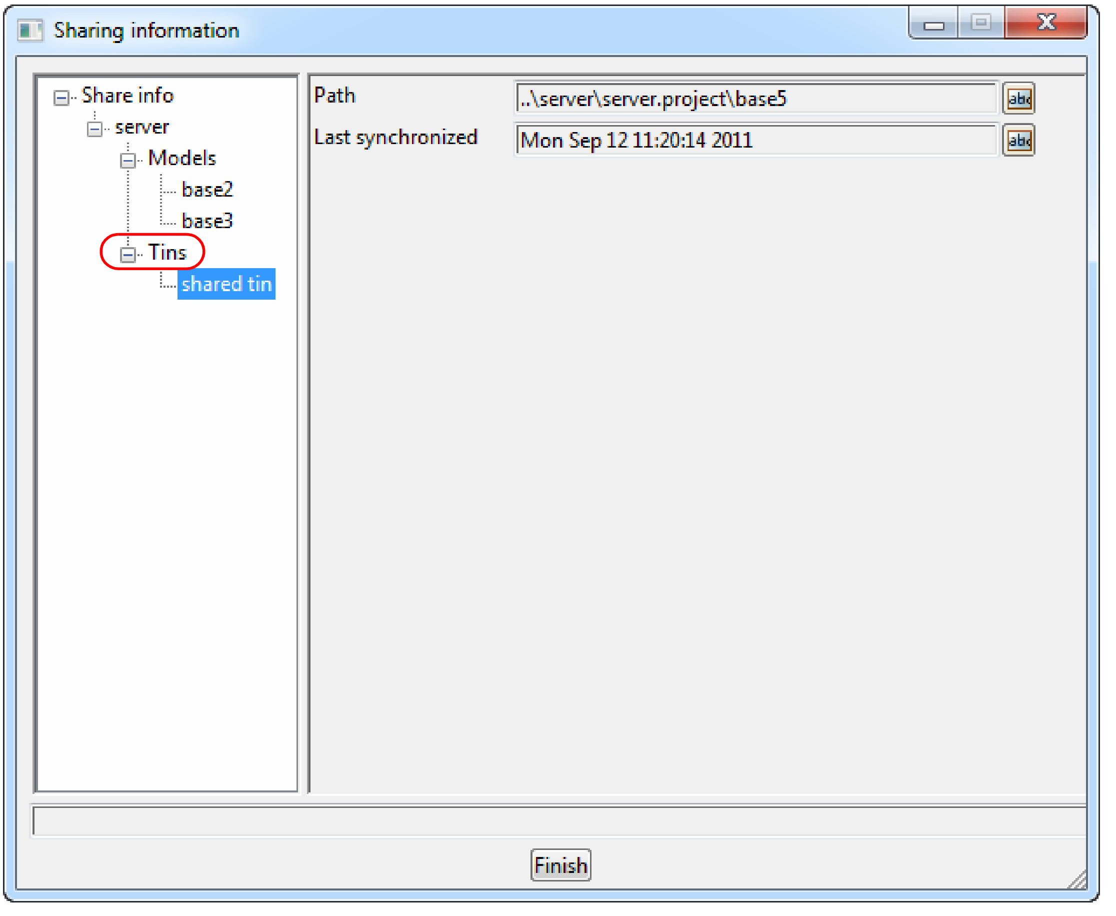
Field Description Type Defaults Pop-Up

Path

the path to the original source of the model

Last Synchronized

the time the model was last synchronized



Tin Node

Each Tin Node shows details about the shared tin.

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Path

the path to the original source of the tin

Last Synchronized

the time the tin was last synchronized

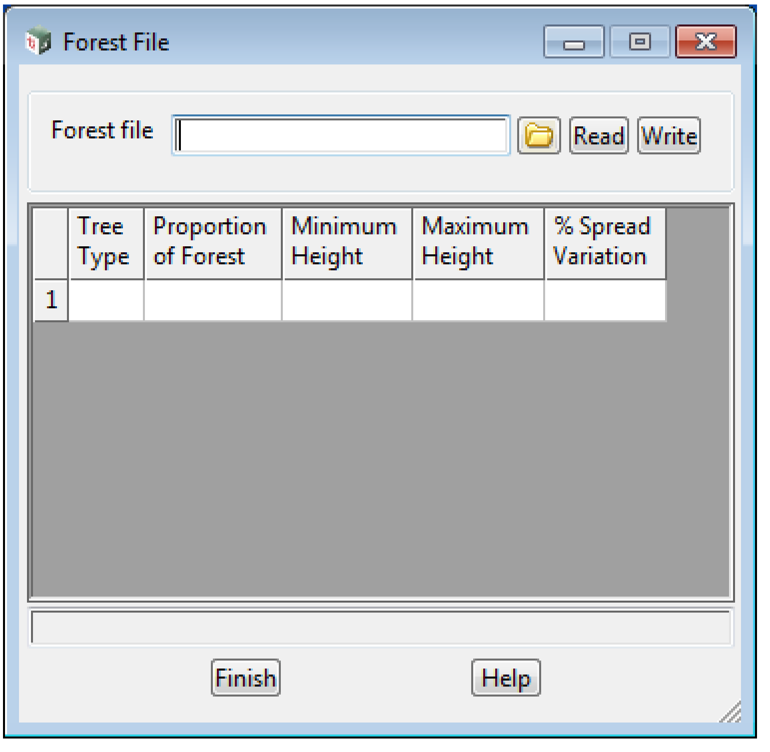
Return to 7.6.13 Sharing.

7.6.14 Forest File

Position of option on menu: **Project =>Management =>Forest file**

The **Forest file** option creates files to control the types of trees, variation in heights and spreads and relative proportions of the various tree types in the file.

Selecting **Forest file** brings up the **Forest file** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Forest file	file box		*.forest

name for the forest definitions.

Read button

read in the forest file given in **Forest file**

Write button

write out to forest file to the file given in **Forest file**.

Tree grid grid

Tree type billboard select

name of the billboard to be placed for the tree - 12d trees are under the Trees, Palms and Shrubs section of the billboards.

Proportion of Forest positive real *.forest

relative proportions of the forest made up of this tree. The actual proportion is this value over the sum of all the values in this column.

Maximum/Minimum Height positive reals

min and maximum heights to use for this tree type.

% Spread Variation	positive real between 0 and 100

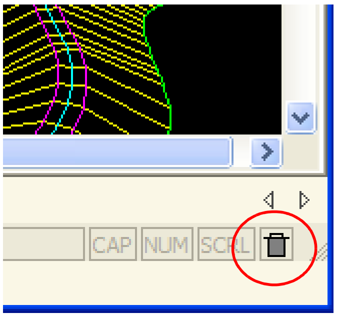
Go to the next section 7.6.15 Trash Bin or return to 7.6 Management.

7.6.15 Trash Bin

Position of option on menu: **Project =>Management =>Trash bin**

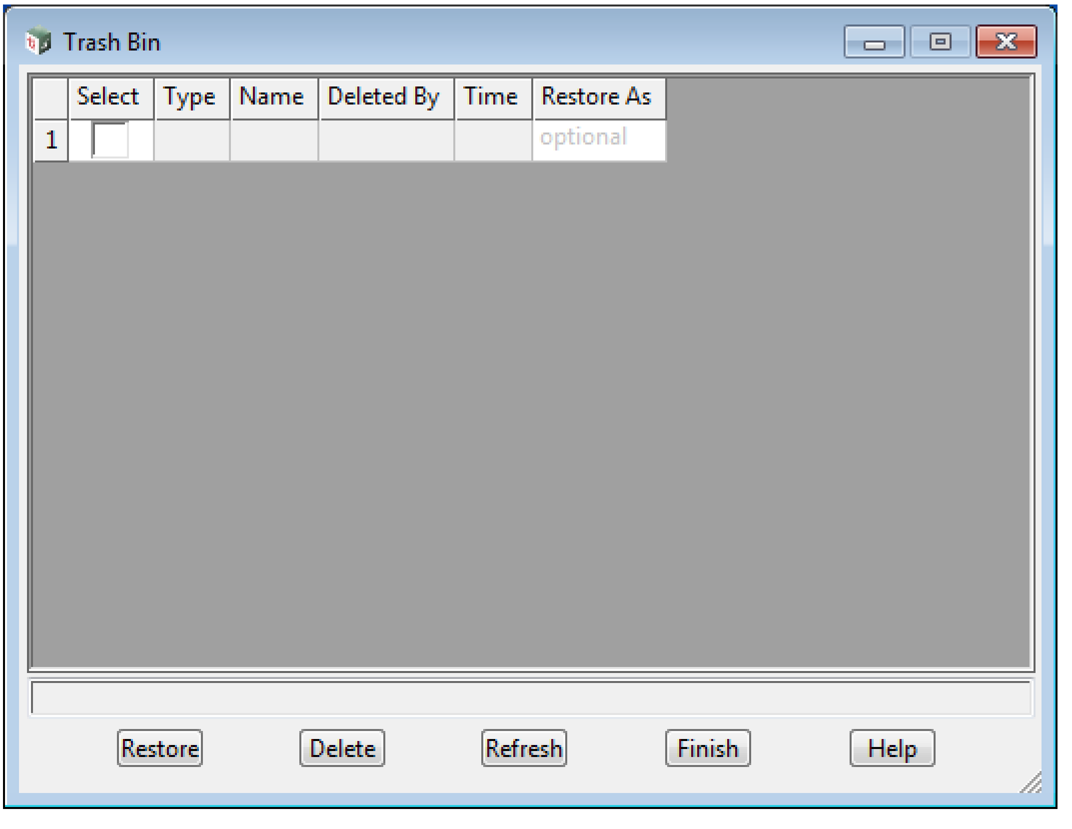
If the ***Trash Bin*** is turned on (by setting the env variable USE_TRASH_BIN_4D) then any deleted models, tins, functions and templates are not deleted from the disk but moved to the 1***2d Trash Bin***. Also for any models that are cleaned, the strings are moved to the trash bin.

If there is any data in the trash bin, a trash bin icon is displayed in the bottom left had corner of the project window.



Clicking on the trash bin icon, brings up the **Trash Bin** panel which displays the tins, templates or functions that have been deleted, and the models that have been deleted or cleaned. The delete/ cleaned objects can be restored to the project.

Selecting **Trash bin** from **Project =>Management =>Trash bin**, also brings up the **Trash Bin** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Trash Bin Grid grid

Type output only model, tin

type of object - model or tin.

Name output only

name of the object.

Deleted by output only

name of the user who deleted/cleaned the object.

Time output only

time of the deletion.

Select tick box

*ticked if this object is to be Restored/Deleted. Note that right bottom on Select brings up a menu to **Toggle** (tick to no tick, no tick to tick), **Set** (set all to tick), or **Clear** (set all to no tick) for the entire column.*

Restore As text

if not blank, the name the restore the object as.

If blank, restore the object with its original name in the **Name** *column.*

Buttons grid

Restore button

restore all the ticked objects.

Delete button

delete all the ticked objects from the trash. These items are then permanently deleted.

Refresh button

refresh the grid.

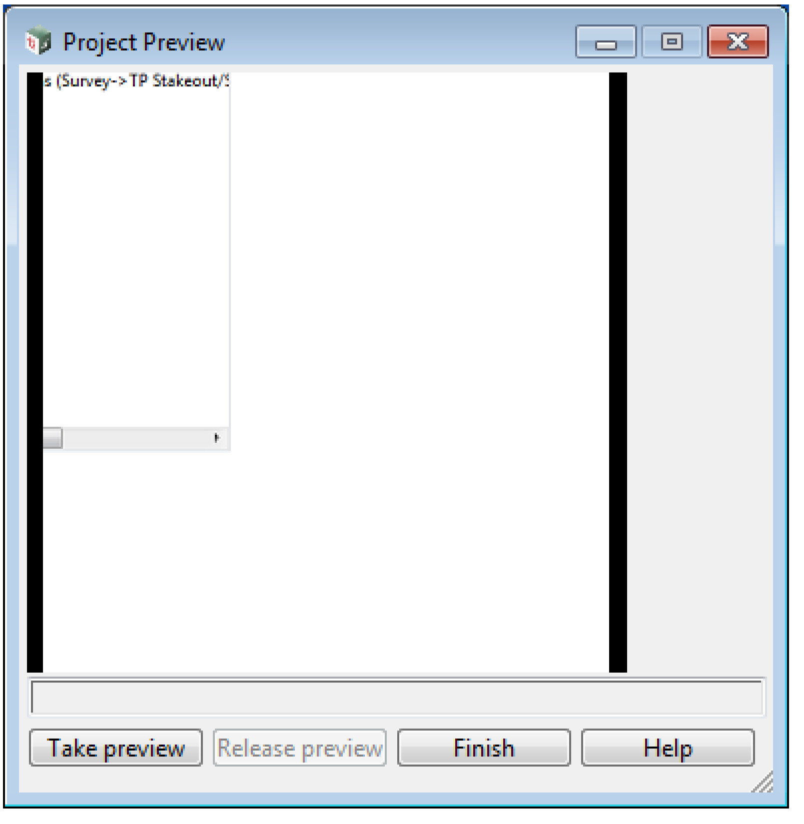
Go to the next section 7.6.16 Project Preview or return to 7.6 Management.

7.6.16 Project Preview

Position of option on menu: **Project =>Management =>Project preview**

When a ***12d Model*** project is exited, an image is saved to display in the New/Open project options. The image can be the last active view or an image defined by the user. The *Project preview* option can write out a user selected image or remove one if a fixed image has been previously set.

Selecting **Project preview** brings up the **Project Preview** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Take preview	button		

if clicked, the image of the current active view is written and set to be used as a fixed project preview image. That is, this image is always used as the project preview.

Release preview	button		

active when a fixed image is being used for the project preview. In that case if it is then clicked, the fixed image is deleted. An automatic image of the last active view will then be saved each time the project is exited.

Go to the next section 7.6.17 Toggle Density Drawing or return to 7.6 Management.

7.6.17 Toggle Density Drawing

Position of option on menu: **Project =>Management =>Toggle density drawing**

Toggles the **Use density drawing** tick box in the **Defaults** panel.

For more information on **density drawing**, see Use density drawing tick box.

Note: This setting is only applicable to the 250M version of ***12d Model.***

Continue to the next section 7.6.18 Toggle Topmost Buttons or return to 7.6 Management.

7.6.18 Toggle Topmost Buttons

Position of option on menu: **Project =>Management =>Toggle topmost buttons**

Toggles on buttons to use instead of a keyboard or mouse when working on a tablets.

Clicking on **Toggle topmost buttons** toggles the following buttons on and off.

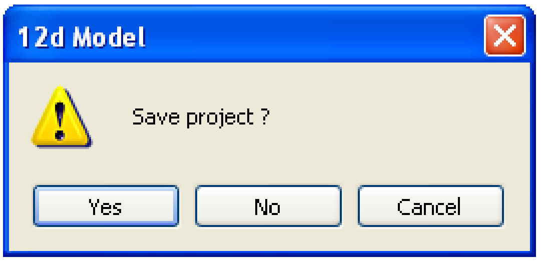
Return to 7.6 Management.

7.7 Restart

Position of option on menu: **Project =>Restart**

The **Restart** option exits the current 12d Model project and then restart using the same project. Useful for testing changes to set up files etc.

It will prompt if project *Save* is required.



Go to the next section 7.8 Save or return to 7 Project.

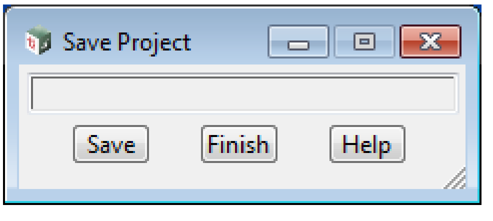
7.8 Save

Position of option on menu: **Project =>Save**

The **Save** option saves the working project to disk.

On the Save from the Main menu, simply select the option and the project is saved.

On selecting **Save** from the floating Projects menu, the **save project** panel is displayed.



The position of the **save project** panel is also saved and the panel automatically placed on the screen when the project is started up again.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Save	button		

after selecting this button, the working project is saved to disk.

Go to the next section 7.9 Tree or return to 7 Project.

7.9 Tree

Position of option on menu: **Project =>Tree**

Position of option on menu: **Project =>Management =>Tree**

The **Tree** option is under continual development and being extended with each version of 12d Model.

Tree is used to drill down through information about the data in the project (for example models, tins and views) and also to interactive create/edit/edit most of the set-up information that is contained in text files. These include:

(a) Name mapping (file names.4d) documented in the section 7.9.1 Name Mappings - Names.4d File

(b) Plotters set-ups (file plotters.4d) which in not yet documented. See the section 7.9.2 Plotters.

(c) Survey data collectors (file survey.4d) documented in the section 7.9.3 Survey Data Collectors

(d) Linestyles (file linestyl.4d) documented in the section 7.9.4 Linestyles

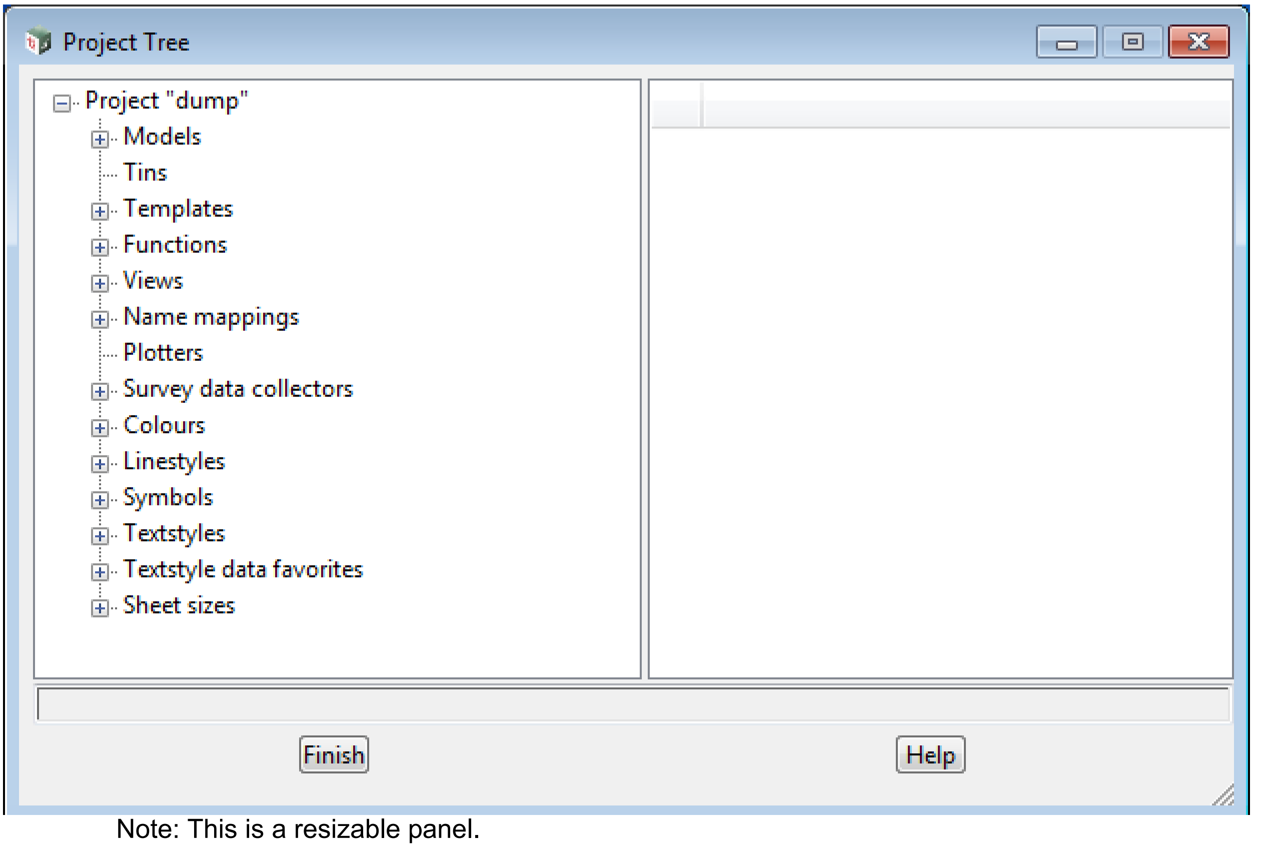
(e) Symbols (file symbols.4d) documented in the section 7.9.5 Symbols

(f) Textstyles (file textstyl.4d) documented in the section 7.9.6 Textstyles

(g) Textstyle data favourites (file textstyle_names.4d) documented in 7.9.7 Textstyle Data Favourites

(h) Sheet sized (file sheets.4d) documented in the section 7.9.8 Sheet Sizes

On selecting the **Tree** option, the **Project Tree** panel is displayed.

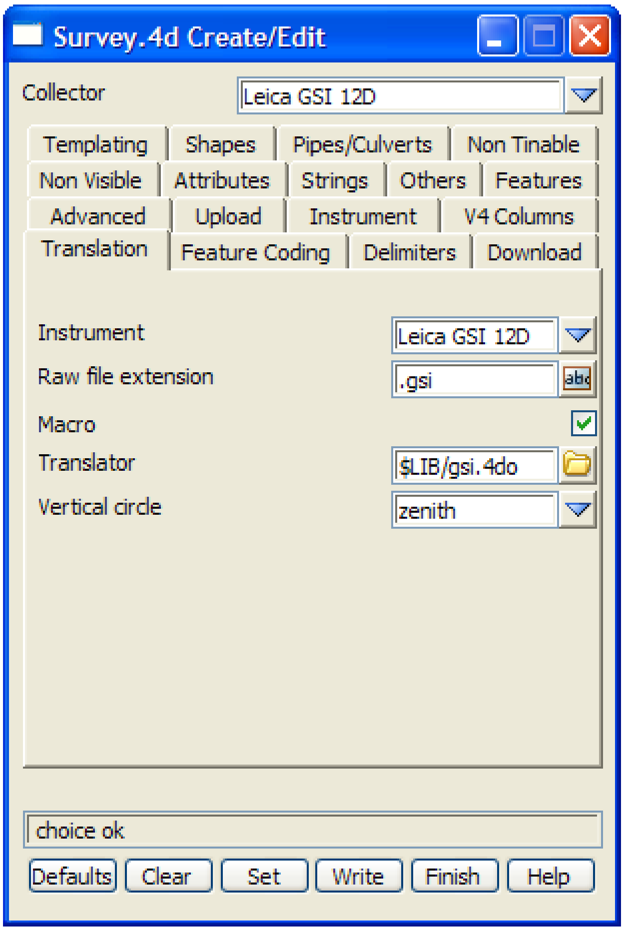


Note: This is a resizable panel.

Clicking on a **+** expands that browse tree node to show what items are available to get more information on. Clicking on a **-** collapses that browse tree node,

For the items *Name mappings*, *Plotters*, *Survey data collectors*, *Textstyles* and *Textstyle data favourites*, once they are expanded then the items displayed need no further expansion (there is no **+** displayed in front of the items) and double clicking on the expanded items brings up the create/edit panels for the item.

For example, clicking on + for *Survey data collector*s and then double clicking on a data collector brings up the *Survey.4d Create/Edit* panel.



For the option *Name mappings*, go to	7.9.1 Name Mappings - Names.4d File
Plotters	7.9.2 Plotters
Survey data collectors	7.9.3 Survey Data Collectors
Linestyles	7.9.4 Linestyles
Symbols	7.9.5 Symbols
Textstyles	7.9.6 Textstyles
Textstyle data favourites	7.9.7 Textstyle Data Favourites
Sheet sizes	7.9.8 Sheet Sizes

7.9.1 Name Mappings - Names.4d File

Position of option on menu: **Project =>Tree**

The **name mapping** file is used in all panel fields requiring the *name* of a string and the name field in the **CAD Controlbar**.

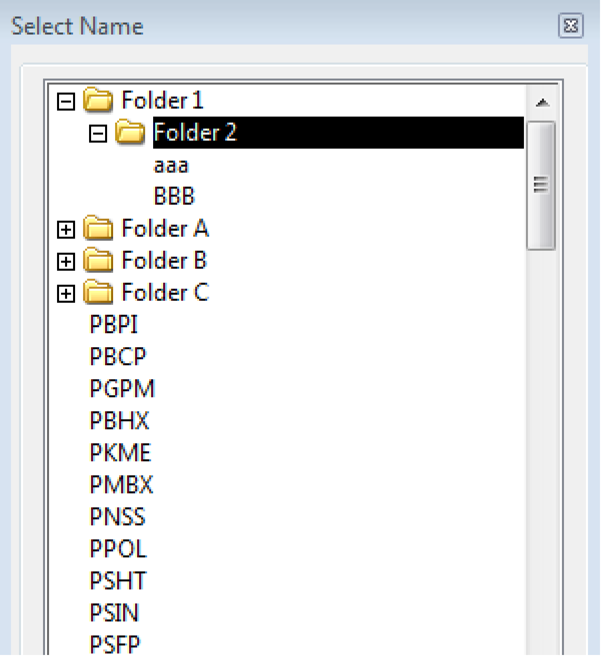
After the **string name** is typed, if <Enter> is pressed then each section of the name mapping file being used for the project is searched for matches with the Key for the Basic node, or Name field for the other nodes (*Symbol*, *Tinable* etc).

If there is a match of the string name with a key/name, then the name, model, colour point/line type etc. for that key/name may be used in the appropriate panel fields, toolbar fields, ControlBars etc that go with the string name.

Note:

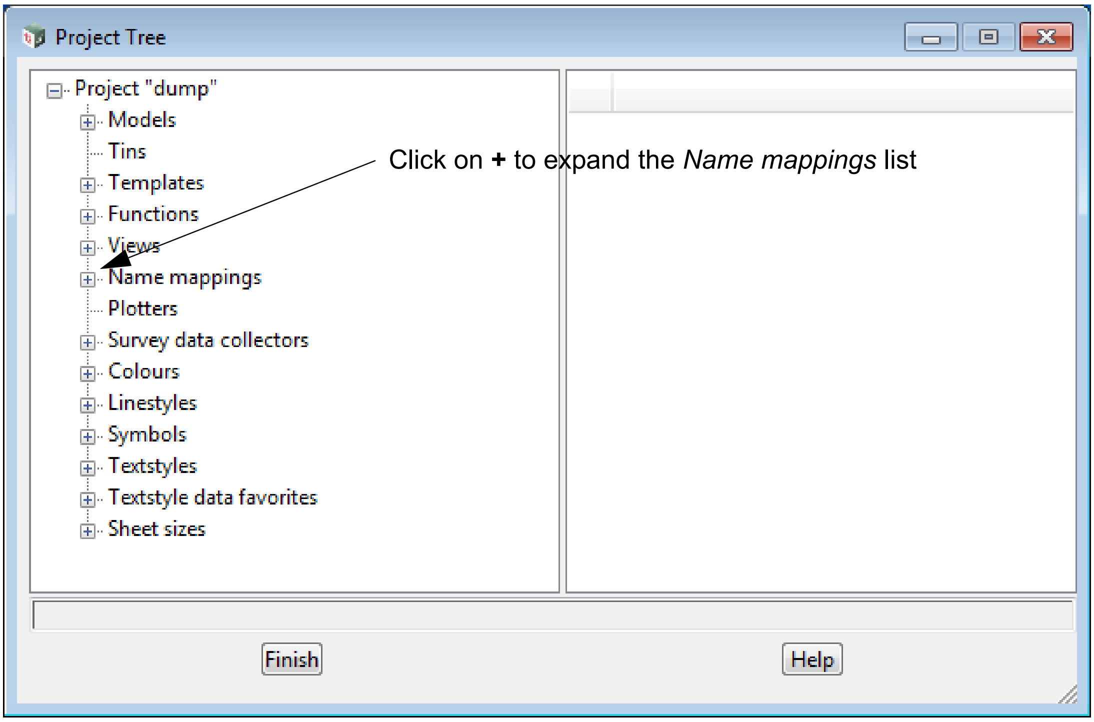
From ***12d Model 11*** onwards, there is a **Group** column in the grid for the **Basic** node in the ***Names.4d*** file so that the individual rows of the grid can be given a group/subgroup structure.

When the ***Groups*** column is used, the ***Names*** pop-up can look like:



To edit the **names.4d** file, either:

(a) Click on the option **Project=> Tree**, to bring up the **Project tree** panel, and then click on the **+** on *Name mapping* to expand the item:

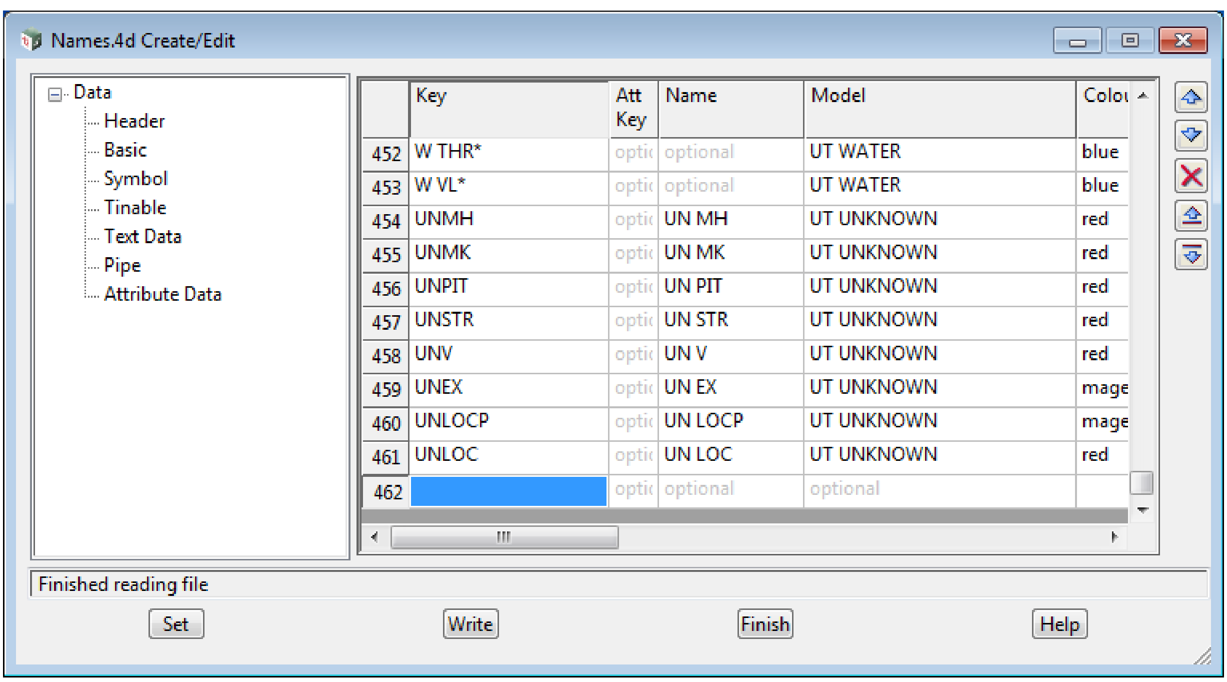


Double clicking LB on the item *Create name mapping* to bring up the **Names.4d Create/Edit** panel which will be already editing the *names.4d* file.

Or double clicking LB on any of the items below *Create name mapping* in the expansion of *Name mappings* will also bring up the **Names.4d Create/Edit** panel.

(b) Clicking LB on the Names icon in any panel or ControlBar field will bring up the **Select name** panel and selecting [Edit] at the bottom of the panel brings up the **Names.4d Create/Edit** panel.

The **Names.4d Create/Edit** panel that appears is.



Buttons

Set button

sets the definition for this editing session of the project - the information will be lost when the project is exited.

Write button

write the information to the file names.4d. The file can then be used when projects are started. The information won't be used for the current session unless the **Set** *button is selected as well. For more information on the* **Write** *button, go to the section* 39.2.6 Writing Out Setup Files *in the Appendix* 39 Setting Up & Configuring 12d

The nodes on the ***Names.4d Create/Edit*** panel are a subset of those on the ***Map File Create/ Edit*** panel. Some information will be given for each node but most of it, a link will be given to the appropriate section of the ***Map File Create/Edit*** panel.

For more information on *Header*, go to 7.9.1.1 Header Node
Basic 7.9.1.2 Basic Node
Symbols 7.9.1.3 Symbol Node
Tinable 7.9.1.4 Tinable Node
Text Data 7.9.1.5 Text Data Node
Pipe 7.9.1.6 Pipe Node
Attribute Data 7.9.1.7 Attributes Data Node

7.9.1.1 Header Node

The **Header** just has lines of comment text. A common use is to document which client the file is to be used for, or the revision history.

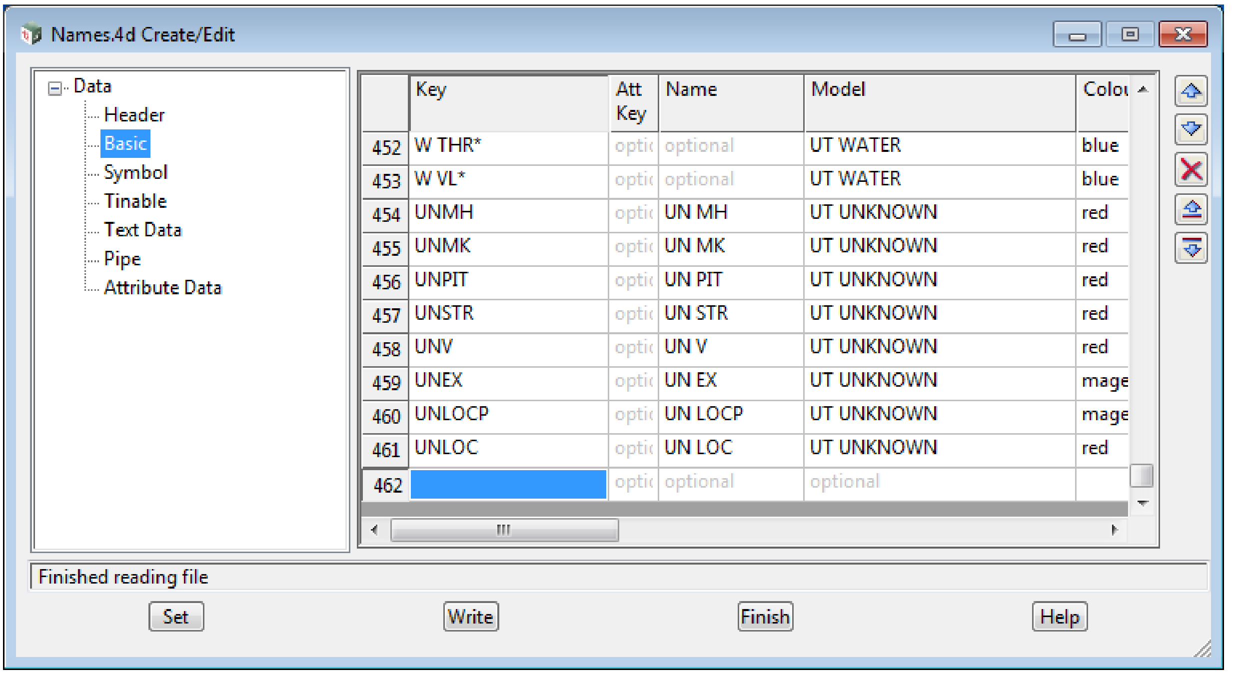
Go to the next section 7.9.1.2 Basic Node or return to 7.9.1 Name Mappings - Names.4d File.

7.9.1.2 Basic Node

The **Basic** grid selects data using the typed in value in the **Name** field of the CAD ControlBar, or any other appropriate **Name** panel fields.

The typed in value is checked against the **Key** column and if a match occurs, sets the name, model, colour, linestyle, point-line type and weight fields. The **Att Key column** is not used.

Note: If the is a name is the ***Name*** column of the **Basic** node, then that name replaces the value typed into the **Name** field in the panel or the CAD ControlBar. If the Name column for the match is blank, the text that is typed in is left and used in the **Name** field of the panel or the CAD ControlBar.



Processing Using Key for Basic Node

*the typed in value is searched against the list of names in the **Key** column until a **first** match with* Key *is found.*

If a match is found, then any non-blank data in the grid is used to fill out the appropriate panel fields or toolbar fields. The Key *column can include wild cards (*) and/or wild characters. The **Att Key** column is not used.*

If no match is found then none of the grid fields are used.

Name input

if not blank, name used to replace the typed in value. This is usually blank.

Model model grid available models

if not blank, model to use

Colour colour grid available colours

if not blank, colour to use

Point Line point/line grid point/line

if not blank, point-line type to use

linestyle available linestyles

if not blank, the linestyle to use

Weight input

if not blank, weight to use for strings with linestyle 1

Comment input

user comment

Group text input

There is now an optional **Group** *column in the grid for the **Basic** node in the **Names.4d** file so that the individual rows of the grid can be given a group/subgroup structure.*

If the Group column for a row is not blank, then the information is used as a group/subgroup structure in any names.4d pop up.

The group/subgroup structure is written as the top group name first, followed by the first level subgroup name, the second level subgroup name, etc., with each of the names separated by a forward slash /.

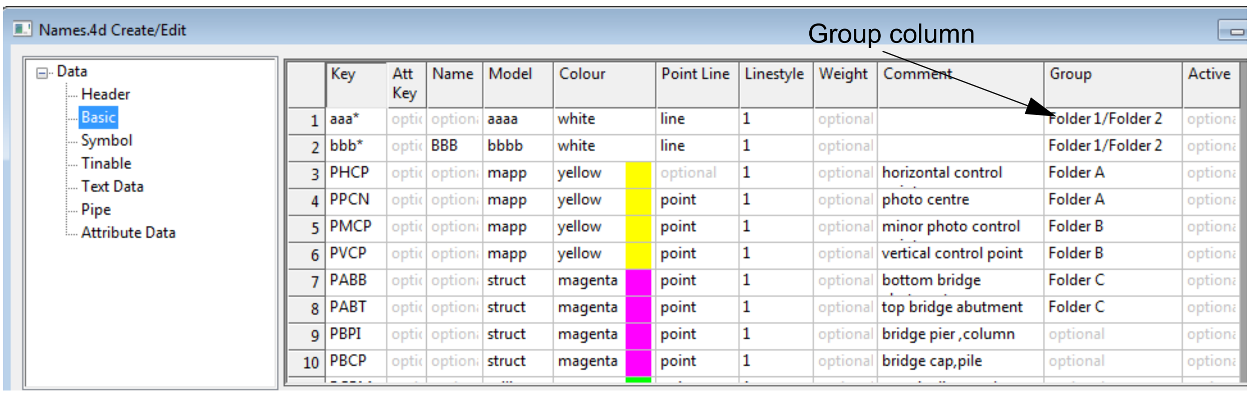
group_name/first_level_subgroup_name/second_level_subgroup_name etc.

For example, a top level group Folder1 with the first level subgroup Folder2 is written as

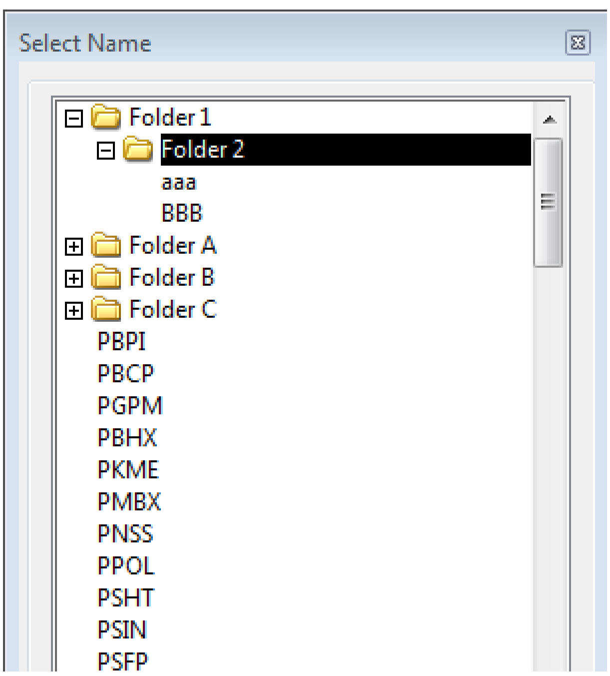
Folder1/Folder2

IMPORTANT NOTE:

Regardless of the group/subgroup structure, the search order for ***finding a match*** *with the key in the* ***Basic*** *node of the* ***Names.4d*** *file is still the order that the rows occur in the* ***Basic*** *grid.*



In a ***Names.4d*** *pop, the above names.,4d file will look like*



If you ***don't*** *want the* ***Group Column*** *to be used then you have to set the environment variable* ***USE_TREE_NAME_BOX_4D*** *to* ***0****. By default is has the value* ***1****.*

Comment and Active

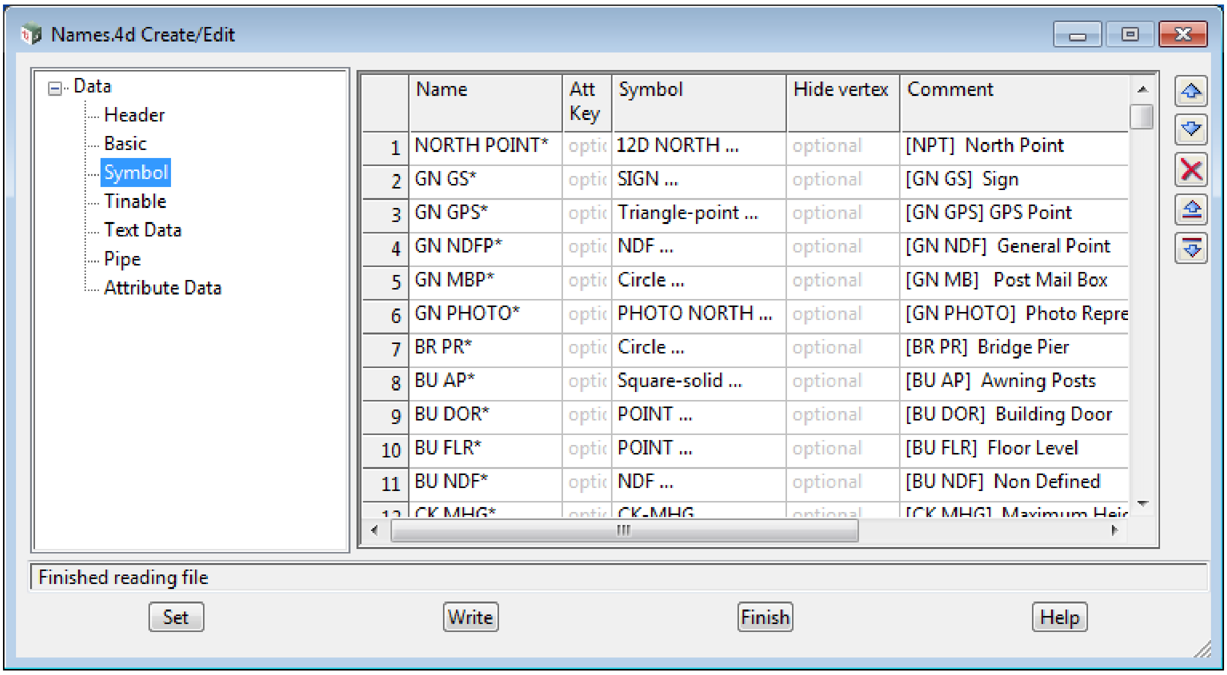
See 8.8.2 Comment and Active Column in Grids.

Go to the next section 7.9.1.3 Symbol Node or return to 7.9.1 Name Mappings - Names.4d File.

7.9.1.3 Symbol Node

The **Symbols** grid selects data using either the typed in value in the **Name** field of the CAD ControlBar or any other appropriate **Name** panel fields. Or, if there was a substituted *Name* from the **Names** column in the **Basic** node, then that is used.

The **Att Key column** is not used.



Processing Using Name in the Symbol Node

*the typed in value, or **Name** if one has been substituted from the **Name** column in the **Basic** node, is searched against the list of names in the **Name** column until a **first** match with* Name *is found.*

If a match is found, then any non-blank data in the grid is used to fill out the appropriate panel fields or toolbar fields. The Name *column an include wild cards (*) and/or wild characters. The **Att Key** column is not used.*

If no match is found then none of the grid fields are used.

Symbol Symbols grid

*when the CAD ControlBar **Name** field is being used the* **Symbol** *name from the **Symbol column** is used in the Symbol name fields in the Symbol ControlBar.*

*When a **Name** field is used in other appropriate panels, the* **Symbol** *name from the **Symbol column** is used in Symbol fields.*

Comment and Active

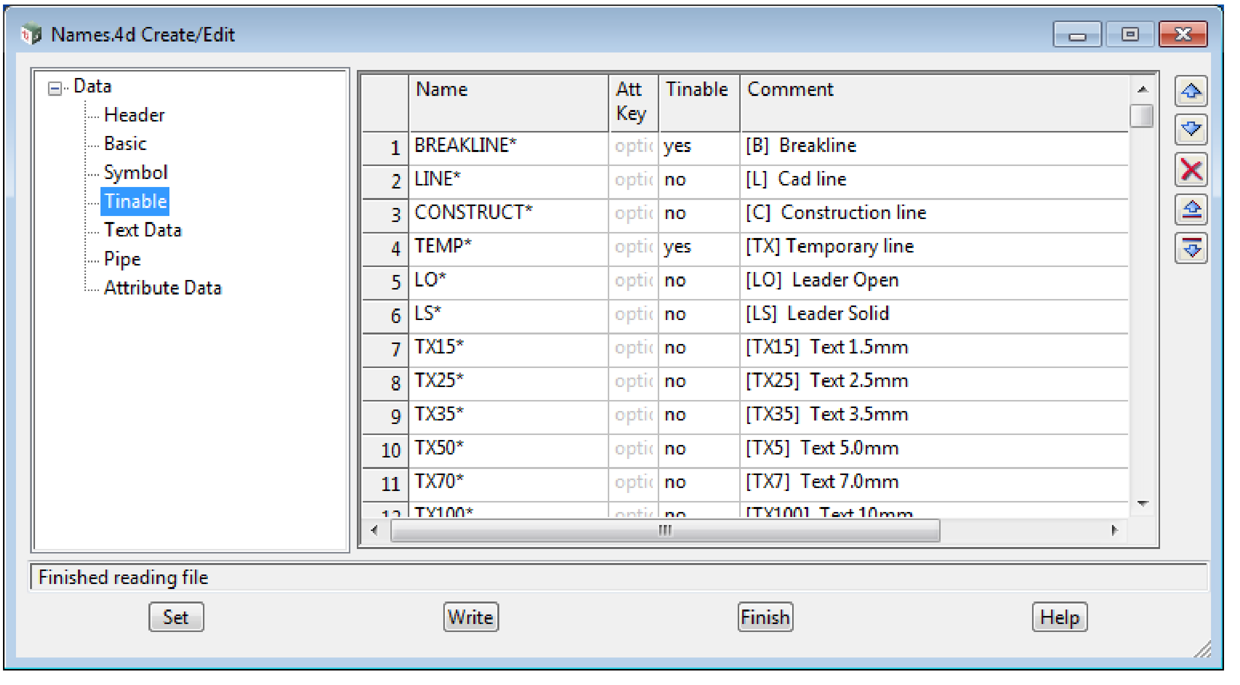
See 8.8.2 Comment and Active Column in Grids.

Go to the next section 7.9.1.4 Tinable Node or return to 7.9.1 Name Mappings - Names.4d File.

7.9.1.4 Tinable Node

The **Tinable** grid selects data using either the typed in value in the **Name** field of the CAD ControlBar or any other appropriate **Name** panel fields. Or, if there was a substituted *Name* from the **Names** column in the **Basic** node, then that is used.

The **Att Key column** is not used.



Processing Using Name in the Tinable Node

*the typed in value, or **Name** if one has been substituted from the **Name** column in the **Basic** node, is searched against the list of names in the **Name** column until a **first** match with* Name *is found.*

If a match is found, then any non-blank data in the grid is used to fill out the appropriate panel fields or toolbar fields. The Name *column an include wild cards (*) and/or wild characters. The **Att Key** column is not used.*

If no match is found then none of the grid fields are used.

Tinable tinability grid

*when the CAD ControlBar **Name** field is being used the* **Tinable** *value from the **Tinable column** is used in the Tinability fields in the CAD ControlBar.*

*When a **Name** field is used in other appropriate panels, the* **Tinable** *value from the **Tinable column** is used in Tinability fields.*

For the Tinable value:

*if the Tinable value is **no**, the entire string is non tinable. That is, the vertices are not included in any triangulation.*

*If **yes**, the entire string is tinable. That is, the vertices are used in any triangulation and the segments between the vertices are used as breaklines.*

*If **points**, the vertices of a string are tinable but the segments are not. That is, the vertices are used in any triangulation but the segments between the vertices are used as breaklines.*

Comment and Active

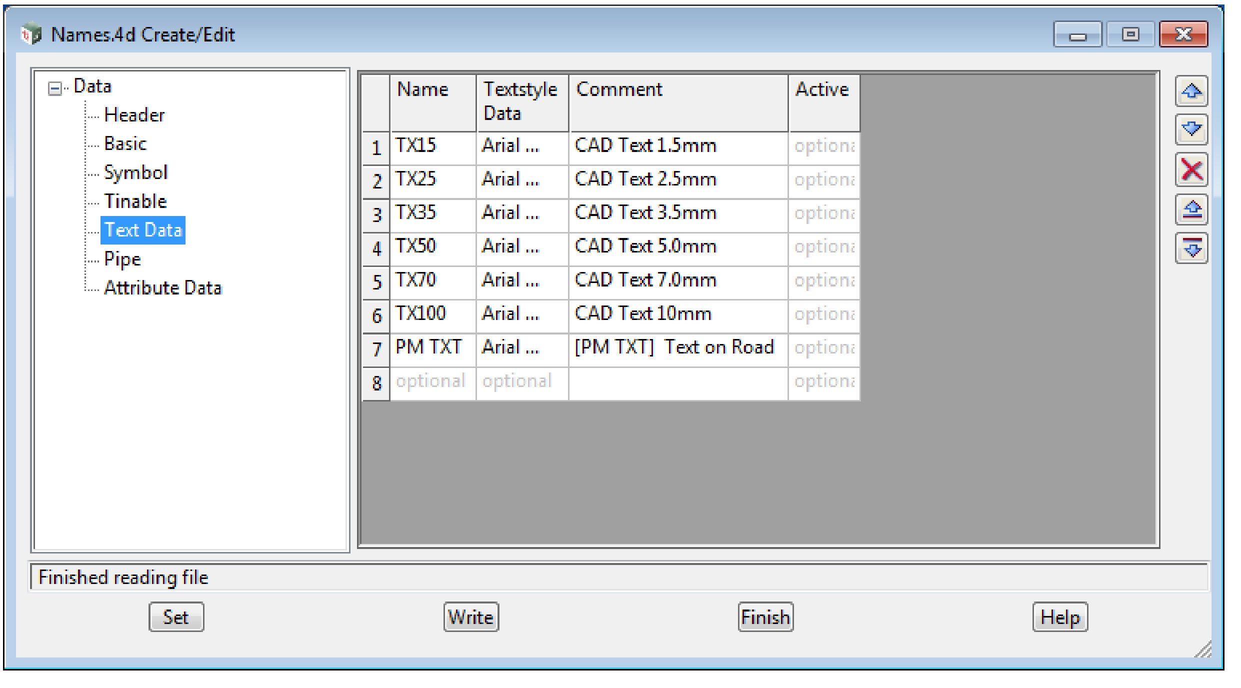
See [8.8.2 Comment and Active Column in Grids](#).

Go to the next section [7.9.1.5 Text Data Node](#) or return to [7.9.1 Name Mappings - Names.4d File](#).

7.9.1.5 Text Data Node

The **Text Data** grid selects data using either the typed in value in the **Name** field of the CAD ControlBar or any other appropriate **Name** panel fields. Or, if there was a substituted *Name* from the **Names** column in the **Basic** node, then that is used.

The **Att Key column** is not used.



Processing Using Name in the Text Data Node

*the typed in value, or **Name** if one has been substituted from the **Name** column in the **Basic** node, is searched against the list of names in the **Name** column until a **first** match with* Name *is found.*

If a match is found, then any non-blank data in the grid is used to fill out the appropriate panel fields or toolbar fields. The Name *column an include wild cards (*) and/or wild characters. The **Att Key** column is not used.*

If no match is found then none of the grid fields are used.

Textstyle Data textstyle Data grid

*when the CAD ControlBar **Name** field is being used, the* **Textstyle Name** *and the* **Textstyle Size** *from the **Textstyle Data** is used in the fields in the Text ControlBar.*

*When a **Name** field is used in other appropriate panels, the **Textstyle Data** is used in Textstyle Data fields.*

Comment and Active

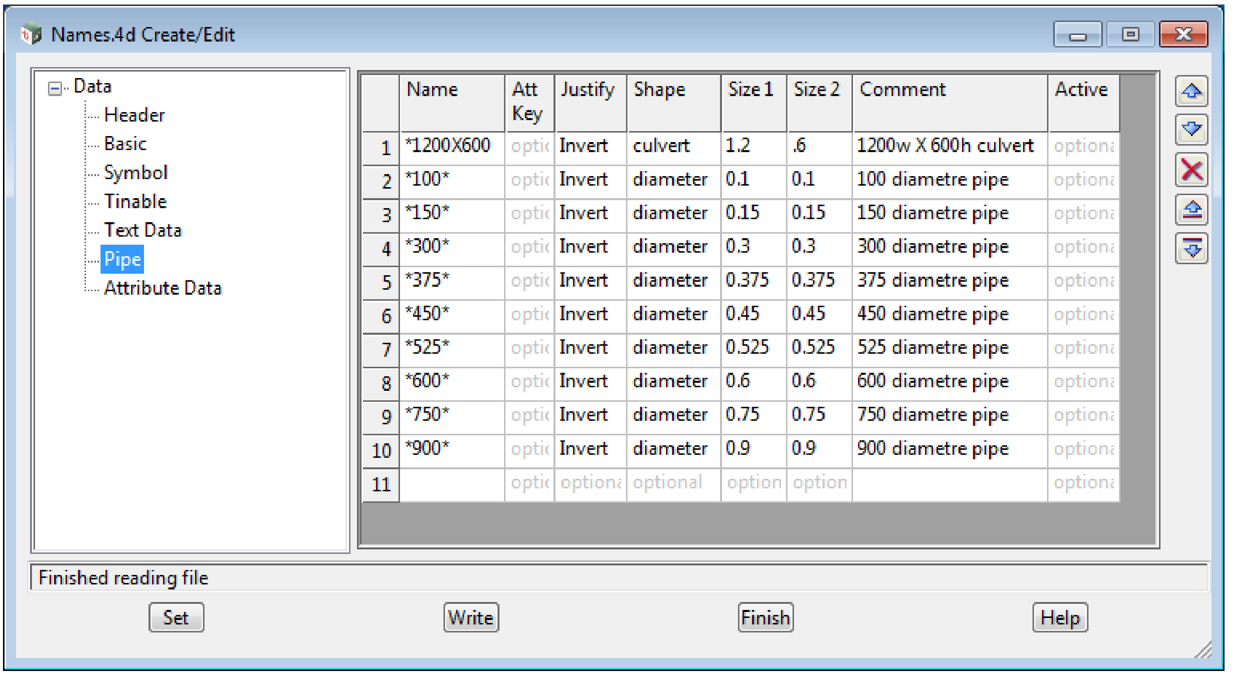
See [8.8.2 Comment and Active Column in Grids](#).

Go to the next section 7.9.1.6 Pipe Node or return to 7.9.1 Name Mappings - Names.4d File.

7.9.1.6 Pipe Node

The **Pipe** grid selects data using either the typed in value in the **Name** field of the CAD ControlBar or any other appropriate **Name** panel fields. Or, if there was a substituted *Name* from the **Names** column in the **Basic** node, then that is used.

The **Att Key column** is not used.



	Name	Att Key	Justify	Shape	Size 1	Size 2	Comment	Active
1	*1200X600	optic	Invert	culvert	1.2	.6	1200w X 600h culvert	optiona
2	*100*	optic	Invert	diameter	0.1	0.1	100 diametre pipe	optiona
3	*150*	optic	Invert	diameter	0.15	0.15	150 diametre pipe	optiona
4	*300*	optic	Invert	diameter	0.3	0.3	300 diametre pipe	optiona
5	*375*	optic	Invert	diameter	0.375	0.375	375 diametre pipe	optiona
6	*450*	optic	Invert	diameter	0.45	0.45	450 diametre pipe	optiona
7	*525*	optic	Invert	diameter	0.525	0.525	525 diametre pipe	optiona
8	*600*	optic	Invert	diameter	0.6	0.6	600 diametre pipe	optiona
9	*750*	optic	Invert	diameter	0.75	0.75	750 diametre pipe	optiona
10	*900*	optic	Invert	diameter	0.9	0.9	900 diametre pipe	optiona
11		optic	optiona	optional	option	option		optiona

Processing Using Name in the Pipe Node

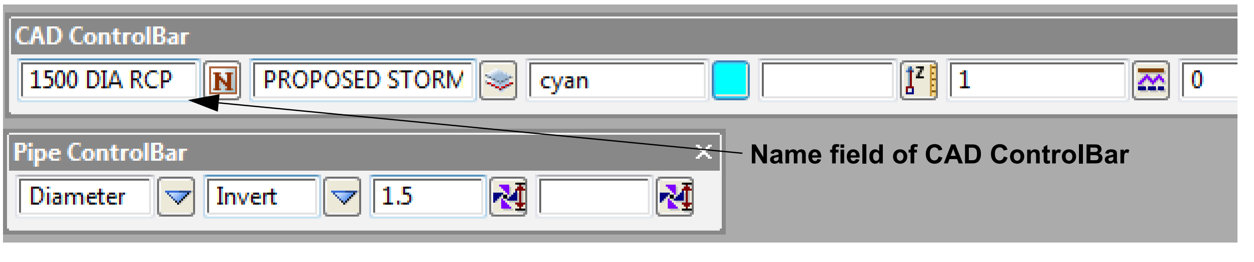
*the typed in value, or **Name** if one has been substituted from the **Name** column in the **Basic** node, is searched against the list of names in the **Name** column until a **first** match with* Name *is found.*

If a match is found, then any non-blank data in the grid is used to fill out the appropriate panel fields or toolbar fields. The Name *column an include wild cards (*) and/or wild characters. The **Att Key** column is not used.*

If no match is found then none of the grid fields are used.

Justify, Shape, Size 1, Size 2 Pipe grid

*when the CAD ControlBar **Name** field is being used, the* **Justify**, **Shape**, **Size 1** *and* **Size 2** *from the **Pipe grid** are used in the fields in the Pipe ControlBar.*



*Note that there does not need to be a match in the **Basic** section of Names.4d, just a match in the* **Pipe** *section of **Names.4d**.BUT if it is not in the **Basic** section of the Names.4d file then it won't appear in the **Names** pop up and the name will have to be typed into the **Name** field by hand followed by <Enter>.*

*When a **Name** field is used in other appropriate panels, the grid data is used in Pipe fields.*

Comment and Active

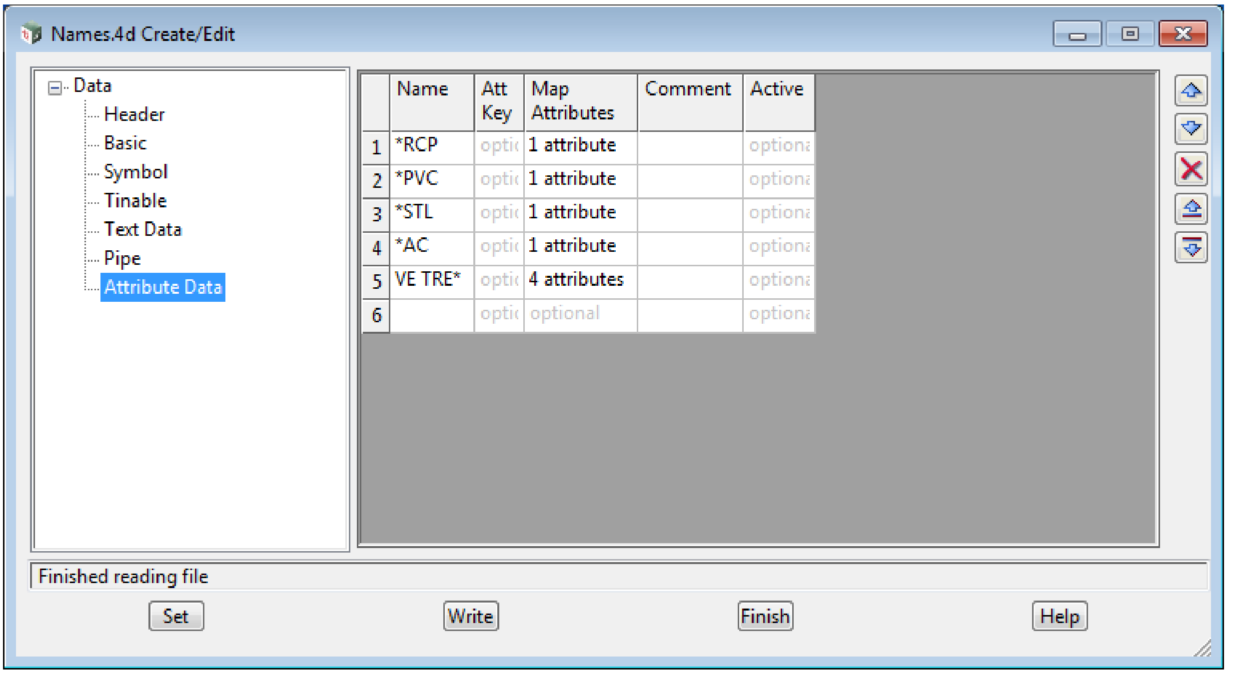
See 8.8.2 Comment and Active Column in Grids.

Go to the next section 7.9.1.5 Text Data Node or return to 7.9.1 Name Mappings - Names.4d File.

7.9.1.7 Attributes Data Node

The **Attribute Data** grid selects data using either the typed in value in the **Name** field of the CAD ControlBar or any other appropriate **Name** panel fields. Or, if there was a substituted *Name* from the **Names** column in the **Basic** node, then that is used.

The **Att Key column** is not used.



Processing Using Name in the Attribute Node

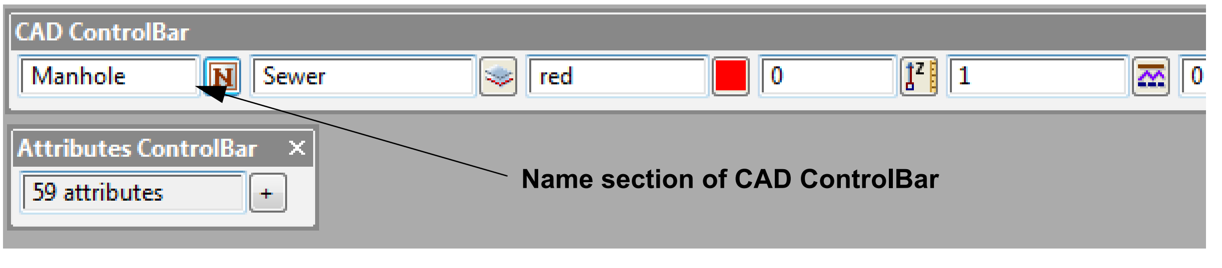
*the typed in value, or **Name** if one has been substituted from the **Name** column in the **Basic** node, is searched against the list of names in the **Name** column until a **first** match with* Name *is found.*

If a match is found, then any non-blank data in the grid is used to fill out the appropriate panel fields or toolbar fields. The Name *column an include wild cards (*) and/or wild characters. The **Att Key** column is not used.*

If no match is found then none of the grid fields are used.

Map Attributes Attributes grid

*when the CAD ControlBar **Name** field is being used, the* **Attributes** *from the **Map Attributes column** are used in the field in the Attributes ControlBar.*



*Note that there does not need to be a match in the **Basic** section of Names.4d, just a match in the* **Attributes Data** *section of **Names.4d**, BUT if it is not in the **Basic** section of the Names.4d file then it won't appear in the **Names** pop up and the name will have to be typed into the **Name** field by hand followed by <Enter>.*

*When a **Name** field is used in other appropriate panels, the Attributes are used in Attribute fields.*

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

Return to 7.9.1 Name Mappings - Names.4d File.

7.9.2 Plotters

User defined plotters can be set up in ***12d Model*** and the information is stored in the file plotters.4d which is fully documented in the section 43.2 Defining Plotters - Plotters.4d in the Appendix 43 Plotters and Plotting.

The interactive editor **Plotters** on the Browse option is not yet fully implemented or documented.

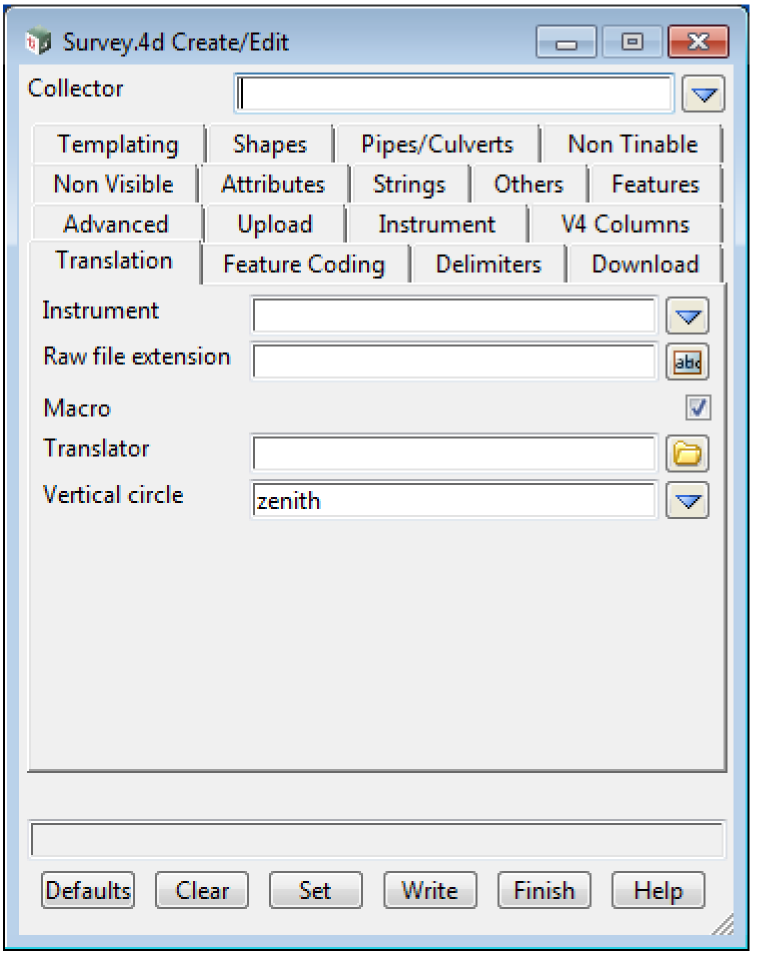
Go to the next section 7.9.3 Survey Data Collectors or return to 7.9 Tree.

7.9.3 Survey Data Collectors

Position of option on menu: **Project =>Tree**

Click on *Survey data collectors* to expand the item and then double click LB on *Create data collector* to create a new data collector, or double click LB on an existing data collector in the list to edit an existing data collector definition.

The **Survey.4d Create/Edit** panel will then appear.



This option is fully documented in the 36.9 Data Collector Definitions section in the Appendix 36 12d Survey Guide.

Go to the next section 7.9.4 Linestyles or return to 7.9 Tree.

7.9.4 Linestyles

Position of option on menu: **Project =>Tree**

Click on *Linestyles* to expand the item and then double click LB on *Create linestyle* to create, edit or delete linestyles, or double click LB on an existing textstyle in the list to edit an existing linestyle definition. The **Linestyle Create/Edit/Delete** panel will then appear.

The linestyle can be created interactively in ***12d Model*** by drawing the new linestyle using strings containing lines, arcs, circles and text. This information can then be used to create the linestyle.

To edit a linestyle, the current definition is written to a model which can then be edited to create the modified linestyle.

The definition of a linestyle includes:

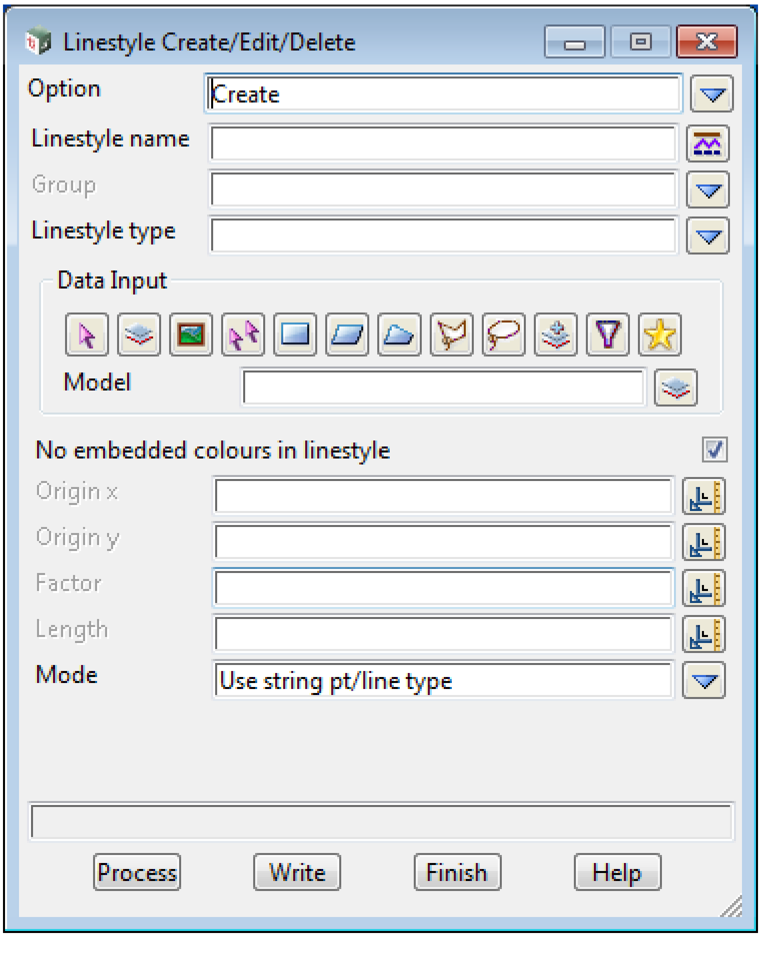
(a) a unique name

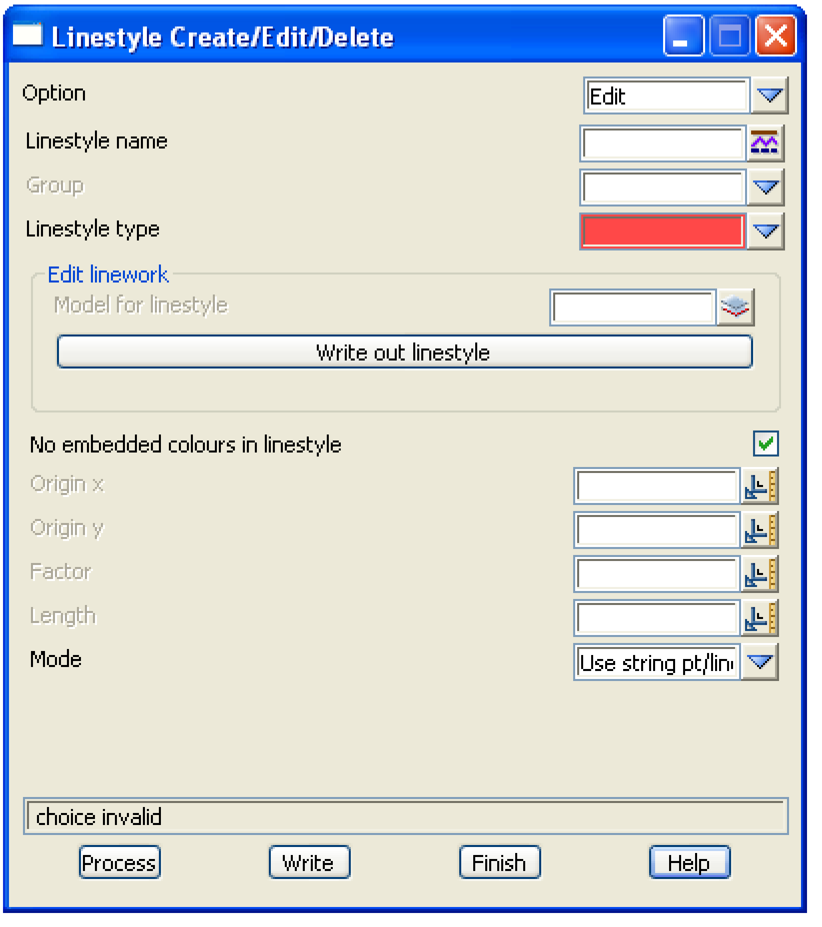
(b) the Group to show the linestyle under in the linestyle pop-up

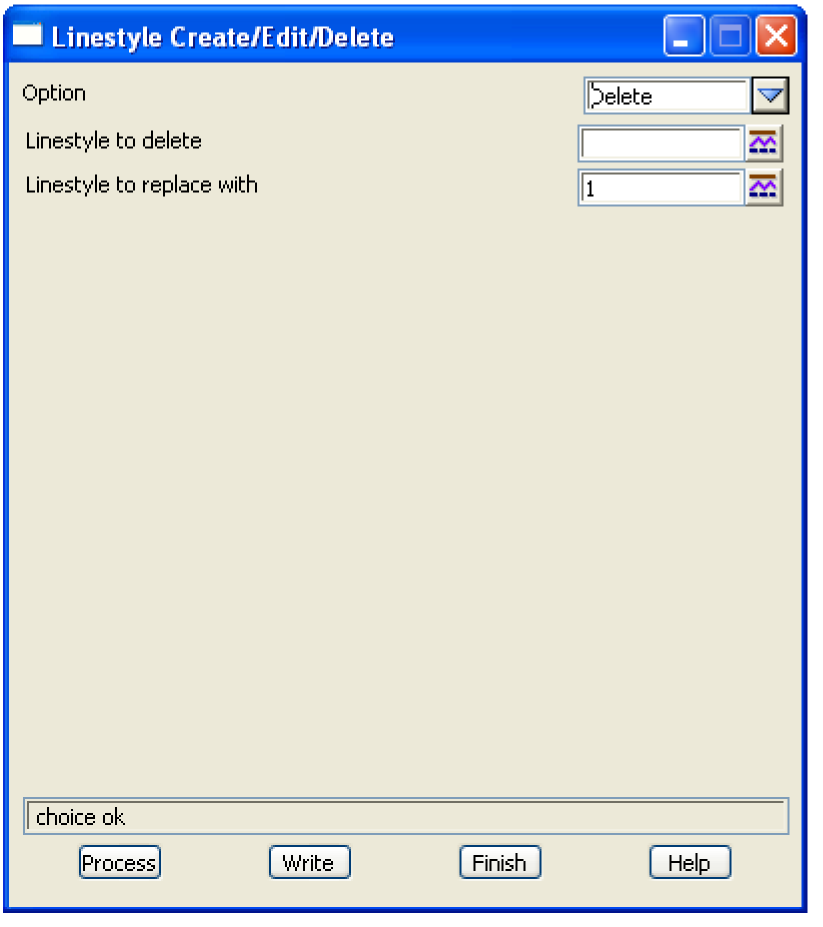
(c) the linestyle type

All the information for linestyles is stored in the text file linestyl.4d but the **Linestyle Create/Edit/Delete** panel is normally used to create and modify the file rather than using a text editor. (the file *linestyl.4d* is fully documented in the 40.1 Line Styles section of the Appendix 40 Linestyles, Symbols, Textstyles &Patterns).

The **Linestyle Create/Edit/Delete** panel is:




Linestyle Create/Edit/Delete
Option
Edit
Linestyle name
Group
Linestyle type
Edit linework
Model for linestyle
Write out linestyle
No embedded colours in linestyle
Origin x
Origin y
Factor
Length
Mode
Use string pt/lin
choice invalid
Process
Write
Finish
Help



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Option	choice box	Create	Create, Edit, Delete
type of edit			
Linestyle name	input		available linestyles
name of the linestyle to be created, edited or deleted.			
Group	input		available groups
name of the group for the linestyle - can be a new group name.			
Linestyle type	choice box	world	user, pixel, world, paper, *2 point, group*
type of the linestyle.			

For Option choice "*Create*" - Data input

Data input	source box		model
data to be used to create the linestyle.			

For Option choice "*Edit*" - Edit line work

Model for linestyle	model box		available models
model to write the linestyle out to so that it can be edited.			
Write out linestyle	button		
when selected, the linestyle is written out to the **Model for linestyle**.			

For Option choice *"Delete"*

Linestyle to delete linestyle box available linestylesl

name of the linestyle to delete.

Linestyle to replace it with linestyle box 1available linestyles

when the linestyle is deleted, any string with that linestyle has to have a new linestyle. This is the linestyle that is set for strings with the deleted linestyle

For the Linestyle type - User, Pixel, World, Paper

No imbedded colours in linestyle tick box tick

*if **ticked** then no colours from the data are used in the linestyle.*
*If **not ticked** then any colours used in the data are used in the linestyle*

Origin x/y double box

if non blank, the x/y value for the origin of the linestyle.
If blank, a value is calculated by 12d.

Factor double box

if non blank, the linestyle is factored up by this value.
If blank, the factor is taken to be 1.

Length double box

if non blank, the linestyle is repeated after this length.
If blank, the linestyle length is calculated by 12d.

Mode choice box use string pt/line type
only at vertices, repeat style

*if **use string pt/line type**, then if the string pt/line type is **line** then the linestyle is repeated after the Length value. If the string pt/line type is **point**, then the linestyle is only drawn at each vertex of the string.*
*If **only at vertices**, then the linestyle is only drawn at each string vertex regardless of the strings pt/line type.*
*If **repeat style**, then the linestyle is repeated after the Length value regardless of the strings pt/line type.*

For the Linestyle type - 2 point

No imbedded colours in linestyle tick box tick

*if **ticked** then no colours from the data are used in the linestyle.*
*If **not ticked** then any colours used in the data are used in the linestyle*

Origin x/y double box

if non blank, the x/y value for the first origin of the linestyle.
If blank, a value is calculated by 12d.

Origin 2 x/y double box

if non blank, the x/y value for the second origin of the linestyle.
If blank, a value is calculated by 12d.

Stretch in choice box one direction, both directions

if "one direction", the linestyle is only stretched along the axis joining the two original points. The linestyle is not stretched perpendicular to that axis.
If "both directions", the linestyle is stretched in all directions.

Drawn on choice box every line/arcevery line/arc,
every second line/arc

if "one direction", the linestyle is only stretched along the axis joining the two original points. The linestyle is not stretched perpendicular to that axis.
If "both directions", the linestyle is stretched in all directions.

For Linestyle type - Group

Linestyle grid

list of linestyles that make up the this one linestyle.

Buttons

Process button

process the data in the panel fields. The internal definitions of the linestyles is modified. The definitions are not written to the linestyle file.

Write button

write the modifications to the file linestyl.4d file. This files can then be used when projects are opened up. The information won't be used for the current session unless the **Process** *button is selected as well. For more information on the* **Write** *button, go to the section* 39.2.6 Writing Out Setup Files *in the Appendix* 39 Setting Up & Configuring 12d

This option updates the files *linestyles.4d* which is fully documented in the 40.1 Line Styles section in the Appendix 40 Linestyles, Symbols, Textstyles &Patterns.

Go to the next section 7.9.5 Symbols or return to 7.9 Tree.

7.9.5 Symbols

Position of option on menu: **Project =>Tree**

Click on *Symbols* to expand the item and then double click LB on *Create symbol* to create, edit or delete symbols, or double click LB on an existing symbol in the list to edit an existing symbol definition. The **Symbol Create/Edit/Delete** panel will then appear.

The symbol can be created interactively in ***12d Model*** by drawing the new symbol using strings containing lines, arcs, circles and text. This information can then be used to create the symbol.

To edit a symbol, the current definition is written to a model which can then be edited to create the modified symbol.

The definition of a symbol includes:

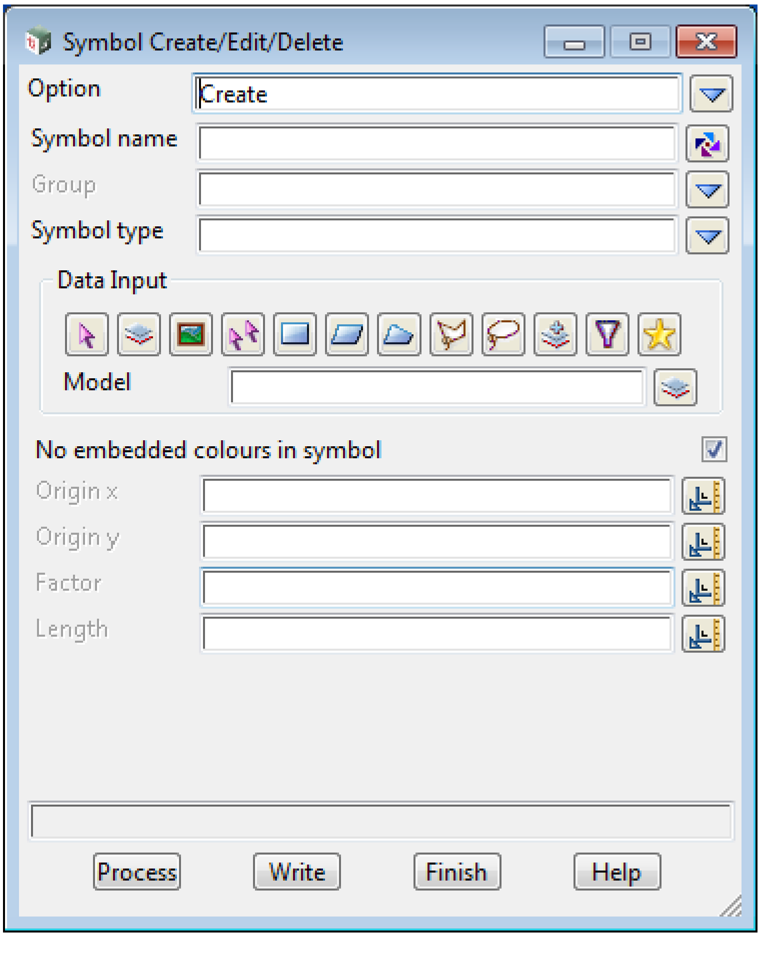
(a) a unique name

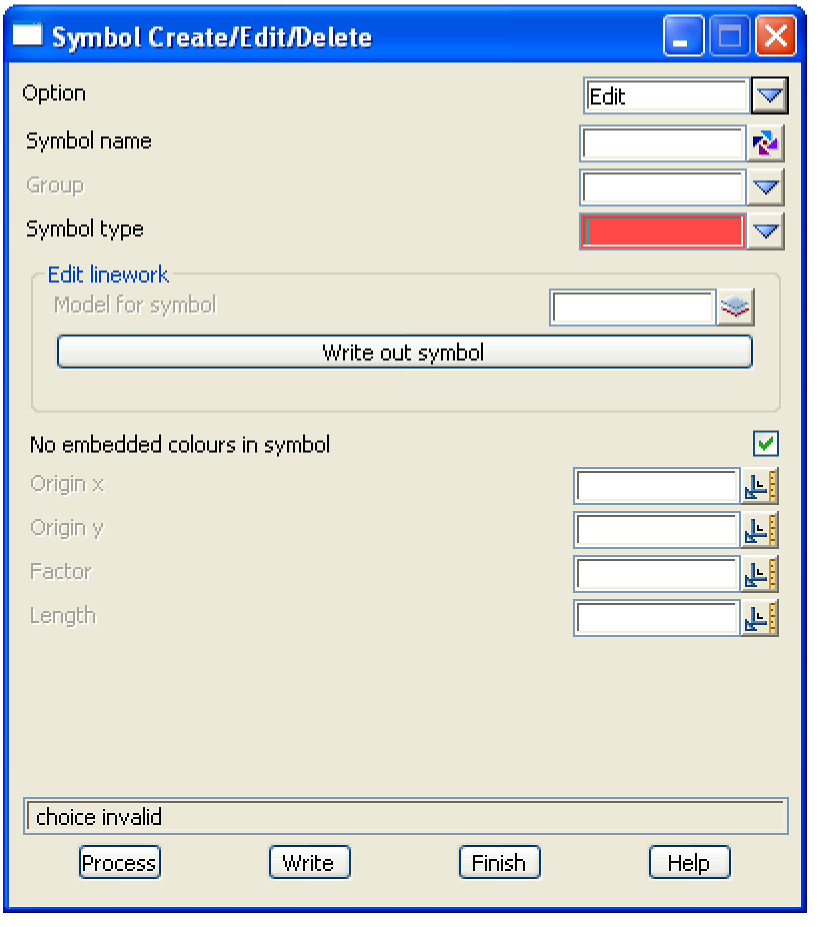
(b) the Group to show the symbol under in the symbol pop-up

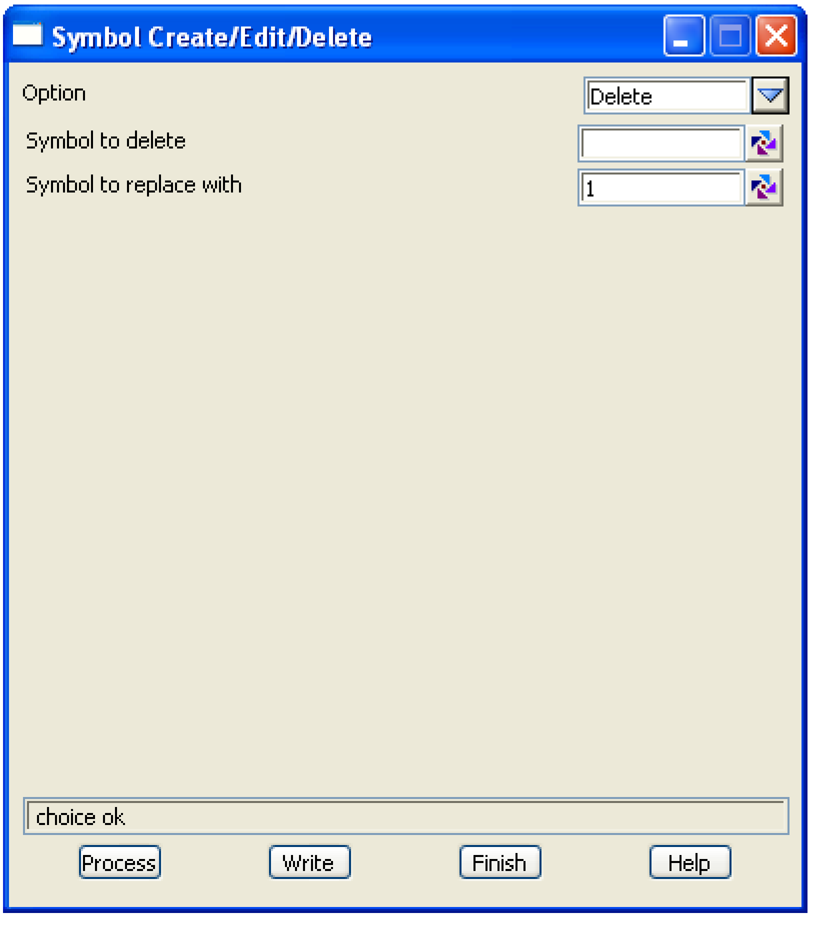
(c) the symbol type

All the information for linestyles is stored in the text file symbols.4d but the **Symbol Create/Edit/Delete** panel is normally used to create and modify the file rather than using a text editor. (the file *symbols.4d* is fully documented in the 40.1 Line Styles section of the Appendix 40 Linestyles, Symbols, Textstyles &Patterns).

The **Symbol Create/Edit/Delete** panel is:




Symbol Create/Edit/Delete
Option
Edit
Symbol name
Group
Symbol type
Edit linework
Model for symbol
Write out symbol
No embedded colours in symbol
Origin x
Origin y
Factor
Length
choice invalid
Process
Write
Finish
Help



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Option	choice box	Create	Create, Edit, Delete
type of edit			
Symbol name	input		available symbols
name of the linestyle to be created, edited or deleted.			
Group	input		available groups
name of the group for the symbol - can be a new group name.			
Symbol type	choice box	world	pixel, world, paper
type of the symbol.			

For Option choice "*Create*" - Data input

Data input	source box		model
data to be used to create the symbol.			

For Option choice *"Edit"* - Edit line work

Model for symbol	model box		available models
model to write the symbol out to so that it can be edited.			
Write out symbol	button		
when selected, the symbol is written out to the **Model for symbol**.			

For Option choice *"Delete"*

Symbol to delete symbol box available symbols

name of the linestyle to delete.

Symbol to replace it with symbol box 1available symbols

when the symbol is deleted, any vertex of a string with that symbol has to have a new symbol. This is the symbol that is set for strings with the deleted symbol

No imbedded colours in symbol tick box tick

*if **ticked** then no colours from the data are used in the symbol.*
*If **not ticked** then any colours used in the data are used in the symbol*

Origin x/y double box

if non blank, the x/y value for the origin of the symbol.
If blank, a value is calculated by 12d.

Factor double box

if non blank, the symbol is factored up by this value.
If blank, the factor is taken to be 1.

Length double box

if non blank, the symbol is repeated after this length.
If blank, the symbol length is calculated by 12d.

Process button

process the data in the panel fields. The internal definitions of the symbols is modified. The definitions are not written to the symbol file.

Write button

write the modifications to the file symbols.4d file. This files can then be used when projects are opened up. The information won't be used for the current session unless the **Process** *button is selected as well. For more information on the* **Write** *button, go to the section* 39.2.6 Writing Out Setup Files *in the Appendix* 39 Setting Up & Configuring 12d

This option updates the files *symbols.4d* which is fully documented in the 40.2 Symbols section of the Appendix 40 Linestyles, Symbols, Textstyles &Patterns.

Go to the next section 7.9.6 Textstyles or return to 7.9 Tree.

7.9.6 Textstyles

Position of option on menu: **Project =>Tree**

Click on *Textstyles* to expand the item and then double click LB on *Create textstyle* to create a new textstyle, or double click LB on an existing textstyle in the list to edit an existing textstyle definition. The **Create/Edit Textstyle** panel will then appear.

The textstyle definition includes

(a) a unique name

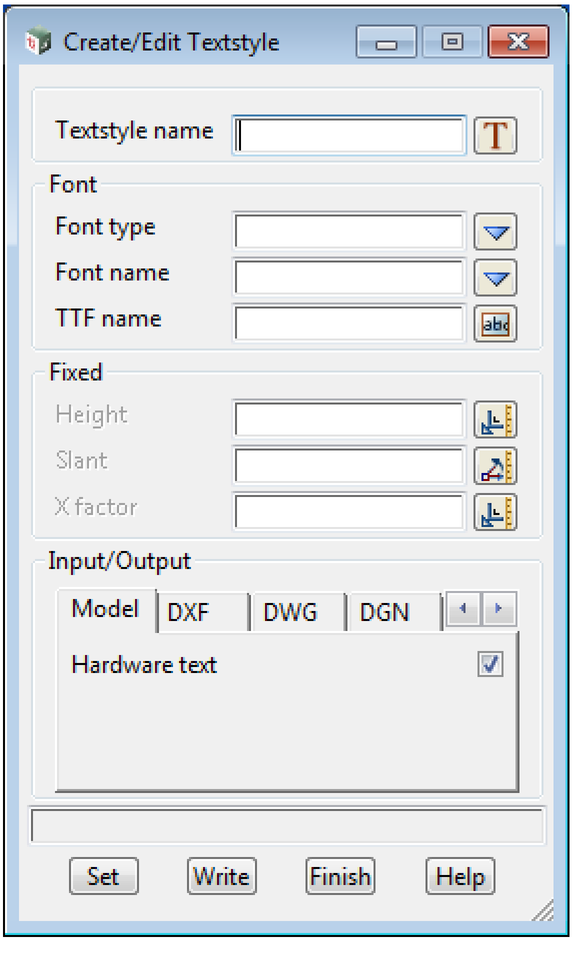
(a) the font used for textstyle - this may be a true type font

(b) mappings of textstyle names when reading in data from AutoCAD and Microstation

(c) mappings of textstyle names when writing data out to AutoCAD, Microstation and ***12d Model*** Models.

All the information for textstyles and fonts is stored in text files (textstyl.4d and fonts.4d) but the **Create/Edit Textstyle** panel is normally used to create and modify the files rather than using a text editor. (the files *textsyles.4d* and *fonts.4d* which are fully documented in the 40.3 Textstyles and Fontssection of the Appendix 40 Linestyles, Symbols, Textstyles &Patterns).

The **Create/Edit Textstyle** panel is:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Textstyle name	input		available textstyles

name of the textstyle to be defined or edited.

Font section

Font name output recent projects

name of the font to be used for the textstyle.

True type font tick box tick

*if **ticked** then the font is a true type font. The next box asks for the True Type Font name.*
*If **not ticked** then the font needs an SHP file to define the characters. The next box asks for the SHP name.*

TTF name input available true type fonts

name of the true type font.

SHP name input available SHP files

name of the SHP file to define the characters.

Fixed section

Input/Output section

Model tab

Hardware text tick box tick

*if **ticked** then any **12d Model** text is written out to a model as text strings.*
*If **not ticked** then each characters of a text string is turned into individual lines and arcs (i.e. each character is stroked).*

DXF tab

Hardware text tick box tick

*if **ticked** then any **12d Model** text is written out to the DXF file as text.*
*If **not ticked** then each characters of a text string is turned into individual lines and arcs (i.e. each character is stroked).*

Input name input

*when a DXF file is read in, any text in the DXF file with style of this **Input name,** is converted to **12d Model** text strings with the textstyle **Textstyle name**.*

Output name input

*if the **Hardware text** flag is set to tick - when data is written out to a DXF file, any **12d Model** text with the textstyle of the name in the field **Textstyle name**, is converted to DXF text with style of this **Output name**.*

*If the **Hardware text** flag is not set to tick, then this field is ignored.*

DWG tab

Hardware text tick box tick

*if **ticked** then any **12d Model** text is written out to the DWG file as text.*
*If **not ticked** then each characters of a text string is turned into individual lines and arcs (i.e. each character is stroked).*

Input name input

*when a DGN file is read in, any text in the DGN file with style of this **Input name**, is converted to **12d Model** text strings with the textstyle **Textstyle name**.*

Output name input

*if the **Hardware text** flag is set to tick - when data is written out to a DWG file, any **12d Model** text with the textstyle of the name in the field **Textstyle name**, is converted to DWG text with style of this **Output name**.*

*If the **Hardware text** flag is not set to tick, then this field is ignored.*

DGN tab

Hardware text tick box tick

*if **ticked** then any **12d Model** text is written out to the DGN file as text.*
*If **not ticked** then each characters of a text string is turned into individual lines and arcs (i.e. each character is stroked).*

Input name input

*when a DGN file is read in, any text in the DGN file with style of this **Input name**, is converted to **12d Model** text strings with the textstyle **Textstyle name**. For DGN, the text style must be a number between 1 and 64.*

Output name input

*if the **Hardware text** flag is set to tick - when data is written out to a DGN file, any **12d Model** text with the textstyle of the name in the field **Textstyle name**, is converted to DGN text with style of this **Output name.** For DGN, the text style name must be a number between 1 and 64.*

*If the **Hardware text** flag is not set to tick, then this field is ignored.*

Other tab

Hardware text tick box tick

*if **ticked** then any **12d Model** text is written out to the file as text.*
*If **not ticked** then each characters of a text string is turned into individual lines and arcs (i.e. each character is stroked).*

Input name input

*when a file is read in, any text in the file with style of this **Input name**, is converted to **12d Model** text strings with the textstyle **Textstyle name**.*

Output name input

*if the **Hardware text** flag is set to tick - when data is written out to a file, any **12d Model** text with the textstyle of the name in the field **Textstyle name**, is converted to text with style of this **Output name**.*

*If the **Hardware text** flag is not set to tick, then this field is ignored.*

Buttons

Set button

sets the definition for this editing session of the project - the information will be lost when the project is exited.

Write button

write the information to the files textstyle.4d and fonts.4d files. These files can then be used when projects are started. The information won't be used for the current session unless the Set button is selected as well. For more information on the **Write** *button, go to the section* 39.2.6 Writing Out Setup Files *in the Appendix* 39 Setting Up & Configuring 12d

This option updates the files *textsyles.4d* and *fonts.4d* which are fully documented in the 40.3 Textstyles and Fontssection of the Appendix 40 Linestyles, Symbols, Textstyles &Patterns.

Go to the next section 7.9.7 Textstyle Data Favourites or return to 7.9 Tree.

7.9.7 Textstyle Data Favourites

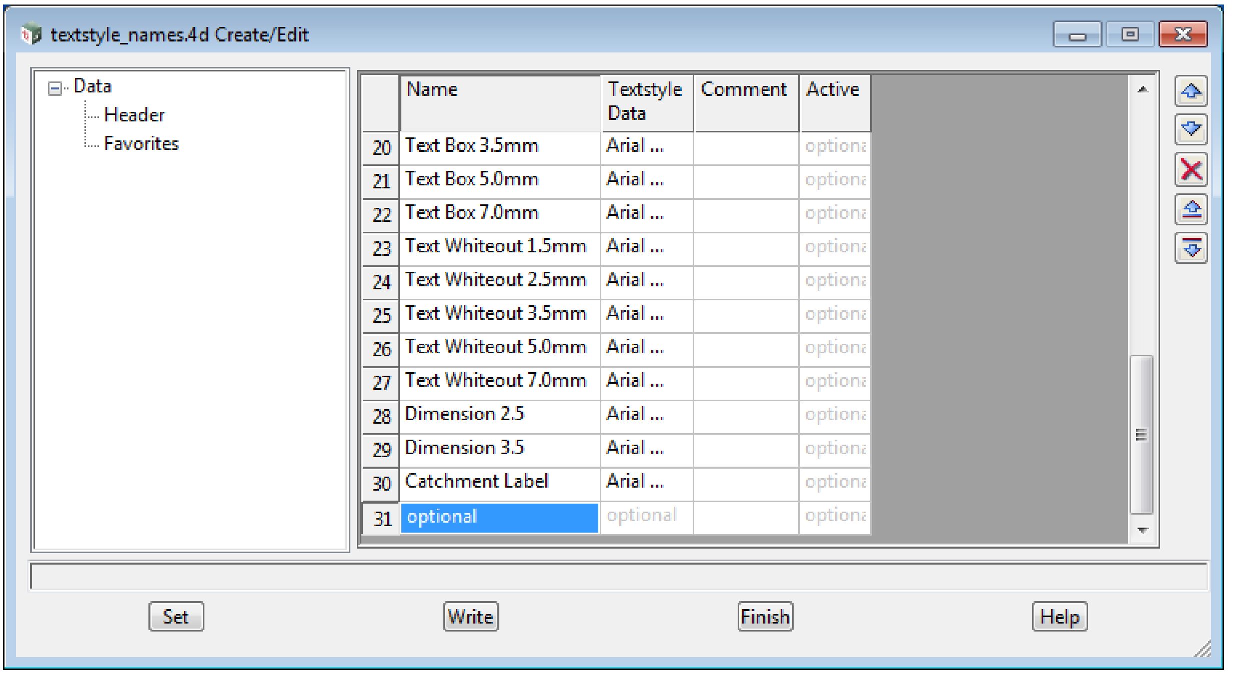
Position of option on menu: **Project =>Tree**

The *textstyle data favourites* option defines a set of text style parameters recorded with a user given name. When ever a text style needs to be defined, the name of a *textstyle data favourite* can be selected and the values of the favourite used for the text.

To define or modify a textstyle favourite, click on *Textstyle data favourites* in the *Browse* list to expand the item and then double click LB on *Create textstyle data favourite* to create/edit the *textstyle_names.4d* file.

Double clicking LB on any of the items below *Create textstyle data favourite* will also create/edit the *textstyle_names.4d* file.

In either case, the **Textstyle_names.4d Create/Edit** panel will then appear.



Buttons

Set button

sets the definition for this editing session of the project - the information will be lost when the project is exited.

Write button

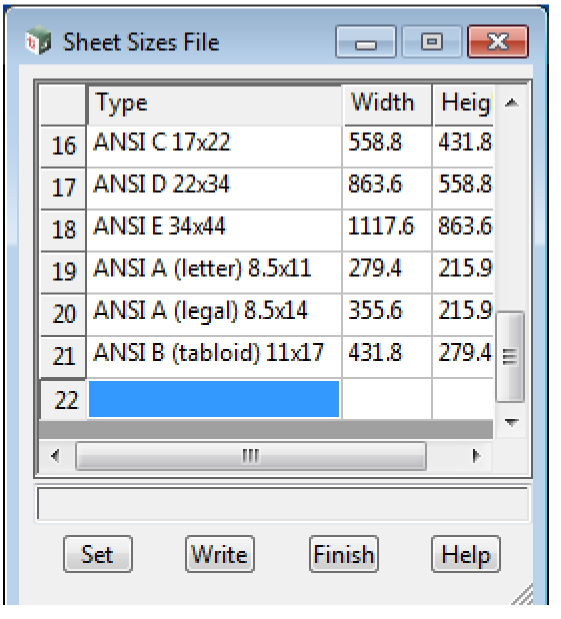
write the information to the textstyle_names.4d file. The file can then be used when projects are started. The information won't be used for the current session unless the Set button is selected as well. For more information on the **Write** *button, go to the section* 39.2.6 Writing Out Setup Files *in the Appendix* 39 Setting Up & Configuring 12d.

Go to the next section 7.9.8 Sheet Sizes or return to 7.9 Tree.

7.9.8 Sheet Sizes

Position of option on menu: **Project =>Tree**

Click on *Sheet sizes* to expand the item and then double click LB on *Create sheet size* to bring up the **Sheet Sizes File** panel to create, edit or delete sheet size definitions, or double click LB on an existing sheet size in the list to also bring up the **Sheet Sizes File** panel.



Buttons

Set button

sets the definition for this editing session of the project - the information will be lost when the project is exited.

Write button

write the information to the file sheets.4d. The file can then be used when projects are started. The information won't be used for the current session unless the **Set** *button is selected as well. For more information on the* **Write** *button, go to the section* 39.2.6 Writing Out Setup Files *in the Appendix* 39 Setting Up & Configuring 12d

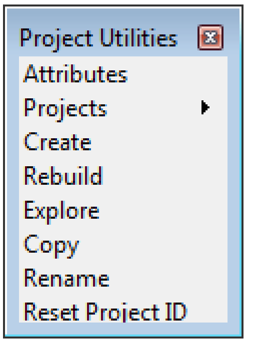
This option updates the file *sheets.4d* which is fully documented in the 39.2.7.4 Sheet Sizes File (sheets.4d) section of the Appendix 39 Setting Up & Configuring 12d.

Return to 7.9 Tree.

7.10 Utilities

Position of menu: **Project =>Utilities**

The **utilities** walk-right menu contains various project items.



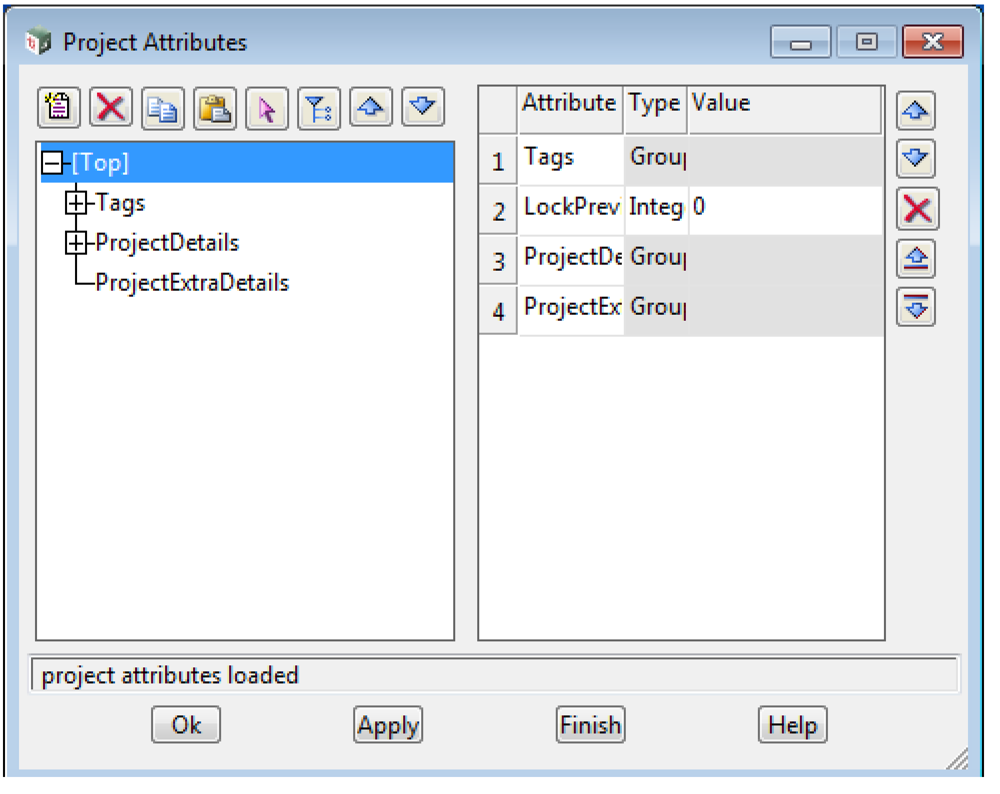
For the option *Attributes*, go to	7.10.1 Attributes
Projects	7.10.2 Projects
Create	7.10.3 Create
Rebuild	7.10.4 Rebuild
Explore	7.10.5 Explore
Copy	7.10.6 Copy
Rename	7.10.7 Rename
Reset Project Id	7.10.8 Reset Project ID

7.10.1 Attributes

Position of option on menu: **Project =>Utilities =>Attributes**

The **Attributes** options displays, creates and edits the attributes for the current project

On selecting **Attributes**, the **Project Attributes** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name			
name of the attribute			
Type	integer, real, text		
type of attribute - integer, real or text			
Data			
value for the attribute			
OK	button		
set the attributes to the values in the panel and then exit the panel.			
Apply	button		
set the attributes to the values in the panel but don't exit the panel.			

Go to the next section 7.10.2 Projects or return to 7.10 Utilities.

7.10.2 Projects

Position of option on menu: **Project =>Utilities =>Projects**

The **projects** walk-right menu provides a list of all the projects available in the working folder. Each project is contained in a sub-folder of the working folder.

Selecting a project from the walk-right list will bring up the **Change Project** panel with the selected project name already in the **New project** panel field.

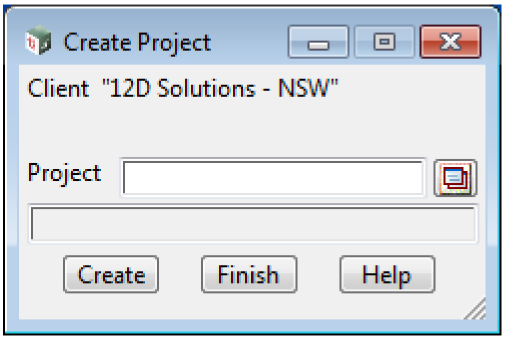
For more in formation on the **Change Project** panel, go to the earlier section 7.2.2 Open.

Go to the next section 7.10.3 Create or return to 7.10 Utilities.

7.10.3 Create

Position of option on menu: **Project =>Utilities =>Create**

On selecting the **Create** option, the **Create Project** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Project	input		

name of the new project to be created.

Create	button		

after selecting this button, a new project folder is created. The working project does not change to the new project.

Go to the next section 7.10.4 Rebuild or return to 7.10 Utilities.

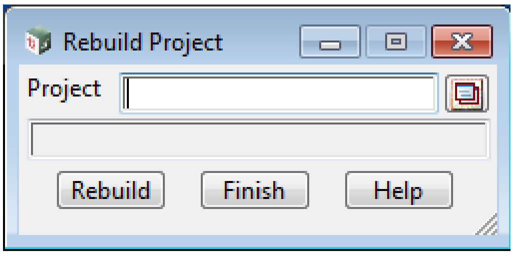
7.10.4 Rebuild

Position of option on menu: **Project =>Utilities =>Rebuild**

The **rebuild** option is used to try and rebuild a project if some of the information has been corrupted.

On selecting the option, the project header file will be deleted and re-created and all the models, tins, templates and functions inside the project's folder added back into the project. The view information for the original project will be lost.

On selecting the **rebuild** option, the **rebuild project** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Project	input		

the name of the project to be rebuilt.

Rebuild	button		

after selecting this button, the project given in the **project** *field is rebuilt.*

Note - the current (working) project cannot be rebuilt.

Go to the next section 7.10.5 Explore or return to 7.10 Utilities.

7.10.5 Explore

Position of option on menu: **Project =>Utilities =>Explore**

The **Explore** brings up the Windows Explorer showing the contents of the working folded. That is, showing the contents of the folder *containing* the project.

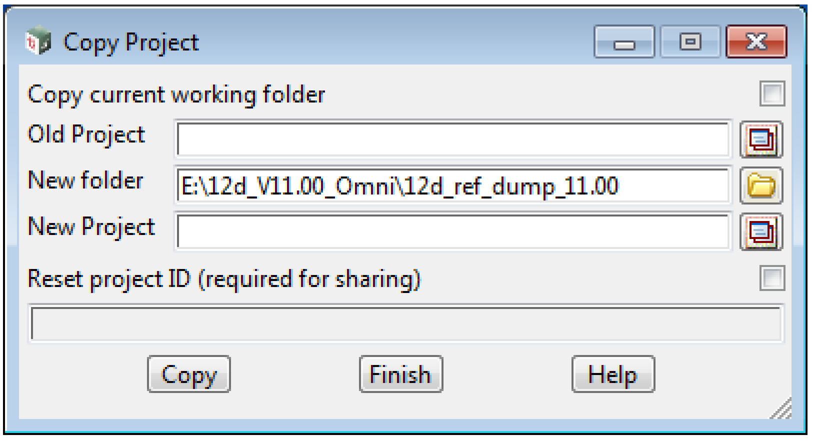
Go to the next section 7.10.6 Copy or return to 7.10 Utilities.

7.10.6 Copy

Position of option on menu: **Project =>Utilities =>Copy**

The **copy** option is used to make a copy of any project (other than the current project) in the working folder.

On selecting the **copy** option, the **copy project** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Copy current working folder	tick box		none

if ticked, the entire working folder containing the project is copied to the **New folder***.*

Old project	input	none	projects in folder

name of the project to be copied.

New folder	input		current working folder

name of the folder to copy the project to.

New project	input		

name of the copy of the project. The new project name cannot be the same as an existing project in the **New folder***.*

Reset project ID (required for sharing) tick box

if ticked, 12d will automatically reset the project id in the new copy of the project. This will prevent multiple projects with the same id, which are not allowed when sharing from one project to another. if not ticked, 12d will not reset the project id in the new copy of the project.

Copy	button		

after selecting this button, a complete copy of the project given in the **old project** *field is made and saved under the name given in the* **new project** *field. If* **Copy current working folder** *is ticked, the contents of the working folder containing the project is also copied.*

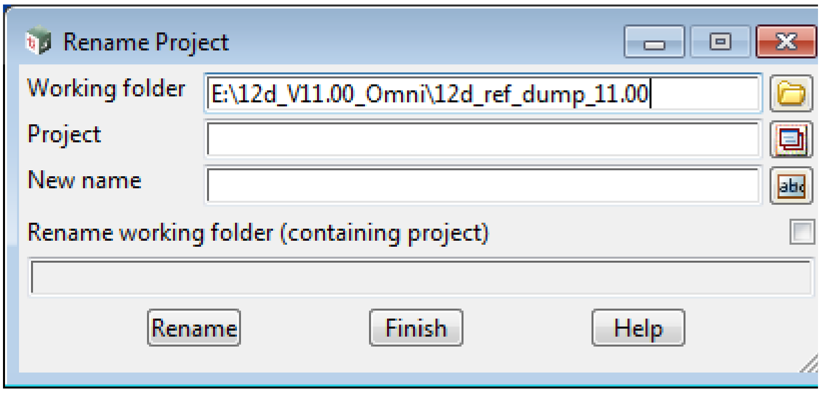
Go to the next section 7.10.7 Rename or return to 7.10 Utilities.

7.10.7 Rename

Position of option on menu: **Project =>Utilities =>Rename**

The **Rename** option is used to rename any project (other than the current project) in the working folder, and if required, the working folder as well.

On selecting **Rename**, displays the **Rename project** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Working folder folder box folder browse

name of the folder that contains the project to be renamed (the working folder for the project).

Project project box projects in working folder

name of the project to rename - this can't be the project currently opened.

New name input

name of the new name for the project. The new project name cannot be the same an existing project in the working folder.

Rename working folder (containing the project) tick box

if tick, if the working folder containing the project to be renamed is the same as the project name, then the working folder will also be renamed.

Rename button

after selecting this button, the selected project will be given the new name. If **Rename working folder (containing project)** *is ticked and the name working folder is the same as the project, then the working folder will also be renamed.*

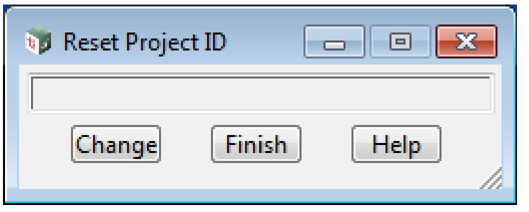
Go to the next section 7.10.8 Reset Project ID or return to 7.10 Utilities.

7.10.8 Reset Project ID

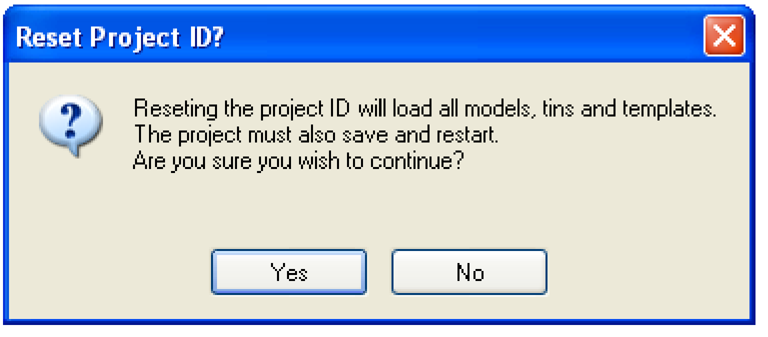
Position of option on menu: **Project =>Utilities =>Reset Project ID**

This option is used when a project has the same id as another project. If a user wishes to share from a server project that is a copy of another project, then the project ID must be reset.

On selecting the **Reset Project ID** option, the **Reset Project ID** panel is displayed.



Selecting **Change** displays the **Reset Project ID?** panel on the screen.



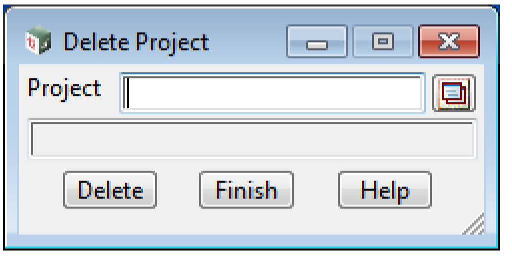
Selecting the **Yes** button will cause the project to resave and ***12d Model*** will restart.

Go to the next section 7.11 Delete or return to 7.10 Utilities.

7.11 Delete

Position of option on menu: **Project =>Delete**

On selecting the **delete project** option, the **delete project** panel is displayed.



The fields and buttons used in the delete project panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Project	input		available projects

name of project to delete

Delete	button		

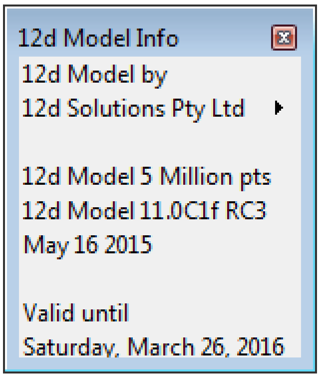
after selecting this button, a **yes-no** *pop-up menu appears to confirm that deleting the project is required. If deletion is confirmed, the selected project is deleted from the computer disk.*

Go to the next section 7.12 12d Model or return to 7 Project.

7.12 12d Model

Position of option on menu: Project =>12d Model

The **12d model** walk-right menu and the walk-right **by** menu are



12d Solutions phone, fax etc.

12d Model version
date version compiled

12d Solutions phone, fax etc.

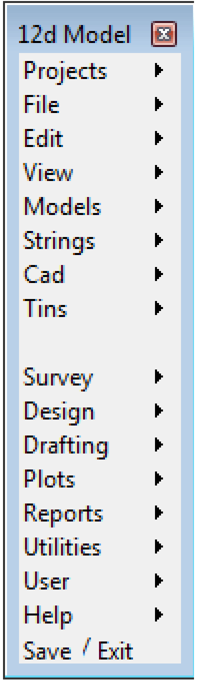
The walk-right menu for **Originators** is:

Go to the next section 7.13 12d Model Menu or return to 7 Project.

7.13 12d Model Menu

Position of option on menu: Project =>12d Model menu

Selecting the **12d Model Menu** option brings up the floating **12d Model** menu as described at the beginning of this chapter.



For the options, see

Projects	7 Project
File I/O	8 File
Edit	11 Edit
Views	12 View
Models	13 Models
Strings	14 Strings
Triangles	16 Tins
Survey	17 Survey
Design	20 Design
Drafting	24 Drafting
Plots	25 Plots
Reports	27 Reports
Utilities	28 Utilities
User	29 User
Help	31 Help
Save/ *Exit*	32 Save and Exit

Go to the next section 7.14 Exit or return to 7 Project.

7.14 Exit

Selecting **Exit** exists ***12d Model***

Also see 32 Save and Exit

Return to 7 Project.

8 File

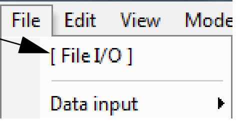
Position of main menu: **File**

The facilities for reading and writing data files, layout files, textstyle definitions, template files into and out of 12d Model, plus screen dumps are collected under the **File I/O** menu.

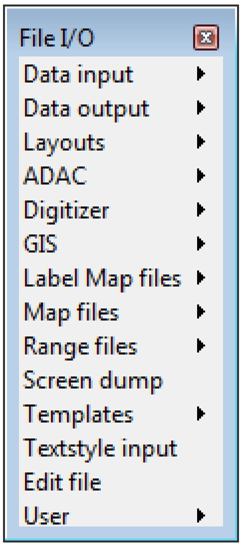
The **File I/O** walk-right menu containing these options is:

on Main menu

create floating
File I/O menu



on 12d Model menu and
floating File I/O menu



read in data files
write out data files
read/write screen layout files
create, read and write ADAC XML files
digitizing option
GIS interface
read/write Label Map Files
read/write Map Files
read/write range files
screen dump of window
read/write design templates
read in textstyle definitions file
edit a file
File I/O User menu

See

Data input	8.1 Data Input
Data output	8.2 Data Output
Layouts	8.3 Layouts
ADAC	10.4 12d ADAC Menu
Digitizer	8.5 Digitizer
GIS	8.6 GIS
Label Map files	8.7 Label Map Files
Map files	8.8 Map Files
Range files	8.9 Range Files
Screen dump	8.10 Screen Dump
Templates	8.11 Templates
Textstyle input	8.12 Textstyle Input
Edit file	8.13 Edit a File

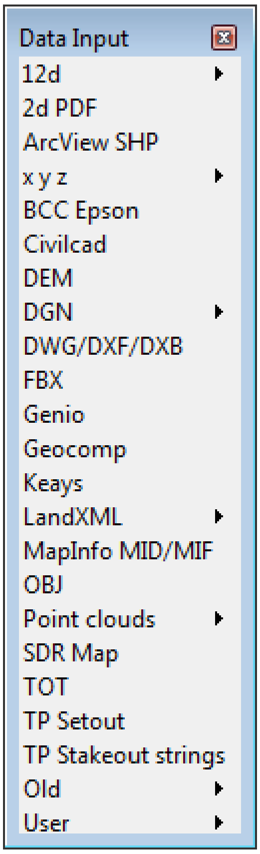
8.1 Data Input

Position of menu: **File I/O =>Data input**

The facilities for reading data files into 12d Model are collected under the **File I/O =>Data input** menu.

Some of the formats are provided in the base product (xyz data, BCC Epson, 12d Model HP plots and 12d Model 12da files) and the rest are optional (dxf, genio, geocomp, *etc*.).

The default *Input null value* is described in the section 8.1.1 Input Null Value.



bring up the 12d Input menu
read a 2d PDF file
ArcView shape files
read in x y z s and attribute data
BCC Epson data file
Civilcad text files
USGS DEM files
dgn V7 and V8 binary files
dwg/dxf Version 12 to 2010
FBX file
genio data
Geocomp.pts,.str data
Keays data
LandXML
MapInfo mid/mif files
Wavefront OBJ
LAS, SRTM files
SDR Map file
TOT files
TP Setout .pta file
TP Stakeout strings file
Old readers
Data Input User menu

The default *Input null value* is described in the section 8.1.1 Input Null Value

For the options:

12d	8.1.2 12d Input
2d PDF	8.1.3 Import 2D PDF
ArcView SHP	8.1.4 ArcView SHP Input
x y z	8.1.5 Input X Y Z Text Files
BCC Epson	8.1.6 BCC Epson Input
Civilcad	8.1.7 CivilCad Input
DEM	8.1.8 DEM Input
DGN	8.1.9 Input DGN Binary Files
DWG/DXF/DXB	8.1.10 DWG/DXF Input
FBX	8.1.11 FBX Input
Genio	8.1.12 Genio Input
Geocomp	8.1.13 Geocomp Input
Keays	8.1.14 Keays Input

8.1.1 Input Null Value

In three dimensional data, it is possible that a point can have a valid plan position but an undefined height. In 12d Model, there is a special **null** value (-9.9e29) which is used internally when height is undefined.

In other software systems, the null value may be different.

To allow for different null values, 12d Model has an **i/o null height** parameter.

As data is read in, the height is checked and if it is equal to the **i/o null height**, then it is replaced by the 12d Model null height.

The **i/o null height** is set in **Default Settings** tab of the **Defaults** panel in the **Project =>Management =>Defaults** (see 7.6.1 Defaults).

Note - some input panels have their own special **null value** field which is used instead of the i/o null height.

Return to 8.1 Data Input

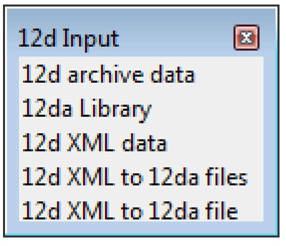
8.1.2 12d Input

Position of menu: **File I/O =>Data input =>12d**

The facilities for reading data files into 12d Model are collected under the **File I/O =>Data input =>12d** menu.

Some of the formats are provided in the base product (xyz data, BCC Epson, 12d Model HP plots and 12d Model 12da files) and the rest are optional (dxf, genio, geocomp, *etc*.).

The default *Input null value* is described in the section 8.1.1 Input Null Value



12d Model archive file
insert a 12d archive file
read 12d XML file
convert 12d XML files to 12da files
convert 12d XML files to a 12da file

For the options:

12da archive data	8.1.2.1 12d Archive Input
12da Library	8.1.2.2 12da Library
12d XML data	8.1.2.3 12d XML Data
12d XML to 12da files	8.1.2.4 12d XML to 12da files
12d XML to 12d file	8.1.2.5 12d XML to 12da file

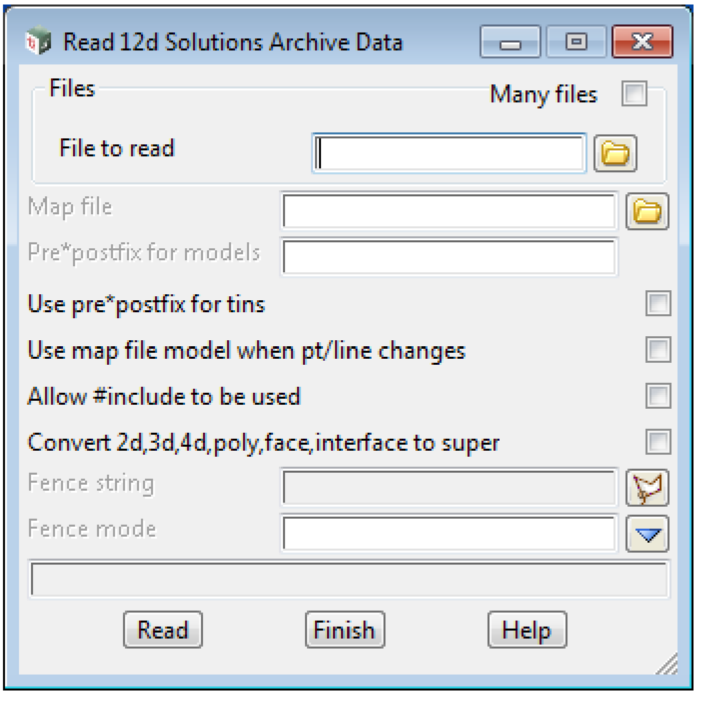
8.1.2.1 12d Archive Input

Position of option on menu: **File I/O =>Data input =>12d =>12d archive data**

The ***12d Archive*** format is a special format defined by 12d Solutions to allow data to be easily transferred from other programs into 12d Solutions software such as 12d Model. The 12d Archive format is given in the 34 12d Archive File Format.

Note that there is also an XML format for writing out the data in ***12d Model***. See 35 12d XML File Format.

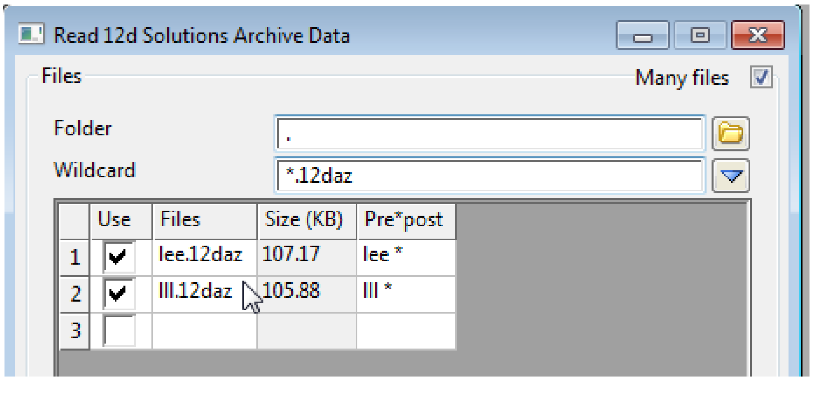
Selecting **12d archive data** brings up the **Read 12d Solutions Archive Data** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Many files	tick box		

*if **ticked**, a grid to allow multiple 12d archive files to be read in, is opened. A wild card is used to select all the files to be read in.*



Folder	folder box	

folder to search for files using the **Wild card**

Wildcard	text box	

wild card to use in search for files in the given folder

Use tick box

*if **ticked**, read in the file*

Files file box

name of the file in the folder

Size (KB) output only

file size

Pre*post text box

*if **non blank**, pre*post text to use for the models in this 12d archive file (see* 4.19.2 Pre*Postfix in Panel Fields*) for information on using pre*postfix.*
*If **blank**, use the pre*post text from the* **Pre*postfix for models** *panel field.*

***Note** - if a non-blank value for **Pre*post is** given in the column for a file then the **Pre*postfix for models** is ignored.*

File to read file box *.12daz, *.12da or *.4da files

name of the **12d Model** *Archive file to be read in.*

*Note that **Drag and Drop** works for 12daz, 12da and 4da files.*
It will also work for files that are email attachments in Outlook 2002 and above.

*If you are dragging and dropping more than one **12daz, 12da** or **4da** file at a time, then one panel will be opened and all the dropped files listed in the **Many Files** grid.*

Map file file box *.mapfile, *.mf files

*if **not blank**, the name of the 12d Map File to be used for all strings read in, including any files given with the **Many files** mode ticked on.*
*If **blank**, no map file is used*

When using a map file, the string name is used as the entity-name for matching with the keys in the map file. See the section 8.8.1 Create/Edit a Map File *for information about 12d map files.*

Pre*postfix for models pre*postfix box

if non-blank, a prefix and a postfix to be applied to the model names used in the map file.
Go to the section 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

***Note** - if a non-blank value for **Pre*post** is given in the column for a file then the **Pre*postfix for models** is ignored.*

Use pre*postfix for tins tick box not ticked

if ticked, a prefix and a postfix are to be applied to any tin names in the 12d Archive data.
Go to the section 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

***Warning** - if a tin already exists in 12d Model with the tin name, then the tin cannot be read in from the 12d Archive file.*

Use map file model when pt/line changes tick box

*if **not ticked** and the pt/line type of the string does not match that in the map file, then the string is placed in.*
*If **tick**, the.*

Allow #include to be used tick box

if ticked, expand an files referenced by an #include.
If not ticked, ignore the #includes

Convert 2d, 3d, 4d, poly, face, interface to super tick box

if ticked, non super string versions of 2d/3d/4d/poly/face/interface strings are converted to super strings.

Fence string polygon box

A polygon is selected/created and used with the **Fence mode** *choice to restrict the selection of strings* *read in from the 12d Archive file.*

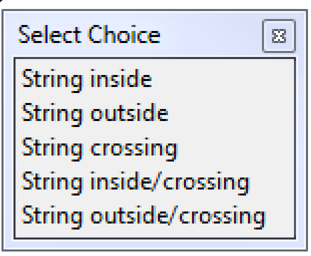
Clicking LB on the Select Polygon icon on the right hand side of the **Fence string** *field allows the user to select a polygon.*

Clicking RB on the Select Polygon icon on the right hand side of the **Fence string** *field brings up the* ***Polygon Choice*** *Box for the user to select a method of creating/selecting a polygon.*

Clicking MB does nothing.

Fence mode choice box available choices

The strings selected are then restricted to those using the polygon and satisfying the **Fence mode**:



String inside - strings totally inside the polygon
String outside - strings totally outside the polygon
String crossing - strings crossing the polygon
String inside/crossing - strings totally inside or crossing the polygon
String outside/crossing - strings totally outside or crossing the polygon

Fence Modes

Read button

reads the data in.

<u>Special Note</u>

Pressing the <Esc> key will interrupt and terminate the reading in of the files.

8.1.2.2 12da Library

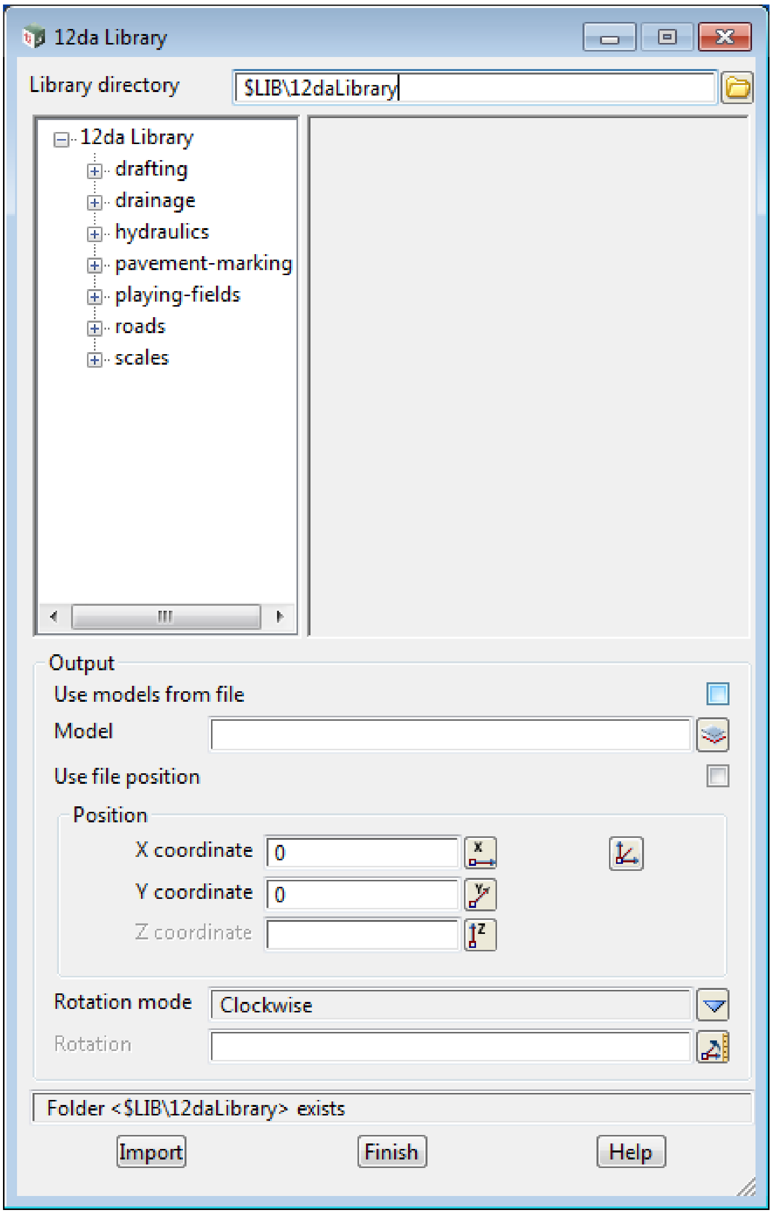
Position of option on menu: **File I/O =>Data input =>12d =>12da Library**

The **12da Library** allows a user to load 12da's from a 'library', defined by a directory structure.

Every folder within the nominated directory will be searched for 12da files, and this directory structure will be shown in the tree on the left hand side of the panel. When selected, the 12da will be shown in a preview on the right.

These 12das can be output into a nominated model, and a specific position, or you can choose to use the file position.

Selecting **12da Library** brings up the **12da Library** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Library directory	file		

the directory to search for 12das in

Use models from file tick box

if ticked use the models specified in the 12da

Model model box

if not using the models listed in the file, the model for the data in the 12da

Use file position tick box

if ticked use the file specified in the file position

Position x coordinate

the x-coordinate for the data, if not using the position in the file

Position y coordinate

the y-coordinate for the data, if not using the position in the file

Position z coordinate

an optional y-coordinate for the data, if not using the position in the file

Rotation mode choice box Clockwise, Anti- clockwise

how to apply an optional rotation (Clockwise or *Anti -clockwise)*

Rotation

an optional rotation to apply

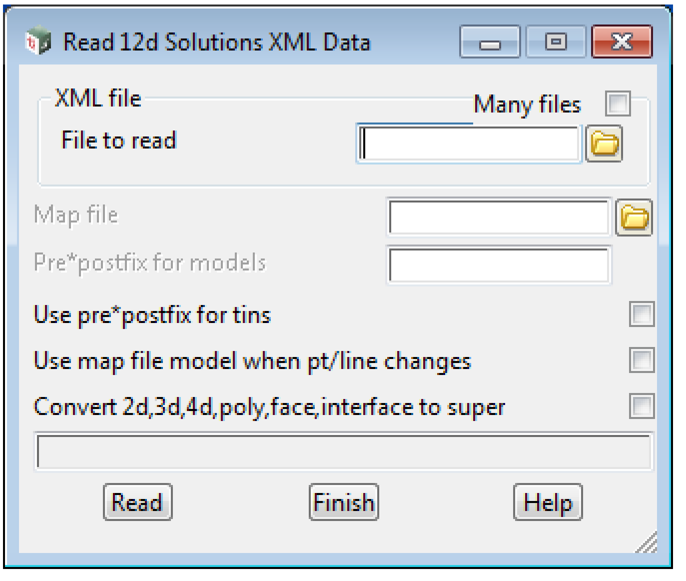
8.1.2.3 12d XML Data

Position of option on menu: **File I/O =>Data input =>12d =>12d XML data**

The ***12d XML*** format is a special format defined by 12d Solutions to allow data to be easily transferred from other programs into 12d Solutions software such as 12d Model. The **12d XML** format is given in the 35 12d XML File Format.

Note - ***12d XML*** contains the same information as a 12da file but it is a XML format.

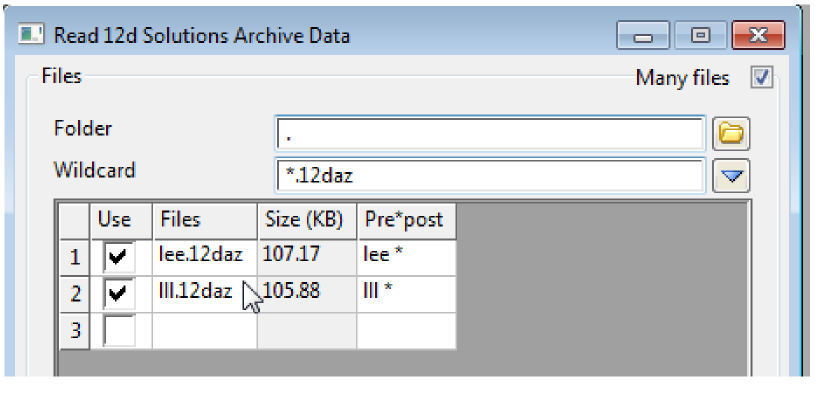
Selecting **12d XML data** brings up the **Read 12d Solutions XML Data** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Many files	tick box		

*if **ticked**, a grid to allow multiple 12d XML files to be read in, is opened. A wild card is used to select all the files to be read in.*



Folder folder box

folder to search for files using the **Wild card**

Wildcard text box

wild card to use in search for files in the given folder

Use tick box

*if **ticked**, read in the file*

Files file box

name of the file in the folder

Size (KB) output only

file size

Pre*post text box

*if **non blank**, pre*post text to use for the models in this 12d XML file (see* 4.19.2 Pre*Postfix in Panel Fields*) for information on using pre*postfix.*
*If **blank**, use the pre*post text from the* **Pre*postfix for models** *panel field.*

***Note** - if a non-blank value for **Pre*post is** given in the column for a file then the **Pre*postfix for models** is ignored.*

File to read file box *.12dxmlz, *.12dxml

name of the **12d Model** *XML file to be read in.*

*Note that **Drag and Drop** works for 12dxmlz and 12dxml files.*
It will also work for files that are email attachments in Outlook 2002 and above.

*If you are dragging and dropping more than one **12dxmlz** or **12dxml** file at a time, then one panel will be opened and all the dropped files listed in the **Many Files** grid.*

Map File file box *.mapfile, *.mf files

*if **not blank**, the name of the 12d Map File to be used for all strings read in, including any files given with the **Many files** mode ticked on.*
*If **blank**, no map file is used*

When using a map file, the string name is used as the entity-name for matching with the keys in the map file. See the section 8.8.1 Create/Edit a Map File *for information about 12d map files.*

Pre*postfix for models pre*postfix box

if non-blank, a prefix and a postfix to be applied to the model names used in the map file.
Go to the section 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

***Note** - if a non-blank value for **Pre*post** is given in the column for a file then the **Pre*postfix for models** is ignored.*

Use pre*postfix for tins tick box not ticked

if ticked, a prefix and a postfix are to be applied to any tin names in the 12d XML data.
Go to the section 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

***Warning** - if a tin already exists in 12d Model with the tin name, then the tin cannot be read in from the 12d XML file.*

Use map file model when pt/line changes tick box

*if **not ticked** and the pt/line type of the string does not match that in the map file, then the string is placed in.*
*If **tick**, the.*

Allow #include to be used tick box

if ticked, expand an files referenced by an #include.
If not ticked, ignore the #includes

Convert 2d, 3d, 4d, poly, face, interface to super tick box

if ticked, non super string versions of 2d/3d/4d/poly/face/interface strings are converted to super strings.

Read button

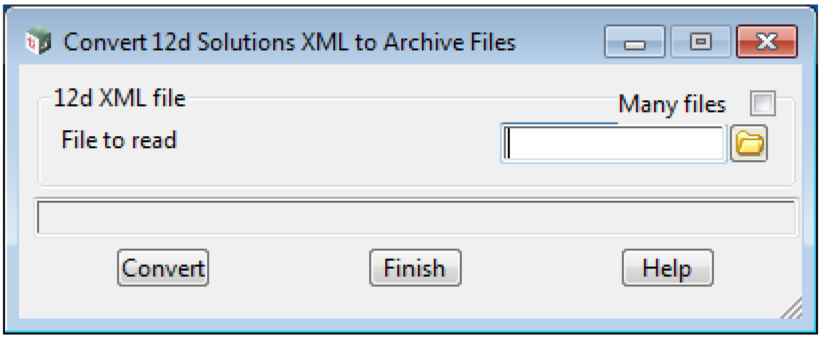
reads the data in.

8.1.2.4 12d XML to 12da files

Position of option on menu: **File I/O =>Data input =>12d =>12d XML to 12da files**

The ***12d XML to 12da*** option converts **12d XML** files to **12da** files.

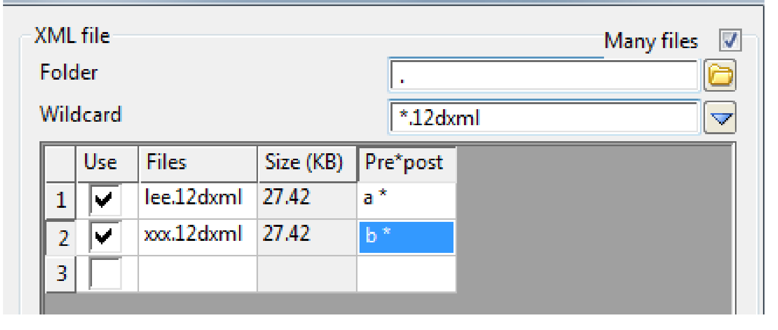
Selecting **12d XML to 12da files** brings up the **Convert 12d Solutions XML to Archive Files** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Many files	tick box		

*if **ticked**, a grid to allow multiple 12d XML files to be converted to 12d XML files. A wild card is used to select all the files to be converted.*



Folder folder box

folder to search for files using the **Wild card**

Wildcard text box

wild card to use in search for files in the given folder

Use tick box

*if **ticked**, convert the file*

Files file box

name of the file in the folder

Size (KB) output only

file size

Pre*post text box

*if **non blank**, pre*post text.*
*If **blank**, use.*

Convert button

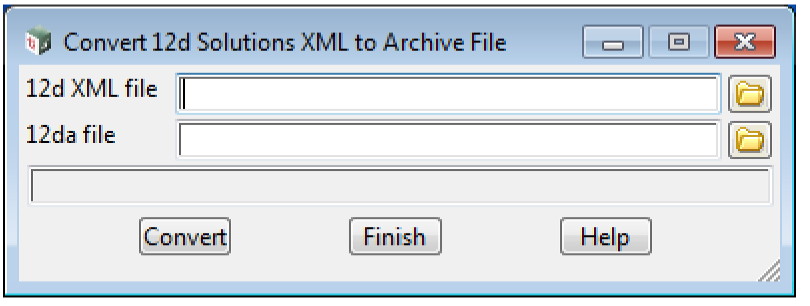
converts the 12d XML files to 12da files with the same name stem.

8.1.2.5 12d XML to 12da file

Position of option on menu: **File I/O =>Data input =>12d =>12d XML to 12da file**

The ***12d XML to 12da*** option converts **12d XML** file to a **12da** file.

Selecting **12d XML to 12da file** brings up the **Convert 12d Solutions XML to Archive File** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
XML file	file box		*.12dxmlz, *.12dxml

name of the 12d XML file to convert to a 12da file.

12da file	file box		*.12a files

name of the 12da file to convert the 12d XML file to.

Convert	button		

convert the 12d XML file to the 12da file.

8.1.3 Import 2D PDF

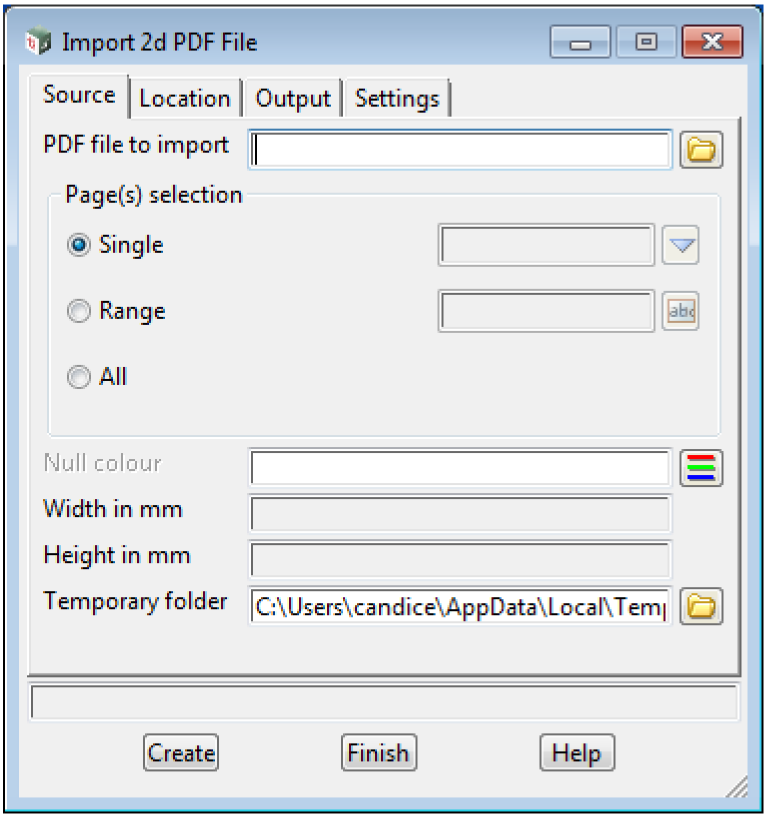
Position of option on menu: **File I/O =>Data input =>2d PDF**

This is option reads in a 2d pdf file and can create an image of the pdf and/or extract any line work (vectors) and text that may exist in the 2d pdf.

Note1

1. Although you can "see" lines and text in the raster does not mean that they exist inside the pdf as lines (vectors) and text data. They may be in the image only. This option **does not** do text and/or line recognition from the raster.
2. If the font for any text inside the pdf file does not exist in 12d, then another existing 12d font will be used. This can mean that the character positions are not correct.
3. if the font is encoded, then in some instances, the incorrect text will be displayed.

Selecting **Import 2d pdf** displays the **Import 2d PDF File** panel.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Source tab

PDF file to import file *.pdf files

name of the PDF file to import/read in.

Page(s) selection

only one of **Single**, **Range** *or* **All** *can be selected.*

Single radio button and choice box page numbers

if selected, this is the page number to import from the pdf file. The pop up has a list of pages and the page size (in mm) of each page in the pdf.

Range radio button, number box

if selected, the range of the page numbers to read in from the pdf file.

The page numbers and/or page ranges are separated by commas, and the pages in a range are separated by a minus. For example 1,3,5-12

All radio button

if selected, all the pages in the pdf file are imported.

Null colour colour box

*if **not blank**, the colour to use for the background colour in the pdf images.*
*If **blank**, leave the colours as they are in the pdf.*

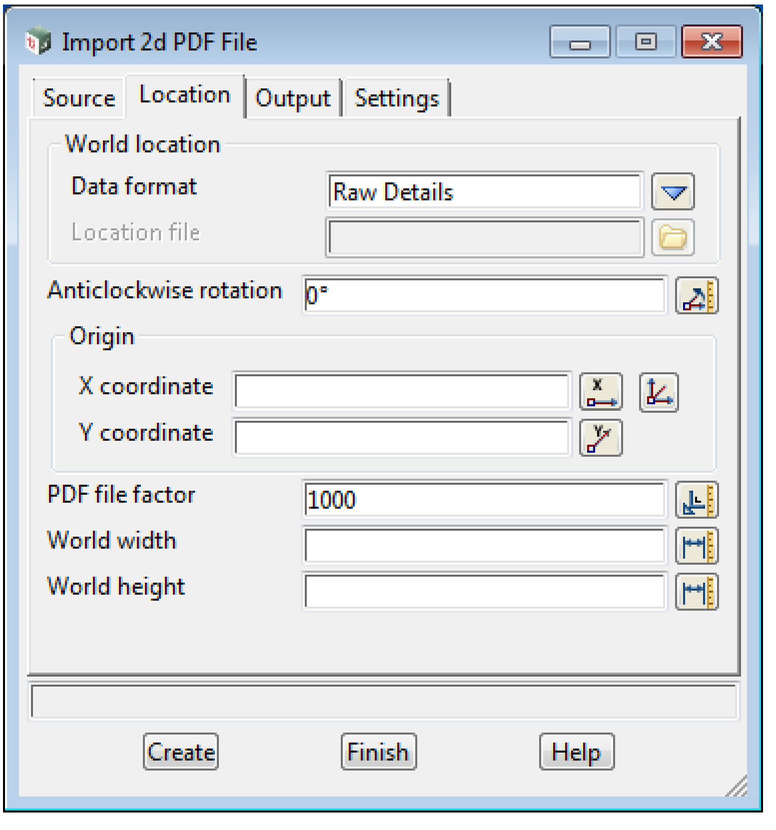
Width/Height in mm output only

the width/height (in mm) of the page selected from the pdf is written out to these fields.

Temporary folder folder box

the folder to use for temporary files created whilst processing the pdf.

Location tab



*The units in the pdf file are paper units and not world coordinates. The location tab supplies the information for positioning the data in the pdf file in world units by either typing in a world origin, anticlockwise rotation angle and world width and height, or by using supplying a file in **ESRI world** format with the information in it.*

Data format choice box Raw details Raw Details or ESRI world

*if **Raw details**, the location details are typed into the* **World Location Details** *section as rotation, world origin, world width and height.*

If **ESRI world**, *the location details are taken from the ESCRI world file given in* **Location file.** *An ESRI world file gives the xscale, row rotation, column rotation, yscale, x origin and y origin. For use with 12d, the row and column rotations must be the same and yscale = - xscale.*

Location file file box

*file with the location details if **ESRI world** is selected as the* **Data format**.

World Location information

*if **Raw Details** is selected as the* **Data format**, *the Anticlockwise rotation, (X,y) origin, World width, World height and Source file factor are used to position and scale the data in the pdf file.*

Anticlockwise rotation angle box

the world rotation to rotate the raster, text and vectors in the pdf file. The units are dms in hp notation and the rotation is measured in a counter clockwise direction from the positive x-axis.

X/Y coordinate real box

the world x/y coordinate of the left hand bottom corner of the raster in the pdf file.

Only one of **PDF file factor, World width**, *and* **World height** *is needed and the others are calculated.*
That is, if one of **PDF file factor**, **World width**, *or* **World height** *is typed and <Enter> pressed, then the other two are automatically calculated using the entered value and the **Width in mm** and **Height in mm** from the first page of the pdf file itself. However only the* **PDF file factor** *is used to scale the pdf's.*

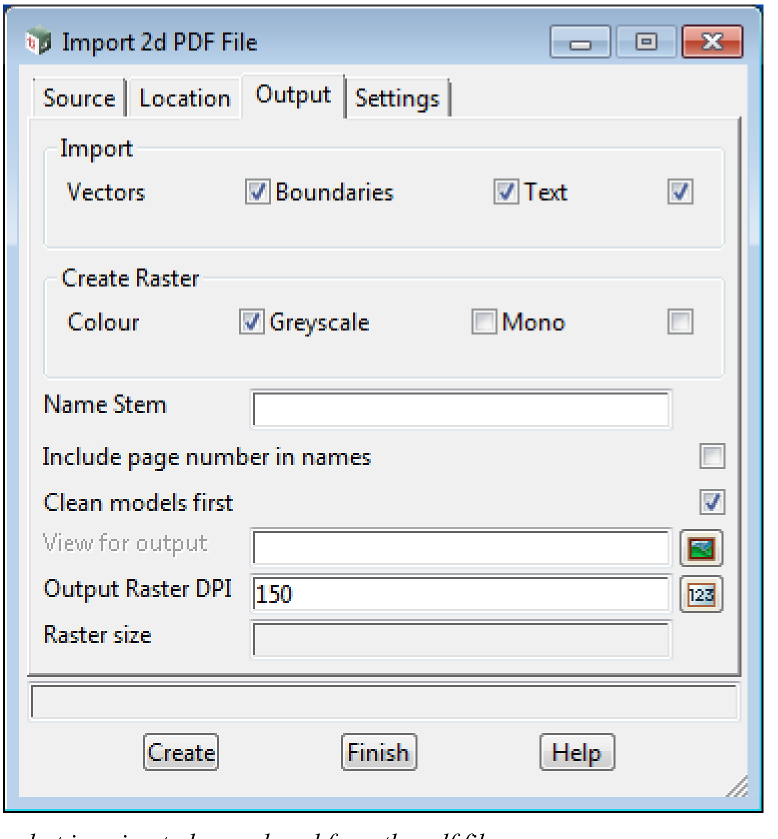
PDF file factor real box 1000

*multiplication factor to apply to the units in the each of the pages of the pdf file. Normally 1000 so millimetres in the pdf go to metres in **12d Model**.*

World width/height real box

the width/height in world units of the raster in the first page of the pdf file. The values are not used to scale the pdf. Only the **PDF file factor** *is used.*

Output tab



Controls what is going to be produced from the pdf file.

Import

*if **Raw Details** is selected as the* **Data format** *in the Location tab, the Anticlockwise rotation, (x,y) origin, World width, World height and Source file factor are used to position and scale the rasters, vectors and the text placement positions.*

Vectors tick box

if ticked, any vectors (line work) in the pdf file are read in and placed in the model given by the field **Name stem**.

Boundaries tick box

if ticked, any polygon boundaries in the pdf file are read in and placed in the model given by the field **Name stem** *followed by " **ClipBnd**"*

Text tick box

if ticked, any text in the pdf file is read in and placed in the model given by the field **Name stem** *followed by " **Text**"*

Create Raster

only one of **Colour***,* **Greyscale** *or* **Mono** *can be ticked. If none are ticked then no rasters are created from the pdf.*
Any created rasters are placed in a model given by the field **Name stem** *followed by " **raster**".*

Colour tick box

if ticked, a raster for each selected page of the pdf is created and the colours in the pdf are retained in the raster.

Greyscale tick box

if ticked, a raster for each selected page of the pdf is created and the colours in the rasters are grey scaled.

Mono tick box

if ticked, a raster for each selected page of the pdf is created and any colours are mapped to black.

Name stem text box name of the pdf file with each non alphanumeric character replaced by a space

the name stem used to generate the model names for the created models.

Include page number in names tick box ticked

if ticked, the page numbers in the pdf are also included as part of the model names.

Clean models first tick box ticked

if ticked, the models are cleaned before the pdf pages are read in.

View for output view box

if non-blank, add the models to this view

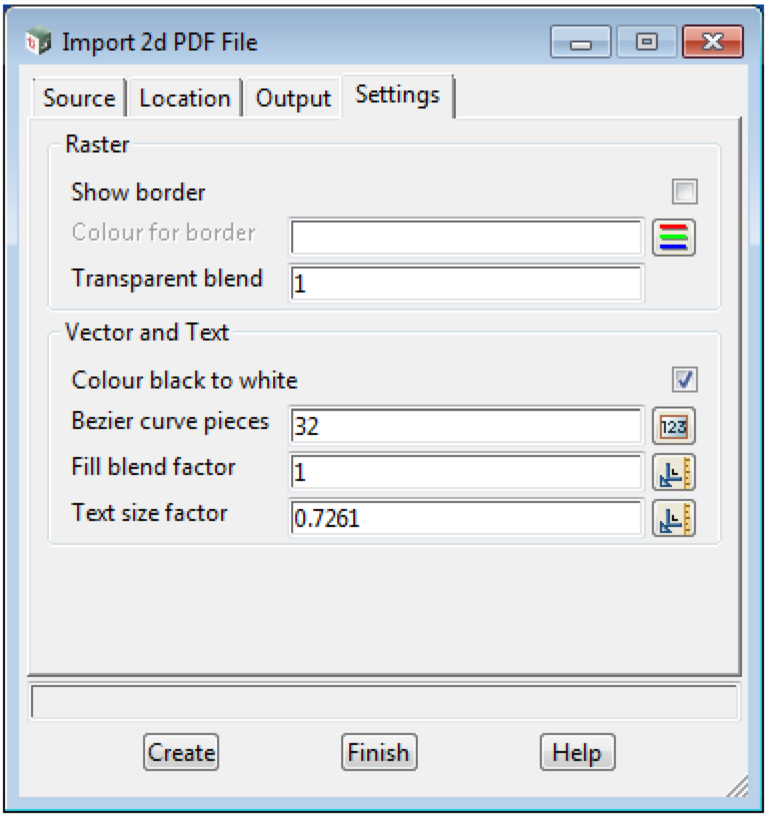
Output Raster DPI number box 150

the number of dots per inch for the created rasters.

Raster size output

the size of the produced raster.

Settings tab



Raster

Settings to use for rasters.

Show border tick box

if ticked, a border is drawn around the raster.

Colour for border colour box available colours

colour to draw the border around the raster.

Transparent blend real box 1

blend (transparency) to apply to the rasters. The value is between 0 and 1 and 1 means it is fully opaque (i.e. no transparency).

Vector and Text

Settings to use for vectors (line work) and text.

Colour black to white tick box

if ticked, black colour are changed to white. Useful when using a black background for a view.

Bezier curve pieces number box 32

if there are Bezier curves in the vectors, then they are broken into this many linear segments.

Fill blend factor real box 1

extra Blend (transparency) to apply to polygon fills. The value is between 0 and 1 and 1 means it is fully opaque (i.e. no transparency).

Text size factor real box 0.7261

the text size in pdf is different to traditional 12d text sizing (pdf size includes descenders). This factor is applied to any text to correct for the difference is size definitions.

Button at bottom

Create button

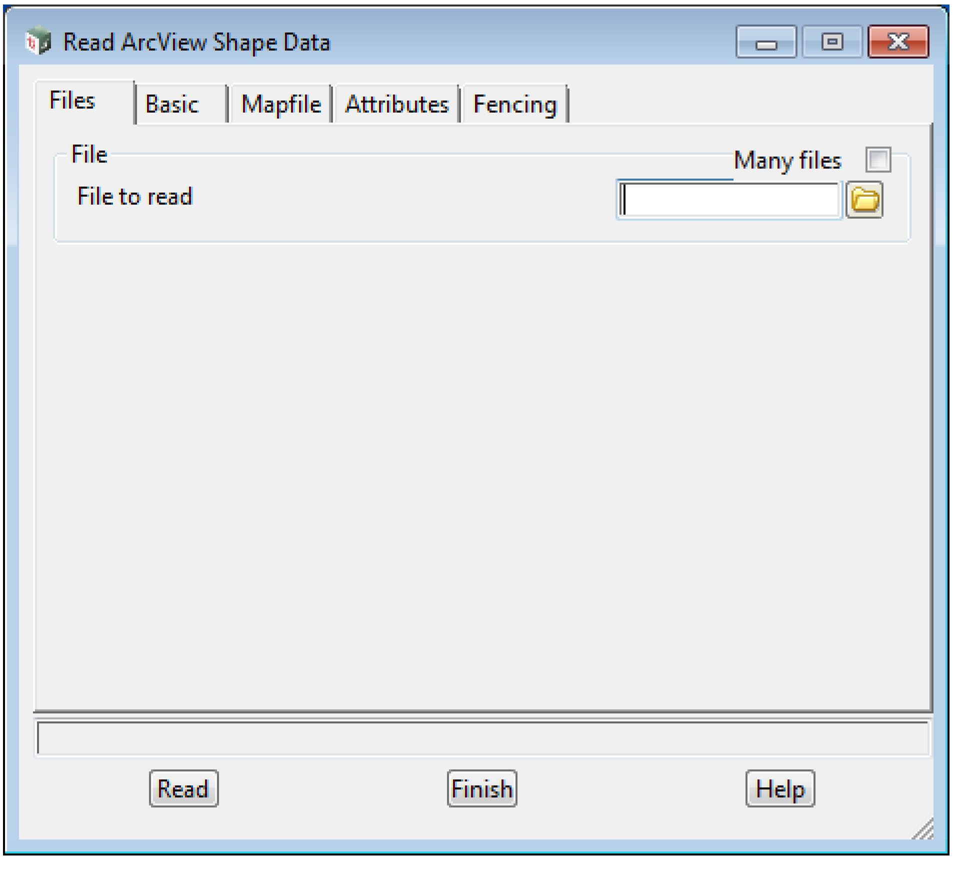
import the 2d PDF file into the appropriate model.

8.1.4 ArcView SHP Input

Position of option on menu: **File I/O =>Data input =>ArcView SHP**

Option to read in ArcView shaped files.

Selecting **ArcView SHP** brings up the **Read ArcView Shape Data** panel.



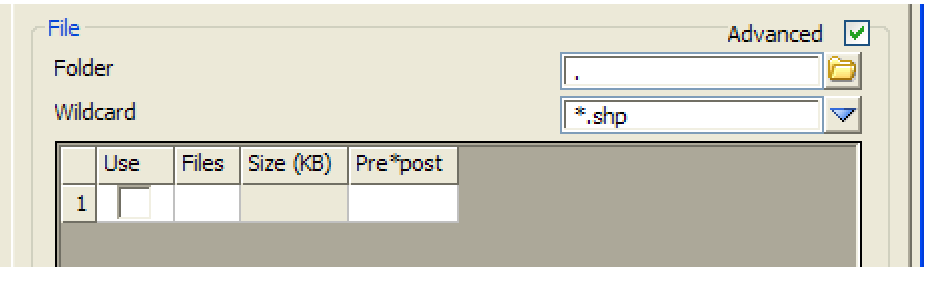
The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Files tab

Advanced tick box

*if **ticked**, a grid to allow multiple ArcView shape files to be read in, is opened. A wild card is used to select all the files to be read in.*



Folder folder box

folder to search for files using the **Wild card**

Wildcard input

wild card to use in search for files in the given folder

Use tick box

*if **ticked**, read in the file*

Files output

name of the file in the folder

Size output

file size

Pre*post text input

*if **non blank**, pre*post text to use for the models in this Arcview shp file (see* 4.19.2 Pre*Postfix in Panel Fields*) for information on using pre*postfix.*

*If **blank**, use the pre*post text from the* **Pre*postfix for models** *panel field.*

***Note** - if a non-blank value for **Pre*post is** given in the column for a file then the **Pre*postfix for models** is ignored.*

File to read file box *.shp files

name of the ArcView shape file. **12d Model** *then reads this and the two (2) other files that go with the shape file. The two other file extensions are *.shx and *.dbf.*

If no attribute is mapped to model, then the file name (minus the .shp) is used as the model for the data.

Map file tab

Map file file box *.mf and *.mapfile files

*if **non-blank**, the name of the 12d map file to be used for all strings read in.*

*If **blank**, no map file is used*

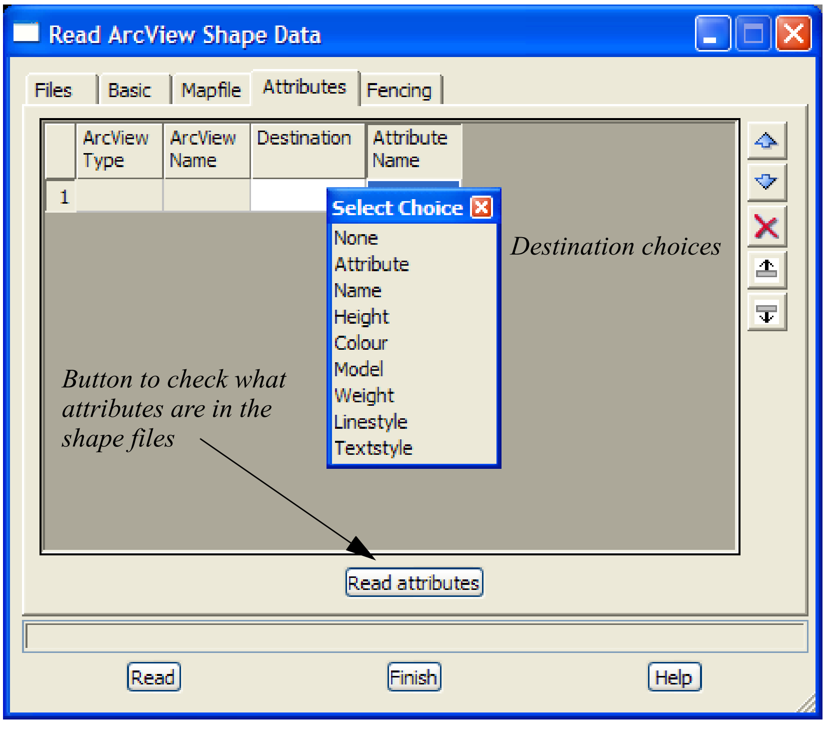
When using a map file, the string name is used as the entity-name for matching with the keys in the map file. See the section 8.8.1 Create/Edit a Map File *for information about 12d map files.*

Pre*postfix for models pre*postfix box

if non-blank, a prefix and a postfix to be applied to the model names used in the Map File.

Go to the section 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

Attributes tab



Read attributes button

Click to check what attributes are present in the ArcView shape files and any ArcView attributes found are listed in the **ArcView Type** *and* **ArcView Name** *columns in the grid.*

Destination Grid column

type of 12d attribute to map the Arcview attribute to:

none - don't use the attribute - the attribute is ignored
attribute - use as 12d attribute
name - use as 12d string name
height - use as 12d vertex height

colour - use as 12d string colour

model - use as model name

weight - use as string weight

linestyle - use as 12d linestyle

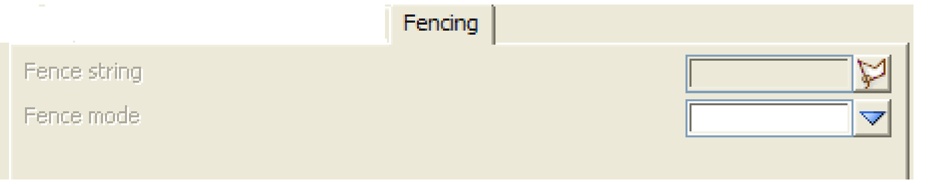
textstyle - use as 12d textstyle

***Note** - if no ArcView attribute is mapped to Model, then the file name (minus the .shp) is used as the model for the data.*

Attribute name Grid column

if non blank and the ArcView attribute is being sent to a 12d attribute, then this is the s12d attribute name

Fencing tab



Fence string polygon box

string to use to restrict the data being read in.

Fence mode choice box

String inside
String inside/crossing
String outside
String outside/crossing
String crossing

String inside - read string in if it is totally inside the polygon

String inside/crossing - read string in if it is totally inside, or crossing the polygon

String outside - read string in if it is totally outside the polygon

String outside/crossing - read string in if it is totally outside, or crossing the polygon

String crossing - string in if it is crossing the polygon

***Note** - only whole strings are read in.*

Read button

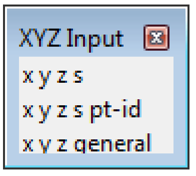
read the data in.

8.1.5 Input X Y Z Text Files

Position of menu: **File I/O =>Data input =>x y z**

The options under **x y z** read in vertices of strings from a text format. It is also possible to read in string names, point id's, string, segment and vertex attributes.

The **X Y Z** walk-right menu is



read in x y z s data
user specified x y z s point id
user specified x y z s point no and attribute

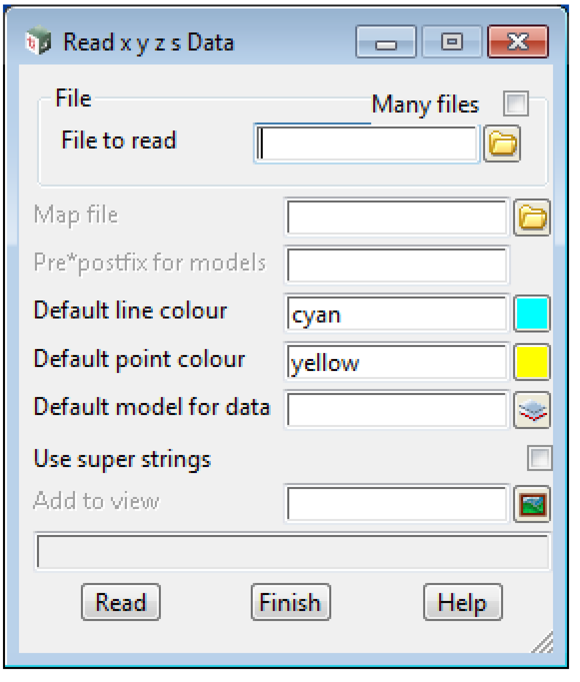
For *x y z s*, go to 8.1.5.1 X Y Z S Input
x y z pt_id 8.1.5.2 X Y Z S Point Id File Format
x y z general 8.1.5.3 X Y Z and Attributes User Format Input

8.1.5.1 X Y Z S Input

Position of option on menu: **File I/O =>Data input =>x y z s**

The x-y-z-s option reads in the xyzs format which is designed so that point and line strings can be quickly and easily coded and entered into 12d Model. It is **not** intended for more complex strings such as alignments and text where the 12d Model 12da format is more suitable. See 8.1.5.1.1 Input X Y Z S File Format)

Selecting **xyzs** brings up the **Read xyzs Data** panel.

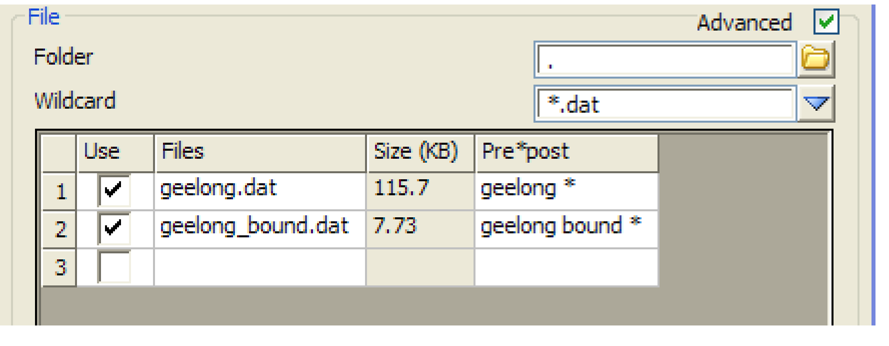


The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Many files tick box

*if **ticked**, a grid to allow multiple files to be read in, is opened. A wild card is used to select all the files to be read in.*



Folder folder box

folder to search for files using the **Wild card**

Wildcard input

wild card to use in search for files in the given folder

Use tick box

*if **ticked**, read in the file*

Files output

name of the file in the folder

Size output

file size

Pre*post text input

*if **non blank**, pre*post text to use for the models in this xyzs file (see* 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

*If **blank**, use the pre*post text from the* **Pre*postfix for models** *panel field.*

***Note** - if a non-blank value for **Pre*post is** given in the column for a file then the **Pre*postfix for models** is ignored.*

File file box *.dat files

name of the data file to be read in

Map file map file box *.mf files

*if **non-blank**, the name of the 12d map file to be used for all strings read in.*

*If **blank**, no map file is used*

When using a map file, the string name is used as the entity-name for matching with the keys in the map file. See the section 8.8.1 Create/Edit a Map File *for information about 12d map files.*

Pre*postfix for models pre*postfix box

if non-blank, a prefix and a postfix to be applied to the model names used in the Map File.

Go to the section 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix*

Default line colour colour box default colour available colours

colour used for line-strings (if no colour is defined in the file)

Default point colour colour box default pt colour available colours

colour used for the crosses in point-strings (if no colour is defined in the file)

Default model for data model box available models

name of the model that the data is to be placed in. The model will be created if it does not already exist. This field must be filled in.

Use super strings tick box

*if **ticked**, super strings will be created.*

If not ticked, 3d strings will be created.

Add to view view box available views

if non blank, the default model will be automatically added to the given view

Read button

read the data into the model given in the model field.

8.1.5.1.1 Input X Y Z S File Format

The x-y-z-s format is designed so that point and line strings can be quickly and easily coded and entered into 12d Model. It is **not** intended for more complex strings such as alignments and text where the 12d Model 12da format is more suitable.

For the x-y-z format, **point** data is set out with one point of x y z data per line. The three values are separated by one or more spaces (free format). For example

990 3 10

112 1001 23.5

A point-string is represented in 12d Model with a cross at each point. The colour of the cross can be defined by including a POINT_COLOUR command on the line before the point-string begins.

For example, a two point point-string with red crosses would be coded as

POINT_COLOUR red

100.3 990.3 10

112 1001 23.5

Line string data is also set out with one point per line (in the order that the points occur in the string) but with a string label included at the end of each line. The string label is repeated for each point in the string. The line string terminates when the string label changes to another name for a new line string or is blank for a point string.

A line-string is drawn with a line connecting a point to its neighbouring points in the string (the string links). The colour of the links of the string can be set using the LINE_COLOUR command.

For example, a green, three point line-string called S1 becomes

LINE_COLOUR green

100.3 990.3 10 S1

112 1001 23.5 S1

119.3 1203.1 29.4 S1

Notes

(a) The name of the line string can include spaces but in that case the name must be enclosed in quotes **"**. For example, the string name may be "toe 1".

(b) If a string of a certain name is created and the string name reoccurs in the file, then a new string with the same name is created. It is not joined to the earlier string.

xyz Map File

See the section 8.8.1 Create/Edit a Map File for information about ***12d Map Files***.

The **name** of the xyzs string is used as the entity-name to be used for matching with a map file. The map file can be used to override the breakline type of line-strings in the x-y-z file.

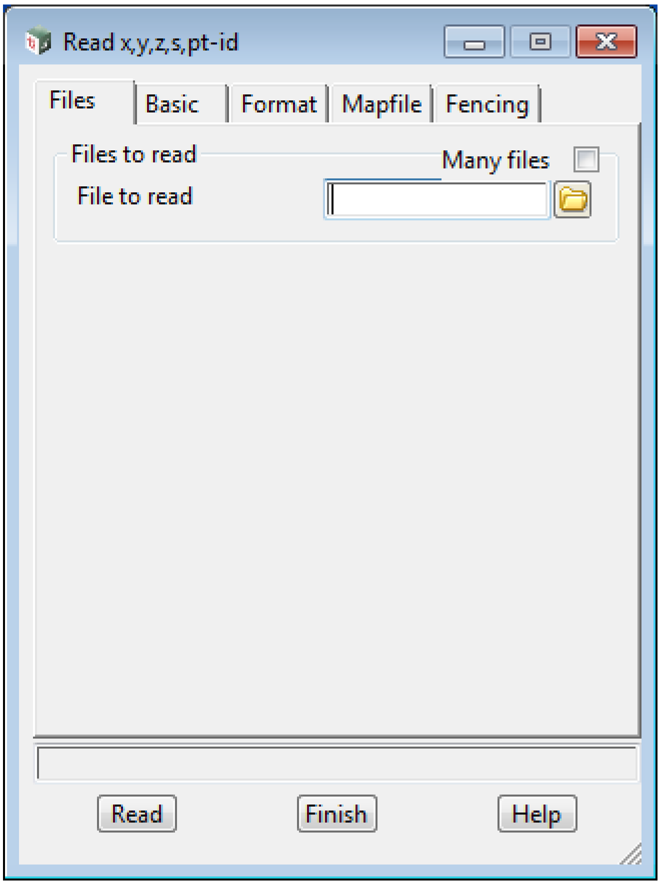
8.1.5.2 X Y Z S Point Id File Format

Position of option on menu: **File I/O =>Data input =>x y z s pt_no**

On selecting the **read x,yzs,pt_no** option, the **read xyzs pt no** panel is displayed.

This option reads data in one line at a time with the values separated by a delimiter (tab, space, semi-colon or comma) or the data on each line can be in fixed width columns.

In either case, the user specifies the order that the x, y, z, s and point number are in and if desired, only an x and y value needs to be read in.



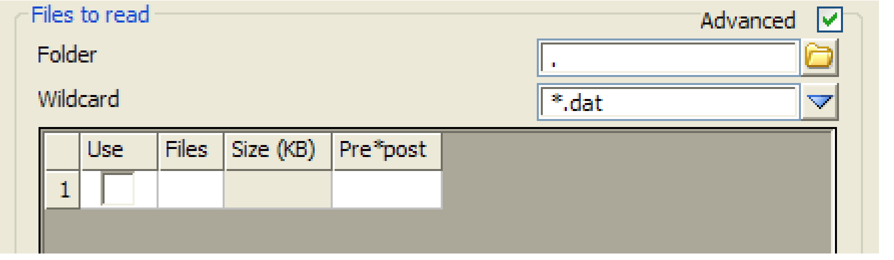
The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Files tab

Advanced tick box

*if **ticked**, a grid to allow multiple files to be read in, is opened. A wild card is used to select all the files to be read in.*



Folder folder box

folder to search for files using the **Wild card**

Wildcard input

wild card to use in search for files in the given folder

Use tick box

*if **ticked**, read in the file*

Files output

name of the file in the folder

Size output

file size

Pre*post text input

*if **non blank**, pre*post text to use for the models in this file (see* 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

*If **blank**, use the pre*post text from the* **Pre*postfix for models** *panel field.*

***Note** - if a non-blank value for **Pre*post** is given in the column for a file then the **Pre*postfix for models** is ignored.*

File to read input *.dat files

name of the data file to be read in

Basic tab

Default line colour colour box default colour available colours

colour used for line-strings (if no colour is defined in the file)

Default point colour colour box default pt colour available colours

colour used for the crosses in point-strings (if no colour is defined in the file)

Default text style input available textdatas

textdata for the point ids

Skip column headers tick box

if ticked, the first line of the file is skipped.

Join all tick box

*if **ticked**, all vertices with the same string names are joined together regardless of where they are in the file. The order of the vertices is the order they occur in the file.*

*If **not ticked** then any time a string name changes in the file, a new string is created. So if the same string name occurs but separated by a different string name, then more than one string of that same name will be created. The order of the vertices is the order they occur in the file.*

Default model for data model box available models

name of the model that the data is to be placed in. The model will be created if it does not already exist. This field must be filled in.

Add to view view box available views

if non blank, the default model will be automatically added to the given views

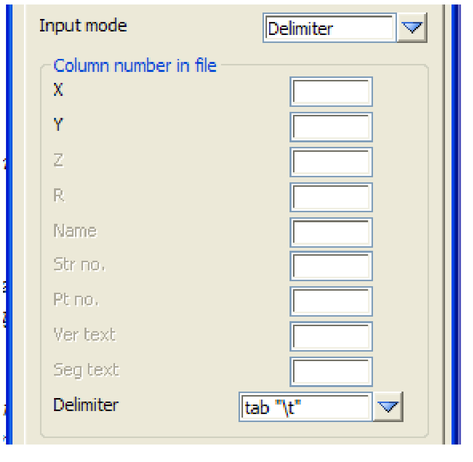
Format tab

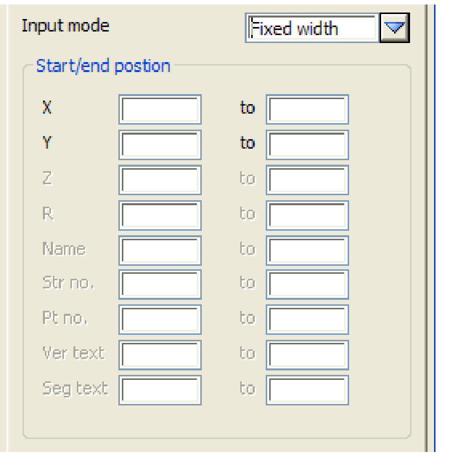
Input mode choice box delimiter delimiter, fixed width

*if **delimiter**, the type of delimiter and the columns for the x, y and optionally z, name and point number are given.*

Delimiter choice box tab \f one space, tab \t,
semi colon, comma
many spaces

*if **fixed width**, the start and end positions are given for x, y and optionally z, name and point number.*





Mapfile tab

Map file file box *.mf files

*if **non-blank**, the name of the 12d map file to be used for all strings read in, including any files given with the **Advanced** mode ticked on.*

*If **blank**, no map file is used*

When using a map file, the string name is used as the entity-name for matching with the keys in the map file. See the section 8.8.1 Create/Edit a Map File *for information about 12d map files.*

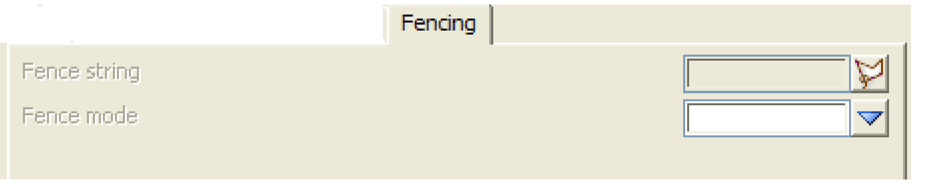
Pre*postfix for models pre*postfix box

if non-blank, a prefix and a postfix to be applied to the model names used in the Map File.

Go to the section 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

***Note** - if a non-blank value for **Pre*post** is given in the column for a file then the **Pre*postfix for models** is ignored.*

Fencing tab



Fence string polygon box

string to use to restrict the data being read in.

Fence mode choice box String inside
String inside/crossing
String outside
String outside/crossing
String crossing

String inside - read string in if it is totally inside the polygon

String inside/crossing - read string in if it is totally inside, or crossing the polygon

String outside - read string in if it is totally outside the polygon

String outside/crossing - read string in if it is totally outside, or crossing the polygon

String crossing - string in if it is crossing the polygon

***Note** - only whole strings are read in.*

Read button

read the data into the model given in the model field.

8.1.5.3 X Y Z and Attributes User Format Input

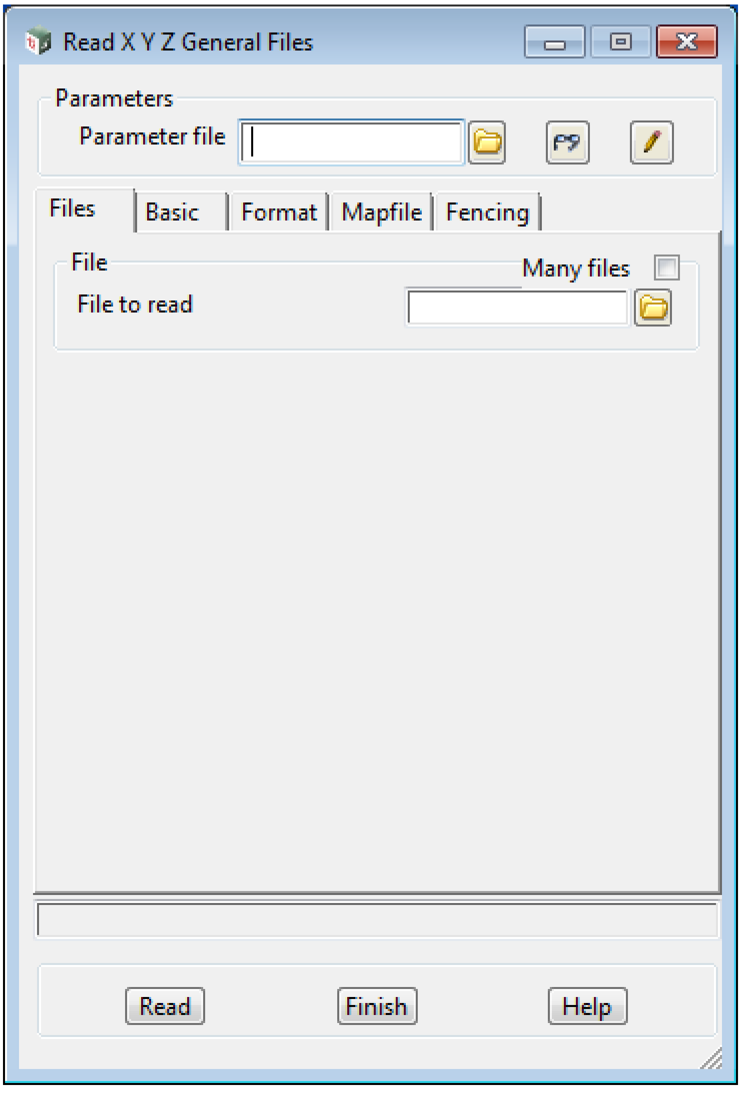
Position of option on menu: **File I/O =>Data input =>x y z general**

On selecting the **read x,y,z general** option, the **Read x y z s General Files** panel is displayed.

This option reads data in one line at a time with the values separated by a delimiter (tab, space, semi-colon or comma) or the data on each line can be in fixed width columns.

In either case, the user specifies the order that the x, y, z, string name, point number and attributes appear in the file. If desired, only an x and y value needs to be read in.

The set-ups for defining all the positions of all the data in the file can be written out to a file (.xyf) for re-use.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Parameters section

Parameter file input *.xyf files

name of the file containing the settings for how the data is positioned in the input file.

Read icon button

read the parameter file in.

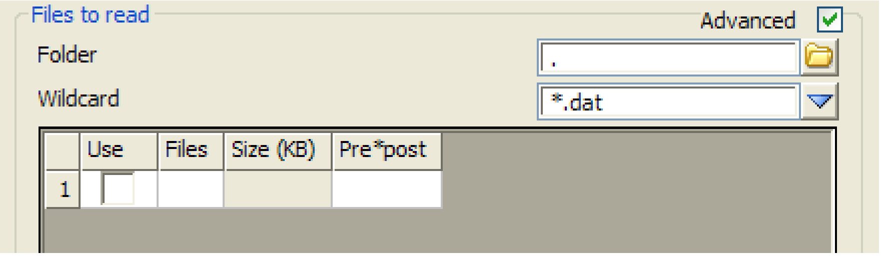
Write icon button

write the setting in the panel out to a parameter file.

Files tab

Advanced tick box

*if **ticked**, a grid to allow multiple files to be read in, is opened. A wild card is used to select all the files to be read in.*



Folder folder box

folder to search for files using the **Wild card**

Wildcard input

wild card to use in search for files in the given folder

Use tick box

*if **ticked**, read in the file*

Files output

name of the file in the folder

Size output

file size

Pre*post text input

*if **non blank**, pre*post text to use for the models in this file (see* 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

*If **blank**, use the pre*post text from the* **Pre*postfix for models** *panel field.*

***Note** - if a non-blank value for **Pre*post** is given in the column for a file then the **Pre*postfix for models** is ignored.*

File to read input *.dat files

name of the data file to be read in

Basic tab

Default line colour colour box default colour available colours

colour used for line-strings (if no colour is defined in the file)

Default point colour colour box default pt colour available colours

colour used for the crosses in point-strings (if no colour is defined in the file)

Default text style input available textdatas

textdata for the point ids

Skip column headers tick box

if ticked, the first line of the file is skipped.

Join all tick box

*if **ticked**, all vertices with the same string names are joined together regardless of where they are in the file. The order of the vertices is the order they occur in the file.*

*If **not ticked** then any time a string name changes in the file, a new string is created. So if the same string name occurs but separated by a different string name, then more than one string of that same name will be created. The order of the vertices is the order they occur in the file.*

Default model for data model box available models

name of the model that the data is to be placed in. The model will be created if it does not already exist. This field must be filled in.

Add to view view box available views

if non blank, the default model will be automatically added to the given views

Format tab

Input mode choice box delimiter delimiter, fixed width

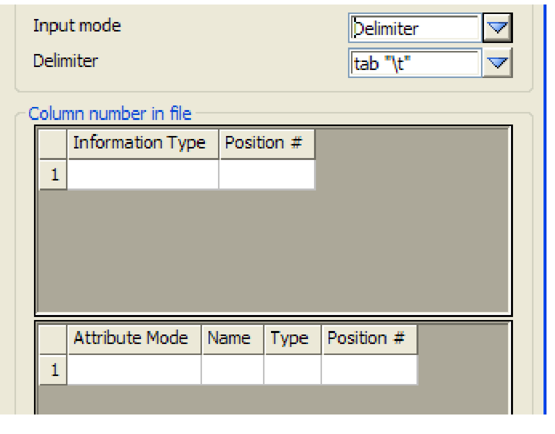
*if **delimiter**, the type of delimiter and the columns for the x, y and optionally z, name and point number are given.*

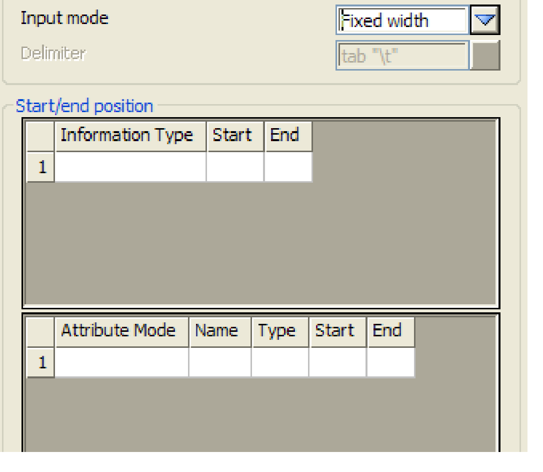
Delimiter choice box tab \f one space, tab \t,
semi colon, comma
many spaces

*if **fixed width**, the start and end column positions are given for x, y and optionally z, name and point number.*

Column number/Start end position section

information to read in (x,y,z, attributes etc.) and its position in the input file.





Mapfile tab

Map file file box *.mf files

*if **non-blank**, the name of the 12d map file to be used for all strings read in, including any files given with the **Advanced** mode ticked on.*

*If **blank**, no map file is used*

When using a map file, the string name is used as the entity-name for matching with the keys in the map file. See the section 8.8.1 Create/Edit a Map File *for information about 12d map files.*

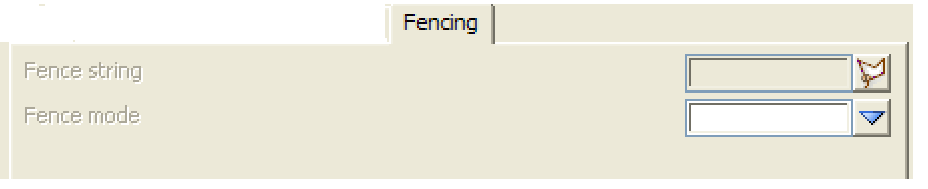
Pre*postfix for models pre*postfix box

if non-blank, a prefix and a postfix to be applied to the model names used in the Map File.

Go to the section 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

Note *- if a non-blank value for* ***Pre*post*** *is given in the column for a file then the* ***Pre*postfix for models*** *is ignored.*

Fencing tab



Fence string polygon box

string to use to restrict the data being read in.

Fence mode choice box

String inside
String inside/crossing
String outside
String outside/crossing
String crossing

String inside - read string in if it is totally inside the polygon

String inside/crossing - read string in if it is totally inside, or crossing the polygon

String outside - read string in if it is totally outside the polygon

String outside/crossing - read string in if it is totally outside, or crossing the polygon

String crossing - string in if it is crossing the polygon

Note *- only whole strings are read in.*

Read button

read in the data

8.1.6 BCC Epson Input

Position of option on menu: **File I/O =>Data input =>BCC Epson**

The **BCC Epson** input option is designed to read in files in the BCC (Brisbane City Council) Epson format.

The BCC Epson format is point based with unique point numbers for each point. It also has the concept of **non-tinable** points. Since the super string supports vertex tinability, it is the best string type to use for storing data coming in BCC Epson format. Also the BCC Epson point ids are stored as the point ids of the super string vertices.

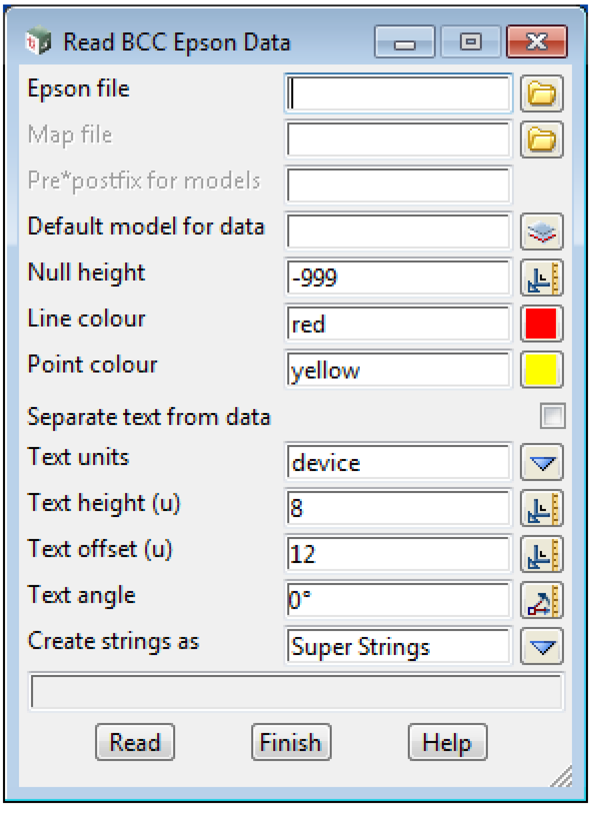
BCC Epson breaks its data up by a feature code and a 12d Model map file can be used to define models, colours, linestyles *etc*. where the BCC **feature code** as the entity-name (key) in the map file.

The best way to read in BCC Epson data is to have a map file which maps BCC Epson feature codes into strings with the same name as the feature code, and also places the strings into sensible models.

See the section 8.8.1 Create/Edit a Map File for information about 12d map files used in this option.

Note:
All text after column 68 in the BCC Epson file is considered to be a note.

On selecting the **BCC Epson** option, the **Read BCC Epson Data** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Epson file	input		.dat and .sur files

name of the BCC Epson file to be read in.

Map file input *.mf files

if non-blank, the name of the map file to be used for all strings read in.

If blank, no map file is used.
The BCC Epson feature code is used as the entity-name to match against the key in the 12d map file.

See the section 8.8.1 Create/Edit a Map File *for information about 12d map files used in this option.*

Pre*postfix for models pre*postfix box

if non-blank, a prefix and a postfix to be applied to the model names used in the Map File.

Go to the section 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

Default model for data input available models

name of the model that any unmapped data is placed in. The model will be created if it does not already exist. This field must be filled in.

Null height input -999

*if **non-blank,** any BCC Epson z values equalling this value are taken as null values.*

Line colour input red available colours

colour for BCC Epson string s.

Point colour input yellow available colours

colour for BCC Epson points that are not part of strings.

Separate text from data tick box

if not ticked, text is used in 4d or super stings
if ticked, separate text strings are created.

Text units input pixels pixels, world

units for the height of the text label.

Text height (u) input

height of the text (in text units).

Text offset (u) input

distance (in text units) to offset the text from its (x,y) placement position.

Text angle input

angle of the text.

Create strings as input Super Strings 3d, 4d, Polyline, Super

type of strings to create.

Read button

read the data in.

8.1.7 CivilCad Input

CivilCad input is a separate chargeable module.

Position of option on menu: **File I/O =>Data input =>Civilcad**

The **civilcad** input option is designed to read in CivilCad Version 4, 5 and some 6 text files.

CivilCad breaks its data up by a layer name only. By default, CivilCad layers are mapped into 12d Model models.

CivilCad has no strings but only points, lines, arcs and circles. CivilCAD spirals are ignored. When reading in CivilCad data, ***12d Model*** will try to head to tail consecutive lines from the same layer to create strings.

The CivilCad format is point based with unique point ids for each point. It also has the concept of **non-contourable** or **non-tinable** points.

Since the super string supports point tinability, it is the best string type to use for storing data coming in CivilCad format. CivilCad point ids are stored as the point numbers of the super string vertices.

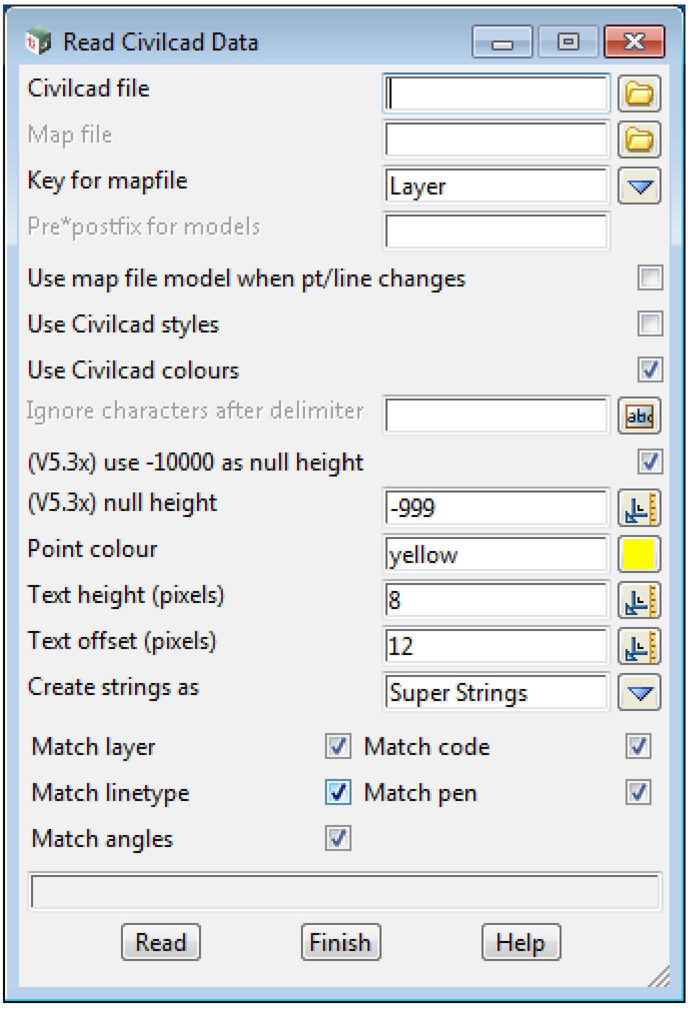
In CivilCad, a layer is defined to contain only breaklines or non-breaklines. In 12d Model, this simply corresponds to strings having a line or point *breakline* type.

If a **map** file is used when reading CivilCad data, either the CivilCad **layer** name, entity code, layer/code or code/layer can be used as the entity-name to match against the **key**.

See the section 8.8.1 Create/Edit a Map File for information about 12d map files.

Warning - CivilCad has a null value of -10,000 - these values may need to be nulled in 12d Model after the CivilCad data is read in.

On selecting **CivilCAD**, the **Read Civilcad Data** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Civilcad file	file box		*.asc and *.as5 files

name of the Civilcad text file to be read in.

Map file	map file box		*.mf files

if non-blank, the name of the map file to be used for all strings read in

If blank, no map file is used.
The **Key for map file** *defines what is used to match against the key in the map file.*

See the section 8.8.1 Create/Edit a Map File *for information about 12d map files.*

Key for map file	choice box	Layer	Code, Layer, Code/Layer, Layer/Code

if Layer, the Civilcad layer is used as the entity-name to match against the key in the map file.
if Code, the Civilcad layer is code as the entity-name to match against the key in the map file.

Pre*postfix for models	pre*postfix box		

if non-blank, a prefix and a postfix to be applied to the model names used in the Map File.

Go to the section 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

Use CivilCad styles	tick box		

*if **ticked,** then the CivilCad linestyle number is used as the 12d Model linestyle name.*

Use CivilCad colours tick box

*if **ticked,** then the CivilCad colour number is used as the 12d Model colour number.*

Ignore characters after delimiter tick box

*if **ticked,** r.*

(V5.3x) use -10000 as null height tick box tick

*if **ticked,** any CivilCad z values of -10,000 are taken as null values.*

(V5.3x) null height input -999

the value in the CivilCAD file to use as null height if User -10000 is not ticked.

Point colours colour box yellow available colours

colour for CivilCad points that are not part of strings.

Text height (pixels) input 8

the height in pixels of any text created, or for point numbers.

Text offset (pix) input 12

the offset (in pixels) from the (x,y) coordinate position for any text or point numbers.

Create strings as input Super Strings Lines and Arcs
3d Strings and Arcs,
Polyline Strings
Super Strings

type of strings to create.

Match flags tick box

CivilCad data consists of individual lines and arcs. When reading CivilCad data, it undergoes head to tail processing and the match flags specify what CivilCad data can be joined.

Match layer/code/linetype/angles/pen tick box

*if **ticked,** then any CivilCad lines and arcs must have the same layer/code/linetype/angle/pen before they can be joined in the head to tail process.*

Read button

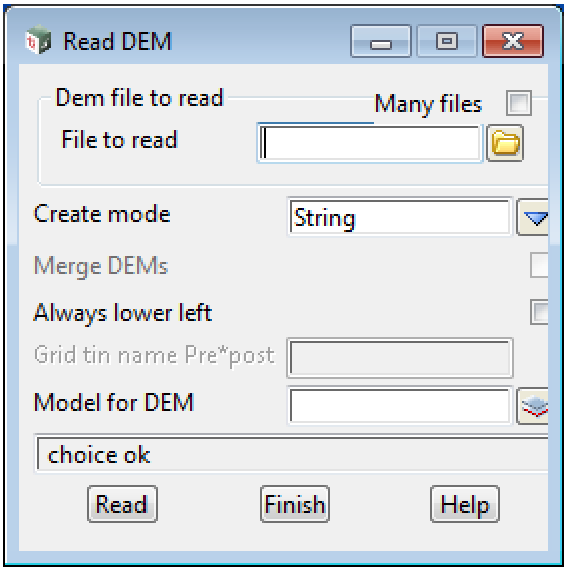
read the data in.

8.1.8 DEM Input

Position of option on menu: **File I/O =>Data input =>DEM**

The **DEM** option is designed to read in Digital Elevation Model data in the Arc/Info format.

On selecting **DEM**, the **Read DEM** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Format	choice box	Arc/Info ASCII	Arc/Info ASCII Grid *.dem

format of the DEM to read in.

Dem file to read	file box		

name of the DEM file to be read in.

File to read

Create mode	choice box	String	String, Grid string, Grid tin

see 14.10 Grids *for an explanation of* **Grid string** *and* **Grid tin**

Single grid	tick box	not ticked	

if ticked, all points from all selected files are used in the creation of a single grid.

Grid tin name Pre*post

Model for DEM	model box		available models

model to put the DEM data in.

Read	button		

read the data in.

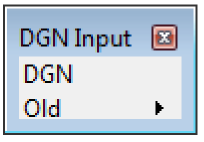
8.1.9 Input DGN Binary Files

Position of menu: **File I/O =>Data input =>DGN**

DGN input and output is a separate chargeable module.

The options under **DGN** read DGN V7 and DGN V8 binary files.

The **DGN** walk-right menu is



read DGN V8 binary file
read DGN V7 binary file

For *DGN V8*, go to 8.1.9.1 Input DGN V8

DGN V7 8.1.9.2 DGN V7 Binary Input

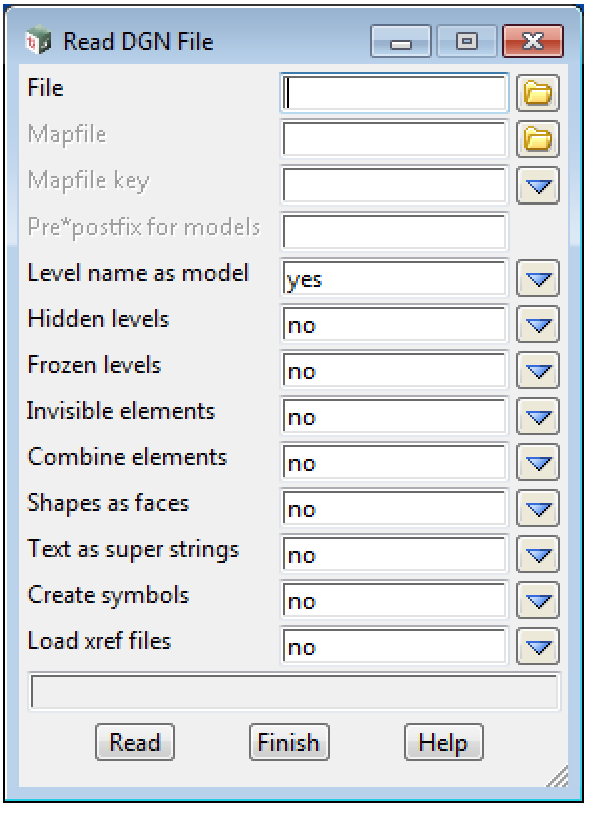
8.1.9.1 Input DGN V8

Position of option on menu: **File I/O =>Data input =>DGN =>DGN V8**

NOTE: the format for DGN V8 has been changed and has not been published. There is a beta version of the DGN reader to try and read DGNV8. At this stage it is probably better to use DWG I/O to go in and out of Microstation V8.

Note: this option is under development, and its documentation is a work in progress.

Selecting **DGN V8** brings up the **Read DGN File** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
File	input		*.dgn files

name of the DGN V8 binary file to be read in

Read	button		

read the data in.

8.1.9.2 DGN V7 Binary Input

DGN binary input is a separate chargeable module.

Position of option on menu: **File I/O =>Data input =>DGN =>DGN V7**

The **DGN** input option is designed to read in Intergraph and Microstation binary models (**.dgn** files) up to V7. Microstation V8 has a new undocumented format which is read in with the DGN V8 options (.

Because of the limited number of levels available in an DGN file, the colour, linestyle and weight of items are often used to differentiate data types. For special files for VicRoads, there is also an DGN attribute which can be used to tag data.

Hence the entity-name used for matching in a map file when reading DGN files into 12d Model can be either the:

- DGN **level**
- VicRoads **attribute**
- DGN level, colour, linestyle, weight
- VicRoads attribute, colour, linestyle, weight

In the DGN map file, the four level key is given as a single key made up of the four items separated by **|** (with no additional spaces) in the order:

level or VicRoads attribute | colour | line style | weight

For example, the key

30|2|0|0 means level 30, colour 2, style 0 and weight 0

A ***** can be used for any of the four items to indicate that no match is required for that item.

30|*|0|* means level 30, any colour, style 0, any weight

If a map file is not used, the DGN data is read into a ***12d Model*** string of the same **name** as the DGN level and the DGN colour numbers are mapped to ***12d Model*** colours. All the strings created go to the **Default model for data** specified in the *Read DGN Data* panel.

See the section 8.8.1 Create/Edit a Map File for information about 12d map files.

Text for Points in DGN

A further complication with DGN is that many Microstation users record individual points as a text entity with a character from a possibly user defined DGN font to represent a symbol at the point. The text then often has a justification other than left-bottom.

However, Microstation does not record the (x,y) coordinates of the justification point but only the left-bottom position to draw the symbol at. The actual (x,y) point needs to be calculated from the text justification and the actual symbol size.

Unfortunately this requires a knowledge of the DGN font symbol being used to calculate the true (x,y) position for the text justification point.

To help read in text as points so that they may be included in processes such as triangulation, in the DGN reader, it is possible to create 4d strings (which consist of an (x,y,z) point plus a piece of text) instead of a text string.

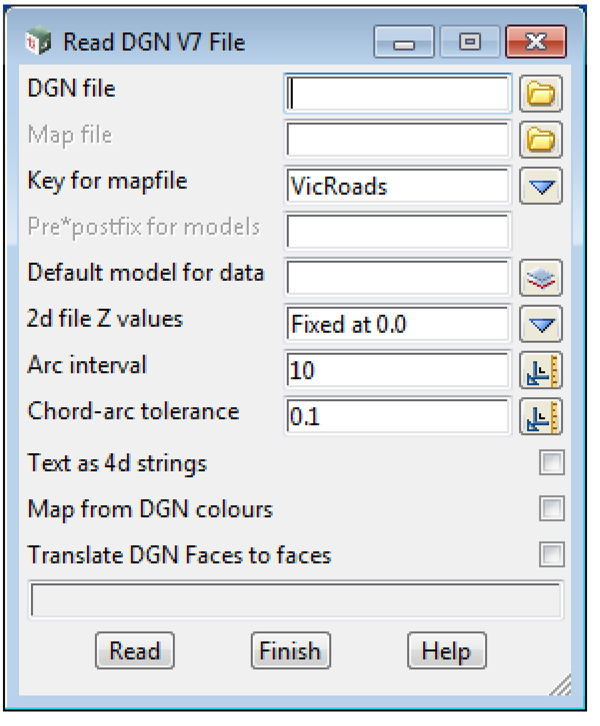
Warning

Because of the problem with needing to know the DGN font information to correctly calculate the (x,y) coordinate of the text, creating a 4d string instead of text will not fix the error in the (x,y) coordinates if an incorrect font is used in 12d Model.

The only safe solution is to avoid data in DGN format if the points are being represented by text.

NOTE: the format for DGN V8 has been changed and has not been published. There is a beta version of the DGN reader to try and read DGNV8. At this stage it is probably better to use DWG I/O to go in and out of Microstation V8.

On selecting the **DGNV7** option, the **Read DGN V7 File** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

DGN file input *.dgn files

name of the DGN binary file to be read in

Map file input *.mf files

if non-blank, the name of the map file to be used for all strings read in.

*If blank, no map file is used.The DGN level is used as the string name and all strings go to the **Default model for data**.*

See the section 8.8.1 Create/Edit a Map File *for information about 12d map files.*

Key for mapfile input VicRoads level, VicRoads, level/colour/linestyle/weight Vicroads/colour/linestyle/weight

specifies how the key from the mapfile is interpreted.

Pre*postfix for models pre*postfix box

if non-blank, a prefix and a postfix to be applied to the model names used in the Map File.

Go to the section 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

Default model for data input available models *model to use for strings not mentioned in the map file or if no map file is used.*

2d file Z values	input	Fixed at 0.0	fixed at 0.0, contour Z low, contour Z high element Z low, element Z high

z value to use for 2d .dgn file

Arc interval	input	10

interval to use to break 3d circles into segments.

***Note** - a 3d circle in DGN is a circle in an included plane. This is not the same as a circle in civil work which is a circle in **plan view**. A 3d circle in an included plane does not project onto a circle in a plan view except in the special case when the inclined plane in parallel to the x-y plane.*

Chord-arc tolerance	input	0.1

chord to arc tolerance to use when breaking 3d circles into segments.

Text as 4d strings tick box

*if **ticked**, text is read in as a 4d string (an (x,y,z) coordinate plus a piece of text)*

Map from DGN Colours tick box

*if **not ticked**, DGN colour number n is mapped to* **12d Model** *colour n.*
*if **ticked**, some of the DGN colours are attempted to be mapped to* **12d Model** *colours.*

Translate DGN faces to faces tick box

*if **ticked**, DGN faces are read in as* **12d Model** *face strings.*

Read button

read the data in.

8.1.10 DWG/DXF Input

DWG/DXF input is a separate chargeable option

Position of option on menu: **File I/O =>Data input =>DWG/DXF/DX8**

The **DWG/DXF** input option is designed to read most Autocad DWG and DXF files.

Each DWG/DXF item has an associated layer. By default, 12d Model creates models of the same name as the layers (or with an additional user supplied prefix) and the DWG/DXF items placed in them.

However, this can be over written using a standard **12d map file** where the key is matched against Autocad **layers** rather than string names.

See the section 8.8.1 Create/Edit a Map File for information about 12d map files.

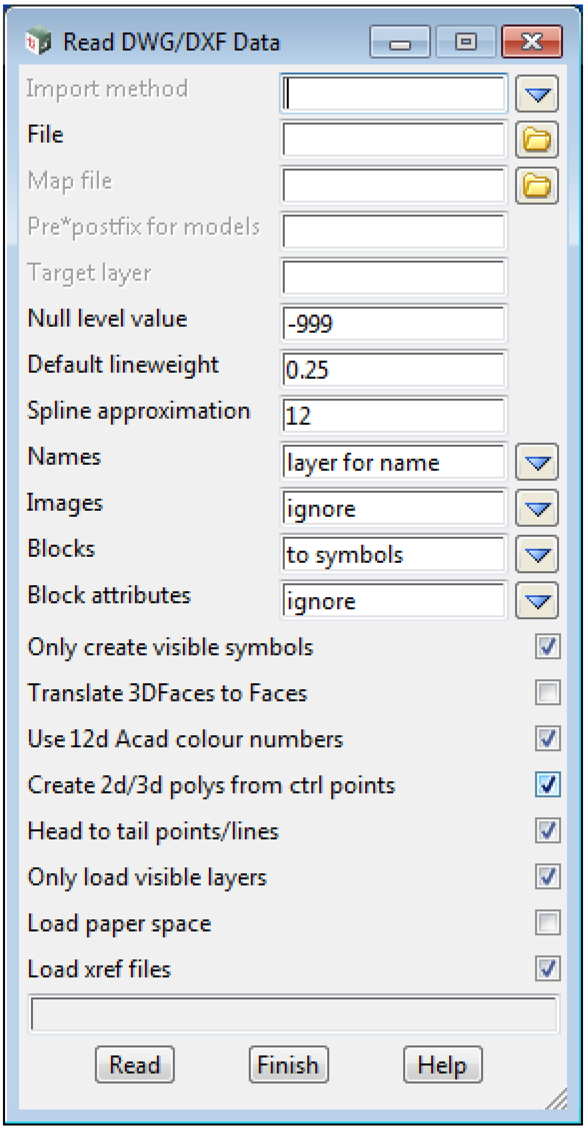
Autocad blocks are recognised and either a point with the block name is created or the blocks are expanded into their components, each time they are referenced in the DWG/DXF file.

Bulges in polylines can only be interpreted correctly when the polyline has a constant z-value. For this case, a 12d Model polyline string is created from the DXF polyline.

A Polyface Mesh in the DWG/DXF file is read in as a Trimesh.

Finally, DWG/DXF POINT entities of the same layer and colour can be concatenated into one point string as they are read in.

Selecting **DWG/DXF/DX8** displays the **Read DWG/DXF Data** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Format	choice box	DWG	DWG, DXF

type of Autocad file to read in

File	file box		*.dwg or *.dxf files

name of the DWG or DXF file to read in

Map file	map file box		*.mf, *.mapfile files

if non-blank, the name of the map file to be used for all DWG/DXF layers read in. The DWG/DXF layer is the entity-name for matching against the key in the map file.
If blank, no map file is used.

See the section 8.8.1 Create/Edit a Map File *for information about 12d map files.*

Pre*postfix for models	pre*postfix box		

if non-blank, a prefix and a postfix to be applied to the model names used in the Map File.

Go to the section 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

Target layer	input		

if non-blank, only autocad items in the layer with the name given in the **target layer** *field will be read in.*

Null level value input -999

z-value to treat as a null level

Default AutoCAD lineweight input 0.25

lineweight to use when it is undefined in an AutoCAD entity.

Spline approximation input 12

splines are broken into small segments

Names choice box layer for name no name, layer for name

*if **no name**, strings are not given a name.*

*If **layer for name**, strings are given the name of the AutoCAD layer they were on.*

Images choice box to plan images to plan images, to rasters, ignore

Blocks choice box to symbols to symbols, explode, to points

*if **to symbols**, blocks are read in as 12d Symbols of the same name.*

*If **explode**, blocks are read in and exploded into vertices and line work.*

*If **to points**, a 12d vertex is created at each block.*

Only create visible symbols tick box tick

Translate 3DFaces to faces tick box

*if **tick**, DWG/DXF faces are read in as* ***12d Model*** *face strings.*

User 12d ACAD colour numbers tick box

Create 2d/3d polys from ctrl points tick box

Head to tail points/lines tick tick

*if **ticked**,* DWG/DXF POINT entities of the same layer and colour are concatenated into one point string as they are read in and DWG/DXF LINE entities of the same layer and colour are

Only load visible layers tick tick

*if **ticked**, only* DWG/DXF visible layers are read in otherwise all layers are read in.

Load paper space tick no tick

*if **ticked**, paper space data will be read in.*

Load xref files tick no tick

*if **ticked**, an xref files in the DWG/DXF are also read in.*

Read button

read the data in.

8.1.11 FBX Input

Position of option on menu: **File I/O =>Data input =>FBX**

Position of option on menu: **Strings =>Trimesh =>Trimeshes from FBX file**

This option is documented in 14.3.4 Trimeshes from FBX File.

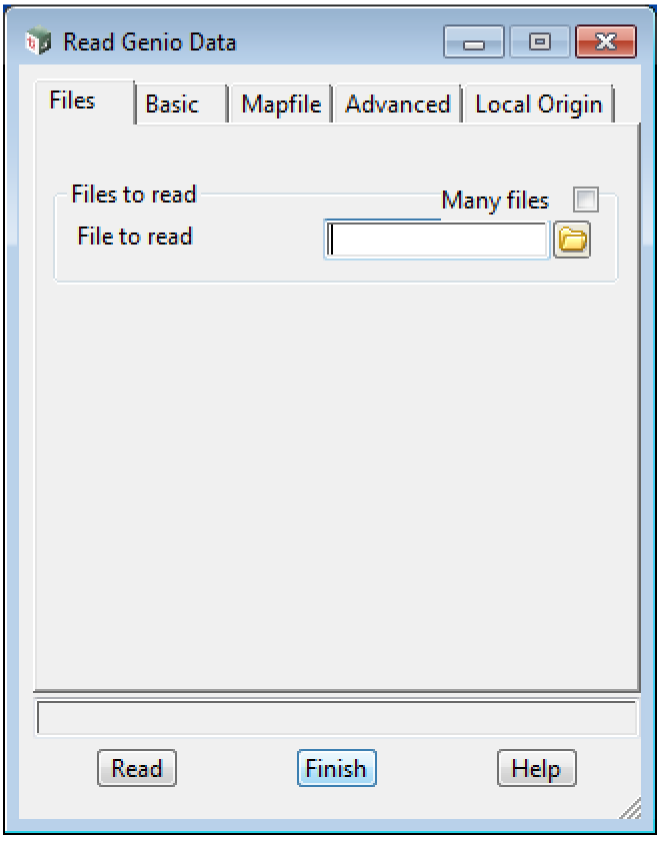
8.1.12 Genio Input

Position of option on menu: **File I/O =>Data input =>Genio**

The software package MX (formerly called Moss) includes a data file format called GENIO for use in transferring data between Moss and other programs (see the MX Manual for a partial description of genio). There is currently two default genio input formats - versions 6 and 7. 12d Model recognises both formats.

For information on the Genio file format support by 12d Model, please go to the section 8.1.12.1 Genio File Format

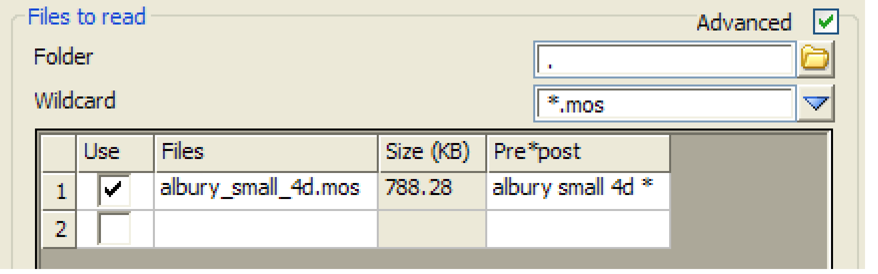
On selecting the **Genio** option, the **Read Genio Data** panel is displayed.



The fields and buttons have the following functions.

Field Description	Type	Defaults	Pop-Up
Files tab			
Advanced	tick box		

*if **ticked**, a grid to allow multiple files to be read in, is opened. A wild card is used to select all the files to be read in.*



Folder folder box

folder to search for files using the **Wild card**

Wildcard input

wild card to use in search for files in the given folder

Use tick box

*if **ticked**, read in the file*

Files output

name of the file in the folder

Size output

file size

Pre*post text input

*if **non blank**, pre*post text to use for the models in this file (see* 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

*If **blank**, use the pre*post text from the* **Pre*postfix for models** *panel field.*

***Note** - if a non-blank value for **Pre*post** is given in the column for a file then the **Pre*postfix for models** is ignored.*

File to read file box default *.mos files
This can be modified by the environment variable GENIO_WILDCARD_4D

name of the genio file to be read in

Basic tab

Transition type transition box clothoid transition pop-up

type of transition used in the genio file

Line colour colour box default colour available colours

colour used for line-strings

Point colour colour box default pt colour available colours

*colour used for the crosses in point-strings (genio string labels starting with **P**).*

Text style data for 4d strings

the text style data to use for the text in any 4d strings in the genio file

Text style data for text

the text style data to use for text strings in the genio file

Multiply text height by 10 **tick box**

if ticked, the text size is multiplied by 10

Mapfile tab

Map file file box *.mf files

*if non-blank, the name of the 12d map file to be used for all strings read in, including any files given with the **Advanced** mode ticked on.*

If blank, no map file is used

When using a map file, the string name is used as the entity-name for matching with the keys in the map file. See the section 8.8.1 Create/Edit a Map File *for information about 12d map files.*

Pre*postfix for models pre*postfix box

if non-blank, a prefix and a postfix to be applied to the model names used in the Map File.

Go to the section 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

***Note** - if a non-blank value for **Pre*post** is given in the column for a file then the **Pre*postfix for models** is ignored.*

Advanced tab

Compress 3d to 2d tick box tick

*if **ticked**, convert any 3d strings with constant z-value to 2d strings.*

Convert 6d to alignment tick box tick

*if **ticked**, convert any 6d strings to 12d Model alignment strings.*

Generate Point ID's tick box no tick

*if **ticked**,.*

Create control stations from SSTN tick box no tick

*if **ticked**, control stations are created in **12d Model** for each point in a genio string named SSTN*

Smigs file ? tick box no tick

*if **ticked**, try to interpret the genio file as a Smigs genio file.*

Use super strings tick box tick

*if **ticked**, all strings are read in as super string*

Use invisible segments for discontinuities tick box

*if **ticked**, MX strings with discontinuities are read in as super strings with invisible segments.*

*If **not ticked**, MX strings with discontinuities are broken into pieces when read in.*

G strings to super alignments tick box no tick

*if **ticked**, MX G strings are read in as 12d super alignments*

Debug G strings tick box no tick

because of the lack of documentation of the G string in MX, a G string may not be interpreted correctly.

*If **ticked**, a super string is created with the information from the MX G string to try and help interpret what the data was.*

Reverse calc super alignments tick box no tick

Better way of interpreting VG tick box tick

*if **ticked**, a different method for interpreting vertical geometry in a MX G strings is used.*

This may or may not be successful.

Local Origin tab

Local origin x y box x y selector

*if **non-blank**, the given coordinates are used as a local origin when reading in the data. That is, the* **local origin** *values are subtracted from each data point as it is read in.*

Read button

read in the genio data from the file given in the file field.

8.1.12.1 Genio File Format

The software package MX (formerly called Moss) includes a data file format called GENIO for use in transferring data between Moss and other programs (see the MX Manual for a partial description of genio). There is currently two default genio input formats - versions 6 and 7. 12d Model recognizes both formats.

MX provides three options (001,003,017) to allow variations in the format of the genio 080 records. 12d Model recognizes and uses each option.

MX free format (that is, using commas and the 'field-number=' syntax) is allowed.

For example

080,ABCD,   ,5=0.0

MX 2d, 3d, 4d, 6d, text and most 12d and text strings are loaded directly into 12d Model strings. For 5d, the strings are processed but only the 3d information is used and loaded into 12d Model. The MX 10d (volume string) is allowed in the genio file but will not be read into 12d Model.

In the genio file, the GENIO card defines the name of the MX model for the following strings. 12d Model loads the strings into a 12d Model model with the same name as the MX model. However, 12d Model allows the user to define a **map file** which may over-ride the Genio card.

A genio file may contain more than one MX model, each genio model being separated by a 999 card. 12d Model will load each separate MX model into a 12d Model model with the same name as the MX model.

If an error occurs whilst reading a genio record, the genio record will be skipped and, if possible, the next genio record read.

12d Map File

See the section 8.8.1 Create/Edit a Map File for information about 12d map files.

The **name** of the MX string is used as the entity-name to be used for matching in a 12d map file.

The MX model given in the GENIO record is taken to be the default model for the genio reader.

Hence the genio map file can be used to over-ride the MX GENIO card and the MX default for point strings (this is necessary for genio files generated from non MX systems that don't use the correct point string convention).

For **text** in a Genio file (the string name for text must start with a * in the Genio file), the *key in the map file must* start with ***geniotext*** followed by the characters to match on which will be matched against the characters following the * in the genio string name. For example

    geniotextEB*   *   text   yellow   point   "PHCP"   // matches text

will match any Genio text with the string name starting with "*EB"

Note that * is not a wild card in the Genio string name but the "EB*" after "geniotext" is for use by the 12d map file and hence is an "EB" followed by a wild card. Hence it matches against any Genio text name starting with "*EB".

Summary

12d Model **recognizes** the following genio options for MX V6 and V7 formats -

GENIO
FINISH

001	format card
003	order card
017	angle card
080	for 2d, 3d, 4d, 6d, 12d and text strings
080	the 3d information only for 5d strings
090	triangulation information
999	

genio comment lines

12d Model accepts MX null values of -999.0

12d Model **ignores** the genio options:

080 for **10d** strings

A12d map file can be used to select the colour and model for any strings read in from the MX genio file whose names match the keys in the map file.

8.1.13 Geocomp Input

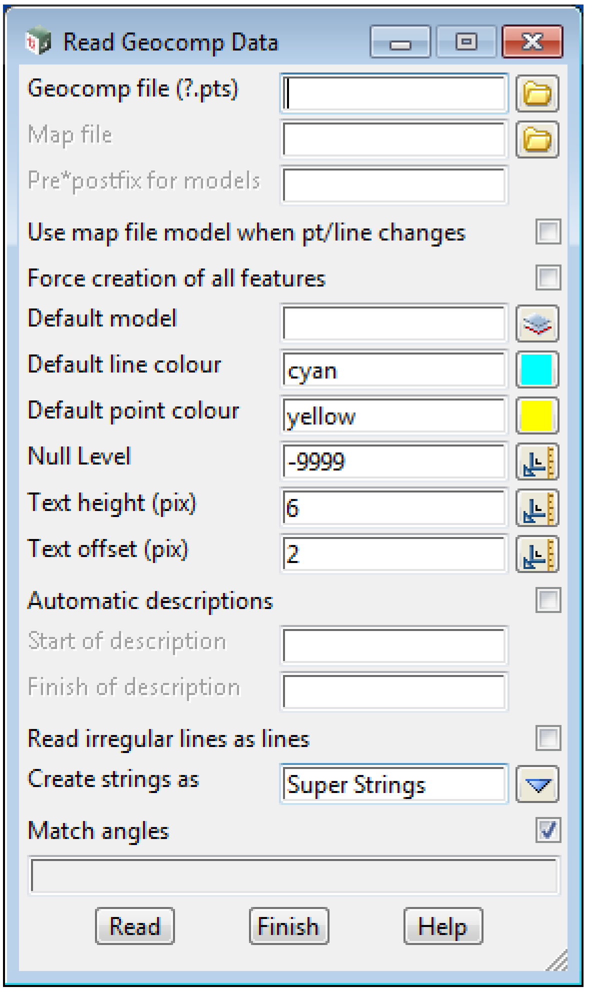
Position of option on menu: **File I/O =>Data input =>Geocomp**

Geocomp input is a separate chargeable module.

Geocomp is a points based software package used for manipulating and reducing survey data on a PC.

Geocomp data is held in two text files - the points file and the strings file. These text files can be used to transfer data from Geocomp to 12d Model. More information on the Geocomp data structure is given in the previous section 8.1.13.1 Geocomp File Format.

Selecting **Geocomp** displays the **Read Geocomp Data** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Geocomp file (?.pts)	file box		*.pts files

name of the Geocomp points file to be read in. The corresponding strings file is also used.

Map file	map file box		*.mf and *.mapfile files

if non-blank, the name of the map file to be used for all strings read in.

If blank, no map file is used.

See the section 8.8.1 Create/Edit a Map File *for information about 12d map files.*

Pre*postfix for models	pre*postfix box		

if non-blank, a prefix and a postfix to be applied to the model names used in the Map File.

Go to the section 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

Use mapfile model when pt/line changes	tick box	no tick	

if ticked, then if the geocomp.str file says the entity is a line (point) feature and the map file says it is a point (line), the mapfile is used.

Default model	model box		available models

model to use for strings not mentioned in the map file

Default line colour	colour box	default colour	available colours

colour used for line-strings

Default point colour	colour box	default pt colour	available colours

colour used for the crosses in point-strings.

Null level	input	-9999

the Geocomp z-values to be considered null z-values.

Text height (pix)	input	8

the height in pixels of any text created for point number or Geocomp descriptions.

Text offset (pix)	input	2

the offset (in pixels) from the (x,y) coordinate position for the text of a 4d string.

Automatic descriptions	tick box

if ticked, then the first three characters of the entity code are compared to the first 3 characters of the 20 character description and if they are the same, the text is only characters 6 to 15 of the description.

Start of description	input

the user can restrict the amount of the geocomp description that is read in. if non-blank, this is the number of the character position to start reading the description from. If blank, the start position is 1.

Finish of description	input

if non-blank, this is the number of the character position to end reading the description from. If blank, it is the end of the geocomp description.

Read irregular lines as lines	tick

if ticked, Geocomp irregular lines are read in as strings.

Create strings as	input	Super Strings	Polyline Strings Super Strings

type of strings to create.

Match angles	tick	tick

if ticked, preference is given to joining geocomp lines of similar angles first.
if not ticked, geocomp lines are joined in the order they are in the file.

Read	button

read in the Geocomp data from the points file and (corresponding strings file) given in the file field.

8.1.13.1 Geocomp File Format

Geocomp input is a separate chargeable module.

Geocomp is a software package used for manipulating and reducing survey data on a PC.

Geocomp is a point based system. The fundamental data consists of individual points with unique point numbers and entities such as lines and arcs defined in terms of the points.

Geocomp data is held in two text files - the points file and the strings file. These text files can be used to transfer data from Geocomp to 12d Model.

The Geocomp **points file** is a sequential list of points, one point per line. Each point consists of the data

easting, northing, elevation, stand-point-number

that is

x-value, y-value, z-value, stand-point-number

The line number of each point in the file is also the unique **point number** for that point. For example, the point defined on the eleventh line of the points file, **is** point number eleven.

The stand-point-number is the point number of the instrument station used for collecting the points coordinates. The stand-point-number is not used in 12d Model.

The Geocomp **strings file** defines how the Geocomp **entity types** are constructed from points in the point file. The Geocomp entity types are

Entity Number	Description
1	two point lines
2	irregular lines
3	arc defined by start point, point on arc, end point
4	arc defined by start point, arc centre, end point
5	point feature
6	circle defined by centre and a point on the circle

The strings file also gives each defined entity an **entity number**.

The 12d Model Geocomp data reader recognises all the entity types in the strings file **except** for irregular lines (type 2).

All point numbers referred to in the strings file are references to the (implied) point numbers of points in the points file. Hence both files are needed to define the Geocomp data and the order of points in the point file is critical and cannot be modified.

The Geocomp naming convention for the two files is to use the six digit Geocomp job number as a name stem and append **.pts** for the points file and **.str** for the strings file.

That is,

geocomp-job-number.pts	points file
geocomp-job-number.str	strings file

For example

099999.pts	is the point file for Geocomp job number 099999
099999.str	is the strings file for Geocomp job number 099999

12d Model uses a wider data set than is directly represented in the Geocomp points and strings files. However, by observing a number of conventions and processing the strings file data according to these conventions, the Geocomp data can be sensibly passed across to 12d Model.

For example, although the Geocomp strings file only defines two-point lines, 12d Model can construct strings of many points from consecutive two-point lines by joining the lines together whenever

(a) the second point of one line is the same as the first point of the next line in the file

and

(b) the lines have the same entity number.

The entity number of the consecutive lines is used as the 12d Model string name.

Similarly, consecutive point features with the same entity number are joined to form a 12d Model 4d **point** string with the entity number as the string name.

In the Geocomp strings file, **descriptive text** can also be included at the end of each line defining an entity. The text is enclosed within double quotes **"**. The descriptive text is ignored for all entities **except point features**.

For a point-feature, which is represented in 12d Model as a 4d string, the descriptive text is recorded as the text label for that point.

By using a systematic entity and descriptive text labelling system in Geocomp, it is possible to sensibly transfer all of the Geocomp data to 12d Model.

Note - in Geocomp, a z-value of -9999 represents a null z-value, that is, a z-value that has **not** been defined. Any Geocomp null values are recorded as 12d Model null values.

Geocomp Map File

See the section 8.8.1 Create/Edit a Map File for information about 12d map files.

Using the conventions described in the previous section, Geocomp data can be interpreted as 12d Model strings with entity-names corresponding to the Geocomp entity numbers.

Hence the Geocomp **entity numbers** are used as the entity-names for matching with a map file.

If no match is found with the map file, the default colours and model given in the Geocomp read panel are used. The point-line type is taken to be **point** for point feature entities (entity type 5) and **line** point-line type for all other entities. The geocomp entity number is used as the string name and the style set to 1.

WARNING

If the **breakline type** in the map file is set to point or line and this does not match the point or string type coming from the geocomp file, then the string is placed in the **defaults** model. This is a consistency check for entities that can only be a point **or** a line breakline type (but not both). To disable this feature, a ***** can be used in the map file for the breakline type and then the breakline type is determined by whether the Geocomp entity is a point feature or not.

Summary

12d Model **recognizes** the following Geocomp entity types

Entity Number	Entity type
1	two point lines
3	arc defined by start point, point on arc, end point
4	arc defined by start point, arc centre, end point
5	point feature
6	circle defined by centre and a point on the circle

12d Model **recognizes** the Geocomp null values of -9999

12d Model **ignores** the entity type

2	irregular lines

A Geocomp map file can be used to specify the string name, breakline type, colour, style and model for strings read from the Geocomp text files.

Note - the breakline type of a string of Geocomp point features (entity type 5) will always be set to **point** regardless of the Geocomp map file.

8.1.14 Keays Input

Position of option on menu: **File I/O =>Data input =>Keays**

The **Keays** input option is designed to read in Keays (RoadPak) files in trf format.

The Keays data has a code and notes, and the code and notes can be used with a 12d Model map file to define models, colours, linestyles *etc*.

The Keays note is made up of individual notes separated by spaces. Each individual note (in order) can be used as part of the entity-name for use with the map file.

In the Keays map file, the multi-level key is given as a single key made up of the required items separated by **|** (with no additional spaces) in the order:

Code | note 1| note 2 | ... | note n

For example, the key:

BD|building| means code BD, note 1 = "building"

A ***** can be used for any of the items to indicate that no match is required for that item.

BD|*|top| means code BD, anything for note 1, note 2 = "top"

A * and nothing else after the code means that all notes are accepted.

BD* means code BD and any notes

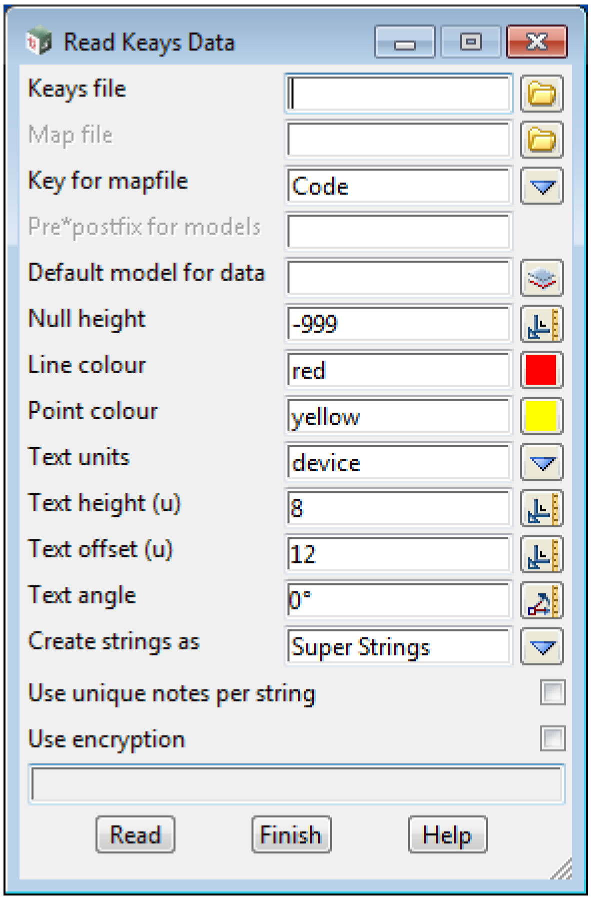
A * at the end of the notes means any note is a match from that point onwards.

BD|*|top|* means code BD, anything for note 1, note2 = "top", and any note after note 2.

If a map file is not used or no match is found in the map file, the Keays data is placed into the default model with its code as the string name.

See the section 8.8.1 Create/Edit a Map File for information about 12d map files.

On selecting the **Keays** option, the **Read Keays Data** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Keays file	file box		*.trf

name of the Keays trf file to be read in.

Map file	map file box		*.mf files

if non-blank, the name of the map file to be used for all strings read in. If blank, no map file is used. The Keays code and note can be used as the entity-name to match against the key in the map file.

See the section 8.8.1 Create/Edit a Map File *for information about 12d map files.*

Key for map file	input	code	code, code/note

*if **code**, the Keays code is used as the key for the map file.*
*If **code/note**, the Keays code and note is used as the key for the map file.*

Pre*postfix for models	pre*postfix box		

if non-blank, a prefix and a postfix to be applied to the model names used in the Map File.

Go to the section 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

Default model for data	model box		available models

name of the model that any unmapped data is placed in. The model will be created if it does not already exist. This field must be filled in.

Null height	input	-999	

*if **non-blank,** any Keays z values equalling this value are taken as null values.*

Line colour	colour box	red	available colours

colour for Keays string s.

Point colour	colour box	yellow	available colours

colour for Keays points that are not part of strings.

Text units	input	pixels	pixels, world

units for the height of the text label.

Text height (u) input

height of the text (in text units).

Text offset (u) input

distance (in text units) to offset the text from its (x,y) placement position.

Text angle input 0

angle of the note text.

Create strings as	input	Super Strings	3d, 4d, Polyline, Super

type of strings to create.

Use unique notes per string tick box

if ticked, a change of code or note is used to break Keays string data into strings. This also applies to the point ids and notes text.

Use encryption tick box

if ticked, then the last 4 characters of the code is used to denote tinability and boundary information.

Read button

read the data in.

8.1.15 LandXML Input

Position of menu: **File I/O =>Data input =>Landxml**

LandXML is a format that attempts to cover some civil and surveying entities. It includes some of the geometry definitions but not display information such as colours, styles *etc*.

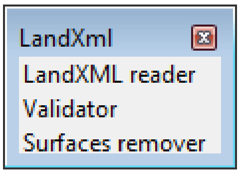
The format was started by Autodesk but the LandXML committee now appears to be defunct.

Unfortunately due to limitations in the LandXML format, many vendors added their own proprietary extensions which makes the LandXML format of only limited value and a different tailored LandXML reader is required for each vendor variation. The names and definitions of transitions can vary from vendor to vendor.

Finally the methodology behind the LandXML model is based on the restricted US approach of using X-sections and not strings to model data. This makes it unsuitable except for simple civil models.

One use of LandXML is to get some access to data from vendor software that only has unpublished encrypted proprietary binary formats. For example, AutoCAD DWG, Civil3D, Microstation DGN V8, InRoads and InRail.

This option is under continual developed as the LandXML standard keep changing.



read a Land XML file
validate a Land XML file
remove the surfaces section from a Land XML file

See

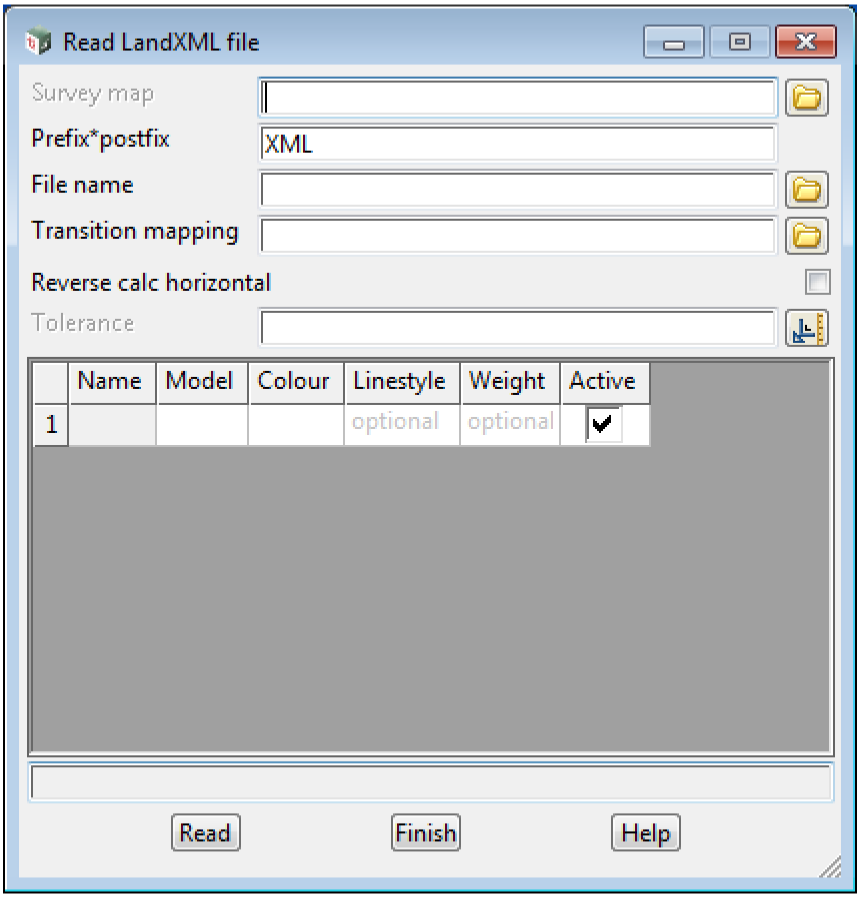
8.1.15.1 Read LandXML
8.1.15.2 LandXML Validator
8.1.15.3 LandXML Surfaces Remover

Note: ***12d Model*** has special LandXML readers for LINZ-XML (for LandOnline NZ), ePlan NSW, ePlan Queensland, ePlan Victoria and Leica-XML.

8.1.15.1 Read LandXML

Position of option on menu: **File I/O =>Data input =>LandXML =>LandXML reader**

Selecting **LandXML reader** brings up the **Read LandXML File** panel.



The fields and buttons used in this panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Survey map map file box *.mf files

*if **not blank,** the name of a map file to used for a special mapping of data in the Survey sections of a Land XML file.*
*If **blank**, no map file is used.*

See the section 8.8.1 Create/Edit a Map File *for information about 12d map files.*

Pre*postfix for models pre*postfix box

if non-blank, a prefix and a postfix to be applied to the model names used in the Model field.
Go to the section 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

File file box *.xml files

name of the LandXML file to read in.

Transition mapping file box *.trans_map files

file containing the mapping of transitions between the names of LandXML transitions and equivalent transitions in 12d Model.

Reverse calc horizontal tick box

if ticked, try and turn the segments of an Alignment in the LandXML file into an editable Super

Alignment.

Tolerance real box

if not zero,

Grid for Sections of the Land XML File

Name

name of the section of the Land xml definition that is in the file. If it is a Survey section, is the name of the plan.

Model model cell available models

model to read the Land xml data into

Colour colour cell available colours

colour to use for the strings in the xml file

Linestyle linestyle cell available linestyles

linestyle to use for the strings

Weight input

weight to use for the strings

Active tick box

if ticked, load this section of the Land xml file in.

Read button

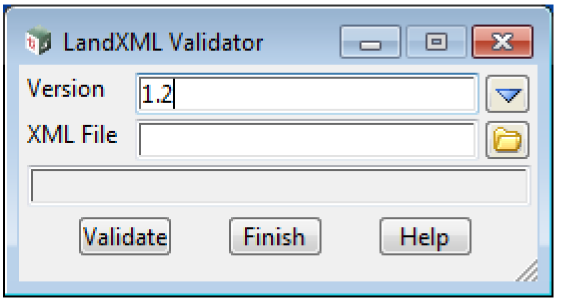
read the data into the models given in the **Model** *field of the grid.*

8.1.15.2 LandXML Validator

Position of option on menu: File I/O =>Data input =>LandXML =>Validator

This option validates an XML file against the LandXML schema.

Selecting **Validator** brings up the **LandXML Validator** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Version	choice box	1.2	LandXML versions

the version of Land XML to validate the XML file against.

XML File	file box		*.xml files

name of the XML file to validate against the Land XML Schema.

Validate	button		

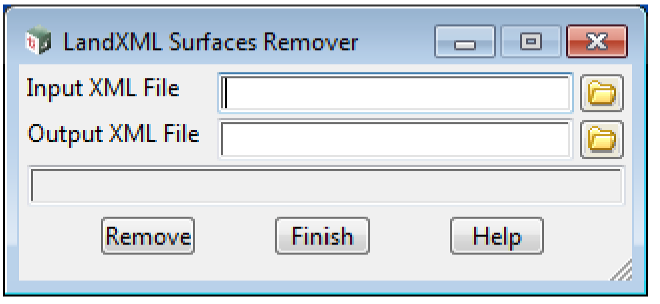
validate the XML file against the LandXML schema.

8.1.15.3 LandXML Surfaces Remover

Position of option on menu: **File I/O =>Data input =>LandXML =>Surfaces remover**

This option removes the Surfaces section from a given LandXML file.

Selecting **Surfaces remover** brings up the **LandXML Surfaces Remover** panel.



The fields and buttons used in this panel have the following functions.

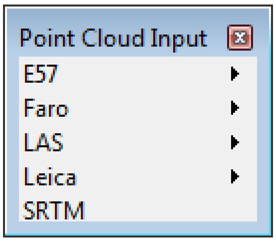
Field Description	Type	Defaults	Pop-Up
Input XML File	file box		*.xml files
name of the Land XML file to remove the Surfaces section from.			
Output XML File	file box		*.xml files
name of the file to write the landXML XML file out to but with the Surfaces section removed.			
Remove	button		
create a new file with the Surfaces sections removed.			

8.1.16 Input Point Cloud Files

Position of menu: **File I/O =>Data input =>Point clouds**

The options under **Point Cloud Input** read data in various formats.

The **Point Cloud Input** walk-right menu is



read data in E57 format
read data in Faro format
read data in LAS format
read data in Leica format
read data in SRTM format

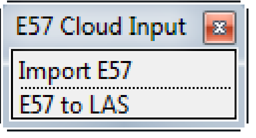
See

E57	8.1.16.1 E57 Cloud Input
Faro	8.1.16.2 Faro Point Cloud Input
LAS	8.1.16.3 LAS Cloud Input
Leica	8.1.16.4 Leica Point Cloud Input
SRTM	8.1.16.5 Read SRTM Files

8.1.16.1 E57 Cloud Input

Position of menu: **File I/O =>Data input =>Point clouds =>E57**

The **E57 Cloud Input** walk-right menu is



import E57 files
convert E57 files to LAS files

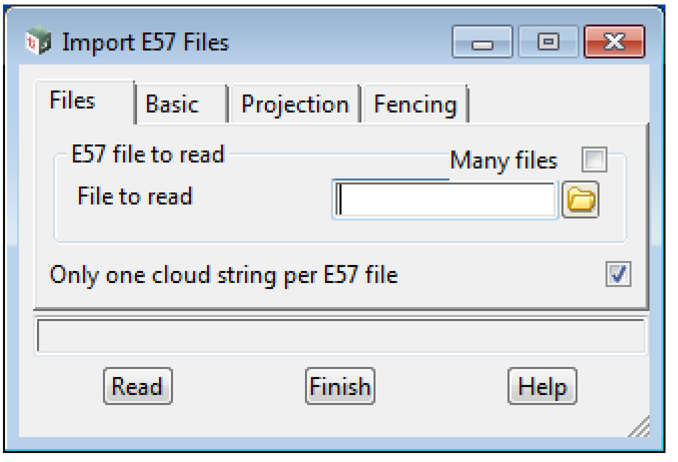
See

Import E57	8.1.16.1.1 Import E57 Files
E57 to LAS	8.1.16.1.2 E57 to LAS Files

8.1.16.1.1 Import E57 Files

Position of option on menu: **File I/O =>Data Input =>Point Clouds =>E57 =>Import E57**

Selecting **Import E57** brings up the **Import E57 Files** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Files tab

Many files tick box not ticked

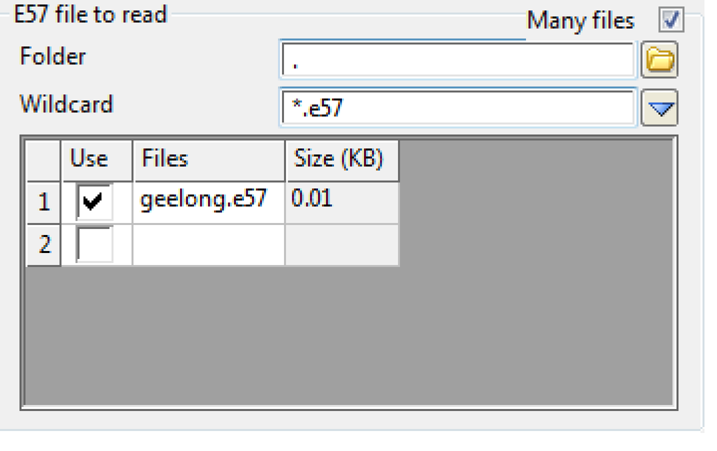
*if **ticked**, a grid to allow multiple e57 files to be read in is opened. A wild card is used to select all the files to be read in.*

Folder file box

folder to search for files using the **Wild card**

Wildcard input

wild card to use in search for files in the given folder



Use tick box

*if **ticked**, read in the file*

Files output

name of the file in the folder

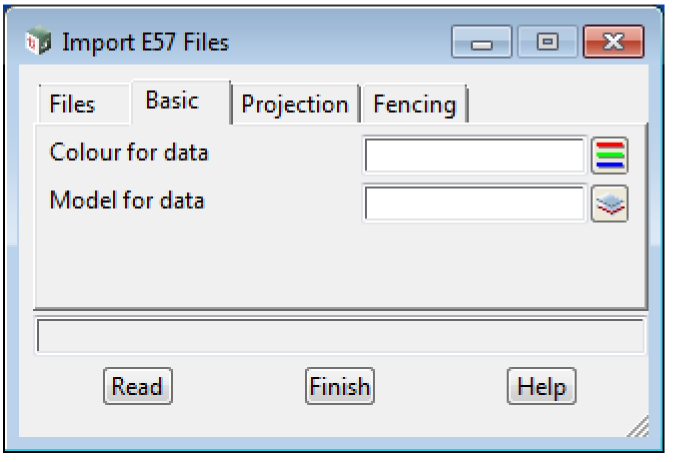
Size output

file size

Only one cloud string per E57 file tick box ticked

if ticked, all cloud points will be combined into one 12d cloud string.

Basic tab



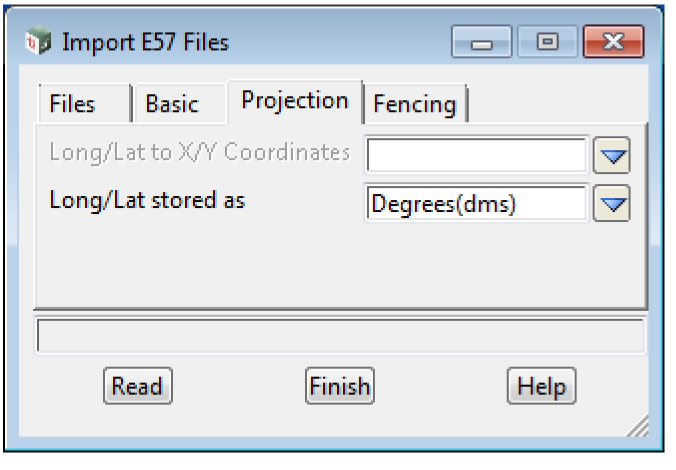
Colour for data colour box

string colour to be used if the cloud object does not have colour for each point

Model for data model box

name of the model that the data is to be placed in. The model will be created if it does not already exist. This field must be filled in.

Projection tab



Long/Lat to X/Y Coordinates choice box available projections

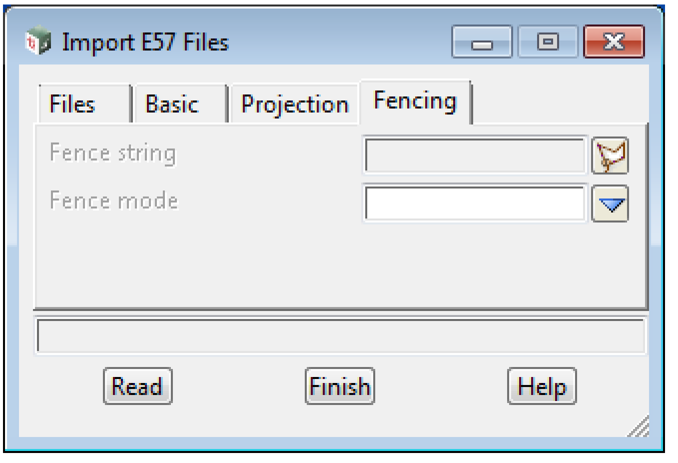
if ***non-blank****, the cartographic projection to apply to the longitude-latitude values.*
If ***blank****, the co-ordinates are not transformed from (longitude, latitude) to (x,y). Hence the initial (x,y) co-ordinates are transformed to (longitude, latitude) by the transformation given in the* **x/y co-ordinates to Long/Lat** *field and then left in (longitude, latitude). Note that in the southern hemisphere, the latitude values are* ***negative****.*

Long/Lat stored as choice box degrees radians
degrees

decimal degrees

format for the longitude and latitudes - either radians, degrees (in 4.17.1 HP Notation *for degrees, minutes and seconds) or decimal degrees.*

Fencing tab



Fence string polygon box

string to use to restrict the data being read in.

Fence mode choice box

File inside
File inside/crossing
File outside
File outside/crossing
File crossing

File inside - read file in if it is totally inside the polygon

File inside/crossing - read file in if it is totally inside, or crossing the polygon

File outside - read file in if it is totally outside the polygon

File outside/crossing - read file in if it is totally outside, or crossing the polygon

File crossing - read file in if it is crossing the polygon

***Note** - only whole files are read in.*

Button at bottom

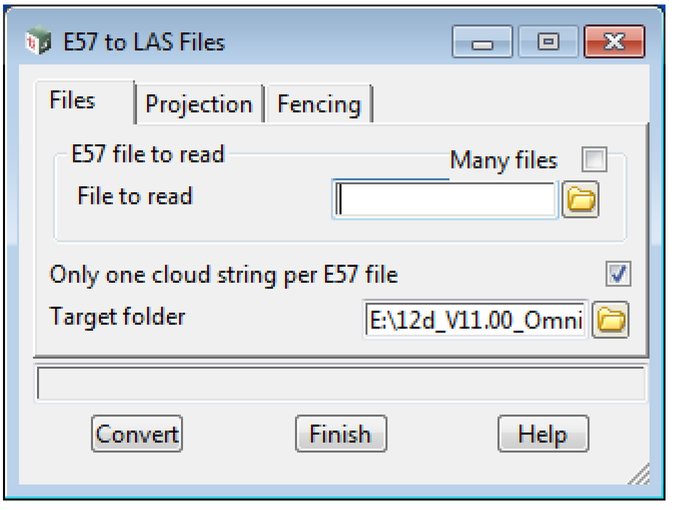
Read button

read the data into the model given in the model field.

8.1.16.1.2 E57 to LAS Files

Position of option on menu: **File I/O =>Data Input =>Point Clouds =>E57 =>E57 to LAS**

Selecting **E57 to LAS** brings up the **E57 to LAS Files** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Files tab

Many files tick box not ticked

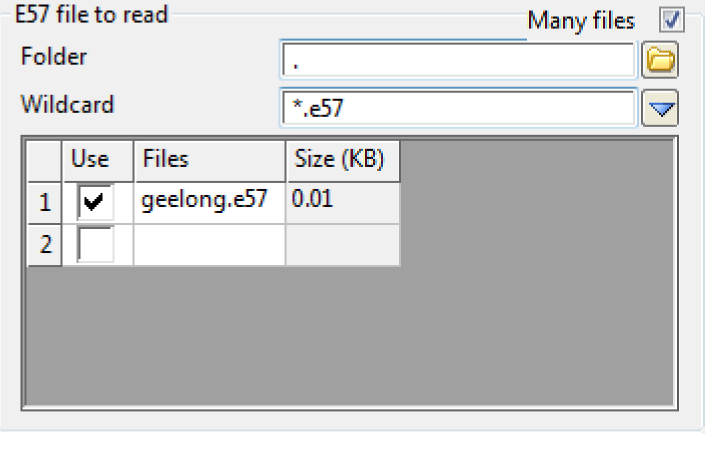
*if **ticked**, a grid to allow multiple e57 files to be converted is opened. A wild card is used to select all the files to be read in.*

Folder file box

folder to search for files using the **Wild card**

Wildcard input

wild card to use in search for files in the given folder



Only one cloud string per E57 file tick box ticked

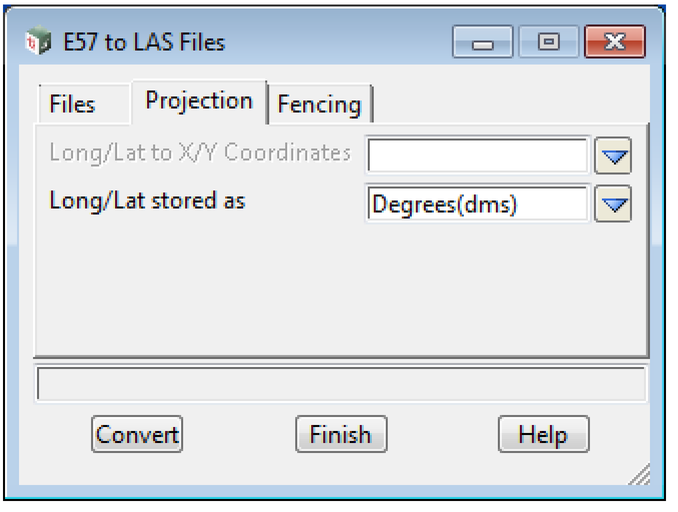
if ticked, all cloud points will be combined into one 12d cloud string.

Target folder file box

*folder where the output LAS file(s) will be created. As it is unknown how many files will be created, it is good practice to specify an empty folder for these files. For example, if the input file is **geelong.e57**, the output file(s) will be **geelong.las** OR **geelong_1.las**, **geelong_2.las**, etc.*

Note: Multiple las files will only be created if the tick box **Only one cloud string per file** *is unticked and the input file has multiple clouds.*

Projection tab



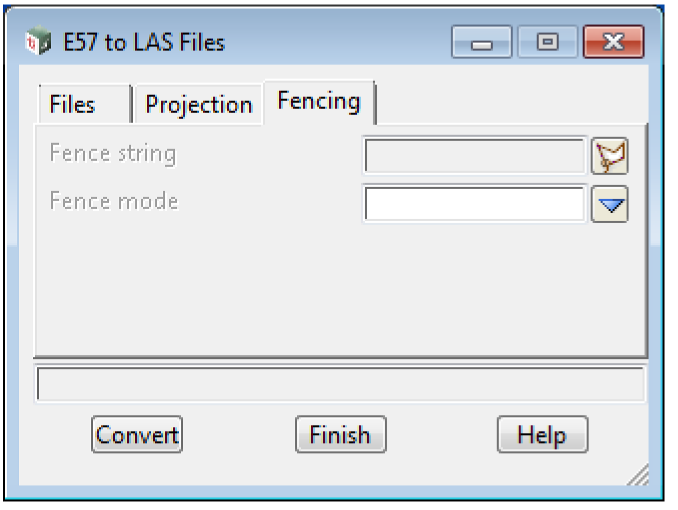
Long/Lat to X/Y Coordinates choice box available projections

*if **non-blank**, the cartographic projection to apply to the longitude-latitude values.*
*If **blank**, the co-ordinates are not transformed from (longitude, latitude) to (x,y). Hence the initial (x,y) co-ordinates are transformed to (longitude, latitude) by the transformation given in the* **x/y co-ordinates to Long/Lat** *field and then left in (longitude, latitude). Note that in the southern hemisphere, the latitude values are **negative**.*

Long/Lat stored as choice box degrees radians
degrees
decimal degrees

format for the longitude and latitudes - either radians, degrees (in 4.17.1 HP Notation *for degrees, minutes and seconds) or decimal degrees.*

Fencing tab



Fence string polygon box

string to use to restrict the data being read in.

Fence mode choice box

File inside
File inside/crossing
File outside
File outside/crossing
File crossing

File inside - read file in if it is totally inside the polygon

File inside/crossing - read file in if it is totally inside, or crossing the polygon

File outside - read file in if it is totally outside the polygon

File outside/crossing - read file in if it is totally outside, or crossing the polygon

File crossing - read file in if it is crossing the polygon

***Note** - only whole files are read in.*

Button at bottom

Convert button

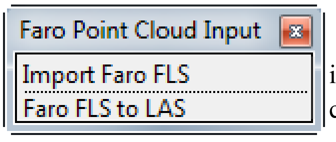
convert the data into the model given in the model field.

8.1.16.2 Faro Point Cloud Input

Position of menu: **File I/O =>Data input =>Point clouds =>Faro**

This section of documentation is a work in progress and will be updated in subsequent releases.

The **Faro Point Cloud Input** walk-right menu is



import Faro FLS files
convert Faro FLS files to LAS files

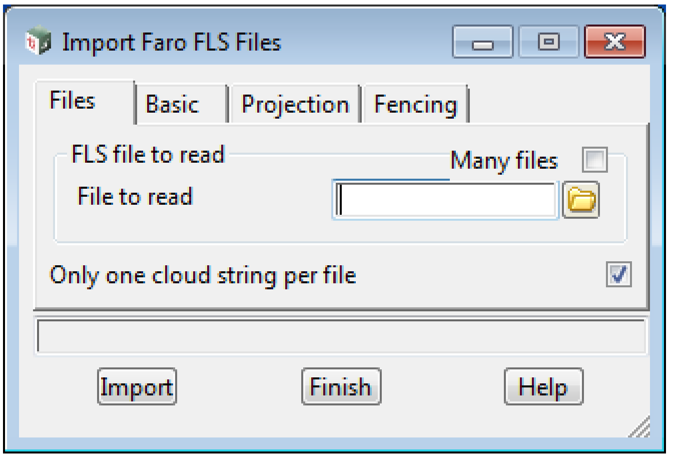
See

Import Faro FLS 8.1.16.2.1 Import Faro FLS Files
Faro FLS to LAS 8.1.16.2.2 Faro Cloud FLS to LAS Files

8.1.16.2.1 Import Faro FLS Files

Position of option on menu: **File I/O =>Data Input =>Point Clouds =>Faro =>Import Faro FLS**

Selecting **Import Faro FLS** brings up the **Import Faro FLS Files** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Files tab			
Many files	tick box	not ticked	

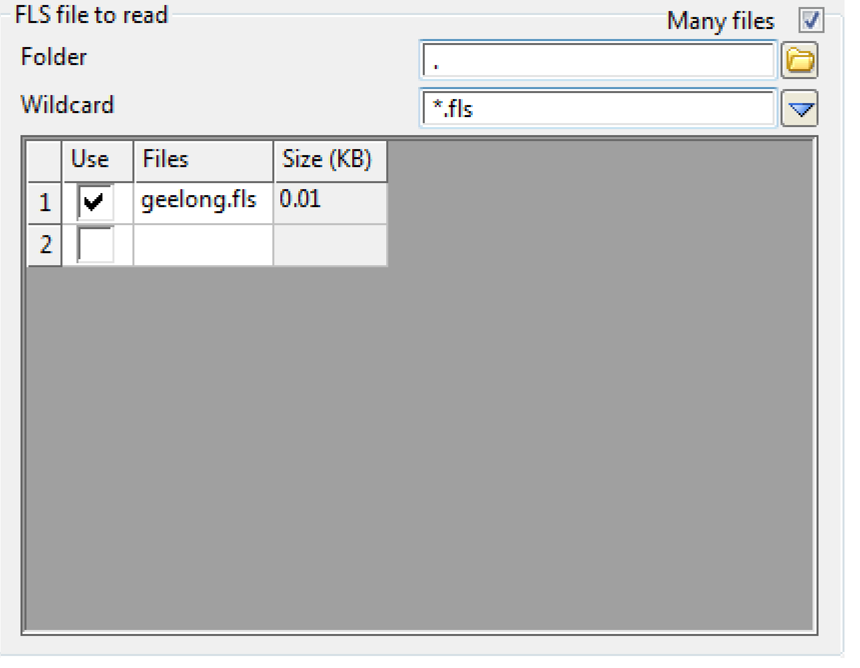
*if **ticked**, a grid to allow multiple fls files to be read in is opened. A wild card is used to select all the files to be read in.*

Folder	file box		

folder to search for files using the **Wild card**

Wildcard	input		

wild card to use in search for files in the given folder



Use tick box

*if **ticked**, read in the file*

Files output

name of the file in the folder

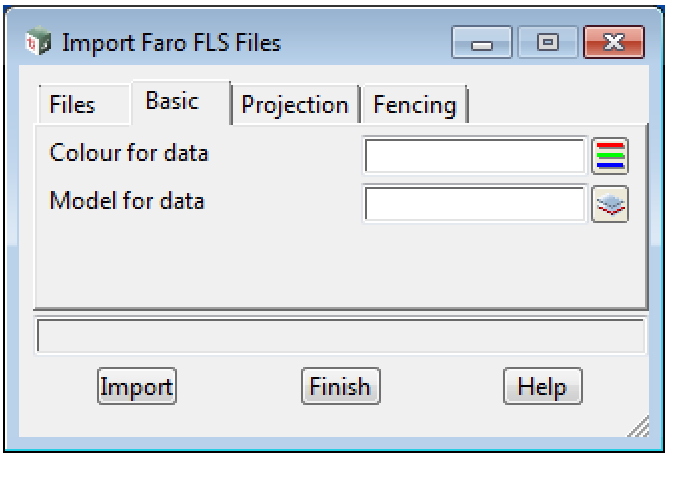
Size output

file size

Only one cloud string per file tick box ticked

if ticked, all cloud points will be combined into one 12d cloud string.

Basic tab



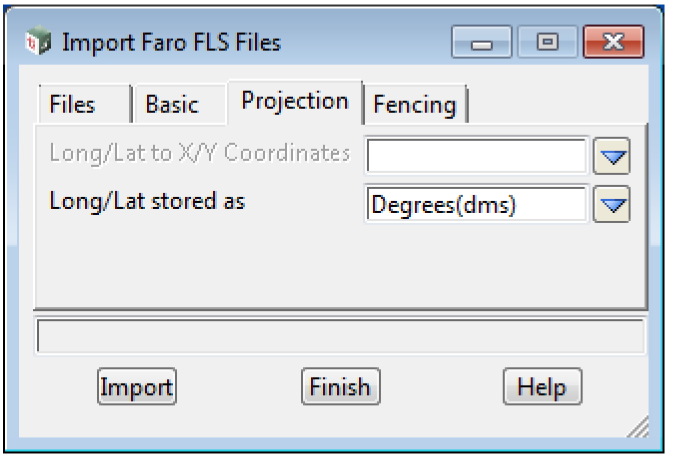
Colour for data colour box

string colour to be used if the cloud object does not have colour for each point

Model for data model box

name of the model that the data is to be placed in. The model will be created if it does not already exist. This field must be filled in.

Projection tab



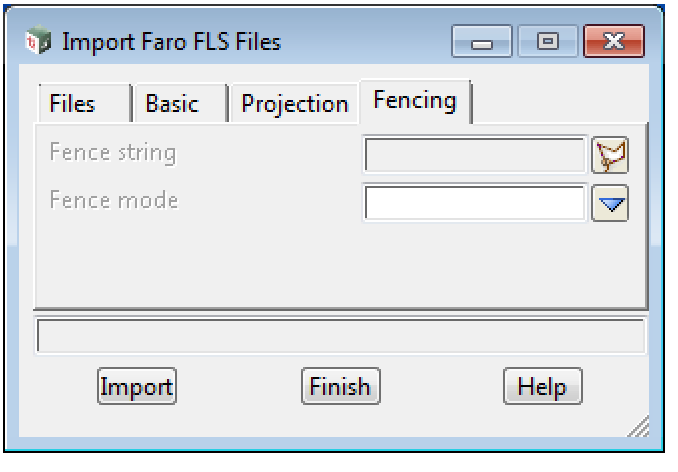
Long/Lat to X/Y Coordinates choice box available projections

*if **non-blank**, the cartographic projection to apply to the longitude-latitude values. If **blank**, the co-ordinates are not transformed from (longitude, latitude) to (x,y). Hence the initial (x,y) co-ordinates are transformed to (longitude, latitude) by the transformation given in the* **x/y co-ordinates to Long/Lat** *field and then left in (longitude, latitude). Note that in the southern hemisphere, the latitude values are **negative**.*

Long/Lat stored as choice box degrees radians
degrees
decimal degrees

format for the longitude and latitudes - either radians, degrees (in 4.17.1 HP Notation *for degrees, minutes and seconds) or decimal degrees.*

Fencing tab



Fence string polygon box

string to use to restrict the data being read in.

Fence mode choice box File inside
File inside/crossing
File outside

File outside/crossing
File crossing

File inside - read file in if it is totally inside the polygon

File inside/crossing - read file in if it is totally inside, or crossing the polygon

File outside - read file in if it is totally outside the polygon

File outside/crossing - read file in if it is totally outside, or crossing the polygon

File crossing - read file in if it is crossing the polygon

***Note** - only whole files are read in.*

Button at bottom

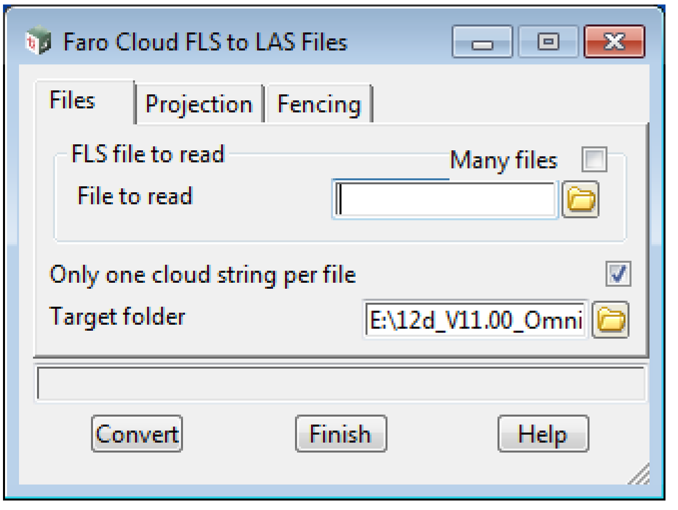
Import button

import the data into the model given in the model field.

8.1.16.2.2 Faro Cloud FLS to LAS Files

Position of option on menu: **File I/O =>Data Input =>Point Clouds =>Faro =>Faro FLS to LAS**

Selecting **Faro FLS to LAS** brings up the **Faro Cloud FLS to LAS Files** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Files tab

Many files tick box not ticked

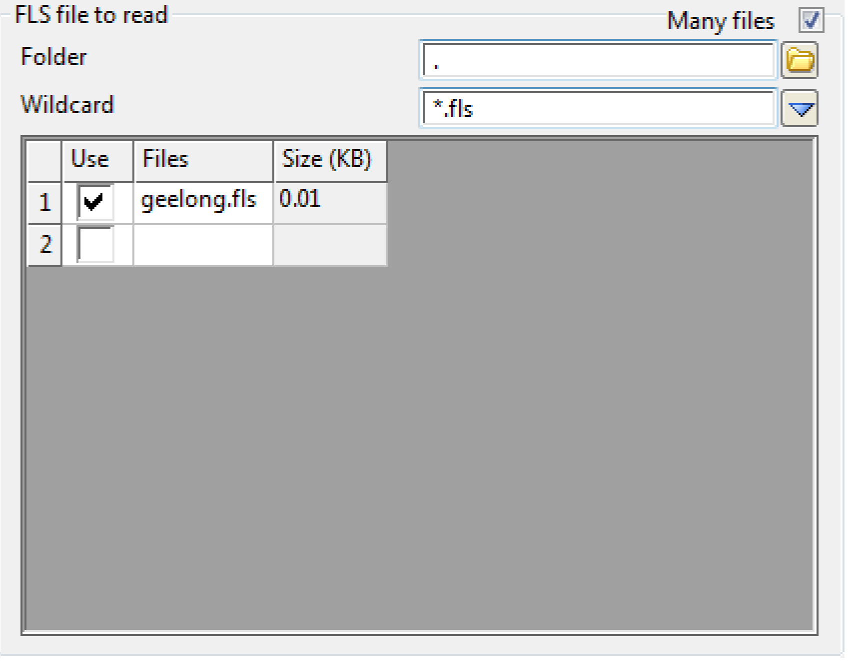
*if **ticked**, a grid to allow multiple fls files to be read in is opened. A wild card is used to select all the files to be read in.*

Folder file box

folder to search for files using the **Wild card**

Wildcard input

wild card to use in search for files in the given folder



Use tick box

*if **ticked**, read in the file*

Files output

name of the file in the folder

Size output

file size

Only one cloud string per file tick box ticked

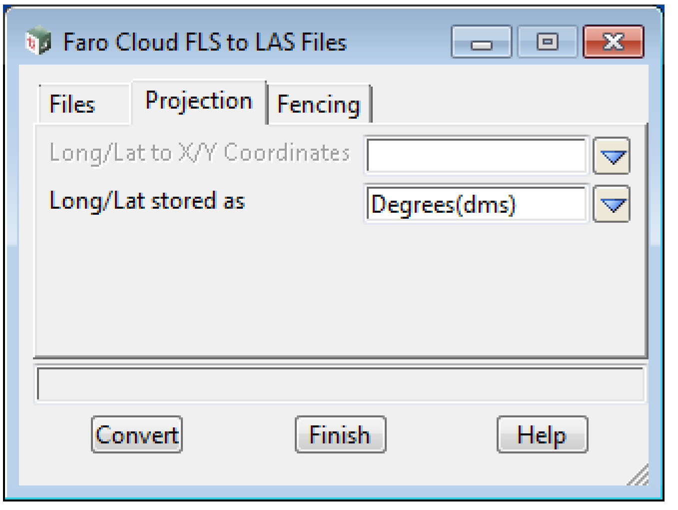
if ticked, all cloud points will be combined into one 12d cloud string.

Target folder file box

*folder where the output LAS file(s) will be created. As it is unknown how many files will be created, it is good practice to specify an empty folder for these files. For example, if the input file is **geelong.fls**, the output file(s) will be **geelong.las** OR **geelong_1.las**, **geelong_2.las**, etc.*

Note: Multiple las files will only be created if the tick box **Only one cloud string per file** *is unticked and the input file has multiple clouds.*

Projection tab



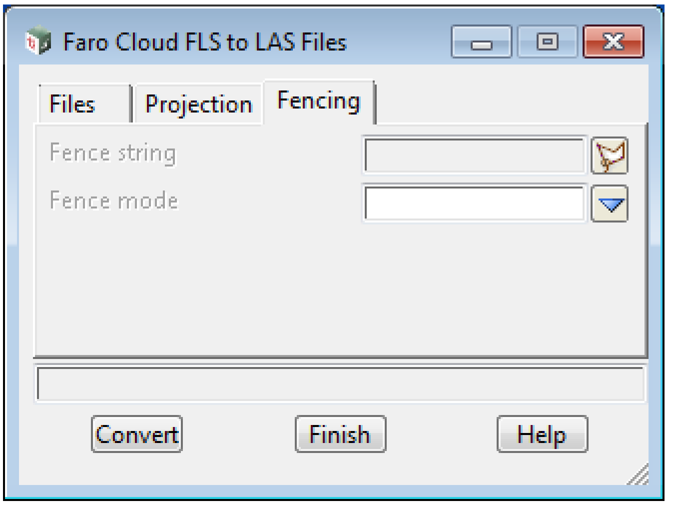
Long/Lat to X/Y Coordinates choice box available projections

*if **non-blank**, the cartographic projection to apply to the longitude-latitude values.*
*If **blank**, the co-ordinates are not transformed from (longitude, latitude) to (x,y). Hence the initial (x,y) co-ordinates are transformed to (longitude, latitude) by the transformation given in the* **x/y co-ordinates to Long/Lat** *field and then left in (longitude, latitude). Note that in the southern hemisphere, the latitude values are **negative**.*

Long/Lat stored as choice box degrees radians
degrees
decimal degrees

format for the longitude and latitudes - either radians, degrees (in 4.17.1 HP Notation *for degrees, minutes and seconds) or decimal degrees.*

Fencing tab



Fence string polygon box

string to use to restrict the data being read in.

Fence mode choice box File inside
File inside/crossing
File outside
File outside/crossing
File crossing

File inside - read file in if it is totally inside the polygon

File inside/crossing - read file in if it is totally inside, or crossing the polygon

File outside - read file in if it is totally outside the polygon

File outside/crossing - read file in if it is totally outside, or crossing the polygon

File crossing - read file in if it is crossing the polygon

***Note** - only whole files are read in.*

Button at bottom

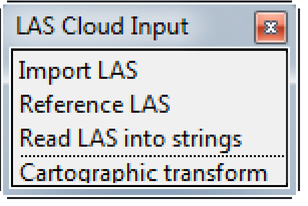
Convert button

convert the data into the model given in the model field.

8.1.16.3 LAS Cloud Input

Position of menu: **File I/O =>Data input =>Point clouds =>LAS**

The **LAS Cloud Input** walk-right menu is



import LAS files
reference LAS files
read LAS files into strings
transform LAS Cartographic

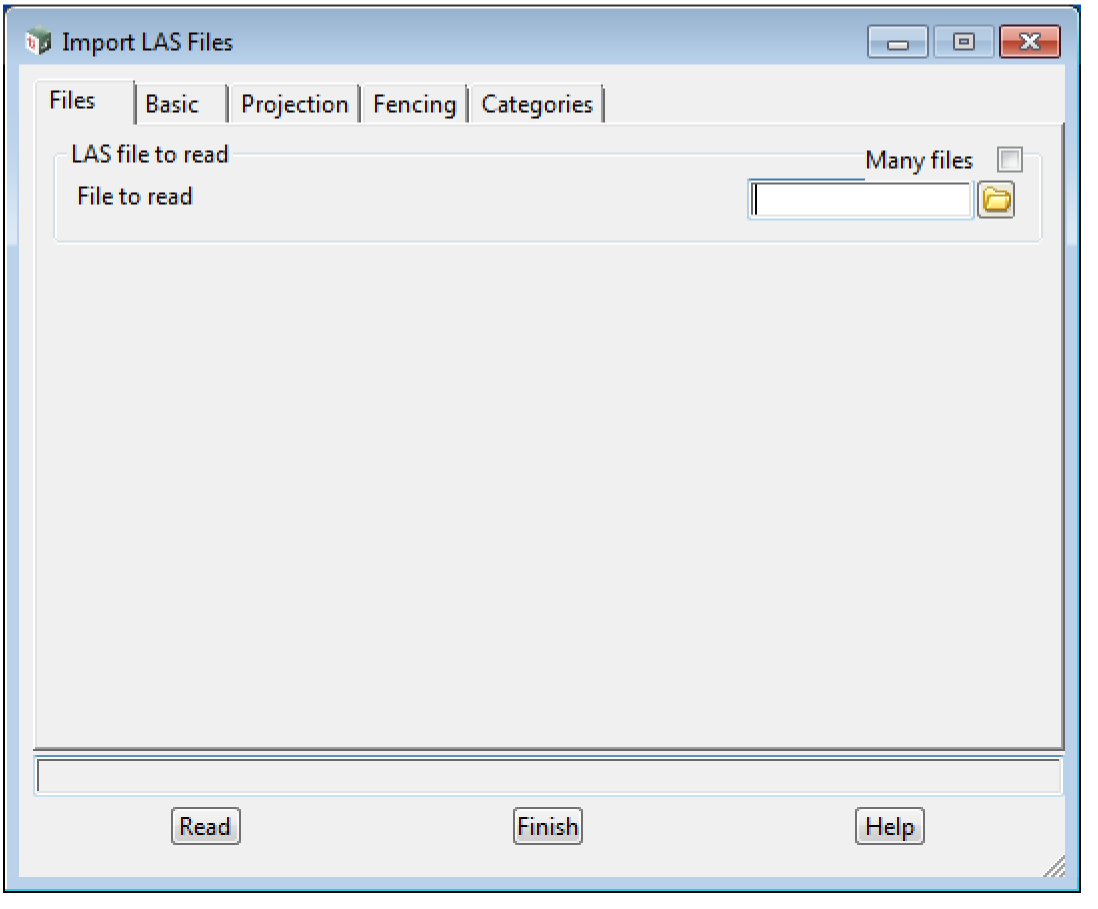
See

Import LAS 8.1.16.3.1 Import LAS Files
Reference LAS 8.1.16.3.2 Reference LAS Files
Read LAS into strings 8.1.16.3.3 Read LAS Files into Strings
Cartographic transform 8.1.16.3.4 LAS Cartographic

8.1.16.3.1 Import LAS Files

Position of option on menu: **File I/O =>Data Input =>Point Clouds =>LAS =>Import LAS**

Selecting **Import LAS** brings up the **Import LAS Files** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Files tab

Many files	tick box	not ticked	

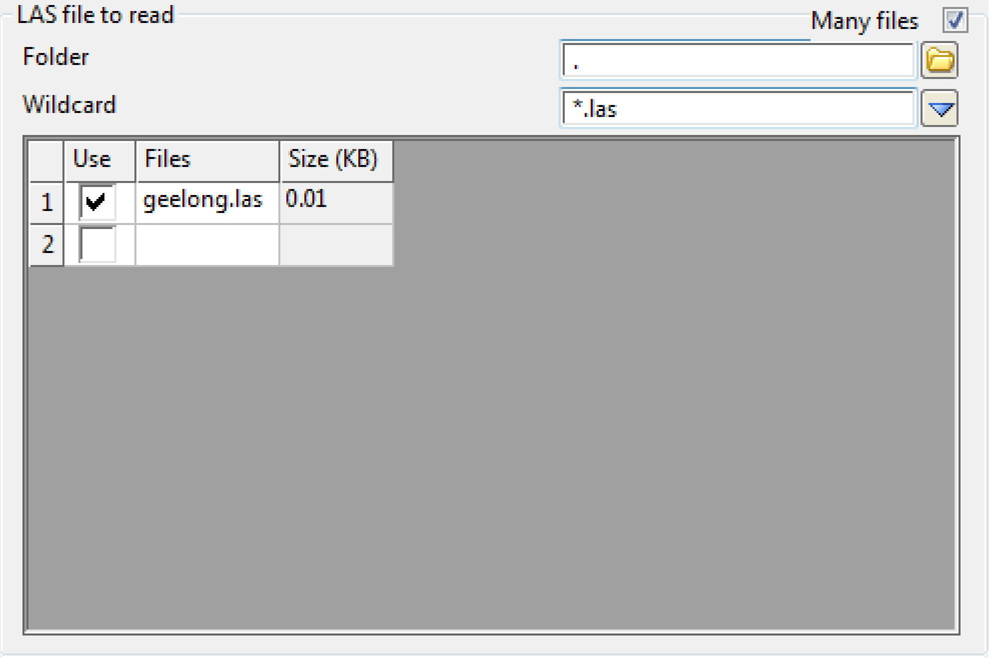
*if **ticked**, a grid to allow multiple las files to be read in is opened. A wild card is used to select all the files to be read in.*

Folder	file box		

folder to search for files using the **Wild card.**

Wildcard	input		

wild card to use in search for files in the given folder.



Use tick box

*if **ticked**, read in the file*

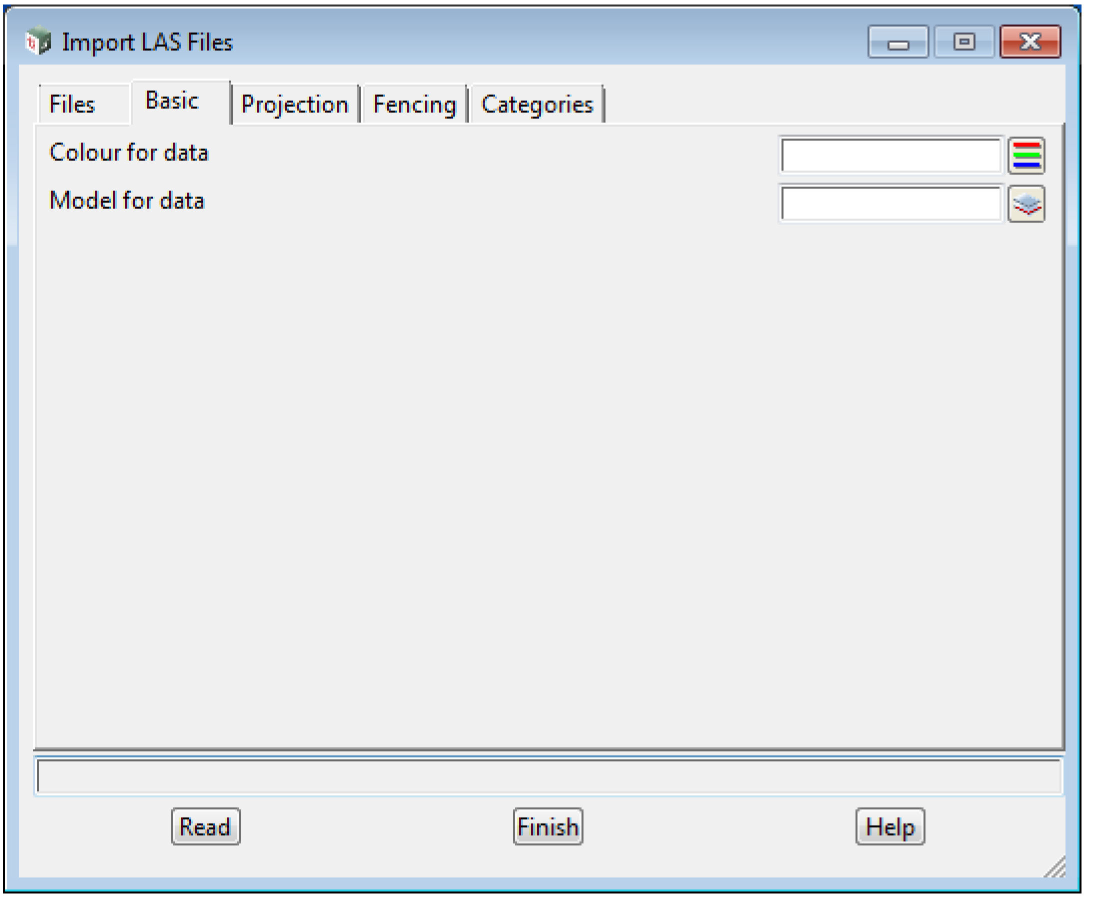
Files output

name of the file in the folder

Size output

file size

Basic tab



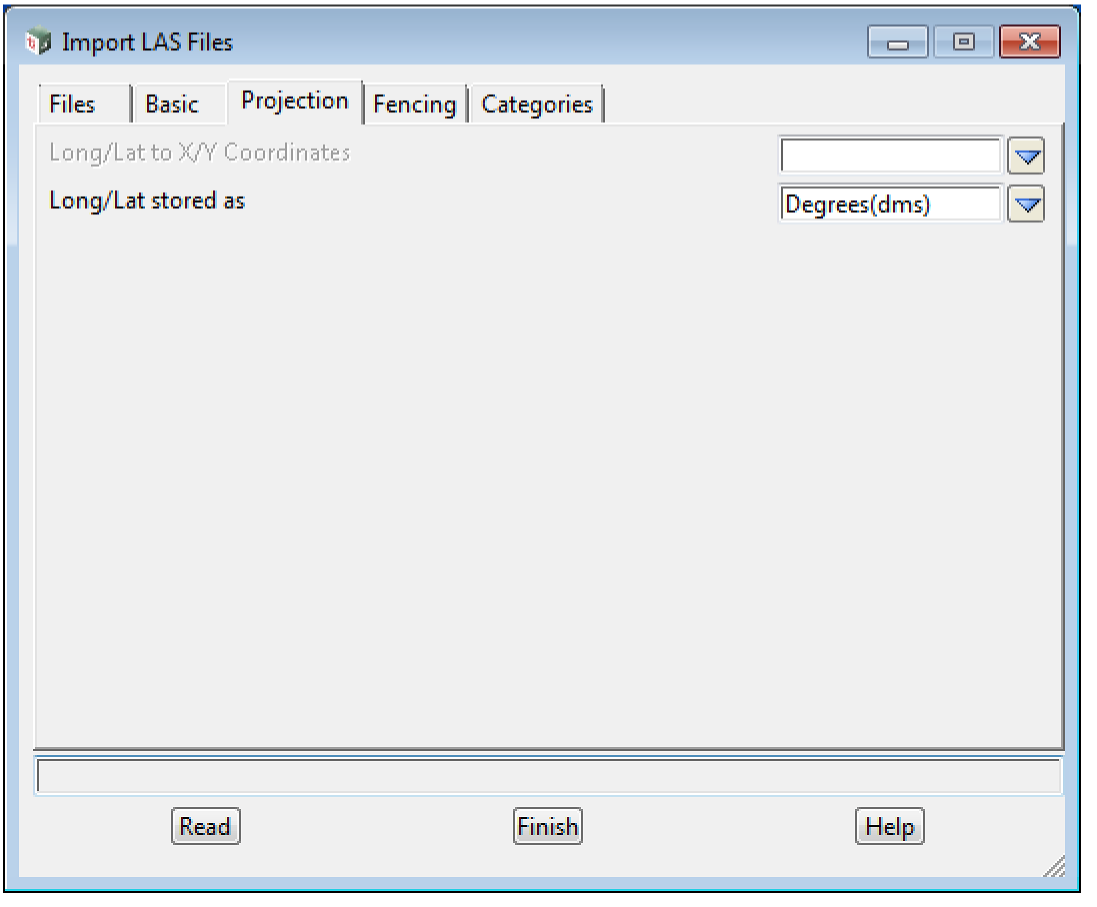
Colour for data colour box

string colour to be used if the cloud object does not have colour for each point

Model for data model box

name of the model that the data is to be placed in. The model will be created if it does not already exist. This field must be filled in.

Projection tab



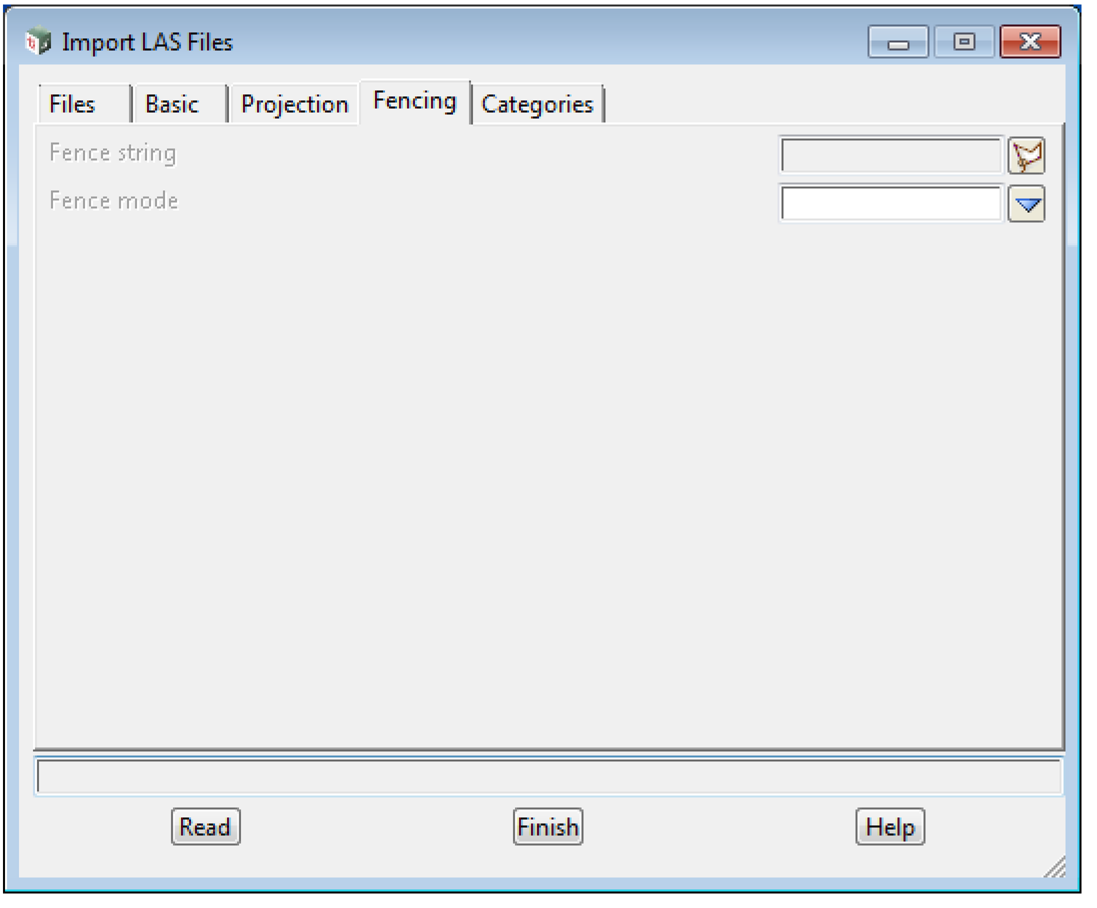
Long/Lat to X/Y Coordinates choice box available projections

*if **non-blank**, the cartographic projection to apply to the longitude-latitude values.*
*If **blank**, the co-ordinates are not transformed from (longitude, latitude) to (x,y). Hence the initial (x,y) co-ordinates are transformed to (longitude, latitude) by the transformation given in the* **x/y co-ordinates to Long/Lat** *field and then left in (longitude, latitude). Note that in the southern hemisphere, the latitude values are **negative**.*

Long/Lat stored as choice box degrees radians
degrees
decimal degrees

format for the longitude and latitudes - either radians, degrees (in 4.17.1 HP Notation *for degrees, minutes and seconds) or decimal degrees.*

Fencing tab



Fence string polygon box

string to use to restrict the data being read in.

Fence mode choice box

File inside
File inside/crossing
File outside
File outside/crossing
File crossing

File inside - read file in if it is totally inside the polygon

File inside/crossing - read file in if it is totally inside, or crossing the polygon

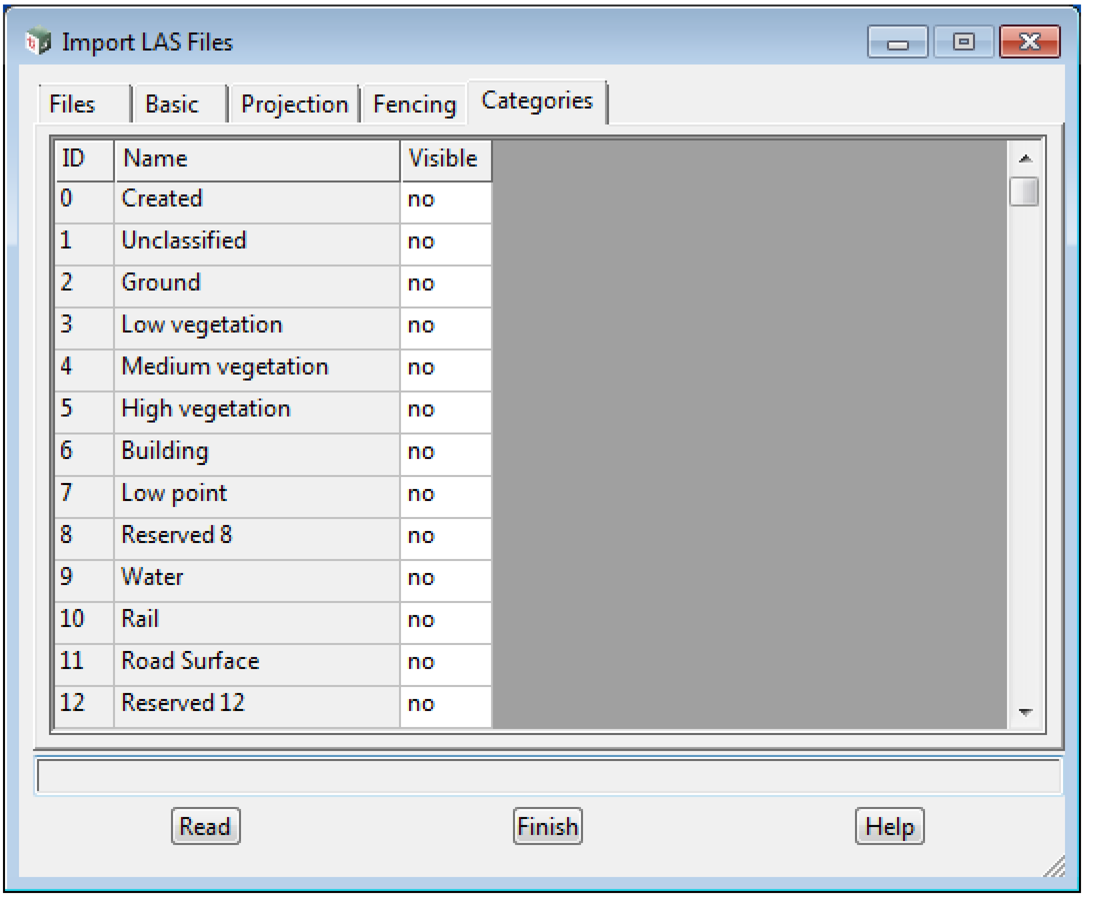
File outside - read file in if it is totally outside the polygon

File outside/crossing - read file in if it is totally outside, or crossing the polygon

File crossing - read file in if it is crossing the polygon

***Note** - only whole files are read in.*

Categories tab



ID

Name

Visible

Button at bottom

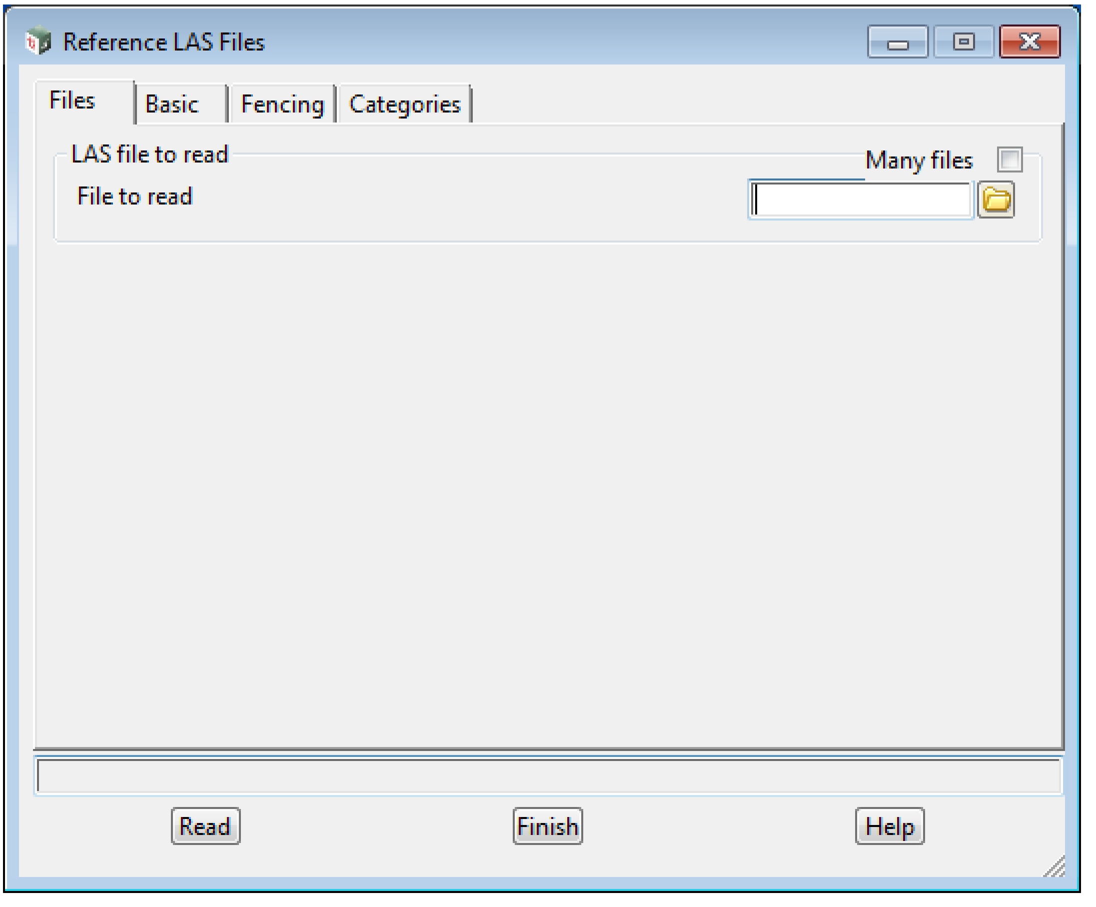
Read **button**

read the data into the model given in the model field.

8.1.16.3.2 Reference LAS Files

Position of option on menu: **File I/O =>Data Input =>Point Clouds =>LAS =>Reference LAS**

Selecting **Reference LAS** brings up the **Reference LAS Files** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Files tab			
Many files	tick box	not ticked	

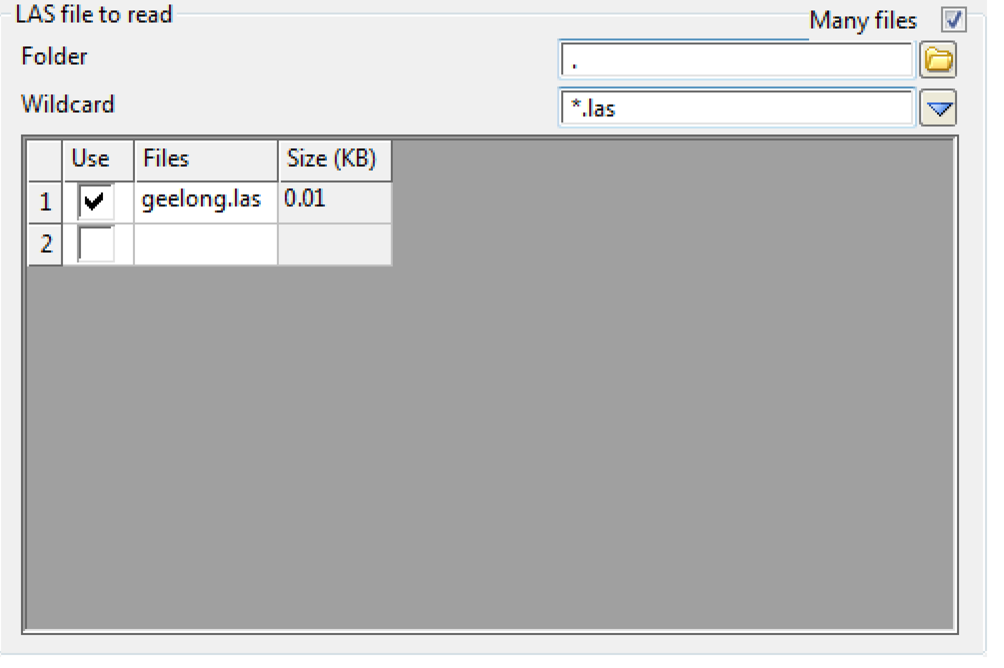
*if **ticked**, a grid to allow multiple las files to be read in is opened. A wild card is used to select all the files to be read in.*

Folder	file box		

folder to search for files using the **Wild card**

Wildcard	input		

wild card to use in search for files in the given folder



Use tick box

*if **ticked**, read in the file*

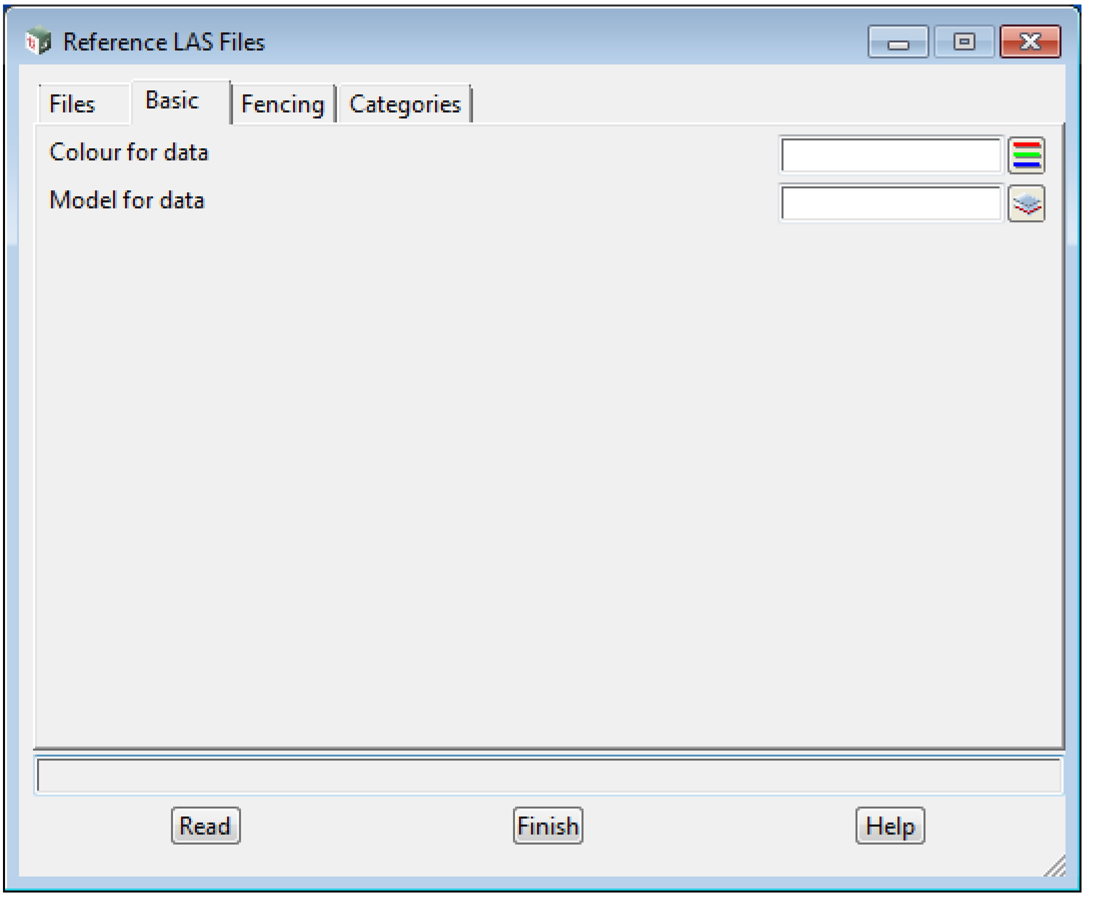
Files output

name of the file in the folder

Size output

file size

Basic tab



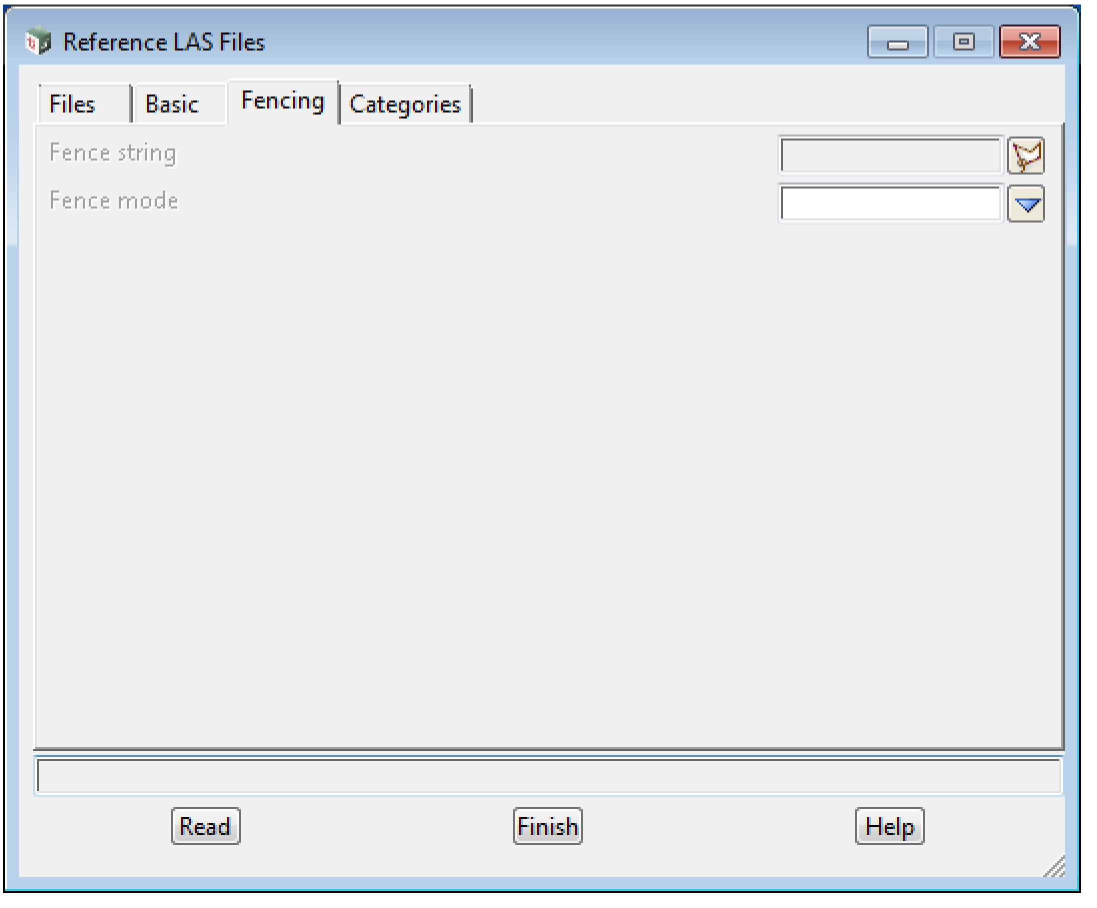
Colour for data colour box

string colour to be used if the cloud object does not have colour for each point

Model for data model box

name of the model that the data is to be placed in. The model will be created if it does not already exist. This field must be filled in.

Fencing tab



Fence string polygon box

string to use to restrict the data being read in.

Fence mode choice box

File inside
File inside/crossing
File outside
File outside/crossing
File crossing

File inside - read file in if it is totally inside the polygon

File inside/crossing - read file in if it is totally inside, or crossing the polygon

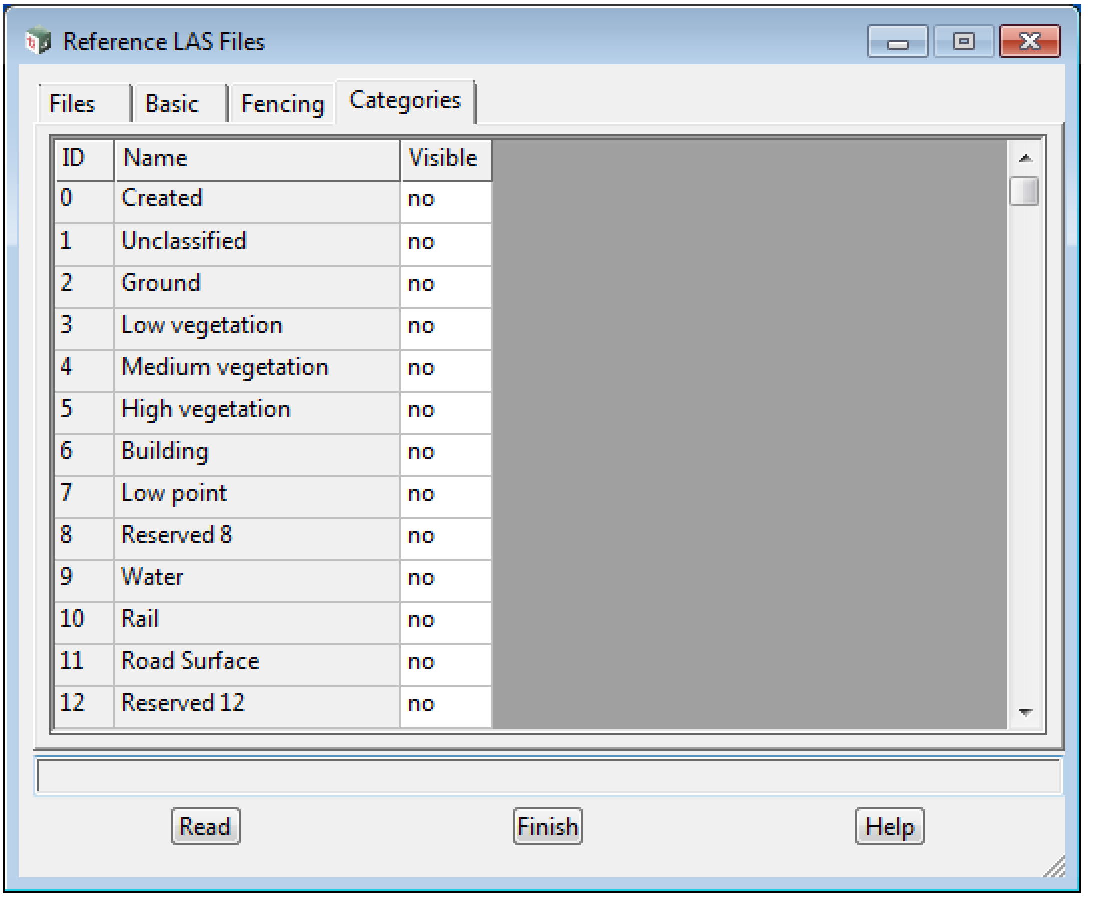
File outside - read file in if it is totally outside the polygon

File outside/crossing - read file in if it is totally outside, or crossing the polygon

File crossing - read file in if it is crossing the polygon

***Note** - only whole files are read in.*

Categories tab



ID

Name

Visible

Button at bottom

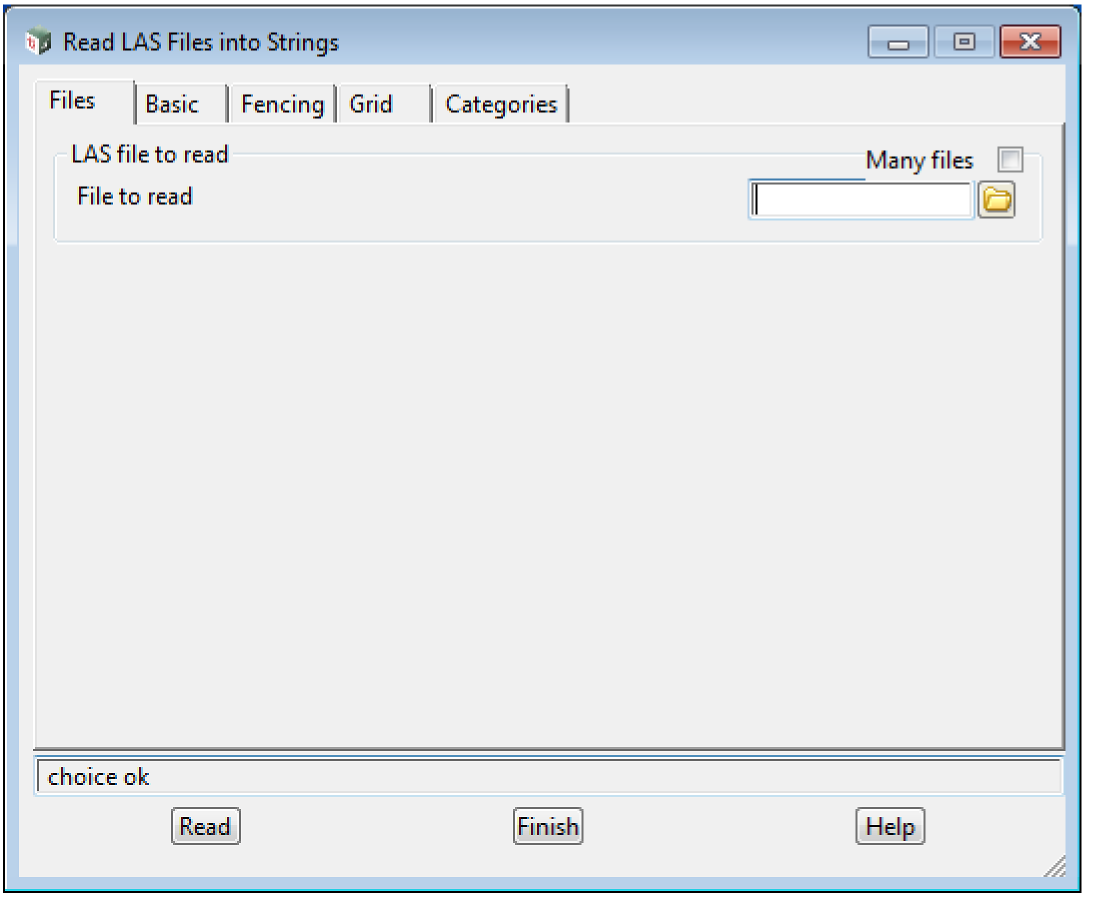
Read **button**

read the data into the model given in the model field.

8.1.16.3.3 Read LAS Files into Strings

Position of option on menu: **File I/O =>Data Input =>Point Clouds =>LAS =>Read LAS into strings**

Selecting **Read LAS into strings** brings up the **Read LAS Files into Strings** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Files tab

Many files tick box not ticked

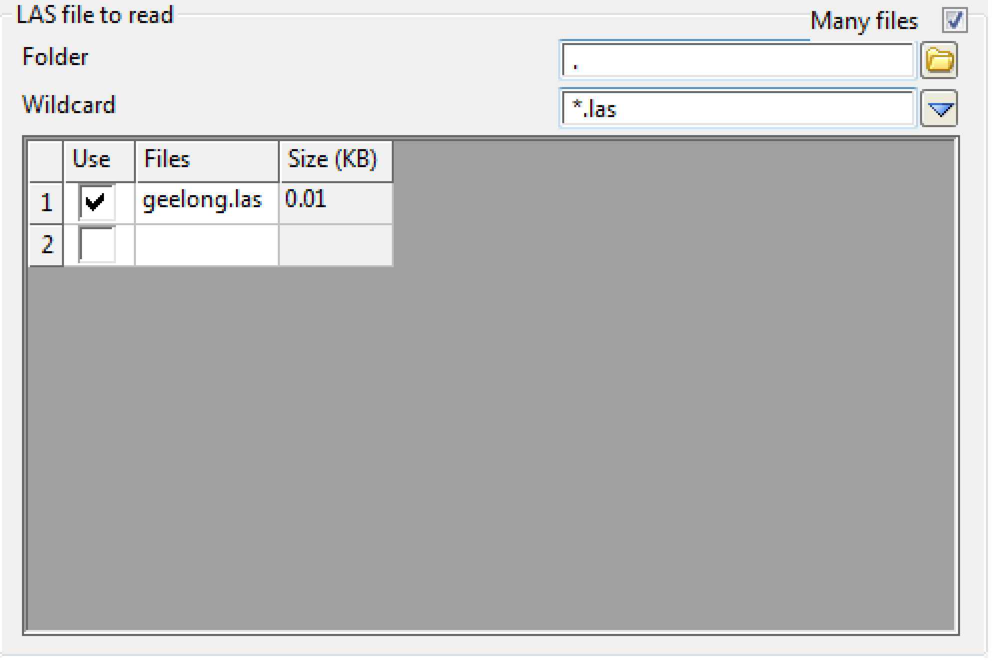
*if **ticked**, a grid to allow multiple las files to be read in is opened. A wild card is used to select all the files to be read in.*

Folder file box

folder to search for files using the **Wild card**

Wildcard input

wild card to use in search for files in the given folder



Use tick box

*if **ticked**, read in the file*

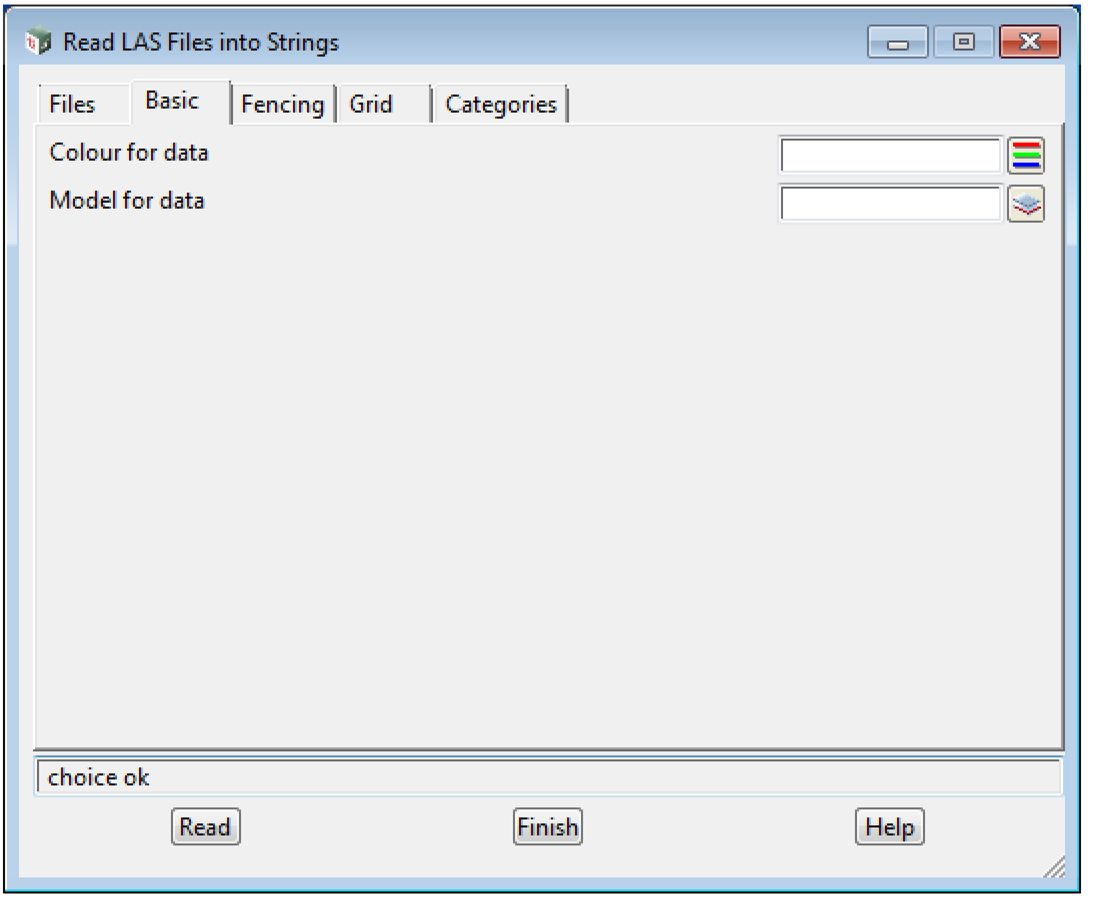
Files output

name of the file in the folder

Size output

file size

Basic tab



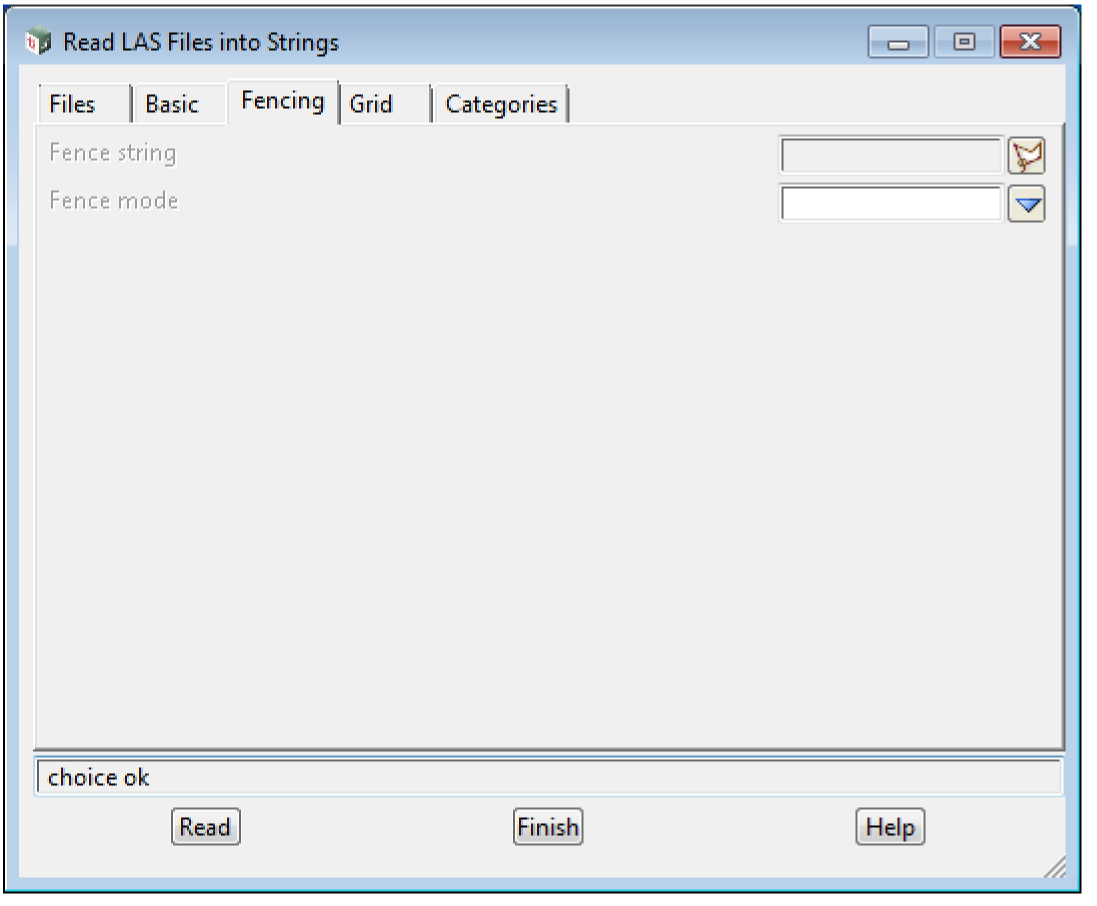
Colour for data colour box

string colour to be used if the cloud object does not have colour for each point

Model for data model box

name of the model that the data is to be placed in. The model will be created if it does not already exist. This field must be filled in.

Fencing tab



Fence string polygon box

string to use to restrict the data being read in.

Fence mode choice box
File inside
File inside/crossing
File outside
File outside/crossing
File crossing

File inside - read file in if it is totally inside the polygon

File inside/crossing - read file in if it is totally inside, or crossing the polygon

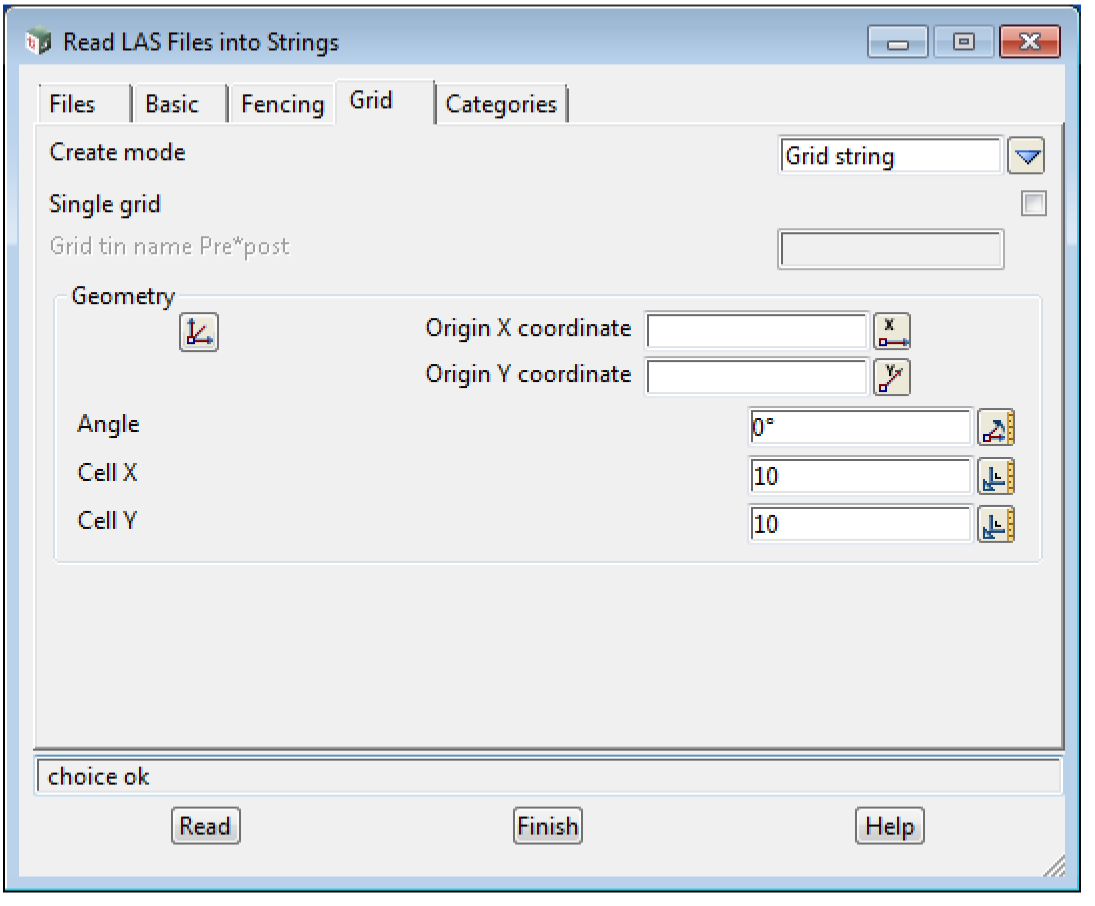
File outside - read file in if it is totally outside the polygon

File outside/crossing - read file in if it is totally outside, or crossing the polygon

File crossing - read file in if it is crossing the polygon

***Note** - only whole files are read in.*

Grid tab



Create mode choice box Grid string Grid string, Grid tin, Strings

see 14.10 Grids *for an explanation of* **Grid string** *and* **Grid tin**

Single grid tick box not ticked

if ticked, all points from all selected files are used in the creation of a single grid.

Grid tin name Pre*post input

only enabled if **Grid tin** *is selected. All grid tins created will have their names derived from the text entered with the * character beginning at 1 and incrementing for each tin created.*

Origin X/Y coordinate X/Y boxes

the Origin X and Y coordinates signify a seed point on the grid. The seed point does not have to be within the extents of the LAS points. The computed grid origin x/y will be within LAS points, and the origin x/y specified will lie on an infinite version of the resulting grid.

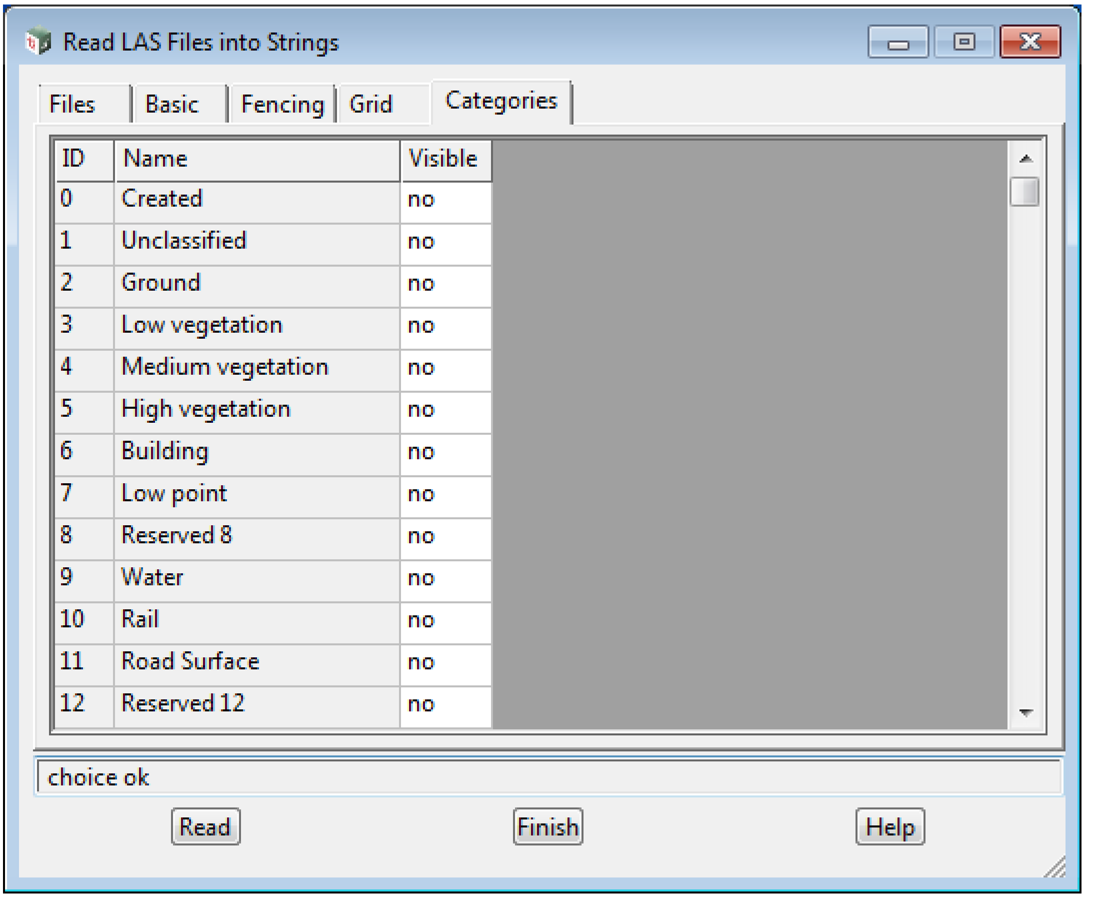
Angle measure box

the angle of the grid's local X axis

Cell X/Y measure boxes

the local X axis and Y axis spacings

Categories tab



ID

the unique number as defined by the LAS standard

Name

the description for the equivalent ID

Visible

if ***yes****, the specific category will be active for all operations on that cloud string.*

Note: If all categories are set to ***no****, then only Category 2 will be active.*

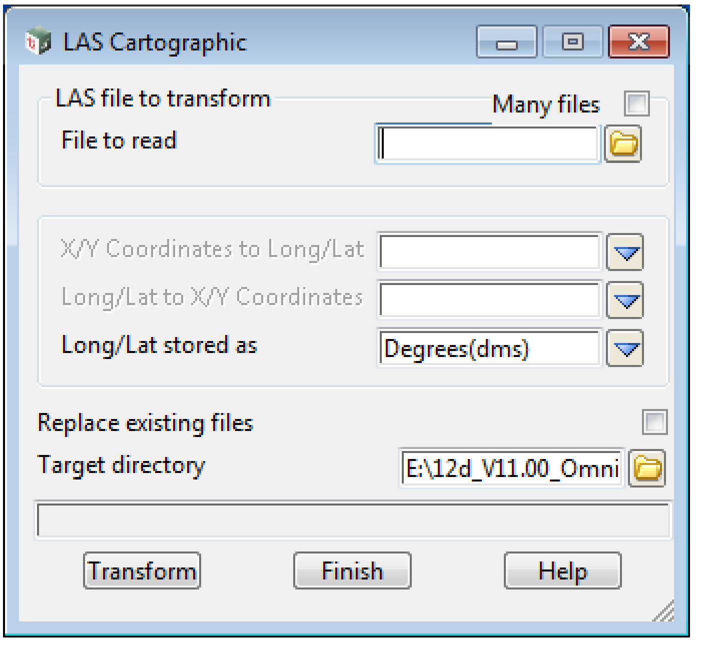
Button at bottom

Read **button**

read the data into the model given in the model field.

8.1.16.3.4 LAS Cartographic

Position of option on menu: **File I/O =>Data Input =>Point Clouds =>LAS =>Cartographic transform**

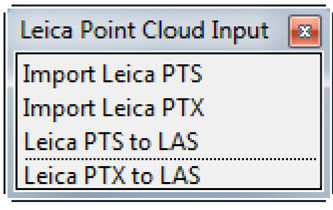


See 14.15.3 Cartographic Transform.

8.1.16.4 Leica Point Cloud Input

Position of menu: **File I/O =>Data input =>Point clouds =>Leica**

The **Leica Point Cloud Input** walk-right menu is



import Leica PTS files
import Leica PTX files
convert Leica PTS files to LAS files
convert Leica PTX files to LAS files

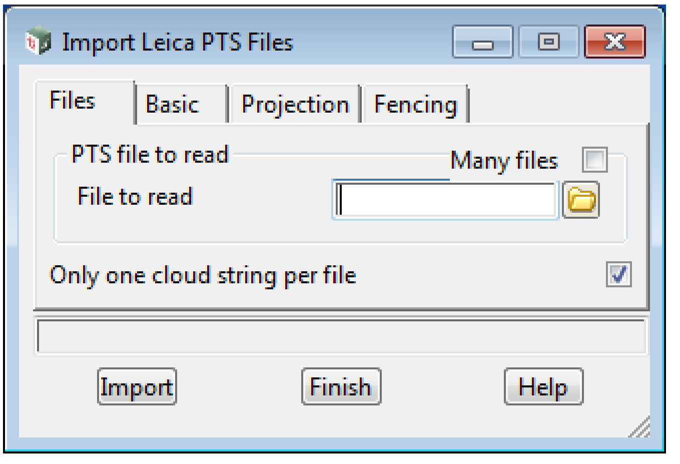
See

Import Leica PTS	8.1.16.4.1 Import Leica PTS Files
Import Leica PTX	8.1.16.4.2 Import Leica PTX Files
Leica PTS to LAS	8.1.16.4.3 Leica PTS to LAS Files
Leica PTX to LAS	8.1.16.4.4 Leica PTX to LAS Files

8.1.16.4.1 Import Leica PTS Files

Position of option on menu: **File I/O =>Data Input =>Point Clouds =>Leica =>Import Leica PTS**

Selecting **Import Leica PTS** brings up the **Import Leica PTS Files** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Files tab

Many files tick box not ticked

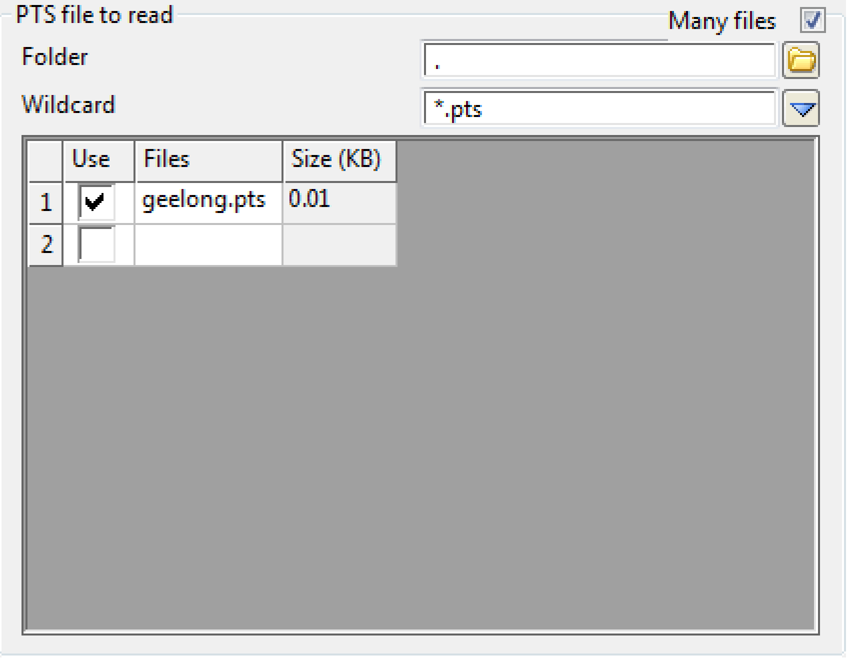
*if **ticked**, a grid to allow multiple pts files to be read in is opened. A wild card is used to select all the files to be read in.*

Folder file box

folder to search for files using the **Wild card**

Wildcard input

wild card to use in search for files in the given folder



Use tick box

*if **ticked**, read in the file*

Files output

name of the file in the folder

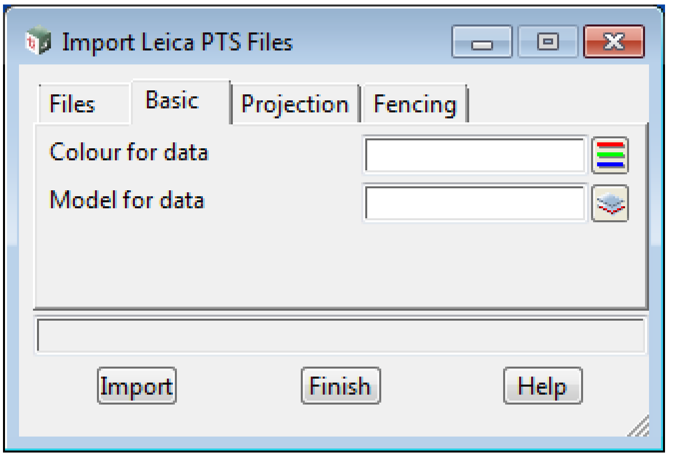
Size output

file size

Only one cloud string per file tick box ticked

if ticked, all cloud points will be combined into one 12d cloud string.

Basic tab



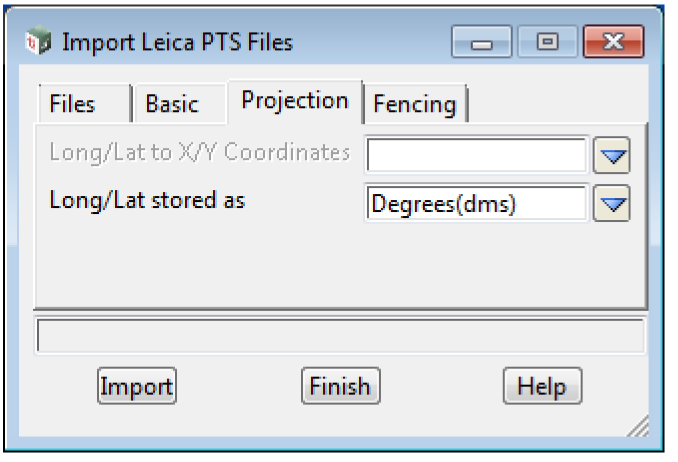
Colour for data colour box

string colour to be used if the cloud object does not have colour for each point

Model for data model box

name of the model that the data is to be placed in. The model will be created if it does not already exist. This field must be filled in.

Projection tab



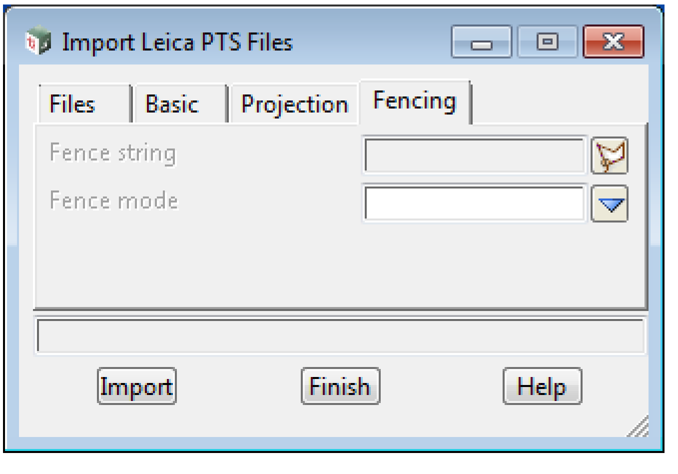
Long/Lat to X/Y Coordinates choice box available projections

*if **non-blank**, the cartographic projection to apply to the longitude-latitude values. If **blank**, the co-ordinates are not transformed from (longitude, latitude) to (x,y). Hence the initial (x,y) co-ordinates are transformed to (longitude, latitude) by the transformation given in the* **x/y co-ordinates to Long/Lat** *field and then left in (longitude, latitude). Note that in the southern hemisphere, the latitude values are **negative**.*

Long/Lat stored as choice box degrees radians
degrees
decimal degrees

format for the longitude and latitudes - either radians, degrees (in 4.17.1 HP Notation *for degrees, minutes and seconds) or decimal degrees.*

Fencing tab



Fence string polygon box

string to use to restrict the data being read in.

Fence mode choice box File inside
File inside/crossing
File outside

File outside/crossing
File crossing

File inside - read file in if it is totally inside the polygon

File inside/crossing - read file in if it is totally inside, or crossing the polygon

File outside - read file in if it is totally outside the polygon

File outside/crossing - read file in if it is totally outside, or crossing the polygon

File crossing - read file in if it is crossing the polygon

***Note** - only whole files are read in.*

Button at bottom

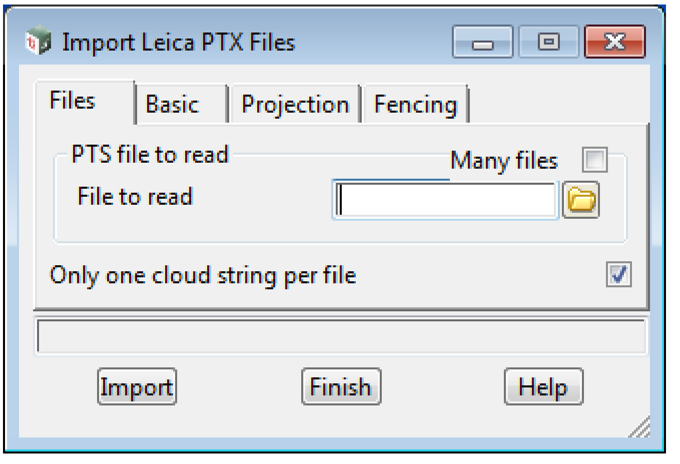
Import button

import the data into the model given in the model field.

8.1.16.4.2 Import Leica PTX Files

Position of option on menu: **File I/O =>Data Input =>Point Clouds =>Leica =>Import Leica PTX**

Selecting **Import Leica PTX** brings up the **Import Leica PTX Files** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Files tab			
Many files	tick box	not ticked	

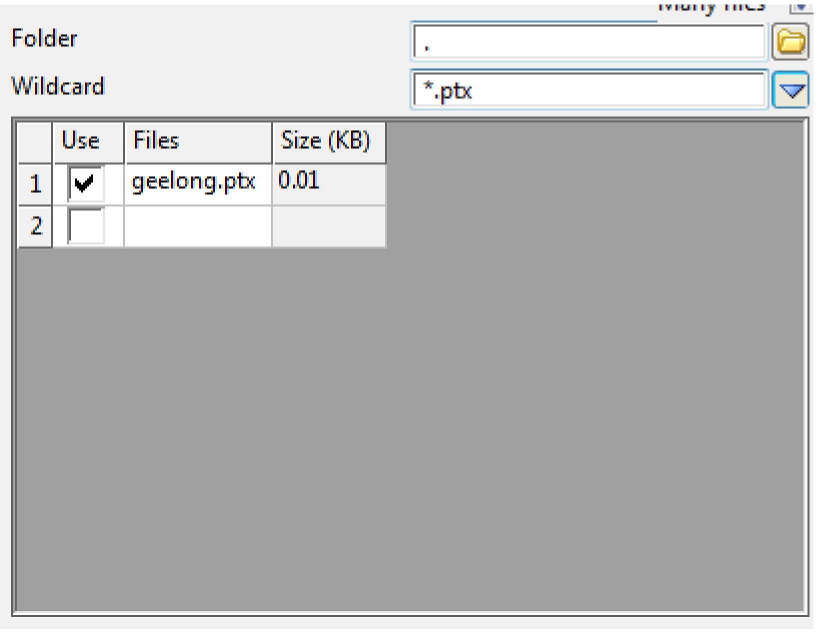
*if **ticked**, a grid to allow multiple ptx files to be read in is opened. A wild card is used to select all the files to be read in.*

Folder	file box		

folder to search for files using the **Wild card**

Wildcard	input		

wild card to use in search for files in the given folder



Use tick box

*if **ticked**, read in the file*

Files output

name of the file in the folder

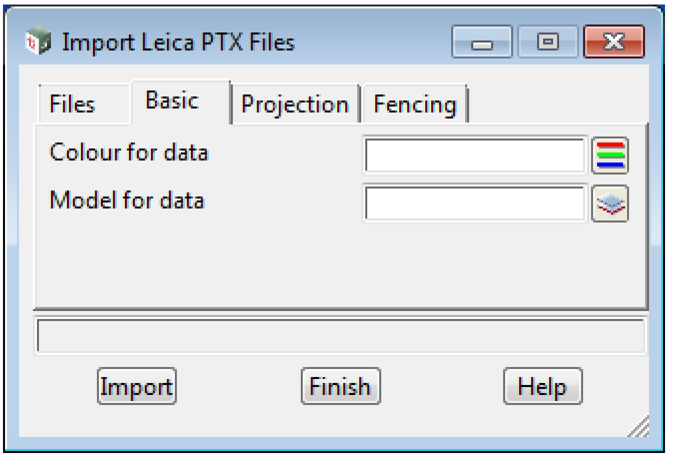
Size output

file size

Only one cloud string per file tick box ticked

if ticked, all cloud points will be combined into one 12d cloud string.

Basic tab



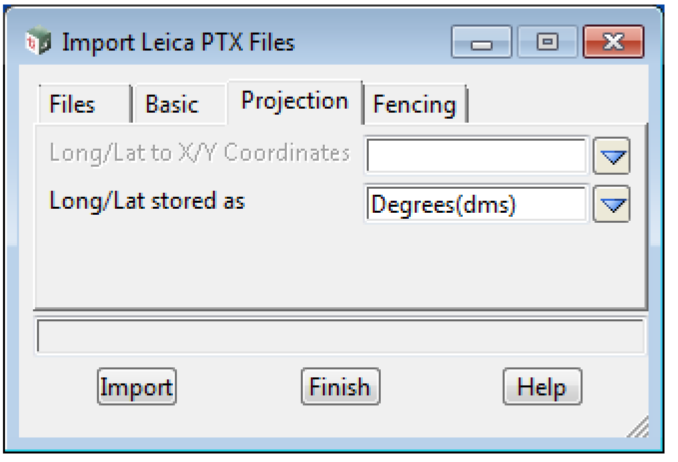
Colour for data colour box

string colour to be used if the cloud object does not have colour for each point

Model for data model box

name of the model that the data is to be placed in. The model will be created if it does not already exist. This field must be filled in.

Projection tab



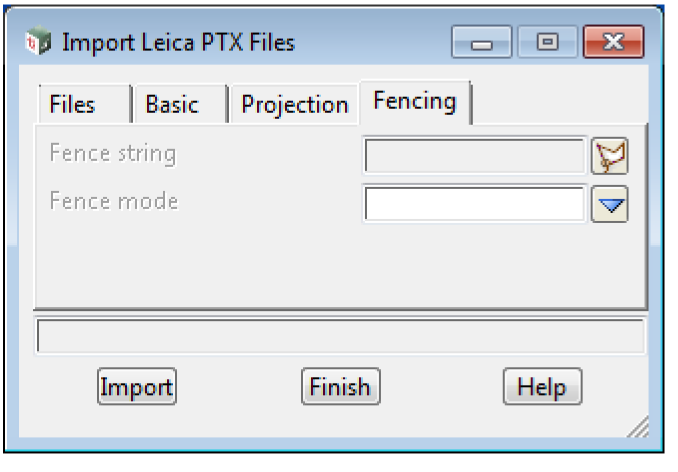
Long/Lat to X/Y Coordinates choice box available projections

if ***non-blank****, the cartographic projection to apply to the longitude-latitude values.*
If ***blank****, the co-ordinates are not transformed from (longitude, latitude) to (x,y). Hence the initial (x,y) co-ordinates are transformed to (longitude, latitude) by the transformation given in the* **x/y co-ordinates to Long/Lat** *field and then left in (longitude, latitude). Note that in the southern hemisphere, the latitude values are* ***negative****.*

Long/Lat stored as choice box degrees radians
degrees
decimal degrees

format for the longitude and latitudes - either radians, degrees (in 4.17.1 HP Notation *for degrees, minutes and seconds) or decimal degrees.*

Fencing tab



Fence string polygon box

string to use to restrict the data being read in.

Fence mode choice box File inside
File inside/crossing
File outside
File outside/crossing
File crossing

File inside - read file in if it is totally inside the polygon

File inside/crossing - read file in if it is totally inside, or crossing the polygon

File outside - read file in if it is totally outside the polygon

File outside/crossing - read file in if it is totally outside, or crossing the polygon

File crossing - read file in if it is crossing the polygon

***Note** - only whole files are read in.*

Button at bottom

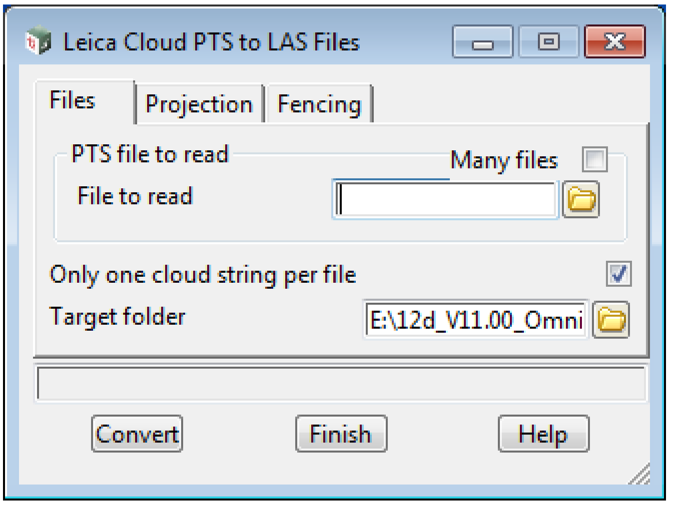
Import button

import the data into the model given in the model field.

8.1.16.4.3 Leica PTS to LAS Files

Position of option on menu: **File I/O =>Data Input =>Point Clouds =>Leica =>Leica PTS to LAS**

Selecting **Leica PTS to LAS** brings up the **Leica Cloud PTS to LAS Files** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Files tab

Many files tick box not ticked

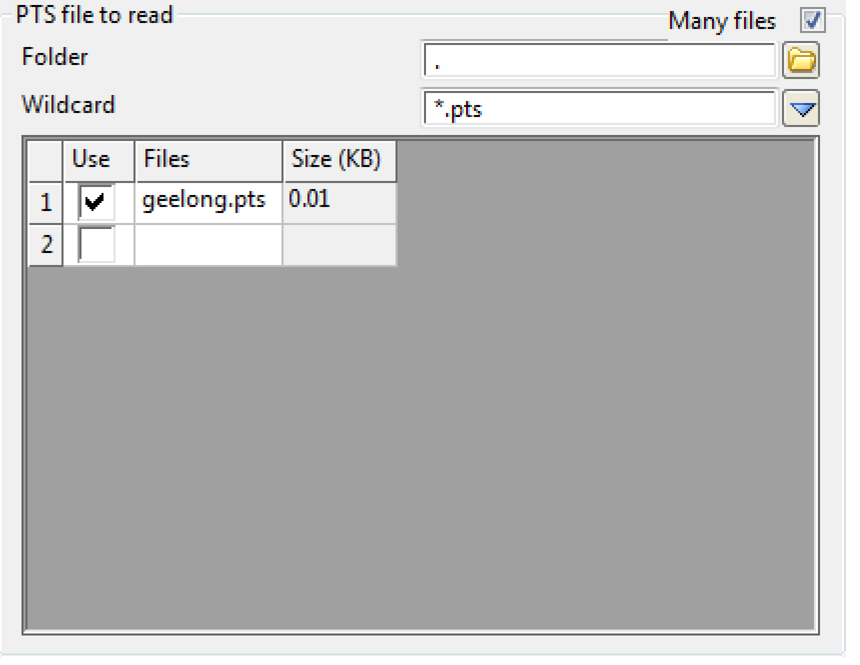
*if **ticked**, a grid to allow multiple files to be converted is opened. A wild card is used to select all the files to be read in.*

Folder file box

folder to search for files using the **Wild card**

Wildcard input

wild card to use in search for files in the given folder



Use tick box

*if **ticked**, read in the file*

Files output

name of the file in the folder

Size output

file size

Only one cloud string per file tick box ticked

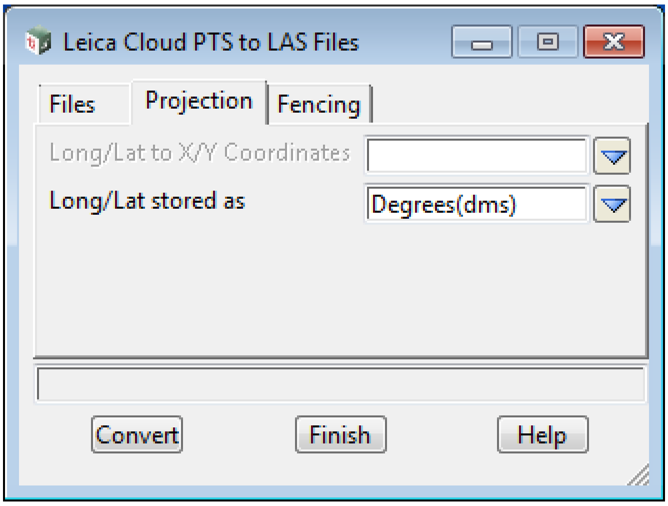
if ticked, all cloud points will be combined into one 12d cloud string.

Target folder file box

*folder where the output LAS file(s) will be created. As it is unknown how many files will be created, it is good practice to specify an empty folder for these files. For example, if the input file is **geelong.pts**, the output file(s) will be **geelong.las** OR **geelong_1.las**, **geelong_2.las**, etc.*

Note: Multiple las files will only be created if the tick box **Only one cloud string per file** *is unticked and the input file has multiple clouds.*

Projection tab



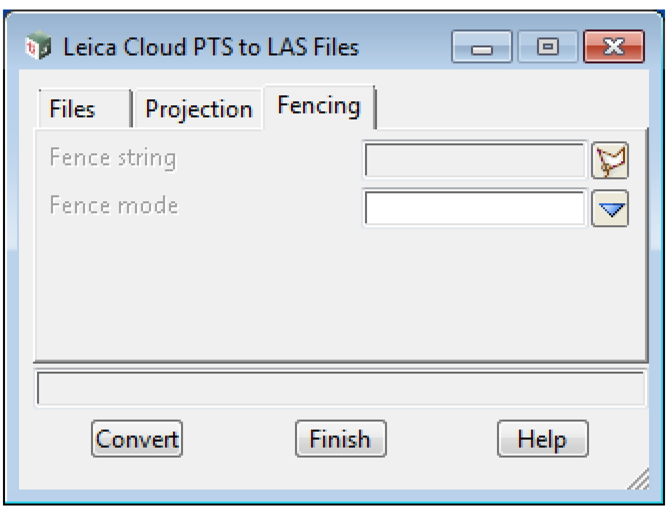
Long/Lat to X/Y Coordinates choice box available projections

*if **non-blank**, the cartographic projection to apply to the longitude-latitude values.*
*If **blank**, the co-ordinates are not transformed from (longitude, latitude) to (x,y). Hence the initial (x,y) co-ordinates are transformed to (longitude, latitude) by the transformation given in the* **x/y co-ordinates to Long/Lat** *field and then left in (longitude, latitude). Note that in the southern hemisphere, the latitude values are **negative**.*

Long/Lat stored as choice box degrees radians
degrees
decimal degrees

format for the longitude and latitudes - either radians, degrees (in 4.17.1 HP Notation *for degrees, minutes and seconds) or decimal degrees.*

Fencing tab



Fence string polygon box

string to use to restrict the data being read in.

Fence mode choice box File inside
File inside/crossing
File outside
File outside/crossing
File crossing

File inside - read file in if it is totally inside the polygon

File inside/crossing - read file in if it is totally inside, or crossing the polygon

File outside - read file in if it is totally outside the polygon

File outside/crossing - read file in if it is totally outside, or crossing the polygon

File crossing - read file in if it is crossing the polygon

***Note** - only whole files are read in.*

Button at bottom

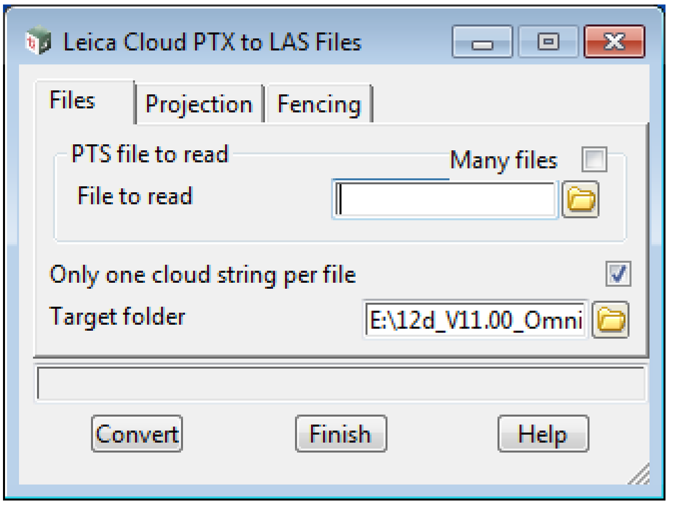
Convert button

convert the data into the model given in the model field.

8.1.16.4.4 Leica PTX to LAS Files

Position of option on menu: **File I/O =>Data Input =>Point Clouds =>Leica =>Leica PTX to LAS**

Selecting **Leica PTX to LAS** brings up the **Leica Cloud PTX to LAS Files** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Files tab

Many files tick box not ticked

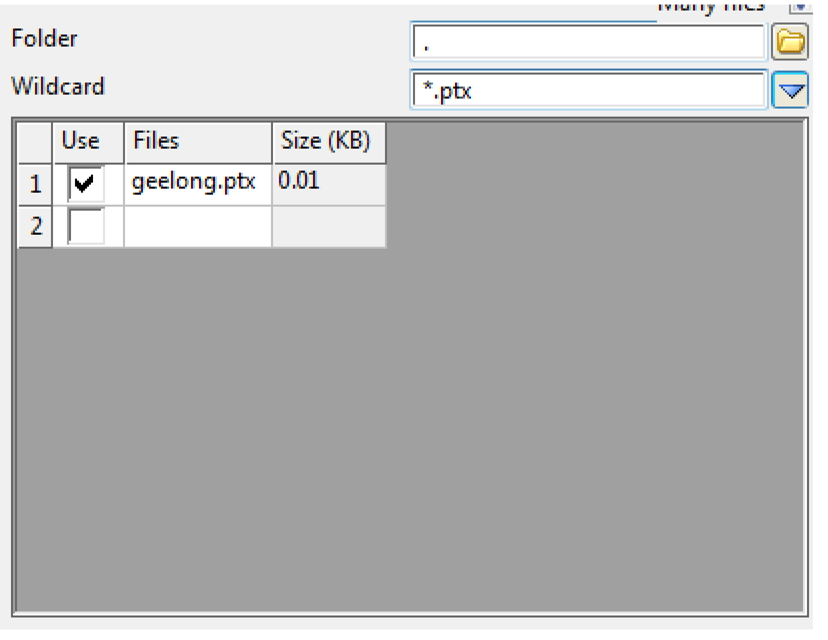
*if **ticked**, a grid to allow multiple files to be converted is opened. A wild card is used to select all the files to be read in.*

Folder file box

folder to search for files using the **Wild card**

Wildcard input

wild card to use in search for files in the given folder



Use tick box

*if **ticked**, read in the file*

Files output

name of the file in the folder

Size output

file size

Only one cloud string per file tick box ticked

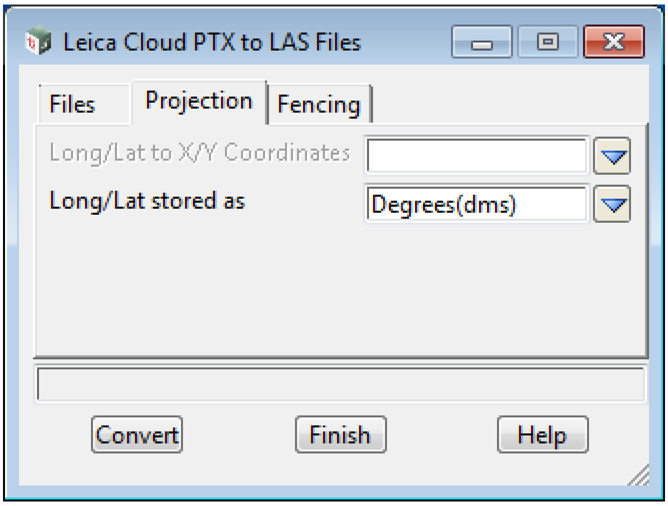
if ticked, all cloud points will be combined into one 12d cloud string.

Target folder file box

*folder where the output LAS file(s) will be created. As it is unknown how many files will be created, it is good practice to specify an empty folder for these files. For example, if the input file is **geelong.ptx**, the output file(s) will be **geelong.las** OR **geelong_1.las**, **geelong_2.las**, etc.*

Note: Multiple las files will only be created if the tick box **Only one cloud string per file** *is unticked and the input file has multiple clouds.*

Projection tab



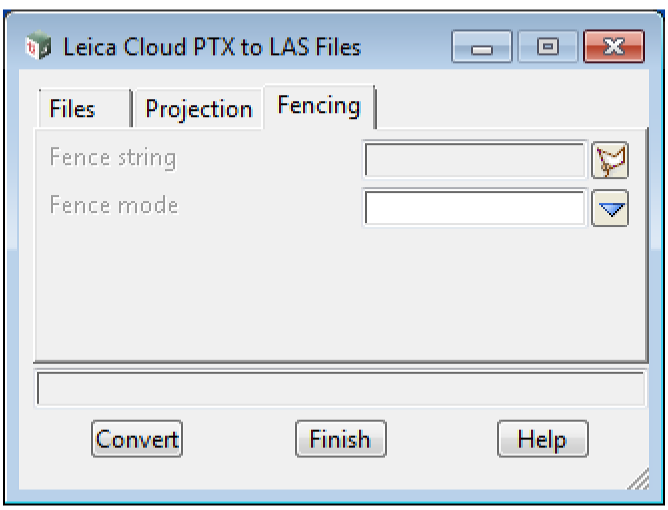
Long/Lat to X/Y Coordinates	choice box		available projections

*if **non-blank**, the cartographic projection to apply to the longitude-latitude values.*
*If **blank**, the co-ordinates are not transformed from (longitude, latitude) to (x,y). Hence the initial (x,y) co-ordinates are transformed to (longitude, latitude) by the transformation given in the* **x/y co-ordinates to Long/Lat** *field and then left in (longitude, latitude). Note that in the southern hemisphere, the latitude values are **negative**.*

Long/Lat stored as	choice box	degrees	radians degrees decimal degrees

format for the longitude and latitudes - either radians, degrees (in 4.17.1 HP Notation *for degrees, minutes and seconds) or decimal degrees.*

Fencing tab



Fence string	polygon box	

string to use to restrict the data being read in.

Fence mode	choice box	File inside File inside/crossing File outside File outside/crossing File crossing

File inside - read file in if it is totally inside the polygon

File inside/crossing - read file in if it is totally inside, or crossing the polygon

File outside - read file in if it is totally outside the polygon

File outside/crossing - read file in if it is totally outside, or crossing the polygon

File crossing - read file in if it is crossing the polygon

***Note** - only whole files are read in.*

Button at bottom

Convert button

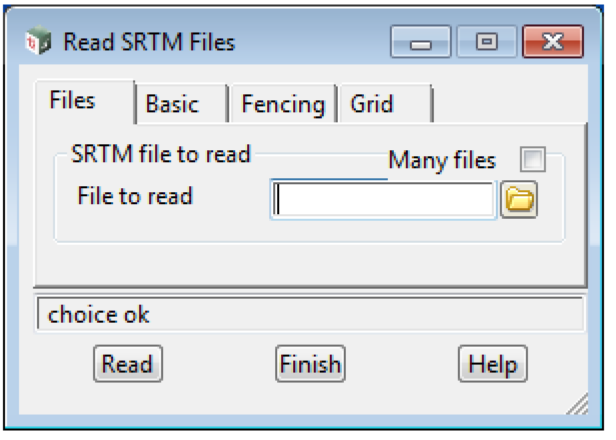
convert the data into the model given in the model field.

8.1.16.5 Read SRTM Files

Position of option on menu: **File I/O =>Data Input =>Point clouds =>SRTM**

SRTM refers to the Shuttle Radar Topography Mission, an international research effort which obtained a high-resolution digital topographic database of Earth.

Selecting **SRTM** brings up the **Read SRTM Files** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Files tab

Many files	tick box	not ticked	

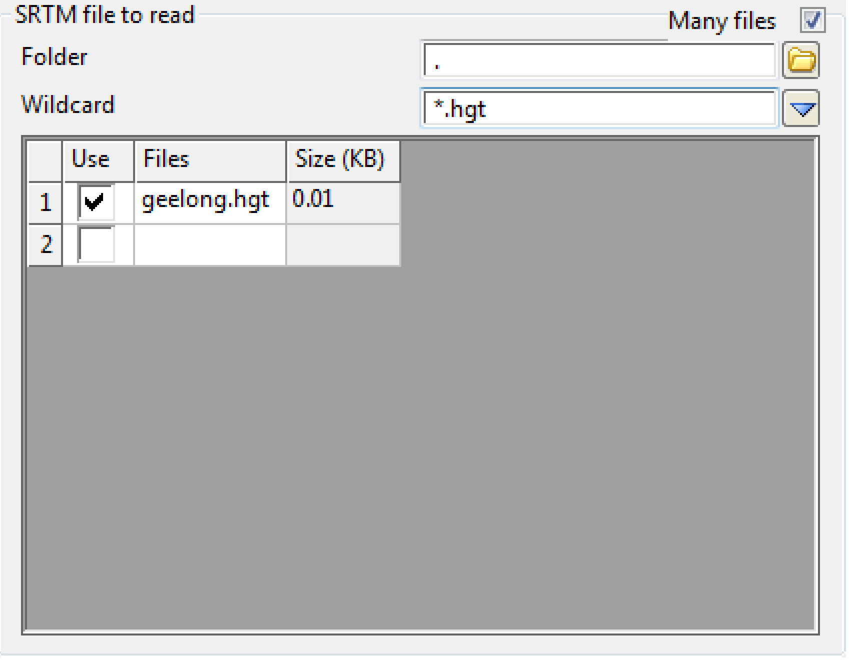
*if **ticked**, a grid to allow multiple hgt files to be read in is opened. A wild card is used to select all the files to be read in.*

Folder	file box		

folder to search for files using the **Wild card**

Wildcard	input		

wild card to use in search for files in the given folder



Use tick box

*if **ticked**, read in the file*

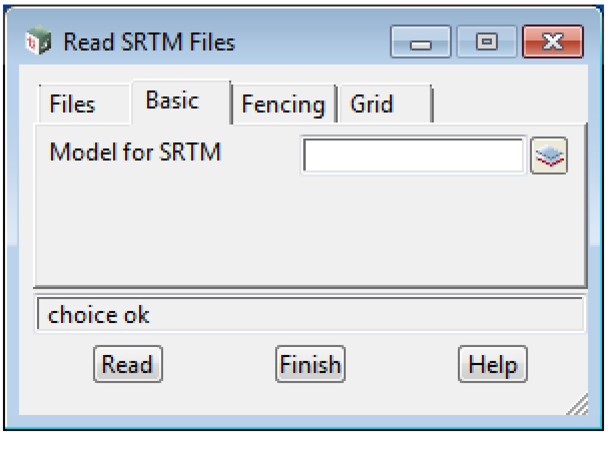
Files output

name of the file in the folder

Size output

file size

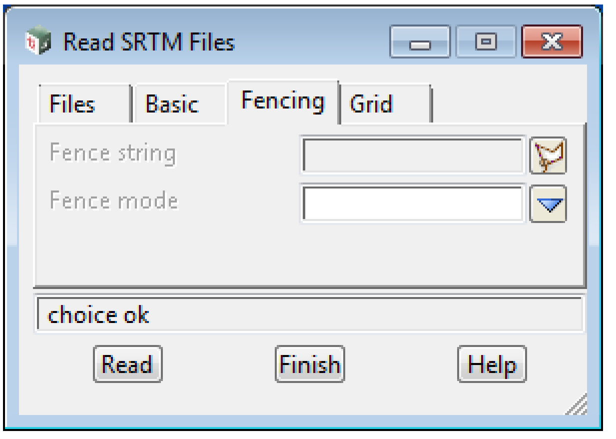
Basic tab



Model for SRTM model box

name of the model that the data is to be placed in. The model will be created if it does not already exist. This field must be filled in.

Fencing tab



Fence string polygon box

string to use to restrict the data being read in.

Fence mode choice box
File inside
File inside/crossing
File outside
File outside/crossing
File crossing

File inside - read file in if it is totally inside the polygon

File inside/crossing - read file in if it is totally inside, or crossing the polygon

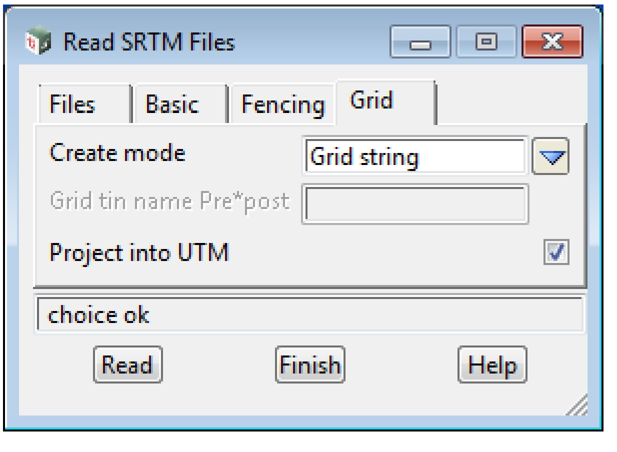
File outside - read file in if it is totally outside the polygon

File outside/crossing - read file in if it is totally outside, or crossing the polygon

File crossing - read file in if it is crossing the polygon

***Note** - only whole files are read in.*

Grid tab



Create mode choice box Grid string Grid string, Grid tin, Strings

see 14.10 Grids *for an explanation of* **Grid string** *and* **Grid tin**

Grid tin name Pre*post input

only enabled if **Grid tin** *is selected. All grid tins created will have their names derived from the text entered with the * character beginning at 1 and incrementing for each tin created.*

Project into UTM tick box ticked

if ticked, the project will be uploaded into the Universal Transverse Mercator coordinate system.

Button at bottom

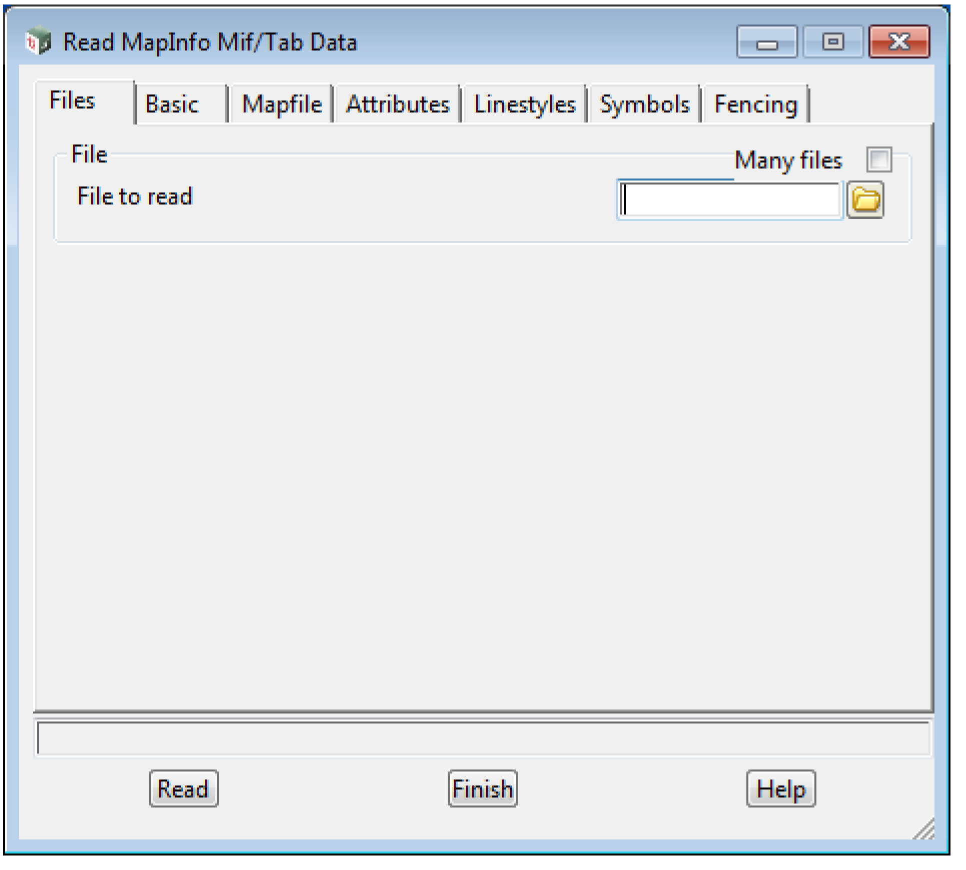
Read button

read the data into the model given in the model field.

8.1.17 Mapinfo Input

Position of option on menu: **File I/O =>Data input =>Mapinfo MID/MIF**

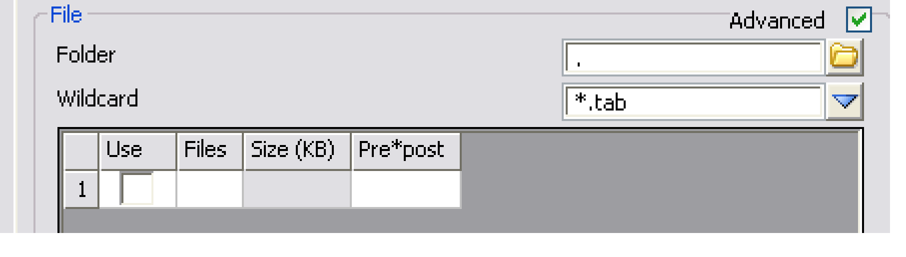
Selecting **MapInfo MID/MIF** brings up the **Read MapInfo Mif/Tab Data** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Advanced	tick box		

*if **ticked**, a grid to allow multiple files to be read in, is opened. A wild card is used to select all the files to be read in.*



Folder	folder box

folder to search for files using the **Wild card**

Wildcard	input

wild card to use in search for files in the given folder

Use	tick box

*if **ticked**, read in the file*

Files output

name of the file in the folder

Size output

file size

Pre*post text input

*if **non blank**, pre*post text to use for the models in this file (see* 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

*If **blank**, use the pre*post text from the* **Pre*postfix for models** *panel field.*

***Note** - if a non-blank value for **Pre*post** is given in the column for a file then the **Pre*postfix for models** is ignored.*

File to read file box *.tab or *.mif files

name of the MapInfo file.

If no attribute is mapped to model, then the file name (minus the .tab or .mif) is used as the model for the data.

Basic tab

Default colour for black colour box white available colours

colour to use in 12d for black in MapInfo

Map file tab

Map file file box *.mf and *.mapfile files

*if **non-blank**, the name of the 12d map file to be used for all strings read in.*

*If **blank**, no map file is used*

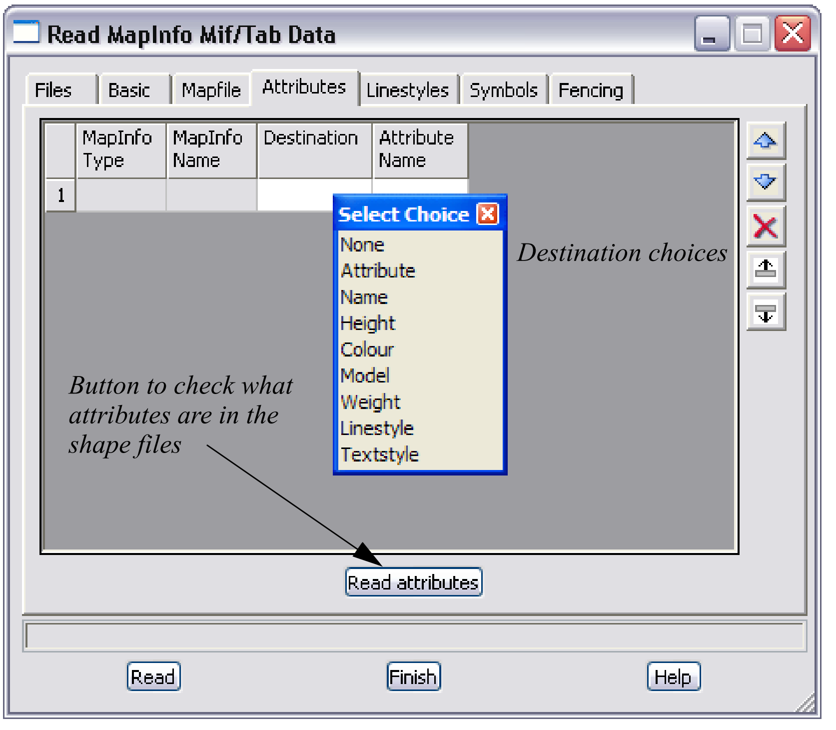
When using a map file, the string name is used as the entity-name for matching with the keys in the map file. See the section 8.8.1 Create/Edit a Map File *for information about 12d map files.*

Pre*postfix for models pre*postfix box

if non-blank, a prefix and a postfix to be applied to the model names used in the Map File.

Go to the section 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

Attributes tab



Read attributes button

Click to check what attributes are present in the MapInfo files and any MapInfo attributes found are listed in the **MapInfo Type** *and* **MapInfo Name** *columns in the grid.*

Destination Grid column

type of 12d attribute to map the MapInfo attribute to:

none - don't use the attribute - the attribute is ignored
attribute - use as 12d attribute
name - use as 12d string name
height - use as 12d vertex height

colour - use as 12d string colour

model - use as model name

weight - use as string weight

linestyle - use as 12d linestyle

textstyle - use as 12d textstyle

***Note** - if no MapInfo attribute is mapped to Model, then the file name (minus the .tab or .mif) is used as the model for the data.*

Attribute name Grid column

if non blank and the MapInfo attribute is being sent to a 12d attribute, then this is the 12d attribute name

Linestyles tab

MapInfo linestyle Grid column

MapInfo linestyle name

12d linestyle Grid column

12d linestyle to map the MapInfo linestyle to

Symbols tab

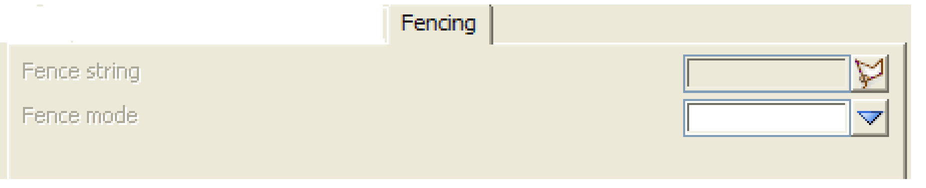
MapInfo symbol Grid column

MapInfo symbol name

12d symbol Grid column

12d symbol to map the MapInfo symbol to

Fencing tab



Fence string polygon box

string to use to restrict the data being read in.

Fence mode choice box

String inside
String inside/crossing
String outside
String outside/crossing
String crossing

String inside - read string in if it is totally inside the polygon

String inside/crossing - read string in if it is totally inside, or crossing the polygon

String outside - read string in if it is totally outside the polygon

String outside/crossing - read string in if it is totally outside, or crossing the polygon

String crossing - string in if it is crossing the polygon

***Note** - only whole strings are read in.*

Read button

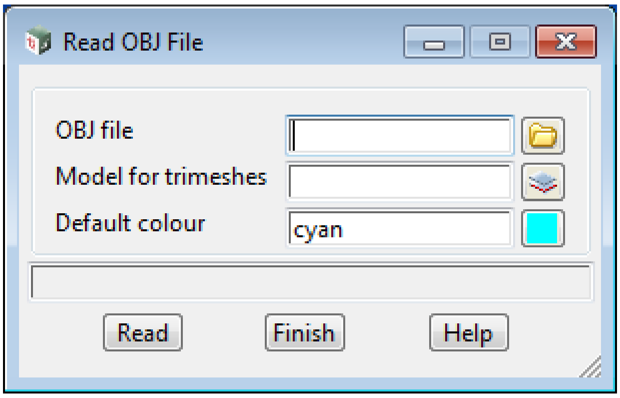
read the data in.

8.1.18 Wavefront OBJ Input

Position of option on menu: **File I/O =>Data input =>OBJ**
Position of option on menu: **Strings =>Trimesh =>Read OBJ file**

The **OBJ** option reads in a Wavefront OBJ file and creates trimeshes.

Selecting **OBJ** brings up the **Read OBJ File** panel.



The fields and buttons used in this panel have the following functions.

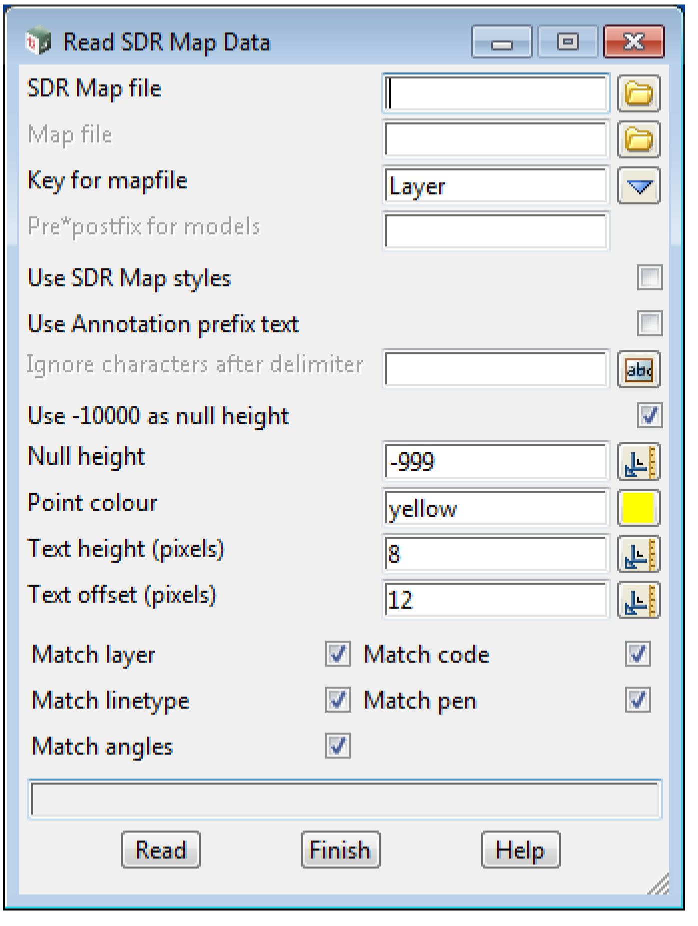
Field Description	Type	Defaults	Pop-Up
Obj file	file box		*.OBJ files
name of the OBJ file to read in			
Model for trimeshes	model box		available models
model to put the created trimeshes in.			
Default colour	colour box		available colours
Read	button		
read the obj data in			

8.1.19 SDR Map Input

Position of option on menu: **File I/O =>Data input =>SDR Map**

The **SDR Map** input option is designed to read in files in the SDR Map ASCII format.

On selecting the **SDR Map** option, the **Read SDR Map Data** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
SDR Map file	file box		*.txt

name of the SDR Map text file to be read in.

Map file	map file box		*.mf and *.mapfile files

if non-blank, the name of the map file to be used for all data read in.

See the section 8.8.1 Create/Edit a Map File *for information about 12d map files.*

Key for map file	choice box	code	code, layer code/layer, layer/code

*if **code**, the SDRmap code is used as the key for the map file.*
*If **layer**, the SDRmap layer is used as the key for the map file.*

*If **code/layer**, the SDRmap code/layer is used as the key for the map file.*

*If **layer/code**, the SDRmap layer/code is used as the key for the map file.*

Pre*postfix for models	pre*postfix box		

if non-blank, a prefix and a postfix to be applied to the model names used in the Map File.

Go to the section 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

Use SDR Map styles tick box

if ticked,

Use Annotation prefix text tick box

if ticked,

Ignore characters after delimiter input

if non-blank, all characters on the line after the given delimiter will be ignored.

Use -10000 as null height tick box tick

if ticked, any z value of -10000 is converted to a null height in ***12d Model****.*

Null height input -999

if ***non-blank,*** *any SDR Map z values equalling this value are converted to null values in* ***12d Model****.*

Point colour colour box yellow available colours

colour for SDR Map points that are not part of strings.

Text height (pixels) input

height of the text in pixels

Text offset (pixels) input

distance in pixels to offset the text from its (x,y) placement position.

Match flags tick box

SDRMap data consists of individual lines and arcs. When reading SDRMap data, it undergoes head to tail processing and the match flags specify what SDRMap data can be joined.

Match layer/code/linetype/angles/pen tick box

if ***ticked,*** *then any SDRMap lines and arcs must have the same layer/code/linetype/angle/pen before they can be joined in the head to tail process.*

Read button

read the data in

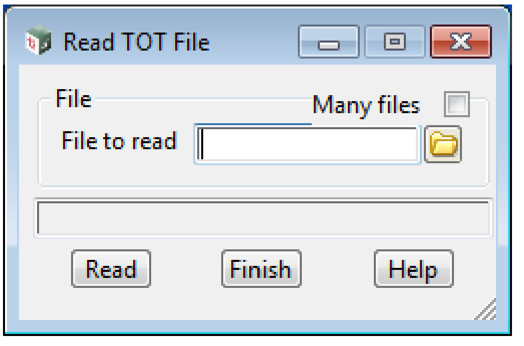
8.1.20 TOT Input

Position of option on menu: **File I/O =>Data input =>TOT**

The **TOT** option attempts to read in TOT files.

Note: TOT is a format used by Model Maker, a software program from South Africa.

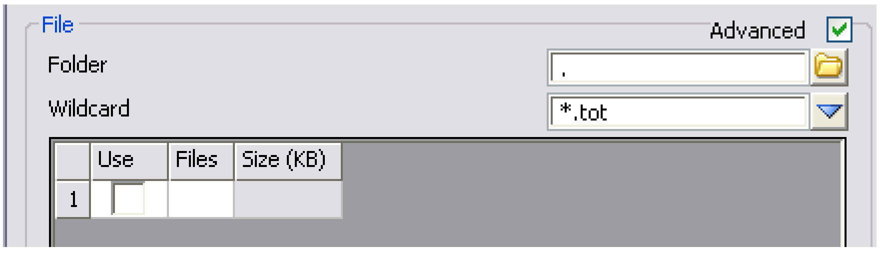
Selecting **TOT** brings up the **Read TOT File** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Advanced	tick box		

*if **ticked**, a grid to allow multiple TOT files to be read in, is opened. A wild card is used to select all the files to be read in.*



Field Description	Type	Defaults	Pop-Up
Folder	folder box		

folder to search for files using the **Wild card**

Field Description	Type	Defaults	Pop-Up
Wildcard	input		

wild card to use in search for files in the given folder

Field Description	Type	Defaults	Pop-Up
Use	tick box		

*if **ticked**, read in the file*

Field Description	Type	Defaults	Pop-Up
Files	output		

name of the file in the folder

Field Description	Type	Defaults	Pop-Up
Size	output		

file size

Field Description	Type	Defaults	Pop-Up
File to read	file box		*.TOT files

name of the TOT file to read in

Read button

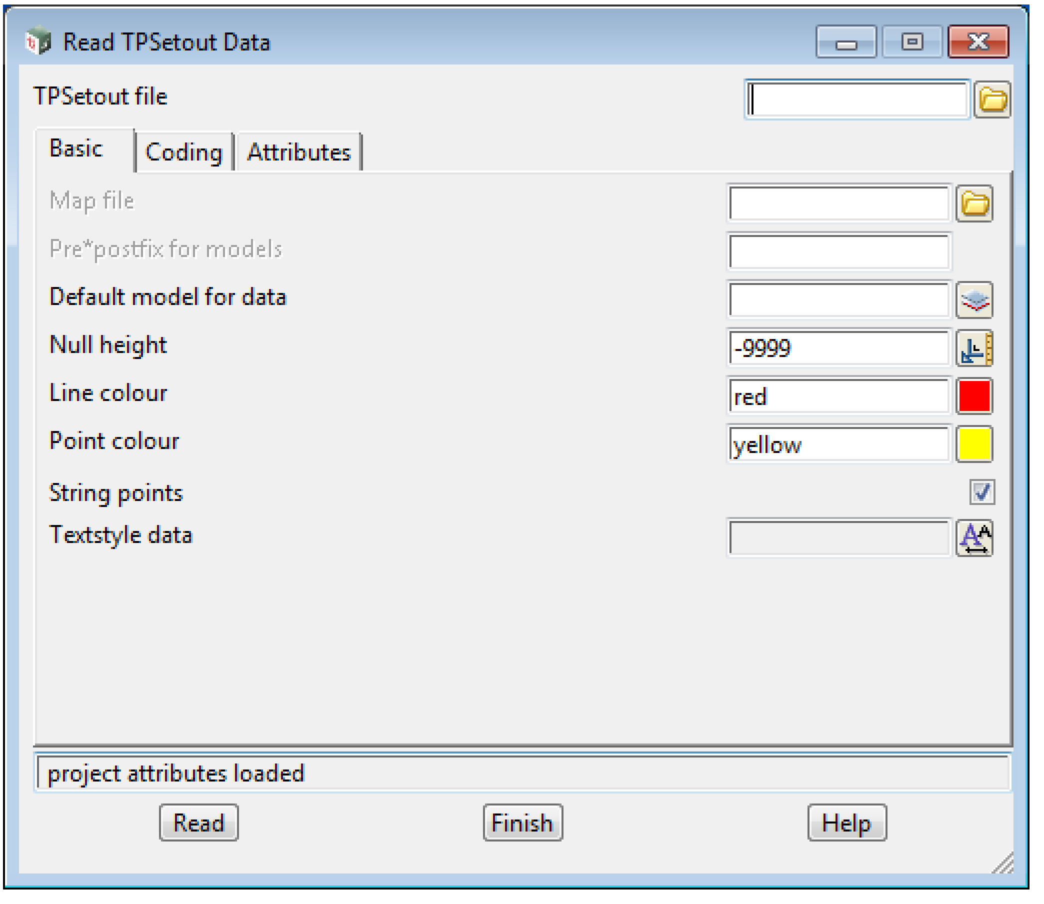
read the data in

8.1.21 TP Setout Input

Position of option on menu: **File I/O =>Data input =>TP Setout**

The **TP Setout** input option is designed to read in .pta files from *TP Setout*

Selecting **TP Setout** brings up the **Read TP Setout Data** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
TP Setout file	file box		*.pta

name of the TP Setout .pta file to be read in.

Basic tab

Map file	map file box		*.mf files

if non-blank, the name of the map file to be used for all strings read in. If blank, no map file is used. The TP Setout code is used as the entity-name to match against the key in the map file.

See the section 8.8.1 Create/Edit a Map File *for information about 12d map files.*

Pre*postfix for models	pre*postfix box		

if non-blank, a prefix and a postfix to be applied to the model names used in the Map File.

Go to the section 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

Default model for data	model box		available models

name of the model that any unmapped data is placed in. The model will be created if it does not already *exist. This field must be filled in.*

Null height input -9999

*if **non-blank,** any TP Setout z values equalling this value are taken as null values.*

Line colour colour box red available colours

colour for TP Setout strings.

Point colour colour box yellow available colours

colour for TP Setout points that are not part of strings.

String points tick box tick

if ticked, points with the same description are joined together.
If not ticked, points with the same description are not joined together.

Textstyle data textstyle data box available textstyle datas

textstyle data used for any text labels

Coding tab

Point ids in column 5 tick box

*if **ticked**, there are points ids in column 5 of the file.*

Feature code input

number of characters in the feature code

String number input

number of characters in the string number

Point ids input

number of characters in the point ids

Attributes tab

Name

Type

Data

Read button

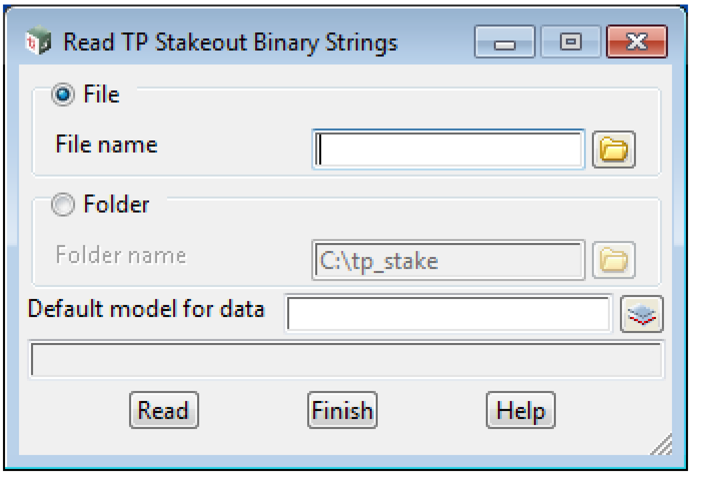
read the data in.

8.1.22 TP Stakeout Strings Input

Position of option on menu: **File I/O =>Data input =>TP Stakeout strings**

The **TP Stakeout strings** input option is designed to read in .3db files (binary strings) from TP Stakeout.

Selecting **TP Stakeout strings** brings up the **Read TP Stakeout Binary Strings** panel.



The fields and buttons used in this panel have the following functions.

Field Description | Type | Defaults | Pop-Up

File radio button

*if **on** then an individual binary string file is to be read in.*

File name file box *.3db files

if non-blank, the name of the binary string file to read in.

Folder radio button

*if **on** then the all the binary string files in the folder are to be read in.*

Folder name folder box

if non-blank, all the binary string files in this folder are read in.

Default model for data input available models

name of the model to read the binary strings into.

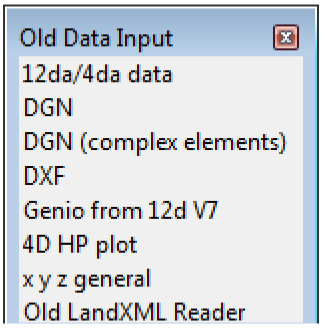
Read button

read the data in.

8.1.23 Old Inputs

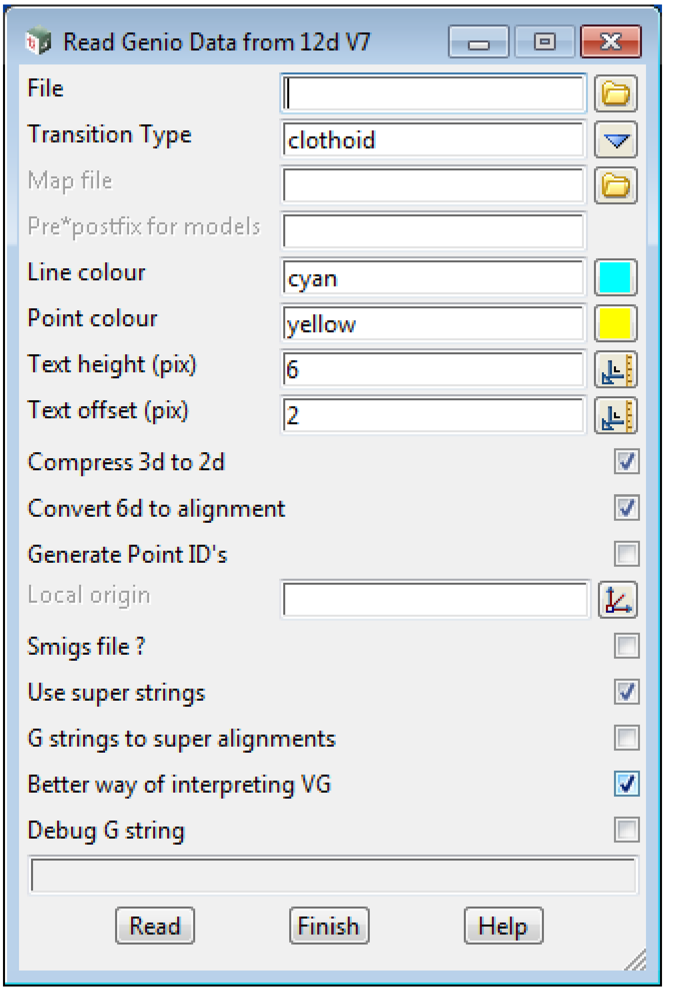
Position of menu: **File I/O =>Data input =>Old**

The **Old** menu contains superseded options. The **Old** walk-right menu is



8.1.23.1 Genio from 12d V7

This section of documentation is a work in progress and will be updated in subsequent releases.



8.1.23.2 12D HP Plot File

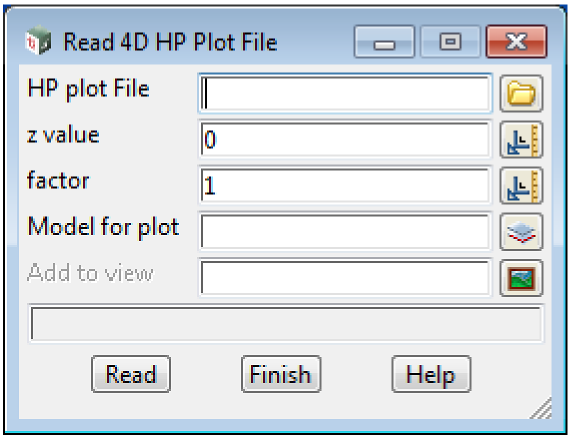
Position of option on menu: **File I/O =>Data input =>Old =>4D HP plot**

12d Model has options to create plots for HPGL compatible plotters. This option reads back into 12d Model any plots created by 12d Model using HPGL plot options.

A plot file is only a two dimensional file defined in millimetres. When read back into 12d Model, the units are automatically multiplied by 1000.

In the option, the user can supply a further factor to multiply the x and y coordinates by and also a z-value which is used as a z coordinate for all the lines in the plot file.

On selecting the 12d HP plot option, the **read 4D HP plot file** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
HP plot file	input		*.hp files

name of the 12d HP plot file to be read in

z value	input	0.0	

z coordinate to use for the lines read in from the plot file

Factor	input	1.0	

the default units used when reading in a plot file are multiplied by 1000. They are then multiplied by this value.

Model for plot	input		available models

name of the model that the plot file is to be placed in. The model will be created if it does not already exist. This field must be filled in.

Add to view	input		available views

if a view name is entered, then the model will be automatically added to the view. This field can be blank.

Read	button		

read the plot file into the model given in the model field.

8.1.23.3 DXF Input

Position of option on menu: **File I/O =>Data input =>Old =>DXF**

DXF input is a separate chargeable option

The **DXF** input option is designed to read most autocad DXF files (up to Version 12), including binary DXF which was introduced in AutoCAD Release 10 as a means of addressing the problems of large file sizes, slow processing and limited accuracy that occur when using the ASCII DXF format. ***12d Model*** will automatically sense whether the input file is binary or ASCII.

The standard Autocad colours can be mapped to the equivalent 12d Model colours or just mapped on a one-to-one basis to 12d Model colour numbers.

Each DXF item has an associated layer. By default, 12d Model creates models of the same name as the layers (or with an additional user supplied prefix) and the DXF items placed in them.

However, this can be over-written using a standard **12d map file** where the key is matched against Autocad **layers** rather than string names.

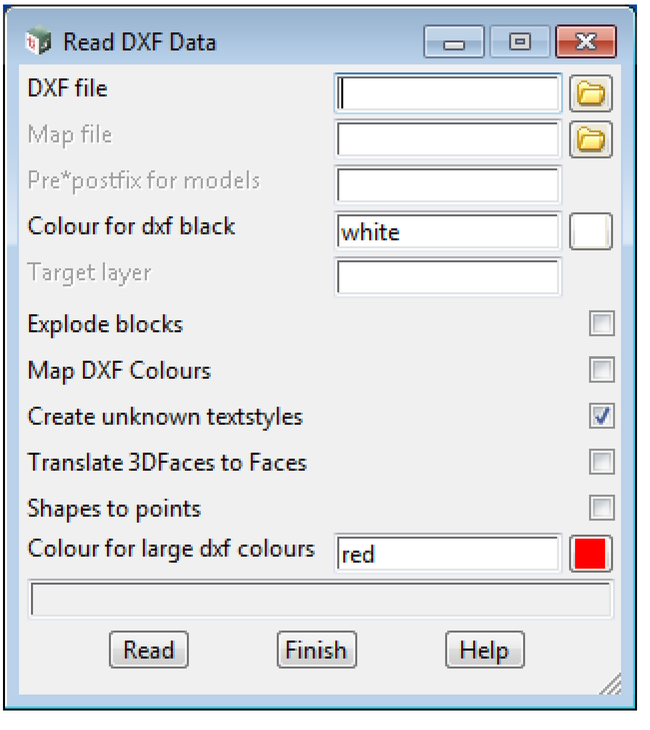
See the section 8.8.1 Create/Edit a Map File for information about 12d map files.

Autocad blocks are recognised and either a point with the block name is created or the blocks are expanded into their components, each time they are referenced in the DXF file.

Bulges in polylines can only be interpreted correctly when the polyline has a constant z-value. For this case, a 12d Model polyline string is created from the DXF polyline.

Finally, DXF POINT entities of the same layer and colour are concatenated into one point string as they are read in.

On selecting the **DXF** option, the **read dxf data** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
DXF File	input		*.dxf files

name of the DXF file to be read in

Map file input *.mf files

if non-blank, the name of the map file to be used for all DXF layers read in. The DXF layer is the entity-name for matching against the key in the map file.
If blank, no map file is used.

See the section 8.8.1 Create/Edit a Map File *for information about 12d map files.*

Prefix for models input

if non-blank, all 12d Model model names created by the reader will be prefixed by this name.

Colour for dxf black input white available colours

if the DXF colour of an item is black, then the colour in the **colour for black** *field is used for the item in 12d Model.*

Target layer input

if non-blank, only autocad items in the layer with the name given in the **target layer** *field will be read in.*

Explode blocks tick no tick

if ***ticked****, autocad blocks are exploded in 12d Model.*
If ***not ticked****, blocks are not exploded and a point string is placed at the position of the block.*

Map DXF colours tick no tick

if ***ticked****, the first seven DXF colours are mapped to the corresponding default 12d Model colours.*
If ***no ticked****, the nth DXF colour is mapped to the nth 12d Model colour.*

Create unknown textstyles tick tick

if ***ticked****, then if a textstyle in the DXF file is not already defined in* ***12d Model****, then a new* ***12d Model*** *textstyle of the same name is created.*
If ***no ticked****, then the* ***12d Model*** *textstyle "1" is used for any unknown DXF textstyles.*

Translate 3DFaces to faces tick no tick

if ***ticked****, DXF faces are read in as* ***12d Model*** *face strings.*

Colour for large dxf colours input red available colours

if the DXF colour is greater than the largest colour number defined in the 12d Model colour map, then the colour in the **colour for large dxf colours** *field is used for the DXF colour.*

Read button

read the data in.

8.1.23.4 User X Y Z and Attributes Input - Pre V9

Position of option on menu: **File I/O =>Data input =>Old => x y z general**

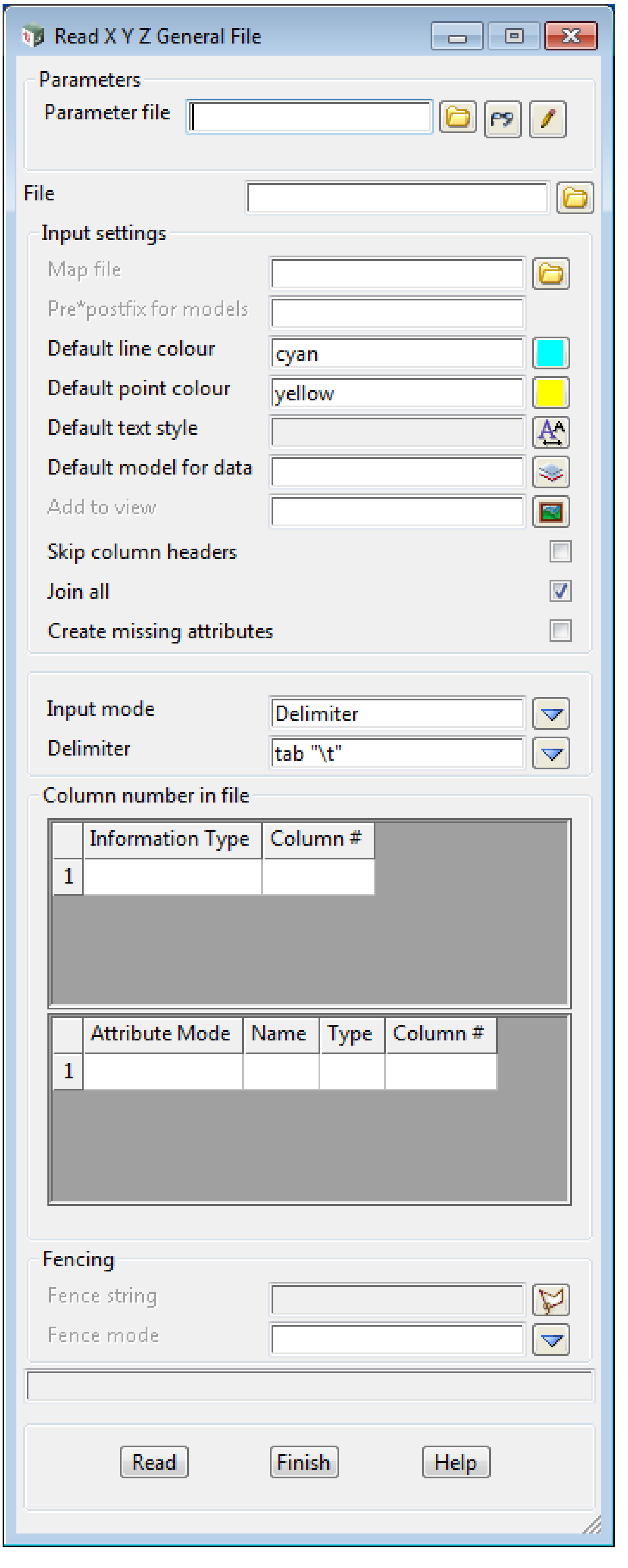
Note - this option was replaced in ***12d Model*** 9.

On selecting the **read x,y,z general** option, the **Read x y z s General File** panel is displayed.

This option reads data in one line at a time with the values separated by a delimiter (tab, space, semi-colon or comma) or the data on each line can be in fixed width columns.

In either case, the user specifies the order that the x, y, z, string name, point number and attributes appear in the file. If desired, only an x and y value needs to be read in.

The set-ups for defining all the positions of all the data in the file can be written out to a file (.xyf) for re-use.


Read X Y Z General File
Parameters
Parameter file
File
Input settings
Map file
Pre*postfix for models
Default line colour
cyan
Default point colour
yellow
Default text style
Default model for data
Add to view
Skip column headers
Join all
Create missing attributes
Input mode
Delimiter
Delimiter
tab "\t"
Column number in file
Information Type
Column #
1
Attribute Mode
Name
Type
Column #
1
Fencing
Fence string
Fence mode
Read
Finish
Help

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Parameters section			
Parameter file	input		*.xyf files

name of the file containing the settings for how the data is positioned in the input file.

Read icon	button		

read the parameter file in.

Write icon	button		

write the setting in the panel out to a parameter file.

Input settings section

File	input		*.dat files

name of the data file to be read in

Map file	file box		*.mf files

if non-blank, the name of the map file to be used for all strings read in. If blank, no map file is used. The string name is used as the entity-name for matching with the keys in the map file.

See the section 8.8.1 Create/Edit a Map File *for information about 12d map files.*

Pre*postfix for models	pre*postfix box		

if non-blank, a prefix and a postfix to be applied to the model names used in the Map File.

Go to the section 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

Default line colour	input	default colour	available colours

colour used for line-strings (if no colour is defined in the file)

Default point colour	input	default pt colour	available colours

colour used for the crosses in point-strings (if no colour is defined in the file)

Default text style	input		available textdatas

textdata for the point ids and other text

Default model for data	input		available models

name of the model that the data is to be placed in. The model will be created if it does not already exist. This field must be filled in.

Add to view	input		available views

if a view name is entered, then the model will be automatically added to the view. This field can be blank.

Skip column headers	tick box		

if ticked, the first line of the file is skipped.

Join all	tick box		

*if **ticked**, all vertices with the same string names are joined together regardless of where they are in the file. The order of the vertices is the order they occur in the file.*

*If **not ticked** then any time a string name changes in the file, a new string is created. So if the same string name occurs but separated by a different string name, then more than one string of that same name will be created. The order of the vertices is the order they occur in the file.*

Input mode	choice box	delimiter	delimiter, fixed width

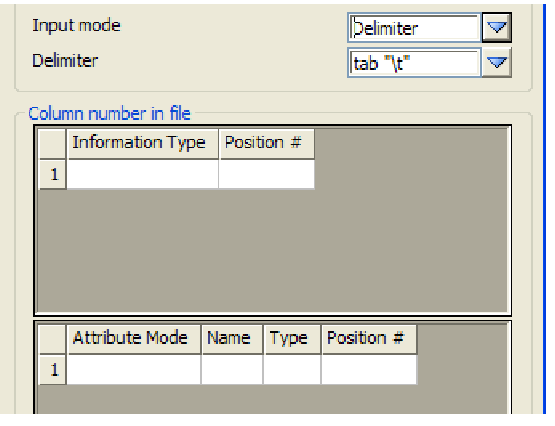
*if **delimiter**, the type of delimiter and the columns for the x, y and optionally z, name and point number are given.*

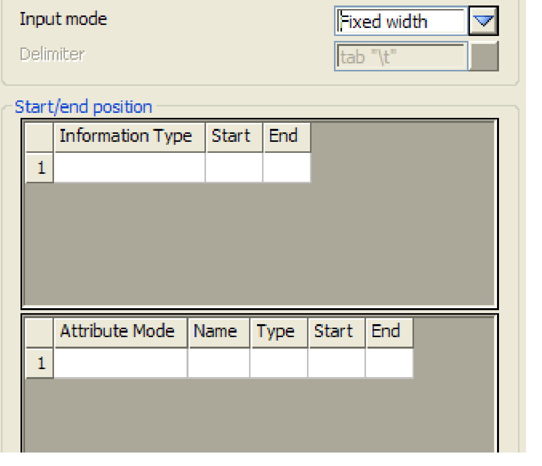
Delimiter	choice box	tab \f	one space, tab \t, semi colon, comma many spaces

*if **fixed width**, the start and end column positions are given for x, y and optionally z, name and point number.*

Column number/Start end position section

information to read in (x,y,z, attributes etc.) and its position in the input file.





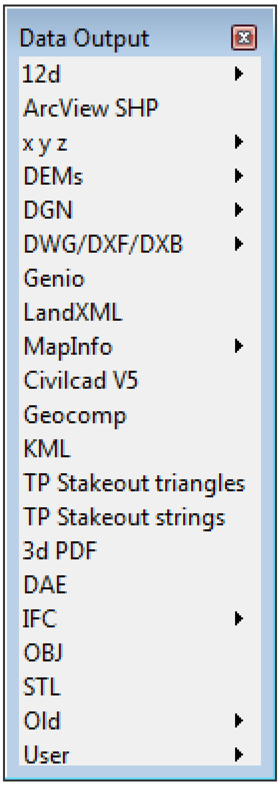
8.2 Data Output

Position of menu: **File I/O =>Data output**

12d Model provides output options so that data in a model or on a view can be written out to a disk file. this may be to allow data to be transferred to other programs for further processing or simply to get a readable list of data.

The default *Output null value* is described in the section 8.2.1 Output Null Value

The **Data Output** walk-right menu containing these options is



bring up the 12d Output panel
ArcView SHP file
x y z data
write out DEM data for a tin
DGN binary
DXF/DWG file
MX (Moss) genio data
LandXML data
MapInfo data
CivilCAD V5.0 file
Geocomp pts and str files
KML output
TP Stakeout triangles
TP Stakeout binary strings
3d PDF file
Collada DAE format
IFC output
Wavefront Obj's
Stereolithography STL file (3D printers)
Old formats
User Data Output menu

For the option *12d*	8.2.2 12d Output
ArcView SHP	8.2.4 ArcView SHP Output
x y z	8.1.5 Input X Y Z Text Files
DEMs	16.14.7 DEMs
DGN	8.1.9 Input DGN Binary Files
DWG/DXF/DBX	8.2.7 Output DWG/DFX/DXB Files
Genio	8.2.8 Genio Output
LandXML	8.2.9 LandXML Output
MapInfo	8.2.10 Output MapInfo MID/MIF Files
Civilcad V5.0	8.2.11 CivilCAD
Geocomp	8.2.12 Geocomp
KML	8.2.15 KML Output
TP Stakeout triangles	8.2.13 TP Stakeout Triangles Output
TP Stakeout strings	8.2.14 TP Stakeout Strings Output
3d PDF	8.2.3 Write 3D PDF
DAE	8.2.16 Export DAE
IFC	8.2.19 IFC Output
OBJ	8.2.17 Export OBJ
STL	8.2.18 Export STL

8.2.1 Output Null Value

When writing out data for other systems, it is usually not appropriate to write out the internal 12d Model null height (- 9.9e29).

Consequently as data is written out, any 12d Model null heights are replaced by the value given by the **I/O null height** parameter.

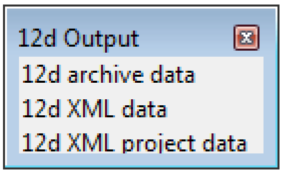
The **I/O null height** is set in **Default Settings** tab of the **Defaults** panel in the option **Utilities=>Defaults.**

Return to 8.2 Data Output

8.2.2 12d Output

Position of menu: **File I/O =>Data output =>12d**

Selecting **12d** brings up this walk-right menu



write out data in ***12d Model*** Archive format
write out XML data
write out XML project data

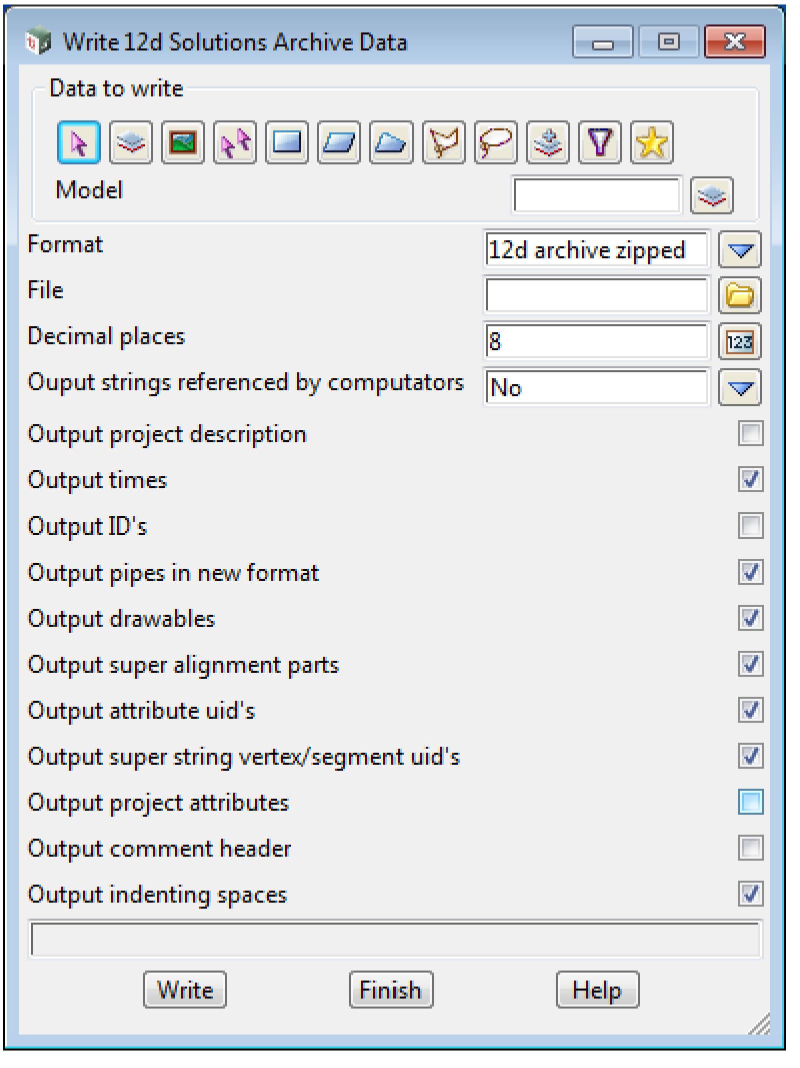
For the option *12d archive data*, 8.2.2.1 12d Archive Output
12d XML data 8.2.2.2 12d XML Data
12d XML project data 8.2.2.3 12d XML Project Data

8.2.2.1 12d Archive Output

Position of option on menu: **File =>Data output =>12d =>12d archive data**

The ***12d Archive*** format is a special format defined by 12d Solutions to allow data to be easily transferred from other programs into 12d Solutions software such as ***12d Model***. The ***12d Archive*** format is given in the Appendix.

Selecting the **12d archive data** brings up the **Write 12d Solutions Archive Data** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

Data to search - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data is to be written out to a file.

Format	choice	12d archive zipped	12d archived zipped, 12d archive

use 12d archive zipped (12daz) or 12d archive format (12a). The only difference is that the 12az file is a zipped 12da file with the same name stem.

File file box *.12da or *.12daz

name of the file for the information to be written out to in 12d Archive format. If the file already exists, then you are asked to Append, Replace or Cancel.

Decimal places integer box 8

the number of decimal places used when writing the data out.

Output string references by computators choice box No Yes, No, De-reference

*if **Yes**, any strings referenced by computators by strings in the data source that are not in the data source, are also written out.*

*If **No**, any strings referenced by computators by strings in the data source that are not in the data source, are NOT written out.*

*If **De-reference**, any super strings in the data source with computators are de-referenced. That is, the computators are replaced by internal elements of the string.*

Output project description tick box

if ticked, the project description is written out as comments at the top of the file.

Output times tick box

if ticked, write out the creation times etc. for the objects.
If not ticked, don't write out the creation times etc. for the objects.

Output ID's tick box

if ticked, write out the object IDs.
If not ticked, don't write out the object IDs.

Output pipes in new format tick box

if ticked,
If not ticked, don't write out.

Output drawables tick box

if ticked, write out the internal super alignment labelling as text.

Output super alignment parts tick box

if ticked, write out all the construction details for super alignments.
If not ticked, don't write out the construction details. Just write out the HG and VG segments.

Output attribute uid's tick box

if ticked, write out the attribute uids.

Output super string vertex/segment uid's tick box

if ticked, write out the any vertex and segment UIDs.

Output project attributes tick box

if ticked, write out the project attributes.

Output comment header tick box

if ticked, write out information as comments at the top of the 12da file.

Output indenting spaces tick box

if ticked, leave out spaces for indenting lines in the file. This will reduce the size of the text 12da file.

Write button

write out the data specified by the Data source.

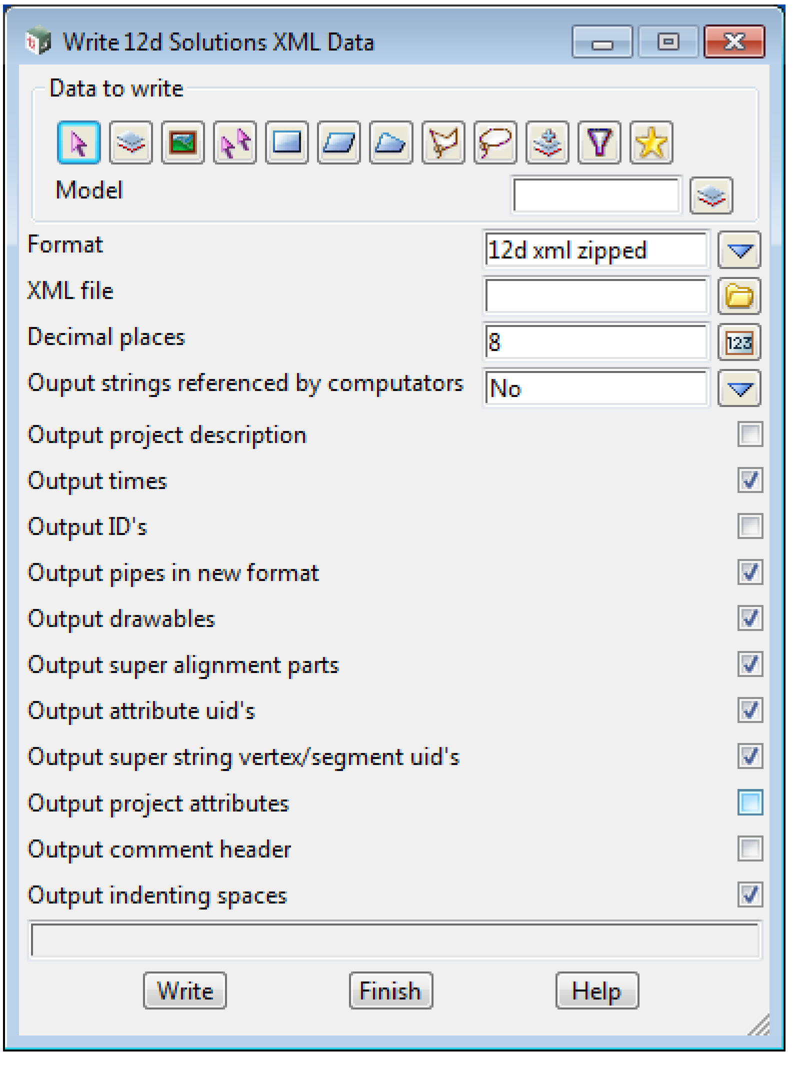
8.2.2.2 12d XML Data

Position of option on menu: **File I/O =>Data output =>12d =>12d XML data**

The ***12d XML*** format is a special format defined by 12d Solutions to allow data to be easily transferred from other programs into 12d Solutions software such as 12d Model. The **12d XML** format is given in 35 12d XML File Format.

Note - 12d XML contains the same information as a 12da file but it is a XML format.

Selecting **12d XML data** brings up the **Write 12d Solutions XML Data** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

Data to search - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be written out to a file.

Format	choice	12d xml zipped	12d xml zipped, 12d xml

use 12d xml zipped (12dxmlz) or 12d xml format (12xml). The only difference is that the 12xmlz file is a zipped 12dxml file with the same name stem.

XML file file box

name of the file for the information to be written out to in 12d XML format. If the file already exists, then you are asked to Append, Replace or Cancel.

Decimal places integer box 8

the number of decimal places used when writing out the data.

Output string references by computators choice box No Yes, No
, De-reference

*if **Yes**, any strings referenced by computators by strings in the data source that are not in the data source, are also written out.*

*If **No**, any strings referenced by computators by strings in the data source that are not in the data source, are NOT written out.*

*If **De-reference**, any super strings in the data source with computators are de-referenced. That is, the computators are replaced by internal elements of the string.*

Output project description tick box

if ticked, the project description is written out as comments at the top of the file.

Output times tick box

if ticked, write out the creation times, etc. for the objects.
If not ticked, don't write out the creation times etc. for the objects.

Output ID's tick box

if ticked, write out the object IDs.
If not ticked, don't write out the object IDs.

Output pipes in new format tick box

if ticked,
If not ticked, don't write out.

Output drawables tick box

if ticked, write out the internal super alignment labelling as text.

Output super alignment parts tick box

if ticked, write out all the construction details for super alignments.
If not ticked, don't write out the construction details. Just write out the HG and VG segments.

Output attribute uid's tick box

if ticked, write out the attribute uids.

Output super string vertex/segment uid's tick box

if ticked, write out the any vertex and segment UIDs.

Output project attributes tick box

if ticked, write out the project attributes.

Output comment header tick box

if ticked, write out information as comments at the top of the 12da file.

Output indenting spaces tick box

if ticked, leave out spaces for indenting lines in the file. This will reduce the size of the text 12d XML file.

Write button

write out the data specified by the Data source.

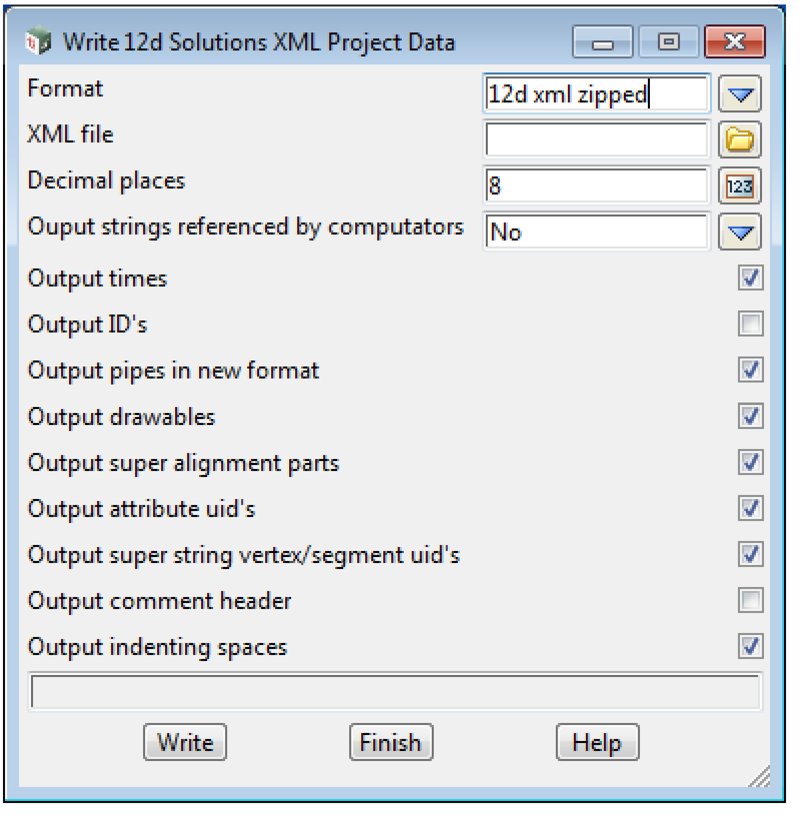
8.2.2.3 12d XML Project Data

Position of option on menu: **File I/O =>Data output =>12d =>12d XML project data**

The ***12d XML*** format is a special format defined by 12d Solutions to allow data to be easily transferred from other programs into 12d Solutions software such as 12d Model. The **12d XML** format is given in the Appendix.

Note - 12d XML contains the same information as a 12da file but it is a XML format.

Selecting **12d XML project data** brings up the **Write 12d Solutions XML Project Data** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
XML file	folder box		

name of the file for the information to be written out to in 12d XML format. If the file already exists, then you are asked to Append, Replace or Cancel.

Decimal places	integer box	8	

the number of decimal places used when writing out the data.

Output strings referenced by computators	choice box	No	Yes, No , De-reference

*if **Yes**, any strings referenced by computators by strings in the data source that are not in the data source, are also written out.*

*If **No**, any strings referenced by computators by strings in the data source that are not in the data source, are NOT written out.*

*If **De-reference**, any super strings in the data source with computators are de-referenced. That is, the* *computators are replaced by internal elements of the string.*

Output times tick box

if ticked, write out the creation times, etc. for the objects.
If not ticked, don't write out the creation times etc. for the objects.

Output ID's tick box

if ticked, write out the object IDs.
If not ticked, don't write out the object IDs.

Output pipes in new format tick box

if ticked,
If not ticked, don't write out.

Output drawables tick box

if ticked, write out the internal super alignment labelling as text.

Output super alignment parts tick box

if ticked, write out all the construction details for super alignments.
If not ticked, don't write out the construction details. Just write out the HG and VG segments.

Output attribute uid's tick box

if ticked, write out the attribute uids.

Output super string vertex/segment uid's tick box

if ticked, write out the any vertex and segment UIDs.

Output comment header tick box

if ticked, write out information as comments at the top of the 12da file.

Output indenting spaces tick box

if ticked, leave out spaces for indenting lines in the file. This will reduce the size of the text 12d XML file.

Write button

write out the data specified by the Data source.

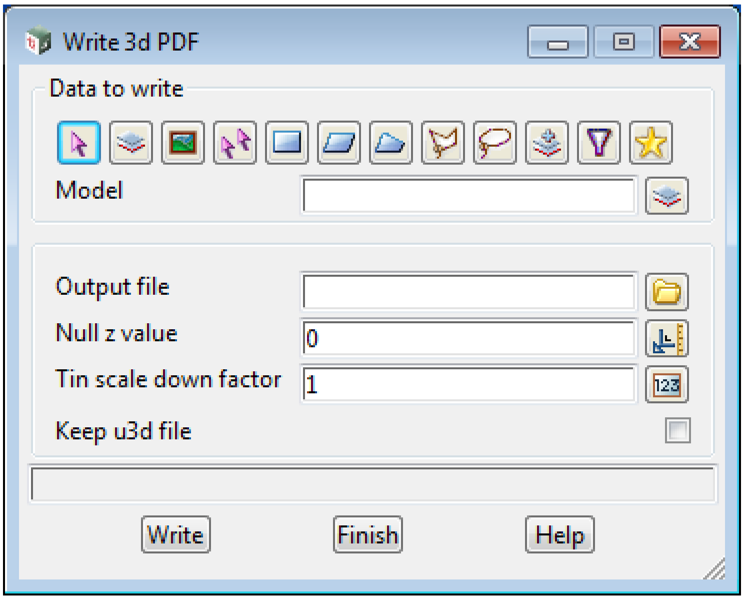
8.2.3 Write 3D PDF

This option writes out a Data Source as a 3D pdf file.

Currently no textures are applied.

Position of option on menu: **File =>Data output =>Export 3d pdf**

Selecting **Export 3d pdf** brings up the **Write 3d pdf** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

Data to search - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data is to be written out to a 3d pdf file.

Output file	file box		*.pdf

name of the 3d pdf that is created.

Null z value real box

z-value to use in place of null z-values in 12d strings.

Tin scale down factor integer box

if this value is greater than one then triangles from tins start to be dropped out.

Keep u3d file tick box

if ticked, the u3d file that becomes part of the 3d PDF is also kept as a separate file.

Write button

write out the specified data to the pdf file.

To view the 3D PDF, see 8.2.3.1 How to View the Generated 3D PDF

8.2.3.1 How to View the Generated 3D PDF

At the moment the option is not setting an eye and target point do to look at the data in the generated 3d pdf file, follow these steps:

1. Open the 3d pdf file with Acrobat pdf viewer. You'll then get a bit of a box drawn on the screen.
2. Move the cursor into the box and you will see the message **Click to activate**.Then click anywhere inside the box.
3. Then Right click to bring up a menu and choose **Full Screen Multimedia** from the menu. You should then see your data fitted in the view and displaying.

   Hold down the left mouse button to rotate. Wheel to zoom in and out. Shift LB to pan.

Return to 8.2.3 Write 3D PDF.

8.2.4 ArcView SHP Output

Position of option on menu: **File I/O =>Data output =>ArcView SHP**

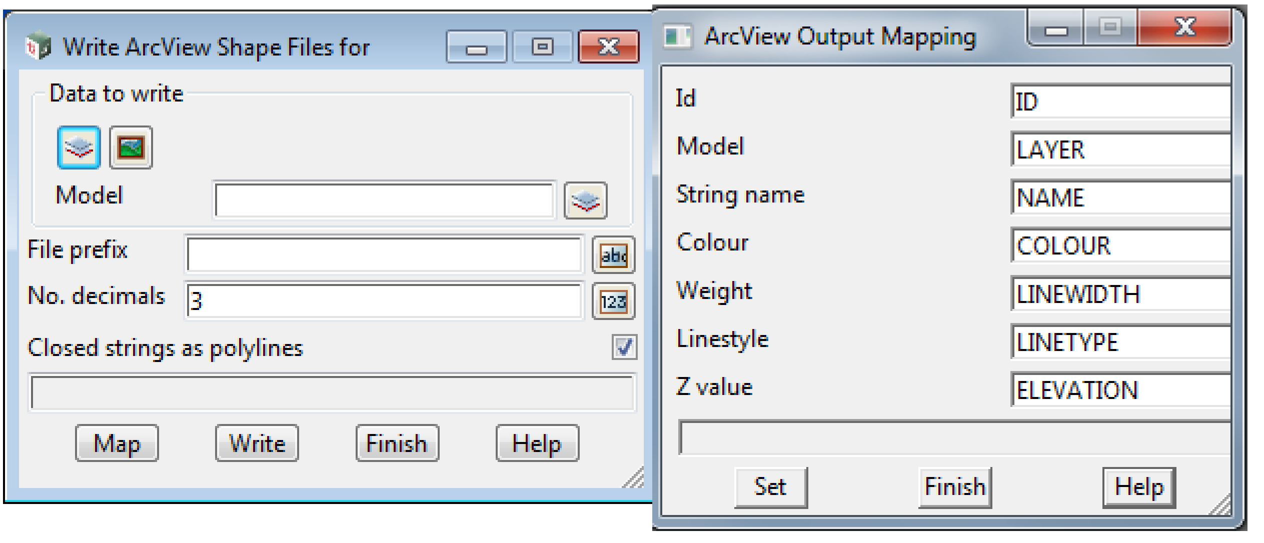
ArcView i/o is a separate chargeable module.

This Option is currently under development.

The *ArcView Shape File* format consists of nine different files containing different data types.

The ArcView shape format has no colour, model or layer information, string names, *etc.*, so this information has to be passed through as attributes for the data.

Selecting **ArcView SHP** displays the **Write ArcView Shape Files for** panel.



The fields and buttons used in the **Write ArcView Shape Files for** panel have the following functions.

Field Description Type Defaults Pop-Up

Data source type Model

Data to search - for a full description go to 4.19.3 Data Source.

Data source input

source of data is to be written out to a file.

File prefix input

prefix to use for the nine different shape files to be produced for the data.

No. decimals integer box 3

number of decimal places to use in the shape files.

Closed strings as polylines tick box 3

if ticked, write out closed strings in polyline shape files

Map button

brings up the **ArcView Mapping** *panel which defines the ArcView attributes to write the standard* ***12d Model*** *string header information out to.*

Write button

write out, in ArcView shape file format, all the string data specified by the Data source.

The fields and buttons used in the **ArcView Mapping** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Id		ID	

if non blank, write out an integer number which increments from 1.

Model		LAYER	

if non blank, write out the model of the string to this attribute.

String name		NAME	

if non blank, write out the name of the string to this attribute.

Colour		COLOUR	

if non blank, write out the colour of the string to this attribute.

Weight		LINEWIDTH	

if non blank, write out the weight of the string to this attribute.

Linestyle		LINETYPE	

if non blank, write out the linestyle of the string to this attribute.

Z value		ELEVATION	

if non blank, write out the first z-value of the string to this attribute.

Set	button		

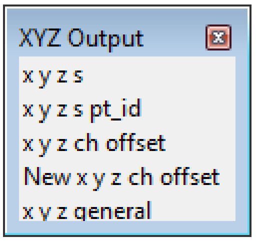
record the attribute names given in the panel as the ones to be use when writing out the data.

8.2.5 Output X Y Z Text Files

Position of menu: **File I/O =>Data output =>x y z**

The options under **x y z** out write out vertices of selected strings in a text format. It is also possible to write out string names, point IDs, string, segment and vertex attributes, and chainages and offsets from a selected string.

The **X Y Z** walk-right menu is



x y z s data
user specified x y z s point ID
user specified x y z chainage and offset
new user specified x y z chainage and offset
user specified x y z s point no and attribute

For *x y z s*, go to	8.2.5.1 X Y Z S Output
x y z pt_id	8.2.5.2 User X Y Z S Output
x y z ch offset	8.2.5.3 User X Y Z Chainage and Offset Output
New *x y z ch offset*	8.2.5.3 User X Y Z Chainage and Offset Output
General *x y z etc.*	8.2.5.4 User X Y Z and Attributes Output

8.2.5.1 X Y Z S Output

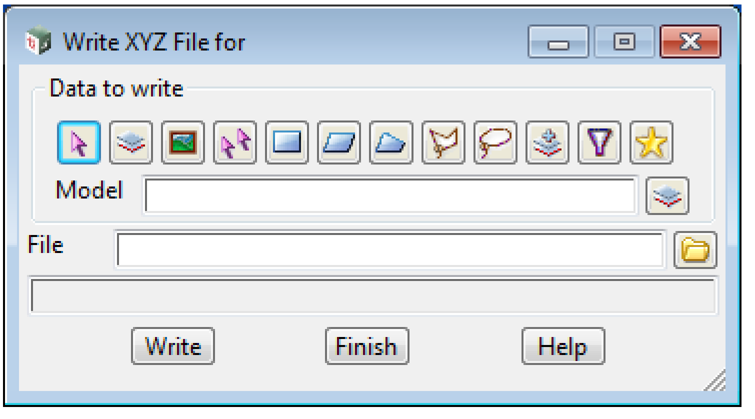
Position of option on menu: **File I/O =>Data output =>x y z =>x y z s**

The xyzs output format is identical to the input format - one point per line with the ***x y*** and ***z*** values separated by spaces, and ***s*** is the string name.

A string's colour and label are used as colour records and string labels in the output file.

4d, interface, super strings and alignment strings are only output as 3d strings. The use of the 12d Model 12da file format is formatted for writing out all ***12d Model*** string types.

Selecting the **xyzs** option brings up the **Write XYZ File for** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	
data selection type - for a full description go to 4.19.3 Data Source.			
Data source	input		
source of data is to be written out to a file.			
File	input		*.dat files
name of the file for the information to be written out to.			
Write	button		
write out, in x y z s format, all the string data specified by the Data source.			

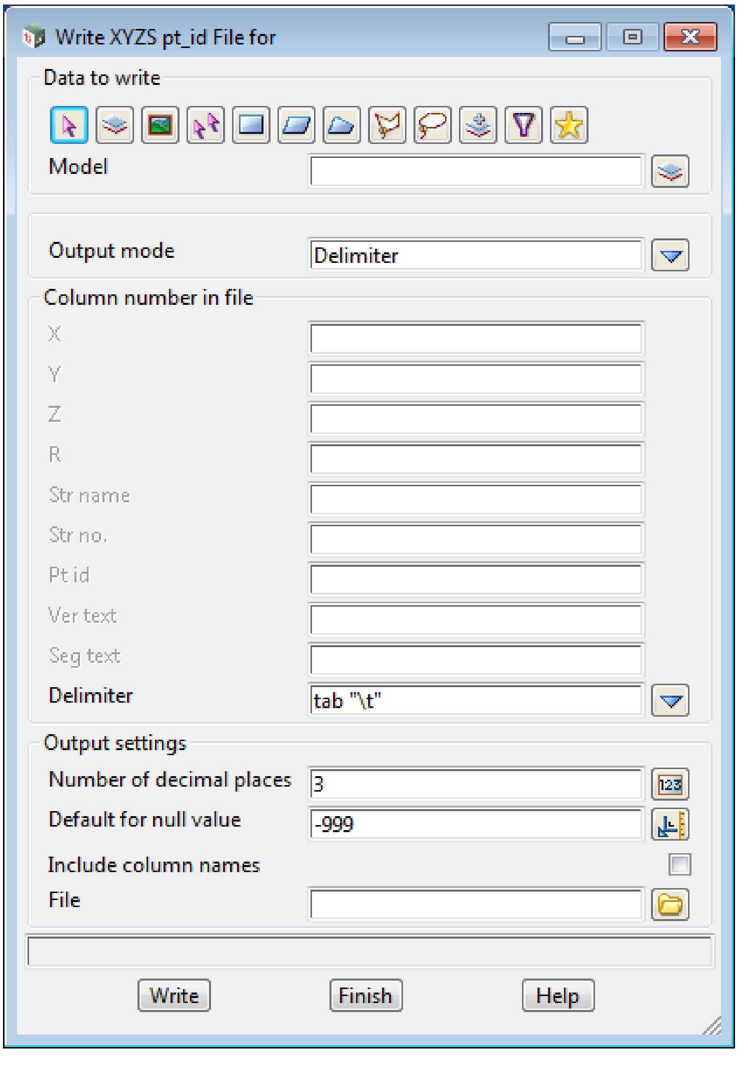
8.2.5.2 User X Y Z S Output

Position of option on menu: **File I/O =>Data output =>x y z =>x y z s pt_id**

The xyzs output format is identical to the user specified input format - one point per line with the x y z s and point id values in a user specified order separated by either delimiters or in fixed width columns.

super strings, 4d, interface and alignment strings are only output as 3d strings. The use of the 12d Model 12da file format is used for writing out all ***12d Model*** strings types.

Selecting the **xyzs pt_id** brings up the **Write XYZS pt_id File for** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data is to be written out to a file.

Output mode	choice box	delimiter	delimiter, fixed width

*if **delimiter**, the type of delimiter and the columns for the x, y, z, name and point number are given (and* *are all optional).*

If ***fixed width****, the start and end column positions are given for x, y, z, name and point number which are all optional.*

Output Settings

Number of decimal places input box 3

number of decimal places to use in the x, y and z values.

Default for null value input box -999

value to write out for z when it is a null value in ***12d Model****.*

Include column names tick box

if ticked, the names x, y, z etc. are written out in the appropriate columns as the first line of the file.

File input *.dat files

name of the file for the information to be written out to.

Write button

write out all the string data specified by the Data source.

8.2.5.3 User X Y Z Chainage and Offset Output

Position of option on menu: **File I/O =>Data output =>x y z =>x y z s ch offset**

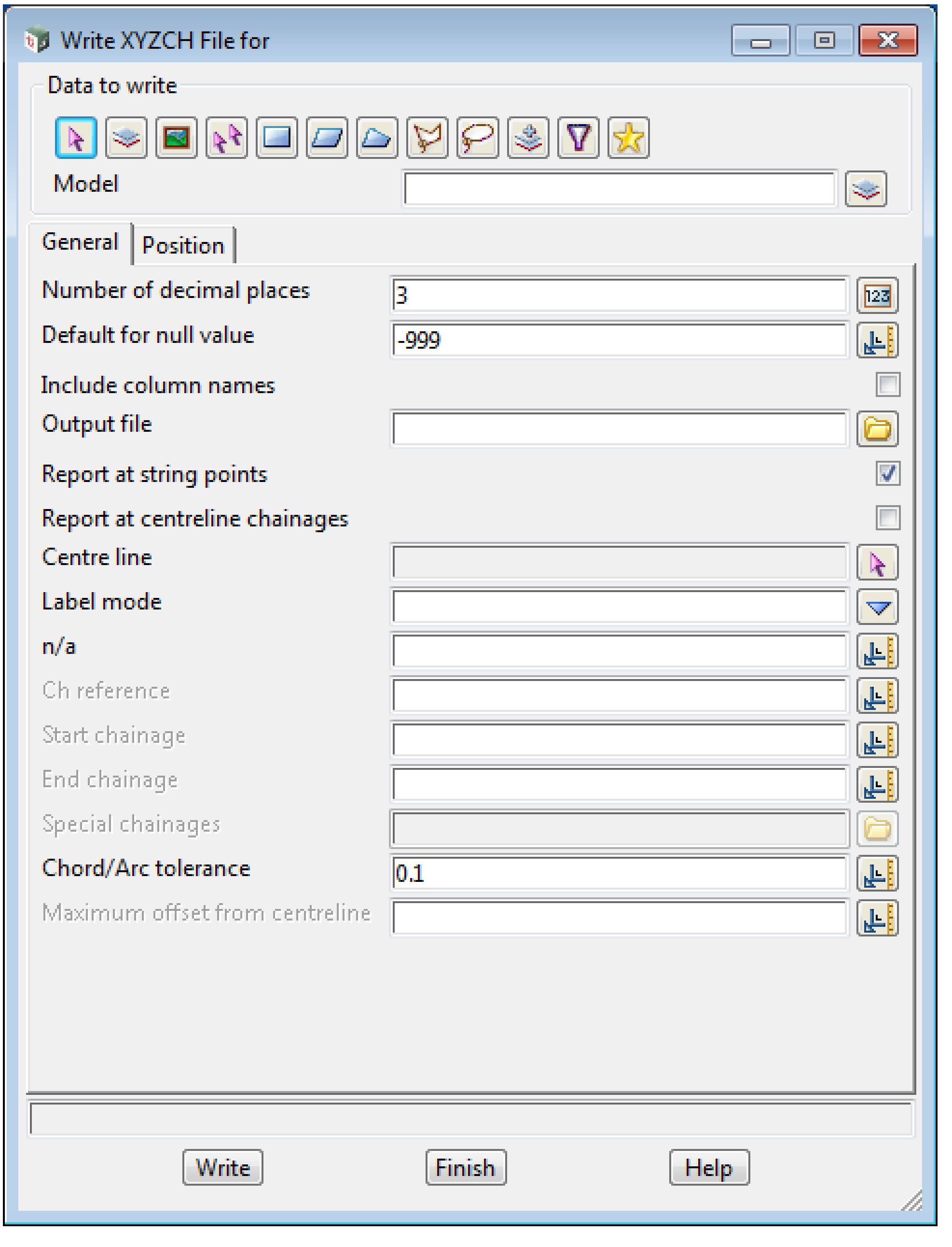
Position of option on menu: **File I/O =>Data output =>x y z =>New x y z s ch offset**

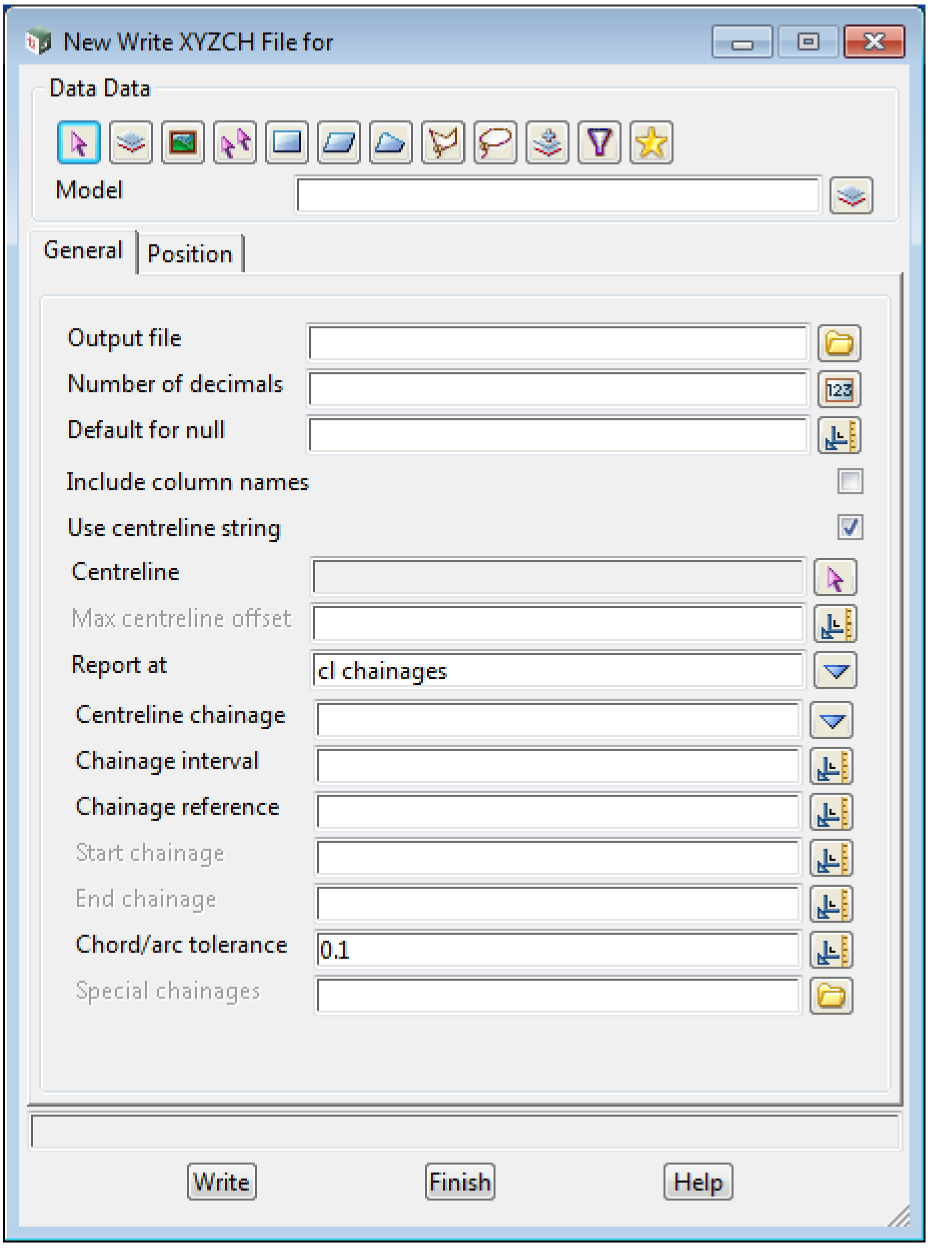
The *New xyz ch offset* option was introduced in V7C1g and will replace the standard option.

The difference is that the new option turns off the panel fields that are not required.

The *xyz and ch* output option writes out the x, y, z and chainage values for each string or a centreline string can be selected and the x, y, z and chainage values from the original string are output along with the chainage and offset from the selected reference string.

Selecting **xyzs ch offset** brings up the **Write XYZSCH File for** panel.





The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Data source type Model

data selection type - for a full description go to 4.19.3 Data Source.

Data source input

source of data is to be written out to a file.

Write button

write out all the data specified by the Data source.

General tab

Number of decimal places input box 3

number of decimal places to use in the x, y, z and chainage values.

Default for null value input box -999

value to write out for z when it is a null value in ***12d Model.***

Include column names tick box

if ticked, the names x, y, z etc. are written out in the appropriate columns as the first line of the file.

Output File input *.dat files

name of the file for the information to be written out to.

Report at string points tick box

if ticked, the values are output for the points (vertices) of the strings from the data source. If a centre line has been selected, then the chainage for the centre line is calculated by dropping the string vertex onto the centre line.

Report at centreline chainages tick box

if ticked, the specified chainages of the selected centre line (given by label mode etc.) are used and then the positions on the strings from the data source are calculated by going out at right angles to the centre line at the chainage.

Centreline string select

optional - a selected centreline to use for chainages.

Label mode input regular interval,
regular interval (plus end pts)
end points only
horizontal TPs,
vertical TPs,
horizontal discontinuities
vertical discontinuities
all discontinuities,
crests/sags
all horizontal points

type of chainages to use from the centreline.

Ch interval or n/a input

the regular interval to use for chainages.

Ch reference input 0

the chainages to user are integer multiples of the chainage interval added to the reference chainage. For example, if the reference chainage is 23.2 and the chainage interval 10, the chainages 3.2, 13.2, 23.2, 33.2 etc. will be used.

Start chainage input

if non-blank, the string chainage to start using.
If blank, start at the beginning of the string.

End chainage input

if non-blank, the string chainage to finish.
If blank, go to the end of the selected string.

Special chainage file box

file of special chainages to use.

Chord/arc tolerance input default chord/arc tolerance

the chord to arc tolerance to use on the centreline string for determining how many points are used around horizontal curves.

Maximum offset from centreline

if non blank, any vertices further than this distance from the centreline will NOT be written out.

If non blank, all vertices will be written out.

Position tab

Output mode choice box delimiter delimiter, fixed width

*if **delimiter**, then the specified data is written out, each value separated by the delimiter given in the* **Delimiter** *field.*

*If **fixed width**, the specified data is written out to a fixed position on the line. An end position is also given.*

String name

if the column number (delimiter) or start-end position (fixed width) is given, then the name of the string from the data source is written out. Otherwise it is not written out.

String vertex index

if the column number (delimiter) or start-end position (fixed width) is given, then the vertex index of the string vertex is written out.

String point id

if the column number (delimiter) or start-end position (fixed width) is given, then the point id of the vertex of the string is written out.

String chainage

if the column number (delimiter) or start-end position (fixed width) is given, then the string chainage of the vertex of the string is written out (not the centre line chainage).

String x/y/z

if the column number (delimiter) or start-end position (fixed width) is given, then the x/y/z co-ordinate of the position on the string is written out (not the centre line x/y/z).

CL offset

if the column number (delimiter) or start-end position (fixed width) is given, then the offset of the string from the centre line is written out.

CL chainage

if the column number (delimiter) or start-end position (fixed width) is given, then the chainage of the centre line is written out.

CL x/y/z

if the column number (delimiter) or start-end position (fixed width) is given, then the x/y/z co-ordinate of the position on the centre line string is written out.

8.2.5.4 User X Y Z and Attributes Output

Position of option on menu: **File I/O =>Data output =>x y z =>x y z general**

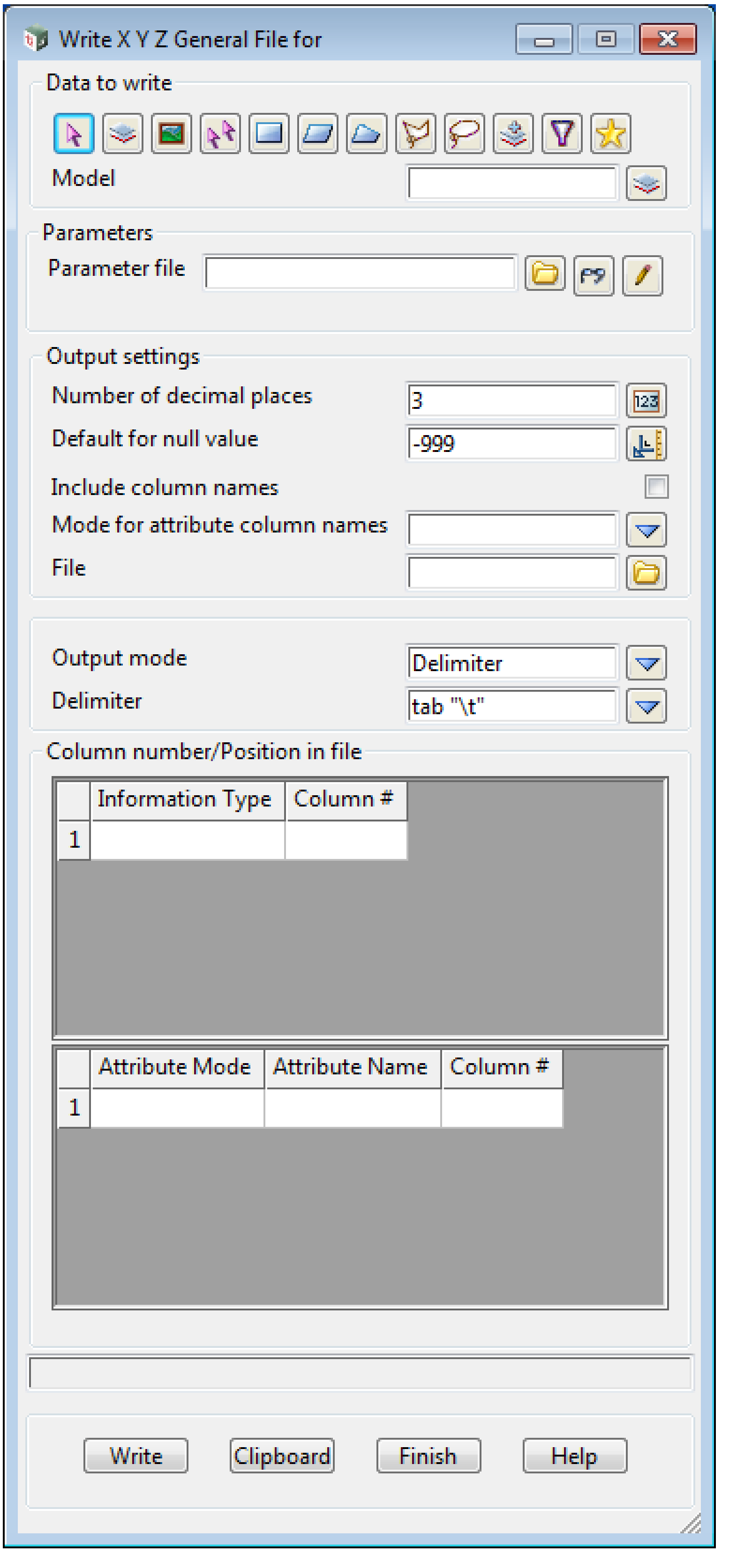
The xyzs output format is identical to the user specified input format - one point per line with the x y z s, point ID and attribute values in a user specified order separated by either delimiters or in fixed width columns.

In either case, the user specifies the order that the x, y, z, string name, point ID and attributes appear in the file. Only one or more of the items needs to be written out.

The set-ups for defining all the positions of all the data in the file can be written out to a file (.xyf) for re-use.

Super strings, 4d, interface and alignment strings are only output as 3d strings. The 12d Model 12da file format is used for writing out all string types.

Selecting **xyz general** brings up the **Write X Y Z General File for** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Data to write section

Data source type Model

data selection type - for a full description go to 4.19.3 Data Source.

Data source input

source of data is to be written out to a file.

Parameters section

Parameter file input *.xyf files

name of the file containing the settings for how the data is positioned in the output file.

Read icon button

read the parameter file in.

Write icon button

write the setting in the panel out to a parameter file.

Output settings section

Number of decimal places input box 3

number of decimal places to use in the x, y and z values.

Default for null value input box -999

value to write out for z when it is a null value in ***12d Model****.*

Include column names tick box

if ticked, the names x, y, z etc. are written out in the appropriate columns as the first line of the file.

File input *.dat files

name of the file for the information to be written out to.

Output mode choice box Delimiter Delimiter, Fixed width

*if **Delimiter**, the type of delimiter and the columns for the x, y, z, string name, attributes etc. are given (and are all optional).*

*If **Fixed width**, the start and end column positions are given for x, y, z, string name, attributes etc. which are all optional.*

Delimiter choice box tab \f one space, tab \t,
semi colon, comma
many spaces

*if **Output mode** is **Delimiter**, then this is the type of delimiter to separate data in the output file.*

Information Type Grid - Column number in file (Delimiter) or **Start/ End** (Fixed Width)

Information Type choice box x coord, y coord, z coord, chainage, radius
segment major, string name, string number
string name/string number, point id
vertex text, segment text

information to write out (x,y,z, attributes etc.) and this position in the output file.

Position # (Delimiter) or **Start/ End** (Fixed Width)

if Delimiter, the position number of the information in the tab separated file.

If Fixed Width, the start and end column position for the information.

Attribute Grid

Attribute Mode choice box string, segment, vertex *type of attribute to write out*

Attribute Name

name of the attribute to write out

Position # (Delimiter) or **Start/ End** (Fixed Width)

if Delimiter, the position number of the information in the tab separated file.

If Fixed Width, the start and end column position for the information.

Write button

write out all the string data specified by the Data source to the given file.

Clipboard button

write out all the string data specified by the Data source to the clipboard.

8.2.6 Output DGN Binary Files

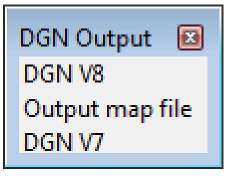
Position of menu: **File I/O =>Data output =>DGN**

DGN output is a separate chargeable module.

The options under **DGN** write DGN V7 and DGN V8 binary files.

There is also an option to create/edit a DGN output map file.

The **DGN** walk-right menu is



write DGN V8 binary file
Create/edit Output map file for DGN V8 binary
write DGN V7 binary file

For *DGN V8*, go to 8.2.6.1 Output DGN V8
Output map file 8.2.6.2 DGN V8 Output Map File Create/Edit
DGN V7 8.2.6.3 Output DGN V7

8.2.6.1 Output DGN V8

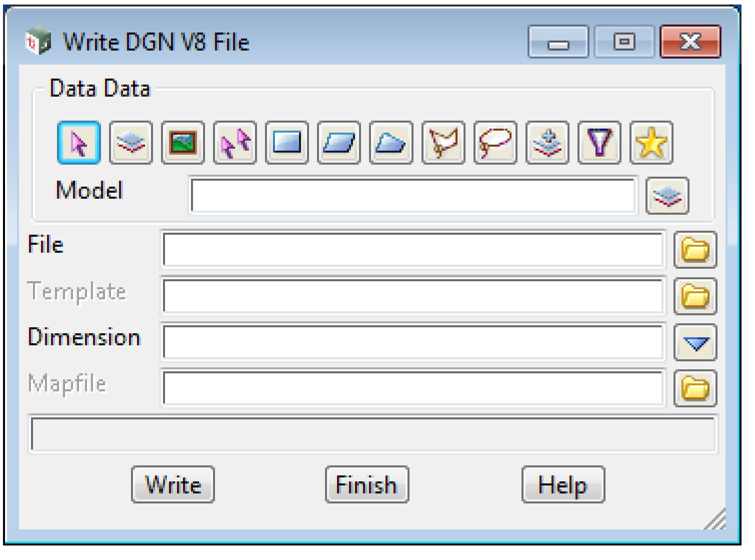
Position of option on menu: **File I/O =>Data output =>DGN = >DGN V8**

DGN output is a separate chargeable module.

NOTE: the format of the DGN file has changed for DGN V8 but it is proprietary to Bentley and it has not been published. This is a beta version of the DGN V8 Writer to try and write DGN V8 binary files. At this stage it is better to use DWG I/O to go in and out of Microstation V8.

Note: this option is under development

On selecting the **DGN V8** option, the **Write DGN V8 File** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data is to be written out to the DGN file.

DGN V8 file	file box		*.dgn files

name of the DGN V8 binary file to be write out to

Dimension	choice box		2d. 3d

dimension of the DGN file

Template	file box		*.dgn files

name of the DGN binary file to use as a DGN template

DGN output map file	file box		*.dgnmfx files

name of the DGN output mapping file to use

Write	button		

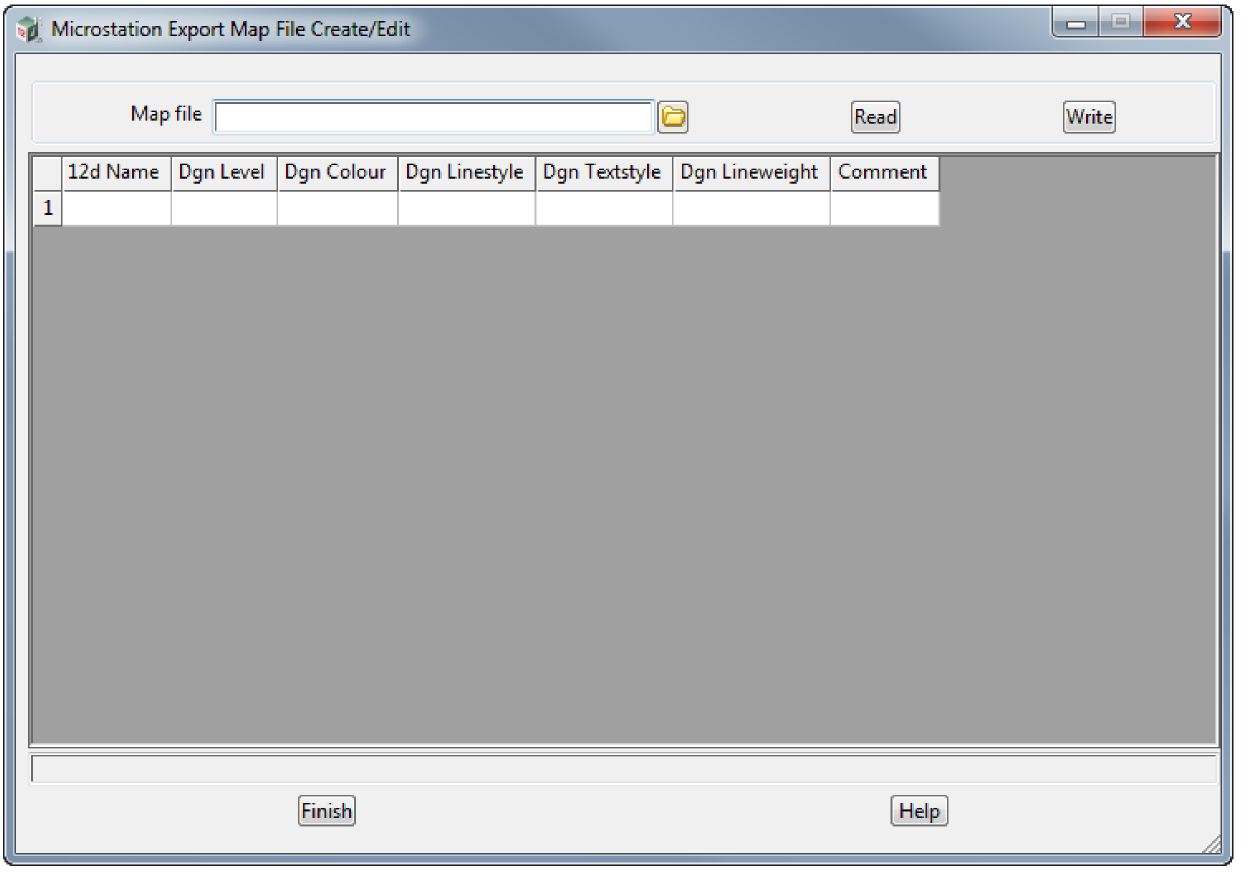
read the data out as a DGN V8 binary file.

8.2.6.2 DGN V8 Output Map File Create/Edit

Position of option on menu: **File I/O => Data Output => DGN => Output map file**

The **DGN output map file** option **creates** or **edits** the map files used when writing data out to Microstation DGN V8 binary format using the DGN V8 option (not used for plotting to DGN V8).

Selecting **DGN output map file** brings up the **DGN Export Map File Create/Edit** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Map file	input		*.acadmf

name of the DGN output map file to be created or edited.

Read button

read in the DGN output map file given in the **map file** *field and load the data into the fields of the table. If the file doesn't exist, an error message is given.*

Write button

write out the data in the table to the file name given in the **map file** *field. If the file already exists, a Replace-Cancel panel checks to see if the existing file is to be over written - if no or cancel is selected, nothing is written out.*

Grid Cells

12d Name

12d Model *string names for the mapping to apply to - wild cards (*) and characters (!) can be used.*

DGN level

*Microstation level to send the selected strings to. Use * to write the **12d Model** model of the string out*

Dgn colour

*Microstation colour to use for the selected strings. Use * for **12d Model** colour number.*

Dgn linestyle

*Microstation linestyle to use for the selected strings. Use * for **12d Model** linestyle*

Dgn Textstyle

*Microstation Textstyle to use for the selected strings. Use * for **12d Model** text style*

Dgn lineweight

Microstation lineweight to use for the selected strings.

8.2.6.3 Output DGN V7

Position of option on menu: **File I/O =>Data output =>DGN =>DGN V7**

DGN output is a separate chargeable module.

Using the **DGN V7** option,***12d Model*** produces DGN binary files (***.dgn** files - as defined in the public domain by Intergraph Corp) as used by Intergraph and Microstation. Since DGN V7 supports 3d faces, the triangles created in 12d Model can be transferred to DGN V7 for use in shaded models and walk-throughs.

DGN V7 files are either 2d or 3d. ***12d Model*** can write either type of file. If a 3d file is being written, the ***12d Model*** strings will be written out as three dimensional DGN lines. The alignment strings and arcs will be approximated by short lines.

If a 2d file is being written, arcs in alignment strings are written out as DGN plan arcs.

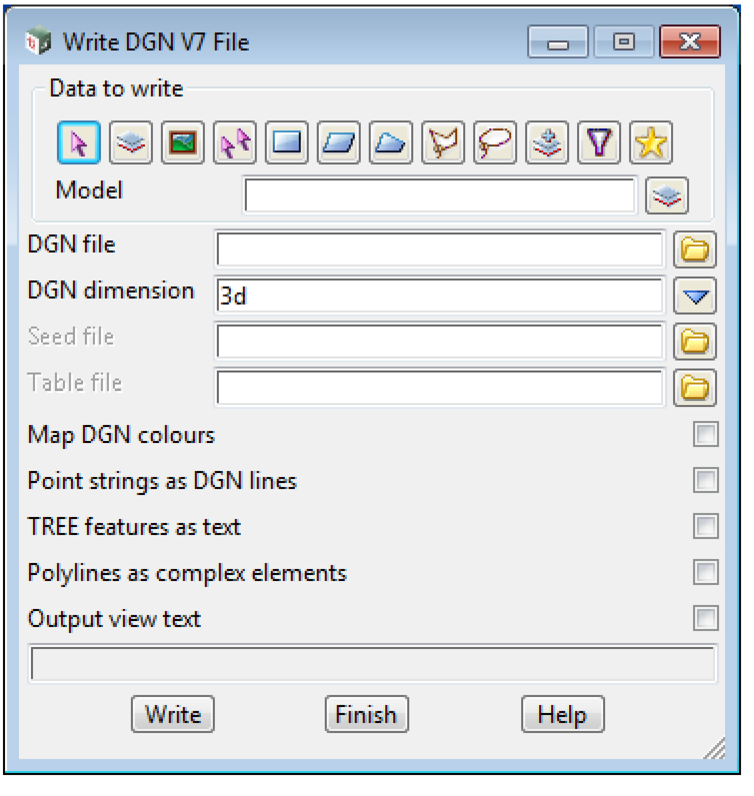
When creating a DGN V7 binary file, it is often convenient to append the information to an existing DGN file known as a **seed file** (the seed file usually contains set up information for the DGN V7 drawing).

In ***12d Model*** the user can specify a folder containing seed files by pointing to it using the environment variable MS_SEEDFILES_4D. In the DGN V7 output option, the pop-up for the **seed file** panel field displays all the files ending in **.dgn** in the folder pointed to by MS_SEEDFILES_4D.

If the environment variable MS_SEEDFILES_4D has not been set, the pop-up uses the current working folder to search for seed files.

It is possible to use an output map file (called a table file) with string name as the key to set DGN V7 levels, colours, weight, style and special symbols (see next section). Because of the limitation of 64 levels in DGN V7, using a table file is normally the best method of getting data across to DGN V7.

On selecting the **DGN V7** option, the **Write DGN V7 File** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source input

source of data is to be written out to a DGN V7 binary file.

DGN file input *.dgn

name of the DGN V7 file to write the model/view data out to. If the file exists, it is appended to, and the settings for TCB variables are read from the file. If the file does not exist, it is created with default TCB settings. The file is created either 2d or 3d depending on the **DGN Dimension** *field.*

DGN Dimension input 3d 2d/3d

this field controls the dimension of DGN files created for the case where the file named in the **Seed file** *does not exist.*

Seed file input files in the seed folder

if non-blank, this field gives the name of the DGN V7 file to be used as a seed file for the DGN V7 output. The folder containing the seed files, the seed folder, is given by the environment variable MS_SEEDFILES_4D.

Table file input *.tbl file

if non-blank, the file is used as an output map file between 12d Model and the .dgn binary file. See 8.2.6.3.1 DGN V7 Output Mapping File (Table File).

Map DGN Colours tick box

*if **not ticked**, 12d Model colour number n is mapped to DGN V7 colour n.*
*if **ticked**, some of the 12d Model colours are attempted to be mapped to DGN V7 colours.*

Point strings as DGN line tick box

*if **ticked**,* **12d Model** *points go out as DGN V7 two point strings with the same value for each point.*
*if not **ticked**,* **12d Model** *points go out to DGN V7 points.*

TREE features as text tick box

*if **ticked** and their is a table file match for TREE, then* **12d Model** *writes out feature strings of name "TREE" as text in DGN. Other feature strings go out as circles.*
*if not **ticked**, all* **12d Model** *feature strings go out as DGN V7 circles.*

Polylines as complex elements tick box

*if **ticked** and **DGN dimension** is **2d**,* **12d Model** *writes out polylines as complex elements made up of lines and arcs.*
*if not **ticked**, the polyline goes out as individual lines and arcs.*

Output view text tick box

*if **ticked**, any view text turned on (point numbers, z-values etc.) are output as text in DGN V7.*
*if not **ticked**, view text is not output.*

Write button

write out the all the data specified by the Data source to the file given in the **DGN file** *field.*

8.2.6.3.1 DGN V7 Output Mapping File (Table File)

When using the output options to write out a DGN V7 file, a DGN V7 output mapping file (table file) can be used, with string names as the key, so that the user can specify DGN level, colour, weight, style and some information which is placed in an DGN attribute. The format is based on the VicRoads specification.

The key can have wild cards (*) and wild characters (?) as for the input map files.

The table consists of lines containing either 8 or 16 free format fields. Fields 9-16 can be left off if they are not needed.

The key for the map is the ***12d Model*** string name which is given as the first field of a line. The rest of the fields on the line are used for .dgn information.

field 1	***12d Model*** string name (any length but if field 2 is a *, only the first four characters are passed)	
field 2	text of which the first four characters are used in a DGN V7 attribute. For example, the VicRoads code	
field 3	AS2482 feature code	
field 4	description	(any length but quotes needed if there are embedded spaces)
field 5	line level	(between 1 and 64)
field 6	line colour	(between 1 and 256)
field 7	line weight	(between 1 and 7)
field 8	line style	

fields 9-16 are only used for special symbols placed at points.

field 9	symbol font	(between 0 and 7)
field 10	symbol character	
field 1	symbol character justification	
field 12	symbol character height	
field 13	symbol character width	
field 14	symbol level	
field 15	symbol weight	
field 16	symbol colour	

Field 1, the 12d Model string name, can contain wild cards * or wild characters ?.

Field 2, a text string, and the first four characters of the text is inserted into a DGN V7 attribute. This DGN V7 attribute is used by the ***12d Model*** DGN V7 reader for matching with a ***12d Model*** input map file.

If field 2 is a * then the first four characters of the ***12d Model*** string name is inserted as an Intergraph attribute.

Line strings and Text strings only use fields 1 to 8 and ignore fields 9-16 (if they exist).

Fields 9 through 16 are only used for ***12d Model*** point strings which are then mapped with the specified Intergraph symbol at each point of the string.

If either field 12 or 13 is zero, then it is interpreted to mean that only the first 8 fields are used.

If any of the fields 5 through 16 (except 9 and 11) is a *, then that field is not used in the map.

Double quotes **"** are entered as text as \", a **'** as \' and a **** as \\\\.

An example of a map table where the first four characters of the ***12d Model*** string names are passed directly through to the DGN attribute.

```
CONT    *    50020000    "Contour - Standard  "    2    4    0    0
TOP     *    25060001    "Top - Cut/fill/bank "    2    11   1    2
TOE     *    25060002    "Toe - Cut/fill/bank "    3    11   1    7
ES*     *    50090001    "Existing surface-spot "  4    9    1    0    110    A    7    1000    1000    4
0   9
```

CIND * 50080000 "Contour - Index " 2 4 1 0

An example of a map table where the ***12d Model*** string names are used for mapping but different names go through to the DGN attribute.

101 CONT 50020000 "Contour - Standard " 2 4 0 0
102 TOP 25060001 "Top - Cut/fill/bank " 2 11 1 2
104 ES 50090001 "Existing surface-spot" 4 9 1 0 110 A 7 1000 1000 4 0 9

Return to 8.2.6.3 Output DGN V7

8.2.7 Output DWG/DFX/DXB Files

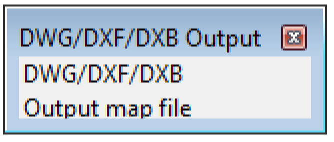
Position of menu: **File I/O =>Data output =>DWG/DXF/DXB**

DWG/DXF/DXB i/o is a separate chargeable module.

The option under **DWG/DXF/DXB** writes DWG, DXF and DXB files in a variety of AutoCAD formats.

There is also an option to create/edit an AutoCAD output map file.

The **DWG/DXF/DXB** walk-right menu is



write DWG/DXF/DXB V8 files
Create/edit Output map file for DWG

For *DWG/DXF/DXB*, go to 8.2.7.1 DWG/DXF/DXB Output
Output map file 8.2.7.2 AutoCAD Output Map File Create/Edit

8.2.7.1 DWG/DXF/DXB Output

Position of option on menu: **File I/O =>Data output =>DWG/DXF/DXB =>DWG/DXF/DXB**

DWG/DXF/DXB i/o is a separate chargeable module.

The DWG/DXF output format is for writing data out in a format compatible with AutoCAD versions 2.5 to 20010.

12d Model line strings are output as POLYLINEs, point strings as a series of AutoCAD POINTs and triangles as 3DFACEs.

By default, the string or triangle colour is used in the Autocad colour record.

Trimeshes can be written out to a DWG file as either Polyface Meshes, 3DFaces or not at all.

The choice is set by the environment variable TRIMESH_TO_DWG_MODE_4D in the External Apps > AutoCAD. node of the **Edit Environment Variables** panel.

An AutoCAD Polyface Mesh can only have one colour so if the choice is **Polyface mesh** then the trimesh colour is used and any differently coloured trimesh triangles will only go out in the trimesh colour.

If the choice is **Faces** then each 3DFace is given the colour of the corresponding triangle of the trimesh.

The DXF layer used is the items model name with any spaces in the model name replaced by a minus (-). However, a DXF output map file with matches on string name can be used to give DXF layers, colours and line type (see next section).

Super strings, 4d, interface and alignment strings can only output as Autocad POLYLINES.

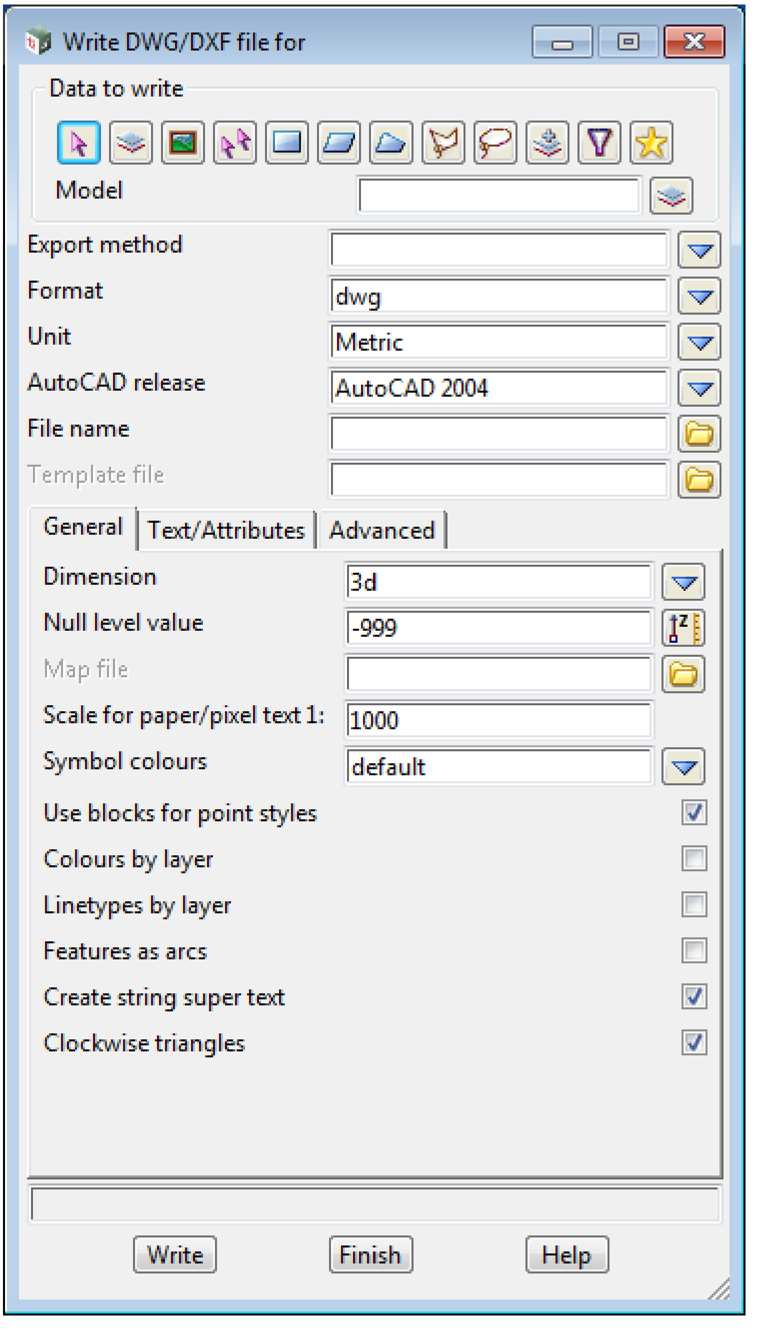
When creating an AutoCAD file, it is often convenient to use an existing AutoCAD file as a template or **seed file** (the seed file usually contains set up information for the AutoCAD drawing).

In ***12d Model***, the user can specify a folder containing seed files by pointing to it using the environment variable ACAD_SEEDFILES_4D. In the AutoCAD DWG/DXF output option, the pop-up for the **seed file** panel field displays all the files with the appropriate ending in the folder pointed to by ACAD_SEEDFILES_4D.

If the environment variable ACAD_SEEDFILES_4D has not been set, the pop-up uses the current working folder to search for seed files.

It is also possible to use a label map file to create labels for vertex information such as z-values, symbols and point numbers which are written directly to the AutoCAD file. See the section 8.7.1 Create/Edit Label Map Fileand 8.7.2 Apply a Label Map Filefor more information on the label map file.

On selecting the **DWG/DXF** output option, the **Write DWG/DXF file for** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data is to be written out to a file.

Format	choice	dwg	dwg, dxf, bdxf

AutoCAD format of the file.

Unit	choice	Metric	Metric, English

AutoCAD units type.

AutoCAD release	choice	AutoCAD 2004	AutoCAD 12, 13, 14 AutoCAD 2000, 2002, 2004 2005, 2006, 2007

AutoCAD version of the file to be produced.

File name input

name of the ACAD file for the data to be written out to.

Template file input

name of the file to use as the AutoCAD template

Button at Bottom

Write button

write out, in DWG/DXF format, all the data specified by the Data source, to the file given in the file field.

General tab

Dimension	choice	3d	2d, 2d and contours 3d,3d

*if **2d**, the z-value of all the data is set to zero and arcs and curves in alignment strings are written out as dxf polylines with bulges for the arcs.*

*If **2d and contours 3d**, the z-value of all the non-contour data is set to zero and arcs and curves in alignment strings are written out as dxf polylines with bulges for the arcs. Contours are written out as 3d polylines.*

*If **3d**, strings will be written out as three dimensional DXF polylines. Alignment strings and arcs will be approximated by short lines.*

Null level value input -999

AutoCAD z-value to use for any vertices with the **12d Model** *null value*

Map file input *.acadmf files

*if non-blank, the name of an **AutoCAD output map file** to map the data. See* 8.2.7.2 AutoCAD Output Map File Create/Edit.

Scale for paper/pixel text 1: input 1000

scale to use to convert **12d Model** *paper and pixel text to ACAD world units*

Symbol colour choice box default default, by block, by layer

*if **default**, use the* **12d Model** *colour.*

*If **By block**, write out as BY BLOCK.*

*If **By layer**, write out as BY LAYER.*

Use blocks for point styles tick box

*if **ticked**, write out blocks for symbols at vertices*

Colour by layer tick box no tick

*if **ticked**, the colour is set to BY LAYER*

Linetypes by layer tick box no tick

*if **ticked**, the linestyle is set to BY LAYER*

Features as arcs tick box no tick

*if **ticked**, feature strings output as arcs in DXF.*
*if not **ticked**, the centre of the feature string is output as a point.*

Create string super text tick box no tick

*if **ticked**, write out text for super string vertex text and segment text*

Clockwise triangles tick box tick

*if **ticked**, the vertices for triangles in a tin are written out in clockwise order.*

Text/Attributes

Label map file *.lmf files

if not blank, the name of a file to be used as label map file for creating labels that will go out to the AutoCAD file.

Layer name for symbols

if not blank, the name of the layer to place symbols in.

Text tick boxes *tick boxes*

*if **Vertex** is ticked, vertex text is written out*
*if **Vertex index** is ticked, vertex indices are written out*
*if **Point id** is ticked, point ids (vertex id's) are written out*
*if **Height** is ticked, the z-values at each vertex are written out*
*if **Code** is ticked, the string name is written out*

Att tick boxes *tick boxes*

Layer Name

name of the AutoCAD layer to write text to

Write attributes: vertex tick box

if ticked, write out vertex text

Write attributes: segment tick box

if ticked, write out segment text

Advanced

Space in model name choice box space space, underscore, hyphen

*if **space**, then spaces in* **12d Model** *names go out as spaces to ACAD layer names.*

*If **underscore**, then spaces in* **12d Model** *names go out as underscores to ACAD layer names.*

*If **hyphen**, then spaces in* **12d Model** *names go out as hyphens to ACAD layer names.*

Acad point style choice style 0 Acad point styles

Point mode choice relative relative, absolute

*if **relative**, then*

*If **absolute**, then.*

Point size (%) input 5

Output view text tick box no tick

*if **ticked**, any view text turned on (point numbers, z-values etc.) are output as text in DXF.*
*if not **ticked**, view text is not output.*

Explode interface strings tick box no tick

*if **ticked**, interface strings are broken into separate strings for the cut or fill colour.*
*If not **ticked**, interface strings are written out as one string in magenta.*

Use model name for mapping tick box no tick

*if **ticked**,*
*if not **ticked**,.*

Use symbol name for mapping tick box no tick

*if **ticked**,*
*if not **ticked**,.*

Use textstyle_data name for mapping tick box no tick

*if **ticked**,*
*if not **ticked**,.*

Associate vertex attributes with symbol tick box no tick

*if **ticked**,*
*if not **ticked**,.*

Vertex attributes height input -999

Transition chainage interval

since AutoCAD does not have a transition element (spirals etc.), a transition can only be approximated. This is the chainage length to break up transitions.

Transition chord to arc

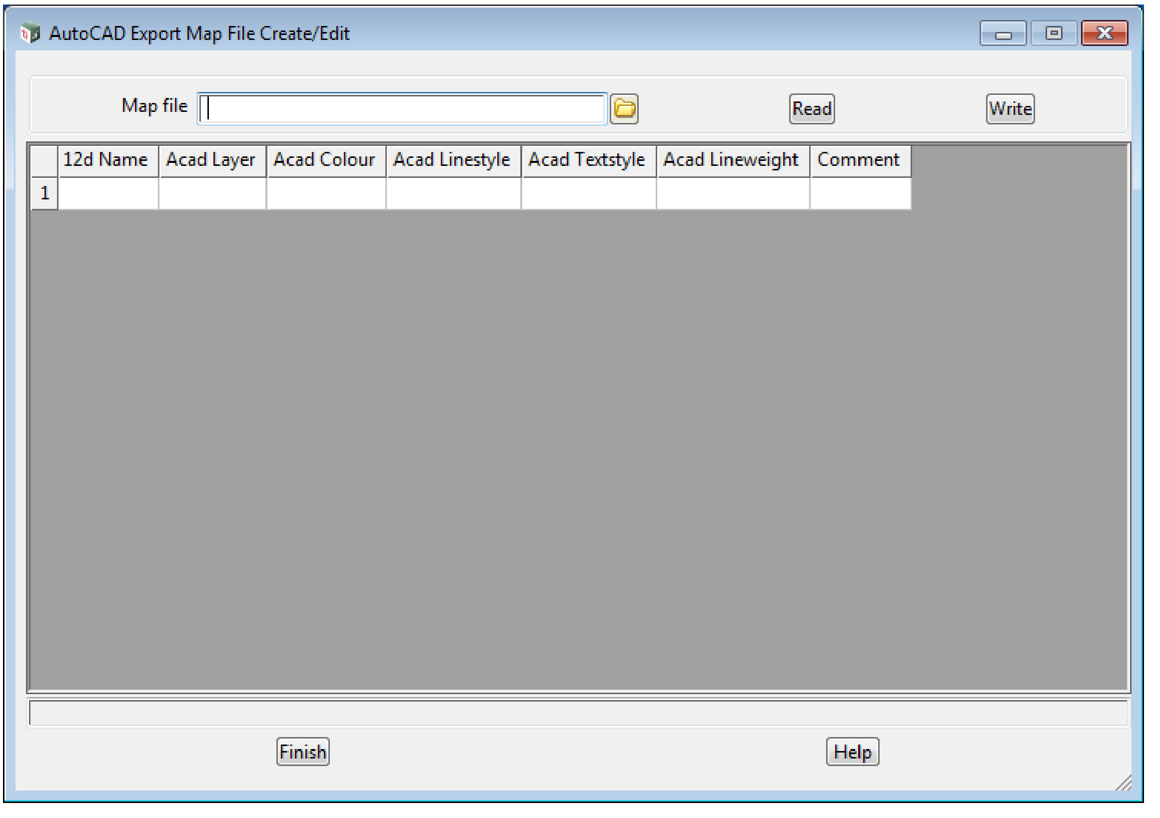
the chord-to-arc tolerance to use when approximating transitions.

8.2.7.2 AutoCAD Output Map File Create/Edit

Position of option on menu: **File I/O =>DWG/DXF/DXB => Output map file**

The **Output map file** option **creates** or **edits** the map files used when writing data out to AutoCAD using the DWG Output option (not used for plotting to AutoCAD).

Selecting **Output map file** brings up the **AutoCAD Export Map File Create/Edit** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Map file	input		*.acadmf

name of the AutoCAD output map file to be created or edited. See 8.2.7.2.1 DWG/DXF Output Map File acadmf.

Read button

read in the AutoCAD output map file given in the **map file** *field and load the data into the fields of the table. If the file doesn't exist, an error message is given.*

Write button

write out the data in the table to the file name given in the **map file** *field. If the file already exists, a Replace-Cancel panel checks to see if the existing file is to be over written - if no or cancel is selected, nothing is written out.*

Grid Cells

12d Name

12d Model *string names for the mapping to apply to - wild cards (*) and characters (!) can be used.*

Acad layer

*AutoCAD layer to send the selected strings to. Use * to write the* ***12d Model*** *model of the string out*

Acad colour

*AutoCAD colour to use for the selected strings - a number between 0 and 256. Use * for* **12d Model** *colour number.* Use BYLAYER *for* ACD BYLAYER

Acad linestyle

*AutoCAD linestyle to use for the selected strings. User * for* **12d Model** *linestyle or* BYLAYER *for* ACD BYLAYER

Acad Textstyle

*AutoCAD Textstyle to use for the selected strings. Use * for* **12d Model** *text style*

Acad lineweight

AutoCAD lineweight to use for the selected strings.

Comment

Place comments in this field.

8.2.7.2.1 DWG/DXF Output Map File acadmf

See 8.2.7.2 AutoCAD Output Map File Create/Edit.

When using the output options to write out a DWG/DXF file, an output map file can be used, with 12d Model string names as the entity-name to match the key, so that the user can specify AutoCAD colour and style. BYLAYER can be used with colour and style.

The key can have wild cards (*) and wild characters (?) as for the input map files.The fields in the DWG/DXF output map file are (ACD = AutoCAD):

field 1	key - string name	can include wild cards * and wild characters ?
field 2	ACD layer	* for ***12d Model*** name
field 3	ACD colour	number between 0 and 256, * for ***12d Model*** colour mapped to ACD, BYLAYER for ACD BYLAYER
field 4	ACD line type	* for ***12d Model*** linestyle, BYLAYER for ACD BYLAYER
field 5	ACD text style	not yet used, * for ***12d Model*** text style

An example of an DWG/DXF output map file is.

```
//      1       2         3         4             5
//      key     layer     colour    linetype      textstyle
        cont*   CONTOURS  1         1             *
        EB*     ROAD      2         CONTINUOUS    *
        103     *         3         *             *
        fe*     FENCE     *         BYLAYER       *
        SURV*   SURVEY    BYLAYER   DASH          *
```

Return to8.2.7.1 DWG/DXF/DXB Output.

8.2.8 Genio Output

Position of option on menu: **File I/O =>Data output =>Genio**

Genio i/o is a separate chargeable module.

The **Genio output** option write out the strings selected by the data source, to a genio file, the format used by MX (formerly Moss). A genio 001 option is used to specify the record format.

Because MX and in 12d Model have different data types, not all 12d Model data can be transferred to MX. The user must be aware of what data can be transferred to MX.

1. MX model names are limited to thirty two characters (in upper case). Hence the first thirty two characters of the 12d Model model name are converted to upper case and used as the MX model name.
2. MX string names can only be four characters and each string in MX must have a unique name. This limits MX to a maximum of 1,679,616 string. Strings can have the same name in the genio file but the strings will be *renamed* when read into MX to give each string a unique name.

   Since 12d Model places no restrictions on string name length, the following rules are used to produce genio string names.

   For a ***line-string***, the first four characters of the string name are used as the genio string name. The four characters are mapped to upper case. Remember that MX line strings can not start with a P.

   For a ***point-strings*** whose names start with **P,** the first four characters of the string name are used as the genio string name. Otherwise the genio string name is **P** plus the first three letters of the point-string name. The three letters are mapped to upper case.
3. 12d Model null heights go out as -999 to MX.
4. The four character limit and uniqueness for MX string names restricts the number of strings that can be read into MX. For example, since points strings must start with P so there can be a maximum of 46,656 point strings. Or if a MX string must start with ABB say, then there can only be 36 distinct ABB strings in the MX project.
5. All MX x and y co-ordinates have to be positive. That is, all co-ordinates must be in the first quadrant.
6. 12d text strings go out as MX text strings but there is a maximum of 44 characters in the MX text string. There is no unit for height in the MX text string as well as no font.The name of a MX text string must start with a * so 12d Model uses * and the first three characters of the text string's name.
7. 2d and 3d strings in 12d Model go out to MX as 2d and 3d strings respectively.
8. 4d strings in 12d Model go out to MX as 4d strings except that the text is restricted to four characters.
9. Alignment and super alignments can be written out to MX as MX 3d, 6d or 12d string.
10. For super strings:

    Point id's are ignored in all cases.
    On the Genio output panel, this is a tick box *Include segment text* and if it is ticked then segment text is written out as MX text strings, otherwise segment text is ignored.
    On the Genio output panel, this is a tick box *Include vertex text* and if it is ticked then the super string is written out as a MX 4d string and with the first four characters as the vertex text for a point.
    Attributes are ignored in all cases.
    Diameter and culvert ***width*** and ***height*** are ignored.
    Colour and **segment colour** are ignored.
    If there is an ***invisibility*** segment, the string is broken into two.
    Tinability is a real problem. MX has no such concept apart from the entire string being a point

string or a line string. There is a tick box to write out non-tinable data. If a segment is non tinable, the string is broken into two.

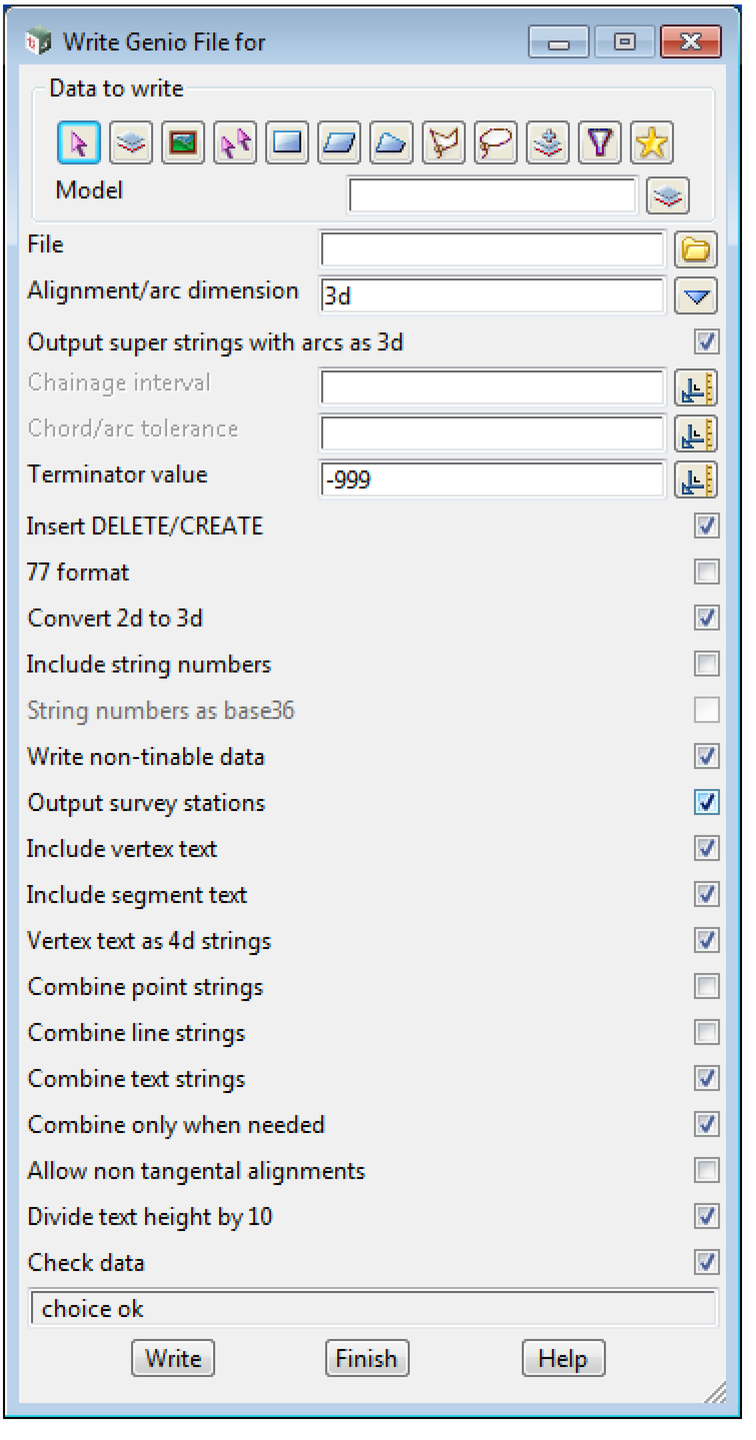
Arcs in super strings or arc strings or circles are broken into line segments using the chord to arc tolerance in the genio output panel, or the system default if there is no value in the panel.

Super strings with a fixed z-value and no vertex text (i.e. that is no z-value at each point, just the one z for the entire string) go out at MX 2d strings.

Super strings with a z-value at each vertex and no vertex text (that is a z-value at each point) go out as MX 3d string

Super strings with a z-value at each vertex and vertex text (that is a z-value at each point and text at each point) go out as MX 4d strings with the four characters of the vertex text as the forth dimension.

On selecting the **Genio** output option, the **Write Genio File for** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data is to be written out to a file.

File	input		*.mos files

name of the genio file the model/view is to be written out to. If the file already exists, the data will be appended to the bottom of the file.

Alignment/arc dimension input 3d 3d,6d,12d, 6d & 12d

*if **3d** is selected, alignment and arc strings as written out as MX 3d string as a series of straight lines (x, y, z at each point).*
*If **6d** is selected, alignment and arc strings are written out as MX 6d strings (chainage, x, y, z, bearing, and radius at each point).*
*If **12d** is selected, alignment and arc strings are written out as MX geometry strings (called 12d strings in MX). This is the only way that full horizontal and vertical geometry can be transferred to MX.*

*If **6d & 12d** is selected, alignment and arc strings are written out as MX 6d strings and MX geometry strings.*

Output super strings with arcs as 3d tick box

if ticked,

Chainage interval input

*if **3d** is selected as the* **Alignment/arc dimension***, the alignments and arcs are approximated using the* **Chainage interval** *and\ the* **Chord/arc tolerance.**

Chord/arc tolerance input

*if **3d** is selected as the* **Alignment/arc dimension***, the alignments and arcs are approximated using the* **Chainage interval** *and\ the* **Chord/arc tolerance.**

Terminator value input -999

value to use to terminate strings in the genio file.

Insert DELETE/CREATE tick box tick

if ticked, the genio commands DELETE and CREATE are written at the top of the file to correctly define the moss models for the data.

77 format tick box

*if **ticked**, use the MX 77 formats for the genio strings.*

Convert 2d to 3d tick box tick

*if **ticked**, write strings with a fixed z-value (2d strings) out in MX 3d string format in the genio file.*

Include string numbers tick box

*if **ticked**, the MX string name includes the surveyors string number. Note - the MX string name can still only be four characters.*

String numbers as base 36 tick box

*if **ticked**, the surveyors string number is converted to Base 36 (that is user 0...9, A... Z). Note - the MX string name can still only be four characters.*

Write non-tinable data tick box tick

*if **ticked**, strings that are not tinable are written out.*

*If **not ticked**, strings that are not tinable are not written out.*

Output survey stations tick box tick

*if **ticked**,.*

Include vertex text tick box tick

*if **ticked**, write out strings with vertex text as MX 4d string with the first four characters of the vertex text as the text on the MX 4d string, otherwise do not write the text out.*

Include segment text tick box tick

*if **ticked**, write out segment text as text strings otherwise do not write the text out.*

Vertex text as 4d strings tick box tick

*if **ticked**, vertex text is written out as a MX 4d string*

Combine point strings tick box

*if **ticked**, point strings of the same name in 12d are combined into one point string when written out.*

Combine line strings tick box

*if **ticked**, strings of the same name in 12d are combined into one MX string (using MX discontinuities to separate the strings) when written out. This is trying to get over the MX four character name limitation.*

Combine text strings tick box tick

*if **ticked**, text strings of the same name in 12d are combined into one MX text string when written out. This is trying to get over the MX four character name limitation.*

Combine only when needed tick box tick

*if **ticked**, when there are more strings of the same name in 12d than the MX four character name restriction would allow, the strings are combined into one MX string when written out. This is trying to get over the MX four character name limitation.*

Allow non tangential alignments tick box

*if **ticked**, alignments where some elements are not tangential are written out.*

*if **not ticked**, alignments with non tangential elements are not written out.*

Divide text height by 10 tick box tick

*if **ticked**, divide the text height by 10*

Write button

write out in genio format all the data specified by the Data source, to the file given in the **file** *field. If the file already exist, the data will be appended to the file.*

8.2.9 LandXML Output

Position of option on menu: **File I/O =>Data output =>LandXML**

LandXML i/o is a separate chargeable module.

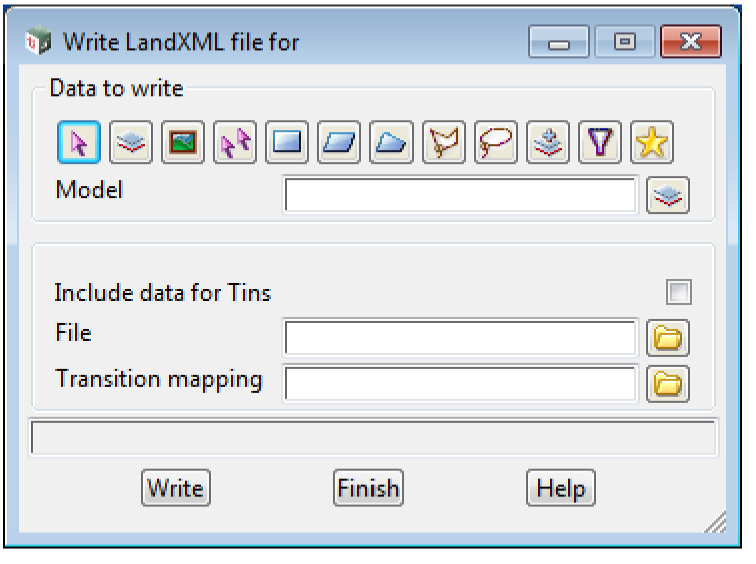
LandXML is a format that attempts to cover some civil and surveying entities. It does not include any information such as colours, styles *etc*. but just some geometry definitions.

Unfortunately to make the format useful, every vendor has their own proprietary extensions which makes the format of only limited value. A different tailored LandXML writer is required for each vendor variation. The names and definitions of transitions vary from vendor to vendor (***12d Model*** has special LandXML writers for LINZ-XML (for LandOnline NZ) and Leica-XML)

Finally the methodology behind the LandXML model is based on US ideas of using X-sections and not strings to model data. This makes is unsuitable except for simple civil models.

This option is under continual developed as the LandXML standard keep changing.

On selecting the **LandXML** option, the **Read LandXML File** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data is to be written out to a file.

Include data for Tins	tick box		

if ticked, tins are included in the LandXML output

File	input		*.xml

name of the LandXML file to write to

Transition mapping			*.trans_map files

file containing the mapping of transitions between the names of LandXML transitions and equivalent transitions in 12d Model

Write	button		

write the selected data to a LandXML file.

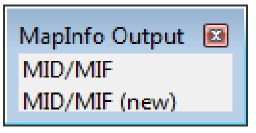
8.2.10 Output MapInfo MID/MIF Files

Position of menu: **File I/O =>Data output =>Mapinfo**

MID/MIF i/o is a separate chargeable module.

The options under **MapInfo** write out MapInfo MID and MIF files.

The **MapInfo** walk-right menu is



write out MID/MIF data
another MID/MIF writer

For *MID/MIF* — 8.2.10.1 Write Mapinfo Tab/Mif Files For

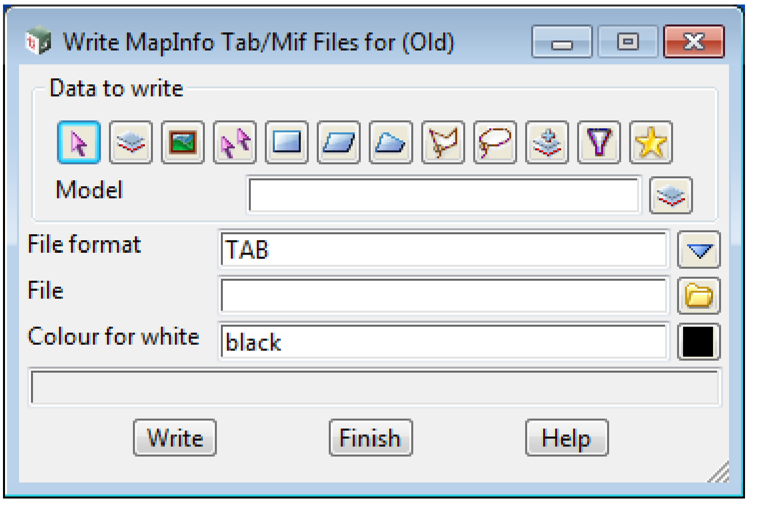
MID/MIF (new) — 8.2.10.2 Write Mapinfo Tab/Mif Files For (New)

8.2.10.1 Write Mapinfo Tab/Mif Files For

Position of option on menu: **File I/O =>Data output =>Mapinfo =>MapInfo MID/MIF**

The *MapInfo MID/MIF* format is used for sending data to MapInfo.

On selecting the **MapInfo MID/MIF** option, the **Write MapInfo Tab/Mif Files for** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data is to be written out to a file.

File format	choice box	TAB	TAB, MIF

MapInfo format to use for writing out the data.

File	choice box		*.tab, *.mif files

name of the file to write the data out to.

Colour for white	colour box		available colours

colour to use in MapInfo for white in ***12d Model*** *(because MapInfo has a white background).*

Write	button		

write out, in MapInfo file format, all the string data specified by the Data source.

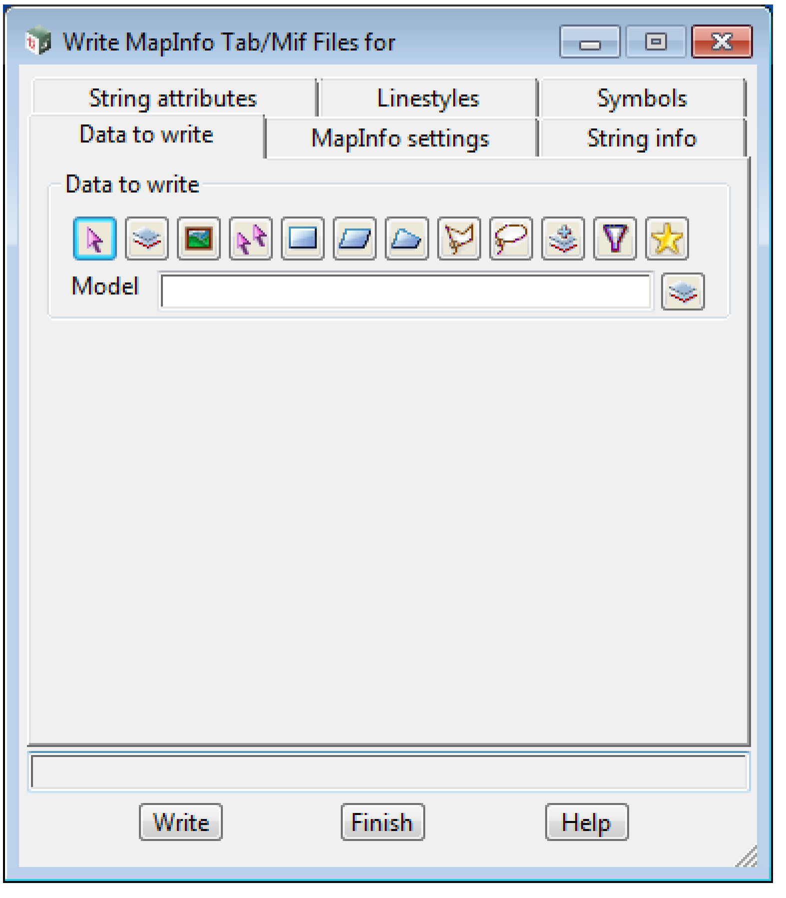
8.2.10.2 Write Mapinfo Tab/Mif Files For (New)

Position of option on menu: **File I/O =>Data output =>Mapinfo =>MapInfo MID/MIF (new)**

This section of documentation is a work in progress and will be updated in subsequent releases.

The *MapInfo MID/MIF* format is used for sending data to MapInfo.

On selecting the **MapInfo MID/MIF** option, the **Write MapInfo Tab/Mif Files for** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Data to write Tab

Data source type Model

data selection type - for a full description go to 4.19.3 Data Source.

Data source input

source of data is to be written out to a file.

MapInfo settings Tab

Coordinate system choice box

MapInfo file of coordinate definitions. The coordinate system that the current data is in is selected from the list.

Note *- data in MapInfo is stored in longitude and attitude. When MapInfo reads data in it needs to know the coordinate system that the data is in so that it can convert the data to longitude/attitude as it* *reads the file. The list of coordinate systems in the* **Coordinate system** *pop-up is in the file mapinfo.4d. Mapinfo.4d has exactly the same format as the file MAPINFOW.PJR used by MapInfo.*

If there are any coordinate systems missing that are in your MapInfo system, simply copy the file MAPINFOW.PRJ from your MapInfo system into the ***12d Model User*** *folder and give it the name mapinfo.4d instead of MAPINFOW.PRJ*

File format choice box TAB TAB, MIF

MapInfo format to use for writing out the data.

File choice box *.tab, *.mif files

name of the file to write the data out to.

Colour for white colour box available colours

colour to use in MapInfo for white in ***12d Model*** *(because MapInfo has a white background).*

Scale for paper/pixel text 1: real box 1000

scale to use to map paper and pixel text sizes

String Info Tab

String Attributes Tab

Linestyles Tab

Symbols Tab

Write button

write out, in MapInfo file format, all the string data specified by the Data source.

8.2.11 CivilCAD

Position of option on menu: **File I/O =>Data output =>Civilcad V5**

CivilCAD output is an *unsupported* option supplied to users with the CivilCAD input option. Civilcad input is a chargeable module.

Because of the limitations of the CivilCAD text format, the CivilCAD output option can only output a limited number of *12d Model* data types.

The **Civilcad output** option writes some ***12d Model*** data to a CivilCAD 5.0 text file. Because CivilCAD text can only take points, lines between two points and arcs, only ***12d Model*** data that can output that way can be written out.

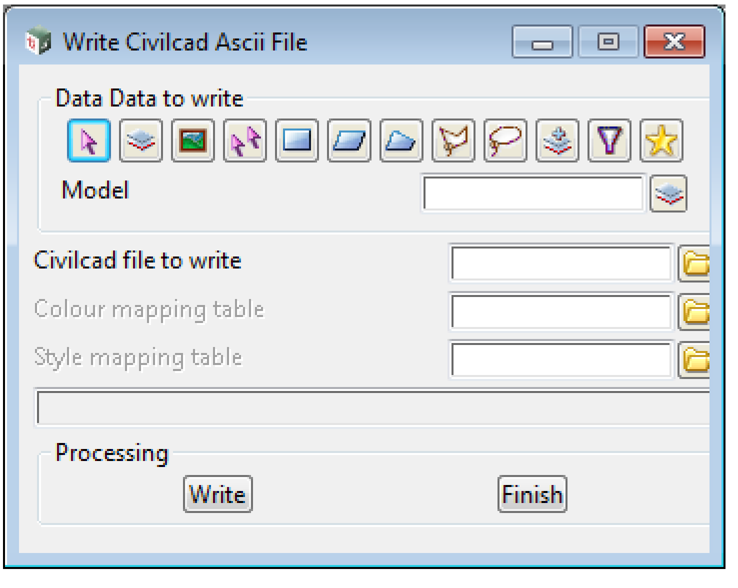
The only ***12d Model*** data types that can be written out to Civilcad text are:

2d strings
3d strings
arcs
circles
polylines
the line work from super strings
text

Note that the following ***12d Model*** data that ***can not*** be output to CivilCAD text:

alignments
super alignments
drainage/sewer strings
pipeline strings
pipe strings
attributes
most super string properties
triangulations
rasters

On selecting the **Civilcad** output option, the **Write CivilCAD Ascii File** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Data source type Model

data selection type - for a full description go to 4.19.3 Data Source.

Data source input

source of data is to be written out to a file.

CivilCAD file to write input *.as5 files

name of the file the model/view is to be written out to. If the file already exists, the data will be appended to the bottom of the file.

Colour mapping table *.as5_colour_map files

mapping of 12d colours to CivilCAD colours

Style mapping table *.as5_style_map files

mapping of 12d linestyles to CivilCAD linestyles

Write button

write the selected data out in CivilCAD format

8.2.12 Geocomp

Position of option on menu: **File I/O =>Data output =>Geocomp**

Geocomp output is a *unsupported* option supplied to users with the Geocomp input option. Geocomp input is a chargeable module.

Because of the limitations of the Geocomp points and strings files, the Geocomp output option can only output a limited number of *12d Model* data types.

The **Geocomp output** option writes some ***12d Model*** data to Geocomp Points and Strings files. Because Geocomp can only take points, lines between two points and arcs, only ***12d Model*** data that can output that way can be written out.

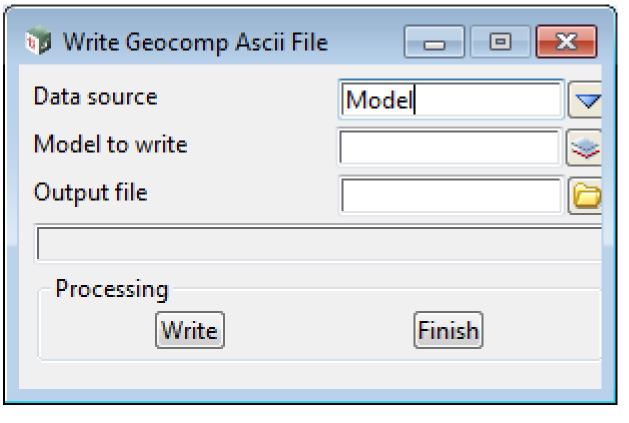
The only ***12d Model*** data types that can be written out to Geocomp points and strings files are:

2d strings
3d strings
arcs
circles
polylines
the line work from super strings
text

Note that the following ***12d Model*** data types that ***can not*** be output to Geocomp:

alignments
super alignments
drainage/sewer strings
pipeline strings
pipe strings
attributes
most super string properties
triangulations
rasters

On selecting the **Geocomp** output option, the **write Geocomp ascii file** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data to write	choice box	Model	Model, View

the type of data to be written out to a file.

Data type source

source of data is to be written out to a file.

Output file file box *.pts/*.str files

name of the file the model/view is to be written out to. If the file already exists, the data will be appended to the bottom of the file.

Write button

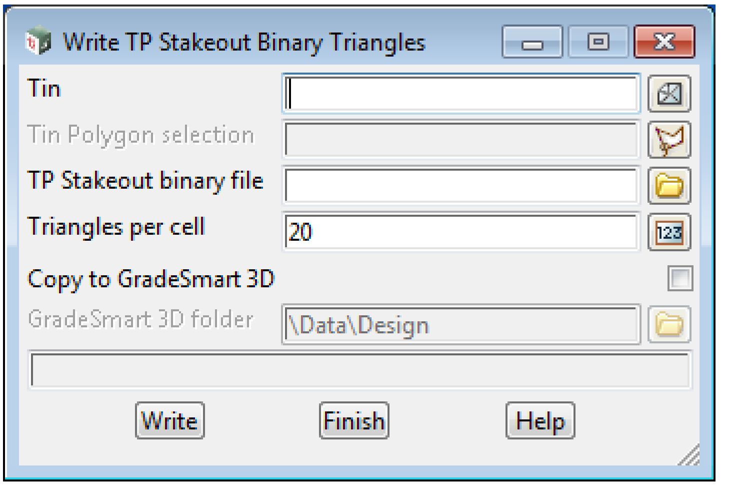
write out Geocomp Points and Strings files the data in the model/view given in the **model/view** *field, to the file given in the* **output file** *field.*

8.2.13 TP Stakeout Triangles Output

Position of option on menu: **File I/O =>Data output =>TP Stakeout triangles**

The *TP Stakeout triangles* option writes out a tin as a TP Stakeout binary file.

On selecting the **TP Stakeout triangles** output option, the **Write TP Stakeout Binary Triangles** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin to write out in TP Stakeout binary format.

Tin polygon selection poly string-select

if selected, only triangles with their centroid inside this string are written out

TP Stakeout binary file input *.tsb

name of the file for the triangles

Triangles per cell

a TP Stakeout parameter

Copy to GradeSmart 3D tick box

if ticked, copy file to GradeSmart folder

GradeSmart 3D folder folder box

folder for GradeSmart files

Write button

write out the tin in TP Stakeout binary format.

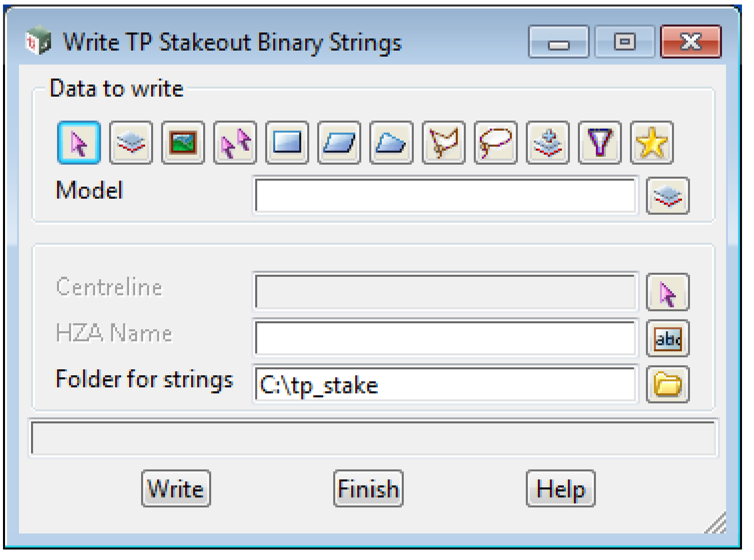
8.2.14 TP Stakeout Strings Output

Position of option on menu: **File I/O =>Data output =>TP Stakeout strings**

Note - this is a chargeable option

The *TPStakeout strings* option writes out selected strings as a TP Stakeout binary file.

Selecting **TP Stakeout strings** brings up the **Write TP Stakeout Binary Strings** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data is to be written out to a file.

Centreline	string select		

centreline string to use as the reference chainage for other strings.

HZA name

Folder for stings	folder		

name of the folder to write out the strings to.

Write	button		

write out the strings in TP Stakeout binary format.

8.2.15 KML Output

Position of option on menu: **File I/O =>Data output =>KML**

The *KML* option writes out selected **super strings** and **tins** in the KML/KMZ file format which is used by Google Earth to load users data into the Google Earth environment.

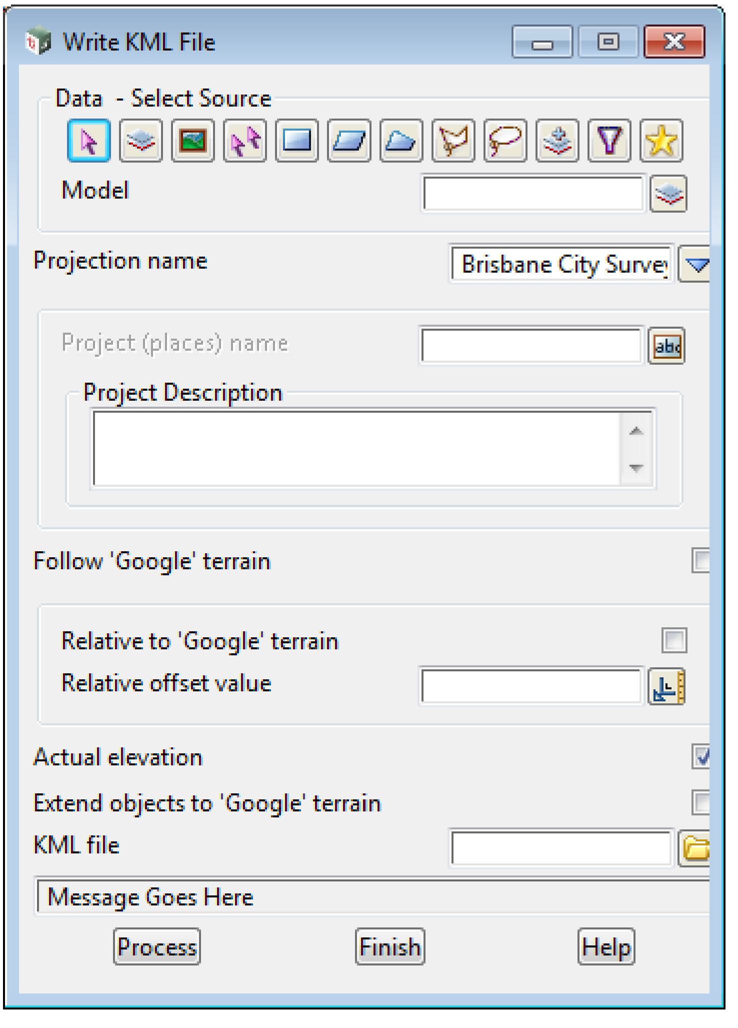
The Google terrain can be significantly different to the terrain from survey data. If the terrain is ticked on in Google Earth it may be higher than the surveyed terrain and obscure the data from 12d.

Google Earth uses the WGS84 Ellipsoid so data must be in a projection based on either the WGS84 or GSR 80 ellipsoids, and the projection for the data known.

Google Earth does not display negative elevations.

If Google Earth does not initialise correctly and centre on the data written from ***12d Model***, selecting a 'model' or 'string' will fit the data to the Google Earth display.

Selecting **KML** brings up the **Write KML** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data is to be written out to a file.
Only super strings and tins are output to the KML file

Projection name projection box

Google Earth uses Latitude and Longitude using the WGS84 spheroid (ellipsoid) to display information in its spatial location. This means that the data to be written to Google Earth should be in a coordinate system that is the same as the a mapping projection and is based on the WGS84 or GSR 80 spheroid. This field is a list of projections created from the users carto.4d file. Select the 'projection name' from the list that matches the coordinate system of the data

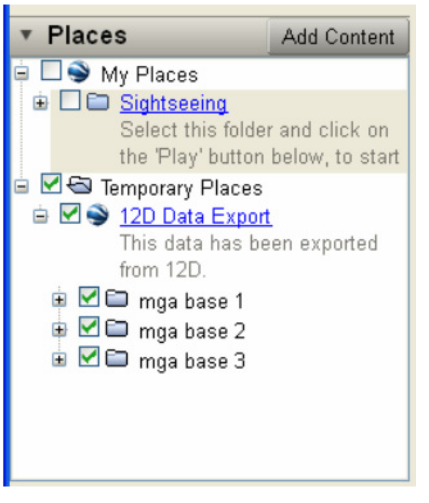
NOTE *- this projection must be based on the WGS84 or GSR80 ellipsoid (for example MGA and NOT AMG).*

Project (Places) name

Enter the name to be shown in the Google Earth 'Places'.
If left blank, the file name from the **KML file field** *is shown in the Google Earth 'Places'.*

Project Description

if non blank, this text will describe the project under the 'Project Name' in Google Earth.



Note:

One and only one of the next three options **Follow Google terrain**, **Relative to Google terrain**, *and* **Actual elevation** *can be ticked on. Ticking one on will untick the other one that was set. You can't untick one except by ticking on a different one.*

Follow Google terrain tick box

If ticked, the elevations of the data in 12d are ignored and the objects are 'clamped' to the terrain in Google Earth.

Relative to Google terrain tick box

If ticked, the elevations of the 12d data are ignored and the objects are 'clamped' to the terrain in Google Earth ***plus*** *the value given in the* **Relative offset value** *field.*

Relative offset value

if **Relative to Google terrain is** *ticked, the data is offset by this amount above the Google terrain. If this value is zero then it is the same as* **Follow Google terrain**.

Actual elevation tick box

if ticked, the z values from the 12d are used to display the objects in Google Earth

Extend objects to Google terrain tick box

if ticked the objects are displayed in Google Earth as a shape that extends from the Google terrain to the object

KML file file box *.kmz files

name of the kmz file to be written. The kml file is an xml style of file and the kmz file is the kml file in the 'zip' compression format.

Process button

write out the data in KML format.

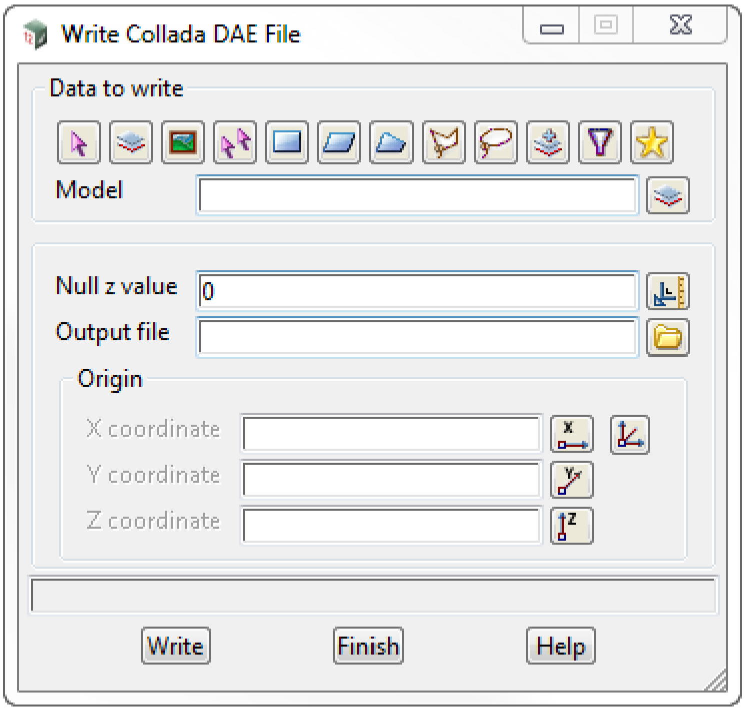
8.2.16 Export DAE

Position of option on menu: **File I/O =>Data output => DAE**

The Collada Digital Asset Exchange (DAE) file format is a published format for writing out 3d shapes for interchange between various graphics packages.

Trimeshes, extrusions, pipes and drainage strings can be written to a DAE file.

Selecting **DAE** brings up the **Write Collada DAE File** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be written out to a DAE file.

Null z value	real box	0	measures menu

value to use in place of any null z values.

Output file	file box		folder browse

name of the file to write the DAE information to.

Origin	point select box		

.

Write	button		

write out the DAE file.

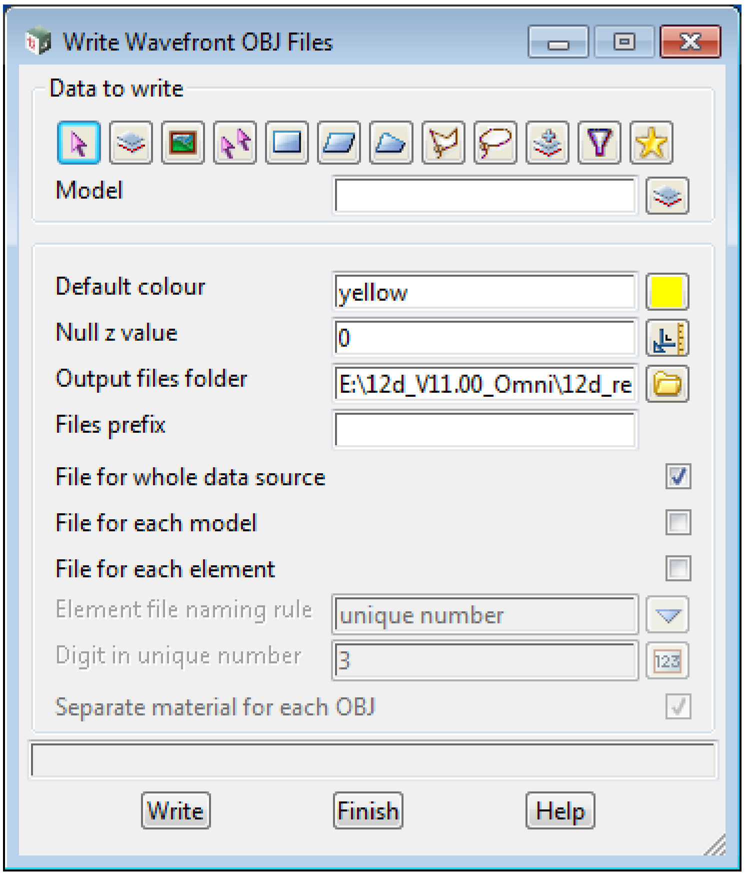
8.2.17 Export OBJ

Position of option on menu: **File I/O =>Data output => OBJ**

The OBJ file is a published format from Wavefront for writing out 3d shapes.

Trimeshes, extrusions, pipes and drainage strings can be written to the one OBJ file, or separate OBJ files.

Selecting **OBJ** brings up the **Write Wavefront OBJ Files** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	
data selection type - for a full description go to 4.19.3 Data Source.			
Data source	input		
source of data to be written out to a file.			
Default colour	colour box		
default colour for the OBJ objects.			
Null z value	real box	0	measures menu
value to use in place of any null z values.			
Output files folder	folder box		folder browse
name of the folder to write the obj files to.			

Files prefix text box

prefix to use in front of any names for the obj files.

File for whole data source tick box

*if **ticked**, all the 3d shapes in the data source are written out as the one obj. The name of the file is* **File prefix.**

File for each model tick box

*if **ticked**, an obj file is written out for each model, and all the 3d shapes in the one model are written out as the one obj. The name of the file is the model name prefixed by* **File prefix.**

File for each model tick box

*if **ticked**, a separate obj file is written out for each model, and all the 3d shapes in the one model are written out as the one obj.*

File for each element tick box

*if **ticked**, a separate obj file is written out for 3d shapes in the data source.*

Element naming rule choice box unique number
name followed by unique number
model name followed by unique number

*if **unique number**, the* **File prefix** *followed by a unique number for each shape, is used for the name of each obj file.*

*If **name followed by unique number**, the* **File prefix** *followed by the string name of the 3d shape, followed by a unique number when two string names are the same, is used for the name of each obj file.*

*If **model name followed by unique number**, the* **File prefix** *followed by the name of the model that the 3d shape is in, followed by a unique number for each shape in the model, is used for the name of each obj file.*

Digits in unique number integer box

number of digits in the unique number. The number will be 0 filled if it has less than this number of digits.

Write button

write out the data in OBJ format.

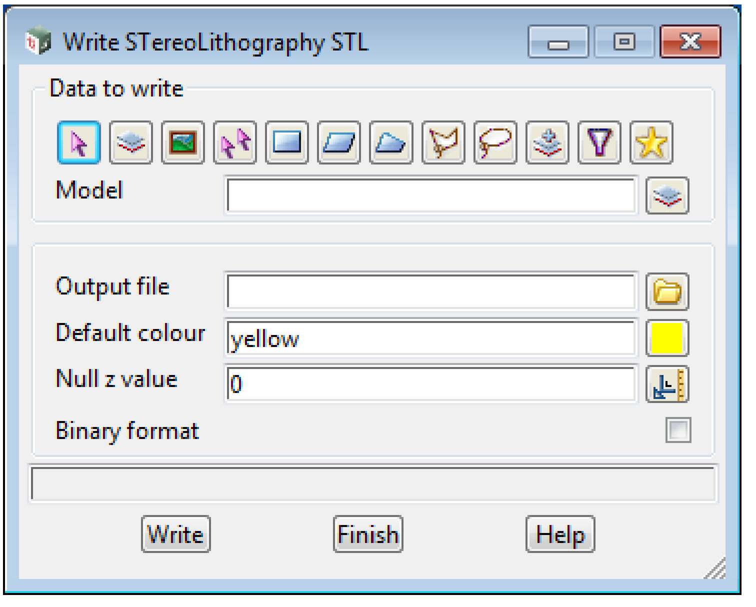
8.2.18 Export STL

Position of option on menu: **File I/O =>Data output =>STL**

The STL file is a format for driving 3D printers for stereolithography. The STL format only accepts solid shapes.

Trimeshes, extrusions, pipes and drainage strings can be written to an STL file.

Selecting **Export STL** brings up the **Write StereoLithography STL** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	
data selection type - for a full description go to 4.19.3 Data Source.			
Data source	input		
source of data to be written out to an STL file.			
Output file	folder box		*.stl files
name of the file for the STL information to be written out to.			
Null z value	real box	0	measures menu
value to use in place of any null z values.			
Binary format	tick box		
*if **ticked**, the binary STL format is written out.* *If **not ticked**, the ASCII STL format is used.*			
Default colour	colour box		
colour for the STL objects.			
Write	button		
write out the STL file.			

8.2.19 IFC Output

Position of menu: **File I/O =>Data output =>IFC**

Recommended related reading: 7.6.10 Tags.

The **IFC Output** options write out ***12d Model*** data in the IFC STEP format, version **IFC 2x3**.

The **Industry Foundation Classes** (IFC) data model is intended to describe building and construction industry data.

It is a platform neutral, open file format specification that is not controlled by a single vendor or group of vendors.

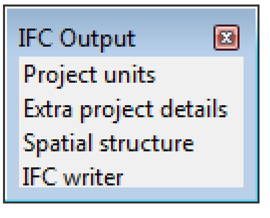
It is an object-based file format with a data model developed by **buildingSMART** (formerly the International Alliance for Interoperability, IAI) to facilitate interoperability in the architecture, engineering and construction (AEC) industry, and is a commonly used collaboration format in **Building Information Modeling** (BIM) based projects.

For more information on BIMs, see 33 BIM and http://www.buildingsmart-tech.org

For more information on IFC's, see 8.2.19.1 IFCs.

The data in ***12d Model*** that can be written out to IFC's includes super strings, TINs, drainage strings and trimeshes.

The **IFC Output** walk-right menu is



For the options:

Project units	8.2.19.2 Project Units
Extra project details	8.2.19.3 Extra Project Details
Special tag structure	8.2.19.4 Defining and Assigning Spatial Structures
IFC Writer	8.2.19.5 IFC 2x3 Writer

For general information on IFCs, see 8.2.19.1 IFCs.

For information on what ***12d Model*** objects are written out to IFCs, and how they are written, see 8.2.19.1.3 Representation of 12d Objects as IFCs.

8.2.19.1 IFCs

The **IFC Output** options write out ***12d Model*** data in the IFC STEP format where IFC stands for **Industry Foundation Classes**.

The IFC data model is intended to describe building and construction industry data, and to date, only covers buildings although there are committees currently looking at extending it to other infrastructure areas such as bridges, roads, railways.

It is a platform neutral, open file format specification that is not controlled by a single vendor or group of vendors.

It is an object-based file format with a data model developed by **buildingSMART** (formerly the International Alliance for Interoperability, IAI) to facilitate interoperability in the architecture, engineering and construction (AEC) industry, and is a commonly used collaboration format in **Building Information Modeling** (BIM) based projects. See http://www.buildingsmart-tech.org.

The IFC model specification is open and available. It is registered by **ISO** and is an official International Standard ISO 16739:2013. See http://www.iso.org/iso/catalogue_detail.htm?csnumber=51622 and http://www.buildingsmart-tech.org/specifications/ifc-releases/summary.

It should be noted that from the ISO site, the following are outside the scope of ISO 16739:2013:

- exchange format definitions outside of the domain of construction and facility maintenance;
- project structure and component breakdown structures **outside of building engineering**;
- behavioural aspects of components and other information items.

Before trying to write out an IFC file, ***12d Model*** Users must be aware of how the IFC data model is structured so that the data to be written out is correct. And how this data is to be structured may vary from client to client wanting their data in an IFC file.

For a general overview of IFC's, see http://en.wikipedia.org/wiki/Industry_Foundation_Classes

The IFC data model is very rich, as you would expect it to be to cover every aspect of a building, but we will only be using a small subset of its capabilities.

All the information and data is contained in **ifcProject** and there is only one of these.

All the data in a ***12d Model*** project can only go out to the one **ifcProject**.

We will mainly be using **ifcProduct** which represents occurrences in space such as physical building elements and spatial locations.

ifcProduct is the **base class** for all physical objects and is subdivided into:

(a) spatial items

Spatial items include ifcSite, ifcBuilding, ifcBuildingStorey and ifcSpace.

The spatial structure elements are linked together, and to the *IfcProject*, by using the objectified relationships *IfcRelContainedInSpatialStructure* and *IfcRelAggregates*.

(b) physical elements

Physical building elements include ifcWall, ifcBeam, ifcDoor, ifcWindow, ifcStair *etc*.

(c) structural analysis items

(d) other concepts.

Products may have associated materials, shape representations, and placements in space.

For more information, wee:

8.2.19.1.1 Spatial Structures

Within the **ifcProject**, an overall Spatial Structure is defined which has a strict hierarchical structure.

At the top level is the **Project**.

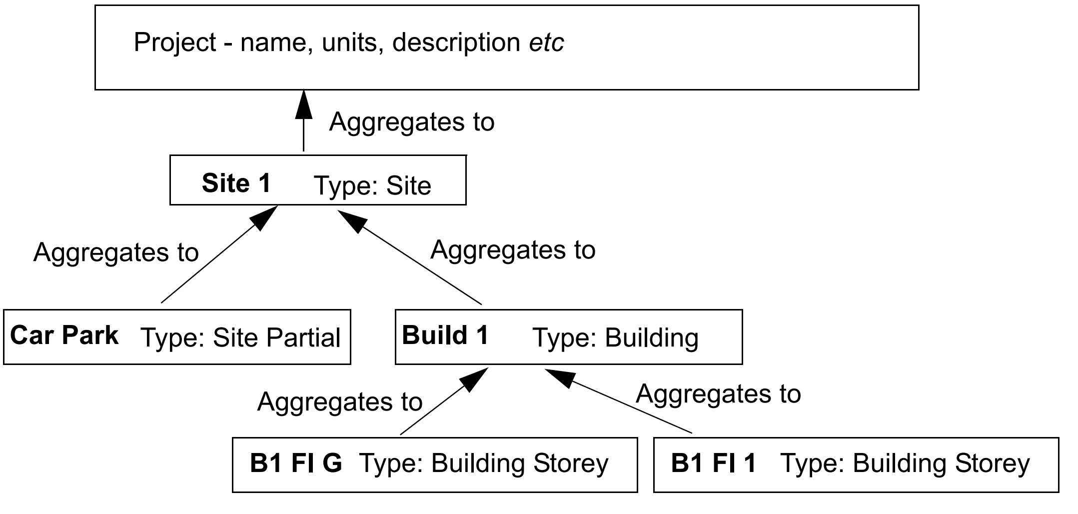
Then within this **Project** are **Sites**, and each **instance** of a **Site** has a **unique name**.

A Site can contain other Sites (called Site Partials) or Buildings, or Building Storeys or Spaces or Products.

A Building can contain Building Storeys, Spaces or Products.

Each instance of Site, Site Partial, Building, Building Storey, Space or Product has a unique name and is directly contained in only one other spatial structure.

The hierarchy of the overall Spatial Structure is specified by defining when each spatial structure **B** is directly contained in another spatial structure **A**. It is expresses as the spatial structure **B** **aggregates to** the spatial structure **A**.



So we only need to refer to a spatial structure by name and all the spatial structures contained beneath it are also uniquely specified.

In ***12d Model*** the Spatial Structure is constructed by the option

File =>Data output =>IFC => Spatial structure

which brings up the panel **IFC Spatial Structure**.

The spatial structures for the project, including the ***Aggregates to***, are defined in the **Define** tab of the panel, and are stored as project attributes. See 8.2.19.4 Defining and Assigning Spatial Structures.

An IFC element is assigned to a spatial structure. Basic types of IFC elements are

(a) building element

(b) opening element

(c) furnishing element

(d) distribution element (including heating, ventilation, air conditioning, electrical and equipment elements)

(e) transportation element

12d Model data that is to be written out to an IFC file are written as **physical** elements and as such must belong to one spatial structure.

12d Model data is assigned a spatial structure by either defined a spatial structure for a whole model and/or for individual strings or trimeshes. The spatial structure for tins is taken to be that of the spatial structure of the model that the tin is in.

The **IFC Writer** panel writes out all the data of a user given spatial structure to a file. See 8.2.19.5 IFC 2x3 Writer.

Continue to the next section 8.2.19.1.2 Some IFC 2x3 Definitions or return to 8.2.19.1 IFCs or 8.2.19 IFC Output.

8.2.19.1.2 Some IFC 2x3 Definitions

For some information on definitions of some of the IFC 2x3 items used when writing the IFC file, see:

8.2.19.1.2.1 ifcProject
8.2.19.1.2.2 ifcSite
8.2.19.1.2.3 ifcElement
8.2.19.1.2.4 ifcSpatialStructureElement
8.2.19.1.2.5 ifcBuildingElement
8.2.19.1.2.6 ifcBuildingElementProxy
8.2.19.1.2.7 ifcFlowStorageDevice
8.2.19.1.2.8 ifcFlowStorageDeviceType
8.2.19.1.2.9 ifcFlowSegment
8.2.19.1.2.10 ifcFlowSegmentType
8.2.19.1.2.11 ifcRelContainedInSpatialStructure
8.2.19.1.2.12 ifcRelAggregates
8.2.19.1.2.13 ifcProduct

Do not be alarmed if the definitions need to be read a number of times to get your head around them. Remember they are only an extract of part of the definition and there are many more definitions that make up IFC's.

For the complete information on IFC's, see http://www.buildingsmart-tech.org/specifications

Or continue to the section 8.2.19.1.3 Representation of 12d Objects as IFCs or return to 8.2.19.1 IFCs or 8.2.19 IFC Output.

8.2.19.1.2.1 ifcProject

Definition from IAI: *The undertaking of some design, engineering, construction, or maintenance activities leading towards a product. The project establishes the context for information to be exchanged or shared, and it may represent a construction project but does not have to.*

All the information and data is contained in **ifcProject** and there is only one of these.

The **ifcProject** has information such as

(a) the default units used

For setting the **ifcProject** units in ***12d Model***, see 8.2.19.2 Project Units.

(b) descriptive information about the project

For setting the **ifcProject** information in ***12d Model***, see 8.2.19.3 Extra Project Details.

Within the **ifcProject**, a spatial structure is defined and it is a strict hierarchical structure. See 8.2.19.1.1 Spatial Structures.

Continue to 8.2.19.1.2.2 ifcSite or return to 8.2.19.1.2 Some IFC 2x3 Definitions or 8.2.19.1 IFCs.

8.2.19.1.2.2 ifcSite

Definition from ISO 6707-1:1989: *Area where construction works are undertaken.*

Definition from IAI: *A defined area of land, possibly covered with water, on which the project construction is to be completed. A site may be used to erect building(s) or other AEC products.*

The geometrical placement of the site, defined by the *IfcLocalPlacement*, shall be always relative to the spatial structure element, in which this site is included, or absolute, i.e. to the world coordinate system, as established by the geometric representation context of the project. The world coordinate system, established at the *IfcProject.RepresentationContexts*, may include a definition of the true north within the XY plane of the world coordinate system.

An IFC project may span over several connected or disconnected sites. Therefore **site complex** provides for a collection of sites included in a project.

A site can also be decomposed in parts, where each part defines a **site section**.

Whether it is a site, site complex or site section is defined by the Composition Type attribute of the supertype ***IfcSpatialStructureElement*** which is interpreted as follow:

If the attribute value is **COMPLEX** then it is a **site complex**

If the attribute value is **ELEMENT** then it is a **site**

If the attribute value is **PARTIAL** then it is a **site section** (**site partial)**.

Note that in ***12d Model***, you can currently only have a **site** and **site partial**.

Continue to 8.2.19.1.2.3 ifcElement or return to 8.2.19.1.2 Some IFC 2x3 Definitions or 8.2.19.1 IFCs.

8.2.19.1.2.3 ifcElement

Definition from IAI: *Generalization of all components that make up an AEC product. Those elements can be logically contained by a spatial structure element that constitutes a certain level within a project structure hierarchy (e.g., site, building, storey or space). This is done by using the IfcRelContainedInSpatialStructure relationship*.

Elements are physically existent objects, although they might be void elements, such as holes.

EXAMPLEs of elements in a building construction context are walls, floors, windows and recesses.

There are various type of Elements derived from ifcElement including ifcBuildingElment, idfFurnishingElment, ifcElelectricalElment and ifcBuildingElmentProxy (see 8.2.19.1.2.6 ifcBuildingElementProxy).

Note that in ***12d Model***, most data is written to the IFC file as *ifcBuildingElmentProxy*.

Continue to 8.2.19.1.2.4 ifcSpatialStructureElement or return to 8.2.19.1.2 Some IFC 2x3 Definitions or 8.2.19.1 IFCs.

8.2.19.1.2.4 ifcSpatialStructureElement

Definition from IAI: *A spatial structure element (**IfcSpatialStructureElement**) is the generalization of all spatial elements that might be used to define a spatial structure. That spatial structure is often used to provide a project structure to organize a building project*.

A spatial project structure might define as many levels of decomposition as necessary for the building project. Elements within the spatial project structure are:

(a) site as *IfcSite*

(b) building as *IfcBuilding*

(c) storey as *IfcBuildingStorey*

(d) space as *IfcSpace*

or

(e) aggregations or parts thereof.

The Composition Type declares an element to be either an element itself, or an aggregation (complex) or a decomposition (part). The interpretation of these types is given at each subtype of *IfcSpatialStructureElement*. For example see 8.2.19.1.2.2 ifcSite.

The *IfcRelAggregates* is defined as an 1-to-many relationship and used to establish the relationship between exactly two levels within the spatial project structure.

Finally the highest level of the spatial structure is assigned to *IfcProject* using the *IfcRelAggregates*. See 8.2.19.1.2.12 ifcRelAggregates.

Continue to 8.2.19.1.2.5 ifcBuildingElement or return to 8.2.19.1.2 Some IFC 2x3 Definitions or 8.2.19.1 IFCs.

8.2.19.1.2.5 ifcBuildingElement

Definition from ISO 6707-1:1989: *Major functional part of a building, examples are foundation, floor, roof, wall*.

Definition from IAI: *The building element comprises all elements that are primarily part of the construction of a building, i.e., its structural and space separating system*.

EXAMPLEs of building elements are walls, beams, or doors, they are all physically existent and tangible things.

Continue to 8.2.19.1.2.6 ifcBuildingElementProxy or return to 8.2.19.1.2 Some IFC 2x3 Definitions or 8.2.19.1 IFCs.

8.2.19.1.2.6 ifcBuildingElementProxy

Definition from IAI: *The **IfcBuildingElementProxy** is a proxy definition that provides the same functionality as an **IfcBuildingElement**, but without having a defined meaning of the special type* *of building element, it represents.*

NOTE1 The *IfcBuildingElementProxy* should be used to exchange special types of building elements for which the current IFC Release does not yet provide a semantic definition.

NOTE2 The *IfcBuildingElementProxy* can also be used to represent building elements for which the participating applications can not provide additional semantic classification.

Continue to 8.2.19.1.2.7 ifcFlowStorageDevice or return to 8.2.19.1.2 Some IFC 2x3 Definitions or 8.2.19.1 IFCs.

8.2.19.1.2.7 ifcFlowStorageDevice

The distribution flow element **IfcFlowStorageDevice** defines the occurrence of a device that participates in a distribution system and is used for temporary storage of a fluid such as a liquid or a gas (e.g., tank) or the voltage potential induced by the induced electron flow (such as a battery).

Its type is defined by ***IfcFlowStorageDeviceType*** or its subtypes. See 8.2.19.1.2.8 ifcFlowStorageDeviceType.

Continue to 8.2.19.1.2.8 ifcFlowStorageDeviceType or return to 8.2.19.1.2 Some IFC 2x3 Definitions or 8.2.19.1 IFCs.

8.2.19.1.2.8 ifcFlowStorageDeviceType

The element type **IfcFlowStorageDeviceType** defines a list of commonly shared property set definitions of a flow storage device and an optional set of product representations. It is used to define a flow storage device specification (the specific product information that is common to all occurrences of that product type).

A flow storage device is a device used for the temporary storage of a fluid (such as a tank) or the voltage potential induced by the induced electron flow (such as a battery). Flow storage types (or the instantiable subtypes) may be exchanged without being already assigned to occurrences.

Continue to 8.2.19.1.2.9 ifcFlowSegment or return to 8.2.19.1.2 Some IFC 2x3 Definitions or 8.2.19.1 IFCs.

8.2.19.1.2.9 ifcFlowSegment

The distribution flow element ***IfcFlowSegment*** defines the occurrence of a segment of a flow distribution system.

The ***IfcFlowSegment*** defines a particular occurrence of a segment inserted in the spatial context of a project. The parameters defining the type of the segment and/or its shape are defined by the ***IfcFlowSegmentType.***

IFC2x4 CHANGE This entity has been deprecated for instantiation and will become ABSTRACT in a future release; new subtypes should now be used instead.

Material Use Definition

The material of the ***IfcFlowSegment*** is defined using one of the following entities:

IfcMaterialProfileSetUsage: for parametric segments, this defines the cross section and alignment to the 'Axis' representation, from which the 'Body' representation may be generated.

IfcMaterialProfileSet: for non-parametric segments (having fixed length or path), this may define the cross section for analysis purposes, however the 'Body' representation is independently generated.

IfcMaterialConstituentSet: for elements containing multiple materials where profiles are not applicable, this indicates materials at named parts.

IfcMaterial: for elements comprised of a single material where profiles are not applicable, this indicates the material.

The material is attached by the RelatingMaterial attribute on the IfcRelAssociatesMaterial relationship. It is accessible by the HasAssociations inverse attribute. Material information can also be given at the IfcFlowSegmentType, defining the common attribute data for all occurrences of the same type. Standard names and material types are defined at subtypes.

Geometry Use Definition

Standard representations are defined at the supertype ***IfcDistrubutionFlowElement***. For parametric flow segments where IfcMaterialProfileSetUsage is defined and an 'Axis' representation is defined, then the 'Body' representation may be generated using the 'SweptSolid' or 'AdvancedSweptSolid' representation types by sweeping the profile(s) along the axis.

Continue to 8.2.19.1.2.10 ifcFlowSegmentType or return to 8.2.19.1.2 Some IFC 2x3 Definitions or 8.2.19.1 IFCs.

8.2.19.1.2.10 ifcFlowSegmentType

The element type ***IfcFlowSegmentType*** defines a list of commonly shared property set definitions of a flow segment and an optional set of product representations.

It is used to define a flow segment specification (the specific product information, that is common to all occurrences of that product type).

A *flow segment type* is used to define the common properties of a flow segment that may be applied to many occurrences of that type. A flow segment is a section of a distribution system, such as a duct, pipe, or conduit, that typically has only two ports. Flow segment types (or the instantiable subtypes) may be exchanged without being already assigned to occurrences.

Material Use Definition

The material of the *IfcDistributionFlowSegmentType* is defined using one of the following entities:

IfcMaterialProfileSet: This defines the material cross section which may be used to generate the 'Body' representation at occurrences (for parametric definitions not having representation), or for analysis purposes.

IfcMaterialConstituentSet: For elements containing multiple materials where profiles are not applicable, this indicates materials at named aspects.

IfcMaterial: For elements comprised of a single material where profiles are not applicable, this indicates the material.

Continue to 8.2.19.1.2.11 ifcRelContainedInSpatialStructure or return to 8.2.19.1.2 Some IFC 2x3 Definitions or 8.2.19.1 IFCs.

8.2.19.1.2.11 ifcRelContainedInSpatialStructure

Definition from IAI: *This objectified relationship, **IfcRelContainedInSpatialStructure**, is used to assign elements to a certain level of the spatial project structure. Any element can only be assigned once to a certain level of the spatial structure. The question, which level is relevant for which type of element, can only be answered within the context of a particular project and might vary within the various regions*.

EXAMPLE A multi-storey space is contained (or belongs to) the building storey at which its ground level is, but it is referenced by all the other building storeys, in which it spans. A lift shaft might be contained by the basement, but referenced by all storeys, through which it spans.

The **containment relationship** of an element within a spatial structure has to be a hierarchical relationship, **an element can only be contained within a single spatial structure element**.

The **reference relationship** between an element and the spatial structure may not be hierarchical, i.e. an element can reference many spatial structure elements.

NOTE The reference relationship is expressed by *IfcRelReferencedInSpatialStructure*.

Predefined spatial structure elements to which elements can be assigned are

(a) site as *IfcSite* . See 8.2.19.1.2.2 ifcSite.

(b) building as *IfcBuilding*

(c) storey as *IfcBuildingStorey*

(d) space as *IfcSpace*

Continue to 8.2.19.1.2.12 ifcRelAggregates or return to 8.2.19.1.2 Some IFC 2x3 Definitions or 8.2.19.1 IFCs.

8.2.19.1.2.12 ifcRelAggregates

Definition from IAI: *The aggregation relationship **IfcRelAggregates** is a special type of the general composition/decomposition (or whole/part) relationship IfcRelDecomposes. The aggregation relationship can be applied to all subtypes of object.*

Some further specializations of decomposition may imply additional constraints and meanings, such as the requirement of aggregates to represent physical containment. In cases of physical containment the representation (within the same representation context) of the whole can be taken from the sum of the representations of the parts.

EXAMPLE: A roof is the aggregation of the roof elements, such as roof slabs, rafters, purlins, *etc*. Within the same representation context, e.g. the detailed geometric representation, the shape representation of the roof is given by the shape representation of its parts

Decompositions imply a dependency, i.e. the definition of the whole depends on the definition of the parts and the parts depend on the existence of the whole. The behaviour that is implied from the dependency has to be established inside the applications.

Continue to 8.2.19.1.2.13 ifcProduct or return to 8.2.19.1.2 Some IFC 2x3 Definitions or 8.2.19.1 IFCs.

8.2.19.1.2.13 ifcProduct

Definition from IAI: *Any object, or any aid to define, organize and annotate an object, that relates to a geometric or spatial context. Subtypes of IfcProduct usually hold a shape representation and a local placement within the project structure*.

This includes manufactured, supplied or created objects (referred to as **elements**) for incorporation into an AEC/FM project. This also includes objects that are created indirectly by other products, as spaces are defined by bounding elements. Products can be designated for permanent use or temporary use, an example for the latter is formwork. Products are defined by their properties and representations.

In addition to **physical products** (covered by the subtype 8.2.19.1.2.3 ifcElement) and **spatial items** (covered by the subtype 8.2.19.1.2.4 ifcSpatialStructureElement) the *IfcProduct* also includes non-physical items, that relate to a geometric or spatial contexts, such as grid, port, annotation, structural actions, *etc*.

Continue to 8.2.19.1.2.14 ifcPropertySet or return to 8.2.19.1.2 Some IFC 2x3 Definitions or 8.2.19.1 IFCs.

8.2.19.1.2.14 ifcPropertySet

Definition from IAI: *The IfcPropertySet defines all dynamically extensible properties. The* *property set is a container class that holds properties within a property tree. These properties are interpreted according to their name attribute.*

Property sets, defining a particular type of object, can be assigned an object type (*IfcTypeObject*). Property sets are assigned to objects (*IfcObject*) through an objectified relationship (*IfcRelDefinedByProperties*). If the same set of properties applies to more than one object, it should be assigned by a single instance of *IfcRelDefinedByProperties* to a set of related objects. Those property sets are referred to as shared property sets.

Use Definition

Instances of *IfcPropertySet* are used to assign named sets of individual properties (complex or single properties). Each individual property has a significant name string. Some property sets have predefined instructions on assigning those significant name. The naming convention "Pset_Xxx" applies to those property sets and shall be used as the value to the *Name* attribute.

In addition any user defined property set can be captured, those property sets shall have a *Name* value not including the **Pset_** prefix.

Continue to 8.2.19.1.2.15 IfcPropertySingleValue or return to 8.2.19.1.2 Some IFC 2x3 Definitions or 8.2.19.1 IFCs.

8.2.19.1.2.15 IfcPropertySingleValue

Definition from IAI: A property with a single value (*IfcPropertySingleValue*) defines a property object which has a single (numeric or descriptive) value assigned. It defines a property - single value combination for which the property name, the value with measure type (and optionally the unit) is given.

The unit is handled by the *Unit* attribute:

If the *Unit* attribute is not given, then the unit is already implied by the type of *IfcMeasureValue* or *IfcDerivedMeasureValue*. The associated unit can be found at the *IfcUnitAssignment* globally defined at the project level (*IfcProject.UnitsInContext*).

If the *Unit* attribute is given, then the unit assigned by the *Unit* attribute overrides the globally assigned unit.

Continue to 8.2.19.1.2.18 IfcCircleHollowProfileDef or return to 8.2.19.1.2 Some IFC 2x3 Definitions or 8.2.19.1 IFCs.

8.2.19.1.2.16 IfcShapeRepresentation

Definition from ISO/CD 10303-42:1992: The shape representation is a specific kind of representation that represents a shape.

Definition from IAI: The ***IfcShapeRepresentation*** represents the concept of a particular geometric representation of a product or a product component within a specific geometric representation context.

The inherited attribute ***RepresentationType*** is used to define the geometric model used for the shape representation, the inherited attribute ***RepresentationIdentifier*** is used to denote the part of the representation captured by the ***IfcShapeRepresentation*** (e.g. Axis, Body, *etc*.).

Several representation types for shape representation are included as predefined types:

Curve2D 2 dimensional curves

GeometricSet points, curves, surfaces (2 or 3 dimensional)

GeometricCurveSet points, curves (2 or 3 dimensional)

Annotation2D points, curves (2 or 3 dimensional), hatches and text (2 dimensional)

BoundingBox simplistic 3D representation by a bounding box

SectionedSpine cross section based representation of a spine curve and planar cross sections. It can represent a surface or a solid and the interpolations of the between the cross sections is not defined

MappedRepresentation representation based on mapped item(s), referring to a representation map. Note: it can be seen as an inserted block reference. The shape representation of the mapped item has a representation type declaring the type of its representation items.

SurfaceModel face based and shell based surface model

SolidModel including swept solid, Boolean results and Brep bodies. More specific types are:

SweptSolid swept area solids, by extrusion and revolution

Brep faceted Brep's with and without voids

CSG Boolean results of operations between solid models, half spaces and Boolean results

Clipping Boolean differences between swept area solids, half spaces and Boolean results

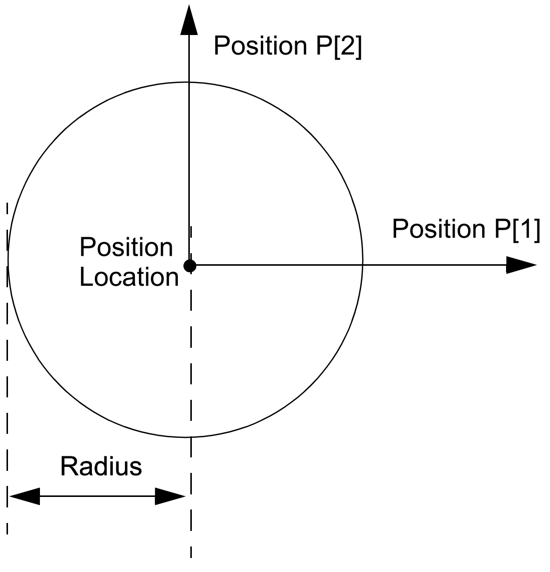
AdvancedSweptSolid swept area solids created by sweeping a profile along a directrix

NOTE The definition of this entity relates to the STEP entity shape_representation. Please refer to ISO/IS 10303-41:1994 for the final definition of the formal standard.

Continue to 8.2.19.1.2.17 IfcCircleProfileDef or return to 8.2.19.1.2 Some IFC 2x3 Definitions or 8.2.19.1 IFCs.

8.2.19.1.2.17 IfcCircleProfileDef

Definition from IAI: The ***IfcCircleProfileDef*** defines a circle as the profile definition used by the swept surface geometry or by the swept area solid. It is given by its ***Radius*** attribute and placed within the 2D position coordinate system, established by the ***Position*** attribute.



Position
The parameterized profile defines its own position coordinate system. The underlying coordinate system is defined by the swept surface or swept area solid that uses the profile definition. It is the xy plane of either:

IfcSweptSurface.Position

IfcSweptAreaSolid.Position

or in case of sectioned spines the xy plane of each list member of IfcSectionedSpine.CrossSectionPositions.

By using offsets of the position location, the parameterized profile can be positioned centric (using x,y offsets = 0.), or at any position relative to the profile. Explicit coordinate offsets are used to define cardinal points (e.g. upper-left bound).

Parameter
The ***Position*** attribute defines the 2D position coordinate system of the circle.

The ***Radius*** attribute defines the radius of the circle.

ifc Attribute Definitions:

Radius: The radius of the circle.

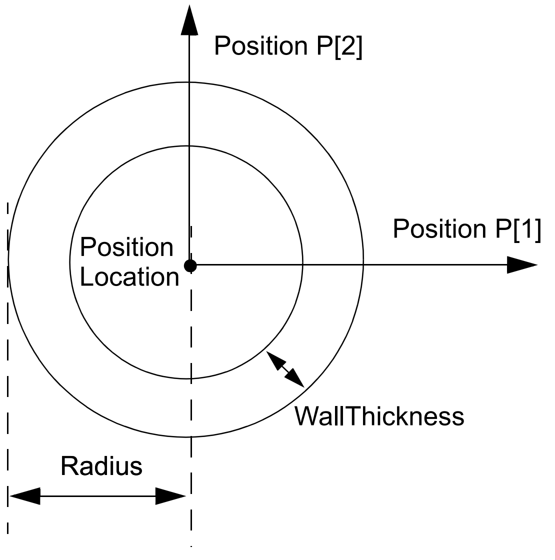
Continue to 8.2.19.1.2.18 IfcCircleHollowProfileDef or return to 8.2.19.1.2 Some IFC 2x3 Definitions or 8.2.19.1 IFCs.

8.2.19.1.2.18 IfcCircleHollowProfileDef

Definition from IAI: The ***IfcCircleHollowProfileDef*** defines a section profile that provides the defining parameters of a circular hollow section (tube) to be used by the swept area solid.

Its parameters and orientation relative to the position coordinate system are according to the following illustration.

The centre of the position coordinate system is in the profile's centre of the bounding box (for symmetric profiles identical with the centre of gravity).



Position
The parameterized profile defines its own position coordinate system. The underlying coordinate system is defined by the swept area solid that uses the profile definition. It is the xy plane of:

IfcSweptAreaSolid.Position

by using offsets of the position location, the parameterized profile can be positioned centric (using x,y offsets = 0), or at any position relative to the profile. Explicit coordinate offsets are used to define cardinal points (e.g. upper-left bound).

Parameter
The parameterized profile is defined by a set of parameter attributes, see attribute definition below.

ifc Attribute Definitions:

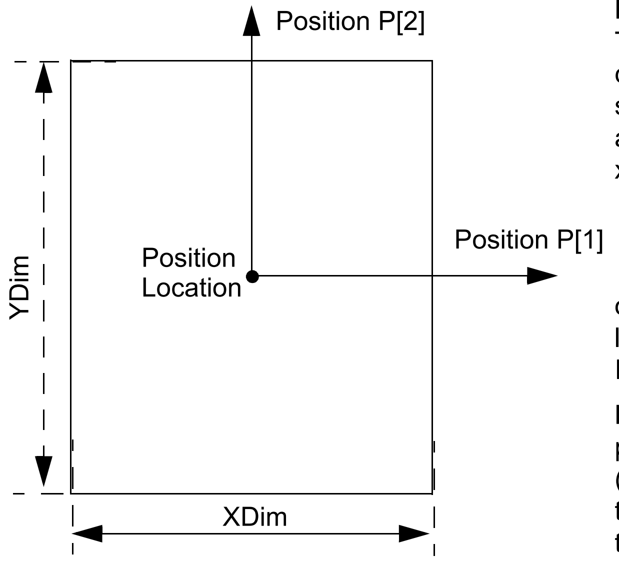
WallThickness: Thickness of the material. It is the difference between the outer and inner radius.

The wall thickness must be smaller than the radius

Continue to 8.2.19.1.2.19 IfcRectangleProfileDef or return to 8.2.19.1.2 Some IFC 2x3 Definitions or 8.2.19.1 IFCs.

8.2.19.1.2.19 IfcRectangleProfileDef

Definition from IAI: The ***IfcRectangleProfileDef*** defines a rectangle as the profile definition used by the swept surface geometry or the swept area solid. It is given by its X extent and its Y extent, and placed within the 2D position coordinate system, established by the Position attribute. It is placed centric within the position coordinate system.



Position
The parameterized profile defines its own position coordinate system. The underlying coordinate system is defined by the swept surface or swept area solid that uses the profile definition. It is the xy plane of either:

IfcSweptSurface.Position

IfcSweptAreaSolid.Position

or in case of sectioned spines the xy plane of each list member of IfcSectionedSpine.CrossSectionPositions.

By using offsets of the position location, the parameterized profile can be positioned centric (using x,y offsets = 0.), or at any position relative to the profile. Explicit coordinate offsets are used to define cardinal points (e.g. upper-left bound).

Parameter
The ***IfcRectangleProfileDef*** is defined within the position coordinate system, where the **XDim** defines the length measure for the length of the rectangle (half along the positive x-axis) and the **YDim** defines the length measure for the width of the rectangle (half along the positive y-axis).

ifc Attribute Definitions:

XDim: The extent of the rectangle in the direction of the x-axis.

YDim: The extent of the rectangle in the direction of the y-axis.

Continue to the next section 8.2.19.1.2.20 IfcRectangleHollowProfileDef or return to 8.2.19.1.2 Some IFC 2x3 Definitions or 8.2.19.1 IFCs.

8.2.19.1.2.20 IfcRectangleHollowProfileDef

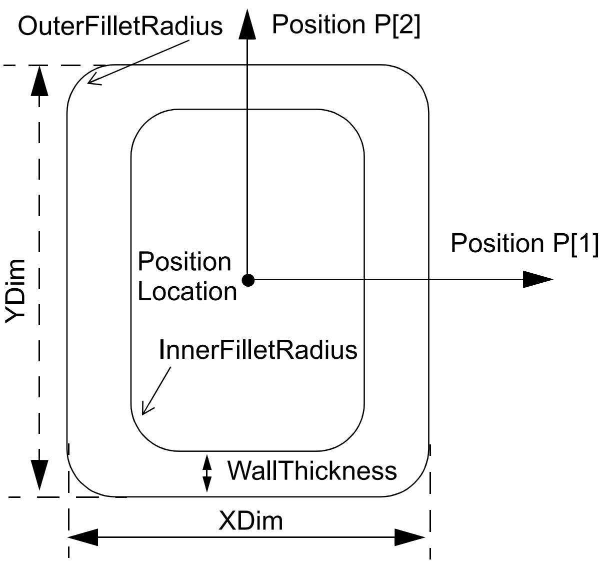
Definition from IAI: The ***IfcRectangleHollowProfileDef*** defines a section profile that provides the defining parameters of a rectangular (or square) hollow section to be used by the swept surface geometry or the swept area solid.

Its parameters and orientation relative to the position coordinate system are according to the following illustration.

A square hollow section can be defined by equal values for h and b.

The centre of the position coordinate system is in the profiles centre of the bounding box (for symmetric profiles identical with the centre of gravity).

Normally, the longer sides are parallel to the y-axis, the shorter sides parallel to the x-axis.



Position
The parameterized profile defines its own position coordinate system. The underlying coordinate system is defined by the swept area solid that uses the profile definition. It is the xy plane of:

IfcSweptAreaSolid.Position

by using offsets of the position location, the parameterized profile can be positioned centric (using x,y offsets = 0), or at any position relative to the profile. Explicit coordinate offsets are used to define cardinal points (e.g. upper-left bound).

Parameter
The parameterized profile is defined by a set of parameter attributes, see attribute definition below

ifc Attribute Definitions:

WallThickness: Thickness of the material. The wall thickness shall be smaller than the X and Y dimension of the rectangle.

InnerFilletRadius: Radius of the circular arcs, by which all four corners of the outer contour of rectangle are equally rounded. If not given, zero (= no rounding arcs) applies.

The inner fillet radius (if given) shall be smaller than or equal to the X and Y dimension of the rectangle minus the wall thickness.

OuterFilletRadius: Radius of the circular arcs, by which all four corners of the outer contour of rectangle are equally rounded. If not given, zero (= no rounding arcs) applies.

The outer fillet radius (if given) shall be smaller than or equal to the X and Y dimension of the rectangle

Continue to the next section 8.2.19.1.2.21 IfcArbitaryClosedProfileDef or return to 8.2.19.1.2 Some IFC 2x3 Definitions or 8.2.19.1 IFCs.

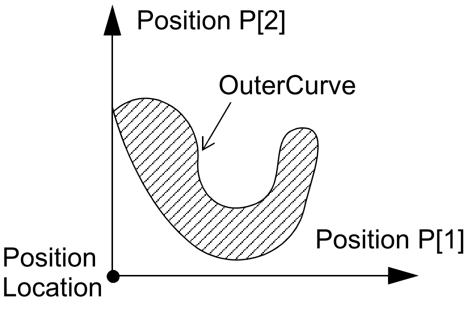
8.2.19.1.2.21 IfcArbitaryClosedProfileDef

Definition from IAI: The closed profile ***IfcArbitraryClosedProfileDef*** defines an arbitrary two-dimensional profile for the use within the swept surface geometry, the swept area solid or a sectioned spine. It is given by an outer boundary from which the surface or solid can be constructed.

Informal proposition:

1.The **OuterCurve** has to be a closed curve.

2.The **OuterCurve** shall not intersect.



Position
The ***OuterCurve*** is defined in the underlying coordinate system. The underlying coordinate system is defined by the swept surface or swept area solid that uses the profile definition. It is the xy plane of either:

IfcSweptSurface.Position

IfcSweptAreaSolid.Position

or in case of sectioned spines the xy plane of each list member of *IfcSectionedSpine.CrossSectionPositions.*

By using offsets of the position location, the parameterized profile can be positioned centric (using x,y offsets = 0.), or at any position relative to the profile. Explicit coordinate offsets are used to define cardinal points (e.g. upper-left bound).

Parameter
The ***OuterCurve*** attribute defines a two dimensional closed bounded curve.

ifc Attribute Definitions:

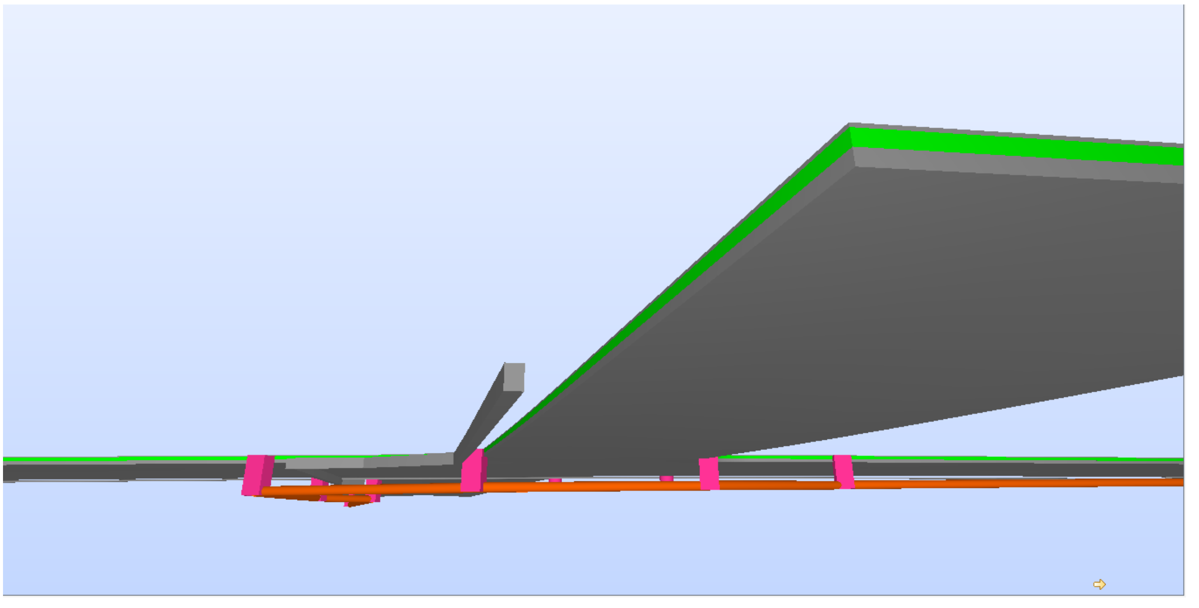
OuterCurve: Bounded curve, defining the outer boundaries of the arbitrary profile.

Continue to the next section 8.2.19.1.3 Representation of 12d Objects as IFCs or return to 8.2.19.1.2 Some IFC 2x3 Definitions or 8.2.19.1 IFCs.

8.2.19.1.3 Representation of 12d Objects as IFCs

Currently IFCs do not support road elements such a alignments or the spatial structures that match a road or rail design but it is still possible to get most of the shape data in***12d Model*** output in ways that will fit in with the IFC schema.

For example, in the image below, a drainage network with circular and rectangular pipes and pits, with thicknesses, a tin and trimeshes for the pavement layers have been written out from ***12d Model*** as an IFC and then read into **Solibri** and displayed.



See

8.2.19.1.3.1 Super Alignment
8.2.19.1.3.2 Super String
8.2.19.1.3.3 Drainage String
8.2.19.1.3.4 Sewer String
8.2.19.1.3.5 Tin
8.2.19.1.3.6 Trimesh

Old string types:
8.2.19.1.3.7 Alignment

8.2.19.1.3.1 Super Alignment

A **Super Alignment** is written out as an **IFCBUILDINGELEMENTPROXY** element.

The string attributes can be written out as an ifcPropertySet.

Continue to the 8.2.19.1.3.1 Super Alignment or return to 8.2.19.1.3 Representation of 12d Objects as IFCs or 8.2.19.1 IFCs.

8.2.19.1.3.2 Super String

A super string with a **rectangular pipe** is written out as an **IfcRectangleProfileDef** element.

A super string with a **circular pipe** is written out as an **IfcCircleRectangleProfileDef** element.

For a circular pit, the shape representation is ***ifcCircleHollowProfileDef***. See 8.2.19.1.2.18

IfcCircleHollowProfileDef.

For a rectangular pit, the shape representation is ***ifcRectangleHollowProfileDef***. Note however that this has only one wall thickness so when writing the pit out, the outer shape is correct but the maximum value of the top, bottom, left and right thicknesses is used for the wall thickness. So if the top, bottom, left and right thickness are not all the same value then the inner rectangle is smaller than it actually is. See 8.2.19.1.2.20 IfcRectangleHollowProfileDef.

Each **Pipe** is written out as an **IFCFLOWSEGMENTS** element.

For a circular pipe, the shape representation is ***ifcCircleHollowProfileDef***. 8.2.19.1.2.18 IfcCircleHollowProfileDef.

For a rectangular pipe, the shape representation is ***ifcRectangleHollowProfileDef***. Note however that this has only one wall thickness so when writing the pipe out, the outer shape is correct but the maximum value of the top, bottom, left and right thicknesses is used for the wall thickness. So if the top, bottom, left and right thickness are not all the same value then the inner rectangle is smaller than it actually is.

Continue to the 8.2.19.1.3.4 Sewer String or return to 8.2.19.1.3 Representation of 12d Objects as IFCs or 8.2.19.1 IFCs.

8.2.19.1.3.3 Drainage String

Each **Pit** is written out as an **IFCFLOWSTORAGEDEVICE** element.

For a circular pit, the shape representation is ***ifcCircleHollowProfileDef***. See 8.2.19.1.2.18 IfcCircleHollowProfileDef.

For a rectangular pit, the shape representation is ***ifcRectangleHollowProfileDef***. Note however that this has only one wall thickness so when writing the pit out, the outer shape is correct but the maximum value of the top, bottom, left and right thicknesses is used for the wall thickness. So if the top, bottom, left and right thickness are not all the same value then the inner rectangle is smaller than it actually is. See 8.2.19.1.2.20 IfcRectangleHollowProfileDef.

Each **Pipe** is written out as an **IFCFLOWSEGMENTS** element.

For a circular pipe, the shape representation is ***ifcCircleHollowProfileDef***. 8.2.19.1.2.18 IfcCircleHollowProfileDef.

For a rectangular pipe, the shape representation is ***ifcRectangleHollowProfileDef***. Note however that this has only one wall thickness so when writing the pipe out, the outer shape is correct but the maximum value of the top, bottom, left and right thicknesses is used for the wall thickness. So if the top, bottom, left and right thickness are not all the same value then the inner rectangle is smaller than it actually is.

Continue to the 8.2.19.1.3.4 Sewer String or return to 8.2.19.1.3 Representation of 12d Objects as IFCs or 8.2.19.1 IFCs.

8.2.19.1.3.4 Sewer String

Each **Maintenance hole** is written out as **IFCFLOWSTORAGEDEVICE** element.

For a circular maintenance hole, the shape representation is ***ifcCircleHollowProfileDef***. 8.2.19.1.2.18 IfcCircleHollowProfileDef.

For a rectangular maintenance hole, the shape representation is

ifcRectangleHollowProfileDef. Note however that this has only one wall thickness so when writing the maintenance hole out, the outer shape is correct but the maximum value of the top, bottom, left and right thicknesses is used for the wall thickness. This may mean that the inner rectangle may be smaller than it actually is.

Each **Pipe** is written out as an **IFCFLOWSEGMENT** element.

For a circular pipe, the shape representation is ***ifcCircleHollowProfileDef***. 8.2.19.1.2.18 IfcCircleHollowProfileDef.

For a rectangular pipe, the shape representation is ***ifcRectangleHollowProfileDef***. Note however that this has only one wall thickness so when writing the pipe out, the outer shape is correct but the maximum value of the top, bottom, left and right thicknesses is used for the wall thickness. So if the top, bottom, left and right thickness are not all the same value then the inner rectangle is smaller than it actually is.

Continue to the 8.2.19.1.3.5 Tin or return to 8.2.19.1.3 Representation of 12d Objects as IFCs or 8.2.19.1 IFCs.

8.2.19.1.3.5 Tin

A **Tin** is written out as an **IFCBUILDINGELEMENTPROXY** element.

The tin attributes can be written out as an ifcPropertySet.

Continue to the 8.2.19.1.3.6 Trimesh or return to 8.2.19.1.3 Representation of 12d Objects as IFCs or 8.2.19.1 IFCs.

8.2.19.1.3.6 Trimesh

A **Trimesh** is written out as an **IFCBUILDINGELEMENTPROXY** element.

The Trimesh attributes can be written out as an ifcPropertySet.

Return to 8.2.19.1.3 Representation of 12d Objects as IFCs or 8.2.19.1 IFCs.

8.2.19.1.3.7 Alignment

An **Alignment** is written out as an **IFCBUILDINGELEMENTPROXY** element.

The string attributes can be written out as an ifcPropertySet.

Continue to the 8.2.19.1.3.3 Drainage String or return to 8.2.19.1.3 Representation of 12d Objects as IFCs or 8.2.19.1 IFCs.

8.2.19.2 Project Units

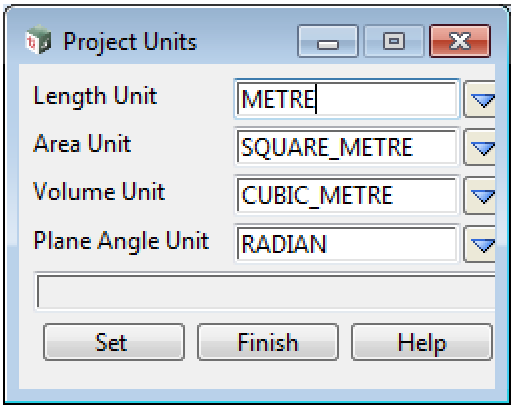
Position of option on menu: **File I/O =>Data output =>IFC => Project units**

This option creates the Project attributes group ***ifc_attributes/ifc_project_units*** and the values of the attributes are written out when an IFC STEP file (IFC file) is created.

Note that this does not change the data stored inside ***12d Model*** or written out to the IFC file.

The user must set the **IFC Units** to match what units they are been using inside ***12d Model***. And these by default are METRE, SQUARE_METRE, CUBIC_METRE and RADIAN.

Selecting **Project units** brings up the **Project Units** panel:



and if the Project attributes group ***ifc_attributes/ifc_project_units*** exists, the values are displayed in the panel. If the Project attributes group ***ifc_attributes/ifc_project_units*** does not exist, then it is created when the **Set** button is pressed.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Length Unit	choice box	as per Project Attributes	MILLIMETRE METRE, KILOMETRE

*length units used globally in the IFC file. Defined as SI units. This must match what you units data is in within **12d Model**.*

Area Unit	choice box	as per Project Attributes	SQUARE)MILLIMETRE, SQUARE_METRE, SQUARE_KILOMETRE

*area units used globally in the IFC file. Defined as SI units. This must match what you units data is in within **12d Model***

Volume Unit	choice box	as per Project Attributes	CUBIC_MILLIMETRE, CUBIC_METRE, CUBIC_KILOMETRE

*volume units used globally in the IFC file. Defined as SI units. This must match what you units data is in within **12d Model***

Plane Angle Unit	choice box	as per Project Attributes	RADIAN, DEGREES

plane angle units used globally in the IFC file. Defined as SI units.

Set	button		

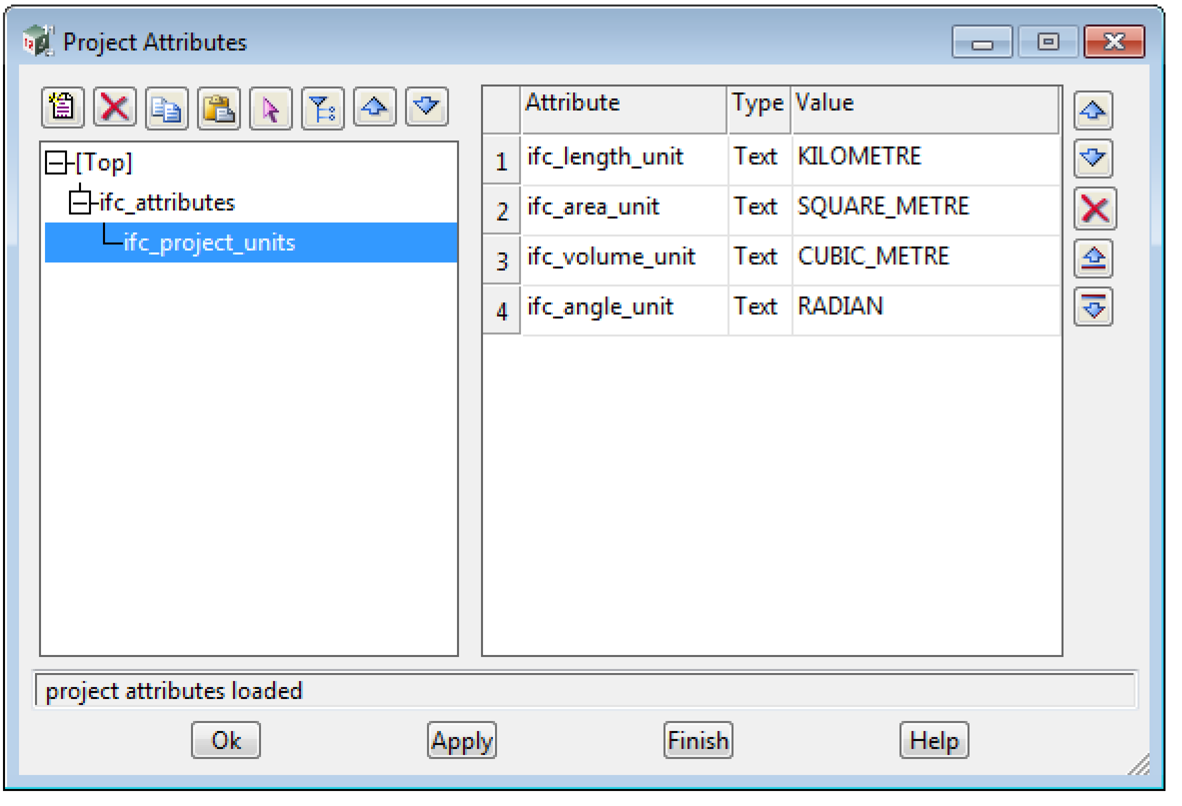
*updates/creates the Project attribute group **ifc_attributes/ifc_project_units** with the values in the* *panel.*

Note

The attributes can be displayed (and modified) using the option

Project =>Utilities =>Attributes

which brings up panel **Project Attributes**.



Warning:

Be careful making any changes using **Project Attributes**. The attribute name can't be and the attribute values must be valid IFC values.

Continue to next section 8.2.19.3 Extra Project Details or go back to 8.2.19 IFC Output.

8.2.19.3 Extra Project Details

Position of option on menu: **File I/O =>Data output =>IFC => Extra project details**

This option

(a) defines the path name of the ***Spatial Schema File*** that defines the types of IFC spatial elements that 12d elements can be tagged as, and the relationship between the allowed IFC spatial elements.

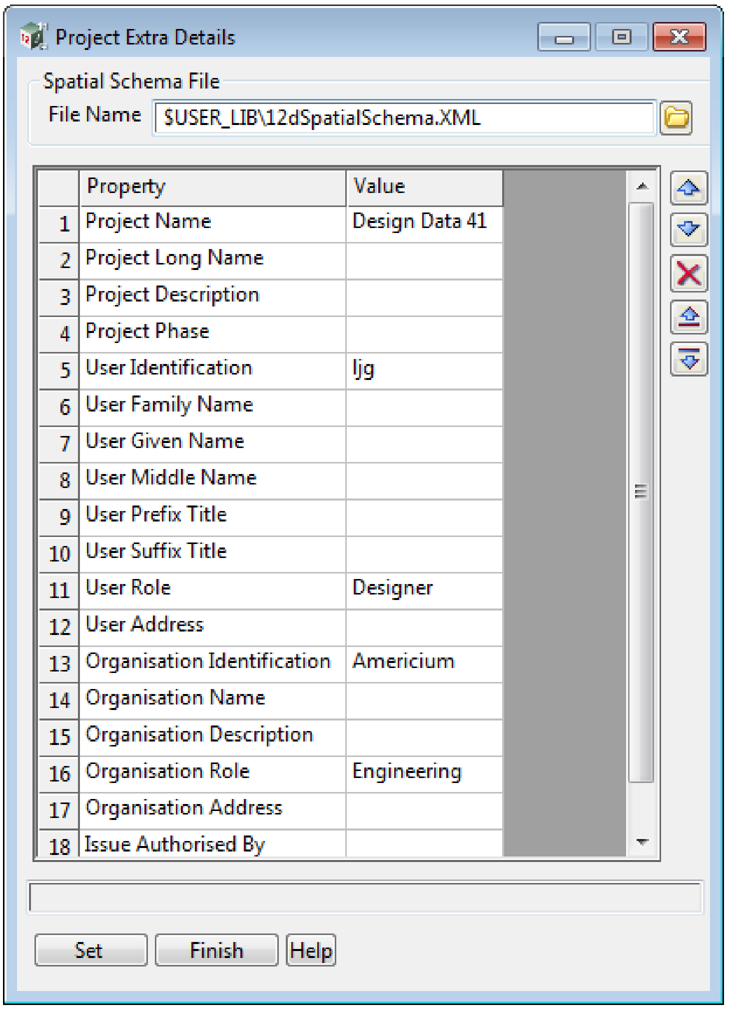
and

(b) creates and updates the **Project Attributes** group

ProjectExtraDetails.

that collects information to be written out as part of ***ifcProject***.

Selecting **Extra project details** displays the panel **Project Extra Details:**



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Spatial Schema File

File Name	file box		available *.xml files

name of the XML file which will be used by the **Spatial tag structure** *option (the* **Spatial Tag**

Aggregation *panel (the next step before using the IFC Writer to create an IFC file). For more information on when and how this file is created, see* 8.2.19.3.1 Spatial Schema File.

Grid

Property/Value

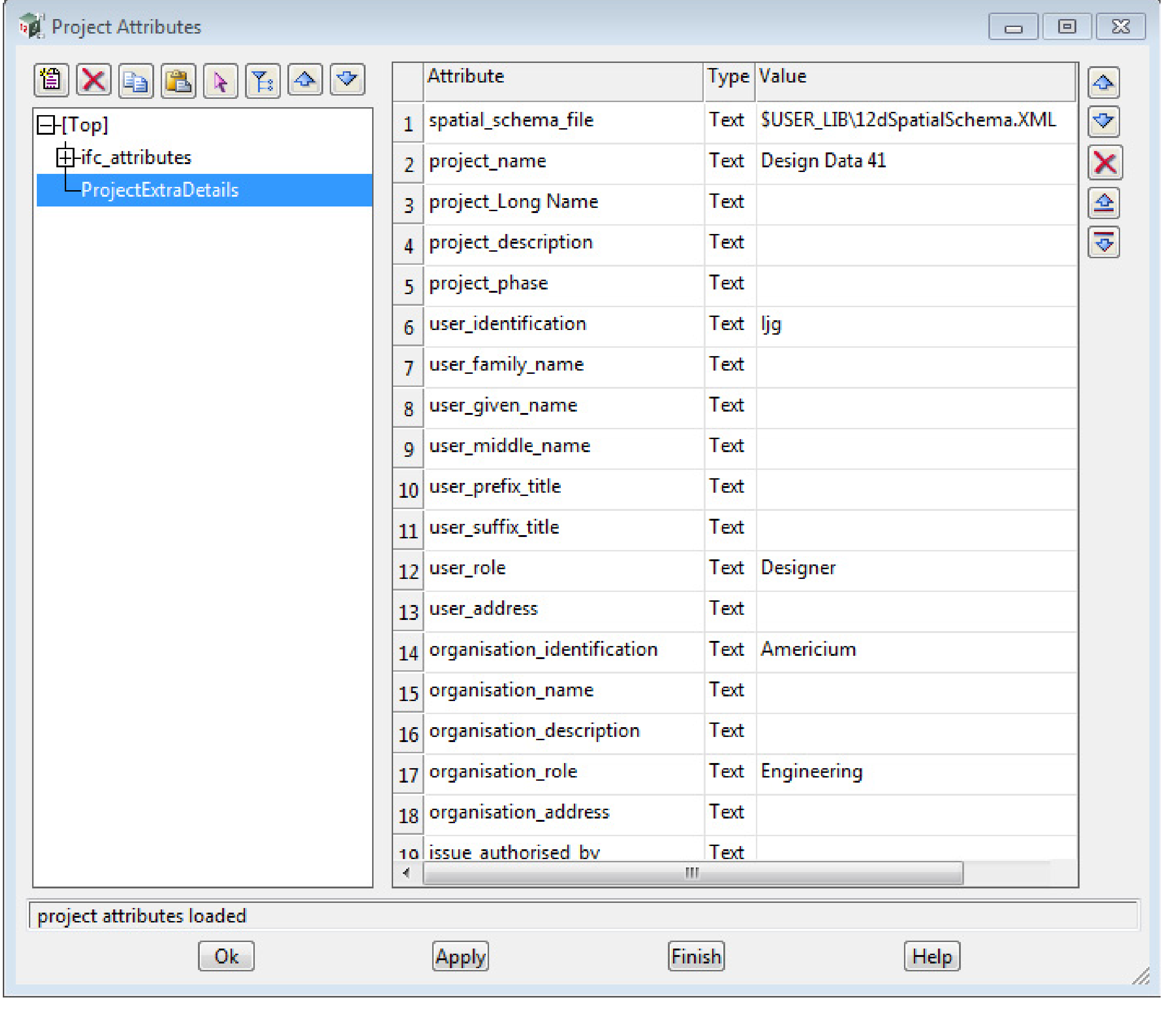
this grid allows the user to enter extra details about the IFC project and the information is stored as Project attributes in the group ***ProjectExtraDetails****. The values of these attributes are written out to the IFC file as part of the* ***ifcProject*** *element.*

Buttons at bottom

Set button

set the value of the attribute ***spatial_schema_file*** *in the* ***ProjectExtraDetails*** *group to the name given in the* **File name** *field.*

Then use the values in the **Property/Value Grid** *to create/update the values of the attributes in the Project attributes group* ***ProjectExtraDetails.*** *Note that these attributes are written out to the ifcProject element in any IFC file that is created by the* **IFC writer** *option (see* 8.2.19.5 IFC 2x3 Writer*).*



Continue to the next sub-section 8.2.19.3.1 Spatial Schema File or the next section 8.2.19.4 Defining and Assigning Spatial Structures, or return to 8.2.19 IFC Output.

8.2.19.3.1 Spatial Schema File

The **Spatial Schema File** is an XML file which is used by the **Spatial tag structure** option (the **Spatial Tag Aggregation** panel to define defines types of IFC spatial elements, and the allowed hierarchical relationship between the types.

For example, the type **Site** (ifcSite) is allowed as a standalone entity.

The type **Site Partial** (ifcBuilding) can't be standalone but must be associated with a specific **Site** and only a **Site**. Whereas **Building** (ifcBuilding) also can't stand alone but it can be associated with a specific **Site** or a specific **Site Partial.** (See 8.2.19.1.1 Spatial Structures).

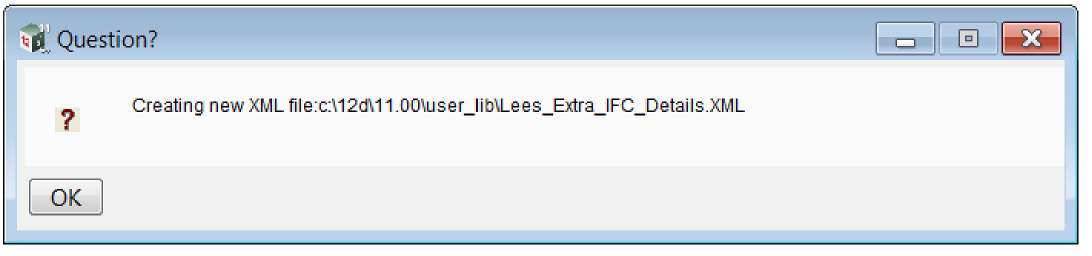
When **Extra project details** is selected the Project attribute:

ProjectExtraDetails/spatial_schema_file

is searched for and

(a) if the **attribute exists** then a file of the name of the value of the attribute is searched for.

If the file does not exist, then a **Question?** panel comes up:

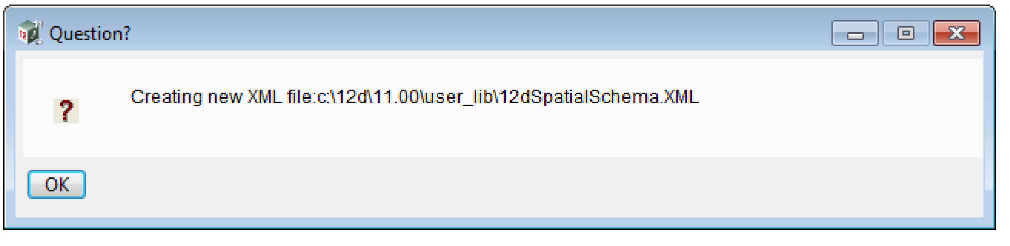


If **OK** is pressed then a file of the given name is created and a default spatial schema written to the file.

OR

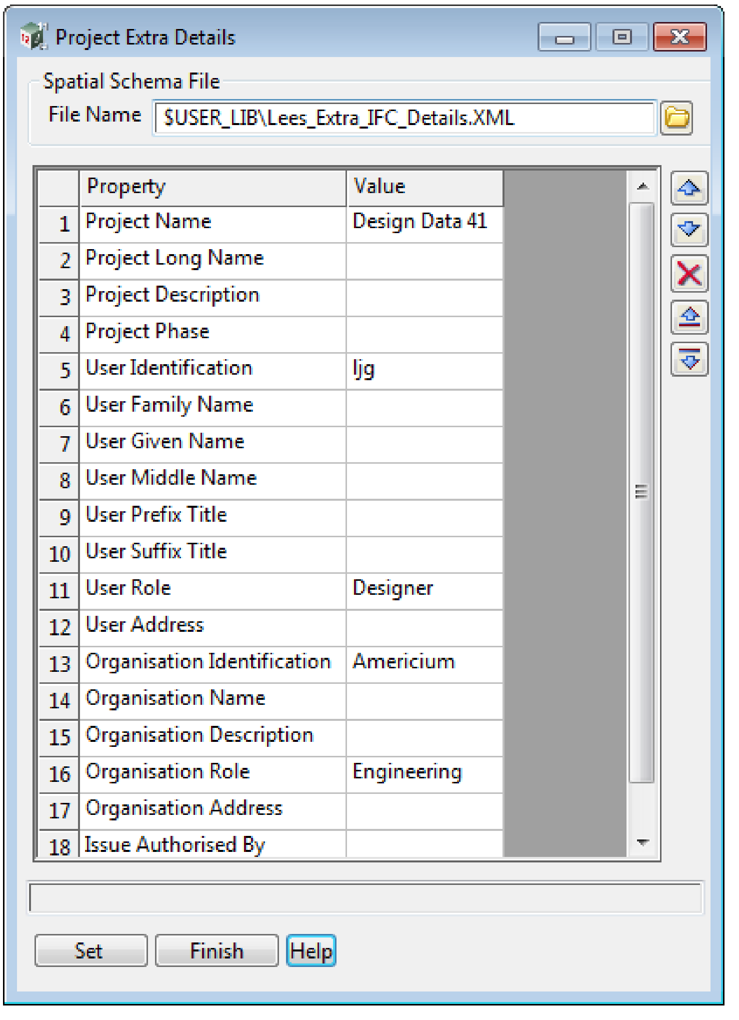
(b) if the **attribute doesn't exit** then a file of the name called *12dSpatialSchema.XML* is searched for in $USER_LIB.

If the file does not exist, then a **Question?** panel comes up:



If **OK** is pressed then the file $USER_LIB*12dSpatialSchema.XML*is created and a default spatial schema written to the file.

The panel **Project Extra Details** is then brought up displaying name of the *Spatial Schema File*, and also the grid of parameters that are stored in the Project attribute group *ProjectExtraDetails* for writing out to the ***ifcProject*** element in the IFC file.



Continue to next section 8.2.19.4 Defining and Assigning Spatial Structures or return to 8.2.19 IFC Output.

8.2.19.4 Defining and Assigning Spatial Structures

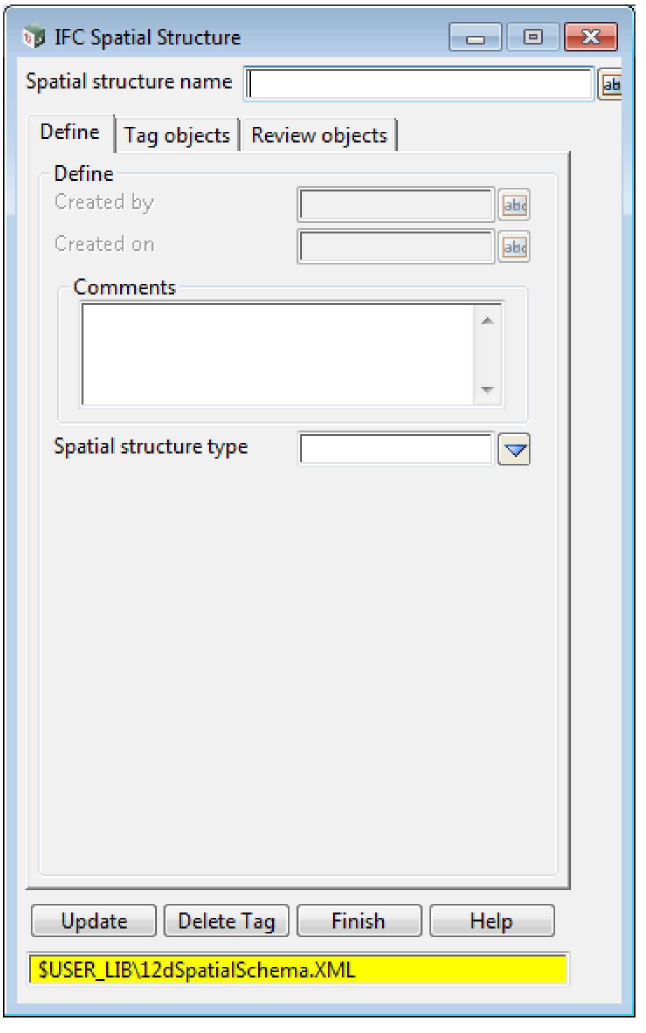
Position of option on menu: **File I/O =>Data output =>IFC => Spatial structures**

This option actually has three functions:

1. It creates the named structures as tags using the *Spatial Schema File* referred to by the attribute ***spatial_schema_file*** in the Project attribute group ***ProjectExtraDetails*** (see 8.2.19.3 Extra Project Details).
2. it tags models and/or strings with a spatial structure defined in 1.
3. it select all items of given spatial structure and optionally remove the spatial structure tag from any of them.

See 8.2.19.1.1 Spatial Structures for more information on Spatial Structures.

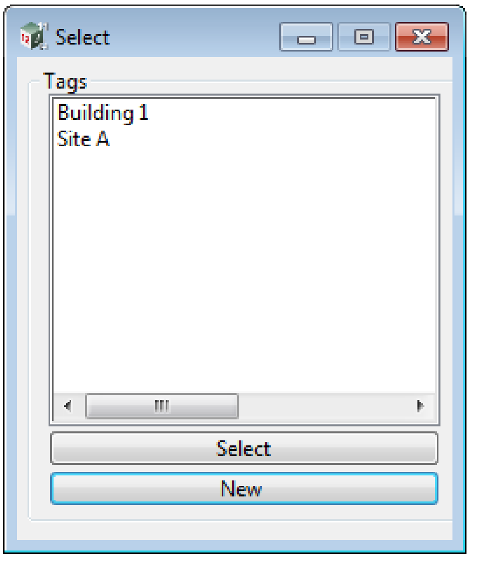
Selecting **Spatial structure** brings up the **IFC Spatial Structure** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Spatial structure name	input		Select/New Spatial structures

*clicking LB **inside** the text entry field bring up the* **Select Spatial Structure** *panel will bring up the New Spatial Structure panel is no spatial structures have been defined (*8.2.19.4.1 Create New Spatial Structure*), or if spatial structures have already been defined, a* **Select** *panel with the choices* *of existing spatial structures:*



If ***New*** *is selected then* **Create Spatial Structure** *panel is brought up to. See* 8.2.19.4.1 Create New Spatial Structure.

If a Spatial Structure is double clicked on in the list, or a Spatial Structure highlighted in the list and ***Select*** *pressed, then the rest of the panel is filled with the information for the selected Spatial Structure. the information can then be edited and then updated by clicking on the Update button.*

For information on each Tab and the Buttons at the bottom, see:
Define Tab
Tag Objects Tab
Review Objects Tab
Buttons at Bottom

Define Tab

this tab is used when a new Spatial Structure is being created or an existing one edited.

Created by output from selected Spatial Structure

cannot be edited from this tab.

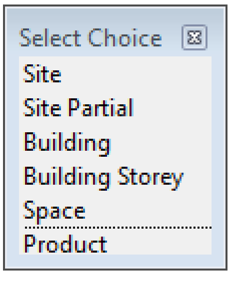
Created on output from selected Spatial Structure

cannot be edited from this tab.

Comments input/output from selected Spatial Structure

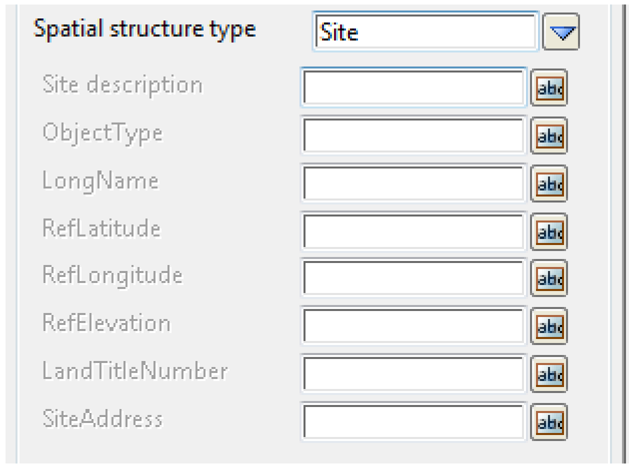
displays the existing comments which can be edited and extra comments added.

Spatial structure type choice box from selected Spatial Structure



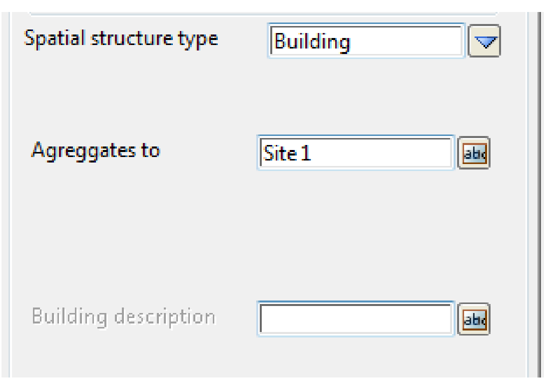
displays the spatial structure types for the selected Spatial Structure. This can be changed and the *choices are defined in the spatial schema file. See* 8.2.19.3.1 Spatial Schema File.

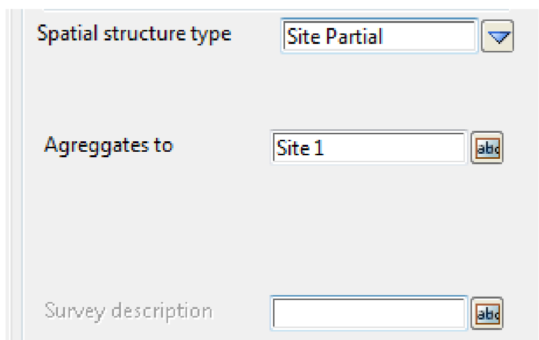
if ***Spatial structure*** *type is set to* ***Site****, the following fields are displayed.*



The fields and names on the left had side of the panel are populated from the Spatial Schema File (see 8.2.19.3.1 Spatial Schema File*). The values for the fields can be created/edited and when the* **Update** *button is pressed, the values are stored as a subgroup of the Tags group of the Project attributes. The subgroup has the name of the Site.*

If the ***Spatial structure*** *type is set to anything other than Site, only the* **Aggregates to** *and fields that are defined the Spatial Schema File (see* 8.2.19.3.1 Spatial Schema File*).*



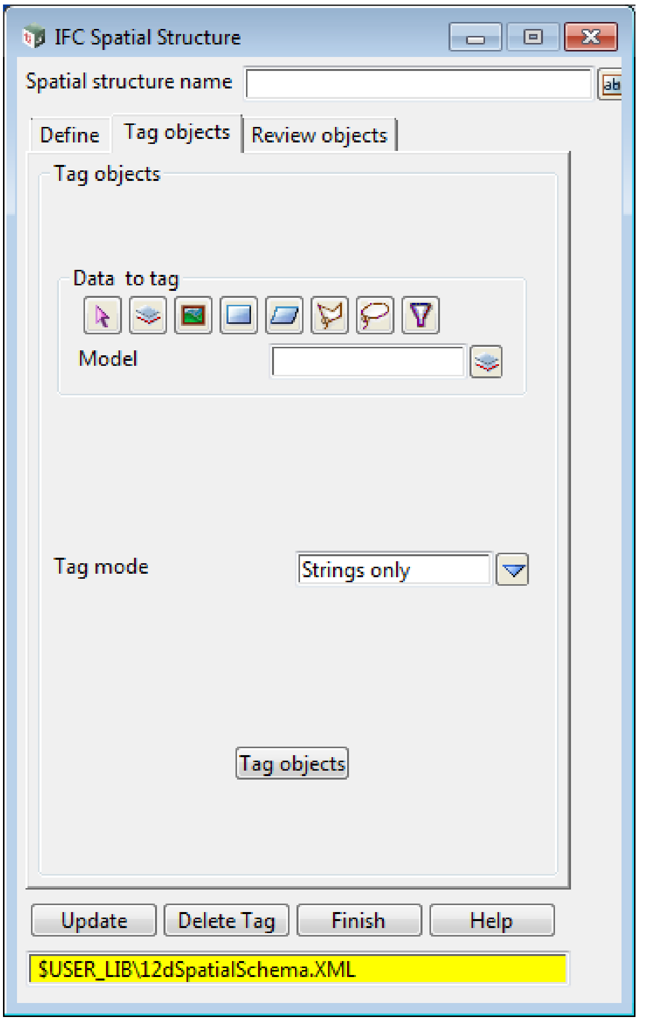


Clicking in the ***Aggregates to*** *field brings up a list of Spatial Structures that have already been created. The existing Spatial Structure that this new spatial structure is in is selected. See* 8.2.19.1.1 Spatial Structures.

The values for the other fields can be created/edited and when the **Update** *button is pressed, the values are stored as a subgroup of the Tags group of the Project attributes. The subgroup has the name of the Spatial Structure.*

Tag Objects Tab

this tab is used to tag data are being in a given Spatial Structure.



Data to tag Model

data selection type - for a full description go to 4.19.3 Data Source.

Data source input

source of data is to be selected for tagging.

Tag mode choice box Models only, Strings only, Strings and models

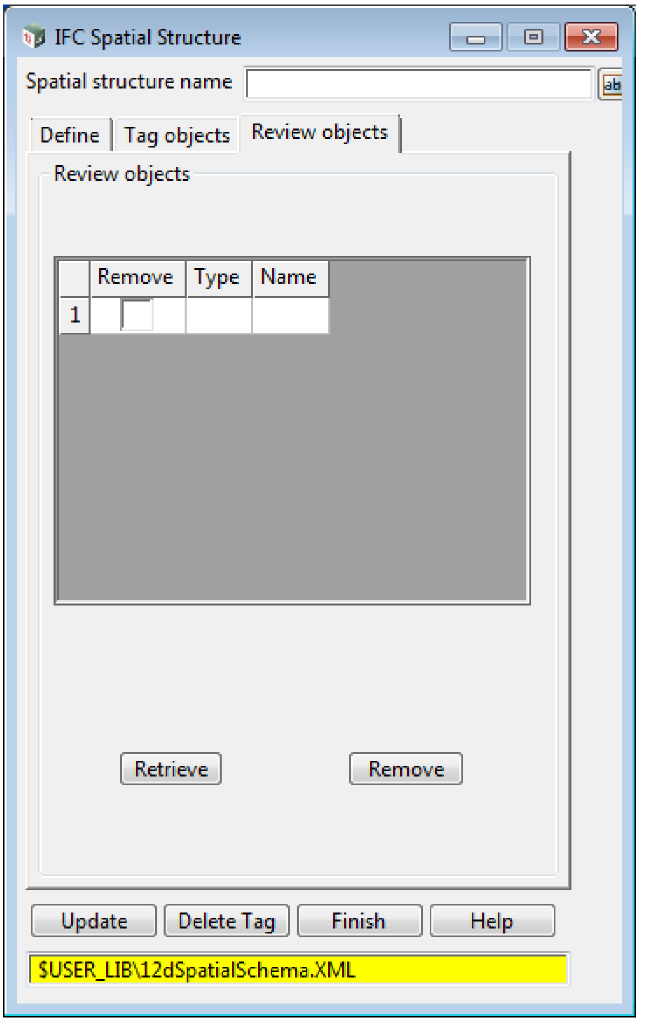
creates attributes on the selected strings and/or models to indicate which spatial structure the strings and/or models belong to.

Tag objects button

tags the selected strings and models with attributes so it is known that they belong to a spatial structure, and which structure it is.

Review Objects Tab

this tab selects data with a given Spatial Structure and can also remove selected items from the spatial structure.



Review Objects Grid

Remove tick box

if ticked, the string or model in that row will be removed when the **Remove** *button is clicked.*

Whether strings, models, or both are available is dependent on Tag mode selected in Tag objects tab.

Type output String, Model or Trimesh

the type of the object - string, model or trimesh.

Name output

the name of the object.

Retrieve button

adds a selected string or model.

Remove button

when pressed the data for the given Spatial Container is listed in the grid.

Buttons at Bottom

Update

*applies each change. Hit **Update** after every change.*

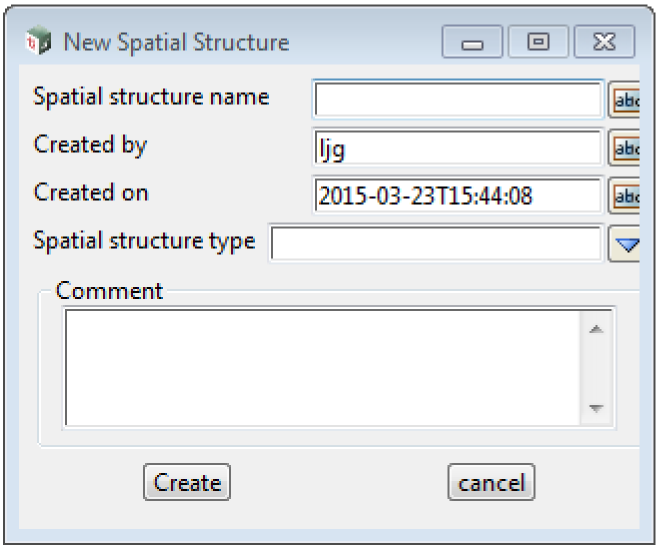
Delete Tag

deletes all objects with the selected tag.

Continue to next section 8.2.19.5 IFC 2x3 Writer or return to 8.2.19 IFC Output.

8.2.19.4.1 Create New Spatial Structure

Selecting **New** brings up the **Create Spatial Structure** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Spatial structure name	input		

the name of the new spatial structure

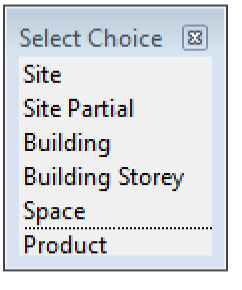
Created by	input	user name	

can be edited to show a different user name if need be

Created on	input	current date and time	

can be edited to show a different date and time

Spatial structure type	choice box	as per the spatial schema file.	



spatial structure types that are defined in the spatial schema file. See 8.2.19.3.1 Spatial Schema File.

Comments	input		as per New Container panel

allows user comments to be entered.

Create	button		

create the new Spatial Structure.

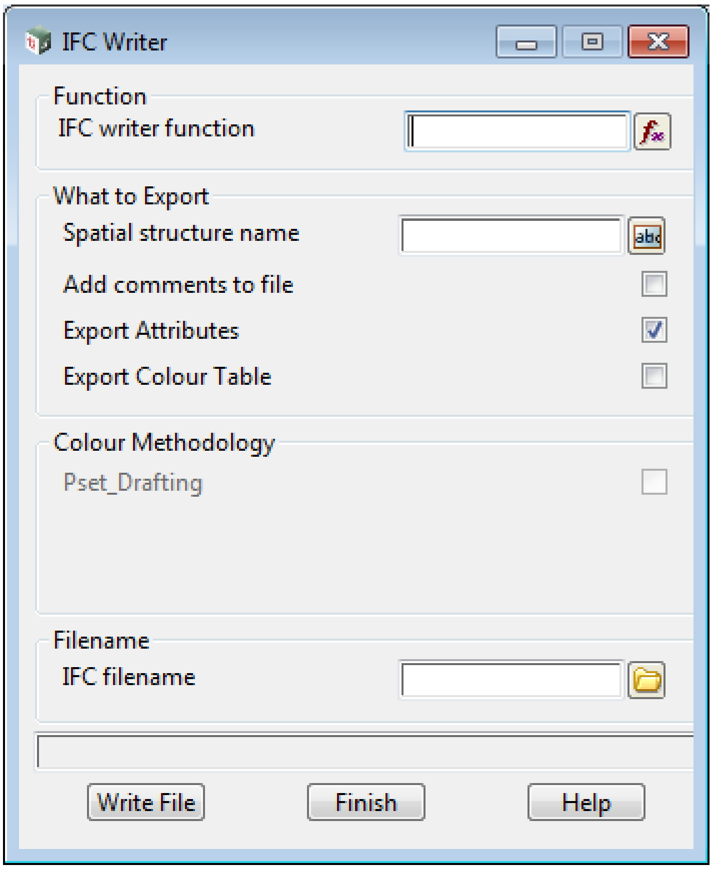
Return to 8.2.19.4 Defining and Assigning Spatial Structures.

8.2.19.5 IFC 2x3 Writer

Position of option on menu: **File I/O =>Data output =>IFC => IFC Writer**

The **IFC Writer** option writes out data with a given tag to a file in the IFC 2x3 format (see 8.2.19.1.2 Some IFC 2x3 Definitions).

Selecting **IFC Writer** brings up the **IFC Writer** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Function

IFC writer function	function box		available IFC functions

the name for the IFC Writer Function. Note that the IFC Writer is a function so that it can remember the settings in the panel and can be easily recalced as a function at any time.

What to Export

Spatial structure name	defined spatial structures		available spatial structures

All data with that has been tagged with this IFC spatial structure name, or has been tagged with an IFC spatial structure name that is in the Spatial Structure hierarchy under this spatial structure, is written out to the IFC file. See 8.2.19.1.1 Spatial Structures *and* 8.2.19.4 Defining and Assigning Spatial Structures*).*

So only the name of one spatial structure is required and all the data in that spatial structure and the data in all those spatial structures under it are selected for writing out.

Add comments to file	tick box	not ticked	

*if **ticked**, comments are added to the IFC file denoting what each group of data is in the IFC file. For example, IFC MODEl INFORMAITON, PROJECT INFORMATION and ELEMENTS CONTAINED IN SPATIAL STRUCTURES.*

Export attributes tick box ticked

*if **ticked**, the 12d Project Attributes will be exported to the IFC file to the ifcProject. For 12d Objects that are represented as ifcBuildingElementProxy elements, the string/tin/trimesh attributes are exported as an ifcPropertySet.*

Export colour table tick box not ticked

*if **ticked**, the Colour Table will be exported to the IFC file.*

Colour Methodology

Pset_Drafting tick box ticked (if Export Colour Table is ticked)

if **Export colour table** *is ticked then this is automatically ticked on.*

Filename

IFC Filename file box available *.ifc files

the name of the IFC file to write all data tagged with the given Spatial Structure.

Buttons at bottom

Write File button

write the IFC 2x3 file.

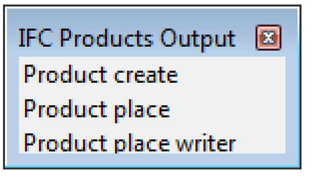
Return to 8.2.19 IFC Output.

8.2.19.6 Products

Position of option on menu: **File I/O =>Data output =>IFC => Products**

This section of documentation is a work in progress and will be updated in subsequent releases.

Selecting **Products** brings up the **Products** menu.



For the options see;

Product create 8.2.19.6.1 Writing 12d Solutions 12da Data with Place Nodes
Product place 8.2.19.6.2 Product Place
Product place writer 8.2.19.6.3 Product Place Writer

8.2.19.6.1 Writing 12d Solutions 12da Data with Place Nodes

Position of option on menu: **File I/O =>Data output =>IFC => Products =>Product create**

This section of documentation is a work in progress and will be updated in subsequent releases.

Selecting **Product create** brings up the **Write 12d Solutions Ascii Data with Place Nodes** panel.

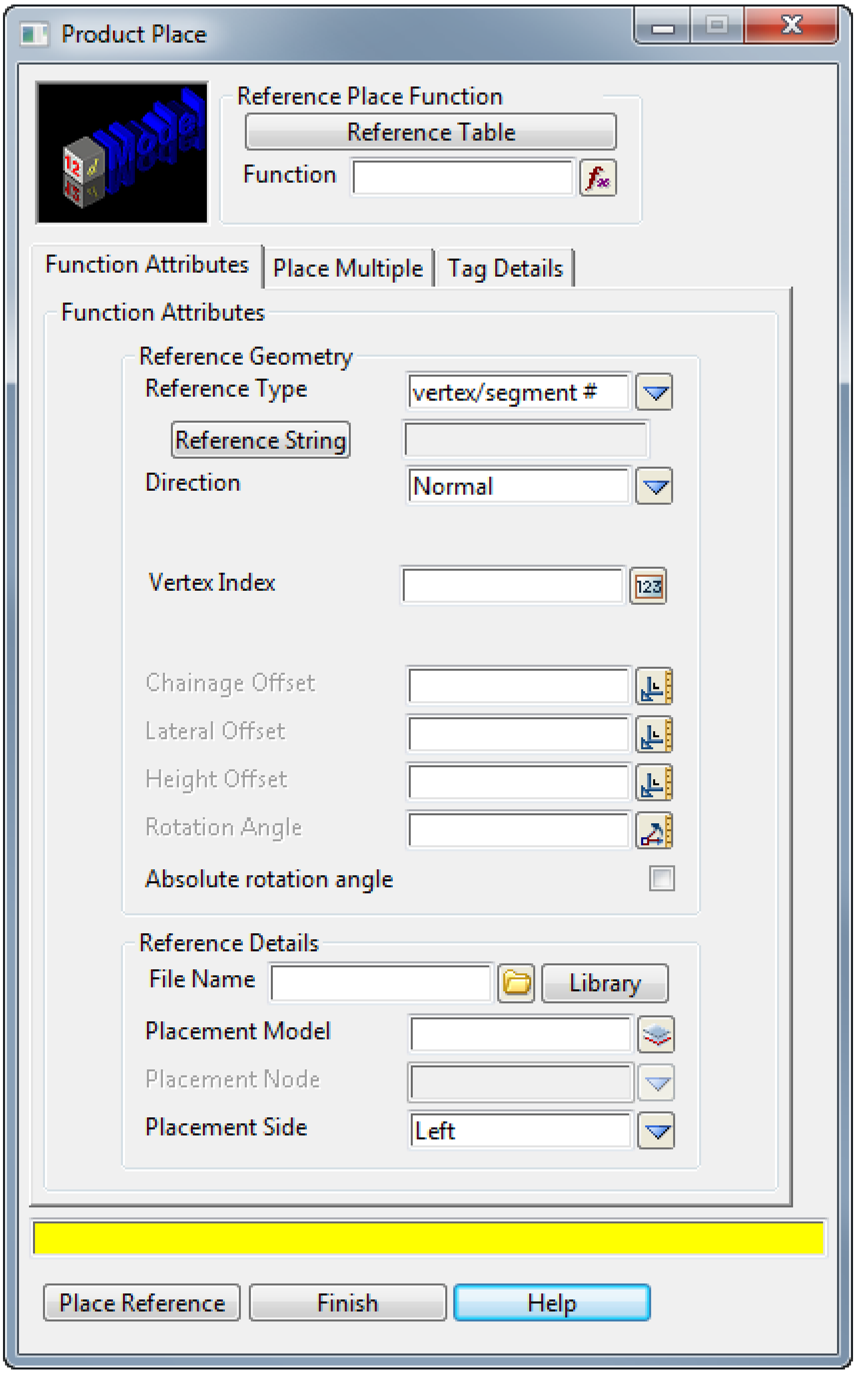
Continue to next section 8.2.19.6.2 Product Place or go back to 8.2.19.6.6 Products.

8.2.19.6.2 Product Place

Position of option on menu: **File I/O =>Data output =>IFC => Products =>Product place**

This section of documentation is a work in progress and will be updated in subsequent releases.

Selecting **Product place** brings up the **Product Place** panel.



Continue to next section 8.2.19.6.3 Product Place Writer or go back to 8.2.19.6 Products.

8.2.19.6.3 Product Place Writer

Position of option on menu: **File I/O =>Data output =>IFC => Products =>Product place writer**

This section of documentation is a work in progress and will be updated in subsequent releases.

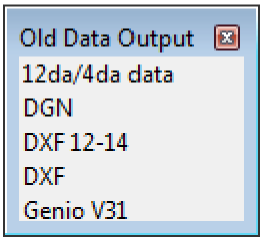
Selecting **Product place writer** brings up the **CSV Tag File Writer** panel.

Go back to 8.2.19.6 Products or go back to 8.2.19 IFC Output.

8.2.20 Old Outputs

Position of menu: **File I/O =>Data output =>Old**

The **Old** menu contains superseded options. The **Old** walk-right menu is



For the option *12da/4da data, go to*

DGN	
DXF 12-14	8.2.20.1 DXF 12-14 Output
DXF	8.2.20.2 DXF Output
Genio V31	8.2.20.3 Genio V3.1 Output

8.2.20.1 DXF 12-14 Output

Position of option on menu: **File I/O =>Data output =>Old =>DXF 12-14**

SUPERSEDED OPTION

The DXF 12-14 output format is for writing data out in a format compatible with AutoCAD. The DXF file produced **does not** have a full DXF header with the required line types, text styles, fonts already defined. Hence the DXF file needs to be loaded into an **existing** Autocad drawing.

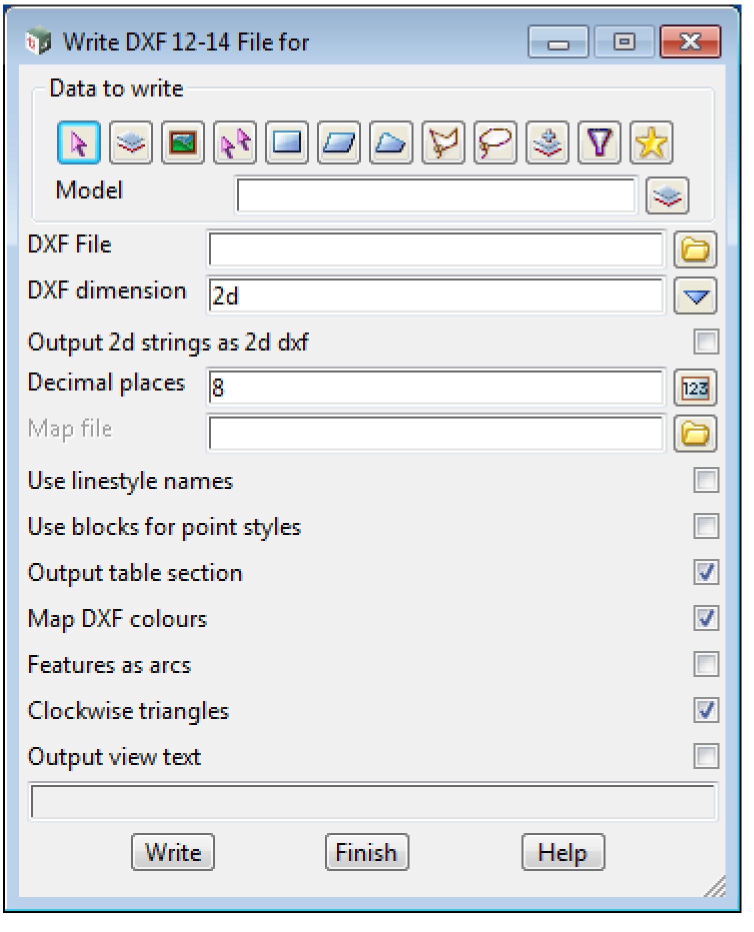
AutoCAD Release 10 introduced binary DXF files as a means of addressing the problems of large file sizes, slow processing and limited accuracy that occur when using the ASCII DXF format. 12d Model can write binary DXF files. See the **Precision** field below.

12d Model line strings are output as POLYLINEs, point strings as a series of AutoCAD POINTs and triangles as 3DFACEs.

By default, the string or triangle colour is used in the Autocad colour record. The DXF layer used is the items model name with any spaces in the model name replaced by a minus (-). However, a DXF output map file with matches on string name can be used to give DXF layers, colours and line type (see next section).

At this stage, 4d, interface and alignment strings are only output as Autocad POLYLINES.

On selecting the **DXF 12-14** output option, the **Write DXF 12-14 File for** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source input

source of data is to be written out to a file.

DXF File input *.dxf files

name of the file for the information to be written out to. If the file already exists, the data will be appended to the file.

DXF Dimension input 2d 2d, 3d

*if **3d**, strings will be written out as three dimensional DXF polylines. Alignment strings and arcs will be approximated by short lines.*
*If **2d**, the z-value of all the data is set to zero and arcs and curves in alignment strings are written out as dxf polylines with bulges for the arcs.*

Decimal places input 8

the number of decimal places to be used for co-ordinates etc. in the DXF file.
*If this number is **negative**, a binary DXF file will be written. The actual number specified becomes irrelevant in this case, since full precision is preserved in a binary DXF file.*

Map file input *.mf files

if non-blank, the name of a file to be used as a 12d *Model to DXF output map file. See* 8.2.20.1.1 DXF Output Map File.

Use linestyle names tick box

*if **ticked**, a* 12d *Model string's linestyle name is used as the DXF element's line type ("1" goes to CONTINUOUS).*
*if **not ticked**, all DXF line types are CONTINUOUS.*

Use blocks for point styles tick box

*if **ticked**, for each* ***12d Model*** *point string, an AutoCAD block of the same name as the* ***12d Model*** *string's linestyle is places at each point.*

Output table section tick box tick

*if **ticked**, an AutoCAD table section is written out at the top of the DXF file.*

Map DXF colours tick box tick

*if **ticked**, a mapping is made between the first seven default* ***12d Model*** *colours and the corresponding DXF colours. The other* ***12d Model*** *colours are mapped to the DXF colour of the same colour number.*
*If **not ticked**, the nth* ***12d Model*** *colour is mapped to the nth DXF colour.*

Features as arcs tick box

*if **ticked**, feature strings output as arcs in DXF.*
*if not **ticked**, the centre of the feature string is output as a point.*

Output view text tick box

*if **ticked**, any view text turned on (point numbers, z-values etc.) are output as text in DXF.*
*if not **ticked**, view text is not output.*

Write button

write out, in DXF format, the data in the model/view given in the model/view field, to the file given in the file field.

8.2.20.1.1 DXF Output Map File

When using the output options to write out a DXF file, an output map file can be used, with 12d Model string names as the entity-name to match the key, so that the user can specify AutoCAD colour and style. BYLAYER can be used with colour and style.

The key can have wild cards (*) and wild characters (?) as for the input map files.The fields in the DXF output map file are (ACD = AutoCAD):

field 1	key - string name	can include wild cards * and wild characters ?
field 2	new name	not output to DXF
field 3	ACD layer	* for ***12d Model*** name
field 4	ACD colour	number between 0 and 256, * for ***12d Model*** colour mapped to ACD, BYLAYER for ACD BYLAYER
field 5	ACD line type	* for ***12d Model*** linestyle, BYLAYER for ACD BYLAYER
field 6	ACD text style	not yet used, * for ***12d Model*** text style

The DXF file produced needs to be loaded into an existing Autocad drawing which has the ACD layers and linestyles defined.

An example of an DXF output map file is.

```
//      1       2       3          4          5             6
//      key     name    layer      colour     linetype      textstyle

        cont*   CONT    CONTOURS   1          1             *
        EB*     TOP     ROAD       2          CONTINUOUS    *
        103     TOP     *          3          *             *
        fe*     TOP     FENCE      *          BYLAYER       *
        SURV*   TOP     SURVEY     BYLAYER    DASH          *
```

8.2.20.2 DXF Output

Position of option on menu: **File I/O =>Data output =>Old =>DXF**

SUPERSEDED OPTION

The DXF output format is for writing data out in a format compatible with AutoCAD version 15. The DXF file produced **does not** have a full DXF header with the required line types, text styles, fonts already defined. Hence the DXF file needs to be loaded into an existing Autocad drawing.

This option is the one used in 12d Model V3.1. It is now superseded by the **DWG/DXF** output option.

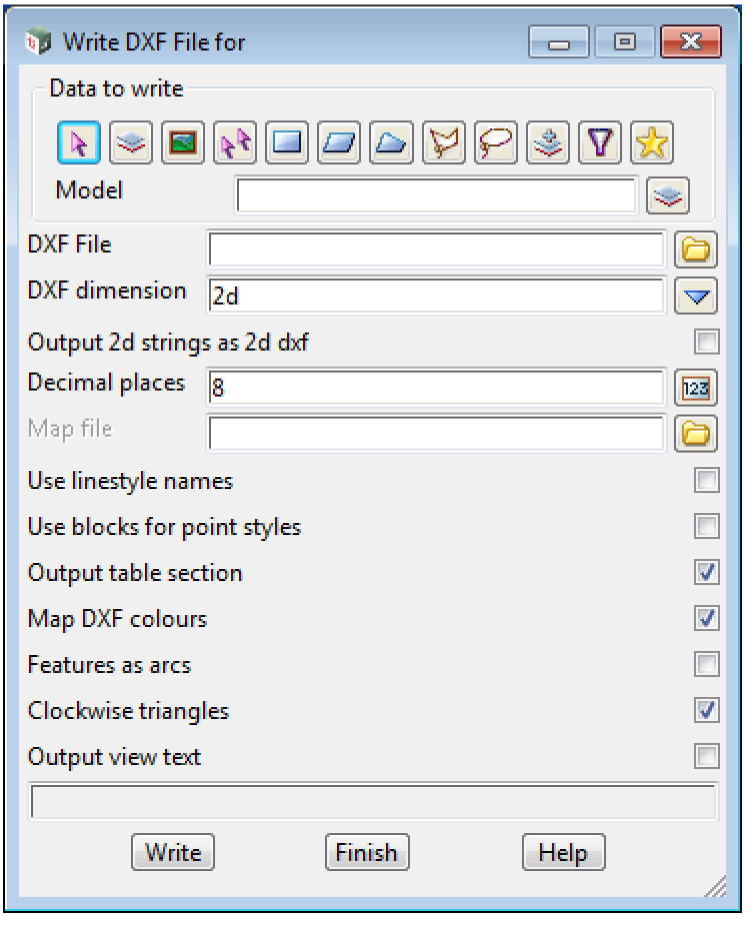
AutoCAD Release 10 introduced binary DXF files as a means of addressing the problems of large file sizes, slow processing and limited accuracy that occur when using the ASCII DXF format. 12d Model can write binary DXF files. See the **Precision** field below.

12d Model line strings are output as POLYLINEs, point strings as a series of AutoCAD POINTs and triangles as 3DFACEs.

By default, the string or triangle colour is used in the Autocad colour record. The DXF layer used is the items model name with any spaces in the model name replaced by a minus (-). However, a DXF output map file with matches on string name can be used to give DXF layers, colours and line type (see next section).

At this stage, 4d, interface and alignment strings are only output as Autocad POLYLINES.

On selecting the **DXF** output option, the **Write DXF file for** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data is to be written out to a file.

DXF File	input		*.dxf files

name of the file for the information to be written out to. If the file already exists, the data will be appended to the file.

DXF Dimension	input	3d	2d, 3d

if 3d, strings will be written out as three dimensional DXF polylines. Alignment strings and arcs will be approximated by short lines.
*If **2d**, the z-value of all the data is set to zero and arcs and curves in alignment strings are written out as dxf polylines with bulges for the arcs.*

Decimal places	input	8	

the number of decimal places to be used for co-ordinates etc. in the DXF file.
*If this number is **negative**, a binary DXF file will be written. The actual number specified becomes irrelevant in this case, since full precision is preserved in a binary DXF file.*

Map file	input		*.mf files

if non-blank, the name of a file to be used as a 12d Model to DXF output map file. See 8.2.20.1.1 DXF Output Map File.

Use linestyle names	tick box		

*if **ticked**, a 12d Model string's linestyle name is used as the DXF element's line type ("1" goes to CONTINUOUS).*
*if **not ticked**, all DXF line types are CONTINUOUS.*

Use blocks for point styles	tick box		

*if **ticked**, for each* ***12d Model*** *point string, an AutoCAD block of the same name as the* ***12d Model*** *string's linestyle is places at each point.*

Output table section	tick box	tick	

*if **ticked**, an AutoCAD table section is placed at the beginning of the DXF file.*

Map DXF colours	tick box	tick	

*if **ticked**, a mapping is made between the first seven default 12d Model colours and the corresponding DXF colours. The other* ***12d Model*** *colours are mapped to the DXF colour of the same colour number.*
*If **not ticked**, the nth* ***12d Model*** *colour is mapped to the nth DXF colour.*

Features as arcs	tick box		

*if **ticked**, feature strings output as arcs in DXF.*
*if not **ticked**, the centre of the feature string is output as a point.*

Output view text	tick box		

*if **ticked**, any view text turned on (point numbers, z-values etc.) are output as text in DXF.*
*if not **ticked**, view text is not output.*

Write	button		

write out, in DXF format, the data in the model/view given in the model/view field, to the file given in the file field.

8.2.20.3 Genio V3.1 Output

Position of option on menu: **File I/O =>Data output =>Old =>Genio V31**

SUPERSEDED OPTION

Genio i/o is a separate chargeable module.

This is the genio output module from 12d Model V3.1. This module has been replaced by the new genio module but the genio v3.1 will be left in V4.0 in case there a any problems with the new module.

The **genio v3.1 output** option uses a non-default format to write out the strings in a model or on a view, to a genio file. A genio 001 option is used to specify the record format.

The first thirty two characters of the model name are converted to upper case and used as the Moss model name.

In Moss, string labels can have a maximum of four characters. Since 12d Model places no restrictions on string name length, the following rules are used to produce genio string labels.

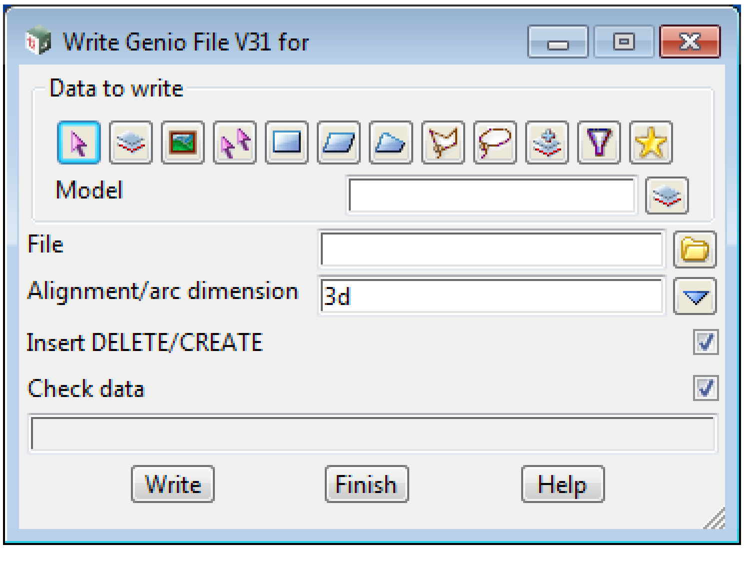
For a line-string, the first four characters of its string name are used as the genio string label. The four characters are mapped to upper case.

Similarly for point-strings whose names start with **P**. Otherwise the genio string label is **P** plus the first three letters of the point-string name. The three letters are mapped to upper case.

The name of a Moss text string must start with a * so 12d Model uses * and the first three characters of the text string's name.

An alignment string can be written out as either a moss 3d, 6d or 12 string.

On selecting the **Genio V31** output option, the **Write Genio File V31 for** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	
data selection type - for a full description go to 4.19.3 Data Source.			
Data source	input		
source of data is to be written out to a file.			
File	input		*.mos files

name of the file the model/view is to be written out to. If the file already exists, the data will be appended to the bottom of the file.

Alignment dimension	input	3d	3d,6d,12d

if 3d is selected, alignment strings as written out as a 3d string as a series of straight lines.
If 6d is selected, alignment strings are written out as Moss 6d strings.
If 12d is selected, alignment strings are written out as Moss geometry strings. This is the only way that full horizontal and vertical geometry can be transferred to Moss.

Insert DELETE/CREATE	tick box	tick

if ticked, the genio commands DELETE and CREATE are written at the top of the file to correctly define the moss models for the data.

Write	button

write out in genio format the data in the model/view given in the **model/view** *field, to the file given in the* **file** *field. If the file already exist, the data will be appended to the file.*

8.2.20.4 Eagle Output

Position of option on menu: **File I/O =>Data output =>Old =>Eagle**

Eagle is a 3D CAD package used throughout Australia, Asia and Europe.

Using the **Eagle output** option, 12d Model produces an Eagle command file which is used by Eagle to create an Eagle model. Since Eagle supports 3D faces, the triangles created in 12d Model can be transferred to Eagle for use in shaded models and walk-throughs.

Instead of colours, Eagle uses pens, dash styles, line thicknesses and frags. To allow the user to map 12d Model colours to appropriate Eagle settings, an **Eagle map file** setting out the ***12d Model*** and Eagle relationships can be provided.

The eagle map file is a user created file consisting of a list of 12d Model colours and the Eagle pen, dash style, pen thickness and frag to be used for the colour.

The map file is set out with one ***12d Model*** colour per line. The line begins with the ***12d Model*** colour followed by the Eagle pen number, dash style, thickness and frag to be used for the ***12d Model*** colour. Each item is separated by one or more spaces.

For example, if the ***12d Model*** colour **red** is to be mapped to Eagle pen 3, dash style 4, thickness 2 and frag 1, then the line in the eagle map file would be

```
red        3          4          2          1
```

If a map file is used, as each string is written out, the map file is searched sequentially until a colour match is made.

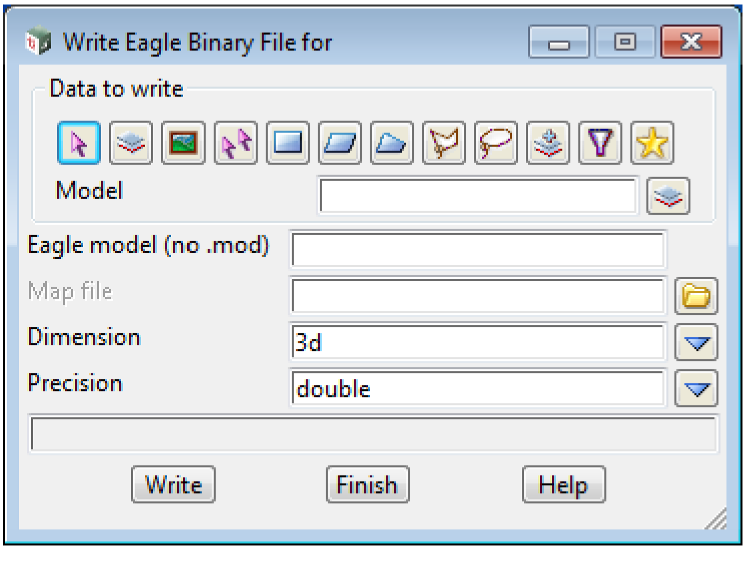
If no match is found, the colour of the string is used as a pen number and default values used for dash, thickness and frag.

Comments can be included in the map file by preceding the comment with a double forward slash (i.e. //). Anything on the line following the // will be ignored.

For example,

// 12d colour	pen	dash	thick	frag
red	3	4	2	1
green	4	1	1	2

On selecting the **Eagle** option, the **Write Eagle Command File for** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source input

source of data is to be written out to a file.

Eagle Command File input *.cmd files

*name of the Eagle command file to write the model/view out to. If the file already exists, the data will be appended to the bottom of it. To be consistent with the Eagle file naming convention, the file name should end in **.cmd***

Eagle model input

*name of the Eagle model to be used for the data. Only the characters A-Z, a-z, 0-9 and - are allowed in an Eagle model name. Do not include a **.mod** in the name.*

Map file input *.emf files

if non-blank, the name of the map file to be used for all string colours. If blank, no map file is used.

Write button

write out the data from the model/view given in the **model/view** *field to the file given in the* **eagle command file** *field. The Eagle model has the name given in the **Eagle model** field. If the file already exists, the data is appended to the file.*

8.2.20.5 Eagle Binary Output

Position of option on menu: **File I/O =>Data output =>Old =>Eagle Binary**

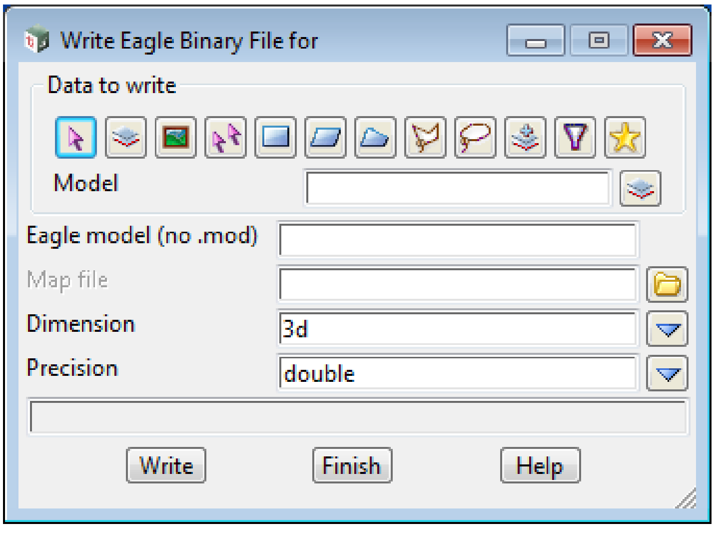
Eagle binary i/o is a separate chargeable module.

Using the **Eagle binary output** option, 12d Model produces Eagle binary models. Since Eagle supports 3D faces, the triangles created in 12d Model can be transferred to Eagle for use in shaded models and walk-throughs.

Since Eagle uses pens, dash styles, line thicknesses and frags rather than colours, an **eagle map file** is used to define the relationships between ***12d Model*** colours and Eagle pens, dash styles, thickness and frags. The map file is the same as described in the **eagle** output option.

The ***12d Model*** string name is written to the second attribute of the corresponding eagle item.

On selecting the **Eagle binary** option, the **Write Eagle Binary File for** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data is to be written out to a file.

Eagle model	input		

stem of the name of the eagle model to write the model/view data out to. Because an eagle model has a maximum size, more than one model may be created. The eagle model names will have a sequence number appended to the name stem given in the **eagle model** *field.*

Map file	input		*.emf files

if non-blank, the name of the map file to be used for all string colours. If blank, no map file is used.

Dimension	input	3d	2d/3d

if 3d, strings will be written out as three dimensional eagle lines. The alignment strings and arcs will be approximated by short lines.

If 2d, the z-value of all the data is set to zero and arcs and curves in alignment strings are written out as eagle plan arcs.

Precision	input	double	single/double

the precision of the eagle models.

Write button

write out the data from the model/view given in the **Model/view** *field to the file given in the* **eagle model** *field.*

8.3 Layouts

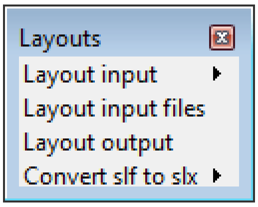
Position of menu: **File I/O =>Layouts**

The options under **Layouts** read and write screen layout files.

The screen layout format to use is the slx format (an XML format) introduced in ***12d Model 10.***

There is also an option to convert the older format, slf, to slx.

The **Layouts** walk-right menu is



read in layout files from working folder, Lib or User Lib
read in one of more layout files
write out a screen layout file
convert a pre V10 layout file (slf) to a V11 layout file (slx)

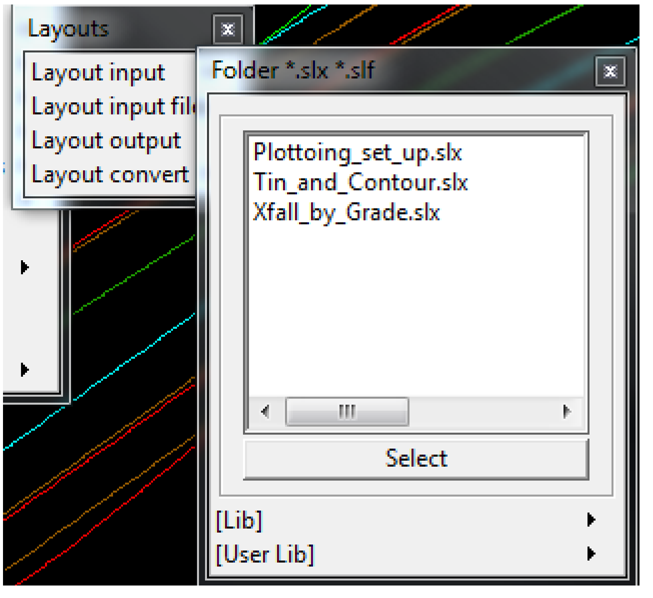
See

Layout input	8.3.1 Layout Input
Layout input files	8.3.2 Layout Input Files
Layout output	8.3.3 Layout Output
Convert slf to slx	8.3.4 Convert slf to slx Format

8.3.1 Layout Input

Position of option on menu: **File I/O =>Layouts =>Layout input**

Walking right on **Layout input** lists all screen layout files (files ending in .slx or .slf) in the working folder, or by further walking right on the [Lib] or [User lib], all screen layout files in [Lib] or [User lib] will be listed.



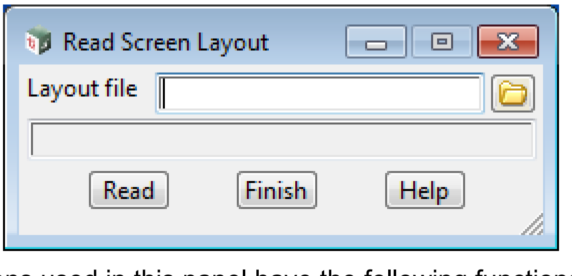
Clicking on a file in a one of the list runs the selected screen layout file and places on the screen, the menus and panels recorded in the screen layout file.

If Clicking on reads in a screen layout file (*.slf) which can create most menus and panels and place them on the screen.

The screen layout file can be created by the **layout output** option, the Menu/Panel Dump option, or by any text editor.

Note

If the *File I/O* menu or the Layouts menu, is pinned, clicking LB on **Layout input** brings up the **Read Screen Layout** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layout file	input		*.slx and *.slf

the name of the screen layout file to read the screen layout information from.

Read	button		

after selecting this button, the screen layout file will be read in.

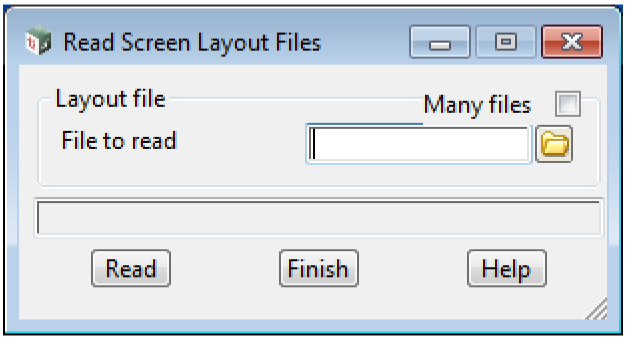
8.3.2 Layout Input Files

Position of option on menu: **File I/O =>Layouts =>Layout input files**

The **layout input files** option reads in one or more screen layout files (files ending in .slx or .slf) which can create most menus and panels and place them on the screen.

The screen layout file can be created by the **Layout output** option or by any editor.

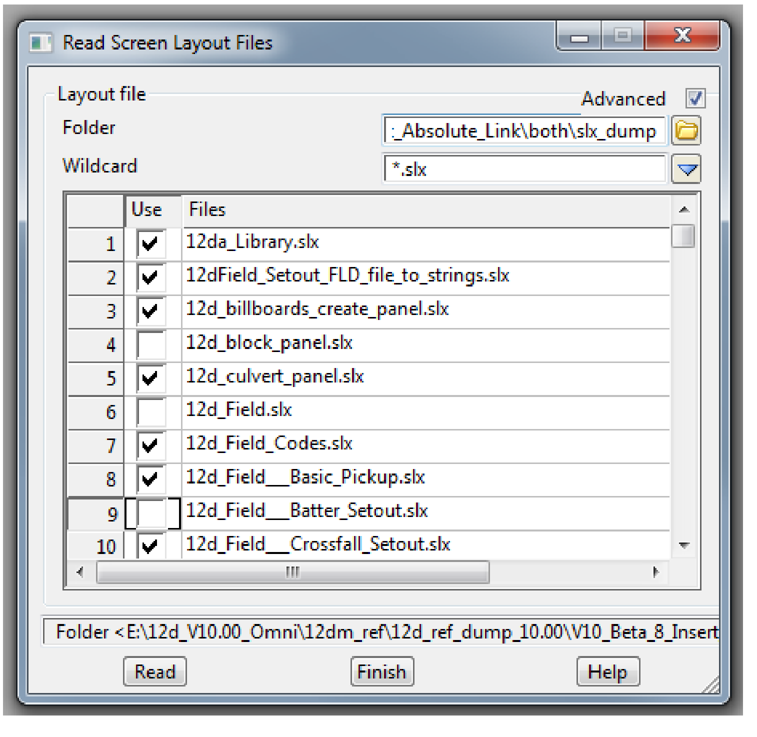
On selecting the **Layout input files** option, the **Read Screen Layout Files** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Advanced	tick box	not ticked	

if ticked, a folder containing slx/slf files can be selected, an optional Wildcard given, and all the slx/slf files satisfying the Wildcard are displayed in a grid. Files in the grid can be selected/not selected.



When **Read** *is pressed, all the selected files are read.*

Layout file	input		*.slx and *.slf

the name of the file to read the screen layout information from.

Read	button		

after selecting this button, the screen layout file will be read in.

8.3.3 Layout Output

Position of option on menu: **File I/O =>Layouts =>Layout Output**

The **layout output** option writes out information on almost all the objects on the screen, and for views, their sizes, into what is called a ***screen layout file***.

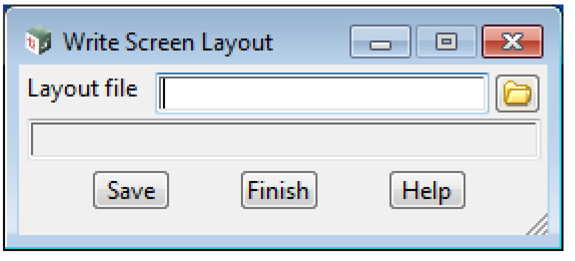
For any menus, its screen position is recorded, and for panels, its position and all the information set for the panel is also recorded.

The screen layout file can then be read back in by options such as **layout input** and **Layout input files**, to recreate on the screen the menus and panels recorded in the screen layout file (but not the views).

Since the screen layout file is editable, users can create their own special layout files.

The view size information is helpful in building up a set_up_file to define the initial screen layout.

On selecting the **Layout output** option, the **Write Screen Layout** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layout file	input		*.slf

the name of the file to print the screen layout information to.

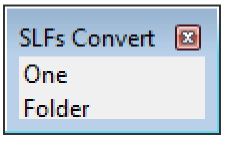
Save	button		

after selecting this button, the screen layout information will be printed out.

8.3.4 Convert slf to slx Format

Position of option on menu: **File I/O =>Layouts =>Layout Convert**

The options on the **Layout Convert** walk-right menu convert the older screen layout format (*.slf) to the new xml format (*.slx) introduced in ***12d Model 9***.
Layout Convert walk-right menu containing these options is:



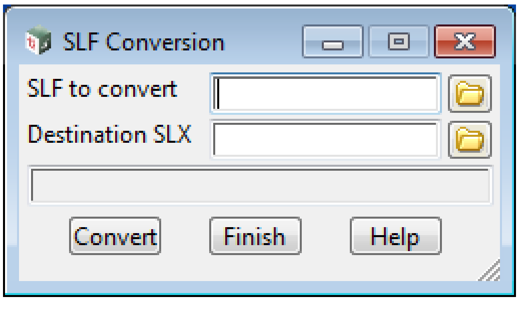
For the option *One*, go to 8.3.4.1 SLF Conversion
Folder 8.3.4.2 Convert a Folder of Screen Layout Files

8.3.4.1 SLF Conversion

Position of option on menu: **File I/O =>Layouts =>Layout Convert=> One**

The **One** option is for converting a single slf to a slx file.

Selecting **One** brings up the **SLF Conversion** panel**.**



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Slf to convert	file box		*.slf

the name of the pre v9 format screen layout file to convert to a XML screen layout file.

When a slf file is selected from the pop-up list, or types in and <enter> hit, then the slf file name, with the slf replaced by slx, is written to the **Destination slx** *field.*

Field Description	Type	Defaults	Pop-Up
Destination slx	file box		*.slx

the slf file is converted to an XML screen layout file and given the name in **Destination slx**.

Field Description	Type	Defaults	Pop-Up
Convert	button		

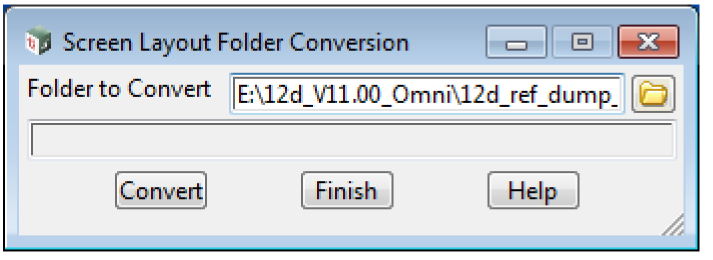
after selecting this button, the slf file will be converted to a slx file

8.3.4.2 Convert a Folder of Screen Layout Files

Position of option on menu: **File I/O =>Layouts =>Layout Convert=> Folder**

The **Folder** option is for converting a folder of slf files to a slx files.

Selecting **Folder** brings up the **Screen Layout Folder Conversion** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Folder to convert	input		*.slf

the folder containing all the slf files to convert to slx files

Convert	button		

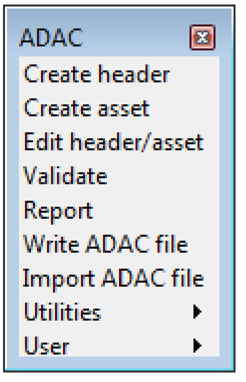
after selecting this button, the all the slf files in the folder **Folder to convert** *are converted to slx files. The converted files are given the same name as the slf files except the slf is replaced by slx.*

8.4 ADAC

Position of menu: **File I/O =>ADAC**

The options under **ADAC** enable use of the new ADAC functionality.

The **ADAC** walk-right menu is



For information on the ADAC menus, see 10.4 12d ADAC Menu

8.5 Digitizer

Position of menu: **File I/O =>Digitizer**

The **Digitizer** option is used with a digitizing table to create 12d Model strings from existing maps or plans.

Contours (2d strings), feature strings (3d strings), spot heights (points strings), 4d strings, circles, arcs and text can all be digitized with this option.

The digitized data can be added to a new or existing 12d Model project.

The digitizing process consists of a number of steps.

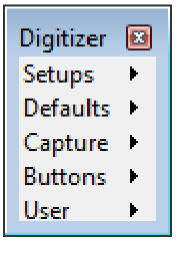
- providing a digitizer definition file
- selecting the type of digitizer to be used
- registering a new plan or map on the digitizer, or continuing with a previously registered plan (resuming).
- selecting a default digitizing tolerance, default model, colour and name for digitized strings.
- digitizing the information

These steps are described totally in the next five sections and are then summarized.

A summary of the step is given in section 8.5.6 Summary.

The digitizer can be set up using WinTab which is preferred method, or by using a definition in the 12d digitizer definitions file, digitize.4d. A quick description of the digitizer definition file is given in the section 8.5.1 Digitizer Definitions. The format for the digitizers_definitions file is given in the section 8.5.7 Digitizer Definitions File.

The **Digitizer** walk-right menu is laid out to reflect the standard sequence of digitizer operations.



define digitizers and plans
default tolerance, model, colour *etc*.
digitize strings
some digitizer button operations

For the option

Setups, go to	8.5.2 Setups
Defaults	8.5.3 Defaults
Capture	8.5.4 Capture
Buttons	8.5.5 Buttons

8.5.1 Digitizer Definitions

Most digitizers now support the WinTab definition for communicating with Windows and that is the preferred method to be used with 12d Model.

If the digitizer supports WinTab then the WinTab drivers need to be installed before the digitizer can be used by 12d Model. The WinTab drivers are supplied with the digitizer, not by 12d Solutions.

Once the WinTab drivers are installed, no further information is required by 12d Model and the rest of this section can be ignored.

Unfortunately if WinTab is not supported, each brand of digitizer has its own method of communicating with a computer and a software package.

To allow for a variety of digitizers not supported by WinTab, 12d Solutions has its own text format for defining the important features of a particular digitizer. The digitizer definitions file is also used to tailor digitizer button usage at a particular site.

For example, the **digitizer file** defines which buttons on the digitizer puck are used to:

digitize new (add) points	**add** button
end the digitizing of a string	**end** button
turn tolerance on/off	**tolerance** button
delete the last digitized point	**delete point** button

When 12d Model starts up, it checks to see if an environment variable called DIGITIZERS_4D exists and if it does, then the file it points to is used to provide the definitions for the digitizers.

If the environment variable is not set, then 12d Model searches for a file called **digitize.4d** in the standard 12d Model search sequence for set up files.

Only digitizers that have been defined in the digitizers definitions file can be selected for use from within 12d Model.

The format for the digitizers definitions file is given in the section 8.5.7 Digitizer Definitions File**.**

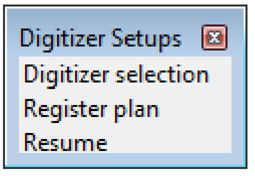
Return to 8.5 Digitizer or look at 8.5.6 Summary.

8.5.2 Setups

Position of menu: **File I/O =>Digitizer =>Setups**

The **setups** menu contains options to select the digitizer to be used, register the co-ordinate system on a new plan or map sheet and for resuming the registration for a plan or map sheet already registered.

The **setups** walk-right menu is



define which digitizer is being used
register a new plan
resume an existing registration

Each options will now be described in detail.

For the option

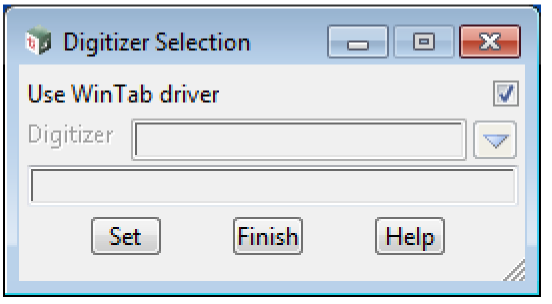
Digitizer selection go to 8.5.2.1 Digitizer Selection
Register plan 8.5.2.2 Register Plan
Resume 8.5.2.3 Resume Plan

For a summary on using the digitizer go to 8.5.6 Summary**.**

8.5.2.1 Digitizer Selection

Position of option on menu: **File I/O=>Digitizer=>Setups=>Digitizer selection**

On selecting the **digitizer selection** option, the **digitizer selection** panel is displayed.



This panel displays the current digitizer selected and allows a different one to be selected from those defined in the digitizers definitions file.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use Wintab drive	tick box		

if ticked then user WinTab drivers.
If not ticked, then the Digitizer field is enabled and must have a digitizer selected.

Digitizer	input	current digitizer	available digitizers

the name of the current digitizer is displayed in this field. The current digitizer is changed by selecting a different name from the pop-up for the **digitizer** *field and then selecting the* **Set** *button.*

This field is only used if ***Use WinTab driver*** *is not ticked.*

Set button

if ***Use Wintab driver*** *is set to tick and the Wintab drivers exist, then the* **Wintab Monitor** *panel is placed on the screen. The WinTab Monitor displays the digitizer co-ordinates and the digitizer button being used. The information describing the use of the digitizer buttons is written to the Output Window (see below). Do not close the Wintab Monitor panel or the digitizer will stop being read.*

If ***Use Wintab driver*** *is set to tick and the Wintab drivers* ***don't*** *exist, an error message is displayed.*

If ***Use Wintab driver*** *is not set to tick, then after selecting this button, the digitizer given in the* **digitizer** *field will be used as the current digitizer.*

Typical Message to Output Window if WinTab Drivers are Installed

Interface: Wintab 32-bit Digitizer Services.
Number of devices: 1
Device 1: GTCO MM-compatible Tablet; Version 1.00.00.08; Format: MM; Firmware: MM III 12 x 12 Tablet by Summagraphics Firmware Version 1.91
The default device is device 1
Number of cursors is 3
Cursor 1: 2 button stylus
Cursor 2: 4 button puck - active
Cursor 3: 16 button puck
Number of buttons on active cursor is 4
Button 1: button 1
Button 2: button 2
Button 3: button 3
Button 4: button 4
Use Button 1 for Add point to string
Use Button 2 for End string
Use Button 3 for Delete point from string
Use Button 4 for Close string

Continue to the next section 8.5.2.2 Register Plan or return to 8.5.2 Setups.

8.5.2.2 Register Plan

Position of option on menu: **File I/O =>Digitizer =>Setups=> Register plan**

Before the information on a plan or map can be digitized, the relationship between the co-ordinate system of the plan and the digitizer must be determined.

In 12d Model, this relationship is given by specifying the plan co- ordinates and then the position on the digitizer of three or more points. These special points are known as control points.

The affine transformation for the control points is then calculated and if accepted by the user, is used to convert digitizer co-ordinates to plan co-ordinates.

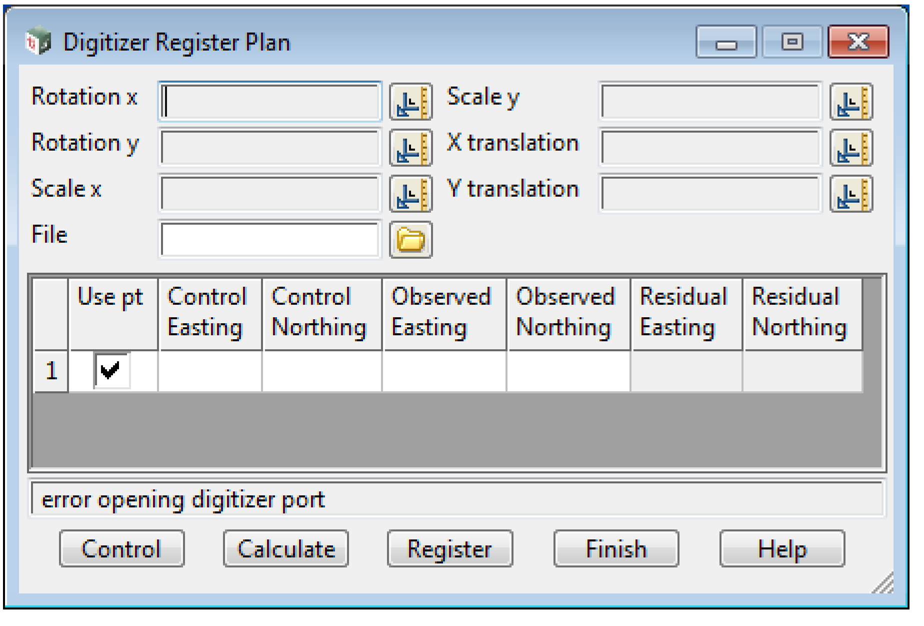
This process is called "registering a plan".

Once a plan has been registered, the information is saved to a file. This file can be used in the **resume plan** option to register the plan without having to re-enter the control points.

Warning

If the plan is moved on the digitizer, it must be re-registered since the current affine transformation will be invalid.

After selecting the **register plan** option, the **digitizer register plan** panel is displayed.



The main purpose of this panel is to display and register the affine transformation parameters. The fields and buttons used in the **digitizer register plan** panel will be discussed after the selection of the control points.

After selecting the **register plan** option, the **digitizer register plan** panel is placed on the screen and the option sits waiting for the user to enter the information for each control point.

This is achieved by

- selecting the control point with the mouse if it already exists in 12d Model, or by typing in its plan co-ordinates (easting northing)

and then

- selecting the control point's position on the digitizer tablet using the digitizers **add point** button. Alternatively, if the control points already exist in 12d Model, they can be selected rather than using typed input

This process is repeated for each control point and is terminated by selecting **cancel** from the **pick ops** pop-up menu instead of selecting a new control point.

The **calculate** button calculates the affine transformation for all the control points in the grid with the Use pt ticked on.

The affine parameters are

- the rotation of the x axis of the plan co-ordinates with respect to the digitizer x axis
- the rotation of the y axis of the plan co-ordinates with respect to the digitizer y axis
- the x and y scale factors for the plan co-ordinates with respect to the digitizer coordinates
- the x and y translations for the plan co-ordinates with respect to the digitizer coordinates

Once calculated, these values are displayed in the **digitizer register plan** panel.

If further control points are required, selecting the **control** button on the **digitizer register plan** panel will restart the control point picking mechanism. Each new control point will be added to the grid.

The new affine transformation can be calculated by selecting the **calculate** button on the **digitizer register plan** panel.

Finally, once the user is satisfied with the affine transformation, it can be **registered** and stored in an text file for future use by entering a filename into the **file** field of the **digitizer register plan** panel and then selecting the **register** button from that panel.

Summarising, the fields and buttons in the **digitizer register plan** panel have the following functions

Field Description	Type	Defaults	Pop-Up
Rotation x/y	output		

the rotation of the x/y axis of the plan co-ordinates with respect to the digitizer x/y axis

Scale x/y	output		

the x/y scale factor for the plan co-ordinates with respect to the digitizer coordinates

Translation x/y	output		

the x/y translation for the plan co-ordinates with respect to the digitizer coordinates

File	input	*.aff	

the name of the file to record the affine parameters.

Control	button		

restarts the selection process for choosing more control points.

Calculate	button		

calculates the affine parameters based on the control point selection.

Register	button		

write the affine transformation to the file given in the **file** *field and register the affine transformation as the current one to be used for the digitising session.*

Finish	button		

remove the panel from the screen.

Warning

The affine transformation must be registered using the **register** button before the **finish** button is selected or it will be lost.

Continue to the next section 8.5.2.3 Resume Plan or return to 8.5.2 Setups.

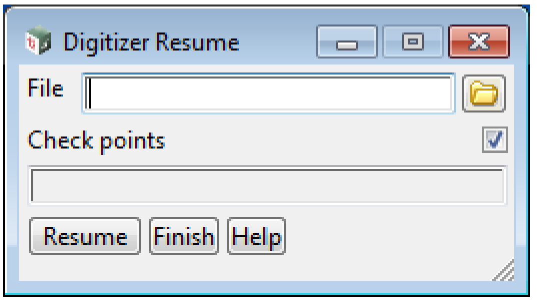
8.5.2.3 Resume Plan

Position of option on menu: **File I/O =>Digitizer =>Setups =>Resume**

When digitising a large plan, it is often necessary to spread the digitising over more than one session in 12d Model.

If the plan has not been moved on the digitizer since it was registered, the affine transformation can be restored with the **resume plan** option rather than having to register the plan again.

On selecting the **resume plan** option, the **digitizer resume** panel is displayed.



To resume digitizing with an affine transformation recorded during an earlier registration session, simply enter the name of the affine file into the **file** field on the **digitizer resume** panel and then select the **resume** button.

Warning

If the plan is moved on the digitizer, it must be re-registered since the current affine transformation will be invalid.

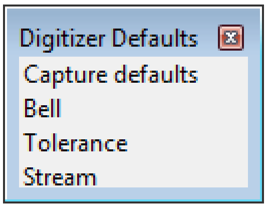
Return to 8.5.2 Setups.

8.5.3 Defaults

Position of menu: **File I/O =>Digitizer =>Defaults**

The **defaults** menu sets default model, colour *etc*. and tolerance which are used when digitizing.

The **defaults** walk-right menu is



default model, colour *etc*.
not yet used
minimum distance between points
set digitizer to stream mode

For the option

Capture defaults, go to	8.5.3.1 Capture Defaults
Bell	8.5.3.2 Bell
Tolerance	8.5.3.3 Tolerance
Stream	8.5.3.4 Stream

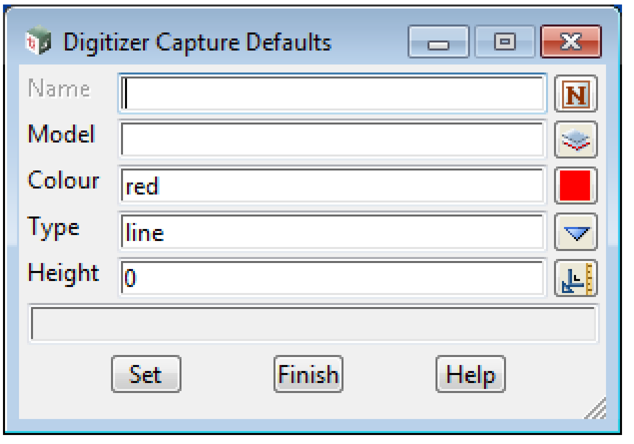
For a summary on using the digitizer go to 8.5.6 Summary**.**

8.5.3.1 Capture Defaults

Position of option on menu: **File I/O =>Digitizer =>Defaults =>Capture defaults**

For any new string a name, model, colour, breakline type and default height are needed. When digitizing, the same values are often required for a large number of the strings. The **capture defaults** panel is used to set default values for digitized strings.

On selecting the **capture defaults** option, the **digitizer capture defaults** panel is displayed.

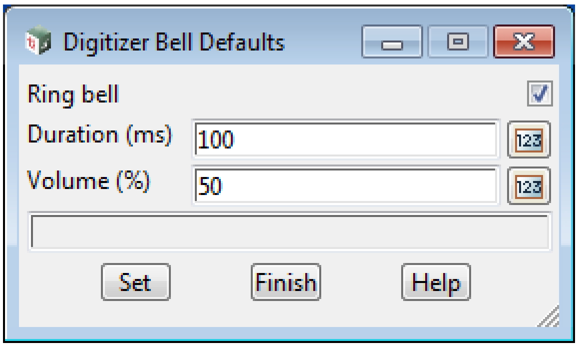


In the digitizing **capture** options, whenever a new string is selected, the values for name, model, colour, breakline type and height are taken from the **digitizer capture defaults** panel.

Continue to the next section 8.5.3.2 Bell or return to 8.5.3 Defaults.

8.5.3.2 Bell

Not yet implemented



Continue to the next section 8.5.3.3 Tolerance or return to 8.5.3 Defaults.

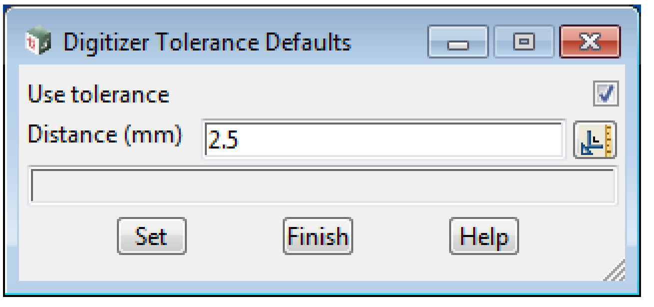
8.5.3.3 Tolerance

Position of option on menu: **File I/O =>Digitizer =>Defaults =>Bell**

When points are being digitized, a new point is often only required when it is a certain distance from the previous point on the digitizer. This is especially important for steam digitizing when hundreds of points per second may be captured.

The **tolerance** option allows the user to specify the minimum plan distance (in millimetres) between successive digitized points in a string.

On selecting the **tolerance** option, the **digitizer tolerance defaults** panel is displayed.



The fields and buttons used in this panel have the following meanings.

Field Description	Type	Defaults	Pop-Up
Use tolerance	tick box	tick	

*if **tick**, a new point taken from the digitizer is only accepted if it is at least the distance given in the* **distance** *field away from the previous recorded point in the string.*

Distance (mm)	input	2.5	

the minimum separation distance, in millimetres, between successive digitized points in a string.

Set	button		

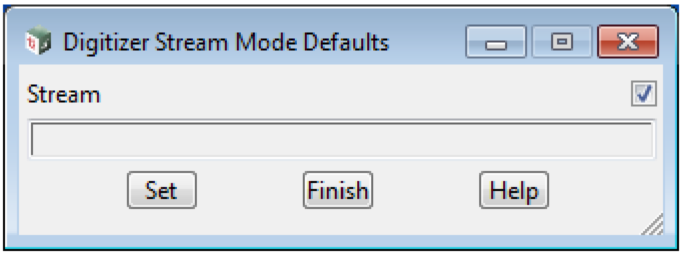
set the **use tolerance** *and* **distance** *values.*

Continue to the next section 8.5.3.4 Stream or return to 8.5.3 Defaults.

8.5.3.4 Stream

Position of option on menu: **File I/O =>Digitizer =>Defaults =>Stream**

Not yet documented.



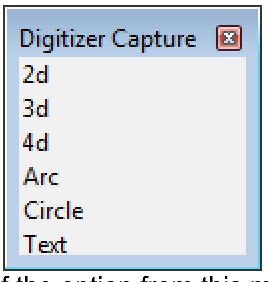
Return to 8.5.3 Defaults.

8.5.4 Capture

Position of menu: **File I/O =>Digitizer =>Capture**

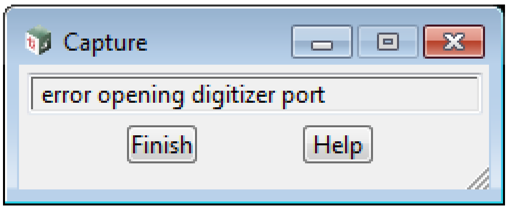
The **capture** menu is for digitizing strings into 12d Model.

The **capture** walk-right menu is



digitize 2d strings
digitize 3d strings
digitize 4d strings
digitize arcs
digitize circles
digitize text

The first time any of the option from this menu is selected, the **capture** panel is placed on the screen.

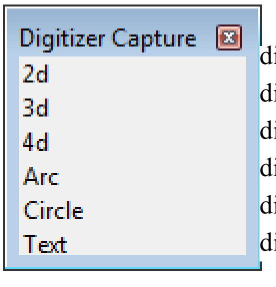


This panel is used to display the transformed co-ordinates for each digitized point that passes the tolerance test, and some special digitizer messages (for example, tolerance on/off).

To **change** the type of string being captured, simply select the new type from the **digitize capture** panel. This is normally only done at the end of capturing a string, not part way through.

To **terminate the digitizing session**, select **finish** on the **capture** panel.

Each of the methods for capturing data available in the **digitizer capture** menu will now be described.



digitize 2d strings
digitize 3d strings
digitize 4d strings
digitize arcs
digitize circles
digitize text

For the option

2d, go to	8.5.4.1 2d
3d	8.5.4.2 3d
4d	8.5.4.3 4d
Arc	8.5.4.4 Arc
Circle	8.5.4.5 Circle
Text	8.5.4.6 Text

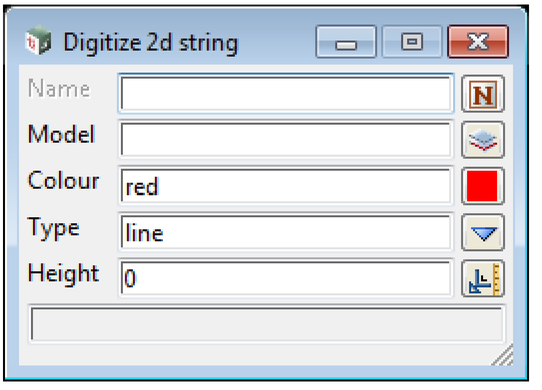
For a summary on using the digitizer go to 8.5.6 Summary.

8.5.4.1 2d

Position of option on menu: **File I/O=>Digitizer=>Capture=>2d**

This option is used to digitize 2d strings.

On selecting the **2d** option, the **Digitize 2d string** panel is displayed.



The fields and buttons in the **digitize 2d string** panel have the following meanings.

Field Description	Type	Defaults	Pop-Up
Name	input	name of new string	name from capture defaults
Model	input	capture defaults model	available models
model for the new string			
Colour	input	capture def colour	available colours
colour of the new string			
Type	input	capture def type	point, line
breakline type of the new string.			
Height	input	capture def height	
height of the 2d string.			

If the **digitize 2d string** panel is created by the **2d** option, the values in the name, model, colour, type and height fields are taken from the **digitizer capture defaults** panel.

Any of the values in the **digitize 2d string** can be modified before digitizing the 2d string begins.

Digitizing begins by pressing the **add** button on the digitizer puck when the puck is above the first point of the new string.

The co-ordinates of the selected point are then displayed in the **capture** panel and the **digitize 2d string** panel removed from the screen.

Further points are digitized by either pressing the **add** button again or if stream mode is on, holding the **add** button down and moving along the string being digitized.

The 2d string is terminated when the **end** button is pressed on the digitizer puck.

After selecting the **end** button, the **digitize 2d string** panel is again placed on the screen with the filed values the same as the 2d string just digitized.

The digitizing process is repeated for the new 2d string.

Continue to the next section 8.5.4.2 3d or return to 8.5.4 Capture.

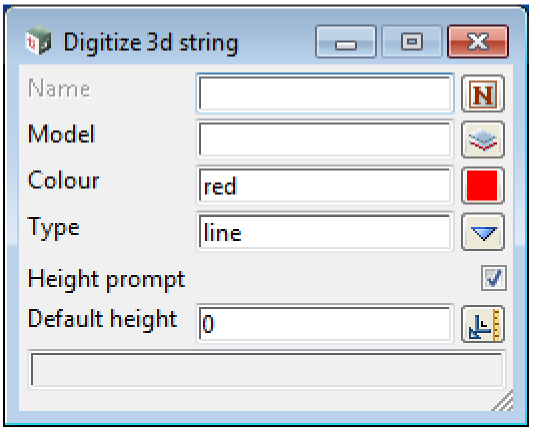
8.5.4.2 3d

Position of option on menu: **File I/O=>Digitizer=>Capture=>3d**

This option is used to digitize 3d strings.

Digitizing 3d strings is similar to 2d strings except that different z-values can exist at each of the digitized points.

On selecting the **3d** option, the **Digitize 3d string** panel is displayed.



The fields and buttons in the **digitize 3d string** panel are similar to the **digitize 2d string** panel except for the fields

Field Description	Type	Defaults	Pop-Up
Height prompt	tick box	tick	

*if **tick**, after each point is digitized, an* enter height *box is placed on the screen. The height of the digitized point is typed into the box.*
*If **not tick**, the* **default height** *is used as the z-value for the digitized point.*

Field Description	Type	Defaults	Pop-Up
Default height	input	capture default height	

height of the point if the height prompt is set to no.

If the **digitize 3d string** panel is created by the **3d** option, the values in the name, model, colour, type and height fields are taken from the **digitizer capture defaults** panel.

As for digitizing 2d strings, any of the values in the **digitize 3d string** can be modified before digitizing begins.

Digitizing begins by pressing the **add** button on the digitizer puck when the puck is above the first point of the new string.

The co-ordinates of the selected point are displayed in the **capture** panel and the **digitize 3d string** panel removed from the screen.

If the **height prompt** field is set to **tick**, an enter height box is placed on the screen with the previous typed height in it. The height for the digitized point in typed into the box, terminated by a <return>. The enter height box then disappears.

Further points are digitized by either pressing the **add** button again or if stream mode is on, holding the **add** button down and moving along the string being digitized.

Again, if the **height prompt** is set to **yes**, the enter height box will appear after each digitized point.

The 3d string is terminated when the **end** button is pressed on the digitizer puck.

After selecting the **end** button, the **digitize 3d string** panel is again placed on the screen with the field values the same as the 3d string just digitized.

The digitizing process is repeated for the new 3d string.

Warning

If stream digitizing is used for the 3d string, the **height prompt** should be set to **not tick**.

Continue to the next section 8.5.4.3 4d or return to 8.5.4 Capture.

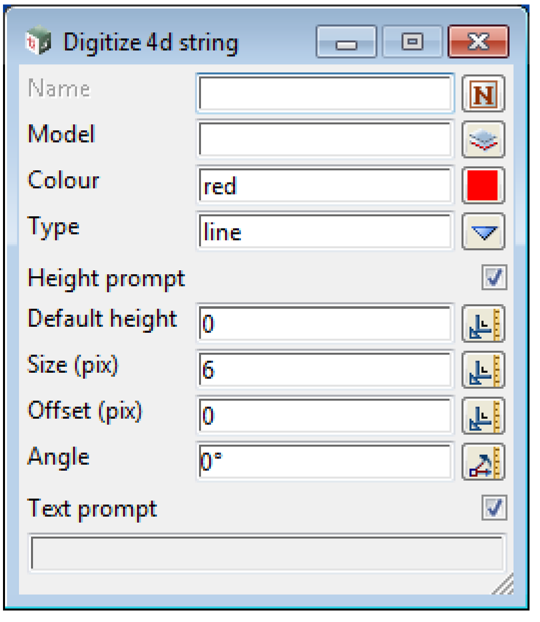
8.5.4.3 4d

Position of option on menu: **File I/O=>Digitizer=>Capture=>4d**

This option is used to digitize 4d strings.

Digitizing 4d strings is similar to 3d strings except that as well as a different z-value at each point, a text string also exits at each of the digitized points.

On selecting the **4d** option, the **Digitize 4d string** panel is displayed.



The fields and buttons in the **digitize 4d string** panel are similar to the **digitize 3d string** panel except for the fields

Field Description	Type	Defaults	Pop-Up
Text prompt	input	yes	yes, no

*if **tick**, after each point is digitized, an* enter text *box is placed on the screen. The text for the digitized point is typed into the box.*
*If **not tick**, no text is used at the digitized point.*

Size (pix)	input	default text size

height (in pixels) of the text at each point.

Angle	input	0

angle of the text at each point.

Offset (pix)	input	0

offset (in pixels) of the text from the digitized point.

If the **digitize 4d string** panel is created by the **4d** option, the values in the name, model, colour, type and height fields are taken from the **digitizer capture defaults** panel.

As for digitizing 3d strings, any of the values in the **digitize 4d string** can be modified before digitizing begins.

Digitizing begins by pressing the **add** button on the digitizer puck when the puck is above the first point of the new string.

The co-ordinates of the selected point are displayed in the **capture** panel and the **digitize 4d string** panel removed from the screen.

If the **height prompt** field is set to **tick**, an enter height box is placed on the screen with the previous typed height in it. The height for the digitized point in typed into the box, terminated by a <return>. The enter height box then disappears.

If the **text prompt** field is set to **tick**, an enter text box is placed on the screen. The text for the digitized point is typed into the box, terminated by a <return>. The enter text box then disappears.

Further points are digitized by either pressing the **add** button again or if stream mode is on, holding the **add** button down and moving along the string being digitized.

Again, the enter height and enter text boxes will appear after each digitized point if the **height prompt** and **text prompts** are set to **tick**.

The 4d string is terminated when the **end** button is pressed on the digitizer puck.

After selecting the **end** button, the **digitize 4d string** panel is again placed on the screen with the values in the fields from the 4d string just digitized.

The digitizing process is repeated for the new 4d string.

Warning

If stream digitizing is used for the 4d string, the **height prompt** and text prompt should be set to **not tick.**

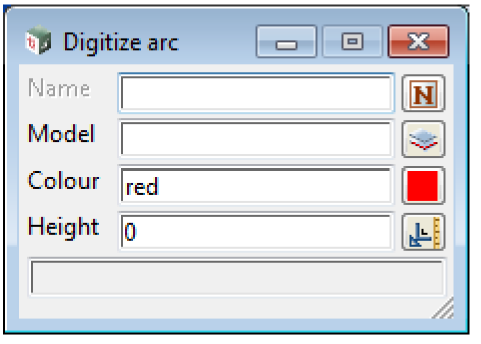
Continue to the next section 8.5.4.4 Arc or return to 8.5.4 Capture.

8.5.4.4 Arc

Position of option on menu: **File I/O =>Digitizer =>Capture =>Arc**

This option is used to digitize arcs by digitizing three points on the arc - the arc start point, a point on the arc and the arc end point.

On selecting the **arc** option, the **Digitize arc** panel is displayed.



The fields and buttons in the **digitize arc** panel are similar to the **digitize 3d string** panel except for the field

Field Description	Type	Defaults
Height	input	capture default height

height given to the two end points of the arc.

If the **digitize arc** panel is created by the **arc** option, the values in the name, model, colour and height fields are taken from the **digitizer capture defaults** panel.

As for digitizing 3d strings, any of the values in the **digitize arc** panel can be modified before digitizing an arc begins.

Digitizing begins by pressing the **add** button on the digitizer puck when the puck is above the first point of the new arc.

The co-ordinates of the selected point are displayed in the **capture** panel and the **digitize arc** panel removed from the screen.

Next a point on the arc between the arc end points is digitized followed by the arc end point.

After the end point is digitized, the arc is automatically created and the **digitize arc** panel again placed on the screen with the field values the same as the arc just digitized.

The digitizing process is repeated for the new arc.

Warning

Stream digitizing should not be used for arcs.

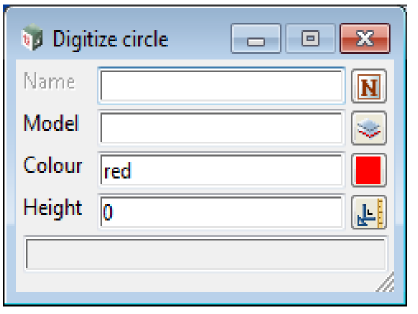
Continue to the next section 8.5.4.5 Circle or return to 8.5.4 Capture.

8.5.4.5 Circle

Position of option on menu: **File I/O =>Digitizer =>Capture =>Circle**

The **circle** option is used to digitize circles by digitizing the centre point and one point on the circle.

On selecting the **circle** option, the **Digitize circle** panel is displayed.



The fields and buttons in the **digitize circle** panel are the same as the **digitize arc** panel.

If the **digitize circle** panel is created by the **circle** option, the values in the name, model, colour and height fields are taken from the **digitizer capture defaults** panel.

As for digitizing arcs, any of the values in the **digitize circle** panel can be modified before digitizing of the circle begins.

Digitizing begins by pressing the **add** button on the digitizer puck when the puck is above the circle point of the new circle.

The co-ordinates of the selected point are displayed in the **capture** panel and the **digitize circle** panel removed from the screen.

Next a point on the circle is digitized. The circle is then fully defined and automatically created.

The **digitize circle** panel is again placed on the screen with the field values the same as the circle just digitized.

The digitizing process is repeated for the new circle.

Warning

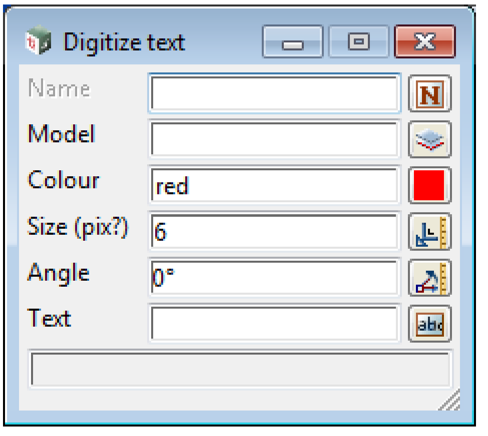
Stream digitizing should **not** be used for circles.

Continue to the next section 8.5.4.6 Text or return to 8.5.4 Capture.

8.5.4.6 Text

Position of option on menu: **File I/O =>Digitizer =>Capture =>Text**

Not yet implemented



Return to 8.5.4 Capture.

For a summary on using the digitizer go to 8.5.6 Summary.

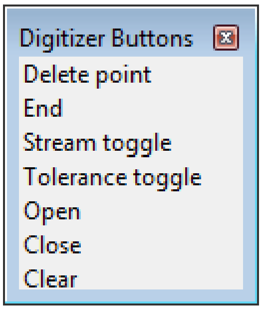
8.5.5 Buttons

Position of menu: **File I/O =>Digitizer =>Buttons**

The various options on the **Buttons** menu can be used instead of pushing buttons on the digitizer puck.

The main use for this is when the digitizer puck has only a small number of buttons.

The **Buttons** walk-right menu is



delete last point
end string
turn stream on/off
turn tolerance on/off
open string
close string
delete all points in string

Return to 8.5 Digitizer.

For a summary on using the digitizer go to 8.5.6 Summary.

8.5.6 Summary

The steps for digitizing are

1. Point to digitizers definitions file

   The file containing the definitions of the digitizers is created and either called **digitize.4d**, or the environment variable DIGITIZERS_4D is set to point to the file.

2. Select a digitizer

   Use the menu item **File I/O =>Digitizer=> Setups=> Digitizer selection**

3. Register a new plan or resume an old plan

   To register a new plan, use menu item **File I/O =>Digitizer => Setups =>Register plan**

   After selecting the option, the software is waiting for the user to give the co-ordinates and position of control points on the digitizer.

   Control Point Loop

   (a) type in easting northing of the control point

   (b) pick point on the digitizer

   The loop is terminated by getting up the **Pick Ops** menu with the mouse and picking **Cancel**

   Then select **calculate** from the **digitizer control point table** panel and check the calculated affine parameters in the **digitizer register plan** panel.

   Once happy with the affine parameters, save them away by typing a filename (ending in **.aff**) into the **file** field of the **digitizer register plan** panel and then select the **register** button.

   Alternatively, to resume digitizing a previously registered plan that has not moved on the digitizing tablet, use menu item **File I/O =>Digitizer =>Setups=> Resume plan**

   Type the affine file name into the **file** field of the **digitizer resume** panel and select **resume**.

4. Set the capture defaults

   Menu items under **Digitizer =>Defaults**

5. Digitize a bounding polygon for the area to be digitized

   Use the menu item **Digitizer =>Capture =>2d**

   This step is not strictly necessary, however, the polygon can be used to easily **fit** the digitized data onto a view.

6. Digitize the data

   Use the menu items under **Digitizer =>Capture**

Return to 8.5 Digitizer.

8.5.7 Digitizer Definitions File

Unfortunately, each brand of Digitizer has its own method of communicating with a computer and a software package.

To allow for a variety of digitizers, 4D Solutions has its own text format for defining the important features of a particular digitizer.

When 12d Model starts up, it checks to see if an environment variable called DIGITIZERS_4D exists and if it does, then the file it points to is used to provide the definitions for the digitizers.

DIGITIZERS_4D filename

If the environment variable is not set, then 12d Model searches for a file called **digitize.4d** in the standard 12d Model search sequence for set up files.

Only digitizers that have been defined in the digitizer definitions file can be selected for use from within 12d Model.

The digitizers definition file format is a simple text format and consists of one or more digitizer definitions. Each digitizer definition in the file begins with the key word **digitizer** followed by the digitizer name and then appropriate digitizer commands enclosed within curly braces **{ }**.

8.5.7.1 Available Digitizer Commands

Commands to define communication with the digitizer:

port	text
baud	300 | 600 | 1200 | 2400 | 4800 | 9600 | 19200| 38400
charbits	5 | 6 | 7 | 8
parity	none | even | odd
stopbits	1 | 2

Commands to define the size of the digitizing area:

xdimension	integer	// length of digitizer area in mm
ydimension	integer	// height of digitizer area in mm
resolution	real	// resolution in mm

Commands to define contents of the text string returned when digitizing:

stringlength	integer	// length of the text string returned
buttonstart	integer	// position in text for button id
buttonend	integer	
xstart	integer	// position for x value
xend	integer	
ystart	integer	// position for y value
yend	integer	

Command to define sequences to send to digitizer:

startup	text	// digitizer start up sequence
finishup	text	// digitizer finish up sequence
pointmode	text	// text to put digitizer into point mode
streammode	text	// text to put into stream mode (track mode)

Commands to define the meaning of the button values returned from the digitizer:

These commands may occur more than once since some digitizers have more than one return value for the same thing.

addpoint	text	// digitize a point
end	text	// end the string being digitized
delpoint	text	// delete the last point digitized
toggletol	text	// toggle tolerance
togglestream	text	// toggle stream mode
clearstr	text	// clear all points in the current string
openstr	text	// open the current string
closestr	text	// closes the current string

Notes

1. spaces in text - any text string that includes spaces or only numbers, must be enclosed in double quotes **“**.
2. comments - anything after // until the end of the line is ignored.
3. blank lines - blank lines are ignored

Return to 8.5 Digitizer.

For a summary on using the digitizer go to 8.5.6 Summary**.**

8.6 GIS

Position of menu: **File I/O =>GIS**

The **FDO (Feature Data Objects**) is used to communicate with external servers such as **GIS** packages and external databases, including **SQL Server**, **Oracle** and **MySQL**.

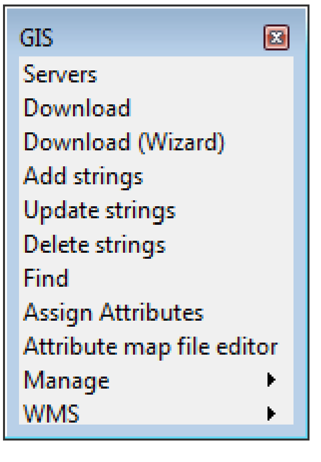
Depending on the services provided by the external data source, vector data may be downloaded, updated or new data inserted. Raster images may also be dynamically downloaded and displayed via the W**MS (Web Mapping Service).**

The list of services ***12d Model*** currently supports includes:

- WFS (Web Feature Service)
- ArcSDE
- Oracle databases
- SQL Server databases
- MySQL databases
- Basic ODBC (Open Database Connectivity), such as Excel and Access
- PostGreSQL
- SQLite
- PostGIS
- Shape files
- SDF files
- OGR formats (see http://www.gdal.org/ogr/ogr_formats.html)
- WMS (Web Mapping Service) for Raster downloads

Some server types may not be available if you do not have the required third party components, as installed with the third party software. You may need to contact the vendors of the data source if you experience difficulty.

The **GIS** walk-right menu containing these options is:

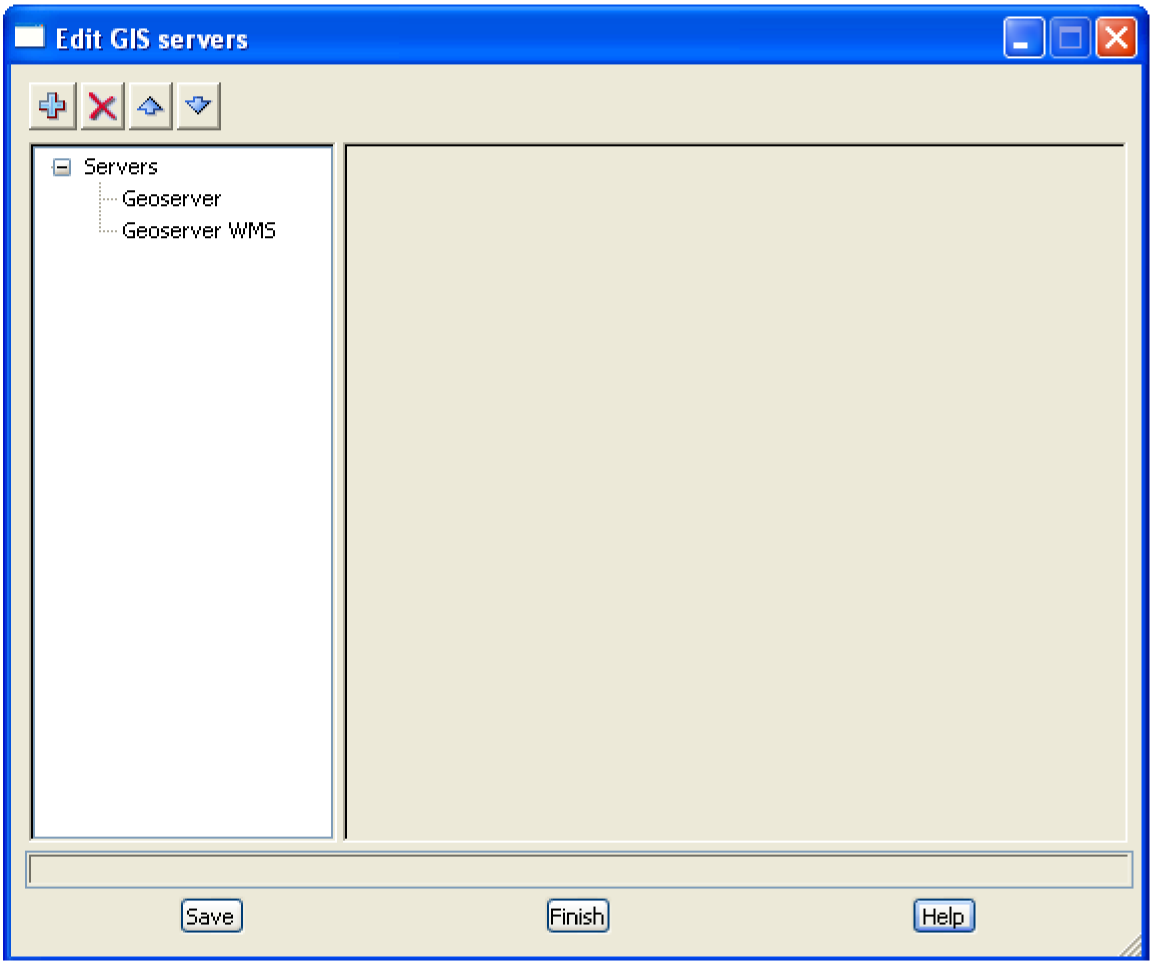


For the option Servers, go to	8.6.1 Edit GIS Servers
Download	8.6.7 GIS Download
Download (Wizard)	8.6.10 GIS Download Wizard
Add strings	8.6.14 GIS Add Strings
Update strings	8.6.15 GIS Update Strings

8.6.1 Edit GIS Servers

Position of option on menu: **File I/O =>GIS =>Servers**

Selecting **Servers** brings up the **Edit FDO servers** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Insert	button		
Delete	button		
Up	button		
Down	button		
Save	button		

Insert — button

Adds a new server

Delete — button

Deletes the selected server

Up — button

Moves the server up in the list

Down — button

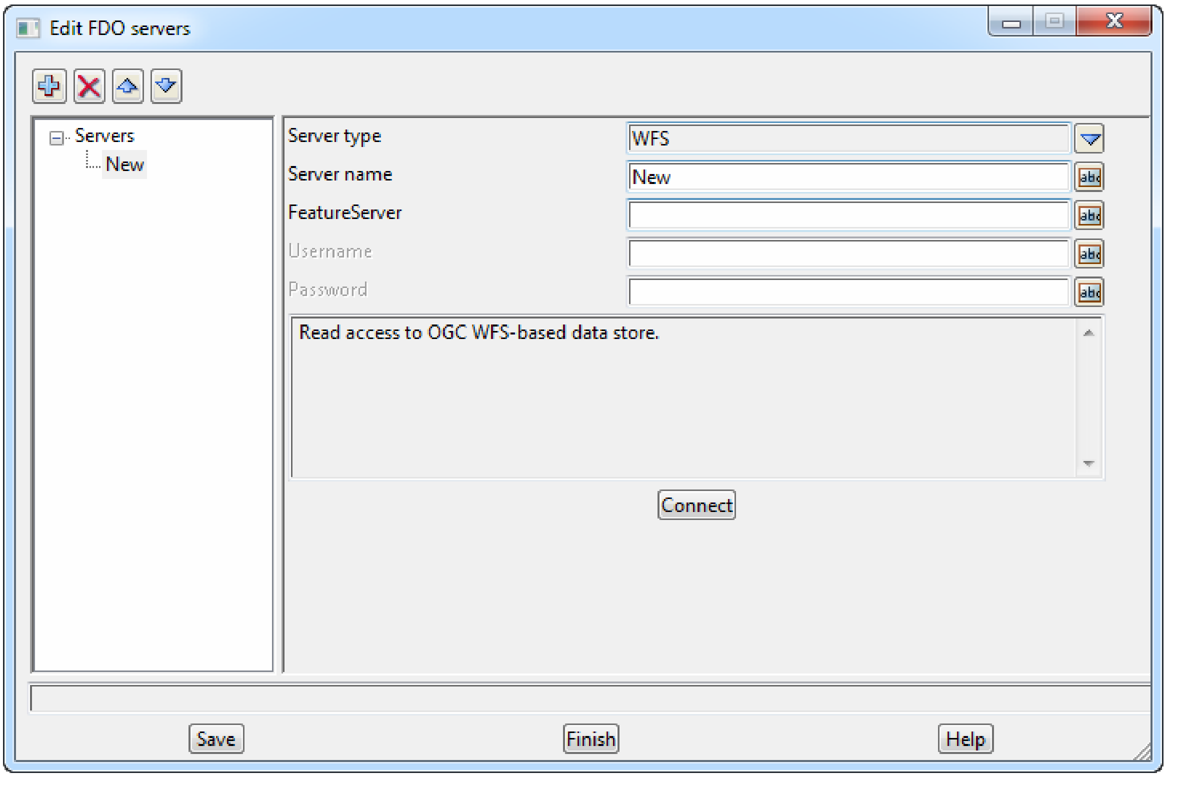
Moves the server down in the list

Save — button

Saves the list of servers.

8.6.2 WFS Server Setup

This panel provides access to edit the required information for connection to a **WFS (Web Feature Service)** server.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Server type	choice box		

the type of server

Server name

the name by which you will refer to the server

FeatureServer

the HTTP address of the server

Username

an optional username

Password

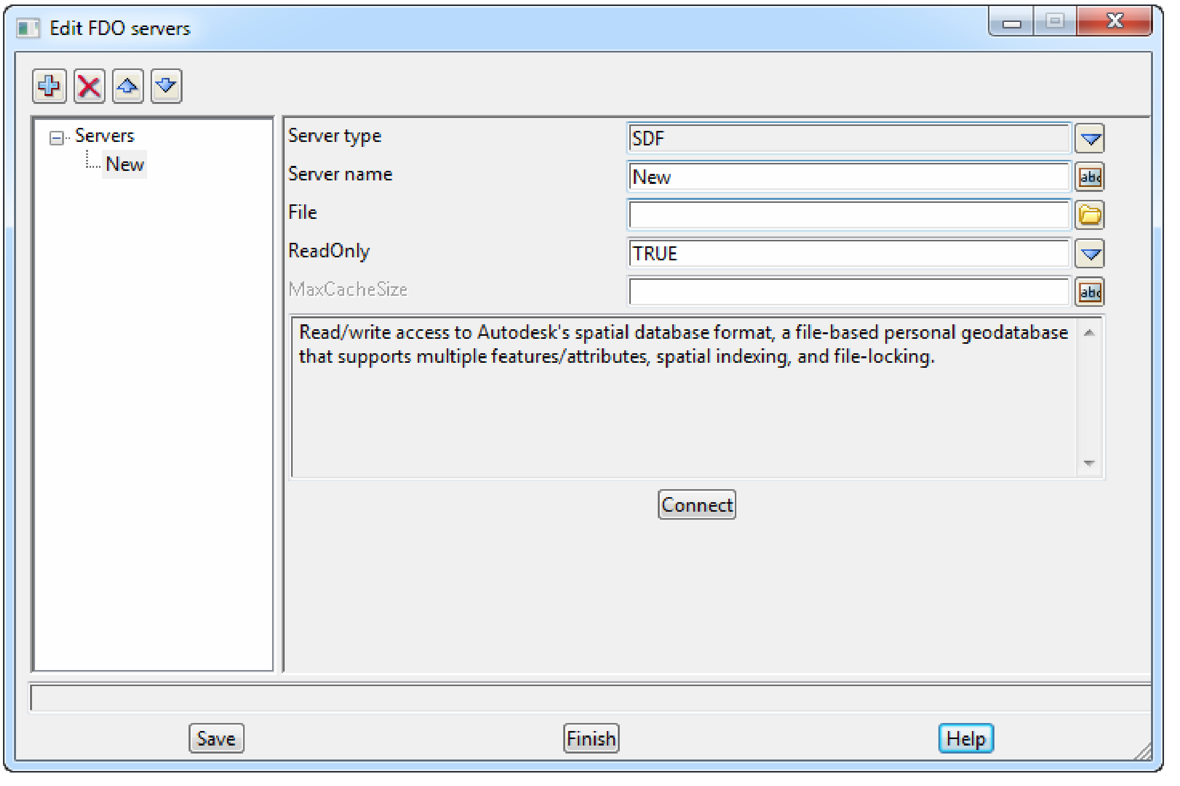
an optional password

Connect button

Tests the connection to the server

8.6.3 SDF Server Setup

This provides access to edit the required information for connection to a file in **AutoDesk's SDF format (spatial database format)**.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Server type	choice box		

the type of server

Server name

the name by which you will refer to the server

File	file		

the file to read / write

ReadOnly	choice box		True, False

whether or not to access the file in a read only fashion

MaxCacheSize

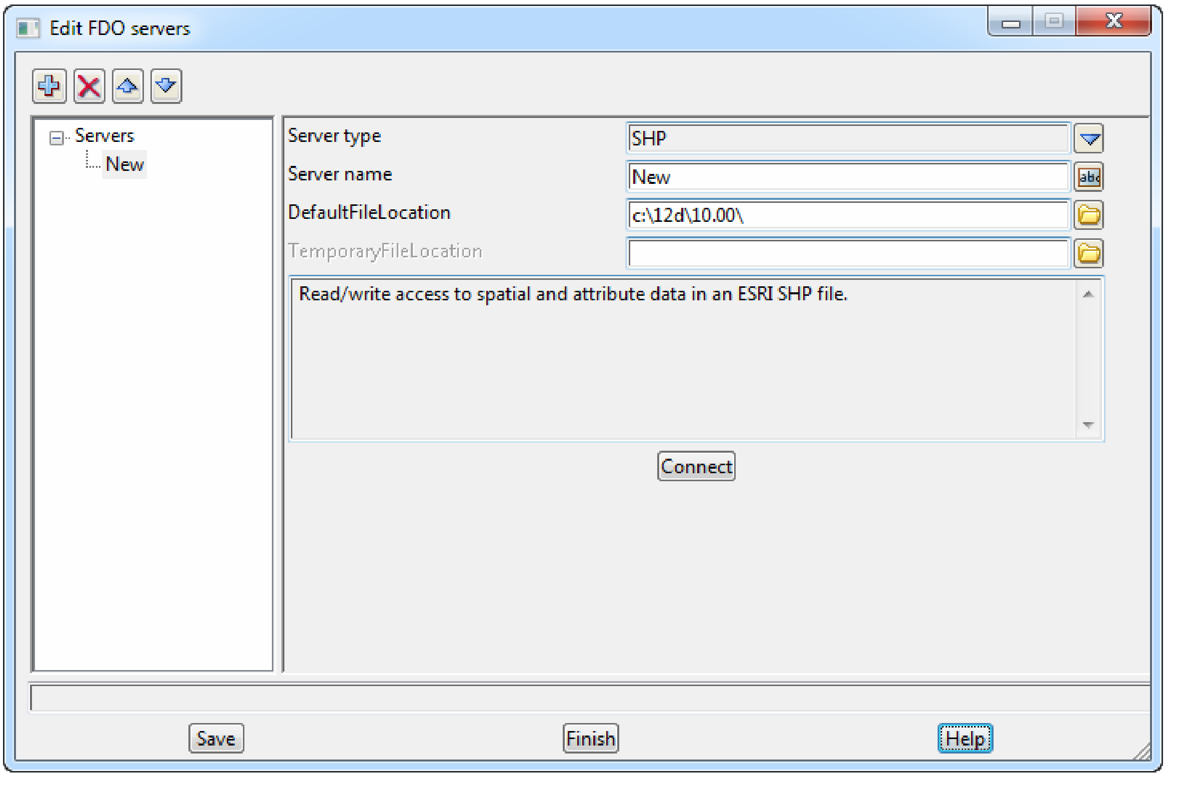
an optional setting to determine the maximum cache size used in ***SDF*** *operations*

Connect	button		

Tests the connection to the file

8.6.4 SHP Server Setup

This provides access to edit the required information for connection to a file or set of files in **ESRI's SHP** file format.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Server type	choice box		

the type of server

Server name

the name by which you will refer to the server

DefaultFileLocation file

where the SHP file(s) to read/write reside

TemporaryFileLocation file

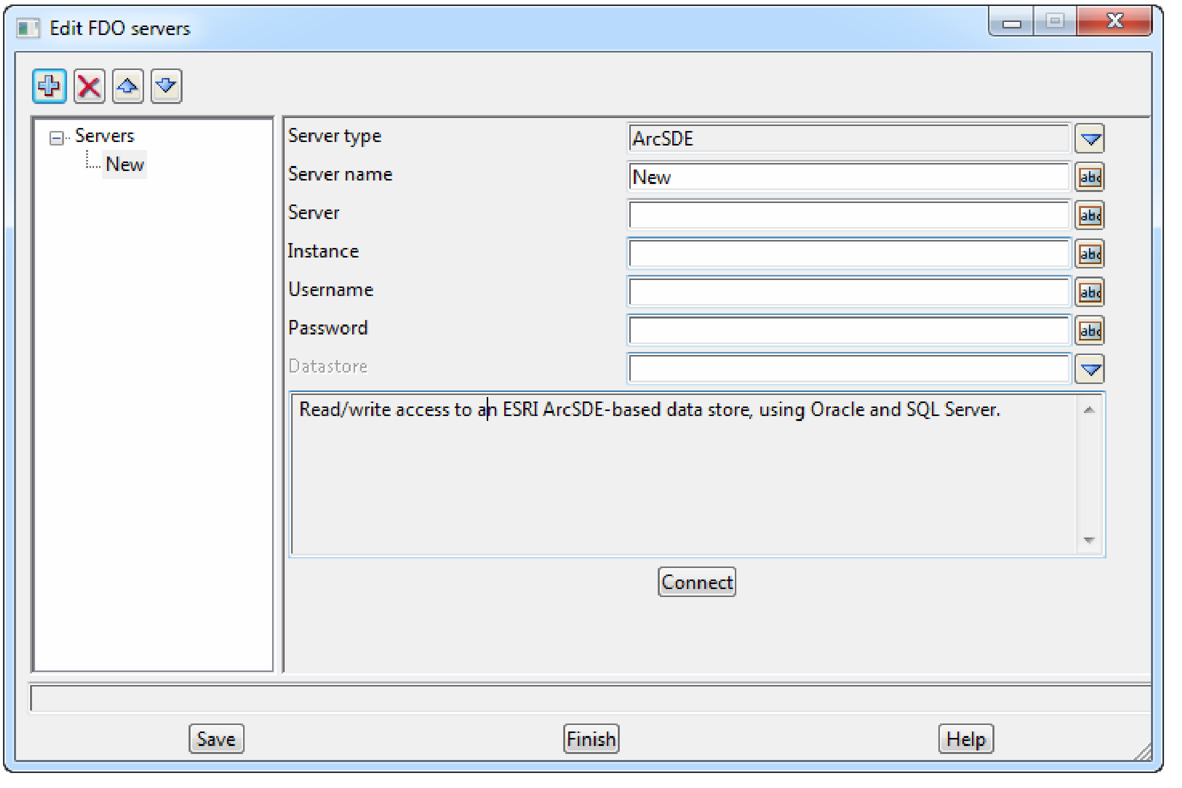
*an optional field to specify where the **FDO** should write any required temporary files*

Connect button

Tests the connection settings

8.6.5 ArcSDE Server Setup

This provides access to edit the required information for connection to an **ArcSDE** server. If you do not know the data store name, you may need to connect to the server once to retrieve the list of names.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Server type	choice box		

the type of server

Server name

the name by which you will refer to the server

Server

the address of the remote ArcSDE server

Instance

the instance name or port number

Username

the username under which you will connect to the server

Password

the required password for connecting to the server

Datastore

*if a datastore is required, you must enter it here. If you do not know the name of the datastore, click **Connect** and the list of datastores will be retrieved*

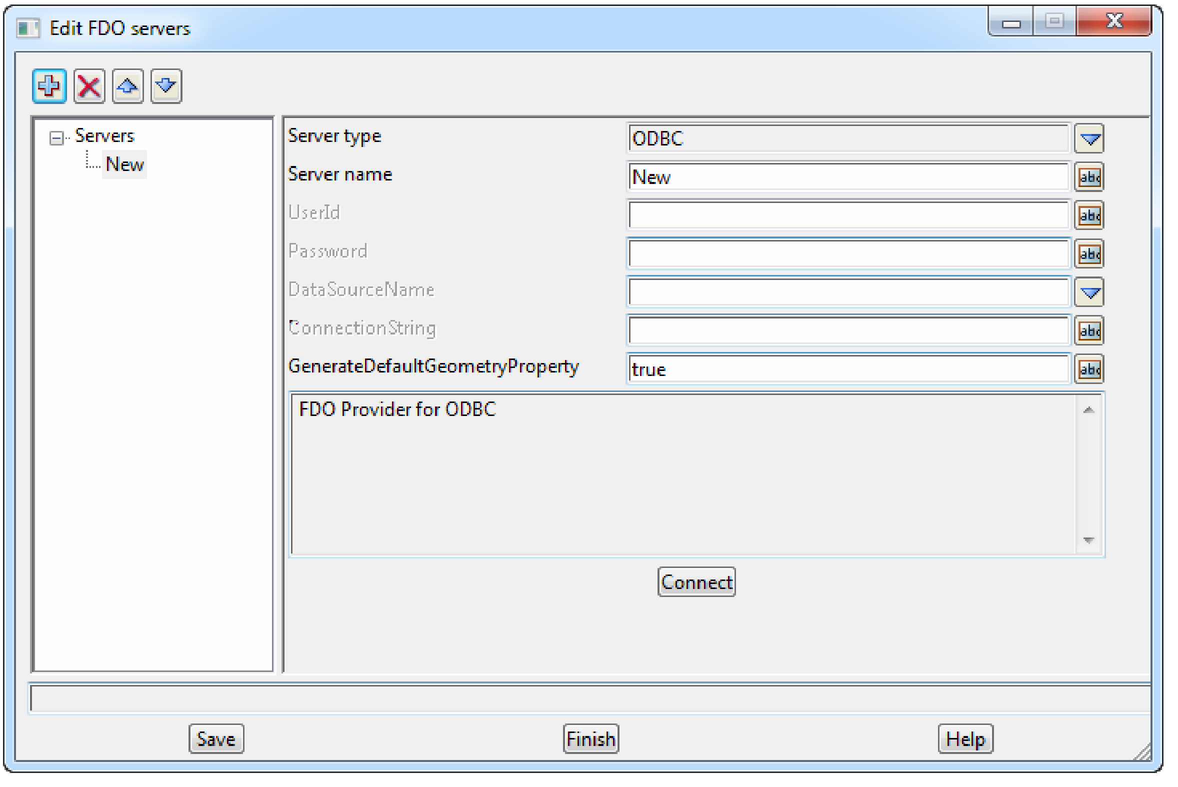
Connect button

Tests the connection to the server and retrieves the list of datastores

8.6.6 ODBC Server Setup

This provides access to setup a connection to a generic **ODBC (Open Database Connectivity)** data source, such as **Microsoft Excel** or **Microsoft Access**.

Because this is a generic connection type, you will be required to know the necessary combination of settings including DSN or Connection String. This may involve knowing the specific driver required to connect to the data source.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Server type	choice box		

the type of server

Server name

the name by which you will refer to the server

UserId

an optional user id for the connection

Password

an optional password for the connection

DataSourceName

an optional data source name (or DSN)

ConnectionString

an optional connection string

GenerateDefaultGeometryProperty

whether or not to generate the default geometry property if one does not exist

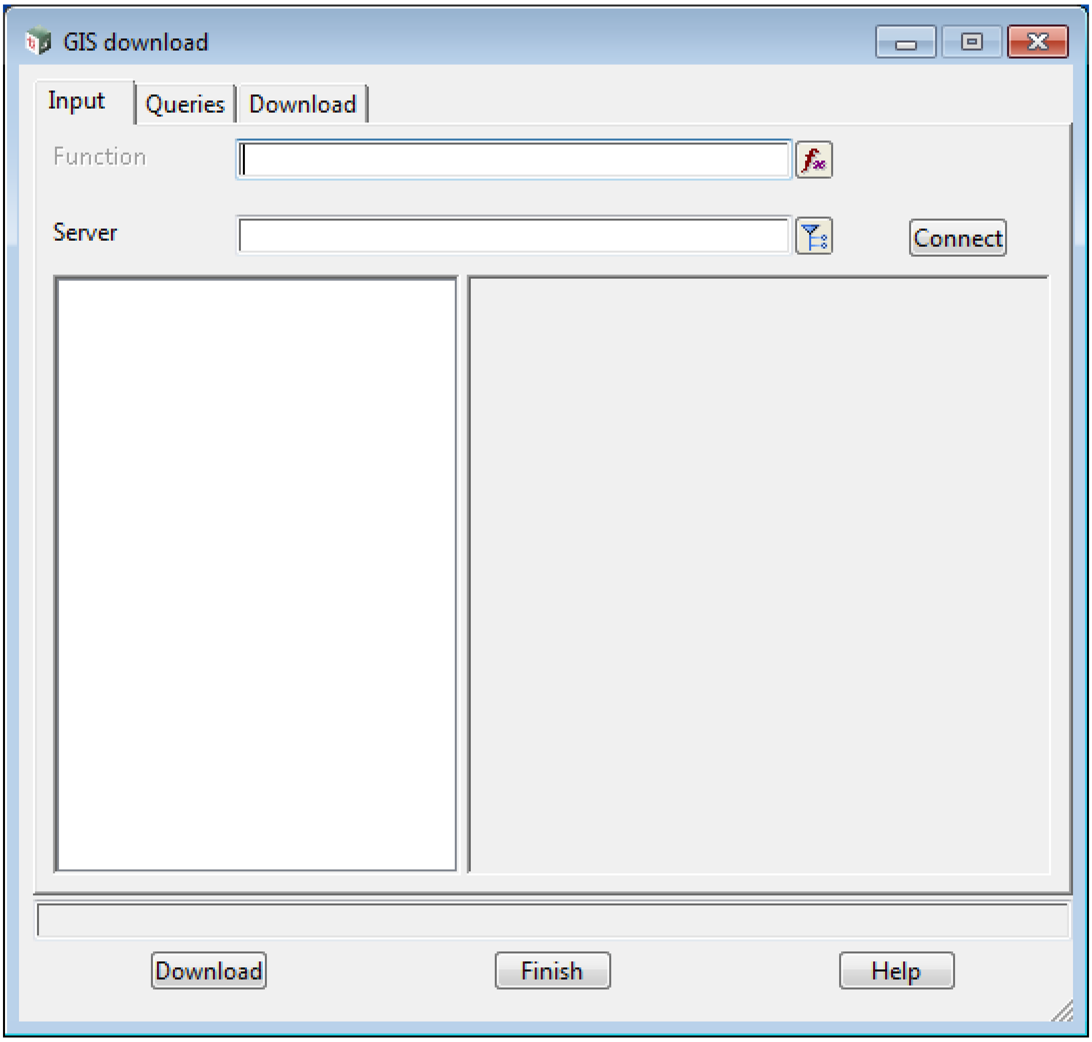
Connect button

Tests connection to the data source

8.6.7 GIS Download

Position of option on menu: **File I/O =>GIS =>Download**

Selecting **Download** brings up the **GIS download** panel.



For a description of the *Input* tab, see Input tab

For a description of the *Queries* tab, see Queries tab

For a description of the *Download* tab, see Download tab

For a description of the *Download* button, see Download button

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Input tab

Function	function box		available GIS functions

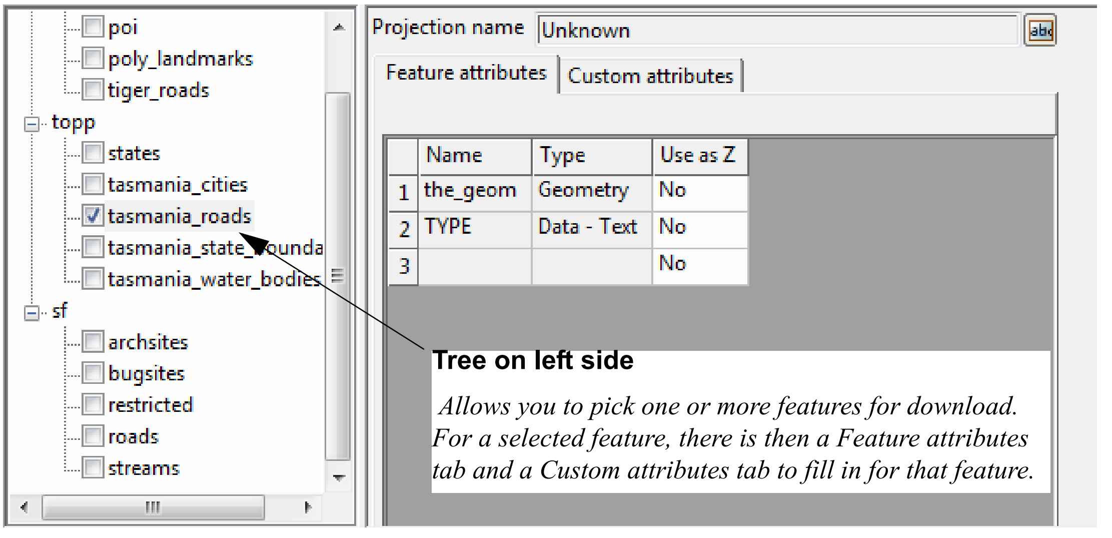
an optional function. A function is required for edits and inserts

Server

the server you wish to connect to

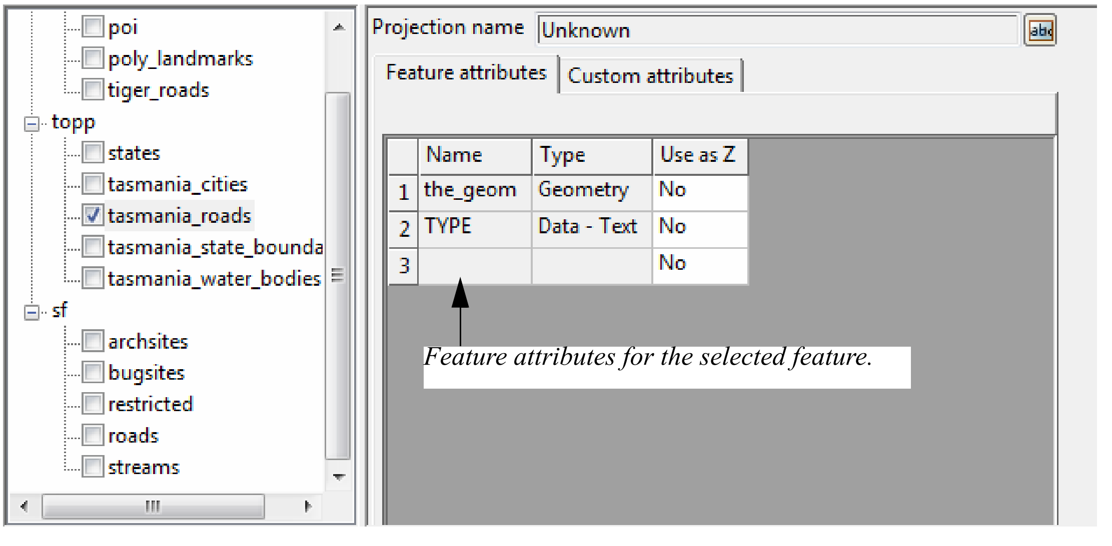
Connect button

connects to the server and retrieves the list of features available for download. The list of features is displayed as a tree control on the left hand side of the panel.



Input tab: Feature Attributes Tab

The Feature attributes grid displays the set of attributes to be downloaded as part of the selected feature.



Name

the name of the downloaded feature attribute

Type

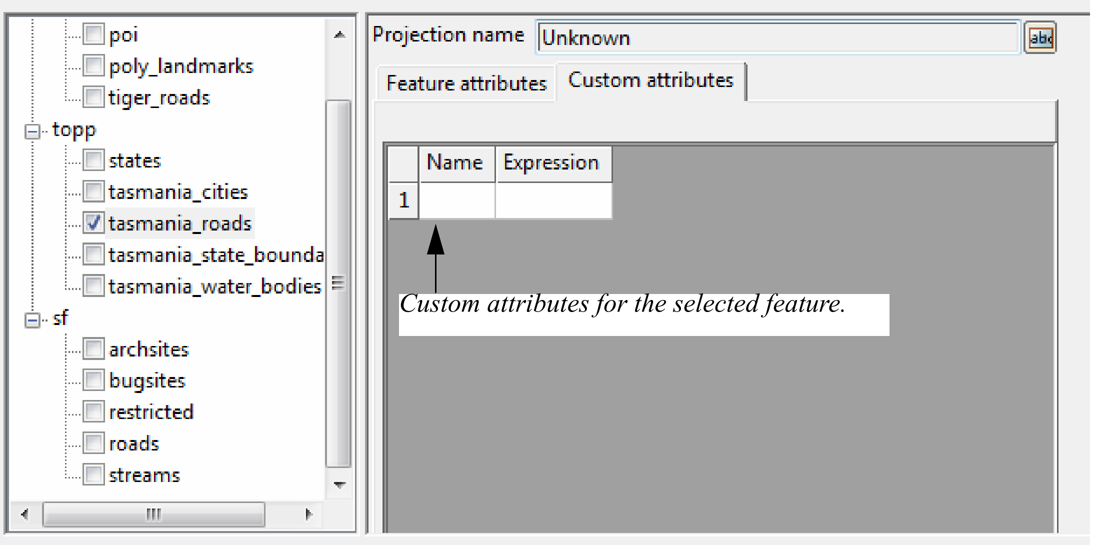
the type of the downloaded feature

Use as Z

whether or not this attribute should be used as a Z level on strings

Input tab: Custom Attributes tab

The Custom attributes grid defines the set of attributes to be downloaded as part of the selected feature.



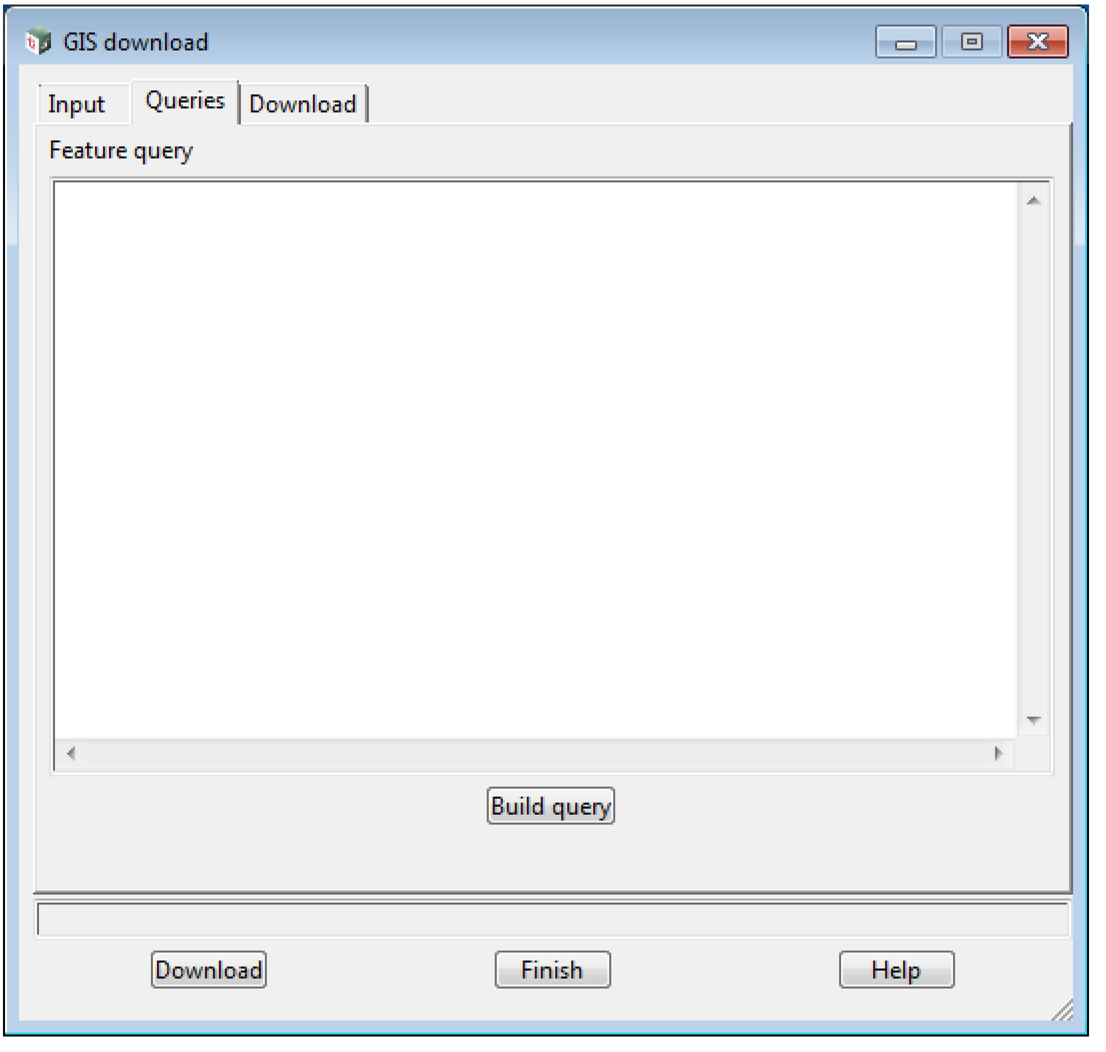
Name

the name of the attribute to be downloaded

Expression

the expression to calculate the attribute (see 8.6.8 Query Building*)*

Queries tab



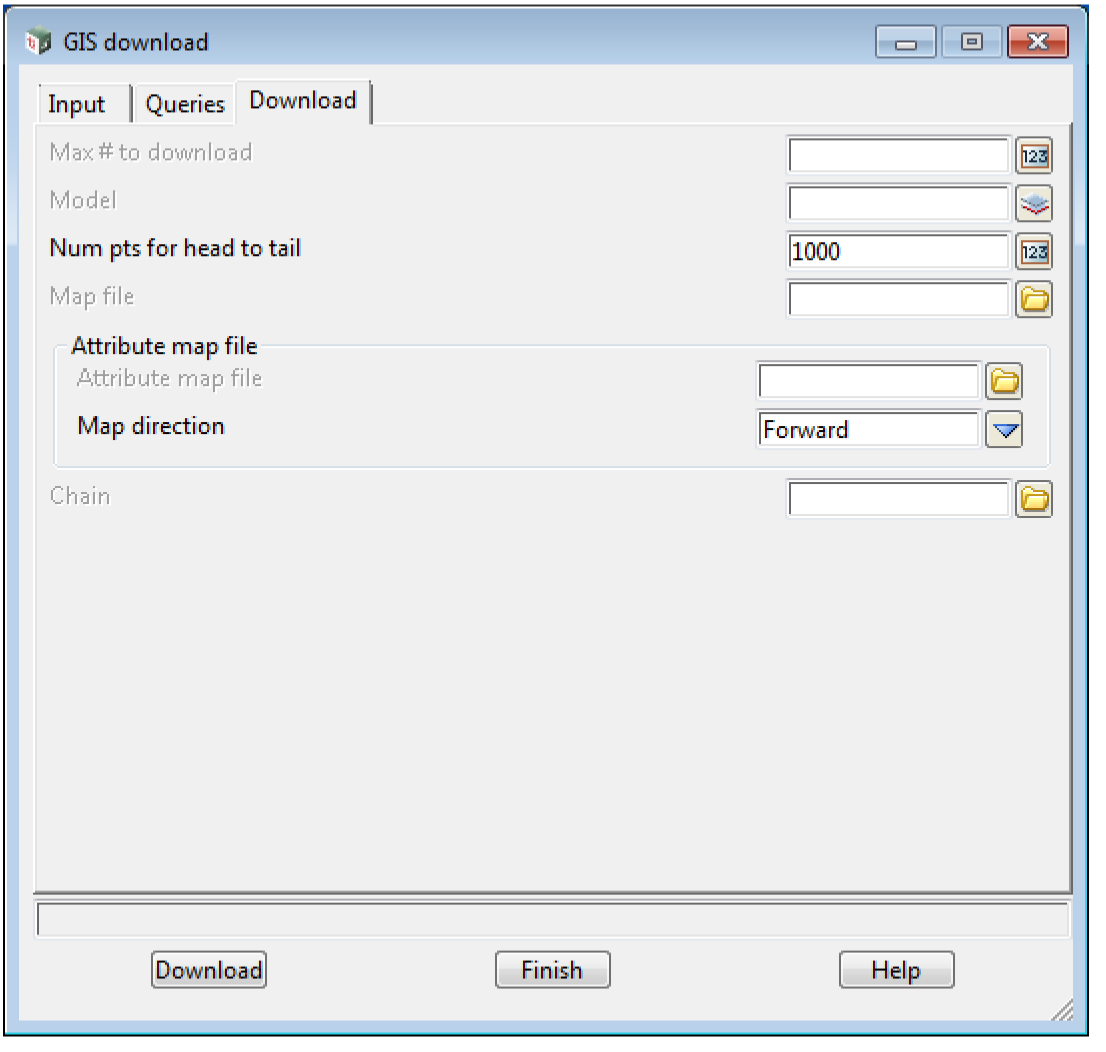
Feature Query

an optional query to send to the server for downloads

Build Query button

shows a helper panel for building queries (see 8.6.8 Query Building*)*

Download tab



Max # to download

the maximum number of strings to download

Model model box

an optional model to download into

Num pts for head to tail

an optional number of points for head to tailing point strings

Map file

an optional map file to apply to downloaded strings

Attribute map file

an optional attribute map file to apply to attributes

Map direction choice box Forward, Backward

the direction to apply to the attribute map file

Chain file box available *.chain files

an optional chain to run after the download completes. This is only available if not run in a function.

Download **button**

downloads the requested features from the server

8.6.8 Query Building

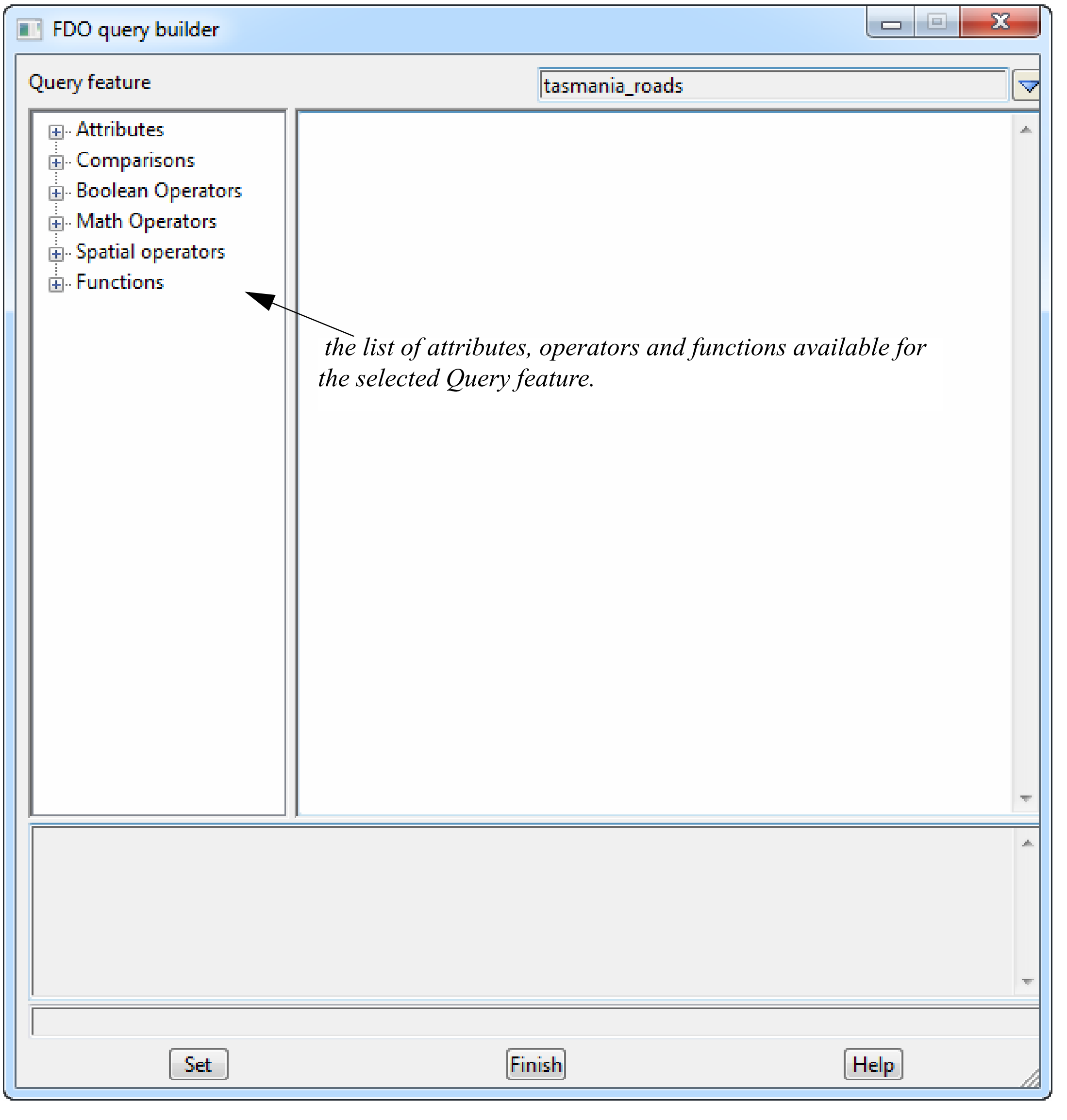
This panel helps build queries for constraining feature downloads.

Note that the features, attributes and queries available are determined by the server you are connecting to and will not be documented here.

If the server publishes help information about the query operation, then it will be listed at the bottom of the panel.

Select **Build query** 

to bring up the FDO query builder panel



Query feature

the feature to build a query for

Left side

the list of attributes, operators and functions available

Right side

the current query

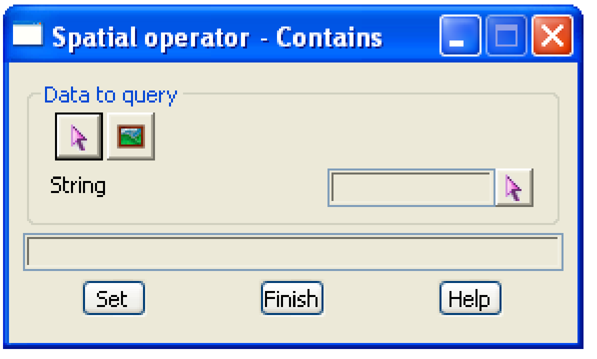
Bottom

any description of the operator

8.6.9 Spatial Operators

Spatial operators are special operators that allow you to refer to an operation against existing geometry. Examples may include whether or not the geometry attribute contains, crosses, intersects or touches another piece of geometry.

For these operators, a panel in the following style will be presented.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Data to query

the data to operate against

Set

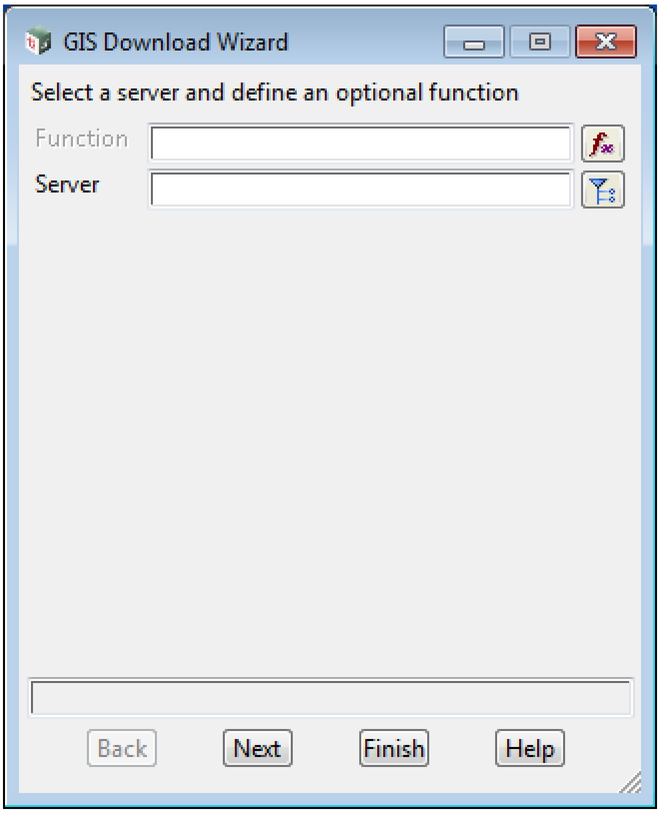
Sets the data for the query

8.6.10 GIS Download Wizard

Position of option on menu: **File I/O =>GIS =>Download (Wizard)**

This is a wizard style interface to assist in setting up an GIS download.

Selecting **Download (Wizard)** brings up the **FDO Download Wizard** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Throughout the wizard, the following buttons **Back** and **Next** will be available

Back

Takes you back to the last step

Next

Takes you to the next step

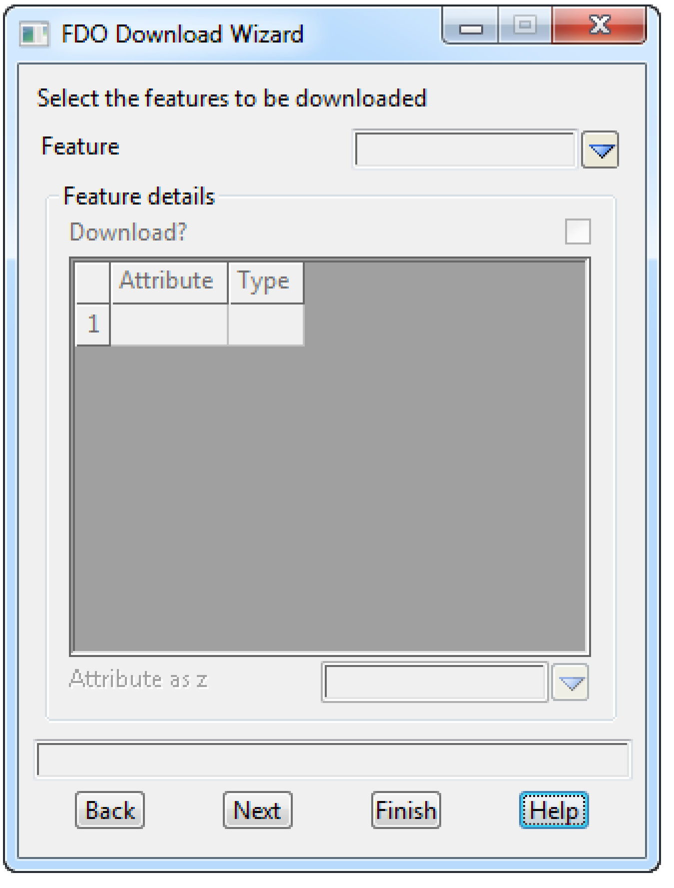
Function

an optional function to create for this FDO download

Server

the server to connect to

8.6.11 Feature Selection Step



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Feature choice box

the feature to view settings for

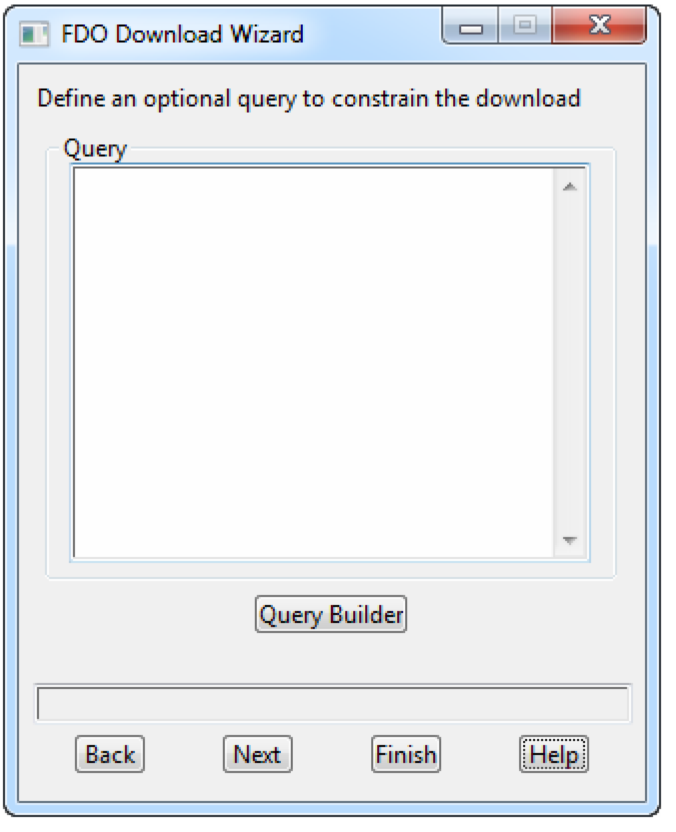
Download? tick box

whether or not to download this feature

Attribute as z

which attribute, if any, to use as the z level for strings

8.6.12 Query Step



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

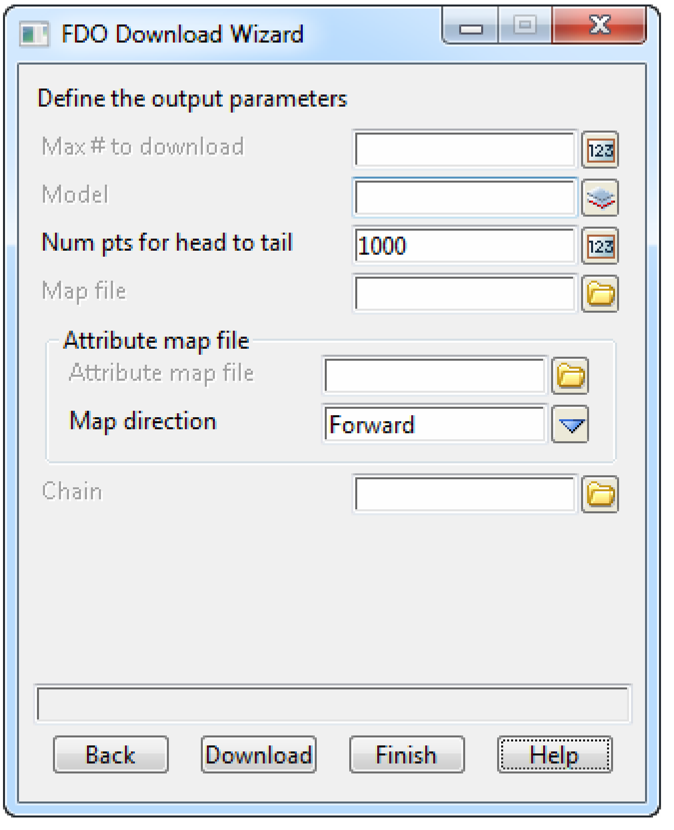
Query

the query to send to the server to constrain features

Query Builder

opens the **FDO Query builder** *panel. See the section on* 8.6.8 Query Building

8.6.13 Output Parameters Step



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Define the Output parameters

Max # to download

the maximum number of strings to download

Model model box

an optional model to download into

Num pts for head to tail

an optional number of points for head to tailing point strings

Map file file

an optional map file to apply to downloaded strings

Attribute map file

Attribute map file file

an optional attribute map file to apply to attributes

Map direction choice box Forward, Backward

the direction to apply to the attribute map file

Chain file

an optional chain to run after the download completes. This is only available if not run in a function.

Download button

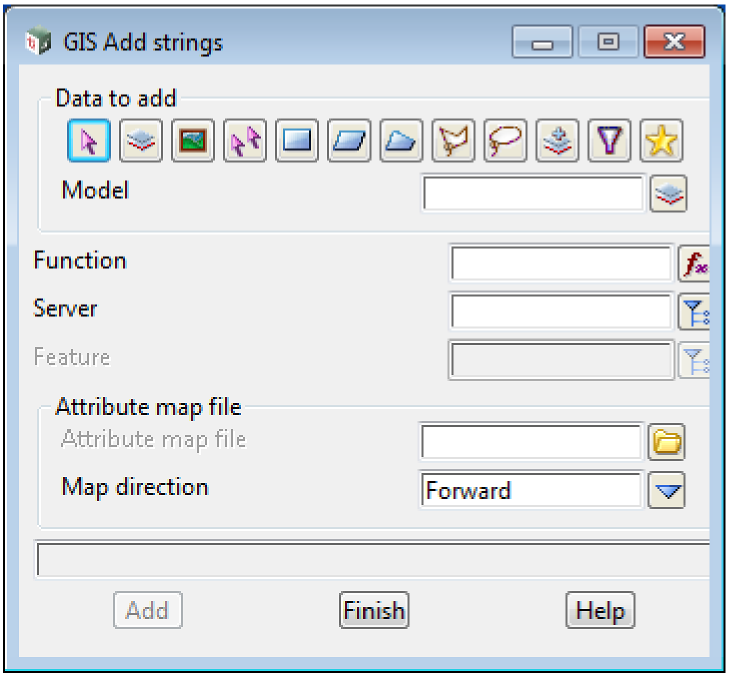
downloads the requested features from the server

8.6.14 GIS Add Strings

Position of option on menu: **File I/O =>GIS =>Add strings**

This allows a user to add strings to a known server. This is only allowed if the server permits the operation.

Selecting **Add strings** brings up the **GIS Add Strings** panel.



The fields and buttons used in this panel have the following functions.

Field Description   Type   Defaults   Pop-Up

Data to add

the data set to add to the server

Function

the existing FDO function to which strings will be added

Server

the server to connect to

Feature

the feature to add strings to

Attribute map file

Attribute map file

an attribute map file to apply to any strings to be uploaded

Map direction   choice box   Forward, Backward

the direction of mapping

Add   button

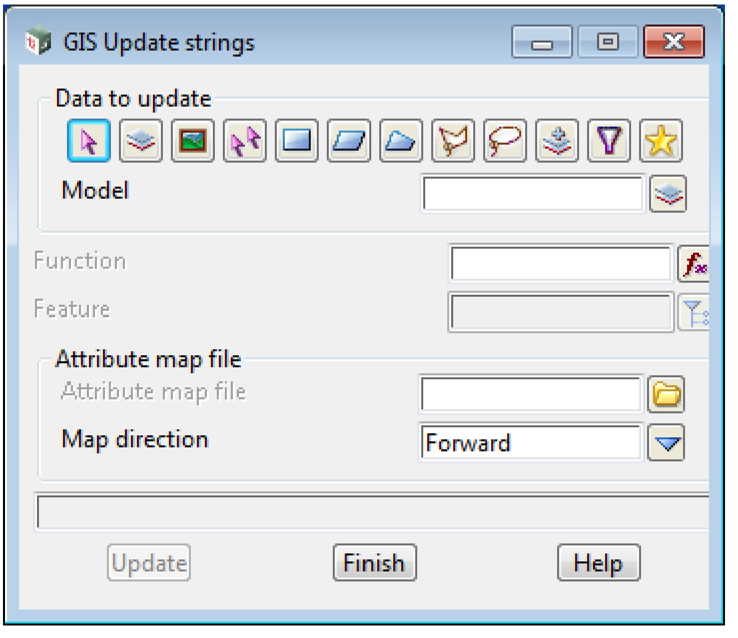
uploads the new strings to the server. Disabled until the function and server are set.

8.6.15 GIS Update Strings

Position of option on menu: **File I/O =>GIS =>Update strings**

This panel allows a user to upload / update strings on the remote server, if the server permits the operation.

Selecting **Update strings** brings up the **GIS Update Strings** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Data to update

the data set of strings to update on the server

Function

the existing FDO function containing strings to be updated

Feature

the feature to update

Attribute map file

Attribute map file	file		

the attribute map file to apply

Map direction	choice box		Forward, Backward

the attribute map file direction to apply to the attribute map file

Update	button		

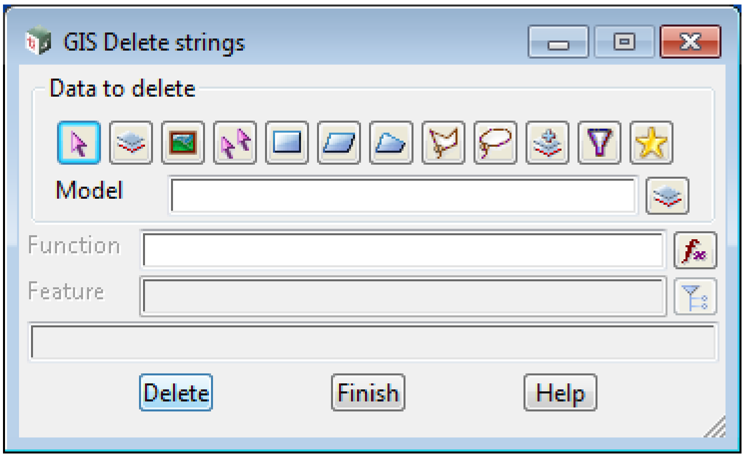
Connects to the server and sends the strings to be updated

8.6.16 GIS Delete Strings

Position of option on menu: **File I/O =>GIS =>Delete strings**

This panel allows a user to delete strings from the FDO server, if the server permits the operation.

Selecting **Delete strings** brings up the **GIS Delete Strings** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Data to delete

the data set to delete from the server

Function

the existing FDO function containing the strings to be deleted

Feature

the feature from which the strings will be deleted

Delete button

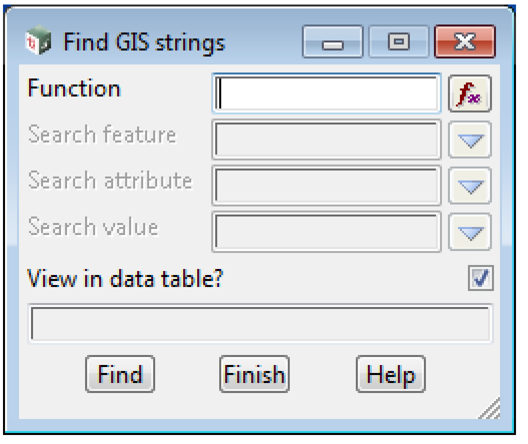
Connects to the server and deletes the specified strings

8.6.17 Find GIS Strings

Position of option on menu: **File I/O =>GIS =>Find**

This allows a user to find downloaded strings by a feature and an optional attribute / value pair.

Selecting **Find** brings up the **Find GIS Strings** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Function

the FDO function containing the strings. Once entered, the list of features will become available

Search feature

the feature to search through

Search attribute

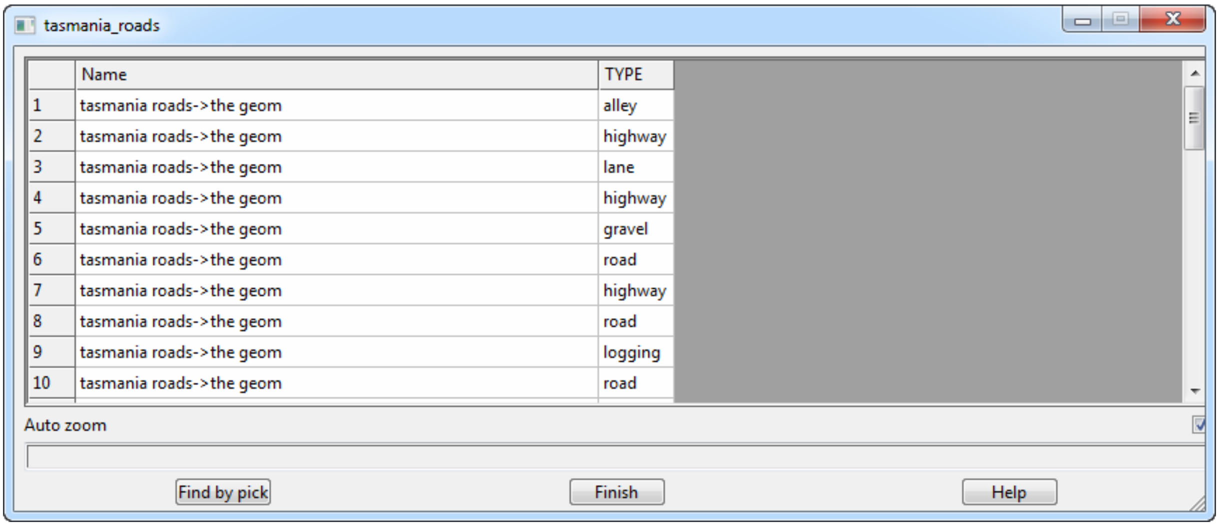
an optional attribute to search for. If not set, all strings in the feature will be found

Search value

the value of the attribute to search for

View in data table? tick box

whether or not to view in a separate data table or in the output window. The data table, below, will pan when the row is selected.



Auto zoom tick box

whether or not to zoom to the selected feature

Find by pick

finds a row based on a string selection

Find button

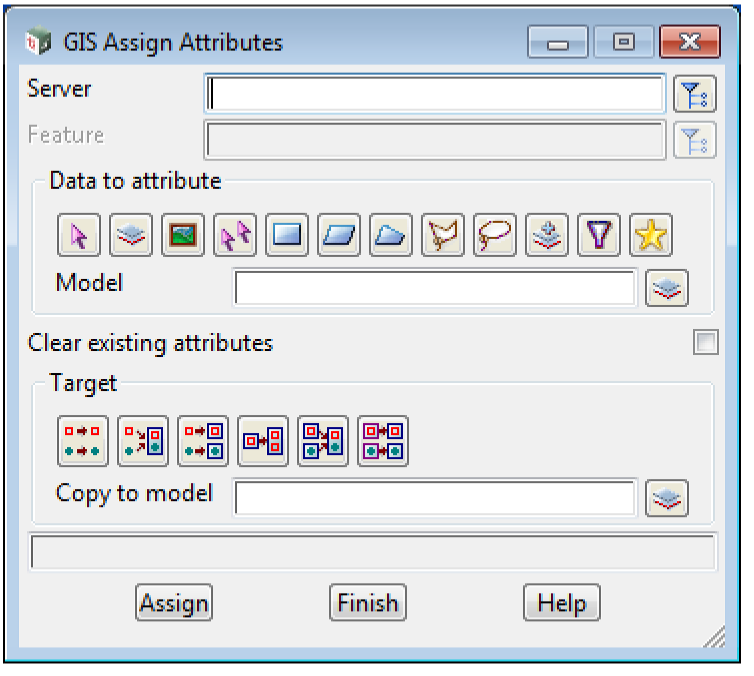
Finds the strings

8.6.18 GIS Assign Attributes

Position of option on menu: **File I/O =>GIS =>Assign Attributes**

This panel allows you to take the schema from a server and feature and create attributes with default values to a data set.

Selecting **Assign Attributes** brings up the **GIS Assign Attributes** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Server

the server to connect to

Feature

the feature to use the schema from

Data to attribute

the data set to apply feature attributes to

Clear existing attributes tick box

whether or not to clear existing attributes from the data set

Target

the target data set

Assign button

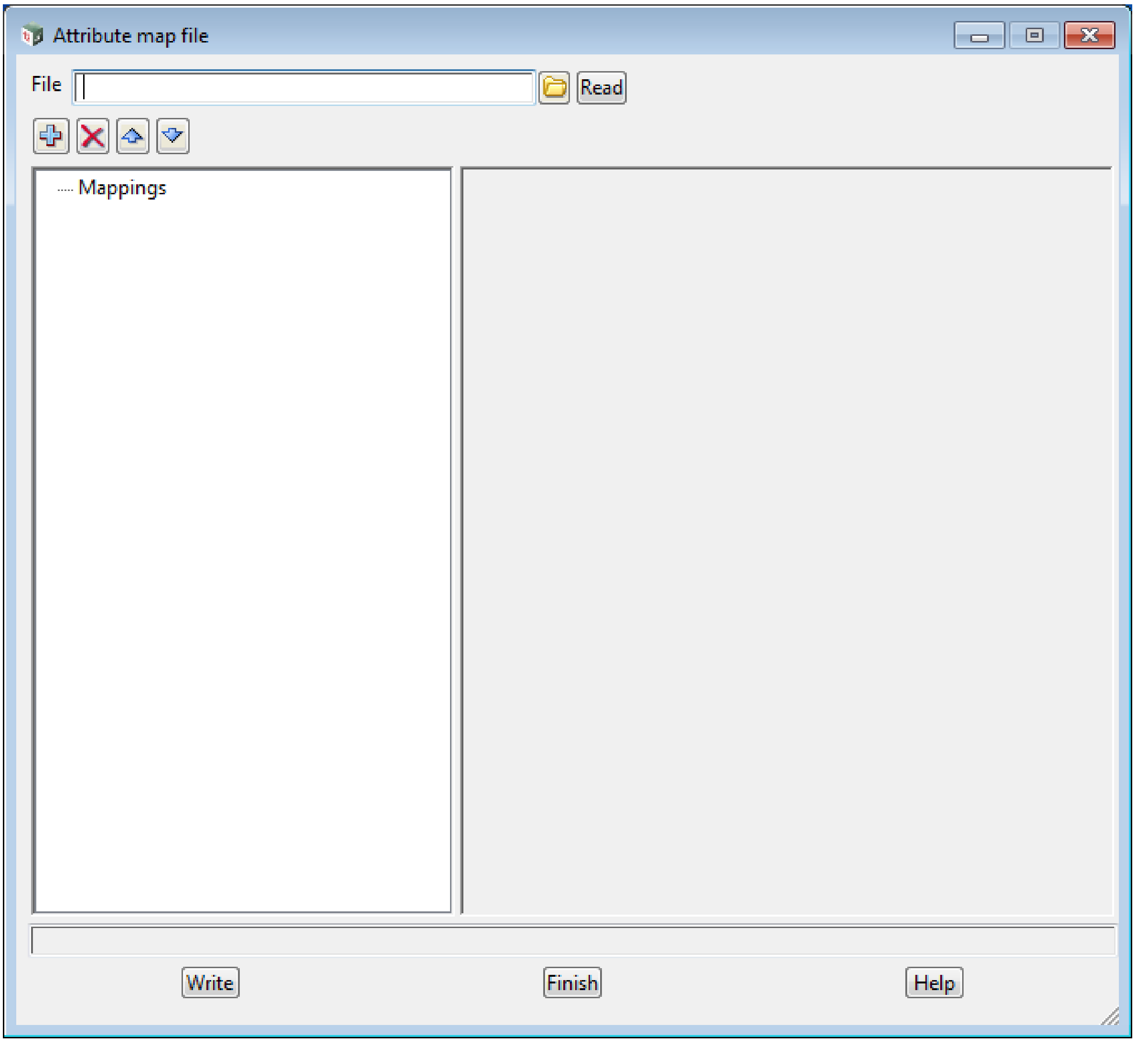
assigns the feature attributes to the data source / target set

8.6.19 Attribute Map File

Position of option on menu: **File I/O =>GIS =>Attribute map file editor**

This panel allows you to define a mapping from one attribute format to another. This allows you to map from attributes defined by an external data source to meet your attribute schema requirements, or vice versa.

Selecting **Attribute map file editor** brings up the **Attribute map file** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

File

the attribute mapping file to edit

Read button

reads the file

Insert

adds a new mapping

Delete

deletes the select mapping

Up

Moves the mapping up in the list

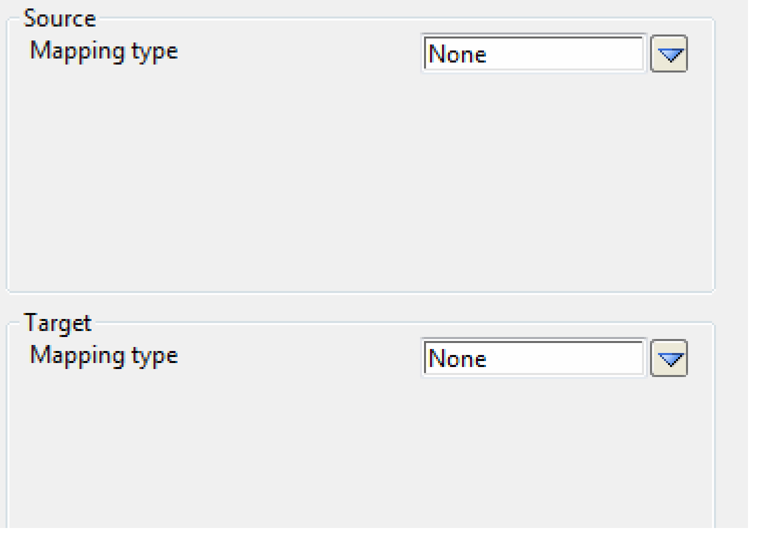
Down

Moves the mapping down in the list

Write button

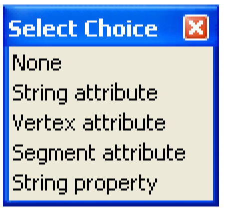
writes the mapping

After selecting **insert** the **Source Mapping Type** choice box and **Target Mapping Type** choice box appear on the screen.



Source Mapping Type choice box

the type of attribute to map from



For String attribute see String/Vertex/Segment attribute

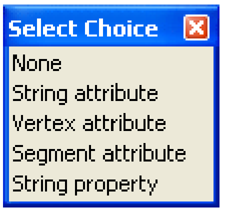
Vertex attribute see String/Vertex/Segment attribute

Segment attribute see String/Vertex/Segment attribute

String property see String Property attribute

Target Mapping Type choice box

the type of attribute to map to



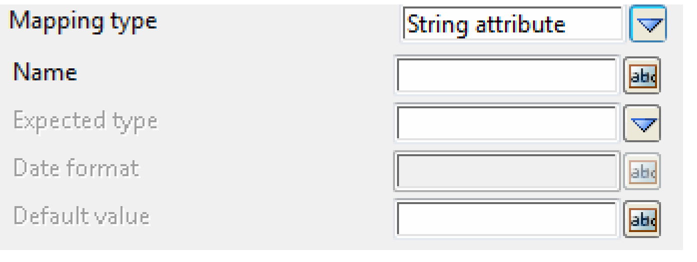
For String attribute see String/Vertex/Segment attribute

Vertex attribute see String/Vertex/Segment attribute

Segment attribute see String/Vertex/Segment attribute

String property see String Property attribute

String/Vertex/Segment attribute



Name

the name of the attribute

Expected type choice box

the type of attribute to be expected

Date format

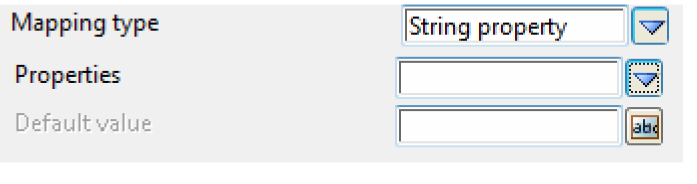
an optional date format (to convert to/from a date string)

Default value

an optional default value to use if the attribute is not found

String Property attribute

Maps to / from a property on a string, such as colour, name, height, weight *etc*.



Properties choice box

the property to map to / from

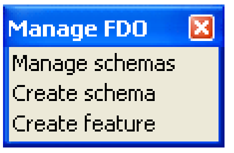
Default value

an optional default value to use if the value is not found

8.6.20 Manage FDO

Position of option on menu: **File I/O =>GIS =>Manage**

The **Manage FDO** walk-right menu containing these options is:



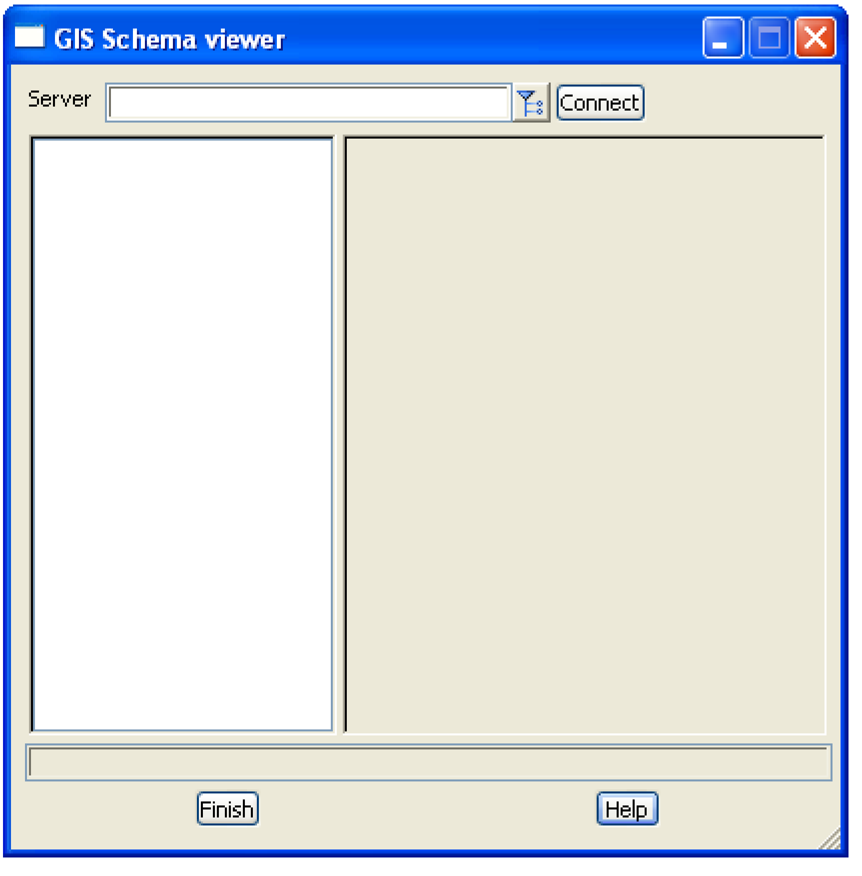
For the option Manage Schemas go to 8.6.20.1 GIS Schema Viewer
Create Schema 8.6.20.2 Create FDO Schema
Create Feature 8.6.20.3 Create GIS Feature

8.6.20.1 GIS Schema Viewer

Position of option on menu: **File I/O =>GIS =>Manage => Manage Schemas**

This panel allows you to view and edit schemas, if editing is permitted by the server.

Selecting **Manage Schemas** brings up the **GIS Schema Viewer** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

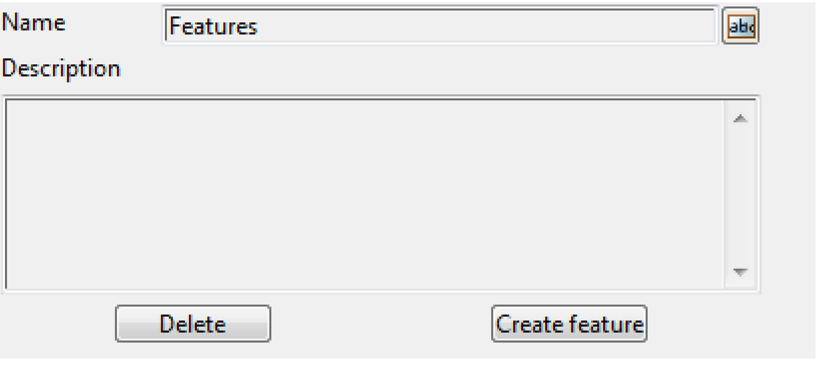
Server

the server to view schemas for

Connect button

connects to the server and downloads all schema information available

When a schema is selected, the right side of the panel will display:



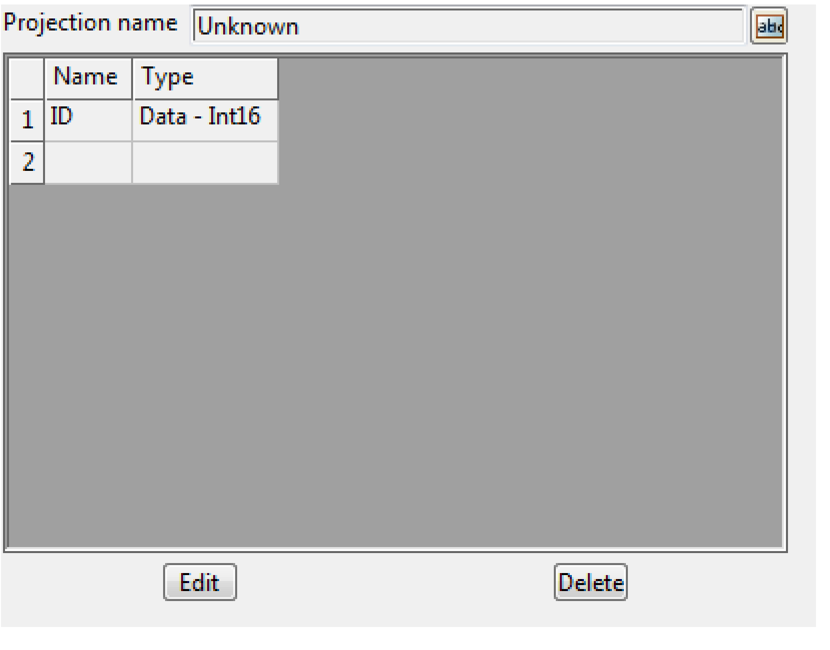
Delete button

deletes the selected schema - only available if permitted by the server

Create feature button

creates a new feature in this schema - only available if permitted by the server

When a feature is selected, the right side of the panel will display the list of attributes and their types.



Edit

edits the selected feature see 8.6.20.3 Create GIS Feature*- only available if permitted by the server*

Delete

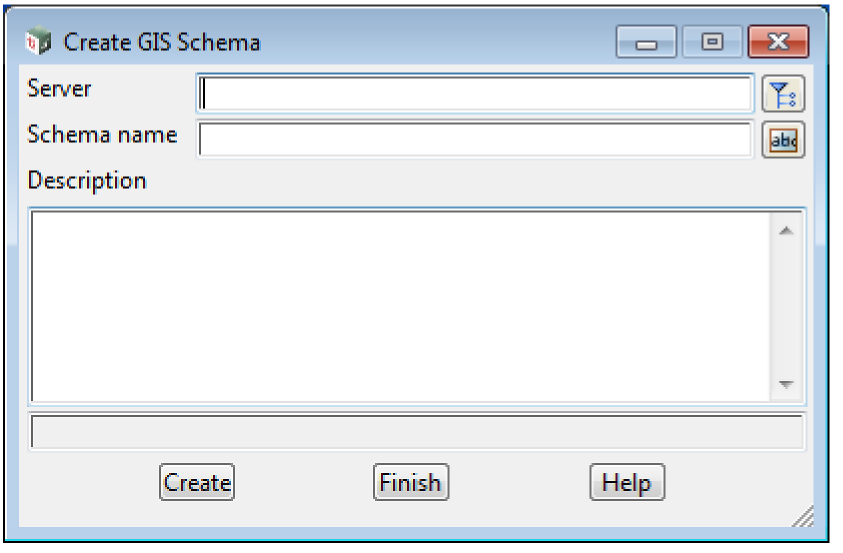
deletes the current feature - only available if permitted by the server

8.6.20.2 Create FDO Schema

Position of option on menu: **File I/O =>GIS =>Manage => Create Schema**

This panel creates a new schema on a remote server, if the operation is permitted by the server.

Selecting **Create Schema** brings up the **Create GIS Schema** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Server

the name of the server to create a schema on

Schema name

the name of the schema

Description

an optional description about the schema

Create button

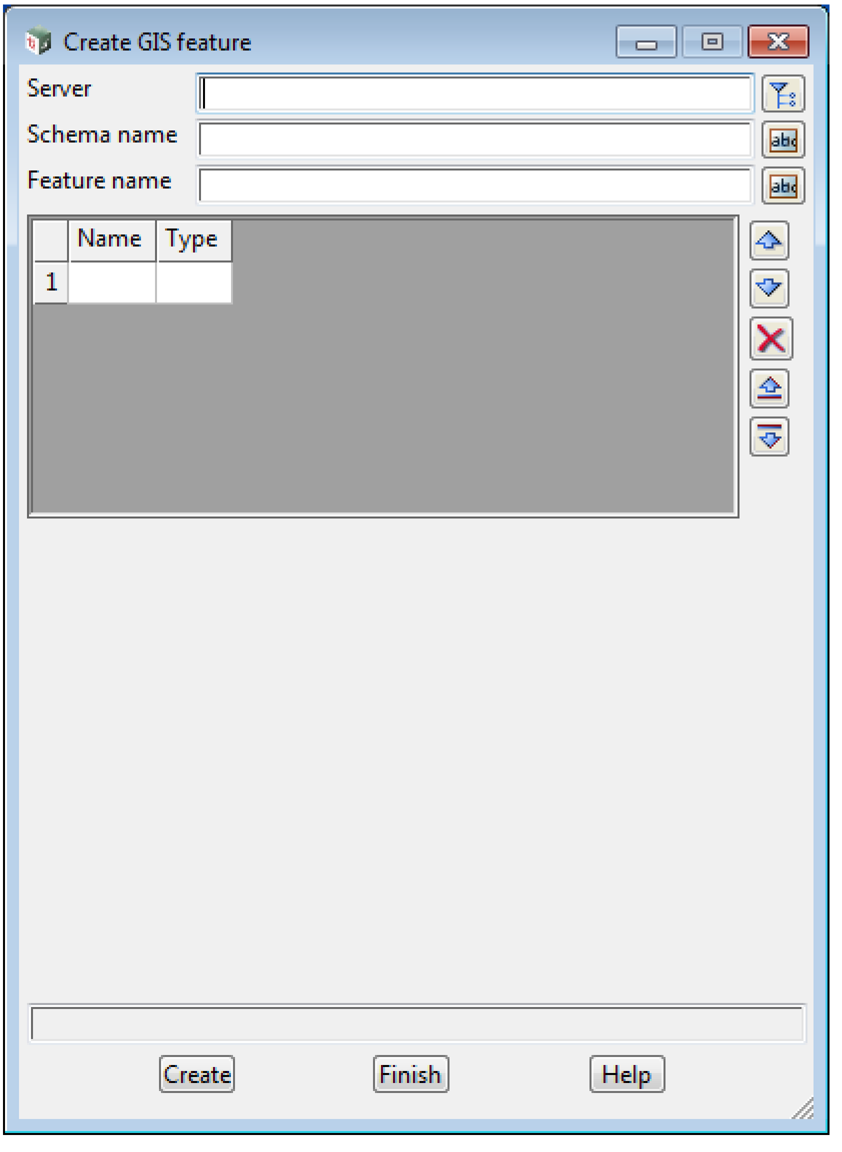
attempts to create the new schema on the remote server

8.6.20.3 Create GIS Feature

Position of option on menu: **File I/O =>GIS =>Manage => Create feature**

This panel creates a new feature in a schema, if the remote server permits the operation.

Selecting **Create feature** brings up the **Create GIS Feature** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Server

the name of the server to create the feature

Schema

the name of the schema to create a feature in

Feature name

the name of the feature to create

Grid Column Name

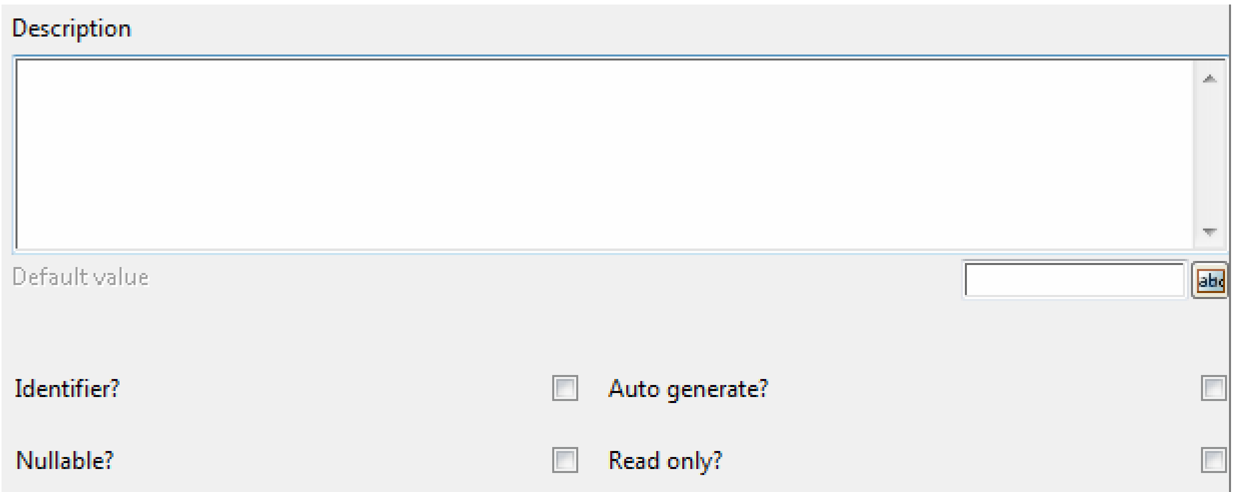
the name of the attribute to create

Grid Column Type

the type of attribute to make

Each row in the grid has a number of additional parameters to fill in.

The standard set of fields are:



Description

a description of the attribute

Default value

an optional default value

Identifier? tick box

whether or not this attribute is an identifier attribute

Nullable? tick box

whether or not this attribute is nullable (can have a null value)

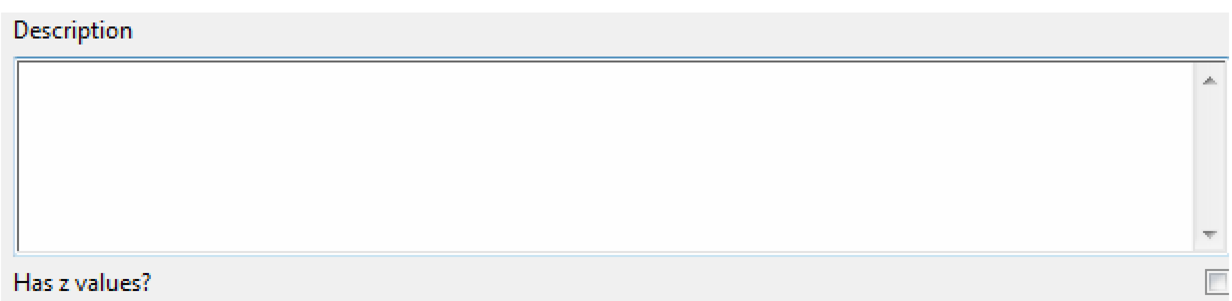
Auto generate? tick box

whether or not the server should auto generate the value of the attribute

Read only? tick box

whether or not the attribute is a read only attribute

Geometry attributes have the following fields:



Description

a description of the attribute

Has z values? tick box

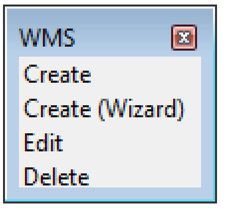
whether or not the geometry contains z values

8.6.21 WMS

Position of option on menu: **File I/O =>GIS =>WMS**

Position of option on menu: **Strings =>WMS**

The **WMS** walk-right menu containing these options is:



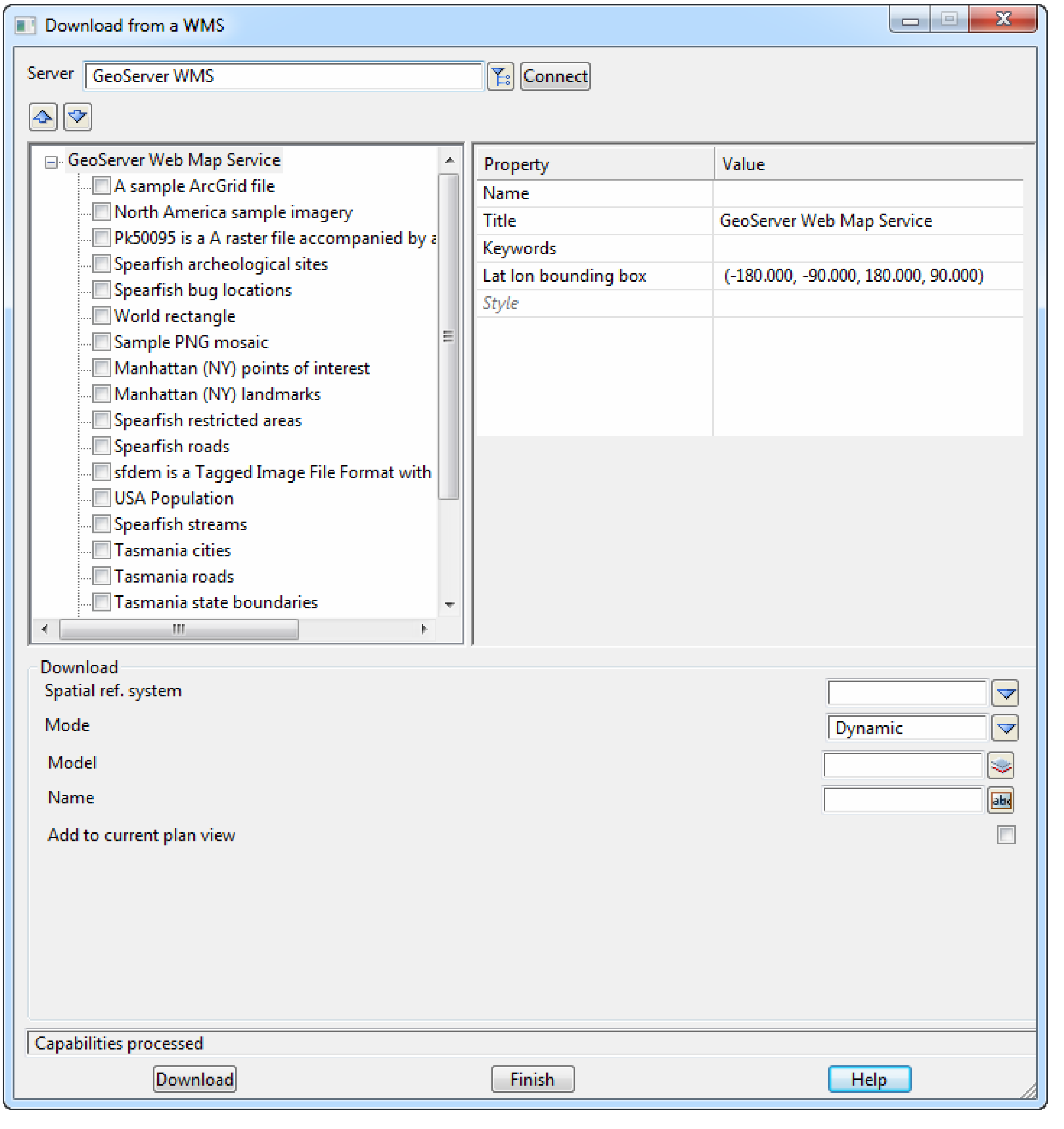
For the option Create go to	8.6.21.1 Download from a WMS
Create (Wizard)	8.6.21.2 WMS Wizard
Edit	8.6.22.1 Edit a WMS Image
Delete	8.6.22.2 Delete a WMS Image

8.6.21.1 Download from a WMS

Position of option on menu: **File I/O =>GIS =>WMS =>Create**

This panel allows you to set up an image to be downloaded from a **WMS (Web Mapping Server)**, either dynamically or as a once off download.

Selecting **Create** brings up the **Download from a WMS** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Server

the WMS server to download from

Connect button

connects to the server and downloads the schema

Up arrow

moves the layer up in order in which they will be downloaded.

Down arrow

moves the layer down in order in which they will be downloaded.

Spatial ref system

the spatial reference system to use when downloading

Mode choice box Dynamic, Save as file

Dynamic (update as you pan around), or Save as file

Model model box

the model to add a dynamic WMS image to

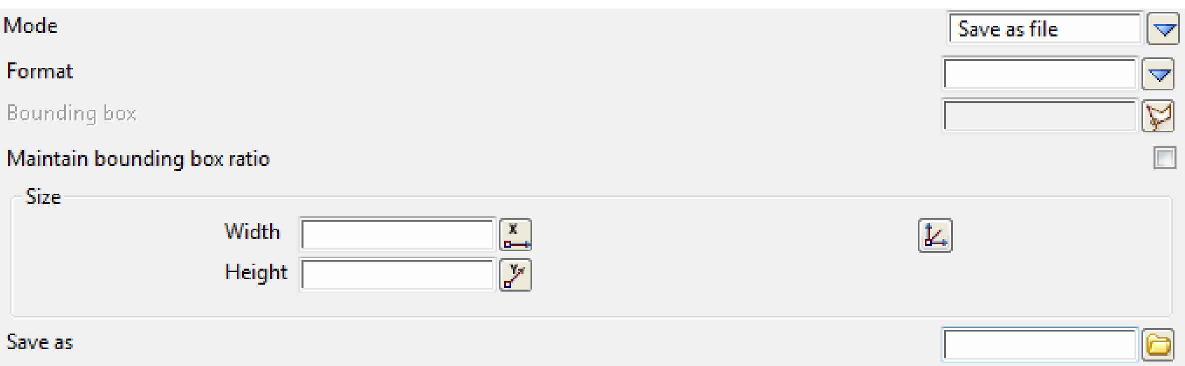
Name

the name of the image

Add to current plan view tick box

Whether or not to add to the current plan view

If the mode is **Save as file**, the following fields will be displayed.:



Format

the format of the image to download

Bounding box

whether or not to define a bounding box

Maintain bounding box ratio

whether or not to maintain the ratio of the bounding box on the server

Size

defines the width and height of the image file

Save as

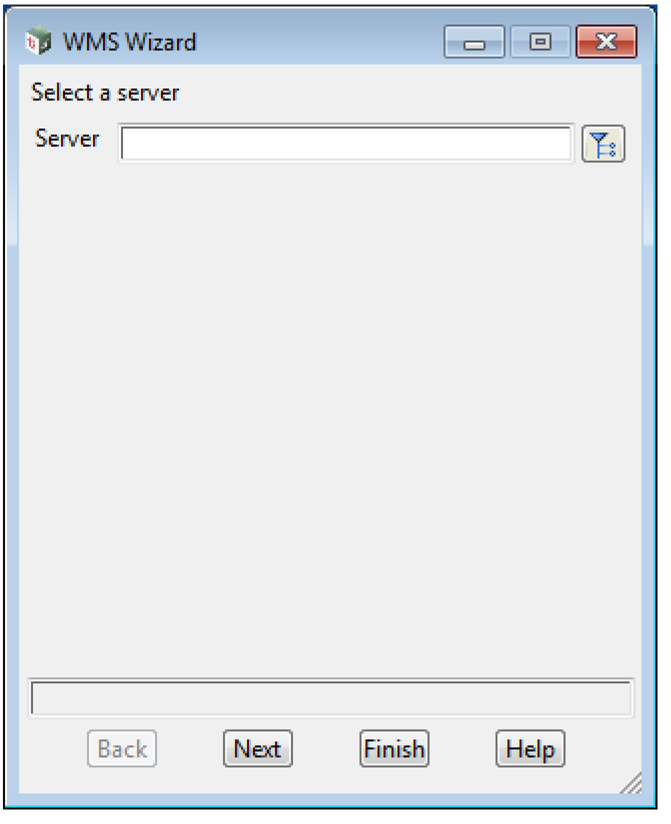
the target file for the image

8.6.21.2 WMS Wizard

Position of option on menu: **File I/O =>GIS =>WMS =>Create (Wizard)**

This panel allows you to set up an image to be downloaded from a **WMS (Web Mapping Server)**, either dynamically or as a once off download, via a wizard like interface.

Selecting **Create (Wizard)** brings up the **WMS Wizard** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Throughout the wizard, the following buttons will be available:

Server

the server to download from

Back

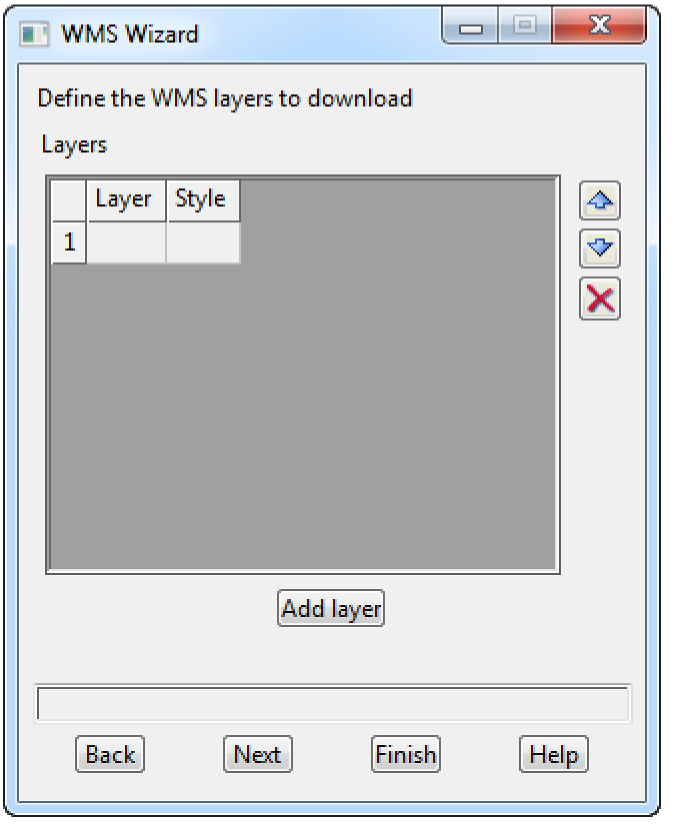
Takes you back to the last step

Next

Takes you to the next step

8.6.22 Layer Selection Step

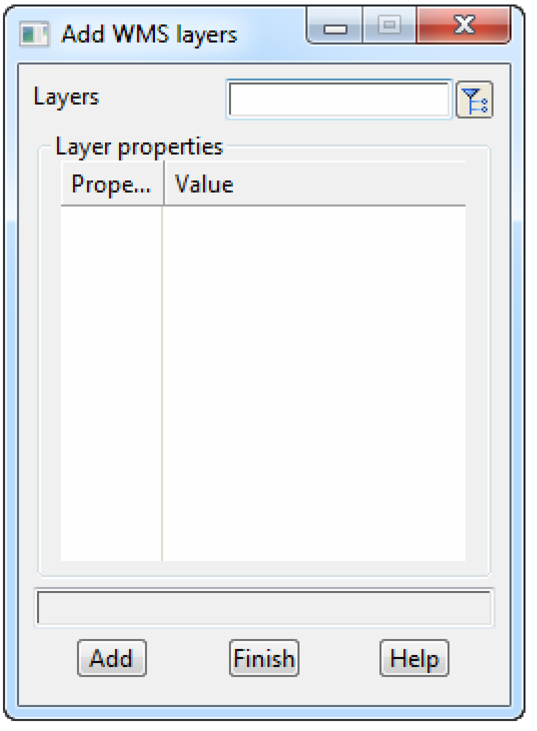
This panel allows you to select the layers to be downloaded



Add layers button

shows the list of layers to be downloaded
*Selecting **Add layers** bring up the **Add WMS layers** panel*

Add WMS layers

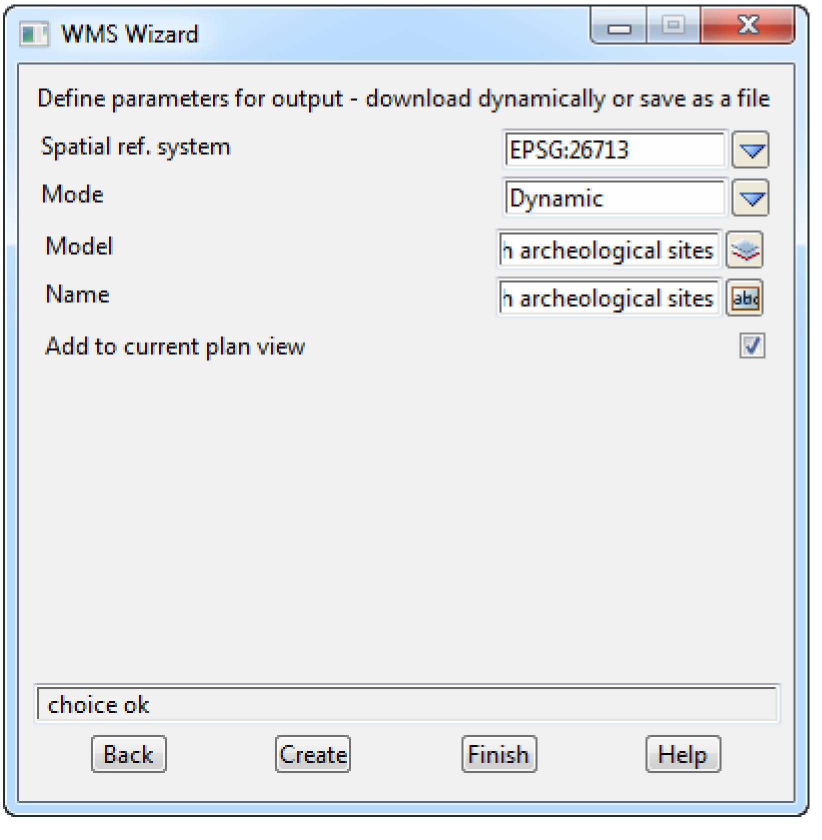


Layers

the layer to add or view properties

Add

adds the layer



Spatial ref. system

the spatial reference system to use when downloading

Mode

Dynamic (update as you pan around), or Save as file

Model model box

the model to add a dynamic WMS image to

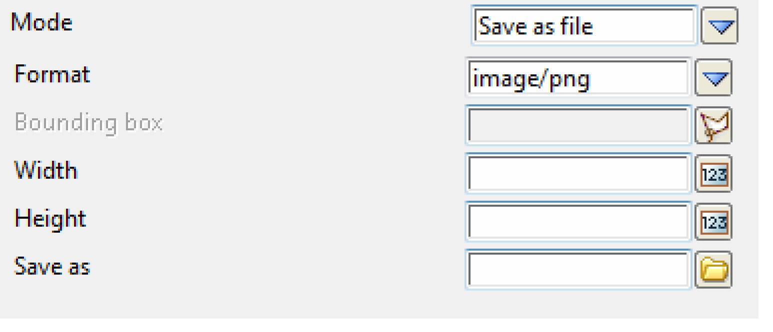
Name

the name of the image

Add to current plan view tick box

Whether or not to add to the current plan view

If the mode is set to '**Save as file**', the following fields will be displayed:



Format choice box

the format of the image to download

Bounding box

whether or not to define a bounding box

Width

the width of the image

Height

the height of the image

Save as

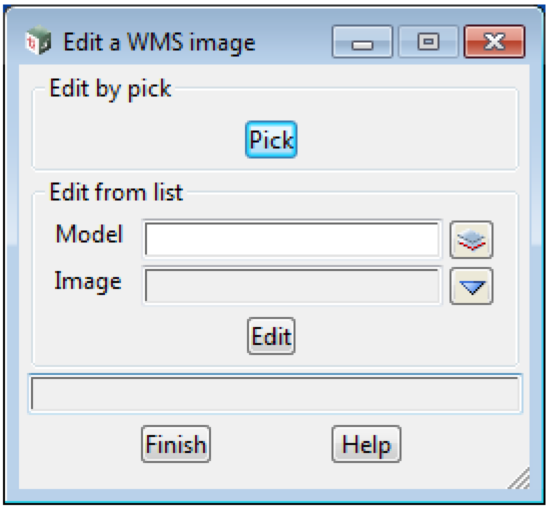
the target file for the image

8.6.22.1 Edit a WMS Image

Position of option on menu: **File I/O =>GIS =>WMS =>Edit**

Edits a **WMS** image, as selected by picking or by specifying by model and image name.

Selecting **Edit** brings up the **Edit a WMS Image** panel.



The fields and buttons used in this panel have the following functions.

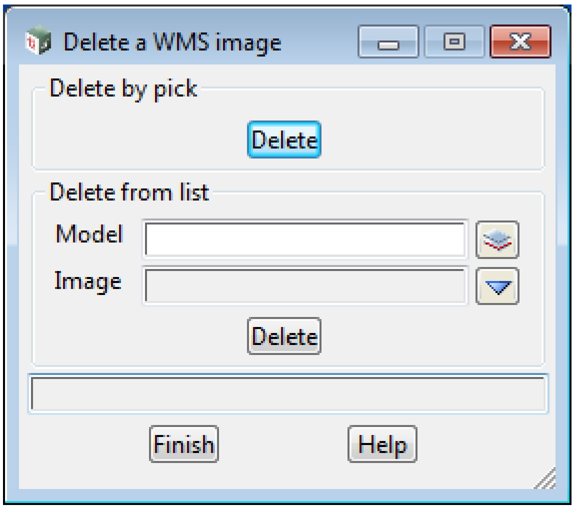
Field Description	Type	Defaults	Pop-Up
Edit by Pick			
Pick	button		
pick the WMS image to string, by picking its extent			
Edit from list			
Model	model box		
the model to look for WMS images in			
Image			
the WMS image within the model to edit			
Edit	button		
edits the selected WMS image			

8.6.22.2 Delete a WMS Image

Position of option on menu: **File I/O =>GIS =>WMS =>Delete**

Deletes a **WMS** image, as selected by picking or by specifying by model and image name.

Selecting **Delete** brings up the **Delete a WMS Image** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Delete by pick

Delete button

pick the WMS image to string, by picking its extent

Delete from list

Model model box

the model to look for WMS images in

Image

the WMS image within the model to delete

Delete

delete the selected WMS image

8.7 Label Map Files

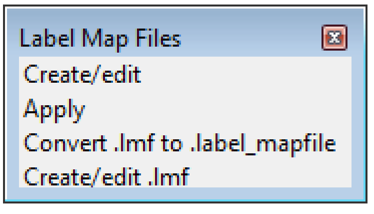
Position of menu: **File I/O =>Label Map files**

The option Create/edit creates and edits ***Label Map*** files which are then used in the Label Map File ***Apply*** option.

From ***12d Model 10***, the Label Map file had an XML format (the file ends in .label_mapfile). Before ***12d Model 10*** there was a format with files ending in .lmt so there is an option to convert from the older format, .lmf, to the newer .label_mapfile format.

Note: if the ***Create/edit*** option is used on an old format file, it is automatically converted to the new format.

The **Map files** walk-right menu is



create/edit a label mapfile
apply a label map file to data
convert pre-V10 label map file format to XML format label map files
create/edit a pre-V10 label map file

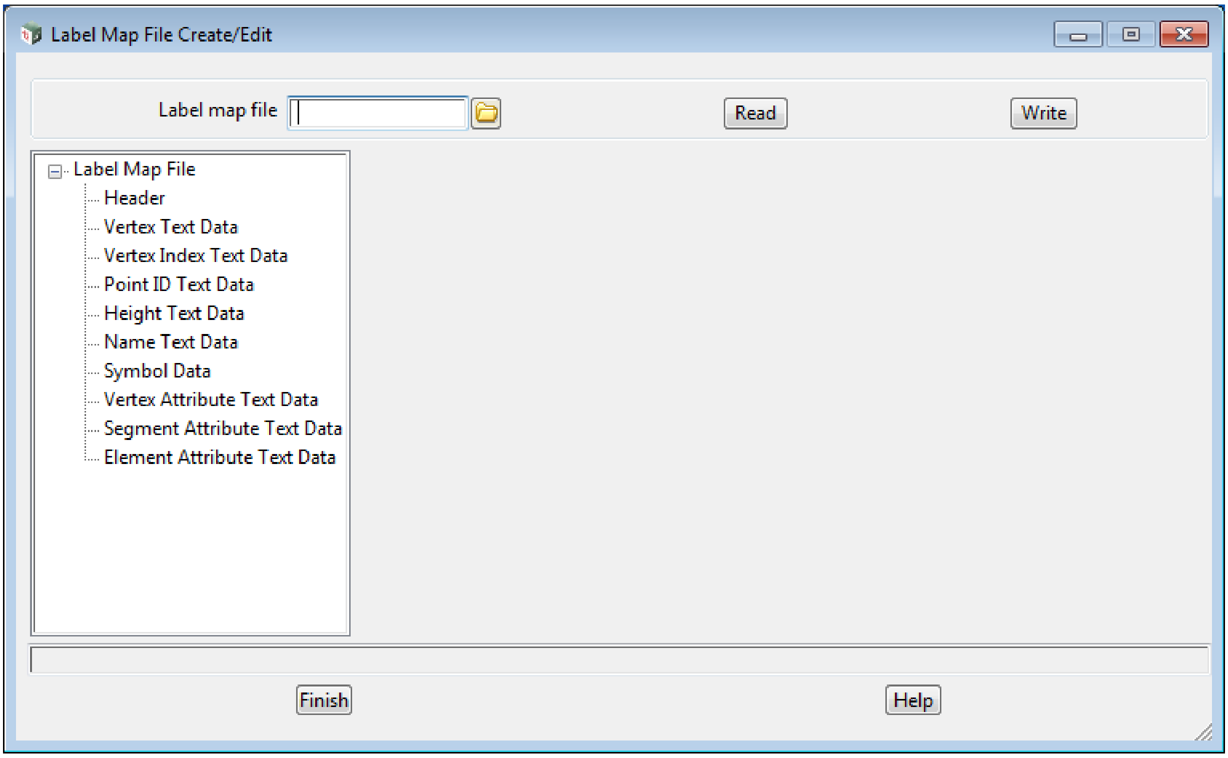
For *Create/edit*, go to 8.7.1 Create/Edit Label Map File
Apply 8.7.2 Apply a Label Map File
Convert .lmf to .label_mapfile 8.7.3 Convert .lmf Files to .label_mapfile
Create/edit .lmf 8.7.4 Create/Edit Pre-V10 Label Map File

8.7.1 Create/Edit Label Map File

Position of option on menu: **File I/O =>Label Map files =>Create/edit**

The **label map file** option **creates** or **edits** the label map files which is used in the ***Apply Label Map File*** option to create various labels at string vertices and segments.

Selecting **Create/edit** brings up the **Label Map File Create/Edit** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Label map file	input		*.label_mapfile and *.lmf

name of the label map file to be created or edited.

If it is a .lmf file, it is automatically converted to the new xml format and written out as a .label_mapfile file.

Bottom buttons

Read button

read in the label map file given in the **Label map file** *field and load the data into the fields of the table. If the file doesn't exist, an error message is given.*

Write button

write out the data in the table to the file name given in the **Label map file** *field. If the file already exists, a Replace-Cancel panel checks to see if the existing file is to be over written - if no or cancel is selected, nothing is written out.*

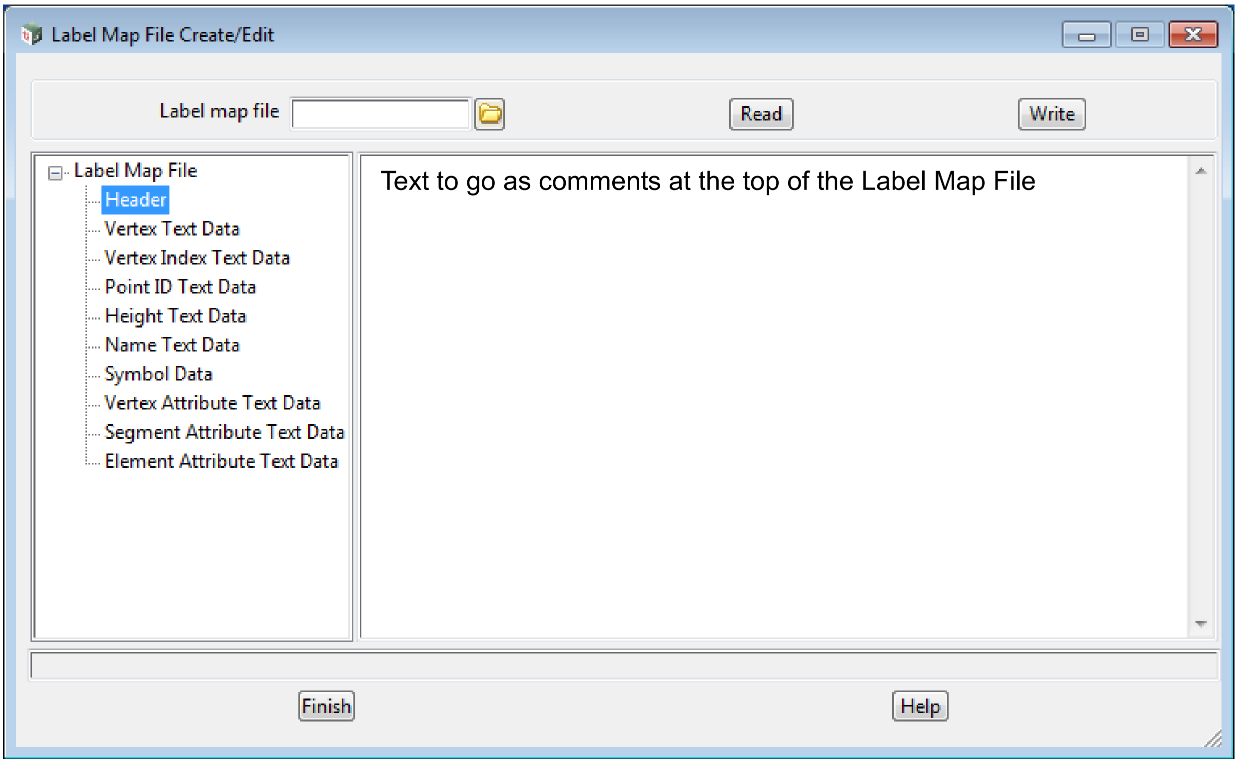
For information on Header 8.7.1.1 Header
Vertex Text Data 8.7.1.2 Vertex Text Data
Vertex Index Text Data 8.7.1.3 Vertex Index Text Data
Point ID Text Data 8.7.1.4 Point Id Text Data
Height Text Data 8.7.1.5 Height Text Data
Name Text Data 8.7.1.6 Name Text Data

Symbols	8.7.1.7 Symbol Data
Vertex Attribute Text Data	8.7.1.8 Vertex Attribute Text Data
Segment Attribute Text Data	8.7.1.9 Segment Attribute Text Data
Element Attribute Text Data	8.7.1.10 Element Attribute Text Data

8.7.1.1 Header

Header just has lines of comment text. A common use is to document which client the file is to be used for, or the revision history.

The text written in the right hand side of the panel is written out as comments at the top of the Label Map file.



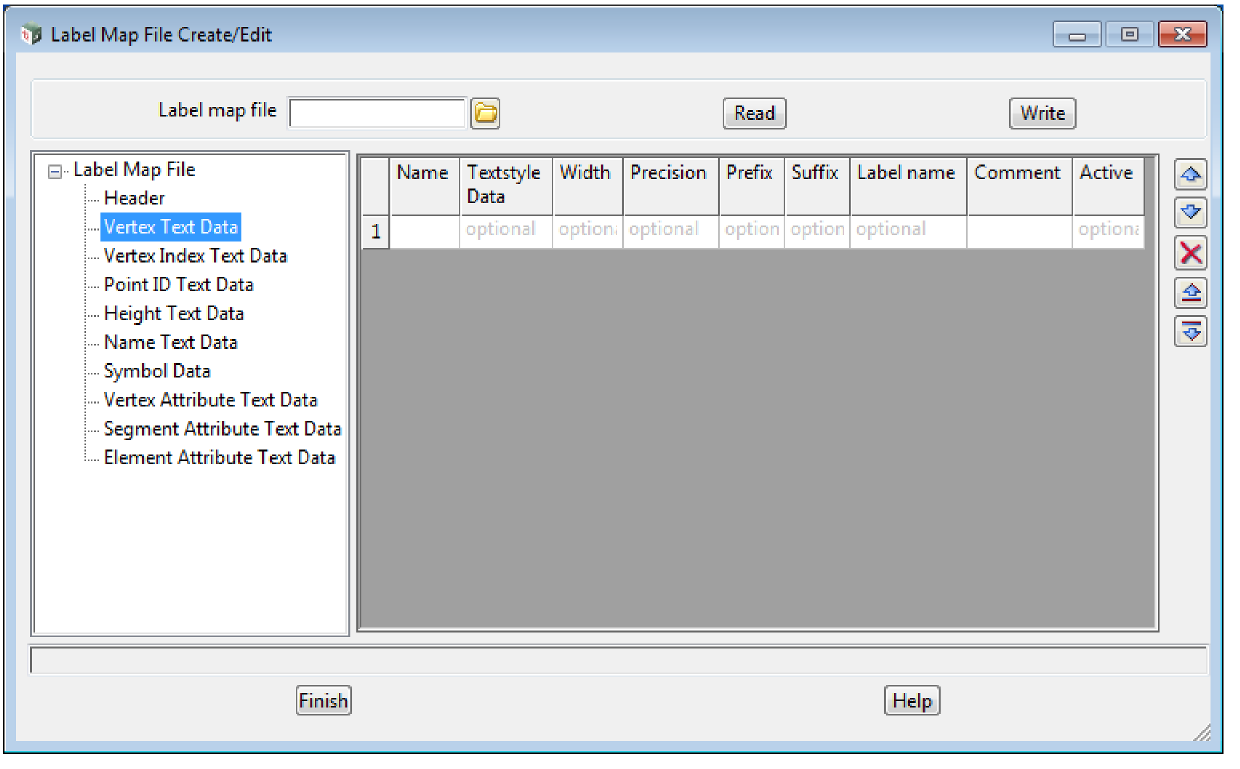
Go to the next section 8.7.1.2 Vertex Text Data or return to 8.7.1 Create/Edit Label Map File.

8.7.1.2 Vertex Text Data

Super strings can have text at ***each*** vertex of the super string. **Vertex Text Data** creates labels from the vertex text at each vertex.

The ***Vertex Text Data*** grid specifies ***which strings are selected*** and how a label containing the ***vertex text*** for each vertex of the selected strings is created.

For each selected string, a new super string is created with vertices at the same position as the vertices of the selected string, and with the labels created as vertex text at the appropriate vertices of the new super string.



Name **select name** menu for fields

for the selected strings, labels are created from the vertex text at each vertex. See Selecting Strings using Name:

For each selected string, a new super string is created with vertices at the same position as the vertices of the selected string, and with the labels created as vertex text at the appropriate vertices of the new super string.

Selecting Strings using Name:

The text in the **Name** *column is the name of the strings to be selected.* **Name** *can include wild cards* ***** *and wild characters* ***?*** *to allow matching of multiple strings.*

For each string, the string name is compared to the text in the **Name** *column for this line and if there is a match then the string is selected to label using the parameters in this line of the Label Map file.*

If a match occurs for a line, then this line of the Label Map file grid is used and no other lines are tested against.

If no match occurs, then this line of the Label Map file grid is ignored and a test for a match is made against the next line of the grid.

Hence the order of the text in the **Names** *column in the grid is important, especially when wild cards and wild characters are used.*

Textstyle data **select textstyle data** menu for fields

textstyle data for labels. This controls the units, size, colour, border etc for the created text

Width

if non blank, the minimum number of number of characters that is allowed for the vertex text before applying the prefix and suffix.

If the number of characters in the vertex text is less than **Width** *characters, then additional spaces are added to the left of the vertex text so that there are* **Width** *characters.*

If the number of characters in the vertex text is greater than **Width***, then* **Width** *is ignored. Hence the vertex text is not truncated.*

For example if **Width** *is set to 20 and the vertex text has only ten characters, then an extra 10 spaces are added to the left of the vertex text before applying the Prefix/Suffix.*

if blank, the vertex text is used with no space padding.

Precision

not applicable

Prefix/Suffix

prefix/suffix for the text

Label name

the created super strings are given this name

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

Go to the next section 8.7.1.3 Vertex Index Text Data or return to 8.7.1 Create/Edit Label Map File.

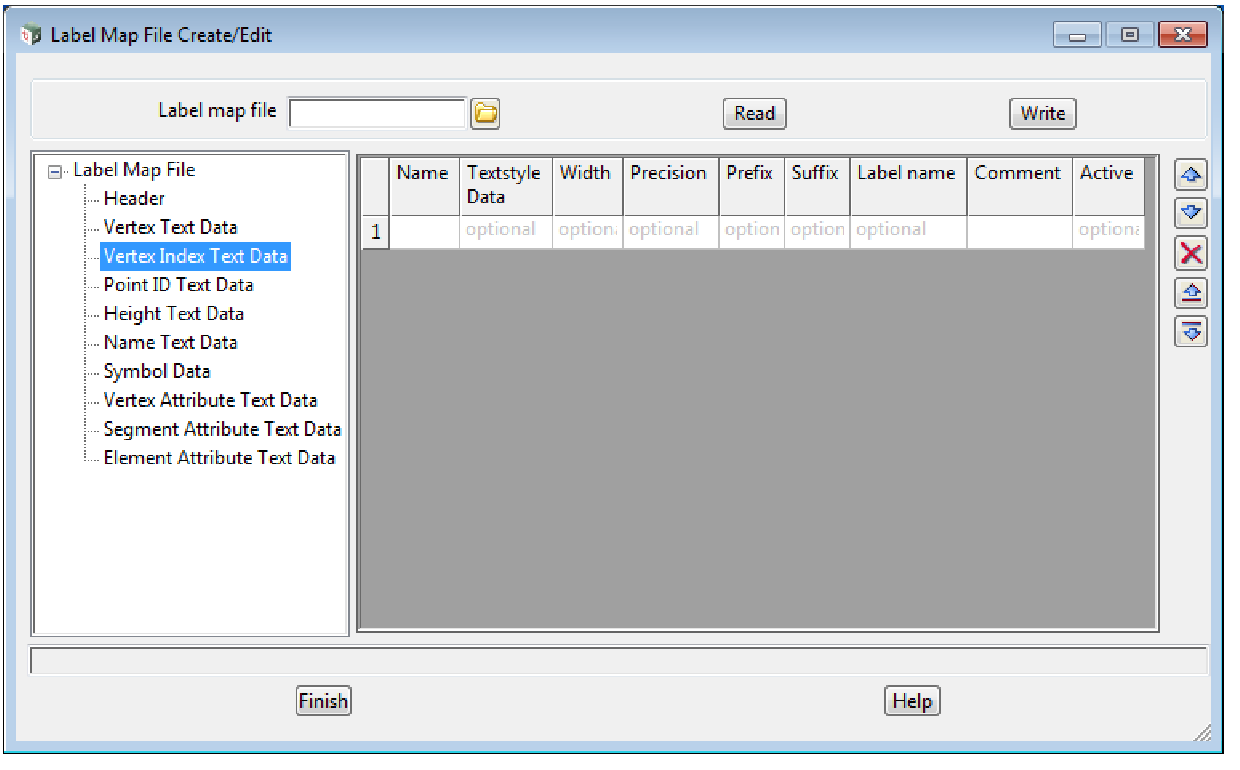
8.7.1.3 Vertex Index Text Data

Vertex Index Text Data labels the vertex index at each vertex.

The ***Vertex Index Text Data*** grid specifies ***which strings are selected*** and how a label containing the ***vertex index*** for each vertex of the selected strings is created.

For each selected string, a new super string is created with vertices at the same position as the vertices of the selected string, and with the labels created as vertex text at the appropriate vertices of the new super string.

Note: the vertex index is the position of the vertex in the string. For example if the vertex is the tenth vertex in the string, then the vertex index is 10.



Name **select name** menu for fields

for the selected strings, labels are created from the vertex index of each vertex. See Selecting Strings using Name:

For each selected string, a new super string is created with vertices at the same position as the vertices of the selected string, and with the labels created as vertex text at the appropriate vertices of the new super string.

Textstyle data **select textstyle data** menu for fields

textstyle data for labels. This controls the units, size, colour, border etc for the created text

Width

if non blank, the minimum number of characters allowed for the vertex index (including **Precision***) before applying the prefix and suffix.*

If the number of digits in the vertex index (including **Precision***) is less than* **Width** *characters, then additional spaces are added to the left of the text so that there are* **Width** *characters.*

If the number of digits in the vertex index (including **Precision***) is greater than* **Width***, then* **Width** *is ignored. Hence the vertex index value is not truncated.*

As an example, if **Width** *is set to 10 and the vertex index is 23 and* **Precision** *is 3, then the label is "   023". That is, an extra seven spaces is added to the left of the vertex index value.*

if blank, the vertex text is used with no modification.

Precision

if the number of digits in the vertex index is less than the absolute value of **Precision***, then left 0 padding is used to bring the number of digits in the text up to the absolute value of* **Precision***.*

As an example, if **Width** *is set to 10 and the vertex index is 23 and* **Precision** *is 3, then the label is " 023". That is, an extra seven spaces is added to the left of the vertex index value.*

Prefix/Suffix

prefix/suffix for the text

Label name

the created super strings are given this name

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

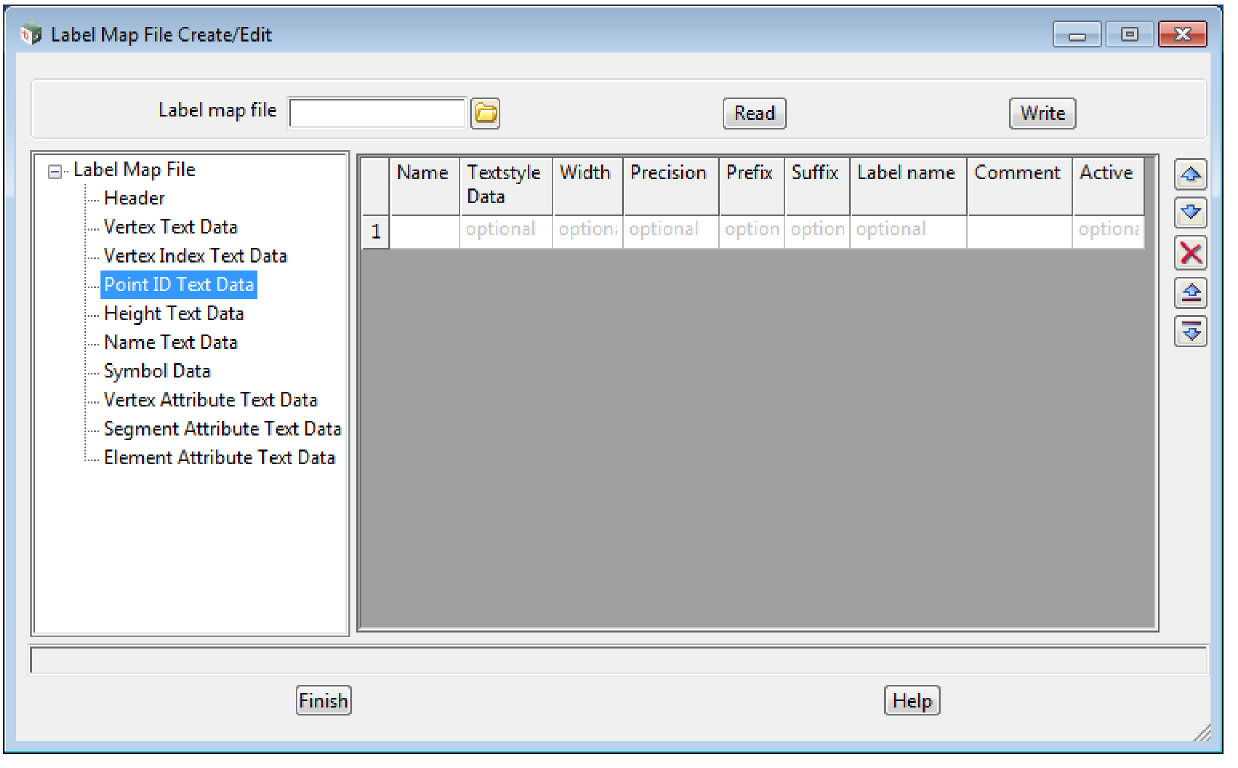
Go to the next section 8.7.1.4 Point Id Text Data or return to 8.7.1 Create/Edit Label Map File.

8.7.1.4 Point Id Text Data

Super strings can have a ***Point ID*** at ***each*** vertex of the super string. **Point ID Text Data** labels the Point ID at each vertex.

The ***Point ID Text Data*** grid specifies ***which strings are selected*** and how a label containing the ***Point ID***'s for each vertex of the selected strings is created.

For each selected string, a new super string is created with vertices at the same position as the vertices of the selected string, and with the labels created as vertex text at the appropriate vertices of the new super string.



Name **select name** menu for fields

for the selected strings, labels are created from the Point ID of each vertex. See Selecting Strings using Name:

For each selected string, a new super string is created with vertices at the same position as the vertices of the selected string, and with the labels created as vertex text at the appropriate vertices of the new super string.

Textstyle data **select textstyle data** menu for fields

textstyle data for labels. This controls the units, size, colour, border etc for the created text

Width

if non blank, the minimum number of characters that is allowed for the Point ID text before applying the prefix and suffix.

If the number of characters in the Point ID text is less than **Width** *characters, then additional spaces are added to the left of the Point ID so that there are* **Width** *characters.*

If the number of characters in the Point ID text is greater than **Width**, *then* **Width** *is ignored. Hence the Point ID text is not truncated.*

For example if **Width** *is set to 20 and the Point ID text has only ten characters, then an extra 10 spaces are added to the left of the Point ID text before applying the Prefix/Suffix.*

if blank, the Point ID text is used with no space padding.

Precision

not applicable

Prefix/Suffix

prefix/suffix for the text

Label name

the created super strings are given this name

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

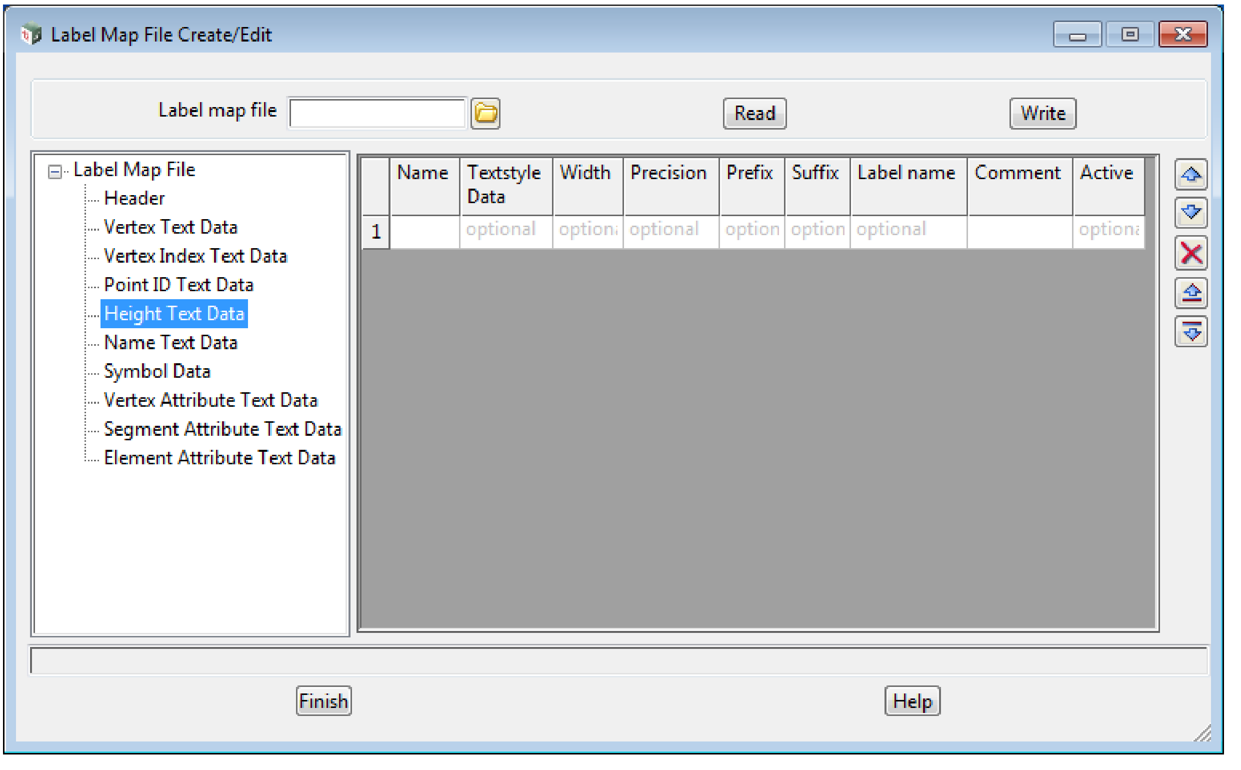
Go to the next section 8.7.1.5 Height Text Data or return to 8.7.1 Create/Edit Label Map File.

8.7.1.5 Height Text Data

Height Text Data labels the height at each vertex.

The ***Height Text Data*** grid specifies ***which strings are selected*** and how a label containing the ***height (z-value)*** for each vertex of the selected strings is created.

For each selected string, a new super string is created with vertices at the same position as the vertices of the selected string, and with the labels created as vertex text at the appropriate vertices of the new super string.



Name — **select name** menu for fields

for the selected strings, labels are created from the height (z-value) of each vertex. See Selecting Strings using Name:

For each selected string, a new super string is created with vertices at the same position as the vertices of the selected string, and with the labels created as vertex text at the appropriate vertices of the new super string.

Textstyle data — **select textstyle data** menu for fields

textstyle data for labels. This controls the units, size, colour, border etc for the created text

Width

if non blank, the minimum number of characters allowed for the z-value (including **Precision***) before applying the prefix and suffix.*

If the number of digits in the z-value (including **Precision***) is less than* **Width** *characters, then additional spaces are added to the left of the text so that there are* **Width** *characters.*

If the number of digits in the z-value (including **Precision***) is greater than* **Width***, then* **Width** *is ignored. Hence the z-value is not truncated (other than by* **Precision***).*

As an example, if **Width** *is set to 10 and the z-value is 23.2 and* **Precision** *is 3, then the label is " 23.200". That is, an extra four spaces is added to the left of the z-value.*

if blank, the z-value with **Precision** *is used with no further modification.*

Precision

if positive, the number of decimal places after the decimal point.

if negative, the absolute number is the number of decimal places to first calculate the text of the z-value for but then any trailing zeros after the decimal place are removed.

Important Note: *this use of positive and negative for not eliminating/eliminating trailing zeros after the decimal place is the opposite to plot parameters.*

Prefix/Suffix

prefix/suffix for the text

Label name

the created super strings are given this name

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

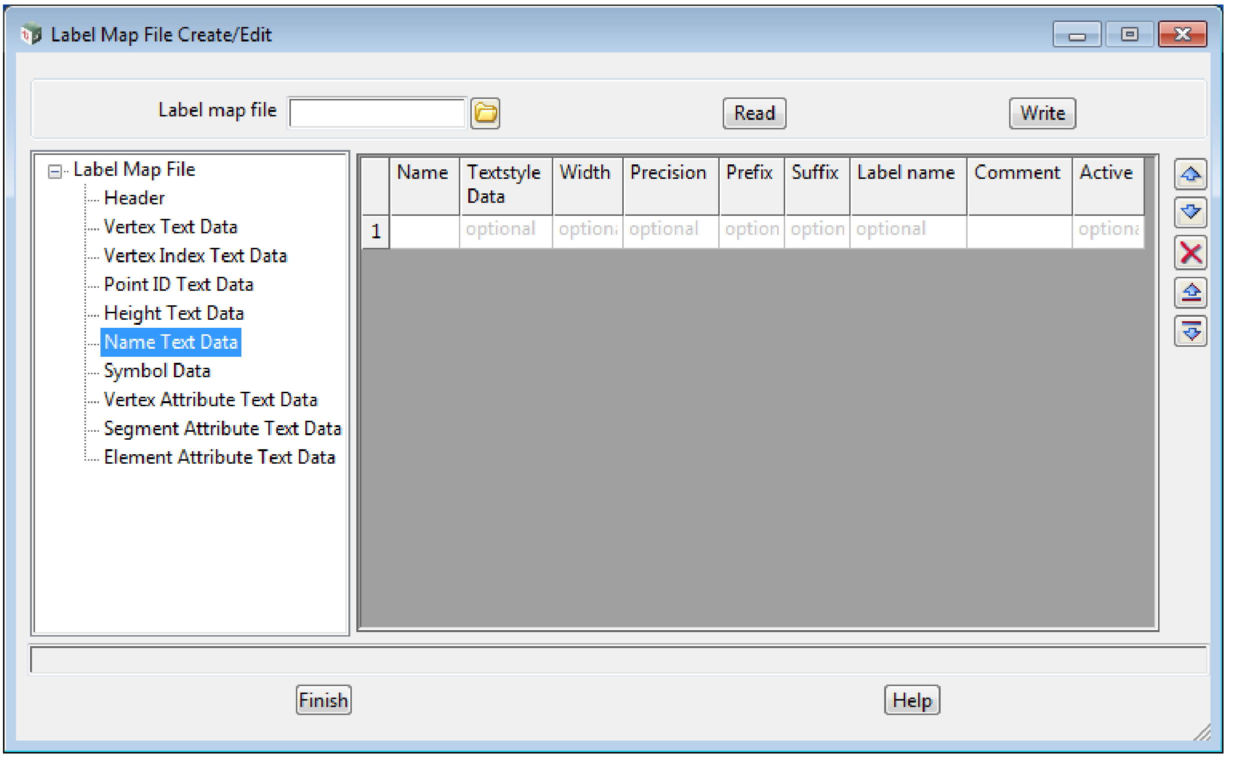
Go to the next section 8.7.1.6 Name Text Data or return to 8.7.1 Create/Edit Label Map File.

8.7.1.6 Name Text Data

Name Text Data labels the string name at each vertex.

The ***Name Text Data*** grid specifies ***which strings are selected*** and how a label containing the ***string name*** (often referred to as string ***Code*** by Surveyors) for each vertex of the selected strings is created.

For each selected string, a new super string is created with vertices at the same position as the vertices of the selected string, and with the labels created as vertex text at the appropriate vertices of the new super string.



Name **select name** menu for fields

for the selected strings, labels are created from the string name. See Selecting Strings using Name:
For each selected string, a new super string is created with vertices at the same position as the vertices of the selected string, and with the labels created as vertex text at the appropriate vertices of the new super string.

Textstyle data **select textstyle data** menu for fields

textstyle data for labels. This controls the units, size, colour, border etc for the created text

Width

if non blank, the minimum number of characters that is allowed for the string name text before applying the prefix and suffix.

If the number of characters in the string name is less than **Width** *characters, then additional spaces are added to the left of the string name text so that there are* **Width** *characters.*

If the number of characters in the string name is greater than **Width**, *then* **Width** *is ignored. Hence the string name is not truncated.*

For example if **Width** *is set to 20 and the string name has only ten characters, then an extra 10 spaces are added to the left of the string name before applying the Prefix/Suffix.*

if blank, the string name text is used with no space padding.

Precision

not applicable

Prefix/Suffix

prefix/suffix for the text

Label name

the created super strings are given this name

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

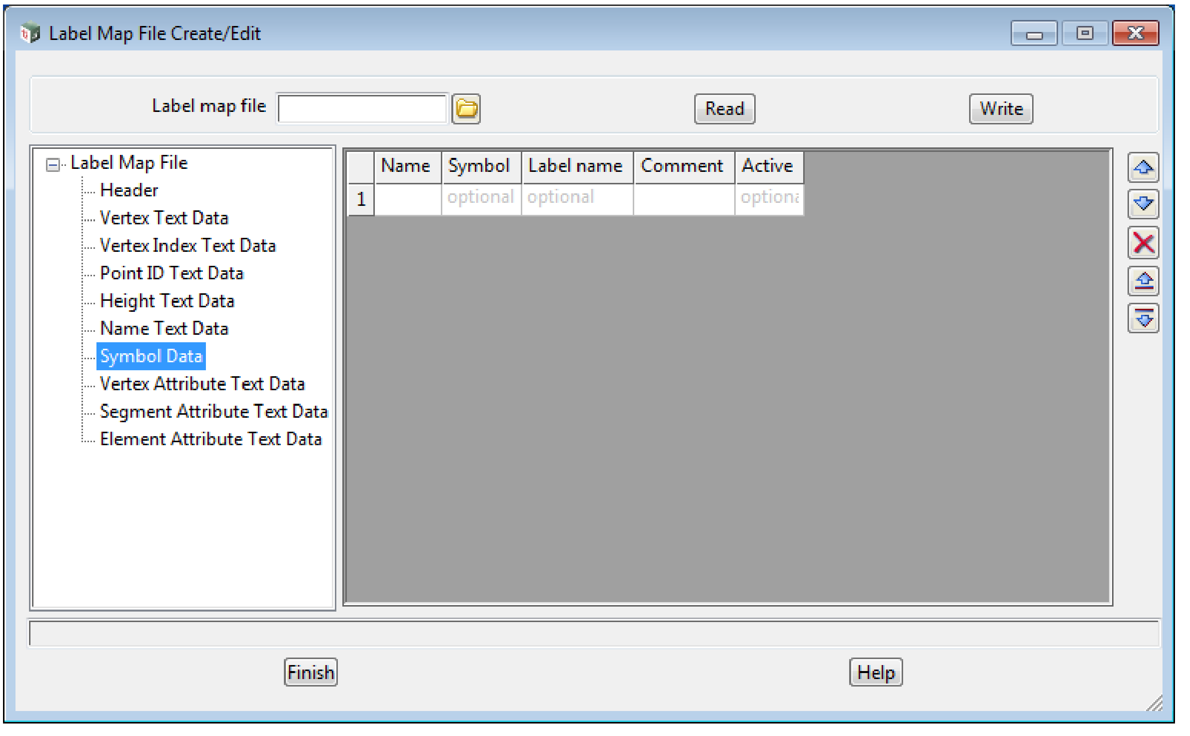
Go to the next section 8.7.1.7 Symbol Data or return to 8.7.1 Create/Edit Label Map File.

8.7.1.7 Symbol Data

Symbol Data creates symbols at each vertex.

The ***Symbol Data*** grid specifies ***which strings are selected*** and what symbol is created at the position of each vertex of the selected strings.

For each selected string, a new super string is created with vertices at the same position as the vertices of the selected string, and with symbols created as vertex symbols at each vertex of the new super string.



Name **select name** menu for fields

for the selected strings, symbols are created at each vertex. See Selecting Strings using Name:
For each selected string, a new super string is created with vertices at the same position as the vertices of the selected string, and with symbols created as vertex symbols at each vertex of the new super string.

Symbol **select symbol** menu for fields

symbol for the vertices. This controls the size, colour, rotation etc for the created symbol.

Label name

the created super strings are given this name

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

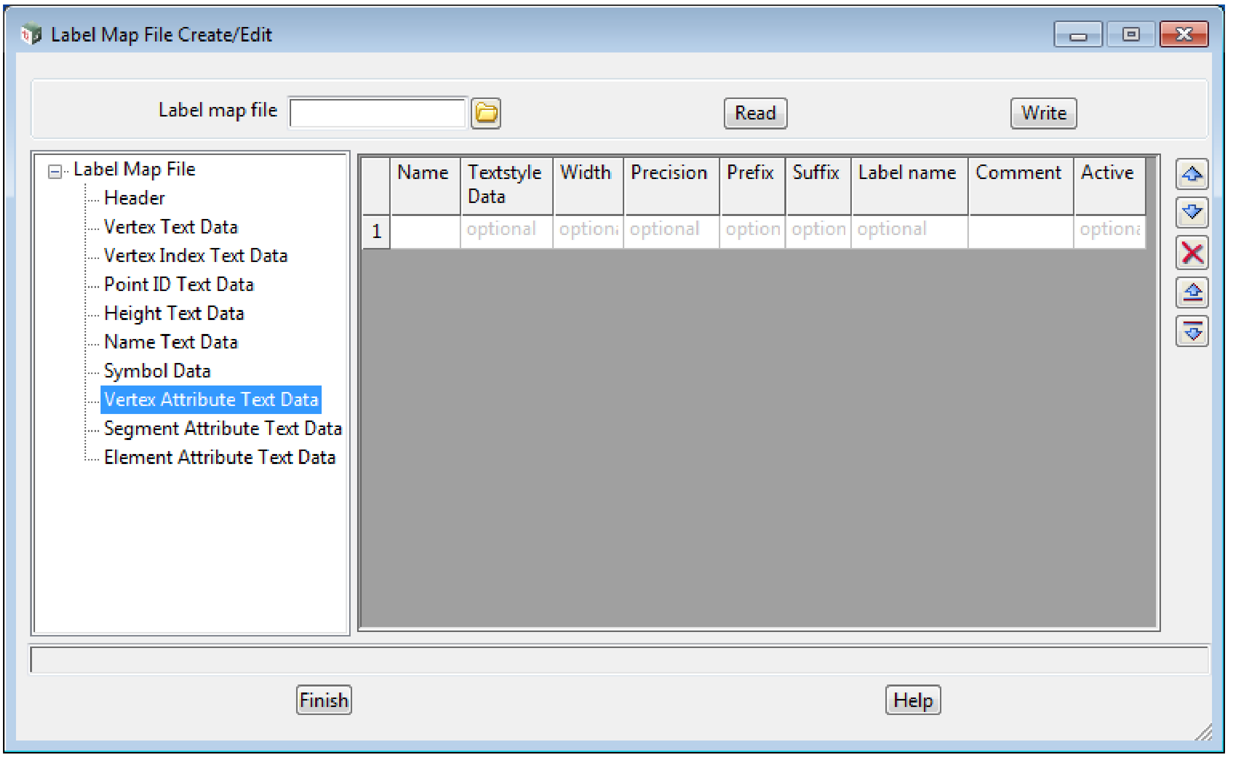
Go to the next section 8.7.1.8 Vertex Attribute Text Data or return to 8.7.1 Create/Edit Label Map File.

8.7.1.8 Vertex Attribute Text Data

Super strings can have attributes at ***each vertex*** of the super string. **Vertex Attribute Text Data** labels the ***value*** of vertex attributes at each vertex.

The ***Vertex Attribute Text Data*** grid specifies ***which strings** and **which vertex attributes are selected*** and how a label containing the ***value of the vertex attribute*** is created for each vertex of the selected strings.

For each selected string, a new super string is created with vertices at the same position as the selected vertices of the selected string, and with the labels created as vertex text at the appropriate vertices of the new super string.



Name **select name** menu for fields

*for the selected strings and the selected **vertex attributes** of those strings, a label is created from the **value** of the vertex attribute using the parameters in this line of the Label Map file. See* Selecting Strings and Attributes Using Name:.

For each selected string, a new super string is created with vertices at the same position as the selected vertices of the selected string, and with the labels created as vertex text at the appropriate vertices of the new super string.

Selecting Strings and Attributes Using Name:

The text in the **Name** *column is made up of the name of the string to be selected, followed by / and then the path name of the attribute (see* Attribute Pathname*) to be selected.* **Name** *can include wild cards* *** *and wild characters* **?** *to allow matching on multiple strings and/or multiple attributes.*

For each string, the string name and attribute path is compared to the text in the **Name** *column for this line and if there is a match then the label from the selected attribute is created using the parameters in this line of the Label Map file.*

For example

Drain/Adac/Pit/*   would search any string with the name starting with "Drain" and gets matches for any third level attribute in the second level attribute group "Adac/Pit".*

***Note** that it is the **name** of the attribute used for **selection** but it is the **values** of the selected attribute that are used in the **labels**.*

If a match occurs for a line, then this line of the Label Map file grid is used and no other lines are tested against.

If no match occurs, then this line of the Label Map file grid is ignored and a test for a match is made against the next line of the grid.

Hence the order of the text in the **Name** *column in the grid is important, especially when wild cards and wild characters are used.*

Textstyle data **select textstyle data** menu for fields

textstyle data for labels. This controls the units, size, colour, border etc for the created text

Width

*If the attribute is type **text**:*

if non blank, the minimum number of characters that is allowed for the attribute text before applying the prefix and suffix.

If the number of characters in the attribute text is less than **Width** *characters, then additional spaces are added to the left of the attribute text so that there are* **Width** *characters.*

If the number of characters in the attribute text is greater than **Width***, then* **Width** *is ignored. Hence the attribute text is not truncated.*

For example if **Width** *is set to 20 and the attribute text has only ten characters, then an extra 10 spaces are added to the left of the attribute text before applying the Prefix/Suffix.*

if blank, the attribute text is used with no space padding.

*If the attribute is type **real**:*

if non blank, the minimum number of characters allowed for the attribute real value (including **Precision***) before applying the prefix and suffix.*

If the number of digits in the attribute real value (including **Precision***) is less than* **Width** *characters, then additional spaces are added to the left of the text so that there are* **Width** *characters.*

If the number of digits in the attribute real value (including **Precision***) is greater than* **Width***, then* **Width** *is ignored. Hence the attribute real value is not truncated (other than by* **Precision***).*

As an example, if **Width** *is set to 10 and the attribute real value is 23.2 and* **Precision** *is 3, then the label is "   23.200". That is, an extra four spaces is added to the left of the vertex index value.*

if blank, the attribute real value with **Precision** *is used with no further modification.*

*If the attribute is type **integer**:*

if non blank, the minimum number of characters allowed for the attribute integer value (including **Precision***) before applying the prefix and suffix.*

If the number of digits in the attribute integer value (including **Precision***) is less than* **Width** *characters, then additional spaces are added to the left of the text so that there are* **Width** *characters.*

If the number of digits in the attribute integer value (including **Precision***) is greater than* **Width***, then* **Width** *is ignored. Hence the attribute integer value is not truncated.*

As an example, if **Width** *is set to 10 and the attribute integer value is 23 and* **Precision** *is 3, then the label is "   023". That is, an extra seven spaces is added to the left of the attribute integer value.*

if blank, the attribute integer value text is used with no modification.

Precision

*If the attribute is type **text** then not applicable.*

*If the attribute is type **real**:*

if positive, the number of decimal places after the decimal point.

if negative, the absolute number is the number of decimal places to first calculate the text of the attribute real value for but then any trailing zeros after the decimal place are removed.

***Important Note:** this use of positive and negative for not eliminating/eliminating trailing zeros* *after the decimal place is the opposite to plot parameters.*

*If the attribute is type **integer**:*

if the number of digits in the attribute integer value is less than the absolute value of **Precision**, *then left 0 padding is used to bring the number of digits in the text up to the absolute value of* **Precision**.

As an example, if **Width** *is set to 10 and the attribute integer value is 23 and* **Precision** *is 3, then the label is "   023". That is, an extra seven spaces is added to the left of the attribute integer value.*

Prefix/Suffix

prefix/suffix for the text

Label name

the created super strings are given this name

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

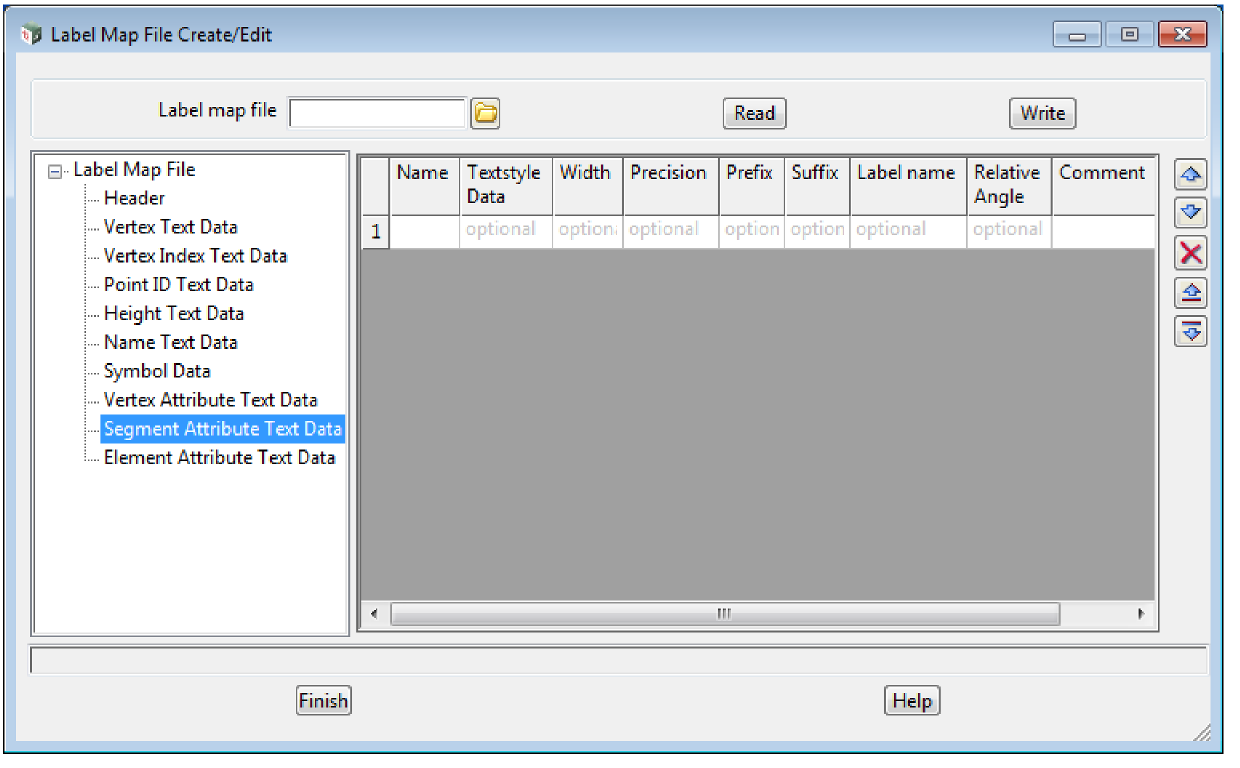
Go to the next section 8.7.1.9 Segment Attribute Text Data or return to 8.7.1 Create/Edit Label Map File.

8.7.1.9 Segment Attribute Text Data

Super strings can have attributes at ***each* segment** of the super string. **Segment Attribute Text Data** labels the segment attribute at each segment.

The ***Segment Attribute Text Data*** grid specifies ***which strings*** *and* ***which segment attributes are selected*** and how a label containing the ***value of the segment attribute*** is created for each segment of the selected strings.

For each selected string and selected segment attribute, text strings are created with vertices at the mid point of the selected segments of the selected string, and with the labels created as the text.



Name **select name** menu for fields

for the selected strings and the selected ***segment attributes*** *of those strings, a label is created from the* ***value*** *of the segment attribute using the parameters in this line of the Label Map file. See* Selecting Strings and Attributes Using Name:.

For each selected string and selected segment attribute, text strings are created with vertices at the mid point of the selected segments of the selected string, and with the labels created as the text.

Textstyle data **select textstyle data** menu for fields

textstyle data for labels. This controls the units, size, colour, border etc for the created text

Width

If the attribute is type ***text****:*

if non blank, the minimum number of characters that is allowed for the attribute text before applying the prefix and suffix.

If the number of characters in the attribute text is less than **Width** *characters, then additional spaces are added to the left of the attribute text so that there are* **Width** *characters.*

If the number of characters in the attribute text is greater than **Width***, then* **Width** *is ignored. Hence the attribute text is not truncated.*

For example if **Width** *is set to 20 and the attribute text has only ten characters, then an extra 10* *spaces are added to the left of the attribute text before applying the Prefix/Suffix.*

if blank, the attribute text is used with no space padding.

*If the attribute is type **real**:*

if non blank, the minimum number of characters allowed for the attribute real value (including **Precision***) before applying the prefix and suffix.*

If the number of digits in the attribute real value (including **Precision***) is less than* **Width** *characters, then additional spaces are added to the left of the text so that there are* **Width** *characters.*

If the number of digits in the attribute real value (including **Precision***) is greater than* **Width***, then* **Width** *is ignored. Hence the attribute real value is not truncated (other than by* **Precision***).*

As an example, if **Width** *is set to 10 and the attribute real value is 23.2 and* **Precision** *is 3, then the label is "   23.200". That is, an extra four spaces is added to the left of the vertex index value.*

if blank, the attribute real value with **Precision** *is used with no further modification.*

*If the attribute is type **integer**:*

if non blank, the minimum number of characters allowed for the attribute integer value (including **Precision***) before applying the prefix and suffix.*

If the number of digits in the attribute integer value (including **Precision***) is less than* **Width** *characters, then additional spaces are added to the left of the text so that there are* **Width** *characters.*

If the number of digits in the attribute integer value (including **Precision***) is greater than* **Width***, then* **Width** *is ignored. Hence the attribute integer value is not truncated.*

As an example, if **Width** *is set to 10 and the attribute integer value is 23 and* **Precision** *is 3, then the label is "   023". That is, an extra seven spaces is added to the left of the attribute integer value.*

if blank, the attribute integer value text is used with no modification.

Precision

*If the attribute is type **text** then not applicable.*

*If the attribute is type **real**:*

if positive, the number of decimal places after the decimal point.

if negative, the absolute number is the number of decimal places to first calculate the text of the attribute real value for but then any trailing zeros after the decimal place are removed.

***Important Note:** this use of positive and negative for not eliminating/eliminating trailing zeros after the decimal place is the opposite to plot parameters.*

*If the attribute is type **integer**:*

if the number of digits in the attribute integer value is less than the absolute value of **Precision***, then left 0 padding is used to bring the number of digits in the text up to the absolute value of* **Precision***.*

As an example, if **Width** *is set to 10 and the attribute integer value is 23 and* **Precision** *is 3, then the label is "   023". That is, an extra seven spaces is added to the left of the attribute integer value.*

Prefix/Suffix

prefix/suffix for the text

Label name

the created super strings are given this name

Relative angle

*if **yes**, the angle of the label is the angle of the segment that the segment text is coming from **PLUS** the Angle of the text given in the Textstyle Data column.*

*if **no**, the angle of the label is the Angle of the text given in the Textstyle Data column.*

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

Go to the next section 8.7.1.10 Element Attribute Text Data or return to 8.7.1 Create/Edit Label Map File.

8.7.1.10 Element Attribute Text Data

Element Attribute Text Data labels the string attributes of string.

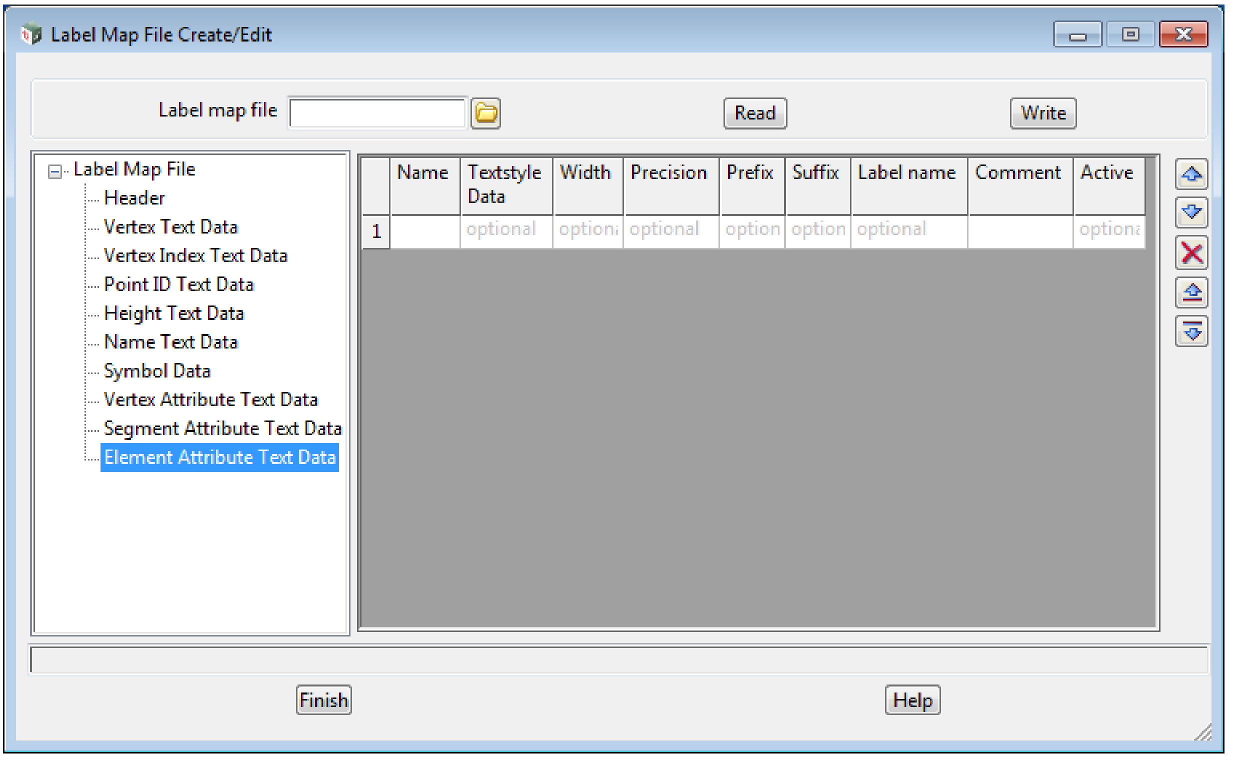
The ***Element Attribute Text Data*** grid specifies ***which strings*** *and* ***which string attributes are selected*** and how a label containing the ***value of the string attribute*** is created for each string attribute of the selected strings.

For each selected string and selected string attribute, text strings are created with the label as the text and the position of the text string as:

the position of the first vertex for a point string

the centroid of the string for closed strings that are not point strings

the mid chainage position for open strings that are not a point strings.



Name **select name** menu for fields

for the selected strings and the selected ***string attributes*** *of those strings, a label is created from the* ***value*** *of the string attribute using the parameters in this line of the Label Map file. See* Selecting Strings and Attributes Using Name:.

For each selected string and selected string attribute, text strings are created with the label as the text and the position of the text string as:

the position of the first vertex for a point string

the centroid of the string for closed strings that are not point strings

the mid chainage position for open strings that are not a point strings.

Textstyle data **select textstyle data** menu for fields

textstyle data for labels. This controls the units, size, colour, border etc for the created text

Width

If the attribute is type ***text****:*

if non blank, the minimum number of characters that is allowed for the attribute text before applying the prefix and suffix.

If the number of characters in the attribute text is less than **Width** *characters, then additional* *spaces are added to the left of the attribute text so that there are* **Width** *characters.*

If the number of characters in the attribute text is greater than **Width***, then* **Width** *is ignored. Hence the attribute text is not truncated.*

For example if **Width** *is set to 20 and the attribute text has only ten characters, then an extra 10 spaces are added to the left of the attribute text before applying the Prefix/Suffix.*

if blank, the attribute text is used with no space padding.

If the attribute is type ***real****:*

if non blank, the minimum number of characters allowed for the attribute real value (including **Precision***) before applying the prefix and suffix.*

If the number of digits in the attribute real value (including **Precision***) is less than* **Width** *characters, then additional spaces are added to the left of the text so that there are* **Width** *characters.*

If the number of digits in the attribute real value (including **Precision***) is greater than* **Width***, then* **Width** *is ignored. Hence the attribute real value is not truncated (other than by* **Precision***).*

As an example, if **Width** *is set to 10 and the attribute real value is 23.2 and* **Precision** *is 3, then the label is "   23.200". That is, an extra four spaces is added to the left of the vertex index value.*

if blank, the attribute real value with **Precision** *is used with no further modification.*

If the attribute is type ***integer****:*

if non blank, the minimum number of characters allowed for the attribute integer value (including **Precision***) before applying the prefix and suffix.*

If the number of digits in the attribute integer value (including **Precision***) is less than* **Width** *characters, then additional spaces are added to the left of the text so that there are* **Width** *characters.*

If the number of digits in the attribute integer value (including **Precision***) is greater than* **Width***, then* **Width** *is ignored. Hence the attribute integer value is not truncated.*

As an example, if **Width** *is set to 10 and the attribute integer value is 23 and* **Precision** *is 3, then the label is "   023". That is, an extra seven spaces is added to the left of the attribute integer value.*

if blank, the attribute integer value text is used with no modification.

Precision

If the attribute is type ***text*** *then not applicable.*

If the attribute is type ***real****:*

if positive, the number of decimal places after the decimal point.

if negative, the absolute number is the number of decimal places to first calculate the text of the attribute real value for but then any trailing zeros after the decimal place are removed.

Important Note: *this use of positive and negative for not eliminating/eliminating trailing zeros after the decimal place is the opposite to plot parameters.*

If the attribute is type ***integer****:*

if the number of digits in the attribute integer value is less than the absolute value of **Precision***, then left 0 padding is used to bring the number of digits in the text up to the absolute value of* **Precision**.

As an example, if **Width** *is set to 10 and the attribute integer value is 23 and* **Precision** *is 3, then the label is "   023". That is, an extra seven spaces is added to the left of the attribute integer value.*

Prefix/Suffix

prefix/suffix for the text

Label name

the created super strings are given this name

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

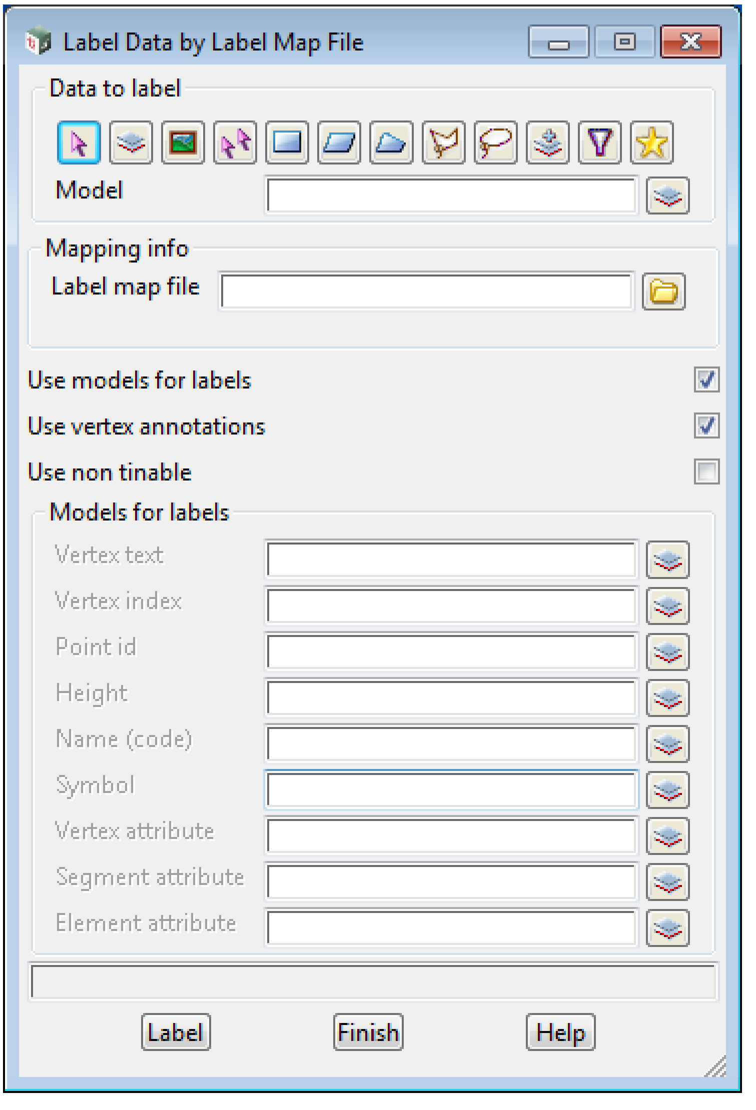
Go to the next section 8.7.2 Apply a Label Map File or return to 8.7.1 Create/Edit Label Map File.

8.7.2 Apply a Label Map File

Position of option on menu: **File I/O =>Label Map files =>Apply**

Use a *Label Map File* to create text labels for all strings in the selected data source and that match the criteria in the sections of the *Label Map File*.

Selecting **Apply** brings up the **Label Data by Label Map File** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data is to be selected and processed by the Label Map File.

Label map file	input		*.lmf

name of the Label Map File to process the selected data.

Use models for labels	tick box	tick	

*if **ticked**, the **Models for labels** section is displayed and labels are only created for the type of data that has a non blank model name in the "Model for labels" area. The created labels go into that model.*

*If **ticked**, then the* **Vertex text** *field is the model for the text.*

*If **not ticked**, then the model for the created text is the original data source model name with the text in the* **Vertex text** *field used as pre*post text for the model name.*

For example, Vertex text labels will only be created if the **Vertex text** *field is not blank.*

*If **not ticked**, the **Pre*Post for models** section is displayed and labels are only created for the type of data that has a non blank Pre*post field. When it is non blank the created text goes into a model of the same name as the original data source, but with a prefix/-postfix given by the pre*post field.*

Use vertex annotations tick box tick

Vertex text, Vertex index, Point id, Height, Name (code), Symbol, Vertex attribute, Segment attribute, Element attribute

*If **Use models for labels** is **ticked**, the above panel fields are models for the created data.*
*If **Use models for labels** is **not ticked**, the above panel fields are used as **Pre*Post** text in conjunction with the original data source model name to create model names.*

If the fields are blank, then no text is created by the Label Map File for that type of text.

Label button

process the selected strings and label them.

8.7.3 Convert .lmf Files to .label_mapfile

Position of option on menu: **File I/O =>Label Map files =>Convert .lmf to .label_mapfile**

The **Label map file convert** option converts a label map file in the pre-V9 format to the XML label map file format introduced in ***12d Model 10***.

Selecting **Convert .lmf to .label_mapfile** displays the **Label Map File Convert .lmf to .label_mapfile** panel.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
File to convert from	file box		*.lmf

name of the pre-V10 label map file to be converted to the new label map file format

File to convert to	file box		*.label_mapfile

name of the new converted label map file

Convert	button		

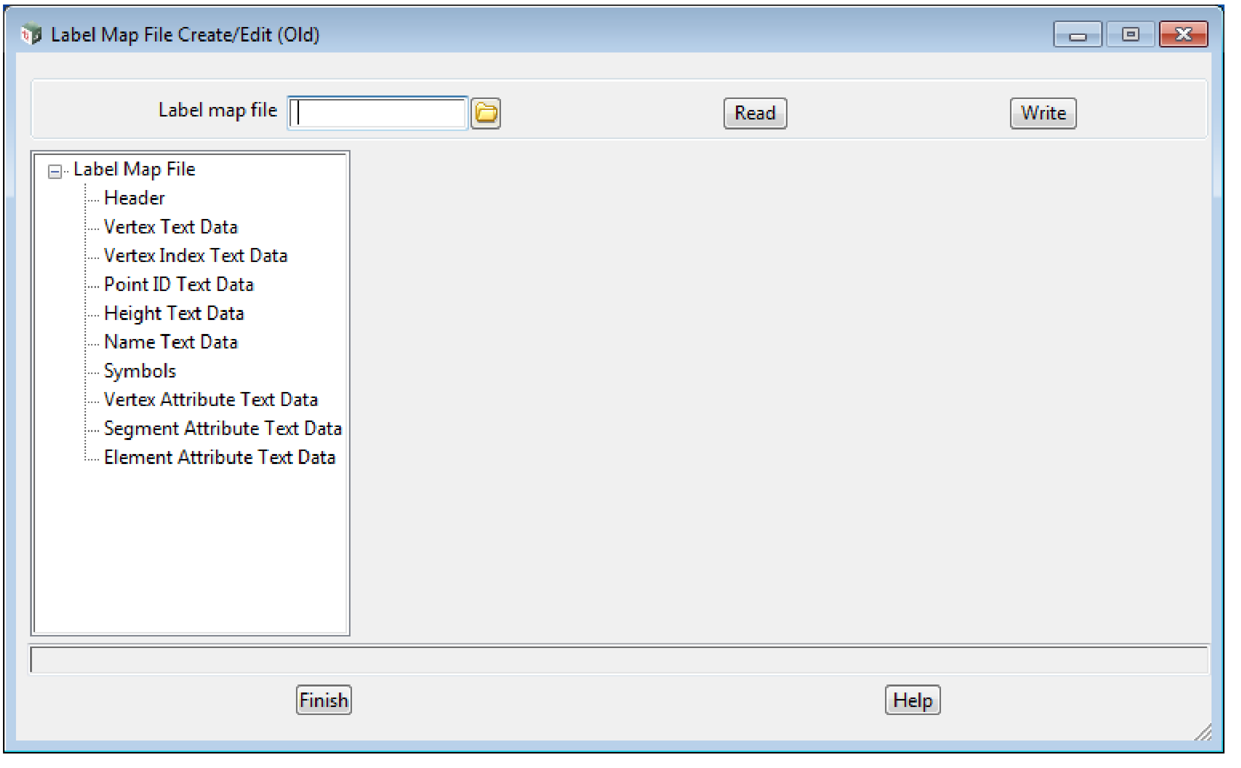
convert a file in the pre-V10 label map file format to the new label map file format.

8.7.4 Create/Edit Pre-V10 Label Map File

Position of option on menu: **File I/O =>Label Map files =>Create/edit .lmf**

This option is for editing the superseded label map file format and should not be used.

Please use the new Label Map File format Created/edit option (8.7.1 Create/Edit Label Map File).



8.8 Map Files

Position of menu: **File I/O =>Map files**

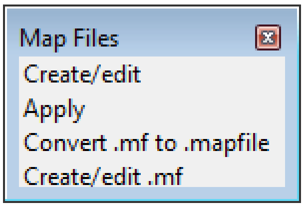
The main options under **Map files** is **Create/edit** which creates and edits ***12d Map Files*** which are used in many ***12d Model*** panels

The Map File format uses is the *.mapfile* format (an XML format) introduced in ***12d Model 10.***

There is also an option to convert the older format, .mf, to the XML .mapfile.

Note: if the ***Create/edit*** option is used on an old format file, it is automatically converted to the new format.

The **Map files** walk-right menu is



create/edit a Map File
apply a Map File
convert pre-V9 Map File format to XML format Map Files
create/edit a pre-V9 Map File

For *Create/edit*, go to 8.8.1 Create/Edit a Map File
Apply 8.8.5 Apply Map File
Convert .mf to .mapfile 8.8.6 Convert .mf Files to .mapfile
Create/edit .mf 8.8.7 Create/Edit .mf File

8.8.1 Create/Edit a Map File

Position of option on menu: **File I/O =>Map files =>Map File**

The **Map file** option is used to **create** or **edit** ***12d Map Files*** which are used in many 12d Model options for reading in data, reducing surveys, mapping existing data *etc*.

A *12d Map file* consists of sections of tables or grids of keys or entity-masks which are used to select specific data to set many string properties such as string name, colours, linestyles, symbols, models, extrusions and tinability for the strings that are being created and/or modified by the ***12d Model*** option.

For example, the 8.1.5.1 X Y Z S Input options reads in an XYS file and creates strings of (x,y,z) data with string names for the lines strings. A ***12d Map File*** can be used in option to give the strings other properties such as colour, linestyle, colour fills, pipe diameters *etc*.

The Map File consists of sections such as Basic, Fills and Pipes, which are displayed as a tree. The strings are processed by each section of the tree, going in order from the top to the bottom of the Map File tree.

Each section of the Map File uses matching of specific string information against a **Key** to select strings and/or vertices and/or segments for processing by that section of the Map File.

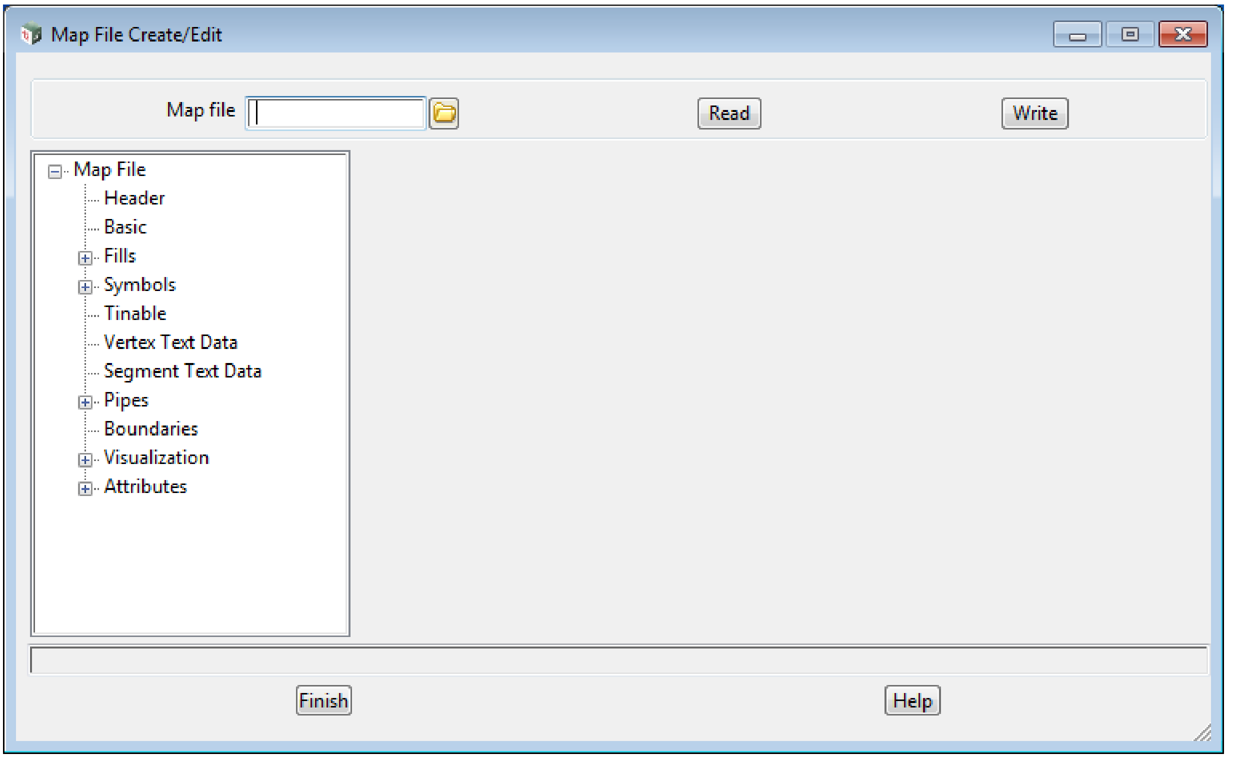
The **Basic** section is the first section used and takes the data that had been created by the option and uses the Entity Name and string attributes created by the option to set string properties such as string name, colour, point-line type, linestyle, weight and model. What the Entity Name actually is depends on the particular option that the Map File is being used in.

For example, the Survey Data Reduction option uses the ***field code*** as the entity-name, the xyzs reader uses the ***string name*** in the xyzs file as the entity-name but in the DWG reader, no entity name exists and the ***AutoCAD layer*** can be used as the entity-name. For the DGN reader, a combination of Level, Colour, Linestyle and Weight can be the entity name.

After the **Basic** section is applied, the string has a definite string name that is used as part of the matching key in all the other sections of the Map File.

In the other sections of the Map File, the ***Key*** is a combination of string names, and/or string attribute values **Att key**, and/or vertex attribute values **Vertex Att key** and/or segment attribute values **Segment Att key**

Selecting **Map file** brings up the **Map File Create/Edit** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Map file	input		*.mapfile, *.mf

name of the map file to be created or edited.

If it is a .mf file, it is automatically converted to the new xml format and written out as a .mapfile.

Read button

read in the map file given in the **map file** *field and load the data into the fields of the table. If the file doesn't exist, an error message is given.*

Write button

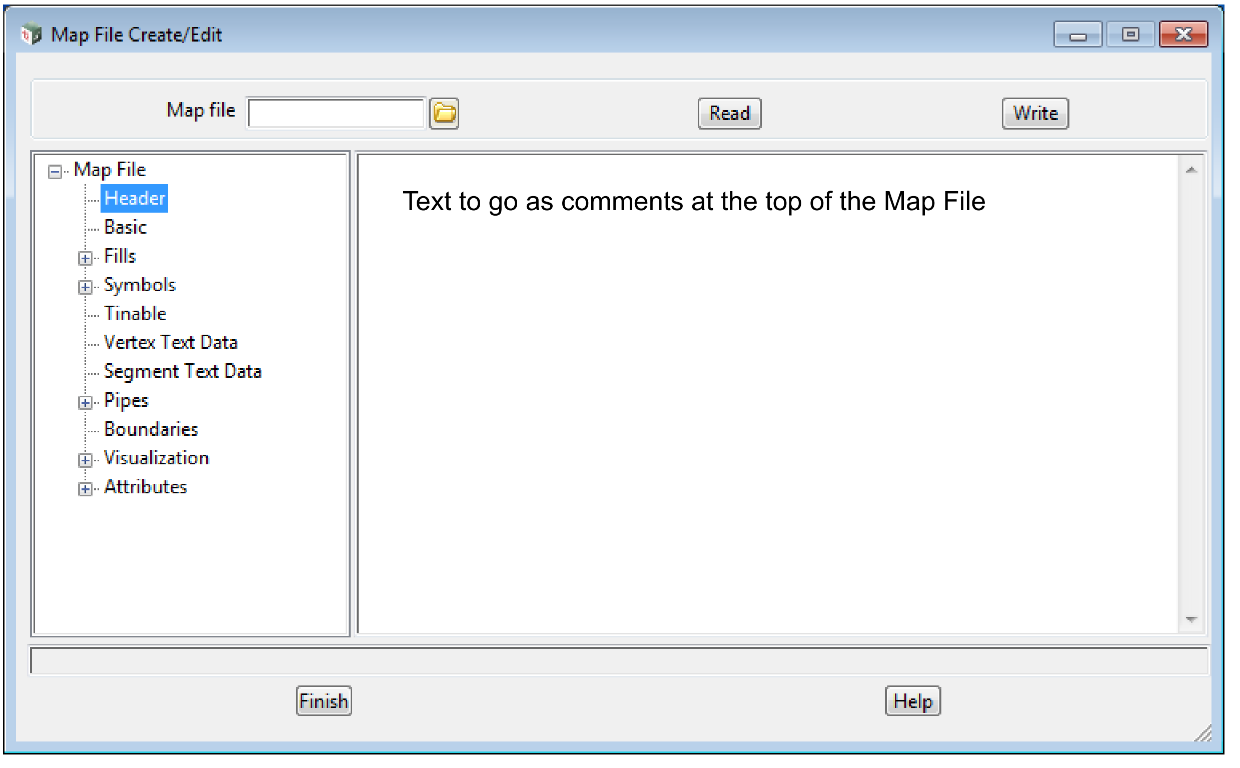
write out the data in the table to the file name given in the **Map file** *field. If the file already exists, a replace-cancel panel checks to see if the existing file is to be over written - if cancel is selected, nothing is written out.*

For more information on *Header*, go to 8.8.1.1 Header
Basic 8.8.1.2 Basic
Fills 8.8.1.3 Fills
Symbols 8.8.1.4 Symbols
Tinable 8.8.1.5 Tinable
Vertex Text Data 8.8.1.6 Vertex Text Data
Segment Text Data 8.8.1.7 Segment Text Data
Pipes 8.8.1.8 Pipes
Boundaries 8.8.1.9 Boundaries
Visualization 8.8.1.10 Visualization
Attributes 8.8.1.11 Attributes

8.8.1.1 Header

Header just has lines of comment text. A common use is to document which client the file is to be used for, or the revision history.

The text written in the right hand side of the panel is written out as comments at the top of the Map file.

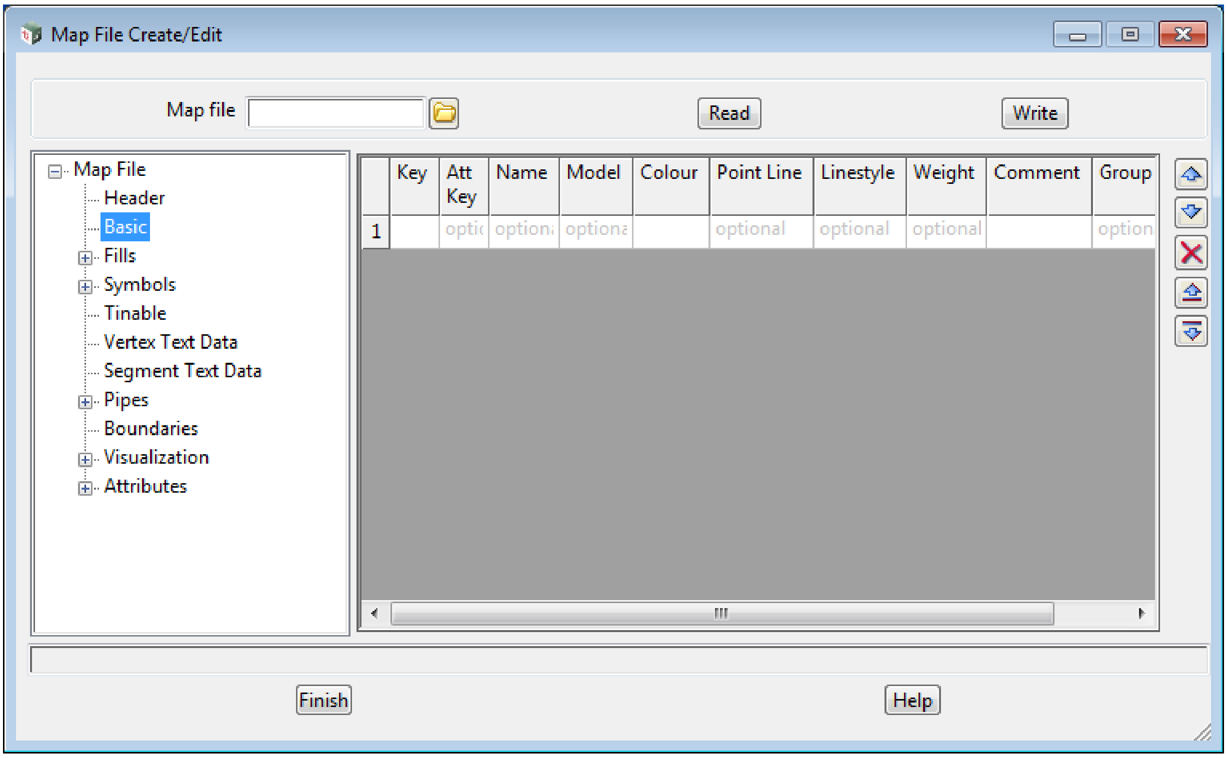


Go to the next section 8.8.1.2 Basic or return to 8.8.1 Create/Edit a Map File.

8.8.1.2 Basic

The **Basic** grid selects data using the **Key** and **Att Key** and sets the name, model, colour, linestyle, point-line type, and weight for the created strings.

Note: the **final string name** is the **Name** set by the ***Basic*** grid, or if none is set in the *Basic* grid, then it is the **Entity Name**.



Field Description Type Defaults Pop-Up

Processing Using Key and Att Key for Basic Node

when an Entity finds a first match with **Key** *and* **Att Key** *in the grid (see* Matching Using Key and Att Key:*), the created string is given the properties as defined in the parameters of the matching line.*

Key and Att Key **select colour pop-up**

Key *is a Text grid cell and the text entered into* **Key** *can include wild cards* *** *and wild characters* ***?***.
The ***Entity Name*** *is matched against* **Key**.
Att Key *is an Attribute Data grid cell which contains the definition of the attributes and their values that are to be matched against. There can be more than one attribute in the Attribute Data but they must have unique names.*
The ***Entity Attributes*** *are matched against the attribute details in* **Att Key.**
To access ***Att Key****, click LB on the* ***Att Key*** *field to highlight the field, then click LB again to bring up the* **Attribute Data** *panel. To enter data see* Attribute Data Panel.

Matching Using Key and Att Key:

The Entity being mapped needs an Entity Name (which may be blank) and can also have Entity Attributes. The Entity may even have some other properties such as colour, linestyle etc.
Starting with the first line of the grid, matching and processing occurs as follows
If **Key** *and* **Att Key** *are not blank, and a match of the Entity Name occurs with* **Key** *and a match of the Entity Attributes occurs with* **Att Key**, *then the rest of the fields for this line of the Map File grid are used on the created string.*
If **Key** *in not blank and* **Att Key** *is blank, and a match of the Entity Name occurs with* **Key**, *then the rest of the fields for this line of the Map File grid are used on the created string.*

If **Key** *in blank then no match occurs and this line of the Map File grid is ignored.*

If a ***match occurs****, then no tests for matches against* **Key** *and* **Att Key** *further down in the grid are made.*

If ***no match occurs****, then this line of the map file grid is ignored and a test for a match is made against the next line of the grid.*

Note: all the fields on the line after **Att Key** can have attribute substitution from string attributes. See 8.8.3 Map File Substitution by Attributes.

Name **select name pop-up**

if non-blank, the created string is given this **Name**.
if blank, the original Entity Name (which matched against **key***) is used as the string name.*

Important Note*: the resulting string name is used for matching in all the subsequent Map File nodes (e.g. Fills, Symbols, Pipes, Visualisation etc).*

Model **select model pop-up**

if non-blank, the model to be used for all strings produced by this line of the map file grid.
if blank, the string's model is not modified. Note: some options may allow **Model** *to be ignored.*

Colour **select colour pop-up**

if non-blank, the colour to be used for all strings produced by this line of the map file grid.
if blank, the string's colour is not modified.

Point Line **select choice point, line**

if non-blank, the breakline type to be used for all strings produced by this line of the map file grid.
if blank, the string's point-line type is not modified. If data is coming from a field file, then the data may be coded as a line or point string.

Linestyle **select linestyles pop-up**

if non-blank, the linestyle to be used for all strings produced by this line of the map file grid.
if blank, the string's linestyle is not modified.

NOTE: *If the point-line type is point then the linestyle is used as a symbol at each vertex of the string. The size of the symbol at each point is given in the linesytle definition. If a symbol at each point is required whose size, colour and rotation may need to be modified at each point, then the* ***Symbol*** *section must be used to place a symbol at each vertex and a world style must be used in the symbol name field in the* ***Symbol*** *tab. See* 8.8.1.4 Symbols.

Weight

if non-blank, the weight to be used for all strings produced by this line of the map file grid.
if blank, the string's weight is not modified.

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

Group

the optional **Group** *column is not used in the* ***Map File*** *but is used in the* ***Names.4d*** *file which has the same XML structure.*

If you ***don't*** *want the* ***Group Column*** *to be used in any grid then you have to set the environment variable* ***USE_TREE_NAME_BOX_4D*** *to* ***0****. By default is has the value* ***1****.*

IMPORTANT NOTE:

Regardless of the group/subgroup structure, the search order for **finding a match** with the key in the ***Basic*** node of the ***Map File*** is still the order that the rows occur in the ***Basic*** grid.

Currently the Group Column will only be visible if you have the environment variable **USE_TREE_NAME_BOX_4D** set to **1**.

To assist is debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

Go to the next section 8.8.1.3 Fills or return to 8.8.1 Create/Edit a Map File.

8.8.1.3 Fills

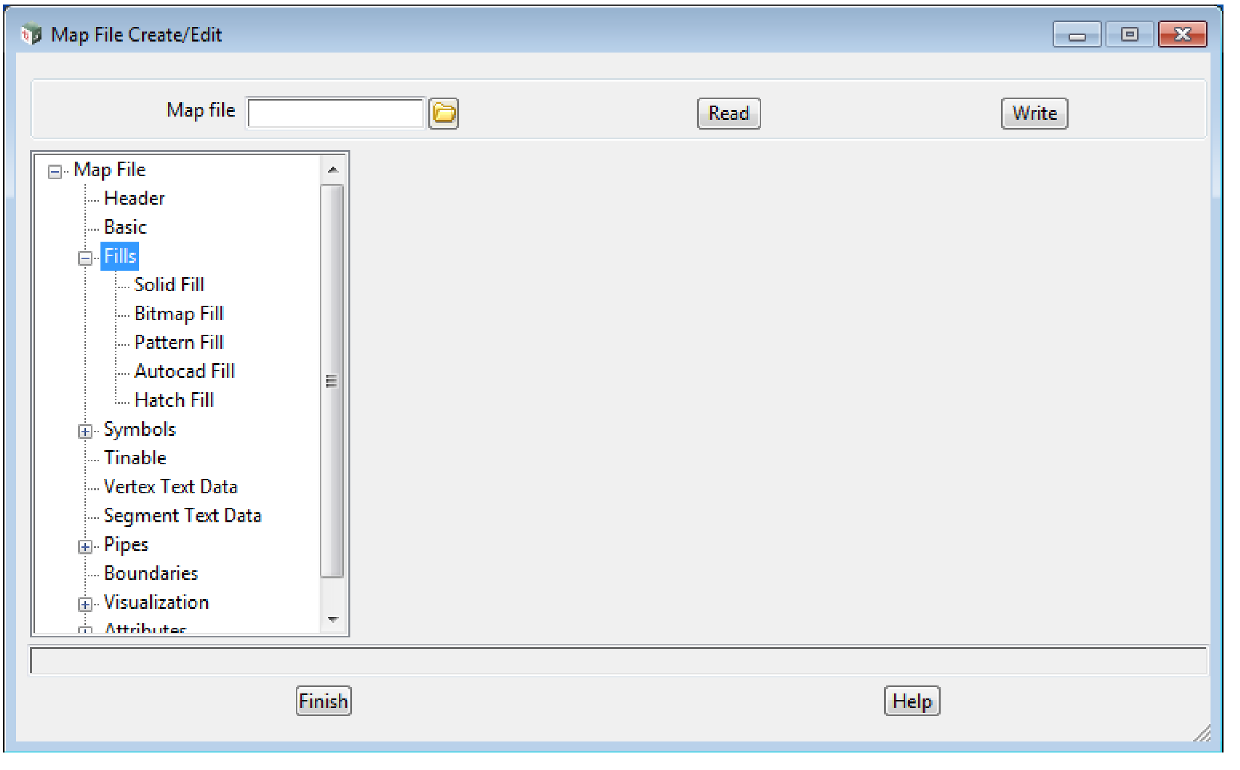
Fills creates fills for the selected super strings (closed and non-closed).

Super strings can support more than one fill type per string. For example, a super string can have a solid fill, and an I Acad fill at the same time.

The order of drawing the fill types is: solid, bitmap, pattern, hatch, acad.

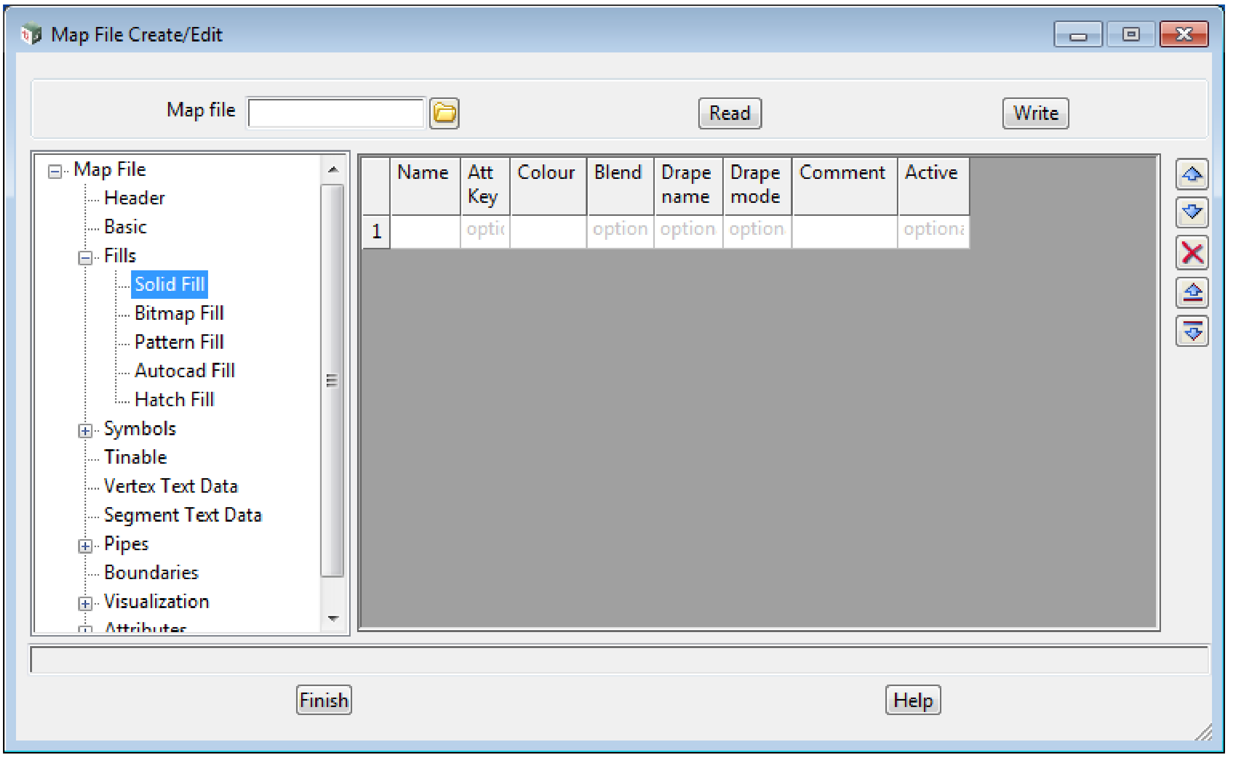
For more information on *Solid Fill*, go to 8.8.1.3.1 Fills >Solid Fill
Bitmap Fill 8.8.1.3.2 Fills >Bitmap Fill
Pattern Fill 8.8.1.3.3 Fills >Pattern Fill
Autocad Fill 8.8.1.3.4 Fills >Autocad Fill
Hatch Fill 8.8.1.3.5 Fills >Hatch Fill
Multiple Fills 8.8.1.3.6 Multiple Fills

Note: there are settings for Plan Views that determines how fills are drawn on plan views (see 9.1.1.4 Face Flags for View).



8.8.1.3.1 Fills >Solid Fill

Solid Fill sets whether super strings (closed and non-closed) are filled with a "solid" colour, and optionally with some level of blending (transparency).



Field Description Type Defaults Pop-Up

Processing Using Name and Att Key for Fills>Solid Fill Node

when a string finds a first match with **Name** *and* **Att Key** *in the grid (see* Matching Using Name and Att Key:*), the string is given a Solid Fill as defined in the parameters of the matching line.*

Name and Att Key

Name *is at Text grid and the text entered into* **Name** *can include wild cards* * *and wild characters* ?*.*
The string name is matched against **Name**.
Att Key *is an Attribute Data grid cell which contains the definition of the attributes and their values that are to be matched against. There can be more than one attribute in the Attribute Data but they must have unique names.*
The ***string*** *attributes are matched against the attribute details in* **Att Key.**
To access ***Att Key*** *data, click LB on the* ***Att Key*** *field to highlight the field, then click LB again to bring up the* **Attribute Data** *panel. To enter data, see* Attribute Data Panel.

Matching Using Name and Att Key:

Strings are first created via the ***Basic*** *node and given a string name of either the Entity Name or the text given in the Name column of the Basic Map File grid.*
Starting with the first line of the grid, matching and processing occurs as follows
If **Name** *and* **Att Key** *are not blank, and a match of the string name occurs with* **Name** *and a match of the string attributes occurs with* **Att Key***, then the rest of the fields for this line of the Map File grid are used on the string.*
If **Name** *in not blank and* **Att Key** *is blank, and a match of the string name occurs with* **Name***, then the rest of the fields for this line of the Map File grid are used on the string.*
If **Name** *in blank then no match occurs and this line of the Map File grid is ignored.*
If a ***match occurs****, then no tests for matches against* **Name** *and* **Att Key** *further down in the grid are*

made.

*If **no match occurs**, then this line of the map file grid is ignored and a test for a match is made against the next line of the grid.*

Note: all the fields on the line after **Att Key** can have attribute substitution from string attributes. See 8.8.3 Map File Substitution by Attributes.

Colour **select colour pop-up**

colour to use to fill the super string with. If the super string is not closed, the fill will apply as if the super string was closed.

This field can not be blank.

Blend measures box **At Point, Point to Point, String from Point, String to Point)**

*if **not blank**, the blend value to use for the solid fill.This value is between 0.0 and 1.0 and controls the amount of transparency of the fill.*

*if **blank** the blend value of 1.0 is assumed.*

TODO: measures box makes no sense....

Drape name

*if **not blank**, the string is given this as a Drape name. That is, a name that is used to specify which strings are draped as the solid filled super string onto a tin.*

*This allows the plan shape of a polygon to be drawn on top of a tin. A good example would be a polygon defining the shape of a "left turn arrow" which the user wants drawn on top of the road design tin. If the user adds the road design tin (and the tin has a **Plan polygon drape name** set to the* **Drape name** *set here), then the model containing the polygon onto an OpenGL perspective view, the "left turn arrow" will be drawn after the tin and draped onto the tin.*

*If **blank**, no drape name is given to the string.*

To access the render settings panel use **View=>Visualisation=>Tin Render Settings** *(see* 12.13.1 Tin Render Settings*). To see the effect of the drape requires the visualisation module.*

Drape mode choice box Colour, Colour to texture

*if **Drape name** is blank this field is not used.*

if non blank, and the value is

Colour, *the solid fill colour of the super string is used to draw onto the surface of the tin.*

Colour to texture, *the value of the solid fill colour is mapped via the **Texture mapping** field of the* **Tin Render Settings** *panel and a texture drawn inside the polygon draped onto the tin.*

***Drape mode** cannot be blank if **Drape name** is present.*

To access the render settings panel use **View=>Visualisation=>Tin Render Settings** *(see* 12.13.1 Tin Render Settings*). To see the effect of the drape requires the visualisation module.*

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

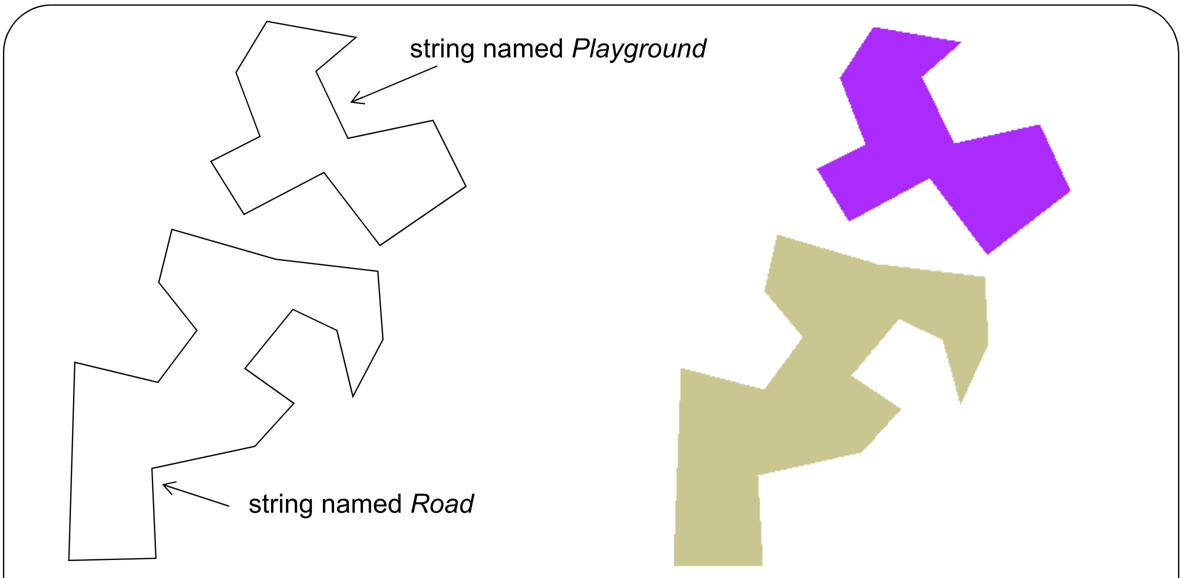
To assist is debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

Example of Solid Fill

Applying the following **Solid Fill** entries to polygons called *Road* and *Playground*:

	Name	Att Key	Colour	Blend	Drape name	Drape mode	Comment
1	Road*		concrete1	1	drape	Colour to texture	
2	Playground		purple	1	drape	Colour	

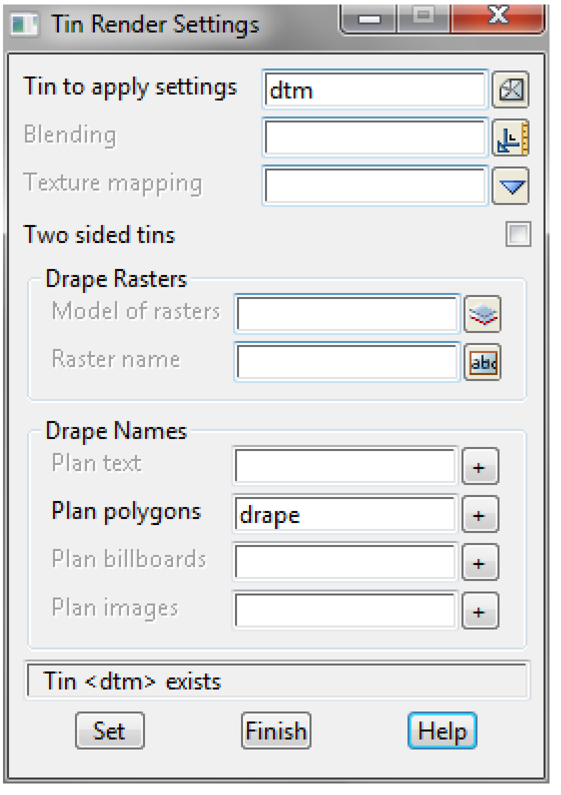
produces the solid filled polygons



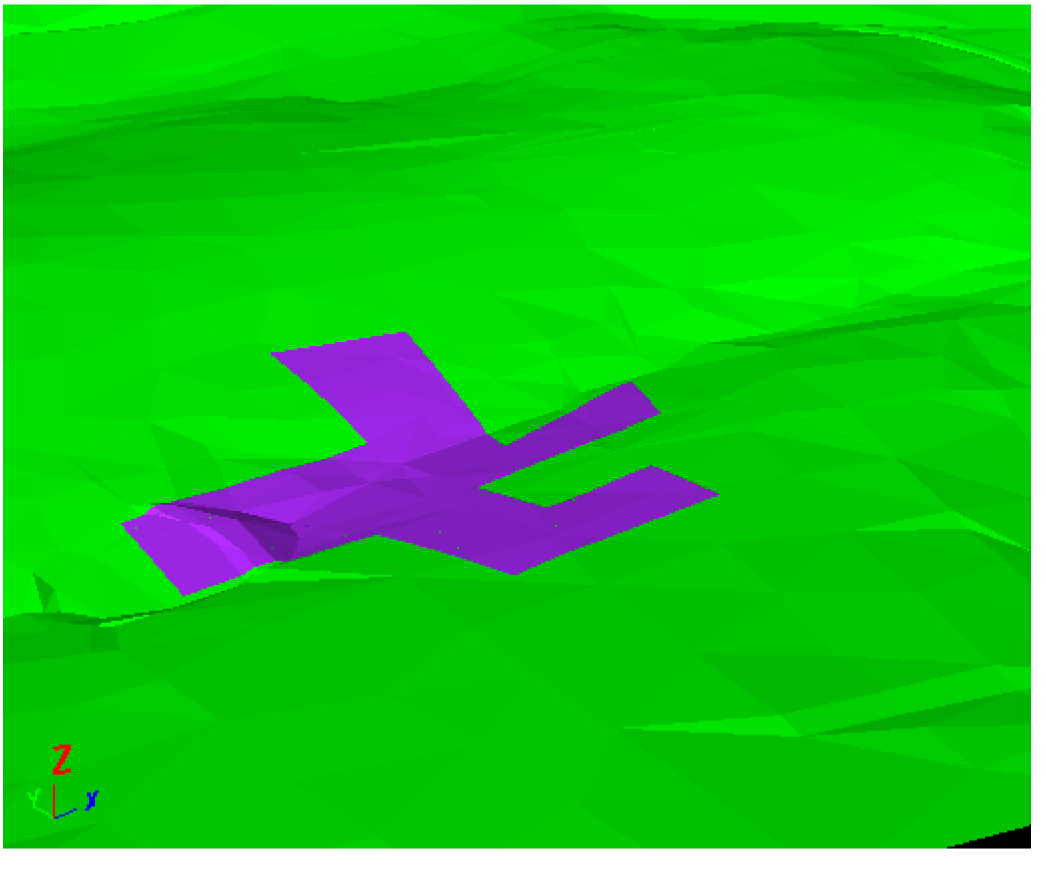
Polygons Before Applying Map File

Polygons After Applying Map File

If the above solid fill details have been applied to a closed super string, and using the **Tin Render Settings** panel to set **Plan polygon** field to "drape" for the tin called "dtm"



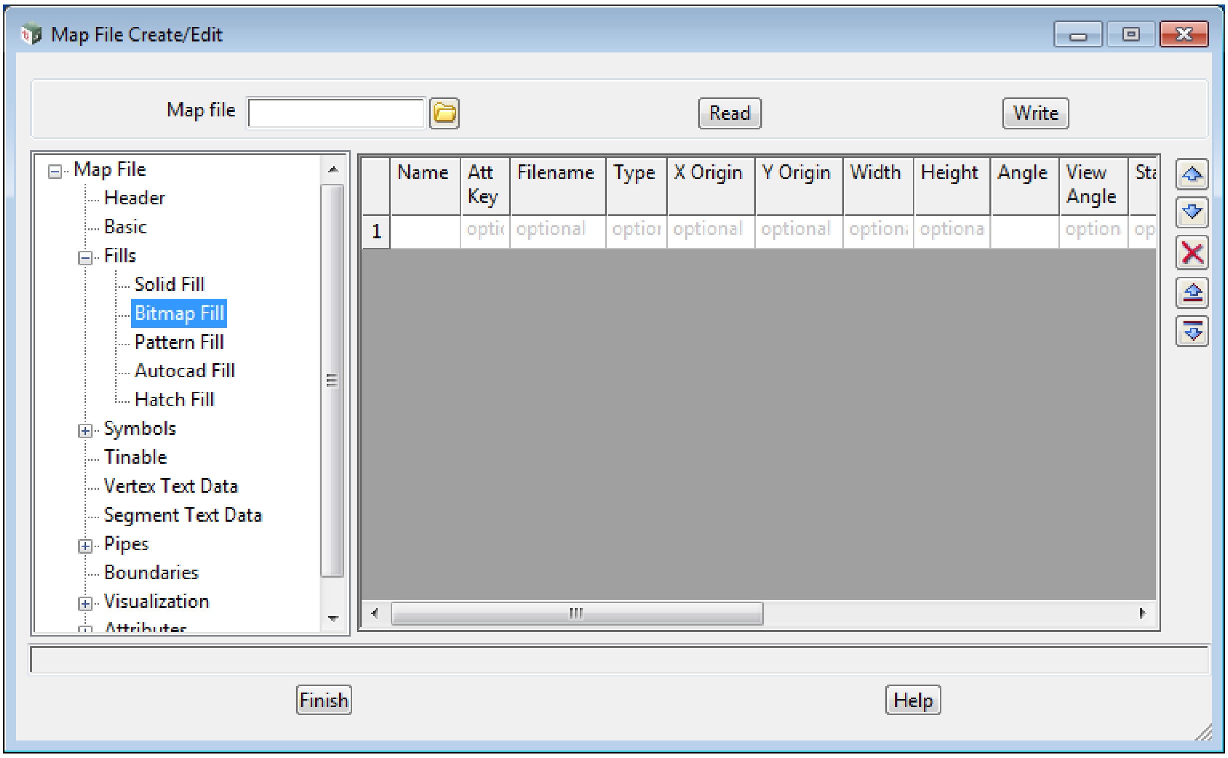
the result will be



Go to the next section 8.8.1.3.2 Fills >Bitmap Fill or return to 8.8.1.3 Fills or 8.8.1 Create/Edit a Map File.

8.8.1.3.2 Fills >Bitmap Fill

Bitmap Fill sets whether super strings (closed and non-closed) are filled with a bitmap pattern.



Processing Using Name and Att Key for Fills>Bitmap Fill Node

when a string finds a first match with **Name** *and* **Att Key** *in the grid (see* Matching Using Name and Att Key:*), the string is given a Bitmap Fill as defined in the parameters of the matching line.*

Name and Att Key

Name *is at Text grid. The* ***string name*** *is matched against* **Name**.
Att Key *is an Attribute Data grid cell. The* ***string attributes*** *are matched against the attribute details in* **Att Key**.
For further information on what is allowed in **Name** *and* **Att Key***, and how to access* **Att Key***, see* Name and Att Key.

Note: all the fields on the line after **Att Key** can have attribute substitution from string attributes. See 8.8.3 Map File Substitution by Attributes.

Filename file .*bmp

The bitmap file to be used for filling. Only BMP files are currently supported.

Type choice box **paper, world, device, screen (device), pixels (device)**

The units of size.This means the bitmap can be drawn with either world size, paper size (mm) or device size.

X/Y Origin measures box **At Point, Point to Point, String= from Point, String to Point)**

The x/y co-ordinate of the origin point of the repeated pattern.

Width/Height measures box **At Point, Point to Point, String= from Point, String to Point)**

The width/height of the bitmap in the selected units.

Angle measures box **At Point, Point to Point, String= from Point, String to Point)**

The orientation of the bitmap pattern relative to the x axis measured in the anti-clockwise direction. The units are dms (degrees minutes and seconds) and typed in using 4.17.1 HP Notation.

View Angle choice box yes, no

if not blank, this field controls whether **Angle** *is relative to the x axis or to the plotting x axis.*

If **yes** *and we are plotting,* **Angle** *is measured relative to the x axis of the plot rotation.*

If **no**, **Angle** *is measured absolute to the world x axis*

If blank, **Angle** *is measured absolute to the world x axis.*

Stagger measures box **At Point, Point to Point, String= from Point, String to Point)**

if not blank, this field allows each alternate row of bitmaps to be staggered (offset) by the specified value. **Stagger** *is measured in the selected units along the x axis in the rotated system.*

If blank, a **Stagger** *of 0.0 is assumed.*

Space X/Y measures box **At Point, Point to Point, String= from Point, String to Point)**

The distance in the selected units along the x/y axis in the rotated system between each instance of the bitmap.

Transparent **select colour pop-up**

The transparency colour within the bitmap.This means any pixel of this colour within the bitmap is not drawn.

Solid Colour **select colour pop-up**

if not blank, when the bitmaps become too small to be legible, instead of drawing the bitmaps, the super string is solid filled with colour ***Solid colour***.

Solid Blend measures box **At Point, Point to Point, String= from Point, String to Point)**

if not blank, and **Solid Colour** *is specified, the solid fill is drawn with blend value* **Solid Blend**. *The blend value is between 0.0 and 1.0.*

If blank, **Solid Blend** *is assumed to be 1.0.*

Drape Name

reserved for future use

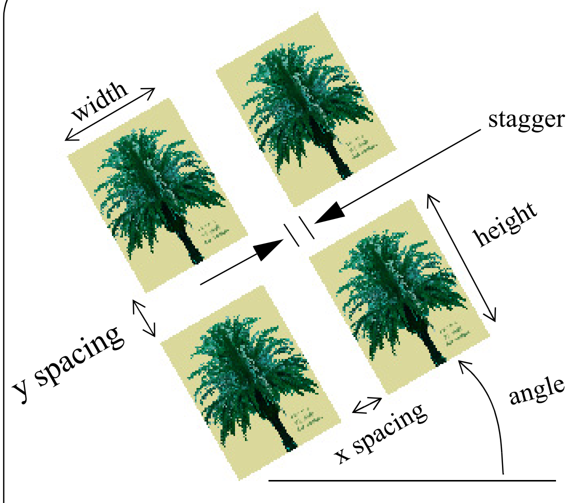
Comment and Active

See 8.8.2 Comment and Active Column in Grids.

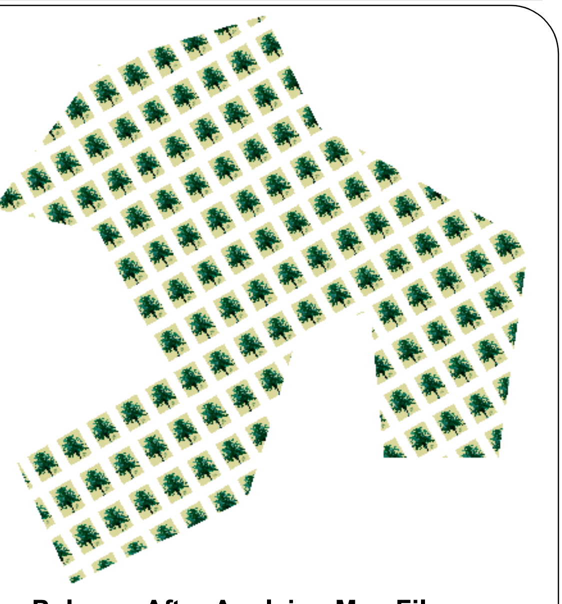
To assist is debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

Examples of Bitmap Fill

	Name	Att Key	Filename	Type	X Origin	Y Origin	Width	Height	Angle	View Angle	Stagger	Space X	Space Y	Transparent	Solid Colour	Solid Blend	Drape name	Comment
1	bfl06		$LIB\12D_palm 1.bmp	world	100	100	9	12	30		1	4	4		off yellow	0.5		



Definitions for Bitmap Fill

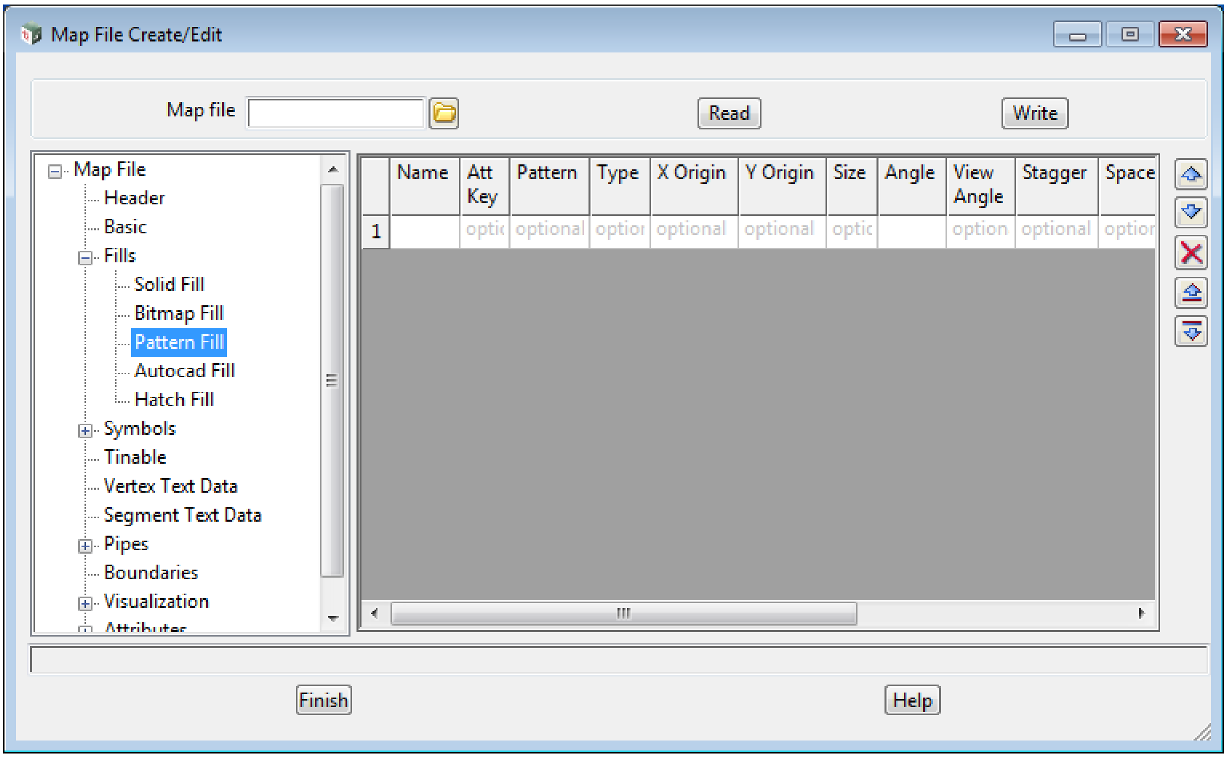


Polygon After Applying Map File

Go to the next section 8.8.1.3.3 Fills >Pattern Fill or return to 8.8.1.3 Fills or 8.8.1 Create/Edit a Map File.

8.8.1.3.3 Fills >Pattern Fill

Pattern Fill uses a pattern from the ***12d Model*** pattern file, *12d.patttern*, to fill the selected super strings.



Processing Using Name and Att Key for Fills>Pattern Fill Node

when a string finds a first match with **Name** *and* **Att Key** *in the grid (see* Matching Using Name and Att Key:*), the string is given a 12d Pattern Fill as defined in the parameters of the matching line.*

Name and Att Key

Name *is at Text grid. The* ***string name*** *is matched against* **Name**.
Att Key *is an Attribute Data grid cell. The* ***string attributes*** *are matched against the attribute details in* **Att Key**.
For further information on what is allowed in **Name** *and* **Att Key**, *and how to access* **Att Key**, *see* Name and Att Key.

Note: all the fields on the line after **Att Key** can have attribute substitution from string attributes. See 8.8.3 Map File Substitution by Attributes.

Pattern **select Pattern pop-up**

The pattern name to be applied.These patterns are defined in the file ***patterns.12d***

For more information on the 12d pattern definition please go to the section 42.2.5 12d Patterns File.

Type choice box paper, world, device

The units of size.This means the pattern can be drawn with either world size, paper size (mm) or device size.

X/Y Origin measures box **At Point, Point to Point, String= from Point, String to Point)**

The x/y co-ordinate of the origin point of the repeated pattern.

Size measures box **At Point, Point to Point, String=**

from Point, String to Point)

The size of the pattern in the selected units given in **Type**.

Angle	measures box	**At Point, Point to Point, String= from Point, String to Point)**

The orientation of the pattern relative to the x axis measured in the anti-clockwise direction. The units are dms (degrees minutes and seconds) and typed in using 4.17.1 HP Notation.

View Angle	choice box	yes, no

if not blank, this field controls whether **Angle** *is relative to the x axis or to the plotting x axis.*

If **yes** *and we are plotting,* **Angle** *is measured relative to the x axis of the plot rotation.*

If **no**, **Angle** *is measured absolute to the world x axis*

If blank, **Angle** *is measured absolute to the world x axis.*

Stagger	measures box	**At Point, Point to Point, String= from Point, String to Point)**

if not blank, this field allows each alternate row of patterns to be staggered (offset) by the specified value. **Stagger** *is measured in the selected units along the x axis in the rotated system.*

If blank, a **Stagger** *of 0.0 is assumed.*

Space X/Y	measures box	**At Point, Point to Point, String= from Point, String to Point)**

The distance in the selected units along the x/y axis in the rotated system between each instance of the pattern.

Solid Colour		**select colour pop-up**

if not blank, when the pattern become too small to be legible, instead of drawing the pattern, the super string is solid filled with colour ***Solid colour***.

Solid Blend	measures box	**At Point, Point to Point, String= from Point, String to Point)**

if not blank, and **Solid Colour** *is specified, the solid fill is drawn with blend value* **Solid Blend**. *The blend value is between 0.0 and 1.0.*

If blank, **Solid Blend** *is assumed to be 1.0.*

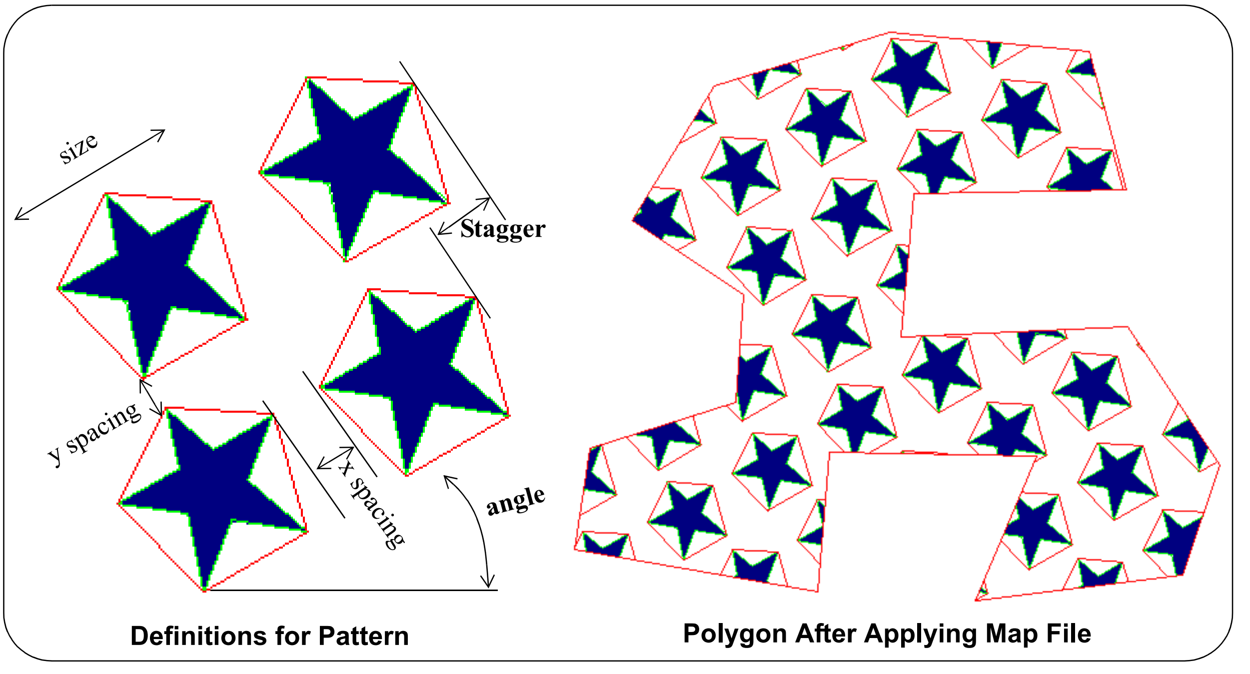
Comment and Active

See 8.8.2 Comment and Active Column in Grids.

To assist is debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

Examples of Pattern Fill

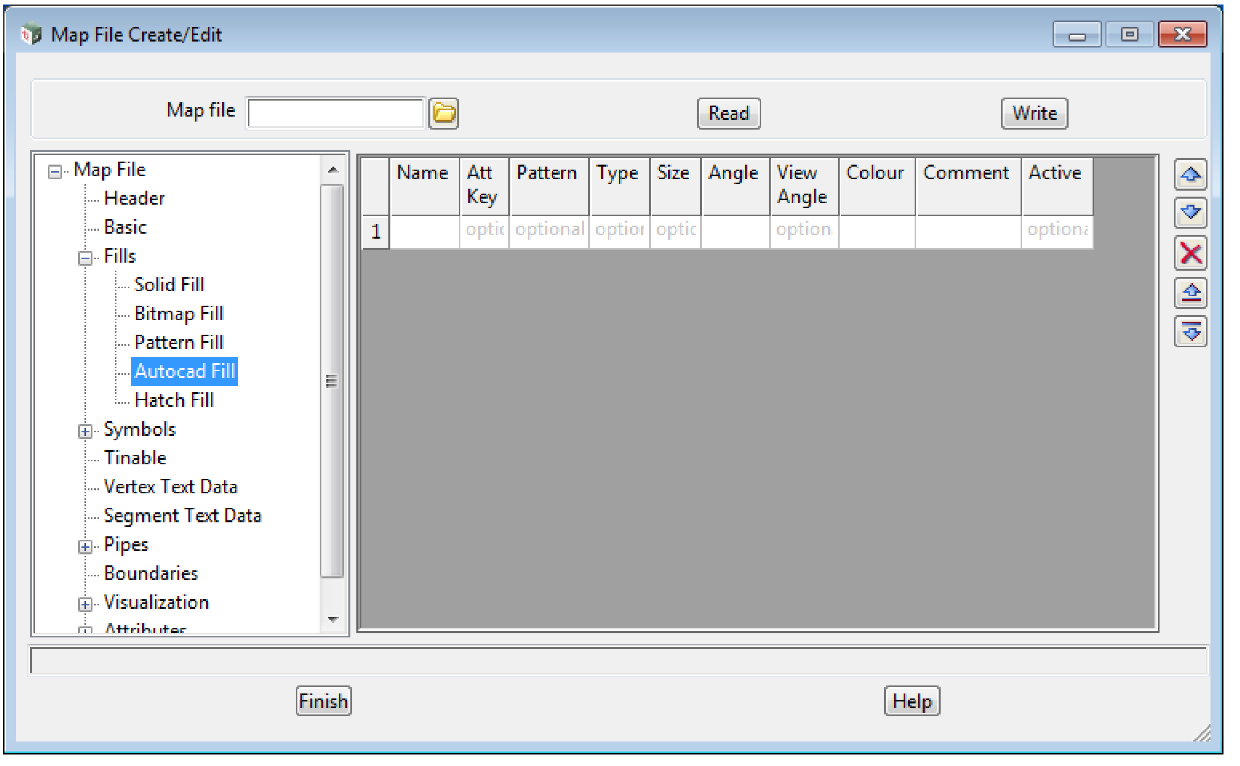
	Name	Att Key	Pattern	Type	X Origin	Y Origin	Size	Angle	View Angle	Stagger	Space X	Space Y	Solid Colour	Solid Blend	Comment
1	pf46		star with bitmap dots	world	0	0	20	30	no	6	6	4			



Go to the next section 8.8.1.3.4 Fills >Autocad Fill or return to 8.8.1.3 Fills or 8.8.1 Create/Edit a Map File.

8.8.1.3.4 Fills >Autocad Fill

AutoCad Fill uses a pattern from the **AutoCAD pattern** file to fill the selected super strings.



Processing Using Name and Att Key for Fills>AutoCAD Fill Node

when a string finds a first match with **Name** *and* **Att Key** *in the grid (see* Matching Using Name and Att Key:*), the string is given an AutoCAD Pattern Fill as defined in the parameters of the matching line.*

Name and Att Key

Name *is at Text grid. The* ***string name*** *is matched against* **Name**.
Att Key *is an Attribute Data grid cell. The* ***string attributes*** *are matched against the attribute details in* **Att Key**.
For further information on what is allowed in **Name** *and* **Att Key***, and how to access* **Att Key***, see* Name and Att Key.

Note: all the fields on the line after **Att Key** can have attribute substitution from string attributes. See 8.8.3 Map File Substitution by Attributes.

Pattern select autocad pattern

The autocad pattern name to be applied.These patterns are defined via acad.pat.

This is an autocad file.

Type

The units of size.This means the pattern can be drawn with either world size, paper size (mm) or device size.

Size

The size of the pattern in the units given by **Type.**

Angle measures box **At Point, Point to Point, String= from Point, String to Point)**

The orientation of the pattern relative to the x axis measured in the anti-clockwise direction. The units are dms (degrees minutes and seconds) and typed in using 4.17.1 HP Notation.

View Angle choice box yes, no

if not blank, this field controls whether **Angle** *is relative to the x axis or to the plotting x axis.*

If **yes** *and we are plotting,* **Angle** *is measured relative to the x axis of the plot rotation.*

If **no**, **Angle** *is measured absolute to the world x axis*

If blank, **Angle** *is measured absolute to the world x axis.*

This is an extension.

Colour

The colour of the pattern.

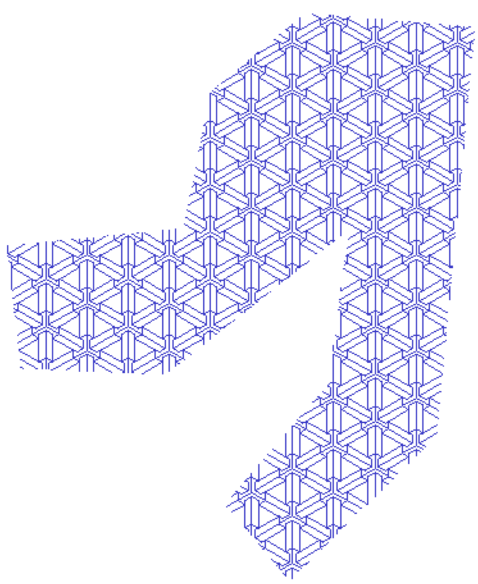
Comment and Active

See 8.8.2 Comment and Active Column in Grids.

To assist is debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

Example of AutoCAD Fill

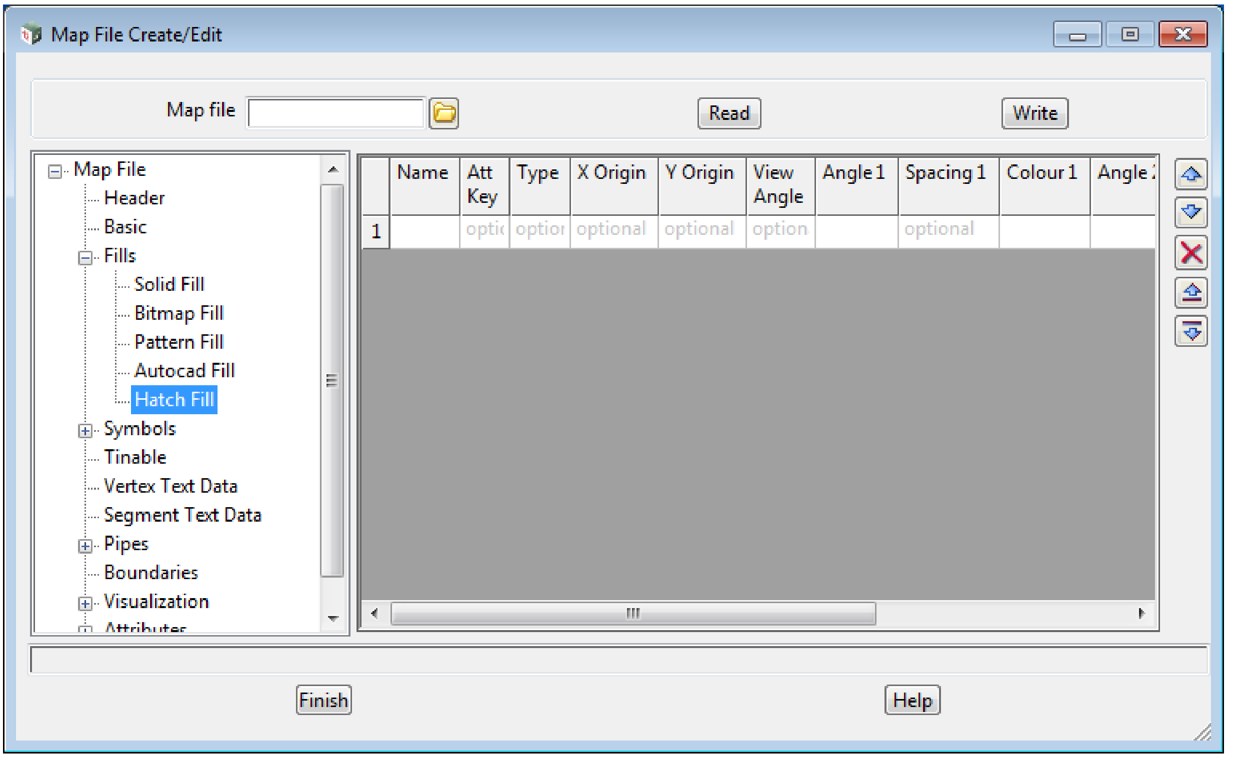
	Name	Att Key	Pattern	Type	Size	Angle	View Angle	Colour	Comment
1	pl04		ESCHER	world	10	30	no	dark blue	



Go to the next section 8.8.1.3.5 Fills >Hatch Fill or return to 8.8.1.3 Fills or 8.8.1 Create/Edit a Map File.

8.8.1.3.5 Fills >Hatch Fill

Hatch Fill sets whether super strings (closed and non-closed) are filled with a hatch pattern. The hatch pattern consists of one or two sets of lines drawn at specified angles.



Processing Using Name and Att Key for Fills>Hatch Fill Node

when a string finds a first match with **Name** *and* **Att Key** *in the grid (see* Matching Using Name and Att Key:*), the string is given a Hatch Pattern Fill as defined in the parameters of the matching line.*

Name and Att Key

Name *is at Text grid. The* ***string name*** *is matched against* **Name**.
Att Key *is an Attribute Data grid cell. The* ***string attributes*** *are matched against the attribute details in* **Att Key**.
For further information on what is allowed in **Name** *and* **Att Key***, and how to access* **Att Key***, see* Name and Att Key.

Note: all the fields on the line after **Att Key** can have attribute substitution from string attributes. See 8.8.3 Map File Substitution by Attributes.

Type select choice box

The units of **Spacing 1** *and* **Spacing 2***. This means the pattern can be drawn with either world size, paper size (mm) or device size.*

X/Y Origin

The x/y co-ordinate of the anchor point of the repeated pattern.

View Angle choice box yes, no

if not blank, this field controls whether **Angle 1** *and* **Angle 2** *are relative to the x axis or to the plotting x axis.*

If **yes** *and we are plotting,* **Angle 1** *and* **Angle 2** *are measured relative to the x axis of the plot rotation.*

If **no**, **Angle 1** *and* **Angle 2** *are measured absolute to the world x axis*

If blank, **Angle 1** *and* **Angle 2** *are measured absolute to the world x axis.*

Angle 1 measures box **At Point, Point to Point, String= from Point, String to Point)**

The orientation of the first set of lines relative to the x axis measured in the anti-clockwise direction. The units are dms (degrees minutes and seconds) and typed in using 4.17.1 HP Notation.

Spacing 1

The distance between the first set of lines (drawn at an angle **Angle 1***) in the units selected in the* **Type** *column.*

Colour 1

The colour of the first set of lines.

Angle 2 measures box **At Point, Point to Point, String= from Point, String to Point)**

The orientation of the second set of lines relative to the x axis measured in the anti-clockwise direction. The units are dms (degrees minutes and seconds) and typed in using 4.17.1 HP Notation.

Spacing 2

The distance between the second set of lines (drawn at an angle **Angle 2***) in the units selected in the* **Type** *column.*

Colour 2

The colour of the second set of lines.

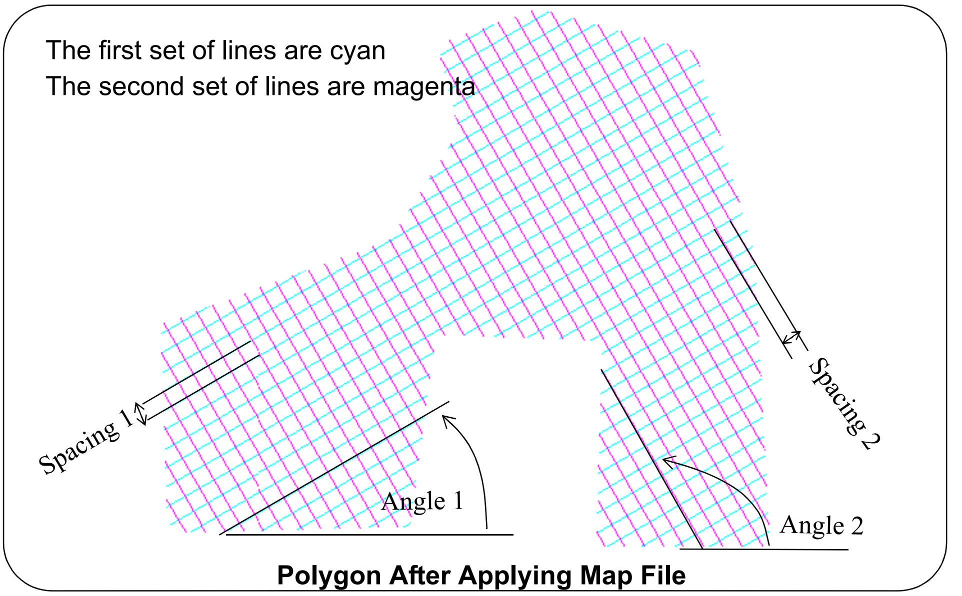
Comment and Active

See 8.8.2 Comment and Active Column in Grids.

To assist is debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

Example of Hatch Fill

	Name	Att Key	Type	X Origin	Y Origin	View Angle	Angle 1	Spacing 1	Colour 1	Angle 2	Spacing 2	Colour 2	Comment
1	ht07		world	0	0		30	10	cyan	120	10	magenta	



Polygon After Applying Map File

Go to the next section 8.8.1.3.6 Multiple Fills or return to 8.8.1.3 Fills or 8.8.1 Create/Edit a Map File.

8.8.1.3.6 Multiple Fills

Super strings support more than one fill type per string. For example, a super string can have a solid fill, and an I Acad fill at the same time.

The order of drawing the fill types is: solid, bitmap, pattern, hatch, acad.

Go to the next section 8.8.1.4 Symbols or return to 8.8.1.3 Fills or 8.8.1 Create/Edit a Map File.

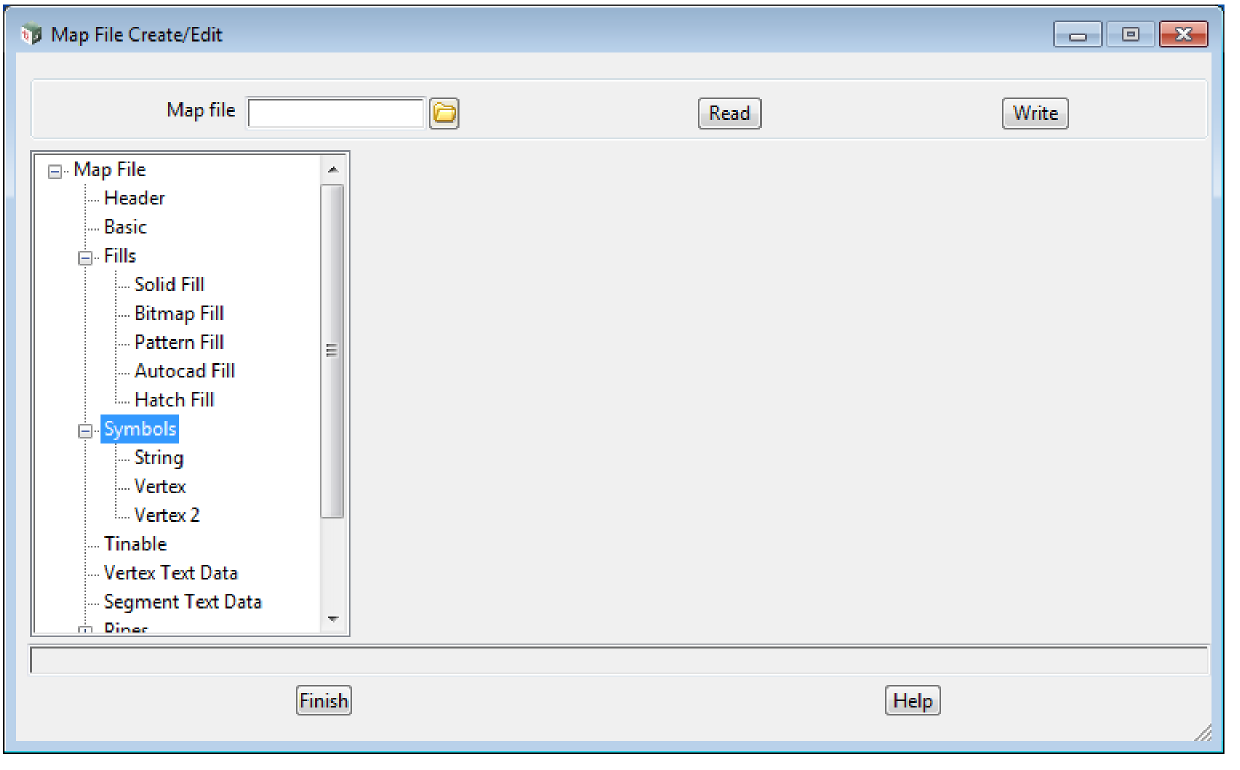
8.8.1.4 Symbols

The **Symbols** node creates symbols at the vertices of the super strings.

There can only be **one** symbol at each vertex so if ***Symbols >String*** is used to set symbols for every vertex of a string, and ***Symbols >Vertex*** applies a different symbol to any of those vertices, then the symbols from ***Symbols >Vertex*** are used at the appropriate vertices.

Symbols >Vertex 2 can be used to generate a second symbol at a vertex position BUT ***Symbols >Vertex 2*** creates a ***new*** string with a vertex and the second symbol.

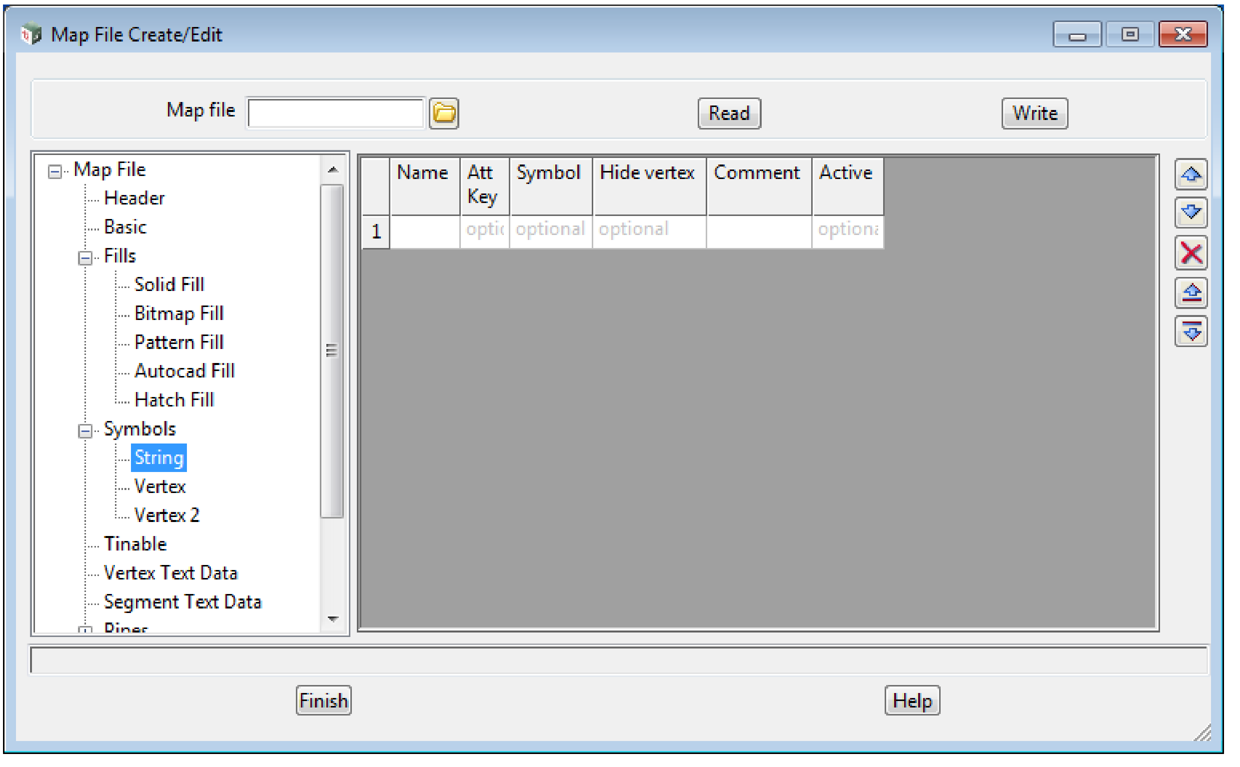
For more information on *String*, go to 8.8.1.4.1 Symbols >String
Vertex 8.8.1.4.2 Symbols >Vertex
Vertex 2 8.8.1.4.3 Symbols >Vertex 2



8.8.1.4.1 Symbols >String

String sets the same symbol at each vertex of a super string

IMPORTANT NOTE:
There can only be **one** symbol at each vertex so if ***Symbols >String*** is used to set symbols for every vertex of a string, and ***Symbols >Vertex*** applies a different symbol to any of those vertices, then the symbols from ***Symbols >Vertex*** are used at the appropriate vertices.



Processing Using Name and Att Key for Symbols>String Node

when a string finds a first match with **Name** *and* **Att Key** *in the grid (see* Matching Using Name and Att Key:*), each vertex of the string is given a Symbol as defined in the parameters of the matching line.*

Name and Att Key

Name *is at Text grid. The* ***string name*** *is matched against* **Name**.
Att Key *is an Attribute Data grid cell. The* ***string attributes*** *are matched against the attribute details in* **Att Key**.
For further information on what is allowed in **Name** *and* **Att Key**, *and how to access* **Att Key**, *see* Name and Att Key.

Symbol **select symbol** data

Symbol *is an* ***Symbol Information*** *grid cell and sets parameters for creating a symbol on the vertices of super strings. The symbol is positioned about the vertex.*
This field can not be blank.
To access ***Symbol***, *click LB on the* ***Symbol*** *field to highlight the field, then click LB again to bring up the* **Symbol Information** *panel. See* 4.7.1 Symbol Data *for information about the field to enter.*

Note: *Fields in the* **Symbol Information** *can have attribute substitution from string attributes. See* 8.8.3 Map File Substitution by Attributes.

Hide vertex **tick box**

if ticked, no default cross is placed at the vertex of the string.

Note: **Hide Vertex** *can have attribute substitution from string attributes. See* 8.8.3 Map File Substitution by Attributes.

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

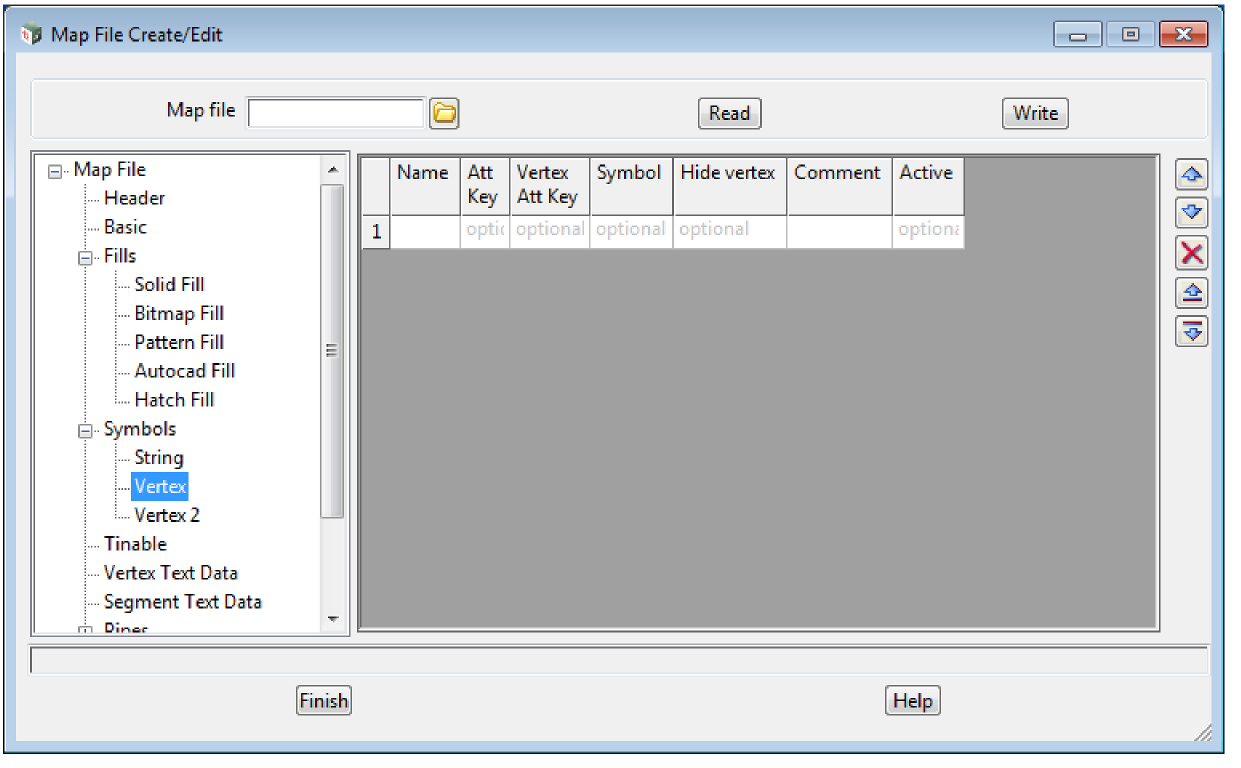
To assist is debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

Go to the next section 8.8.1.4.2 Symbols >Vertex or return to 8.8.1.4 Symbols or 8.8.1 Create/ Edit a Map File.

8.8.1.4.2 Symbols >Vertex

Vertex sets a given symbol for each vertex of a super string that matches a string name, a string attribute value and a vertex attribute value.

There can only be **one** symbol at each vertex so if ***Symbols >String*** is used to set symbols for every vertex of a string, and ***Symbols >Vertex*** applies a different symbol to any of those vertices, then the symbol from ***Symbols >Vertex*** is used at the appropriate vertices.



Processing Using Name, Att Key and Vertex Att Key

when a string finds a first match with **Name**, **Att Key** *and* **Vertex Att Key** *in the grid (see* Matching Using Name, Att Key and Vertex Att Key:*), each matching vertex of the string is given a Symbol as defined in the parameters of the matching line.*

Name, Att Key and Vertex Att Key

Name *is at Text grid and the text entered into* **Name** *can include wild cards* ******* *and wild characters* ***?****.*
The string name is matched against **Name***. This field can not be blank.*
Att Key *is an Attribute Data grid cell which contains the definition of the attributes and their values that are to be matched against. There can be more than one attribute in the Attribute Data but they must have unique names.*
The ***string*** *attributes are matched against the attribute details in* **Att Key.**
Vertex Att Key *is an Attribute Data grid cell which contains the definition of the vertex attributes and their values that are to be matched against. There can be more than one vertex attribute in the Attribute Data but they must have unique names.*
The ***vertex*** *attributes are matched against the attribute details in* **Vertex Att Key**.
To access ***Att Key*** *or* **Vertex Att Key** *data, click LB on the* ***Att Key/*** **Vertex Att Key** *field to highlight the field, then click LB again to bring up the* **Attribute Data** *panel. To enter data, see* Attribute Data Panel.

Matching Using Name, Att Key and Vertex Att Key:

Strings are first created via the ***Basic*** *node and given a string name of either the Entity Name or the text given in the Name column of the Basic Map File grid.*
Starting with the first line of the grid, matching and processing occurs as follows

If **Name, Att Key** *and* **Vertex Att Key** *are not blank, and a match of the string name occurs with* **Name**, *a match of the string attributes occurs with* **Att Key** *and a match of the string vertex attributes occurs with* **Vertex Att Key**, *then the rest of the fields for this line of the Map File grid are used on this string vertex.*
If **Name** *and* **Att Key** *are not blank, and* **Vertex Att Key** *is blank, and a match of the string name occurs with* **Name** *and a match of the string attributes occurs with* **Att Key**, *then the rest of the fields for this line of the Map File grid are used on each string vertex.*
If **Name** *and* **Vertex Att Key** *are not blank, and* **Att Key** *is blank, and a match of the string name occurs with* **Name** *and a match of a string vertex attribute occurs with* **Vertex Att Key**, *then the rest of the fields for this line of the Map File grid are used on this string vertex.*

If **Name** *in not blank and* **Att Key** *and* **Vertex Att Key** *are blank, and a match of the string name occurs with* **Name**, *then the rest of the fields for this line of the Map File grid are used on each string vertex.*

If **Name** *in blank then no match occurs and this line of the Map File grid is ignored.*

If a ***match occurs****, then no tests for matches against* **Name**, **Att Key** *and* **Vertex Att Key** *further down in the grid are made.*

If ***no match occurs****, then this line of the map file grid is ignored and a test for a match is made against the next line of the grid.*

Symbol **select symbol** data

Symbol *is Symbol Information grid cell and sets parameters for creating a symbol on the vertices of super strings. The symbol is positioned about the vertex.*
This field can not be blank.
To access ***Symbol****, click LB on the* ***Symbol*** *field to highlight the field, then click LB again to bring up the* **Symbol Information** *panel.*

Note*: Fields in the* **Symbol Information** *can have attribute substitution from vertex attributes. See* 8.8.3 Map File Substitution by Attributes.

Hide vertex tick box

if ticked, no default cross is placed at the vertex of the string.

Note*:* **Hide Vertex** *can have attribute substitution from string attributes. See* 8.8.3 Map File Substitution by Attributes.

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

To assist is debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

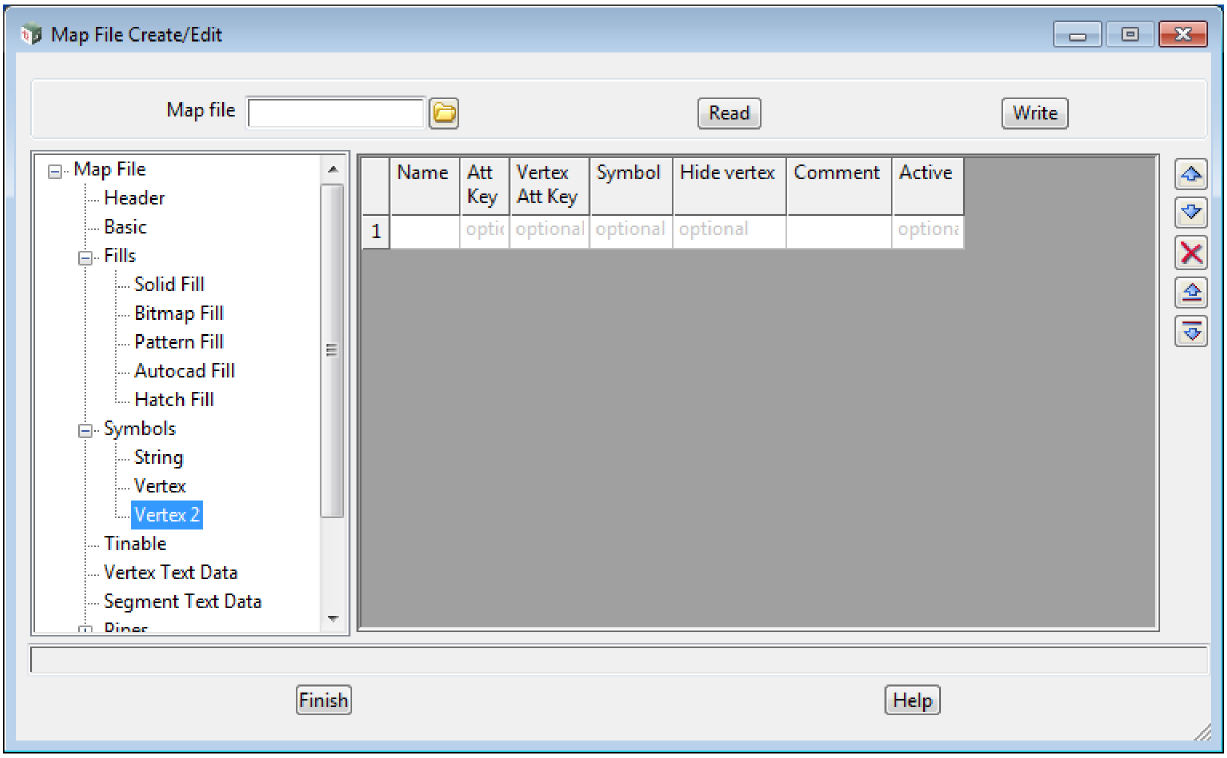
Go to the next section 8.8.1.4.3 Symbols >Vertex 2 or return to 8.8.1.4 Symbols or 8.8.1 Create/Edit a Map File.

8.8.1.4.3 Symbols >Vertex 2

Vertex 2 creates a ***new*** string with vertices and symbols at those super strings that match a string name, a string attribute value and a vertex attribute value in the ***Symbols >Vertex 2*** grid.

IMPORTANT RESTRICTION:

Symbols >Vertex 2 does not work for **Utilities =>H-Z =>Map.**



Processing Using Name, Att Key and Vertex Att Key

when a string finds a first match with **Name**, **Att Key** *and* **Vertex Att Key** *in the grid (see* Matching Using Name, Att Key and Vertex Att Key:*), each matching vertex of the string is given a Symbol as defined in the parameters of the matching line.*

Name, Att Key and Vertex Att Key

Name *is at Text grid. The* ***string name*** *is matched against* **Name**.
Att Key *is an Attribute Data grid cell. The* ***string attributes*** *are matched against the attribute details in* **Att Key**.
Vertex Att Key *is an Attribute Data grid cell. The* ***vertex attributes*** *are matched against the attribute details in* **Att Key**.
For further information on what is allowed in **Name** *and* **Att Key**, *and how to access* **Att Key**, *see* Name, Att Key and Vertex Att Key.

Symbol **select symbol** data

Symbol *is Symbol Information grid cell and sets parameters for creating a symbol on the vertices of super strings. The symbol is positioned about the vertex.*
This field can not be blank.
To access ***Symbol***, *click LB on the* ***Symbol*** *field to highlight the field, then click LB again to bring up the* **Symbol Information** *panel.*

Note*: Fields in the* **Symbol Information** *can have attribute substitution from vertex attributes. See* 8.8.3 Map File Substitution by Attributes.

Hide vertex **tick box**

if ticked, no default cross is placed at the vertex of the string.

***Note**: **Hide Vertex** can have attribute substitution from string attributes. See* 8.8.3 Map File Substitution by Attributes.

Comment and Active

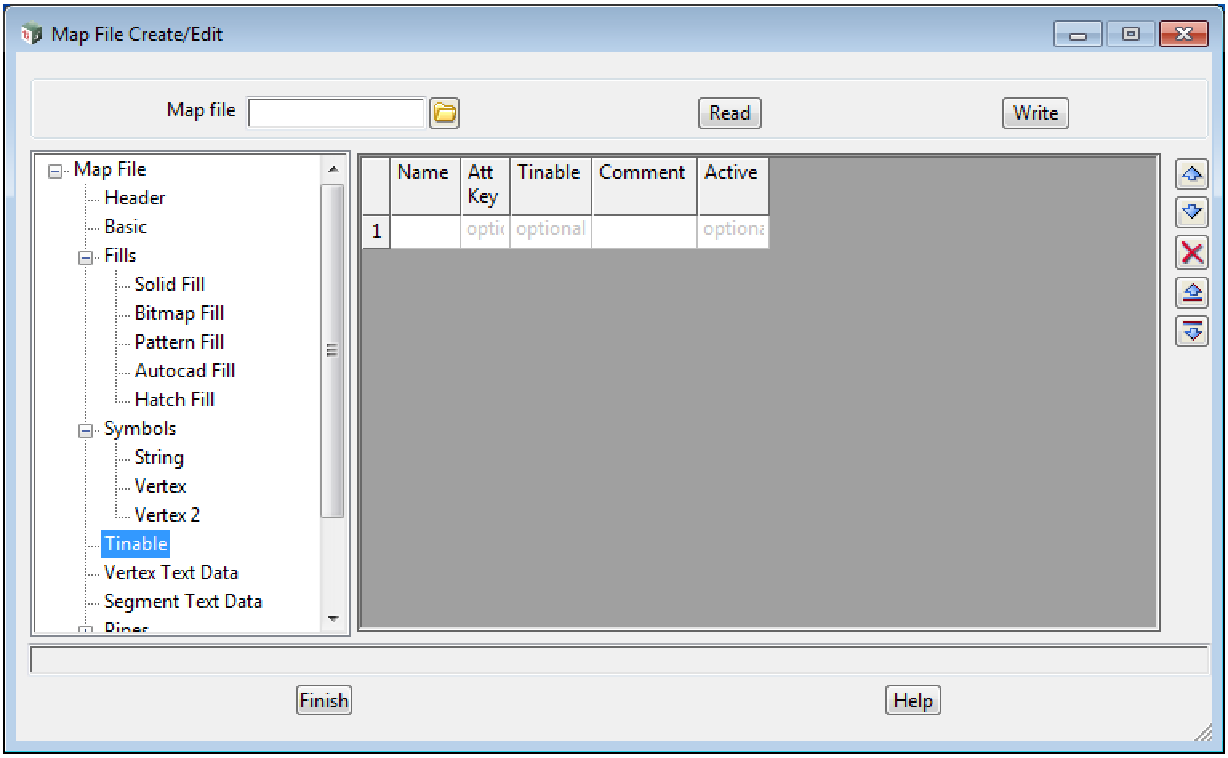
See 8.8.2 Comment and Active Column in Grids.

To assist is debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

Go to the next section 8.8.1.5 Tinable or return to 8.8.1.4 Symbols or 8.8.1 Create/Edit a Map File.

8.8.1.5 Tinable

Tinable sets whether the vertices and segments are tinable (used in triangulations), not tinable (not used in triangulations) or only the vertices (points) are tinable.



Processing Using Name and Att Key for Tinable Node

when a string finds a first match with **Name** *and* **Att Key** *in the grid (see* Matching Using Name and Att Key:*), each vertex and segment of the string is given a tinability value as defined in the parameters of the matching line.*

Name and Att Key

Name *is at Text grid. The* ***string name*** *is matched against* **Name**.
Att Key *is an Attribute Data grid cell. The* ***string attributes*** *are matched against the attribute details in* **Att Key**.
For further information on what is allowed in **Name** *and* **Att Key***, and how to access* **Att Key***, see* Name and Att Key.

Tinable **yes, no, points**

if ***yes****, the vertices and segments for all the strings matching the* **Name** *and* **Att Key** *are tinable.*
If ***no****, the vertices and segments for all the strings matching the* **Name** *and* **Att Key** *are not tinable.*
If ***points****, the segments are not tinable and the vertices are tinable for all the strings matching the* **Name** *and* **Att Key**.
This field can not be blank.

Note*:* **Tinable** *can have attribute substitution from string attributes. See* 8.8.3 Map File Substitution by Attributes.

Comment and Active

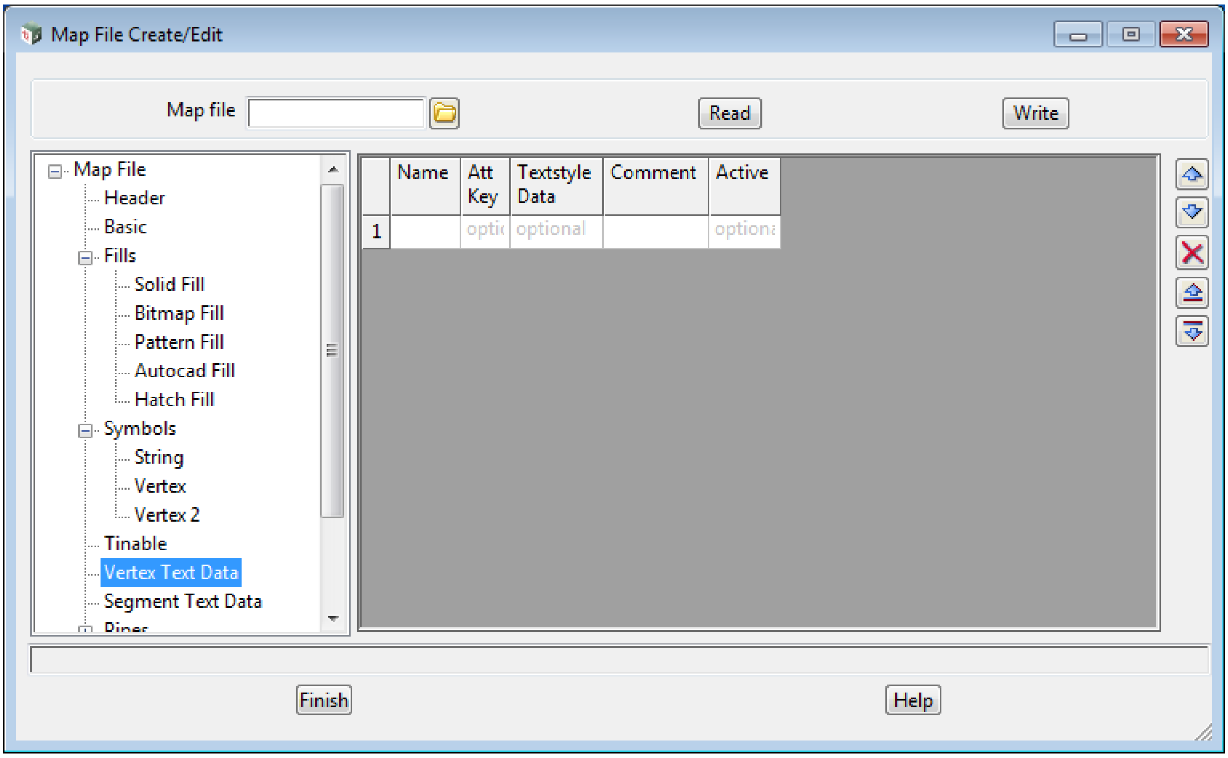
See 8.8.2 Comment and Active Column in Grids.

To assist is debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

Go to the next section 8.8.1.6 Vertex Text Data or return to 8.8.1 Create/Edit a Map File.

8.8.1.6 Vertex Text Data

Vertex Text Data sets parameters for displaying vertex text on super strings.



Processing Using Name and Att Key

when a string finds a first match with **Name** *and* **Att Key** *in the grid (see* Matching Using Name and Att Key:*), a Textstyle Data for use with vertex text on the string is given in the parameters of the matching line.*

Name and Att Key

Name *is at Text grid. The* ***string name*** *is matched against* **Name**.
Att Key *is an Attribute Data grid cell. The* ***string attributes*** *are matched against the attribute details in* **Att Key**.
For further information on what is allowed in **Name** *and* **Att Key***, and how to access* **Att Key***, see* Name and Att Key.

Textstyle Data **select textstyle data** menu

Textstyle Data *is a Textstyle Data grid cell. The Textstyle Data sets text parameters (see* 4.6 Text Definitions*) for displaying vertex text on all strings matching* **Name** *and* **Att Key**.
To access the **Textstyle Data***, click LB on the* **Textstyle Data** *field to highlight the field, then click LB again to bring up the* **Textstyle Data** *panel.*

Note*: Fields in the* **Textstyle Data** *can have attribute substitution from string attributes. See* 8.8.3 Map File Substitution by Attributes.

Comment and Active

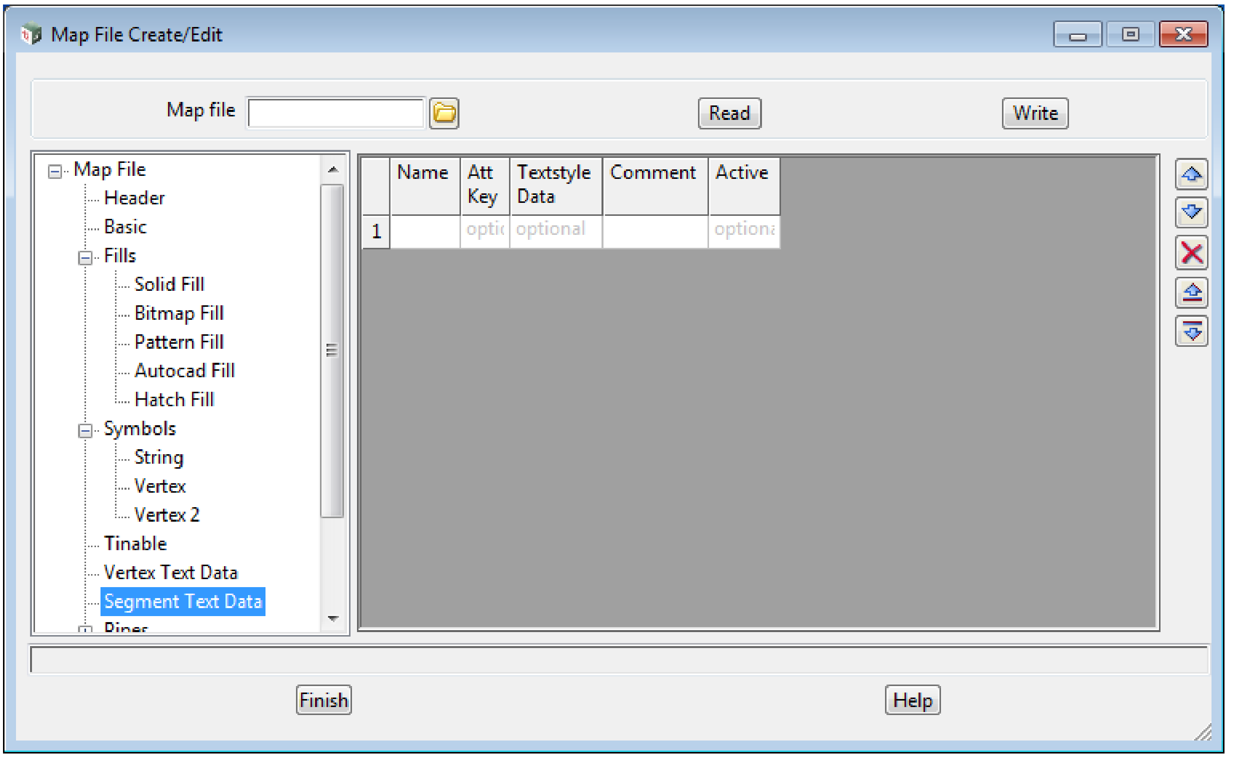
See 8.8.2 Comment and Active Column in Grids.

To assist in debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

Go to the next section 8.8.1.7 Segment Text Data or return to 8.8.1 Create/Edit a Map File.

8.8.1.7 Segment Text Data

Segment Text Data sets parameters for displaying segment text on super strings.



Processing Using Name and Att Key

when a string finds a first match with **Name** *and* **Att Key** *in the grid (see* Matching Using Name and Att Key:*), a Textstyle Data for use with segment text on the string is given in the parameters of the matching line.*

Name and Att Key

Name *is at Text grid. The* ***string name*** *is matched against* **Name**.
Att Key *is an Attribute Data grid cell. The* ***string attributes*** *are matched against the attribute details in* **Att Key**.
For further information on what is allowed in **Name** *and* **Att Key***, and how to access* **Att Key***, see* Name and Att Key.

Textstyle Data **select textstyle data** menu

Textstyle Data *is a Textstyle Data grid cell. The Textstyle Data sets text parameters (see* 4.6 Text Definitions*) for displaying segment text on all strings matching* **Name** *and* **Att Key**.
To access the **Textstyle Data***, click LB on the* **Textstyle Data** *field to highlight the field, then click LB again to bring up the* **Textstyle Data** *panel.*

Note*: Fields in the* **Textstyle Data** *can have attribute substitution from string attributes. See* 8.8.3 Map File Substitution by Attributes.

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

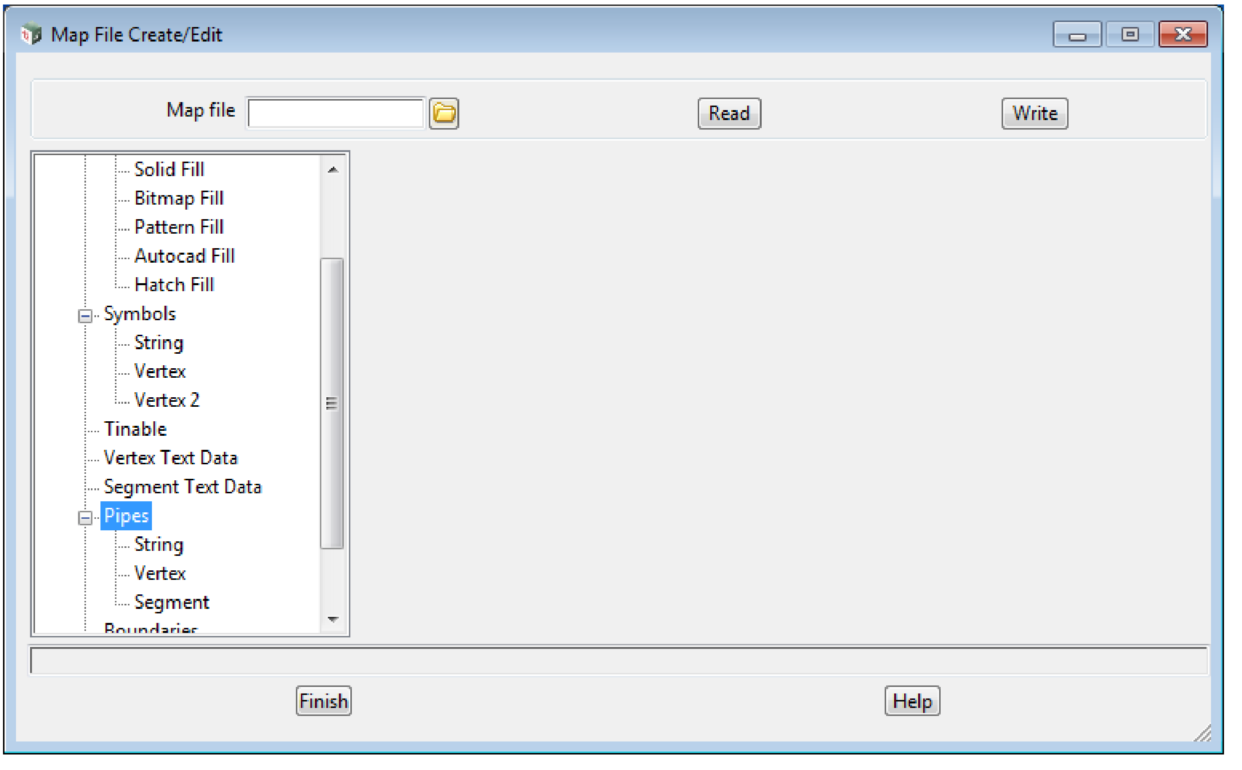
To assist in debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

Go to the next section 8.8.1.8 Pipes or return to 8.8.1 Create/Edit a Map File.

8.8.1.8 Pipes

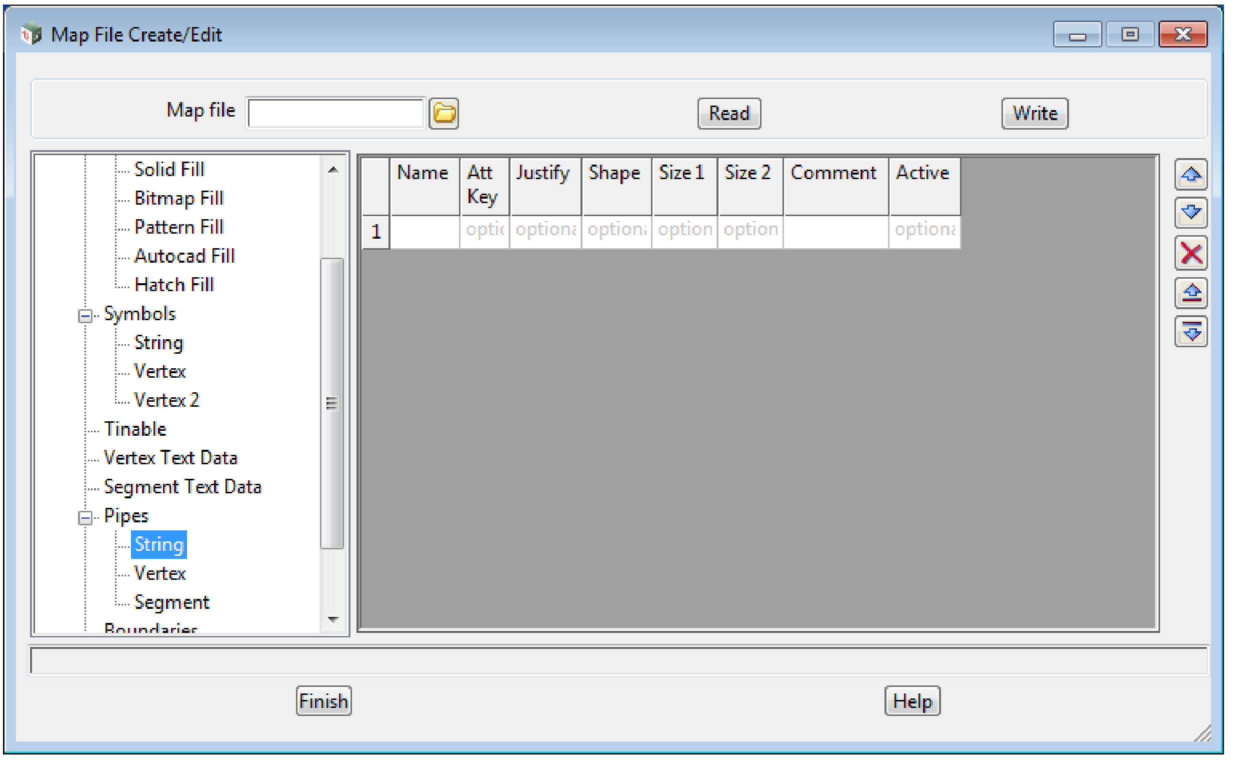
The **Pipes** node turns the string into a super pipe and sets parameters for pipe justification, whether it is round pipe and its diameter, or a rectangular pipe (culvert) and its width and height.

For information on *String*, go to 8.8.1.8.1 Pipes >String
Vertex 8.8.1.8.2 Pipes >Vertex
Segment 8.8.1.8.3 Pipes >Segment



8.8.1.8.1 Pipes >String

Strings matches on string name and string attributes to make the selected string a pipe string with either a diameter (round pipe) or a width and height (rectangular pipe), and a justification for the z-values of the string.



Processing Using Name and Att Key for Pipes>String Node

when a string finds a first match with **Name** *and* **Att Key** *in the grid (see* Matching Using Name and Att Key:*), the super string is turned into a super pipe using the parameters of the matching grid line.*

Name and Att Key

Name *is at Text grid. The* ***string name*** *is matched against* **Name**.
Att Key *is an Attribute Data grid cell. The* ***string attributes*** *are matched against the attribute details in* **Att Key**.
For further information on what is allowed in **Name** *and* **Att Key***, and how to access* **Att Key***, see* Name and Att Key.

Note: all the fields on the line after **Att Key** can have attribute substitution from string attributes. See 8.8.3 Map File Substitution by Attributes.

Justify invert, centre, obvert

the justification for the pipe string.
This field can not be blank.

Shape diameter, culvert

sets whether the pipe is a round pipe or a rectangular pipe (culvert).
If **diameter***, the string is made a round pipe.*
If **culvert***, the string is made a rectangular pipe.*
This field can not be blank.

Size 1 **measures** menu

for **Shape** *choice* **diameter***, the diameter of all the segments of the pipe are set to the value in* **Size 1**.

For **Shape** *choice* **culvert***, the width of all the segments of the pipe are set to the value in* **Size 1***.*
This field can not be blank.

Size 2 **measures** menu

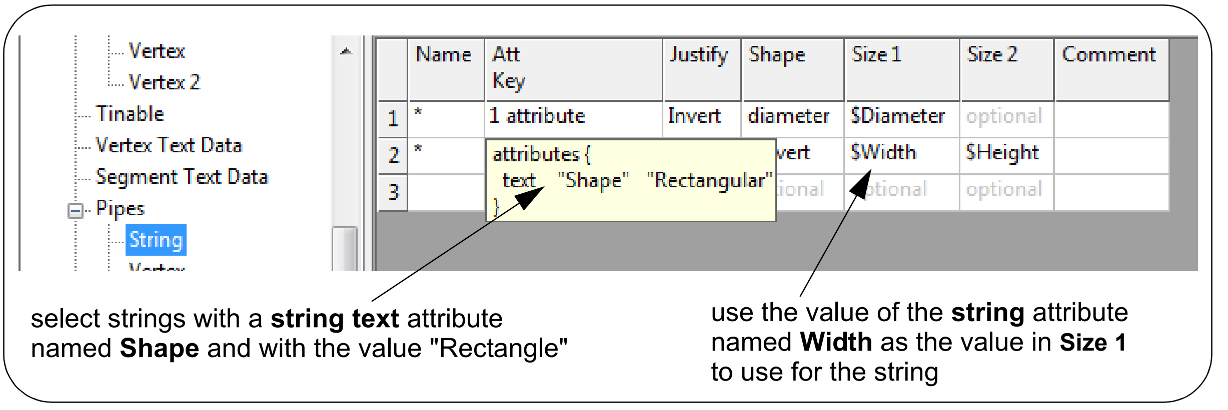
for **Shape** *choice* **culvert***, the height of all the segments of the pipe are set to the value in* **Size 2***.*
for **Shape** *choice* **diameter***, this field is ignored.*
This field can only be blank for a **Shape** *choice* **diameter***.*

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

To assist in debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

Example of Pipes >String

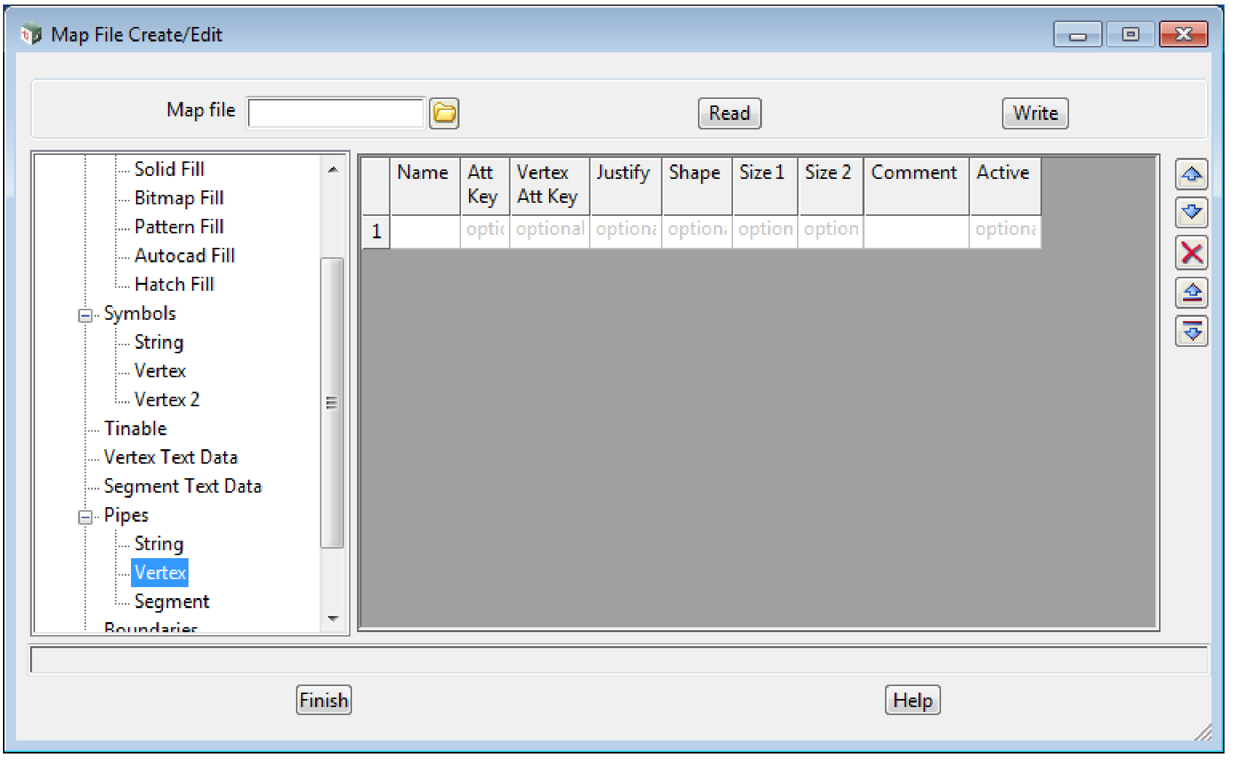


Go to the next section 8.8.1.8.2 Pipes >Vertex or return to 8.8.1.8 Pipes or 8.8.1 Create/Edit a Map File.

8.8.1.8.2 Pipes >Vertex

Vertex sets parameters for the sizing and justification of the **segment starting** with the selected vertex of the super string.

Note: All the segments of a super pipe string must have the same Shape. That is, all the segments must be **Shape *diameter*** (round pipe segment) or all the segments must be **Shape *culvert*** (rectangular pipe segment). And **Shape** must be the same as any **Shape** set for the string in 8.8.1.8.1 Pipes >String.



Processing Using Name, Att Key and Vertex Att Key

when a string finds a first match with **Name**, **Att Key** *and* **Vertex Att Key** *in the grid (see* Matching Using Name, Att Key and Vertex Att Key:*), the segment that starts with the matching vertex of the string, is given the pipe justification, pipe shape and pipe size as defined in the parameters of the matching line.*

Name, Att Key and Vertex Att Key

Name *is at Text grid. The* ***string name*** *is matched against* **Name**.
Att Key *is an Attribute Data grid cell. The* ***string attributes*** *are matched against the attribute details in* **Att Key**.
Vertex Att Key *is an Attribute Data grid cell. The* ***vertex attributes*** *are matched against the attribute details in* **Att Key**.
For further information on what is allowed in **Name** *and* **Att Key**, *and how to access* **Att Key**, *see* Name, Att Key and Vertex Att Key.

Note: all the fields on the line after **Vertex Att Key** can have attribute substitution from vertex attributes. See 8.8.3 Map File Substitution by Attributes.

Justify invert, centre, obvert

the justification for this segment of the pipe string.
This field can not be blank.

Shape diameter, culvert

sets whether this segment of the pipe is a round pipe or a rectangular pipe (culvert).
If **diameter***, the segment is made a round pipe.*
If **culvert***, the segment is made a rectangular pipe.*
This field can not be blank.

Note*: the shape can not be different from a shape set by Pipes >String*

Size 1 **measures** menu

for **Shape** *choice* **diameter***, the diameter of this segments of the pipe is set to the value in* **Size 1**.
For **Shape** *choice* **culvert***, the width of this segments of the pipe is set to the value in* **Size 1**.
This field can not be blank.

Size 2 **measures** menu

for **Shape** *choice* **culvert***, the height of this segment of the pipe is set to the value in* **Size 2**.
for **Shape** *choice* **diameter***, this field is ignored.*
This field can only be blank for a **Shape** *choice* **diameter**.

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

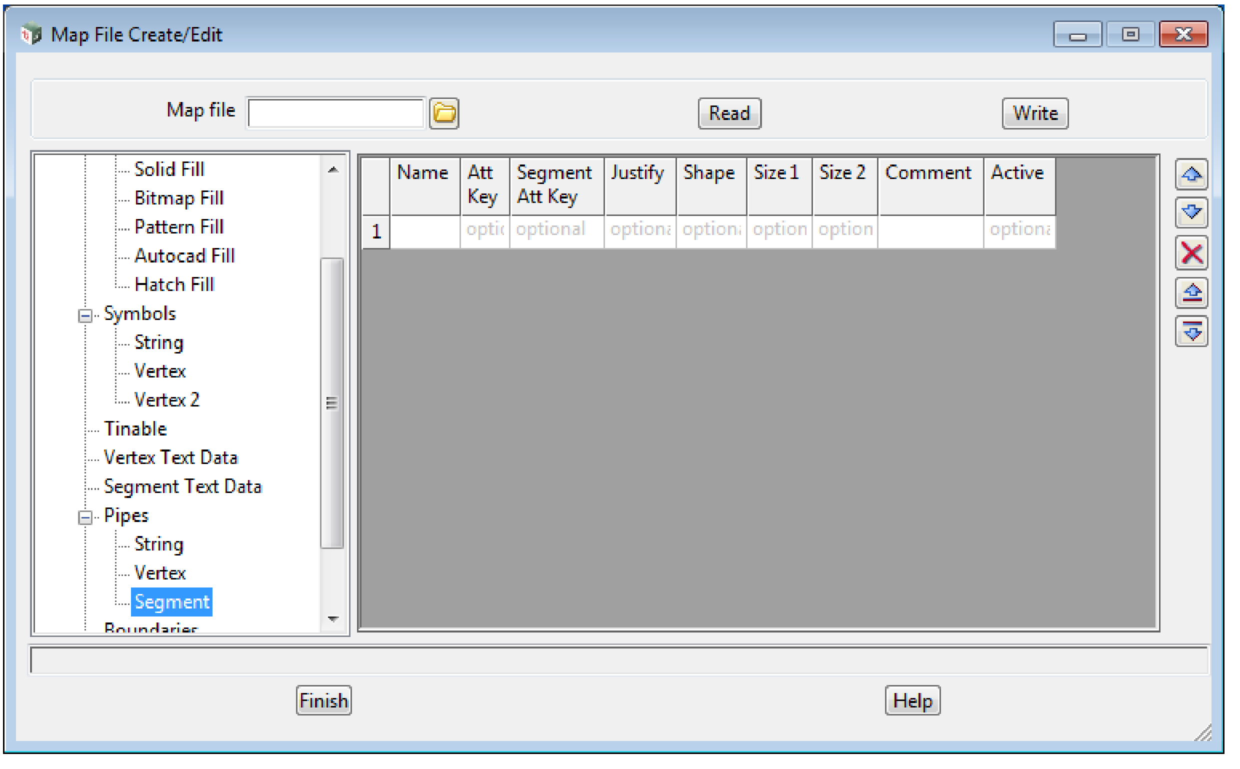
To assist in debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

Go to the next section 8.8.1.8.3 Pipes >Segment or return to 8.8.1.8 Pipes or 8.8.1 Create/Edit a Map File.

8.8.1.8.3 Pipes >Segment

Segment sets parameters for the sizing and justification of the selected segment of the super string.

Note: All the segments of a super pipe string must have the same Shape. That is, all the segments must be **Shape** ***diameter*** (round pipe segment) or all the segments must **Shape** ***culvert*** (rectangular pipe segment). And **Shape** must be the same as any **Shape** set for the string in 8.8.1.8.1 Pipes >String.



Processing Using Name, Att Key and Segment Att Key

when a string finds a first match with **Name**, **Att Key** *and* **Segment Att Key** *in the grid (see* Matching Using Name, Att Key and Segment Att Key:*), each matching segment of the string is the pipe, justification, pipe shape and pipe size as defined in the parameters of the matching line.*

Name, Att Key and Segment Att Key

Name *is at Text grid and the text entered into* **Name** *can include wild cards* *** *and wild characters* **?**. *The string name is matched against* **Name**. *This field can not be blank.*
Att Key *is an Attribute Data grid cell which contains the definition of the attributes and their values that are to be matched against. There can be more than one attribute in the Attribute Data but they must have unique names.*
The ***string*** *attributes are matched against the attribute details in* **Att Key.**
Segment Att Key *is an Attribute Data grid cell which contains the definition of the segment attributes and their values that are to be matched against. There can be more than one segment attribute in the Attribute Data but they must have unique names.*
The ***segment*** *attributes are matched against the attribute details in* **Segment Att Key**.
To access ***Att Key*** *or* **Segment Att Key** *data, click LB on the* ***Att Key/*** **Segment Att Key** *field to highlight the field, then click LB again to bring up the* **Attribute Data** *panel. To enter data, see* Attribute Data Panel.

Matching Using Name, Att Key and Segment Att Key:

Strings are first created via the ***Basic*** *node and given a string name of either the Entity Name or the text given in the Name column of the Basic Map File grid.*

Starting with the first line of the grid, matching and processing occurs as follows
If **Name, Att Key** *and* **Segment Att Key** *are not blank, and a match of the string name occurs with* **Name**, *a match of the string attributes occurs with* **Att Key** *and a match of the string segment attributes occurs with* **Segment Att Key**, *then the rest of the fields for this line of the Map File grid are used on this string segment.*
If **Name** *and* **Att Key** *are not blank, and* **Segment Att Key** *is blank, and a match of the string name occurs with* **Name** *and a match of the string attributes occurs with* **Att Key**, *then the rest of the fields for this line of the Map File grid are used on each string segment.*
If **Name** *and* **Segment Att Key** *are not blank, and* **Att Key** *is blank, and a match of the string name occurs with* **Name** *and a match of a string segment attribute occurs with* **Segment Att Key**, *then the rest of the fields for this line of the Map File grid are used on this string segment.*
If **Name** *in not blank and* **Att Key** *and* **Segment Att Key** *are blank, and a match of the string name occurs with* **Name**, *then the rest of the fields for this line of the Map File grid are used on each string segment.*

If **Name** *in blank then no match occurs and this line of the Map File grid is ignored.*

If a ***match occurs***, *then no tests for matches against* **Name**, **Att Key** *and* **Segment Att Key** *further down in the grid are made.*

If ***no match occurs***, *then this line of the map file grid is ignored and a test for a match is made against the next line of the grid.*

Note: all the fields on the line after **Segment Att Key** can have attribute substitution from segment attributes. See 8.8.3 Map File Substitution by Attributes.

Justify invert, centre, obvert

the justification for this segment of the pipe string.
This field can not be blank.

Shape diameter, culvert

sets whether this segment of the pipe is a round pipe or a rectangular pipe (culvert).

If **diameter**, *the segment is made a round pipe.*

If **culvert**, *the segment is made a rectangular pipe.*

This field can not be blank.

Note: *the shape can not be different from a shape set by Pipes >String*

Size 1 **measures** menu

for **Shape** *choice* **diameter**, *the diameter of this segments of the pipe is set to the value in* **Size 1**.
For **Shape** *choice* **culvert**, *the width of this segments of the pipe is set to the value in* **Size 1**.
This field can not be blank.

Size 2 **measures** menu

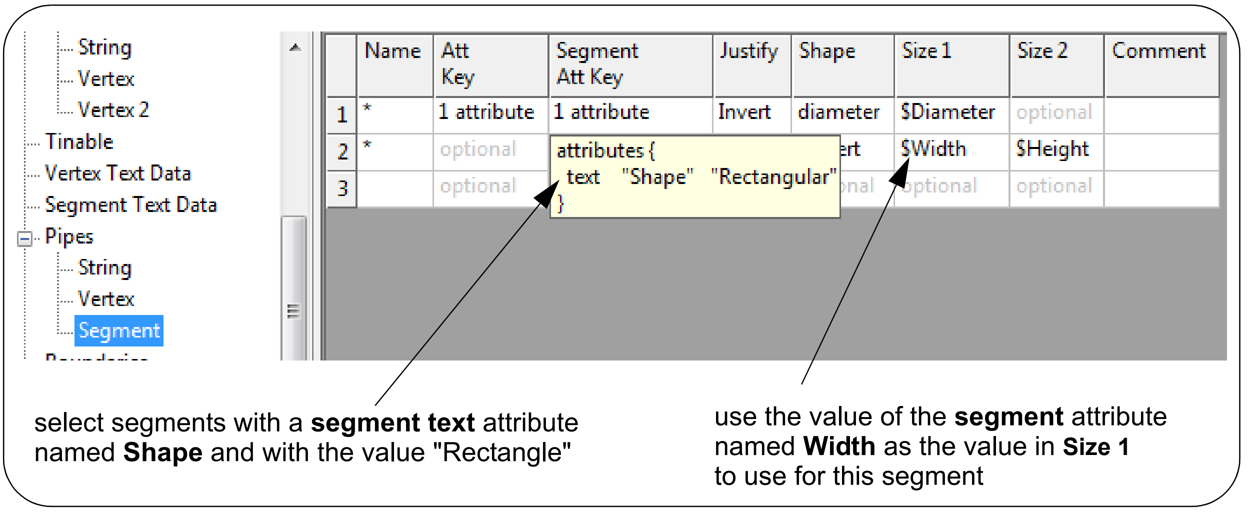
for **Shape** *choice* **culvert**, *the height of this segment of the pipe is set to the value in* **Size 2**.
for **Shape** *choice* **diameter**, *this field is ignored.*
This field can only be blank for a **Shape** *choice* **diameter**.

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

To assist in debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

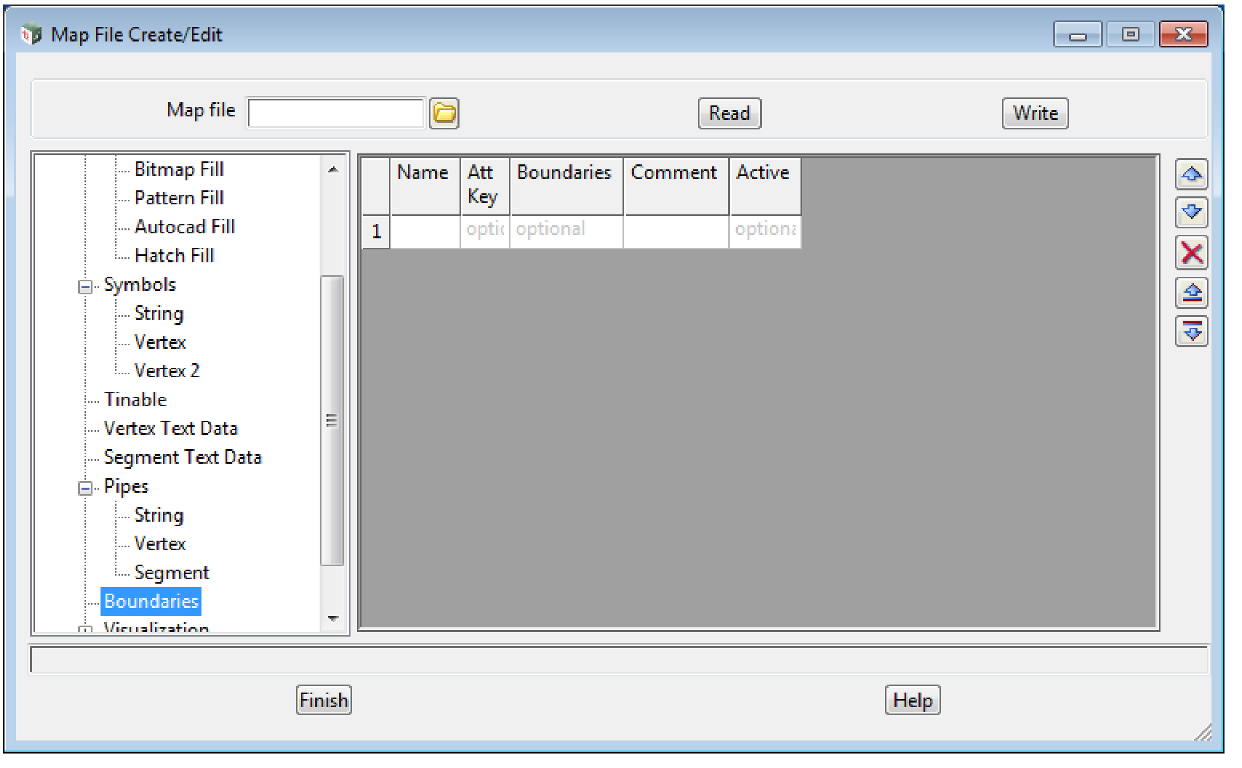
Example of Pipes >Segment



Go to the next section 8.8.1.9 Boundaries or return to 8.8.1.8 Pipes or 8.8.1 Create/Edit a Map File.

8.8.1.9 Boundaries

Boundaries sets an attribute for the string so that when it is used for Nulling of triangles in a Tin, the triangles inside the string will either be set to null (exclude), or set back to not null (include).



Processing Using Name and Att Key

when a string finds a first match with **Name** *and* **Att Key** *in the grid (see* Matching Using Name and Att Key:*), the string is given a Boundaries value as defined in the parameters of the matching line.*

Name and Att Key

Name *is at Text grid. The* ***string name*** *is matched against* **Name**.
Att Key *is an Attribute Data grid cell. The* ***string attributes*** *are matched against the attribute details in* **Att Key**.
For further information on what is allowed in **Name** *and* **Att Key**, *and how to access* **Att Key**, *see* Name and Att Key.

Boundaries exclude, include

If ***exclude****, then any triangles inside the string are set to null.*

If ***include****, then any triangles inside the string are reset so that they aren't null.*

This field can not be blank.

This field can not be blank.

Note: **Boundaries** *can have attribute substitution from segment attributes. See* 8.8.3 Map File Substitution by Attributes.

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

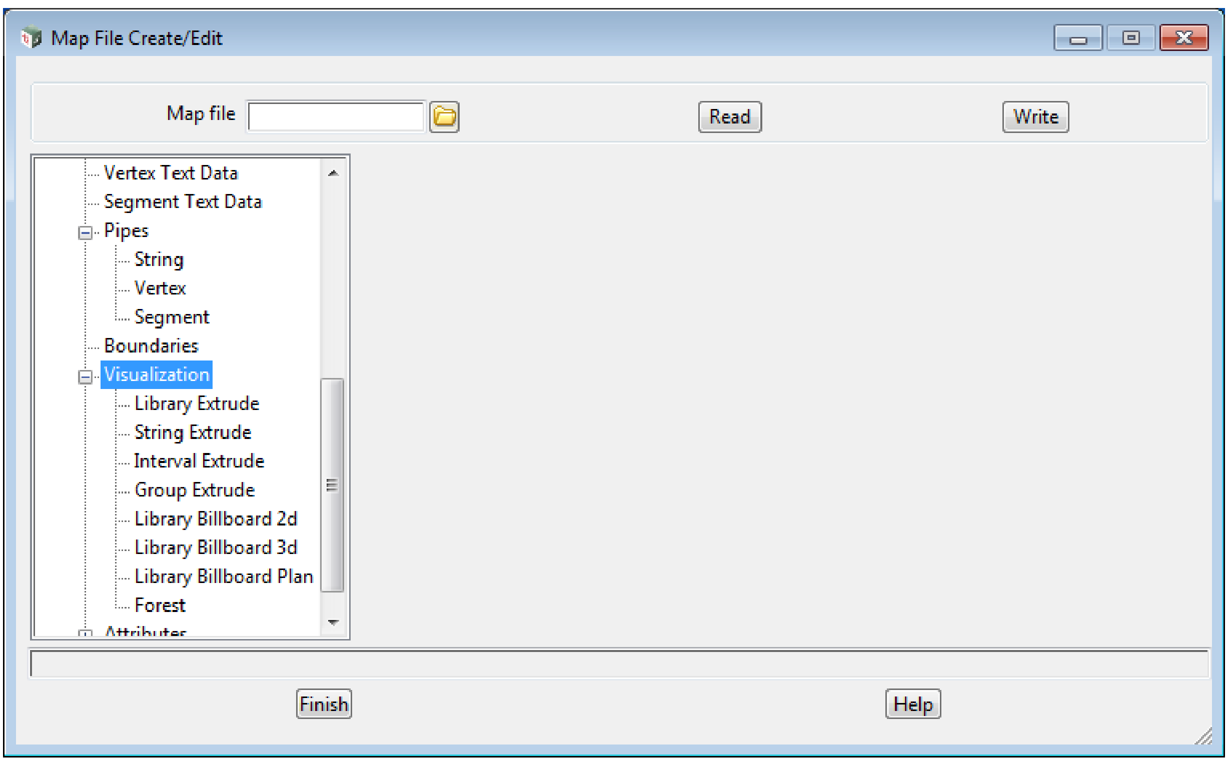
To assist in debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

Go to the next section 8.8.1.10 Visualization or return to 8.8.1 Create/Edit a Map File.

8.8.1.10 Visualization

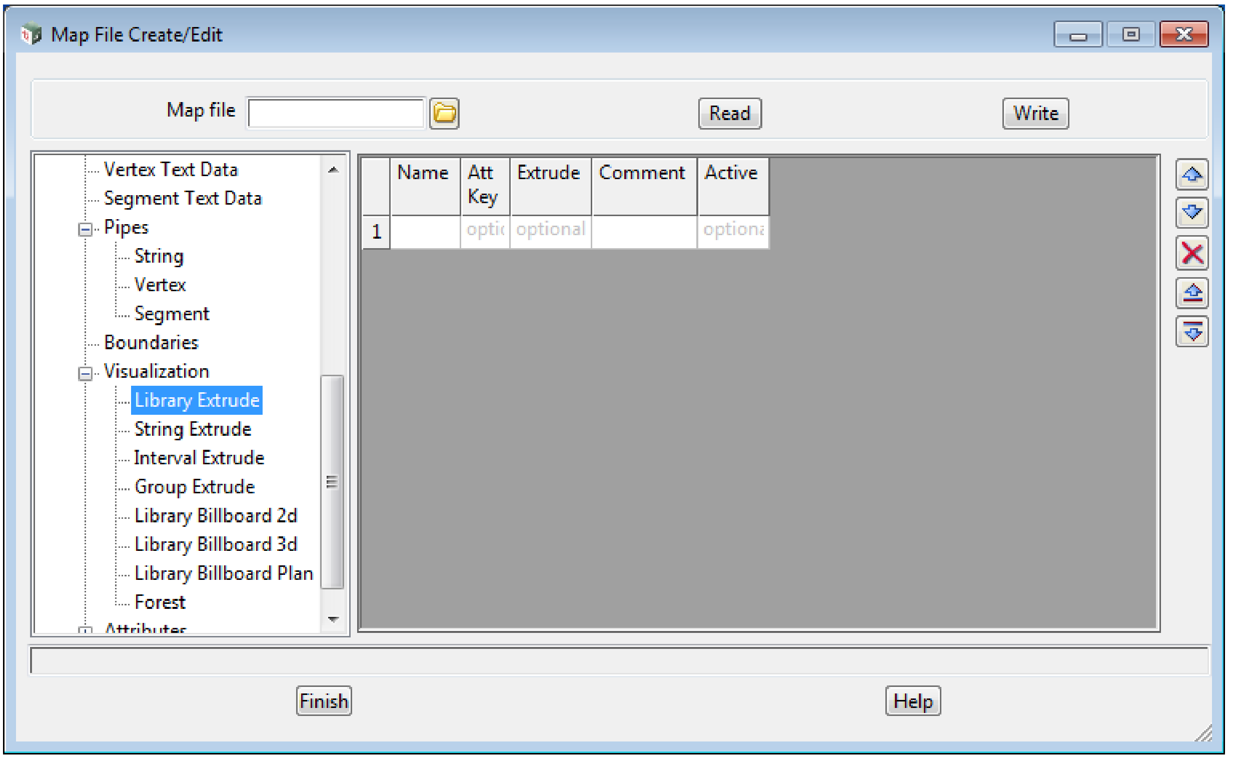
Note that when any Visualization mapping is applied, it is additive. This means that these mappings are added to any existing visualization element of the super string. This allows for a number of extrusions or billboards to be applied on a mapping. It does have the unexpected result that if you run Map File **Apply** more than once, multiple instances of Visualization mappings can end up on a same string. For more information on extrusions, see 12.13.6 Extrusions.

For Visualisation >*Library Extrude*, go to 8.8.1.10.1 Visualisation >Library Extrude
String Extrude 8.8.1.10.2 Visualisation >String Extrude
Interval Extrude 8.8.1.10.3 Visualisation >Interval Extrude
Group Extrude 8.8.1.10.4 Visualisation >Group Extrude
Library Billboard 2d 8.8.1.10.5 Visualisation >Library Billboard 2d
Library Billboard 3d 8.8.1.10.6 Visualisation >Library Billboard 3d
Library Billboard Plan 8.8.1.10.7 Visualisation >Library Billboard Plan
Forest 8.8.1.10.8 Visualisation >Forest



8.8.1.10.1 Visualisation >Library Extrude

Library Extrude applies an extrude from the 12d library to selected super strings. Note that extrudes can only be applied to super strings.



Processing Using Name and Att Key

when a string finds a first match with **Name** *and* **Att Key** *in the grid (see* [Matching Using Name and Att Key:](#)*), the string is given the Library Extrude as defined in the parameter of the matching line.*

Name and Att Key

Name *is at Text grid. The* ***string name*** *is matched against* **Name**.
Att Key *is an Attribute Data grid cell. The* ***string attributes*** *are matched against the attribute details in* **Att Key**.
For further information on what is allowed in **Name** *and* **Att Key***, and how to access* **Att Key***, see* Name and Att Key.

Extrude **select extrudes** menu

the library extrude to apply to the super string. This extrude can any extrude type, and is applied to the entire length of the string.

Note: **Extrude** *can have attribute substitution from string attributes. See* 8.8.3 Map File Substitution by Attributes.

Comment and Active

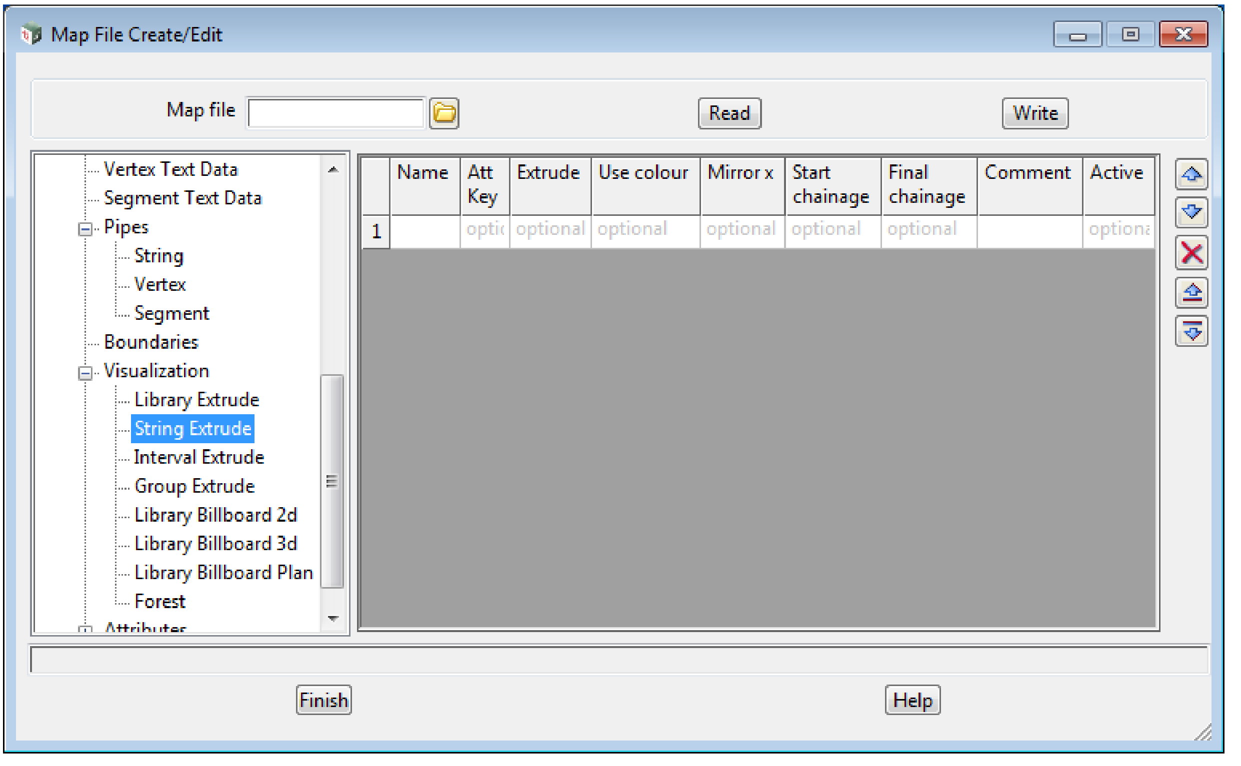
See 8.8.2 Comment and Active Column in Grids.

To assist in debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

Go to the next section 8.8.1.10.2 Visualisation >String Extrude or return to 8.8.1.10 Visualization or 8.8.1 Create/Edit a Map File.

8.8.1.10.2 Visualisation >String Extrude

String Extrude applies an extrude to selected super strings. Note that extrudes can only be applied to super strings or super alignments.



Processing Using Name and Att Key

when a string finds a first match with **Name** *and* **Att Key** *in the grid (see* Matching Using Name and Att Key:*), the string is given the extrude as defined in the parameter of the matching line.*

Name and Att Key

Name *is at Text grid. The* ***string name*** *is matched against* **Name***.*
Att Key *is an Attribute Data grid cell. The* ***string attributes*** *are matched against the attribute details in* **Att Key***.*
For further information on what is allowed in **Name** *and* **Att Key***, and how to access* **Att Key***, see* Name and Att Key*.*

Note: all the fields on the line after **Att Key** can have attribute substitution from string attributes. See 8.8.3 Map File Substitution by Attributes.

Extrude

the library extrude to be applied. This extrude must be of the type ***string extrude****.*

If ***blank****, no extrusion is applied.*

Use Colour

if ***yes****, the colours in the extrusion are used.*

If ***no*** *or* ***blank****, the colour of the super string is used.*

Mirror X

if ***yes****, the x values of the extrusion are mirrored. This allows for one extrusion definition to be used in both a left and right context.*

If ***no*** *or* ***blank****, not mirroring occurs.*

Start Chainage

*if **not blank**, the extrusion starts at this chainage.*
*If **blank**, the extrusion starts at the beginning of the string.*

Final Chainage

*if **not blank**, the extrusion ends at this chainage.*
*If **blank**, the extrusion end at the end of the string.*

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

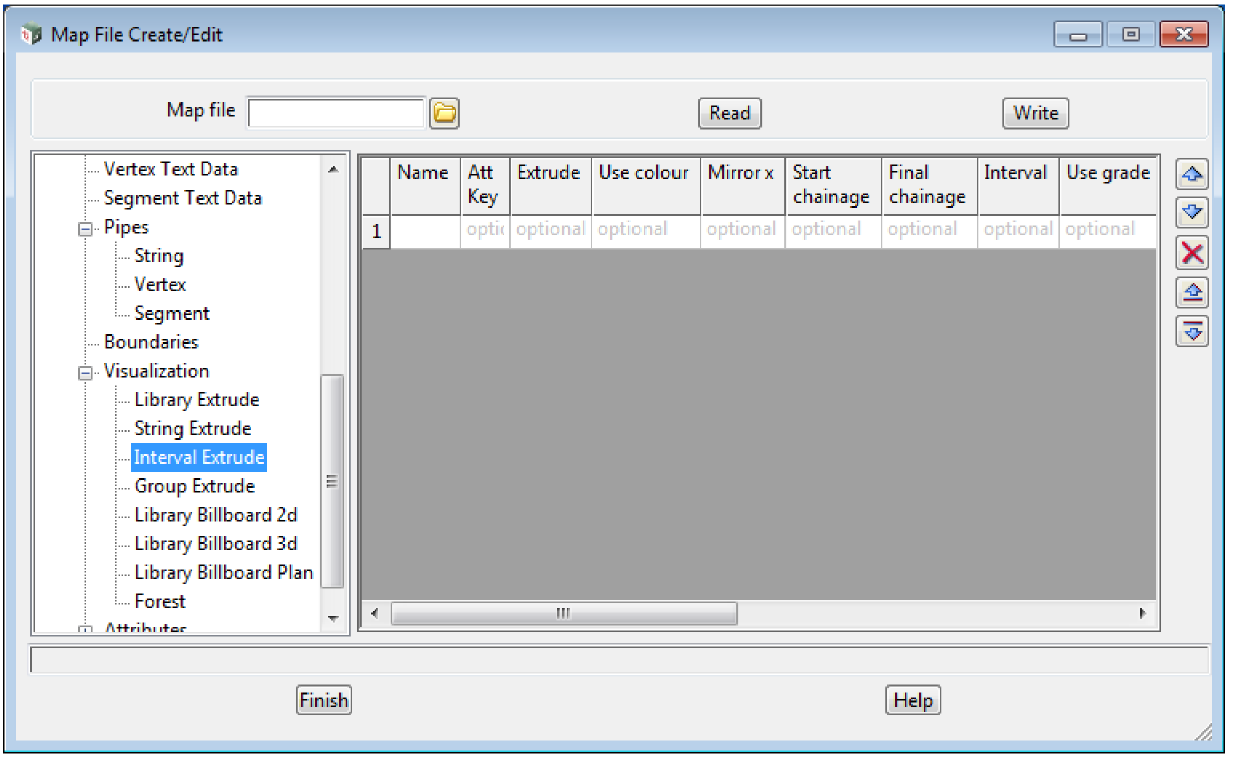
To assist in debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

Go to the next section 8.8.1.10.3 Visualisation >Interval Extrude or return to 8.8.1.10 Visualization or 8.8.1 Create/Edit a Map File.

8.8.1.10.3 Visualisation >Interval Extrude

Interval Extrude applies an extrude to selected super strings. Note that extrudes can only be applied to super strings and super alignments. A typical use of interval extrudes is placing guide posts down the length of a string at a regular interval.The process here is that you reference an existing interval extrude, and can optionally apply one more components of a 3d transformation to it. For more information on transformations, see 12.13.8 3D Transformations.

Each **extrude** drawn at the interval will take the x,y,z, and direction from the string first, then any transformation is applied. So in the example of the post, if no transformation is applied, each post will be positioned on the string and oriented in the direction of the string.



Processing Using Name and Att Key

when a string finds a first match with **Name** *and* **Att Key** *in the grid (see* Matching Using Name and Att Key:*), the string is given the interval extrude as defined in the parameters of the matching line.*

Name and Att Key

Name *is at Text grid. The* ***string name*** *is matched against* **Name**.
Att Key *is an Attribute Data grid cell. The* ***string attributes*** *are matched against the attribute details in* **Att Key**.
For further information on what is allowed in **Name** *and* **Att Key***, and how to access* **Att Key***, see* Name and Att Key.

Note: all the fields on the line after **Att Key** can have attribute substitution from string attributes. See 8.8.3 Map File Substitution by Attributes.

Extrude

the library extrude to be applied. This extrude must be of the type ***interval extrude***.
If ***blank****, no extrusion is applied.*

Use Colour

if ***yes****, the colours in the extrusion are used.*

*If **no** or **blank**, the colour of the super string is used.*

Mirror X

*if **yes**, the x values of the extrusion are mirrored. This allows for one extrusion definition to be used in both a left and right context.*

*If **no** or **blank**, not mirroring occurs.*

Start Chainage

*if **not blank**, the extrusion starts at this chainage.*

*If **blank**, the extrusion starts at the beginning of the string.*

Final Chainage

*if **not blank**, the extrusion ends at this chainage.*

*If **blank**, the extrusion end at the end of the string.*

Interval

the spacing between instances of the extrude along the string.

Use grade

*if **no** or **blank**, the interval extrude is drawn upright.*

*If **yes**, the interval extrude is on the same angle as the segment of the string it is on.*

X/Y/Z rotation

the relative rotation about the x/y/z-axis.

*If **blank**, no rotation is applied.*

X/Y/Z factor

the factor applied to x/y/z coordinates of the extrusion.

*If **blank**, no factor is applied.*

X/Y/Z offset

the offset applied to the x/y/z coordinate of each interval point on the string.

*If **blank**, no offset is applied.*

Comment and Active

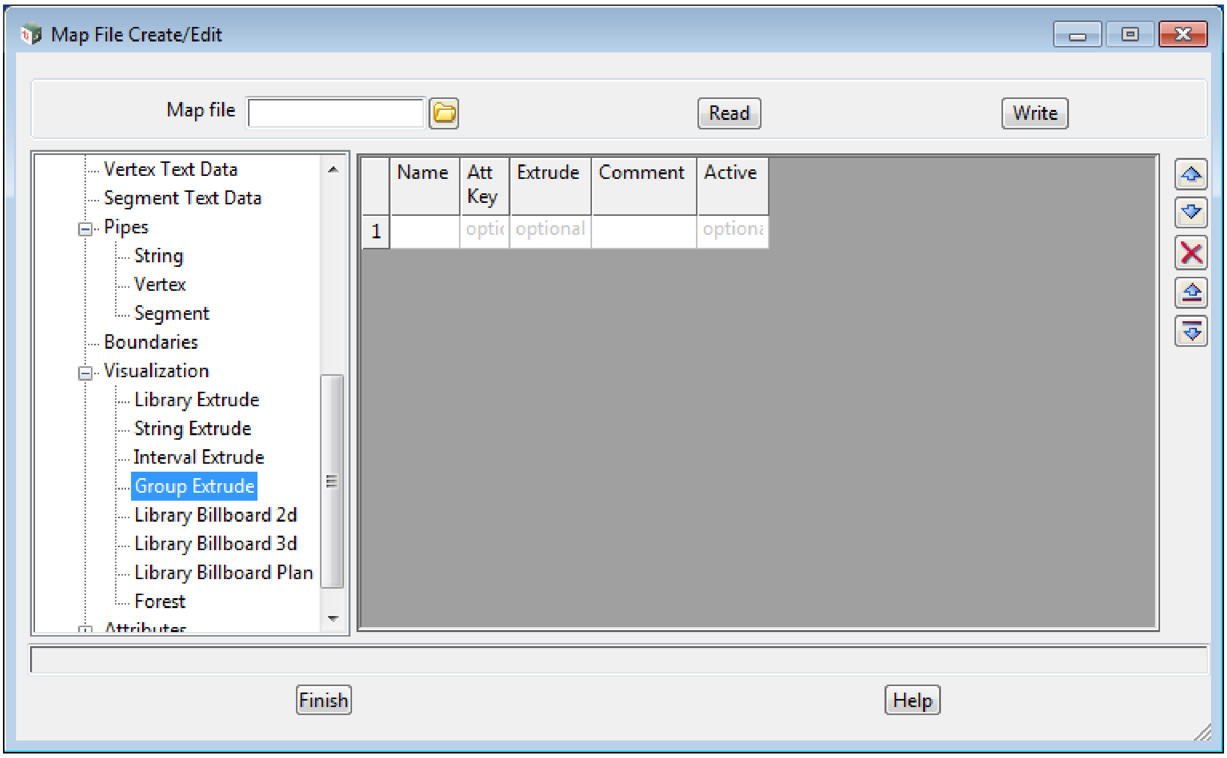
See 8.8.2 Comment and Active Column in Grids.

To assist in debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

Go to the next section 8.8.1.10.4 Visualisation >Group Extrude or return to 8.8.1.10 Visualization or 8.8.1 Create/Edit a Map File.

8.8.1.10.4 Visualisation >Group Extrude

Group Extrude applies a group extrude to selected super strings. Note that extrudes can only be applied to super strings, or super alignments.



Processing Using Name and Att Key

when a string finds a first match with **Name** *and* **Att Key** *in the grid (see* Matching Using Name and Att Key:*), the string is given the group extrude as defined in the parameter of the matching line.*

Name and Att Key

Name *is at Text grid. The* ***string name*** *is matched against* **Name**.
Att Key *is an Attribute Data grid cell. The* ***string attributes*** *are matched against the attribute details in* **Att Key**.
For further information on what is allowed in **Name** *and* **Att Key***, and how to access* **Att Key***, see* Name and Att Key.

Extrude

the library extrude to be applied. This extrude must be of the type ***group extrude***.

If ***blank****, no extrusion is applied.*

Note*:* **Extrude** *can have attribute substitution from string attributes. See* 8.8.3 Map File Substitution by Attributes.

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

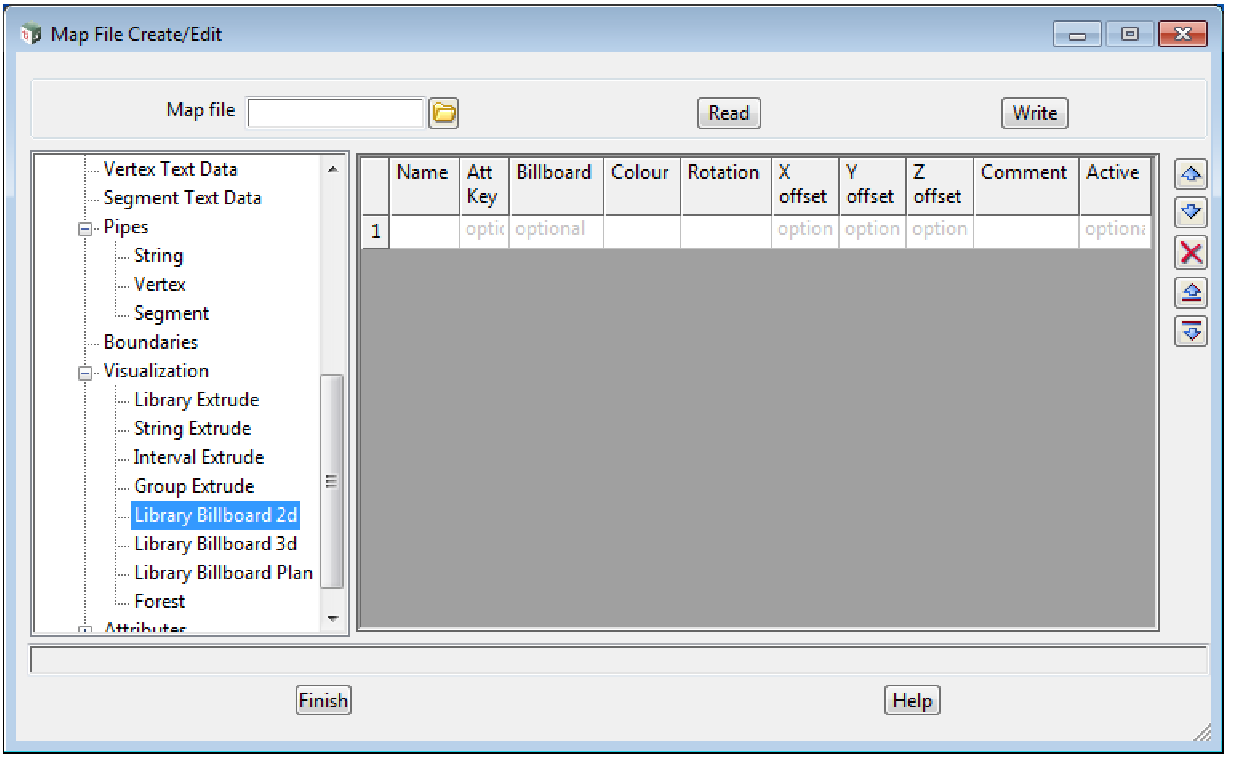
To assist in debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

Go to the next section 8.8.1.10.5 Visualisation >Library Billboard 2d or return to 8.8.1.10 Visualization or 8.8.1 Create/Edit a Map File.

8.8.1.10.5 Visualisation >Library Billboard 2d

This section of documentation is a work in progress and will be updated in subsequent releases.

The process here is that you reference a library billboard, and can optionally apply one more components of a 3d transformation to it. For more information on transformations, see 12.13.8 3D Transformations. For more information on library billboards, see 12.13.5 Billboards.



Processing Using Name and Att Key

when a string finds a first match with **Name** *and* **Att Key** *in the grid (see* Matching Using Name and Att Key:*), the string is given the a library 2d billboard as defined in the parameter of the matching line.*

Name and Att Key

Name *is at Text grid. The* ***string name*** *is matched against* **Name**.
Att Key *is an Attribute Data grid cell. The* ***string attributes*** *are matched against the attribute details in* **Att Key**.
For further information on what is allowed in **Name** *and* **Att Key***, and how to access* **Att Key***, see* Name and Att Key.

Note: all the fields on the line after **Att Key** can have attribute substitution from string attributes. See 8.8.3 Map File Substitution by Attributes.

Billboard

Colour

Rotation

X/Y/Z offset

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

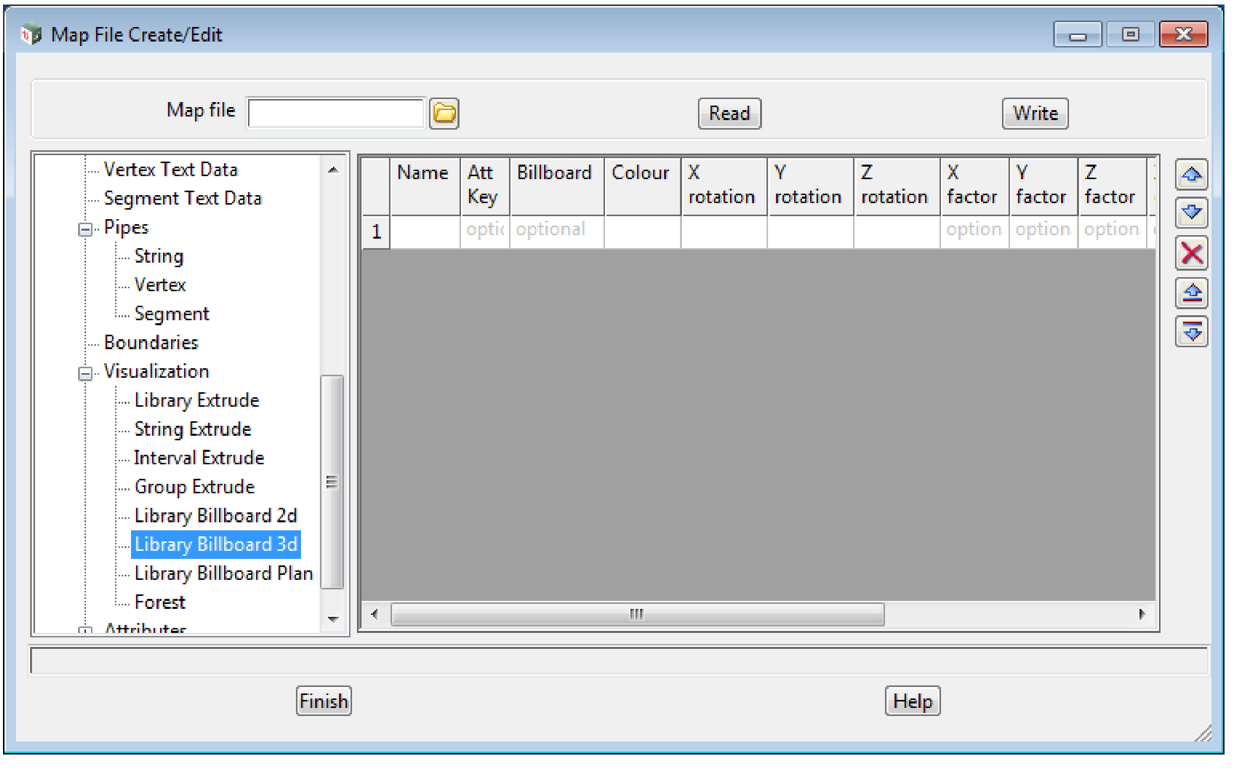
To assist is debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

Go to the next section 8.8.1.10.6 Visualisation >Library Billboard 3d or return to 8.8.1.10 Visualization or 8.8.1 Create/Edit a Map File.

8.8.1.10.6 Visualisation >Library Billboard 3d

This section of documentation is a work in progress and will be updated in subsequent releases.

The process here is that you reference a library billboard, and can optionally apply one more components of a 3d transformation to it. For more information on transformations, see 12.13.8 3D Transformations. For more information on library billboards, see 12.13.5 Billboards.



Processing Using Name and Att Key

when a string finds a first match with **Name** *and* **Att Key** *in the grid (see* Matching Using Name and Att Key:*), the string is given the a library 3d billboard as defined in the parameter of the matching line.*

Name and Att Key

Name *is at Text grid. The* ***string name*** *is matched against* **Name**.
Att Key *is an Attribute Data grid cell. The* ***string attributes*** *are matched against the attribute details in* **Att Key**.
For further information on what is allowed in **Name** *and* **Att Key***, and how to access* **Att Key***, see* Name and Att Key.

Note: all the fields on the line after **Att Key** can have attribute substitution from string attributes. See 8.8.3 Map File Substitution by Attributes.

Billboard

Colour

X/Y/Z rotation

X/Y/Z factor

X/Y/Z offset

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

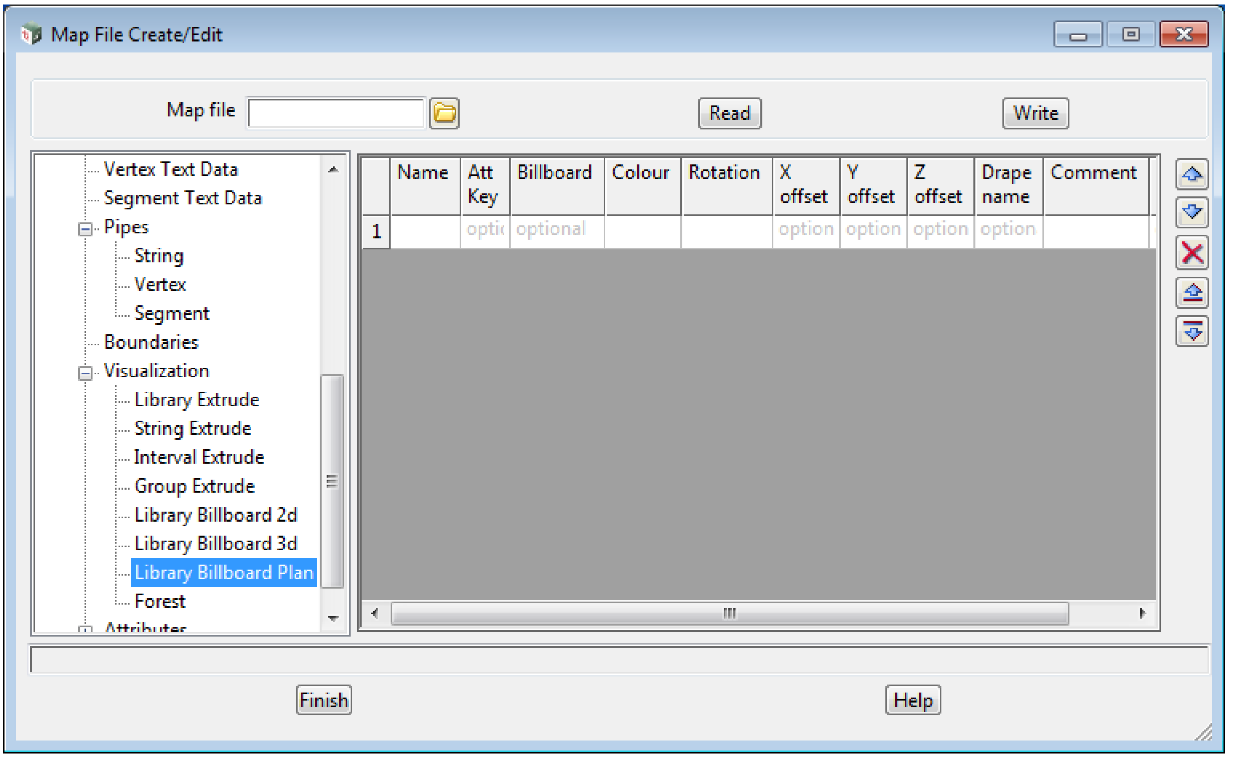
To assist is debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

Go to the next section 8.8.1.10.7 Visualisation >Library Billboard Plan or return to 8.8.1.10 Visualization or 8.8.1 Create/Edit a Map File.

8.8.1.10.7 Visualisation >Library Billboard Plan

This section of documentation is a work in progress and will be updated in subsequent releases.

The process here is that you reference a library billboard, and can optionally apply one more components of a 3d transformation to it. For more information on transformations, see 12.13.8 3D Transformations. For more information on library billboards, see 12.13.5 Billboards.



Processing Using Name and Att Key

when a string finds a first match with **Name** *and* **Att Key** *in the grid (see* Matching Using Name and Att Key:*), the string is given the a library plan billboard as defined in the parameter of the matching line.*

Name and Att Key

Name *is at Text grid. The* ***string name*** *is matched against* **Name**.
Att Key *is an Attribute Data grid cell. The* ***string attributes*** *are matched against the attribute details in* **Att Key**.
For further information on what is allowed in **Name** *and* **Att Key***, and how to access* **Att Key***, see* Name and Att Key.

Note: all the fields on the line after **Att Key** can have attribute substitution from string attributes. See 8.8.3 Map File Substitution by Attributes.

Billboard

Colour

Rotation

X/Y/Z offset

Drape name

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

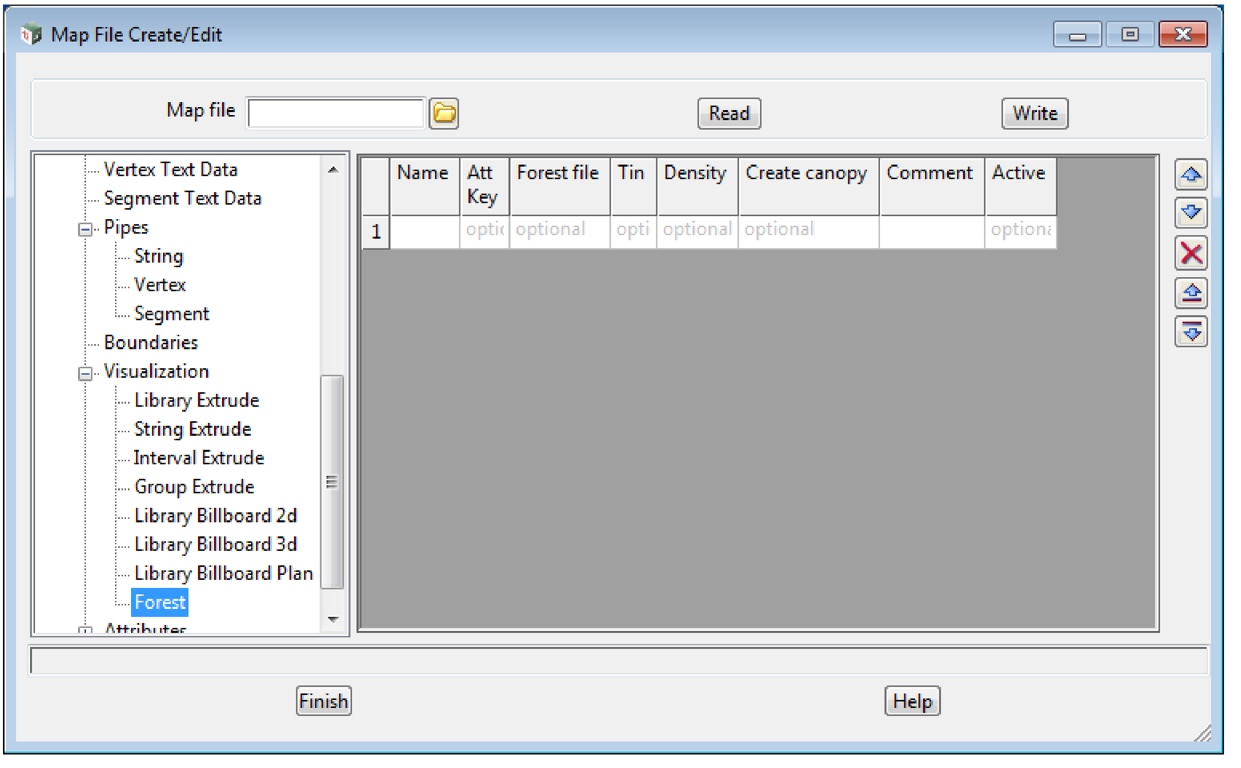
To assist in debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

Go to the next section 8.8.1.10.8 Visualisation >Forest or return to 8.8.1.10 Visualization or 8.8.1 Create/Edit a Map File.

8.8.1.10.8 Visualisation >Forest

Forest applies a Forest file to selected strings. If a string is open then its is taken as closed by joining the first and last points before applying the Forest file.

This section of documentation is a work in progress and will be updated in subsequent releases.



Processing Using Name and Att Key

when a string finds a first match with **Name** *and* **Att Key** *in the grid (see* Matching Using Name and Att Key:*), a forest file is applied to the string as specified by the parameter of the matching line.*

Name and Att Key

Name *is at Text grid. The* ***string name*** *is matched against* **Name**.
Att Key *is an Attribute Data grid cell. The* ***string attributes*** *are matched against the attribute details in* **Att Key**.
For further information on what is allowed in **Name** *and* **Att Key***, and how to access* **Att Key***, see* Name and Att Key.

Note: all the fields on the line after **Att Key** can have attribute substitution from string attributes. See 8.8.3 Map File Substitution by Attributes.

Forest file

the name of the Forest file to use for the string.

Tin

the tin to put the base on the trees on

Density

the number of tries per hectare

Create canopy

Comment and Active

See 8.8.2 Comment and Active Column in Grids.

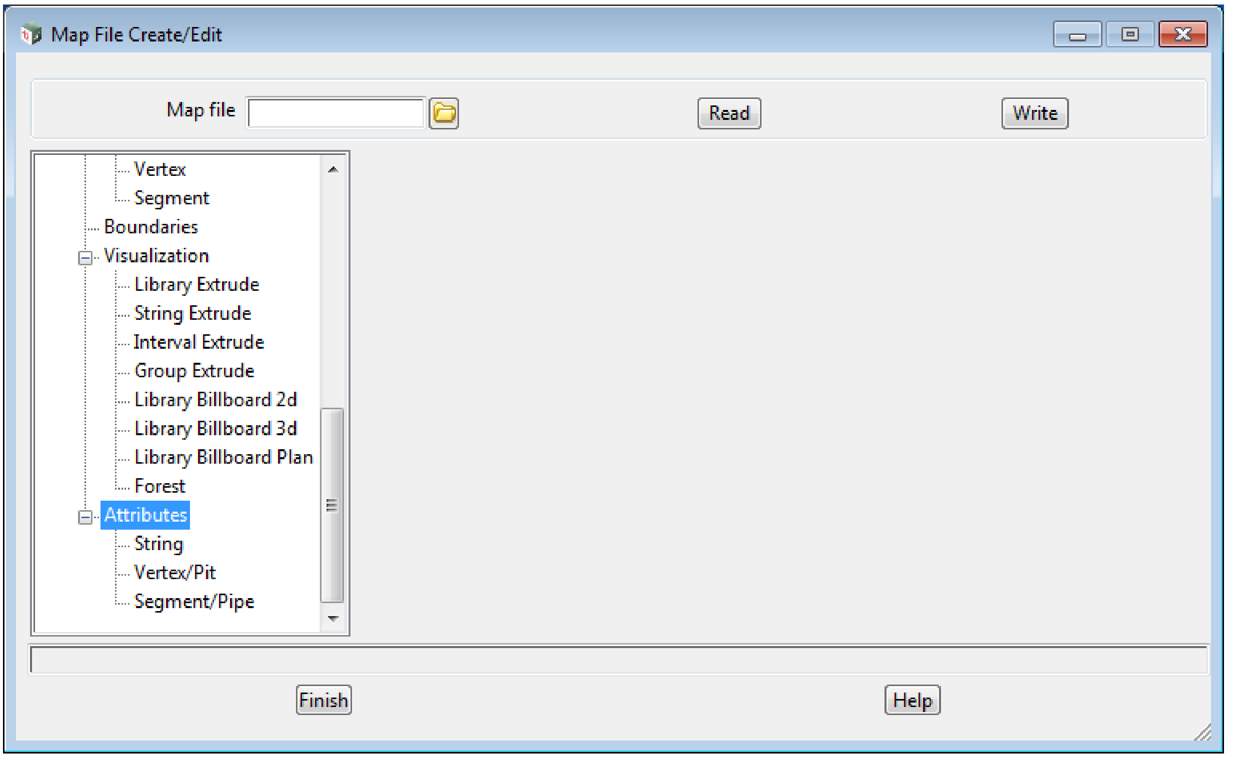
To assist in debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

Go to the next section 8.8.1.11 Attributes or return to 8.8.1.10 Visualization or 8.8.1 Create/Edit a Map File.

8.8.1.11 Attributes

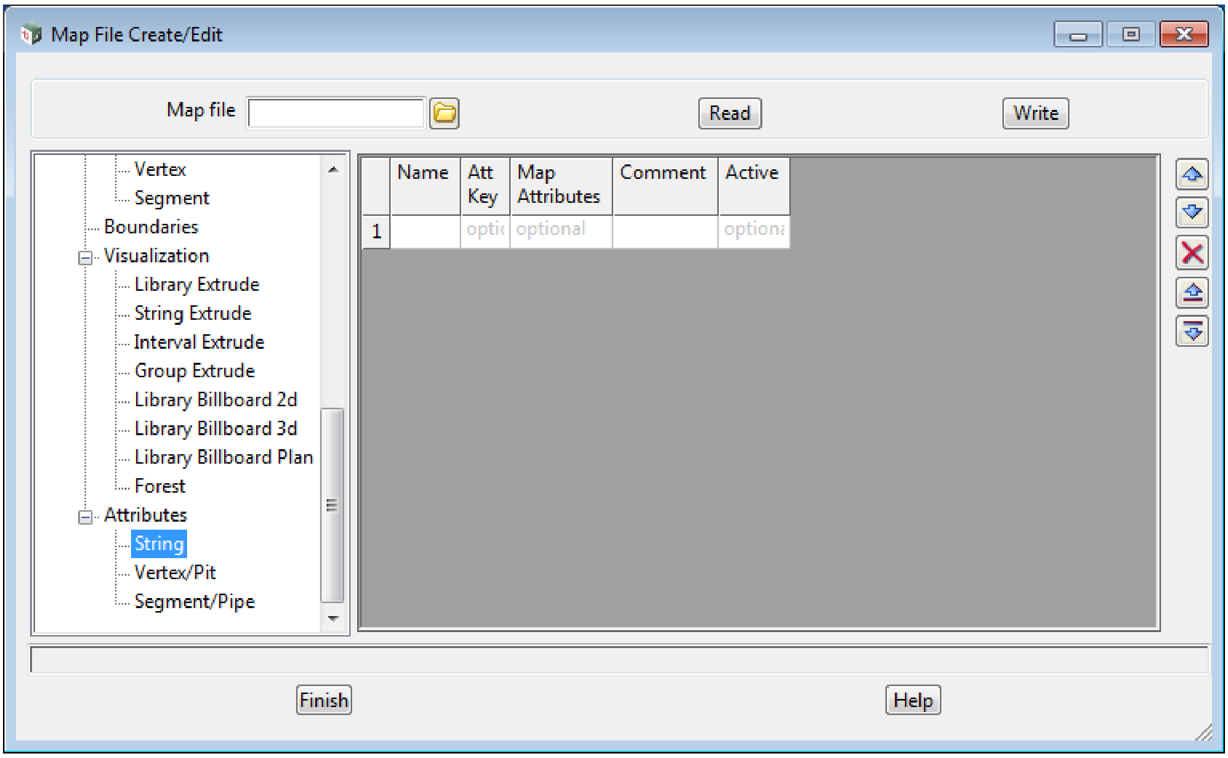
The **Attributes** node in the ***Map File*** which is used to apply attributes to Strings, Vertices and/or Segments.

For information on *String*, go to 8.8.1.11.1 Attributes >String
Vertex 8.8.1.11.2 Attributes >Vertex/Pit
Segment 8.8.1.11.3 Attributes >Segment/Pipe



8.8.1.11.1 Attributes >String

Strings matches on string name and string attributes and adds the attributes in the ***Map Attributes*** column to the string.



Processing Using Name and Att Key for Attributes>String Node

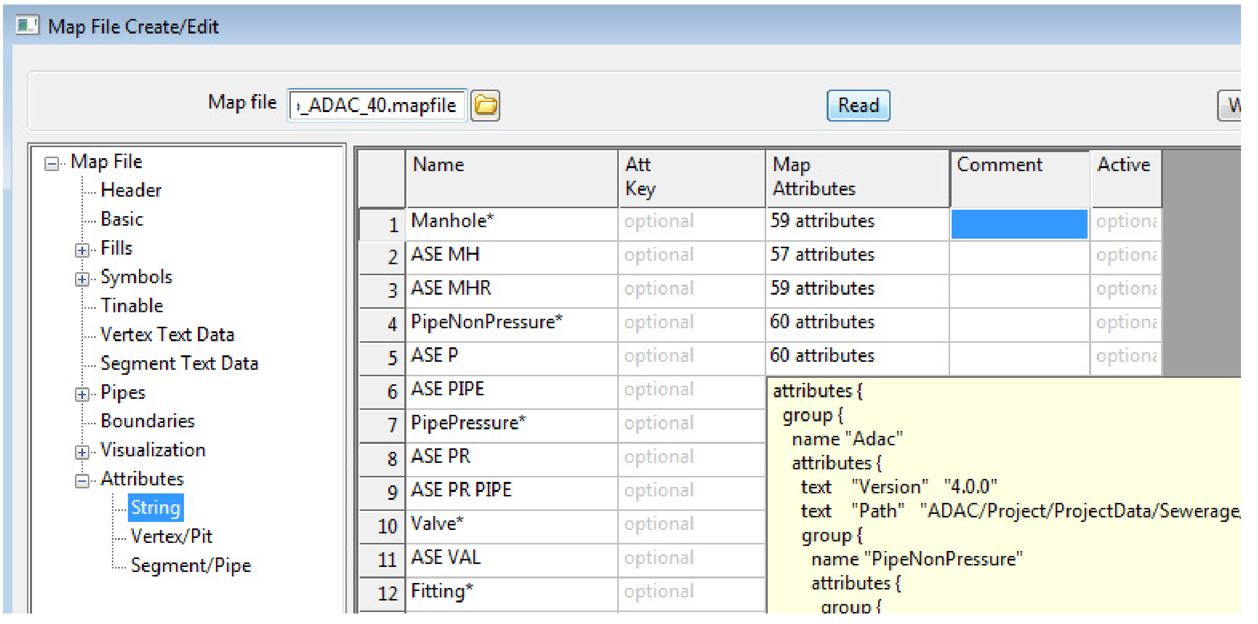
when a string finds a first match with **Name** *and* **Att Key** *in the grid (see* [Matching Using Name and Att Key:](#)*), the Map Attributes are added as string attributes to the matching string.*

Name and Att Key

Name *is at Text grid. The* ***string name*** *is matched against* **Name**.
Att Key *is an Attribute Data grid cell. The* ***string attributes*** *are matched against the attribute details in* **Att Key**.
For further information on what is allowed in **Name** *and* **Att Key**, *and how to access* **Att Key**, *see* [Name and Att Key](#).

Map Attributes

attributes that are added as string attributes.



Comment and Active

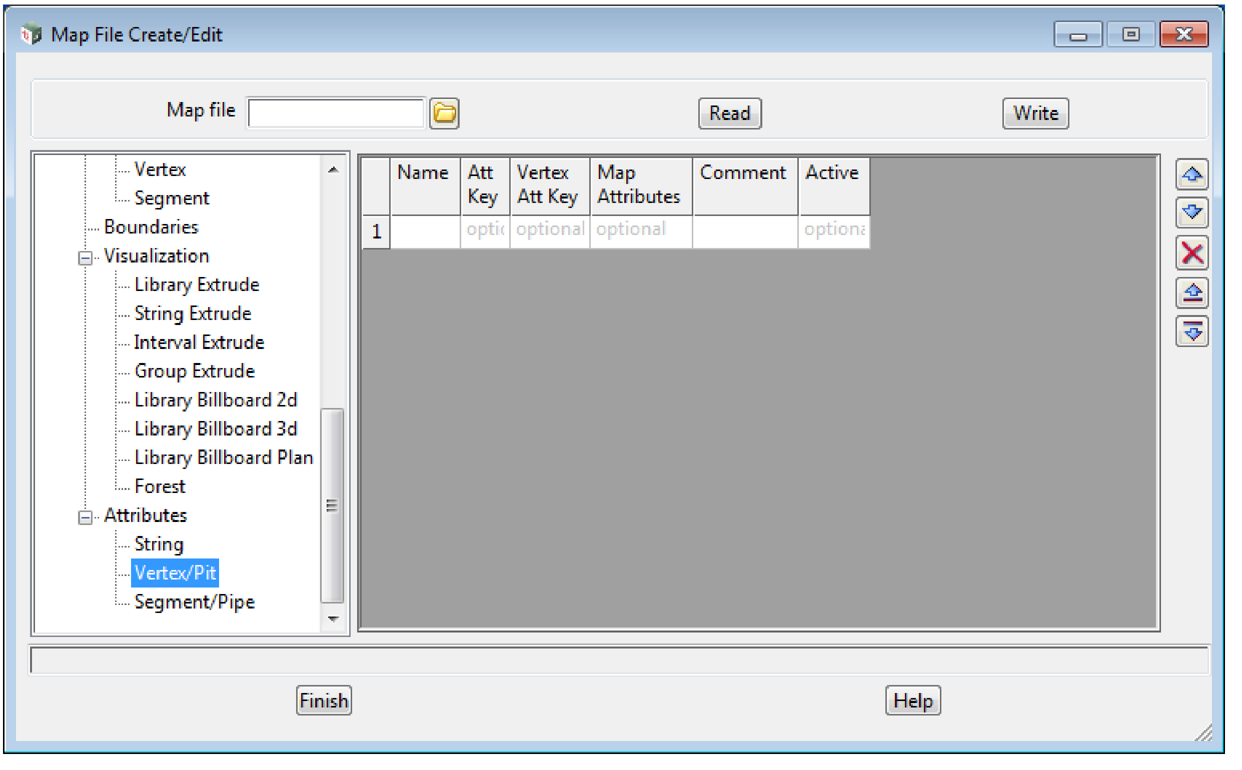
See 8.8.2 Comment and Active Column in Grids.

To assist in debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

Go to the next section 8.8.1.11.1 Attributes >String or return to 8.8.1.11 Attributes or 8.8.1 Create/Edit a Map File.

8.8.1.11.2 Attributes >Vertex/Pit

Vertex/Pit matches on string name, string attributes and vertex attributes and adds the attributes in the ***Map Attributes*** column as vertex attributes to the selected vertex of the super string or drainage string.



Processing Using Name, Att Key and Vertex Att Key

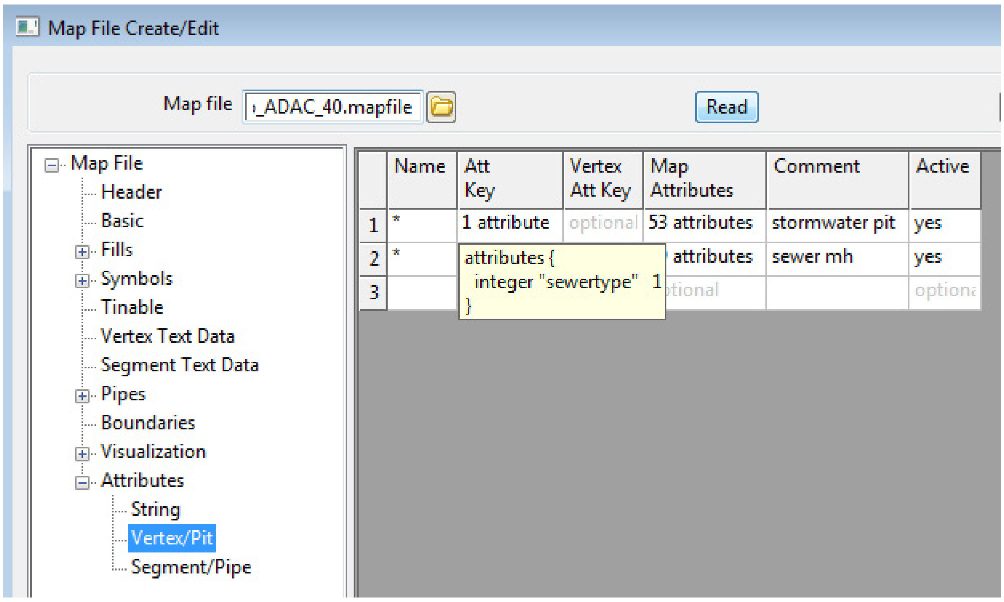
when a string finds a first match with **Name**, **Att Key** *and* **Vertex Att Key** *in the grid (see* Matching Using Name, Att Key and Vertex Att Key:*), the selected vertex has the attributes in **Map Attributes** added to it.*

Name, Att Key and Vertex Att Key

Name *is at Text grid. The **string name** is matched against* **Name**.
Att Key *is an Attribute Data grid cell. The **string attributes** are matched against the attribute details in* **Att Key**.
Vertex Att Key *is an Attribute Data grid cell. The **vertex attributes** are matched against the attribute details in* **Att Key**.
For further information on what is allowed in **Name** *and* **Att Key**, *and how to access* **Att Key**, *see* Name, Att Key and Vertex Att Key.

Map Attributes

attributes that are added as vertex/pit attributes.



Comment and Active

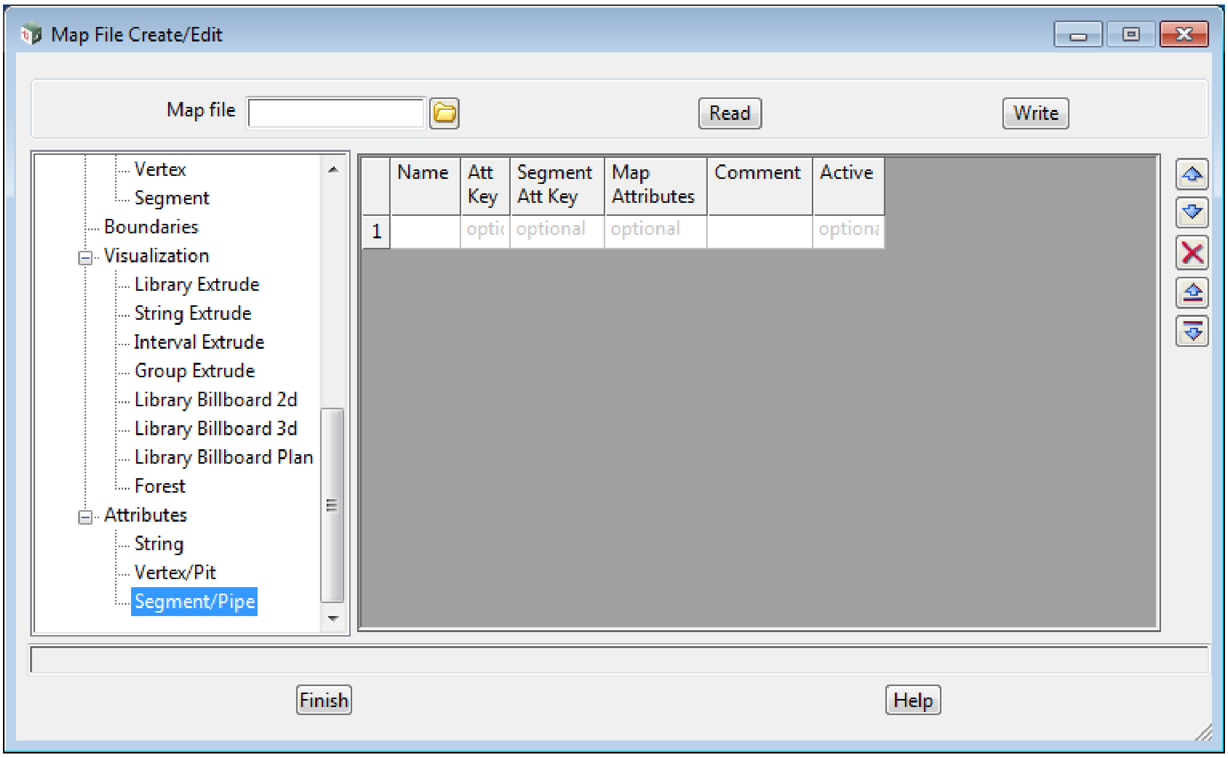
See 8.8.2 Comment and Active Column in Grids.

To assist in debugging a ***Map File***, match logging can be turned on. *See* 8.8.4 Map File Logging.

Go to the next section 8.8.1.11.3 Attributes >Segment/Pipe or return to 8.8.1.11 Attributes or 8.8.1 Create/Edit a Map File.

8.8.1.11.3 Attributes >Segment/Pipe

Segment/Pipe matches on string name, string attributes and vertex attributes and adds the attributes in the ***Map Attributes*** column as segment attributes to the selected segment of the super string or drainage string.



Processing Using Name, Att Key and Segment Att Key

when a string finds a first match with **Name**, **Att Key** *and* **Segment Att Key** *in the grid (see* Matching Using Name, Att Key and Segment Att Key:*), the selected segments has the attributes in* ***Map Attributes*** *added to it*

Name, Att Key and Segment Att Key

Name *is at Text grid and the text entered into* **Name** *can include wild cards* ***** *and wild characters* ***?****.*
The string name is matched against **Name**. *This field can not be blank.*
Att Key *is an Attribute Data grid cell which contains the definition of the attributes and their values that are to be matched against. There can be more than one attribute in the Attribute Data but they must have unique names.*
The ***string*** *attributes are matched against the attribute details in* **Att Key.**
Segment Att Key *is an Attribute Data grid cell which contains the definition of the segment attributes and their values that are to be matched against. There can be more than one segment attribute in the Attribute Data but they must have unique names.*
The ***segment*** *attributes are matched against the attribute details in* **Segment Att Key**.
To access ***Att Key*** *or* **Segment Att Key** *data, click LB on the* ***Att Key/*** **Segment Att Key** *field to highlight the field, then click LB again to bring up the* **Attribute Data** *panel. To enter data, see* Attribute Data Panel.

Matching Using Name, Att Key and Segment Att Key:

Strings are first created via the ***Basic*** *node and given a string name of either the Entity Name or the text given in the Name column of the Basic Map File grid.*
Starting with the first line of the grid, matching and processing occurs as follows
If **Name, Att Key** *and* **Segment Att Key** *are not blank, and a match of the string name occurs with* **Name**, *a match of the string attributes occurs with* **Att Key** *and a match of the string segment attributes*

occurs with **Segment Att Key**, *then the rest of the fields for this line of the Map File grid are used on this string segment.*

If **Name** *and* **Att Key** *are not blank, and* **Segment Att Key** *is blank, and a match of the string name occurs with* **Name** *and a match of the string attributes occurs with* **Att Key**, *then the rest of the fields for this line of the Map File grid are used on each string segment.*

If **Name** *and* **Segment Att Key** *are not blank, and* **Att Key** *is blank, and a match of the string name occurs with* **Name** *and a match of a string segment attribute occurs with* **Segment Att Key**, *then the rest of the fields for this line of the Map File grid are used on this string segment.*

If **Name** *in not blank and* **Att Key** *and* **Segment Att Key** *are blank, and a match of the string name occurs with* **Name**, *then the rest of the fields for this line of the Map File grid are used on each string segment.*

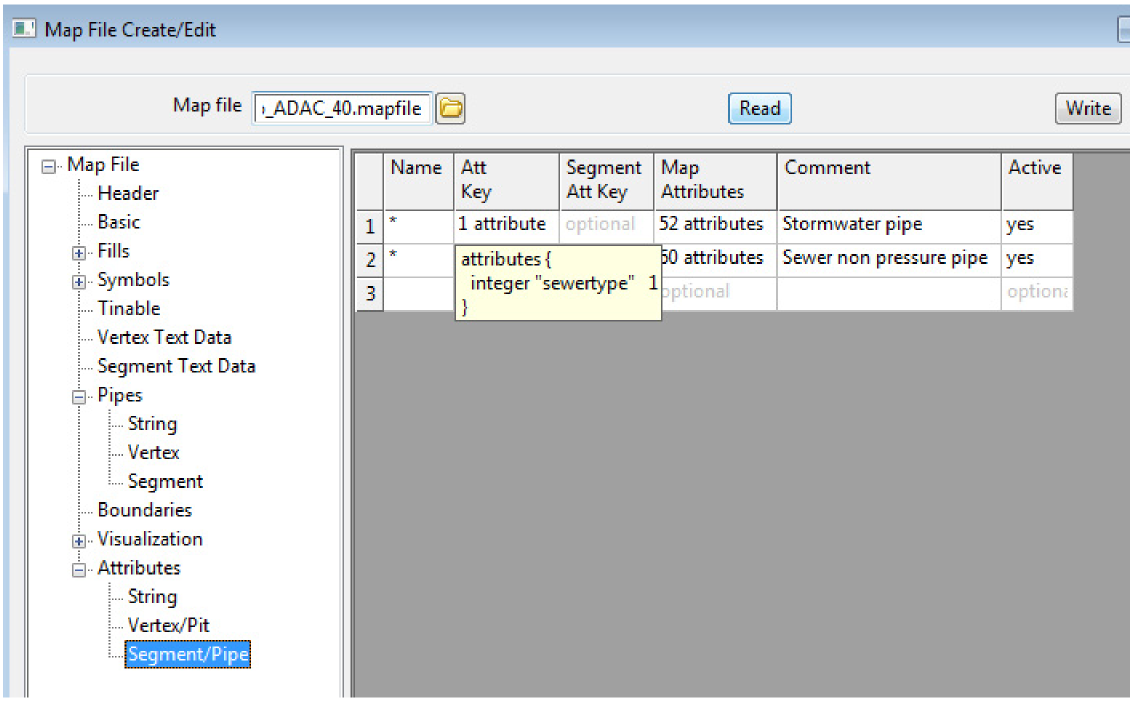
If **Name** *in blank then no match occurs and this line of the Map File grid is ignored.*

If a ***match occurs***, *then no tests for matches against* **Name**, **Att Key** *and* **Segment Att Key** *further down in the grid are made.*

If ***no match occurs***, *then this line of the map file grid is ignored and a test for a match is made against the next line of the grid.*

Map Attributes

attributes that are added as segment/pipe attributes.



Comment and Active

See 8.8.2 Comment and Active Column in Grids.

To assist in debugging a *Map File*, match logging can be turned on. *See* 8.8.4 Map File Logging.

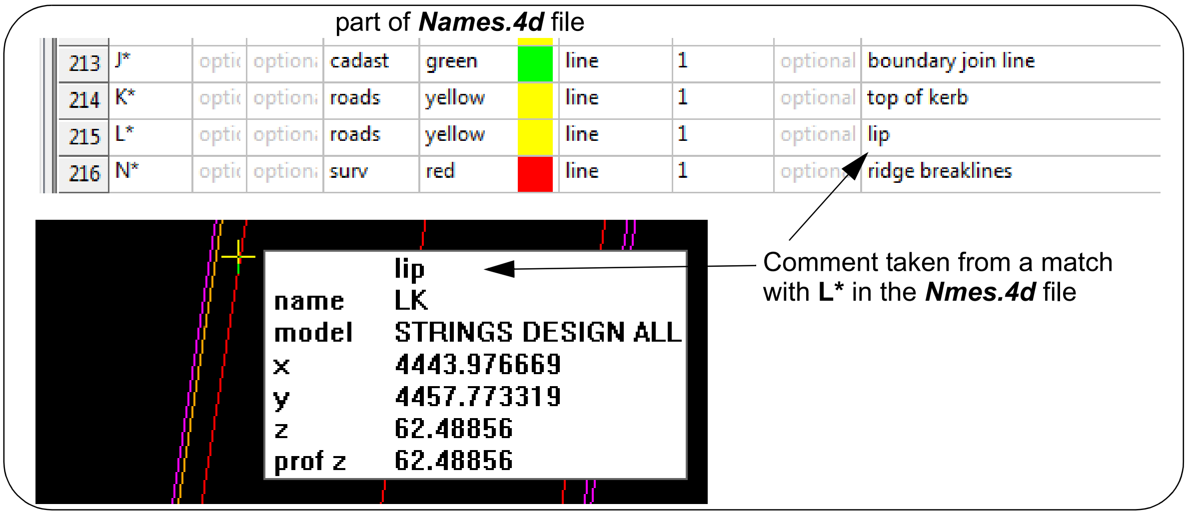
Go to the next section 8.8.3 Map File Substitution by Attributes or return to 8.8.1.11 Attributes or 8.8.1 Create/Edit a Map File or 8.8 Map Files.

8.8.2 Comment and Active Column in Grids

Comment

comment to be to be used in this line of the map file or names files.

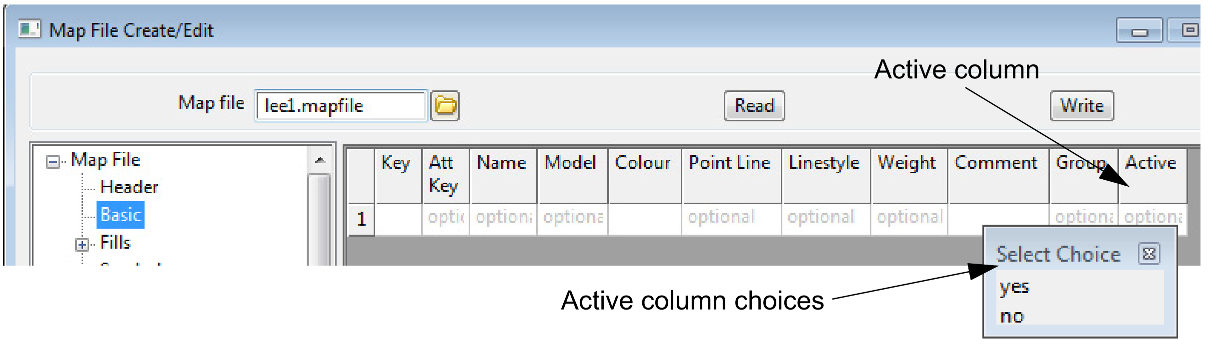
When there is a comment in the ***names.4d*** *file and* ***Data tooltips*** *are turned on (****D*** *snap) and a string matches a row in the* ***Names.4d*** *file with a comment, then the comment is displayed at the top of the Data Tool tip.*



Active

There is an **Active** *column in each grid so that the individual rows of the grid can be turned off and so not used. This means that the row does not have to be deleted.*

If the value in the Active column for a row column is left ***blank****, then the row is used.*
If the value in the Active column for a row column is ***yes****, then the row is used.*
If the value in the Active column for a row column is ***no****, then the row is* ***not*** *used.*



Return to 8.8 Map Files.

8.8.3 Map File Substitution by Attributes

Once there has been a first match for an element with the **Name and Att key (or Vertex Att key or Segment Att key)** of a line in the grid, the remaining fields in the rest of the grid line are applied to the element.

To increase the usefulness of the remaining fields in the line, it is possible to use the ***value of an attribute*** from the element, rather then just the static value in the field.

To use the value of an attribute in the gird field, simply ***specify the attribute pathname prefixed by the $ sign***.

For example, if it was the ***Tinable*** section of the Map File (see 8.8.1.5 Tinable,) and the tinability of the string is to be given by a string attribute called ***12d Field/My Tinability***, type

$12d Field/My Tinability

into the ***Tinable*** cell of the grid.

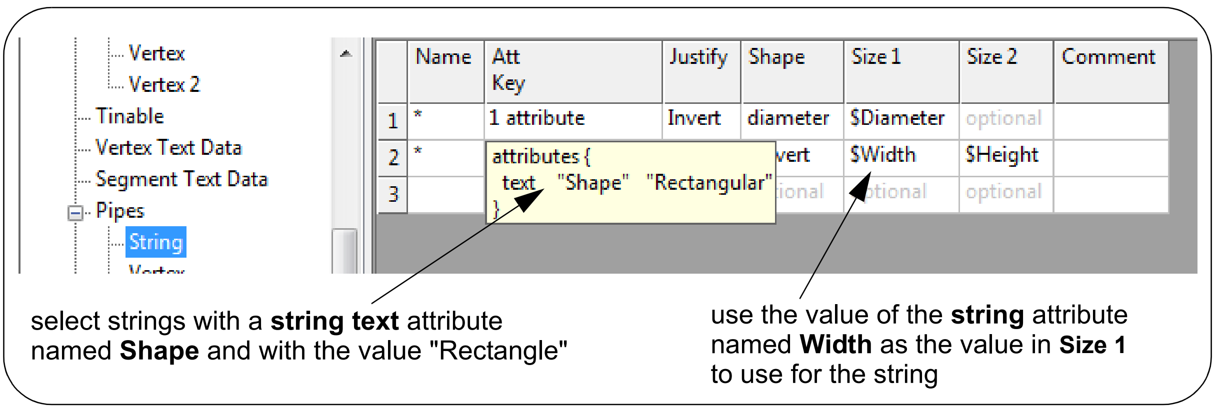
So **$** denotes the start of the attribute substitution.

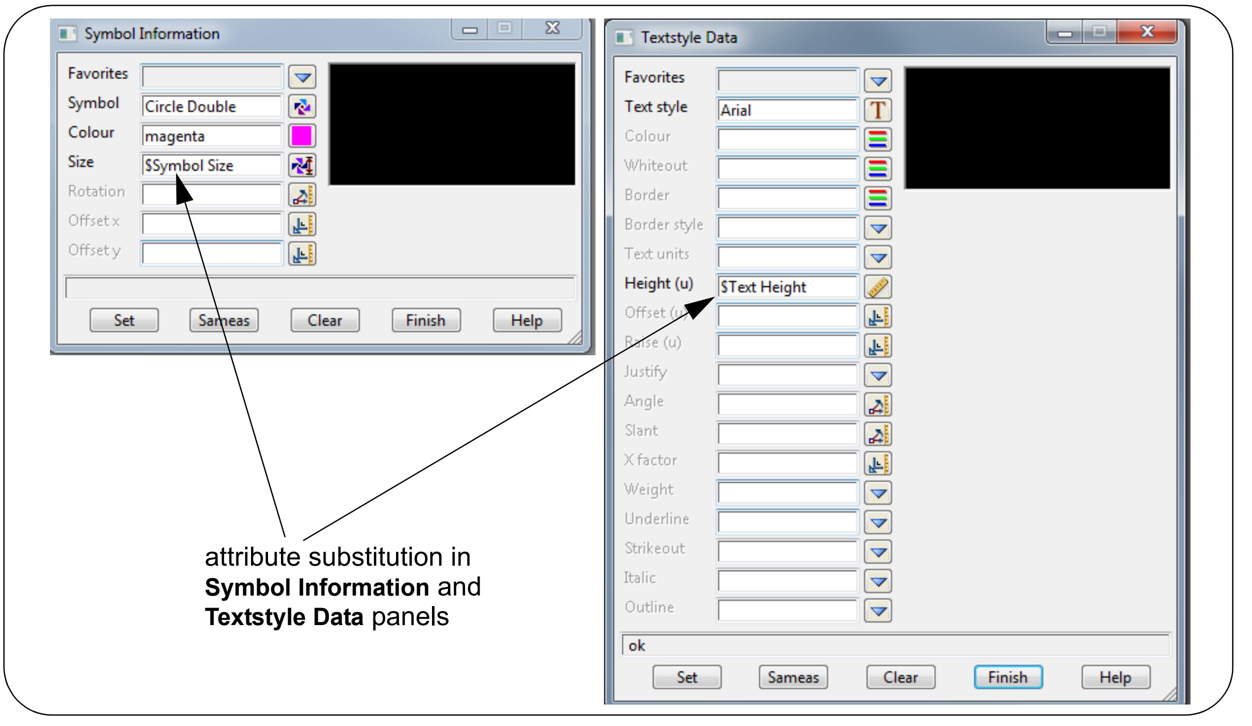
The ***value of the attribute*** must be a **valid** answer (in the case of the Tinable cell, it must be **no**, **yes**, or **point**). If the value of the attribute is **not** a valid answer, a logline in the output window will be generated to identify the element in error and the line in the mapfile.

Note that the **$** character must be the first character in the field, and that the rest of the field is taken as the attribute pathname so quotes should not be used even if there are spaces in the attribute pathname. Hence only one attribute can be used in the substitution process for a grid cell.

Remember that attribute names are case sensitive and in the case where more than one attribute of the same name exists, the first attribute found will be used.

Note - Attribute substitution is valid in the **Textstyle Data** and **Symbol Information** panels on a grid line.





An Important Restriction

There is currently no way of distinguishing between string, vertex or segment attributes in the grid cells.

So if you are in a section of the Map File that has:

(a) only an **Att Key**, then ***only string attributes*** can be used for substitution in the grid cells in that section. For example 8.8.1.5 Tinable.

(b) an **Att Key** and a **Vertex Att Key** then ***only vertex attributes*** can be used for substitution in the grid cells in that section. For example 8.8.1.4.2 Symbols >Vertex.

(c) an **Att Key** and a **Segment Att Key** then ***only segment attributes*** can be used for substitution in the grid cells in that section. For example 8.8.1.8.3 Pipes >Segment.

Go to the next section 8.8.4 Map File Logging or return to 8.8 Map Files.

8.8.4 Map File Logging

To help determine what is going on, or not going on, when using a Map File, ***Logging*** can be turned on.

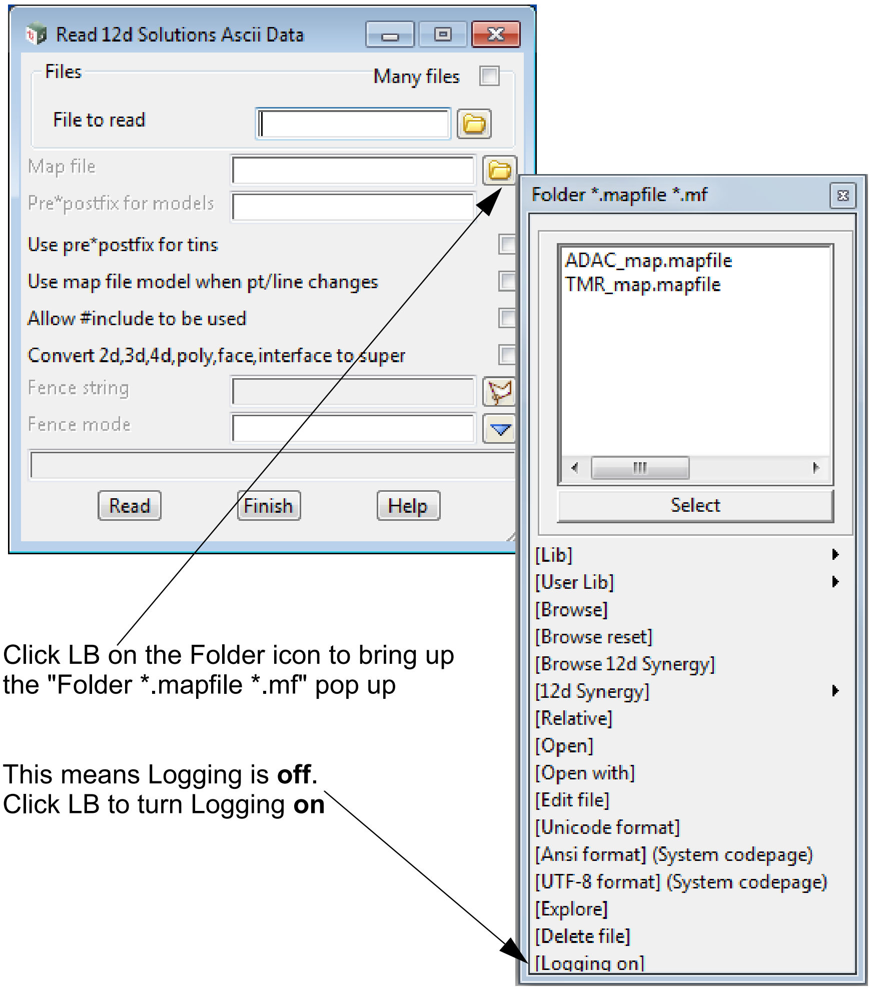
When turned on, as a Map File processes the data and a match occurs, an entry is written to the ***Output Window*** giving in what section of the Map File section the match occurred, the line number of that section of the grid that the match occurred (written as item number), and by clicking onto the created log line, the mapped object will be highlighted.

To turn **Map File Logging** on or off, click LB or RB on the folder icon at the end of the **Map File** panel field to bring up the ***Folder *mapfile *.mf*** pop up.

If **[Logging off]** is displayed then Logging is now turned **on** and clicking LB on [Logging off] will turn Logging off.

If **[Logging on]** is displayed then Logging is now turned **off** and clicking LB on [Logging on] will turn Logging on.

For example, when reading in a ***12da*** file, clicking LB on the **Map file** panel field brings up the ***Folder *mapfile *.mf*** pop up.



Note: Logging can generate a large number of log lines and this could affect the performance of ***12d Model***. So it is a good idea to save the project first, or run the logging in a temporary project.

Go to the next section 8.8.5 Apply Map File or return to 8.8 Map Files.

8.8.5 Apply Map File

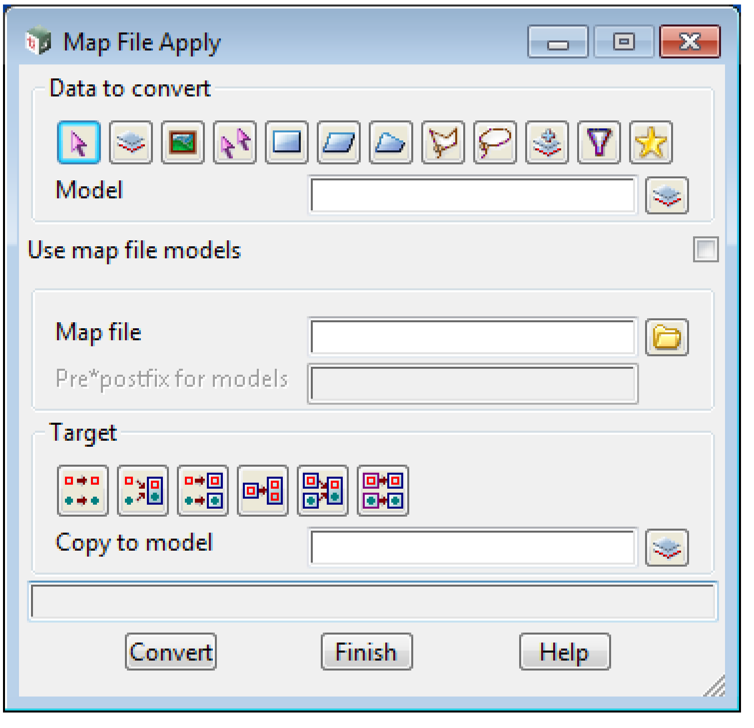
Position of option on menu: **File =>Map file =>Apply**

Position of option on menu: **Utilities =>H-Z =>Map**

Map applies a 12d Model Map File to the selected strings. The map file can be used to change string attributes such as string names, models, colours, breakline type and style, apply extrudes, apply polygon fills *etc*.

The layout and operation of a Map File is described in the section 8.8.1 Create/Edit a Map File.

Selecting **Map** displays the **Map File Apply** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be mapped.

Use map file models	tick box		

*if **ticked**, use the models from the map file.*

*If **not ticked**, use the models from the Target.*

Map file	map file box		*.mapfile, *.mf files

the Map File to use.

Prefix for models	input		available models

*If **non-blank**, the* **prefix for models** *field gives the characters to be prepended and appended to the model names given in the Map File. The prepended and appended characters are entered into the* **prefix for models** *field, separated by a *. For example, **pre*pos** would add **pre** before each model name and **pos** after each model name. For prepend only, no * is required. The **pre** and **pos** can included spaces.*

Target type

data selection type - for a full description go to 4.19.3 Data Source.

Target info input

extra information required for the target.

Map button

run the option - process the selected strings

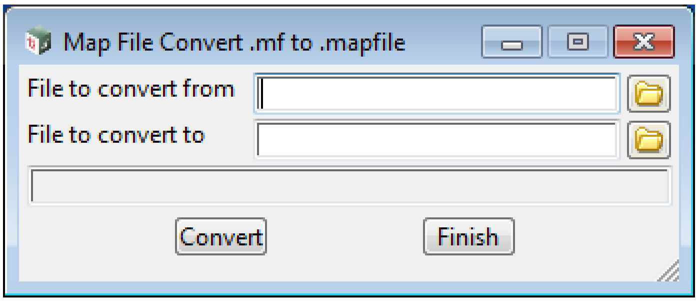
Go to the next section 8.8.6 Convert .mf Files to .mapfile or return to 8.8 Map Files.

8.8.6 Convert .mf Files to .mapfile

Position of option on menu: **File I/O =>Map files =>Convert .mf to .mapfile**

The **map file convert** option converts a map file in the pre-V9 format to the XML map file format introduced in ***12d Model 9***.

Selecting **Convert .mf to .mapfile** displays the **Map File Convert .mf to .mapfile** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
File to convert from	file box		*.mf
name of the pre-V9 map file to be converted to the new map file format			
File to convert to	file box		*.mapfile
name of the new converted map file			
Convert	button		
convert a file in the pre-V9 map file format to the new map file format.			

Go to the next section 8.8.7 Create/Edit .mf File or return to 8.8 Map Files.

8.8.7 Create/Edit .mf File

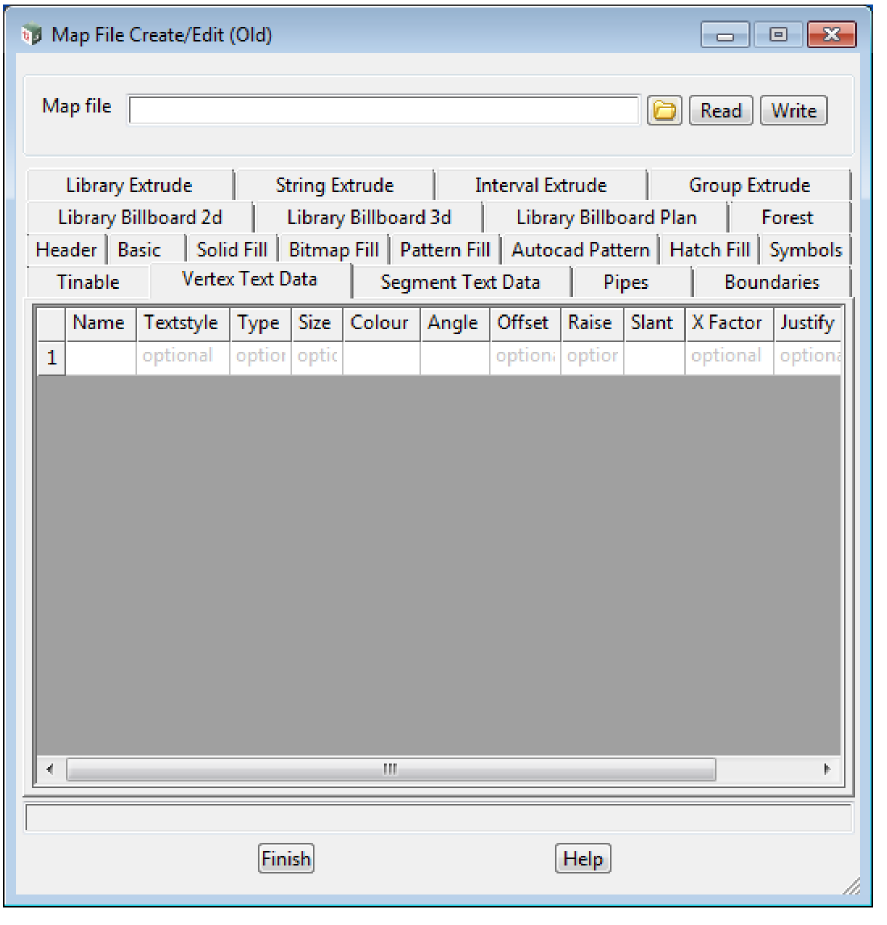
Position of option on menu: **File I/O =>Map files =>Create/edit .mf**

The **Create/edit .mf** option **creates** and/or **edit** the 12d Map Files in the pre-V9 format.

This old format is now superseded so the user should use 8.8.6 Convert .mf Files to .mapfile to convert old .mf files to the .mapfile format, or the option 8.8.1 Create/Edit a Map File does this conversion when you select a .mf file. This editor is only supplied for a while and will be removed in a future version.

Vertex Text Data

The **Vertex Text Data** tab sets parameters for drawing text on the vertices of super strings.



Key **select name** menu

key to match the string name against in the Vertex Text Data tab. The key can include wild cards (*) and wild characters (?).

If a match occurs, then the rest of the grid fields for this line of this tab of the map file are used and no other tests for matches against keys further down in the table are made.

If a match does not occur, then a test for a match is made against the key on the next line of this tab of the map file.

Textstyle **select text style** menu

the textstyle used for any vertex text for all strings matching the key.
This field can not be blank.

Type screen, paper, world

the textstyle type used for any vertex text for all strings matching the key. This field can not be blank.

Size **measures** menu

the size of the vertex text to be used for each vertex of all strings matching the key. This field can not be blank.

Colour **select colour** menu

the colour of the vertex text to be used for each vertex of all strings matching the key. This field can not be blank.

Angle **measures** menu

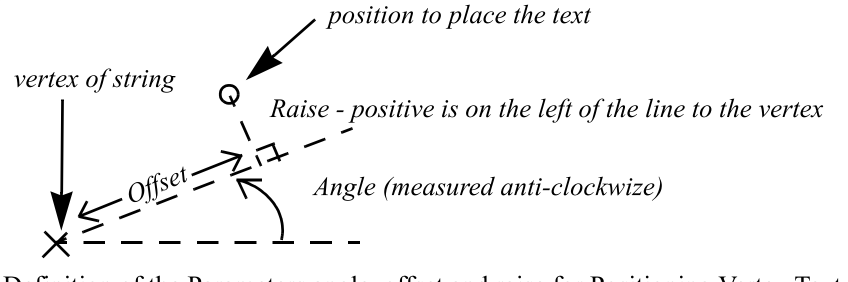
the angle of the vertex text to be used for each vertex of all strings matching the key. This field can not be blank.

Offset **measures** menu

the distance to offset the text from each vertex of the strings matching the key. This field can not be blank.

Raise **measures** menu

the distance to perpendicularly raise the text for each vertex of all strings matching the key. This field can not be blank.



Definition of the Parameters angle, offset and raise for Positioning Vertex Text

Slant **Measures** menu

the slant for the vertex text of all strings matching the key. This field can not be blank.

X factor **Measures** menu

the x-factor for the vertex text of all strings matching the key. This field can not be blank.

Justify **Select Justification Choice** menu

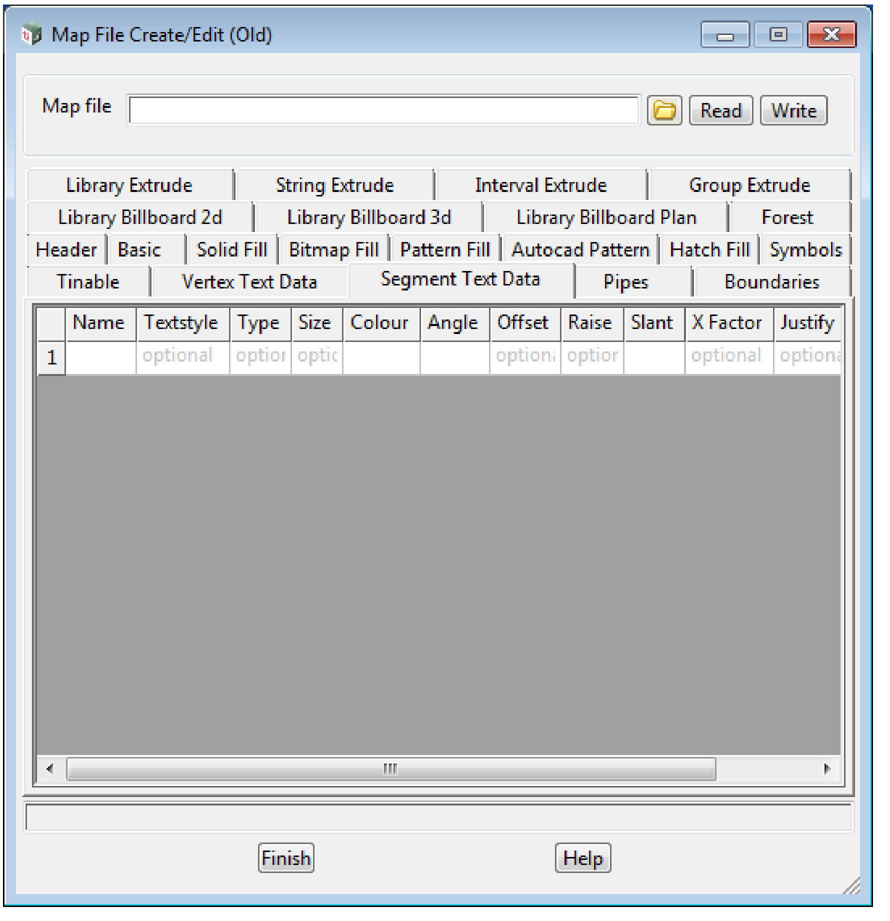
the justification for the vertex text of all strings matching the key. This field can not be blank.

Comment

comment to be to be used in this line of the map file. A double slash // is written out to the text map file before the comment, and a comment in an text map file is read in and displayed without the double slash.

Segment Text Data

the Segment Text Data tab sets parameters for drawing text on the segments of super strings. The text is positioned about the mid point of the segment.



Key **select name** menu

key to match the string name against in the ***Segment Text Data*** *tab. The key can include wild cards (*) and wild characters (?).*

If a match occurs, then the rest of the grid fields for this line of this tab of the map file are used and no other tests for matches against keys further down in the table are made.

If a match does not occur, then a test for a match is made against the key on the next line of this tab of the map file.

Textstyle **select text style** menu

the textstyle used for any segment text for all strings matching the key.
This field can not be blank.

Type screen, paper, world

the textstyle type used for any segment text for all strings matching the key.

This field can not be blank.

Size **measures** menu

the size of the segment text to be used for each segment of all strings matching the key.
This field can not be blank.

Colour **select colour** menu

the colour of the segment text to be used for each segment of all strings matching the key.
This field can not be blank.

Angle **measures** menu

the angle of the segment text to be used for each segment of all strings matching the key.

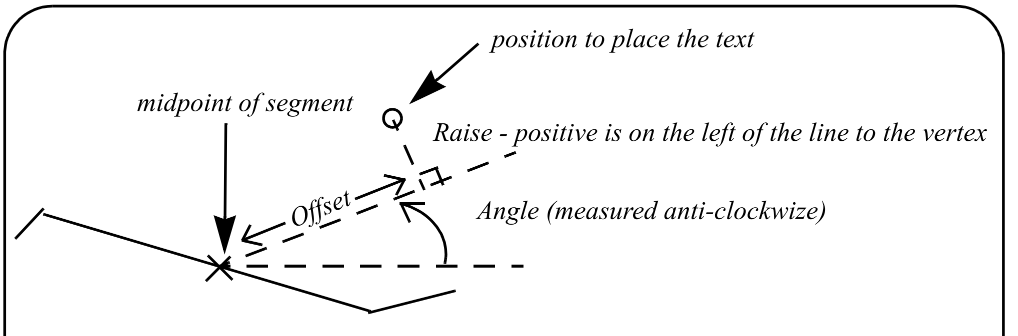
This field can not be blank.

Offset **measures** menu

the distance to offset the text from each segment of the strings matching the key. This field can not be blank.

Raise **measures** menu

the distance to perpendicularly raise the text for each segment of all strings matching the key. This field can not be blank.



Definition of the Parameters angle, offset and raise for Positioning Segment Text

Slant **Measures** menu

the slant for the segment text of all strings matching the key. This field can not be blank.

X factor **Measures** menu

the x-factor for the segment text of all strings matching the key. This field can not be blank.

Justify **Select Justification Choice** menu

the justification for the segment text of all strings matching the key. This field can not be blank.

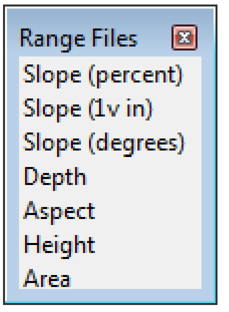
Comment

comment to be to be used in this line of the map file. A double slash // is written out to the text map file before the comment, and a comment in an text map file is read in and displayed without the double slash.

Go to the next section 8.9 Range Files or return to 8.8 Map Files.

8.9 Range Files

Position on menu: ***File =>Range files***



Range files consist of a list of ranges and colours, one set per line

lower_value upper_value range_colour

where this line represents all values satisfying

lower_value < value <= upper_value.

See

8.9.1 Percent Slope Range File
8.9.2 Degrees Slope Range File
8.9.3 '1v in' Slope Range File
8.9.4 Depth Range File
8.9.5 Aspect Range File
8.9.6 Height Range File
8.9.7 Area Range File
8.9.8 Creating Ranges in a Range File

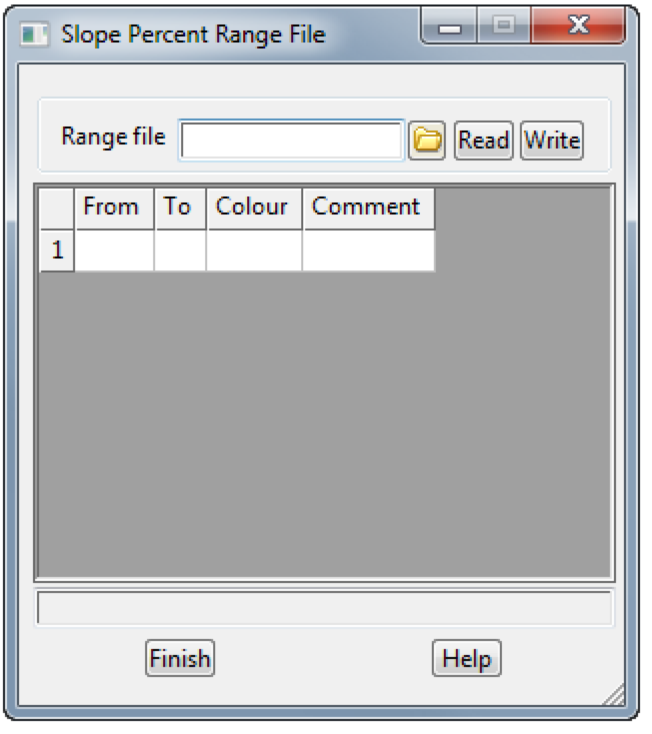
8.9.1 Percent Slope Range File

For **Percent slope colouring**, the range file consists of a list of ranges of percent cross fall of the slopes, and colours, one set per line, in the grid

lower_value upper_value range_colour

where this line represents all percentage slopes satisfying

lower_value < value <= upper_value.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Range file	file box		available *.srf files

the name of the percent slope range file

Read button

read in the file in the **Range file** *field*

Write button

write out the file in the **Range file** *field*

Grid Cells

for **Percent slope colouring**, *the range file consists of a grid of ranges and colours, one set per line, in the grid*

lower_value upper_value range_colour

where this line represents all percentage slopes satisfying

lower_value < value <= upper_value

See 8.9.8 Creating Ranges in a Range File *for quickly creating numbers of range file lines with colours.*

From real

the lower value for this line of the range file

To	real	

the upper value for this line of the range file

Colour	colour box	available colours

colour for the range

Comment	text	

a comment for this line of the grid

Go to the next section 8.9.2 Degrees Slope Range File or return to 8.9 Range Files.

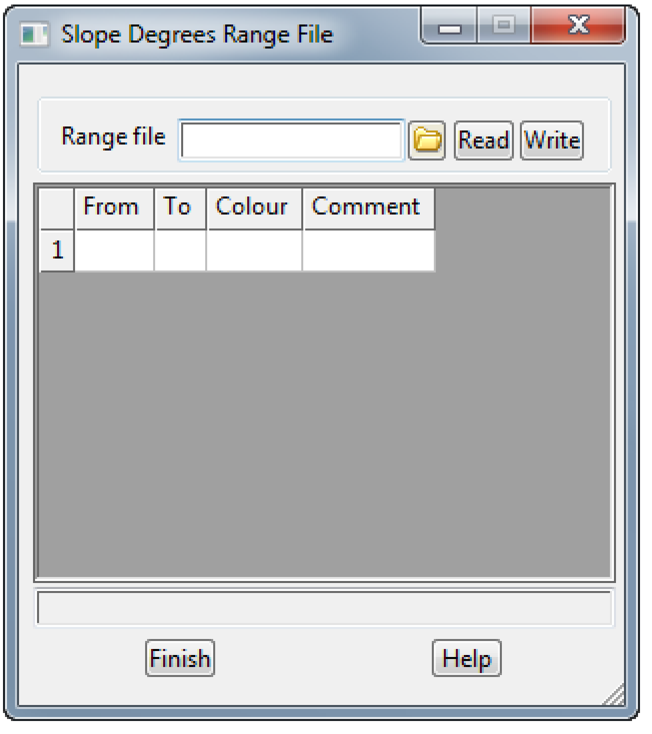
8.9.2 Degrees Slope Range File

For **Degrees slope colouring**, the range file consists of a list of ranges of angles of the slopes (in degrees in 4.17.1 HP Notation), and colours, one set per line, in the grid

lower_value upper_value range_colour

where this line represents all degrees satisfying

lower_value < value <= upper_value.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Range file	file box		available *.srf files

the name of the angles of the slope (in degrees) range file

Read button

read in the file in the **Range file** *field*

Write button

write out the file in the **Range file** *field*

Grid Cells

for **Degrees slope colouring**, *the range file consists of a grid of ranges and colours, one set per line, in the grid*

lower_value upper_value range_colour

where this line represents all angles (in degrees in 4.17.1 HP Notation*) of slopes satisfying*

lower_value < value <= upper_value

See 8.9.8 Creating Ranges in a Range File *for quickly creating numbers of range file lines with colours.*

From real

the lower value for this line of the range file

To	real	

the upper value for this line of the range file

Colour	colour box	available colours

colour for the range

Comment	text	

a comment for this line of the grid

Go to the next section 8.9.3 '1v in' Slope Range File or return to 8.9 Range Files.

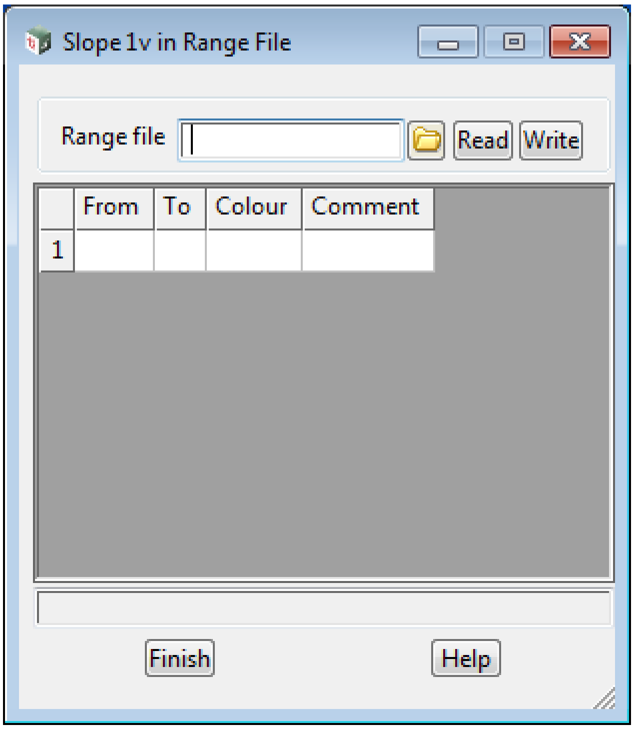
8.9.3 ′1v in′ Slope Range File

For **1v in slope colouring**, the range file consists of a list of ranges of slopes (in "1 in" values) and colours, one set per line, in the grid

lower_value upper_value range_colour

where this line represents all "1 in" slopes satisfying

lower_value < value <= upper_value.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Range file	file box		available *.srf files

the name of the 1v in slope range file

Read button

read in the file in the **Range file** *field*

Write button

write out the file in the **Range file** *field*

Grid Cells

for **1v in slope colouring***, the range file consists of a grid of ranges and colours, one set per line, in the grid*

lower_value upper_value range_colour

where this line represents all "1 in" slopes satisfying

lower_value < value <= upper_value

See 8.9.8 Creating Ranges in a Range File *for quickly creating numbers of range file lines with colours.*

From real

the lower value for this line of the range file

To	real	

the upper value for this line of the range file

Colour	colour box	available colours

colour for the range

Comment	text	

a comment for this line of the grid

Go to the next section 8.9.4 Depth Range File or return to 8.9 Range Files.

8.9.4 Depth Range File

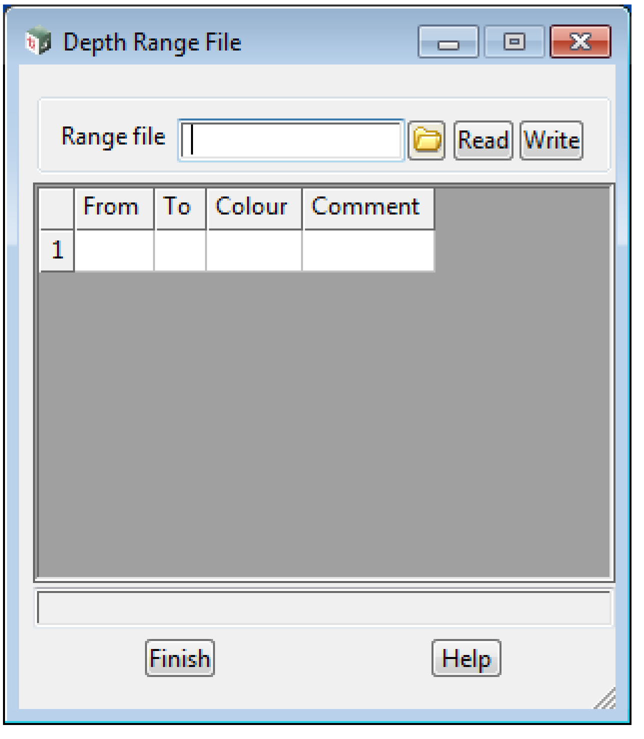
For **Depth colouring**, the range file consists of a list of ranges of depths and colours, one set per line, in the grid

lower_value upper_value range_colour

where this line represents all depths satisfying

lower_value < value <= upper_value.

Note: depths are measured with down being the positive direction.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Range file	file box		available *.drf files

the name of the depth range file

Read	button		

read in the file in the **Range file** *field*

Write	button		

write out the file in the **Range file** *field*

Grid Cells

for **Depth colouring**, *the range file consists of a grid of depth ranges and colours, one set per line, in the grid*

lower_value upper_value range_colour

where this line represents all depths satisfying

lower_value < value <= upper_value

See 8.9.8 Creating Ranges in a Range File *for quickly creating numbers of range file lines with colours.*

From	real		

the lower value for this line of the range file

To real

the upper value for this line of the range file

Colour colour box available colours

colour for the range

Comment text

a comment for this line of the grid

Go to the next section 8.9.5 Aspect Range File or return to 8.9 Range Files.

8.9.5 Aspect Range File

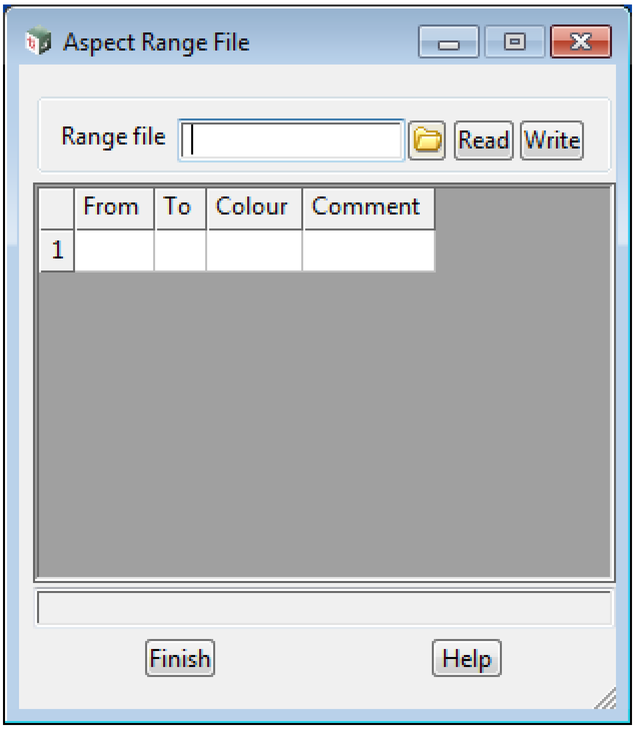
For **Aspect colouring**, the range file consists of a list of aspect ranges and colours, one set per line, in the grid

lower_value   upper_value   range_colour

where this line represents all aspect values satisfying

lower_value <   value   <= upper_value.

aspect is in degrees in 4.17.1 HP Notation measured clockwise from North.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Range file	file box		available *.arf files

the name of the aspect range file

Read button

read in the file in the **Range file** *field*

Write button

write out the file in the **Range file** *field*

Grid Cells

for **Aspect colouring**, *the range file consists of a grid of ranges of aspect and colours, one set per line, in the grid*

lower_value   upper_value   range_colour

where this line represents all aspect values satisfying

lower_value <   value   <= upper_value

See 8.9.8 Creating Ranges in a Range File *for quickly creating numbers of range file lines with colours.*

From real

the lower value for this line of the range file

To	real	

the upper value for this line of the range file

Colour	colour box	available colours

colour for the range

Comment	text	

a comment for this line of the grid

Go to the next section 8.9.6 Height Range File or return to 8.9 Range Files.

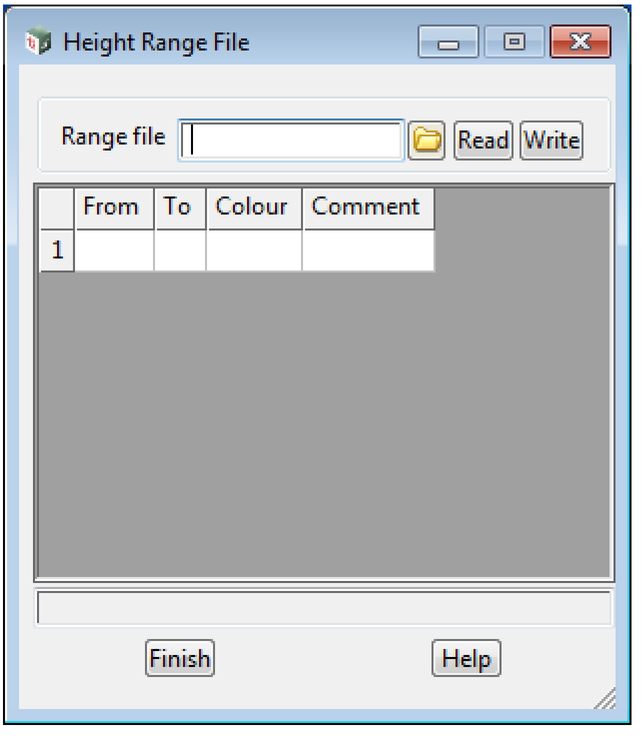
8.9.6 Height Range File

For **Height colouring**, the range file consists of a list of height ranges and colours, one set per line, in the grid

lower_value   upper_value   range_colour

where this line represents all percentage slopes satisfying

lower_value <   value   <= upper_value



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Range file	file box		available *.hrf files

the name of the height range file

Read	button		

read in the file in the **Range file** *field*

Write	button		

write out the file in the **Range file** *field*

Grid Cells

for **Height colouring**, *the range file consists of a grid of height ranges and colours, one set per line, in the grid*

lower_value   upper_value   range_colour

where this line represents all heights satisfying

lower_value <   value   <= upper_value

See 8.9.8 Creating Ranges in a Range File *for quickly creating numbers of range file lines with colours.*

From	real		

the lower value for this line of the range file

To	real	

the upper value for this line of the range file

Colour	colour box	available colours

colour for the range

Comment	text	

a comment for this line of the grid.

Go to the next section 8.9.7 Area Range File or return to 8.9 Range Files.

8.9.7 Area Range File

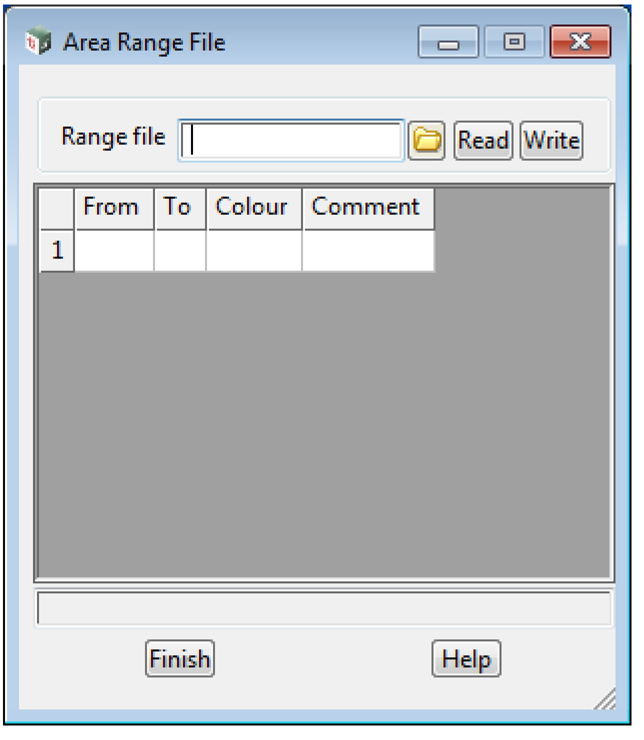
For **area colouring**, the range file consists of a list of ranges of areas and colours, one set per line, in the grid

lower_value upper_value range_colour

where this line represents all percentage slopes satisfying

lower_value < value <= upper_value.

When **Area** is selected, it brings up the **Area Range File** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Range file	file box		available *.arf files

the name of the area range file

Read button

read in the file in the **Range file** *field*

Write button

write out the file in the **Range file** *field*

Grid Cells

for **area colouring**, *the range file consists of a grid of ranges and colours, one set per line, in the grid*

lower_value upper_value range_colour

where this line represents all percentage slopes satisfying

lower_value < value <= upper_value

See 8.9.8 Creating Ranges in a Range File *for quickly creating numbers of range file lines with colours.*

From real

the lower value for this line of the range file

To	real	

the upper value for this line of the range file

Colour	colour box	available colours

colour for the range

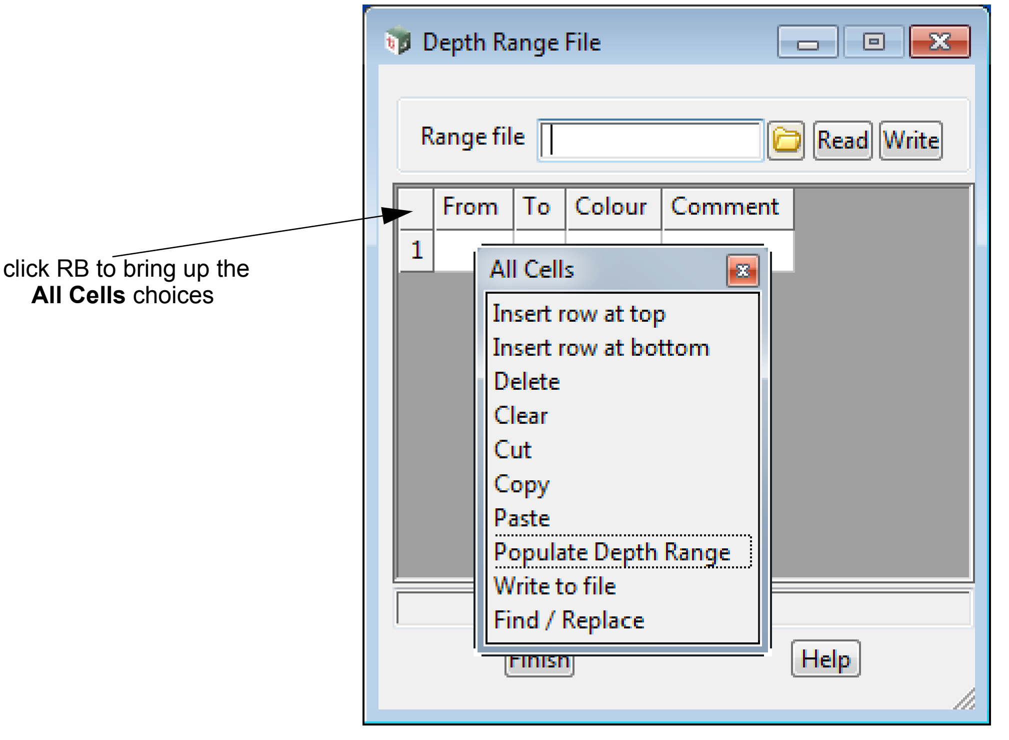
Comment	text	

a comment for this line of the grid

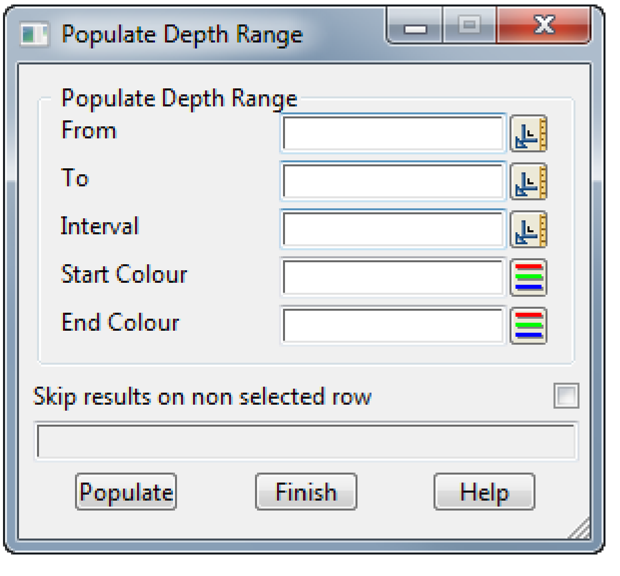
Go to the next section 8.9.8 Creating Ranges in a Range File or return to 8.9 Range Files.

8.9.8 Creating Ranges in a Range File

To quickly create a number of lines of ranges in a Range file, when a **Range File** panel is displayed, click RB in the blank cell at the top left hand corner of the grid title area (to the left of the **From** text) to bring up the **All Cells** choices and select **Populate Range.**



Selecting ***Populate Range*** brings up the **Populate Range** panel which is used to insert a number of range lines into the grid of the **Range File** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
From	real		

the lower value for the generated range

To real

the upper value for the generated range

Interval measure box

the interval between values for the generated range

Start colour colour box available colours

the start colour for the generated range. The colours are interpolated in RGB between the **Start colour** *and the* **End colour**.

End colour colour box available colours

the end colour for the generated range. The colours are interpolated in RGB between the **Start colour** *and the* **End colour**.

Skip results on non selected row tick box not ticked

Populate button

generate the defined grid lines and insert them into the grid. The new lines will be inserted at the highlighted grid lines and can replace any highlighted lines in the grid.

The ***Populate Range*** panel can be used a number of times to generate range lines in various colours.

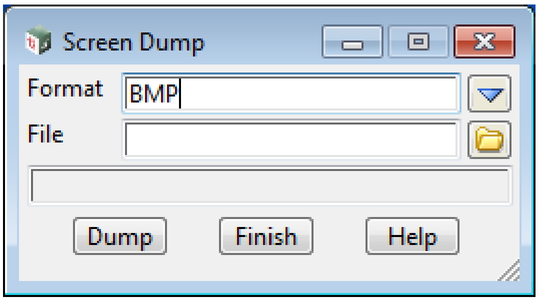
Return to 8.9 Range Files or continue to the next section 8.10 Screen Dump.

8.10 Screen Dump

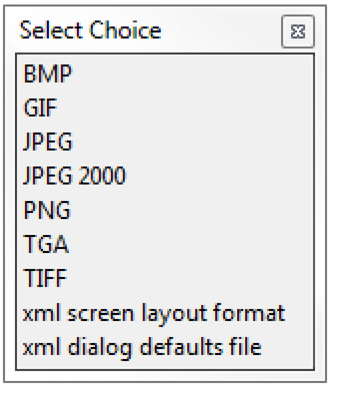
Position of option on menu: **File I/O =>Screen dump**

The 12d Model screen can be dumped to disk in a variety of images formats, or as a screen layout file.

On selecting the **Screen dump** option, the **Screen Dump** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Format	choice box	BMP	

format to write the screen dump out in.

File	file box		*.bmp files

file to write the screen. dump to.

Dump	button		

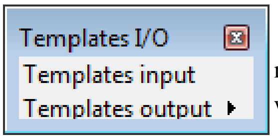
after selecting this button, the screen/window is dumped out in the selected format.

8.11 Templates

Position of menu: **File I/O =>Templates**

The option under **Template** read and write Design Template files.

The **Templates** walk-right menu is



read in a template file
write out template files

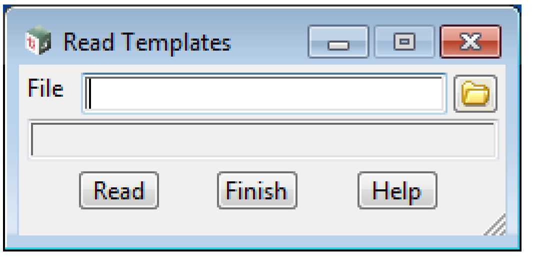
For *Templates input*, go to 8.11.1 Templates Input
Templates output 8.11.2 Templates Output

8.11.1 Templates Input

Position of option on menu: **File I/O =>Templates =>Templates Input**

This option is used to read in templates from files in the special 12d Model template format.

On selecting the **Templates input** option, the **Read Templates** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
File	input		*.tpl files

name of the 12d template file to be read in

Read	button		

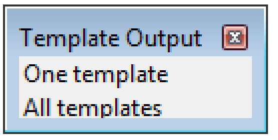
read the data in.

Note: If a template already exists, then the template in the file is ignored.

8.11.2 Templates Output

Position of menu: **File I/O =>Templates =>Templates Output**

The **templates output** option writes out one or all templates in the 12d Model template format. The **Output** walk-right menu is



For the option One template, go to 8.11.3 One Template

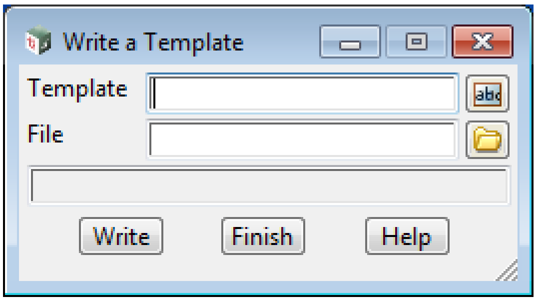
All templates 8.11.4 All Templates

8.11.3 One Template

Position of option on menu: **File I/O =>Templates =>Templates Output =>One Template**

This option is for writing a single template out in the special 12d template format.

On selecting the **One template** option, the **Write a Template** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Template	input		available templates

name of the template to be written out.

File	input		

name of the file to write the template to.

Write	button		

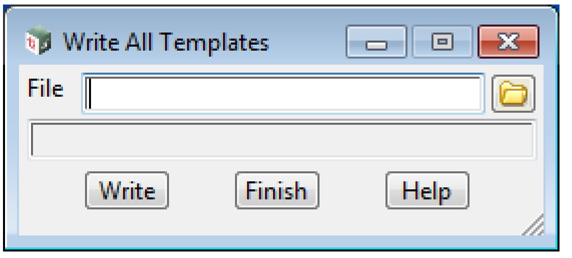
after selecting this button, the template given in the **template** *field will be written out to the file with the name given in the* **file** *field.*

8.11.4 All Templates

Position of option on menu: **File I/O =>Templates =>Templates output =>All templates**

This option is for writing all template out in the special 12d Model template format.

On selecting the **All templates** option, the **Write All Templates** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
File	input		

name of the file to write all the templates to.

Write	button		

all the templates will be written out to the file with the name given in the **file** *field.*

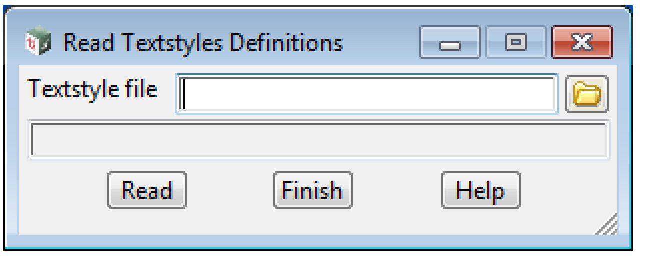
8.12 Textstyle Input

Position of option on menu: **File I/O =>Textstyle Input**

When 12d Model starts up, a textstyles definition file can be automatically read in. The textstyles definitions file defines the textstyles and fonts used in the project.

It is possible to redefine the textstyles by reading in another textstyles definitions file using the **Textstyle input** option.

On selecting the **Textstyle input** option, the **Read Textstyles Definitions** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Textstyle file	input		*.tsf

the name of the file to read the textstyle information from.

Read	button		

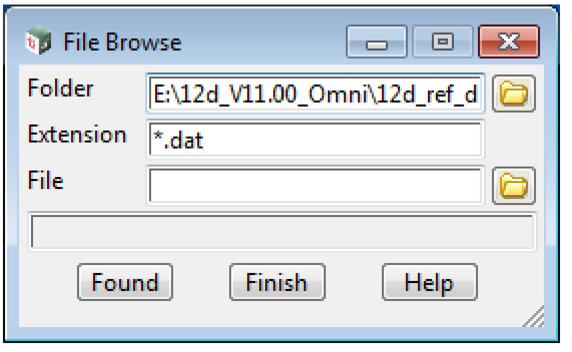
after selecting this button, the textstyles definitions file will be read in.

8.13 Edit a File

Position of option on menu: **File I/O =>Edit**

Files can be searched for and then displayed in the editor using the browse option.

Selecting **Edit** brings up the **File Browse** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Folder	input	current folder	Microsoft browser

folder to search for files with the extension given in the **extension** *field.*

Extension	input	*.dat	

wild cards and extension to limit the search of files. An <enter> needs to be typed after changing the extension.

File	input		files satisfying the **folder** and **extension** fields.

select the file to be displayed in the editor.

Found	button		

display in the editor the file of the name given in the **file** *field.*

Note

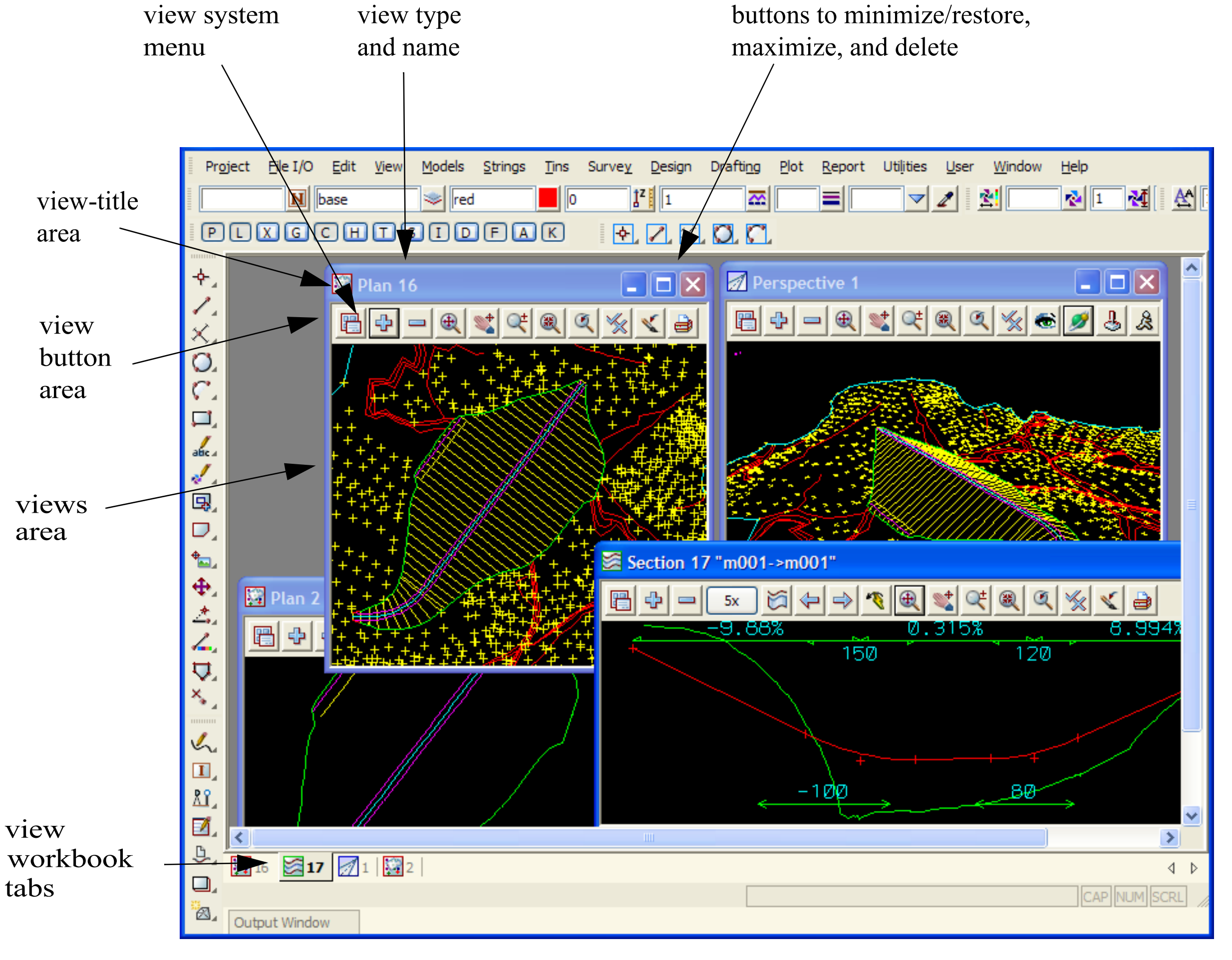
The **editor** is pointed to by the environment variable EDITOR_4D (see Appendix A).

9 Menus on Views

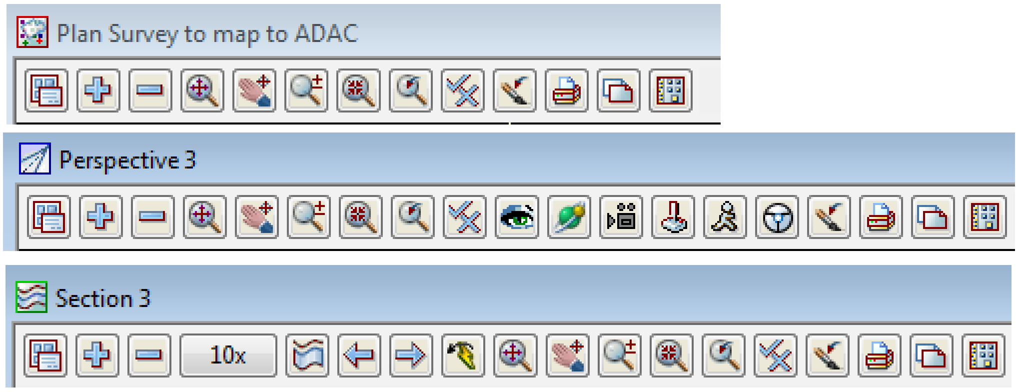
Views are the screen drawing areas for 12d Model and come in three flavours - plan, perspective and section. Views can be created and deleted as required by the user and there is no limit to the number of views on the screen. The views can overlap and be minimized.

Each view has a unique name of up to two hundred characters.

Each view has a *view title area* which is used to display the view name and a *view button area* which displays the view buttons.



view buttons



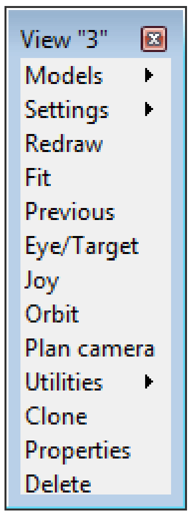
The *View Buttons* are documented in the section 9.5 View Buttons

If the **Menu** button is selected in the *view button area* of any view, or if RB is clicked in the *view-title area* or in the *view-button area*, a new menu called the **View** menu appears.

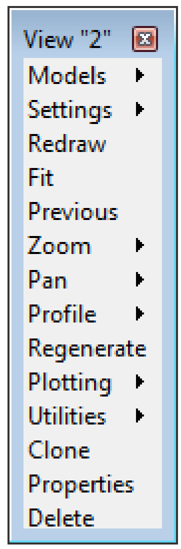
Because of the differences between plan, perspective and section views, the options on the **View** menu vary for each view type. The plan, perspective and section **View** menus are

Plan View Menu

Perspective View Menu



Section View Menu



The **View** menu is removed by selecting the **[X]** button or if the **View** menu hasn't been moved, by simply clicking **RB** again in the *view-title* or *view button* area. If the **View** menu has been moved, clicking **RB** will warp the cursor to the moved View menu.

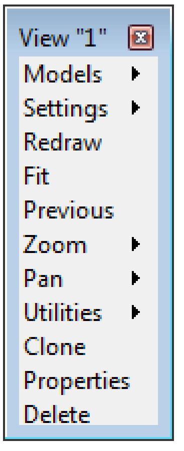
For further documentation on the items on each menu, please go to:

9.1 Plan View Menu
9.2 Perspective View Menu
9.3 Section View Menu

9.1 Plan View Menu

Position of menu: **Plan View Menu** **View**

The **Plan view** menu is



For the options:

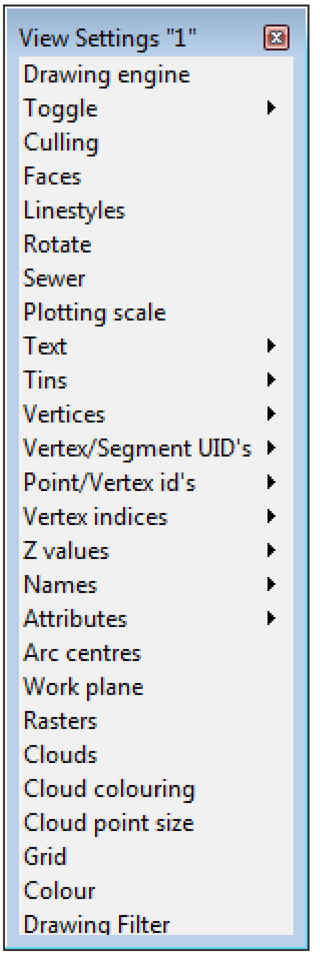
Models go to	9.4.1 Model Ops
Settings	9.1.1 Plan View Settings
Redraw	9.4.2 Redraw
Fit	9.4.3 Fit
Previous	9.4.4 Previous
Zoom	9.1.2 Zoom
Pan	9.1.3 Pan
Utilities	9.1.4 Plan Utilities
Clone	9.4.5 Clone
Properties	9.4.6 Properties
Delete	9.4.7 Delete

9.1.1 Plan View Settings

Position of menu: **Plan View Menu** **View => Settings**

If the **Settings** option is picked rather than moving onto the walking right, then the **toggle** menu from the **Toggle** walk-right menu is displayed on the screen. The **Toggle** menu will be described in the next section.

The **View Settings** walk-right menu for the plan view is



For the option:

Drawing engine go to	9.1.1.1 Drawing Engine
Drawing density	
Toggle	9.1.1.2 Plan Toggle
Culling	9.1.1.3 Culling Plan View
Faces	9.1.1.4 Face Flags for View
Linestyles	9.1.1.5 Linestyles for Plan View
Rotate	9.1.1.6 Rotate Plan View
Sewer	9.1.1.7 Sewer Annotations for Plan View
Plotting scale	9.1.1.8 Plan Plotting Scale
Text	9.1.1.9 Text
Tins	9.1.1.10 Tins
Vertices	9.1.1.11 Vertices
Vertex/Segment UID's	9.1.1.12 Vertex/Segment UID's
Point/Vertex id's	9.1.1.13 Point/Vertex Ids

Vertex indices	9.1.1.14 Vertex Indices
Z values	9.1.1.15 Z Values
Names	9.1.1.16 Names
Attributes	9.1.1.17 Attributes
Arc centres	9.1.1.18 Arc Centres for Plan View
Work plane	9.1.1.19 Work Plane
Rasters	9.1.1.20 Draw Rasters for Plan View
Clouds	9.1.1.21 Clouds
Cloud colouring	
Cloud point size	
Grid	9.1.1.22 Grid on View
Colour	9.1.1.23 View Background Colour
Drawing filter	9.1.1.24 View Drawing Filter

9.1.1.1 Drawing Engine

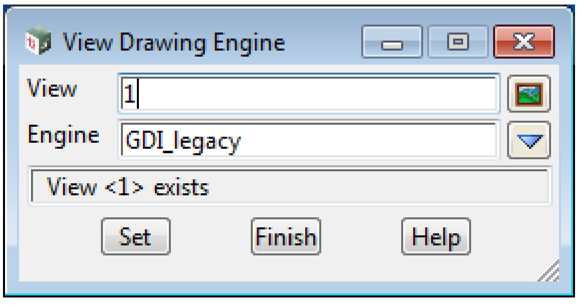
Position of option on menu: **Plan View Menu** **View =>Settings =>Drawing engine**

Position of option on menu: **Perspective View Menu** **View =>Settings =>Drawing engine**

The **Drawing engine** option allows the type of engine that draws the information on a perspective view to be changed.

You can draw in GDI or OpenGL drawing for Plan and Perspective views, and there are choices for the GDI or OpenGL engines,

Selecting **Drawing engine** brings up the **View Drawing Engine** panel.

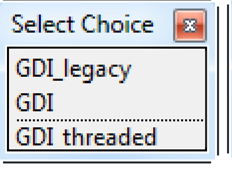


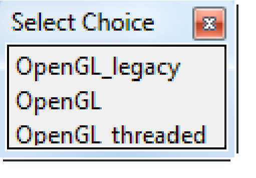
The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box	current view	available views

view to set the drawing engine for.

Engine	choice box	GDI_legacy	





For views using GDI drawing

*if **GDI_legacy**, drawing in GDI as in V10.*

***GDI**, some improvements over GDI_Legacy*

***GDI threaded**, allows <Escape> to stop view drawing, and update on where the drawing is up to, it avoids application not responding (TDR), stops endless looping when an application takes too long, and better cooperating with GPU's.*

For views using OpenGL drawing

*if **OpenGL_legacy**, drawing in OpenGL as in V10.*

***OpenGL**, some improvements over OpenGL_Legacy*

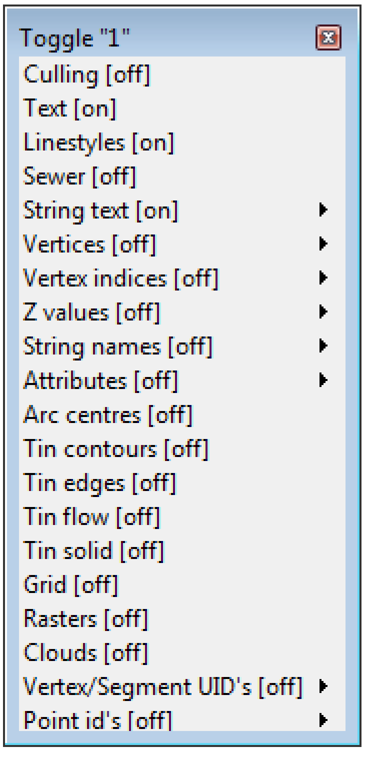
***OpenGL threaded**, allows <Escape> to stop view drawing, and update on where the drawing is up to, it avoids application not responding (TDR), stops endless looping when an application takes too long, and better cooperating with GPU's.*

Return to 9.1.1 Plan View Settings or 9.2.1 Perspective View Settings or 9 Menus on Views.

9.1.1.2 Plan Toggle

Position of menu: **Plan View Menu** **View => Settings =>Toggle**

The **Toggle** walk right brings up the **Toggle** plan view menu.



Selecting any options from this menu will toggle the option on/off.

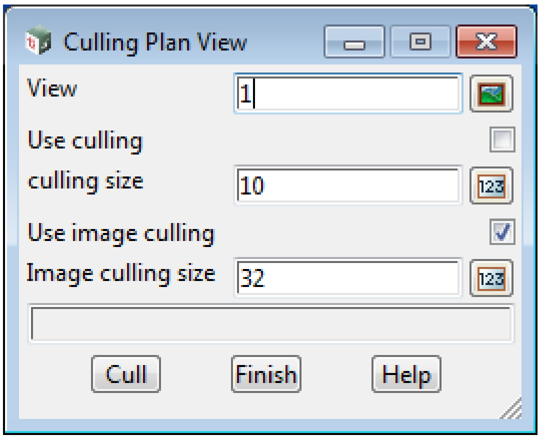
Continue to 9.1.1.3 Culling Plan View or return to 9.1.4 Plan Utilities or 9.1.1 Plan View Settings.

9.1.1.3 Culling Plan View

Position of option on menu: **Plan View Menu** **View =>Settings => Culling**

The **Culling** option is used to suppress the drawing of strings whose on-screen extent is less than a user defined pixel size.

Selecting **Culling** raises the **Culling Plan View** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views

name of the view to set cutting for.

Use culling	tick box		

*if **ticked**, a string is **not** drawn on the plan view whenever the string's extent box when drawn on the view would be smaller than the culling size given in the **culling size** field.*

Culling size	input	10	

pixel size used for culling

Use image culling	tick box		
Image culling size			
Cull	button		

*record the culling size given in the **culling field**. If the **use culling** field is set to tick, the recorded culling size will be used whenever the plan view is redrawn.*

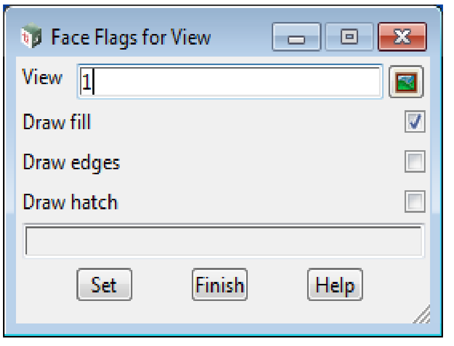
Continue to 9.1.1.4 Face Flags for View or return to 9.1.4 Plan Utilities or 9.1.1 Plan View Settings.

9.1.1.4 Face Flags for View

Position of option on menu: **Plan View Menu** **View =>Settings => Faces**

The **Faces** option allows the user to specify how faces are displayed in the plan view and on any plan view plots.

Selecting **Faces** fires up the **Face Flags for View** panel.



The fields and buttons have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views

name of the view to set for drawing faces.

Draw fill	tick box	tick	

if ticked, all faces in the view are drawn in their fill colour.

Draw edges	tick box		

if ticked, all face edges in the view are drawn.

Draw hatch	tick box		

if ticked, all faces in the view are drawn in their hatch pattern.

Set	button		

set the **draw fill/edges/hatch** *fields to the value in the panel fields. The plan view is then redrawn using this value.*

Continue to 9.1.1.5 Linestyles for Plan View or return to 9.1.4 Plan Utilities or 9.1.1 Plan View Settings.

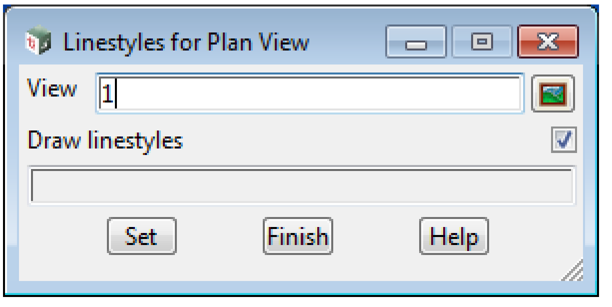
9.1.1.5 Linestyles for Plan View

Position of option on menu: **Plan View Menu** **View => Settings => Linestyles**

The **Linestyles** option allows the user to specify how linestyles (**styles**) are displayed in the plan view and on any plan view plots.

If linestyles are not used for drawing in a view then all strings are drawn according to their breakline type. That is, line strings are drawn with solid lines for each line in the string and point strings with only crosses at the string points. This style is the default style, **1**.

Selecting **Linestyles** fires up the **Linestyles for View** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views

name of the view to modify draw linestyle flag for.

Draw linestyles	tick box	tick	

if ticked, all strings in the view are drawn with linestyles.
If not ticked, all lines strings are drawn as solid lines and point strings with crosses at their vertices.

Set	button		

set the **draw linestyles** *field to the value in the panel field. The plan view is then redrawn using this value.*

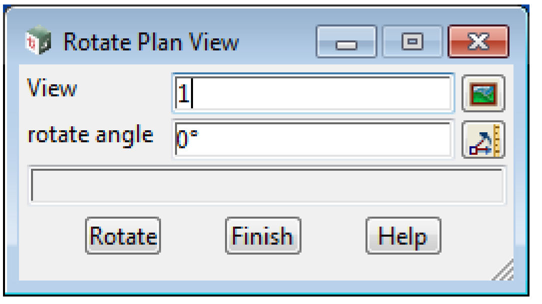
Continue to 9.1.1.6 Rotate Plan View or return to 9.1.4 Plan Utilities or 9.1.1 Plan View Settings.

9.1.1.6 Rotate Plan View

Position of option on menu: **Plan View Menu** **View => Settings => Rotate**

The **rotate** option is used to rotate a plan view through a user supplied angle about the centre point of the plan view.

After selecting the **Rotate** option, the **Rotate Plan View** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views

name of the view to set rotation angle for.

Rotate angle	input	0	angles

angle (in degrees) to rotate the view to.

Rotate	button		

rotate the view about the view centre point until the angle the view makes with the positive x axis is the angle given in the **rotate angle** *field.*

Continue to 9.1.1.7 Sewer Annotations for Plan View or return to 9.1.4 Plan Utilities or 9.1.1 Plan View Settings.

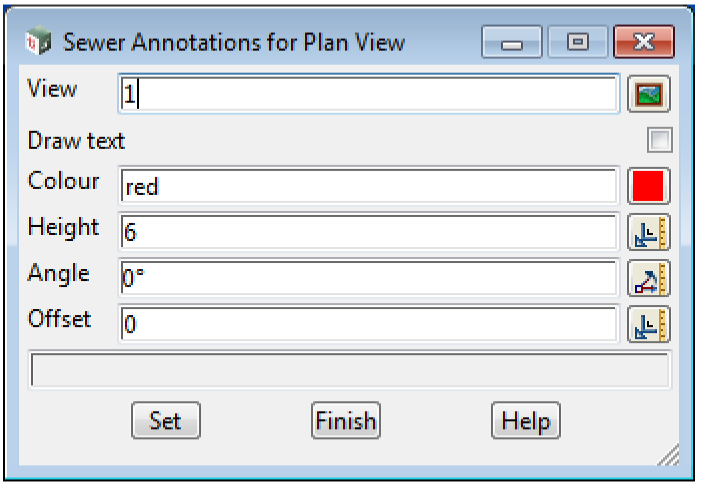
9.1.1.7 Sewer Annotations for Plan View

Position of option on menu: **Plan View Menu** **View => Settings => Sewer**

This is part of the optional sewer module.

The **Sewer** option allows the user to specify how sewer text is displayed in the plan view and on any plan view plots.

Selecting **sewer** fires up the **Sewer Annotations for Plan View** panel.



The fields and buttons have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views
name of the view to set sewer annotation information for.			
Draw text	tick box		
*if **ticked**, sewer text annotation will be drawn for any sewer strings in the plan view.*			
Colour	input	red	available colours
the colour for any sewer text annotation.			
Height	input	default text height	
the height for any sewer text annotation.			
Angle	input	0	
the angle for any sewer text annotation.			
Offset	input	0	
the offset for any sewer text annotation.			
Set	button		
set the values in the panel fields and then redraw the view.			

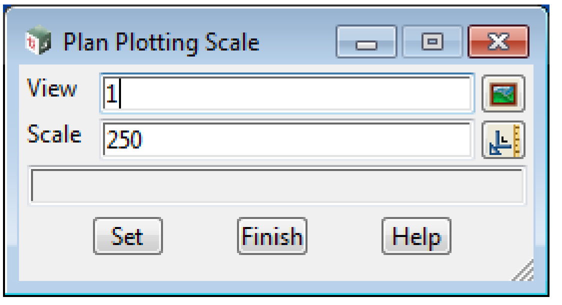
Continue to 9.1.1.8 Plan Plotting Scale or return to 9.1.4 Plan Utilities or 9.1.1 Plan View Settings.

9.1.1.8 Plan Plotting Scale

Position of option on menu: **Plan View Menu** **View => Settings => Plotting scale**

The **Plotting scale** option is used to set a scale for the plan view which is used to determine the size to draw any *paper* text on the plan view.

After selecting **Plotting scale**, the **Plan Plotting Scale** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views

name of the view to set the scale for.

Scale	input	250	

"1 in" to use to calculate a drawing size for any paper text on the view.

Set	button		

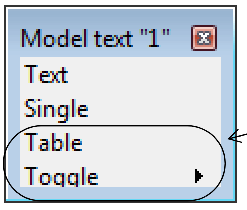
set the plotting scale for the view.

Continue to 9.1.1.9 Text or return to 9.1.4 Plan Utilities or 9.1.1 Plan View Settings.

9.1.1.9 Text

Position of menu: **Plan View Menu** **View => Settings =>Text**

The **text** walk-right menu is



These items are only on the menu if PLAN_TABLE_SETTNGS_4D is non zero

The options on the **Text** menu allows

(a) **Text**

If the text is defined in world units, the user can specify whether the text is displayed in **full**, **quick mode** or **not** drawn **at all** depending on how the text height would be as pixels on the screen.

This setting also applies to the Z-values and Vertex ids controlled by Plan View Settings.

(a) **Single, Table, Toggle**

If the text is defined in pixel units, then when the screen scale is such that the height of the text in world units is too large, then the text is not drawn.

This stops the drawing of *pixel* text when its displaying world size is large in proportion to other world units. For example, this happens to pixel text when zooming out.

Hence the options on the **Text** menu tries to control the drawing of ***both*** pixel and world text in models on the view so that the text stops displaying in the view as the user *zooms out*

For more information on each option on the menu, go to

Text 9.1.1.9.1 Plan Text
Single, Table and Toggle 9.1.1.9.2 Single, Table and Toggle for String Text on a Plan View

9.1.1.9.1 Plan Text

Position of option on menu: **Plan View Menu** **View => Settings => Text => Text**

The **Text** option allows the user to specify how text is displayed in the plan view and on any plan view plots.

For world text, there are also two text threshold values (***quick*** and ***none***) designed for use with world text. When the *screen* size of text in pixels drops below the thresholds, the mode of display is automatically adjusted.

Hence world text can be drawn either in

(a) full

(b) **quick mode** which consists of the top half of a rectangle indicating the height and the length of the text.

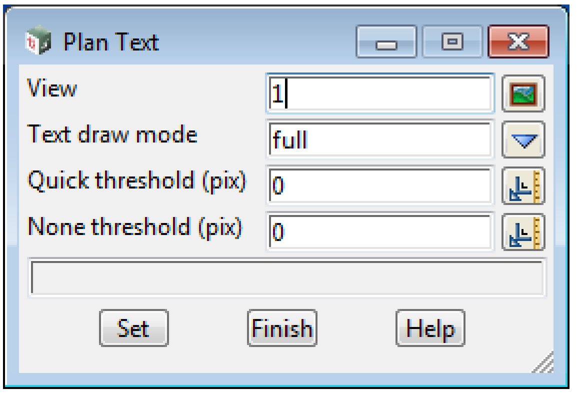
or

(c) **not** drawn **at all**.

So when the *screen* size of **world** text gets small, the text can automatically be drawn as **quick** text or even **not** displayed **at all**.

This setting also applies to the Z values and Vertex Ids controlled by View Settings.

Selecting **Text** brings up the **Plan Text** panel.



The fields and buttons used in this panel have the following functions.

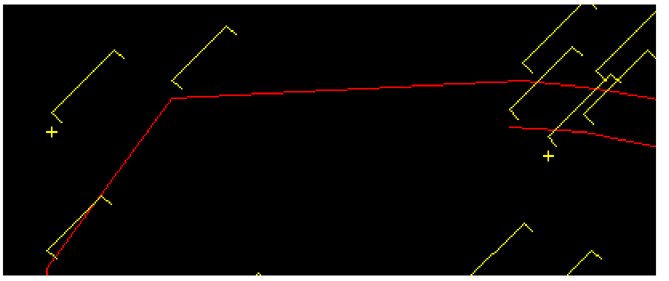
Field Description	Type	Defaults	Pop-Up
View	view box	current view	available views

name of the view to modify text drawing flags for.

Text draw mode	choice box	full	full, quick, none

*if **full**, all text in the plan view is tested against the **Quick threshold** and **None threshold** modes and if neither of them apply, the text is drawn in full.*

*If **quick**, all text in the plan view is tested against the **None threshold** modes and if it does not apply, the text characters are not drawn and the whole text strings is replaced by a three sided box to indicate where the text is.*



*If **none**, nothing is displayed on the plan view for text.*

Quick threshold (pix) input 4.5

*If **Text draw mode** is **none** then no text is drawn in the plan view.*

*If **Text draw mode** is **full** or **quick** then when text is displayed on the screen and the screen size of the displayed text in pixels would be below the **Quick threshold** value but above the None threshold, then the text is drawn in **quick** mode.*

None threshold (pix) input 2

*If **Text draw mode** is **none** then no text is drawn in the plan view.*

*If **Text draw mode** is **full** or **quick**, then when text is displayed on the screen and the screen size of the displayed text in pixels would be below the **None threshold** value, then the text is **not drawn**.*

Set button

set the text draw mode and thresholds to the values given in the panel fields. The plan view is then redrawn using the new values.

Important Note

The **Text** =>**Text** option applies to *all* text on the view. That is, for any text from strings in models on the view, or for the z-values and vertex ids text controlled by view settings (view text).

The **Single** and **Table** options of **Text** are *only* for text in models. There are **Single** and **Table** options for view text under the **Vertex no.s** and **Z values Settings** options.

Continue to 9.1.1.9.2 Single, Table and Toggle for String Text on a Plan View or return to 9.1.1.2 Plan Toggle or 9.1.1 Plan View Settings.

9.1.1.9.2 Single, Table and Toggle for String Text on a Plan View

The text for text strings, and the vertex text and segment text for super strings is known as **String Text**.

Plan View =>Settings =>Text has options **Single**, **Table** and **Toggle** for controlling the display of String Text and allow the user to specify whether **String Text** is

(a) not drawn at all

or

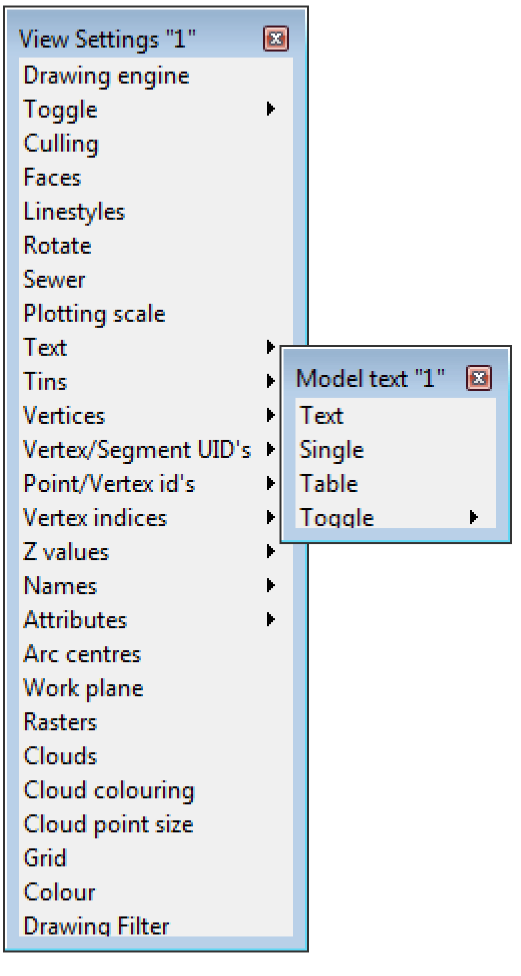
(b) if set to draw, then the *String Text* is only drawn if the equivalent world size of the text/vertex is not too large. This stops the drawing of *the String Text* when the displaying world size is large in proportion to other world units (e.g. when zooming out).

Hence the **String Text** options try to control the drawing of values/vertices on the view so that the **String Text stop displaying** as the **user *zooms out***.

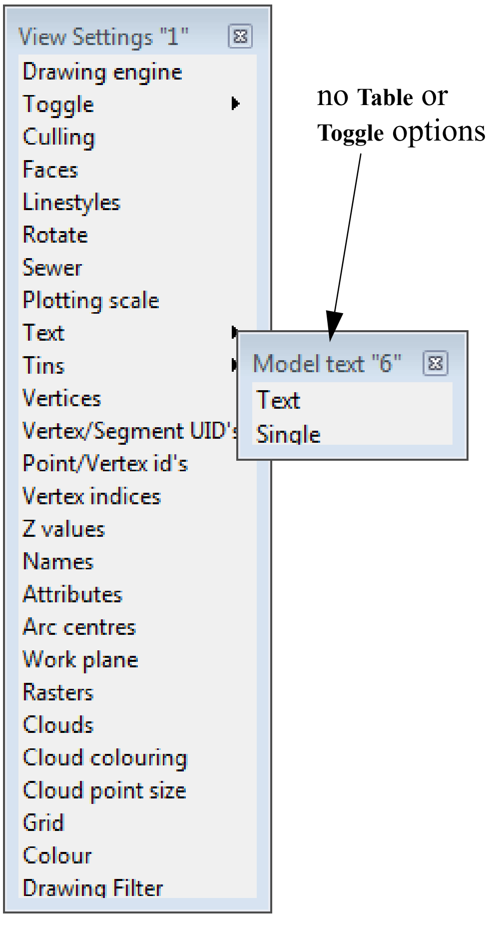
There is also an environment variable PLAN_TABLE_SETTINGS_4D that specifies whether the **Text** options draw the ***String Text*** for strings for ***all models*** on the view, or if the drawing of the *String Text* can be controlled for ***individual models***.

In either

PLAN_TABLE_SETTINGS_4D = 1
Models can be individually controlled



PLAN_TABLE_SETTINGS_4D = 0
Models can't be individually controlled



For both values of PLAN_TABLE_SETTINGS_4D there is also a there is a **Toggle** button on the **Plan View** that brings up the **Toggle** menu to make it easy to toggle the drawing of the ***String Text*** **on** and **off**. See 9.6.4 Toggle Menu for Values Text, Crosses and Text on Plan Views.

The options ***Single*** and ***Table*** tor the two cases of PLAN_TABLE_SETTINGS_4D documented in.

9.6 Displaying Values Text, Vertices and Text on a Plan View (the default)

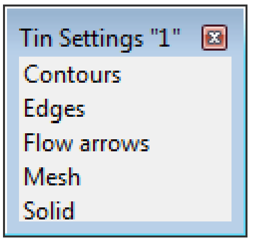
9.6.5 Env Var PLAN_TABLE_SETTINGS_4D Set to 0:

Continue to 9.1.1.10 Tins or return to 9.1.1 Plan View Settings.

9.1.1.10 Tins

Position of menu: **Plan View Menu** **View => Settings => Tins**

The options on the **Tins** walk-right menu control the display of tins on the view.The **Tins** walk-right menu is



For the options:

Contours	9.1.1.10.1 Tin Draw Contours for View
Edges	9.1.1.10.2 Tin Draw Edges for View
Flow arrows	9.1.1.10.3 Tin Draw Flow Arrows for View
Mesh	9.1.1.10.4 Tin Draw Mesh for View
Solid	9.1.1.10.5 Tin Draw Solid for View
Tuflow	

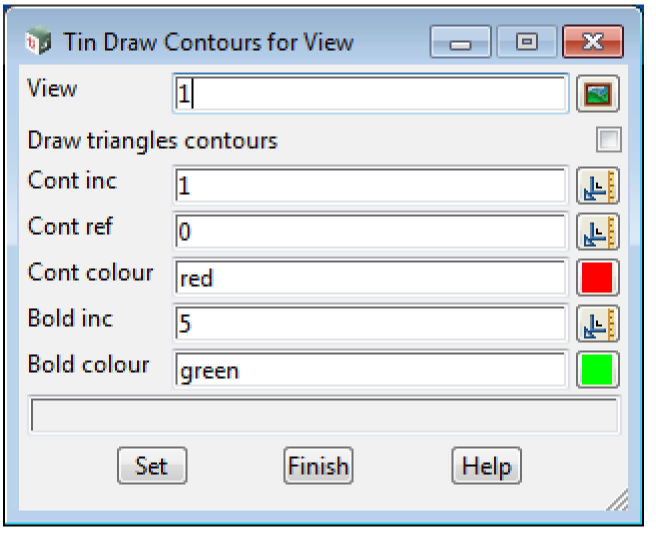
9.1.1.10.1 Tin Draw Contours for View

Position of option on menu: **Plan View Menu** **View => Settings => Tins => Contours**

Position of option on menu: **Perspective View Menu** **View => Settings => Tins => Contours**

The **Contours** option defines contour and bold increments and colours, and also whether these contours are displayed for the triangles from any tins on the view. Because the contours are just drawn separately for each triangle, they are know as **quick** or **fast contours**.

Selecting **Contours** fires up the **Tin Draw Contours for View** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views

name of the view to modify fast contour drawing parameters for.

Draw triangles contours tick box

*if **ticked**, the contours for any triangles in any tins on the view are displayed.*

Cont inc input 1.0

increment between contoured values.

Cont ref input 0.0

reference value for the contour increments.

Cont colour input cyan available colours

colour of the contours

Bold inc input 5.0

increment for the bold contours. If the bold increment is blank or zero, then no bold contours are drawn. If the bold increment is non-zero, it must be an integer multiple of the contour increment

Bold colour input magenta available colours

colour of the bold contours

Set button

set the value in the panel and then redraw the plan view.

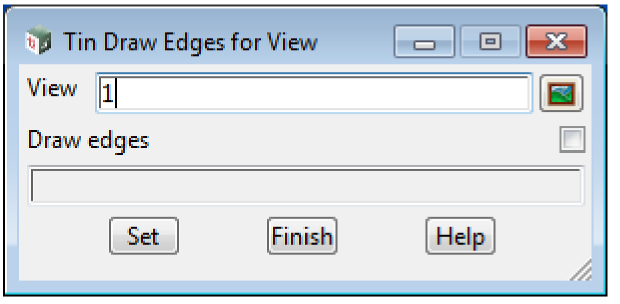
Continue to 9.1.1.10.1 Tin Draw Contours for View or return to 9.1.1.10 Tins or 9.1.1 Plan View Settings or 9.1 Plan View Menu.

9.1.1.10.2 Tin Draw Edges for View

Position of option on menu: **Plan View Menu** **View => Settings => Tins => Edges**

Position of option on menu: **Perspectives View Menu** **View => Settings => Tins => Edges**

The **Edges** option allows the user to specify whether the edges of triangles from any tins on the view are displayed. Selecting **Edges** fires up the **Tin Draw Edges for View** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views

name of the view to modify tin edge drawing flag for.

Draw triangles edges tick box

*if **ticked**, the edges of any triangles in any tins on the view are displayed.*

Set button

set the value and then redraw the plan view.

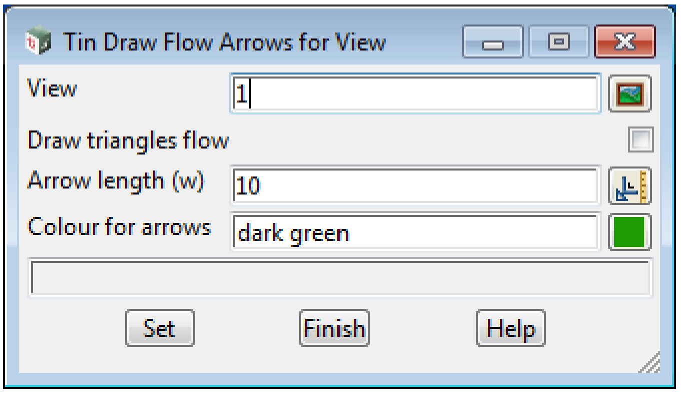
Continue to 9.1.1.10.2 Tin Draw Edges for View or return to 9.1.1.10 Tins or 9.1.1 Plan View Settings or 9.1 Plan View Menu.

9.1.1.10.3 Tin Draw Flow Arrows for View

Position of option on menu: Plan View Menu **View =>Settings =>Tins =>Flow arrows**

Position of option on menu: Perspective View Menu **View =>Settings =>Tins =>Flow arrows**

The **Flow arrows** option defines the colour and length of flow arrows, and also whether the flow arrows are displayed for the triangles from any tins on the view. Selecting **Flow arrows** fires up the **Tin Draw Flow Arrows for View** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views

name of the view to modify tin arrow drawing flag for.

Draw triangles flow arrows tick box

*if **ticked**, the flow arrows for any triangles in any tins on the view are displayed.*

Arrow length (w) input

length in world units to draw flow arrows.

Colour for arrows	input	cyan	available colours

colour for the arrows.

Set button

set the value in the panel and then redraw the plan view.

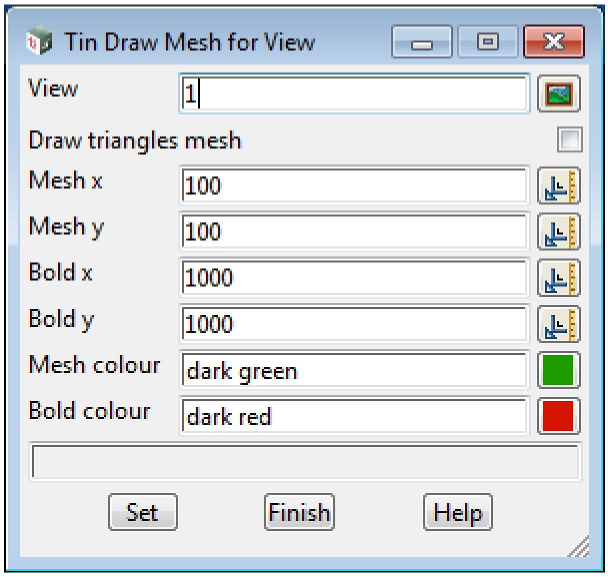
Continue to 9.1.1.10.3 Tin Draw Flow Arrows for View or return to 9.1.1.10 Tins or 9.1.1 Plan View Settings or 9.1 Plan View Menu.

9.1.1.10.4 Tin Draw Mesh for View

Position of option on menu: **Plan View Menu** **View => Settings => Tins => Mesh**

Position of option on menu: **Perspective View Menu** **View => Settings => Tins => Mesh**

The **mesh** option defines a rectangular mesh, and also whether the mesh is displayed for the triangles from any tins on the view. Selecting **Mesh** fires up the **Tin Draw Mesh for View** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views

name of the view to modify tin mesh drawing flag for.

Draw triangles mesh tick box

*if **ticked**, a rectangular mesh for any triangles in any tins on the view are displayed.*

Mesh x input 100

the distance between the x mesh lines. If this value is zero, the x mesh lines will not be drawn.

Mesh y input 100

the distance between the y mesh lines. If this value is zero, the y mesh lines will not be drawn.

Bold x input 100

the distance between the bold x mesh lines. If this value is zero, the bold x mesh lines will not be drawn.

Bold y input 100

the distance between the bold y mesh lines. If this value is zero, the bold y mesh lines will not be drawn.

Mesh colour input dark green available colours

colour that the mesh is drawn in.

Bold colour input dark green available colours

colour that the bold mesh is drawn in.

Set button

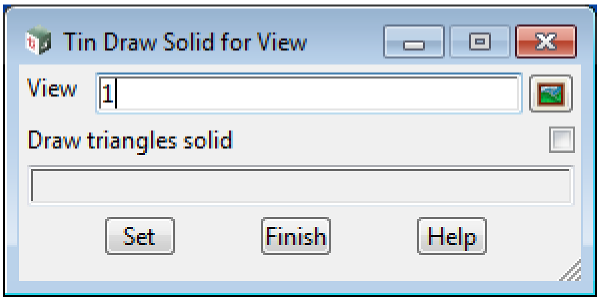
set the value in the panel and then redraw the plan view.

Continue to 9.1.1.10.4 Tin Draw Mesh for View or return to 9.1.1.10 Tins or 9.1.1 Plan View Settings or 9.1 Plan View Menu.

9.1.1.10.5 Tin Draw Solid for View

Position of option on menu: **Plan View Menu** **View => Settings => Tins => Solid**

The **solid** option allows the user to specify whether the triangles from any tins on the view are displayed as solid colour. Selecting **Solid** fires up the **Tin Draw Solid for View** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views

name of the view to modify tin solid drawing flag for.

Draw triangles solid	tick box		

*if **ticked**, any triangles in any tins on the view are displayed in solid colour.*

Set	button		

set the value and then redraw the plan view.

Return to 9.1.1.10 Tins or 9.1.1 Plan View Settings or 9.1 Plan View Menu.

9.1.1.11 Vertices

Position of menu: **Plan View Menu** **View => Settings => Vertices**

Vertices displays **crosses at the vertices of strings**. For a Text string, a cross is placed at the text justification position.

Plan View =>Settings has an option **Text** to draw ***String Text,*** an option **Vertices** to draw ***Crosses at*** all the **Vertices** in a string (often just called ***Vertices***) and the options ***Vertex/ Segment UIDs***, ***Point/Vertex Id's***, ***Vertex indices***, ***Z values***, ***Names*** and ***Attributes*** for labelling specific values from a string as text (for these options we call the **text** of the **value**, the **Values Text**.).

All these options work in a similar manner and are being documented together. For more information on each options, go to 9.6 Displaying Values Text, Vertices and Text on a Plan View.

Continue to 9.1.1.12 Vertex/Segment UID's or return to 9.1.1 Plan View Settings or 9.1 Plan View Menu.

9.1.1.12 Vertex/Segment UID's

Position of menu: **Plan View Menu** **View => Settings => Vertex/Segment UID's**

Vertex/Segment UID's displays the **Vertex UID's** any **Segment UIDs** for the vertices and segments of super strings.

Plan View =>Settings has an option **Text** to draw ***String Text,*** an option **Vertices** to draw ***Crosses at*** all the **Vertices** in a string (often just called ***Vertices***) and the options ***Vertex/ Segment UIDs***, ***Point/Vertex Id's***, ***Vertex indices***, ***Z values***, ***Names*** and ***Attributes*** for labelling specific values from a string as text (for these options we call the **text** of the **value**, the **Values Text**.).

All these options work in a similar manner and are being documented together. For more information on each options, go to 9.6 Displaying Values Text, Vertices and Text on a Plan View.

Continue to 9.1.1.13 Point/Vertex Ids or return to 9.1.1 Plan View Settings or 9.1 Plan View Menu.

9.1.1.13 Point/Vertex Ids

Position of menu: **Plan View Menu** **View => Settings => Point/Vertex id's**

Point/Vertex IDs displays the **Point/Vertex ID** (a point ID recorded for a vertex) for the vertices of super strings.

Note - Point/Vertex IDs are not to be confused with *Vertex Indices* which are simply the position of the vertex in the string.

Plan View =>Settings has an option **Text** to draw ***String Text,*** an option **Vertices** to draw ***Crosses at*** all the **Vertices** in a string (often just called ***Vertices***) and the options ***Vertex/ Segment UIDs***, ***Point/Vertex Id's***, ***Vertex indices***, ***Z values***, ***Names*** and ***Attributes*** for labelling specific values from a string as text (for these options we call the **text** of the **value**, the **Values Text**.).

All these options work in a similar manner and are being documented together. For more information on each options, go to 9.6 Displaying Values Text, Vertices and Text on a Plan View.

Continue to 9.1.1.14 Vertex Indices or return to 9.1.1 Plan View Settings or 9.1 Plan View Menu.

9.1.1.14 Vertex Indices

Position of menu: **Plan View Menu** **View => Settings => Vertex indices**

Vertex Indices displays the **Vertex Indices** (position of the vertex in the string) for all the vertices of strings.

Note - Vertex Indices are not to be confused with *Point/Vertex IDs* which may not even exist for a vertex in the string.

Plan View =>Settings has an option **Text** to draw ***String Text,*** an option **Vertices** to draw ***Crosses at*** all the **Vertices** in a string (often just called ***Vertices***) and the options ***Vertex/ Segment UIDs***, ***Point/Vertex Id's***, ***Vertex indices***, ***Z values***, ***Names*** and ***Attributes*** for labelling specific values from a string as text (for these options we call the **text** of the **value**, the **Values Text**.).

All these options work in a similar manner and are being documented together. For more information on each options, go to 9.6 Displaying Values Text, Vertices and Text on a Plan View.

Continue to 9.1.1.15 Z Values or return to 9.1.1 Plan View Settings or 9.1 Plan View Menu.

9.1.1.15 Z Values

Position of menu: **Plan View Menu** **View => Settings => Z values**

Z values displays the z coordinate to a user specified number of decimal places, for all the vertices of strings.

Plan View =>Settings has an option **Text** to draw ***String Text,*** an option **Vertices** to draw ***Crosses at*** all the **Vertices** in a string (often just called ***Vertices***) and the options ***Vertex/ Segment UIDs***, ***Point/Vertex Id's***, ***Vertex indices***, ***Z values***, ***Names*** and ***Attributes*** for labelling specific values from a string as text (for these options we call the **text** of the **value**, the **Values Text**.).

All these options work in a similar manner and are being documented together. For more information on each options, go to 9.6 Displaying Values Text, Vertices and Text on a Plan View.

Continue to 9.1.1.16 Names or return to 9.1.1 Plan View Settings or 9.1 Plan View Menu.

9.1.1.16 Names

Position of menu: **Plan View Menu** **View => Settings => Names**

Names displays the string name at each vertex of a string.

Plan View =>Settings has an option **Text** to draw ***String Text,*** an option **Vertices** to draw ***Crosses at*** all the **Vertices** in a string (often just called ***Vertices***) and the options ***Vertex/ Segment UIDs***, ***Point/Vertex Id's***, ***Vertex indices***, ***Z values***, ***Names*** and ***Attributes*** for labelling specific values from a string as text (for these options we call the **text** of the **value**, the **Values Text**.).

All these options work in a similar manner and are being documented together. For more information on each options, go to 9.6 Displaying Values Text, Vertices and Text on a Plan View.

Continue to 9.1.1.17 Attributes or return to 9.1.1 Plan View Settings or 9.1 Plan View Menu.

9.1.1.17 Attributes

Position of menu: **Plan View Menu** **View => Settings => Attributes**

Attributes displays the vertex attributes for all the vertices of super strings.

Plan View =>Settings has an option **Text** to draw ***String Text,*** an option **Vertices** to draw ***Crosses at*** all the **Vertices** in a string (often just called ***Vertices***) and the options ***Vertex/ Segment UIDs***, ***Point/Vertex Id's***, ***Vertex indices***, ***Z values***, ***Names*** and ***Attributes*** for labelling specific values from a string as text (for these options we call the **text** of the **value**, the **Values Text**.).

All these options work in a similar manner and are being documented together. For more information on each options, go to 9.6 Displaying Values Text, Vertices and Text on a Plan View.

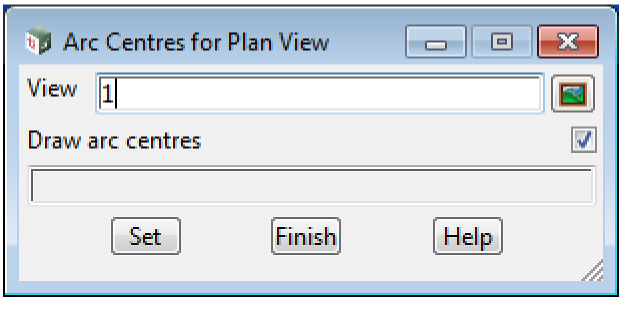
Continue to 9.1.1.18 Arc Centres for Plan View or return to 9.1.1 Plan View Settings or 9.1 Plan View Menu.

9.1.1.18 Arc Centres for Plan View

Position of option on menu: **Plan View Menu** **View => Settings => Arc centres**

The **arc centres** option allows the user to specify whether the centres of arcs and circles are displayed on the view.

Selecting **Arc centres** fires up the **Arc Centres for Plan View** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views

name of the view to modify arc centres flag for.

Draw arc centres	tick box	tick	

*if **ticked** is selected, any arcs or circles on the view will also have their centres drawn. Otherwise the centres are not displayed.*

Set	button		

set the draw arc centres mode to the value in the **draw arc centre** *tick box panel box. The plan view is then redrawn using the mode.*

Continue to 9.1.1.19 Work Plane or return to 9.1.1 Plan View Settings or 9.1 Plan View Menu.

9.1.1.19 Work Plane

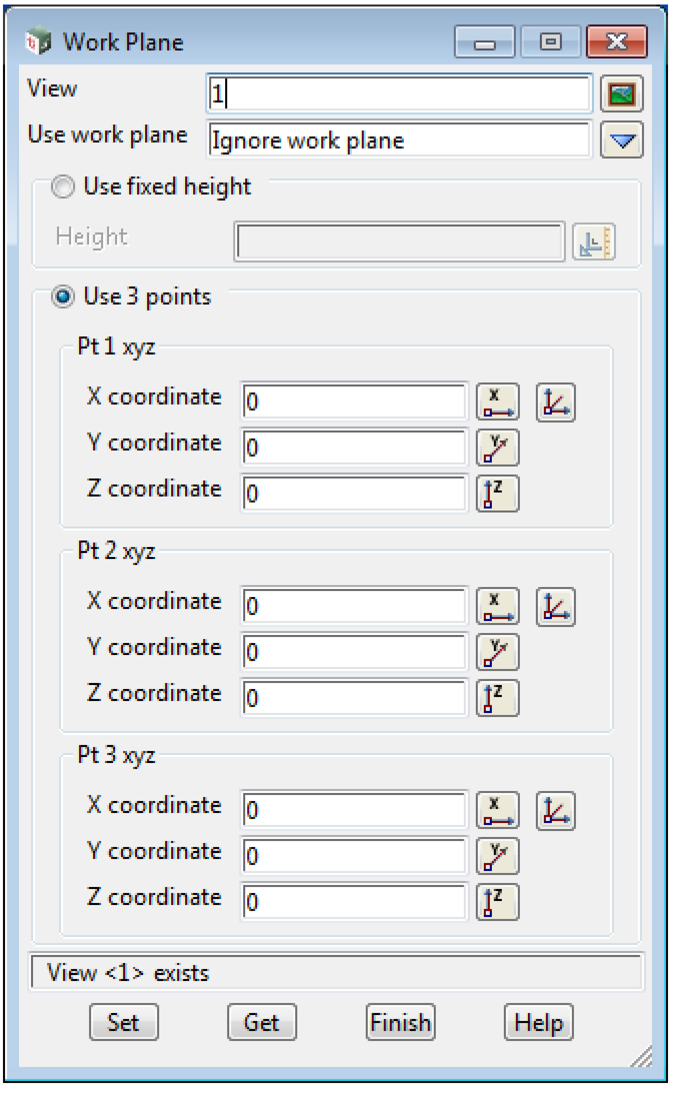
Position of option on menu: **Plan View Menu** **View => Settings => Work plane**

A plane not perpendicular to the (x,y) plane, can be used to define a unique z-value for any (x,y) co-ordinate pair. This fact is often invaluable when trying to defined three dimensional points using two-dimensional views.

In ***12d Model***, a **work plane** can be defined by giving three non-colinear points. If the work-plane is used in a plan or perspective view, then the (x,y,z) values are restricted to lie in the defined work plane.

The **Work plane** option is used to define and set work planes.

After selecting the **Work Plane** option, the **Work Plane** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box		available views

name of the view to get or set the work plane in.

User work plane	choice box	ignore work plane	ignore work plane

use work plane

the work plane is used or not depending on this field value.

Use fixed height tick box

if ticked, all the created points are given the height from the Height panel field, instead of the z-value from the plane.

Height Real value

if User fixed height is ticked, this is the height to use for all the created points instead of the z-value from the plane.

Pt 1/2/3 xyz input/output 0 0 0

points in the work plane

Set button

A plane is defined by the three points given in the **Pt xyz** *fields. This plane is set as the work plane for the view given by the* **view** *field. Although the work plane is set, it is not active unless the* **use work plane** *field is set to* ***use work plane****.*

Get button

get the three points of the defined work plane for the view given in the view field. The points are displayed in the **Pt xyz** *fields of the panel.*

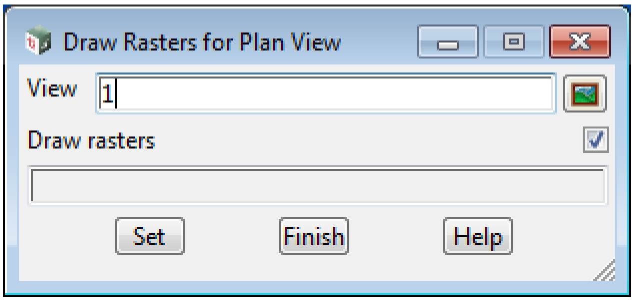
Continue to 9.1.1.20 Draw Rasters for Plan View or return to 9.1.1 Plan View Settings or 9.1 Plan View Menu.

9.1.1.20 Draw Rasters for Plan View

Position of option on menu: **Plan View Menu** **View => Settings => Rasters**

The **Rasters** option is used to turn on or off the drawing of any rasters in models on the view.

After selecting the **Rasters** option, the **Draw Rasters for Plan View** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box	current view	available views

name of the view to toggle the drawing of rasters on or off.

Field Description	Type	Defaults	Pop-Up
Draw rasters	tick box	ticked	

*if **ticked**, draw any rasters in any models on the view.*

Field Description	Type	Defaults	Pop-Up
Set	button		

set the draw rasters mode.

Continue to 9.1.1.21 Clouds or return to 9.1.1 Plan View Settings or 9.1 Plan View Menu.

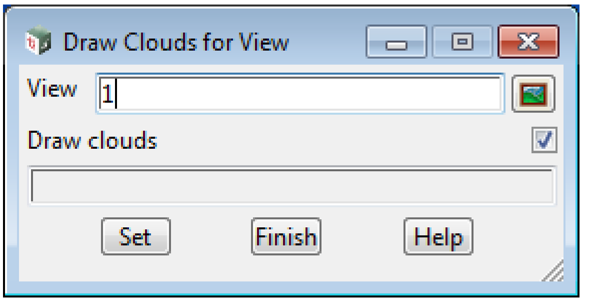
9.1.1.21 Clouds

Draw_Clouds_for_View**Position of option on menu:** **Plan View Menu** **View => Settings => Clouds**

Point clouds can consist of millions, or even billions, of individual points. The drawing of them on a plan view may slow things considerably.

The **Clouds** option is used to optionally draw point clouds in any models on the view.

After selecting the **Clouds** option, the **Draw Clouds for View** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box	current view	available views

name of the view to toggle the drawing of point clouds on or off.

Draw clouds	tick box	ticked

*if **ticked**, draw any point clouds in any models on the view.*
*if **not ticked**, don't draw any point clouds in any models on the view.*

Set	button

sets the draw clouds mode.

Continue to 9.1.1.22 Grid on View or return to 9.1.1 Plan View Settings or 9.1 Plan View Menu.

9.1.1.22 Grid on View

Position of option on menu: **Plan View Menu** **View => Settings => Grid**

Section View Menu **View => Settings => Grid**

Perspective View Menu **View => Settings => Grid**

A regular grid of x (East) and/or y (North) lines can be displayed in a view. The grid is set in a plane of constant z (the level of the grid). The x and y spacing for the grid lines can be set and labelled independently.

For a plan or section view, the grid covers the entire view. However, for a perspective view, the grid only covers the maximum extents of the models in the view.

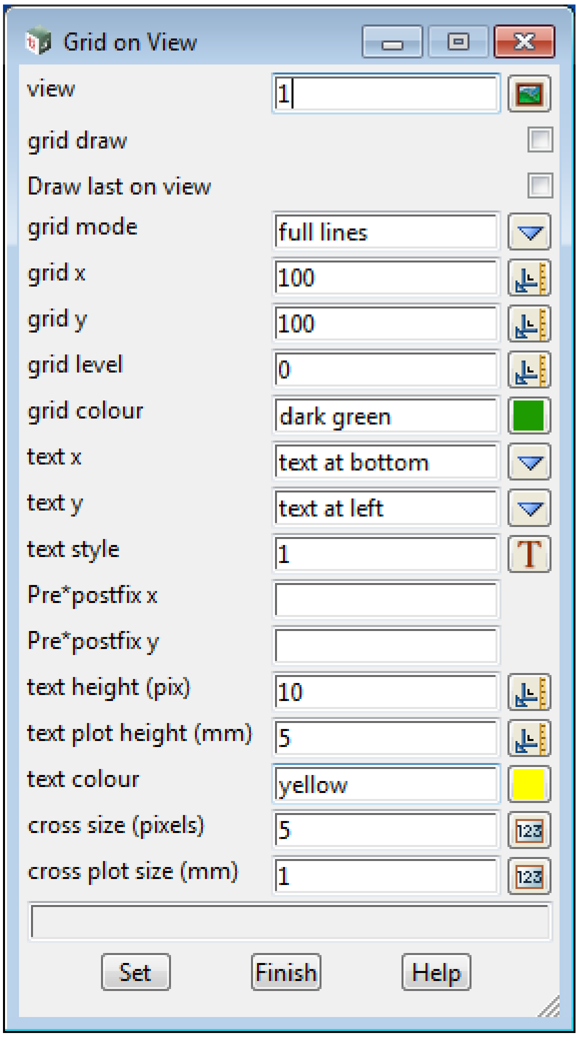
If the grid mode is set to

full lines solid lines are drawn at the grid x (East) and y (North) spacing
crosses crosses are drawn at the intersection of the grid x and y spacing

marks lines are drawn at the beginning/end of the grid at x and y spacing

marks and crosses marks and crosses are drawn

On selecting the option, the **Grid on View** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views

the name of the view to define a grid for.

Grid draw	tick box		

*if **ticked**, the grid is drawn according to parameters in the rest of the panel.*

Grid mode	input	full lines	full lines, crosses, marks marks and crosses

the style of the grid being drawn - full lines, crosses or marks.
*If **full lines**, then solid lines are drawn at the grid x and y spacing.*
*If **crosses**, then crosses of size **cross size** are drawn at the intersection of the grid x and y spacing.*
*If **marks**, then solid lines size **cross size** are drawn at the beginning/end of the grid at the x and y spacing.*
*If **marks and crosses**, then marks at the beginning/end of the grid at the x and y spacing, and crosses are drawn.*

Grid x	input	100	

the distance between the x (East) grid lines. If this value is zero, the x grid lines will not be drawn.

Grid y	input	100	

the distance between the y (North) grid lines. If this value is zero, the y grid lines will not be drawn.

Grid level	input	0	

the (x,y) grid is set in a plane of constant z. The **grid level** *is the value of the constant z. This will only be used for a perspective view.*

Grid colour	input	dark green	available colours

colour that the grid is drawn in.

Text x	input	text at bottom	text off, text at top, text at bottom, text at top and bottom

the x (East) grid lines can be labelled with their x value. If the grid lines are labelled, the labels can be drawn on the top, the bottom or both ends of the x-grid lines

Text y	input	text at left	text off, text at left, text at right, text at left and right

the y (North) grid lines can be labelled with their y value. If the grid lines are labelled, the labels can be drawn on the left, the right or both ends of the y-grid lines

Text style	input	1	

the text style of any grid labels.

Pre*postfix x	input		

*prefix/postfix (pre*post) to be applied to the value of x (East). If pretext only, just give the text. If post text is required, precede it by a *. For example E*m will place E before the x value and m after the number.*

Pre*postfix y	input		

*prefix/postfix (pre*post) to be applied to the value of y (North). If pretext only, just give the text. If post text is required, precede it by a *. For example N*m will place N before the y value and m after the number.*

Text height (pix)	input	10.0	

height in screen pixels of any grid labels.

Text plot height (mm)	input	10.0	

height in millimetres of any grid labels in a plot.

Text colour	input	yellow	available colours

colour that any grid labels are drawn in.

Cross size (pixels)	input	5.0	

size in screen pixels of grid crosses.

Cross plot size (mm)	input	1.0	

size in millimetres of grid crosses in a plot.

Set	button		

the grid parameters are set to the values given in the panel. The grid is then redrawn in the view.

Note - If the view name is selected from the pop-up, or a <enter> is done after the view name is entered into the view field, the data for the grid on the named view is placed in the panel fields.

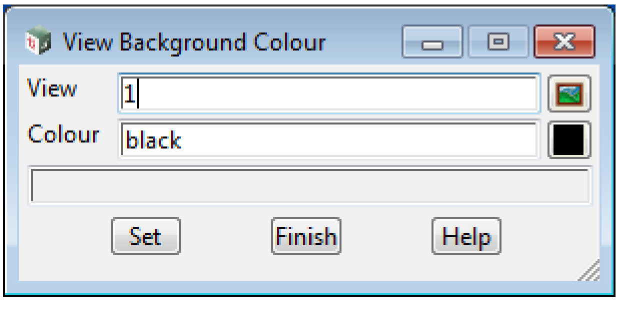
Continue to 9.1.1.23 View Background Colour or return to 9.1.1 Plan View Settings or 9.1 Plan View Menu.

9.1.1.23 View Background Colour

Position of option on menu: **Plan View Menu** **View => Settings => Colour**

The **Colour** option is used to set the background colour for the view.

After selecting the **Colour** option, the **View Background Colour** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views

name of the view to set the background colour for.

Colour	colour box	black	available colours

if ticked, draw any rasters in any models on the view.

Set	button		

set the background colour.

Continue to 9.1.1.24 View Drawing Filter or return to 9.1.1 Plan View Settings or 9.1 Plan View Menu.

9.1.1.24 View Drawing Filter

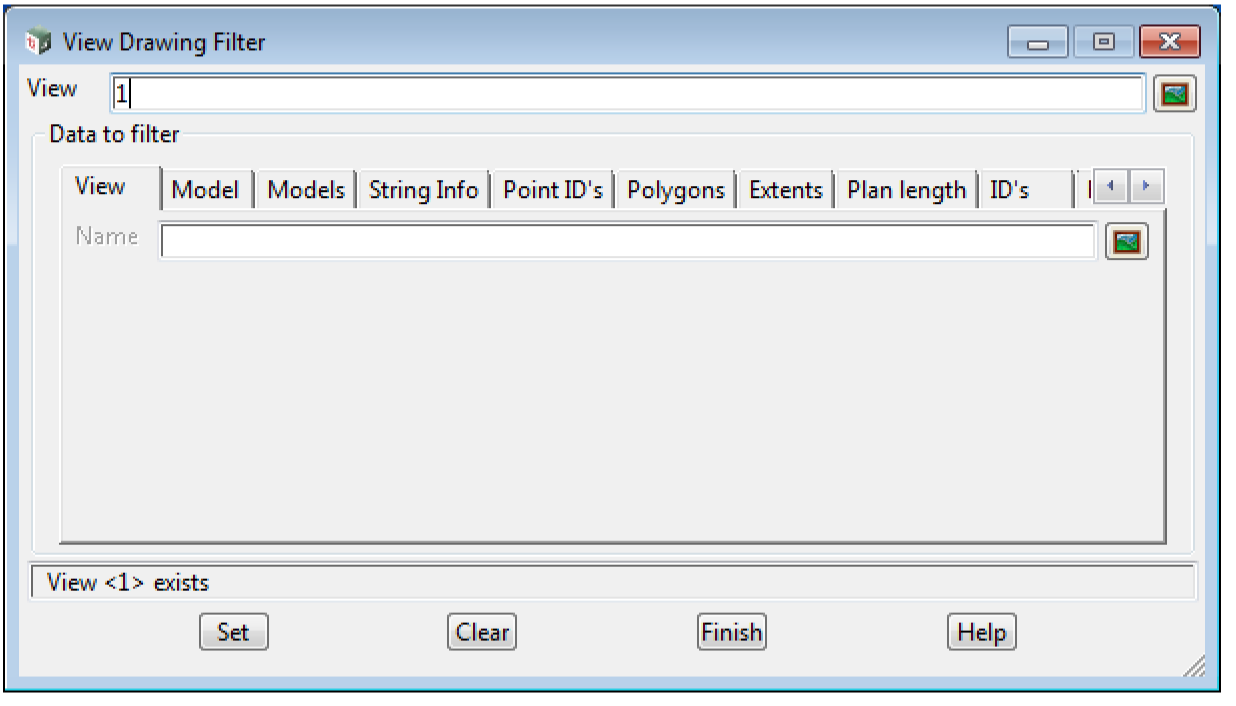
Position of option on menu: **Plan View Menu** **View => Settings => Drawing filter**

Position of option on menu: **Perspective View Menu** **View => Settings => Drawing filter**

The **Drawing filter** option is used to set a filter on the view so that only the data satisfying the filter is drawn.

NOTE: the filter is not saved with the project.

After selecting the **Drawing filter** option, the **View Drawing Filter** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box	current view	available views
name of the view to apply the drawing filter to.			
Data filter	view box	current view	available views
only the data satisfying the drawing filter is drawn on the view.			
Set	button		
apply the drawing filter to the view.			
Clear	button		
clear the drawing filter being applied to the view.			

Return to 9.1.1 Plan View Settings or 9.2.1 Perspective View Settings.

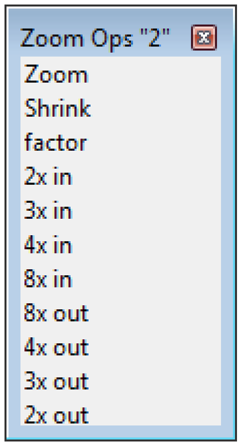
9.1.2 Zoom

Position of menu: **Plan View Menu** **View => Settings => Zoom**

Position of menu: **Section View Menu** **View => Settings => Zoom**

The zoom option operators in the same way for Plan and Section views.

The **zoom** option can be selected in three ways. By clicking LB on **zoom** on a view button area, selecting the **zoom option** itself by clicking LB when the zoom option is highlighted on the zoom menu, or by activating the zoom option's walk-right menu, **Zoom Ops**.



The zoom uses either a *dynamics zoom* or a *zoom box*

For more information on **zoom ops** go to 9.1.2.3 Zoom Ops menu

9.1.2.1 Dynamic Zoom

After selecting the **Zoom** option, *dynamic zoom* is selected by clicking MB or by typing **d**.

Then press LB at the point that is to stay put during the dynamic zoom and then move the mouse up to zoom in or move the mouse down to zoom out. Another point can be selected to dynamically zoom in or out about.

The dynamic zoom continues until either RB is clicked, or MB is clicked (or **d** typed) to place the zoom option into *zoom box* mode.

9.1.2.2 Zoom Box

Position of option on menu: **Plan View Menu** **View => Settings => Zoom =>Zoom**

After selecting the **zoom** option, the user can define the zoom box by

(a) pressing down LB at one corner of the box and dragging the mouse and releasing LB when at the other corner of the zoom box

or

(b) picking two points from any plan views (by clicking LB in a plan view for each point) to define the diagonally opposite points of the zoom box. Note that the two points **do not** have to be from the same plan view.

Next the user selects the view (by clicking LB in the view) which is to be redrawn using this rectangle as the new drawing limits.

Often the same view is chosen for both defining the rectangle and redrawing. However, the power of the option is that the area to be “zoomed in” or “blown up” can be selected from one view and the zoomed area displayed in another view. The other view may not only be different to the views used to defined the zoom box, but may also have different models attached to it.

It is a useful technique to have a master view displaying the entire area of interest in one view, and use it to define zoom-rectangles for use in viewing smaller regions of data in other plan views.

To cancel the zoom after the zoom operation has been selected, click RB.

Next step messages - sent to the screen message area

<View Zoom> [Select][][] select 1st corner of box - RB to cancel
<View Zoom> [Select][][] select 2nd corner of box - RB to cancel
<View Zoom> [Select][][] select destination view - RB to cancel

WARNING - this option can not be applied to a perspective view.

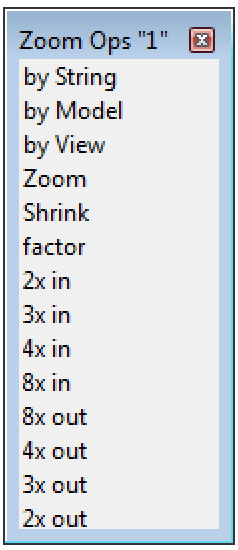
Please continue to the next section 9.1.2.3 Zoom Ops menu.

9.1.2.3 Zoom Ops menu

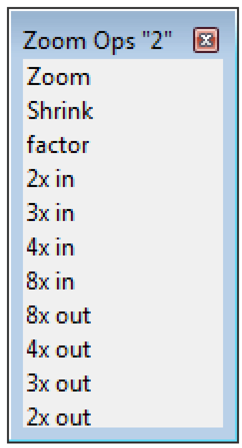
Position of menu: **Plan View Menu** **View => Settings => Zoom**

Position of menu: **Section View Menu** **View => Settings => Zoom**

The **Zoom** walk-right menu is



a string defines the extent of the view
a model defines the extent of the view
a view defines the extent of the view
two point zoom - see previous section
two point shrink
zoom by a user supplied factor
zoom into the view by factor of 2
zoom into the view by factor of 3
zoom into the view by factor of 4
zoom into the view by factor of 8
zoom out of the view by a factor of 8
zoom out of the view by a factor of 4
zoom out of the view by a factor of 3
zoom out of the view by a factor of 2



The **by String**, **by Model** and **by View** options use the x and y extents of the String, Model or View to define the display area for the view. These options will be described in more detail.

The option **zoom** is the same operation as the option just discussed, **factor** - to enlarge or reduce the view by a given magnification/reduction factor, and a number of preset view magnification and reduction factors.

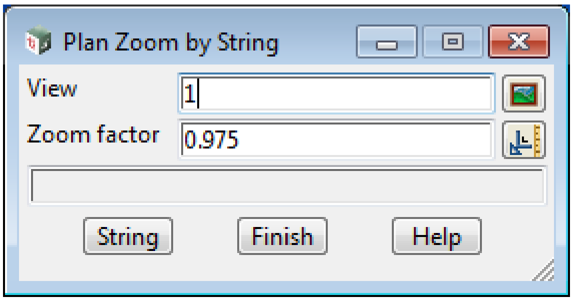
9.1.2.3.1 By String, by Model and by View

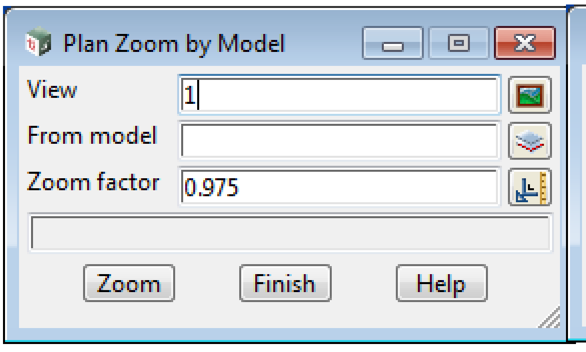
Position of option on menu: **Plan View Menu** **View =>Settings =>Zoom =>by string**

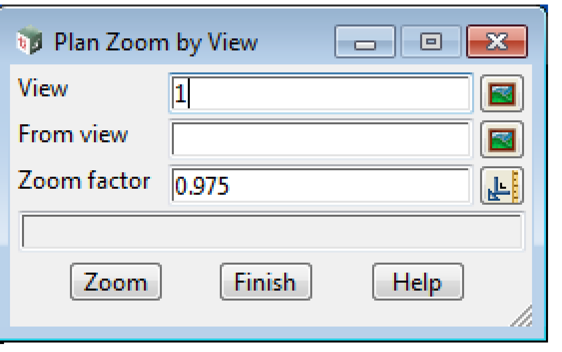
Position of option on menu: **Plan View Menu** **View =>Settings =>Zoom =>by model**

Position of option on menu: **Plan View Menu** **View =>Settings =>Zoom =>by view**

The three zoom options **by String**, **by Model** and **by View** work in a similar manner. After selecting the option, the **Plan Zoom by String/ Model/ View** panel is brought up.







For **by String**, the user selects a string using the **String** button and the data on the view is redrawn so the selected string is fitted to the view, adjusted by the **Zoom factor given in the panel**.

Note that the selected string can be from a different view.

For **by Model**, the user enters a model name into the **Model** field and then selects the **Zoom** button. The data on the view is redrawn so the specified model is fitted to the view, adjusted by the **Zoom factor given in the panel**.

Note that the specified model does not have to be on the view.

For **by View**, the user enters a view name into the **View** field and then selects the **Zoom** button. The data on the view is redrawn so the specified view is fitted to the current view, adjusted by the **Zoom factor given in the panel**.

Note that the specified view does not have to be on the current view.

9.1.2.3.2 Shrink

Position of option on menu:

Plan, Section, Perspective View Menu **View =>Settings =>Zoom =>Shrink**

Shrink is the opposite of zoom: instead of the information in the view box being expanded to fill the view, the information in the view is shrunk to fit into the zoom box.

After selecting the shrink option, the user picks two points from the appropriate plan/section/perspective views (by clicking LB in a view for each point) to define the diagonally opposite points of a rectangle. Note that the two points **do not** have to be from the same plan/section/perspective view.

Next the user selects the view (by clicking LB in the view) which is to be redrawn by shrinking the data on the view to fit into this rectangle.

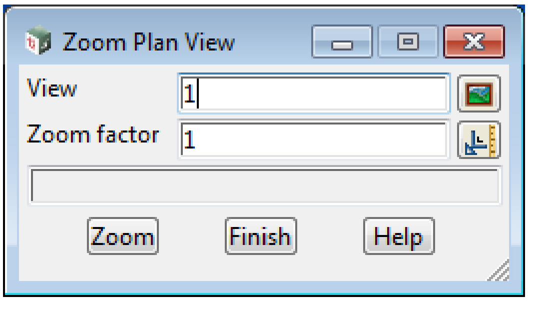
To cancel the zoom after the zoom operation has been selected, click RB.

9.1.2.3.3 Factor

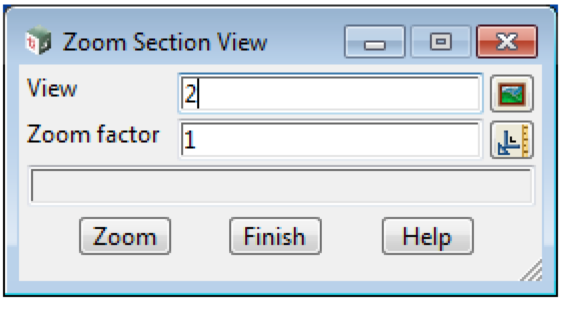
Position of option on menu: **Plan View Menu** **View => Settings => Zoom => Factor**

Position of option on menu: **Section View Menu** **View => Settings => Zoom => Factor**

In Plan View, when selecting the **Factor** option, the **Zoom Plan View** panel is displayed.



In Section View, when selecting the **Factor** option, the **Zoom Section View** panel is displayed.



This option is used to magnify or reduce the information in the view by a user given amount.

The field and buttons in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views
name of the view to set zoom factor for.			
Zoom factor	input	1.0	

factor to magnify/reduce the information inside the view by. The value can be greater than 1.0 (zoom in) or less than 1.0 (zoom out).

Zoom button

the information displayed in the view is magnified/reduced by the factor given in the **zoom factor** *field.*

WARNING - this option can not be applied to a perspective view.

9.1.2.3.4 Pre-set Zoom Factors

Upon selecting a particular zoom factor (e.g. 4 x in), the user is asked to indicate a point in a view which is to become the new view-centre. The user then indicates (by clicking LB in the view) which view is to be redrawn with this point as the new centre and with the selected zoom factor applied to that view.

To cancel the zoom after the zoom operation has been selected, click RB.

Next step messages - sent to the screen message area

<View Magnify> [Select][][] select new centre of view - RB to cancel
<View Magnify> [Select][][] select destination view - RB to cancel

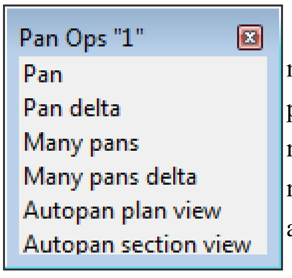
9.1.3 Pan

Position of menu: **Plan View Menu** **View => Pan**

Position of menu: **Section View Menu** **View => Pan**

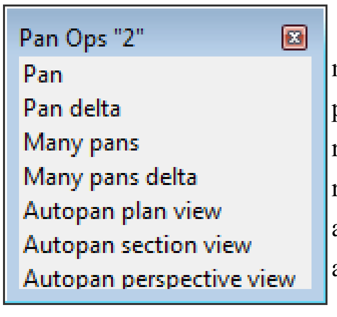
The **Pan** walk-right menu for Plan and Section views are

Plan View Pan Menu



make selected point the new view centre
pick a point and the place it is to be moved to on the screen
repeated pan
repeated pan delta
autopan for plan locked to plan view

Section View Pan Menu



make selected point the new view centre
pick a point and the place it is to be moved to on the screen
repeated pan
repeated pan delta
autopan for plan locked to section view
autopan for section locked to section view

Pan operates the same way for *Plan* and *Section* views.

The **pan** option can be selected in four ways:

(a) by clicking LB on **Pan** on a view button area

(b) by clicking on **<Control> LB** whilst in the view

(c) selecting the **Pan option** itself by clicking LB when the **Pan** option is highlighted on the **Pan** menu

(d) by activating the Pan option's walk-right menu, **Pan Ops**.

The *pan* uses either a *dynamics pan* or two point mode.

9.1.3.1 Dynamic Pan

After selecting the **pan** option, *dynamic pan* is selected by clicking MB or by typing **d**.

Then press LB at the point that is to be dragged across the view during the dynamic pan and then move the mouse to the new position for the point. Another point can then be selected to dynamically pan.

The *dynamic pan* continues until either RB is clicked, or MB is clicked (or **d** typed) to place the pan option into *two point* mode.

After activating the **Pan** walk-right menu, the user is presented with the option **Pan** (which is the same operation as the option just discussed), **Pan delta**, and **Many pans** and **Many pans delta.**

9.1.3.1.1 Pan

Position of option on menu: **Plan View Menu** **View => Pan => Pan**

Position of option on menu: **Section View Menu** **View => Pan => Pan**

After selecting **Pan**, the user is asked to indicate a point in a view which is to become the new view-centre. The user then indicates (by clicking LB in the view) which view is to be redrawn with this point as the new centre.

To cancel the **Pan** after the *pan* operation has been selected, click **RB**.

Next step messages - sent to the screen message area

<View Pan> [Select][][] select new centre of view - RB to cancel
<View Pan> [Select][][] select destination view - RB to cancel

9.1.3.1.2 Pan Delta

Position of option on menu: **Plan View Menu** **View => Settings => Pan => Pan delta**

Position of option on menu: **Section View Menu** **View => Settings => Pan => Pan delta**

After selecting the **Pan delta** option, the user is asked to indicate a point in a view and the point on a view where the first point will be moved to. The user then indicates (by clicking LB in the view) which view is to be redrawn with the new position of the point on the view.

To cancel the **Pan delta** after the *pan delta* operation has been selected, click RB.

Next step messages - sent to the screen message area

<View Pan> [Select][][] select 1st position of pan - RB to cancel
<View Pan> [Select][][] select 2nd position of pan - RB to cancel
<View Pan> [Select][][] select destination view - RB to cancel

9.1.3.1.3 Many Pans

Position of option on menu: **Plan View Menu** **View => Pan => Many pans**

Position of option on menu: **Section View Menu** **View => Pan => Many pans**

The **Many pans** option is simply one *pan* operation followed by another.

After one *pan* operation has been performed, a new *pan* operation is begun.

The sequence is terminated by clicking RB.

9.1.3.1.4 Many Pans Delta

Position of option on menu: **Plan View Menu** **View => Pan => Many pans delta**

Position of option on menu: **Section View Menu** **View => Pan => Many pans delta**

The **Many pans delta** option is simply one *pan delta* operation followed by another.

After one *pan delta* operation has been performed, a new *pan delta* operation is begun.

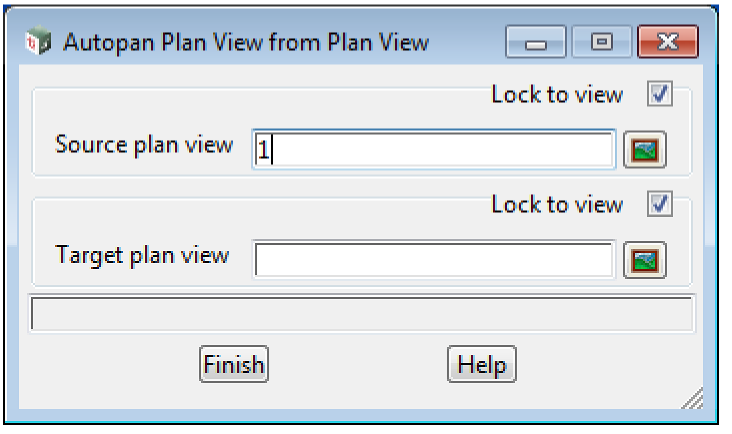
The sequence is terminated by clicking RB.

9.1.3.1.5 Autopan Plan View Locked to a Plan View

Position of option on menu: **Plan View Menu   View =>Pan =>Autopan plan view**

The **Autopan plan view** option locks a target plan view to a source plan view. As the cursor is moved in the source plan view, the target plan view is automatically panned to keep the cursor in the middle of it. The target plan view keeps its scale.

Selecting **Autopan plan view** brings up the **Autopan Plan View from Plan View** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Lock to view	tick box	tick	
Source plan view	view box	current view	available plan views

plan view to lock the target plan view to.

Field Description	Type	Defaults	Pop-Up
Lock to view	tick box	tick	
Target plan view	view box		available plan views

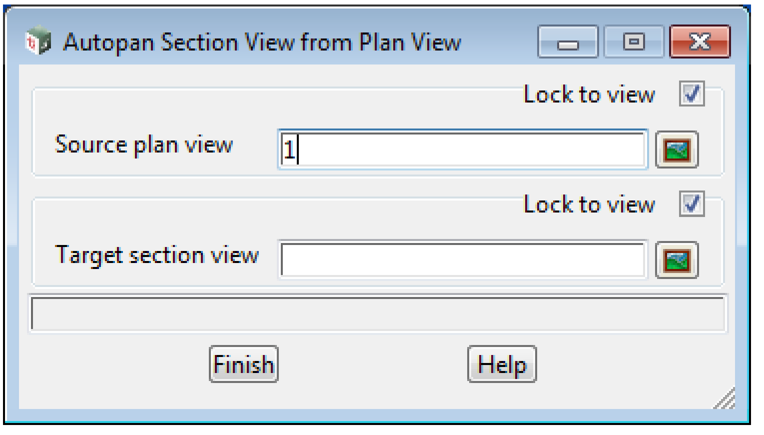
plan view to lock the source plan view. As the cursor moves in the source plan view, the target view is automatically panned so that the cursor is in the centre of the target plan view.

9.1.3.1.6 Autopan Section View Locked to a Plan View

Position of option on menu: **Plan View Menu** **View =>Pan =>Autopan section view**

The **Autopan section view** option locks a target section view to a source plan view. As the cursor is moved in the source plan view, the cursor position is dropped onto the profiled string in the target section view and the target section view is automatically panned to keep the dropped cursor in the middle of it. The target section view keeps its scale.

Selecting **Autopan section view** brings up the **Autopan Section View from Plan View** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Lock to view	tick box	tick	
Source plan view	view box	current view	available plan views

plan view to lock the section view to.

Field Description	Type	Defaults	Pop-Up
Lock to view	tick box	tick	
Target section view	view box		available section views

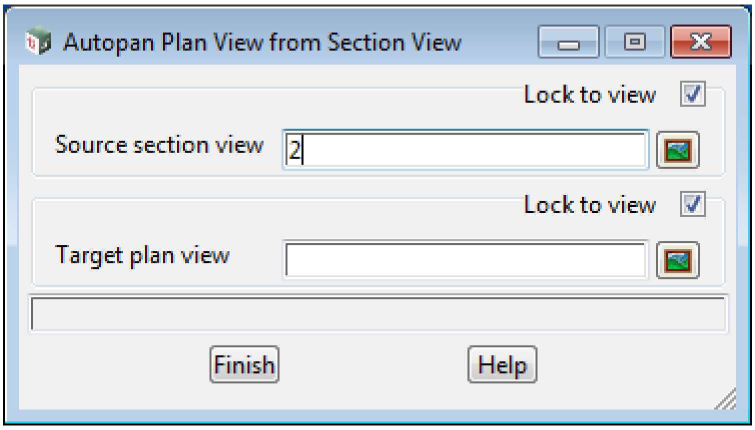
section view to lock the source plan view. As the cursor is moved in the source plan view, the cursor position is dropped onto the profiled string in the target section view and the target section view is automatically panned to keep the dropped cursor in the middle of it. The target section view keeps its scale.

9.1.3.1.7 Autopan Plan View Locked to a Section View

Position of option on menu: **Section View Menu** **View => Settings => Pan => Autopan plan view**

The **Autopan plan view** option locks a target plan view to a source section view. As the cursor is moved in the source section view, the target plan view is automatically panned to keep the cursor in the middle of it. The target plan view keeps its scale.

Selecting **Autopan plan view** brings up the **Autopan Plan View from Section View** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Lock to view	tick box	tick	
Source section view	view box	current view	available plan views

section view to lock the target plan view to.

Lock to view	tick box	tick	
Target plan view	view box		available plan views

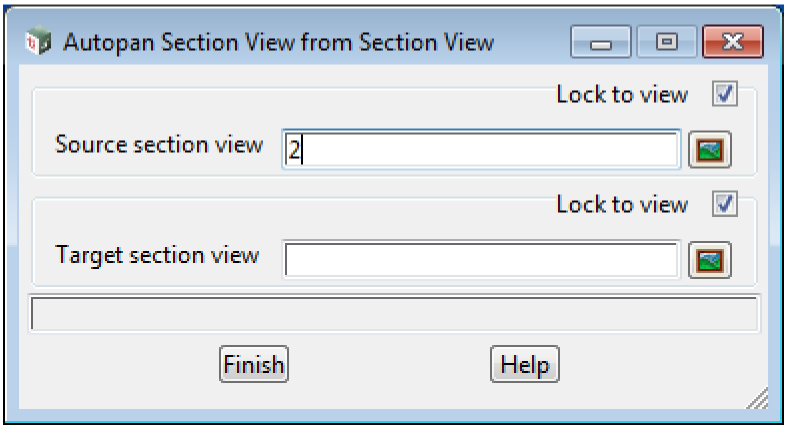
plan view to lock the source section view. As the cursor moves in the source section view, the target view is automatically panned so that the cursor is in the centre of the target plan view.

9.1.3.1.8 Autopan Section View Locked to a Section View

Position of option on menu:Section View Menu View =>Pan =>Autopan section view

The **Autopan section view** option locks a target section view to a source plan view. As the cursor is moved in the source section view, the cursor position is dropped onto the profiled string in the target section view and the target section view is automatically panned to keep the dropped cursor in the middle of it. The target section view keeps its scale.

Selecting **Autopan section view** brings up the **Autopan Section View from Section View** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Lock to view	tick box	tick	
Source section view	view box	current view	available section views

section view to lock the target section view to.

Field Description	Type	Defaults	Pop-Up
Lock to view	tick box	tick	
Target section view	view box		available section views

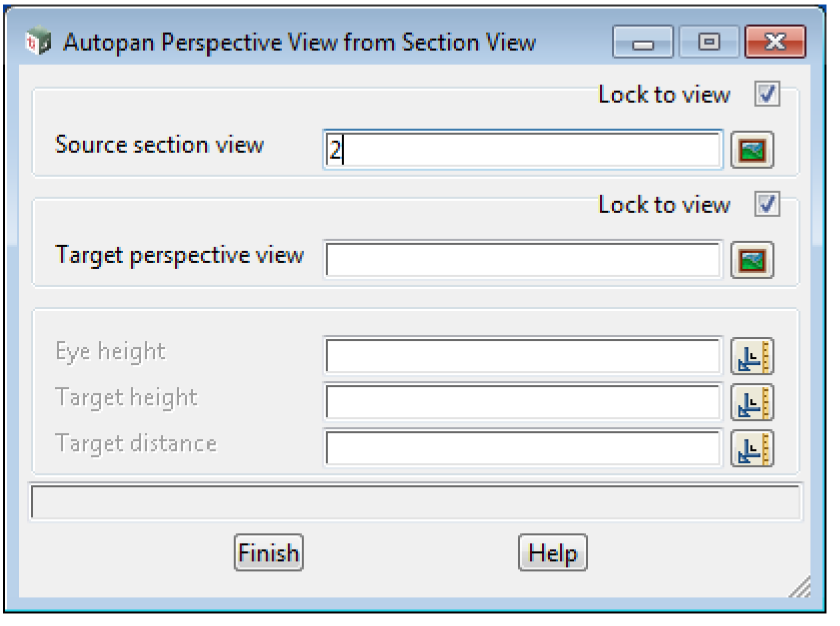
section view to lock the source section view. As the cursor is moved in the source section view, the cursor position is dropped onto the profiled string in the target section view and the target section view is automatically panned to keep the dropped cursor in the middle of it. The target section view keeps its scale.

9.1.3.1.9 Autopan Perspective View Locked to a Section View

Position on menu: **Section View Menu** **View => Settings => Pan => Autopan perspective view**

The **Autopan perspective view** option locks a target perspective view to a source section view. As the cursor is moved in the source section view, the eye and target for the perspective view is automatically changed. Only the eye and target positions are modified for the target perspective view.

Selecting **Autopan perspective view** brings up the **Autopan Perspective View from Section View** panel.



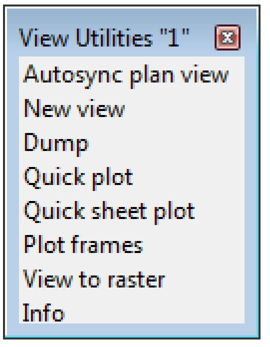
The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Lock to view	tick box	tick	
if ticked, the source view is used.			
Source section view	view box	current view	available section views
section view to lock the target perspective view to.			
Lock to view	tick box	tick	
if ticked, the target view is used.			
Target perspective view	view box		available perspective views
perspective view to lock to the source section view. As the cursor moves in the source section view, the eye and target position for the target perspective view is calculated from the (x,y,z) of the profiled string in the section view, and the eye and target information in the panel.			

9.1.4 Plan Utilities

Position of menu: **Plan View Menu** **View => Settings => Utilities**

The **Utilities** menu contains miscellaneous options involving the plan view. The **Utilities** walk-right menu is



For the option:

Autosync plan view, go to 9.1.4.1 Autosync Plan View Locked to a Plan View

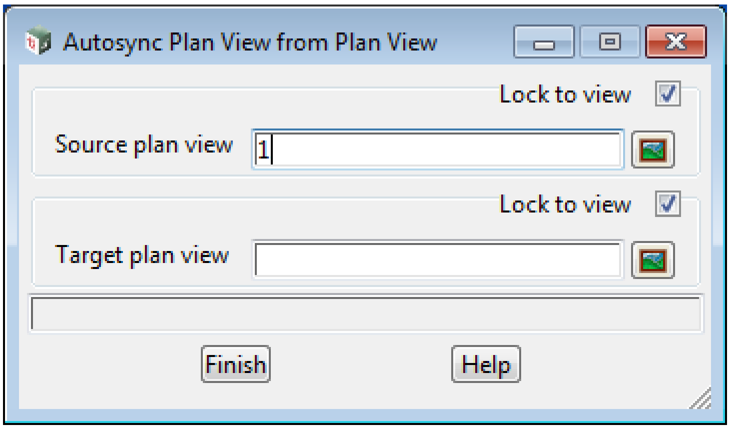
New view settings	9.1.4.2 New View Settings
Dump	9.1.4.3 View Dump
Quick *plot*	9.1.4.4 Quick Plan Plot
Quick sheet *plot*	9.1.4.5 Quick Sheet Plot
Plot frames	9.1.4.6 Plot Frames
View to raster	9.1.4.7 Create Raster from Plan View
Info	9.1.4.8 View Info

9.1.4.1 Autosync Plan View Locked to a Plan View

Position of option on menu: **Plan View Menu** **View =>Utilities =>Autosync plan view**

The **Autosync plan view** option locks a target plan view to a source plan view. As the cursor is moved in the source plan view, the same movement is made in the target plan view.

Selecting **Autosync plan view** brings up the **Autosync Plan View from Plan View** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Lock to view	tick box	tick	
Source plan view	view box	current view	available plan views

plan view to lock the target plan view to.

Lock to view	tick box	tick	
Target plan view	view box		available plan views

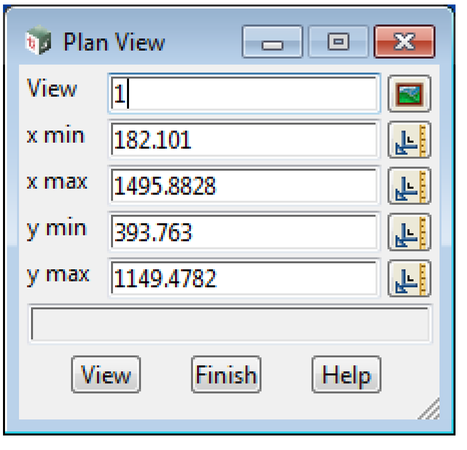
plan view to lock to the source plan view. As the cursor is moved in the source plan view, the same movement is made in the target plan view.

9.1.4.2 New View Settings

Position of option on menu: Plan View Menu View =>Settings =>Utilities =>New view settings

The **new view settings** option displays the current view setting and permits the user to specify new x and y minimum and maximum values as the viewing parameters.

Selecting **New view settings** fires up the **Plan View** panel.



Because the aspect ratio of the view may be different from the aspect ratio defined by the user supplied x and y minimums and maximums, slightly more data than the specified range may need to be drawn in the view. However, the view will always **include** the user given range.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views

view to set min and max for.

x min/max	input/output	current x min/max	

minimum/maximum x co-ordinate for the view

y min/max	input/output	current y min/max	

minimum/maximum y co-ordinate for the view

View	button		

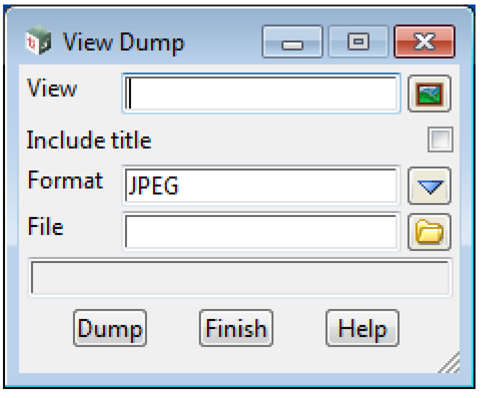
define the view parameters according to the information in the panel. If the box given by the x and y values in the panel is not the same shape as the view itself (that is, has the same aspect ratio), the x and y values will be modified so that the x and y values match the view shape. The final x and y values are *displayed in the panel.*

9.1.4.3 View Dump

Position of option on menu: **Plan View Menu** **View => Settings => Utilities => Dump**

This option is used to write the image of the view out to disk in a user selected format.

Selecting **Dump** brings up the **View Dump** panel.



The fields and buttons have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input	view option picked from	available views

view to dump out in the specified format.

Include title	tick box		

*if **ticked**, the view title area is included in the dump.*

Format	input	gif	gif, xwd, colour postscript

the format to dump the view out in.

File	input	*.gif	

the file to dump the view images out to

Dump	button		

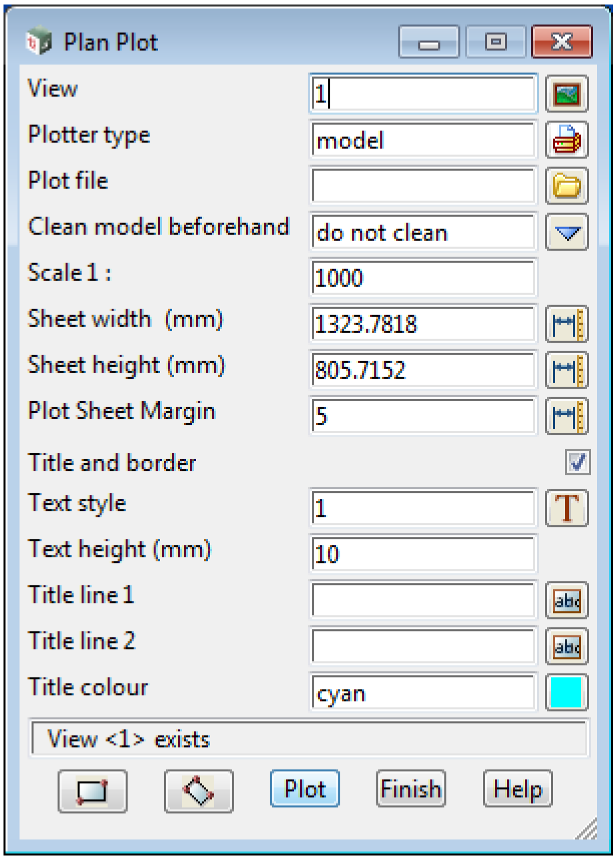
dump in the given format the image of the view given in the **view** *field to the file given in the* **file** *field.*

9.1.4.4 Quick Plan Plot

Position of option on menu: **Plan View Menu** **View =>Settings =>Utilities =>Quick plot**

The **Quick plot** option writes out all the information displayed in the plan view to either a plot file (in a user selected format) or to a 12d Model model. The user selects a scale for the plot and the sheet width and height are calculated by 12d Model.

Selecting **Quick plot** brings up the **Plan Plot** panel.



The fields and buttons have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box	current view	available views

view to plot.

Plotter type	input	hp	hp, dxf, postscript *etc.*

format for the plot information.

Plot file	input	depends on plotter type	

stem of the name of the file to write the plot of the view to. The name ending is added automatically and depends on the plotter type.

Clean model beforehand	choice box		do not clean prompt for clean always clean

only applicable if plotting to a model.

*if **always clean**, the model is cleaned before the plot is created.*
*If **prompt for clean**, the user is prompted that the model will be cleaned before the plot is created.*
*If **do not clean**, the model is not cleaned before the plot is created.*

Scale 1:	input		

if a value is entered by the user and an <enter> given, the sheet width and height required by the plot are calculated and displayed in the **sheet width** *and* **sheet height** *fields.*

Sheet width (mm) output

if a value is entered by the user and an <enter> given, the scale and height required by the plot are calculated and displayed in the **scale** *and* **sheet height** *fields. The units for sheet width are* ***millimetres****.*

Sheet height (mm) output

if a value is entered by the user and an <enter> given, the scale and width required by the plot are calculated and displayed in the **scale** *and* **sheet width** *fields. The units for sheet height are* ***millimetres***

Title and border tick box

if ***ticked****, a border and two lines of title are placed on the bottom of the plot.*

Title line 1/2 input

first/second line of title information.

Title height (mm) input 10

height (in ***millimetres****) to draw the two lines of title information.*

Title colour input cyan available colours

colour used for the border and the title information.

Rectangle button

create a rectangle (parallel to the x,y axis) that the plot is restricted to.

Rotated Rectangle button

create a rotated rectangle that the plot is restricted to.

Plot button

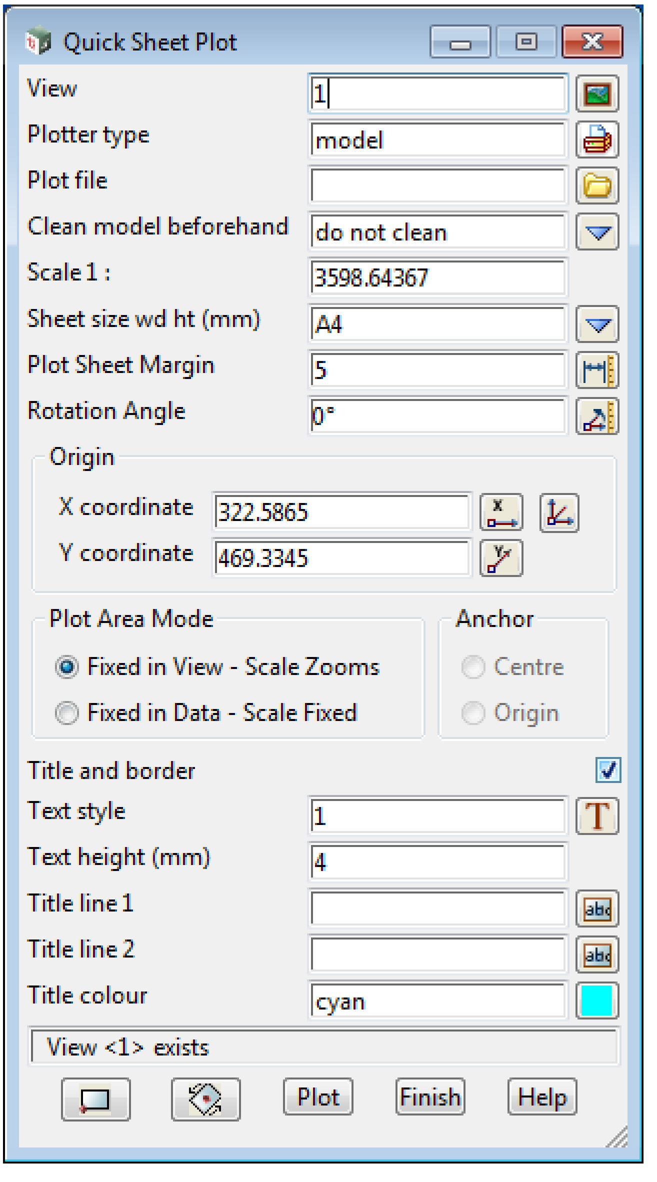
write out the plot of the information displayed in the view to the file given in the **plot file** *field. The format of the file is given by the* **plotter type.**

9.1.4.5 Quick Sheet Plot

Position of option on menu: Plan View Menu View =>Settings =>Utilities =>Quick sheet plot

The **Quick sheet plot** option takes a user selected plot sheet and writes out all the information displayed in the plan view in the sheet to either a plot file (in a user selected format) or to a 12d Model model. The user selects the plot sheet and a scale for the plot.

Selecting **Quick sheet plot** displays the **Quick Sheet Plot** panel.



The fields and buttons have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box	current view	available views

view to plot.

Plotter type	choice box	hp	hp, dxf, postscript *etc.*

format for the plot information.

Plot file	input	depends on plotter type	

stem of the name of the file to write the plot of the view to. The name ending is added automatically and *depends on the plotter type.*

Clean model beforehand choice box do not clean / prompt for clean / always clean

only applicable if plotting to a model.

*if **always clean**, the model is cleaned before the plot is created.*
*If **prompt for clean**, the user is prompted that the model will be cleaned before the plot is created.*
*If **do not clean**, the model is not cleaned before the plot is created.*

Scale 1: real value

the plot scale.

Sheet size wd ht (mm) sheet box defined sheet sizes

a sheet is selected from the pop-up or the width and height (in mm) are entered, separated by a space.

Rotation angle dms value

the anti-clockwise rotation (in dms) for the plot sheet.

Origin

select or type in the position of the bottom left hand corner of the plot region.

View controlled/ Data controlled radio buttons

*if **Data controlled**, the plot sheet maintains its size and position with **respect to the data in the view.** That is, the plot sheet origin is locked to the selected data co-ordinates, and the plot scale is fixed so that the plot sheet then has a set size in world (data) units. If pans and zooms are made to the data in the view, the plot sheet will then move/zoom with data.*

*if **View controlled**, the plot sheet maintains its size and position with **respect to the view itself**. That is, the plot sheet is locked in size and position with respect to the view itself. If pans and zooms are made to the data in the view, the sheet scale and origin are automatically modified to keep the plot frame in exactly the same position and size in the view.*

Title and border tick box

*if **ticked**, a border and two lines of title are placed on the bottom of the plot.*

Title line 1/2 input

first/second line of title information.

Title height (mm) input 10

height (in millimetres) to draw the two lines of title information.

Title colour input cyan available colours

colour used for the border and the title information.

Rectangle button

if selected, the plot frame is placed by selecting a cursor position from a view. This defines the Origin of the plot sheet. The sheet size, plot sheet scale and sheet rotation are taken from the panel fields.

Centre and Rotate button

if selected, a centre of for the plot sheet is selected and then an second point to define the rotation of the plot sheet. This defines the Origin and Rotation of the plot sheet. The sheet size and scale are taken from the panel fields.

Plot button

write out the plot of the information displayed in the plot sheet to the file given in the **plot file** *field. The format of the file is given by the* **plotter type.**

9.1.4.6 Plot Frames

Position of option on menu: **Plan View Menu** **View => Settings => Utilities => Plot frames**

To create plan plots of an arbitrary size, rotation and scale, 12d Model uses **plot frames**.

Basically a plot frame consists of a sheet size (in mm), margins within the sheet and a scale for the plot. This will define a plotting area in world co-ordinates (the plot frame).

The plot frame can be arbitrarily positioned and rotated on a plan view.

This option is documented in the section 25.8.12 Plotting Old Plot Frames.

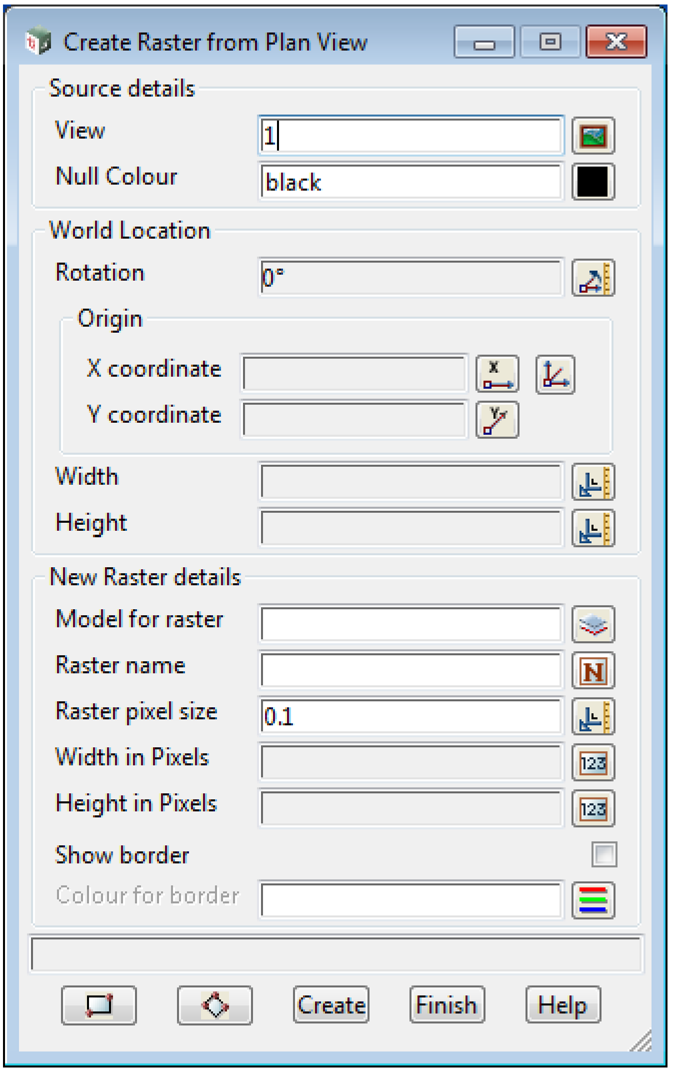
9.1.4.7 Create Raster from Plan View

Position of option on menu: **Plan View Menu** **View => Settings => Utilities => View to raster**

Position of option on menu: **Strings =>Raster => Plan to raster**

This option creates a 12d raster of a user specified pixel size, for the image on a plan view. This is useful for creating a 12d raster from an ECW file to use in draping on a tin in visualisations.

On selecting the **View to Raster** option, the **Create Raster from Plan View** panel is displayed.



The fields and buttons have the following functions.

Field Description	Type	Defaults	Pop-Up

Source Details

View	view box	current view	available views

view to create raster from.

Null colour	colour box		available colours

colour to set to no pixel in the raster

World Location

Anticlockwise rotation	angle box		

the world rotation of the selected rectangle.

X/Y co-ordinate real box

the world x/y co-ordinate of the corner of the bottom left corner of the rectangle.

Width/height real box

the width/height in world units of the rectangle.

New Raster Details

Model for raster model box available models

name of the model for the raster element.

Raster name text box

the name for the raster.

Raster pixel size input

the size (in world units) for each pixel in the created raster.

Width/Height in pixels output only

the width/height in pixels of the created raster.

Show border tick box tick

if ticked then the border of the created raster element is displayed.

Colour for border input default colour available colours

the colour of the border for the created raster.

Rectangle button

create a rectangle (parallel to the x,y axis) that the created raster is restricted to.

Rotated Rectangle button

create a rotated rectangle that the created raster is restricted to.

Create button

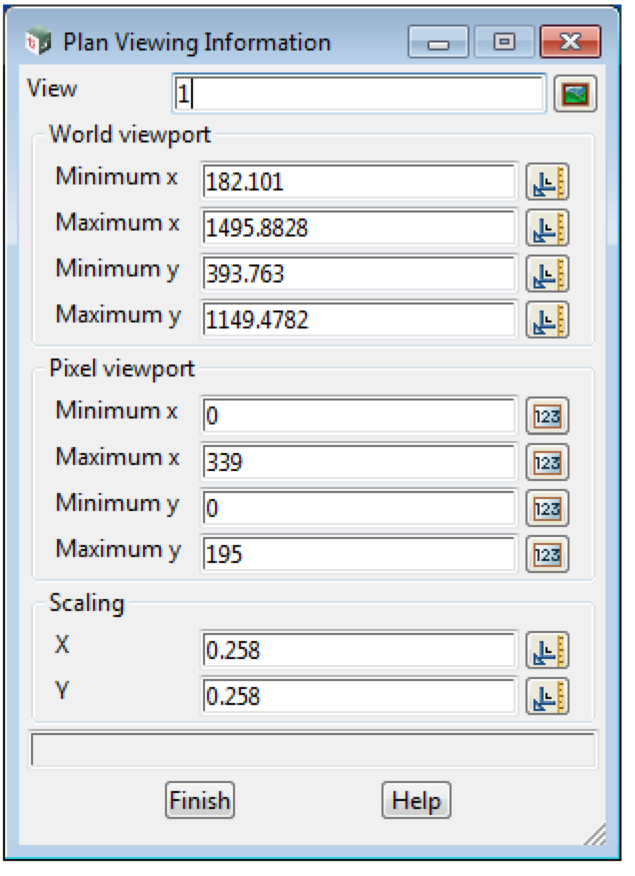
create a 12d raster from the image on the plan view.

9.1.4.8 View Info

Position of option on menu: **Plan View Menu** **View => Utilities => Info**

This option displays information about the world and pixel size of a plan view.

On selecting the **Info** option, the **Plan Viewing Information** panel is displayed.



The fields and buttons have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box	current view	available plan views

when activated from the Plan View menu, the name of the view is placed into this field and the information about this view is displayed in the panel fields.

Once the panel is up, a different Plan View name can be selected and the information will be displayed for that view.

World viewport

Minimum, Maximum x display only

display the world minimum and maximum x coordinates for the view.

Minimum, Maximum y display only

display the world minimum and maximum y coordinates for the view.

Pixel viewport

Minimum, Maximum x display only

display the minimum and maximum x pixels coordinates for the view.

Minimum, Maximum y display only

display the world minimum and maximum y pixels coordinates for the view.

Note - for pixels coordinates, (0,0) is at the top left hand corner of the view and positive x goes to the right, and positive y goes down the view.

Scaling

X display only

the ratio of x pixel length of the view to the x world unit length of the view. That is,

(maximum x pixels - minimum x pixels)/ (maximum x world units - minimum x world units)

Y display only

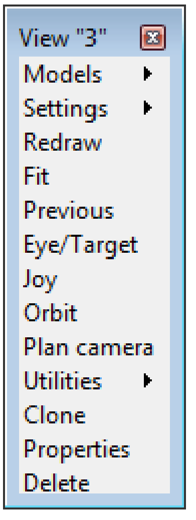
the ratio of y pixel length of the view to the y world unit length of the view. That is,

(maximum y pixels - minimum y pixels)/ (maximum y world units - minimum y world units)

9.2 Perspective View Menu

Position of menu: **Perspective View Menu** **View**

The **Perspective view** menu is



For the options

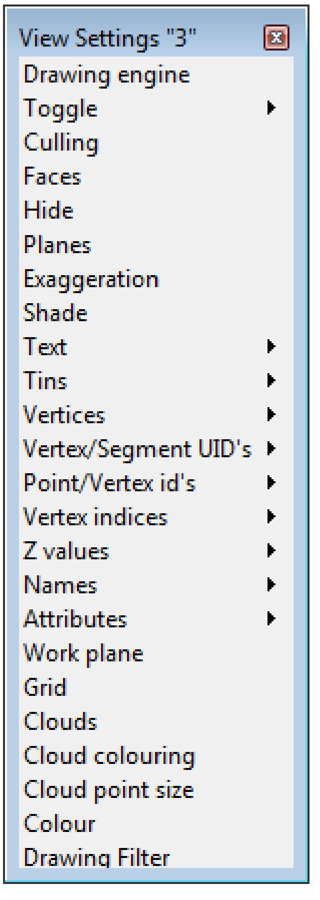
Models, go to	9.4.1 Model Ops
Settings	9.2.1 Perspective View Settings
Redraw	9.4.2 Redraw
Fit	9.4.3 Fit
Previous	9.4.4 Previous
Eye/Target	9.2.2 Eye/Target
Joy	9.2.3 Joy for View
Orbit	9.2.4 Orbit
Plan Camera	9.2.5 Plan Camera
Utilities	9.2.6 Perspective Utilities
Clone	9.4.5 Clone
Properties	9.4.6 Properties
Delete	9.4.7 Delete

9.2.1 Perspective View Settings

Position of menu: **Perspective View Menu** **View =>Settings**

If the **Settings** option is picked rather than moving onto the walking right, then the **Toggle** menu from the **Toggle** walk-right menu is displayed on the screen. The **Toggle** menu will be described in the next section.

The **Settings** walk-right menu for the perspective view is



For the options:

Drawing engine go to	9.1.1.1 Drawing Engine
Drawing density	
Toggle	9.2.1.2 Perspective Toggle
Culling	9.2.1.3 Culling
Faces	9.2.1.4 Face Flags for View
Hide	9.2.1.5 Hide View
Planes	9.2.1.6 Clipping Planes for View
Exaggeration	9.2.1.7 Perspective View Exaggeration
Shade	9.2.1.9 Shade
Text	
Tins	9.2.1.8 Tins
Work Plane	9.1.1.19 Work Plane
Grid	9.1.1.22 Grid on View
Colour	9.1.1.23 View Background Colour
Drawing filter	9.1.1.24 View Drawing Filter

9.2.1.1 Drawing Engine

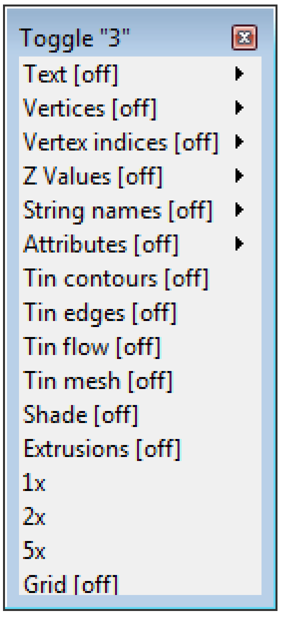
This is documented in 9.2.1.1 Drawing Engine.

Continue to 9.2.1.2 Perspective Toggle or return to 9.2.1 Perspective View Settings.

9.2.1.2 Perspective Toggle

Position of menu: **Perspective View Menu** **View =>Settings =>Toggle**

The **Toggle** walk right brings up the **Toggle** perspective view menu.



Selecting any options from this menu will toggle the option on/off.

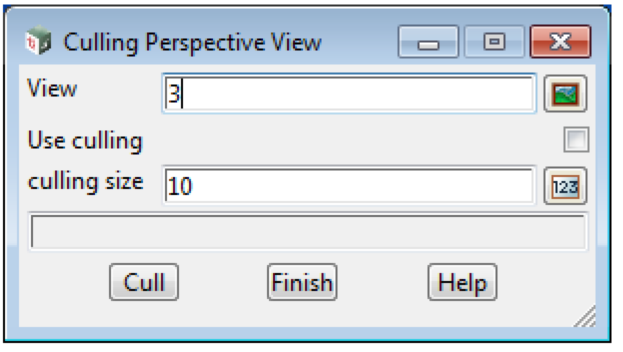
Continue to 9.2.1.3 Culling or return to 9.2.1 Perspective View Settings.

9.2.1.3 Culling

Position of option on menu: **Perspective View Menu** **View =>Settings =>Culling**

The **Culling** option is used to suppress the drawing of strings whose on-screen extent is less than a user defined pixel size.

Selecting **Culling** raises the **Culling Perspective View** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views

view to set culling parameters for.

Use culling	tick box		

*if **ticked**, a string is **not** drawn on the perspective view whenever the string's extent box when drawn on the view would be smaller the culling size given in the* **culling size** *field.*

Culling size	input	10	

pixel size used for culling

Cull	button		

record the culling size given in the **culling field**. *If the* **use culling** *field and redraw the view.*

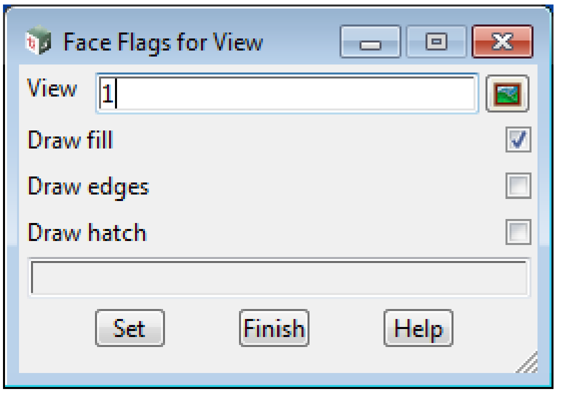
Continue to 9.2.1.4 Face Flags for View or return to 9.2.1 Perspective View Settings.

9.2.1.4 Face Flags for View

Position of option on menu: **Perspective View Menu** **View =>Settings =>Faces**

The **faces** option allows the user to specify how faces are displayed in the perspective view and on any perspective view plots.

Selecting **Faces** fires up the **Face Flags for View** panel.



The fields and buttons have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views

view to face flags for.

Draw fill	tick box	tick	

*if **ticked**, all faces in the view are drawn in their fill colour.*

Draw edges	tick box		

*if **ticked**, all face edges in the view are drawn.*

Draw hatch	tick box		

*if **ticked**, all faces in the view are drawn in their hatch pattern.*

Set	button		

set the **draw fill/edges/hatch** *fields to the value in the panel fields. The plan view is then redrawn.*

Continue to 9.2.1.5 Hide View or return to 9.2.1 Perspective View Settings.

9.2.1.5 Hide View

Position of option on menu: **Perspective View Menu** **View =>Settings =>Hide**

When looking across a landscape, part of the terrain is often hidden by other parts of the terrain. For example hills in the foreground will hide hills behind them.

In ***12d Model***, a landscape is represented by a triangulated surface (a tin) and a tin can be used by a hidden line algorithm to determining what is visible is a scene.

The **hide** option in***12d Model*** uses a tin to define a surface and processes any strings on the perspective against the tin to determine what is visible or hidden.

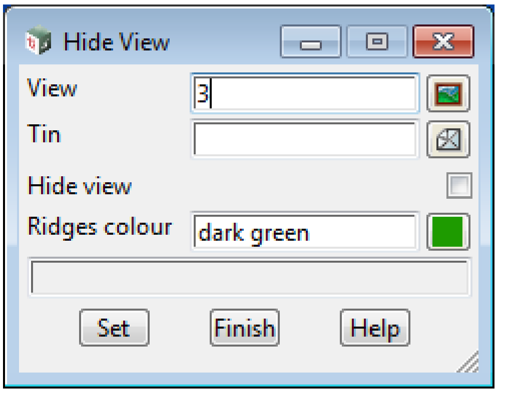
Only strings that lie on the tin can be sensibly processed. For example, contours, meshes, draped strings and any of the data that was used to create the tin.

When viewing a terrain from an eye point, the **ridge lines** are the edges of transition between visible and invisible areas. The **hide** option displays the ridge lines to delineate the hidden regions.

Once the **hide** option is set on, a hide will be done on the view whenever the view parameters are changed. If a model is added to the view when hide is set on, the strings in the model are processed and only the visible sections displayed in the view.

Note - strings can still be selected in a hidden view.

Selecting **Hide** raises the **Hide View** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views

view to set hide parameters for.

Tin	input		available tins

the name of the triangulated surface (tin) to be used for determining what is visible or not.

Hide view	tick box		

*if **ticked**, then the view is processed when the* **set** *button is selected.*

Ridges colour	input		available colours

the colour for the ridge lines.

Set	button		

record the answers for the **hide view**, **tin** *and* **ridges colour** *fields and then process the perspective view using the new parameters. The view type will be changed to **hidden**.*

WARNING - a hide can be very slow for large tins and/or large data sets.

Continue to 9.2.1.6 Clipping Planes for View or return to 9.2.1 Perspective View Settings.

9.2.1.6 Clipping Planes for View

Position of option on menu: **Perspective View Menu** **View =>Settings =>Planes**

The perspective viewing direction is defined by specifying an eye and target point.

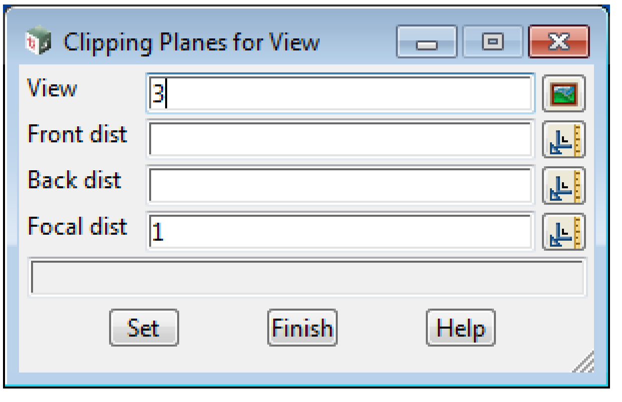
The three dimensional view is then projected onto a plane at right angles to the eye-target line, and at a user defined distance from the eye-point. This plane is called the focal plane and the distance from the eye-point to the focal plane is the focal distance.

The projection of all the data onto the focal plane is what is seen on the screen in a perspective view. Moving the focal plane has the same effect as varying the telephoto lens on a camera - it gives the impression of zooming into and out of the view.

It is not always desirable to include all the data in the perspective view. For example, data behind the viewer is not normally required to be seen. The user can specify the position of two planes (called the front and back clipping planes) parallel to the focal plane. Only data lying between the clipping planes will be seen. Thus the view is "clipped" using the front and back clipping planes.

The distance to the clipping planes is measured from the eye point along the line joining the eye and target points (the eye-target line).

Selecting **Planes** fires up the **Clipping Planes for View** panel. This panel allows the user to set the focal distance and the front and back clipping plane distances.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views

view to set clipping plane for.

Front dist	input/output

distance from the eye point to the front clipping plane. If this field is blank, no front clipping plane is used.

Back dist	input/output

distance from the eye point to the back clipping plane. If this field is blank, no back clipping plane is use.

Focal dist	input/output

distance from the eye point to the focal plane. This distance must be non-zero.

Set	button

set the distances to the front and back clipping planes and the focal plane. If the front or back clipping distance is blank, that plane is not set. After the set button is chosen, the view is redrawn using the new parameters,

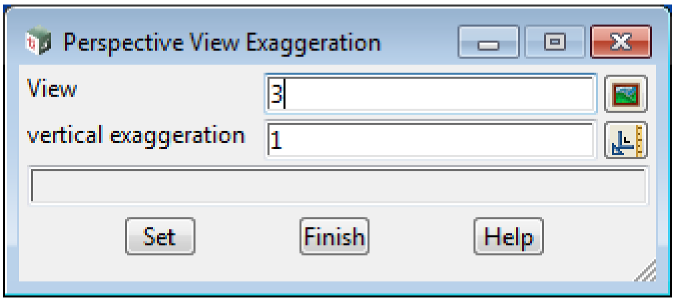
Continue to 9.2.1.7 Perspective View Exaggeration or return to 9.2.1 Perspective View Settings.

9.2.1.7 Perspective View Exaggeration

Position of option on menu: **Perspective View Menu** **View =>Settings =>Exaggeration**

The exaggeration option allows the user to specify the vertical exaggeration in the perspective view. The heights (z values) are multiplied by the vertical exaggeration value before drawing on the perspective view.

Selecting **Exaggeration** fires up the **Perspective View Exaggeration** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views

view to set exaggeration for.

Vertical exaggeration	input	10	1,5,10

value to multiply the heights (z values) by before drawing in the perspective view.

Set	button		

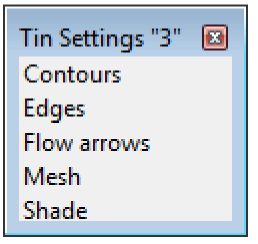
set the vertical exaggeration to the value in the vertical exaggeration field.

Continue to 9.2.1.8 Tins or return to 9.2.1 Perspective View Settings.

9.2.1.8 Tins

Position of menu: **Perspective View Menu** **View =>Settings =>Tins**

The options on the **Tins** walk-right menu control the display of tins on the view.The **Tins** walk-right menu is



For the option Contours, go to	9.2.1.8.1 Contours
Edges	9.2.1.8.2 Edges
Flow arrows	9.2.1.8.3 Flow Arrows
Mesh	9.2.1.8.4 Mesh
Shade	9.2.1.8.5 Shade

9.2.1.8.1 Contours

Position of option on menu: **Perspective View Menu** **View =>Settings =>Tins =>Contours**

The **contours** option defines contour and bold increments and colours, and also whether these contours are displayed for the triangles from any tins on the view. The panel is the same as for the plan view option. If **Shade** is set on, the triangles will be drawn in back to front order.

Continue to 9.2.1.8.2 Edges or return to 9.2.1.8 Tins or 9.2.1 Perspective View Settings.

9.2.1.8.2 Edges

Position of menu: **Perspective View Menu** **View =>Settings =>Tins =>Edges**

The **Edges** option allows the user to specify whether the edges of triangles from any tins on the view are displayed. The panel is the same as for the plan view option. If **Shade** is set on, the triangles will be drawn in back to front order.

Continue to 9.2.1.8.3 Flow Arrows or return to 9.2.1.8 Tins or 9.2.1 Perspective View Settings.

9.2.1.8.3 Flow Arrows

Position of option on menu: **Perspective View Menu** **View =>Settings =>Tins =>Flow arrows**

The **Flow arrows** option defines the colour and length of flow arrows, and also whether the flow arrows are displayed for the triangles from any tins on the view. The panel is the same as for the plan view option. If **Shade** is set on, the triangles will be drawn in back to front order.

Continue to 9.2.1.8.4 Mesh or return to 9.2.1.8 Tins or 9.2.1 Perspective View Settings.

9.2.1.8.4 Mesh

Position of option on menu: **Perspective View Menu** **View =>Settings =>Tins =>Mesh**

The **Mesh** option defines a rectangular mesh, and also whether the mesh is displayed for the triangles from any tins on the view. The panel is the same as for the plan view option. If **Shade** is set on, the triangles will be drawn in back to front order.

Continue to 9.2.1.8.5 Shade or return to 9.2.1.8 Tins or 9.2.1 Perspective View Settings.

9.2.1.8.5 Shade

Position of option on menu: **Perspective View Menu** **View =>Settings =>Tins =>Shade**

Position of option on menu: **Perspective View Menu** **View =>Settings =>Shade**

This is exactly the same option as the **Shade** described in the section 9.2.1.9 Shade.

Return to 9.2.1.8 Tins or 9.2.1 Perspective View Settings.

9.2.1.9 Shade

Position of option on menu: **Perspective View Menu** **View =>Settings =>Tins =>Shade**

Position of option on menu: **Perspective View Menu** **View =>Settings =>Shade**

In ***12d Model***, a landscape is represented by a triangulated surface - a tin. Each triangle in the tin is a part of a plane and has its own colour.

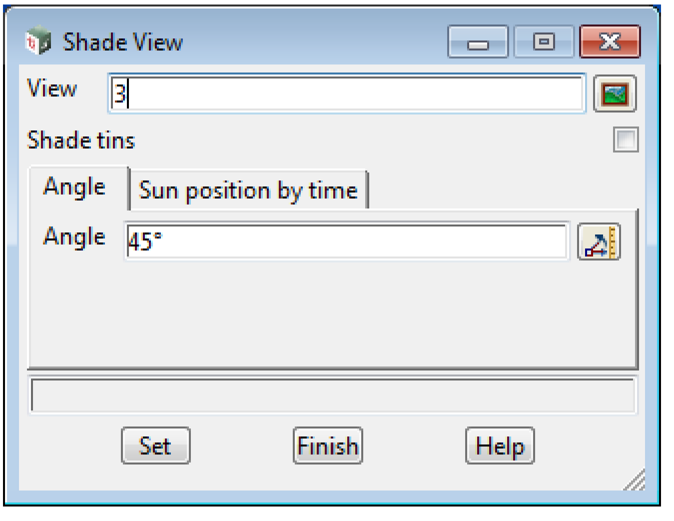
In a perspective view, each triangle can be drawn with a colour that is modified depending on the angle that the triangle makes with the sun (a point light source at infinity). This is called a **flat shade**.

If the triangles are drawn in a back to front order, then when the shaded landscape is drawn, any triangles in the foreground obscure triangles in the background and it looks like a hidden view.

If **shade** is set for a view, all the triangles in all the tins on the view are drawn as shaded triangles and the triangles are drawn in a back to front order.

Once the **Shade** is set on, a shade will be done on the view whenever the view parameters are changed. A plot of the shaded view can be made using the **dump** option on the view.

Selecting **Shade** fires up the **Shade View** panel:



The fields and buttons used in this panel have the functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views

view to set shade parameters for.

Shade tins	tick box		

*if **ticked**, draw all the tins on the perspective view in shade mode.*

Angle	input	45	

angle of the sun (measured from the horizon).

Set	button		

record and redraw the view using the current parameters.

Return to 9.2.1 Perspective View Settings.

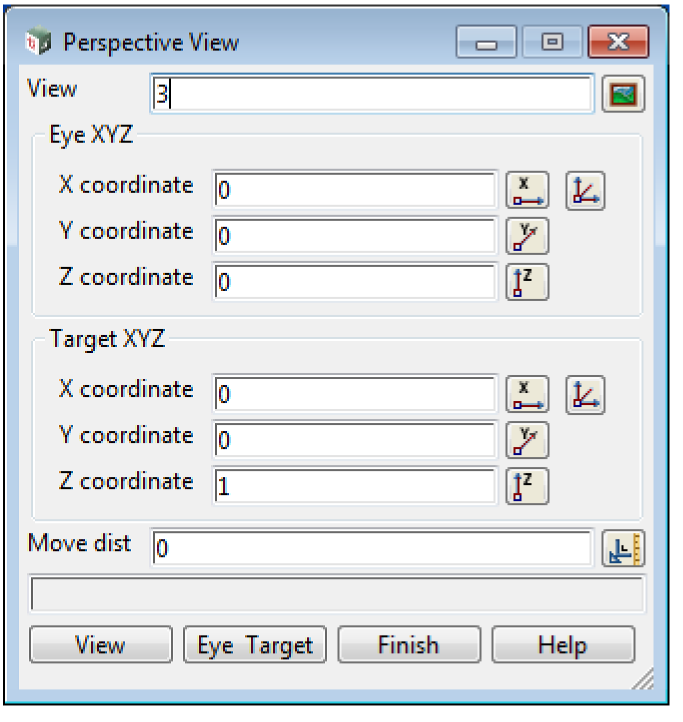
9.2.2 Eye/Target

Position of option on menu: **Perspective View Menu** **View =>Eye/Target**

The **Eye/Target** option displays the current eye and target co-ordinates and also allows the user to specify new eye and target points. The points can be chosen from any plan view using the cursor or by typed-input.

The view is projected onto a focal plane using a perspective transformation, and clipped using front and back clipping planes. The focal distance and the front and back clipping plane distances are set and modified in the **planes** option.

Selecting **Eye/Target** fires up the **Perspective View** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box	current view	available views

view to set perspective parameters for.

Eye X Y Z	XYZ select	current eye position	xyz ops menu

co-ordinates of the eye viewing point. New values can be typed, or a point selected using the **xyz ops** *pop-up menu.*

Target X Y Z	XYZ select	current target position	xyz ops menu

co-ordinates of the target viewing point. New values can be typed, or a point selected using the **xyz ops** *pop-up menu.*

Move dist	input/output		

distance to move the eye point along the eye-target line in the direction of the target point. The target point is also moved the same distance along the eye-target line so that the eye-target distance is kept constant.

View	button		

define the perspective parameters according to the information in the panel. If the move distance is *non-zero, then the eye and target points are both moved along the eye-target line by the distance given in the Move field. In this case, the new eye and target positions will be written to the* **eye** *and* **target** *fields.*

Eye Target button

allows the user to define the (x, y) position of the eye and target points using the cursor and views.

After selection the button, the user is asked to select the eye position from a view by clicking LB in a view. A level *input box is then displayed on the screen for the user to enter the level (z value) for the eye point. The level is recorded and the level box removed after a <enter> is entered in the level input box.*

The user is then asked to select the target point from any view by clicking LB at the position of the target point. A level *input box is then displayed for the user to enter the level (z value) for the target point.*

After the target point is defined, the view is redrawn using the new eye and target positions.

How to Use the Panel and Panel Messages

(a) The eye and target points are set by using typed-input, or by selection in either the **eye XYZ** or **target XYZ** fields, or the **Eye Target** button.

(b) When the **View** button is selected, the perspective viewing parameters are calculated according to the information in the panel. If a non-zero move distance is given, the eye and target points are both moved along the eye-target line through the move distance The new values for the eye and target points are written into the panel. If no errors are detected in calculating the new viewing parameter, the view is redrawn using the new parameters.

Continue to 9.2.3 Joy for View or return to 9.2 Perspective View Menu.

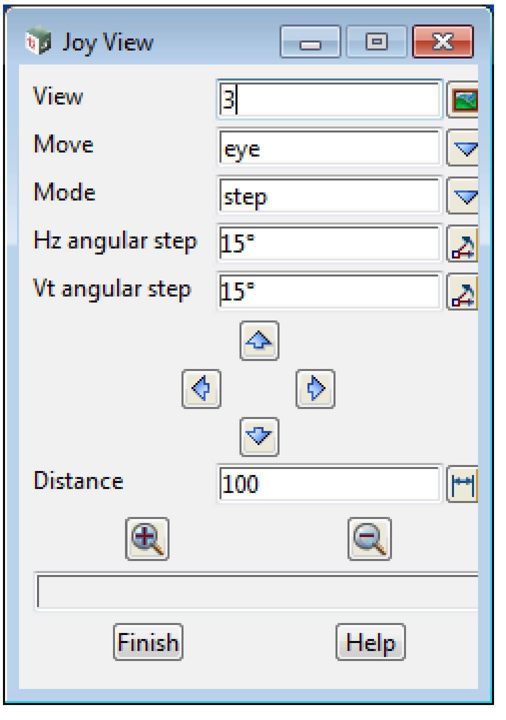
9.2.3 Joy for View

Position of option on menu: **Perspective View Menu** **View =>Joy**

The hardest thing in a perspective view is finding a good viewing position. The joy option makes it easy to construct a new perspective viewing position relative to the existing position.

The panel can be used to move the eye or target point by rotating them through a given angle to the left, right, up or down, or to move one towards or away from the other. It is also possible to move the eye and target points keeping the distance between them the same.

Selecting **Joy** fires up the **Joy view** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box	current view	available views

view to set joy parameters for.

Move	choice box	eye	eye, target, eye & target

move the eye point, the target point or both.

Mode	choice box	step	step, continuous

in step mode, the view is modified each time an appropriate button is picked. In continuous mode, after the view is redrawn, the picked joy option is continually re-applied until either a new joy button is picked, button LB is clicked anywhere else in the joy panel.

Hz angular step	angle box	15	

angle in degrees (in 4.17.1 HP Notation*) that is used with the joy buttons* **Left** *and* **Right** *when moving just the eye or target point.*

Vt angular step	angle box	15	

angle in degrees (in 4.17.1 HP Notation*) that is used with the joy buttons* **Up** *and* **Down** *when moving just the eye or target point.*

Joy Buttons

*The Joy buttons **Up**, **Down**, **Left**, **Right**, **In** and **Out** are used to modify the perspective parameters and then redraw the view with the new parameters.*

*The effect of each joy button depends on whether the **Move** field is set to **eye**, **target** or **eye & target**.*

Up, Down, Left, Right buttons

Move field set to eye

Up/Down button

*rotate the eye-target line about the target point up/down by the angle given in the **Angle step** field. That is, the eye point is moved upwards/downwards, the target point kept fixed.*

Left/Right button

*rotate the eye-target line about the target point to the left/right by the angle given in the **Angle step** field. The target point kept fixed.*

In/Out button

*move the eye point towards/away the target point along the eye-target line by the distance given in the **Distance** field. The target point kept fixed.*

Move field set to target

*similar to the movements when the **Move** field is set to **eye** except the eye point is kept fixed and the target point is moved.*

Move field set to eye & target

Up/Down/Left/Right button

*the eye and target points are both moved up/down/left/right by the distance given in the **Distance** field. That is, the entire eye-target line is moved up/down/left/right and neither the eye or target point is kept fixed.*

Distance input 100

distance used to move the eye or target point.

In, Out button

*move the eye and target points along the eye-target line by the distance given in the **Distance** field. The direction is towards/away the target point.*

How to Use the Panel

(a) The **move**, **mode**, **angular step** and **distance** field are given the desired values and then one of the joy buttons (up, down, left, right, in or out) is selected. The perspective view is then redrawn

(b) If the mode is set to continuous, once drawing is completed, the selected joy option will be re-applied. This sequence is repeated until either a new joy button is picked, button LB is clicked anywhere else in the joy panel or a **c** key is typed.

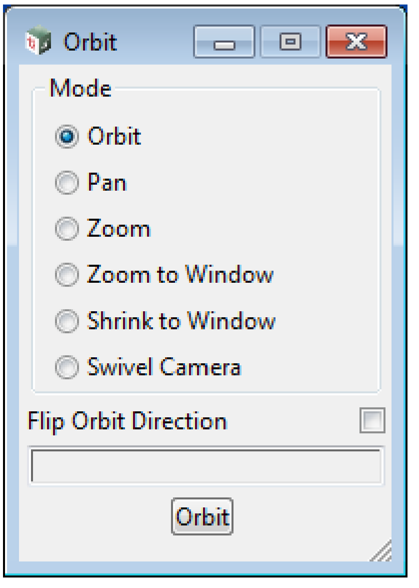
Continue to 9.2.4 Orbit or return to 9.2 Perspective View Menu.

9.2.4 Orbit

Position of option on menu: **Perspective View Menu** **View =>Orbit**

The orbit option quickly rotates the data around in the perspective view.

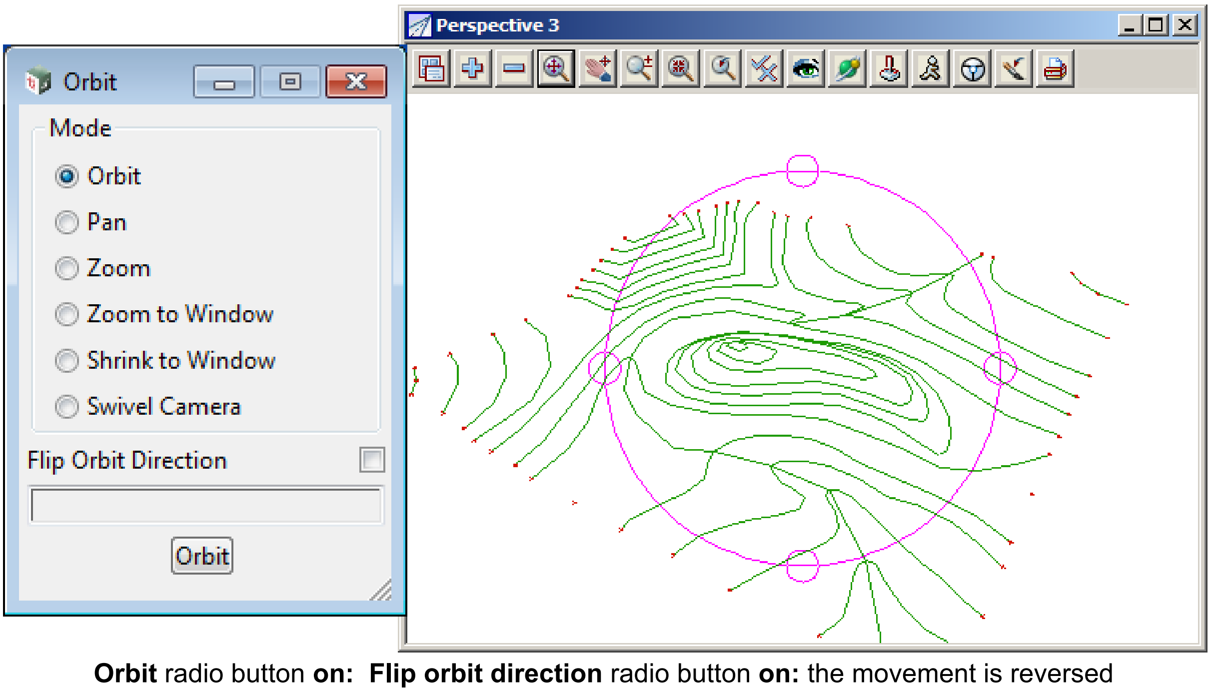
Selecting **Orbit** fires up the **Orbit** panel.



To use Orbit:

Orbit radio button **on:** **Flip orbit direction** radio button **off:**

when the left button is clicked in the perspective view, the orbit circle appears and
*holding LB down and **moving up** in the orbit circle tilts the **object up***
*holding LB down and **moving down** in the orbit circle tilts the **object down***
*holding LB down and **moving left** rotates the **object to the left***
*holding LB down and **moving right** rotates the **object to the right***



Orbit radio button **on:** **Flip orbit direction** radio button **on:** the movement is reversed

Pan radio button **on:**

*holding LB down and **moving up** in the view **raises** the **object***
*holding LB down and **moving down** in the view **lowers** the **object***
*holding LB down and **moving left** in the view **pans** the **object to the left***
*holding LB down and **moving right** in the view **pans** the **object to the right***

Zoom radio button **on:**

*holding LB down and **moving up** in the view **zooms into** the object*
*holding LB down and **moving down** in the view **zooms out** from the object*

Zoom to window radio button **on:**

when a zoom box is drawn on the view, the perspective is modified so that the contents of the zoom box fill the entire view.

Shrink to window radio button **on:**

when a zoom box is drawn on the view, the perspective is modified so that the contents of the window are shrunk to fit into the zoom box.

Swivel camera radio button **on:**

holding LB down moves the target point.

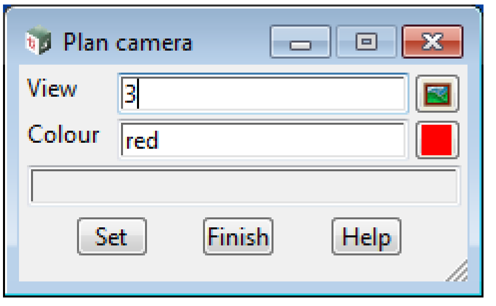
Continue to 9.2.5 Plan Camera or return to 9.2 Perspective View Menu.

9.2.5 Plan Camera

Position of option on menu: **Perspective View Menu** **View =>Plan Camera**

This creates a camera for the current perspective OpenGL view, which can be controlled from a plan view. This allows you to see where a camera is and where it is looking and move those points together or independently.

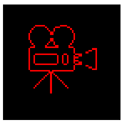
Selecting **Plan Camera**  brings up the **Plan Camera** panel.



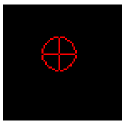
The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box		select view
the view to control			
Colour	colour box		
the colour to draw the plan camera and target symbol in			
Set	button		
Create the plan camera			

Setting the plan camera will create two symbols on your plan view, as shown below:



Camera / Eye



The target

To move the camera, simply click on one of these images and drag it. The perspective OpenGL view will move with it. If you wish to lift or lower the selected component, use the scroll wheel on your mouse. The distance you move is proportional to the zoom level on your plan view.

Key Strokes

There are a number of key strokes available when running the plan camera.

(**S**)et

sets a component (x, y, z) coordinate of the camera to be equal to the target or vice versa

(**L**)ock

locks the camera and target together, so they will maintain the same distance from each other when *either one or the other is moved.*

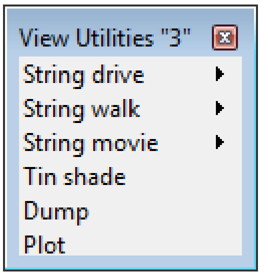
Continue to 9.2.6 Perspective Utilities or return to 9.2 Perspective View Menu.

9.2.6 Perspective Utilities

Position of option on menu: **Perspective View Menu** **View => Utilities**

The **Utilities** menu contains miscellaneous options involving the perspective view.

The **Utilities** walk-right menu is



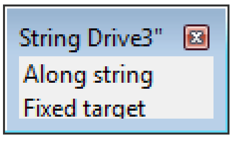
For the option *String drive*, go to [9.2.6.1 String Drive](#)
String walk [9.2.6.2 String Walk](#)
String movie [9.2.6.3 String Movie](#)
Tin shade [9.2.6.4 Tin Shade](#)
Dump [9.2.6.5 Dump](#)
Plot [9.2.6.6 Perspective Plot](#)

9.2.6.1 String Drive

Position of menu: **Perspective View Menu** **View => Utilities =>String drive**

The string drive option makes it easy to construct the perspective view one gets when moving along a string (for example, driving along a road centre line).

The two options on the **String drive** walk-right menu are



and they specify the positioning of eye and target points by:

along string - the eye and target points are automatically moved along the selected string

fixed target - the eye point automatically moves along the selected string but the target point is a selected fixed point.

Note - the **string drive** option can be aborted by pressing the **<esc>** key or clicking RB.

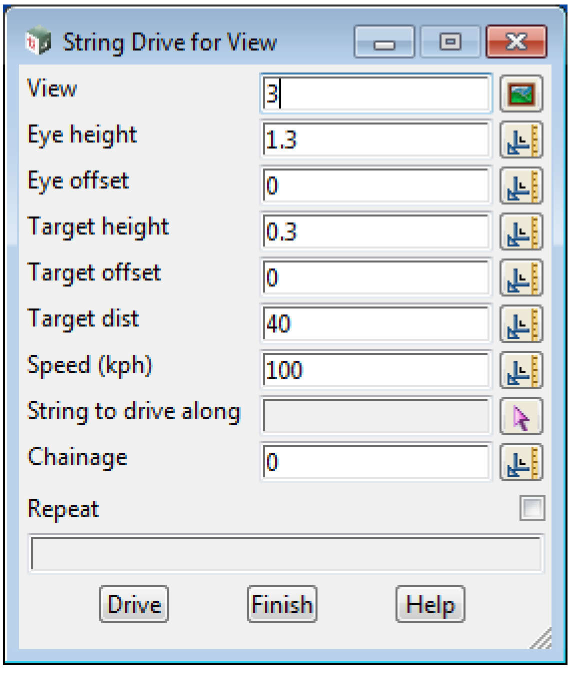
For the option Along string, go to [9.2.6.1.1 String Drive - Along String](#)
Fixed target [9.2.6.1.2 String Drive - Fixed Target](#)

Or return to [9.2 Perspective View Menu](#).

9.2.6.1.1 String Drive - Along String

Position of option on menu: **Perspective View Menu** **View => Utilities =>String drive =>Along string**

Selecting **Along string** fires up the **String Drive for View** panel.



For **string drive along a string**, a selected string is used to provide the (x, y, z) positions for the eye and target points.

The z-values can be adjusted by typing in a non-zero value for the eye and target heights in the **string drive** panel. These heights are relative to the z-values of the points on the string and are **added** to the z-values.

Like the new view option, the view is projected onto a focal plane and “clipped” using front and back clipping planes.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box	current view	available views

view to set drive parameters for

Eye height	input	1.3	

height of the eye viewing point above the picked string

Eye offset	input	0	

offset of the eye viewing point from the picked string

Target height	input	0.3	

height of the target viewing point above the picked string

Target offset	input	0	

offset of the target viewing point from the picked string

Target distance	input	40	

distance (in string chainage) from the eye point to the target point.

Speed (kph) input 100

speed in kilometres per hour (that is, one thousand base units per hour) that the eye-target points move along the string.

String to drive along string select box

a string to drive along is selected from any view. The string must have z-values.

Chainage input/output 0

the chainage that the eye point is at on the string. The chainage is updated as the eye and target points are automatically moved along the string. The drive can be set to start at a specific start chainage by simply entering the start chainage into the **chainage** *field and then selecting the* **Drive** *button.*

Repeat tick box

if ***ticked****, the drive starts again at the beginning of the string.*

Drive button

When the **drive** *button is selected, the eye and target points are placed above the string as specified in the panel fields, and the view redrawn. The eye and target points are then moved along the string at the given speed until either reaches the end of the string. The chainage of the eye point is continually updated.*

Notes

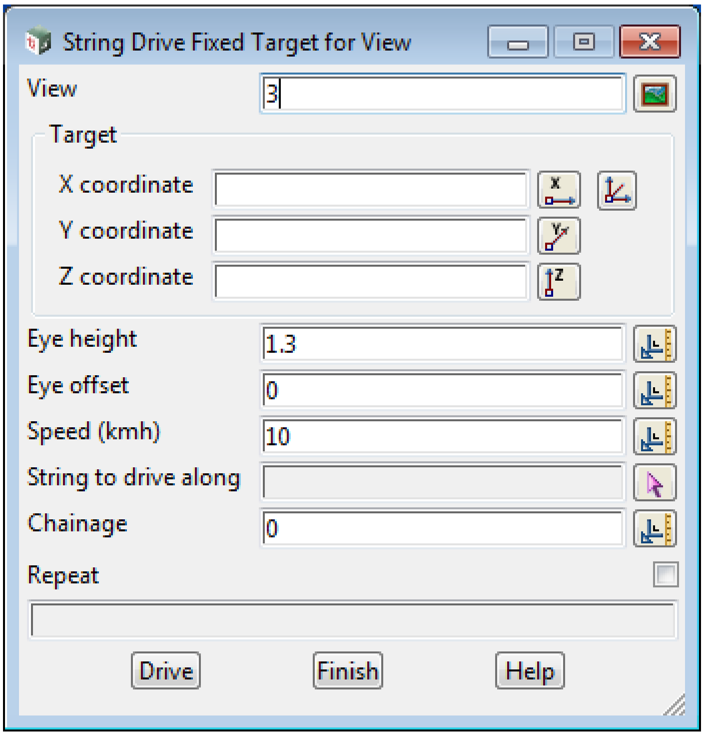
- the target distance and speed can be positive or negative.
- the eye and target points are shown on any view displaying the string being driven along. The eye point is drawn as a red cross, the target point, a green cross.
- if the front distance for the clipping plane of the perspective view is set to zero by the user, it is automatically reset to 2 so that the red cross drawn at the eye point is not visible in the perspective view.

Return to 9.2.6 Perspective Utilities or 9.2 Perspective View Menu.

9.2.6.1.2 String Drive - Fixed Target

Position of option on menu: **Perspective View Menu** **View => Utilities =>String drive =>Fixed target**

Selecting **Fixed target** fires up the **String Drive Fixed Target for View** panel.



For **string drive fixed target**, a selected string is used to provide the (x, y, z) positions for the eye point, and the target is a selected fixed point. The z-value for the eye is adjusted by adding to it the eye height in the **eye height** field in the **String Drive Fixed Target fro View** panel.

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

View view box current view available views

view to set drive parameters for.

Target xyz select box

a fixed target point is selected from any view. The point's co-ordinates are displayed in the X, Y, Z coordinates panel fields.

Eye height input 1.3

height of the eye viewing point above the picked string

Eye offset input 0

offset of the eye viewing point from the picked string

Speed (kph) input 100

speed in kilometres per hour (that is, in one thousand of the base units per hour that the eye point moves along the string.

String to drive along string select box

a string to drive along is selected from any view. The string must have z-values.

Chainage input/output 0

the chainage that the eye point is at on the string. The chainage is updated as the eye point automatically moves along the string. The drive can be set to start at a specific start chainage by simply entering the start chainage into the **chainage** *field and then selecting the* **Drive** *button.*

Target button

a fixed target point is selected from any view. The point's co-ordinates are displayed in the **target** *panel field.*

Repeat tick box

if ***ticked****, the drive starts again at the beginning of the string.*

Drive button

When the **drive** *button is selected, the eye point is placed above the string at the chainage given in the* **chainage** *field, and the view redrawn. The eye point is then moved along the string at the given speed until it reaches the end of the string. The chainage of the eye point is continually updated.*

Notes

- the speed can be positive or negative.
- the eye point is shown on any view displaying the string being driven along. The eye point is drawn as a red cross, the target point, a green cross.
- if the front distance for the clipping plane of the perspective view is set to zero by the user, it is automatically reset to 2 so that the red cross drawn at the eye point is not visible in the perspective view

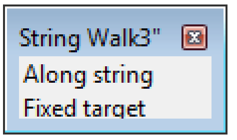
Return to 9.2.6 Perspective Utilities or 9.2 Perspective View Menu.

9.2.6.2 String Walk

Position of menu: **Perspective View Menu** **View => Utilities =>String walk**

The *string walk* option makes it easy to construct the perspective view one gets when walking along a string (for example, walking along a road centre line).

The two options on the **String Walk** walk-right menu are



and they specify the positioning of eye and target points by:

along string - the eye and target points are automatically moved along the selected string

fixed target - the eye point automatically moves along the selected string but the target point is a selected fixed point.

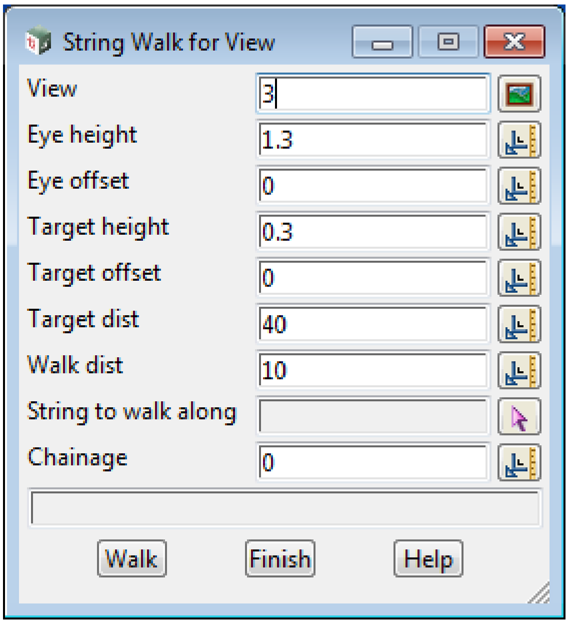
For the option Along string, go to 9.2.6.2.1 String Walk - Along String
Fixed target 9.2.6.2.2 String Walk - Fixed Target

Or to 9.2.6 Perspective Utilities or 9.2 Perspective View Menu.

9.2.6.2.1 String Walk - Along String

Position of option on menu: **Perspective View Menu** **View => Utilities =>String walk =>Along string**

Selecting **along string** fires up the **string walk for view** panel



For **string walk along a string**, a selected string is used to provide the (x, y, z) positions for the eye and target points.

The z-values can be adjusted by typing in a non-zero value for the eye and target heights in the **string walk** panel. These heights are relative to the z-values of the points on the string and are **added** to the z-values.

Like the new view option, the view is projected onto a focal plane and “clipped” using front and back clipping planes.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box	current view	available views

view to set walk parameters for.

Eye height	input	1.3	

height of the eye viewing point above the picked string

Eye offset	input	0	

offset of the eye viewing point from the picked string

Target height	input	0.3	

height of the target viewing point above the picked string

Target offset	input	0	

offset of the target viewing point from the picked string

Target distance	input	40	

distance (in string chainage) from the eye point to the target point.

String to walk along	string select box		

a string to walk along is selected from any view. The string must have z-values.

Walk dist input 10

distance to move the eye point along string from the previous eye point chainage. The target point is also moved the same distance along the eye-target line so that the eye-target distance is kept constant.

Chainage input/output 0

the chainage that the eye point is at on the string. The chainage is updated whenever the walk button is activated. The eye point can be set to a specific chainage by simply entering the chainage into the **chainage** *field and selecting the walk button.*

Walk button

the perspective parameters are defined according to the information in the panel. The first time the **Walk** *button is selected, the eye point is placed above the string at the chainage given in the* **chainage** *field. For subsequent selections of the* **Walk** *button, the eye and target points are moved along the eye-target line by the chainage distance given in the* **walk dist** *field.*

The view is redrawn with the new view parameters and the new eye-chainage displayed in the panel message area and the chainage field.

Notes

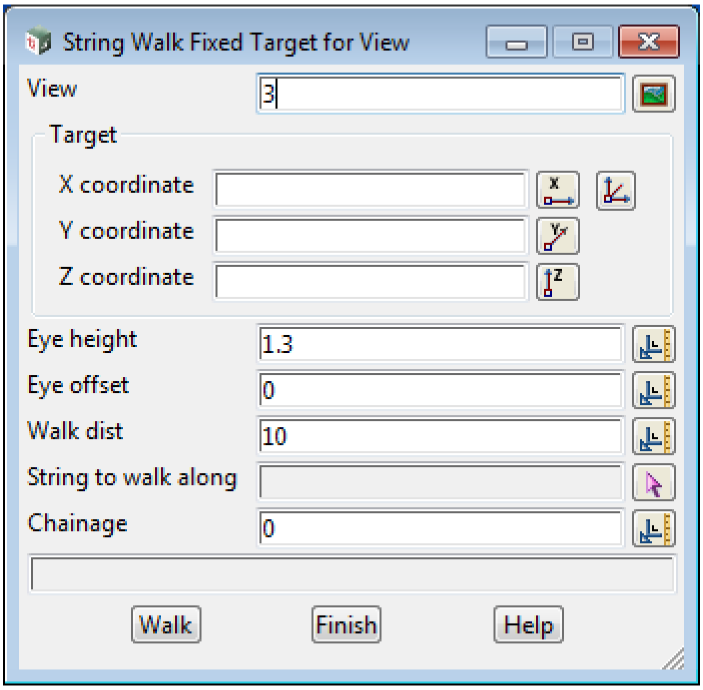
- the target and move distances can be positive or negative.
- the eye and target points are shown on any view displaying the string being walked along. The eye point is drawn as a red cross, the target point, a green cross.
- if the front distance for the clipping plane of the perspective view is set to zero by the user, it is automatically reset to 2 so that the red cross drawn at the eye point is not visible in the perspective view.

Return to 9.2.6 Perspective Utilities or 9.2 Perspective View Menu.

9.2.6.2.2 String Walk - Fixed Target

Position of option on menu: **Perspective View Menu** **View => Utilities =>String walk =>Fixed target**

Selecting **Fixed target** fires up the **String Walk Fixed Target for View** panel.



For **string walk fixed target**, a selected string is used to provide the (x, y, z) positions for the eye point, and the target is a selected fixed point. The z-value for the eye is adjusted by adding to it the eye height in the **eye height** field in the **String Walk Fixed target for View** panel.

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

View view box current view available views

view to set walk parameters for.

Target xyz box

a fixed target point is selected from any view. The point's co-ordinates are displayed in the X, Y, Z coordinates panel fields.

Eye height input 1.3

height of the eye viewing point above the picked string

Eye offset input 0

offset of the eye viewing point from the picked string

Walk dist input 10

distance to move the eye point along string from the previous eye point chainage.

String to walk along string select

a string to walk along is selected from any view. The string must have z-values.

Chainage input/output 0

the chainage that the eye point is at on the string. The chainage is updated as the eye point automatically moves along the string. The walk can be set to start at a specific start chainage by simply *entering the start chainage into the* **chainage** *field and then selecting the* **Walk** *button.*

Walk button

the perspective parameters are defined according to the information in the panel. The first time the **Walk** *button is selected, the eye point is placed above the string at the chainage given in the* **chainage** *field. For subsequent selections of the* **Walk** *button, the eye point is moved along the string by the chainage distance given in the* **walk dist** *field.*

The view is redrawn with the new view parameters and the new eye-chainage displayed in the panel message area and the chainage field.

Notes

- the walk distance can be positive or negative.
- the eye point is shown on any view displaying the string being walked along. The eye point is drawn as a red cross, the target point, a green cross.
- if the front distance for the clipping plane of the perspective view is set to zero by the user, it is automatically reset to 2 so that the red cross drawn at the eye point is not visible in the perspective view

Return to 9.2.6 Perspective Utilities or 9.2 Perspective View Menu.

9.2.6.3 String Movie

Position of menu: **Perspective View Menu** **View => Utilities =>String movie**

The **String movie** option makes it easy to construct and save to disk the sequence of perspective views one gets when walking along a string (for example, walking along a road centre line).

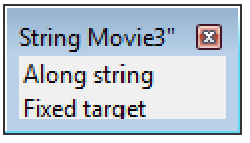
The screen images are automatically written out to disk in gif format so that they can be replayed at a later time.

This is especially useful when it takes a while to construct each view, for example, when hidden line is turned on.

Under NT/95, the screen images are written to an **avi** file.

Under Unix, the screen images are written to a **gif** file and there can be up to 99,999 of them in the one movie.

The two options on the **String movie** walk-right menu are



and they specify the positioning of eye and target points by:

along string - the eye and target points are automatically moved along the selected string

fixed target - the eye point automatically moves along the selected string but the target point is a fixed selected point.

Note - the **string movie** option can be aborted by pressing the **<esc>** key or clicking RB.

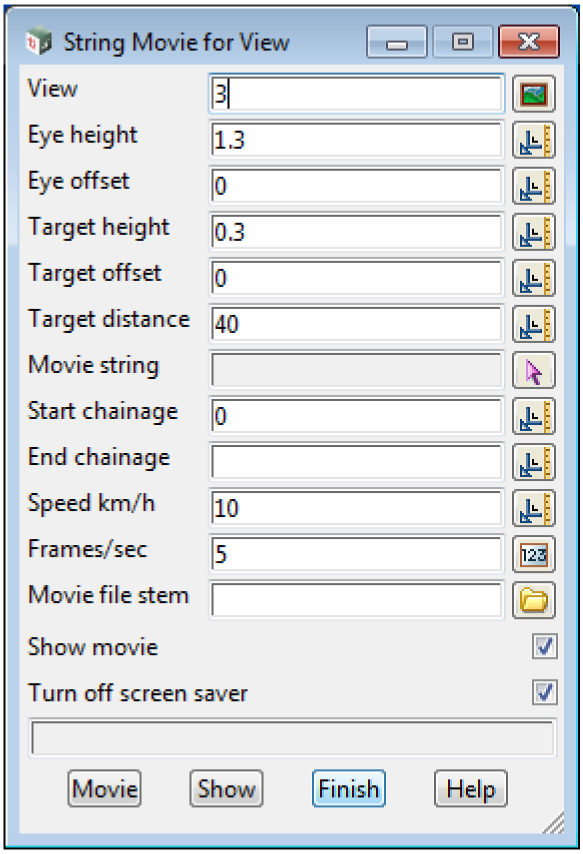
For the option Along string, go to 9.2.6.3.1 String Movie - Along String
Fixed target 9.2.6.3.2 String Movie - Fixed Target

Or return to 9.2.6 Perspective Utilities or 9.2 Perspective View Menu.

9.2.6.3.1 String Movie - Along String

Position of option on menu: **Perspective View Menu** **View => Utilities =>String movie =>Along string**

Selecting **Along string** fires up the **String Movie for View** panel



For **string movie along a string**, a selected string is used to provide the (x, y, z) positions for the eye and target points.

The z-values can be adjusted by typing in a non-zero value for the eye and target heights in the **string movie** panel. These heights are relative to the z-values of the points on the string and are **added** to the z-values.

Like the new view option, the view is projected onto a focal plane and “clipped” using front and back clipping planes

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box	current view	available views

view to set movie parameters for.

Eye height	input	1.3	

height of the eye viewing point above the picked string

Eye offset	input	0	

offset of the eye viewing point from the picked string

Target height	input	0.3	

height of the target viewing point above the picked string

Target offset input 0

offset of the target viewing point from the picked string

Target distance input 40

distance (in string chainage) from the eye point to the target point.

Movie string string select

a string to drive along is selected from any view. The string must have z-values.

Start chainage input/output 0

the chainage that the eye point starts on the string for the movie.

End chainage input/output 0

the chainage to stop the movie at.

Speed (kph) input 10

speed in kilometres per hour (that is, one thousand base units per hour) that the eye-target points move along the string.

Frames/sec) input 5

number of frames per second to produce for the movie.

Movie file stem input 4d-

each frame of the drive in the view is written to disk in avi format using the **movie file stem** *plus the ending* ***.avi*** *as the file name.*

Show movie tick box tick

if ***ticked****, after all the views along the string are written to disk, a new window, the same size as the perspective view, will be created and movie then run.*

Turn off screen saver tick box tick

if ***ticked****, the screen saver is disabled so it doesn't come on whilst creating the movie.*

Movie button

the perspective parameters are defined according to the information in the panel. When the **Movie** *button is selected, the eye point is placed above the string at the chainage given in the* **chainage** *field. The view is drawn with these view parameters. After that view is processed and written to disk, the eye and target points are moved along the eye-target line and frames written out to simulate the given speed.*

Show button

This button is used to display a previously created movie. If selected, the movie created with the stem given in the **movie file stem** *field is displayed. If Unix, there is a pause between each frame by the number of seconds given in the* **delay** *field.*

Displaying the Movie

If a movie has just been created and the **show movie** field is set to **tick**, or an existing movie name is given in the **movie file stem** field and the **show** button is selected, then a new window will be created and the movie will be displayed frame by frame. Outside of ***12d Model***, the movie can be displayed by simply double clicking on the **avi** file.

Notes

- the target and move distances can be positive or negative.
- the eye and target points are shown on any view displaying the string being walked along. The eye point is drawn as a red cross, the target point, a green cross.
- if the front distance for the clipping plane of the perspective view is set to zero by the user, it is

automatically reset to 2 so that the red cross drawn at the eye point is not visible in the perspective view.

- the movie can also be displayed from outside 12d Model by simply double clicking on the avi file

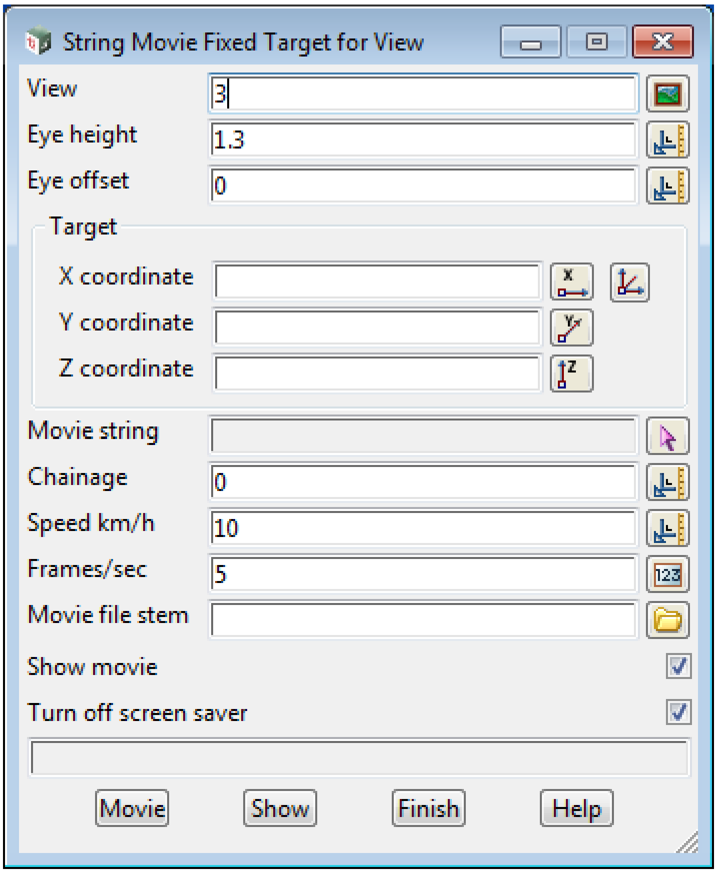
  movie-file-stem.avi

Return to 9.2.6 Perspective Utilities or 9.2 Perspective View Menu.

9.2.6.3.2 String Movie - Fixed Target

Position of option on menu: **Perspective View Menu** **View => Utilities =>String movie =>Fixed target**

Selecting **Fixed target** fires up the **String Movie Fixed Target for View** panel.



For **string movie fixed target**, a selected string is used to provide the (x, y, z) positions for the eye point, and the target is a selected fixed point. The z-value for the eye is adjusted by adding to it the eye height in the **eye height** field in the **String Movie Fixed Target for View** panel.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box	current view	available views

view to set movie parameters for.

Eye height	input	1.3	

height of the eye viewing point above the picked string

Eye offset	input	0	

offset of the eye viewing point from the picked string

Target	xyz box		

a fixed target point is selected from any view. The point's co-ordinates are displayed in the X, Y, Z coordinates panel fields.

Movie string	string select		

a string to drive along is selected from any view. The string must have z-values.

Chainage input/output 0

the chainage that the eye point starts on the string for the movie.

Speed (kph) input 10

speed in kilometres per hour (that is, one thousand base units per hour) that the eye-target points move along the string.

Frames/sec) input 5

number of frames per second to produce for the movie.

Movie file stem input 4d-

each frame of the drive in the view is written to disk in avi format using the **movie file stem** *plus the ending* ***.avi*** *as the file name.*

Show movie tick box tick

if ***ticked****, after all the views along the string are written to disk, the movie is then run.*

Turn off screen saver tick box tick

if ***ticked****, the screen saver is disabled so it doesn't come on whilst creating the movie.*

Movie button

the perspective parameters are defined according to the information in the panel. When the **Movie** *button is selected, the eye point is placed above the string at the chainage given in the* **chainage** *field. The view is drawn with these view parameters. After that view is processed and written to disk, the eye and target points are moved along the eye-target line by the chainage distance given in the* **step distance** *field and the new view drawn and processed.*

This sequence is repeated until the end of the string is reached or the ***c*** *key is pressed.*

Show button

This button is used to display a previously created movie. If selected, the movie created with the stem given in the **movie file stem** *field is displayed.*

Return to 9.2.6 Perspective Utilities or 9.2 Perspective View Menu.

9.2.6.4 Tin Shade

Position of option on menu: **Perspective View Menu** **View => Utilities =>Tin shade**

In ***12d Model***, a landscape is represented by a triangulated surface - a tin. Each triangle in the tin is a part of a plane and has its own colour.

In a perspective view, each triangle can be drawn with a colour that is modified depending on the angle that the triangle makes with the sun (a point light source at infinity). This is called a **flat shade.**

If the triangles are drawn in a back to front order, then when the shaded landscape is drawn, any triangles in the foreground obscure triangles in the background and it looks like a hidden view.

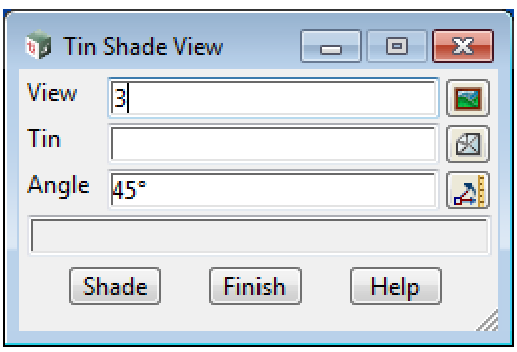
If **Shade** is set for a view, all the triangles in all the tins on the view are drawn as shaded triangles and the triangles are drawn in a back to front order.

Once the **Shade** is set on, a shade will be done on the view whenever the view parameters are changed. A plot of the shaded view can be made using the **Dump** option on the view.

The **Tin shade** option draws the selected tin over the perspective view as a shade. The selected tin does not have to be on the view for the shade to function.

When the view is refreshed, the shade is lost (the **Shade** option under settings is used for setting a shade permanently on). A plot of the shaded view can be made using the **Dump** option on the view.

Selecting **Tin shade** fires up the **Tin Shade View** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box	current view	available views

view to set shade parameters for.

Tin	input		available tins

name of the tin to be used for the shade.

Angle	input	45	

angle of the sun.

Shade	button		

draw the given tin on the view as a shaded tin.

Return to 9.2.6 Perspective Utilities or 9.2 Perspective View Menu.

9.2.6.5 Dump

Position of option on menu: **Perspective View Menu** **View => Utilities =>Dump**

This option is used to write the view image out to disk in a user selected format. It is the same as the **Dump** option for a plan view.

Return to 9.2.6 Perspective Utilities or 9.2 Perspective View Menu.

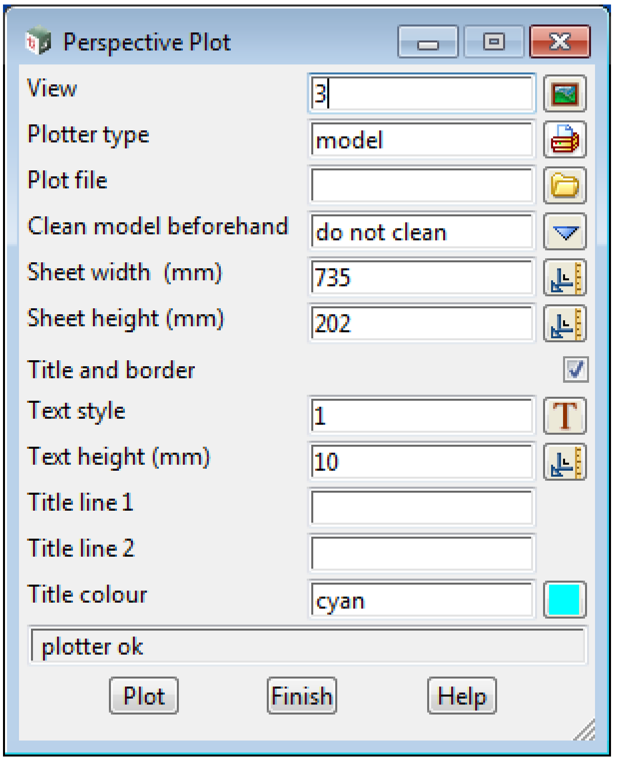
9.2.6.6 Perspective Plot

Position of option on menu: **Perspective View Menu** **View => Utilities =>Perspective plot**

The **Perspective plot** option is used to make a plot file of all the information displayed in the perspective view.

The user simply gives a sheet width and height and ***12d Model*** calculates the scaling factor required to best fit the plot of the view to the sheet.

Selecting **Plot** displays the **Perspective Plot** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box	current view	available views
view to set perspective plot parameters for.			
Plotter type	input	hp	hp, dxf, postscript *etc.*
file format for the plot information.			
Plot file	input	depends on plotter type	

stem of the name of the file to write the plot of the view to. The name ending is added automatically and *depends on the plotter type.*

Clean model beforehand	choice box		do not clean prompt for clean always clean

only applicable if plotting to a model.

if ***always clean,*** *the model is cleaned before the plot is created.*
*If p****rompt for clean****, the user is prompted that the model will be cleaned before the plot is created.*
If ***do not clean****, the model is not cleaned before the plot is created.*

Sheet width/height (mm)	input		

if a value is entered by the user and an <enter> given, the height/width required by the plot is calculated and displayed in the **sheet height/width** *field. The units for sheet width and height are* ***millimetres****.*

If both Sheet width and height are given with no <enter>, then a scaling factor is calculated by 12d *Model to best fit the plot of the view to a rectangle of size given by the sheet width and height.*

Title and border	tick box		

if ***ticked****, a border and two lines of title are placed on the bottom of the plot*

Title line 1/2	input		

first/second line of title information

Title height (mm)	input	10	

height (in millimetres) to draw the two lines of title information

Title colour	input	cyan	available colours

colour used for the border and the title information

Plot	button		

write out the plot of the information displayed in the view to the file given in the **plot file** *field. The format of the file is given by the* **plotter type.**

Return to 9.2.6 Perspective Utilities or 9.2 Perspective View Menu.

9.3 Section View Menu

Position of menu: **Section View Menu** **View**

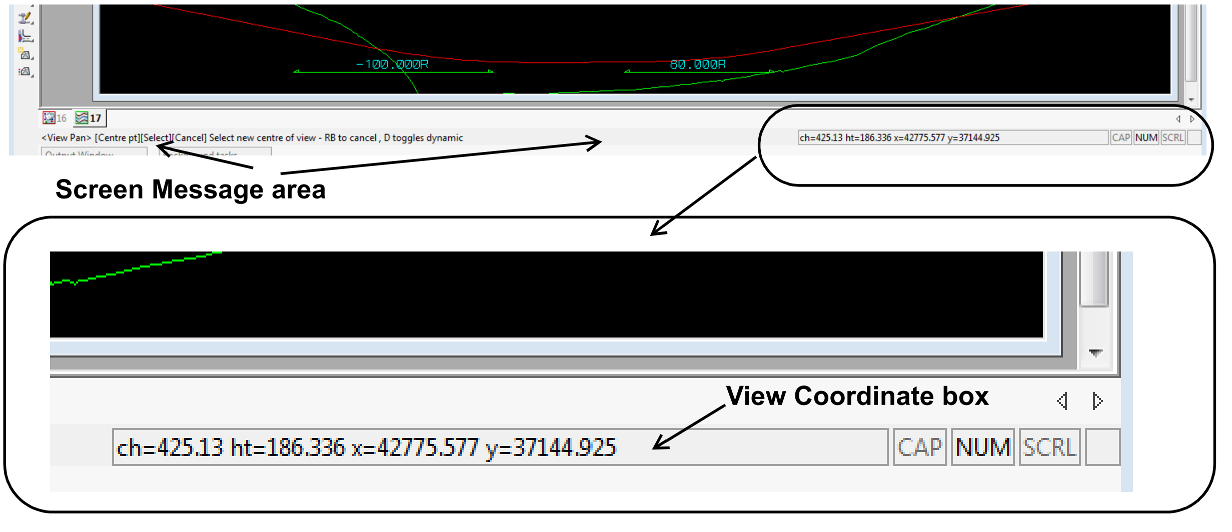
The section view is used to display string profiles and sections through tins and service items. The horizontal units are taken to be the **chainage** along a particular string called the **primary string** for the view. The z-values (heights) are the vertical units for the section. Hence, the section view is a (chainage, z-value) diagram with respect to the primary string on the view. The primary string's model and name is displayed in the view title area of the section view.

Since a tin represents a two dimension continuous surface, sectioning through a tin along a string appears as a continuous line string in a section view. For example, the natural surface on a road long-section is the section of the road centre-line through the natural surface tin.

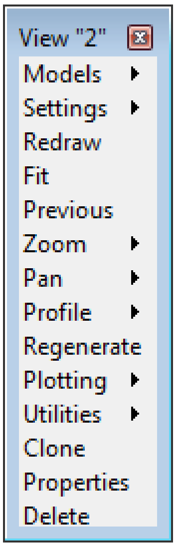
When a corridor is set, any strings in models added to the section view (service items) will be drawn on the section view wherever they are inside the corridor.

A section view has a **vertical exaggeration** and whenever a string is profiled on the view, the vertical exaggeration is displayed inside square brackets after the view name in the view title area. For example **[10x]** is a vertical exaggeration of 10.

When in a Section view that has a string profiled on it, the chainage, height, and (x,y) coordinates of the cursor position are also displayed in the **View Coordinate** box in the **Screen Message Area** at the bottom of the *12d Model* window.



The **Section views** menu is



For the options:

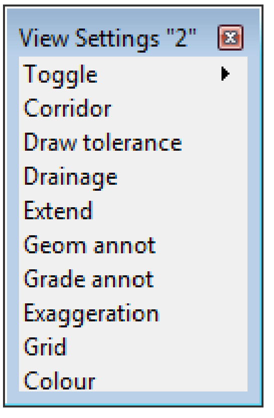
Models, go to	9.4.1 Model Ops
Settings	9.3.1 Section View Settings
Redraw	9.4.2 Redraw
Fit	9.4.3 Fit
Previous	9.4.4 Previous
Zoom	9.1.2 Zoom
Pan	9.1.3 Pan
Profile	9.3.2 Profile
Regenerate	9.3.3 Regenerate
Plotting	9.3.4 Section Plotting
Utilities	9.3.5 Section Utilities
Clone	9.4.5 Clone
Properties	9.4.6 Properties
Delete	9.4.7 Delete

9.3.1 Section View Settings

Position of menu: **Section View Menu** **View => Settings**

If the **Settings** option is picked rather than moving onto the walking right, then the **Toggle** menu from the **Toggle** walk-right menu is displayed on the screen. The **Toggle** menu will be described in the next section.

The **Settings** walk-right menu for the section view is



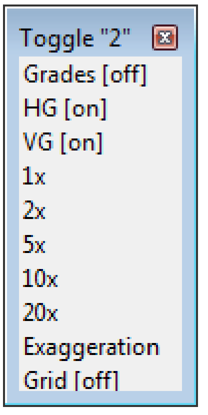
For the option:

Toggle go to	9.3.1.1 Section View Toggle
Corridor	9.3.1.2 Corridor
Draw tolerance	9.3.1.3 Draw Tolerance
Drainage	9.3.1.4 Drainage HGL
Extend	9.3.1.5 Extend
Geom annot	9.3.1.6 Geometry Annotation
Grade annot	9.3.1.7 Grade Annotation
Exaggeration	9.3.1.8 Section View Exaggeration
Grid	9.1.1.22 Grid on View
Colour	9.1.1.23 View Background Colour

9.3.1.1 Section View Toggle

Position of menu: **Section View Menu** **View => Settings =>Toggle**

The **Toggle** walk right brings up the **Toggle** section view menu.



Selecting any options from this menu will toggle the option on/off.

Continue to 9.3.1.2 Corridor or return to 9.3.5 Section Utilities to 9.3 Section View Menu.

9.3.1.2 Corridor

Position of option on menu: **Section View Menu** **View => Settings =>Corridor**

The standard section view is designed to profile any selected string and create and display sections along the profiled string through any tins in any models added to the section view.

Hence the section view is like a vertical ribbon tracing out the path of the primary string.

The **Corridor** option extends the ribbon to the left and right and projects any strings in any models added to the section view back onto the section view.

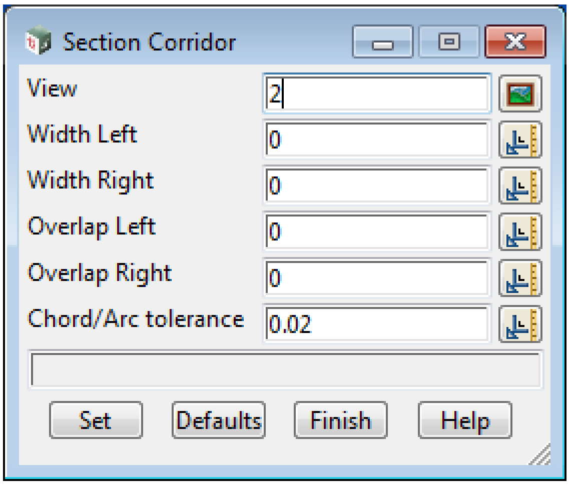
Consequently, the section view will display any strings or parts of strings that cross the corridor, run parallel to the primary string but stay within the corridor as well as displaying the standard sections through any tins in any models added to the section view.

The corridor is particularly useful for examining any **clashes** of strings with the primary string.

For example, services such as telephone cables, water and gas pipes and electricity cables could be modelled as 3d or pipe strings and added to the section view.

The section view with non-zero corridor widths would then display any of the services that lie within the corridor about the chosen primary string.

Selecting **Corridor** fires up the **Section Corridor** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	
View	view box	current view	available views

view to set corridor parameters for.

Width left/right	input	0.01

width of the corridor to the left/right of the primary string.

Overlap left/right	input	0.01

for each straight/arc in the primary string, extend the straight/arc to the left/right by the value of the **overlap left/right** *field before drawing the straight/arc. Section through the extended string.*

Chord/Arc tolerance	input	0.02

value of the chord to arc ratio to be used when approximating circles in alignments, arcs and pipeline strings.

Set	button

define the corridor parameters according to the information in the panel. The section view for the new corridor will then be calculated and displayed.

Defaults button

reset the corridor settings to the default values.

Continue to 9.3.1.3 Draw Tolerance or return to 9.3.5 Section Utilities to 9.3 Section View Menu.

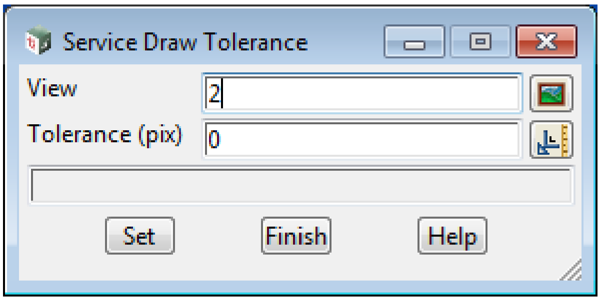
9.3.1.3 Draw Tolerance

Position of option on menu: **Section View Menu** **View => Settings =>Draw tolerance**

It is possible for the bits of services displayed on a section view to be very small and possibly too small to be easily seen. This is often the case for strings that are perpendicular to the corridor.

To overcome this visualization difficulty, a tolerance can be defined for the section view and when the size of any piece of a service is below the tolerance, it will be drawn as a cross.

Selecting **Draw tolerance** fires up the **Service Draw Tolerance** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	
View	view box	current view	available views

view to set service parameters for.

Tolerance (pix)	input	4

if the size of any piece of a service is below this tolerance, it will be drawn as a cross.

Set	button

set the service draw tolerance.

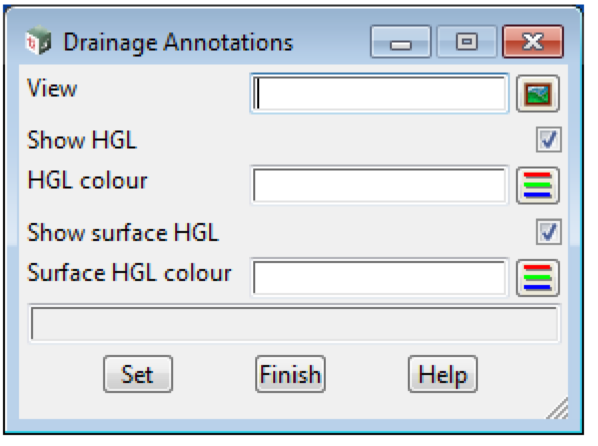
Continue to 9.3.1.4 Drainage HGL or return to 9.3.5 Section Utilities to 9.3 Section View Menu.

9.3.1.4 Drainage HGL

Position of option on menu: **Section View Menu** **View => Settings =>Drainage**

If the data exists in the drainage string, the HGL line can be drawn when profiling a drainage string.

Selecting **Drainage** fires up the **Drainage Annotations** panel.



The fields and buttons used in this panel have the following functions.

Field Description | Type | Defaults

View | input/output | current view | available views

view to set HGL parameters for.

Show HGL | tick box | tick

*if **ticked** then the HGL line is drawn for any drainage strings that are profiled (and have HGL data)*

HGL colour | colour box | | available colours

colour to draw the HGL line

Show surface HGL | tick box

if ticked then the surface HGL (the HGL of any bypass channels) will be shown on the long section

Note: not available for the Rational Method

Surface HGL colour | colour box | | available colours

colour to draw the HGL surface

Set | button

set the HGL drawing parameters.

Continue to 9.3.1.5 Extend or return to 9.3.5 Section Utilities to 9.3 Section View Menu.

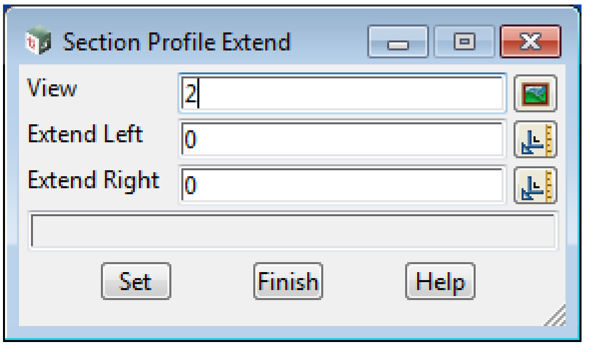
9.3.1.5 Extend

Position of option on menu: **Section View Menu** **View => Settings =>Extend**

When displaying the profile of a string on a section view, the user often wishes to see sections through the information before the string begins and after the string ends. This is particularly useful if it is intended to extend the string in either direction.

The **Extend** option allows the user to extend the profile length.

Selecting **Extend** fires up the **Section Profile Extend** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	
View	view box	current view	available views

view to set extension parameters for.

Extend left	input	0

pre-extend the profile by this distance before sectioning through any tins or models on the view.

Extend right	input	0

post-extend the profile by this distance before sectioning through any tins or models on the view.

Set	button

define the profile length according to the length of the primary string plus the two **extend** *lengths. The section view for the new length will then be calculated and displayed.*

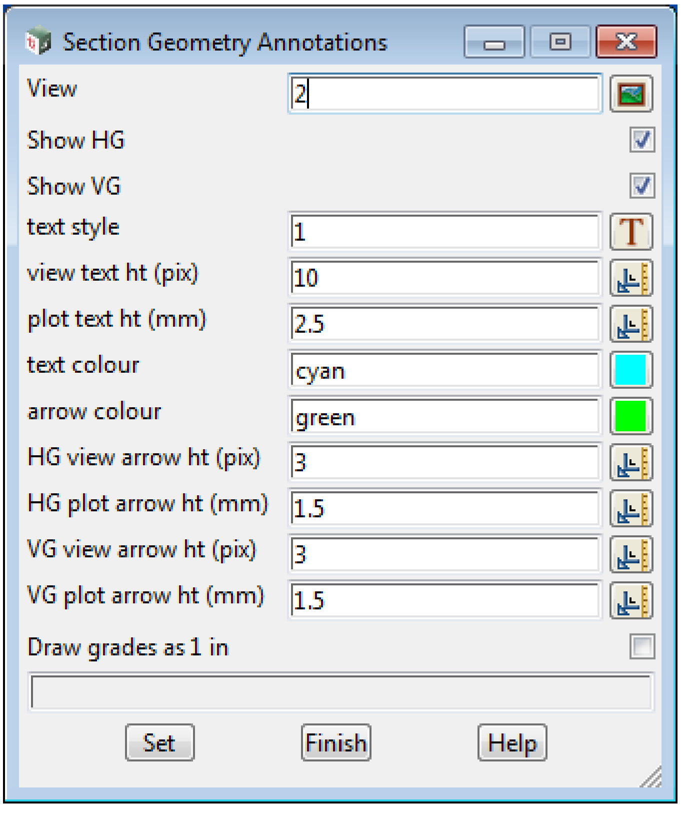
Continue to 9.3.1.6 Geometry Annotation or return to 9.3.5 Section Utilities to 9.3 Section View Menu.

9.3.1.6 Geometry Annotation

Position of option on menu: **Section View Menu** **View => Settings =>Geom annot**

The **Geom annot** option allows the user to specify whether horizontal geometry and/or vertical geometry is displayed for any alignment and pipeline strings profiled in the section view and hence on any section view plots.

Selecting **Geom annot** fires up the **Section Geometry Annotations** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box	current view	available views

view to set alignment geometry parameters for.

Show HG	tick box	tick	

*if **ticked**, the value of the radii and spiral lengths for any horizontal curves selected as the primary string will be shown plus an arrow indicating the chainage extent of the radii.*
*If **no ticked**, the horizontal geometry will not be displayed in the section view or on any plots of the section view.*

Show VG	tick box	tick	

*if **ticked**, the value of the curve length for any vertical curves selected as the primary string will be shown, plus an arrow indicating the chainage extent of the curve length. The percentage grades for any vertical straights will also be displayed with an arrow indicating the chainage extent of the vertical straight.*
*If **no ticked**, the vertical curve lengths and percentage grades will not be displayed in the section view or on any plots of the section view.*

Text style	input	1	

text style to be used for all text in the geometry annotation.

View text ht (pix) input 10

height (in pixels) to draw the geometry annotation values in the section view (the screen width is approximately 1000 pixels).

Plot text ht (mm) input 10

height (in mm) to draw the geometry annotation values on any plots of the section view.

Text colour input cyan available colours

colour of the geometry annotation values drawn on the screen and any plots.

Arrow colour input green available colours

colour of the HG arrows drawn on the screen and any plots.

HG view arrow ht (pix) input 3

height (in pixels) of the arrow head above the arrow line when drawing the HG arrows in the view.

HG plot arrow ht (mm) input 1.5

height (in mm) of the arrow head above the arrow line when drawing the HG arrows on any plots of the section view.

VG view arrow ht (pix) input 3

height (in pixels) of the arrow head above the arrow line when drawing the VG arrows in the view.

VG plot arrow ht (mm) input 1.5

height (in mm) of the arrow head above the arrow line when drawing the VG arrows on any plots of the section view.

Draw grades as 1 in tick box

if ticked, grades are drawn as 1:in values.
if not ticked, grades as drawn as percent (%)

Set button

set the section settings values to the values in the above panel fields. The section view is then redrawn using these values.

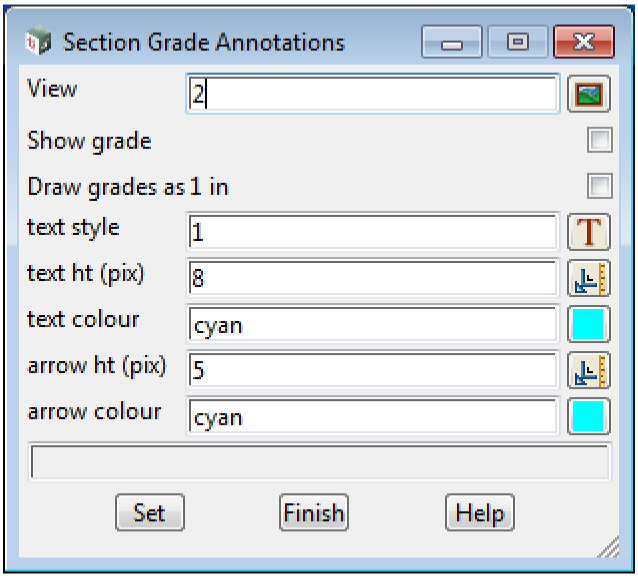
Continue to 9.3.1.7 Grade Annotation or return to 9.3.5 Section Utilities to 9.3 Section View Menu.

9.3.1.7 Grade Annotation

Position of option on menu: **Section View Menu** **View => Settings =>Grade annot**

The **grade annot** option allows the user to specify whether the grades and plan widths of string links are displaced for strings profiled in the section view.

Selecting **Grade annot** fires up the **Section Grade Annotations** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box	current view	available views

view to set grade annotation parameters for.

Show grade	tick box	tick	

*if **ticked**, the value of the grades (in percent grade) and plan widths of string links for any strings selected as the primary string will be shown plus an arrow indicating the chainage extent of the grade. If **no ticked**, the grades and widths will not be displayed in the section view or on any plots of the section view.*

Text style	input	1	

text style to be used for all text in the grade annotation.

Text ht (pix)	input	8	

height (in pixels) to draw the grade annotation values in the section view (the screen width is approximately 1000 pixels).

Text colour	input	cyan	available colours

colour of the grade annotation values drawn on the screen and any plots.

Arrow ht (pix)	input	3	

height (in pixels) of the arrow head above the arrow line when drawing the arrows in the section view.

Arrow colour	input	cyan	available colours

colour of the arrows drawn on the screen.

Set	button		

set the section settings values to the values in the above panel fields. The section view is then redrawn *using these values.*

Continue to 9.3.1.8 Section View Exaggeration or return to 9.3.5 Section Utilities to 9.3 Section View Menu.

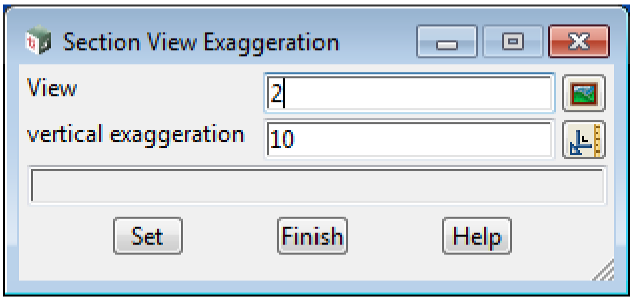
9.3.1.8 Section View Exaggeration

Position of option on menu: **Section View Menu** **View =>Settings =>Exaggeration**

The **Exaggeration** option allows the user to specify the vertical exaggeration for the section view. The heights (z values) are multiplied by the vertical exaggeration value before drawing on the section view.

Whenever a string is profiled on the section view, the **vertical exaggeration** is displayed inside square brackets after the view name in the view title area. For example **[10x]** is a vertical exaggeration of 10.

Selecting **Exaggeration** fires up the **Section View Exaggeration** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box	current view	available views

view to set exaggeration parameters for.

Vertical exaggeration	input	10	1,5,10

value to multiply the heights (z values) by before drawing in the section view.

Set	button		

set the vertical exaggeration to the value in the vertical exaggeration field.

Return to 9.3.5 Section Utilities to 9.3 Section View Menu.

9.3.2 Profile

Position of menu: **Section View Menu** **View => Profile**

The **Profile** option is used to display a long-section view of a selected string. That is, the coordinate system for the section view is defined in terms of the chosen string by

(a) the chainage along the selected string defines the horizontal axis for the section view

(b) the vertical axis is the z-axis (heights).

Hence a (chainage,height) diagram of the selected string is drawn in the section view. The scale is automatically chosen so that all of the selected string fits into the section view.

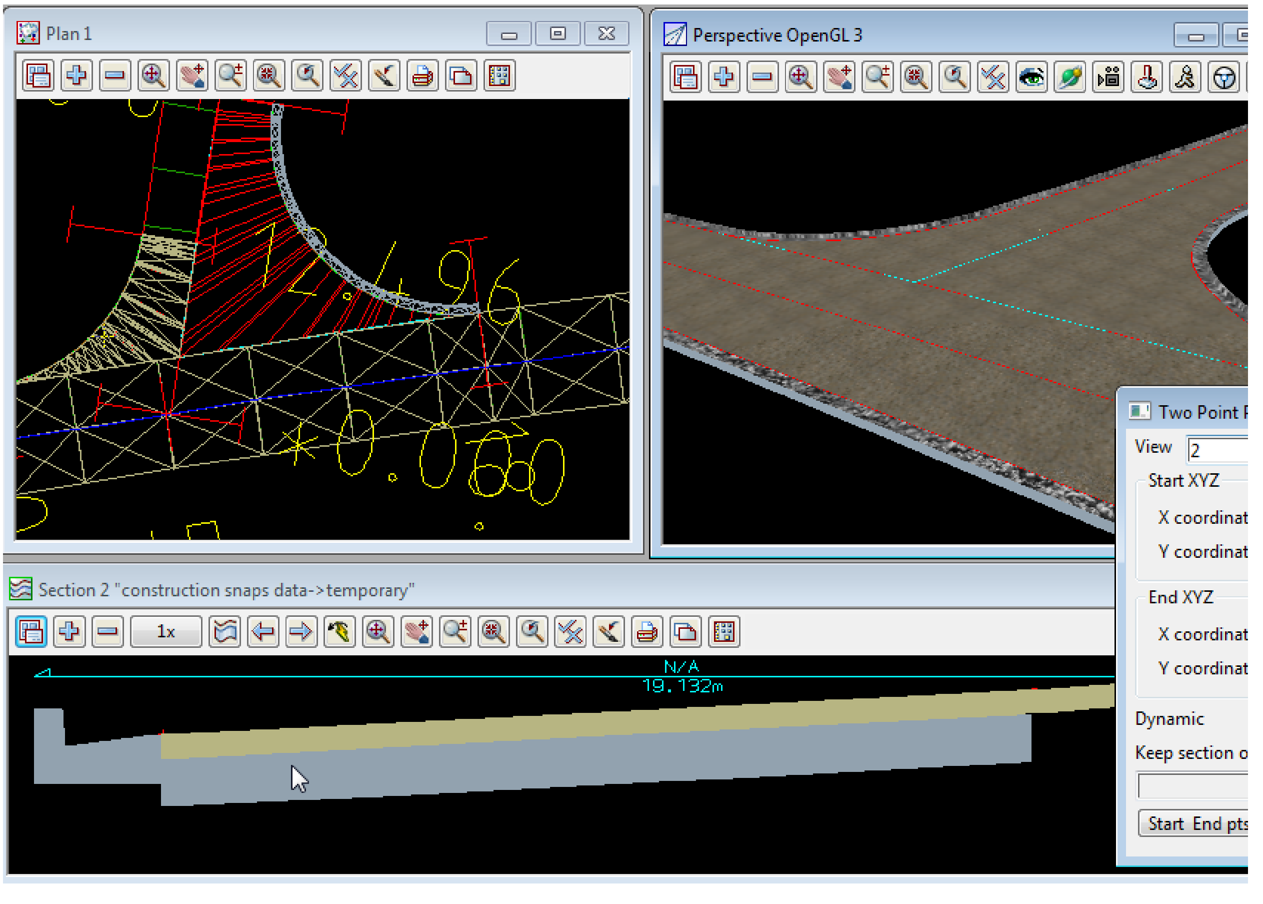
The string selected to be profiled is called the **primary string** for the section view.

The model and name of the primary string is displayed in the section view's title area (after the vertical exaggeration).

The primary string is drawn on the section view in the primary string's colour.

If the primary string is an alignment or pipeline string, the radii and chainage extent of any horizontal curves can be displayed in the section view plus any vertical curve lengths and percent-grades of vertical straights. For other strings, the grade and widths of each string link can be displayed.

For any tins in the models added to the section view, the section along the primary string through the tin will be calculated and drawn on the section view in the same colour as the tin colour.

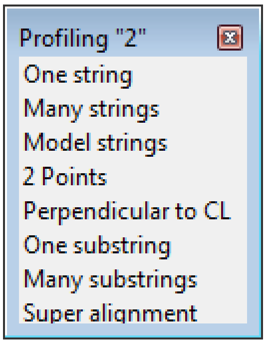


Any strings (service items) in models added to the section view, the parts of the strings in any corridor defined for the primary string will be calculated and drawn on the section view in the same colour as the strings.

For any extrusions, trimeshes and meshes in models added to the section, the section along the primary string through the objects will be calculated and drawn on the section view.

Each time a new primary string is chosen with the profile option, the sections along the new primary string are calculated and displayed for all the tins and strings in the models added to the section view.

The **Profile** walk-right menu is



The **One string** option allows one string to be profiled and then the option terminates (see 9.3.2.1 One String).

The **Many strings** option profiles one string and then allows the user to select another string to be profiled without leaving the profile option. The **Many strings** option is very useful when a number of strings need to be examined one after another in a section view (see 9.3.2.2 Many Strings).

The **Model strings** option profiles the strings from a given model, one at a time. For example, it is used to sequentially examine each cross-section down a road when all the cross-sections are stored in one model (see 9.3.2.3 Model Strings).

The **2 points** option dynamically profiles between two selected points. The profile between the first selected point and the cursor position is dynamically recalculated as the cursor is moved around the view (see 9.3.2.4 2 Points).

The **Perpendicular to CL** option dynamically profiles along a cross-section which is perpendicular to a selected string. The cross section profile is dynamically recalculated as the cursor is moved along the selected string (see 9.3.2.5 Perpendicular to CL).

The **One substring** and **Many substrings** options are used to section along house connections and property controls for sewer and drainage strings (see 9.3.2.1 One String, 9.3.2.2 Many Strings).

For the option *One String*, go to	9.3.2.1 One String
Many strings	9.3.2.2 Many Strings
Model strings	9.3.2.3 Model Strings
2 points	9.3.2.4 2 Points
Perpendicular to CL	9.3.2.5 Perpendicular to CL
One substring	9.3.2.6 One Substring
Many substrings	9.3.2.7 Many substrings
Super alignment	9.3.2.8 Super alignment

9.3.2.1 One String

Position of option on menu: **Section View Menu** **View =>Profile =>One string**

After choosing the **One string** option, the user selects the string to be profiled. The option then terminates.

To cancel the option without selecting any string, click LB to bring up the **Pick Ops** menu and select **Cancel**.

Return to 9.3.2 Profile or 9.3 Section View Menu.

9.3.2.2 Many Strings

Position of option on menu: **Section View Menu** **View =>Profile =>Many strings**

The **many strings** option is designed for profiled a number of strings one after another.

After selecting **Many strings**, a string is selected for profiling. After the string has been profiled, another string is selected for profiling.

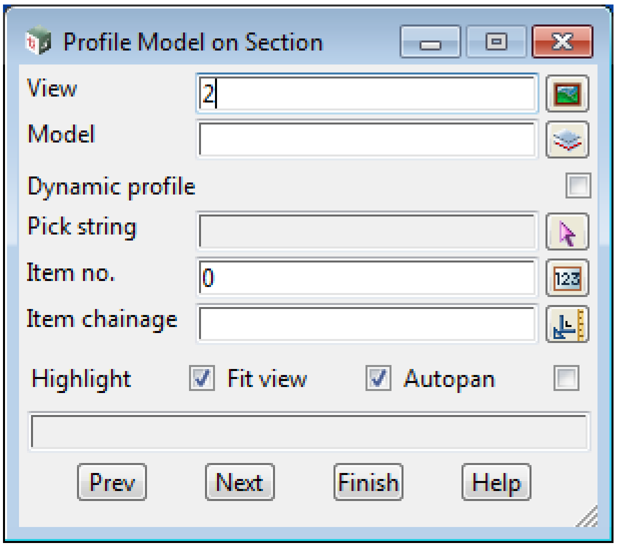
The option will keep profiling selected strings until the option is terminated by clicking LB to get the **Pick Ops** menu and selecting **Cancel**.

Return to 9.3.2 Profile or 9.3 Section View Menu.

9.3.2.3 Model Strings

Position of option on menu: **Section View Menu** **View =>Profile =>Model strings**

After selecting **Model strings**, the **Profile Model on Section** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box	current view	available views

view to set profile parameters for.

Model	model box		available models

name of the model containing the strings to be profiled one at a time

Dynamic profile	tick box		

if ticked, as you move over each string in the given model in a plan or perspective view, the string under the mouse is selected and profiled on the section view.

Item no. input/output

number of the string in the model being profiled.

Item chainage input/output

only applicable if the selected strings are X-section strings

The chainage of the section string in the model being profiled. Also if a chainage is typed in followed by an <Enter> then the model will be searched for a X-section of that chainage and if it exists, it will be profiled.

Highlight tick box tick

*if **ticked**, the string being profiled in the section view is locked and highlighted on any views where it is visible.*

Fit view tick box tick

*if **ticked**, the string being profiled is fitted to the section view.*
*if **not ticked**, the same scale is used for the next section being profiled.*

Autopan tick box

*if **ticked**, then if the profiled string is not visible on any plan views that the model containing the string is on, then the plan view is modified so that the profiled string is in the centre of the plan view.*

Pick button

after selecting the **pick** *button, a string is selected. The string is automatically profiled and its model becomes the model being profiled and* **item no.** *is set to the string's number in the model.*

Prev button

the previous string given by the **item no.** *field is profiled and the* **item no.** *field decremented by one.*

Next button

the next string given by the **item no.** *field is profiled and the* **item no.** *field incremented by one.*

How to Use the Panel

(a) Either a string is picked after selecting the **Pick** button, or the name of the model containing the strings to be profiled is entered into the **model** field.

(b) The **item no.** field is initially set to the picked strings item no. or 0, but can be set to the number of any string in the model.

(c) When the **Prev** or **Next** button is selected, the string given by the number in the **item no.** field is decremented or incremented and the string given by the new item no. is profiled in the section view.

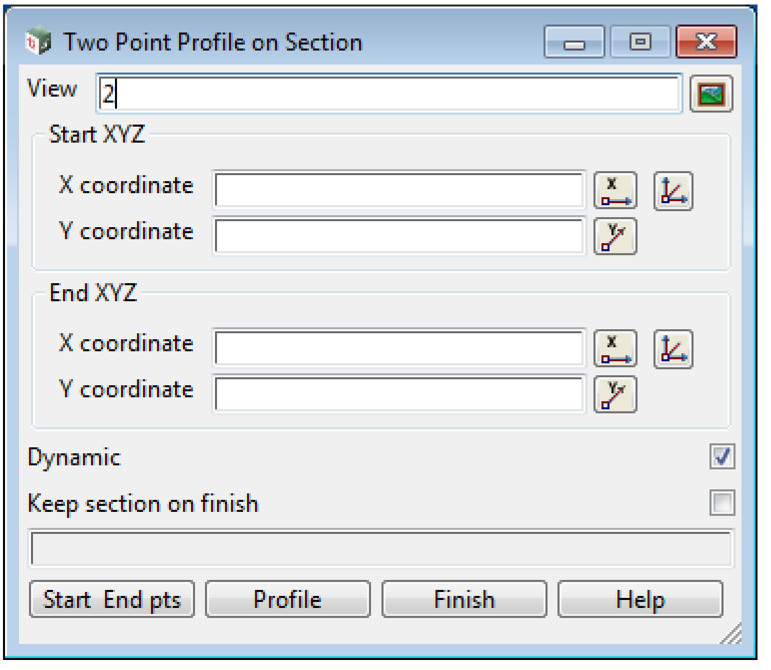
Hence with the **Model strings** option, a number of strings in the one model can be profiled (one after another) by simply repeatedly selecting the **Prev** or **Next** buttons.

Return to 9.3.2 Profile or 9.3 Section View Menu.

9.3.2.4 2 Points

Position of option on menu: **Section View Menu** **View =>Profile =>2 points**

After selecting **2 points**, the **Two Point Profile on Section** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views

view to set profile parameters for.

Start X Y Z	input/output		xyz ops menu

co-ordinates of the start section point. New values can be typed, or a point selected using the **xyz ops** *pop-up menu.*

End X Y Z	input/output		xyz ops menu

co-ordinates of the end section point. New values can be typed, or a point selected using the **xyz ops** *pop-up menu.*

Dynamic	tick box	tick	

*if **ticked**, the profile will be dynamically calculated between the first selected point and the cursor position.*

Keep section on finish	tick box		

*if **ticked**, the two point profile will be remembered by the section view when the* **Finish** *button is selected.*

*if **not ticked**, the definition of the two points to profile is lost when the* **Finish** *button is selected.*

Start End pts	button		

after selecting the **Start End pts** *button, the first point of the section to be profiled is selected and its value written to the* **start xyz** *field. If* **dynamic** *is set on, then a profile from the first point to the cursor position is automatically calculated as the cursor is moved around the view. Selecting the second points finalises the section and the co-ordinates of the final point is written to* **end xyz** *field.*

Profile	button		

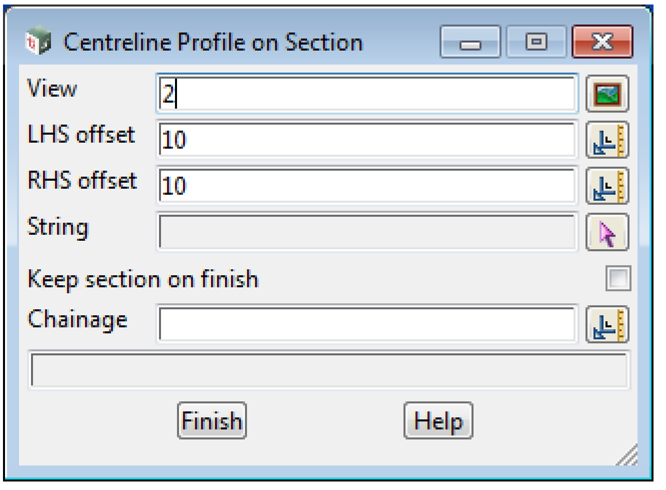
after selecting the **Profile** *button, a section is calculated between the* **start xyz** *and* **end xyz** *points.*

Return to 9.3.2 Profile or 9.3 Section View Menu.

9.3.2.5 Perpendicular to CL

Position of option on menu: **Section View Menu** **View =>Profile =>Perpendicular to CL**

After selecting **Perpendicular to CL**, the **Centreline Profile on Section** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box	current view	available views

view to set profile parameters for.

LHS offset	input	10

distance (in world units) to take the section to the left of the selected string.

RHS offset	input	10

distance (in world units) to take the section to the right of the selected string.

String	string-select

the selected string is used to specify the position where a cross-section is defined. Once the string is selected, then as the cursor is moved around in a view, its position will be dynamically be dropped perpendicularly onto the selected string, and a cross-section defined at right angle to the selected string. A profile along the cross-section will then be drawn in the section view. The cross-section profile will be dynamically modified as the cursor position is modified.

Keep section on finish	tick box

*if **ticked**, the final definition of the profile for the section view will be remembered by the section view when the* **Finish** *button is selected.*

*if **not ticked**, the definition of the profile for the section view is lost when the* **Finish** *button is selected.*

Chainage	output

dynamically displays the centreline chainage of the selected string

Return to 9.3.2 Profile or 9.3 Section View Menu.

9.3.2.6 One Substring

Position of option on menu: **Section View Menu** **View =>Profile =>One substring**

As well as profiling strings, the **One substring** option can profile a house connection or property control (**substrings**) in a drainage or sewer string.

After choosing **One substring** from the menu, the user selects a substring for profiling. After the string or substring has been profiled, the option terminates.

Return to 9.3.2 Profile or 9.3 Section View Menu.

9.3.2.7 Many substrings

Position of option on menu: **Section View Menu** **View =>Profile =>Many substrings**

The **Many substrings** option is designed for profiled a number of substrings, one after another.

After choosing **Many substring**, the user selects a substring for profiling. After the substring has been profiled, another string or substring can be chosen for profiling.

The option is terminated by clicking LB to get the pick ops menu and selecting **cancel**.

Return to 9.3.2 Profile or 9.3 Section View Menu.

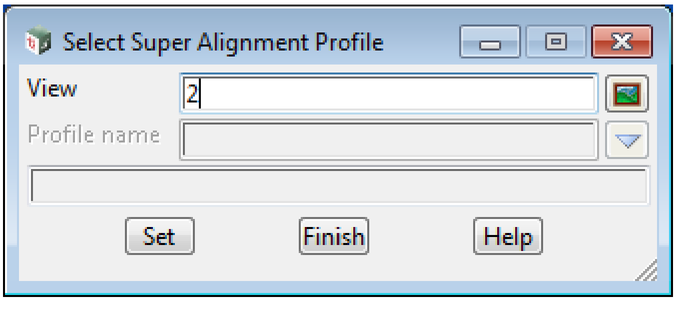
9.3.2.8 Super alignment

Position of option on menu: **Section View Menu** **View =>Profile =>Super alignment**

A super alignment contains one horizontal string but can have more than one vertical dimension For example, vertical geometry, super elevation, widening and sight distance.

When a super alignment is being profiled on the section view, this option allows you to specify which of the vertical dimensions is displayed on the section view.

Selecting **Super alignment** brings up the **Select Super Alignment Profile** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current section view	all views

the view that the super alignment is profiled on.

Field Description	Type	Defaults	Pop-Up
Profile Name	choice box		super alignment dimensions on the profiled super alignment

If profile name is blank, all profiles are shown or the user can select an individual profile to display from the from the choice box.

The possible vertical dominions are Left Super Elevation, Right Super Elevation, Left Widening, Right

Widening, Sight Distance and Stopping Distance.

Set

profile the super alignment with the selected vertical dimension.

Return to 9.3.2 Profile or 9.3 Section View Menu.

9.3.3 Regenerate

Position of option on menu: **Section View Menu** **View =>Regenerate**

The **Regenerate** option is used to re-calculate and re-display the profile of the primary string of the section view.

This is necessary because modifications to the primary string or any strings or tins one the view may make the initial profile invalid. A **Regenerate** is a quick method of bringing the profile up to date.

Continue to 9.3.4 Section Plotting or return to 9.3 Section View Menu.

9.3.4 Section Plotting

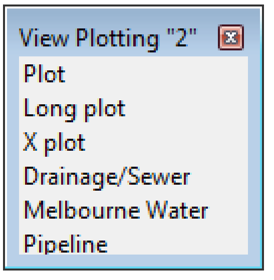
Position of menu: **Section View Menu** **View =>Plotting**

The **Plotting** walk-right menu contains the options **Plot**, **Long plot, X plot, Drainage/Sewer, Melbourne Water** and **Pipeline.**

The **Plot** option is used to make a plot file of all the information displayed in the section view.

The **Long plot** option is used to make a traditional long section plot with string chainages and elevations displayed for each of the strings in the section view. The form of the long plot can be finely controlled by a file of plot parameters.

The **X plot** creates the traditional stacked x-section plots for each x-section string in a given model. The form of the x-plot can be finely controlled by a file of plot parameters.



For the options:

Plot	9.3.4.1 Section Plot
Long plot	25.8.4 Long Plot
X plot	25.8.3 X Plot
Drainage/Sewer	22.9.2 Drainage Longsections
Melbourne water	22.9.3 Melbourne Water
Pipeline	22.5 Plots

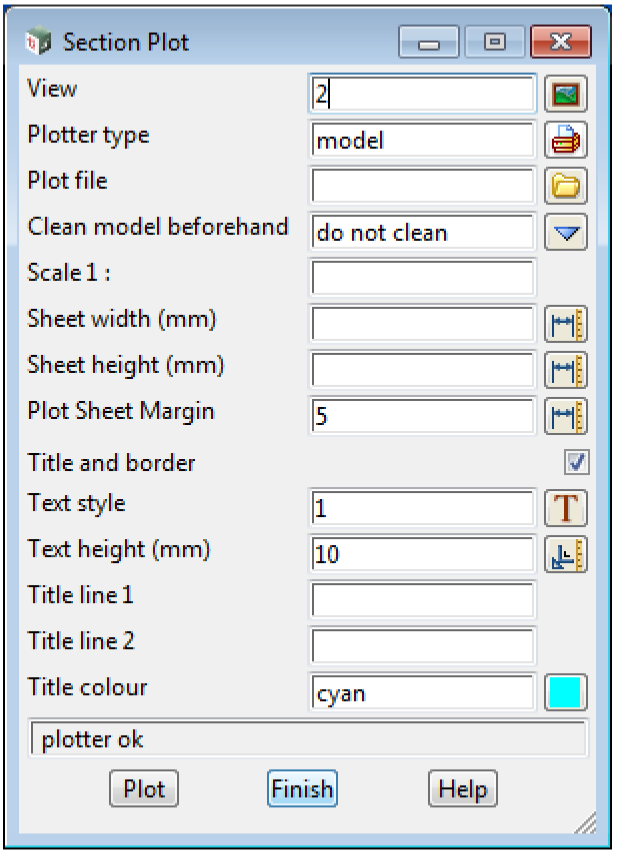
Or continue to 9.3.5 Section Utilities or return to 9.3 Section View Menu.

9.3.4.1 Section Plot

Position of option on menu: **Section View Menu** **View =>Plotting =>Plot**

The **Plot** option writes out all the information displayed in the section view to either a plot file (in a user selected format) or to a 12d Model model. The user selects a scale for the plot and the sheet width and height are calculated by 12d Model.

After selecting **Plot** plot, the **Section Plot** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views

view to set plot parameters for.

Plotter type	input	hp	hp, dxf, postscript *etc.*

file format for the plot information.

Plot file	input	depends on plotter type	

stem of the name of the file to write the plot of the view to. The name ending is added automatically and depends on the plotter type.

Clean model beforehand	choice box		do not clean prompt for clean always clean

only applicable if plotting to a model.
*if **always clean**, the model is cleaned before the plot is created.*
*If **prompt for clean**, the user is prompted that the model will be cleaned before the plot is created.*

*If **do not clean**, the model is not cleaned before the plot is created.*

Scale 1: input

if a value is entered by the user and an <enter> given, the sheet width and height required by the plot are calculated and displayed in the **sheet width** *and* **sheet height** *fields.*

Sheet width (mm) output

if a value is entered by the user and an <enter> given, the scale and height required by the plot are calculated and displayed in the **scale** *and* **sheet height** *fields. The units for sheet width are **millimetres**.*

Sheet height (mm) output

if a value is entered by the user and an <enter> given, the scale and width required by the plot are calculated and displayed in the **scale** *and* **sheet width** *fields. The units for sheet height are **millimetres***

Title and border tick box tick

*if **ticked** then a border and two lines of title are placed on the bottom of the plot*

Text style input 1

text style for the title information

Text Height (mm) input 10

height (in millimetres) to draw the two lines of title information

Title line 1/2 input

first/second line of title information

Title colour input cyan available colours

colour used for the border and the title information

Plot button

write out the plot of the information displayed in the view to the file given in the **plot file** *field. The format of the file is given by the* **plotter type**

Continue 9.3.4.2 Long Plot or return to 9.3.4 Section Plotting or 9.3 Section View Menu.

9.3.4.2 Long Plot

Position of option on menu: **Section View Menu** **View =>Plotting =>Long plot**

The option is for creating long section plots.

The **Long plot** option is documented in the section 25.8.4 Long Plot.

Continue 9.3.4.3 X Plot or return to 9.3.4 Section Plotting or 9.3 Section View Menu.

9.3.4.3 X Plot

Position of option on menu: **Section View Menu** **View =>Plotting =>X plot**

The **x plot** option is for creating x-section plots.

The option is documented in the section 25.8.3 X Plot.

Continue 9.3.4.4 Drainage/Sewer Plot or return to 9.3.4 Section Plotting or 9.3 Section View Menu.

9.3.4.4 Drainage/Sewer Plot

Position of option on menu: **Section View Menu** **View =>Plotting =>Drainage/Sewer**

The ***Drainage/Sewer plot*** option is used to make the long section plots for a drainage or sewer network.

This option is documented in the section 22.9.2 Drainage Longsections.

Continue 9.3.4.5 Melbourne Water Plot or return to 9.3.4 Section Plotting or 9.3 Section View Menu.

9.3.4.5 Melbourne Water Plot

Position of option on menu: **Section View Menu** **View =>Plotting =>Melbourne Water**

The ***Melbourne Water plot*** option is used to make the long section plots for a sewer network to Melbourne Water standards.

This option is documented in the section 22.9.3 Melbourne Water.

Continue 9.3.4.6 Pipeline Plot to return to 9.3.4 Section Plotting or 9.3 Section View Menu.

9.3.4.6 Pipeline Plot

Position of option on menu: **Section View Menu** **View =>Plotting =>Pipeline**

The ***Pipeline plot*** option is used to make the long section plots for a sewer network to Melbourne Water standards.

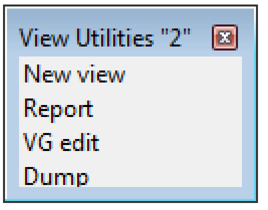
This option is documented in the section 22.5.1 Longsections.

Return to 9.3.4 Section Plotting or 9.3 Section View Menu.

9.3.5 Section Utilities

Position on menu: **Section View Menu** **View =>Utilities**

The **Utilities** menu contains miscellaneous options involving the section view. The **Utilities** walk-right menu is



For the option *New view settings*, go to the section 9.3.5.1 New View Settings
Report 9.3.5.2 Report
VG edit 9.3.5.3 VG Edit
Dump 9.3.5.4 View Dump

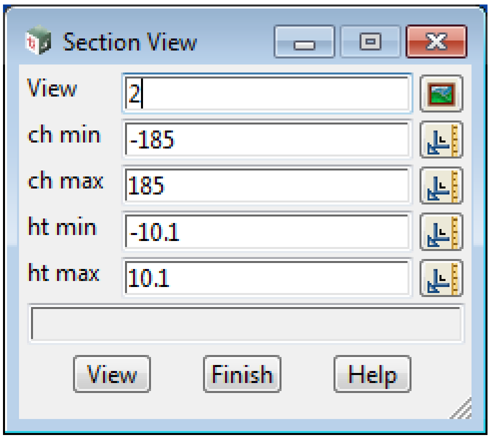
Or return to 9.3 Section View Menu.

9.3.5.1 New View Settings

Position of option on menu: **Section View Menu** **View =>Utilities =>New view settings**

The **New view settings** option displays the current minimum and maximum chainage and height values for the view, and permits the user to specify new values as the section's viewing parameters.

Selecting **New view settings** fires up the **Section View** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults

View input/output current view available views

view to set section parameters for.

chain min/max input/output current chainage min/max

minimum/maximum string chainage for the view. Chainage is used as the x-axis co-ordinates in a section view.

Ht min/max input/output current height min/max

minimum/maximum height (z value) for the view. Height is used as the y-axis co-ordinates in a section view.

View button

define the section view parameters according to the information in the panel. If the box given by the chainage and height values in the panel is not the same shape as the view itself, the chainage and height values will be modified so that the chainage and height values match the view shape. The final chainage and height values are displayed in the panel.

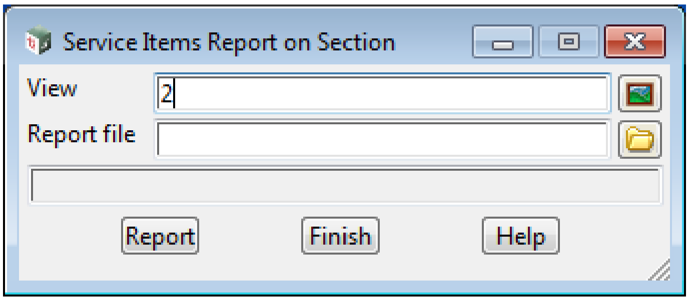
Continue to 9.3.5.2 Report or return to 9.3.5 Section Utilities or 9.3 Section View Menu.

9.3.5.2 Report

Position of option on menu: **Section View Menu** **View =>Utilities =>Report**

This **Report** option is for producing information on all the items displayed in the section view.

After selecting **Report**, the **Service Items Report on Section View** panel is displayed



For the profiled string, this option generates a report which includes the

- section through any tins on the section view
- name and model of any services in the corridor defined by the section view
- co-ordinates and chainages of the parts of the service in the corridor, and the chainage and offset for each of the point of the parts projected onto the profiled string.
- clearance at the point where any service goes under or over the profiled string.

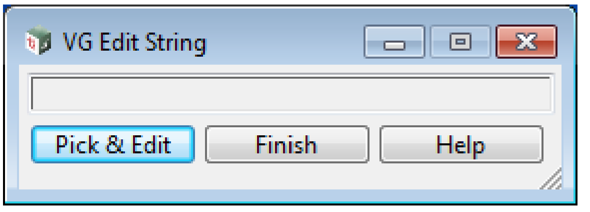
Continue to 9.3.5.3 VG Edit or return to 9.3.5 Section Utilities or 9.3 Section View Menu.

9.3.5.3 VG Edit

Position of option on menu: **Section View Menu** **View =>Utilities =>VG Edit**

The **VG edit** option is used to interactively create and modify the vertical geometry of a 12d Model alignment string, or the vertical information of any string that can be edited in a section view (e.g. drainage string). Vertical geometry can only be created for an existing string.

After selecting the **VG edit** option, the **VG edit string** panel is placed on the screen.



The option is already in **Pick** mode and the user simply picks and accepts the string whose vertical geometry is to be created and/or edited The **Pick Edit** button only needs to be used if the pick was cancelled from the **Pick Ops** menu.

Once the string has been selected, it is automatically made the primary string for the section view and a string profile created along the string.

As discussed in the section view profile options, for each tin in a model added to the section view, the section through the tin along the primary string (the selected string) is calculated and displayed in the section view.

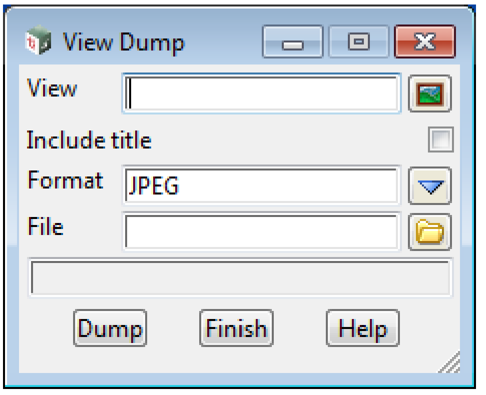
The **VG edit** option then fires up the standard string editor which is described in detail in the **string edit** options.

Continue to 9.3.5.4 View Dump or return to 9.3.5 Section Utilities or 9.3 Section View Menu.

9.3.5.4 View Dump

Position of option on menu: **Section View Menu View =>Utilities =>Dump**

This option is used to write the view out in a user selected format. It is the same as the dump option for the plan view and is fully documented in the plan **View => Utilities** section.



For more information on this option please go to 12.7 Dump.

Return to 9.3.5 Section Utilities or 9.3 Section View Menu.

9.4 Options Common to All Views

Some options are common to all view types and will be documented in the one place rather than for each View type.

See

9.4.1 Model Ops
9.4.2 Redraw
9.4.3 Fit
9.4.4 Previous
9.4.5 Clone
9.4.6 Properties
9.4.7 Delete

9.4.1 Model Ops

Position of menu: **Plan View Menu View =>Models**

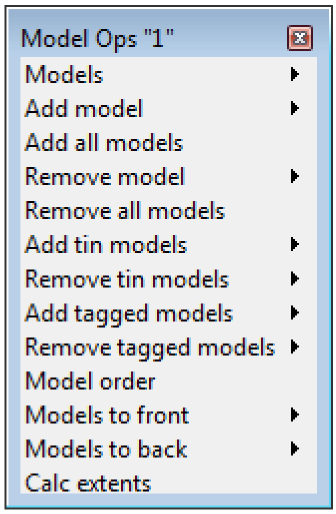
Position of menu: **Section View Menu View =>Models**

Position of menu: **Perspective View Menu View =>Models**

Walking right on **Models** brings up the **Model Ops** walk-right menu which is used to add and remove models from the view.

These options are the same for Plan, Perspective and Section views.

The **Model Ops** walk-right is



For the option:

Models, go to	9.4.1.1 Models
Add model	9.4.1.2 Add Model
Add all models	9.4.1.3 Add All Models
Remove model	9.4.1.4 Remove Model
Remove all models	9.4.1.5 Remove All Models
Add tin models	9.4.1.6 Add Tin Models
Remove tin models	9.4.1.7 Remove Tin Models
Add tagged models	9.4.1.8 Add Tagged Models
Remove tagged models	9.4.1.9 Remove Tagged Models
Model order	9.4.1.10 Model Order
Models to front	9.4.1.11 Models to Front
Models to back	9.4.1.12 Models to Back
Calc extents	9.4.1.13 Calc Extents

Or return to 9 Menus on Views

9.4.1.1 Models

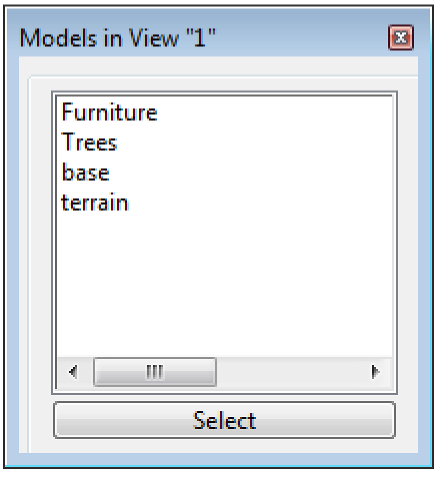
Position of menu: **Plan View Menu** **View => Models =>Models**

Position of menu: **Section View Menu** **View => Models =>Models**

Position of menu: **Perspective View Menu** **View => Models =>Models**

Models is a walk-right option which lists the models currently attached to that view.

If a model is selected from the displayed list, the model will be redrawn on the view. It will also be brought to the top of the display list, *i.e.*, it is in the foreground and all the other models on the view are drawn before it.



Continue to 9.4.1.2 Add Model or return to 9 Menus on Views

9.4.1.2 Add Model

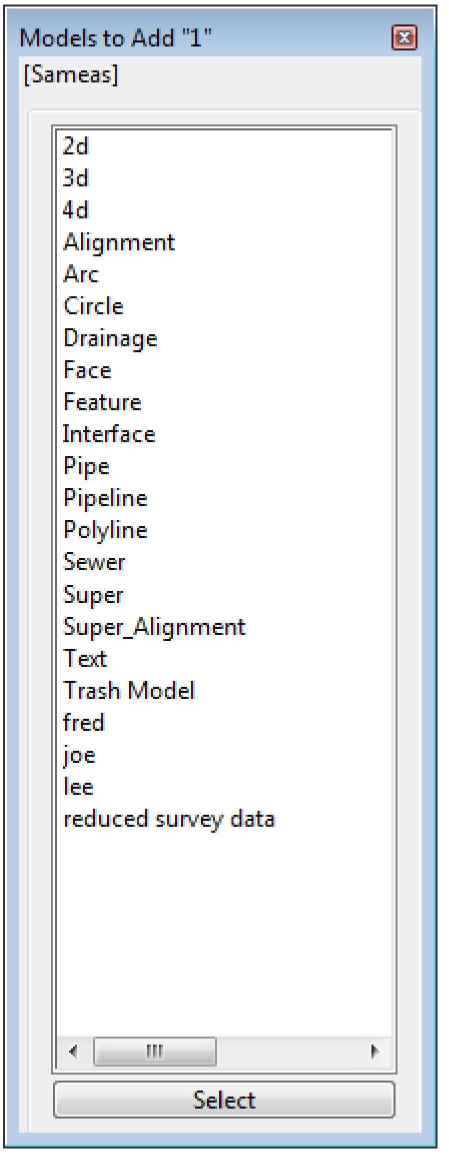
Position of menu: **Plan View Menu** **View => Models =>Add model**

Position of menu: **Section View Menu** **View => Models =>Add model**

Position of menu: **Perspective View Menu** **View => Models =>Add model**

The **Add model** menu item operates two ways.

The **Add model** walk-right brings up the **Models to Add** list which is a list of all the models not currently added to the view.

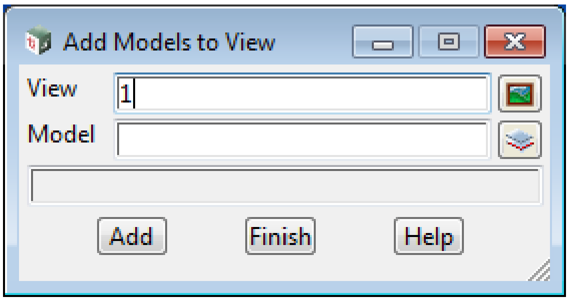


By double clicking on one of the models from the **Models to Add** list, it is added to the view. The list is then removed. To add a number of models from the list to the view, select the models in the list in the standard Microsoft way and then click on **Select**.

If the **Models to Add** list has been moved or pinned, models can be selected from the list and added to the view and the **Models to Add** list will remain with the selected models removed from the list. When all the required models have been added, delete the list using the **[X]** button.

If **Add model** itself is activated (by clicking LB when **Add model** is highlighted), the **Add Model to a**

View panel appears. The **Add Model to a View** panel can be used to create new models as well as add existing models to any view. Wild cards and characters are allowed for adding models.



Any models added to a view are immediately drawn on the view.

Continue to 9.4.1.3 Add All Models or return to 9 Menus on Views

9.4.1.3 Add All Models

Position of option on menu: **Plan View Menu** **View => Models =>Add all models**

Position of option on menu: **Section View Menu** **View => Models =>Add all models**

Position of option on menu: **Perspective View Menu** **View => Models =>Add all models**

The **Add all models** option adds all the models to the view.

Continue to 9.4.1.4 Remove Model or return to 9 Menus on Views

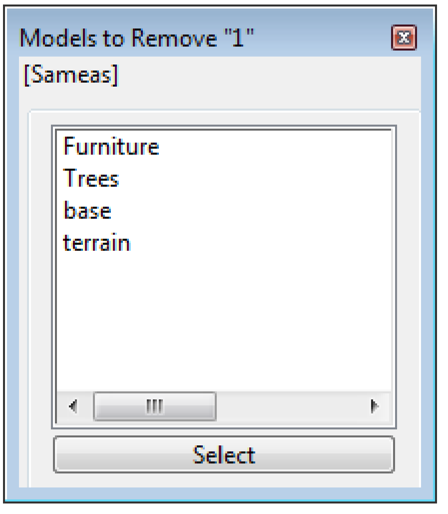
9.4.1.4 Remove Model

Position of menu: **Plan View Menu** **View => Models =>Remove model**

Position of menu: **Section View Menu** **View => Models =>Remove model**

Position of menu: **Perspective View Menu** **View => Models =>Remove model**

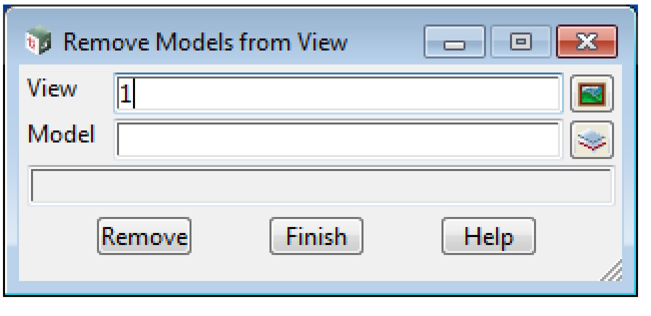
The **Remove model** menu item also operates in two ways. The **Remove model** walk-right brings up the **Models to Remove** list which is a list of all the models currently added to the view.



By double clicking on one of the models from the **Models to Remove** list, it is removed from the view. The list is then removed. To remove a number of models on the list from the view, select the models in the list and then click on **Select**.

If the **Models to Remove** list has been moved or pinned, models can be selected from the list and removed from the view and the **Models to Remove** list will remain with the selected models removed from the list. When all the required models have been removed, delete the list using the **[X]** button.

If **Remove model** itself is activated, the **Remove Model from a View** panel appears. The **Remove Model from a View** panel can be used to remove any models from any view. Wild cards and characters are allowed for removing models.



NOTE
Models still exist after they are removed from a view. Models can only be deleted from 12d Model by using the **Models=>Delete** option.

Continue to 9.4.1.5 Remove All Models or return to 9 Menus on Views

9.4.1.5 Remove All Models

Position of option on menu: **Plan View Menu** **View => Models =>Remove all models**

Position of option on menu: **Section View Menu** **View => Models =>Remove all models**

Position of option on menu: **Perspective View Menu** **View => Models =>Remove all models**

The **Remove all models** option removes all the models from the view and clears the view.

Continue to 9.4.1.6 Add Tin Models or return to 9 Menus on Views

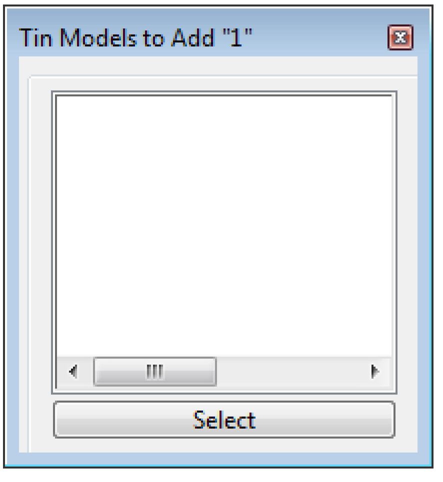
9.4.1.6 Add Tin Models

Position of menu: **Plan View Menu** **View => Models =>Add tin models**

Position of menu: **Section View Menu** **View => Models =>Add tin models**

Position of menu: **Perspective View Menu** **View => Models =>Add tin models**

The **Add tin models** walk-right brings up the **Tin Models to Add** list of all tins in the project whose models have not been added to the view



By double clicking on one of the tins from the **Tin Models to Add** list, all the models in the tin are added to the view. The list is then removed. To add the models from a number of tins from the list to the view, select the tins in the list in the standard Microsoft way and then click on **Select**.

If the **Tin Models to Add** list has been moved or pinned, tins can be selected from the list and their models added to the view and the **Tin Models to Add** list will remain with the selected tins removed from the list. When all the required tins have been added, delete the list using the **[X]** button.

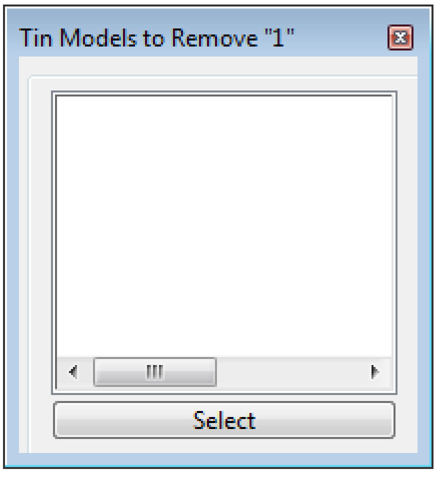
Any models added to a view are immediately drawn on the view.

Continue to 9.4.1.7 Remove Tin Models or return to 9 Menus on Views

9.4.1.7 Remove Tin Models

Position of menu:	**Plan View Menu**	**View => Models =>Remove tin models**
Position of menu:	**Section View Menu**	**View => Models =>Remove tin models**
Position of menu:	**Perspective View Menu**	**View => Models =>Remove tin models**

The **Remove tin models** walk-right brings up the **Tin Models to Remove** list of all tins whose models are currently added to the view.



By double clicking on one of the tins from the **Tin Models to Remove** list, all the models in the tin are removed from the view. The list is then removed. To remove the models from a number of tins on the list from the view, select the tins in the list and then click on **Select**.

If the **Tin Models to Remove** list has been moved or pinned, tins can be selected from the list and removed from the view and the **Tin Models to Remove** list will remain with the selected tins removed from the list. When all the required tins have been removed, delete the list using the **[X]** button.

Continue to 9.4.1.8 Add Tagged Models or return to 9 Menus on Views

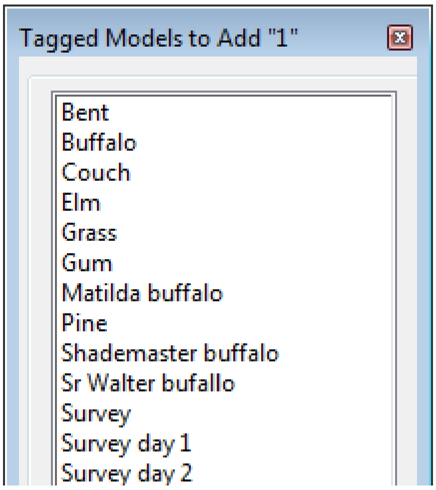
9.4.1.8 Add Tagged Models

Position of menu: **Plan View Menu** **View => Models =>Add tagged models**

Position of menu: **Section View Menu** **View => Models =>Add tagged models**

Position of menu: **Perspective View Menu** **View => Models =>Add tagged models**

The **Add tagged models** walk-right brings up the **Tagged Models to Add** list of all the tags (first level and subtags) defined in the project.



By double clicking on one of the tag names from the **Tagged Models to Add** list, all the models with that tag name, or models whose tag is a subtag of the tag name, are added to the view. The list is then removed. To select a number of tags from the list, select the tags in the standard Microsoft way and then click on **Select**.

If the **Tagged Models to Add** list has been moved or pinned, tags can be selected from the list and their models added to the view and the **Tagged Models to Add** list will remain with the selected tags removed from the list. When all the required tags have been added, delete the list using the **[X]** button.

Any models added to a view are immediately drawn on the view.

Continue to 9.4.1.9 Remove Tagged Models or return to 9 Menus on Views

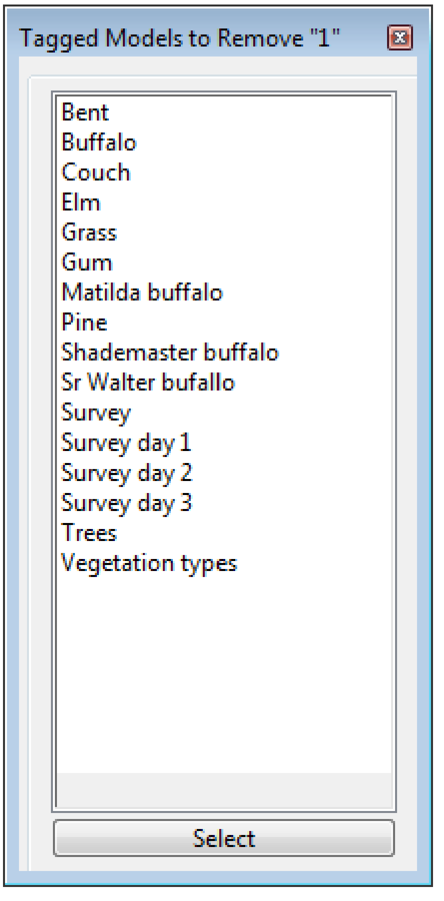
9.4.1.9 Remove Tagged Models

Position of menu: **Plan View Menu** **View => Models =>Remove tagged models**

Position of menu: **Section View Menu** **View => Models =>Remove tagged models**

Position of menu: **Perspective View Menu** **View => Models =>Remove tagged models**

The **Remove tagged models** walk-right brings up the **Tagged Models to Remove** list of all tins whose models are currently added to the view.



By double clicking on one of the tag name from the **Tagged Models to Remove** list, all the models with that tag name, or models whose tag is a subtag of the tag name, are removed from the view. The list is then removed.To select a number of tags from the list, select the tags in the standard Microsoft way and then click on **Select**.

If the **Tagged Models to Remove** list has been moved or pinned, tags can be selected from the list and removed from the view and the **Tagged Models to Remove** list will remain with the selected tags removed from the list. When all the required tags have been removed, delete the list using the **[X]** button.

Continue to 9.4.1.10 Model Order or return to 9 Menus on Views

9.4.1.10 Model Order

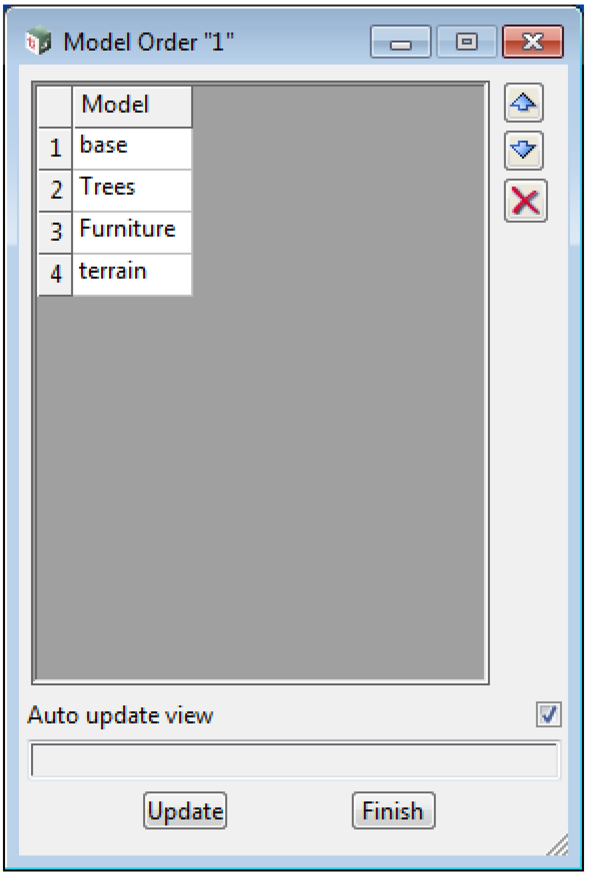
Position of menu: **Plan View Menu** **View => Models =>Model order**

Position of menu: **Section View Menu** **View => Models =>Model order**

Position of menu: **Perspective View Menu** **View => Models =>Model order**

The **Model order** options displays the list of all models on a view in the reverse model drawing order. That is, the first model on the list is the last model drawn and hence the most visible model.

Selecting **Model order** brings up the **Model Order** panel.



The fields and buttons used in this panel have the following functions.

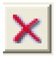
Field Description Type Defaults Pop-Up

Model

list of models on the view in the reverse drawing order. That is, the first model on the list is drawn last and so is the most visible.

Up Arrow, Down Arrow

when a model name is highlighted, clicking the Up/Down arrow will change the order of the model.



Delete

when a model name is highlighted, clicking the Delete icon will remove the model from the view.

when a model name is highlighted, clicking on the Cross will remove the model from the view.

Auto update view tick box tick

if ticked then the view is redrawn each time a model is moved in the list or added/removed from the list.

Update button

redraws the view using the new model order.

Continue to 9.4.1.11 Models to Front or return to 9 Menus on Views

9.4.1.11 Models to Front

Position of menu: **Plan View Menu** **View => Models =>Models to front**

Position of menu: **Section View Menu** **View => Models =>Models to front**

Position of menu: **Perspective View Menu** **View => Models =>Models to front**

Models to front is a walk-right option. When the walk-right is chosen, the list of models currently attached to that view is displayed.

If a model is selected from the displayed list, the model will be the last to be drawn on the view. Hence, it is in the foreground and all the other models on the view are drawn before it.

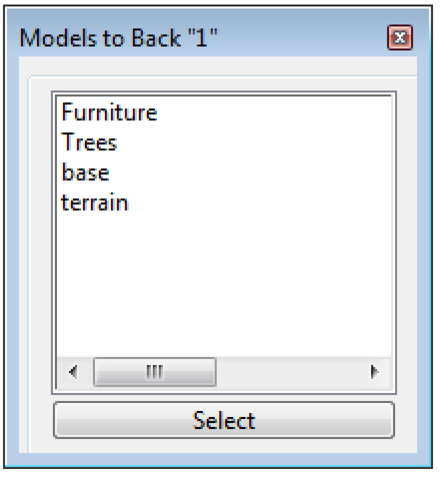
Continue to 9.4.1.12 Models to Back or return to 9 Menus on Views

9.4.1.12 Models to Back

Position of menu: **Plan View Menu** **View => Models =>Models to back**

Position of menu: **Section View Menu** **View => Models =>Models to back**

Position of menu: **Perspective View Menu** **View => Models =>Models to back**



Models to back is a walk-right option. When the walk-right is chosen, the list of models currently attached to that view is displayed.

If a model is selected from the displayed list, the model will be the first to be drawn on the view. Hence, it is in the background and all the other models on the view are drawn after it.

Continue to 9.4.1.13 Calc Extents or return to 9 Menus on Views

9.4.1.13 Calc Extents

Position of option on menu: **Plan View Menu** **View => Models =>Calc extents**

Position of option on menu: **Section View Menu** **View => Models =>Calc extents**

Position of option on menu: **Perspective View Menu** **View => Models =>Calc extents**

For each model on the view, the size of the x, y, z box required to enclose the data in the model is calculated.That is, the option calculates the model bounding box for each model on the view.

Return to 9 Menus on Views

9.4.2 Redraw

Position of option on menu: **Plan View Menu** **View => Models =>Redraw**

Position of option on menu: **Section View Menu** **View => Models =>Redraw**

Position of option on menu: **Perspective View Menu** **View => Models =>Redraw**

As its name implies, this option redraws all the models on the view using the current drawing parameters. This is the same as clicking MB in the view title area.

NOTES

1. When a model is removed from a view, it is “undrawn”, that is, drawn in black. This may also black out important details of other models still attached to the view. If this happens, use the **redraw** option to refresh the view.
2. Clicking MB in the view-title area is the easiest method of redrawing a view.

Return to 9 Menus on Views

9.4.3 Fit

Position of option on menu: **Plan View Menu** **View => Models =>Fit**

Position of option on menu: **Section View Menu** **View => Models =>Fit**

Position of option on menu: **Perspective View Menu** **View => Models =>Fit**

When the **Fit** option is chosen,***12d Model*** calculates viewing parameters which will allow all the models attached to the view to be fully displayed. The view is then redrawn using these new drawing parameters.

Note if the data does not fill the view after a fit, then a **Calc Extents** may be needed to re-calculate the model bounding boxes.

Return to 9 Menus on Views

9.4.4 Previous

Position of option on menu: **Plan View Menu** **View => Models =>Previous**

Position of option on menu: **Section View Menu** **View => Models =>Previous**

Position of option on menu: **Perspective View Menu** **View => Models =>Previous**

Each time a view has its viewing parameters changed, the old set is recorded as the previous parameter set.The previous option sets the viewing parameters back to this previous parameter set.

The view is then redrawn using those settings. The last set then becomes the old set.

Return to 9 Menus on Views

9.4.5 Clone

Position of option on menu: **Plan View Menu** **View => Models =>Clone**

Position of option on menu: **Section View Menu** **View => Models =>Clone**

Position of option on menu: **Perspective View Menu** **View => Models =>Clone**

If LB is clicked on **Clone**, then a new view identical to the original view with all the same models on the new plus and apart form the view name, all the new view setting are identical to the original view.

The view name of the Cloned view is the name of the original view with " Copy" appended to it.

A **Plan/Perspective/Section View Properties** panel for the new view is then brought up so that any of the View properties (for example the view name) can be quickly changed. See 9.4.6 Properties

Return to 9 Menus on Views

9.4.6 Properties

Position of option on menu: **Plan View Menu** **View => Models =>Properties**

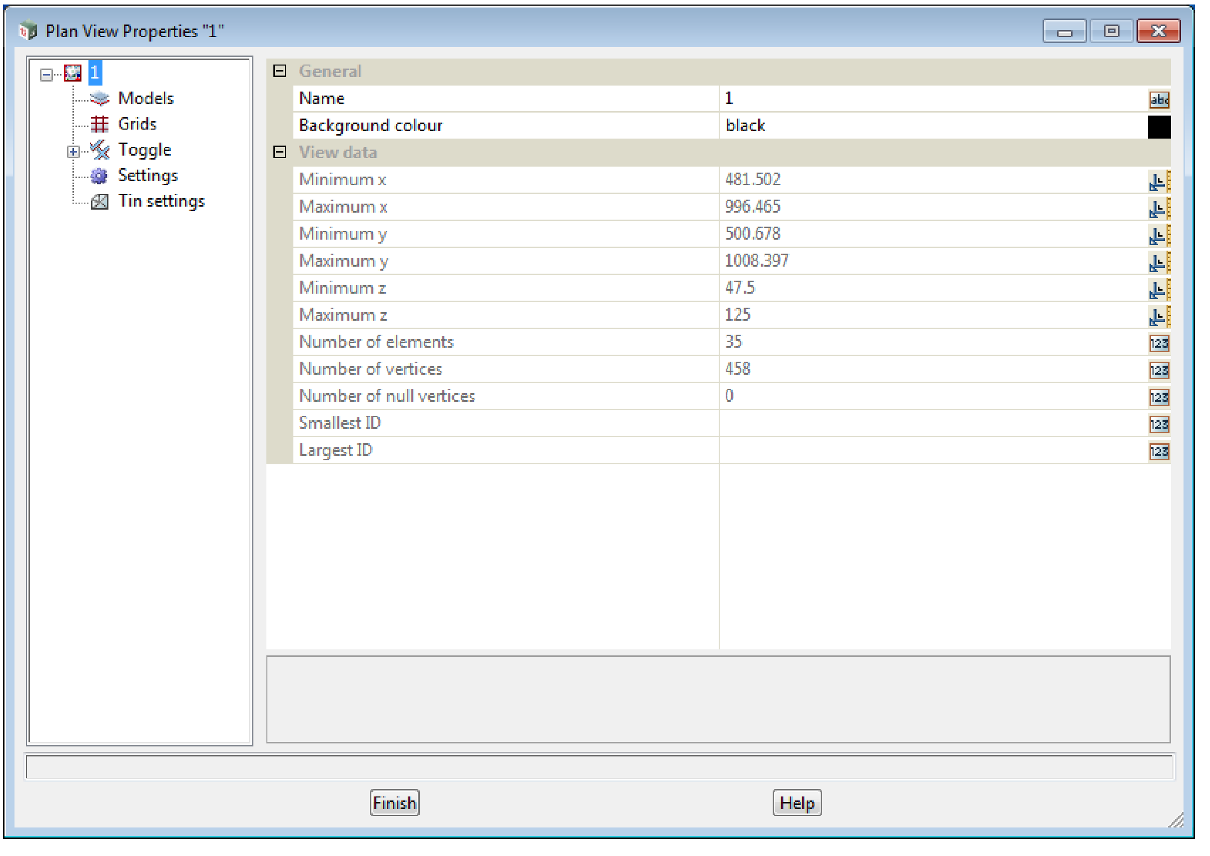
Position of option on menu: **Section View Menu** **View => Models =>Properties**

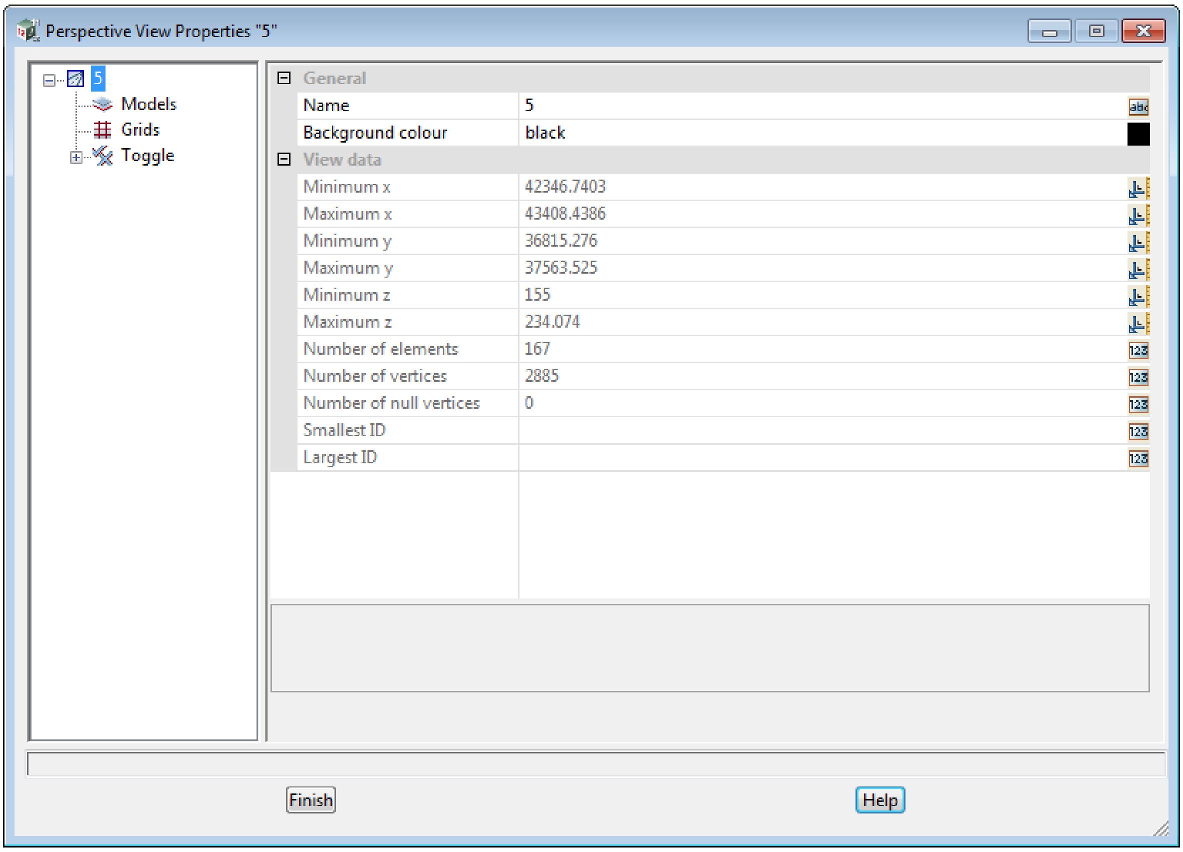
Position of option on menu: **Perspective View Menu** **View => Models =>Properties**

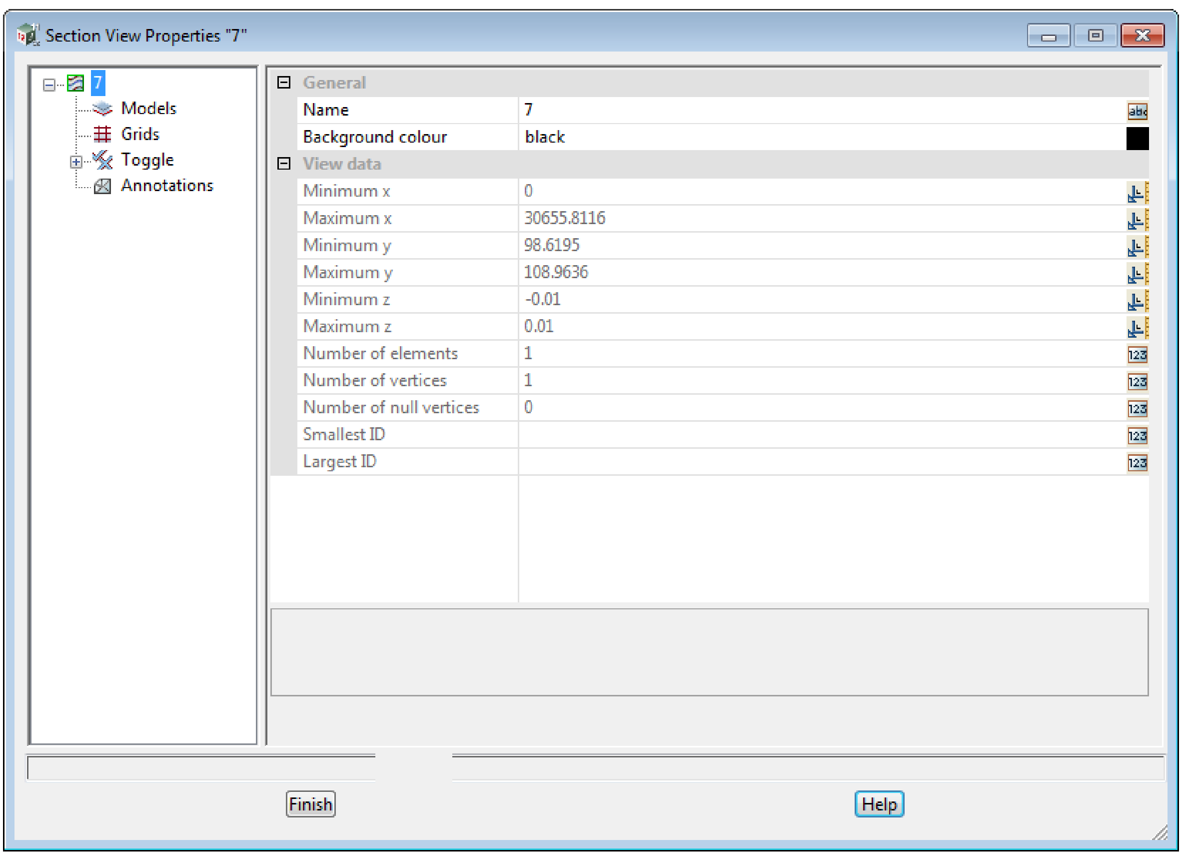
Clicked LB on the on **Properties** button brings up the **Plan View Properties**, **Perspective View Properties**, or **Section View Properties** panel for that view and view type.

The **View Properties** panels display all the settings for the view in a tree structure and clicking on a node brings up various settings on the right hand side of the panel

Many of the settings can then be changed, and the changes take place immediately without having to press another button.







Return to 9 Menus on Views

9.4.7 Delete

Position of option on menu: **Plan View Menu** **View =>Delete**

Position of option on menu: **Section View Menu** **View =>Delete**

Position of option on menu: **Perspective View Menu** **View =>Delete**

This option is used to delete the view and remove it from the screen.

After selecting the **Delete** option, a **Delete View** yes-no panel appears.

If **Yes** is selected, the view is deleted and removed from the screen.

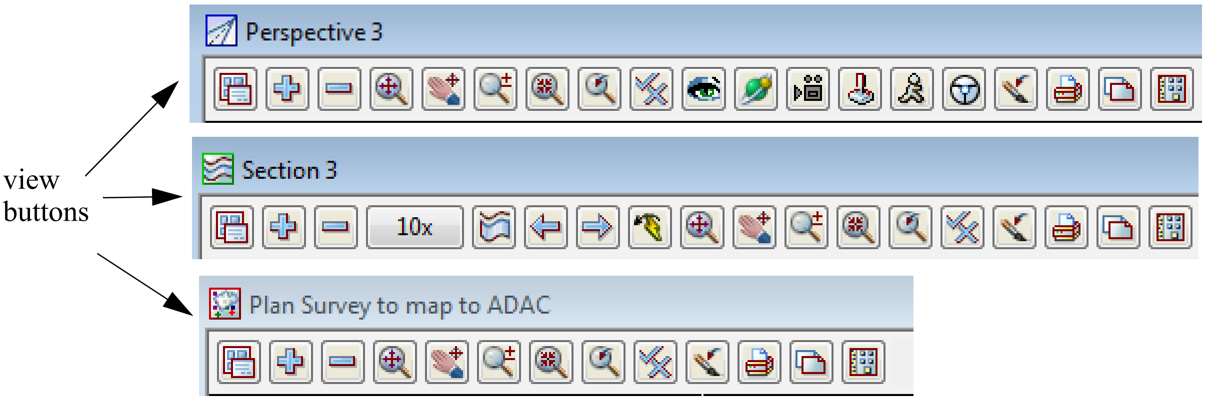
Selecting **No** removes the **Yes-No** pop-up and leaves the view as it is.

Return to 9 Menus on Views

9.5 View Buttons

For each view type, there are a number of selected **View** menu options available as buttons (view buttons) in the *view title area*.

The options on each view type are



To select one of the view buttons, click LB whilst the cursor is over the button.

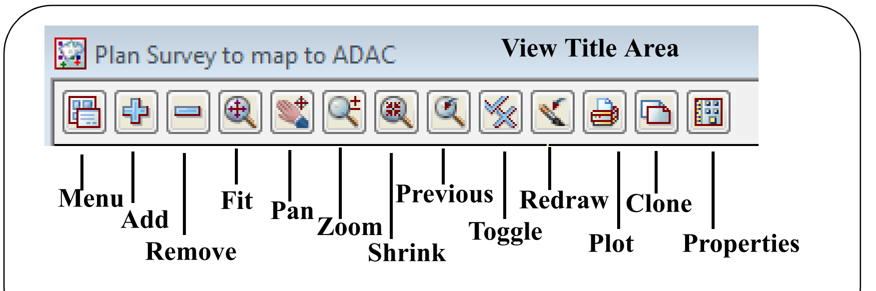
Note that the view buttons do not highlight when the cursor passes over them.

If RB is clicked over any view button other than **Profile** on the section view, the **View** menu for the view type will be displayed.

For documentation on the view buttons see:

9.5.1 Plan View Buttons.
9.5.2 Perspective View Buttons.
9.5.3 Section View Buttons.

9.5.1 Plan View Buttons



Plan View Buttons

Menu

If the **Menu** button is selected in the *view button area* of any view, or if **RB** is clicked in the *view-title area* or in the *view-button area*, a new menu called the **View** menu appears.

Because of the differences between plan, perspective and section views, the options on the **View** menu vary for each view type.

The plan **View** menus is

The **View** menu is removed by selecting the **[X]** button or if the **View** menu hasn't been moved, by simply clicking **RB** again in the *view-title* or *view button* area. If the **View** menu has been moved, clicking **RB** will warp the cursor to the moved View menu.

+ or Add

The **+** view button works in four modes.

- clicking LB whilst the cursor is over the **+** brings up the **Models to Add** list. This list contains all of the models not yet added to the view.
- typing a keyboard character whilst the cursor is over the **+** brings up a **Models to Add** list containing only those models not on the view **with names beginning** with the typed character.
- typing a * character whilst the cursor is over the **+** brings up a **Add Models to View** panel which can then be used with wild cards and characters.
- clicking MB whilst the cursor is over the **+** brings up the **Add Models to View by Pick** panel and starts up a **same as** for a view. Models to add to the view are then selected by simply picking a string from the model. See 4 Tools and Concepts for further documentation on **same as**.
- clicking RB whilst the cursor is over the **+** brings up the **View** panel documented earlier in this

chapter.

For the *Same as* option for the + and - view buttons, go to 9.5.4 Same As for Views.

-or Remove

The **-** view button works in four modes.

- clicking LB whilst the cursor is over the **-** brings up the **Models to Remove** list. This menu lists all of the models on the view.
- typing a keyboard character whilst the cursor is over the **-** brings up a **Models to Remove** list containing only those models on the view with names beginning with the typed character.
- typing a * character whilst the cursor is over the **-** brings up a **Remove Models from View** panel which can then be used with wild cards and characters.
- clicking MB whilst the cursor is over the **-** brings up the **Remove Models from View by Pick** panel and starts up a **Same as** for a view. Models to remove from the view are then selected by simply picking a string from the model. See the next chapter for further documentation on **Same as**.
- clicking RB whilst the cursor is over the **-** brings up the **View** panel documented earlier in this chapter.

For the *Same as* option for the + and - view buttons, go to 9.5.4 Same As for Views.

Fit

This is the same as **Fit** from the **view** menu.
If LB is clicked on the **Fit**, then the **Fit** option is activated.
If RB is clicked on the **Fit**, then the **View** option is activated.

Pan

This is the same as **Pan** from the **View=>Pan** menu.
If LB is clicked on the **Pan**, then the **Pan** option is activated.

Zoom

This is the same as the two point **Zoom** option from the plan **View=>Zoom** menu.
If LB is clicked on the **Zoom**, then the **zoom** option is activated.

Shrink

This is the same as the two point **Shrink** option from the plan **View=>Zoom** menu.
If LB is clicked on the **Shrink**, then the **Shrink** option is activated.

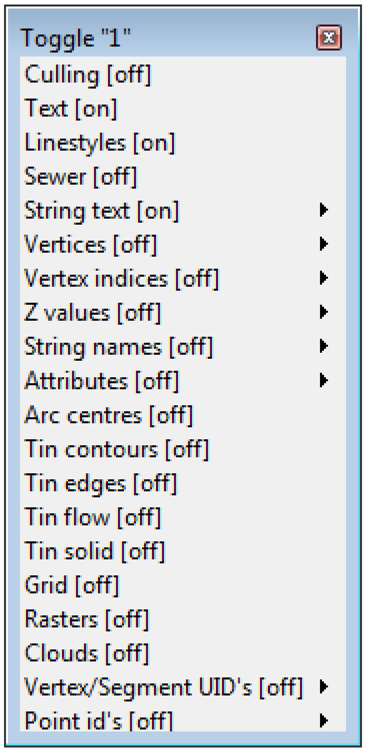
Previous

This is the same as **Previous** from the **view** menu.
If LB is clicked on the **Prev**, then the **Previous** option is activated.
If RB is clicked on the **Prev**, then the **View** option is activated.

Toggle

This is the same as **Toggle** from the **View=>Settings** menu.
If LB is clicked on the **Toggle**, then the **Toggle** menu is activated.
If RB is clicked on the **Toggle**, then the **View** option is activated.

The **Toggle** menu for the plan view types is:

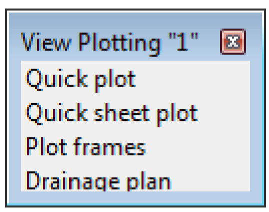


Refresh

As its name implies, this option redraws all the models on the view using the current drawing parameters. This is the same as clicking MB in the view title area.

Plot

If LB is clicked on **Plot**, then the **View Plotting** menu for the plan view is activated.



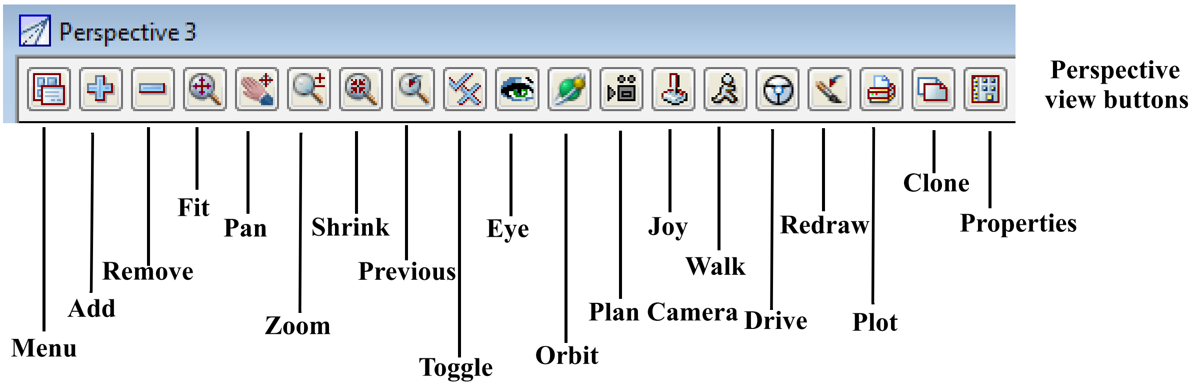
Clone

See 9.4.5 Clone.

Properties

See 9.4.6 Properties.

9.5.2 Perspective View Buttons



See

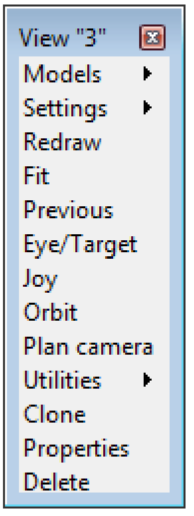
[Menu](#)
[+ or Add](#)
[- or Remove](#)
[Fit](#)
[Previous](#)
[Toggle](#)
[Eye](#)
[Orbit](#)
[Plan Camera](#)
[Joy](#)
[Walk](#)
[Drive](#)
[Redraw](#)
[Plot](#)
[9.4.5 Clone](#)
[9.4.6 Properties](#)

Menu

If the **Menu** button is selected in the *view button area* of any view, or if **RB** is clicked in the *view-title area* or in the *view-button area*, a new menu called the **View** menu appears.

Because of the differences between plan, perspective and section views, the options on the **View** menu vary for each view type.

The **Perspective View** menus is



The **View** menu is removed by selecting the **[X]** button or if the **View** menu hasn't been moved, by simply clicking **RB** again in the *view-title* or *view button* area. If the **View** menu has been moved, clicking **RB** will warp the cursor to the moved View menu.

+ or Add

The **+** view button works in four modes.

- clicking LB whilst the cursor is over the **+** brings up the **Models to Add** list. This list contains all of the models not yet added to the view.
- typing a keyboard character whilst the cursor is over the **+** brings up a **Models to Add** list containing only those models not on the view **with names beginning** with the typed character.
- typing a * character whilst the cursor is over the **+** brings up a **Add Models to View** panel which can then be used with wild cards and characters.
- clicking MB whilst the cursor is over the **+** brings up the **Add Models to View by Pick** panel and starts up a **Same as** for a view. Models to add to the view are then selected by simply picking a string from the model. See 4 Tools and Conceptsfor further documentation on **Same as**.
- clicking RB whilst the cursor is over the **+** brings up the **View** panel documented earlier in this chapter.

For the *Same as* option for the + and - view buttons, go to 9.5.4 Same As for Views.

- or Remove

The **-** view button works in four modes.

- clicking LB whilst the cursor is over the **-** brings up the **Models to Remove** list. This menu lists all of the models on the view.
- typing a keyboard character whilst the cursor is over the **-** brings up a **Models to Remove** list containing only those models on the view with names beginning with the typed character.
- typing a * character whilst the cursor is over the **-** brings up a **Remove Models from View** panel which can then be used with wild cards and characters.
- clicking MB whilst the cursor is over the **-** brings up the **Remove Models from View by Pick** panel and starts up a **Same as** for a view. Models to remove from the view are then selected by simply picking a string from the model. See the next chapter for further documentation on **Same as**.
- clicking RB whilst the cursor is over the **-** brings up the **View** panel documented earlier in this chapter.

For the *Same as* option for the + and - view buttons, go to 9.5.4 Same As for Views.

Fit

This is the same as **Fit** from the **view** menu.
If LB is clicked on the **Fit**, then the **Fit** option is activated.
If RB is clicked on the **Fit**, then the **View** option is activated.

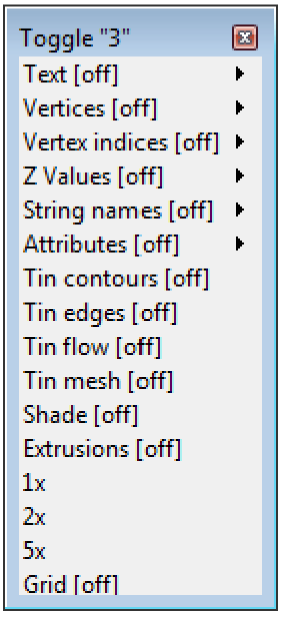
Previous

This is the same as **Previous** from the **View** menu.
If LB is clicked on the **Prev**, then the **Previous** option is activated.
If RB is clicked on the **Prev**, then the **View** option is activated.

Toggle

This is the same as **Toggle** from the **View=>Settings** menu.
If LB is clicked on the **Toggle**, then the **Toggle** menu is activated.
If RB is clicked on the **Toggle**, then the **View** option is activated.

The **Toggle** menu for the perspective view is:



Eye

This is the same as **eye/target** from the **View** menu.
If LB is clicked on the **Eye**, then the **Eye-target** option is activated.
For more information please see 9.2.2 Eye/Target

Orbit

For more information please see 9.2.4 Orbit

Plan Camera

For more information please see 9.2.5 Plan Camera

Joy

This is the same as **Joy** from the **View** menu.
If LB is clicked on the **Joy**, then the **Joy** option is activated.
For more information please see 9.2.3 Joy for View

Walk

This is the same as **String walk =>Along string** from the **View =>Utilities** menu.
If LB is clicked on the **Walk**, then the **Walk along string** option is activated.

Drive

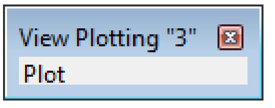
This is the same as **String drive => Along string** from the **View =>Utilities** menu.
If LB is clicked on the **Drive**, then the **Drive along string** option is activated.

Redraw

As its name implies, this option redraws all the models on the view using the current drawing parameters. This is the same as clicking MB in the view title area.

Plot

If LB is clicked on **Plot**, then the **View Plotting** menu for the perspective view is activated.



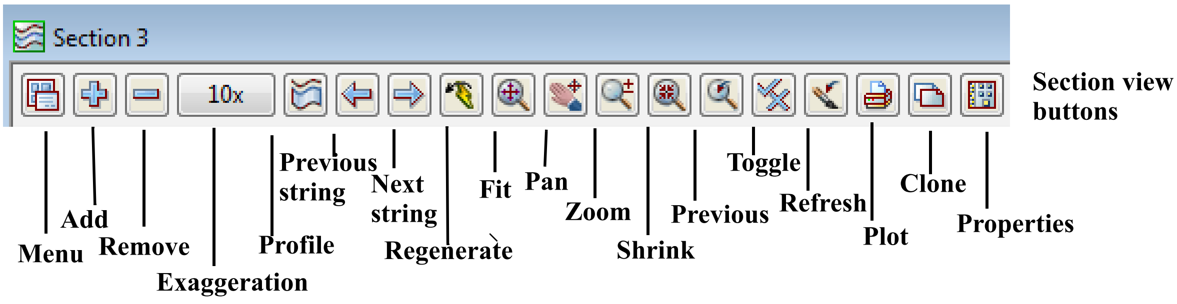
Clone

See 9.4.5 Clone.

Properties

See 9.4.6 Properties.

9.5.3 Section View Buttons

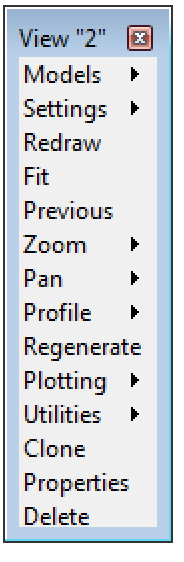


Menu

If the **Menu** button is selected in the *view button area* of any view, or if **RB** is clicked in the *view-title area* or in the *view-button area*, a new menu called the **View** menu appears.

Because of the differences between plan, perspective and section views, the options on the **View** menu vary for each view type.

The **Section View** menus is



The **View** menu is removed by selecting the **[X]** button or if the **View** menu hasn't been moved, by simply clicking **RB** again in the *view-title* or *view button* area. If the **View** menu has been moved, clicking **RB** will warp the cursor to the moved View menu.

+ or Add

The **+** view button works in four modes.

- clicking LB whilst the cursor is over the **+** brings up the **Models to Add** list. This list contains all of the models not yet added to the view.
- typing a keyboard character whilst the cursor is over the **+** brings up a **Models to Add** list containing only those models not on the view **with names beginning** with the typed character.
- typing a * character whilst the cursor is over the **+** brings up a **Add Models to View** panel which can then be used with wild cards and characters.
- clicking MB whilst the cursor is over the **+** brings up the **Add Models to View by Pick** panel and starts up a **Same As** for a view. Models to add to the view are then selected by simply picking a string from the model. See 4 Tools and Concepts for further documentation on **Same As**.
- clicking RB whilst the cursor is over the **+** brings up the **View** panel documented earlier in this

chapter.

For the *Same As* option for the + and - view buttons, go to 9.5.4 Same As for Views.

- or Remove

The **-** view button works in four modes.

- clicking LB whilst the cursor is over the **-** brings up the **Models to Remove** list. This menu lists all of the models on the view.
- typing a keyboard character whilst the cursor is over the **-** brings up a **Models to Remove** list containing only those models on the view with names beginning with the typed character.
- typing a * character whilst the cursor is over the **-** brings up a **Remove Models from View** panel which can then be used with wild cards and characters.
- clicking MB whilst the cursor is over the **-** brings up the **Remove Models from View by Pick** panel and starts up a **Same As** for a view. Models to remove from the view are then selected by simply picking a string from the model. See the next chapter for further documentation on **Same As**.
- clicking RB whilst the cursor is over the **-** brings up the **View** panel documented earlier in this chapter.

For the *Same as* option for the + and - view buttons, go to 9.5.4 Same As for Views.

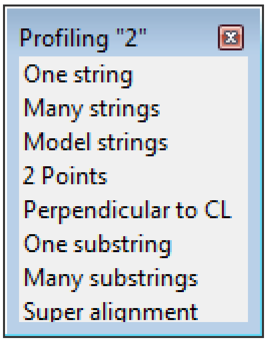
[Exagg]

This is the vertical exaggeration for the view.
If LB is clicked on the exaggeration, then the **Toggle** option for the section view is activated. This has options to quickly change the vertical exaggeration.

Profile

If LB is clicked on **Profile**, then the **Profile** option is activated. This option is the same as **View=>Profiling=>One string**. If a string is being profiled, then the string name is displayed after the Section view name.

If **RB** is clicked on **Profile**, then the **Profiling** menu is activated.



<<

Profile previous string.

>>

Profile next string.

Regen

This is the same as the option **Regenerate** option from the section **view** menu.
If LB is clicked on the **Regen**, then the **regen** option is activated.

Fit

This is the same as **Fit** from the **view** menu.

If LB is clicked on the **Fit**, then the **Fit** option is activated.
If RB is clicked on the **Fit**, then the **View** option is activated.

Pan

This is the same as **Pan** from the **View=>Pan** menu.
If LB is clicked on the **Pan**, then the **Pan** option is activated.

Zoom

This is the same as the two point **Zoom** option from the plan **View=>Zoom** menu.
If LB is clicked on the **Zoom**, then the **Zoom** option is activated.

Shrink

This is the same as the two point **Shrink** option from the plan **View=>Zoom** menu.
If LB is clicked on the **Shrink**, then the **Shrink** option is activated.

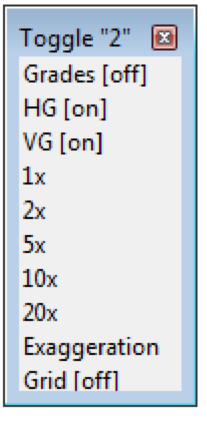
Previous

This is the same as **Previous** from the **View** menu.
If LB is clicked on the **Prev**, then the **Previous** option is activated.
If RB is clicked on the **Prev**, then the **View** option is activated.

Toggle

This is the same as **Toggle** from the **View=>Settings** menu.
If LB is clicked on the **Toggle**, then the **Toggle** menu is activated.
If RB is clicked on the **Toggle**, then the **View** option is activated.

The **Toggle** menu for the section view is:

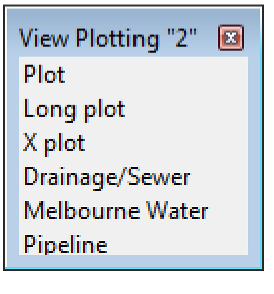


Refresh

As its name implies, this option redraws all the models on the view using the current drawing parameters. This is the same as clicking MB in the view title area.

Plot

If LB is clicked on **Plot**, then the **View Plotting** menu for the section view is activated.



Clone

See 9.4.5 Clone.

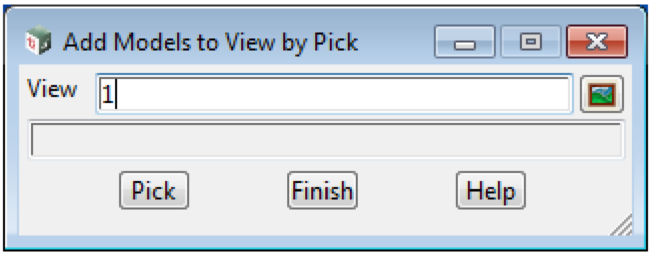
Properties

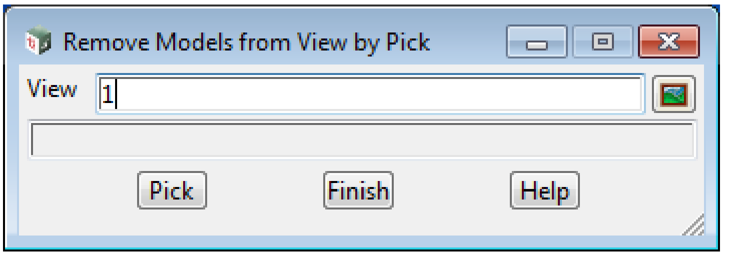
See 9.4.6 Properties.

9.5.4 Same As for Views

Clicking the middle mouse button (MB) over the **+** or **-** view buttons will activate a **Same As** pick which allows the user to select the model to add to (+) or remove from (-) the view by simply picking any string from the model to be added or removed.

After clicking MB over the **+** or **-** view button, the **Add Models to View by Pick** or **Remove Models from View by Pick** panel is brought up.





The option is already running and when a string is picked,. the model containing the string is added to/removed from the view. Further strings can then be selected to add or remove another model.

The option is terminated by selecting **[X]** or **Finish** from the panels, or **Cancel** from the **Pick Ops** menu.

9.6 Displaying Values Text, Vertices and Text on a Plan View

Plan_View_Vertex_Segment_UIDs_Settings_MenuOn ***Plan View =>Settings*** there are a number of options that work in a similar way and control the drawing of string text, drawing crosses at vertices and the labelling of a number of string properties such as vertices, z-values, string name etc.

The first two of these ***Plan View =>Settings*** options are:

1. **Text** for controlling the drawing of *Text strings* and any other text drawn on the view.
2. **Vertices** for drawing crosses at all the vertices in a string. For a text string, a cross is placed at the text justification position.

The other options draw string values as text on the view and are:

3. **Vertex/Segment UID's** for displaying the **Vertex UID's** any **Segment UIDs** for the vertices and segments of super strings.
4. **Point/Vertex IDs** for displaying the **Point/Vertex ID** (a point ID recorded for a vertex) for the vertices of super strings.

   Note - Point/Vertex IDs are not to be confused with *Vertex Indices* which are simply the position of the vertex in the string.
5. **Vertex Indices** for displaying the **Vertex Indices** (position of the vertex in the string) for all the vertices of strings.

   Note - Vertex Indices are not to be confused with *Point/Vertex IDs* which may not even exist for a vertex in the string.
6. **Z values** for displaying the z coordinate to a user specified number of decimal places, for all the vertices of strings.
7. **Names** for displaying the string name at each vertex of a string.
8. **Attributes** for displaying the vertex attributes for all the vertices of super strings.

Vertex/Segment UIDs, ***Point/Vertex Id's***, ***Vertex indices***, ***Z values***, ***Names*** and ***Attributes*** that draw specific values from a string as text on the view.

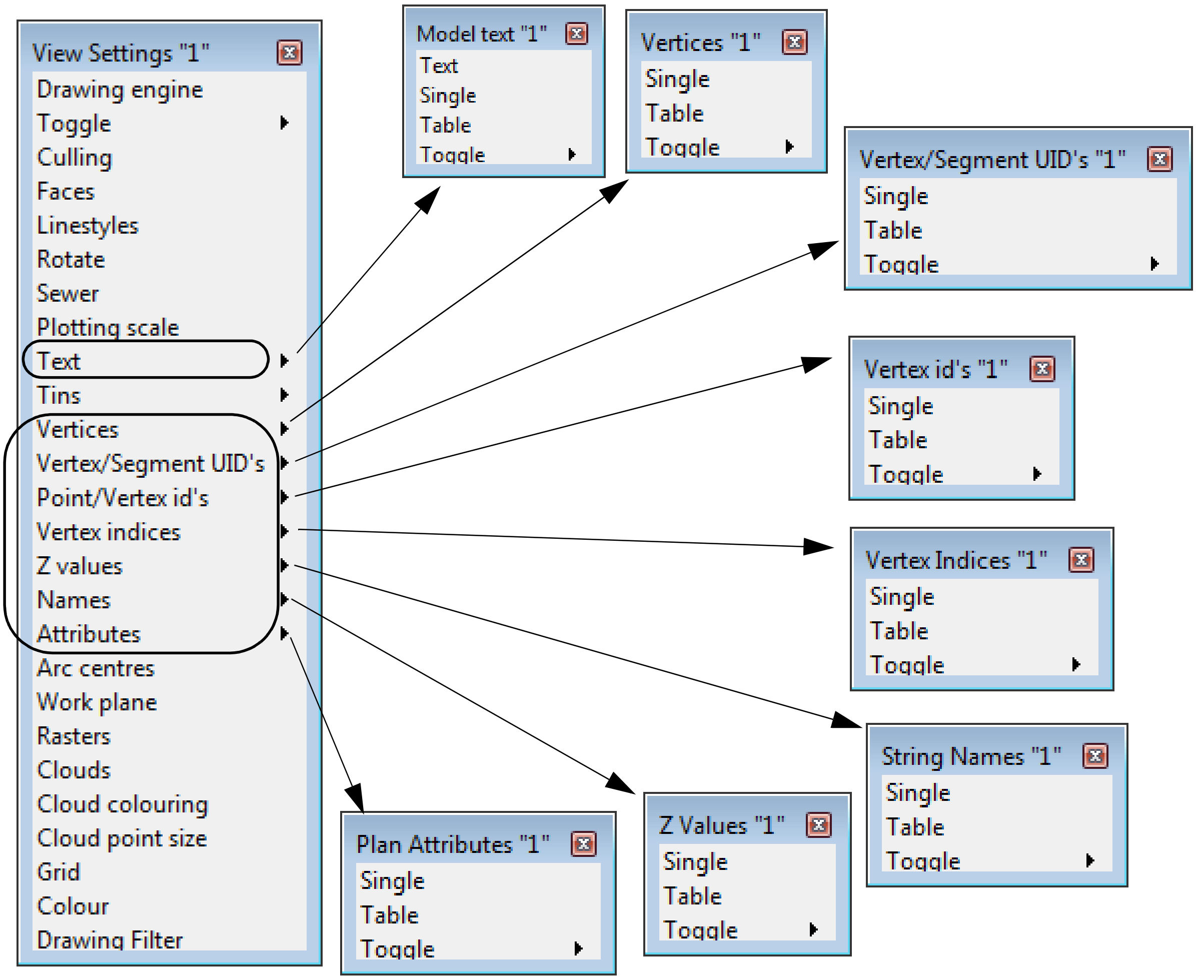
For the options 3 to 8 we refer to the **text** of the **value** that will be drawn as the **Values Text**.

For each of the options there is a setting (the default setting) to say whether the text/crosses are displayed/not displayed and also a table that lists any models that user wants to individually control the draw/don't draw. So each option has a walk-right menu with the items **Single**, **Table** and **Toggle** where

(a) **Single** specifies the ***defaults*** for drawing the ***Value Text/Crosses at Vertices/Text*** for any model on the view.

(b) **Table** contains a list of any models that have their own settings to use instead of the ***defaults***, plus the settings for that model. Each model in the list also has its own ***Draw*** field that overrides the ***Draw*** tick box in **Single**.

If a model is not listed in **Table** then it uses the ***defaults*** set by **Single**.



The **Single** options also have setting so that *Values Text/crosses at vertices/Text* are only drawn if the equivalent world size of the cross/text is not too large. This stops the drawing of *the text/ cross* when the displaying world size is large in proportion to other world units (e.g. when zooming out).

For a description of the options

Single, go to 9.6.1 Single for Values Text, Crosses and Text on Plan Views

Table 9.6.2 Table for Values Text, Crosses and Text on Plan Views

Toggle 9.6.3 Toggle Walk-Right for Value Text, Crosses and Text on Plan Views

Note

Using the environment variable **Env.4d** >***General >Plan table settings*** (see General and PLAN_TABLE_SETTINGS_4D) it is possible to turn off the control by individual models of the ***Values Text/Crosses at Vertices/Text*** drawing. In that case, apart from the Text, there is no walk-right menus. See 9.6.5 Env Var PLAN_TABLE_SETTINGS_4D Set to 0:

9.6.1 Single for Values Text, Crosses and Text on Plan Views

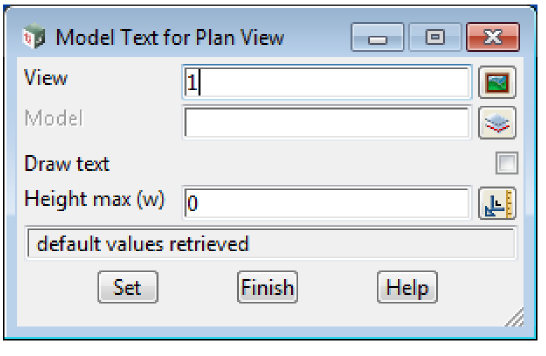
Vertex_Segment_UID_s_for_Plan_View***Plan View =>Settings*** has the option **Text** to draw ***String Text,*** the option **Vertices** to draw ***Crosses at*** all the **Vertices** in a string (often just called ***Crosses***) and the options ***Vertex/Segment UIDs***, ***Point/Vertex Id's***, ***Vertex indices***, ***Z values***, ***Names*** and ***Attributes*** for labelling specific values from a string as text (the **Values Text**). For information on these options see 9.6 Displaying Values Text, Vertices and Text on a Plan View.

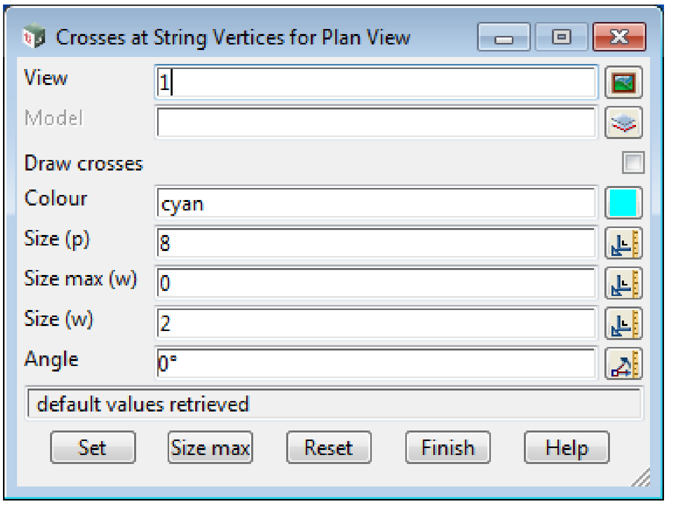
All these options work similarly and will be documented together.

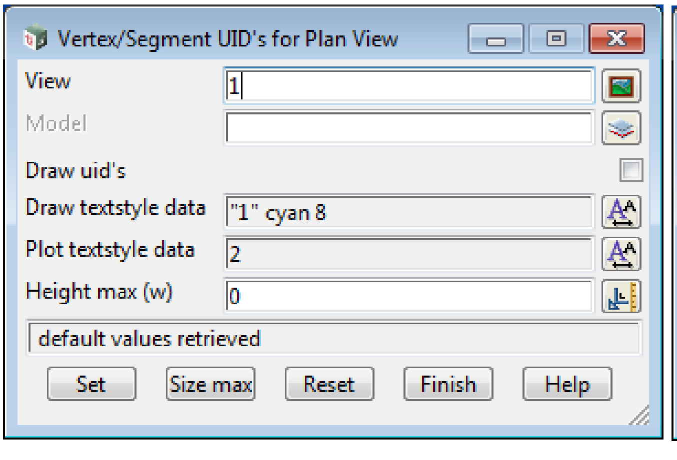
The **Single** option is used to set ***default*** parameters for drawing the ***Value Text***/***Crosses at Vertices/Text*** for strings in all models **not mentioned** in the *Table*, or for an individual model on the plan view.

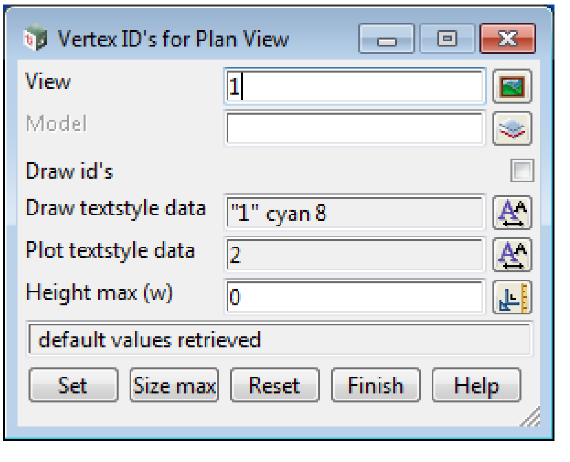
The **Single** options also have setting so that *Values Text/Crosses/Text* are only drawn if the equivalent world size of the cross/text is not too large. This stops the drawing of *the text/cross* when the displaying world size is large in proportion to other world units (for example when zooming out).

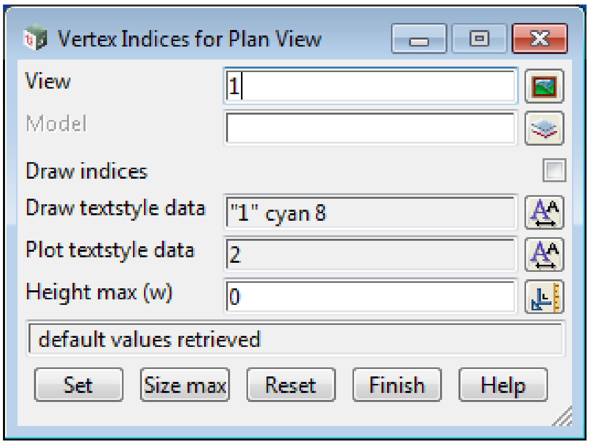
Selecting **Single** on the walk-right menu for the **Settings** option brings up the appropriate panel:

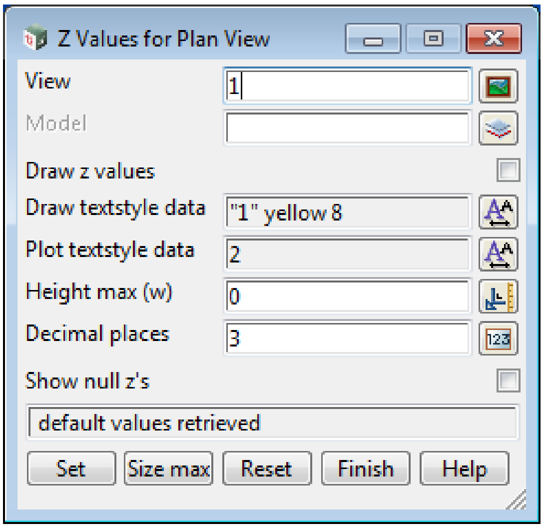


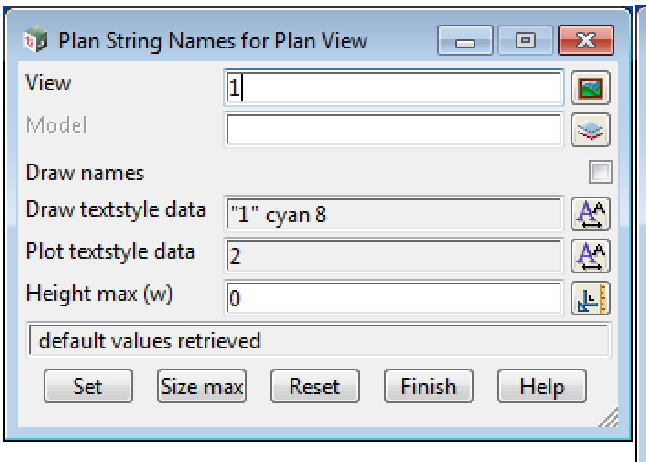


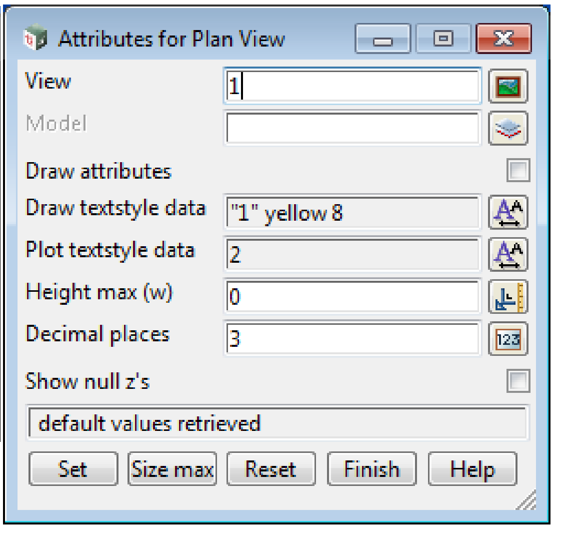





Vertex Indices for Plan View
View
1
Model
Draw indices
Draw textstyle data
"1" cyan 8
Plot textstyle data
2
Height max (w)
0
default values retrieved
Set
Size max
Reset
Finish
Help


Z Values for Plan View
View
1
Model
Draw z values
Draw textstyle data
"1" yellow 8
Plot textstyle data
2
Height max (w)
0
Decimal places
3
Show null z's
default values retrieved
Set
Size max
Reset
Finish
Help


Plan String Names for Plan View
View
1
Model
Draw names
Draw textstyle data
"1" cyan 8
Plot textstyle data
2
Height max (w)
0
default values retrieved
Set
Size max
Reset
Finish
Help


Attributes for Plan View
View
1
Model
Draw attributes
Draw textstyle data
"1" yellow 8
Plot textstyle data
2
Height max (w)
0
Decimal places
3
Show null z's
default values retrieved
Set
Size max
Reset
Finish
Help

The fields and buttons used in these panels have the following functions.

Field Description Type Defaults Pop-Up

View View box - input/output current view available views

name of the view to modify the drawing parameters for Values Text/Crosses/Text.

Model model box

*if **not blank**, then the String Text/Vertices/Values Text will only be displayed for this model.*

*If **blank**, then all models **not mentioned** in the Table option use these parameters for drawing String Text/Vertices/Values Text on a view.*

Note that his field will not be present if PLAN_TABLE_SETTINGS_4D *is zero.*

For Text

Draw text tick box

*If **ticked**, then Text for strings (text strings, vertex text, segment, text, values text etc) will be drawn (as long as it passes the check against* **Height max (w)***).*
*If **not ticked**, then **no** Text is drawn for strings on the plan view.*

For Crosses at Vertices

Draw Crosses tick box

*If **ticked**, then the Crosses at Vertices will be drawn (as long as it passes the check against* **Size max (w)***).*
*If **not ticked**, then **no** Crosses at vertices are drawn for strings on the plan view.*

Colour colour box default pt colour available colours

colour of the crosses at the vertices.

Size (p) real box default text height

size in pixels that the crosses are drawn on a plan view

Size max (w) real box 0

when you have a plan view, then for the current zoom settings of the view, the vertices which are drawn at a screen size (pixels) has an equivalent size in world units.

As you zoom in, the equivalent size in world units decreases. That is, the screen crosses gets smaller in equivalent world units.

As you zoom out, the equivalent size in world units increases. That is, the screen crosses gets larger in equivalent world units.

If **Size max (w)** *is non-zero then when the zoom settings on the plan view is such that the equivalent height of the crosses in world units is **greater** than* **Size max (w)***, then the crosses are not drawn. Consequently as one zooms out, the crosses at the vertices will eventually stop drawing.*

If **Size max (w)** *is zero, the crosses at the vertices are always drawn.*

Summarising:** if non-zero: when the plan view is such that the height of the crosses in world units is greater than* **Size max (w), then the crosses are not drawn. Hence as one zooms out, the crosses will eventually stop drawing.*

Size (w) real box 0

size in world units of the cross when plotted or output.

Angle angle box 0

the angle of rotation of the cross about the (x,y) position of the string vertices.

The angle is measured in a counterclockwise direction from the positive x-axis. The units for angle are degrees minutes and seconds and it is entered in 4.17.1 HP Notation*.*

For Values Text only

Draw Values Text tick box

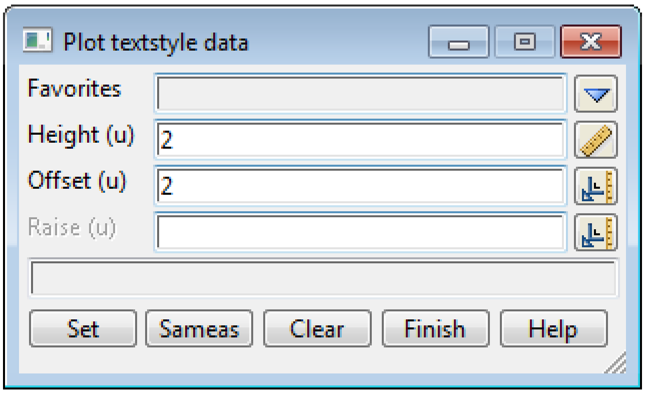
*If **ticked**, then the Values Text will be drawn (as long as it passes the check against* **Height max (w)***).*
*If **not ticked**, then **no** Values Text is drawn for strings on the plan view.*

Draw textstyle data textstyle data box available textstyle datas

textstyle data used when drawing the Values Text on the plan view. The height, offset and raise are given in screen units (pixels).

Plot textstyle data plot textstyle data box available textstyle datas

a special plot textstyle data to define the height, offset and raise values to use when the Values Text is plotted or output (using the File =>Data output options).
When the Values Text is plotted, the units are millimetres (paper size).
When the data is Values Text, the units are world (world size).



For String Text and Values Text

Height max (w) real box 0

when you have a plan view, then for the current zoom settings of the view, the String Text/Values Text which is drawn at a screen size (pixels), has an equivalent size in world units.
As you zoom in, the equivalent size in world units decreases. That is, the screen texts gets smaller in equivalent world units.

As you zoom out, the equivalent size in world units increases. That is, the screen texts gets larger in equivalent world units.

*If **Height max (w)** is non-zero then when the zoom settings on the plan view is such that the equivalent height of the String Text/Values Text in world units is **greater** than* **Height max (w)***, then the Strings Text/Values Text is not drawn. Consequently as one zooms out, the Strings Text/Values Text will eventually stop drawing.*
*If **Height max (w)** is zero, the Strings Text/Values Text is always drawn.*

Summarising:** if non-zero: when the plan view is such that the height of the Strings Text/Values Text in world units is greater than* **Height max (w), then the Strings Text/Values Text is not drawn. Hence as one zooms out, the Strings Text/Values Text will eventually stop drawing.*

For Z Values and Attributes

Decimal places real box 3

number of decimal places when displaying the Values Text for Z values and any real Attributes.

***Note** - this field only appears when the Values are a real number.*

Show null z's tick box

*if **ticked**, null z values and null real attributes values are displayed as null.*

*If **not ticked**, no null z value or null real attributes are displayed.*

***Note** - this field only appears when the Values Text are **z values** or Attributes.*

Buttons at Bottom

Set button

set the values in the panel and then redraw the plan view.

Size max button

*when clicked, the **Size max (w)** or **Height max (w)** field is given the current equivalent world size of the String Text/Crosses at Vertices/Values Text displayed on the given plan view. Hence if any further zoom out is done, the String Text/Crosses at Vertices/Values Text will stop drawing.*

After clicking **Size max***, the* **Set** *button must then be clicked to set the new **Size max (w)** or **Height max (w)** value for the view.*

Reset button

reset all the parameters to the defaults for plan views.

Note

Even though some of the drawing is controlled by the models in **Table**, the drawing for the models that are not controlled by **Table** can be quickly turned **on** and **off** by using the **Toggle** menu. See 9.6.4 Toggle Menu for Values Text, Crosses and Text on Plan Views.

Go to the next section 9.6.2 Table for Values Text, Crosses and Text on Plan Views or return to 9.6 Displaying Values Text, Vertices and Text on a Plan View.

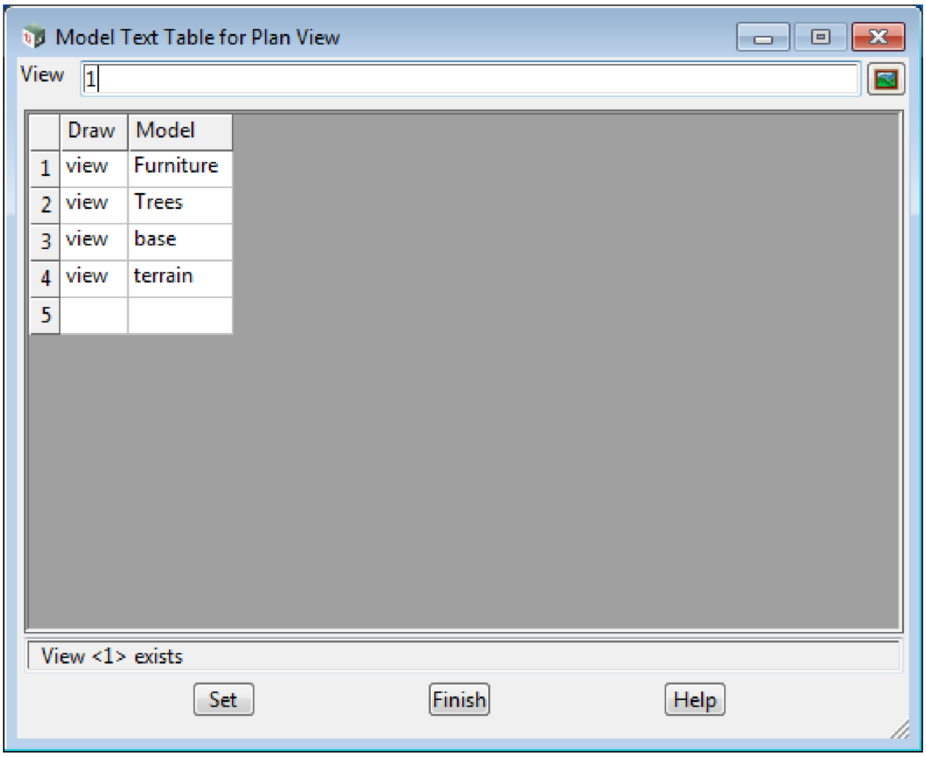
9.6.2 Table for Values Text, Crosses and Text on Plan Views

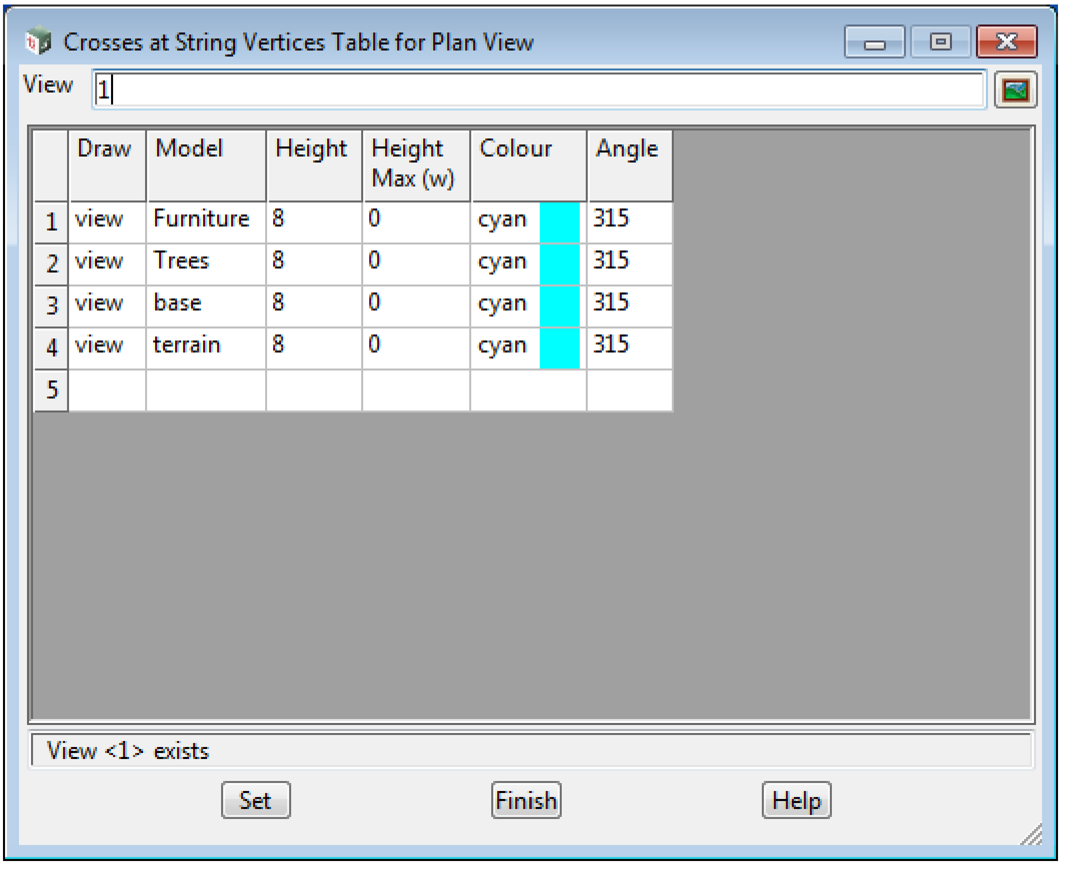
Vertex_Segment_UID_s_Table_for_Plan_View***Plan View =>Settings*** has the option **Text** to draw ***String Text,*** the option **Vertices** to draw ***Crosses at*** all the **Vertices** in a string (often just called ***Crosses***) and the options ***Vertex/Segment UIDs***, ***Point/Vertex Id's***, ***Vertex indices***, ***Z values***, ***Names*** and ***Attributes*** for labelling specific values from a string as text (the **Values Text**). For information on these options see 9.6 Displaying Values Text, Vertices and Text on a Plan View.

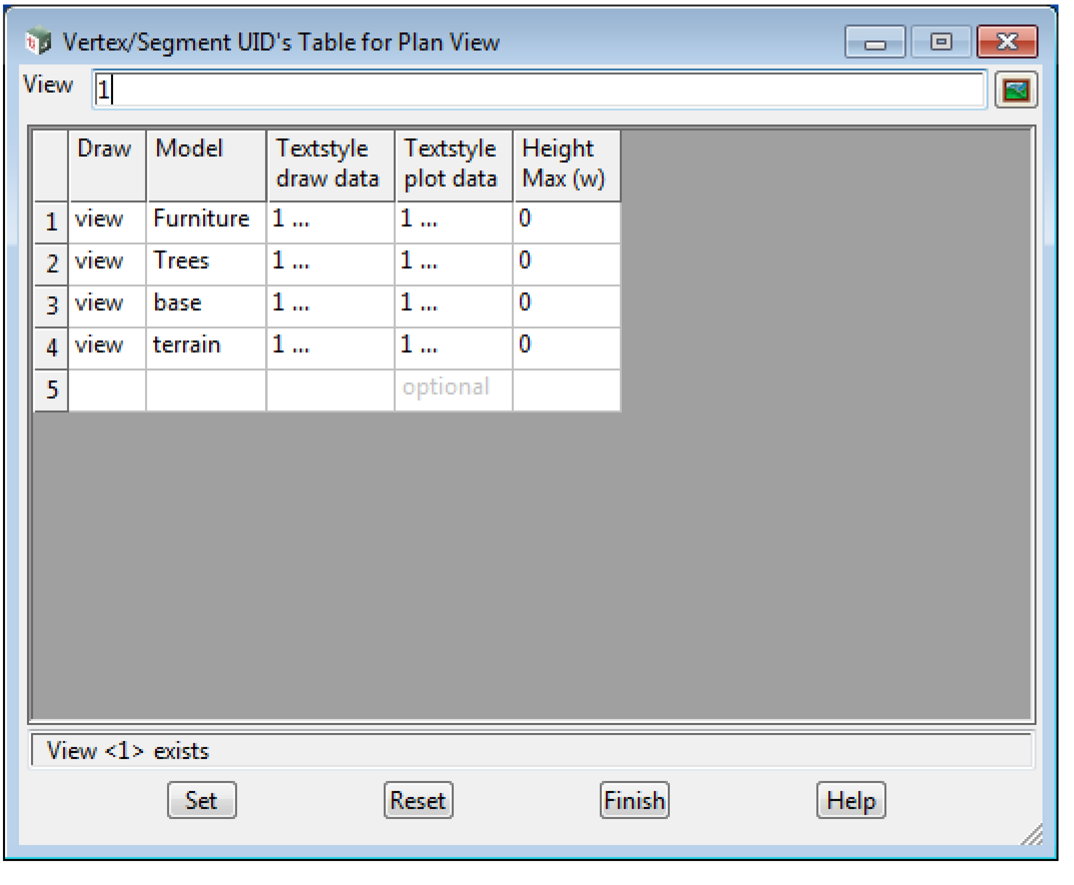
All these options work similarly and will be documented together.

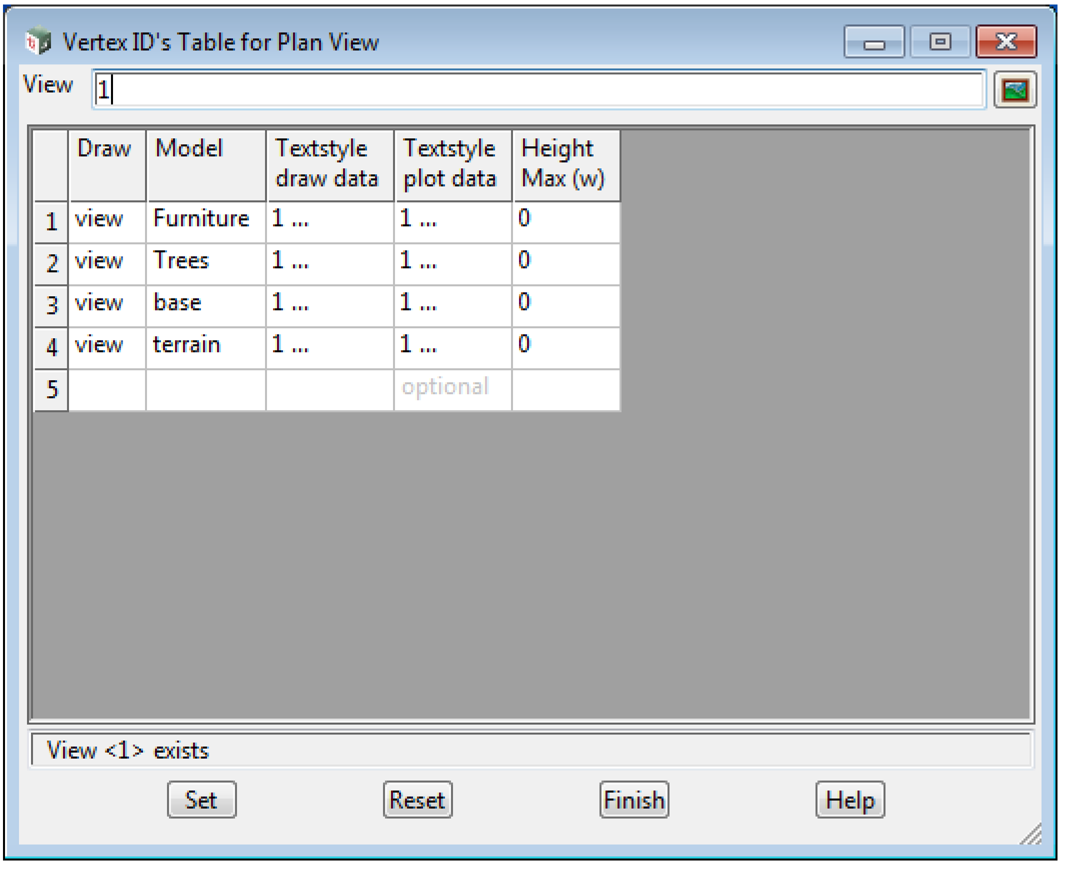
The **Table** option displays a list of models on the view with their ***Values Text/Crosses/Text*** drawing parameters. These parameters can then be modified in the table.

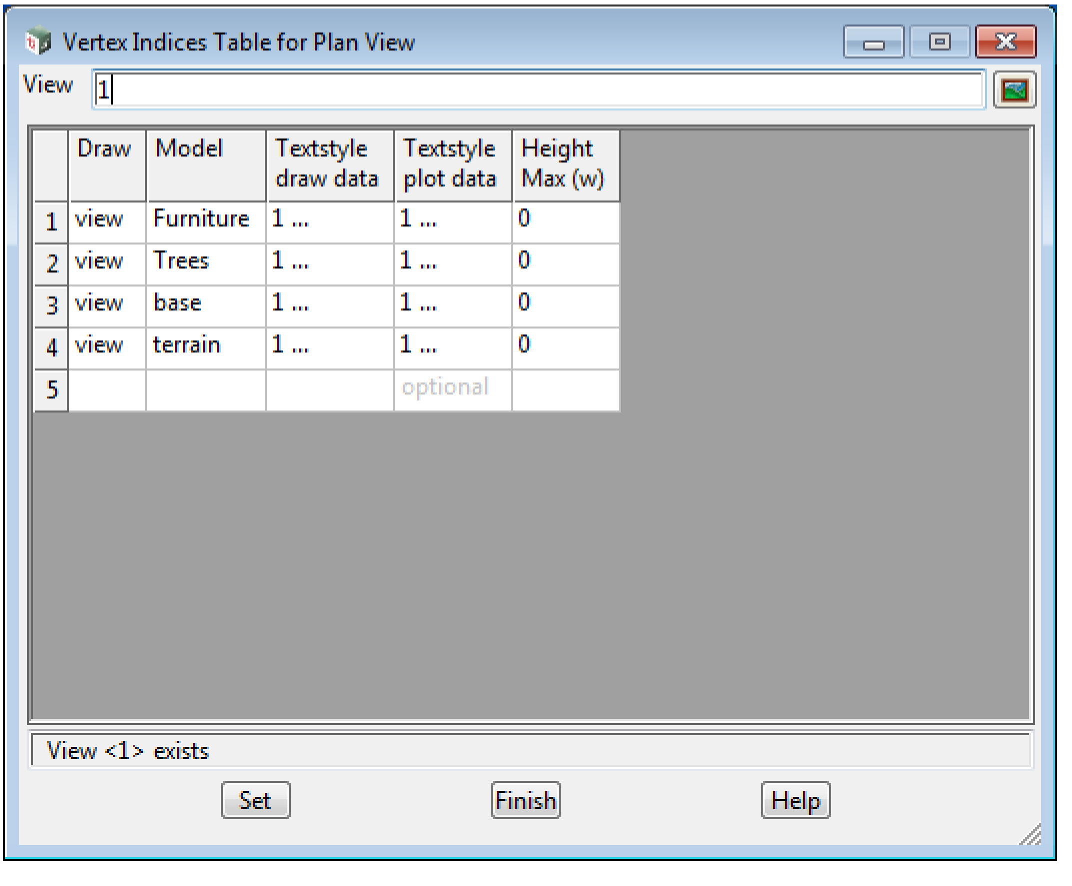
Selecting **Table** brings up the panel appropriate to the **Settings** option:

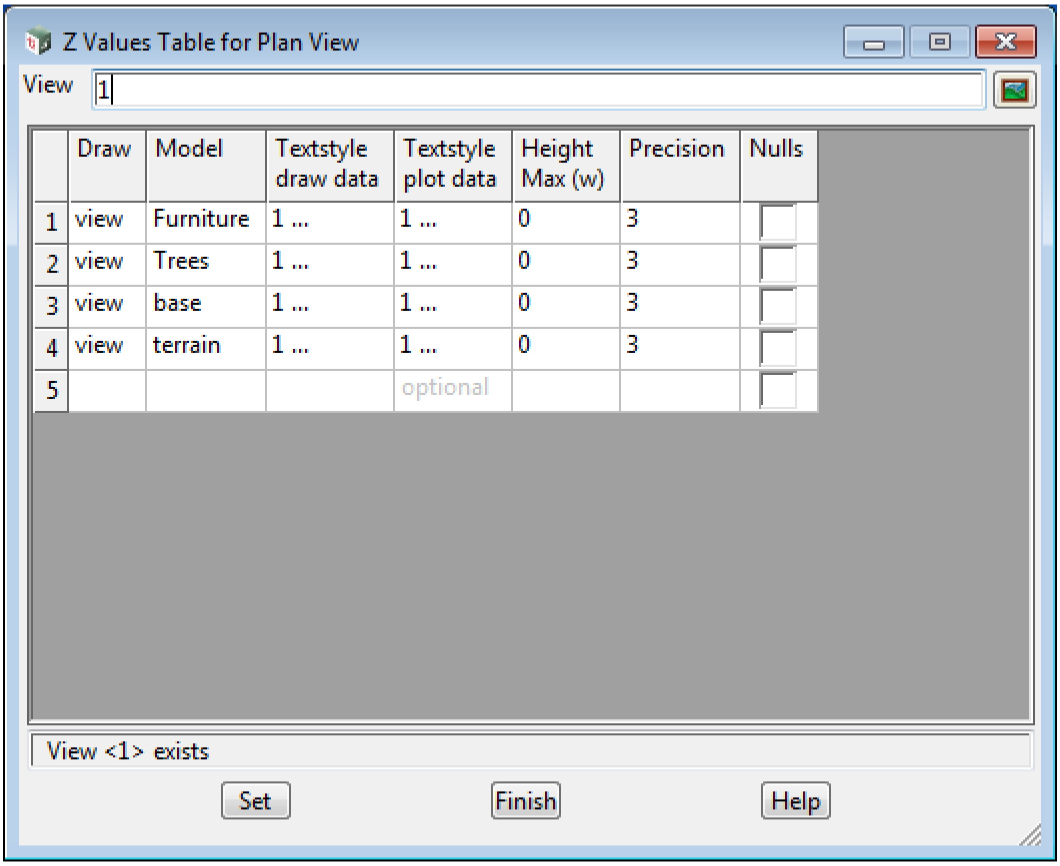


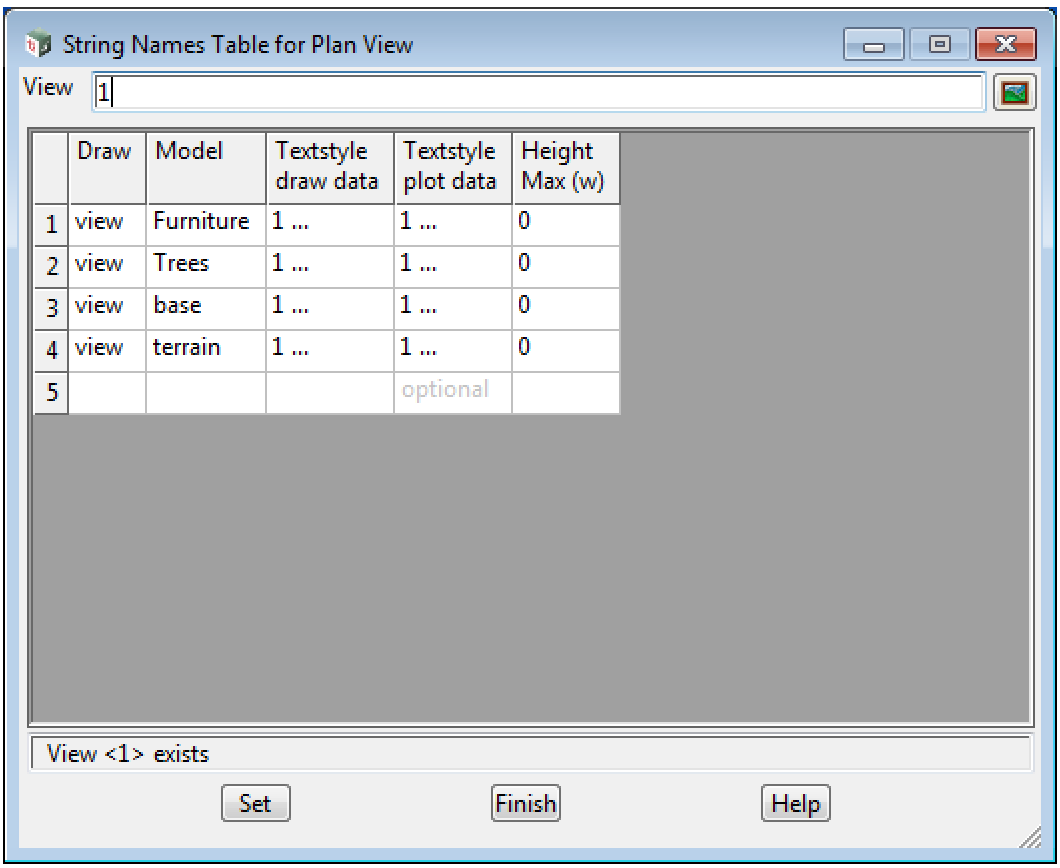

Crosses at String Vertices Table for Plan View
View
1
Draw
Model
Height
Height Max (w)
Colour
Angle
1 view Furniture 8 0 cyan 315
2 view Trees 8 0 cyan 315
3 view base 8 0 cyan 315
4 view terrain 8 0 cyan 315
5
View <1> exists
Set
Finish
Help

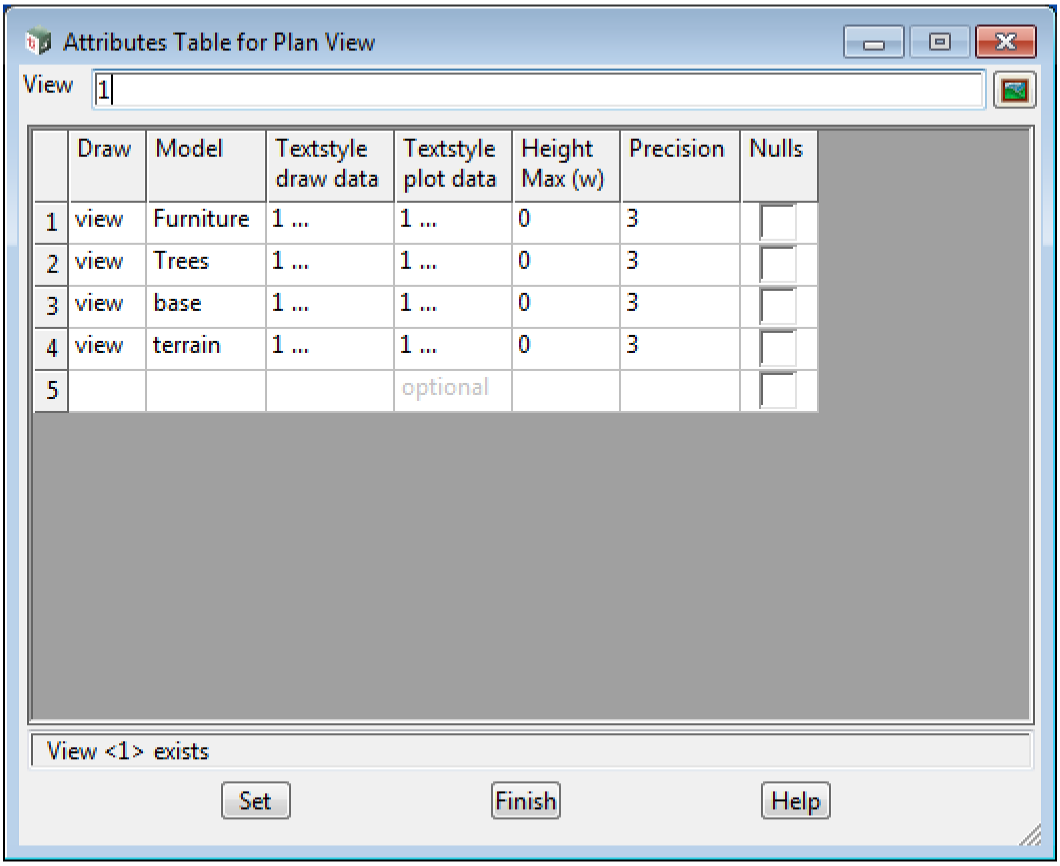

Vertex/Segment UID's Table for Plan View
View
1
Draw
Model
Textstyle draw data
Textstyle plot data
Height Max (w)
1 view Furniture 1 ... 1 ... 0
2 view Trees 1 ... 1 ... 0
3 view base 1 ... 1 ... 0
4 view terrain 1 ... 1 ... 0
5
optional
View <1> exists
Set
Reset
Finish
Help


Vertex ID's Table for Plan View
View 1
Draw | Model | Textstyle draw data | Textstyle plot data | Height Max (w)
1 view Furniture 1 ... 1 ... 0
2 view Trees 1 ... 1 ... 0
3 view base 1 ... 1 ... 0
4 view terrain 1 ... 1 ... 0
5 optional
View <1> exists
Set
Reset
Finish
Help


Vertex Indices Table for Plan View
View 1
Draw | Model | Textstyle draw data | Textstyle plot data | Height Max (w)
1 view Furniture 1 ... 1 ... 0
2 view Trees 1 ... 1 ... 0
3 view base 1 ... 1 ... 0
4 view terrain 1 ... 1 ... 0
5 optional
View <1> exists
Set
Finish
Help


Z Values Table for Plan View
View
1
Draw
Model
Textstyle draw data
Textstyle plot data
Height Max (w)
Precision
Nulls
1 view Furniture 1 ... 1 ... 0 3
2 view Trees 1 ... 1 ... 0 3
3 view base 1 ... 1 ... 0 3
4 view terrain 1 ... 1 ... 0 3
5 optional
View <1> exists
Set
Finish
Help


String Names Table for Plan View
View
1
Draw
Model
Textstyle draw data
Textstyle plot data
Height Max (w)
1 view Furniture 1 ... 1 ... 0
2 view Trees 1 ... 1 ... 0
3 view base 1 ... 1 ... 0
4 view terrain 1 ... 1 ... 0
5 optional
View <1> exists
Set
Finish
Help



The fields and buttons used in these panels have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input/output	current view	available views

name of the view to modify the parameters for displaying the Values Text/Crosses/Text.

Grid

Draw	choice box		on, off, view

*If set to **on**, then regardless of the **Single** value, if that model is on the view, the Values Text/Crosses/Text for the model are **drawn** (as long as it passes the check against* **Height max (w)**).

*If set to **off** then regardless of the* **Single** *value, if that model is on the view, the Values Text/Crosses/Text for the model are **not drawn**.*

*If set to **view**, then this row is ignored and if that model is on the view, the Values Text/Crosses/Text for the model **obey** the values set by* **Single**.

*So the **Draw** column in* **Table** *is an **override** for* **Single**.

For information on changing the values of the cells in the Draw column, see Changing the Values in the Draw Column*).*

Model	model box		

*if **not blank**, then the display of String Text/Crosses at Vertices/Values Text for the strings in this model will obey the choice for this row of the* **Draw** *column.*

For Crosses at Vertices

Colour	colour box	default pt colour	available colours

colour of the crosses at the vertices

Height real box default text height

size in pixels that the crosses are drawn on a plan view.

Height max (w) real box 0

*if **non-zero**:*
then if the size of the crosses in world units is greater than **Size max (w)**, *the crosses are not drawn. Hence zooming out will eventually stop the crosses from drawing.*

Angle angle box 0

the angle of rotation of the cross about the (x,y) position of the string vertices.

The angle is measured in a counterclockwise direction from the positive x-axis. The units for angle are degrees minutes and seconds and it is entered in 4.17.1 HP Notation.

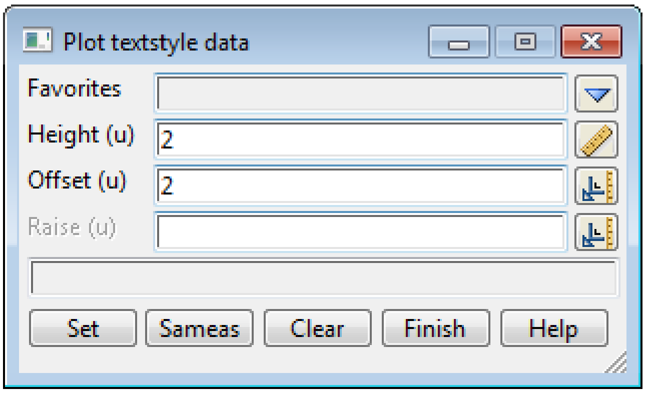
For Values Text

Textstyle draw data textstyle data box available textstyle datas

textstyle data to define the drawing on the view of the Values Text in this model. The height, offset and raise are given in screen units (pixels).

Textstyle plot data plot textstyle data box available textstyle datas

a plot textstyle data to define the height, offset and raise values to use when the Values Text in the model is plotted or output (using the File =>Data output options).
When the Values Text in the model is plotted, the units are millimetres (paper size).
When the Values Text in the model is output, the units are world (world size).



Height max (w) real box 0

when you have a plan view then for the current settings for the Values Text, the Values Text which is drawn at a screen size (pixels), will have an equivalent size in world units.
As you zoom in, the equivalent size in world units will decreases. That is, the screen texts gets smaller in equivalent world units.

As you zoom out, the equivalent size in world units increases. That is, the screen texts gets larger in equivalent world units.

If **Height max (w)** *is non-zero then when the zoom settings on the plan view is such that the equivalent height of the Values Text in world units is* ***greater*** *than* **Height max (w)**, *then the Values Text is not drawn. Consequently as one zooms out, the Values Text will eventually stop drawing.*
If **Height max (w)** *is zero, the Values Text is always drawn.*

Summarising: *if non-zero: when the plan view is such that the height of the Values Text in world units is greater than* **Height max (w)**, *then the Values Text is not drawn. Hence as one zooms out, the Values Text will eventually stop drawing.*

Precision real box 3

number of decimal places when displaying the Values Text.

***Note** - this column only appears when the Values are a real number.*

Nulls tick box

*if **ticked**, null z values are displayed as null.*
*If **not ticked**, no z value is displayed at null z-values.*

***Note** - this column only appears when the Values Text are **z values**.*

Buttons at Bottom

Set button

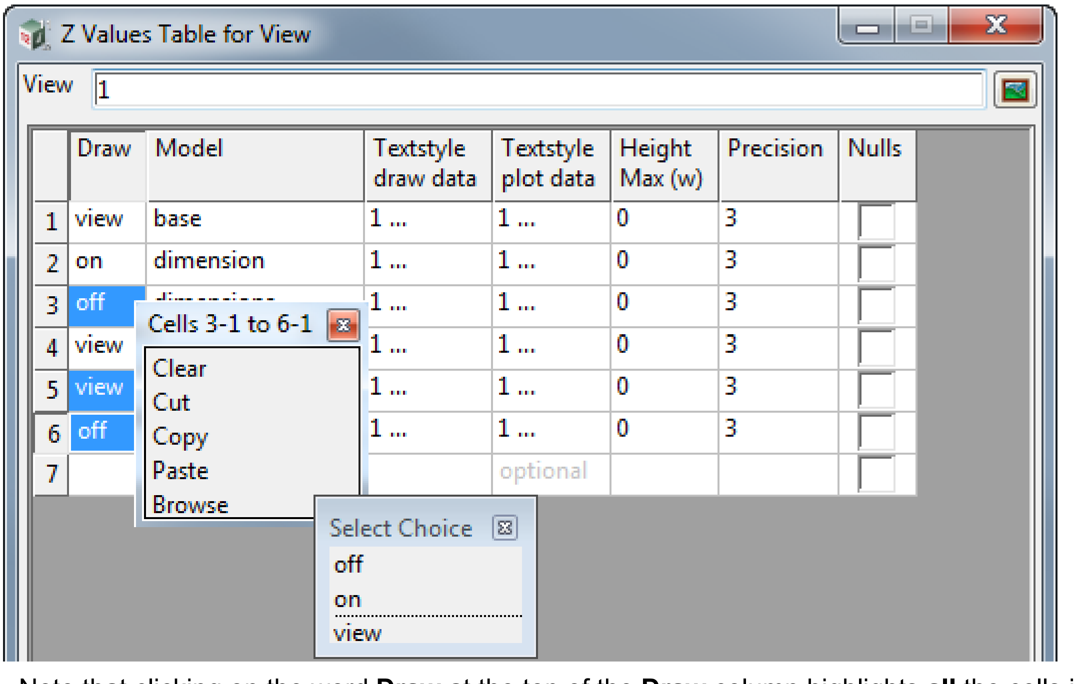
set the values in the panel and then redraw the plan view.

Changing the Values in the Draw Column

To change the values in a cell in the **Draw** column, you can type into the cell, or it is better (and safer if you can't type very well) to click **RB** over the cell and bring up the **Select Choice** menu and select **off**, **on** or **view** from the menu.

If for some reason you do not get the menu with the choices on it coming up, the menu you do get will have **Browse** on it and clicking on **Browse** will bring up the choices.**off**, **on** and **view**.

To set all, or a number of the values in the **Draw** column at once, highlight the ones to change and click **RB** to bring up the **Cells** menu and select **Browse**. The choice of **off**, **on** and **view** will be made to all the highlighted cells.



Note that clicking on the word **Draw** at the top of the **Draw** column highlights **all** the cells in the column. Holding down **<Ctrl>** and clicking **LB** when over a cell highlights/unhighlights individual cells.

If for some reason you do not get the menu with the choices on it coming up, the menu you do get will have **Browse** on it and clicking on **Browse** will bring up the choices **Draw**, **Don't draw** and **Ignore**.

Go to the next section 9.6.3 Toggle Walk-Right for Value Text, Crosses and Text on Plan Views or return to 9.6 Displaying Values Text, Vertices and Text on a Plan View.

9.6.3 Toggle Walk-Right for Value Text, Crosses and Text on Plan Views

Plan_View_Toggle_Vertex_Segment_UID_s

Plan View =>Settings has the option **Text** to draw ***String Text,*** the option **Vertices** to draw ***Crosses at*** all the **Vertices** in a string (often just called ***Crosses***) and the options ***Vertex/ Segment UIDs***, ***Point/Vertex Id's***, ***Vertex indices***, ***Z values***, ***Names*** and ***Attributes*** for labelling specific values from a string as text (the **Values Text**). For information on these options see 9.6 Displaying Values Text, Vertices and Text on a Plan View.

All these options work similarly and will be documented together.

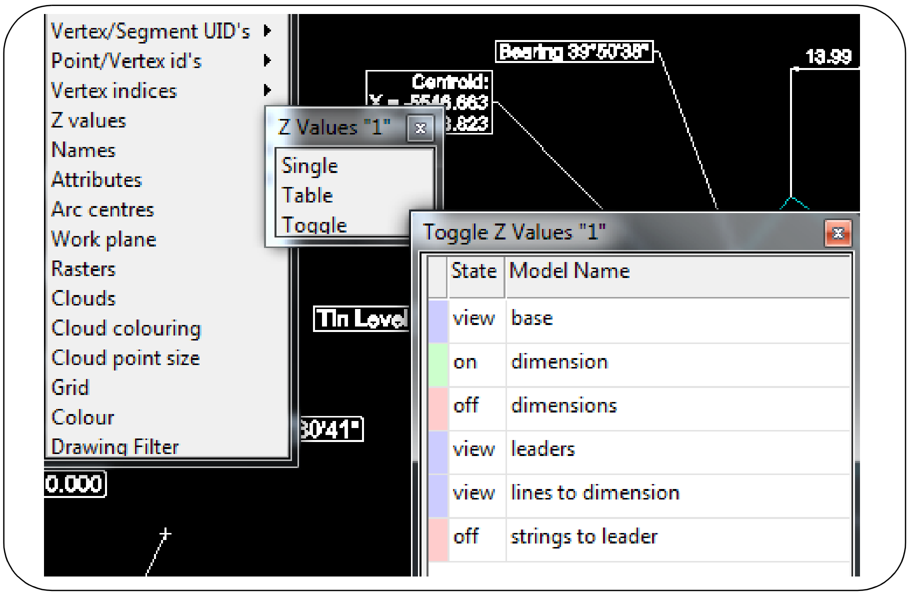
Walking-right on the **Toggle** option displays a list of models on the view with their ***Value Text/ Crosses***/***Text*** *Draw* flag from the **Table** for that **Setting**.

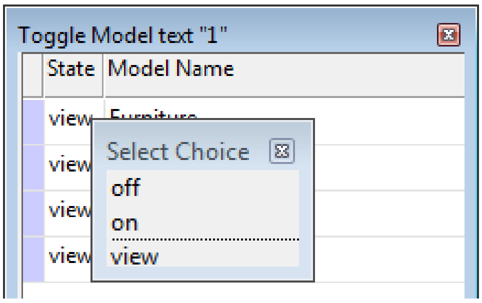
The **String Text/Crosses at Vertices/Values** ***Text*** *Draw* flag can be set to **yes**, **no** or **view** for the model by selecting the model on the list, where

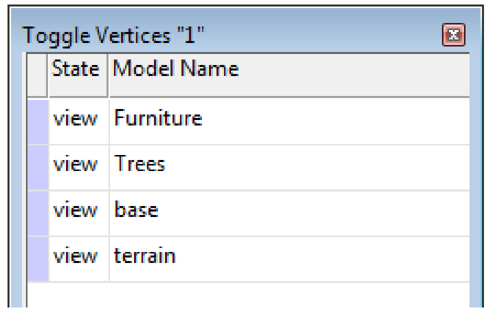
If **State** is set to **on**, then if that model is on the view, the Values Text/Crosses/Text for the model are **drawn regardless** of the **Single** value. Green is displayed beside the state when it is **on**

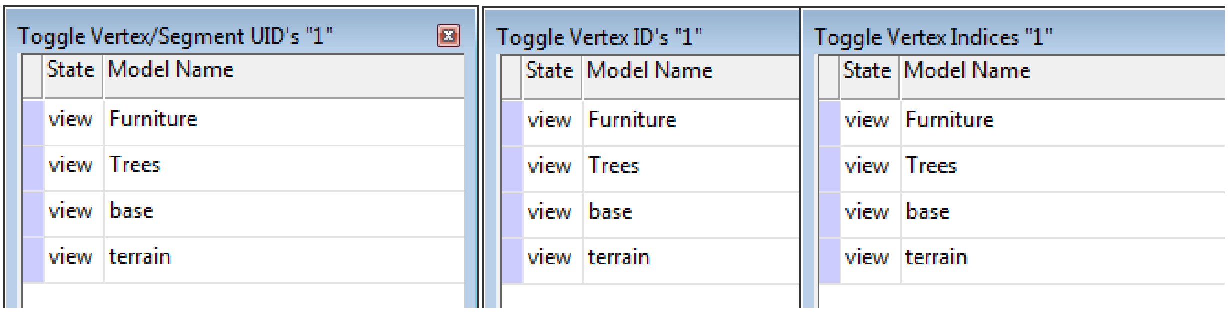
If **State** is set to **off**, then if that model is on the view, the Values Text/Crosses/Text for the model are **not drawn regardless** of the **Single** value. Red is displayed beside the state when it is **off.**

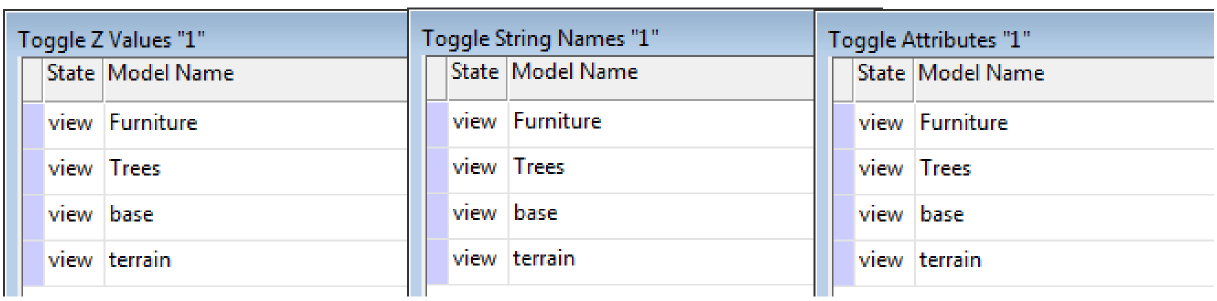
If **State** is set to **view**, then this row is ignored and if that model is on the view, Values Text/ Crosses/Text for the model **obey** the **Single** value. Green is displayed beside the state when it is **view**



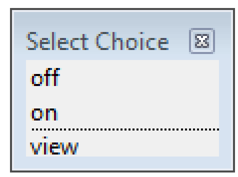








Clicking **RB** in a row in the **State** column will bring up a **Select Choice** menu.



Selecting **on**, **off** or **view** changes the value for the appropriate row in the associated **Table** and also redraws the view using the modified values.

Go to the next section 9.6.4 Toggle Menu for Values Text, Crosses and Text on Plan Views or return to 9.6 Displaying Values Text, Vertices and Text on a Plan View.

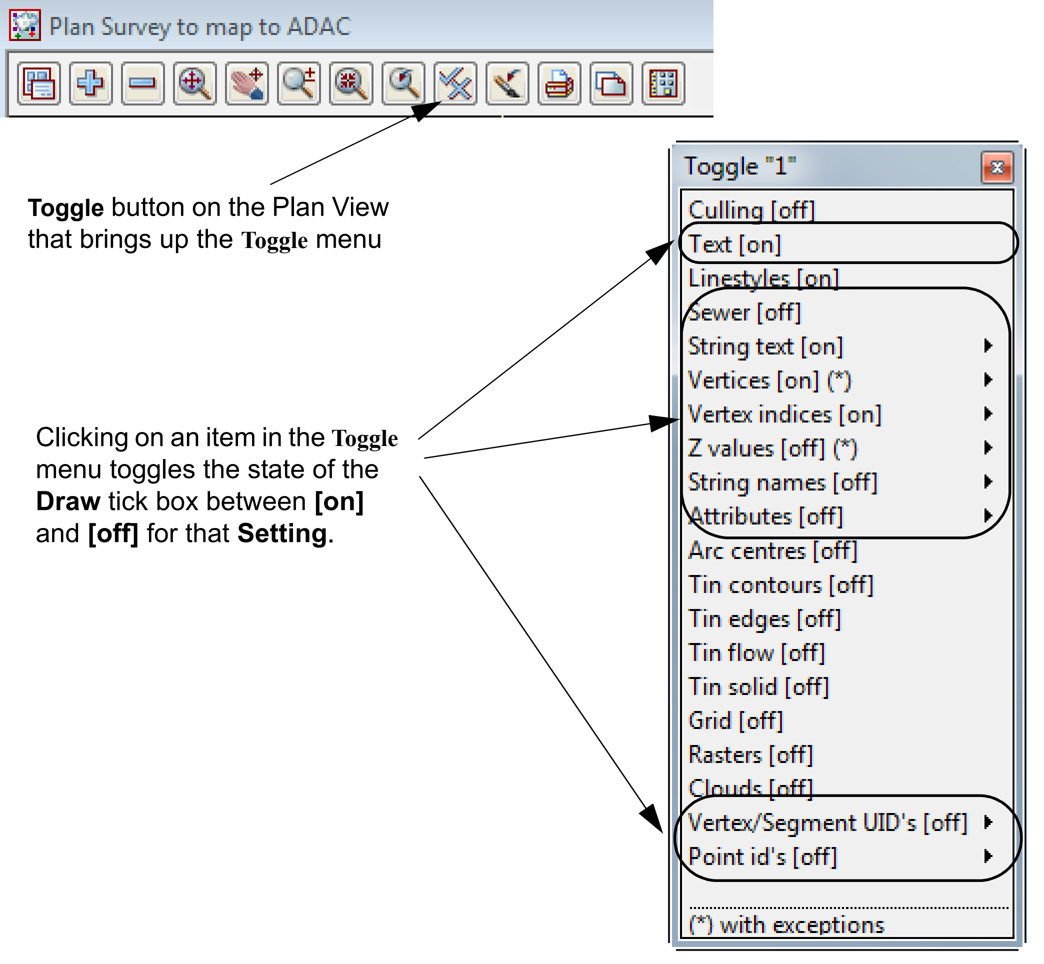
9.6.4 Toggle Menu for Values Text, Crosses and Text on Plan Views

Plan View =>Settings has the option **Text** to draw ***String Text,*** the option **Vertices** to draw ***Crosses at*** all the **Vertices** in a string (often just called **Vertices** or ***Crosses***) and the options ***Vertex/Segment UIDs***, ***Point/Vertex Id's***, ***Vertex indices***, ***Z values***, ***Names*** and ***Attributes*** for labelling specific values from a string as text (the **Values Text**). For information on these options see 9.6 Displaying Values Text, Vertices and Text on a Plan View.

All these options work similarly and will be documented together.

To make it easy to toggle the ***Draw*** tick box in the **Single** panel for any of the **Setting** without having to bring the **Single** panel up, there is a **Toggle** button on the **Plan View** to bring up the **Toggle** menu for the view.

Each item on the **Toggle** menu can be clicked to toggle the **Draw** tick box for that particular **Setting**.



On the **Toggle** menu there is an **[on]** and **[off]** after the menu item to indicate the default state for the view (the value of **Single**). And if there is also a **(*)** after the **[on]** or **[off]** to indicate there are models in Table is that are different to the **Single** state (an **exception**).

That is:

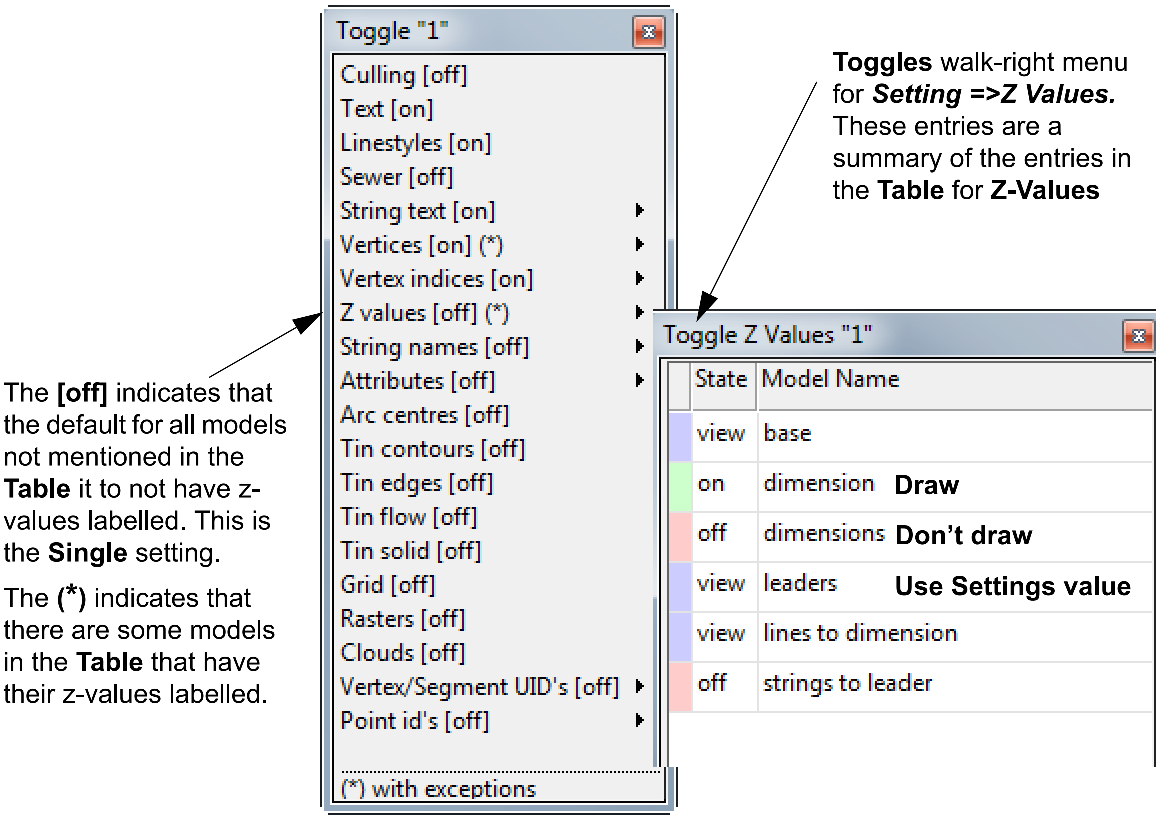
[on] means that the default for all models on the view is to draw the value, and that the default is applying to all the models currently on the view.

[on] (*) means that the default for all models on the view is to draw the value, BUT there are some models in Table that **will not** draw the value.

[off] means that the default for all models on the view is to not draw the value, and that the default is applying to all the models currently on the view.

[off] (*) means that the default for all models on the view is to not draw the value, BUT there are some models in Table that **will** draw the value.

There are also walk-right on each of the appropriate menu items and the walk-right menu is the **Toggles** walk-right menu for that **Setting**.



Unlike the **Toggle** walk-right menus where there a walk-right menu for each of the **Settings** and they toggle the **Draw** mode for the models in the **Table** for that **Setting**, the **Toggle** menu has **one line** on the menu for each of the **Settings**. And the walk-rights on the **Toggle** menu are the individual **Toggle walk-right** menus.

So the **Toggle** menu toggles the **Draw** mode in the **Single** panel for each **Setting**, and the **Toggle walk-rights** toggle the **Draw** modes for each model in the **Table** for that **Setting**.

Warning for the Toggle Menu

Although clicking on an item in the **Toggles** menu immediately reverses the **[on]**/**[off]** state of the view and updates the view accordingly, the **[on]/[off]** on the **Toggles** menu does **NOT** dynamically change.

So if the **Toggles** menu is pinned up, it does not refresh itself and show the change in the [**on]/[off]** state. You only see the change if you close the **Toggle** menu and then bring it up again.

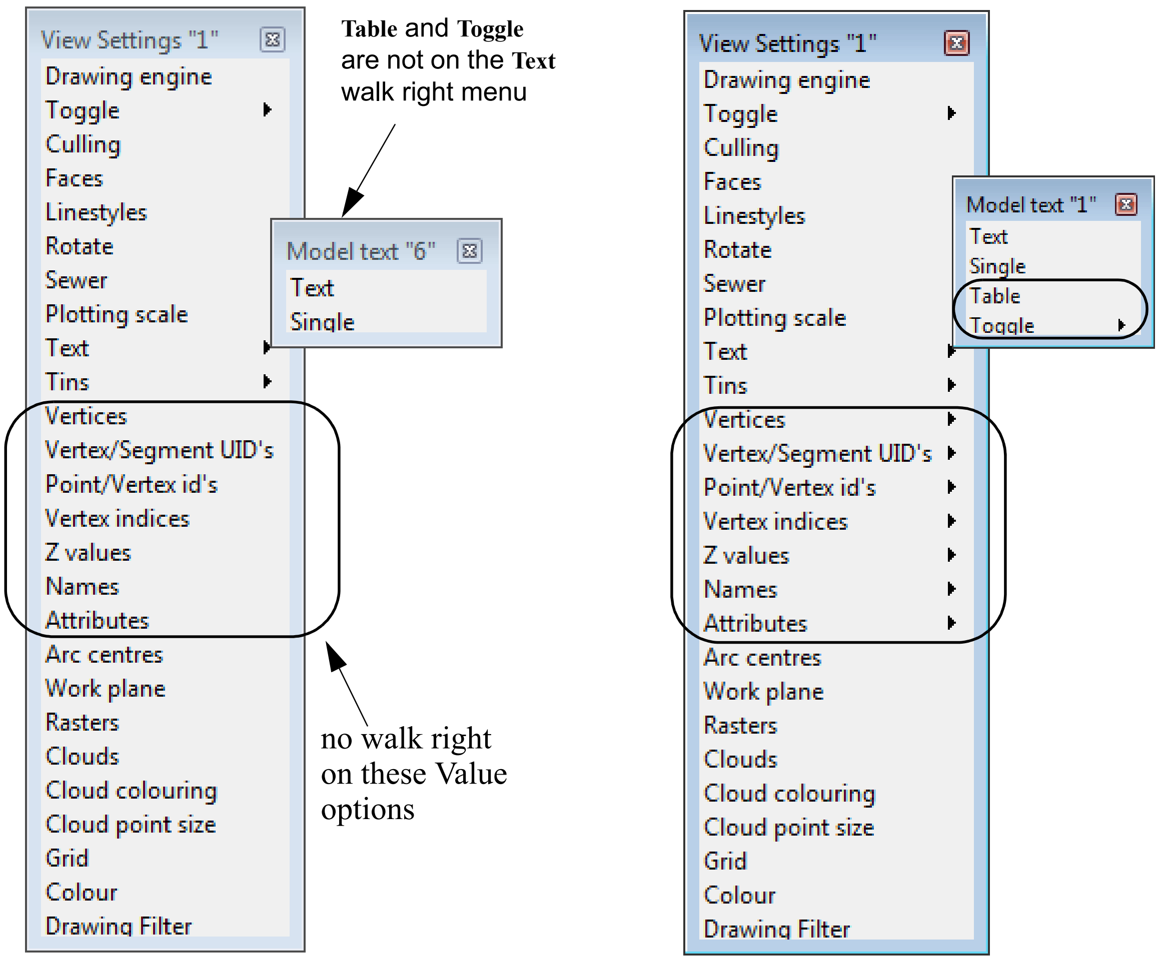
9.6.5 Env Var PLAN_TABLE_SETTINGS_4D Set to 0:

Plan View =>Settings has an option **Text** to draw ***String Text,*** an option **Vertices** to draw ***Crosses at*** all the **Vertices** in a string (often just called ***Vertices***) and the options ***Vertex/ Segment UIDs***, ***Point/Vertex Id's***, ***Vertex indices***, ***Z values***, ***Names*** and ***Attributes*** for labelling specific values from a string as text (for these options we call the **text** of the **value**, the **Values Text**.).

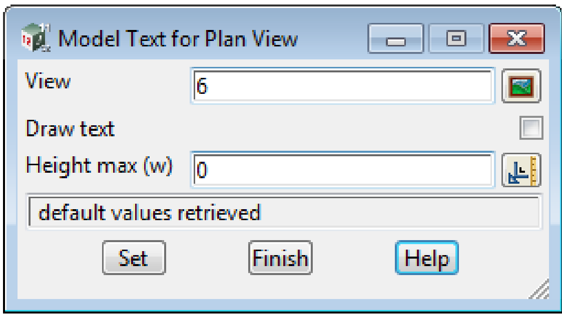
Normally the drawing of the ***Values Text/Crosses/Text*** is controlled by a setting with optional model overrides but it is possible to turn off the control for individual models by setting the environment variable PLAN_TABLE_SETTINGS_4D to 0. This is done in **Env.4d** > ***General >Plan table settings*** (see General and PLAN_TABLE_SETTINGS_4D).

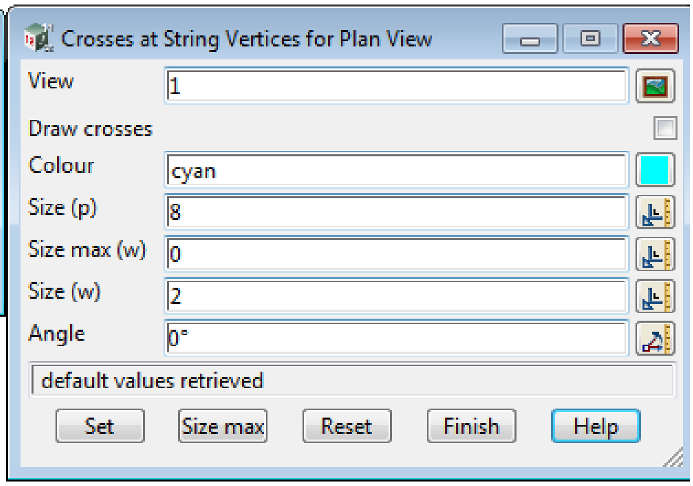
When individual model control is turned off, in the **Plan View Setting** menu there are no options **Table** or **Toggle** on the **Text** walk right menu, and **no walk right menu** on the ***Values*** menu items.

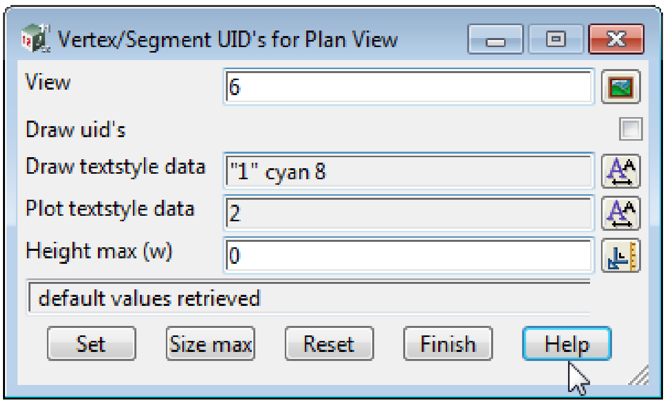
PLAN_TABLE_SETTINGS_4D = 0 **PLAN_TABLE_SETTINGS_4D = 1**

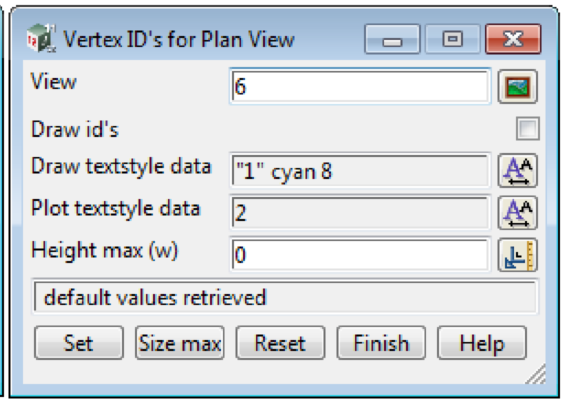


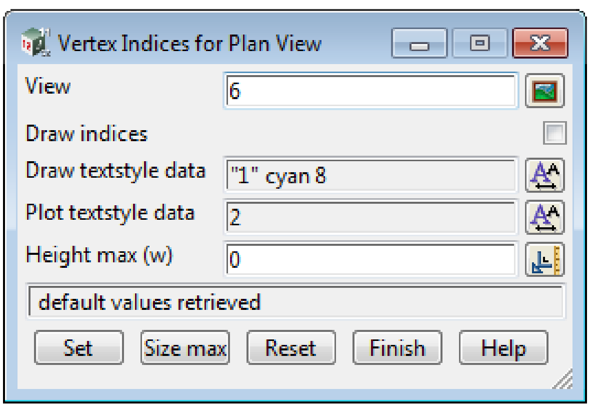
Also when PLAN_TABLE_SETTINGS_4D is 0, clicking on **Vertices/Values** item brings up the **Model Text for Plan View**, **Crosses at String Vertices for Plan View** or **Values for Plan View** panels, **without** the **Model** field. The fields in the panels then apply to the strings in ***all models*** on the view.

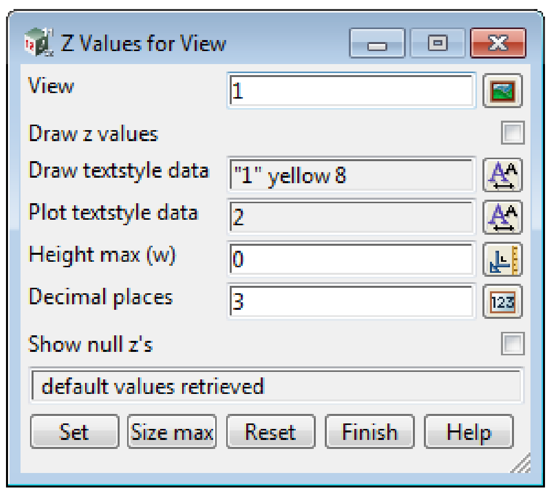

Model Text for Plan View
View
6
Draw text
Height max (w)
0
default values retrieved
Set
Finish
Help

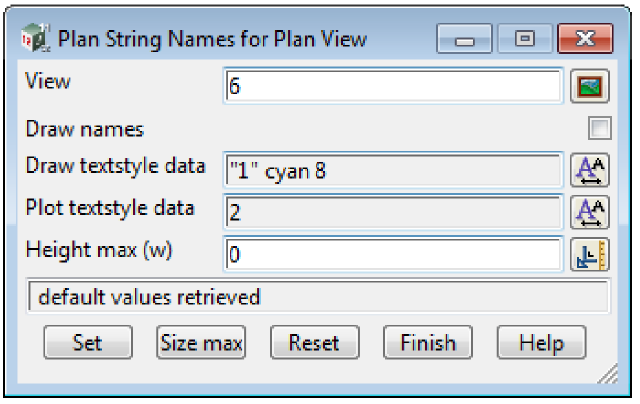

Crosses at String Vertices for Plan View
View
1
Draw crosses
Colour
cyan
Size (p)
8
Size max (w)
0
Size (w)
2
Angle
0°
default values retrieved
Set
Size max
Reset
Finish
Help

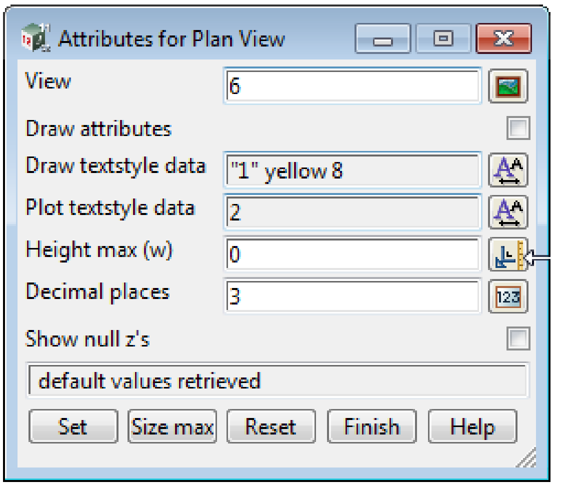

Vertex/Segment UID's for Plan View
View
6
Draw uid's
Draw textstyle data
"1" cyan 8
Plot textstyle data
2
Height max (w)
0
default values retrieved
Set
Size max
Reset
Finish
Help


Vertex ID's for Plan View
View
6
Draw id's
Draw textstyle data
"1" cyan 8
Plot textstyle data
2
Height max (w)
0
default values retrieved
Set
Size max
Reset
Finish
Help


Vertex Indices for Plan View
View
6
Draw indices
Draw textstyle data
"1" cyan 8
Plot textstyle data
2
Height max (w)
0
default values retrieved
Set
Size max
Reset
Finish
Help


Z Values for View
View
1
Draw z values
Draw textstyle data
"1" yellow 8
Plot textstyle data
2
Height max (w)
0
Decimal places
3
Show null z's
default values retrieved
Set
Size max
Reset
Finish
Help





These panels are identical to those when PLAN_TABLE_SETTINGS_4D is non zero except that there is no **Model** field. The fields in these panels then apply to the strings in ***all models*** on the view.

So the documentation for these panels is only given in the case where PLAN_TABLE_SETTINGS_4D is non zero and the **Model** field needs to be ignored. See 9.6.1 Single for Values Text, Crosses and Text on Plan Views.

10 ADAC

ADAC - Asset Design and As Constructed

The **ADAC XML Schema** is a vendor independent XML format introduced to help solve the problems of collecting and exchanging Asset data.

The purpose of ADAC XML is to allow many different Authorities to work with one standardised file format. The format is fully defined and accessible to everyone.

The ADAC XML schema is controlled by the **Institute of Public Works Engineering Australasia** (**IPWEA**). More information on ADAC at IPWEA can be found at www.engicom.com.au/products/adac2/

See

10.1 ADAC.XML Overview
10.2 12d Approach to ADAC XML
10.4 12d ADAC Menu

10.1 ADAC.XML Overview

See

10.1.1 ADAC XML Structure
10.1.2 ADAC Assets
10.1.3 ADAC Geometry Element
10.1.4 Guidelines from an Authority Requesting ADAC.XML

10.1.1 ADAC XML Structure

The ADAC Schema is fully described by the *ADAC XSD* (**X**ML **S**chema **D**efinition), which formally defines the structure and all the elements inside an ADAC XML file. That means that any ADAC.XML can be **validated** against the ADAC Schema XSD to see that it conforms to the Schema.

The **ADAC.XML Schema** and the ADAC.XSD are controlled and published by the IPWEA.

ADAC Version 4.1.0 is the latest version and IPWEA publishes the *ADAC Schema Help Files* for Version 4.1 (and also the earlier 4.0) (*see* www.engicom.com.au/products/adac2/).

The **ADAC XML Schema** is best thought of as a root or folder structure where the **Project** element encloses all data that is common to the whole project. So it is the root element or top folder.

The ADAC standard currently says there can only be **one** Project in an ADAC.XML file.

Elements inside the **Project** element can be single values or multi-levelled structures (nodes) containing other single elements and substructures.

At the first level inside **Project**, the elements are:

Version, ExportDateTime, Name, Owner, Receiver, DrawingNumber, Drawing Revision, ConstructionDate, CoordinageSystem, Drawing Extents, Description, Status, Software, Surveyor, Engineer and ProjectData.

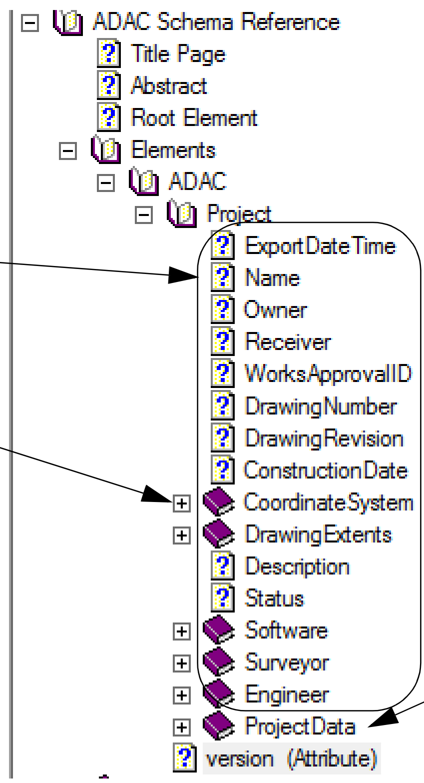
Some of these element are just single values (for example, *ExportDateTime* and *Owner*, and others, contain multi-levelled substructures of data (for example, *Surveyor*, *Engineer and ProjectData*).

Single values are shown with **?** before them

The **+** indicates that the element is not a single value but contains a multi-levelled substructure of values

NOTE:

The image is from the ADAC Version: 4.1.0 Schema Help File



Header Data - common to all the elements under *Project Data*

ProjectData - common to all the elements under Project Data values

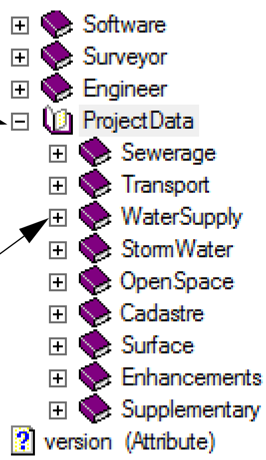
We will refer to all of these elements except **ProjectData** as **Header Data**.

The **Header Data** occurs only once and is at the beginning of the ADAC file.

The **ProjectData** element may appear insignificant in the expansion of the first level of **Project**, **BUT** it is the element that contains **all** the **physical items** covered by ADAC.

Expanding the **ProjectData** node shows it has the nine subelements *Sewerage, Transport, WaterSupply, OpenSpace, Cadastre, Surface, Enhancements* and *Supplementary.*

The **-** indicates that the element contains a multi-levelled substructure and it has been expanded in the diagram

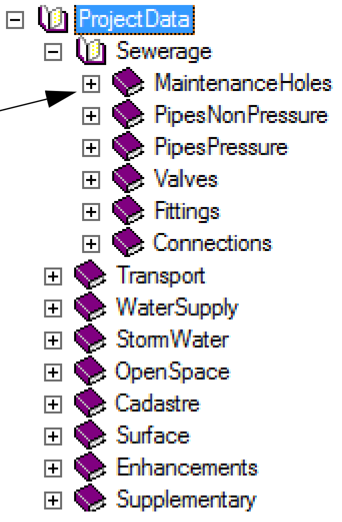


The **+** indicates that the element contains a multi-levelled substructure and it has not been expanded in the diagram

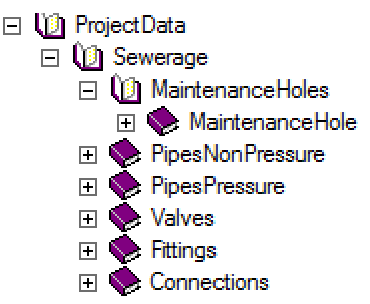
And each of the nine subelements have their own substructures

For example, **Sewerage** has the sub elements *MaintenanceHoles, PipesNonPressure, PipesPressure, Valves, Fittings* and *Connections*

The **+** indicates that the element **MaintenanceHoles** contains a multi-levelled substructure

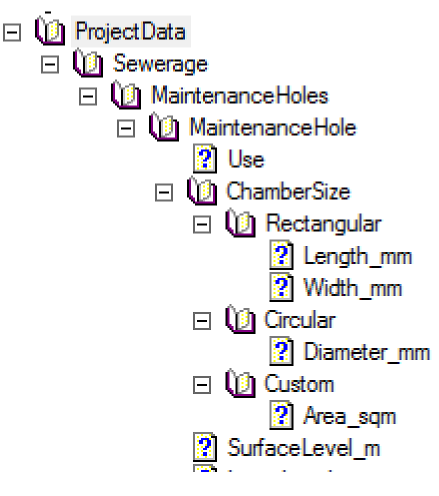


And **MaintenanceHoles** has a substructure that can be expanded out although it has only the one substructure, **MaintenanceHole**.



MaintenanceHole itself can be further expanded and part of **MaintenanceHole** is

.



All the structures and substructures are totally defined in the **ADAC XML Schema** and the structures must be strictly adhered to.

The **ADAC XML Schema** not only completely defines all the structures and substructures, but also has the **full definition** of all the elements, including the elements that can't be expanded any further (the **final** elements).

For the final elements, the ADAC Schema specifies everything. For example,

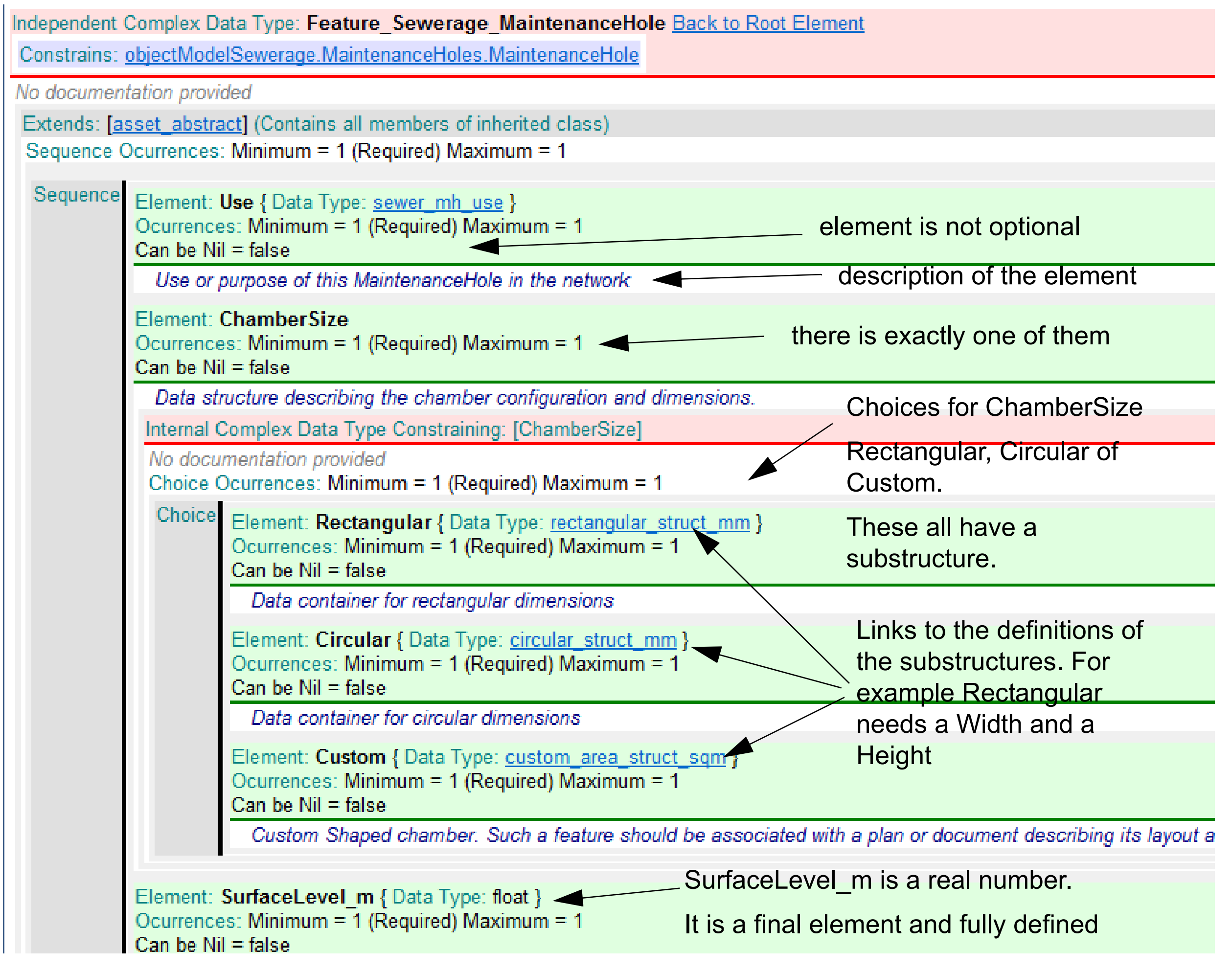
(a) their type - text, integer, real (float) or choices of these.

(b) what ranges the values may, or may not have

(c) if choices, then the list of possible choices (enumerations)

(d) if the item is optional (nillable = true) or mandatory (compulsory nillable = false)

For example, for the expansion of **MaintenanceHole,** the extract from the *ADAC XML 4.1.0 Schema* is



Hence this is how the **ADAC Schema** defines the **structure** of an ADAC.XML file and the definition of every element in the **ADAC Schema**.

If an XML file does not obey all the rules in the **ADAC Schema**, then the file is an **invalid XML file with respect to the ADAC Schema**.

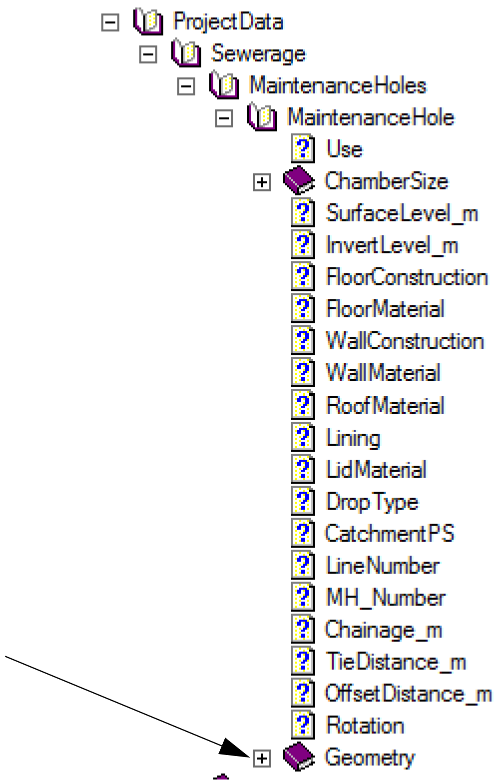
Continue to the next section 10.1.2 ADAC Assets or return to 10.1 ADAC.XML Overview or to 10 ADAC .

10.1.2 ADAC Assets

Expanding substructures can go on at every level until no more expansions can occur.

But there is one important time to stop further expanding an element and that is when **one** of the substructures is **Geometry**.

The presence of **Geometry** indicates that the element is a **fundamental** ADAC element that represents an **ADAC Asset** or **ADAC Feature**.



The presence of **Geometry** indicates that this element **MaintenanceHole** is a fundamental ADAC element representing an **ADAC Asset**

So although **MaintenanceHole** has the substructure **ChamberSize** that can be further expanded, the presence of the **Geometry** element says that **MaintenanceHole** is a fundamental ADAC element representing an **ADAC Asset**.

For the ADAC 4.1.0 Schema, there are sixty-eight (68) such **ADAC Assets** including MaintenanceHole, PipeNonPressure, Pavement, RoadEdge, SubSoilDrain, Hydrant, RetainingWall, ElectricalConduit, Tree, Sign, Lot, Easement, SpotHeight and Contour.

Everything that is being recorded in an ADAC.XML file must be classified as one of the 68 ADAC Assets.

To further define what an ADAC Asset is, most of the ADAC Asset elements have a **Use** or **Type** which provides further information about that particular instance of an ADAC Asset.

For example, **MaintenanceHole** has a **Use** which **must exist** and must have one of the values (enumerations):

ıdependent Simple Data Type: **sewer_mh_use** Back to Root Element
Constrains: Feature_Sewerage_MaintenanceHole.Use

Allowed uses for a Maintenance Hole

Restriction of: [String_32]

Enumeration:	Overflow	*Overflow Maintenance Hole*
Enumeration:	Blank End	*Blank End*
Enumeration:	Pump Station	*Pit housing a sewer pump station*
Enumeration:	Valve Pit	*Access Pit for a Sewer Valve*
Enumeration:	Grit Collector MH	*Grit Collector Maintenance Hole*
Enumeration:	Outlet	*Outlet*
Enumeration:	Rising Main Discharge MH	*Rising Main Discharge Maintenance Hole*
Enumeration:	Vacuum Sewerage Pump Station	*Vacuum Sewerage Pump Station*
Enumeration:	Vacuum Sewerage MH	*Vacuum Sewer Maintenance Hole*
Enumeration:	Vacuum Lift	*Vacuum Lift*
Enumeration:	Storage Tank	*Storage Tank*
Enumeration:	Maintenance Hole	*Maintenance Hole*
Enumeration:	Maintenance Shaft	*Maintenance Shaft*

Continue to the next section 10.1.3 ADAC Geometry Element or return to 10.1 ADAC.XML Overview or 10 ADAC .

10.1.3 ADAC Geometry Element

Like everything in ADAC, the ADAC **Geometry** element is fully defined, but it is **different** for each of the **ADAC Assets**.

An **ADAC Geometry** can only be one of:

1. a **Point**. That is, an (x,y,z) coordinate.

   For example, ADAC Assets with a Point Geometry include Sewer/MaintenanceHoles/MaintenanceHole, WaterSupply/Hydrants/Hydrant, OpenSpace/Barbecues/Barbecue.

2. a **Polyline**. That is, a string made up of (x,y,z) points with straights and/or arcs between them (segments).

   How many segments can be in the string, and whether arcs are allowed, can be different for each ADAC Asset with a Polyline Geometry. This can also vary with ADAC versions.

   For example, in ADAC 4.0, Sewerage/PipesNonPressure/PipeNonPressure can only have one segment, and that segment must be a straight.

   In ADAC 4.1, Transport/PipesNonPressure/PipeNonPressure can have any number of segments, and they can be straights or arcs.

3. a **Polygon**. That is, a closed polyline.

   Whether arcs are allowed as segments of the polygon is specified for each ADAC Asset with a Polygon Geometry. This can also vary with ADAC versions.

   ADAC Assets with a **Polygon Geometry** include *Transport/PavementAreas/Pavement* and *OpenSpace/LandscapeAreas/LandscapeArea.* Both of these can have any number of segments, and the segments can be straights or arcs.

If in an ADAC XML file, the Geometry written out for an ADAC Asset does not match the definition of the Geometry for that ADAC Asset, then it is breaking the rules and is an **invalid XML file with respect to the ADAC Schema**.

Continue to the next section 10.1.4 Guidelines from an Authority Requesting ADAC.XML or return to 10.1 ADAC.XML Overview or 10 ADAC .

10.1.4 Guidelines from an Authority Requesting ADAC.XML

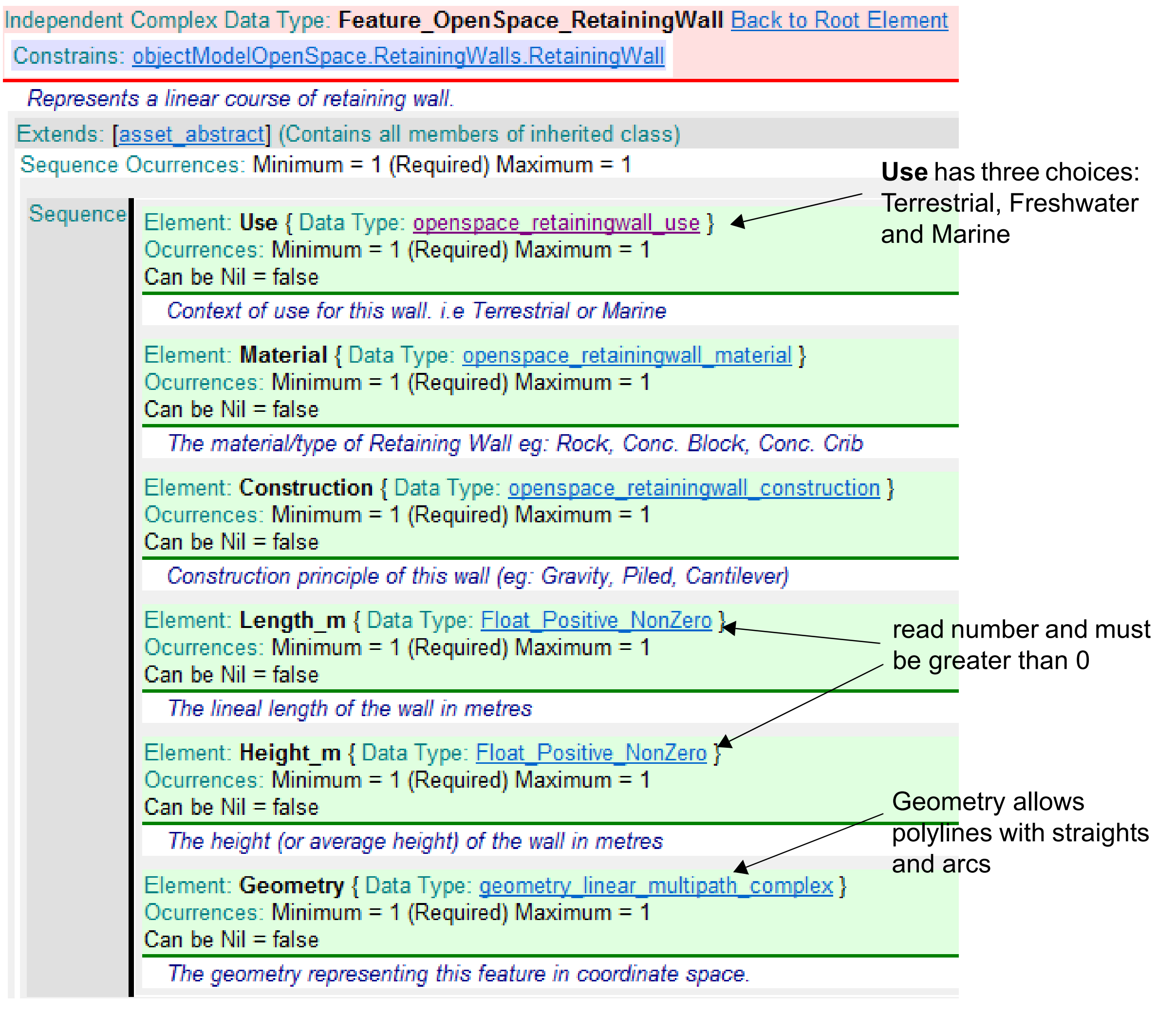
Although ADAC XML has **sixty-eight** Assets, which extends to hundreds of different objects once **Use** and **Type** are considered, not every ADAC Asset is required by every Authority.

Even when an ADAC Asset is required by an Authority, not every element within that ADAC Asset may be wanted by that Authority.

Also for any ADAC Assets that are required for a particular Authority, what exactly does that **Geometry** represent?

For example, in the ADAC version 4.1.0 Schema, an OpenSpace/RetainingWalls/RetainingWall is defined as

"Represents a continuous retaining wall feature. Includes terrestrial, freshwater and marine revetment walls."



Independent Complex Data Type: **Feature_OpenSpace_RetainingWall** Back to Root Element
Constrains: objectModelOpenSpace.RetainingWalls.RetainingWall

Represents a linear course of retaining wall.

Extends: [asset_abstract] (Contains all members of inherited class)
Sequence Ocurrences: Minimum = 1 (Required) Maximum = 1

Sequence

Element: **Use** { Data Type: openspace_retainingwall_use }
Ocurrences: Minimum = 1 (Required) Maximum = 1
Can be Nil = false
Context of use for this wall. i.e Terrestrial or Marine

Element: **Material** { Data Type: openspace_retainingwall_material }
Ocurrences: Minimum = 1 (Required) Maximum = 1
Can be Nil = false
The material/type of Retaining Wall eg: Rock, Conc. Block, Conc. Crib

Element: **Construction** { Data Type: openspace_retainingwall_construction }
Ocurrences: Minimum = 1 (Required) Maximum = 1
Can be Nil = false
Construction principle of this wall (eg: Gravity, Piled, Cantilever)

Element: **Length_m** { Data Type: Float_Positive_NonZero }
Ocurrences: Minimum = 1 (Required) Maximum = 1
Can be Nil = false
The lineal length of the wall in metres

Element: **Height_m** { Data Type: Float_Positive_NonZero }
Ocurrences: Minimum = 1 (Required) Maximum = 1
Can be Nil = false
The height (or average height) of the wall in metres

Element: **Geometry** { Data Type: geometry_linear_multipath_complex }
Ocurrences: Minimum = 1 (Required) Maximum = 1
Can be Nil = false
The geometry representing this feature in coordinate space.

And so the Geometry is a polyline with straights and arc segments.

But for, say, a *Terrestrial* RetainingWall, are the (x,y,z) coordinates of the Geometry the top of the retaining wall, the bottom of the retaining wall, or it doesn't matter?

So when asked to provide an ADAC XML file to an Authority, you need to ascertain from the Authority the following:

1. What ADAC Version is required. It will be either 4.0 or 4.1.
2. What ADAC Assets are required.

For example, Unity Water in Queensland is a Water Authority using ADAC 4.1 but they only require the ADAC Assets in *WaterSupply*.

3. What elements inside the required ADAC Assets are required.
4. What the geometry represents for the required ADAC Asset.

For example

StormWater

From the ADAC.XML Guidelines from Bundaberg, Gladstone and Rockhampton Regional Councils

EndStructure

Asset Capture: Simple point feature representing the top of the headwall.

Spatial Relationship: Headwall "floats" adjacent to the end of a StormWater pipe feature.

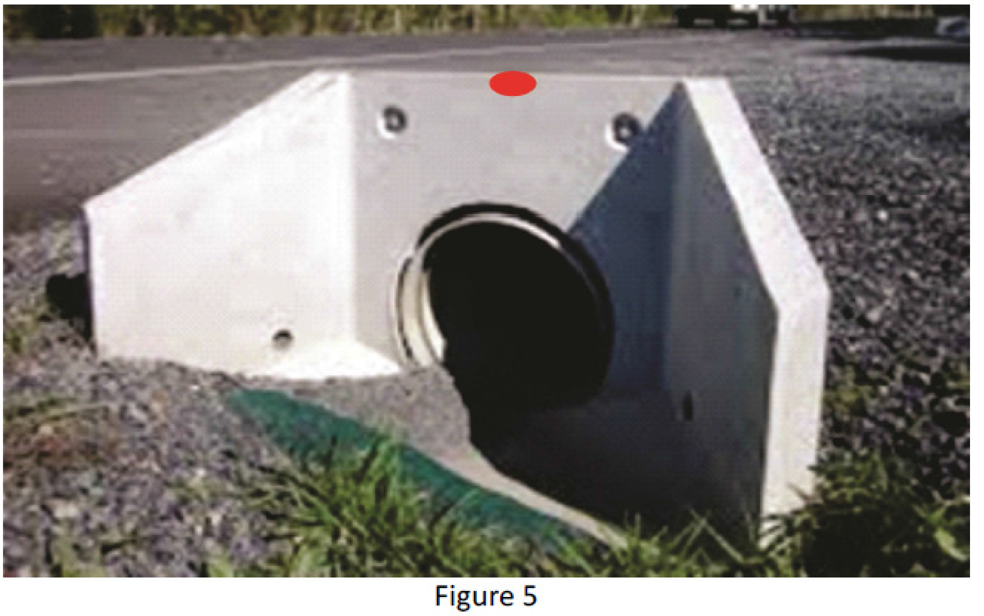


Figure 5

Transport

From the ADAC.XML Guidelines from Moreton Bay Regional Council

- **Roadways, including seals and pavement** to be captured from "Nominal kerb line to Nominal kerb line" as a closed polyline as per "red-dashed" example pictured in figures 1 & 2 below. Note: Separate polygons will be required at changes in pavement and/or surfacing.
- **Kerb line** is captured on the nominal kerb line (invert of kerb and channel, face of kerb only) as shown by "yellow-dashed" line shown in figures 1 and 2 shown below.
- **Sub-soil drains**, where installed, are to be captured at kerb/seal junction as per the "blue-dashed" examples shown in figures 1 and 2.
- **Road islands** are captured as closed polyline/object from back of kerb. Individual sub-sections of traffic islands to be identified by different material types (i.e. paving, concrete, grassed) as per "green-dashed" line in figure 1.

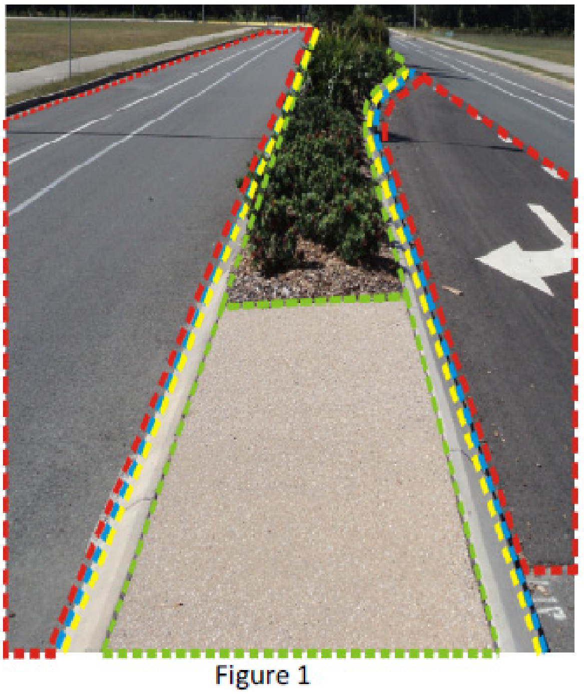


Figure 1

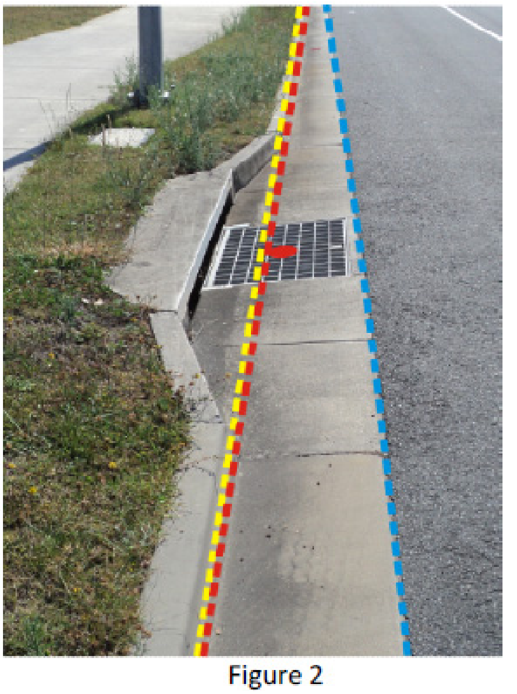


Figure 2

Return to 10.1 ADAC.XML Overview or 10 ADAC .

10.2 12d Approach to ADAC XML

Creating an ADAC XML file could be approached as a spreadsheet type exercise where you type the values into a document, but this is little more than a manual data collection exercise, performed (usually as an extra expense) after the design or as-built surveys are completed.

This is **NOT** the ***12d Model*** approach.

The guiding principles to the ***12d Model*** approach are:

1. ADAC Data Fidelity

   The ADAC Schema needs to be completely reflected inside ***12d Model*** so there is no ADAC data loss when writing and reading ADAC XML files.

2. 3D Engineering Data

   All data is held as 3D Engineering data.

3. Round Tripping of Data

   12d Model must be able to write out and read in ADAC data.

   That is

   WRITE OUT an ADAC XML file from data in ***12d Model***

   and

   READ IN an ADAC XML file to ***12d Model***.

4. Data Reuse

   Data reuse must be maximised. That is, wherever possible, any required ADAC data that is already part of the ***12d Model*** data should NOT have to be re-entered for the ADAC XML.

   So manual editing is to be minimised.

5. Integrated Work Flow

   The creation/collection of the ADAC data must be **integrated** into the normal ***12d Model*** design-construct-as built workflow so that there is minimal extra work involved in producing ADAC XML files.

Continue to the next section 10.3 12d ADAC Workflow or return to 10 ADAC .

10.3 12d ADAC Workflow

In ***12d Model***, the **source** of the data to write out to an ADAC XML is usually from either a **design** or a **survey**.

1. For **Design** - the design has been created in ***12d Model*** and it is some of the data from the Design that is to be written out as an ADAC XML file.
2. For **Survey** - the assets have been constructed and surveyed, and the survey data has been read into ***12d Model***. And it is some of the survey data that is to be written out as an ADAC XML file.

Although the type of data from a Design or a Survey can be totally different, the approach and steps inside ***12d Model*** for producing the ADAC XML file are very similar.

Step 0.Before Doing any ADAC Projects - this is only done once, by one or two people

Before doing any 12d-ADAC processing, the first thing to do is set up the Company procedures that everyone will use for ADAC Design or ADAC Survey project. This is usually done once, by one or two people, **before** a company starts doing ADAC work with ***12d Model***.

It involves:

(a) deciding what **Data Preparation** is needed for your company's survey or design before it is ready to produce an ADAC XML file.

(b) setting up the Company **User ADAC** menu.

(c) to allow for fast **bulk** processing, set up an ADAC ***12d Map File*** **for Design** and/or an ADAC ***12d Map File*** **for Survey** to assign the ADAC Asset type to a string from design or survey.

This is not essential and can be done, or added to, as you get more ADAC experience. But using an ADAC Map File is where the big benefits come from on most projects.

For information on each of these activities, see 10.6 Setting Up for ADAC.

Step 1.Setting Up a New ADAC Project - this is done once for each new ADAC project

For a new ***12d Model*** project, the first step is to set it up as either an ADAC Design or an ADAC Survey project.

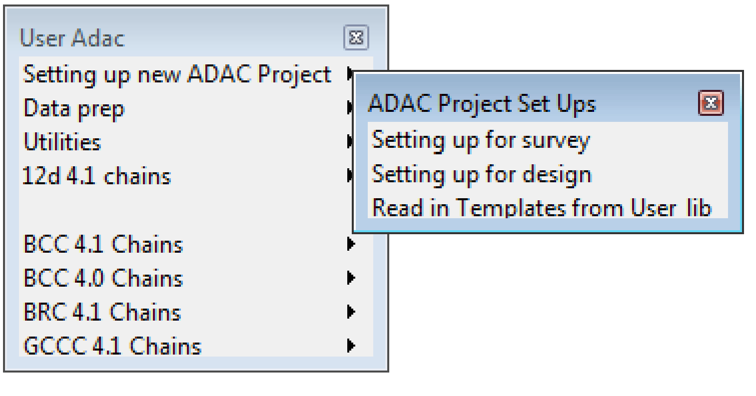
This is done **once for a new project** and only involves clicking a menu item:

File I/O =>ADAC =>User =>Setting up for survey - for surveys

For more information on what this option does, see 10.4.9.1.1 Setting Up for Survey.

File I/O =>ADAC =>User =>Setting up for design - for designers.

For more information on what this option does, see 10.4.9.1.2 Setting Up for Design.



Step 2.Creating the ADAC Header Data - only done once for each ADAC project.

The *ADAC Header Data* is only entered once for a project. This requires only a small amount of information such as the ADAC project name, the Owner and Receiver, the coordinate system used and (optionally) the Surveyor and the Engineer. See 10.4.1 Create Header .

If a *Header template* is used then most of the Header Data is automatically loaded.

Step 3.Preparing the Data to go to ADAC - done as required in each ADAC project

The design or survey data may need some preparation before it can go out to an ADAC XML file.

There is a variety of tools provided in ***12d Model*** ranging from setting Lot numbers, Plan Numbers and Areas, to splitting up a road edge into separate kerb types as required by ADAC.

See 10.4.9.2 Data Prep

Step 4.For Some Items, Assigning the ADAC Asset Type by Hand

Some data may be selected by hand to assign its ADAC Asset type using the **Create ADAC Asset** option. See 10.4.2 Create ADAC Asset

As your ***ADAC Map Files*** are created and extending, less data will need to be selected by hand. Instead the ADAC Asset type it will be automatically assigned.

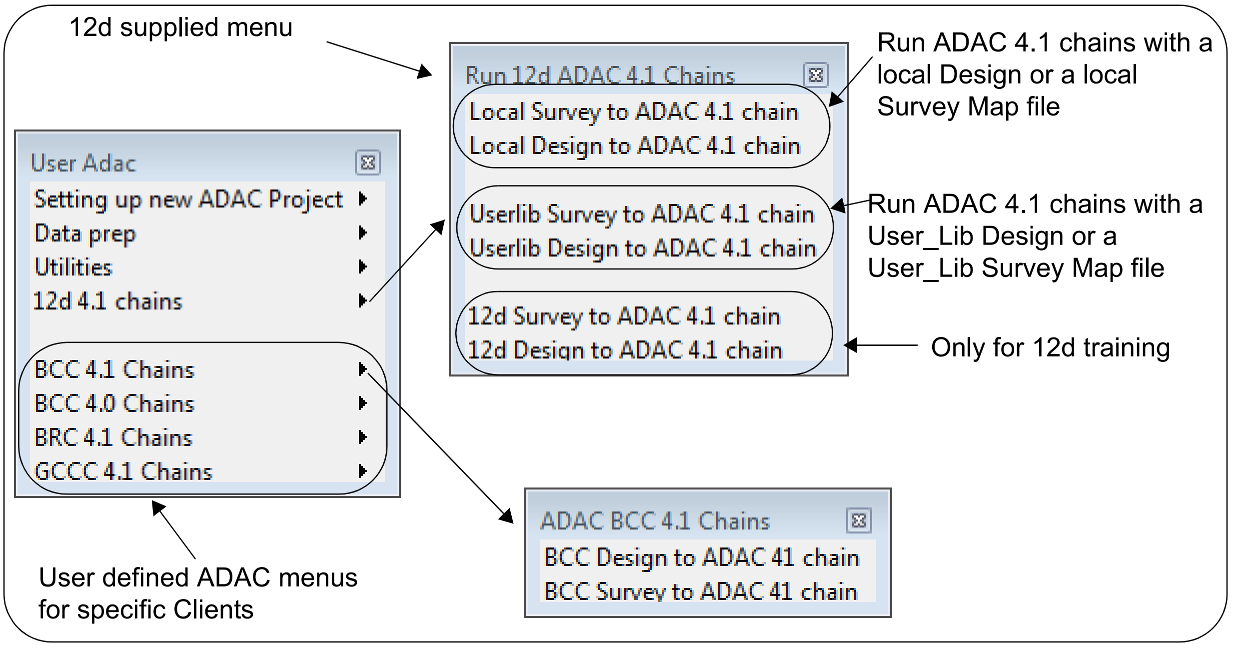
Step 5.Running a Design or Survey Chain - run regularly in an ADAC project

All the hard work for ADAC is done by either running an ADAC Design Chain or an ADAC Survey Chain. And to do that only involves clicking a menu item such:

File I/O =>ADAC =>User =>Appropriate Client Survey chain - for surveys

File I/O =>ADAC =>User =>Appropriate Client Design chain - for designers.

For more information on these chains, see 10.5 ADAC Design and Survey Chains .



In summary, what the *ADAC Design* or *ADAC Survey* chain does is:

(a) Uses the ADAC Map File to Assign the ADAC Asset Type

Any data going out to ADAC must be one of the ADAC Assets.

So either the ADAC Asset type has been assigned by hand, or the assignment is automatically done using an ***ADAC Map File*** that has already been set up.

(b) Sets ADAC values from the 12d String Geometry and User Attributes.

Much of the required ADAC data is already contained within the 12d strings or in User Attributes.

For example, if the Drainage or Sewer was designed with ***12d Model***, then much of the ADAC data can come directly from the drainage and sewer strings. For example, maintenance holes and chambers, depths, and pipe dimensions come directly from the drainage string, and any required 2D and 3D lengths are calculated from the drainage string.

User attributes picked up in the field by the surveyors, or added in the **Data Prep** step, are also loaded into the specified ADAC attributes.

(c) Creates the ADAC Geometry.

For each ADAC Asset, the ADAC Geometry is automatically generated from the 12d string geometry.

For example, if the ADAC Asset has Polyline Geometry, the (x,y,z) coordinates for each vertex of the string are loaded into the ADAC Geometry.

Step 6.Validating the Data, Generating Reports, Generating the ADAC Geometry - done as required

A Validation option checks the data against the ADAC Schema, and any errors are flagged and are easily found using ***12d Model's*** Intelligent logs lines. This is a simple process and is done by running only one panel. See 10.4.4 Validate .

A report of the ADAC Data can be produced in HTML, Text or PDF format. This could be for your records or used as another checking tool. This is a simple process and is done by running only one panel. See 10.4.5 Report

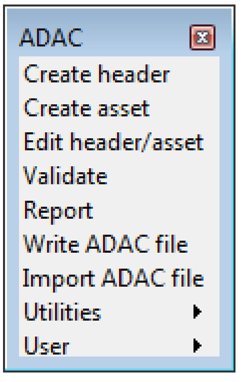
Step 7.Writing the Data out to ADAC XML - done as required

This is a simple process and is done by running just the one panel. See 10.4.6 Write ADAC XML File

Continue to the next section 10.4 12d ADAC Menu or return to 10 ADAC .

10.4 12d ADAC Menu

Position of menu: **File I/O =>ADAC**



See

[10.4.1 Create Header](#)

[10.4.2 Create ADAC Asset](#)

[10.4.3 Edit ADAC Header/Asset](#)

[10.4.4 Validate](#)

[10.4.5 Report](#)

[10.4.6 Write ADAC XML File](#)

[10.4.7 Import ADAC XML File](#)

[10.4.8 ADAC Utilities](#)

[10.4.9 User ADAC](#)

10.4.1 Create Header

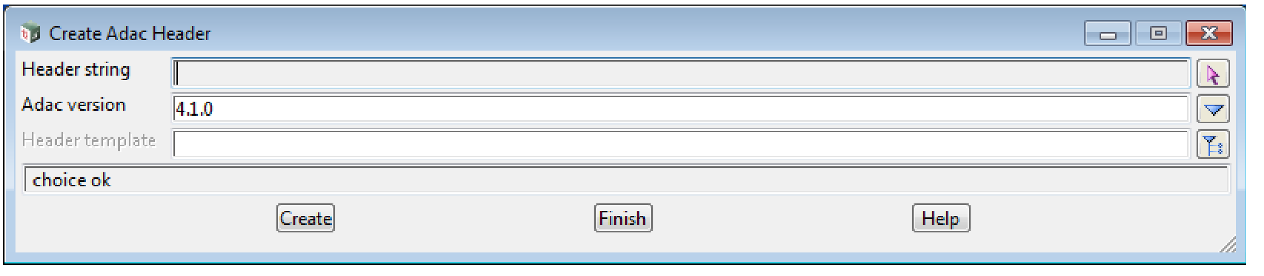
Position of option on menu: **File I/O =>ADAC =>Create header**

To produce an ADAC XML file, ***12d Model*** needs exactly one string with the ADAC Header data, and zero or more strings representing ADAC Asset data.

The **Create header** option takes a one vertex string, and creates as string attributes, the attributes needed to make it an ADAC Header. A user selected ADAC Header Template can be used to set the values for any of the ADAC attributes in the ADAC Header.

The ***ADAC Header Editor*** is then brought up so any of the created ADAC Header attributes can be edited.

Selecting **Create header** brings up the **Create ADAC Header** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Header string	string select		

select a one vertex string to set up with the ADAC Header Data attributes.

ADAC version	choice box	from env.4d	4.0.0, 4.1.0

the version of ADAC to use when creating the ADAC Header data.

The default value is set by the environment variable ADAC_VERSION_4D which is set in the **Edit Environment Variables** *panel at* **External Apps >ADAC**. *See* 7.6.3 env.4d.

Header template	ADAC Header box		available ADAC Header templates

the name of an ADAC Header template that is used to fill in some of the ADAC Header information for the Header string.

Create	button		

clicking **Create** *creates an ADAC Header structure as 12d ADAC attributes for the string, and if there is a* **Header template**, *it loads the string's ADAC attributes with values from the Header Template.*

The 10.4.2 Create ADAC Asset *panel is then brought up so that the ADAC Header data can be create/edited.*

Continue to the next section 10.4.2 Create ADAC Asset or return to 10.4 12d ADAC Menu or 10 ADAC .

10.4.2 Create ADAC Asset

Position of option on menu: **File I/O =>ADAC =>Create Asset**

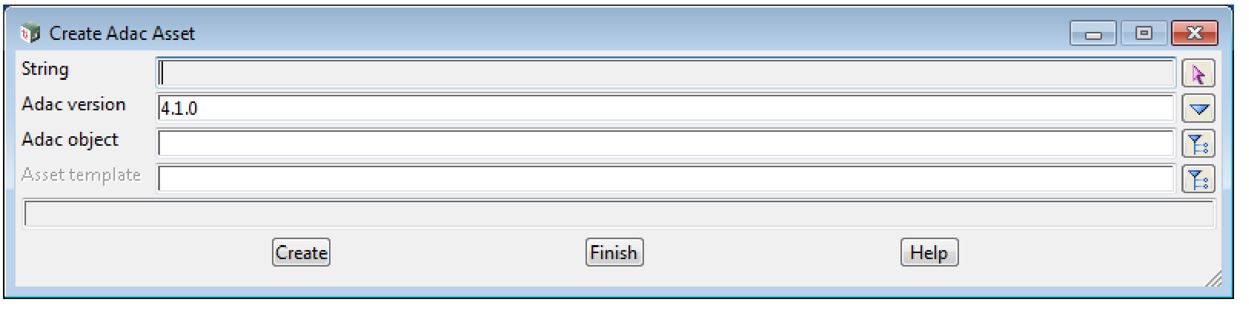
To produce an ADAC XML file, ***12d Model*** needs one string with the ADAC Header data, and zero or more strings with ADAC Asset data.

The **Create ADAC Asset** option takes a string with the appropriate geometry and creates as string attributes the ADAC Asset attributes for the user selected ADAC Asset type. This is referred to as **assigning the ADAC Asset type**, or **assigning the ADAC type**, to the string.

Also, a user selected ADAC Asset Template can be used to set the values for any of the ADAC attributes.

After creating the appropriate ADAC Asset attributes, the ADAC Asset Editor for the particular ADAC Asset is then brought up so any of the created ADAC Asset attributes can be edited.

Selecting **Create asset** brings up the **Create ADAC Asset** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
String	string select		

select a string that is to be set up with the ADAC Asset.

ADAC version	choice box	from env.4d	4.0.0, 4.1.0

the version of ADAC to use when creating the ADAC Header data.

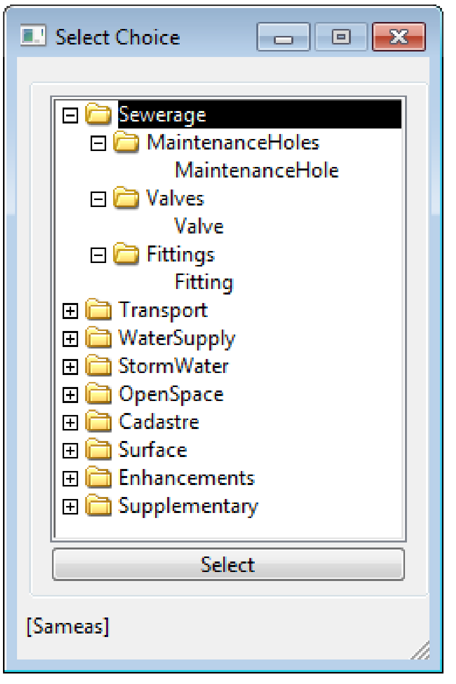
The default value is set by the environment variable ADAC_VERSION_4D which is set in the env.4d editor at **External Apps >ADAC**.

ADAC Asset	choice box	depends on the string geometry

If the geometry of the string selected is a:

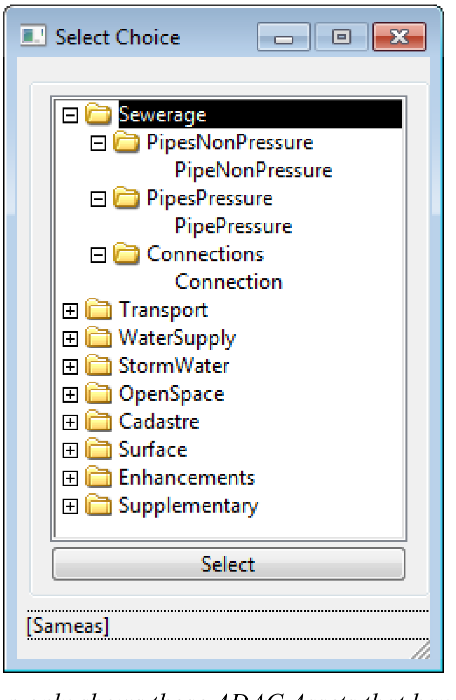
Point *(a one vertex string), then the pop-up only shows those ADAC Assets that have a Point Geometry.*

For example, for ADAC 4.1, the only ADAC Assets in Sewerage with a Point Geometry are Sewerage>MaintenanceHoles>MaintenanceHole, Sewerage>Valves>Valve and Sewerage>Fittings>Fitting



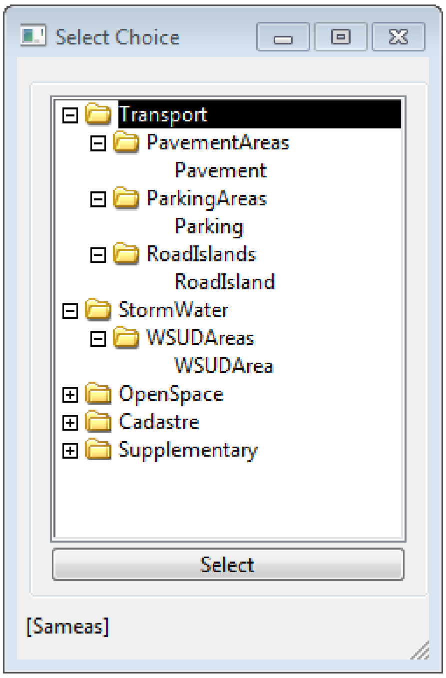
Polyline *then the pop-up only shows those ADAC Assets that have Polyline Geometry.*

For example, for ADAC 4.1, the only ADAC Assets in Sewerage with a Polyline Geometry are Sewerage>PipesNonPressure>PipeNonPressure, Sewerage>PipesPressure>PipePressure and Sewerage>Connections>Connection



Polygon *then the pop-up only shows those ADAC Assets that have Polygon Geometry.*

For example, for ADAC 4.1, there are NO ADAC Assets in Sewerage with a Polygon Geometry. In Transport there are Transport>PavementAreas>Pavement, Transport>ParkingAreas>Parking and Transport>RoadIslands>RoadIsland



Asset template ADAC Asset box available ADAC Asset templates

the name of an ADAC Asset template that if selected, is used to fill in some of the ADAC Asset attributes for the string.

The pop-up looks at the **ADAC Asset** *and only displays the templates for that type of ADAC Asset and for the ADAC version.*

Create button

clicking **Create** *creates an ADAC Asset structure as 12d ADAC attributes and if there is an* **Asset template,** *it loads the string's attributes with values from the Asset Template.*

The 10.4.3 Edit ADAC Header/Asset *panel is then brought up so that the ADAC Header data can be create/edited.*

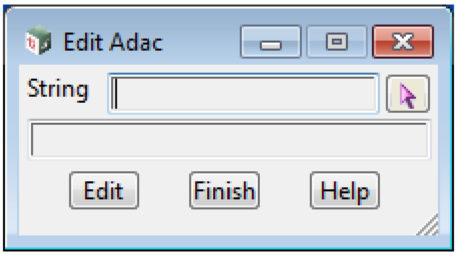
Continue to the next section 10.4.3 Edit ADAC Header/Asset or return to 10.4 12d ADAC Menu or 10 ADAC .

10.4.3 Edit ADAC Header/Asset

Position of option on menu: **File I/O =>ADAC =>Edit header/asset**

The ***ADAC* Editor** edits a string and looks at its ADAC attributes and determines whether it is an ADAC ***Header*** string or an ADAC ***Asset*** string (and then which ***Asset* type**), or is not an ADAC string at all.

Selecting **Edit** brings up the **Edit ADAC** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
String	string select		

select a string to have its ADAC attributes edited.

Edit	button		

if the selected string is a ADAC Header string, then the 10.4.2 Create ADAC Asset *is brought up.*

If the selected string is an ADAC Asset string, the 10.4.3 Edit ADAC Header/Asset *panel is brought up for its ADAC Asset type.*

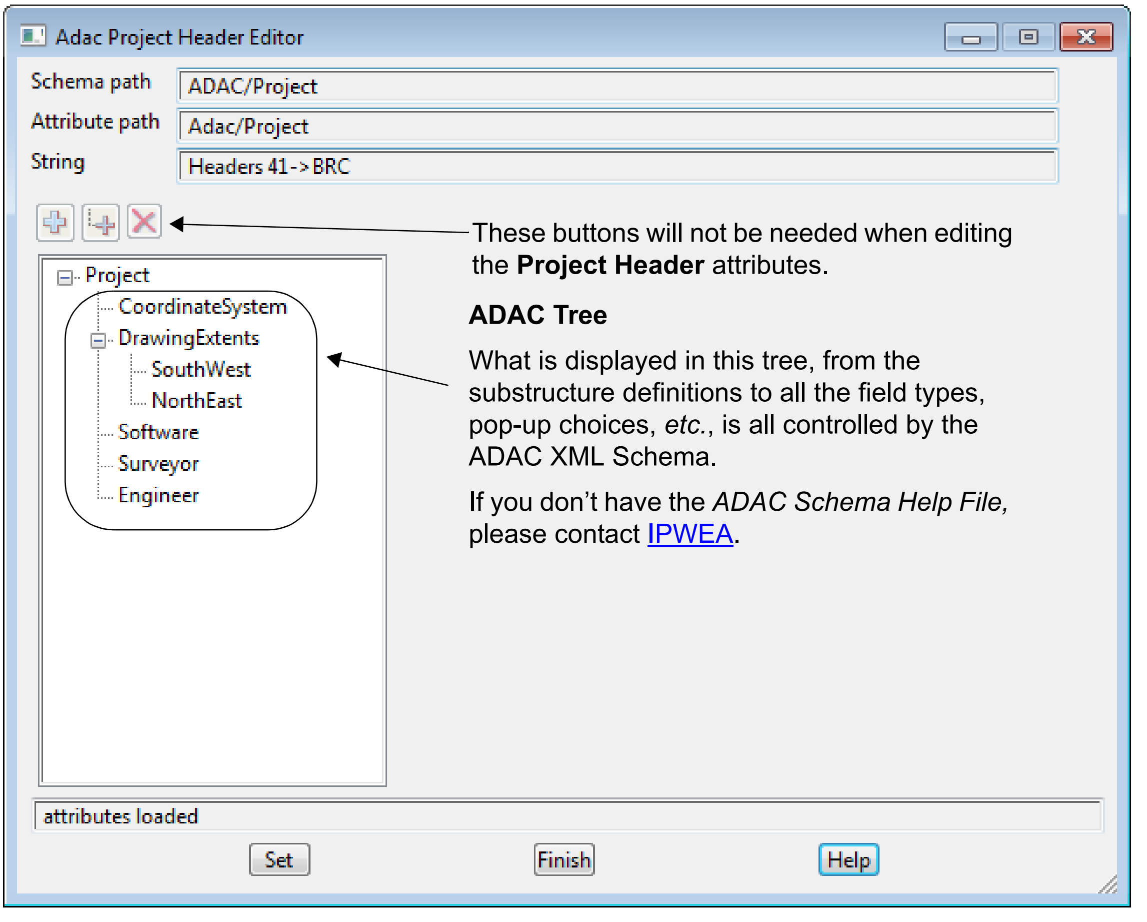
If the selected string is not an ADAC Header or an ADAC Asset string, then an error message is written to the panel message area.

Continue to the next section 10.4.3.1 ADAC Header Editor or return to 10.4.3 Edit ADAC Header/Asset, 10.4 12d ADAC Menu or 10 ADAC.

10.4.3.1 ADAC Header Editor

Position of option on menu: File I/O =>ADAC =>Edit

When a string is selected with the *ADAC Editor* and the string has **ADAC Header Data** attributes, then the **ADAC Project Header Editor** panel is brought up with any ADAC Header Data already with the string loaded into the Editor.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Schema path	output only		

*the path to the ADAC Schema that is used to generate all the items under **Project**.*

Attribute path output only

the path in the ADAC Schema to the ADAC element being edited. In this case it is the ADAC Project element.

String output only

displays the string that has the ADAC Header Data attributes that are being displayed and edited. This will already be set when the panel is brought up by the ADAC Editor or the ADAC Create Header panel.

ADAC Tree

What is displayed in this tree, from the substructure definitions to all the field types, pop-up choices, etc., is all controlled by the ADAC XML Schema.

Even what is shown in these images may vary with the ADAC Version.

If you don't have the ADAC Schema Help File for the ADAC Version you are creating, please contact IPWEA.

*The **Drawing extents** and **Software** do not have to be filled out because they are updated by **12d Model**.*

*For information about all the other ADAC elements in the **ADAC Tree**, please see the ADAC XML Schema Help file.*

Drawing extents these do not have to be filled in

these fields do not have to be filled in. The drawing extents will be calculated and updated when the data is written out to an ADAC XML file.

Software these do not have to be filled in

these fields do not have to be filled in. The **Software product** *and* **Version** *will be updated when the data is written out to an ADAC XML file.*

Buttons at Bottom

Set button

clicking **Set** *updates the ADAC Header attributes for the string being edited, using the values in this panel.*

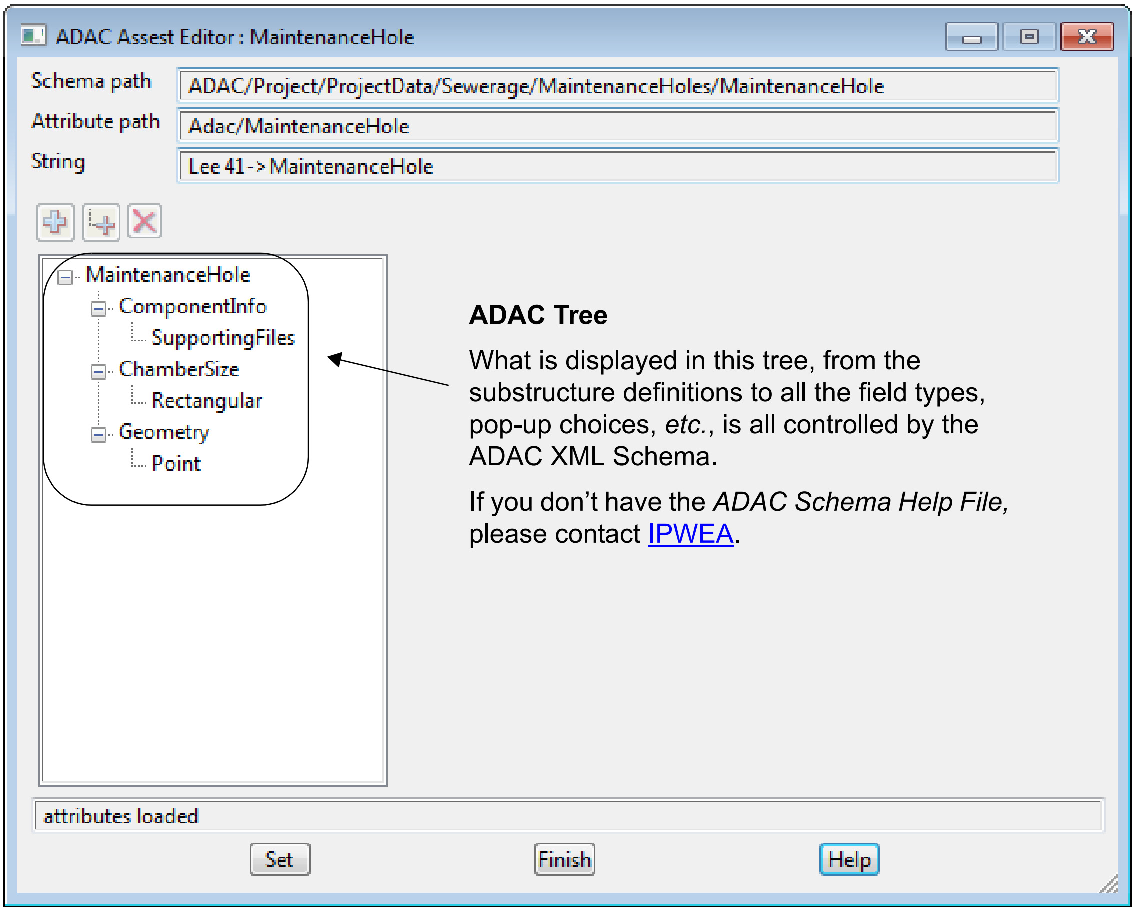
Continue to the next section 10.4.3.2 ADAC Asset Editor or return to 10.4.3 Edit ADAC Header/ Asset , 10.4 12d ADAC Menu or 10 ADAC .

10.4.3.2 ADAC Asset Editor

Position of option on menu: **File I/O =>ADAC =>Edit**

When a string is selected with the *ADAC Editor* and the string has **ADAC Asset Data** attributes, then the **ADAC Asset Editor** for that particular ADAC Asset is brought up, with any ADAC Asset Data already on the string, loaded into the Editor.

For example, for the ADAC Asset Sewerage>MaintenanceHoles>MaintenanceHole, the **Editor** panel is



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Schema path output only

the path to the ADAC Schema that is used to generate all the items for this ADAC Asset.

Attribute path output only

the path in the ADAC Schema to the ADAC element being edited. In this case it is the ADAC Sewerage MaintenanceHole element.

String output only

displays the string that has the ADAC Asset Data attributes that are being displayed and edited. This will already be set when the panel is brought up by the ADAC Editor or the ADAC Create Asset panel.

ADAC Tree

What is displayed in this tree, from the substructure definitions to all the field types, pop up choices, etc., is all controlled by the ADAC XML Schema.

Even what is shown in these images may vary with the ADAC Version.

If you don't have the ADAC Schema Help File for the ADAC Version you are creating, please contact IPWEA.

For information about all the ADAC elements in the ***ADAC Tree*** for an ***ADAC Asset****, please see the ADAC XML Schema Help file.*

Geometry this do not have to be filled in

the Geometry is generated by 12d Model from the strings geometry.

Buttons at Bottom

Set button

clicking **Set** *updates the ADAC Asset attributes for the string being edited, with the values in this panel.*

Continue to the next section 10.4.3.3 Common Features of ADAC Editors or return to 10.4.3 Edit ADAC Header/Asset , 10.4 12d ADAC Menu or 10 ADAC .

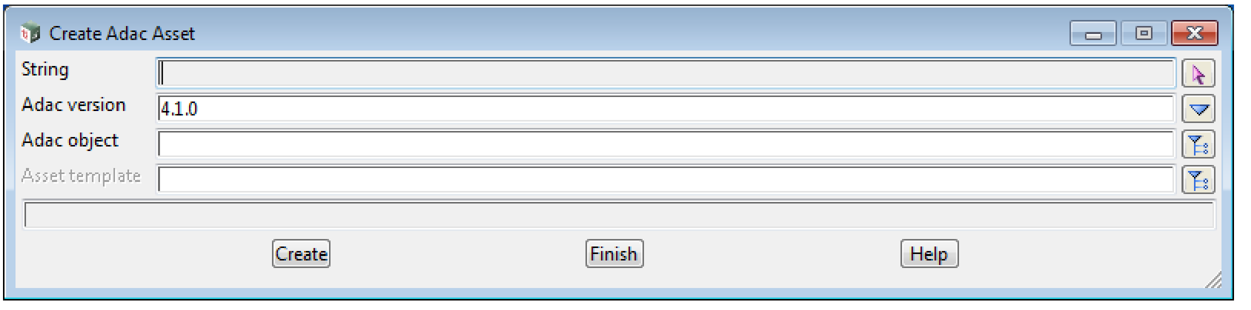
10.4.3.3 Common Features of ADAC Editors

The *ADAC Schema* is fully described by the *ADAC XSD* (**X**ML **S**chema **D**efinition), which formally defines the structure and all the elements inside an *ADAC XML* file.

And the ***ADAC Header Editor*** and the ***ADAC Asset Editors*** for each of the different *ADAC Assets*, are also fully defined by the *ADAC Schema XSD*.

What that means is that everything in the *ADAC Editors*, and how they behave, follows the *ADAC XSD*.

That starts with **Create ADAC Asset** panel:



where once a string is selected, the choice of ***ADAC Asset*** choices is totally determined by which *ADAC Assets* in the *ADAC XSD* have that geometry.

Once in the **Editors** (*ADAC Asset* or *ADAC Header*), every item in the *Editor* is again controlled by the *ADAC XSD*.

So to know why each items exists, and what it is, you need to refer to the *ADAC Schema*.

What we'll now do is look at how the Editor is structured, navigated and used.

See

10.4.3.3.1 Editor Tree, Nodes and Elements

10.4.3.3.2 Delete Icon

10.4.3.3.3 Add Sibling and Add Child Icons

10.4.3.3.4 Nillable Elements and Nodes

10.4.3.3.5 Nodes with Choices for Subnodes

10.4.3.3.1 Editor Tree, Nodes and Elements

When the ADAC **Editors** start up, they have a tree structure on the left hand side.

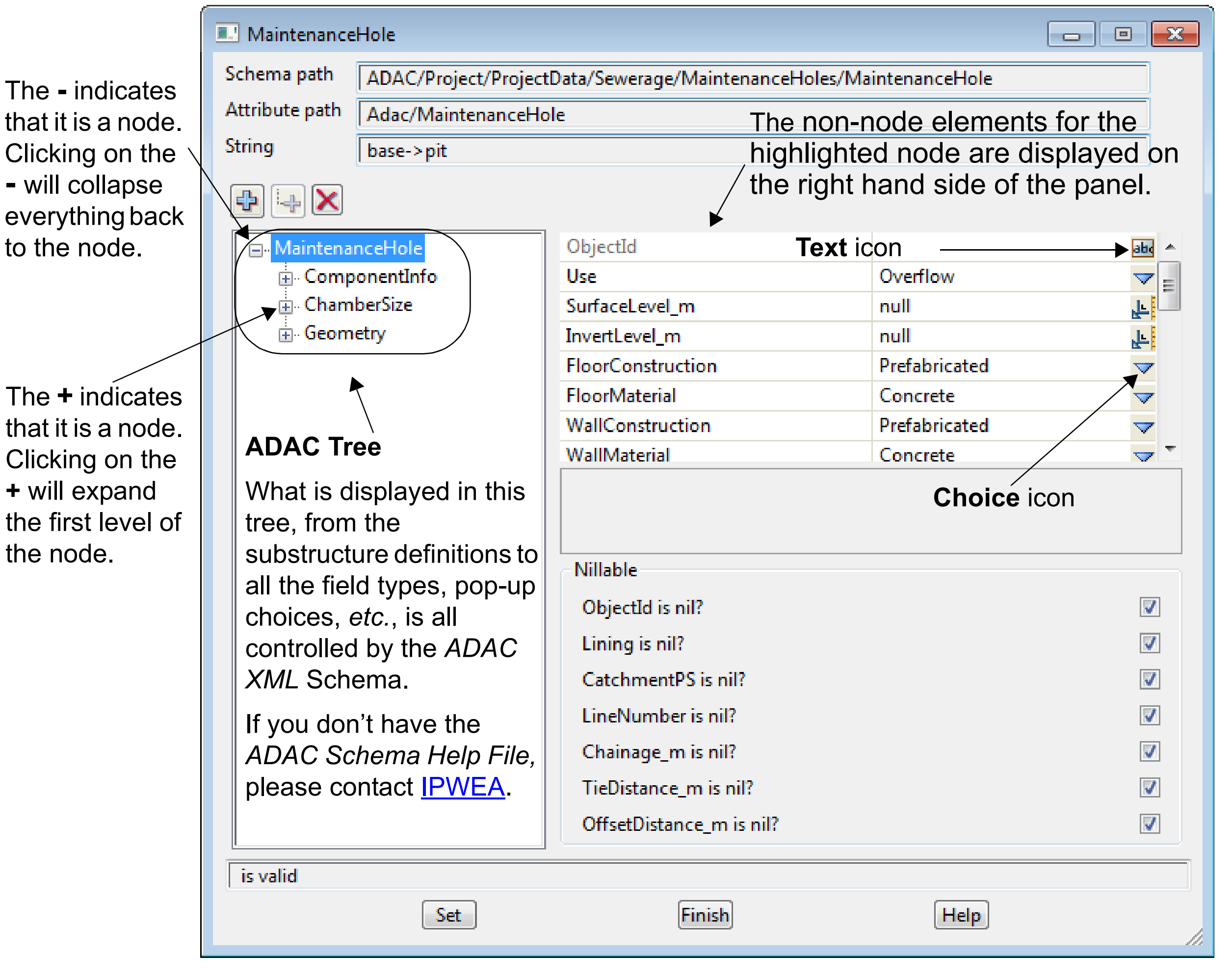
Each item in the tree are known as nodes and **nodes** contain ADAC elements or can have their own substructure of nodes and elements. A node with a substructure is identified by having either a **+** or a **-** to the left of it.

The **+** indicates that the node has a substructure that has not been expanded and by clicking on the **+**, the first level of the sub structure is displayed.

The **-** indicates that the node contains a substructure and it has been expanded in the tree. Clicking on the **-** will collapse the expansion back into just the node.

When a node is clicked on and hence highlighted, all the element that belong to the first level of the node that are not other nodes, are displayed on the right hand side of the panel.

It is the non-node elements displayed on the right that hold the data about the *ADAC Asset*.



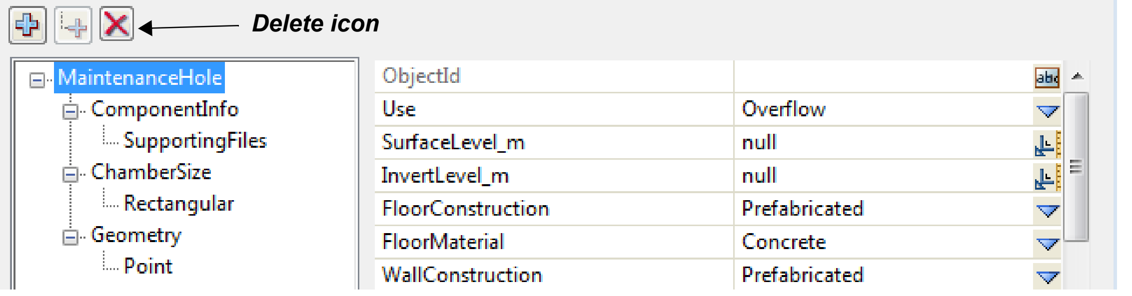
The elements on the right hand side of the panel mainly consist of choice, text, real, integer and date boxes and they are all defined in the ***ADAC XSD***. Where ever possible, the icons at the end of each field are the same as those used in ***12d Model*** panels.

Where it is a **Choice** box, the allowable choices (which are in the pop up) are all defined in the ***ADAC XSD*** where they are known as **Enumerations**.

Continue to the next section 10.4.3.3.2 Delete Icon or return to 10.4.3.3 Common Features of ADAC Editors , 10.4.3 Edit ADAC Header/Asset , 10.4 12d ADAC Menu or 10 ADAC .

10.4.3.3.2 Delete Icon

There is a **Delete** icon on the ADAC **Editors** but most of the time is disabled because there are only a few places in the *ADAC Schema* that data can be deleted.



If a node in the tree is selected and highlighted, and the **Delete** icon is enabled and then clicked, the node is deleted.

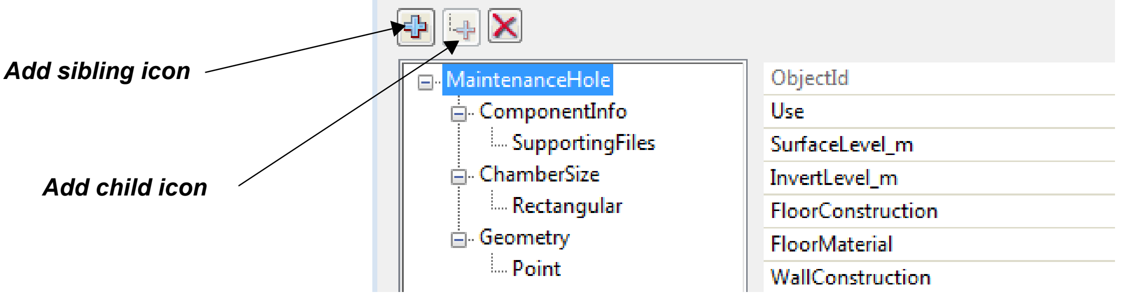
The **Delete** icon is enabled when the *ADAC Asset* itself has been selected as in the image above but deleting the *ADAC Asset* itself is rarely done.

The one time when the **Delete** is usable is when a node has choices for one of its subnodes (for example, ChamberSize). In that case to change the choice of subnode, the existing subnode needs to be deleted and then a new one created. replaced by a new one. This process is documented in the section 10.4.3.3.5 Nodes with Choices for Subnodes .

Continue to the next section 10.4.3.3.3 Add Sibling and Add Child Icons or return to 10.4.3.3 Common Features of ADAC Editors , 10.4.3 Edit ADAC Header/Asset , 10.4 12d ADAC Menu or 10 ADAC .

10.4.3.3.3 Add Sibling and Add Child Icons

There is are two Add icons - **Add** a sibling and **Add** a child.



The **Add a sibling** icon adds a new item at **the same level** (a sibling) as the highlighted node.

The **Add a child** icon adds a node as a **subnode** of the highlighted node.

As in the **Delete** icon case, most of the time the **Add a sibling** and **Add a child** icons are disabled because there are only a few places in the *ADAC Schema* that it makes sense to add a new sibling or child node.

The **Add a sibling** icon is enabled when the *ADAC Asset* itself has been selected as in the image above but adding a second *ADAC Asset* is not useful because a second or subsequent ADAC Asset on the one string will be ignored in the 12d-ADAC process.

One time when the **Add a child** is usable is when a node has choices for one of its subnodes (for example, ChamberSize). In that case to change the choice of subnode, the existing subnode needs to be deleted and then a new one created. replaced by a new one using **Add a child**. This process is documented in the section 10.4.3.3.5 Nodes with Choices for Subnodes.

Continue to the next section 10.4.3.3.4 Nillable Elements and Nodes or return to 10.4.3.3 Common Features of ADAC Editors, 10.4.3 Edit ADAC Header/Asset, 10.4 12d ADAC Menu or 10 ADAC.

10.4.3.3.4 Nillable Elements and Nodes

One concept used in the *ADAC XSD* that has not been expressed in the same way in other ***12d Model*** panels, is **nillable**.

In the ADAC XSD, if a *node* or *element* is **nillable** (i.e. has **nil** set to true in the XSD), then the node or element may or may not have a value. So it is optional.

If a *node* or *element* is **not nillable** (i.e. has **nil** set to **false** in the XSD) then the node or element is not optional and must have values.

See

10.4.3.3.4.1 Nillable Elements

10.4.3.3.4.2 Nillable Node

Or return to 10.4.3.3 Common Features of ADAC Editors , 10.4.3 Edit ADAC Header/Asset , 10.4 12d ADAC Menu or 10 ADAC .

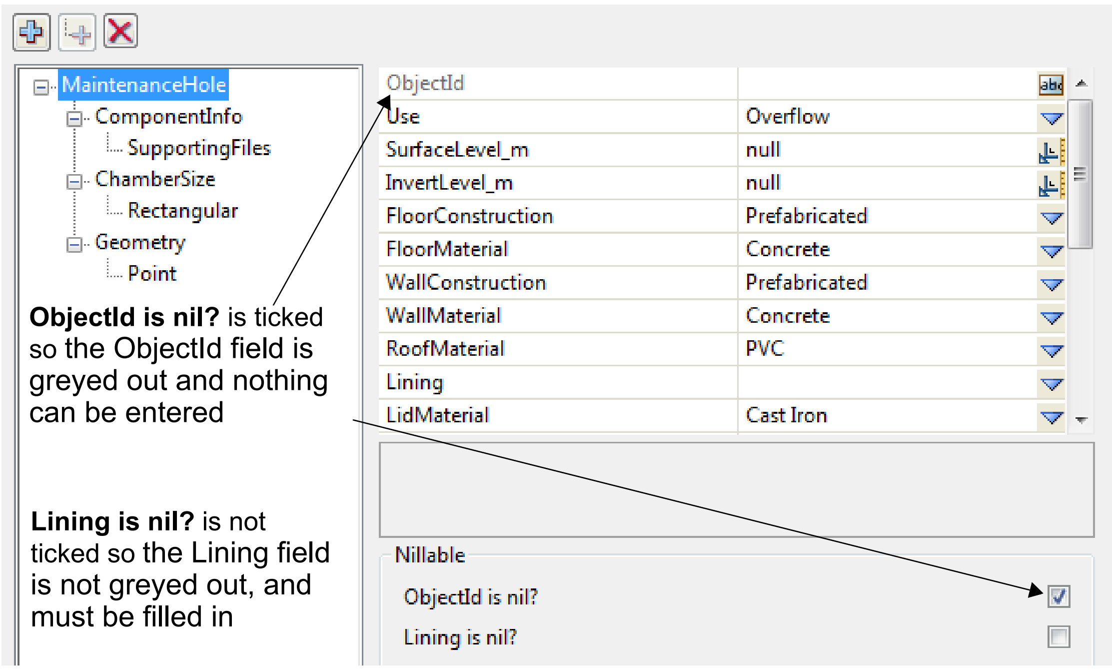
10.4.3.3.4.1 Nillable Elements

In the ADAC **Editor**, if an element **is nillable** (and hence optional) then it is listed on the right hand side of the panel under the title **Nillable** and with a tick box beside its name.

If the tick box is **ticked** then the item is not to be filled in and the panel field for it is greyed out so nothing can be entered.

If the tick box is **not ticked** then the item is to be filled in and the panel field is no longer greyed out. In this is the case a value must be entered.

All elements not mentioned in the Nillable area are **not nillable** and must always be filled in.



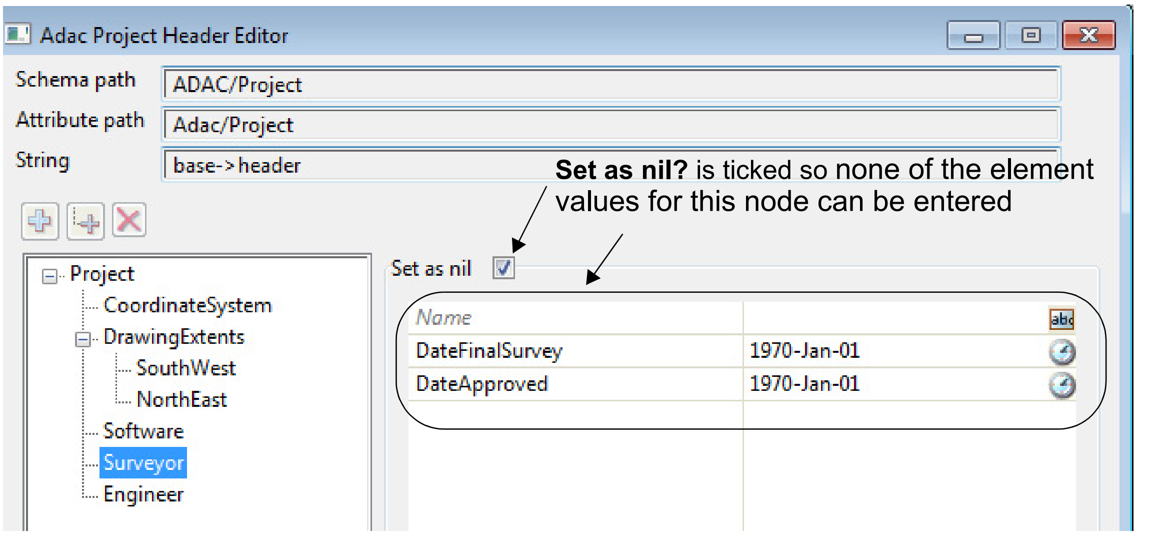
Continue to the next section 10.4.3.3.4.2 Nillable Node or return to 10.4.3.3.4 Nillable Elements and Nodes .

10.4.3.3.4.2 Nillable Node

In the ADAC **Editor**, if an node **is nillable** (and hence optional) then on top of the listing of the elements for the node is a tick box **Set as nil.**

If the tick box is **ticked** then none of the elements can be filled in.

If the tick box is **not ticked** then the element have the potential of being filled it. Whether they are to be filled in depends on if the individual element is nillable or not.

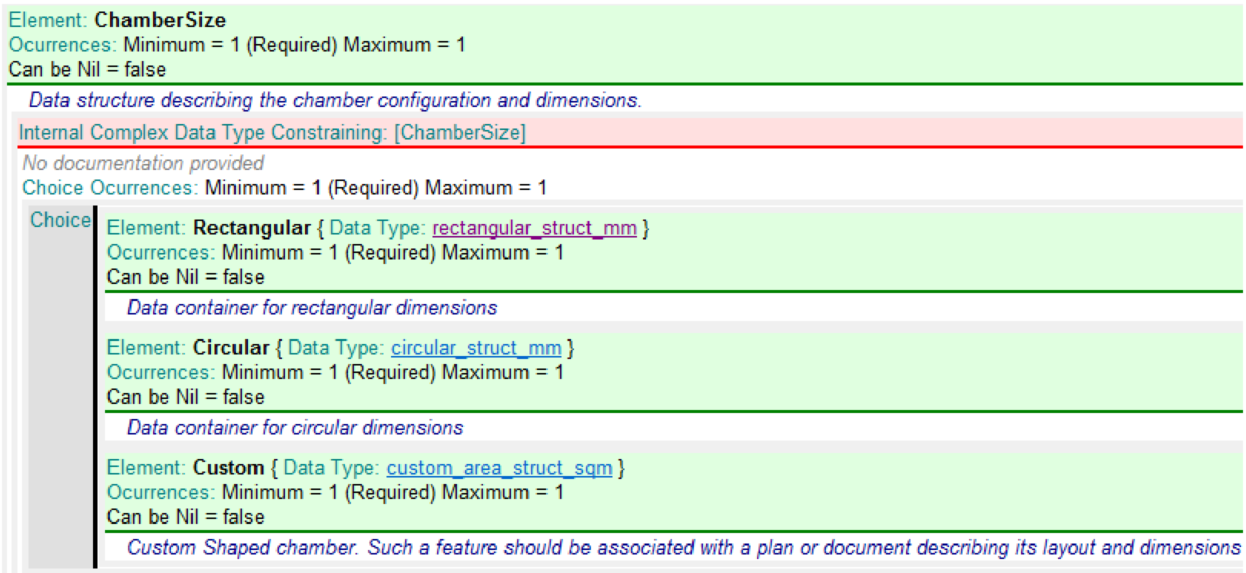


Return to 10.4.3.3.4 Nillable Elements and Nodes .

10.4.3.3.5 Nodes with Choices for Subnodes

In the *ADAC Schema*, there are some nodes that have choices for subnodes.

For example, ***Sewerage>MaintenanceHoles>MaintenanceHole>ChamberSize*** is not just a since value but is a choice of subnodes - *Rectangular*, *Circular* and *Custom* - which each contain one or more elements.



Element: **ChamberSize**
Ocurrences: Minimum = 1 (Required) Maximum = 1
Can be Nil = false

Data structure describing the chamber configuration and dimensions.

Internal Complex Data Type Constraining: [ChamberSize]

No documentation provided
Choice Ocurrences: Minimum = 1 (Required) Maximum = 1

Choice

Element: **Rectangular** { Data Type: rectangular_struct_mm }
Ocurrences: Minimum = 1 (Required) Maximum = 1
Can be Nil = false

Data container for rectangular dimensions

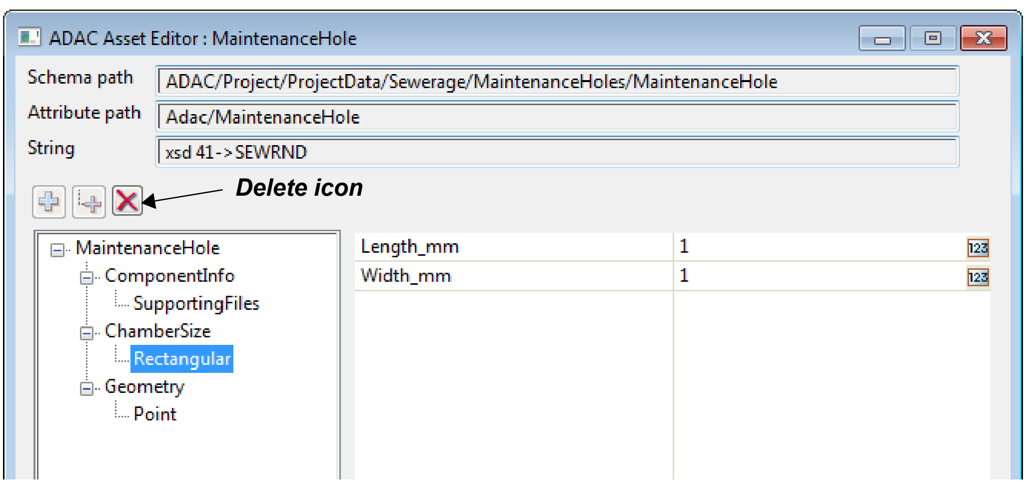
Element: **Circular** { Data Type: circular_struct_mm }
Ocurrences: Minimum = 1 (Required) Maximum = 1
Can be Nil = false

Data container for circular dimensions

Element: **Custom** { Data Type: custom_area_struct_sqm }
Ocurrences: Minimum = 1 (Required) Maximum = 1
Can be Nil = false

Custom Shaped chamber. Such a feature should be associated with a plan or document describing its layout and dimensions

-

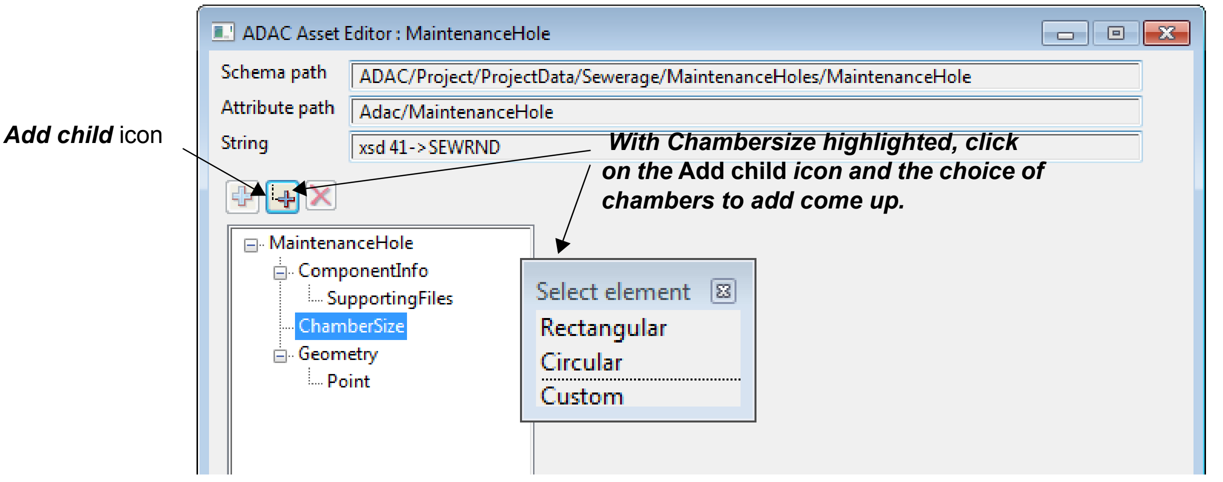


To change the subnode choice, you need to click on the **subnode** to highlight it and then click on the **Delete** icon.

To create a new subnode now that one no longer exists, click on the **node** to highlight it and then click on the **Add Child** icon. A **Select Element** panel will then appear with the available choices.

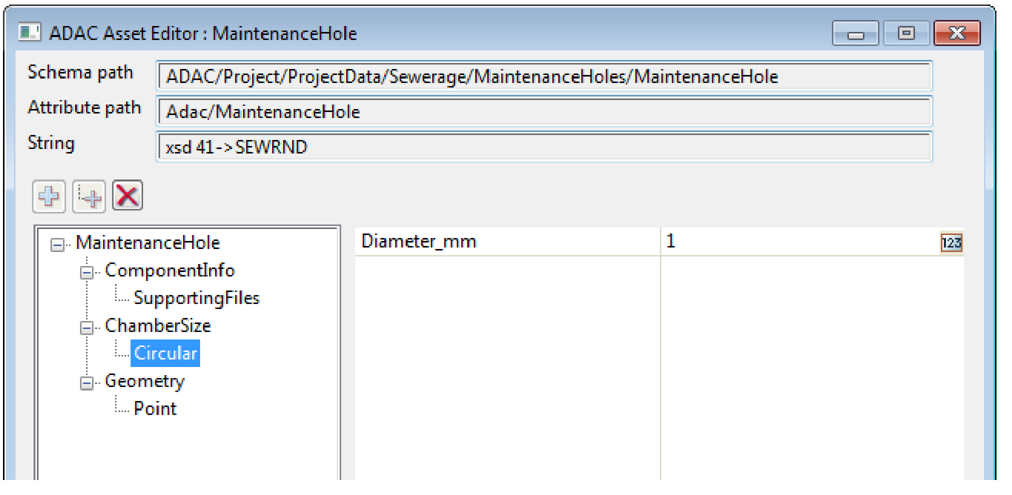
For example, for the ***Sewerage>MaintenanceHoles>MaintenanceHole>ChamberSize***, the choices are *Rectangular*, *Circular* and *Custom*.

-



Select the choice from the **Select Element** panel and the new **subnode** is created for the highlighted node.

For example, if *Circular* is chosen as the choice of subnode of *ChamberSize*



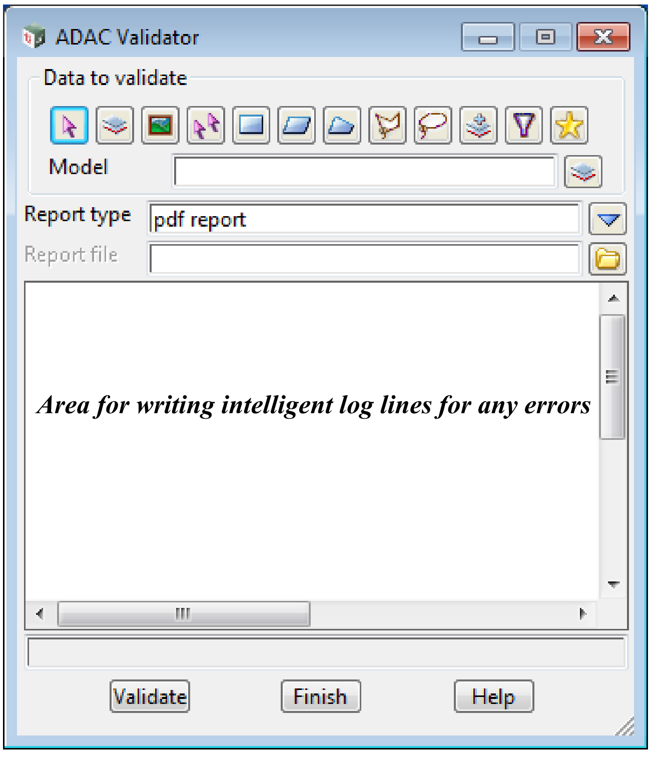
Return to 10.4.3.3 Common Features of ADAC Editors .

10.4.4 Validate

Position of option on menu: **File I/O =>ADAC =>Validate**

The **ADAC Validator** validates a *Data Source* of string against the ADAC XML Schema.

Selecting **Validate** brings up the **ADAC Validator** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source .

Data source	input	

source of data to be validated against the ADAC Schema.

Report type	choice box	ADAC pdf, ADAC html, original xml, custom

output format for the report. An XML file will be produced and then if required, converted to the selected report.

For information on setting up custom reports from the generated XML file using xslts, see 4.30 Setting Up XML Reports .

Report file	file box	

*if **not blank**, an XML file will be created and a report of this name, and of the type given by* **Report type** *will be generated from the XML file.*

*If **blank**, no report is created but errors are still written to the panel's message area.*

Validate	button	

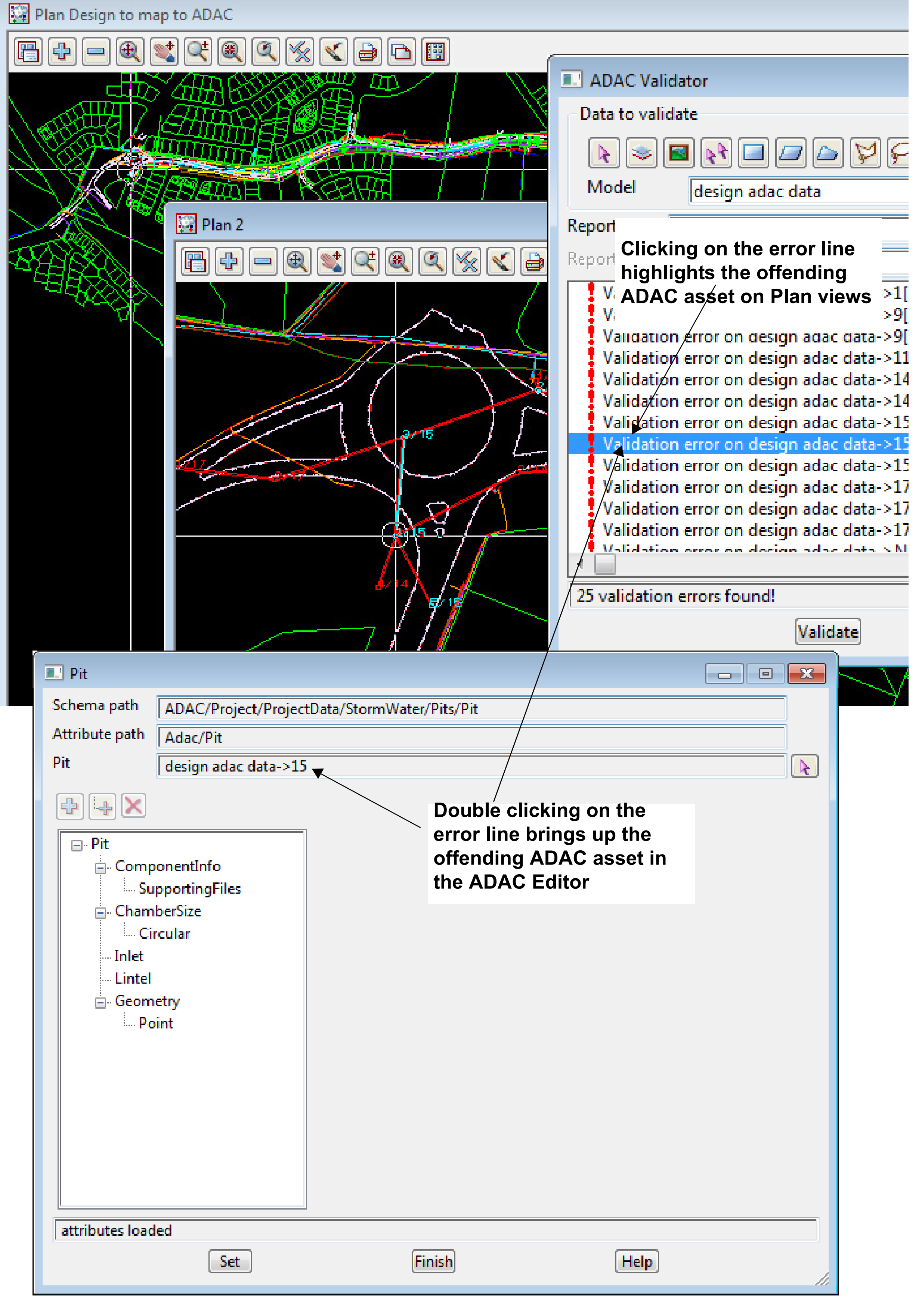
when **Validate** *is pressed, the string in the Data Source is validated against the ADAC XML Schema.*

*If no errors are found, then **No errors found!** is written to the panel's message area.*

For each error found, an error message is written to the area on the panel for error messages as an intelligent log line, and if **Report file** *is not blank, written in XML to the report file.*

Clicking on an error line will highlight and pan to the offending ADAC Asset in any Plan view the asset is on.

Double clicking on the log line, brings up the ADAC Asset Editor with the data for the string loaded into it.



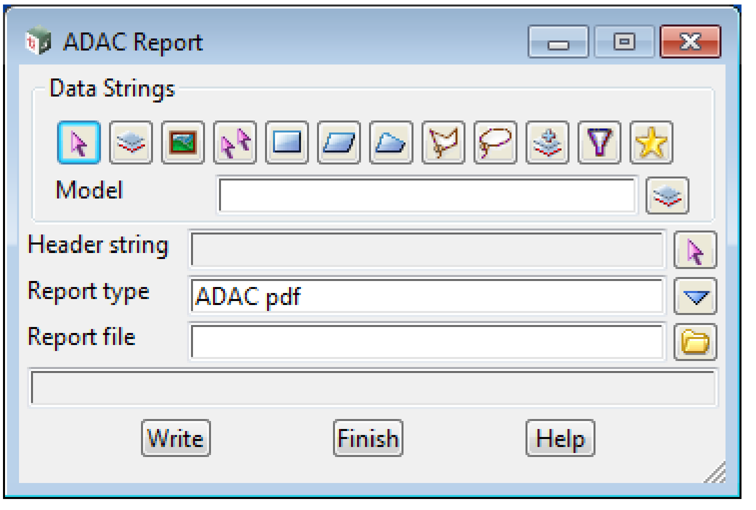
Continue to the next section 10.4.5 Report or return to 10.4 12d ADAC Menu .

10.4.5 Report

Position of option on menu: **File I/O =>ADAC =>Report**

Report produces a report in a variety of formats of a ADAC Header string and a user selection of ADAC Asset strings.

Selecting **Report** brings up the **ADAC Report** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source .

Data source	input		

source of data to be included in the ADAC report.

Header string	string select		

select the string that has the ADAC Header information for this report.

Report type	choice box	ADAC pdf, ADAC html, original xml, custom	

output format for the report. An XML file will be produced and then if required, converted to the selected report.

For information on setting up custom reports from the generated XML file using xslts, see 4.30 Setting Up XML Reports .

Report file	file box		

*if **not blank**, an XML file will be created and a report of this name, and of the type given by* **Report type** *will be generated from the XML file.*
*If **blank**, no report is created.*

Write	button		

writes out the report.

12D ADAC REPORT

Current Date: 31-01-2014, 19:00:01

Project name:	x	**Work Approval ID:**	
ADAC version:	4.0.0	**Drawing Number:**	x
Description:	x	**Drawing Revision:**	
Status:	Preliminary	**Construction date:**	
Export date time:	2013-06-05T15:46:35	**Software product:**	12d Model 11.0 Alpha 301 Lees tinkers 16 - Not For Production
Owner:		**Software version:**	11.0.0.301
Receiver:	x		

	Surveyor	Engineer
Name:		
Date approved:		
Date final survey:		

Coordinate system

Horizontal coordinate:	x
Horizontal datum:	x
Vertical datum:	x
Is approximate:	false
Origin mark:	
Notes:	

Drawing Extents

			GNSS Metadata						
	X	Y	X	Y	Z	Time	H_prec	V_prec	Std dev
South West	68920.586025	79890.501585							
North East	73334.968672	82603.065224							

Transport

Road Edge:

Object ID	14797225	14797227	14797229	14797232
Infrastructure code				
Owner				
Status	As-Constructed	As-Constructed	As-Constructed	As-Constructed
Notes				
Supporting File				
Type	Bitumen	Bitumen	Bitumen	Bitumen
Length(m)	87.788	127.719	72.756	25.983
Pavement Extension(mm)	1	1	1	1

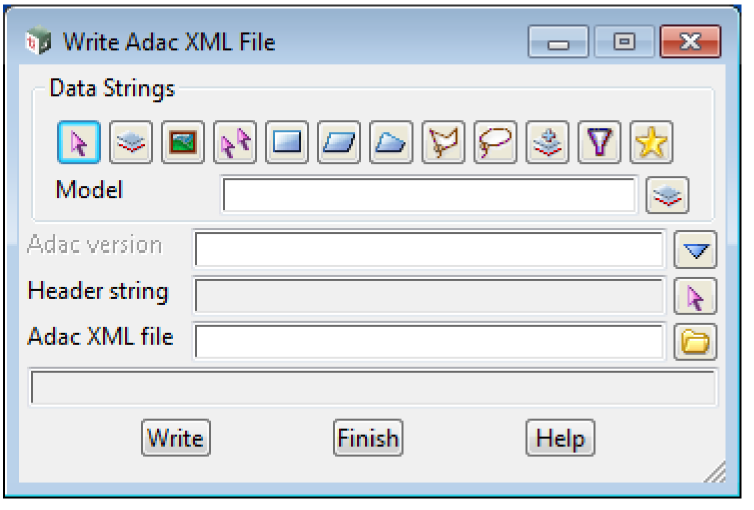
Continue to the next section 10.4.6 Write ADAC XML File or return to 10.4 12d ADAC Menu .

10.4.6 Write ADAC XML File

Position of option on menu: **File I/O =>ADAC =>Write ADAC file**

Write ADAC file produces an ADAC XML file using a user selected ADAC Header string and a user selection of ADAC Asset strings.

Selecting **Write ADAC file** brings up the **Write ADAC XML file** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source .

Data source input

source of ADAC Asset data to write out to an ADAC XML file.

Header string string select

select the string that has the ADAC Header information for this ADAC.XML file

Report file file box

name of the file to write the ADAC XML data to.

Write button

write out the ADAC XML file using the Header string as the ADAC Header data and the Data source of ADAC Asset strings.

Continue to the next section 10.4.7 Import ADAC XML File or return to 10.4 12d ADAC Menu .

10.4.7 Import ADAC XML File

Position of option on menu: **File I/O =>ADAC =>Import ADAC file**

Import ADAC file reads in an ADAC XML file and

(a) For the ***ADAC Header Information***, creates a ***12d Model*** one vertex string with an identical attribute structure to an ADAC Header string created in ***12d Model***. All the ADAC Header data in the ADAC XML file is placed in the attribute structure.

(i) The string is given the ***string name*** *header.*

(ii) This string is placed in the model ***adac project header*** with a **Pre*postfix for models** from the panel applied to the model name.

(b) For each ***ADAC Asset*** in the file, creates a ***12d Model*** string with the same geometry as the *ADAC Asset* has in the ADAC XML file.

(i) The ***12d Model*** string is given an identical attribute structure to an *ADAC Asset* created within ***12d Model***. All the *ADAC Asset* data in the ADAC XML file for that asset is read in and placed in the attribute structure.

(ii) The created string is given the **final** part of the ADAC Asset name as a ***string name***. For example, A *Transport>RoadEdges>RoadEdge* is given the name *RoadEdge*.

(iii) The string is placed in the model ***adac project data*** with a **Pre*postfix for models** from the panel applied to the model name.

It is unfortunate that with only sixty-eight ADAC Assets, there is a double up in such names because it is much easier to refer to *MaintenanceHole* than *Sewerage>MaintenanceHoles>MaintenanceHole*.

In *ADAC 4.1*, the double ups are:

MaintenanceHole - in both *Sewerage* and *WaterSupply*

Valve - in both *Sewerage* and *WaterSupply*

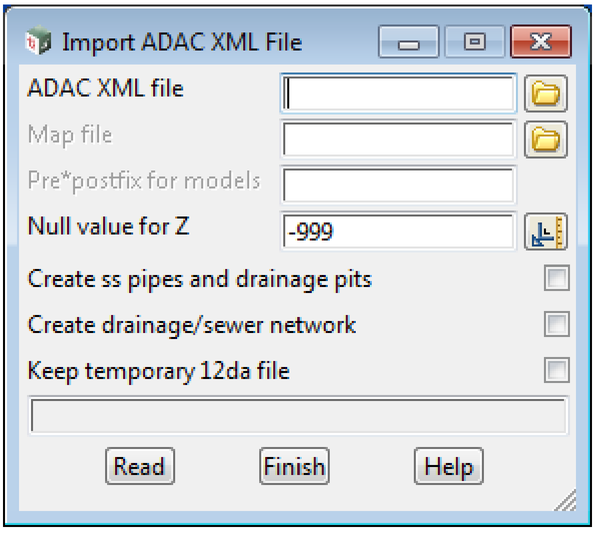
Fitting - in all three of *Sewerage*, *WaterSupply* and *StormWater*

Pipe - in both *WaterSupply* and *StormWater*

Connection - in both *Sewerage* and *Cadastre*.

To allow them to be distinguished, when ***12d Model*** reads the *ADAC Asset* data, the ***full name of the ADAC Asset*** is stored in the string text attribute ***Adac>Path***. So if a *Map File* is used when ***reading in*** the ADAC.MXL file, using this attribute as the **Att Key** in the **Basic** node allows the string's *ADAC Asset* type to be uniquely identified, and the string ***renamed***.

Selecting **Import ADAC file** brings up the **Import ADAC XML file** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
ADAC XML file to import	file box		available .XML files

name of the ADAC XML file to import.

*The default model names (with the **Pre*postfix for models** applied) to hold the data read in are:*

adac project header *for the string containing the ADAC Header data.*

adac project data *for the strings for each ADAC Asset*

Field Description	Type	Defaults	Pop-Up
Map file	file box		*.mapfile, *.mf files

*if **blank**, no **12d Map File** is used*

*If **not blank**, the name of the **Map File** to be used when importing data from the ADAC.XML file.*

*When using a **Map File**, the short **ADAC Asset** name in the ADAC XML file is used as the entity-name for matching with the key in the Map File. For example, a* Transport>RoadEdges>RoadEdge *is given the name* RoadEdge.

*That is, when the ADAC Asset is read in, it can be mapped by having an entry in the **Base** node of the Map File with short ADAC Asset name as the* **Key**.

For example, RoadEdge, *or* RoadEdge* *as the* **Key** *will map all the* Transport>RoadEdges>RoadEdge *that are read in.*

*When importing, all the ADAC elements associated with the ADAC Asset are placed in a 12d attribute structure identical to that used when ADAC Assets are created within **12d Model**. So the ADAC Asset attributes are also available to use in the **Base** node of the Map File as an* **Att Key** *when importing the ADAC XML file.*

*Note that the **full** ADAC Asset name is stored in the string attribute **Adac>Path** so this is available to use as the* **Att Key** *to distinguish between ADAC Assets with the same short name.*

See the section 8.8.1 Create/Edit a Map File *for information about 12d map files.*

Field Description	Type	Defaults	Pop-Up
Pre*postfix for models	pre*postfix box	name of xml file (without the XML)	

*When an XML is selected, the file name (without the .xml) plus " * " is written to this field as the default.*

*if **not blank**, a prefix and a postfix applied to the default model names that are created.*

If there is a 12d Map File then this will also be applied to any models created by the map file.

Go to the section 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix.*

Field Description	Type	Defaults	Pop-Up
Null value for Z	real box	-999	

if in the ADAC geometry there is no z-coordinate, then the coordinate is given this value in 12d Model.

Create ss pipes and drainage pits tick box

*if **not ticked**, the string created by ADAC are super strings with no diameters.*

*If **ticked**, ADAC pipes are created as super strings with round or box sections.*

*If **ticked and you have the drainage module**, Sewerage MaintenanceHoles and Stormwater Pits create a pit on a one vertex drainage string, and the pit geometry is defined by the ADAC elements.*

Create drainage network tick box

*If **ticked and you have the drainage module**, 12d Model will try to join the Sewerage MaintenanceHoles and NonPressurePipes into Sewer lines, and the Stormwater Pits and Pipes into drainage lines.*

However, ADAC has no information about which Pits/MaintenanceHoles go with which Pipes, so the results can only be a best guess.

If the ADAC XML file was created by 12d Model from 12d Model drainage and sewer strings, then extra information is added to the ADAC XML data so that the original networks can be reconstructed.

When drainage strings are created, they are given a name starting with StormWater *so to map them when reading in the data you will need an entry in the **Base** node of the Map File with* StormWater* *as the* **Key**.

When sewer strings are created, they are given a name starting with Sewerage *so to map them when reading in the data you will need an entry in the **Base** node of the Map File with* Sewerage* *as the* **Key**.

Keep temporary 12da file tick box

when an ADAC XML file is being imported, a temporary 12da file is created.

*If **not ticked**, the temporary 12da file is deleted.*

*If **ticked**, the temporary 12da file is not deleted.*

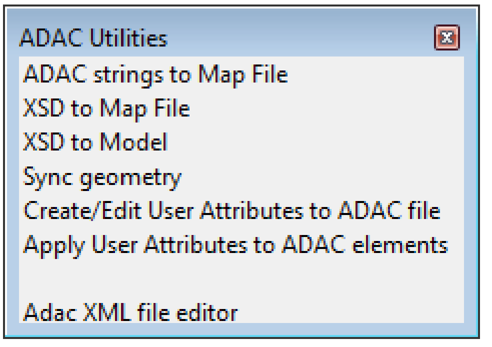
Import button

when clicked, the ADAC XML file is processed and the ADAC data imported into 12d Model.

Continue to the next section 10.4.8 ADAC Utilities or return to 10.4 12d ADAC Menu .

10.4.8 ADAC Utilities

Position of menu: **File I/O =>ADAC =>Utilities**



See

- 10.4.8.1 ADAC Strings to Map File
- 10.4.8.2 XSD to Map File
- 10.4.8.3 XSD to Model
- 10.4.8.4 Sync Geometry
- 10.4.8.5 Create/Edit User Attributes to ADAC File
- 10.4.8.6 Apply User Attributes to ADAC Elements
- 10.4.8.7 ADAC XML File Editor

10.4.8.1 ADAC Strings to Map File

Position of option on menu: **File I/O =>ADAC =>Utilities =>ADAC strings to Map File**

ADAC strings to Map File takes the selected strings representing *ADAC Assets* and creates a **Map File** from them in the following way:

For applications that **don't** have ***12d Model*** drainage or sewer strings:

each ADAC string creates a row in the *Attributes>String* grid of the ***Map File*** with the **string name** followed by a * placed in the **Name** column. The first string attribute's name, type and value is placed in the **Att Key** column, and the **ADAC attribute group** placed in the **Map Attributes** column.

For applications that **do** have ***12d Model*** drainage or sewer strings then have the **User Vertex and segment attributes** tick box **ticked** and

(a) each ADAC *Sewerage>MaintenanceHoles>MaintenanceHole* string creates a row in the *Attributes>Vertex/Pit* grid of the ***Map File*** with * placed in the **Name** column, an Integer attribute named **sewertype** with value **1**, placed in the **Att Key** column. The first string attribute's name, type and value is placed in the **Vertex Att Key** column and the **ADAC attribute group** placed in the **Map Attributes** column. The string name is written to **Comment** column.

(b) each ADAC *StormWater>Pits>Pit* string creates a row in the *Attributes>Vertex/Pit* grid of the ***Map File*** with * placed in the **Name** column, an Integer attribute named **sewertype** with value **0**, placed in the **Att Key** column. The first string attribute's name, type and value is placed in the **Vertex Att Key** column and the **ADAC attribute group** placed in the **Map Attributes** column. The string name is written to **Comment** column.

(c) each ADAC *Sewerage>PipesNonPressure>PipeNonPressure* string creates a row in the *Attributes>Segment/Pipe* grid of the ***Map File*** with * placed in the **Name** column, an Integer attribute named **sewertype** with value **1**, placed in the **Att Key** column. The first string attribute's name, type and value is placed in the **Segment Att Key** column and the **ADAC attribute group** placed in the **Map Attributes** column. The string name is written to **Comment** column

(d) each ADAC *StormWater>Pipes>Pipe* string creates a row in the *Attributes>Segment/Pipe* grid of the ***Map File*** with * placed in the **Name** column, an Integer attribute named **sewertype** with value **0**, placed in the **Att Key** column. The first string attribute's name, type and value is placed in the **Segment Att Key** column and the **ADAC attribute group** placed in the **Map Attributes** column. The string name is written to **Comment** column.

(e) all other ADAC strings create a row in the *Attributes>String* grid of the ***Map File*** with the **string name** followed by a * placed in the **Name** column, and the first string attribute's name, type and value placed in the **Att Key** column, and the **ADAC attribute group** placed in the **Map Attributes** column.

For example, if you had a model with three ADAC Assets in it

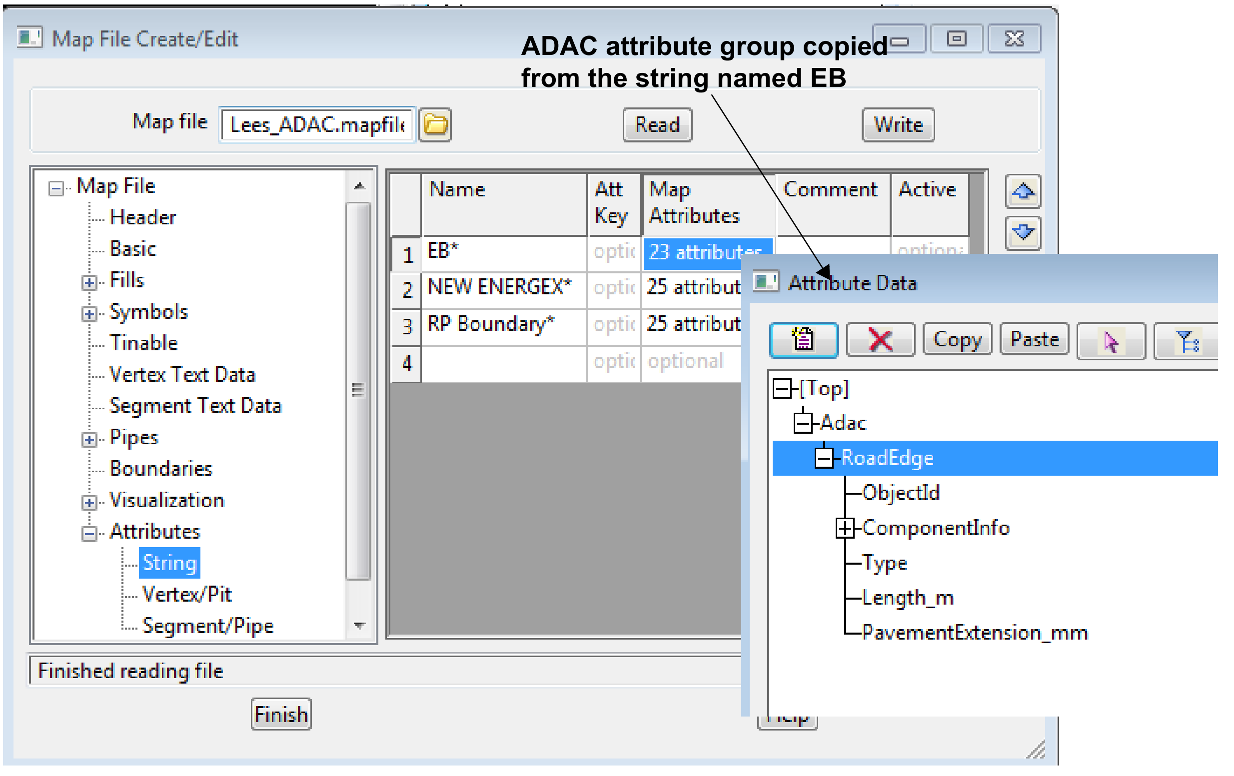
(a) a Road Edge with the string name EB

(b) an ElectricalConduit with string name NEW ENERGEX

and

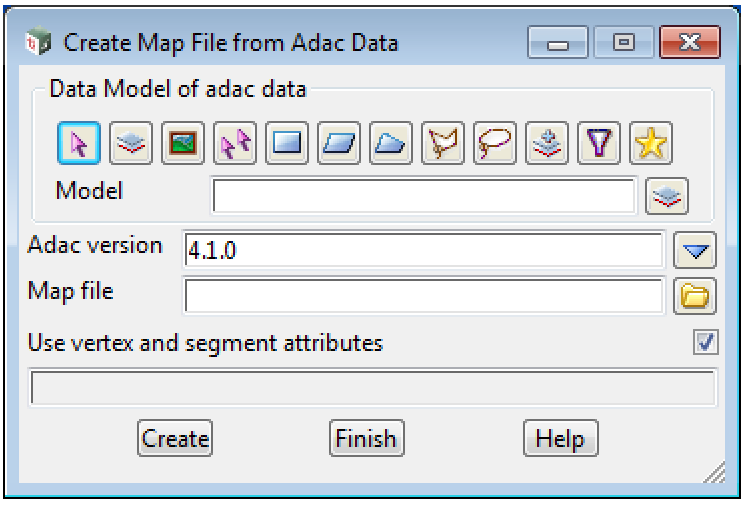
(c) a Lot with string name RP Boundary

then running the option **Assets to Map File** creates a ***Map File*** with three entries in the *Attributes >Strings* section:



Assets to Map File is the best method for creating a **12d Map File** to be used in the ADAC Survey and Designers Chains.

Selecting **Assets to Map file** brings up the **Create Map File from ADAC Data** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source *.*

Data source	input		

source of ADAC Asset data to create a 12d Map File from.

ADAC version choice box from env.4d 4.0.0, 4.1.0

this field is optional, but if it is set, then only strings representing ADAC Assets of this version of the ADAC XML Schema will have entries created in the Map File.

Map file file box *.mapfile, *.mf files

*the name of the **12d Map File** that is created and has the **Attributes >String** section.*

For each string that is an ADAC Asset of the ADAC version given in the **Version** *field, create a row in the Attributes >String section of the this **Map File** with the name of the string followed by an * in the **Name** column, and in the same row, all the string's string attributes in the Adac group are copied to the **Map Attributes** column*

See the section 8.8.1 Create/Edit a Map File *for information about 12d map files.*

User vertex and segment attributes tick box not ticked

*if **not ticked**, each ADAC string creates an entry in the Attributes >String section of the Map File.*

*If **ticked,** ADAC Sewerage>MaintenanceHoles>MaintenanceHole and StormWater>Pits>Pit create entries in the Attributes>Vertex/Pit section of the Map File, ADAC Sewerage>PipesNonPressure>PipeNonPressure and StormWater>Pipes>Pipe create entries in the Attributes>Segment/Pipe section of the Map File, and all other ADAC strings create entries in the Attributes >String section of the Map File.*

Create button

*when clicked, the **Map File** is created.*

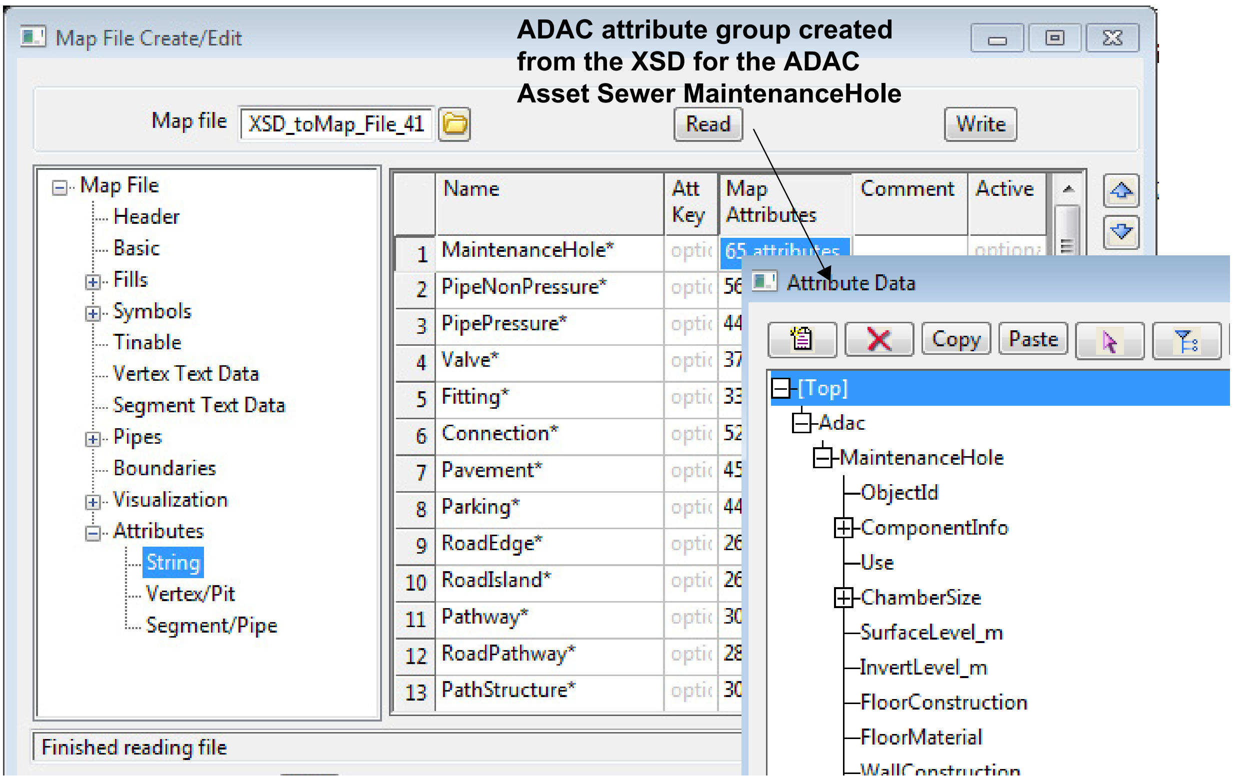
Continue to the next section 10.4.8.2 XSD to Map File or return to 10.4.8 ADAC Utilities .

10.4.8.2 XSD to Map File

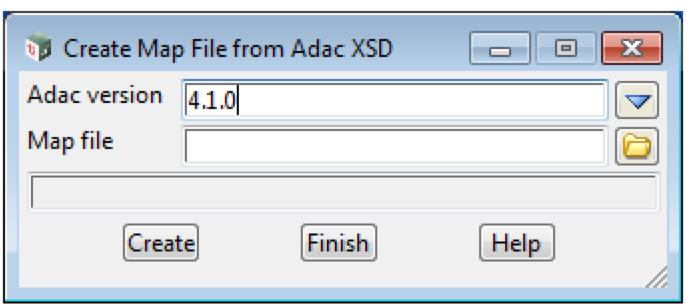
Position of option on menu: **File I/O =>ADAC =>Utilities =>XSD to Map File**

XSD to Map File takes the *ADAC XML Schema* XSD for a given version and creates a **12d Map File** (or just **Map file**), and in the *Attributes >String section*, creates a row for each Asset type in the XSD (sixty-eight of them) with the short name of the ADAC Asset followed by an * in the **Name** column, and in the same row, creates an ADAC group of attributes for that Asset type in the **Map Attributes** column with default values for each of the ADAC elements in the ADAC group.

For example, running the **XSD to Map File** option for ADAC XML version 4.1.0 Schema, creates the **Map File**:



Selecting **XSD to Map file** brings up the **Create Map File from ADAC XSD** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
ADAC version	choice box	from env.4d	4.0.0, 4.1.0

this is the ADAC XML XSD to create the 12d Map File from.

Map file file box *.mapfile, *.mf files

*the name of the **Map File** that is created. And in the **Attributes >String** section, for each ADAC Asset in the XSD a row is created with the short name for the ADAC asset followed by an * in the **Name** column, and in the same row, an ADAC group is created in the **Map Attributes** column of the type, and all the elements in the XSD for that ADAC Asset are copied as attributes into the ADAC group.*

See the section 8.8.1 Create/Edit a Map File *for information about 12d map files.*

Create button

*when clicked, the **Map File** is created.*

Continue to the next section 10.4.8.3 XSD to Model or return to 10.4.8 ADAC Utilities .

10.4.8.3 XSD to Model

Position of option on menu: **File I/O =>ADAC =>Utilities =>XSD to model**

XSD to Model takes a given version of the *ADAC XML Schema* XSD and, for each ADAC Asset in the XSD, creates a super string using:

(a) the short name of the ADAC Asset as the string name

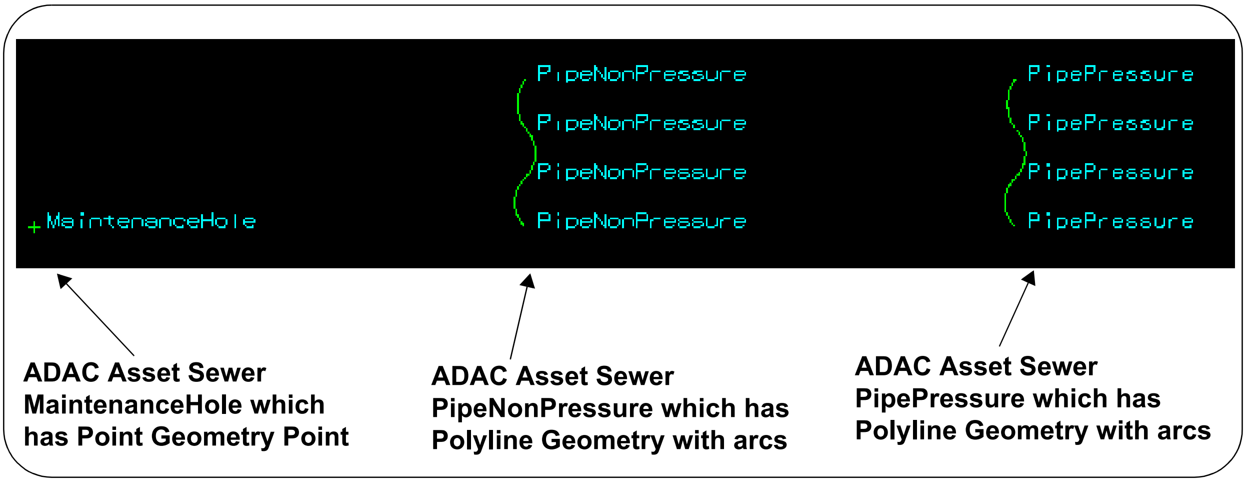
(b) in the string attributes, an ADAC group with default values for each of the ADAC elements

(c) the ADAC Asset Geometry with default values as the string geometry

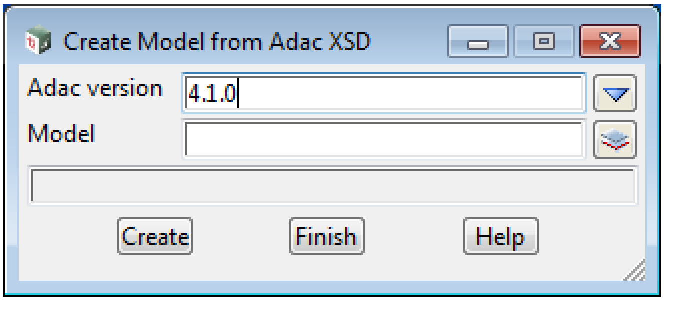
(d) green for the string colour.

So there will be sixty-eight strings created for ADAC versions 4.1 or 4.0.

For example, running the **XSD to model** option for ADAC XML version 4.1.0 Schema creates a **model**, and the first three strings in that model are:



Selecting **XSD to model** brings up the **Create Model from ADAC XSD** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
ADAC version	choice box	from env.4d	4.0.0, 4.1.0

this is the ADAC XML XSD to create the strings for.

Field Description	Type	Defaults	Pop-Up
Model	model box		available models

for each ADAC Asset in the XSD, a super string is created using:

(a) the short name of the ADAC Asset as the string name

(b) in the string attributes, an ADAC group with default values for each of the ADAC elements

(c) the ADAC Asset Geometry with default values as the string geometry

(d) green for the string colour.

These ADAC strings are added to this model.

See the section 8.8.1 Create/Edit a Map File *for information about 12d map files.*

Create button

when clicked, the model of strings for ADAC Assets in the XSD is created.

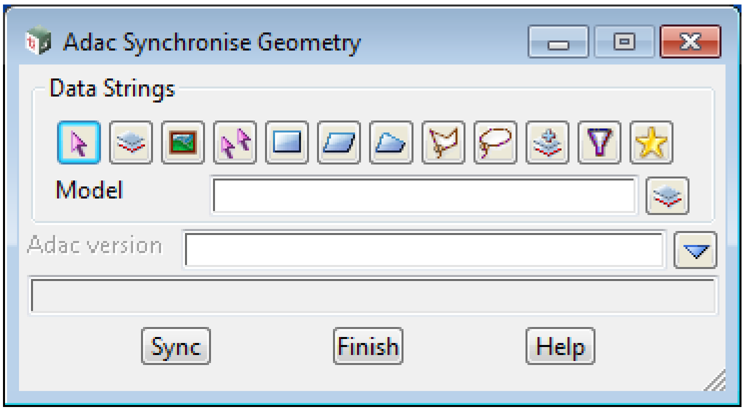
Continue to the next section 10.4.8.4 Sync Geometry or return to 10.4.8 ADAC Utilities .

10.4.8.4 Sync Geometry

Position of option on menu: **File I/O =>ADAC =>Utilities =>Sync geometry**

Sync geometry takes selected strings representing ADAC Assets and updates the Geometry section of the ADAC attributes for the string with the actual geometry of the string.

Selecting **Sync geometry** brings up the **ADAC Synchronise Geometry** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source .

Data source	input		

source of data to process. The ADAC Asset strings will have their ADAC Geometry attributes updated.

ADAC version	choice box	from env.4d	4.0.0, 4.1.0

this field is optional but if it is set, then only strings representing ADAC Assets of this version of the ADAC XML Schema will have their ADAC Geometry updated.

Sync	button		

when clicked, the selected data will have its ADAC Geometry attributes updated from the actual string geometry.

Continue to the next section 10.4.8.5 Create/Edit User Attributes to ADAC File or return to 10.4.8 ADAC Utilities .

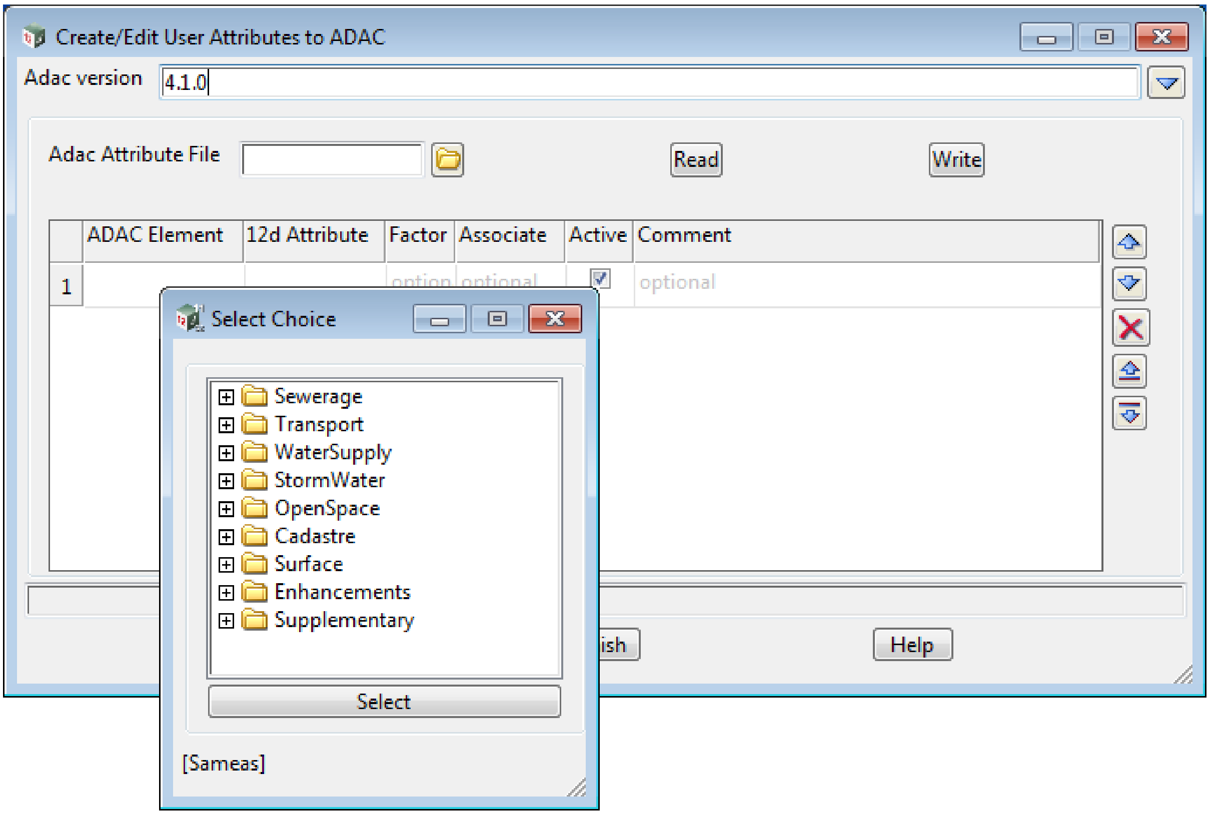
10.4.8.5 Create/Edit User Attributes to ADAC File

Position of option on menu: **File I/O =>ADAC =>Utilities =>Create/Edit User Attributes to ADAC file**

This option creates and edits a **12duaf** file which defines what 12d string, vertex or segment User attributes are used to update user specified ADAC Asset element values.

The **12duaf** file is used to update the values in ADAC Assets by the **Apply User Attributes to ADAC** panel (see 10.4.8.6 Apply User Attributes to ADAC Elements).

Selecting **Create/Edit User Attributes to ADAC file** brings up the **Create/Edit User Attributes to ADAC** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
ADAC version	choice box	from env.4d	4.0.0, 4.1.0

the version of the ADAC XML Schema that the ADAC Asset elements are from. The version is necessary because the names can change between versions.

Field Description	Type	Defaults	Pop-Up
ADAC Attributes file	file box		*.12duaf files

name of the 12d User Attribute to ADAC file to read or write.

Read button

click on this button and the file given in the **ADAC Attribute file** *field is read in and loaded into the editor.*

Write button

writes out the data in the editor to the file given in the **ADAC Attribute file** *field.*

12duaf File Grid

when an 12duaf file is read in, it is displayed in the 12duaf tree.

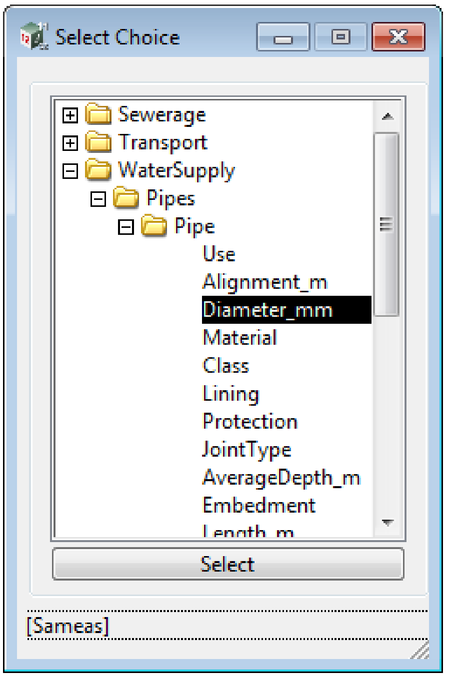
ADAC Element text column

the full name of the ADAC element to have its value updated.

For example,

WaterSupply/Pipes/Pipe/Diameter_mm

*The name can be typed in but it is easiest to select it from the ADAC Schema by clicking RB in the **ADAC Element** row and expanding the tree and double clicking on the required ADAC element.*



*This ADAC element is given the value from the same row of the following **12d Attribute**, **multiplied**, when appropriate, by **Factor** when Factor is non zero.*

12d Attribute text column

*the full path name of the **12d User Attribute** with*

String/ *preceding the name if it is a **String Attribute**.*

Vertexn/ *preceding the name if it is a **Vertex Attribute** on the **n**'th vertex.*

Segmentn/ *preceding the name if it is a **Segment Attribute** on the **n**'th segment.*

For example,

Vertex1/Survey/Diameter

*is the Vertex attribute called **Survey/Diameter** on vertex 1*

*Note that **case** is important in **User Attribute** names.That is, upper and lower case characters are considered different.*

Factor real column

*for a row where **Factor** is non zero, the value of the* **ADAC Element** *column is set equal to the value in the* **12d Attribute** *column multiplied by the value in the* **Factor** *column.*

*For example, **Factor** would be set to 1000 if the **12d Attribute** was in metres and the **ADAC Element** was in millimetres.*

Active tick box tick

*if **ticked**, use this row of the grid.*
*If **not ticked**, do not use this row of the grid.*

Comment text

a comment for this row of the grid

Buttons at Bottom

Validate button

when clicked, check that the ADAC Element names against the appropriate ADAC Schema.

Continue to the next section 10.4.8.6 Apply User Attributes to ADAC Elements or return to 10.4.8 ADAC Utilities .

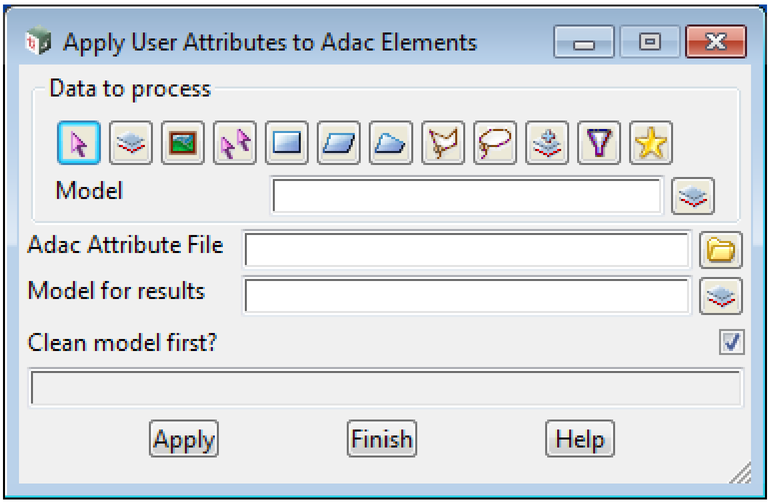
10.4.8.6 Apply User Attributes to ADAC Elements

Position of option on menu: **File I/O =>ADAC =>Utilities =>Apply User Attributes to ADAC elements**

This option applies a **12duaf** file to ADAC Assets to update the *ADAC elements* with values from the 12d *User Attributes* of the ADAC Asset string.

The **12duaf** file is created and edited by the **Create/Edit User Attributes to ADAC** panel (see 10.4.8.5 Create/Edit User Attributes to ADAC File).

Selecting **Apply User Attributes to ADAC elements** brings up the **Apply User Attributes to ADAC Elements** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to process. The ADAC Asset strings will have the ADAC Elements updated by User Attributes.

ADAC Attributes file	file box		*.12duaf files

name of the 12d User Attribute to ADAC file to use.

Model for results	model box		available models

model that the processed strings are added to.

Clean model first ?	tick box		

*if **ticked**, the model* **Model for results** *is cleaned before any strings are added to it.*
*If **not ticked**, the model* **Model for results** *is NOT cleaned.*

Apply	button		

when clicked, the selected data will have any ADAC Elements updated by User Attributes according to the selected 12duaf file.

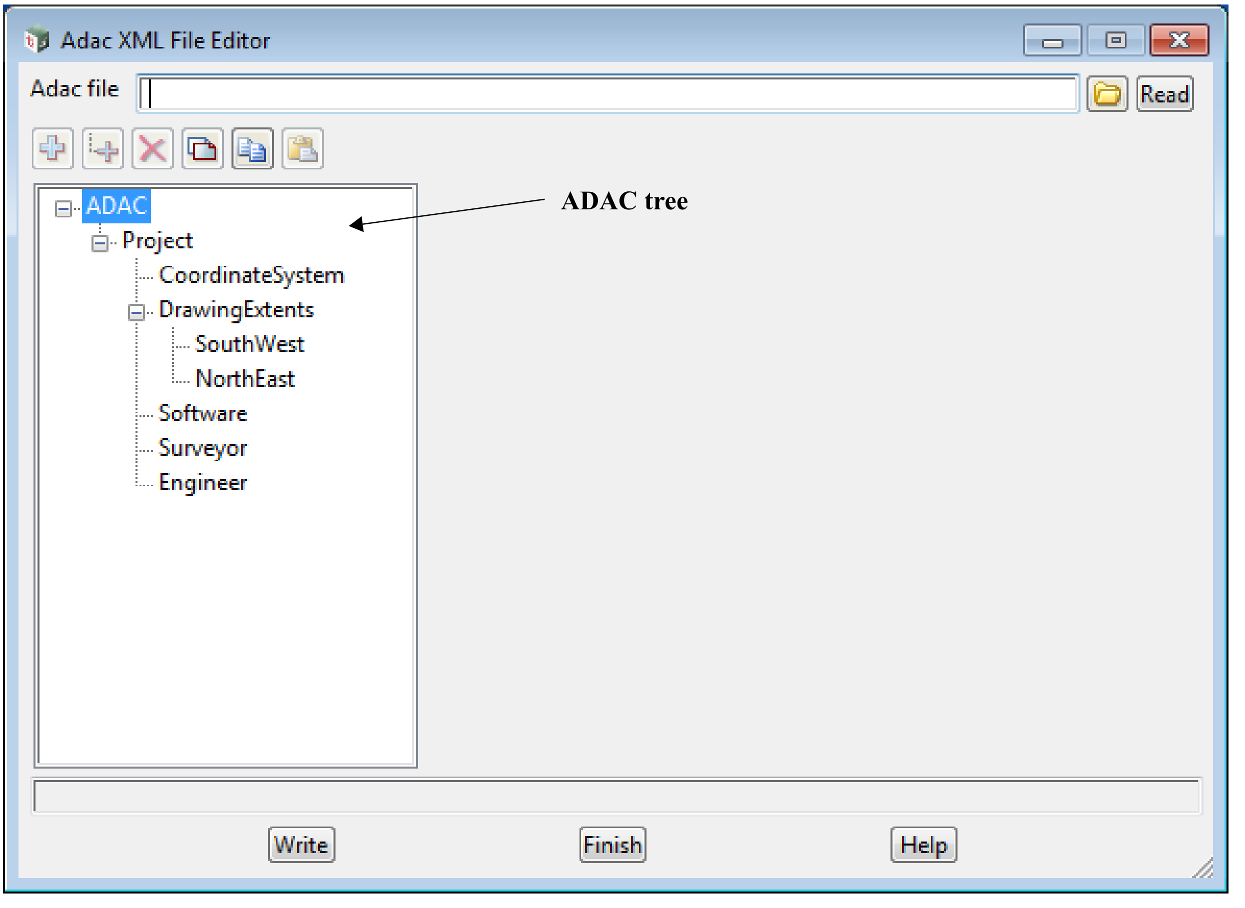
Continue to the next section 10.4.8.7 ADAC XML File Editor or return to 10.4.8 ADAC Utilities .

10.4.8.7 ADAC XML File Editor

Position of option on menu: **File I/O =>ADAC =>Utilities =>ADAC file editor**

ADAC XML file editor reads an ADAC XML file and loads it into the **ADAC XML File Editor**.

Selecting **ADAC file editor** brings up the **ADAC XML File Editor** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
ADAC XML file	file box		*.XML files

name of the XML file to read or write.

Read button

click on this button and the file given in the **ADAC XML file** *field is read in and loaded into the editor.*

ADAC tree

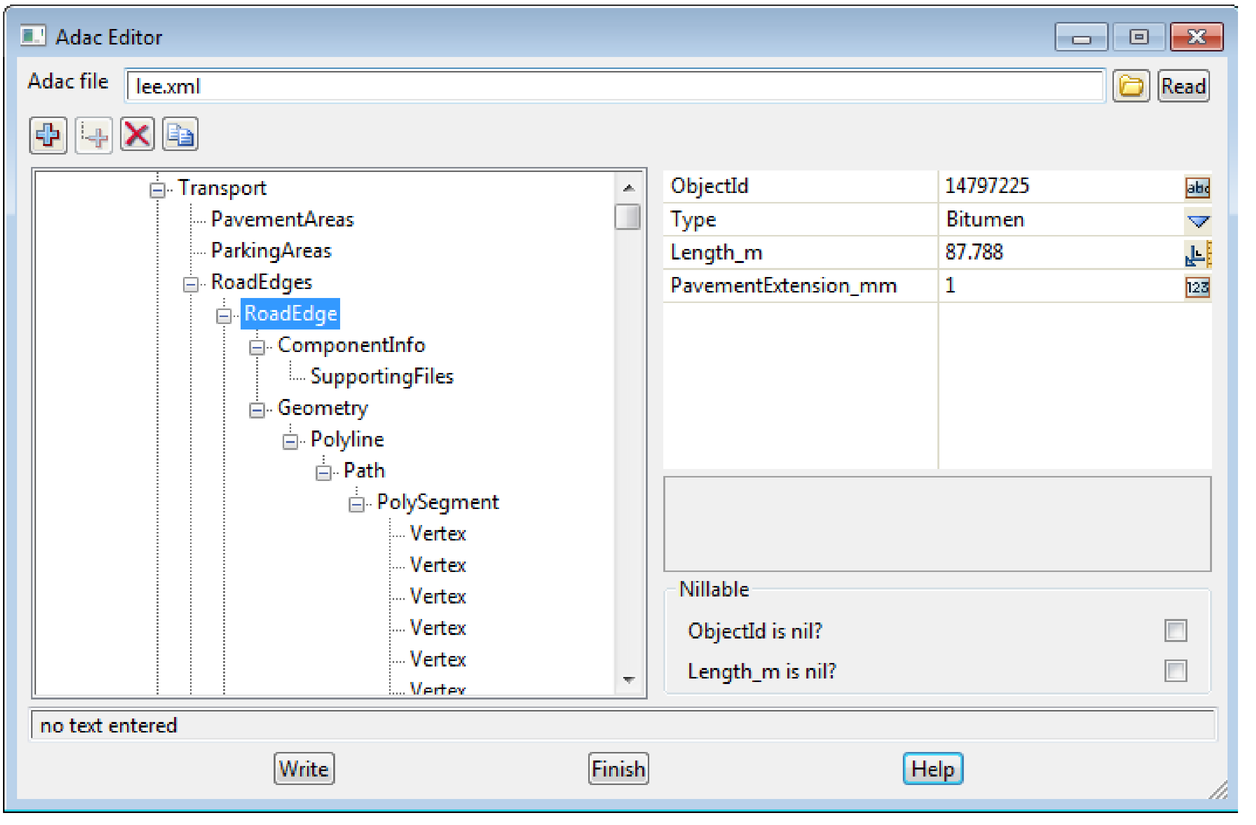
when an ADAC XML file is read in, it is displayed in the ADAC tree.

Elements in the tree can be edited according to the ADAC XSD. New ADAC Assets can also be added.

Write button

writes out the data in the editor to the file given in the **ADAC XML file** *field.*

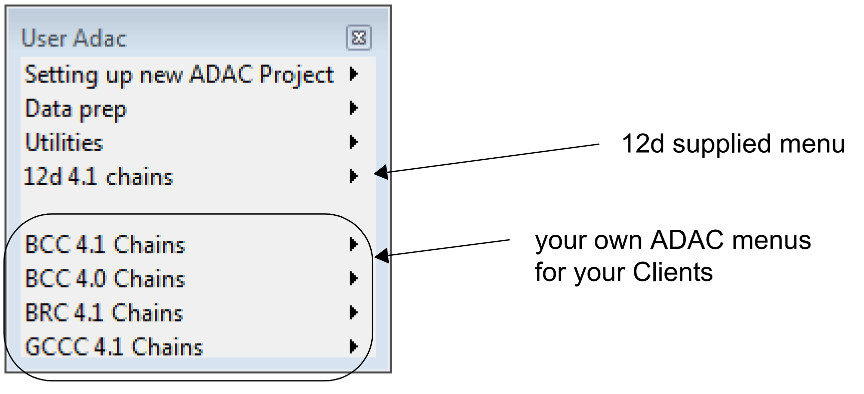
As an example,



Return to 10.4.8 ADAC Utilities .

10.4.9 User ADAC

Position of menu: **File I/O =>ADAC =>User**



See

10.4.9.1 Setting Up a New ADAC Project
10.4.9.2 Data Prep
10.4.9.3 User ADAC Utilities

The easiest way to run the ADAC *Survey* or *Design* chains with specific pvf files is to run them from a menu. This can be done in two ways:

(a) Running Specific Map and 12duaf files from Local and User_Lib

12d Model provides options on the walk right menu **Run 12d ADAC 4.1 Chains** on the option **12d 4.1 chains** to run ADAC *Survey* or *Design* chains for ADAC 4.1 with specially named Map and 12duaf files in the working folder for the project (local), or User_Lib or Library.

See 10.4.9.4 12d 4.1 Chains.

(b) Running Client Specific Map and 12duaf files

Once you are producing ADAC files for different Clients, you may need to produce different ADAC versions, or need different Map or 12duaf files.

For example you may need to use a different naming convention, or the information required in the ADAC assets is different.

For these situations, you can add your own options to **User Adac** menu.

The user options can run either the ADAC Base Survey or Design chains with Client specific chain pvf files.

For information on setting up your own options on the **User ADAC** menu, see 10.4.9.5 Client Specific ADAC Design & Survey Menus.

Or return to 10.4.8 ADAC Utilities.

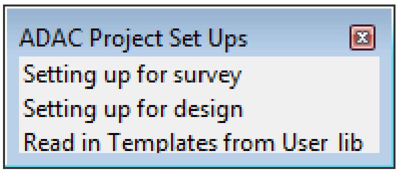
10.4.9.1 Setting Up a New ADAC Project

Position of menu: **File I/O =>ADAC =>User =>Setting up a new ADAC Project**

A new project for producing ADAC XML needs to be set up a certain way for the ADAC Chains to work.

There is a different setup for a **Survey** Project to that for a **Design** Project.

ADAC Header and Asset templates can also be read in from the file **ADAC_Templates.12da** in User_Lib.



See

10.4.9.1.1 Setting Up for Survey
10.4.9.1.2 Setting Up for Design
10.4.9.1.3 Reading in Templates from User_Lib

10.4.9.1.1 Setting Up for Survey

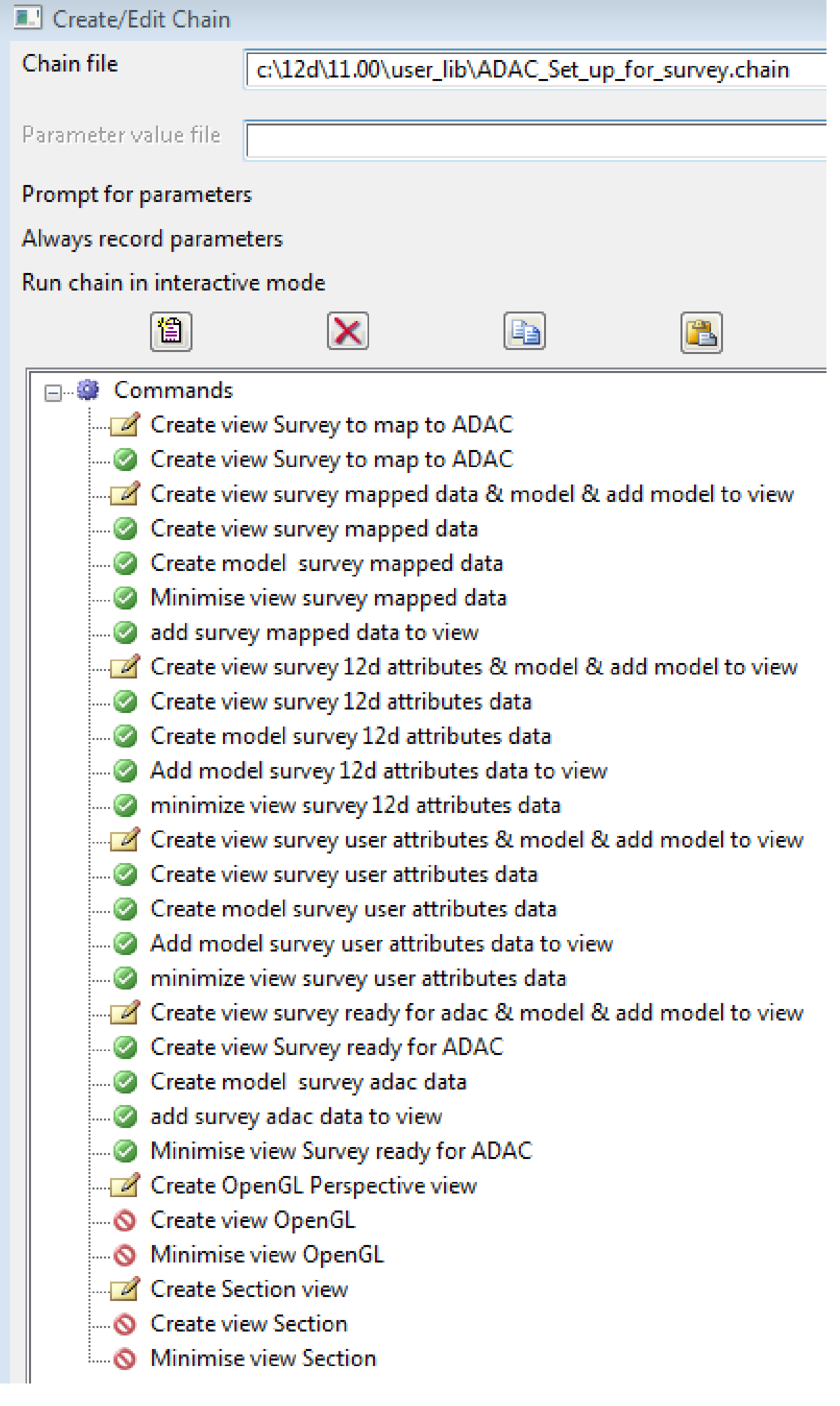
Position of option on menu:

File I/O =>ADAC =>User =>Setting up a new ADAC Project =>Setting up for survey

Setting up for survey runs the Chain ***ADAC_Set_up_for_survey.chain***, which is in *$LIB.*

This chain creates some new Plan views and some models which are added to the views, and the views then minimised.

The Plan views and models are used by the ADAC Survey Chain.



Continue to the next section 10.4.9.1.2 Setting Up for Design or return to 10.4.9.1 Setting Up a New ADAC Project.

10.4.9.1.2 Setting Up for Design

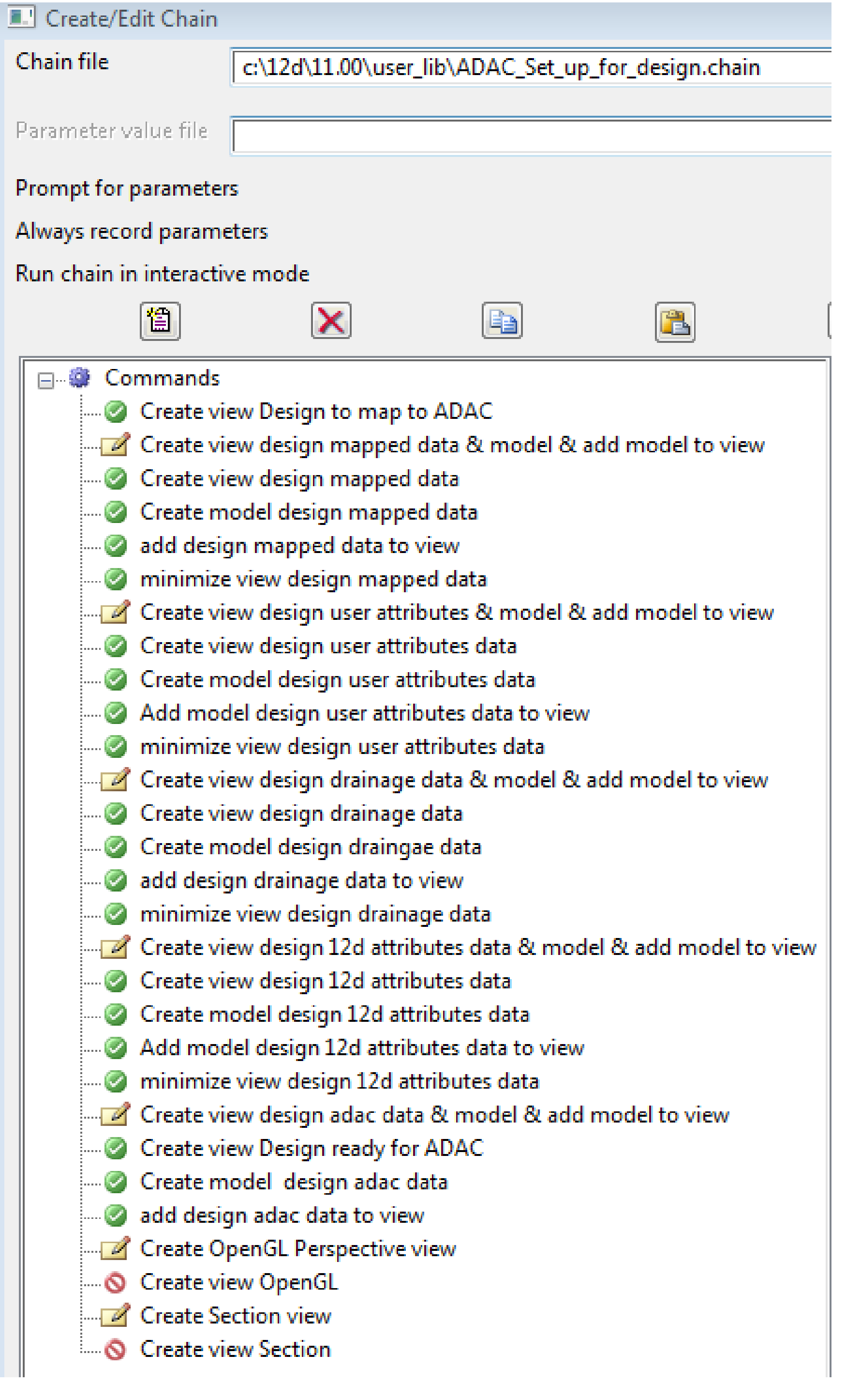
Position of option on menu:

File I/O =>ADAC =>User =>Setting up a new ADAC Project =>Setting up for design

Setting up for design runs the Chain ***ADAC_Set_up_for_design.chain***.which is in *$LIB*

This chain creates some new Plan views and some models which are added to the views, and the views then minimised.

The Plan views and models are used by the ADAC Design Chain.



Continue to the next section 10.4.9.1.3 Reading in Templates from User_Lib or return to 10.4.9.1 Setting Up a New ADAC Project.

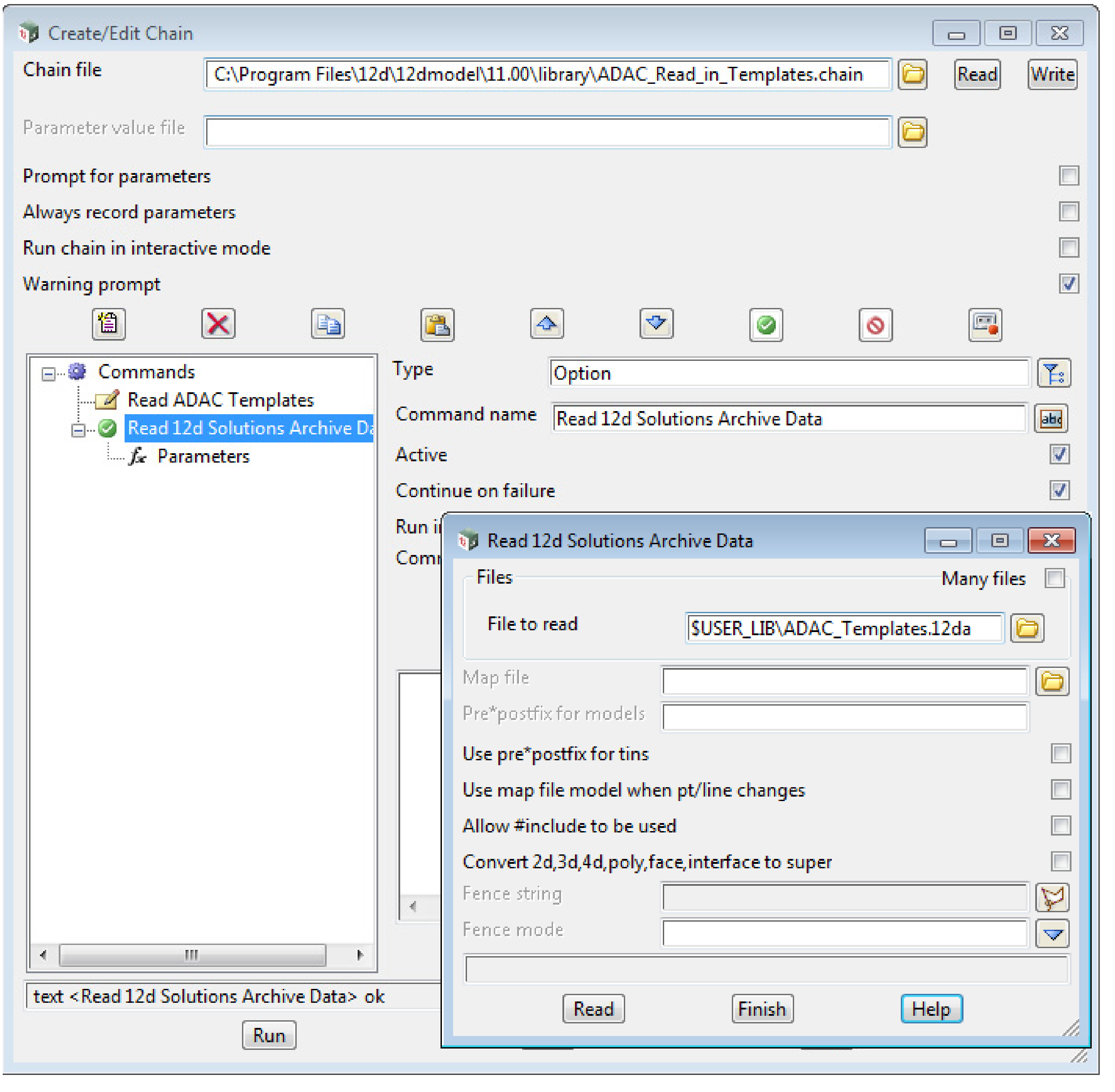
10.4.9.1.3 Reading in Templates from User_Lib

Position of option on menu:

File I/O =>ADAC =>User =>Setting up a new ADAC Project =>Read in templates from User_Lib

Reading Templates from User_Lib runs the Chain ***ADAC_Read_in_Templates.chain*** which is in *Library.*

This chain reads in the file ***ADAC_Templates.4da*** from *User_Lib*.



So if you have created Header and Assets Templates then by writing the models ***ADAC Header Templates*** and ***ADAC Asset Templates*** which contain the Templates out to this then this option can be used to read them into a new project.

For information on ADAC Header and Asset Templates, see 10.6.4 Setting Up ADAC Templates .

Return to 10.4.9.1 Setting Up a New ADAC Project .

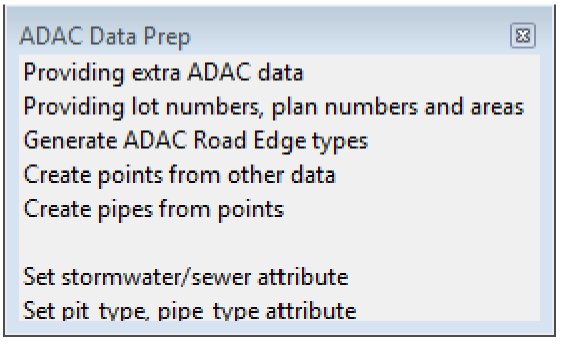
10.4.9.2 Data Prep

Position of menu: **File I/O =>ADAC =>User =>Data prep**

When you start with a Survey and Design in ***12d Model***, not all the information required for ADAC will necessarily be there. In fact, some of the ADAC information may even be added at a later stage by someone other than the Surveyor or Designer.

Similarly, some of the strings may not be of the correct type to become an ADAC Asset. For example, a string may not have the correct geometry as defined for the ADAC Asset in the *ADAC XML Schema XSD*.

So before running an ADAC Survey or ADAC Design chain, one or more of the **ADAC Data Prep** options may need to be run to prepare the strings that will become ADAC Assets.



See

10.4.9.2.1 Providing Extra Data for ADAC

10.4.9.2.2 Providing Lot and Plan Numbers and Areas for ADAC

10.4.9.2.3 Generate ADAC Road Edge Types

10.4.9.2.4 Create Points from Other Data

10.4.9.2.5 Create Pipes from Points

10.4.9.2.6 Set Stormwater/Sewer Property

10.4.9.2.7 Set a Drainage-Sewer Pit and Pipe Type Attribute

10.4.9.2.1 Providing Extra Data for ADAC

Position of option on menu: File I/O =>ADAC =>User =>Data prep =>Providing extra ADAC data

This option creates, as string attributes, information that is required for ADAC but is not normally part of a survey or a design.

The extra information is stored as attributes with the string, but not as ADAC attributes. A macro in the *ADAC Survey* or *ADAC Design* chain moves them into ADAC attributes. See (10.5.1.2.4 Update ADAC Elements from 12d Created Attributes on the String - Survey)

A major reason not to create ADAC attributes straightaway is that although this option allows you to enter the values, in the future the values might already exist in another form and routines written to automatically get them. In this case it will be much easier for anyone to put them into a simpler attribute structure and not have to know anything about the complex ADAC attribute structure.

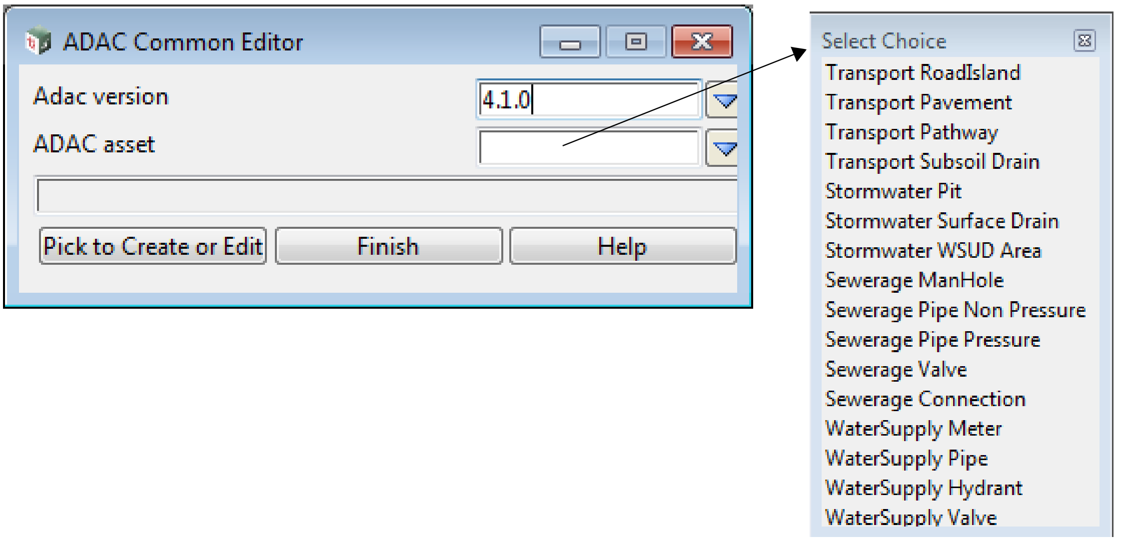
Another reason is that at this stage the *ADAC Survey* and *Design* chains have not been run to mark strings as ADAC Assets and set up the ADAC attributes group.

And when the *ADAC Survey* and *Design* chains are run, at each step in the chain they work with copies of the original data and clean out the models from a previous run rather than updating the original data. This is for safety so that it is easy to see that each step in the chain has worked, and if there is a problem, the chains can be run again and again.

So placing the attributes on the original data ensures they will still be there when the *ADAC Survey* or *Design* chains are rerun.

One important thing to keep in mind when running this option is that although you are using it to set up attributes that will go to ADAC, when you are picking the strings they are not yet marked as ADAC Assets. So you have to know **what ADAC Asset** the string is **going to become**.

Selecting **Providing extra data for ADAC** brings up the **ADAC Common Editor** panel

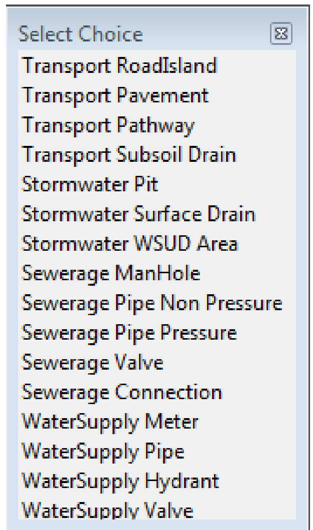


The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
ADAC version	choice box	from env.4d	4.0.0, 4.1.0

because the strings to be picked have not yet been marked as ADAC Assets, you need to let the option know what the ADAC version is going to be. This is required so that the correct ADAC Schema XSD is used to get information for the choices.

ADAC asset choice box



the ADAC Asset that this string will become.

Pick to Create or Edit button

click on the button and then pick a string.

If the picked string already has attributes set by this option then those values will be displayed and can be edited.

If the picked string has not yet had attributes set by this option AND **ADAC asset** *has a value, then a panel with the attributes that this option can set for the given* **ADAC asset** *is displayed and then created for the picked string.*

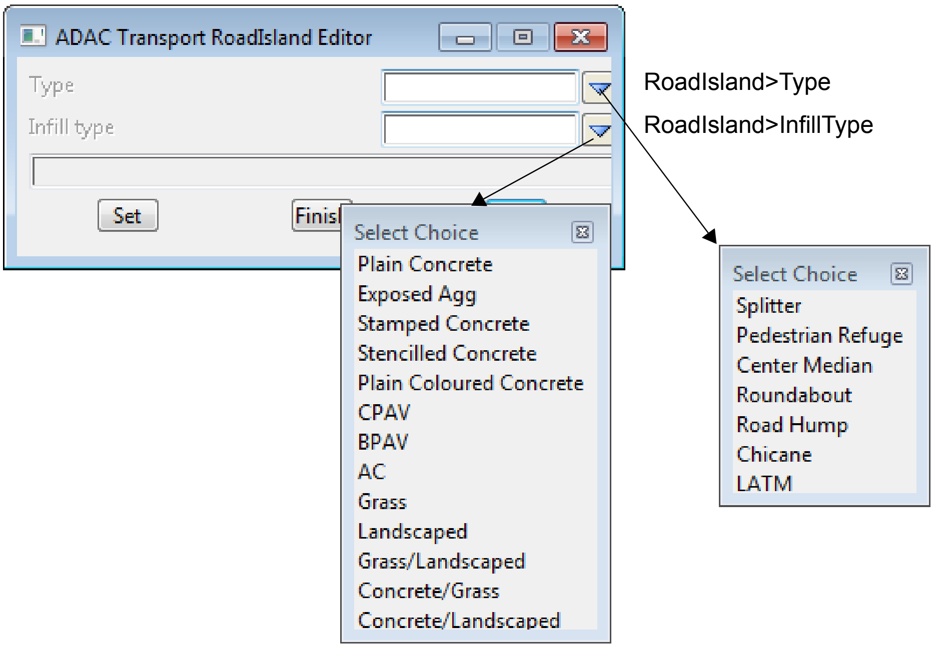
10.4.9.2.1.1 Transport RoadIsland
10.4.9.2.1.2 Transport Pavement
10.4.9.2.1.3 Transport Pathway
10.4.9.2.1.4 Transport Subsoil Drain
10.4.9.2.1.5 Stormwater Pit
10.4.9.2.1.6 Stormwater Surface Drain
10.4.9.2.1.7 Stormwater WSUD Area
10.4.9.2.1.8 Sewerage Maintenance Hole
10.4.9.2.1.9 Sewerage Pipe Non Pressure
10.4.9.2.1.10 Sewerage Pipe Pressure Editor
10.4.9.2.1.11 Sewerage Valve Editor
10.4.9.2.1.12 Sewerage Connection Editor
10.4.9.2.1.13 Water Supply Meter Editor
10.4.9.2.1.14 Water Supply Pipe Editor
10.4.9.2.1.15 Water Supply Hydrant Editor
10.4.9.2.1.16 Water Supply Valve Editor

10.4.9.2.1.1 Transport RoadIsland

The choice of ***Transport RoadIsland*** brings up a panel for setting parameters that will become the values for the ADAC attributes **Type** and **InfillType** for the ADAC Asset ***Transport>RoadIslands>RoadIsland.***

Because the option knows the *ADAC Schema* version and the *ADAC Asset,* the choices for **Type** and **InfillType** come from the *ADAC XSD*.

The selected string must be a Polygon with straight or arc segments.



Set writes the values in the panel as string attributes.

Clicking **Finish** removes the panel and replaces it with the ADAC Common Editor panel so another value of ADAC Asset can be chosen.

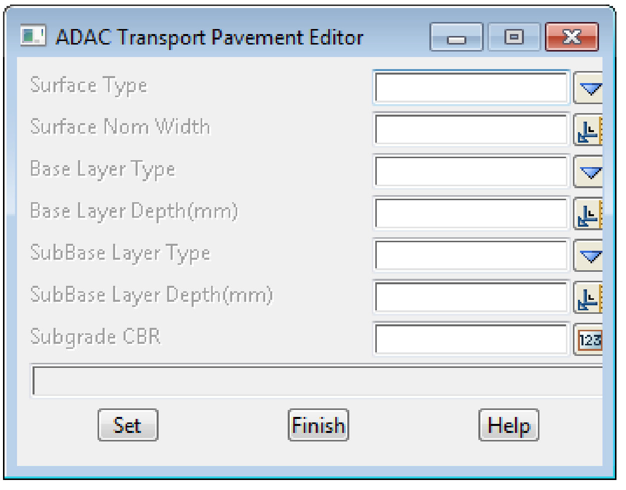
Continue to the next section 10.4.9.2.1.2 Transport Pavement or return to 10.4.9.2.1 Providing Extra Data for ADAC .

10.4.9.2.1.2 Transport Pavement

The choice of ***Transport Pavement*** brings up a panel for setting some of the parameters for the ADAC Asset ***Transport>Pavement Areas>Pavement.***

Because the option knows the ADAC version and the ADAC Asset, the types for the attributes and any pop-up choices all come from the ADAC XSD.

The selected string must be a Polygon with straight or arc segments.



Surface>SurfaceType
Surface>SurfaceNomWidth_m
PavementStructure>BaseLayer>LayerType
PavementStructure>BaseLayer>LayerDepth_mm
PavementStructure>SubBaseLayer>LayerType
PavementStructure>SubBaseLayer>LayerDepth_mm
Subgrade>CBR

Set writes the values in the panel as string attributes.

Clicking **Finish** removes the panel and replaces it with the ADAC Common Editor panel so another value of ADAC Asset can be chosen.

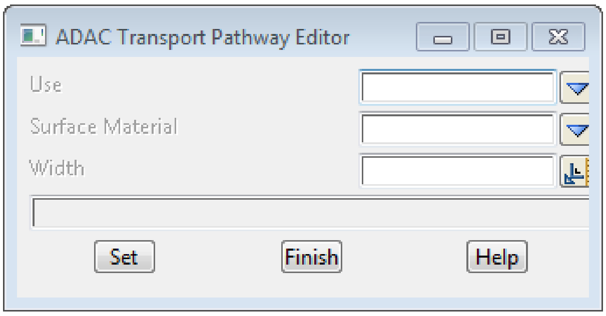
Continue to the next section 10.4.9.2.1.3 Transport Pathway or return to 10.4.9.2.1 Providing Extra Data for ADAC .

10.4.9.2.1.3 Transport Pathway

The choice of ***Transport Pathway*** brings up a panel for setting some of the parameters for the ADAC Asset ***Transport>Pathways>Pathway.***

Because the option knows the ADAC version and the ADAC Asset, the types for the attributes and any pop-up choices all come from the ADAC XSD.

The selected string must be a Polyline with straight or arc segments.



Pathway>Use

Pathway>SurfaceMaterial

Pathway>Width_m

Set writes the values in the panel as string attributes.

Clicking **Finish** removes the panel and replaces it with the ADAC Common Editor panel so another value of ADAC Asset can be chosen.

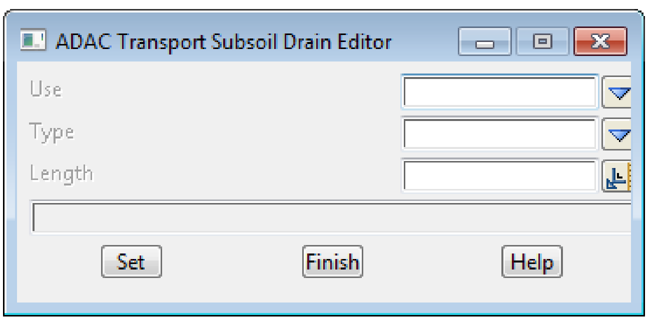
Continue to the next section 10.4.9.2.1.4 Transport Subsoil Drain or return to 10.4.9.2.1 Providing Extra Data for ADAC .

10.4.9.2.1.4 Transport Subsoil Drain

The choice of ***Transport Subsoil Drain*** brings up a panel for setting some of the parameters for the ADAC Asset ***Transport>SubSoilDrains>SubSoilDrain***

Because the option knows the ADAC version and the ADAC Asset, the types for the attributes and any pop-up choices all come from the ADAC XSD.

The selected string must be a Polyline with only straight segments.



SubSoilDrain>Use

SubSoilDrain>Type

SubSoilDrain>Lenght_m

Set writes the values in the panel as string attributes.

Clicking **Finish** removes the panel and replaces it with the ADAC Common Editor panel so another value of ADAC Asset can be chosen.

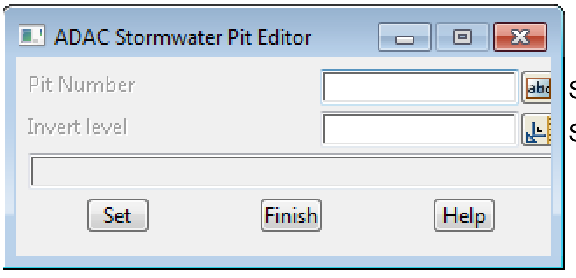
Continue to the next section 10.4.9.2.1.5 Stormwater Pit or return to 10.4.9.2.1 Providing Extra Data for ADAC .

10.4.9.2.1.5 Stormwater Pit

The choice of ***Stormwater Pit*** brings up a panel for setting some of the parameters for the ADAC Asset ***StormWater>Pits>Pit***

Because the option knows the ADAC version and the ADAC Asset, the types for the attributes and any pop-up choices all come from the ADAC XSD.

The selected string must be a one vertex string.



StormWater>Pits>PitNumber

StormWater>Pits>dimensions_point_levels_m>InvertLevel_m

Set writes the values in the panel as string attributes.

Clicking **Finish** removes the panel and replaces it with the ADAC Common Editor panel so another value of ADAC Asset can be chosen.

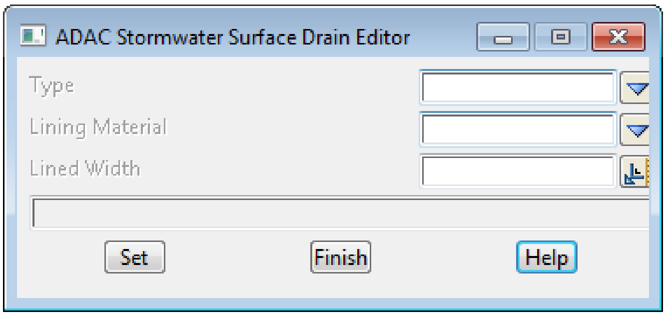
Continue to the next section 10.4.9.2.1.6 Stormwater Surface Drain or return to 10.4.9.2.1 Providing Extra Data for ADAC .

10.4.9.2.1.6 Stormwater Surface Drain

The choice of ***Stormwater Surface Drain*** brings up a panel for setting some of the parameters for the ADAC Asset ***StormWater>SurfaceDrains>SurfaceDrain***

Because the option knows the ADAC version and the ADAC Asset, the types for the attributes and any pop-up choices all come from the ADAC XSD.

The selected string must be a Polyline with only straight segments.



StormWater>SubSurfaceDrain>Type

StormWater>SubSurfaceDrain>LiningMaterial

StormWater>SubSurfaceDrain>LinedWidth_m

Set writes the values in the panel as string attributes.

Clicking **Finish** removes the panel and replaces it with the ADAC Common Editor panel so another value of ADAC Asset can be chosen.

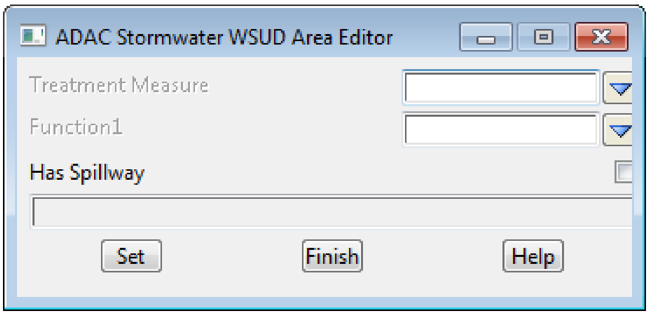
Continue to the next section 10.4.9.2.1.7 Stormwater WSUD Area or return to 10.4.9.2.1 Providing Extra Data for ADAC .

10.4.9.2.1.7 Stormwater WSUD Area

The choice of ***Stormwater WSUD Area*** brings up a panel for setting some of the parameters for the ADAC Asset ***StormWater>WSUDAreas>WSUDArea.***

Because the option knows the ADAC version and the ADAC Asset, the types for the attributes and any pop-up choices all come from the ADAC XSD.

The selected string must be a Polyline with straight or arc segments.



StormWater>WSUDArea>TreatmentMeasure

StormWater>WSUDArea>Function1

StormWater>WSUDArea>HasSpillway

Set writes the values in the panel as string attributes.

Clicking **Finish** removes the panel and replaces it with the ADAC Common Editor panel so another value of ADAC Asset can be chosen.

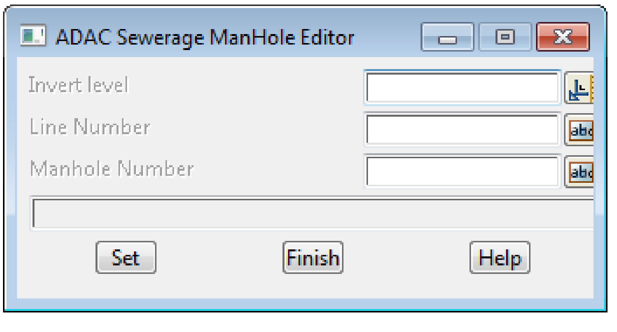
Continue to the next section 10.4.9.2.1.8 Sewerage Maintenance Hole or return to 10.4.9.2.1 Providing Extra Data for ADAC .

10.4.9.2.1.8 Sewerage Maintenance Hole

The choice of ***Sewerage Maintenance Hole*** brings up a panel for setting some of the parameters for the ADAC Asset ***Sewerage>MaintenanceHoles>MaintenanceHole***

Because the option knows the ADAC version and the ADAC Asset, the types for the attributes and any pop-up choices all come from the ADAC XSD.

The selected string must be a one vertex string.



MaintenanceHole>InvertLevel_m

MaintenanceHole>LineNumber

MaintenanceHole>MH_Number

Set writes the values in the panel as string attributes.

Clicking **Finish** removes the panel and replaces it with the ADAC Common Editor panel so another value of ADAC Asset can be chosen.

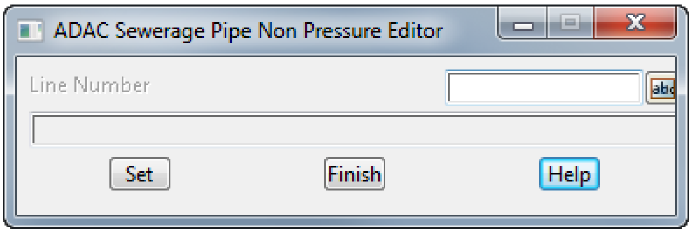
Continue to the next section 10.4.9.2.1.9 Sewerage Pipe Non Pressure or return to 10.4.9.2.1 Providing Extra Data for ADAC .

10.4.9.2.1.9 Sewerage Pipe Non Pressure

The choice of ***Sewerage Pipe Non Pressure*** brings up a panel for setting some of the parameters for the ADAC Asset ***Sewerage>PipesNonPressure>PipeNonPressure***

Because the option knows the ADAC version and the ADAC Asset, the types for the attributes and any pop-up choices all come from the ADAC XSD.

The selected string must be a polyline, including arcs.



PipeNonPressure >LineNumber

Set writes the values in the panel as string attributes.

Clicking **Finish** removes the panel and replaces it with the ADAC Common Editor panel so another value of ADAC Asset can be chosen.

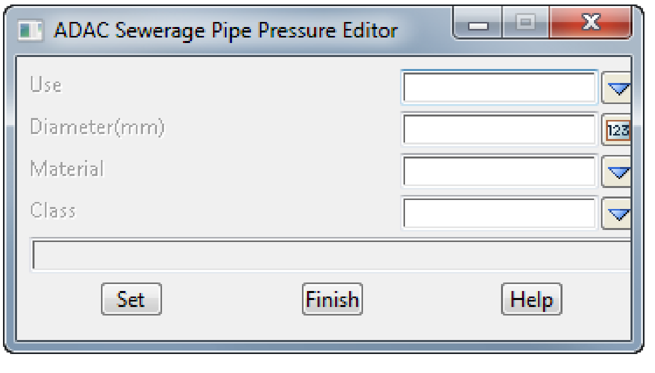
Continue to the next section 10.4.9.2.1.10 Sewerage Pipe Pressure Editor or return to 10.4.9.2.1 Providing Extra Data for ADAC .

10.4.9.2.1.10 Sewerage Pipe Pressure Editor

The choice of ***Sewerage Pipe Pressure*** brings up a panel for setting some of the parameters for the ADAC Asset ***Sewerage>PipesPressure>PipePressure***

Because the option knows the ADAC version and the ADAC Asset, the types for the attributes and any pop-up choices all come from the ADAC XSD.

The selected string must be a polyline which may include arcs.



PipePressure>Use

PipePressure>Diameter_mm

PipePressure>Material

PipePressure>Class

Set writes the values in the panel as string attributes.

Clicking **Finish** removes the panel and replaces it with the ADAC Common Editor panel so another value of ADAC Asset can be chosen.

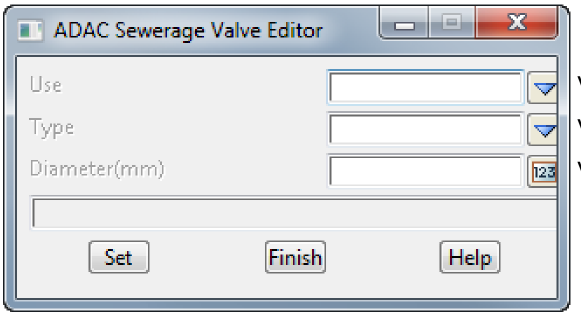
Continue to the next section 10.4.9.2.1.11 Sewerage Valve Editor or return to 10.4.9.2.1 Providing Extra Data for ADAC .

10.4.9.2.1.11 Sewerage Valve Editor

The choice of ***Sewerage Valve*** brings up a panel for setting some of the parameters for the ADAC Asset ***Sewerage>Valves>Valve***

Because the option knows the ADAC version and the ADAC Asset, the types for the attributes and any pop-up choices all come from the ADAC XSD.

The selected string must be a one vertex string.



Valve>Use

Valve>Type

Valve>Diameter_mm

Set writes the values in the panel as string attributes.

Clicking **Finish** removes the panel and replaces it with the ADAC Common Editor panel so another value of ADAC Asset can be chosen.

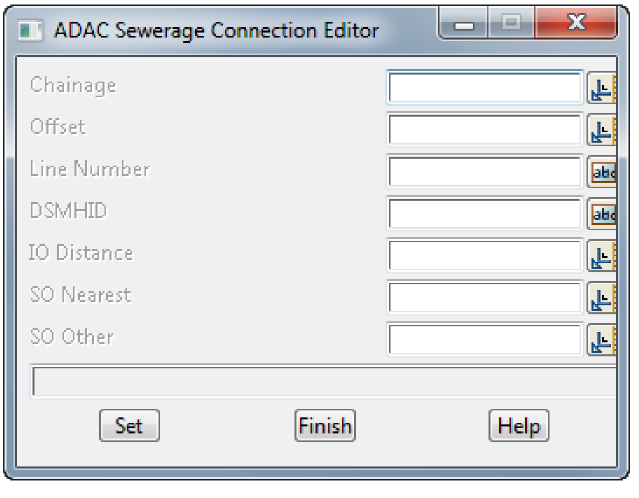
Continue to the next section 10.4.9.2.1.12 Sewerage Connection Editor or return to 10.4.9.2.1 Providing Extra Data for ADAC .

10.4.9.2.1.12 Sewerage Connection Editor

The choice of ***Sewerage Connection*** brings up a panel for setting some of the parameters for the ADAC Asset ***Sewerage>Connections>Connection***

Because the option knows the ADAC version and the ADAC Asset, the types for the attributes and any pop-up choices all come from the ADAC XSD.

The selected string must be a polyline which may include arcs.



Connection>Chainage_m

Connection>Offset_m

Connection>LineNumber

Connection>DSMHID

Connection>IO_Distance_m

Connection>SO_Nearest_m

Connection>SO_Other_m

Set writes the values in the panel as string attributes.

Clicking **Finish** removes the panel and replaces it with the ADAC Common Editor panel so another value of ADAC Asset can be chosen.

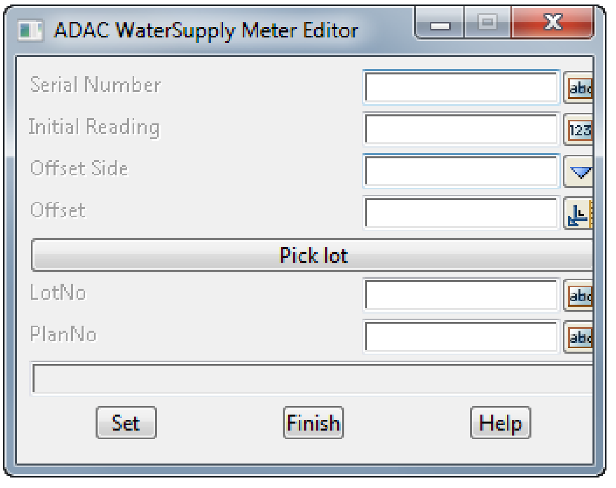
Continue to the next section 10.4.9.2.1.13 Water Supply Meter Editor or return to 10.4.9.2.1 Providing Extra Data for ADAC .

10.4.9.2.1.13 Water Supply Meter Editor

The choice of ***WaterSupply Meter*** brings up a panel for setting some of the parameters for the ADAC Asset ***WaterSupply>Meters>Meter***

Because the option knows the ADAC version and the ADAC Asset, the types for the attributes and any pop-up choices all come from the ADAC XSD.

The selected string must be a one vertex string.



Meter>SerialNumber

Meter>InitialReading

Meter>OffsetSide

Meter>Offset_m

Meter>Group: lot_plan_details>LotNo

Meter>Group: lot_plan_details>PlanNo

Pick lot - after clicking on **Pick lot,** if a polygon is selected that has Lot and Plan attributes set using the **Provide lot numbers, plan numbers and areas** option (see 10.4.9.2.2 Providing Lot and Plan Numbers and Areas for ADAC), then the values for LotNo and PlanNo are written into the **LotNo** and **PlanNo** fields.

Set writes the values in the panel as string attributes.

Clicking **Finish** removes the panel and replaces it with the ADAC Common Editor panel so another value of ADAC Asset can be chosen.

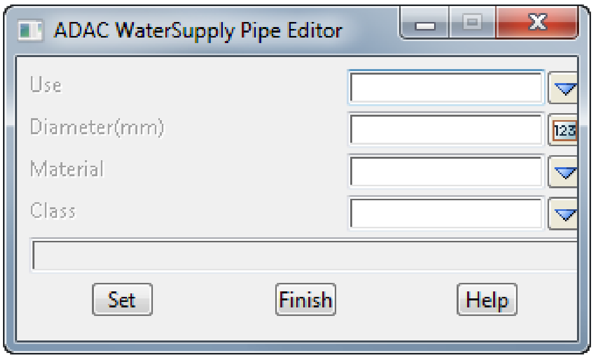
Continue to the next section 10.4.9.2.1.14 Water Supply Pipe Editor or return to 10.4.9.2.1 Providing Extra Data for ADAC .

10.4.9.2.1.14 Water Supply Pipe Editor

The choice of ***WaterSupply Pipe*** brings up a panel for setting some of the parameters for the ADAC Asset ***WaterSupply>Pipes>Pipe***

Because the option knows the ADAC version and the ADAC Asset, the types for the attributes and any pop-up choices all come from the ADAC XSD.

The selected string must be a polyline but only with straight segments.



Pipe>Use

Pipe>Diameter_mm

Pipe>Material

Pipe>Class

Set writes the values in the panel as string attributes.

Clicking **Finish** removes the panel and replaces it with the ADAC Common Editor panel so another value of ADAC Asset can be chosen.

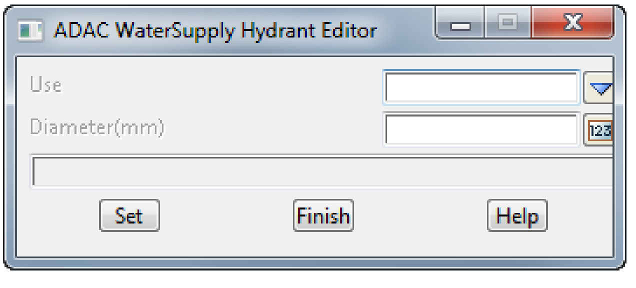
Continue to the next section 10.4.9.2.1.15 Water Supply Hydrant Editor or return to 10.4.9.2.1 Providing Extra Data for ADAC .

10.4.9.2.1.15 Water Supply Hydrant Editor

The choice of ***WaterSupply Hydrant*** brings up a panel for setting some of the parameters for the ADAC Asset ***WaterSupply>Hydrants>Hydrant***

Because the option knows the ADAC version and the ADAC Asset, the types for the attributes and any pop-up choices all come from the ADAC XSD.

The selected string must be a one vertex string.



Hydrant>Use

Hydrant>Diameter_mm

Set writes the values in the panel as string attributes.

Clicking **Finish** removes the panel and replaces it with the ADAC Common Editor panel so another value of ADAC Asset can be chosen.

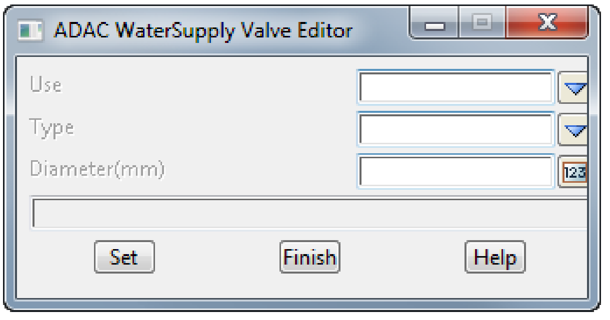
Continue to the next section 10.4.9.2.1.16 Water Supply Valve Editor or return to 10.4.9.2.1 Providing Extra Data for ADAC .

10.4.9.2.1.16 Water Supply Valve Editor

The choice of ***WaterSupply Valve*** brings up a panel for setting some of the parameters for the ADAC Asset ***WaterSupply>Valves>Valve***

Because the option knows the ADAC version and the ADAC Asset, the types for the attributes and any pop-up choices all come from the ADAC XSD.

The selected string must be a one vertex string.



Valve>Use

Valve>Type

Valve>Diameter_mm

Set writes the values in the panel as string attributes.

Clicking **Finish** removes the panel and replaces it with the ADAC Common Editor panel so another value of ADAC Asset can be chosen.

Continue to the next section 10.4.9.2.2 Providing Lot and Plan Numbers and Areas for ADAC or return to 10.4.9.2 Data Prep .

10.4.9.2.2 Providing Lot and Plan Numbers and Areas for ADAC

Position of menu:

File I/O =>ADAC =>User =>Data prep =>Providing lot numbers, plan numbers and areas

This option creates as string attributes: Plan numbers, Lot numbers and areas of lots. The area is calculated from the lot polygon, but that can be manually changed. However, having the actual area there to start with usually means that only the last couple of digits need to be entered.

The extra information is stored as attributes with the string, but not as ADAC attributes. A macro in the *ADAC Survey* or *ADAC Design* chain moves them into ADAC attributes. See (10.5.1.2.4 Update ADAC Elements from 12d Created Attributes on the String - Survey)

A major reason not to create ADAC attributes straightaway is that although this option allows you to enter the values, in the future the values might already exist in another form and routines written to automatically get them. In this case it will be much easier for anyone to put them into a simpler attribute structure and not have to know anything about the complex ADAC attribute structure.

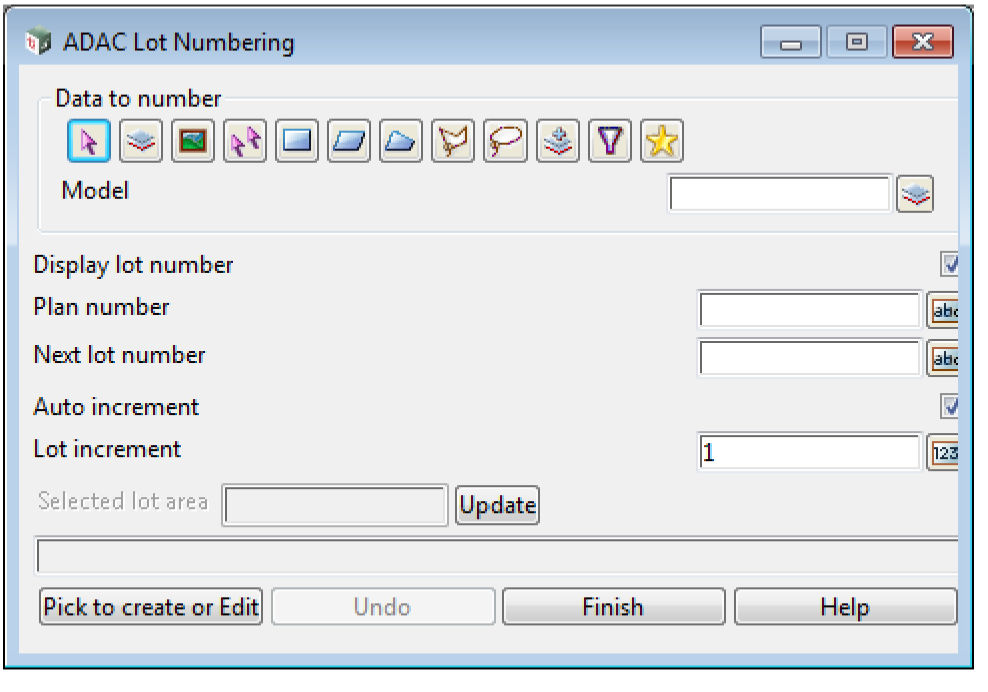
Another reason is that at this stage the *ADAC Survey* and *Design* chains have not been run to mark strings as ADAC Assets and set up the ADAC attributes group.

Also, when the *ADAC Survey* and *Design* chains are run, at each step in the chain they work with copies of the original data and clean out the models from a previous run rather than updating the original data. This is for safety so that it is easy to see that each step in the chain has worked, and if there is a problem, the chains can be run again and again.

So placing the attributes on the original data ensures they will still be there when the *ADAC Survey* or *Design* chain is rerun.

One important thing to keep in mind when running this option is that although you are using it to set up Lot attributes that will go to ADAC, when you are picking the strings they are not yet marked as ADAC *Cadastre>LandParcels>Lot* Assets. So you have to know **what ADAC Asset** that the string is **going to become** in ADAC *Cadastre>LandParcels>Lot***.**

Selecting **Providing lot number, plan numbers and areas** brings up the **Lot Numbering and Areas** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source .

Data source input

source of data that is be searched for any existing lot numbers and if they exist, the minimum and maximum existing lot numbers are written to the panel's message area and the highest lot number plus the **Lot increment** *is written to the* **Next lot no.** *field.*

If a Model is selected by MB same as, you need to type <Enter> after it for the search to go ahead.

Display lot number tick box tick

if ***ticked****, the lot numbers for existing lots are temporarily displayed on the view.*

If ***not ticked,*** *the lot numbers are not temporarily displayed on the view.*

Plan number text box

number of the plan the lots are from.

Next lot number text box

the next lot number to use.

Auto increment tick box tick

if ***ticked****, as each lot number is labelled, the* **Next lot number** *is incremented by the* **Lot increment**

If ***not ticked,*** *the* **Next lot number** *field is not incremented.*

Lot increment number box 1

the amount to increment the **Next lot number** *by.*

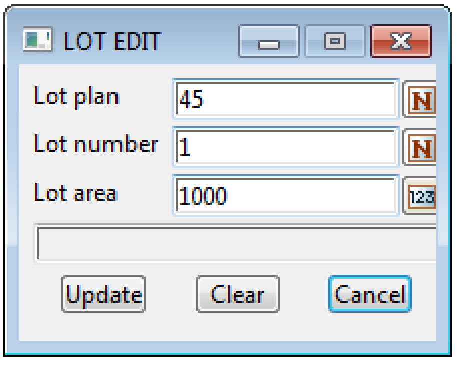
Selected lot area real box

when a string that is not already a lot is selected, the area of the string is calculated and written to this field. If a different area is required, change the value in this field and click **Update**.

Pick to Create or Edit button

click on the button and then pick a string.

If the picked string already has Lot attributes set by this option (or other options) then those values will be displayed in the **Lot Edit** *panel, and modified and saved by clicking* **Update** *on that panel.*



If the picked string has not yet had Lot attributes set by this option then the string will be given Lot attributes using the **Plan number**, **Next lot number** *and* **Selected lot area** *fields.*

Continue to the next section 10.4.9.2.3 Generate ADAC Road Edge Types or return to 10.4.9.2 Data Prep .

10.4.9.2.3 Generate ADAC Road Edge Types

Position of menu:

File I/O =>ADAC =>User =>Data prep =>Generate ADAC Road Edge types

There is an ADAC Asset *Transport>RoadEdges>RoadEdge* and it has the element **Type** with the choices

Independent Simple Data Type: **transport_edge_type** Back to Root Element

Constrains: Feature_Transport_RoadEdge.Type

Type of Road Edge - As per DMR drawing 1033

Restriction of: [String_32]

Enumeration:	B1	*Barrier Kerb Type 5*
Enumeration:	B2	*Barrier Kerb with Channel Type 6*
Enumeration:	B3	*Barrier Kerb with Channel Type 7*
Enumeration:	B4	*Barrier Kerb with Tray Type 23*
Enumeration:	B5	*Barrier Kerb with Tray Type 24*
Enumeration:	SM1	*Semi-Mountable Kerb Type 8*
Enumeration:	SM2	*Semi-Mountable Kerb Type 10*
Enumeration:	SM3	*Semi-Mountable Kerb Type 12*
Enumeration:	SM4	*Semi-Mountable Kerb with Channel Type 14*
Enumeration:	SM5	*Semi-Mountable Kerb with Channel Type 15*
Enumeration:	M1	*Mountable Kerb and Channel*
Enumeration:	M2	*Mountable Kerb and Channel*
Enumeration:	M3	*Mountable Kerb and Channel*
Enumeration:	M4	*Mountable Kerb*
Enumeration:	M5	*Mountable Kerb*
Enumeration:	M6	*Mountable Kerb*
Enumeration:	ER1	*Edge Restraint*
Enumeration:	ER2	*Edge Restraint*
Enumeration:	ER3	*Edge Restraint*
Enumeration:	ER4	*Edge Restraint*
Enumeration:	ER5	*Edge Restraint*
Enumeration:	INV600	*Concrete Channel Type 22*
Enumeration:	INV900	*Concrete Channel Type 28*
Enumeration:	Bitumen	*Bitumen Road Edge treatment*
Enumeration:	Concrete	*Concrete Road Edge treatment*

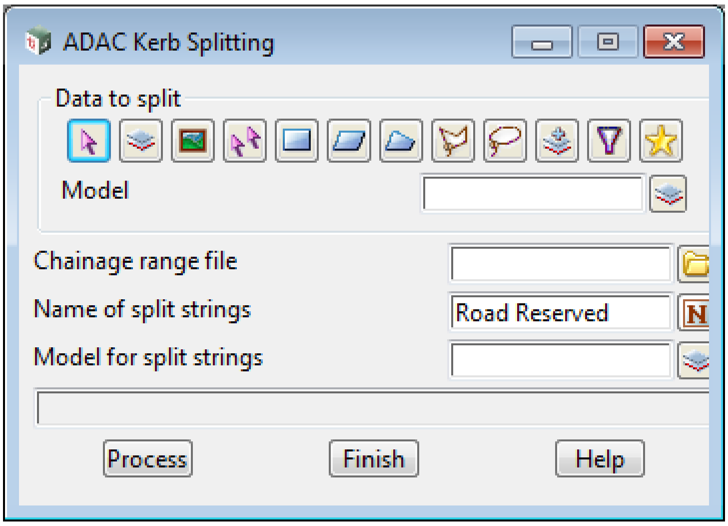
However, many companies do not have a naming convention that differentiates between the choices and may use just the one string name for all of them. So there is no way to automatically know the **Type** which is **mandatory** for an ADAC *Transport>RoadEdges>RoadEdge* Asset.

To help get over this problem, this option takes a file which contains any number of string names and for each string, a list of chainage ranges and kerb types.

For each string listed in this file, the option creates new strings for each of the chainage ranges. The new strings also have a User string attribute called **RoadEdge_Type** created, and it is given the value of **Type**.

It is the strings produced by this 12dPL, and not the original strings, that ADAC requires, and they are the ones added to the ***Design to map to ADAC*** or view ***Survey to map to ADAC*** for processing.

Selecting **Generate ADAC Road Edge types** brings up the **ADAC Kerb Splitting** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source .

Data source	input		

source of strings that is searched for strings to split.

Chainage range file	file box		

the file giving the names of the strings to be split and the chainage ranges to split the string into and the Type that goes with that chainage range. Each new string is given a text string attribute called RoadEdge_Type *which has the given kerb_type_i.*

In the file, the format for giving the chainage ranges and kerb types for a string is:

STRING *model_name ->string_name*
start_chainage_1 end_chainage_1 kerb_type_1
start_chainage_2 end_chainage_2 kerb_type_2
start_chainage_i end_chainage_i kerb_type_i

For example,

STRING MC01 DESIGN->KIR
0 100 M1
100 9999 B1

Name of split strings	text box		

*if **blank**, the original string name is used for the new strings.*
*If **not blank**, the created split strings will be given this name.*

Model for split strings	model box		available models

the model for the created split strings.

Process	button		

go and create the split strings.

Continue to the next section 10.4.9.2.4 Create Points from Other Data or return to 10.4.9.2 Data Prep .

10.4.9.2.4 Create Points from Other Data

Position of menu:

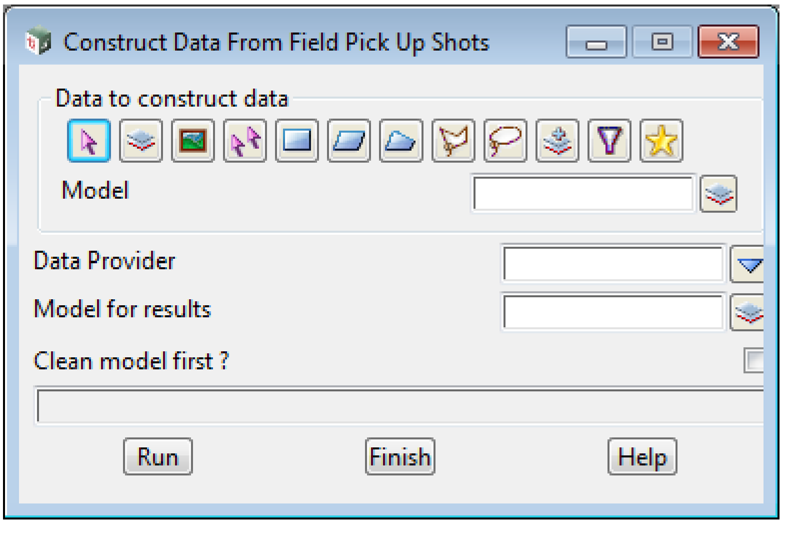
File I/O =>ADAC =>User =>Data prep =>Create points from other data

Sometimes strings are not suitable to go straight to ADAC but can be used to create suitable strings without the user having to do anything manually.

For example, ADAC only has a one point object for signs and a user picks up large signs as a two vertex string. This option takes the two vertex large sign and creates a new one vertex string that **can** be mapped to ADAC as an ADAC sign.

This option has various methods of creating ADAC suitable data.

Selecting **Create points from other data** brings up the **Construct Data from Field Pick Up Shots** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source .

Data source	input		

source of strings that is process to generate strings suitable for ADAC.

Data Provider	choice box		available data providers

selects the table of modifications to make.

Model for results	model box		available models

model that the processed strings are added to.

Clean model first ?	tick box		

*if **ticked**, the model* **Model for results** *is cleaned before any strings are added to it.*
*If **not ticked**, the model* **Model for results** *is NOT cleaned.*

Run	button		

process the data.

Continue to the next section 10.4.9.2.5 Create Pipes from Points or return to 10.4.9.2 Data Prep
.

10.4.9.2.5 Create Pipes from Points

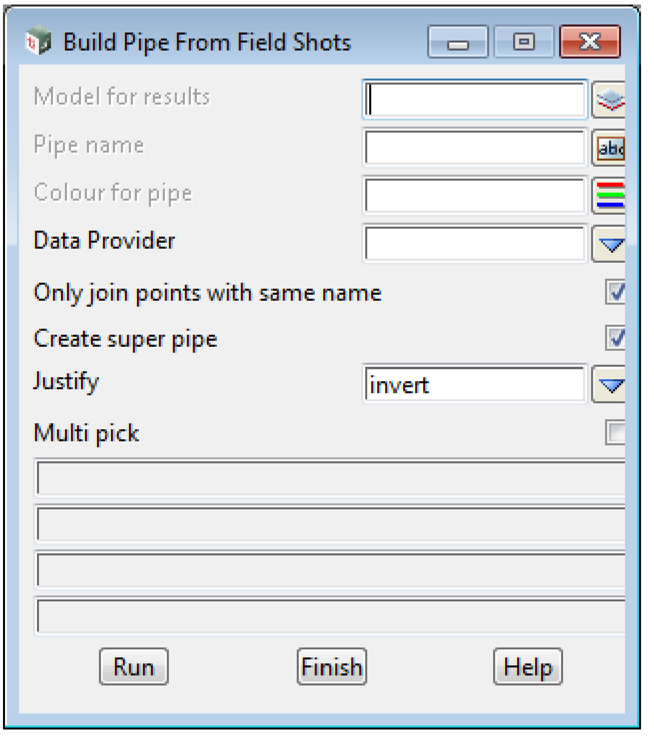
Position of menu:

File I/O =>ADAC =>User =>Data prep =>Create pipes from points

Sometimes pipes can not be picked up as string in the field and only one vertex strings are picked up at say the ends of the pipes.

This option joins two selected one vertex strings to create a two vertex pipe string.

Selecting **Create pipes from points** brings up the **Build Pipe from Field Shots** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model for results	model box		available models

model that the create pipes are added to.

Pipe name	text box		

*if **not blank**, the created string is given this name.*
If blank,

Colour for pipe	colour box		available colours

*if **not blank**, the created string is given this colour.*
If blank,

Data Provider	choice box		available data providers

selects the table of attributes to use for diameter etc.

Only join points with same name tick box

*if **ticked**, after picking the first one vertex string, the second one vertex string must have the same name as the first picked string.*

Create super pipe tick box

*if **ticked**, and there is an attribute specified in the selected* **Data provider** *for diameter, then a super string pipe with the given diameter is created.*

Justify	choice box	invert, centre, obvert

justification to use if a super pipe is created.

Multi pick	tick box

*if **ticked**, once one pipe is created, the option starts asking for the first point of a new pipe.*

Run	button

start the option.

Continue to the next section 10.4.9.2.6 Set Stormwater/Sewer Property or return to 10.4.9.2 Data Prep .

10.4.9.2.6 Set Stormwater/Sewer Property

Position of menu:

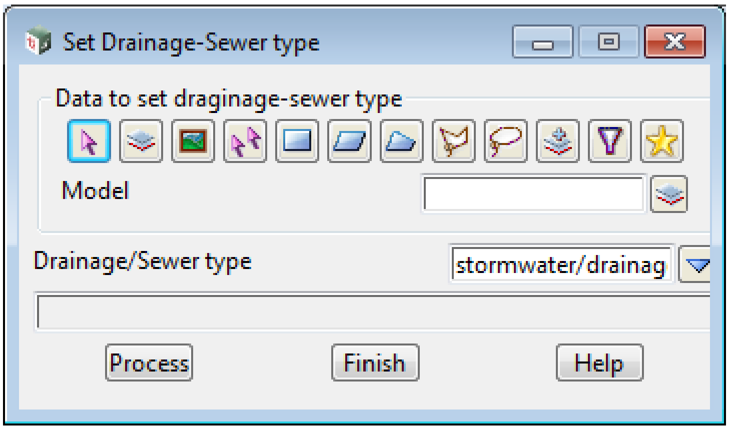
File I/O =>ADAC =>User =>Data prep =>Set stormwater/sewer purpose

Up to ***12d Model 10***, there was no way of easily differentiating between a drainage or sewer string. Users normally kept them in different models.

In ***12d Model 11***, there is a string property (called **Purpose** on the **Create Drainage String** menu) which is set to either **stormwater/drainage** or **wastewater/sewer**. So the string itself now knows if it is a drainage or a sewer string.

This option sets the **Purpose** for drainage/sewer strings that have come from ***12d Model 10*** and so do not have a **Purpose** already set, or may have it defaulted to **stormwater/drainage**.

Selecting **Set stormwater/sewer purpose** brings up the **Set Drainage/Sewer Purpose** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	
data selection type - for a full description go to 4.19.3 Data Source .			
Data source	input		
source of strings to process.			
Drainage/Sewer purpose	choice box	stormwater/drainage, wastewater/sewer	
the Purpose property of the selected strings are set to this value.			
Run	button		
set the Purpose property of the selected strings.			

Continue to the next section 10.4.9.2.7 Set a Drainage-Sewer Pit and Pipe Type Attribute or return to 10.4.9.2 Data Prep .

10.4.9.2.7 Set a Drainage-Sewer Pit and Pipe Type Attribute

Position of menu:

File I/O =>ADAC =>User =>Data prep =>Set pit type, pipe type attribute

When drainage and sewer strings are created, the vertices can have a **Pit Type** and the segments a **Pipe Type**.

When the Drainage Network Editor (DNE) is used, for each vertex, a vertex attribute called **pit_type** is created with the vertex value of **Pit Type**. Similarly for each segment, a segment attribute called **pipe_type** is created with the segment value of **Pipe Type**.

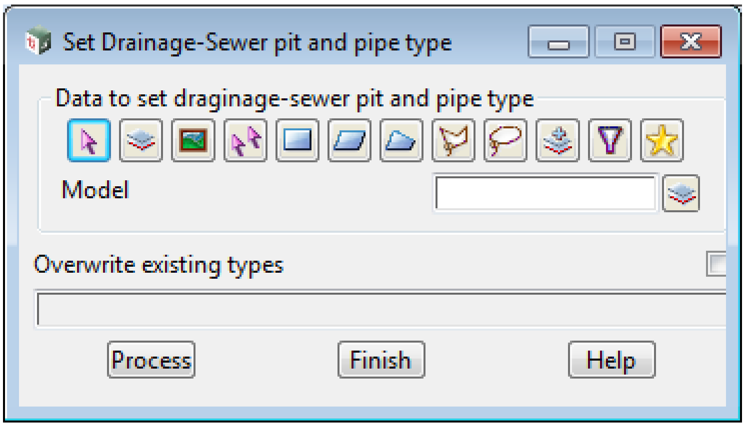
So after using the DNE, the attributes pit_type and pipe_type are available to be used in the ADAC ***Map File***. However if the DNE is not run, the attributes pit_type and pipe_type will not exist.

This option does the same thing as the DNE for creating the **pit_type** and **pipe_type** attributes.

That is, for each vertex, a vertex attribute called **pit_type** is created with the vertex value of **Pit Type** and for each segment, a segment attribute called **pipe_type** is created with the segment value of **Pipe Type**.

So in case th DNE has not been run, this option creates the attributes **pit_type** and **pipe_type** so they can be used in the ***Map File***.

Selecting **Set pit type, pipe type attribute** brings up the **Set Attribute for Drainage/Sewer Pit and Pipe Types** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source .

Data source input

source of strings to process.

Overwrite existing type tick box

*if **ticked** and a pit_type/pipe_type already exists for a vertex/segment, then it is over written by the value of Pit type/Pipe Type for the vertex/segment.*

*if **not ticked** then if a pit_type/pipe_type already exists for a vertex/segment, then it is **not** modified.*

Run button

set the pit_type and pipe_type's of the selected strings vertices and segments.

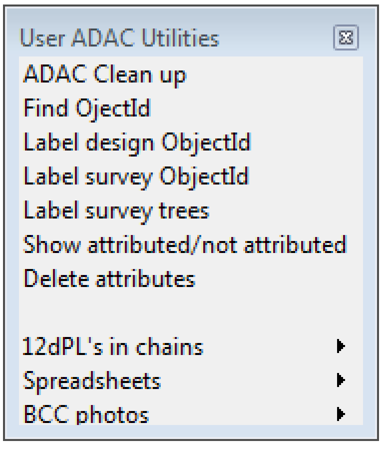
Return to 10.4.9.2 Data Prep or 10.4.9 User ADAC .

10.4.9.3 User ADAC Utilities

Position of menu: **File I/O =>ADAC =>User =>Utilities**

A new project to be set up for producing ADAC XML needs to be set up a certain way for the ADAC Chains to work.

There is a different setup for a **Survey** Project than for a **Design** Project.



See

10.4.9.3.1 ADAC Clean Up

10.4.9.3.2 Find Object Id

10.4.9.3.3 Label Design/Survey Object Id

10.4.9.3.4 Show Attributed/Not Attributed

10.4.9.3.5 Delete ADAC Attributes

10.4.9.3.6 12dPL's in Chains

10.4.9.3.1 ADAC Clean Up

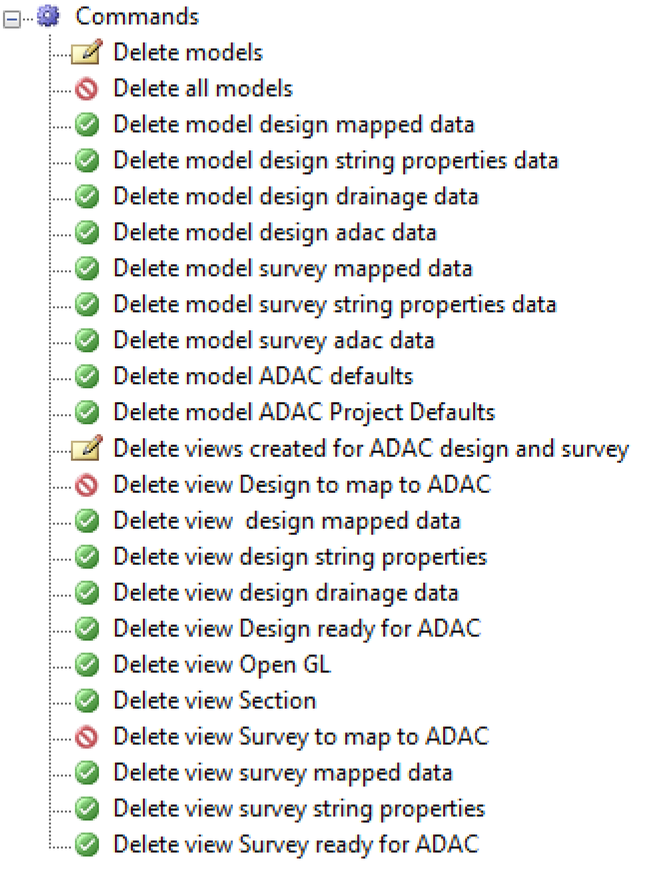
Position of option on menu: **File I/O =>ADAC =>User =>Utilities =>ADAC clean up**

This option runs a chain that deletes all the models created by the **Setting up for survey** and **Setting up for design** options.

The chain also deletes all the views, except for the views *Design to map to ADAC* and *Survey to map to ADAC*, created by the **Setting up for survey** and **Setting up for design** options.

Selecting **ADAC clean up** runs the chain

$LIB/ADAC_clean_up.chain



The chain needs no interaction with the user; it just runs and ends.

Continue to the next section 10.4.9.3.2 Find Object Id or return to 10.4.9.3 User ADAC Utilities .

10.4.9.3.2 Find Object Id

Position of option on menu: **File I/O =>ADAC =>User =>Utilities =>Find Objectid**

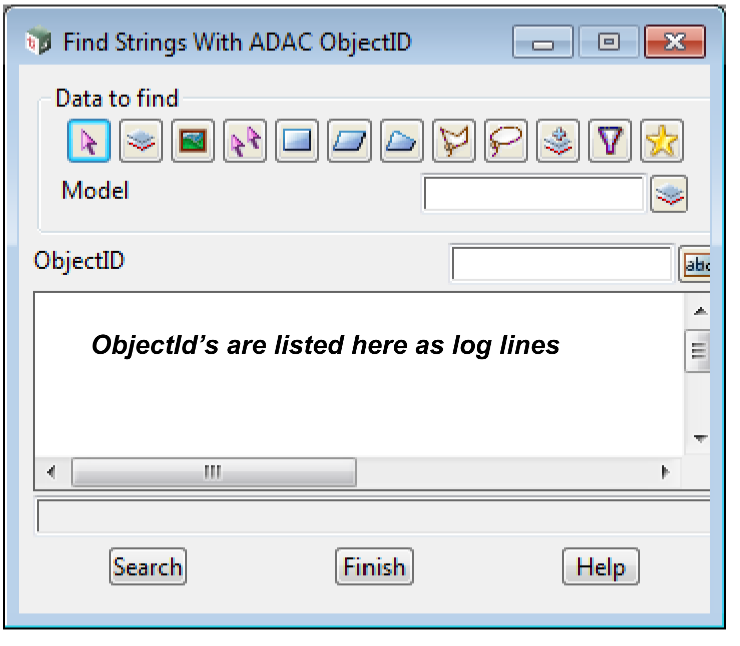
From the *ADAC XML Schema, the* **Objectid**

> *Represents a place for an object identifier, usually generated by the data creation software. Also useful as a placeholder for record ID generated by back-end databases or GIS*
> *The ObjectId is permitted to be nil because the capturing system may not have the capacity to generate one. If features are exported from a GIS or Asset Management system, however, it is highly recommended to carry them out into this element.*

When the *ADAC Survey* and *ADAC Design* chains are run, the ADAC Assets are all given a unique ADAC Objectid by ***12d Model*** and this is given in the ADAC Report (see 10.4.5 Report) and any ADAC XML file that is created (see 10.4.6 Write ADAC XML File).

The **Find Objectid** option lists as log lines the ObjectId of all ADAC Assets whose ObjectIds start with given text. Clicking on an ObjectId in the list highlights it in any Plan view that it is on.

Selecting **Find Objectid** brings up the **Find Strings with ADAC ObjectId** panel:



Data source type Model

> *data selection type - for a full description go to* 4.19.3 Data Source .

Data source input

> *source of data to be searched for any with ObjectIds that start with the characters in* **ObjectId start characters**.

ObjectId start characters text box

> *the characters the ObjectIDs of the strings must start with.*

List Area

> *the ObjectID of each string that starts with the characters in the* **ObjectId start characters** *is listed in this area as a log line. Clicking on an ObjectId in the list will pan to and highlight the string in any Plan views that the string is on.*

Search button

> *search for all ADAC strings that start with the characters in* **ObjectId start characters**.

Continue to the next section 10.4.9.3.3 Label Design/Survey Object Id or return to 10.4.9.3 User ADAC Utilities .

10.4.9.3.3 Label Design/Survey Object Id

[check why it is not working]

Position of option on menu: **File I/O =>ADAC =>User =>Utilities =>Label design Objectid**

Position of option on menu: **File I/O =>ADAC =>User =>Utilities =>Label survey Objectid**

From the *ADAC XML Schema, the* **Objectid**

> *Represents a place for an object identifier, usually generated by the data creation software. Also useful as a placeholder for record ID generated by back-end databases or GIS*
> *The ObjectId is permitted to be nil because the capturing system may not have the capacity to generate one. If features are exported from a GIS or Asset Management system, however, it is highly recommended to carry them out into this element.*

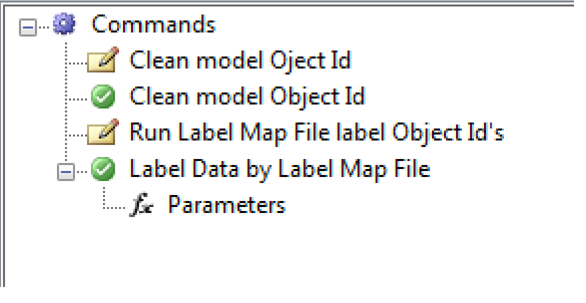
This option runs an Interactive chain that includes a **Label Data by Label Map File** panel that creates labels for all the ADAC assets in the model **design adac data** or **survey adac data** and cleans and adds them to the model **ObjectId**.

Note: the model **design adac data/survey adac data** is the model that the *ADAC Design/ Survey* chain puts all the ADAC Assets in.

Selecting **Label design/survevObjectid** runs the Interactive chain

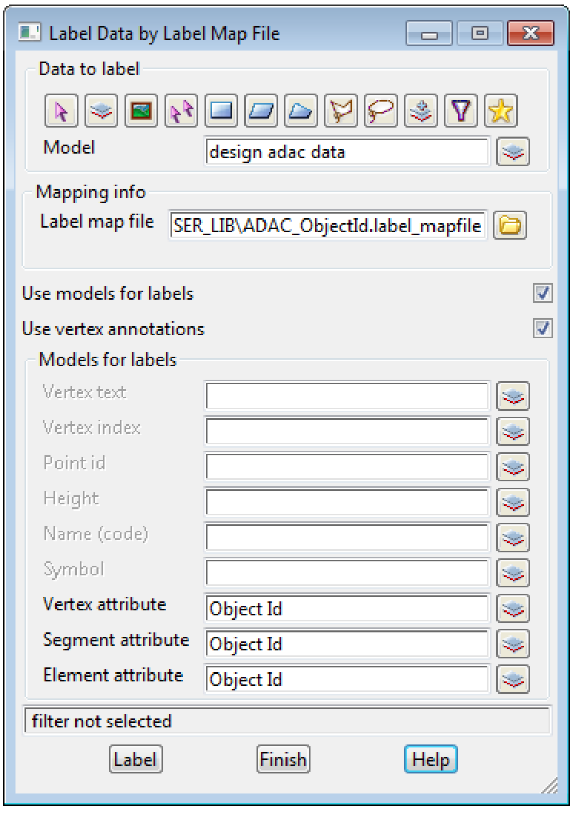
$LIB/ADAC_Label_design_ObjectId.chain

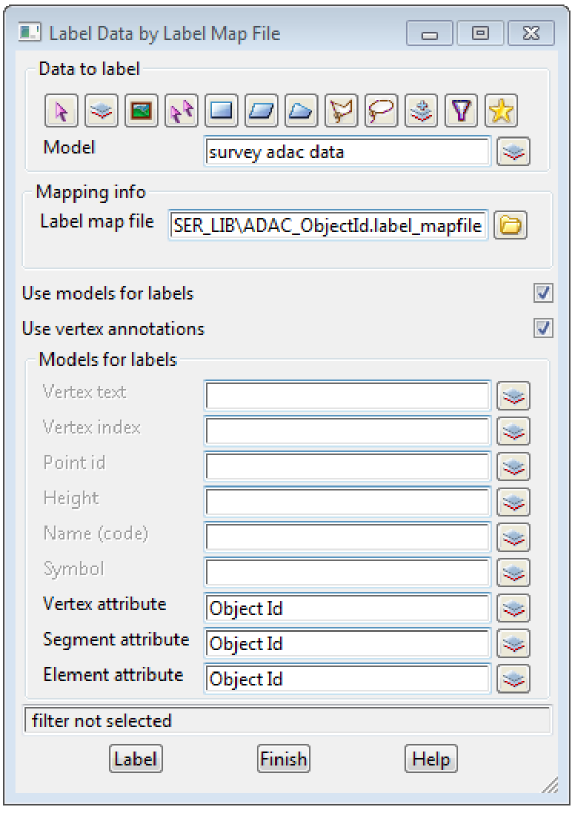
$LIB/ADAC_Label_survey_ObjectId.chain



The chain first cleans out the model **ObjectId**

It then runs a **Label Data by Label Map File** panel in Interactive mode so the panel is placed on the screen.





Click on **Label** and when the labelling is finished, click on **Finish**.

The chain then ends.

Continue to the next section 10.4.9.3.4 Show Attributed/Not Attributed or return to 10.4.9.3 User ADAC Utilities .

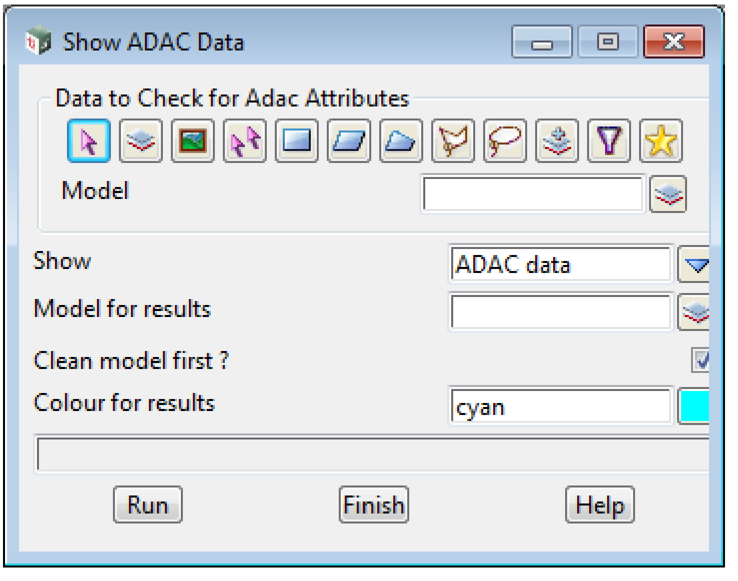
10.4.9.3.4 Show Attributed/Not Attributed

Position of option on menu: **File I/O =>ADAC =>User =>Utilities =>Show attributed/not attributed**

When checking whether your 12d Map File is doing the correct thing, it is often necessary to know which strings have been given an ADAC attribute group and hence represent an ADAC Asset, and which ones weren't.

This option searches for all strings that are either ADAC Assets, or are not ADAC Assets, and make copies of the string and adds them to a given model.

Selecting **Show attributes/not attributes** brings up the **Show ADAC Attributed Data** panel



Data source type Model

data selection type - for a full description go to 4.19.3 Data Source .

Data source input

source of data to be check if it has/doesn't have ADAC attributes.

Show choice box ADAC data ADAC data, non ADAC data

if ***ADAC data****, for each string in the data source that is an ADAC Asset (i.e. has the ADAC attributes), a copy of the string is made in the colour given in* **Colour for results** *and is added to the model in* **Model for results.**

if ***non ADAC data****, for each string in the data source that is NOT an ADAC Asset (i.e. doesn't have the ADAC attributes), a copy of the string is made in the colour given in* **Colour for results** *and is added to the model in* **Model for results.**

Model for results model box available models

the model for the created strings.

Clean model first tick box

if ***ticked****, the model* **Model for results** *is cleaned before any strings are added to it.*
If ***not ticked****, the model* **Model for results** *is NOT cleaned.*

Colour for results colour box cyan available colours

the colour to give the created strings.

Run button

finds all the ADAC/not ADAC strings.

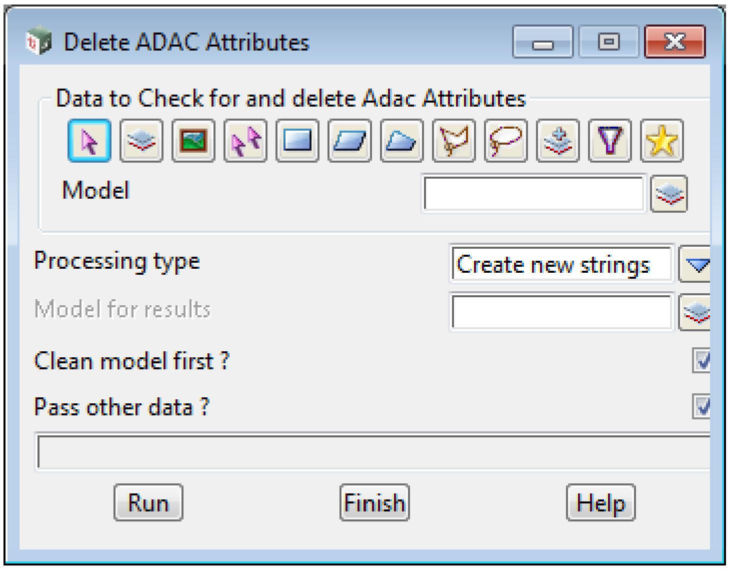
Continue to 10.4.9.3.5 Delete ADAC Attributes or return to 10.4.9.3 User ADAC Utilities .

10.4.9.3.5 Delete ADAC Attributes

Position of option on menu: **File I/O =>ADAC =>User =>Utilities =>Delete attributes**

This option removes the ADAC attributes from strings (and only those attributes).

Selecting **Delete attributes** brings up the **Delete ADAC Attributes** panel:



Data source type Model

data selection type - for a full description go to 4.19.3 Data Source .

Data source input

source of data to process for deleting ADAC attributes.

Processing type choice box Create new strings Use existing strings
Create new strings

*if **Create new strings**, for each string in the data source that is an ADAC Asset (i.e. has the ADAC attributes), a copy of the string is made and the ADAC attributes deleted from the **copy** of the string and added to the model* **Model for results.**

*if **Use existing strings**, for each string in the data source that is an ADAC Asset (i.e. has the ADAC attributes), the ADAC attributes are deleted from the string.*

Model for results model box available models

when **Processing type** *is **Create New Strings** this is the model for the created strings.*

Clean model first ? tick box

*if **ticked**, the model* **Model for results** *is cleaned before any strings are added to it.*
*If **not ticked**, the model* **Model for results** *is NOT cleaned.*

Pass other data ? tick box

*if **ticked** and* **Processing type** *is **Create new strings**, then any strings that are not ADAC Assets are also copied and added to the model* **Model for results.**
Otherwise nothing is done with strings that are not ADAC Assets.

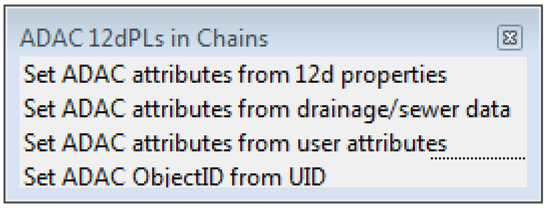
Run button

removed the ADAC attributes from the selected strings.

Continue to 10.4.9.3.6 12dPL's in Chains or return to 10.4.9.3 User ADAC Utilities .

10.4.9.3.6 12dPL's in Chains

Position of menu: **File I/O =>ADAC =>User =>Utilities**



There are the options used in the ADAC Design and Survey chains.

See

10.4.9.3.6.1 Separate Assigned/Unassigned ADAC Data

10.4.9.3.6.2 Set ADAC Attributes from String Properties

10.4.9.3.6.3 Set ADAC Attributes from Drainage & Sewer Data

10.4.9.3.6.4 Set ADAC Attributes from User Attributes

10.4.9.3.6.5 Set ADAC ObjectId from String UID

10.4.9.3.6.5 Set ADAC ObjectId from String UID

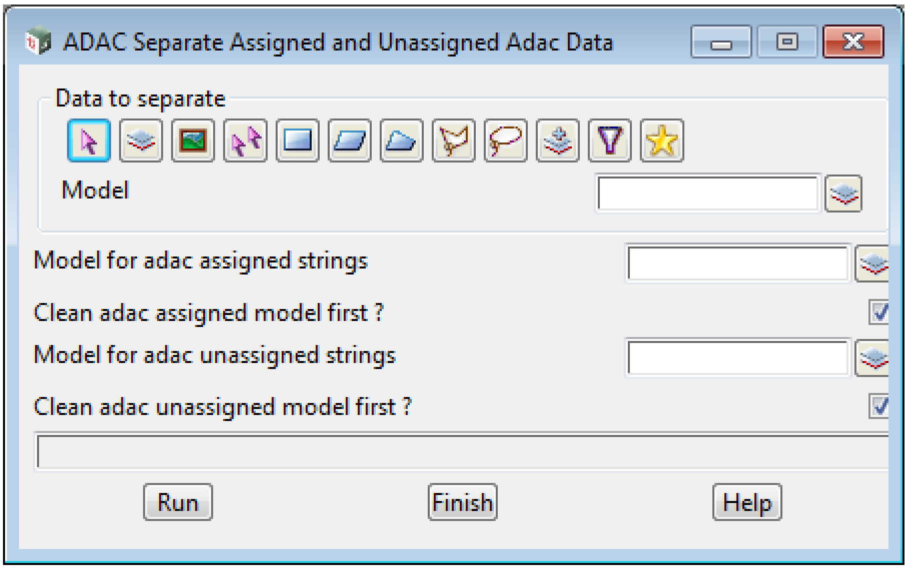
10.4.9.3.6.1 Separate Assigned/Unassigned ADAC Data

Position of option on menu:

File I/O =>ADAC =>User =>Utilities =>12dPLs in chains =>Separate assigned data

This option looks at the data in the given Data Source and copies the strings that have been assigned (marked) as ADAC Assets to one model and all the data that has not yet been assigned to another model.

Selecting **Separate Assigned data** brings up the **ADAC Separate Assigned and Unassigned Data** panel.



Data source type Model

data selection type - for a full description go to 4.19.3 Data Source .

Data source input

source of data to process.

Model for ADAC assigned strings model box available models

any strings that have been assigned (marked) as ADAC assets are copied to this model.

Clean ADAC assigned model first ? tick box

if ***ticked****, the model* **Model for ADAC assigned strings** *is cleaned before any strings are added to it.*
If ***not ticked****, the model* **Model for ADAC assigned strings** *is NOT cleaned.*

Model for ADAC unassigned strings model box available models

any strings that have NOT been assigned (marked) as ADAC assets are copied to this model.

Clean ADAC unassigned model first ? tick box

if ***ticked****, the model* **Model for ADAC unassigned strings** *is cleaned before any strings are added to it.*
If ***not ticked****, the model* **Model for ADAC assigned strings** *is NOT cleaned.*

Run button

separate the strings that have been assigned/unassigned as ADAC Assets into separate models.

The copied strings are added to the model ***Model for results****.*

Continue to the next section 10.4.9.3.6.2 Set ADAC Attributes from String Properties or return to 10.4.9.3.6 12dPL's in Chains

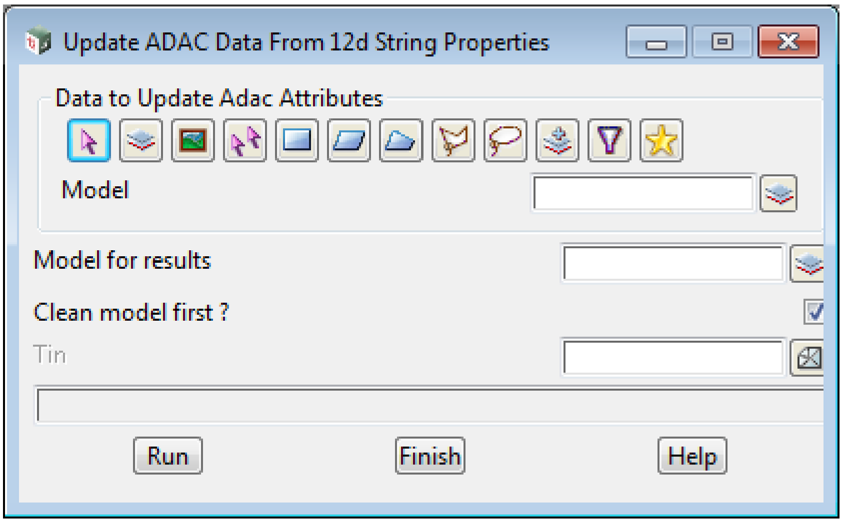
10.4.9.3.6.2 Set ADAC Attributes from String Properties

Position of option on menu:

File I/O =>ADAC =>User =>Utilities =>12dPLs in chains =>Set ADAC attributes from string properties

Many of the ADAC entities can be calculated directly from the 12d strings and some z values may be taken from a tin.

Selecting **Set ADAC attributes from 12d properties** brings up the **Update ADAC Data from 12d String Properties** panel.



Data source type Model

data selection type - for a full description go to 4.19.3 Data Source .

Data source input

source of data to process.

Model for results model box available models

model that the processed strings are added to.

Clean model first ? tick box

*if **ticked**, the model* **Model for results** *is cleaned before any strings are added to it.*
*If **not ticked**, the model* **Model for results** *is NOT cleaned.*

Tin tin box available tins

tin that can be used to get z-values from.

Run button

update the ADAC Asset attributes with values from the 12d string's properties.

If the string has been marked as an ADAC Asset then it is copied and then for the copied string, any relevant ADAC elements updated from the properties of the 12d string and the tin.

Strings that are not marked as ADAC Assets are NOT copied

*The copied strings are added to the model **Model for results**.*

ADAC Attributes Generated from 12d String Geometry

ADAC path	Action	Feature name	Nillability
Adac/Contour/Elevation_m	z value from string	Surface	No
Adac/SpotHeight/Elevation_m	z value from vertex	Surface	No
Adac/RoadEdge/Length_m	calculate 3d length of string	Transport	Yes
Adac/SubSoilDrain/Length_m	calculate 3d length of string	Transport	Yes
Adac/RoadIsland/Area_sqm	calculate area of string	Transport	Yes
Adac/Parking/Surface/SurfaceArea_sqm	calculate plan area of a string	Transport	Yes
Adac/BarrierContinuous/Length_m	calculate 3d length of string	OpenSpace	No
Adac/RetainingWall/Length	calculate 3d length of string	OpenSpace	No
Adac/ElectricalConduit/Size	take diameter from super string	OpenSpace	No
Adac/ElectricalConduit/Diameter_mm	take diameter from super string	OpenSpace	No
Adac/ElectricalConduit/Length_m	calculate 3d length of string	OpenSpace	No
Adac/Pipe/Diameter_mm	take diameter from super string	WaterSupply	No
Adac/Pipe/Length_m	calculate 3d length of string	WaterSupply	Yes
Adac/Maintenancehole/SurfaceLevel_m	z value from string	WaterSupply	No
Adac/Manhole/SurfaceLevel_m	z value from string	Sewerage	No
Adac/Maintenancehole/SurfaceLevel_m	z value from string	Sewerage	No
Adac/Connection/HCB_Length	calculate 3d length of string	Sewerage	No
Adac/Connection/Length_m	calculate 3d length of string	Sewerage	No
Adac/Connection/InvertLevel_m	calculate us invert level (higher end)	Sewerage	No
Adac/Connection/SurfaceLevel_m	calculate tin us surface level (higher end, tin required)	Sewerage	No
Adac/PipeNonPressure/DS_SurfaceLevel_m	calculate tin ds surface level (higher end, tin required)	Sewerage	No
Adac/PipeNonPressure/AverageDepth_m	get average depth based on other attributes	Sewerage	No
Adac/PipeNonPressure/PipeGrade	calculate grade	Sewerage	yes
Adac/PipeNonPressure/Length_m	calculate 3d length	Sewerage	yes
Adac/PipeNonPressure/US_InvertLevel_m	calculate us invert level (higher end)	Sewerage	no
Adac/PipeNonPressure/DS_InvertLevel_m	calculate ds invert level (lower end)	Sewerage	no
Adac/PipeNonPressure/Diameter_mm	calculate diameter	Sewerage	no
Adac/Pipe/US_SurfaceLevel_m	calculate tin us surface level (higher end, tin required)	StormWater	No
Adac/Pipe/DS_SurfaceLevel_m	calculate tin ds surface level (higher end, tin required)	StormWater	No
Adac/Pit/Depth_m	calculate stormwater pit depth from other attributes	StormWater	No

4.0.0 & 4.1.0	Adac/Pipe/Length_m	calculate 3d length	Storm\
4.0.0 & 4.1.0	Adac/Pipe/Grade	calculate grade	Storm\
4.0.0 & 4.1.0	Adac/SurfaceDrain/Length_m	calculate 3d length	Storm\
4.0.0 & 4.1.0	Adac/Pipe/US_InvertLevel_m	calculate us invert level (higher end)	Storm\
4.0.0 & 4.1.0	Adac/Pipe/DS_InvertLevel_m	calculate ds invert level (lower end)	Storm\
4.0.0 & 4.1.0	Adac/Pipe/PipeStructure/CircPipe/Diameter_mm	calculate circ diameter	Storm\
4.0.0 & 4.1.0	Adac/Pipe/PipeStructure/BoxPipe/Height_mm	calculate box height	Storm\
4.0.0 & 4.1.0	Adac/Pipe/PipeStructure/BoxPipe/Width_mm	calculate box width	Storm\
4.0.0 & 4.1.0	Adac/SurfaceDrain/Length_m	calculate 3d length of string	Storm\
4.0.0 & 4.1.0	Adac/Annotation/Text	get the actual text of the text string	Enhance
4.0.0 & 4.1.0	Adac/Annotation/Justification	get justification of text string	Enhance
4.0.0 & 4.1.0	Adac/Annotation/Type	get text type of text string	Enhance
4.0.0 & 4.1.0	Adac/Annotation/Font	get text font of text string	Enhance
4.0.0 & 4.1.0	Adac/Annotation/Height_m	get text height of text string	Enhance
4.0.0 & 4.1.0	Adac/Annotation/Width_m	get text width of text string	Enhance
4.0.0 & 4.1.0	Adac/Annotation/Rotation	get text rotation of text string	Enhance

In the *ADAC Survey* and *Design* chains, this option is called and passed arguments for all the panel field and so runs without needing to display a panel.

See 10.5.1.2.3 Update ADAC Elements from Non Drainage/Sewer String Properties - Survey *and* 10.5.2.2.3 Update ADAC Elements from Non Drainage/Sewer String Properties - Design .

Continue to the next section 10.4.9.3.6.3 Set ADAC Attributes from Drainage & Sewer Data or return to 10.4.9.3.6 12dPL's in Chains .

10.4.9.3.6.3 Set ADAC Attributes from Drainage & Sewer Data

Position of option on menu:

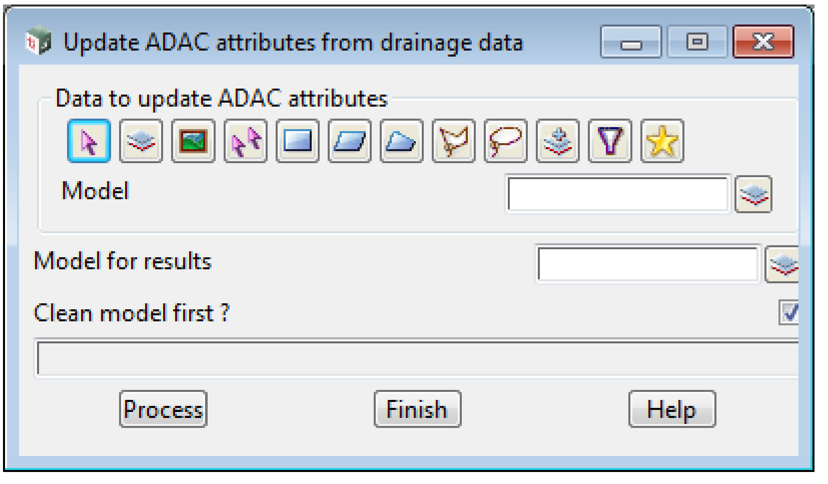
File I/O =>ADAC =>User =>Utilities =>12dPLs in chains =>Set ADAC attributes from drainage/ sewer data

The documentation for this section is a work in progress.

Drainage and sewer strings have an incredible amount of data that can go directly into the ADAC *StormWater Pits* and *Pipes* and the *Sewerage MaintenanceHoles* and *NonPressurePipes*.

Many of the ADAC entities can be calculated directly from the 12d strings and some values may be taken from the tins **survey_finished_tin** or **design_finished_tin**.

Selecting **Set ADAC attributes from Drainage & Sewer Data** brings up the **Update ADAC Data from Drainage Data** panel:



Data source type Model

data selection type - for a full description go to 4.19.3 Data Source .

Data source input

source of data to process.

Model for results model box available models

model that the processed strings are added to.

Clean model first ? tick box

*if **ticked**, the model* **Model for results** *is cleaned before any strings are added to it.*
*If **not ticked**, the model* **Model for results** *is NOT cleaned.*

Process button

update the ADAC Asset attributes with values from the 12d strings properties.

If the string has been marked as an ADAC Asset then it is copied and then for the copied string, any relevant ADAC elements updated from the properties of the 12d string and the tin.

Strings that are not marked as ADAC Assets are NOT copied

*The copied strings are added to the model **Model for results**.*

In the *ADAC Design* chain, this option is called and passed arguments for all the panel field and so runs without needing to display a panel.

See 10.5.2.2.4 Update ADAC Elements from Drainage/Sewer String Properties .

Continue to the next section 10.4.9.3.6.5 Set ADAC ObjectId from String UID or return to 10.4.9.3.6 12dPL's in Chains .

10.4.9.3.6.4 Set ADAC Attributes from User Attributes

NO LONGER USED

Position of option on menu:

File I/O =>ADAC =>User =>Utilities =>12dPLs in chains =>Set ADAC attributes from user attributes

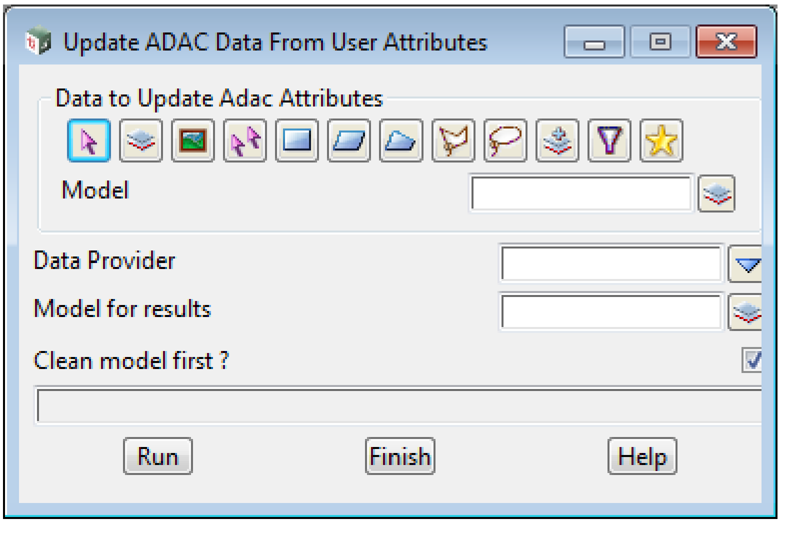
The option is no longer used and had been replaced by a 12d supplied **12duaf file**. See 10.4.8.5 Create/Edit User Attributes to ADAC File and 10.4.8.6 Apply User Attributes to ADAC Elements .

There are a number of 12d options, mainly in the **ADAC =>User =>Data prep** menu, that create User attributes that are to pass data to ADAC Assets. These 12d created user attributes update the ADAC Assets by running this option with the **Data Provider** set to **12d**.

Similarly a user can crate their own user attributes by running their own *12dPLs,* or collecting attributes in the field, or even by hand editing and entering them.

However because these attributes are not know known to *12d Solutions*, the user must set up a **Data Provider** table so the option knows what user attributes to use and what ADAC Assets to update.

Selecting **Set ADAC attributes from user attributes** brings up the **Update ADAC Data from User Attributes** panel.



Data source type Model

data selection type - for a full description go to 4.19.3 Data Source .

Data source input

source of data to process.

Data Provider choice box available data providers

selects the table of User Attributes to use.

Model for results model box available models

model that the processed strings are added to.

Clean model first ? tick box

if ***ticked****, the model* **Model for results** *is cleaned before any strings are added to it.*
If ***not ticked****, the model* **Model for results** *is NOT cleaned.*

Run button

update the ADAC Asset attributes with values from User attributes.

If the string has been marked as an ADAC Asset then it is copied and then for the copied string, any relevant ADAC elements updated from the user attributes.

Strings that are not marked as ADAC Assets are NOT copied

*The copied strings are added to the model **Model for results**.*

Continue to 10.4.9.3.6.5 Set ADAC ObjectId from String UID or return to 10.4.9.3.6 12dPL's in Chains .

10.4.9.3.6.5 Set ADAC ObjectId from String UID

Position of option on menu:
File I/O =>ADAC =>User =>Utilities =>12dPLs in chains =>Set ADAC ObjectId from UID

From the *ADAC XML Schema, the* **Objectid**

Represents a place for an object identifier, usually generated by the data creation software. Also useful as a placeholder for record ID generated by back-end databases or GIS
The ObjectId is permitted to be nil because the capturing system may not have the capacity to generate one. If features are exported from a GIS or Asset Management system, however, it is highly recommended to carry them out into this element.

Every string in ***12d Model*** has a unique identifier called a UID, so ***12d Model*** creates a unique ObjectId by:

(a) for drainage and sewer strings, creating ObjectIds for

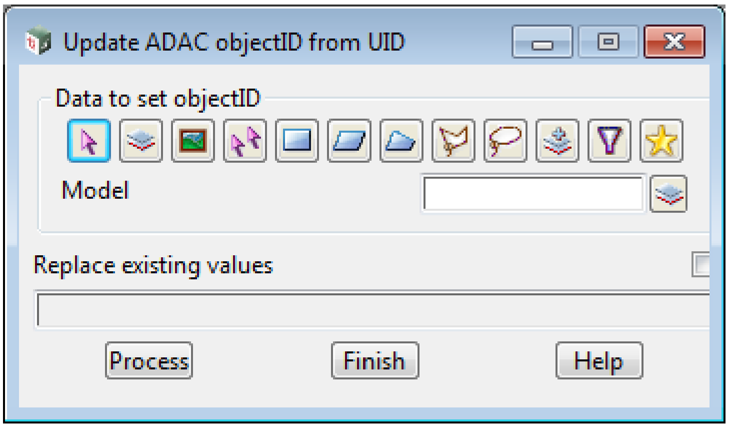
(i) each pit/manhole. The ObjectId is made up of the string UID, the pit/manhole vertex index and information about the to help connect the string together.

(ii) each pipe. The ObjectId is made up of the string UID, the pipe segment index and information about which pits are at the ends of the pipe.

This is only relevant for designers and not surveyors.

(b) for all other strings, setting the ObjectId to the string UID.

Selecting **Set ADAC ObjectID from UID** brings up the **Update ADAC objectID from UID** panel:



Data source type Model

data selection type - for a full description go to 4.19.3 Data Source .

Data source input

source of data to process.

Replace existing values tick box

if ***ticked****, the model* **Model for results** *is cleaned before any strings are added to it.*
If ***not ticked****, the model* **Model for results** *is NOT cleaned.*

Process button

update the ADAC Asset attributes with values from the 12d string's properties.

If the string has been marked as an ADAC Asset then it is copied and then for the copied string, any relevant ADAC elements updated from the properties of the 12d string and the tin.

Strings that are not marked as ADAC Assets are NOT copied

The copied strings are added to the model ***Model for results****.*

In the *ADAC Survey* and *Design* chains, this option is called and arguments are passed from the chain for all the panel fields and so the option runs in the chain without needing to display a panel.

See 10.5.1.2.6 Update ADAC ObjectId - Survey *and* 10.5.2.2.7 Update ADAC ObjectId - Design .

Return to 10.4.9.3.6 12dPL's in Chains .

10.4.9.4 12d 4.1 Chains

Position of menu: **File I/O =>ADAC =>User =>12d 4.1 chains**

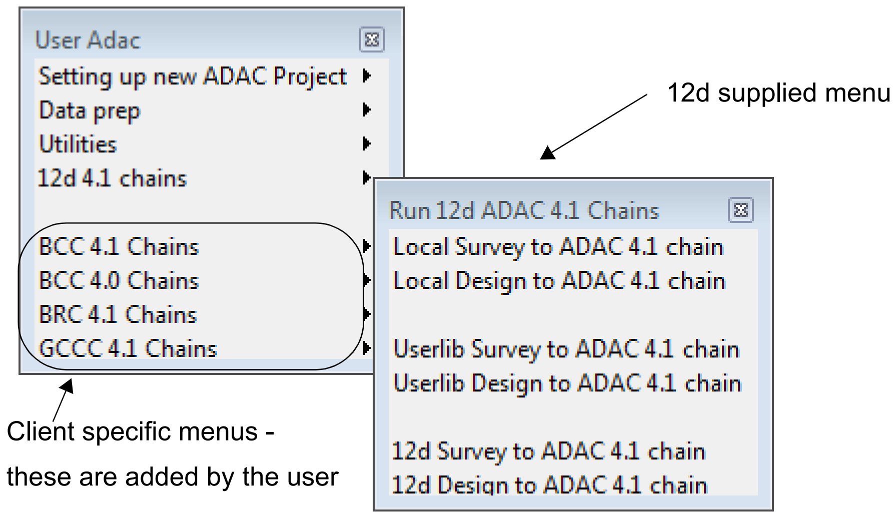
There is a base *ADAC Design* chain and a base *ADAC Survey* chain that does all the work of taking ***12d Model*** strings and producing ADAC Assets ready to be validated, reported on, or written out to an ADAC XML file.

12d Model provides options on the walk right menu **Run 12d ADAC 4.1 Chains** to run the base *ADAC Survey* or *Design* chains for ADAC 4.1 with specially named Map and *12duaf* files in either:

(a) the working folder for the project (local)

(b) User_Lib

(c) Library.



See

10.4.9.4.1 Local Survey to ADAC 41 chain
10.4.9.4.2 Local Design to ADAC 41 chain

10.4.9.4.3 Userlib Survey to ADAC 41 chain
10.4.9.4.4 Userlib Design to ADAC 41 chain

10.4.9.4.5 Lib Survey to ADAC 41 chain
10.4.9.4.6 Lib Design to ADAC 41 chain

Note: **Client Specific ADAC Design and Survey Menus**

The above options are to make it easy to get up and running with ADAC when you have only one or two different Map files.

However, different Clients may do some things differently. For example each company may have its own survey and/or design naming convention or different Authorities may require different information in the ADAC Assets.

For these situations, you can add your own options to **User Adac** menu.

The user options can run either the ADAC Base Survey or Design chains with specific chain pvf files.

For information on setting up your own options on the **User ADAC** menu, see 10.4.9.5 Client Specific ADAC Design & Survey Menus .

10.4.9.4.1 Local Survey to ADAC 41 chain

Position on menu:
File I/O =>ADAC =>User =>12d 4.1 chains =>Local Survey to ADAC 41 chain

The option **Local Survey to ADAC 41 chain** runs *ADAC_Survey_Base* chain but with a pvf file that looks in the **working folder for the project** (local) for the files:

ADAC_Local_Map_Survey_to_ADAC_41.mapfile

Local_41.12duaf *this file is optional*

and uses the a **tin** called **ground** to get surface levels.

So to use this menu item, you only need to create the two files and have them in the folder containing the project. This means they are only available for that project.

Continue to 10.4.9.4.2 Local Design to ADAC 41 chain or return to 10.4.9.4 12d 4.1 Chains .

10.4.9.4.2 Local Design to ADAC 41 chain

Position on menu
File I/O =>ADAC =>User =>12d 4.1 chains =>Local Design to ADAC 41 chain

The option **Local Design to ADAC 41 chain** runs *ADAC_Design_Base Base* chain but with a pvf file that looks in the **working folder for the project** (local) for the files:

ADAC_Local_Map_Design_to_ADAC_41.mapfile

Local_41.12duaf

and uses the a **tin** called **ground** to get surface levels.

So to use this menu item, you only need to create the two files and have them in the folder containing the project. This means they are only available for that project.

Continue to 10.4.9.4.3 Userlib Survey to ADAC 41 chain or return to 10.4.9.4 12d 4.1 Chains .

10.4.9.4.3 Userlib Survey to ADAC 41 chain

Position on menu:
File I/O =>ADAC =>User =>12d 4.1 chains =>Userlib Survey to ADAC 41 chain

The option **Userlib Survey to ADAC 41 chain** runs *ADAC_Survey_Base* chain but with a pvf file that looks in the User_Lib folder for the files:

ADAC_Userlib_Map_Survey_to_ADAC_41.mapfile

Userlib_41.12duaf *this file is optional*

and uses the a **tin** called **ground** to get surface levels.

So to use this menu item, you only need to create the two files and have them in the User_Lib folder. This means they can be used with any project.

Note the word Userlib and not User_Lib in the name of the Map and 12duaf files.

Continue to 10.4.9.4.4 Userlib Design to ADAC 41 chain or return to 10.4.9.4 12d 4.1 Chains .

10.4.9.4.4 Userlib Design to ADAC 41 chain

Position on menu:
File I/O =>ADAC =>User =>12d 4.1 chains =>Userlib Design to ADAC 41 chain

The option **Userlib Design to ADAC 41 chain** runs *ADAC_Design_Base* chain but with a pvf file that looks in the User_Lib folder for the files:

ADAC_Userlib_Map_Design_to_ADAC_41.mapfile

Userlib_41.12duaf *this file is optional*

and uses the a **tin** called **ground** to get surface levels.

So to use this menu item, you only need to create the two files and have them in the User_Lib folder. This means they can be used with any project.

Note the word Userlib and not User_Lib in the name of the Map and 12duaf files.

Continue to 10.4.9.4.5 Lib Survey to ADAC 41 chain or return to 10.4.9.4 12d 4.1 Chains .

10.4.9.4.5 Lib Survey to ADAC 41 chain

Position on menu:
File I/O =>ADAC =>User =>12d 4.1 chains =>12d Survey to ADAC 41 chain

The option **12d Survey to ADAC 41 chain** runs *ADAC_Survey_Base* chain but with a pvf file that looks in the Library folder that is installed with ***12d Model*** for the files:

ADAC_12d_Map_Survey_to_ADAC_41.mapfile

12d_41.12duaf *this file is optional*

and uses the a **tin** called **ground** to get surface levels.

WARNING: This option is for demonstration and training purposes only because the files will be overwritten whenever a new version of ***12d Model 11*** is installed.

Note the word 12d and not Library or Lib in the name of the Map and 12duaf files.

Continue to 10.4.9.4.6 Lib Design to ADAC 41 chain or return to 10.4.9.4 12d 4.1 Chains .

10.4.9.4.6 Lib Design to ADAC 41 chain

Position on menu:
File I/O =>ADAC =>User =>12d 4.1 chains =>Lib Design to ADAC 41 chain

The option **12d Design to ADAC 41 chain** runs *ADAC_Design_Base* chain but with a pvf file that looks in the Library folder that is installed with ***12d Model*** for the files:

ADAC_12d_Map_Design_to_ADAC_41.mapfile

12d_41.12duaf *this file is optional*

and uses the a **tin** called **ground** to get surface levels.

WARNING: This option is for demonstration and training purposes only because the files will be overwritten whenever a new version of ***12d Model 11*** is installed.

Note the word 12d and not Library or Lib in the name of the Map and 12duaf files.

Return to 10.4.9.4 12d 4.1 Chains .

10.4.9.5 Client Specific ADAC Design & Survey Menus

Position of menu: **File I/O =>ADAC =>User =>Client chains**

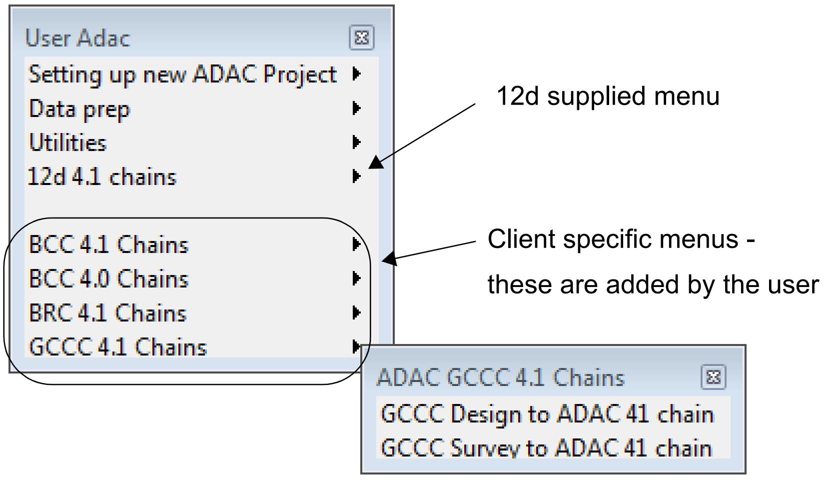
There is a base ADAC Design chain and a base ADAC Survey chain that does all the work of taking ***12d Model*** strings and producing ADAC Assets ready to be validated, reported on, or written out to an ADAC XML file.

However different Clients may do things differently. For example some Clients may have their own survey or design naming convention and some Clients may need ADAC 4.1 data whilst others want ADAC 4.0.

To allow for such variations, the base ADAC Design chain and Survey chains have parameters passed down to them via a chain pvf file. And you can add your **own** items to the **User Adac** menu that use the base ADAC Survey or Design chains with different chain pvf files.

For these situations, you can add your own options to **User Adac** menu.

A **User ADAC** menu with extra Client specific items would then look like:



How to set up these user defined ADAC menus for your company is described in 10.6.3.2 Setting Up Your User ADAC Menu .

Note: **12d Supplied ADAC 4.1 Design and Survey Menu**

If you are only have one or two different Map files, ***12d Model*** supplies options on the walk right menu **Run 12d ADAC 4.1 Chains** of the option **12d 4.1 chains**, to run the base *ADAC Survey* or *Design* chains for ADAC 4.1 with specially named Map and *12duaf* files in either the working folder for the project (local), User_Lib or Library.

For information on the 12d supplied walk-right menu **Run 12d ADAC 4.1 Chains**, see 10.4.9.4 12d 4.1 Chains .

Return to 10.4.9.3 User ADAC Utilities .

10.5 ADAC Design and Survey Chains

The full explanation of the ADAC Design and ADAC Survey chains is usually only needed for the one or two people in a company who are **setting up** the **ADAC procedures** in the company.

Other users do not need to read it but may find it interesting to read.

In the steps outlined in 12d ADAC Workflow (see 10.3 12d ADAC Workflow) it states that the *ADAC Design and ADAC Survey* chains:

(a) Assign the ADAC Asset Type

Any data going out to ADAC must be one of the ADAC Assets.

The assignment may already have been done by the user with the **Create ADAC Asset** option (see 10.4.2 Create ADAC Asset) but it can also be done automatically using a ***12d Map File.*** This ADAC Map File must have already been set up for the company.

(b) Sets ADAC attribute values from the 12d String Geometry and User Attributes

Much of the required ADAC data is already contained within the 12d strings or in User Attributes.

For example, if the Drainage or Sewer was designed with ***12d Model*** then much of the ADAC data can come directly from the drainage and sewer strings. For example, maintenance holes and chambers, depths, and various pipe dimensions are used, required 2D and 3D lengths and areas are calculated.

User attributes picked up in the field by the surveyors, or added in the Data Prep Step, are also loaded into the required ADAC attributes.

(c) Creates the ADAC Geometry

For each ADAC Asset, the ADAC Geometry is automatically generated from the 12d string geometry.

For example, if the ADAC Asset has Polyline Geometry, the (x,y,z) coordinates for each vertex of the string are loaded into the ADAC Geometry.

To go through the workings of the ADAC Design and ADAC Survey chains, see

10.5.1 Survey to ADAC Chains

10.5.2 Design to ADAC Chains

10.5.1 Survey to ADAC Chains

The **ADAC Survey Chain** takes all the string on the view **Survey to map to adac** and produces ADAC Asset strings in a model **survey adac data** which is added to the view **Survey ready for ADAC**.

So the user simply adds all the strings that are to go into the ADAC XML file onto the view called **Survey to map to adac** and clicks on the appropriate button on a walk right menu on the **User ADAC** menu, to run the chain for a particular customer and ADAC version.

There is actually no **Survey to ADAC Chain** and what the menu option does is run the chain **ADAC_survey_base.chain** with a particular chain parameter value file (**pvf** file).

So to fully understand the process, you first need to look at the **pvf** file for the chain ADAC_survey_base.chain, and then examine in detail the *ADAC_survey_base.chain* itself.

See

10.5.1.1 ADAC Survey Base Chain pvf File

10.5.1.2 ADAC Survey Base Chain

10.5.1.1 ADAC Survey Base Chain pvf File

The **pvf** file for the *ADAC_survey_base* chain passes **nine** parameters to the chain - **five** that the user must set and four that are constructed from these five parameters.

1. The text parameter **company** which has as its value an abbreviated customer name. For example, **BCC**.

   This is used to build up the name of the **ADAC 12d Map File** to use.
2. The text parameter **adac_version_by_ten**, which has the value **41** if you are generating ADAC 4.1.0 files, or **40** if you are generating ADAC 4.0 files.

   This is used to build up the name of the **ADAC 12d Map File** to use.
3. The text parameter **user_attributes_conversion_type**
4. The text parameter **survey_finished_tin**
5. The text parameter **design_finished_tin** is not used in the ADAC_survey_base chain and can be left blank.

There are two parameters to define which ***12d Map Files*** to use but they are fully determined once *company* and *adac_version_by_ten* have values.

6. The text parameter **survey_map_file** has the value $USER_LIB\ADAC_[company]_Map_Survey_to_ADAC_[adac_version_by_ten].mapfile
7. The text parameter **design_map_file** has the value $USER_LIB\ADAC_[company]_Map_Design_to_ADAC_[adac_version_by_ten].mapfile

   design_map_file is not actually used in the ADAC_survey_base chain.

There are two parameters to define which ***12d User Adac to ADAC Elements Files*** to use and again they are fully determined once *company* and *adac_version_by_ten* have values.

8. The text parameter **adac_attribute_file** has the value $USER_LIB\[company]_[adac_version_by_ten].12duaf

   This file is different for each company and must exist (it can contain no information).
9. The text parameter **adac_12d_attribute_file** has the value $USER_LIB\ADAC_12d_[adac_version_by_ten].12duaf

   This file is the same for each company and is for user attributes that *12d Solutions* has created.

The **pvf** file is passed to the **ADAC_survey_base.chain** by having the chain run as a menu option on one of the walk right menus on the **User ADAC** menu.

```
Menu "ADAC BCC 4.1 Chains" {
  Button "BCC Design to ADAC 41 chain" {
    Command "chain  -pvf $USER/BCC/ADAC_BCC_41.pvf   $LIB/ADAC_design_base.chain"
  }
  Button "BCC Survey to ADAC 41 chain" {
    Command "chain  -pvf $USER/BCC/ADAC_BCC_41.pvf  $LIB/ADAC_survey_base.chain"
  }
}
```

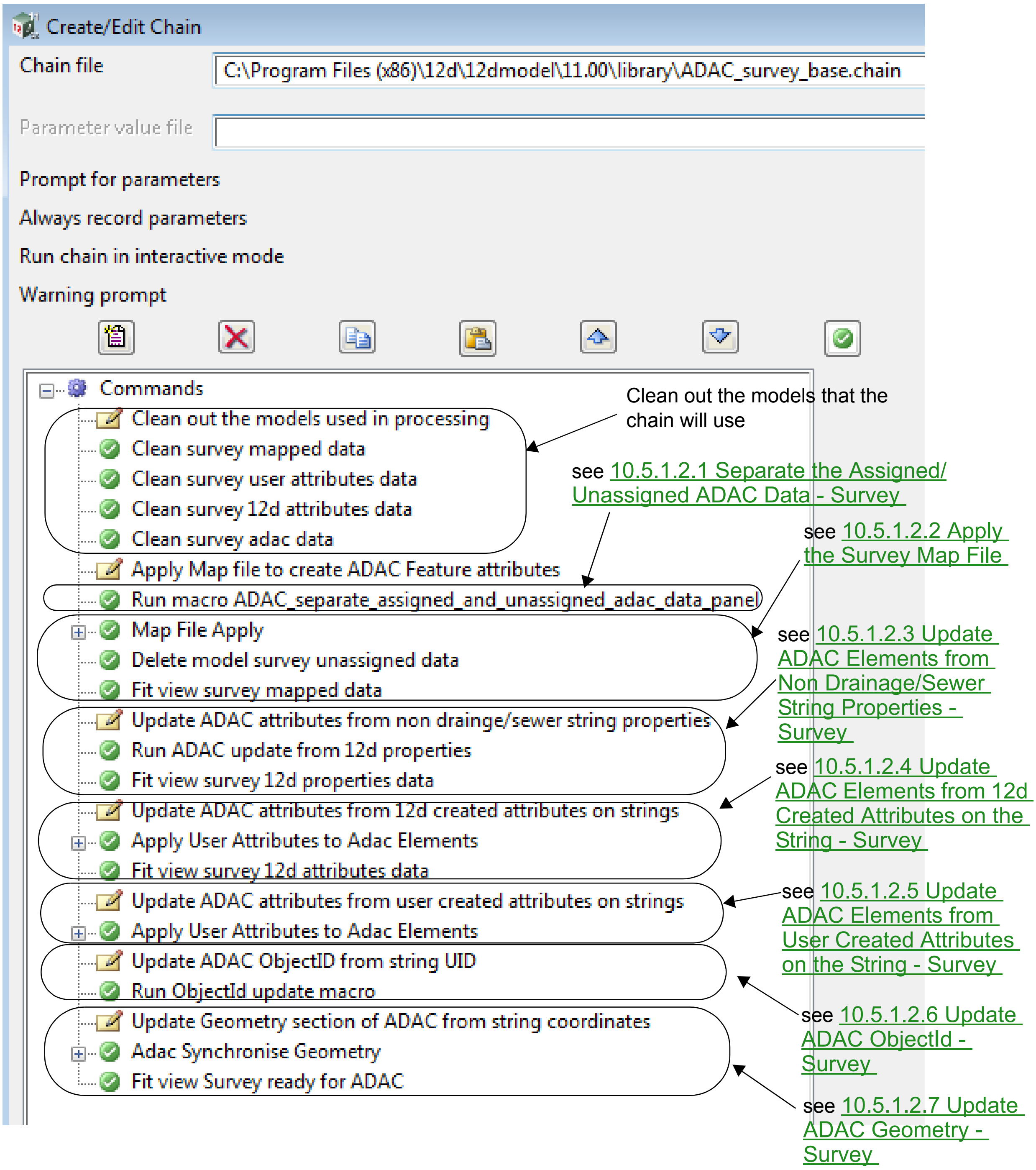
pvf file to set the parameters for customer BCC and ADAC 4.1

base chain

ADAC_Survey_base.chain is discussed in detail in 10.5.1.2 ADAC Survey Base Chain

10.5.1.2 ADAC Survey Base Chain

The chain *ADAC_survey_base.chain* takes all the data on the view **Survey to map to adac** and ends up producing ADAC Assets in a model **survey adac data** which is added to the view **Survey ready for ADAC**.



10.5.1.2.1 Separate the Assigned/Unassigned ADAC Data - Survey

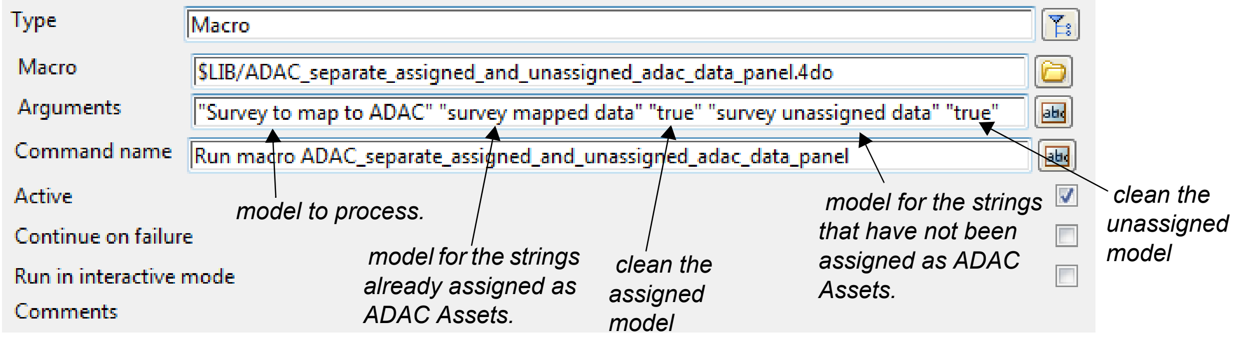
Some strings on the **View** called **Survey to map to ADAC** may have already been assigned (marked) as ADAC Assets and so should not have the ADAC Map file applied to them.

So the data is first processed to separate the assigned and unassigned data into separate models.

This is achieved by running the 12dPL (macro)

$LIB/ADAC_separate_assigned_and_unassigned_adac_data_panel.4do

which looks at all the data on the **View** called **Survey to map to ADAC** and copies the strings that have been assigned as ADAC Assets to the model **survey mapped data** and all the data that has not yet been assigned to the model **survey unassigned data**.



The model **survey mapped data** is on the view called **survey ready for adac**.

Note - this 12d PL program is described in more detail in 10.4.9.3.6.1 Separate Assigned/ Unassigned ADAC Data

Continue to the next section 10.5.1.2.2 Apply the Survey Map File or return to 10.5.1.2 ADAC Survey Base Chain or 10.5.1 Survey to ADAC Chains .

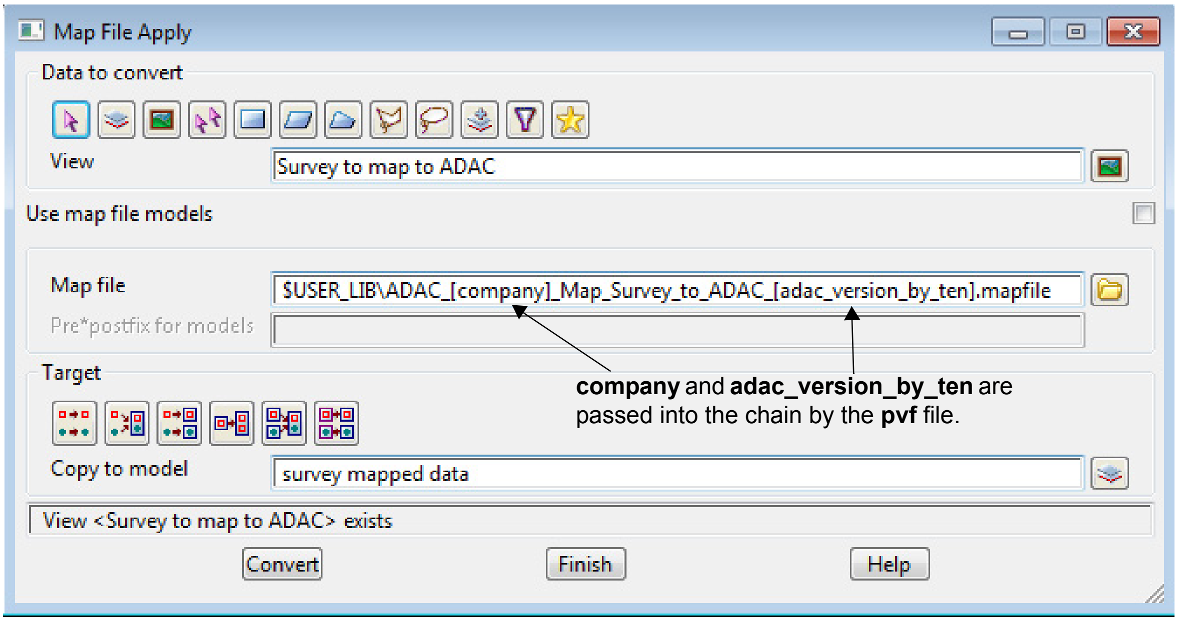
10.5.1.2.2 Apply the Survey Map File

This section of the chain applies the ***12d Map File***

$USER_LIB\ADAC_[company]_Map_Survey_to_ADAC_[adac_version_by_ten].mapfile

to the data in the model **survey unassigned data** and copies all of the string to the model **survey mapped data**.

Most importantly, the **12d Map File** also assigns (marks) some strings as ADAC Assets by giving them ADAC Attributes as defined by the **12d Map File**.



The model **survey mapped data** is on the view called **survey mapped data** and a fit is done on the view.

The model **survey unassigned data** is deleted.

Continue to the next section 10.5.1.2.3 Update ADAC Elements from Non Drainage/Sewer String Properties - Survey or return to 10.5.1.2 ADAC Survey Base Chain or 10.5.1 Survey to ADAC Chains.

10.5.1.2.3 Update ADAC Elements from Non Drainage/Sewer String Properties - Survey

Many of the ADAC entities can be calculated directly from the 12d strings and some values may be taken from the tin **survey_finished_tin** that is passed into the chain by the **pvf** file.

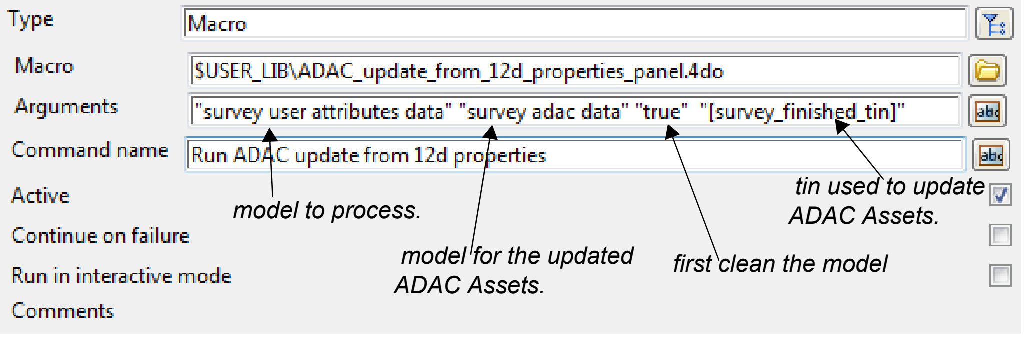
This is achieved by running the 12dPL (macro)

$LIB\ADAC_update_from_12d_properties_panel.4do

which looks at all the data in the model produced in the previous step and if a string has been marked as an ADAC Asset, then it is copied and any relevant ADAC elements updated from the properties of the12d string itself and the tin **survey_finished_tin**.

The copied string is added to the model **survey adac data**.

Only ADAC Assets are added to the model **survey adac data.**



The model **survey adac data** is on the view called **survey ready for adac**.

Note - this 12d PL program is described in more detail in 10.4.9.3.6.2 Set ADAC Attributes from String Properties

Continue to the next section 10.5.1.2.4 Update ADAC Elements from 12d Created Attributes on the String - Survey or return to 10.5.1.2 ADAC Survey Base Chain or 10.5.1 Survey to ADAC Chains .

10.5.1.2.4 Update ADAC Elements from 12d Created Attributes on the String - Survey

Some of the values for ADAC elements can be taken from attributes that 12d options have placed on the string. For example, using the ADAC Common Editor panel documented in the section 10.4.9.2.1 Providing Extra Data for ADAC.

This is achieved by running the panel

ADAC User Attributes to ADAC Element

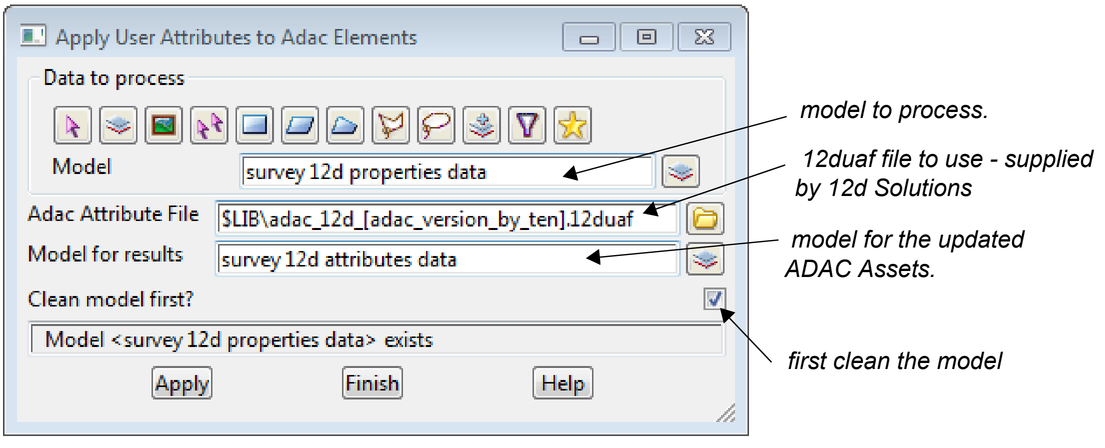
with a *12d Solutions* supplied *12duaf* file for the required version of ADAC.

(this is the option ADAC =>Utilities =>Apply User Attributes to ADAC elements).

The option looks at all the data in the model produced in the previous step and each string is copied and then any relevant ADAC elements updated from User attributes of the string according to the given *12duaf* file.

Note that these User attributes have been created by ***12d Model*** options that 12d Solutions knows about and hence can provide the *12duaf* file.

The copied strings are added to the model **survey 12d attributes data**.



The model **survey 12d attributes data** is on the view called **survey 12d attributes data** and a fit is done on the view.

Note - this option is described in more detail in 10.4.8.6 Apply User Attributes to ADAC Elements

Continue to the next section 10.5.1.2.5 Update ADAC Elements from User Created Attributes on the String - Survey or return to 10.5.1.2 ADAC Survey Base Chain or 10.5.1 Survey to ADAC Chains.

10.5.1.2.5 Update ADAC Elements from User Created Attributes on the String - Survey

Some of the values for ADAC elements can be taken from attributes that users have placed on the string. For example, from attributes picked up by surveyors in the field.

Because these User attributes are defined by the user, *12d Solutions* has no idea of what they are so the user must set up their own *12duaf* file that can be used to updated the appropriate ADAC elements from these user created User Attributes.

This user supplied *12duaf* file is placed in the users User_Lib.

The user supplied *12duaf* file is used by running the panel

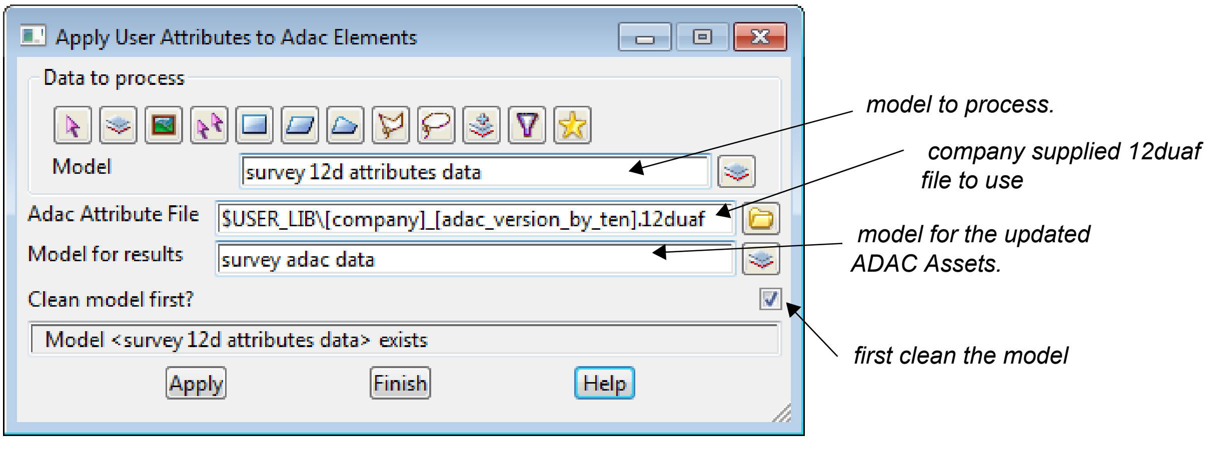
ADAC User Attributes to ADAC Element

with the user supplied 12duaf file for the required version of ADAC.

(this is the option ADAC =>Utilities =>Apply User Attributes to ADAC elements).

The option looks at all the data in the model produced in the previous step and each string is copied and then any relevant ADAC elements updated from User attributes of the string according to the given *12duaf* file.

The copied strings are added to the model **survey user attributes data**.



The model **survey user attributes data** is on the view called **survey string properties data.**

Note - this option is described in more detail in 10.4.8.6 Apply User Attributes to ADAC Elements

Continue to the next section 10.5.1.2.6 Update ADAC ObjectId - Survey or return to 10.5.1.2 ADAC Survey Base Chain or 10.5.1 Survey to ADAC Chains .

10.5.1.2.6 Update ADAC ObjectId - Survey

From the *ADAC XML Schema, the* **Objectid**

> *Represents a place for an object identifier, usually generated by the data creation software. Also useful as a placeholder for record ID generated by back-end databases or GIS*
> *The ObjectId is permitted to be nil because the capturing system may not have the capacity to generate one. If features are exported from a GIS or Asset Management system, however, it is highly recommended to carry them out into this element.*

Every string in ***12d Model*** has a unique identifier called a UID, so ***12d Model*** creates a unique ObjectId by:

(a) for drainage and sewer strings, creating ObjectIds for

(i) each pit/manhole. The ObjectId is made up of the string UID, the pit/manhole vertex index and information about them to help connect the string together.

(ii) each pipe. The ObjectId is made up of the string UID, the pipe segment index and information about which pits are at the ends of the pipe.

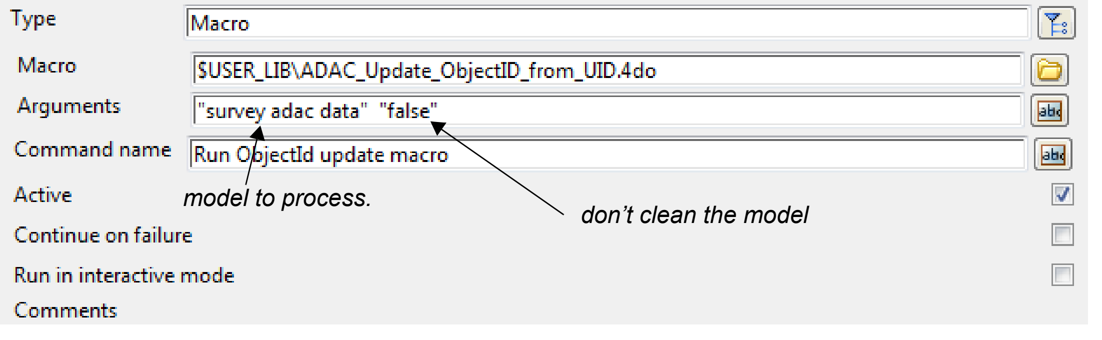
This is only relevant for designers and not surveyors.

(b) for all other strings, setting the ObjectId to the string UID.

The *ObjectId* is created by running the 12dPL (macro)

$LIB\ADAC_update_ObjectId_from_UID_panel.4do

The *12dPL* looks at all the strings in the model produced in the previous step, **survey adac data,** and updates the string's own ADAC ObjectId attribute. So unlike the previous sections, the string is not copied.



The model **survey adac data** is on the view called **survey ready for adac.**

Note - this *12dPL* program is described in more detail in 10.4.9.3.6.5 Set ADAC ObjectId from String UID

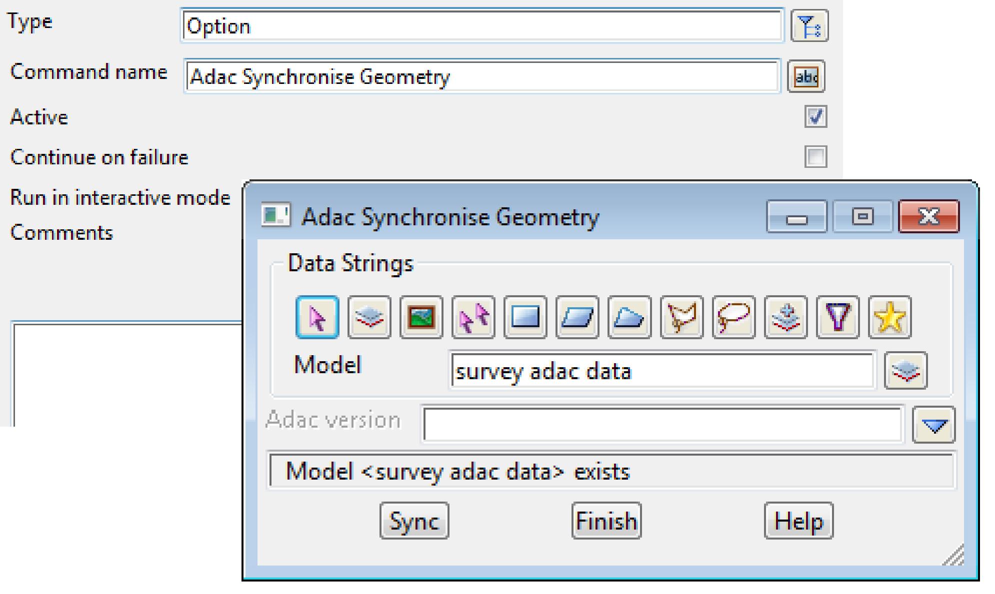
Continue to the next section 10.5.1.2.7 Update ADAC Geometry - Survey or return to 10.5.1.2 ADAC Survey Base Chain or 10.5.1 Survey to ADAC Chains .

10.5.1.2.7 Update ADAC Geometry - Survey

The final step is for every ADAC Asset, updating its ADAC Geometry attributes from the string geometry.

This is done by running in the chain, the option in 12d

File I/O =>ADAC =>Utilities =>Sync geometry



The model **survey adac data** is on the view called **survey ready for adac** and a fit is done on the view**.**

Note - this 12d PL program is described in more detail in 10.4.8.4 Sync Geometry

Return to 10.5.1.2 ADAC Survey Base Chain or 10.5.1 Survey to ADAC Chains .

10.5.2 Design to ADAC Chains

The **ADAC Design Chain** takes all the strings on the view **Design to map to adac** and produces ADAC Asset strings in a model **design adac data** which is added to the view **Design ready for ADAC**.

So the user simply adds all the strings that are to go into the ADAC XML file onto the view called **Design to map to adac** and clicks on the appropriate button on a walk right menu on the **User ADAC** menu, to run the chain for a particular customer and ADAC version.

There is actually no **Design to ADAC Chain** and what the menu option does is run the chain **ADAC_design_base.chain** with a particular chain parameter value file (**pvf** file).

So to fully understand the process, you first need to look at the **pvf** file for the chain ADAC_design_base.chain, and then examine in detail the *ADAC_design_base.chain* itself.

See

10.5.2.1 ADAC Design Base Chain pvf File

10.5.2.2 ADAC Design Base Chain

10.5.2.1 ADAC Design Base Chain pvf File

The **pvf** file for the *ADAC_design_base* chain passes **nine** parameters to the chain - **five** that the user must set and four that are constructed from these five parameters.

1. The text parameter **company** which has as its value an abbreviated customer name. For example, **BRC**.

   This is used to build up the name of the **ADAC 12d Map File** to use.

2. The text parameter **adac_version_by_ten**, which has the value **41** if you are generating ADAC 4.1.0 files, or **40** if you are generating ADAC 4.0 files.

   This is used to build up the name of the **ADAC 12d Map File** to use.

3. The text parameter **user_attributes_conversion_type**
4. The text parameter **survey_finished_tin**
5. The text parameter **design_finished_tin** is not used in the ADAC_survey_base chain and can be left blank.

There are two parameters define which ***12d Map Files*** to use but they are fully determined once *company* and *adac_version_by_ten* have values.

6. The text parameter **survey_map_file** has the value $USER_LIB\ADAC_[company]_Map_Survey_to_ADAC_[adac_version_by_ten].mapfile

   survey_map_file is not actually used in the ADAC_design_base chain.

7. The text parameter **design_map_file** has the value $USER_LIB\ADAC_[company]_Map_Design_to_ADAC_[adac_version_by_ten].mapfile

There are two parameters to define which ***12d User Adac to ADAC Elements Files*** to use and again they are fully determined once *company* and *adac_version_by_ten* have values.

8. The text parameter **adac_attribute_file** has the value $USER_LIB\[company]_[adac_version_by_ten].12duaf

   This file is different for each company and must exist (it can contain no information).

9. The text parameter **adac_12d_attribute_file** has the value $USER_LIB\ADAC_12d_[adac_version_by_ten].12duaf

   This file is the same for each company and is for user attributes that *12d Solutions* has created.

The **pvf** file is passed to the **ADAC_design_base.chain** by having the chain run as a menu option on one of the walk right menus on the **User ADAC** menu.

```
Menu "ADAC BRC 4.1 Chains" {
  Button "BCC Survey to ADAC 41 chain" {
    Command "chain  -pvf $USER/BRC/ADAC_BRC_41.pvf   $LIB/ADAC_survey_base.chain"
  }
  Button "BRC Survey to ADAC 41 chain" {
    Command "chain  -pvf $USER/BRC/ADAC_BRC_41.pvf  $LIB/ADAC_design_base.chain"
  }
}
```

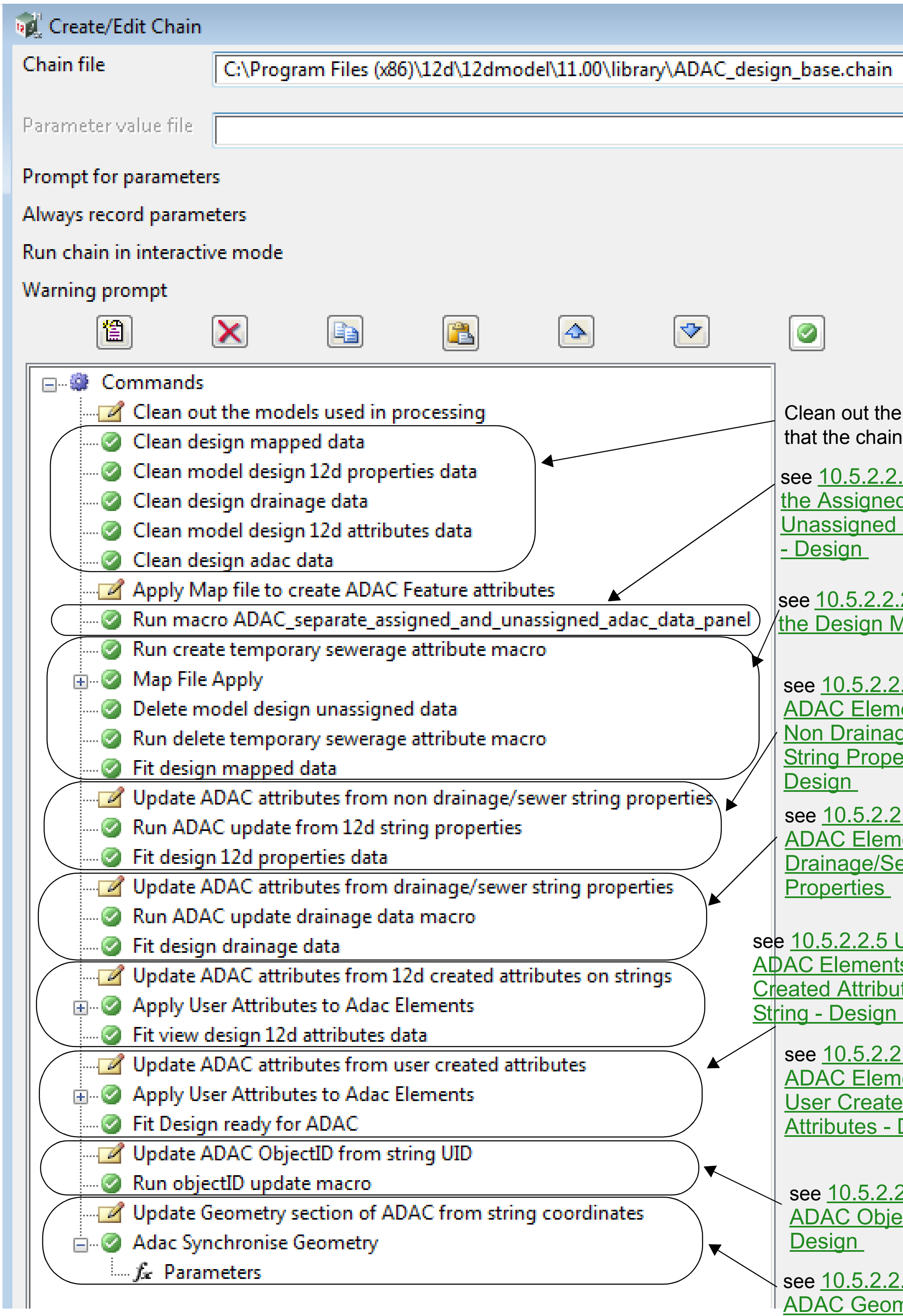
pvf file to set the parameters for customer BRC and ADAC 4.1

base chain

ADAC_Design_base.chain is discussed in detail in 10.5.2.2 ADAC Design Base Chain

10.5.2.2 ADAC Design Base Chain

The chain *ADAC_design_base.chain* takes all the data on the view **design to map to adac** and ends up producing ADAC Assets in a model **design adac data** which is add to the view **Design ready for ADAC**.



Clean out the models that the chain will use

see 10.5.2.2.1 Separate the Assigned/ Unassigned ADAC Data - Design

see 10.5.2.2.2 Applying the Design Map File

see 10.5.2.2.3 Update ADAC Elements from Non Drainage/Sewer String Properties - Design

see 10.5.2.2.4 Update ADAC Elements from Drainage/Sewer String Properties

see 10.5.2.2.5 Update ADAC Elements from 12d Created Attributes on the String - Design

see 10.5.2.2.6 Update ADAC Elements from User Created Attributes - Design

see 10.5.2.2.7 Update ADAC ObjectId - Design

see 10.5.2.2.8 Update ADAC Geometry - Design

10.5.2.2.1 Separate the Assigned/Unassigned ADAC Data - Design

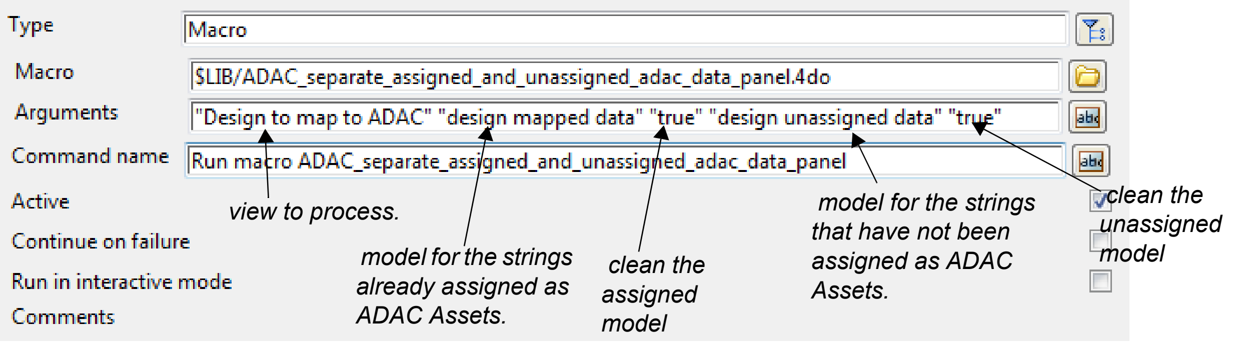
Some strings on the **View** called **Survey to map to ADAC** may have already been assigned (marked) as ADAC Assets and so should not have the ADAC Map file applied to them.

So the data is first processed to separate the assigned and unassigned data into separate models.

This is achieved by running the 12dPL (macro)

$LIB/ADAC_separate_assigned_and_unassigned_adac_data_panel.4do

which looks at all the data on the **View** called **Design to map to ADAC** and copies the strings that have been assigned as ADAC Assets to the model **design mapped data** and all the data that has not yet been assigned to the model **design unassigned data**.



The model **design mapped data** is on the view called **deign ready for adac**.

Note - this 12d PL program is described in more detail in 10.4.9.3.6.1 Separate Assigned/ Unassigned ADAC Data

Continue to the next section 10.5.2.2.2 Applying the Design Map File or return to 10.5.2.2 ADAC Design Base Chain or 10.5.2 Design to ADAC Chains .

10.5.2.2.2 Applying the Design Map File

First the 12dPL option *Create_or_delete_temporary_sewerage_attribute_panel.4do* is run on the data in the model called **design unassigned data** and for each

(a) **drainage string**, creates a **string attribute** called **sewertype** of type Integer with 0.

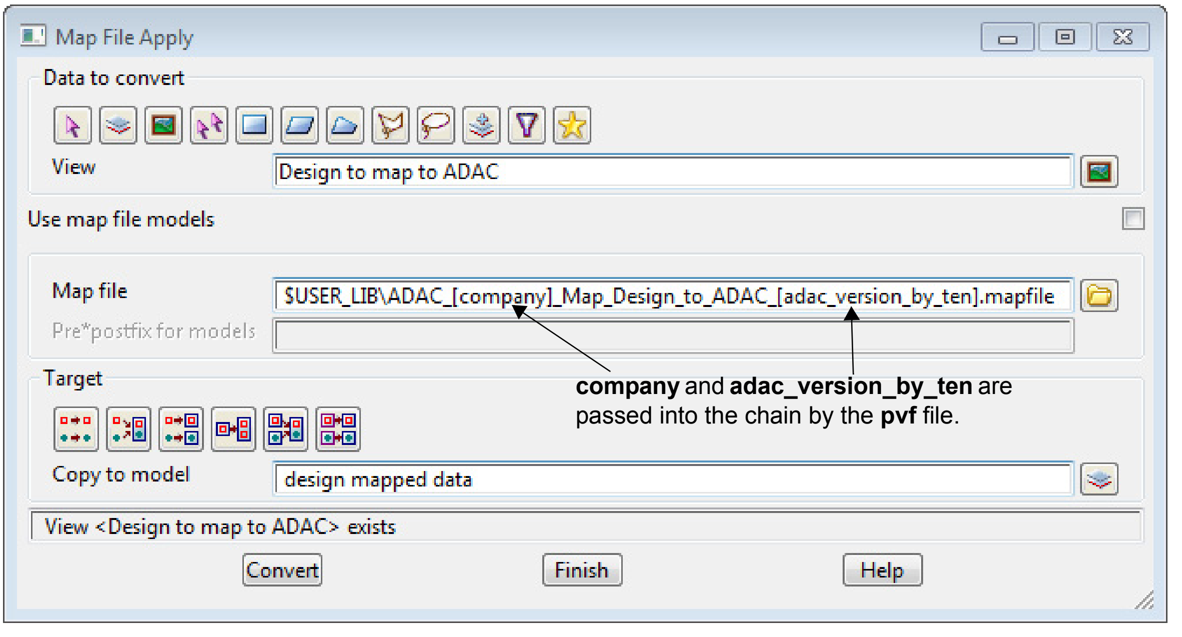
(b) **sewer string**, create a **string attribute** called **sewertype** of type Integer with value 0.

Next the chain applies the ***Map File***

$USER_LIB\ADAC_[company]_Map_Design_to_ADAC_[adac_version_by_ten].mapfile

to the data on the **View** called **Design to map to ADAC** and copies all of the string and adds to the model **design mapped data**.

Most importantly the ***Map File*** marks some strings, and the vertices and segments for drainage and sewer strings, as ADAC Assets by giving them ADAC Attributes as defined by the ***Map File***.



Next the model **design unassigned data** is deleted.

The 12dPL option *Create_or_delete_temporary_sewerage_attribute_panel.4do* is then run on the data on the view called **Design to map to ADAC** and on the model **design mapped data** to remove the string attribute **sewertype**.

The model **design mapped data** is on the view called **design mapped data** and a fit is done on that view.

Continue to the next section 10.5.2.2.3 Update ADAC Elements from Non Drainage/Sewer String Properties - Design or return to 10.5.2.2 ADAC Design Base Chain or 10.5.2 Design to ADAC Chains .

10.5.2.2.3 Update ADAC Elements from Non Drainage/Sewer String Properties - Design

Many of the ADAC entities can be calculated directly from the 12d strings, and some values may be taken from the tin **design_finished_tin** that is passed into the chain by the **pvf** file.

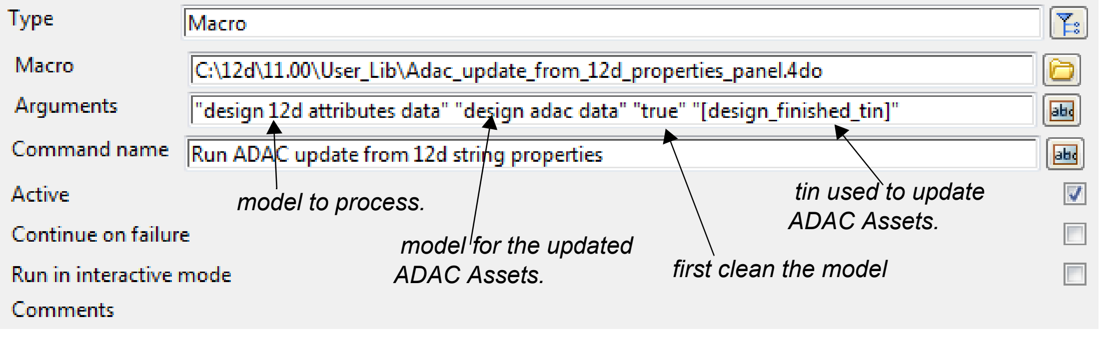
This is achieved by running the 12dPL (macro)

$LIB\ADAC_update_from_12d_properties_panel.4do

which looks at all the data in the model produced in the previous step and if a string has been marked as an ADAC Asset, then it is copied and any relevant ADAC elements updated from the properties of the12d string itself and the tin **design_finished_tin**.

The copied string is added to the model **design string properties data**.

Only ADAC Assets are added to the model **design string properties data.**



The model **design adac data** is on the view called **design ready for adac**.

Note - this 12d PL program is described in more detail in 10.4.9.3.6.2 Set ADAC Attributes from String Properties

Continue to the next section 10.5.2.2.4 Update ADAC Elements from Drainage/Sewer String Properties or return to 10.5.2.2 ADAC Design Base Chain or 10.5.2 Design to ADAC Chains .

10.5.2.2.4 Update ADAC Elements from Drainage/Sewer String Properties

Many of the ADAC entities can be calculated directly from the maintenance holes/pits and pipes from the 12d drainage and sewer strings.

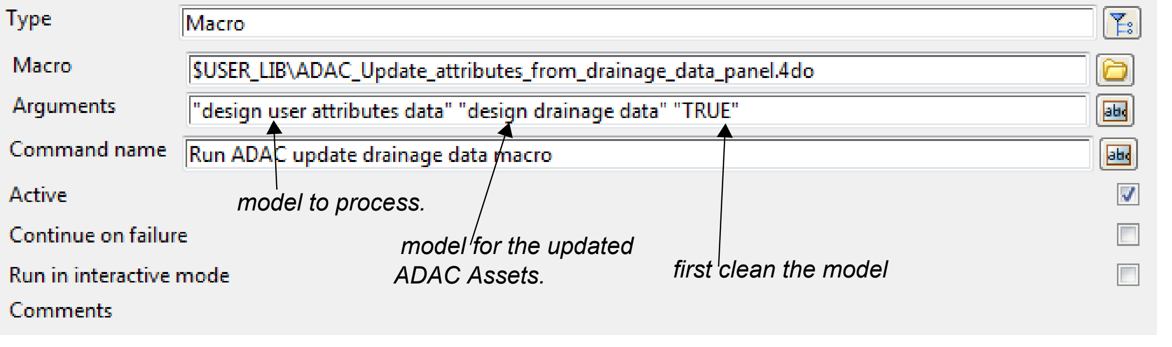
This is achieved by running the 12dPL (macro)

$LIB\ADAC_update_from_drainage_data_panel.4do

which looks at all the drainage and sewer strings in the model produced in the previous step and if the pits and pipes have been marked as ADAC Assets, then the string is copied and any relevant ADAC pits and pipes in the strings updated from the pits and pipes properties of the12d string itself.

The copied string is added to the model **design drainage data**.

Only ADAC Assets are added to the model **design drainage data.**



The model **design drainage data** is on the view called **design drainage data** and a fit is done on the view.

Note - this 12d PL program is described in more detail in 10.4.9.3.6.3 Set ADAC Attributes from Drainage & Sewer Data

Continue to the next section 10.5.2.2.5 Update ADAC Elements from 12d Created Attributes on the String - Design or return to 10.5.2.2 ADAC Design Base Chain or 10.5.2 Design to ADAC Chains .

10.5.2.2.5 Update ADAC Elements from 12d Created Attributes on the String - Design

Some of the values for ADAC elements can be taken from User attributes that 12d options have placed on the string. For example, using the ADAC Common Editor panel documented in the section 10.4.9.2.1 Providing Extra Data for ADAC .

This is achieved by running the panel

ADAC User Attributes to ADAC Element

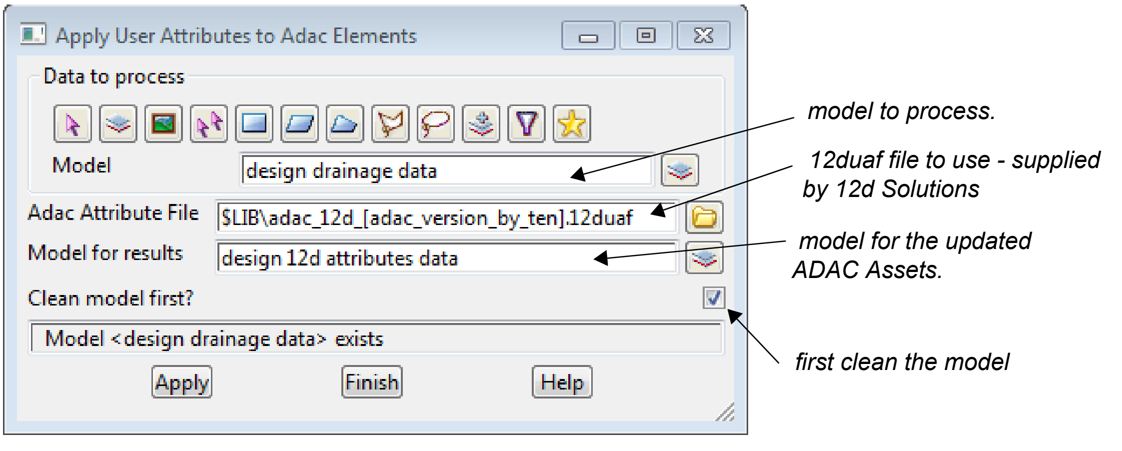
with a *12d Solutions* supplied *12duaf* file for the required version of ADAC.

(this is the option ADAC =>Utilities =>Apply User Attributes to ADAC elements).

The option looks at all the data in the model produced in the previous step and each string is copied and then any relevant ADAC elements updated from User attributes of the string according to the given *12duaf* file.

Note that these User attributes have been created by ***12d Model*** options that *12d Solutions* knows about and hence can provide the *12duaf* file.

The copied strings are added to the model **design 12d attributes data**.



The model **design 12d attributes data** is on the view called **design 12d attributes data** and a fit is done on the view.

Note - this option is described in more detail in 10.4.8.6 Apply User Attributes to ADAC Elements

Continue to the next section 10.5.2.2.4 Update ADAC Elements from Drainage/Sewer String Properties or return to 10.5.2.2 ADAC Design Base Chain or 10.5.2 Design to ADAC Chains .

10.5.2.2.6 Update ADAC Elements from User Created Attributes - Design

Some of the values for ADAC elements can be taken from attributes that users have placed on the string.

Because these User attributes are defined by the user, *12d Solutions* has no idea of what they are so the user must set up their own *12duaf* file that can be used to updated the appropriate ADAC elements from these user created User Attributes.

This user supplied *12duaf* file is placed in the users User_Lib.

The user supplied *12duaf* file is used by running the panel

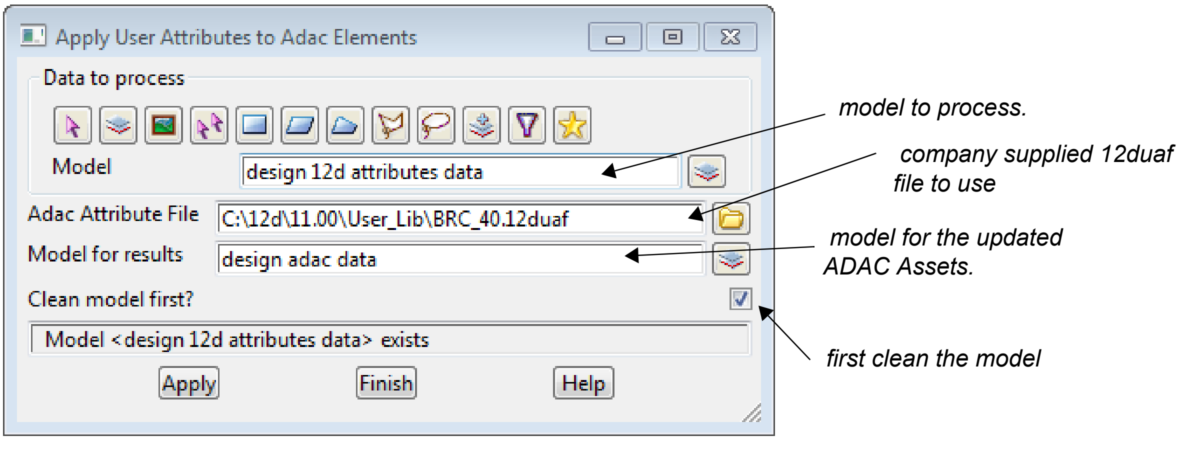
ADAC User Attributes to ADAC Element

with the user supplied 12duaf file for the required version of ADAC.

(this is the option ADAC =>Utilities =>Apply User Attributes to ADAC elements).

The option looks at all the data in the model produced in the previous step and each string is copied and then any relevant ADAC elements updated from User attributes of the string according to the given *12duaf* file.

The copied strings are added to the model **design user attributes data**.



The model **design user attributes data** is on the view called **design string properties data.**

Note - this option is described in more detail in 10.4.8.6 Apply User Attributes to ADAC Elements

Continue to the next section 10.5.2.2.7 Update ADAC ObjectId - Design or return to 10.5.2.2 ADAC Design Base Chain or 10.5.2 Design to ADAC Chains .

10.5.2.2.7 Update ADAC ObjectId - Design

From the *ADAC XML Schema, the* **Objectid**

> *Represents a place for an object identifier, usually generated by the data creation software. Also useful as a placeholder for record ID generated by back-end databases or GIS*
> *The ObjectId is permitted to be nil because the capturing system may not have the capacity to generate one. If features are exported from a GIS or Asset Management system, however, it is highly recommended to carry them out into this element.*

Every string in ***12d Model*** has a unique identifier called a UID, so ***12d Model*** creates a unique ObjectId by:

(a) for drainage and sewer strings, creating ObjectIds for

(i) each pit/manhole. The ObjectId is made up of the string UID, the pit/manhole vertex index and information about the to help connect the string together.

(ii) each pipe. The ObjectId is made up of the string UID, the pipe segment index and information about which pits are at the ends of the pipe.

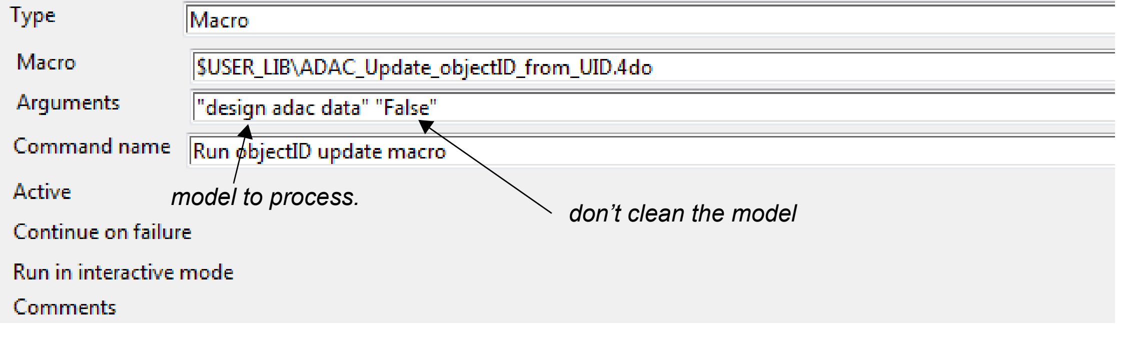
This is only relevant for designers and not surveyors.

(b) for all other strings, setting the ObjectId to the string UID.

The *ObjectId* is created by running the 12dPL (macro)

$LIB\ADAC_update_ObjectId_from_UID_panel.4do

The 12dPL looks at all the strings in the model produced in the previous step, **design adac data,** and updates the string's own ADAC ObjectId attribute. So unlike the previous sections, the string is not copied.



The model **design adac data** is on the view called **design ready for adac.**

Note - this 12d PL program is described in more detail in 10.4.9.3.6.5 Set ADAC ObjectId from String UID .

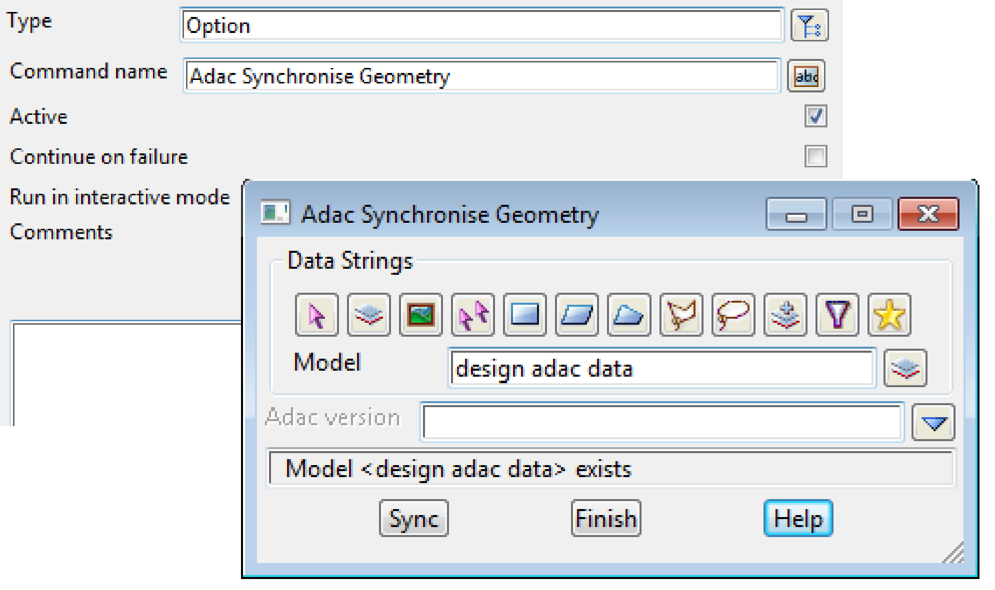
Continue to the next section 10.5.2.2.8 Update ADAC Geometry - Design or return to 10.5.2.2 ADAC Design Base Chain or 10.5.2 Design to ADAC Chains .

10.5.2.2.8 Update ADAC Geometry - Design

The final step is for every ADAC Asset, updating its ADAC Geometry attributes from the string geometry.

This is done by running in the chain, the option 12d

File I/O =>ADAC =>Utilities =>Sync geometry



The model **design adac data** is on the view called **design ready for adac** and a fit is done on the view**.**

Note - this 12d PL program is described in more detail in 10.4.8.4 Sync Geometry

Return to 10.5.2.2 ADAC Design Base Chain or 10.5.2 Design to ADAC Chains .

10.6 Setting Up for ADAC

This section is only for the one or two people in a company who are **setting up** the **ADAC procedures** in the company.

Other users should probably avoid reading it.

See

10.6.1 Setting Up Map Files for ADAC

10.6.2 What Data Prep is Needed for ADAC

10.6.3.2 Setting Up Your User ADAC Menu

10.6.4 Setting Up ADAC Templates

10.6.5 Setting Up Your User Keys

10.6.1 Setting Up Map Files for ADAC

For a ***12d Model*** string to represent an ADAC Asset, it must have

(a) string geometry that matches the ADAC Geometry for the ADAC Asset

(b) as part of its string attributes, the ADAC attribute Group for the particular ADAC Asset.

Picking each string and giving it the correct ADAC attributes for an ADAC Asset can be done using the **ADAC =>Create asset** option and although templates can be used to speed up the process, it is still very time consuming and hence slow.

So in ***12d Model***, a ***12d Map File*** is used to automatically give the appropriate ADAC Asset attributes to strings using the *Attributes* section of the ***12d Map File***.

To do this you need to be able to identify each ADAC Asset by a combination of the ***string name*** and the ***string attributes***.

This process is usually much easier for surveyors because when picking up in the field, you could easily set up coding system with a unique string name for each ADAC Asset.

For designers the process can be more difficult and they may need changes to their string naming convention, or some data preparation steps. On the other hand, designers already have most of the information for the ADAC Assets Sewerage>MaintenanceHoles>MaintenanceHole, Sewerage>NonPressurePipes>NonPressurePipe, StormWater>Pits>Pit and StormWater>Pipes>Pipe as part of their drainage and sewer strings and we'll be able to use that without manual intervention.

First we'll look at ADAC in general and then follow on with sections on setting up the ADAC ***12d Map File***.

See

10.6.1.1 Know What ADAC XML Is

10.6.1.2 Know What Your Client Wants in the ADAC XML File

10.6.1.3 How the ADAC Map File is Used

10.6.1.4 Creating the ADAC Attributes for Map Files

10.6.1.5 Notes On Creating the ADAC Attribute Structure

10.6.1.6 Setting Up the Map File Selection Criteria

10.6.1.7 More on Creating the ADAC Map Files

10.6.1.1 Know What ADAC XML Is

There is no magic way of being able to set up a **Map File** for ADAC without knowing what the ADAC Assets actually are, what key elements they contain and what Geometry they require.

For example, if you need **WaterSupply>Pipes>Pipe** in the ADAC XML file, then it is critical to know that a **WaterSupply>Pipes>Pipe** can only be a polyline with straight segments. So no arcs.

Or an **OpenSpace>BoatingFacilities>BoatingFacility**, which has **Type** choices of *Jetty*, *Pier*, *Ramp* or *Spillways*, can only be a single **Point**.

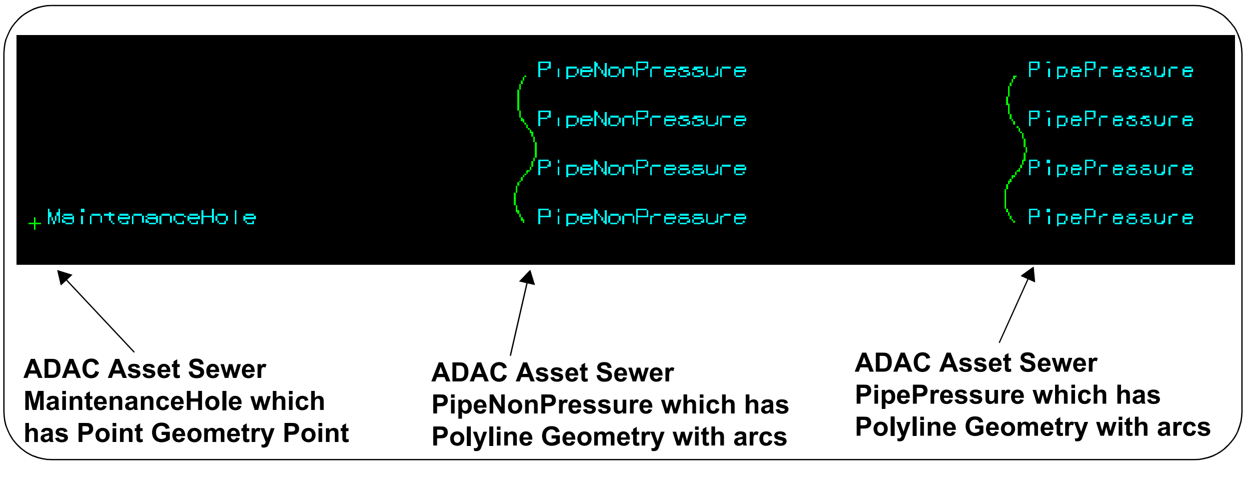
Or that a **Transport>RoadEdges>RoadEdge** must have a **Type** and it can only be one of B1, B2, B3, B4, B5, SM1, SM2, SM3,SM4,SM5, M1, M2, M3, M4, M5, M6, ER1, ER2, ER3, ER4, ER5, INV660, INV90o0, Bitumen and Concrete (as per DMR drawing 1033 - now TMR).

In all cases, any variations from the ADAC Schema will make it an invalid ADAC XML file.

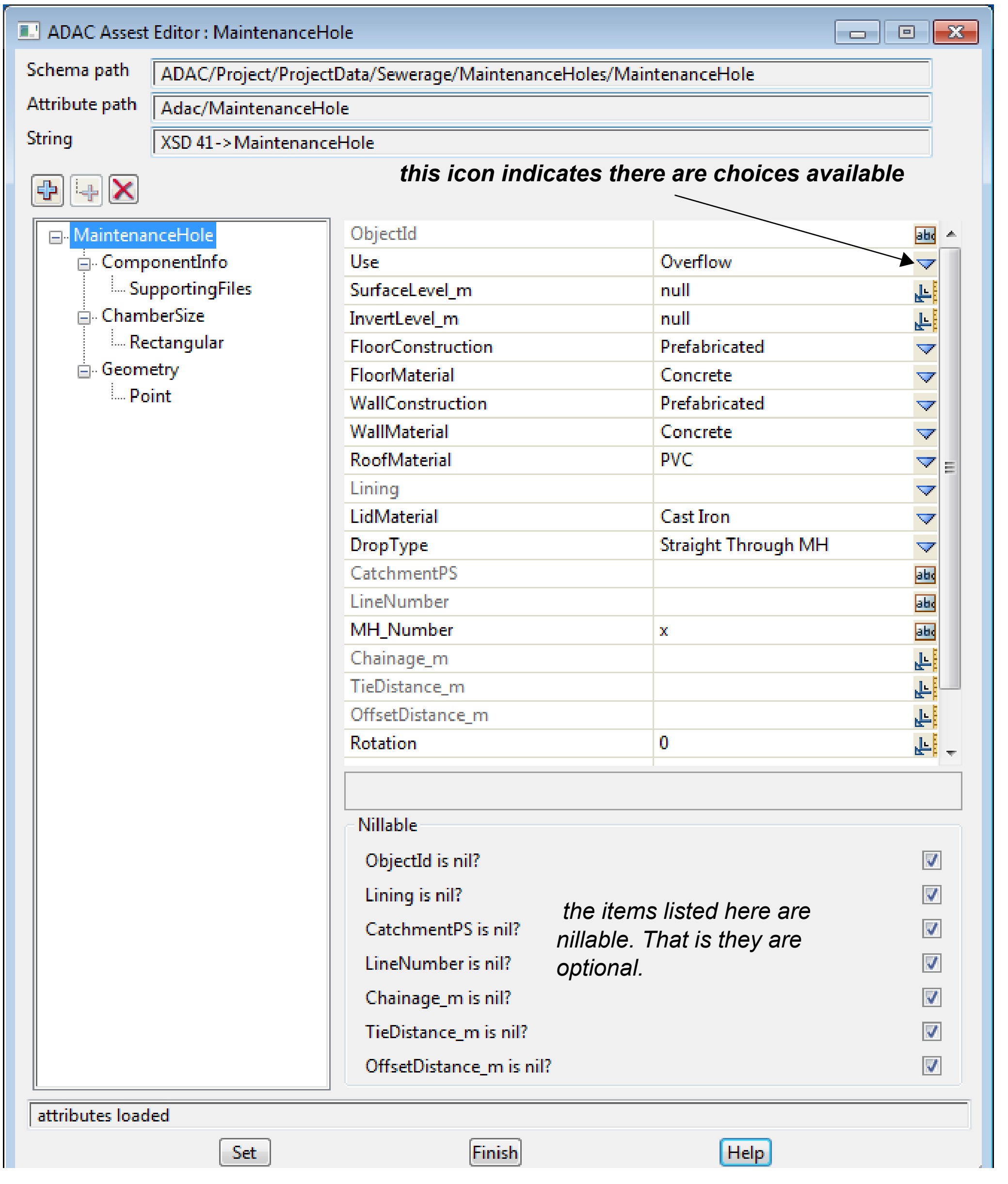
The authorised version of **ADAC Asset Design and As Constructed Schema Help File** (for Version 4.1.0 or Version 4.0.0) is controlled by the **IPWEA** and available on their web site www.engicom.com.au/products/adac2/. This is read to learn what the ADAC Assets all are.

To help in the learning process, a very useful **12d** tool is the option **XSD to Model** which takes a given version of the *ADAC XML Schema* XSD and for **each ADAC Asset** in the **XSD**, creates a green super string with the short ADAC Asset name as its string name, for it string geometry an example of the Geometry in the XSD for the ADAC Asset, and in its string attributes, an ADAC group with default values for each of the ADAC element in the ADAC Asset in the XSD. See 10.4.8.3 XSD to Model .

For example, the first three strings in that model are:



The 12d string geometry shows graphically what type of Geometry is allowed for that ADAC Asset, and editing any of the strings with the ADAC Edit will display the entire ADAC element structure for the ADAC Asset and see what choices are available, which elements, whether elements are nillable, *etc.*



Continue to the next section 10.6.1.2 Know What Your Client Wants in the ADAC XML File or return to 10.6.1 Setting Up Map Files for ADAC .

10.6.1.2 Know What Your Client Wants in the ADAC XML File

The next step is to find out for the Authority you are supplying the ADAC XML file to:

(a) exactly which ADAC Assets you have to supply in the ADAC XML file

(b) for each ADAC Asset, what Types/Uses are wanted

(c) for an ADAC Asset, exactly which elements of the asset they want

(d) for an ADAC Asset and a compulsory (not nillable) element they do not want, what default value can you put in for it.

(e) what exactly are the coordinates in the Geometry.

From this you may discover that you only have a small subset of ADAC Assets, and an even smaller subset of the elements of those ADAC Assets, to worry about.

Now that you know exactly what ADAC Assets you need to provide, and what specific elements inside those ADAC Assets, you are ready to set up a 12d Map File to apply to your strings to mark them as particular ADAC Assets.

Continue to the next section 10.6.1.3 How the ADAC Map File is Used or return to 10.6.1 Setting Up Map Files for ADAC .

10.6.1.3 How the ADAC Map File is Used

In *12d ADAC*, the ***12d Map File*** (or just ***Map File***) is the major method to create an *ADAC Asset* group of attributes for a string (the process referred to as **marking** the string with an *ADAC Asset*).

The power of this method is that once the ***Map File*** has been set up to mark all the required strings as *ADAC Assets*, it runs on any number of strings with no manual intervention.

And once set up, the ***Map File*** is used for every job without modification.

In ***12d Model 11,*** a new section called **Attributes** was added to the **Map File** and the **12d Map Create/File** panel has a section called **Attributes** which is used to create *String*, *Vertex* and *Segment* attributes for a string.

As with the other sections of the **12d Map Create/File** panel, for each of *Attributes>String*, *Attributes>Vertex* and *Attributes>Segment*, there is a grid.

And for each row in the grid, criteria to check strings being processed against.

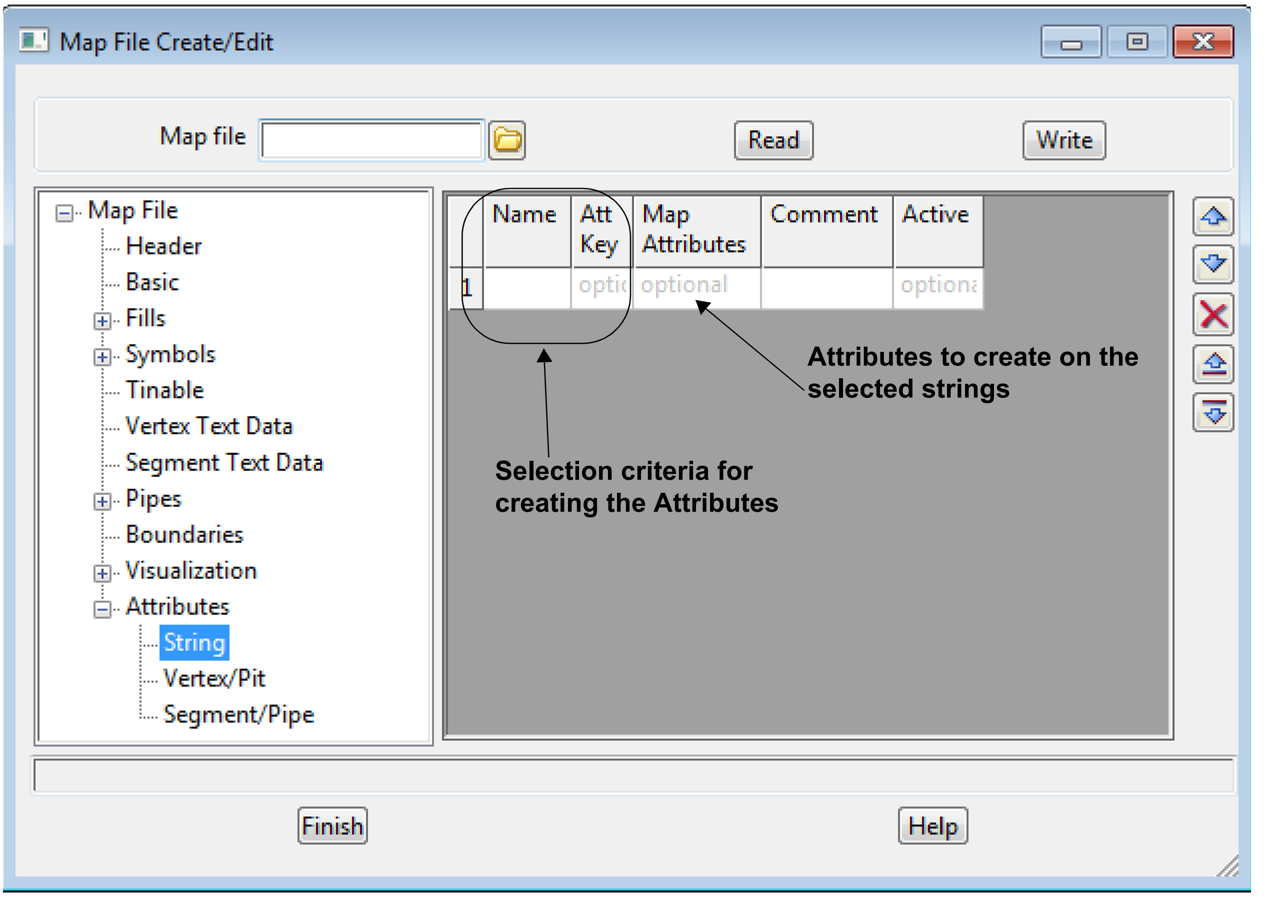
If the criteria are satisfied by a string, then the **Attributes** in the **Map Attributes** column are applied as string/vertex/segment attributes to that string. The next string is then processed.

Once a match to the criteria for a row is made, no more rows of the grid are checked against for that string and the processing starts again for the next string.

The selection criteria for creating *String*, *Vertex* or *Segment* attributes are all different.

(a) Creating String attributes

The selection criteria for creating **String** attributes is by **string name** and **string attributes**.



Apart from drainage and sewer strings, this is the only way strings are marked with and *ADAC Asset* by a ***Map File***.

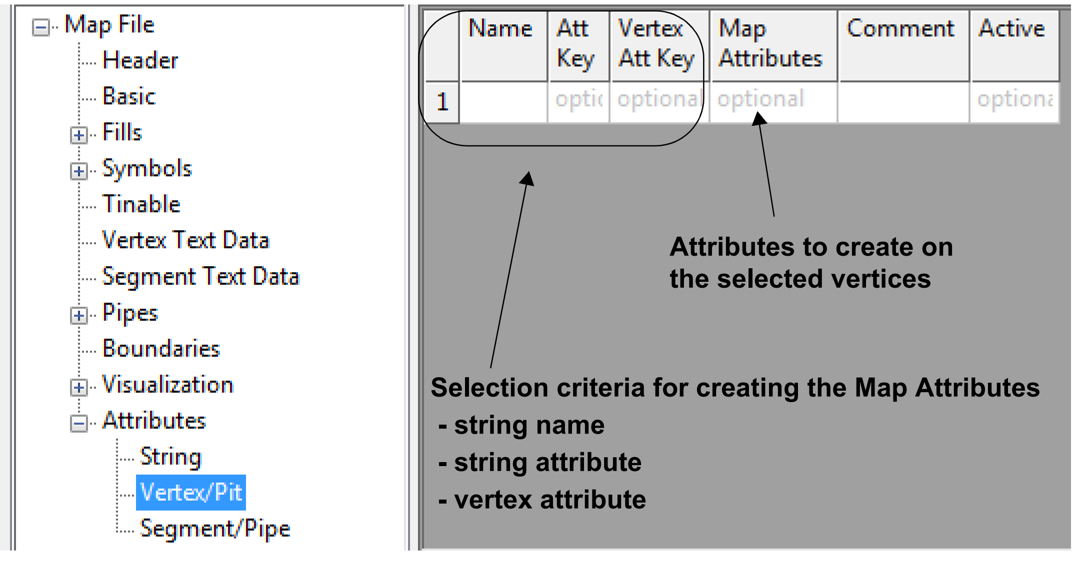
The string names are those created during the Design process, or for Surveyors from the field codes when a survey is reduced in ***12d Model***.

String attributes can be added by ***12d Model*** options, 12dPLs (macros), by hand, or picked up in the field by surveyors and brought in the by the survey reduction

(b) Vertex Attributes

The selection criteria for creating **Vertex** attributes is by **string name**, **string attributes** and **vertex attributes**.

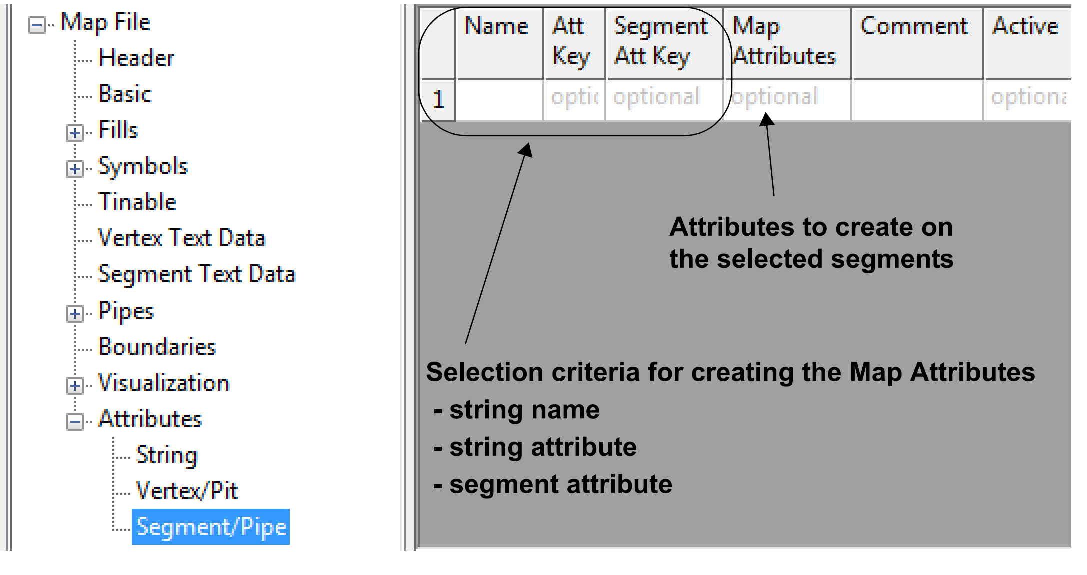
In *12d ADAC*, this is only used for processing drainage and sewer strings. Their vertices are often referred to as pits or maintenance holes.



(c) Segment attributes

The selection criteria for creating **Segment** attributes is by **string name**, **string attributes** and **segment attributes**.

In *12d ADAC*, this is only used for processing drainage and sewer strings. Their segments are often referred to as **pipes**.



So there are two distinct things that need to be looked at when using a ***Map File*** to mark strings as ADAC Asset:

(a) How to set up the ADAC attributes to go in the **Map Attributes** section of the ***Map File*** that will be applied to a string once it passes the section criteria.

See 10.6.1.4 Creating the ADAC Attributes for Map Files

(b) How to set up the selection criteria to uniquely determine the ADAC Asset that the string belongs to.

See 10.6.1.6 Setting Up the Map File Selection Criteria

Continue to the next section 10.6.1.4 Creating the ADAC Attributes for Map Files or return to 10.6.1 Setting Up Map Files for ADAC .

10.6.1.4 Creating the ADAC Attributes for Map Files

The common thing to each of the *Attributes>String*, *Attributes>Vertex* and *Attributes>Segment* grids is that they need sets of ADAC Asset attributes to go in the **Map Attributes** column for each row of the grids.

So we now look at how to best create those ADAC Asset attributes.

You will have noticed when looking at the *ADAC XML Schema Help* File, that its structure is very complex, especially with nillability being used.

And this means that the attribute structure required to replicate the *ADAC Schema* faithfully inside ***12d Model*** must also be very complex and definitely not the sort of thing you want to do by hand.

So ***12d Model*** has two powerful tools that simplify the process of creating the ADAC attributes so that the user never has to know the details of how they are stored inside ***12d Model***.

1. XSD to Map File

The option

File I/O =>ADAC =>Utilities =>XSD to Map File

takes the user specified version of the *ADAC XML Schema* XSD and creates a **Map File** with an ADAC group of attributes in the *Attributes>String section* for each ADAC Asset (see 10.4.8.2 XSD to Map File for more details on how the option works).

Although attribute and attribute substructures can be edited, copied, deleted, *etc.* in the *Attributes>String* section of the **Map File**, it is cumbersome when you are dealing with such a complex attribute structure like ADAC.

Also, as a user, you don't want to learn the complex details of how ***12d Model*** is holding the ADAC data just so that you can make modifications to it.

So although this method is very direct, it is **NOT** recommended.

2. Using XSD to Model and ADAC strings to Map File

At first this method may appear more roundabout but it is much more powerful and needs no knowledge of how the ADAC Attributes are stored in ***12d Model***.

First the option

File I/O =>ADAC =>Utilities =>XSD to model

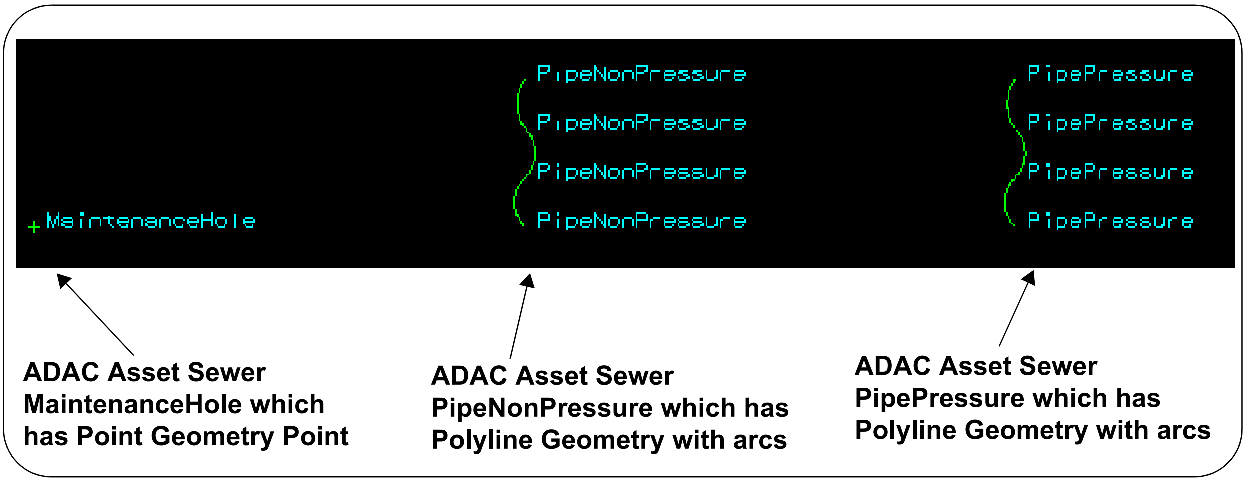
is used to take a user specified version of the *ADAC XML Schema* XSD and produces a model with a 12d string for each of the ADAC Assets (so sixty eight strings) and each string has

(a) a set of ADAC attributes that are typical for that ADAC Asset

(b) the appropriate string geometry that is allowed for the ADAC Asset Geometry.

See 10.4.8.3 XSD to Model for more information.

For example, running the **XSD to model** option for ADAC XML version 4.1.0 Schema, creates a **model** containing sixty-eight strings and the first three strings in that model are:



You can create an ADAC ***12d Map File*** from this model by using the option

File I/O =>ADAC =>Utilities =>ADAC strings to Map File

and you'll get exactly the same ***Map File*** that the option **XSD to Map file** produced.

But what we will be doing is creating **new** ADAC Assets in the model by **making a copy of one of the sixty eight**, and then using the **ADAC =>Edit header/asset** option to bring up the **Edit ADAC** panel (10.4.3 Edit ADAC Header/Asset) and make changes to the copied ADAC Asset.

The benefit of using this editor is that it is totally driven by the *ADAC XML Schema* and so knows all about the ADAC structures and all the ADAC elements.

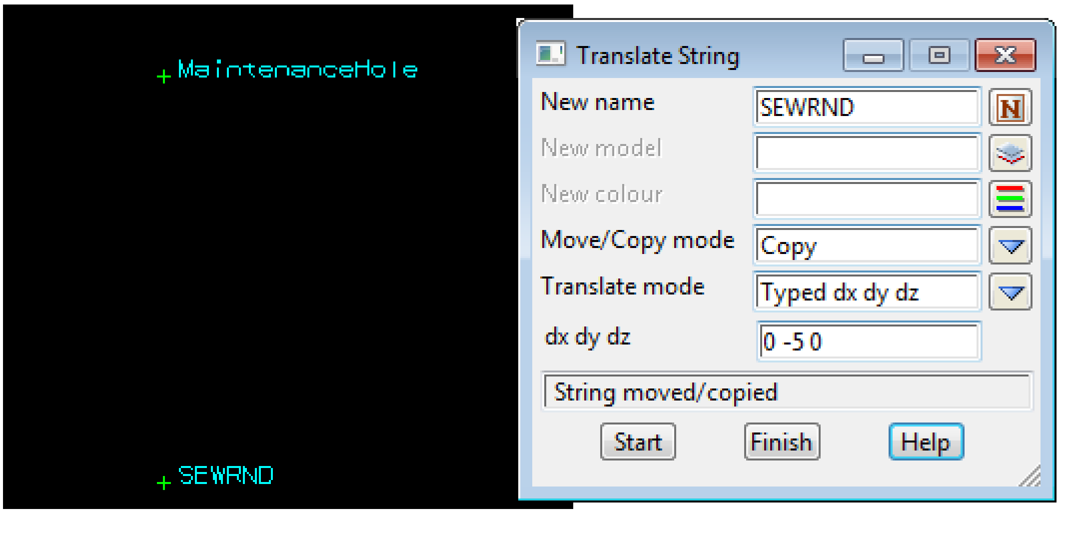
When you run **ADAC Strings to Map File**, each ADAC string creates an entry in either the *Attributes>String*, *Attributes>Vertex/Pit* or *Attributes>Segment/Pipe* section of the ***Map File*** (see 10.4.8.1 ADAC Strings to Map File).

As an example, the **Sewerage>MaintenanceHoles>MaintenanceHole** created in the model has a rectangular chamber and **Use** is *Overflow.* So we want to create a new MaintenanceHole that has a circular chamber with **Use** as *MaintenanceHole*.

The first step is to copy and translate the existing *Maintenancehole* using

Strings =>Strings edit =>Translate

so it isn't on top of the original one, and give the copied string the name SEWRND.

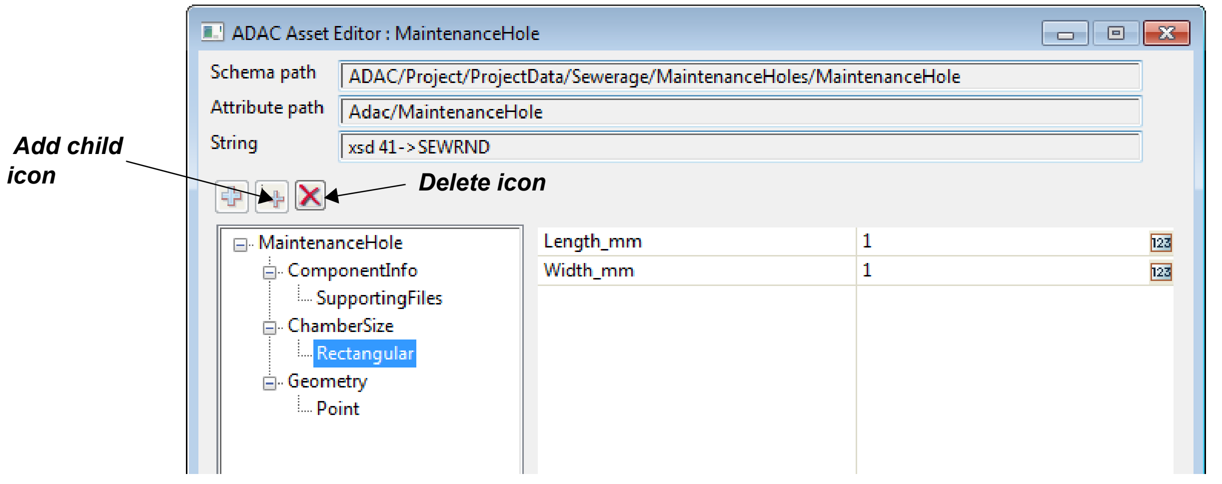


Edit the string SEWRND with the **ADAC =>Edit header/asset** option, and because SEWRND is an ADAC Asset, the **ADAC Asset Editor** for *Sewerage>MaintenanceHoles>MaintenanceHole* is brought up.

Of course because it is a copy, the ADAC attributes are all identical to the string *MaintenanceHole* which had a *Rectangular* chamber.

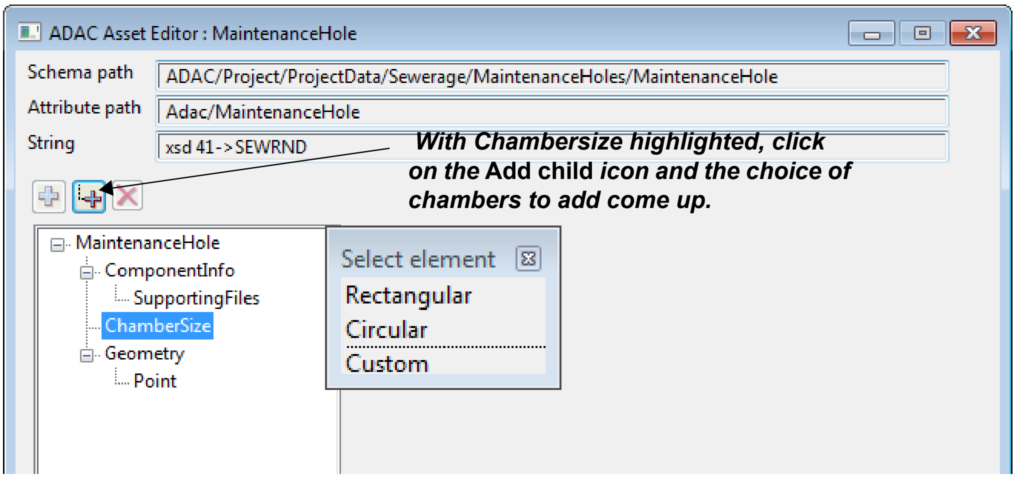
To change from *Rectangular* to *Circular* chamber, you need to highlight *Rectangular* and then click on the **Delete** icon.

-



Then click on the **Add Child** icon and select *Circular* from the **Select Element** panel that pops up.

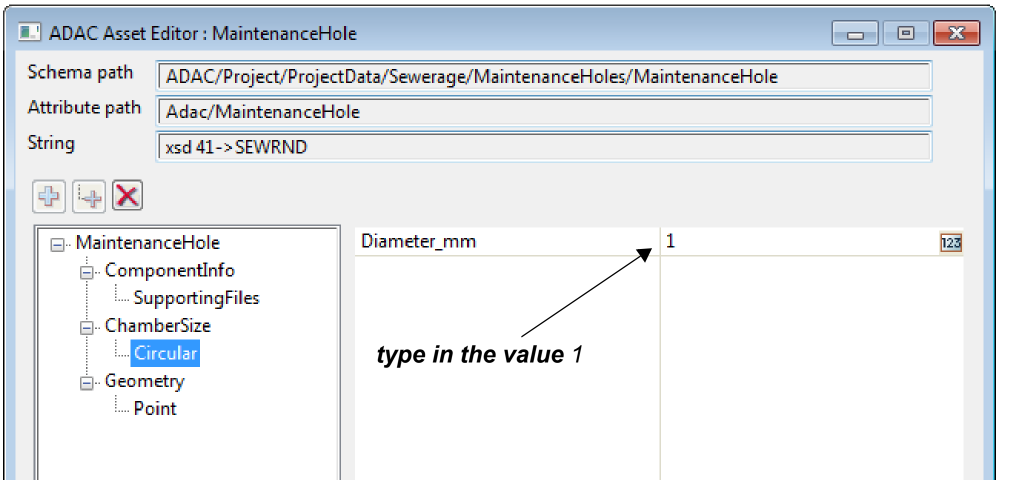
-



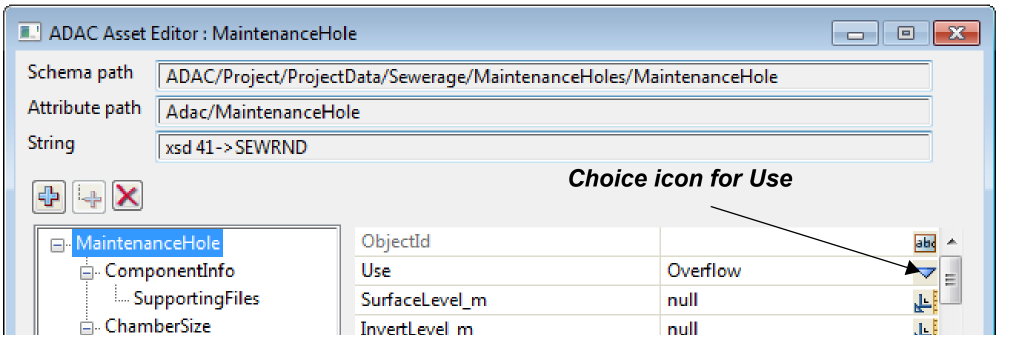
A *Circular* chamber is then added.

A *Circular* chamber has only the one value *Diameter_mm* and type in a value 1.

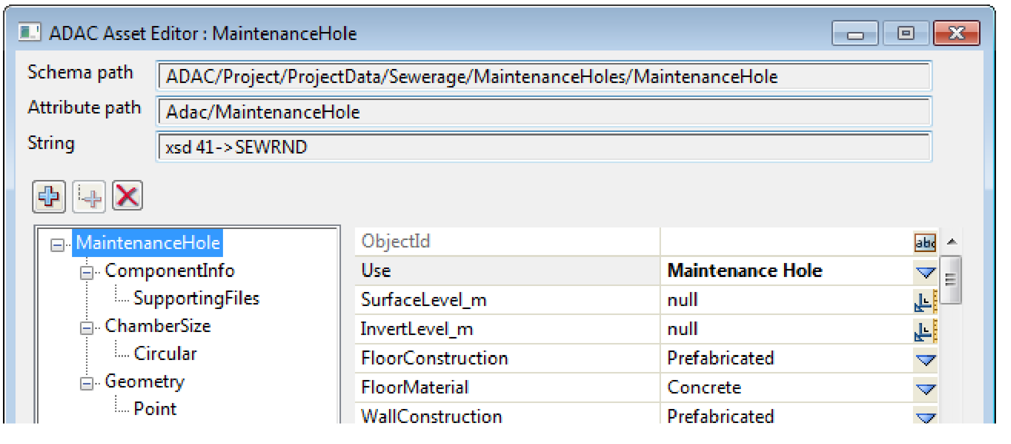
Note that the *ADAC Schema* says that *Diameter_mm* is a positive integer, so you must have a valid value in the field or you get an error message when you move to another node.



Similarly, click on the node **MaintenanceHole** and you'll get the first level of ADAC attributes for **MaintenanceHole** displayed on the right hand side, and you'll see that **Use** is one of them with the value *Overflow*.



Click on the **Choice** icon for **Use** and choose *MaintenanceHole* from the list.

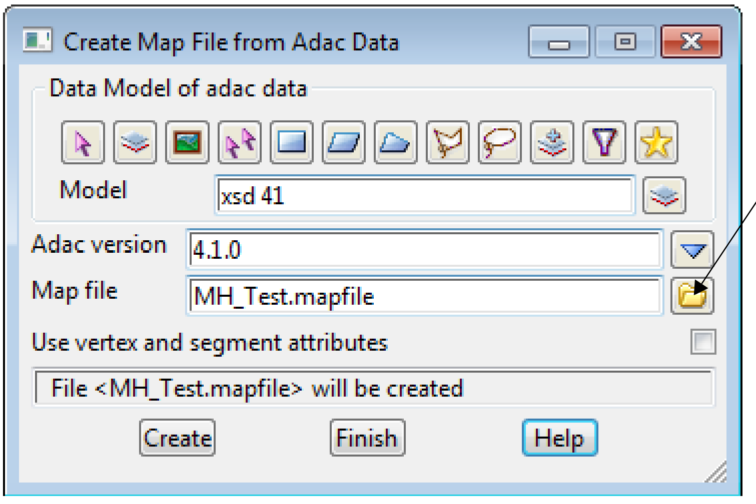


Clicking on **Set** will save the new ADAC attributes to SEWRND.

We will now run the option

File I/O =>ADAC =>Utilities =>ADAC strings to Map File

and select the model containing the SEWRND string, the Map file name as **MH_Test** and the tick box **Use vertex and segment attributes** not ticked:

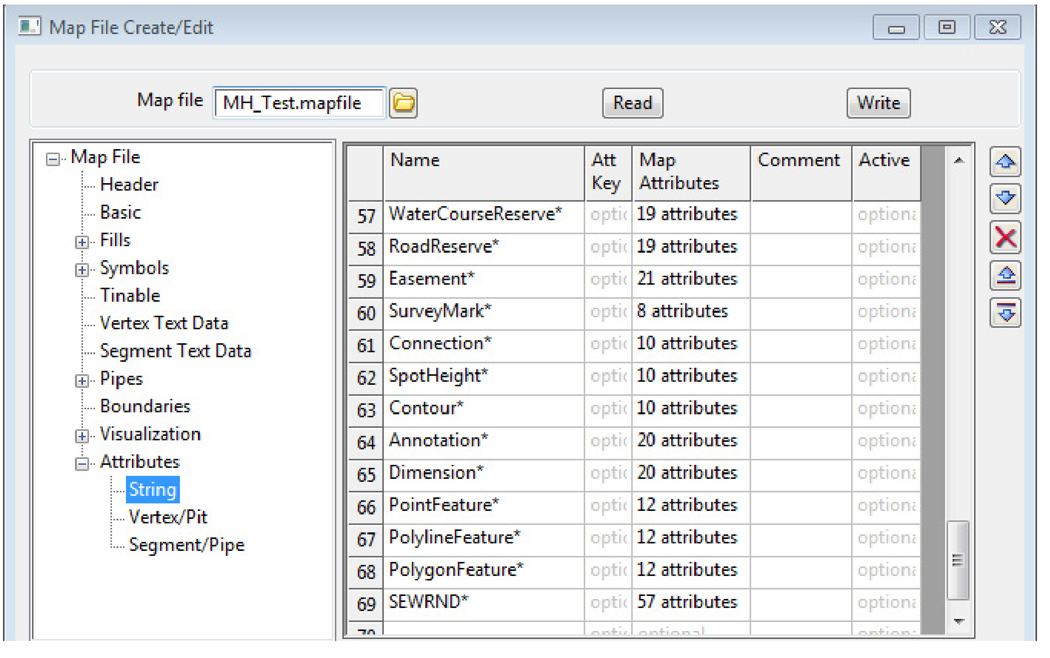


Clicking **Create** produces the ***Map File*** *MH_Test.mapfile*.

Don't click on **Finish**.

After creating the ***Map File,*** clicking on the folder icon to the right of the **Map file** field and selecting **Open** will bring up the ***Map File*** *MH_Test* in the **Map File Create/Edit** panel.

Click on ***Attributes>String*** and scroll to the bottom of the grid and you'll see the entry for SEWRND.



Note: the grid is filled in the order that the strings are added to the model and SEWRND was the last one added. If needed, the row can be moved up by clicking on the **up** arrow icon on the right hand side.

This ***Map file*** is now set up with ADAC Asset attributes in the **Map Attributes** column.

Note

If this map file is applied to a string starting with **SEWRND** then the string will be given the ADAC Asset attributes for **Sewerage>MaintenanceHoles>MaintenanceHole** and it has a circular chamber and **Use** *MaintenanceHole*.

Continue to the next section 10.6.1.5 Notes On Creating the ADAC Attribute Structure or return to 10.6.1 Setting Up Map Files for ADAC .

10.6.1.5 Notes On Creating the ADAC Attribute Structure

1. Although the process just outlined produced an entry in the ***Map File*** for SEWRND, the **more important thing** is that it **created** an **ADAC Attributes structure** in the *Map Attributes* column.

   And we had total control over what was in the ADAC attribute structure simply by using the **ADAC Editor** to set the attributes up exactly as we wanted them without ever needing to know how the attribute structure is stored in ***12d Model***.

   So this is the process we suggest for creating all the ADAC Attribute structures, even if they are never used with the string name you used to create them.

   Permanently keep the model (or models) of the strings with their Attribute Structures so they can be used to generate new ***Map Files*** at any time, and as examples to copy when you are creating new Attribute Structures in the future.

2. In the example we only created a ***Map File*** with an *Attributes>String* section, but the panel **Create Map File from ADAC Data** can also produce entries in the *Attributes>Vertex/Pit* and *Attributes>Segment/Pipe* section for special ADAC strings (see 10.4.8.1 ADAC Strings to Map File).

   However, the important thing is that the ***ADAC Attribute structure*** has been created for a *MaintenanceHole.* It can then be copied and pasted to other **Map Attributes** columns as required.

Continue to the next section 10.6.1.6 Setting Up the Map File Selection Criteria or return to 10.6.1 Setting Up Map Files for ADAC .

10.6.1.6 Setting Up the Map File Selection Criteria

There are three ways we select strings for ADAC in the Attributes section of the ***Map File***.

(a) Use the **string name** only

This is for not for drainage and sewer strings.

See 10.6.1.6.1 Using String Names to Select Strings

(b) Using the **string name** and/or a **string attribute**

This is for not for drainage and sewer strings.

See 10.6.1.6.2 Using String Name and/or String Attributes to Select Strings

(c) For drainage and sewer strings only

Using vertices and segments of the drainage and sewer strings and **vertex names** and **segment names**.

See 10.6.1.6.3 Using Vertices and Segments - Drainage/Sewer Strings

10.6.1.6.1 Using String Names to Select Strings

This is only for strings that are not Drainage or Sewer strings

See 10.6.1.6.1.1 Designers' Approach

See 10.6.1.6.1.2 Surveyor Approach - Using Survey Codes in the Field

10.6.1.6.1.1 Designers' Approach

The first question to ask, for a string that is not a drainage or sewer string, is:

for a string in the design, is the **string name** enough to uniquely identify it as a particular ADAC Asset, or even better, identify it as a particular ADAC Asset and one of its choices for **Type** (or **Use**)?

For example, the string name **Kerb** is enough to identify it as being a **Transport>RoadEges>RoadEdge** but it would not be enough to identify it as a **RoadEge** of **Type B1** as opposed to being Type **B2** or Type **Bitumen**.

The easiest way of overcoming that problem is to have a **different string name** for each **RoadEdge** and **Type** that is required.

For example, **EB** for RoadEdge of type **Bitumen** (a common one), **EB1** for RoadEdge of Type **B1** and so on.

So where ever possible, try and adopt a string naming convention that has a unique name for each of the ADAC Assets of a particular **Type** (or **Use**) that are required by the Authority.

The **Map File** already created works for this situation - just have a model with a string of each of the names and the attributes that go with it and the **ADAC Strings to Map File** option will create the **Map File**.

What if that is not possible and the string naming convention cannot be changed?

For example, the one string name **Kerb** is used for **all** the **RoadEdge** of **Type** B1, B2, B3, SM1, *etc.*

In that case, it would be necessary to use the **string name and** a **User string attribute**, or **just a User string attribute**. See 10.6.1.6.2 Using String Name and/or String Attributes to Select Strings

10.6.1.6.1.2 Surveyor Approach - Using Survey Codes in the Field

Once you know exactly what ADAC Assets, and what parts of those ADAC Assets, you need information on, can you code the information in the field so that once it is read into ***12d Model***, a ***Map File*** using the string name and if that is not enough, string attributes, to determine which ADAC Asset the string represents?

This assignment is often much easier for surveyors because you may be able to easily add extra field codes to distinguish the different ADAC Assets. And you will know what geometry is required for the ADAC Asset, even though in other circumstances you would have picked it up another way.

For example, for **OpenSpace>BoatingFacilities>BoatingFacility**, it only can have a single **Point**. So if all you are giving is ADAC XML, you only have to pick up **one** point.

Also think about the code being able to determine what the ADAC Asset is, but it may provide more information about the ADAC Asset.

In the **OpenSpace>BoatingFacilities>BoatingFacility** example, a compulsory element is **Type** with the choices of *Jetty*, *Pier*, *Ramp* or *Spillways*.

You could set up your codes so not did you know it was a BoatingFacility but also what **Type**.

For example, the codes could be

BFJ is an **OpenSpace>BoatingFacilities>BoatingFacility** of **Type** *Jetty*.

BFP is an **OpenSpace>BoatingFacilities>BoatingFacility** of **Type** *Pier*.

BFR is an **OpenSpace>BoatingFacilities>BoatingFacility** of **Type** *Ramp*.

BFS is an **OpenSpace>BoatingFacilities>BoatingFacility** of **Type** *Spillway*.

The **Map File** already created works for this situation - just have a model with a string of each of the names and the attributes that go with it and the **ADAC Strings to Map File** option will create the **Map File**.

What if that is not possible and the string naming convention can not be changed?

For example, the one string name **Kerb** is used for **all** the **RoadEdge** of **Type** B1, B2, B3, SM1, *etc.*

In that case, it would be necessary to use the **string name and** a **User string attribute**, or **just a User string attribute**. See 10.6.1.6.2 Using String Name and/or String Attributes to Select Strings

10.6.1.6.2 Using String Name and/or String Attributes to Select Strings

A string has three types of *User* defined attributes

(a) string attributes - there is one set of these for the whole sting

(b) vertex attributes - there is a set for each vertex

(c) segment attributes - there is a set for each segment.

However **only string attributes** are used by the ADAC ***Map File***.

User string attributes can be created by ***12d Model*** options, 12PLs (macros), manually, or if you are a surveyor, by picking them up in the field.

See

10.6.1.6.2.1 Entering User Attributes in the Field
10.6.1.6.2.2 Running a 12d Option or 12dPL that Produce User Attributes
10.6.1.6.2.3 Manually Entering User Attributes
10.6.1.6.2.4 Using the Name and Att Key in the ADAC Map File

10.6.1.6.2.1 Entering User Attributes in the Field

As well as using field codes, a surveyor may also have the capability of picking up attributes in the field.

An example of when attributes are useful is when picking up a **WaterSupply>Pipes>Pipe** - its diameter may be entered as an attribute.

Or when picking up a **Sewerage>MaintenanceHoles>MaintenanceHole**, which can only have a **Point** for its **Geometry**, you could enter the width and length as attributes.

Note that some other survey software can only produce vertex attributes. So if any of those attributes are to be used with the ***Map File***, a method must be found to turn them into string attributes.

If you can do field pick ups with attributes, even if the attributes are not used with the ***ADAC Map File***, they may be very useful in the step *Update ADAC Elements from User Attributes* that is further along in the ADAC chain. See 10.5.1.2.5 Update ADAC Elements from User Created Attributes on the String - Survey .

Continue to the next section 10.6.1.6.2.2 Running a 12d Option or 12dPL that Produce User Attributes or return to 10.6.1.6.2 Using String Name and/or String Attributes to Select Strings .

10.6.1.6.2.2 Running a 12d Option or 12dPL that Produce User Attributes

Any ***12d Model*** options that create *User Attributes* will be document with what those *User attributes* are.

For example, 10.4.9.2.3 Generate ADAC Road Edge Types splits a string into user defined chainage ranges and adds a User string attribute named **RoadEdge_Type** for each chainage range. That is, the 12dPL spits the Kerb into distinct parts, each with the same name Kerb but each part having a *User string attribute* called *RoadEdge_Type* which has only one of the RoadEdge **Types** as its value.

So this is a method for transcending the problem of having only the single string name of **Kerb**.

Hopefully any 12dPLs (macros) that you use are also documented, and in that it describes how the 12dPL works and what User string attributes it sets.

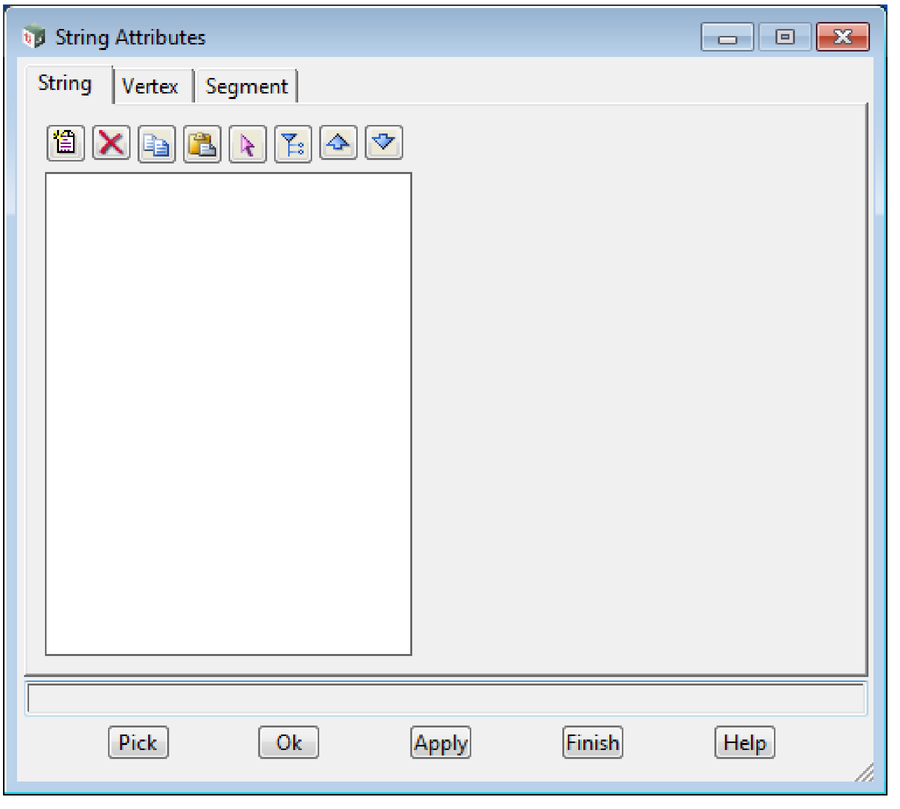
Continue to the next section 10.6.1.6.2.3 Manually Entering User Attributes or return to 10.6.1.6.2 Using String Name and/or String Attributes to Select Strings .

10.6.1.6.2.3 Manually Entering User Attributes

If you are adding a *User string attribute* manually, (or wanting to display existing attributes for the string), pick the option

Strings =>Properties =>Attributes

This brings up the **Strings Attributes** panel



Click on **Pick** and then select the string to add attributes to. Any existing string, vertex or segment attributes are displayed by clicking on the **String**, **Vertex** or **Segment** tabs.

With **String Attributes**, User string attributes of virtually any name and value can be created to distinguish between Types of an *ADAC Asset.*

But this is a very manual process.

If you need to do it regularly, then it is best to write a small 12d PL (macro) that lets you pick a string, asks for the value for the attribute and then updates the string attribute. This will save lots of time and errors.

User string attributes are used to uniquely identify the type of *ADAC Asset* that a string represents, but they are also very useful for supply values for elements inside an ADAC Asset as well. See 10.5.1.2.5 Update ADAC Elements from User Created Attributes on the String - Survey and 10.5.2.2.6 Update ADAC Elements from User Created Attributes - Design .

Continue to the next section 10.6.1.6.2.4 Using the Name and Att Key in the ADAC Map File or return to 10.6.1.6.2 Using String Name and/or String Attributes to Select Strings .

10.6.1.6.2.4 Using the Name and Att Key in the ADAC Map File

The ***Map File*** that is created with the option ***ADAC Strings to Map File*** automatically creates a ***Map File*** for matching the string name against the **Name** in the ***Map File***.

But when using a **string name** AND a **string attribute**, or **just a string attribute,** then you need to use both the **Name** column and the **Att Key** (Attribute key) column in the grid for the *Attributes>String* of the ***Map File***.

Luckily, there is a way of setting up an ADAC string so that the **ADAC string to Map File** option creates both the **Name** column **and** the **Att Key** column entries - you simply need to add the required **Att Key** attribute with its type and value as the **string attribute** of the **ADAC string**.

So creating the row of the ***Map File*** when the criteria involves a **string name** and a **string attribute** is a two step process.

Step 1.Create the required ADAC attribute structure

This has already been described in the section 10.6.1.4 Creating the ADAC Attributes for Map Files .

Step 2.Edit the ADAC string and add the Att Key criteria

Here we take the **ADAC string** produced in *Step 1* and manually edit it with the **Strings Attributes** panel and add the required string attribute that will be copied to the **Att Key** column, and hence used as part of the criteria for selecting a string.

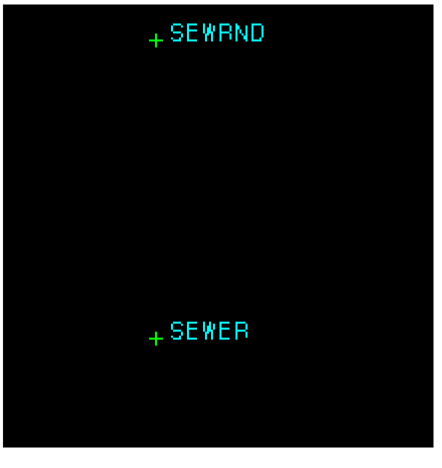
To explain the process of adding a string attribute that becomes an **Att Key**, we will build on the previous example and

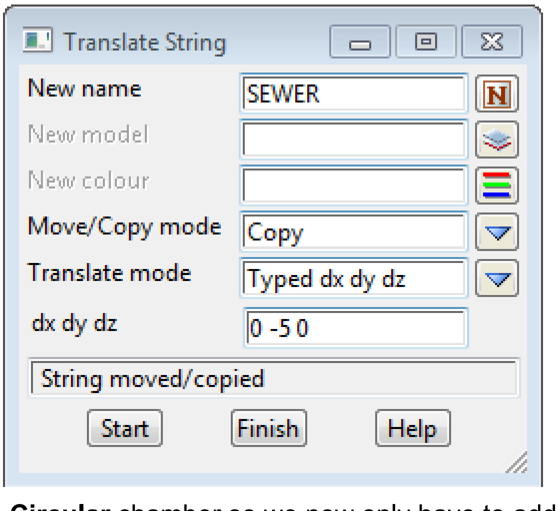
Instead of having a **string name** SEWRND to indicate the *Sewerage>MaintenanceHoles>MaintenanceHole* has a circular chamber, we only have the **nondescript name** SEWER and a **string attribute** called **Chamber** which has the text value of **Circular** or **Rectangular**.

To do the **Circular** case, copy and translate the existing ADAC *Sewerage>MaintenanceHoles>MaintenanceHole* called ***SEWRND*** using the option

Strings =>Strings edit =>Translate

and name the copied string **SEWER**.



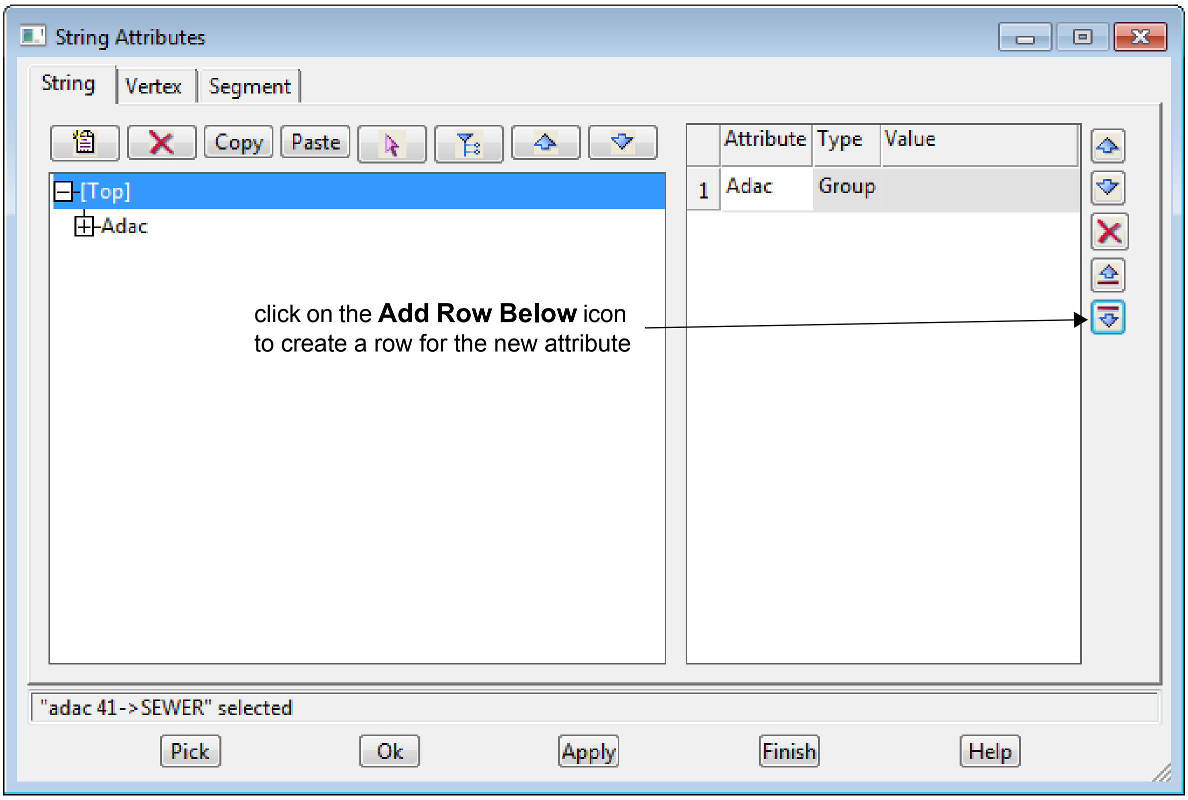


The ADAC string SEWER already has a **Circular** chamber so we now only have to add the text string attribute **Chamber** with the value **Circular**.

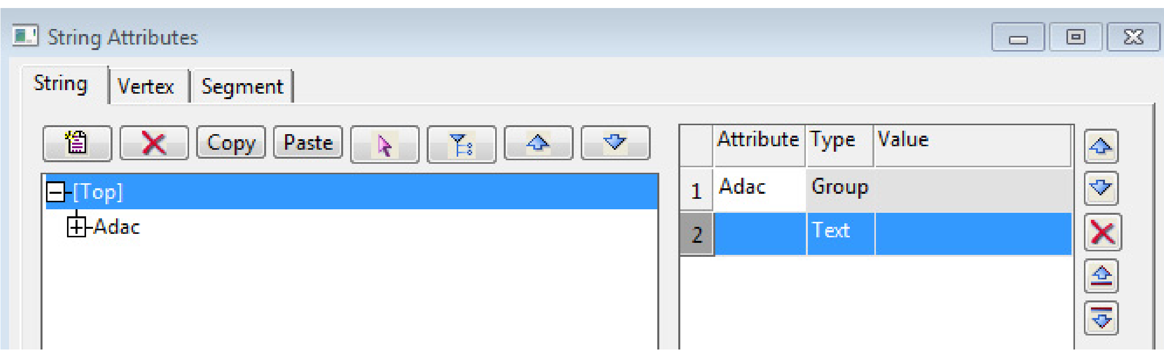
So the add attributes to the SEWER string, bring up **String Attributes** panel selecting the option

Strings =>Properties =>Attributes

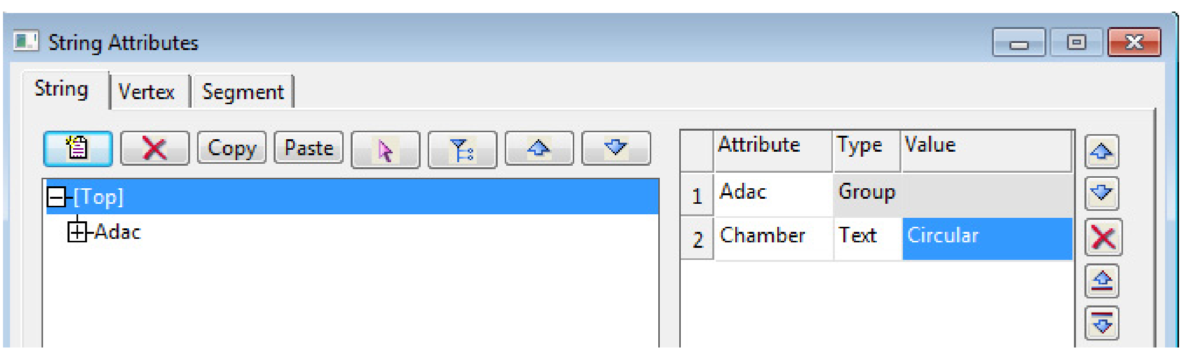
Click on **Pick** and select the string SEWER.



Highlight the node **[Top]** and then click on the **Add Row Below** icon to create a new row to use for the attribute **Circular**.



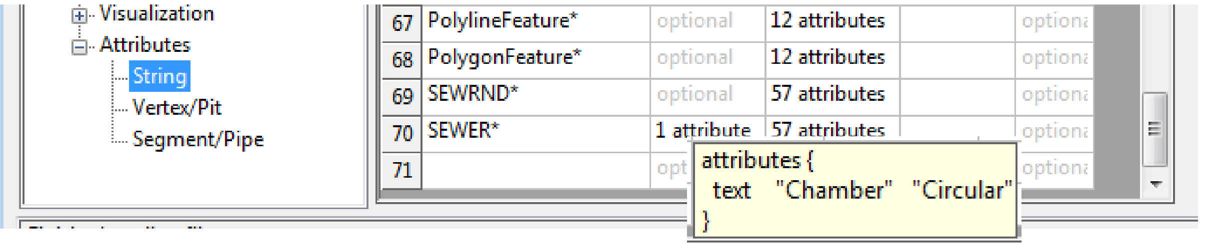
Type in the attribute name **Chamber** and the value **Circular** and then click on the **Apply** button **twice** to add the attribute to the string.



To create the ***Map File,*** we repeat the process used in 10.6.1.4 Creating the ADAC Attributes for Map Files which is to run the option

File I/O =>ADAC =>Utilities =>ADAC strings to Map File

to create the ***Map File*** and then open the ***Map File*** and go to the bottom of the *Attributes>String* grid and hover over the **Att Key** column to see that the attribute Chamber is defined.



So this ***Map File*** will select any string with the string name starting with SEWER and with a text attribute called **Chamber** with the value **Circular**, and give it the ADAC attributes for a *Sewerage>MaintenanceHoles>MaintenanceHole* with a Circular chamber.

As an alternative to having the attribute Chamber as part of the ADAC string SEWER, it is possible to add the Att Key values into the ***Map File*** by hand. So **Step 2** would be replaced by:

Step 2'.Edit the Map File and add the Att Key criteria

Here we take the **Map File** produced in *Step 1* and manually edit it with the **Map File Create/Edit** panel and add the required **Att Key** in for the string attribute that is to be used as part of the criteria for selecting a string. This will now be described.

The problem with this approach is this that if the Map File is generated again from the ADAC strings, any manual edits such as in Step 2' will be lost and have to be repeated, or some other more complicated strategy employed such a keeping ADAC strings that require an **Att Key** for identification in a **separate** model.

So it is recommended NOT to use **Step 2'** but to use **Step 2** where a string attribute is added to the to the ADAC string.

Continue to the next section 10.6.1.6.3 Using Vertices and Segments - Drainage/Sewer Strings or return to 10.6.1.6 Setting Up the Map File Selection Criteria .

10.6.1.6.3 Using Vertices and Segments - Drainage/Sewer Strings

The Designers drainage and sewer strings are unique in that they hold data for four different ADAC Assets and the four *ADAC Assets* are known straight away.

1. Each **pit** of a Drainage string is a *StormWater>Pits>Pit*.
2. Each **pipe** of a Drainage string is a *StormWater>Pipes>Pipe*
3. Each **pit (maintenance hole)** of a Sewer string is a *Sewerage>MaintenanceHoles>MaintenanceHole*.
4. Each **pipe** of a Sewer string is a *Sewerage>PipesNonPressure>PipeNonPresssure.*

So for drainage or sewer string, each **vertex** and each **segment** of the string needs to be marked as a *ADAC Asset*. Consequently, unlike all the other strings for ADAC, the ***Attributes>Vertex*** and ***Attributes>Segment*** sections of the ***Map File*** are used.

Unfortunately a ***Map File*** doesn't know the difference between a drainage and a sewer string as it is an internal property of the string so just before the ***Map File*** is applied in the *Design Chain*, the 12dPL *Create_or_delete_temporary_sewerage_attribute_panel* is run and for each drainage and sewer string, it creates a **string attribute** called **sewertype** of type Integer with the value **0** if it is a **drainage string**, or **1** if it a **sewer** string.

Being a **string attribute**, the attribute **sewertype** CAN be used in a ***Map File***.

So when going through drainage and sewer strings, if it is a

(i) vertex with an integer string attribute **sewertype** of value **1** then it is a *Sewerage>MaintenanceHoles>MaintenanceHole*

(ii) vertex with an integer string attribute **sewertype** of value **0** then it is a *StormWater>Pits>Pit*

(iii) segment with an integer string attribute **sewertype** of value **1** then it is a *Sewerage>PipesNonPressure>PipeNonPresssure*.

(iv) segment with an integer string attribute **sewertype** of value **0** then it is a *StormWater>Pipes>Pipe*.

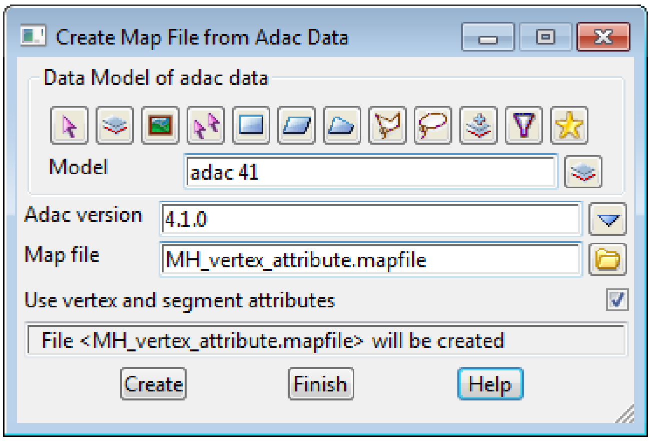
Important Note - Actually a ***Map File*** can't tell the difference between ANY string types, so the presence of a **string attribute** called **sewertype** with value **0** or **1** can be used in a ***Map File*** to select only a sewer string or a drainage string.

Knowing this we can now set up the *Attributes>Vertex* and *Attributes>Segment* sections of the ***Map File*** to mark the vertices and segments of drainage and sewer strings as ADAC Assets

So far we have only used the option

File I/O =>ADAC =>Utilities =>ADAC strings to Map File

to create rows in the *Attributes>String* section of the ***Map File*** but if we click on the tick box **Use vertex and segment attributes** in the **Create Map File from ADAC Data** panel,



then

the ADAC attributes for ADAC *Sewerage>MaintenanceHoles>MaintenanceHole's* and ADAC *StormWater>Pits>Pit's* will be placed in the *Attributes>Vertex/Pit* section of the ***Map File*** with the **Att Key** set with the appropriate **sewertype** attribute value

and

the ADAC attributes for ADAC *Sewerage>PipesNonPressure>PipeNonPressure's* and ADAC *StormWater>Pipes>Pipe's* will be placed in the *Attributes>Segment/Pipe* section of the ***Map File*** with the **Att Key** set with the appropriate **sewertype** attribute value

(For more information on **Use vertex and segment attributes** in the **Create Map File from ADAC Data** panel see 10.6.1.4 Creating the ADAC Attributes for Map Files)

A string attribute on the ADAC Asset with also go through as a Vertex Att Key or Segment Att Key, and the string name is written as a comment.

For examples of the vertex and the segment cases, see

10.6.1.6.3.1 Setting Up the Attributes>Vertex Section of the Map File
10.6.1.6.3.2 Setting Up the Attributes>Segment Section of the Map File
10.6.1.6.3.3 Bonuses for Drainage and Sewer Strings

10.6.1.6.3.1 Setting Up the Attributes>Vertex Section of the Map File

The strings for ADAC Assets that create an entry in the *Attributes>String* section of the ***Map File*** use the string **name** and **Att Key** as a method of deciding which strings have the ***Map Attributes*** applied to them.

But the ADAC strings for *Sewerage>MaintenanceHoles>MaintenanceHole* or *StormWater>Pits>Pit* which create an entry in the *Attributes>Vertex/Pit* section of the ***Map File***, already have the **Name** column set to * and the **Att Key** is used with **sewertype** to select only drainage or sewer strings.

So the only part of the *Attributes>Vertex/Pit* row left help to differentiate between different types of maintenance holes/pits, is the **Vertex Att Key**.

Luckily the vertices of ***12d Model*** drainage and sewer string usually have a text **Vertex attribute** called **pit_type**, or some other vertex attributes set during creation, and these can be used to select **Map Attributes** that are more appropriate to that vertex.

With that in mind, the **Create Map File from ADAC Data** panel with **Use vertex and segment attributes** ticked **on**, copies the string attribute of the ADAC Asset *Sewerage>MaintenanceHoles>MaintenanceHole* or *StormWater>Pits>Pit*, to the **Vertex Att Key**.

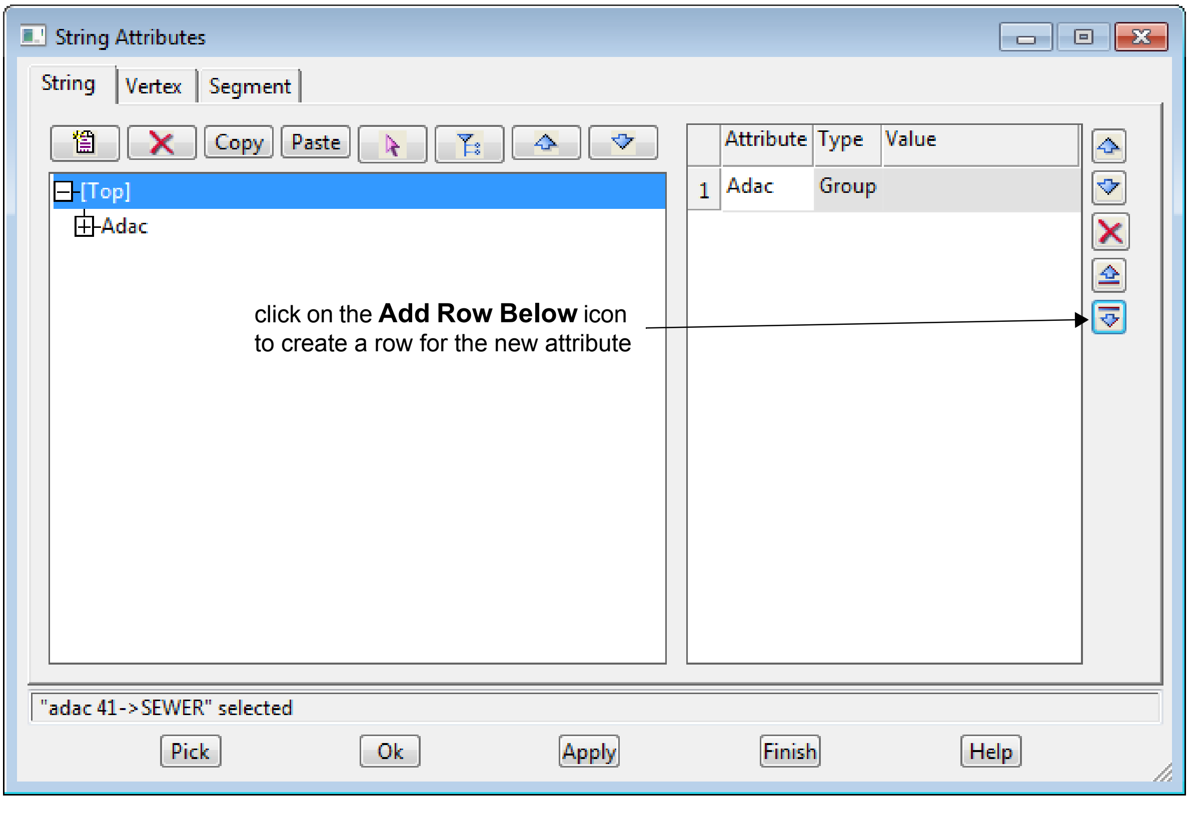
Note: To allow the user to identify which *ADAC Asset* string creates a particular row in the *Attributes>Vertex/Pit* grid, the string **name** is copied into the **Comment** column.

For example, if you wanted the *ADAC Attribute* group on the string SEWER to be applied only to vertices of a sewer string with the **pit_type** attribute equal to **B3**, we would use the **String Attributes** panel to create a text string attribute called **pit_type** with value **B3**.

To add attributes to the SEWER string, bring up **String Attributes** panel by selecting the option

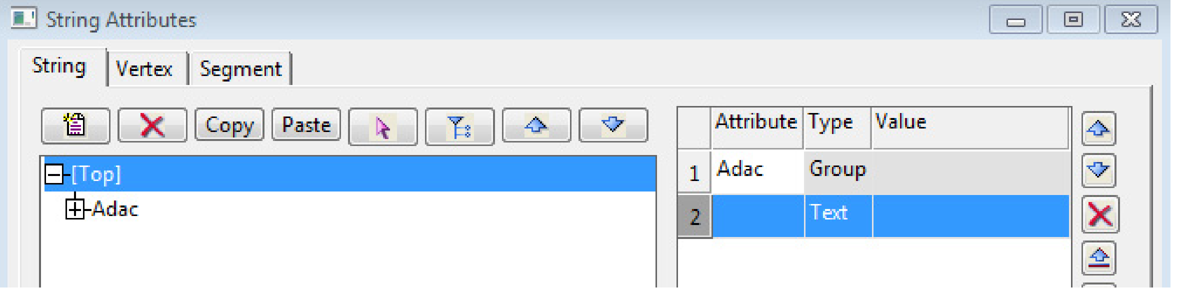
Strings =>Properties =>Attributes

Click on **Pick** and select the string SEWER.

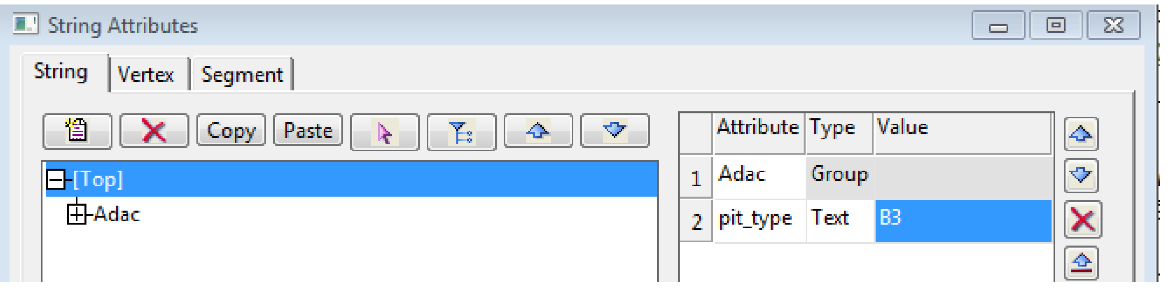


For the **String** tab, highlight the node **[Top]** and then click on the **Add Row Below** icon to create a new row to use for the attribute **pit_type**.

Note: if there are other first level attributes other than ADAC, highlight delete them.



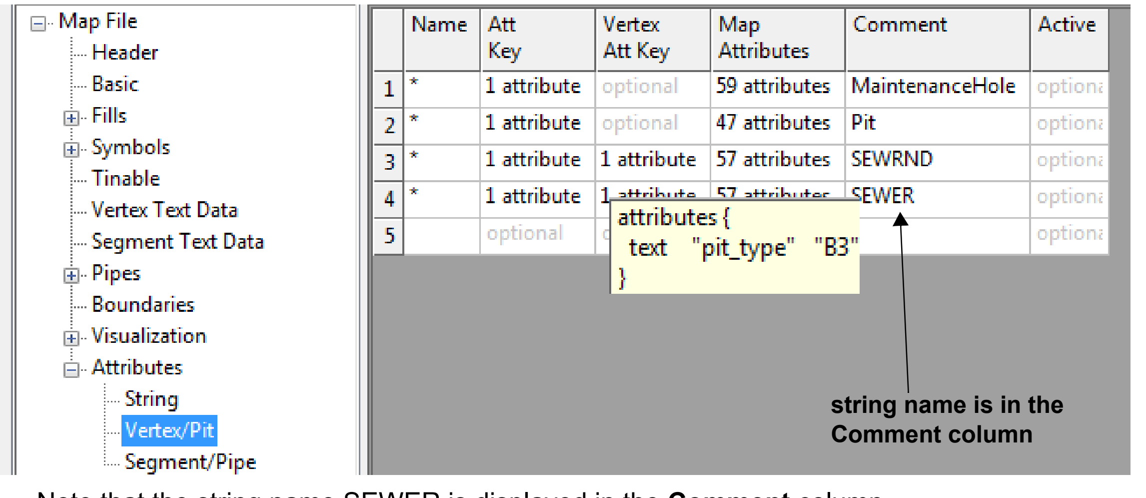
Type in the attribute name **pit_type** and the value **B3** and then click on the **Apply** button **twice** to add the attribute to the string.



To create the ***Map File,*** we repeat the process used in 10.6.1.4 Creating the ADAC Attributes for Map Files which is to run the option

File I/O =>ADAC =>Utilities =>ADAC strings to Map File

to create the ***Map File*** and then open the ***Map File*** and go to the bottom of the *Attributes>Vertex* grid and hover over the **Vertex Att Key** column to see that the attribute **pit_type** is defined.



Note that the string name SEWER is displayed in the **Comment** column.

So this combination of **Name**, **Att Key** and **Vertex Att Key** in the ***Map File*** will select the vertex of any sewer string with a text attribute called **pit_type** with the value **B3**, and give it the ADAC attributes in the **Map Attributes** column.

The ADAC Asset attributes for a **pit** on a **drainage** string are set up the same way except that the value for **sewertype** is **0** and the ADAC Asset attributes to use are those of a *StormWater>Pits>Pit*.

Continue to the next section 10.6.1.6.3.2 Setting Up the Attributes>Segment Section of the Map File or return to 10.6.1.6.3 Using Vertices and Segments - Drainage/Sewer Strings .

10.6.1.6.3.2 Setting Up the Attributes>Segment Section of the Map File

The strings for ADAC Assets that create an entry in the *Attributes>String* section of the ***Map File*** use the string **name** and **Att Key** as a method of deciding which strings have the ***Map Attributes*** applied to them.

But the ADAC strings for *Sewerage>PipesNonPressure>PipeNonPressure* or *StormWater>Pipes>Pipe* which create an entry in the *Attributes>Segment/Pipe* section of the ***Map File***, already have the **Name** column set to ***** and the **Att Key** is used with **sewertype** to select only drainage or sewer strings.

So the only part of the *Attributes>Segment/Pipe* row left help to differentiate between different types of pipes, is the **Segment Att Key**.

Luckily the segments of ***12d Model*** drainage and sewer string usually have a text **Segment attribute** called **pipe_type**, or some other segment attributes set during creation, and these can be used to select **Map Attributes** that are more appropriate to that segment.

With that in mind, the **Create Map File from ADAC Data** panel with **Use vertex and segment attributes** ticked **on**, copies the string attribute of the ADAC Asset *Sewerage>PipesNonPressure>PipeNonPressure* or *StormWater>Pipes>Pipe*, to the **Segment Att Key**.

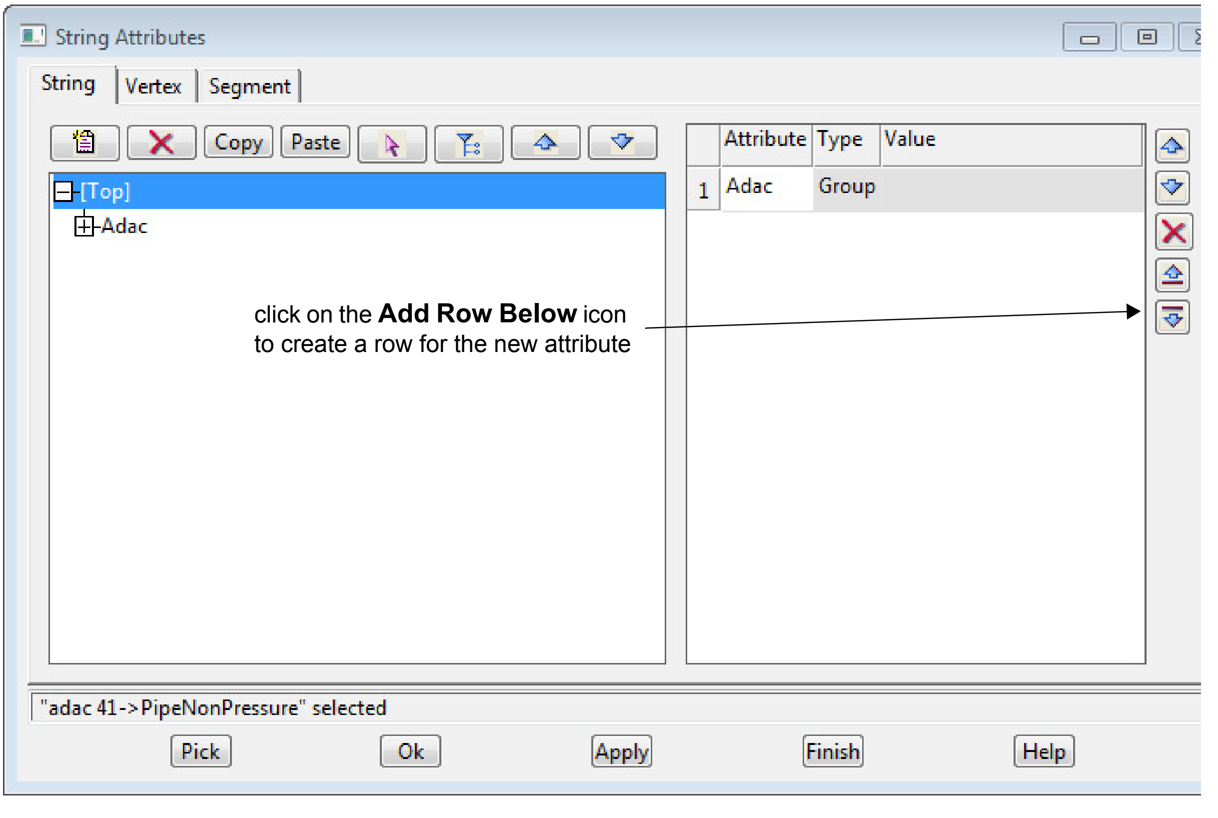
Note: To allow the user to identify which *ADAC Asset* string creates a particular row in the *Attributes>Segment/Pipe* grid, the string **name** is copied into the **Comment** column.

For example, if you wanted the *ADAC Attribute* group on the string called **PipeNonPressure** to be applied only to segments of a sewer string with the **pipe_type** attribute equal to **P3**, we would use the **String Attributes** panel to create a text string attribute called **pipe_type** with value **P3**.

To add attributes to the **PipeNonPressure** string, bring up **String Attributes** panel by selecting the option

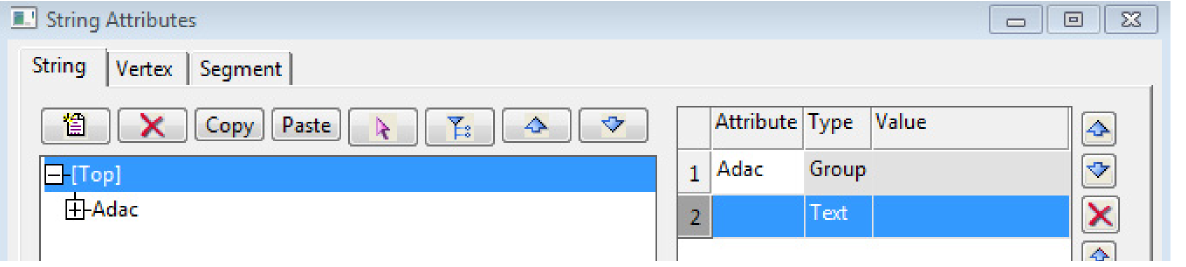
Strings =>Properties =>Attributes

Click on **Pick** and select the string **PipeNonPressure**.

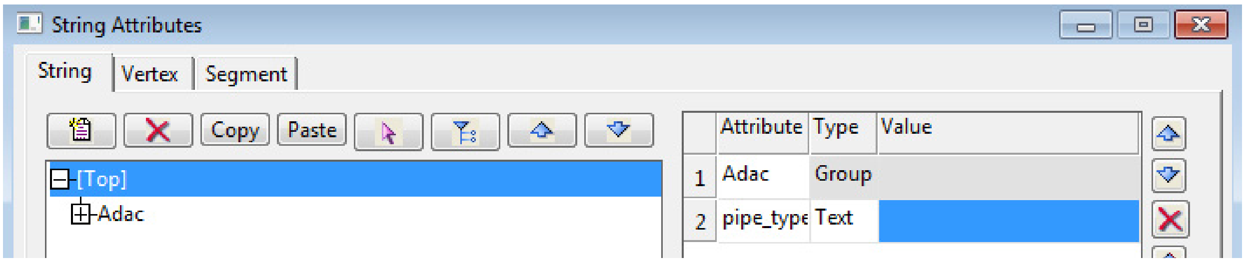


For the **String** tab, highlight the node **[Top]** and then click on the **Add Row Below** icon to create a new row to use for the attribute **pipe_type**.

Note: if there are other first level attributes other than ADAC, highlight delete them.



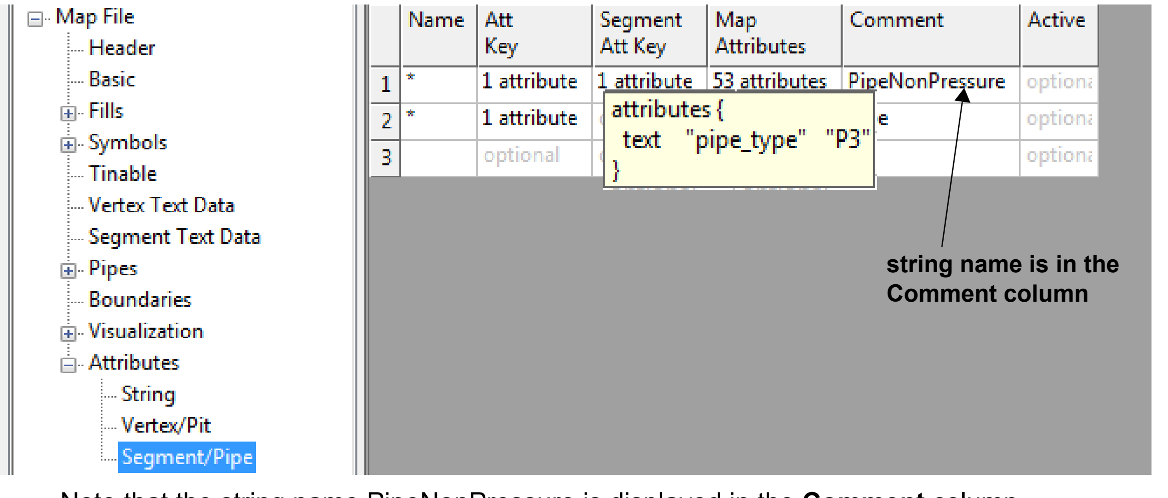
Type in the attribute name **pipe_type** and the value **P3** and then click on the **Apply** button **twice** to add the attribute to the string.



To create the ***Map File,*** we repeat the process used in 10.6.1.4 Creating the ADAC Attributes for Map Files which is to run the option

File I/O =>ADAC =>Utilities =>ADAC strings to Map File

to create the ***Map File*** and then open the ***Map File*** and go to the bottom of the *Attributes>Segment* grid and hover over the **Segment Att Key** column to see that the attribute **pipe_type** is defined.



Note that the string name PipeNonPressure is displayed in the **Comment** column.

So this combination of **Name**, **Att Key** and **Segment Att Key** in the ***Map File*** will select the segment of any sewer string with a text attribute called **pipe_type** with the value **P3**, and give it the ADAC attributes in the **Map Attributes** column.

The ADAC Asset attributes for a **pipe** on a **drainage** string are set up the same way except that the value for **sewertype** is **0** and the ADAC Asset attributes to use are those of a *StormWater>Pipes>Pipe*.

Continue to the next section 10.6.1.6.3.3 Bonuses for Drainage and Sewer Strings or return to 10.6.1.6.3 Using Vertices and Segments - Drainage/Sewer Strings.

10.6.1.6.3.3 Bonuses for Drainage and Sewer Strings

An ADAC bonus for designers with drainage and sewer strings is that each pit and pipe already has most of the values needed for the ADAC Assets.

For example, a sewer string pit (maintenance hole) knows if the chamber is rectangular or circular, and in both cases, what the dimensions of the chamber and the invert level and the surface level are. The pit name usually contains the MH_Number and the string name is the LineNumber.

Also from its *pit type*, the **Use** and most of the other elements of the ADAC Asset *Sewerage>MaintenanceHoles>MaintenanceHole* are known.

Similarly for a sewer string pipe, and the pits and pipes of a drainage string.

And a further bonus is that many of the extra attribute values **do not** have to be the values set for the *ADAC Asset* set by the ***Map File*** because later on in the Design chain, the *12dPL* ***ADAC_Update_attributes_from_drainage_data_panel.4do*** is run and it updates as many of the ADAC elements for the asset as possible.

For example, it doesn't matter if the ***Map File*** marked a vertex as having a Circular chamber or a Rectangular chamber because the *12dPL* ***ADAC_Update_attributes_from_drainage_data_panel.4do*** will look at the **actual** chamber type of the pit and update the *ADAC Asset* attributes on the pit with the proper chamber type and size.

For more information on ***ADAC_Update_attributes_from_drainage_data_panel.4do***, see 10.5.2.2.4 Update ADAC Elements from Drainage/Sewer String Properties .

Continue to the next section 10.6.1.7 More on Creating the ADAC Map Files or return to 10.6.1.6.3 Using Vertices and Segments - Drainage/Sewer Strings or 10.6.1.6 Setting Up the Map File Selection Criteria .

10.6.1.7 More on Creating the ADAC Map Files

In the examples, many of the ADAC Asset values were set by the ***Map File***, but others were not and they need to be set in the steps in the *Design* and *Survey* chains that occur after the ***Map File*** has been applied.

Good procedures and some **data preparation before** running the *Design* or *Survey* chains, will mean that most of the values will be set by the chains and any manual filling in of the missing values by using the **ADAC Editor** will be avoided.

How values such as *Diameter_mm* are set will vary from Asset to Asset, but they can also vary within the ‘same Asset.

For example, for a surveyor picking up *MaintenanceHoles*, they can come prefabricated in standard sizes with known Chamber type and sizes. In fact almost all of the elements for the *MaintenanceHole* would be known - FloorConstruction, FloorMaterial, WallConstruction, WallMaterial, RoofMaterial, LidMaterial. Similarly for Pipes.

So if the surveyor had a special code (string name) for each of the standard *MaintenanceHoles* or *Pipes*, then the ADAC Asset attributes for them can be set up with all the known standard values. Then using the special string name in the ***Map File*** means that all the values are already set correctly just by applying the ***Map File***.

Even if things are not totally standard, some attributes may have a certain value **most of the time**. If that is the case, the "most likely" value is the one that should be used in ADAC Asset in the ***Map File*** and then only the few that **vary** from the "most likely" value need to be modified at a later stage.

There may also be elements in an ADAC Asset that **the Authority is not interested in** so whatever value they are given by the ***Map File*** won’t have to be changed at any time.

Continue to the next section 10.6.2 What Data Prep is Needed for ADAC or return to 10.6.1 Setting Up Map Files for ADAC .

10.6.2 What Data Prep is Needed for ADAC

The Data Preparation that is needed on strings before they become ADAC Assets is different for each company. The options that are supplied are on the menu

File I/O =>ADAC =>User =>Data prep. See 10.4.9.2 Data Prep .

Continue to the next section 10.6.3 Setting up and Running ADAC Chains from the Menus or return to 10.6 Setting Up for ADAC .

10.6.3 Setting up and Running ADAC Chains from the Menus

The easiest way to run the ADAC *Survey* or *Design* chains with specific pvf files is to run them from a menu. This can be done in two ways:

(a) Running Specific Map and 12duaf files from Special areas

12d Model provides options on the walk right menu **Run 12d ADAC 4.1 Chains** to run ADAC *Survey* or *Design* chains for ADAC 4.1 with specially named Map and 12duaf files in either the working folder for the project (local), or User_Lib or Library.

See 10.6.3.1 Using the 12d Supplied Menu to Run ADAC Chains .

(b) Running Client Specific Map and 12duaf files

Once you are producing ADAC files for different Clients, you may need to produce different ADAC versions, or need different Map or 12duaf files.

For example you may need to use a different naming convention, or the information required in the ADAC assets is different.

For these situations, you can add your own options to **User Adac** menu.

The user options can run either the ADAC Base Survey or Design chains with specific chain pvf files.

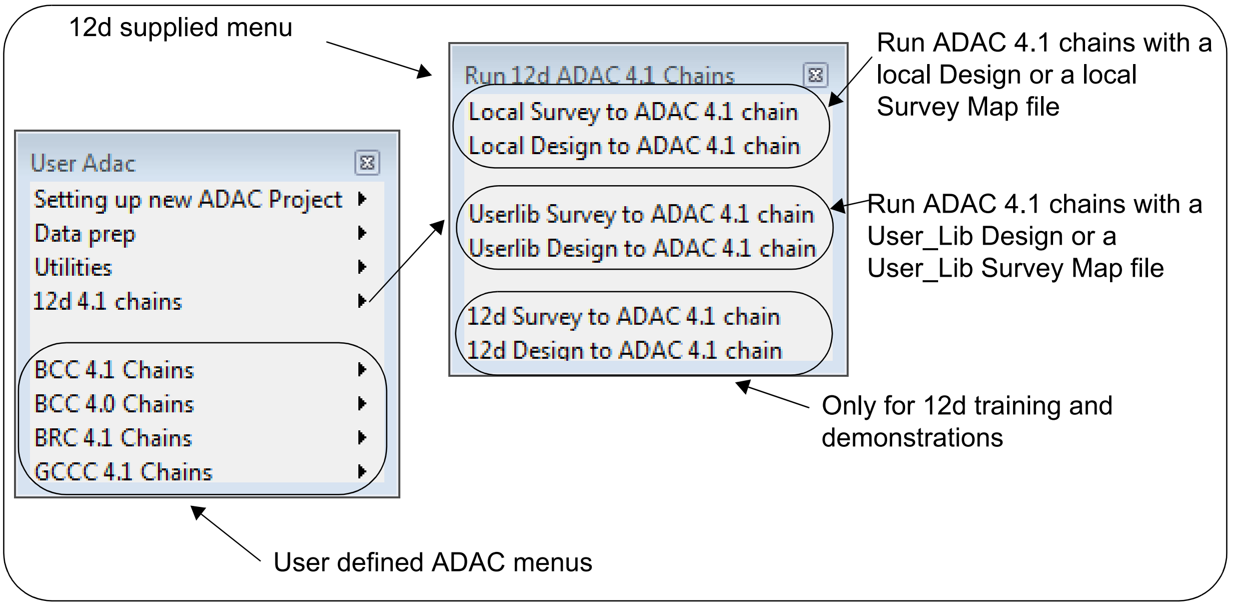
For information on setting up your own options on the **User ADAC** menu, see 10.6.3.2 Setting Up Your User ADAC Menu .

Or return to 10.6 Setting Up for ADAC .

10.6.3.1 Using the 12d Supplied Menu to Run ADAC Chains

12d Model provides options on the walk right menu **Run 12d ADAC 4.1 Chains** for the option **12d 4.1 chains** to run the base *ADAC Survey* or *Design* chains for ADAC 4.1 with specially named Map and *12duaf* files in either the **working folder** for the project (**local**) or **User_Lib**.

These options also use a **tin** called **ground** for surface levels.



(a) for the working folder for the project (local)

Local Survey to ADAC 41 chain runs *ADAC_Survey_Base* chain using the files from the **working folder for the project** (local):

ADAC_Local_Map_Survey_to_ADAC_41.mapfile

Local_41.12duaf *this file is optional*

Local Design to ADAC 41 chain runs *ADAC_Design_Base* chain using the files from the **working folder for the project (local)**:

ADAC_Local_Map_Design_to_ADAC_41.mapfile

Local_41.12duaf

Both options use the a **tin** called **ground** to get surface levels.

So to use the **Local** menu items you only need to create the two files and have them in the folder containing the project. This means that they are only available for that project.

(b) for User_Lib

Userlib Survey to ADAC 41 chain runs *ADAC_Survey_Base* chain using the files from the **USER_LIB** folder:

ADAC_Userlib_Map_Survey_to_ADAC_41.mapfile

Local_41.12duaf *this file is optional*

Userlib Design to ADAC 41 chain runs *ADAC_Design_Base* chain using the files from the **USER_LIB** folder:

ADAC_Userlib_Map_Design_to_ADAC_41.mapfile

Local_41.12duaf

Both options use a **tin** called **ground** to get surface levels.

So to use the **Userlib** menu items, you only need to create the two files and have them in the **User_Lib** folder. This means they can be used with any project.

Consequently you can use the **Userlib** options for any project and just set up **Local** ones for certain projects.

Continue to the next section 10.6.3.2 Setting Up Your User ADAC Menu or return to 10.6 Setting Up for ADAC .

10.6.3.2 Setting Up Your User ADAC Menu

To make it easy for your users to run the ADAC chains for either *Survey* or *Design*, or for different versions of ADAC, the easiest way is for the running of the options to be on a menu.

And since what is required could vary from Authority to Authority, and string naming conventions could differ between ***12d Model*** users, each user needs to set up there own ADAC menus.

The menu **ADAC User** can be modified by users so that is the one we will use.

To allow for all the variations, parametrised base ADAC Design and Survey chains are supplied in the ***12d Model*** Set_Ups folder (which most users do not have permission to access) and the variations required for all the different customer cases will be handled by passing parameter values through to these parametrised chains.

Placing the options on the **User ADAC** menu is done by setting up some folders and files in **User** which most users have access to.

To help explain things and the ease of setting up ADAC to work with multiple customers who may vary in what ADAC Assets they require in their ADAC XML files, we will set up a system to satisfy the following scenario:

You have three customers with the abbreviated names **BCC**, **BRC** and **GCCC**.

BRC started with ADAC 4.0 but are now moving to ADAC 4.1 and you have jobs in both versions. On some jobs you are providing BRC with Design ADAC XML files and on other jobs you providing BRC with Survey ADAC XML files.

BCC are using ADAC 4.1 only and you are providing them with Design and Survey ADAC XML files.

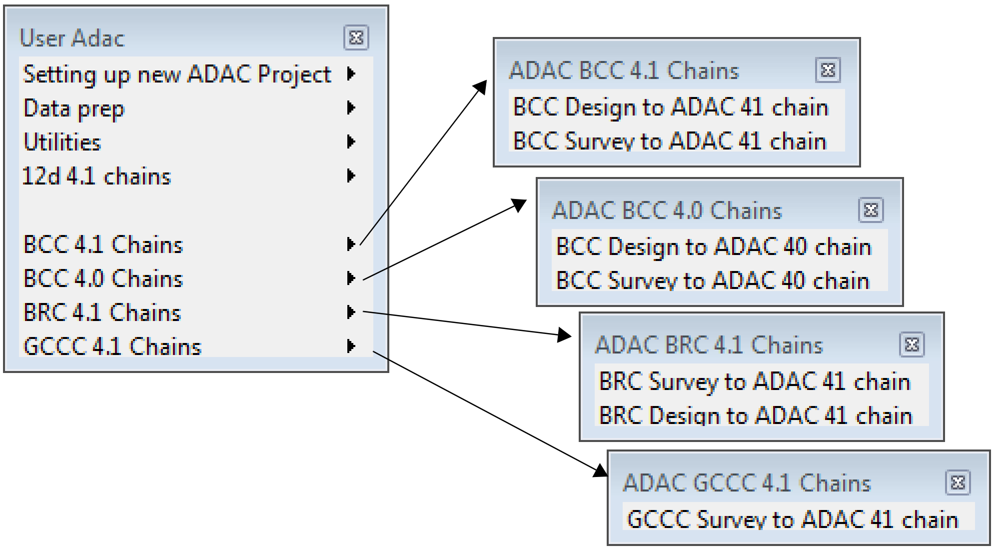
GCCC are using ADAC 4.1 only and at this stage and you are only providing them with Survey ADAC XML files.

Due to the nature of design and survey data, the 12d Map Files required for ADAC are always different. But for various reasons, the 12d Map files need to be different for each of the your customers as well.

Of course in real life your customers would never want something totally different but it is good for the scenario.

Note: If you are lucky enough that you can use the same configuration for all your customers (or you are an Authority that only has to supply itself) then you only need to set up the one subfolder and it is easiest to use your own abbreviated company name.

What we want to easily set up is an **User ADAC** menu with walk-rights:

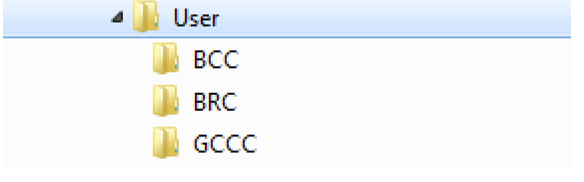


The steps to achieve this are:

(a) In the file ***usermenu.4d*** (which is usually in ***User***), add the following line at the top of the file.

#include "ADAC_customer_user_menu.4d"

(b) Create subfolders of **User** with the abbreviated name for each of your customers.



(c) Inside the subfolder (for example, inside **User\BCC**) there will be an appropriate **ADAC pvf** file

The ADAC **pvf** file is a ***parameter value file*** which is used in to pass parameters to both the *ADAC Design chain* and *ADAC Survey chain*.

In the ADAC **pvf** files themselves, there is five parameters that you have to set the values for:

(i) For the text parameter **company**, the Value which is the abbreviated company name. For example, **BCC**.

(ii) For the text parameter **adac_version_by_ten**, the Value **41** if you are generating ADAC 4.1.0 files, or **40** if you are generating ADAC 4.0 files.

(iii) user_attributes_conversion_type

(iv) For the text parameter **survey_finished_tin**, the name of a tin which is used to get some z-values for some tins in some macros.

(v) For the text parameter **design_finished_tin**. The name of a tin which may be used to get some z-values for some Assets in some macros

There are four other parameters in the **pvf** files, *survey_map_file*, *design_map_file*, *adac_attribute_file* and *adac_12d_attribute_file* but they are fully defined once company and adac_version_by_ten have values.

(vi) survey_map_file is

$USER_LIB\ADAC_[company]_Map_Survey_to_ADAC_[adac_version_by_ten].mapfile

(vii) design_map_file is
$USER_LIB\ADAC_[company]_Map_Design_to_ADAC_[adac_version_by_ten].mapfile

(viii) adac_attribute_file is
$USER_LIB\[company]_[adac_version_by_ten].12duaf

(ix) adac_attribute_file is
$LIB\adac_12d_[adac_version_by_ten].12duaf

To easily tell the different **pvf** files apart when you have them for different companies and ADAC versions, the **abbreviated company name** and the **adac_version_by_ten** are used in the name of the **pvf** file.

So for **company** with value **BCC** and adac_version_by_ten with value **41**, the **pvf** file is called

ADAC_BCC_41.pvf

And this **pvf** file must have the following contents:

```
<Chain_Parameters>
 <Text_Parameter>
  <Name>"company"</Name>
  <Comment>""</Comment>
  <Value>"BCC"</Value>
 </Text_Parameter>
 <Text_Parameter>
  <Name>"design_map_file"</Name>
  <Comment>""</Comment>

<Value>"$USER_LIB\ADAC_[company]_Map_Design_to_ADAC_[adac_version_by_ten].mapfile"<
/Value>
 </Text_Parameter>
 <Text_Parameter>
  <Name>"survey_map_file"</Name>
  <Comment>""</Comment>
<Value>"$USER_LIB\ADAC_[company]_Map_Survey_to_ADAC_[adac_version_by_ten].mapfile"<
/Value>
 </Text_Parameter>
 <Text_Parameter>
  <Name>"user_attributes_conversion_type"</Name>
  <Comment>""</Comment>
  <Value>"BCC"</Value>
 </Text_Parameter>
 <Text_Parameter>
  <Name>"design_finished_tin"</Name>
  <Comment>"ground"</Comment>
  <Value>""</Value>
 </Text_Parameter>
 <Text_Parameter>
  <Name>"survey_finished_tin"</Name>
  <Comment>""</Comment>
  <Value>"ground"</Value>
 </Text_Parameter>
 <Text_Parameter>
  <Name>"adac_version_by_ten"</Name>
  <Comment>""</Comment>
  <Value>"41"</Value>
 </Text_Parameter>
 <Text_Parameter>
  <Name>"adac_attribute_file"</Name>
  <Comment>""</Comment>
  <Value>"$USER_LIB\[company]_[adac_version_by_ten].12duaf"</Value>
 </Text_Parameter>
```

```
  <Text_Parameter>
    <Name>"adac_12d_attribute_file"</Name>
    <Comment>""</Comment>
    <Value>"$LIB\adac_12d_[adac_version_by_ten].12duaf"</Value>
  </Text_Parameter>
</Chain_Parameters>
```

The different **pvf** files can be created by copying the ***ADAC_BCC_41.pvf*** file and changing the **BCC** and **41** to what is needed. This can be done using a text editor, or the ***12d Model*** options

Utilities =>Chains => Parameters => Create or *Copy* or *Edit*.

So in the folder **User\BCC**, you need two pvf files - ***ADAC_BCC_41.pvf*** *and* ***ADAC_BCC_40.pvf***

In the folder **User\BRC**, you need the one pvf file - ***ADAC_BRC_41.pvf***

And in the folder **User\GCCC**, you need the one pvf file - ***ADAC_GCCC_41.pvf***.

(d) Inside each customer subfolder (for example, inside **User\BCC**) there must be a file called ***ADAC_customer_user_menu.4d*** which creates the walk right menus for that customer that go on your **User** menu.

In our example, **BCC** requires both **ADAC 4.1** and **ADAC 4.0** so two **walk** right menus are needed.

So in the folder **User\BCC**, the contents for

ADAC_customer_user_menu.4d

are:

```
Button "BCC 4.1 Chains" {
  Walk_Right "ADAC BCC 4.1 Chains"
}
Button "BCC 4.0 Chains" {
  Walk_Right "ADAC BCC 4.0 Chains"
}
```

the text on the menu will be **BCC 4.1 Chains**

and it has a walk right menu and the definition for the walk right menu is in a menu called **ADAC BCC 4.1 Chains**

BRC only required ADAC 4.1 so only one walk right menu is required. and ***ADAC_customer_user_menu.4d*** in the folder **User\BRC** is

```
Button "BRC 4.1 Chains" {
  Walk_Right "ADAC BRC 4.1 Chains"
}
```

the text on the menu will be **BRC 4.1 Chains**

and it has a walk right menu and the definition for the walk right menu is in a Menu called **ADAC BRC 4.1 Chains**

In the folder **User\GCCC**, the ***ADAC_customer_user_menu.4d*** file is almost identical to that for BRC that BRC is replaced by GCCC.

So now we need to define walk-right menus mentioned in the ***ADAC_customer_user_menu.4d*** files.

(e) Inside each customer folder, the walk right menus are defined in a file called ***ADAC_customer_walk_right_menu.4d***

The walk right menus contain the options that run the required ADAC Design and ADAC Survey chains.

In our example, **BCC** has tow walk right menus and they both need options to run Design and Survey chains.

So in the folder **User\BCC**, the contents of ***ADAC_customer_walk_right_menu.4d*** are

```
Menu "ADAC BCC 4.1 Chains" {                                    ADAC design chain
  Button "BCC Design to ADAC 41 chain" {
    Command "chain  -pvf $USER/BCC/ADAC_BCC_41.pvf  $LIB/ADAC_design_base.chain"
  }
  Button "BCC Survey to ADAC 41 chain" {
    Command "chain  -pvf $USER/BCC/ADAC_BCC_41.pvf  $LIB/ADAC_survey_base.chain"
  }
}                  pvf file for BCC and ADAC 4.1              ADAC survey chain

Menu "ADAC BCC 4.0 Chains" {
  Button "BCC Design to ADAC 40 chain" {
    Command "chain   -pvf $USER/BCC/ADAC_BCC_40.pvf   $LIB/ADAC_design_base.chain"
  }
  Button "BCC Survey to ADAC 40 chain" {
    Command "chain   -pvf $USER/BCC/ADAC_BCC_40.pvf   $LIB/ADAC_survey_base.chain"
  }
}
```

BRC only require ADAC 4.1 but do need both Design and Survey so only one walk right menu is required but it needs an option for Design and an option for Survey on it. So the ***ADAC_customer_walk_right_menu.4d*** in the folder **User\BRC** is

```
Menu "ADAC BRC 4.1 Chains" {                                    ADAC design chain
  Button "BRC Design to ADAC 41 chain" {
    Command "chain  -pvf $USER/BRC/ADAC_BRC_41.pvf  $LIB/ADAC_design_base.chain"
  }
  Button "BRC Survey to ADAC 41 chain" {
    Command "chain  -pvf $USER/BRC/ADAC_BRC_41.pvf  $LIB/ADAC_survey_base.chain"
  }
}                  pvf file for BRC and ADAC 4.1              ADAC survey chain
```

GCCC only require ADAC 4.1 and only need Survey so only one walk right menu is required with just the one option for Survey on it. So the ***ADAC_customer_walk_right_menu.4d*** in the folder **User\GCCC** is

```
Menu "ADAC GCCC 4.1 Chains" {
    Button "GCCC Survey to ADAC 41 chain" {
     Command "chain -pvf $USER/GCCC/ADAC_BRC_41.pvf $LIB/ADAC_survey_base.chain"
  }
}
```

pvf file for GCCC and ADAC 4.1

ADAC survey chain

(f) Finally the ***ADAC_customer_user_menu.4d***'s for each of the customers needs to be included in the **ADAC User** menu

This is done by having a file in the **User** folder called ***ADAC_customer_user_menu.4d***

(Aside: Yes this is the same name as in the customer folders. If it causes confusion then we will change it. Maybe it should have an s in it ***ADAC_customers_user_menu.4d***)

This is the file that was included in the **User** menu by the
    #include "ADAC_customer_usr_menu.4d "statement.

This file first includes all the walk right menus for the different customers on the **User ADAC** menu, and then includes the definitions of the walk right menus.

\

```
Menu "User Adac" {

#include "BCC\ADAC_customer_user_menu.4d"
#include "BRC\ADAC_customer_user_menu.4d"
#include "GCCC\ADAC_customer_user_menu.4d"

}

#include "BCC\ADAC_customer_walkright_menu.4d"
#include "BRC\ADAC_customer_walkright_menu.4d"
#include "GCCC\ADAC_customer_walkright_menu.4d"
```

places each walk right menu on the User ADAC menu

defines the walk right menus

Continue to the next section 10.6.4 Setting Up ADAC Templates or return to 10.6.3 Setting up and Running ADAC Chains from the Menus or 10.6 Setting Up for ADAC .

10.6.4 Setting Up ADAC Templates

When creating *ADAC Header* using the **ADAC =>Create Header**, or *ADAC Assets* using the **ADAC =>Create Asset** option, Templates can be used to automatically fill in many of the values of ADAC elements.

See

10.6.4.1 ADAC Header Templates

10.6.4.2 ADAC Asset Templates

10.6.4.1 ADAC Header Templates

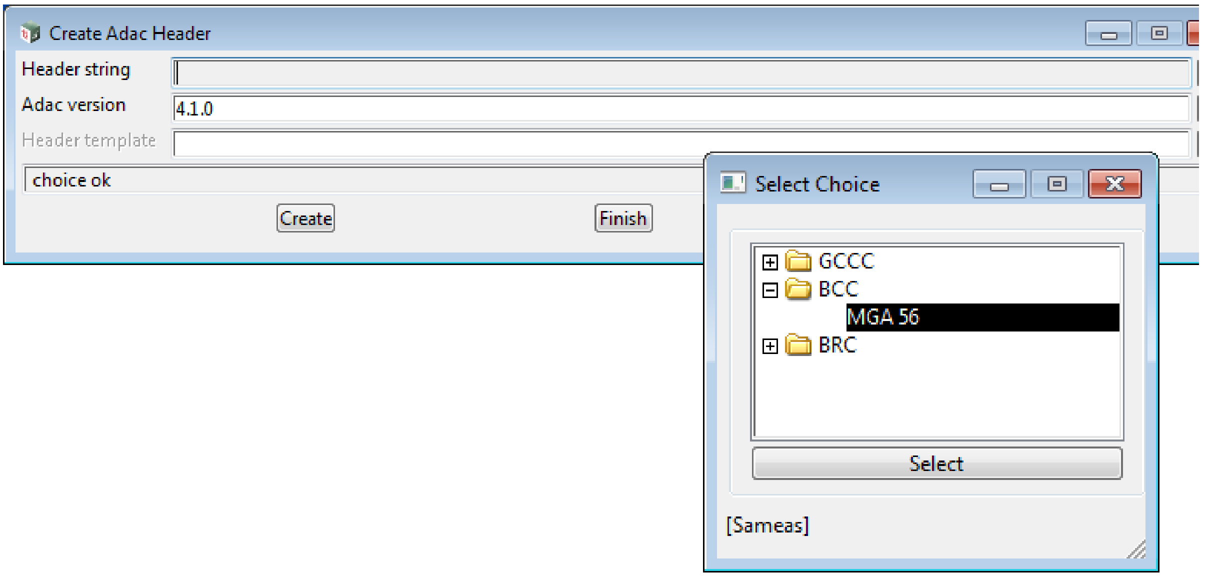
In most firms, many of the values in an *ADAC Header* would be the same for each new ADAC job.

For example, all your jobs could be in *MGA zone 56*, using the *Horizontal Datum GDA 94* and *Vertical Datum AHD*. Or you always have the three surveyor names in the *ADAC Header.*

To prevent having to reenter this data every time you create a new ADAC Header, users can create *ADAC Header* **Templates** with information for any new ADAC Header already filled in.

When you have *ADAC Header Templates*, the **Template** field pop up in the **Create ADAC Header** panel lists all your *ADAC Header* **Templates** for the selected **ADAC version**, as a folder structure.

When the new *ADAC Header* is created, it will have **all** the values from the selected *Header Template*.

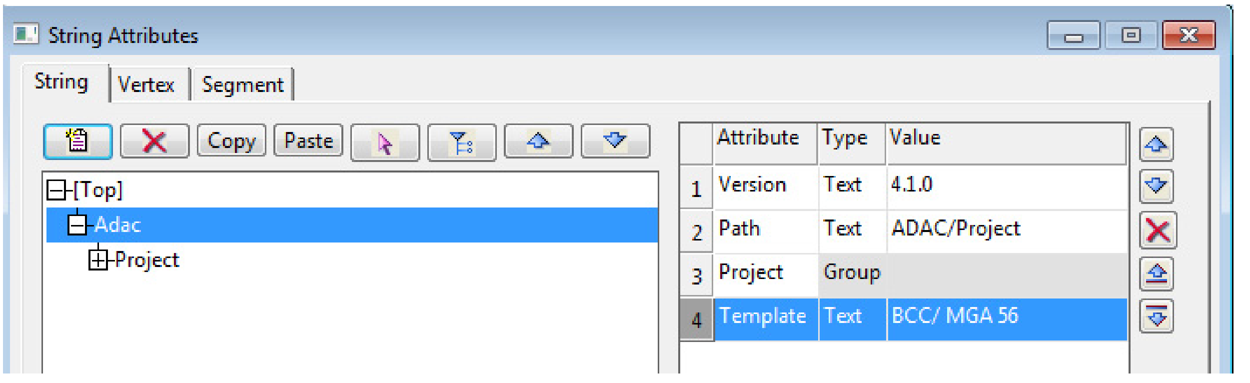


So for the new ***ADAC Header***, only those values not covered by the ***Header Template*** need to be entered.

To create an ***ADAC Header Template***, you

1. create an ***ADAC Header*** for the required *ADAC version*, and with the *ADAC values* that you want the *Header Template* to have
2. place it in the model ***ADAC Header Templates***
3. in the string attributes, add a **Text** attribute called **Template** into the **top** level of the ***ADAC attribute group***, with its value being the required folder structure for displaying the ***Header Template*** with the names for each folder level being separated by a forward slash (**/**).

For example, to have a ***Header Template*** called **MGA 96** and have it display in a folder called **BCC**, the value for the attribute **Template** is ***BCC/MGA 96***



To use the ***Header Templates*** in another project you can either:

(a) write the model ***ADAC Header Templates*** out as a 12da file and read it into new ADAC projects.

(b) have the model ***ADAC Header Templates*** in a project and share that project into your new ADAC projects.

(c) write the models ***ADAC Header Templates*** and ***ADAC Asset Templates*** out to the 12da file ***ADAC_Templates.4da*** and place the file in User_Lib.

The option:

ADAC =>User =>Setting up a new ADAC Project =>Read in templates from User_Lib

can then be used in any project to read *User_Lib\ADAC_Templates.4da* in, and hence the models ***ADAC Header Templates*** and ***ADAC Asset Templates*** into the new project.

See 10.4.9.1.3 Reading in Templates from User_Lib .

Continue to the next section 10.6.4.2 ADAC Asset Templates or return to 10.6.4 Setting Up ADAC Templates .

10.6.4.2 ADAC Asset Templates

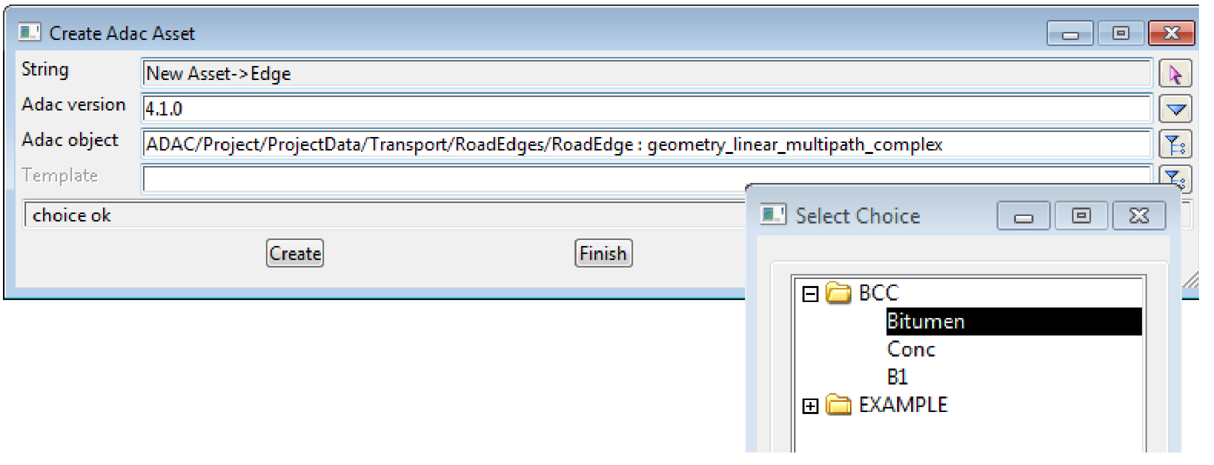
When creating *ADAC Assets* by hand, there would be some the values in an *ADAC Asset* that would be the same each time the *ADAC Asset* was use in ADAC job.

For example, a *Transport>RoadEdges>RoadEdge* with **Type *Bitumen*** and **Owner** *Council* may be regularly required.

To prevent having to reenter this data every time you create a new ADAC Asset, users can create *ADAC Asset* **Templates** with information for any new ADAC Asset already filled in.

When you have *ADAC Asset Templates*, the **Template** field pop up in the **Create ADAC Asset** panel, lists all your *ADAC Asset* **Templates** for the selected **ADAC version** and the selected *ADAC Asset,* in a folder structure.

When the new *ADAC Asset* is created, it will have **all** the values from the selected *Asset Template*.

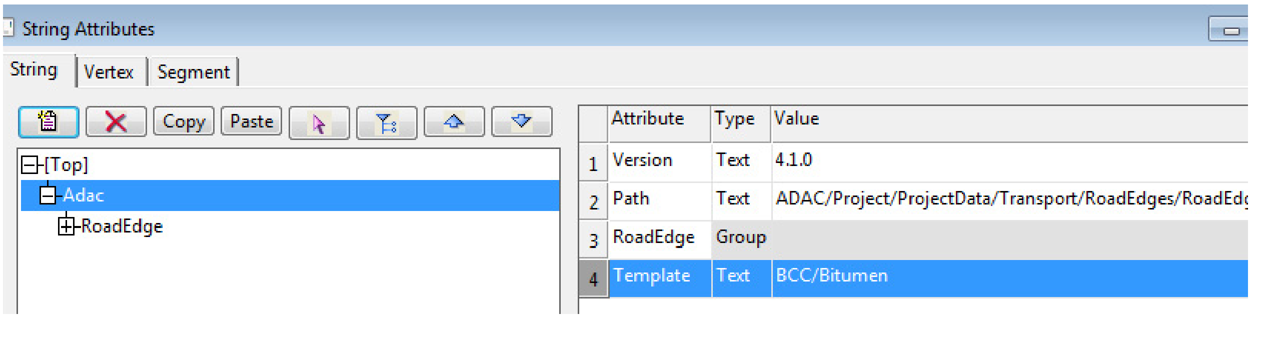


So for the new ***ADAC Asset***, only those values not covered by the ***Asset Template*** need to be entered.

To create an ***ADAC Asset Template***, you

1. create an ***ADAC Asset*** for the required *ADAC version*, and with the *ADAC values* that you want the *Asset Template* to have
2. place it in the model ***ADAC Asset Templates***
3. in the string attributes, add a **Text** attribute called **Template** into the **top** level of the ***ADAC attribute group***, with its value being the required folder structure for displaying the ***Asset Template*** with the names for each folder level being separated by a forward slash (**/**).

   For example, to have an ***Asset Template for a Road Edge*** called **Bitumen,** and have it display in a folder called **BCC**, the value for the attribute **Template** is ***BCC/Bitumen***



To use the ***Asset Templates*** in another project you can either:

(a) write the model ***ADAC Asset Templates*** out as a 12da file and read it into new ADAC projects.

(b) have the model ***ADAC Asset Templates*** in a project and share that project into your new ADAC projects.

(c) write the models ***ADAC Header Templates*** and ***ADAC Asset Templates*** out to the 12da file ***ADAC_Templates.4da*** and place the file in *User_Lib*.

The option:

ADAC =>User =>Setting up a new ADAC Project =>Read in templates from User_Lib

can then be used in any project to read *User_lib\ADAC_Templates.4da* in, and hence the models ***ADAC Header Templates*** and ***ADAC Asset Templates*** into the new project.

See 10.4.9.1.3 Reading in Templates from User_Lib .

Continue to the next section 10.6.5 Setting Up Your User Keys or return to 10.6 Setting Up for ADAC .

10.6.5 Setting Up Your User Keys

Two thing you will do regularly when doing ADAC work is to bring up the ADAC menu and to bring up the **String Attributes** panel.

So in the standard ***12d Model 11*** installation, the function keys are assigned so that:

Pressing **F12** brings up the **ADAC** menu.
Pressing **F8** brings up the **Strings Attributes** panel.

If you wish to change the use of functions keys then you can define the way you want the *function keys* to work.

For example, you may want

Press **F11** to bring up the **ADAC** menu.
Press **F12** to bring up the **String Attributes** panel.

In ***12d Model*** the actions of the function keys is controlled by ***userkeys.4d***. There may already be a copy in your **User** folder. If not, copy it over from the folder in ***Program Files***:

12d\12dmodel\11.00\set_ups.

Then just replace the existing lines for defining **f11** and **f12** with (lines preceded by **//** are comment lines).

// mod for f11

f11 panel "ADAC"

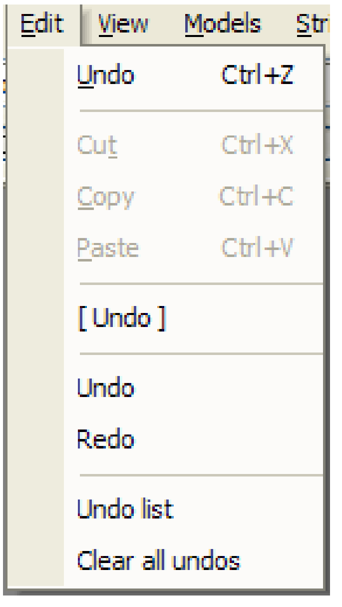
// mod for f12

f12 panel "String Attributes"

Return to 10.6 Setting Up for ADAC .

11 Edit

The **Edit** menu contains the **Undo** and **Redo** options.



Infinite **Undo** and **Redo** facilities have been implemented for many of the 12d Model options.

However, because much of the power of 12d Model comes from being able to leave options hanging and editing more than one string at a time, this made the standard concept of *Undo/ Redo* of limited benefit to 12d Model users.

Consequently, the following methodology has been adopted for **Undo** and **Redo** in 12d Model:

Independent undo and redo lists are maintained for each string being edited and these list are cleared once the editor is left.

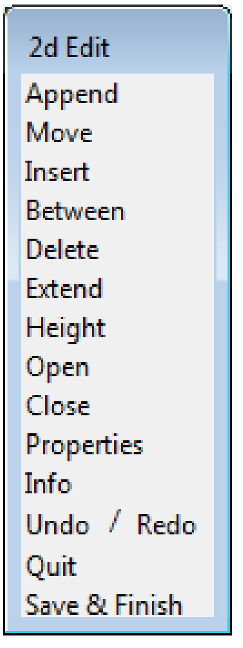
There is also a main undo/redo list which is used for all other options that can be undone. The main undo/redo lists are cleared when the 12d Model session is exited.

Please continue to the next section Undo and Redo for Editors.

11.1 Undo and Redo for Editors

Each editor contains its own **Undo/Redo** option on its **nd Edit** menu and whilst a string is being edited, the editor maintains its own undo/redo lists.

For example, for a 2d string, the **Undo/Redo** option is just above the **Quit** option.



As each edit is made to the string, the state of the string before the edit is added to the top of the string's **undo** list and the **redo** list is emptied.

If **undo** is selected from the **nd edit** menu, the last edit operation for that string will be **undone** and the undone operation added to the top of the string's **redo** list.

If another **undo** is selected before another edit is made to the string, what is now was last operation on the string will be **undone** (is was the second last operation before the previous undo) and the undone operation added to the top of the string's **redo** list.

If **redo** is selected from the **nd edit** menu, then the top operation on the **redo** list is redone, and the state of the string before the redo is added to the top of the undo list.

Hence edits for he string can be **undone** and **redone** whilst in the edit session for the string.

When the edit session is completed by selecting either **quite** or **finish** from the **nd edit** menu, the undo and redo lists for the edit session are deleted and the operations for the edit session can no longer be undone or redone.

Note

If more than one string is being edited at the same time, each string editor maintains its own undo/redo lists so that the undo/redo operations for the different strings do not get intertwined.

Hence if **undo** or **redo** is selected from a particular strings **nd edit** menu, the undo or redo applies **only** to the operations performed on that particular string, no matter what other **nd edit** options where performed on other strings.

Please continue to the next section Undo and Redo for other Options.

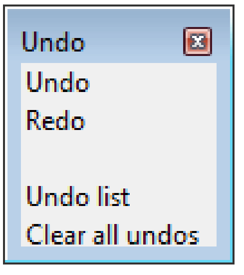
11.2 Undo and Redo for other Options

As well as the undo/redo lists maintained by the string editors, there is another set of undo/redo list maintained for all other options that support undo and redo. These are called the **main undo** and **redo** lists.

For a description of the undo in ***12d Model***, please go to the section Edit.

The **undo** option which controls the main undo/redo lists, is the on the Edit menu from the main menu.

If **undo** is selected from the **undo** menu, the last undoable option performed (apart from editor operation on strings) will be **undone** and the undone operation added to the top of the main **redo** list.



If another **undo** is selected before another undoable option is performed, what is now was last operation will be **undone** (is was the second last operation before the previous undo) and the undone operation added to the top of the main **redo** list.

If **redo** is selected from the **undo** menu, then the top operation on the main **redo** list is redone, and the state before the redo is added to the top of the undo list.

Hence many operations can be **undone** and **redone**.

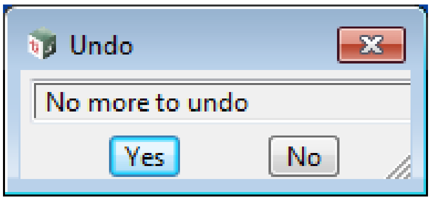
When the 12d Model session is completed by selecting either **exit** from the **12d Model** menu, the undo and redo lists for the session are deleted and the operations for the session can no longer be undone or redone.

For the option *Undo*, go to the section	Undo.
Redo	Redo
Undo list	Undo List
Clear all undos	Clear all Undos

For some restrictions on the Undo/Redos, please continue to the section Some Restrictions on Undo and Redo.

11.2.1 Undo

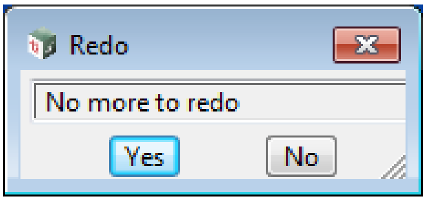
If **undo** is selected from the **undo** menu, the **undo** panel showing the last operation to *undo* is displayed.



For some restrictions on the Undo/Redos, please continue to the section Some Restrictions on Undo and Redo.

11.2.2 Redo

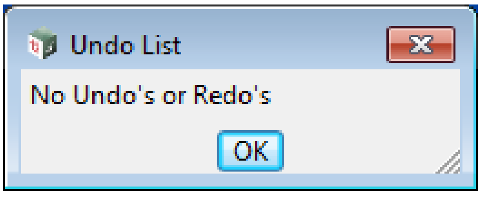
If **redo** is selected from the **undo** menu, the **redo** panel showing the last operation to *redo* is displayed.



For some restrictions on the Undo/Redos, please continue to the section Some Restrictions on Undo and Redo.

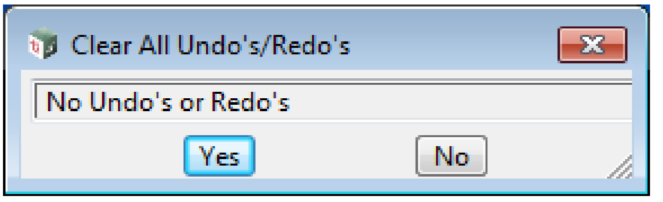
11.2.3 Undo List

If **undo list** is selected from the **undo** menu, a list of all the items on the undo list is displayed in the **undo list** panel.



11.2.4 Clear all Undos

Selecting **clear all undo** from the **undo** menu will bring up a clear all undo's/redo's **yes-no** panel.



If **yes** is selected, the main undo and redo lists will be cleared.

11.2.5 Some Restrictions on Undo and Redo

Undo and Redo are available for most of the **Strings** options and most of the **Utilities** options.

Undo is available for all **File=> Input** options but there are no Redos for these Undos.

Undo is available for all **Utilities=>Global** options that use the **Output** option with the Mode set to one of the three copy modes. There are no Redos with these Undos.

There are no Undos for **Models=>Delete** and **Models=>Clean**.

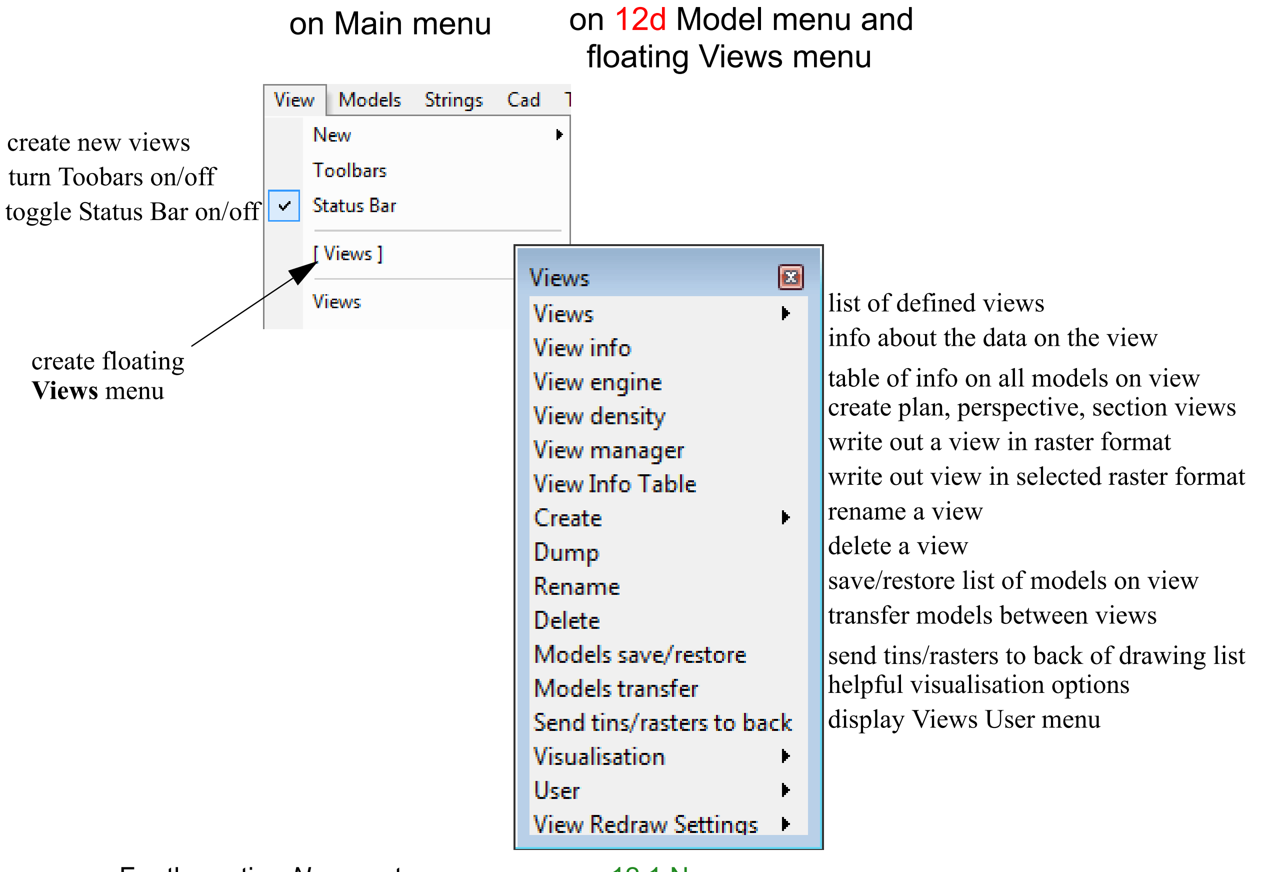
12 View

Position of menu: **View**

Views are the drawing display areas in 12d Model.

The **View** walk-right menu contains options to list existing views and create new views. The view drawing and manipulation options are in the **View** menus attached to each view.

The **Views** walk-right menu is

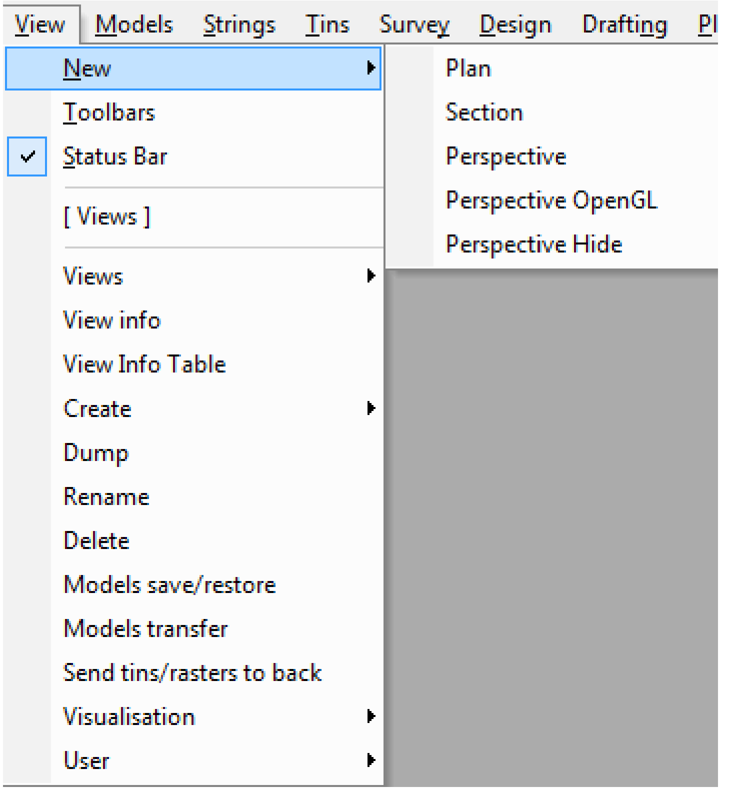


For the option *New*, go to 12.1 New
Toolbars 12.2 Toolbars
Status bar Status Bar
Views 12.3 Views
View info 12.4 View Info
View info table 12.5 View Information Table
Create 12.6 Create
Dump 12.7 Dump
Rename 12.8 Rename
Delete 12.9 Delete
Models save/restore 12.10 Models Save/Restore
Models transfer 12.11 Models Transfer
Send tins/rasters to back 12.12 Send Tins/Rasters to Back
Visualisation 12.13 Visualisation
View Redraw Settings ALG??

12.1 New

Position of option on menu: **View =>New**

The **New** walk-right menu provides options to create views of type *Plan*, *Section*, *Perspective*, *Perspective OpenGL* and *Perspective Hide.*

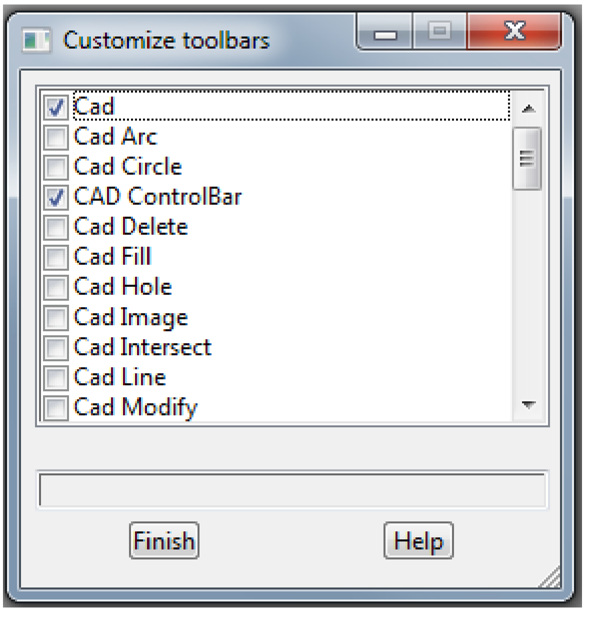


After selecting one of the options, a new view of the appropriate type with the next sequential view number is created in the Views Area.

12.2 Toolbars

Position of option on menu: **View =>Toolbars**

Clicking on **Toolbars** brings up the **Customize Toolbars** panel



Clicking the tick on/off for a given toolbar or controlbar in the *Toolbars* list turns the toolbar or controlbar on/off.

Status Bar

Position of option on menu: **View =>Status Bar**

Toggles the *Status Bar* on/off.

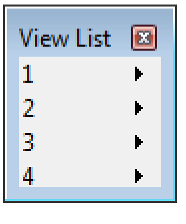
For more information please see 4.3.8 Status Bar.

12.3 Views

Position of option on menu: **View =>Views**

The **Views** walk-right menu provides a list of the all the views defined in this session of 12d Model.

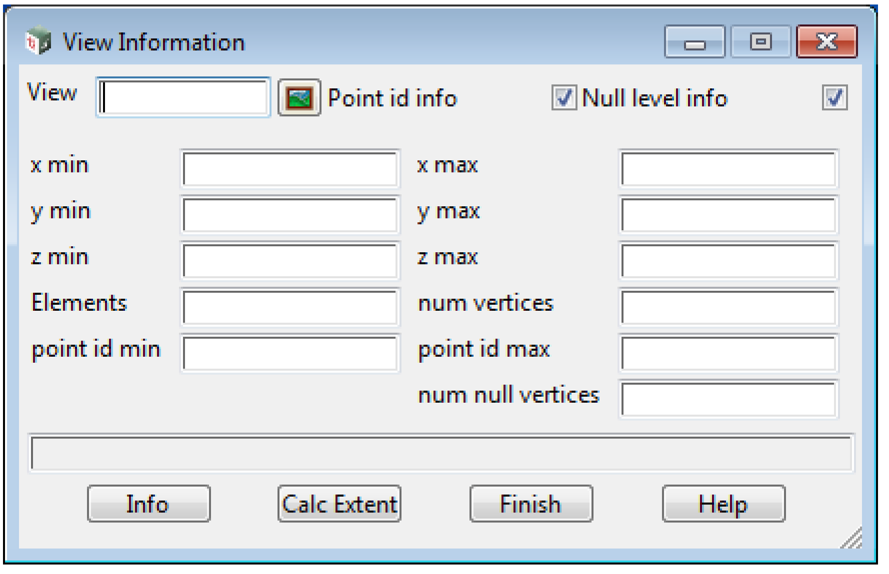
For each view in the list, a further walk-right menu lists all the models added to that view.



12.4 View Info

Position of option on menu: **View =>View Info**

Selecting **View info** fires up the **View information** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	input		available views

input the name of the view to get information about

Point id info	tick box	tick	

if ticked, the Point id min *and* Point id max *fields are shown on the panel and when the* **Info** *button is pressed, their values calculated for the data on the view.*
If not ticked, the Point id min *and* Point id max *fields are not shown on the panel.*

Null level info	tick box	tick	

if ticked, the Num null vertices *field is shown on the panel and when the* **Info** *button is pressed, the value calculated for the data on the view.*
If not ticked, the he Num null vertices *field is not shown on the panel.*

xmin/ymin/zmin, xmax/ymax/zmax output

returns the limits of all the models on the view

Point id min/max output

minimum/maximum integer point id in the model

Elements output

returns the number of elements in all the models on the view

Num vertices output

returns the number of points in all the models on the view

Point id min output

the minimum Point id for the data on the view.

Point id max output

the maximum Point id for the data on the view.

Num null vertices output

returns the number of null vertices in all the models on the view

Info button

get the information for the view given in the view field.

Calc Extent button

recalculate the x, y, z bounding box for the models on the view given in the view field.

How to Use the Panel

The view information for the view given in the view field is retrieved and placed in the appropriate panel fields when the view name is entered into the view field from the pop-ups, or a <enter> is entered after entering the view name into the view field, or on selecting the **info** button.

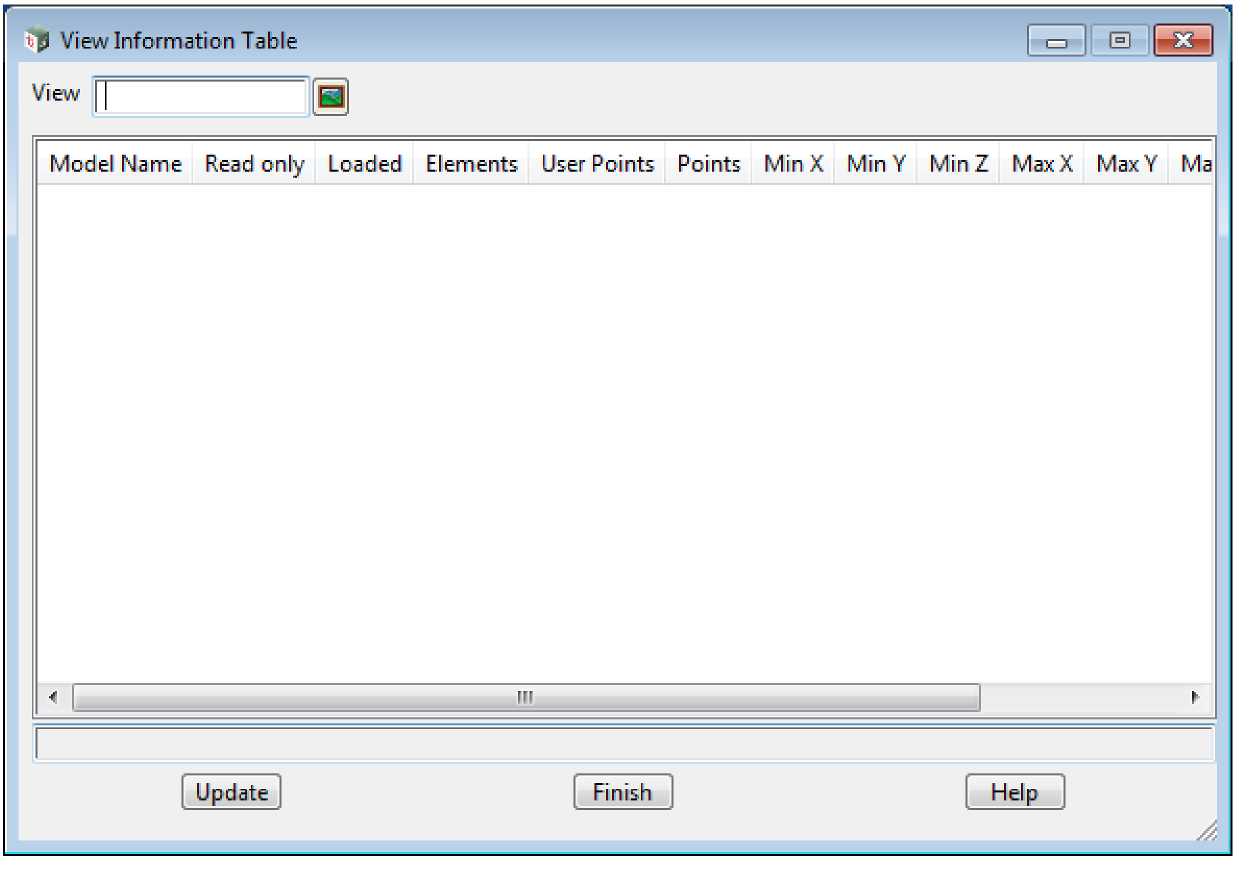
12.5 View Information Table

Position of option on menu: **View =>View info table**

The **View information table** option displays the minimum and maximum x, y and z values for every model on the view, and by double-clicking on a model name in the list, brings up a String Information Table that lists all the strings in that model (see 13.3 Model Information Table).

The models and minimum and maximum columns can be sorted into ascending or descending order by using bringing up the **sort** menu on the column header fields.

Selecting **View info table** fires up the **View Information Table** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box		available views
the name of the view to list all the models for			
Model	column		**sort** menu
all the models in the view are listed in the model *column.*			
Read only	column		**sort** menu
Loaded	column		**sort** menu
*if no, in this current opening of the project, the model has not yet been fully loaded into **12d Model**.*			
Min X, Min Y, Min Z **Max X, Max X, Max Z**	columns		**sort** menu
the minimum/maximum values for the model are displayed in the columns			

Created/Updated columns dates

dates the model was first created/updated

Update button

recalculate the minimum/maximum information in the table.

Notes

1. This is a scrolling panel. If there is too much information to fit into the table, then the scrolling arrow on the right hand side of the table must be used to display the extra information.
2. The grid can be sorted by any of the columns.

12.6 Create

Position of menu: **View =>Create**

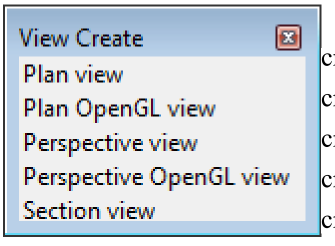
The floating **Create** menu item operates in two ways.

First, if **Create** itself is activated (by clicking LB when **Create** is highlighted), the **New View** panel appears. The **New View** panel can be used to create plan, perspective and sections views.

Secondly, the **Create** walk-right brings up a the **View Create** menu. This walk-right menu has separate options for creating plan, perspective and section views.

The Main menu option **View** => **Create** will only work the second way

The **Create** walk-right menu is



create a plan view
create a plan OpenGL view
create a perspective view
create a perspective OpenGL view
create a section view

The **New View** panel obtained by clicking LB on the **Create** menu item and the three options on the **View Create** menu will now be described.

For the option *New view*, go to 12.6.1 New View

Plan view 12.6.2 Plan, Perspective and Section Views

Plan OpenGL view ALG??

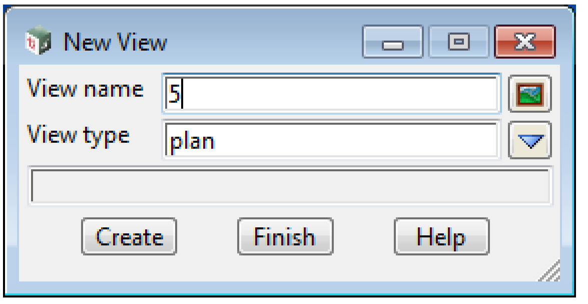
Perspective view 12.6.2 Plan, Perspective and Section Views

Perspective OpenGL view 12.6.2 Plan, Perspective and Section Views

Section view 12.6.2 Plan, Perspective and Section Views

12.6.1 New View

On selecting the **Create** option, the **New View** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View name	view box	next view number	

name of the new view to be created.

View type	choice box	plan	plan, persp, section, persp openGL

type of the view to be created. There are three types of views - plan, perspective, perspective openGL and sections views.

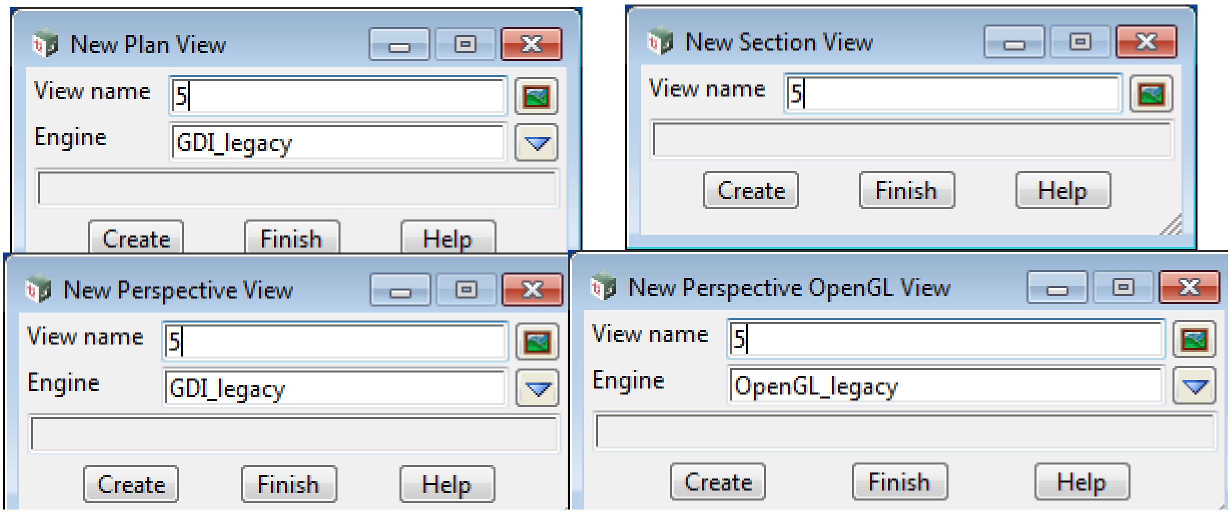
Create	button

after selecting this button, a new view of the name and type given in the panel is created in the Views Area. The panel then disappears.

12.6.2 Plan, Perspective and Section Views

Position of option on menu: **View =>Create =>Plan/Perspective/Perspective OpenGL/Section view**

On the **Plan/ Perspective/Perspective OpenGL/Section view** option, the **New Plan/Perspective/ Perspective OpenGL/ Section view** panel is displayed.



The fields and buttons used in this panels have the following functions.

Field Description	Type	Defaults	Pop-Up
View name	input	next view number	

name of the new plan/perspective/section view to be created.

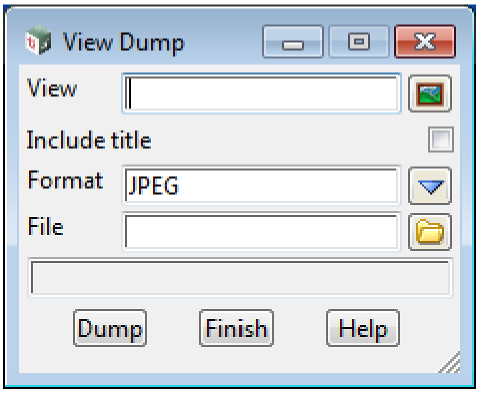
Create	button		

after selecting this button, the new view of the specified name is created in the Views Area. The panel then disappears.

12.7 Dump

Position of option on menu: **View =>Dump**

This option is used to write the view out in a user selected raster format. The option is identical to the **Dump** option on the **View System** menu.



The fields and buttons used in this panel have the following functions.

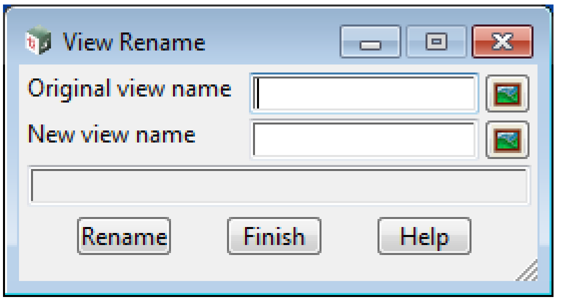
Field Description	Type	Defaults	Pop-Up
View	view box		available views
name of the view to be dumped			
Include title	tick box		
if ticked, include the view title area in the dump			
Format	choice box	jpeg	bmp, gif, jpeg, jpeg 2000, png, tga
format of the dump of the view			
File	file box		
name of the file for the dump of the view			
Dump	button		
dump the view in the given format			

12.8 Rename

Position of option on menu: **View =>Rename**

This option renames an existing view.

Selecting **Rename** brings up the **View Rename** panel:



The fields and buttons used in this panels have the following functions.

Field Description	Type	Defaults	Pop-Up
Original view name	view box		available views
current name of the view			
New view name	view box		available views
new name for the view			
Rename	button		
rename the view			

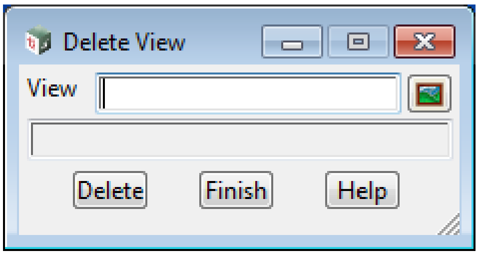
12.9 Delete

Position of option on menu: **View =>Delete**

This option deletes an existing view.

Since the easiest method to delete a view is to use the **[X]** in the top right hand corner of the view, the *Delete View* option is mainly used in rare cases such as when the view is not responding due a graphics card problem.

Selecting **Delete** brings up the **Delete View** panel:



The fields and buttons used in this panels have the following functions.

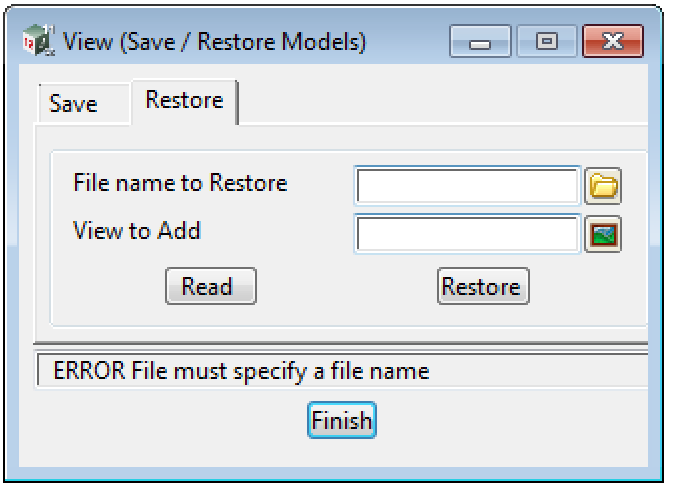
Field Description	Type	Defaults	Pop-Up
View	view box		available views
name of the view to delete			
Delete	button		
delete the view			

12.10 Models Save/Restore

Position of option on menu: **View =>Models save/restore**

This option is used to write out a list of models on the view and also to read in a list of models and add them to a given view.

On selecting the **Models save/restore** option, the **View save/restore models** panel is displayed.



The fields and buttons used in this panels have the following functions.

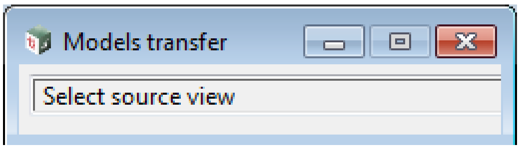
Field Description	Type	Defaults	Pop-Up
Save tab	tab		
File name for save	file		*.vml files
file for model list.			
View to save	view		available views
view to write out list of model for.			
Save	button		
after selecting this button, the names of all the model on the view are written out to the file.			
Restore tab	tab		
File name to restore	file		*.vml files
file of model list.			
View to add	view		available views
view to add the models in the model list to.			
Read	button		
read the list of models.			
Restore	button		
add the models in the list just read in to the given view.			

12.11 Models Transfer

Position of option on menu: **View =>Models transfer**

This option is used to add all the models on one view to a different view.

Selecting **Models transfer** brings up the **Models Transfer** panel:



The user is then asked to click in the view to get the models from (source view) and then to select the destination view where the models are to be added to.

Note - if the destination view is a new view, click on **Fit** first to give the new view a valid co-ordinate system.

12.12 Send Tins/Rasters to Back

Position of option on menu: **View =>Send tins/rasters to back**

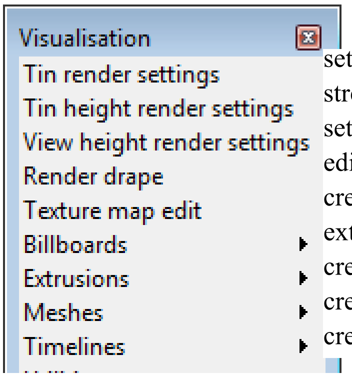
This option is used to send tins and rasters to the back of the drawing list.

12.13 Visualisation

Position of menu: **View =>Visualisation**

This is a chargeable module which uses the Perspective OpenGL view to create realistic 3d pictures using raster drapes, bitmaps and extrusions along super strings.

The **Visualisation** walk-right menu is



sets textures for tins
stretches an image of a tin using a height range
set rasters for draping on
edit texture map file
creating/edit/place billboards
extrusions
create/place meshes
create/run timelines
create clouds, fences, forests etc

For *Tin render settings*, go to 12.13.1 Tin Render Settings

Tin height render settings 12.13.2 Tin Height Render Settings

Render drape 12.13.3 Render Drape

Texture map edit 12.13.4 Texture Map Edit

Billboards 12.13.5 Billboards

Extrusions 12.13.6 Extrusions

Meshes 12.13.9 Meshes

Timelines 12.13.10 Timelines

Utilities 12.13.35 Utilities

12.13.1 Tin Render Settings

Position of option on menu: **View =>Visualisation =>Tin render settings**
Position of option on menu: **Tins =>Edit =>Render settings**

For a tin added to a Perspective OpenGL view with Shade ticked on, the triangles of the tin can be displayed as solid colour or various degrees of transparency depending of the Blending value for the tin which can vary between 1 and 0. If Blending is 1 the tin is opaque and triangles display as a solid colour, and Blending of 0 means the triangles are totally transparent.

A tin can be *one sided* and only viewable from above, or *two sided* and viewable from above or below.

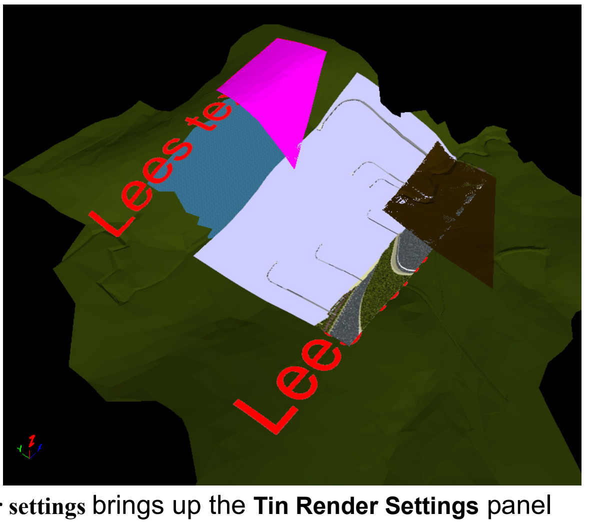
Rasters, such as photos, can also be associated with a tin and when the tin is displayed in a Perspective OpenGL view with the shade turned on, the rasters are draped onto the tin wherever the triangles are still the base colour.

For triangles that are not the base tin colour, textures can be applied to the triangle using a **Texture mapping** file where the triangle colour is the index into the mapping file.

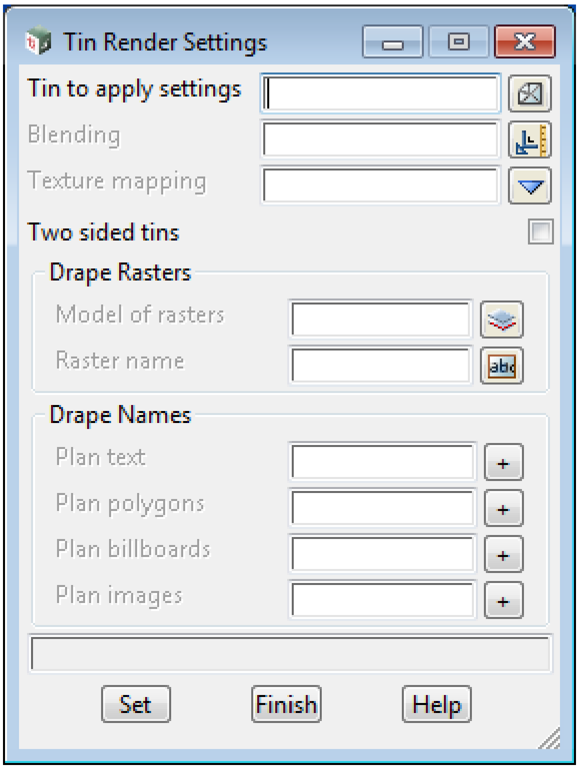
Finally text, filled polygons, text, billboards and images in models added to any Perspective OpenGL view that the tin is on can also be attributed so that they can be draped and displayed on the tin.

All these properties are known as ***Tin Render Settings*** and are set using this option.

IMPORTANT NOTE: the ***Visualisation*** module is required for the *Tin Render Settings* to be used on a Perspective OpenGL view.



Selecting **Tin render settings** brings up the **Tin Render Settings** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin to apply settings	tin box		available tins

name of the tin to apply/modify the Tin Render Settings. Any existing Tin Render Settings of the tin are loaded into the appropriate panel fields so that they can be modified.

Blending input

Blending control the level of transparency of the tin.
***0** means the tin is totally transparent (and hence invisible) and **1** means that the tin is opaque (non-transparent) and can't be seen through at all.*

If blank, then the Blending value defaults to 1 and the tin is opaque (non-transparent).

Texture mapping texture mapping file box

*if non blank, the name of the set of colour-to-texture mappings that is defined in the file **textures_maps.4d.** The Texture Mapping set defines for a colour, what texture from the **textures.4d** file is applied to the triangles of the tin that have been coloured this colour (and the colour in no the base tin), rather than applying any raster to those triangles.*

Two sided tins tick box

if ticked, the tin is visible when viewed from both above and below. For example, the tin used for the bottom of a bridge needs to be two-sided.
If non-tick, then the tin is one sided and only shows when viewed from above.

Drape Rasters

Model of rasters model box available models

if not blank, the model of ortho-rectified rasters to be associated with the tin and draped onto the tin when the tin is on a Perspective OpenGL view. The raster images will only be draped onto triangles that have the base tin colour.

Raster name input

if non blank, the name of an ortho-rectified raster to be associated with the tin (and hence draped onto the tin when the tin in on an Perspective OpenGL view).

Drape Names

WARNING WHEN USING DRAPE NAMES
Because draped text, billboards, images and textures can all occur at the same (x,y) value and hence have the same z-value at that point, there are certain ordering rules controlling what you see.

*First the **model** containing the **tin** needs to be brought to the **FRONT** of the view using **Perspective OpenGL View Menu: Models =>Models to front**.*

Then textures using the tin colour and the Texture mapping file will be drawn first, draped text will be next, then draped polygons and draped images fight it out.

Plan text

*if not blank, when the tin is on a Perspective OpenGL view, then for all super strings in models also added to that view, all **vertex text** that is using a **true type font** are draped onto the tin.*

Plan polygons

*when the tin is on a Perspective OpenGL view, then for all super strings in models also added to that view, if a super string in the model has an Integer string attribute with this name then it is draped onto the tin and if the value of the attribute is zero (0) the super string is draped and coloured using the strings fill colour, or if the value is one (1), a texture is draped onto the tin and the fill colour is used as the **reference** to the* **Texture mapping** *file given in the* Texture mapping *field, to specify which texture to drape.*

*The Integer string attribute name and value can be set using the option **View =>Visualisation =>Render drape** (see* 12.13.3 Render Drape*) or by setting an Integer string attribute of the required name and value using the option **Strings =>Properties =>Attributes** (see* 14.12.8 Attributes*).*

Plan billboards

when the tin is on a Perspective OpenGL view, then for all super strings in models also added to that view, if a super string in the model has an Integer string attribute with this name and any value, all horizontal billboards (i.e. have Vertical *turned off) are draped onto the tin.*

The Integer string attribute with this name can be set using the Drape name *field in the option **View =>Visualisation=>Billboards =>Add/Remove (ss) (**see* 12.13.5.2 Add and Remove Billboard from Super String*) or by setting an Integer string attribute of the required name using the option **Strings =>Properties =>Attributes** (see* 14.12.8 Attributes*).*

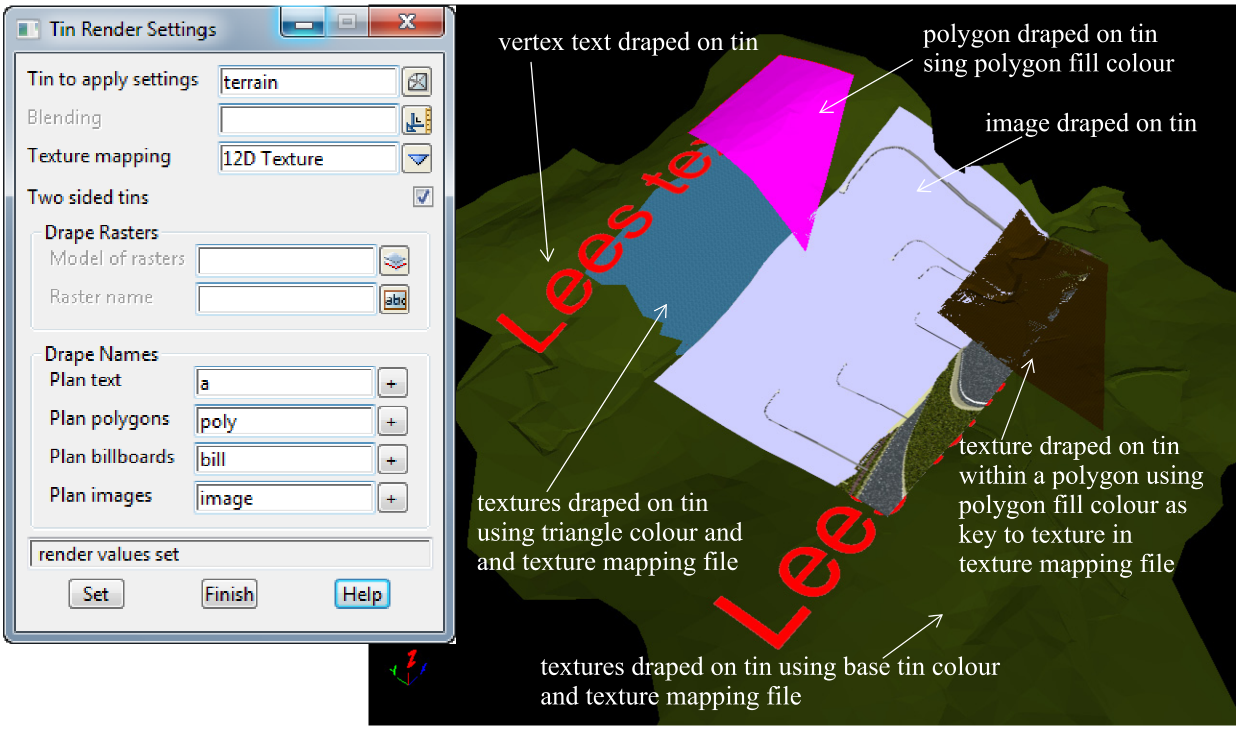
Plan images

when the tin is on a Perspective OpenGL view, then for all super strings in models also added to that view, if a super string in the model has an Integer string attribute with this name and any value, all images on the super string are draped onto the tin.

*The Integer string attribute of this name is set using the option **Strings =>Properties =>Attributes** (see* 14.12.8 Attributes*).*

Set button

apply the data in the panels fields to the tin.


Tin Render Settings
Tin to apply settings
terrain
Blending
Texture mapping
12D Texture
Two sided tins
Drape Rasters
Model of rasters
Raster name
Drape Names
Plan text
a
Plan polygons
poly
Plan billboards
bill
Plan images
image
render values set
Set
Finish
Help
vertex text draped on tin
polygon draped on tin
sing polygon fill colour
image draped on tin
textures draped on tin
using triangle colour and
and texture mapping file
texture draped on tin
within a polygon using
polygon fill colour as
key to texture in
texture mapping file
textures draped on tin using base tin colour
and texture mapping file

12.13.2 Tin Height Render Settings

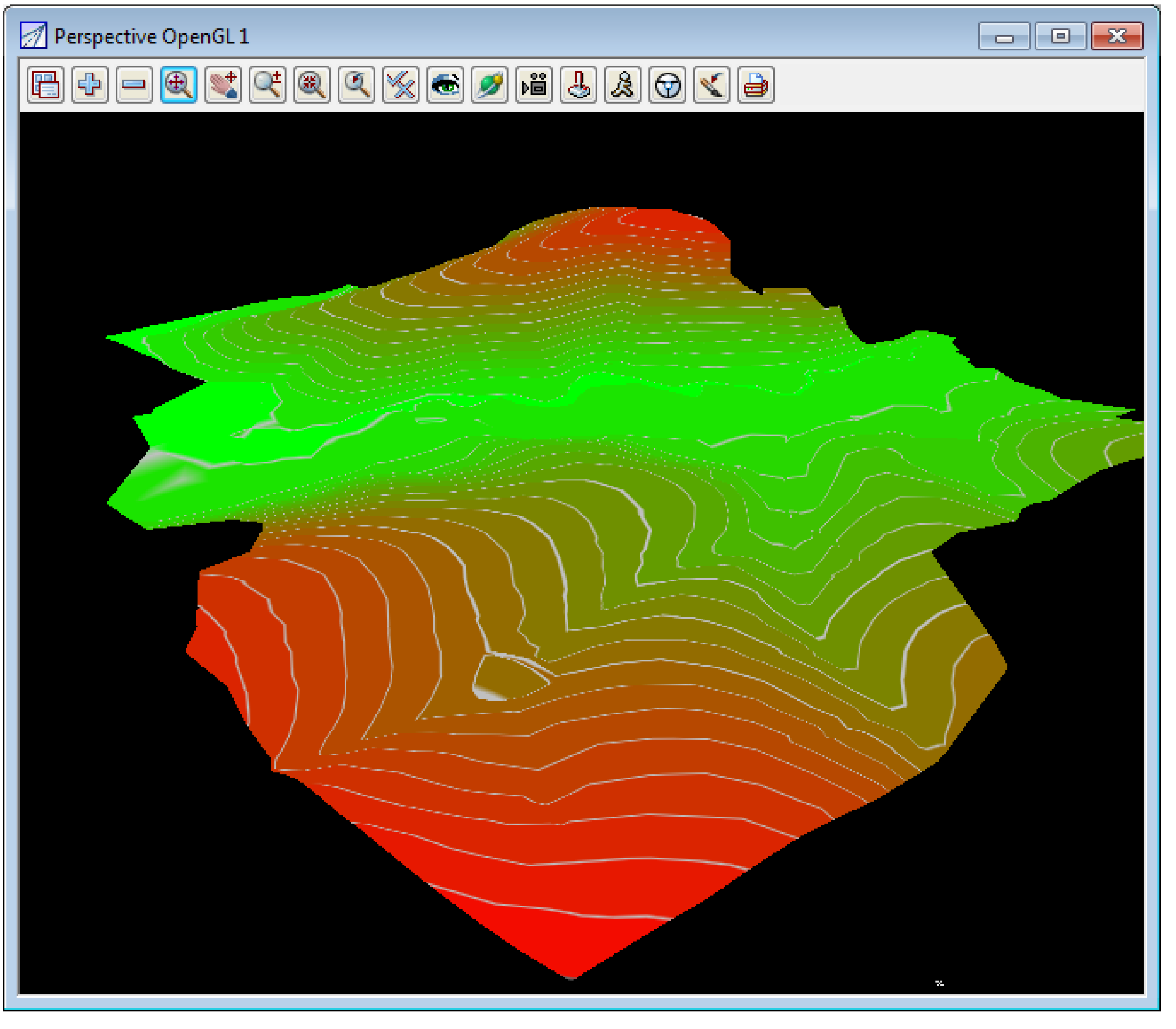
Position of option on menu: **View =>Visualisation => Tin height render settings**

For a tin (that is not a super tin) the *Tin Height Render Setting* option sets the a texture map and parameters that can be stretched over the specified height range.

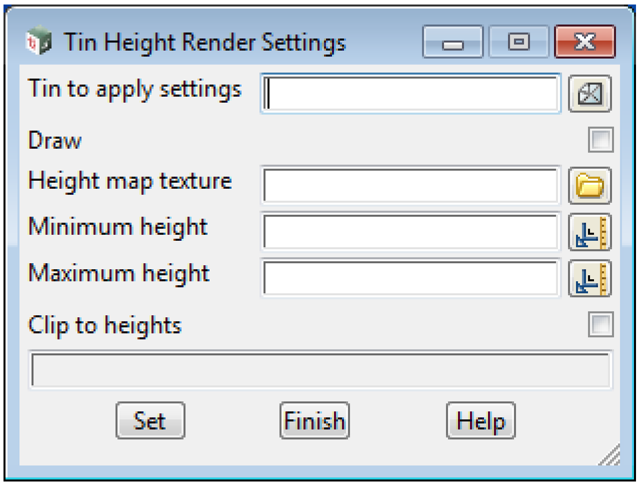
To see the effect, the tin must be on a *Perspective OpenGL* view with *Shade* ticked on, and you have the **Visualisation** module.

Notes

1. this only displays when you have the **Visualisation** module.
2. This option does not work for a Super Tin.



Selecting **Tin height render settings** brings up the **Tin Height Render Settings** panel:



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin to apply settings	tin box		available tins

the tin to apply the height texture map to. This can then be toggle on/off with the Draw *tick box.*

Draw	tick box	not ticked	

*if **ticked**, the height texture map will be applied to the tin when it is on an Perspective OpenGL view.*

*If **not ticked**, the height texture map is **not** applied to the tin. However the height texture map is still associated with the tin and* Draw *can be ticked back on at any time.*

Height map texture	file box		.bmp, png file

the texture map to stretched over the height range given in the Minimum height *and* Maximum height *fields.*

The texture map is a bitmap that is one pixel high by n pixels wide, usually with a colour gradation going from left to right. The bitmap is stretched over the minimum and maximum height with the left side of the texture map being at the minimum height and the right side of the texture map being at the maximum height.

For example, the file Height_map_texture.bmp from Library.

Minimum height	measure box		available measures

the minimum height to stretch the height map texture over.

Maximum height	measure box		available measures

the maximum height to stretch the height map texture over.

Clip to heights	tick box	not ticked	

*if **ticked**, the tin is only displayed where the height map texture is applied. That is, the tin is only displayed between the* Minimum height *and the* Maximum height.

*If **not ticked**, the tin is displayed over the entire range of the tin.*

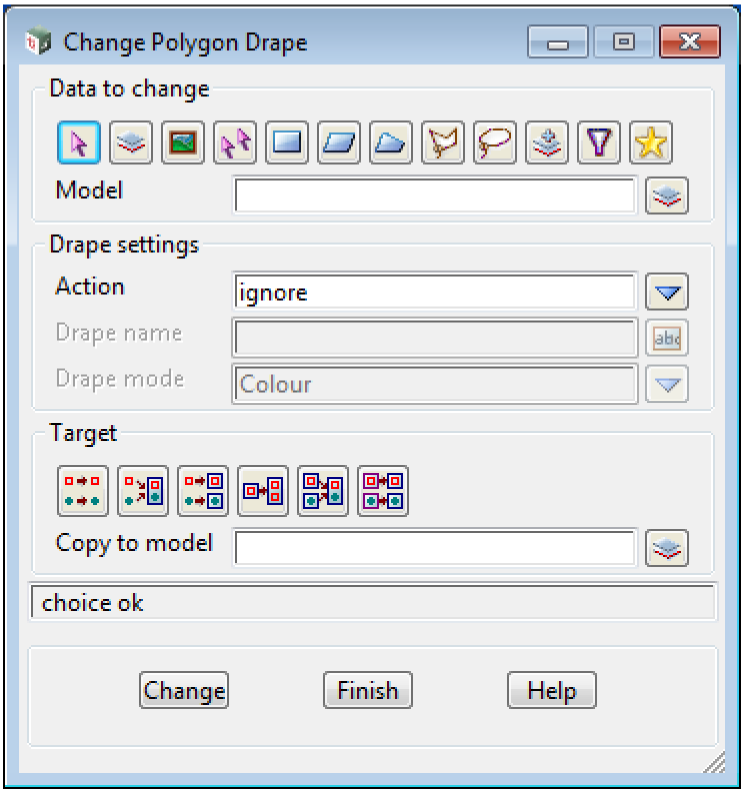
Set	button		

set the values in the panel for the tin given in Tin to apply settings.

12.13.3 Render Drape

Position of option on menu: **View =>Visualisation => Render drape**

Selecting **Render drape** brings up the **Change Polygon Drape** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Action	choice box	ignore	set, clear, ignore

*if **set**, for each selected polygon, an Integer string attribute is created with the atttribute name* Drape name *and the value of the attribute is*

0 if Drape mode *is* **Colour** *and*

1 if Drape mode *is* ***Colour to texture.***

*If **clear**, for each selected polygon the Integer string attribute with name* Drape name *is removed.*

*If **ignore**, nothing is done.*

Drape name

*name of the Integer attribute that is set for the string. The attribute name is used to identify these polygon in the **View->Visualisation->Tin Render Settings*** 16.6.7 Render Settings *(see* Plan polygons *in the Drape Names section)*

Drape mode	choice box		Colour, Colour to texture

if Colour *- the fill colours of the selected polygons are used for the drape*

If Colour to texture *- the fill colours of the selected polygons are used to reference the texture in the **Texture Mapping** file*

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target .

Target info input

extra information required for the target.

Change button

process the selected strings.

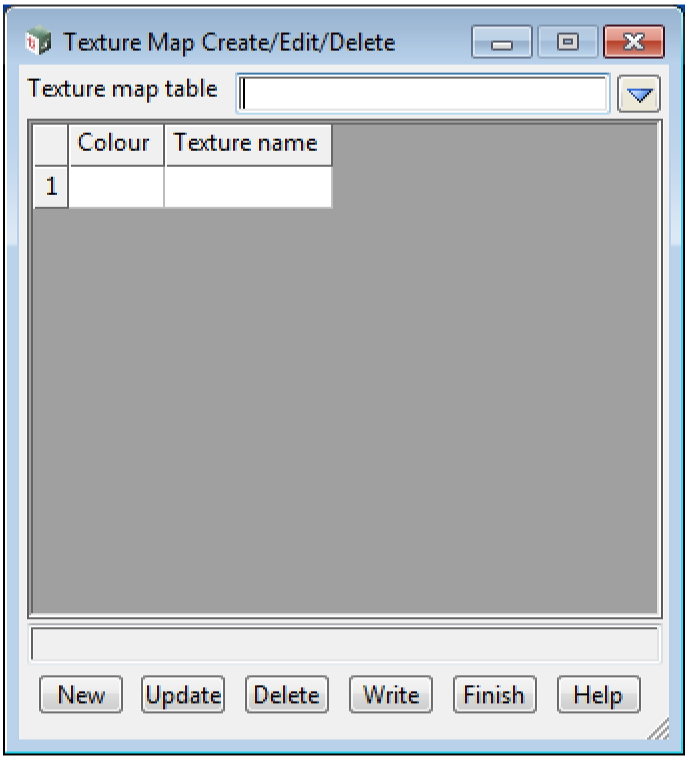
12.13.4 Texture Map Edit

Position of option on menu: **Visualisation =>Texture map edit**

This option edits the texture mapping file (texture_map.4d) which contains the *texture map tables* that can be applied to the coloured triangles of a tin in renderings.

Each texture map table must have a unique name.

On selecting the **Text map edit** option, the **Texture Map Create/Edit/Delete** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

Texture map table

name of the texture map table inside the texture map file.

Colour colour box available colours

colour to apply the texture to.

Texture name texture box available textures

texture to apply to all triangles with the specified colour

New button

to create a new texture map table, click on the **New** *button and then type the texture map table name into the* Texture map table *field. Then start filling in the* Colour *and* Texture name *columns.*

Update button

update the given Texture map table *with the values in the* Colour *and* Texture name *columns.*

Delete button

Delete the texture map table named in the Texture map table *field.*

Write button

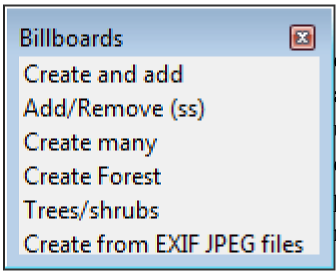
write out the texture map information to the texture map file, texture_map.4d.

12.13.5 Billboards

Position of menu: **View =>Visualisation =>Billboards**

The billboard options plane an image onto a rectangular billboard. this can be used to display background scenes, signs etc.

The **Billboard** walk-right menu is



create a billboard and add an image to it
add/remove a billboard to the vertices of a super string
use images to create entries in the billboards.4d file
create a forest within a polygon
place a billboard tree/shrub
read in JPEG files which include EXIF and GPS information

For the option *Create and add, go to* 12.13.5.1 Create and Place Billboard.
Add/Remove (ss) 12.13.5.2 Add and Remove Billboard from Super String
Create many 12.13.5.3 12D - Billboard Signs Create
Create forest 12.13.5.4 Create Forest
Trees/shrubs 12.13.5.5 Trees/Shrubs as Faces and Billboards
Create from EXIF JPEG files 12.13.5.6 Create Billboards from EXIF JPEG Files

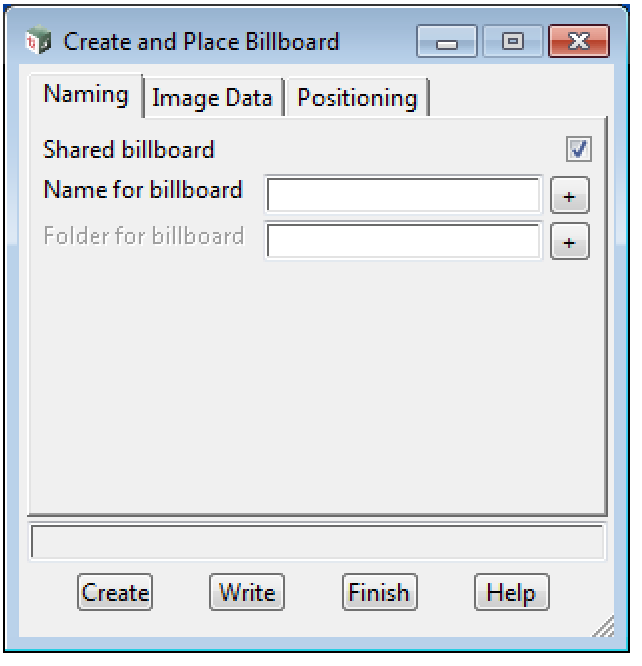
12.13.5.1 Create and Place Billboard

Position of option on menu: **View =>Visualisation => Billboards =>Create and add**

This option defines a billboard (its size and position, the image on the billboard, *etc.*) and adds it to a super string and can **also** add it to the file *billboards.4d* which is used by other options.

The option **always** creates a super string with the billboard on it, so if you just wanted to add a billboard definition to the *billboards.4d* file, then use this option which creates a super string with the billboard, but then **delete** the created super string.

On selecting the **Create and add** option, the **Create and Place Billboards** panel is displayed.



The fields and buttons used in this panels have the following functions.

Field Description	Type	Defaults	Pop-Up
Naming tab	**tab**		
Shared billboard	tick box	ticked	

*if **ticked**, the information about the billboard is added to the billboards.4d file and then when the super string is created, it only references the required billboard information from the billboards.4d file. Note: the* **Write** *button needs to be clicked to write out the billboards.4d file with the new billboard definition. If **not ticked**, all the information about the billboard is saved with the super string and nothing is written to, or used from the billboards.4d file.*

Name for billboard billboard box

the name of the billboard to use in the billboards.4d file. This must be unique amongst all the billboard names in the billboards.4d file.

Folder for billboard text box

the billboard folder in which to place the billboard named above. This is just a grouping mechanism for all the billboard names. If the Folder does not exist in the billboards.4d file, it will be created.

Image Data tab **tab**

Image file file box

name of the file containing the image to place on the billboard

Alpha file file box

currently not supported

Image polygon polygon box

this polygon is used to crop the image. The billboard origin (0,0) is at the centre of the image and the world size of the image is determined by the Width *and* Height *fields below. The Image polygon must also be drawn with the same (0,0) reference and in world units.*

Number of fans integer box

the billboard can consist of multiple copies of the image rotated equally around 180 degrees of the z-axis at x=0. This is mainly used for trees so that you see a tree no matter which direction you look from.
If Number of fans *= 1, then the image is used on both sides of the billboard.*
If Number of fans i*s left blank, then the image is used on one side of the billboard and the side is given the colour specified in the* Colour *field on the Positioning tab.*

Width measure box

the width in world units for the billboard (before the billboard it is cropped by any Image polygon).

Height measure box

the height in world units for the billboard (before the billboard is cropped by any Image polygon).

Positioning tab **tab**

Model model box

model for the created super string with the billboard.

Coordinate measure box

the (x,y,z) coordinate of the vertex of the super string which has the billboard placed on it.

Colour colour box

the colour used for the back of a billboard when Number of fans is left blank. Note if number of fans is 1 or more then the same image is used on both sides of the blades of the fan.

Billboard type choice box Vertical Vertical, Threed, Plan

*if **Vertical**, the string is placed vertically (upright) and nothing else is needed.*
*If **Threed**, a grid for entering Factor, Rotate and Offset values is displayed and these are used to control the position of the billboard in 3d.*
*If **Plan**, the billboard is a horizontal billboard and will draw on a Plan view. A billboard must be a Plan (or Horizontal) billboard when using* Drape names *for* **Plan billboards** *in the* **Tin Render Settings panel** *to drape horizontal billboards onto a tin (see* 12.13.1 Tin Render Settings*).*

Angle measure box

the angle the billboard is rotated in the (x,y) plane. Angle is measured counterclockwise in degrees in 4.17.1 HP Notation *with zero along the x-axis.*

Offset x, Offset y, Offset z

x/y/z distance to place the billboard away from the super string vertex.

Buttons at Bottom

Create button

creates the super string with the billboard on it.

Write button

writes out the new billboards.4d file.

12.13.5.2 Add and Remove Billboard from Super String

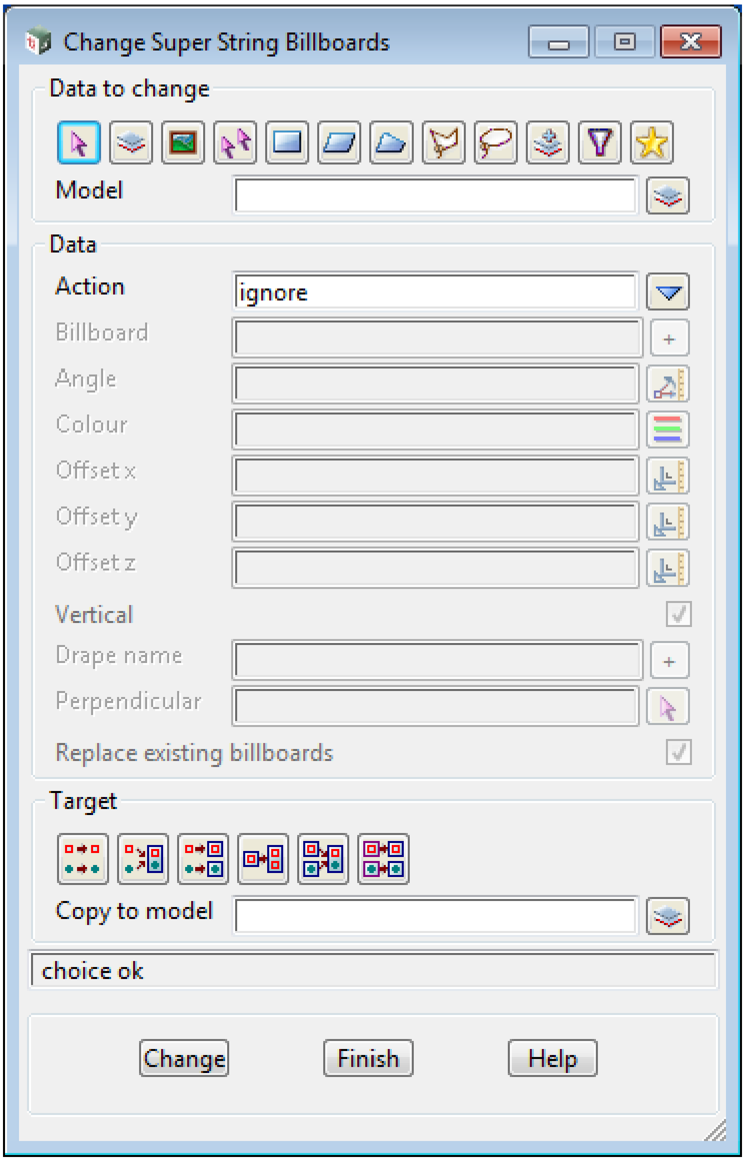
Position of option on menu: **View =>Visualisation => Billboards =>Add/Remove (ss)**

Each vertex of a super string can have its own billboard.

A billboard consists of an image and a width and height.

The ***Add/remove from super string*** option places a selected billboard either in a vertical plane or as a plan image at a user given distance from each vertex of a super string. The image it at a given angle or perpendicular to a selected string. An example could be the image on a stop sign which is in the vertical plane.

Selecting **Add/Remove (ss)** brings up the **Change Super String Billboard** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Action choice box ignore set, clear, ignore

*if **set**, the billboard is placed at each vertex in the super string.*
*If **clear**, the billboard information at each vertex is cleared.*
*If **ignore**, nothing is done to the billboard information.*

Billboard billboard box

billboard to be used.

Angle angle box

angle to rotation the billboard - positive is counter clockwise. This is ignored if Perpendicular *is used.*

Colour colour box

colour for the back of the billboard.

Offset x, Offset y, Offset z

x/y/z distance to place the billboard away from the super string vertex.

Vertical tick box

*if **ticked**, the string is placed vertically and nothing else is needed.*
*If **not ticked**, the billboard is a horizontal billboard and will draw on a Plan view. This is also needed when using* Drape names *for* **Plan billboards** *in the* **Tin Render Settings panel** *to drape horizontal billboards onto a tin (see* 12.13.1 Tin Render Settings*).*

Drape name input

if Vertical is not ticked, then this field can be modified.
If not blank, an Integer string Attribute is created with this name. The Drape name *is used in the* **Tin Render Settings panel** *(see* 12.13.1 Tin Render Settings*) for draping horizontal billboards onto a tin.*

Perpendicular string select box

if a string is selected, then the billboard is placed on the line going through the vertex and perpendicular to the selected string. Angle *is ignored.*

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target.

Target info input

extra information required for the target.

Change button

process the selected strings.

12.13.5.3 12D - Billboard Signs Create

Position of option on menu: View =>Visualisation =>Billboards =>Create Many

This option takes images and creates entries in the billboards.4d file.

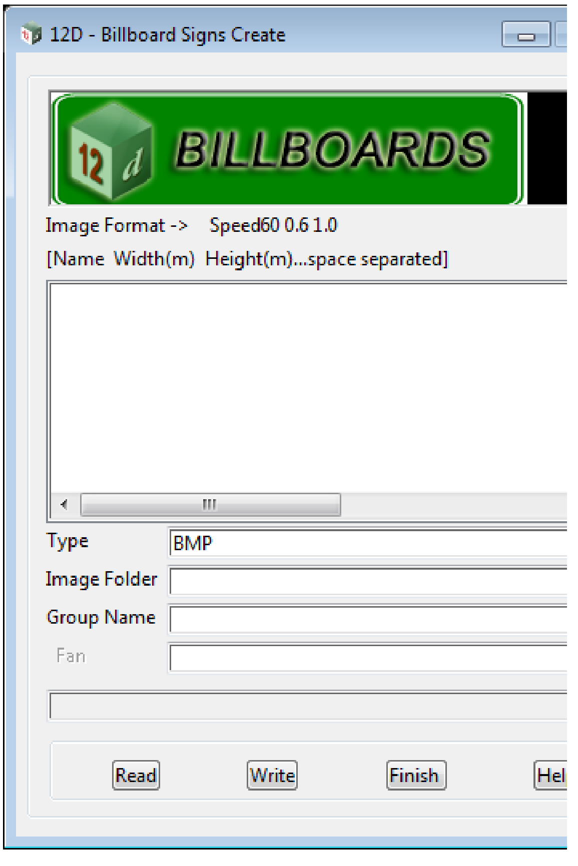
Note that it does not create a super string with a billboard on it.

The option requires that you rename your images in the appropriate format:

Name Width (m) Height (m) -> space separated.

For example, "sign 20 5.jpg" creates a billboard with the name "sign" which is 20 units wide and 5 units high.

Selecting **Create Many** displays the **12D - Billboard Signs Create** panel:



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Type	choice box	JPG	BMP, JPG, PNG,TGA, ALL

Type of image files to read in.

Image Folder file box

Browse to a dedicated folder. File names are used to create the billboard name, width and height, e.g. sign 20 5.jpg *creates a billboard with the name* sign *20 units wide and 5 units high.*

Group Name text box

The billboard group to add the selected images to.

Fan number box

*If **not blank**, the number of extra fans to add to the billboard.*
*If **blank**, no extra fans will be added to the billboard.*

Read button

Loads all the images of the specified type into 12d and adds them to the Group specified in the billboards.4d file. If the Group does not exist it is created.

Write button

Appends the new group to the billboards.4d file. You must restart 12d load the new billboards.4d file.

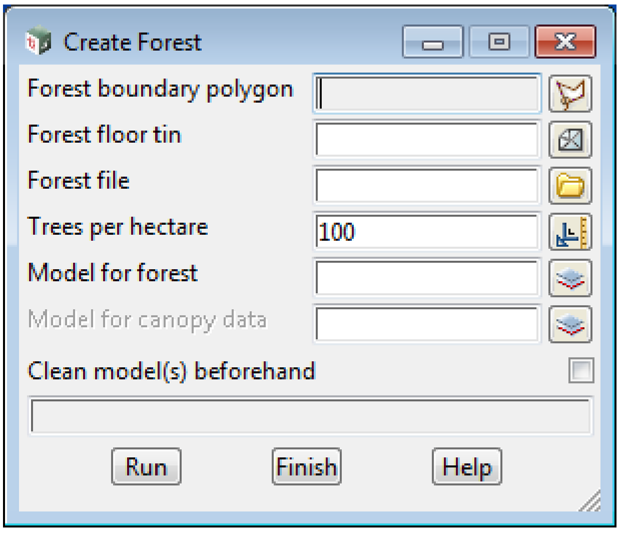
12.13.5.4 Create Forest

Position of option on menu: **Visualisation =>Billboards =>Create forest**

Create forest creates randomly placed trees and shrubs in a selected polygon.

The forest make up is controlled by a user defined forest file which gives the types and distribution of trees, and variation in size and spread.

Selecting **Create forest** displays the **Create Forest** panel



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Forest boundary polygon	string select		

polygon to create the forest within

Forest floor tin	tin box		available tins

tin to use for the z-value for the base of the trees

Forest file	file box		.forests files

file to give the make up of the forest. For more information, go to 7.6.14 Forest File

Trees per hectare	input		

density of the forest given by the number of trees per hectare

Model for forest	model box		available models

model to place the trees in

Model for canopy data	model box		available models

*if **non blank**, points are created for the tops of the trees in the forest. These can be used to triangulated to give a top of trees tin.*

Clean model (s) beforehand	tick box		

*if **ticked**, the forest and forest canopy models are cleaned before the option is run*

Run	button		

run the option

12.13.5.5 Trees/Shrubs as Faces and Billboards

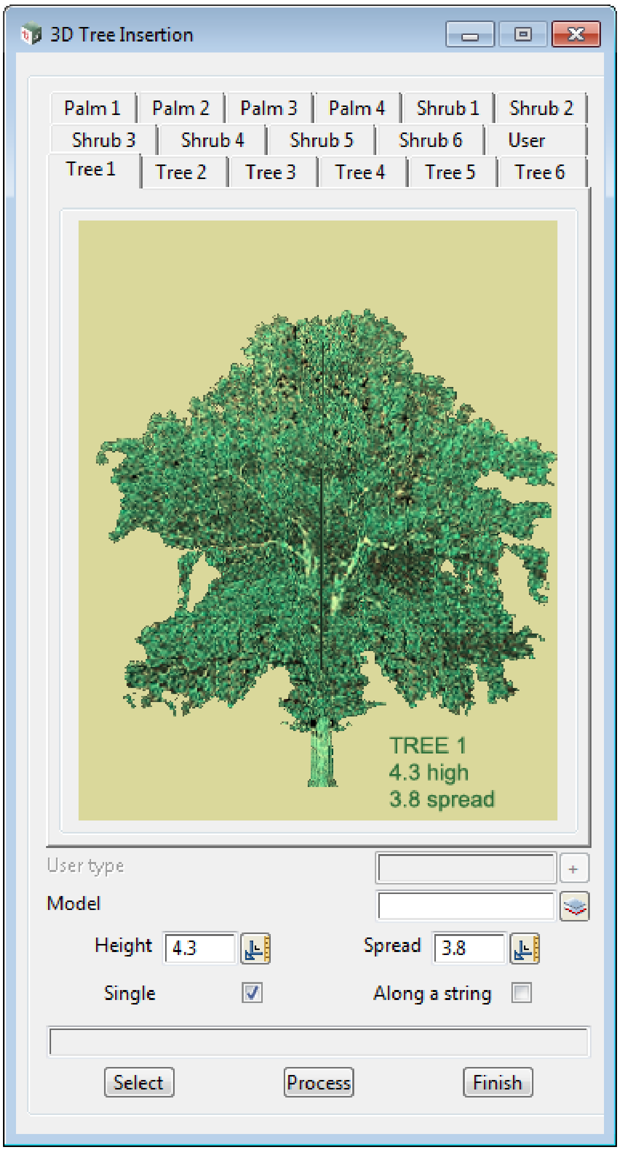
Position of option on menu: View =>Visualisation => Billboards =>Trees/shrubs

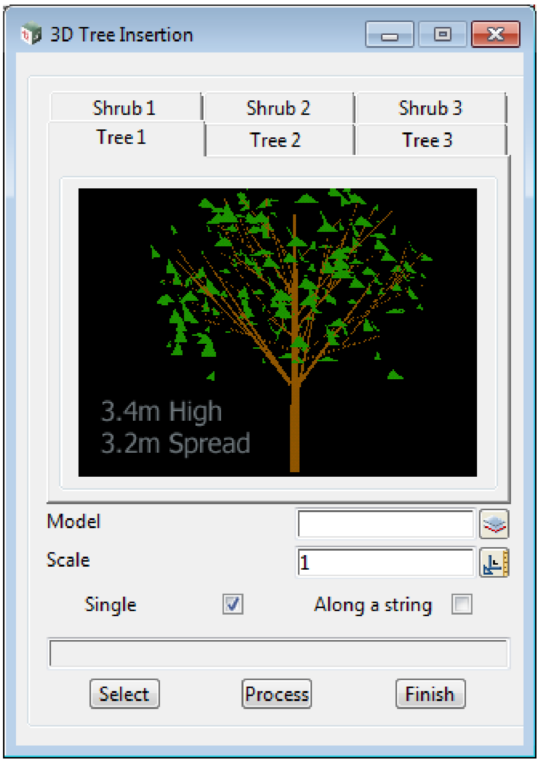
Position of option on menu: View =>Visualisation => Utilities =>Trees/shrubs faces

The **Trees/shrubs - faces** option inserts the trees and shrubs as face elements. This means that each tree may consist of hundreds or thousands of faces.

The **Trees/shrubs** option inserts the images of trees and shrubs as a fanned billboard. This means that each tree consists of only one points.

On selecting the **Trees/shrubs** option, the **3D Tree Insertion** panel is displayed and on selecting the **Trees/shrubs faces** option, the **3D Tree Insertion - Faces** panel is displayed.





The fields and buttons used in these panels have the following functions.

Field Description Type Defaults Pop-Up

Tree and Shrub tabs

click on the tab to display the type of tree/shrub to be inserted. The picture shows the tree height and spread.

Model model box available models

model to add the tree/shrub to.

Height and Spread real value varies measures

parameters for scaling

Single radio button

if ticked then the tree is placed at the selected position.

Along a string radio button

if ticked then the tree is placed along a string. When the string is selected then a **Tree Interval** *panel is brought up to specify how the trees are placed along the string.*

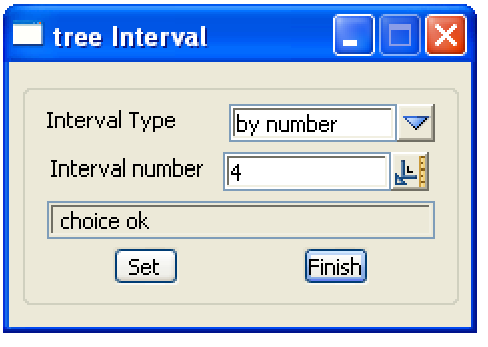
Select button

select the string or position for the tree.

Process button

insert the trees/shrubs.

Tree Interval



This panel is for specifying how often the trees/shrubs are to be inserted along the string.

Field Description	Type	Defaults	Pop-Up
Interval type	choice box	by number	by number by distance by special chg file by vertex

if ***by number****: an* Interval number *field is displayed on the panel and the tree is inserted that many times along the string.*

if ***by distance****: an* Interval distance *field is displayed on the panel and the tree is inserted with that distance separation along the string.*

if ***by special chg file****: a* Special chg file *field is displayed on the panel and the tree is inserted along the string at the chainages given in the file.*

if ***by vertex****: the tree is inserted at every vertex of the string.*

Set button

set the parameters for tree interval to those in the panel.

12.13.5.6 Create Billboards from EXIF JPEG Files

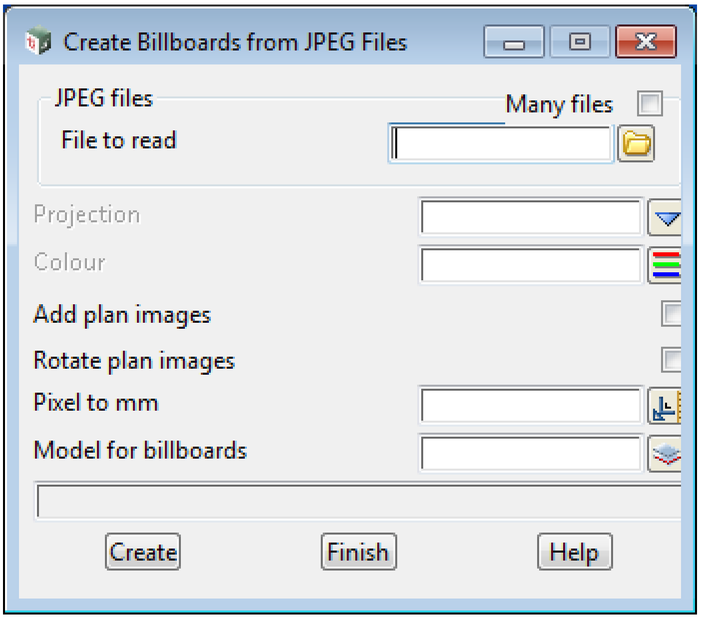
Position of option on menu: **View =>Visualisation => Billboards =>Create from EXIF JPEG Files**

This option reads in JPEG files which must include EXIF and GPS information. This information is then used to create and position a billboard with the image on the billboard.

If a Projection is selected by the user, then it is used to convert the lat/long of the GPS values to the XYZ co-ordinates. If no Projection is given, then the co-ordinates are left as lat/long.

Note: EXIF is short for Exchangeable Image File, a format that is a standard for storing interchange information in digital photography image files using JPEG compression. Almost all new digital cameras use the EXIF annotation, storing information on the image such as shutter speed, exposure compensation, F number, what metering system was used, if a flash was used, ISO number, date and time the image was taken, white balance, auxiliary lenses that were used and resolution.

Selecting **Create from EXIF JPEG Files** brings up the **Create Billboards from JPEG Files** panel.



The fields and buttons in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

JPEG files

Advanced	tick box	not ticked	

clicking **Advanced** *brings up a grid to allow the user to enter many JPEG files.*

File to read	file box		available *.jpg files

the JPEG files to read in and create images from. The JPEG file must include EXIF and GPS information, which is used to position the image. If Projection *is not blank, then it is used to convert the lat/long of the GPS values to the XYZ co-ordinates. If* Projection *is blank, then the co-ordinates are left as lat/long.*

Projection	choice box		

if a Project Projection is set, then it is placed in the Projection *field. This can be changed to any other projection. The projection is used to convert the lat/long of the GPS values to the XYZ co-ordinates. If no projection is set, then the co-ordinates are left as lat/long.*

Colour	colour box		available colours

*if **not blank** and* **Add plan images** *is not ticked, then this colour is used for the back of the billboard. If **blank**, the colour is white.*

Add plan images tick box not ticked

*if **ticked**, plan images are created instead of billboards.*

Rotate plan images tick box not ticked

*if **ticked** and* **Add plan images** *is ticked, the rotate value from the EXIF will be used to rotate the image.*

Pixel to mm measure box

the units for images are pixels (width and height). The pixel width and height of the image are multiplied by the **Pixel to mm** *(mm being millimetres) value to give width and height size in world unit (metres) which is needed when inserting the image.*

Model for billboards model box

the model for the image to be inserted into.

Create button

*when all the fields have been entered, the **Create** button creates images from the selected files.*

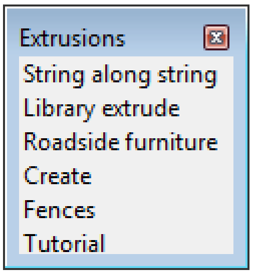
12.13.6 Extrusions

Position of menu: **View =>Visualisation =>Extrusions**

In it basic form, an extrusion is taking a cross-section and pushing it (extrude it, or sweep it) along a string to create a 3d object. For example, a circle extruded along a string creates a pipe. For more information, go to the section 12.13.7 Defining Extrudes and Extrusions.

There is a library of extrusions read in for the project, usually stored in the file extrusions.4d.

The **Extrusions** walk-right menu is



extrude along s string
add/remove a extrusion for a super string
insert roadside furniture
create a complicated extrusion
extrude fences along strings
tutorial on creating complicated extrusions

For the option String along string, go to	12.13.8.1 String Extrude
Library extrude	12.13.8.2 Change Library Extrude
Roadside furniture	12.13.8.3 Roadside Furniture
Create	12.13.8.4 Create Group Extrusions
Fences	12.13.35.3 Fences
Tutorial	12.13.8.5 Tutorial to Create Group Extrusions

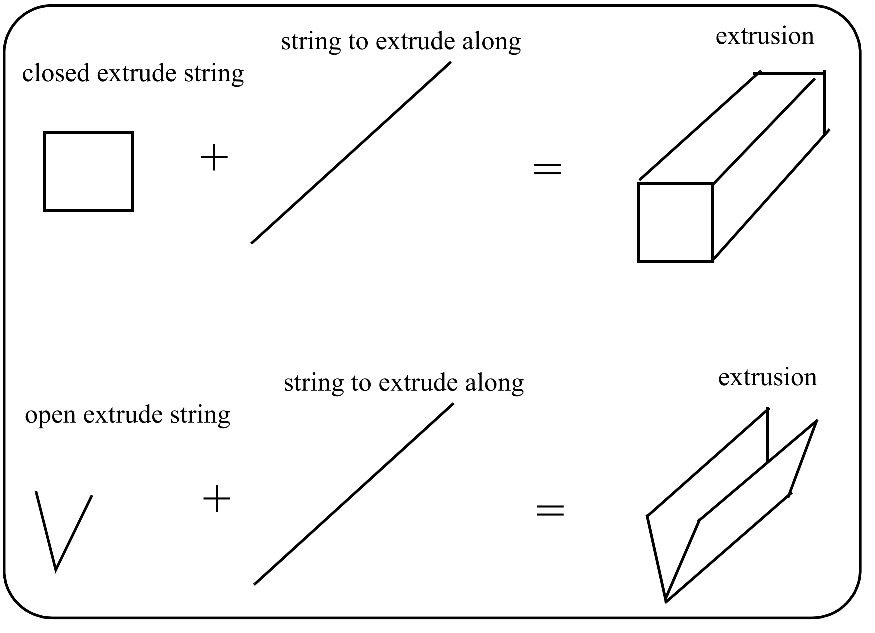
More information on extrudes is found in the section 12.13.7 Defining Extrudes and Extrusions

12.13.7 Defining Extrudes and Extrusions

In it basic form, an extrusion is simply taking a cross-section and pushing it (extrude it) along a string to create a 3d object. For example, a circle extruded along a string creates a pipe.

The string that is pushed along the string is called the ***extrude***.

Note: this is also know as a Sweep or Swept Path.

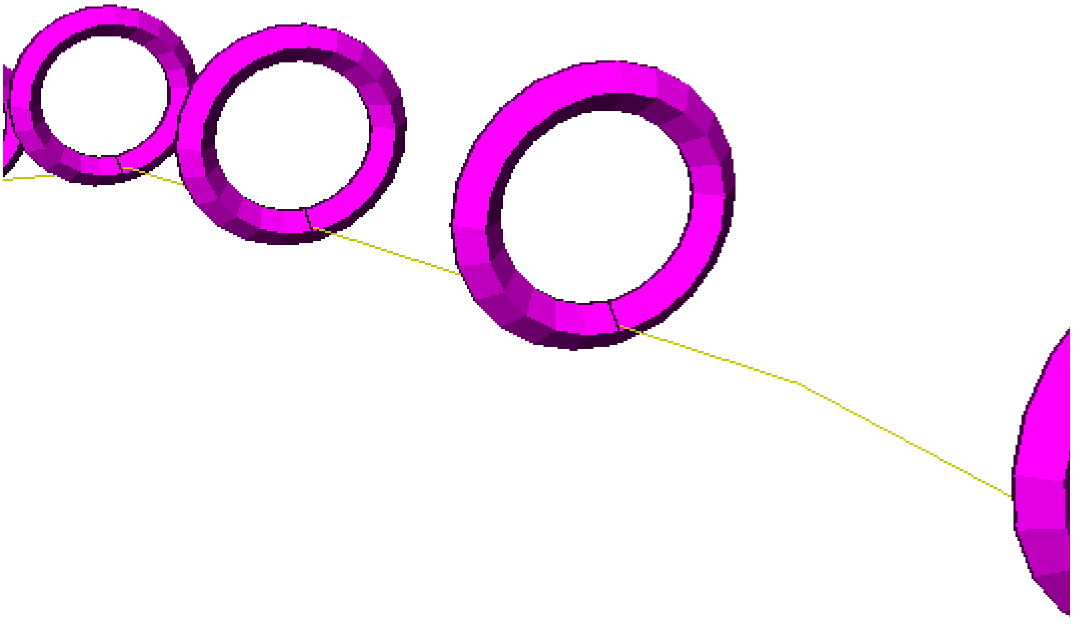


Only the (x,y) co-ordinates of the string used as the extrude are used and that gives the shape of the cross section with the ((0,0) point being the point where the string being extruded along sit.

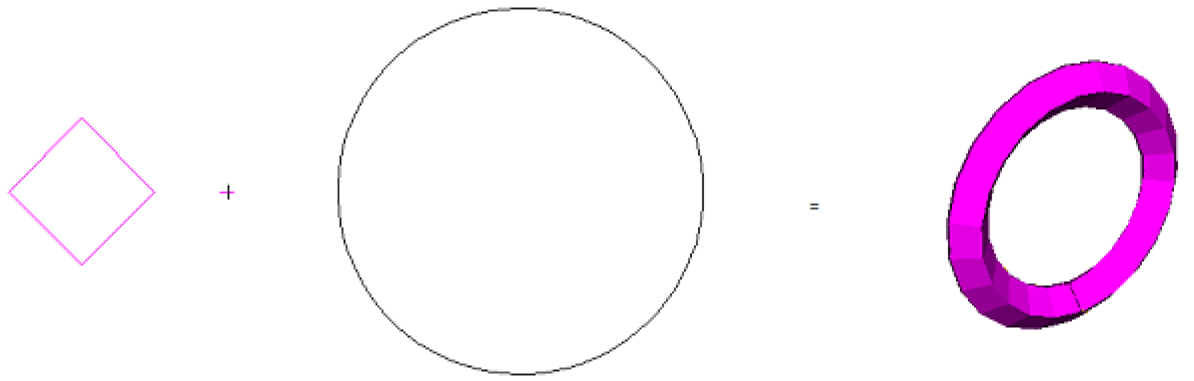
So the (x,y) co-ordinates of the extrude are being taken as (offset, height) in the plane perpendicular to the string being extruded along. So extrudes are normally defined around (0,0) in the (x,y) plane.

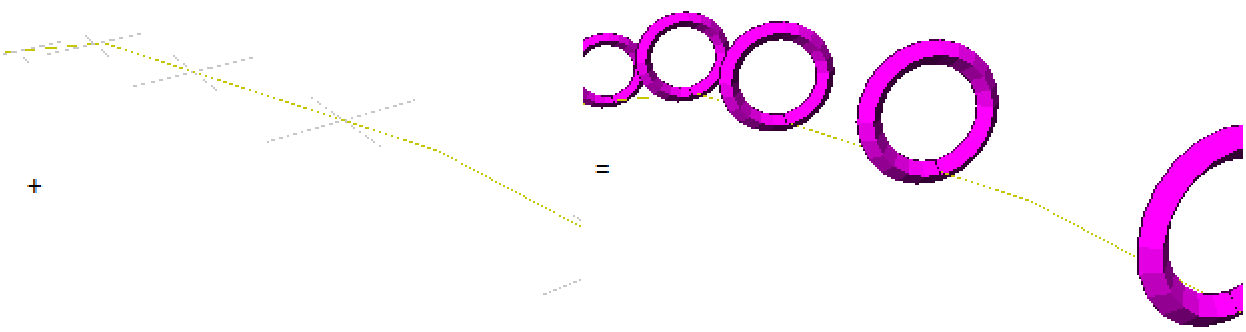
Interval extrude

The interval extrude can be seen as a series of string extrudes applied along the string at a specified interval distance. Here is a example of a 'Ring' interval extrude:



This extrude is defined by a diamond being extruded along a circle, and finally being instanced along the super string at the specified interval.





The extrusions.4d fragment defining this extrude is:

```
interval_extrude {

  name      "Vertical Ring"
  colour    true
  mirror_x  false

  interval   4.0

  factor_x   1.0
  factor_y   1.0
  factor_z   1.0

  rotate_x   0.0
  rotate_y   90.0
  rotate_z   0.0

  offset_x   0.0
  offset_y   0.0
  offset_z   0.0
```

```
    at_grade   false

    path {

// the string that the extrude is drawn along

      name      "circle path"
      breakline line
      colour    white
      style     1
      closed    1

      data_3d {         // note must be defined with x,y,z data
        0 0.0 0.0
        1 0.0 0.0
      }
      radius_data {
        0.5
        0.5
      }
      interval {
        chord_arc   0.005
        distance    -999
      }
    }
    data {

// the shape of the extrude

      name      "path data"
      breakline line
      colour    magenta
      style     1
      closed    1

      data_2d {
        -0.1  0.0
         0.0  0.1
         0.1  0.0
         0.0 -0.1
      }
      interval {
        chord_arc   0.001
        distance    -999
      }
    }
  }
```

TODO: define group extrude:

A group extrude is a combination of string and interval extrudes.

12.13.8 3D Transformations

This transformation consists of factor (scale), rotation, and offset (translation) for x, y, and z. Transformations are always applied in the following order

rotate, factor, offset	about the axes	z, y, x

rotate z	occurs in the x-y plane (plan rotation)
rotate y	occurs in the x-z plane (slope rotation)
rotate x	occurs in the y-z plane (super-elevation rotation)
factor z	
factor y	note: factors are commutative (any order is the same)
factor x	
offset z	
offset y	note: offsets are commutative (and order is the same)
offset x	

Note that the rotations about each axis are in the ***local*** coordinate system of the object being rotated. So after each rotation, the axes are also rotated.

12.13.8.1 String Extrude

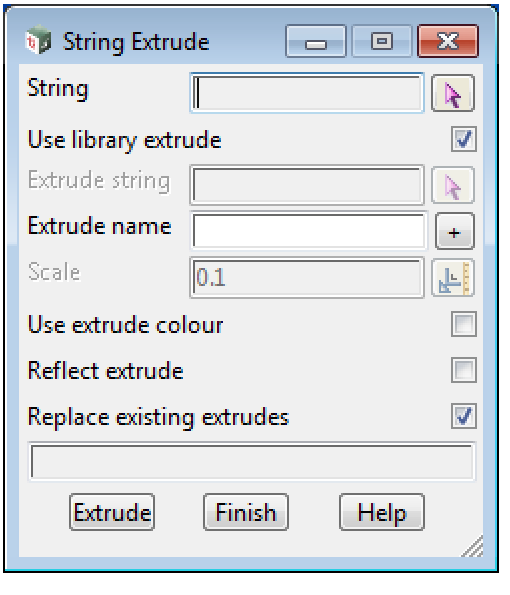
Position of option on menu: **View =>Visualisation => Extrude along string**

The **String along string** option extrudes a selected super string along a super string.

It can also be used to extrude a library extrude along string. But if a library extrude is to be applied to many strings, the Library Extrude option is more appropriate (12.13.8.2 Change Library Extrude)

More information on extrudes is found in the section 12.13.7 Defining Extrudes and Extrusions

On selecting the **Extrude** option, the **String Extrude** panel is displayed.



The fields and buttons in this panel have the following functions.

Field Description Type Defaults Pop-Up

String string select

super string to extrude along

Use library extrude tick box

if ticked, an existing extrusion is selected from the extrusion library. The extrude from the library is selected in the Extrude name *field.*
If not ticked, the Extrude string and Scale panel fields are used.

Extrude name extrude box

select the name of the extrude to use from the extrude library.

Extrude string string select

if a library extrude is not being used, then a string is selected to define the extrusion shape.

Scale real value input 0.1

value to multiply the extrude string co-ordinates by.

Use extrude colour tick box

if ticked, the placement uses the information in the grid on the panel.
if no ticked, then the string segment colour is used for the extrusion.

Reflect extrude tick box

if ticked, the extrude is reflected in the local y axis before it is applied.

Replace existing extrudes tick box

if ticked, all existing extrudes are cleared before the new one is applied.

Extrude button

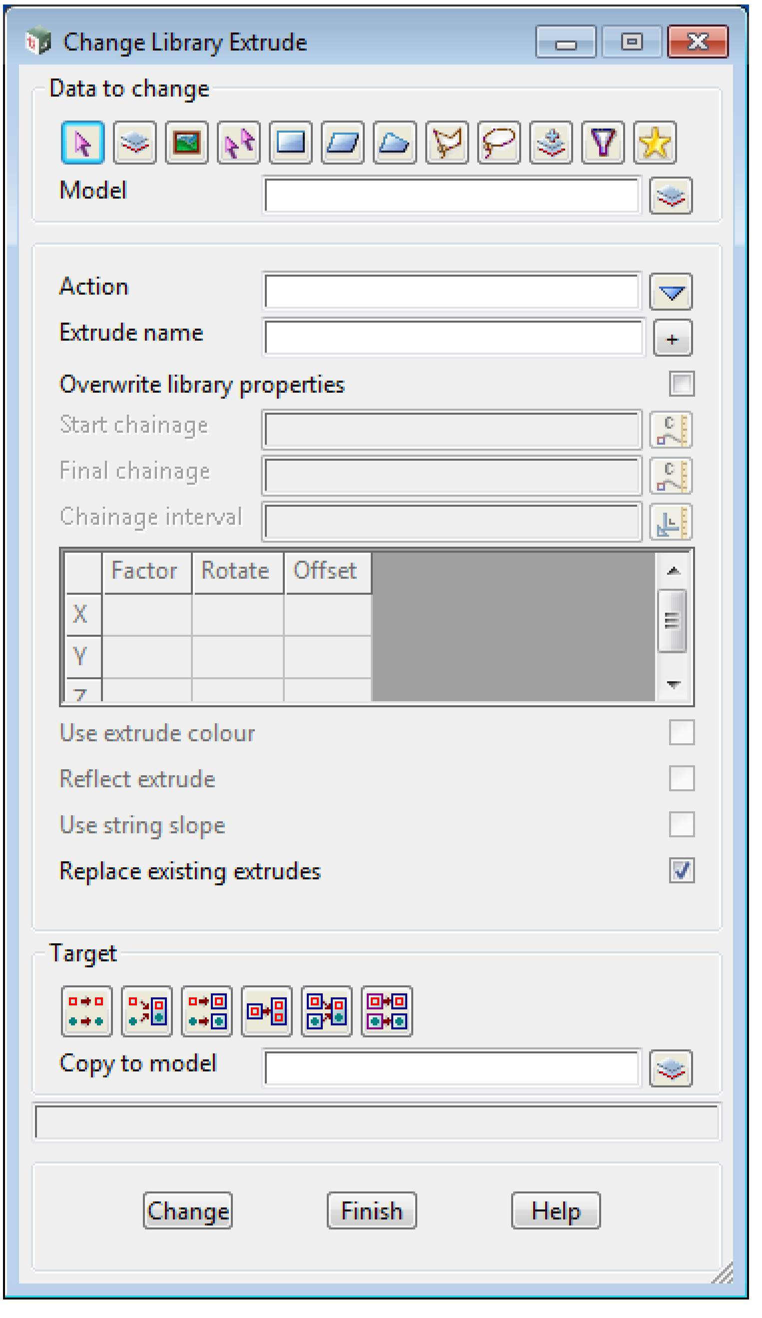
Create the extrusion.

12.13.8.2 Change Library Extrude

Position of option on menu: **Visualisation =>Library extrude**

This option can apply an extrusion from the extrusions library, extrusions.4d, to a super string, or change the extrusion applied to a super string, or remove an extrusion from a super string.

Selecting **Library extrude** brings up the **Change Library Extrude** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Action choice box set, clear

*if **clear**, all existing extrusions are removed from the selected super strings. The only other fields to be filled in is the* Target.
*If **set**, a library extrude is applied to the selected super strings.*

Extrude name extrude box list of extrusions

extrude to be applied to the super strings.

Overwrite library properties tick box

if ticked, the placement of the extrude uses the information in the grid on the panel.
If not ticked, the placement properties are taken from the extrude in the library.

Start/end chainage chainage box

start/end chainage for applying the extrusion.

Start/end chainage chainage box

start/end chainage for applying the extrusion. If blank, use the strings start/end.

Chainage interval

for an interval extrude only - interval to use for the extrude.

Grid of X,Y,X against Factor, Rotate, Offset

placement information for applying the extrude.

Use extrude colour tick box

if ticked, the placement uses the information in the grid on the panel.
if no ticked, then the string segment colour is used for the extrusion.

Reflect extrude tick box

if ticked, the extrude is reflected in the local y axis before it is applied.

Replace existing extrudes tick box

if ticked, all existing extrudes are cleared before the new one is applied.

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target.

Target info input

extra information required for the target.

Change button

process the selected string

12.13.8.3 Roadside Furniture

Position of option on menu: **View =>Visualisation => Extrusions =>Roadside furniture**

This option is for inserting extruded objects that have been supplied by 12D Solutions Pty Ltd.

It includes street lights, log barriers, park bench, posts and columns, walls, timber fences, guard rails, sighs etc.

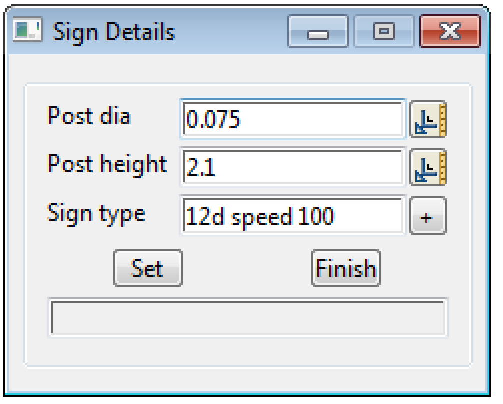
The list of items that can be inserted is being continuously upgraded by 12D Solutions.

Selecting **Roadside furniture** brings up the **Roadside Furniture** panel.

Clicking on the tabs brings up the information required for that piece of roadside furniture.

For all the tabs other than Park Bench, Street Light and Guard Rail, once a tab had been selected, moving the cursor over the picture brings up a special panel for the extra information required.

For example, for the Sign tab, moving the cursor over the picture brings up the Sign Details panel:



New 12D Ascii

This option is similar to the others e.g. Street Light, but allows the user to create their own features and place them in the same manner as the others on the panel.

Specifications for an ascii file*:*

The stem of the ascii file name, the model name in the ascii file and the stem of the bitmap name showing the picture of the object...... "MUST" be the same e.g. File: User Street Light.12da Model: User Street Light   Bitmap: User Street Light.bmp

Bitmap size to be 345W x 250H pixels

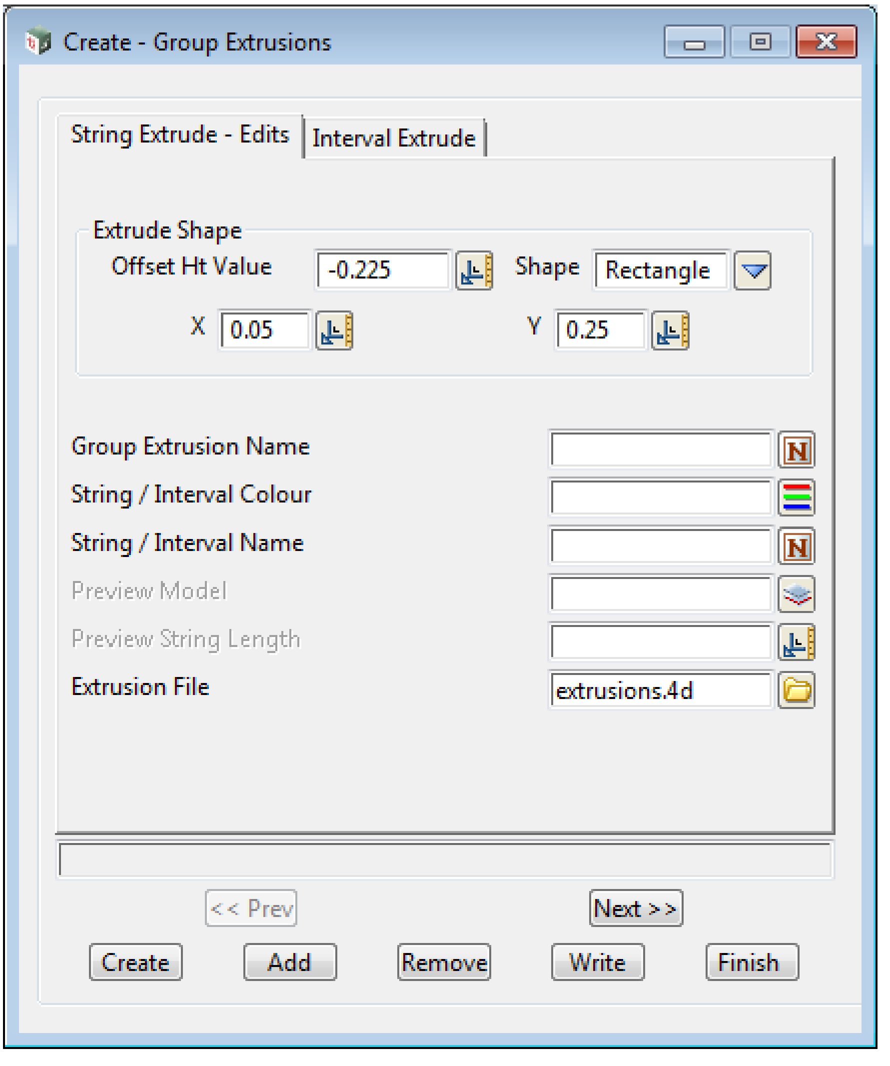
Place the 12da file and the bitmap in your User_Library

12.13.8.4 Create Group Extrusions

Position of option on menu: **View =>Visualisation => Extrusions =>Create**

The **Create** option creates groups extrusions and saves then to the extrusions.4d file.

Selecting **Create** brings up the **Create - Group Extrusions** panel.

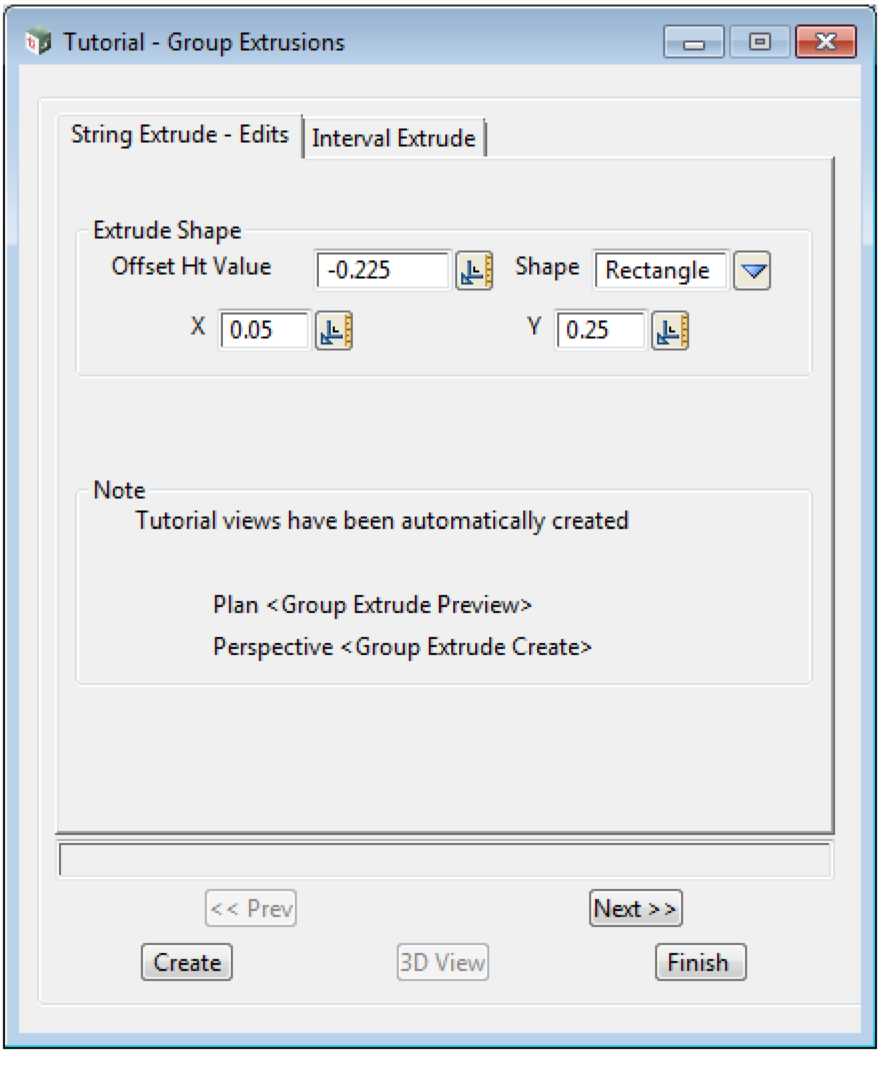


12.13.8.5 Tutorial to Create Group Extrusions

Position of option on menu: **View =>Visualisation => Extrusions =>Tutorial**

The **Tutorial** option demonstrates how to creates groups extrusions and will also create groups extrusions.

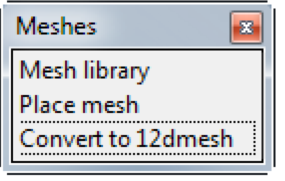
Selecting **Tutorial** brings up the **Tutorial - Group Extrusions** panel.



12.13.9 Meshes

Position of menu: **View =>Visualisation =>Meshes**

The **Meshes** walk-right menu is



For the option Mesh library go to 12.13.9.1 Mesh Library
Place Mesh 12.13.9.3 Place a Mesh
Convert to 12dmesh 12.13.9.4 Convert to 12dMesh Format

12.13.9.1 Mesh Library

Position of option on menu: **View =>Visualisation =>Meshes =>Mesh Library**

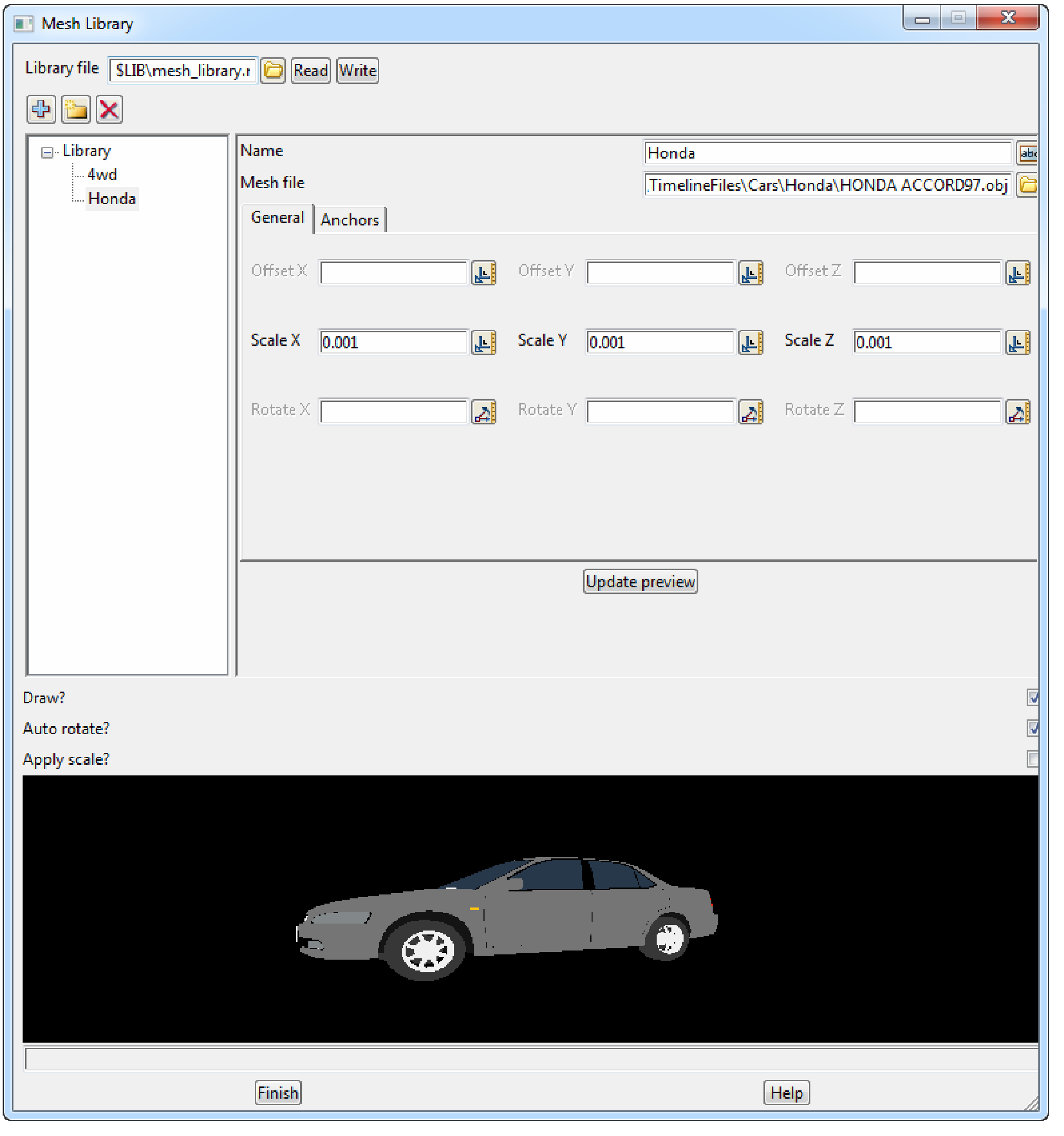
Position of option on menu: **View =>Visualisation =>Timelines =>Mesh Library**

The **Mesh Library** allows you to predefine meshes and the standard transformations required to get them into a world co-ordinate system. As meshes may come from a variety of packages that use arbitrary scales, rotations or offsets, this may be necessary.

It can also be used to define anchors, to anchor a mesh to the tin. This can be useful to get your mesh to conform to a terrain, such as a vehicle running down a road.

The mesh can then be written to the mesh library file, mesh_library.4d

Selecting **Mesh Library** brings up the **Mesh Library** panel.



The fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

Library file file

the mesh library file to read/write

Read button

reads a mesh library file

Write button

writes a mesh library file

+ (Add) button

adds a new mesh

Folder (Create Folder)

Creates a new folder for categorising meshes

Delete

Deletes a mesh or folder

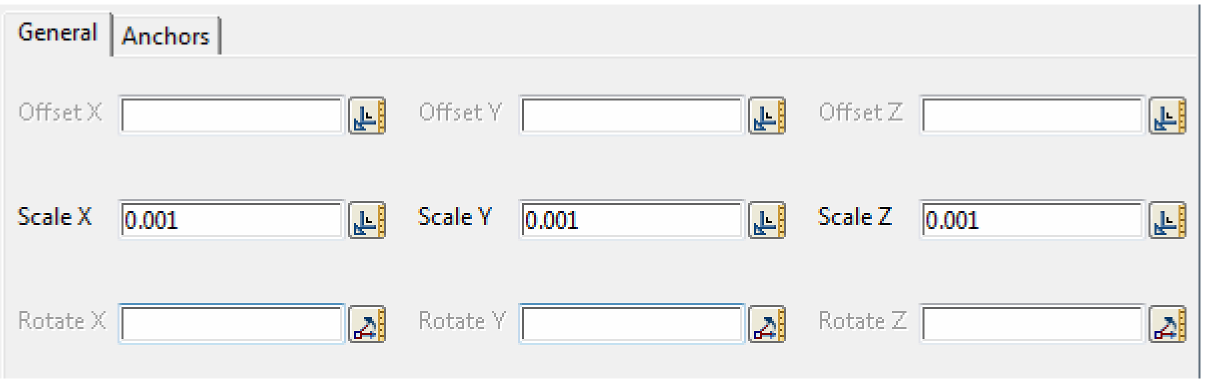
Name

the name of the current mesh

Mesh file

the mesh file, in OBJ format

General tab



Offset x

an optional offset along the x axis to apply

Offset y

an optional offset along the y axis to apply

Offset z

an optional offset along the z axis to apply

Scale x

an optional scale to apply to the x scale of the object

Scale y

an optional scale to apply to the y scale of the object

Scale z

an optional scale to apply to the z scale of the object

Rotate x

an optional rotation around the x axis

Rotate y

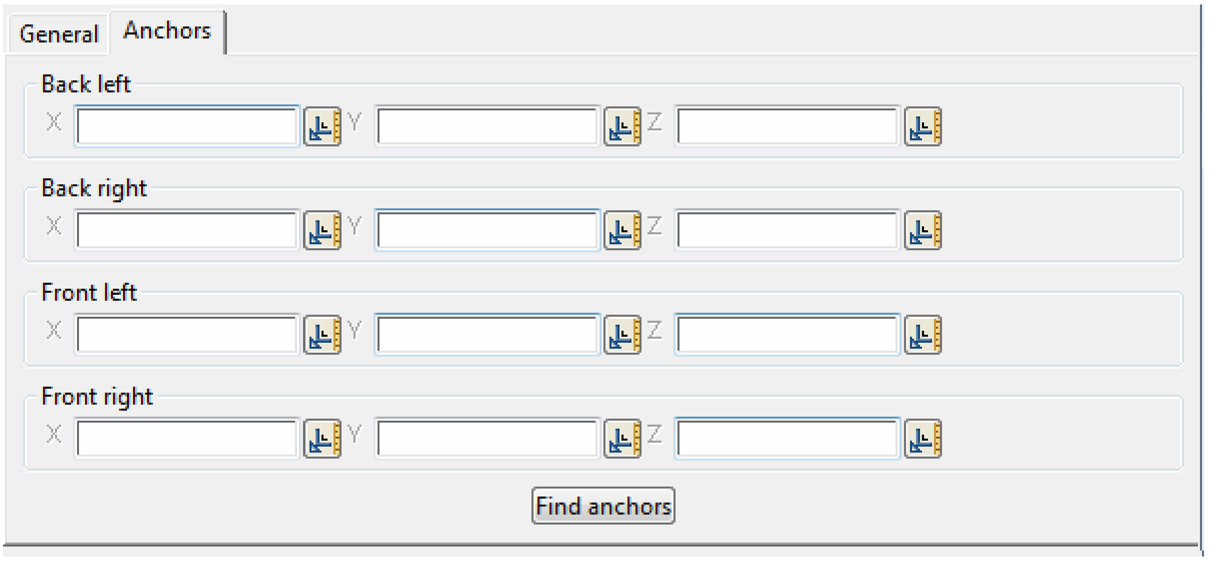
an optional rotation around the y axis

Rotate z

an optional rotation around the z axis

Anchors tab

The **Anchors** tab is used to define anchors, for anchoring a mesh to a tin. To do this, you must define four anchors, at each corner of the mesh.



Back left

defines the co-ordinates of the back left point of the mesh

Back right

defines the co-ordinates of the back right point of the mesh

Front left

defines the co-ordinates of the front left point of the mesh

Front right

defines the co-ordinates of the front right point of the mesh

Find anchors button

*As you may not know the co-ordinates, you may use the **Find Anchors** button to see a list of a number of the lowest points on the mesh. For more information see* 12.13.9.2 Find Anchors

Update Preview button

updates the preview based on the transformation settings

Draw? tick box

whether or not to draw the mesh in the viewer at the bottom

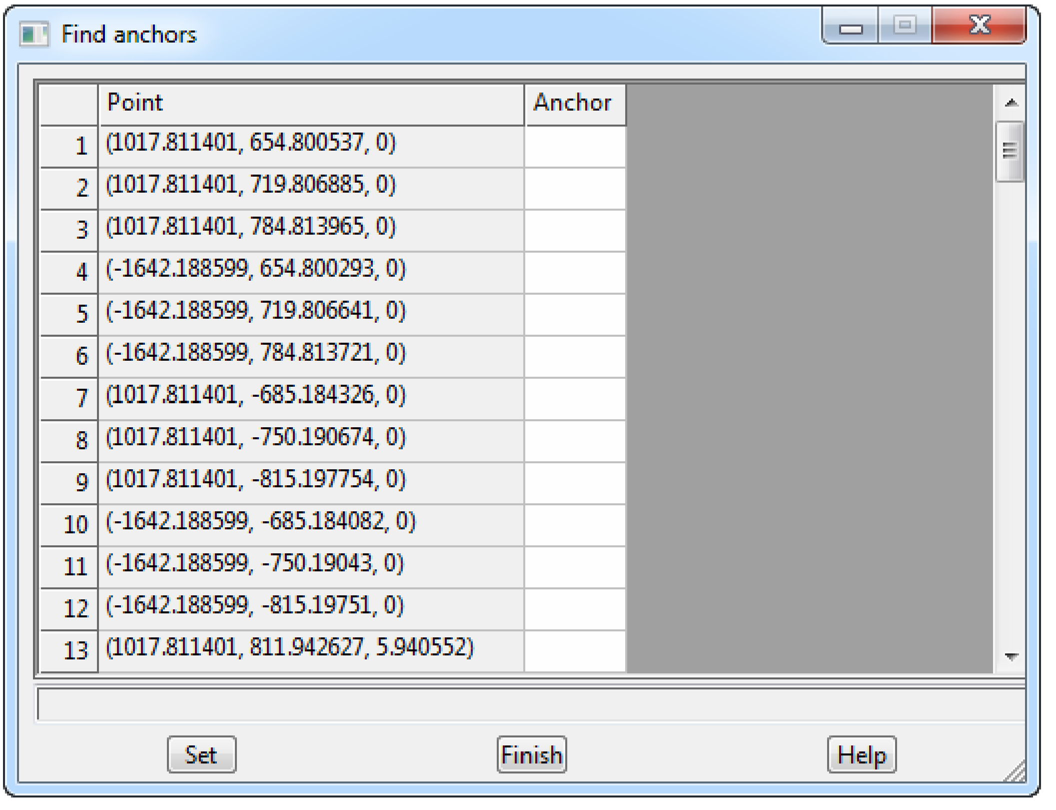
Auto rotate? tick box

whether or not to automatically rotate the mesh

Auto scale? tick box

whether or not to scale the mesh based on supplied transformations

12.13.9.2 Find Anchors



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Point

a known anchor point

Anchor

the anchor to set. You only need to set four anchors.

Set button

sets the anchor points

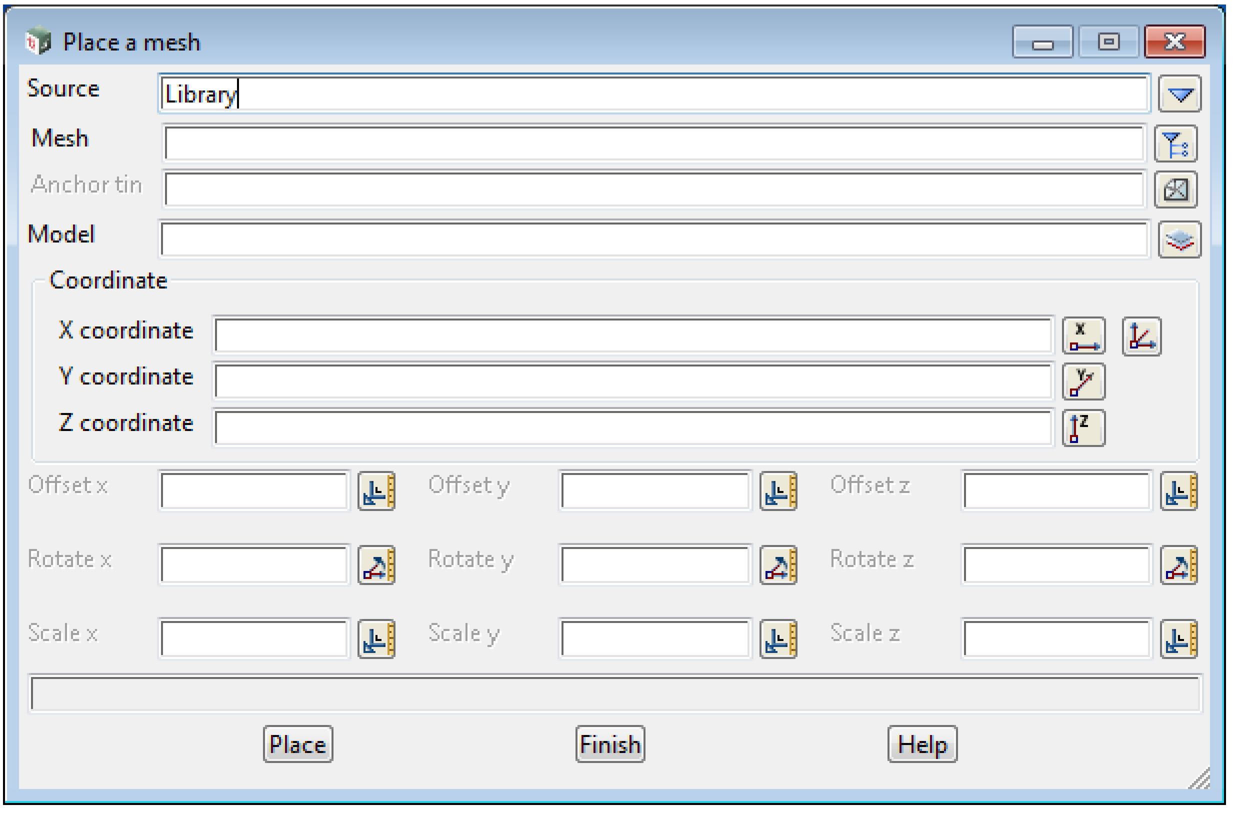
When you select a row, the preview on the 12.13.9.1 Mesh Library will show a symbol to indicate where that co-ordinate is located.

12.13.9.3 Place a Mesh

Position of option on menu: **View =>Visualisation =>Meshes =>Place a mesh**

This panel allows you to place a mesh as the vertex of a super string.

Selecting **Place a mesh** brings up the **Place a mesh** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Source	choice box		Library, File

Whether the mesh comes from the Mesh Library or a separate file

if Sources is Library:

Mesh

the name of the mesh (from the mesh library) to place

Anchor tin

an optional tin to anchor the mesh to

if Source is File:

Mesh file

the file (containing the mesh definition) to use

Model	model box		

the model to place the mesh in

Coordinate

the co-ordinate to place the mesh at

Offset x

an optional offset along the x axis to apply

Offset y

an optional offset along the y axis to apply

Offset z

an optional offset along the z axis to apply

Scale x

an optional scale to apply to the x scale of the object

Scale y

an optional scale to apply to the y scale of the object

Scale z

an optional scale to apply to the z scale of the object

Rotate x

an optional rotation around the x axis

Rotate y

an optional rotation around the y axis

Rotate z

an optional rotation around the z axis

Place button

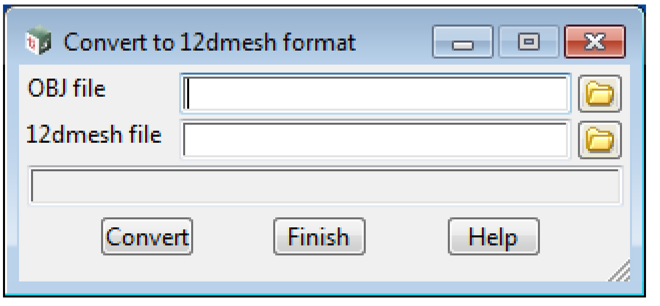
places the mesh

12.13.9.4 Convert to 12dMesh Format

Position of option on menu: **View =>Visualisation =>Meshes =>Convert to 12dmesh**

This option converts most files in OBJ format to the 12dmesh format.

Selecting **Convert to 12dmesh** brings up the **Convert to 12dmesh format** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
OBJ file	file box		available *.obj files

the OBJ file to convert to 12dmesh format.

12dmesh file	file box		available *.12dmesh files

the name of the new .12dmesh file.

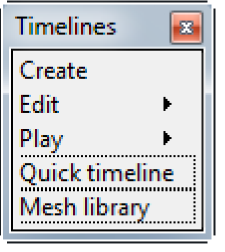
Convert	button		

converts the OBJ file.

12.13.10 Timelines

Position of menu: **View =>Visualisation =>Timelines**

The **Timelines** walk-right menu is



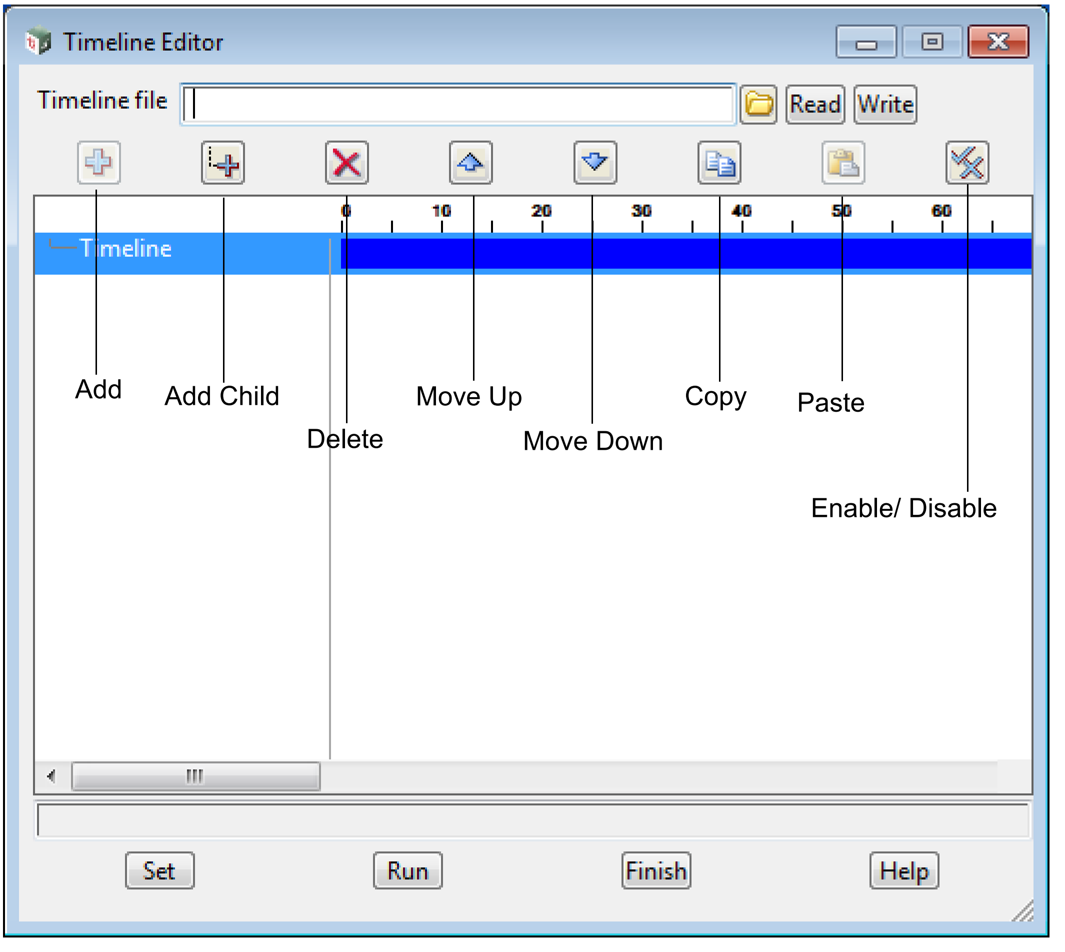
For the option *Create* go to [12.13.10.1 Timeline Editor](#)
Edit [12.13.30.1 Edit a Timeline](#)
Play [12.13.30.2 Play a Timeline](#)
Quick timeline [12.13.34.1 Quick Timeline Create](#)
Mesh library [12.13.9.1 Mesh Library](#)

12.13.10.1 Timeline Editor

Position of menu: **View =>Visualisation =>Timelines => Create**

This is the main **Timeline Editor**. The left side shows the tree of timelines. The right side shows the time, in seconds, during which any given timeline will be active.To edit a timeline, simply double click it.

Selecting **Create** brings up the **Timeline Editor** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Timeline file	file		
the timeline file to edit			
Read	button		
reads the supplied timeline file			
Write	button		
writes the current timeline file			
Add	button		
adds a new timeline at the current level			
Add Child	button		
adds a child to the current timeline			
Delete	button		
deletes the current timeline			
Move Up	button		
moves the current timeline up			
Move Down	button		
moves the current timeline down			

Copy button

copies the current timeline

Paste button

pastes the last copied timeline

Enable / Disable button

enables or disables the current timeline

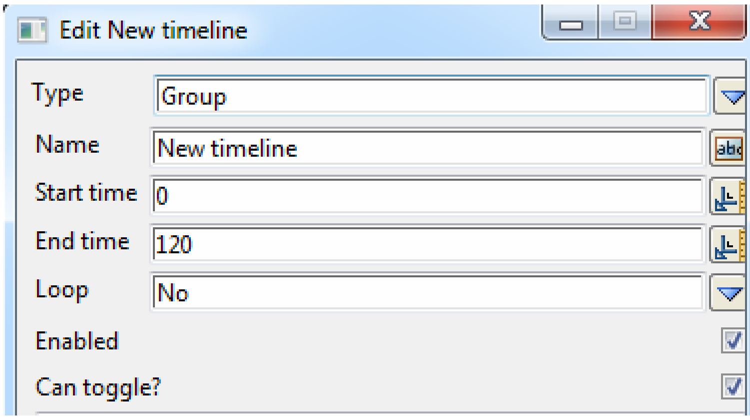
Set button

sets the current timeline details

Run button

runs the current timeline

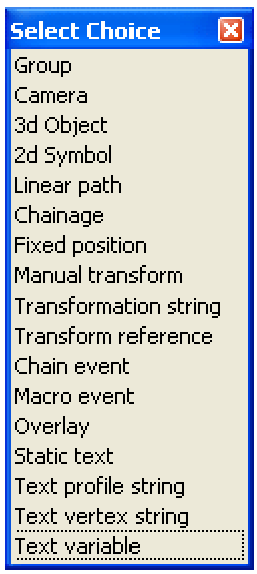
12.13.11 Timeline Common Fields and Buttons



The fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

Type choice box



Sets the type of timeline to edit. For more information on each Type, see 12.13.12 Edit Timeline Types

Name

the name of the timeline

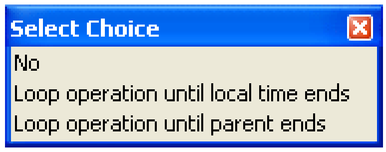
Start Time

the time this timeline should start playing

End Time

the time this timeline should stop playing

Loop choice box



the loop mode to apply for this timeline

No *- no looping*

Loop operation until local time ends *- if the operation of the timeline takes less time than the supplied duration, you may nominate for it to repeat the operation until the end time is reached*

Loop until parent ends *- loop this timeline until the parent time completes*

Enabled tick box

whether or not it is enabled

Can Toggle? tick box

whether or not this timeline can be toggled on or off during playback

Set button

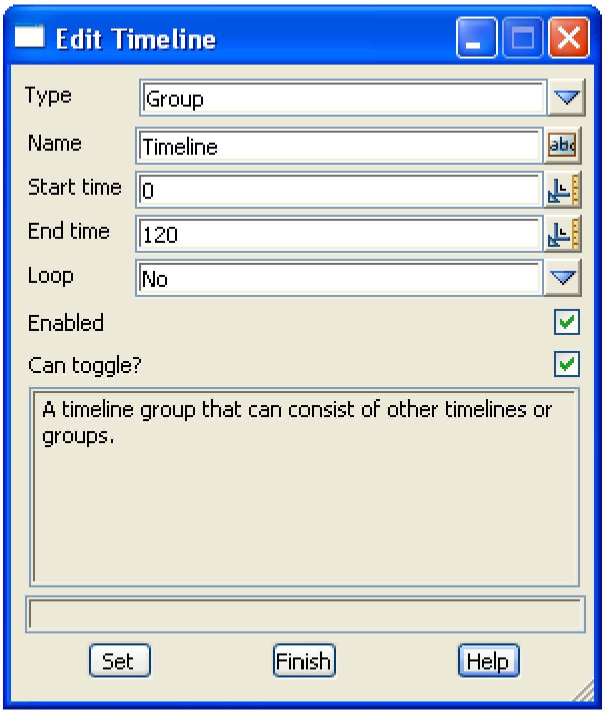
sets the details for the timeline

12.13.12 Edit Timeline Types

For Group, go to	12.13.13 Group Timeline
Camera	12.13.14 Camera Timeline
3d Object	12.13.16 3d Object Timeline
2d Symbol	12.13.17 2d Symbol Timeline
Linear path	12.13.18 Linear Path Timeline
Chainage	12.13.19 Chainage Timeline
Fixed Position	12.13.20 Fixed Position Timeline
Manual transform	12.13.21 Manual Transform Timeline
Transformation string	12.13.22 Transformation String Timeline
Transform reference	12.13.23 Transform Reference Timeline
Chain event	12.13.24 Chain Event Timeline
Macro event	12.13.25 Macro Event Timeline
Overlay	12.13.26 Overlay Timeline
Static text	12.13.27 Static Text Timeline
Text profile string	12.13.28 Text Profile String Timeline
Text vertex string	12.13.29 Text Vertex String Timeline
Text variable	12.13.30 Text Variable Timeline

12.13.13 Group Timeline

The editor for a group timeline, which consists of other timelines.



The fields and buttons used in the panel have the following functions.

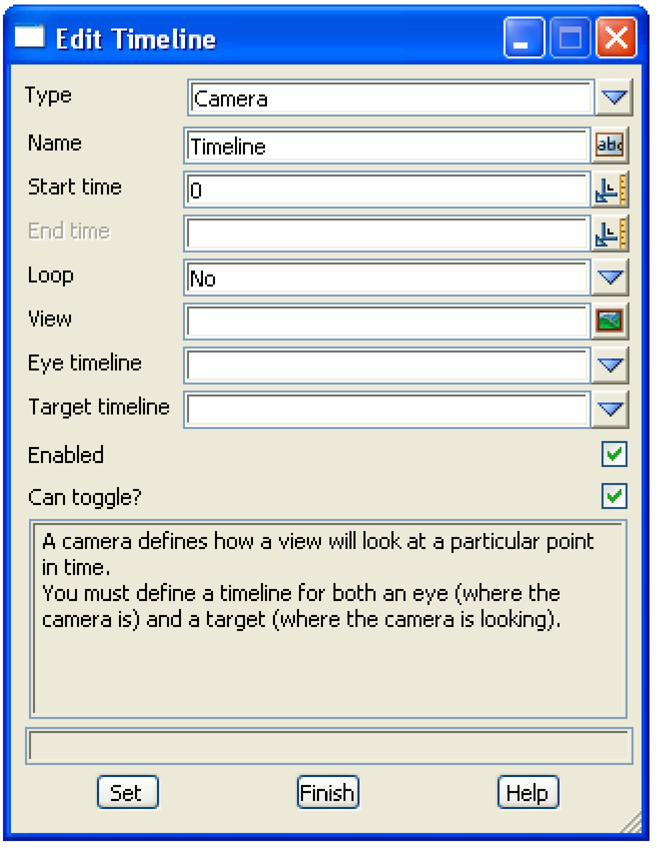
Field Description Type Defaults Pop-Up

There are no fields other than common fields and buttons.For more information on common fields and buttons please see 12.13.11 Timeline Common Fields and Buttons.

12.13.14 Camera Timeline

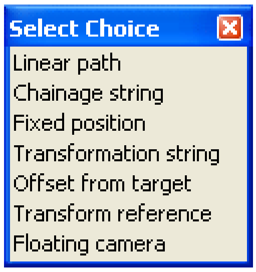
The **Camera Timeline** directs a perspective view where to look at a given time. It is defined by two other timelines - an **eye** and a **target**. There are several different types of timelines available for both **eye** and **target**. When the details are set, two children timelines will be created, one for the **eye** and one for the **target**.

The details for how the **eye** and **target** should behave should be accessed through these two children timelines.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box		
the view to control			
Eye timeline	choice box		
the type of eye timeline			



Linear Path *- See* 12.13.18 Linear Path Timeline

Chainage String *- See* 12.13.19 Chainage Timeline

Fixed Position *- See* 12.13.20 Fixed Position Timeline

Transformation String *- See* 12.13.22 Transformation String Timeline

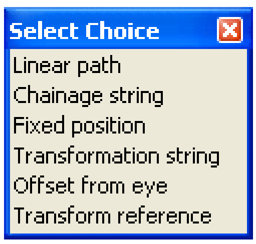
Offset from Target *- Offsets the eye from the target. See* 12.13.21 Manual Transform Timeline

Transform Reference *- See* 12.13.23 Transform Reference Timeline

Floating Camera *- See* 12.13.15 Floating Eye Timeline

Target timeline choice box

the type of target timeline



Linear Path *- See* 12.13.18 Linear Path Timeline

Chainage String *- See* 12.13.19 Chainage Timeline

Fixed Position *- See* 12.13.20 Fixed Position Timeline

Transformation String *- See* 12.13.22 Transformation String Timeline

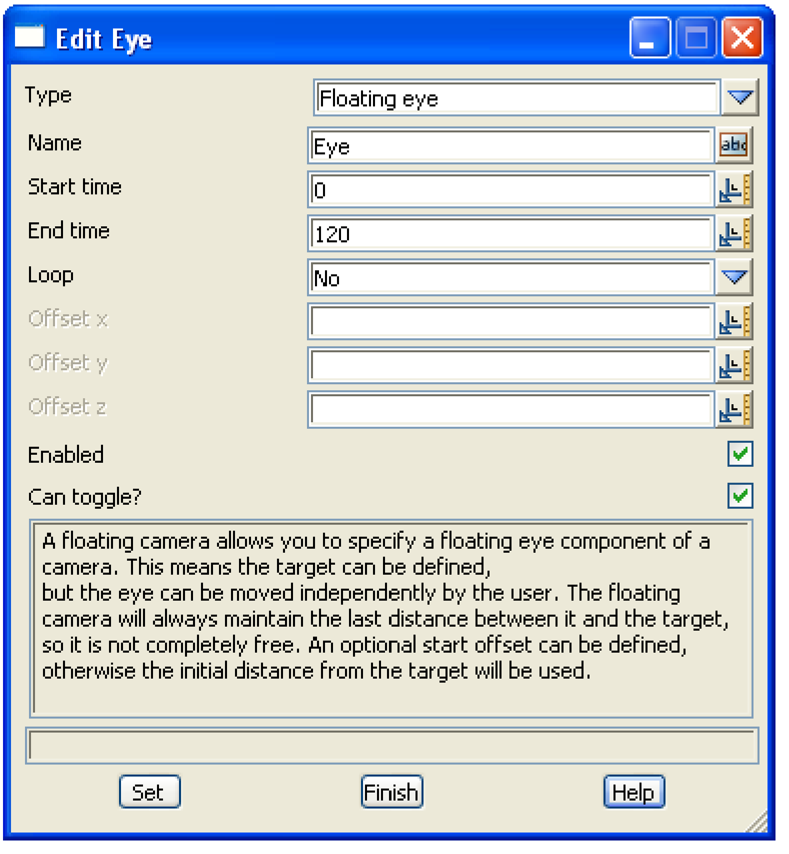
Offset from eye *- Offsets the target from the eye. See* 12.13.21 Manual Transform Timeline

Transform Reference *- See* 12.13.23 Transform Reference Timeline

For all other fields and buttons please see 12.13.11 Timeline Common Fields and Buttons.

12.13.15 Floating Eye Timeline

A **Floating Eye** allows the camera to be fixed on a known target, but the user can retain control over the position of the eye. The **Floating Eye** is a camera type. It can only be accessed by first setting the eye type of a camera to **Floating Camera**.



The fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

Offset x

an optional x offset from the target. If not set, the current x offset from the target will be used.

Offset y

an optional y offset from the target. If not set, the current y offset from the target will be used.

Offset z

an optional z offset from the target. If not set, the current y offset from the target will be used.

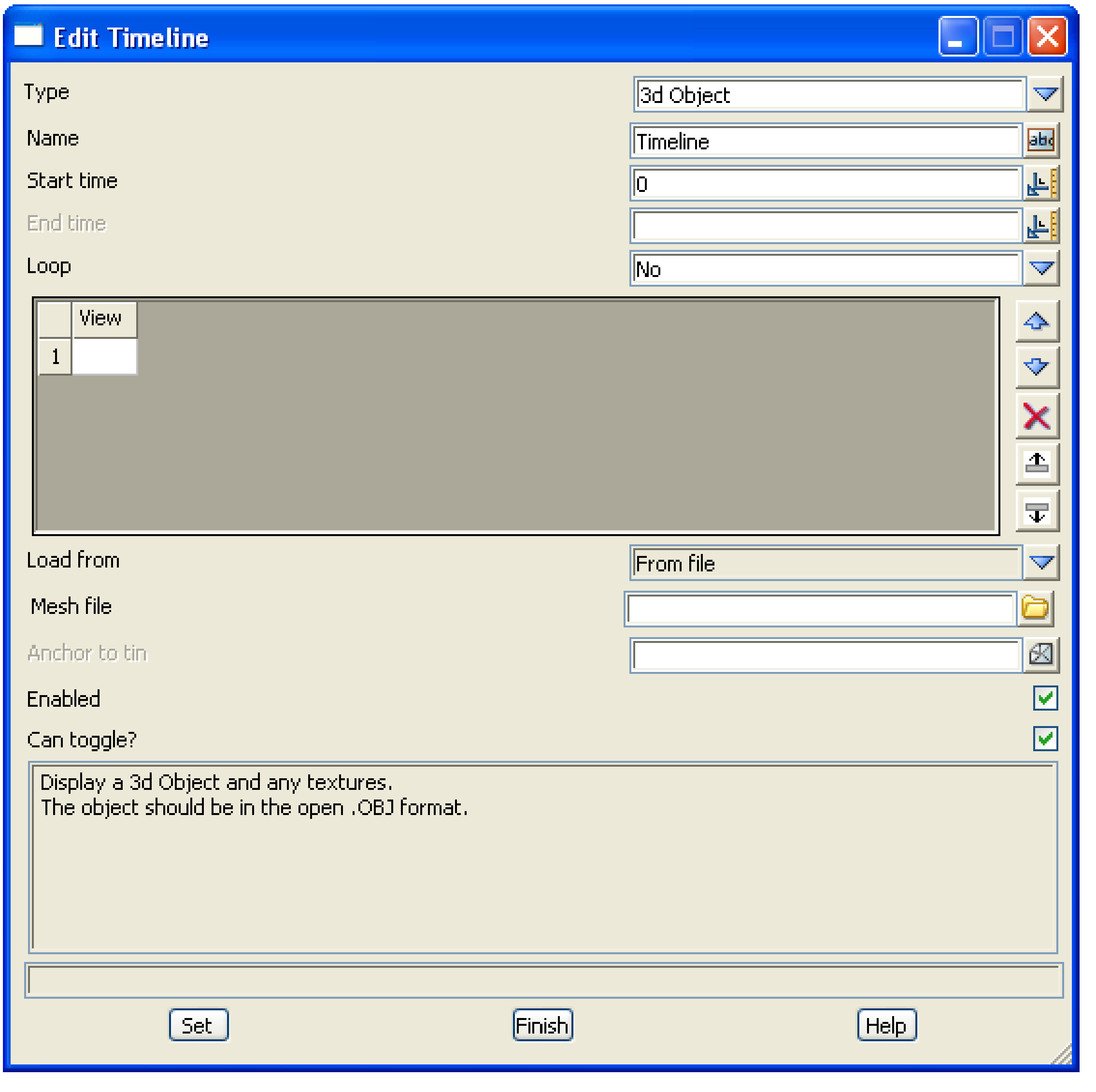
For all other fields and buttons please see 12.13.11 Timeline Common Fields and Buttons.

12.13.16 3d Object Timeline

The **3d Object Timeline** draws a 3d object on any number of views at a given time. A 3d object may include a mesh from another package, in the OBJ (Alias Wavefront) format.

Creating a **3d Object Timeline** will create two sub timelines automatically - the initial transformation and other transformations.

Initial transformations should be used to convert your mesh into world co-ordinates, as meshes may come from a number of packages with arbitrary scale, offsets or rotations.

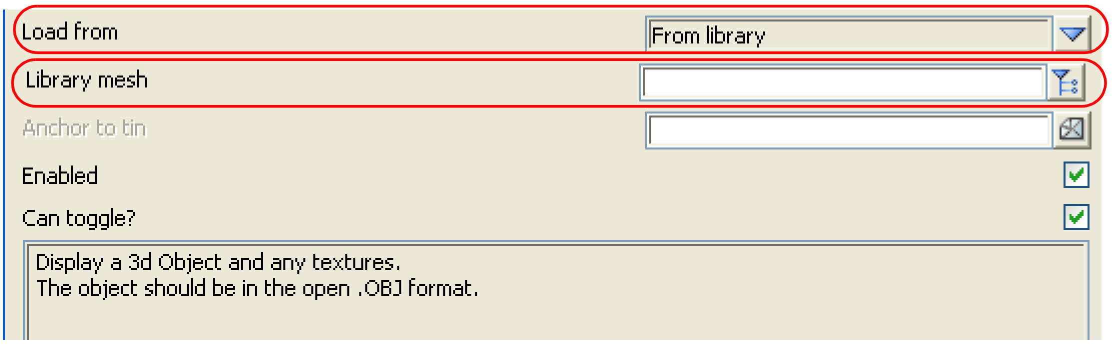


The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View Grid			
the list of OpenGL perspective views to draw on			
Load From	choice box		From library, From file
whether to load from a known file or the library			
Mesh file	file		
when loading from file, the mesh file to use			

Anchor to tin

a tin to anchor the mesh to - useful for vehicles on terrain. Note that this will only work for a library mesh that has anchor points defined.



Library mesh

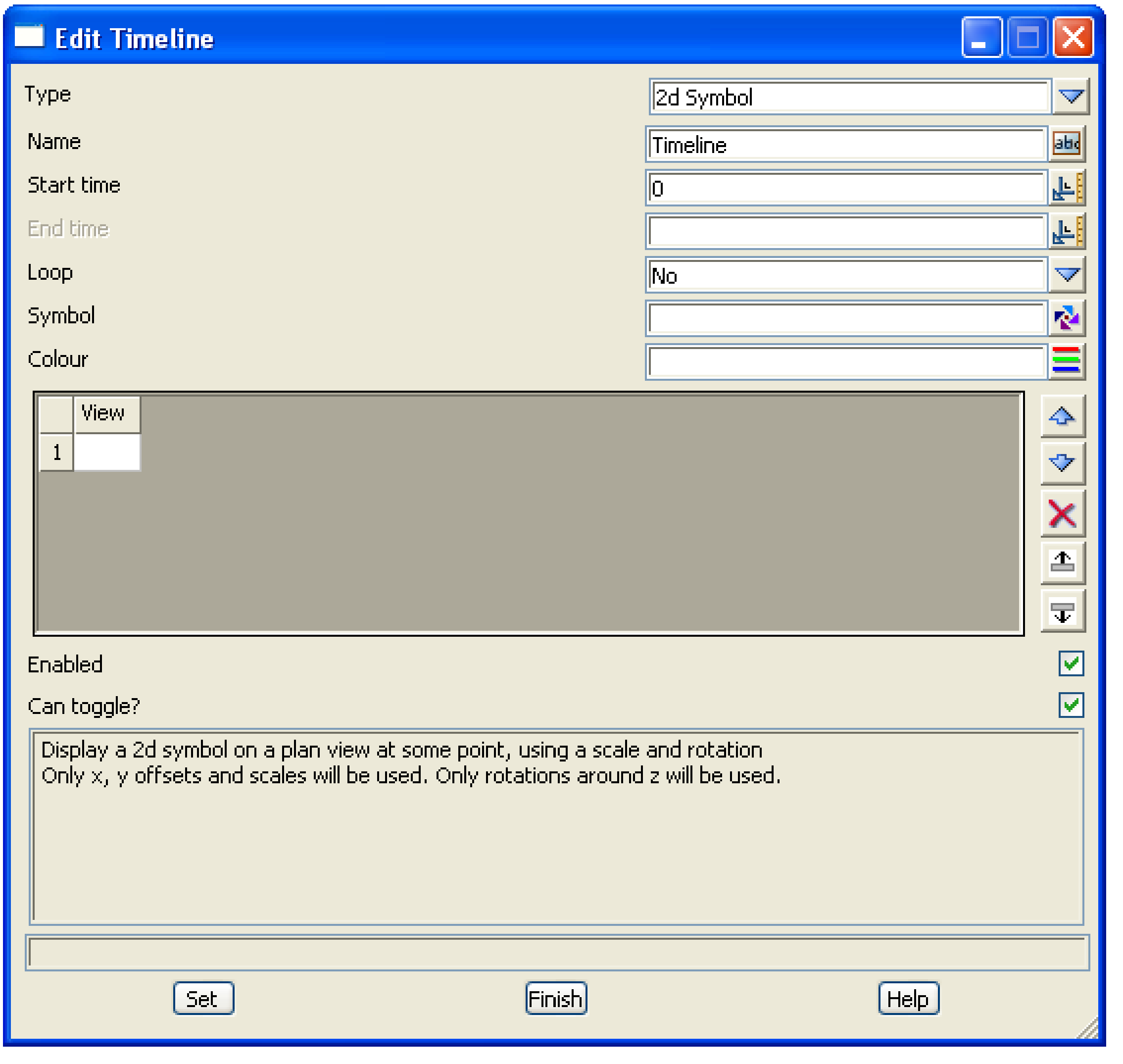
*this field is displayed when **From Library** is selected from the **Load From** choice box.*

when loading from a library, the mesh to use

For all other fields and buttons please see 12.13.11 Timeline Common Fields and Buttons.

12.13.17 2d Symbol Timeline

This allows you to define the settings for drawing a 2d symbol on a number of plan views.



The fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

Symbol symbol box

the symbol to draw

Colour colour box

the colour for the symbol

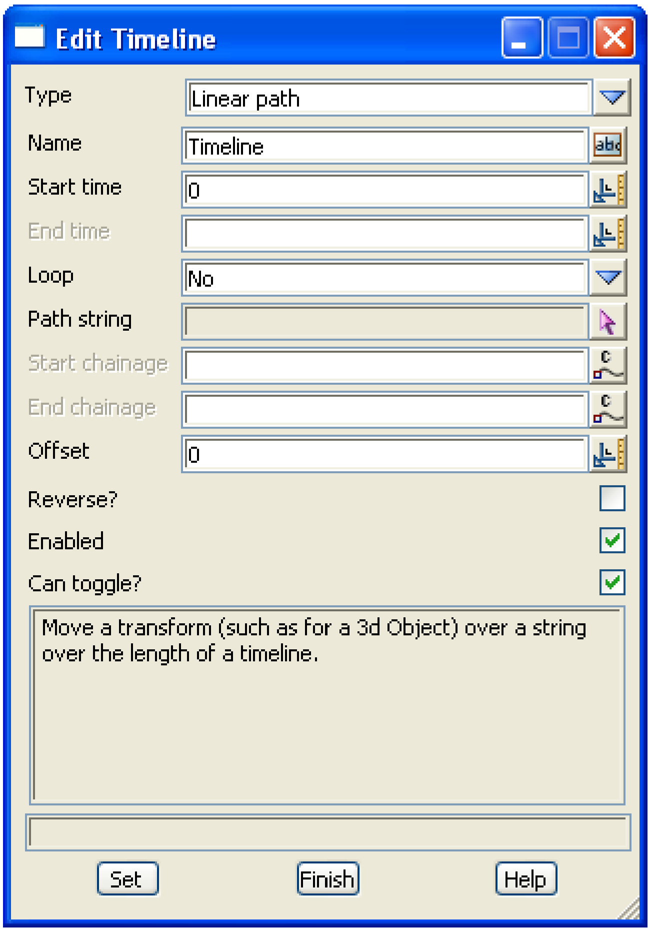
View grid

the list of plan views to draw the symbol

For all other fields and buttons please see 12.13.11 Timeline Common Fields and Buttons.

12.13.18 Linear Path Timeline

This editor defines the settings for a **Linear Path Timeline**. A **Linear Path Timeline** returns the position and direction on a string linearly, to fit the amount of time supplied.



The fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

Path string string select

the string to follow

Start chainage

the optional start chainage

End chainage

the optional end chainage

Offset

an offset from the path string

Reverse? tick box

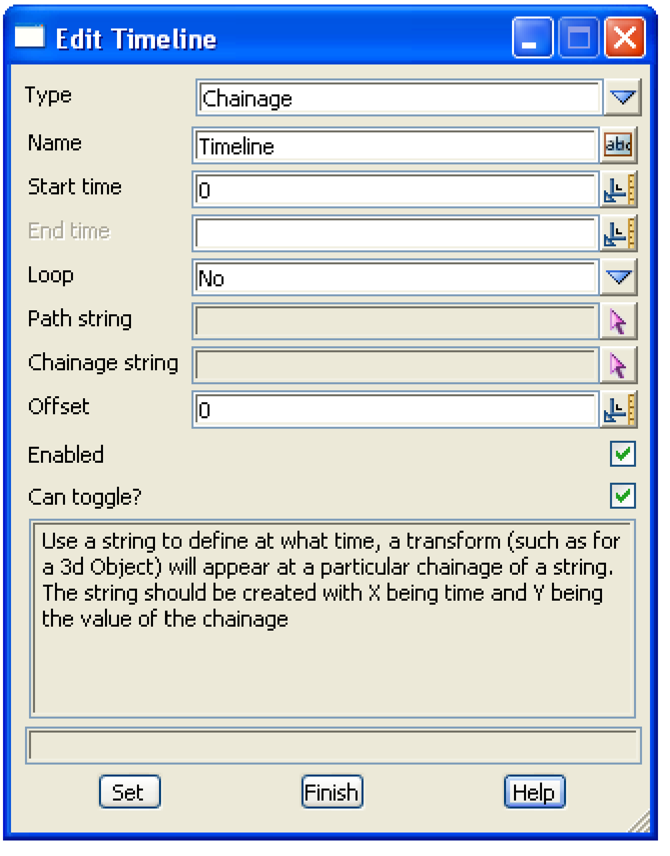
whether or not to reverse the direction (start at end)

For all other fields and buttons please see 12.13.11 Timeline Common Fields and Buttons.

12.13.19 Chainage Timeline

This editor allows you to define the settings for a **Chainage Timeline**. The **Chainage Timeline** follows a string or path by querying another string that defines the chainage to use at a given time. For the **Chainage Timeline**, the chainage string should be defined with X being time and Y being the chainage.

The **Chainage Timeline** will return the position and direction at the current chainage of the path string.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Path string	string select		

the string to follow

Chainage string	string select		

the string that defines what chainage to use at what time

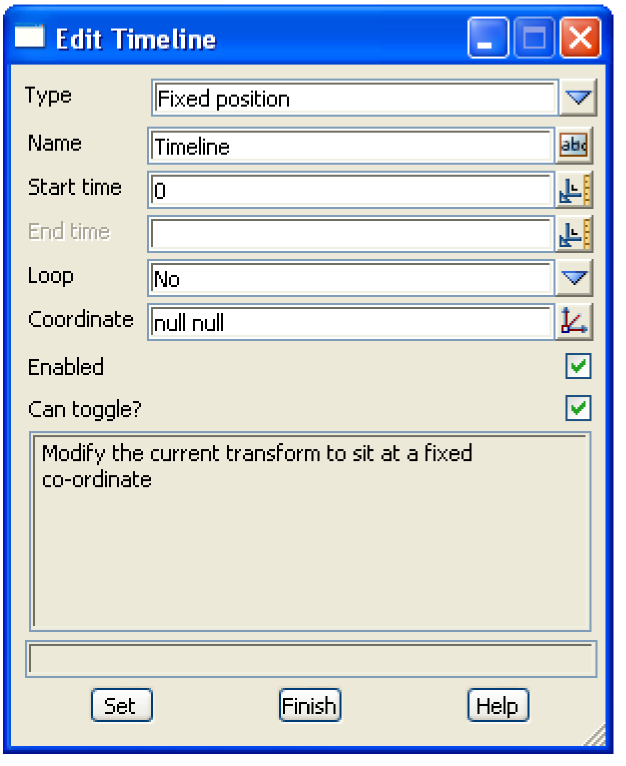
Offset	input		

an offset from the path string

For all other fields and buttons please see 12.13.11 Timeline Common Fields and Buttons.

12.13.20 Fixed Position Timeline

This editor allows you to define the settings for a **Fixed Position Timeline**. A **Fixed Position Timeline** will always return the supplied position.



The fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

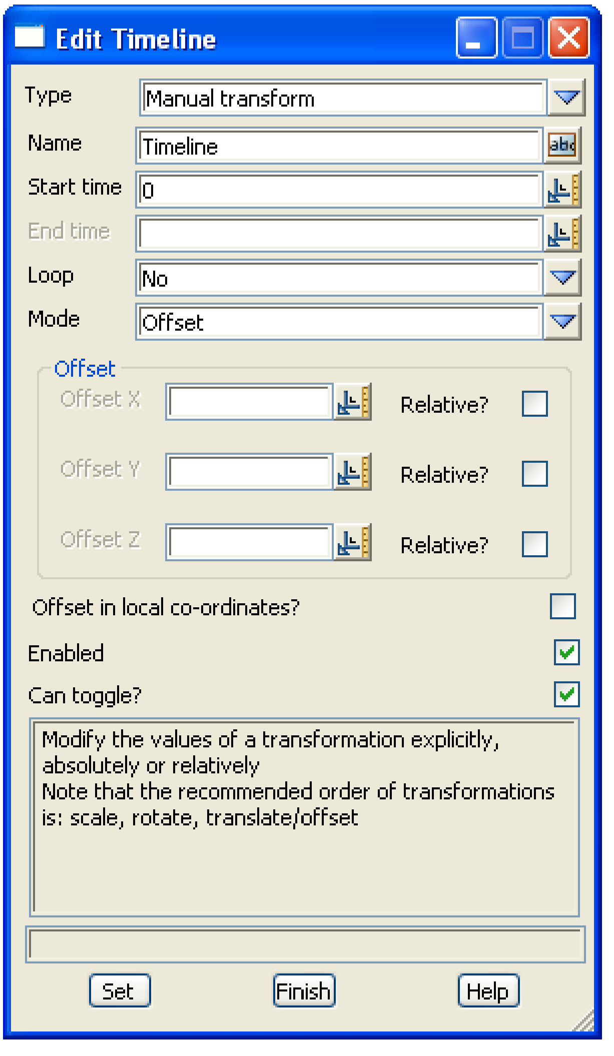
Coordinate

the coordinate of the fixed position

For all other fields and buttons please see 12.13.11 Timeline Common Fields and Buttons.

12.13.21 Manual Transform Timeline

This editor defines the settings for a **Manual Transform Timeline**. A manual transform is either an offset, rotation or scale on the current transformation being applied to a timeline.

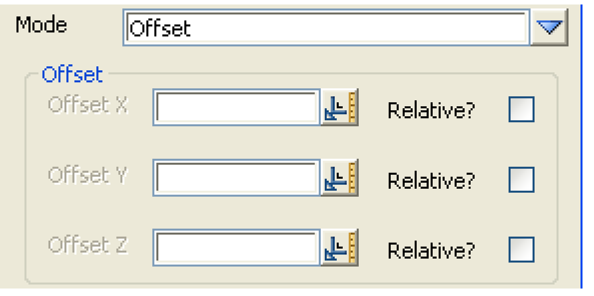


The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Mode	choice box		Offset, Rotate, Scale

the mode of transformation: offset, rotate or scale

Offset defines the parameters for an offset transformation



Offset X

the optional offset to x

Relative? tick box

whether or not Offset X is a relative or absolute offset

Offset Y

the optional offset to y

Relative? tick box

whether or not Offset Y is a relative or absolute offset

Offset Z

the optional offset to z

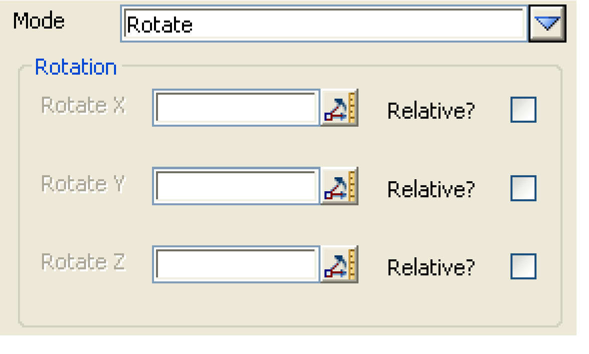
Relative? tick box

whether or not Offset Z is a relative or absolute offset

Offset in local co-ordinates tick box

whether or not the offset should be applied before or after the transformation to local co-ordinates

Rotate defines the parameters for a rotation transformation



Rotate X

the optional rotation around the x axis

Relative? tick box

whether or not the X rotation is relative or absolute

Rotate Y

the optional rotation around the y axis

Relative? tick box

whether or not the Y rotation is relative or absolute

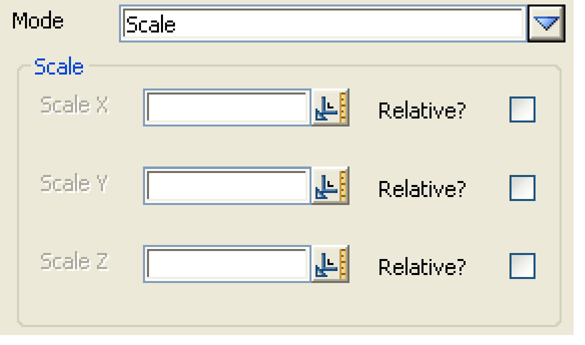
Rotate Z

the optional rotation around the z axis

Relative? tick box

whether or not the Z rotation is relative or absolute

Scale defines the parameters for a scaling transformation



Scale X

the optional x scale

Relative? tick box

whether or not the x scale is relative

Scale Y

the optional y scale

Relative? tick box

whether or not the y scale is relative

Scale Z

the optional z scale

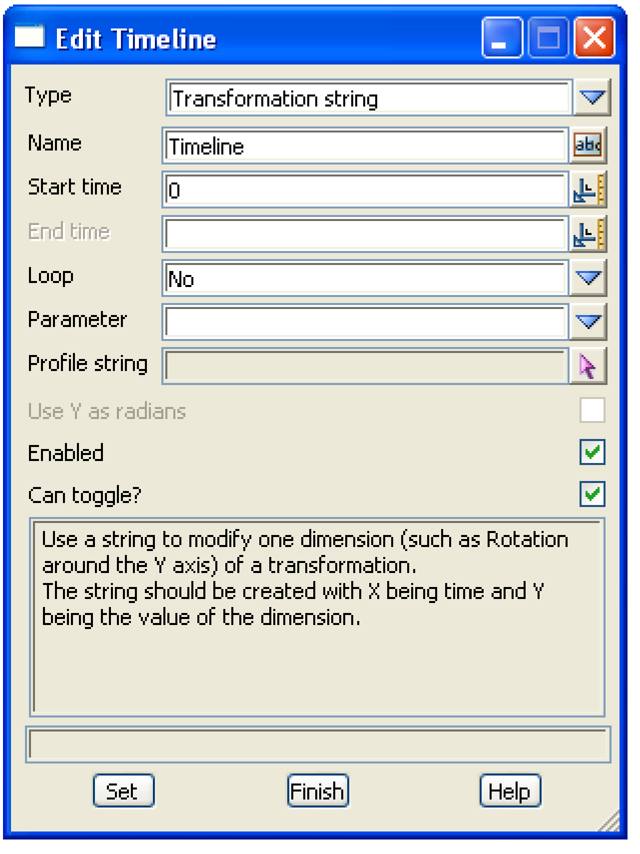
Relative? tick box

whether or not the z scale is relative

For all other fields and buttons please see 12.13.11 Timeline Common Fields and Buttons.

12.13.22 Transformation String Timeline

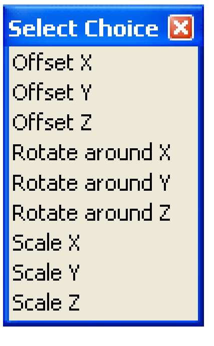
The **Transformation String Timeline** modifies one parameter of a transformation, based on a profile string. The profile string should be created such that x is time and y is the value of the parameter.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Parameter	choice box		

the parameter to update.



Options are: Offset X, Offset Y, Offset Z, Rotate around X, Rotate around Y, Rotate around Z, Scale X, Scale Y, Scale Z

Profile string	string select		

the profile string to use to modify the parameter

Use Y as Radians tick box

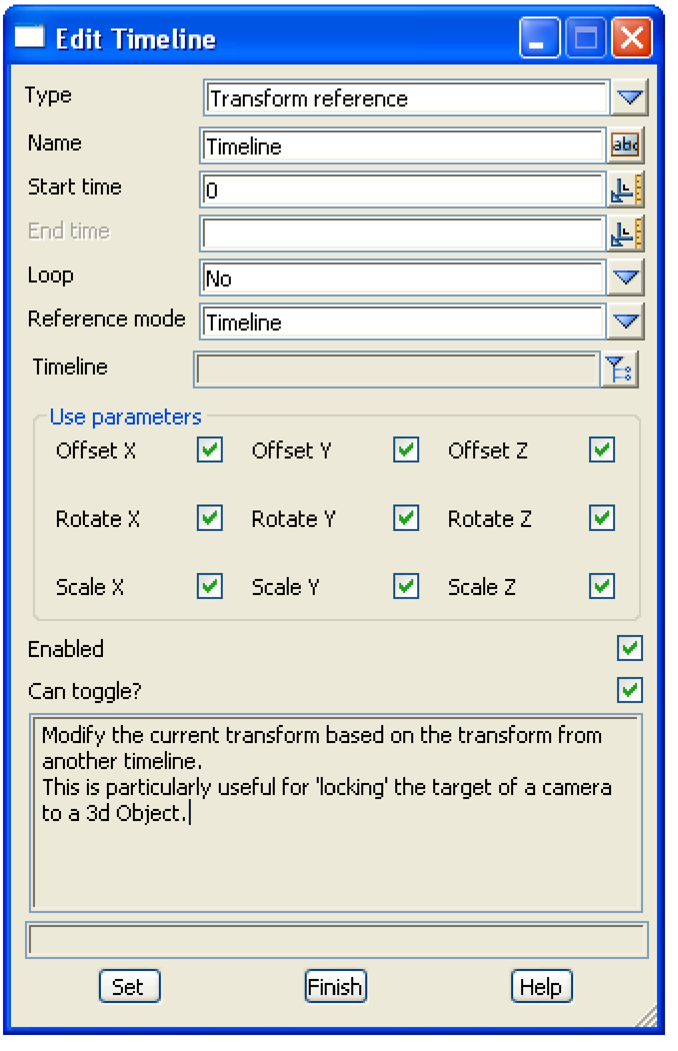
enabled when the parameter is a rotation parameter.

if ticked, the value of Y will be considered to be radians rather than degrees.

For all other fields and buttons please see 12.13.11 Timeline Common Fields and Buttons.

12.13.23 Transform Reference Timeline

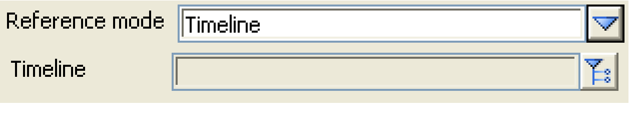
This editor defines the settings for a **Transform Reference Timeline**. A **Transform Reference Timeline** uses the transformation as supplied by another timeline. It is useful, for example, to make the target of a camera follow a vehicle, as it will return all the transformations from the reference timeline.



The fields and buttons used in the panel have the following functions.

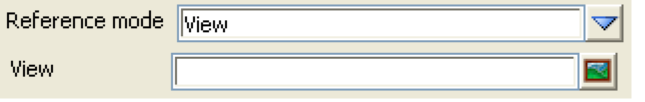
Field Description	Type	Defaults	Pop-Up
Reference mode	choice box		Timeline, View

whether to reference another timeline or a view



Timeline

*the timeline to reference, if **Reference mode** is set to **Timeline***



View

*the view to reference, if **Reference mode** is set to **View***

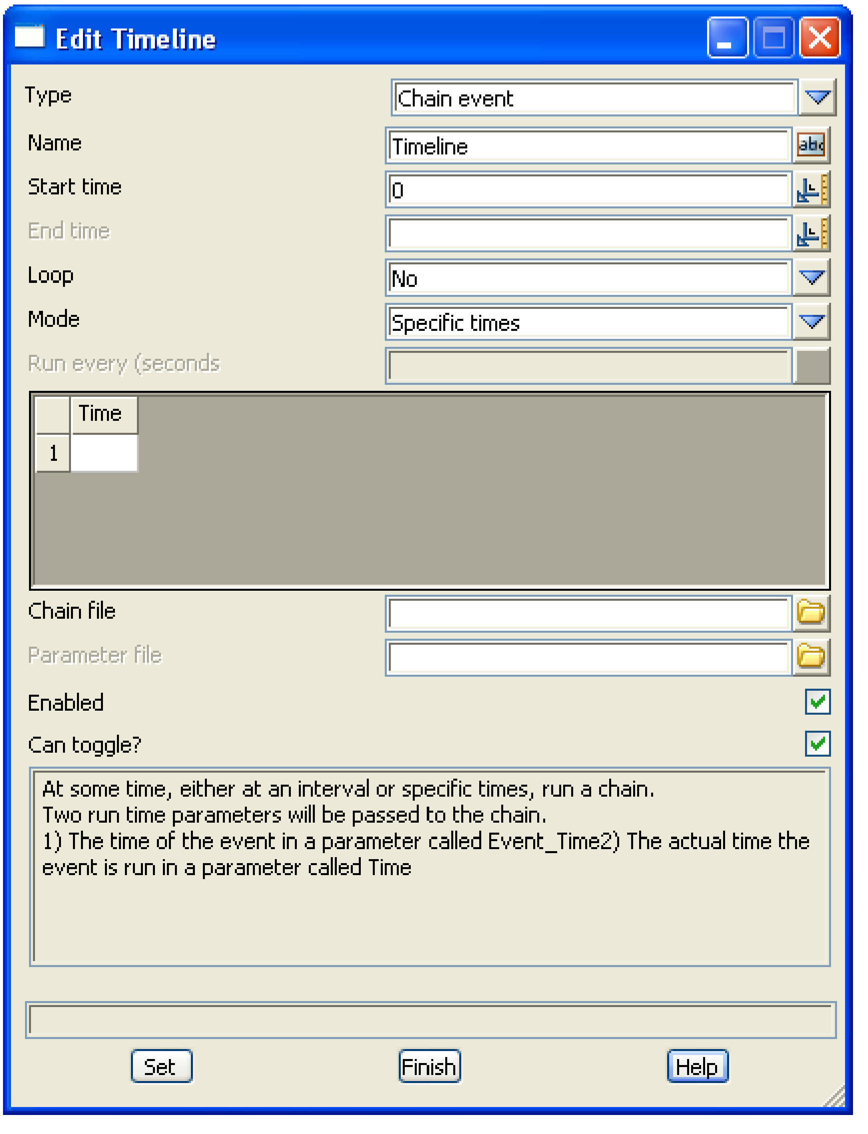
Use Parameters

defines which of the parameters to use from the transformation

For all other fields and buttons please see 12.13.11 Timeline Common Fields and Buttons.

12.13.24 Chain Event Timeline

This allows you to define the settings for a **Chain Event Timeline.** A **Chain Event Timeline** runs a chain at discrete intervals or specific times.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Mode	choice box		Specific times, On interval

the mode - either specific times, or on an interval

Run every (seconds)

*if mode is set to **On Interval**, the interval, in seconds, between runs of the chain*

Time Grid

*if mode is set to **Specific times**, the list of times, in seconds, the chain should be run*

Chain file	file		

the chain file to run

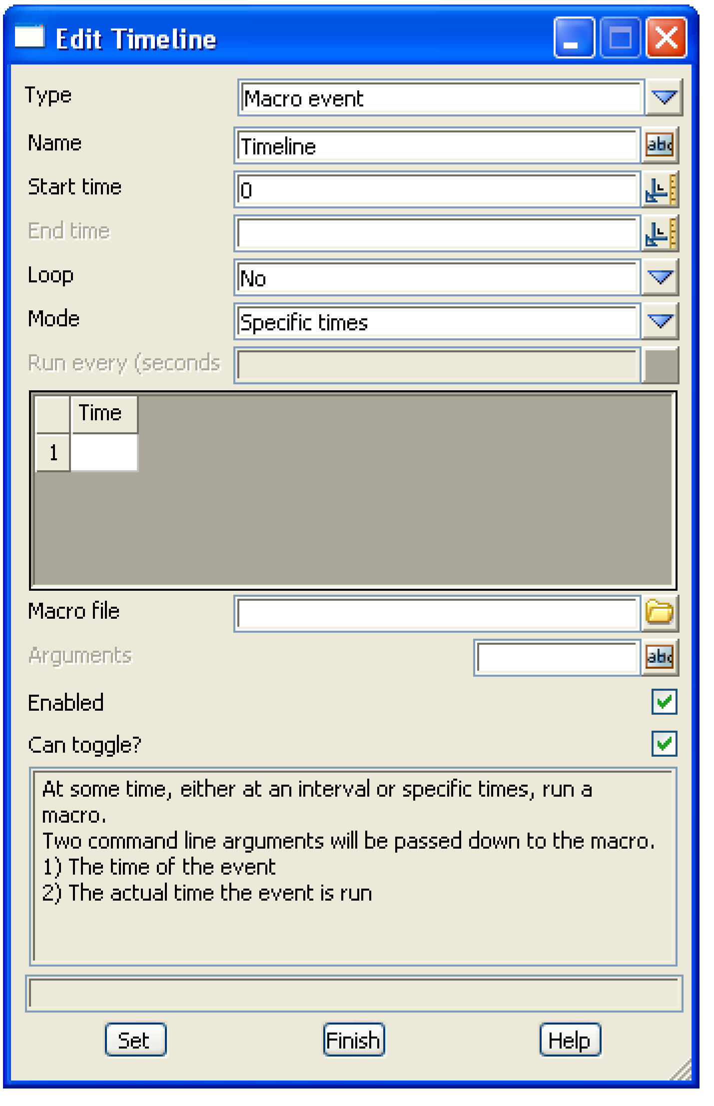
Parameter file	file		

the optional parameter file to pass to the chain

For all other fields and buttons please see 12.13.11 Timeline Common Fields and Buttons.

12.13.25 Macro Event Timeline

This allows you to define the settings for a **Macro Event Timeline**. A **Macro Event Timeline** runs a macro at discrete intervals or specific times.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Mode	choice box		Specific times, On interval

the mode - either specific times, or on an interval

Run every (seconds)

*if mode is set to **On interval**, the interval, in seconds, between runs of the macro*

Time Grid

*if mode is set to **Specific times**, the list of times, in seconds, the macro should be run*

Macro file file

the macro file to run

Arguments

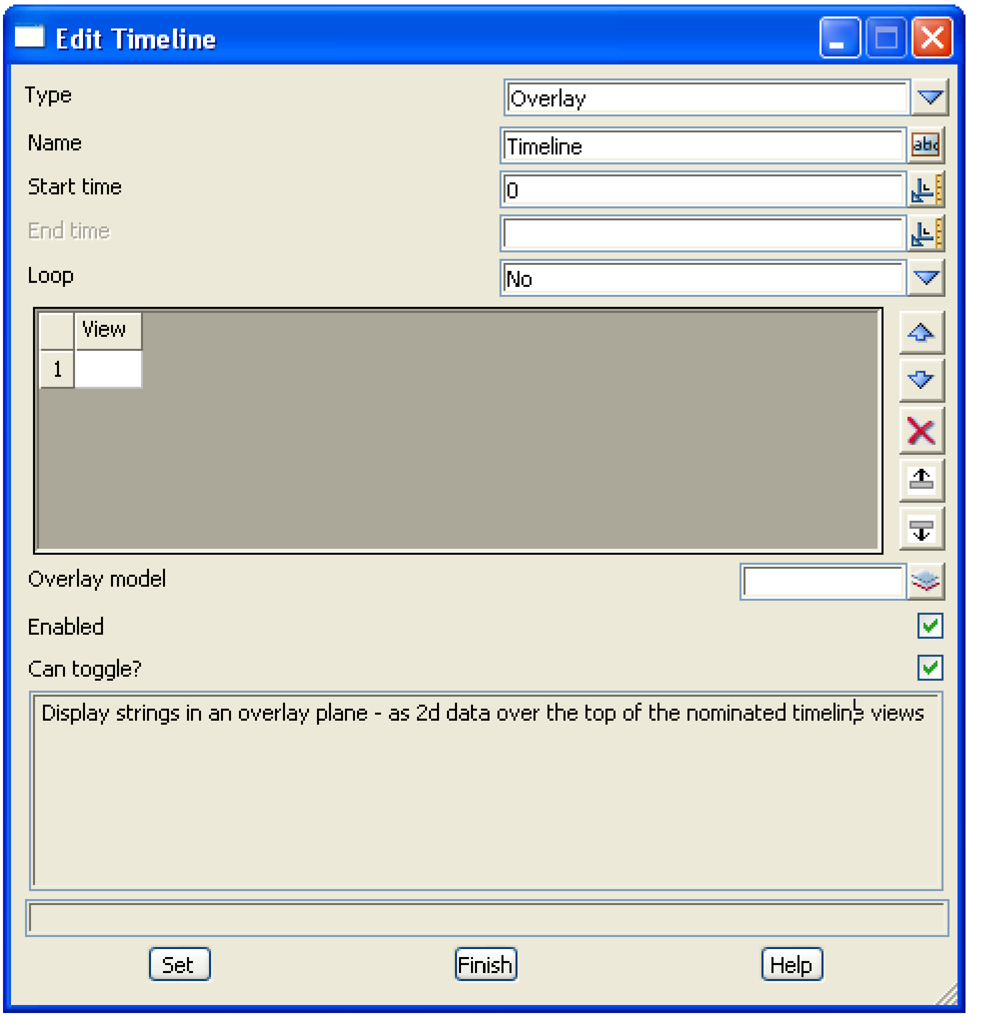
the optional arguments to pass to the macro

For all other fields and buttons please see 12.13.11 Timeline Common Fields and Buttons.

12.13.26 Overlay Timeline

This editor defines the setting for an **Overlay Timeline**. An **Overlay Timeline** draws a model as an overlay, or on a 2 dimensional plane, on top of a number of perspective views. Any children timelines will also appear in the overlay plane.

An **Overlay Timeline** is quite useful for showing diagnostic or other information.



The fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

View Grid

the set of views to draw the overlay plane on

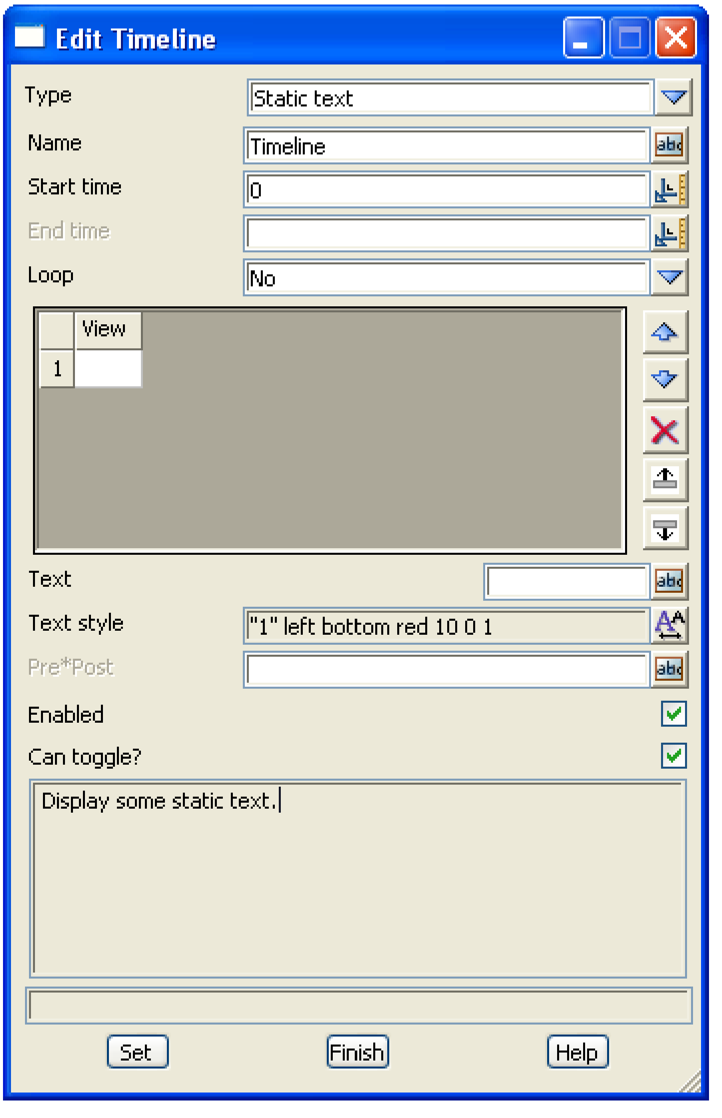
Overlay model model box

the model to draw as an overlay plane

For all other fields and buttons please see 12.13.11 Timeline Common Fields and Buttons.

12.13.27 Static Text Timeline

This editor edits the settings for a **Static Text Timeline**. A **Static Text Timeline Draws** user defined text on a perspective view.



The fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

View Grid

the list of perspective views to draw on

Text

the text to draw

Text style

the style of the text

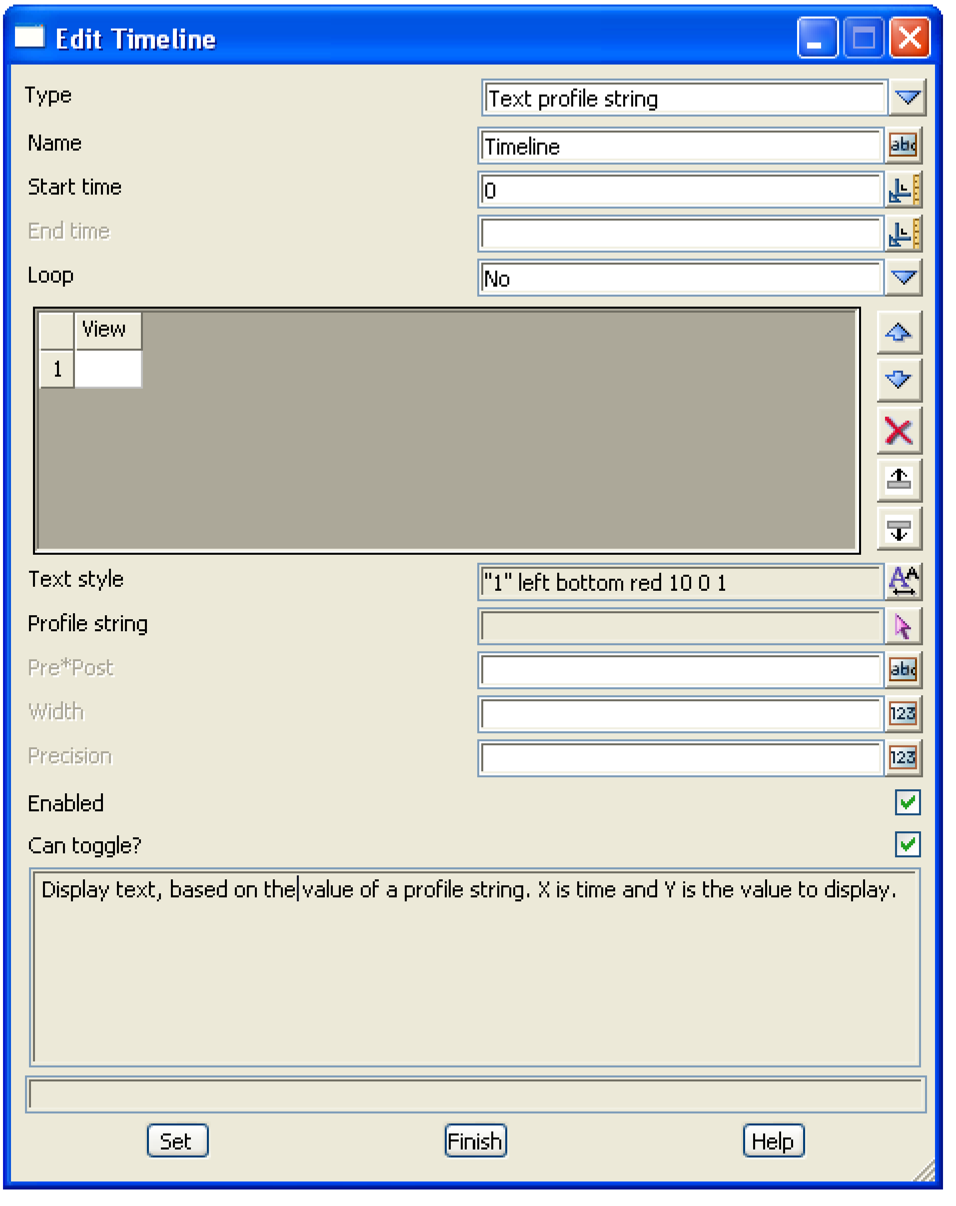
Pre*Post

*an optional pre*post to apply to the text*

For all other fields and buttons please see 12.13.11 Timeline Common Fields and Buttons.

12.13.28 Text Profile String Timeline

This edits the settings for a **Text Profile String Timeline**. A **Text Profile String Timeline** displays text as defined by a profile string, at each time step in the timeline. The profile string should be created with X being time and Y being the value to display.



The fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

View grid

the list of OpenGL views to draw text on

Text style

the style of the text

Profile string string select

the profile string to use

Pre*Post

*an optional Pre*Post to be applied to the text*

Width

an optional width of the text

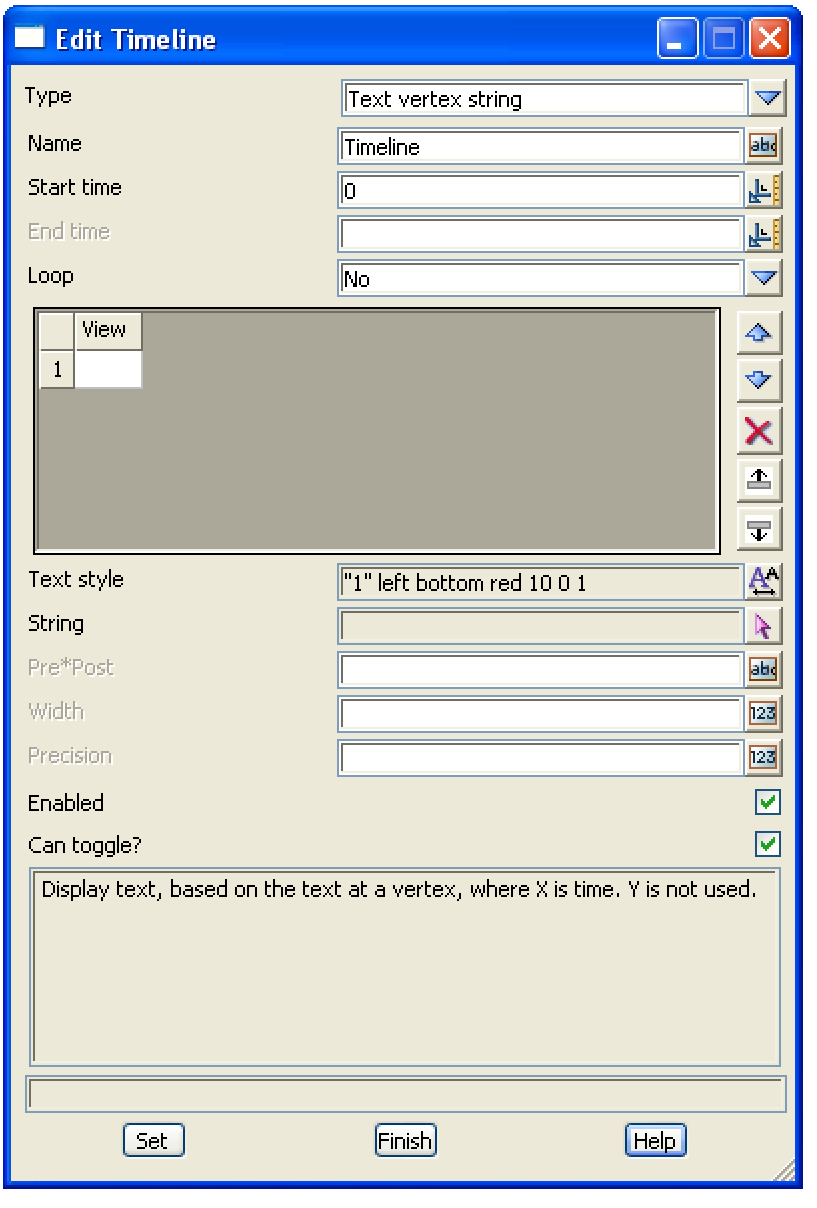
Precision

an optional precision for the text - used when displaying variables of type Real

For all other fields and buttons please see 12.13.11 Timeline Common Fields and Buttons.

12.13.29 Text Vertex String Timeline

This editor allows you to define the settings for a **Text Vertex String Timeline**. A T**ext Vertex String Timeline** draws the text on the vertex of a supplied profile string at the given time. The string should be created such that X is time.



The fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

View Grid

the list of OpenGL views to draw the text on

String string select

the string to draw

Text style

the style of the text

Pre*Post

*an optional Pre*Post to be applied to the text*

Width

an optional width of the text

Precision

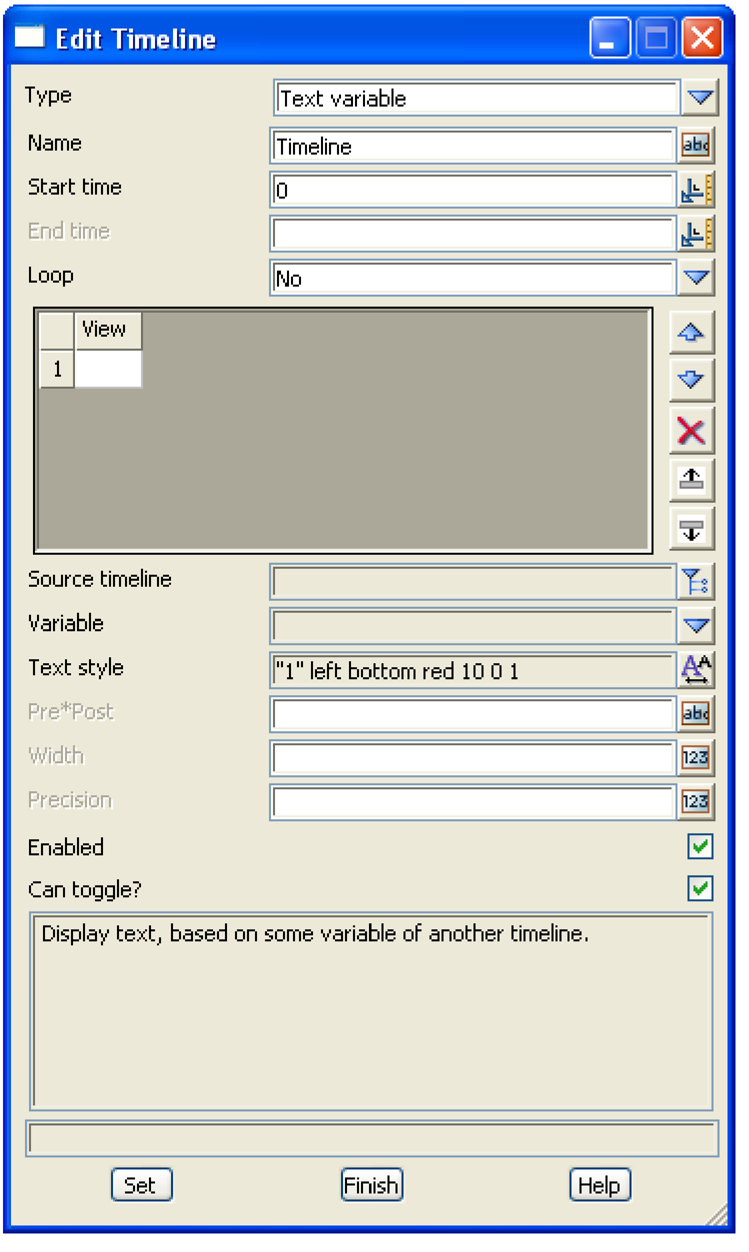
an optional precision for the text - used when displaying variables of type Real

For all other fields and buttons please see 12.13.11 Timeline Common Fields and Buttons.

12.13.30 Text Variable Timeline

This edits the settings for a **Text Variable Timeline**. A **Text Variable Timeline** displays text on a number of OpenGL perspective views, by specifying a variable or property of another timeline.

Each timeline publishes a number of variables or properties that can be used - such as start time, end time or other context specific items.



The fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

View grid

the list of OpenGL views to draw the text on

Source timeline

the timeline whose variable you wish to draw as text

Variable choice box

the variable to be drawn

Text style

the style of the text

Pre*Post

*an optional Pre*Post to be applied to the text*

Width

an optional width of the text

Precision

an optional precision for the text - used when displaying variables of type Real

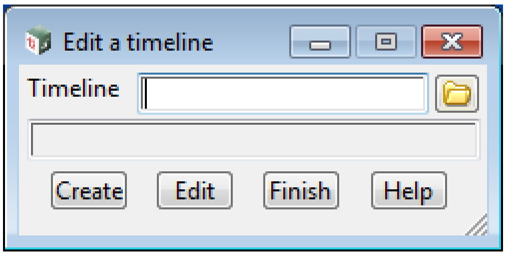
For all other fields and buttons please see 12.13.11 Timeline Common Fields and Buttons.

12.13.30.1 Edit a Timeline

Position of option on menu: **View =>Visualisation =>Timelines => Edit**

This panel creates a new timeline file or edits an existing one.

Selecting **Edit** brings up the Edit a timeline panel.



The fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

Timeline file

the file to edit or create

Create button

creates the new file

Edit button

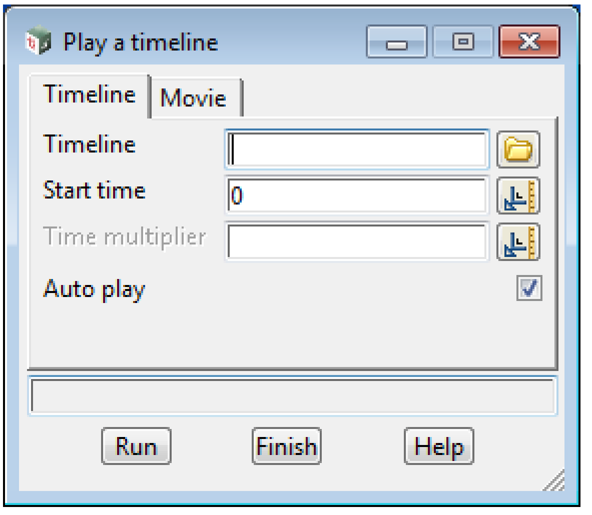
edits an existing timeline file

12.13.30.2 Play a Timeline

Position of option on menu: **View =>Visualisation =>Timelines => Play**

This panel plays a timeline file.

Selecting **Play** brings up the **Play a timeline** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Timeline tab

Timeline file

the timeline file to run

Start time

the time at which to start playing the file

Time multiplier

an optional multiplier (for example, setting this to 2 would run the timeline at 2x, meaning each second would take half a second to play)

Auto play tick box

whether or not to start playing the timeline when it is run

Movie tab

Create movie? tick box

if ticked, a movie is generated

Movie file file box available *.avi files

name of the movie file

Record whole application tick box

if ticked, the whole application will be recorded, rather than just a selected view

View view box

(only if Record whole application *is not ticked) the view of which to record the movie*

Frames per second number box 10

the number of frames per second

Buttons at bottom

Run button

runs the timeline

12.13.31 Timeline Playback Control Bar

The timeline playback control bar appears when a timeline is running. It offers playback controls while the timeline runs.

The fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

Restart

Restarts the timeline

Step back

Steps back by the nominated time step (the input box on the right)

Play / Pause /

Plays or pauses the playback of the timeline

Stop

stops running the timeline

Step forward

Steps forward by the nominated time step (the input box on the right)

Time multiplier

*Sets the time multiplier. For more information on the **Change Timeline Multiplier** panel, please see* 12.13.32 Change Timeline Multiplier

Interactive camera

Defines interactive cameras at runtime

*For more information on the **Timeline Camera** panel, please see* 12.13.34 Timeline Camera

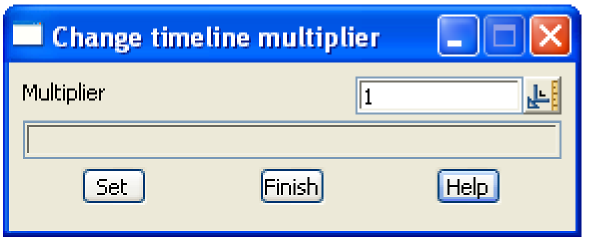
Toggle

*Toggles timelines enabled state on or off. For more information on the **Enable/ Disable Timelines** panel, please see* 12.13.33 Enable / Disable Timelines

12.13.32 Change Timeline Multiplier

This sets the current time multiplier for the timeline. For example, setting this to 2 would run the timeline at 2x, meaning each second would take half a second to play.

Clicking  on the **Timeline Playback Control Bar** brings up the **Change timeline multiplier** panel**.**



The fields and buttons used in the panel have the following functions.

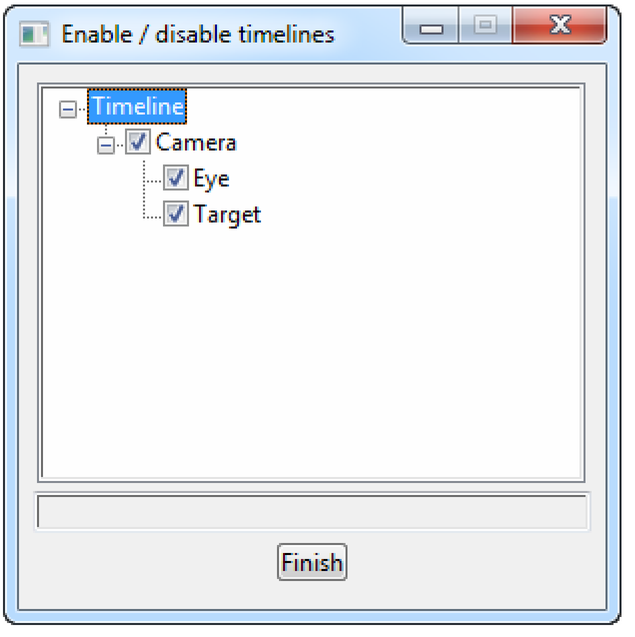
Field Description	Type	Defaults	Pop-Up
Multiplier	input		
the multiplier to apply			
Set	button		
sets the multiplier			

12.13.33 Enable / Disable Timelines

This panel provides a list of the timelines currently running and allows you to enable or disable them quickly. They can only be enabled or disabled if they are set to **Can toggle?**. Note that if you disable a parent, all child timelines will also be disabled.The **Can toggle?** tick box is found on the editor for each timeline.

To disable, simply tick off the required timeline and vice versa to enable them again.

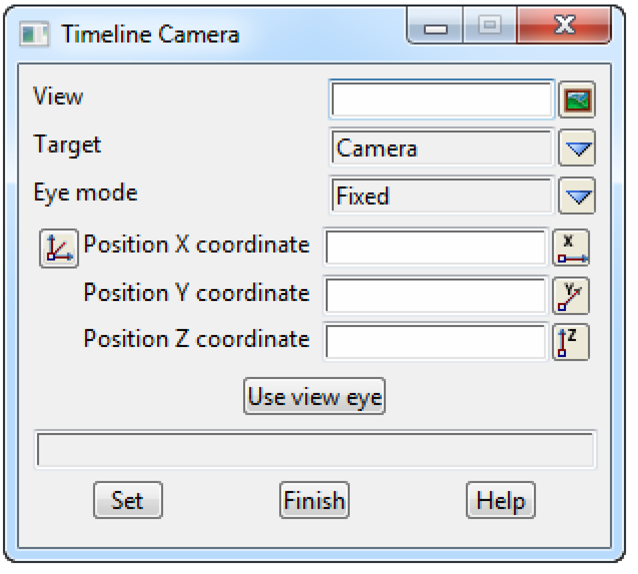
Clicking  on the **Timeline Playback Control Bar** brings up the **Enable/ disable timelines** panel.



12.13.34 Timeline Camera

This panel allows you to define cameras for a supplied view during run time.

Clicking on the **Timeline Playback Control Bar** brings up the **Timeline Camera** panel.



The fields and buttons used in the panel have the following functions.

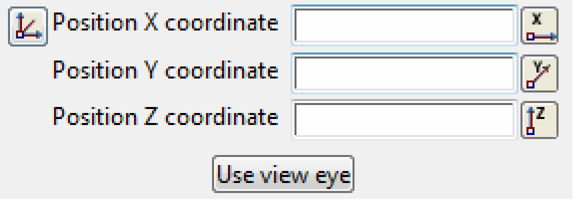
Field Description	Type	Defaults	Pop-Up
View	view box		
Target	choice box		
Eye Mode	choice box		Fixed, Offset, Floating

View — *the perspective view to control*

Target — *a target timeline to reference*

Eye Mode — *the mode for the eye of the camera, either fixed, offset or floating*

Fixed



Position X coordinate

the x coordinate of the fixed eye

Position Y coordinate

the y coordinate of the fixed eye

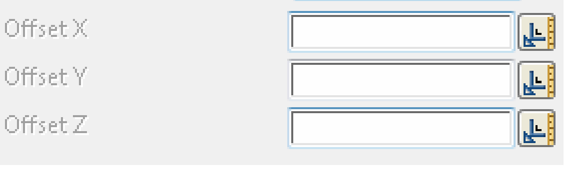
Position Z coordinate

the z coordinate of the fixed eye

Use view eye *button*

loads the eye settings from the supplied view

Offset



Offset X

the x offset of the eye from the target

Offset Y

the y offset of the eye from the target

Offset Z

the z offset of the eye from the target

Floating

Floating mode sets the timeline camera into a floating mode. This means the camera will be set to a fixed target, but the user will retain control over the camera. When started, the camera will start at the current offset from the target.

Set button

Creates and sets the details for the camera

12.13.34.1 Quick Timeline Create

Position of option on menu: **View =>Visualisation =>Timelines => Quick timeline**

This option creates a timeline along a selected Drive String, with the choice of a set of vehicles and/or a floating camera.

An optional Start and End chainage can be entered to limit the length of travel.

Enter a speed for the drive and an anchor tin for the four tyre anchor points of the vehicles.

The Calc Time button will calculate, in seconds, the travel time Duration at the selected speed.

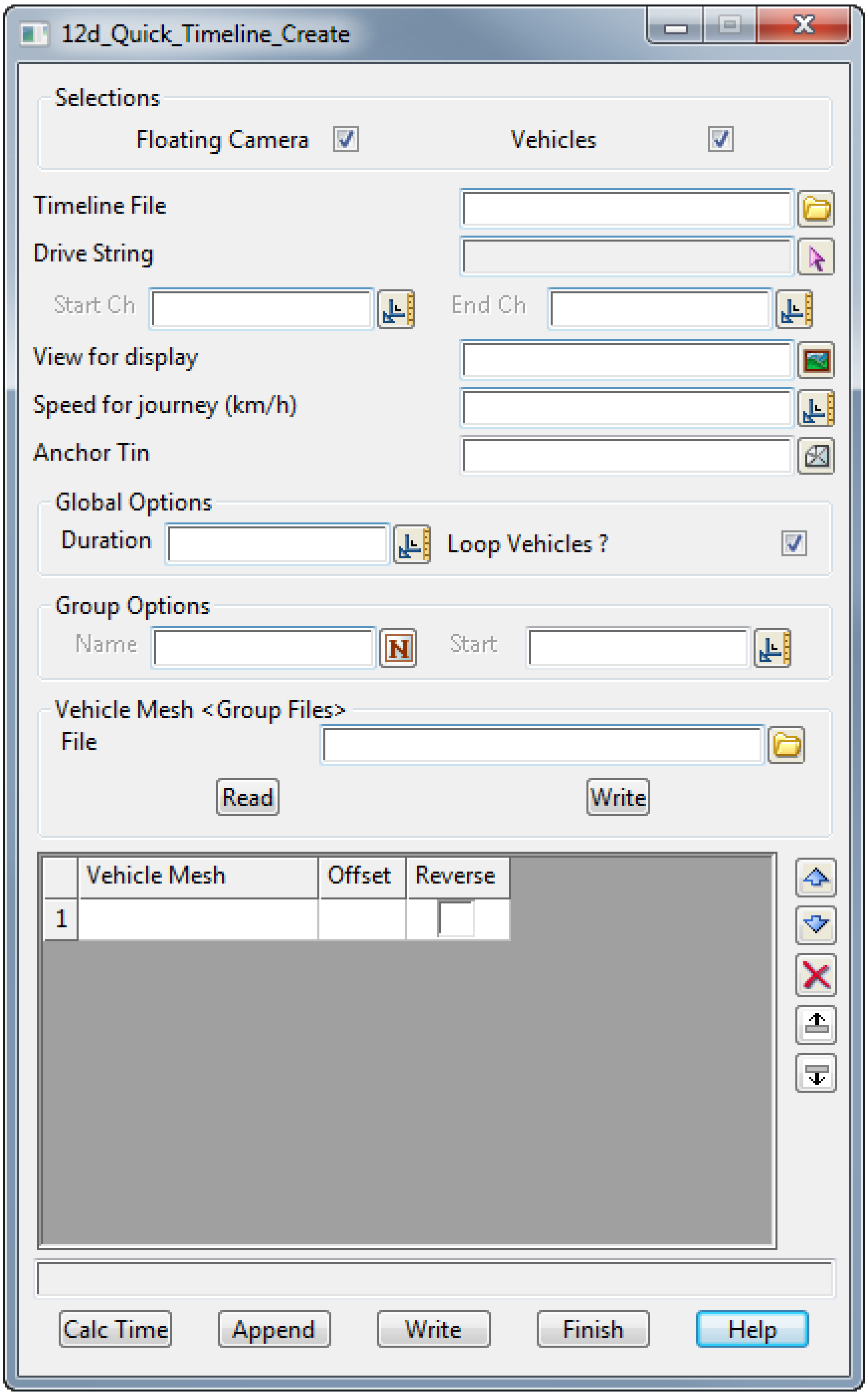
Vehicles are defined as meshes and can be Grouped together under a Name and a start time (in seconds)....optional.

Upon completing your selections (Drive String, Speed, Vehicles etc).... Save the selection using the Append button, and Write the file out.

Note: More than one Drive string and set of vehicles can be selected, appended and finally all written out in the same timeline file. Depending on the location of your drive string selection and its direction, you may have to enter an offset for the vehicles, and/or tick Reverse.

Once the timeline is written, it can be played under **View=>Visualisation=>Timelines=>Play**

Selecting **Quick timeline** brings up the 12d Quick timeline create panel.



The fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

Selections

Floating Camera tick box ticked

A floating camera allows you to specify the floating eye component of a camera.

The target is defined as the Drive String, but during the playing of the timeline, the eye can be moved independently by the user (e.g. using the Orbit).

Vehicles tick box ticked

Ticking this option on turns on appropriate fields for setting up timelines for vehicles.

Timeline File file box available files

If non-blank, entry used as timeline file name.

Drive String string select

If selected, string is used as timeline drive string and timeline duration can be calculated.

If non-blank, entry used as timeline file name.

Start Ch real

If non-blank, entry used as start of timeline...optional.

End Ch real

If non-blank, entry used as end of timeline...optional.

View for display view box available views

If non-blank, entry used to play back the timeline (an Open GL Perspective View is required).

Speed for journey (km/h) real

If non-blank, entry used to calculate the timeline duration in seconds.

Anchor Tin tin box

If non-blank, entry used as the location for the four vehicle tyre points.

Global Options

Duration real

If non-blank, entry used as timeline length (can be calculated initially via the Calc Time button).

Loop Vehicles? tick box ticked

If ticked, the set of vehicles will be replayed during the timeline.

Group Options

Name name box

If non-blank, entry used for the grouping of the vehicles under that name and can be toggled off during the playing of the timeline...optional.

Start real

If non-blank, entry used for start time of the group (in seconds)...optional.

Vehicle Mesh <Group Files>

File file box

If non-blank, entry used for populating the grid selection of vehicle meshes.

Refer library for standard examples.

Read button

Reads in the file above and populates the vehicle grid.

Write button

Writes out the data currently in the vehicle grid to the file.

Vehicle Mesh Grid:

Vehicle Mesh

Vehicle choices (can be viewed under **View=>Visualisation=>Meshes=>Mesh library**

Offset Real

If non-blank entry used for offset from drive string

Reverse tick box not ticked

If ticked, vehicles will run in the opposite direction to the drive string direction.

Calc Time button

Calculates the time of travel in seconds using the Drive String and Speed of journey.

Append button

One or more timelines (Drive String, speed, vehicle etc) can be appended, before finally being written out to the timeline file.

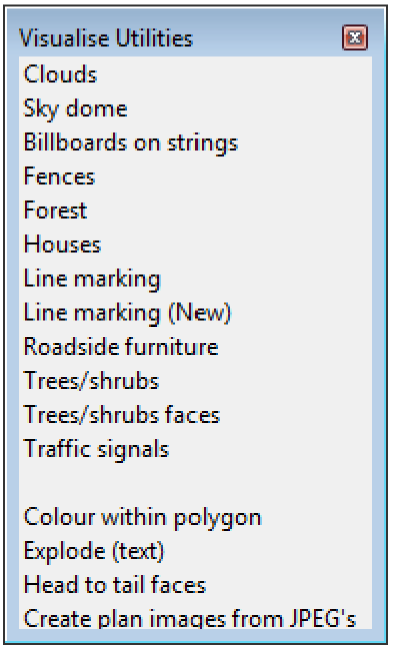
Write button

After appending, the Timeline File is written out.

12.13.35 Utilities

Position of menu: **View =>Visualisation =>Utilities**

The **Utilities** walk-right menu is



create box of clouds as background
create sky dome of clouds as background
change billboards on super strings
inserting fences
create forest
insert houses
create strings for line marking

insert fences, guard rails, signs, walls etc.
insert trees and shrubs as billboards
insert trees and shrubs as faces

colour within a polygon
explode text into vectors so it can be draped onto surface

create plan images from JPEG files which include EXIF and GPS data

For the options, see

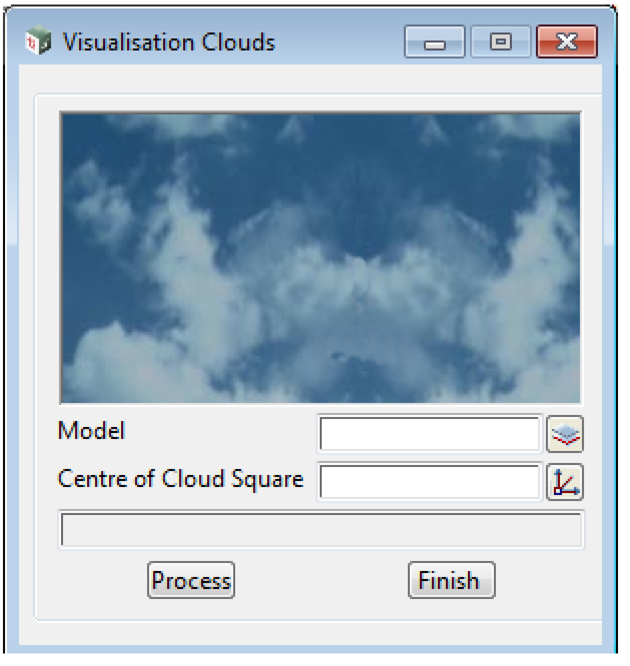
Clouds	12.13.35.1 Clouds
Sky dome	12.13.35.2 Sky dome
Billboards on strings	12.13.5.1 Create and Place Billboard
Fences	12.13.35.3 Fences
Forest	12.13.5.4 Create Forest
Houses	12.13.35.4 Houses
Line marking	12.13.35.5 Line Marking
Line marking (New)	12.13.35.6 Line Marking (New)
Roadside furniture	12.13.8.3 Roadside Furniture
Trees/shrubs billboards	12.13.5.5 Trees/Shrubs as Faces and Billboards
Trees/shrubs faces	12.13.5.5 Trees/Shrubs as Faces and Billboards
Traffic signals	12.13.35.7 Traffic Lighting Placement
Colour within polygon	16.8.2 Colour Within Polygon
Explode (text)	28.9.18 Explode Text
Head to tail faces	12.13.35.10 Head to tail faces
Create plan images from JPEG's	12.13.35.11 Create Plan Images from JPEG Files

12.13.35.1 Clouds

Position of option on menu: **View =>Visualisation =>Utilities =>Clouds**

This option is used to create a square box around the view with the cloud image on the inside of it. From within a scene, this will then appears as a sky with clouds as the background.

On selecting the **Clouds** option, the **Visualisation Clouds** panel is displayed:



The fields and buttons used in the panel have the following functions.

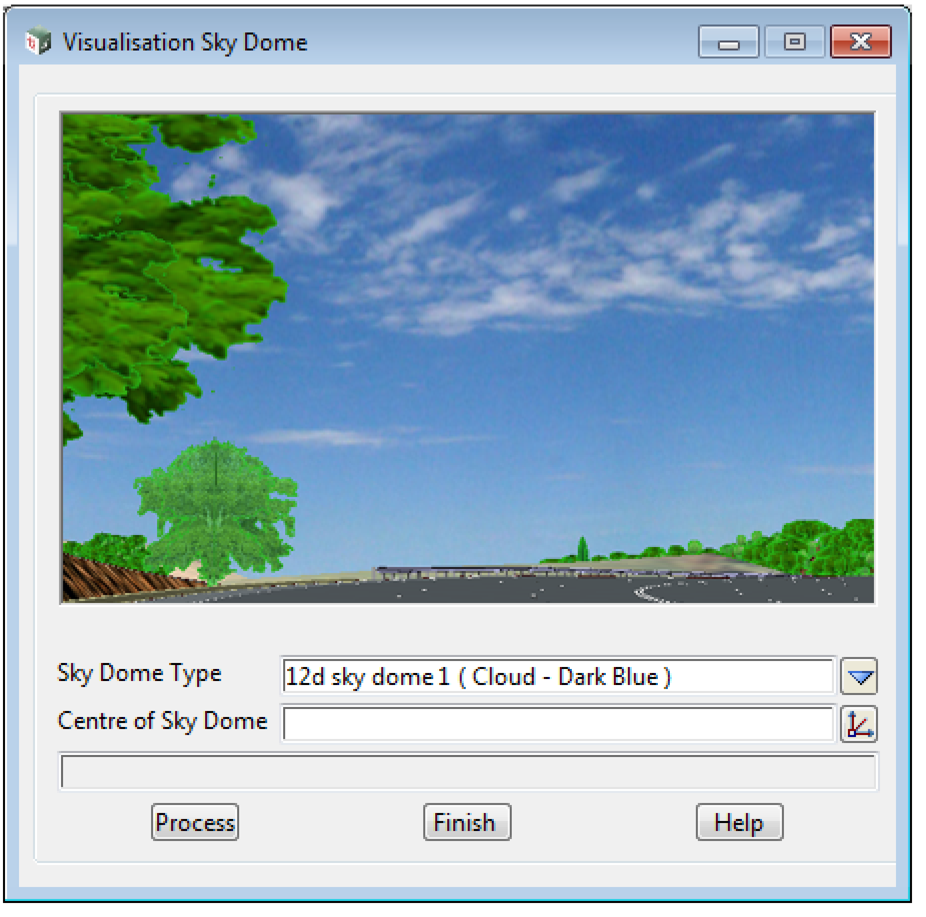
Field Description	Type	Defaults	Pop-Up
Model	model box		available models
model to place the cloud square in.			
Centre of cloud square	xyz box		position select
give the x y z co-ordinates for the centre of the cloud square.			
Process	button		
create the cloud square.			

12.13.35.2 Sky dome

Position of option on menu: **View =>Visualisation =>Utilities =>Sky dome**

This option reads in a 12da of a triangulation and a raster image to form a dome, that represents the sky.

On selecting the **Sky dome** option, the **Visualisation Sky Dome** panel is displayed:



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Sky Dome Type	choice box		Drop down selection available, with a preview above.

12d Sky Dome 1 (Cloud - Dark Blue)

12d Sky Dome 2 (Cloud - Haze)

12d Sky Dome 3 (Cloud - Light Blue)

12d Sky Dome 4 (Cloud - Snow Mtns)

12d Sky Dome 5 (Clear Sky)

Centre of Sky Dome	valid x,y,z		
Process	button		

Read in data from Library and translate to centre coords

Model Format

12d Sky Dome Data	(3 dimensional data for the triangulation)
12d Sky Dome Tin	(3 dimensional triangulation)
12d Sky Dome 1 Raster	(2 dimensional raster for the triangulation)

Note:

More than one sky type can be read in and used but the selected raster is then applied to the same tin.

To view the sky dome in the OpenGL perspective view, add the 1**2d Sky Dome Tin model** only to the view.

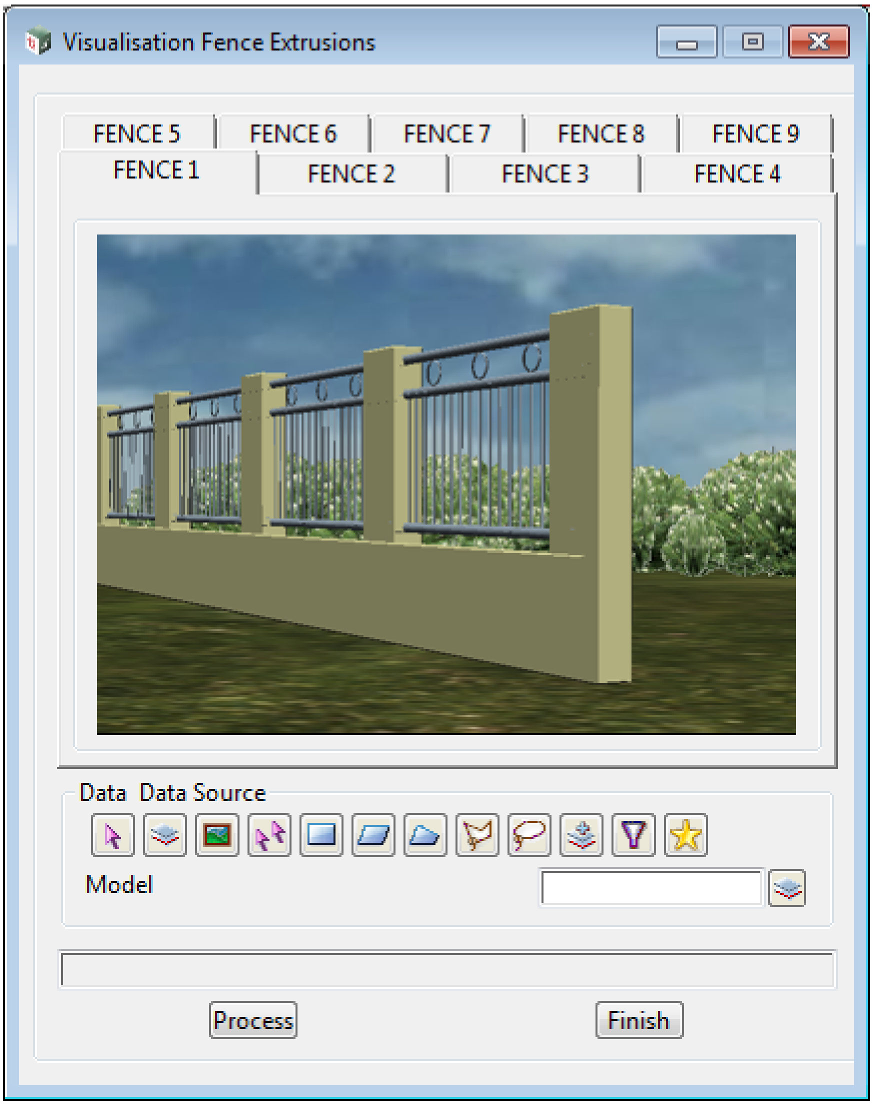
12.13.35.3 Fences

Position of option on menu: **View =>Visualisation => Extrusions =>Fences**

This option is for extruded 12d Solutions supplied fences along strings.

The list of fences is being continuously upgraded by 12d Solutions.

Selecting the **Fences** brings up the **Visualisation Fence Extrusions** panel.



The fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

Clicking on the tabs brings up the type of fence to be extruded along selected super strings.

Data source type Model

data selection type - for a full description go to 4.19.3 Data Source.

Data source input

source of super string to extrude fences along

Process button

apply the fence to the selected super strings

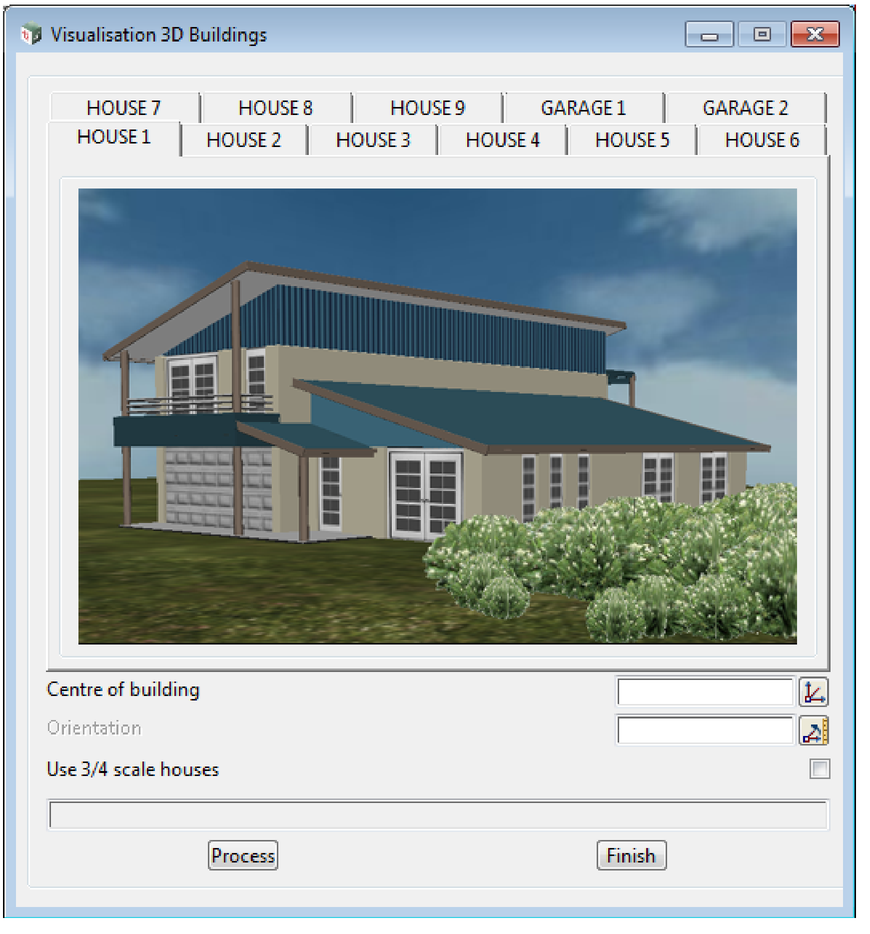
12.13.35.4 Houses

Position of option on menu: **View =>Visualisation => Utilities =>Houses**

This option is for inserting 12d Solutions supplied houses.

The list of houses is being continuously upgraded by 12d Solutions.

Selecting the **Houses** brings up the **Visualisation 3D Buildings** panel.



The fields and buttons used in the panel have the following functions.

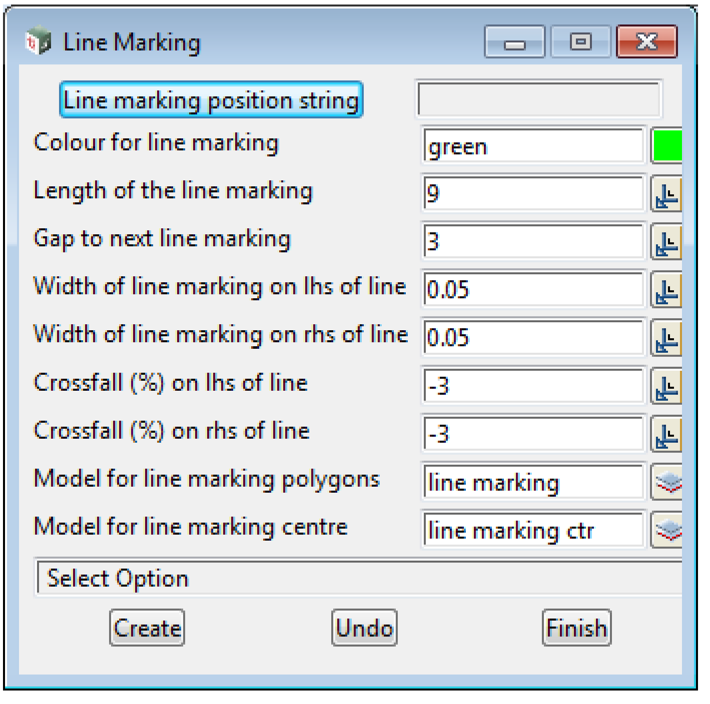
Field Description	Type	Defaults	Pop-Up
Clicking on the tabs brings up the type of house to be inserted.			
Centre of building	xyz box		position select
give the x y z co-ordinates for the centre of the base of the inserted building			
Orientation	angle box		angle select
give the angle to orient the building			
Process	button		
insert the selected building			

12.13.35.5 Line Marking

Position of option on menu: **View =>Visualisation =>Utilities => Line marking**

This option is used to create polygon data in 3d representing the line marking on a road. The polygons can then be included in the design tin for triangulating and the “Colour Triangles Within a Polygon” option used to colour the triangles for use in shades and renderings.

The line marking polygons created by the option represents a dashed line style centred around the selected line marking string marking the position of the dashed line. The length and separation of the dashed lines are user defined and the width of the dash is defined by a width to the left and a width to the right of the line marking string.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up

line marking position stringstring select

select the string to define the position of the line marking on the road. The string will also provide z-values for use in creating the line marking.

Colour for line marking input box green available colours

colour for the line marking polygons.

Length of the line marking input box 9

length of each line mark dash.

Gap to next line marking input box 3

distance between the line marking dashes.

Width of line marking on lhs of line input box 0.05

width of the line marking on the left hand side of the line marking position string.

Width of line marking on rhs of line input box 0.05

width of the line marking on the right hand side of the line marking position string.

Crossfall (%) on lhs of line input box -3

cross fall for the part of the line marking on the left hand side of the line marking position string (so z-

values for the polygon can be calculated).

Crossfall (%) on rhs of line input box -3

cross fall for the part of the line marking on the right hand side of the line marking position string (so z-values for the polygon can be calculated).

Model for line marking polygons model box line marking available models

model for the polygons representing the line marking.

Model for line marking centre model box line marking ctr available models

model for a string with just the part of the line marking position string which is inside the line marking polygons.

Create button

run the option and create the line marking polygons.

Undo button

undo the last set of polygons created since the panel was up.

12.13.35.6 Line Marking (New)

Position of option on menu: **View =>Visualisation =>Utilities => Line marking (New)**

This option is made up of several parts.

1. Create line marking polygons:

The drop down menu has a list of linemarking definitions, that match the linestyles in 12d.

The length, gap and width values are used to create polygons along the strings selected.

Fill in the Linemarking model, colour and the optional tin (used to drape the polygons).

2. Apply Extrusions Only:

In this option extrusions are matched to linestyles, rather than producing individual polygons.

3. Pavement Arrows:

In this option polygons are created from the symbol definition.

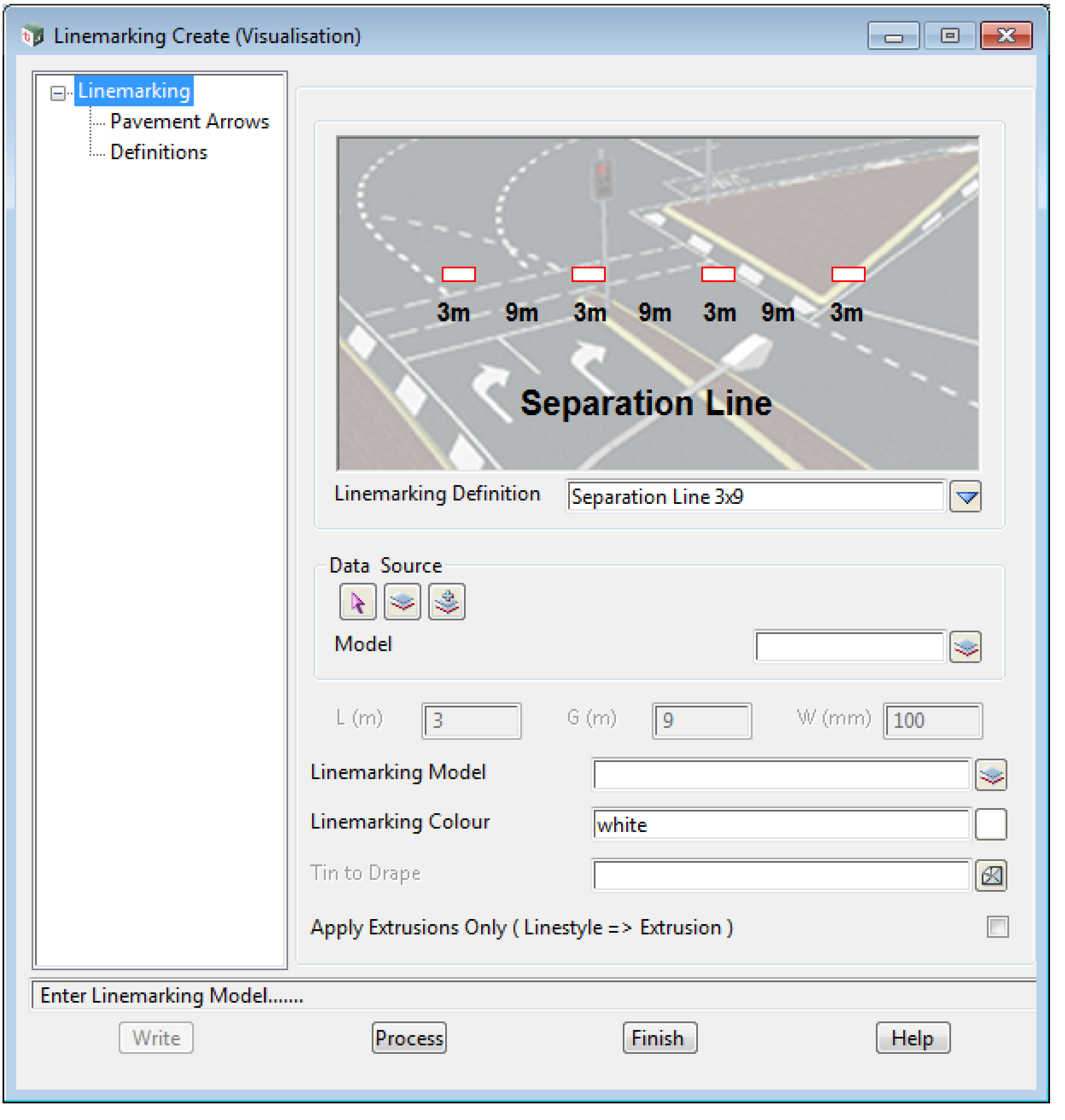
Single Arrows can be placed, using the **Dynamic Orientation** button to select an x,y point on screen.

Multiple Arrows:

Arrow Edge Line Reference	"Pick with direction" along a string
Arrow Spacing	Distance apart
Arrow Offset	Relative to the directional pick
Arrow Start Point	If you select a right turn lane string, picking with direction towards an intersection, then the start point would be at the intersection and subsequent arrows placed at the Arrow Spacing away from the intersection.

Selecting **Line marking (New)** brings up the **Linemarking Create (Visualisation)** panel.

.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Linemarking Menu			
Linemarking Definition	choice box	Separation Line 3x9	various
selected definition is displayed in draw box above			
Data source		Model	
data selection type - for a full description go to 4.19.3 Data Source.			
Linemarking Model	model box		select model
model name for linemarking			
Linemarking Colour	input	white	available colours

the colour of linemarking polygons or arrows

Tin to Drape input select tin

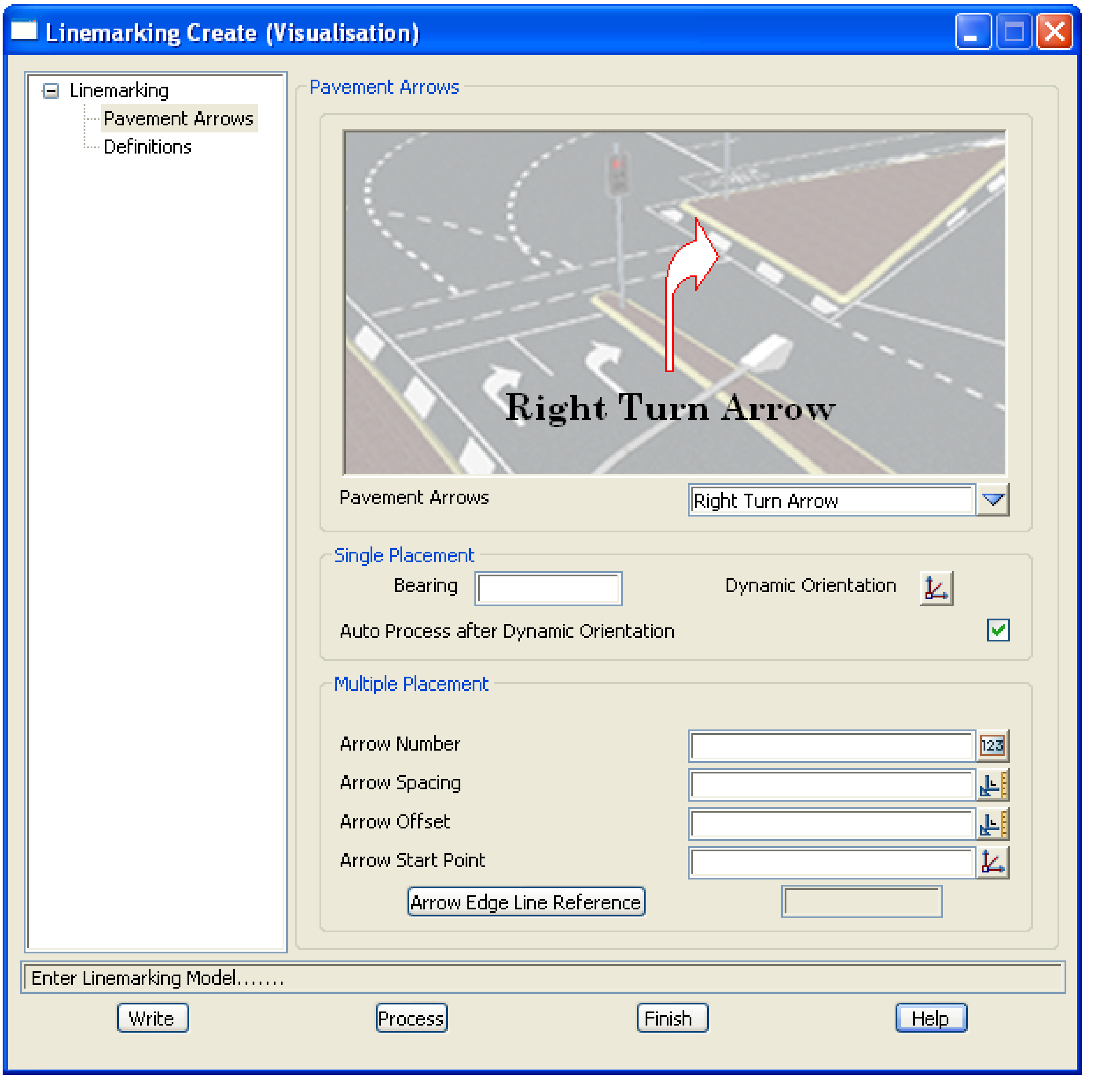
if non-blank, then the strings created are draped onto this tin

Box Offset input 10

Distance to place the IP Info boxes from the IP

Apply Extrusions Only (Linestyle => Extrusion) tick box

If ticked, extrusions are matched to linestyles in the data source, rather than producing individual polygons



Field Description	Type	Defaults	Pop-Up

Pavement Arrows Menu

Pavement Arrows choice box Right Turn Arrow various

selected arrow type is displayed in draw box above

Single Placement:

Bearing real

enter bearing if doing a single arrow placement, rather than dynamic

Dynamic Orientation xyz pick

select point, look for rotating arrow if auto process ticked on

Auto Process after Dynamic Orientation tick box

If ticked, auto rotation activated after insertion point is accepted

Multiple Placement:

Arrow Number integer

number of arrows required

Arrow Spacing measures box At Point, Point to Point, String from Point, String to Point)

interval distance along arrow edge string to place arrows

Arrow Offset measures box At Point, Point to Point, String from Point, String to Point)

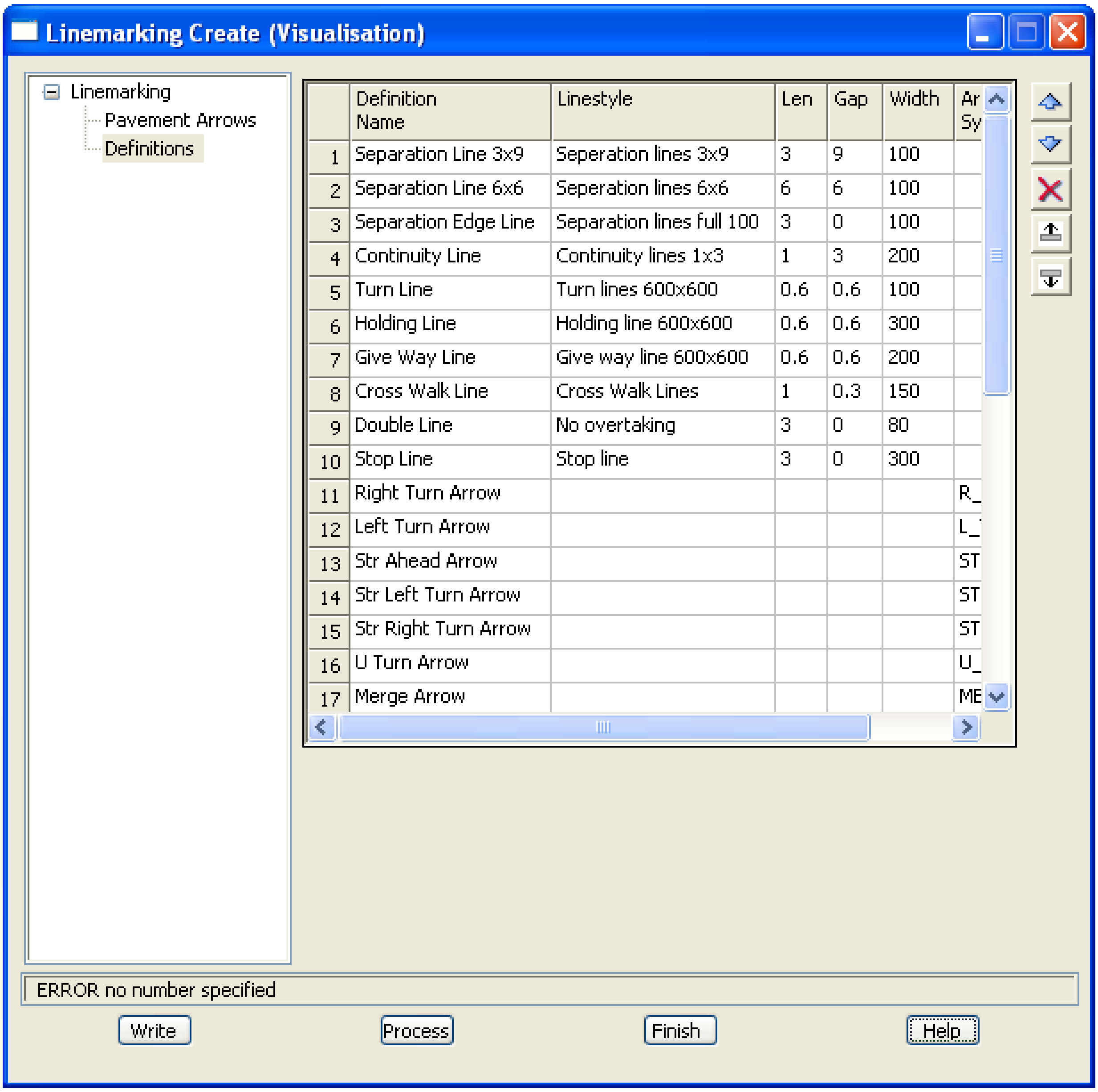
offset from arrow edge string

Arrow Start Point xyz pick

Example: If you select a right turn lane string, picking with direction towards an intersection, then the start point would be at the intersection and subsequent arrows placed at the ***Arrow Spacing*** *away from the intersection.*

Arrow Edge Line Reference select pick

reference and orientation string for arrows..."pick with direction"



	Definition Name	Linestyle	Len	Gap	Width	Ar Sy
1	Separation Line 3x9	Seperation lines 3x9	3	9	100	
2	Separation Line 6x6	Seperation lines 6x6	6	6	100	
3	Separation Edge Line	Separation lines full 100	3	0	100	
4	Continuity Line	Continuity lines 1x3	1	3	200	
5	Turn Line	Turn lines 600x600	0.6	0.6	100	
6	Holding Line	Holding line 600x600	0.6	0.6	300	
7	Give Way Line	Give way line 600x600	0.6	0.6	200	
8	Cross Walk Line	Cross Walk Lines	1	0.3	150	
9	Double Line	No overtaking	3	0	80	
10	Stop Line	Stop line	3	0	300	
11	Right Turn Arrow					R_
12	Left Turn Arrow					L_
13	Str Ahead Arrow					ST
14	Str Left Turn Arrow					ST
15	Str Right Turn Arrow					ST
16	U Turn Arrow					U_
17	Merge Arrow					ME

Definitions

A default file [12d_linemarking.def] is read in from the library and saved to the.project area.

Edits can be carried out to the following definitions.

Field Description	Type	Defaults		Pop-Up
Definition	**Linestyle**	**Len**	**Gap**	**Width**
Separation Line 3x9	Separation lines 3x9	3	9	100
Separation Line 6x6	Separation lines 6x6	6	6	100
Separation Edge Line	Separation lines full 100	3	0	100
Continuity Line	Continuity lines 1x3	1	3	200
Turn Line	Turn lines 600x600	0.6	0.6	100
Holding Line	Holding line 600x600	0.6	0.6	300
Give Way Line	Give way line 600x600	0.6	0.6	200
Cross Walk Line	Cross Walk Lines 1x0.3	1	0.3	150

Double Line	No overtaking	3	0	80
Stop Line	Stop line	3	0	300

Arrow Symbol

Right Turn Arrow	R_TURN
Left Turn Arrow	L_TURN
Str Ahead Arrow	ST_AHEAD
Str Left Turn Arrow	ST_L_TURN
Str Right Turn Arrow	ST_R_TURN
U Turn Arrow	U_TURN
Merge Arrow	MERGE

Write button

Write out any definition changes to 12d_linemarking.def

Process button

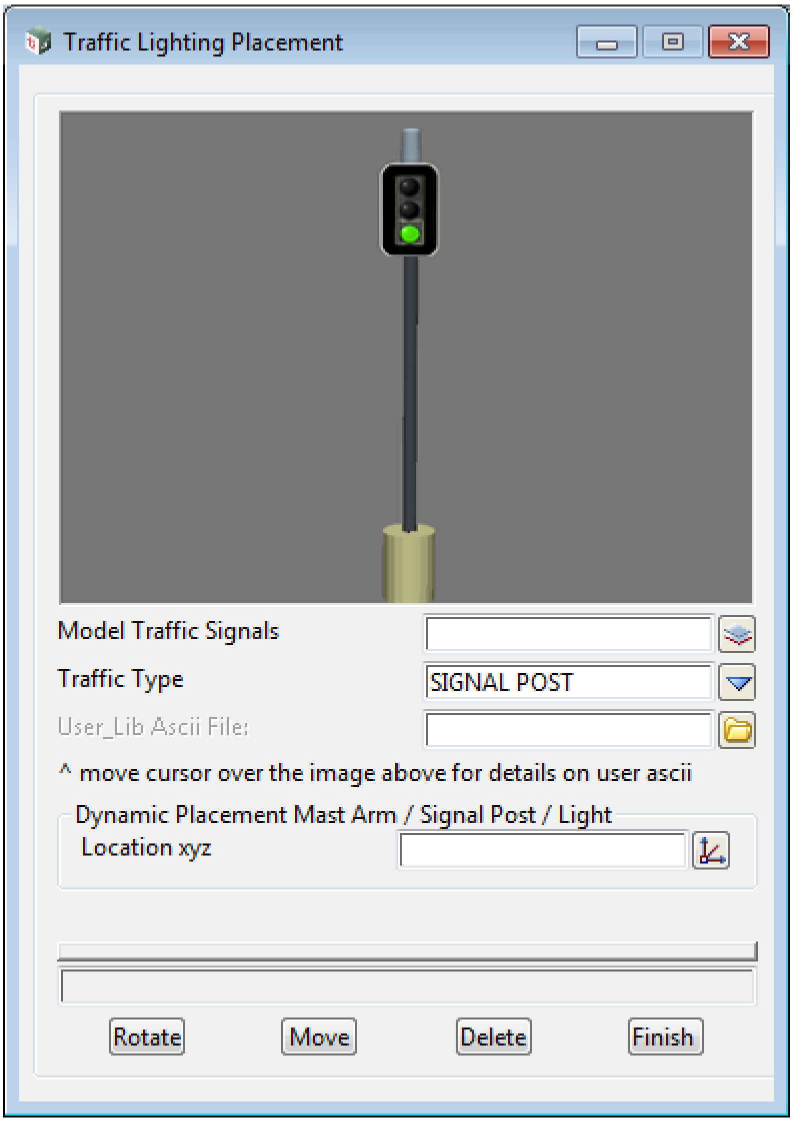
Runs and creates the linemarking

12.13.35.7 Traffic Lighting Placement

Position of option on menu: **View =>Visualisation =>Utilities =>Traffic signals**

This section of documentation is a work in progress and will be updated in subsequent releases.

On selecting the **Traffic signals** option, the **Traffic Lighting Placement** panel is displayed:



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model Traffic Signals	model box		available models
Traffic Type	choice box	SIGNAL POST	available choices
User_Lib Ascii File:	file		
Location xyz			
Rotate	button		
Move	button		
Delete	button		

12.13.35.8 Colour Within Polygon

Position of option on menu: **View =>Visualisation =>Utilities =>Colour within polygon**

This option colours triangles in a tin. It has already been documented as

Tins =>Colour =>Colour within polygon

in the section 16.8.2 Colour Within Polygon.

12.13.35.9 Explode Text

Position of option on menu: **View =>Visualisation =>Utilities =>Explode text**

This option explodes text into lines and arcs. It has already been documented as

Utilities =>A-G => Explode text

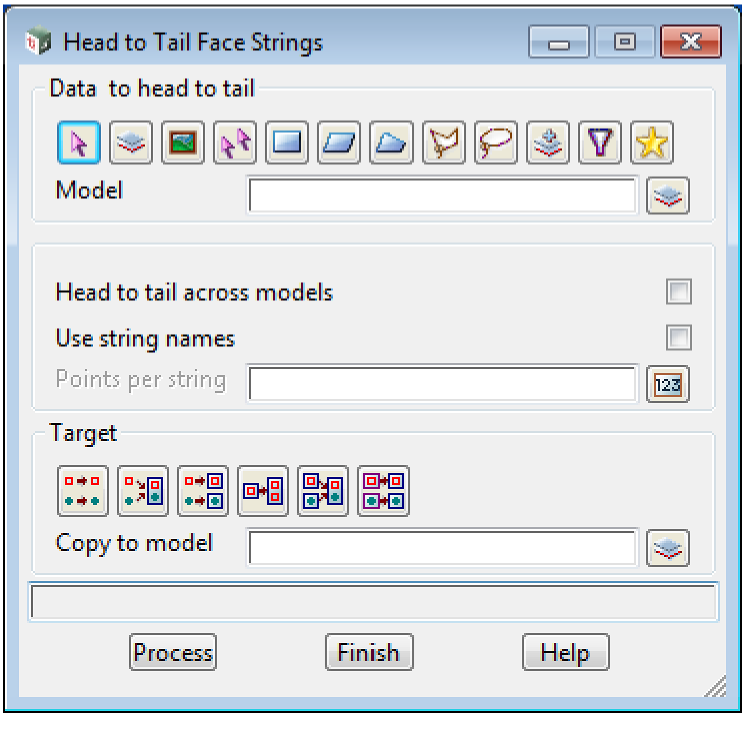
in the section 28.9.18 Explode Text.

12.13.35.10 Head to tail faces

Position of option on menu: **View =>Visualisation =>Utilities =>Head to tail faces**

This section of documentation is a work in progress and will be updated in subsequent releases.

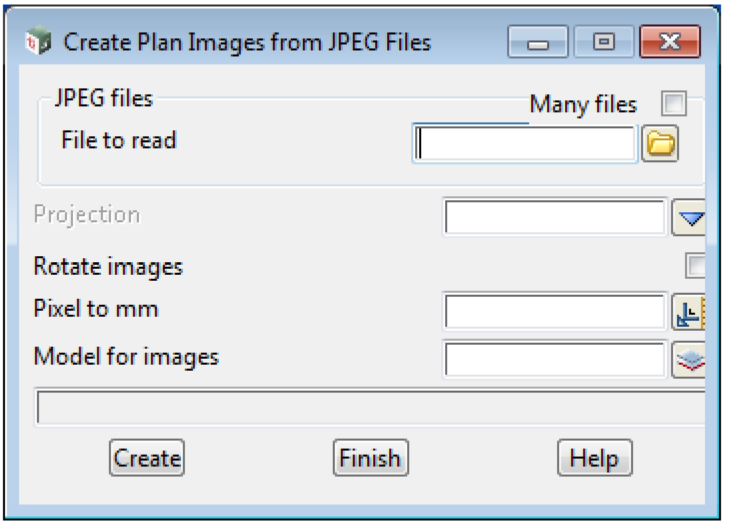
Selecting **Head to tail faces** brings up the **Head to Tail Face Strings** panel.



12.13.35.11 Create Plan Images from JPEG Files

Position of option on menu: **View =>Visualisation =>Utilities =>Create plan images from JPEG's**

Selecting **Create plan images from JPEG's** brings up the **Create Plan Images from JPEG Files** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Advanced	tick box	not ticked	

clicking **Advanced** *brings up a grid to allow the user to enter many JPEG files.*

File to read	file box		available *.jpg files

the JPEG files to read in and create images from. The JPEG file must include EXIF and GPS information, which is used to position the image. If Projection *is not blank, then it is used to convert the lat/long of the GPS values to the XYZ co-ordinates. If* Projection *is blank, then the co-ordinates are left as lat/long.*

Projection	choice box		

if a Project Projection is set, then it is placed in the Projection *field. This can be changed to any other projection. The projection is used to convert the lat/long of the GPS values to the XYZ co-ordinates. If no projection is set, then the co-ordinates are left as lat/long.*

Rotate images	tick box	not ticked	

*if **ticked**, the rotate value from the EXIF will be used to rotate the image.*

Pixel to mm	input box		

the units for images are pixels (width and height). The pixel width and height of the image are multiplied by the **Pixel to mm** *(mm being millimetres) value to give width and height size in world unit (metres) which is needed when inserting the image.*

Model for images	input box		

the model for the image to be inserted into.

Create	button		

*when all the fields have been entered, the **Create** button creates images from the selected files.*

13 Models

Position of menu: **Model**

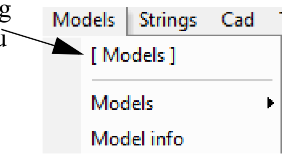
In 12d Model, models contain all the terrain and design information in the form of strings, tins and trimeshes.

The **Models** menu contains the options needed to create new models, rename models, report, clean models and delete models from the project.

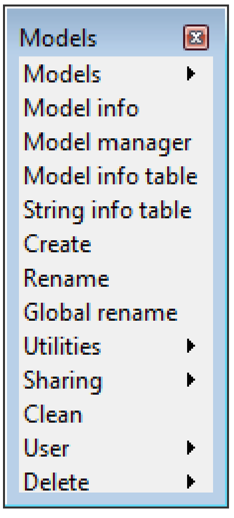
The **Models** walk-right menu is

on Main menu

create floating
Models menu



on 12d Model menu and
floating Models menu



list of models and model info
information on models

in table form, gives min, max x,y,z for each model
" " gives min, max x,y,z for each string in model
create a model
rename a model
rename many models
options to add, remove and save models
sharing models
clean all items in a model
Models User menu
delete models from disk

For the option *Models*, go to section	13.1 Models
Model info	13.2 Model Information
Model manager	
Model info table	13.3 Model Information Table
String info table	13.4 String Information Table
Create	13.5 Create
Rename	13.6 Rename
Global rename	13.7 Global Rename
Utilities	13.8 Utilities
Sharing	13.9 Sharing
Clean	13.10 Clean
Delete	13.11 Delete

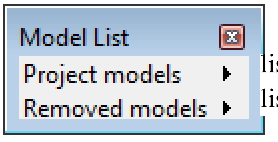
13.1 Models

Position of menu: **Model =>Models**

The **Models** walk-right menu provides options to list all the models in the project, and list all the models in the project area but not added to the project (removed models).

If a model is selected from the project model list, then it is automatically loaded into the **Model Information** panel.

The **Models** walk-right menu is



list of models in project
list of non-project models

For Project models go to 13.1.1 Project Models
Removed models 13.1.2 Removed Models

13.1.1 Project Models

Position of option on menu: **Model =>Models =>Project models**

The **project models** walk-right menu provides a list of all the models in the project and if a model name is selected from the list, a model information panel is fired up with the selected model name already in the model field.

13.1.2 Removed Models

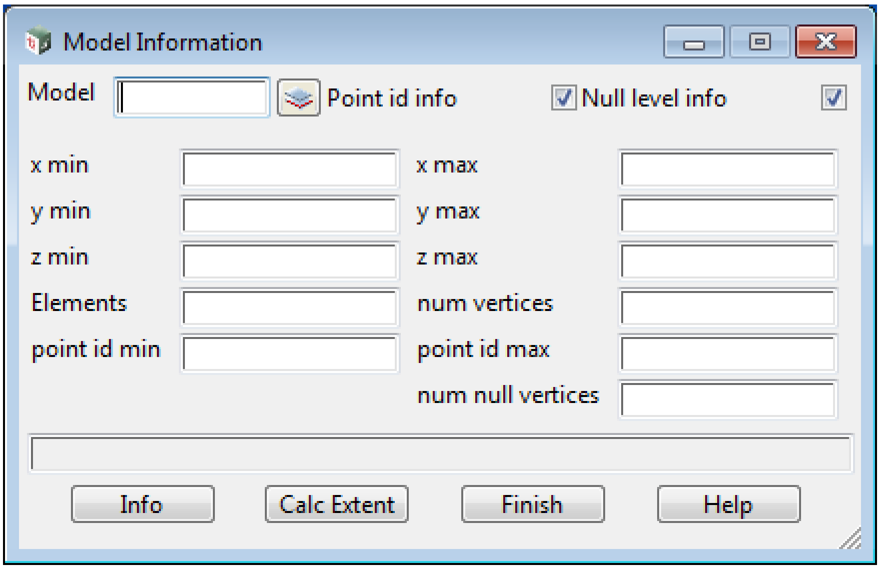
Position of option on menu: **Model =>Models =>Removed Models**

The **removed models** walk-right menu provides a list of all the models in the project area that are not in the project.

They would be mainly models that were in the project but have been removed from the project but not deleted from the disk.

13.2 Model Information

Position of option on menu: **Model =>Model Info**
Selecting **Model info** fires up the **Model Information** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model	input		available models

input the name of the model to get information about

xmin/ymin/zmin, xmax/ymax/zmax output

returns the model limits

point id min/max output

minimum/maximum integer point id in the model

Elements output

returns the number of elements in the model

num vertices output

returns the number of vertices in the model

num null vertices output

returns the number of null vertices in the model

Info button

get the information for the model given in the Model field.

Calc Extent button

recalculate the x, y, z bounding box for the model given in the Model field.

How to Use the Panel

The model information for the model given in the model field is retrieved and placed in the appropriate panel fields when the model name is entered into the model field from the pop-ups, or a <return> is entered after entering the model name into the model field, or on selecting the **Info** button.

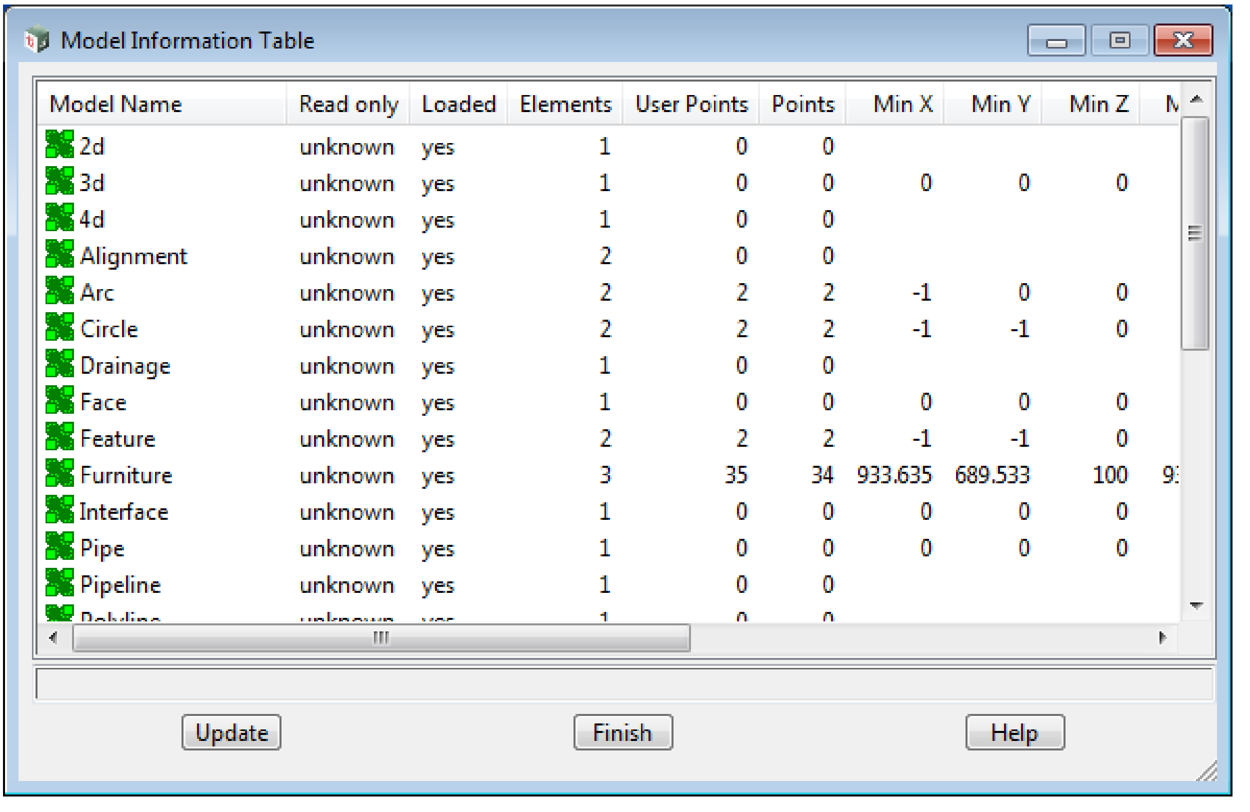
13.3 Model Information Table

Position of option on menu: **Model =>Model Info Table**

The **Model info table** option displays the minimum and maximum x, y and z values for every model in the project in one scrolling table.

The models and minimum and maximum columns can be sorted into ascending or descending order by using bringing up the **sort** menu on the column header fields.

Selecting **Model info table** fires up the **Model Information Table** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model	column		**sort** menu

all the models in the project are listed in the model *column.*

Read only	column		**sort** menu
Loaded	column		**sort** menu

if no, in this current opening of the project, the model has not yet been fully loaded into **12d Model**.

Min X, Min Y, Min Z **Max X, Max Y, Max Z**	columns		**sort** menu

the minimum/maximum values for the model are displayed in the columns

Created/Updated	columns	dates	

date the model was first created/updated

Update	button		

recalculate the minimum/maximum information in the table.

Notes

1. This is a scrolling panel. If there is too much information to fit into the table, then the scrolling arrow on the right hand side of the table must be used to display the extra information.
2. The grid can be sorted by any of the columns.

13.4 String Information Table

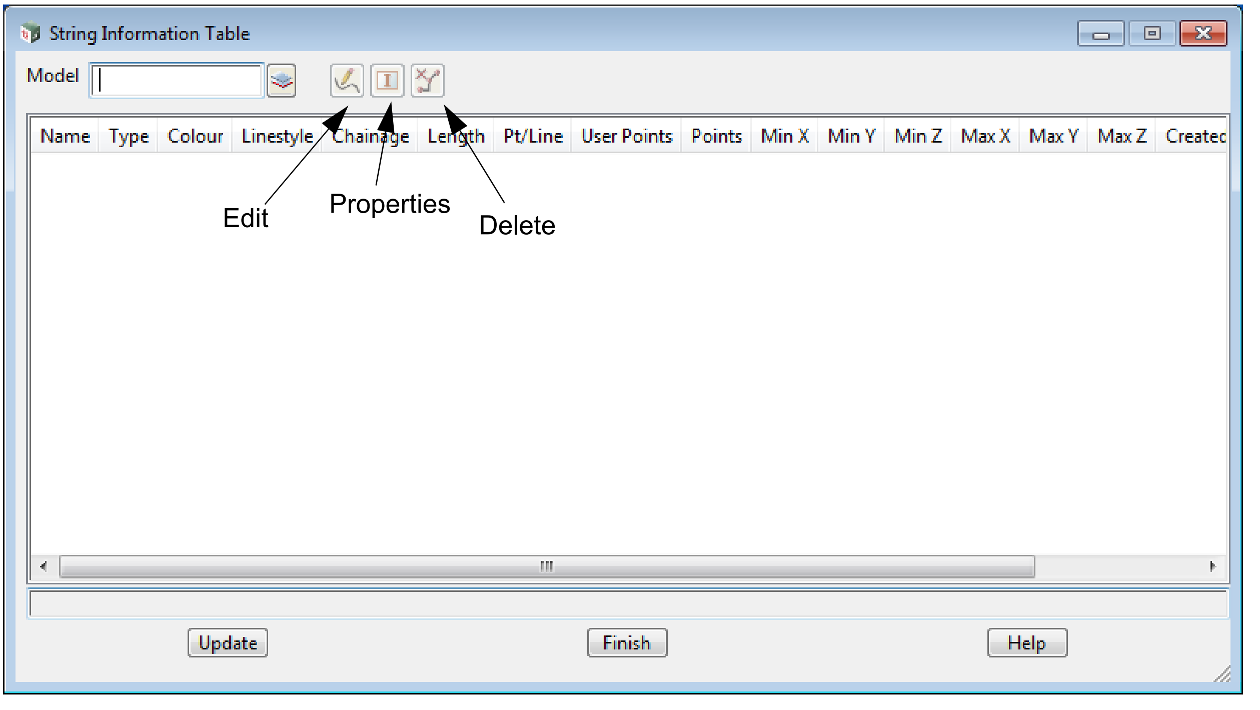
Position of option on menu: **Model =>String info table**

The **String info table** option displays the minimum and maximum x, y and z values for every string in a given model in one scrolling table.

The models and minimum and maximum columns can be sorted into ascending or descending order by using bringing up the **sort** menu on the column header fields.

Selecting **String info table** fires up the **String Information Table** panel.

The greyed out icons ***Edit***, ***Properties***, ***Delete*** only appear when a model has been selected and a row of the table selected.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model	Model box		available models
give the name of the model to calculate the minimum/maximum string information for the table.			
Name	column		**sort** menu
all the strings in the selected model are listed in the name *column.*			
Type, Colour, Linestyle	column		**sort** menu
type, colour and linestyle of the string			
Chainage	column		**sort** menu
start chainage of the string			
Length	column		**sort** menu
2d length of the string			
Pt/Line	column		**sort** menu
point/line breakline type for the string			
Min X, Min Y, Min Z	columns		**sort** menu

Max X, Max Y, Max Z

the minimum/maximum values for each string in the model are displayed in the columns

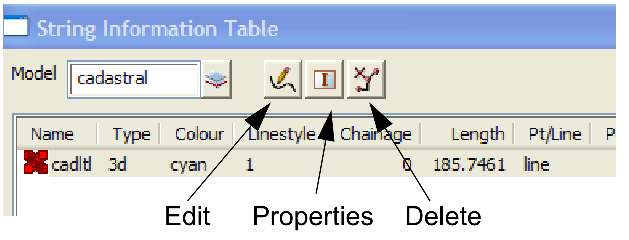
Created/Updated columns dates

date the string was first created/updated

Update button

recalculate the minimum/maximum information in the table.

When data has been read in for a model, the three icons, ***Edit***, ***Properties***, ***Delete*** appear



After clicking on a row in the table (and hence specifying a string), the specified string is highlighted on any plan views it is on, and the icons can be used to *edit* the string, bring up the string properties table or *delete* the string.

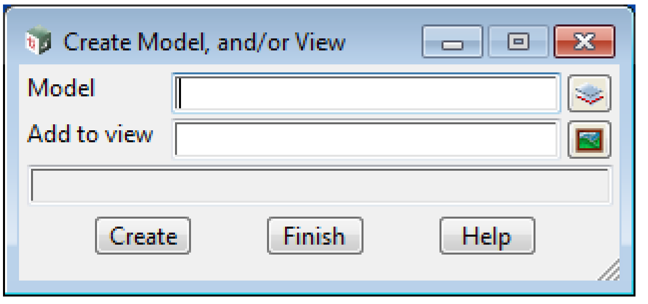
Notes

1. This is a scrolling panel. If there is too much information to fit into the table, then the scrolling arrow on the right hand side of the table must be used to display the extra information.
2. The grid can be sorted by any of the columns.

13.5 Create

Position of option on menu: **Model =>Create**

On selecting the **Create** option, the **Create Model, and/or View** panel is displayed.This panel can be used to create new models and add a model to a view.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model	model box		all models

name of the model to be created and/or added to a model and/or view.

Add to view	view box		all views

if non-blank, the name of the view to which the model given in the model-field will be added.

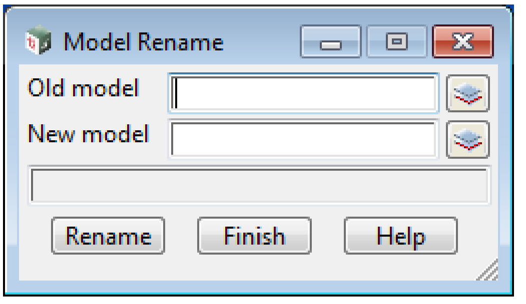
Create	button		

If the model given in the Model field does not exist, it will be created.
If the add to view *field is non-blank, then the model given in the model field will be added to the view given in the* add to view *field.*

13.6 Rename

Position of option on menu: **Model =>Rename**

On selecting the **rename** option, the **Model Rename** panel is displayed. This panel can be used to change the names of existing models



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Old model	model box		all models

name of the model to be renamed.

New model	model box		

new name for the model

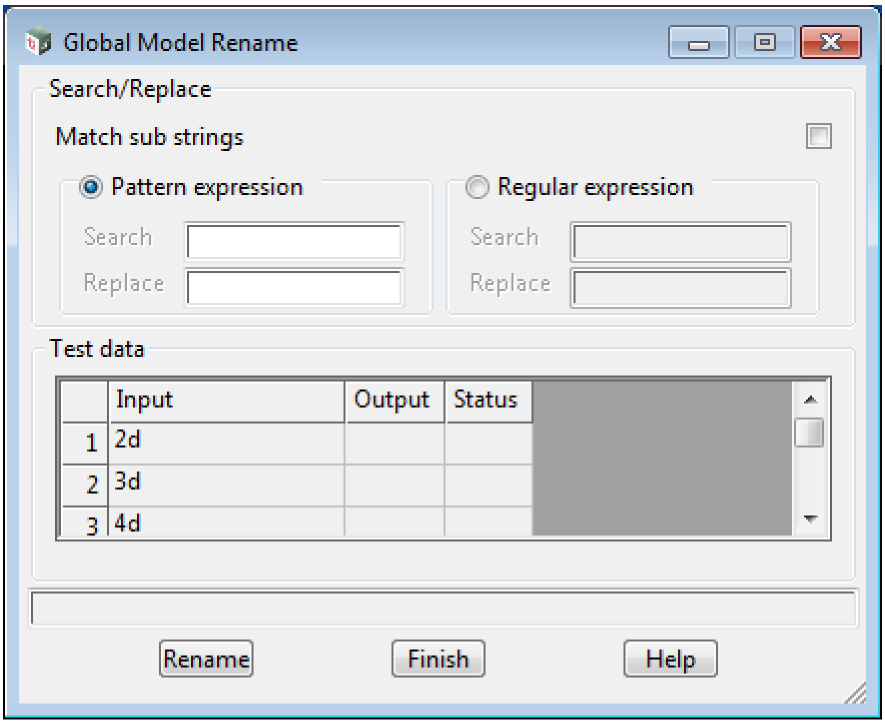
Rename	button		

Change the name of the model in the old model *field to the name given in the* new model *field.*

13.7 Global Rename

Position of option on menu: **Model =>Global rename**

Selecting the **Global rename**, displays the **Global Model Rename** panel. This panel is used to change the names of many existing models by matching according to a *pattern* expression or a *regular* expression.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Match sub strings	tick box		

*if **ticked**, the* Search *expression is used to match against part of each model name.*
*If **not ticked**, the* Search *expression is used to match against the entire model name.*

Pattern expression radio button

*if set on, then **Pattern** expressions given in the* Search *and* Replace *fields are used to modify model names. Pattern expressions include the standard wild card * and wild character !.*

Search input

pattern to search for in the model names. For example " tin" will select all models with a name ending with " tin"*

Replace input

replacement for the search pattern found in the model name. For example, "tin " in the Replace *field, takes the matched part of the model name and adds "tin " to the front of it.*

Hence the Search pattern " tin" and Replace pattern "tin *" finds all models with names ending in " tin" and renames them with the name starting with "tin " (and the " tin" at the end of the name is dropped off).*

Regular expression radio button

*if set on, then **Regular** expressions given in the* Search *and* Replace *fields are used to modify model names.*

Search input

regular expression to search for in the model names.

Replace input

replacement for the search expression found in the model name.

Test data grid

The Test data *grid shows the effect of the Search and Replace on all the model names in the project.*

*The **Input** column shows the existing model name.*
*The **Output** column shows the name after applying the* Search *and* Replace.
*The **Status** column displays if their has been a **match** or **no match**.*

Rename button

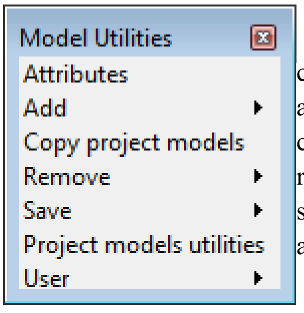
change the name of all the models in the project according to the selected Search *and* Replace *fields.*

13.8 Utilities

Position of menu: **Model =>Utilities**

The **utilities** menu contains miscellaneous options involving models.

The **utilities** walk-right menu is



create/edit attributes for models
add removed and other project models to project
copy model from another project
remove models from project
save models to disk
add/removes, saves, cleans, deletes models

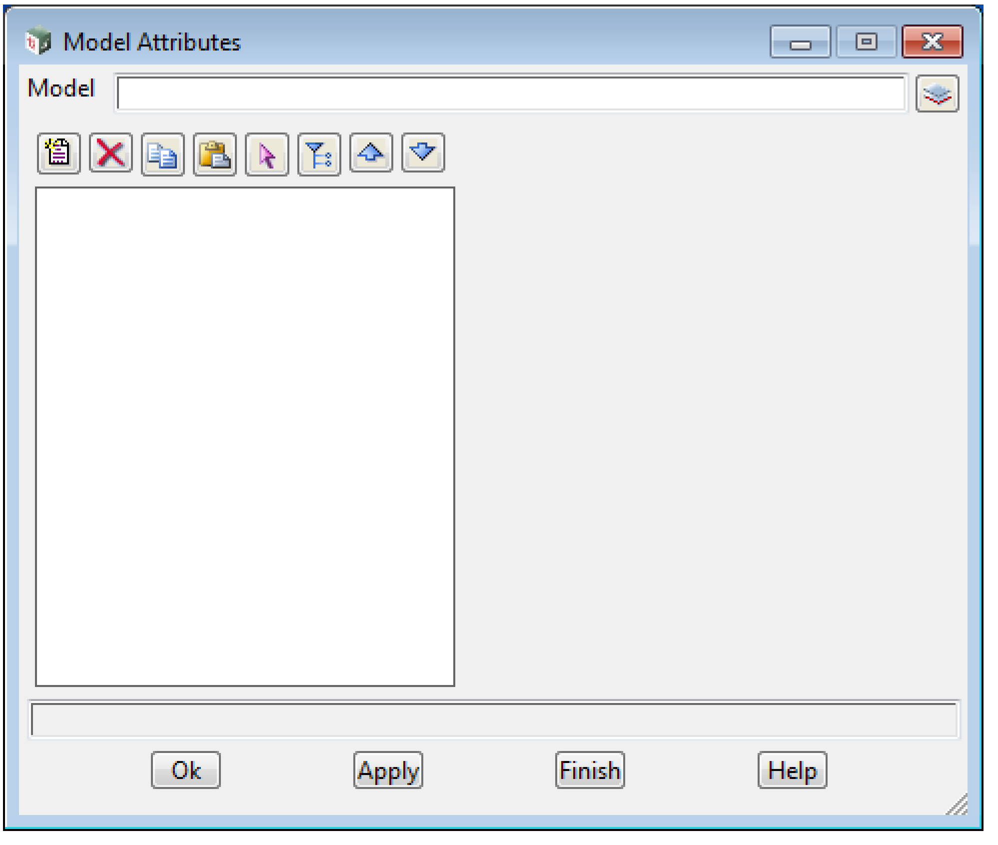
For the option *Attributes*, go to 13.8.1 Attributes
Add 13.8.2 Add
Copy project models 13.8.3 Copy Project Models
Remove 13.8.4 Remove
Save 13.8.5 Save
Project model utilities 13.8.6 Project Model Utilities

13.8.1 Attributes

Position of option on menu: **Model =>Utilities =>Attributes**

The **Attributes** options displays, creates and edits attributes for models.

On selecting **Attributes**, the **Model Attributes** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	DefaultsPop-Up
Model	model box	all available models

model to create/edit attributes of

Name

name of the attribute

Type integer, real, text

type of attribute - integer, real or text

Data

value for the attribute

OK button

set the attributes to the values in the panel and then exit the panel.

Apply button

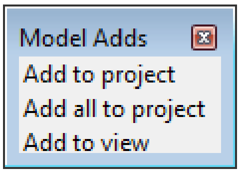
set the attributes to the values in the panel but don't exit the panel.

13.8.2 Add

Position of menu: **Model =>Utilities =>Add**

Models can be added to the project and to views.

The **model adds** walk-right menu is



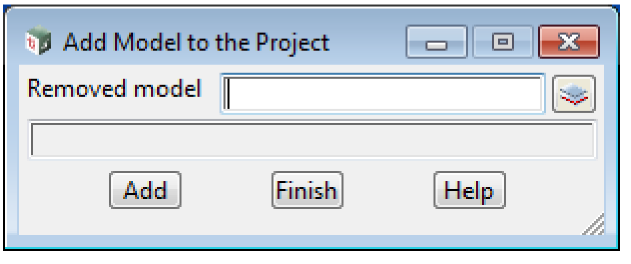
For the option Add to project go to [13.8.2.1 Add To Project](#)
Add all to project [13.8.2.2 Add All To Project](#)
Add to view [13.8.2.3 Add to View](#)

13.8.2.1 Add To Project

Position of option on menu: **Model =>Utilities =>Add =>Add to project**

The **add to project** option is used to add a removed model back into the project.

On selecting the **add to project** option, the **add model to the project** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Removed model	input		removed models

name of the model not in the project that is to be added to the project.

Add	button

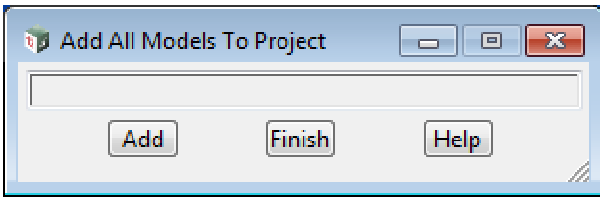
add the model given in the removed model *field to the working project.*

13.8.2.2 Add All To Project

Position of option on menu: **Model =>Utilities =>Add => Add all to Project**

The **add all to project** option is used to add all the removed models back into the project.

On selecting the **add all to project** option, the **add all models to project** panel is displayed.



The fields and buttons used in this panel have the following functions.

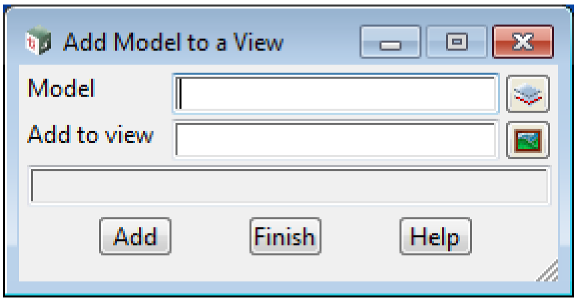
Field Description	Type	Defaults	Pop-Up
Add	button		

after selecting this button, all removed models in the working project will be added to the project.

13.8.2.3 Add to View

Position of option on menu: **Model =>Utilities =>Add =>Add to view**

On selecting the **add to view** option, the **add model to a view** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model	input		available models

name of the model to be added to a view.

Field Description	Type	Defaults	Pop-Up
Add to view	input		available views

name of the view to which the model given in the model *field will be added.*

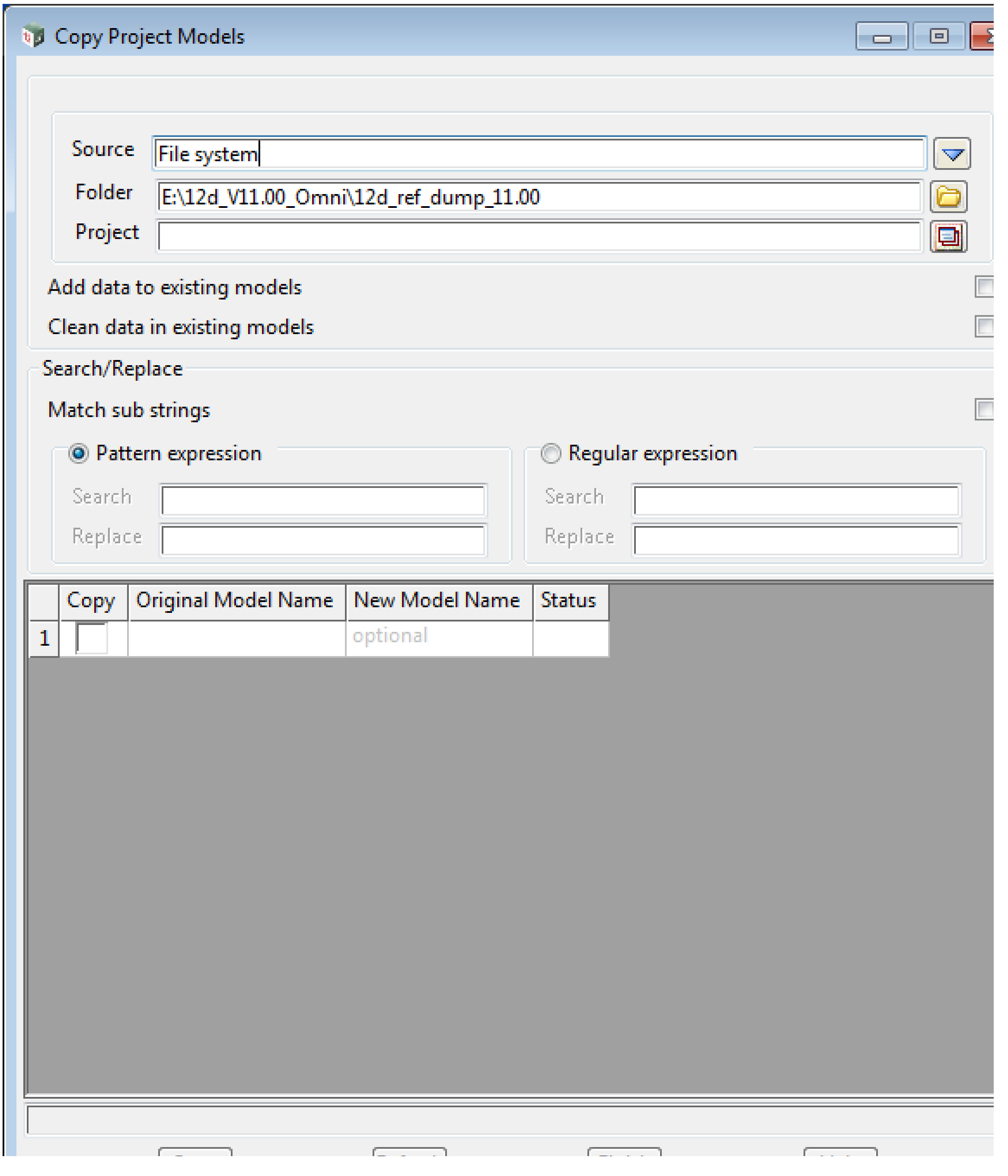
Field Description	Type	Defaults	Pop-Up
Add	button		

add the model given in the model *field to the view given by the* add to view *field.*

13.8.3 Copy Project Models

Position of option on menu: **Model =>Utilities =>Copy Project Models**

Selecting **copy project models** brings up the **Copy Project Models** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Folder	input		Select folder panel

name of the folder that the project to get the model from, is in.

Project	input		projects in the folder

name of the project in the folder given in the folder *field, that the model is to be copied from. Once a project is selected, all the models from that project will be listed in the Original Model Name* *column.*

Add data to existing models tick box

if ticked, if the model that the data is being read into already exists, the new data is copied into the model

Clean data in existing models tick box

if ticked, any existing data in the models being read into, is first cleaned out.

*NOTE - If neither tick box is ticked then if the model already exists, **no data is copied**.*

Search/Replace

section for renaming models from the selected project

Match sub strings tick box

*if **ticked**, the **Search** expression is used to match against part of each model name.*
*If **not ticked**, the **Search** expression is used to match against the entire model name.*

Pattern expression radio button

*if set on, then **Pattern** expressions given in the* Search *and* Replace *fields are used to modify model names. Pattern expressions include the standard wild card * and wild character !.*

Search input

pattern to search for in the model names. For example " tin" will select all models with a name ending with "tin "*

Replace input

replacement for the search pattern found in the model name. For example, "tin " takes the matched part of the model name and adds " tin " to the front of it.

Hence the Search pattern " tin" and Replace pattern "tin *" finds all models with names ending in " tin" and renames them with the name starting with "tin " (and the " tin" at the end of the name is dropped off).*

Regular expression radio button

*if set on, then **Regular** expressions given in the* Search *and* Replace *fields are used to modify model names.*

Search input

regular expression to search for in the model names.

Replace input

replacement for the search expression found in the model name.

Model Names Grid

*The models selected by the Search and Replace expressions are shown in the **Original Model Name** column. Note that if the model already exists in the current project, then the cell for that model in Original Model Name column will be displayed in yellow.*

Copy tick box

if ticked, the model will be copied.
If not ticked, the model will not be copied

*Clicking RB on **Copy** at the top of the column brings up a menu to **Clear** which turns all the ticks off.*

Original Model Name column

name of the existing model in the selected project

New Model Name column

*if **non-blank**, the new name to be given to the copied model.*
If blank, the original model name is used.

The New Model Names can be from applying the Search *and* Replace, or just typing them in.

Status column

displays if their has been a match *or* no match *for the search/replace for renaming models*

Copy button

copy to this project, the model given in the model *field from the project given in the* project *field.*

Refresh button

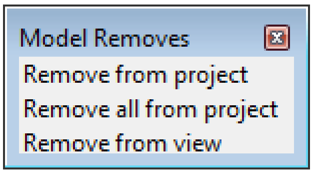
refresh the list of all models in the selected project

13.8.4 Remove

Position of menu: **Model =>Utilities =>Remove**

Models can be removed from the project and from views.

The **model removes** walk-right menu is



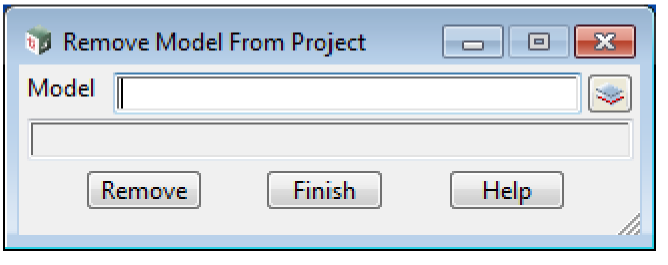
The options in the menu will now be described.

For the option Remove from project go to 13.8.4.1 Remove Model from Project
Remove all from project 13.8.4.2 Remove All Models From Project
Remove from view 13.8.4.3 Remove From View

13.8.4.1 Remove Model from Project

Position of option on menu: **Model =>Utilities =>Remove =>Remove from project**

On selecting the **remove from project** option, the **remove model from project** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model	input		available models

name of the model to be remove from the working project.

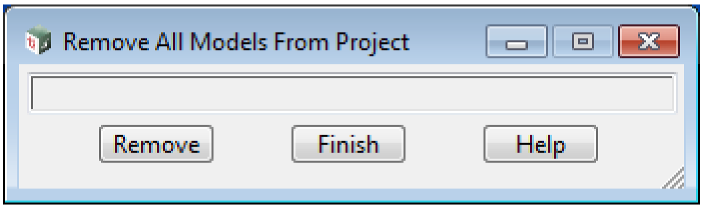
Remove	button		

after selecting this button, the model given in the model field will be removed from the working project.

13.8.4.2 Remove All Models From Project

Position of option on menu: **Model =>Utilities =>Remove =>Remove from all projects**

On selecting the **remove all** option, the **remove all models from project** panel is displayed.



The fields and buttons used in this panel have the following functions.

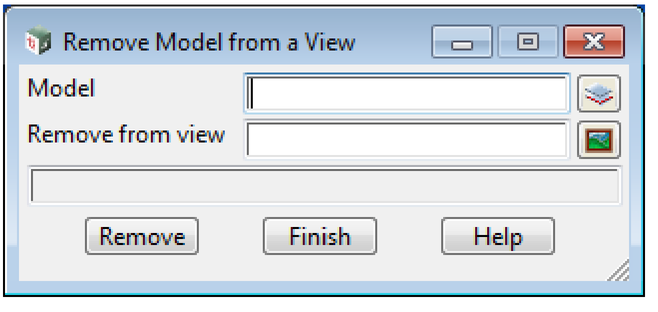
Field Description	Type	Defaults	Pop-Up
Remove	button		

after selecting this button, all models in the working project will be removed. Next, all views will be refreshed. Finally, unless an error occurs, the panel will be removed.

13.8.4.3 Remove From View

Position of option on menu: **Model =>Utilities =>Remove =>Remove from view**

On selecting the **remove from view** option, the **remove model from a view** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model	input		available models

name of the model to be removed from a view.

Field Description	Type	Defaults	Pop-Up
Remove from view	input		available views

name of the view from which the model given in the model *field will be removed.*

Field Description	Type	Defaults	Pop-Up
Remove	button		

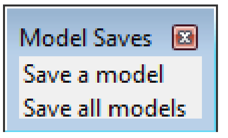
remove the model given in the model *field from the view given in the* remove from view *field.*

13.8.5 Save

Position of menu: **Model =>Utilities =>Save**

Models can be saved on disk so that they can be used for future project work or in other projects.

The **model saves** walk-right menu is



The options in the menu will now be described.

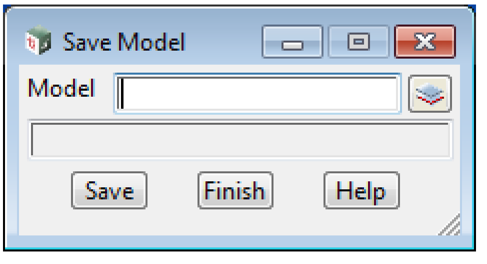
For the option Save a model go to 13.8.5.1 Save a Model

Save all models 13.8.5.2 Save All Models

13.8.5.1 Save a Model

Position of option on menu: **Model =>Utilities =>Save =>Save a model**

On selecting the **save a model** option, the **save model** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model	input		available models

name of the model to be saved.

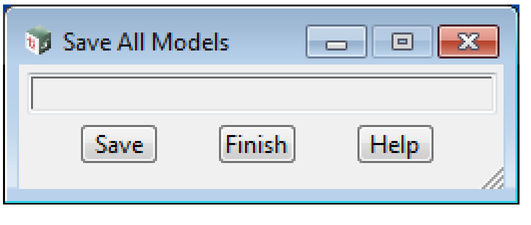
Save	button		

after selecting this button, the model given in the model field will be saved to disk.

13.8.5.2 Save All Models

Position of option on menu: **Model =>Utilities =>Save =>Save all models**

On selecting the **save all models** option, the **save all models** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Save button

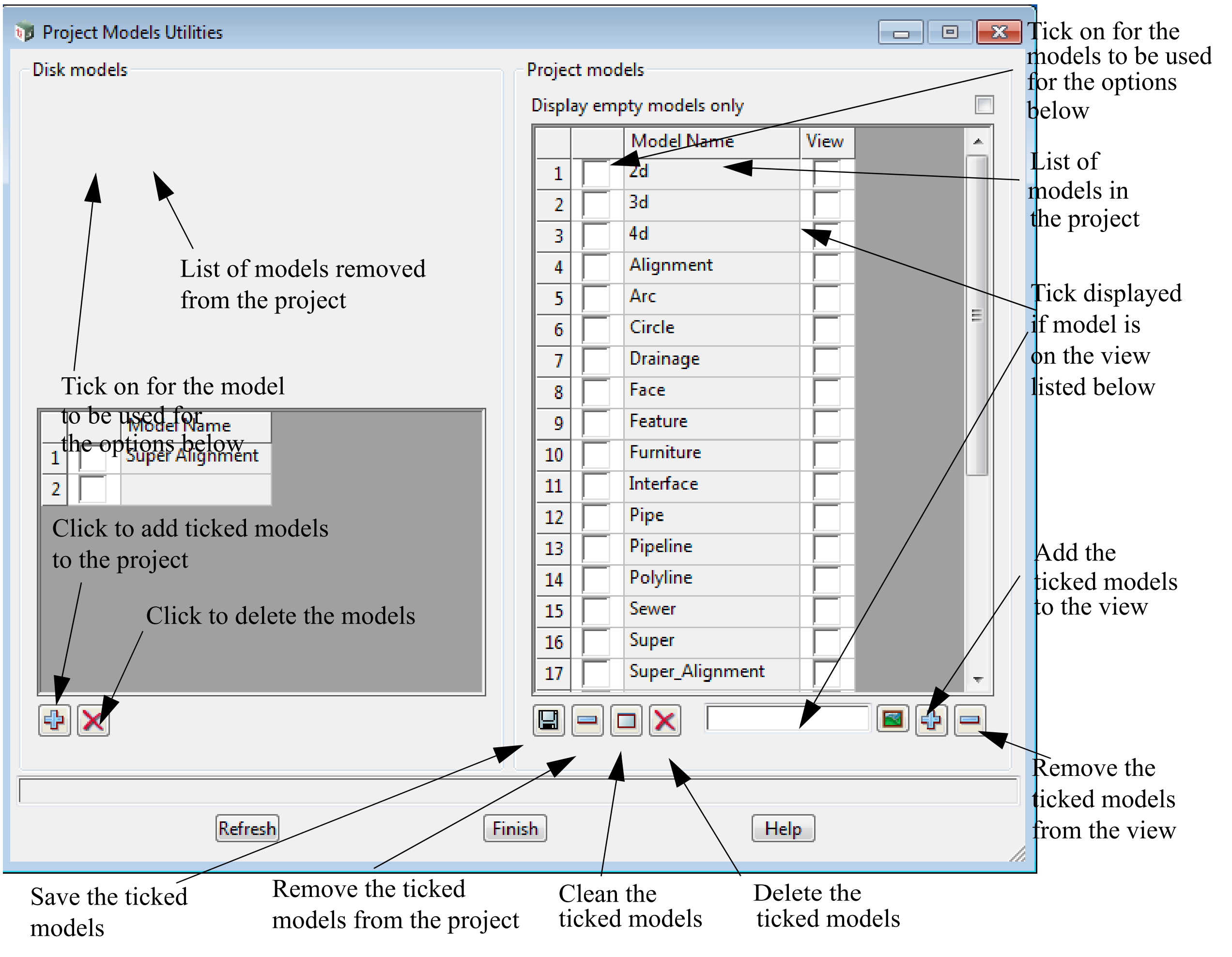
after selecting this button, all models in the working project that have been modified since they were last saved, will be saved to disk. Unless an error occurs, the panel will be removed after the saving is completed.

13.8.6 Project Model Utilities

Position of option on menu: **Model =>Utilities =>Project model utilities**

The **Project Model Utilities** panels can add/remove models from the project, clean and delete models, add/remove models from views.

On selecting **Project model utilities**, the **Project Model Utilities** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description Type DefaultsPop-Up

Disk Models section

options to add removed models to the project and delete removed models

Project Models section

options to save/remove/clean/delete models, add/remove models from a view

Display empty models only tick box

*if **ticked**, only empty models are listed.*

Refresh button

refresh the model properties displayed in the panel

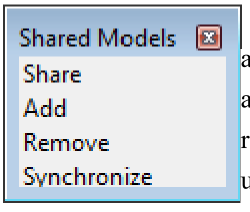
13.9 Sharing

Position of menu: **Model =>Sharing**

Sharing allows models from a project (the server project) to be added to other projects (client projects).

Before any models can be added to a client project, they must first be tagged in the server project as allowed to be shared.

The **Sharing** walk-right menu is



allow models in this project to used by other projects
add a shared model from another project
remove a model shared from another project
updated models shared from other projects

For the option *Share*, go to	13.9.1 Share Models
Add	13.9.2 Add Shared Models
Remove	13.9.3 Remove Shared Models
Synchronize	13.9.4 Synchronize Shared Models

13.9.1 Share Models

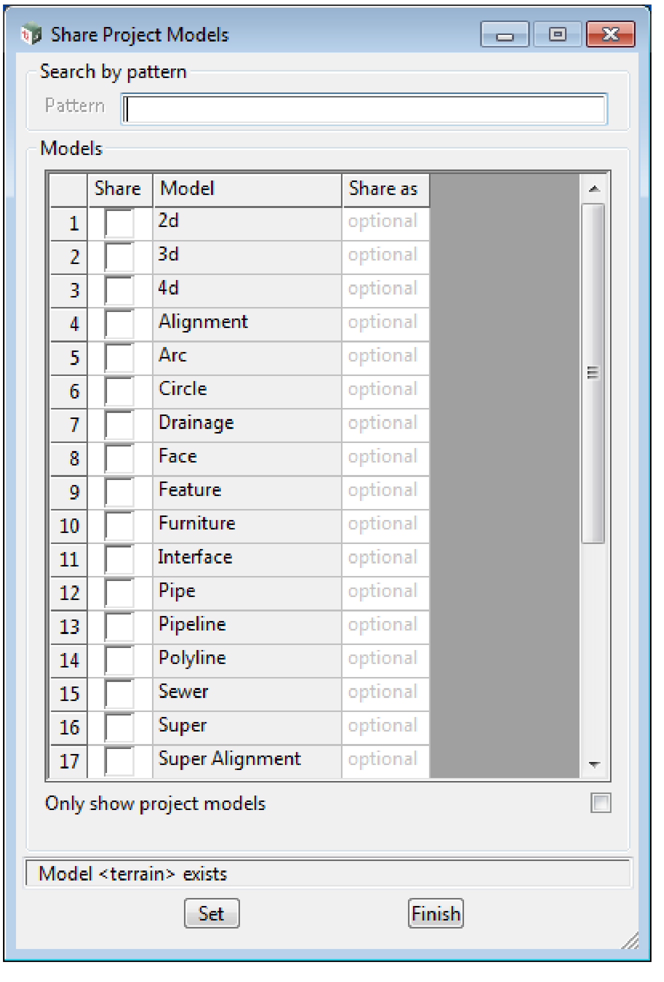
Position of option on menu: **Model =>Sharing =>Share**

Before models in a server project can be added to client projects, the models must be made available for sharing in the server project by using the **Share** option.

All models that have been made available for sharing are displayed in model lists in a colour defined by an environment variable. The default colour is a yellow (RGB of (255,166,0))

This option also can reverse the process. That is, remove the availability of a model for sharing. If this is done then clients who have shared this model will be warned.when starting up their project and/or synchronizing occurs

Selecting **Share** displays the **Share Project Models** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type DefaultsPop-Up

All the models in the project are listed in the **Share Project Models** *panel and if a model has been tagged for Sharing, a tick will be displayed in the* ***Share*** *column of the grid.*

Pattern

if a pattern is typed then all the models matching the pattern will have a tick placed in the ***Share*** *column. If * is typed then all models are ticked for sharing.*

Share grid column

tick if the model is to be made available for sharing when the **Set** *button is pressed.*

All models that have been made available for sharing are displayed in model lists in a colour defined by an environment variable. The default colour is a yellow (RGB of (255,166,0)).

Clicking RB on ***Share*** *at the top of the column brings up a menu to* ***Toggle*** *the ticks,* ***Set*** *all the ticks on,* ***Clear*** *to turn all the ticks off.*

Model grid column

this column lists all the models in the project

Share as grid column

an optional name with which the model will be shared out to clients. Leave it blank to share under the original name.

Only show project models tick box

if ticked, only the models which originated from the current project are shown.

Set button

clicking ***Set*** *marks all the models with a tick in the* ***Tick*** *column as being available for sharing. Those models without a tick will not be available for sharing.*

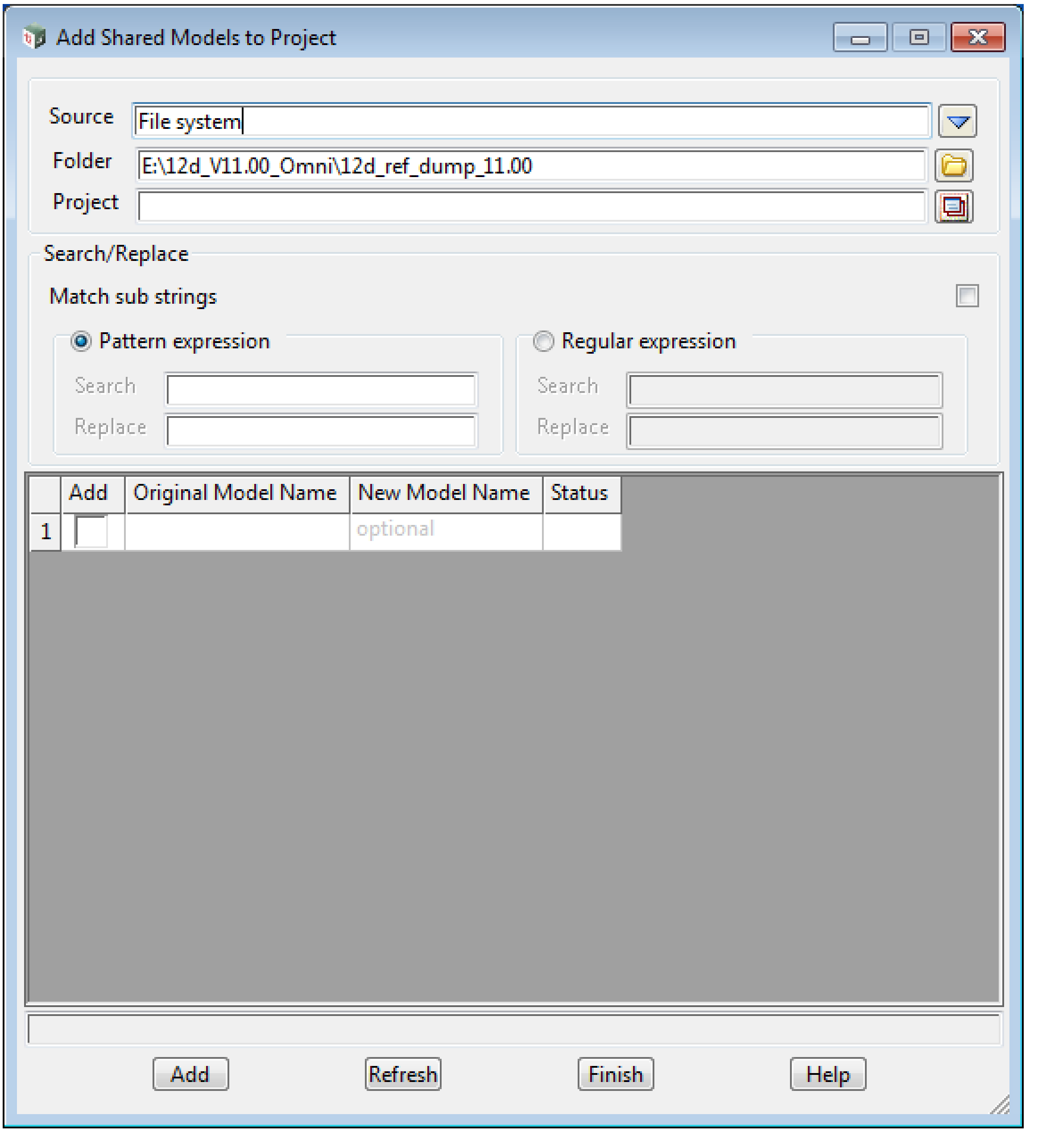
13.9.2 Add Shared Models

Position of option on menu: **Model =>Sharing =>Add**

The **Add** option is used to add shared models from a server project, to this project (a client project).

All models that have been added as shared models are displayed in model lists in a colour defined by an environment variable. The default colour is blue.

Selecting **Add** displays the **Add Shared Models to Project** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	DefaultsPop-Up
Folder	folder box	select folder

folder to look for ***12d Model*** *projects. When a folder is selected, all the* ***12d Model*** *projects in the* *folder will be displayed in a **Projects** pop-up list.*

Project folder box select folder

name of the **12d Model** *project in the folder given in the* Folder *field, to search for models marked for sharing. Once a project is selected, all the models marked for sharing from that project will be listed in the* Original Model Name *column.*

Search/Replace

section for renaming models from the selected project

Match sub strings tick box

*if **ticked**, the **Search** expression is used to match against part of each model name.*
*If **not ticked**, the **Search** expression is used to match against the entire model name.*

Pattern expression radio button

*if set on, then **Pattern** expressions given in the* Search *and* Replace *fields are used to modify model names. Pattern expressions include the standard wild card * and wild character !.*

Search input

pattern to search for in the model names. For example " exist" will select all models with a name ending with " exist"*

Replace input

replacement for the search pattern found in the model name. For example, "exist " in the Replace *field takes the matched part of the model name and adds " exist " to the front of it.*

Hence the Search pattern " exist" and Replace pattern "exist *" finds all models with names ending in " exist" and renames them with the name starting with "exist " (and the " exist" at the end of the name is dropped off).*

Regular expression radio button

*if set on, then **Regular** expressions given in the* Search *and* Replace *fields are used to modify model names.*

Search input

regular expression to search for in the model names.

Replace input

replacement for the search expression found in the model name.

Model Names Grid

The models available for sharing in the selected project are shown in the Original Model Name *column. Any renaming by the Search and Replace expressions are shown in the* New Model Name *column. Note that if the model already exists in the current project, then the cell for that model in* Original Model Name *column will be displayed in yellow.*

Add tick box

if ticked, the model will be added to this project as a shared model.
If not ticked, the model will not be added to this project as a shared model.

*Clicking RB on **Add** at the top of the column brings up a menu to **Clear** which turns all the ticks off.*

Note - a shared model that has been previously added can only be removed by using the **Model =>Sharing =>Remove** *option.*

Original Model Name column

this column lists all the models in the server project available for sharing.

New Model Name column

*if **non-blank**, the new name to be given to shared model in this project.*
If blank, the original model name is used.

The New Model Names *can be from applying the* Search *and* Replace, *or by just typing them in.*

Status output column

displays if their has been a match *or* no match *for the search/replace for renaming models*

Add button

add to this project (a client project) as a shared model, the ticked models given in the Original Model Name *field from the project given in the* Project *field.*

Refresh button

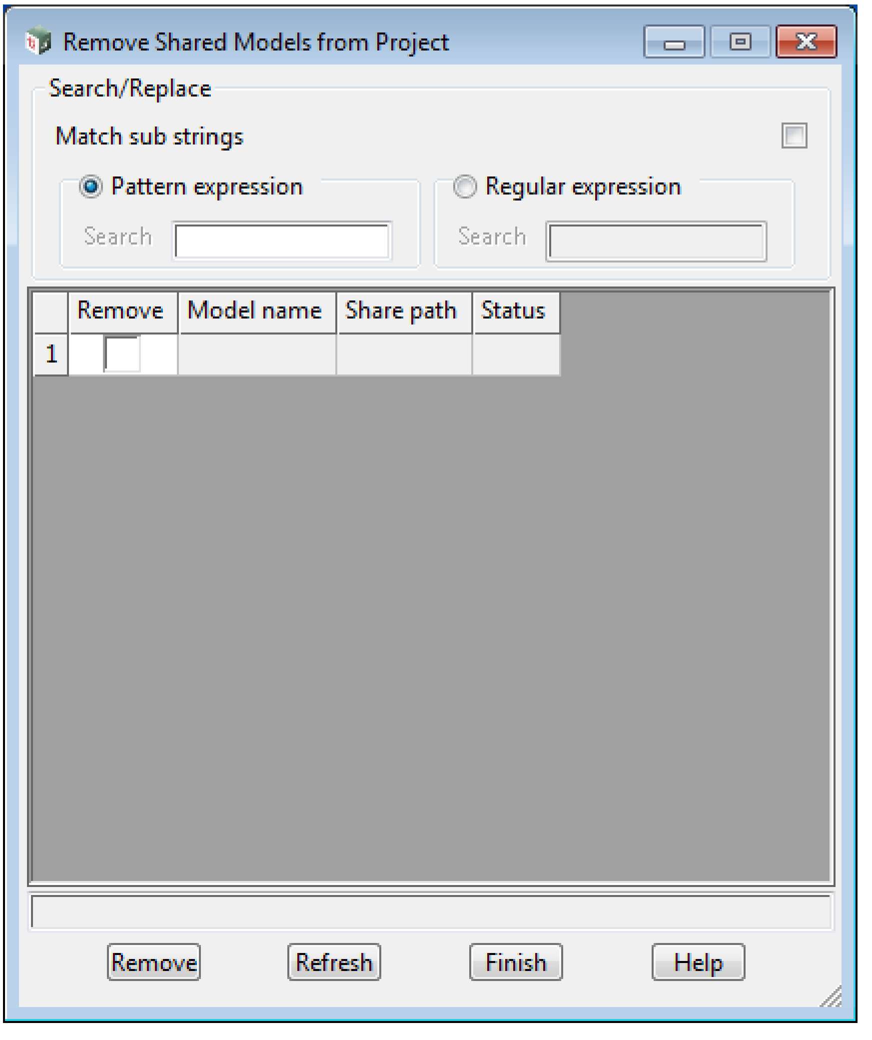
refresh the list of all models available for sharing in the selected server project

13.9.3 Remove Shared Models

Position of option on menu: **Model =>Sharing =>Remove**

The **Remove** option is used to remove shared models from the.project. The shared models would have been previously added to the project with the **Model =>Sharing =>Add** option.

Selecting **Remove** displays the **Remove Shared Models from Project** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type DefaultsPop-Up

Search/Replace

section for selecting by expressions, models in this project that are shared from other projects

Match sub strings tick box

*if **ticked**, the **Search** expression is used to match against part of each model name.*
*If **not ticked**, the **Search** expression is used to match against the entire model name.*

Pattern expression radio button

*if set on, then **Pattern** expressions given in the* Search *fields are used to select shared model names. Pattern expressions include the standard wild card * and wild character !.*

Search input

pattern to search for in the model names. For example " exist" will select all models with a name ending with " exist"*

Regular expression radio button

*if set on, then **Regular** expressions given in the* Search *field are used to select shared model names.*

Search input

regular expression to search for in the shared model names.

Model Names Grid

list of all shared models added to the project.

Remove tick box

if ticked, the model will be removed from this project as a shared model.
If not ticked, the model will not be removed.

*Clicking RB on **Remove** at the top of the column brings up a menu to **Clear** which turns all the ticks off.*

Model Name column

this column lists all the models added as shared models from other server projects

Share path

this column lists the path name to the server project and the original model name in the server project

Status column

displays if their has been a match *or* no match *for the* Search *for selecting shared models*

Remove button

*clicking **Remove** removes as shared models from this project, all the models with a tick.*

Refresh button

*clicking **Refresh** refreshes the list of all shared models previously added to the project*

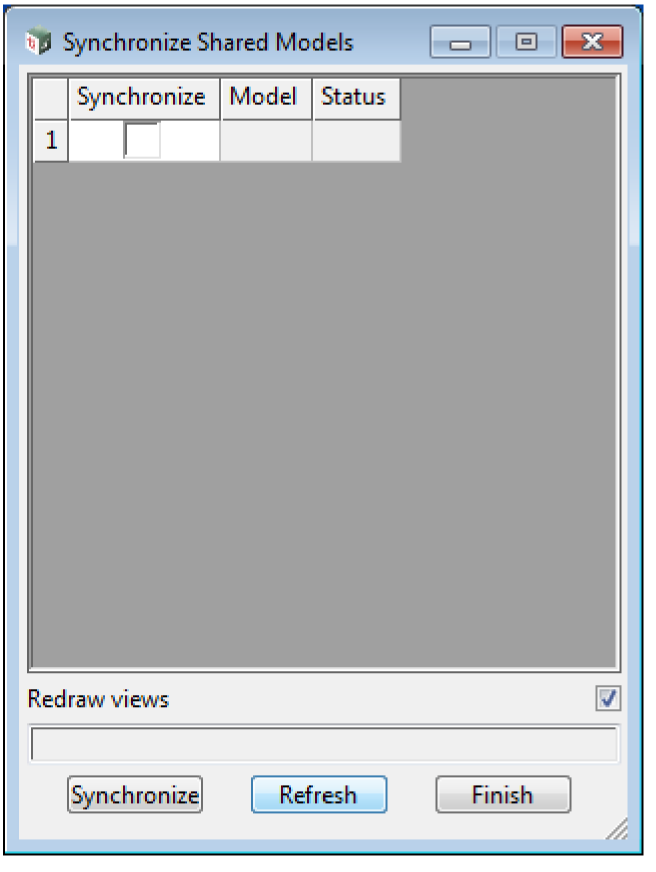
13.9.4 Synchronize Shared Models

Position of option on menu: **Model =>Sharing =>Synchronize**

The **Synchronize** option is used to updated any added shared models. The shared models would have been previously added to the project with the **Model =>Sharing =>Add** option.

Three environment variables control the synchronization of updates of added shared models and tins. For more information on synchronizing, go to the section 7.6.13.4 Project Share Settings.

Selecting **Synchronize** displays the **Synchronize Shared Models** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type DefaultsPop-Up

Synchronize tick boxes in grid column

*if ticked and the model has changed in the server project, then it will be re-copied from the server project when the **Synchronize** button is pressed.*

*Clicking RB on **Copy** at the top of the column brings up a menu to **Toggle** the ticks, **Set** all the ticks on, **Clear** to turn all the ticks off.*

Model grid column

this column lists all the models in this project that have been added from other (server) projects

Synchronize button

*clicking **Synchronize** re-copies any models with a tick from the server projects*

Status

*Status will either be **Ok** or **No Longer Shared** to reflect the status of the model on that row.*

Redraw views tick box

Whether or not to redraw all the views any of the synchronized models are on.

Refresh button

*clicking **Refresh** refreshes the list of all shared models previously added to the project*

13.10 Clean

Position of option on menu: **Model => Clean**

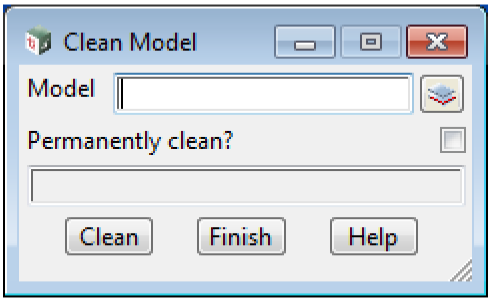
The **clean** option is used to delete all the strings in a model but the model itself is not deleted. The model is still attached to any views it had been added to.

To help protect the user against disasters, when a model is selected for cleaning, a **yes-no** pop-up menu is invoked to confirm that the user did intend cleaning out the model.

Wild cards (*) and wild characters (!) can be used to select the models to clean and then a list of selected models is displayed for confirmation.

If cleaning is confirmed, all the strings in the selected models are deleted.

Selecting **Clean** displays the **Clean Model** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model	model box		available models

*name of the model to be cleaned out. Wild cards * and characters ? can be used to give a list of models to be cleaned.*

Permanently clean?	tick box		

if ticked, the cleaned strings will not go to the trash bin but will be permanently deleted.

Clean	button		

after selecting this button, all strings in the model given in the model *field will be deleted. A* **yes-no** *pop-up is used to confirm that cleaning is required.*

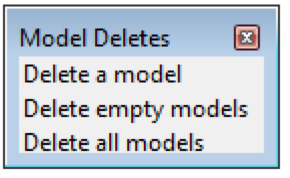
13.11 Delete

Position of menu: **Model => Delete**

Using the **delete** option, models can be deleted from disk so that they no longer can be accessed or take up disk space.

To help protect the user against disasters, when a model is selected for deletion, a **yes-no** pop-up menu is used to confirm that deleting the model is intended. If deletion is confirmed, the selected model is deleted from the project and the disk. For extra security, there is also a ***Trash Bin*** (see 7.6.15 Trash Bin) where models can automatically be moved to on delete before being permanently deleted from disk.

The model **delete** walk-right menu is



For the option *Delete a model*, go to	13.11.1 Delete a Model
Delete empty models	13.11.2 Delete Empty Models
Delete all models	13.11.3 Delete All Models

13.11.1 Delete a Model

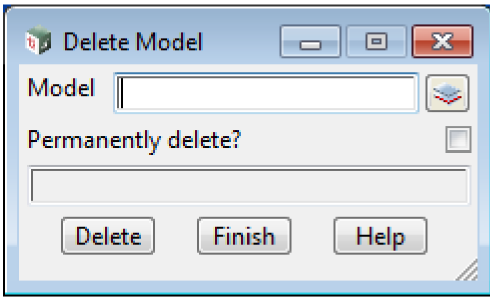
Position of option on menu: **Model =>Delete =>Delete a Model**

The **delete a model** option is used to delete a model in the working project.

Wild cards (*) and wild characters (!) can be used to select the models to delete and then a list of selected models is displayed for confirmation.

If a ***12d Model*** *trash bin* is being used, the deleted models may be moved to the trash bin rather than deleted from disk (see 7.6.15 Trash Bin).

Selecting **Delete a model** brings up the **Delete Model** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model	input		available models

*name of the model to be deleted. Wild cards * and characters ? can be used to give a list of models to be deleted.*

Permanently delete	tick box		

if ticked, the deleted models will not go to the trash bin but will be permanently deleted from disk.

Delete	button		

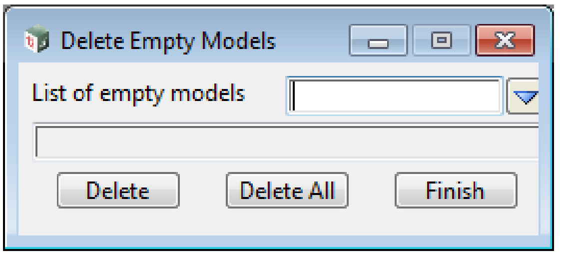
after selecting this button, the model given in the model field will be deleted from the computer disk. A **yes-no** *pop-up is used to confirm that deletion is required.*

13.11.2 Delete Empty Models

Position of option on menu: **Model =>Delete =>Delete Empty Models**

The **delete empty models** option displays a list of all the empty models in the project and can delete selected empty models or all of them.

Selecting **Delete empty models** brings up the **Delete Empty Models** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
List of empty models	models		available models

the pop-up brings up a list of all the empty models in the project. A model can be selected from the list and it will be deleted after selecting the **delete** *button.*

Delete	button		

delete the model given in the list of empty models *field.*

Delete all	button		

delete all the empty models in the project.

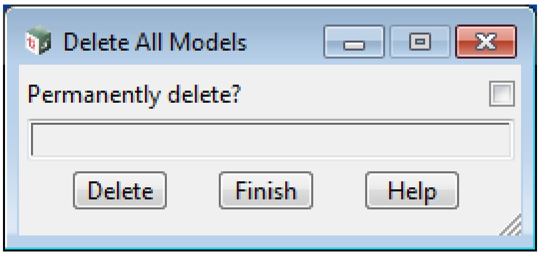
13.11.3 Delete All Models

Position of option on menu: **Model =>Delete =>Delete All Models**

The **delete all models** option will delete all the models in the project. It does not delete models that are in the project area but not yet added to the project.

If a ***12d Model*** *trash bin* is being used, the deleted models may be moved to the trash bin rather than deleted from disk (see 7.6.15 Trash Bin).

On selecting the **Delete all models** option, the **Delete All Models** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Permanently delete	tick box		

if ticked, the deleted models will not go to the trash bin but will be permanently deleted from disk.

Delete	button		

after selecting this button, a **yes-no** *pop-up is used to confirm that deletion is required. If it is, all models in the working project will be deleted from disk. All views will be refreshed and unless an error occurs, the panel will be removed.*

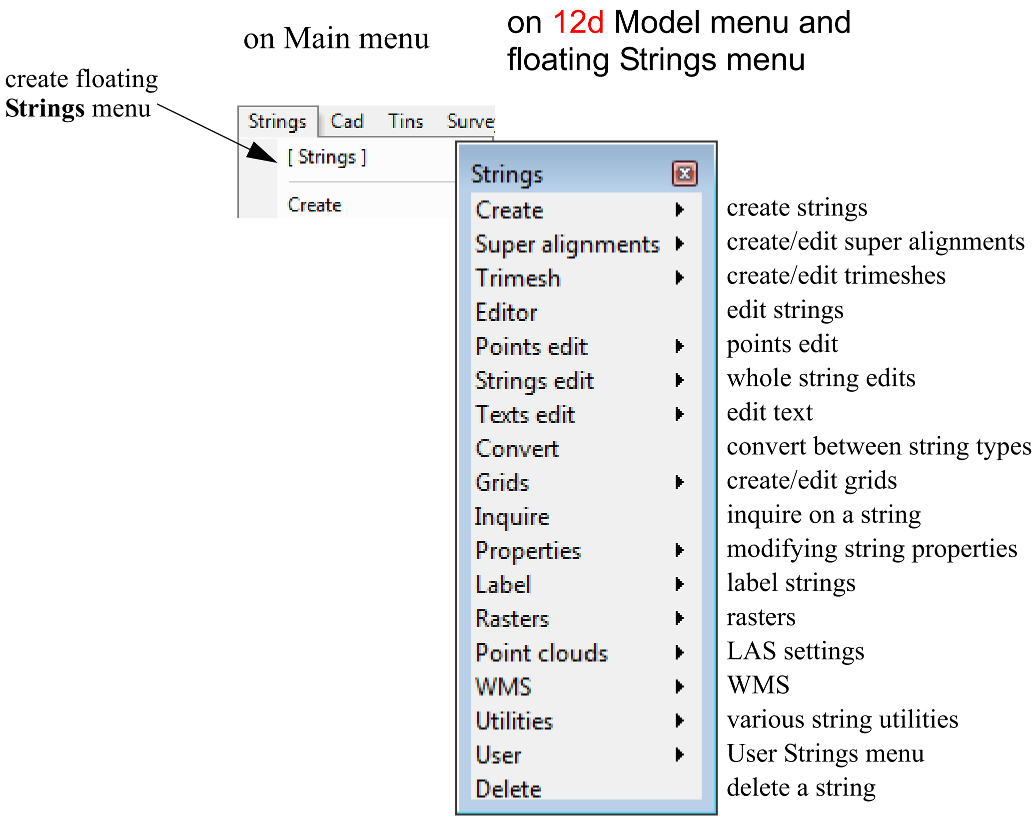
14 Strings

Position of menu: **Strings**

Strings are the basic modelling elements of 12d Model.

The Strings walk-right menu contains options to create, edit and manipulate strings in a variety of ways including copying, moving strings between models and deleting strings.

The **Strings** walk-right menu is



For the options:

Create	14.1 Create
Super alignments	14.2 Super Alignments
Trimesh	14.3 Trimesh
Editor	14.4 Editor
Points edit	14.5 Points Edit
Strings edit	14.6 Strings Edit
Texts edit	14.8 Texts Edit
Convert	14.9 Convert
Grids	14.10 Grids
Inquire	14.11 Inquire
Properties	14.12 Properties
Label	14.13 Label
Rasters	14.14 Rasters
Point clouds	14.15 Point Clouds
WMS	14.16 WMS

Note

For the floating menu, the **Strings** option has another mode of operation. Rather than moving onto the walk-right arrow, if LB is clicked when the **Strings** button is highlighted on the 12d Model menu, the **String inquire** panel is displayed on the screen. Unfortunately this does not work on the *Main* menu.

14.1 Create

Position of option on menu: **Strings =>Create**

The **Create** option is used to produce new strings. If a string already exists, the **Editor** option is used to modify it.

The **Create** walk-right menu contains options to create 2d, 3d, 4d, super alignment, circles, arcs, feature, pipe, polyline, super and text strings.

Each string type has its own special information so by choosing the appropriate type to be created, only information needed for that type of string is asked for.

Note that the 2d, 3d, 4d, pipe and polyline strings are actually super strings but automatically set some of the properties of the super string. For example a 2d string is a super string with just one constant z-value. Any of these strings can be made a full super string by clicking on the *Advanced Mode* icon in the string editor.

Once a string has been created, it can be converted to most of the other string types using the string **Convert** option.

To create a new string of the same type and with similar header information as an existing string, the **same as** option is selected from the **String Create** menu.

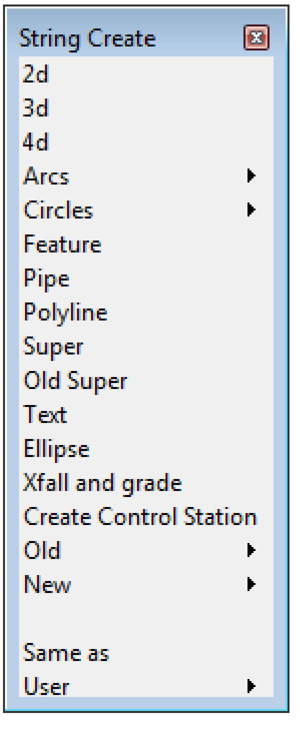
Notes

1. Depending on its breakline (point-line) type, a string with default style ("1") is displayed with crosses at each of its vertex (point type) or with straight lines joining the vertices (line type).

   Linestyles can be defined that drawn lines between the points even though the breakline type is **point**.

2. The string vertices are also called intersection points (IP's).

The **String Create** walk-right menu is



The string creation process is similar for each string type and for editing strings as well.

For the option *2d,* go to 14.1.1 Create - 2d Super

3d 14.1.2 Create - 3d Super

4d	14.1.3 Create - 4d Super
Super alignment	14.1.4 Create Arcs
Arcs	14.1.4 Create Arcs
Circles	14.1.5 Create - Circles
Feature	14.1.6 Create - Feature
Pipe	14.1.7 Create - Pipe Super
Polyline	14.1.8 Create - Polyline Super
Super	14.1.9 Create - Super
Text	14.1.10 Create - Text
Ellipse	14.1.11 Create - Ellipse
Xfall and grade	20.8.11 String by Xfall and Grade
Control stations	14.1.12 Create - Control Stations
Old	14.20.1 Old
New	14.1.13 New
Same as	14.1.14 Create - Same As

14.1.1 Create - 2d Super

Position of option on menu: **Strings =>Create =>2d**

A *2d super string* is a *super string* with the restriction that it has:

(a) vertices of (x,y) co-ordinates joined by straight segments only

(b) One z-value for the entire string

That is, a 2d string consists of a series of (x,y) vertices joined only by straight segments, and all with the same z-value (height). 2d strings are often referred to as “contour strings” because contours are the most common example of a string with a constant height.

The Many strings tick-box is used when more than one string of the same type is to be created, If many strings is set to *tick*, when the current string creation is **Finished** or **Quit**, a new **Create Super 2d String** panel is placed on the screen with the same information in it as the string just created. If any of the information needs to be modified for the new string, simply change it in the **Create Super 2d String** panel fields before selecting the **Create** button for the new string. Hence a new string of the same type can be created without going back to the **Create** menu.

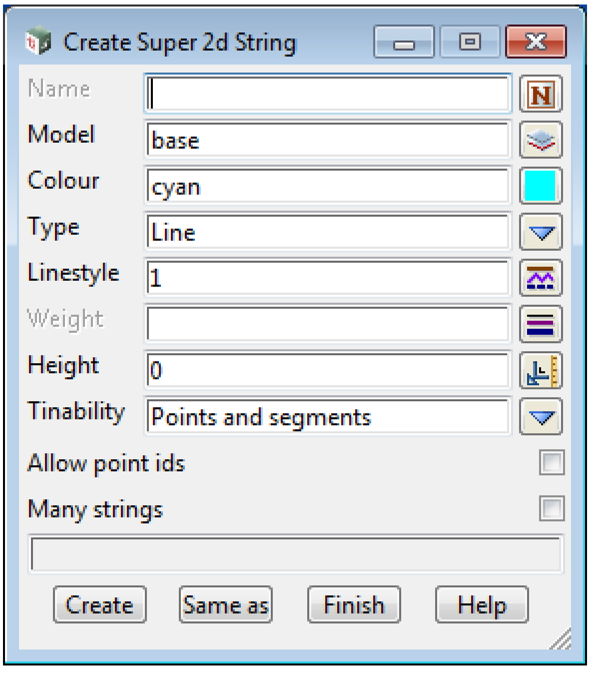
The Same as button is used to obtain information from and existing string (not necessarily of the same type) and pipe it into the name, model, colour, type, linestyle, weigh of the **Create Super 2d String** panel.

From *12d Model 8* onwards, the default is to create *2d super strings* rather than the *2d strings* used up to *12d Model 7*. A *2d super string* is a *super string* with *Constant height*. That is, there is only one z-value for the entire string. The advantage is using a *2d super string* is that all the CAD options will work for it, vertices and segments can be assigned tinability and it can have point id’s for setout.

The older *2d string* create options are still available under

Strings =>Create =>Old (go to the section 14.20.1.2 Create - 2d (pre V8))

On selecting **2d** the **Create Super 2d String** panel is displayed.



The default values for the panel fields are taken from the *CAD Controlbar (see* 15.1 Controlbars*)*

To create a *new 2d super string* (a *2d super string* is a *super string* with the *constant height* flag set on), the panel fields are filled in and the **Create** button selected.

The fields and buttons used in the **Create Super 2d String** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input	from CAD controlbar	
the name of the new string			
Model	model box	from CAD controlbar	available models
name of the model that the new string is in			
Colour	colour box	from CAD controlbar	available colours
the colour of the new string			
Type	choice box	line	line, point
breakline type (point-line type) of the string			
Linestyle	input	from CAD controlbar	available line styles
line style of the string			
Weight	input	from CAD controlbar	
thickness of the string			
Height	input	from CAD controlbar	
for a 2d super string, there is only one height (z-value) for the entire string.			
Tinability	choice box	Points and segments	Points and segments Points only Not tinable

*if **Vertices and segments**, all the vertices and segments of the string are set to **tinable**.
If **Vertices only**, all the vertices are set to tinable and the segments to not tinable.
If **Not tinable**, all the vertices or segments are set to **not tinable**.*

Allow point ids tick box

*if **ticked**, the 2d super string can have point ids for each vertex.
If **not ticked**, the 2d super string will not have point ids. This can be reversed if point ids are required in the future.*

Many strings tick box

if ticked then after the current string is finished, a new create panel is placed on the screen with all the same values for the panel fields as the current string.

Create button

After the **Create** *button is chosen, the* **Edit 2d** *menu is displayed.*

Same as button

After the **Same as** *button is chosen, another string is selected and information about it is used for the fields in this panel.*

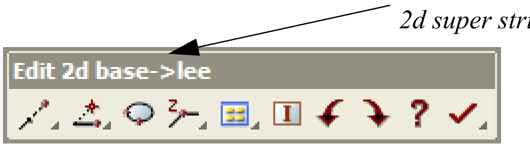
Finish button

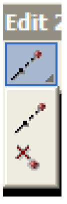
*end the option, **don't** proceed to the edit stage.*

Start Edit - 2d Super

On selecting the **Create** button in the **Create Super 2d String** panel the **Edit 2d** menu is placed on the screen. The **Edit 2d** menu for a *2d super string* is

model ->string_name *for the 2d super string being edited*





append or prepend an IP

delete an IP

To create a new *2d super string*, select the **Append** icon.

The **Append** option is used to add vertices to either end of an existing string, or in the case of a new string, places the first vertex and then begins appending vertices to the first vertex.

For all 2d strings, a cross is then drawn in each plan view that the string's model is on, and the cross follows the cursor around the screen. If the string's model is not added to any plan view, the model is automatically added to all plan views.

After the cross is on the screen (moving with the cursor), clicking LB and accepting with MB selects the first vertex of the string (using the appropriate snaps).

The string is then drawn from the first vertex to the cursor position, which represents the second vertex of the string. Clicking LB and accepting with MB selects the second string vertex and the process repeats for subsequent string vertices.

Now that the string is created, all the edit option on the string's **Edit 2d** menu are usable.

The options in the **Edit 2d** menu are not only used for placing the initial vertices of the string, but for editing the string once it is created. Since the **Append** and other options in the **Edit 2d** menu are identical to the options used when editing an existing string, they will be discussed in detail in the string **Editor** section.

For full information in the 2d Editor, go to the section 14.4.2 Edit 2d

14.1.2 Create - 3d Super

Position of option on menu: **Strings =>Create =>3d**

A 3d super string consists of a series of (x,y,z) vertices, joined only by straight segments.

The difference between a 2d and a 3d string is that for a 3d string, the z-value (height) can vary at each string vertex, whereas a 2d string has a constant height for the entire string. Hence a 2d string is simply a special case of a 3d string where the heights at all the vertices are the same.

Creating and editing a 3d string is very similar to a 2d string. The only major difference is that a height is required at each string vertex.

The Many strings tick-box is used when more than one string of the same type is to be created, If many strings is set to *tick*, when the current string creation is **Finished** or **Quit**, a new **Create Super 3d String** panel is placed on the screen with the same information in it as the string just created. If any of the information needs to be modified for the new string, simply change it in the **Create Super 3d String** panel fields before selecting the **Create** button for the new string. Hence a new string of the same type can be created without going back to the **Create** menu.

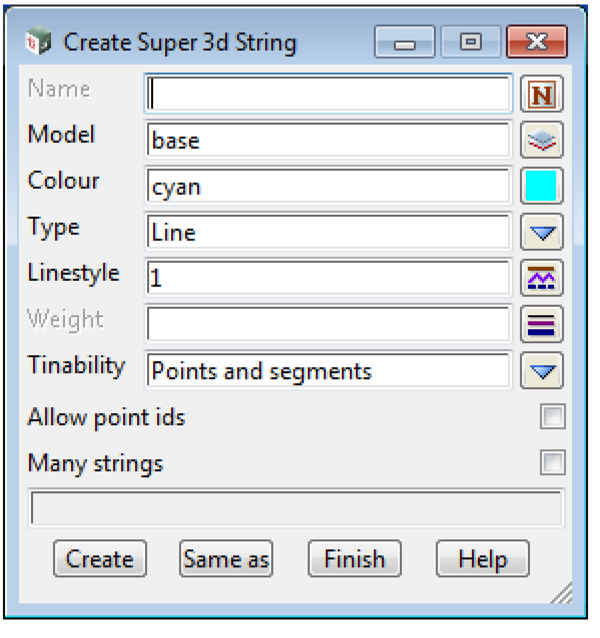
The Same as button is used to obtain information from and existing string (not necessarily of the same type) and pipe it into the name, model, colour, type, linestyle and weight of the **Create Super 3d String** panel.

From *12d Model 8* onwards, the default is to create *3d super strings* rather than the *3d strings* used up to *12d Model 7*. The advantage is using a *3d super string* is that all the CAD options will work for it, vertices and segments can be assigned tinability and it can have point id's for setout.

The older *3d string* create option is still available under

Strings =>Create =>Old (go to the section 14.20.1.3 Create - 3d (pre V8))

On selecting the **3d** option, the **Create Super 3d String** panel is displayed.



The default values for the panel fields are taken from the *CAD Controlbar (see* 15.1 Controlbars*)*

To create a *new 3d super string*, the panel fields are filled in and the **Create** button selected.

The fields and buttons used in the **Create Super 3d String** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input	from CAD controlbar	

the name of the new string

Model	model box	from CAD controlbar	available models

name of the model that the new string is in

Colour	colour box	from CAD controlbar	available colours

the colour of the new string

Type	choice box	line	line, point

breakline type (point-line type) of the string

Linestyle	input	from CAD controlbar	available line styles

line style of the string

Weight	input	from CAD controlbar	

thickness of the string

Tinability	choice box	Points and segments	Points and segments Points only Not tinable

*if **Vertices and segments**, all the vertices and segments of the string are set to **tinable**.*
*If **Vertices only**, all the vertices are set to tinable and the segments to not tinable.*
*If **Not tinable**, all the vertices or segments are set to **not tinable**.*

Allow point ids tick box

*if **ticked**, the super string can have point ids for each vertex.*
*If **not ticked**, the super string will not have point ids. This can be reversed if point ids are required in the future.*

Many strings tick box

if ticked then after the current string is finished, a new create panel is placed on the screen with all the same values for the panel fields as the current string.

Create button

after the **Create** *button is chosen, the* **Edit 3d** *menu is displayed*

Same as button

after the **Same as** *button is chosen, another string is selected and information about it is used for the fields in this panel*

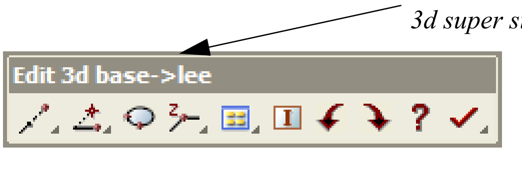
Finish button

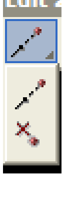
*end the option, **don't** proceed to the edit stage*

Start Edit - 3d Super

On selecting the **Create** button in the **Create Super 3d String** panel the **Edit 3d** menu is placed on the screen. The **Edit 3d** menu for a *3d super string* is

model ->string_name *for the 3d super string being edited*





append or prepend an IP

delete an IP

To create a new *3d super string*, select the **Append** icon.

The **Append** option is used to add vertices to either end of an existing string, or in the case of a new string, places the first vertex and then begins appending vertices to the first vertex.

For all 3d strings, a cross is then drawn in each plan view that the string's model is on, and the cross follows the cursor around the screen. If the string's model is not added to any plan view, the model is automatically added to all plan views.

After the cross is on the screen (moving with the cursor), clicking LB and accepting with MB selects the first vertex of the string (using the appropriate snaps).

The string is then drawn from the first vertex to the cursor position, which represents the second vertex of the string. Clicking LB and accepting with MB selects the second string vertex and the process repeats for subsequent string vertices.

Now that the string is created, all the edit option on the string's **Edit 3d** menu are usable.

The options in the **Edit 3d** menu are not only used for placing the initial vertices of the string, but for editing the string once it is created. Since the **Append** and other options in the **Edit 3d** menu are identical to the options used when editing an existing string, they will be discussed in detail in the string **Editor** section.

For full information in the 3d Editor, go to the section 14.4.3 Edit 3d

14.1.3 Create - 4d Super

Position of option on menu: **Strings =>Create =>4d**

A 4dsuper string consists of a series of (x,y,z) co-ordinates joined only by straight segments, and a text label at each vertex.

Creating and editing a 4d string is very similar to a 3d string.- the major difference is that a height and a text label is required at each string vertex.

The Many strings tick-box is used when more than one string of the same type is to be created, If many strings is set to *tick*, when the current string creation is **Finished** or **Quit**, a new **Create Super 4d String** panel is placed on the screen with the same information in it as the string just created. If any of the information needs to be modified for the new string, simply change it in the **Create Super 4d String** panel fields before selecting the **Create** button for the new string. Hence a new string of the same type can be created without going back to the **Create** menu.

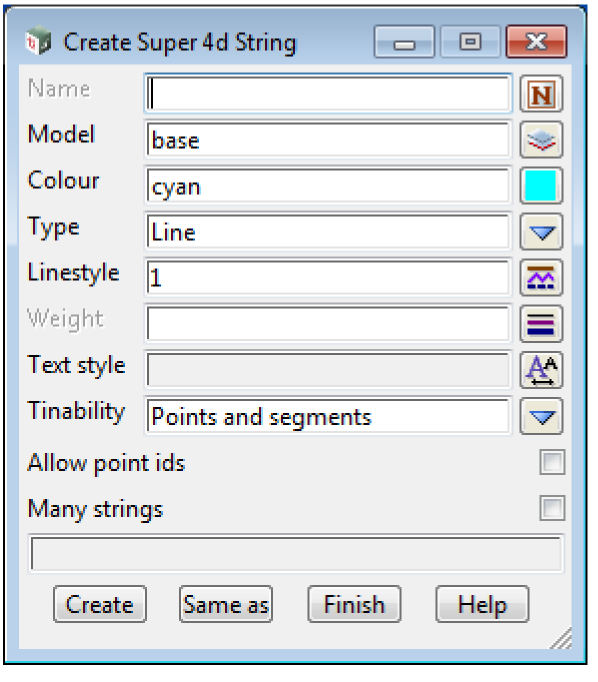
The Same as button is used to obtain information from and existing string (not necessarily of the same type) and pipe it into the name, colour, model, type, linestyle and weight of the **Create Super 4d String** panel.

From *12d Model 8* onwards, the default is to create *4d super strings* rather than the *4d strings* used up to *12d Model 7*. The advantage is using a *4d super string* is that all the CAD options will work for it, vertices and segments can be assigned tinability and it can have point id's for setout.

The older *4d string* create option is still available under

Strings =>Create =>Old (go to the section 14.20.1.4 Create - 4d (Pre V8))

On selecting the **4d** option, the **Create Super 4d String** panel is displayed.



The default values for the panel fields are taken from the *CAD Controlbar (*see 15.1 Controlbars*)*

To create a *new 4d super string*, the panel fields are filled in and the **Create** button selected.

The fields and buttons used in the **Create Super 4d String** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input	from CAD controlbar	
the name of the new string			
Model	model box	from CAD controlbar	available models
name of the model that the new string is in			

Colour colour box from CAD controlbar available colours

the colour of the new string

Type choice box line line, point

breakline type (point-line type) of the string

Linestyle linestyle box from CAD controlbar available line styles

line style of the string

Weight input from CAD controlbar

thickness of the string

Text style textstyle box

textstyle to use for text

Tinability choice box Points and segmentsPoints and segments
Points only
Not tinable

*if **Vertices and segments**, all the vertices and segments of the string are set to **tinable**.*
*If **Vertices only**, all the vertices are set to tinable and the segments to not tinable.*
*If **Not tinable**, all the vertices or segments are set to **not tinable**.*

Allow point ids tick box

*if **ticked**, the super string can have point ids for each vertex.*
*If **not ticked**, the super string will not have point ids. This can be reversed if point ids are required in the future.*

Many strings tick box

if ticked then after the current string is finished, a new create panel is placed on the screen with all the same values for the panel fields as the current string.

Create button

after the **Create** *button is chosen, the* **Edit 4d** *menu is displayed*

Same as button

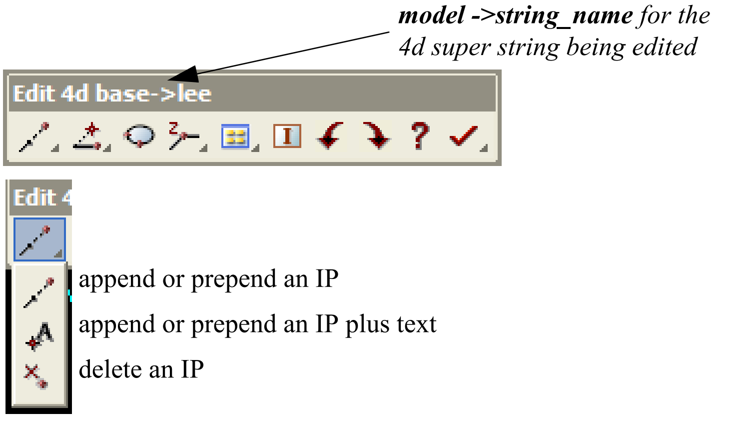
after the **Same as** *button is chosen, another string is selected and information about it is used for the fields in this panel*

Finish button

*end the option, **don't** proceed to the edit stage*

Start Edit - 4d Super

On selecting the **Create** button in the **Create Super 4d String** panel the **Edit 4d** menu is placed on the screen. The **Edit 4d** menu for a *4d super string* is



To create a new *4d super string*, select the **Append** or **Append** + **Text** icon.

The **Append** and or **Append** + **Text** option are used to add vertices to either end of an existing string, or in the case of a new string, places the first vertex and then begins appending vertices to the first vertex. **Append** + **Text** also asks for text at each vertex.

For all 4d strings, a cross is then drawn in each plan view that the string's model is on, and the cross follows the cursor around the screen. If the string's model is not added to any plan view, the model is automatically added to all plan views.

After the cross is on the screen (moving with the cursor), clicking LB and accepting with MB selects the first vertex of the string (using the appropriate snaps).

If **Append** + **text** was selected, the user will then be prompted for the text to be placed at that vertex.

The string is then drawn from the first vertex to the cursor position, which represents the second vertex of the string. Clicking LB and accepting with MB selects the second string vertex and the process repeats for subsequent string vertices.

Now that the string is created, all the edit option on the string's **Edit 4d** menu are usable.

The options in the **Edit 4d** menu are not only used for placing the initial vertices of the string, but for editing the string once it is created. Since the **Append** and other options in the **Edit 4d** menu are identical to the options used when editing an existing string, they will be discussed in detail in the string **Editor** section.

For full information in the 4d Editor, go to the section 14.4.4 Edit 4d

14.1.4 Create Arcs

Position of menu: **Strings =>Create =>Arcs**

The **Create arcs** option is used to create 12d Model arcs.

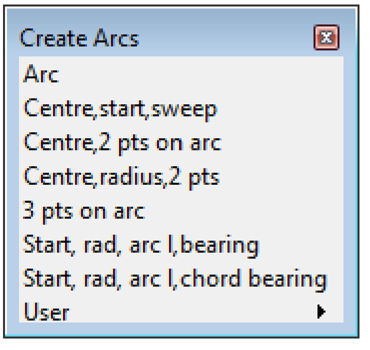
A 12d Model arc consists of a radius, a centre point, and a start and end point on the arc.

The radius can be positive or negative. If the radius is positive, the arc is drawn from the start-point to the end-point in a clockwise direction. If the radius is negative, the arc is drawn from the start-point to the end-point in counter-clockwise direction.

The height at the start point and at the end point can be different. In that case, the arc is still a plan arc but the z- value varies linearly from the start point to the end point as one moves around the arc. Hence the 12d Model arc string is actually a helix with its centre-line perpendicular to the xy-plane.

For convenient 12d Model includes a variety of methods for creating arcs.

On selecting the **Arcs** option, the **Create arcs** menu is displayed giving all the different methods for creating arcs.



Each option in the **Create arcs** menu fires up its own special panel to collect the necessary information for defining the arc.

The mouse is used to select the special points required for the arc and circle definitions. For example, in the **arc - 3 points on arc** option, the three points are selected using the mouse.

Unlike the other strings created under strings=>create, the arc can be created immediately without needing any options from the arc editors.

Consequently, after an arc is defined, the create panel is not removed and is left on the screen to create another arc. Any information that needs modifying is changed in the **arc** panel and is used for the new create.

New arcs are created until either the panel is removed by selecting the **finish** or **[X]** button, or the create terminated by selecting **cancel** from the **pick ops** menu in which case the panel will be left on the screen. In the last case, the arc button is then used to begin a new create sequence.

Each of the methods for creating arcs will now be described.

For the option

Arc, go to 14.1.4.1 Create - Arc
Centre,start,sweep 14.1.4.2 Create Arc - Centre Point, Start Point and Sweep
Centre,2 pts on arc 14.1.4.3 Create Arc - Centre Point, Start and End Points
Centre,radius,2 pts 14.1.4.4 Create Arc - Centre, Radius, Start and End Points
3 pts on arc 14.1.4.5 Create Arc - Three Points on Arc
Start, rad, arc l, bearing 14.1.4.6 Create Arc - Start Point, Radius, Arc Length and Start Bearing
Start, rad, arc l, chord bearing 14.1.4.7 Create Arc - Start Point, Radius, Arc Length and Chord Bearing

14.1.4.1 Create - Arc

Position of option on menu: **Strings =>Create =>Arcs =>Create Arcs**

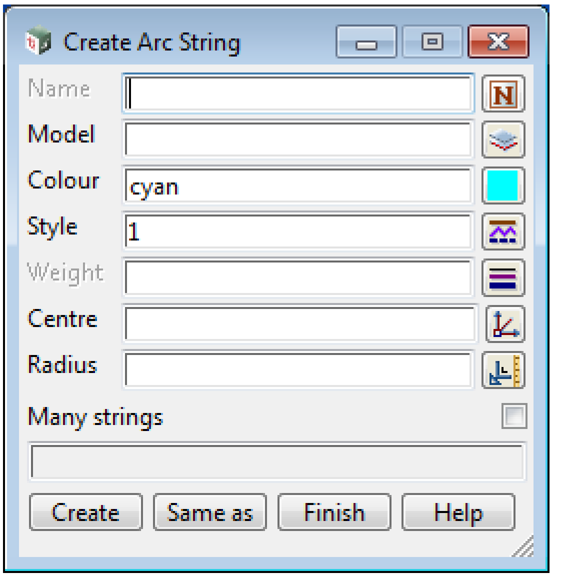
Arcs and circles are normally created using the **strings=>create=> arcs** or **strings=>create=> circles** menu. Please see the documentation on that option for creating arcs and circles.

This option is a special arc creator so that a menu exists if the **strings=>create=>same as** option is used on an arc. It creates an arc with an initial sweep angle of 360 degrees.

The Many strings tick-box is used when more than one string of the same type is to be created, If many strings is set to *tick*, when the current string creation is **Finished** or **Quit**, a new **Create Arc String** panel is placed on the screen with the same information in it as the string just created. If any of the information needs to be modified for the new string, simply change it in the **Create Arc String** panel fields before selecting the **Create** button for the new string. Hence a new string of the same type can be created without going back to the **Create** menu.

The Same as button is used to obtain information from and existing string (not necessarily of the same type) and pipe it into the name, colour, model, style breakline type and height field of the **Create Arc String** panel.

After selecting the option, the **Create arc string** panel is displayed.



The new fields and buttons used in the **create arc string** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input		
the name of the new string.			
Model	input		available models
name of the model that the new string is in.			
Colour	input	default colour	available colours
the colour of the new string.			
Style	input	1	available line styles
line style of the string.			
Weight	input	0	
thickness of the string.			
Centre	input		**xyz ops** menu
co-ordinates of the centre of the arc.			

Radius input

the radius for the arc.

Many strings tick

if ticked then after the current string is finished, a new create panel is placed on the screen with all the same values for the panel fields as the current string.

Create button

After the **create** *button is chosen, the* **arc edit** *menu and* **arc edit info** *panel are displayed.*

Same as button

After the **same as** *button is chosen, another string is selected and information about it is used for the fields in this panel.*

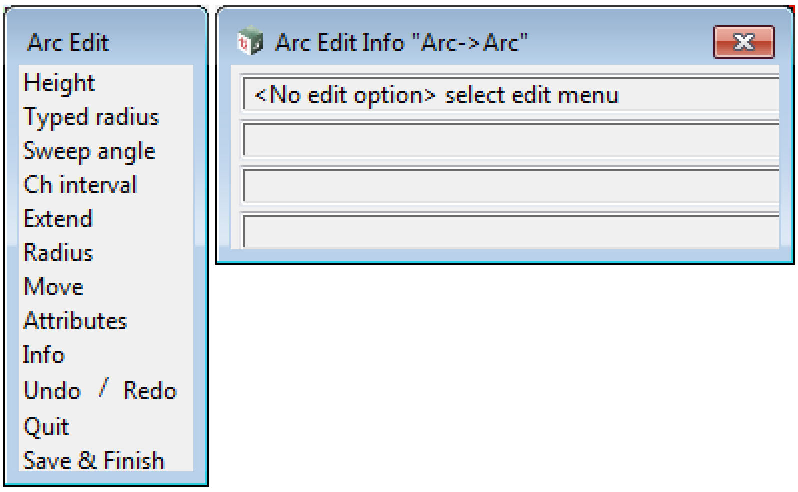
Note - the centre co-ordinates can either be typed into the centre panel field, or if LB is clicked on **[+]** for the centre panel field, the **xyz ops** menu comes up and the **pick xyz** option used to select a point as the arc centre.

Continue to the next section 14.1.4.2 Create Arc - Centre Point, Start Point and Sweep or go back to 14.1.4 Create Arcs.

Start Edit - Arc

Unlike the other string creates, on selecting the **Create** button the arc string is immediately created with a sweep angle of 360 degrees.The **arc edit** menu and **arc edit info** panel are also placed on the screen at the same time

The **arc edit** menu and **arc edit info** panel for an arc string are



For full information in the Arc Editor, go to the section 14.4.5 Arc Edit

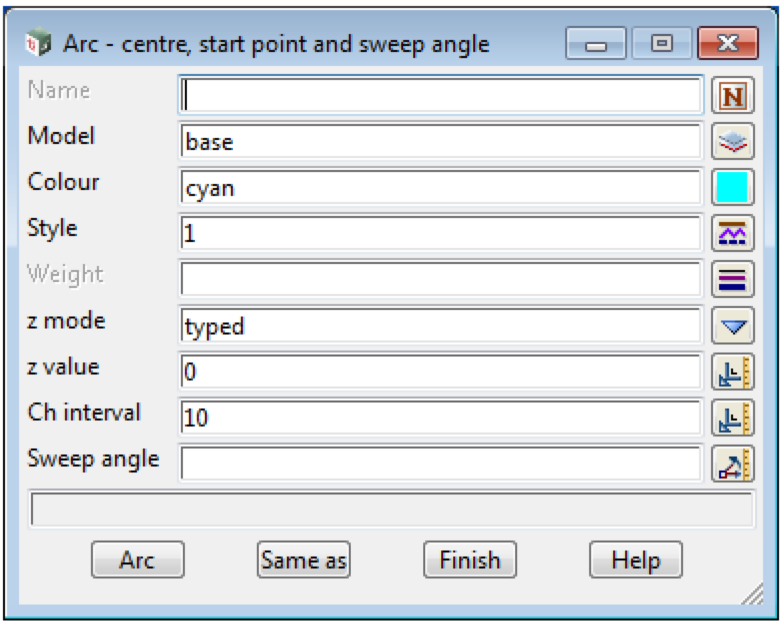
14.1.4.2 Create Arc - Centre Point, Start Point and Sweep

Position of option on menu: **Strings =>Create =>Arcs =>Centre,Start,Sweep**

This option defines an arc by selecting a centre point, the start point of the arc and the sweep angle of the arc. The sweep angle is measured in the clockwise direction.

The radius is calculated by the option and is simply the distance between the centre and start points. The arc end point is also automatically calculated using the known points, radius and sweep angle.

After selecting the option, the **arc - centre, start point and sweep angle** panel is displayed.



To create an arc, the name, colour, model, chainage interval and sweep angle are entered into the appropriate fields and then the centre point and start point of the arc are selected with the cursor. The arc is then created using the information provided in the panel.

The new fields and buttons used in the **arc - centre, start point and sweep angle** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input		

the name of the new string.

Model	input		available models

name of the model that the new string is in.

Colour	input	default colour	available colours

the colour of the new string.

Style	input	1	available line styles

line style of the string.

Weight	input	0	

thickness of the string.

z mode	input	typed	snap, typed

the method of specifying the z-value for the end points of the arc. It can be typed in or taken from the z-value of the selected end points.

z value	input	0	

if the z mode *method is typed, then the height (z-value) of the end points is the value in the field.*

Ch interval	input	10	

the chainage interval used when a chord approximation is needed for the arc.

Sweep angle	input		

sweep angle for the arc - Note: clockwise is positive for sweep angles

Arc	button		

*The **arc** button is used to begin a new create if the previous one was cancelled.*

Same as	button		

After the **same as** *button is chosen, another string is selected and information about it is used for the fields in this panel.*

Panel Messages

Next step messages sent to the screen message area when selecting and accepting the centre point

<Arc centre> [picks][][menu]
<Arc centre> [picks][accepts][menu]

When selecting the start point of the arc

<Arc start> [picks][][menu]
<Arc start> [picks][accepts][menu]

Continue to the next section 14.1.4.3 Create Arc - Centre Point, Start and End Points or go back to 14.1.4 Create Arcs.

14.1.4.3 Create Arc - Centre Point, Start and End Points

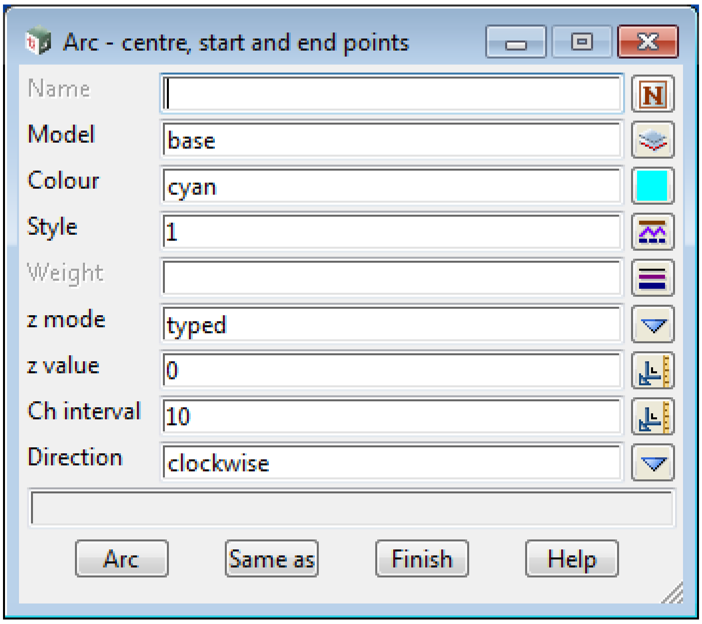
Position of option on menu: **Strings =>Create =>Arcs =>Centre, 2 pts on arc**

This option defines an arc by selecting a centre point, a start and end points for the arc and the direction (clockwise, anti-clockwise) that the arc travels in. The radius is calculated by the option and is simply the distance between the centre and start points.

It is difficult to choose the end point correctly so that it lies exactly on the arc since it only occurs when the distance between the centre and selected end point is exactly the radius of the arc.

Hence, the selected end point and arc direction is used to determine the sweep angle from the start point to the “end point to centre point” line. This is enough information to define the arc. The real end point is then calculated so that it lies on the arc.

After selecting the option, the **arc - centre, start and end points** panel is displayed.



To **create** an arc, the name, colour, model, chainage interval and direction are entered into the appropriate fields and then the centre point, start and end points of the arc selected with the cursor. The arc is then created using the information provided in the panel.

The fields and buttons used in the **arc - centre, start and end points** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input		
the name of the new string.			
Model	input		available models
name of the model that the new string is in.			
Colour	input	default colour	available colours
the colour of the new string.			
Style	input	1	available line styles
line style of the string.			
Weight	input	0	
thickness of the string.			
z mode	input	typed	snap, typed
the method of specifying the z-value for the end points of the arc. It can be typed in or taken from the z-value of the selected end points.			
z value	input	0	
if the z mode *method is typed, then the height (z-value) of the end points is the value in the field.*			
Ch interval	input	10	
the chainage interval used when a chord approximation is needed for the arc.			

Direction	input	clockwise	clockwise, anti-clockwise

the direction for the arc

Arc button

*The **arc** button is used to begin a new create if the previous one was cancelled.*

Same as button

*After the **same as** button is chosen, another string is selected and information about it is used for the fields in this panel.*

Panel Messages

Next step messages sent to the screen message area when selecting and accepting the centre point

<Arc centre> [picks][][menu]
<Arc centre> [picks][accepts][menu]

When selecting the start point of the arc

<Arc start> [picks][][menu]
<Arc start> [picks][accepts][menu]

When selecting the end point of the arc

<Arc end> [picks][][menu]
<Arc end> [picks][accepts][menu]

Continue to the next section 14.1.4.4 Create Arc - Centre, Radius, Start and End Points or go back to 14.1.4 Create Arcs.

14.1.4.4 Create Arc - Centre, Radius, Start and End Points

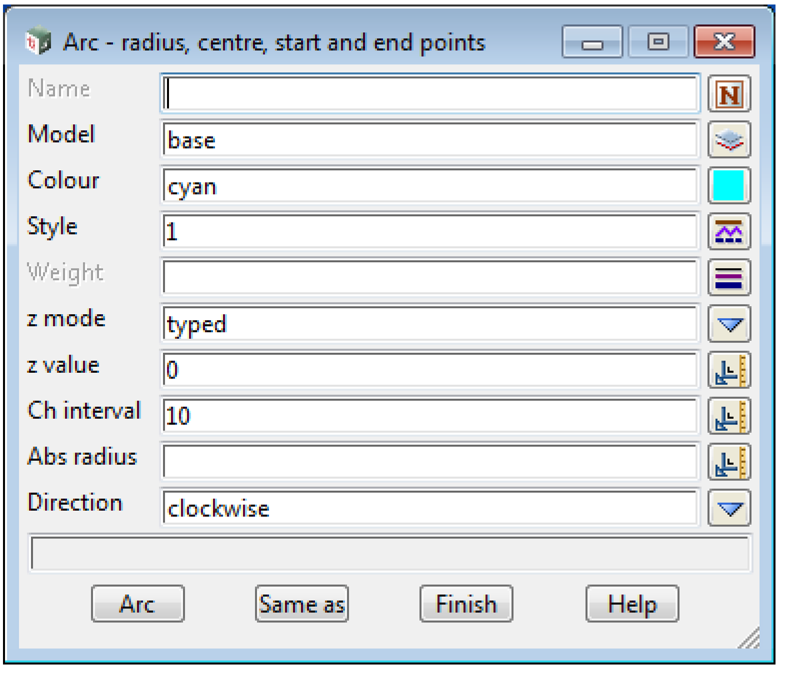
Position of option on menu: **Strings =>Create =>Arcs =>Centre, radius, 2pts**

This option defines an arc by the user setting a radius and direction of arc and then selecting the centre, start and end points for the arc.

Since the radius is given exactly, it is difficult to choose the start and end points so that they lie exactly on the arc. This only occurs when the distance between the centre and selected start and end points is exactly the radius of the arc.

Hence, the selected start point is dropped perpendicularly onto the arc to give the real start point. Similarly, for the real end point.

After selecting the option, the **arc - radius, centre, start and end points** panel is displayed.



To **create** an arc, the name, colour, model, chainage interval, radius and direction are entered into the appropriate fields and then the centre point, start and end points of the arc selected with the cursor. The arc is then created using the information provided in the panel.

The fields and buttons used in the **arc - radius, centre, start and end points** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input		
the name of the new string.			
Model	input		available models
name of the model that the new string is in.			
Colour	input	default colour	available colours
the colour of the new string.			
Style	input	1	available line styles
line style of the string.			
Weight	input	0	
thickness of the string.			
z mode	input	typed	snap, typed
the method of specifying the z-value for the end points of the arc. It can be typed in or taken from the z-value of the selected end points.			
z value	input	0	
if the z mode *method is typed, then the height (z-value) of the end points is the value in the field.*			
Ch interval	input	10	
the chainage interval used when a chord approximation is needed for the arc.			
Abs radius	input		
the absolute radius of the arc.			

Direction input clockwise clockwise, anti-clockwise

the direction for the arc

Arc button

*The **arc** button is used to begin a new create if the previous one was cancelled.*

Same as button

*After the **same as** button is chosen, another string is selected and information about it is used for the fields in this panel.*

Panel Messages

Next step messages sent to the screen message area when selecting and accepting the centre point

<Arc centre> [picks][][menu]
<Arc centre> [picks][accepts][menu]

When selecting the start point of the arc

<Arc start> [picks][][menu]
<Arc start> [picks][accepts][menu]

When selecting the end point of the arc

<Arc end> [picks][][menu]
<Arc end> [picks][accepts][menu]

Continue to the next section 14.1.4.5 Create Arc - Three Points on Arc or go back to 14.1.4 Create Arcs.

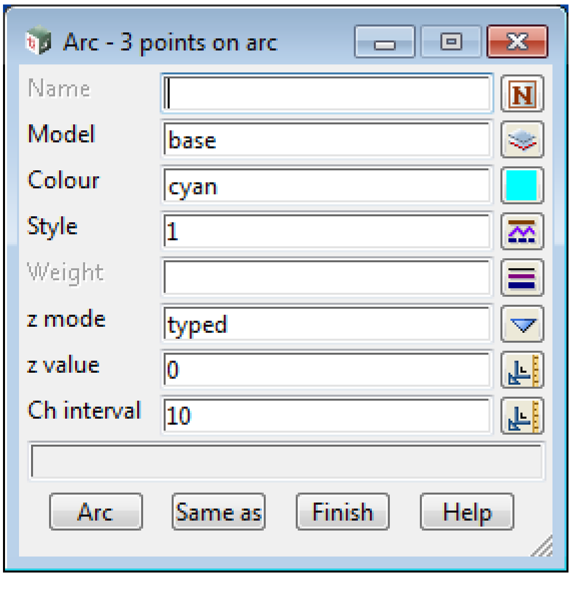
14.1.4.5 Create Arc - Three Points on Arc

Position of option on menu: **Strings =>Create =>Arcs =>3 pts on arc**

This option defines an arc by selecting three points that lie on the arc.

The selected arc points are the start point, a point between the start and end point, and the end point. The three points must be selected in that order.

After selecting the option, the **arc - 3 points on arc** panel is displayed.



To **create** an arc, the name, colour, model and chainage interval are entered into the appropriate fields and then three points - start point, point between the start and end point, and the end point- which lie on the arc are selected with the cursor.

After the three points are picked, the arc is created using the information provided in the panel.

After a arc is defined, the option does not terminate. Another arc can be created by simply modifying any of the information that needs changing in the **arc - 3 points on arc** panel and then selecting three points on a new arc.

The fields and buttons used in the **arc - 3 points on arc** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input		

the name of the new arc.

Model	input		available models

name of the model that the new arc is in.

Colour	input	default colour	available colours

the colour of the new arc.

Style	input	1	available line styles

line style of the arc.

Weight	input	0	

thickness of the string.

z mode	input	typed	snap, typed

the method of specifying the z-value for the arc. It can be typed in or taken from the z-value of the selected end points.

z value	input	0	

if the z mode *method is typed, then the height (z-value) of the end points is the value in the field.*

Ch interval	input	10	

the chainage interval used when a chord approximation is needed for the arc.

Arc	button		

*The **arc** button is used to begin a new create if the previous one was cancelled.*

Same as	button		

After the **same as** *button is chosen, another string is selected and information about it is used for the fields in this panel.*

Panel Messages

Next step messages sent to the screen message area when selecting and accepting the start point

<Arc start> [picks][][menu]
<Arc start> [picks][accepts][menu]

When selecting and accepting the second point

<Point on arc> [picks][][menu]
<Point on arc> [picks][accepts][menu]

When selecting and accepting the end point

<End of arc> [picks][][menu]
<End of arc> [picks][accepts][menu]

Continue to the next section 14.1.4.6 Create Arc - Start Point, Radius, Arc Length and Start Bearing or go back to 14.1.4 Create Arcs.

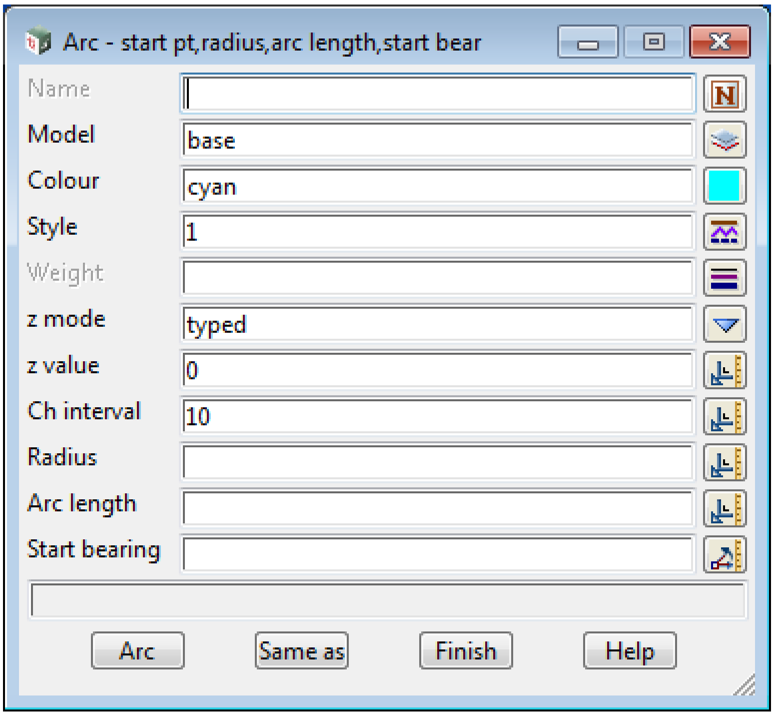
14.1.4.6 Create Arc - Start Point, Radius, Arc Length and Start Bearing

Position of option on menu: **Strings =>Create =>Arcs =>Start, rad, arc, bear**

This option defines an arc by giving the radius, the arc length and start bearing and then selecting a start point for the arc.

This is enough information for the option to calculate the centre and end points and the direction of the arc, and hence, fully define the arc.

After selecting the option, the **arc - start pt, radius, arc length, start bear** panel is displayed.



To **create** an arc, the name, colour, model, chainage interval, radius, arc length and start bearing are entered into the appropriate fields and then the start point of the arc selected with the cursor. The arc is then created using the information provided in the panel.

The new fields and buttons used in the **arc - start pt, radius, arc length, start bear** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input		
the name of the new string.			
Model	input		available models
name of the model that the new string is in.			
Colour	input	default colour	available colours
the colour of the new string.			
Style	input	1	available line styles
line style of the string.			
Weight	input	0	
thickness of the string.			
z mode	input	typed	snap, typed
the method of specifying the z-value for the end points of the arc. It can be typed in or taken from the z-value of the selected end points.			
z value	input	0	
if the z mode *method is typed, then the height (z-value) of the end points is the value in the field.*			

Ch interval input 10

the chainage interval used when a chord approximation is needed for the arc.

Radius input

radius of the arc.

Arc length input

length of the arc from the start to the end point.

Start bearing input

bearing of the tangent to the arc at the start point.

Arc button

*The **arc** button is used to begin a new create if the previous one was cancelled.*

Same as button

*After the **same as** button is chosen, another string is selected and information about it is used for the fields in this panel.*

Panel Messages

Next step messages sent to the screen message area when selecting the start point of the arc

<Arc start> [picks][][menu]
<Arc start> [picks][accepts][menu]

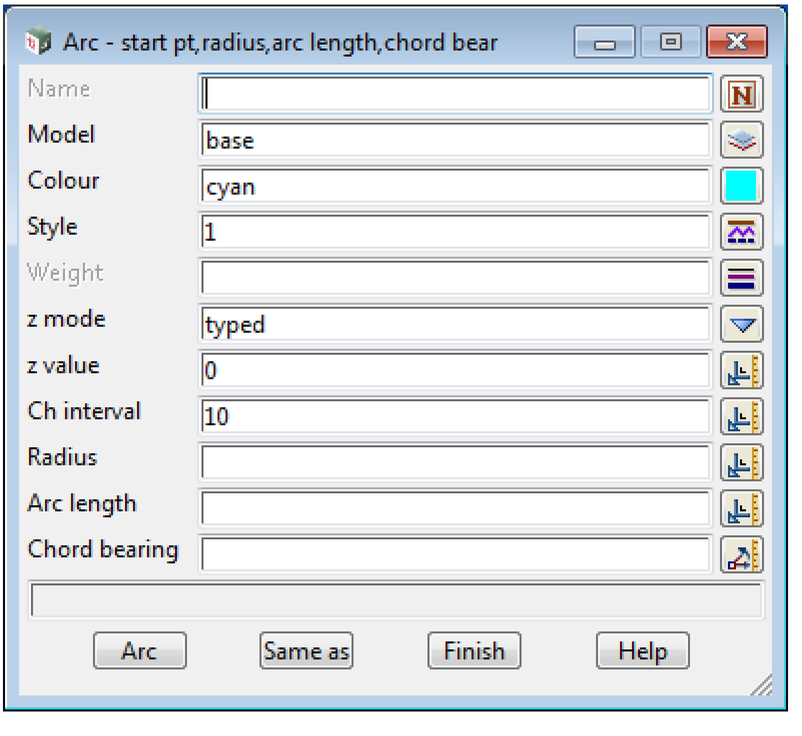
Continue to the next section 14.1.4.7 Create Arc - Start Point, Radius, Arc Length and Chord Bearing or go back to 14.1.4 Create Arcs.

14.1.4.7 Create Arc - Start Point, Radius, Arc Length and Chord Bearing

Position of option on menu: **Strings =>Create =>Arcs =>Start, rad, arc, ch bear**

This option defines an arc by giving the radius, the arc length and chord bearing and then selecting a start point for the arc. This is enough information for the option to calculate the centre and end points and the direction of the arc, and hence, fully define the arc.

After selecting the option, the **arc - start pt, radius, arc length, chord bear** panel is displayed.



To create an arc, the name, colour, model, chainage interval, radius, arc length and chord bearing are entered into the appropriate fields and then the start point of the arc selected with the cursor. The arc is then created using the information provided in the panel.

The new fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input		
the name of the new string.			
Model	input		available models
name of the model that the new string is in.			
Colour	input	default colour	available colours
the colour of the new string.			
Style	input	1	available line styles
line style of the string.			
Weight	input	0	
thickness of the string.			
z mode	input	typed	snap, typed
the method of specifying the z-value for the end points of the arc. It can be typed in or taken from the z-value of the selected end points.			
z value	input	0	
if the z mode *method is typed, then the height (z-value) of the end points is the value in the field.*			
Ch interval	input	10	
the chainage interval used when a chord approximation is needed for the arc.			
Radius	input		
radius of the arc.			
Arc length	input		
length of the arc from the start to the end point.			
Chord bearing	input		
bearing of the chord to the arc at the start point.			
Arc	button		
The ***arc*** *button is used to begin a new create if the previous one was cancelled.*			
Same as	button		
After the **same as** *button is chosen, another string is selected and information about it is used for the fields in this panel.*			

Panel Messages

Next step messages sent to the screen message area when selecting the start point of the arc

<Arc start> [picks][][menu]
<Arc start> [picks][accepts][menu]

Go back to 14.1.4 Create Arcs.

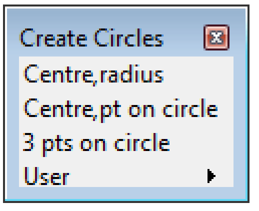
14.1.5 Create - Circles

Position of menu: **Strings =>Create =>Circles**

The create circles option is used to create 12d Model circles.

A 12d Model circle has no start and end points but is defined by a centre point and a radius. The circle has a constant height (z-value).

For convenient 12d Model includes a variety of methods for creating circles. On walking-right on the **circles** option, the **create circles** menu is displayed giving all the different methods for creating circles.



For Centre, radius, go to 14.1.5.1 Create Circle - Centre Point, Radius
Centre, pt on circle 14.1.5.2 Create Circle - Centre Point, Point on Circle
3 pts on circle 14.1.5.3 Create Circle - Three Points on Circle

Each option in the **create circles** menu fires up its own special panel to collect the necessary information for defining the circle.

The mouse is used to select the special points required for the circle definitions. For example, the **circle - centre, radius** option required the value of the radius to be entered into the **circle - centre point, radius** panel and the centre point is selected using the cursor.

Unlike the other strings created under **strings=>create**, the circles can be created immediately without needing any options from the circle editor. Consequently, after a circle is defined, the create panel is not removed and is left on the screen to create another circle. Any information that needs modifying is changed in the **circle** panel and is used for the new create.

New circles are created until either the panel is removed by selecting the **finish** or **[X]** button, or the create terminated by selecting **cancel** from the **pick ops** menu in which case the panel will be left on the screen. In the last case, the circle button is then used to begin a new create sequence.

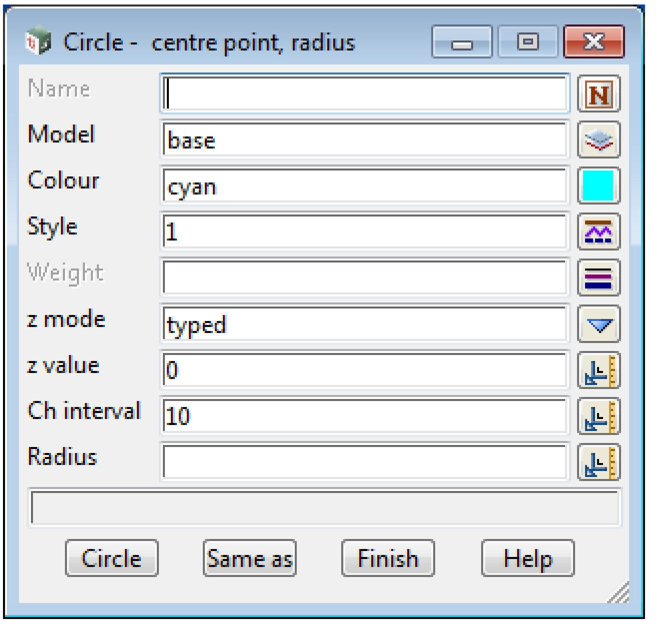
Each of the methods for creating circles and arcs will now be described.

14.1.5.1 Create Circle - Centre Point, Radius

Position of option on menu: **Strings =>Create =>Circles =>Centre, radius**

This option defines a circle by supplying a typed radius and selecting a centre point with the mouse

After selecting the option, the **circle - centre point, radius** panel is displayed.



To **create** a circle, the name, colour, model, chainage interval and radius are entered into the appropriate fields and the centre point selected with the cursor. The circle is then created using the information provided in the panel.

The new fields and buttons used in the **circle - centre point, radius** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input		

the name of the new string.

Model	input		available models

name of the model that the new string is in.

Colour	input	default colour	available colours

the colour of the new string.

Style	input	1	available line styles

line style of the string.

Weight	input	0	

thickness of the string.

z mode	input	typed	snap, typed

the method of specifying the z-value for the circle. It can be typed in or taken from the z-value of the selected centre point.

z value	input	0	

if the z mode *method is typed, then the height (z-value) of the circle is the value in the field.*

Ch interval	input	10	

the chainage interval used when a chord approximation is needed for the circle.

Radius	input	10	

the radius for the circle.

Circle	button		

*The **circle** button is used to begin a new create if the previous one was cancelled.*

Same as button

*After the **same as** button is chosen, another string is selected and information about it is used for the fields in this panel.*

Panel Messages

Next step messages sent to the screen message when selecting and accepting the centre point are

<Circle centre> [picks][][menu]

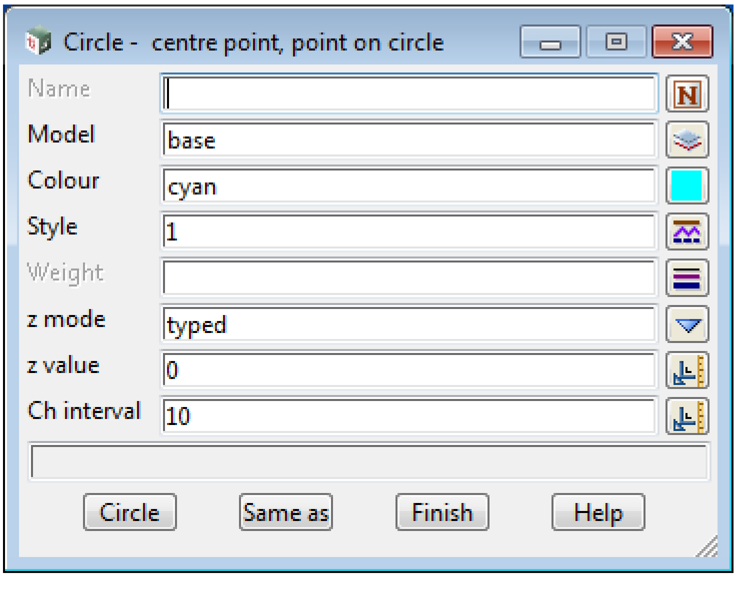
<Circle centre> [picks][accepts][menu]

14.1.5.2 Create Circle - Centre Point, Point on Circle

Position of option on menu: **Strings =>Create =>Circles =>Centre, pt on circle**

This option defines a circle by selecting a centre point and a point on the circle. The radius is automatically calculated by the option and is simply the distance between the two selected points.

After selecting the option, the **circle - centre point, point on circle** panel is displayed.



To create a circle, the name, colour, model and chainage interval are entered into the appropriate fields and the centre point and a point on the circle selected with the cursor. Then the circle is created using the information provided in the panel.

The new fields and buttons used in the **circle - centre point, point on circle** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input		
the name of the new string.			
Model	input		available models
name of the model that the new string is in.			
Colour	input	default colour	available colours
the colour of the new string.			
Style	input	1	available line styles
line style of the string.			
Weight	input	0	
thickness of the string.			

z mode input typed snap, typed

the method of specifying the z-value for the circle. It can be typed in or taken from the z-value of the selected centre point.

z value input 0

if the z mode *method is typed, then the height (z-value) of the circle is the value in the field.*

Ch interval input 10

the chainage interval used when a chord approximation is needed for the circle.

Circle button

*The **circle** button is used to begin a new create if the previous one was cancelled.*

Same as button

After the **same as** *button is chosen, another string is selected and information about it is used for the fields in this panel.*

How to Use the Panel to Create a Circle

(a) The panel fields are filled in with the appropriate data for use in defining the circle.

(b) The mouse is then used to select the centre point for the circle.

Messages sent to the screen message area when selecting and accepting the centre point

<Circle centre> [picks][][menu]
<Circle centre> [picks][accepts][menu

When selecting a point on the circle

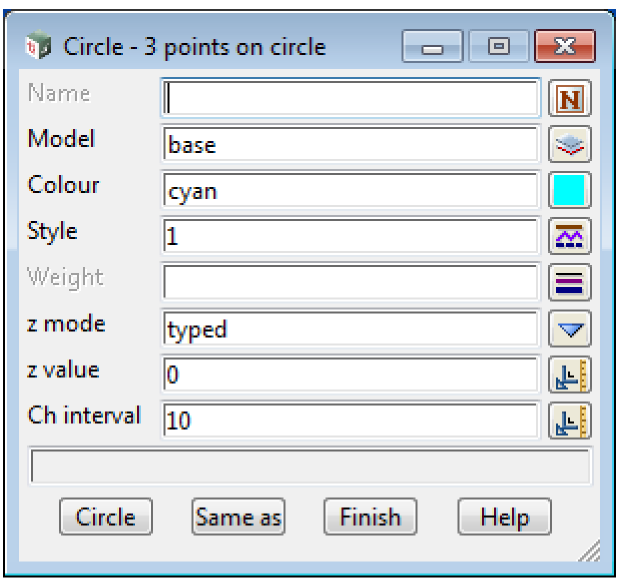
<Point on circle> [picks][][menu]
<Point on circle> [picks][accepts][menu

14.1.5.3 Create Circle - Three Points on Circle

Position of option on menu: **Strings =>Create =>Circles =>3 pts on circle**

This option defines a circle by selecting three points that all lie on the circle.The radius and centre point are automatically calculated by the option.

After selecting the option, the **circle - 3 points on circle** panel is displayed.



To create a circle, the name, colour, model and chainage interval are entered into the appropriate fields and then three points which lie on the circle are selected with the cursor. The circle is then created using the information provided in the panel.

The fields and buttons used in the **circle - 3 points on circle** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input		

the name of the new string.

Model	input		available models

name of the model that the new string is in.

Colour	input	default colour	available colours

the colour of the new string.

Style	input	1	available line styles

line style of the string.

Weight	input	0	

thickness of the string.

z mode	input	typed	snap, typed

the method of specifying the z-value for the circle. It can be typed in or taken from the z-value of the selected centre point.

z value	input	0	

if the z mode *method is typed, then the height (z-value) of the circle is the value in the field.*
if the z mode *method is typed, then the height (z-value) of the circle is the value in the field.*

Ch interval	input	10	

the chainage interval used when a chord approximation is needed for the circle.

Circle	button		

*The **circle** button is used to begin a new create if the previous one was cancelled.*

Same as	button		

After the **same as** *button is chosen, another string is selected and information about it is used for the fields in this panel.*

Panel Messages

Next step messages sent to the screen message area when selecting and accepting the 1st point

<First point on circle> [picks][][menu]
<First point on circle> [picks][accepts][menu]

When selecting and accepting the second point

<Second point on circle> [picks][][menu]
<Second point on circle> [picks][accepts][menu

When selecting and accepting the third point

<Third point on circle> [picks][][menu]
<Third point on circle> [picks][accepts][menu]

14.1.6 Create - Feature

Position of option on menu: **Strings =>Create =>Feature**

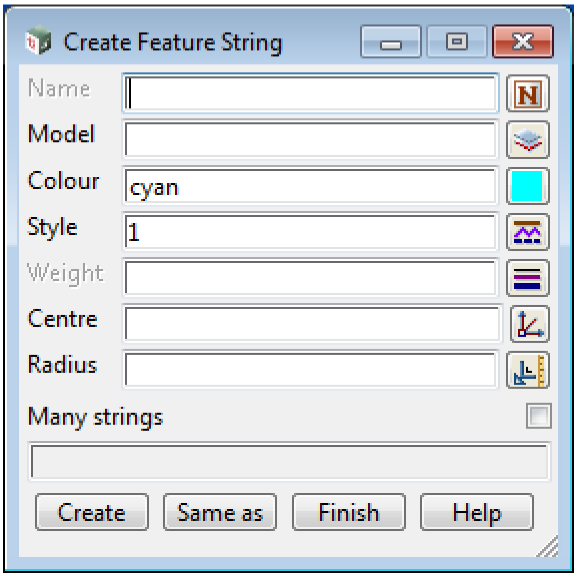
A feature string consists of a centre point (x,y,z) and a user given radius.

The difference between a circle and a feature string is that for a feature string the centre point is considered to be a valid (x,y,z) point but all the points on the circumference of the feature string are null values. Feature strings can be used to represent objects such as trees.

The Many strings tick-box is used when more than one string of the same type is to be created, If many strings is set to *tick*, when the current string creation is **Finished** or **Quit**, a new **Create Feature String** panel is placed on the screen with the same information in it as the string just created. If any of the information needs to be modified for the new string, simply change it in the **Create Feature String** panel fields before selecting the **Create** button for the new string. Hence a new string of the same type can be created without going back to the **Create** menu.

The Same as button is used to obtain information from and existing string (not necessarily of the same type) and pipe it into the name, colour, model, style breakline type and height field of the **Create Feature String** panel.

After selecting the option, the **Create feature string** panel is displayed.



The new fields and buttons used in the **create feature string** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input		
the name of the new string.			
Model	input		available models
name of the model that the new string is in.			
Colour	input	default colour	available colours
the colour of the new string.			
Style	input	1	available line styles
line style of the string.			
Weight	input	0	
thickness of the string.			
Centre	input		**xyz ops** menu
co-ordinates of the centre of the feature.			

Radius input

the radius for the feature.

Many strings tick

if ticked then after the current string is finished, a new create panel is placed on the screen with all the same values for the panel fields as the current string.

Create button

After the **create** *button is chosen, the* **Feature Edit** *menu and* **Feature Edit Info** *panel are displayed.*

Same as button

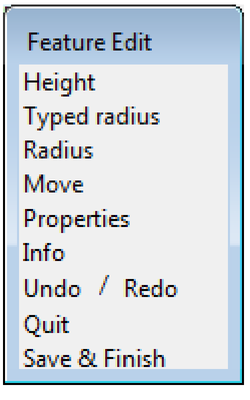
After the **same as** *button is chosen, another string is selected and information about it is used for the fields in this panel.*

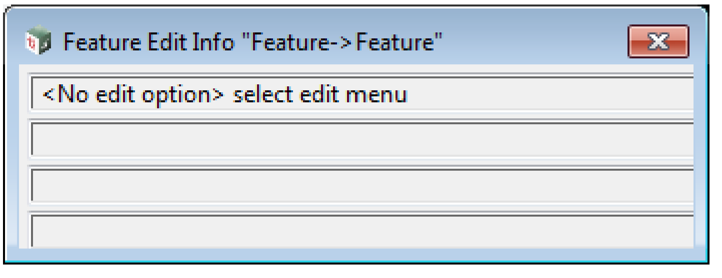
Note - the centre co-ordinates can either be typed into the centre panel field, or if LB is clicked on **[+]** for the centre panel field, the **xyz ops** menu comes up and the **pick xyz** option can be used from it to select a point as the arc centre.

Start Edit - Feature

Like the arc creates, on selecting the **Create** button the feature string is immediately created. The **Feature Edit** menu and **feature edit info** panel are also placed on the screen at the same time

The **Feature Edit** menu and panel for a feature string are





For full information in the Feature Editor, go to the section 14.4.7 Feature Edit

14.1.7 Create - Pipe Super

Position of option on menu: **Strings =>Create =>Pipe**

A pipe super string is the same as a 3d string except that the pipe string also has a diameter hence the options for creating and editing a pipe string are almost the same as for a 3d string.

The main difference is that the pipe diameter can be modified in the Properties panel for the string.

The Many strings tick-box is used when more than one string of the same type is to be created, If many strings is set to *tick*, when the current string creation is **Finished** or **Quit**, a new **Create Super Pipe String** panel is placed on the screen with the same information in it as the string just created. If any of the information needs to be modified for the new string, simply change it in the **Create Super Pipe String** panel fields before selecting the **Create** button for the new string. Hence a new string of the same type can be created without going back to the **Create** menu.

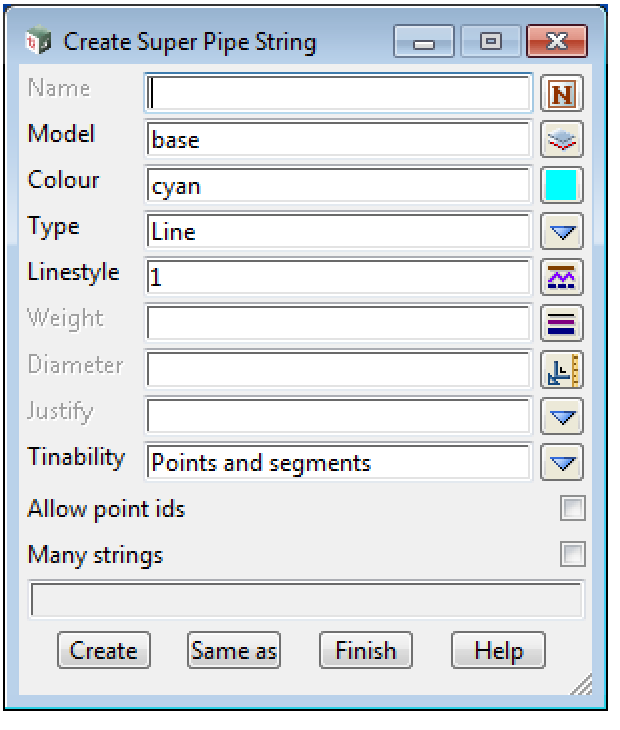
The Same as button is used to obtain information from and existing string (not necessarily of the same type) and pipe it into the name, colour, model, type, linestyle style, weight, diameter and tinability fields of the **Create Super Pipe String** panel.

From *12d Model 8* onwards, the default is to create *super pipe strings* rather than the *pipe strings* used up to *12d Model 7*. The advantage is using a *pipe super string* is that all the CAD options will work for it, vertices and segments can be assigned tinability and it can have point id's for setout.

The older *pipe string* create option is still available under

Strings =>Create =>Old   (go to the section 14.20.1.5 Create - Pipe (Pre V8))

On selecting the **Pipe** option, the **Create Super Pipe String** panel is displayed.



The default values for the panel fields are taken from the *CAD Controlbar (see* 15.1 Controlbars*)*

To create a *new pipe super string*, the panel fields are filled in and the **Create** button selected.

The fields and buttons used in the **Create Super Pipe String** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input	from CAD controlbar	

the name of the new string

Model	model box	from CAD controlbar	available models

name of the model that the new string is in

Colour	colour box	from CAD controlbar	available colours

the colour of the new string

Type	choice box	line	line, point

breakline type (point-line type) of the string

Linestyle	input	from CAD controlbar	available line styles

line style of the string

Weight	input	from CAD controlbar	

thickness of the string

Diameter	input	0	

diameter of the pipe

Justify	input	invert	invert, centre, overt

justification of the pipe with respect to the co-ordinates given for the pipe string.

Tinability	choice box	Points and segments	Points and segments Points only Not tinable

*if **Vertices and segments**, all the vertices and segments of the string are set to **tinable**.*
*If **Vertices only**, all the vertices are set to tinable and the segments to not tinable.*
*If **Not tinable**, all the vertices or segments are set to **not tinable**.*

Allow point ids tick box

*if **ticked**, the super string can have point ids for each vertex.*
*If **not ticked**, the super string will not have point ids. This can be reversed if point ids are required in the future.*

Many strings tick box

if ticked then after the current string is finished, a new create panel is placed on the screen with all the same values for the panel fields as the current string.

Create button

after the **Create** *button is chosen, the* **Edit Pipe** *menu is displayed*

Same as button

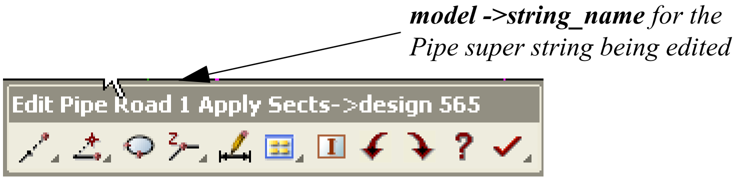
after the **Same as** *button is chosen, another string is selected and information about it is used for the fields in this panel*

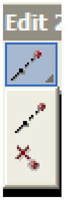
Finish button

*end the option, **don't** proceed to the edit stage*

Start Edit - Pipe Super

On selecting the **Create** button in the **Create Super Pipe String** panel the **Edit Pipe** menu is placed on the screen. The **Edit Pipe** menu for a *pipe super string* is





To create a new *pipe super string*, select the **Append** icon.

The **Append** option is used to add vertices to either end of an existing string, or in the case of a new string, places the first vertex and then begins appending vertices to the first vertex.

For all pipe strings, a cross is then drawn in each plan view that the string's model is on, and the cross follows the cursor around the screen. If the string's model is not added to any plan view, the model is automatically added to all plan views.

After the cross is on the screen (moving with the cursor), clicking LB and accepting with MB selects the first vertex of the string (using the appropriate snaps).

The string is then drawn from the first vertex to the cursor position, which represents the second vertex of the string. Clicking LB and accepting with MB selects the second string vertex and the process repeats for subsequent string vertices.

Now that the string is created, all the edit options on the string’s **Edit Pipe** menu are usable.

The options in the **Edit Pipe** menu are not only used for placing the initial vertices of the string, but for editing the string once it is created. Since the **Append** and other options in the **Edit Pipe** menu are identical to the options used when editing an existing string, they will be discussed in detail in the string **Editor** section.

For full information in the Pipe Editor, go to the section 14.4.8 Edit Pipe

14.1.8 Create - Polyline Super

Position of option on menu: **Strings =>Create =>Polyline**

A polyline string is similar to a 3d string except that it can have either straight lines or arcs joining the (x,y,z) vertices of the string. The arcs are plan arcs with possibly a different z at either end and the z values are linearly interpolated between the end points. Hence in a long section, the end vertices are joined by straight lines for both line and arc segments of the polyline.

Creating and editing a polyline string is very similar to a 3d string. The only major difference is that a radius is required at each string segment (a radius of 0 means no arc, just a straight line).

The Many strings tick-box is used when more than one string of the same type is to be created, If many strings is set to *tick*, when the current string creation is **Finished** or **Quit**, a new **Create Super Polyline String** panel is placed on the screen with the same information in it as the string just created. If any of the information needs to be modified for the new string, simply change it in the **Create Super Polyline String** panel fields before selecting the **Create** button for the new string. Hence a new string of the same type can be created without going back to the **Create** menu.

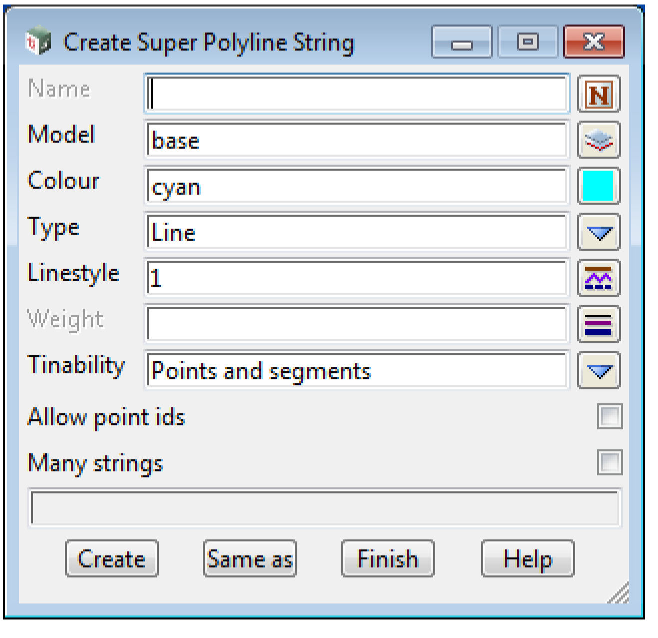
The Same as button is used to obtain information from and existing string (not necessarily of the same type) and pipe it into the name, colour, model, type, linestyle style and weight fields of the **Create Super Polyline String** panel.

From *12d Model 8* onwards, the default is to create *super polyline strings* rather than the *polyline strings* used up to *12d Model 7*. The advantage is using a *super polyline string* is that all the CAD options will work for it, vertices and segments can be assigned tinability and it can have point id's for setout.

The older *pipe string* create option is still available under

Strings =>Create =>Old (go to the section 14.20.1.6 Create - Polyline (Pre V8))

On selecting the **Polyline** option, the **Create Super Polyline String** panel is displayed.



The default values for the panel fields are taken from the *CAD Controlbar (see* 15.1 Controlbars*)*

To create a *new polyline super string*, the panel fields are filled in and the **Create** button selected.

The fields and buttons used in the **Create Super Polyline String** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input	from CAD controlbar	

the name of the new string

Model	model box	from CAD controlbar	available models

name of the model that the new string is in

Colour	colour box	from CAD controlbar	available colours

the colour of the new string

Type	choice box	line	line, point

breakline type (point-line type) of the string

Linestyle	input	from CAD controlbar	available line styles

line style of the string

Weight	input	from CAD controlbar	

thickness of the string

Tinability	choice box	Points and segments	Points and segments Points only Not tinable

*if **Vertices and segments**, all the vertices and segments of the string are set to **tinable**.*
*If **Vertices only**, all the vertices are set to tinable and the segments to not tinable.*
*If **Not tinable**, all the vertices or segments are set to **not tinable**.*

Allow point ids tick box

*if **ticked**, the super string can have point ids for each vertex.*
*If **not ticked**, the super string will not have point ids. This can be reversed if point ids are required in the future.*

Many strings tick box

if ticked then after the current string is finished, a new create panel is placed on the screen with all the same values for the panel fields as the current string.

Create button

after the **Create** *button is chosen, the* **Edit Polyline** *menu is displayed*

Same as button

after the **Same as** *button is chosen, another string is selected and information about it is used for the fields in this panel*

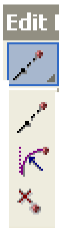
Finish button

*end the option, **don't** proceed to the edit stage*

Start Edit - Polyline Super

On selecting the **Create** button in the **Create Super Polyline String** panel the **Edit Polyline** menu is placed on the screen. The **Edit Polyline** menu for a *polyline super string* is

***model ->string_name** for the*
Polyline super string being edited



append or prepend a vertex

append or prepend a vertex plus radius

delete an IP

To create a new polyline string, the user must select one of the two append options (**Append** or **Append + radius**) on the **Append** drop-down toolbar on the **Edit Polyline** toolbar. The **Append** option simply creates vertices which are joined by a straight segment whereas the **Append + radius** creates vertices with a radius for the segment.

The **Append** options are used to add vertices to either end of an existing string, or in the case of a new string, places the 1st vertex and then begins appending vertices to the 1st vertex.

For all polyline strings, a cross is then drawn in each plan view that the string's model is on, and the cross follows the cursor around the screen. If the string's model is not added to any plan view, the model is automatically added to all plan views.

After the cross is on the screen (moving with the cursor), clicking LB and accepting with MB selects the 1st vertex of the string (using the appropriate snaps).

If **Append + radius** was selected, the user will then be prompted for the radius of the segment being placed.

The string is then drawn from the 1st vertex to the cursor position, which represents the second vertex of the string. Clicking LB and accepting with MB selects the second string vertex and the process repeats for subsequent string vertices.

The options in the **Edit Polyline** menu are not only used for placing the initial vertices of the string, but for editing the string once it is created. Since the **Append** and other options in the **Edit Polyline** menu are identical to the options used when editing an existing string, they will be discussed in detail in the string **Editor** section.

For full information in the Polyline Editor, go to the section 14.4.9 Edit Polyline

14.1.9 Create - Super

Position of option on menu: **Strings =>Create =>Super**

A super string is a combination and generalisation of a 4d string, a pipe string and a polyline string.

A super strings consists of a series of (x,y) vertices, plus it can have either straight lines or arcs joining the vertices of the string.

At each vertex, the super string has:

height
text
visibility
tinability (contourability)
point number
symbol
user defined attributes

For each segment, the super string has:

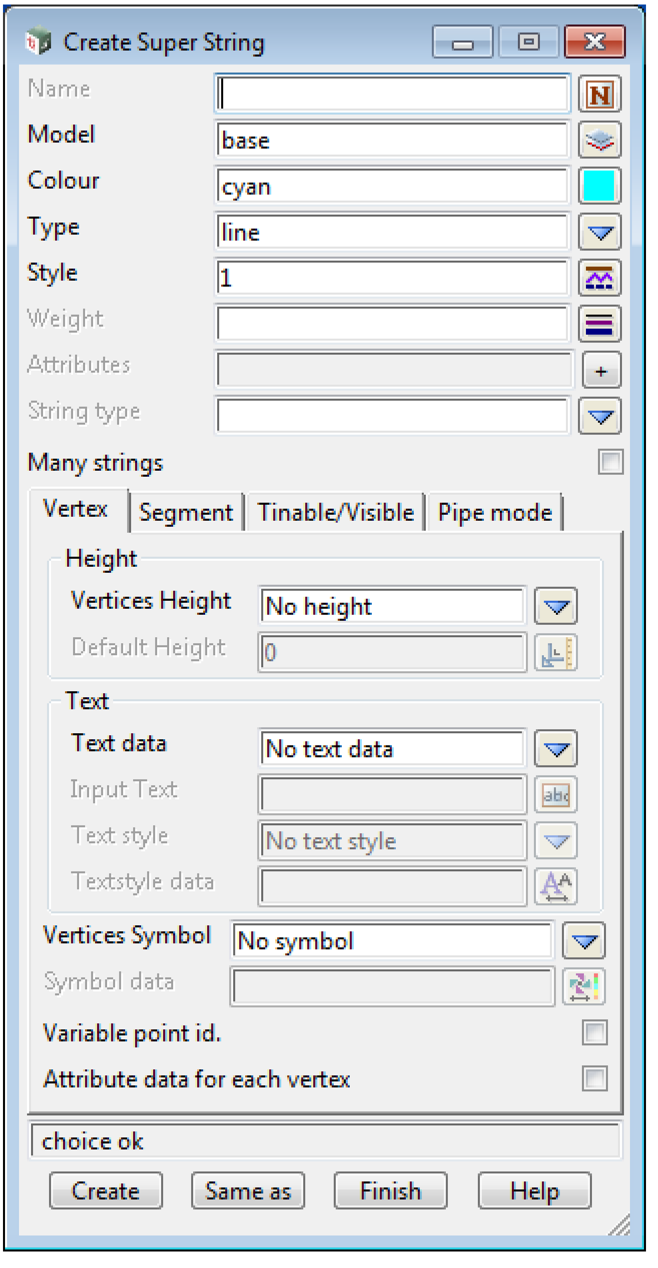
plan radius
colour
text
tinability (breakline)
pipe diameter or box dimensions
visibility
user defined attributes

Creating and editing a super string is very similar to a polyline string in that a radius is can be given at each string segment (a radius of 0 means no arc, just a straight line), and also like a 4d string in that there can be text at each vertex. However there are many more choices about what is defined for each vertex and segment of the super string.

The Many strings tick-box is used when more than one string of the same type is to be created, If many strings is set to *tick*, when the current string creation is **Finished** or **Quit**, a new **Create Super String** panel is placed on the screen with the same information in it as the string just created. If any of the information needs to be modified for the new string, simply change it in the **Create Super String** panel fields before selecting the **Create** button for the new string. Hence a new string of the same type can be created without going back to the **Create** menu.

The Same as button is used to obtain information from and existing string (not necessarily of the same type) and pipe it into the name, colour, model, style breakline type and height field of the **Create Super String** panel.

On selecting the **Super string** option, the **Create Super String** panel is displayed.



When the panel opens, various panel field values are taken from the **ControlBars**.

The information in **Name**, **Model**, **Colour**, **Style** and **Tinable** are taken from the CAD ControlBar, **Attributes** from the Attributes ControlBar and **Pipe** information from the Pipe ControlBar.

If the **Name** icon is used in the **Create Super String** panel to select a name from the names.4d pop up, then the panel fields are updated from the names.4d file.

To **create** a new super string, the appropriate panel fields are filled in and the **Create** button clicked.

The new fields and buttons used in the **Create Super String** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	names.4d box		names.4d file

the name of the new string.

Model	model box		available models

name of the model for the new string.

Colour	colour box	default colour	available colours

the colour of the new string.

Note that each segment can have a different colour if in the **Segment** *tab, the* **Segment colour** *choice is* ***Varied****.*

Type	choice box	line	line, point

breakline type (point-line type) of the string

As an overall string property:
If a string has type **line***, then all vertices and all segments are tinable.*
If a string has type **point***, then all vertices are tinable but the segments are not tinable.*

Note that each vertex and segment can have different tinabilities and it is controlled in the **Tinable** *tab.*

Style	linestyle box	1	available line styles

line style of the new string.

Weight	weight box	0	

thickness of the string when plotted.

String type	choice box		2d like, 3d like, 4d like

if ***not blank****, the like type sets a number of the panel field settings.*

Many strings	tick box		

if ticked then after the current string is finished, a new **Create Super String** *panel is placed on the screen with all the same values for the panel fields as in the current panel.*

Vertex tab

The fields for setting up what information is required for vertices.

Vertices height	choice box		Constant for all Varied No height

if ***Constant for all****, there is only one height that is used for all vertices of the string (2d like).*
If ***Varied****, there is a different height for each vertex (3d like)*
If ***No height****, there is no height at all for the string.*

if the Vertices height *choice is* Constant for all *then the field* Default height *is enabled.*

Default height	real box		

the one height that is used for every vertex of the string.

Text data	choice box		Constant for all Varied No text data

if ***Constant for all****, there is one text for all the vertices of the string.*
If ***Varied****, there is different text for each vertex.*
If ***No text data****, there is no text at all for the string.*

if the Text data *choice is* Constant for all *then the fields* Input text *and T*ext style *are enabled.*

Input text	text box		

the one bit of text that is used for every vertex of the string.

Text style	choice box		Constant for all Varied

No text style

*if **Constant for all**, there is one textstyle that is used for all the vertices of the string.*
*If **Varied**, there is a different textstyle for each vertex.*
*If **No text data**, there is no textstyle at all for the string.*

if the Text style *choice is* Constant for all *then the field* Textstyle data *is enabled.*

Textstyle data textstyle data box

the one textstyle data that is used for every vertex of the string.

Symbol data choice box Constant for all
Varied
No symbol

*if **Constant for all**, there is one symbol that is used for all vertices of the string.*
*If **Varied**, there is a different symbol for each vertex*
*If **No symbol**, there is no symbols at all for the string.*

if the Symbol data *choice is* Constant for all *then the field* Symbol data *is enabled.*

Symbol data symbol data box

the one symbol data that is used for every vertex of the string.

Variable point id tick box

if ticked, there is a different point id for each vertex.
If not ticked, there are no vertex ids for any vertex of the string.

Attribute data for each vertex tick box

if ticked, there can be attributes for each vertex.
If not ticked, there are no attributes on any vertex.

Segment tab

The fields for setting up what information is required for segments.

Segments colour choice box Constant for all
Varied

*if **Constant for all**, there is one colour for every segment of the string and that is the colour in the* **Colour** *field.*

*If **Varied**, there is a different colour for each segment.*

Text data choice box Constant for all
Varied
No text data

*if **Constant for all**, there is one text for all the segments of the string.*
*If **Varied**, there is different text for each segment.*
*If **No text data**, there is no text at all for the string.*

if the Text data *choice is* Constant for all *then the fields* Input text *and* Text style *are enabled.*

Input text text box

the one bit of text that is used for every segment of the string.

Text style choice box Constant for all
Varied
No text style

*if **Constant for all**, there is one textstyle that is used for all the segments of the string.*

*If **Varied**, there is a different textstyle for each segment.*
*If **No text data**, there is no textstyle at all for the string.*

if the Text style *choice is* Constant for all *then the field* Textstyle data *is enabled.*

Textstyle data textstyle data box

the one textstyle data that is used for every segment of the string.

Radius/Major tick box

if ticked, each segment can have a radius and a major flag.
If not ticked, all segments are straight lines.

***Note:** the radius is a signed value and in the direction of travel along the arc, a negative radius means the arc bends to the left and a positive radius means the arc bends to the right. However this is still not enough to uniquely define the arc and it can be a minor major arc (turns through less than or equal to 180 degrees), or a major arc (turns through more than 180 degrees). The Major flag is 0 for a minor arc and 1 for a major arc.*

Attribute data for each segment tick box

if ticked, there can be attributes for each segment.
If not ticked, there are no attributes on any segment.

Tinable/Visible tab

The fields for setting up what tinability is allowed for vertices and segments.

***Tinability** for a **vertex** is the property that the vertex is included in a tin (triangulation).*

If a vertex is tinable then it is included in a tin.

If a vertex is not tinable, then it is not included in a tin.

***Tinability** for a **segment** is the property that the segment is to be included in a tin (triangulation). That is, the segment is to be a side of a triangle in a tin. Another name for a tinable segment is that it is a **breakline**. Even though a segment is tinable, it may not end up in a tin if there are crossing breaklines. Also a tinable segment can not be included in a tin if either of its end vertices are not tinable.*

If a segment is tinable then if possible, it is included as the side of a triangle in a tin.

If a segment is not tinable, then it does not have to be included as a side of a triangle in a tin.

As an overall string property,:
If a string has type **line**, *then all vertices and all segments are tinable.*
If a string has type **point**, *then all vertices are tinable but the segments are not tinable.*

Vertex tinability choice box Constant
Varied
Not tinable

*if **Constant**, there is one tinability and it is the same for all vertices in the string.*
*If **Varied**, each vertex can have a different tinability.*
*If **Not tinable**, then all vertices are not tinable.*

Segment tinability choice box Constant
Varied
Not tinable

*if **Constant**, there is one tinability and it is the same for all segments in the string.*
*If **Varied**, each segment can have a different tinability.*
*If **Not tinable**, then all segments are not tinable.*

Pipe mode tab

The fields for setting up what pipe modes are allowed for segments.

Segments can be all round for a string, or all box (culvert) for a string but can't be a mixture of round and box in the one string.

If a segment is round then it just needs a diameter.

If a segment is culvert (or box), it needs a width and a height.

Pipe choice

There are five choice: the string is a
constant pipe string and has one diameter for the entire string
a variable pipe string and can have a different diameter for each segment
a constant culvert string and has the one width and height for each segment
a variable culvert string and can have a different width and height for each segment
or the string has no diameter or width and height for any segment.

Pipe mode	choice box	no round pipe or culvert round pipe entire string round pipe each segment culvert entire string culvert each segment

if **no round pipe or culvert**, *there is no diameters, or any widths or heights for any segment.*
if **round pipe entire string,** *there is* ***one*** *diameter for all segments in the string.*
if **round pipe each segment**, *there is a different diameter for each segments of the string.*
if **culvert entire string**, *there is* ***one*** *width and a height for all segments in the string.*
if **culvert each segment**, *there is a different width and height for each segment of the string.*

if the Pipe mode *choice is* round pipe entire string *then the field* Diameter *is enabled*.

Diameter real box

diameter to use for every segment of the string.

if the Pipe mode *choice is* culvert entire string *then the fields* Width *and* Height *are enabled*.

Width/Height real box

width/height to use for every segment of the string.

if the Pipe mode *choice is NOT* no round pipe or culvert *then the field* Justify *is enabled.*

Justify choice box Invert, centre, obvert

If **invert**, *the coordinates of the vertices are on the top of the pipe/culvert.*
If **centre**, *the coordinates of the vertices are in the centre of the pipe/culvert.*
If **obvert**, *the coordinates of the vertices are at the bottom of the pipe/culvert.*

Buttons at bottom

Create button

After the **create** *button is chosen, the* **super edit** *menu and* **super edit info** *panel are displayed.*

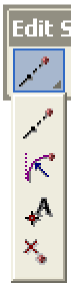
Same as button

After the **same as** *button is chosen, another string is selected and information about it is used for the fields in this panel.*

Start Edit - Super String

On selecting the **Create** button in the **Create Super String** panel the **Super Edit** menu and **Super Edit Info** panels are placed on the screen.

model ->string_name *for the super string being edited*



append or prepend an IP

append or prepend an IP plus radius

append or prepend and IP plus text

delete an IP

To create a new super string, select the **Append** or **Append + radius** or **Append + text** icons.

The **Append** and **Append + text** options simply creates vertices that are joined by a straight segments. **Append + text** also asks for text at each vertex.

The **Append + radius** create vertices with a radius for the segment.

For all super strings, a cross is then drawn in each plan view that the string's model is on, and the cross follows the cursor around the screen. If the string's model is not added to any plan view, the model is automatically added to all plan views.

After the cross is on the screen (moving with the cursor), clicking LB and accepting with MB selects the first vertex of the string (using the appropriate snaps).

If **Append + radius** was selected, the user will then be prompted for the radius of the segment being placed.

If **Append + text** was selected, the user will then be prompted for the text to be placed at the vertex.

The string is then drawn from the first vertex to the cursor position, which represents the second vertex of the string. Clicking LB and accepting with MB selects the second string vertex and the process repeats for subsequent string vertices.

Now that the string is created, all the edit option on the string's **Super Edit** menu are usable.

The options in the **Super Edit** menu are not only used for placing the initial vertices of the string, but for editing the string once it is created. Since the **Append** and other options in the **Super Edit** menu are identical to the options used when editing an existing string, they will be discussed in detail in the string **Editor** section.

For full information in the Super Editor, go to the section 14.4.10 Edit Super

14.1.10 Create - Text

Position of option on menu: **Strings =>Create =>Text**

A text string consists of

(a) the text

(b) an (x,y) position for the text

(c) a text style

(d) a justification

(e) the height of the text in pixels, paper or world units

(f) the x factor and slant of the text

(g) the colour of the text

(h) the angle to write the text at

The default text style uses a non-proportional font but user defined fonts can be proportional or non-proportional.

The justification of the text with respect to text string's (x,y) position is user specified.

The text angle is measured in a counter- clockwise direction with respect to the horizontal axis.

For more information on text definitions, go to the section 4.6 Text Definitions.

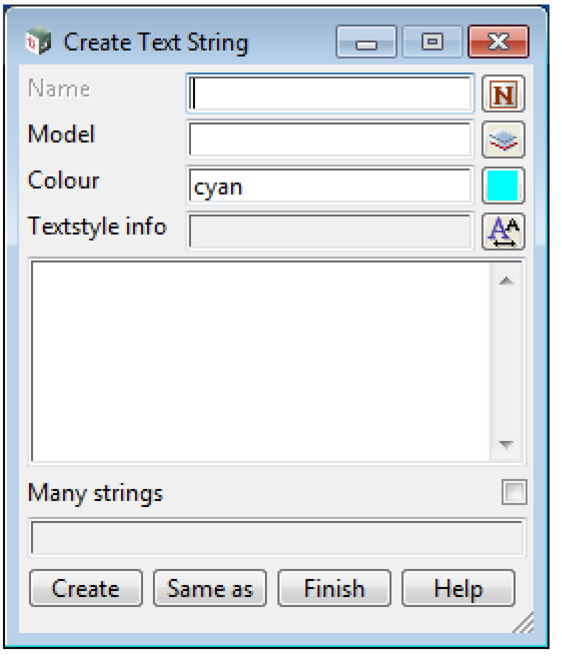
Note - deleting text

Text is deleted as a normal string with the picking point at the text string's definition point. The relationship of the definition point and the text depends on the text offset and justification

The Many strings tick-box is used when more than one string of the same type is to be created, If many strings is set to *tick*, when the current string creation is **Finished** or **Quit**, a new **Create Text String** panel is placed on the screen with the same information in it as the string just created. If any of the information needs to be modified for the new string, simply change it in the **Create Text String** panel fields before selecting the **Create** button for the new string. Hence a new string of the same type can be created without going back to the **Create** menu.

The Same as button is used to obtain information from and existing string (not necessarily of the same type) and pipe it into the name, colour, model, style breakline type and height field of the **Create Text String** panel.

On selecting the **Text string** option, the **Create text string** panel is displayed.



To **create** a new text string, the name, model, colour and actual text of the new string are entered into the appropriate fields, and the textstyle info such as height, offset, justification, angle are entered by clicking on the Textstyle info box. The **create** button is then selected.

The new fields and buttons used in the **create text string** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	name box		

the name of the new string.

Model	model box		available models

name of the model that the new string is in.

Colour	colour box	default colour	available colours

the colour of the new string.

Textstyle info	textstyle info box		

textstyle information.

Text	multi-line text box		

the actual text.

Many strings	tick box	not ticked	

if ticked then after the current string is finished, a new create panel is placed on the screen with all the same values for the panel fields as the current string.

Create	button		

*After the **create** button is chosen, the **text edit** menu and **text edit info** panel are displayed.*

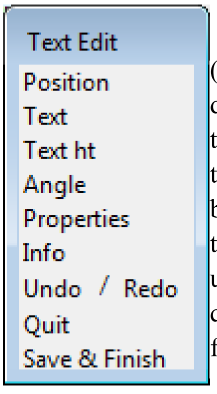
Same as	button		

*After the **same as** button is chosen, another string is selected and information about it is used for the fields in this panel.*

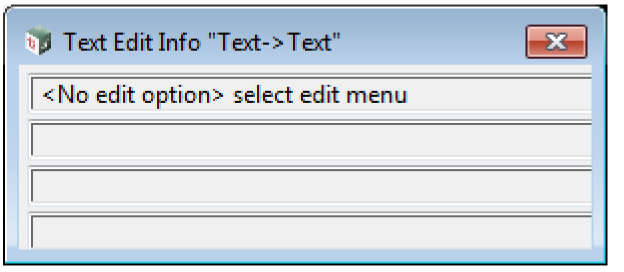
Start Edit - Text

On selecting the **create** button in the **create text string** panel the **text edit** menu and **text edit info** panel are placed on the screen.

The **text edit** menu for a text string is and the **text edit** panel is



(x,y) position of text
define the actual text
text height
text angle
bring up attribute panel
toggle edit info panel
undo/redo
quit the create
finish the text edit



To creating a new text string, the user must first select the **Position** option from the **Text edit** menu. Then after selecting a position, the text is created and displayed (if the model is on the view).

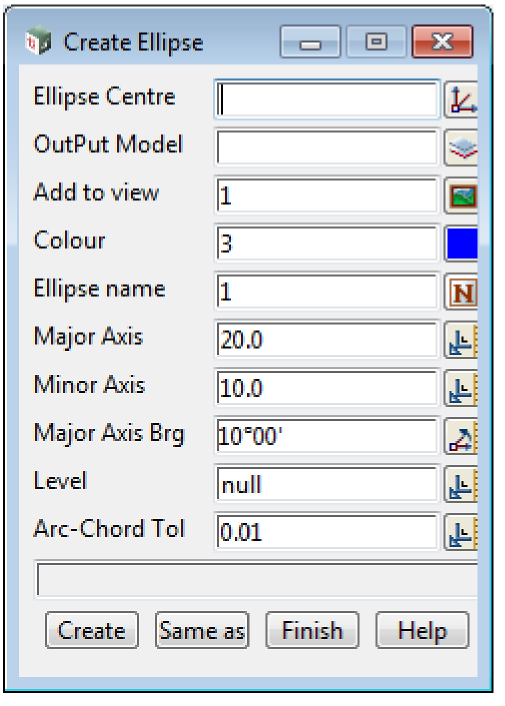
For full information in the Text Editor, go to the section 14.4.12 Text Edit

14.1.11 Create - Ellipse

Position of option on menu: **Strings =>Create =>Ellipse**

.This option creates a super string of points along the ellipse defined in the panel. The frequency of points is determined by a user defined chord-to-arc tolerance.

After selecting the **Ellipse** option, the **Create Ellipse** panel is displayed.



Field Description	Type	Defaults	Pop-Up
Ellipse centre	model select		
centre of the ellipse			
OutPut model	model box		available models
name of the model to place put the ellipse string in			
Add to view	view select	1	
add the model to the given view			
Colour	colour box	default colour	available colours
the colour of the new string.			
Ellipse name	name mapping box		names in name mapping file
the name of the new string.			
Major Axis	measure box	20	
the length of the major axis of the ellipse			
Minor Axis	measure box	10	
the length of the minor axis of the ellipse			
Major Axis Brg	measure box		
bearing of the major axis of the ellipse			
Level	measure box	null	
z-value for the points on the ellipse			

Arc-Chord Tol measure box 0.01

the chord to arc tolerance to use for deciding how often to create points around the ellipse

Create button

after the **Create** *button is chosen, a string of points around the ellipse is created*

Same as button

after the **same as** *button is chosen, another string is selected and information about it is used for the fields in this panel.*

14.1.12 Create - Control Stations

Position of option on menu: **Strings =>Create =>Create Control Station**

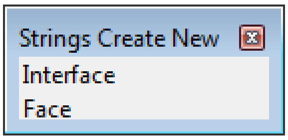
This option is documented in **Survey =>Extras =>Create Control stations** in the section 17.14.5 Create Control Stations.

14.1.13 New

Position of option on menu: **Strings =>Create =>New**

This section of documentation is a work in progress and will be updated in subsequent releases.

The **New** walk-right menu contains options *Interface* and *Face*.



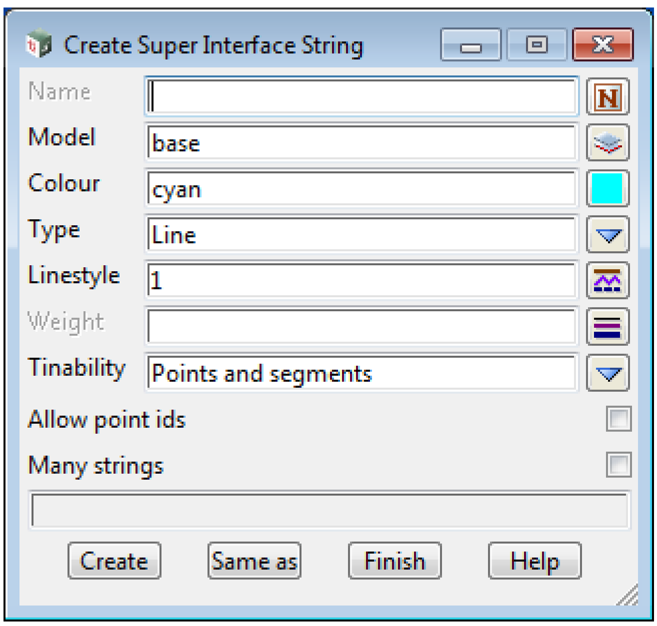
For the option *Interface*, go to 14.1.13.1 Interface
Face 14.1.13.2 Face

14.1.13.1 Interface

Position of option on menu: **Strings =>Create =>New =>Interface**

This section of documentation is a work in progress and will be updated in subsequent releases.

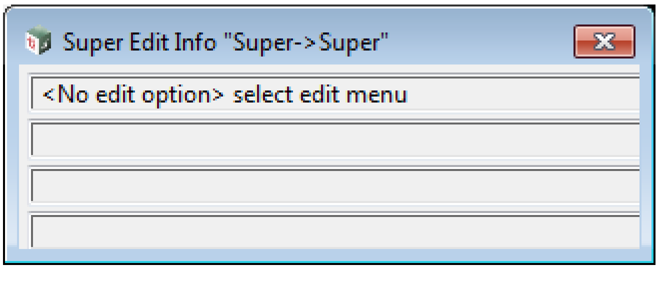
On selecting the **Interface** option, the Create Super Interface String panel is displayed.



The fields and buttons used in the **Create Super Interface String panel** have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input		
Model	model box		available models
Colour	colour box		available colours
Type	choice box	line	point, line
Linestyle	input	from CAD controlbar	available line styles
Weight	input	from CAD controlbar	
Tinability	choice box	Points and segments	Points and segments Points only Not tinable
Allow point ids	tick box		
Many strings	tick box		
Create	button		
Same as	button		

Once the Create button has been pressed, the Edit Interface toolbar and Super edit Info panel appears on the screen, containing the list of available edit options for the Super Interface String.

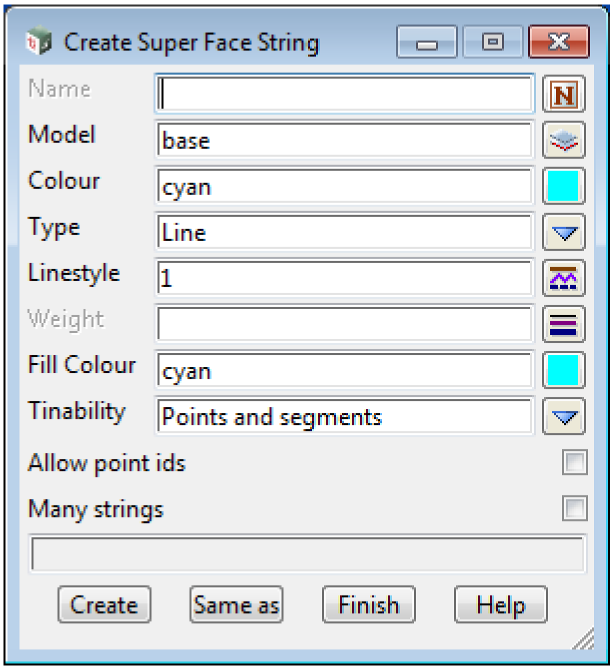


14.1.13.2 Face

Position of option on menu: **Strings =>Create =>New =>Face**

This section of documentation is a work in progress and will be updated in subsequent releases.

On selecting the **Face** option, the Create Super Face String panel is displayed.

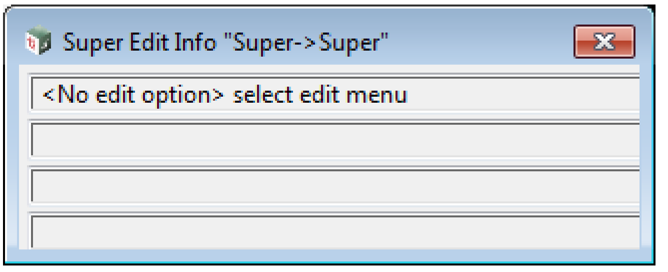


The fields and buttons used in the **Create Super Face String panel** have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input		
Model	model box		available models
Colour	colour box		available colours
Type	choice box	line	point, line
Linestyle	input	from CAD controlbar	available line styles
Weight	input	from CAD controlbar	
Fill Colour	colour box		available colours
Tinability	choice box	Points and segments	Points and segments Points only Not tinable
Allow point ids	tick box		
Many strings	tick box		
Create	button		
Same as	button		

Once the Create button has been pressed, the Edit Face toolbar and Super edit Info panel appears on the screen, containing the list of available edit options for the Super Face String


Edit Face BUILDING PAD->FOOTING EDGE

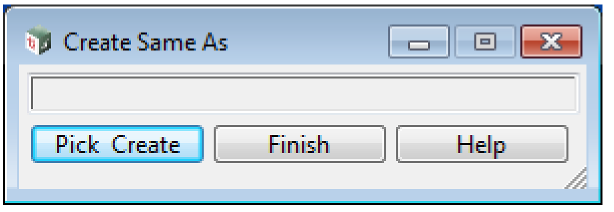

Super Edit Info "Super->Super"
<No edit option> select edit menu

14.1.14 Create - Same As

Position of option on menu: **Strings =>Create =>Same as**

The **same as** option is used to create a new string of the same type and with similar header information as an existing string.

On selecting the **same as** create option, the **create same as** panel is displayed.



After selecting the option, the option is already running and waiting for a string to be selected.

Once a string is selected from a view, a new **create nd string** panel of the same type as the selected string and with all the header information from the selected string, is displayed on the screen.

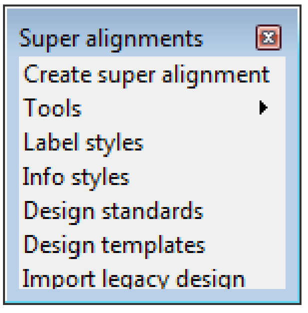
The user then proceeds as a normal create string option.

If the pick is terminated by clicking RB to get up the **pick ops** menu and selecting **cancel** from it, the option can be started again by selecting the **pick & create** button.

14.2 Super Alignments

Position of menu: **Strings =>Super alignments**

The **Super Alignment** walk-right menu contains options to create a super alignment string, plus options to created/edit labelling and information styles, and design standards and templates.



create super alignment
super alignment tools
create/edit a super alignment label style
create/edit a style for super alignment information panel
define design standards
create/edit a design template using selections from Design Standards
read in V9 design tables

For the option *Super alignment*, go to	14.2.1 Create Super Alignment
Tools	14.2.2 Super Alignment Tools
Label styles	14.2.3 Super Alignment Style
Info styles	14.2.4 Super Alignment Info Styles
Design standards	14.2.5 Design Standards
Design templates	14.2.6 Design Templates
Import legacy design	14.2.7 Import Legacy Design

14.2.1 Create Super Alignment

Position of option on menu: **Strings =>Super Alignments =>Super alignments**

A super alignment string is defined by specifying both its horizontal and vertical geometry.

If the user does not have the *Alignment Module*, then the *super alignment* acts like the pre-V10 *alignment* string and consists of only horizontal and vertical intersection points consisting of leading transitions-arc-trailing transition for each horizontal intersection point, and parabolas or arcs on the intersection points for vertical geometry. The resultant elements are linked tangentially to form the horizontal and vertical geometry. The intersection points and the tangent points can be moved.

If the user has the *Alignment Module* then the *Super Alignment* includes IP constructions but also allows much more complicated parametric design.

The Horizontal Geometry of a Super Alignment is made up of **parts**, which may be as simple as a straight line between two points or an arc with a given centre and radius, or a horizontal intersection point with given leading and trailing transitions or the **parts** can be very complicated where for example the points of a line are defined by the offset intersection of the two arcs from other strings.

Often a horizontal part is not fully defined (parts are classified as fixed which are fully defined, floating which have one degree of freedom and free which have two degrees of freedom) and the extra restriction that the horizontal parts must be linked tangentially is necessary to fully defined the horizontal geometry. The horizontal geometry is then said to be solved.

Once solved, the horizontal geometry simply consists of linked known segments of types lines, arcs and transition, and so can be drawn in a plan view.

So the horizontal geometry is made up of the constructive definitions called parts, and if the horizontal geometry solves, it also contains the resultant horizontal segments.

Similarly the Vertical Geometry is made up of parts, which are made up of simple or complicated combinations of straight lines, arcs or a parabolas. Again the extra restriction that the vertical parts must be linked tangentially is often necessary to fully defined the vertical geometry. The vertical geometry is then said to be solved and the solved vertical geometry simply consists of linked known segments of types lines, arcs and parabolas, and so can be drawn in a section view.

So the vertical geometry is made up of the constructive definitions called **parts**, and if the vertical geometry solves, it also contains the resultant **vertical segments**.

Important Note

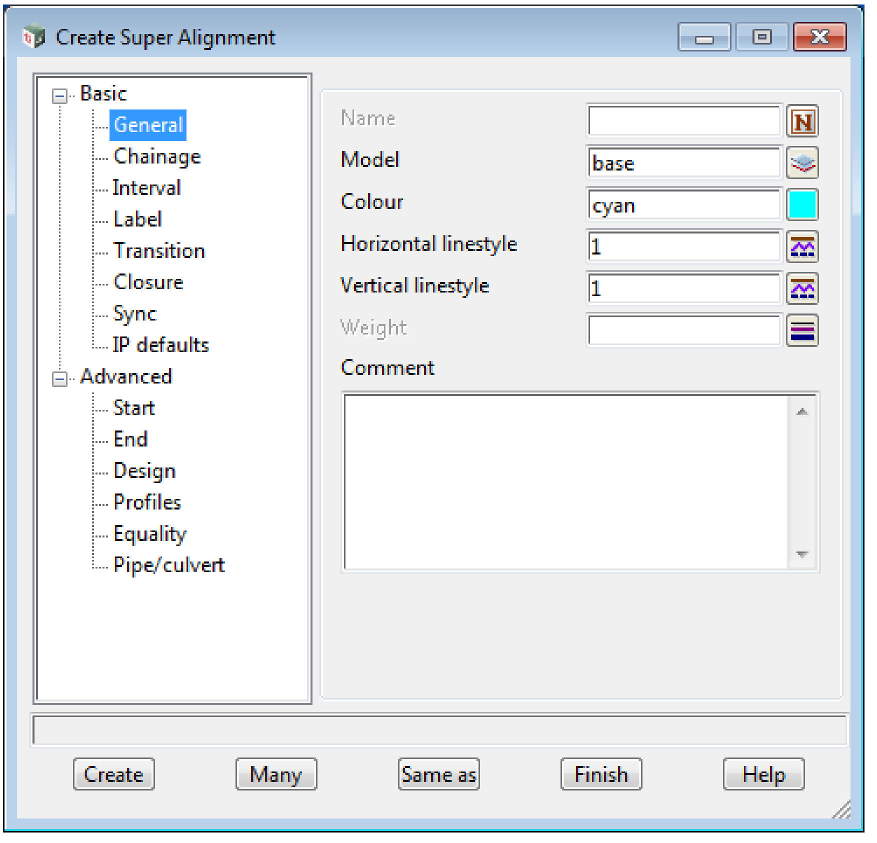
The horizontal and vertical parts of a super alignment are only displayed when the super alignment is being edited. When the super alignment is not being edited, the solved segments are displayed and any unsolved parts are displayed as red crosses.

For more information on Super Alignments and Parts, see 21.9 Placing Parts for Super Alignments.

Notes:

1. vertical geometry can only be added to an **existing** super alignment string. That is, to a string with some horizontal geometry.
2. for both the alignment and super alignment, the horizontal geometry is defined in a **plan** view and the vertical geometry in a ***section*** view.

On selecting the **Super Alignment** option, the **Create Super Alignment** panel is displayed.



To create a new super alignment string, the name, model, colour, *etc.* of the new string are entered into the appropriate fields for each branch on the tree on the left-hand side of the panel, and the **Create** or **Many** button selected.

For information on each branch and the buttons (Create, Many, Same as), and starting to edit after selecting the **Create** or **Many** button, go to:

Basic > General branch

Basic > Chainage branch

Basic > Interval branch

Basic > Label branch

Basic > Transition branch

Basic > Closure branch

Basic > Sync branch

Basic > IP defaults branch

Advanced > Start branch

Advanced > End branch

Advanced > Design branch

Advanced > Profiles branch

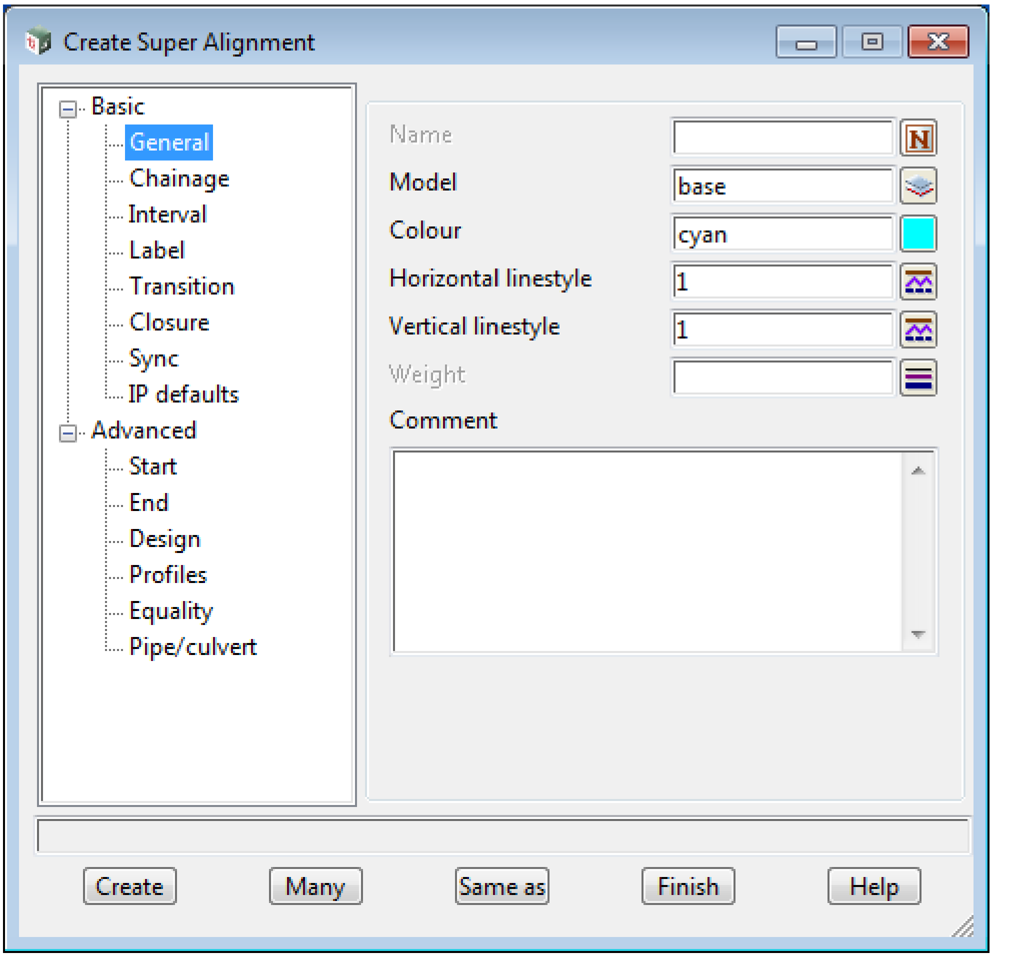
Advanced > Equality branch

Advanced > Chain branch (not yet implemented)

Buttons at Bottom

Start Edit - Super Alignment

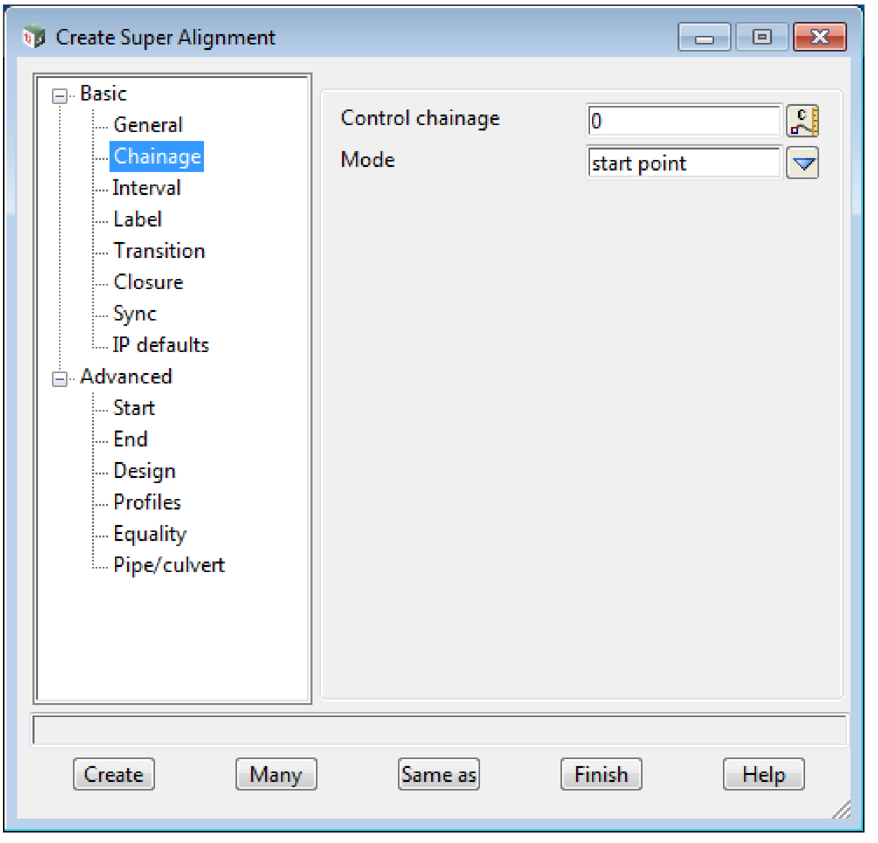
Basic > General branch



The fields and buttons used in the **Basic > General** branch of the **Create Super Alignment** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	name box	from CAD controlbar	available names
the name of the new string			
Model	model box	from CAD controlbar	available models
name of the model that the new string is in			
Colour	colour box	from CAD controlbar	available colours
the colour of the new string. Note that the super alignment may not appear to have this colour if a Label style is set. The Label can define different colours to parts of the super alignment.			
Horizontal Linestyle	linestyle box	from CAD controlbar	available linestyles
linestyle of the string when displayed on a Plan view			
Vertical Linestyle	linestyle box	from CAD controlbar	available linestyles
linestyle of the string when profiled on a Section view			
Weight	weight box	from CAD controlbar	
thickness of the string when plotted			

Basic > Chainage branch



The fields and buttons used in the **Basic > Chainage** branch of the **Create Super Alignment** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Control chainage	measure box	0	Measure Point/String from point

defines the control chainage for the horizontal geometry. What this value refers to depends on the Mode.

Field Description	Type	Defaults	Pop-Up
Mode	choice box		start point end point control point start part end part

if ***start point****, the* Control chainage *value is applied to the start of the usable horizontal geometry.*

If ***end point****, the* Control chainage *value is applied to the end of the usable horizontal geometry.*

If ***control point****, a user given control point is dropped perpendicularly onto the horizontal geometry and that position is given the* Control chainage *value.*

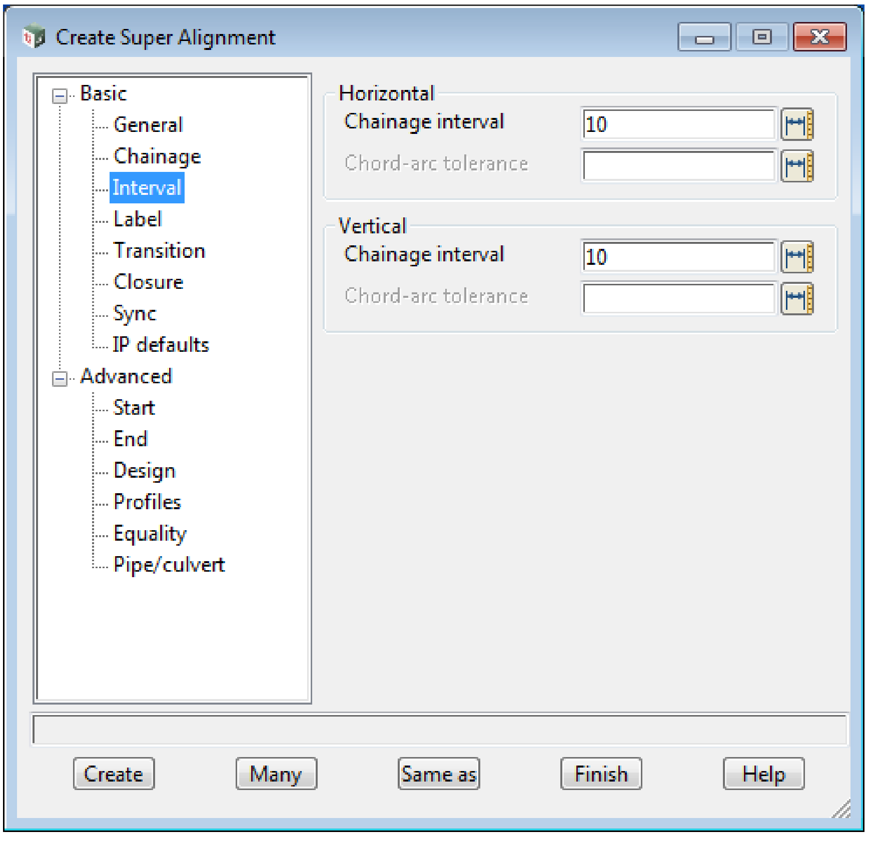
Control point X/Y coordinate select x, y box

coordinates of the control point

If ***start part****, the* Control chainage *value is applied to the start of the first part of horizontal geometry, even if it is invisible.*

If ***end part****, the* Control chainage *value is applied to the end of the last part of horizontal geometry, even if it is invisible.*

Basic > Interval branch



The fields and buttons used in the **Basic > Interval** branch of the **Create Super Alignment** panel have the following functions.

Field Description Type Defaults Pop-Up

Horizontal Chainage interval measure box Point to point/String to point

horizontal chainage interval for the string

If blank, the default is 10.

Horizontal Chord to arc tol measure box Point to point/String to point

horizontal chord to arc tolerance to use for any arcs and transitions

If blank, the default is 0.1.

Vertical Chainage interval measure box Point to point/String to point

vertical chainage interval for the string

If blank, the default is 10.

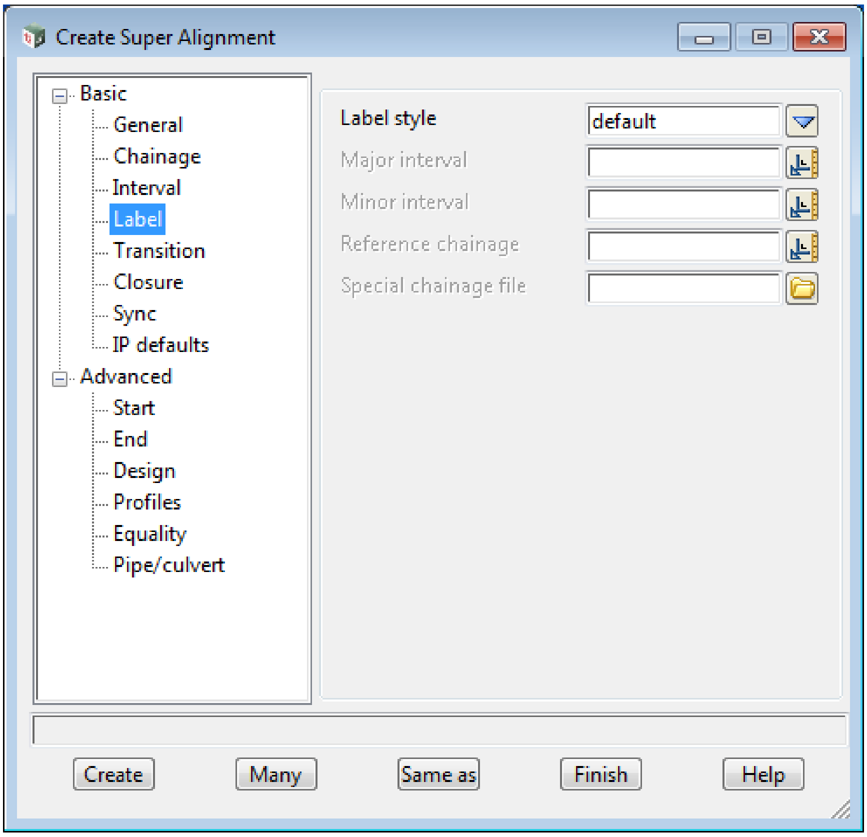
Vertical Chord to arc tol measure box Point to point/String to point

vertical chord to arc tolerance to use for any parabolas or circular curves.

If blank, the default is 0.1.

***Note**: the horizontal and vertical intervals and chord to arc tolerances are used when the super alignment is approximated by straights in three dimensions.*

Basic > Label branch



The fields and buttons used in the **Basic > Label** branch of the **Create Super Alignment** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Label style	choice box	default	available label styles

The super alignment style controls the way the super alignment draws and highlights on the screen.
For more information please go to the section 14.2.3 Super Alignment Style

Label Major Chainage interval measure box

major chainage interval to use for labelling

If blank, the default is 10.

Label Minor Chainage interval measure box

minor chainage interval to use for labelling

Label Reference chainage interval measure box

reference value for the chainage labelling intervals

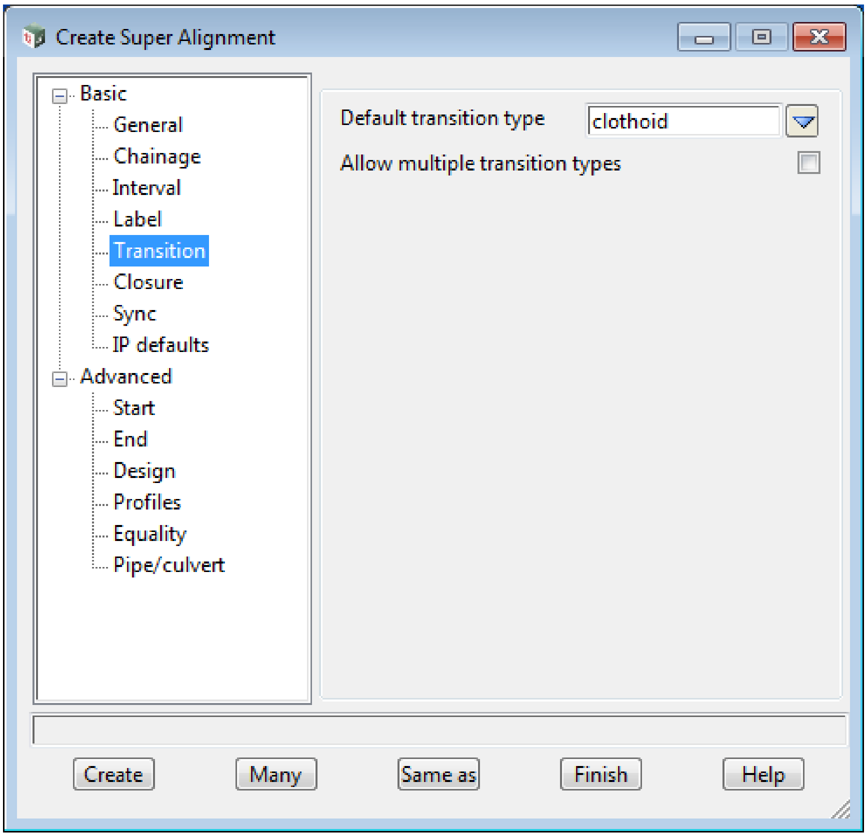
Label Special chainage file file box available *.spc files

file of special chainages for labelling

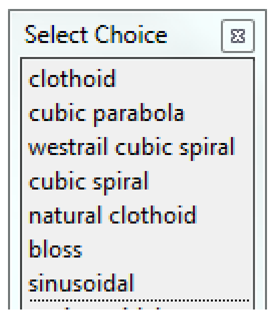
***Note**: the labelling of the super alignment is given by the **Label style** on the **General** tab.*

For more information please go to the section 14.2.3 Super Alignment Style

Basic > Transition branch

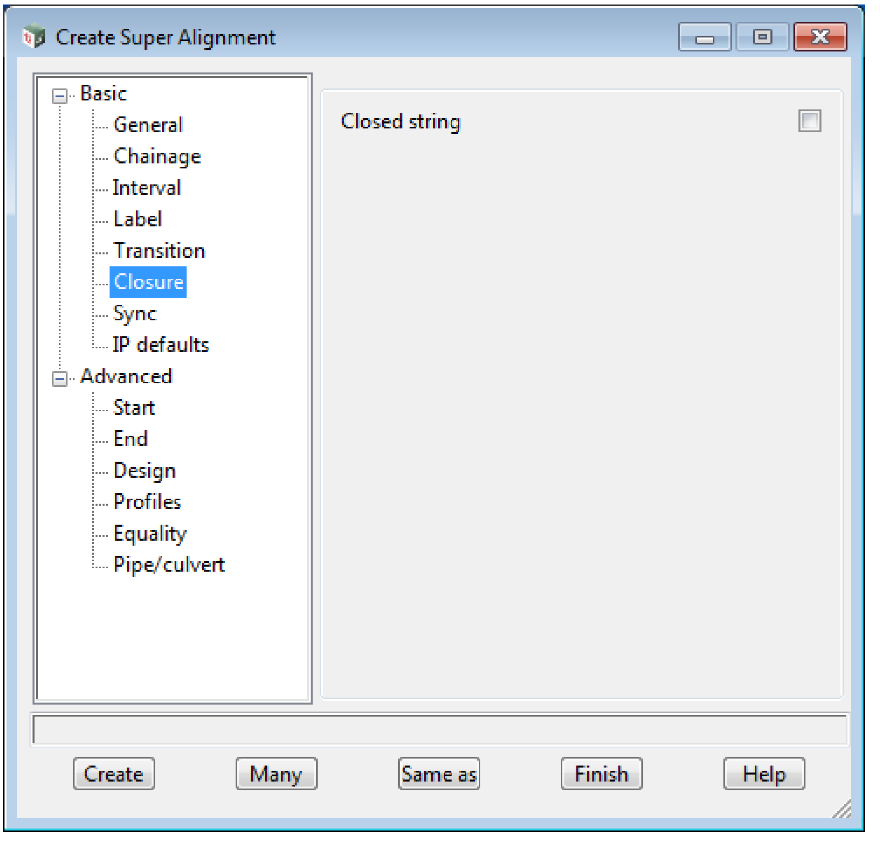


The fields and buttons used in the **Basic > Transition** branch of the **Create Super Alignment** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Transition type	choice box	clothoid	

transition (for example clothoid) to be used for this super alignment string.
For more information see 39.2.7.5 Transitions and Spirals File.

Basic > Closure branch

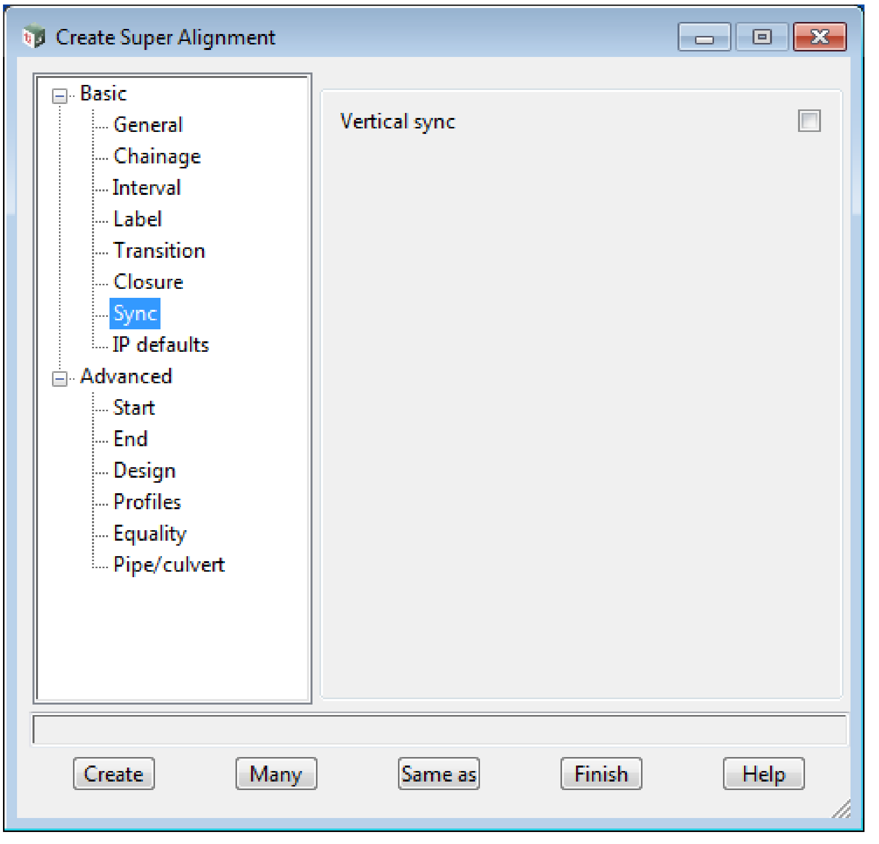


The fields and buttons used in the **Basic > Closure** branch of the **Create Super Alignment** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Closed string	tick box	not ticked	

*if **ticked**, the super alignment is automatically closed*

Basic > Sync branch



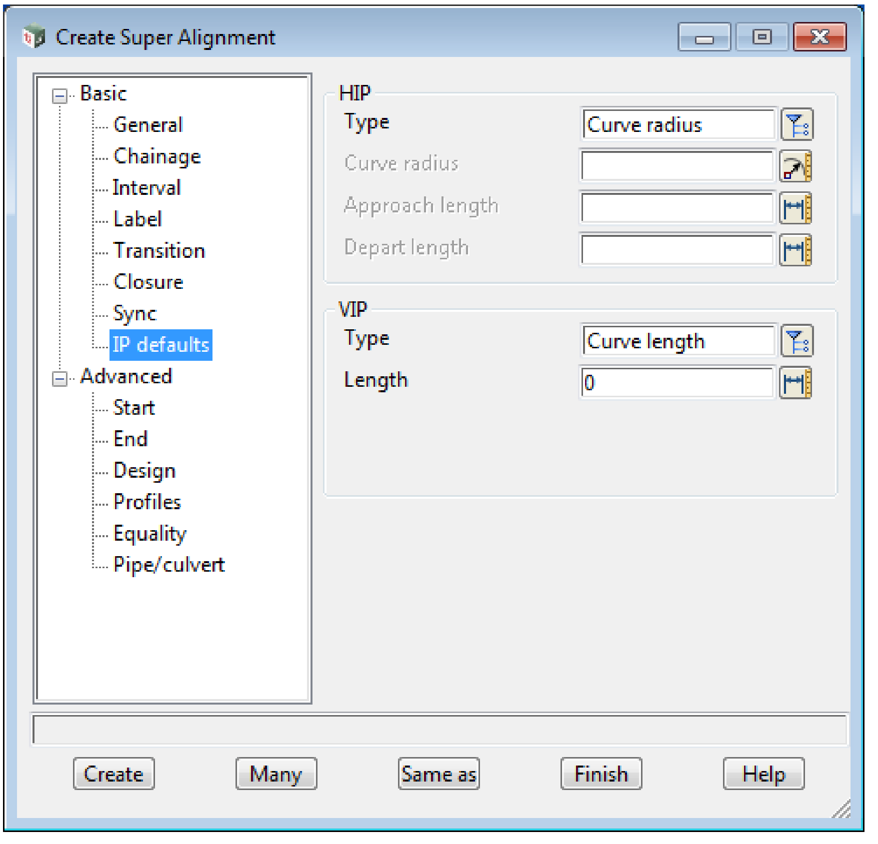
The fields and buttons used in the **Basic > Sync** branch of the **Create Super Alignment** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Vertical sync	tick box	not ticked	

*if **ticked**, when the horizontal geometry is modified, an attempt is made to modify the chainage position of the vertical geometry so that it remains locked to the same parts of the horizontal geometry as before the horizontal geometry was modified.*
*If **not ticked**, after horizontal geometry is modified, the vertical geometry will possibly be moved in relation to the horizontal geometry because the (x,y) position for a given chainage will change.*

Basic > IP defaults branch

*the **IP Defaults** branch defines the types of HIPs and VIPs that are placed when laying down horizontal and vertical geometry.*

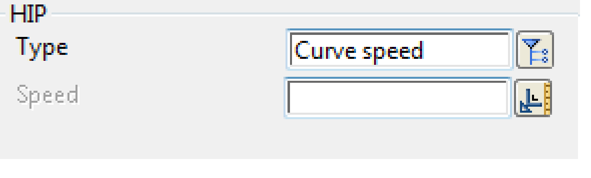


The fields and buttons used in the **Basic > IP defaults** branch of the **Create Super Alignment** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
HIP type	choice box	Curve radius	Curve speed/radius/length

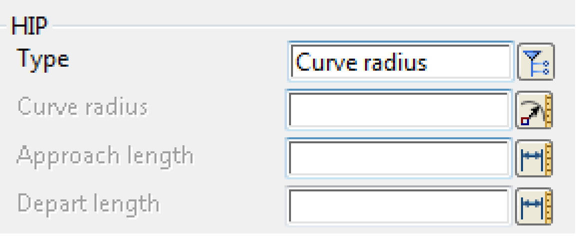
The default HIP type given to any placed HIPs. The default for the panel is a Curve radius *with blank Curve radius, Approach and Depart length. That is, just a HIP with no curve or transitions on it.*

*If **Curve speed**, any new HIP being created will be a Speed IP with the speed specified in the* Speed *field.*



If the Speed *field is blank, the default alignment speed is used.*

*If **Curve radius**, any new HIP being created will be a Radius IP with the radius of the arc on the HIP specified in the* Curve Radius *field. Optional approaching and departing transitions can also be defined in the* Approach length *and* Depart length *fields.*



If the Curve Radius *field is non-zero, an arc of the given value is placed on the HIP.*

If the Curve Radius *field is blank, the HIP won't have an arc associated with it.*

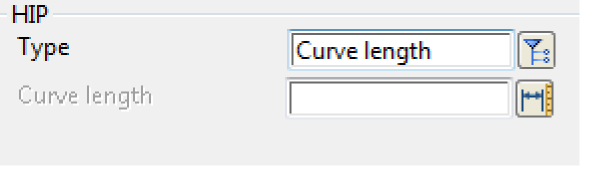
If the Approach length *field is non-zero, then a leading transition of the given value is placed on the HIP. If there is a non-zero value then there must also be a non-zero* Curve Radius *value.*

If the Depart length *field is non-zero, then a trailing transition of the given value is placed on the HIP. If there is a non-zero value then there must also be a non-zero* Curve Radius *value.*

***Note** - if either the* Approach length *or* Depart length *fields are non zero then there must be a non zero* Curve radius.

If the HIP is defined by **Curve radius***, then as the HIP is moved around, the values in the Curve radius definition are held for the HIP. That is, the radius of the arc and the transition lengths are kept constant and the length of the arc is modified.*

If ***Curve length****, any new HIP being created will be a Length IP with the length specified in the* Curve length *field.*



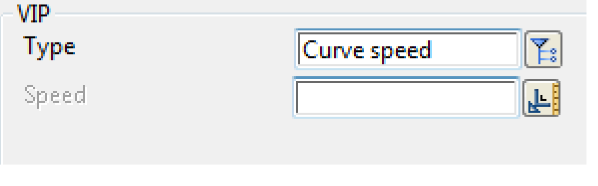
If the Curve length *field is blank, the HIP will have an arc of length zero.*

If the HIP is defined by **Curve length***, then as the HIP is moved around, the arc length is held for the arc on the HIP and arc radius is modified.*

VIP type	choice box	Curve length	Curve speed/radius/length Asymmetric

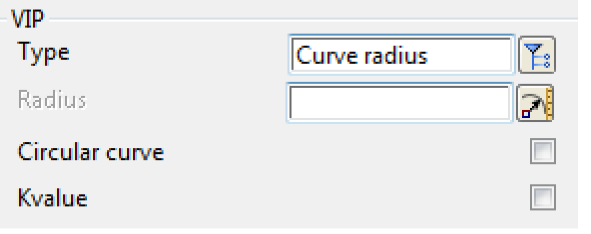
The default VIP type given to any placed VIPs. The default for the panel is a Curve length *with a* Length *of 0.*

If ***Curve speed****, any new VIP being created will be a Speed IP with the speed specified in the* Speed *field.*



If the Speed *field is blank, the default alignment speed is used.*

*If **Curve radius**, any new VIP being created will be a Radius IP with the radius specified in the* Radius *field.*



If the Radius *field is blank, the VIP won't have an parabola/arc associated with it. The value in the Radius field can be a k-value for a parabola, an effective radius (100 x k-value) for a parabola, or the radius of an arc.*

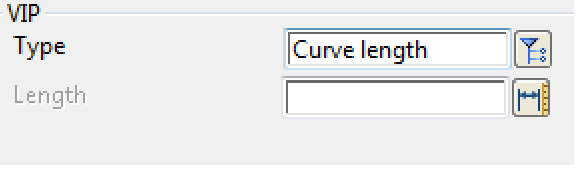
If the Kvalue *tick box is ticked, a parabola is placed on the VIP and the k-value of the parabola is the value in the* Radius *field.*

If the Circular curve *tick box is ticked, an arc is placed on the VIP and the radius of the arc is the value in the* Radius *field.*

*If **neither** Kvalue or Circular curve is ticked, a parabola is placed on the VIP and the effective radius of the parabola is the value in the* Radius *field.*

If the VIP is defined by **Curve radius***, then as the VIP is moved up or down, the values in the Curve radius definition is held for the VIP. That is, the radius or k-value will be kept constant and the length of the parabola/arc is varied.*

*If **Curve length**, any new VIP being created will be a parabolic Length IP with the length specified in the* Length *field.*

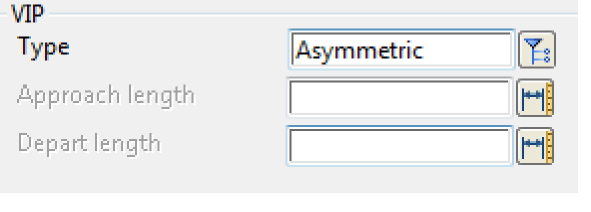


If the Length *field is 0, then just the VIP will be drawn with no parabola on it.*

If the Length *field is blank, the VIP will be associated with an parabola of maximum length.*

If the VIP is defined by **Curve length***, then as the VIP is moved up or down, the length of the parabola given in the Curve Length definition is held for the VIP. That is, the length is kept constant and the radius/k-value of the parabola is varied.*

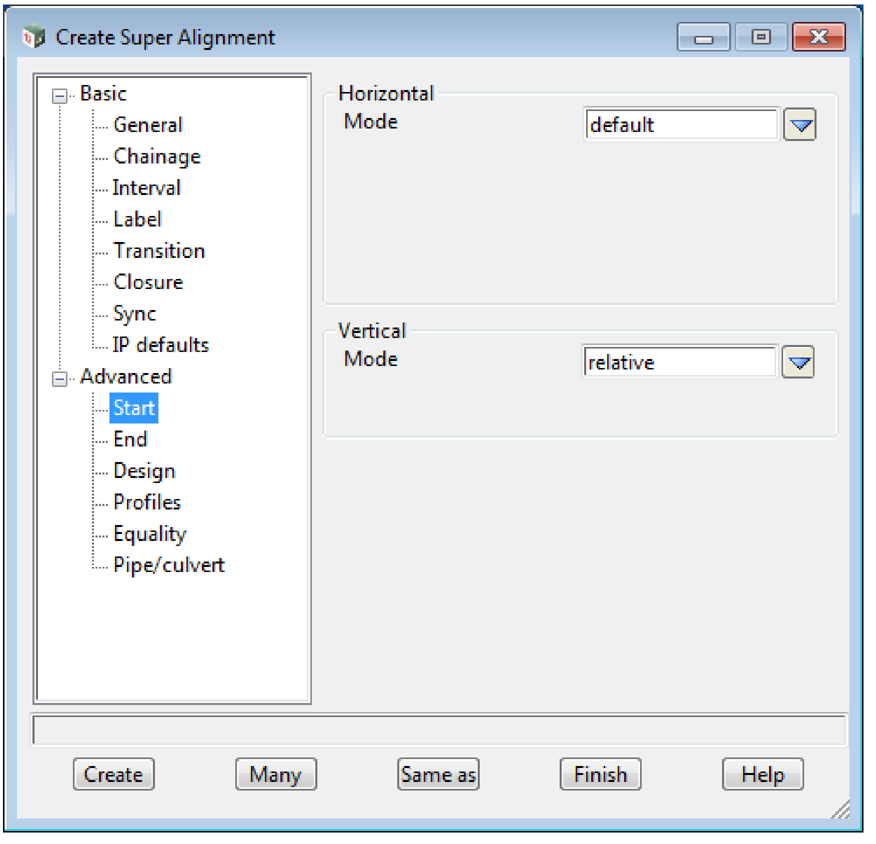
*If **Asymmetric**, any new VIP being created will have parabolas of the combined lengths of* Approach length *and* Depart length*.*



If the VIP is defined by **Asymmetric***, then as the VIP is moved up or down, the Approach and Depart Lengths of the parabolas given in the Asymmetric definition are held for the VIP. That is, the lengths are kept constant and the radius/k-value of the parabolas are varied.*

Advanced > Start branch

*the **Start** branch defines where the usable horizontal and vertical geometry starts for the super alignment. Note that the super alignment can have more horizontal and vertical geometry than is actually used.*



The fields and buttons used in the **Advanced > Start** branch of the **Create Super Alignment** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Start Horizontal mode	choice box		default segment point

specifies how the start of the usable horizontal geometry is defined.

*If **default**, the beginning of the first visible horizontal part is the start of the usable horizontal geometry.*

*If **segment**, a user selected segment of the horizontal geometry is the start of the usable horizontal geometry.*

*If **point**, a user selected position is dropped perpendicularly onto the super alignment to define the start of the usable horizontal geometry.*

Start Control point X/Y coordinate select x, y box

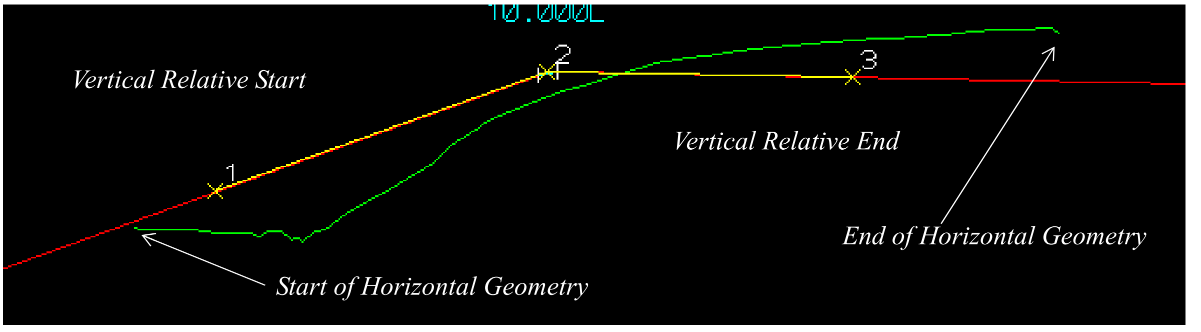
*used in the **segment** and **point** cases of the **Start Horizontal mode**.*

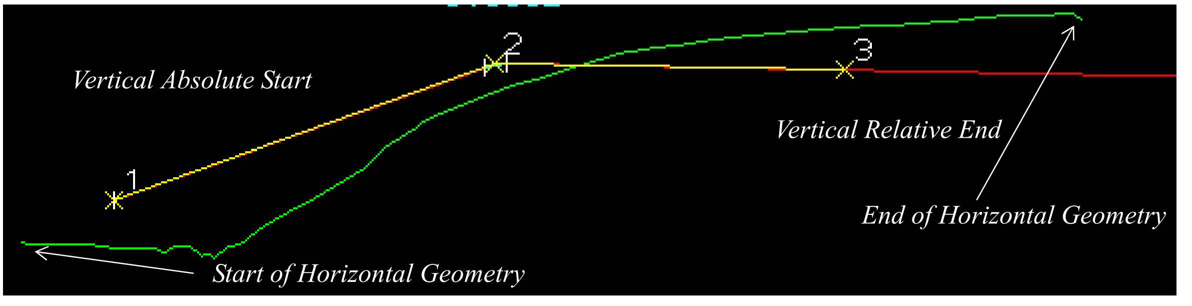
Start Vertical mode	choice box		absolute relative chainage

specifies how the start of the usable vertical geometry is defined.

*If **absolute**, the beginning of the first visible vertical part is the start of the usable vertical geometry.*

*If **relative**, the chainage of the start of the usable horizontal geometry is used as the start of* *the usable vertical geometry. If the beginning of the defined vertical geometry is **after** this chainage, then the first segment is extrapolated back to the required chainage.*





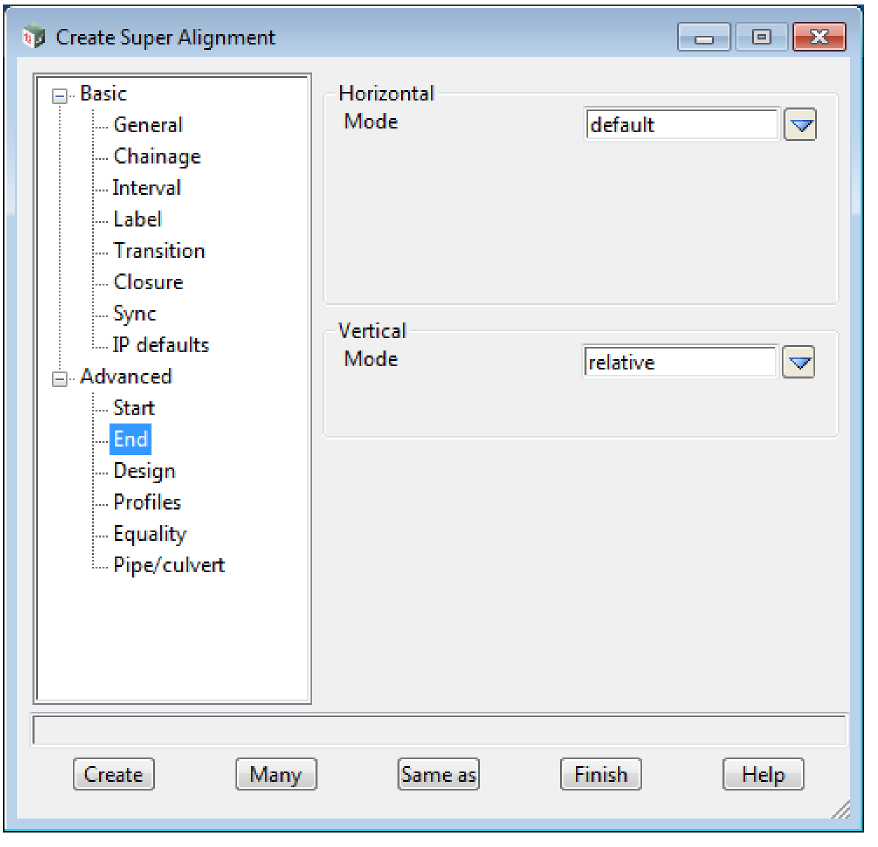
*If **chainage**, a user given chainage defines the start of the usable vertical geometry.*

Chainage measure box available measures

*used in the **chainage** case of the **Start Vertical mode**.*

Advanced > End branch

*the **End** branch defines where the usable horizontal and vertical geometry ends for the super alignment. Note that the super alignment can have more horizontal and vertical geometry than is actually used.*



The fields and buttons used in the **Advanced > End** branch of the **Create Super Alignment** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
End Horizontal mode	choice box		default segment point

specifies how the end of the usable horizontal geometry is defined.

*If **default**, the end of the last visible horizontal part is the end of the usable horizontal geometry.*

*If **segment**, a user selected segment of the horizontal geometry is the end of the usable horizontal geometry.*

*If **point**, a user selected position is dropped perpendicularly onto the super alignment to define the end of the usable horizontal geometry.*

End Control X/Y coordinate select x, y box

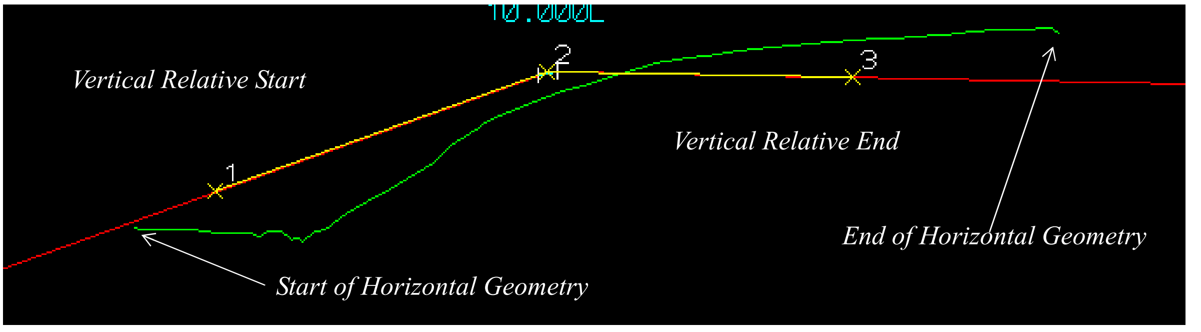
*used in the **segment** and **point** cases of the **End Horizontal mode**.*

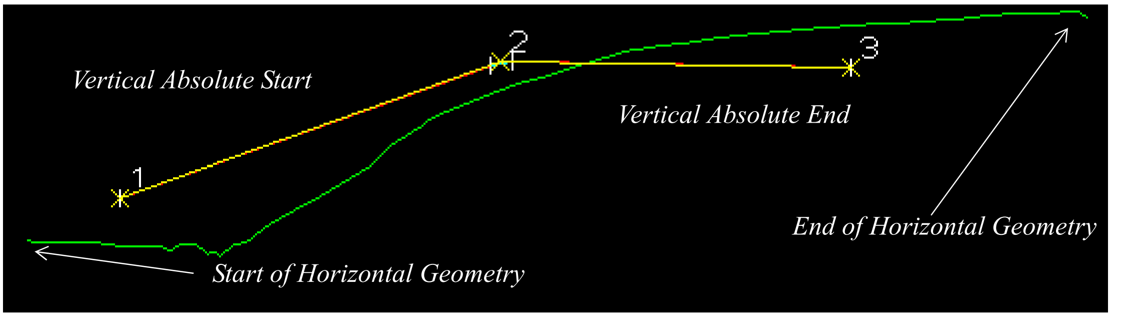
Field Description	Type	Defaults	Pop-Up
End Vertical mode	choice box		absolute relative chainage

specifies how the end of the usable vertical geometry is defined.

*If **absolute**, the end of the last visible vertical part is the end of the usable vertical geometry.*

*If **relative**, the chainage of the end of the usable horizontal geometry is used as the end of the* *usable vertical geometry. If the end of the defined vertical geometry is* ***before*** *this chainage, then the last segment is extrapolated to the required chainage.*



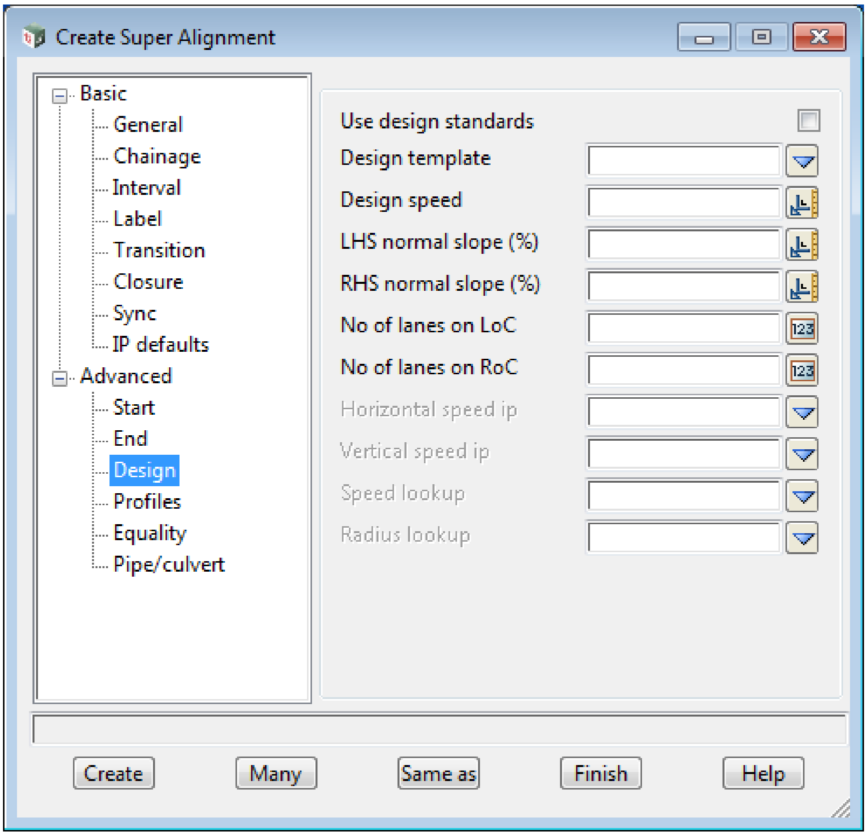


If ***chainage****, a user given chainage defines the end of the usable vertical geometry.*

Chainage measure box available measures

used in the ***chainage*** *case of the* ***End Vertical mode****.*

Advanced > Design branch



*the **Design** branch is only used if a design template has been set up and is to be used for automatically inserting curves and transition curves in horizontal geometry, apply super elevation and widening, and automatically inserting vertical curves in the vertical geometry.*

The fields and buttons used in the **Advanced > Design** branch of the **Create Super Alignment** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use design standards	tick box	not ticked	

*if **ticked**, the selected design template is used when creating horizontal and vertical geometry.*
*If **not ticked**, no design template is used with **Alignment speed** elements.*

Design template	choice box		available design templates

Design template to use when creating horizontal and vertical geometry. For more information see 14.2.6 Design Templates.

Design speed			*.design files

the general design speed for the road. This is used with the design template when creating horizontal and vertical geometry. This can be over written by Speed IPs.

LHS normal slope (%)

RHS normal slope (%)

No of lanes on LoC

number of lanes on the left of the centreline. The number of lanes can be zero.

No of lanes on RoC

number of lanes on the right of the centreline. The number of lanes can be zero.

Horizontal speed IP	choice box	minimum maximum

*For **Alignment speed** and **Curve speed HIPs**.*

*If **minimum**, when an arc is automatically placed on an HIP using the design table, the arc with the smallest radius for the design speed is used.*
*If **maximum**, when an arc is automatically placed on an HIP using the design table, the arc with the largest radius for the design speed is used.*

Vertical speed IP	choice box	minimum maximum

*For **Alignment speed** and **Speed VIPs**.*

*If **minimum**, when a parabola is automatically placed on an VIP using the design table, the parabola with the smallest kvalue for the design speed is used.*
*If **maximum**, when a parabola is automatically placed on an VIP using the design table, the parabola with the largest kvalue for the design speed is used.*

Speed lookup	choice box	ceiling floor round up round down

Different method to achieve the actual speed from the speed table.

*If **ceiling**, then the design speed is the upper speed from the table.*

*If **floor**, then the design speed is the lower speed from the table.*

*If **round up**, the speed will be the upper speed from the table when the calculated speed is greater than or equal to the half way speed. Otherwise it will be the lower speed.*

*If **round down**, the speed will be the lower speed from the table when the calculated speed is less than or equal to the half way speed. Otherwise it will be the upper speed.*

Radius lookup	choice box	ceiling floor round up round down interpolate

Different method to achieve the actual radius from the radius table.

*If **ceiling**, then the design radius is the upper radius from the table.*

*If **floor**, then the design radius is the lower radius from the table.*

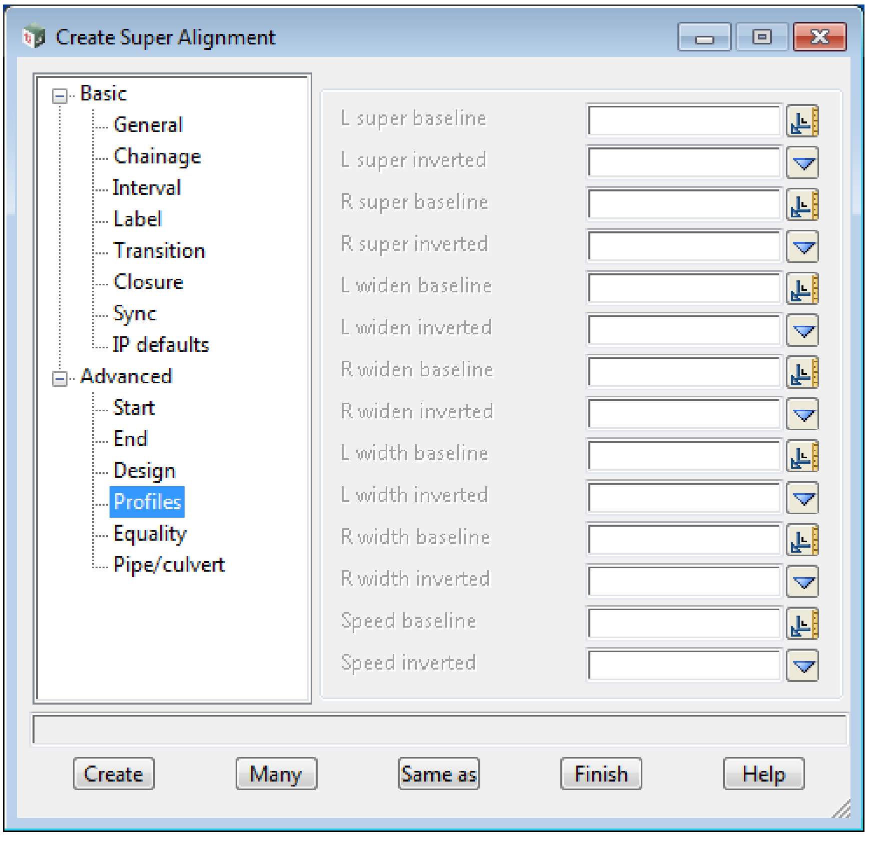
*If **round up**, the radius will be the upper radius from the table when the calculated radius is greater than or equal to the half way radius. Otherwise it will be the lower radius.*

*If **round down**, the radius will be the lower radius from the table when the calculated radius is less than or equal to the half way radius. Otherwise it will be the upper radius.*

*If **interpolate**, the radius is the calculated based on the specified speed.*

*For **non speed HIPs only** (that is, not **Alignment speed** or **Curve speed HIPs).***
The radius has been defined by the user and LJG?

Advanced > Profiles branch



The fields and buttons used in the **Advanced > Profiles** branch of the **Create Super Alignment** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
L super baseline	measure box		available measures

the left super elevation diagram will be drawn at this Z value in the Section view

L super inverted	choice box	no	yes, no

*if **no**, positive super elevation is drawn above the left super baseline and negative super elevation is drawn below the left super baseline. This is the default.*

*if **yes**, positive super elevation is drawn below the left super baseline and negative super elevation is drawn above the left super baseline, i.e. the diagram is turned upside down*

R super baseline	measure box		available measures

the right super elevation diagram will be drawn at this Z value in the Section view

R super inverted	choice box	no	yes, no

*if **no**, positive super elevation is drawn above the right super baseline and negative super elevation is drawn below the right super baseline. This is the default.*

*if **yes**, positive super elevation is drawn below the right super baseline and negative super elevation is drawn above the right super baseline, i.e. the diagram is turned upside down*

L widen baseline	measure box		available measures

the left widening diagram will be drawn at this Z value in the Section view

L widen inverted	choice box	no	yes, no

*if **no**, widening is drawn above the left widen baseline. Widening is always positive (or zero). This is the* *default.*

*if **yes**, positive widening is drawn below the left widen baseline, i.e. the diagram is turned upside down*

R widen baseline	measure box		available measures

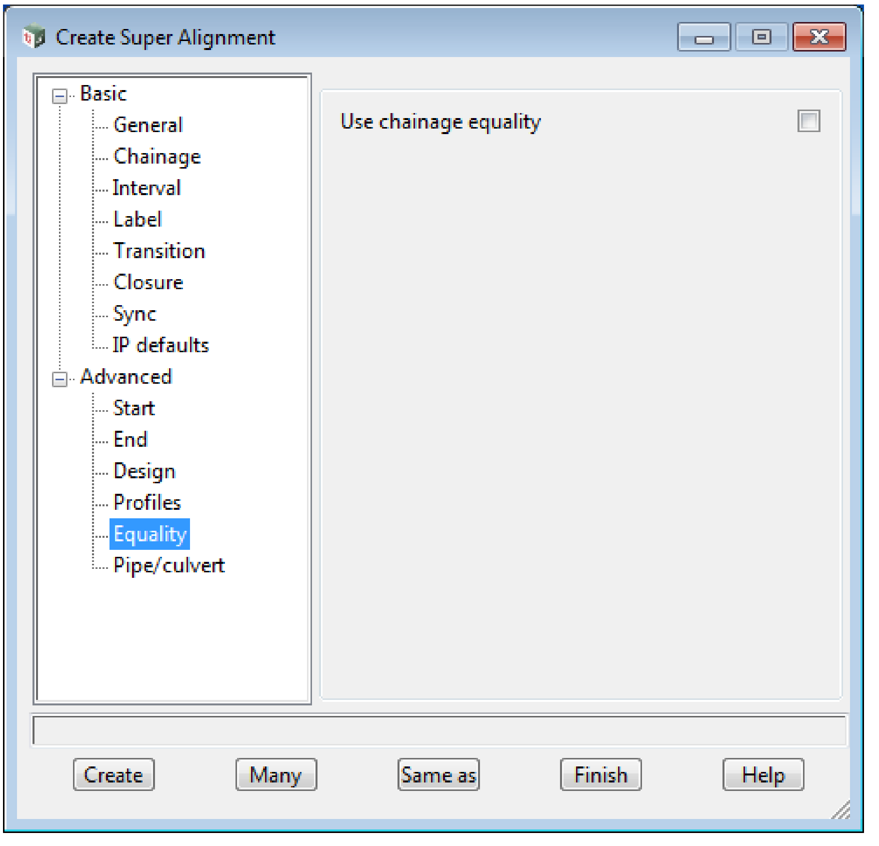
the right widening diagram will be drawn at this Z value in the Section view

R widen inverted	choice box	no	yes, no

*if **no**, widening is drawn above the right widen baseline. Widening is always positive (or zero). This is the default.*

*if **yes**, positive widening is drawn below the right widen baseline, i.e. the diagram is turned upside down*

Advanced > Equality branch



The fields and buttons used in the **Advanced > Equality** branch of the **Create Super Alignment** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use chainage equality	tick box	not ticked	

*if **ticked**, chainage equalities can be defined and the automatic labelling of the string will use the equalities.*

Advanced > Chain branch (not yet implemented)

The fields and buttons used in the **Advanced > Chain** branch of the **Create Super Alignment** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Chain file	file box		available *.chain files

the chain parameter file to run every time the super alignment is resolved/recalced.

For more information see the section 28.3 Chains *in the book* 28 Utilities.

Run chain after set	tick box	not ticked	

Buttons at Bottom

Create button

when only one super alignment is to be created, click on the **Create** *button. The* **Edit Super Alignment** *toolbar is displayed.*

Many button

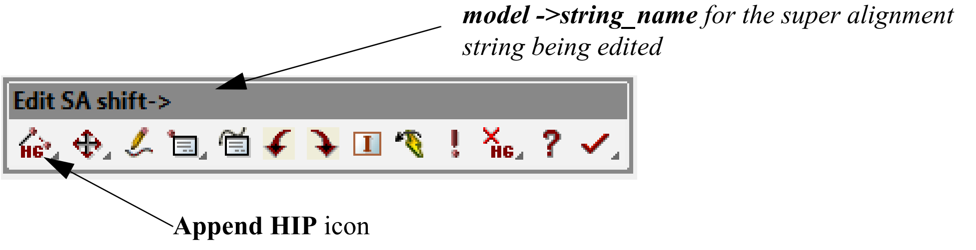
when more than one super alignment is to be created using the same set-up information, then click on the **Many** *button.* **Edit Super Alignment** *toolbar is displayed.*
After the editing for this string is finished, a new ***Create Super Alignment*** *panel is placed on the screen with the same values for the panel fields as the string just created.*
Hence new super alignments can be created (by clicking on the **Create** *or* **Many** *button) without having to go back to the* ***String Create*** *menu.*
If no more super alignments are to be created, click on the **Finish** *button*

Same as button

after the **Same as** *button is chosen, another string is selected and information about it is used to fill in fields for this panel*

Start Edit - Super Alignment

After selecting the **Create** or **Many** button, the **Super Alignment** toolbar is created and placed on the screen.



To create a new Super Alignment, select the **Append HIP** icon.

The **Append HIP** simply creates horizontal intersection points (HIPs) that are joined by a straight segments.

For Appending HIPs, a cross is then drawn in each plan view that the string's model is on, and the cross follows the cursor around the screen. If the string's model is not added to any plan view, the model is automatically added to all plan views.

After the cross is on the screen (moving with the cursor), clicking LB and accepting with MB selects the first HIP of the string (using the appropriate snaps).

The string is then drawn from the first HIP to the cursor position, which represents the second HIP of the string. Clicking LB and accepting with MB selects the second string HIP and the process repeats for subsequent string HIPs.

Now that the string is created, all the edit option on the string's **Edit SA** toolbar are usable.

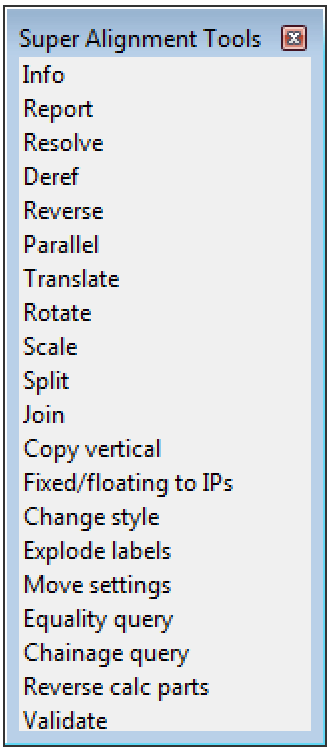
The options in the **Edit SA** toolbar are not only used for placing the initial HIPs of the string, but for editing the super alignment once it is created. Since the **Append HIP** and other options in the **Edit SA** toolbar are identical to the options used when editing an existing string, they will be discussed in detail in the **Editor** section.

For full information on the Super Alignment Editor, go to the section 14.4.11 Edit Super Alignment

14.2.2 Super Alignment Tools

Position of Menu: **Strings =>Super alignments =>Tools**

The Super Alignment Tools walk-right menu is



information about a super alignment
create a report on a super alignment
resolve a super alignment
dereference a super alignment
reverse the super alignment direction
parallel a super alignment
translate a super alignment
rotate a super alignment
scale a super alignment
split a super alignment
join two super alignments
copy the vertical geometry between super alignments
convert fixed/floating to IPs
change the style for super alignments
explode the labelling to text and lines
setting to lock the movement of IPs
inquire about chainage equalities
inquire about chainages
make construction parts from super alignment segments
run validation on a super alignment

See

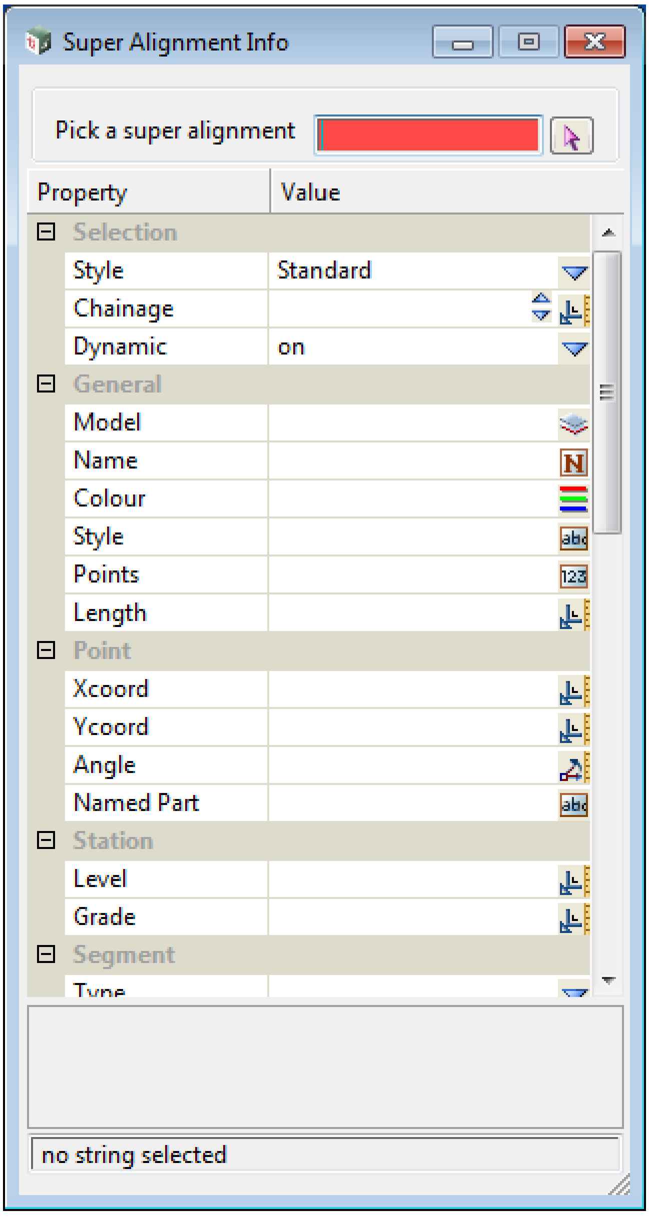
Info	14.2.2.1 Super Alignment Info
Report	14.2.2.2 Super Alignment Report
Resolve	14.2.2.3 Resolve a Super Alignment
Deref	14.2.2.4 Dereference Super Alignments
Reverse	14.2.2.5 Reverse Super Alignment
Parallel	14.2.2.6 Parallel a Super Alignment
Translate	14.2.2.7 Translate Super Alignment
Rotate	14.2.2.8 Rotate Super Alignment
Scale	14.2.2.9 Scale Super Alignment
Split	14.2.2.10 Split a Super Alignment
Join	14.2.2.11 Joining Two Super Alignments
Copy vertical	14.2.2.12 Copy Vertical Geometry
Fixed/floating to IPs	14.2.2.13 Convert Fixed/Floating Elements to IPs
Change style	14.2.2.14 Change the Styles for Selected Super Alignments
Explode labels	14.2.2.15 Explode the Super Alignments Labelling
Move settings	14.2.2.16 Constrain the Movement of Super Alignment Points
Equality query	14.2.2.17 Equality Query
Chainage query	14.2.2.18 Chainage Query
Reverse calc parts	14.2.2.19 Reverse Calc Parts
Validate	14.2.2.20 Validate Super Alignment

14.2.2.1 Super Alignment Info

Position of option on menu: **Strings =>Super alignments =>Tools =>Info**

This section of documentation is a work in progress and will be updated in subsequent releases.

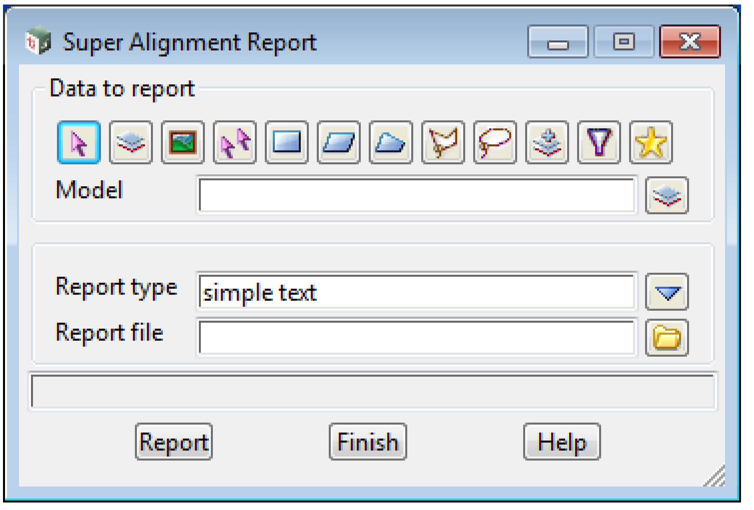
Selecting **Info** displays the **Super Alignment Info** panel.



14.2.2.2 Super Alignment Report

Position of option on menu: **Strings =>Super alignments =>Tools =>Report**

Selecting **Report** displays the **Super Alignment Report** panel.



For more information, please see 27.2.4 Super Alignment Report.

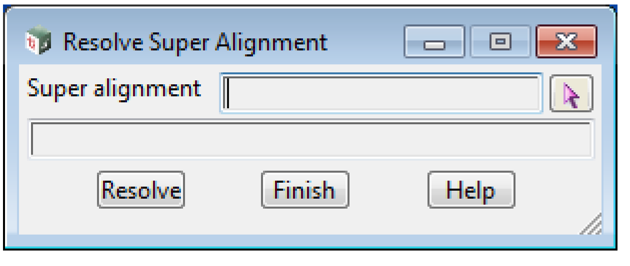
14.2.2.3 Resolve a Super Alignment

Position of option on menu: **Strings =>Super alignments =>Tools =>Resolve**

When a super alignment is created using *computators*, the super alignment is defined with *references* to other strings. If the referenced strings are modified, then the super alignment will not automatically reflect the changes in the referenced strings.

Resolve forces the string to be resolved using the current position and properties of any referenced strings.

Selecting **Resolve** displays the **Resolve Super Alignment** panel.



The new fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Super alignment	string select		

select a super string to resolve.

Resolve	button		

*after the **Resolve** button is clicked, the super alignment is resolved using the new position and properties of any referenced strings.*

14.2.2.4 Dereference Super Alignments

Position of option on menu: **Strings =>Super alignments =>Tools =>Deref**

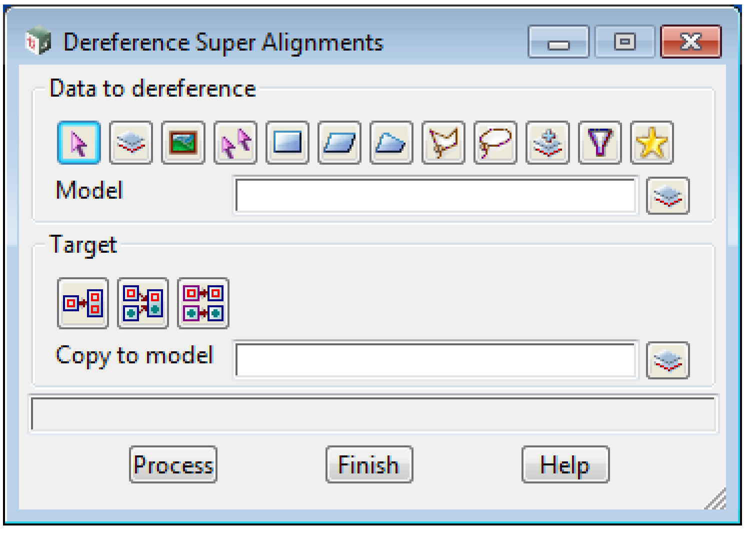
When a super alignment is created using *computators*, the super alignment is defined with *references* to other strings.

Dereferencing removes the *references* to other strings in a super alignment, and inserts copies of the referenced string in its place. Hence a dereferenced string looks like the original string, but it no longer has links to any other strings.

The **Deref** option copies the original string, and *dereferences* the copied string.

So after running **Deref** there are no longer any references to other strings in the copy of the super alignment.

Selecting **Deref** displays the **Dereference Super Alignments** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data to dereference		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data info	input		

extra information required for the Data source.

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target.

Target info	input		

extra information required for the target. Only the Copy Targets are allowed.

Process	button		

after the **Process** *button is clicked, the selected strings are copied and dereferenced.*

14.2.2.5 Reverse Super Alignment

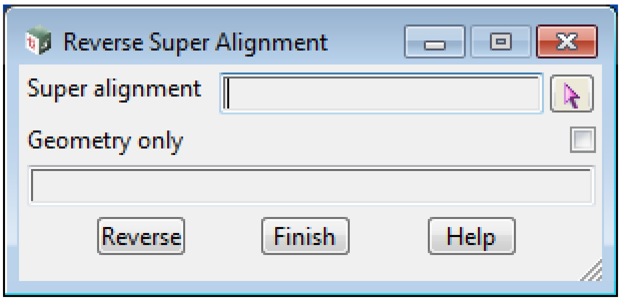
Position of option on menu: **Strings =>Super alignments =>Tools =>Reverse**

Reverse creates a copy of the selected super alignment, and then reverses the direction of the copied super alignment.

If there are any computators in the super alignment, the copy of the super alignment will be dereferenced before being reversed.

Even if the super alignment will not solve, it can still be reversed. This is useful when strings come in from other systems and have geometry (that is horizontal and vertical segments) but won't solve. These strings can now be reversed.

Selecting **Reverse** displays the **Reverse Super Alignment** panel.



The new fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Super alignment	string select		

select the super alignment to copy and reverse the direction of the copied string

Geometry only	tick box	not ticked	

*if **ticked** then only the Geometry of the Super Alignment (the segments for the horizontal and the segments for the vertical) is used, and a reverse string is created by simply reversing the segments. All construction data is removed. So this will work for super alignments that have geometry but won't solve.*

The reversed string no longer has any constructive definition but just consists of the horizontal and vertical data.

This is useful when strings come in from other systems and have geometry (that is, horizontal and vertical segments) but won't solve.

*If **not ticked** then the string will need to solve before it can be reversed.*

***Note 1**: This reverses and replaces the existing string so if you don't want to lose the data in your existing string, make a copy of it first.*

***Note 2:** The horizontal segments can be straights, arcs, transitions and offset transitions. The vertical segments can be straights, arcs and parabolas.*

Reverse	button		

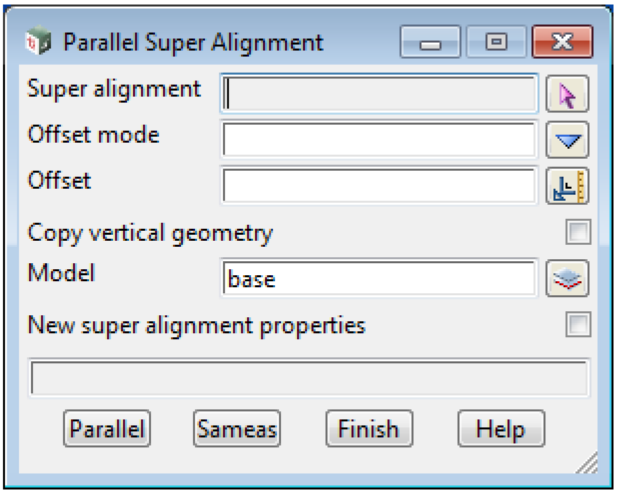
*after the **Reverse** button is clicked, the direction of the copied super alignment is reversed.*

14.2.2.6 Parallel a Super Alignment

Position of option on menu: **Strings =>Super alignments =>Tools =>Parallel**

Parallel parallels the horizontal geometry of a super alignment by a user given offset.

Selecting **Parallel** displays the **Parallel Super Alignment** panel.



The new fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Super alignment	string select		

select a super alignment to parallel

Offset mode	choice box		fix and free, floating, ip

*if **fix and free**, fixed free fixed elements will be used to parallel the super alignment.*
*If **floating**, the super alignment will start with a fixed element followed by floating elements.*
*If **ip**, an IP based super alignment will be created.*

Offset	input		

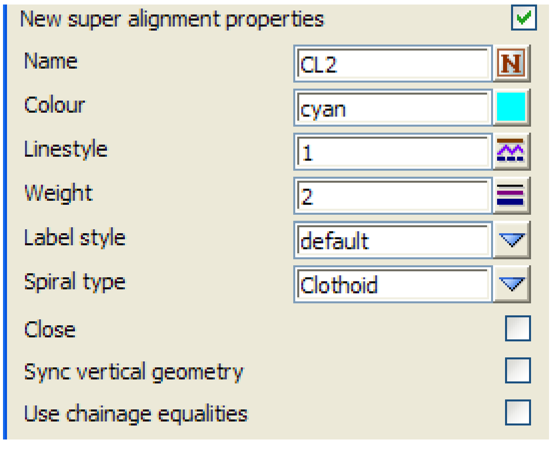
distance to offset the selected super alignment

Model	model box		available models

if not blank, the model for the paralleled string.
If blank, the model is the same as the selected string.

New super alignment properties tick box

*if **ticked**, the panel is expanded to show all the properties that can be set for the new paralleled string.*



Extra panel fields when ***New super alignment properties*** is ticked on

Name	input	from CAD controlbar	

the name of the new string

Colour	input	from CAD controlbar	available colours

the colour of the new string

Linestyle	linestyle box	from CAD controlbar	available line styles

line style of the string

Weight	input		

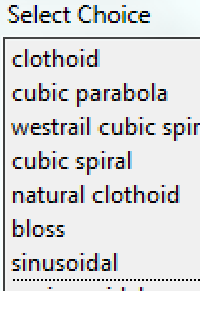
thickness of the string

Label style	choice box	default	available label styles

The super alignment style controls the way the super alignment draws and highlights on the screen.
For more information please go to the section 14.2.3 Super Alignment Style

Transition type	input	clothoid	clothoid, cubic parabola westrail-cubic, cubic spiral natural clothoid, bloss sinusoidal, cosinusoidal

transition (for example clothoid spiral) to be used for this super alignment string. For more information see 39.2.7.5 Transitions and Spirals File.



Chain file	file		

The chain parameter file to run every time the super alignment is resolved/recalced.

For more information see the section 28.3 Chains in the book 28 Utilities

Close	tick box		

*if **ticked**, the super alignment is automatically closed.*

Sync vertical geometry tick box

*if **ticked**, when the horizontal geometry is modified, an attempt is made to modify the chainage position of the vertical geometry so that remains locked to the same parts of the horizontal geometry as before the horizontal geometry was modified.*
*If **not ticked**, after horizontal geometry is modified, the vertical will possibly be moved in relation to the horizontal geometry because the (x,y) position for a given chainage will change.*

Use chainage equalities tick box

*if **ticked**, chainage equalities can be defined and the automatic labelling of the string will use the equalities.*

Parallel button

after the **Parallel** *button is clicked, a dereferenced copy of the horizontal geometry of the super alignment is paralleled. The vertical geometry is deleted.*

Same as button

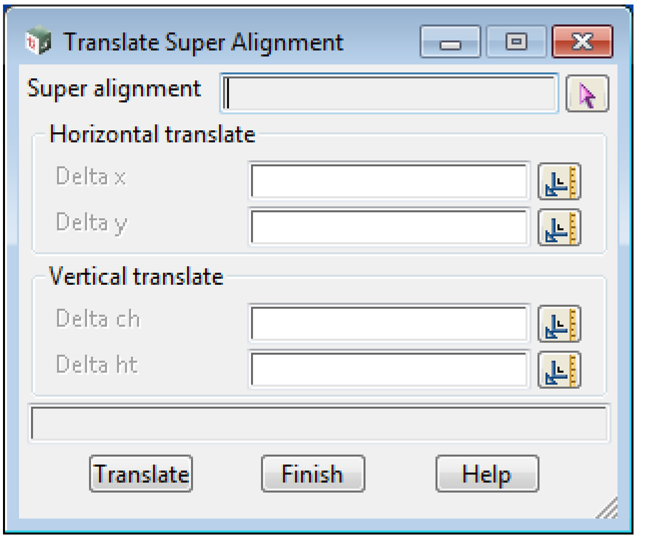
after clicking **Same as** *and then select a string. All the picked string properties will be loaded into the panel fields.*

14.2.2.7 Translate Super Alignment

Position of option on menu: **Strings =>Super alignments =>Tools =>Translate**

Translate copies the selected super alignment, and then dereferences and translates the copied string.

Selecting **Translate** displays the **Translate Super Alignment** panel.



The new fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Super alignment	string select		

select the super alignment to translate

Horizontal translate

Delta x, Delta y: the x/y amounts to translate the string

Vertical translate

Delta ch, Delta z: the chainage/z amounts to translate the string

Translate button

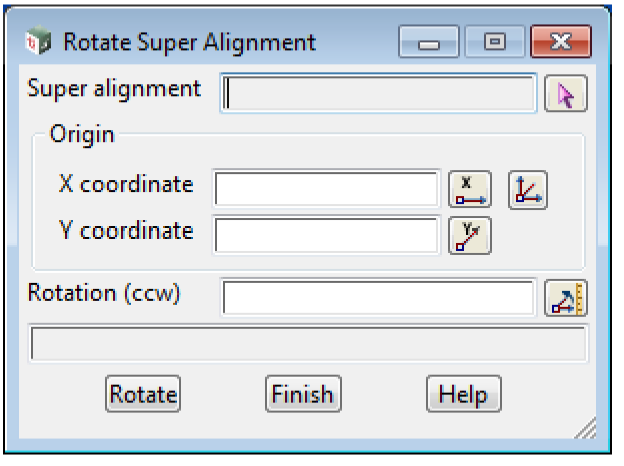
after the **Translate** *button is clicked, the selected string is copied, and the copied string is translated.*

14.2.2.8 Rotate Super Alignment

Position of option on menu: **Strings =>Super alignments =>Tools =>Rotate**

Rotate copies the selected super alignment, and then dereferences and rotates the copied string about a user defined origin.

Selecting **Rotate** displays the **Rotate Super Alignment** panel.



The new fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

Super alignment string select

select the super alignment to copy and rotate the copied string

Origin xyz ops

select the centre (origin) of the rotation. The origin can be changed at any time.

Rotation string select

the clockwise angle in dms to rotate through.

Rotate button

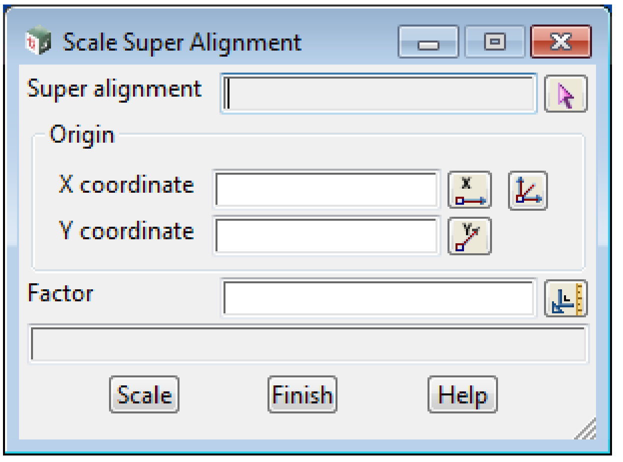
after the **Rotate** *button is clicked, the selected string is copied, and the copied string is rotated about the origin.*

14.2.2.9 Scale Super Alignment

Position of option on menu: **Strings =>Super alignments =>Tools =>Scale**

Scale copies the selected super alignment, and then dereferences and scales the copied string about a user defined origin.

Selecting **Scale** displays the **Scale Super Alignment** panel.



The new fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Super alignment	string select		

select the super alignment to copy and scale

Origin xyz ops

select the origin to scale about. The origin can be changed at any time.

Factor

amount to scale the copy of the super alignment by.

Scale button

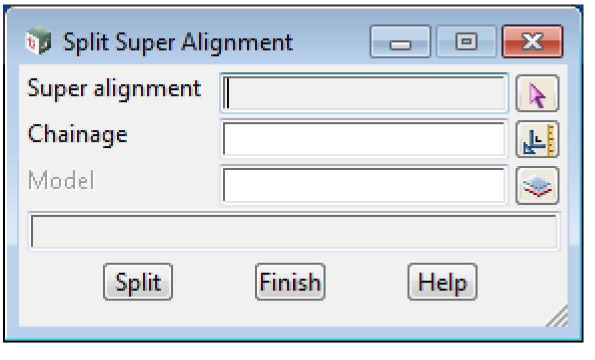
*after the **Scale** button is clicked, the selected string is copied, and the copied string then scaled about the origin.*

14.2.2.10 Split a Super Alignment

Position of option on menu: **Strings =>Super alignments =>Tools =>Split**

Split copies and splits the copy of the super alignment at a user given chainage.

Selecting **Split** displays the **Split Super Alignment** panel.



The new fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Super alignment	string select		

select a super alignment to split

Chainage input

chainage to split the super alignment at

Model model box available models

if not blank, model for the split string
If blank, use the model of the selected string

Split button

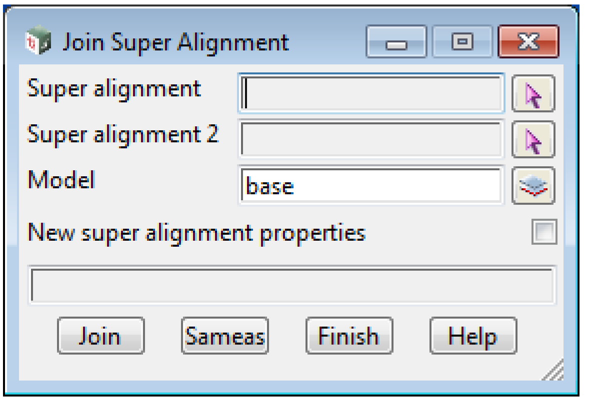
after the **Split** *button is clicked, the super alignment is copied and split at the given chainage.*

14.2.2.11 Joining Two Super Alignments

Position of option on menu: **Strings =>Super alignments =>Tools =>Join**

Join joins two super alignments at their ends to make a new super alignment.

Selecting **Join** displays the **Join Super Alignment** panel.



The new fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

Super alignment string select

select a super alignment to join

Super alignment 2 string select

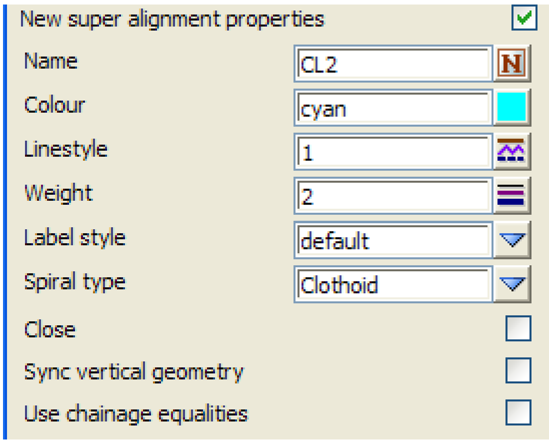
select a super alignment to join to the first selected super alignment

Model model box available models

if not blank, model for the joined string
If blank, use the model of the selected string for the joined string.

New super alignment properties tick box

if ***ticked****, the panel is expanded to show all the properties that can be set for the new joined string.*



Extra panel fields when
New super alignment properties
is ticked on

Join button

*after the **Join** button is clicked, a new super alignments is created by joining the two super alignments*

Same as button

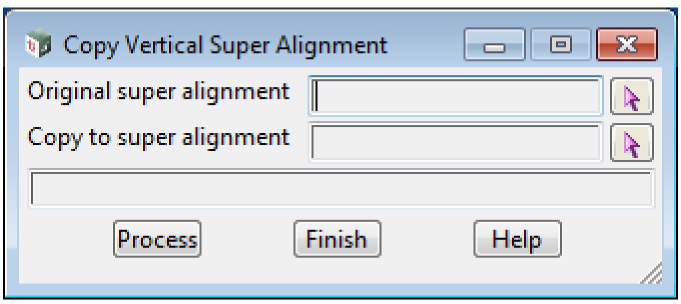
*after clicking **Same as** and then select a string. All the picked string properties will be loaded into the panel fields.*

14.2.2.12 Copy Vertical Geometry

Position of option on menu: **Strings =>Super alignments =>Tools =>Copy vertical**

Copies the vertical geometry from one super alignment to another super alignment. This will not work for a super alignment with computators.

Selecting **Copy vertical** displays the **Copy Vertical Super Alignment** panel.



The new fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

Original super alignment string select

select the super alignment to copy vertical geometry from

Copy to super alignment string select

select the super alignment to copy vertical geometry to

Process button

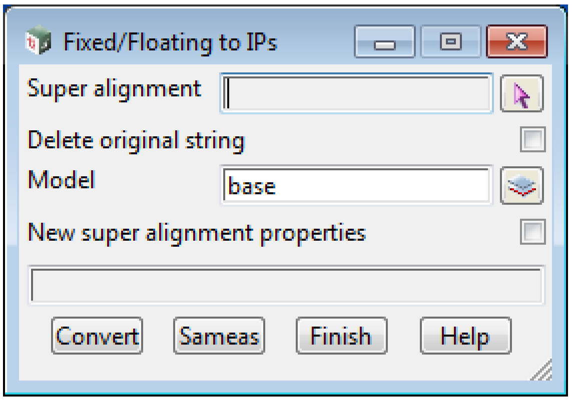
*after the **Process** button is clicked, the vertical geometry is copied from the original alignment to the second super alignment.*

14.2.2.13 Convert Fixed/Floating Elements to IPs

Position of option on menu: **Strings =>Super alignments =>Tools =>Fixed/floating to IPs**

Fixed/floating to IPs converts the fixed and floating elements of the horizontal and vertical geometry to horizontal and vertical intersection points joins with arcs and transitions for HIPs and parabolas or vertical arcs for VIPs.

Selecting **Fixed/floating to IPs** displays the **Fixed/floating to IPs** panel.



The new fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

Super alignment string select

select a super alignment to convert

Tolerance real box

when the string is converted, the new IP string may be slightly different to the original string. The underlying vertices of the original and new strings are compared and if they differ by the given tolerance value, then a warning is given.
If blank, the tolerance is 0.001 (1 mm)

Delete original string tick box

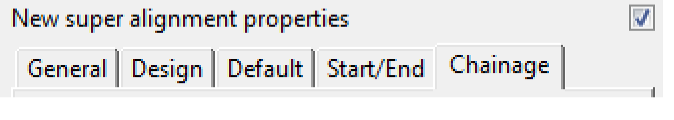
if ticked, the original super alignment is deleted after the conversion is successful

Model model box available models

if not blank, model for the converted string.
If blank, use the model of the selected string for the converted string.

New super alignment properties tick box

*if **ticked**, the panel is expanded to show all the properties that can be set for the new converted string.*



Extra panel fields when New super alignment properties is ticked.

The tabs are the same as when Creating a Super Alignment - go to On selecting the Super Alignment option, the Create Super Alignment panel is displayed.
*Please note that under **New Super Alignment properties => General Tab*** **Many Strings** *is not applicable to* **Fixed/Floating to IPs.**

Convert button

after the **Convert** *button is clicked, a new super alignments is created by converting all the elements to IPs*

Same as button

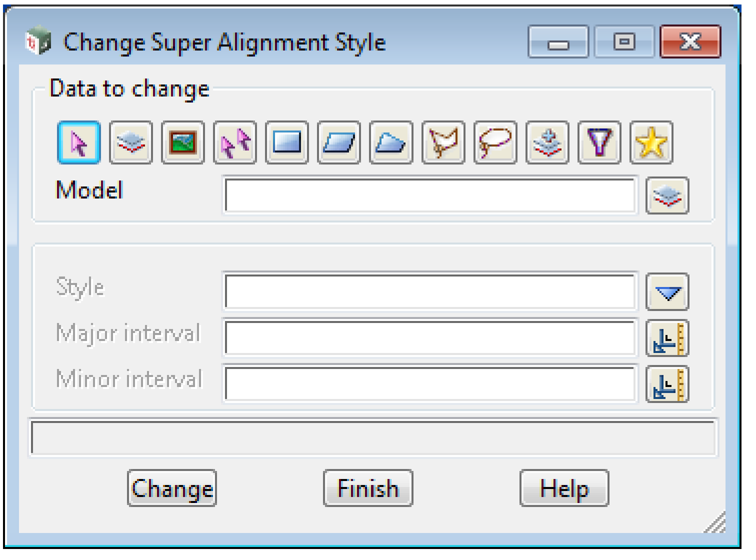
after clicking **Same as** *and then select a string. All the picked string properties will be loaded* *into the panel fields.*

14.2.2.14 Change the Styles for Selected Super Alignments

Position of option on menu: **Strings =>Super alignments =>Tools =>Change style**

Change style changes the super alignment style for all the selected super alignments.

Selecting **Change style** displays the **Change Super Alignment Style** panel.



The new fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Style	SA style box		available SA styles

super alignment style to apply to all the selected super alignments

Change	button		

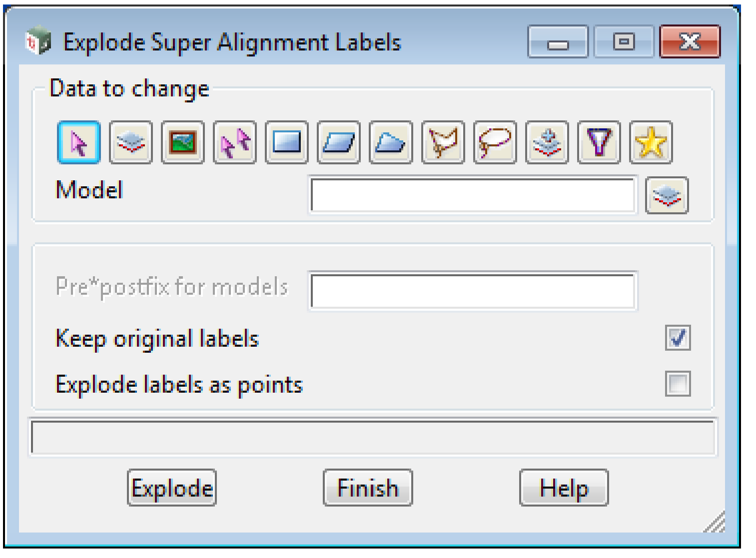
*after the **Change** button is clicked, all the selected super alignments are given the super alignment style*

14.2.2.15 Explode the Super Alignments Labelling

Position of option on menu: **Strings =>Super alignments =>Tools =>Explode labels**

Explode labels creates text and lines from the automatic labelling of the selected super alignments.

Selecting **Explode labels** displays the **Explode Super Alignment Labels** panel.



The new fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

Data source type Model

data selection type - for a full description go to 4.19.3 Data Source.

Data source input

source of data to be processed.

pre*postfix for models text

the models for the text are the model name containing the super alignment, with prefixing and postfixing as given by this field

Keep original labels tick box

if not ticked, the label style of the original strings are modified to ***no labels***

Explode labels as points tick box

if ticked, the labels are created as single point super strings with the label as vertex text.

If not ticked, the labels are created in the one super string with the text as vertex text.

Explode button

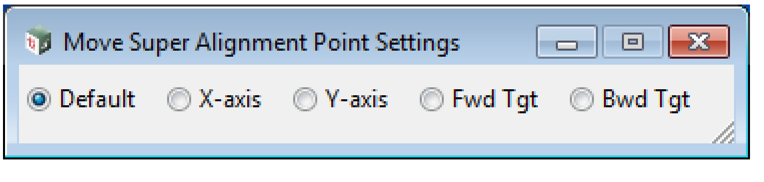
create text strings for all the text in the automatic labels of the selected super alignments

14.2.2.16 Constrain the Movement of Super Alignment Points

Position of option on menu: **Strings =>Super alignments =>Tools =>Move settings**

Move settings constrains the movement of horizontal and vertical intersection points for any super alignment being edited.

Selecting **Move settings** displays the **Move Super Alignment Point Settings** panel.



The buttons are radio buttons - that is, only one can be selected:

Default

allows any movement in the selected HIP or VIP point

X-axis

constrains an HIPs to only move along the x-axis. That is, the y coordinate is fixed.

Constrains a VIPs to only move along the chainage-axis. That is, the height coordinate is fixed.

Y-axis

constrains an HIPs to only move along the y-axis. That is, the x coordinate is fixed.

Constrains a VIPs to only move along the height-axis. That is, the chainage coordinate is fixed.

Fwd Tgt

constrains an HIPs to only move along the incoming straight to the HIP.

Constrains a VIPs to only move along the incoming straight to the VIP.

Bwd Tgt

constrains an HIPs to only move along the outgoing straight from the HIP.

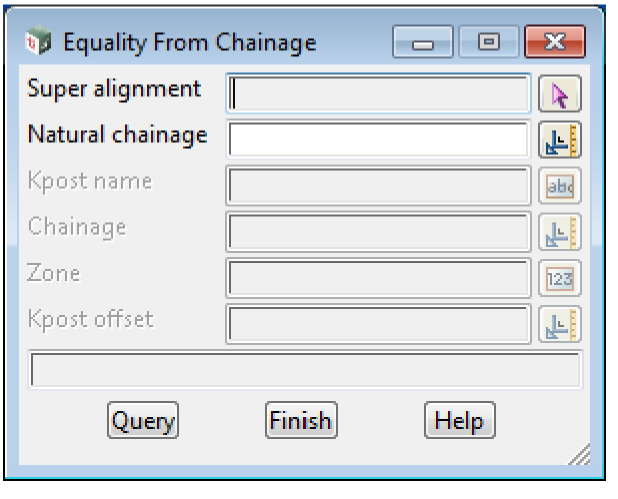
Constrains a VIPs to only move along the outgoing straight from the VIP.

14.2.2.17 Equality Query

Position of option on menu: **Strings =>Super alignments =>Tools =>Equality query**

Displays the Kpost name, offset from Kpost and Zone of the Kpost for a selected position on a super alignment.

Selecting **Equality query** displays the **Equality From Chainage** panel.



The new fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Super alignment	string select		

select the super alignment to display the Kpost name, Offset and Zone for

Chainage

the raw chainage to calculate the Kpost name, Offset and Zone at. When the super alignment is first selected, the raw chainage of the selected position is written to the Chainage *box.*

Query button

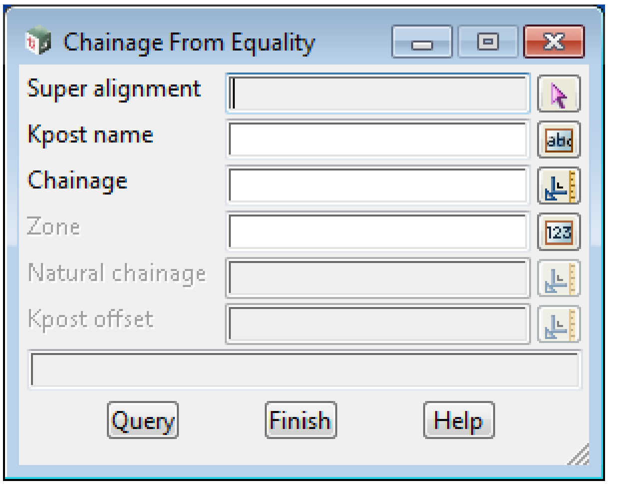
*after the **Query** button is clicked, the vertical geometry is copies from the original alignment to the second super alignment.*

14.2.2.18 Chainage Query

Position of option on menu: **Strings =>Super alignments =>Tools =>Chainage query**

For a user given Kpost name, offset from Kpost and Zone, this option displays the raw chainage of that point.

Selecting **Chainage query** displays the **Chainage From Equality** panel.



The new fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

Super alignment string select

select the super alignment to get the raw chainage for

Kpost name

name of a Kpost on the selected string

Offset

offset from the Kpost

Zone

Kpost zone

Query button

*after the **Query** button is clicked, the raw chainage of the point given by the Kpost name, Offset from Kpost and Zone.*

14.2.2.19 Reverse Calc Parts

Position of option on menu: **Strings =>Super alignments =>Tools =>Reverse calc parts**

A super alignment is made up of the methods for constructing each part of the super alignment (one set for the horizontal geometry and another set for the vertical geometry), and if the methods are consistent, then they define a string of tangential segments which defines the super alignment in plan and in section.

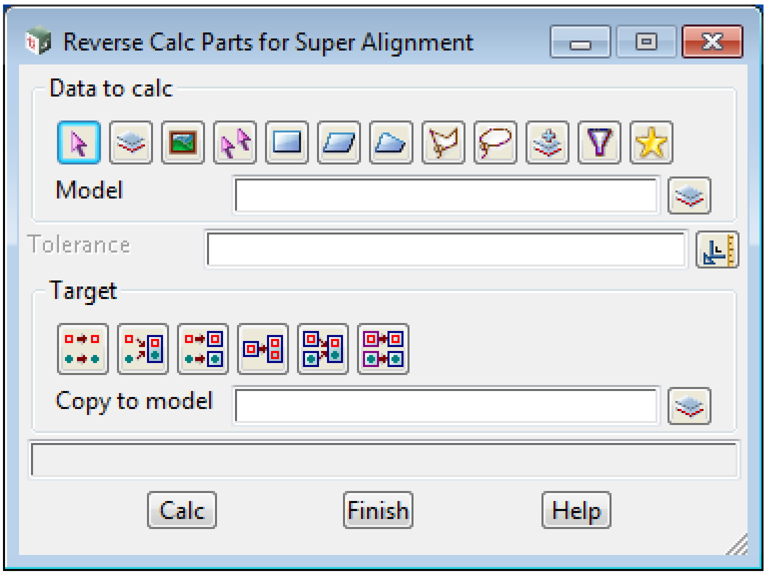
In plan the segments can be lines, arcs or transitions. In section the segments are lines, arcs or parabolas. Normally the segments are tangential to each other.

Sometimes a super alignment is missing the construction parts and only consists of the underlying segments. Then the super alignment can not be edited since there are no methods defining the string.

The *Reverse Calc Parts* option takes a super alignment consisting only of segments, and tries to build construction methods for the super alignment. If it is successful, then the super alignment can then be edited.

Note - the segments should be tangential to each other but if that is not the case, the option will try and move the vertices by a user defined tolerance to find a tangential solution.

Selecting **Reverse calc parts** displays the **Reverse Calc Parts For Super Alignment** panel.



The new fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source input

source of data to be processed.

Tolerance Measure box

the segments need to be tangential to each other for the reverse calculation of parts to work. If segments are not tangential, the vertices will be moved by up to the given Tolerance to make them tangential. There is a warning if the resultant super alignment can't solve.

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target.

Target info input

extra information required for the target.

Calc button

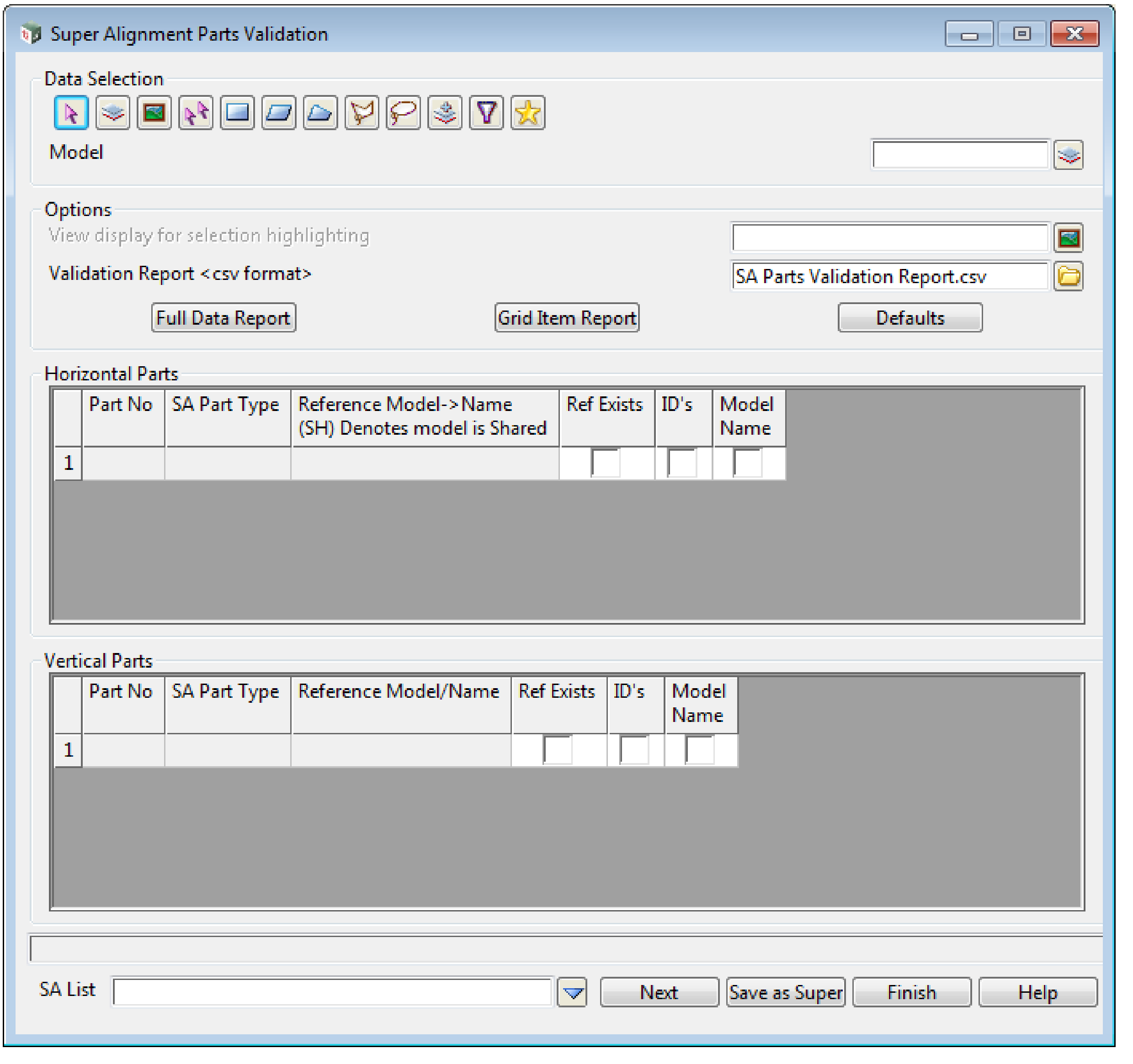
*after the **Calc** button is clicked, the option tries to create calculation parts for any super alignments with only segments and no calculation parts.*

14.2.2.20 Validate Super Alignment

Position of option on menu: **Strings =>Super alignments =>Tools =>Validate**

This section of documentation is a work in progress and will be updated in subsequent releases.

Selecting **Validate** displays the **Super Alignment Parts Validation** panel.

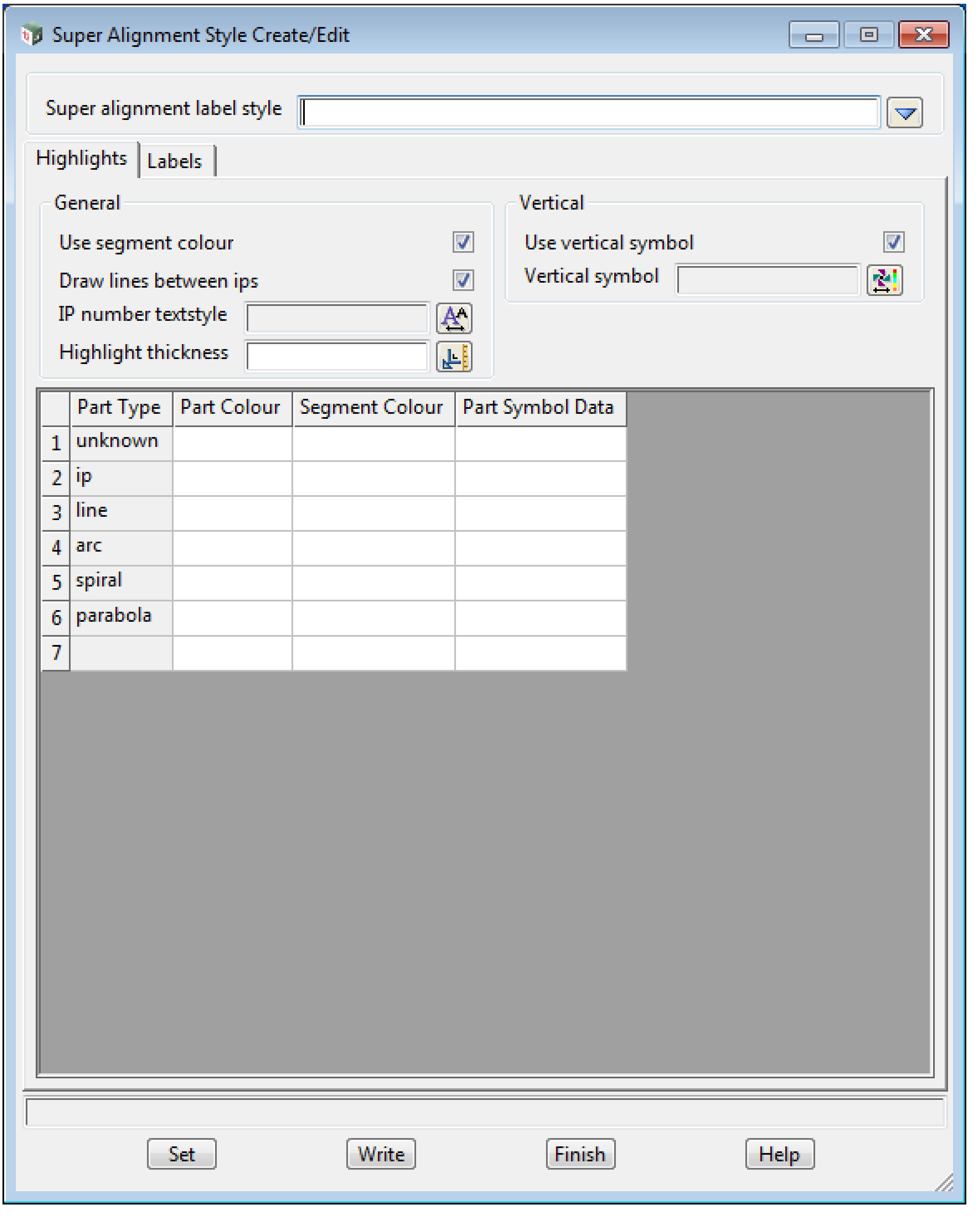


14.2.3 Super Alignment Style

Position of option on menu: **Strings =>Super alignments =>Label styles**

The super alignment label styles control the way that the super alignment draws on the screen and when it is being edited.

Selecting **Super alignment style** displays the **Super Alignment Style Create/Edit** panel.



For the tab ***Highlights*** *go to* Highlights tab
For the tab ***Labels*** go to Highlights Table

For the ***Buttons at the bottom*** go to Buttons at Bottom

The fields and buttons used in the **Super Alignment Style Create/Edit** panel have the following functions.

Field Description Type Defaults Pop-Up

Super alignment label style choice box design styles defined in astyles.4d

*all the information in the panel refers to the selected labelling style. Note - the labelling styles are stored in the file **astyles.4d**.*

Highlights tab

the definition of how to show the horizontal and vertical parts of the super alignment and also how to colour and label the segments when in and

Use segment colour tick box

*if **ticked**, each of the segment types are drawn in the colour given in the Highlights table given below (when the super alignment is on any view).*

Draw lines between IPs tick box

*if **ticked**, lines are drawn between intersection points (IPs).*

IP number textstyle textstyle box

the textstyle used to draw IP numbers in Plan and Section views.

Highlight thickness input

line thickness in pixels

Use vertical symbols tick box

*if **ticked**, vertical symbols are drawn when the super alignment is profiled on a Section view.*

Vertical symbol symbol box

the symbol to use for common points between segments in the Section views.

Highlights Table

the definition of the segments that make up the definition of the part (part colour and part symbol for the fixed points) and the colour of the solved segments as a result of the part definition.
Defined for ip, line, arc, spiral (transition), parabola and unknowns.

Part type

the part type - unknown, ip, line, arc, spiral (transition), parabola

Part colour colour box

the definition of the part is only shown when in the edit mode.
When the string is being edited, the segments of the part are drawn in these colours.
For ip, this is the colour of the cross at the ip.
For line/arc/spiral/parabola, this is the colour of any line/arc/spiral/parabola segment.

Segment colour colour box

the resulting segments defined by the part are drawn in these colours.
for ip, this is the colour of the cross at the ip
line/arc/spiral/parabola, this is the colour of any line/arc/spiral/parabola segment

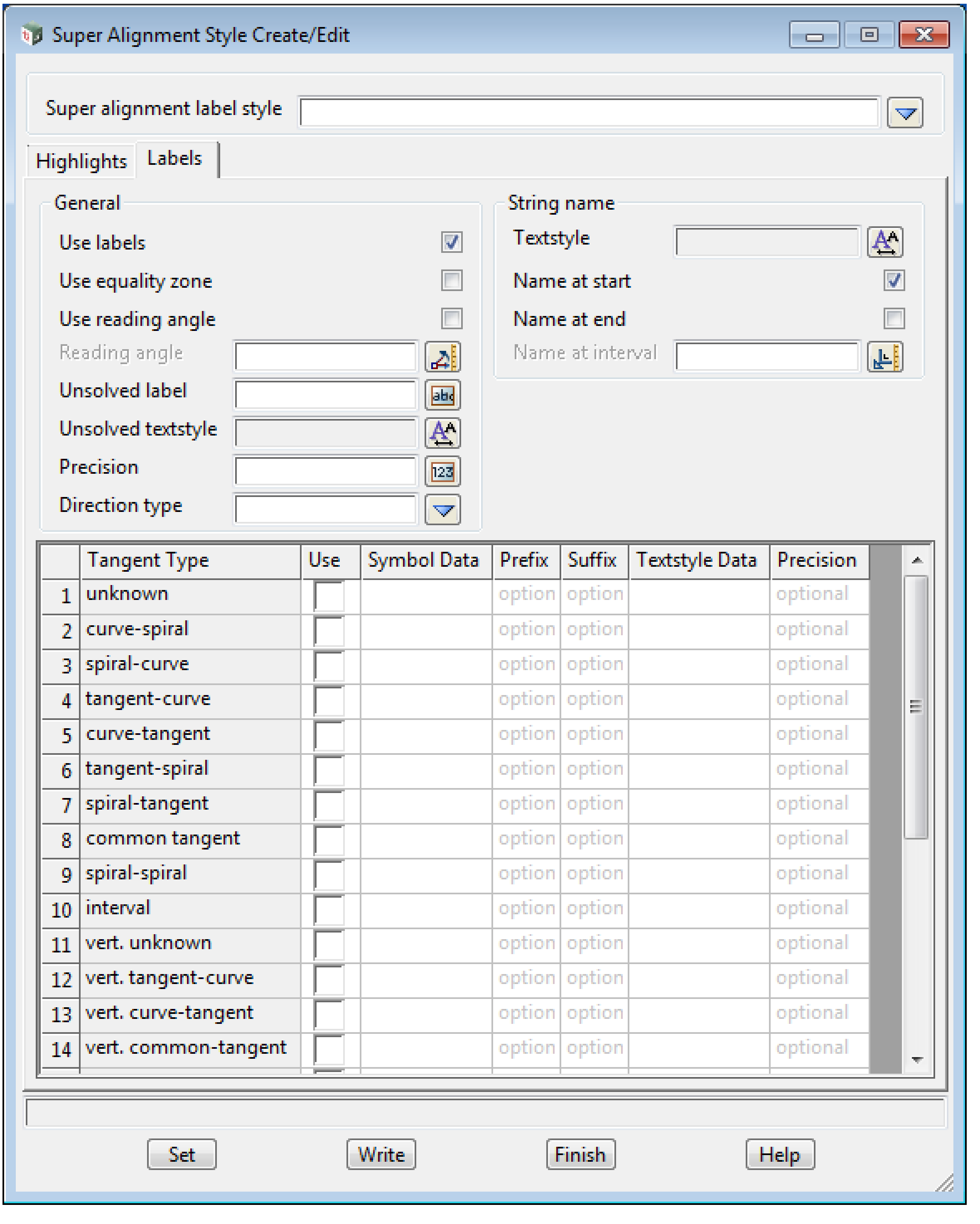
Part symbol data symbol box

when a string is being edited, the fixed points of the parts are displayed with these symbols.
for ip, the symbol at the ip of a part.

For line/arc/spiral/parabola, this is the symbols of the fixed points of any line/arc/spiral/ parabola in the part. For example, if an arc is defined by going through three points, the three points will be drawn with the symbol for Part type arc.

Labels tab

on a plan view, the definition of what text, symbols and decimal places (precision) should be used in labelling tangent points, crest and sag points, major and minor intervals, segment radii and lengths, and special chainages.



The fields and buttons used in the **Super Alignment Style Create/Edit** panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Use labels tick box

*if **ticked**, labels are automatically drawn when the super alignment is on a Plan view.*

Use equality zone tick box

*if **ticked** and chainage equalities exist in the super alignment, the chainage intervals are relative to the beginning of the zone it is in.*

Use reading angle tick box

*if **ticked** and the angle of the label relative to the Reading angle is over 180 degrees, the label will be flipped.*

Reading angle input

the angle to take as a base line for reading text. The angle is in 4.17.1 HP Notation *and is measured from the x axis in a counter clockwise direction.*

Unsolved label text box

the text label to use when a part is not solved. It is placed in the middle of the unsolved segment.

Unsolved textstyle textstyle data box

the textstyle of the unsolved text label

Precision input

the number of decimal places to use in the chainage labels

Direction type choice box none, at start, each segment

how to draw the string direction symbol which is given in row 18 of the table below (horz. direction)
*If **none**, no direction symbol is drawn.*
*If **at start**, a single direction symbol is drawn at the start of the string*
*If **each segment**, a direction symbol is drawn at the middle of each segment of the string*

String name textstyle textstyle data box

the textstyle to use for drawing the name of the string

Name at start tick box

*if **tick**, draw the name of the string at the start of the string.*
*If **not ticked**, no string name is drawn at the start of the string.*

Name at end tick box

*if **tick**, draw the name of the string at the end of the string.*
*If **not ticked**, no string name is drawn at the end of the string.*

Name at interval double box

*if **not blank**, the name of the string is drawn at this chainage interval along the string, excluding the start and the end of the string.*

Label Type Table

*For each label type in the table, the Use flag is ticked if it is to be drawn, and if it is drawn, the symbol is given in the Symbol Data column, text is prefixed/postfixed as given in the Pre*Postfix column, the text used the textstyle data given in the Textstyle Data column with the number of decimal places given in the Precision column.*
There are twenty four (24) types:

unknown - when it is not any of the other horizontal critical points
curve-spiral - the tangent point between an arc and a transition. Eg CS
spiral-curve - the tangent point between a transition and an arc. Eg SC

tangent-curve - the tangent point between a straight line and an arc. Eg TC
curve-tangent - the tangent point between an arc and a straight line. Eg CT
tangent-spiral - the tangent point between a straight line and a transition. Eg TS
spiral-tangent - the tangent point between a transition and a straight line. Eg ST
common tangent - the tangent point between two arcs. Eg CTP
spiral-spiral - the tangent point between two transitions Eg SS
interval - the chainage label at the major interval
vert. unknown - when it is not any of the other vertical critical points
vert. tangent-curve - the tangent point between a straight grade and a vertical curve (parabola or an arc). Eg VTC
vert. curve- tangent- - the tangent point between a vertical curve (parabola or an arc) and a straight grade. Eg VCT
vert. common tangent - the tangent point between two vertical curves. Eg VCTP
crest - a vertical crest (local maximum) point. Eg CREST
sag - a vertical sag (local minimum) point. Eg SAG
special chainages - chainages from a special chainage file given in the Special chainage file field in the Basic Label section of the Super Alignment Properties
horz. direction - the direction symbol of the string
minor interval - the chainage label at the minor interval
k post equality - label the k posts equations. Eg 15 Km 998 = 16 Km
internal equality - label the internal equality equation Eg 17 Km 500 = 17 Km 600
segment radius - label the radius of an arc segment Eg R 500.0
segment length - label the length of a segment Eg L 500.0
segment bearing - label the bearing of a segment in dms format Eg 125 30 20

Buttons at Bottom

Set button

clicking the **Set** *button means that for this session, all the definitions in the panel will be used for the style given in the* ***Super alignment label style*** *field. Unless the* **Write** *button is used to write the values out to the* ***astyles.4d*** *file, the values will be lost when the project is exited.*

Write button

clicking the **Write** *button writes out all the values in this panel to the* ***astyles.4d*** *file as the style given in the* ***Super alignment label style*** *field*

Note

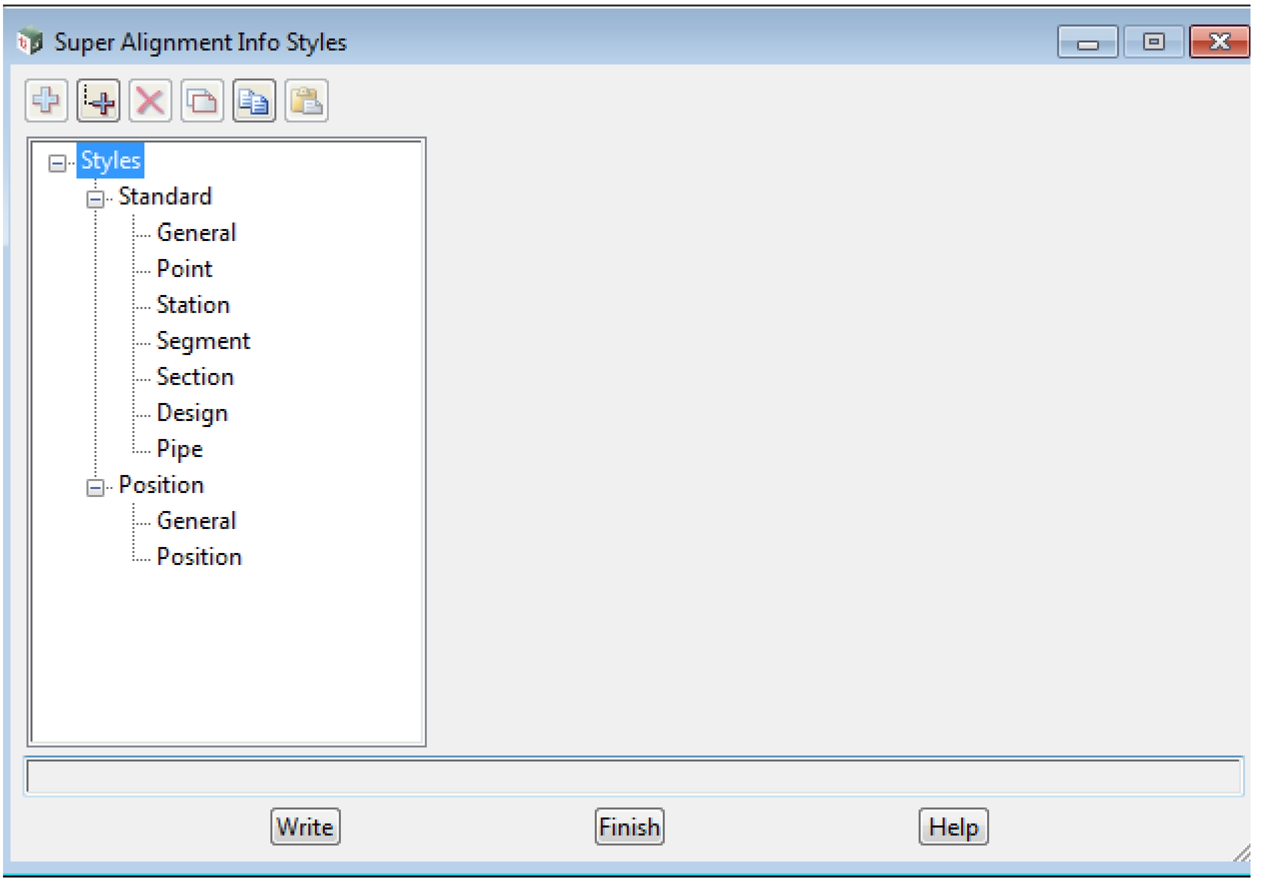
The new Super alignment label style is not used until a resolve is done on the super alignment string.

14.2.4 Super Alignment Info Styles

Position of option on menu: **Strings =>Super alignments =>Info style**

The *super alignment info styles* control what information is displayed in the **Super Alignment Info** panel brought up by the 14.2.2.1 Super Alignment Info option on the **SA Tools** menu or toolbar.

Selecting **Super alignment info** displays the **Super Alignment Info Styles** panel.



14.2.5 Design Standards

Position of option on menu: **Strings =>Super alignments =>Design standards**

The ***design standards*** sets up numerous standard tables for

(a) horizontal speed (speed_standards)

defines for various speeds, the use of horizontal arcs and transition curves, super elevation for various numbers of lanes.

(b) horizontal widening (widening_standards)

defines the road widening for various lane widths, speeds and curve radii.

(c) horizontal small deflection (small_deflection_standards)

(d) vertical crest (crest_standards)

defines for crest curves the k-values of allowed parabolas for different speeds

(e) vertical sag (sag_standards)

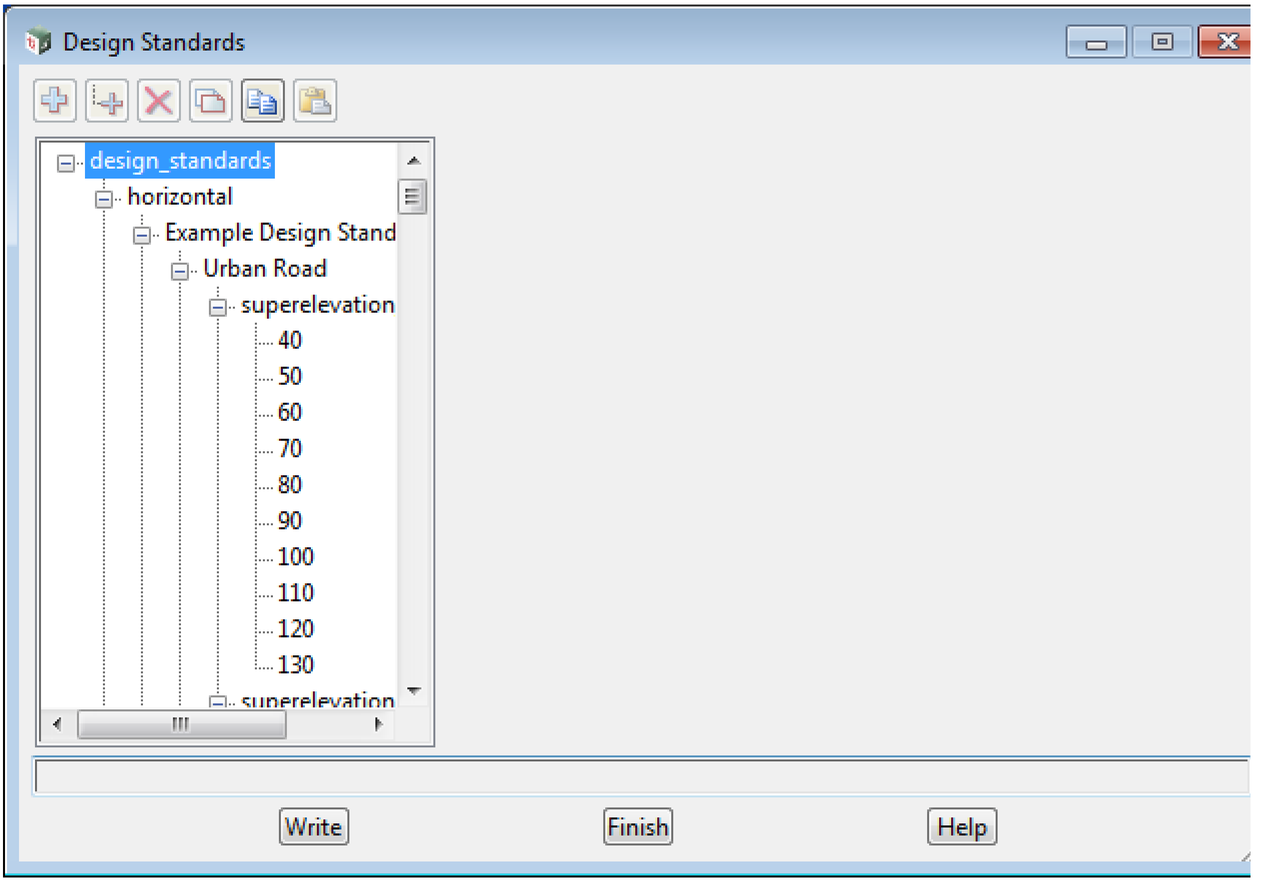
defines for sag curves the k-values of allowed parabolas for different speeds

For each of the five types of tables, specific tables of values are set up by the user for each of the different design scenarios that the user will undertaken. Each specific table is given a name, and the names must be unique within each of the five types of tables.

The tables are written to a file (design_standards.xml) which is then read in whenever a project is opened.

The tables in the design standards file are used in the option **Strings =>Create =>Super alignments =>Design templates** to define ***Design Templates*** (see 14.2.6 Design Templates), and when a super alignment is created, ***one*** *Design Template* is selected for use with that super alignment (see the section Advanced > Design branch from 14.2.1 Create Super Alignment).

Selecting **Design standards** displays the **Design Standards** panel.



DOCO TO GO INTO DEFINITIONS OF DESIGN PARAMETERS

Ease-off choice box none
maximum
by length

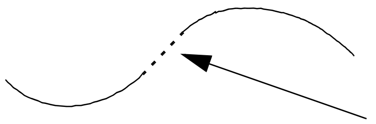
method of transitioning between minimum and maximum super elevation.
*If **none**, no transition is done.*
*If **maximum**, maximum back to back curves are added to the super diagram between the minimum and maximum super elevation and are used to transition the super.*
*If **by length**, the ease off distance is given by the **Ease off length** value.*

Ease-off length input

*the length of the ease off for the transition of super elevation when the **by length** method is selected*

Min on reverse input

*if the chainage distance between two curves of opposite sign is less than the **Minimum on reverse** value then the two curves are treated as reverse curves when applying the design rules.*

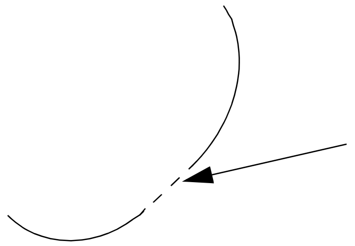


If this distance is less than
Min on reverse then the two curves
are treated as reverse curves

Plan View

Min on broken back input

*if the chainage distance between two curves of the same sign is less than the **Minimum on broken-back** value then the two curves are treated as broken-back curves when applying the design rules.*



If this distance is less than
Min on broken-back then the two curves
are treated as broken back curves

Plan View

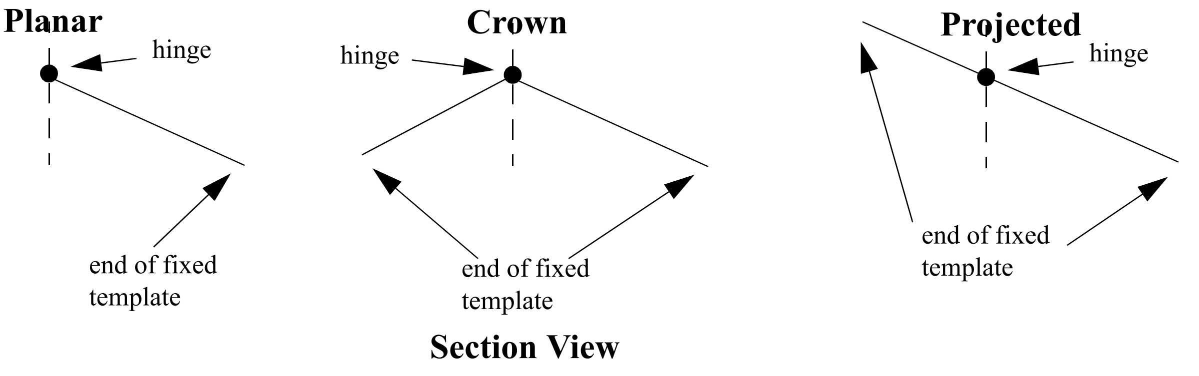
FROM V9 DOCO

Slope type choice box planar
crowned
projected

*If **planar**, then the design table is only applied for half of the road (from the hinge string to the end of the fixed template). The side used is given by **Highside**.*

*If **crowned**, then the design table is used for each side of the roads (from the left most fixed template to the hinge string, and from the hinge string to the right most fixed template).*

*If **projected**, then the design table is first applied to half of the road given by **Highside** (from the hinge string to the end of the fixed template). Then the grade from the **Highside** is projected through the hinge string and used on the other side of the road.*



Highside	choice box	left edge	left edge, right edge

*used if the **Slope type** is planar or projected to define the higher side of the road pavement.*

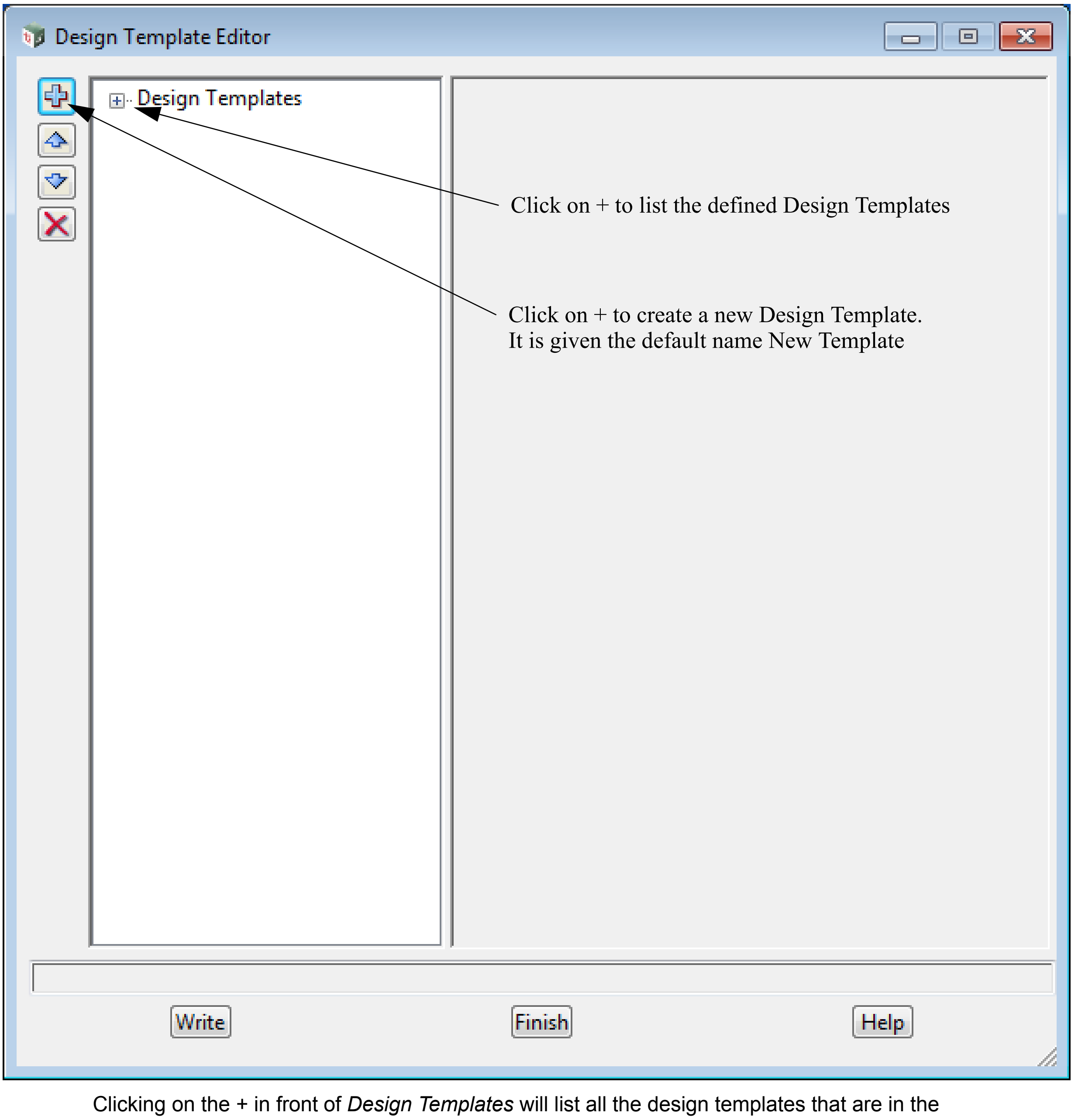
14.2.6 Design Templates

Position of option on menu: **Strings =>Super alignments =>Design templates**

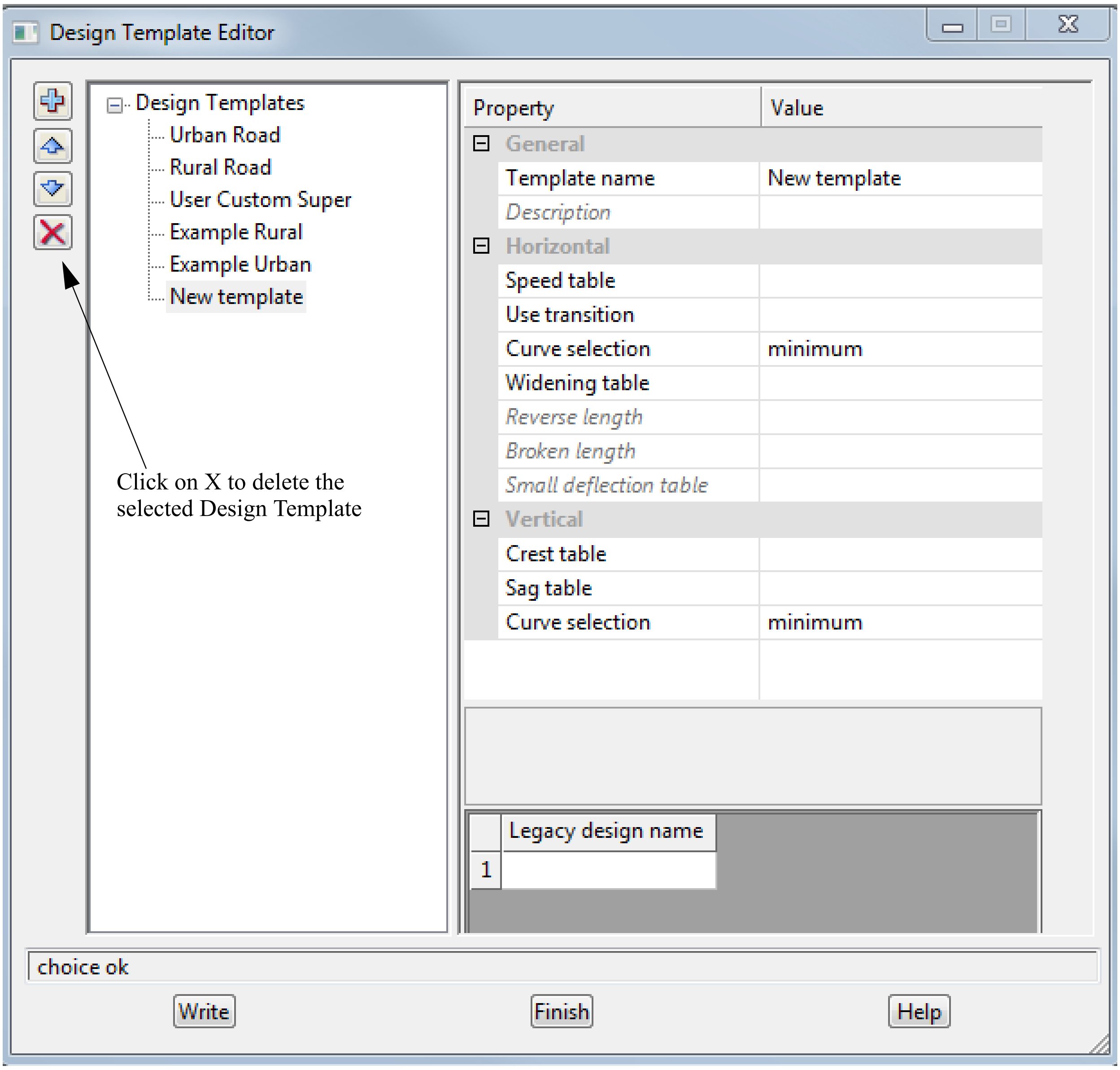
The ***Design Templates*** option is used to create Design Templates (each with a unique name) that have one of each of the five design table types (horizontal speed, horizontal widening, horizontal small deflection, vertical crest, vertical sag) that have been defined by the **Strings =>Create =>Super alignments =>Design standards** option (see 14.2.5 Design Standards), plus whether curve transitions are allowed, whether minimum or maximum curve radii are used, and reverse and broken back curve lengths.

When a super alignment is created, ***one*** *Design Template* is selected for use with that super alignment (see the section Advanced > Design branch from 14.2.1 Create Super Alignment).

Selecting **Design templates** displays the **Design Template Editor** panel.



Clicking on the + in front of *Design Templates* will list all the design templates that are in the *design_templates.xml* file currently be used in the *12d Model* project.



The fields and buttons used in the **Design Template Editor** panel when creating a new *Design Template*, or when clicking on an existing *Design Template*, have the following functions.

Field Description	Type	Defaults	Pop-Up
Template name	text box		

the name of the Design Template. This can not be blank and must be unique amongst the list of Design Templates. The default name is New Template and this can be changed by typing a new name into this field.

Description	text box		

a description for this Design Template. This can be blank.

Speed table	choice box		list of speed_standards

the name of the Speed Table to use. This is selected from the all the speed_standards defined in the Design Standards file (design_standards.xml) being used for the project.

User transitions	yes/no box		

*if **yes**, transition curves are allowed in the horizontal geometry.*
*If **no**, transition curves are not allowed in the horizontal geometry.*

Curve selection choice box minimum minimum. maximum

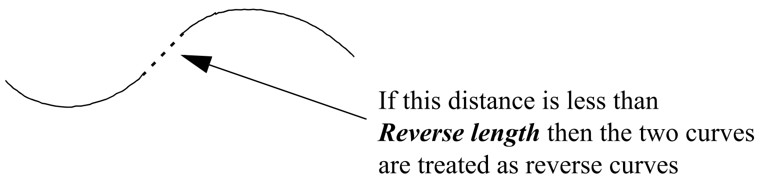
*if **minimum** and more than one horizontal curve radius exists for the speed, the minimum curve radius is used.*
*If **maximum** and more than one horizontal curve radius exists for the speed, the maximum curve radius is used.*

Widening table choice box list of widening_standards

the name of the Widening Table to use. This is selected from the all the widening_standards defined in the Design Standards file (design_standards.xml) being used for the project.

Reverse length input

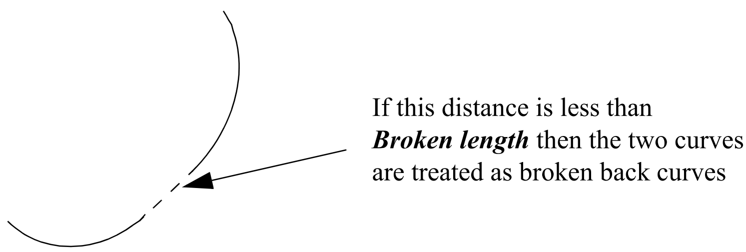
*if the chainage distance between two curves of opposite sign is less than the **Reverse length** value then the two curves are treated as reverse curves when applying the design rules.*



Plan View

Broken length input

*if the chainage distance between two curves of the same sign is less than the **Broken length** value then the two curves are treated as broken-back curves when applying the design rules.*



Plan View

Small deflection table choice box list of small_deflection_standards

the name of the Small Deflection Table to use. This is selected from the all the small_deflection_standards defined in the Design Standards file (design_standards.xml) being used for the project.

Crest table choice box list of crest_standards

the name of the Crest Table to use. This is selected from the all the crest_standards defined in the Design Standards file (design_standards.xml) being used for the project.

Sag table choice box list of sag_standards

the name of the Sag Table to use. This is selected from the all the sag_standards defined in the Design Standards file (design_standards.xml) being used for the project.

Curve selection choice box minimum minimum. maximum

*if **minimum** and more than one vertical curve k-value exists for the speed, the minimum curve k-value is used.*
*If **maximum** and more than one vertical curve k-value exists for the speed, the maximum curve k-value is used.*

Buttons at Bottom

Write button

clicking the **Write** *button writes out all the Design Templates defined in this panel to the* ***design_templates.xml*** *file.*

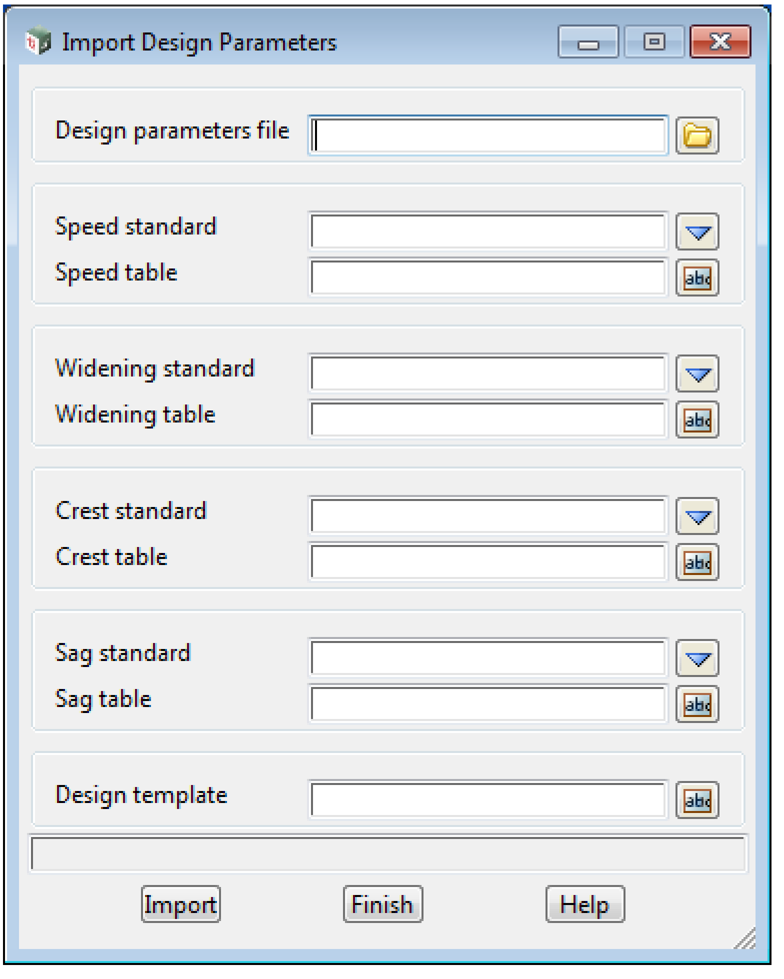
14.2.7 Import Legacy Design

Position of option on menu: **Strings =>Super alignments =>Import legacy design**

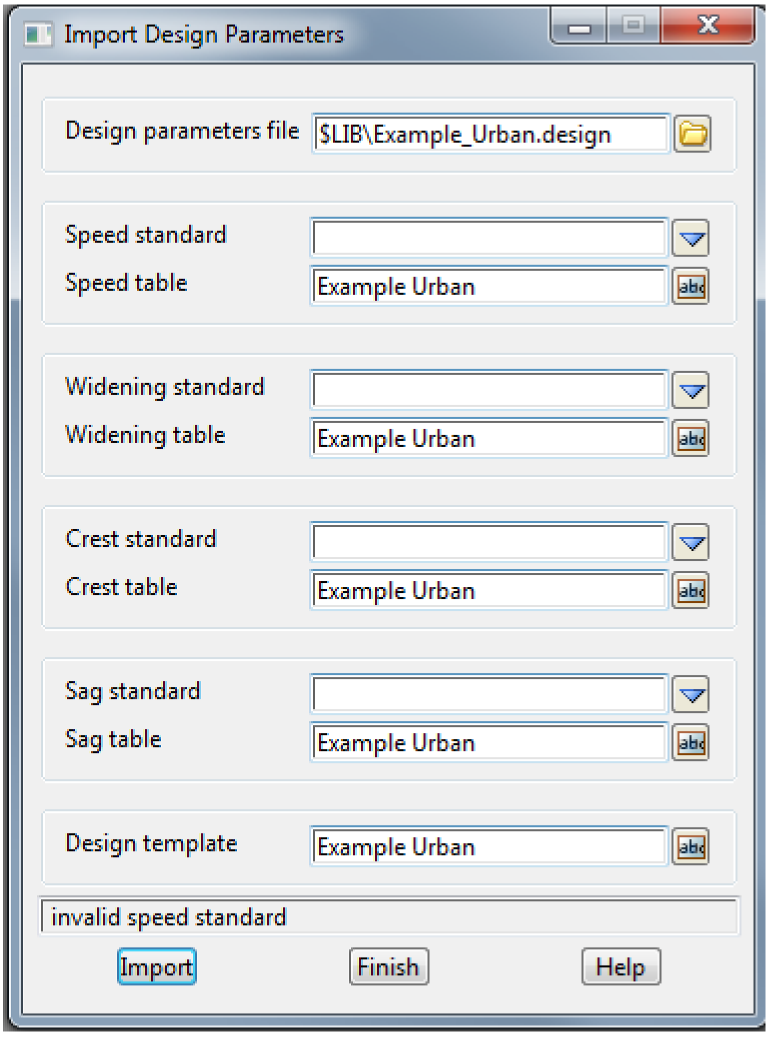
The ***Import legacy design*** option reads a ***12d Model 9*** *Design Parameter File* and converts it to a ***Design Template*** for use in ***12d Model 10*** or higher.

Note:

Selecting **Import legacy design** displays the **Import Design Parameters** panel.



Selecting a *Design parameters file* will read in the selected *.design* file and populate fields in the panel.


Import Design Parameters
Design parameters file
$LIB\Example_Urban.design
Speed standard
Speed table
Example Urban
Widening standard
Widening table
Example Urban
Crest standard
Crest table
Example Urban
Sag standard
Sag table
Example Urban
Design template
Example Urban
invalid speed standard
Import
Finish
Help

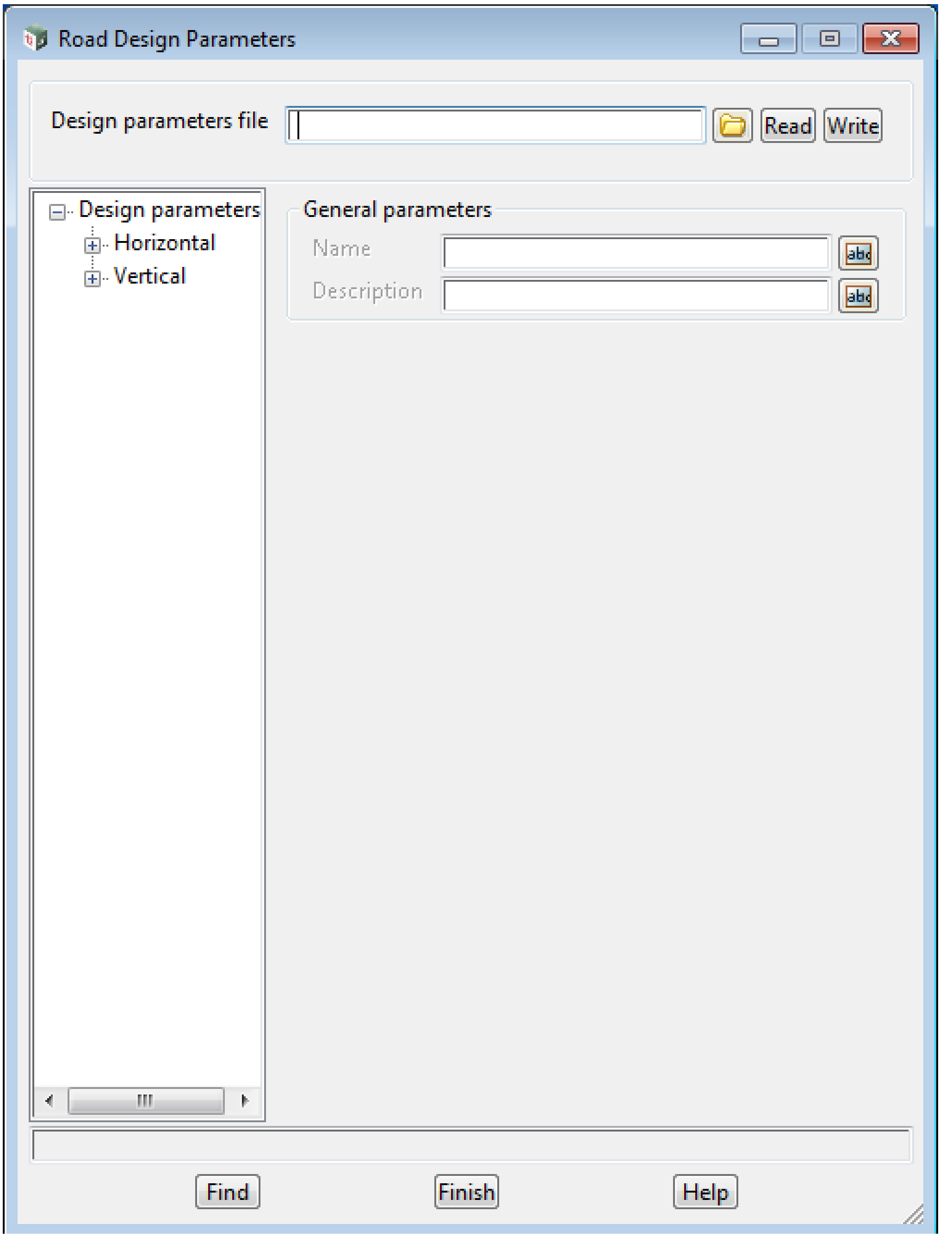
14.2.8 Define Design Parameters - from V9

Position of option on menu: **Strings =>Create =>Super alignments =>Design parameters**

This was superseded in 12d Model 10 by Design Tables and Design Templates.

The ***design parameters table*** defines the use arcs, transition curves, parabolas, super elevation and road widening for given design speeds. The table is used to use by a super alignment in conjunction with user supplied information given in the ***Design*** tab of the super alignment.

Selecting **Design parameters** displays the **Road Design Parameters** panel.

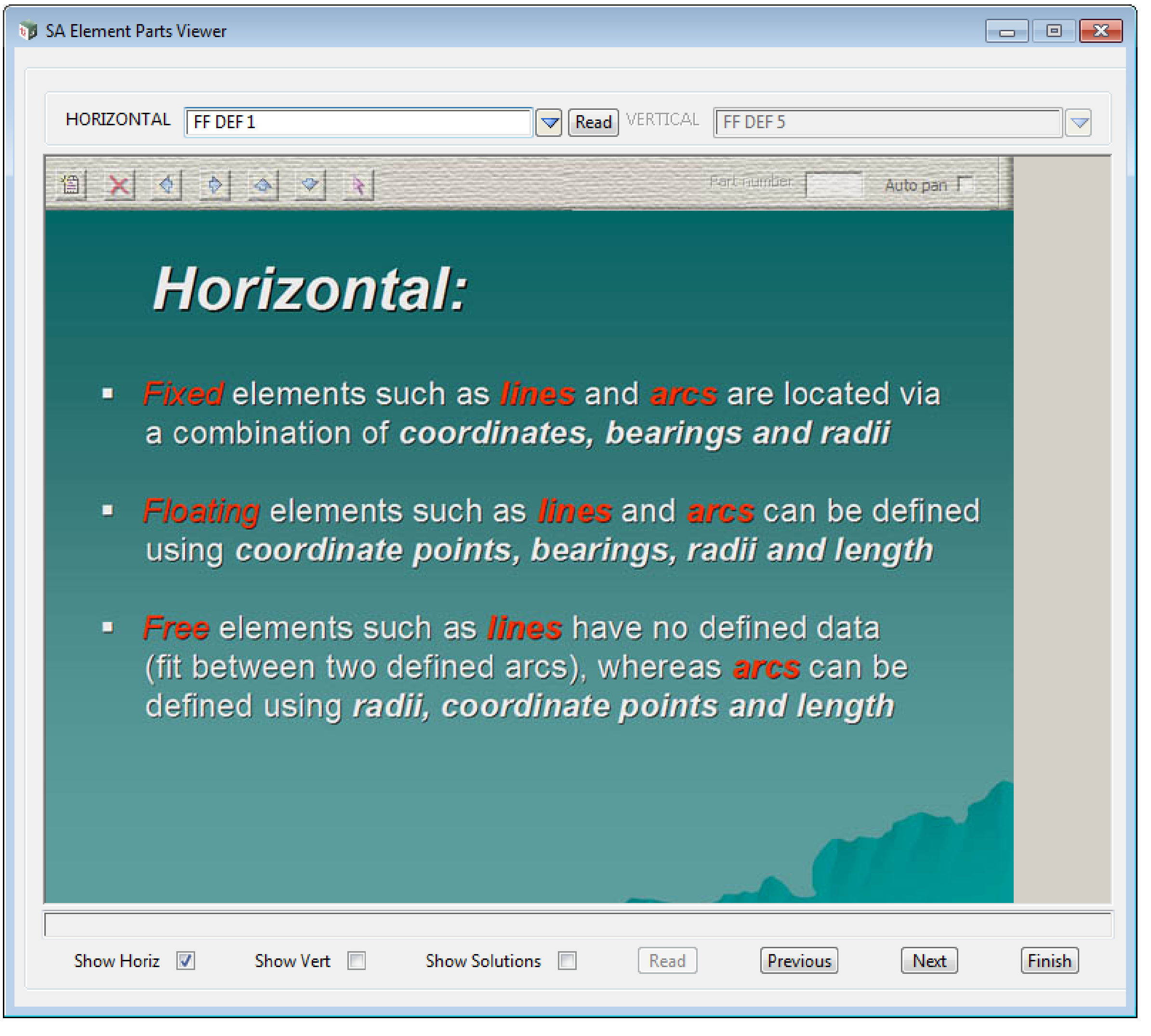


14.2.9 SA Element Parts Viewer

Position of option on menu: **Super Alignment Toolbar=>Help**

This option displays and explains all the parts in both horizontal and vertical, available in the Super Alignment.

Examples from the library can be read in for editing.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Horizontal	choice box		various
Quick selection of Horizontal parts			
Vertical	choice box		various

Quick selection of Vertical parts

Read Both button

Will read in 12da examples of all the parts. Most include vertical parts that are similar to their horizontal counter parts.

A plan view is created, called "SA Examples", and a model called "12D HOR ELEMENTS" is added to the plan view.

Show Horiz tick box

If ticked, allows the explanation and viewing of all horizontal parts

Show Vert tick box

If ticked, allows the explanation and viewing of all vertical parts

Show Solutions tick box

If ticked, allows the viewing of examples that can be read in

Read button

Will read in 12da examples of Super Alignments. Most include vertical geometry.

A plan view is created, called "SA Examples", and a model in the format "12D SA EXAMPLE 1" is added to the plan view.

Previous button

Step backward through examples

Next button

Step forward through examples

14.2.10 Super to SA Element Convert

Position of option on menu: **Road Toolbar**

This option is used to convert super strings that may have come in from other software.

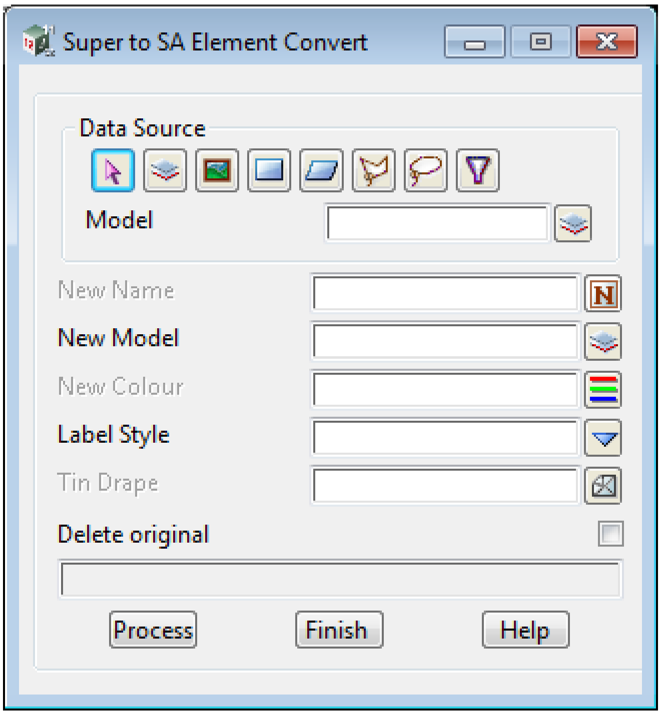
The conversion will attempt to create a Super Alignment made up in the horizontal of element parts.

The parts will mainly include at least one fixed part followed by floating parts of varying types (lines and arcs).

A label style can be set and a tin set in the vertical as a drape computator.

Note: If the original string had a geometry error (non-tangential arcs *e.g.*), then the resulting Super Alignment will show an error.
The error however will be easy to find as any parts prior to the error, will still show as solved.

Selecting the convert alignment option brings up the Super to SA Element Convert panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

New name	name box		select name

if non-blank, then the name of the selected strings will be changed to the name given in the new name *field.*

New model	model box		select model

model name for converted data

New colour	input		available colours

if non-blank, then the colour of the selected strings will be changed to the colour given in the new colour field.

Label style	choice box	full	various

Astyles for Super alignments

Tin Drape			tin select

if non-blank, then the tin is used as a vertical draped points computator for all converted strings

Delete original	tick box		tick

If ticked, delete original strings

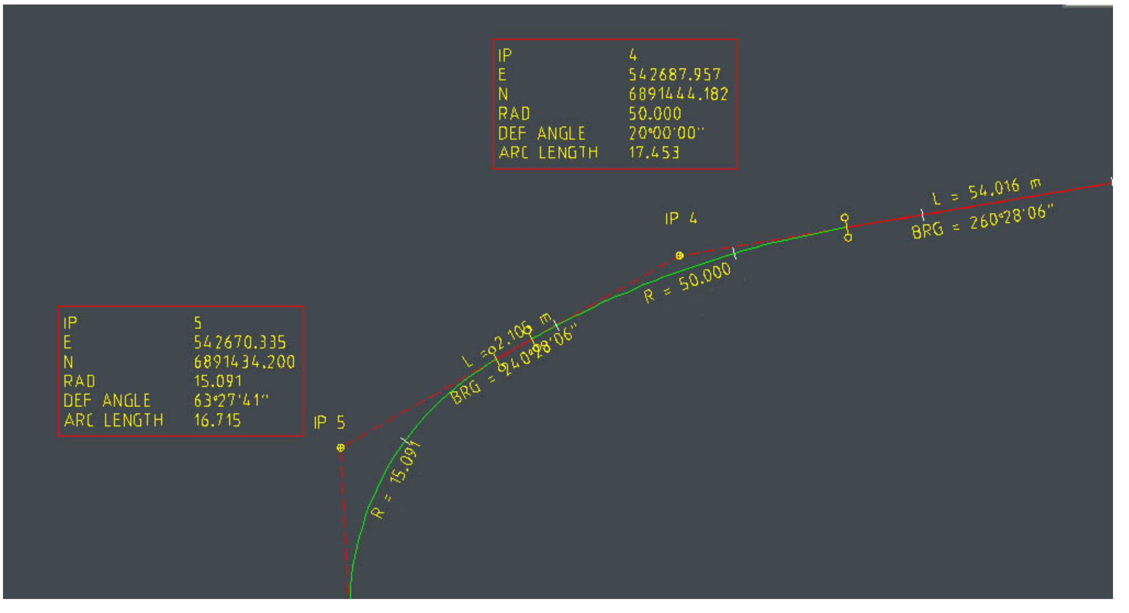
Process	Button		

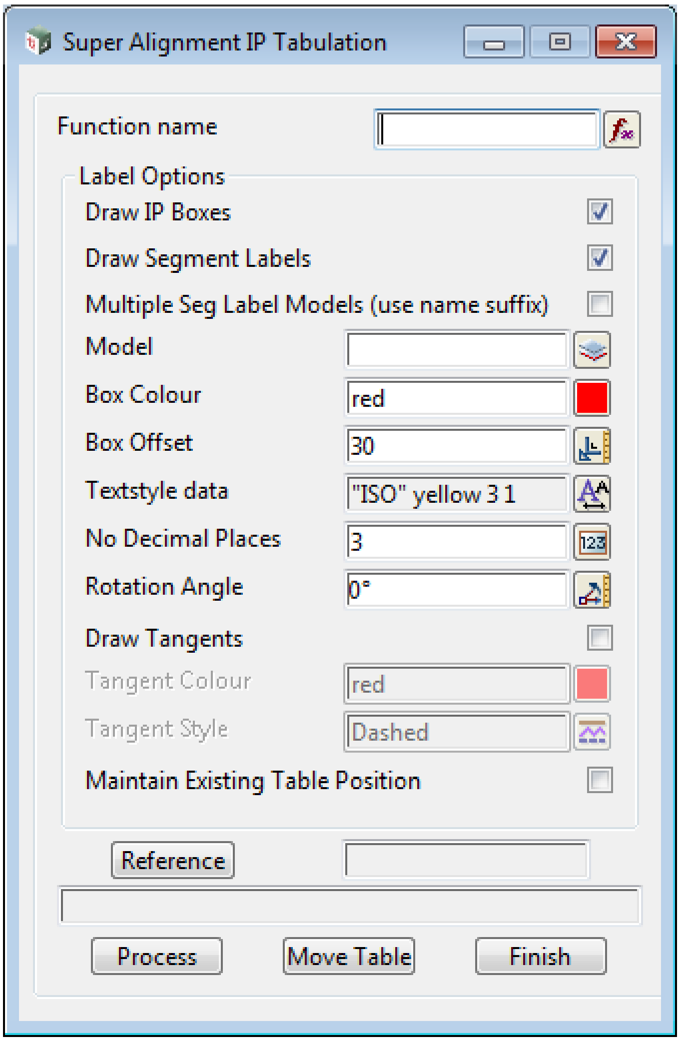
runs the option

14.2.11 Super Alignment IP Tabulation

Position of option on menu: **Super Alignment Toolbar=>Super Alignment IP tabulation**

This option tabulates a Super Alignment by placing individual information boxes at each calculated IP position.





The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function Name	function box		select function

name for use in recalc or chains

Label Options

Draw IP Boxes tick box

If ticked, create info boxes for all IP points

Draw Segment Labels tick box

If ticked, create segment labels

Multiple Seg Label Models (use name suffix) tick box

If ticked, create and label all segment parameters

Segment labels can be created and placed on individual models, using the name type as a suffix

Tabulation Model + "Radius"

Tabulation Model + "Arc Length"

Tabulation Model + "Length"

Tabulation Model + "Bearing"

Tabulation Model + "IP"

Tabulation Model + "Spiral"

Tabulation Model + "Tangents"

Note: Some of the segment labels have been superseded by the Super Alignment

Model model box select model

model name for tabulation

Box Colour input red available colours

the colour of the IP Info boxes

Box Offset measures box 75 At Point, Point to Point, String from Point, String to Point)

Distance to place the IP Info boxes from the IP

Textstyle Data input

text parameters for all text options

No Decimal Places input 3

Number of decimals for text

Rotation Angle input 0

Rotation angle if view rotated

Draw Tangents tick box

If ticked, tangent lines to be drawn at IPs

Tangent Colour input red available colours

the colour for the tangent lines

Tangent Style input dashed available styles

the linestyle for the tangent lines

Maintain Existing Table Positiontick box

If ticked, maintain IP Box positions during re-run of function

Reference string select

Super Alignment selection only

Process button

Creates the function and runs

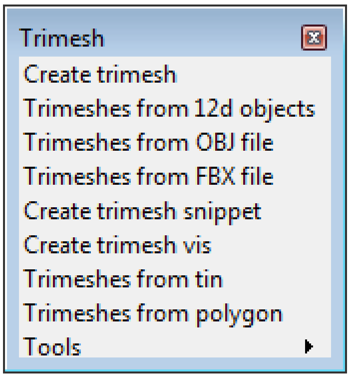
Move Table button

Select IP Tabulation Box to move for clarity (esc or right mouse button to cancel)

14.3 Trimesh

Position of option on menu: **Strings =>Trimesh**

The Trimesh options create trimeshes, and snippets to create trimeshes, when used in an Apply MTF.



See

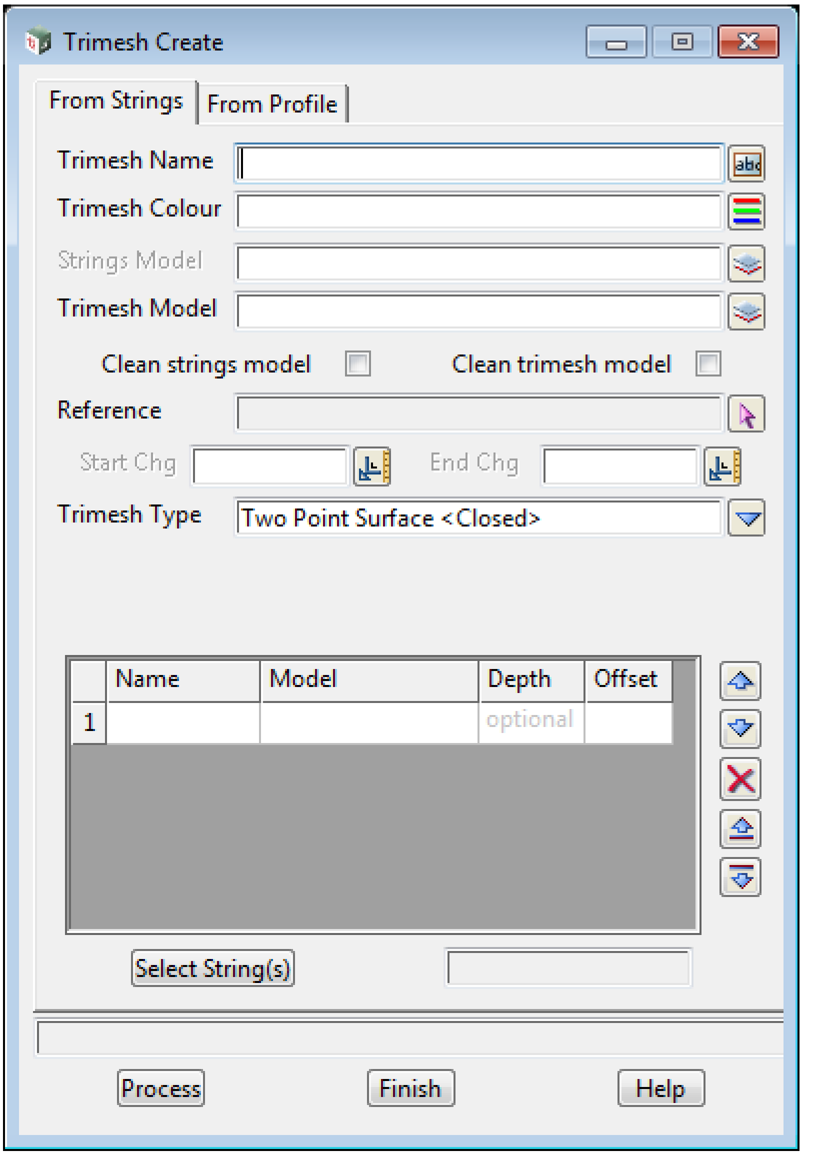
14.3.1 Create Trimesh
14.3.2 Trimeshes from 12d Objects
8.1.18 Wavefront OBJ Input
14.3.5 Create Trimesh Snippet
14.3.6 Create Trimesh Vis
14.3.7 Trimeshes from Tin
14.3.8 Trimeshes from Polygon

14.3.1 Create Trimesh

Position of option on menu: **Strings =>Trimesh =>Create trimesh**

THIS OPTION IS CURRENTLY UNDER DEVELOPMENT.

The option is for creating trimeshes from ***12d Model*** strings and a profile.



Continue to 14.3.2 Trimeshes from 12d Objects or return to 14.3 Trimesh.

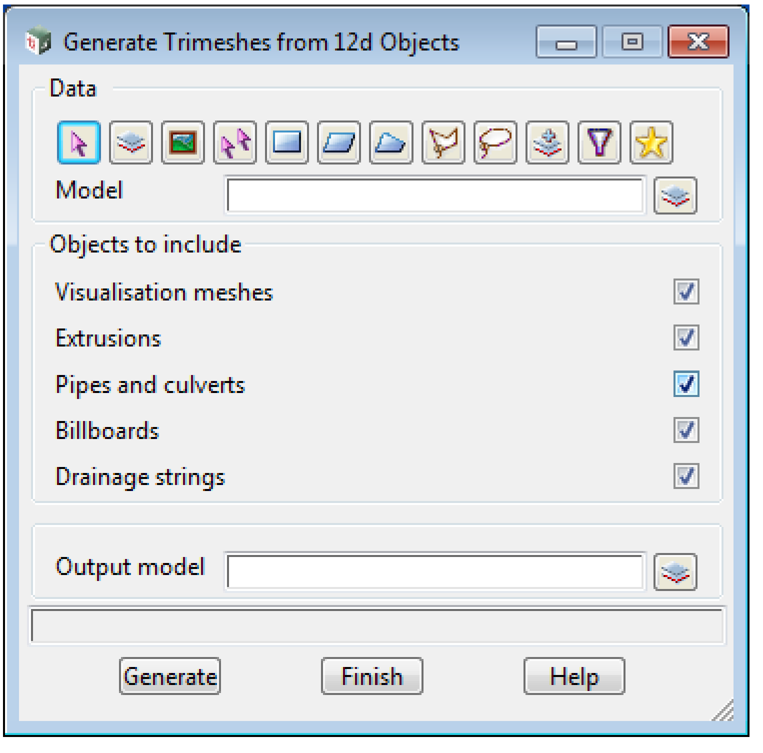
14.3.2 Trimeshes from 12d Objects

Generate_Trimeshes_from_12d_ObjectsPosition of option on menu: **Strings =>Trimesh =>Trimeshes from 12d objects**

This option creates trimeshes for any mesh, extrude, super string pipe or culvert, drainage/sewer string or billboard, in the selected data.

The trimeshes can then be added to a section view, or cross section plots, to see that they are correctly sectioned through.

Selecting **Trimeshes from 12d Objects** brings up the **Generate Trimeshes from 12d Objects** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to process. For any meshes, a trimesh is created.

Visualisation meshes	tick box	tick	

*if **ticked**, any meshes in the data source are converted to trimeshes.*
*If **not ticked,** then meshes are not converted to trimeshes.*

Extrusions	tick box	tick	

*if **ticked**, any extrudes in the data source are converted to trimeshes.*
*If **not ticked,** then extrudes are not converted to trimeshes.*

Pipes and culverts	tick box	tick	

*if **ticked**, any super string pipe or culvert in the data source are converted to trimeshes.*
*If **not ticked,** then super string pipe or culverts are not converted to trimeshes.*

Billboard tick box tick

*if **ticked**, any billboard in the data source are converted to trimeshes.*
*If **not ticked,** then billboards are not converted to trimeshes.*

Drainage strings tick box tick

*if **ticked**, any drainage/sewer strings in the data source are converted to trimeshes.*
*If **not ticked,** then drainage/sewer strings are not converted to trimeshes.*

Output model model box available models

model to add all the created trimeshes to.

Generate button

create trimeshes for all the selected type that are in the data source.

Continue to 14.3.3 Trimeshes from Wavefront OBJ File or return to 14.3 Trimesh.

14.3.3 Trimeshes from Wavefront OBJ File

Position of option on menu: **File I/O =>Data input =>OBJ**

Position of option on menu: **Strings =>Trimesh =>Trimeshes from OBJ file**

This is documented in 8.1.18 Wavefront OBJ Input.

Continue to 14.3.4 Trimeshes from FBX File or return to 14.3 Trimesh.

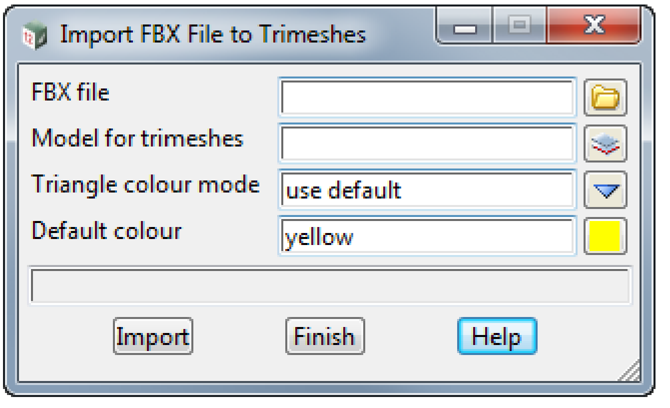
14.3.4 Trimeshes from FBX File

Position of option on menu: **Strings =>Trimesh =>Trimeshes from FBX file**

Position of option on menu: **File I/O =>Data input =>FBX**

FBX (Filmbox) is a proprietary format developed by Kaydara and owned by AutoDesk since 2006, to provide interoperability between digital content creation applications.

This option reads in a FBX file and converts all possible objects to trimeshes.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
FBX file	file box		*.fbx files

name of the FBX file to be read in

Model for trimeshes	model box		available models

model to add all the created trimeshes to.

Triangle colour mode	choice box		use default, RGB of diffuse colour material name

*if **user default**, use the* Default colour *for the trimeshes.*

*If **RGB diffuse colour**, use the RGB diffuse colour of the FBX triangle object as the triangle colour.*

*If **material name**, use the material name of the FBX triangle as the triangle colour.*

Default colour	colour box		available colours

default colour for the trimeshes.

Import	button		

read the FBX file and where ever possible create trimeshes from the FBX data.

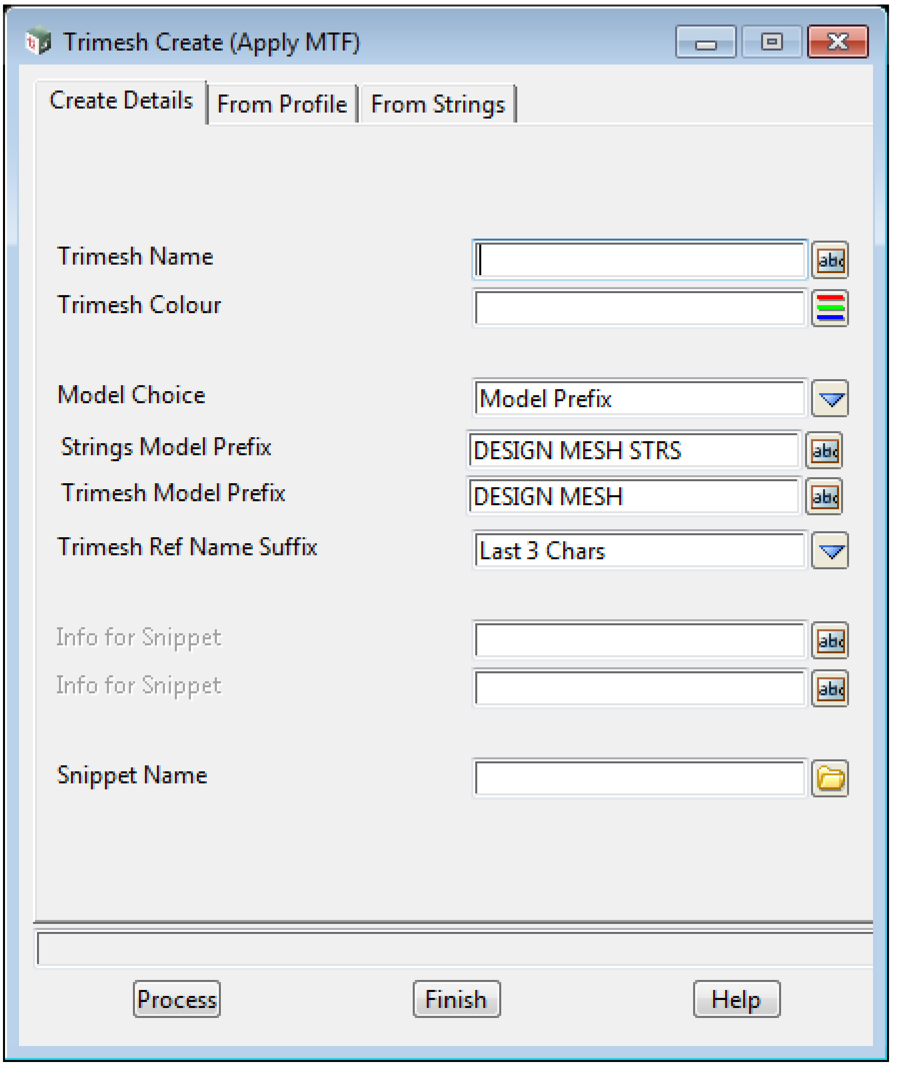
Continue to 14.3.5 Create Trimesh Snippet or return to 14.3 Trimesh.

14.3.5 Create Trimesh Snippet

Position of option on menu: **Strings =>Trimesh =>Create trimesh snippet**

THIS OPTION IS CURRENTLY UNDER DEVELOPMENT.

The option is for creating snippets that will create trimeshes using an *Apply MTF*.



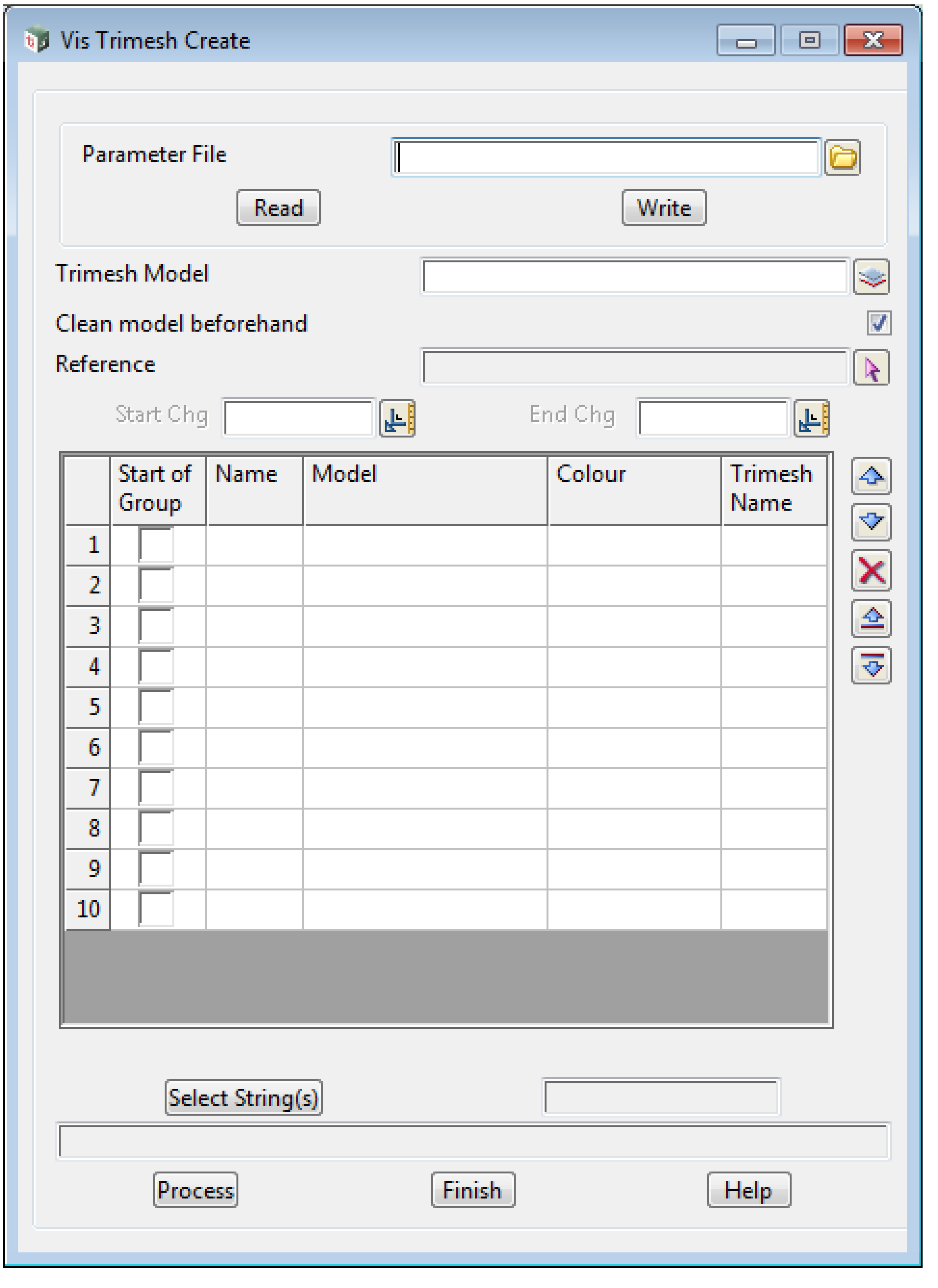
Continue to 14.3.6 Create Trimesh Vis or return to 14.3 Trimesh.

14.3.6 Create Trimesh Vis

Position of option on menu: **Strings =>Trimesh =>Create trimesh vis**

THIS OPTION IS CURRENTLY UNDER DEVELOPMENT.

The option is for creating.



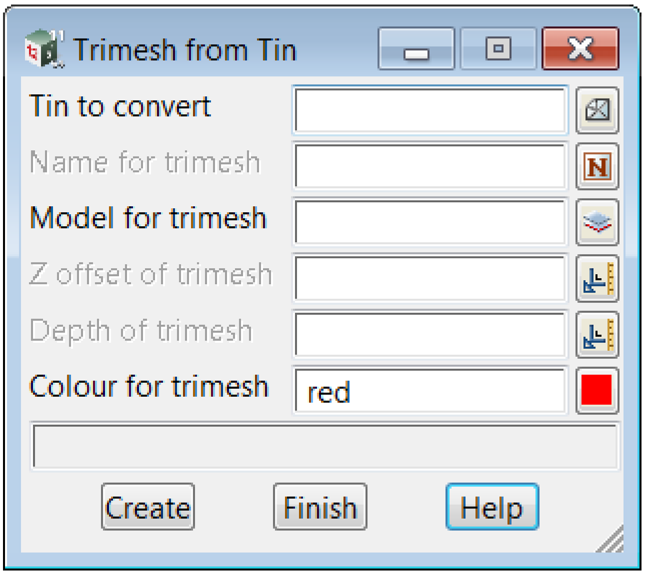
Continue to 14.3.6 Create Trimesh Vis or return to 14.3 Trimesh.

14.3.7 Trimeshes from Tin

Position of option on menu: **Strings =>Trimesh =>Trimeshes from tin**

This option takes a tin and creates a trimesh by first moving the tin up vertically by a use given Z offset of trimesh to form the top of the trimesh, and then copying the top a vertical down by a user given Depth of trimesh.

Selecting **Trimeshes from tin** brings up the **Generate Trimesh from tin** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin to convert	tin box		
tin to create a trimesh from.			
Name for trimesh	name box		
name for the created trimesh.			
Model for trimesh	model box		available models
model to add the created trimesh to.			
Z offset of trimesh	real box		
the top of the trimesh is the tin translated vertically up by this value. *This value can be negative and then the top of the trimesh is lower rather than above the tin.*			
Depth Z offset of trimesh	real box		
the bottom of the trimesh is this value vertically below the top of the trimesh. *This value can be negative then the "bottom" of the trimesh is above the "top" of the trimesh.*			
Colour for trimesh	colour box		available colours
colour for the created trimesh.			
Create	button		
create trimesh from the tin.			

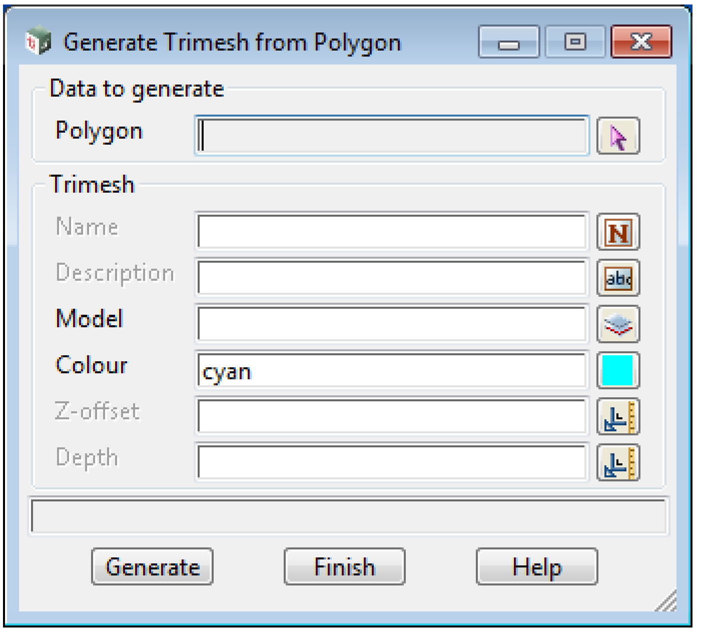
Continue to 14.3.8 Trimeshes from Polygon or return to 14.3 Trimesh.

14.3.8 Trimeshes from Polygon

Position of option on menu: **Strings =>Trimesh =>Trimeshes from polygon**

This option takes a polygon and first creates a tin of the polygon, and then creates a trimesh by first moving the tin up vertically by a use given Z offset of trimesh to form the top of the trimesh, and then copying the top a vertical down by a user given Depth of trimesh.

Selecting **Trimeshes from polygon** brings up the **Generate Trimesh from Polygon** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Polygon	string select		

polygon to create the trimesh from.

Field Description	Type	Defaults	Pop-Up
Name	name box		

name for the created trimesh.

Field Description	Type	Defaults	Pop-Up
Description	input		

a description that is recorded in the trimesh.

Field Description	Type	Defaults	Pop-Up
Model	model box		available models

model to add the created trimesh to.

Field Description	Type	Defaults	Pop-Up
Colour	colour box		available colours

colour for the created trimesh.

Field Description	Type	Defaults	Pop-Up
Z-offset	real box		

the top of the trimesh is the tin of the polygon translated vertically up by this value.

This value can be negative and then the top of the trimesh is lower rather than above the tin of the polygon.

Field Description	Type	Defaults	Pop-Up
Depth	real box		

the bottom of the trimesh is this value vertically below the top of the trimesh of the polygon.

This value can be negative then the "bottom" of the trimesh is above the "top" of the trimesh.

Field Description	Type	Defaults	Pop-Up
Generate	button		

create trimesh from the polygon.

Continue to 14.3.9 Trimesh Tools or return to 14.3 Trimesh.

14.3.9 Trimesh Tools

Position of option on menu: **Strings =>Trimesh**

See

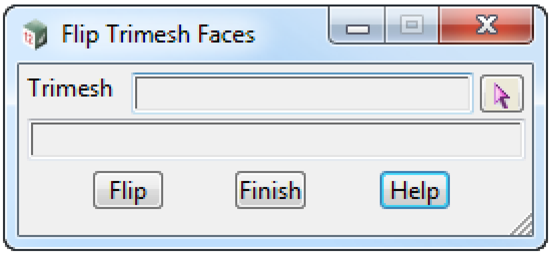
14.3.9.1 Trimesh Flip Faces

14.3.9.1 Trimesh Flip Faces

Position of option on menu: **Strings =>Trimesh =>Tools =>Flip Trimesh Faces**

This option reverses the order of the vertices in all the triangle in a trimesh.

Selecting **Flip faces** brings up the **Flip Trimesh Faces** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Trimesh	trimesh select		

trimesh to flip the faces.

Flip	button		

reverse the order of the vertices in all the triangles in the selected trimesh.

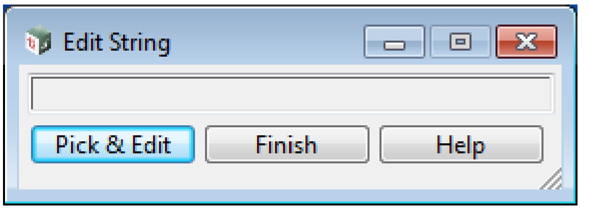
Return to 14.3 Trimesh.

14.4 Editor

Position of option on menu: **Strings =>Editor**

The string editor is used for modifying any of the *12d Model* strings.

After selecting the **editor** option, the **edit string** panel is placed on the screen to record any error messages.



The option is already in **pick** mode (the **pick & edit** button only needs to be selected if the pick was cancelled) and the user simply picks and accepts the string to be edited. From the picked string's type, the editor is able to determine the appropriate editor and placed on the screen.

Each string type has its own edit operations which will now be discussed in detail.

Note - if the **edit** menu is deleted from the screen using the **menu delete** option, it can be raised again by clicking RB in any view.

For the option *2d*, go to 14.4.2 Edit 2d
3d 14.4.3 Edit 3d
4d 14.4.4 Edit 4d
Arcs 14.4.5 Arc Edit
Circles 14.4.6 Circle Edit
Feature 14.4.7 Feature Edit
Edit Pipe 14.4.8 Edit Pipe
Edit Polyline 14.4.9 Edit Polyline
Super 14.4.10 Edit Super
Super Alignment 14.4.11 Edit Super Alignment
Text 14.4.12 Text Edit

For the old string options:
2d, go to 14.20.3 2d Edit - Old.
3d - old 14.20.4 3d Edit - Old
4d - old 14.20.5 4d Edit - Old
Alignment 14.20.2 Alignment Edit
Pipe - old 14.20.6 Pipe Edit - Old
Polyline - *old* 14.20.7 Polyline Edit - Old

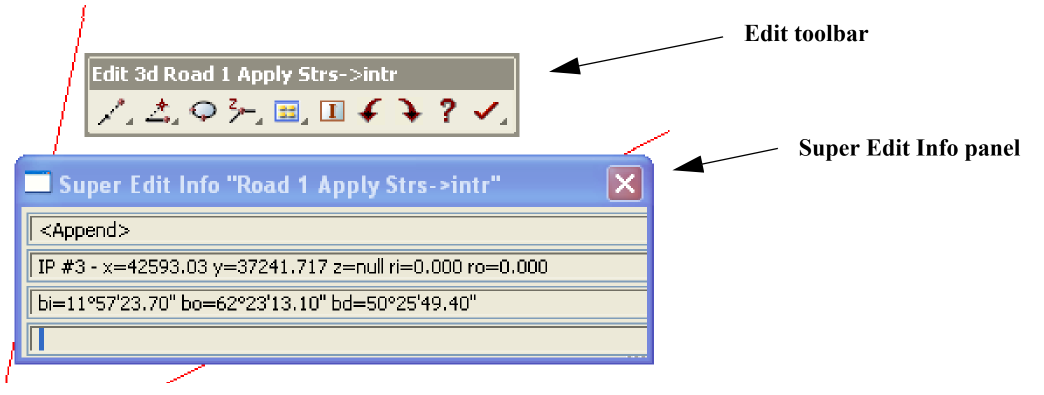
Please continue to the next section 14.4.1 Super String Edit - Common Information.

14.4.1 Super String Edit - Common Information

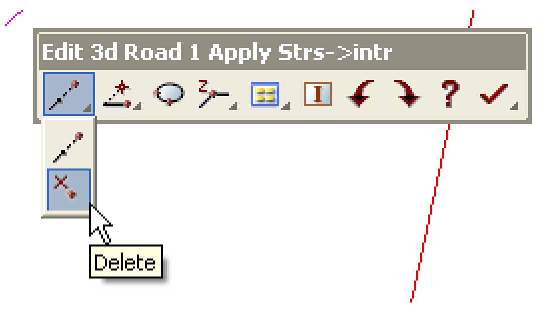
14.4.1.1 nd edit menu

After selecting the string to be edited, the toolbar containing the list of available edit options (the string's **Edit nd** menu) is placed on the screen. If **Display edit info** is ticked on in the **Default** panel (**Project** =>**Management** =>**Defaults**), an information panel, (the string's **Edit nd info** panel) is also displayed.

For example, the **Edit String** menu and **Super Edit Info** panel for a 3d super string are

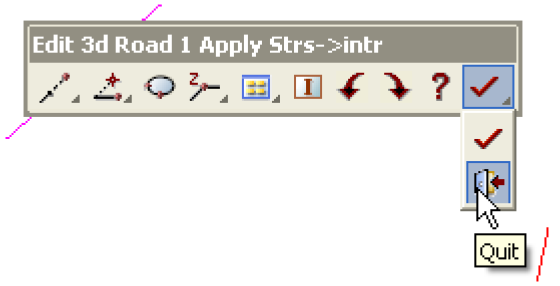


Most of the options chosen from the **Edit nd** menus repeat until cancelled by the user. That is, the option remains in force and can be applied repeatedly until terminated by the user.

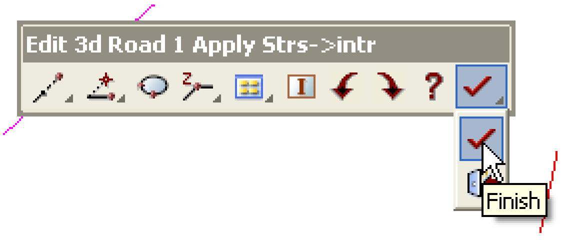


For example, if the **Delete** option is selected, any number of vertices can be deleted from the chosen string without re-selecting the **Delete** option.For full information on the **Delete** option, go to the section 14.4.1.2 Delete

To select a new edit nd option, simply select the option from the **Edit nd** toolbar. The current option is automatically terminated and the new option begins.



An option is also terminated when the entire string editing option is ended by selecting either the **Quit** or **Finish** button in the **Edit nd** menu.

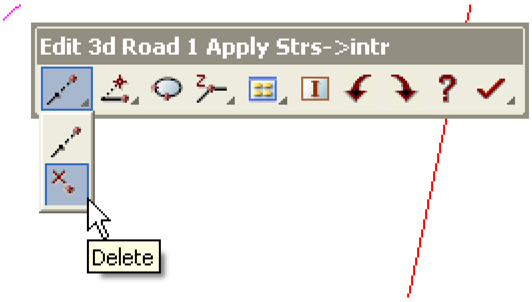


Notes

1. Depending on its breakline (point-line) type, a string with default style ("1") is displayed with crosses at each of its vertices (point type) or with straight lines joining the vertices (line type). Linestyles can be defined that drawn lines between the vertices even though the breakline type is **point**.
2. The string vertices are called intersection points (IPs).
3. The string **Super Edit Info** panel will not automatically appear if **Display edit info** is set to **off** in the ***System Settings*** tab of the **Project =>Management =>Defaults** panel.

See 14.4.1.2 Delete
14.4.1.3 Move
14.4.1.4 Insert
14.4.1.5 Between
14.4.1.6 Extend
14.4.1.7 Open/Close
14.4.1.8 Properties
14.4.1.9 Info
14.4.1.10 Undo/Redo
14.4.1.11 ‘?’ Help
14.4.1.13 Quit
14.4.1.12 Finish
14.4.1.14 Typed input

14.4.1.2 Delete



The **delete** option is used to delete selected vertices from the string.

The vertex to be deleted is picked (LB) and accepted (MB). When the vertex is accepted, it is deleted. The string, minus the deleted point, is then redrawn.

Once a vertex has been deleted, another vertex in the string can then be selected and deleted. Hence any number of vertices from the string can be deleted one after another.

message area 1 <Delete>

Screen message area

<Select a vertex to delete> [picks][fast][menu]

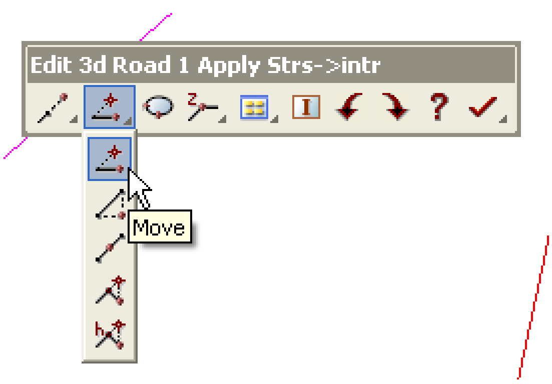
The **delete** option is terminated by selecting **cancel** from the **pick ops** menu or by selecting a new option from the **Edit nd** menu.

Typed input can be used to select a point for deletion.

Continue to the next section 14.4.1.3 Move or return to 14.4.1 Super String Edit - Common Information.

14.4.1.3 Move

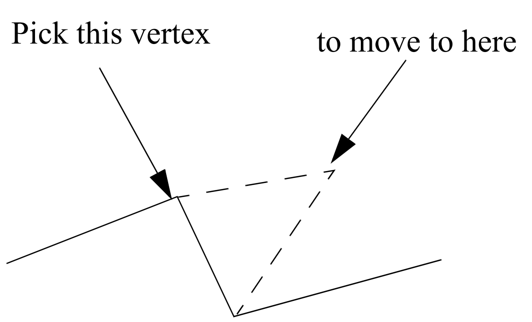
The **move** option is for moving individual vertices (intersection points - IPs) of the string.



The move cycle consists of two steps:

(a) selecting the vertex to move

(b) selecting the new position for the vertex.



Step (a)

First the point to be moved is selected. The selected vertex will then move around the view and the string redrawn to show the change as the cursor is moved.

message area 1 <Move>

Screen message area

<Pick vertex to move> [picks][fast][Menu

Step (b)

The current cursor position is selected as the new position for the vertex by selecting (LB) and accepting (MB). The vertex being moved is then anchored at the cursor position for the vertex, and the string redrawn.

message area 1 <Move>

Screen message area

<Pick new location of vertex> [picks][fast][menu]

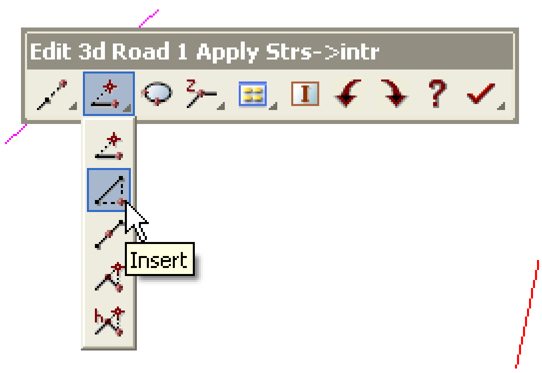
Typed input can be used in either step.

Once the move cycle is completed and the vertex moved, the move option is still current and can be repeated for other vertices without having to re-select the **move** option.

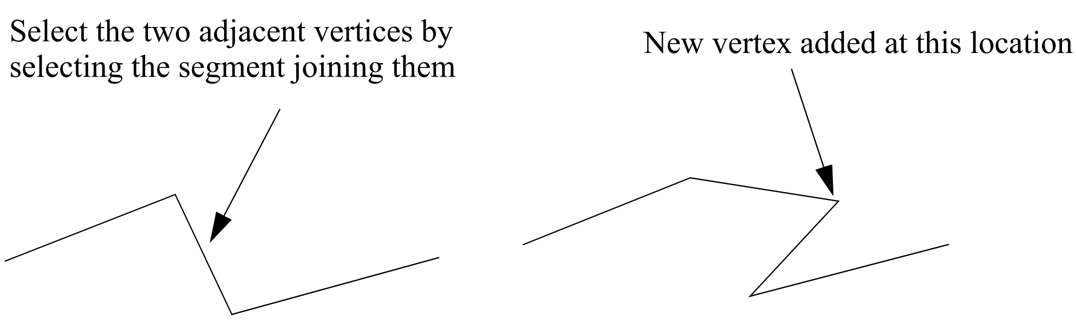
The **Move** option is terminated by selecting **cancel** from the **pick ops** menu or by selecting a new **Edit nd** option.

Continue to the next section 14.4.1.4 Insert or return to 14.4.1 Super String Edit - Common Information.

14.4.1.4 Insert



The **insert** option is designed to place a new vertex in a string between two adjacent vertices (note that the inserted point **does not** have to be on the segment joining the two vertices).



Inserting a vertex, like moving a vertex, is a two step process.

Step (a) - selecting the vertices to be on either side of the new vertex by selected the segment

The two adjacent vertices are chosen by selecting the segment connecting the two vertices. Once the segment is selected, the new vertex is assumed to be at the current cursor position. As the cursor is moved, the string is redrawn reflecting the changing position of the inserted vertex.

message area 1 <Insert >

Screen message area

<Pick position to insert vertex> [picks][fast][menu]

Step (b) - selecting the position for the new vertex

The position of the new vertex is selecting

message area 1 <Insert >
Screen message area
<Pick new location of vertex> [picks][fast][menu]

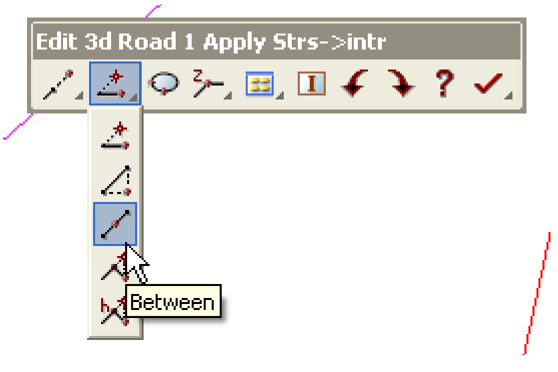
Once the insert cycle is completed and the vertex inserted, the insert option is still current and can be repeated for other insertions without having to re-select the **insert** option.

The **insert** option is terminated by selecting **Cancel** from the **Pick ops** menu or by selecting a new option from the **Edit nd** menu.

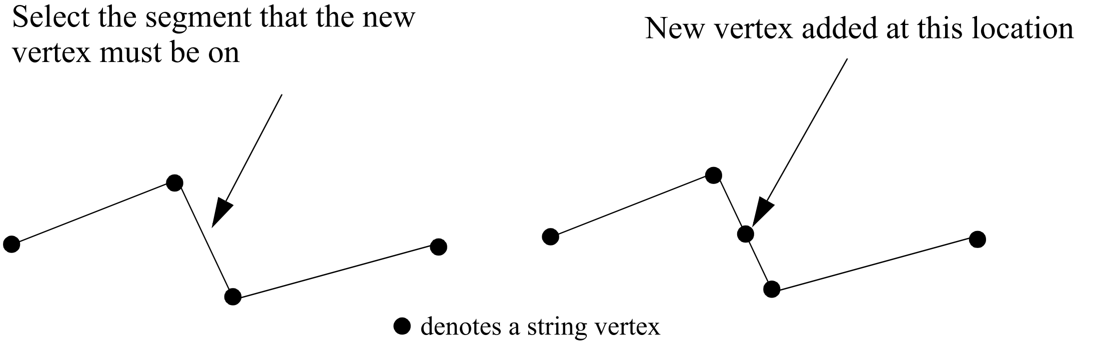
Typed input can be used in either step.

Continue to the next section 14.4.1.5 Between or return to 14.4.1 Super String Edit - Common Information.

14.4.1.5 Between



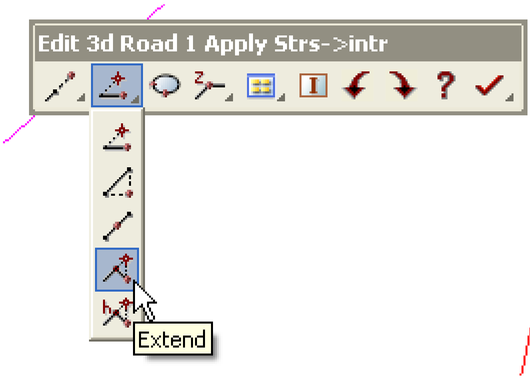
The **between** option is similar to the **insert** option except that the inserted vertex does have to be on the segment joining the two vertices. To accomplish this, the cursor position is projected onto the segment to give the new vertex position.



The **between** option is terminated by selecting **cancel** from the **pick ops** menu or by selecting a new option from the **Edit nd** menu.

Continue to the next section 14.4.1.6 Extend or return to 14.4.1 Super String Edit - Common Information.

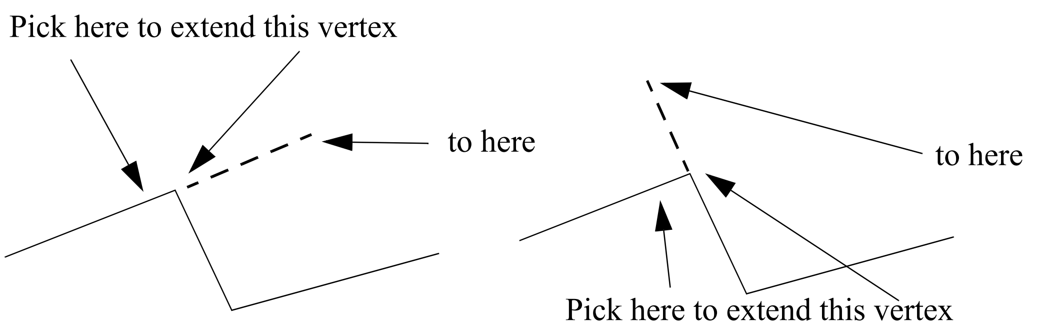
14.4.1.6 Extend



The **extend** option is used to move a vertex along the line joining the vertex to its neighbouring vertex.

The **extend** option is used to move a vertex along the line joining the vertex to its neighbouring vertex.

That is, the bearing of the segment is kept constant and the vertex is moved along that segment either towards or away from its neighbouring vertex on the segment.



Extending, like moving a vertex, is a two step process.

Step (a) - selecting the segment and the vertex to be moved along that segment

The segment and the vertex to be moved are chosen in the one operation by picking (LB) and accepting (MB) at a co- ordinate point near the segment and close to the vertex to be moved along that segment.

Once the segment and vertex are selected, the new position of the selected vertex is assumed to be at the current cursor position projected along the selected segment.

As the cursor is moved, the string is redrawn reflecting the changing position of the moved vertex.

message area 1 <Extend>
Screen message area
<Pick position to extend> [picks][fast][Menu]

Step (b) - selecting the final position for the vertex

The final position for the vertex is set to the projection of the selected cursor position onto the segment.

message area 1 <Extend>
Screen message area
<Pick new location of vertex> [picks][fast][menu]

Once the extend is completed, the extend option is still current and can be repeated without re-selecting the extend option.

The **extend** option is terminated by selecting **cancel** from the **pick ops** menu or by selecting a new option from the **Edit nd** menu.

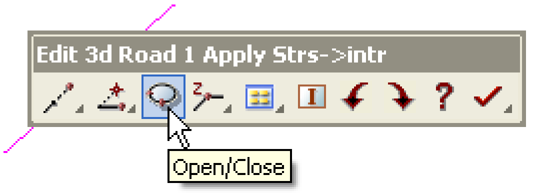
Typed input can be used in either step.

Note - Extend can be used on the end points of the string.

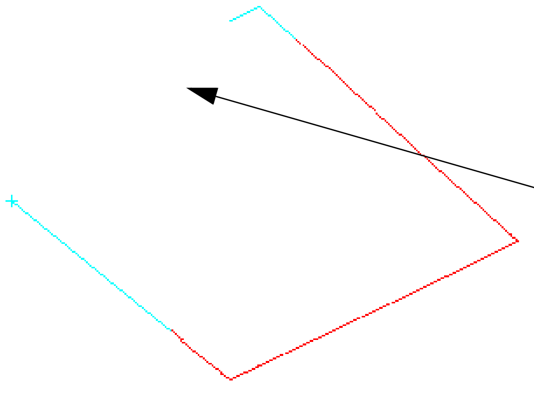
Continue to the next section 14.4.1.7 Open/Close or return to 14.4.1 Super String Edit - Common

Information.

14.4.1.7 Open/Close



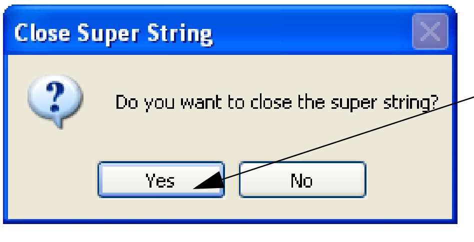
Toggles a string between Open and Closed.



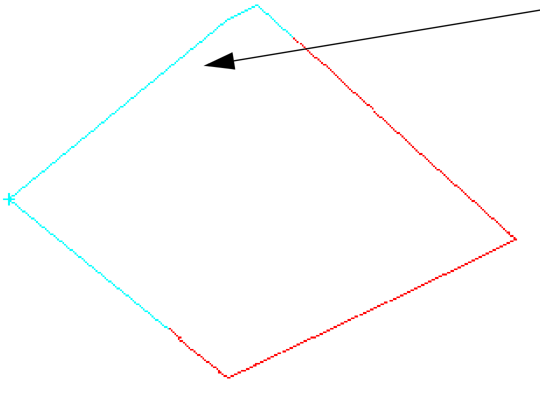
If the string is open (that is, the start and end vertices are not the same), selecting the **Open/ Close** option adds a segment joining the first and last vertices of the string.

open string

Selecting **Yes** If the string is open adds a segment, joining the first and last vertices of the string.



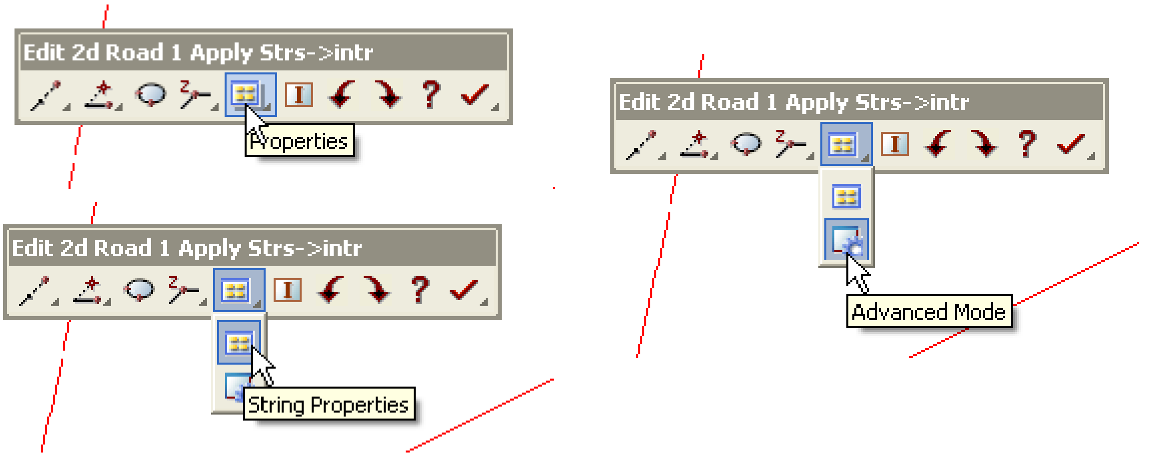
The string is now closed.



If the string is closed (that is, there is a segment from the start to the end vertices), selecting the **Open/Close** option removes the last segment of the string.

Continue to the next section 14.4.1.8 Properties or return to 14.4.1 Super String Edit - Common Information.

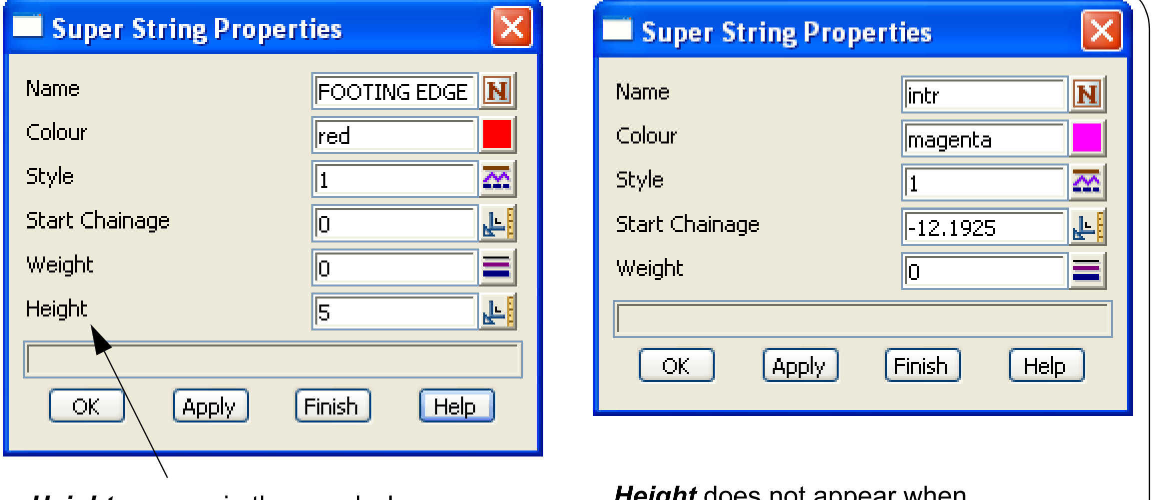
14.4.1.8 Properties



Selecting Properties brings up the **Super String Properties** panel which is used to modify the string's header information.

What is displayed in the **Super String Properties** panel depends on the type of the super string. That is, it depends on what properties of the super string have been enabled.

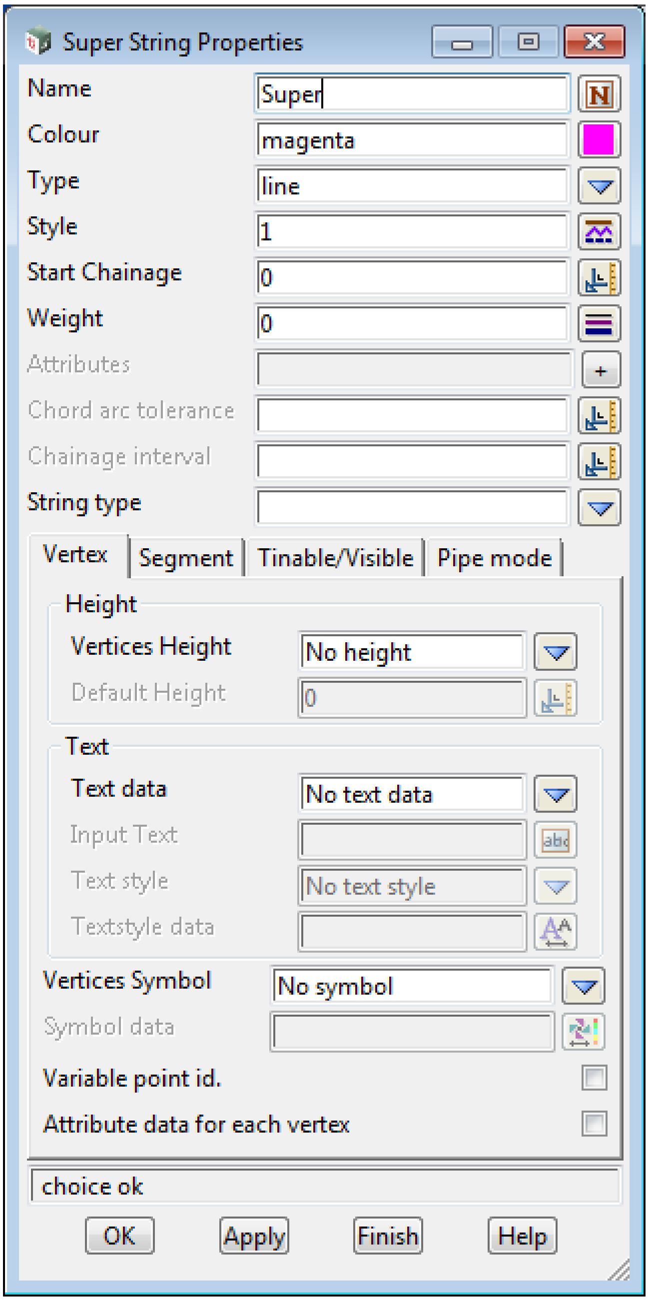
For example, constant height enabled means that there is only one height for the entire super string so height appears in the S**uper String Properties** panel. If variable height is enabled, there is a height for each vertex and so height does not appear in the S**uper String Properties** panel.



Height appears in the panel when ***Constant height*** is set for the super string (*e.g.* 2d super string)

Height does not appear when ***Variable height*** s set for the super string (*e.g.* 3d super string)

In the ***Advanced*** mode, all the super string properties can be modified and so all possible super string fields are displayed in the **Super String Properties** panel.



The fields in the panels are similar to those in the **Create Super String** panels. The only new fields are the **OK** and **Apply** which are used in the normal way.

Continue to the next section 14.4.1.9 Info or return to 14.4.1 Super String Edit - Common Information.

14.4.1.9 Info

Toggles the **Super Edit Info** panel on/off for the string.

For more information on the Super Edit Info panel see 14.4.1.1 nd edit menu in the section 14.4.1 Super String Edit - Common Information.

Continue to the next section 14.4.1.10 Undo/Redo or return to 14.4.1 Super String Edit - Common Information.

14.4.1.10 Undo/Redo

An undo and redo list is defined for each editor on the screen. That is, each editor has its own undo/redo lists. If an option is undone, it is added to the redo list so that it can be redone.

When the editor is exited by either **Quit** or **Finish**, the undo and redo lists are deleted and are no longer usable.

14.4.1.10.1 Undo

Undoes the last editor operation for this string and adds it to the top of the redo list.

Undoes the last editor operation for this string and adds it to the top of the redo list.

14.4.1.10.2 Redo

Redo the last editor operation that was undone for this string. Add the undone operation to the top of the undo list.

Continue to the next section 14.4.1.11 ‘?’ Help or return to 14.4.1 Super String Edit - Common Information.

14.4.1.11 ‘?’ Help

Help button to go to the Help topic for the panel.

See the earlier section 14.4.1 Super String Edit - Common Information for general information about editing strings, the panel **Super Edit Info**, and the options, 14.4.1.2 Delete,14.4.1.4 Insert,14.4.1.3

Move,14.4.1.5 Between,14.4.1.6 Extend, 14.4.1.7 Open/Close, 14.4.1.8 Properties, 14.4.1.9 Info, 14.4.1.10.1 Undo, 14.4.1.10.2 Redo, 14.4.1.11 '?' Help, 14.4.1.12 Finish, 14.4.1.13 Quit and 14.4.1.14 Typed input.

For the **Edit 2d** options not mentioned in 14.4.1 Super String Edit - Common Information see 14.4.2 Edit 2d

For the **Edit 3d** options not mentioned in 14.4.1 Super String Edit - Common Information see 14.4.3 Edit 3d

For the **Edit 4d** options not mentioned in 14.4.1 Super String Edit - Common Information see 14.4.4 Edit 4d

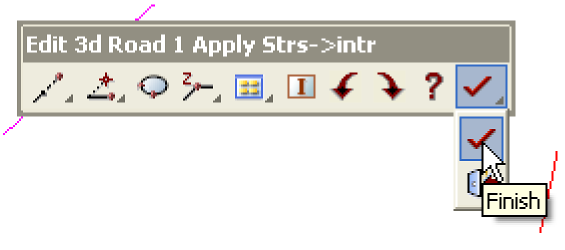
For the **Edit Super** options not mentioned in 14.4.1 Super String Edit - Common Information see 14.4.10 Edit Super

For the **Edit Pipe** options not mentioned in 14.4.1 Super String Edit - Common Information see 14.4.8 Edit Pipe

For the **Edit Polyline** options not mentioned in 14.4.1 Super String Edit - Common Information see 14.4.9 Edit Polyline

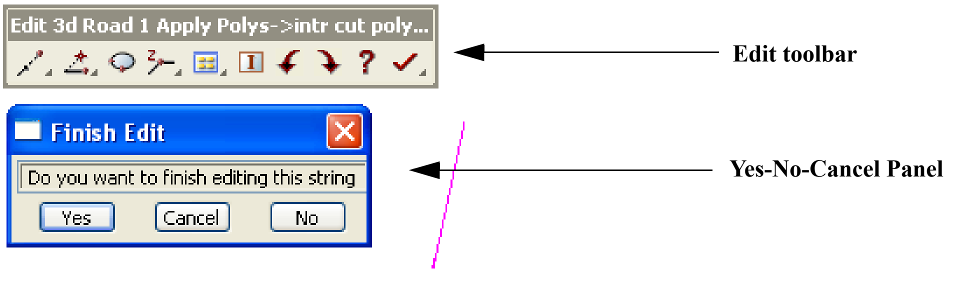
Continue to the next section 14.4.1.12 Finish or return to 14.4.1 Super String Edit - Common Information.

14.4.1.12 Finish



The edit process is completed and the modified string stored in the given model when the **Finish** option is chosen from the **Edit nd** menu.

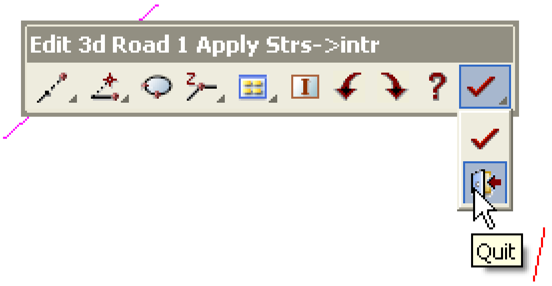
After selecting the **Finish** option, a **Yes-No-Cancel** panel is displayed to confirm that the edit is to be ended. If **Yes** is selected, the edits will be stored.



Continue to the next section 14.4.1.13 Quit or return to 14.4.1 Super String Edit - Common Information.

14.4.1.13 Quit

Even after points are created for the new string, the create process can be aborted by selecting the **Quit** option from the **Edit nd** menu. The option then terminates and no string is created.



The **Quit** option is used to terminate the nd string edit session without storing any of the edits. That is, the string is left as it was *before* the edit was begun

After selecting the **Quit** option, a **Yes-No-Cancel** panel is displayed to confirm that the edit is to be ended. If **yes** is selected, the edits will be ignored and the unedited string kept.

Note that if it was a new string being created, the option terminates by Quit and no string is created.

Continue to the next section 14.4.1.14 Typed input or return to 14.4.1 Super String Edit - Common Information.

14.4.1.14 Typed input

Typed input can be used wherever a select is required to obtain a co-ordinate. For example an (x,y), (x, y, z) or (chainage, height).

To start typed input, start typing, and a special typed-input box appears on the screen

If the focus is on a Plan or Perspective view, then an **Enter X Y Z** box comes up when you are placing horizontal geometry for an alignment or super alignment, or vertices for all other string types. If you are creating vertical geometry for an alignment or super alignment and the focus is on a Section view, then a **Enter Ch Ht** box appears.

These are also called the *XYZ typed input* box and the *Chainage Height typed input* box respectively.

An <enter> terminates the typed input and the entered values taken as the position of the required point. The typed-input box then disappears.

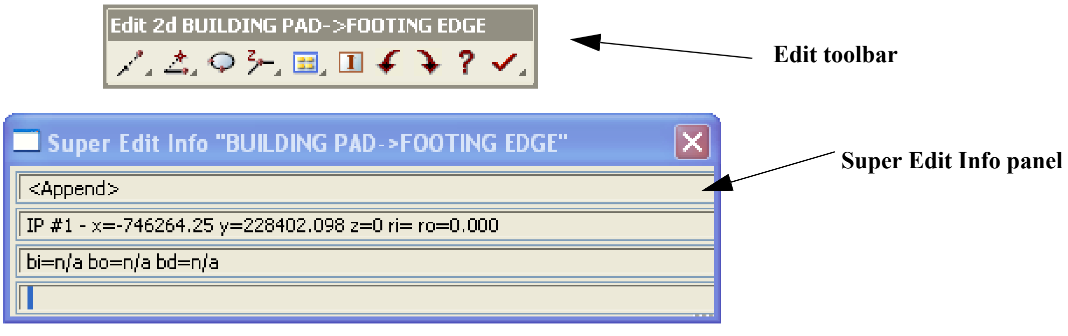
If the user wishes to abort the typed input and return to mouse input, simply select the [X] button on the typed input box or type <enter> with no values in the typed input box.

Continue to the next section 14.4.2 Edit 2d or return to 14.4.1 Super String Edit - Common Information.

14.4.2 Edit 2d

Position of option on menu: **Strings =>Editor**

On picking a 2d string, the **Edit 2d** toolbar is placed on the screen. If **Display edit info** is ticked on in the **Default** panel (**Project =>Management =>Defaults**), an information panel, (the string's **Super Edit Info** panel) is also displayed.



See the earlier section 14.4.1 Super String Edit - Common Information for general information about editing strings, the panel **Super Edit Info**, and the options, 14.4.1.2 Delete,14.4.1.4 Insert,14.4.1.3 Move,14.4.1.5 Between,14.4.1.6 Extend, 14.4.1.7 Open/Close, 14.4.1.8 Properties, 14.4.1.9 Info, 14.4.1.10.1 Undo, 14.4.1.10.2 Redo, 14.4.1.13 Quit, 14.4.1.11 '?' Help, 14.4.1.12 Finish and 14.4.1.14 Typed input.

Each **Edit 2d** option not mentioned in the earlier section 14.4.1 Super String Edit - Common Information will now be described.

See 14.4.2.1 2d - Append
14.4.2.2 2d - Height
14.4.2.3 2d - Height (Nav)
14.4.2.4 2d Properties

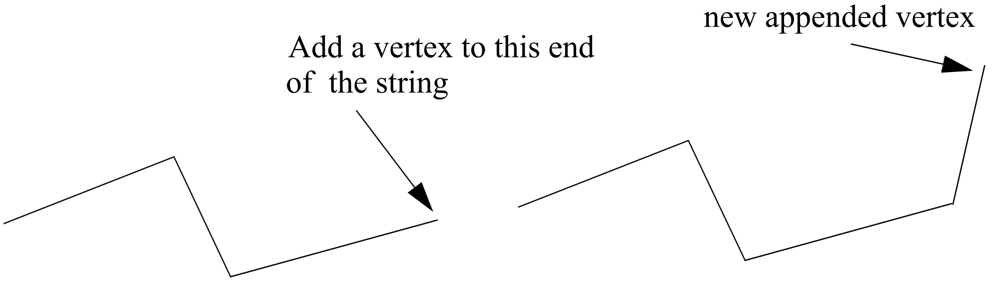
14.4.2.1 2d - Append

The **append** option is used to

- create the 1st vertex in a new string
- or , to append a new vertex to the end of the string
- or to prepend a new vertex to the beginning of the string.

In this option, both appending and prepending will be referred to as *appending.*

14.4.2.1.1 Existing 2d Strings



Appending a vertex is a two step process.

Step (a) - For an existing string - selecting the end of the string to append the vertex to.

After the **Append** option has been selected, the end of the string to append the vertex to is selected. Once the string end is selected, the new intersection point is assumed to be at the current cursor position. As the cursor is moved, the string is redrawn reflecting the changing position of the appended intersection point

message area 1 <Append>
Screen message area
<Pick end to append to> [picks][fast][menu]

Step (b) - selecting the position for the new appended vertex.

The position of the new appended vertex is set to the current cursor position by picking (LB) and accepting (MB).

message area 1 <Append>
Screen message area
<Pick location of vertex> [picks][fast][Menu]

Once a vertex has been appended to the string, the appended vertex is considered to be the selected string end and a new step (b) begins. The current cursor position indicates the new position for the next appended vertex.

Hence a series of string vertices is easily entered by first selecting the string end that the new vertices are to be appended to (step (a)) and then moving the cursor to the position of each new vertex and selecting them in turn (setup (b)).

Once the move cycle is completed and the vertex moved, the move option is still current and can be repeated for other vertices without having to re-select the **move** option.

The **Append** option is terminated by selecting **Cancel** from the **Pick Ops** menu or by selecting a new **Edit 2d** option.

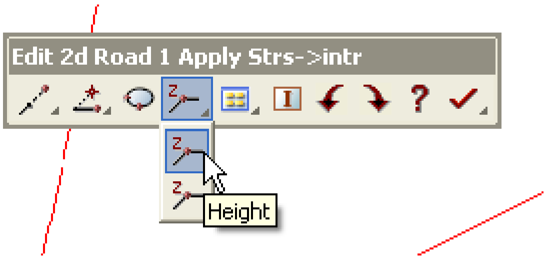
14.4.2.1.2 New 2d String

For creating a new string, the cursor is used to select the 1st vertex of the string. The option then continues as if appending to an existing 2d string where the end vertex has already been selected.

Typed input can be used in either step.

Continue to next section 14.4.2.2 2d - Height or return to 14.4.2 Edit 2d.

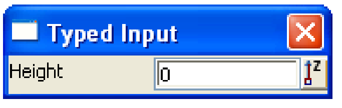
14.4.2.2 2d - Height



The **height** option is used to modify the height (z value) of the 2d string.

After the height option is chosen, a height typed input box is displayed on the screen with the string's current height (z value).

The height typed-input box looks like:



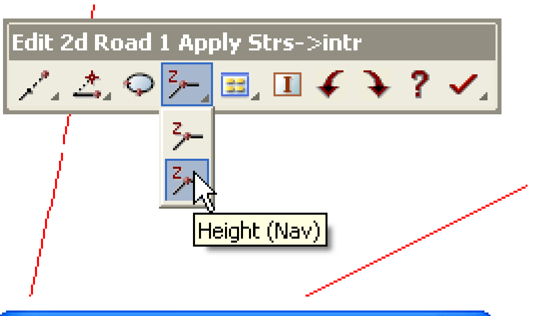
The height is entered into the typed-input box, terminated by <enter>. The entered value is taken as the height of the 2d string and the string redrawn with the new height. The typed-input box then disappears.

The height option automatically terminates and a new option is selected from the **Edit 2d**.

Note - all the points in a 2d string have the same height.

Continue to next section 14.4.2.3 2d - Height (Nav) or return to 14.4.2 Edit 2d.

14.4.2.3 2d - Height (Nav)



The **height (Nav)** option is used to modify the height (z value) of the 2d string.
After the **height (Nav)** option is chosen, a **Super Vertex Height panel** is displayed on the screen with the string's current height (z value).

Super Vertex Height panel

To change the height of the selected string, enter the value into the height field and press **ok** or **apply**. The entered value is then taken as the height of the 2d string and the string redrawn with the new height.

Note - all the points in a 2d string have the same height.

The fields and buttons used in the **Super Vertex Height** panel have the following functions.

Height input height of vertex/string

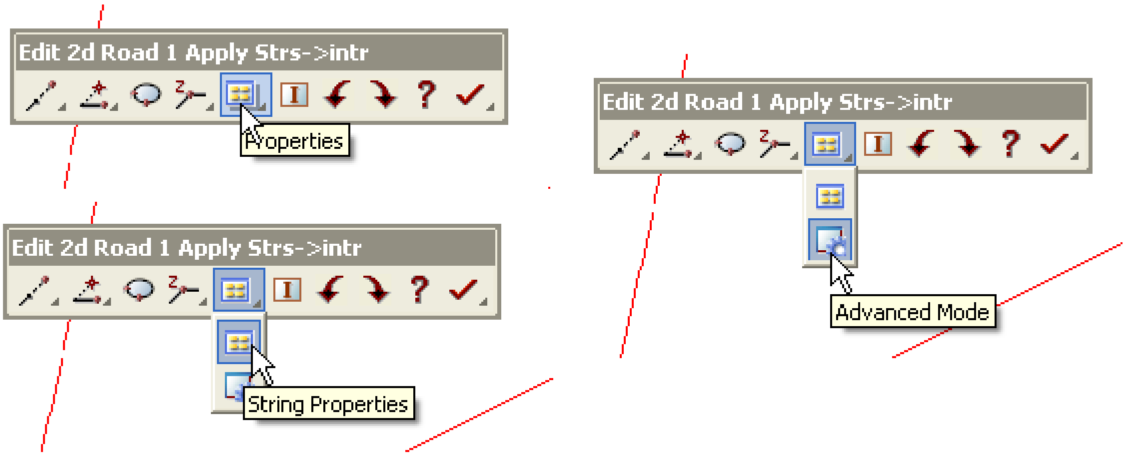
the height used for the entire string.

OK/Apply button

OK *sets the string with the value in the panel field and removes the panel.* **Apply** *set the string with the value in the panel field and leaves the panel on the screen.*

Continue to next section 14.4.2.4 2d Properties or return to 14.4.2 Edit 2d.

14.4.2.4 2d Properties

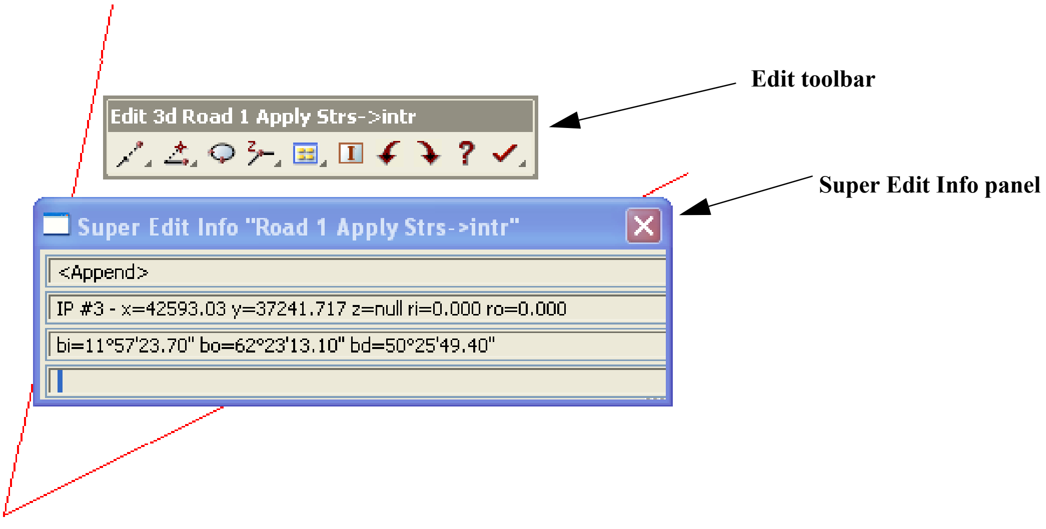


For information on the **Super String Properties** panel, go to 14.4.1.8 Properties

14.4.3 Edit 3d

Position of option on menu: **Strings =>Editor**

On picking a 3d string, the **Edit 3d** toolbar is placed on the screen. If **Display edit info** is ticked on in the **Default** panel (**Project =>Management =>Defaults**), an information panel, (the string's **Super Edit Info** panel) is also displayed.



Each option in the **Edit 3d** menu will now be described.

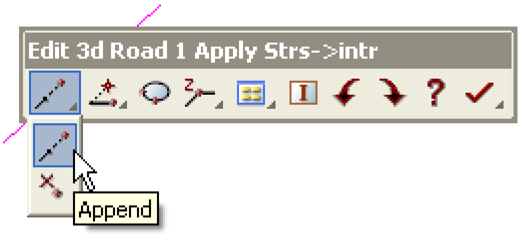
See 14.4.3.1 3d - Append
14.4.3.2 3d - Extend and Extend ht
14.4.3.3 3d - Height
14.4.3.4 3d - Height (Nav)
14.4.3.5 3d Properties

The difference between a 2d and a 3d string is that all points in a 2d string have the same z-value, whereas for a 3d string, each point can have a different z-value.

Hence most of the options in the **Edit 3d** menu are similar to the **Edit 2d** options of the same name, and only the differences for each option will be discussed. The **Edit 2d** options are given in the section 14.4.2 Edit 2d

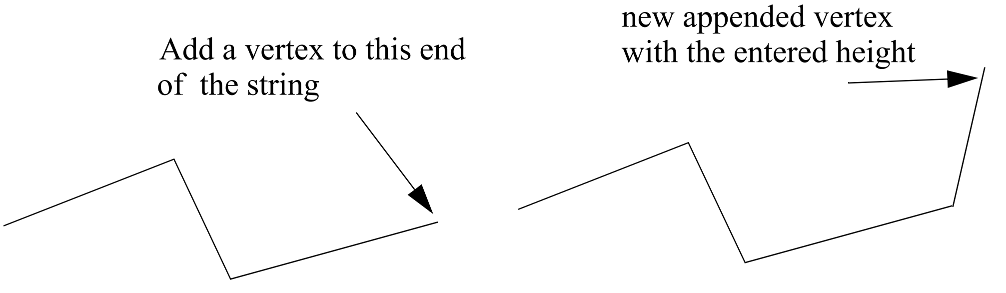
See the earlier section 14.4.1 Super String Edit - Common Informationfor general information about editing strings, the panel **Super Edit Info**, and the options, 14.4.1.2 Delete,14.4.1.4 Insert,14.4.1.3 Move,14.4.1.5 Between,14.4.1.6 Extend, 14.4.1.7 Open/Close, 14.4.1.8 Properties, 14.4.1.9 Info, 14.4.1.10.1 Undo, 14.4.1.10.2 Redo, 14.4.1.13 Quit, 14.4.1.11 '?' Help, 14.4.1.12 Finish and 14.4.1.14 Typed input.

14.4.3.1 3d - Append



The **Append** operation for a 3d string is similar to that for a 2d string except that a height (z-value) is required for each new vertex.

Since in most cases, it would be tiresome to ask for a height every time a vertex is added, the entry of a new height is controlled by the **height** toggle in the snaps menu. If **height** is toggled to on, then every time a vertex is placed or moved an enter height typed-input box is displayed on the screen.



The enter height typed-input box looks like:

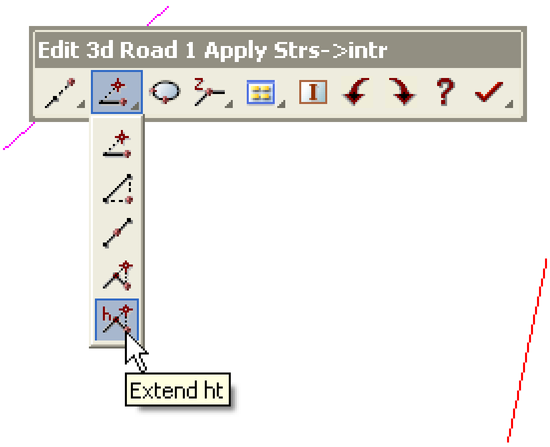
The height is entered into the typed-input box, terminated with <enter>. The entered value is taken as the height of the 3d string vertex and the string redrawn with the new height at the vertex. The typed-input box then disappears.

When the enter height box is placed on the screen, it will already have a value in it depending on the circumstances preceding the operation.

For example, if a vertex or segment was snapped to, the height at that vertex or segment will be displayed in the box.

Continue to next section 14.4.3.2 3d - Extend and Extend ht or return to 14.4.3 Edit 3d.

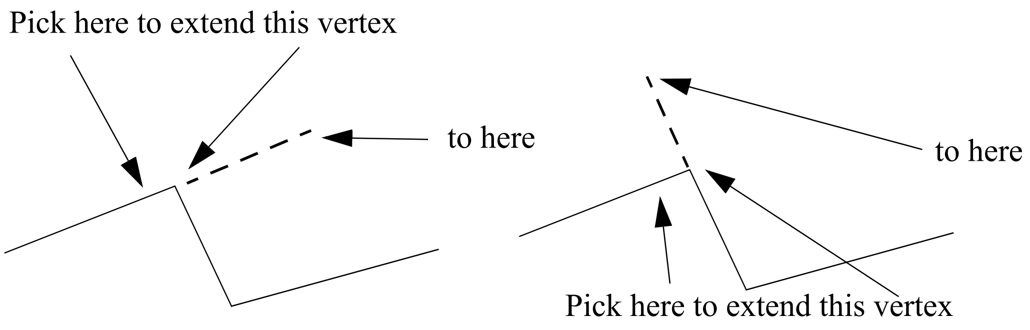
14.4.3.2 3d - Extend and Extend ht



The **Extend ht** option is used to move a vertex along the line joining the vertex to its neighbouring vertex plus the z-value is interpolated from the z-values at either end of the segment.

The **Extend ht** option is similar to the **Extend** option except that the z-value of the moved vertex is modified by linearly interpolating the z-value from the original vertices at the end of the selected segment.

Most vertices are the end vertex of two segments and the extension could be along either segment. The vertex and the segment to extend along are both selected at the same time by picking on the segment to extend along, near the vertex to extend.



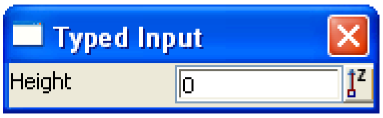
Continue to next section 14.4.3.3 3d - Height or return to 14.4.3 Edit 3d.

14.4.3.3 3d - Height

The **height** option is used to modify the height (z value) of any vertex in the string.

After the height option is chosen, the user must select which vertex is going to have its height modified.

After the vertex has been selected, a Height typed-input box is displayed on the screen with the vertex's current height (z value).



The height is entered into the typed-input box, terminated with <enter>. The entered value is taken as the height of the vertex in the 3d string and the string redrawn with the new height at that point. The typed-input box then disappears.

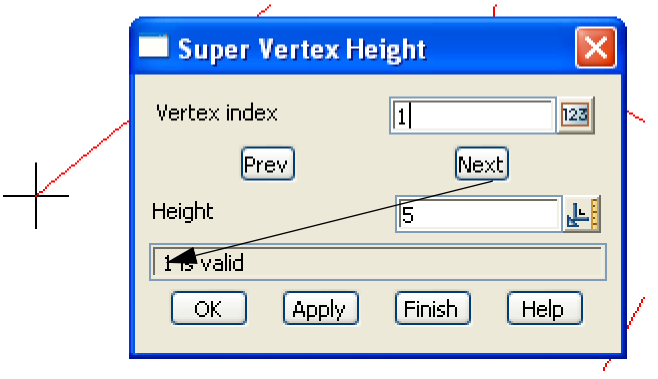
The **height** option is terminated on selecting **cancel** from the **pick ops** menu or by selecting a new option from the **Edit 3d** menu.

Continue to next section 14.4.3.4 3d - Height (Nav) or return to 14.4.3 Edit 3d.

14.4.3.4 3d - Height (Nav)

As soon as **Height (Nav)** is chosen, a **Super Vertex Height panel** is displayed on the screen with the string's current height (z value) in the Height field.

The **Super Vertex Height** panel is used to set the height value for vertices.



When the **Super Vertex Height** panel appears on the screen, the vertex index field is the vertex index of the selected vertex, with its associated height in the Height field of the panel. To select another vertex to alter, the vertex to modify is selected by typing the vertex index into the Vertex index field or by selecting **Prev** or **next** to switch to another vertex. The selected vertex is highlighted on the string and its *vertex index*, and *height* are written to the appropriate panel fields in the **Super Vertex Height** panel. The values can be changed and either **OK** or **Apply** the changed value to the vertex.

Another vertex can then be selected or the **Prev** and **Next** buttons used to move to adjacent vertices.

The fields and buttons used in the **Super Vertex Height** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Vertex index	input	selected vertex	

if a vertex is selected, then its vertex index is displayed in this field. An index can also be typed in and any information in the panel will then be applied to that vertex if **OK** *or* **Apply** *is selected.*

Prev	button		

move to the previous vertex (predecessor). The information for the previous vertex is displayed in the panel fields.

Next	button		

move to the next vertex (successor). The information for the next vertex is displayed in the panel fields.

Height	input	height of vertex/string	

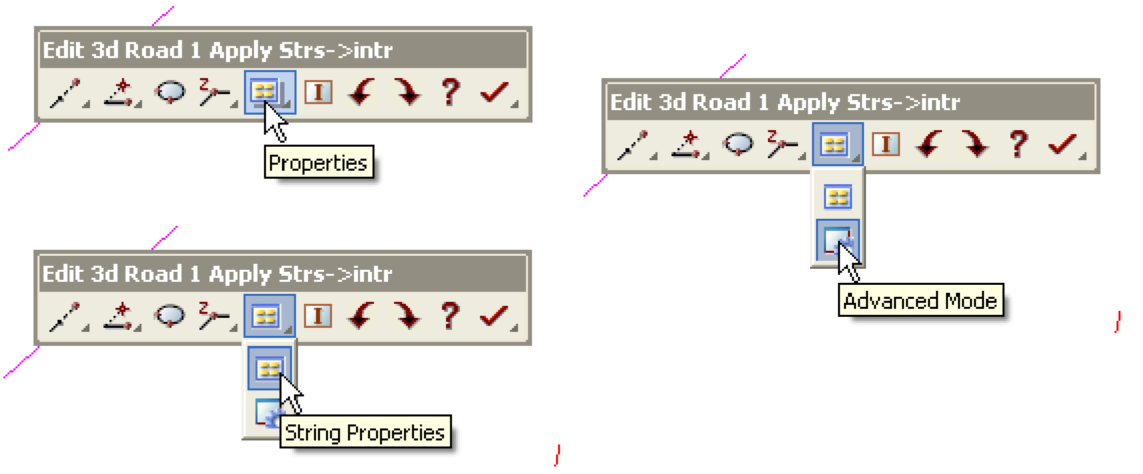
the height used for the vertex or for the entire string.

OK/Apply	button		

for the vertex being edited, **OK** *sets the vertex/string with the values in the panel fields and removes the panel.* **Apply** *sets the vertex/string with the values in the panel fields and leaves the panel on the screen.*

Continue to next section 14.4.3.5 3d Properties or return to 14.4.3 Edit 3d.

14.4.3.5 3d Properties



For information on the **Super String Properties** panel, go to 14.4.1.8 Properties.

14.4.4 Edit 4d

Position of option on menu: **Strings =>Editor**

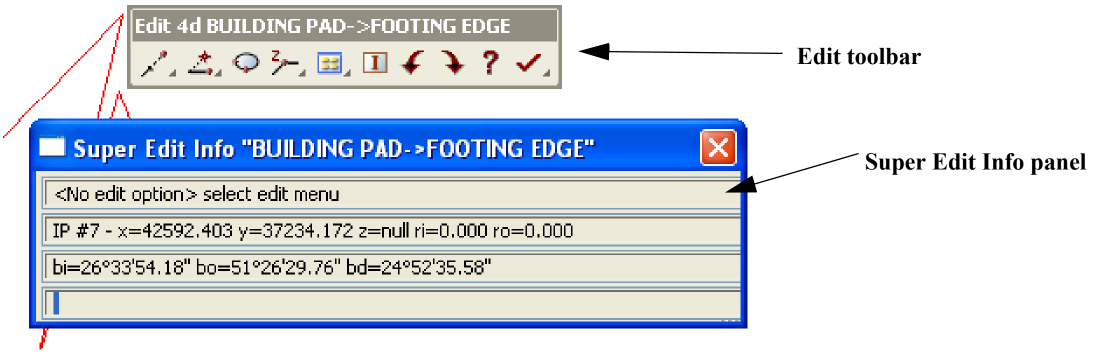
A *4d super string* is a *super string* with the restriction that it has:

(a) vertices of (x,y,z) co-ordinates joined by straight segments only

(b) Text at each vertex

(c) All text has the same height, rotation, *etc.* So it is controlled by the one Text Style for the entire string

On picking a 4d string, the **Edit 4d** toolbar is placed on the screen. If **Display edit info** is ticked on in the **Default** panel (**Project =>Management =>Defaults**), an information panel, (the string's **Super Edit Info** panel) is also displayed.



Only the options **append**, **Append** + **Text,** text, size and angle will be described in detail since all the other options are similar to the equivalent 3d string option (see the section 14.4.3 Edit 3d).

See the earlier section 14.4.1 Super String Edit - Common Information for general information about editing strings, the panel **Super Edit Info**, and the options, 14.4.1.2 Delete,14.4.1.4 Insert,14.4.1.3 Move,14.4.1.5 Between,14.4.1.6 Extend, 14.4.1.7 Open/Close, 14.4.1.8 Properties, 14.4.1.9 Info, 14.4.1.10.1 Undo, 14.4.1.10.2 Redo, 14.4.1.13 Quit, 14.4.1.11 '?' Help, 14.4.1.12 Finish and 14.4.1.14 Typed input.

See 14.4.4.1 4d - Append +Text

14.4.4.2 4d - Height

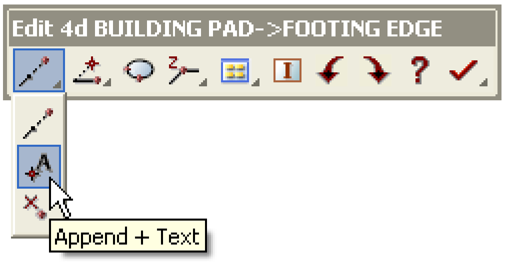
14.4.4.3 4d - Height (Nav)

14.4.4.4 4d - Text (Nav)

14.4.4.5 4d - Text Info

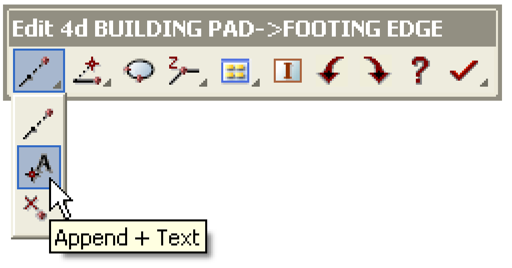
14.4.4.6 4d Properties

14.4.4.1 4d - Append +Text



The **append** operation for a 4d string is similar to the 3d string case except that a height and an optional text label is required for each new vertex.

Since in many cases a text label is not required at every vertex of a 4d string, there is an append option that asks for text at every vertex **(Append** + **Text)** and another that does not ask **(Append)**.



Append + **text** is a three step process.

Step (a) - For an existing string - selecting the end of the string to append the vertex and text to.

After the **Append** + **text** option has been selected, the end of the string to append the vertex to is selected. Once the string end is selected, the new intersection point is assumed to be at the current cursor position. As the cursor is moved, the string is redrawn reflecting the changing position of the appended intersection point

message area 1 <Append + Text>
Screen message area
<Pick end to append to> [picks][fast][menu]

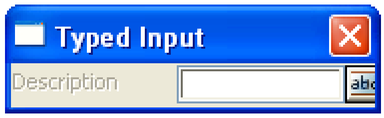
Step (b) - selecting the position for the new appended vertex.

The position of the new appended vertex is set to the current cursor position by picking (LB) and accepting (MB).

message area 1 <Append + Text>
Screen message area
<Pick location of vertex> [picks][fast][Menu]

Step (c) - Creating a text label for the vertex.

Once the new appended vertex is set to the current cursor position a Description typed-input box is displayed on the screen.



message area 1 <Append + Text>
Screen message area
<Super edit> [][][Menu]

Type the description into the typed-input box, finishing with <enter> and the typed-input box then disappears.

The appended vertex is now considered to be the selected string end and a new step (b) begins. The current cursor position indicates the new position for the next appended vertex.

Hence a series of string vertices is easily entered by first selecting the string end that the new vertices are to be appended to (step (a)) and then moving the cursor to the position of each new vertex and selecting them in turn (setup (b)).

The **append** option is terminated on selecting **cancel** from the **pick ops** menu or by selecting a new option from the **Edit 4d** menu.

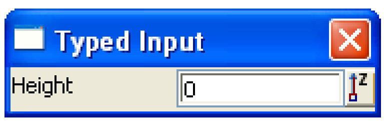
Continue to next section 14.4.4.2 4d - Height or return to 14.4.4 Edit 4d.

14.4.4.2 4d - Height

The **height** option is used to modify the height (z value) of any vertex in the string.

After the height option is chosen, the user selects which vertex is going to have its height modified.

After the vertex has been selected, a Height typed-input box is displayed on the screen with the vertex current height (z value).



The height is entered into the typed-input box, terminated with <enter>. The entered value is taken as the height of the vertex in the 4d string and the string redrawn with the new height at that vertex. The typed-input box then disappears.

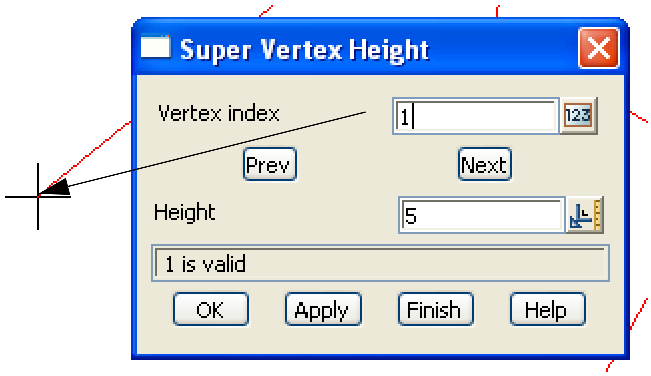
Another vertex can then be selected for modifying its height.

The **height** option is terminated on selecting **cancel** from the **pick ops** menu or by selecting a new option from the **Edit 4d** menu.

Continue to next section 14.4.4.3 4d - Height (Nav) or return to 14.4.4 Edit 4d.

14.4.4.3 4d - Height (Nav)

Selecting **Height (Nav)** brings up the **Super Vertex Height** panel with the height at the first vertex of the string displayed.
The **Super Vertex Height** panel is used to set the height value for any vertex of the string.



Any vertex can then be selected by clicking on the number icon at the end of the Vertex index panel field and selecting the required vertex, or by typing in its Vertex index and pressing <Enter>, or by using the **Prev** and **Next** buttons used to move to adjacent vertices until the required vertex is selected.

When the vertex to modify is selected, the vertex is highlighted on the string and its *vertex index*, and *height* are displayed in the appropriate panel fields in the **Super Vertex Height** panel. The new vertex height is stored by clicking either **OK** or **Apply**.

The fields and buttons used in the **Super Vertex Height** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Vertex index	input	selected vertex	

if a vertex is selected, then its vertex index is displayed in this field. An index can also be typed in and any information in the panel will then be applied to that vertex if **OK** *or* **Apply** *is selected.*

Prev	button		

move to the previous vertex (predecessor). The information for the previous vertex is displayed in the panel fields.

Next	button		

move to the next vertex (successor). The information for the next vertex is displayed in the panel fields.

Height	input	height of vertex/string	

the height used for the selected vertex.

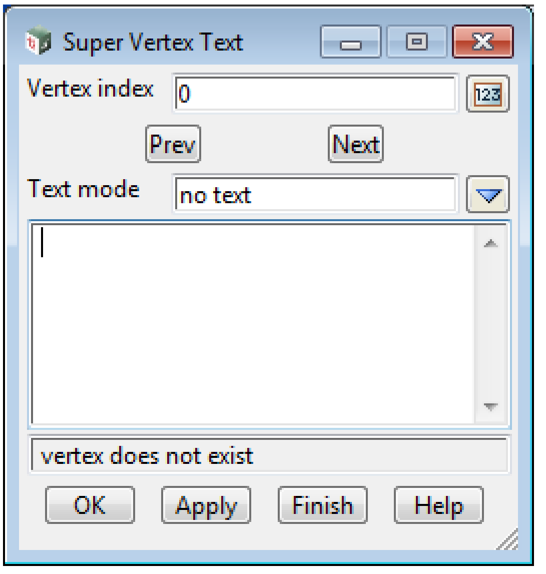
OK/Apply	button		

for the vertex being edited, **OK** *sets the vertex/string with the values in the panel fields and removes the panel.* **Apply** *sets the vertex/string with the values in the panel fields and leaves the panel on the screen.*

Continue to next section 14.4.4.4 4d - Text (Nav) or return to 14.4.4 Edit 4d.

14.4.4.4 4d - Text (Nav)

Selecting **Text (Nav)** brings up the **Super Vertex Text** panel with the text at the first vertex of the string displayed.
The **Super Vertex Text** panel is used to set the text for any vertex of the string.



Any vertex can then be selected by clicking on the number icon at the end of the Vertex index panel field and selecting the required vertex, or by typing in its Vertex index and pressing <Enter>, or by using the **Prev** and **Next** buttons used to move to adjacent vertices until the required vertex is selected.

When the vertex to modify is selected, the vertex is highlighted on the string and its *vertex index* and vertex *text* are displayed in the appropriate panel fields in the **Super Vertex Text** panel. The new vertex text is stored by clicking either **OK** or **Apply**.

The fields and buttons used in the **Super Vertex Text** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Vertex index	input	selected vertex	

if a vertex is selected, then its vertex index is displayed in this field. A number can also be typed in and any information in the panel will then be applied to that vertex if **OK** *or* **Apply** *is selected.*

Prev button

move to the previous vertex (predecessor). The information for the previous vertex is displayed in the panel fields.

Next button

move to the next vertex (successor). The information for the next vertex is displayed in the panel fields.

Text mode choice box no text, entire string, each vertex

*if **no text**, there is no text for the vertex.*
*if **entire string**, then the string has the same text for each vertex.*
*if **each vertex**, then each vertex has a separate text value.*

Text input text of vertex/string

the text used for the vertex.

OK/Apply button

for the vertex being edited, **OK** *sets the vertex/string with the values in the panel fields and removes the panel.* **Apply** *sets the vertex/string with the values in the panel fields and leaves the panel on the screen.*

Continue to next section 14.4.4.5 4d - Text Info or return to 14.4.4 Edit 4d.

14.4.4.5 4d - Text Info

For a 4d super string, the size, colour, slant, offset, *etc.* of vertex text is all the same and is controlled by the *Text Style* for the 4d super string.

Selecting **Text Info** brings up the **Super Vertex Annotate** panel with the text style for the string displayed.

The fields and buttons used in the **Super Vertex Annotate** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Text style	text style box	string's text style	available text styles

the text style string. This can be modified to change the values for all the text on the 4d string.

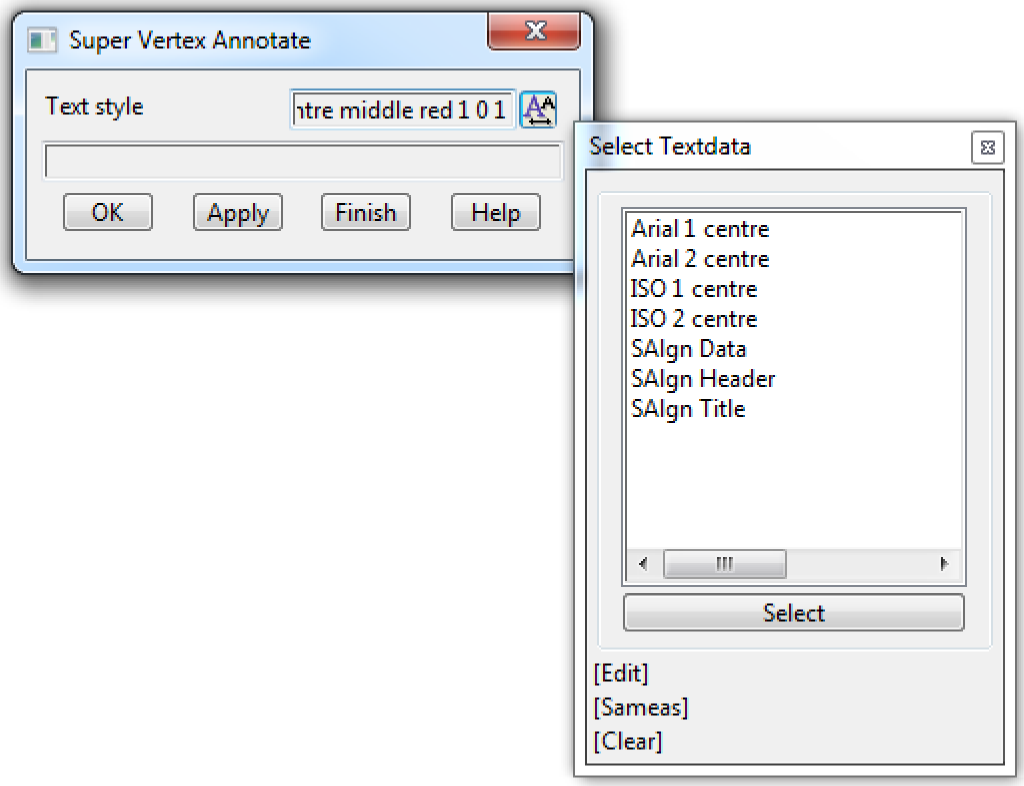
Prev	button		

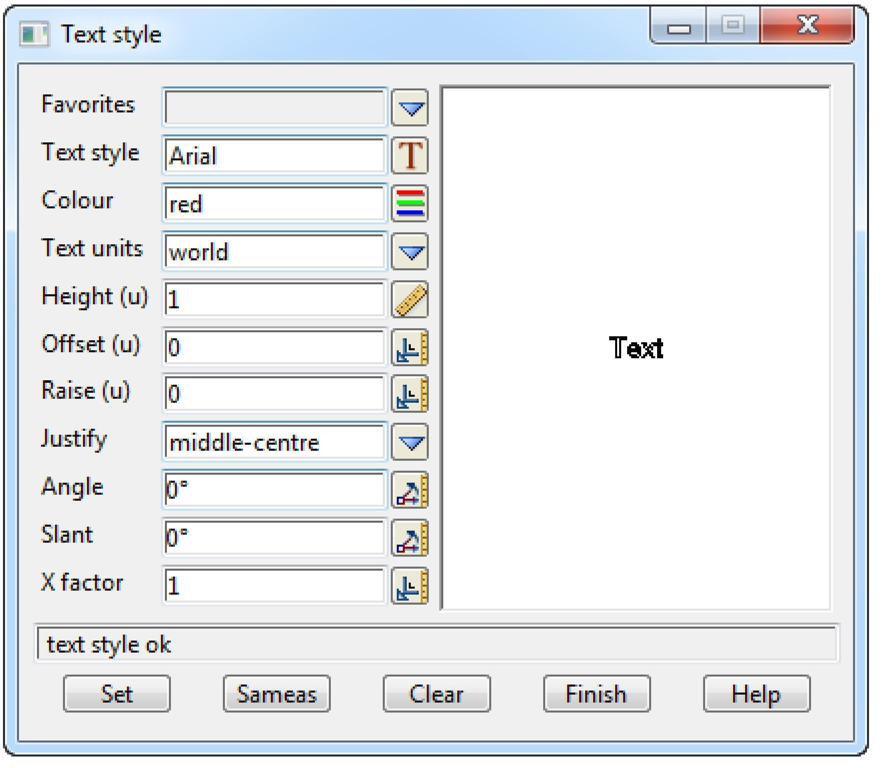
move to the previous vertex (predecessor). The information for the previous vertex is displayed in the panel fields.

OK/Apply	button		

for the string being edited, **OK** *sets the text style for the string and removes the panel.* **Apply** *sets the text style for the string and leaves the panel on the screen.*

To modify any of the values of the ***Text Style***, click on the ***Text style*** icon at the end of the Text style panel field to bring up the ***Select Textdata*** panel, and then select ***[Edit].***





To change any of the values in the **Text Style** panel:

1. Modify the value in the Text Style panel and then click on **Set** and **Finish**. The panel will then close.
2. Click on **Apply** or **OK** in the **Super Vertex Annotate** panel.

Modifying the Text Height

All the text labels in the 4d string have the same height.

The height of all the text is modified by changing the Height (u) in the *Text Style Info* for the string (see 14.4.4.5 4d - Text Info

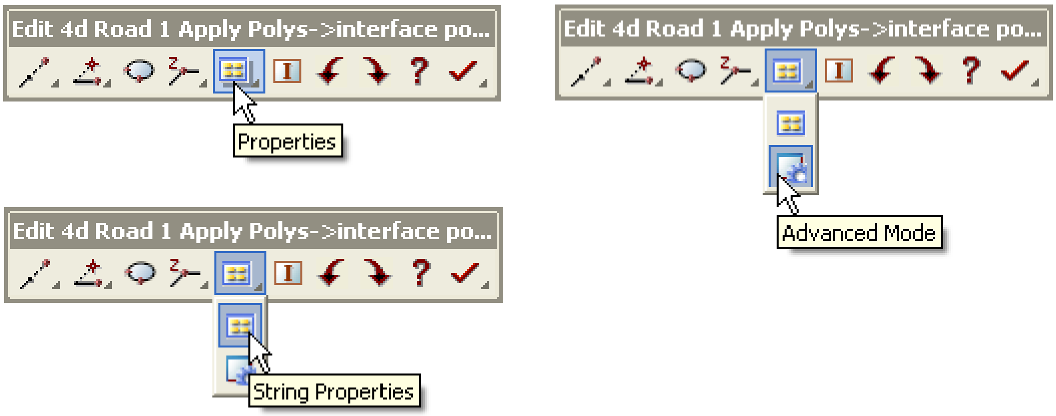
Modifying the Text Angle

All the text labels in the 4d string are drawn rotated about their defining string point with the *same* rotation angle. The angle, in degrees, is measured in a counter-clockwise direction about the horizontal axis.

The angle of all the text is modified by changing the Angle in the *Text Style Info* for the string (see 14.4.4.5 4d - Text Info).

Continue to next section 14.4.4.6 4d Properties or return to 14.4.4 Edit 4d.

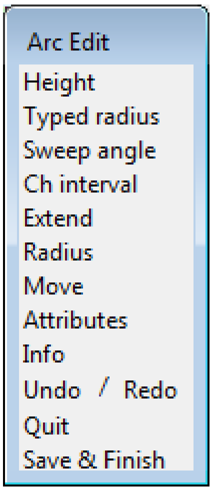
14.4.4.6 4d Properties

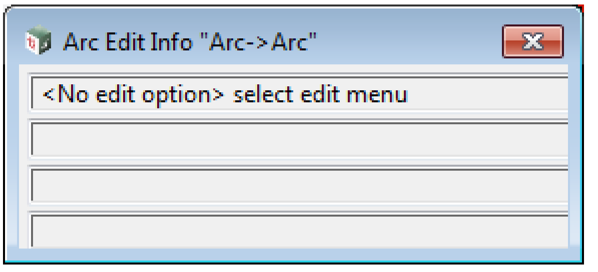


For information on the **Super String Properties** panel, go to 14.4.1.8 Properties

14.4.5 Arc Edit

On picking an Arc string, the **Arc Edit** menu and **Arc Edit Info** panel are placed on the screen.





Each option in the **arc edit** menu will now be described.

See the earlier section 14.4.1 Super String Edit - Common Information for general information about editing strings.

14.4.5.1 Height

The **Height** option is used to define the z-value at the start or end point of the arc.

The z-value at any point on the arc is an interpolation on arc length of the values at the end points. Hence, the arc is an arc in plan only - in three dimensions, it is a helix.

Height is a two step process.

Step (a) - selecting the start or end point of the arc

After picking **height**, the end point of the arc whose height is to be modified is selected.

message area 1 <Height of arc>

Screen message area

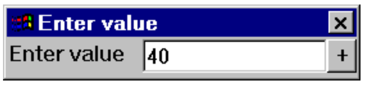
<Select point to change height> [picks][][menu]

<Select point to change height> [picks][accepts][menu]

Step (b) - entering the new height

After the start or end point of the arc is selected, an enter value typed-input box is placed on the screen with the points current height displayed in it.

The **enter value** typed-input box looks like:



The new height for the point is entered into the typed-input box, terminated with <enter>. The typed-input box then disappears.

The **height** option repeats until cancelled by the user by either bringing up the **pick ops** menu and selecting **cancel** or by selecting a new option from the **arc edit** menu.

If, after bringing up the **pick ops** menu, it is decided to continue with the height option, simply select the **restart** option from the **pick ops** menu and the **pick ops** menu will disappear leaving the **height** option still current.

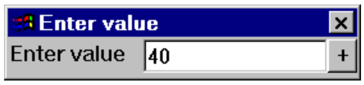
Continue to next section 14.4.5.2 Typed Radius or return to 14.4.5 Arc Edit.

14.4.5.2 Typed Radius

The **Typed radius** option is used to modify the radius of the selected arc.

After the **typed radius** option is chosen, an enter value typed-input box is placed on the screen with the arcs current radius displayed in it.

The **enter value** typed-input box looks like:



The arc radius is entered into the typed-input box, terminated with <enter>. The entered value is taken as the radius of the arc and the arc is redrawn with its new radius.

The typed-input box then disappears.

The **typed radius** option is automatically terminated and a new option needs to be selected from the **arc edit** menu.

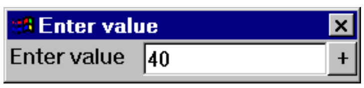
Continue to next section 14.4.5.3 Sweep Angle or return to 14.4.5 Arc Edit.

14.4.5.3 Sweep Angle

The **Sweep angle** option is used to modify the sweep angle of the selected arc.

After the **sweep angle** option is chosen, an enter value typed-input box is placed on the screen with the arcs current sweep angle displayed in it.

The **enter value** typed-input box looks like:



The arc sweep angle is entered into the typed-input box, terminated with <enter>. The entered value is taken as the sweep angle of the arc and the arc is redrawn with its new sweep angle.

The typed-input box then disappears.

The **sweep angle** option is automatically terminated and a new option needs to be selected from the **arc edit** menu.

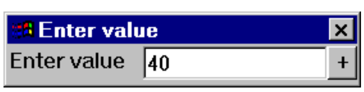
Continue to next section 14.4.5.4 Chainage Interval or return to 14.4.5 Arc Edit.

14.4.5.4 Chainage Interval

The **ch interval** option is used to modify the chainage interval of the selected arc.

After the **ch interval** option is chosen, an enter value typed-input box is placed on the screen with the arcs current chainage interval displayed in it.

The **enter value** typed-input box looks like:



The new chainage interval is entered into the typed-input box, terminated with <enter>. The entered value is taken as the chainage interval of the arc and the arc is redrawn with its new chainage interval.

The typed-input box then disappears.

The **ch interval** option is automatically terminated and a new option needs to be selected from the **arc edit** menu.

Continue to next section 14.4.5.5 Move End or return to 14.4.5 Arc Edit.

14.4.5.5 Move End

The **move end** option is used to re-position either the start or the end point of the arc.

Move end is a two step process.

Step (a) - selecting the start or end point of the arc

After selecting the **move end** option, the end point to be move is selected.

message area 1 <Move End>

Screen message area

<Select point to move> [picks][][menu]
<Select point to move> [picks][accepts][menu]

Step (b) - selecting the new position for the end point

After selecting the end point to move, the new position for the end point is selected.

message area 1 <Move End>

Screen message area

<Select final position of point> [picks][][menu]
<Select final position of point> [picks][accepts][menu]

The **move end** option repeats until cancelled by the user by either bringing up the **pick ops** menu and selecting **cancel** or by selecting a new option from the **arc edit** menu.

If, after bringing up the **pick ops** menu, it is decided to continue with the **move end** option, simply select the **restart** option from the **pick ops** menu and the **pick ops** menu will disappear leaving the **move end** option still current.

Continue to next section 14.4.5.6 Radius or return to 14.4.5 Arc Edit.

14.4.5.6 Radius

The **radius** option is used to change the radius of the arc by leaving the centre point alone and re-positioning a point on the arc. The new position for the point will define a new radius for the arc.

Radius is a two step process.

Step (a) - selecting the point on the arc to move

After selecting the **radius** option, the point on the arc to be moved is selected.

message area 1 <Radius>

Screen message area

<Select point to move> [picks][][menu]
<Select point to move> [picks][accepts][menu]

Step (b) - selecting the new position for the point

After selecting the point to move, the new position for the point is selected.

message area 1 <Radius>

Screen message area

<Select final position of point> [picks][][menu]
<Select final position of point> [picks][accepts][menu]

The **radius** option repeats until cancelled by the user by either bringing up the **pick ops** menu and selecting **cancel** or by selecting a new option from the **arc edit** menu.

If, after bringing up the **pick ops** menu, it is decided to continue with the **radius** option, simply select the **restart** option from the **pick ops** menu and the **pick ops** menu will disappear leaving the **radius** option still current.

Continue to next section 14.4.5.7 Move or return to 14.4.5 Arc Edit.

14.4.5.7 Move

The **Move** option is used to change the position of the entire arc by selecting and moving the arc centre or any point on the arc.

Move is a two step process.

Step (a) - selecting the point on the arc to move

After selecting the **move** option, the point on the arc to move is selected.

message area 1 <Move>

Screen message area

<Select point to move> [picks][][menu]

<Select point to move> [picks][accepts][menu]

Step (b) - selecting the new position for the point

After selecting the point to move, the new position for the point is selected.

message area 1 <Move>

Screen message area

<Select final position of point> [picks][][menu]

<Select final position of point> [picks][accepts][menu]

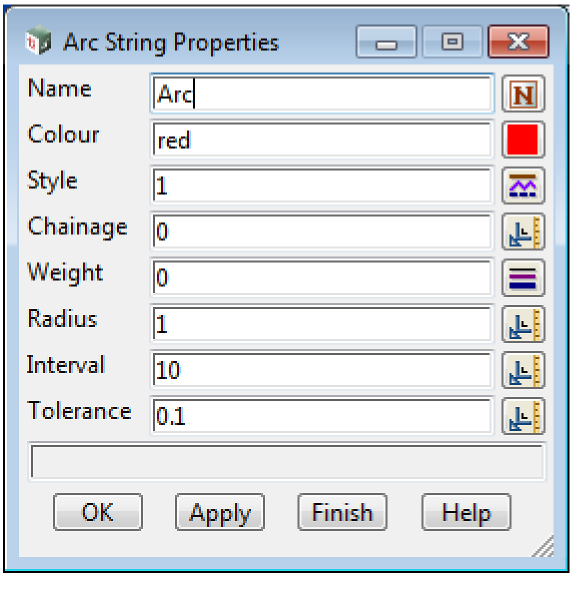
The **Move** option repeats until cancelled by the user by either bringing up the **Pick ops** menu and selecting **Cancel** or by selecting a new option from the **Arc edit** menu.

If, after bringing up the **Pick ops** menu, it is decided to continue with the **move** option, simply select the **restart** option from the **Pick ops** menu and the **Pick ops** menu will disappear leaving the **move** option still current.

Continue to next section 14.4.5.8 Properties or return to 14.4.5 Arc Edit.

14.4.5.8 Properties

Selecting **Properties** brings up the **Arc String Properties** panel which is used to modify the string's header information.



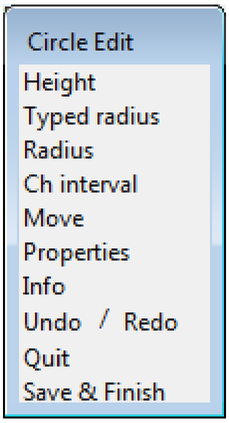
The fields in this panel are similar to those in the **create arc string** panel and the arc editor options. The only new field is

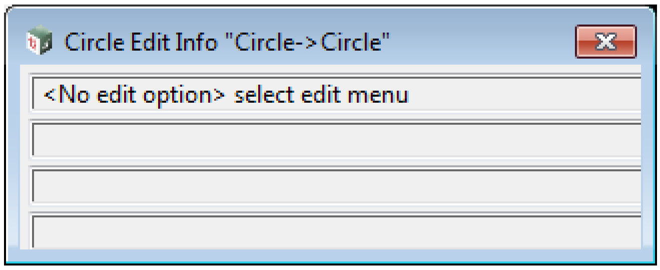
Field Description	Type	Defaults	Pop-Up
OK/Apply	button		

for the string being edited, **OK** *sets the string with the values in the panel fields and removes the panel.* **Apply** *sets the string with the values in the panel fields and leaves the panel on the screen.*

14.4.6 Circle Edit

On picking a circle string, the **circle edit** menu and **circle edit info** panel are placed on the screen.





Each option in the **circle edit** menu will now be described.

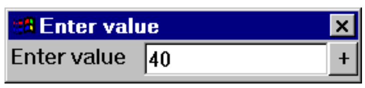
See the earlier section 14.4.1 Super String Edit - Common Information for general information about editing strings.

14.4.6.1 Height

The **height** option is used to define the z-value for the entire circle.

After selecting the **height** option, an enter value typed-input box is placed on the screen with the circles current height displayed in it.

The enter value typed-input box looks like:



The new height for the circle is entered into the typed-input box, terminated with <enter>.

The typed-input box then disappears.

The **height** option is automatically terminated and a new option needs to be selected from the **circle edit** menu.

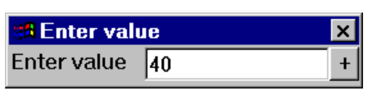
Continue to next section 14.4.6.2 Typed Radius or return to 14.4.6 Circle Edit.

14.4.6.2 Typed Radius

The **typed radius** option is used to modify the radius of the selected circle.

After the **typed radius** option is chosen, an enter value typed-input box is placed on the screen with the circles current radius displayed in it.

The enter value typed-input box looks like:



The circle radius is entered into the typed-input box, terminated with <enter>. The entered value is taken as the radius of the circle and the circle redrawn with its new radius.

The typed-input box then disappears.

The **typed radius** option is automatically terminated and a new option needs to be selected from the **circle edit** menu.

Continue to next section 14.4.6.3 Radius or return to 14.4.6 Circle Edit.

14.4.6.3 Radius

The **radius** option is used to change the radius of the circle by leaving the centre point alone and re-positioning a point on the circle. The new position for the point will define a new radius for the circle.

Radius is a two step process.

Step (a) - selecting the point on the circle to move

After selecting the **radius** option, the point on the circle to be moved is selected.

message area 1 <Radius>
Screen message area
<Select point to move> [picks][][menu]
<Select point to move> [picks][accepts][menu]

Step (b) - selecting the new position for the point

After selecting the point to move, the new position for the point is selected.

message area 1 <Radius>
Screen message area
<Select final position of point> [picks][][menu]
<Select final position of point> [picks][accepts][menu]

The **radius** option repeats until cancelled by the user by either bringing up the **pick ops** menu and selecting **cancel** or by selecting a new option from the **circle edit** menu.

If, after bringing up the **pick ops** menu, it is decided to continue with the **radius** option, simply select the **restart** option from the **pick ops** menu and the **pick ops** menu will disappear leaving the **radius** option still current.

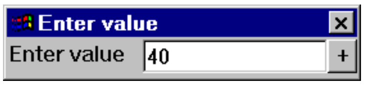
Continue to next section 14.4.6.4 Chainage Interval or return to 14.4.6 Circle Edit.

14.4.6.4 Chainage Interval

The **ch interval** option is used to modify the chainage interval of the selected circle.

After the **ch interval** option is chosen, an enter value typed-input box is placed on the screen with the circles current chainage interval displayed in it.

The enter value typed-input box looks like:



The new chainage interval is entered into the typed-input box, terminated with <enter>. The entered value is taken as the chainage interval of the circle.

The typed-input box then disappears.

The **ch interval** option is automatically terminated and a new option needs to be selected from the **circle edit** menu.

Continue to next section 14.4.6.5 Move or return to 14.4.6 Circle Edit.

14.4.6.5 Move

The **move** option is used to change the position of the entire circle by selecting and moving the circle centre or any point on the circle.

Move is a two step process.

Step (a) - selecting the point on the circle to move

After selecting the **move** option, the point on the circle to move is selected.

message area <Move>
Screen message area
<Select point to move> [picks][][menu]
<Select point to move> [picks][accepts][menu]

Step (b) - selecting the new position for the point

After selecting the point to move, the new position for the point is selected.

message area 1 <Move>
Screen message area
<Select final position of point> [picks][][menu]
<Select final position of point> [picks][accepts][menu]

The **move** option repeats until cancelled by the user by either bringing up the **pick ops** menu and selecting **cancel** or by selecting a new option from the **circle edit** menu.

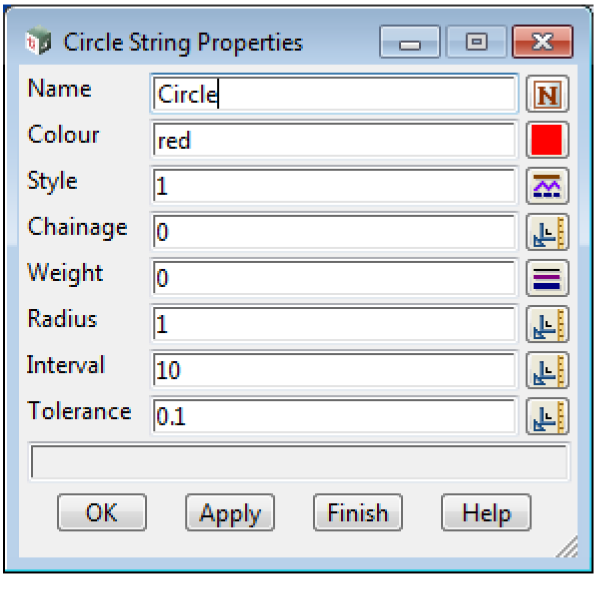
If, after bringing up the **pick ops** menu, it is decided to continue with the **move** option, simply select the **restart** option from the **pick ops** menu and the **pick ops** menu will disappear leaving the **move** option still current.

Note - the centre point of the circle can be selected and moved with the **move** option.

Continue to next section 14.4.6.6 Properties or return to 14.4.6 Circle Edit.

14.4.6.6 Properties

Selecting **Properties** brings up the **Circle String Properties** panel which is used to modify the string's header information.



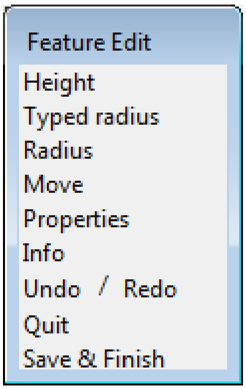
The fields in this panel are similar to those in the **create circle string** panel and the circle editor options. The only new field is

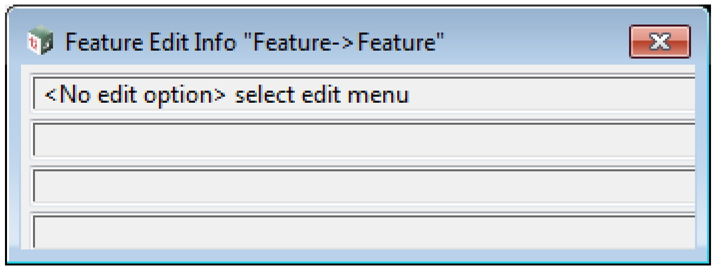
Field Description	Type	Defaults	Pop-Up
Set	button		

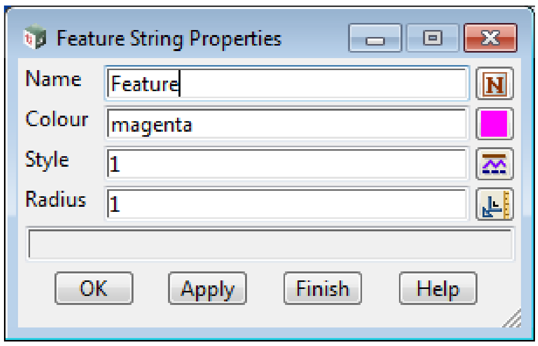
for the circle being edited, set all the items in the **circle string Properties** *panel to the values given in the panel.*

14.4.7 Feature Edit

On picking a feature string, the **Feature Edit** menu and panel are placed on the screen.







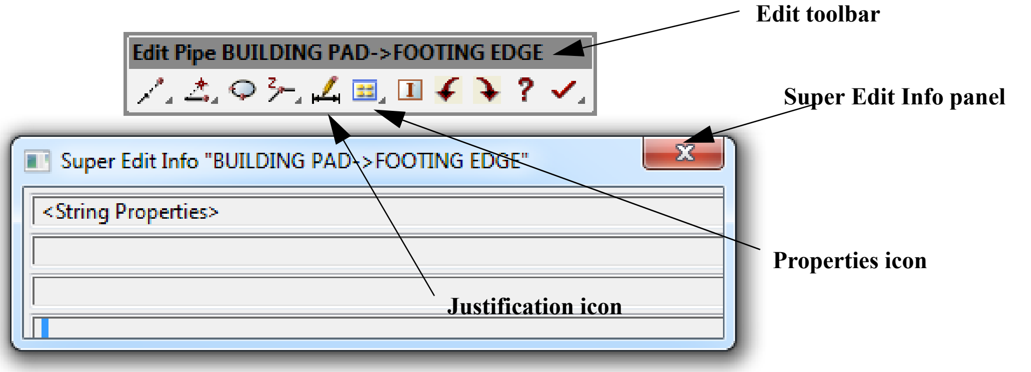
Each option in the **Feature Edit** menu is identical to those from the **circle edit** menu and will not be described again. See the section 14.4.6 Circle Edit

See the earlier section 14.4.1 Super String Edit - Common Information for general information about editing strings.

14.4.8 Edit Pipe

Position of option on menu: **Strings =>Editor**

On picking a pipe string, the **Edit Pipe** toolbar is placed on the screen. If **Display edit info** is ticked on in the **Default** panel (**Project =>Management =>Defaults**), an information panel, (the string's **Super Edit Info** panel) is also displayed.



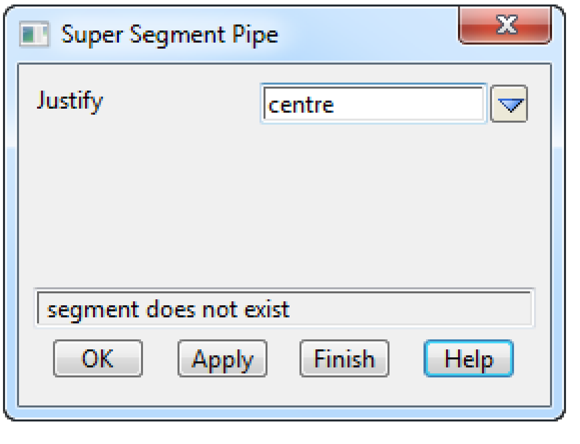
The only difference between a 3d and a pipe string is that the pipe string has a diameter.

Hence most of the options in the **pipe edit** toolbar are similar to the Edit **3d** options of the same name, and only the **Properties** option which includes **diameter** need to be discussed. See the section 14.4.3 Edit 3d for information on the Edit 3d editor.

See the earlier section 14.4.1 Super String Edit - Common Information for general information about editing strings.

14.4.8.1 Pipe Justification

Selecting the **Pipe** icon brings up the **Super Segment Pipe** panel which is used to modify the string's ***pipe justification***.



Justify choice box invert, centre, obvert

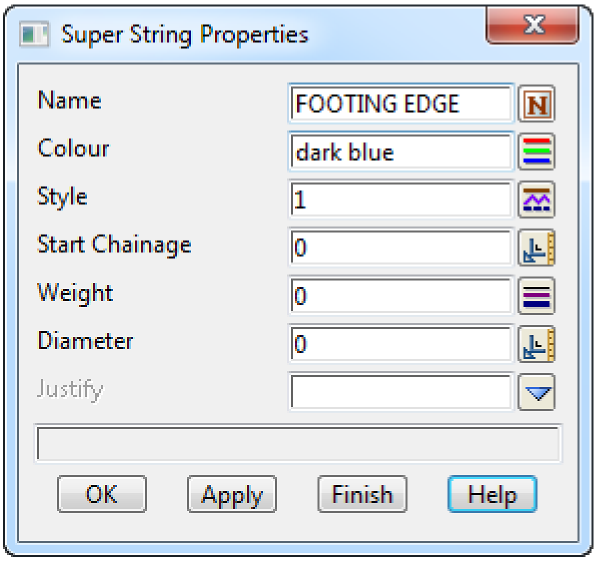
*The **Justify** field is used to modify the justification of the pipe string.*

***Note** - the default pipe string has only one justification for the entire string, however by using the Advanced mode, the pipe string can be turned into a more general super string where each segment can have its own justification.*

Continue to next section 14.4.8.2 Super Pipe String Properties or return to 14.4.8 Edit Pipe.

14.4.8.2 Super Pipe String Properties

Selecting the **Properties** icon brings up the **Pipe String Properties** panel which is used to modify the string's header information, the ***pipe diameter*** and the ***pipe justification***.



The fields in this panel are similar to those in the **create pipe string** panel and the 3d string editor options. The only new field is

Field Description	Type	Defaults	Pop-Up
Diameter	real box		

The **diameter** *field is used to modify the diameter of the pipe string.*

Note *- the default pipe string has only one diameter for the entire string, however by using the Advanced mode, the pipe string can be turned into a more general super string where each segment can have its own diameter, or can be a box culvert rather than a pipe.*

Justify	choice box		invert, centre, obvert

The **Justify** *field is used to modify the justification of the pipe string.*

Note *- the default pipe string has only one justification for the entire string, however by using the Advanced mode, the pipe string can be turned into a more general super string where each segment can have its own justification.*

OK/Apply	button		

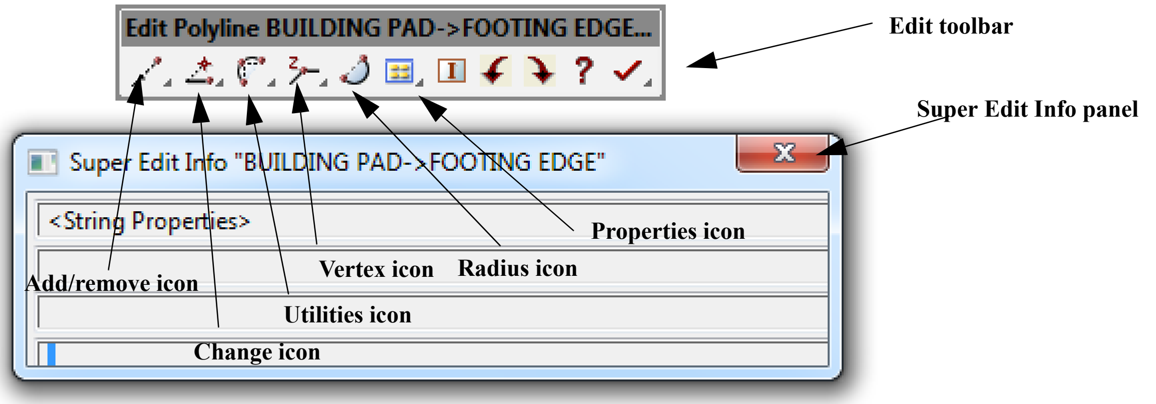
for the string being edited, **OK** *sets the string with the values in the panel fields and removes the panel.* **Apply** *sets the string with the values in the panel fields and leaves the panel on the screen.*

14.4.9 Edit Polyline

Position of option on menu: **Strings =>Editor**

On picking a Polyline string, the **Edit Polyline** toolbar is placed on the screen. If **Display edit info** is ticked on in the **Default** panel (**Project =>Management =>Defaults**), an information panel, (the string's **Super Edit Info** panel) is also displayed.

On picking a polyline string, the **Polyline edit** menu and **Polyline edit info** panel are placed on the screen.



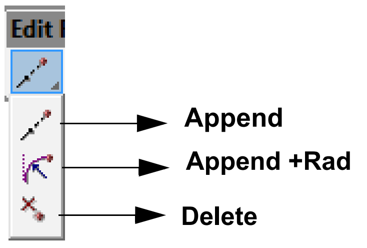
The major difference between a *polyline* and a 3d string is that a polyline string can have an arc instead of a line joining adjacent string vertices. Hence most of the options in the **Edit Polyline** toolbar are similar to the **Edit 3d** options of the same name, and only the differences for each option will be discussed. See the section 14.4.3 Edit 3d for information on the 3d string editor.

See the earlier section 14.4.1 Super String Edit - Common Information for general information about editing strings.

For the option *Add/ Remove Toolbar* go to 14.4.9.1 Add/Remove Toolbar
Change Toolbar 14.4.9.2 Change Toolbar
Utilities Toolbar 14.4.9.3 Utilities Toolbar
Vertex Toolbar
Radius Toolbar 14.4.9.4 Radius
Properties Toolbar 14.4.9.5 Properties toolbar
Info Toolbar 14.4.1 Super String Edit - Common Information
Undo Toolbar 14.4.1 Super String Edit - Common Information
Redo Toolbar 14.4.1 Super String Edit - Common Information
Help Toolbar 14.4.1 Super String Edit - Common Information
Finish Toolbar 14.4.1 Super String Edit - Common Information

14.4.9.1 Add/Remove Toolbar

The **Add/remove** toolbar is



The **Append/Prepend** options allow for simply adding vertices, or adding vertices plus radii for the segments.

Append

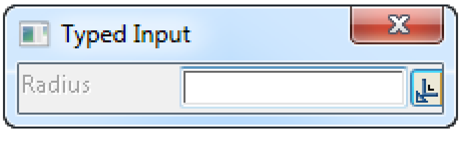
For the cases when no arc is required, the Append option simply places vertices.

Append + Radius

If an arc radius is required for every segment of the super string, there is an Append + Radius option that asks for the radius for every segment.

If the **Append** + **Radius** option is selected, then before each vertex is appended, a Radius typed-input box is placed on the screen.

The **Radius** typed-input box looks like



The radius is entered into the typed-input box, terminated with <enter>. The entered value is taken as the radius of the arc to the next vertex and the arc will be drawn correctly as the cursor is moved to the next vertex.

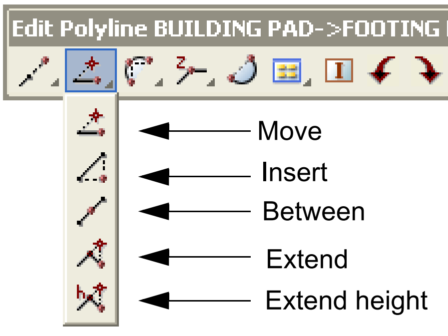
A radius value of 0 is taken to mean no arc.

The height question for each point is toggled on/off in the snaps menu just as it was for a 3d string.

The **Append** options are terminated by selecting **Cancel** from the **Pick Ops** menu or by selecting a new option from the **Edit polyline** toolbar

Go to the next section 14.4.9.2 Change Toolbar

14.4.9.2 Change Toolbar

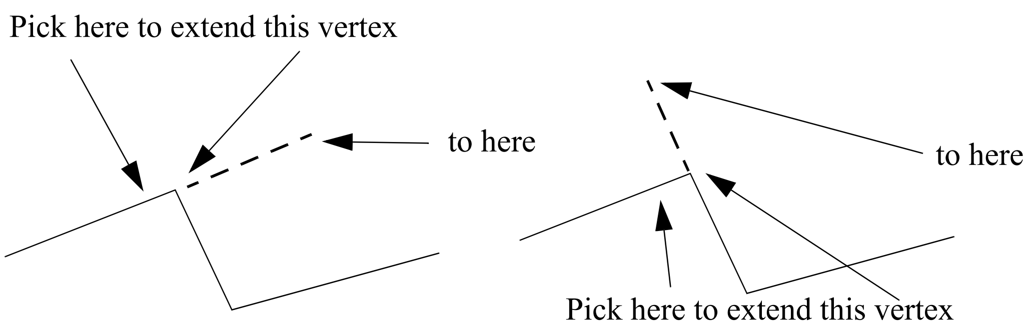


The **Extend ht** option is used to move a vertex along the line joining the vertex to its neighbouring vertex plus the z-value is interpolated from the z-values at either end of the segment.

The Move, Insert and Between options are the same as for a 3d string.

There are two **Extend** options for a polyline string - **Extend** that is identical to the 3d string case where the z-value of the point being extended is kept constant, and a second option, **Extend ht** where the z-value of the point being extended is linearly interpolated by the extension distance.

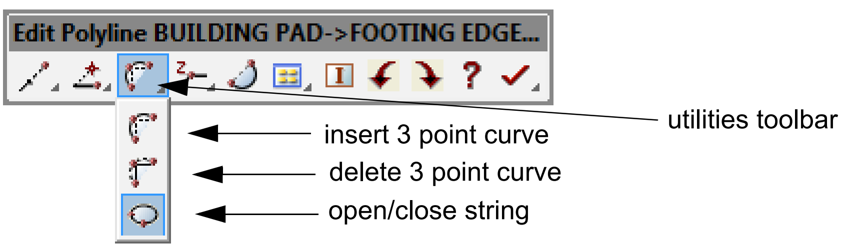
Most vertices are the end vertex of two segments and the extension could be along either segment. The vertex and the segment to extend along are both selected at the same time by picking on the segment to extend along, near the vertex to extend. The segment to extend can be a straight or an arc segment.



Go to the next section 14.4.9.3 Utilities Toolbar

14.4.9.3 Utilities Toolbar

The **utilities** toolbar contains a number of useful miscellaneous option for the polyline string. The menu is



Each of the new options will now be discussed.

14.4.9.3.1 Ins 3 Point Curve

The **Ins 3 Pt Curve** option is used to insert a curve through three adjacent vertices.

After selecting the option, the middle vertex of the three adjacent vertices is selected.

When the vertex is accepted, the radius required to fit a curve through the vertex and the two adjacent vertices is calculated, and this radius is then applied to the segments joining the adjacent vertices.

14.4.9.3.2 Del 3 Point Curve

The **del 3 pt curve** option is used to delete the curves on either side of a vertex.

After selecting the option, a vertex is selected and when the vertex is accepted, the radii of the segments on either side are set to zero.

Hence the curves on either side of the vertex are effectively removed.

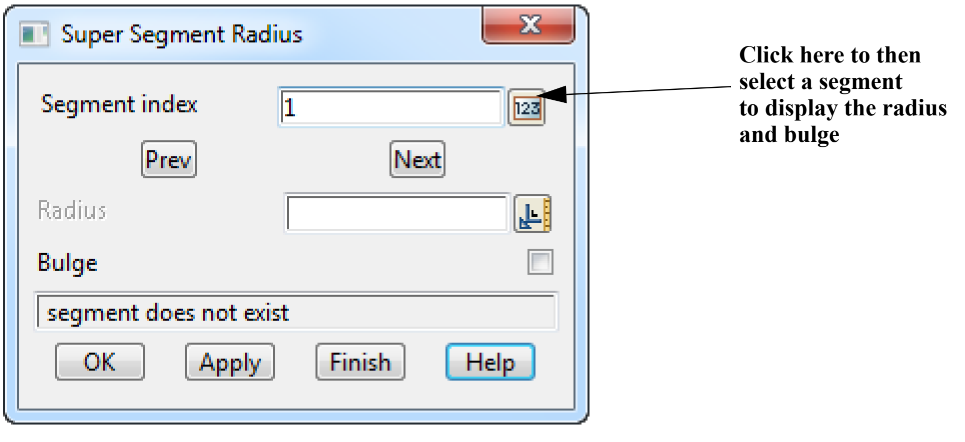
14.4.9.3.3 Open/Close

Toggles the string between being open and closed.

Go to the next section 14.4.9.4 Radius

14.4.9.4 Radius

Selecting **Radius** icon brings up the **Super Segment Radius** panel which is used to modify the radius of any arc/line joining adjacent polyline points.



The panel displays the arc radius and bulge setting for segment 1 and the next/previous buttons can be used to display the values for any segment.

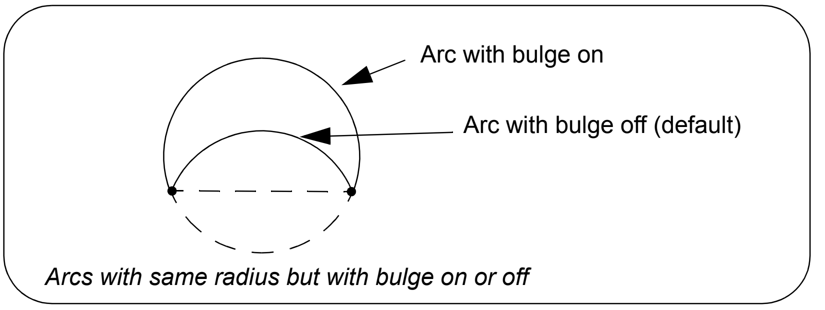
A particular segment can be selected by clicking on the value box for the Segment index field and then picking the segment. The current arc radius and bulge for the segment are displayed.

New values can then be entered into the panel field and the arc modified by selecting the **OK** or **Apply** button.

If the radius is positive, the arc is drawn from the start point to the next point on the polyline in a clockwise direction. If the radius is negative, the arc is drawn from the start point to the next point on the polyline in a counter-clockwise direction.

For a given radius (positive or negative), there are two possible cases for the arc- one where the arc is less than a semi-circle, the other when the arc is greater than a semi-circle.

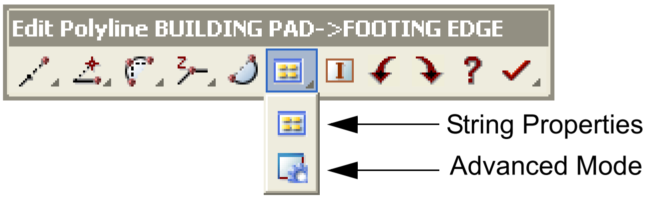
If *bulge* is ticked, the larger arc is used. The default is *bulge* turned off.

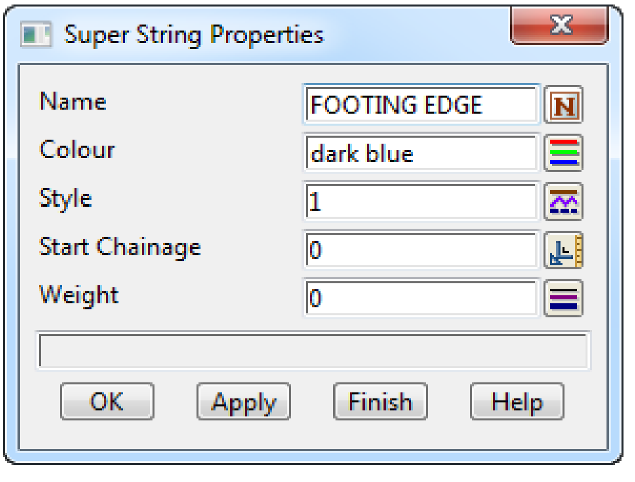


Go to the next section 14.4.9.5 Properties toolbar

14.4.9.5 Properties toolbar

Selecting the **Properties** icon brings up the **Polyline String Properties** toolbar which is used to modify the string's header information.

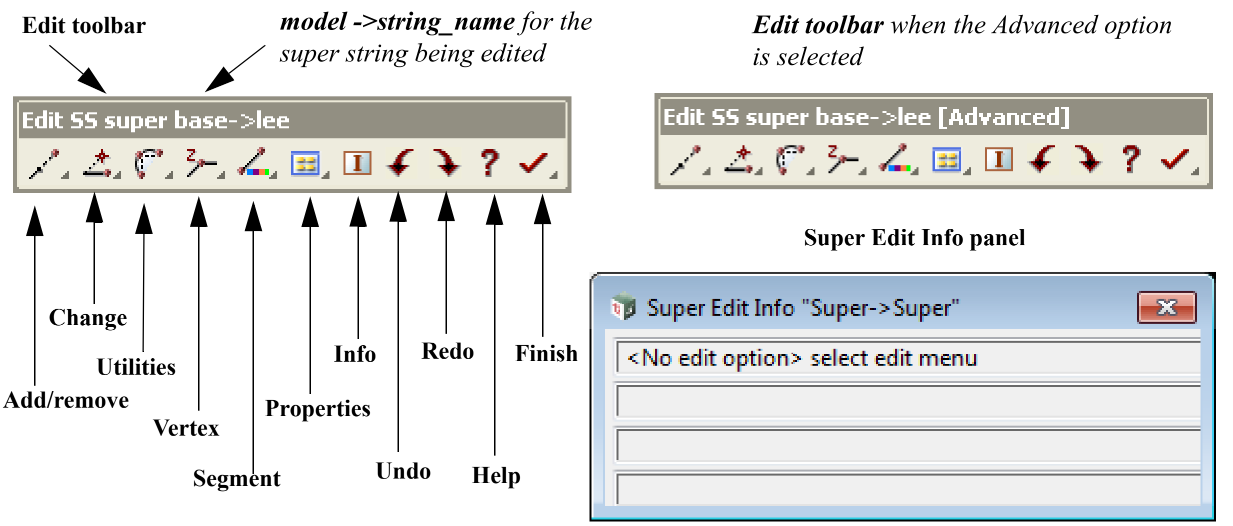




The fields in this panel are the same as for a 3d string.

14.4.10 Edit Super

On picking a super string, the **Super Edit** menu and **Super Edit Info** panel are placed on the screen.



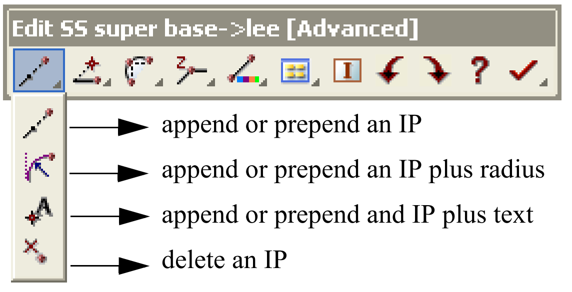
The super string is similar to a polyline string in that it can have an arc instead of a line joining adjacent string points. Hence most of the options in the **super edit** menu are similar to the **polyline edit** options of the same name, and only the differences for each option will be discussed.

See the earlier section 14.4.1 Super String Edit - Common Information for general information about editing strings.

For the option Add/ Remove Toolbar go to 14.4.10.1 Add/Remove Toolbar
Change Toolbar 14.4.10.2 Change Toolbar
Utilities Toolbar 14.4.10.3 Utilities Toolbar
Vertex Toolbar 14.4.10.4 Vertex Toolbar
Segment Toolbar 14.4.10.5 Segment Toolbar
Properties Toolbar
Info Toolbar 14.4.1 Super String Edit - Common Information
Undo Toolbar 14.4.1 Super String Edit - Common Information
Redo Toolbar 14.4.1 Super String Edit - Common Information
Help Toolbar 14.4.1 Super String Edit - Common Information
Finish Toolbar 14.4.1 Super String Edit - Common Information

14.4.10.1 Add/Remove Toolbar

The Add/Remove toolbar is



The **Append/Prepend** options allow for simply adding vertices, or adding a vertex plus a radius for the segment, or adding vertices and text for each vertex.

Append

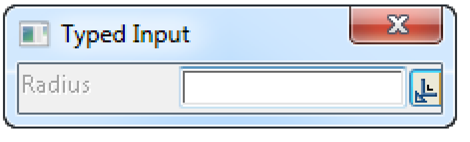
For the cases when no arc or text is required, the Append option simply places vertices.

Append + Radius

If an arc radius is required for every segment of the super string, there is an Append + Radius option that asks for the radius for every segment.

If the **Append + Radius** option is selected, then before each vertex is appended, a radius typed-input box is placed on the screen.

The radius typed-input box looks like



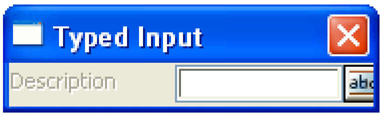
The radius is entered into the typed-input box, terminated with <enter>. The entered value is taken as the radius of the arc to the next super string vertex and the arc will be drawn correctly as the cursor is moved to the next vertex.

A radius value of 0 is taken to mean *no arc*.

Append +Text

If text is required at every vertex of the super string, there the Append + Text option that asks for the text at every vertex.

When **Append + Text** is selected, then before each vertex is appended, a Description typed-input box is placed on the screen.



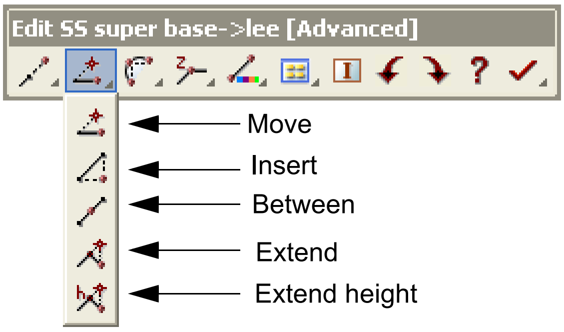
The text is entered into the typed-input box, terminated with <enter> and the entered text is taken as the vertex text,

The height question for each vertex is toggled on/off in the snaps menu just as it was for a polyline string.

The **Append** options are terminated by selecting **Cancel** from the **Pick Ops** menu or by selecting a new option from the **Super Edit** toolbar

Go to the next section 14.4.10.2 Change Toolbar

14.4.10.2 Change Toolbar



The **Extend ht** option is used to move a vertex along the line joining the vertex to its neighbouring vertex plus the z-value is interpolated from the z-values at either end of the segment.

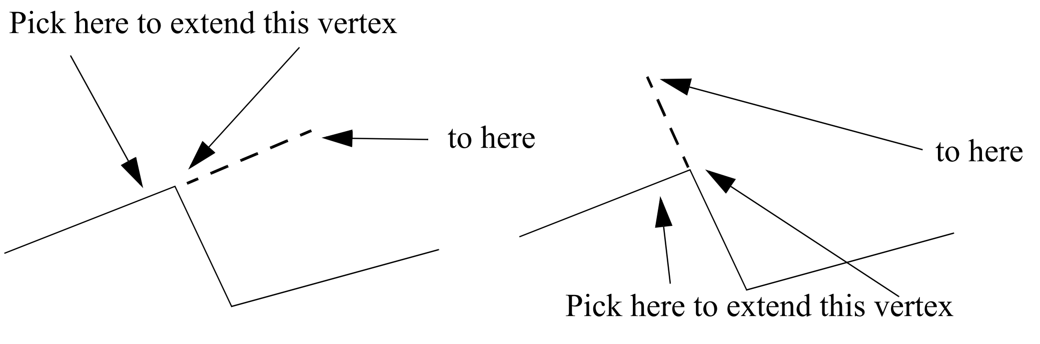
Move, Insert and Between

The **Move**, **Insert** and **Between** options are the same as for a 3d string. See

Extend and Extend Height

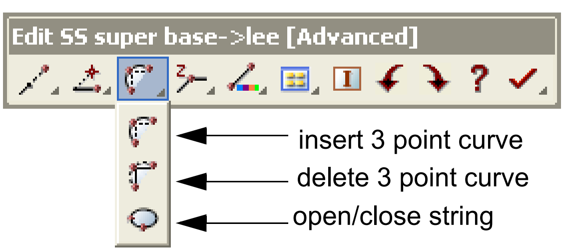
There are two **Extend** options for a super string - **Extend** that is identical to the 3d string case where the z-value of the point being extended is kept constant, and a second option, **Extend ht** where the z-value of the point being extended is linearly interpolated by the extension distance (either along the straight or arc segment).

Most vertices are the end vertex of two segments and the extension could be along either segment. The vertex and the segment to extend along are both selected at the same time by picking on the segment to extend along, near the vertex to extend. The segment to extend can be a straight or an arc segment.



Please continue to the next section 14.4.10.3 Utilities Toolbar.

14.4.10.3 Utilities Toolbar



Insert 3 Point Curve

The **Ins 3 Pt Curve** option is used to insert a curve through three adjacent vertices.

After selecting the option, the middle vertex of the three adjacent vertices is selected.

When the vertex is accepted, the radius required to fit a curve through the vertex and the two adjacent vertices is calculated, and this radius is then applied to the segments joining the adjacent vertices.

Delete 3 Point Curve

The **del 3 pt curve** option is used to delete the curves on either side of a vertex.

After selecting the option, a vertex is selected and when the vertex is accepted, the radii of the segments on either side are set to zero.

Hence the curves on either side of the vertex are effectively removed.

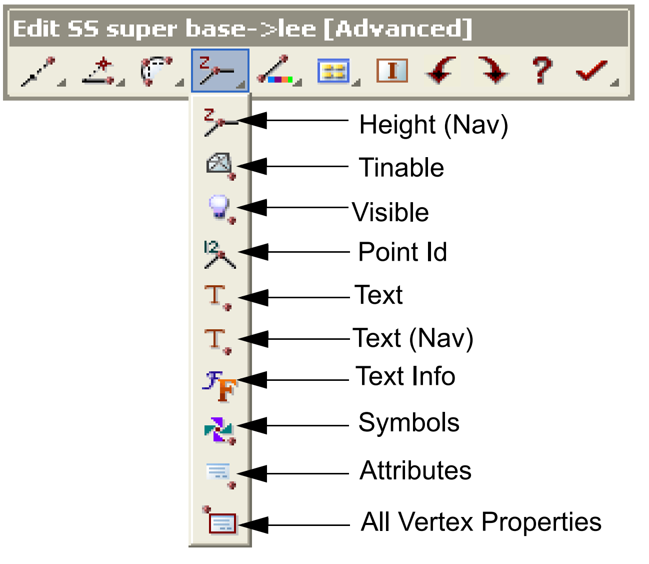
Open/Close

Toggles the string between being open and closed.

Please continue to the next section 14.4.10.4 Vertex Toolbar.

14.4.10.4 Vertex Toolbar

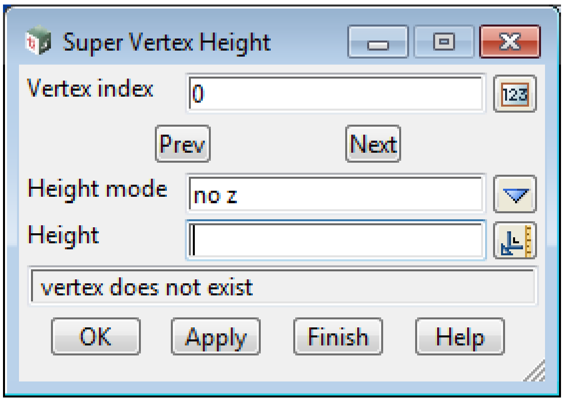
The **Vertex** toolbar contains options to modify information at any vertex of the super string.



For the option *Height (Nav)* go to	14.4.10.4.1 Height (Nav)
Tinable	14.4.10.4.2 Tinable
Visible	14.4.10.4.3 Visible
Point Id	14.4.10.4.4 Point Id
Text	14.4.10.4.5 Text
Text (Nav)	14.4.10.4.6 Text (Nav)
Text Info	14.4.10.4.7 Text Info
Symbols	14.4.10.4.8 Symbols
Attributes	14.4.10.4.9 Attributes
All vertex properties	14.4.10.4.10 All Vertex Properties

14.4.10.4.1 Height (Nav)

Selecting **Height (Nav)** brings up the **Super Vertex Height** panel which is used to set the height value for vertices.



As soon as **Height (Nav)** is chosen, the height information for the first vertex is shown.

A particular vertex can be selected by first clicking on the icon at the end of the Vertex index field, and then selecting the required vertex, or by typing a number into the Vertex index field and pressing <enter>.

Similarly the **Prev** and **Next** buttons can be used to move to adjacent vertices.

When a vertex is selected, its *vertex number*, *height mode* and *height* are written to the appropriate panel fields.

if any panel fields are modified, selecting either **OK** or **Apply** will store the new information for the vertex.

The fields and buttons used in the **Super Vertex Height** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Vertex index	number box	selected vertex	

if a vertex is selected, then its vertex number is displayed in this field.

Prev	button		

move to the previous vertex (predecessor). The information for the new vertex is displayed in the panel fields.

Next	button		

move to the next vertex (successor). The information for the new vertex is displayed in the panel fields.

Height mode	choice box		no z, entire string, each vertex

*if **no z**, there is no z value for the vertex.*
*if **entire string**, then the string has the same z value for each vertex.*
*if **each vertex**, then each vertex has a separate z value.*

Height	input	height of vertex/string	

the height used for the vertex or for the entire string.

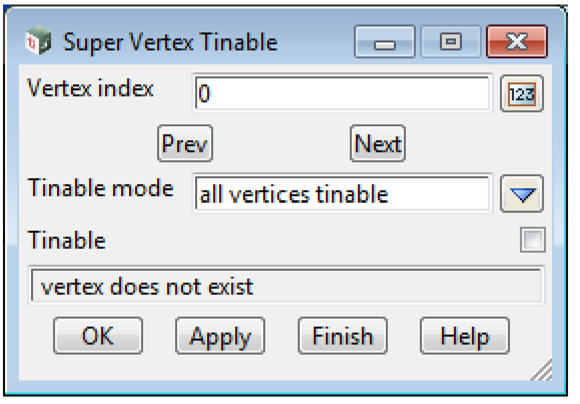
OK/Apply	button		

for the vertex being edited, **OK** *sets the vertex with the values in the panel fields and removes the panel.* **Apply** *sets the vertex with the values in the panel fields and leaves the panel on the screen.*

Please continue to the next section 14.4.10.4.2 Tinable

14.4.10.4.2 Tinable

Selecting **Tinable** brings up the **Super Vertex Tinable** panel which is used to set the tinability information for vertices.



As soon as **Tinable** is chosen, the tinability information for the first vertex is shown.

A particular vertex can be selected by first clicking on the icon at the end of the Vertex index field, and then selecting the required vertex, or by typing a number into the Vertex index field and pressing <enter>.

Similarly the **Prev** and **Next** buttons can be used to move to adjacent vertices.

When a vertex is selected, its *tinability* information is written to the appropriate panel fields.

if any panel fields are modified, selecting either **OK** or **Apply** will store the new information for the vertex.

The fields and buttons used in the **Super Vertex Tinable** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Vertex Index	number box	selected vertex	

if a vertex is selected, then its vertex number is displayed in this field.

Prev	button		

move to the previous vertex (predecessor). The information for the new vertex is displayed in the panel fields.

Next	button		

move to the next vertex (successor). The information for the new vertex is displayed in the panel fields.

Tinable mode	choice box		all vertices tinable all vertices not tinable each vertex

*if **all vertices tinable**, all vertices are included in a triangulation.*
*if **all vertices not tinable**, no vertices are included in a triangulation.*
*if **each vertex**, then each vertex has a separate tinability value.*

Tinable	tick box		

*if **ticked**, the vertex is tinable. That is, it is included in tins.*
*if **not ticked**, then the vertex is ignored when triangulating.*

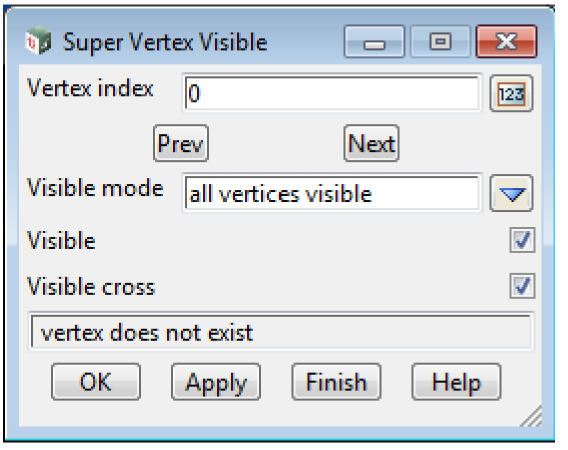
OK/Apply	button		

for the vertex being edited, **OK** *sets the vertex/string with the values in the panel fields and removes the panel.* **Apply** *sets the vertex/string with the values in the panel fields and leaves the panel on the screen.*

Please continue to the next section 14.4.10.4.3 Visible

14.4.10.4.3 Visible

Selecting **Visible** brings up the **Super Vertex Visible** panel which is used to set the visibility flag for vertices.



As soon as **Visible** is chosen, the visibility information for the first vertex is shown.

A particular vertex can be selected by first clicking on the icon at the end of the Vertex index field, and then selecting the required vertex, or by typing a number into the Vertex index field and pressing <enter>.

Similarly the **Prev** and **Next** buttons can be used to move to adjacent vertices.

When a vertex is selected, its *visibility* information is written to the appropriate panel fields.

if any panel fields are modified, selecting either **OK** or **Apply** will store the new information for the vertex.

The fields and buttons used in the **Super Vertex Visible** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Vertex index	number box	selected vertex	

if a vertex is selected, then its vertex number is displayed in this field.

Prev button

move to the previous vertex (predecessor). The information for the new vertex is displayed in the panel fields.

Next button

move to the next vertex (successor). The information for the new vertex is displayed in the panel fields.

Visible mode choice box — all vertices visible / visibility for entire string / visibility for each vertex

*if **all vertices visible**, all vertices are visible.*
*if **visibility for the entire string**, the* Visible *and* Visible cross *setting is used for all vertices in the string.*
*if **visibility each vertex**, then each vertex has its own* Visible *and* Visible cross *setting.*

Visible tick box

*if **ticked**, the vertex is visible.*
*if **not ticked**, then the vertex is invisible.*

Visible cross tick box

*if **ticked**, when it is a point string, a cross is drawn at each vertex.*
*if **not ticked**, when it is a point string, no cross is drawn at each vertex.*

***Note** - this is important when there is a symbol at the vertex. In that case, the vertex needs to be visible so that the symbol is drawn, but the standard cross drawn at vertices is not wanted.*

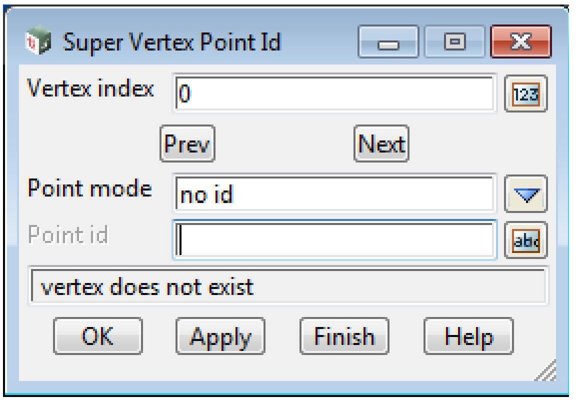
OK/Apply button

for the vertex being edited, **OK** *sets the vertex/string with the values in the panel fields and removes the panel.* **Apply** *sets the vertex/string with the values in the panel fields and leaves the panel on the screen.*

Please continue to the next section 14.4.10.4.4 Point Id

14.4.10.4.4 Point Id

Selecting **Point Id** brings up the **Super Vertex Point Id** panel which is used to set the point IDs for vertices.



As soon as **Point Id** is chosen, the Point id information for the first vertex is shown.

A particular vertex can be selected by first clicking on the icon at the end of the Vertex index field, and then selecting the required vertex, or by typing a number into the Vertex index field and pressing <enter>.

Similarly the **Prev** and **Next** buttons can be used to move to adjacent vertices.

When a vertex is selected, its *Point id* information written to the appropriate panel fields.

if any panel fields are modified, selecting either **OK** or **Apply** will store the new information for the vertex.

The fields and buttons used in the **Super Vertex Point Id** panel have the following functions.

Field Description Type Defaults Pop-Up

Vertex index number box selected vertex

if a vertex is selected, then its vertex number is displayed in this field.

Prev button

move to the previous vertex (predecessor). The information for the new vertex is displayed in the panel fields.

Next button

move to the next vertex (successor). The information for the new vertex is displayed in the panel fields.

Point id mode choice box no ids, each vertex

*if **no ids**, there are no Point ids for any vertex.*
*if **each vertex**, then each vertex has a Point id.*

Point id text box point id of vertex

the point id for the vertex.

OK/Apply button

for the vertex being edited, **OK** *sets the vertex/string with the values in the panel fields and removes the* *panel.* **Apply** *sets the vertex/string with the values in the panel fields and leaves the panel on the screen.*

Please continue to the next section 14.4.10.4.5 Text

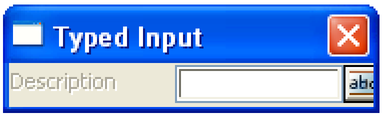
14.4.10.4.5 Text

As soon as **Text** is chosen, the Vertex Select is running and the message

<Pick vertex to set text> [Picks][Accept][Menu]

is displayed in the screen message area.

The vertex to have its text modified is then selected and the Description typed-input box is displayed on the screen with the current text for the vertex in it.

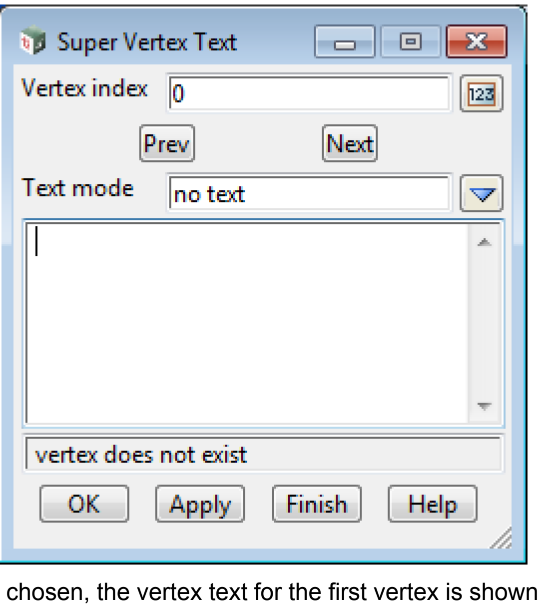


To modify the vertex text, type the new text into the Description typed-input box, finishing with <enter> and the vertex text is updated and the typed-input box.

Please continue to the next section 14.4.10.4.6 Text (Nav)

14.4.10.4.6 Text (Nav)

Selecting **Text (Nav)** brings up the **Super Vertex Text** panel which is used to set the text for vertices.



As soon as **Text (Nav)** is chosen, the vertex text for the first vertex is shown.

A particular vertex can be selected by first clicking on the icon at the end of the Vertex index field, and then selecting the required vertex, or by typing a number into the Vertex index field and pressing <enter>.

Similarly the **Prev** and **Next** buttons can be used to move to adjacent vertices.

When a vertex is selected, its vertex *text* is written to the appropriate panel fields.

if any panel fields are modified, selecting either **OK** or **Apply** will store the new information for the vertex.

The fields and buttons used in the **Super Vertex Text** panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Vertex Index input selected vertex

if a vertex is selected, then its vertex number is displayed in this field.

Prev button

move to the previous vertex (predecessor). The information for the new vertex is displayed in the panel fields.

Next button

move to the next vertex (successor). The information for the new vertex is displayed in the panel fields.

Text mode input no text, entire string, each vertex

*if **no text**, there is no text for any vertex.*
*if **entire string**, then the string has the same text for each vertex.*
*if **each vertex**, then each vertex has a separate text value.*

Text input text of vertex/string

the text used for the vertex or for the entire string.

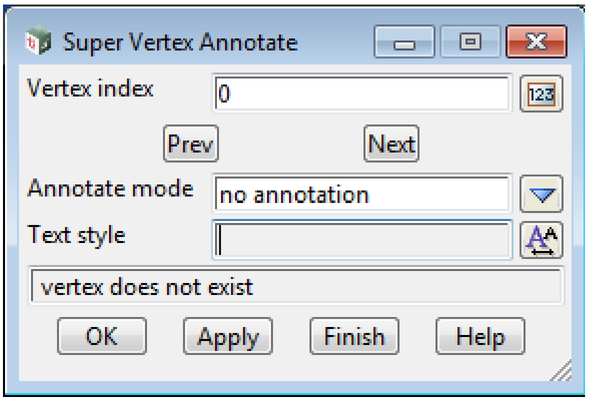
OK/Apply button

for the vertex being edited, **OK** *sets the vertex/string with the values in the panel fields and removes the panel.* **Apply** *sets the vertex/string with the values in the panel fields and leaves the panel on the screen.*

Please continue to the next section 14.4.10.4.7 Text Info

14.4.10.4.7 Text Info

Selecting **Text info** brings up the **Super Vertex Annotate** panel which is used to set the annotation styles for the text at vertices.



As soon as **Text info** is chosen, the text annotation information for the first vertex is shown.

A particular vertex can be selected by first clicking on the icon at the end of the Vertex index field, and then selecting the required vertex, or by typing a number into the Vertex index field and pressing <Enter>.

Similarly the **Prev** and **Next** buttons can be used to move to adjacent vertices.

When a vertex is selected, its *text annotation* information is written to the appropriate panel fields.

if any panel fields are modified, selecting either **OK** or **Apply** will store the new information for the vertex.

The fields and buttons used in the **Super Vertex Annotate** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Vertex Index	input	selected vertex	

if a vertex is selected, then its vertex number is displayed in this field.

Prev	button		

move to the previous vertex (predecessor). The information for the new vertex is displayed in the panel fields.

Next	button		

move to the next vertex (successor). The information for the new vertex is displayed in the panel fields.

Annotate mode	input		no annotation, entire string, each vertex

*if **no annotation**, then the text at the vertex is **not** displayed.*
*if **entire string**, then the same annotation settings are used for each vertex.*
*if **each vertex**, then each vertex has separate annotations settings.*

Text style	input	1	available text styles

text annotation for the text at the vertex.

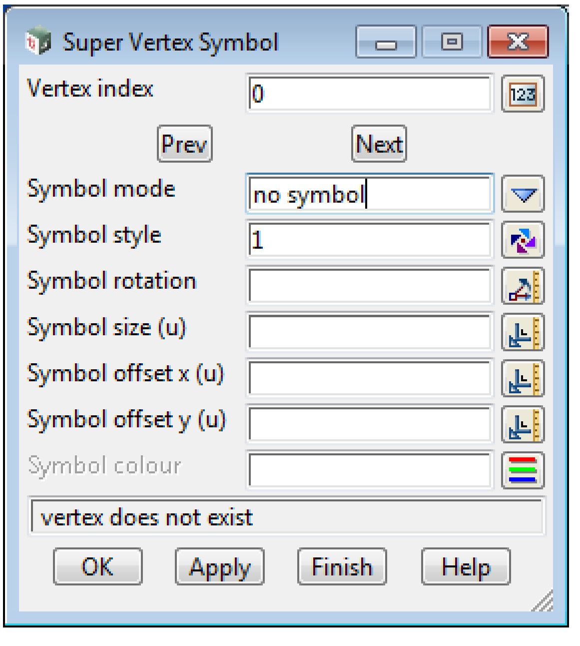
OK/Apply	button		

for the vertex being edited, **OK** *sets the vertex/string with the values in the panel fields and removes the panel.* **Apply** *sets the vertex/string with the values in the panel fields and leaves the panel on the screen.*

Please continue to the next section 14.4.10.4.8 Symbols

14.4.10.4.8 Symbols

Selecting **Symbols** brings up the **Super Vertex Symbol** panel which is used to set symbols and their display parameters at vertices.



As soon as **Symbols** is chosen, the symbol annotation information for the first vertex is shown.

A particular vertex can be selected by first clicking on the icon at the end of the Vertex index field, and then selecting the required vertex, or by typing a number into the Vertex index field and pressing <Enter>.

Similarly the **Prev** and **Next** buttons can be used to move to adjacent vertices.

When a vertex is selected, its *symbol* information is written to the appropriate panel fields.

if any panel fields are modified, selecting either **OK** or **Apply** will store the new information for the vertex.

The fields and buttons used in the **Super Vertex Symbol** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Vertex index	input	selected vertex	

if a vertex is selected, then its vertex number is displayed in this field.

Prev button

move to the previous vertex (predecessor). The information for the new vertex is displayed in the panel fields.

Next button

move to the next vertex (successor). The information for the new vertex is displayed in the panel fields.

Symbol mode input no symbol, entire string, each vertex

*if **no symbol**, then there is no symbol at any vertex.*
*if **entire string**, then the same symbol and settings are used for each vertex.*
*if **each vertex**, then each vertex has separate symbols and settings.*

Symbol style symbol box 1 available symbols

symbol at the vertex

Symbol rotation angle box

rotation angle of the symbol (measured from the positive x-axis in the counter-clockwise direction).

Symbol size (u) real box

size of the symbol (in units for the symbol).

Symbol offset x (u) real box 0

distance (in units for the symbol) to offset the symbol in the x direction from its (x,y) placement position.

Symbol offset y (u) real box 0

distance (in units for the symbol) to offset the symbol in the y direction from its (x,y) placement position.

Symbol colour input available colours

colour of the symbol if none is defined in the symbol definition

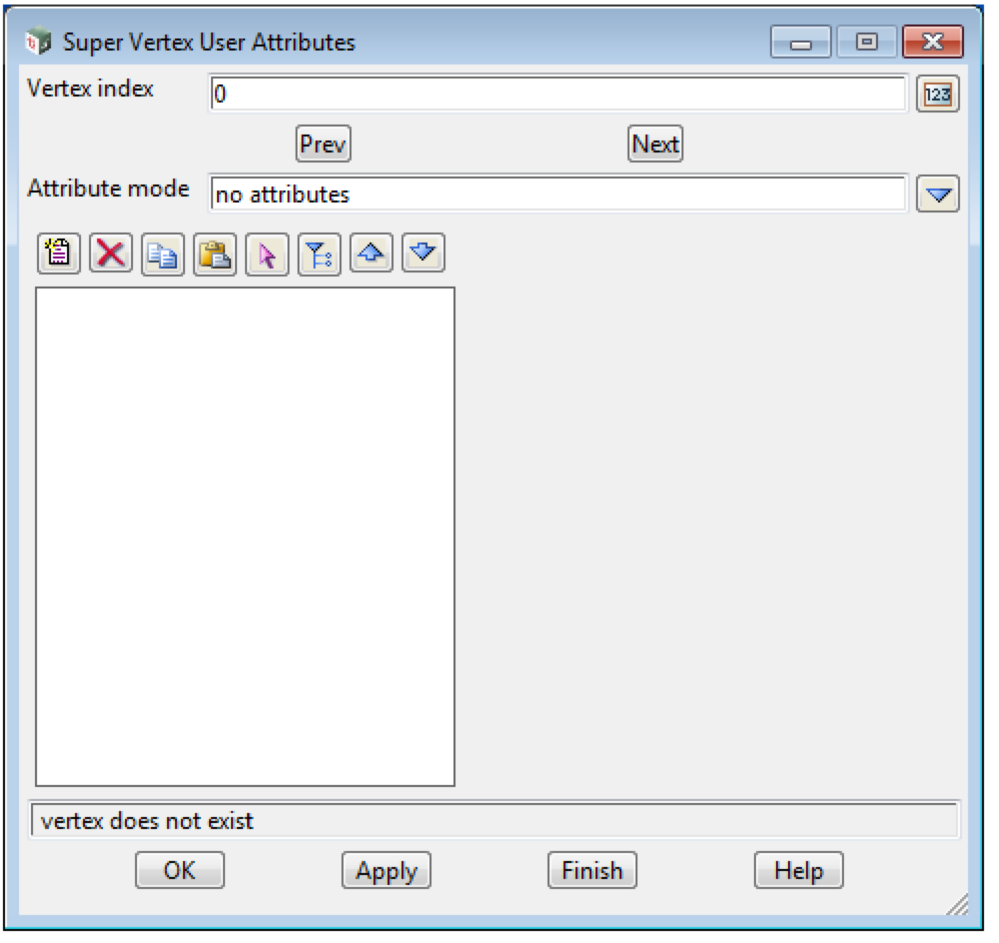
OK/Apply button

for the vertex being edited, **OK** *sets the vertex/string with the values in the panel fields and removes the panel.* **Apply** *sets the vertex/string with the values in the panel fields and leaves the panel on the screen.*

Please continue to the next section 14.4.10.4.9 Attributes

14.4.10.4.9 Attributes

Selecting **Attributes** brings up the **Super Vertex User Attributes** panel which is used to display and edit user defined attributes at vertices of the super string.



As soon as **Attribute** is chosen, the attribute information for the first vertex is shown.

A particular vertex can be selected by first clicking on the icon at the end of the Vertex index field, and then selecting the required vertex, or by typing a number into the Vertex index field and pressing <Enter>.

Similarly the **Prev** and **Next** buttons can be used to move to adjacent vertices.

When a vertex is selected, its *attribute* information is written to the appropriate panel fields.

if any panel fields are modified, selecting either **OK** or **Apply** will store the new information for the vertex.

The fields and buttons used in the **Super Vertex User Attributes** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Vertex index	input	selected vertex	

if a vertex is selected, then its vertex number is displayed in this field.

Prev	button		

move to the previous vertex (predecessor). The information for the new vertex is displayed in the panel fields.

Next	button		

move to the next vertex (successor). The information for the new vertex is displayed in the panel fields.

Attribute mode	choice box		no attributes, each vertex

*if **no attributes**, then no vertices have user attributes.*

*if **each vertex**, then each vertex can have user attributes.*

Name/Type/Data Grid

Name input

name for the user attribute. This must be unique for all attributes at this vertex.

Type choice box integer, real, text

type of the attribute.

Data input

value for the attribute.

OK/Apply button

for the vertex being edited, **OK** *sets the vertex with the values in the panel fields and removes the panel.* **Apply** *sets the vertex with the values in the panel fields and leaves the panel on the screen.*

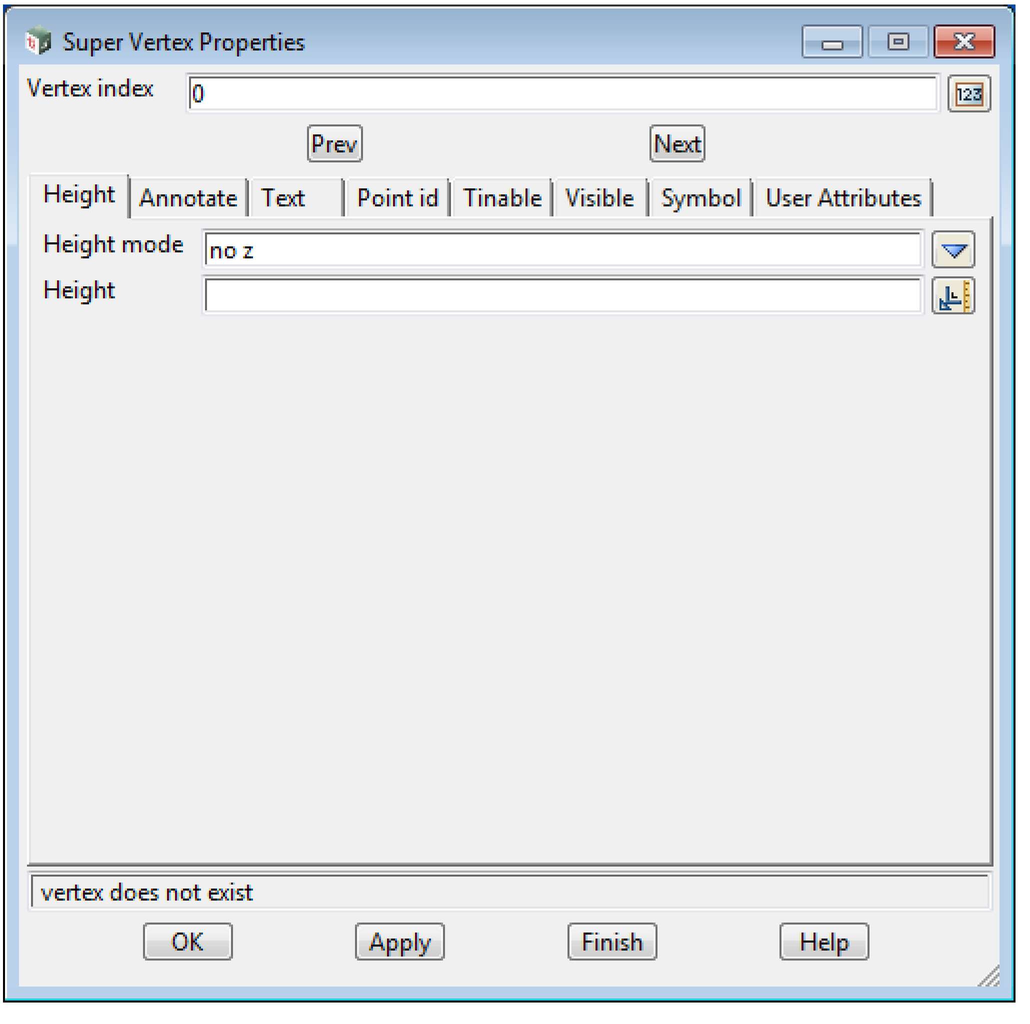
Please continue to the next section 14.4.10.4.10 All Vertex Properties

14.4.10.4.10 All Vertex Properties

Selecting **All properties** brings up the **Super Vertex Properties** panel which is used to display all the properties for a vertex.

This option is also available from the *Strings* menu

Position of option on menu: **Strings =>Properties =>Vertex (all)**



As soon as **All vertex properties** is chosen, the height information for the first segment is shown in the panel. Clicking on the various tabs will show the various vertex information for the first vertex.

A particular vertex can be selected by first clicking on the icon at the end of the Vertex index field, and then selecting the required vertex, or by typing a number into the Vertex index field and pressing <Enter>.

Similarly the **Prev** and **Next** buttons can be used to move to adjacent vertices.

When a vertex is selected, its *attribute* information is written to the appropriate tabs and panel fields.

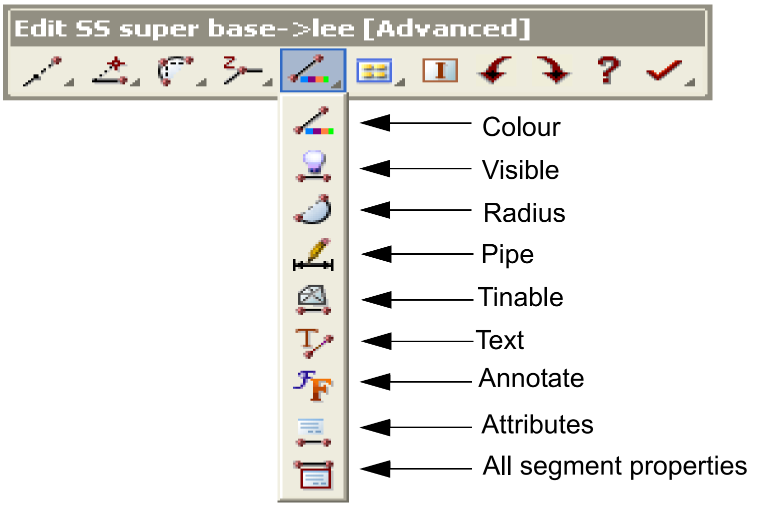
If any panel fields are modified, selecting either **OK** or **Apply** will store the new information for the vertex.

The fields in each of the tabs for the **Super Vertex Properties** panel have already been described in the other options on the Vertex Toolbar. See 14.4.10.4 Vertex Toolbar.

For information on the Segment Toolbar, please continue to the next section 14.4.10.5 Segment Toolbar.

14.4.10.5 Segment Toolbar

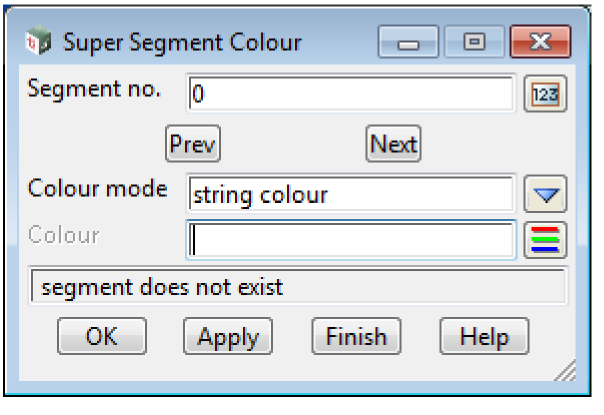
The **Segment** toolbar contains options to modify information at any segment of the super string.



For the option *Colour* go to	14.4.10.5.1 Colour
Visible	14.4.10.5.2 Visible
Radius	14.4.10.5.3 Radius
Pipe	14.4.10.5.4 Pipe
Tinable	14.4.10.5.5 Tinable
Text	14.4.10.5.6 Text
Annotate	14.4.10.5.7 Annotate
Attributes	14.4.10.5.8 Attributes
All segment properties	14.4.10.5.9 All Segment Properties

14.4.10.5.1 Colour

Selecting **Colour** brings up the **Super Segment Colour** panel which is used to set the colour of the string segments.



As soon as **Colour** is chosen, the colour information for the first segment is shown.

A particular segment can be selected by first clicking on the icon at the end of the Segment index field, and then selecting the required segment, or by typing a number into the Segment index field and pressing <Enter>.

Similarly the **Prev** and **Next** buttons can be used to move to adjacent segments.

When a segment is selected, its *colour* information is written to the appropriate panel fields.

if any panel fields are modified, selecting either **OK** or **Apply** will store the new information for the segment.

The fields and buttons used in the **Super Segment Colour** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Segment index	number box	selected vertex	

if a segment is selected, then its segment index is displayed in this field.

Prev	button		

move to the previous segment (predecessor). The information for the new segment is displayed in the panel fields.

Next	button		

move to the next segment (successor). The information for the next new is displayed in the panel fields.

Colour mode	choice box		string colour, each segment

*if **string colour**, then all the segments in the string have the same colour.*
*If **each segment**, then each segment has a separate colour.*

Colour	colour box	colour of segment/string	available colours

the colour used for the segment or for the entire string.

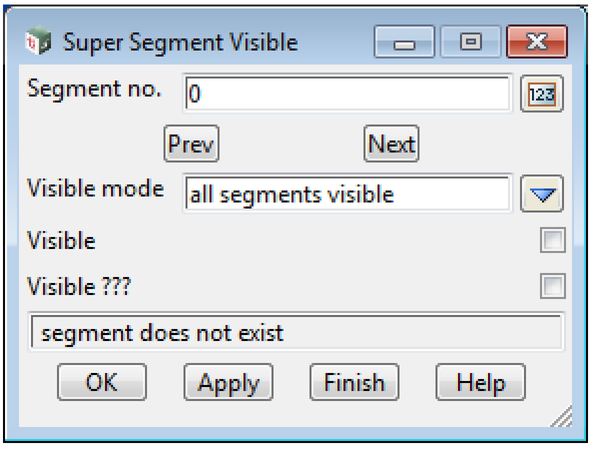
OK/Apply	button		

for the segment being edited, **OK** *sets the segment/string with the values in the panel fields and removes the panel.* **Apply** *sets the segment/string with the values in the panel fields and leaves the panel on the screen.*

Please continue to the next section 14.4.10.5.2 Visible

14.4.10.5.2 Visible

Selecting **Visible** brings up the **Super Segment Visible** panel which is used to set the visibility flag for segments.



As soon as **Visible** is chosen, the visibility information for the first segment is shown.

A particular segment can be selected by first clicking on the icon at the end of the Segment index field, and then selecting the required segment, or by typing a number into the Segment index field and pressing <Enter>.

Similarly the **Prev** and **Next** buttons can be used to move to adjacent segments.

When a segment is selected, its *visibility* information is written to the appropriate panel fields.

if any panel fields are modified, selecting either **OK** or **Apply** will store the new information for the segment.

The fields and buttons used in the **Super Segment Visible** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Segment index	number box	selected vertex	

if a segment is selected, then its segment index is displayed in this field.

Prev button

move to the previous segment (predecessor). The information for the new segment is displayed in the panel fields.

Next button

move to the next segment (successor). The information for the next new is displayed in the panel fields.

Visible mode choice box — all segments visible / visibility for entire string / visibility for each segment

*if **all segments visible**, all segments are visible.*
*if **visibility for the entire string**, the* Visible *and* Visible ??? *setting is used for all vertices in the string.*
*if **visibility each segment**, then each vertex has its own* Visible *and* Visible ??? *setting.*

Visible tick box

*if **ticked**, the segment is visible.*
*if **not ticked**, then the segment is invisible.*

Visible??? tick box

*if **ticked**,*
*if **not ticked**,.*

OK/Apply button

for the segment being edited, **OK** *sets the segment/string with the values in the panel fields and removes the panel.* **Apply** *sets the segment/string with the values in the panel fields and leaves the panel on the screen.*

Please continue to the next section 14.4.10.5.3 Radius

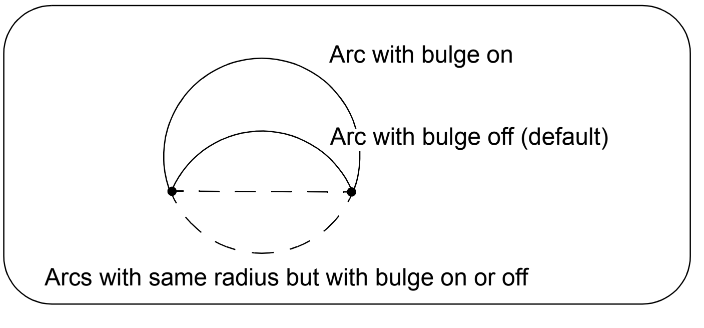
14.4.10.5.3 Radius

When viewed in plan, the segments of a super string can be joined by string lines or arcs.
If the radius is positive, the arc is drawn from the start vertex to the end vertex of the segment in a clockwise direction. If the radius is negative, the arc is drawn from the start vertex to the end vertex on the segment in a counter-clockwise direction.

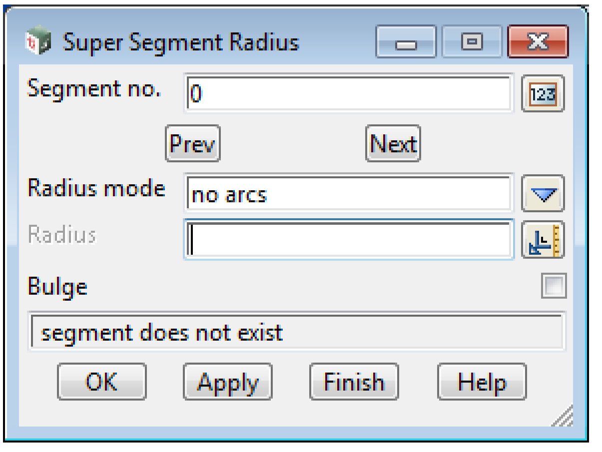
For a given radius (positive or negative), there are two possible cases for the arc- one where the arc is less than a semi-circle, the other when the arc is greater than a semi-circle.

If bulge is turned *on*, the larger arc is used. The default is bulge turned off.

A zero radius is interpreted to be a just a straight line segment with no arc.



Selecting **Radius** brings up the **Super Segment Radius** panel which is used to set the plan radius of the segment.



As soon as **Radius** is chosen, the radius information for the first segment is shown.

A particular segment can be selected by first clicking on the icon at the end of the Segment index field, and then selecting the required segment, or by typing a number into the Segment index field and pressing <Enter>.

Similarly the **Prev** and **Next** buttons can be used to move to adjacent segments.

When a segment is selected, its *radius* information is written to the appropriate panel fields.

if any panel fields are modified, selecting either **OK** or **Apply** will store the new information for the segment.

The fields and buttons used in the **Super Segment Radius** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Segment index	number box	selected vertex	

if a segment is selected, then its segment index is displayed in this field.

Prev	button		

move to the previous segment (predecessor). The information for the new segment is displayed in the panel fields.

Next	button		

move to the next segment (successor). The information for the new segment is displayed in the panel fields.

Radius mode	choice box		no arcs, each segment

*if **no arcs**, then no segments have a diameter (that is, they are all line segments).*
*If **each segment**, then each segment has a radius.*

Radius	real box	0	

radius of the segment arc. A radius of 0 mean no arc.

Bulge	tick box		

*if **ticked**, the larger arc is used.*
if not-ticked, the smaller arc is used.

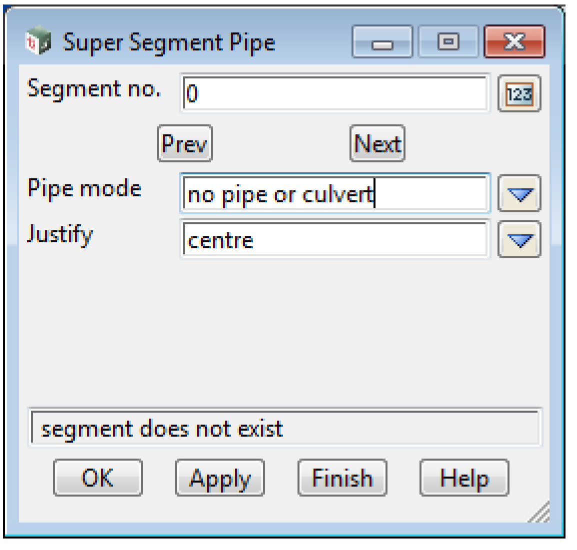
OK/Apply	button		

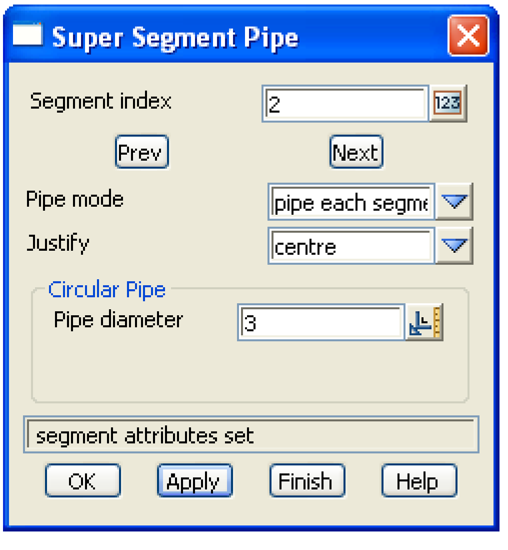
for the segment being edited, **OK** *sets the segment/string with the values in the panel fields and removes the panel.* **Apply** *sets the segment/string with the values in the panel fields and leaves the panel on the screen.*

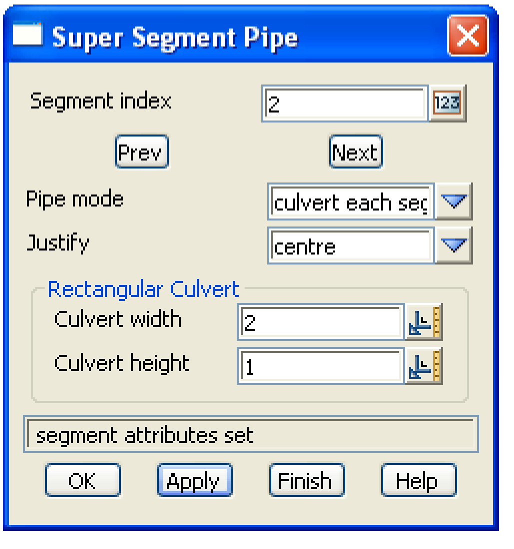
Please continue to the next section 14.4.10.5.4 Pipe

14.4.10.5.4 Pipe

The segments of a super string can have either a pipe or box cross section, or none.
Selecting **Pipe** brings up the **Super Segment Pipe** panel which is used to set the pipe mode and size for the string segments.







As soon as **Pipe** is chosen, the pipe information for the first segment is shown.

A particular segment can be selected by first clicking on the icon at the end of the Segment index field, and then selecting the required segment, or by typing a number into the Segment index field and pressing <Enter>.

Similarly the **Prev** and **Next** buttons can be used to move to adjacent segments.

When a segment is selected, its *pipe* information is written to the appropriate panel fields.

if any panel fields are modified, selecting either **OK** or **Apply** will store the new information for the segment.

The fields and buttons used in the **Super Segment Pipe** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Segment index	number box	selected vertex	

if a segment is selected, then its segment index is displayed in this field.

Prev button

move to the previous segment (predecessor). The information for the new segment is displayed in the panel fields.

Next button

move to the next segment (successor). The information for the new segment is displayed in the panel fields.

Pipe mode choice box — no pipe or culvert, pipe entire string, pipe each segment, culvert entire string, culvert each segment

*if **no pipe or culvert**, there is no pipe or culvert for the segment.*
*If **pipe entire string**, then all the segments in the string are pipes and have the same diameter.*
*If **pipe each segment**, then each segment is a pipe and can have a different diameter.*
*If **culvert entire string**, then all the segments in the string are culverts and have the same width and height.*
*If **culvert each segment**, then each segment is a culvert and can have a different width and height.*

Justify choice box — invert, centre, overt

justification of the pipe/culvert with respect to the co-ordinates given for the vertices of the super string.

Pipe diameter real box

diameter of the pipe in world units.

Culvert width real box

width of the culvert section in world units.

Culvert height real box

height of the culvert section in world units.

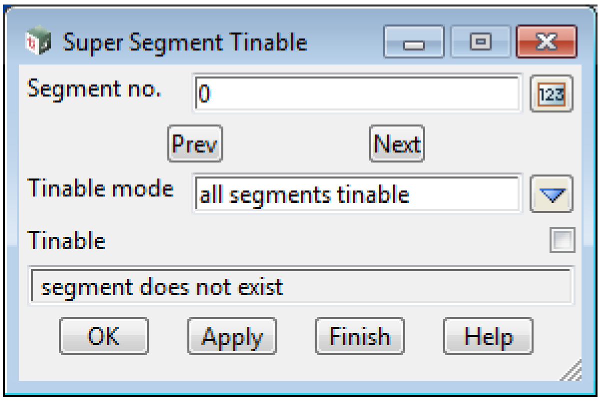
OK/Apply button

for the segment being edited, **OK** *sets the segment/string with the values in the panel fields and removes the panel.* **Apply** *sets the segment/string with the values in the panel fields and leaves the panel on the screen.*

Please continue to the next section 14.4.10.5.5 Tinable

14.4.10.5.5 Tinable

Selecting **Segment=>Tinable** brings up the **Super Segment Tinable** panel which is used to set the tinable flag for segments.



As soon as **Tinable** is chosen, the tinability information for the first segment is shown.

A particular segment can be selected by first clicking on the icon at the end of the Segment index field, and then selecting the required segment, or by typing a number into the Segment index field and pressing <enter>.

Similarly the **Prev** and **Next** buttons can be used to move to adjacent segments.

When a segment is selected, its *tinability* information is written to the appropriate panel fields.

if any panel fields are modified, selecting either **OK** or **Apply** will store the new information for the segment.

The fields and buttons used in the **Super Segment Tinable** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Segment index	number box	selected vertex	

if a segment is selected, then its segment index is displayed in this field.

Prev	button		

move to the previous segment (predecessor). The information for the new segment is displayed in the panel fields.

Next	button		

move to the next segment (successor). The information for the next new is displayed in the panel fields.

Tinable mode	choice box		all segments tinable all segments not tinable each segment

*if **all segments tinable**, all segments are included as break lines in a triangulation.*
*if **all segments not tinable**, no segments are included as break lines in a triangulation.*
*if **each segment**, then each segment has a separate tinability value.*

Tinable	tick box		

*if **ticked**, the segment is used as a break line in a triangulation. That is, the triangulation process tries to preserve the segment as a side of a triangle in the tin.*
*if **not ticked**, then the segment is used as a break line when triangulating.*

***Note** - for a tinable segment to be used as a breakline, the vertex at the segment ends must also be tinable.*

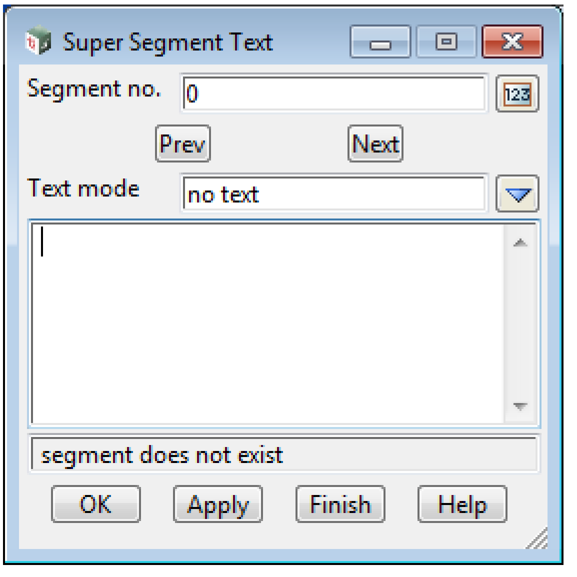
OK/Apply	button		

for the segment being edited, **OK** *sets the segment/string with the values in the panel fields and removes the panel.* **Apply** *sets the segment/string with the values in the panel fields and leaves the panel on the screen.*

Please continue to the next section 14.4.10.5.6 Text

14.4.10.5.6 Text

Selecting **Segment=>Text** brings up the **Super Segment Text** panel which is used to set the text for segments.



As soon as **Text** is chosen, the segment text for the first segment is shown.

A particular segment can be selected by first clicking on the icon at the end of the Segment index field, and then selecting the required segment, or by typing a number into the Segment index field and pressing <enter>.

Similarly the **Prev** and **Next** buttons can be used to move to adjacent segments.

When a segment is selected, its *segment text* is written to the appropriate panel fields.

if any panel fields are modified, selecting either **OK** or **Apply** will store the new information for the segment.

The fields and buttons used in the **Super Segment Text** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Segment index	number box	selected vertex	

if a segment is selected, then its segment index is displayed in this field.

Prev	button		

move to the previous segment (predecessor). The information for the new segment is displayed in the panel fields.

Next	button		

move to the next segment (successor). The information for the next new is displayed in the panel fields.

Text mode	choice box		no text, entire string, each segment

*if **no text**, there is no text for the segment.*
*if **entire string**, then the string has the same text for each segment.*
*if **each segment**, then each segment has a separate text value.*

Text	text box	text of segment/string	

the text used for the segment or for the entire string.

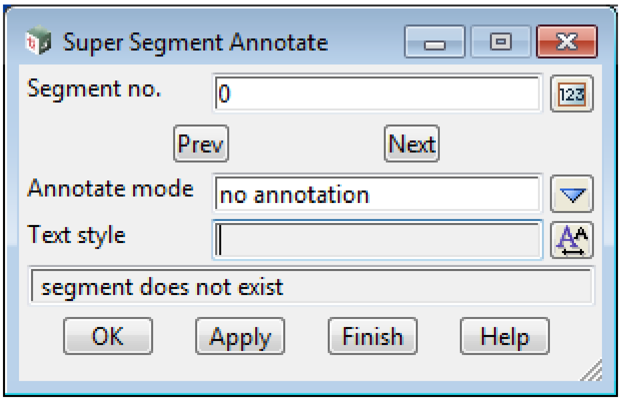
OK/Apply	button		

for the segment being edited, **OK** *sets the segment/string with the values in the panel fields and removes the panel.* **Apply** *sets the segment/string with the values in the panel fields and leaves the panel on the screen.*

Please continue to the next section 14.4.10.5.7 Annotate

14.4.10.5.7 Annotate

Selecting **Annotate** brings up the **Super Segment Annotate** panel which is used to set the annotation styles for the text at segments.



As soon as **Annotate** is chosen, the annotation information for the first segment is shown.

A particular segment can be selected by first clicking on the icon at the end of the Segment index field, and then selecting the required segment, or by typing a number into the Segment index field and pressing <enter>.

Similarly the **Prev** and **Next** buttons can be used to move to adjacent segments.

When a segment is selected, its *annotation* information is written to the appropriate panel fields.

if any panel fields are modified, selecting either **OK** or **Apply** will store the new information for the segment.

The fields and buttons used in the **Super Segment Annotate** panel have the following functions.

Field Description Type Defaults Pop-Up

Segment index number box selected vertex

if a segment is selected, then its segment index is displayed in this field.

Prev button

move to the previous segment (predecessor). The information for the new segment is displayed in the panel fields.

Next button

move to the next segment (successor). The information for the new segment is displayed in the panel fields.

Annotate mode choice box no annotation, entire string, each segment

if ***no annotation****, then the text at the segment is not displayed.*
if ***entire string****, then the same annotation settings are used all each segments and the Text style information is given in the* Text style *field.*
if ***each segment****, then each segment has separate annotations settings and the Text style information for each segment is given in the* Text style *field.*

Text style textstyle data box

textstyle information.

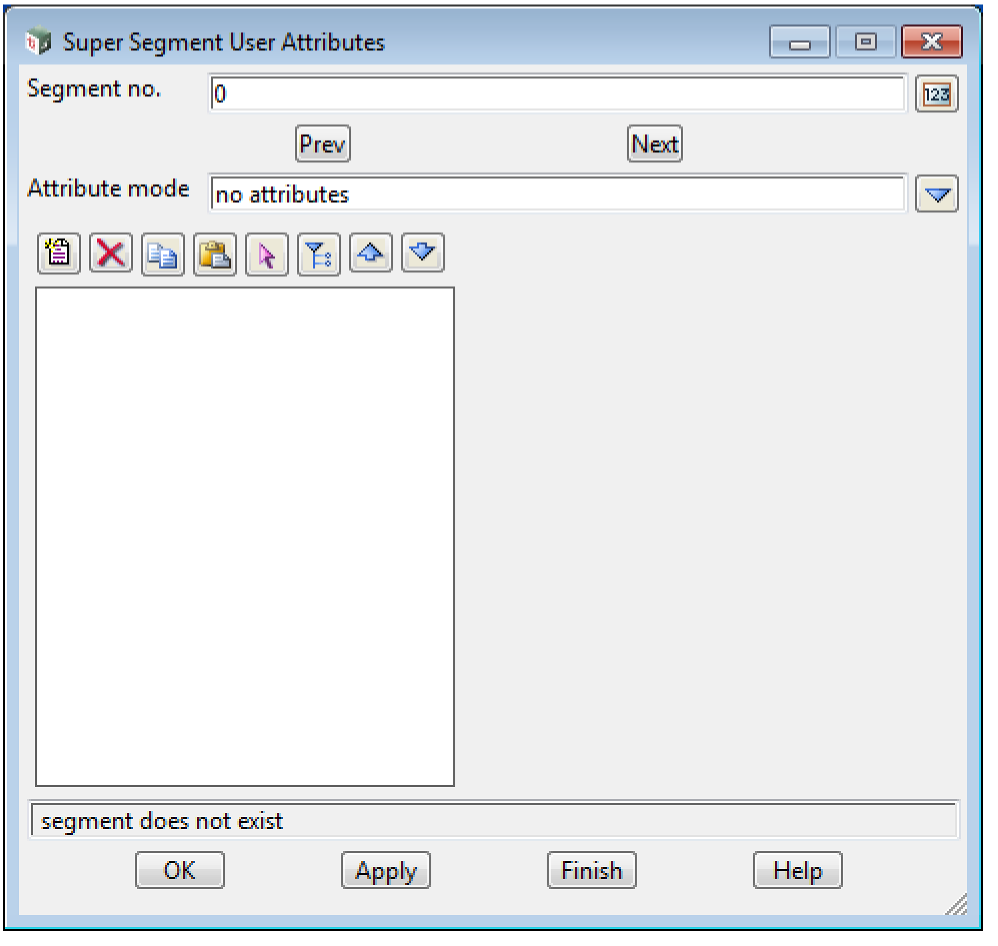
OK/Apply button

for the segment being edited, **OK** *sets the segment/string with the values in the panel fields and removes the panel.* **Apply** *sets the segment/string with the values in the panel fields and leaves the panel on the screen.*

Please continue to the next section 14.4.10.5.8 Attributes.

14.4.10.5.8 Attributes

Selecting **Attributes** brings up the **Super Segment User Attributes** panel which is used to display and edit user defined attributes at segments of the super string.



As soon as **Attributes** is chosen, the attributes for the first segment are shown.

A particular segment can be selected by first clicking on the icon at the end of the Segment index field, and then selecting the required segment, or by typing a number into the Segment index field and pressing <Enter>.

Similarly the **Prev** and **Next** buttons can be used to move to adjacent segments.

When a segment is selected, its *attribute* information is written to the appropriate panel fields.

if any panel fields are modified, selecting either **OK** or **Apply** will store the new information for the segment.

The fields and buttons used in the **Super Segment User Attributes** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Segment index	number box	selected vertex	

if a segment is selected, then its segment index is displayed in this field.

Prev button

move to the previous segment (predecessor). The information for the new segment is displayed in the panel fields.

Next button

move to the next segment (successor). The information for the new segment is displayed in the panel fields.

Attribute mode choice box no attributes, each segment

*if **no attributes**, then no segments have user attributes.*
*if **each segment**, then each segment can have user attributes.*

Name/Type/Data Grid

Name input

name for the user attribute. This must be unique for all attributes at this segment.

Type choice box integer, real, text

type of the attribute.

Data Grid grid

values for the attributes.

OK/Apply button

for the segment being edited, **OK** *sets the segment with the values in the panel fields and removes the panel.* **Apply** *sets the segment with the values in the panel fields and leaves the panel on the screen.*

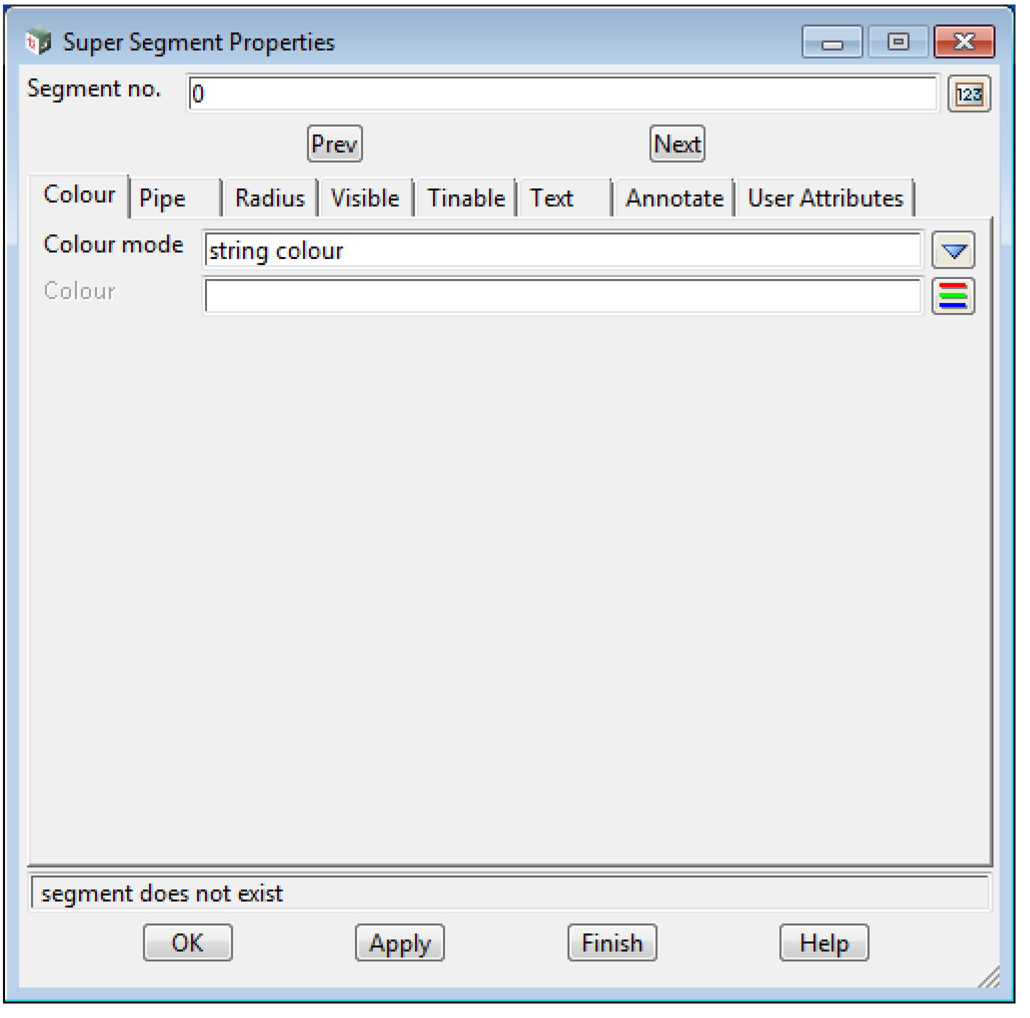
Please continue to the next section 14.4.10.5.9 All Segment Properties

14.4.10.5.9 All Segment Properties

Selecting **All properties** brings up the **Super Segment Properties** panel which is used to display all the properties of a segment.

This option is also available from the *Strings* menu

Position of option on menu: **Strings =>Properties =>Segment (all)**



As soon as **All segment properties** is chosen, the colour information for the first segment is shown. Clicking on the various tabs will show the various segment information for the first segment.

A particular segment can be selected by first clicking on the icon at the end of the Segment index field, and then selecting the required segment.

When the segment to modify is selected, its *segment index* and all other *information* are written to the appropriate panel fields.

Another segment can then be selected or the **Prev** and **Next** buttons used to move to adjacent segments and display data for that segment.

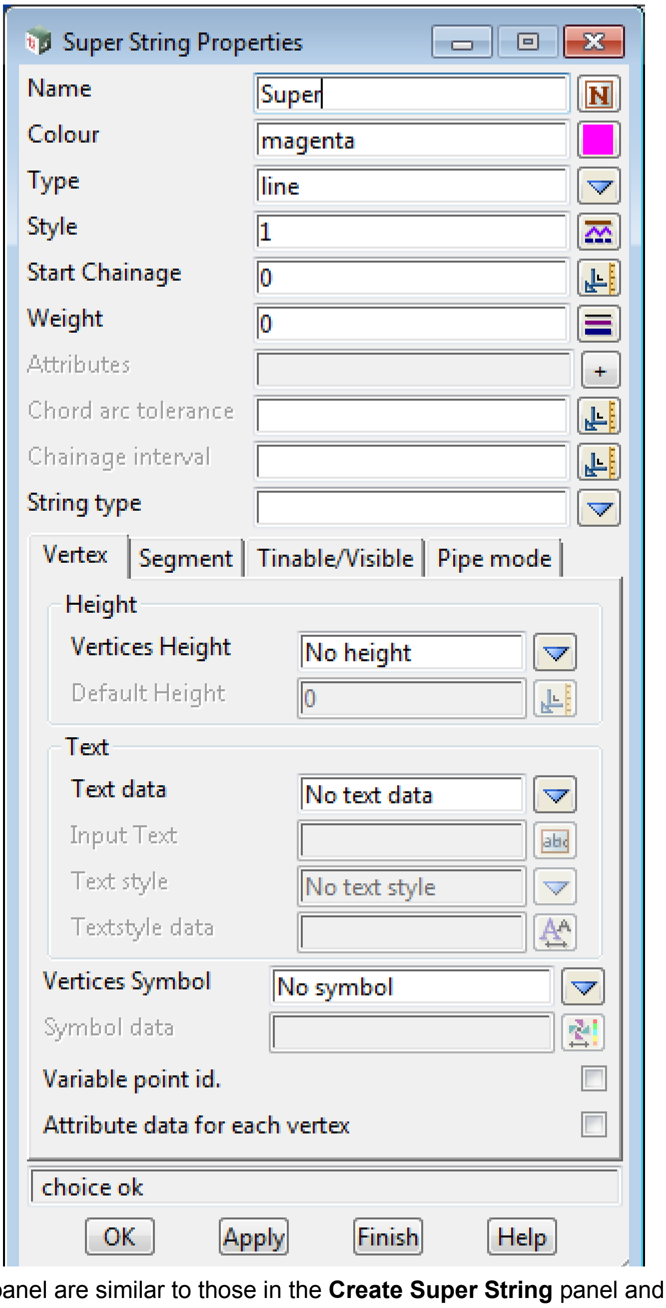
After any panel fields are modified, selecting either **OK** or **Apply** will change the information for the segment.

The fields in each of the tabs for the **Super Segment Properties** panel have already been described in the other options on the Segment Toolbar. See 14.4.10.5 Segment Toolbar.

For information on the Properties Toolbar, please continue to the next section 14.4.10.5.10 Properties Toolbar

14.4.10.5.10 Properties Toolbar

Selecting **Properties** brings up the **Super String Properties** panel which is used to modify the string's header information.



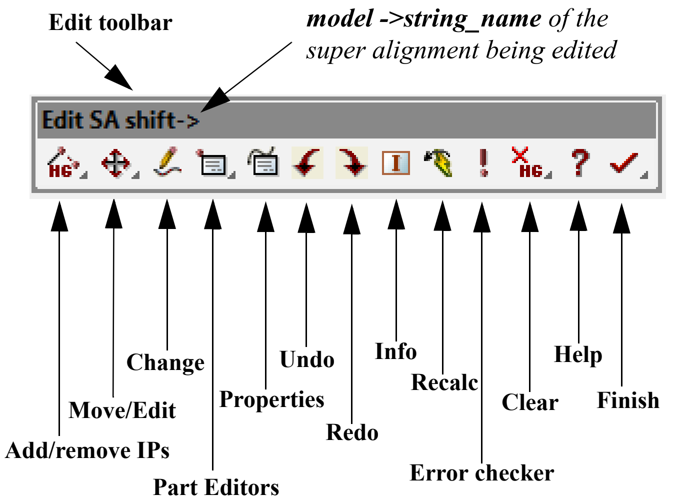
The fields in this panel are similar to those in the **Create Super String** panel and the Super String Editor options. The only new field is

Field Description	Type	Defaults	Pop-Up
OK/Apply	button		

for the string being edited, **OK** *sets the string with the values in the panel fields and removes the panel.* **Apply** *sets the string with the values in the panel fields and leaves the panel on the screen.*

14.4.11 Edit Super Alignment

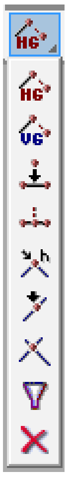
On picking a super alignment, the **Super Alignment Edit** menu is placed on the screen.



For the option *Add/ Remove* toolbar go to 14.4.11.1 SA - Add/Remove IPs Toolbar
Move/Edit toolbar 14.4.11.2 SA - Move/Edit Toolbar
Change icon 14.4.11.3 SA - Change Icon
Part Editors toolbar 14.4.11.4 SA - Part Editors Toolbar
Properties toolbar 14.4.11.5 SA - Super Alignment
Properties Icon
Undo icon 14.4.11.6 SA - Undo and Redo Icons
Redo icon 14.4.11.6 SA - Undo and Redo Icons
Info icon 14.4.11.7 SA - Info Icon
Recalc icon 14.4.11.8 SA - Recalc Icon
Error Checker icon 14.4.11.9 SA - Error Checker Icon
Clear toolbar 14.4.11.10 SA - Clear Toolbar
Help icon 14.4.11.11 SA - Help Icon
Finish toolbar 14.4.11.12 SA - Finish Toolbar

14.4.11.1 SA - Add/Remove IPs Toolbar

The Add/Remove IPs toolbar is



append or prepend a HIP
append or prepend a VIP
insert IP
between IP
insert VIP height
insert VIP grade
intersect VIP grades
filter IPs
delete an IP

The **Append** operations for a super alignment string are similar to the 2d string case except that there are separate **Append** options for the horizontal and vertical geometry of the string.

For the option:

Append HIP go to	14.4.11.1.1 SA - Append/Prepend HIP - Horizontal Geometry
Append VIP	14.4.11.1.2 SA - Append/Prepend VIP - Vertical Geometry

The **Insert IP, Between IP** and **Delete IP** work on the horizontal geometry if used on a plan view, and if used on a section view that the super alignment is profiled on, they work on the vertical geometry.

For the option *Insert IP* go to	14.4.11.1.3 SA - Insert IP
Between IP	14.4.11.1.4 SA - Between IP
Delete IP	14.4.11.1.9 SA - Delete IP

For the option *Insert VIP Height*, go to	14.4.11.1.5 SA - Insert VIP Height
Insert VIP Grade	14.4.11.1.6 SA - Insert VIP Grade
Intersect VIP Grades	14.4.11.1.7 SA - Intersect VIP Grades
Filter IPs	14.4.11.1.8 SA - Filter IPs
Delete an IP	14.4.11.1.9 SA - Delete IP

14.4.11.1.1 SA - Append/Prepend HIP - Horizontal Geometry

The **Append HIP** option is used in a plan view to create the first horizontal intersection point (HIP) in a new super alignment string and then append further HIPs, or for an existing super alignment string to append news HIPs to the end of the string, or to new HIPs to the beginning of the string. In this option, both appending and prepending will be referred to as Appending.

14.4.11.1.1.1 Existing Super Alignment

Appending an HIP is a two step process.

Step (a) - selecting the end to Append the HIP to.

After picking **Append HIP**, the end of the string to append the point to is selected. Once the string end is selected, the new HIP is assumed to be at the current cursor location. As the cursor is moved, the string is redrawn reflecting the changing position of the appended intersection point

screen message area <Pick end to append to> [picks][fast][Menu>

Step (b) - selecting the position for the new appended HIP.

The position of the new appended HIP is set to the current cursor position by picking (LB) and accepting (MB).

screen message area <Pick IP location or (r)elative, (b)earing/distance> [picks][fast][Menu

Once an HIP has been appended to the string, the appended HIP is considered to be the selected string end and a new Append cycle begins. That is, stage (a) is already set up. The current cursor position indicates the new position of the next appended HIP.

Hence a series of HIPs is easily entered by first selecting the string end in a plan view that the new points are to be appended to (step (a)) and then moving the cursor to the position of each new point in turn and selecting it.

Typed input: **r** for relative, **b** for bearing-distance can be used in either step.

14.4.11.1.1.2 New Super Alignment

When the **Append HIP** option is selected, a cross will appear on any plan views that have the super alignment string's model on them.

screen message area <Pick start IP location> [picks][fast][Menu>

The first HIP is then selected by the standard pick (LB) and accept (MB).

The option then continues as if appending to an existing super alignment string where the end HIP has already been accepted.

Typed input: **r** for relative, **b** for bearing-distance can be used in either step.

The **Append HIP** option is terminated by either pressing RB to bring up the **Pick Ops** menu and selecting **Cancel** or by selecting a new option from the **Edit SA** toolbar.

If, after bringing up the **Pick Ops** menu, it is decided to continue with the **Append HIP** option, simply select the **Restart** option from the **Pick Ops** menu and the **Pick Ops** menu will disappear leaving the **Append HIP** option still current.

14.4.11.1.2 SA - Append/Prepend VIP - Vertical Geometry

The **Append HIP** option is used in a section view to create vertical geometry for a super alignment that must already have some horizontal geometry. The existing super alignment needs to be the primary string on the section view. That is, it is the string profiled on the section view that is going to be used to create the vertical geometry. This can be achieved by using either the **VG edit** or the **Profile** option from the section view menu

Note: on a section view, the option **Utilities =>VG edit** takes the selected super alignment and profiles it on the section view and also opens the **SA Editor** for the string.

The **Append HIP** option can the first vertical intersection point (VIP) in an existing super alignment string that has **no vertical geometry**, and then appends further VIPs, or for an existing super alignment string **with vertical geometry**, OR appends new VIPs to the end or the beginning of the vertical geometry of the string. In this option, both appending and prepending will be referred to as Appending.

14.4.11.1.2.1 Existing Vertical Geometry

Appending a VIP point is almost the same as for a HIP except that the co-ordinate system is (chainage,height) rather than (x,y) values.

Appending an VIP is a two step process.

Step (a) - selecting the end to Append the VIP to.

After picking **Append VIP**, the end of the vertical geometry to append the point to is selected in the section view. Once the vertical geometry end is selected, the new VIP is assumed to be at the current cursor location and as the cursor is moved, the string is redrawn reflecting the changing position of the appended VIP

screen message area <Pick end to append to> [picks][fast][Menu>

Step (b) - selecting the position for the new appended HIP.

The position of the new appended VIP is set to the current cursor position by picking (LB) and accepting (MB).

screen message area <Pick IP location or (r)elative, (g)rade/height> [picks][fast][Menu

Once a VIP has been appended to the string, the appended VIP is considered to be the selected vertical geometry end and a new Append cycle begins. That is, stage (a) is already set up. The current cursor position in the section view indicates the new position of the next appended VIP.

Typed input: r for relative, **g** for grade-height can be used in either step.

Hence a series of VIPs is easily entered by first selecting the vertical geometry end that the new points are to be appended to (step (a)) and then moving the cursor to the position of each new point in turn and selecting them in.

14.4.11.1.2.2 New Vertical Geometry

Creating the **first VIP** is similar to the first HIP.

When the **Append VIP** option is selected, a cross will appear on any section views that have the super alignment string profiled on them.

screen message area <Pick start IP location> [picks][fast][Menu>

The first VIP is then selected by the standard pick (LB) and accept (MB).

The option then continues as if appending to an existing vertical geometry where the end point has already been accepted.

Typed input: r for relative, **g** for grade-height can be used in either step.

The **Append VIP** option is terminated by either pressing RB to bring up the **Pick Ops** menu and selecting **Cancel** or by selecting a new option from the **Edit SA** toolbar.

If, after bringing up the **Pick Ops** menu, it is decided to continue with the **Append VIP** option, simply select the **Restart** option from the **Pick Ops** menu and the **Pick Ops** menu will disappear leaving the **Append VIP** option still current.

14.4.11.1.3 SA - Insert IP

The **Insert** option is designed to place a new HIP or VIP between two adjacent IPs.

Insert IP works on the horizontal or the vertical geometry depending on whether a plan or section view is used for the editing.

The inserted IP does not have to be on the line joining the two intersection points - the Between IP option is used to guarantee that the IP is on the IP-IP line.

Inserting an IP is a two step process.

Step (a) - selecting the IP's to be on either side of the new intersection point

The two adjacent IPs are chosen by selecting the line connecting the two intersection points. Once the line is selected, the new IP is assumed to be at the current cursor position. As the cursor is moved, the string is redrawn reflecting the changing position of the inserted IP.

For horizontal and vertical inserts

screen message area <Pick position to insert point> [picks][fast][Menu>

Step (b) - selecting the position for the new IP

The position of the new IP is set to the current cursor position by picking (LB) and accepting (MB).

screen message area <Pick final position of point> [picks][fast][Menu>

Once the insert cycle is completed and the IP inserted, the Insert IP option is still current and can be repeated for other insertions (in either plan or section views) without having to re-select the Insert IP option.

The **Insert IP** option is terminated by selecting **Cancel** from the **Pick Ops** menu or by selecting a new option from the **Edit SA** toolbar.

If, after bringing up the **Pick Ops** menu, it is decided to continue with the **Insert IP** option, simply select the **Restart** option from the **Pick Ops** menu and the **Pick Ops** menu will disappear leaving the **Insert IP** option still current.

14.4.11.1.4 SA - Between IP

The **Between IP** option is similar to the Insert IP option (see 14.4.11.1.3 SA - Insert IP) except the inserted point does have to be on the line joining the two IPs. To accomplish this, the cursor position is automatically projected onto the IP-IP line to give the new IP position.

The **Between IP** option is terminated by selecting **Cancel** from the **Pick Ops** menu or by selecting a new option from the **Edit SA** toolbar.

If, after bringing up the **Pick Ops** menu, it is decided to continue with the **Between IP** option, simply select the **Restart** option from the **Pick Ops** menu and the **Pick Ops** menu will disappear leaving the **Between IP** option still current.

14.4.11.1.5 SA - Insert VIP Height

The **Insert VIP Height** option is used to insert a vertical intersection point (VIP) into the super alignment by clicking on the super alignment in either plan or section view to select the chainage that the VIP is to be inserted at, and then giving the height for the VIP.

So inserting the VIP is a two step process.

Step (a) - selecting the chainage for the VIP to inserted

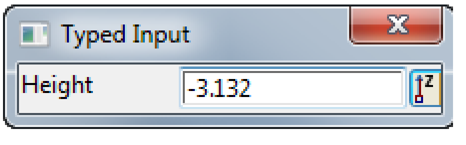
After selecting **Insert VIP Height**, a chainage to insert the new VIP at is selected by picking (LB) and accepting (MB) at that position on the super alignment.

For horizontal and vertical inserts

screen message area <Select point to add VIP> [picks][fast][Menu>

Step (b) - giving the height for the new VIP

After the chainage of the new VIP is selected, a *Height* Typed Input box is placed on the screen with the height of the super alignment at the selected chainage already displayed.



The required value is type into the *Height* Typed Input box and <Enter>.

The *Height* Typed Input box is removed from the screen and the new VIP inserted into the super alignment.

Once the Insert VIP cycle is completed and the VIP inserted, the **Insert VIP Height** is still current and can be repeated for other insertions (in either plan or section views) without having to re-select the **Insert VIP Height** option.

The **Insert VIP Height** option is terminated by selecting **Cancel** from the **Pick Ops** menu or by selecting a new option from the **Edit SA** toolbar.

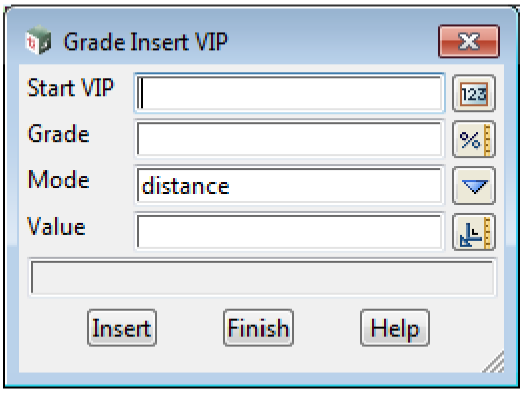
If, after bringing up the **Pick Ops** menu, it is decided to continue with the **Insert VIP Height** option, simply select the **Restart** option from the **Pick Ops** menu and the **Pick Ops** menu will disappear leaving the **Insert VIP Height** option still current.

14.4.11.1.6 SA - Insert VIP Grade

The **Insert VIP Grade** option inserts a vertical intersection point (VIP) at a given grade from an existing VIP.

The new VIP position is given by either a distance from the picked VIP or at given chainage on the super alignment.

After selecting **Insert VIP Grade**, the **Grade Insert VIP** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Start VIP	VIP number box		

the VIP number whose (chainage, height) position is used to start the Grade Insert from. The number can either be typed in, or if MB is clicked in the panel field, a VIP can be selected from a section view and its VIP number is piped into the Start VIP *field.*

Grade	measure x-fall box		

percent grade to be used.

Note ***positive*** *is* ***up*** *when going in the direction of* ***increasing chainage*** *and* ***negative*** *is* ***up*** *when going in the direction of* ***decreasing chainage****.*

Mode	choice	distance	distance, chainage

if ***distance****, the new VIP is inserted at the (chainage) distance given in the* Value *field from the start VIP, but with height determined by going with the given grade from the start VIP.*
If ***chainage****, the new VIP is inserted at the chainage given in the* value *field but with height determined by going with the given grade from the start VIP.*

Insert	button		

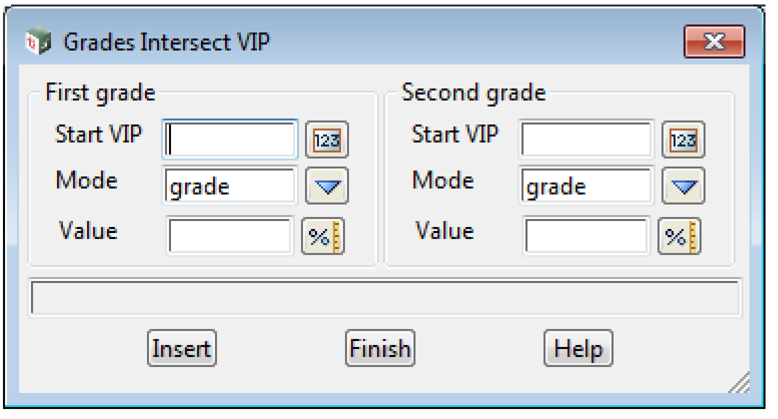
insert the new VIP into the alignment string.

14.4.11.1.7 SA - Intersect VIP Grades

The **Intersect VIP Grades** option inserts a VIP which is given by intersecting lines of given grades from two existing VIPs in the (chainage,height) plane.

The grades are either typed in or calculated by giving another VIP that the line goes through.

After selecting the option, the **Grades Intersect VIP** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
First Grade			
Start VIP	input		
Mode	choice	grade	grade, VIP #
Value or **End VIP**	input		

First Grade

Start VIP input

the VIP number whose (chainage,height) position is the start of a line.
The VIP number can either be typed in, or if MB is clicked in the panel field, a VIP can be selected from a section view and its VIP number is piped into the Start VIP *field.*

Mode choice grade grade, VIP #

*if **grade**, the field* Value *is a percent grade and a line is defined as going through the start VIP and with the given grade.*
*If **VIP #**, the field is* End VIP *and is the number of a VIP and a line is defined as going through the start VIP and this End VIP.*
Note for grades ***positive** is **up** when going in the direction of **increasing chainage** and*
***negative** is **up** when going in the direction of **decreasing chainage**.*

Value or **End VIP** input

if mode *is grade, the box is called* Value *and the value is a percent grade.*
if mode *is VIP #, the box is called* End VIP *and the value is the number of a VIP.*
For End VIP*, The VIP number can either be typed in, or if MB is clicked in the panel field, a VIP can be selected from a section view and its VIP number is piped into the* End VIP *field.*

Second Grade

Start VIP input

the VIP number whose (chainage,height) position is the start of a line.
The VIP number can either be typed in, or if MB is clicked in the panel field, a VIP can be selected from a section view and its VIP number is piped into the Start VIP *field.*

Mode choice grade grade, VIP #

*if **grade**, the field* Value *is a percent grade and a line is defined as going through the start VIP and with the given grade.*
*If **VIP #**, the field is* End VIP *and is the number of a VIP and a line is defined as going through the start VIP and this End VIP.*
Note for grades ***positive** is **up** when going in the direction of **increasing chainage** and*
***negative** is **up** when going in the direction of **decreasing chainage**.*

Value or **End VIP** input

if mode *is grade, the box is called* Value *and the value is a percent grade.*
if mode *is VIP #, the box is called* End VIP *and the value is the number of a VIP.*
For End VIP*, The VIP number can either be typed in, or if MB is clicked in the panel field, a VIP can be selected from a section view and its VIP number is piped into the* End VIP *field.*

Insert button

calculate in (chainage,height) space the intersection of the two lines given above and insert it as a new VIP into the super alignment.

14.4.11.1.8 SA - Filter IPs

The **Filter IPs** option tries to remove redundant HIPs and VIPs. That is, IP's that are very close to one another, or extra IPs on a straight line.

The extra IPs are often created when data is loaded from a CAD package and the elements are not tangential.

The **Filter IPs** option terminates after it is run and then a new option needs to be selected from the **Edit SA** toolbar.

14.4.11.1.9 SA - Delete IP

The **Delete IP** option is used to delete HIPs or VIPs points from the super alignment.

After picking **Delete IP**, any selected IP in the string is deleted. The string, minus the deleted IP, is redrawn after each deletion.

Once an IP has been deleted, another IP in the selected string can be deleted. Hence any number of the IPs can be deleted, one after another.

A HIP is deleted if the IP is chosen from a plan view and a VIP deleted if the IP is selected from a section view.

For HIP and VIP deletes

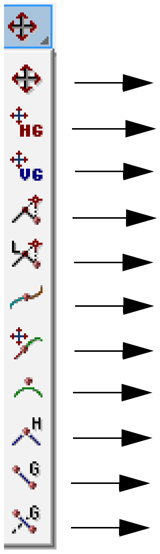
screen message area <Pick point to delete> [picks][fast][Menu>

The **Delete IP** option is terminated by selecting **Cancel** from the **Pick Ops** menu or by selecting a new option from the **Edit SA** toolbar.

If, after bringing up the **Pick Ops** menu, it is decided to continue with the **Delete IP** option, simply select the **Restart** option from the **Pick Ops** menu and the **Pick Ops** menu will disappear leaving the **Delete IP** option still current.

14.4.11.2 SA - Move/Edit Toolbar

The Move/Edit toolbar is



move IPs and Part definition points
move horizontal geometry parts
move vertical geometry parts
extend IP
extend by length
Tangent Wizard
move tangent point
change curve
change height
change grade
change grade 2

The **Move IP, Move Tangent** and **Extend IP** work on the horizontal geometry if used in a plan view and if used in a section view that the super alignment is profiled on, they work on the vertical geometry.

For the option *Move IP* go to	14.4.11.2.1 SA - Move
Move HG parts	14.4.11.2.2 SA - Move HG Parts
Move VG parts	14.4.11.2.3 SA - Move VG Parts
Extend IP	14.4.11.2.5 SA - Extend IP
Extend by length	14.4.11.2.6 SA - Extend by Length
Tangent Wizard	14.4.11.2.7 SA - Tangent Wizard
Move Tangent	14.4.11.2.4 SA - Move Tangent Point
Change curve	14.4.11.2.8 SA - Change Curve
Change height	14.4.11.2.9 SA - Change Height
Change grade	14.4.11.2.10 SA - Change Grade
Change grade 2	14.4.11.2.11 SA - Change Grade 2

14.4.11.2.1 SA - Move

The **Move** option allows the user to move

individual HIPs or Horizontal Parts definition points if the points are selected in a plan view

individual VIPs or Vertical Parts definition points if the points are selected in a section view that the super alignment is profiled on.

The **Move** cycle consists of two steps:

(a) selecting the point to be moved

(b) selecting the new position for the point.

Step (a)

First the point to be moved is selected. The selected point will then move around the view as the cursor is moved. The IPs/Parts recalculate and redraw as the point is moved.

screen message area <Pick point to move> [picks][fast][Menu>

Step (b)

The current cursor position is selected as the new position for the point by picking (LB) and accepting (MB). The point being moved is then anchored at the cursor position and the string redrawn.

screen message area <Pick final position of point> [picks][fast][Menu>

Once the **Move** cycle is completed and the point moved, the **Move** option is still current and can be repeated for other points without having to re-select the **Move** option.

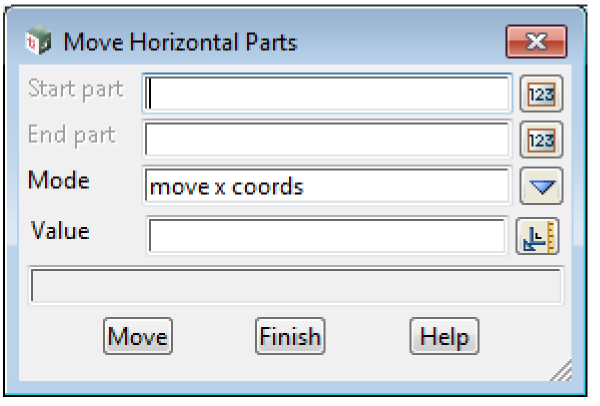
The **Move** option is terminated by selecting **Cancel** from the **Pick Ops** menu or by selecting a new option from the **Edit SA** toolbar.

If, after bringing up the **Pick Ops** menu, it is decided to continue with the **Move** option, simply select the **Restart** option from the **Pick Ops** menu and the **Pick Ops** menu will disappear leaving the **Move** option still current.

14.4.11.2.2 SA - Move HG Parts

Move HG Parts translates the X or the Y coordinates for a given range of Horizontal Parts in the super alignment being edited.

Clicking on the **Move HG Parts** icon brings up the **Move Horizontal Parts** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Start part	input box		

the number of the Part in the Horizontal Geometry to start translating in x or y. Clicking MB in the field or RB on the number icon brings up the **Same as** *picker and a Horizontal Part from the super alignment can be selected from a plan view and the Part Number will be piped into the field.*

End part input box

the number of the Part in the Horizontal Geometry to end the translating in x or y. Clicking MB in the field or RB on the number icon brings up the **Same as** *picker and a Horizontal Part from the super alignment can be selected from a plan view and the Part Number will be piped into the field.*

Mode choice box move x coords, move y coords

if Mode *is* move x coords, *the x coordinates of all the parts from the* Start part *to* End part *are translated by* Value.
if Mode *is* move y coords, *the y coordinates of all the parts from the* Start part *to* End part *are translated by* Value.

Value measure box Measure X or Measure Y

the distance to move the coordinates.

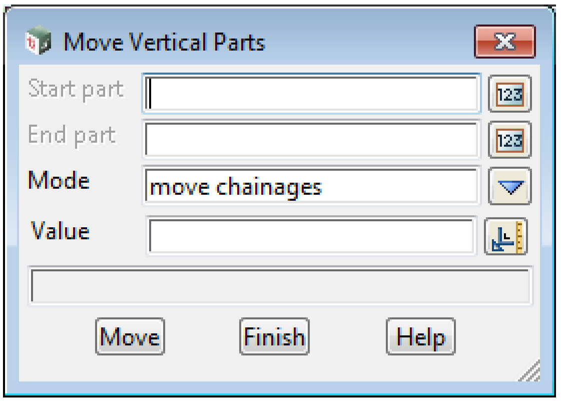
Move button

move the specified coordinates for the given range of Horizontal Parts.

14.4.11.2.3 SA - Move VG Parts

Move VG Parts translates the chainages or the heights for a given range of Vertical Parts in the super alignment being edited.

Clicking on the **Move VG Parts** icon brings up the **Move Vertical Parts** panel.



The fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

Start part input box

the number of the Part in the Vertical Geometry to start translating in chainage or height. Clicking MB in the field or RB on the Number icon brings up the **Same as** *picker and a Vertical Part from the super alignment can be selected from a section view, and the Part Number piped into the field.*

End part input box

the number of the Part in the Vertical Geometry to end the translating in chainage or height. Clicking MB in the field or RB on the Number icon brings up the **Same as** *picker and a Vertical Part from the super alignment can be selected from a section view and the Part Number will be piped into the field.*

Mode choice box move x coords, move y coords

if Mode *is* move chainages*, the chainage coordinates of all the parts from the* Start part *to* End part *are translated by* Value*.*
if Mode *is* move heights*, the height coordinates of all the parts from the* Start part *to* End part *are translated by* Value*.*

Value measure box Measure Chainage or Measure Z

the distance to move the coordinates

Move button

move the specified coordinates for the given range of Vertical Parts

14.4.11.2.4 SA - Move Tangent Point

The **Move Tangent** option allows the user to move

individual horizontal tangent points in a plan view if the horizontal tangent point (HTP) is part of an HIP with a curve on it but no approaching or departing transitions.

individual vertical tangent points in a section view that the super alignment is profiled on if the vertical tangent point (VTP) is part of a VIP with a parabola or curve on it.

The **Move Tangent** cycle consists of two steps:

(a) selecting the tangent point (TP) to be moved

(b) selecting the new position for the TP.

Step (a)

First the tangent point to be moved is selected. The selected point will then move around the view as the cursor is moved.

If a horizontal tangent point is selected, the string will be redrawn with the HTP moved to the cursor position and the circular curve radius will be modified appropriately.

If a vertical tangent point is selected, the string will be redrawn with the VTP moved to the cursor position and the vertical curve modified appropriately.

screen message area <Pick tangent to extend> [picks][fast][Menu>

Step (b)

The current cursor position is selected as the new position for the point by picking (LB) and accepting (MB). The tangent being moved is then anchored at the cursor position and the string redrawn.

screen message area <Pick final position of tangent> [picks][fast][Menu>

Once the Move Tangent cycle is completed and the TP moved, the **Move Tangent** option is still current and can be repeated for other tangent points without having to re-select the **Move Tangent** option.

The **Move Tangent** option is terminated by selecting **Cancel** from the **Pick Ops** menu or by selecting a new option from the **Edit SA** toolbar.

If, after bringing up the **Pick Ops** menu, it is decided to continue with the **Move Tangent** option, simply select the **Restart** option from the **Pick Ops** menu and the **Pick Ops** menu will disappear leaving the **Move Tangent** option still current.

14.4.11.2.5 SA - Extend IP

The **Extend IP** option is used to move an IP along the line joining the IP to its neighbouring IP.

That is, in a plan view, the bearing of the HIP-HIP line is kept constant and the intersection point is moved along that line either towards or away from its neighbouring intersection point on the HIP-HIP line.

In a section view, the grade of the VIP-VIP line is kept constant and the vertical intersection point is moved along that line either towards or away from its neighbouring intersection point on the VIP-VIP line.

Extending, like moving an IP, is a two step process.

Step (a) - selecting the IP-IP line and the IP to be moved along that line

The IP-IP line and the intersection point to be moved are chosen in the one operation by selecting a position near the IP-IP line and close to the intersection point to be moved along that line.

screen message area <Pick point to extend> [picks][fast][Menu>

Once the line and IP are selected, the new position of the selected IP is assumed to be at the current cursor position projected perpendicularly onto the IP-IP line. As the cursor is moved, the string is redrawn reflecting the changing position of the moved IP.

Step (b) - selecting the final position for the IP

The current cursor position is selected as the new position for the IP by picking (LB) and accepting (MB). The final position for the intersection point is set to the projection of the selected cursor position onto the IP-IP line.

screen message area <Pick final position of point> [picks][fast][Menu>

Once the **Extend IP** is completed, the extend option is still current and can be repeated without re-selecting the **Extend IP** option.

The **Extend IP** option is terminated by selecting **Cancel** from the **Pick Ops** menu or by selecting a new option from the **Edit SA** toolbar.

If, after bringing up the **Pick Ops** menu, it is decided to continue with the **Extend IP** option, simply select the **Restart** option from the **Pick Ops** menu and the **Pick Ops** menu will disappear leaving the **Extend IP** option still current.

Note - **Extend IP** can be used on the end points of the horizontal or vertical geometry of a super alignment string.

14.4.11.2.6 SA - Extend by Length

The **Extend by length** option is used to move a horizontal IP or vertical IP by a given length along the line joining the IP to its neighbouring IP.

A horizontal IP is modified in a plan view and the bearing of the HIP-HIP line is kept constant and the intersection point is moved by a given length along that line either towards or away from its neighbouring intersection point on the HIP-HIP line.

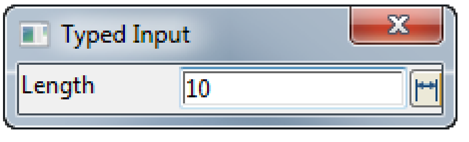
A vertical IP is modified in a section view and the grade of the VIP-VIP line is kept constant and the vertical intersection point is moved along that line either towards or away from its neighbouring intersection point on the VIP-VIP line.

To modify a horizontal IP, the string is edited in a plan view. To modify a vertical IP, the string is edited in a section view.

The IP-IP line and the intersection point to be moved are chosen in the one operation by selecting a position near the IP-IP line and close to the intersection point to be moved along that line.

screen message area <[F]orward: pick IP to extend, (or [B]ackward)> [picks][fast][Menu]

Once the line and IP are selected, a **Length Typed Input** box is displayed and the required length typed into the box and <Enter> pressed.



The IP then moves to its new position.

Note: When in Forward mode, the IP moves along the selected segment. When in Backward mode, the IP moves along the segment on the other side of the selected IP.

The default is Forward. To select Backward mode, type ’B’. To select Forward mode, type ’F’.

Once the **Extend by length** is completed, the extend option is still current and can be repeated without re-selecting the **Extend by length** option.

The **Extend by length** option is terminated by selecting **Cancel** from the **Pick Ops** menu or by selecting a new option from the **Edit SA** toolbar.

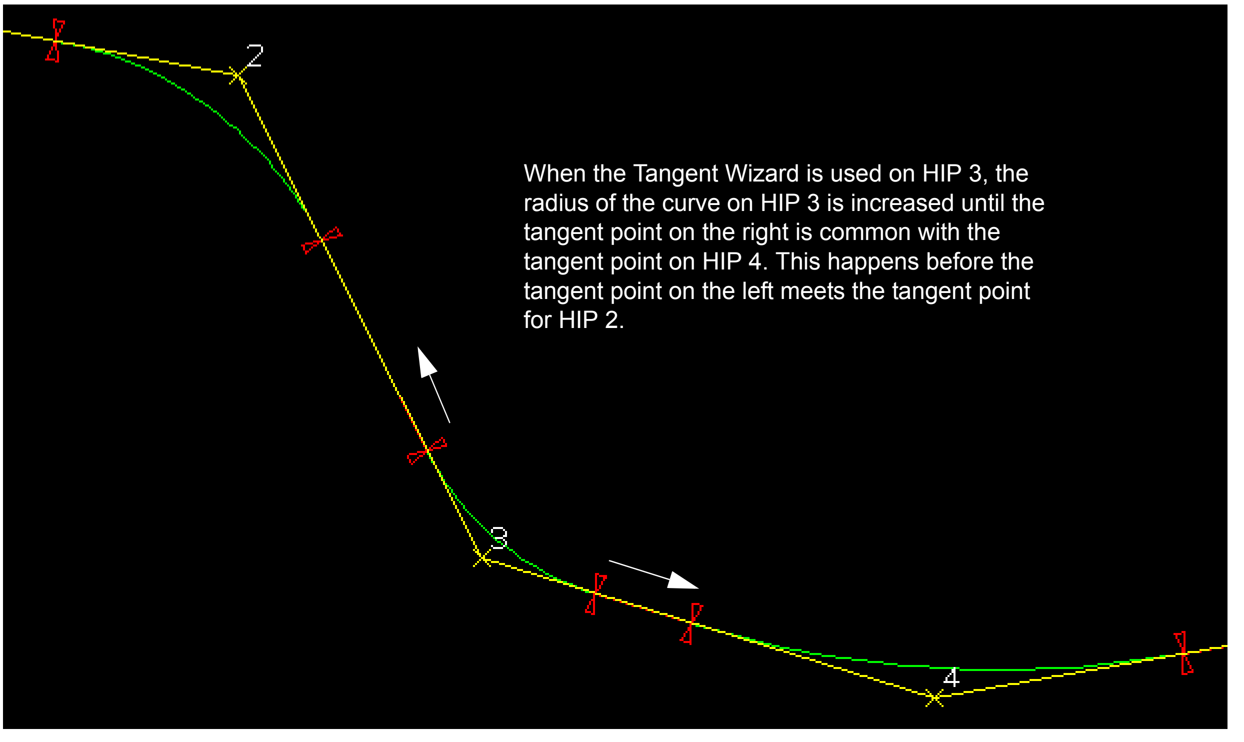
If, after bringing up the **Pick Ops** menu, it is decided to continue with the **Extend by length** option, simply select the **Restart** option from the **Pick Ops** menu and the **Pick Ops** menu will disappear leaving the **Extend by length** option still current.

Extend by length can be used on the end points of the horizontal or vertical geometry of a super alignment string.

14.4.11.2.7 SA - Tangent Wizard

The **Tangent Wizard** works for a HIP defined with just an arc on it and no transitions, or a VIP with a parabola or curve defined on it.

If a suitable HIP or VIP is selected, the **Tangent Wizard** modifies the curve/parabola on the selected HIP/VIP so that a common curve-curve tangent point is created on at least one side of the HIP/VIP.



When the **Tangent Wizard** run successfully, the **Tangent Wizard** option is still current and can be repeated without re-selecting **Tangent Wizard**. If an incorrect IP is selected, the **Tangent Wizard** terminates and an option needs to be selected from the **Edit SA** toolbar.

The **Tangent Wizard** option is terminated by selecting **Cancel** from the **Pick Ops** menu or by selecting a new option from the **Edit SA** toolbar.

If, after bringing up the **Pick Ops** menu, it is decided to continue with the **Tangent Wizard** option, simply select the **Restart** option from the **Pick Ops** menu and the **Pick Ops** menu will disappear leaving the **Tangent Wizard** option still current.

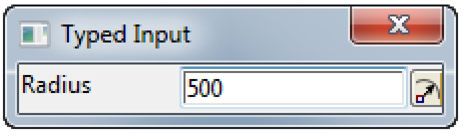
14.4.11.2.8 SA - Change Curve

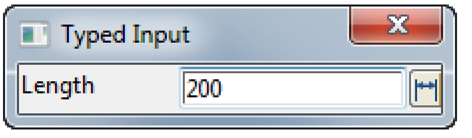
The **Change Curve** option is used to change the defining value for the horizontal curve (radius or length) of a curve on a horizontal intersection point, or the defining value for the vertical curve (length, k-value or radius) on a vertical intersection point.

After selecting **Change Curve**:

screen message area <Pick IP to change curve info> [picks][fast][Menu>

If an HIP is selected from a plan view, the value defining the horizontal curve (radius or length) on the HIP is displayed.

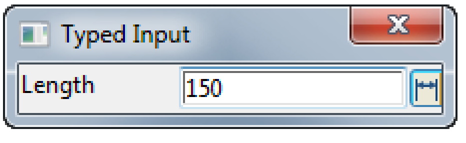




The required Radius (or Length) is typed into the Typed Input box and <Enter> pressed.

The HIP is given the new value and the Typed Input box is removed from the screen.

If an VIP is selected from a section view that the super alignment is profiled on, the value defining the vertical curve (length, k-value or length) on the VIP is displayed.



The required Length (or k value or Length) is typed into the Typed Input box and <Enter> pressed.

The VIP is given the new value and the Typed Input box is removed from the screen.

Once the curve value is changed, the **Change Curve** is still current and can be repeated for other IPs in the super alignment without having to re-select the **Change Curve** option.

The **Change Curve** option is terminated by selecting **Cancel** from the **Pick Ops** menu or by selecting a new option from the **Edit SA** toolbar.

If, after bringing up the **Pick Ops** menu, it is decided to continue with the **Change Curve** option, simply select the **Restart** option from the **Pick Ops** menu and the **Pick Ops** menu will disappear leaving the **Change Curve** option still current.

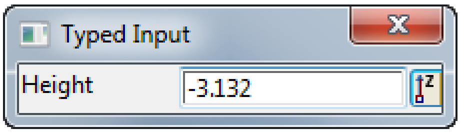
14.4.11.2.9 SA - Change Height

The **Change Height** option is used to change the height of an existing vertical intersection point (VIP).

After selecting **Change Height**, a VIP is selected from a section view that the super alignment is profiled on:

screen message area <Pick IP to change height> [picks][fast][Menu>

After the VIP is selected, a ***Height*** Typed Input box is placed on the screen with the height of the selected VIP already displayed.



The required height is typed into the *Height* Typed Input box and <Enter> pressed.

The *Height* Typed Input box is removed from the screen and the VIP is given the new height.

Once the height is changed, the **Change Height** is still current and can be repeated for other VIPs in the super alignment without having to re-select the **Change Height** option.

The **Change Height** option is terminated by selecting **Cancel** from the **Pick Ops** menu or by selecting a new option from the **Edit SA** toolbar.

If, after bringing up the **Pick Ops** menu, it is decided to continue with the **Change Height** option, simply select the **Restart** option from the **Pick Ops** menu and the **Pick Ops** menu will disappear leaving the **Change Height** option still current.

14.4.11.2.10 SA - Change Grade

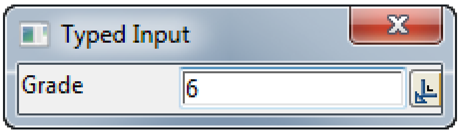
The **Change Grade** option is used to change the grade of the line between two consecutive vertical intersection points (VIPs) by moving the VIP at the selected end of the line up or down to achieve the given grade.

Note that the chainage position of the VIP remains the same and the grade on both sides of the moved VIP change.

After selecting **Change Grade**, the line joining consecutive VIP is picked with direction from a section view that the super alignment is profiled on

screen message area <Pick grade to change> [picks][fast][Menu>

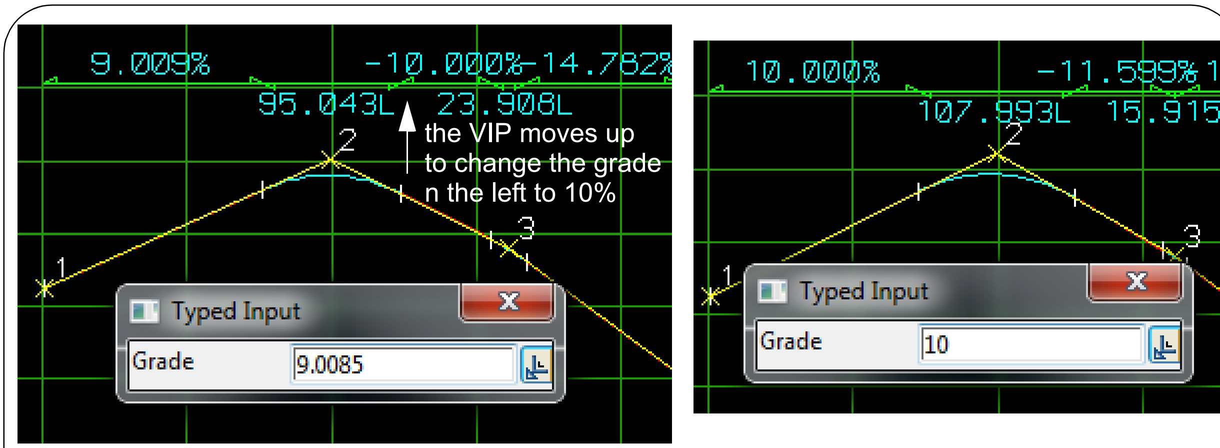
After the VIP is selected, a ***Grade*** Typed Input box is placed on the screen with the grade of the selected line, measured from left to right, is already displayed.



Note: going from left to right in the section view (that is, increasing chainage), positive grade is up and negative grade is down.

If the line was picked with direction going from left to right in the section view (increasing chainage), then the position of the VIP at the left end of the line is kept constant and the line is graded with the new grade and the height of the VIP at the right end of the line is modified so that it sits on the new graded line. The chainage of the VIP is kept constant. That is, the VIP just moves up or down until the new grade is reached.

The *Grade* Typed Input box is removed from the screen.



Change Grade: The line going from VIP 1 to VIP 2 with direction going from left to right.
VIP 2 is moved up to give the new grade for the line between VIP 1 and VIP 2

If the line was picked with direction going from right to left in the section view (decreasing chainage), then the position of the VIP at the right end of the line is kept constant and the line is graded with the new grade and the height of the VIP at the left end of the line is modified so that it sits on the new graded line. The chainage of the VIP is kept constant. That is, the VIP just moves up or down until the new grade is reached.

The *Grade* Typed Input box is removed from the screen.

Once the grade is changed, the **Change Grade** is still current and can be repeated for other lines between VIPs in the super alignment without having to re-select the **Change Grade** option.

The **Change Grade** option is terminated by selecting **Cancel** from the **Pick Ops** menu or by selecting a new option from the **Edit SA** toolbar.

If, after bringing up the **Pick Ops** menu, it is decided to continue with the **Change Grade** option, simply select the **Restart** option from the **Pick Ops** menu and the **Pick Ops** menu will disappear leaving the **Change Grade** option still current.

14.4.11.2.11 SA - Change Grade 2

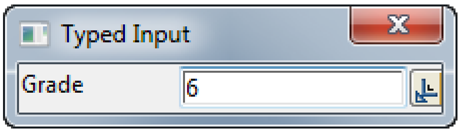
The **Change Grade 2** option is used to change the grade of the line between two consecutive vertical intersection points (VIPs) by sliding the VIP at the selected end of the line along the VIP-VIP line on the other side of the VIP so that the grade of the other line is not modified.

Note that the chainage position of the VIP changes and only the grade on the selected line of the moved VIP changes.

After selecting **Change Grade**, the line joining consecutive VIP is picked with direction from a section view that the super alignment is profiled on

screen message area <Pick grade to change> [picks][fast][Menu>

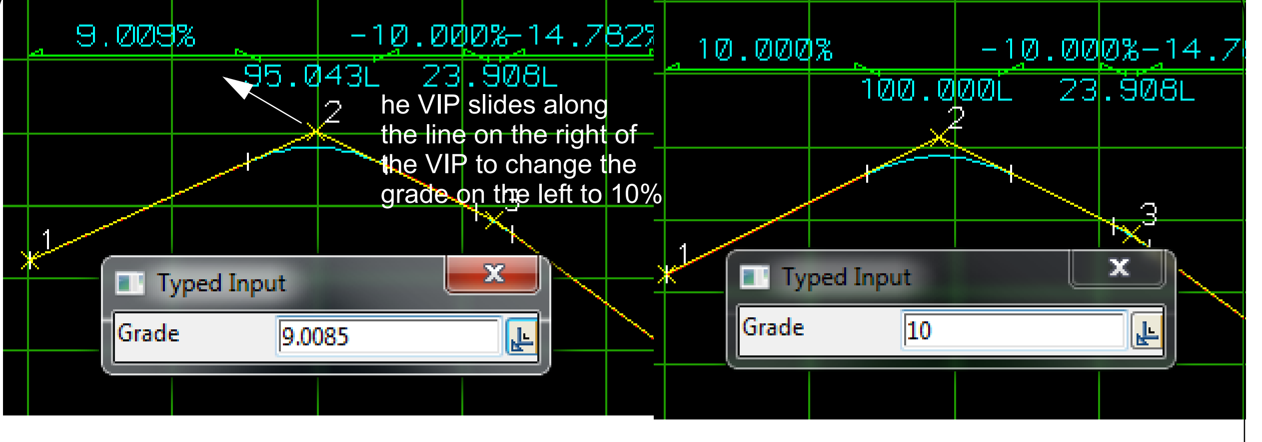
After the VIP is selected, a ***Grade*** Typed Input box is placed on the screen with the grade of the selected line, measured from left to right, is already displayed.



Note: going from left to right in the section view (that is, increasing chainage), positive grade is up and negative grade is down.

If the line was picked with direction going from left to right in the section view (increasing chainage), then the position of the VIP at the left end of the line is kept constant and the line is graded with the new grade by sliding the VIP at the right end of the line along the VIP-VIP line on the other side of the VIP to keep that grade constant. So the grade on the other side of the VIP is kept constant. That is, the VIP slides up or down the other line until the new grade is reached.

The *Grade* Typed Input box is removed from the screen.



Change Grade 2: The line going from VIP 1 to VIP 2 with direction going from left to right.
VIP 2 is slid along the line between VIP2 and VIP 3 (keeping its grade constant)
to give the new grade for the line between VIP 1 and VIP 2

If the line was picked with direction going from right to left in the section view (decreasing chainage), then the position of the VIP at the right end of the line is kept constant and the line is graded to the new grade by sliding the VIP at the left end of the line along the VIP-VIP line on the other side of the VIP to keep that grade constant. So the grade on the other side of the VIP is kept constant. That is, the VIP slides up or down the other VIP-VIP line until the new grade is reached.

The *Grade* Typed Input box is removed from the screen.

Once the grade is changed, the **Change Grade 2** is still current and can be repeated for other lines between VIPs in the super alignment without having to re-select the **Change Grade 2** option.

The **Change Grade 2** option is terminated by selecting **Cancel** from the **Pick Ops** menu or by selecting a new option from the **Edit SA** toolbar.

If, after bringing up the **Pick Ops** menu, it is decided to continue with the **Change Grade 2** option, simply select the **Restart** option from the **Pick Ops** menu and the **Pick Ops** menu will disappear leaving the **Change Grade 2** option still current.

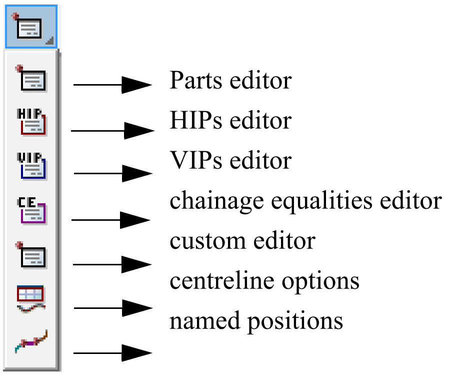
14.4.11.3 SA - Change Icon

The Change icon is

Selecting this icon allows the user to change the radius of arcs placed by Parts.

14.4.11.4 SA - Part Editors Toolbar

The Part Editors toolbar is



For the option *Parts Editor*, go to [14.4.11.4.1 SA - Parts Editor](#)
HIPs Editor [14.4.11.4.2 SA - HIPs Editor](#)
VIPs Editor [14.4.11.4.3 SA - VIPs Editor](#)
Chainage Equalities Editor [14.4.11.4.4 SA - Chainage Equalities Editor](#)
Custom Editor [14.4.11.4.5 SA - Custom Superelevation and Widening Editor](#)
Centreline Options [14.4.11.4.6 SA - Centreline Options](#)
Named Positions [14.4.11.4.7 SA - Named Positions](#)

14.4.11.4.1 SA - Parts Editor

Placing a super alignment consists of:

(a) defining the horizontal geometry consisting of lines, arcs, transitions and tapers

(b) defining the vertical geometry consisting of lines, parabolas and arcs.

A simple way to create horizontal and vertical geometry for the super alignment is by using horizontal intersection points (HIPs) with arcs and transitions (eg spirals), and vertical intersection points (VIP's) with parabolas or arcs for placing the vertical geometry.

However the super alignment also allows for more complex construction methods to define the lines, arcs and transitions (the horizontal elements) that within ***12d Model*** are be grouped together to make up the Parts of the horizontal geometry.

For example, a horizontal line is defined to be a given parallel offset of a selected line segment from another string. Or a more complex part such *Free Arc with Known Radius* that consists of the three elements: an arc of a known radius plus a leading and a trailing transition.

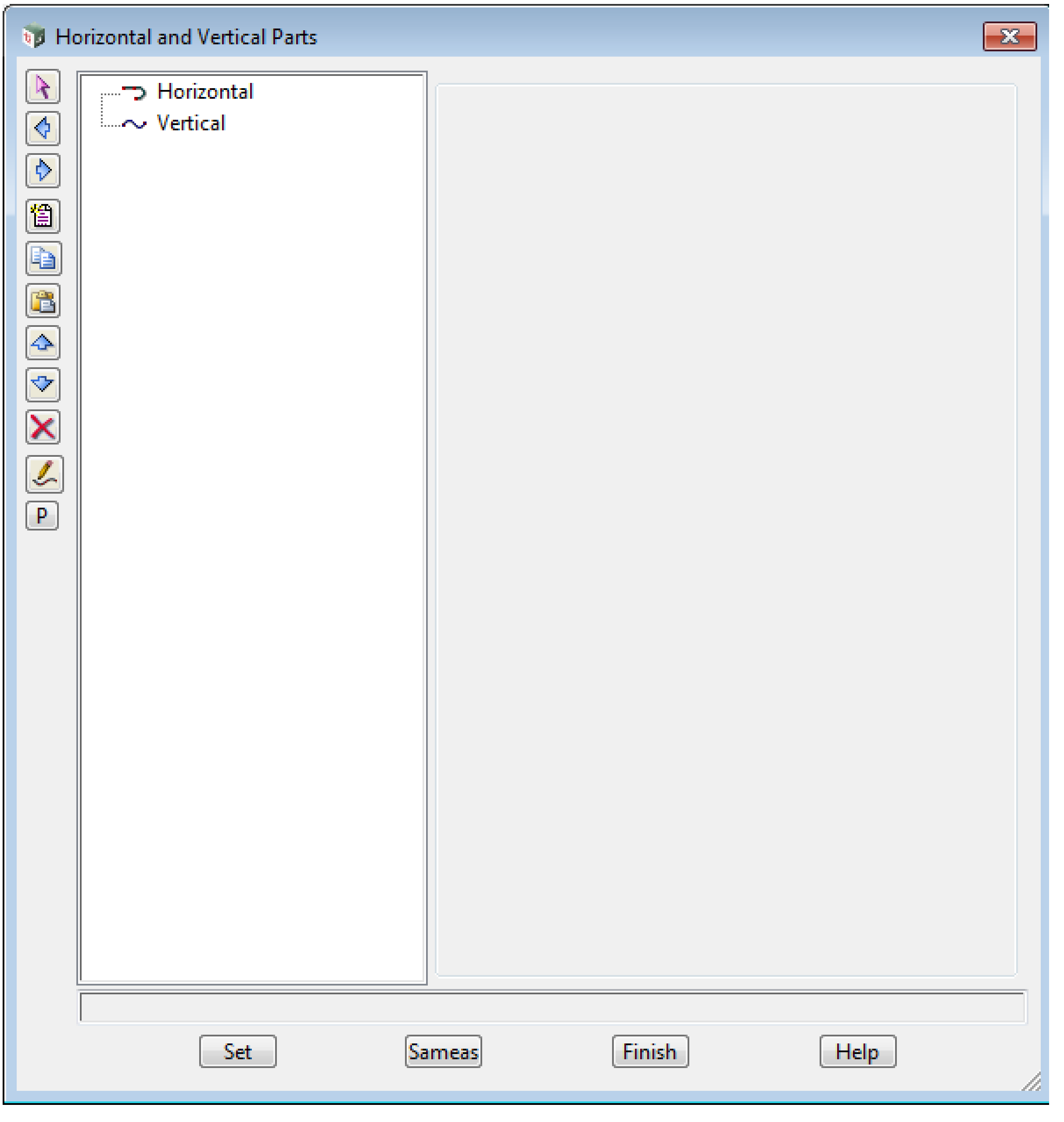
Similarly more complex construction methods are available to define the lines, parabolas and arcs (the vertical elements) that are grouped together within***12d Model*** to make up the Parts of the vertical geometry.

Some of the Parts definitions have some degree of freedom in the definition of the elements in the Parts, and the sequencing and types of the Parts, plus the condition that, if possible, each element is tangential to the adjacent elements, allows *12d Model* to calculate and solve for some unknowns in the Part definitions that make up the super alignment.

For example, if a *Free Arc with Known Radius* placed between two known lines then the tangentiality condition means that the position of the *Free Arc with Known Radius* (arc and two transitions) is uniquely defined as a arc and leading and trailing transitions on the intersection point of the two lines. Or just a fillet between the two lines if the two transitions have zero length.

For more information on placing parts, fixed and floating parts and solving, go to 21.9 Placing Parts for Super Alignments.

The general creating and editing of the horizontal and vertical Parts inside ***12d Model*** is done in the **Horizontal and Vertical Parts** panel and this is brought up by clicking on the on the **Parts Editor** icon.

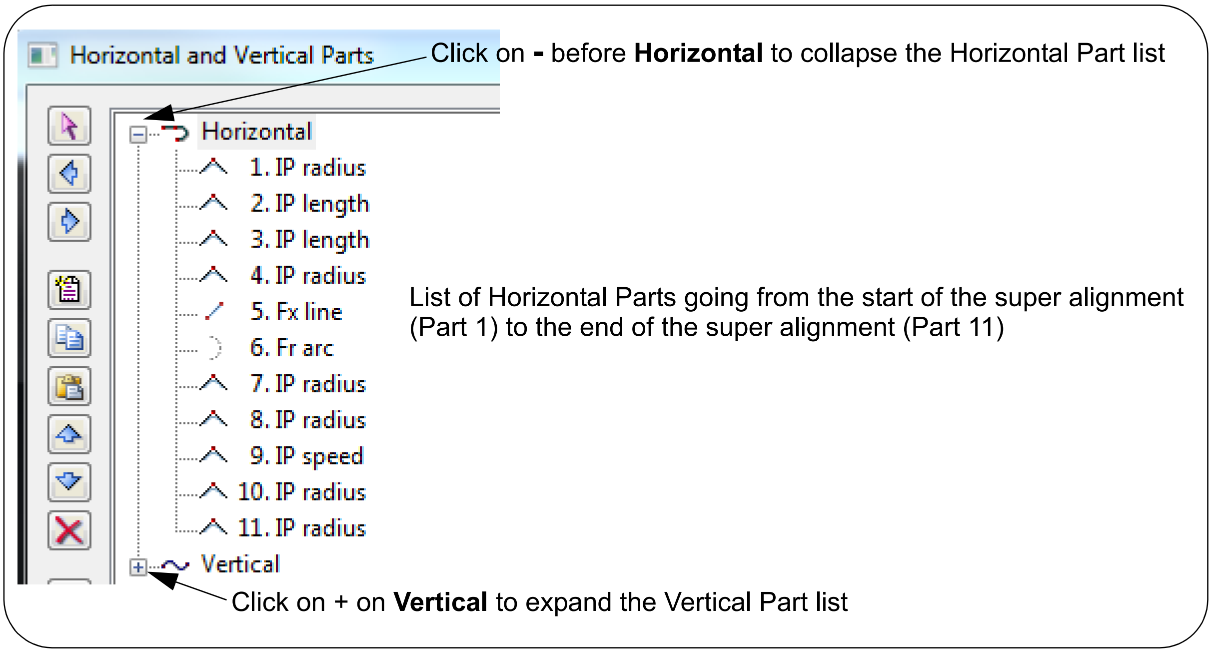


See 14.4.11.4.1.1 Icons on the Horizontal and Vertical Parts Panel
See 14.4.11.4.1.2 Creating and Editing a Part
See 14.4.11.4.1.3 Types of Horizontal Parts
See 14.4.11.4.1.4 Types of Vertical Parts

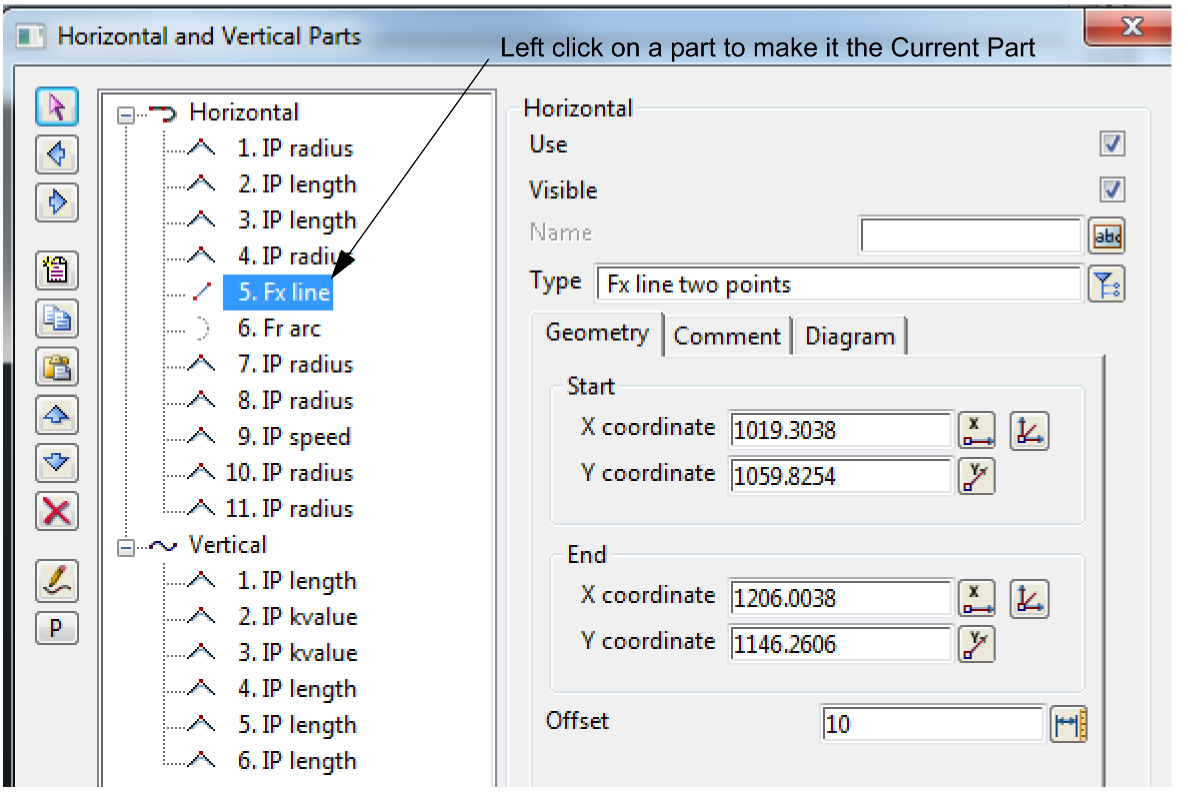
14.4.11.4.1.1 Icons on the Horizontal and Vertical Parts Panel

If there are any existing horizontal parts in the super alignment, then a + will appear to the left of Horizontal in the tree. Similarly if there are any existing vertical parts then a + will appear next to Vertical in the tree.

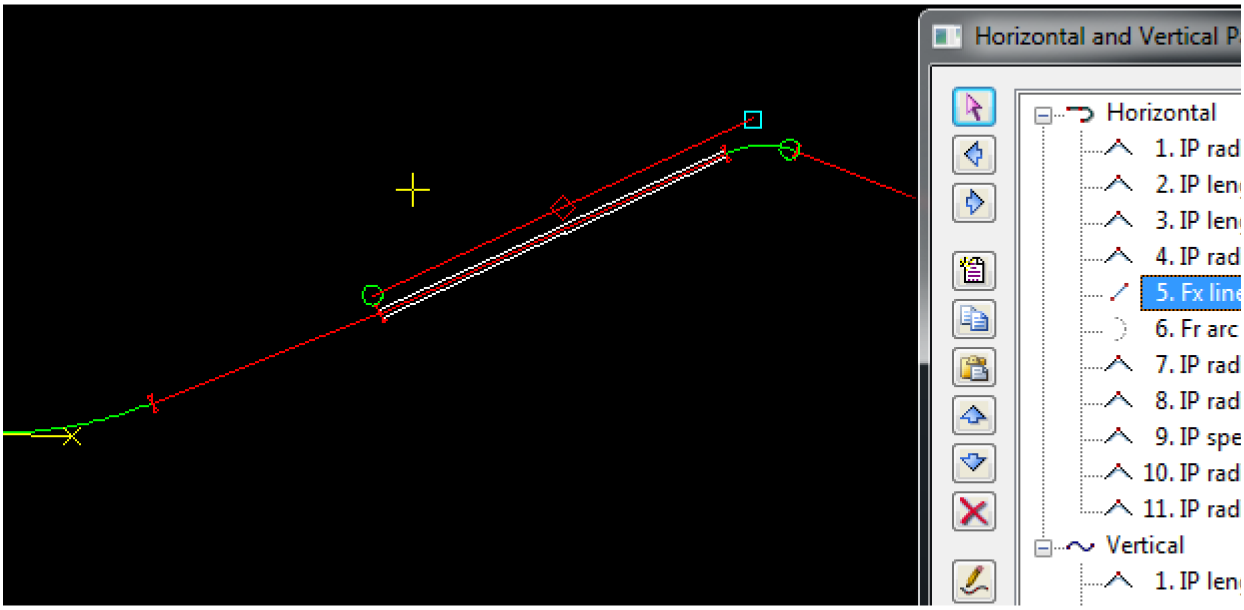
Clicking on the **+** next to Horizontal/Vertical will then list all the horizontal/vertical parts in order from the start of the super alignment to the end of the super alignment.

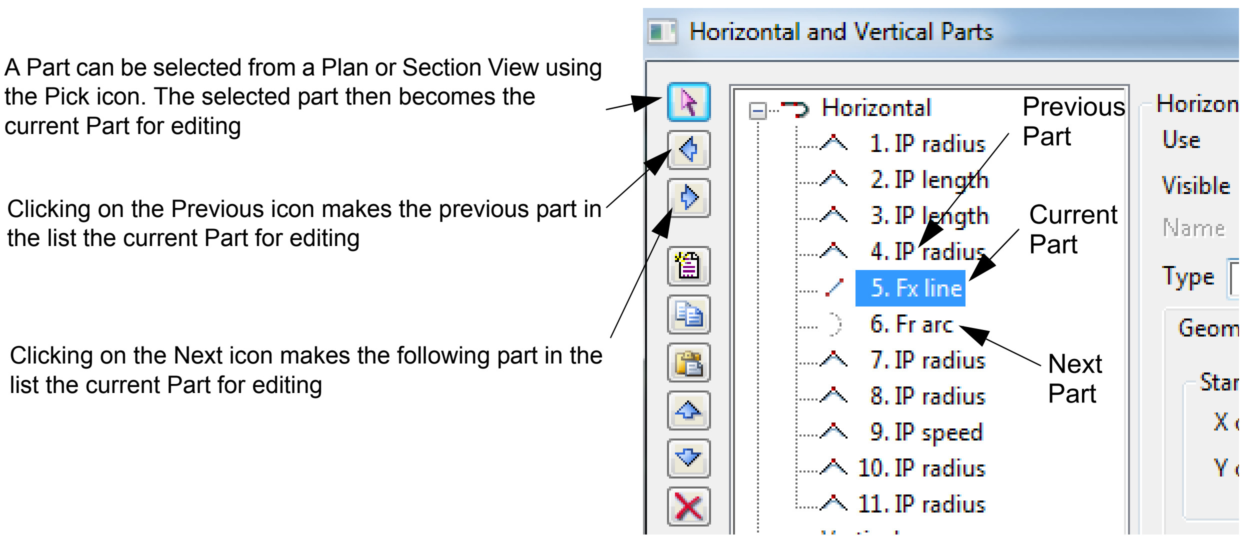


To make a Part the current Part being edited, simply Left click on the Part in the expanded list and the selected Part will be highlighted and the information about the part displayed on the right hand side of the panel.

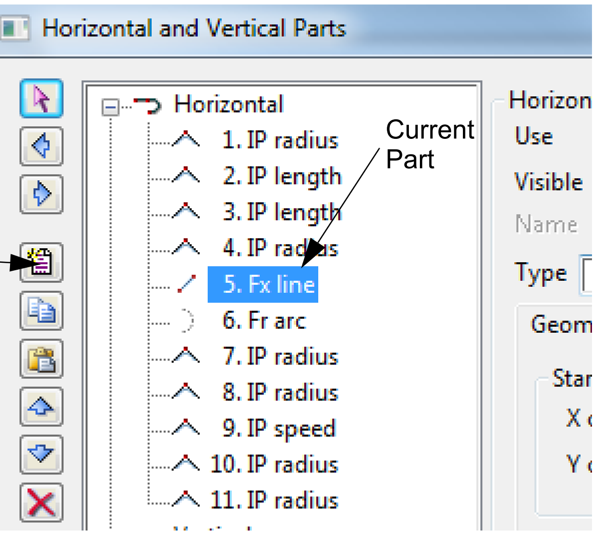


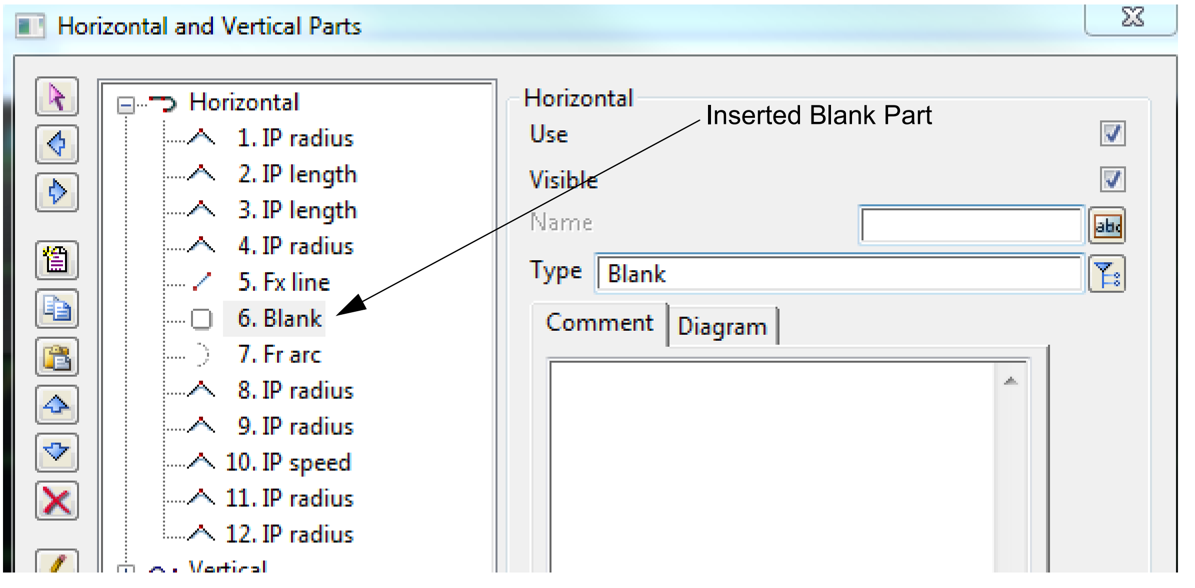
The selected part will also highlight in any Plan views the super alignment is on if it is a Horizontal Part, or any Section views that the super alignment is profiled on if it is a Vertical Part.





A new Blank Part is inserted after the current Part by clicking on the Insert icon.

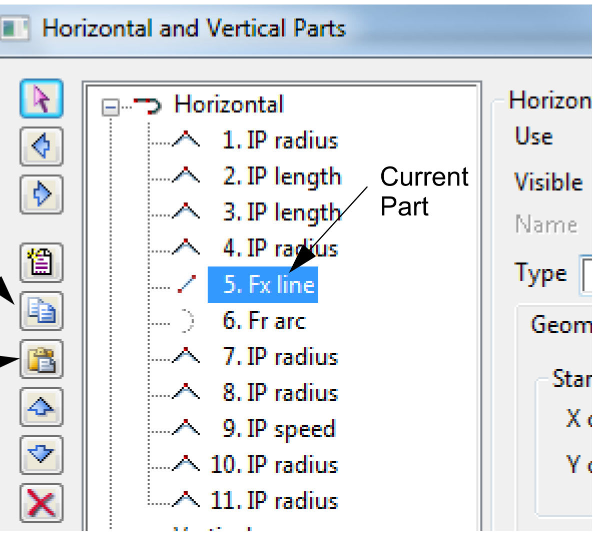




The current Part can be copied by clicking on the Copy icon.

The part that the copied Part is to be inserted after is then made the current Part (by Picking, clicking on the Part in the Editor, *etc.*)

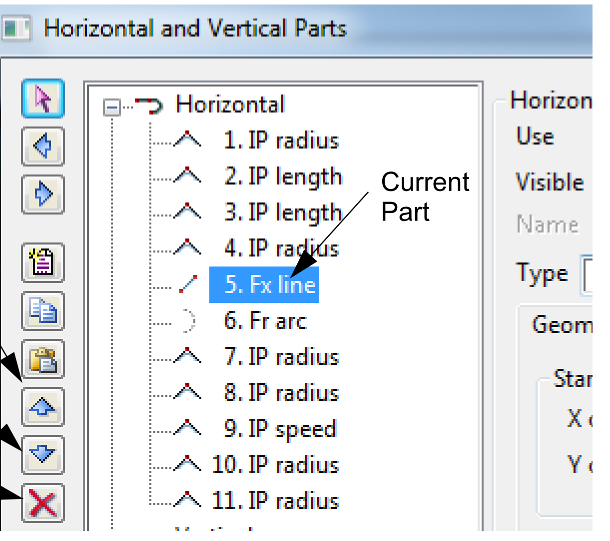
And clicking on the Paste icon then inserts the copied part after the current Part.



The current Part is moved one place closer to the start of the list by clicking on the Move up icon.

The current Part is moved one place down towards the bottom of the list by clicking on the Move down icon.

The current Part is deleted by clicking on the Delete icon.

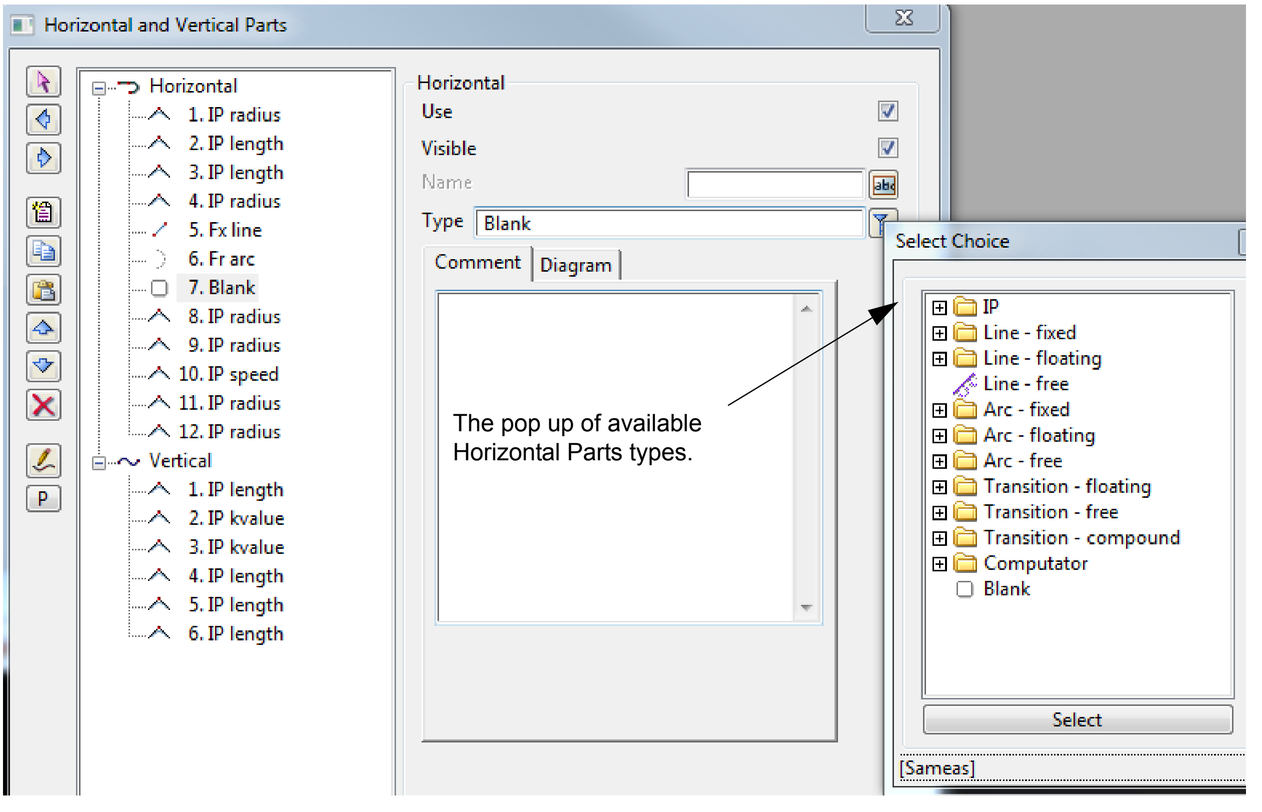


14.4.11.4.1.2 Creating and Editing a Part

A new Part is created by Inserting a new Blank Part and then clicking on the Type choice box to get the list of available Parts. After choosing a Part, the information required to define the selected part is displayed on the right hand side of the **Horizontal and Vertical Parts** panel. The required information is filled in and then the **Set** button at the bottom of the panel is clicked to create the new Part.

An existing Part can be edited by selecting it to make it the current Part which will display its defining information on the right hand side of the **Horizontal and Vertical Parts** panel. Any required changes to the information are made and then click on the **Set** button at the bottom of the panel to update the Part.

An existing Part can be changed to a different type of Part by selecting it to make it the current Part, then click on the Type choice box and select the new Type for the Part from the pop-up list.



The list of the available Horizontal and Vertical Parts for Type are given below with links to the full definition of each Part.

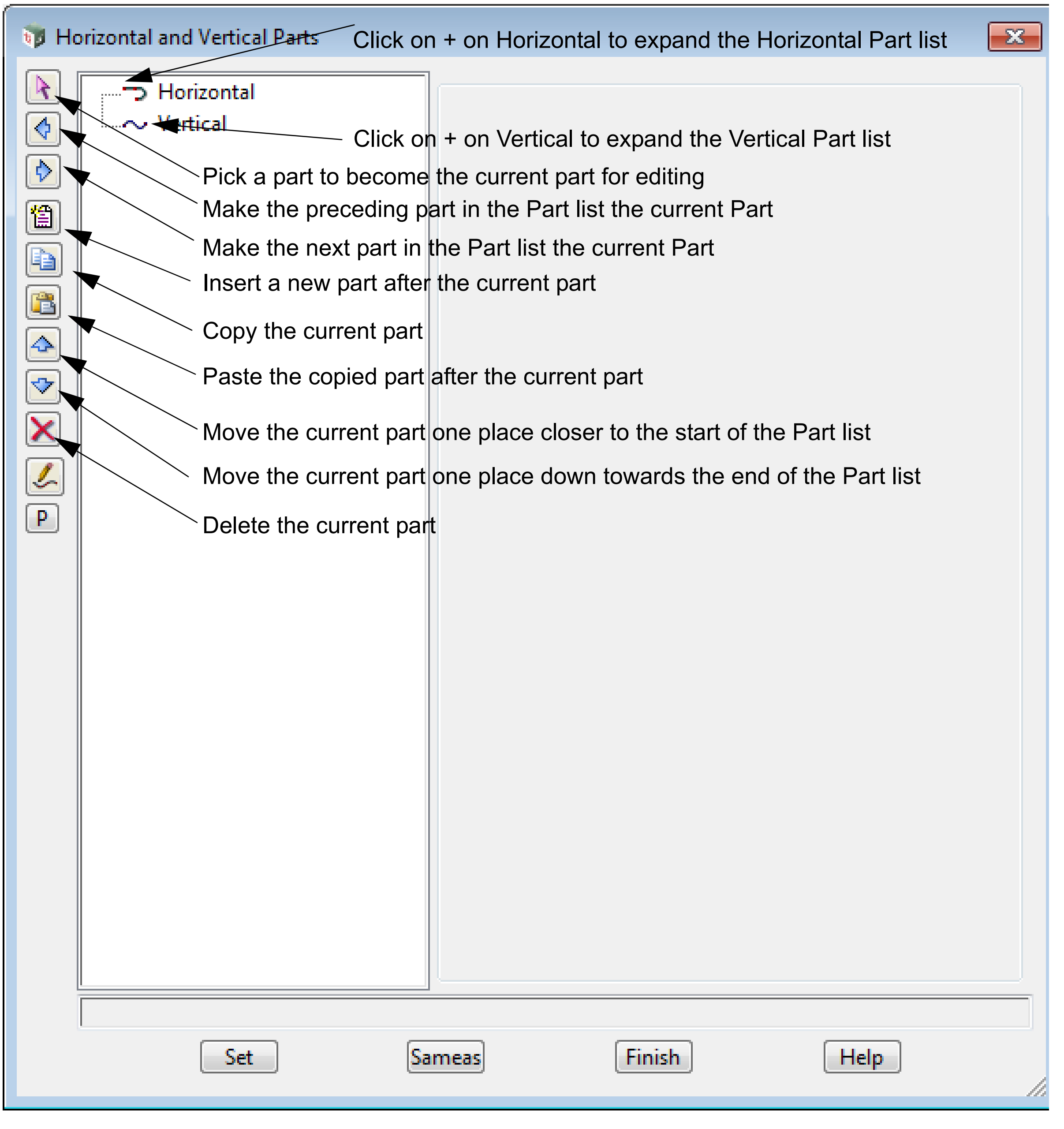
14.4.11.4.1.3 Types of Horizontal Parts

See 21.9.2.1 Horizontal IPs
See 21.9.2.2 Horizontal Lines - Fixed
See 21.9.2.3 Horizontal Lines - Floating
See 21.9.2.4 Horizontal Lines - Free
See 21.9.2.5 Horizontal Arcs - Fixed
See 21.9.2.6 Horizontal Arcs - Floating
See 21.9.2.7 Horizontal Arcs - Free
See 21.9.2.8 Horizontal Transitions - Floating
See 21.9.2.9 Horizontal Transitions - Free
See 21.9.2.10 Horizontal Transitions - Compound

14.4.11.4.1.4 Types of Vertical Parts

See 21.9.3.1 Vertical IPs
See 21.9.3.2 Vertical Lines - Fixed

See 21.9.3.3 Vertical Lines - Floating
See 21.9.3.4 Vertical Lines - Free
See 21.9.3.5 Vertical Parabolas - Fixed
See 21.9.3.6 Vertical Parabolas - Floating
See 21.9.3.7 Vertical Parabolas - Free
See 21.9.3.8 Vertical Arcs - Fixed
See 21.9.3.9 Vertical Arcs - Floating
See 21.9.3.10 Vertical Arcs - Free

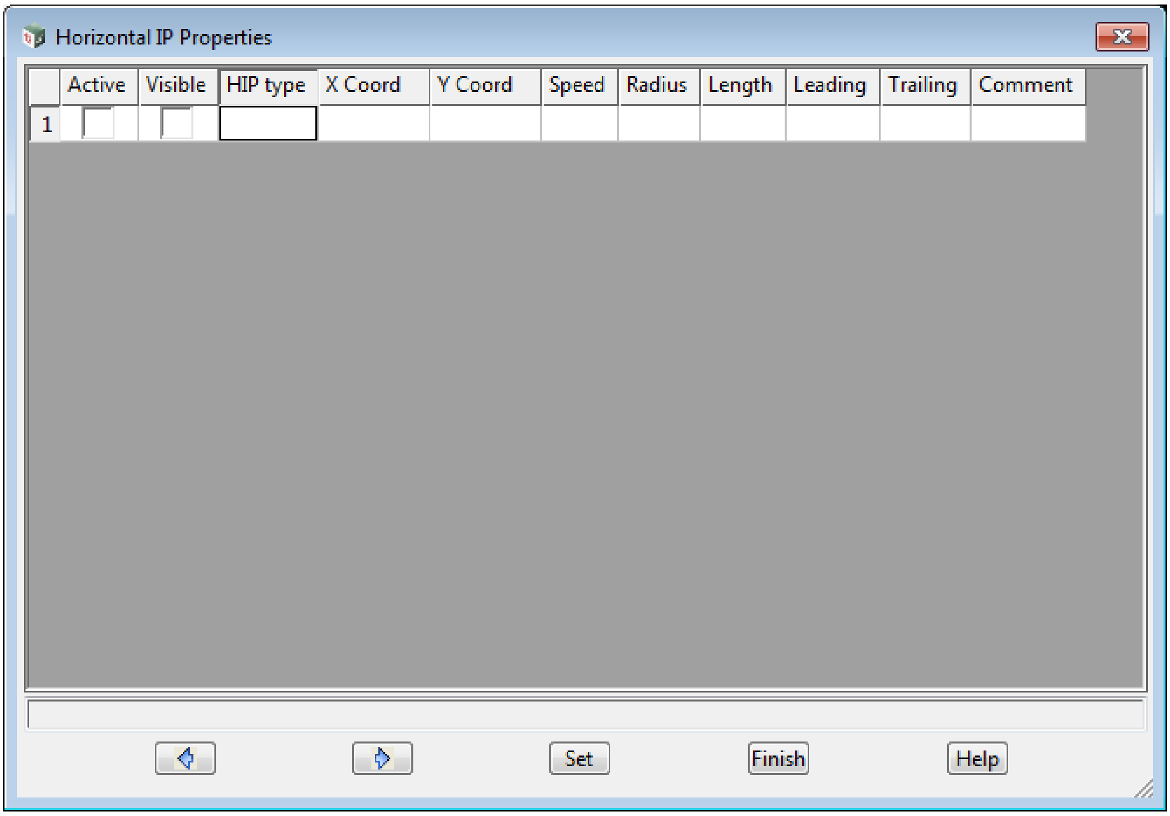


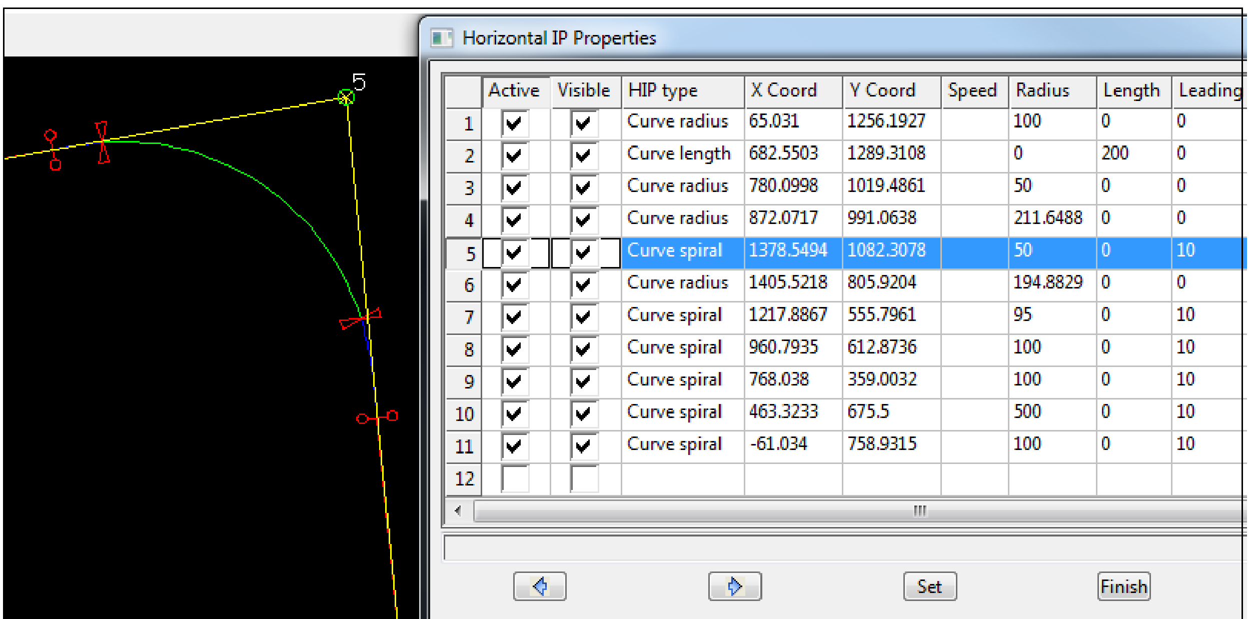
For definitions of the horizontal and vertical elements, go to 21.9 Placing Parts for Super Alignments

14.4.11.4.2 SA - HIPs Editor

If the horizontal geometry of the super alignment consists of only HIP's (that is, no horizontal parts) then clicking on **HIPs Editor** brings up the **Horizontal IP Properties** panel which lists all the HIP's in the super alignment showing how the curves on the HIP are defined and the values for the HIP and accompanying arcs and transitions (spirals).

The **Horizontal IP Properties** panel is:





Super Alignment Profiled on a Plan View Table of HIP Information

Clicking on the row number in the table draws a green circle around the corresponding HIP in any plan views where the HIP is visible.

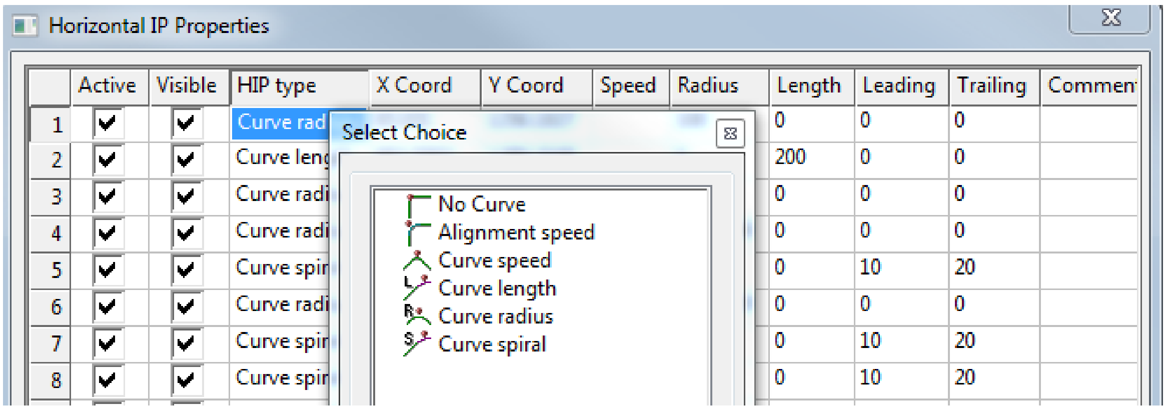
The Left arrow and Right arrow moves the highlighting to the previous HIP and next HIP respectively.

If Active is ticked for a row then that HIP is used in the super alignment.
If Active is not ticked for a row then that HIP is not used in the super alignment.

If Visible is ticked for a row then the curve at the HIP is used in the super alignment.
If Visible is not ticked for a row then the curve at the HIP is not used in the super alignment.

Warning: the row number will not be the same as the HIP numbers shown on a plan view if there are HIP's that are not Active.

The **HIP type** can be modified by clicking RB in the **HIP Type** column for the HIP to be modified to bring up the choices for the HIP.



When the HIP Type is changed, the new information can be typed into the appropriate cells for that HIP and then the **Set** button is pressed to make the change.

Note: if the entire line is already highlighted, you will need to first click LB in the appropriate HIP Type cell to get focus just on that cell and then click RB. If a pop-up with Browse comes up, select Browse to get the HIP Type choices.

Any of the values in the other columns X Coord, Y Coord, Speed, Radius, Length, Leading, Trailing and Comment can be modified and the **Set** button pressed to use the new values (if they are appropriate for the HIP Type).

Note:

Length is the length of the arc when the HIP Type is *Curve length*

Speed is the design speed for the curves at the HIP Type is *Curve speed*.

Radius is the radius of the arc when HIP Type is *Curve radius* or *Curve spiral.*

Leading is the length of the leading transition (spiral) when HIP Type is *Curve spiral*.

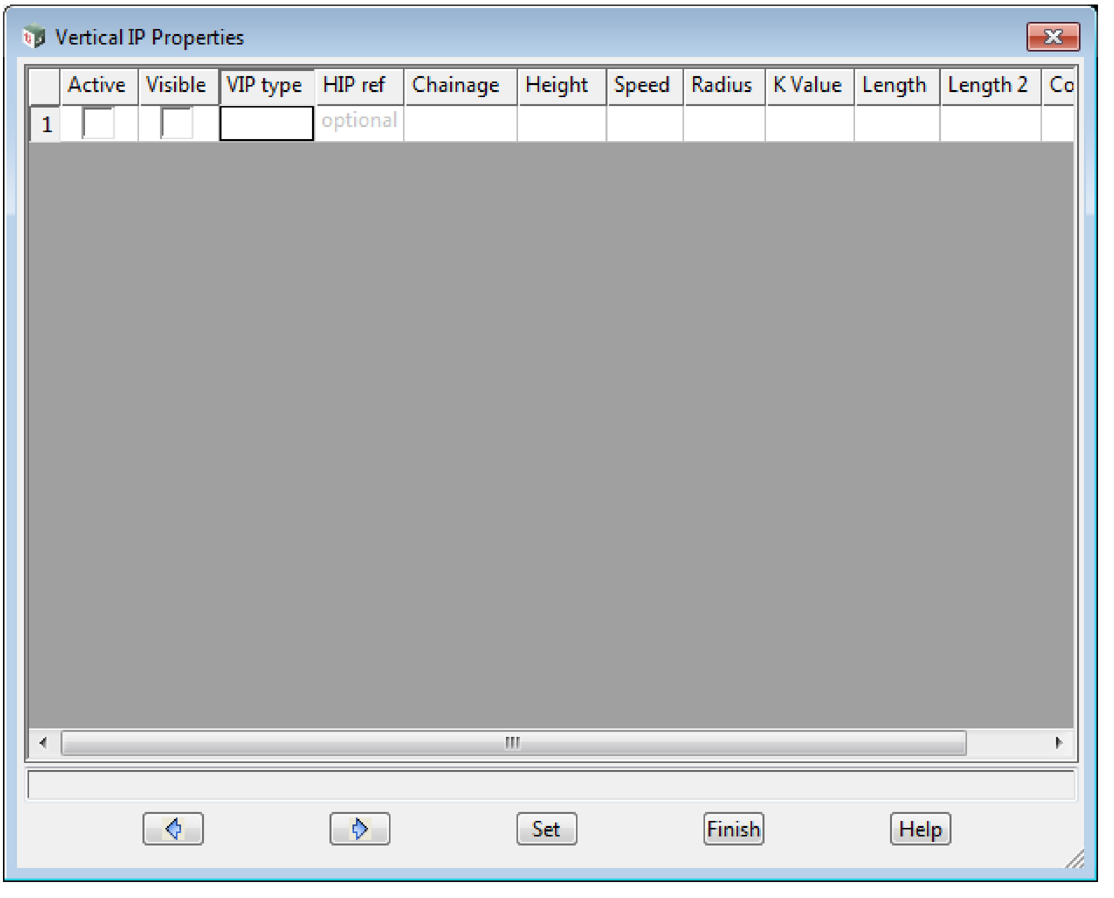
Trailing is the length of the trailing transition (spiral) when HIP Type is *Curve spiral*.

Warning: the *not Active* HIP's are still listed in the HIP table but the labelling of the HIP's on a plan view does not include them so in that case, the line number in the **Horizontal IP Properties** panel is not the same as the HIP number displayed on a plan view.

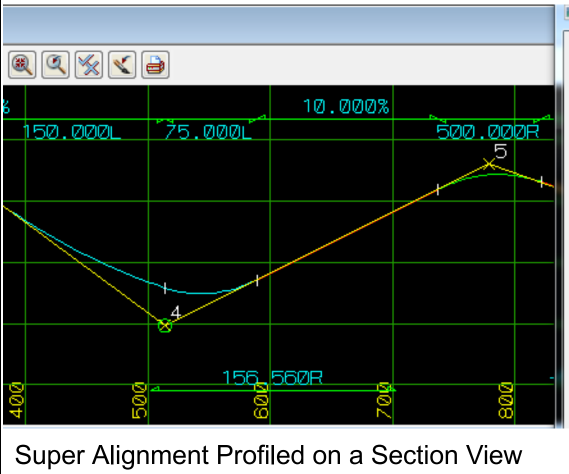
14.4.11.4.3 SA - VIPs Editor

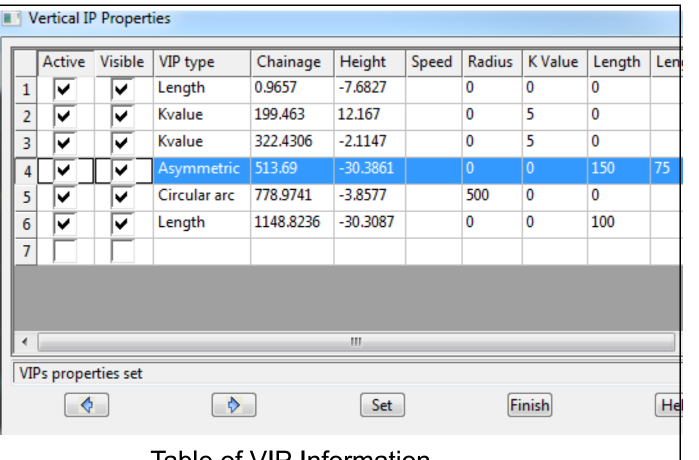
If the vertical geometry of the super alignment consists of only VIPs (that is, no vertical parts) then clicking on **VIPs Editor** brings up the **Vertical IP Properties** panel which lists all the VIP's in the super alignment showing how the parabolas, asymmetric parabolas and arcs on the VIP are defined and the values for the VIP and accompanying parabolas and arcs.

The **Vertical IP Properties** is:



and when there is VIP information to display, looks like:





	Active	Visible	VIP type	Chainage	Height	Speed	Radius	K Value	Length	Len
1	✓	✓	Length	0.9657	-7.6827		0	0	0	
2	✓	✓	Kvalue	199.463	12.167		0	5	0	
3	✓	✓	Kvalue	322.4306	-2.1147		0	5	0	
4	✓	✓	Asymmetric	513.69	-30.3861		0	0	150	75
5	✓	✓	Circular arc	778.9741	-3.8577		500	0	0	
6	✓	✓	Length	1148.8236	-30.3087		0	0	100	
7										

Super Alignment Profiled on a Section View

Table of VIP Information

Clicking on the row number in the table draws a green circle around the corresponding HIP in any plan views where the HIP is visible.

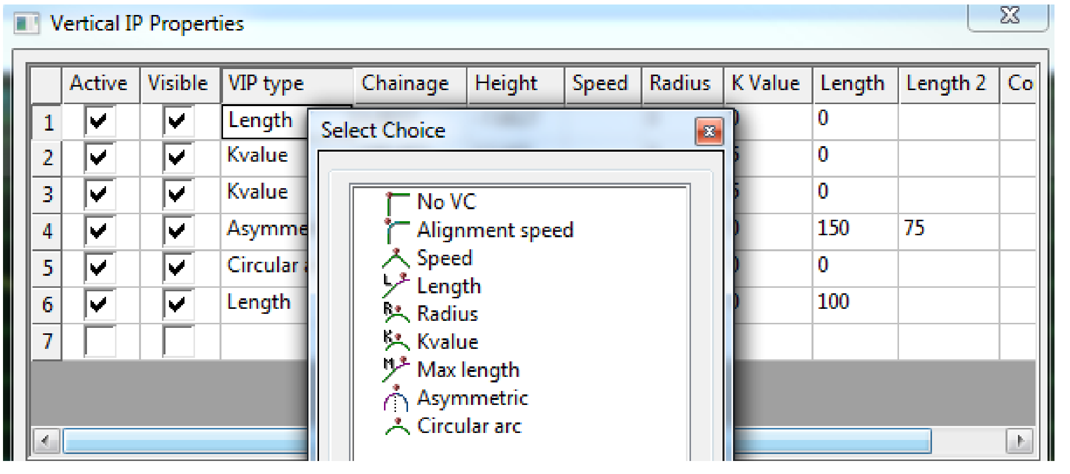
The Left arrow and Right arrow moves the highlighting to the previous VIP and next VIP respectively.

If Active is *ticked* for a row then that VIP is used in the super alignment.
If Active is *not ticked* for a row then that VIP is not used in the super alignment.

If Visible is *ticked* for a row then the curve at the VIP is used in the super alignment.
If Visible is *not ticked* for a row then the curve at the VIP is not used in the super alignment.

Warning: the row number will not be the same as the VIP numbers shown on a section view if there are VIP's that are not Active.

The **VIP type** can be modified by clicking RB in the **VIP Type** column for the VIP to be modified to bring up the choices for the VIP.



When the VIP Type is changed, the new information can be typed into the appropriate cells for that VIP and then the **Set** button is pressed to make the change.

Note: if the entire line is already highlighted, you will need to first click LB in the appropriate VIP Type cell to get focus just on that cell and then click RB. If a pop-up with Browse comes up, select Browse to get the HIP Type choices.

Any of the values in the other columns Chainage, Height, Speed, Radius, K Value, Length, Length2 and Comment can be modified and the **Set** button pressed to use the new values (if they are appropriate for the VIP Type).

Note:

Length is the length of the parabola when the VIP Type is *Length*

Length and Length2 are the lengths for an Asymmetric parabola when VIP Type is *Asymmetric*

Speed is the design speed for the parabola when VIP Type is *Speed*.

Radius is the radius of the arc when VIP Type is *Circular arc* or the effective radius for a parabola when VIP Type is *Radius.*

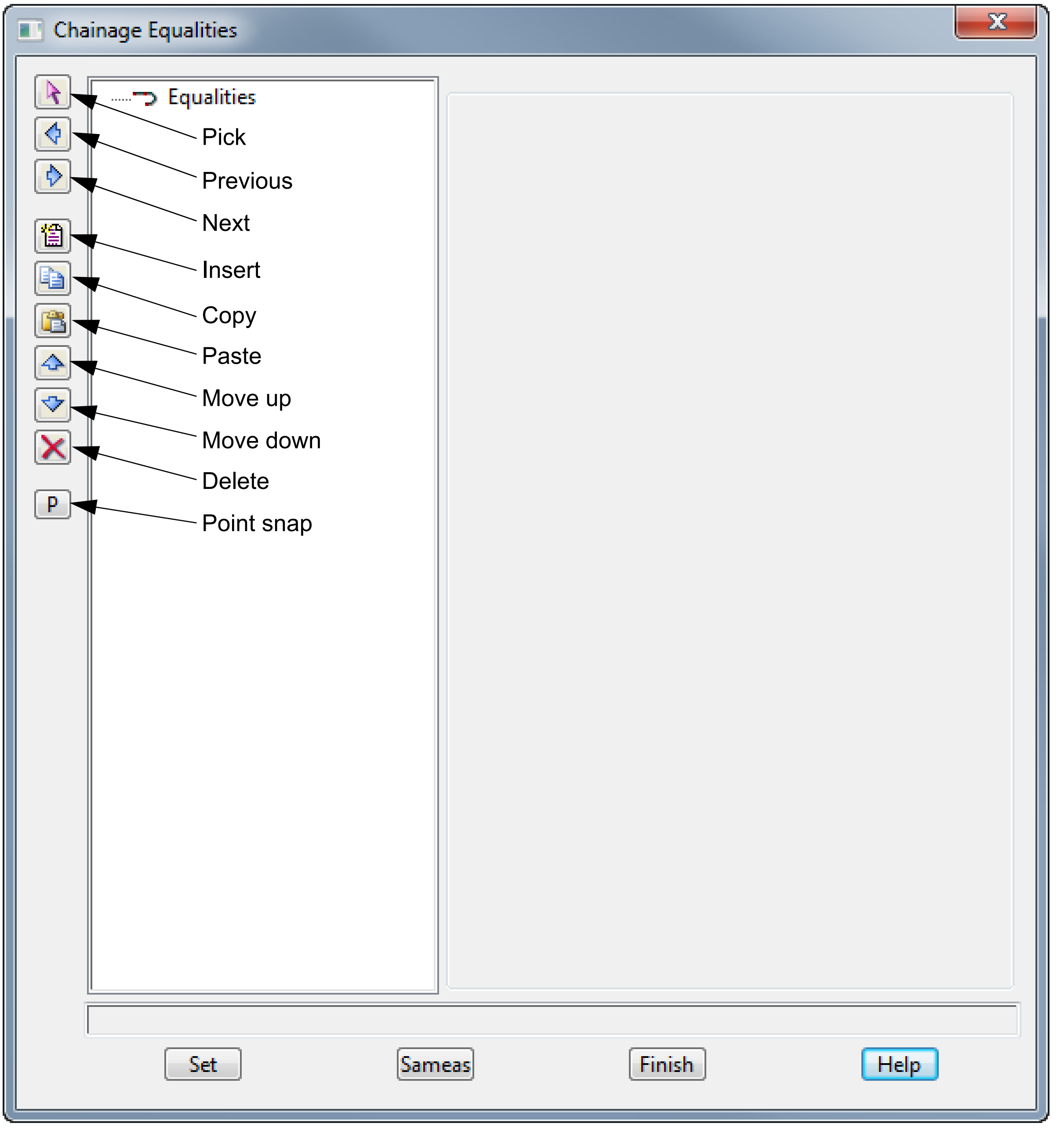
K Value is the K value for the parabola when HIP Type is *K value*.

Warning: the *not Active* are still listed in the VIP table but the labelling of the VIP's on a section view does not include them so in that case, the line number in the **Vertical IP Properties** panel is not the same as the VIP number displayed on a section view.

14.4.11.4.4 SA - Chainage Equalities Editor

This section of documentation is a work in progress and will be updated in subsequent releases.

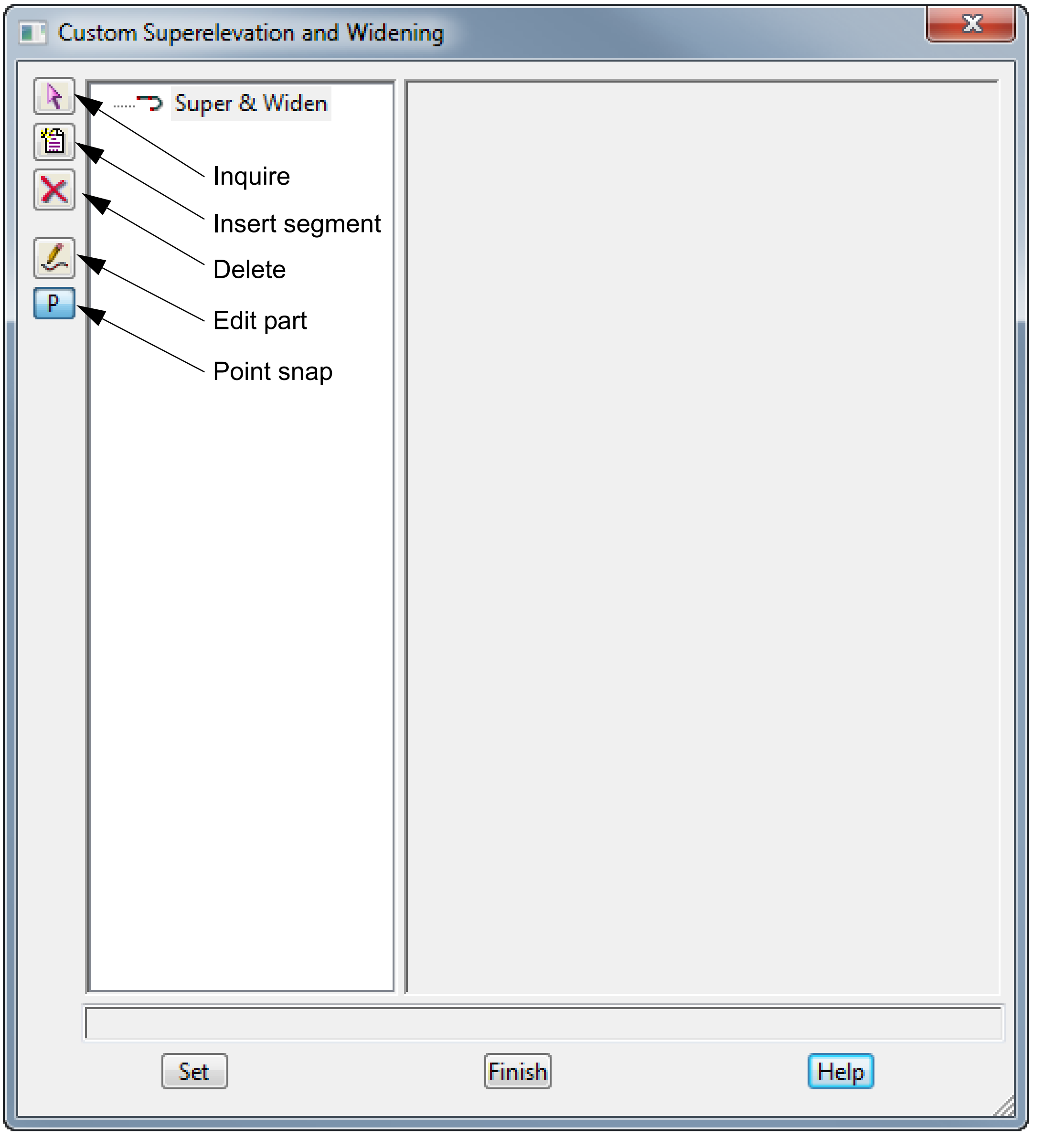
If Use chainage equality is ticked in the Advanced > Equality section of the Properties for the super alignment, then the **Chainage Equalities** panel is displayed.



14.4.11.4.5 SA - Custom Superelevation and Widening Editor

This section of documentation is a work in progress and will be updated in subsequent releases.

If Use design standards is ticked in the Advanced > Design section of the Properties for the super alignment, then the **Custom Superelevation and Widening** panel is displayed.

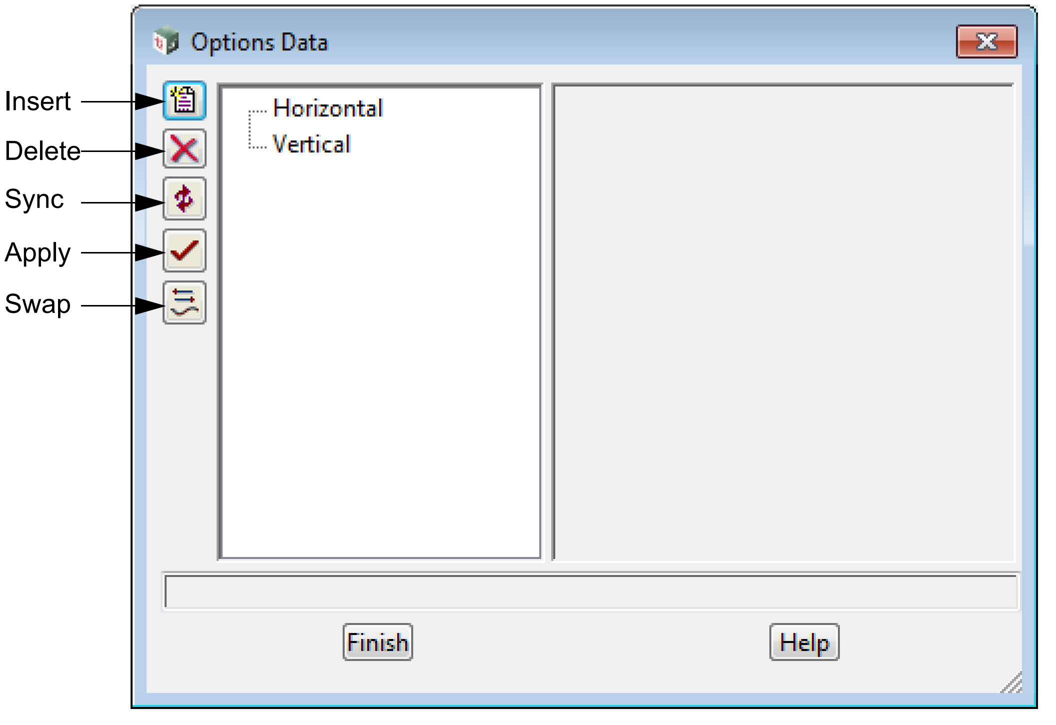


14.4.11.4.6 SA - Centreline Options

At any time, a copy of the current horizontal or vertical geometry can be made and stored under an Option name.

So an Option is like a backup (safe copy) of the current horizontal or vertical geometry and can be restored as the super alignment geometry at a later time.

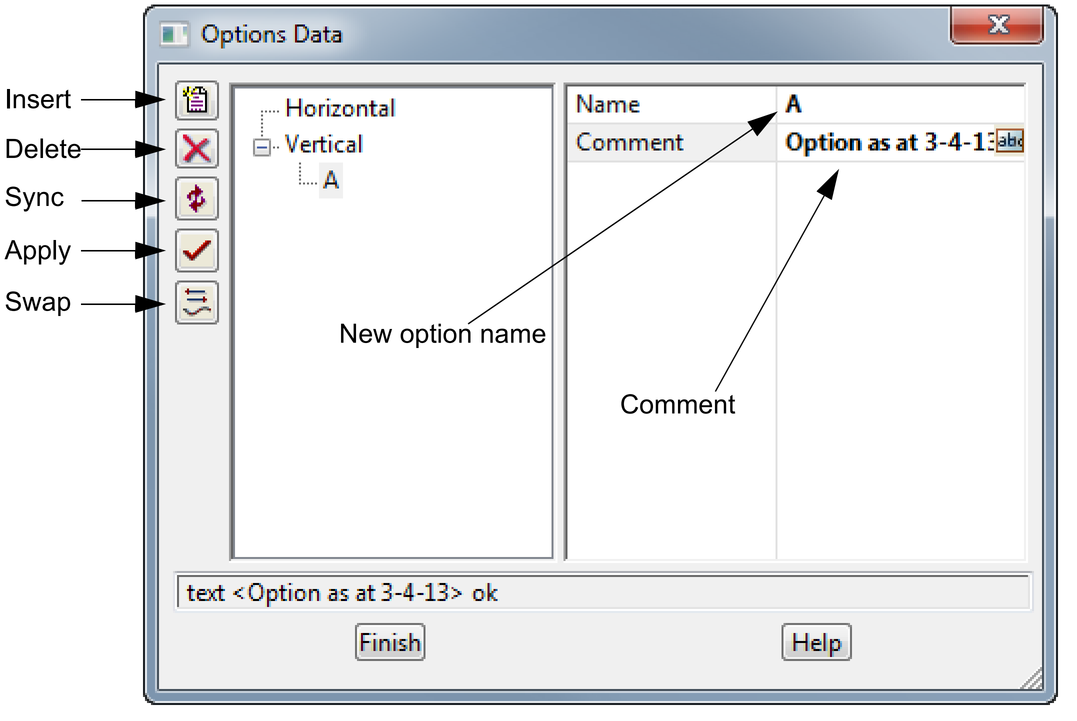
Clicking Centreline Options brings up the **Options Data** panel.



Insert

If the Horizontal/Vertical branch is highlighted and **Insert** clicked, a new Option is created under Horizontal/ Vertical with the default name Horizontal/Vertical, and a copy of the current horizontal/vertical geometry of the super alignment is stored with the new option.

Name and Comment can be modified and are changed when <Enter> is pressed.



Delete

If the Horizontal/Vertical Option is highlighted and **Delete** clicked, then the Option is deleted.

Sync

If the Horizontal/Vertical Option is highlighted and **Sync** clicked, the horizontal/vertical geometry of the Option is deleted and replaced by the current horizontal/vertical geometry of the super alignment. That is, the Option geometry is taken from the super alignment.

Apply

If the Horizontal/Vertical branch is highlighted and **Apply** clicked, the horizontal/vertical geometry of the super alignment is deleted and replaced by the horizontal/vertical geometry of the Option. That is, the super alignment geometry is taken from the Option.

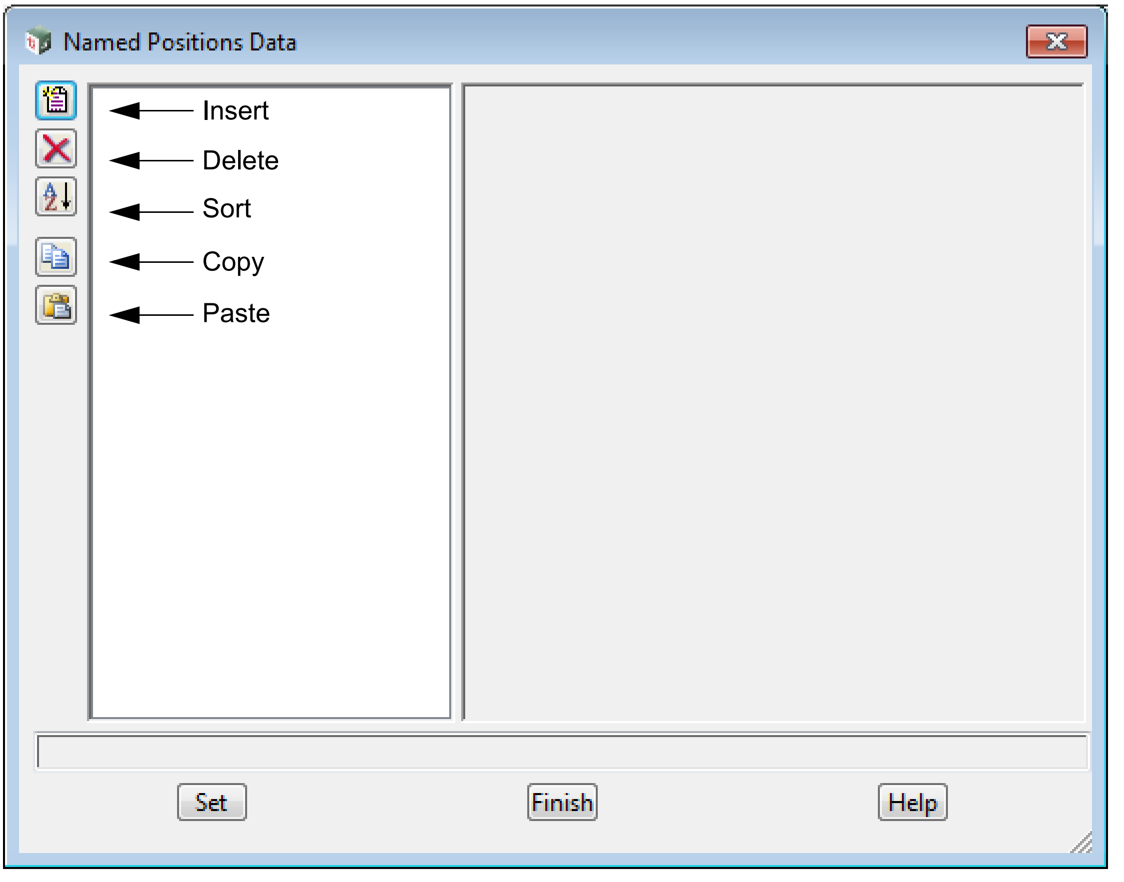
Swap

If the Horizontal/Vertical branch is highlighted and **Swap** clicked, the current horizontal/vertical geometry of the super alignment is replaced by the horizontal/vertical geometry of the Option and the horizontal/vertical geometry of the Option is replaced by the current horizontal/vertical geometry of the super alignment. That is, the two geometries are swapped.

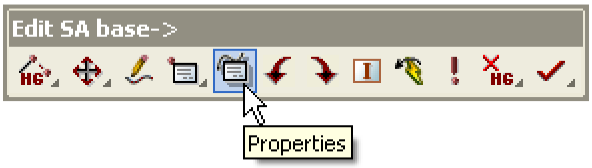
14.4.11.4.7 SA - Named Positions

This section of documentation is a work in progress and will be updated in subsequent releases.

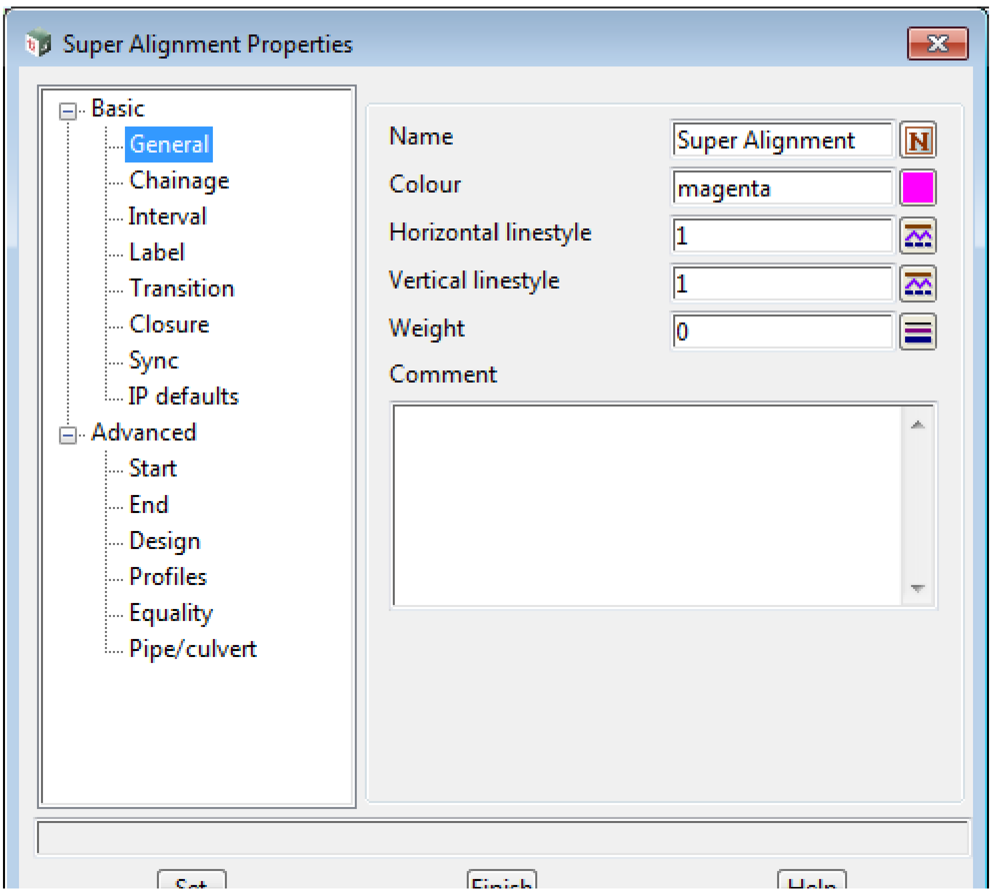
Clicking Named Positions brings up the **Named Positions Data** panel.



14.4.11.5 SA - Super Alignment Properties Icon



Selecting **Properties**  displays the **Super Alignment Properties** panel



The information in the **Super Alignment Properties** panel is the same as for the **Create Super Alignment** panel.

The only difference is that the **Set** button only exists on the **Super Alignment Properties** panel and if any changes are made to the values on any branch, then the **Set** button must be clicked before leaving that branch to modify the values for the string.

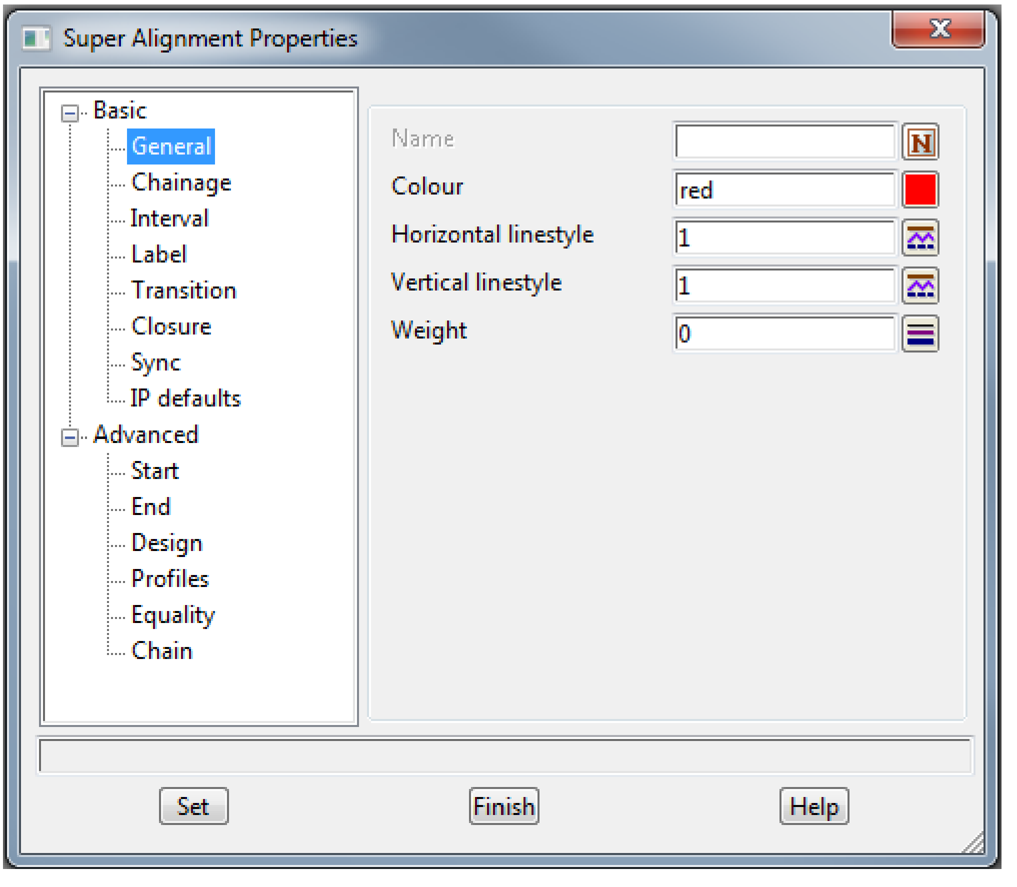
WARNING: this means that you must click on **Set** after modifying each branch - you can't change values on a number of branches and then click **Set**.

See 14.2.1 Create Super Alignment for details on all the fields and values in the branches of this panel.

For information on each branch, go to:

Basic > General branch
Basic > Chainage branch
Basic > Interval branch
Basic > Label branch

Basic > General branch



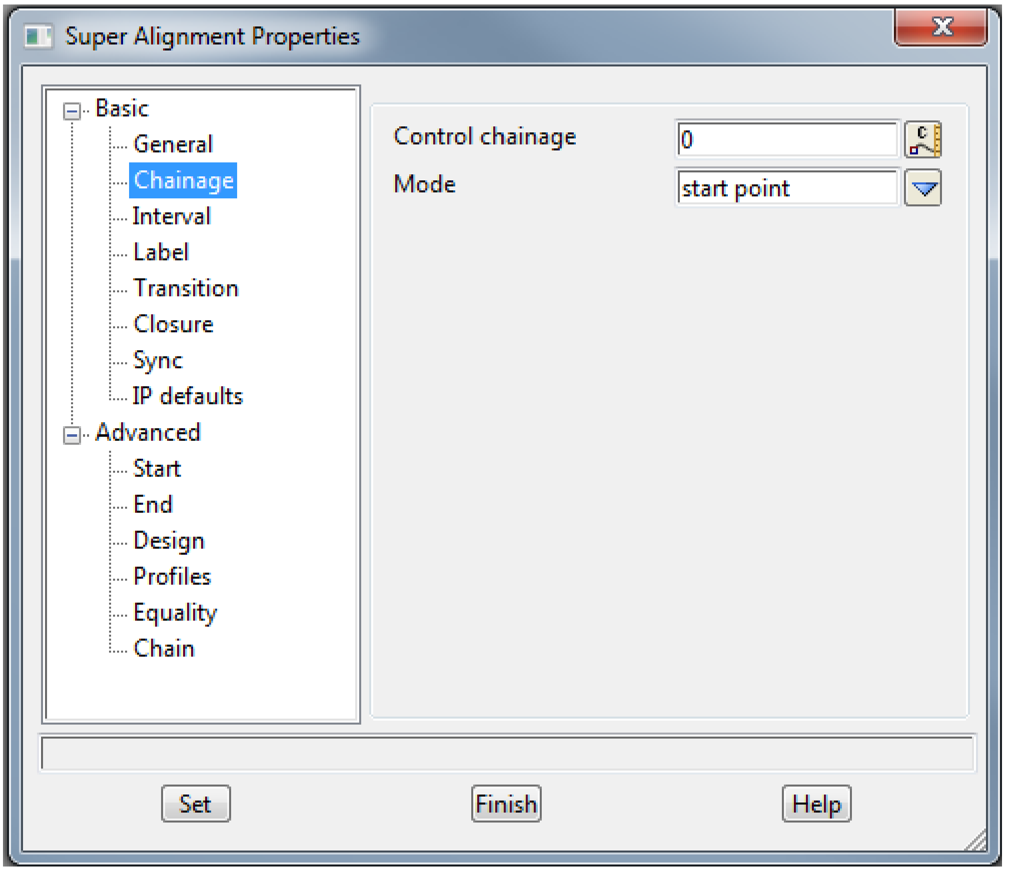
The information in the Basic >General branch is the same as for the Basic >General branch of the **Create Super Alignment** panel.

The only difference is that the **Set** button only exists on the **Super Alignment Properties** panel and if any changes are made to the values on the Basic >General branch, then the **Set** button must be clicked before leaving the Basic >General branch to modify the values for the string.

WARNING: this means that you must click on **Set** after modifying each branch - you can't change values on a number of branches and then click **Set**.

See Basic > General branch for details on all the fields and values in the Basic >General branch.

Basic > Chainage branch



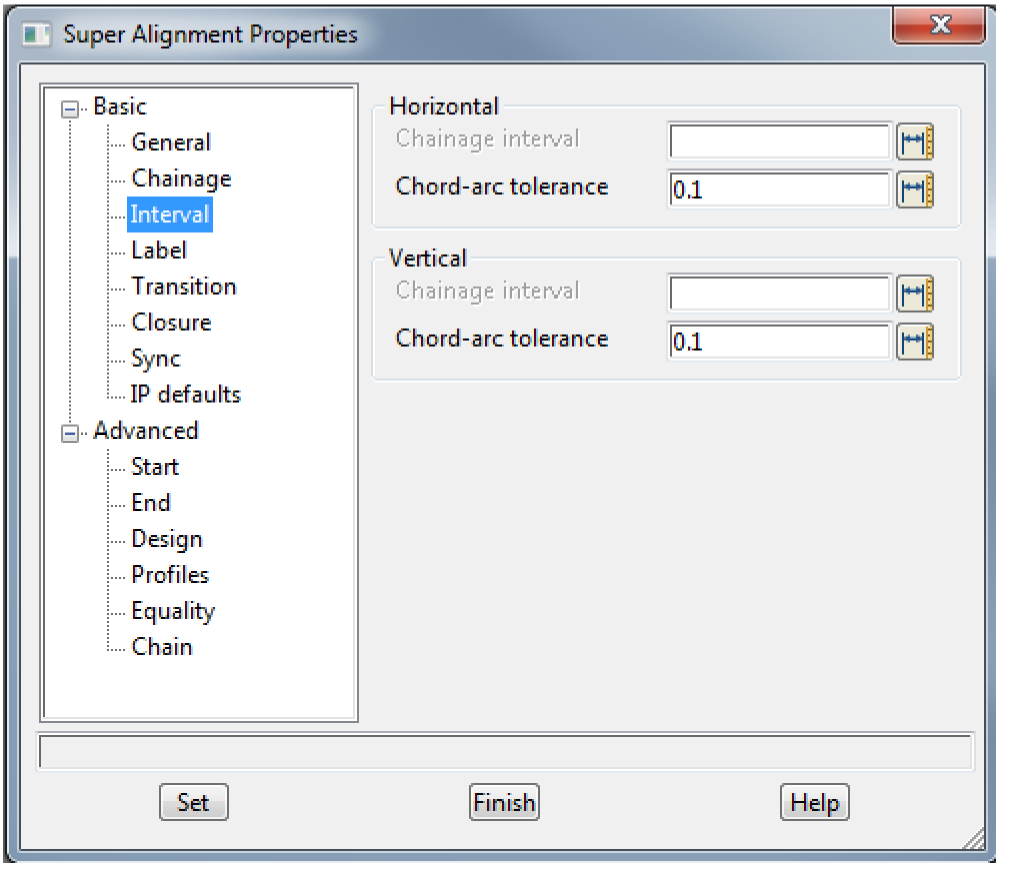
The information in the Basic >Chainage branch is the same as for the Basic >Chainage branch of the **Create Super Alignment** panel.

The only difference is that the **Set** button only exists on the **Super Alignment Properties** panel and if any changes are made to the values on the Basic >Chainage branch, then the **Set** button must be clicked before leaving the Basic >Chainage branch to modify the values for the string.

WARNING: this means that you must click on **Set** after modifying each branch - you can't change values on a number of branches and then click **Set**.

See Basic > Chainage branch for details on all the fields and values in the Basic >Chainage branch.

Basic > Interval branch



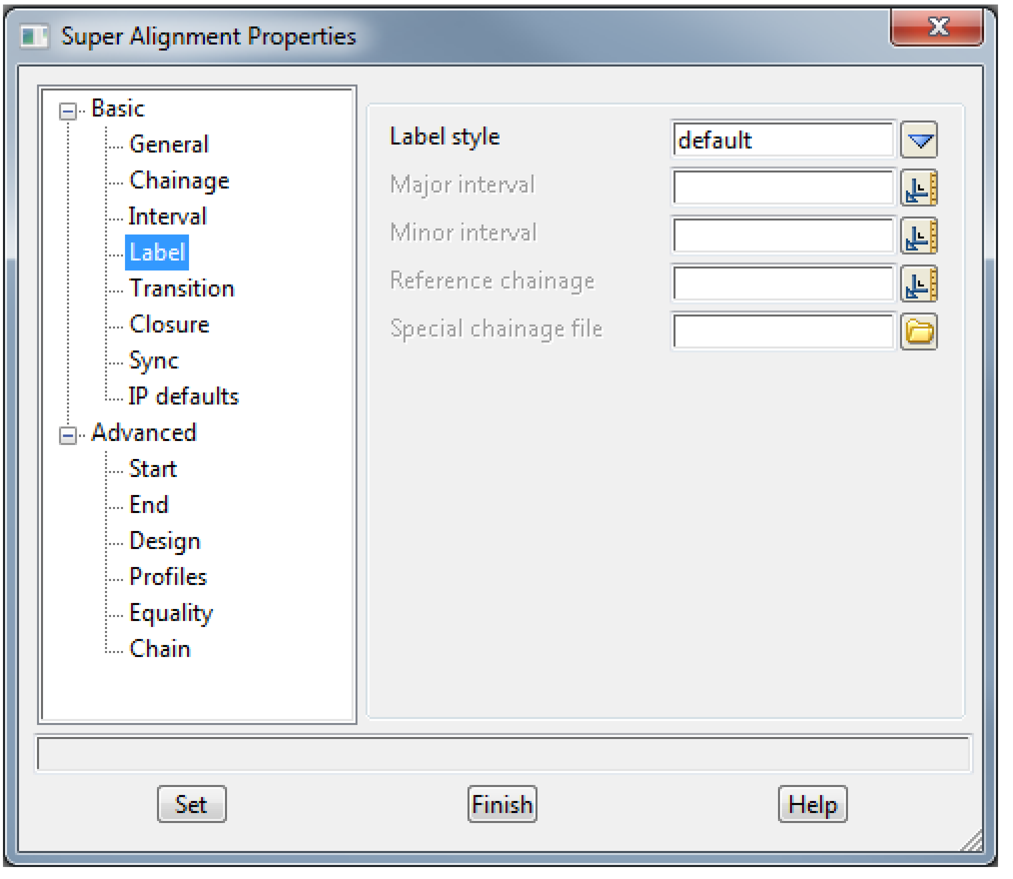
The information in the Basic >Interval branch is the same as for the Basic >Interval branch of the **Create Super Alignment** panel.

The only difference is that the **Set** button only exists on the **Super Alignment Properties** panel and if any changes are made to the values on the Basic >Interval branch, then the **Set** button must be clicked before leaving the Basic >Interval branch to modify the values for the string.

WARNING: this means that you must click on **Set** after modifying each branch - you can't change values on a number of branches and then click **Set**.

See Basic > Interval branch for details on all the fields and values in the Basic >Interval branch.

Basic > Label branch



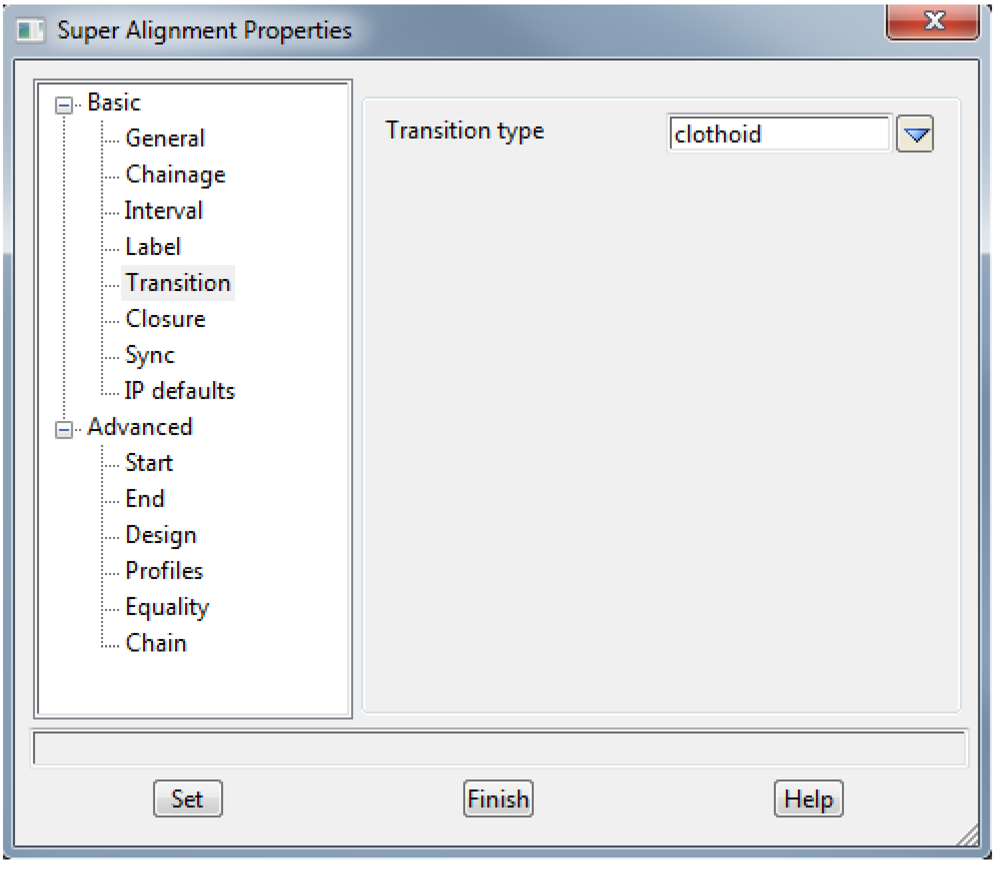
The information in the Basic >Label branch is the same as for the Basic >Label branch of the **Create Super Alignment** panel.

The only difference is that the **Set** button only exists on the **Super Alignment Properties** panel and if any changes are made to the values on the Basic >Label branch, then the **Set** button must be clicked before leaving the Basic >Label branch to modify the values for the string.

WARNING: this means that you must click on **Set** after modifying each branch - you can't change values on a number of branches and then click **Set**.

See Basic > Label branch for details on all the fields and values in the Basic >Label branch.

Basic > Transition branch



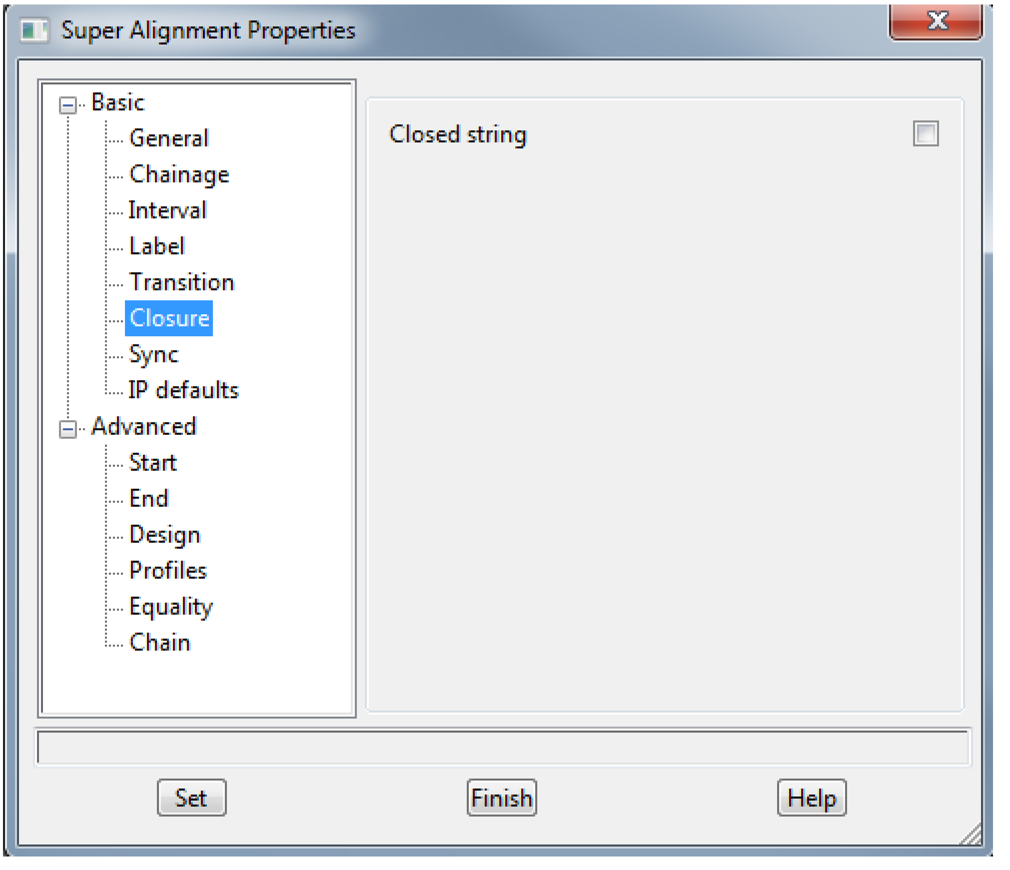
The information in the Basic >Transition branch is the same as for the Basic >Transition branch of the **Create Super Alignment** panel.

The only difference is that the **Set** button only exists on the **Super Alignment Properties** panel and if any changes are made to the values on the Basic >Transition branch, then the **Set** button must be clicked before leaving the Basic >Transition branch to modify the values for the string.

WARNING: this means that you must click on **Set** after modifying each branch - you can't change values on a number of branches and then click **Set**.

See Basic > Transition branch for details on all the fields and values in the Basic >Transition branch.

Basic > Closure branch



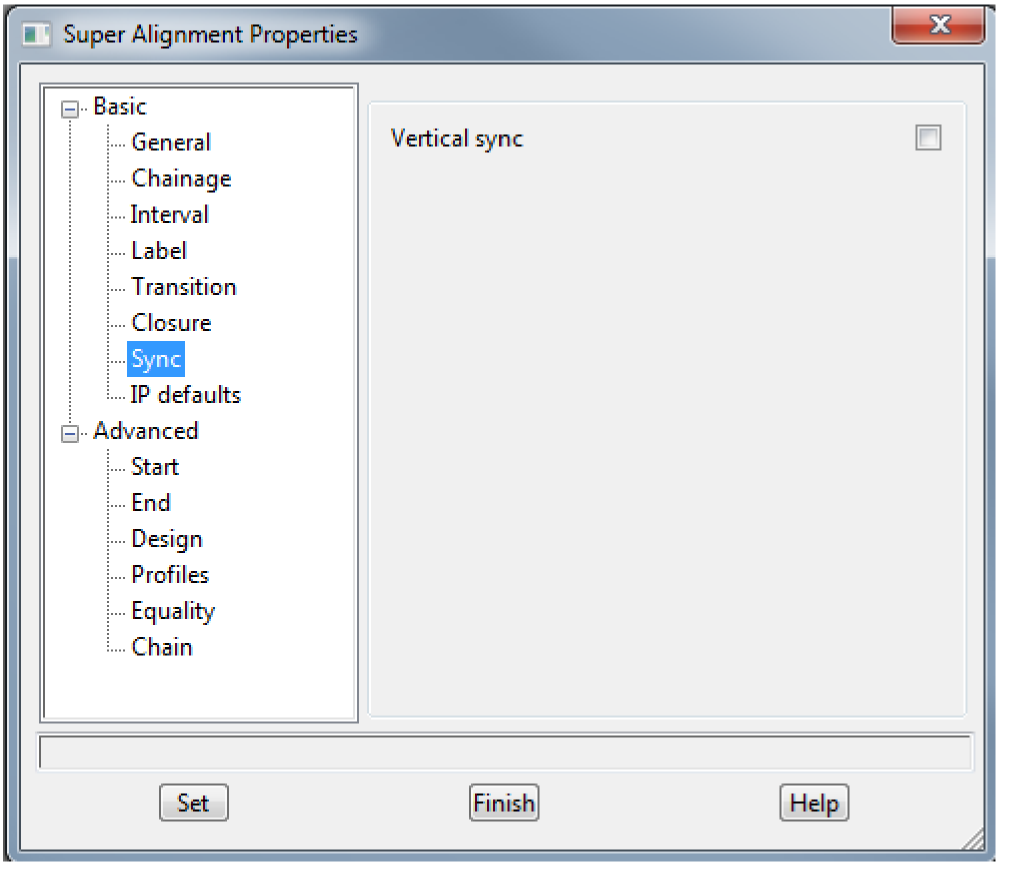
The information in the Basic >Closure branch is the same as for the Basic >Closure branch of the **Create Super Alignment** panel.

The only difference is that the **Set** button only exists on the **Super Alignment Properties** panel and if any changes are made to the values on the Basic >Closure branch, then the **Set** button must be clicked before leaving the Basic >Closure branch to modify the values for the string.

WARNING: this means that you must click on **Set** after modifying each branch - you can’t change values on a number of branches and then click **Set**.

See Basic > Closure branch for details on all the fields and values in the Basic >Closure branch.

Basic > Sync branch



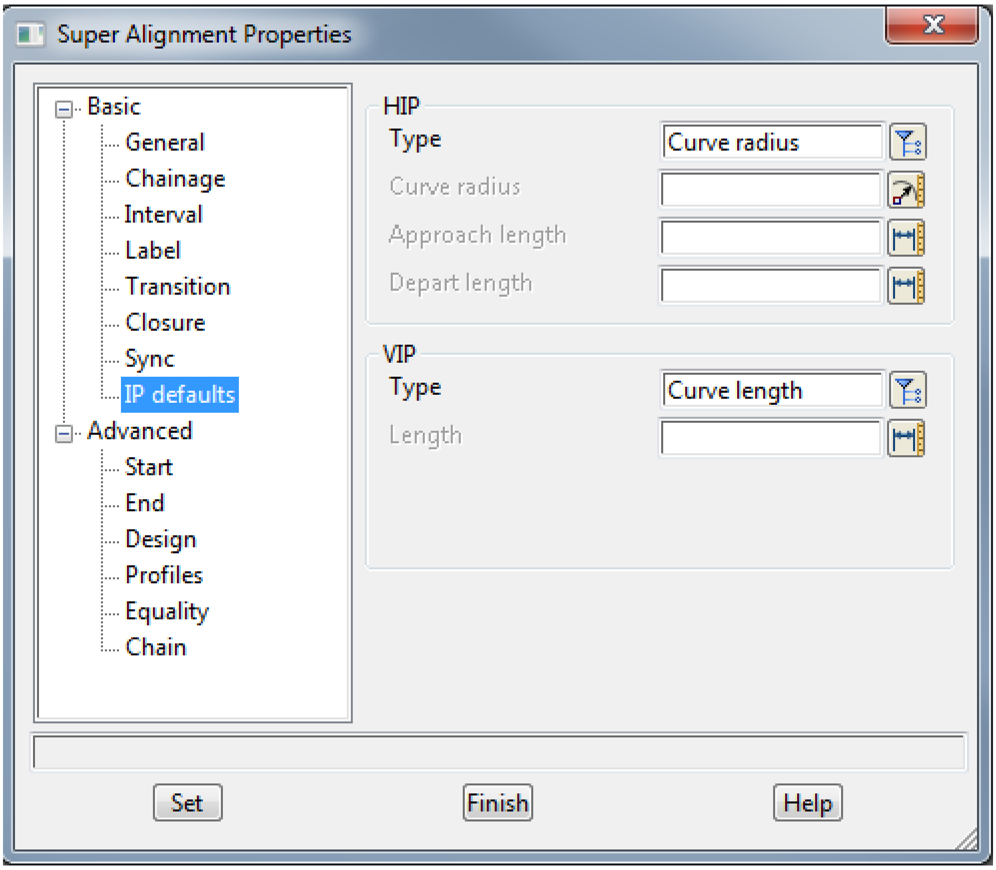
The information in the Basic >Sync branch is the same as for the Basic >Sync branch of the **Create Super Alignment** panel.

The only difference is that the **Set** button only exists on the **Super Alignment Properties** panel and if any changes are made to the values on the Basic >Sync branch, then the **Set** button must be clicked before leaving the Basic >Sync branch to modify the values for the string.

WARNING: this means that you must click on **Set** after modifying each branch - you can't change values on a number of branches and then click **Set**.

See Basic > Sync branch for details on all the fields and values in the Basic >Sync branch.

Basic > IP defaults branch



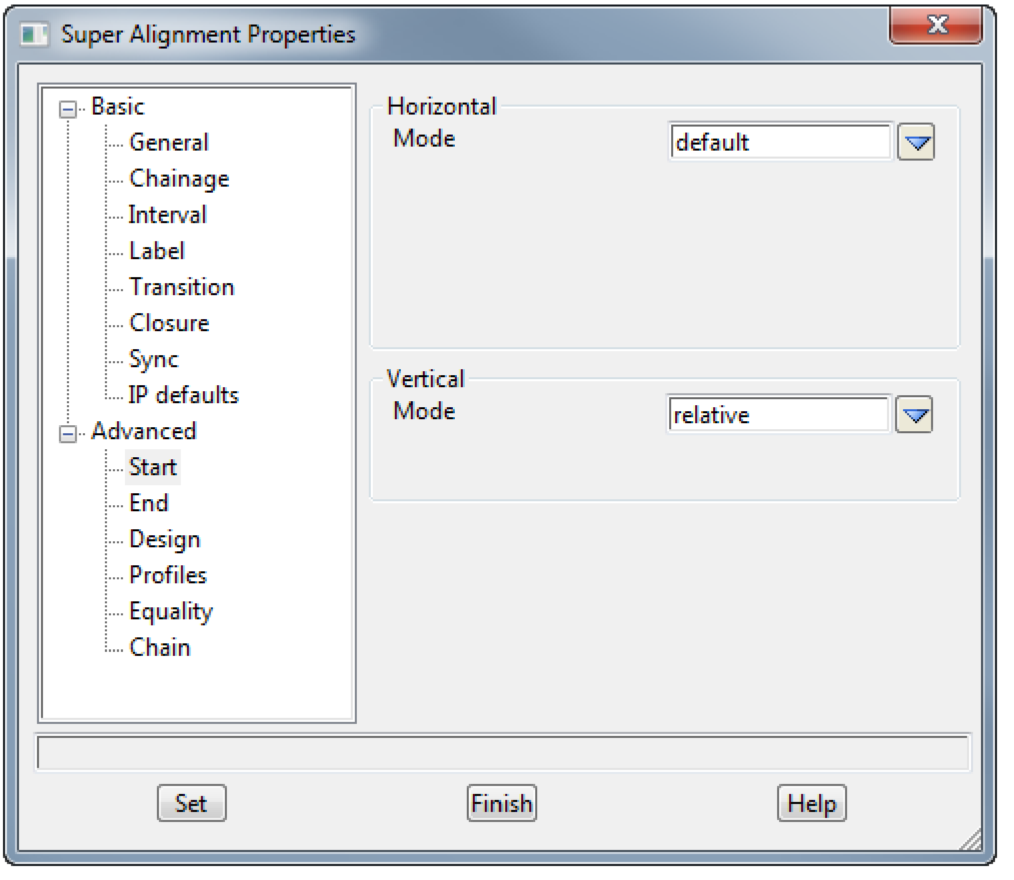
The information in the Basic >IP defaults branch is the same as for the Basic >IP defaults branch of the **Create Super Alignment** panel.

The only difference is that the **Set** button only exists on the **Super Alignment Properties** panel and if any changes are made to the values on the Basic >IP defaults branch, then the **Set** button must be clicked before leaving the Basic >IP defaults branch to modify the values for the string.

WARNING: this means that you must click on **Set** after modifying each branch - you can't change values on a number of branches and then click **Set**.

See Basic > IP defaults branch for details on all the fields and values in the Basic >IP defaults branch.

Advanced > Start branch



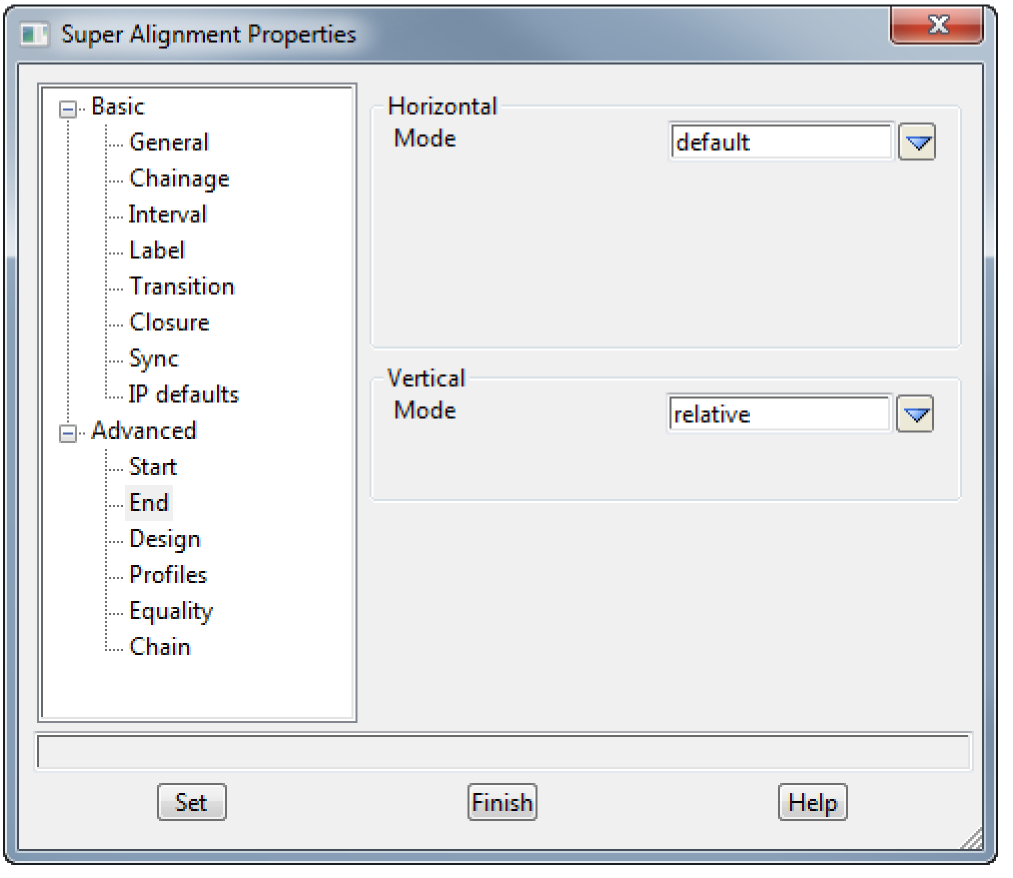
The information in the Advanced >Start branch is the same as for the Advanced >Start branch of the **Create Super Alignment** panel.

The only difference is that the **Set** button only exists on the **Super Alignment Properties** panel and if any changes are made to the values on the Advanced >Start branch, then the **Set** button must be clicked before leaving the Advanced >Start branch to modify the values for the string.

WARNING: this means that you must click on **Set** after modifying each branch - you can't change values on a number of branches and then click **Set**.

See Advanced > Start branch for details on all the fields and values in the Advanced >Start branch.

Advanced > End branch



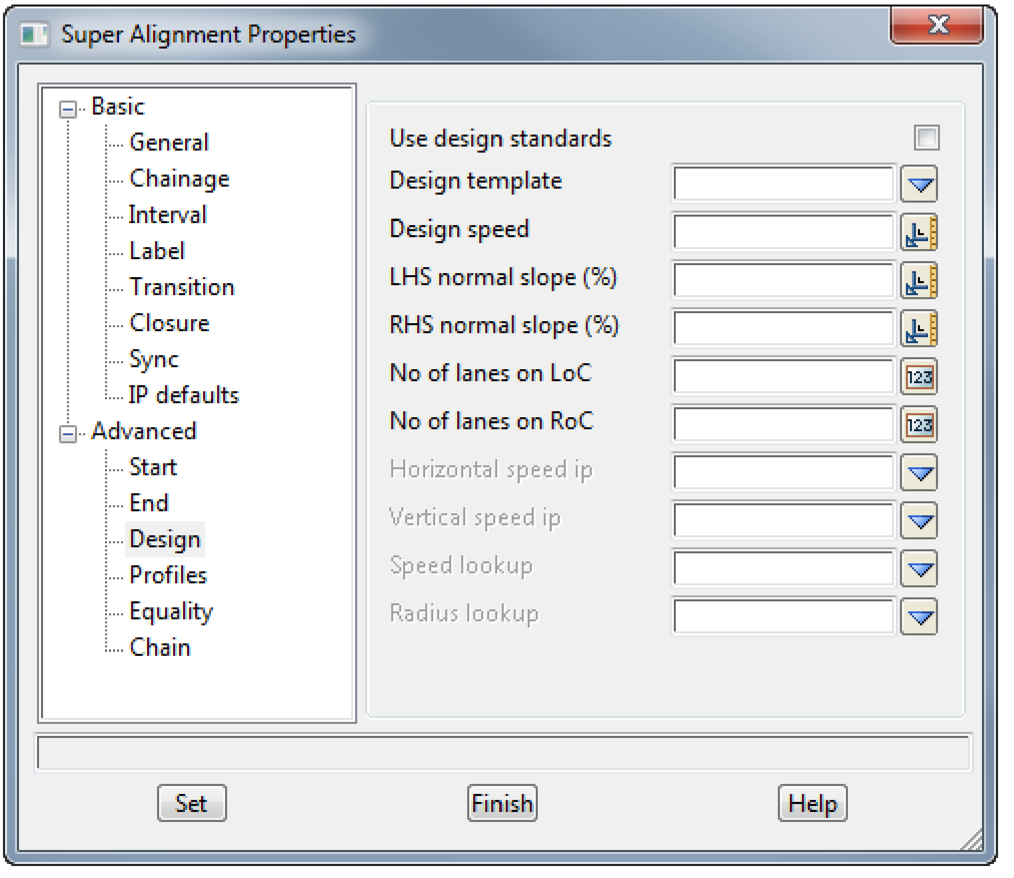
The information in the Advanced >End branch is the same as for the Advanced >End branch of the **Create Super Alignment** panel.

The only difference is that the **Set** button only exists on the **Super Alignment Properties** panel and if any changes are made to the values on the Advanced >End branch, then the **Set** button must be clicked before leaving the Advanced >End branch to modify the values for the string.

WARNING: this means that you must click on **Set** after modifying each branch - you can't change values on a number of branches and then click **Set**.

See Advanced > End branch for details on all the fields and values in the Advanced >End branch.

Advanced > Design branch



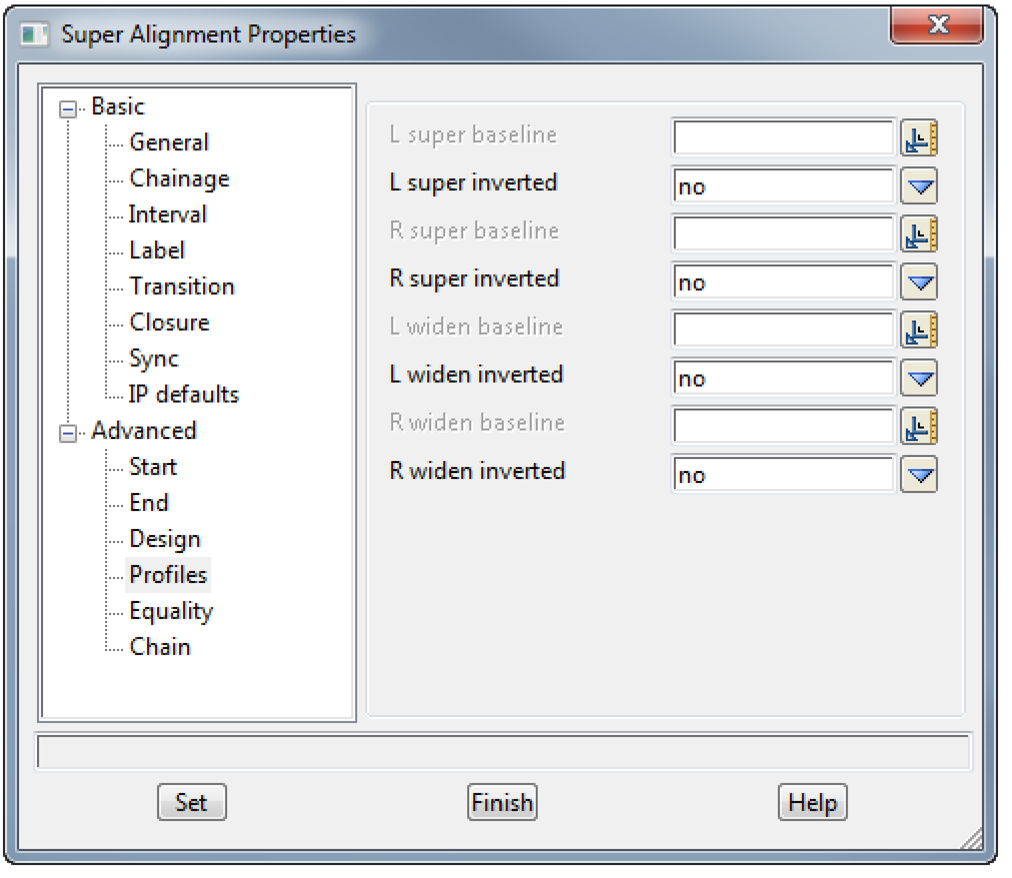
The information in the Advanced >Design branch is the same as for the Advanced >Design branch of the **Create Super Alignment** panel.

The only difference is that the **Set** button only exists on the **Super Alignment Properties** panel and if any changes are made to the values on the Advanced >Design branch, then the **Set** button must be clicked before leaving the Advanced >Design branch to modify the values for the string.

WARNING: this means that you must click on **Set** after modifying each branch - you can't change values on a number of branches and then click **Set**.

See Advanced > Design branch for details on all the fields and values in the Advanced >Design branch.

Advanced > Profiles branch



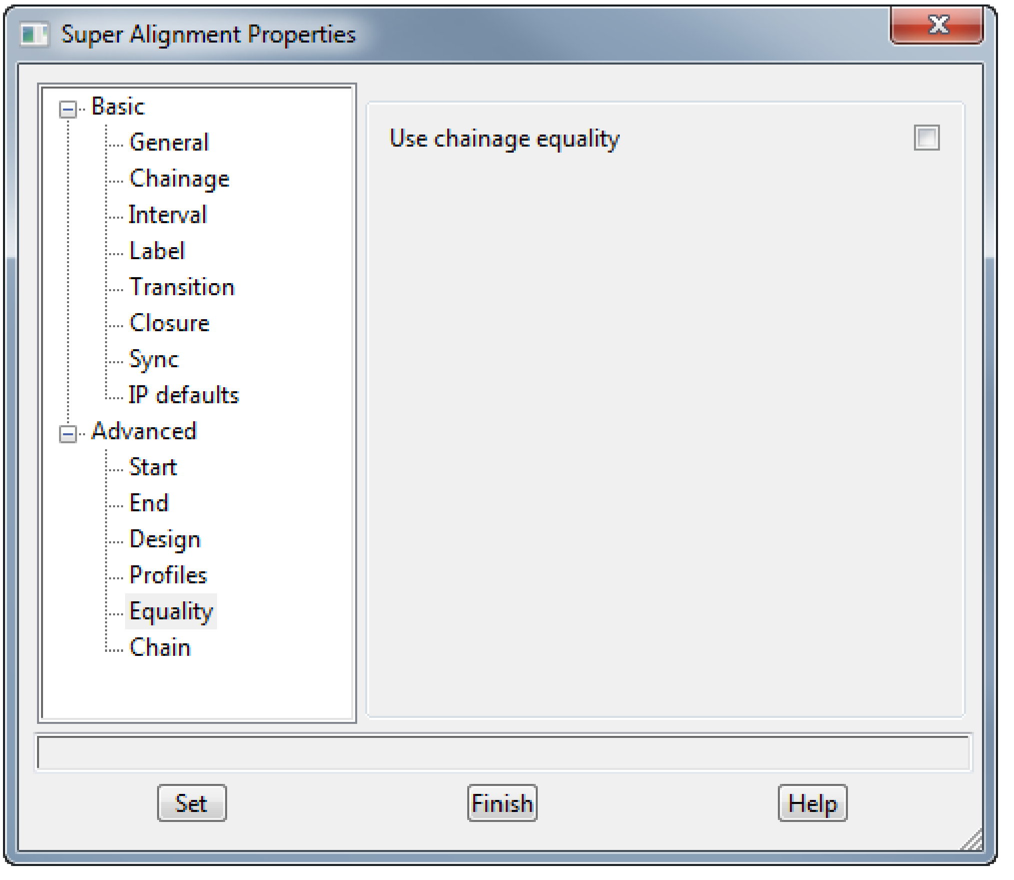
The information in the Advanced >Profiles branch is the same as for the Advanced >Profiles branch of the **Create Super Alignment** panel.

The only difference is that the **Set** button only exists on the **Super Alignment Properties** panel and if any changes are made to the values on the Advanced >Profiles branch, then the **Set** button must be clicked before leaving the Advanced >Profiles branch to modify the values for the string.

WARNING: this means that you must click on **Set** after modifying each branch - you can't change values on a number of branches and then click **Set**.

See Advanced > Profiles branch for details on all the fields and values in the Advanced >Profiles branch.

Advanced > Equality branch



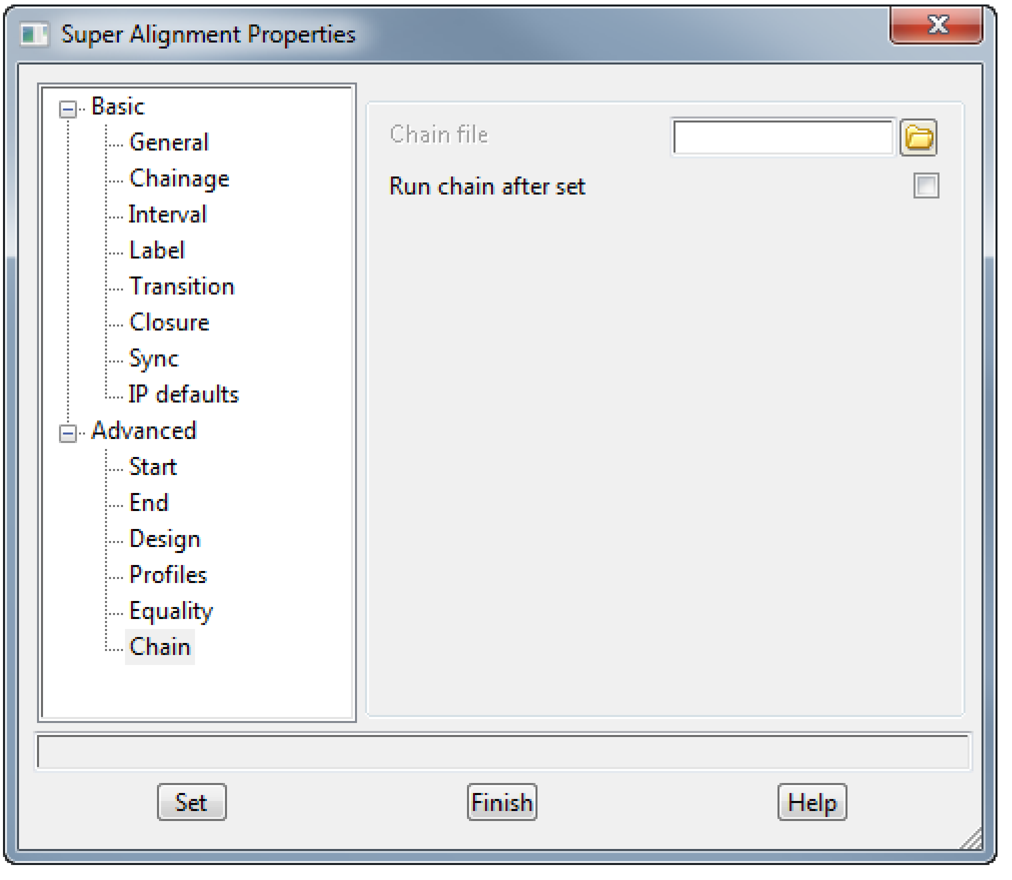
The information in the Advanced >Equality branch is the same as for the Advanced >Equality branch of the **Create Super Alignment** panel.

The only difference is that the **Set** button only exists on the **Super Alignment Properties** panel and if any changes are made to the values on the Advanced >Equality branch, then the **Set** button must be clicked before leaving the Advanced >Equality branch to modify the values for the string.

WARNING: this means that you must click on **Set** after modifying each branch - you can’t change values on a number of branches and then click **Set**.

See Advanced > Equality branch for details on all the fields and values in the Advanced >Equality branch.

Advanced > Chain branch



The information in the Advanced >Chain branch is the same as for the Advanced >Chain branch of the **Create Super Alignment** panel.

The only difference is that the **Set** button only exists on the **Super Alignment Properties** panel and if any changes are made to the values on the Advanced >Chain branch, then the **Set** button must be clicked before leaving the Advanced >Chain branch to modify the values for the string.

WARNING: this means that you must click on **Set** after modifying each branch - you can't change values on a number of branches and then click **Set**.

See Advanced > Chain branch (not yet implemented) for details on all the fields and values in the Advanced >Chain branch.

14.4.11.6 SA - Undo and Redo Icons

An Undo and Redo list is defined for each Editor open on the screen. That is, each open Editor has its own Undo/Redo lists.

As each option is performed in an open Editor, the reverse of the option is added to the Editors Undo list. Hence an option can be undone (reversed) by immediately clicking on the Undo icon.

Note: the Undo icon is.

If an option is undone, it is added to the Redo list so that it can be redone.

So clicking on the **Undo** icon undoes the last Editor operation for this super alignment and adds it to the top of the Redo list

Note: the Redo icon is

Clicking on the **Redo** icon redoes the last Editor operation that was undone for this super alignment and the undone operation is added to the top of the Undo list.

Important Note: as each option is added to an Undo or Redo list, the options already on the list are pushed down and none of the options pushed down can be undone/redone until all the options above it have been undone/redone. So both the Undo and Redo lists are last in, first out lists.

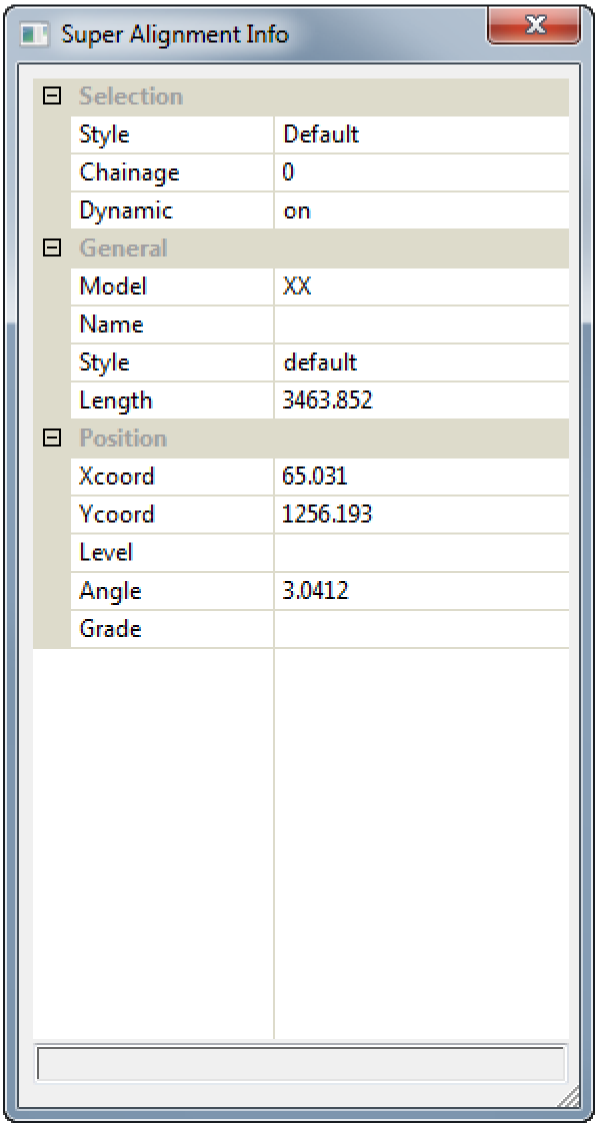
So the order in the Undo/Redo lists is important and can not be altered.

When the Editor is exited by either **Quit** or **Finish**, the undo and redo lists are deleted and are no longer usable.

14.4.11.7 SA - Info Icon

The Info icon is

and clicking on the **Info** icon brings up the **Super Alignment Info** panel (see 14.2.2.1 Super Alignment Info).


Super Alignment Info
Selection
Style
Default
Chainage
0
Dynamic
on
General
Model
XX
Name
Style
default
Length
3463.852
Position
Xcoord
65.031
Ycoord
1256.193
Level
Angle
3.0412
Grade

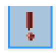
14.4.11.8 SA - Recalc Icon

The Recalc icon is

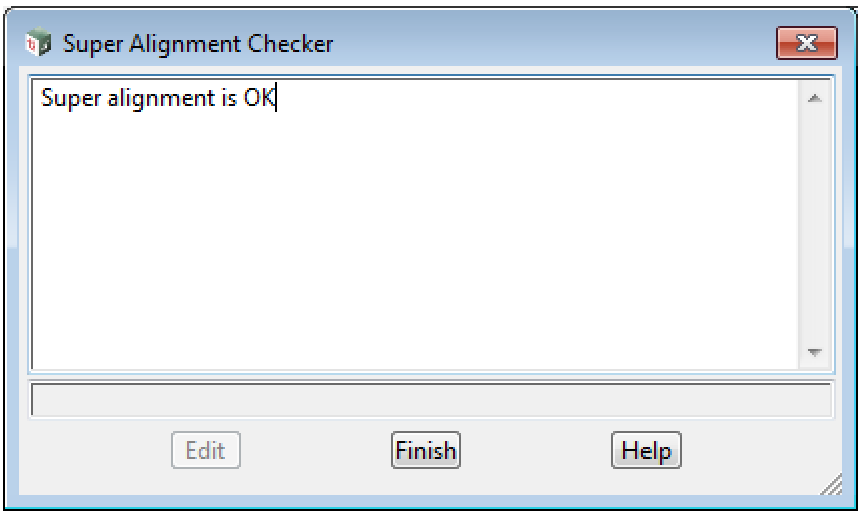
and clicking on the Recalc icon does a *recalc* for the super alignment.

14.4.11.9 SA - Error Checker Icon

The Error Checker icon is



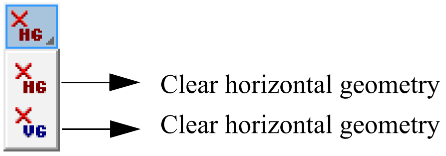
and clicking on the Error Checker icon brings up the **Super Alignment Checker** panel and runs the SA checking routines.



Warning: This options is still under development.

14.4.11.10 SA - Clear Toolbar

The Clear toolbar is

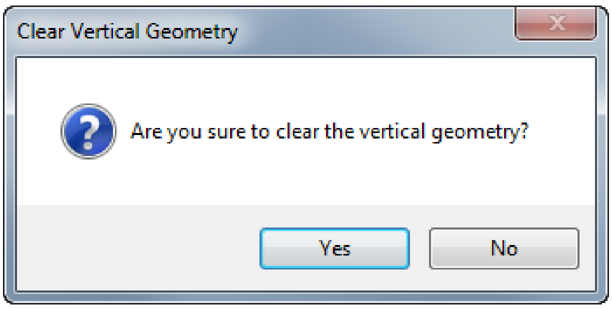


The **Clear VG** and **Clear HG** options are used to delete all the horizontal and/or vertical geometry points in the super alignment string.

14.4.11.10.1 SA - Clear VG

The **Clear VG** option removes all the vertical geometry from the string.

The horizontal geometry still exists plus all the other string properties (model, name, colour and type).

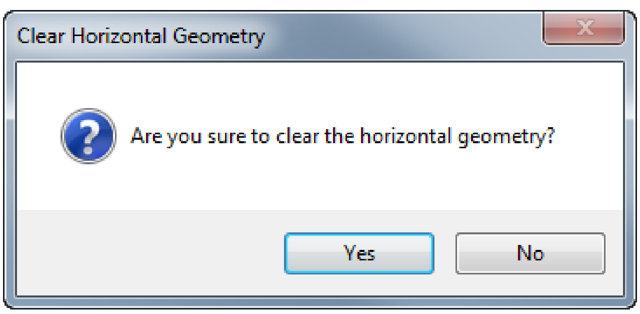


After a **Clear VG**, the vertical geometry can be re- entered using the SA Edit options.

14.4.11.10.2 SA - Clear HG

The **Clear HG** option removes all the horizontal **and** vertical geometry from the string.

The string still exists and keeps its other properties such as model, name, colour and type.



After a **Clear HG**, the horizontal and vertical geometry can be re- entered using the SA Edit options.

14.4.11.11 SA - Help Icon

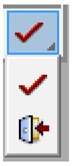
The Help icon is

Clicking on **Help** icon opens the context sensitive Help and jumps to this section.

For more information on the super alignment options, see 14.4.11 Edit Super Alignment.

14.4.11.12 SA - Finish Toolbar

The Finish toolbar is



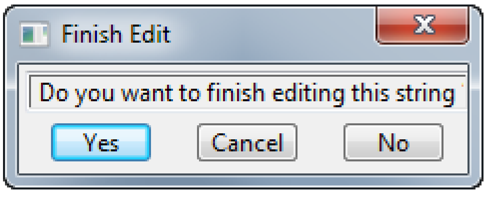
finish - save the SA and finish the editing session

quit - end the editing session without saving

14.4.11.12.1 SA - Finish

When editing of the super alignment is completed, **Finish** is used to save the modifications and exit the SA Editor.

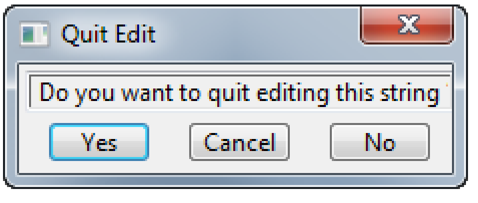
After selecting **Finish**, a **Yes-No-Cancel** panel is displayed to confirm that the edit session has ended and all the edits are to be kept. If **Yes** is selected, the edits will be stored and the SA Editor exited.



14.4.11.12.2 SA - Quit

At any time the Editing process can be aborted and all the edits thrown away by selecting the **Quit** option. That is, the super alignment is left as it was before the *Edit* was begun.

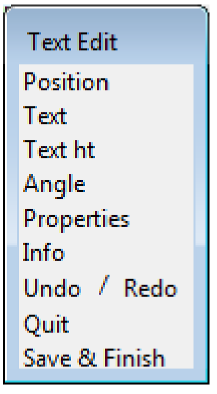
After selecting**Quit** , a**Yes-No-Cancel** panel is displayed to confirm that the edit is to be terminated. If **Yes** is selected, the edits will be ignored and the original unedited super alignment kept.

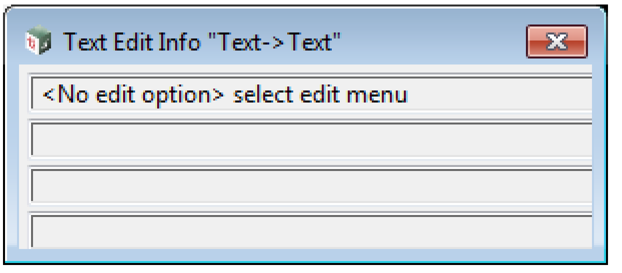


Note that if it was a new string being created, the Editor is terminated by Quit and no string is created.

14.4.12 Text Edit

On picking a text string, the **text edit** menu and the panel are placed on the screen.





Each option in the **text edit** menu will now be described.

See the earlier section 14.4.1 Super String Edit - Common Information for general information about editing strings.

14.4.12.1 Position

The **position** option is used to place the text string for the first time and to re-position (move) an existing text string.

Position is a one step process.

After selecting the **position** option, the new position for the text is selected. The text is then redrawn at the new position.

message area 1 <Position>

Screen message area

<Select point position> [picks][][menu]

<Select point position> [picks][accepts][menu]

The **position** option repeats until cancelled by the user by either bringing up the **pick ops** menu and selecting **cancel** or by selecting a new option from the **text edit** menu.

If, after bringing up the **pick ops** menu, it is decided to continue with the position option, simply select the **restart** option from the **pick ops** menu and the **pick ops** menu will disappear leaving the **position** option still current.

Typed input can be used at any stage.

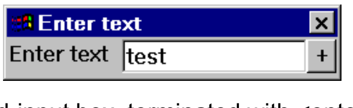
Continue to the next section 14.4.12.2 Text or return to 14.4.12 Text Edit.

14.4.12.2 Text

The **text** option is used to modify the text of the text string.

After the **text** option is chosen, an enter text typed-input box is placed on the screen with the string's current text displayed in it.

The enter text typed-input box looks like:



The text is entered into the typed-input box, terminated with <enter>. The entered value is taken as the text of the text string and the string redrawn with the new text. The typed-input box then disappears.

The **text** option automatically terminates and a new option needs to selected from the **text edit** menu.

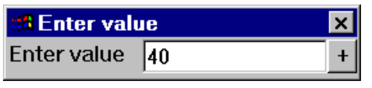
Continue to the next section 14.4.12.3 Text ht or return to 14.4.12 Text Edit.

14.4.12.3 Text ht

The **text ht** option is used to modify the height of the text in the text string.

After the **text ht** option is chosen, an enter value typed-input box is placed on the screen with the string's current text height displayed in it.

The enter value typed-input box looks like:



The text height is entered into the typed-input box, terminated with <enter>. The entered value is taken as the height of the text in the text string and the string redrawn with the new height.

The typed-input box then disappears.

The **height** option is automatically terminated and a new option needs to selected from the **text edit** menu.

Continue to the next section 14.4.12.4 Angle or return to 14.4.12 Text Edit.

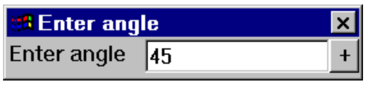
14.4.12.4 Angle

The **angle** option is used to modify the angle that the text in the text string is drawn at.

The text angle is measured in a counter- clockwise direction with respect to the horizontal axis.

After the **angle** option is chosen, an enter angle typed-input box is placed on the screen with the string's current text angle displayed in it.

The enter angle typed-input box looks like:



The text angle is entered into the typed-input box, terminated with <enter>. The entered value is taken as the angle of the text in the text string and the string redrawn with the new angle.

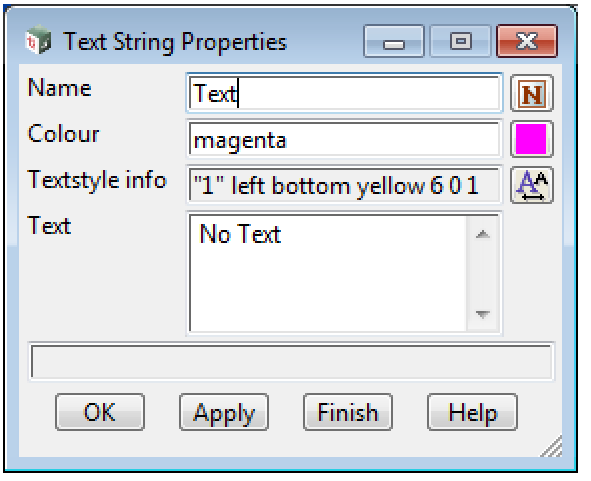
The typed-input box then disappears.

The **angle** option is automatically terminated and a new option needs to selected from the **text edit** menu.

Continue to the next section 14.4.12.5 Properties or return to 14.4.12 Text Edit.

14.4.12.5 Properties

Selecting **Properties** brings up the **Text String Properties** panel which is used to modify the string’s header information.



Field Description	Type	Defaults	Pop-Up
Name	name box		
the name of the string.			
Colour	colour box	default colour	available colours
the colour of the string.			
Textstyle info	textstyle info box		
textstyle information.			
Text	multi-line text box		
the actual text.			
OK/Apply	buttons		

for the string being edited, **OK** *sets the string with the values in the panel fields and removes the panel.* **Apply** *sets the string with the values in the panel fields and leaves the panel on the screen.*

Continue to the next section 14.5 Points Edit or return to 14.4.12 Text Edit.

14.5 Points Edit

Position of menu: **Strings =>Points Edit**

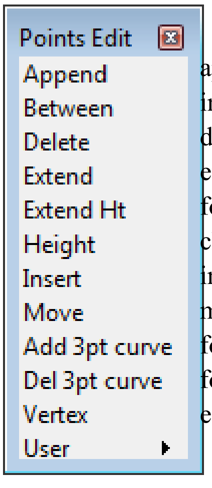
The points operations available in the **Points edit** menu are similar to the create/edit options.

However, in the **Points edit** options, the user does not begin by picking the particular string to be edited. Instead, the required option is selected followed by the point to be modified from any string.

The option can then be applied to another point on that or any other string, or another point operation chosen, and applied to any point.

If the user wants to make a large number of point edits to the one string, the **editor** option discussed in the last section is the easiest option to use. However, if a number of strings are going to be edited with the same operation or a selection of operations, then the **points edit** option is more suitable.

After selecting the **Points edit** option, the **Points edit** menu is displayed.



append to a string
insert on the line between string points
delete a string point
extend a string point
for polyline/superstring - extend point and extrapolated height
change height of a point
insert a point into a string
move a string point, or a text string
for polyline/superstring - add curve through 3 points
for polyline/superstring - delete adjacent curves
edit vertex data for a super string

Each of the options performs the same work as the option of the same name in the **Edit** string option.

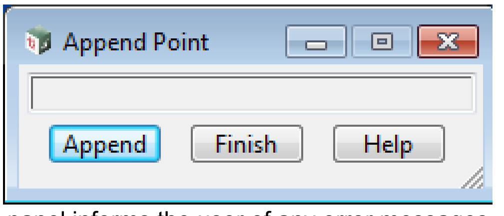
The individual options in the **Points Edit** menu will now be discussed in detail.

For the option *Append*, go to 14.5.1 Append Point
Between 14.5.2 Between Point
Delete 14.5.3 Delete Point
Extend 14.5.4 Extend Point
Extend Ht 14.5.5 Extend Height
Height 14.5.6 Height
Insert 14.5.7 Insert Point
Move 14.5.8 Move Point
Add 3pt curve 14.5.9 Add 3 Point Curve
Del 3pt curve 14.5.10 Delete 3 Point Curve
Vertex 14.5.11 Edit Vertex

14.5.1 Append Point

Position of option on menu: **Strings =>Points Edit => Append**

The **append** option adds additional points to the end of a string. On selecting the **Append** option, an **append point** panel is displayed.



The message area of this panel informs the user of any error messages.

Once in the **append** option, the particular end of the string to append points to is selected using the standard LB and MB pick and accept sequence.

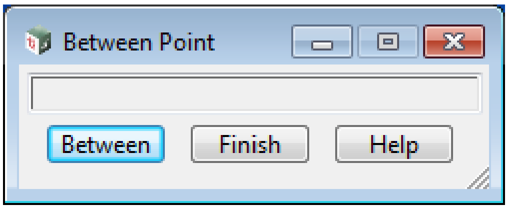
The points to be appended are then selected. The option keeps adding points to the selected string until the **pick ops** menu is brought up using RB and **cancel** is selected.

After the *append* has been completed, the option repeats. That is, the user is asked to select another string to append to. This continues until **finish** or **[X]** is selected from the **append point** panel.

14.5.2 Between Point

Position of option on menu: **Strings =>Points Edit => Between**

On selecting the **between** option, a **between point** panel is displayed.



The message area of this panel informs the user of any error messages.

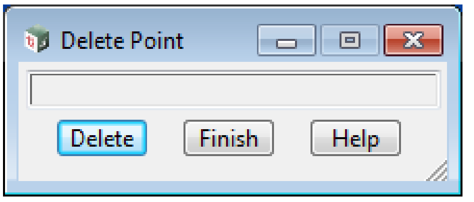
Once in the **between** option, the particular segment to insert a point on is selected using the standard LB and MB pick and accept sequence. The position of the point to insert is then selected and accepted.

After the *between* has been completed, the option repeats. That is, the user is asked to select another segment to insert a point on. This continues until **finish** or **[X]** is selected from the **between point** panel.

14.5.3 Delete Point

Position of option on menu: **Strings =>Points Edit => Delete**

On selecting the **delete** option, a **delete point** message panel is displayed.



The message area of this panel informs the user of any error messages.

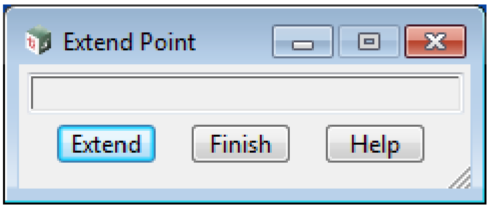
Once in the **delete** option, the particular point to delete is selected using the standard LB and MB pick and accept sequence. When the point is accepted, it is deleted.

After the *delete* has been completed, the option repeats. That is, the user is asked to select another point to delete (the point can be on any string). This continues until **finish** or **[X]** is selected from the **delete point** panel.

14.5.4 Extend Point

Position of option on menu: **Strings =>Points Edit => Extend**

On selecting the **extend** option, a **extend point** panel is displayed.



The message area of this panel informs the user of any error messages.

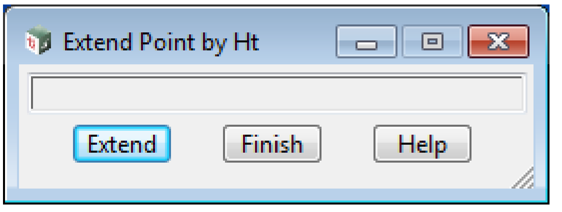
Once in the **extend** option, the particular point on a segment to extend is selected by picking the segment near to the point using the standard LB and MB pick and accept sequence. The position of the point is then moved along the segment until the final position is selected and accepted.

After the *extend* has been completed, the option repeats. That is, the user is asked to select another segment to extend a point on. This continues until **finish** or **[X]** is selected from the **extend point** panel.

14.5.5 Extend Height

Position of option on menu: **Strings =>Points Edit => Extend Ht**

The extend point by height option is similar to the extend option except that the z-value of the moved point is modified by linearly interpolating the z-value from the original points at the end of the selected segment. On selecting the **extend by ht** option, an **extend point by ht** panel is displayed.



The message area of this panel informs the user of any error messages.

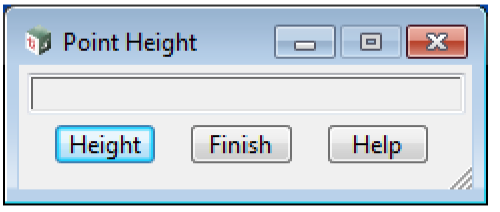
Once in the **extend point by ht** option, the particular point on a segment to extend is selected by picking the segment near to the point using the standard LB and MB pick and accept sequence. The position of the point is then moved along the segment until the final position is selected and accepted. The z-value of the moved point is the linear interpolation of the selected segment.

After the *extend point by ht* has been completed, the option repeats. That is, the user is asked to select another segment to extend a point on. This continues until **finish** or **[X]** is selected from the **extend point by ht** panel.

14.5.6 Height

Position of option on menu: **Strings =>Points Edit => Height**

On selecting the **height** option, a **point height** panel is displayed.



The message area of this panel informs the user of any error messages.

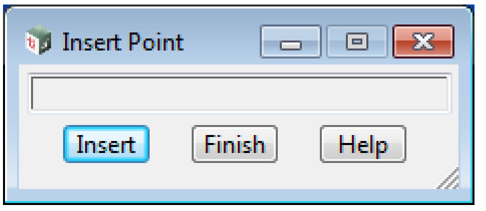
Once in the **height** option, the particular point to modify the height for is selected using the standard LB and MB pick and accept sequence. When the point is accepted, the height existing height is displayed in a **height** box which can be modified and accepted by typing <enter>.

After the *height* has been completed, the option repeats. That is, the user is asked to select another point to modify the height for (the point can be on any string). This continues until **finish** or **[X]** is selected from the **point height** panel.

14.5.7 Insert Point

Position of option on menu: **Strings =>Points Edit => Insert**

On selecting the **insert** option, an **insert point** message panel is displayed.



The message area of this panel informs the user of any error messages.

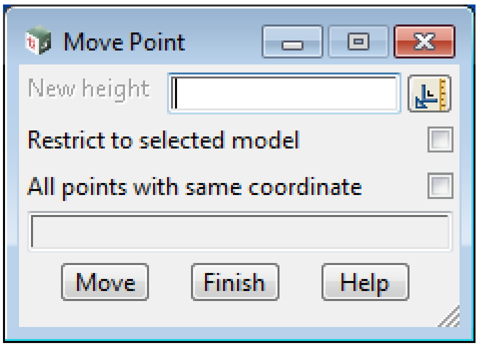
Once in the **insert** option, the particular segment to identify the two points to insert the point is selected using the standard LB and MB pick and accept sequence. The position of the point to insert is then selected and accepted.

After the *insert* has been completed, the option repeats. That is, the user is asked to select another segment to insert a point on. This continues until **finish** or **[X]** is selected from the **insert point** panel.

14.5.8 Move Point

Position of option on menu: **Strings =>Points Edit => Move**

On selecting the **move** option, a **move point** panel is displayed.



The message area of this panel informs the user of any error messages. There are similar panels for each of the other **points edit** options.

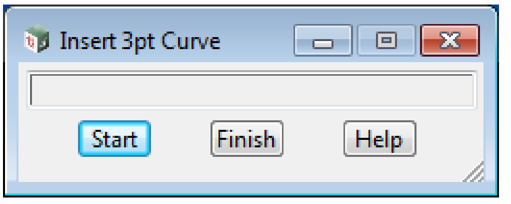
Once in the **move** option, the particular point to move is selected using the standard LB and MB pick and accept sequence. The position that the point is to be moved to is then selected and accepted.

After the move has been completed, the option repeats. That is, the user is asked to select another point to move (the point can be on any string). This continues until **finish** or **[X]** is selected from the **move point** panel.

14.5.9 Add 3 Point Curve

Position of option on menu: **Strings =>Points Edit => Add 3pt Curve**

On selecting the **add 3pt curve** option, a **insert 3pt curve** panel is displayed.



The message area of this panel informs the user of any error messages.

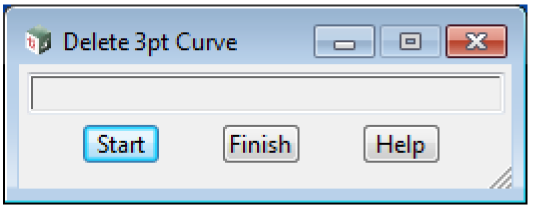
Once in the **add 3pt curve** option, the middle point of the three points of the super string to fit a curve for, is selected using the standard LB and MB pick and accept sequence. When the point is accepted, the curve is fitted between it and the left and right adjacent points of the super string.

After the *add 3pt curve* has been completed, the option repeats. That is, the user is asked to select another point to add a curve to (the point can be on any super string). This continues until **finish** or **[X]** is selected from the **insert 3pt curve** panel.

14.5.10 Delete 3 Point Curve

Position of option on menu: **Strings =>Points Edit => Del 3pt Curve**

On selecting the **del 3pt curve** option, a **delete 3pt curve** panel is displayed.



The message area of this panel informs the user of any error messages.

Once in the **Delete 3pt curve** option, a vertex from a super string is selected using the standard LB and MB pick and accept sequence and if there is curves on both sides of the vertex, the radii are removed and the vertices are joined by straight line segments.

After the *Delete 3pt curve* has been completed, the option repeats. That is, the user is asked to select another point on a super string to remove the curves on either side (the point can be on any super string). This continues until **finish** or **[X]** is selected from the **Delete 3pt curve** panel.

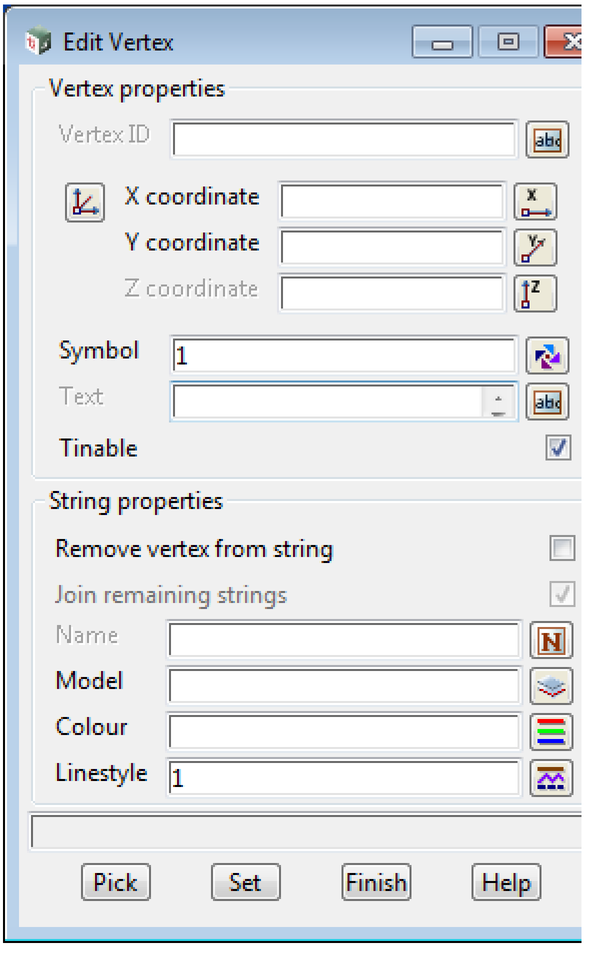
14.5.11 Edit Vertex

Position of option on menu: **Strings =>Points Edit => Vertex**

The Edit Vertex option allows the quick editing of the co-ordinates, vertex id, symbol, vertex text and tinability of a selected string vertex.

The name, model, colour and linestyle, which are string properties, can also be modified however the user has the choice of modifying the string properties for the entire string, or have the option remove the selected vertex from the string and only give the string properties to the newly created one vertex string. If a vertex is removed from the string, the adjacent vertices of the removed vertex can be joined.

On selecting the **Vertex** option, a **Edit Vertex** panel is displayed.



The **Edit Vertex** option is already running and a super string vertex is selected.

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Vertex properties

Vertex id input

the vertex id (point number) for the selected vertex. Change this to the required value.

X coordinate Measure X panel

the x coordinate (easting) of the selected vertex. Change this to the required value.

Y coordinate Measure Y panel

the y coordinate (northing) of the selected vertex. Change this to the required value.

Z coordinate Measure Z panel

the y coordinate (elevation) of the selected vertex. Change this to the required value.

Symbol symbol box available symbols

the symbol on the selected vertex. Change this to the required symbol.

Text input

the text on the selected vertex. Change this to the required text.

Tinable tick box

ticked if the selected vertex is tinable. Change if required.

String/New vertex properties

Remove vertex from string tick box

tick if the selected vertex is to be removed from its current string.

Join remaining points tick box

*only setable if **Remove vertex from string** is ticked.*
If ticked then the vertices on either side of the removed vertex are joined.
If not ticked then the original string will be broken into two strings when the vertex is removed.

Name input

the name of the string containing the selected vertex. Change this to the required name.

Model model box available models

the model of the string containing the selected vertex. Change this to the required model.

Colour colour box available colours

the colour of the string containing the selected vertex. Change this to the required colour.

Linestyle linestyle box

the linestyle of the string containing the selected vertex. Change this to the required linestyle.

Pick button

chose Pick and then select the vertex to be edited.

Set button

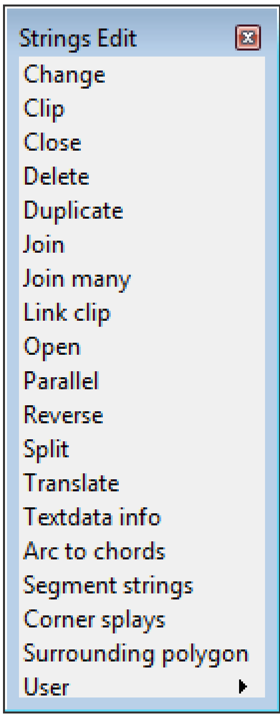
apply the values in the panel to the selected vertex. Pick is still active so that another vertex can be selected.

14.6 Strings Edit

Position of menu: **Strings =>Strings Edit**

The **Strings edit** option contains operations to be applied, not to individual points of a string, but to entire strings.

After selecting the **Strings edit** option, the **Strings edit** menu is displayed.



change string colour, type, name or model
delete or keep part of a string
connect the first and last points
delete a string
duplicate a string
join two strings to form one string
join many strings to form one string
delete a string link
disconnect the first and last points
parallel the string
reverse the order of string points
split a string into two strings
translate a string to a new position
change the text information
create inner and outer chords for an arc
segment a super string
create corner splay

The individual options in the **Strings edit** menu will now be discussed in detail.

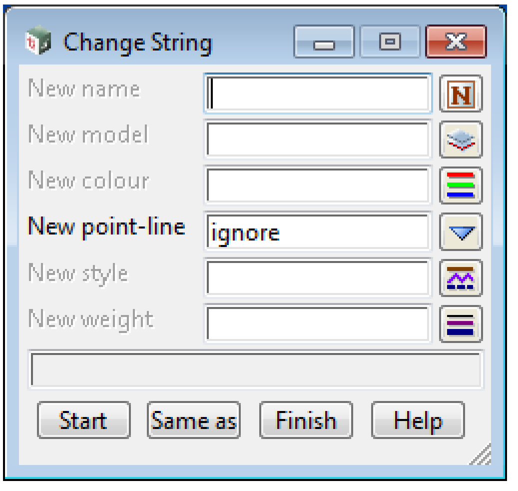
For the option *Change*, go to	14.6.1 Change
Clip	14.6.2 Clip
Close	14.6.3 Close
Delete	14.6.4 Delete
Duplicate	14.6.5 Duplicate
Join	14.6.6 Join
Join many	14.6.7 Join Many
Link clip	14.6.8 Link Clip
Open	14.6.9 Open
Parallel	14.6.10 Parallel
Reverse	14.6.11 Reverse
Split	14.6.12 Split
Translate	14.6.13 Translate
Textdata info	14.6.14 Textdata Info
Arc to chords	14.6.15 Arc to Chords
Segment strings	14.6.16 Segment Strings
Corner splays	14.6.17 Corner Splays
Surrounding polygon	14.6.18 Surrounding Polygon

14.6.1 Change

Position of option on menu: **Strings =>Strings Edit =>Change**

The **Change** option can be used to change the mode, colour, style, name and point-line type of a string.

On selecting the **Change** option, the **Change string** panel is displayed.



The **Change** option is already is progress and if a string is selected, the model, colour, style, name and point-line type are changed according to the fields in the **change string** panel.

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
New name	input		

if non-blank, the name of the selected string will be changed to the name given in the new name *field.*

New model	input		available models

if non-blank, the selected string will be moved to the model given in the new model *field.*

New colour	input		available colours

if non-blank, the colour of the selected string will be changed to the colour given in the new colour *field.*

New point-line	input		point, line

if non-blank, the breakline type of the selected string will be changed to the type given in the new point-line *field (point or line type).*

New style	input		

if non-blank, the style of the selected string will be changed to the name given in the new style *field.*

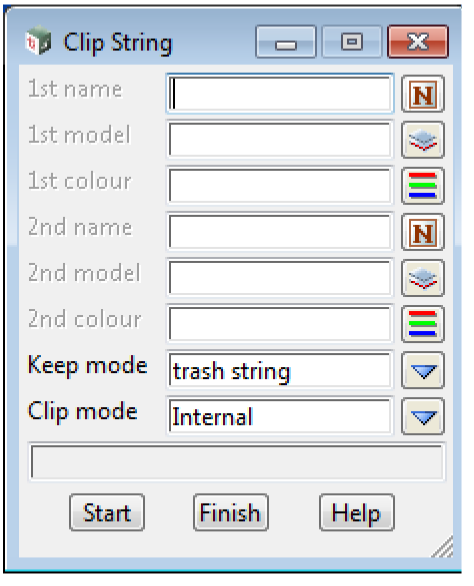
Please continue to the next section 14.6.2 Clip.

14.6.2 Clip

Position of option on menu: **Strings =>Strings Edit =>Clip**

The **Clip** option can be used to delete or keep part of a string.

On selecting the Clip option, the **Clip string** panel is displayed.



After selecting the **clip** option, the user is asked to select the string to be clipped. The user is then asked to select two points on the string to define the section to be clipped.

Depending on the clip mode, either

(a) a new string is created from the section of the string between the two points

(b) two new strings are created by deleting the section of the string between the two points

(c) one new string is created by deleting the section of the string between the two points and then joining the two selected points.

If the 1st name, 1st model and/or 1st colour panel fields are non-blank, the values are used for the 1st half of the clipped string. Otherwise the values of the original string are used for the 1st half of the string.

If the 2nd name, 2nd model and/or 2nd colour panel fields are non-blank, the values are used for the second half of the clipped string. Otherwise the values of the original string are used for the second half of the string.

If the keep mode is set to **keep string**, the original string is kept
trash string, the original string is moved to the trash model
delete string, the original string will be deleted.

Hence, the fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
1st name	input		

if blank, the first half of the clipped string is given the name of the original string.
If non-blank, the first half of the clipped string is given the name in the 1st name *field.*

1st model	input		available models

if blank, the first half of the clipped string is placed in the same model as the original selected string.
If non-blank, the first half of the clipped string is placed in the model given in the 1st model *field.*

1st colour input available colours

if blank, the first half of the clipped string is given the colour of the original string. If non-blank, the first half of the clipped string is given the colour specified in the 1st colour *field.*

2nd name input

if blank, the second half of the clipped string is given the name of the original string. If non-blank, the second half of the clipped string is given the name in the 2nd name *field.*

2nd model input available models

if blank, the second half of the clipped string is placed in the same model as the original selected string. If non-blank, the second half of the clipped string is placed in the model given in the 2nd model *field.*

2nd colour input available colours

if blank, the second half of the clipped string is given the colour of the original string. If non-blank, the second half of the clipped string is given the colour specified in the 2nd colour *field.*

Keep mode input keep string delete, keep, trash string

if ***delete string****, the string selected to be clipped is deleted.*
keep string*, the selected string is not deleted.*
trash string*, the string selected to be clipped is moved to the trash model.*

Clip mode input internal internal, external, join external

if ***internal****, a new string is created from the section of string between the two selected points.*
external*, two new strings are created by deleting the section of the string between the two points.*
join external*, one new string is created by deleting the section of the string between the two points and then joining the two selected points.*

How to Use the Panel and Panel Messages

(a) Enter the new model, colour and name for the first and second parts of the clipped string.

(b) The string to be clipped is selected.

(c) The points to be used as the clipped point are then selected.

(d) The section of the string is then clipped between the selected points and depending on the clip mode, the selected string is then clipped or the clipped section kept.

Please continue to the next section 14.6.3 Close.

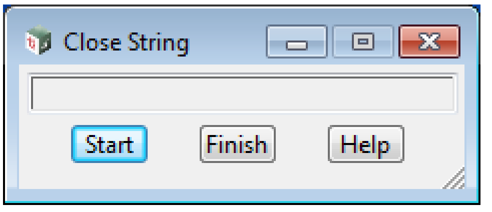
14.6.3 Close

Position of option on menu: **Strings =>Strings Edit =>Close**

A closed string is simply a string whose 1st and last points are the same.

The **close** string option adds to the end of the string, a point that is identical to the 1st string point. This then forms a closed string.

On selecting the **close** option, the **close string** panel is displayed.



The **close** option is already is progress and if a string is selected, it is closed.

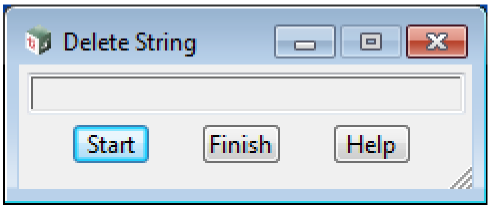
Please continue to the next section 14.6.4 Delete.

14.6.4 Delete

Position of option on menu: **Strings =>Strings Edit =>Delete**

Entire strings can be deleted from the model using the **Delete** option.

On selecting the **Delete** string option, the **Delete string** panel is displayed.



The **delete** option is already in progress and if a string is selected (LB) and accepted (MB), it is deleted

If the trash mode in the **trash defaults** panel is set to **trash string**, the original string is not deleted but moved to the trash model.

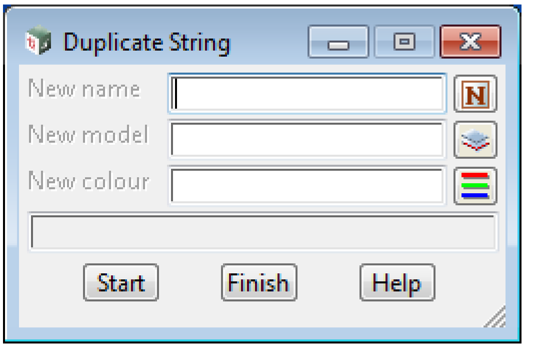
Please continue to the next section 14.6.5 Duplicate.

14.6.5 Duplicate

Position of option on menu: **Strings =>Strings Edit =>Duplicate**

The **duplicate** option is used to make a copy of the string with the option of giving the duplicate string a new model, colour and/or name.

On selecting the **duplicate** option, the **duplicate string** panel is displayed.



The **duplicate** option is already is progress and if a string is selected, then a copy of the string will be created and possibly given a new model, colour or name depending on the fields in the **duplicate string** panel.

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
New name	input		

if the new name field is blank, the duplicate string has the same name as the original string. If the new name field is non-blank, then the duplicate string is given the name in the new name field.

New model	input		available models

if new model *is blank, the duplicate string is placed in the same model as the original string. If the* new model *field is non-blank, then the duplicate string is placed in the model given in the* new model *field.*

New colour	input		available colours

if the new colour *field is blank, the duplicate string has the same colour as the original string. If the* new colour *field is non-blank, then the duplicate string is given the colour specified in the* new colour *field.*

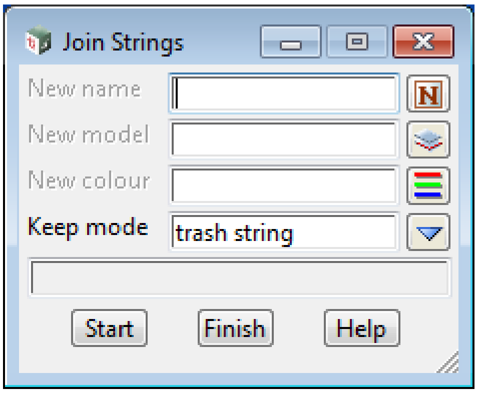
Please continue to the next section 14.6.6 Join.

14.6.6 Join

Position of option on menu: **Strings =>Strings Edit =>Join**

The **Join** option is used to create a new string by joining two existing strings together.

On selecting the **Join** option, the **Join strings** panel is displayed.



After selecting the **Join** option, the user is asked to select in turn the two strings that make up the joined string.

First, the string that will become the first half of the new joined string is selected with the direction that the string will have as the first half of the new string.

Next, the string that will become the second half of the new joined string is selected with the direction that the string will have as the second half of the new string.

The end of the first selected directed string and the beginning of the second selected directed string are joined to form the new joined string.

If the model, colour and/or name panel fields are non-blank, their values are used for the new joined string. Otherwise the values from the first selected string are used for the joined string.

If the keep mode is set to **keep string**, the original strings are kept
trash string, the original strings are moved to the trash model
delete string, the original strings are deleted.

Hence, the fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
New name	input		

if the new name *field is blank, the joined string is given the name of the first selected string. If the* new name *field is non-blank, the joined string is given the name in the* new name *field.*

Field Description	Type	Defaults	Pop-Up
New model	input		available models

if new model *is blank, the joined string is placed in the same model as the first selected string. If the* new model *field is non-blank, the joined string is placed in the model given in the* new model *field.*

Field Description	Type	Defaults	Pop-Up
New colour	input		available colours

if the new colour *field is blank, the joined string is given the colour of the first selected string. If the* new colour *field is non-blank, then the joined string is given the colour specified in the* new colour *field.*

Field Description	Type	Defaults	Pop-Up
Keep mode	input	keep strings	delete string, keep strings

if ***delete string****, the selected strings selected are deleted.*
keep string*, the selected strings are not deleted.*
trash string*, the strings selected to be joined are moved to the trash model.*

How to Use the Panel

(a) Enter the new model, colour and name for the joined string.

(b) The 1st string is selected by picking it with the required direction.
(c) The string to be joined to the 1st selected string is selected by picking it with the required direction.
(d) The last point of the 1st directed string and the 1st point of the second directed string are joined to create a new string with model, colour, name as specified in the **join strings** panel.

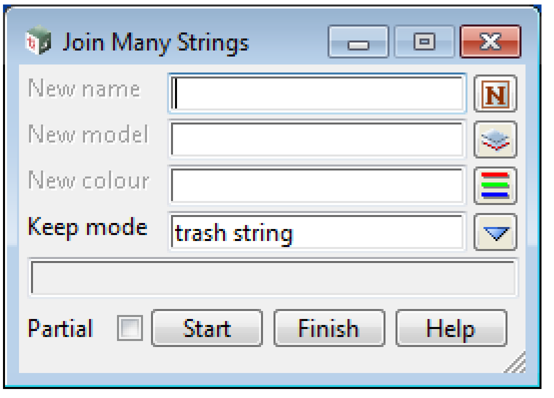
Please continue to the next section 14.6.7 Join Many.

14.6.7 Join Many

Position of option on menu: **Strings =>Strings Edit =>Join many**

The **Join many** option is used to create a new string by joining two or more existing strings, or parts or strings, together.

On selecting the **Join many** option, the **Join many strings** panel is displayed.



After selecting the **Join many** option, the user is asked to select in turn the strings that will make up the joined string. Using the partial mode, parts of strings can be joined.

First, the string that will become the first part of the new joined string is selected with the direction that the string will have as the first part of the new string.

If partial is set (tick), two points are then picked on the string and only the part of the string between the picked points is used in the join.

Next, the strings that will become the subsequent parts of the new joined string are selected in order with the direction that the strings will have in the new joined string.

Again, whenever partial is set (tick), two points are picked on the selected string and only the part of the string between the picked points is used in the join.

The end of each selected directed string (or partial string) is joined to the beginning of the subsequent selected directed string (or partial string) when forming the joined string.

If the name, model and/or colour panel fields are non-blank, their values are used for the new joined string. Otherwise the values from the first selected string are used for the joined string.

If the keep mode is set to **keep string**, the original strings are kept
trash string, the original strings are moved to the trash model
delete string, the original strings are deleted.

Hence, the fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
New name	input		

if blank, the joined string is given the name of the first selected string.
If non-blank, the joined string is given the name in the new name *field.*

Field Description	Type	Defaults	Pop-Up
New model	input		available models

if blank, the joined string is placed in the same model as the first selected string.
If non-blank, the joined string is placed in the model given in the **new model** *field.*

Field Description	Type	Defaults	Pop-Up
New colour	input		available colours

if blank, the joined string is given the colour of the first selected string.
If non-blank, the joined string is given the colour specified in the new colour *field.*

Field Description	Type	Defaults	Pop-Up
Keep mode	input	keep strings	delete strings, keep strings

if ***delete string****, the selected strings are deleted.*
keep string*, the selected strings are not deleted.*

***trash string**, the strings selected to be joined are moved to the trash model.*

Partial tick

*if **ticked**, two points are then picked on the string and only the part of the string between the picked points is used in the join.*
*if **not ticked**, no extra points are required and the entire string is used.*

How to Use the Panel and Panel Messages

(a) Enter the new model, colour and name for the joined string.

(b) The first string to be joined is then selected by picking the required string with the required direction, plus two points to restrict the string if partial is set on.

(c) The second and subsequent strings to be joined are then selected (or partial string if partial is set on) in order and with the required direction.

(d) The last point of each directed string is connected to the first point of the subsequent directed string to create a new joined string. The model, colour and name as specified in the **join many strings** panel are used for the new string.

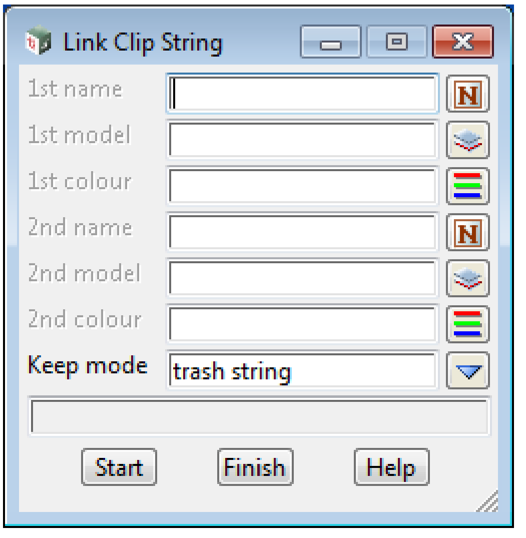
Please continue to the next section 14.6.8 Link Clip.

14.6.8 Link Clip

Position of option on menu: **Strings =>Strings Edit =>Link clip**

The **link clip** option can be used to delete the link joining two adjacent vertices of a string.

On selecting the link clip option, the **link clip string** panel is displayed.



After selecting the **link clip** option, the user selects the link of the string to be clipped. On acceptance, the selected link is deleted from the string, thus creating two new strings.

If the 1st name, 1st model and/or 1st colour panel fields are non-blank, the values are used for the first half of the link clipped string. Otherwise the values of the original string are used for the first half of the string.

If the 2nd name, 2nd model and/or 2nd colour panel fields are non-blank, the values are used for the second half of the link clipped string. Otherwise the values of the original string are used for the second half of the string.

If the keep mode is set to **keep string**, the original string is kept
trash string, the original string is moved to the trash model
delete string, the original string will be deleted.

Hence, the fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
1st name	input		

if blank, the first half of the clipped string is given the name of the original string.
If non-blank, the first half of the clipped string is given the name in the 1st name *field.*

1st model	input		available models

if blank, the first half of the clipped string is placed in the same model as the original selected string.
If non-blank, the first half of the clipped string is placed in the model given in the 1st model *field.*

1st colour	input		available colours

if blank, the first half of the clipped string is given the colour of the original string.
If non-blank, the first half of the clipped string is given the colour specified in the 1st colour *field.*

2nd name	input		

if blank, the second half of the clipped string is given the name of the original string.
If non-blank, the second half of the clipped string is given the name in the 2nd name *field.*

2nd model	input		available models

if blank, the second half of the clipped string is placed in the same model as the original selected string. If non-blank, the second half of the clipped string is placed in the model given in the 2nd model *field.*

2nd colour		input	available colours

if blank, the second half of the clipped string is given the colour of the original string.
If non-blank, the second half of the clipped string is given the colour specified in the 2nd colour *field.*

Keep mode	input	keep string	delete, keep, trash string

if ***delete string****, the string selected to be clipped is deleted.*
keep string*, the selected string is not deleted.*
trash string*, the string selected to be clipped is moved to the trash model.*

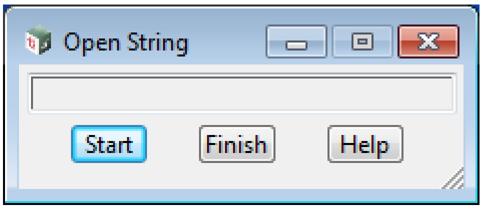
Please continue to the next section 14.6.9 Open.

14.6.9 Open

Position of option on menu: **Strings =>Strings Edit =>Open**

A closed string is opened by removing the last point of the string.

On selecting the **open** option, the **open string** panel is displayed.



The **open** option is already is progress and if a closed string is selected, it is opened.

Please continue to the next section 14.6.10 Parallel.

14.6.10 Parallel

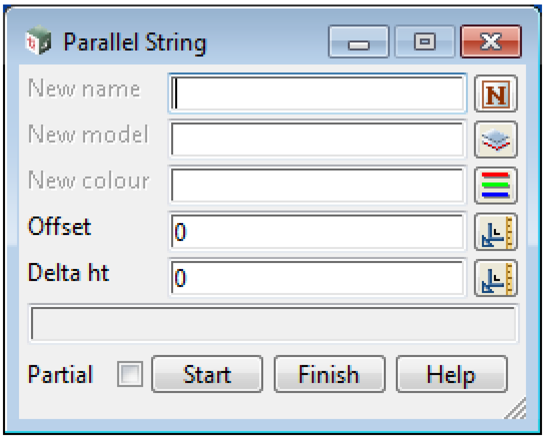
Position of option on menu: **Strings =>Strings Edit =>Parallel**

Strings, or parts of strings, can be translated perpendicularly to either the left or right using the **parallel** option. The z-values of the string can also be adjusted by a constant value.

The difference between a **translate** and a **parallel** is that for a straight **translate**, the translation vector is the same for the whole string. In a **parallel**, the translation is at right angles to each link of the string and hence the translation direction varies along the string.

For example, the edge of a road is a parallel of the centre-line, not a copy of the centre-line.

On selecting the **parallel** option, the **Parallel String** panel is displayed.



The user is then asked to select the string to parallel.

If partial is not set, the string is copied parallel through the distance given in the offset field.

If partial is set (tick), two points are then picked on the string and only the part of the string between the picked points is copied parallel through the distance given in the offset field.

The value in the delta ht field is then added to the z-values of the string.

The direction imposed upon the string when selecting it is used to determine what is the left and right side of the string in the **parallel** operation.

A positive offset parallels the string to the right with respect to the direction of picking. A negative offset parallels to the left with respect to the direction of picking.

Hence, the fields and buttons used in the **parallel string** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
New name	input		

if blank, the paralleled string is given the name of the original string.
If non-blank, the parallel string is given the name in the new name *field.*

New model	input		available models

if blank, the paralleled string is placed in the same model as the original selected string.
If non-blank, the paralleled string is placed in the model given in the new model *field.*

New colour	input		available colours

if blank, the paralleled string is given the colour of the original string.
If non-blank, the paralleled string is given the colour specified in the new colour *field.*

Offset	input		

distance (in world units) that the string will be copied parallel through. A positive distance denotes that the string will be paralleled to the right of the original string.

Delta ht	input		

value to add to the z-values of the string.

Partial tick

*if **ticked**, two points are then picked on the string and only the part of the string between the picked points is used in the parallel.*
*if **no ticked**, no extra points are required and the entire string is used.*

How to Use the Panel

(a) Enter the new model, colour and name for the paralleled string.

(b) Enter the offset distance for the parallel - left and right is determined by picking direction.

(c) Enter the height to add to the string’s z-values in the delta ht field.

(d) The selected string (or partial string if partial is set on) is then copied parallel as required.

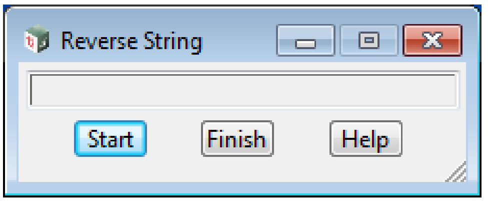
Please continue to the next section 14.6.11 Reverse.

14.6.11 Reverse

Position of option on menu: **Strings =>Strings Edit =>Reverse**

The **reverse** string option simply reverses the order of the points in a string.

On selecting the **reverse** option, the **reverse string** panel is displayed.



After selecting the **reverse** option, the user simply picks (LB) the string that is to have its point order reversed and on acceptance (MB) of the string, the reversing takes place.

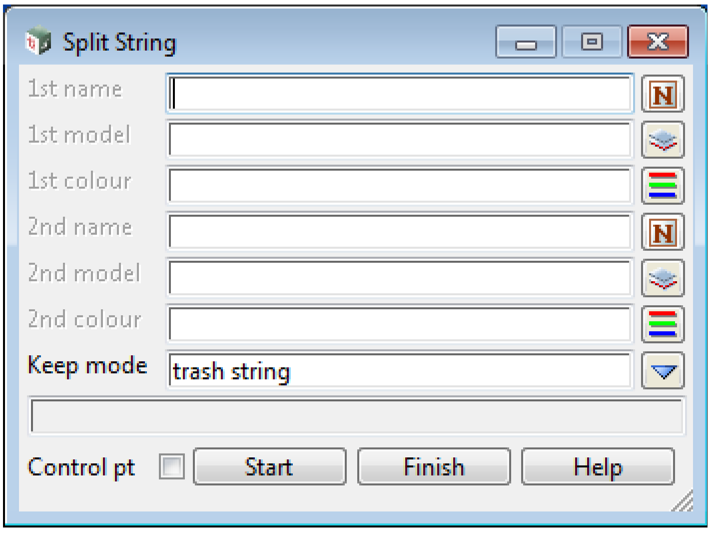
Please continue to the next section 14.6.12 Split.

14.6.12 Split

Position of option on menu: **Strings =>Strings Edit =>Split**

The **Split** option is used to create two new strings by splitting an existing string about a selected point on the string.

On selecting the **Split** option, the **Split string** panel is displayed.



If **control point** is set **off**, after selecting the **split** option, the user is asked to select the string and the selection point is **also** used as a split point. Hence the one point is used to select the string and as the split point.

If **control point** is set **on**, the user is asked to select the string to split and then pick the point that is dropped perpendicularly onto the selected string as the point on the string to be used as a split point. Hence the **string select** and the **split point** are selected separately.

Two strings are then created by splitting the chosen string about the split point.

If the 1st name, 1st model and/or 1st colour panel fields are non- blank, the values are used for the first half of the split string. Otherwise the values of the original string are used for the first half of the string.

If the 2nd name, 2nd model and/or 2nd colour panel fields are non-blank, the values are used for the second half of the split string. Otherwise the values of the original string are used for the second half of the string.

If the keep mode is set to **keep string**, the original string is kept
trash string, the original string is moved to the trash model
delete string, the original string will be deleted.

Hence, the fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
1st name	input		

if blank, the first half of the split string is given the name of the original string.
If non-blank, the first half of the split string is given the name in the 1st name *field.*

1st model	input		available models

if blank, the first half of the split string is placed in the same model as the original selected string.
If non-blank, the first half of the split string is placed in the model given in the 1st model *field.*

1st colour	input		available colours

if blank, the first half of the split string is given the colour of the original string.
If non-blank, then the first half of the split string is given the colour specified in the 1st colour

field.

2nd name input

if blank, the second half of the split string is given the name of the original string.
If non-blank, the second half of the split string is given the name in the 2nd name *field.*

2nd model input available models

if blank, the second half of the split string is placed in the same model as the original selected string.
If non-blank, the second half of the split string is placed in the model given in the 2nd model *field.*

2nd colour input available colours

if blank, the second half of the split string is given the colour of the original string.
If non-blank, then the second half of the split string is given the colour specified in the 2nd colour *field.*

Keep mode input keep string delete, keep, trash string

if ***delete string****, the string selected to be split is deleted.*
keep string*, the selected string is not deleted.*
trash string*, the string selected to be split is moved to the trash model.*

Control pt tick

If ***ticked****, the user is asked to select the string to split and then pick the point that is dropped perpendicularly onto the selected string as the point on the string to be used as a split point.*
If ***not ticked****, the user is asked to select the string and the selection point is* ***also*** *used as a split point.*

How to Use the Panel and Panel Messages

(a) Enter the new model, colour and name for the first and second parts of the split string.

(b) If control is on, the string is selected and then the split point is selected.
If control is off, the string is selected and the same point is to be used as the split point.

(c) The chosen string is then split about the split point and the two halves given models, colours and names as defined by the split string panel.

Please continue to the next section 14.6.13 Translate.

14.6.13 Translate

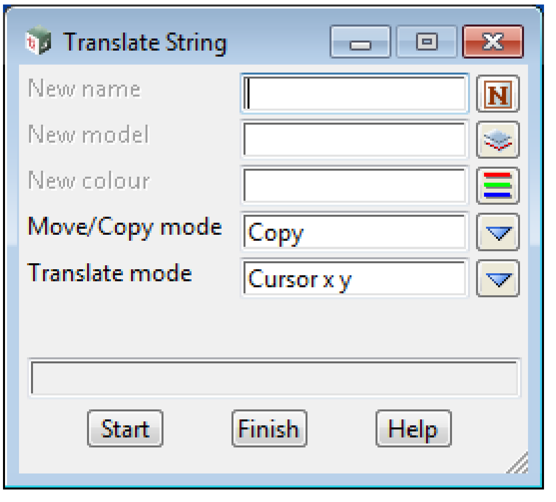
Position of option on menu: **Strings =>Strings Edit =>Translate**

Entire strings can be translated in the x,y and z directions using the **translate** option.

The translated string can either be moved or copied to its new translated position.

The difference between a **copy** or a **move** is simply that if a string is copied, a copy of the string is translated and the original string left untouched, whereas for a move, the actual string is moved from its original position to the new translated position.

On selecting the **Translate** option, the **Translate string** panel is displayed.



The selected string is translated by using the cursor to select a position on the string (the before point) and then selecting the position (not necessarily on a string) that the point will be translated to (after point).

Hence the translate operation requires two positions - a before and an after point - for the translation to be defined.

The first position (the before point) is selected using the normal select procedures. The second position (the after point) depends entirely on the translate mode.

If the translate mode is set to

cursor x y z or **cursor x y**

the cursor is used to not only select the string point but also to indicate where it is to be moved/copied to.

typed x y z

the second point has the absolute coordinates given in the x y z field.

typed dx dy dz

the second point is defined relative to the first point by the adding the dx dy dz given in the dx dy dz field.

Selected strings will be translated until the option is finished.

The fields and buttons used in the **translate** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
New name	input		

if blank, the string is given the name of the original string.
If non-blank, the string is given the name in the new name *field.*

New model	input		available models

if blank, the string is placed in the same model as the original selected string.
If non-blank, the string is placed in the model given in the New model *field.*

New colour	input		available colours

if blank, the string is given the colour of the original string.
If non-blank, then the string is given the colour specified in the new colour *field.*

Translate mode	input	cursor xy	cursor xy, cursor xyz, typed xyz, typed dxdydz

see comments about translate mode in the general description of move/copy

x y z	input		

if Translate mode *is* ***Typed x y z*** *then this field appears and the position on the string where the string was selected is translated to this co-ordinate.*

dx dy dz	input		

if Translate **mode** *is* ***Typed dx dy dz*** *then this field appears and the selected string is translated by the given dx dy dz.*

Move/copy mode	input	move	copy, move

if the mode is set to ***move,*** *the string is translated as defined in the panel fields and the original string is deleted. If the mode is* ***copy,*** *the original string is left alone.*

How to Use the Panel and Panel Messages

(a) Enter the new model, colour and name for the translated string.

(b) The string to be translated and its “before” point is chosen.

(c) The “after” point is chosen according to the translate mode.

(d) The selected string is then translated (moved or copied) as required.

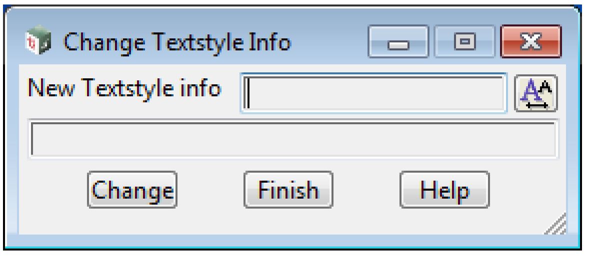
Please continue to the next section 14.6.14 Textdata Info.

14.6.14 Textdata Info

Position of option on menu: **Strings =>Strings Edit =>Textdata info**

The **Textdata info** option is used to change the information about the text for a text string, super strings and text for 4d strings.

Selecting the option, the **Change Textstyle Info** panel:



The parameters for the textstyle information are set and then any selected string is given the new textstyle parameters. If any of the parameters are left blank, then that parameter is not modified for the selected string.

To define the Textstyle info parameters, click on the text button and then the **edit** button.

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
New textstyle info	textstyle data		

textstyle information to use for the selected strings.

Change	button		

selected strings will have their text modified by the parameters in the new textstyle info *field.*

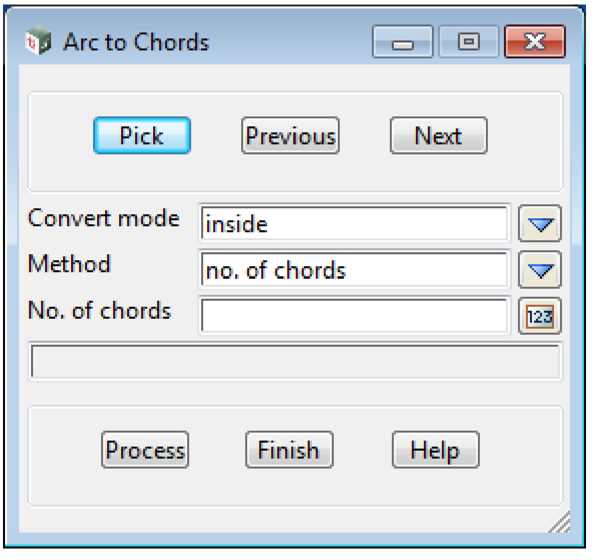
14.6.15 Arc to Chords

Position of option on menu: **Strings =>Strings Edit =>Arc to chords**

The *Arc to Chord* option works on a super string and creates inside or outside chords for any arc segments in the super string.

The number of chords created can be defined by giving the number of chords required, a chord length to use for each chord, or an arc to chord tolerance.

On selecting the **Arc to chords** option, the **Arc to Chords** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		
select the super string to edit.			
Previous	button		
move to the previous segment of the super string.			
Next	button		
move to the next segment of the super string.			
Convert mode	choice box	inside	inside outside
create the chords on the inside or the outside of the arc.			
Method	choice box	no. of chords	no. of chords chord length arc to chord tolerance
method for creating the chord.			
No. of Chords, Chord length, Arc to chord tolerance			
value to use with the Method.			
Process	button		
create chords for the selected segment of the super string.			

14.6.16 Segment Strings

Position of option on menu: **Strings =>Strings Edit =>Segment string**

The *Segment string* option works on a super string and inserts extra points on an arc or line segment of the super string to break it into smaller arc or line segments.

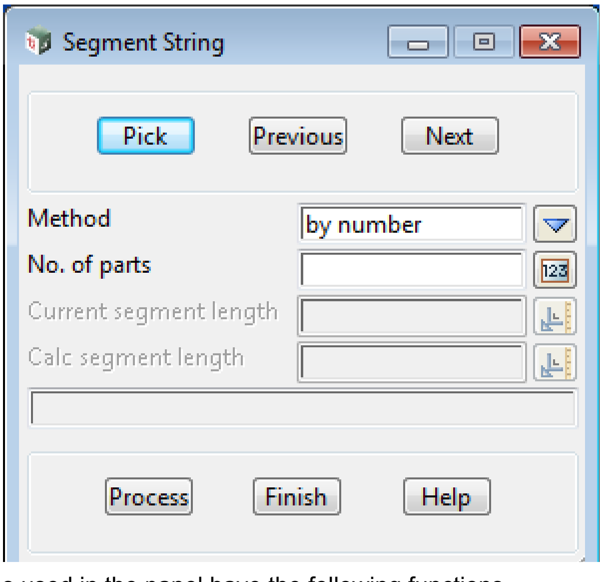
The number of arcs/lines created can be controlled by specifying either

(a) the number of sub-segments required

(b) a single sub-segment length to break the start of the segment into that length

(c) a sub-segment length to break the segment into as many sub-segments of that length as possible.

On selecting the **Segment strings** option, the **Segment String** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

select the super string to edit.

Field Description	Type	Defaults	Pop-Up
Previous	button		

move to the previous segment of the super string.

Field Description	Type	Defaults	Pop-Up
Next	button		

move to the next segment of the super string.

Field Description	Type	Defaults	Pop-Up
Method	choice box	by number	by number one distance by distance

method for creating the sub-segments.

*by number - the segment is broken into **No of parts** equal segments*
*one distance - the segment is broken into two parts with the first part having the length **Distance** (which end will depend on the direction that the segment was selected.*
*by distance - the segment is broken into as many sub-segments of length **Distance** as possible (which end the division starts at depends on the direction that the segment was selected).*

No of parts, Distance

value to use with the Method.

Current segment length output only

length of the currently selected segment of the super string.

Calc segment length output only

calculated length of the segment that will be created.

Process button

create sub-segments for the selected segment of the super string.

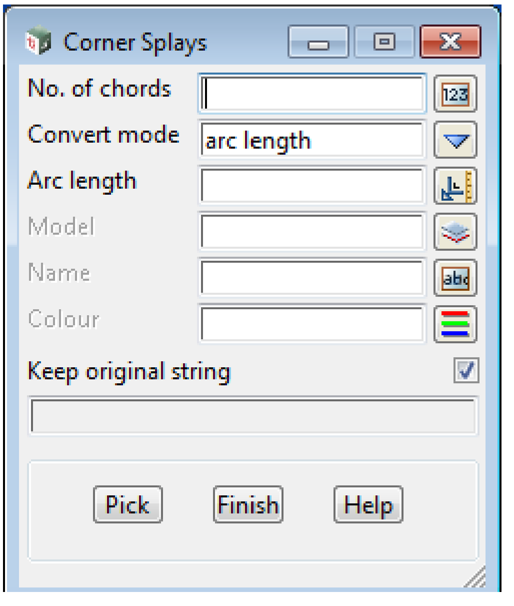
14.6.17 Corner Splays

Position of option on menu: **Strings =>Strings Edit =>Corner splays**

The *Corner splays* option works on a super string and creates splays between two adjacent straight segments.

The number of chords created can be defined by giving the number of sub-segments required, one sub-segment length to beak of the segment or a sub-segment length to beak the segment into as many sub-segments as possible.

On selecting the **Corner splays** option, the **Corner splays** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
No of chords			

value to use with the Method.

Convert mode	choice box	arc length	arc length arc radius chord length total chords length tangent length

method for creating the corner splays.

Arc length, Arc radius, Chord length, Total chords length, tangent length

value to use with the Convert mode.

Name	input		

the name of the new string.

Model	input		available models

name of the model that the new string is in.

Colour	input	default colour	available colours

the colour of the new string.

Keep original string	tick box	tick	

if ticked, the original string is not modified and a new string is created with the corner splays. If not ticked, the original string is modified.

Pick button

select a super string and the splays will be created.

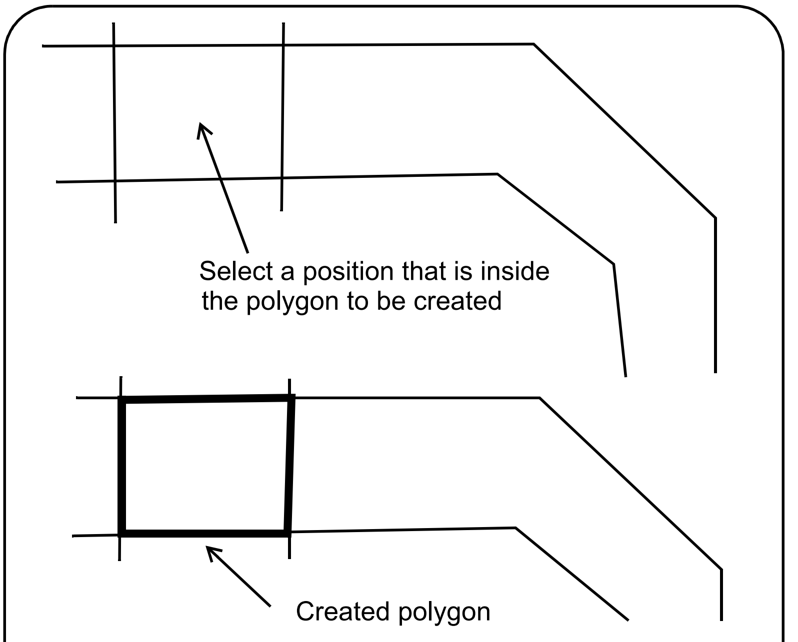
14.6.18 Surrounding Polygon

Position of option on menu: **Strings =>Strings Edit =>Surrounding polygon**

The *Surrounding polygon* option works on all the super strings on the view surrounding a selected position (point) and, if possible, creates a closed polygon from parts of the strings surrounding the selected point.

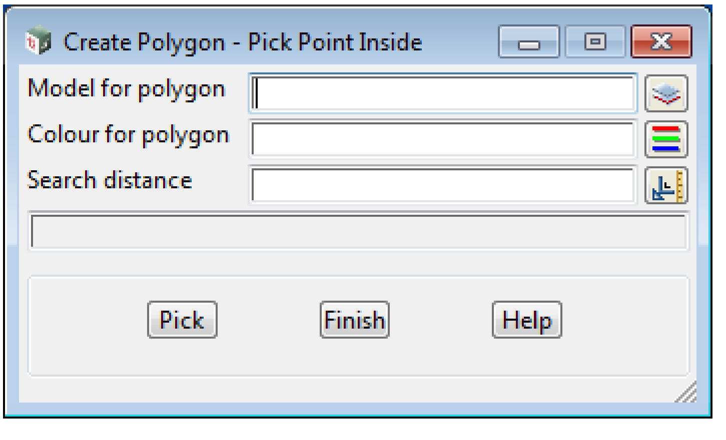
A straight line must be able to be drawn from the selected point to the lines/arcs that could be used in the surrounding strings.

This option creates a lot/polygon by a picking inside a collection of strings and the lot/polygon is created from the closest strings to the picked position. The picked position must be selected so that all sides of the lot/polygon can be "seen" from the picked position. That is, a straight line can be drawn from the picked position to the lot/polygon side without cutting any other segment.



Creating a Polygon by Picking Inside the Polygon

On selecting the **Surrounding polygon** option, the **Create Polygon - Pick Point Inside** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input		

the name of the new created polygon.

Model	input		available models

model for the created polygon.

Colour	input	default colour	available colours

colour for the created polygon.

Pick	button		

pick a position on a view and the surrounding polygon will be created from all the super strings on the view.

14.7 Cogo

Position on menu: **Strings =>Cogo**

Note: This option is no longer available.

The **Cogo (co-ordinate geometry)** options provide methods of constructing points, lines and arcs from a variety of operations such as picking, filleting, intersecting and finding tangents.

Selecting the **cogo** option creates the **Cogo edit ops** menu and the **Cogo edit** panel.

The options in the **cogo edit ops** menu all create either a point, string or arc and the **cogo edit** panel remains on the screen for each of the menu options and provides the string properties for any created objects.

All of the options chosen from the **Cogo edits ops** menu and its sub-menus repeat until cancelled by the user. That is, the option remains in force and can be applied repeatedly until terminated by the user.

The name of the current selected option is displayed in the first message area of the **cogo edit** panel.

To select a new **Cogo edits ops** option, simply select the new option from the **Cogo edit ops** menu or sub-menus. The current option is automatically terminated and the new option begun.

An option is also terminated when either the **cancel** or **finish** button is selected in the **Cogo edit** panel, or **finish** is elected from the **Cogo edit ops** menu.

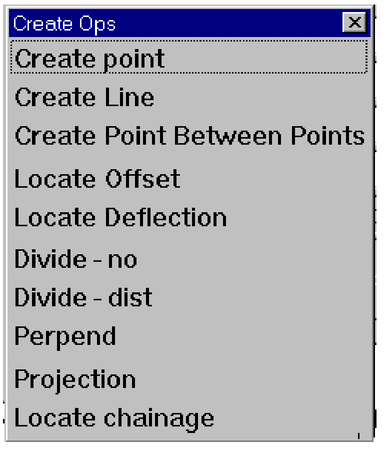
Each of the options in the **Cogo edit ops** menu will now be discussed.

For the option *Create ops*, go to 14.7.1 Create Ops
Intersect ops 14.7.2 Intersect Ops
Vector 14.7.3 Vector
Parallel 14.7.4 Parallel
Fillet ops 14.7.5 Fillet Ops
Fillet ops (T) 14.7.6 Fillet Ops (T)
Fillet ops (TJ) 14.7.7 Fillet Ops (TJ)
Tangent 14.7.8 Tangent

Please continue to the next section 14.7.1 Create Ops.

14.7.1 Create Ops

The **Create ops** walk-right menu is



Each of the options in the **Create Ops** menu will now be discussed.

For the option *Create point*, go to	14.7.1.1 Create Point
Create line	14.7.1.2 Create Line
Create point between points	14.7.1.3 Create Point Between Points
Locate offset	14.7.1.4 Locate Offset
Locate deflection	14.7.1.5 Locate Deflection
Divide - no	14.7.1.6 Divide - Number
Divide - dist	14.7.1.7 Divide - Distance
Perpedn	14.7.1.8 Perpendicular
Projection	14.7.1.9 Projection
Locate chainage	14.7.1.10 Locate Chainage

14.7.1.1 Create Point

After selecting the **Create point** option, the user simply selects a point using the normal picking sequence.

The point will be created with name, colour and model from the **cogo edit** panel.

If no z-value is given in the **cogo edit** panel, then a null z-value is used.

If height snap is on and the point is created by snapping to an existing point or line, the z-value for the point is taken from the snapped point or if tin snap is on, from the underlying triangulation. The snapped z-value will be displayed in an enter height typed input box.

If required, this value can be modified by typing into the enter height box. If the box is empty, the z-value is taken from the **cogo edit** panel. If no value exists in the **cogo edit**, an error message **please specify z value** is displayed in the **cogo edit** message area.

The z-value in the enter height box is accepted by entering a <enter> in the box. The enter height box then disappears.

Screen message area

<<Create Point> Select position for point> [picks][][menu]

14.7.1.2 Create Line

After selecting the **create line** option, the user simply selects two points to be the end points of the line using the normal picking sequence.

The line will be created with name, colour and model from the **cogo edit** panel.

If either end point is created by snapping to an existing point or line, the z-value for the point is taken from the snapped point or if tin snap is on, from the underlying triangulation. If no z-value exists, the z-value is taken from the **cogo edit** panel.

If height snap is on, the snapped z-value will be displayed in an enter height typed input box.

If required, this value can be modified by typing into the enter height box. If the box is empty, the z-value is taken from the **cogo edit** panel. If no value exists in the **cogo edit**, an error message **please specify z value** is displayed in the **cogo edit** message area.

The z-value in the enter height box is accepted by entering a <enter> in the box. The enter height box then disappears.

Screen message area

<<Create Line> Select 1st point on line> [picks][][menu]
<<Create Line> Select 2nd point on line> [picks][][menu]

14.7.1.3 Create Point Between Points

The **create point between points** option allows the user to select two point to act as the end points of a line and then create a new point a user supplied given distance along that line.

After selecting the **create point between points** option, the user simply selects two points to act as the end points of the line using the normal picking sequence.

An enter distance from 1st point box is then displayed on the screen and the user simply enters the appropriate distance into the box, terminated with a <enter>. The box then disappears.

The point is then created with name, colour and model and **z-value** from the **cogo edit** panel.

If no z-value exists in the panel, no point is created.

Screen message area

<<Between> Select 1st point> [picks][][menu]
<<Between> Select 2nd point> [picks][][menu]
<Between> Enter distance from 1st point [caret][][menu] select a button

14.7.1.4 Locate Offset

The **locate offset** option is used to create a point at a given perpendicular offset distance from a user defined point on a selected item.

The point on the item to offset from is determined by picking a control point on the item and projecting along the item by a given distance.

Hence the **locate offset** option needs

(a) a selected item

(b) a control point

(c) a projection distance

(d) an offset distance

After selecting the **locate offset** option, the user picks the item to project along using the normal picking sequence. The direction imparted when picking the item determines the sense for direction along the item and offset left and right.

Next a control point on the item is picked.

A distance along item box is then displayed on the screen and the user enters the appropriate distance into the box, terminated with a <enter>. The box then disappears. A positive distance is in the picking direction of the item.

Finally an offset distance box is displayed on the screen. The user enters the appropriate offset distance into the box, terminated with a <enter>. The box then disappears. A positive offset is to the right of the item when moving in the picking direction of the item.

The appropriate point is then created with name, colour, model and z-value from the **cogo edit** panel.

If no z-value exists in the panel, no point is created.

Screen message area

<<Locate Offset> Select item to project along> [picks][][menu]
<<Locate Offset> Select control point> [picks][][menu]
<Locate Offset> distance along item [caret][][menu] select a button
<Locate Offset> offset distance [caret][][menu] select a button

14.7.1.5 Locate Deflection

The **locate deflection** option is used to create a point at a given deflection angle and distance from a user defined point on a selected item.

The **locate deflection** option is very similar to the **locate offset** option except instead of going out perpendicular to the item, the user provides a deflection angle. Hence the **locate offset** is a special case of the **locate deflection** option when the deflection angle is ninety degrees.

The point on the item to offset from is determined by picking a control point on the item and projecting along the item by a given distance.

Hence the **locate deflection** option needs

(a) a selected item

(b) a control point

(c) a projection distance

(d) a deflection angle

(e) a deflection distance

After selecting the **locate offset** option, the user picks the item to project along using the normal picking sequence. The direction imparted when picking the item determines the sense for direction along the item and offset left and right.

Next a control point on the item is picked.

Boxes for the distance along item, deflection angle and deflection distance are then (sequentially) displayed on the screen and the user enters the appropriate values into each box, terminated with a <enter>. The box then disappears.

A positive distance is in the picking direction of the item and a positive angle is an angle to the right of the item when moving in the picking direction of the item.

The appropriate point is then created with name, colour, model and z-value from the **cogo edit** panel.

If no z-value exists in the panel, no point is created.

Screen message area

<<Locate Deflection> Select item to project along> [picks][][menu]
<<Locate Deflection> Select control point> [picks][][menu]
<Locate Deflection> distance along item [caret][][menu] select a button
<Locate Deflection> deflection distance [caret][][menu] select a button
<Locate Deflection> deflection angle [caret][][menu] select a button

14.7.1.6 Divide - Number

The **divide - nos** option is used to place points at positions which would divide an arc or a line into an equal number of pieces.

For example, if the option was used to show how and an arc could be divided into seven pieces, points would be placed at the six division marks. No point is placed at the start or end of the arc. Hence, the start and end points plus the division points divide the arc into the required number of divisions.

Since the option will only divide an arc or a line, when an item is selected only the picked line or arc sub-element of the item will be divided.

On selecting the **divide - nos** option, the user picks the item to divide using the normal picking sequence.

A no of divisions box is then displayed on the screen and the user enters the required number of divisions into the box, terminated with a <enter>. The box then disappears.

A point string is then created containing points at each of the required division positions. The string has the name, colour, model and z-value from the **cogo edit** panel.

Screen message area

<<Divide by #> Select item to divide> [picks][][menu]
<Divide by #> no division [caret][][menu] select a button

14.7.1.7 Divide - Distance

The **divide - dist** option is used to place points at positions which would divide an arc or a line into pieces of a given distance.

Note - unless the arc or line is exactly divisible by the distance, the last division will be smaller than the required distance.

Since the option will only divide an arc or a line, when an item is selected only the picked line or arc sub-element of the item will be divided.

On selecting the **divide - dist** option, the user picks the item to divide using the normal picking sequence.

A distance box is then displayed on the screen and the user enters the required distance for each division into the box, terminated with a <enter>. The box then disappears.

A point string is then created containing points at each of the required division positions. The string has the name, colour, model and z-value from the **cogo edit** panel.

Screen message area

<<Divide by distance> Select item to divide> [picks][][menu]
<Divide by #> distance [caret][][menu] select a button

14.7.1.8 Perpendicular

The **perpend** option is used to create the point which is the perpendicular projection of a selected point onto a selected line or arc.

Since the option can only drop a point onto an arc or a line, when the item is selected to project onto, only the picked line or arc sub-element of the item is used.

On selecting the **perpend** option, the user picks the point to drop perpendicularly.

The item to drop perpendicularly onto is then selected.

The point representing the perpendicular projection of the chosen point onto the selected item is then created. The created point has the name, colour, model and z-value from the **cogo edit** panel.

Screen message area

<<Perpend> Select point to drop> [picks][][menu]
<<Perpend> Select item to drop onto> [picks][][menu]

14.7.1.9 Projection

The **projection** option is used to create a point which is given distance along a line or arc.

Since the option can only project a point along an arc or a line, when the item is selected to project along, only the picked line or arc sub-element of the item is used.

On selecting the **projection** option, the user picks the item to project along.

A distance box is then displayed on the screen and the user enters the appropriate projection distance into the box, terminated with a <enter>. The box then disappears. A positive distance is in the picking direction of the item.

The point representing the projected point along the selected item is then created using the name, colour, model and z-value from the **cogo edit** panel.

Screen message area

<<Projection> Select item to project along> [picks][][menu]
<Projection> distance [caret][][menu] select a button

14.7.1.10 Locate Chainage

The **locate chainage** option is used to create a point at a given chainage on a selected string.

On selecting the **locate chainage** option, the user picks the item to locate the chainage on.

A locate chainage box is then displayed on the screen and the user enters the appropriate chainage into the box, terminated with a <enter>. The box then disappears.

A point at the given chainage on the selected item is then created using the name, colour, model and z-value from the **cogo edit** panel.

Screen message area

<<Locate chainage> Select item to locate on> [picks][][menu]
<Locate chainage> chainage [caret][][menu] select a button

Please continue to the next section 14.7.2 Intersect Ops.

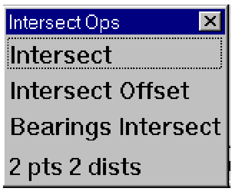
14.7.2 Intersect Ops

The **Intersection** options are for finding various intersections of lines and arcs.

Most of intersect cases can have more that one solution. For example, the intersection of a line with an arc.

To easily distinguish between the cases, 12d Model uses the sense of direction implied when picking items to determine which case was required by the user.

The **Intersect ops** walk-right menu is



Each of the options in the **Intersect Ops** menu will now be discussed.

For the option *Intersect*, go to	14.7.2.1 Intersect
Create line	14.7.1.2 Create Line
intersect offset	14.7.2.2 Intersect Offset
Bearings intersect	14.7.2.3 Bearings Intersect
2 pts 2 dists	14.7.2.4 Two Points, Two Distances

14.7.2.1 Intersect

The **Intersect** option creates a point at the intersection of lines and arcs.

That is, it will find the intersection of

(a) a line with a line

(b) a line with an arc

or

(c) an arc with an arc

After selecting the **Intersect** option, the user simply selects the two items to intersect using the normal picking sequence.

If an intersect exists, a point is created at the intersection with name, colour, model and z-value from the **Cogo edit** panel.

If no z-value exists in the panel, no point is created.

When there is more than one possible solution, the order of picking and the directions imparted when picking the items are used to determine which solution is required.

Since the option will only intersect arcs and lines, when an item is selected only the picked line or arc sub-element of the item is used in the intersect.

Screen message area

<<Intersect> Select 1st item to intersect> [picks][][menu]
<<Intersect> Select 2st item to intersect> [picks][][menu]

14.7.2.2 Intersect Offset

Like the **Intersect** option, **Intersect offset** is for creating points at the intersection of lines and arcs.

However, before the line or arc is used, it is offset by a user supplied distance.

Hence the intersection is between displaced lines and arcs.

A practical example of **intersect offset** is for finding the position of a manhole in a subdivision which is three metres from one property line and two metres from the culdesac arc.

After selecting the **intersect** option, the user selects the first item to intersect using the normal picking sequence.

A 1st offset box is then displayed on the screen and the user enters the appropriate offset for the first line into the box, terminated with a <enter>. The box then disappears. A positive offset is to the right of the item when moving in the picking direction of the item.

The second item to intersect is then selected and a 2nd offset box is displayed on the screen for the user to enter the offset for the second line.

If an intersect exists for the two offset items, a point is created at the intersection with name, colour, model and z-value from the **cogo edit** panel.

If no z-value exists in the panel, no point is created.

As in the **intersect** option, when more than one solutions is possible, the order of picking and the directions imparted when picking the items are used to determine which solution is required.

Since the option will only intersect arcs and lines, when an item is selected only the picked line or arc sub-element of the item is used in the intersect.

Screen message area

<<Intersect Offset> Select 1st item to intersect> [picks][][menu]
<Intersect Offset> 1st offset [caret][][menu] select a button
<<Intersect Offset> Select 2st item to intersect> [picks][][menu]
<Intersect Offset> 2nd offset [caret][][menu] select a button

14.7.2.3 Bearings Intersect

The **Bearings intersect** option finds the intersect of the lines going through two user selected points at user supplied bearings.

Hence the option will calculate the point that is at given bearings from two selected points

After selecting the **bearings intersect** option, the user selects the first point using the normal picking sequence.

A bearing 1 box is then displayed on the screen and the user enters the bearing for the line through the point into the box, terminated with a <enter>. The box then disappears. The bearing is entered in degrees, minutes and seconds.

The second point is then selected and a bearing 2 box is displayed on the screen for the user to enter the bearing for the line through the second point.

The intersection of the two lines is then calculated and if it exists, a point is created at the intersection with name, colour, model and z-value from the **cogo edit** panel.

If no z-value exists in the panel, no point is created.

Note

The selected points do not have to be actual points in a point string. Any snap or cursor position is a valid **point** for the option

Screen message area

<<Bearings Intersect> Select 1st point> [picks][][menu]
<Bearings Intersect> bearing 1 [caret][][menu] select a button
<<Bearings Intersect> Select 2nd point> [picks][][menu]
<Bearings Intersect> bearing 2 [caret][][menu] select a button

14.7.2.4 Two Points, Two Distances

The **2 pts 2 dists** option find the point that given distances from two user selected points.

This problem is equivalent to finding the intersection of two circles centred on the points and with radii the same as the distances.

When there is more than one solution, the solution to the right of the line joining the two points is used. (the direction of the line is from the first point to the second point). To obtain the other solution, simply select the points in the opposite order.

After selecting the **2 pts 2 dists** option, the user selects the first point using the normal picking sequence.

A 1st distance box is then displayed on the screen and the user enters the distance that the new point is from the selected point into the box, terminated with a <enter>. The box then disappears.

The second point is then selected and a 2nd distance box is displayed on the screen for the user to enter the distance to the second point.

The point that is the required distances from the two points is created with name, colour, model and z-value from the **cogo edit** panel. If no z-value exists in the panel, no point is created.

Note

The selected points do not have to be actual points in a point string. Any snap or cursor position is a valid **point** for the option

Screen message area

<<Locate 2 pts 2 dists> Select 1st point> [picks][][menu]
<Locate 2 pts 2 dists> 1st distance [caret][][menu] select a button
<<Locate 2 pts 2 dists> Select 2nd point> [picks][][menu]
<Locate 2 pts 2 dists> 2nd distance [caret][][menu] select a button

Please continue to the next section 14.7.3 Vector.

14.7.3 Vector

The **vector** options are for creating points which are a given as at a given vector from another point.

The default definition of the vector is as (bearing, distance) but other combinations such as (angle, distance), relative (x,y) and absolute (x,y) are possible using normal typed input.

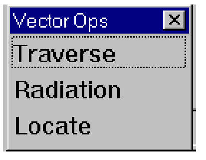
The methods for creating the new points are

(a) locate - create a new point using a vector

(b) radiation - the points created are all a vector from the initial point selected

(c) traverse - the last point created becomes the next point to work from

The **vector** walk-right menu is



Each of the options in the **Vector Ops** menu will now be discussed.

For the option *Traverse*, go to 14.7.3.3 Traverse
Radiation 14.7.3.2 Radiation
Intersect offset 14.7.2.2 Intersect Offset
Locate 14.7.3.1 Locate

14.7.3.1 Locate

In the **locate** option, a point is selected and a vector supplied to produce a new point which is the vector away from the initial point.

The sequence is then repeated by selecting another point and giving another vector.

After selecting the **locate** option, the user simply selects the point to be used using the normal picking sequence.

A bearing distance box is then displayed on the screen and the user enters the required bearing and distance (separated by a space) into the box, terminated with a <enter>. The box then disappears.

A new point is then created which is the given bearing and distance from the selected point. The new point is given the name, colour, model and z-value from the **cogo edit** panel. If no z-value exists in the panel, the given z-value could be nonsense.

The sequence of picking a point and giving a vector is then repeated.

The option terminates when another cogo option is selected or the **cancel** button is selected on the **cogo edit** panel.

Screen message area

<<Locate> Select point> [picks][][menu]
<Locate> bearing distance [caret][][menu] select a button

14.7.3.2 Radiation

In the **radiation** option, a point is selected and a vector supplied to produce a new point which is the vector away from the initial point.

After selecting the **radiation** option, the user simply selects the point to be used using the normal picking sequence.

A bearing distance box is then displayed on the screen and the user enters the required bearing and distance (separated by a space) into the box, terminated with a <enter>.

A new point is then created which is the given bearing and distance from the selected point. The new point is given the name, colour, model and z-value from the **cogo edit** panel. If no z-value exists in the panel, the given z-value could be nonsense.

The bearing distance box remains on the screen.

If another value is entered into the box (or the previous one left) terminated by a <enter>, a new point is created which is the new given bearing and distance from the original selected point.

The option terminates when another cogo option is selected or the **cancel** button is selected on the **cogo edit** panel.

Screen message area

<<Radiation> Select point> [picks][][menu]
<Radiation> bearing distance [caret][][menu] select a button

14.7.3.3 Traverse

In the **traverse** option, an initial point is chosen and a vector supplied to produce a new point which is the vector away from the initial point.

The created point is then taken to be the start point for the next sequence of creating a new point by a vector.

After selecting the **traverse** option, the user simply selects the first point to be used using the normal picking sequence.

A bearing distance box is then displayed on the screen and the user enters the required bearing and distance (separated by a space) into the box, terminated with a <enter>.

A new point is then created which is the given bearing and distance from the selected point.

The new point is given the name, colour, model and z-value from the **cogo edit** panel. If no z-value exists in the panel, the given z-value could be nonsense.

The bearing distance box remains on the screen.

If another value is entered into the box (or the previous one left) terminated by a <enter>, a new point is created which is the given bearing and distance from the last created point.

Hence, the last created point takes the place of the original selected point.

The option terminates when another cogo option is selected or the **cancel** button is selected on the **cogo edit** panel.

Screen message area

<<Traverse> Select point> [picks][][menu]
<Traverse> bearing distance [caret][][menu] select a button

Please continue to the next section 14.7.4 Parallel.

14.7.4 Parallel

Items can be translated perpendicularly to either the left or right using the parallel option (also known as a copy parallel operation).

After selecting the **parallel** option, the user simply selects the item to paralleled using the normal picking sequence.

A offset box is then displayed on the screen and the user enters the required offset distance into the box, terminated with a <enter>. The box then disappears. A positive offset is to the right of the item when moving in the picking direction of the item.

A new item is then created which is a copy parallel of the selected item by the given offset distance. The new item has the name, colour and model from the **cogo edit** panel.

The option terminates when another cogo option is selected or the **cancel** button is selected on the **cogo edit** panel.

Screen message area

<<Parallel> Select item to parallel> [picks][][menu]
<Parallel> offset> [caret][][menu] select a button

Please continue to the next section 14.7.5 Fillet Ops.

14.7.5 Fillet Ops

The **fillet** options are for creating an arc between lines and arcs.

That is, it will try and fit an arc between

(a) two lines

(b) a line and an arc

or

(c) two arcs

The created arc is determined by tangents from the items it is a fillet for and another piece of information such as a radius or a start point.

Since the option can only fillet between arcs and lines, when an item is selected only the picked line or arc sub-element of the item is used in the fillet.

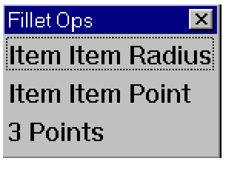
Most fillet cases can have more that one solution. For example, the fillet of a given radius between two lines can have up to four solutions.

To easily distinguish between the cases, 12d Model uses the order of picking and the sense of direction implied when picking the items to determine which case was required by the user.

The direction is used as follow -

> the arc comes off the first item in the direction that the first item is picked and lands on the second in the direction that the second item is picked.

The **fillet ops** walk-right menu is



Each of the options in the **Fillet Ops** menu will now be discussed.

For the option *Item Item Radius*, go to 14.7.5.1 Fillet Item Item Radius
Item Item Point 14.7.5.2 Fillet Item Item Point
3 points 14.7.5.3 Fillet Three Points

14.7.5.1 Fillet Item Item Radius

This fillet option creates an arc of a user given radius between two user selected items.

After selecting the **fillet item item radius** option, the user selects the first item to be used in the fillet using the normal picking sequence.

Next the second item to be used in the fillet is selected.

An enter radius box is then displayed on the screen and the user types the appropriate radius for the fillet into the box, terminated with a <enter>. The box then disappears.

A positive radius implies that the arc travels in a clockwise direction and a negative radius for an arc travelling in an anti-clockwise direction.

If a fillet exists for the two items and the radius, an fillet arc is created with name, colour, model and z-value from the **cogo edit** panel.

If no z-value exists in the panel, no arc is created.

Screen message area

<<Fillet by radius> Select 1st item> [picks][][menu]

<<Fillet by radius> Select 2nd item> [picks][][menu]
<Fillet by radius> Enter radius [caret][][menu] select a button

14.7.5.2 Fillet Item Item Point

This fillet option creates an arc between two user selected items starting at a user defined control point.

After selecting the **fillet item item pt** option, the user selects the first item to be used in the fillet using the normal picking sequence.

Next the second item to be used in the fillet is selected.

Finally, a point is selected to act as the starting point for the fillet. This point, called the control point, must lie on the first item selected. If the selected point doesn't lie on the first item, the selected point is projected perpendicularly onto the item and the projected point used as the control point.

If a fillet exists for the two items and the control point, a fillet arc is created with name, colour, model and z-value from the **cogo edit** panel.

If no z-value exists in the panel, a null z-value is used.

Screen message area

<<Fillet by start point> Select 1st item> [picks][][menu]
<<Fillet by start point> Select 2nd item> [picks][][menu]
<<Fillet by start point> Select start point> [picks][][menu]

14.7.5.3 Fillet Three Points

This option creates an arc between three user selected points.

After selecting the **fillet 3 pts** option, the user selects the three points one after another using the normal picking sequence.

The three arc points must be selected in the order

(a) the start point
(b) a point between the start and end point
(c) the end point

If an arc does exist containing the three points, it will be unique including its direction. The arc will be created with name, colour, model and z-value from the **cogo edit** panel.

If no z-value exists in the panel, a null z-value is used.

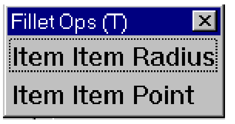
Screen message area

<<Fillet 3 points> Select 1st item> [picks][][menu]
<<Fillet 3 points> Select 2nd item> [picks][][menu]
<<Fillet 3 points> Select start point> [picks][][menu]

Please continue to the next section 14.7.6 Fillet Ops (T).

14.7.6 Fillet Ops (T)

The **fillet (T)** options are the almost identical to the fillet options.



The only difference is that after the fillet arc is created, the original items are trimmed back to the start and end of the arc.

The trimmed items are not joined to the fillet arc.

The original strings obey the trash defaults from the **utilities=>defaults=>trash defaults** option.

That is, if the trash mode is set to *keep string*, the original strings are untouched.
delete string, deleted.
trash string, moved to the trash model.

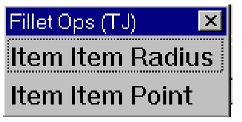
For information on the fillet options, see the section 14.7.5 Fillet Ops

For the option *Item Item Radius*, please go to the section 14.7.5.1 Fillet Item Item Radius.
Item Item Point 14.7.5.2 Fillet Item Item Point.

Please continue to the next section 14.7.7 Fillet Ops (TJ).

14.7.7 Fillet Ops (TJ)

The **fillet (TJ)** options are the almost identical to the fillet and fillet (T) options.



The only difference is that after the fillet arc is created, the original items are trimmed back to the start and end of the arc and the three items joined to form a new item.

That is, the trimmed items are joined to the fillet arc.

The original strings obey the trash defaults from the **Utilities=>Defaults=>Trash defaults** option.

That is, if the trash mode is set to *keep string*, the original strings are untouched.
delete string, deleted.
trash string, moved to the trash model.

For information on the fillet options, see the section 14.7.5 Fillet Ops

For the option *Item Item Radius*, please continue to the section 14.7.5.1 Fillet Item Item Radius.
Item Item Point 14.7.5.2 Fillet Item Item Point.

Please continue to the next section 14.7.8 Tangent.

14.7.8 Tangent

The **tangent** option creates tangent lines between point and arcs.

That is, tangents can be defined between

(a) a point and an arc

(b) two arcs

or

(c) two points - simply a line between the two points

Since the option can only fillet between points and arcs, when an item is selected, only the picked sub-arc of the item is used as an arc.

Most tangent cases can have more that one solution. For example, two tangents from a point to an arc.

To easily distinguish between the cases, 12d Model uses the order of picking the items and the sense of direction implied when picking an arc to determine which case was required by the user.

After selecting the **tangent** option, the user selects the first item to be used in the tangent calculations using the normal picking sequence.

Next the second item to be used is selected.

If a tangent line exists between the two items selected, a line string is created with name, colour, model and z-value from the **cogo edit** panel.

If no z-value exists in the panel, a null z-value is used for the line.

Screen message area

<<Tangent> Select 1st item> [picks][][menu]
<<Tangent> Select 2nd item> [picks][][menu]

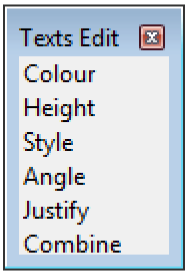
14.8 Texts Edit

Position of menu: **Strings =>Texts Edit**

These options are like the **Strings =>Points edit** options in that the options only does the one thing to a selected string. You then pick another string to do the same type of change.

For each of the Texts Edit options, press <Esc> or select **Cancel** from the **Pick Ops** menu to terminate the option. Selecting another CAD option will also terminate the option and start the new option.

Walking right on **Texts edit** displays the **Texts edit** menu

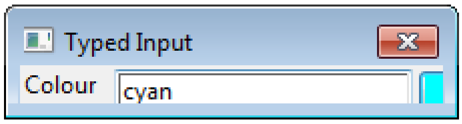


set/edit colour of text
set/edit height of text
set/edit style of text
set/edit angle of text
justify text
combine text

For the option *Colour*, go to 14.8.1 Change Text Colour
Height 14.8.2 Change Text Height
Style 14.8.3 Change Text Style
Angle 14.8.4 Change Text Angle
Justify 14.8.5 Change Text Justify
Combine 14.8.6 Change Combine

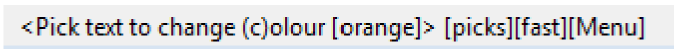
14.8.1 Change Text Colour

Click on **Colour** and a *Colour* **Typed Input** box comes up



The new colour is picked from the pop up list, or typed into the box and <Enter> pressed.

The box is then removed and the following message is written to the screen message area



Pick text and its colour will be changed to that shown in the square brackets ([]). Then pick another text to change its colour.

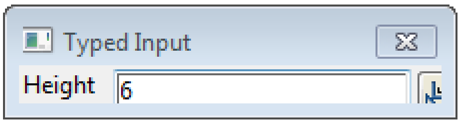
Typing c or C brings up the *Colour* **Typed Input** box to select a new colour.

All subsequent selected texts will have their colour changed to this colour until either the option is terminated or a new colour is selected

To terminate the option, press <Esc>, select **Cancel** from the **Pick Ops** menu, or select another CAD option.

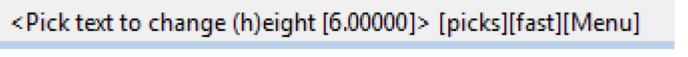
14.8.2 Change Text Height

Click on **Height** and a *Height* **Typed Input** box comes up



The new height is typed into the box and <Enter> pressed.

The box is then removed and the following message is written to the screen message area



Pick text and its height will be changed to that shown in the square brackets ([]). Then pick another text to change its height.

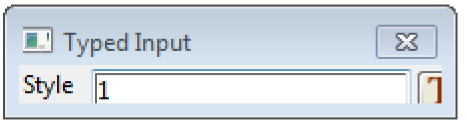
Typing h or H bring up the *Height* **Typed Input** box to select a new height.

All subsequent selected texts will have their height changed to this height until either the option is terminated or a new height is selected

To terminate the option, press <Esc>, select **Cancel** from the **Pick Ops** menu, or select another CAD option.

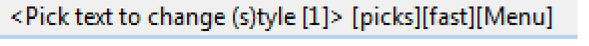
14.8.3 Change Text Style

Click on **Style** and a *Style* **Typed Input** box comes up



The new style is picked from the pop up list, or typed into the box and <Enter> pressed.

The box is then removed and the following message is written to the screen message area



Pick text and its style will be changed to that shown in the square brackets ([]). Then pick another text to change its style.

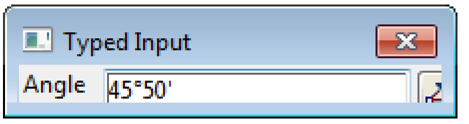
Typing s or S brings up the *Style* **Typed Input** box to select a new style.

All subsequent selected texts will have their style changed to this style until either the option is terminated or a new style is selected

To terminate the option, press <Esc>, select **Cancel** from the **Pick Ops** menu, or select another CAD option.

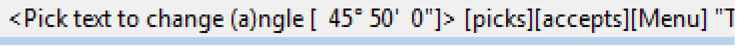
14.8.4 Change Text Angle

Click on **Angle** and an *Angle* **Typed Input** box comes up



The new angle is typed into the box and <Enter> pressed.

The box is then removed and the following message is written to the screen message area



Pick text and its angle will be changed to that shown in the square brackets ([]). Then pick another text to change its angle.

Typing a or A brings up the *Angle* **Typed Input** box to select a new angle.

All subsequent selected texts will have their angle changed to this angle until either the option is terminated or a new angle is selected

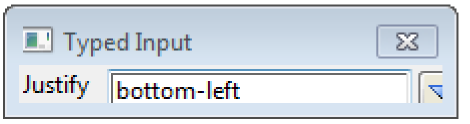
To terminate the option, press <Esc>, select **Cancel** from the **Pick Ops** menu, or select another CAD option.

Note

The angle is type in in hp notation and measured counter clockwise from the positive x-axis. The box will convert the hp notation and display it in degrees, minutes and seconds. For example, 45.5 in hp notation become 45 degrees 50 minutes.

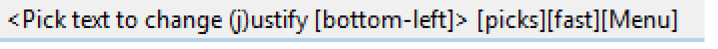
14.8.5 Change Text Justify

Click on **Justify** and a *Justify* **Typed Input** box comes up



The new justification is picked from the pop up list, or typed into the box and <Enter> pressed.

The box is then removed and the following message is written to the screen message area



Pick text and its justification will be changed to that shown in the square brackets ([]). Then pick another text to change its justification.

Typing j or J brings up the *Justify* **Typed Input** box to select a new justification.

All subsequent selected texts will have their justification changed to this justification until either the option is terminated or a new justification is selected

To terminate the option, press <Esc>, select **Cancel** from the **Pick Ops** menu, or select another CAD option.

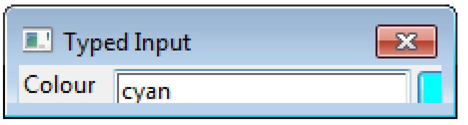
14.8.6 Change Combine

This option allows you to change one or more of Colour, Height, Angle, Textstyle and Justification.

Selecting **Combine** writes the following message to the screen message area

<Pick text to change (c)olour (h)eight (a)ngle (s)tyle (j)ustify> [picks][fast][Menu]

Type **c** or **C** and a *Colour* **Typed Input** box comes up



The new colour is picked from the pop up list, or typed into the box and <Enter> pressed.

The box is then removed and the following message is written to the screen message area

<Pick text to change (c)olour[cyan] (h)eight (a)ngle (s)tyle (j)ustify> [picks][fast][Men

This sets the colour to cyan.

Pick text and its colour will be changed to that shown in the square brackets ([]). Then pick another text to change its colour.

Typing c or C when there is a colour shown between the square brackets (for example [cyan]), removes the colour from the square brackets.

If any text is picked when there is no colour in the square brackets then its colour will not be changed.

Similarly:

Height

IF there is no height in square brackets after *(h)eight,* typing **h** or **H** brings up the *Height* **Typed Input** box to select a new height. Entering a height and pressing <Enter> **sets** the new height and it will appear in square brackets after *(h)eight* in the message written to the screen message area.

Typing **h** or **H** when there is a height shown between the square brackets (for example [2.00]), **removes** the height from the square brackets.

Angle

IF there is no angle in square brackets after *(a)ngle,* typing **a** or **A** brings up the *Angle* **Typed Input** box to select a new angle. Entering an angle and pressing <Enter> **sets** the new angle and it will appear in square brackets after *(a)ngle* in the message written to the screen message area.

Typing **a** or **A** when there is an angle shown between the square brackets, **removes** the angle from the square brackets.

Textstyle

IF there is no textstyle in square brackets after *(s)tyle,* typing **s** or **S** brings up the *Style* **Typed Input** box to select a new textstyle. Entering an textstyle and pressing <Enter> **sets** the new textstyle and it will appear in square brackets after *(s)tyle* in the message written to the screen message area.

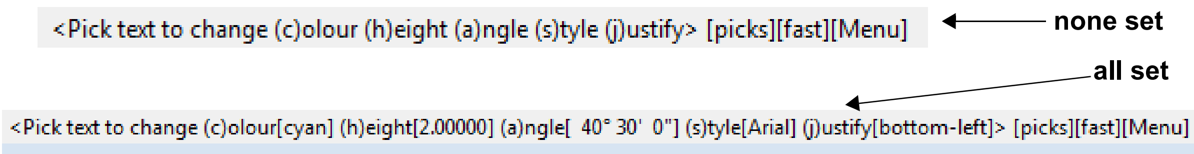
Typing **s** or **S** when there is an textstyle shown between the square brackets, **removes** the textstyle from the square brackets.

Justification

IF there is no justification in square brackets after *(j)ustify,* typing **s** or **S** brings up the *Justify*

Typed Input box to select a new justification. Entering a justification and pressing <Enter> **sets** the new justification and it will appear in square brackets after *(j)ustify* in the message written to the screen message area.

Typing **j** or **J** when there is an justification shown between the square brackets, **removes** the justification from the square brackets.



If any text is picked then the values of the text are changed for any that there is a value in the square brackets otherwise that value of the text is not changed.

To terminate the option, press <Esc>, select **Cancel** from the **Pick Ops** menu, or select another CAD option.

14.9 Convert

Position of option on menu: **Strings =>Convert**

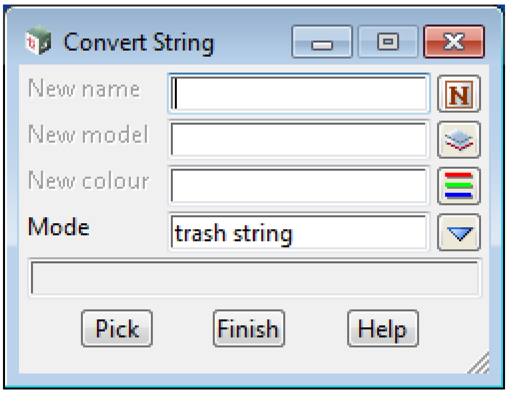
The **Convert** option is used to create a new string of possibly a different string type from a chosen string.

Since not all string conversion are possible, or even make sense (for example, converting a 3d string to a text string), the list of defined conversion depends on the type of the string chosen for conversion.

After selecting the string to convert, a menu containing the possible conversions is raised and the required conversion selected from it.

After the type of conversion is selected, a string Properties panel containing the attribute information for the new string is displayed and can be used to modify any properties of the new string.

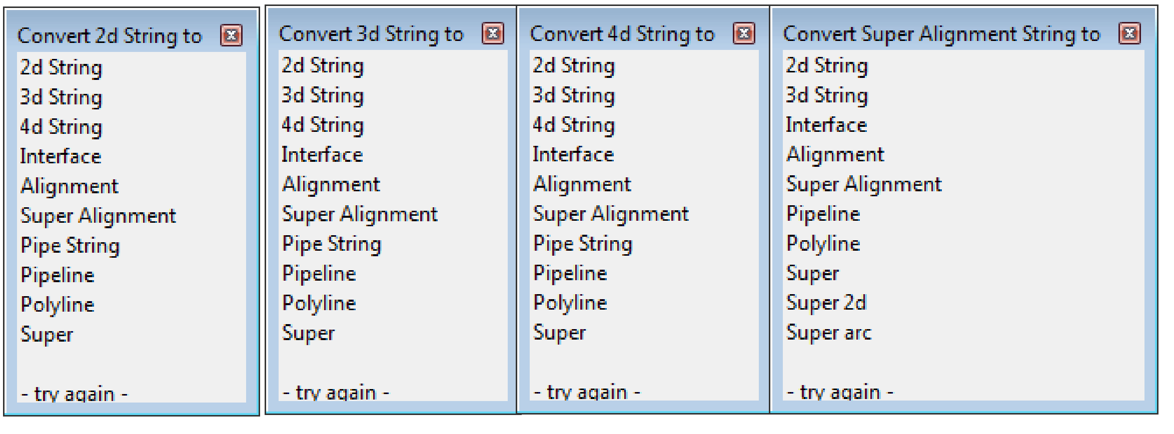
After selecting the **Convert option**, the **Convert String** panel is placed on the screen.

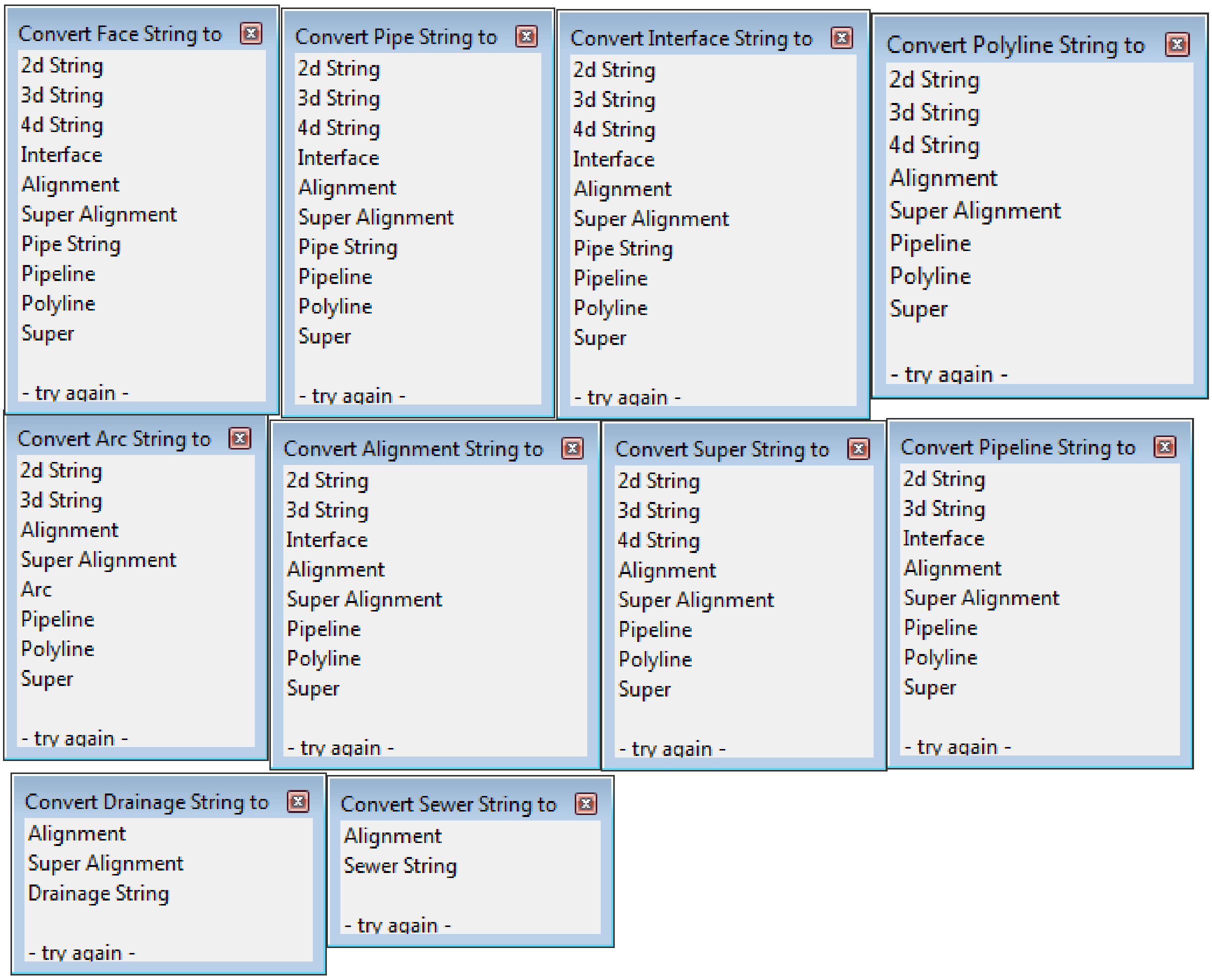


If a new name, model or colour is required for the converted string, the new data is entered into the **Convert String** panel. If a change isn't required, leave the appropriate panel field blank.

The **Convert** option is already in progress and if a string is selected and accepted, a menu containing all the available conversions for that particular string type is displayed.

For example, for each string type the choices are:





After selecting the conversion to be made, a new string of the selected type is created and given the model, colour and name according to the fields in the **Convert String** panel.

The Properties panel for the new string is also displayed and hence any of the strings properties can be modified for the new string.

The fields and buttons used in the **Convert String** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
New name	input		

If blank, the converted string is given the same name as the original string.
If non-blank, then the converted string is given the name in the **new name** *field.*

New model	input		available models

if blank, the converted string is placed in the same model as the original string.
If non-blank, then the converted string will be placed in the model given in the new model *field.*

New colour	input		available colours

if blank, the converted string is given the same colour as the original string.
If non-blank, then the converted string is given the colour in the new colour *field.*

Mode	input	keep string	keep, delete, trash string

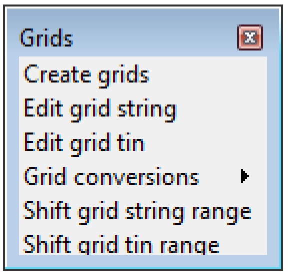
if ***delete string****, the string selected to be converted is deleted.*
keep string*, the selected string is not deleted.*
trash string*, the string selected to be converted is moved to the trash model.*

14.10 Grids

Position of option on menu: **Strings =>Grids**

The Grids menu allows the user to create and edit grids.

After selecting **Grids**, the **Grids** menu is placed on the screen.



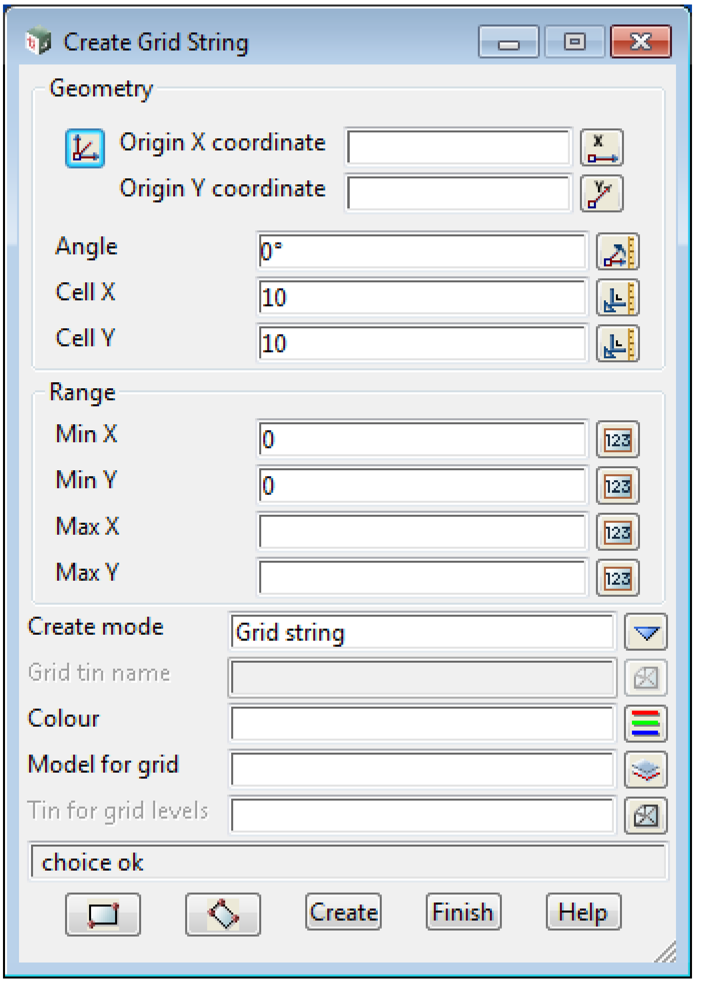
For the option:

Create grids	14.10.1 Create Grids
Edit grid string	14.10.2 Edit Grid String
Edit grid tin	14.10.3 Edit Grid Tin
Grid conversions	14.10.4 Grid Conversions
Shift grid string range	14.10.5 Shift Grid String Range
Shift grid tin range	14.10.6 Shift Grid Tin Range

14.10.1 Create Grids

Position of option on menu: **Strings =>Grids =>Create grids**

Selecting **Create grids** brings up the **Create Grid String** panel



The fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

Geometry

Origin X/Y coordinate coordinate box

coordinates signifying the position where the range 0,0 exists. The Range group below specifies how many cells exist either side of this origin.

Angle angle box 0

the angle of the grid's local X axis

Cell X/Y measurement box 10

the local X axis and Y axis spacings

Range

Min X number box 0

the number of cells relative to the origin in the grid's local X axis. The sign of the number denotes whether it is to the left or right of the origin.

Min Y number box 0

the number of cells relative to the origin in the grid's local Y axis. The sign of the number denotes whether it is below or above the origin.

Max X number box

the number of cells relative to the origin in the grid's local X axis. The sign of the number denotes whether it is to the left or right of the origin.

Max Y number box

the number of cells relative to the origin in the grid's local Y axis. The sign of the number denotes whether it is below or above the origin.

Create mode choice box Grid string Grid string, Grid tin

the type of grid to be created.

Grid tin name tin box

enabled only if Create mode *is set to* Grid tin. *The name of the grid tin to be created. The tin must not already exist.*

Colour colour box

string colour for all points on the grid

Model for grid model box

name of the model in which the data is to be placed. The model will be created if it does not already exist. This field must be filled in.

Tin for grid levels tin box

optional information. If tin is selected, the height on each grid point is obtained from the grid. If blank, the height of each grid point is null.

Rectangle button

if selected, the origin X/Y, angle and all range values are calculated from two corners of the selected rectangle. ***Note****: Cell X/Y must be defined.*

Rectangle by 3 Points button

if selected, the origin X/Y, angle and all range values are calculated from three corners of the selected rectangle. The first point defines the origin, the second point defines the angle and range X values, and the third point defines the range Y values. ***Note****: Cell X/Y must be defined.*

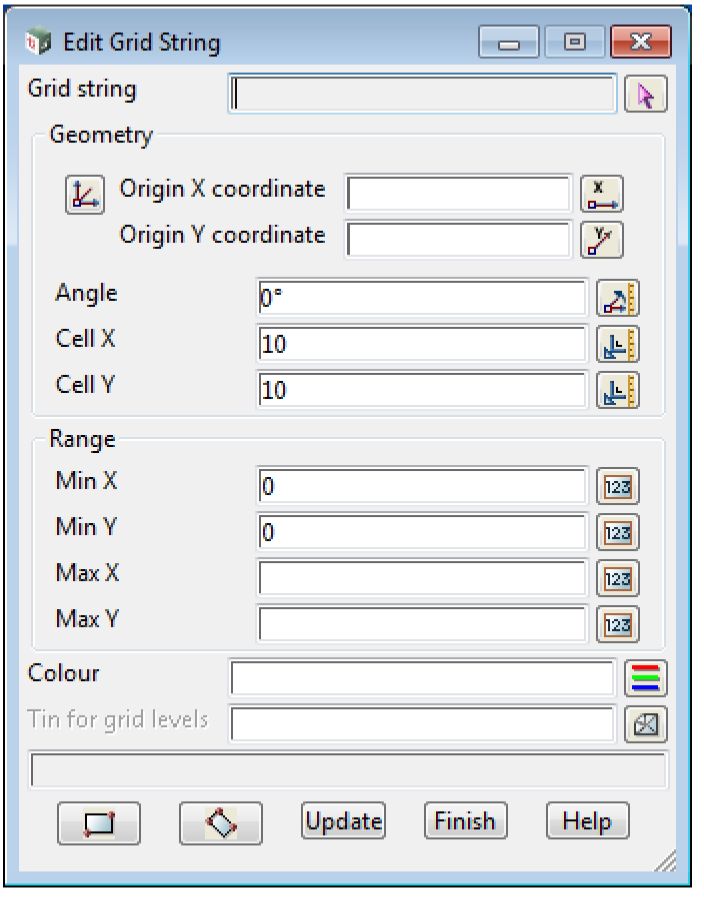
Create button

creates the grid string/tin.

14.10.2 Edit Grid String

Position of option on menu: **Strings =>Grids =>Edit grid string**

Selecting **Edit grid string** brings up the **Edit Grid String** panel



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Grid string	string select		

the grid string to be edited.

Geometry

Origin X/Y coordinate	coordinate box		

coordinates signifying the position where the range 0,0 exists. The Range group below specifies how many cells exist either side of this origin.

Angle	angle box	0	

the angle of the grid's local X axis

Cell X/Y	measurement box	10	

the local X axis and Y axis spacings

Range

Min X	number box	0	

the number of cells relative to the origin in the grid's local X axis. The sign of the number denotes whether it is to the left or right of the origin.

Min Y	number box	0	

the number of cells relative to the origin in the grid's local Y axis. The sign of the number denotes whether it is below or above the origin.

Max X number box

the number of cells relative to the origin in the grid's local X axis. The sign of the number denotes whether it is to the left or right of the origin.

Max Y number box

the number of cells relative to the origin in the grid's local Y axis. The sign of the number denotes whether it is below or above the origin.

Colour colour box

string colour for all points on the grid

Tin for grid levels tin box

optional information. If tin is selected, the height on each grid point is obtained from the grid. If blank, the height of each grid point is null.

Rectangle button

if selected, the origin X/Y, angle and all range values are calculated from the two corners of the selected rectangle. Note: Cell X/Y must be defined.

Rectangle by 3 Points button

if selected, the origin X/Y, angle and all range values are calculated from three corners of the selected rectangle. The first point defines the origin, the second point defines the angle and range X values, and the third point defines the range Y values. Note: Cell X/Y must be defined.

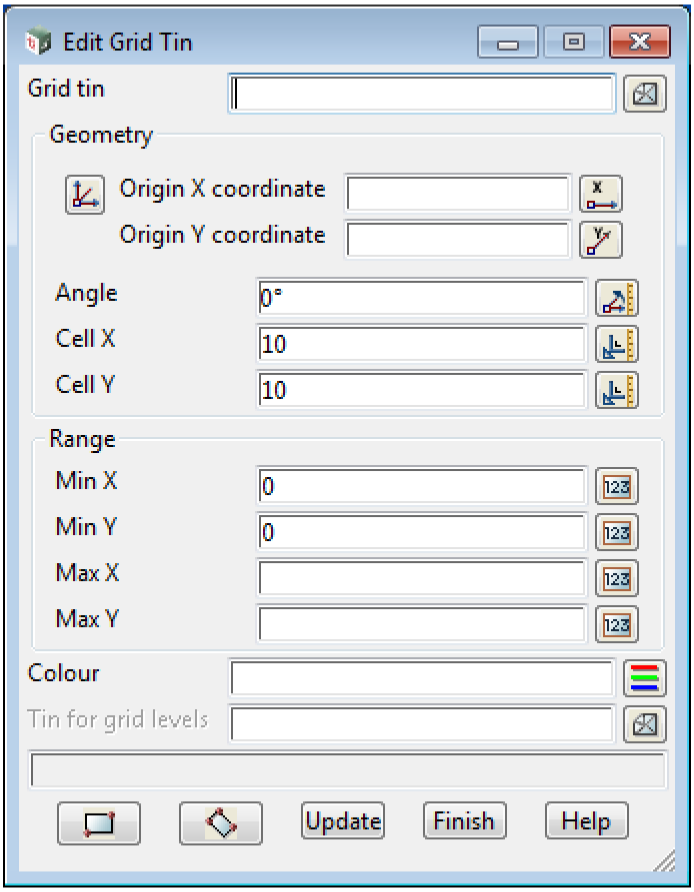
Update button

updates the grid string.

14.10.3 Edit Grid Tin

Position of option on menu: **Strings =>Grids =>Edit grid tin**

Selecting **Edit grid tin** brings up the **Edit Grid Tin** panel



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Grid string	string select		

the grid tin to be edited.

Geometry

Origin X/Y coordinate coordinate box

coordinates signifying the position where the range 0,0 exists. The Range group below specifies how many cells exist either side of this origin.

Angle angle box 0

the angle of the grid's local X axis

Cell X/Y measurement box 10

the local X axis and Y axis spacings

Range

Min X number box 0

the number of cells relative to the origin in the grid's local X axis. The sign of the number denotes whether it is to the left or right of the origin.

Min Y number box 0

the number of cells relative to the origin in the grid's local Y axis. The sign of the number denotes whether it is below or above the origin.

Max X number box

the number of cells relative to the origin in the grid's local X axis. The sign of the number denotes whether it is to the left or right of the origin.

Max Y number box

the number of cells relative to the origin in the grid's local Y axis. The sign of the number denotes whether it is below or above the origin.

Colour colour box

string colour for all points on the grid

Tin for grid levels tin box

optional information. If tin is selected, the height on each grid point is obtained from the grid. If blank, the height of each grid point is null.

Rectangle button

if selected, the origin X/Y, angle and all range values are calculated from the two corners of the selected rectangle. Note: Cell X/Y must be defined.

Rectangle by 3 Points button

if selected, the origin X/Y, angle and all range values are calculated from three corners of the selected rectangle. The first point defines the origin, the second point defines the angle and range X values, and the third point defines the range Y values. Note: Cell X/Y must be defined.

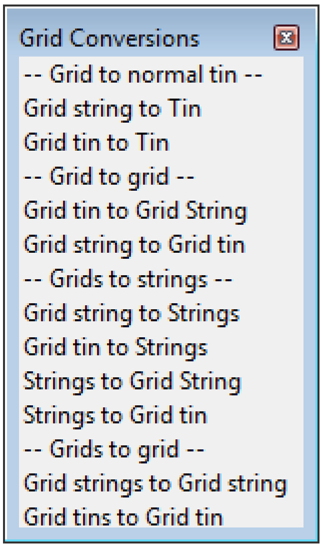
Update button

updates the grid tin.

14.10.4 Grid Conversions

Position of option on menu: **Strings =>Grids =>Grid conversions**

Selecting **Grid conversions** brings up the **Grid conversions** menu



The fields and buttons used in the panel have the following functions.

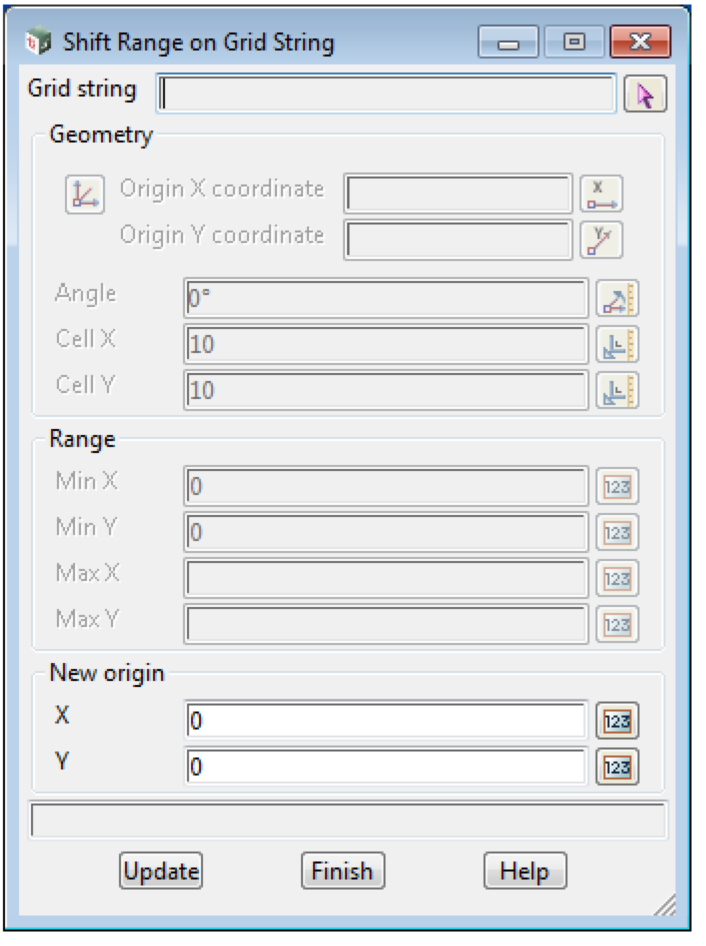
Field Description Type Defaults Pop-Up

Origin X/Y coordinate

14.10.5 Shift Grid String Range

Position of option on menu: **Strings =>Grids =>Shift grid string range**

Selecting **Shift grid string range** brings up the **Shift Range on Grid String** panel



The fields and buttons used in the panel have the following functions.

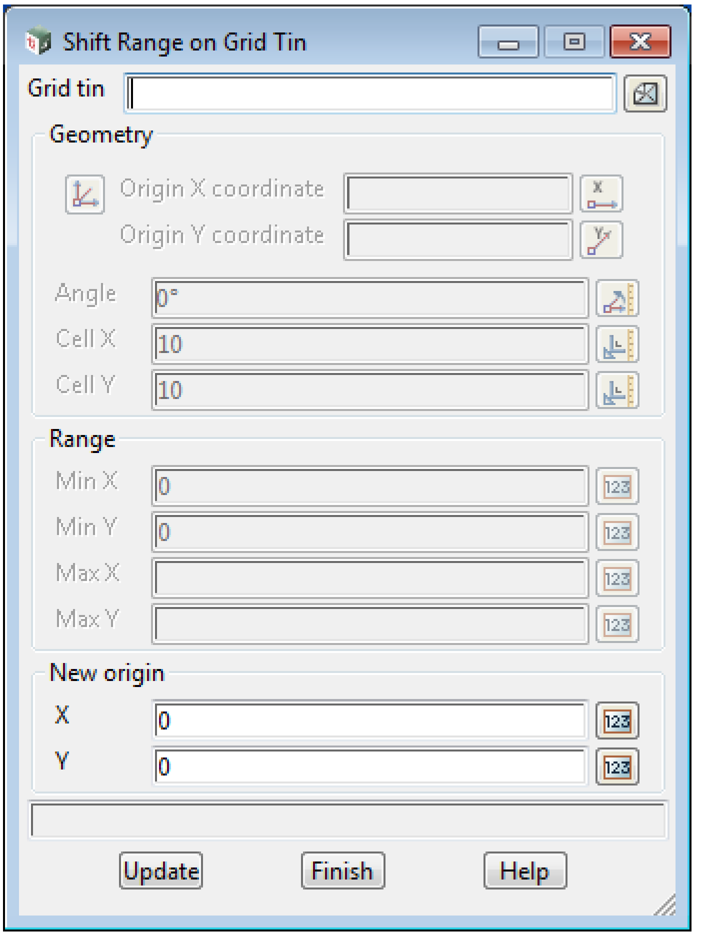
Field Description	Type	Defaults	Pop-Up
Geometry			
Origin X/Y coordinate	coordinate box		
Angle	angle box	0	
Cell X	measurement box	10	
Cell Y	measurement box	10	
Range			
Min X	number box	0	
Min Y	number box	0	
Max X	number box		
Max Y	number box		
New origin			
X	number box		

Y	number box
Update	button

14.10.6 Shift Grid Tin Range

Position of option on menu: **Strings =>Grids =>Shift grid tin range**

Selecting **Shift grid tin range** brings up the **Shift Range on Grid Tin** panel



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Grid tin	tin box		
Geometry			
Origin X/Y coordinate	coordinate box		
Angle	angle box	0	
Cell X	measurement box	10	
Cell Y	measurement box	10	
Range			
Min X	number box	0	
Min Y	number box	0	
Max X	number box		
Max Y	number box		

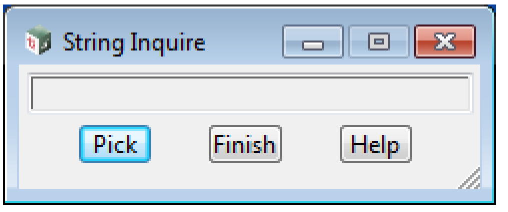
New origin	
X	number box
Y	number box
Update	button

14.11 Inquire

Position of option on menu: **Strings =>Inquire**

The **inquire** option is used to obtain information about a string displayed on the screen.The amount and type of information varies between string types.

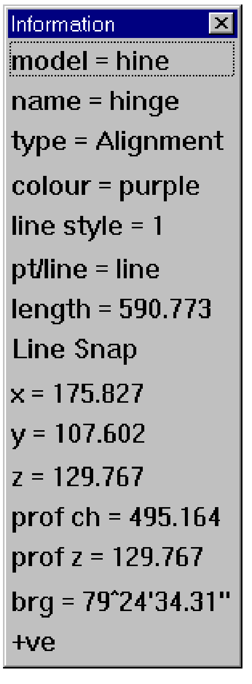
On selecting the **Inquire** option, the **String Inquire** panel is displayed.



The panel is used for any special messages and to end the option.

After selecting the **inquire** option, the strings are selected using the normal 12d Model selecting mechanism (LB to pick a string, MB to accept a string, RB for the **pick ops** menu). Whenever a string is picked, the information about the picked string is displayed on the screen in the **Information** menu.

An example of the **Information** menu for an Alignment string is



For more details on the **Information** menu, go to the section 14.11.1 String Information

Any number of string inquiries can be made with the option by repeated use of the mouse buttons LB and MB to pick and accept strings.

The inquire option is terminated by either selecting the **Cancel** option from the **Pick Ops** menu (raised by clicking RB) or selecting **Finish** or **[X]** from the **String Inquire** panel.

14.11.1 String Information

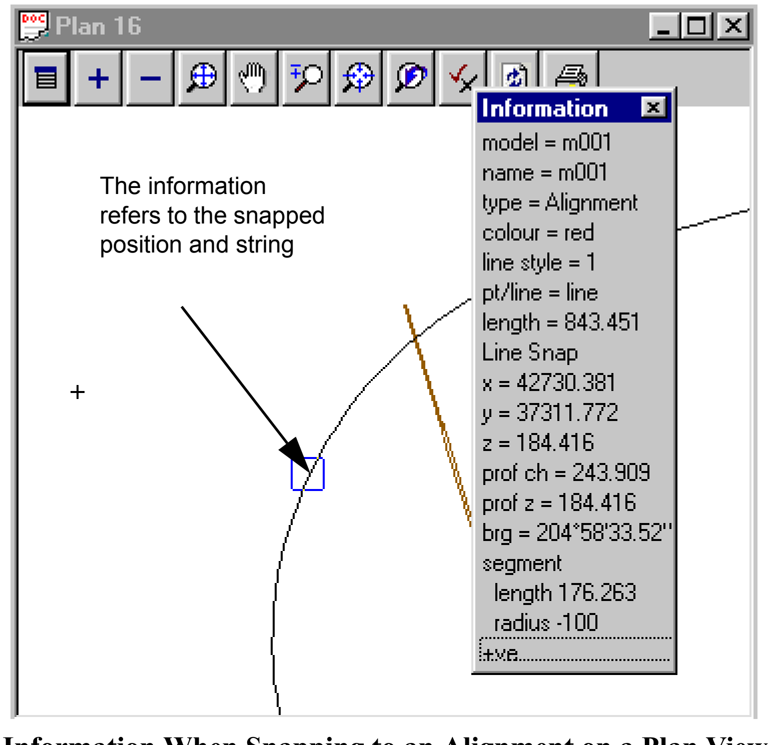
Whenever a string is picked, the information about the picked string is displayed on the screen in the **Information** menu

The **Information** menu displays different information depending on whether the string was picked in a Plan or Perspective view or in a Section view.

For a Plan or Perspective View

The details on the Information menu can include:

model =	- gives the model of the selected string
name =	- " name " " "
type =	- " type " " "
colour =	- " colour " " "
line style =	- " line style " " "
pt/line =	- " " breakline type " " "
# pts =	- " " number of points " " "
area =	- " " area if it is a closed string
length =	- " " length of the string
Snap type	- *e.g.* Line Snap, Point Snap
x =	- " " x co-ordinate of the selected position
y =	- " " y co-ordinate of the selected position
prof ch =	- " " chainage of the selected position on the string
prof z =	- " " z-value of the selected position on the string
brg =	- " instantaneous bearing of the selected position on the string
segment length	- " length of the string segment containing the selected position
+ve or -ve	- if the direction of the pick was the same as the direction of the string, then +ve is displayed otherwise the direction of the pick is opposite to the string direction and a -ve is displayed.



Information When Snapping to an Alignment on a Plan View

For a Section View

For a section view, the x-axis of the view is defined by the chainage along the string that is being profiled on the section view (the ***profiled*** string). The profiled string is displayed on the section view.

For any tins in models added to the section view, the sections through the tins along the profiled string are also displayed in the section view.

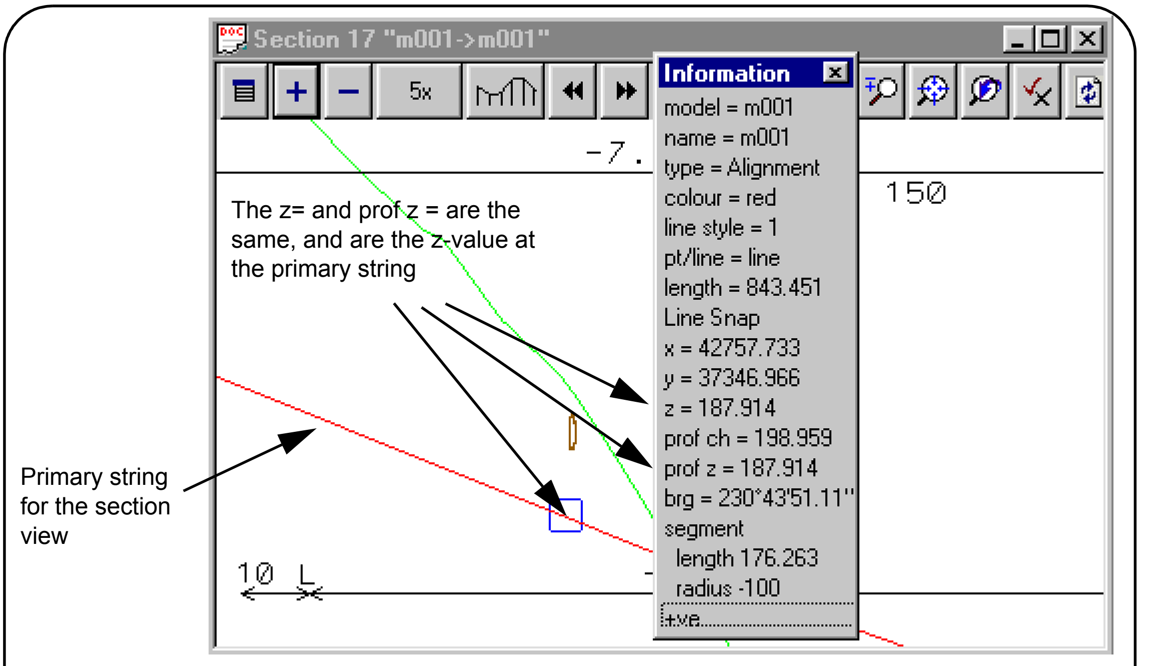
Finally, any parts of any strings in models added to the section view that are in the corridor defined for the section view are projected onto the section view and displayed.

The details on the Information menu can include:

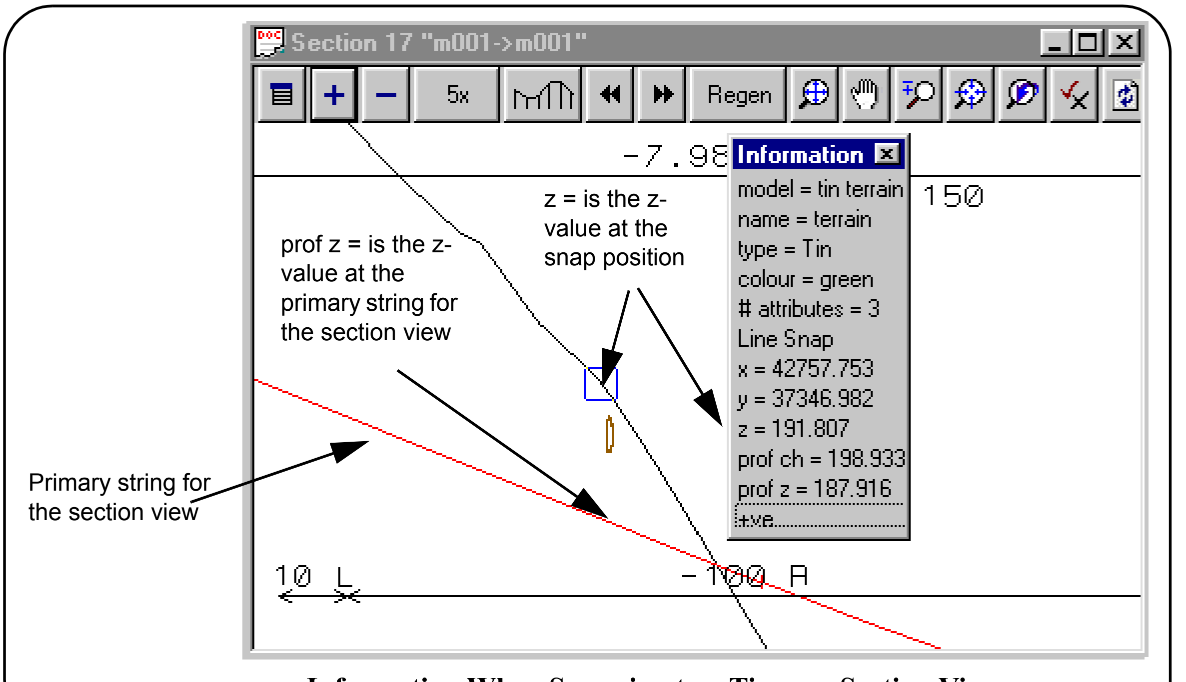
model =	- gives the model of the selected string
name =	- " name " " "
type =	- " type " " "
colour =	- " colour " " "
line style =	- " line style " " "
pt/line =	- " breakline type " " "
# pts =	- " number of points " " "
area =	- " area if it is a closed string
length =	- " length of the string
Snap type	- *e.g.* Line Snap, Point Snap
x =	- " x co-ordinate of the selected position
y =	- " y co-ordinate of the selected position
z =	- " z co-ordinate of the selected position
prof ch =	- " chainage of the *profiled* string at the selected position Note that this is not the chainage of the selected string unless the selected string is the profiled string
prof z =	- " " z-value of the *profiled* string at the profile chainage of the selected position Note that this is not the z of the selected position unless the selected string is the profiled string
brg =	- " instantaneous bearing of the selected position on the string
segment length	- " length of the string segment containing the selected position
+ve or -ve	- if the direction of the pick was the same as the direction of the string, then +ve is displayed otherwise the direction of the pick is opposite to the string direction and a -ve is displayed.

IMPORTANT NOTES

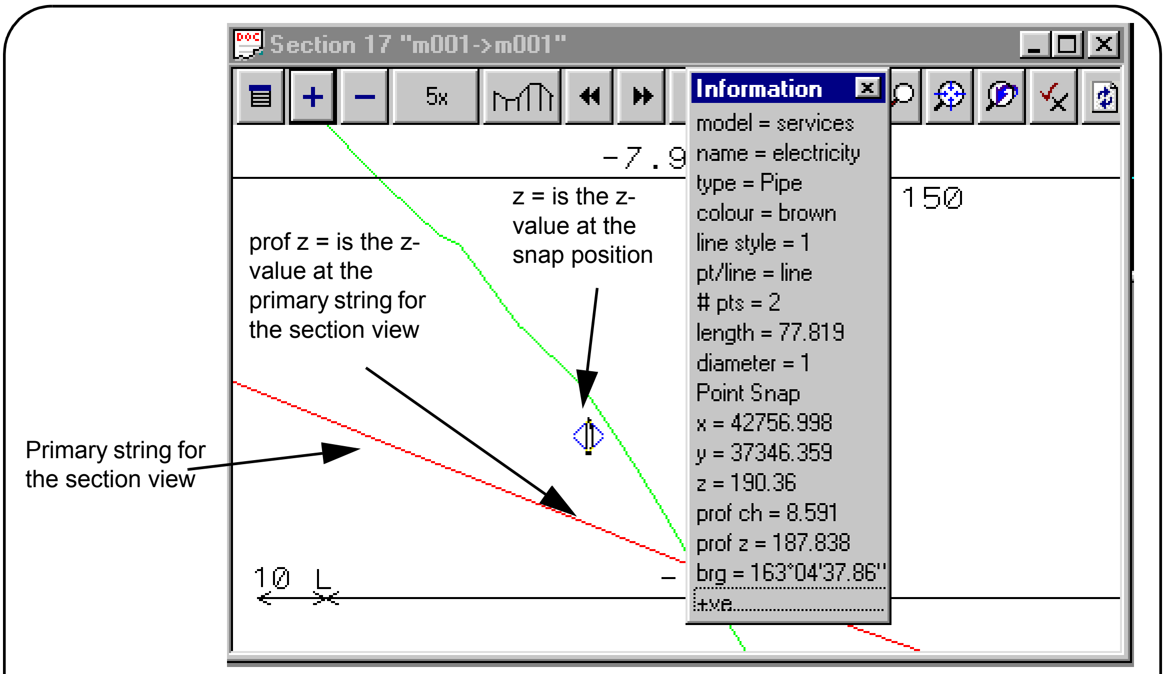
1. for a string selected on a section view, the " ***z =*** " value is the z co-ordinate of the selected position. The "***prof z =*** " value is the z co-ordinate of the string profiled on the section view.
2. The "***prof ch =*** " value is the chainage of the string profiled on the section view, not the chainage of the selected string.



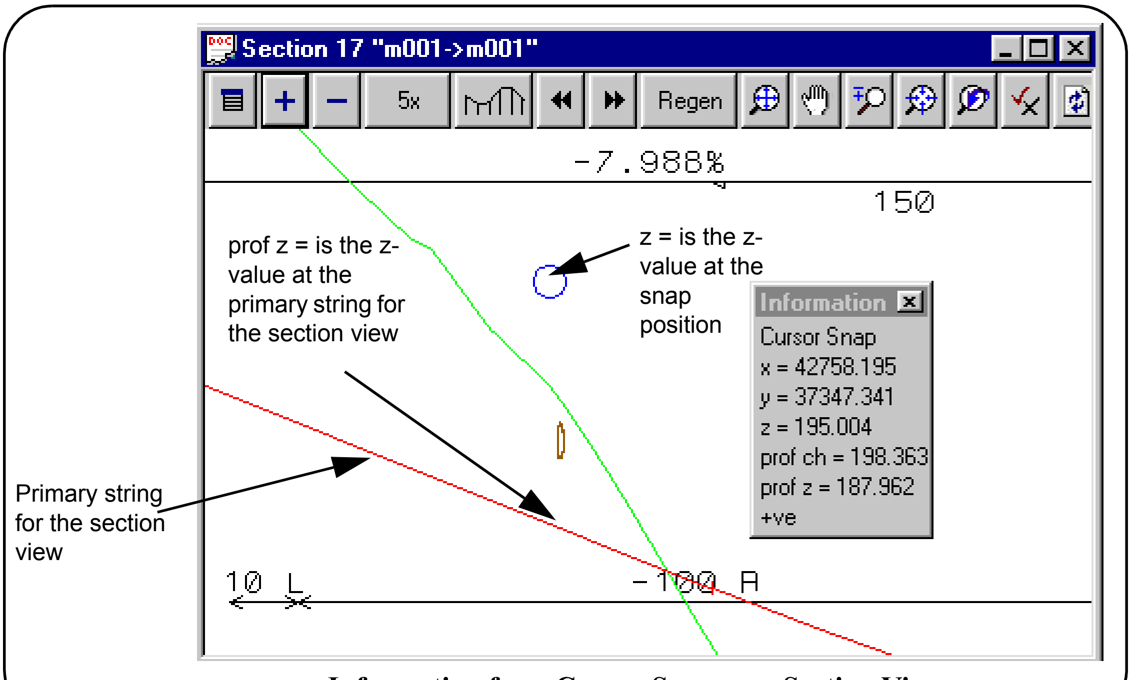
Information When Snapping to the Primary String for a Section View



Information When Snapping to a Tin on a Section View



Information When Snapping to a Service String on a Section View



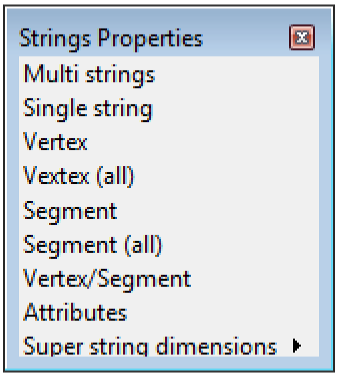
Information for a Cursor Snap on a Section View

14.12 Properties

Position of menu: **Strings =>Properties**

The **Properties** walk-right menu contains options to obtain and/or modify the properties of any string displayed on the screen, or obtain and/or modify the vertices and segments of a super string.

The **Properties** walk-right menu is



For the option:

Multi strings, go to	14.12.1 Multi String Properties
Single string	14.12.2 Single String
Vertex	14.12.3 Vertex
Vertex (all)	14.12.6 Vertex (all)
Segment	14.12.4 Segment
Segment (all)	14.12.5 Segment (all)
Vertex/Segment	14.12.7 Vertex/Segment
Attributes	14.12.8 Attributes
Super string dimensions	14.12.9 Super Strings

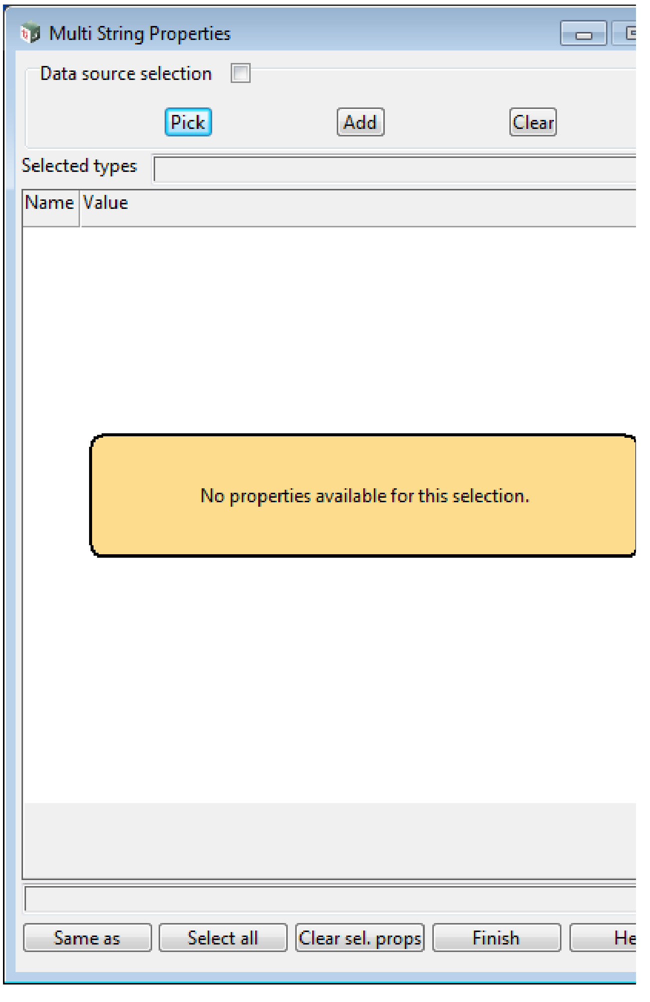
14.12.1 Multi String Properties

Position of option on menu: **Strings =>Properties =>Multi strings**

The **Multi String Properties** panel allows you to select strings and the panel displays all the common properties of the selected strings, and the value of a property if it is the same for all the selected elements.

The value of any of these properties can then be modified and all the selected strings will have the property changed to the modified value.

Selecting **Multi strings** brings up the **Multi String Properties** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source selection	tick box	not ticked	

*if **ticked**, a Data Source is displayed and used to define the strings in the Selection Set.*

Data Source	data source box		

all the strings in the Data Source are selected for the Selection Set.

*If **not ticked**,* **Add** *and* **Clear** *buttons are displayed and used to defined the strings in the selection Set.*

Add	button		

when **Add** *is clicked, strings are selected/deselected using the Multipick method, and are added to any string already selected for the Selection Set by the* **Add** *button.*

The **Add** *button and be used any number of times and each time it adds the new selected items to the Selection Set.*

Clear button

When **Clear** *is clicked, the Selection Set is cleared. That is, there are no strings in the Selection Set.*

The common properties of all the strings in the Selection Set are displayed in the **Name-Value** *property list. If the value of a common property is the same for* ***all*** *the strings in the Selection Set, then the common value is displayed in the* **Value** *column for that property. Otherwise the* **Value** *column for that property is left blank.*

Common Property List

the Common Property Tree displays all the common properties of the selected strings that can be modified.

The name of each common property is displayed in a row in the **Name** *column.*

For a particular common property, if all the selected strings have the same value for that common property then the common value is displayed in the **Value** *column for the row of the common property. Otherwise the value column is blank.*

If a value is ***modified*** *in the* **Value** *column, then for all the strings in the Selection Set, that property is* ***changed*** *to the new value.*

Same As, Select All and Clear Selection Buttons

these three buttons work together. Rows in the ***Property List*** *can be selected (highlighted), and then the* ***Same As*** *button used to select a string whose properties are used to update the strings in the selection set for the highlighted common properties.*

Same As button

when **Same As** *is clicked and an string is selected, then for all the strings in the Selection Set, the values of the highlighted properties in the* ***Common Property List*** *are set to the values of the* ***Same As*** *string.*

Select All button

clicking **Select All** *highlights all the rows of properties in the* ***Property List****. If required, individual rows can then be deselected using Ctrl-LB.*

Clear Property Selection button

clicking **Clear Property Selection** *clears all the highlighted property rows.*

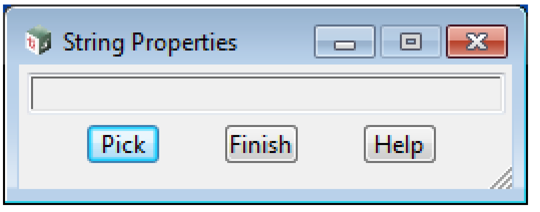
14.12.2 Single String

Position of option on menu: **Strings =>Properties =>Single string**

The **Single string** option obtains and/or modify property information about any string displayed on the screen.

For example, the string's name, colour and style are string properties. Each string type has its own particular set of properties.

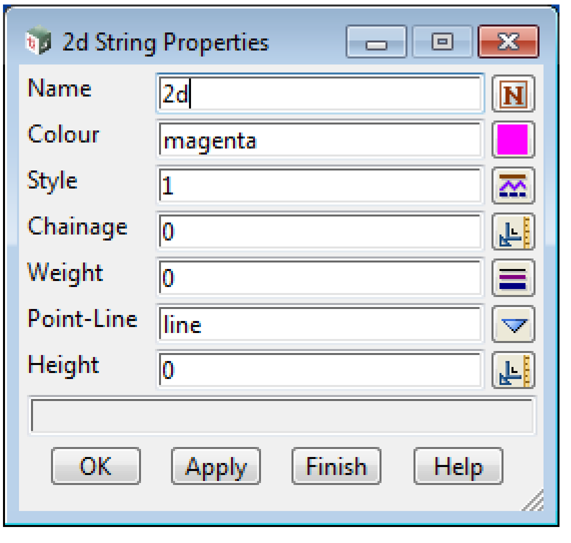
On selecting the **Single string** option, the **String Properties** panel is displayed.



The panel is only used for any special messages and to end the option.

The **String Properties** option is automatically in a pick mode and the user selects strings using the normal 12d Model picking mechanism (LB to select a string, MB to accept a string, RB for the **pick ops** menu). Whenever a string is selected, the appropriate **Properties** panel for the string type is displayed on the screen with all the picked string's properties shown.

For example, for a 2d string, the **Properties** panel is



To modify any of the properties for the selected string, simply change the information in the appropriate panel field and select the **OK or Apply** button.

The **Properties** panel is also brought up by the **Properties** option on each string editor and the **Properties** panel have been shown for each string in the Editor section of the manual.

Please continue to the next section 14.12.3 Vertex.

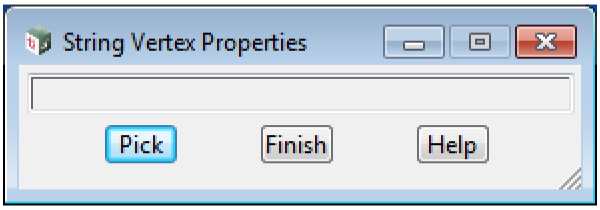
14.12.3 Vertex

Position of option on menu: **Strings =>Properties =>Vertex**

The **Vertex** option obtains and/or modify property information about the vertex of any super string displayed on the screen.

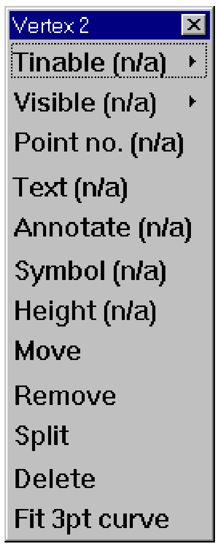
For example, the height, symbol, annotation, point no are all properties of a super string vertex.

On selecting the **Vertex** option, the **String Vertex Properties** panel is displayed.



The panel is only used for any special messages and to end the option.

The **String Vertex Properties** option is automatically in a pick mode and the user selects the *vertex* of a super string using the normal 12d Model picking mechanism. Whenever a *vertex* is selected, a **Vertex** menu showing the properties of the vertex is displayed on the screen.



To modify any of the properties for the selected vertex, simply click on the appropriate item on the **Vertex** menu and the appropriate panel will come up to modify the vertex property.

Please continue to the next section 14.12.4 Segment.

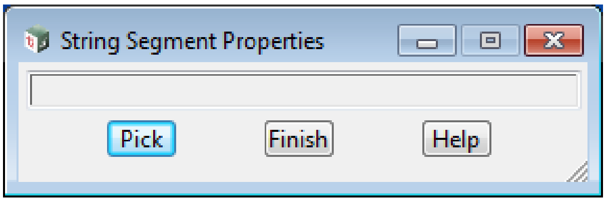
14.12.4 Segment

Position of option on menu: **Strings =>Properties =>Segment**

The **Segment** option obtains and/or modify property information about the segment of any super string displayed on the screen.

For example, the visibility, radius, colour, text are all properties of a super string segment.

On selecting the **Segment** option, the **String Segment Properties** panel is displayed.



The panel is only used for any special messages and to end the option.

The **String Segment Properties** option is automatically in a pick mode and the user selects the *segment* of a super string using the normal 12d Model picking mechanism. Whenever a *segment* is selected, a **Segment** menu showing the properties of the segment is displayed on the screen.

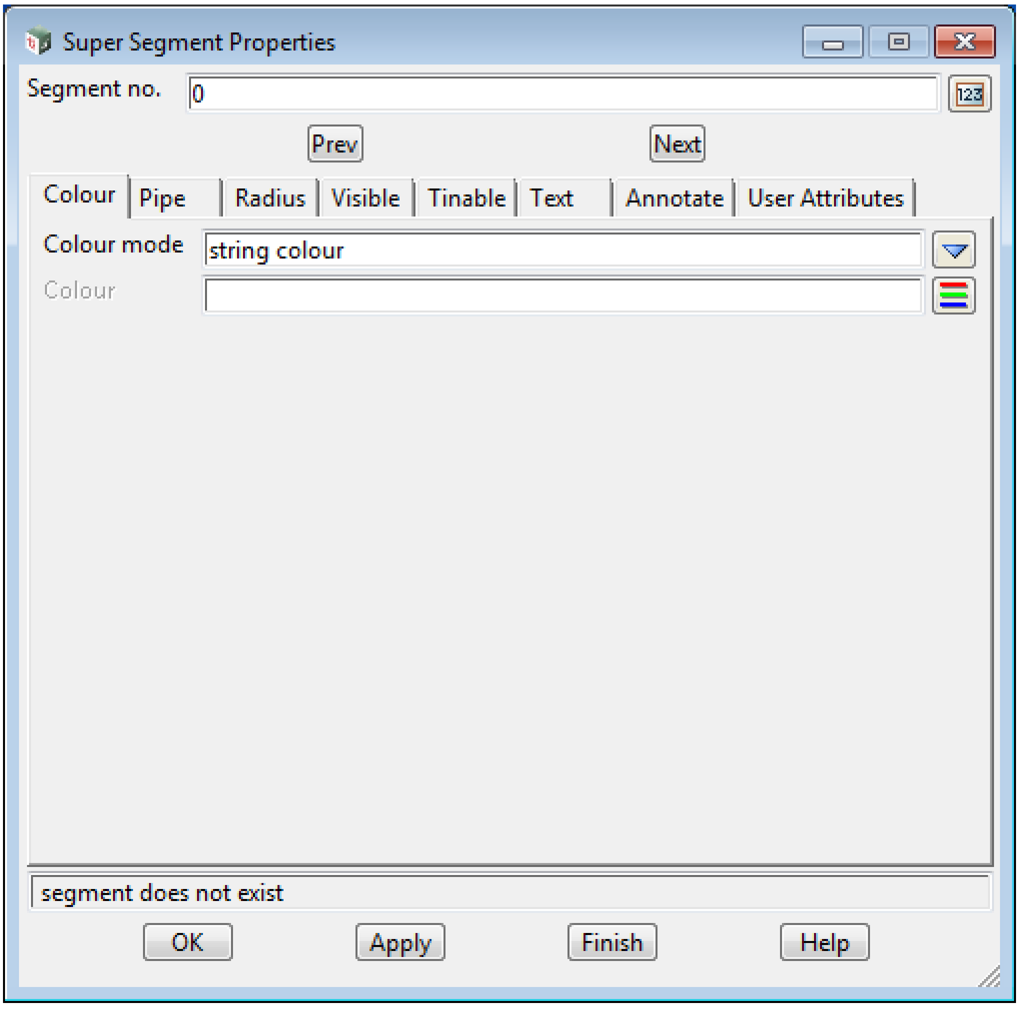
To modify any of the properties for the selected segment, simply click on the appropriate item on the **Segment** menu and the appropriate panel will come up to modify the segment property.

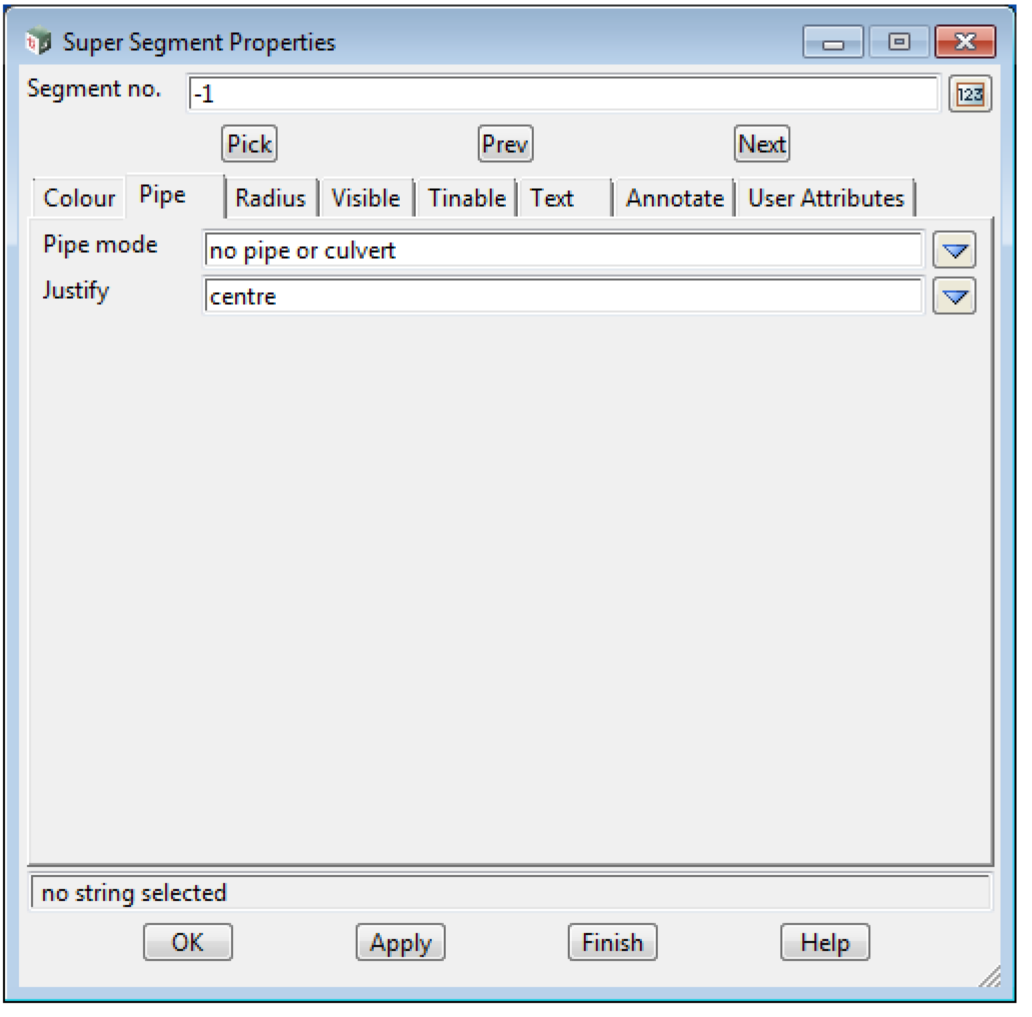
Please continue to the next section 14.12.7 Vertex/Segment.

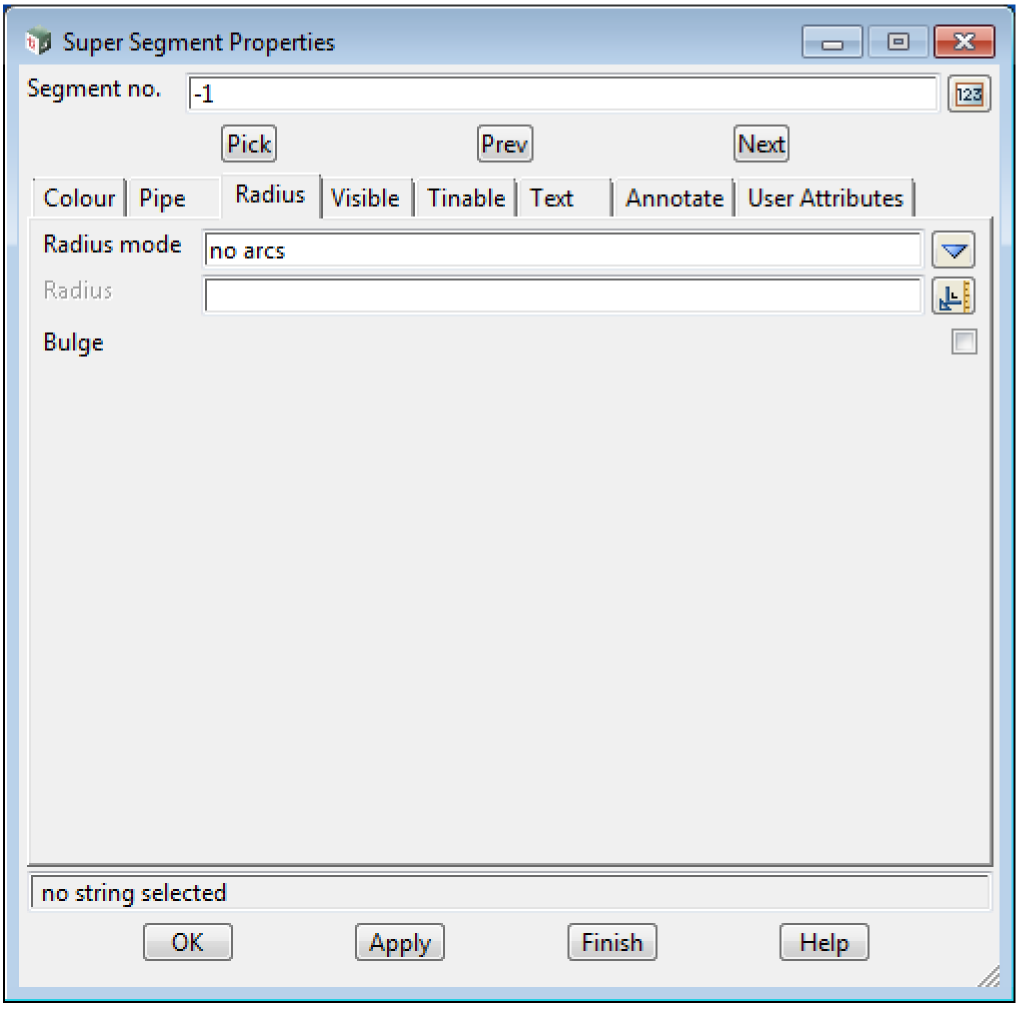
14.12.5 Segment (all)

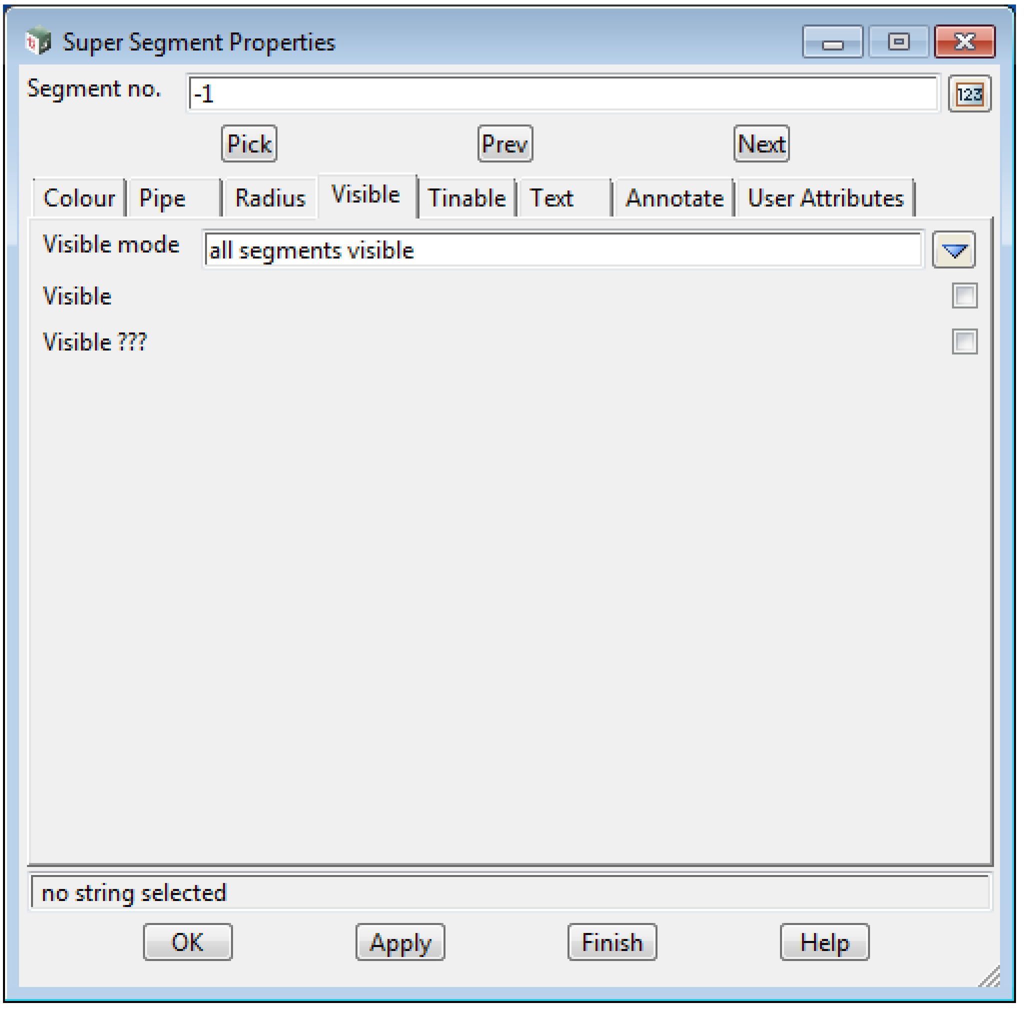
Position of option on menu: **Strings =>Properties =>Segment (all)**

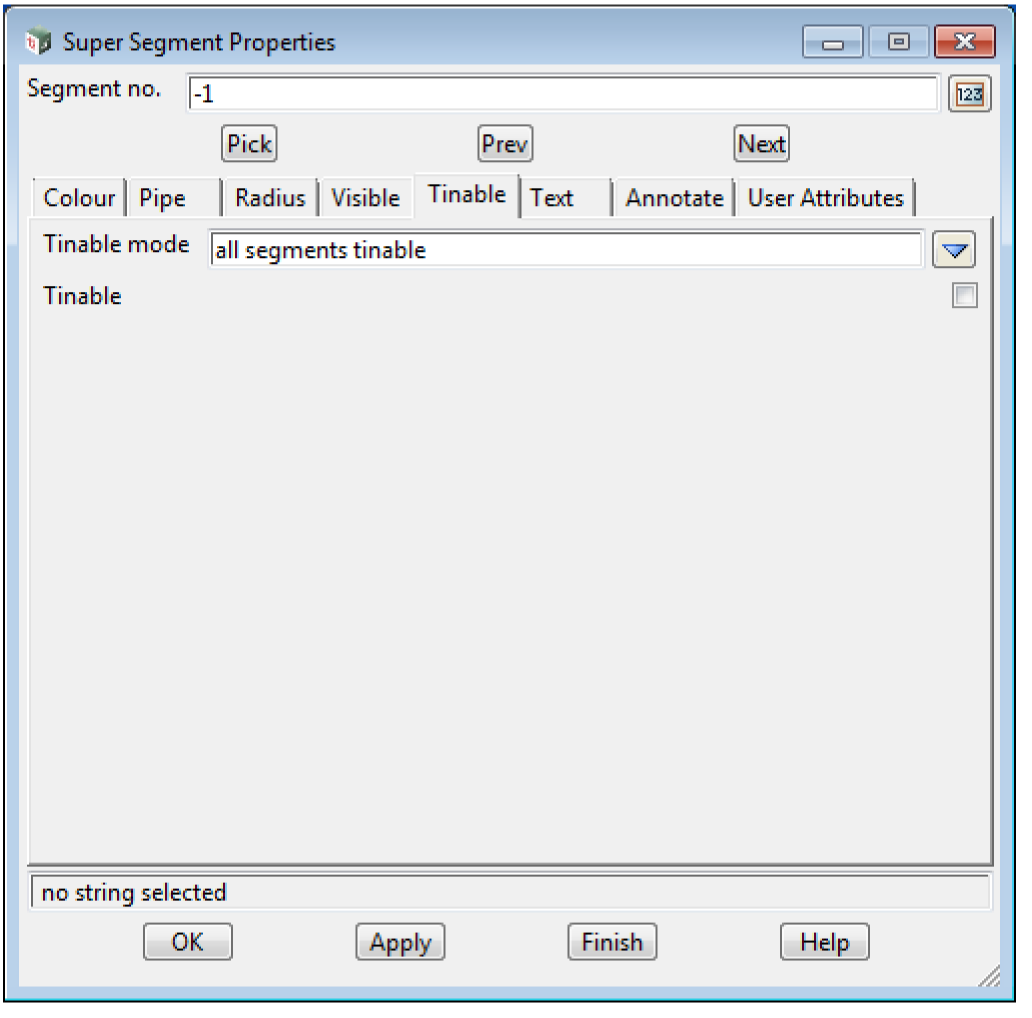
For more information please go to the section 14.4.10.5.9 All Segment Properties



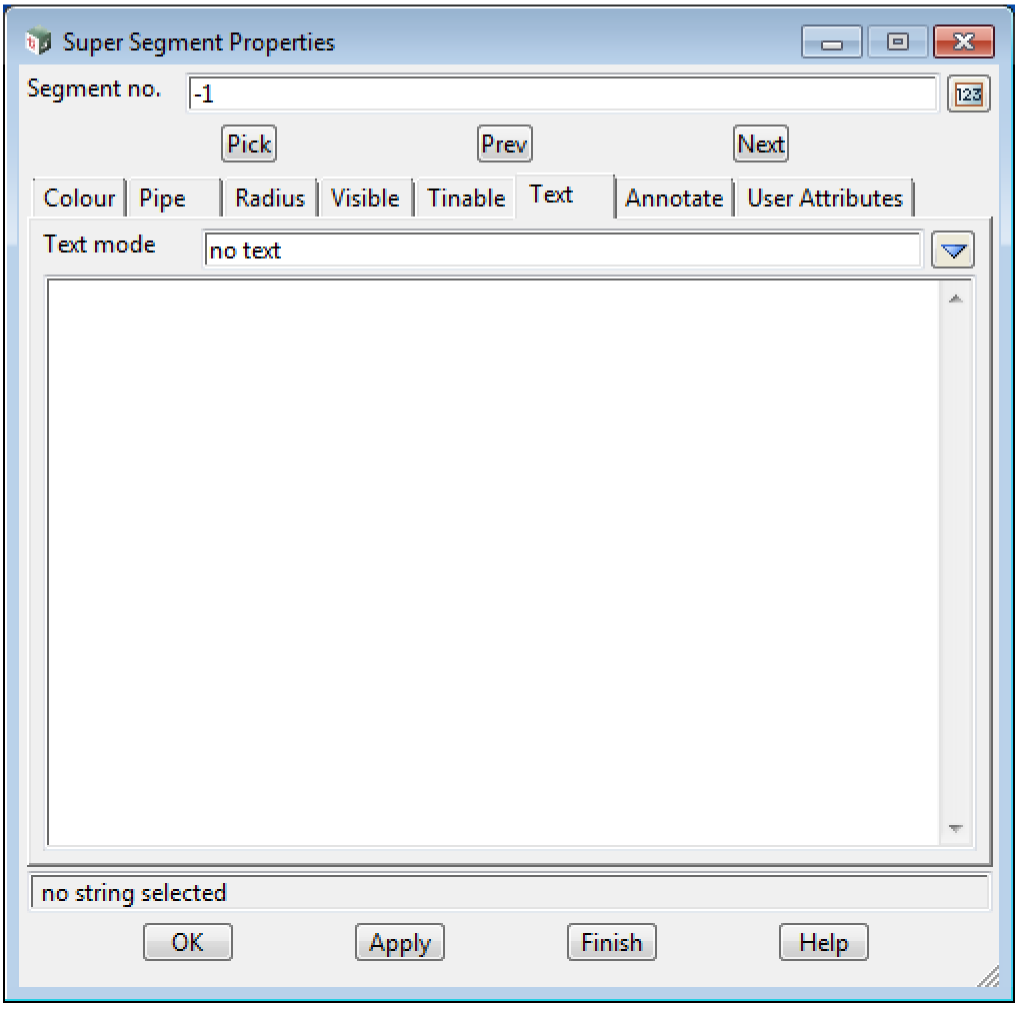

Super Segment Properties
Segment no.
-1
Pick
Prev
Next
Colour
Pipe
Radius
Visible
Tinable
Text
Annotate
User Attributes
Pipe mode
no pipe or culvert
Justify
centre
no string selected
OK
Apply
Finish
Help

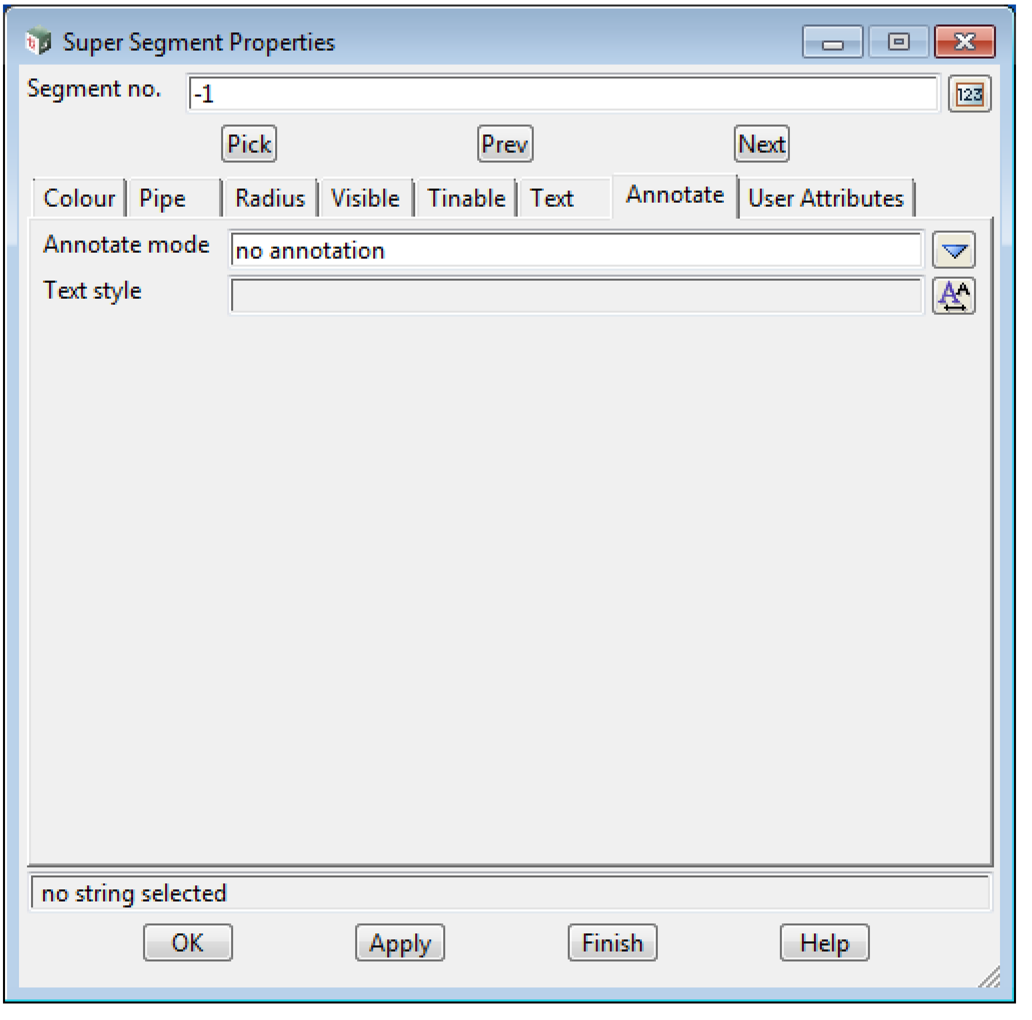

Super Segment Properties
Segment no.
-1
Pick
Prev
Next
Colour
Pipe
Radius
Visible
Tinable
Text
Annotate
User Attributes
Radius mode
no arcs
Radius
Bulge
no string selected
OK
Apply
Finish
Help

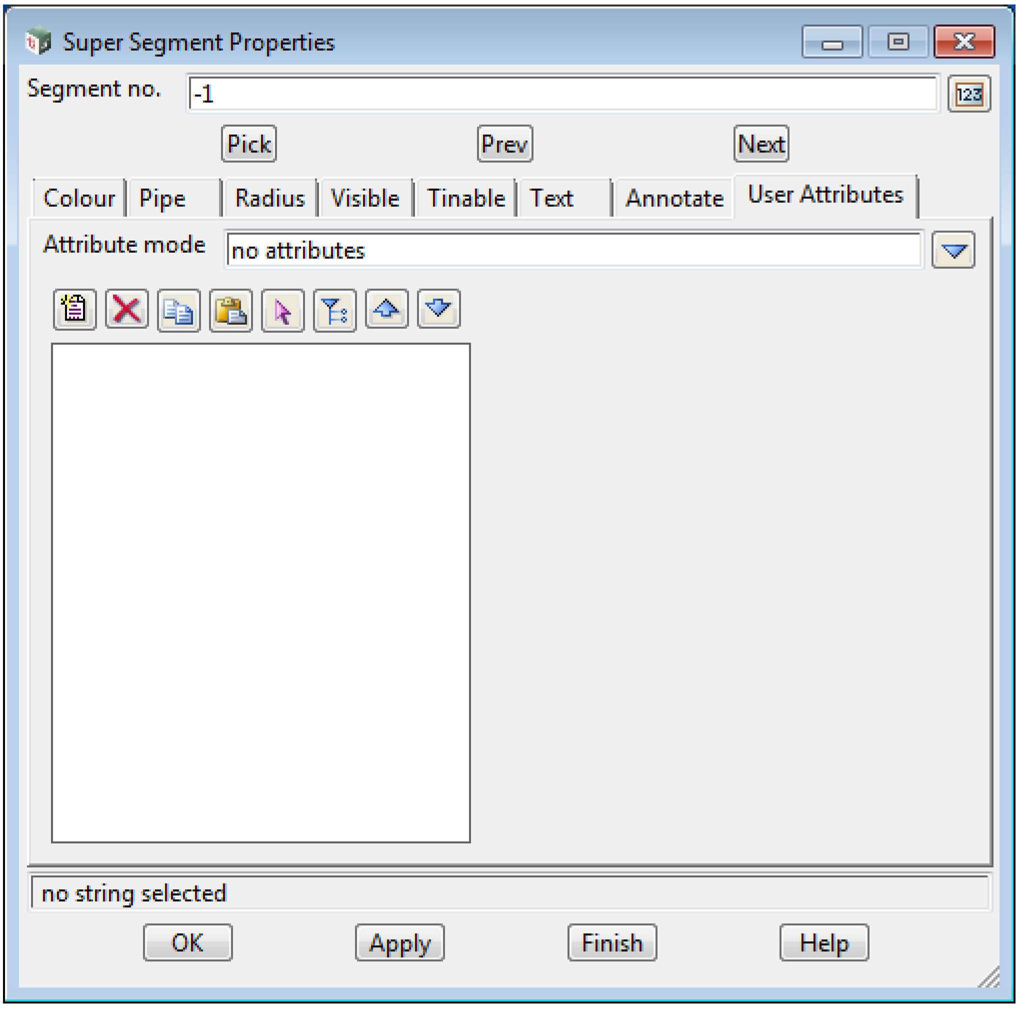

Super Segment Properties
Segment no.
-1
Pick
Prev
Next
Colour
Pipe
Radius
Visible
Tinable
Text
Annotate
User Attributes
Visible mode
all segments visible
Visible
Visible ???
no string selected
OK
Apply
Finish
Help



Super Segment Properties
Segment no.
-1
Pick
Prev
Next
Colour
Pipe
Radius
Visible
Tinable
Text
Annotate
User Attributes
Tinable mode
all segments tinable
Tinable
no string selected
OK
Apply
Finish
Help


Super Segment Properties
Segment no.
-1
Pick
Prev
Next
Colour
Pipe
Radius
Visible
Tinable
Text
Annotate
User Attributes
Text mode
no text
no string selected
OK
Apply
Finish
Help

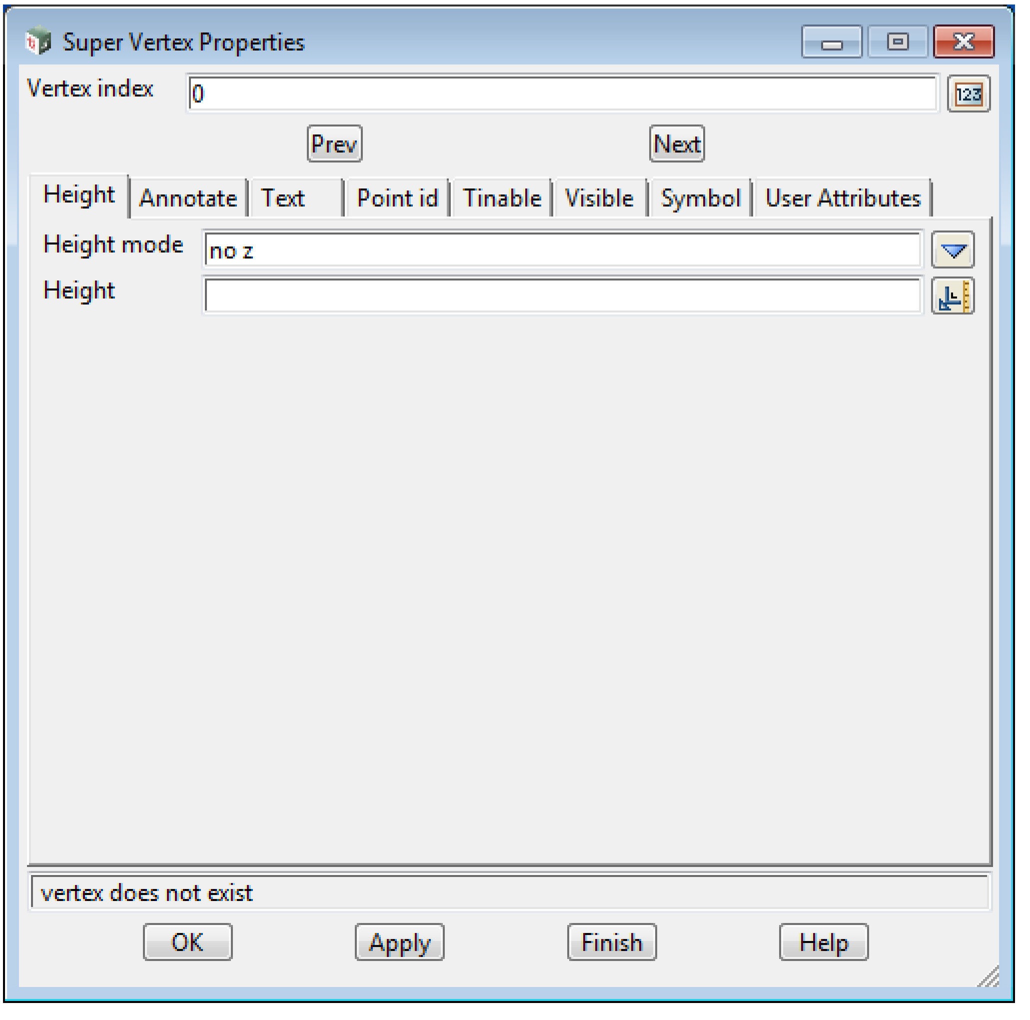

Super Segment Properties
Segment no.
-1
Pick
Prev
Next
Colour
Pipe
Radius
Visible
Tinable
Text
Annotate
User Attributes
Annotate mode
no annotation
Text style
no string selected
OK
Apply
Finish
Help

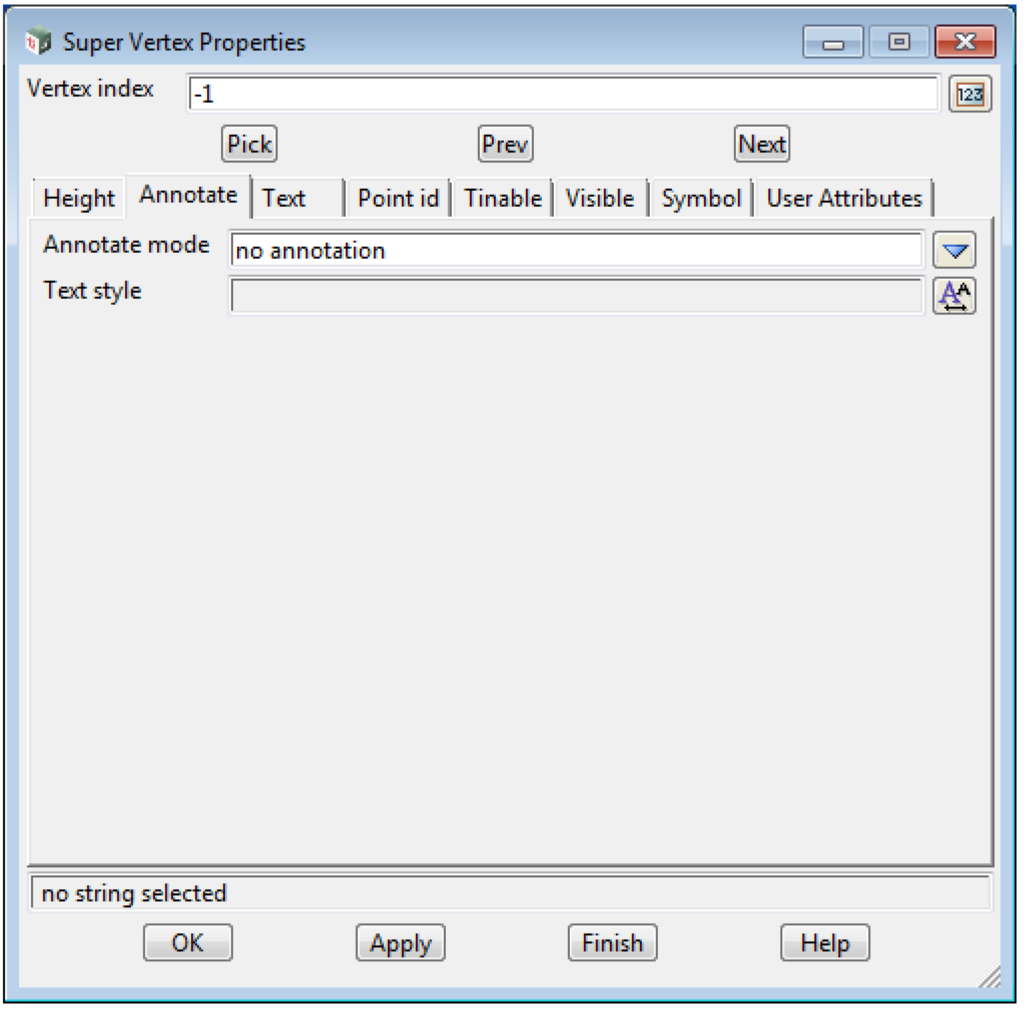

Super Segment Properties
Segment no.
-1
Pick
Prev
Next
Colour
Pipe
Radius
Visible
Tinable
Text
Annotate
User Attributes
Attribute mode
no attributes
no string selected
OK
Apply
Finish
Help

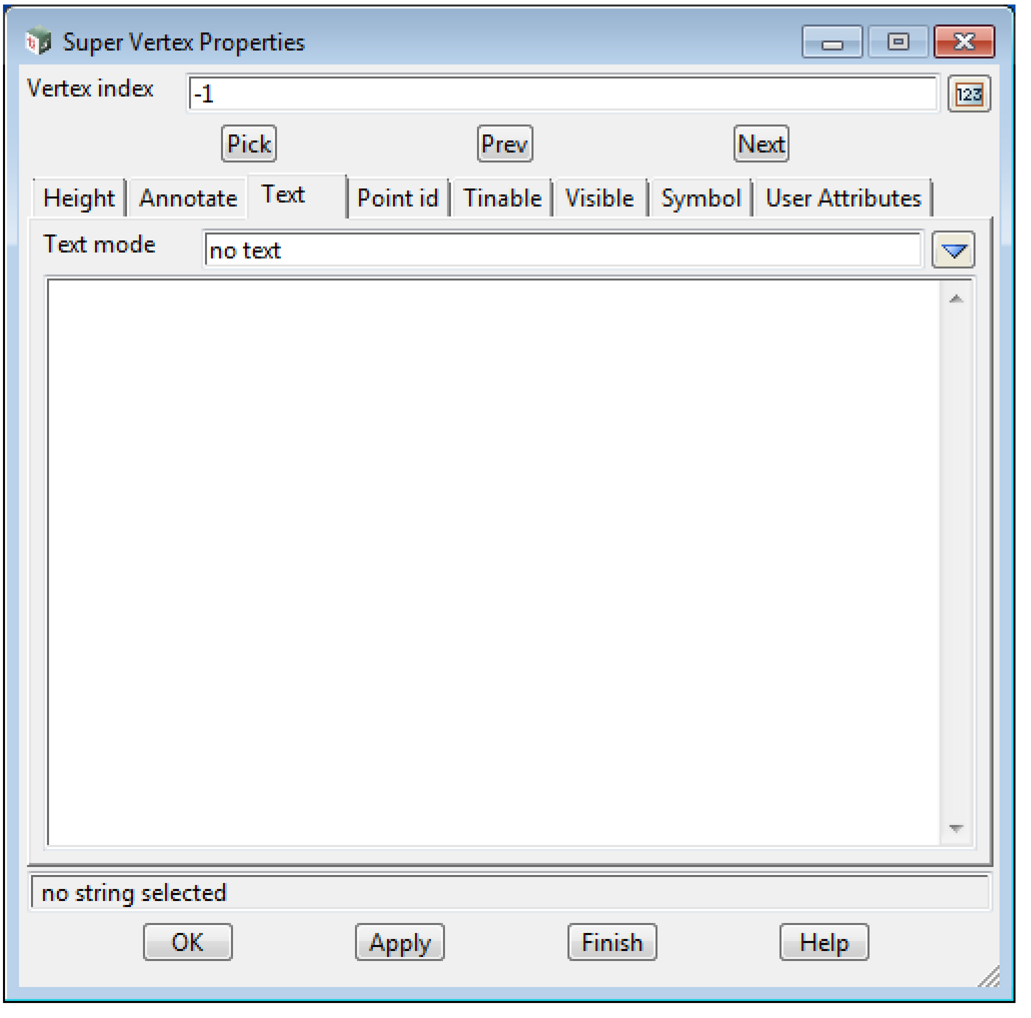
14.12.6 Vertex (all)

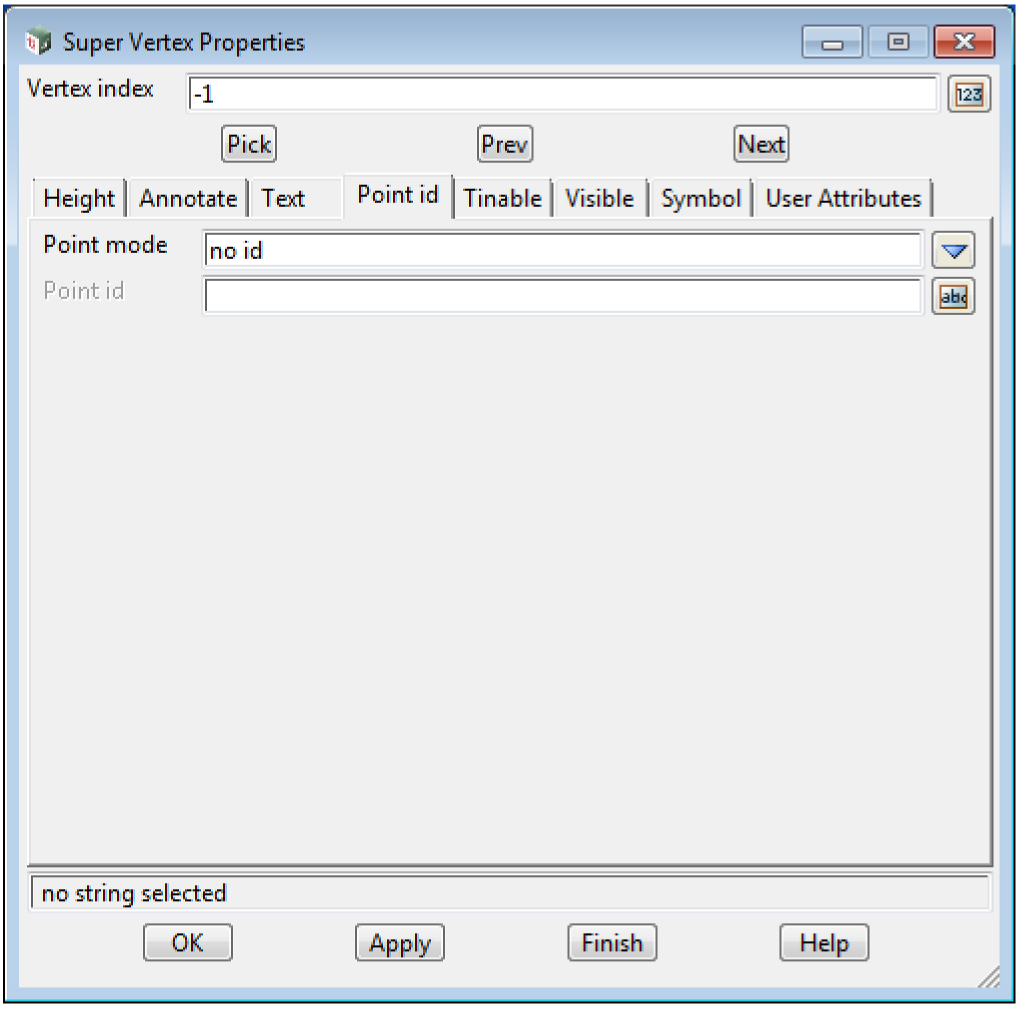
Position of option on menu: **Strings =>Properties =>Vertex (all)**

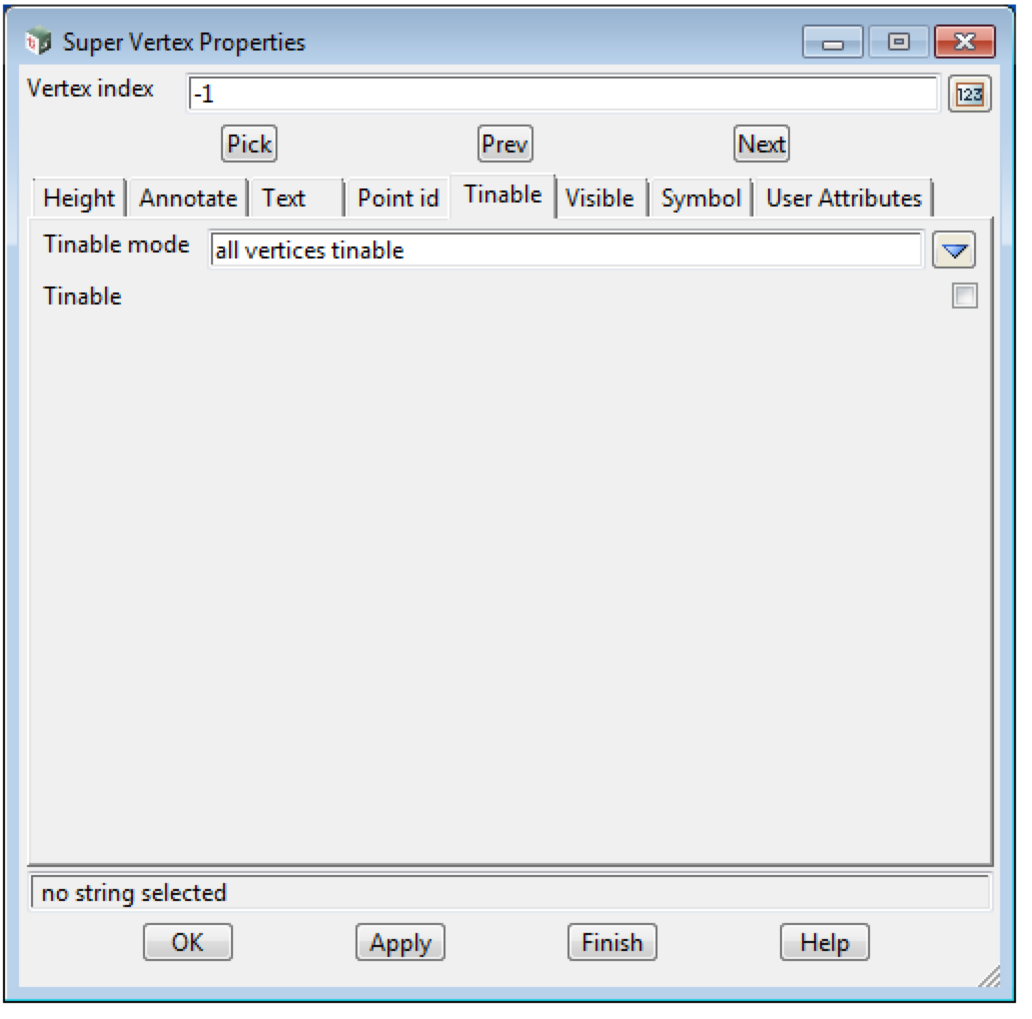
For more information please go to the section 14.4.10.4.10 All Vertex Properties

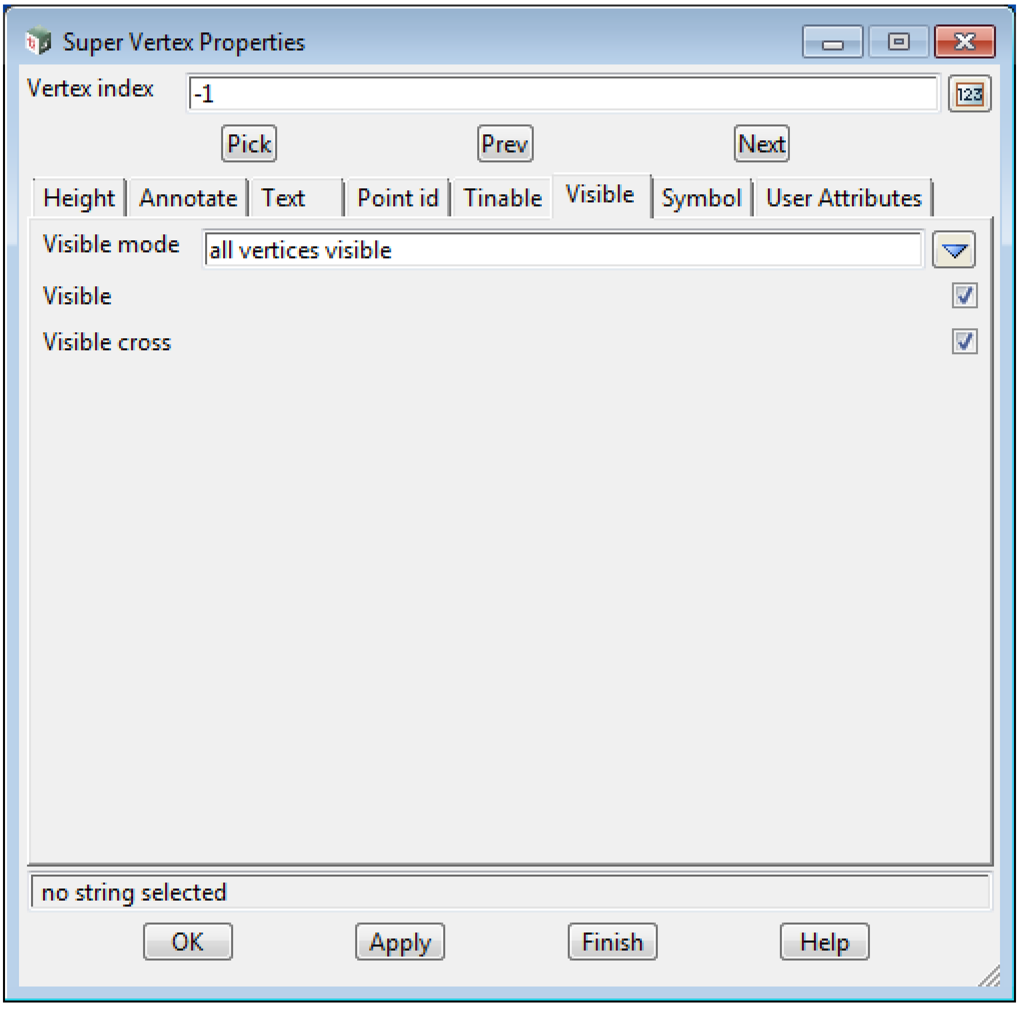


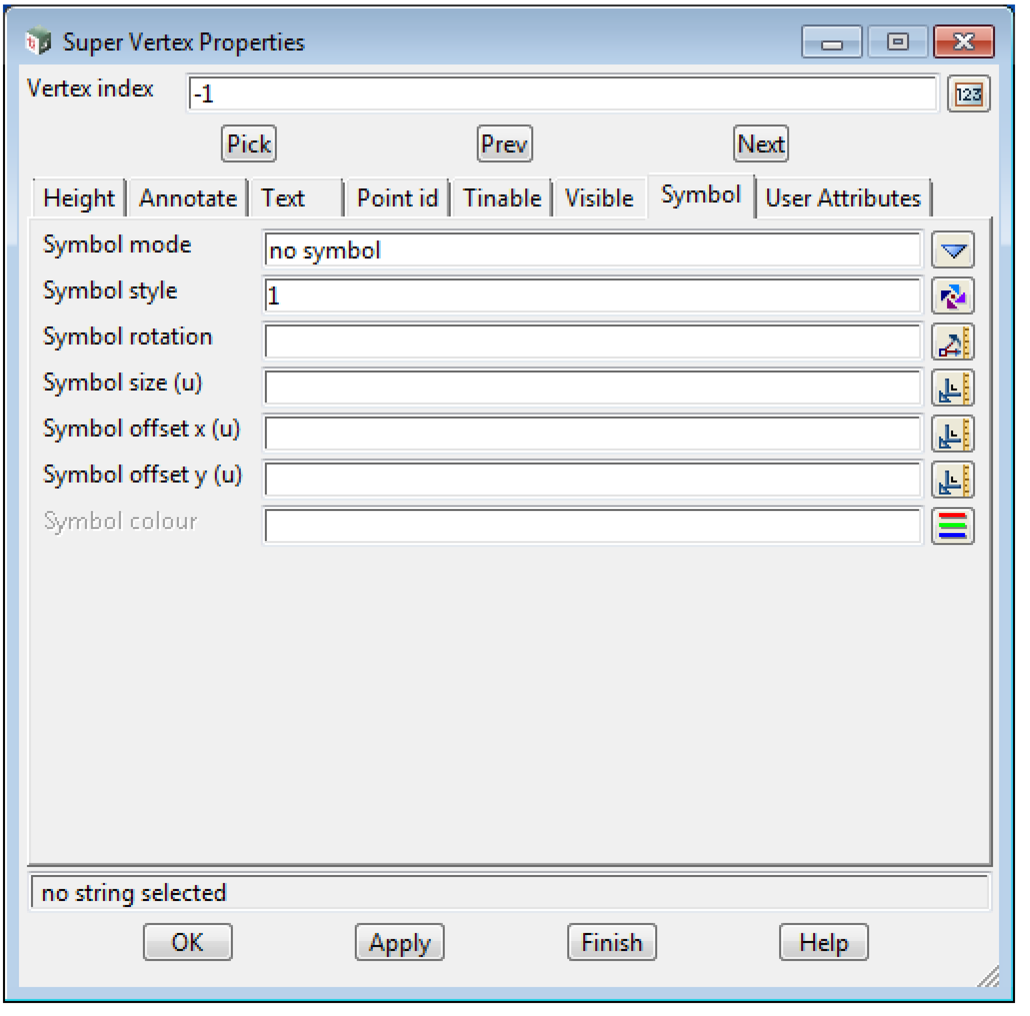

Super Vertex Properties
Vertex index
-1
Pick
Prev
Next
Height
Annotate
Text
Point id
Tinable
Visible
Symbol
User Attributes
Annotate mode
no annotation
Text style
no string selected
OK
Apply
Finish
Help

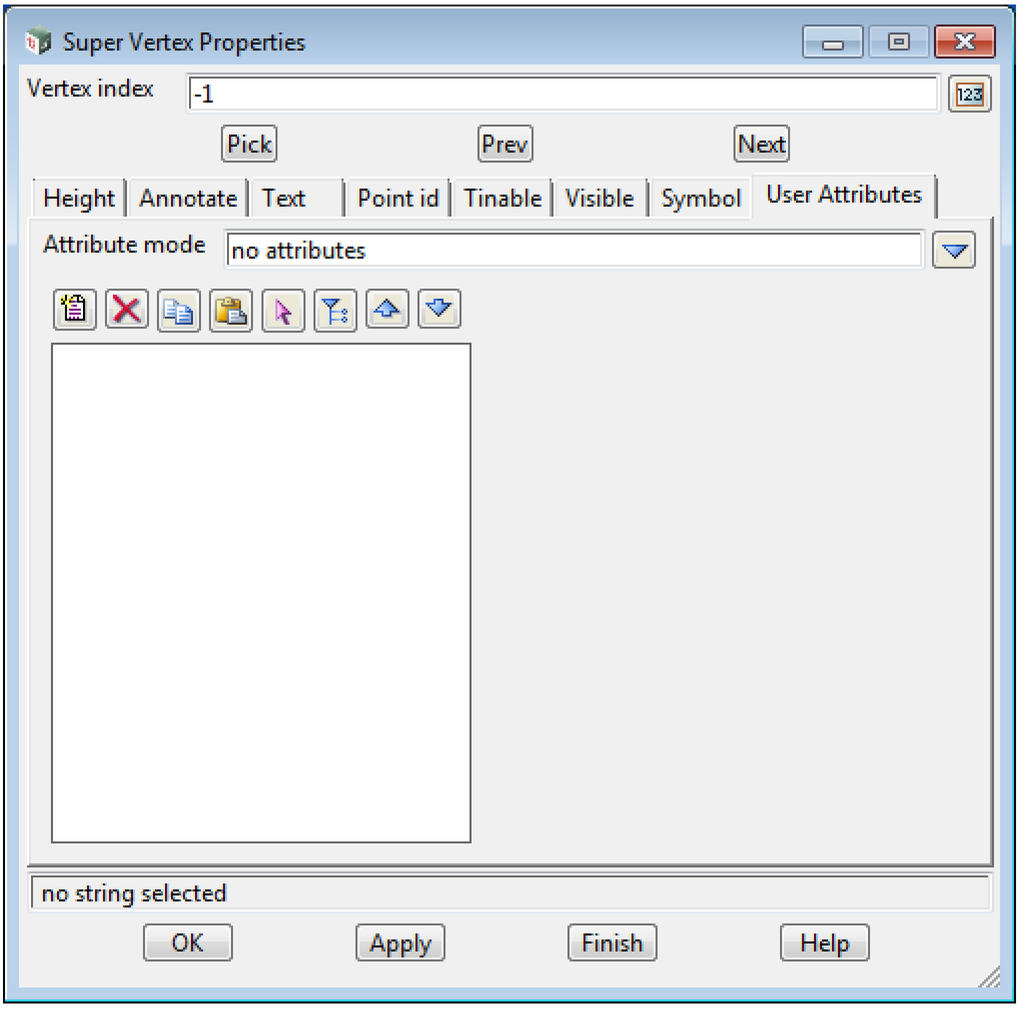

Super Vertex Properties
Vertex index
-1
Pick
Prev
Next
Height
Annotate
Text
Point id
Tinable
Visible
Symbol
User Attributes
Text mode
no text
no string selected
OK
Apply
Finish
Help


Super Vertex Properties
Vertex index
-1
Pick
Prev
Next
Height
Annotate
Text
Point id
Tinable
Visible
Symbol
User Attributes
Point mode
no id
Point id
no string selected
OK
Apply
Finish
Help


Super Vertex Properties
Vertex index
-1
Pick
Prev
Next
Height
Annotate
Text
Point id
Tinable
Visible
Symbol
User Attributes
Tinable mode
all vertices tinable
Tinable
no string selected
OK
Apply
Finish
Help


Super Vertex Properties
Vertex index
-1
Pick
Prev
Next
Height
Annotate
Text
Point id
Tinable
Visible
Symbol
User Attributes
Visible mode
all vertices visible
Visible
Visible cross
no string selected
OK
Apply
Finish
Help


Super Vertex Properties
Vertex index
-1
Pick
Prev
Next
Height
Annotate
Text
Point id
Tinable
Visible
Symbol
User Attributes
Symbol mode
no symbol
Symbol style
1
Symbol rotation
Symbol size (u)
Symbol offset x (u)
Symbol offset y (u)
Symbol colour
no string selected
OK
Apply
Finish
Help

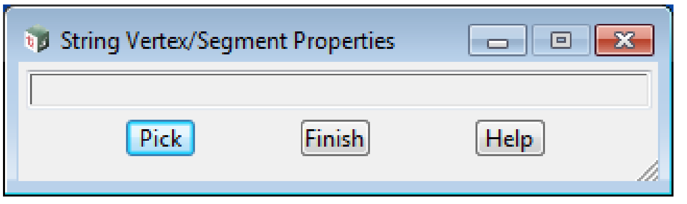

Super Vertex Properties
Vertex index
-1
Pick
Prev
Next
Height
Annotate
Text
Point id
Tinable
Visible
Symbol
User Attributes
Attribute mode
no attributes
no string selected
OK
Apply
Finish
Help

14.12.7 Vertex/Segment

Position of option on menu: **Strings =>Properties =>Vertex/Segment**

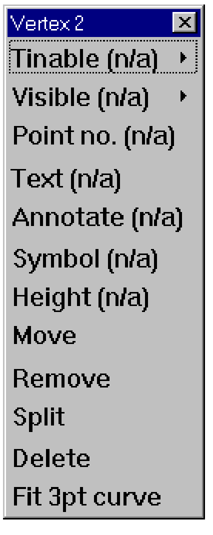
This option is a combination of the Vertex and Segment options already documented.

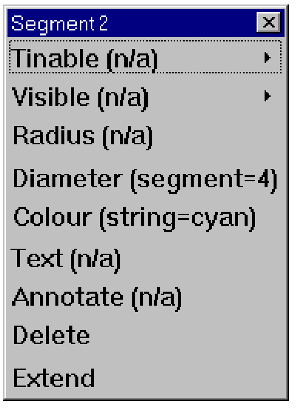
The **Vertex/ Segment** option obtains and/or modify property information about the vertex or segment of any super string displayed on the screen.
On selecting the **Vertex/Segment** option, the **String Vertex/Segment Properties** panel is displayed.



The panel is only used for any special messages and to end the option.

The **String Vertex/Segment Properties** option is automatically in a pick mode and the user selects either the *vertex* or the *segment* of a super string using the normal 12d Model picking mechanism. Whenever a *vertex* or *segment* is selected, the **Vertex** or **Segment** menu showing the properties of the selected vertex or segment is displayed on the screen.



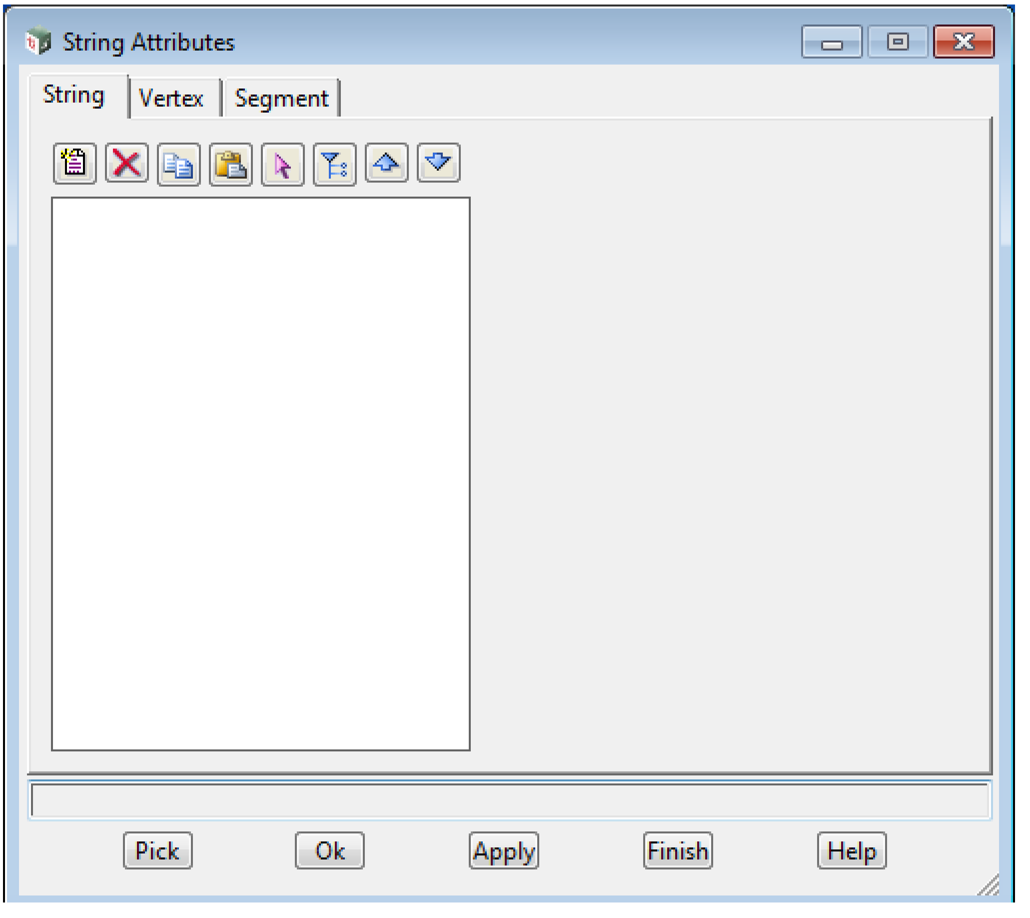


14.12.8 Attributes

Position of option on menu: **String =>Properties =>Attributes**

The **Attributes** options displays, creates and edits attributes for any strings in the project.

On selecting **Attributes**, the **String Attributes** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

String tab

this tab displays/edits the attributes that apply for the entire string. All string types have these.

Name

name of the string attribute

Type integer, real, text

type of attribute - integer, real or text

Data

value for the string attribute

Segment tab

this tab displays/edits the attributes for each segment of a super string or pipe of a drainage or sewer string.

Segment no.

number of the segment of the string to display/edit attributes for

Prev button

go to the previous segment of the string.

Next button

go to the next segment of the string.

Name

name of the attribute for the segment

Type integer, real, text

type of attribute - integer, real or text

Data

value for the attribute for the segment

Vertex tab

this tab displays/edits the attributes for each vertex of a super string or pit of a drainage or sewer string.

Vertex no.

number of the vertex of the string to display/edit attributes for

Prev button

go to the previous vertex of the string.

Next button

go to the next vertex of the string.

Name

name of the attribute for the vertex

Type integer, real, text

type of attribute - integer, real or text

Data

value for the attribute for the vertex

Pick button

pick the string to display/edit the attributes for

OK button

set the attributes to the values in the panel and then exit the panel.

Apply button

set the attributes to the values in the panel but don't exit the panel.

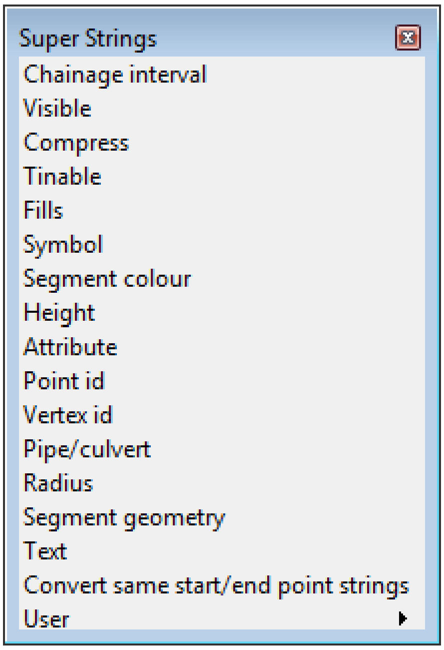
14.12.9 Super Strings

Position of menu: **Strings =>Properties =>Super string dimensions**

Position of menu: **Utilities =>Super strings**

These options change many of the super string properties.

The **Super strings** walk-right menu is:



set/reset the chainage interval and chord/arc tolerance
set visibility flags to constant/variable or clear the flag
compress a super string
set tinability to constant/variable or clear the flag
set fill styles for the string
define symbols constant/each vertex or have no symbols
set a colour for every segment in the super string
set the height(s) of the super string
clear the super string attributes for the string, vertex and/or segment
set the Vertex IDs to sequential numbers or clears the existing vertex IDs
ss segments can have a diameter or a box cross-section
clear any segment radii
clear any segment geometry
set or clear vertex and segment text and annotations
convert strings with same start and end points to closed super strings

For the options

Chainage interval, go to 14.12.9.1 Chainage Interval
Visible 14.12.9.2 Visible
Compress 14.12.9.3 Compress
Tinable 14.12.9.4 Tinable
Fills 14.12.9.5 Fills
Symbol 14.12.9.6 Symbol
Segment colour 14.12.9.7 Segment Colour
Height 14.12.9.8 Height
Attribute 14.12.9.9 Attribute
Vertex id 14.12.9.10 Vertex ID
Pipe/culvert 14.12.9.11 Pipe/Culvert
Radius 14.12.9.12 Radius
Segment geometry 14.12.9.13 Segment Geometry
Text 14.12.9.14 Text
Convert same start/end points strings 28.9.12.7 Same Start/End Point Strings

14.12.9.1 Chainage Interval

Position of option on menu: **Strings =>Properties =>Super string dimensions =>Chainage interval**

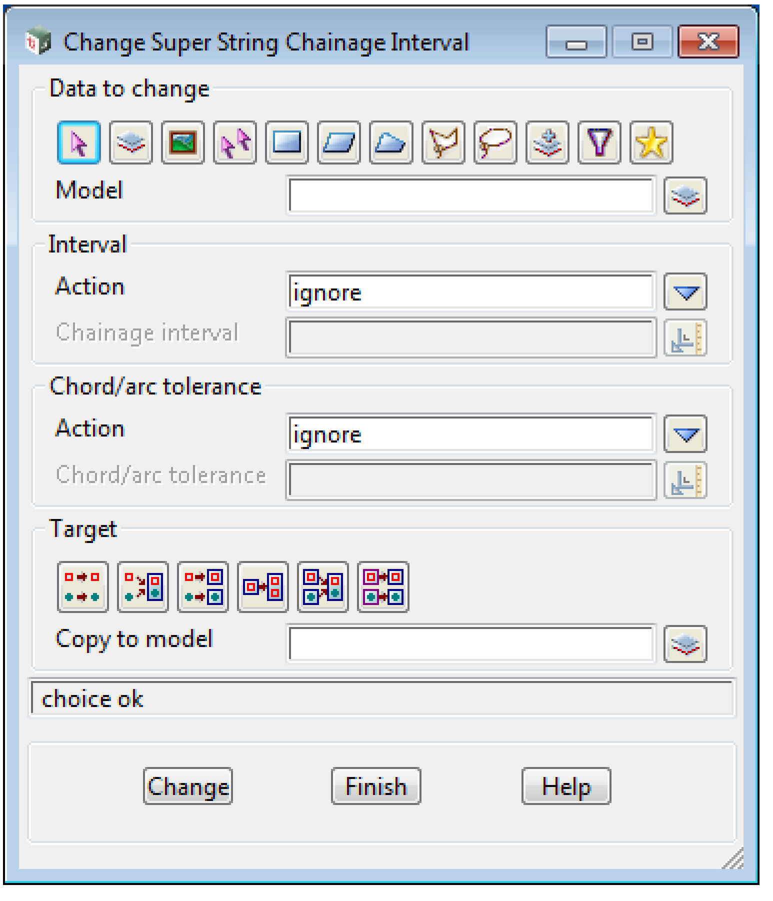
Position of option on menu: **Utilities =>Super strings =>Chainage interval**

A super string has one *chainage interval* value for the entire string. If the value is non-zero, then any line segments of the string are subdivided by the chainage interval when the super string is used in triangulations.

A super string also has one *arc-chord tolerance* value for the entire string. If the value is non-zero, then any curved segments are subdivided by the arc-chord tolerance when used in triangulations.

The **Chainage interval** option can set the *chainage interval* value and/or the *chord-to-arc tolerance* value for selected super strings.

On selecting the **Chainage interval** option, the **Change Super String Chainage Interval** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Interval tab

Action	choice box	ignore	set, clear, ignore

*if **set**, set the chainage intervals of the selected super strings to the value in* Chainage interval*.*
*If **clear**, set all the selected super strings to not use a chainage interval.*
*If **ignore**, don't modify the chainage interval of the selected super strings.*

Chainage interval input

the chainage interval to use for a super string.

Chord/arc tolerance tab

Action choice box ignore set, clear, ignore

*if **set**, set the chord/arc tolerance of the selected super strings to the value in* Chord/arc tolerance*.*
*If **clear**, set all the selected super strings to not use a chord/arc tolerance.*
*If **ignore**, don't modify the chord/arc tolerance of the selected super strings.*

Chord/arc tolerance input

the chord/arc tolerance to use for a super string.

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target.

Target info input

extra information required for the target.

Change button

process the selected strings

14.12.9.2 Visible

Position of option on menu: **Strings =>Properties =>Super string dimensions =>Visible**

Position of option on menu: **Utilities =>Super strings =>Visible**

For selected super strings, the **Visible** option can set the flags that determine how visibility is defined for a super string.

Segment visibility can

(a) be not setable at all for the string - it is always visible

(b) have one setting for the entire string - visible only

(c) have separate settings for each segment (and all set to visible)

Similarly, vertex visibility can

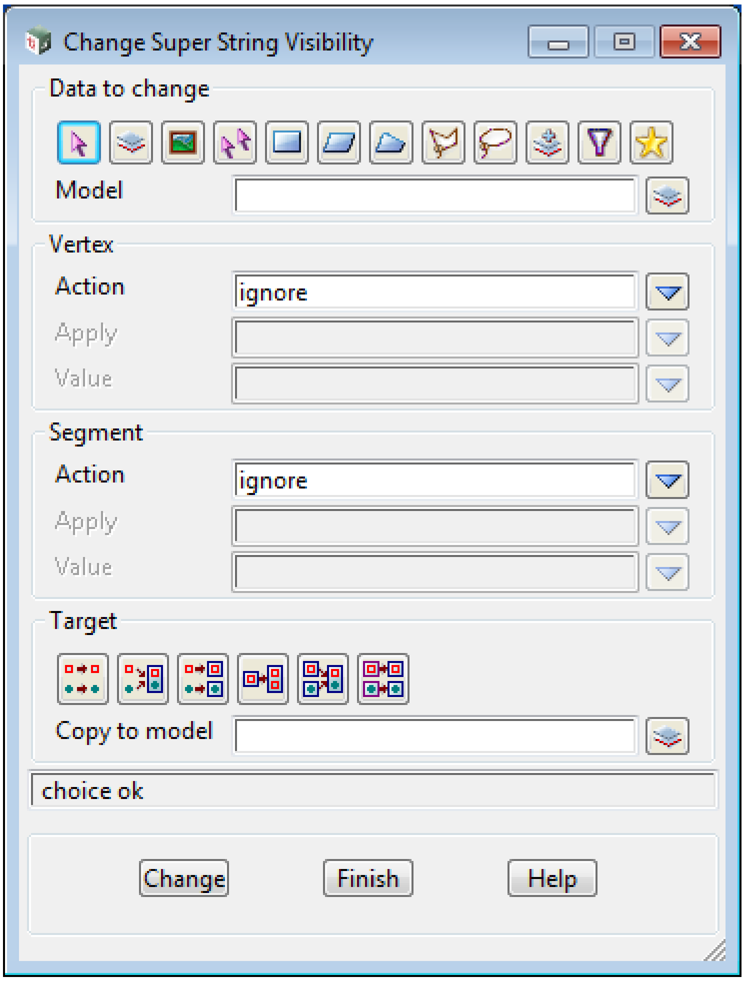
(a) be not setable at all for the string - it is always visible

(b) have one setting for the entire string - visible only

(c) have separate settings for each segment (and all set to visible).

The **Visible** option allows any of the above cases to be set up for the selected super strings.

On selecting the **Visible** option, the **Change Super String Visibility** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source input

source of data to be processed.

Vertex tab

Action choice box ignore set, clear, ignore

*if **set**, the vertex visibility can be allowed to be a constant for all vertices in the string, or to be different for each vertex in the string.*
*If **clear**, the vertex visibility flags are removed for the string and the string vertices are always visible.*
*If **ignore**, don't modify the visibility flags for the vertices of the string.*

Apply choice box

*if **Action** is **set**, then the Apply field is used.*
*If Apply is **constant**, the string has only one visibility flag and it applies to all vertices in the string. It is set to visible.*
*If Apply is **variable**, each vertex in the string has its own visibility flag and each flag is set to visible.*

Segment tab

Action choice box ignore set, clear, ignore

*if **set**, the segment visibility can be allowed to be a constant for all segments in the string, or to be different for each segment in the string.*
*If **clear**, the segment visibility are removed for the string and the string segments are always visible.*
*If **ignore**, don't modify the visibility flags for segments of the string.*

Apply choice box

*if **Action** is **set**, then the Apply field is used.*
*If Apply is **constant**, the string has only one visibility flag and it applies to all segments in the string It is set to visible*
*If Apply is **variable**, each segment in the string has its own visibility flag and each flag is set to visible.*

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target.

Target info input

extra information required for the target.

Change button

process the selected strings.

14.12.9.3 Compress

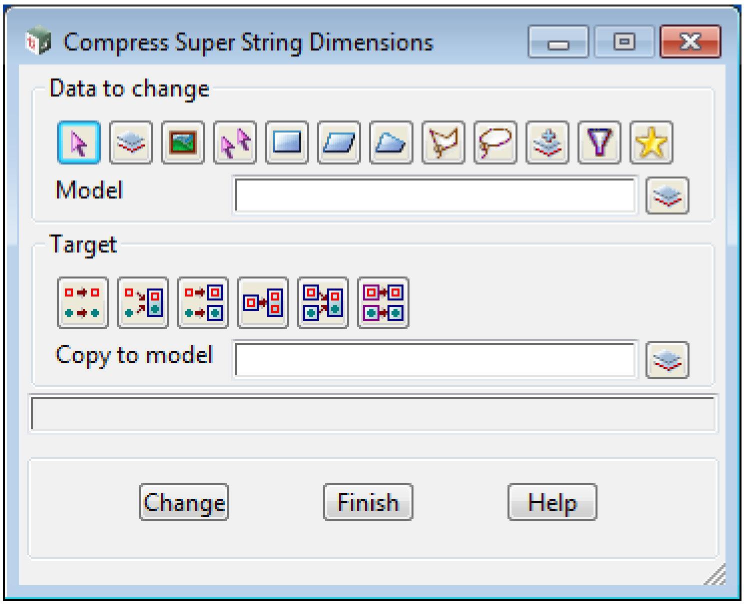
Position of option on menu: **Strings =>Properties =>Super string dimensions =>Compress**

Position of option on menu: **Utilities =>Super strings =>Compress**

The compress option removes any super string dimensions that are not needed. For example, dimensions such radius, variable z, *etc*. that are not being used in the string are removed.

This reduces the storage required for the super string.

On selecting the **Compress** option, the **Compress Super String Dimensions** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Data source type | | Model |

data selection type - for a full description go to 4.19.3 Data Source.

Data source | input

source of data to be processed.

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target.

Target info | input

extra information required for the target.

Change | button

process the selected strings

14.12.9.4 Tinable

Position of option on menu: **Strings =>Properties =>Super string dimensions =>Tinable**

Position of option on menu: **Utilities =>Super strings =>Tinable**

For selected super strings, the **Tinable** option can set the flags that determine how tinability is defined for a super string and can also set whether segments or vertices are tinable or not.

Segment tinability can

(a) be not setable at all for the string - it is always tinable

(b) have one setting for the entire string

(c) have separate settings for each segment.

Similarly, vertex tinability can

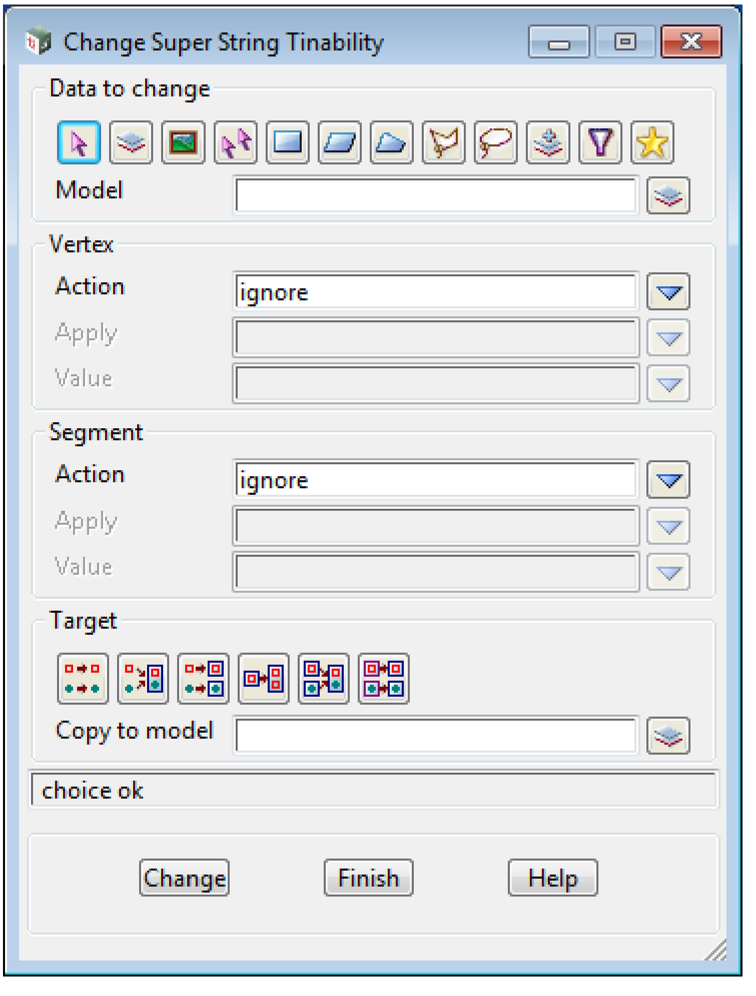
(a) be not setable at all for the string - it is always tinable

(b) have one setting for the entire string

(c) have separate settings for each segment.

The **Tinable** option allows any of the above cases to be set up for the selected super strings.

On selecting the **Tinable** option, the **Change Super String Tinability** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source input

source of data to be processed.

Vertex tab

Action choice box ignore set, clear, ignore

*if **set**, the vertex tinability can be allowed to be a constant for all vertices in the string, or to be different for each vertex in the string.*
*If **clear**, the vertex tinability can not be modified for the string and the string vertices are always tinable.*
*If **ignore**, don't modify the tinability flags for the vertices of the string.*

Apply choice box

*if **Action** is **set**, then the Apply field is used.*
*If Apply is **constant**, the string has only one tinability flag and it applies to all vertices in the string. The tinability flag it is set to the value in the Value field.*
*If Apply is **variable**, each vertex in the string has its own tinability flag. The tinability flag it is set to the value in the Value field.*

Value choice box

*if **Value** is **on**, then the tinable flag is set to **tinable**.*
*If **Value**' is **off**, then the tinable flag is set to **not tinable**.*

Segment tab

Action choice box ignore set, clear, ignore

*if **set**, the segment tinability can be allowed to be a constant for all segments in the string, or to be different for each segment in the string.*
*If **clear**, the segment tinability can not be modified for the string and the string segments are always tinable.*
*If **ignore**, don't modify the tinability flags for segments of the string.*

Apply choice box

*if **Action** is **set**, then the Apply field is used.*
*If Apply is **constant**, the string has only one tinability flag and it applies to all segments in the string.The tinability flag it is set to the value in the Value field.*
*If Apply is **variable**, each segment in the string has its own tinability flag. The tinability flag it is set to the value in the Value field*

Value choice box

*if **Value** is **on**, then the tinable flag is set to **tinable**.*
*If **Value** is **off**, then the tinable flag is set to **not tinable**.*

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target.

Target info input

extra information required for the target.

Change button

process the selected strings

14.12.9.5 Fills

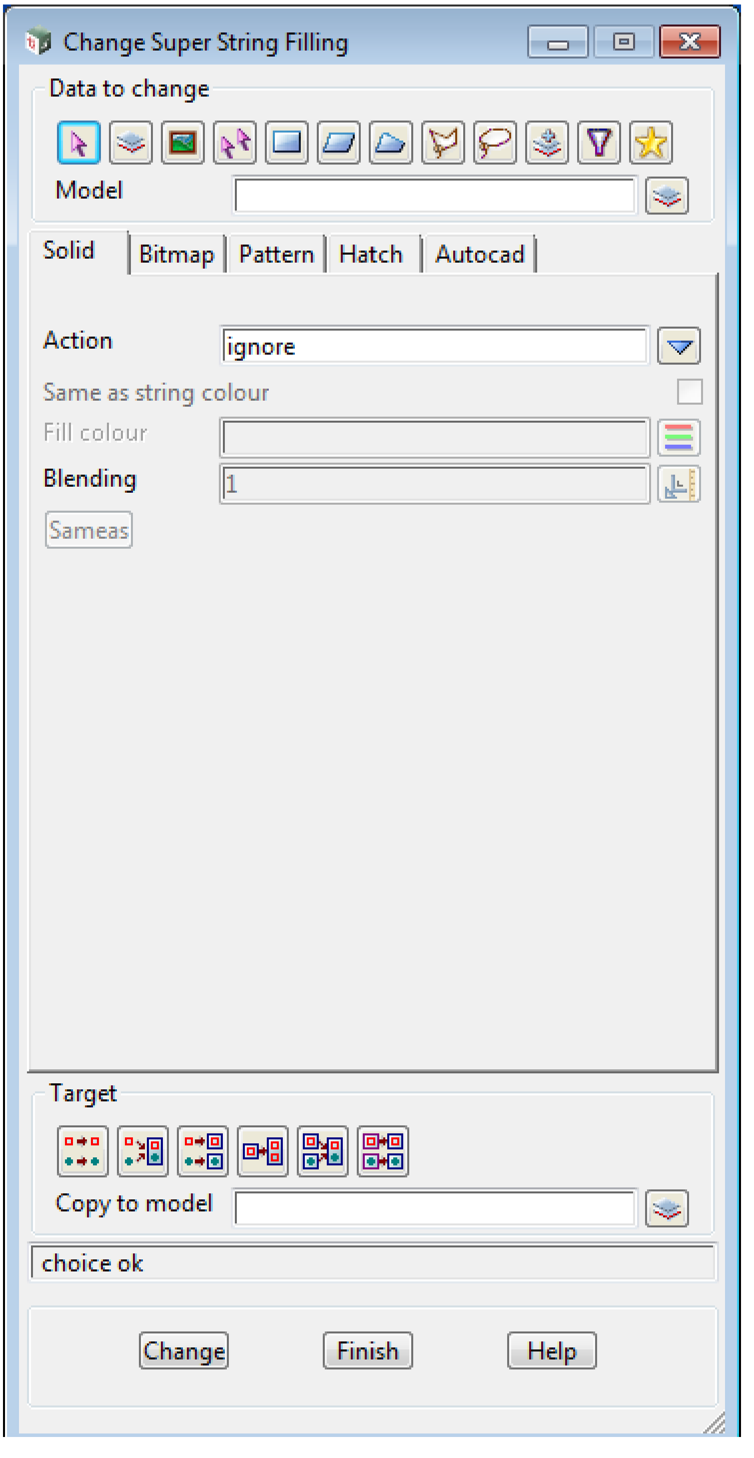
Position of option on menu: **Strings =>Properties =>Super string dimensions =>Fills**

Position of option on menu: **Utilities =>Super strings =>Fills**

The **Fills** option set the flags that determine how filling is defined for a super string. If the string is not closed, the first and last vertices are joined to define the closed region to fill.

A string can have zero, one or more types of fills from the types Solid, Bitmap, Pattern, Hatch and Autocad.

On selecting the **Fill** option, the **Change Super String Filling** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data to change		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Target type

data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target.

Target info input

extra information required for the target.

Same as button on each tab

after selecting the **Same as** *button a string with a fill is selected and if the string has a fill the same as the Tab containing the* **Same as** *button then the values for that fill in of the string is used to fill in the appropriate fields of the Tab.*

Buttons at Bottom

Change button

process the selected strings

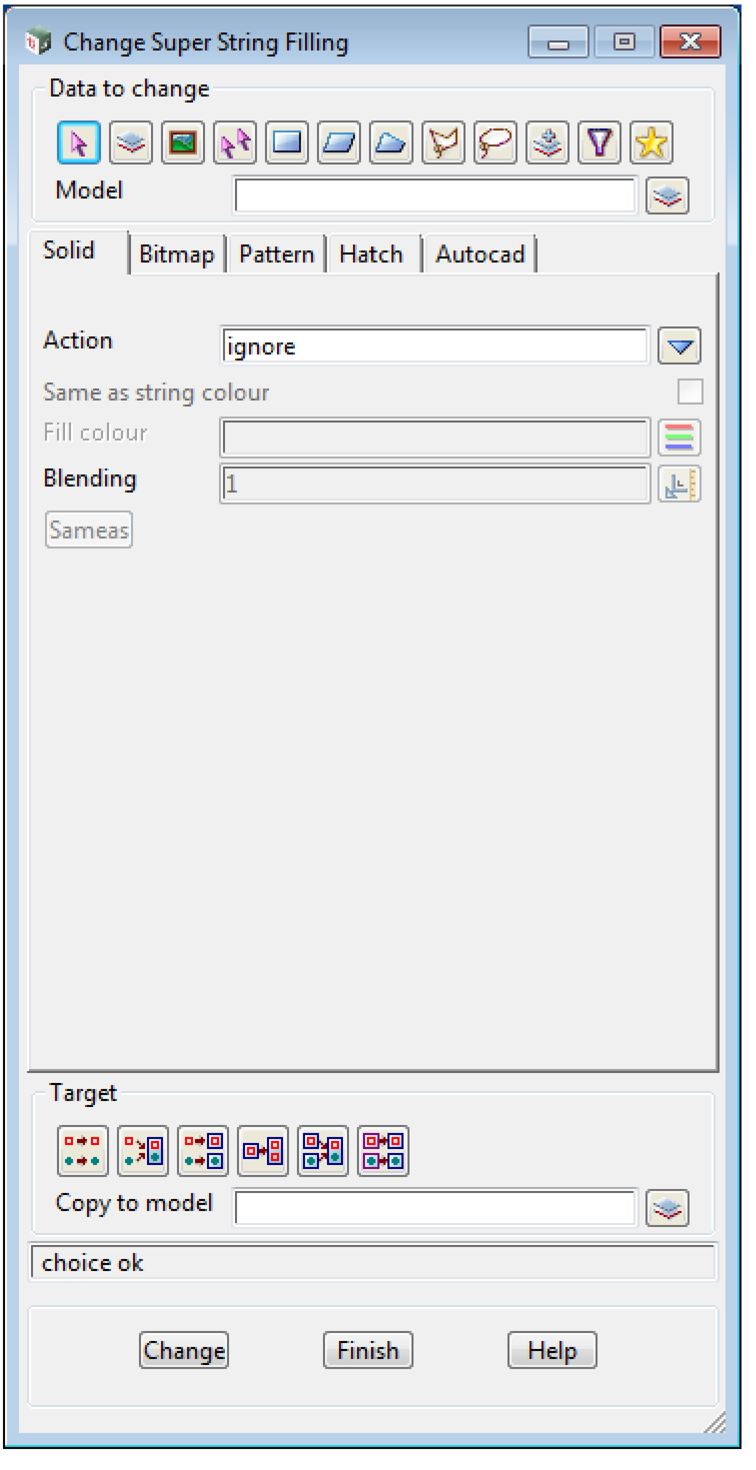
Solid, Bitmap, Pattern, Hatch and Autocad Tabs

each tab defines how that type of fill is applied to the string. Zero, one or more fill types can be applied to a string.Each tab will now be described.

See

14.12.9.5.1 Solid Tab
14.12.9.5.2 Bitmap Tab
14.12.9.5.3 Pattern Tab
14.12.9.5.4 Hatch tab
14.12.9.5.5 Autocad tab

14.12.9.5.1 Solid Tab



The fields and buttons used in the panel have the following functions.

Action choice box ignore set, clear, ignore

*if **set**, this type of fill is set for the string.*
*If **clear**, this type of fill is turned off for the string.*
*If **ignore**, don't modify this type of fill for the string.*

Same as string colour tick box

*if **ticked**, then for all the strings that are to have their **Solid fills** modified, the colour of the super string* *solid fill is set to be the same as the string colour.*

If ***not ticked****, the fill colour is used for the* ***Solid fill****.*

Fill colour colour box

if Same as string colour *is not ticked, the* Fill colour *is used to fill the super string. If the super string is not closed, the fill will apply as if the super string was closed.*

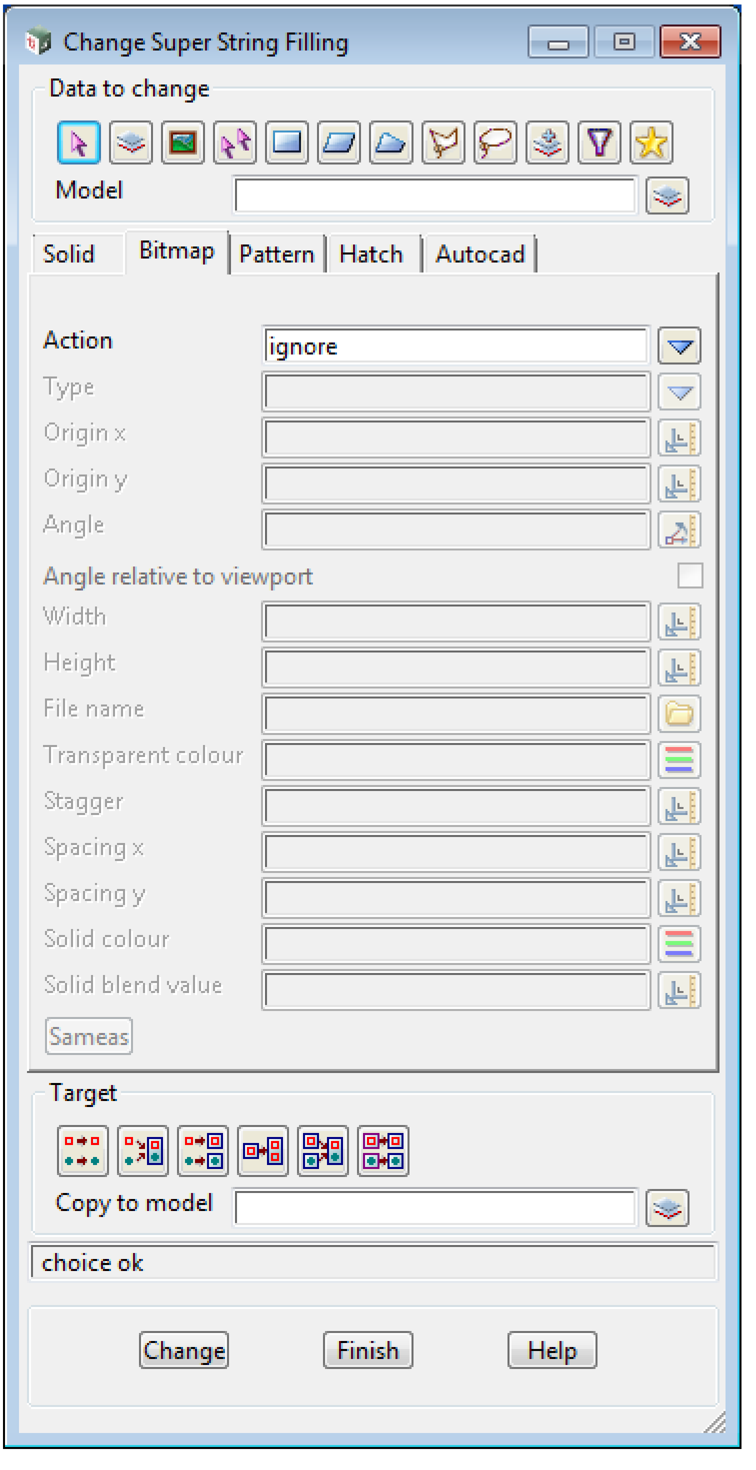
Blending real box

the blend value is used for the solid fill.This value is between 0.0 and 1.0 and controls the amount of transparency of the fill.

if blank, a blend value of 1.0 is assumed.

Continue to 14.12.9.5.2 Bitmap Tab or return to 14.12.9.5 Fills.

14.12.9.5.2 Bitmap Tab



The fields and buttons used in the panel have the following functions.

Action choice box ignore set, clear, ignore

*if **set**, this type of fill is set for the string.*
*If **clear**, this type of fill is turned off for the string.*
*If **ignore**, don't modify this type of fill for the string.*

Type choice box device, world, paper

the units of Width, Height, Stagger, *and* Spacing. *This means the bitmap can be drawn in either world* *size, paper size (mm) or device size.*

Origin X real box

the x co-ordinate of the origin point of the repeated pattern.

Origin Y real box

the y co-ordinate of the origin point of the repeated pattern.

Angle angle

the orientation of the bitmap pattern relative to the x axis measured in the anti-clockwise direction in dms.

Angle relative to view tick box

this field controls whether the Angle is relative to the x axis or to the plotting x axis.

*If **ticked** and we are plotting,* **Angle** *is measured relative to the x axis of the plot rotation.*

*If **not ticked**,* **Angle** *is always absolute to the world x axis.*

Height real box

the height of the bitmap in the selected units.

File name file box .*bmp files

the bitmap file to be used for filling. Only BMP files are currently supported.

Transparent colour colour box **Select Colour** pop up

the transparency colour within the bitmap.This means any pixel of this colour within the bitmap is not drawn.

Stagger real box

if not blank, this field allows each alternate row of bitmaps to be staggered (or offset) by the specified value. Stagger is measured in the selected units along the x axis in the rotated system.

If blank, a Stagger value of 0.0 is assumed.

Spacing X input

the distance in the selected units along the x axis in the rotated system between each instance of the bitmap.

Spacing Y input

the distance in the selected units along the y axis in the rotated system between each instance of the bitmap.

Solid colour colour box **Select Colour** pop up

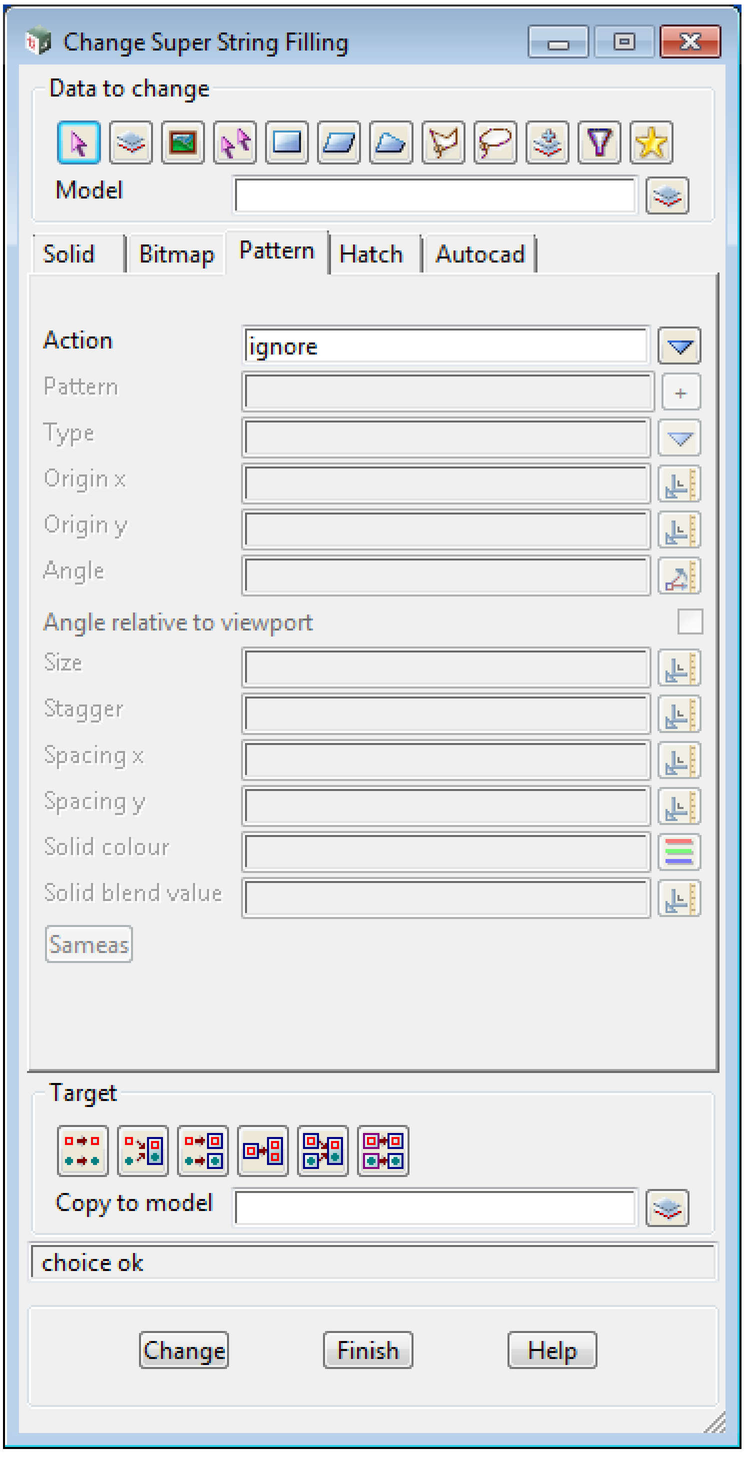
if not blank, when the bitmaps become too small to be legible, the super string is solid filled instead with **Solid colour**.

Solid blend value input

if not blank, and **Solid Colour** *is specified, the solid fill is drawn with transparency. The blend value is between 0.0 and 1.0 and controls the transparency of the solid colour.*

Continue to 14.12.9.5.3 Pattern Tab or return to 14.12.9.5 Fills.

14.12.9.5.3 Pattern Tab



The fields and buttons used in the panel have the following functions.

Action choice box ignore set, clear, ignore

*if **set**, this type of fill is set for the string.*
*If **clear**, this type of fill is turned off for the string.*
*If **ignore**, don't modify this type of fill for the string.*

Pattern **select pattern** pop up

the pattern name to be applied. The patterns are defined in the patterns.4d

For more information on the 12d pattern definition go to the section 40.4 Patterns.

Type choice box device, world, paper

the units of Size*. This means the pattern can be drawn as either world size, paper size (mm) or screen (device) size (pixels).*

Origin X real box

the x co-ordinate of the origin point of the repeated pattern.

Origin Y real box

the y co-ordinate of the origin point of the repeated pattern.

Angle angle

the orientation of the pattern relative to the x axis measured in the anti-clockwise direction in dms in hp notation.

Angle relative to view tick box

this field controls whether the Angle is relative to the x axis or to the plotting x axis.

*If **ticked** and a plot is being generated,* **Angle** *is measured relative to the x axis of the rotation of the plot.*

*If **not ticked**,* **Angle** *is always absolute to the world x axis.*

Size real box real greater of equal to 1.

the size of the pattern in the selected units.

If 0 then the pattern is not drawn.

Stagger real box

if not blank, this field allows each alternate row of patterns to be staggered (or offset) by the specified value. Stagger is measured in the selected units along the x axis in the rotated system.

If blank, a Stagger value of 0 is assumed.

Spacing X real box

the distance in the selected units along the x axis in the rotated system between each instance of the pattern.

Spacing Y real box

the distance in the selected units along the y axis in the rotated system between each instance of the pattern.

Solid colour colour box **Select Colour** pop up

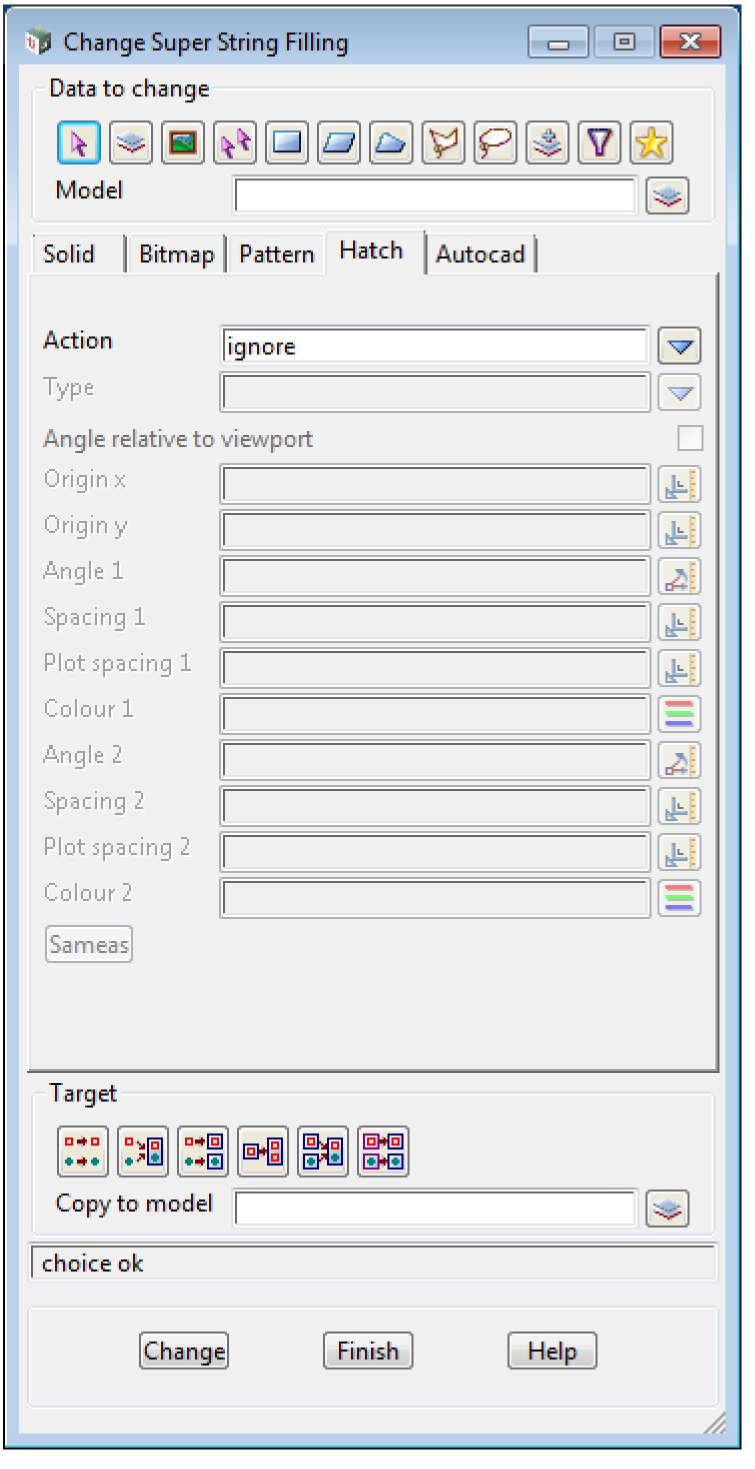
if not blank, when the patterns become too small to be legible, the super string is solid filled instead with **Solid colour**

Solid blend value real box

if not blank, and **Solid Colour** *is specified, the solid fill is drawn with transparency. The blend value is between 0.0 and 1.0 and controls the transparency of the solid colour.*

Continue to 14.12.9.5.4 Hatch tab or return to 14.12.9.5 Fills.

14.12.9.5.4 Hatch tab



The fields and buttons used in the panel have the following functions.

Action choice box ignore set, clear, ignore

*if **set**, this type of fill is set for the string.*
*If **clear**, this type of fill is turned off for the string.*
*If **ignore**, don't modify this type of fill for the string.*

Type choice box device, world, paper

the units of Spacing and Plot spacing. *This means the bitmap can be drawn in either world size, paper* *size (mm) or screen (device) size (pixels).*

Angle relative to view tick box

this field controls whether the Angle is relative to the x axis or to the plotting x axis.

*if **ticked** and we are plotting,* **Angle** *is measured relative to the x axis of the plot rotation.*

*If **not ticked**,* **Angle** *is always absolute to the world x axis.*

Origin X real box

the x co-ordinate of the anchor point of the repeated pattern.

Origin Y real box

the y co-ordinate of the anchor point of the repeated pattern.

Angle 1 angle

the first orientation of the hatch pattern relative to the x axis measured in the anti-clockwise direction in dms.

Spacing 1 real box

the first distance in the units selected in Type between the lines drawn at **Angle 1**.

Plot spacing 1 real box

the first distance in the units selected in Type between the lines drawn at **Angle 1** *except it is the value used when you are plotting to or doing a plot of the super string containing the fill.*
if the field is blank it uses the **Spacing 1** *value.*

Colour 1 colour box **Select Colour** pop up

the colour of all the lines drawn at **Angle 1**.

Angle 2 angle

the second orientation of the hatch pattern relative to the x axis measured in the anti-clockwise direction in dms.

Spacing 2 real box

the second distance in the units selected in Type column between the lines drawn at **Angle 2**.

Plot spacing 2 real box

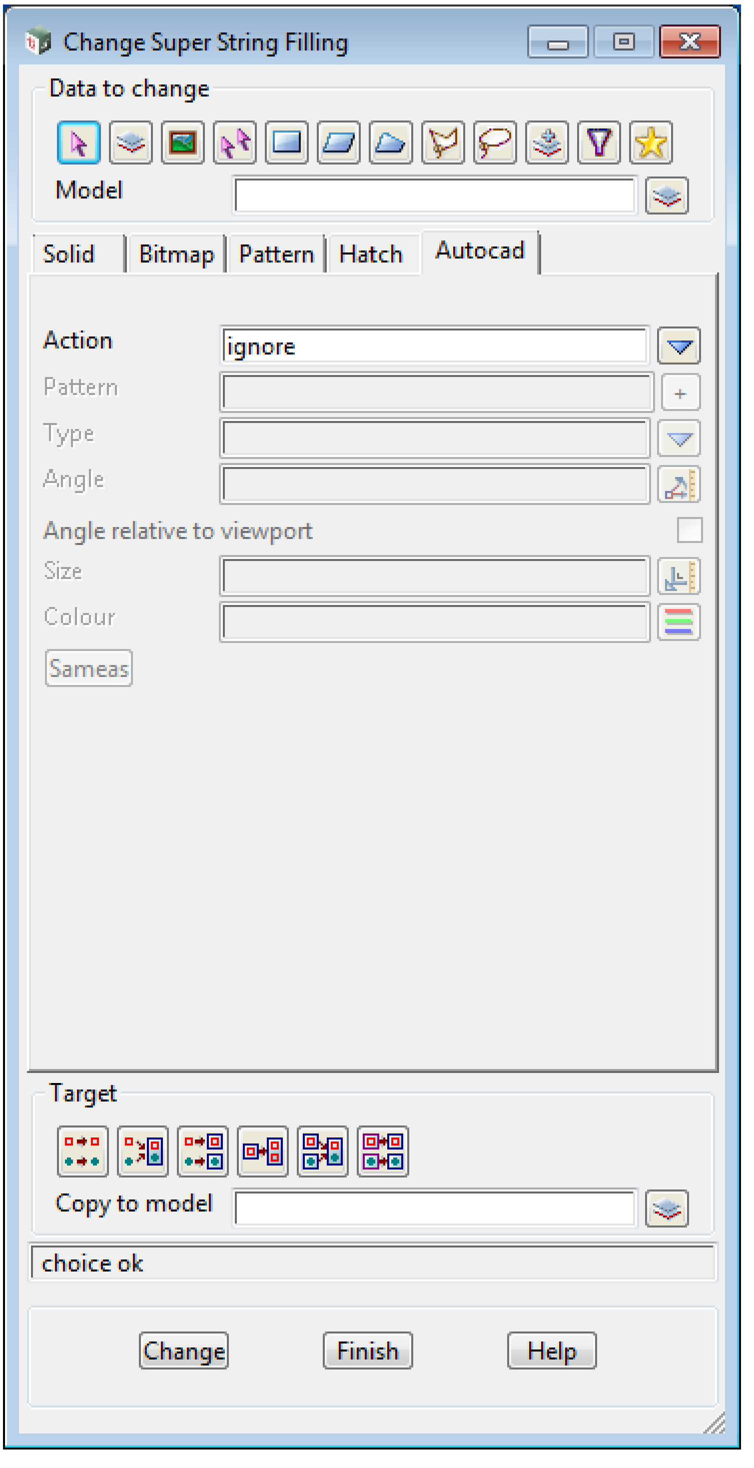
the first distance in the units selected in Type between the lines drawn at **Angle 2** *except it is the value used when you are plotting to or doing a plot of the super string containing the fill.*
if the field is blank it uses the **Spacing 2** *value.*

Colour 2 colour box **Select Colour** pop up

the colour of all the lines drawn at **Angle 2**.

Continue to 14.12.9.5.5 Autocad tab or return to 14.12.9.5 Fills.

14.12.9.5.5 Autocad tab



The fields and buttons used in the panel have the following functions.

Action	choice box	ignore	set, clear, ignore

*if **set**, this type of fill is set for the string.*
*If **clear**, this type of fill is turned off for the string.*
*If **ignore**, don't modify this type of fill for the string.*

Pattern **select autocad pattern** pop-up

The Autocad pattern name to be applied. These patterns are defined in the file acad.pat.

This file is an Autocad file.

Type choice box device, world, paper

the units of Size*. This means the bitmap can be drawn in either world size, paper size (mm) or screen (device) size (pixels).*

Angle angle

the orientation of the pattern relative to the x axis measured in the anti-clockwise direction in dms.

Angle relative to view tick box

this field controls whether the Angle is relative to the x axis or to the plotting x axis.

If ticked and we are plotting, **Angle** *is measured relative to the x axis of the plot rotation.*

If not ticked, **Angle** *is always absolute to the world x axis.*

Size real box

the size of the pattern in the selected type.

Colour colour box **Select Colour** pop up

the colour of the pattern.

Return to 14.12.9.5 Fills.

14.12.9.6 Symbol

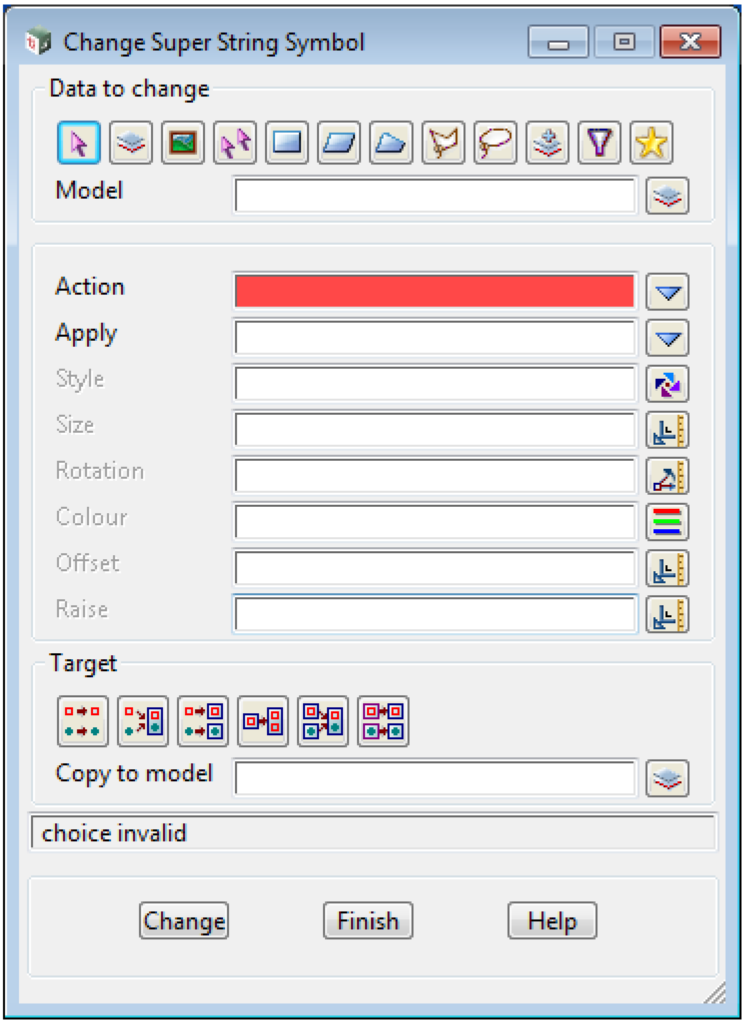
Position of option on menu: **Strings =>Properties =>Super string dimensions =>Symbol**

Position of option on menu: **Utilities =>Super strings =>Symbol**

For selected super strings, the **symbols** option can set the flags that determine how symbols are defined for a super string.

A super string can have no symbols, a constant symbol definition for the entire string or different symbol definitions at each vertex.

On selecting the **symbol** option, the **Change Super String Symbols** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Action	choice box	ignore	set, clear, ignore

*if **set**, a symbol can be constant for all vertices in the string, or can be different for each vertex in the string.*
*If **clear**, there are no symbols for the string.*
*If **ignore**, don't modify the symbol flags for the vertices of the string.*

Apply choice box constant, variable

*if **Action** is **set**, then the Apply field is used.*
*If Apply is **constant**, the string has only one symbol and it used for each vertices in the string. It is set to visible.*
*If Apply is **variable**, each vertex in the string has its own symbol.*

Symbol symbol box

symbol to be used.

Size

size of the symbol.

Rotation angle box

angle to rotation the symbol - positive is counter-clockwise.

Colour colour box

colour for the symbol (if the symbol has no imbedded colours).

Offset

distance the centre of the symbol is from the vertex.

Raise

distance the centre of the symbol is raised off the line through the vertex with the given angle.

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target.

Target info input

extra information required for the target.

Change button

process the selected strings

14.12.9.7 Segment Colour

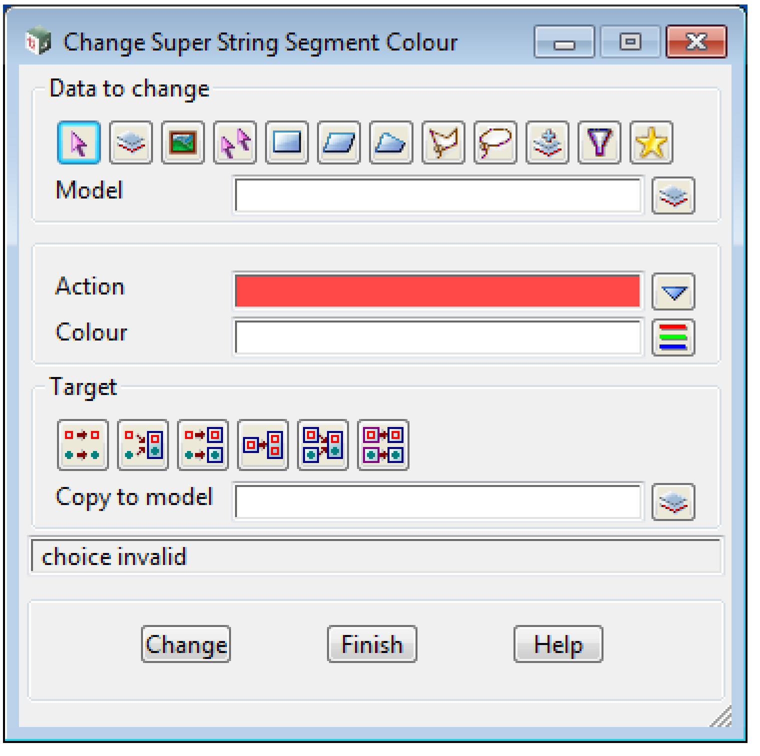
Position of option on menu: **Strings =>Properties =>Super string dimensions =>Segment colour**

Position of option on menu: **Utilities =>Super strings =>Segment colour**

A super string has a default colour used for the entire string and segment colours setable for each segment which override the string colour for that segment.

The **Segment colour** option can set a segment colour for every segment in the super string or clear all the set segment colours so that the default string colour is used.

On selecting the **Segment colour** option, the **Change Super String Segment Colour** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Action	choice box		set, clear

*if **set**, each segment colour is set to the colour in the* Colour *box.*
*If **clear**, any segment colours are removed and each segment set to having no segment colour.*

Colour	colour box		

*if Action is **set**, then every segment is set to this colour.*

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target.

Target info	input		

extra information required for the target.

Change button

process the selected strings

14.12.9.8 Height

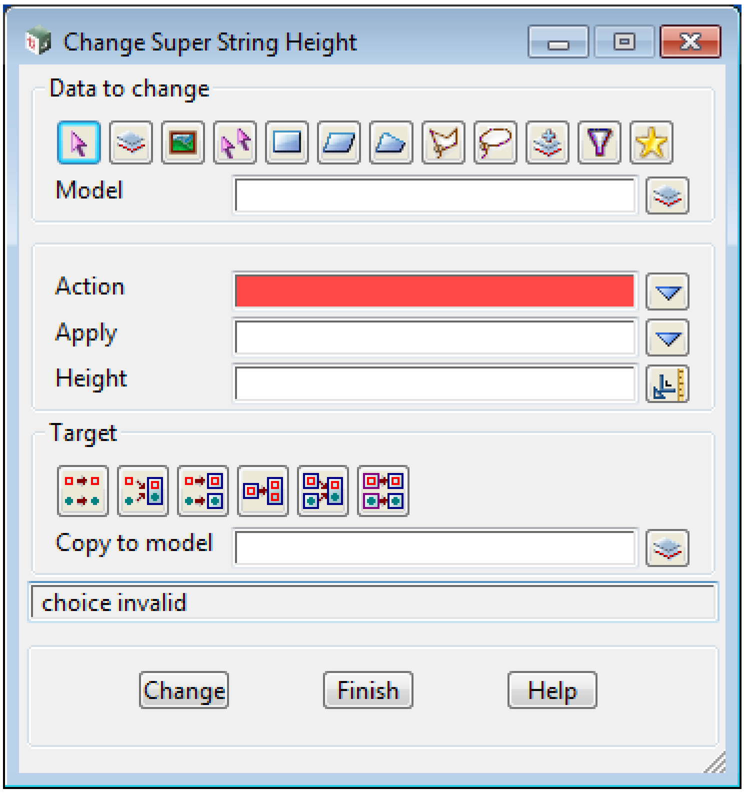
Position of option on menu: **Strings =>Properties =>Super string dimensions =>Height**

Position of option on menu: **Utilities =>Super strings =>Height**

A super string can have the one height that is used for every vertex in the super string (constant height) or it can have a different height at each vertex.

The **Height** option can set the super string to either having a constant height or to allow each vertex to have independent heights.

On selecting the **Height** option, the **Change Super String Height** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Action	choice box	ignore	set, clear

if ***set****, the string height can be a constant for the entire string, or to be different for each vertex in the string.*
If ***clear****, the entire string and each vertex has no height.*

Apply	choice box		constant, variable

if ***Action*** *is* ***set****, then the Apply field is used.*
If Apply is ***constant****, the string has only one height and no vertex has its own height. The height for the string is set to the height in the* Height *field.*
If Apply is ***variable****, each vertex in the string is set to have has its own height and it is set to the value in the* Height *field.*

Height

*if Action is **set**, then either the entire string or every vertex is set to this height.*

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target.

Target info input

extra information required for the target.

Change button

process the selected strings

14.12.9.9 Attribute

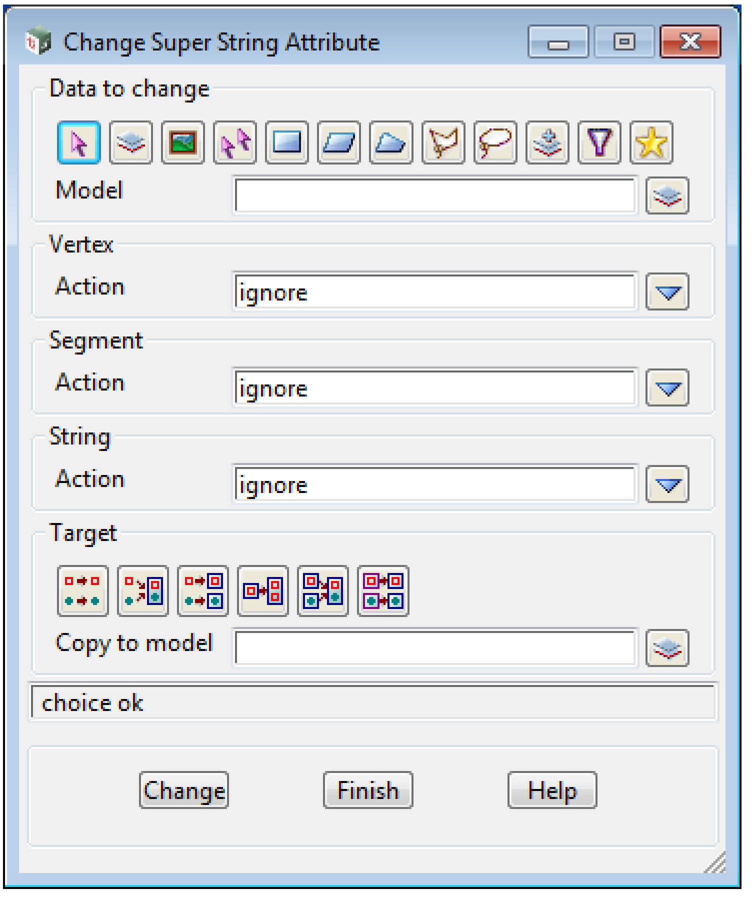
Position of option on menu: **Strings =>Properties =>Super string dimensions =>Attribute**

Position of option on menu: **Utilities =>Super strings =>Attribute**

A super string can have independent attributes for the entire string, independent attributes for each segment and for each vertex.

The **Attributes** option can clear the super string attributes for the entire string, each vertex and/or each segment.

On selecting the **Attributes** option, the **Change Super String Attributes** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Vertex Action	choice box	ignore	ignore, clear

*if **clear**, the attributes are cleared at each vertex.*
*If **ignore**, nothing is done for the vertex attributes.*

Segment Action	choice box	ignore	ignore, clear

*if **clear**, the attributes are cleared on each segment.*
*If **ignore**, nothing is done for the segment attributes.*

String Action	choice box	ignore	ignore, clear

*if **clear**, the string attributes are cleared.*

*If **ignore**, nothing is done for the string attributes.*

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target.

Target info input

extra information required for the target.

Change button

process the selected strings.

14.12.9.10 Vertex ID

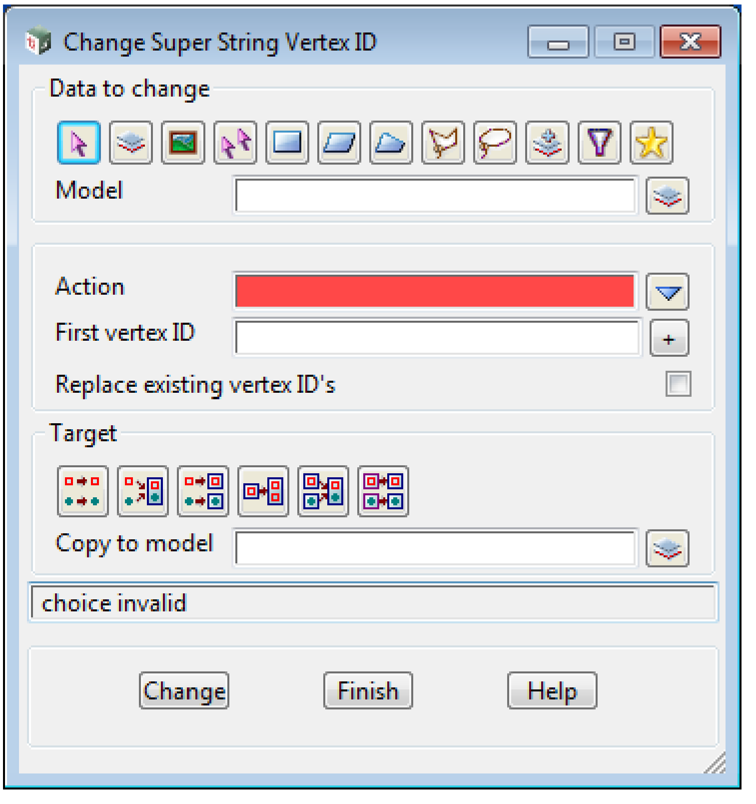
Position of option on menu: **Strings =>Properties =>Super string dimensions =>Vertex id**

Position of option on menu: **Utilities =>Super strings =>Vertex id**

A super string has a *Vertex id* for each vertex.

The **Vertex id** option can set the Vertex IDs to sequential numbers (optionally ignoring existing Vertex ids) or clear all the existing vertex IDs for the super string.

On selecting the **Vertex id** option, the **Change Super String Vertex ID** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Action	choice box		set, clear

*if **clear**, the vertex IDs are cleared at each vertex.*
*If **set**, the vertex IDs are given sequential numbers starting with the number given in* First vertex id *field.*

First vertex id

*if Action is **set**, the vertex IDs start with this value.*

Replace existing vertex id's tick box

*if **ticked**, any vertex with an existing vertex is not given a new vertex ID.*
*If **not ticked**, all vertices are given new vertex IDs.*

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target.

Target info input

extra information required for the target.

Change button

process the selected strings.

14.12.9.11 Pipe/Culvert

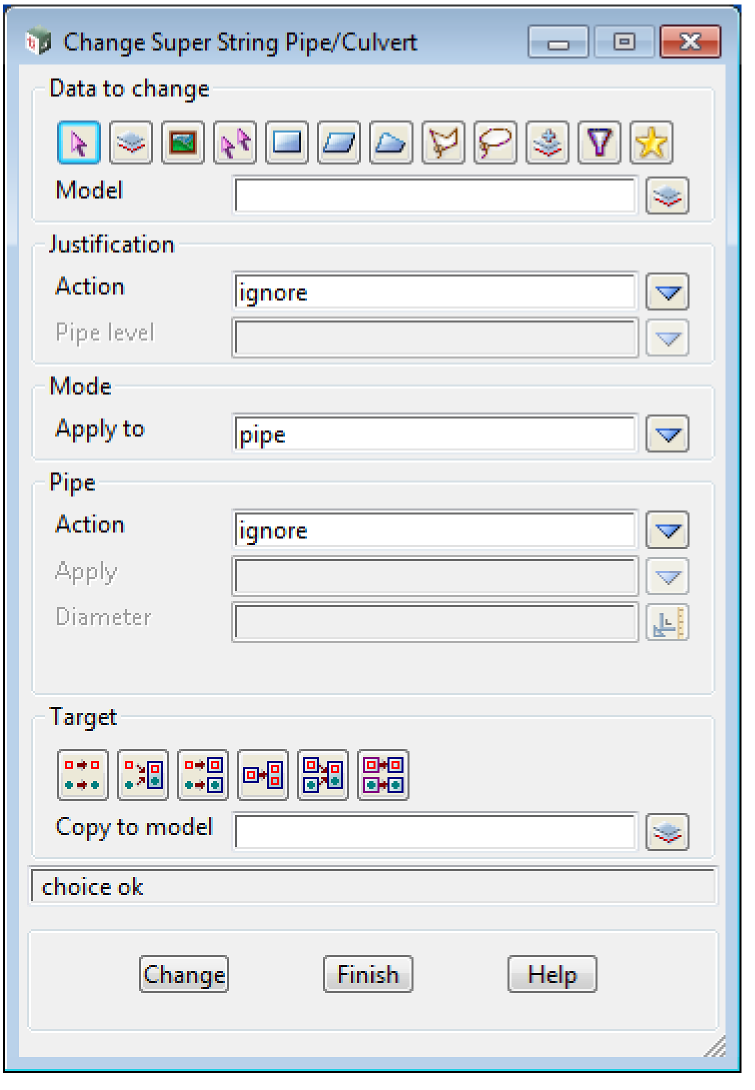
Position of option on menu: **Strings =>Properties =>Super string dimensions =>Pipe/culvert**

Position of option on menu: **Utilities =>Super strings =>Pipe/culvert**

Each segment of a super string can have a diameter or a rectangular cross-section (box culvert).

This option can define or clear the pipe diameter, width and height, and justification.

On selecting the **Pipe/culvert** option, the **Change Super String Pipe/Culvert** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Justification

Action	choice box		set, clear, ignore

*if **set**, the string justification is set to the type given in the* Pipe level *field.*
*If **clear**, the string justification information is cleared.*
*If **ignore**, nothing is done to the string justification.*

Pipe level

*if Action is **set**, the string justification is set to this value.*

Mode

Apply to choice box pipe, culvert

*if **pipe**, the next section of the panel has fields for setting pipe information.*
*if **culvert**, the next section of the panel has fields for setting culvert information.*

Pipe - when Mode is Pipe

Action choice box set, clear, ignore

*if **set**, then the* Pipe Apply *field specifies how the pipe information is modified.*
*If **clear**, the pipe information is cleared at each segment.*
*If **ignore**, nothing is done to the pipe information.*

Apply choice box constant, variable

*if **Pipe Action** is **set**, then the Apply field is used.*
*If Apply is **constant**, the string has only one diameter and no segment has its own diameter. The diameter for the string is set to the value in the* Diameter *field.*
*If Apply is **variable**, each segment in the string is set to have has its own diameter and it is set to the value in the* Diameter *field.*

Diameter

diameter for each pipe segment.

Culvert - when Mode is Culvert

Action choice box set, clear, ignore

*if **set**, then the* Culvert Apply *field specifies how the culvert information is modified.*
*If **clear**, the culvert information is cleared at each segment.*
*If **ignore**, nothing is done to the culvert information.*

Apply choice box constant, variable

*if **Culvert Action** is **set**, then the Apply field is used.*
*If Apply is **constant**, the string has only one width and height and no segment has its own width and height. The width and height for the string is set to the values in the* Width *and* Height *fields.*
*If Apply is **variable**, each segment in the string is set to have has its own width and height and they are set to the values in the* Width *and* Height *fields.*

Width/Height

width/height for each culvert segment.

Target type

data selection type - for a full description go to 4.19.3 Data Source.

Target info input

extra information required for the target.

Change button

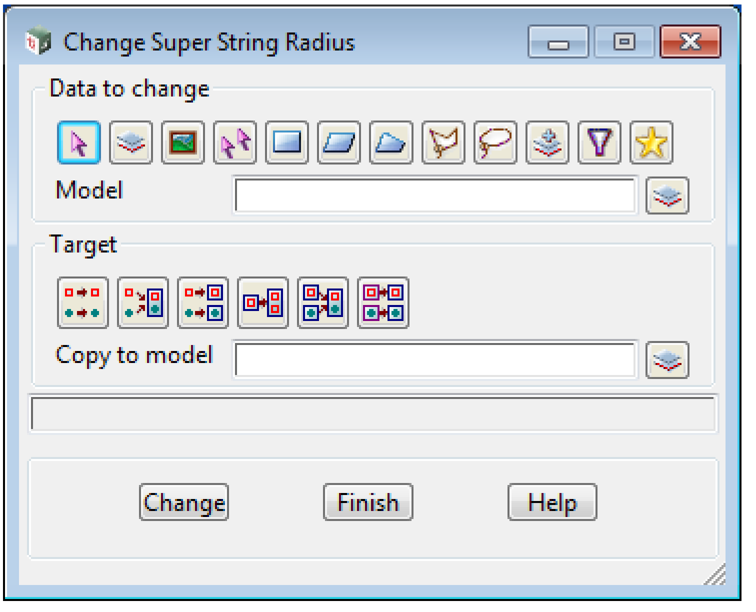
process the selected strings.

14.12.9.12 Radius

Position of option on menu: **Strings =>Properties =>Super string dimensions =>Radius**

Position of option on menu: **Utilities =>Super strings =>Radius**

The **Radius** option clears any segment radii.

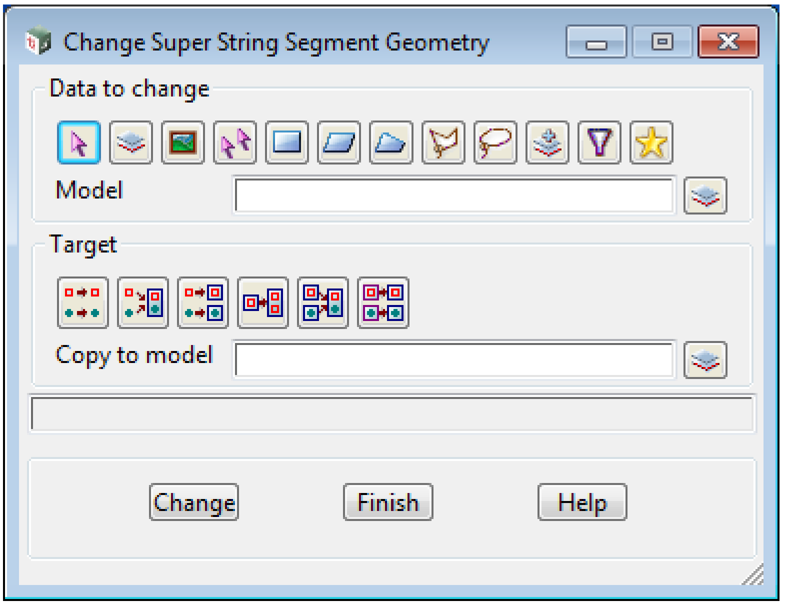


14.12.9.13 Segment Geometry

Position of option on menu: **Strings =>Properties =>Super string dimensions =>Segment geometry**

Position of option on menu: **Utilities =>Super strings =>Segment geometry**

The **Segment geometry** option clears any segment geometry.



14.12.9.14 Text

Position of option on menu: **Strings =>Properties =>Super string dimensions =>Text**

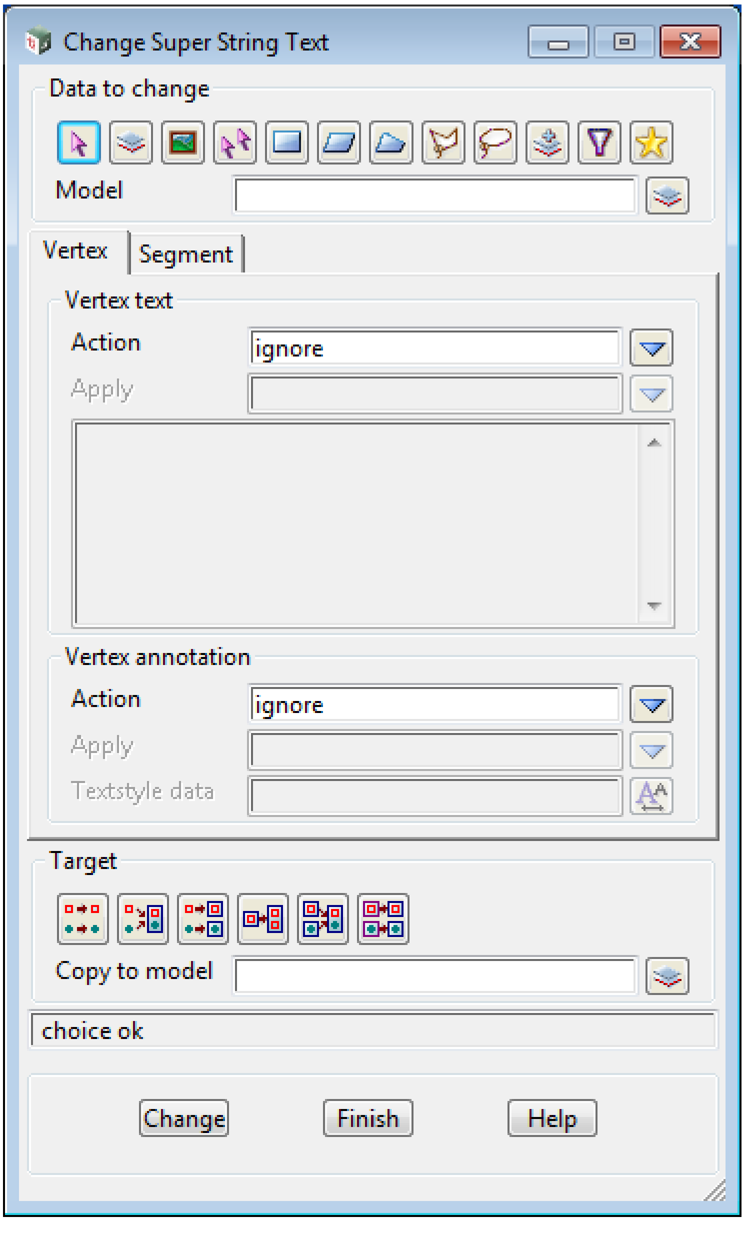
Position of option on menu: **Utilities =>Super strings =>Text**

This section of documentation is a work in progress and will be updated in subsequent releases.

Each vertex and segment of a super string can have its own text.

The **Text** option can set, clear the vertex or segment text and/or textstyle data for the super string.

On selecting the **Text** option, the **Change Super String Text** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Model	model box		available models

Vertex tab

Vertex text

Action	choice box	ignore	set, clear, ignore

*if **set**, the vertex text can be a constant for the entire string, or to be different for each vertex in the string.*
*If **clear**, the vertex text is cleared.*
*If **ignore**, nothing is done to the vertex text.*

Apply	choice box		constant, variable

*if **Action** is **set**, then the Apply field is used.*
*If Apply is **constant**, the string has only one vertex text value and no vertex has its own vertex text. The vertex text for the string is set to the text in the* Text *field.*
*If Apply is **variable**, each vertex in the string is set to have has its own text and it is set to the value in the* Text *field.*

Text

*if Action is **set**, then either the entire string or every vertex text is set to this value.*

Vertex annotation

Action	choice box	ignore	set, clear, ignore

*if **set**, the vertex textstyle data can be a constant for the entire string, or to be different for each vertex in the string.*
*If **clear**, the vertex textstyle data is cleared.*
*If **ignore**, nothing is done to the vertex textstyle data.*

Apply	choice box		constant, variable

*if **Action** is **set**, then the Apply field is used.*
*If Apply is **constant**, the string has only one vertex textstyle data and no vertex has its own vertex textstyle data. The vertex textstyle data for the string is set to the text in the* Textstyle data *field.*
*If Apply is **variable**, each vertex in the string is set to have has its own textstyle data and it is set to the value in the* Textstyle data *field.*

Textstyle data

*if Action is **set**, then the vertex textstyle data is set to this value.*

Segment tab

Segment text

Action

Apply

Segment annotation

Action

Apply

Textstyle data

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target.

Target info	input

extra information required for the target.

Change button

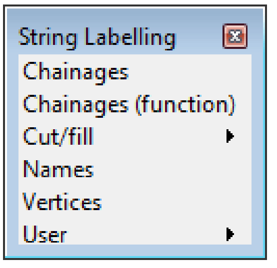
process the selected strings.

14.13 Label

Position of menu: **Strings =>Label**

The **Label** menu contains options to label string chainages, create tadpoles for cut and fill, label string vertices with their x, y, z or point numbers and names.

The **Label** walk-right menu is



For the option *Chainages*, go to 14.13.1 Label Chainages
Chainages (function) 14.13.2 Chainages (function)
Cut/fill 14.13.3 Label Cut/Fill
Names 14.13.4 Label Names
Vertices 14.13.5 Label Vertices

14.13.1 Label Chainages

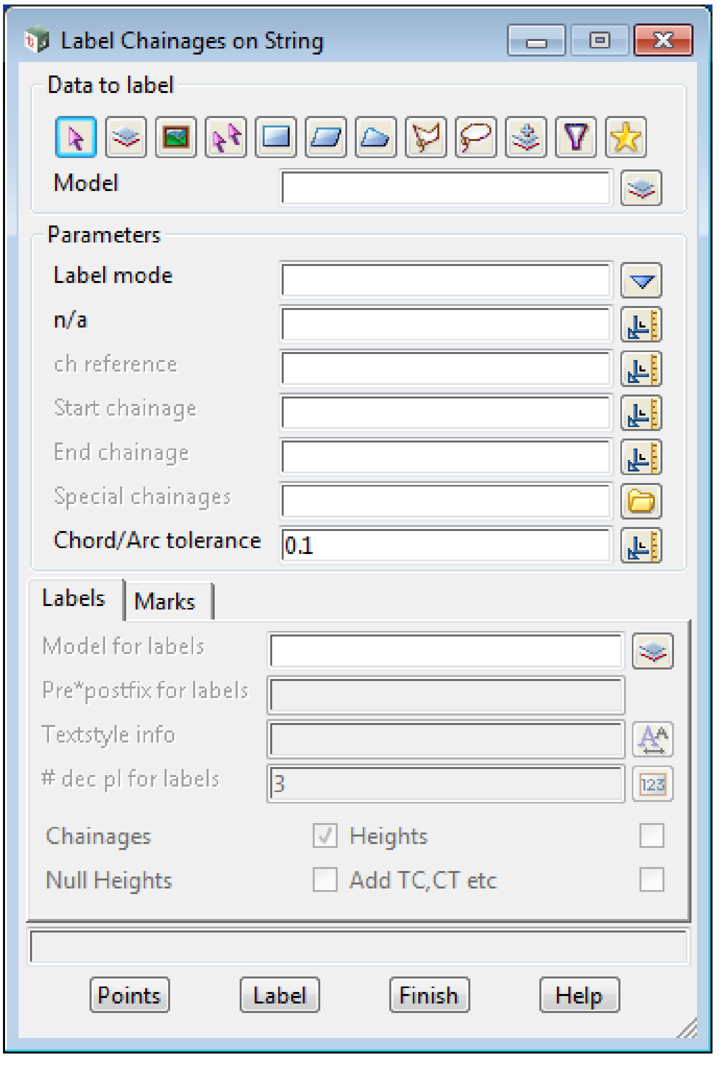
Position of option on menu: **Strings =>Label => Chainages**

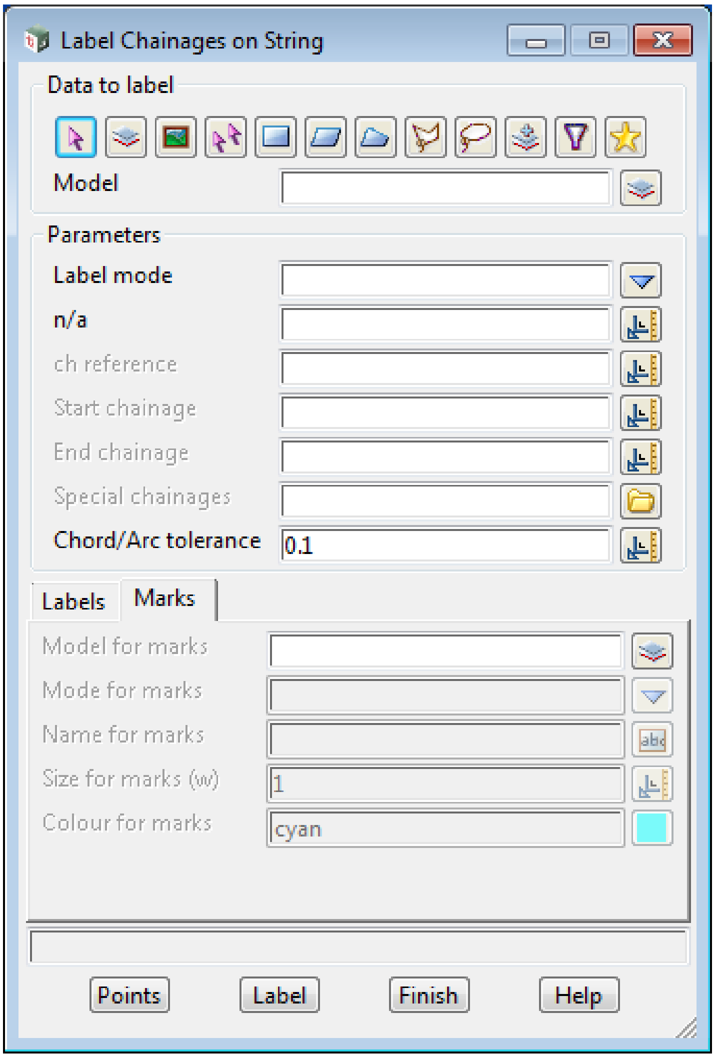
The **Label chainages** option is used to create text at regular chainages and special points on a string such as horizontal and vertical tangent points, crest and sag points and user selected points. It can also create tick marks at selected chainages.

The user has control over the label height, colour, angle, offset and the number of decimal places displayed, plus the size and colour of the tick marks.

As well as being used for labelling centre-lines, this option is used in conjunction with the sewer option to created special labels for sewer long section plots.

On selecting the **Label chainages** option, the **Label chainages on string** panel is displayed.





Individual points can be labelled by selecting them after picking the **Points** button.

The **marks** button brings up the **label chainages on string (marks)** panel which is used to place marks at the chosen chainages.

The fields and buttons used in the **label chainages on string** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type *data source type.*			
Data source *data source for strings to label.*			
Label mode	input		regular interval, regular interval plus end pts end points only horizontal TPs, vertical TPs, horizontal discontinuities vertical discontinuities all discontinuities, crests/sags

all horizontal points

type of labelling required.

Ch interval or **n/a** input

the regular interval to use for labelling points.

Ch reference input 0

the chainages to be labelled are integer multiples of the chainage interval added to the reference chainage. For example, if the reference chainage is 23.2 and the chainage interval 10, the chainages 3.2, 13.2, 23.2, 33.2 etc. will be labelled.

Start chainage input

if non-blank, the string chainage to start labelling from.
If blank, start at the beginning of the string.

End chainage input

if non-blank, the string chainage to end the labelling at.
If blank, go to the end of the selected string.

Special chainage input

file of special chainages to create labels at.

Chord/arc tolerance input default chord/arc tolerance

the chord to arc tolerance to use on the strings being labelled for determining how many points are labelled around horizontal curves.

Labels tab

Model for labels model box available models

*if **non-blank**, labels are generated and placed in this model.*
*If **blank**, no labels are created (but tick marks may be).*

Pre*postfix for labels input

*the beginning and ending of the label to be given at each point. Spaces are significant. This uses the standard 12d method of pre-text*post-text. That is, 'Ch * m' would add 'Ch ' before the value and ' m' after the label.*

Textstyle info textstyle box 1 available textstyle data

textstyle data to use when creating the labels.

 # dec pl for labels input 0

number of decimal places used in the labels.

Chainages tick tick

*if **ticked**, the chainage of each point is appended to the label stem to create the label for the point.*
*If **not ticked**, only the label stem is used as the label.*

Heights tick

*if **ticked**, the height of each point is used in the label for the point.*
*If **not ticked**, heights are not used in the label.*

Null heights tick

*if **ticked** and the height of a point null, **(null)** is used in the label for the point.*
*If **not ticked**, null heights are not used in the label.*

Add TC, CT, *etc.* tick

*if **ticked** and labelling an alignment string, the critical point types are used in the label for the point.*
*If **not ticked**, critical point types are not used in the label.*

Marks tab

Model for marks	input		available models

*if **non-blank**, tick marks are generated according to the* label mode *and placed in this model. If **blank**, no tick marks are created.*

Mode for marks	input	ticks centred	ticks on lhs, ticks on rhs, ticks centred

mode for the tick marks.

Name for marks	input	1	

name to give the tick marks (mainly used for mapping files)

Size for marks (w)	input	1	

size in world units for the tick marks

Colour for marks	input	orange	available colours

colour of the tick marks.

Points	button		

pick individual positions to label with a label stem and chainage (if the chainages *field is set to* ***tick****).*

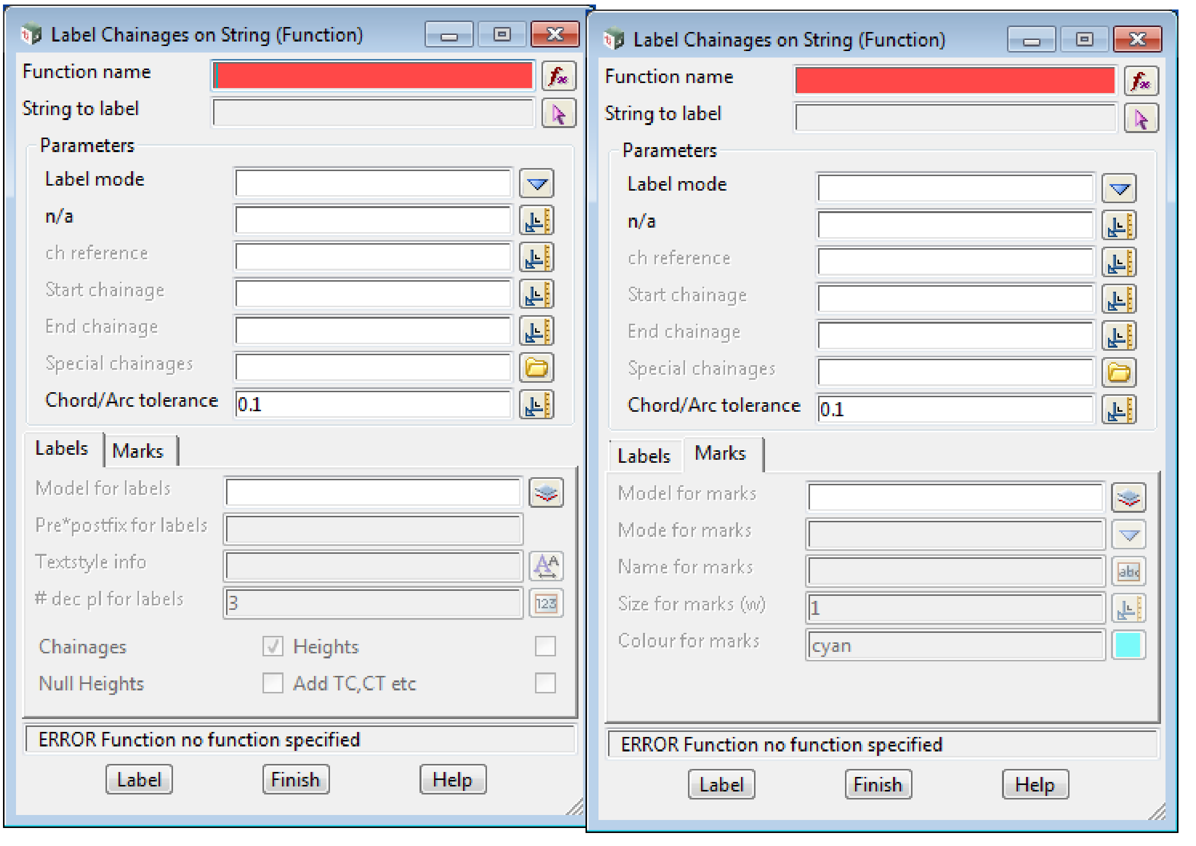
Label	button		

label the selected strings as specified by the label mode, mode for marks *and other fields in the panel.*

14.13.2 Chainages (function)

Position of option on menu: **Strings =>Label => Chainages (function)**

The parameters for this panel are identical to **Label Chainages on String,** however instead of selecting data source for labels this option will create a function associated with a specified string.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function name	function box		select function

The name of the function to run.

String to label	string select		

The string to label.

For more information on this panel and its fields and buttons, please go to 14.13.1 Label Chainages.

Please continue to the next section 14.13.3 Label Cut/Fill.

14.13.3 Label Cut/Fill

Position of menu: **Strings =>Label => Cut/Fill**

The **cut/fill** menu contains options to create cut and fill tick marks between selected strings.

The options can generate standard tadpoles, tick marks or use user specified model as the tick symbol.

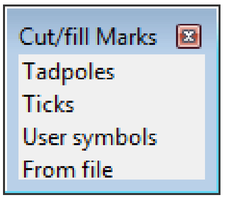
For each option, a reference string is selected which defines the chainages used for labelling cut/fill. The cut/fill symbols are drawn perpendicular to the reference string.

Two strings, str1 and str2 are selected for labelling.

The tick symbols are drawn at right angles to the reference string at a user specified chainage going between the two strings, str1 and str2, from the higher string point to the lower string point (which string is higher or lower may vary along the strings). The tick symbol is repeated at the given chainage interval.

The tick symbol is drawn as a percentage of the distance from the high point to the low point between the two strings, str1 and str2 and this percentage is specified separately for the odd and even numbered ticks.

The **cut/fill** walk-right menu is



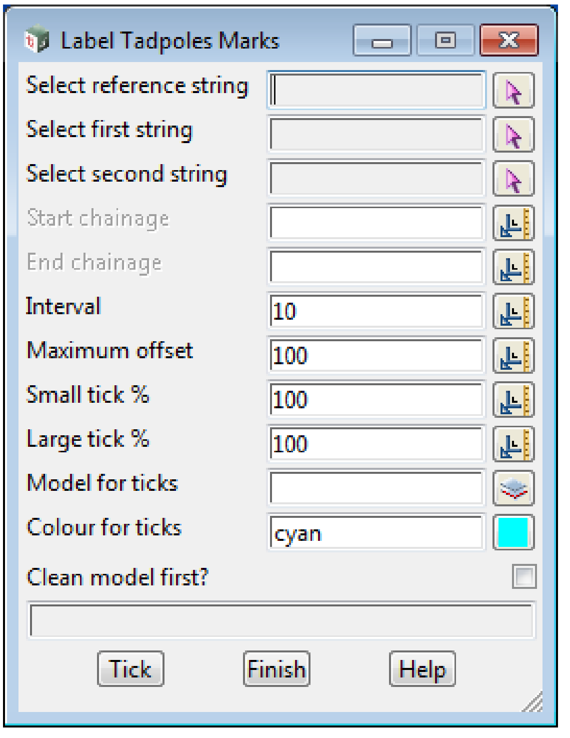
For the option *Tadpoles*, go to 14.13.3.1 Tadpoles
Ticks 14.13.3.2 Ticks
User symbols 14.13.3.3 User Symbols
From file 14.13.3.4 From File

14.13.3.1 Tadpoles

Position of option on menu: **Strings =>Label => Cut/Fill => Tadpoles**

The **Tadpoles** option is used to generate tadpole symbols between two strings.

On selecting the **Tadpoles** option, the **Label tadpoles marks** panel is displayed.



The fields and buttons in this panel are used as follows

Field Description	Type	Defaults	Pop-Up
Start chainage	input		

*if **non-blank**, the reference string chainage to start creating tadpoles from.*
*if **blank**, the reference string start chainage is used.*

End chainage	input		

*if **non-blank**, the reference string chainage to end the tadpoles at.*
*if **blank**, the reference string end chainage is used.*

Interval	input	10	

the chainage interval to use for creating tadpoles.

Maximum offset	input	100	

the maximum distance to search from the reference string to find strings str1 and str2.

Small tick%	input	100	

the percentage of the distance between the two strings, str1 and str2, that is taken up by the odd numbered tadpoles.

Large tick%	input	100	

the percentage of the distance between the two strings, str1 and str2, that is taken up by the even numbered tadpole.

Model for ticks	input		available models

the models for the tadpoles to placed into.

Colour for ticks	input	cyan	available colours

the colour for the tadpoles.

Ref/Str1/Str	button		

*the selected string is used as the **reference/str1/str2** string.*

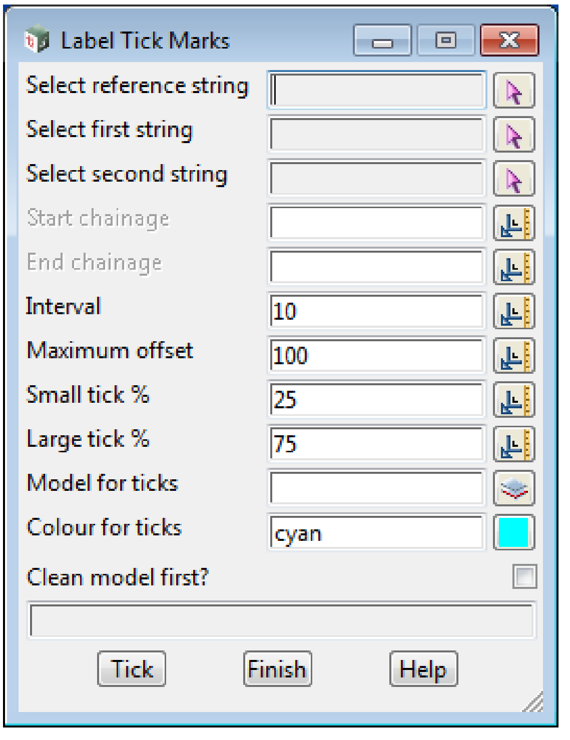
Continue to the next section 14.13.3.2 Ticks or return to 14.13.3 Label Cut/Fill.

14.13.3.2 Ticks

Position of option on menu: **Strings =>Label => Cut/Fill => Ticks**

The **ticks** option is used to generate ticks (straight lines) between two strings.

On selecting the **ticks** option, the **label tick marks** panel is displayed.



The fields and buttons in this panel are used in exactly the same way as for the **label tadpoles marks** panels. The only difference is that the defaults for the small tick % and large tick % panel fields are 25 and 75 respectively.

14.13.3.3 User Symbols

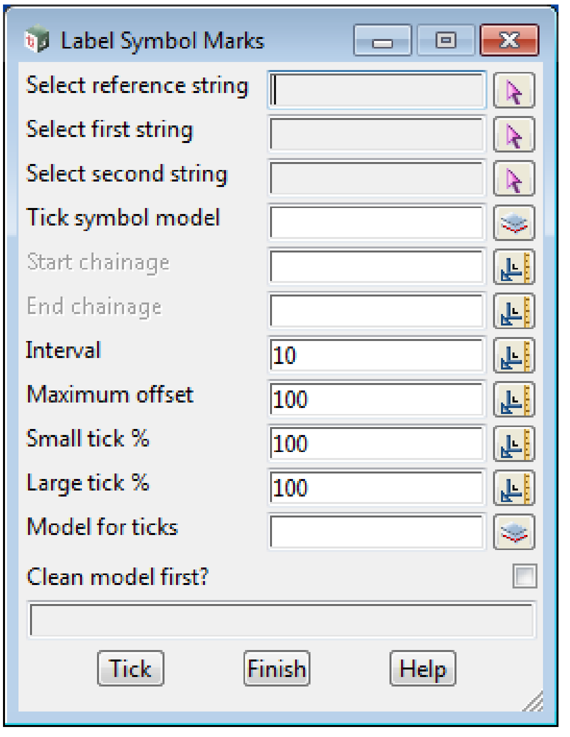
Position of option on menu: **Strings =>Label => Cut/Fill => User symbols**

The **user symbols** option uses a given model, the tick symbol model, as the symbol to drawn between the low and high points on the two strings.

The tick symbol model is aligned so that the model origin is at the high string point and the model's positive x-axis goes from the high string point to the low string point.

On selecting the **ticks** option, the **label tick marks** panel is displayed.

Continue to the next section 14.13.3.3 User Symbols or return to 14.13.3 Label Cut/Fill.



The fields and buttons in this panel are used in exactly the same way as for the **label tadpoles marks** panels except that a tick symbol model panel field is used to specify the symbol to be drawn for the cut/fill tick marks.

Field Description	Type	Defaults	Pop-Up
Tick symbol model	input		available models

the model to be used as the tick mark symbols.

Continue to the next section 14.13.3.4 From File or return to 14.13.3 Label Cut/Fill.

14.13.3.4 From File

Position of option on menu: **Strings =>Label => Cut/Fill => From file**

This is the same option as **Drafting =>Create cut/fill symbols**

Please go to the section 14.13.3 Label Cut/Fill.

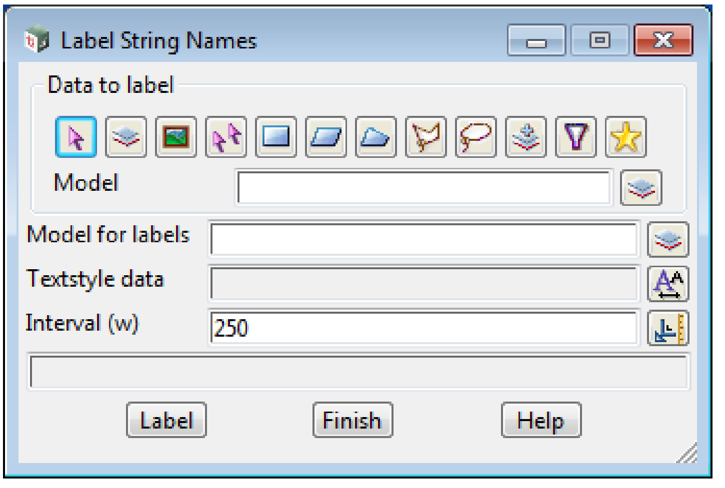
14.13.4 Label Names

Position of option on menu: **Strings => Label => Names**

The **label names** option is used to label individual strings or all the strings in a model, with their string names.

The user has control over the label height, colour, and the distance between the labels.

On selecting the **label names** option, the **label string names** panel is displayed.



By using the **pick** button, individual strings can be labelled using the values in the panel fields. The model to label field is ignored.

If the **label** button is selected, then all the strings in the model given by the model to label field will be labelled according to the parameters in the panel fields.

The fields and buttons used in the **label names** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type			
data source type.			
Data source			
data source for strings to label.			
Textstyle data	input		
textstyle information.			
Interval (w)	input	250	
chainage interval in world units between the labels.			
Label	button		
label all the strings in the model given in the model to label *field.*			

Please continue to the next section 14.13.5 Label Vertices.

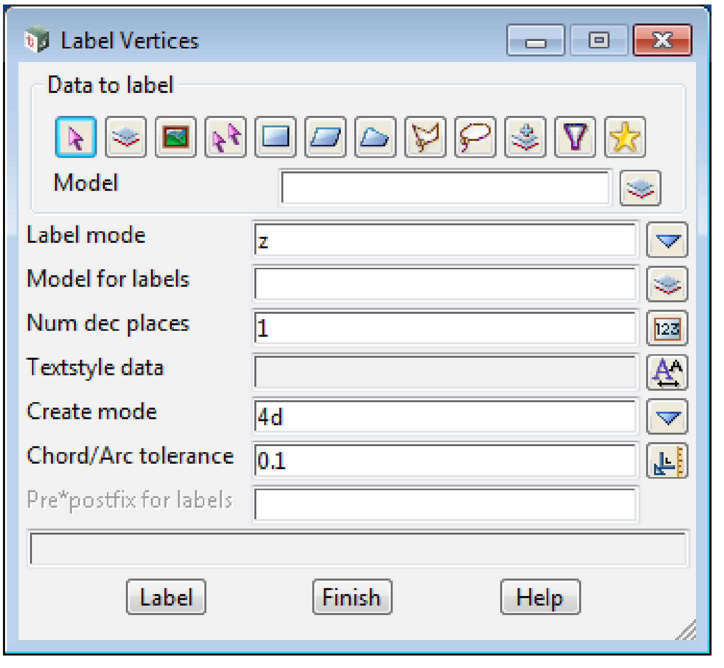
14.13.5 Label Vertices

Position of option on menu: **Strings =>Label => Vertices**

The **Label vertices** option can be used to label the x, y, z, z no nulls, point numbers, id-pt no, name and descriptions of strings selected by the data source.

The user has control over the label height, colour, the number of decimal places, the distance between the labels, the distance the label is from the point position, and the angle to draw the label at.

On selecting the **Label vertices** option, the **Label vertices** panel is displayed.



Individual strings can be labelled by using the **pick** button and then selecting the strings, to be labelled using the values in the panel fields. The string model field is ignored.

If the **label** button is selected, then all the strings in the model given by the string model field will be labelled according to the parameters in the panel fields.

The fields and buttons used in the label strings panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type			

data source type.

Field Description	Type	Defaults	Pop-Up
Data source			

data source for strings to label.

Field Description	Type	Defaults	Pop-Up
Label mode	input	z	x,y,z, point number, Vertex no x and y id-vertex no z no nulls name

type of labelling required - x, y, z, point number, vertex no. for each point in the string.

Field Description	Type	Defaults	Pop-Up
String model	input		available models

name of the model containing the strings to be labelled. Not used with the pick button.

Model for labels	input		available models

name of the model to place the labels in.

Num dec places	input	0	0,1,2,3,4,5

number of decimal places used in the x,y,z labels.

Textstyle data	input		

textstyle information.

Create mode	input	4d	4d, text

*if **4d**, the vertex labels for the one string are placed in a 4d string.*
***text**, the vertex labels are created as individual text strings.*

Chord/arc tolerance	input	default chord/arc tolerance

the chord to arc tolerance to use on any alignment string being labelled for determining how many points are labelled around horizontal curves.

Label	button

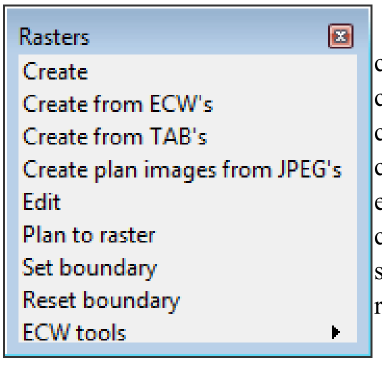
label all the strings in the model given in the string model *field.*

14.14 Rasters

Position of menu: **Strings =>Rasters**

The **Rasters** menu contains options for working with rasters.

The **Rasters** walk-right menu is



create a raster
create a raster from an ECW file
create a raster from an TAB file
create plan images from JPEGs
edit a raster
convert plan view to a raster
set a boundary for a raster
remove the boundary from a raster

For the option *Create* go to	14.14.1 Create a Raster
Create from ECW's	14.14.2 Create Rasters from ECW Files
Create from TAB's	14.14.3 Create Rasters from TAB Files
Create plan images from JPEG's	14.14.4 Create Plan Images from JPEG Files
Edit	14.14.5 Edit a Raster
Plan to raster	9.1.4.7 Create Raster from Plan View
Set boundary	14.14.6 Set a Boundary for a Raster
Reset boundary	14.14.7 Reset the Boundary for a Raster
ECW tools	14.14.8 ECW Tools - Not Supported!

14.14.1 Create a Raster

Position of option on menu: **Strings =>Rasters =>Create**

A raster element consists of a raster image and positioning information for the raster so that it can be mapped into world units.

A raster element is added to a model just like any other string but for convenience it is suggested that no other strings are in the same model as a raster.

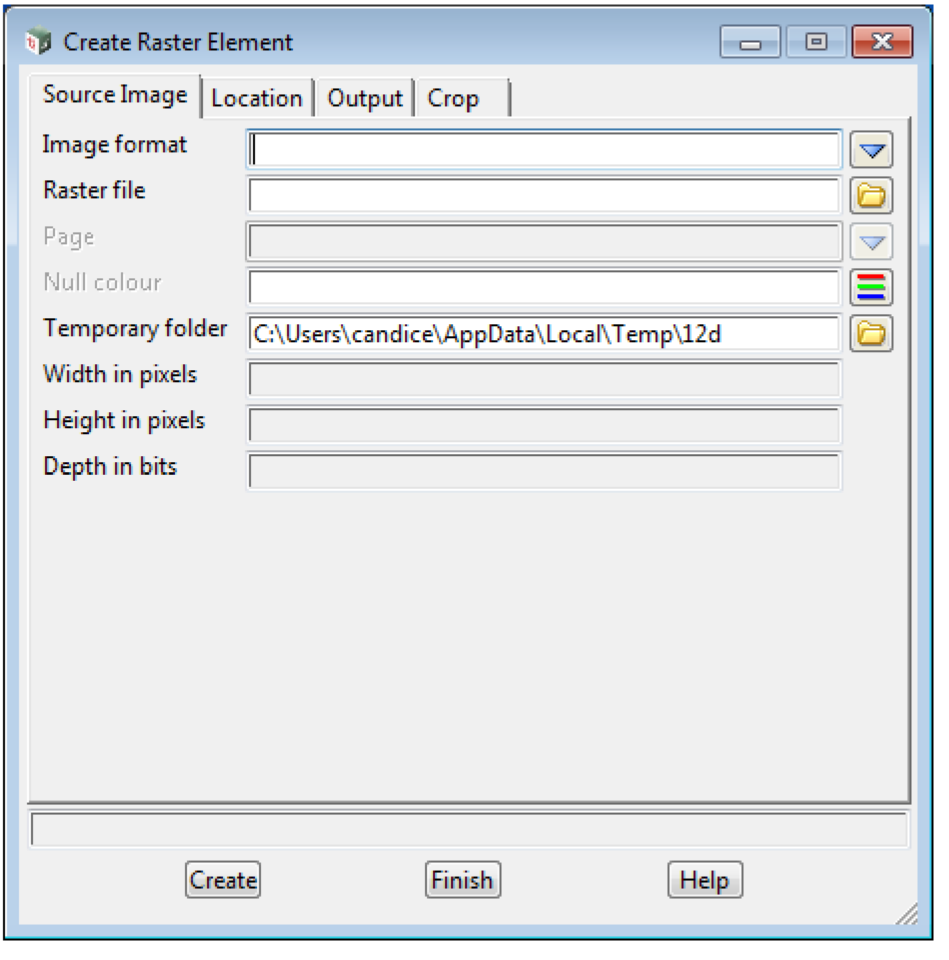
To help speed up drawing on a view, there is toggle for displaying/not displaying rasters on plan views.

Raster Restrictions

(a) The Raster file must have read/write permission

(b) Rasters are only supported on Win 2000 and above.

Selecting **Create** brings up the **Create Raster Element** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Source Image tab			
Image format	choice box		BMP, DIB, GIF ECW, JPEG, JPEG 2000 PNG, TGA, TIFF

format that the original image is in. Non ECW rasters must be 24 bit colour.

Raster file	file box		

name of the original raster image. The file must have read/write access and be 24 bit colour.

Page

page number of the raster when the format can contain more than one image (e.g. in a TIFF)

Null colour colour box available colours

colour when there is no pixel in the raster

Temporary folder file box c:\temp select folder

folder to use as a temporary storage area whilst converting the raster to the 12d raster format. The temporary folder needs to have enough free disk space to convert the largest of the rasters to bmp format.

Width/Height in pixels output only

the width/height in pixels of the selected raster

Depth in bits output only

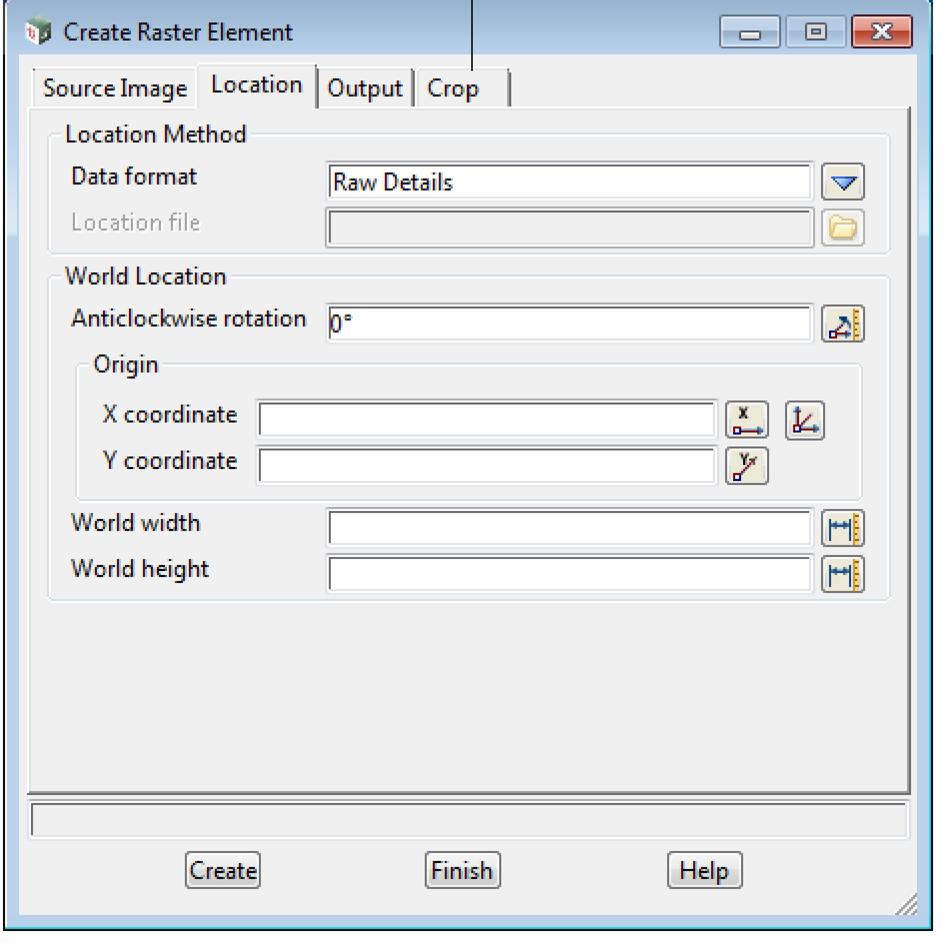
display the colour depth of the source data

Projection output only

display the projection of the source data if available

Location tab

*The Location tab supplies the information for positioning the raster in world units in **12d Model**. Most rasters formats, other than ECW or Geotiff, do not include this information so it must be supplied by other means, either in a file or by typing in a world origin, anticlockwise rotation angle and world width and height.*



Data format	choice box	Raw details	Raw Details, Autocad scr file ESRI world, Geo Tiff, Mapinfo tab

if ***Raw details****, the location details are typed into the* World Location Details *section as rotation, world origin, world width and height.*
If **Autocad scr file***, the location details are taken from the scr file given in* Location file*. An Autocad scr file gives the world co-ordinates of the corners of the raster starting in the bottom left hand corner, bottom right hand corner, top left hand corner and top right hand corner. An example of a scr file is:*

line
22109.639,148090.695
23109.639,148090.695
22109.639,149090.695
23109.639,149090.695

If **ESRI world***, the location details are taken from the ESCRI world file given in* Location file*. An ESRI world file gives the xscale, row rotation, column rotation, yscale, x origin and y origin. For use with 12d, the row and column rotations must be the same and yscale = - xscale.*
If **Geo Tif**f*, the Location details are taken from the geotiff file given in* Location file *panel field.*
If **Mapinfo Tab***, the location details are taken from the Mapinfo tab file given in* Location file.

Location file file box

file with the location details

Anticlockwise rotation angle box

the world rotation of the raster.

X/Y co-ordinate real box

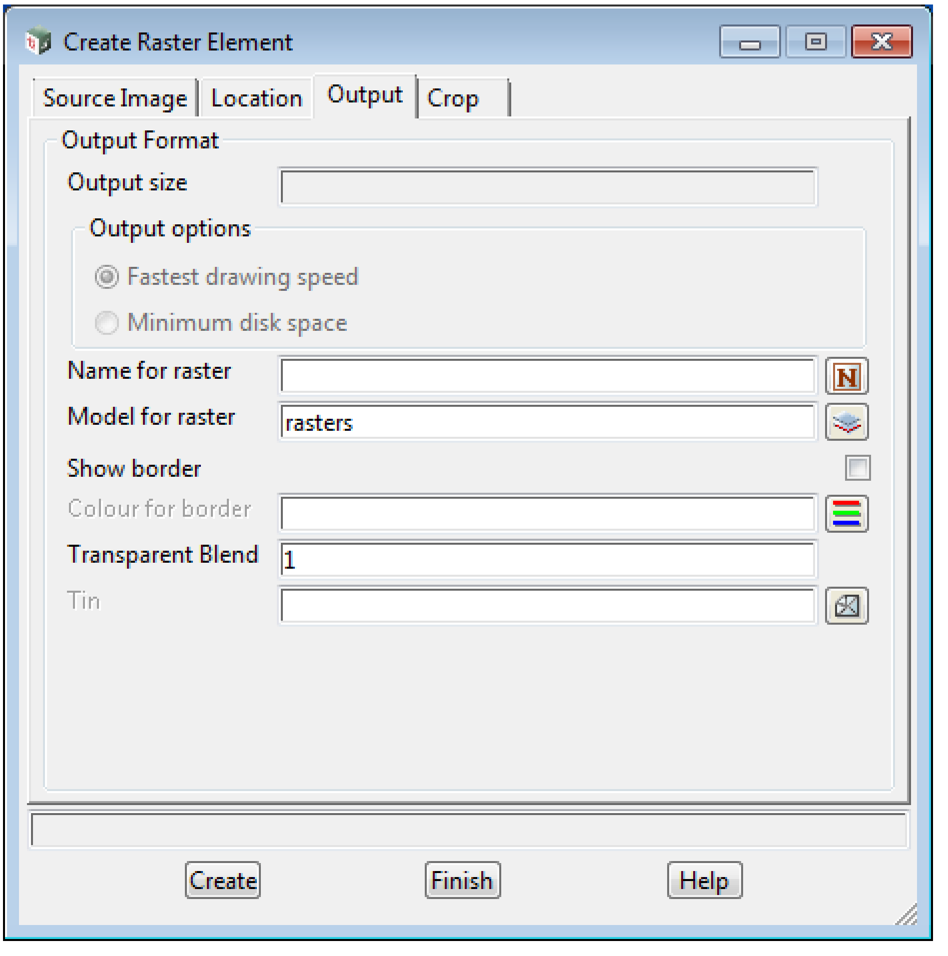
the world x/y co-ordinate of the left hand bottom corner of the raster.

World width/height real box

the width/height in world units of the raster.

Output tab

the parameters for the created 12d raster.



Output size output

an estimate of the amount of disk space the raster will use in ***12d Model***

Fastest drawing speed input

only if raster is less than 24 bit depth - if selected, the raster is converted to a 24 bit image.

Minimum disk space input

only if raster is less than 24 bit depth - if selected, the raster is not converted to a 24 bit image and hence will take up less disk space. However it is converted at drawing time and so will be slower to display than a 24 bit image.

Name for raster input

the name for the raster

Model for raster input available models

name of the model for the raster element

Show border tick tick

if tick then the border of the raster element is displayed

Colour for border input default colour available colours

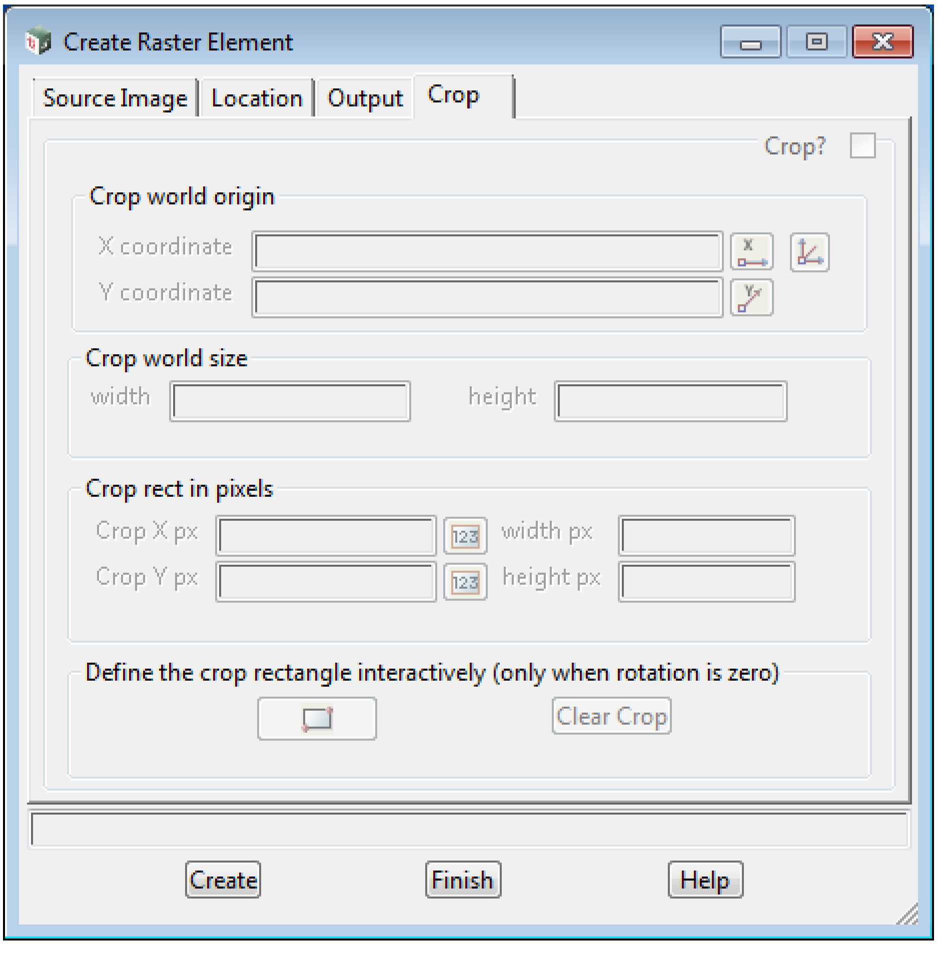
the colour of the border for the raster

Tin tin box available tins

*if **not blank**, this tin is tagged with this raster so the raster is used for draping on the tin when the tin is used in visualisations*

Crop tab

This section of documentation is a work in progress and will be updated in subsequent releases.



Crop? tick box

if ticked,

Crop world origin

X coordinate

Y coordinate

Crop world size

Width

Height

Crop rect in pixels

Crop X px

Crop Y px

Width px

Height px

Define the crop rectangle interactively (only when rotation is zero)

Clear crop

Create button

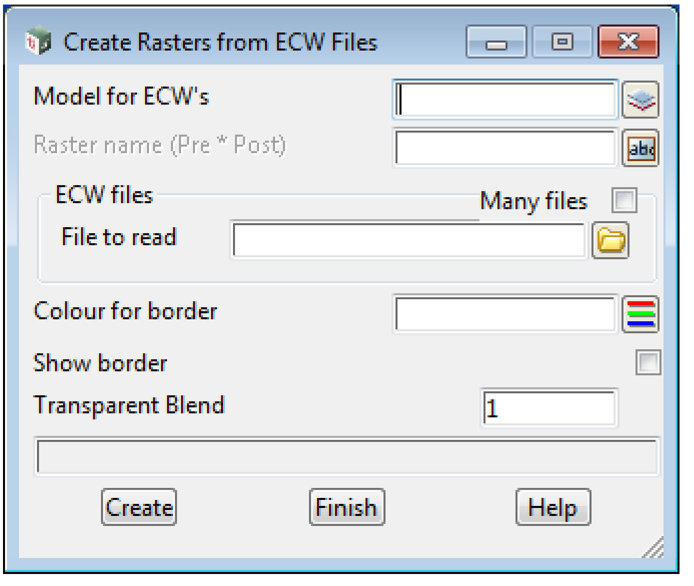
after the **Create** *button is chosen, the raster element is created.*

14.14.2 Create Rasters from ECW Files

Position of option on menu: **Strings =>Rasters =>Create from ECWs**

This option attaches one or more ECW files to a model.

Selecting **Create from ECWs** brings up the **Create Rasters from ECW Files** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model for ECWs	input		available models

name of the model to hold the new ECW rasters

Raster name (Pre*post)	text input		

*if **not blank**, pre*post text to use for the name of the raster(s) from the ECW file(s) (see* 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix)*

ECW files

Advanced	tick box	not ticked	

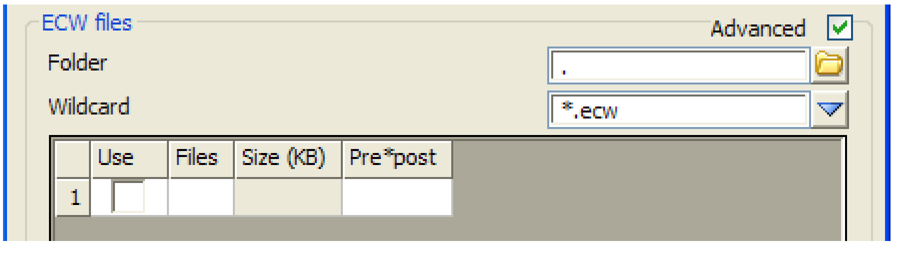
*if not ticked, **File To Read** is visible.*

File to read	file box		*.ecw files

name of the ECW file to attach to the model

Advanced	tick box		

*if ticked**,** the **File To Read** field is replaced by a grid to allow multiple ECW files to be read in, is opened. A wild card is used to select all the files to be read in.*



Folder folder box

folder to search for files using the Wild card

Wildcard choice box *.ecw, *

wild card to use in search for files in the given folder

Grid

The following fields are repeated for every row.

Use tick box

if ticked, read in the file on this row

Files output

name of the file found in the folder

Size output

file size

Colour for border colour box

colour for the border around the raster

Show border tick box

if ticked, the border will be visible

Transparent Blend

The degree of opacity, 1 = opaque, 0.1 = very transparent

Create button

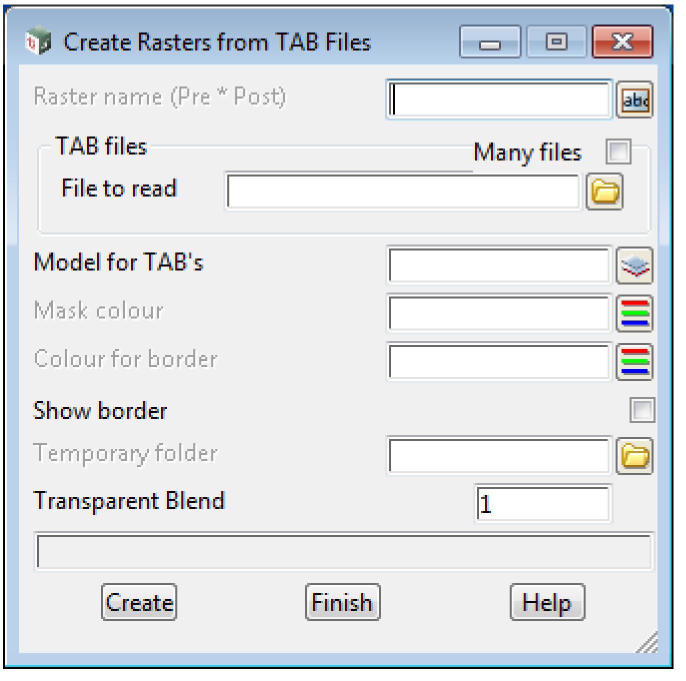
after the **Read** *button is chosen, the raster elements are created.*

14.14.3 Create Rasters from TAB Files

Position of option on menu: **Strings =>Rasters =>Create from TAB's**

This option reads one or more rasters as define by TAB files and places them in a model.

Selecting **Create from TAB's** brings up the **Create Rasters from TAB Files** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Raster name (Pre*post)	text input		

*if **not blank**, pre*post text to use for the name of the raster(s) from the ECW file(s) (see* 4.19.2 Pre*Postfix in Panel Fields *for information on using pre*postfix)*

TAB files

Advanced	tick box	not ticked	

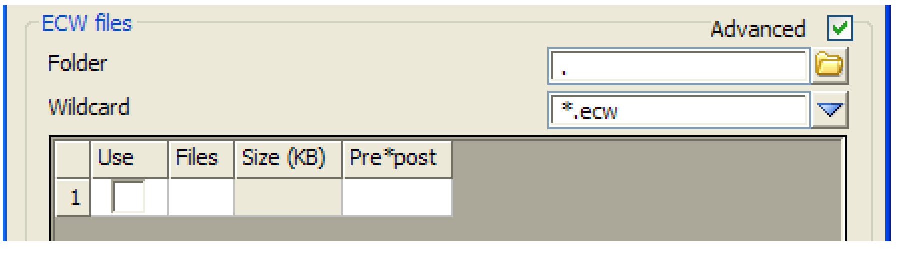
*if not ticked, **File To Read** is visible.*

File to read	file box		*.tab files

name of the TAB file to attach to the model

Advanced	tick box		

*if ticked, the **File To Read** field is replaced by a grid to allow multiple TAB files to be read in, is opened. A wild card is used to select all the files to be read in.*



Folder folder box

folder to search for files using the Wild card

Wildcard choice box *.tab, *

wild card to use in search for files in the given folder

Grid

The following fields are repeated for every row.

Use tick box

if ticked, read in the file on this row

Files output

name of the file found in the folder

Size output

file size

Model for TAB's input available models

name of the model for the rasters defined by the TAB files

Mask colour colour box available colours

colour when there is no pixel in the raster

Colour for border input default colour available colours

the colour of the border for the raster

Show border tick tick

if ticked then the border of the raster element is displayed

Temporary folder file box c:\temp select folder

folder to use as a temporary storage area whilst converting the raster to the 12d raster format. The temporary folder needs to have enough free disk space to convert the largest of the rasters to bmp format.

Transparent Blend

The degree of opacity, 1 = opaque, 0.1 = very transparent

Create button

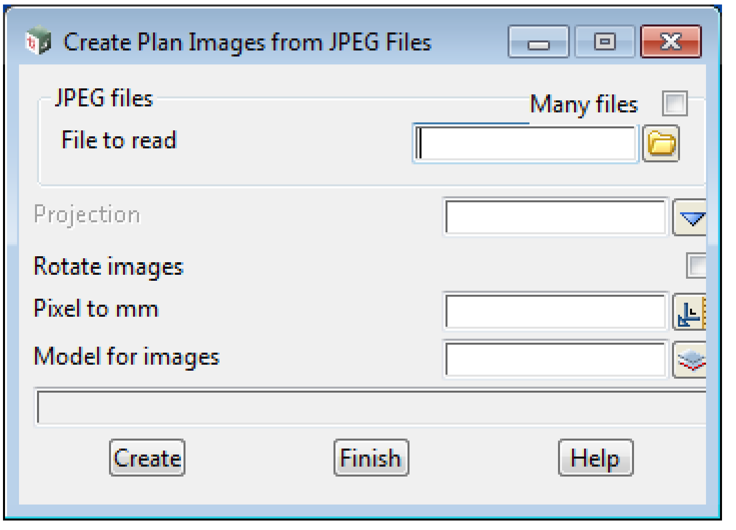
after the **Read** *button is chosen, the raster elements are created.*

14.14.4 Create Plan Images from JPEG Files

Position of option on menu: **Strings =>Rasters =>Create plan images from JPEG's**

This section of documentation is a work in progress and will be updated in subsequent releases.

Selecting **Create plan images from JPEG's** brings up the **Create Plan Images from JPEG Files** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
File to read	file box		*.jpg files

name of the JPEG file to attach to the model

Advanced	tick box		

*if ticked, the **File To Read** field is replaced by a grid to allow multiple JPEG files to be read in, is opened. A wild card is used to select all the files to be read in.*

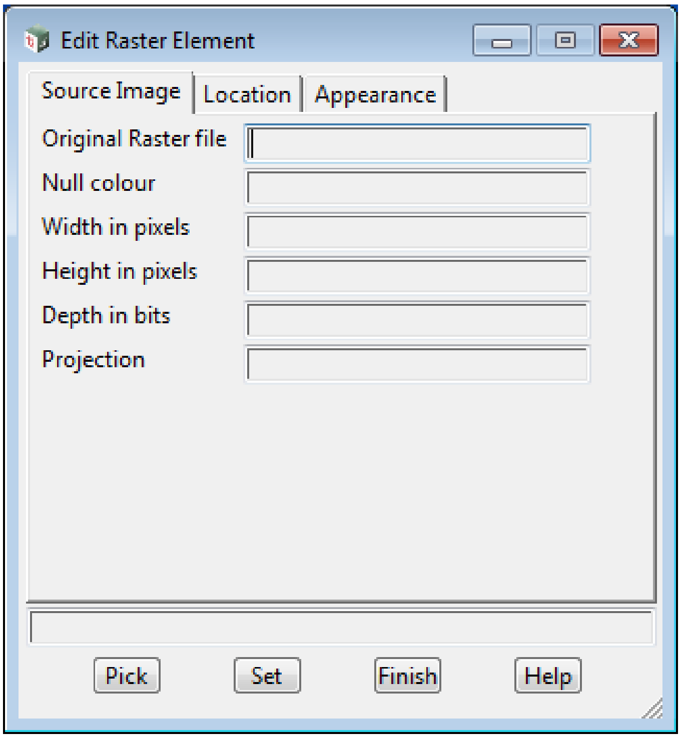
Projection	choice box		
Rotate images	tick box		
Pixel to mm	measure box		
Model for images	model box		available models

14.14.5 Edit a Raster

Position of option on menu: **Strings =>Rasters =>Edit**

The **Edit** option modifies the properties of any raster displayed in a *12d Model* view.

Selecting **Edit** brings up the **Edit Raster Element** panel.



The fields and buttons used in the panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Source Image tab

Original raster file output

file that the raster originally came from

Null colour colour box available colours

colour when there is no pixel in the raster

Width/Height in pixels output only

the width/height in pixels of the selected raster

Depth in bits output only

display the colour depth of the raster

Projection output only

display the original projection of the raster (if available)

Location tab

*When the raster is selected, the raw details for the raster (or ECW details) are displayed. The location details can then be modified and hence how the raster is positioned in world units in **12d Model** is modified.*

Data format	choice box	Raw details	Raw Details, Autocad scr file ESRI world, Geo Tiff, Mapinfo tab

*if **Raw details**, the location details are typed into the* World Location Details *section as rotation, world origin, world width and height.*
If **Autocad scr file**, *the location details are taken from the scr file given in* Location file. *An Autocad scr file gives the world co-ordinates of the corners of the raster starting in the bottom left hand corner, bottom right hand corner, top left hand corner and top right hand corner. An example of a scr file is:*

line
22109.639,148090.695
23109.639,148090.695
22109.639,149090.695
23109.639,149090.695

If **ESRI world**, *the location details are taken from the ESCRI world file given in* Location file. *An ESRI world file gives the xscale, row rotation, column rotation, yscale, x origin and y origin. For use with 12d, the row and column rotations must be the same and yscale = - xscale.*
If **Geo Tiff**, *the Location details are taken from the geotiff file given in* Location file *panel field.*
If **Mapinfo Tab**, *the location details are taken the Mapinfo tab file given in* Location file.

Location file file box
file with the location details

Anticlockwise rotation angle box
the world rotation of the raster

X/Y co-ordinate real box
the world x/y co-ordinate of the bottom left hand corner of the raster

World width/height real box
the width/height in world units of the raster

Appearance tab

Name for raster	input		

the name of the raster element

Model for raster	input		available models

model for the raster element.

Show border	tick	tick	

if ticked then the border of the raster element is displayed

Colour for border	input	default colour	available colours

the colour of the border for the raster element

Buttons at Bottom

Pick button
select the raster element to edit by clicking on the border of the raster. The details for the selected raster are then displayed in the panel.

Set button
give the raster element the new values from the panels fields

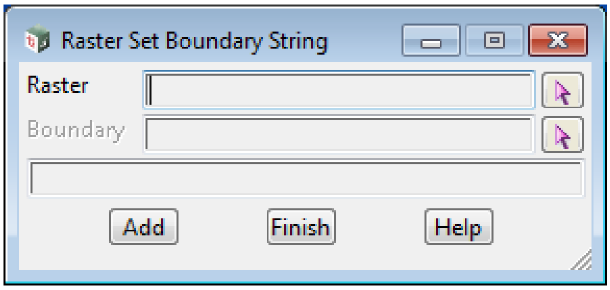
NOTE: A raster is **deleted** by using the string *Delete* option. The raster is selected for deleting by picking on the border of the raster.

14.14.6 Set a Boundary for a Raster

Position of option on menu: **Strings =>Rasters =>Set boundary**

The **Set boundary** option defines a boundary polygon for the raster. Only the parts of the raster inside the boundary polygon are displayed. The boundary polygon can have holes.

Selecting **Set boundary** brings up the **Raster Set Boundary String** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Raster	raster select box		

select the raster to have a drawing polygon set for it

Field Description	Type	Defaults	Pop-Up
Boundary	string select box		

select a polygon

Field Description	Type	Defaults	Pop-Up
Add	button		

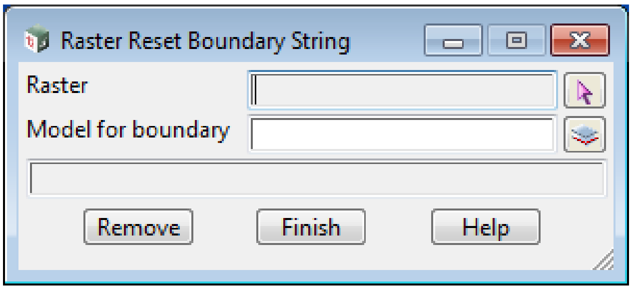
add the selected polygon as a boundary polygon for the selected raster

14.14.7 Reset the Boundary for a Raster

Position of option on menu: **Strings =>Rasters =>Reset boundary**

The **Reset boundary** option removes the boundary polygon from a raster.

Selecting **Remove boundary** brings up the **Raster Reset Boundary String** panel.



The fields and buttons used in the panel have the following functions.

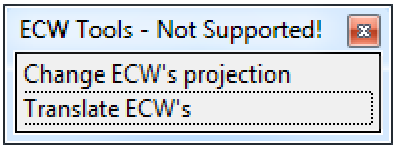
Field Description	Type	Defaults	Pop-Up
Raster	raster select box		
select the raster to have its drawing polygon removed			
Model for boundary	model box		available models
model to place the removed boundary polygon in			
Remove	button		
remove the boundary polygon from the selected raster			

14.14.8 ECW Tools - Not Supported!

Position of option on menu: **Strings =>Rasters =>ECW tools**

Selecting **ECW tools** brings up the **ECW tools -Not Supported!** menu.

This section of documentation is a work in progress and will be updated in subsequent releases.



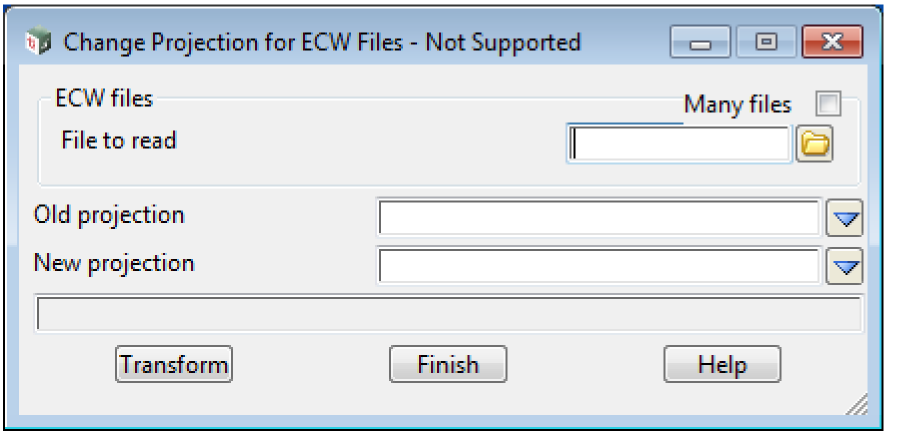
For the option Change ECW's projection go to 14.14.8.1 Change ECWs Projection
Translate ECW's 14.14.8.2 Translate ECWs

14.14.8.1 Change ECWs Projection

Position of option on menu: **Strings =>Rasters =>ECW tools=> Change ECW's Projection**

Selecting **Change ECW's Projection** brings up the **Change Projection for ECW Files - Not Supported** panel.

This section of documentation is a work in progress and will be updated in subsequent releases.

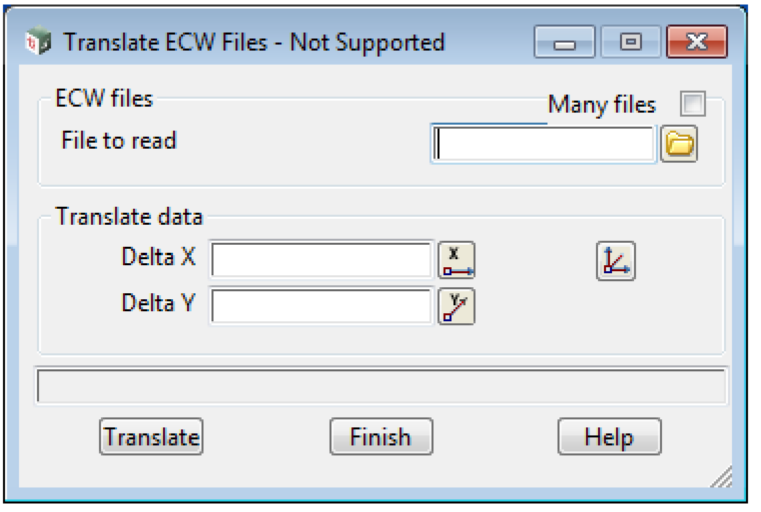


14.14.8.2 Translate ECWs

Position of option on menu: **Strings =>Rasters =>ECW tools=> Translate ECW's**

Selecting **Translate ECW's** brings up the **Translate ECW Files - Not Supported** panel.

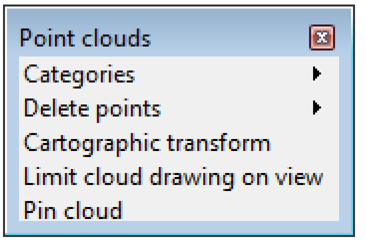
This section of documentation is a work in progress and will be updated in subsequent releases.


Translate ECW Files - Not Supported
ECW files
Many files
File to read
Translate data
Delta X
Delta Y
Translate
Finish
Help

14.15 Point Clouds

Position of menu: **Strings =>Point clouds**

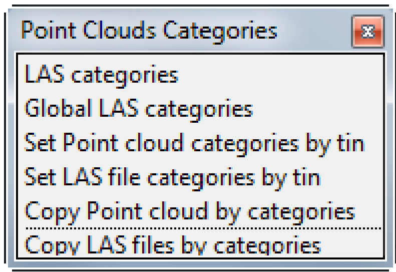
The **Point clouds** walk-right menu is



For the options go to:

14.15.1 Categories

Position of menu: **Strings =>Point clouds =>Categories**



For the options go to:

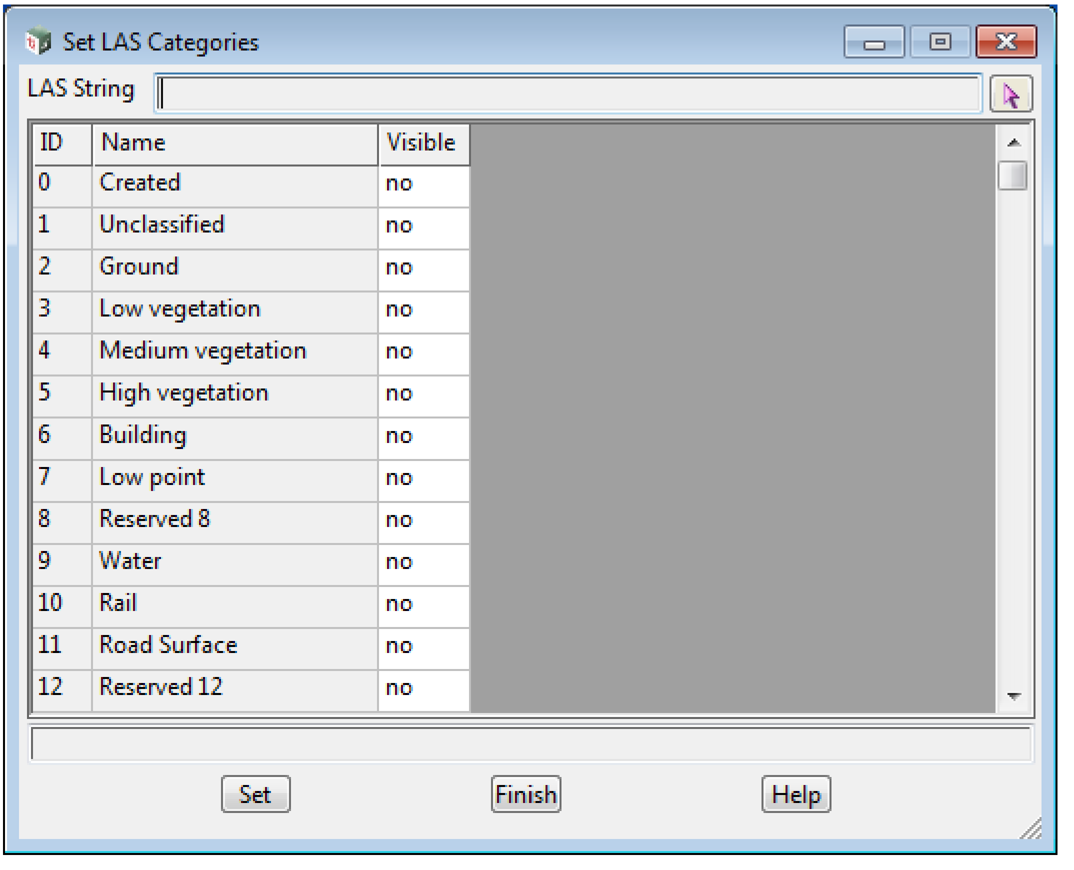
14.15.1.1 Set LAS Categories
14.15.1.2 Global Set LAS Categories
14.15.1.3 Setting Point Cloud Categories by TIN
14.15.1.4 Setting LAS File Categories by TIN
14.15.1.5 Copying Point Cloud Files by Categories
14.15.1.6 Copying LAS Files by Categories

14.15.1.1 Set LAS Categories

Position of option on menu: **Strings =>Point clouds =>Categories =>LAS Categories**

This section of documentation is a work in progress and will be updated in subsequent releases.

Selecting **LAS Categories** fires up the **Set LAS Categories** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
LAS String	string select		
Created	tick box		
Unclassified	tick box		
Ground	tick box		
Low vegetation	tick box		
Medium vegetation	tick box		
High vegetation	tick box		
Building	tick box		
Low point	tick box		
Model key	tick box		
Water	tick box		
Overlap points	tick box		

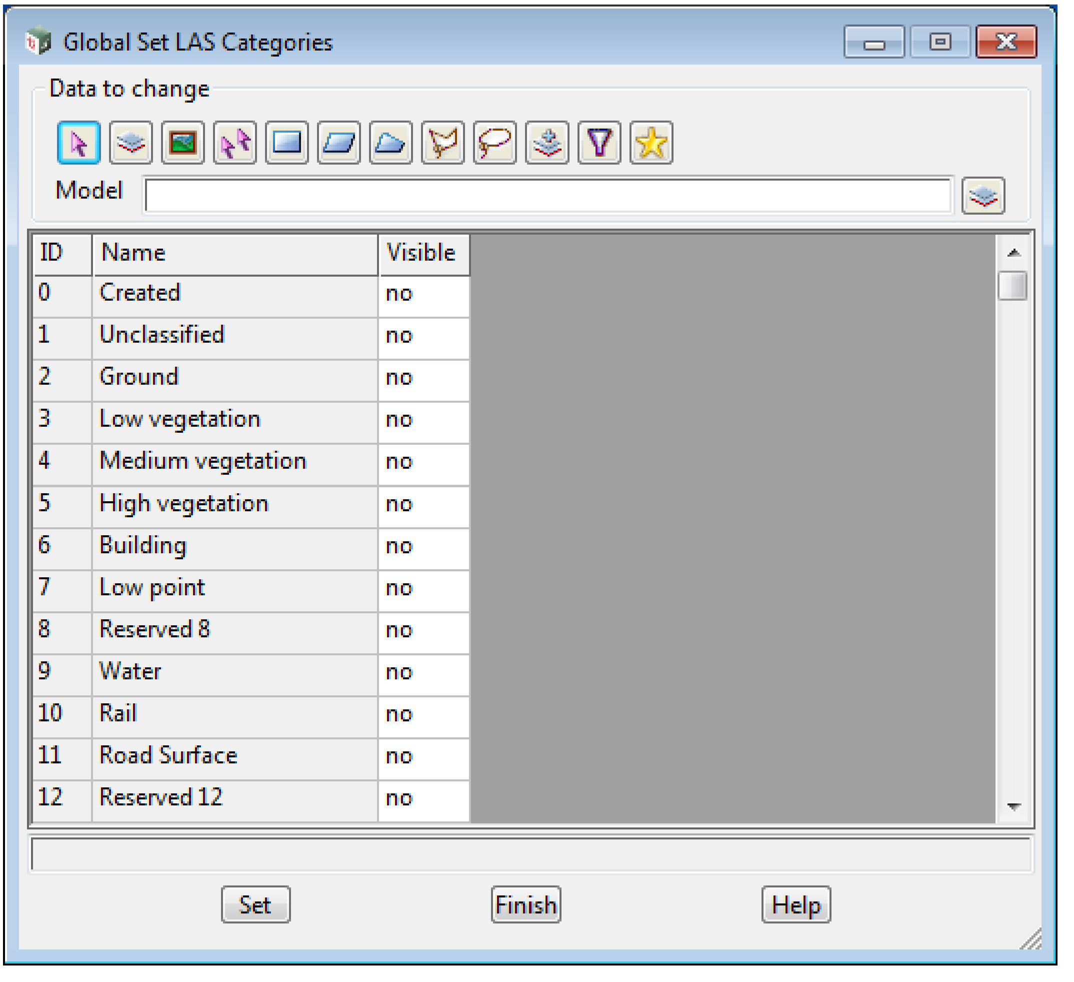
Set button

14.15.1.2 Global Set LAS Categories

Position of option on menu: **Strings =>Point clouds =>Categories =>Global LAS Categories**

This section of documentation is a work in progress and will be updated in subsequent releases.

Selecting **Global LAS Categories** fires up the **Global Set LAS Categories** panel.



The fields and buttons used in this panel have the following functions.

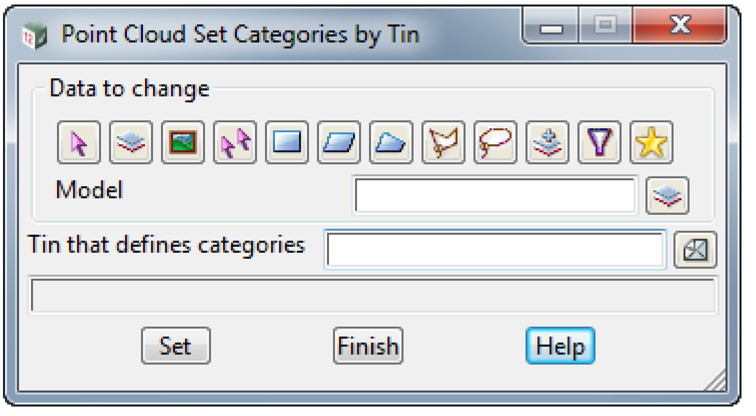
Field Description	Type	Defaults	Pop-Up
Data Data to change			
Data source type		Model	
data selection type - for a full description go to 4.19.3 Data Source.			
Data source	input		
source of data to be processed.			
Created	tick box		
Unclassified	tick box		
Ground	tick box		
Low vegetation	tick box		
Medium vegetation	tick box		
High vegetation	tick box		

Building	tick box
Low point	tick box
Model key	tick box
Water	tick box
Overlap points	tick box
Set	button

14.15.1.3 Setting Point Cloud Categories by TIN

Position of option on menu: **Strings =>Point clouds =>Categories =>Set Point cloud categories by tin**

Selecting **Set Point cloud categories by tin** fires up the **Point Cloud Set Categories by Tin** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data to change			
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Tin that defines categories	tin box		available tins

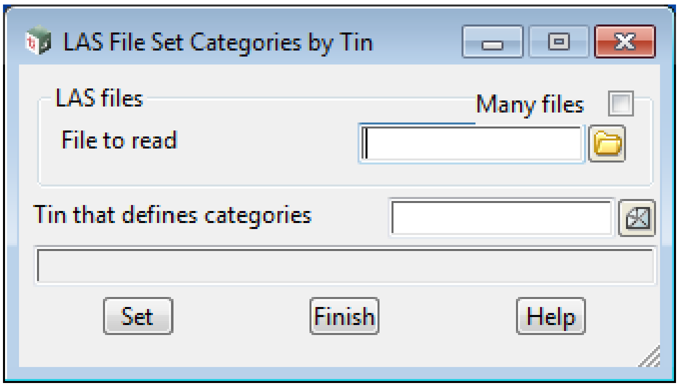
the category of each point cloud point is set from the colour of the triangle the point is over, so if red is colour number 1 within colours.4d, the category becomes '1'.

Set	button		

assign categories to cloud string from the selected tin.

14.15.1.4 Setting LAS File Categories by TIN

Position of option on menu: **Strings =>Point clouds =>Categories =>Set LAS file categories by tin**
Selecting **Set LAS file categories by tin** fires up the **LAS File Set Categories by Tin** panel.



The fields and buttons used in this panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Many files tick box

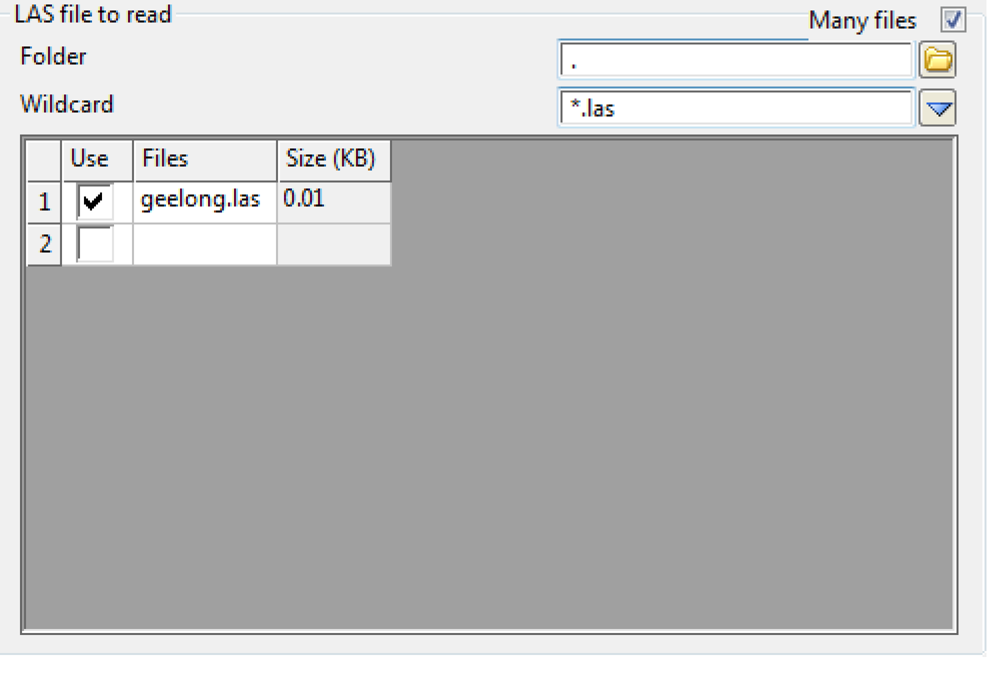
*if **ticked**, a grid to allow multiple .las files to be set is opened. A wild card is used to select all the files to be read in.*

Folder file box

folder to search for files using the Wild card.

Wildcard input

wild card to use in search for files in the given folder.



Use tick box

*if **ticked**, read in the file*

Files output

name of the file in the folder

Size output

file size

Tin that defines categories tin box

the category of each point cloud point is set from the colour of the triangle the point is over, so if red is colour number 1 within colours.4d, the category becomes '1'.

Set button

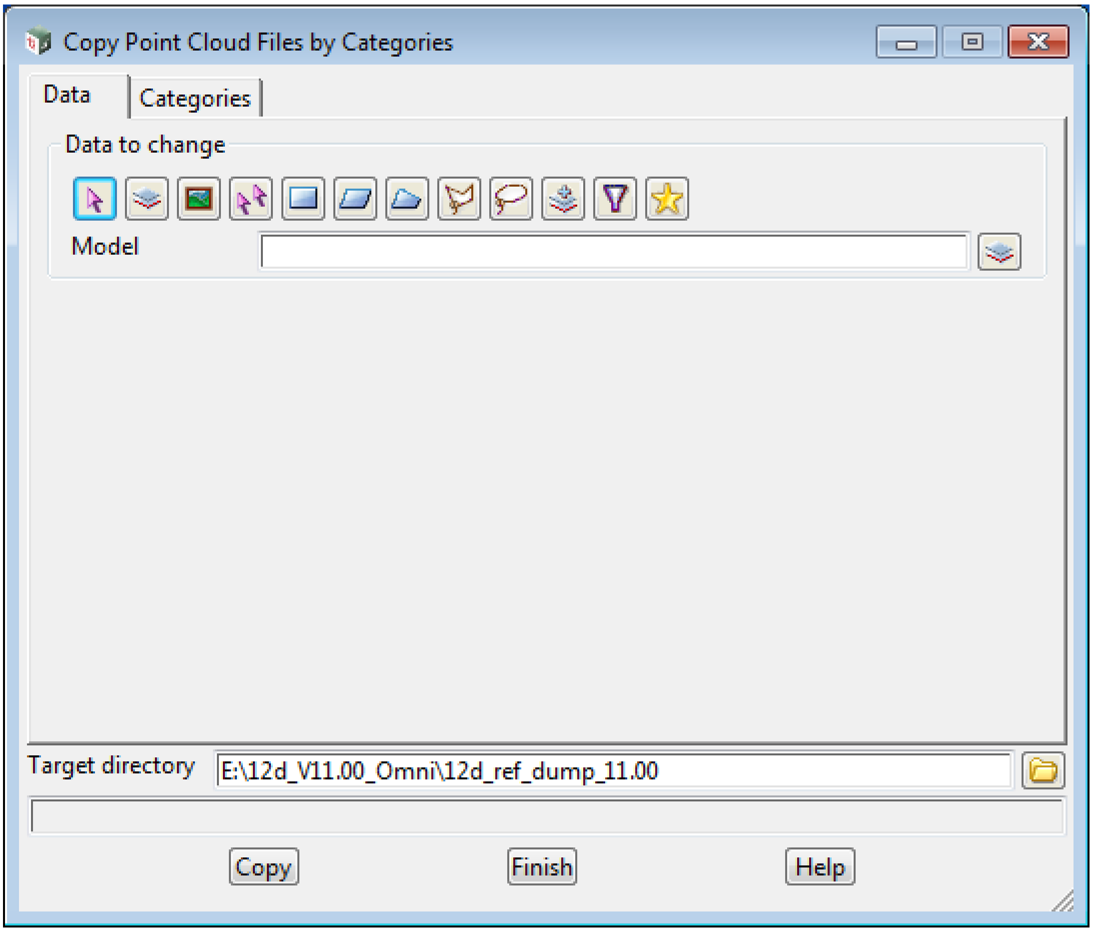
assign categories to LAS files from the selected tin.

14.15.1.5 Copying Point Cloud Files by Categories

Position of option on menu: **Strings =>Point clouds =>Categories =>Copy Point cloud by categories**

This section of documentation is a work in progress and will be updated in subsequent releases.

Selecting **Copy Point cloud by categories** fires up the **Copy Point Cloud Files by Categories** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Data tab

Data to change

Data source type Model

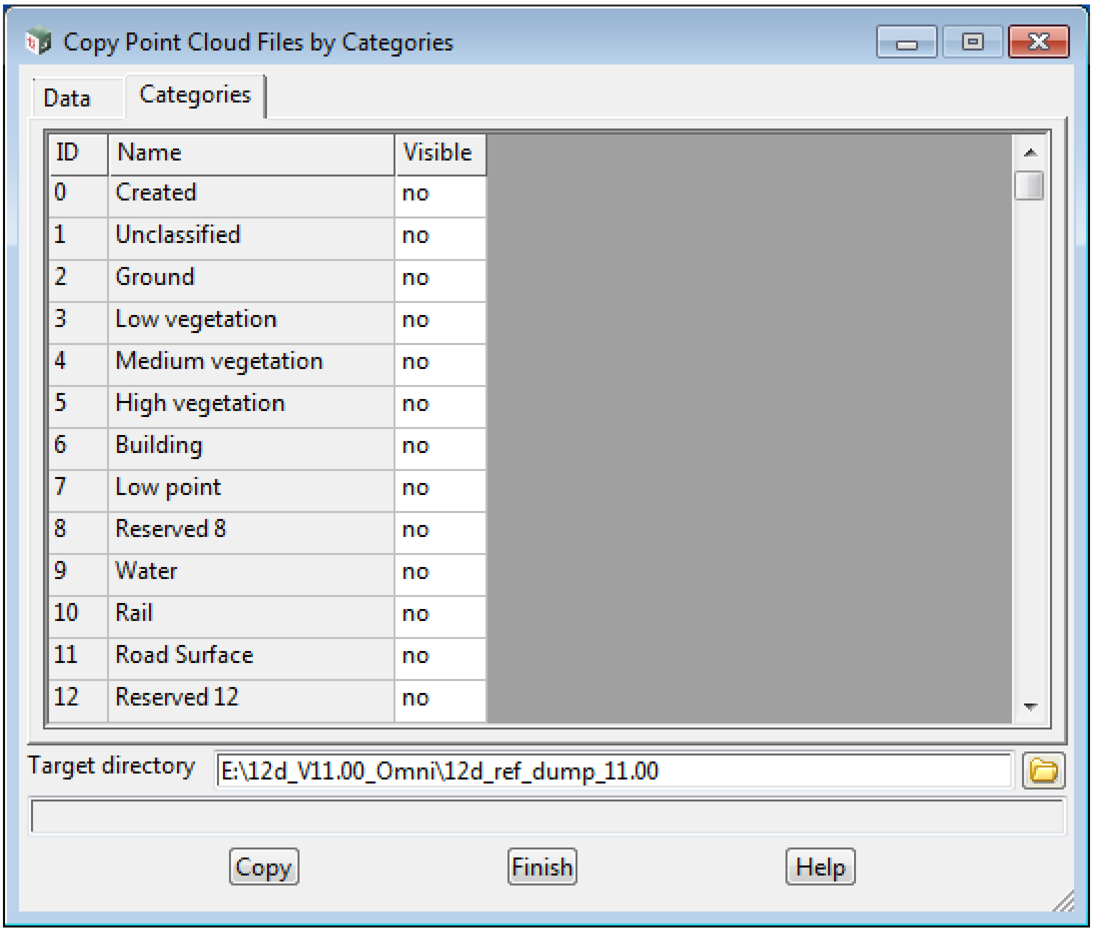
data selection type - for a full description go to 4.19.3 Data Source.

Data source input

source of data to be processed.

Target directory file box

Categories tab



ID

Name

Visible

Button at bottom

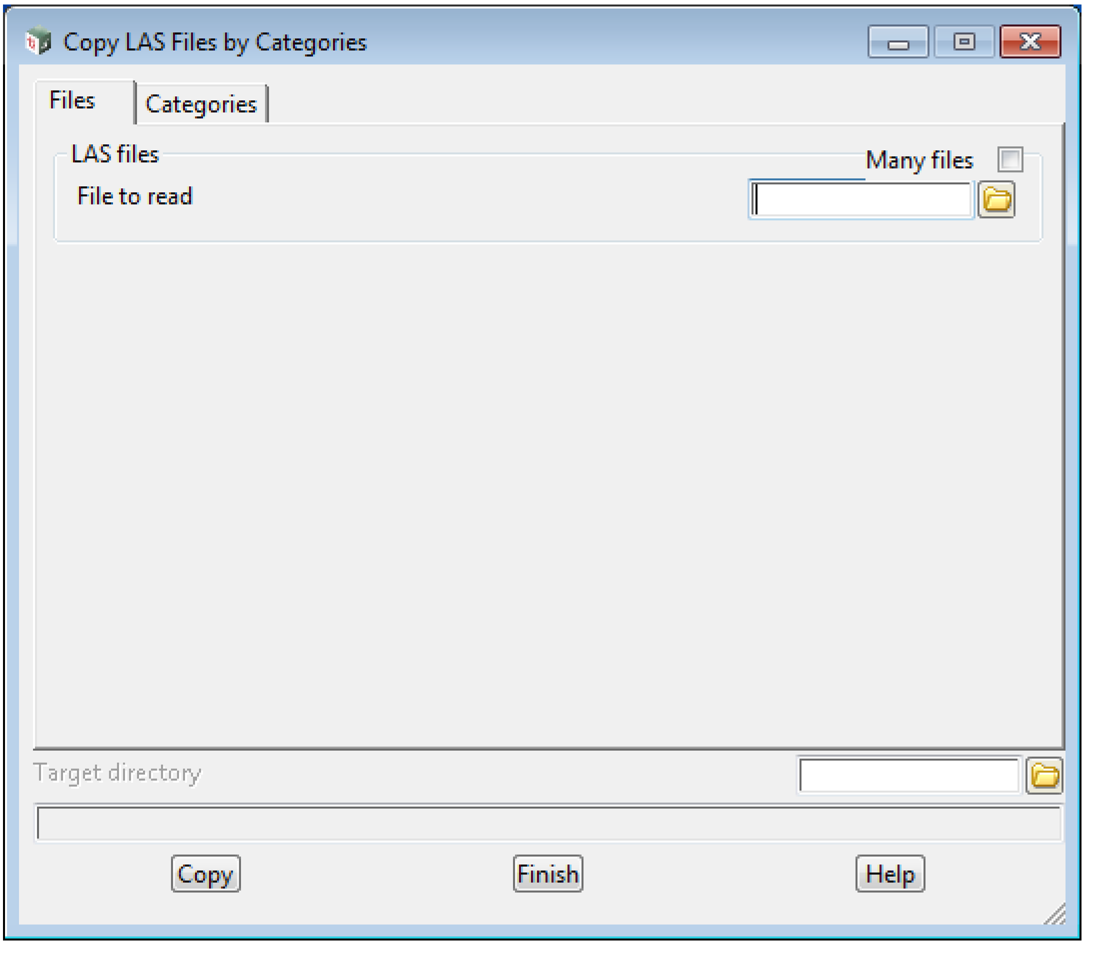
Copy button

14.15.1.6 Copying LAS Files by Categories

Position of option on menu: **Strings =>Point clouds =>Categories =>Copy LAS files by categories**

This section of documentation is a work in progress and will be updated in subsequent releases.

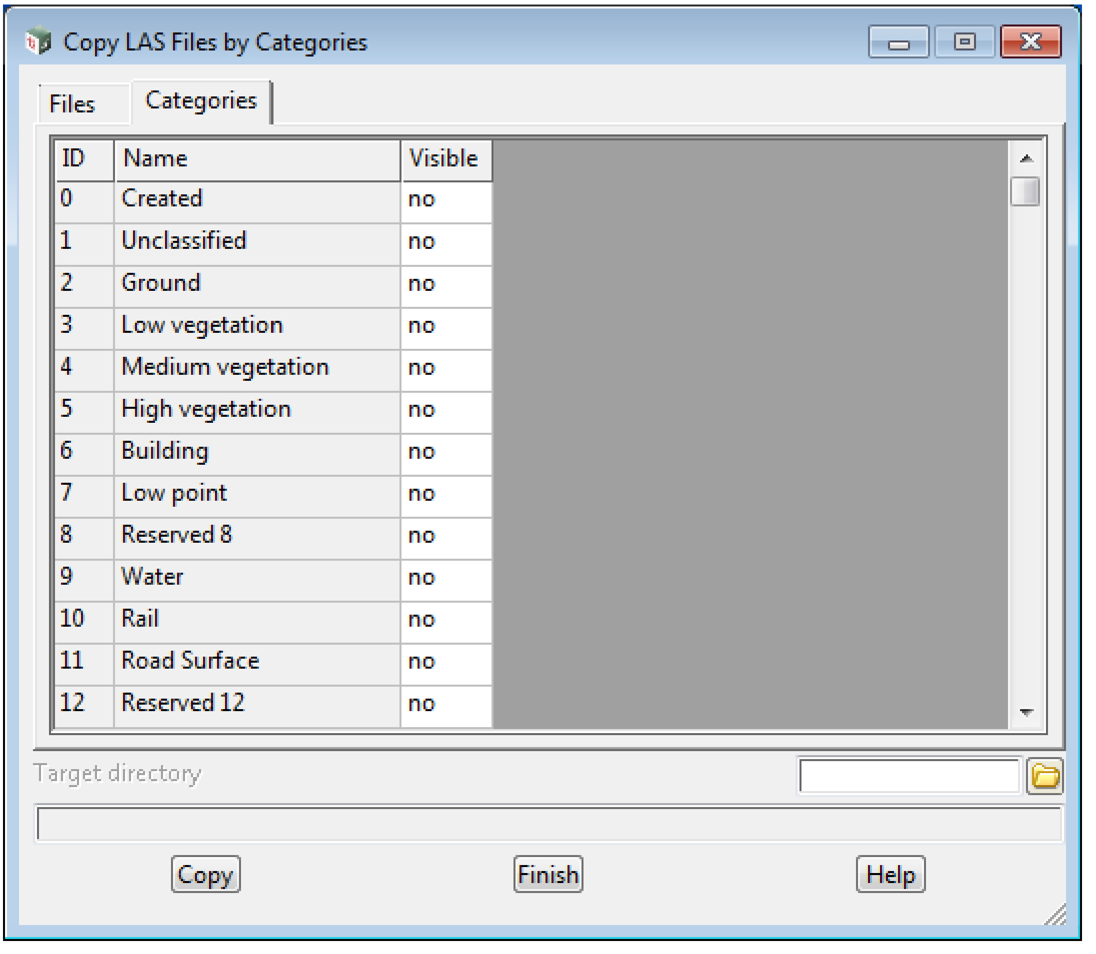
Selecting **Copy LAS files by categories** fires up the **Copy LAS Files by Categories** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Files tab			
Many files	tick box		
File to read	file box		
Target directory	file box		

Categories tab



ID

Name

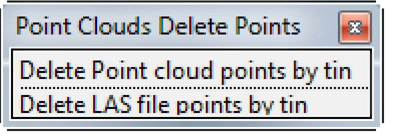
Visible

Button at bottom

Copy button

14.15.2 Delete Points

Position of menu: **Strings =>Point clouds =>Delete Points**



See

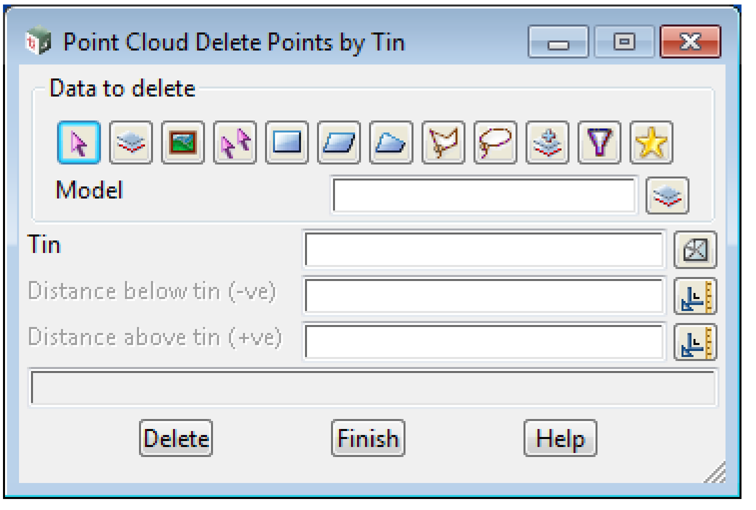
14.15.2.1 Deleting Point Cloud Points by TIN
14.15.2.2 Deleting LAS File Points by TIN

14.15.2.1 Deleting Point Cloud Points by TIN

Position of option on menu: **Strings =>Point clouds =>Delete Points =>Delete Point cloud points by tin**

This section of documentation is a work in progress and will be updated in subsequent releases.

Selecting **Delete Point cloud points by tin** fires up the **Point Cloud Delete Points by Tin** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Data to change

Data source type — Defaults: Model

data selection type - for a full description go to 4.19.3 Data Source.

Data source — Type: input

source of data to be processed.

Tin — Type: tin box — Pop-Up: available tins

the delete status of each point cloud point is set from the triangle the point is over, so if the point is over a valid triangle, the point is marked as active. Points not over any triangle are marked as deleted.

Distance below tin (-ve) — Type: measure box

the delete status of each point cloud point is set from the distance relative to the tin, so if the distance is less than the number entered, the point is marked as deleted.

Note: If 'Distance above tin' is also entered, the point must lie in between the two values for the point to be marked as active.

Distance above tin (+ve) — Type: measure box

the delete status of each point cloud point is set from the distance relative to the tin, so if the distance is greater than the number entered, the point is marked as deleted.

Note: If 'Distance below tin' is also entered, the point must lie in between the two values for the point to be marked as active.

Delete — Type: button

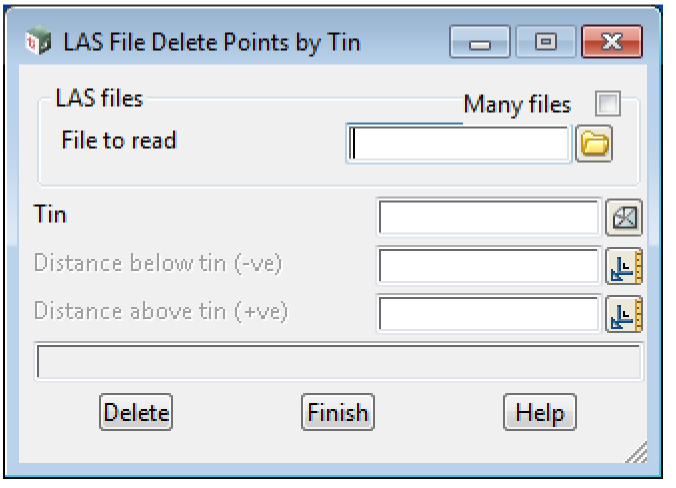
mark point cloud points as active or deleted according to the data entered above.

14.15.2.2 Deleting LAS File Points by TIN

Position of option on menu: **Strings =>Point clouds =>Delete Points =>Delete LAS file points by tin**

This section of documentation is a work in progress and will be updated in subsequent releases.

Selecting **Delete LAS file points by tin** fires up the **LAS File Delete Points by Tin** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Many files tick box

File to read file box

Tin tin box

the delete status of each point in the LAS file is set from the triangle the point is over, so if the point is over a valid triangle, the point is marked as active. Points not over any triangle are marked as deleted.

Distance below tin (-ve) measure box

the delete status of each point in each LAS file is set from the distance relative to the tin, so if the distance is less than the number entered, the point is marked as deleted.

Note: If 'Distance above tin' is also entered, the point must lie in between the two values for the point to be marked as active.

Distance above tin (+ve) measure box

the delete status of each point in each LAS file is set from the distance relative to the tin, so if the distance is greater than the number entered, the point is marked as deleted.

Note: If 'Distance below tin' is also entered, the point must lie in between the two values for the point to be marked as active.

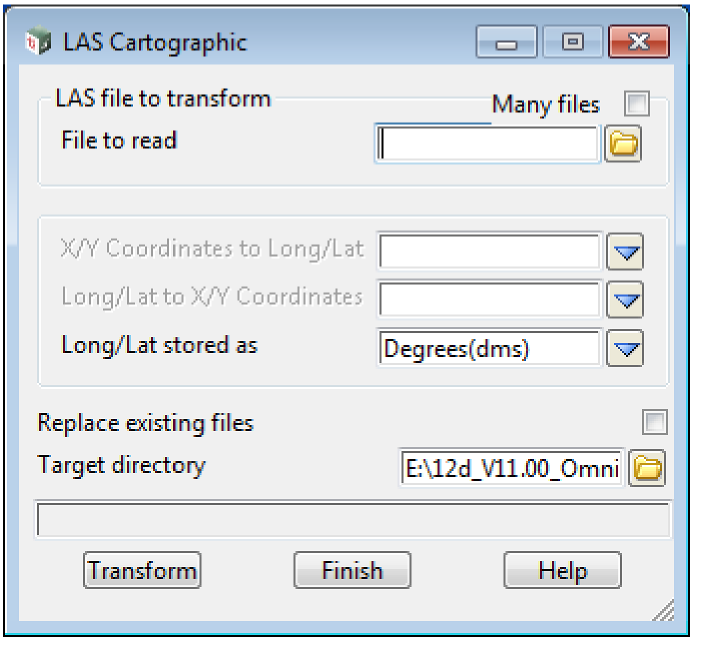
Delete button

mark point cloud points as active or deleted according to the data entered above.

14.15.3 Cartographic Transform

Position of option on menu: **Strings =>Point clouds =>Cartographic transform**

Selecting **Cartographic transform** brings up the **LAS Cartographic** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Many files	tick box	not ticked	

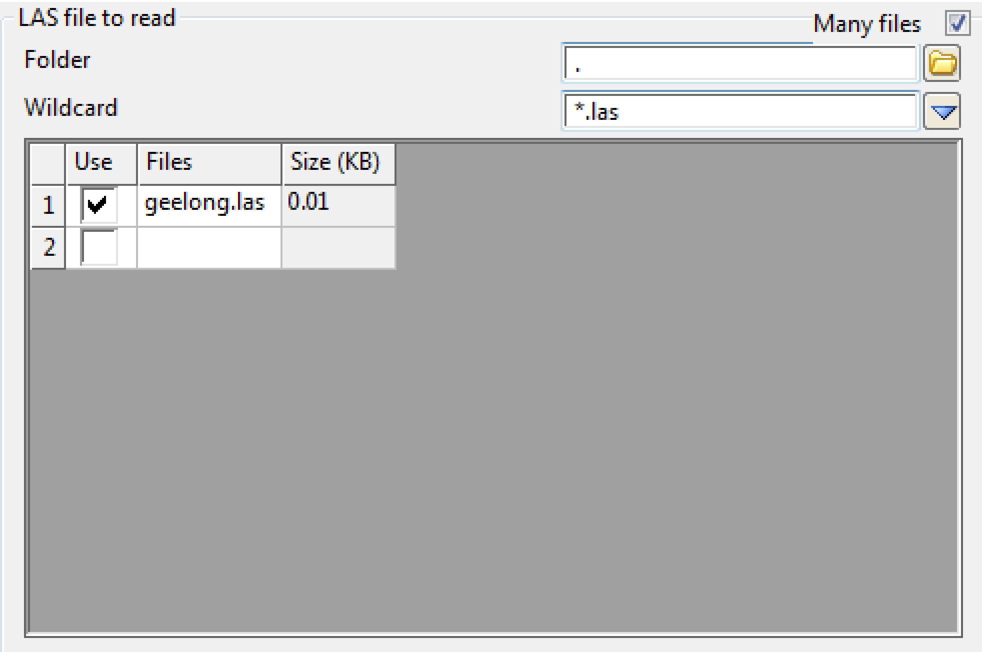
*if **ticked**, a grid to allow multiple .las files to be read in is opened. A wild card is used to select all the files to be read in.*

Folder	file box		

folder to search for files using the Wild card

Wildcard	input		

wild card to use in search for files in the given folder



Use tick box

*if **ticked**, read in the file*

Files output

name of the file in the folder

Size output

file size

X/Y Coordinates to Long/Lat choice box

Long/Lat to X/Y Coordinates choice box available projections

*if **non-blank**, the cartographic projection to apply to the longitude-latitude values.*
*If **blank**, the co-ordinates are not transformed from (longitude, latitude) to (x,y). Hence the initial (x,y) co-ordinates are transformed to (longitude, latitude) by the transformation given in the* **x/y co-ordinates to Long/Lat** *field and then left in (longitude, latitude). Note that in the southern hemisphere, the latitude values are **negative**.*

Long/Lat stored as choice box degrees radians
degrees
decimal degrees

format for the longitude and latitudes - either radians, degrees (in 4.17.1 HP Notation *for degrees, minutes and seconds) or decimal degrees.*

Replace existing files tick box not ticked

Target directory file box

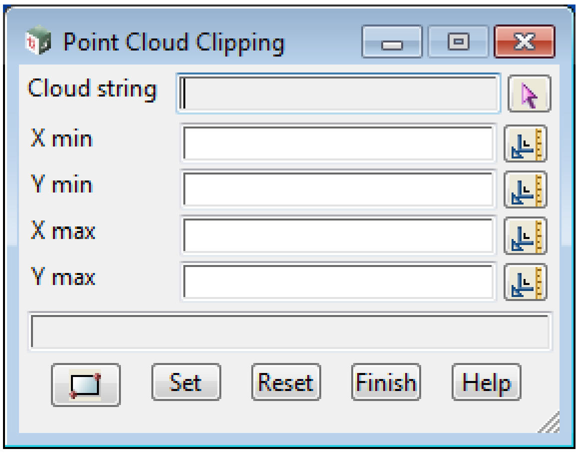
directory where the file will be saved

Transform button

14.15.4 Limiting Cloud Drawing on View

Position of option on menu: **Strings =>Point clouds =>Limit cloud drawing on view**

Selecting **Limit cloud drawing on view** fires up the **Point Cloud Clipping** panel.



The fields and buttons used in this panel have the following functions.

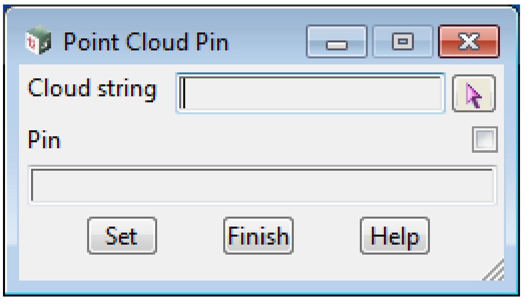
Field Description	Type	Defaults	Pop-Up
Cloud string	string select		
X/Y min	measure boxes		
X/Y max	measure boxes		
Set	button		
Reset	button		

14.15.5 Point Cloud Pinning

Position of option on menu: **Strings =>Point clouds =>Pin cloud**

The **Point Cloud Pin** option allows a point cloud to be stored in memory rather than read from disk on demand. If on demand, once the point cloud is not needed, it is removed from memory, so this 'pinning' may improve performance.

Selecting **Pin cloud** fires up the **Point Cloud Pin** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Cloud string	string select		

select the point cloud string to be pinned or unpinned.

Pin	tick box		

if ticked, the selected point cloud string will be pinned.

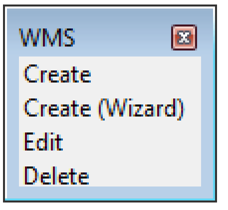
Set	button		

runs the option.

14.16 WMS

Position of option on menu: **File I/O =>WMS**
Position of option on menu: **Strings =>WMS**

The **WMS** walk-right menu containing these options is:



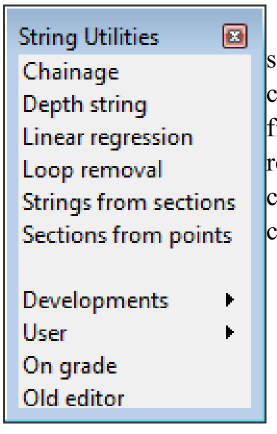
For the option *Create* go to	8.6.21.1 Download from a WMS
Create (Wizard)	8.6.21.2 WMS Wizard
Edit	8.6.22.1 Edit a WMS Image
Delete	8.6.22.2 Delete a WMS Image

14.17 Utilities

Position of menu: **Strings =>Utilities**

The **Utilities** menu contains miscellaneous options involving strings.

The Utilities walk-right menu is



set string chainage at a point
calculate depth string, report offset & xfall
fit lines and arcs
remove loops from strings
create strings from labelled sections
create section from projecting points onto a line

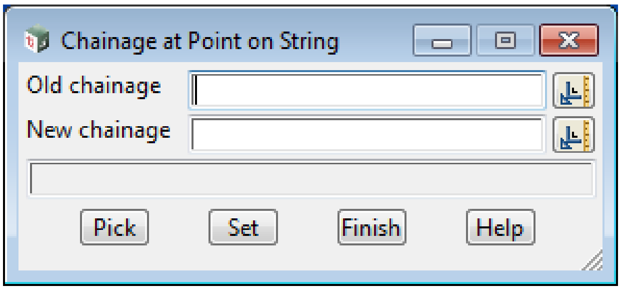
For the option *Chainage* go to	14.17.1 Chainage.
Depth string	14.17.2 Depth String
Linear regression	17.14.7 Linear Regression
Loop removal	14.17.4 Loop Removal
Strings from sections	14.17.5 Strings from Sections
Sections from points	20.14.9 Sections from Points
Developments	14.17.6 Developments
On grade	14.17.7 On Grade
Old editor	14.17.8 Old Editor

14.17.1 Chainage

Position of option on menu: **Strings =>Utilities =>Chainage**

The **chainage** menu option allows the user to specify the start chainage for a string by specifying what the chainage will be at a selected point on the string. The string's start chainage is then adjusted so that the point has the given chainage.

Selecting **chainage** fires up the **chainage at point on string** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Old chainage output

reports the initial chainage of the dropped point

New chainage input

the new chainage for the dropped point.

Pick button

two objects to pick - pick the string to modify the chainage at a point and then pick a point which will be dropped onto the selected string to give the point that will have its chainage modified. The existing chainage on the string of the dropped point is then written to the old chainage *panel field.*

Set button

on selecting set, the start chainage is modified so that the chainage at the picked point is the value in the new chainage *field. The* old chainage *field is then updated with the new chainage.*

Please continue to the next section 14.17.2 Depth String.

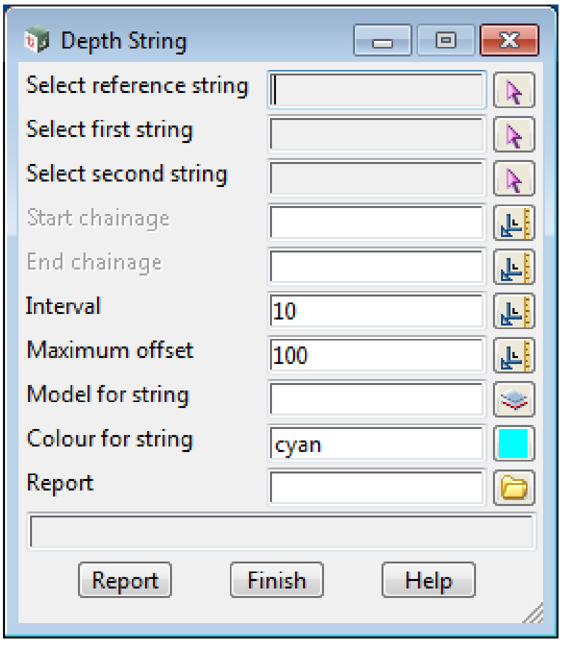
14.17.2 Depth String

Position of option on menu: **Strings =>Utilities =>Depth string**

The **depth string** option calculates the horizontal and vertical offsets and the cross-fall between two strings and then creates and/or reports on the string.

A reference string is selected which is used to define chainages. Lines perpendicular to the reference string are taken at regular chainages and intersected (in plan) with the first and second strings. The horizontal and vertical offsets and the cross-fall between the two strings are calculated at the intersection points.

On selecting the **Depth string** option, the **Depth string** panel is displayed.



The fields and buttons in this panel are used as follows

Field Description	Type	Defaults	Pop-Up

Start/End chainage input

if blank, the start/end chainage of the reference string is used.
if non-blank, the given chainage is used as the start/end chainage.

Interval input

chainage interval to calculate values at.

Maximum offset input

if non-blank, the maximum distance to search from the reference string to find the 1st and 2nd strings.

Model for string input available models

if non-blank, a depth string is created and placed in this model.

Colour for string input available colours

the colour for the depth string.

Report file input *.rpt

if non-blank, the file for the offset and cross-fall report

Ref/1st/2nd button

select the reference/first/second string.

Report button

calculate a depth string and produce an offset and cross-fall report on it.

Please continue to the next section 14.17.3 Linear Regression.

14.17.3 Linear Regression

Position of option on menu: **Strings =>Utilities =>Linear regression**

Position of option on menu: **Survey =>Extras =>Linear regression**

This option is documented in 17.14.7 Linear Regression.

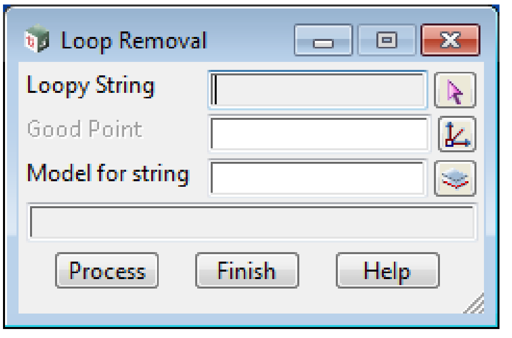
Please continue to the next section 14.17.4 Loop Removal.

14.17.4 Loop Removal

Position of option on menu: **Strings =>Utilities =>Loop removal**

The **loop removal** option tries to remove loops from strings by adding points where the string self intersects and removing the loops at those points. The level of the added points are the mean of the self intersect points.

Selecting **loop removal** fires up the **loop removal** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Loopy string	string select		
pick the string to try and remove loops from.			
Good point	xyz box		
If the automatic loop removal is incorrect, pick a good point on the string.			
Model for string	model box		available models
model for the processed string to go to.			
Process	button		
try and remove string points from the loops in the string.			

Please continue to the next section 14.17.5 Strings from Sections.

14.17.5 Strings from Sections

Position of option on menu: **Strings =>Utilities =>Strings from sections**

The **strings from sections** option creates strings by joining the common named points on successive 4d strings (usually generated as sections). This option is documented under

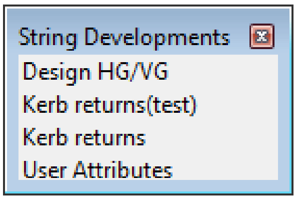
Design => X-Sections => Strings from Sections

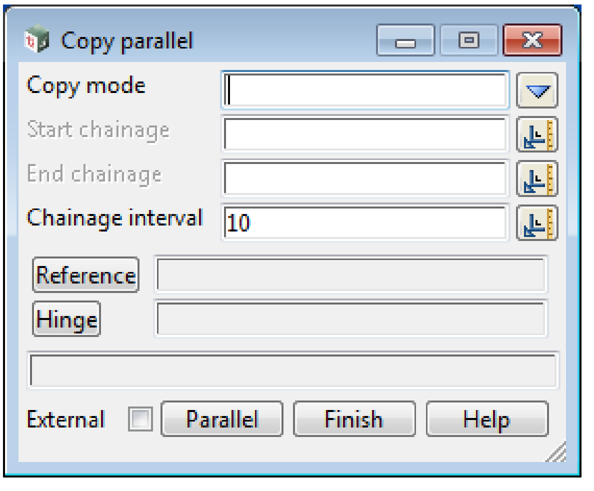
Please go to the section 20.14.8 Strings from Sections.

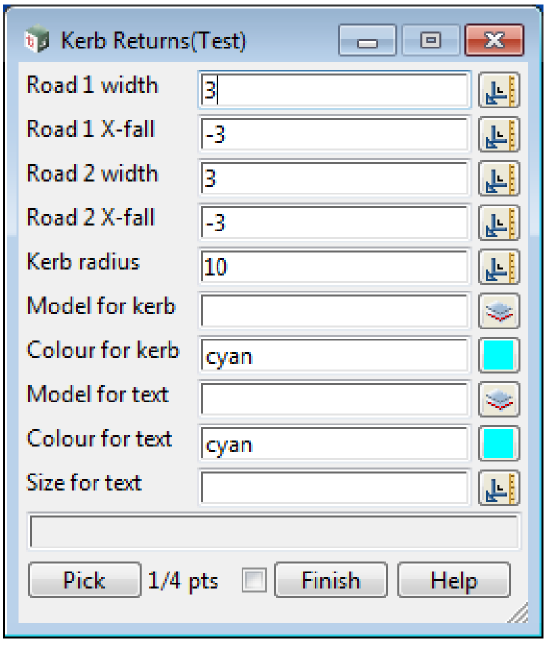
14.17.6 Developments

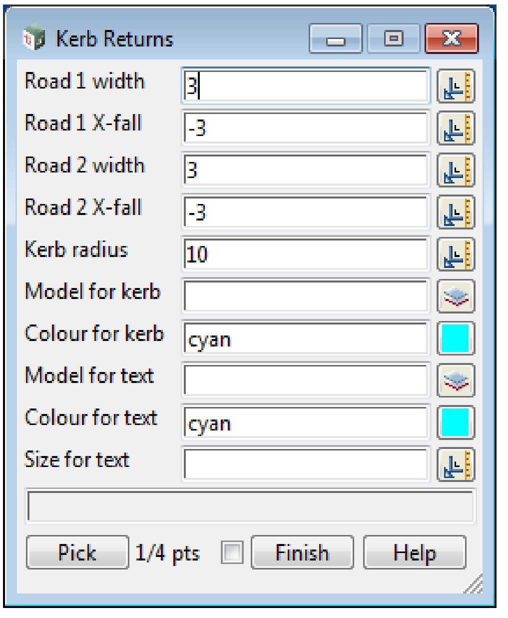
Position of menu: **Strings =>Utilities =>Developments**

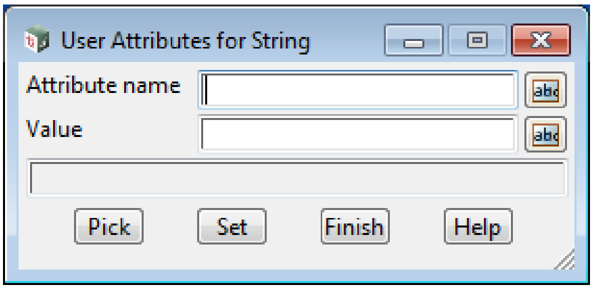
Some of the options under **string utilities** are still under development, or are being phased out (Cogo (trial)) and won't be documented or released until future versions of 12d Model. The current options on the *Developments* menu are

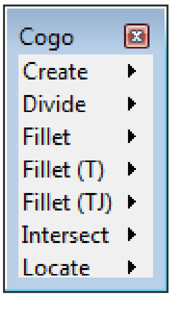


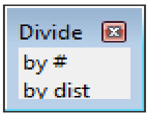


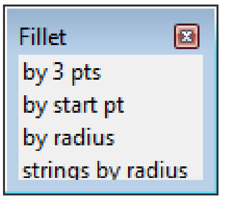


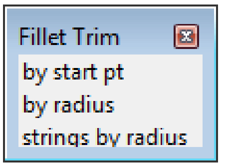

Kerb Returns
Road 1 width
3
Road 1 X-fall
-3
Road 2 width
3
Road 2 X-fall
-3
Kerb radius
10
Model for kerb
Colour for kerb
cyan
Model for text
Colour for text
cyan
Size for text
Pick
1/4 pts
Finish
Help


User Attributes for String
Attribute name
Value
Pick
Set
Finish
Help

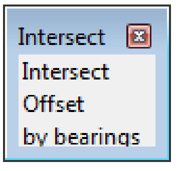

Cogo
Create
Divide
Fillet
Fillet (T)
Fillet (TJ)
Intersect
Locate

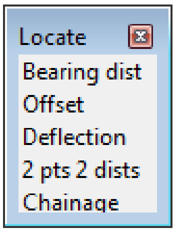

Divide
by #
by dist


Fillet
by 3 pts
by start pt
by radius
strings by radius


Fillet Trim
by start pt
by radius
strings by radius


Fillet Trim Join
by start pt
by radius
strings by radius


Intersect
Intersect
Offset
by bearings


Locate
Bearing dist
Offset
Deflection
2 pts 2 dists
Chainage

14.17.7 On Grade

Position of option on menu: **Survey =>Extras =>On grade**

Position of option on menu: **Strings =>Utilities =>On grade**

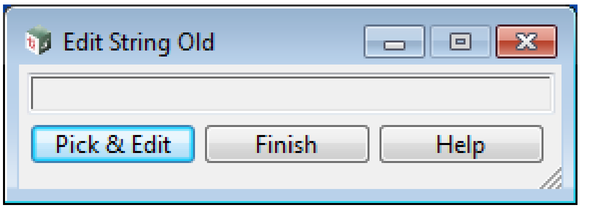
For Information on this option please go to 17.14.9 On Grade in the section 17.14 Extras

14.17.8 Old Editor

Position of option on menu: **Strings =>Utilities =>Old Editor**

This section of documentation is a work in progress and will be updated in subsequent releases.

Selecting **Old editor** brings up the **Edit String Old** panel.

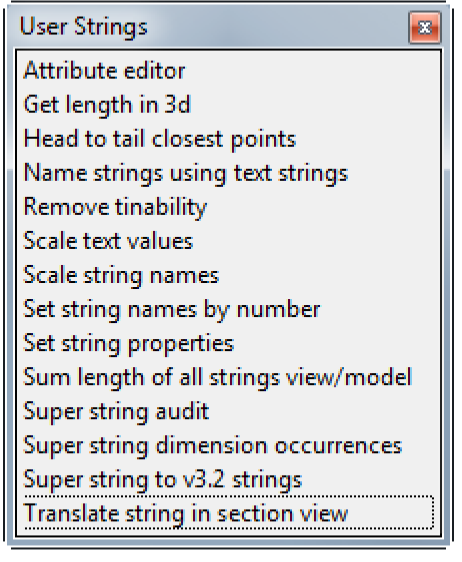


14.18 User

Position of option on menu: **Strings =>User**

The walk-right menu **Strings =>User** has extra macro options.These could be from 12d Solutions or users.

The **User Strings** walk-right menu is



Edit the top 10 attributes
Calculates 3d length

For the option *Attribute Editor*, go to 14.18.1 Attribute Editor

Get length in 3d 29.3.14 Length in 3d

Head to tail closest points 29.3.15 XXX Head to Tail Closest Points

Name strings using text strings 29.3.21 XXX Name Section Strings by Picking Text

Remove tinability 29.3.26 XXX Remove Tinability

Scale text values 29.3.28 XXX Scale Text Values

Scale string names 29.3.29 XXX Scale String Names

Set string names by number 29.3.30 XXX Set String Names by Number

Set string properties 29.3.31 XXX String Operations

Sum length of all strings view/model 29.3.36 Total Length of Strings

Super string audit 29.3.37 XXX Super String Dimension Occurrences

Super string dimension occurrences 29.3.37 XXX Super String Dimension Occurrences

Super string to v3.2 strings 29.3.38 XXX Transform V4 to V3.2

Translate string in section view 29.3.39 XXX Section Move

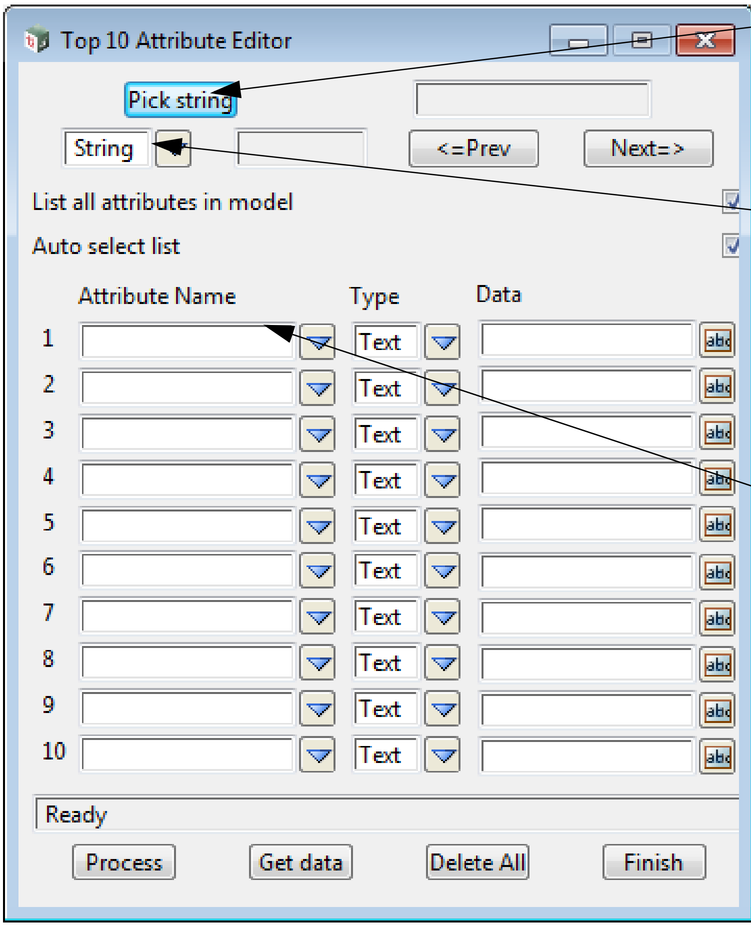
14.18.1 Attribute Editor

On selecting the **Attribute editor**, the **Top 10 Attribute Editor** panel is displayed.

This option displays ten (10) attributes of type string, vertex or segment for a selected string.

This option was written before the **Strings =>Properties =>Attributes** editor existed and was mainly used for editing drainage data. For more documentation, go to 23.10.2 Attribute Editor.

Most of the detailed catchment data is stored within 12d as user defined attributes. These attributes are automatically created by 12d when required but you are free to change them or add more as desired. The attributes may be exported to a spreadsheet and edited and then imported back into 12d or edited inside 12d using this panel.



First Select **Pick** to select the string that contains the user attributes (the drainage string). The strings will be highlighted in white when they are selected.

All catchment data is store with the pits in drainage strings. To access the pit attributes, select the drop down icon and then select **Pit**. A circle will be drawn around the pit selected. **Next** and **Prev** will now move you from pit to pit.

Select the drop down icon and then select the **Attribute Name** from the list of existing user defined attributes. These attributes include all of the attributes in the model that the string exists in.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
pick string	button		

used to pick the initial string in a model

string-pit-pipe	choice box		**string, pit, pipe**

select the type of attribute to be displayed. Pit and pipe attributes are only available for drainage strings.

attribute name	input box		

3 top 10 attributes lists are maintained (pit, pipe and string). The attributes that you can select from are all of the attributes that exist on all of the strings in the model. If the attribute does not exist for the string/pit/pipe that you are displaying the ***data*** *field will display* ***Not found.***

type choice box **Text, Real, Integer**

for existing attributes this will display Text, Real or Integer.
When defining a new attribute select the type of data to be stored in the attribute

data input box

the data stored in the attribute is displayed/edited/created in this field.

<= prev button

move to next *string in the model*
pit on the string
pipe on the string

next => button

move to next *string in the model*
pit on the string
pipe on the string

process button

updates the attributes displayed in the dialogue.

Notes:

First select Pick to select the string that contains the user attributes. All catchment data is stored with the pits in drainage strings. The strings will be highlighted in white when they are selected.

To access the pit attributes LB this field then select Pit. A circle will be drawn around the pit selected.

LB the Attribute Name field and then select from the list of existing user defined attributes. These attributes include all of the attributes in the model that the string exists in. They may not be defined for the string you are editing. If the string does not have that attribute defined not found will be displayed in the Data field.

To change the value for the attribute enter the new value in the data field. If the attribute does not exist, deleting the not found text and adding data will create it. The following message will be displayed whenever you are creating a new attribute.

For more documentation on setting the attributes for this option, go to 23.10.2 Attribute Editor

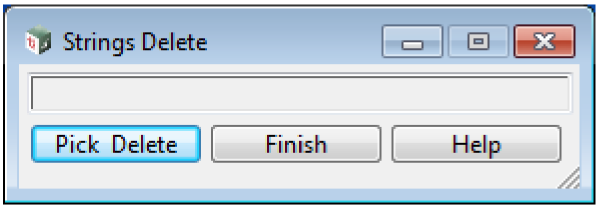
14.19 Delete

Position of option on menu: **Strings =>Delete**

Entire strings can be deleted from *12d Model* using the **delete** option.

Any number of strings can be deleted by successively selecting the strings.

On selecting the **delete** string option, the **strings delete** panel is displayed.



Field Description	Type	Defaults	Pop-Up
Pick & Delete	button		

any selected strings are deleted.
The cycle is terminated by clicking RB to raise the **pick ops** *menu and selecting* ***cancel*** *from it.*

14.20 Old String Create and Editors

14.20.1 Old

Position of option on menu: **Strings =>Create =>Old**

These old options are the string creation options that were available before *12d Model 8* was released.

The **Create** option is used to produce new strings. If a string already exists, the **Editor** option is used to modify it.

The **Create** walk-right menu contains options to create 2d, 3d, 4d, alignment, super alignment, circles, arcs, feature, pipe, polyline, super and text strings.

Each string type has its own special information so by choosing the appropriate type to be created, only information needed for that type of string is asked for.

Once a string has been created, it can be converted to most of the other string types using the string **Convert** option.

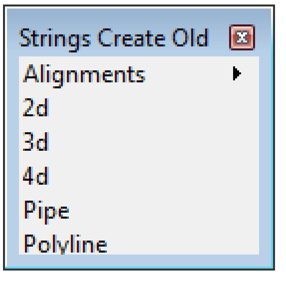
To create a new string of the same type and with similar header information as an existing string, the **same as** option is selected from the **String Create** menu.

Notes

1. Depending on its breakline (point-line) type, a string with default style ("1") is displayed with crosses at each of its vertex (point type) or with straight lines joining the vertices (line type).

   Linestyles can be defined that drawn lines between the points even though the breakline type is **point**.
2. The string vertices are also called intersection points (IP's).

The **String Create Old** walk-right menu is



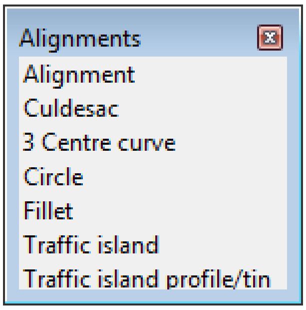
The string creation process is similar for each string type and for editing strings as well.

For the option *Alignments*, go to 14.20.1.1 Alignments
2d 14.20.1.2 Create - 2d (pre V8)
3d 14.20.1.3 Create - 3d (pre V8)
4d 14.20.1.4 Create - 4d (Pre V8)
Pipe 14.20.1.5 Create - Pipe (Pre V8)
Polyline 14.20.1.6 Create - Polyline (Pre V8)

14.20.1.1 Alignments

Position of menu: **Strings =>Create =>Old =>Alignments**

The **Alignment** walk-right menu contains options to create an alignment string, plus quick methods of creating an alignment string with horizontal geometry already created for culdesacs, circles and fillets.



create alignment
create culdesac
create three centred curve
create alignment forming a circle
create alignment fillet
create alignment for traffic island
create traffic island profile/tin

For the option *Alignment*, go to 14.20.1.1.1 Create - Alignment.
Culdesac 14.20.1.1.2 Create - Culdesac
3 centre curve 14.20.1.1.3 Create - Three centre curve
Circle 14.20.1.1.4 Create - Circle
Fillet 14.20.1.1.5 Create - Fillet
Traffic island 20.8.9.1 Create Traffic Island
Traffic island profile/tin 20.8.9.2 Traffic Island Profile/Tin

14.20.1.1.1 Create - Alignment

Position of option on menu: **Strings =>Create =>Old =>Alignments =>Alignment**

An alignment string is defined by specifying both its horizontal and vertical geometry.

The horizontal geometry consists of a series of (x,y) points (called horizontal intersection points) with circular curves and transition spirals applied to the intersection points.

Vertical geometry also consists of a series of points but they are defined with respect to the plan length of the string (chainage) and height. Hence, the vertical geometry is defined by a series of (chainage,height) points (called vertical intersection points) and either parabolic or circular curves applied to the vertical intersection points.

The horizontal geometry is defined in a plan view and the vertical geometry in a section view.

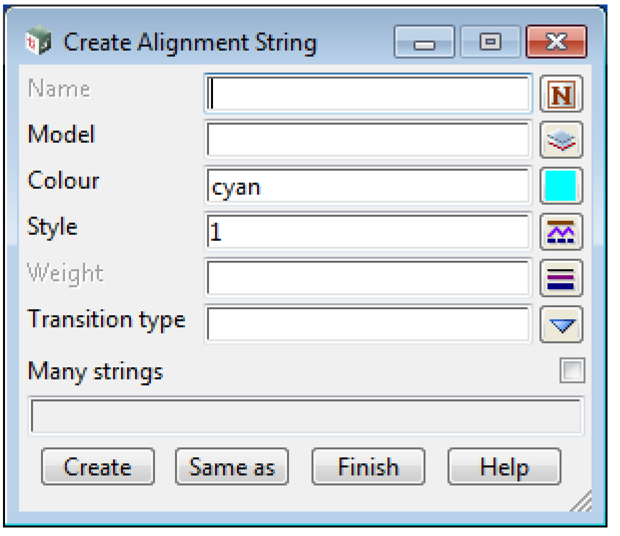
Alignment strings are often referred to as “centre-line strings” because road centre-lines are common examples of alignment strings.

Note vertical geometry can only be added to an existing string.

The Many strings tick-box is used when more than one string of the same type is to be created, If many strings is set to *tick*, when the current string creation is **Finished** or **Quit**, a new **Create Alignment String** panel is placed on the screen with the same information in it as the string just created. If any of the information needs to be modified for the new string, simply change it in the **Create Alignment String** panel fields before selecting the **Create** button for the new string. Hence a new string of the same type can be created without going back to the **Create** menu.

The Same as button is used to obtain information from and existing string (not necessarily of the same type) and pipe it into the name, colour, model, style breakline type and height field of the **Create Alignment String** panel.

On selecting the **Alignment string** option, the **Create Alignment String** panel is displayed.



To create a new alignment string, the name, model, colour, style and transition spiral type of the new string are entered into the appropriate fields and the **Create** button selected.

The new fields and buttons used in the **Create Alignment String** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input		

the name of the new string.

Model	input		available models

name of the model that the new string is in.

Colour	input	default colour	available colours

the colour of the new string.

Style	input	1	available line styles

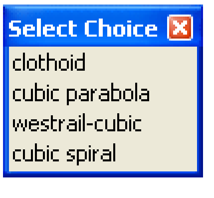
line style of the string.

Weight	input	0	

thickness of the string.

Transition type	input	clothoid	clothoid, cubic parabola westrail-cubic, cubic spiral

transition (for example clothoid spiral) to be used for this alignment string. For more information see 39.2.7.5 Transitions and Spirals File.



Many strings	tick		

if ticked then after the current string is finished, a new create panel is placed on the screen with all the same values for the panel fields as the current string.

Create	button		

After the **create** *button is chosen, the* **alignment edit** *menu and* **alignment edit info** *panel are* *displayed.*

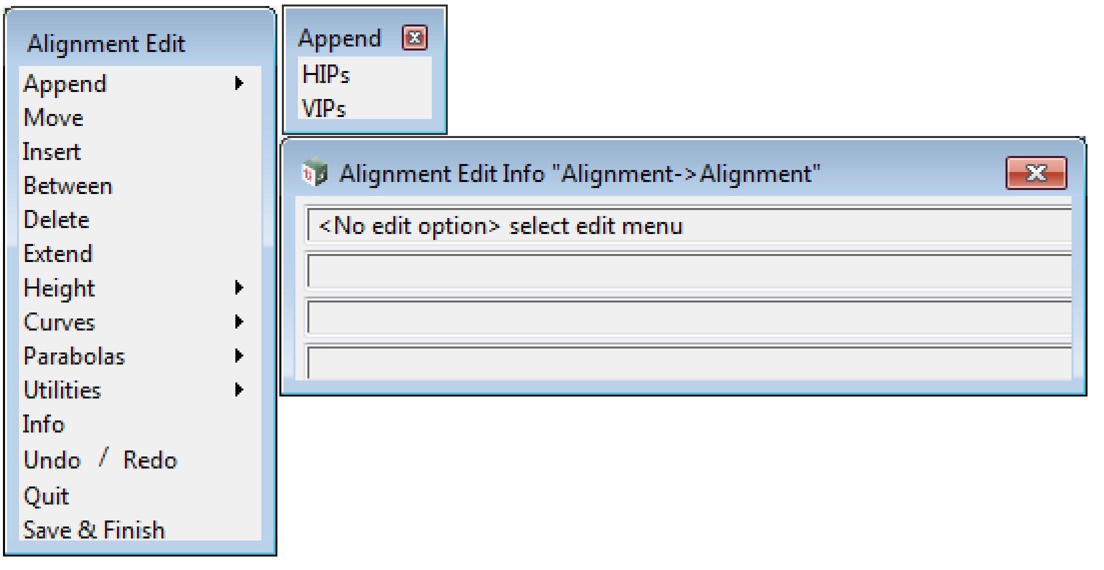
Same as button

After the **same as** *button is chosen, another string is selected and information about it is used for the fields in this panel.*

14.20.1.1.1.1 Alignment Edit

Selecting the create button in the **create alignment string** panel the **alignment edit** menu and **alignment edit info** panel are placed on the screen.

The **Alignment edit** menu and **Alignment edit info** panel for an alignment string are



To begin creating an alignment string, the user must select the **Append=>HIPs** option from the **Alignment Edit** menu and start placing points in a plan view.

To create a new alignment string, select the **Append** => **HIPs** option from the **Alignment Edit** menu.

The **Append** =>**HIPs** option is used to add horizontal intersection points to either end of an existing alignment string, or in the case of a new string, places the 1st point and then begins appending points to the 1st point.

For all alignment strings, a cross is then drawn in each plan view that the string's model is on, and the cross follows the cursor around the screen. If the string's model is not added to any plan view, the model is automatically added to all plan views.

After the cross is on the screen (moving with the cursor), clicking LB and accepting with MB selects the 1st vertex of the string (using the appropriate snaps).

The string is then drawn from the 1st vertex to the cursor position, which represents the second vertex of the string. Clicking LB and accepting with MB selects the second string vertex and the process repeats for subsequent string vertices.

Now that the string is created, all the edit option on the string's **Alignment Edit** menu are usable.

The options in the **Alignment Edit** menu are not only used for placing the initial vertices of the string, but for editing the string once it is created. Since the **Append** and other options in the **Alignment edit** menu are identical to the options used when editing an existing string, they will be discussed in detail in the string **Editor** section.

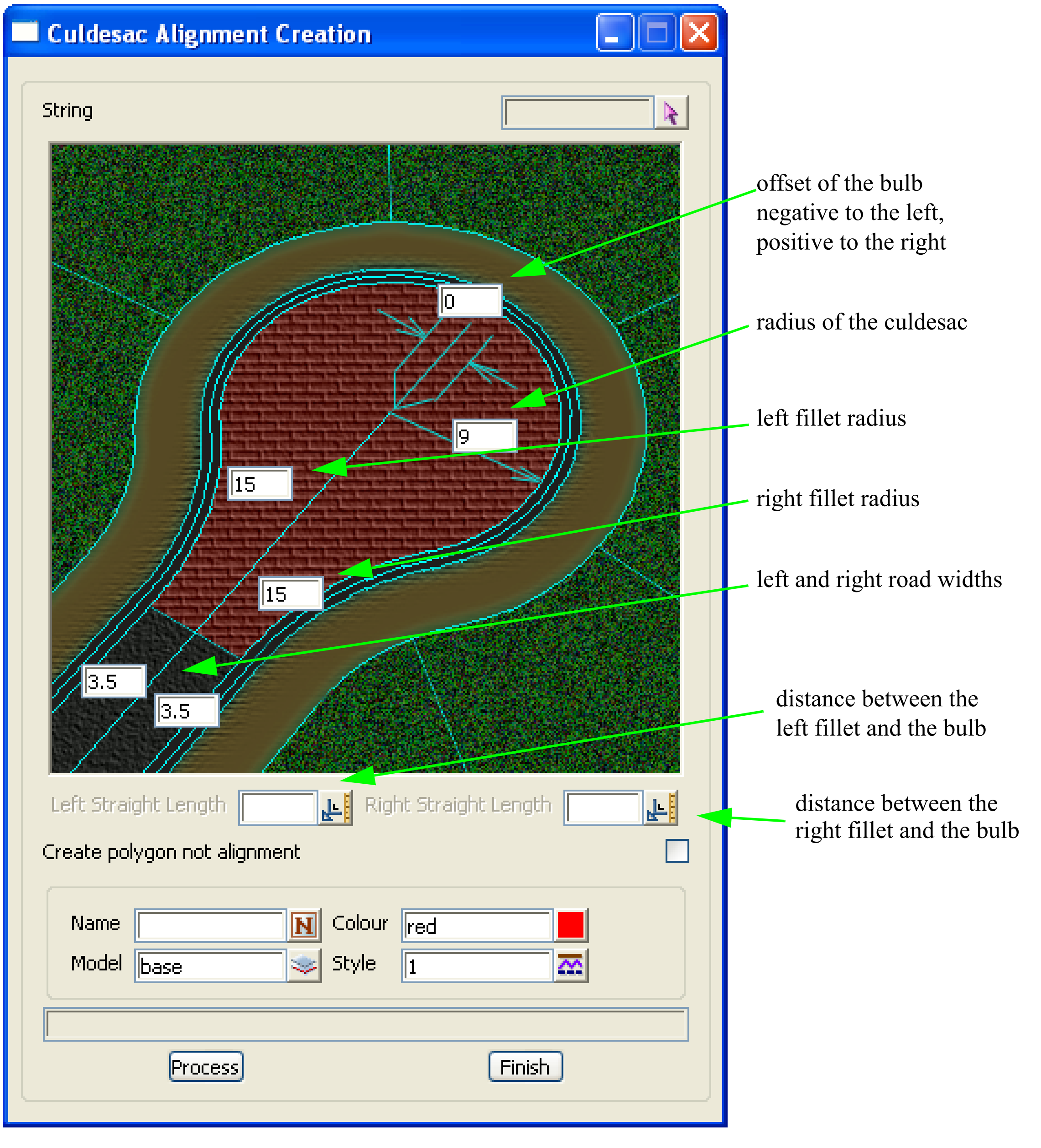
For full information on the Alignment Editor, go to the section 14.20.2 Alignment Edit

14.20.1.1.2 Create - Culdesac

Position of option on menu: **Strings =>Create =>Old =>Alignments =>Culdesac**

Create an alignment string for a culdesac for user defined road widths, culdesac radius and offset. No vertical geometry is defined.

On selecting the **Culdesac** option, the **Culdesac Alignment Creation** panel is displayed.



The position of the panel fields indicate what the values are for. Tool tips appear when the cursor is passed over the panel.fields.

The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Offset	input box		

offset of the centre of the culdesac bulb. Use a negative value if the offset is to the left, positive if the offset is to the right.

Radius input box

radius of the culdesac bulb

Left/Right fillet radii input box

radii for the left and right fillets from the roads to the culdesac bulb.

Left/Right road widths input box

left and right widths of the road.

Name/Model/Colour/Style input box

name/model/colour/style of the created alignment string.

Select string select

select with direction the string to create the culdesac for.

Preview string select

draw the culdesac with the given parameters in the panel draw box.

Process button

create the alignment string

Undo button

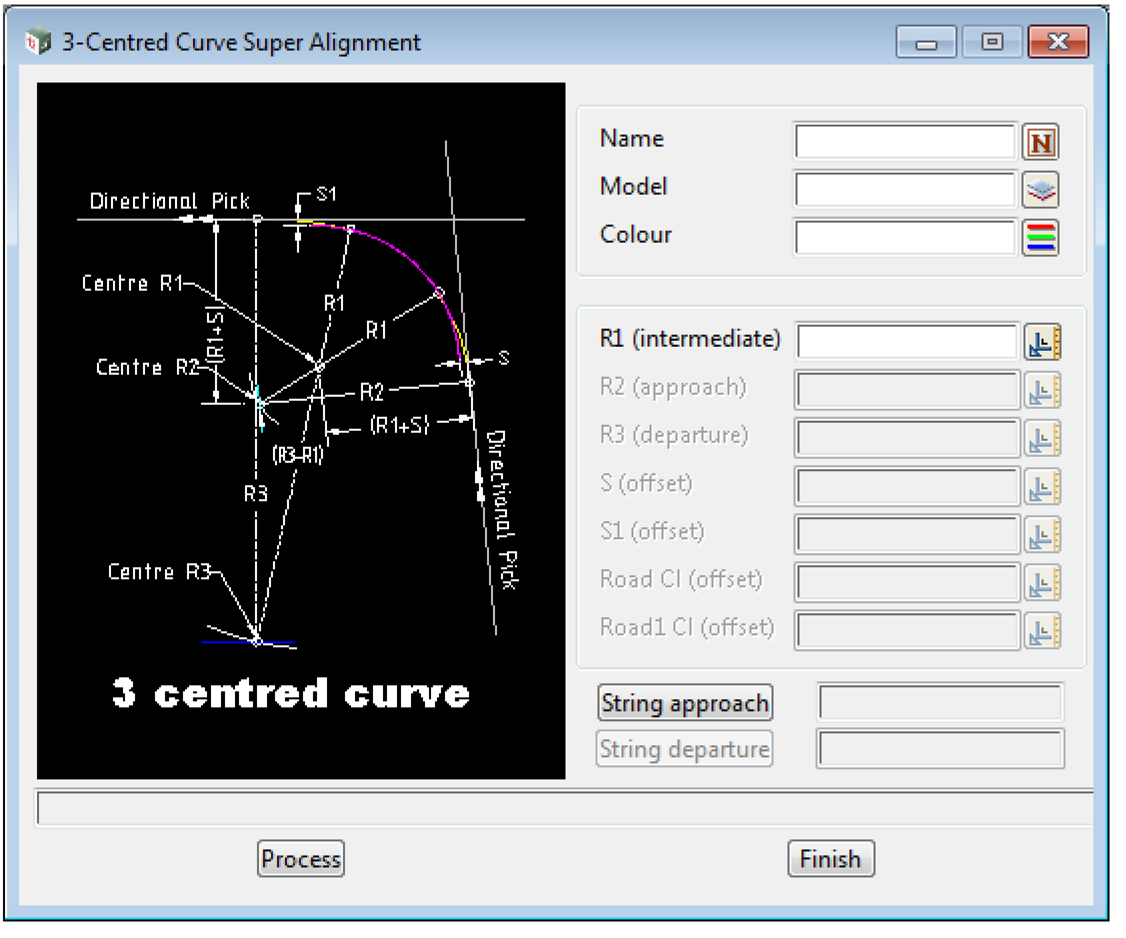
undo the last alignment string created whilst the panel has been active.

14.20.1.1.3 Create - Three centre curve

Position of option on menu: **Strings =>Create =>Old =>Alignments =>3 Centre Curve**

Create an alignment string for a three centred curve for user defined parameters and selected strings.

On selecting the **3 centre curve** option, the **3 centred curve construction** panel is displayed.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Name/Model/Colour/Style input box

name/model/colour/style of the created alignment string.

R1 (intermediate) input box

radius of the intermediate curve. If a value is entered and <enter> typed, values for R2, R3, S and S1 are computed and place in the appropriate fields

R2 (approach) input box

radius of the approach curve.

R3 (departure) input box

radius of the departure curve.

S (offset) input box

approach offset.

S1 (offset) input box

departure offset.

Kerb line approach string select

select the approach string

Kerb line departure string select

select the departure string

Process button

create the alignment string

Undo button

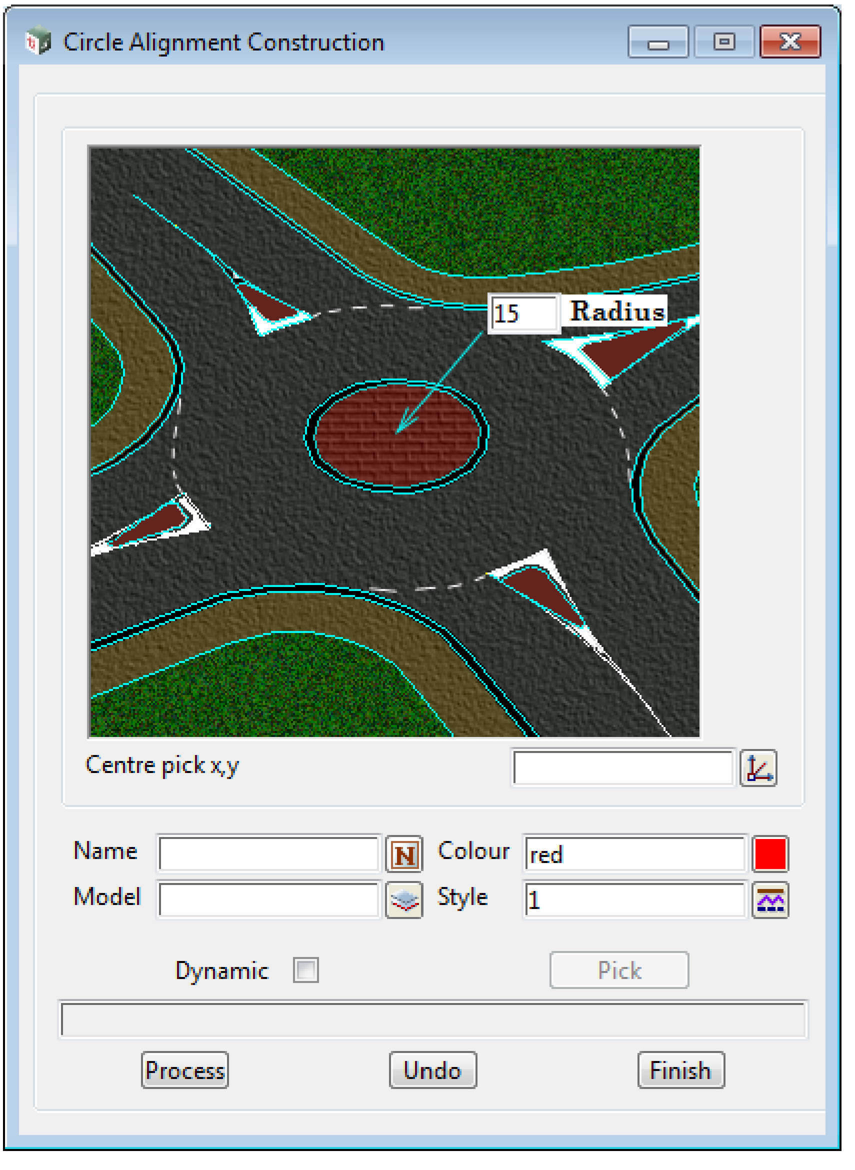
undo the last alignment string created whilst the panel has been active.

14.20.1.1.4 Create - Circle

Position of option on menu: **Strings =>Create =>Old =>Alignments =>Circle**

Create an alignment string for a circle with a for user radius and selected centre or with a dynamic mode set, selected centre and interactively positioned circumference.

On selecting the **curve** option, the **circle alignment construction** panel is displayed.



The diagram indicates what the panel fields are for. Tool tips appear when the cursor is passed over the panel.fields.

The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Radius	input box		

the radius of the circle.

Centre pick x,y	x,y,z box		

if Dynamic *in not ticked, select the centre of the circle.*

Name/Model/Colour/Style	input box		

name/model/colour/style of the created alignment string.

Dynamic tick box

if not ticked, then the Centre pick x,y *and* radius *fields are used to define the circle.*

if ticked, then the Pick *button is used to select the centre and then the cursor is taken to be on the circle to dynamically define the radius.*

Pick button

pick the circle centre when in dynamic mode.

Process button

create the alignment string.

Undo button

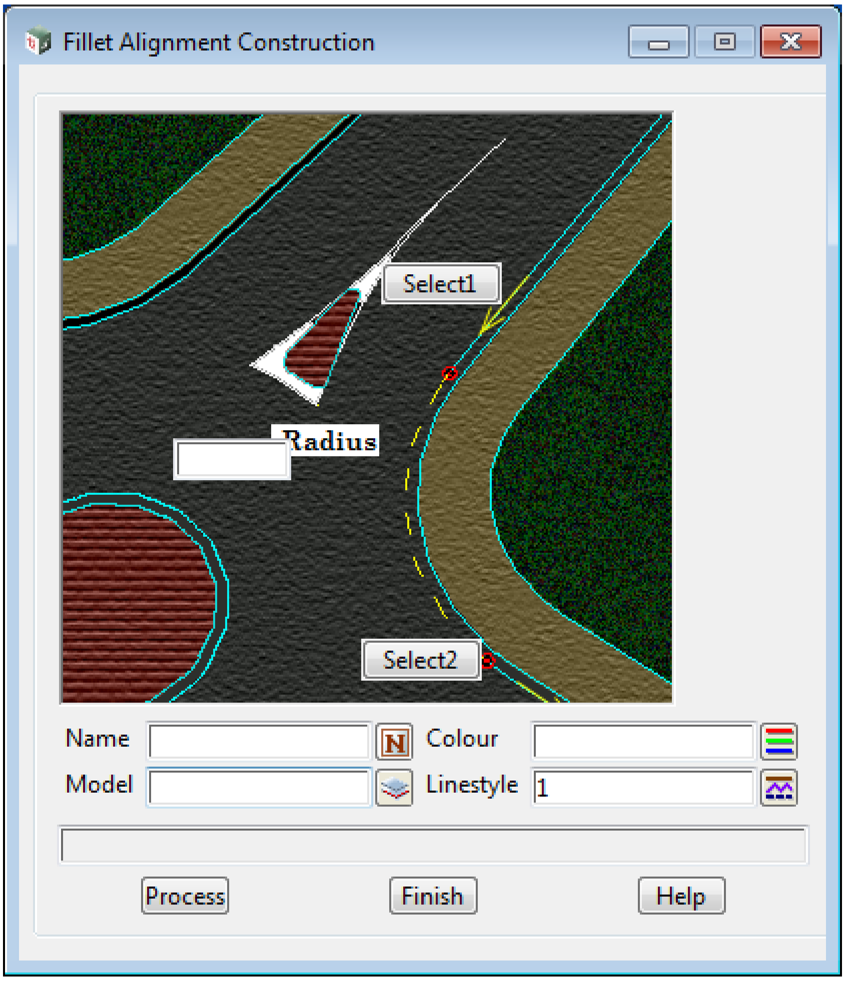
undo the last alignment string created whilst the panel has been active.

14.20.1.1.5 Create - Fillet

Position of option on menu: **Strings =>Create =>Old =>Alignments =>Fillet**

Create an alignment string for a fillet with a for user radius and selected in and out strings.

On selecting the **fillet** option, the **fillet alignment construction** panel is displayed.



The diagram indicates what the panel fields are for. Tool tips appear when the cursor is passed over the panel.fields.

The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Select 1	string select		

select the approach string. Pick with direction.

Radius	input box		

the radius of the fillet.

Select 2	string select		

select the departure string. Pick with direction.

Name/Model/Colour/Style	input box		

name/model/colour/style of the created alignment string.

Process	button		

create the alignment string.

Undo button

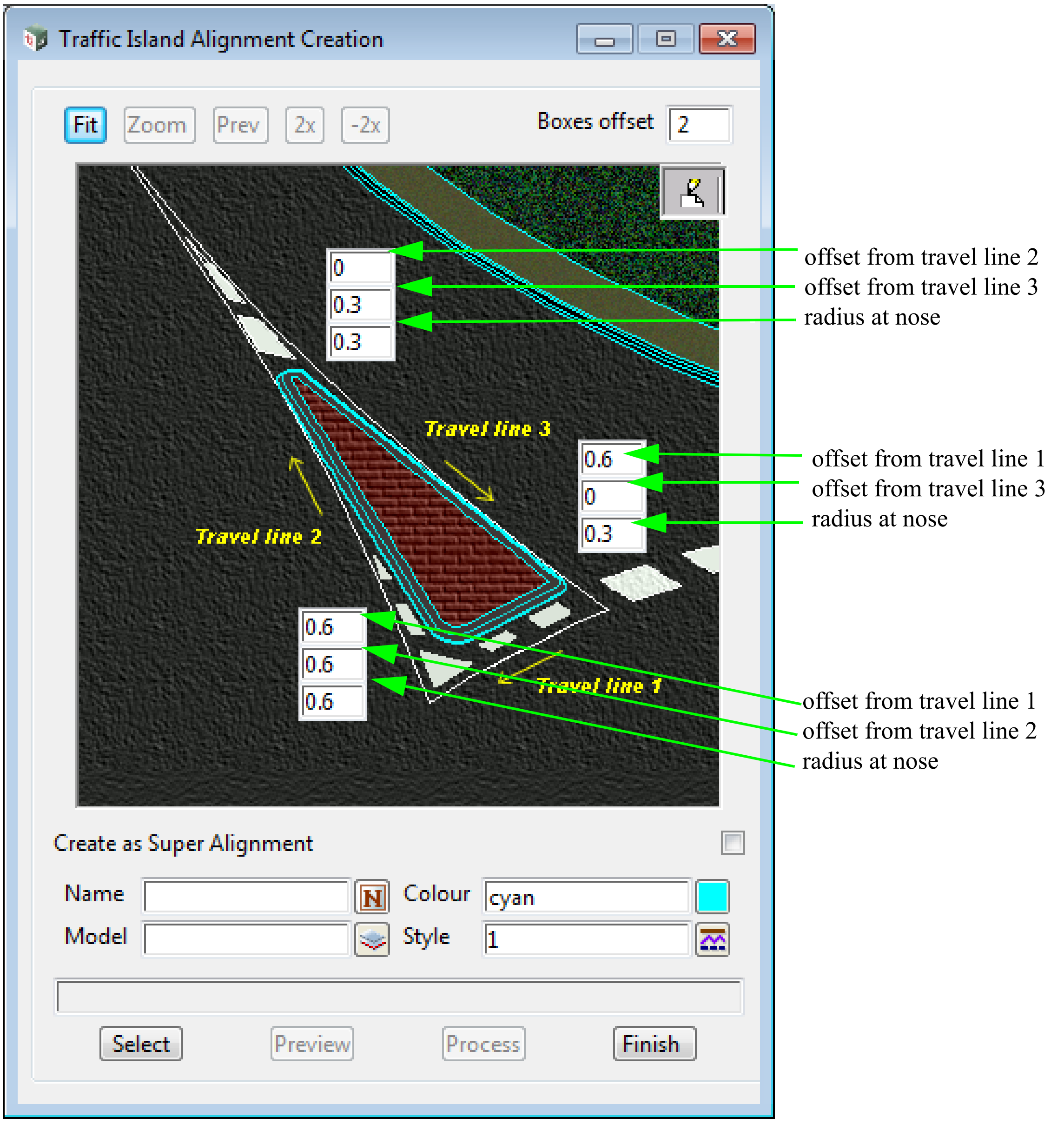
undo the last alignment string created whilst the panel has been active.

14.20.1.1.6 Create - Traffic Island

Position of option on menu: **Strings =>Create =>Old =>Alignments =>Traffic Island**

Create an alignment string for a traffic island between user selected strings and with user defined offsets from the strings and nose radii. No vertical geometry is defined.

On selecting the **traffic island** option, the **traffic island alignment creation** panel is displayed.



The position of the panel fields indicate what the values are for. Tool tips appear when the cursor is passed over the panel.fields.

The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Offset from travel lines	input box		

offset distance from the selected travel lines.

Radius at nose	input box		

radius of the corner of the traffic island between the selected travel lines.

Name/Model/Colour/Style input box

name/model/colour/style of the created alignment string.

Select strings select

select in order and with direction, the three strings to use in creating the traffic island. Terminate by clicking RB and selecting cancel from the pick ops menu.

Preview string select

draw the traffic island with the given parameters in the panel draw box.

Process button

create the alignment string

Undo button

undo the last alignment string created whilst the panel has been active.

14.20.1.2 Create - 2d (pre V8)

Position of option on menu: **Strings =>Create =>Old =>2d**

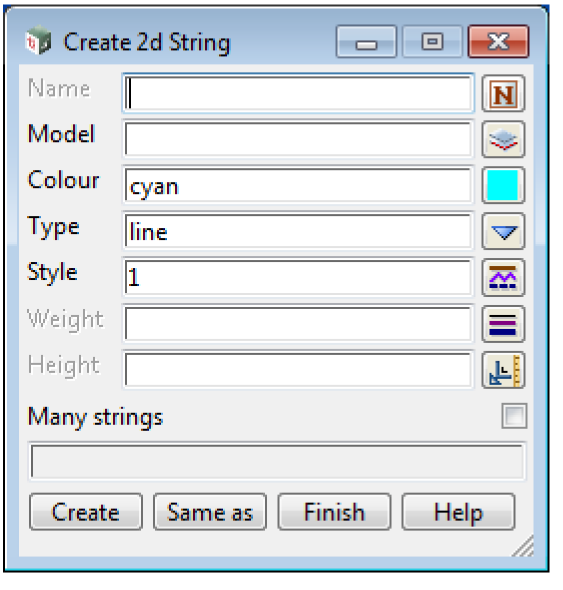
A 2d string consists of a series of (x,y) vertices all with the same z-value (height). 2d strings are often referred to as “contour strings” because contours are the most common example of a string with a constant height.

The Many strings tick-box is used when more than one string of the same type is to be created, If many strings is set to *tick*, when the current string creation is **Finished** or **Quit**, a new **Create 2d String** panel is placed on the screen with the same information in it as the string just created. If any of the information needs to be modified for the new string, simply change it in the **Create 2d String** panel fields before selecting the **Create** button for the new string. Hence a new string of the same type can be created without going back to the **Create** menu.

The Same as button is used to obtain information from and existing string (not necessarily of the same type) and pipe it into the name, colour, model, style breakline type and height field of the **Create 2d String** panel.

From *12d Model 8* onwards, the default is to create *2d super strings* rather than the *2d strings* used up to *12d Model 7*. A *2d super string* is a *super string* with *Constant height*. That is, there is only one z-value for the entire string. The advantage is using a *2d super string* is that all the CAD options will work for it and it can be given point id’s for setout.

On selecting the **2d string** option, the **create 2d string** panel is displayed.



The default values for the panel fields are taken from the *CAD Controlbar (see* 15.1 Controlbars*)*

To create a *new* 2d string, the name, colour, model, point-line type and height of the new string are entered into the appropriate fields and the **Create** button selected.

The new fields and buttons used in the **create 2d string** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input		
the name of the new string.			
Model	input		available models
name of the model that the new string is in.			
Colour	input	default colour	available colours
the colour of the new string.			
Type	input	line	line, point
breakline type (point-line type) of the string.			

Style	input	1	available line styles

line style of the string.

Weight	input	0

thickness of the string.

Height input

height (z-value) of the entire string.

Many strings tick

if ticked then after the current string is finished, a new create panel is placed on the screen with all the same values for the panel fields as the current string.

Create button

After the **create** *button is chosen, the* **2d edit** *menu and* **2d edit info** *panel are displayed.*

Same as button

After the **same as** *button is chosen, another string is selected and information about it is used for the fields in this panel.*

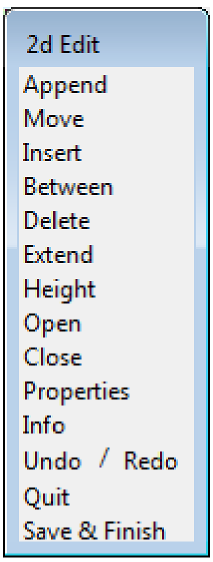
Finish button

end the option, ***don't*** *proceed to the edit stage.*

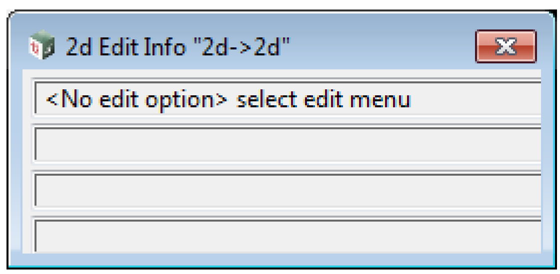
14.20.1.2.1 2d Edit for New String

On selecting the **Create** button in the **Create 2d String** panel the **2d edit** menu and **2d Edit Info** panel are placed on the screen.

The **2d edit** menu for a 2d string is and the **2d edit info** panel is



append or prepend an IP
move an IP
insert an IP
insert IP on line joining IPs
delete an IP
extend an IP
modify the string's z-value
open a closed string
close a string
bring up Properties panel
toggle edit info panel
undo/redo
quit the create
finish the 2d create



To create a new 2d string, select the **Append** option from the **2d Edit** menu.

The **Append** option is used to add points to either end of an existing string, or in the case of a new string, places the 1st point and then begins appending points to the 1st point.

For all 2d strings, a cross is then drawn in each plan view that the string's model is on, and the cross follows the cursor around the screen. If the string's model is not added to any plan view, the model is automatically added to all plan views.

After the cross is on the screen (moving with the cursor), clicking LB and accepting with MB selects the 1st vertex of the string (using the appropriate snaps).

The string is then drawn from the 1st vertex to the cursor position, which represents the 2nd vertex of the string. Clicking LB and accepting with MB selects the second string vertex and the process repeats for subsequent string vertices.

Now that the string is created, all the edit option on the string's **2d Edit** menu are usable.

The options in the **2d Edit** menu are not only used for placing the initial vertices of the string, but for editing the string once it is created. Since the **Append** and other options in the **2d Edit** menu are identical to the options used when editing an existing string, they will be discussed in detail in the string **Editor** section.

For full information in the 2d Editor, go to the section 14.20.3 2d Edit - Old

14.20.1.3 Create - 3d (pre V8)

Position of option on menu: **Strings =>Create =>Old =>3d**

A 3d string consists of a series of (x,y,z) points.

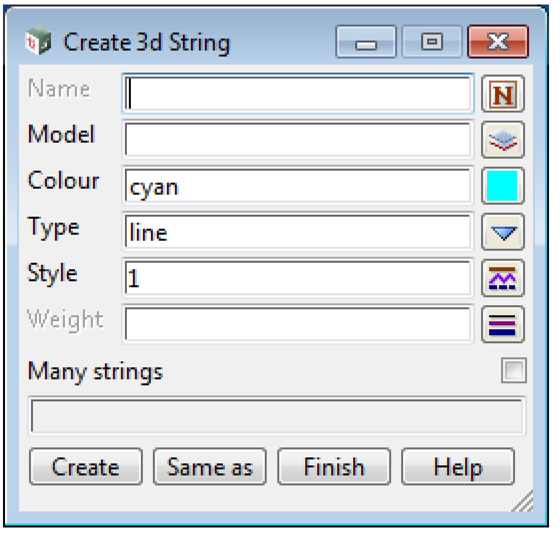
The difference between a 2d and a 3d string is that for a 3d string, the z-value (height) can vary at each string point, whereas a 2d string has a constant height for the entire string. Hence a 2d string is simply a special case of a 3d string where the heights at all the points are the same.

Creating and editing a 3d string is very similar to a 2d string. The only major difference is that a height is required at each string point.

The Many strings tick-box is used when more than one string of the same type is to be created, If many strings is set to *tick*, when the current string creation is **Finished** or **Quit**, a new **Create 3d String** panel is placed on the screen with the same information in it as the string just created. If any of the information needs to be modified for the new string, simply change it in the **Create 3d String** panel fields before selecting the **Create** button for the new string. Hence a new string of the same type can be created without going back to the **Create** menu.

The Same as button is used to obtain information from and existing string (not necessarily of the same type) and pipe it into the name, colour, model, style breakline type and height field of the **Create 3d String** panel.

On selecting the **3d string** option, the **create 3d string** panel is displayed.



To **create** a new 3d string, the name, colour, model and point-line type of the new string are entered into the appropriate fields and the **create** button selected.

To create a new 2d string, the name, colour, model, point-line type and height of the new string are entered into the appropriate fields and the **Create** button selected.

The new fields and buttons used in the **create 3d string** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input		
the name of the new string.			
Model	input		available models
name of the model that the new string is in.			
Colour	input	default colour	available colours
the colour of the new string.			
Type	input	line	line, point
breakline type (point-line type) of the string.			

Style	input	1	available line styles

line style of the string.

Weight	input	0

thickness of the string.

Many strings tick

if ticked then after the current string is finished, a new create panel is placed on the screen with all the same values for the panel fields as the current string.

Create button

After the **create** *button is chosen, the* **3d edit** *menu and* **3d edit info** *panel are displayed.*

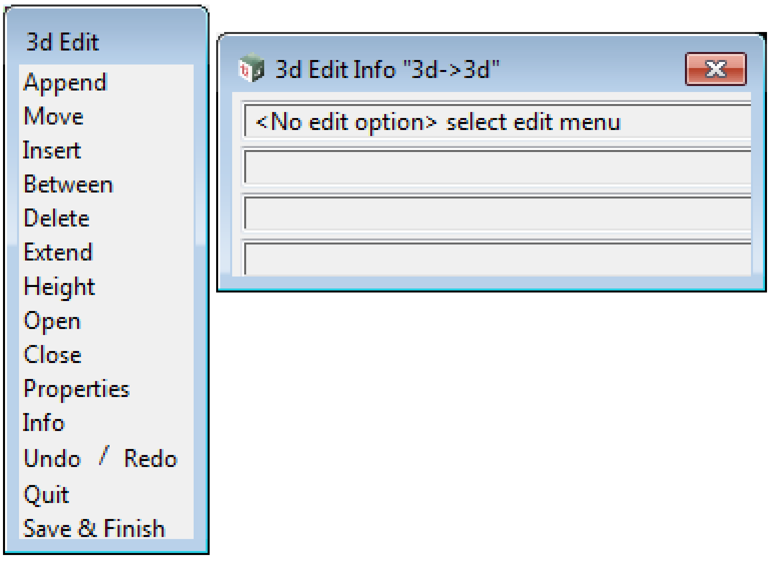
Same as button

After the **same as** *button is chosen, another string is selected and information about it is used for the fields in this panel.*

14.20.1.3.1 3d Edit

Selecting the **create** button in the **create 3d string** panel the **3d edit** menu and **3d edit info** panel are placed on the screen.

The **3d edit** menu and panel for a 3d string are



To create the new 3d string, select the **Append** option from the **3d Edit** menu.

The **Append** option is used to add points to either end of an existing string, or in the case of a new string, places the 1st point and then begins appending points to the 1st point.

For all 3d strings, a cross is then drawn in each plan view that the string's model is on, and the cross follows the cursor around the screen. If the string's model is not added to any plan view, the model is automatically added to all plan views.

After the cross is on the screen (moving with the cursor), clicking LB and accepting with MB selects the 1st vertex of the string (using the appropriate snaps).

The string is then drawn from the 1st vertex to the cursor position, which represents the second vertex of the string. Clicking LB and accepting with MB selects the second string vertex and the process repeats for subsequent string vertices.

Now that the string is created, all the edit option on the string's **3d Edit** menu are usable.

The options in the **3d Edit** menu are not only used for placing the initial vertices of the string, but for editing the string once it is created. Since the **Append** and other options in the **3d edit** menu are identical to the options used when editing an existing string, they will be discussed in detail in the string **Editor** section.

For full information in the 3d Editor, go to the section 14.20.4 3d Edit - Old

14.20.1.4 Create - 4d (Pre V8)

Position of option on menu: **Strings =>Create =>4d**

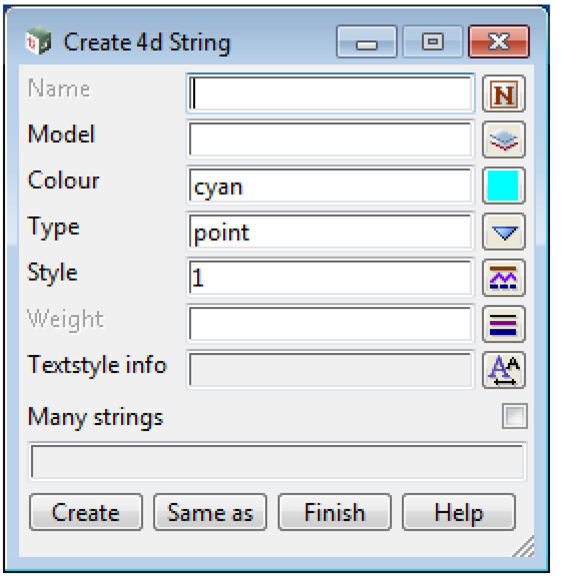
A 4d string consists of a series of (x,y,z) points and a text label at each point.

Creating and editing a 4d string is very similar to a 3d string.- the major difference is that a height and a text label is required at each string point.

The Many strings tick-box is used when more than one string of the same type is to be created, If many strings is set to *tick*, when the current string creation is **Finished** or **Quit**, a new **Create 4d String** panel is placed on the screen with the same information in it as the string just created. If any of the information needs to be modified for the new string, simply change it in the **Create 4d String** panel fields before selecting the **Create** button for the new string. Hence a new string of the same type can be created without going back to the **Create** menu.

The Same as button is used to obtain information from and existing string (not necessarily of the same type) and pipe it into the name, colour, model, style breakline type and height field of the **Create 4d String** panel.

On selecting the **4d string** option, the **create 4d string** panel is displayed.



To **create** a new 4d string, the name, colour, model, point-line type, text size and units, offset, justification and angle for the new string are entered into the appropriate fields and the **create** button selected.

The new fields and buttons used in the **create 4d string** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input		
the name of the new string.			
Model	input		available models
name of the model that the new string is in.			
Colour	input	default colour	available colours
the colour of the new string.			
Type	input	line	line, point
breakline type (point-line type) of the string.			
Style	input	1	available line styles
line style of the string.			

Weight input 0

thickness of the string.

Textstyle info input

textstyle information.

Many strings tick

if ticked then after the current string is finished, a new create panel is placed on the screen with all the same values for the panel fields as the current string.

Create button

After the **create** *button is chosen, the* **4d edit** *menu and* **4d edit info** *panel are displayed.*

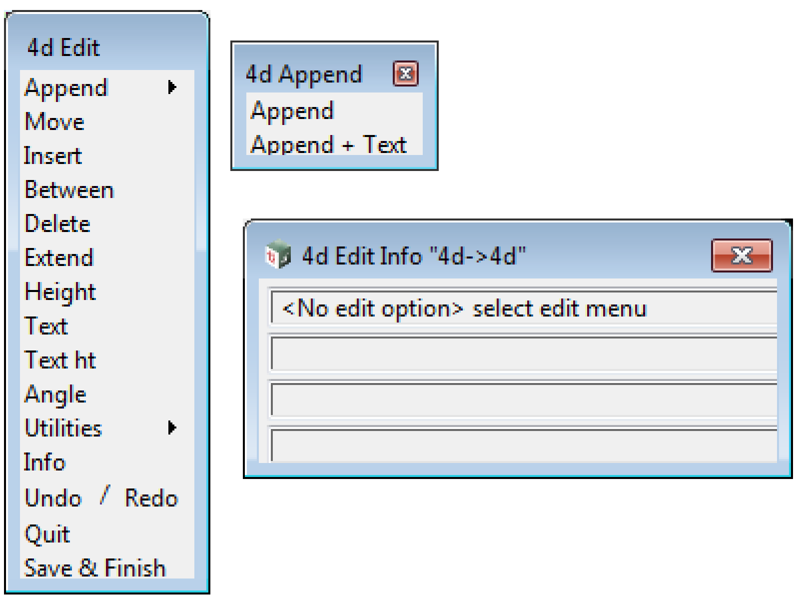
Same as button

After the **same as** *button is chosen, another string is selected and information about it is used for the fields in this panel.*

14.20.1.4.1 4d Edit

Selecting the **create** button in the **create 4d string** panel the **4d edit** menu and **4d edit info** panel are placed on the screen.

The **4d edit** menu and panel for a 4d string are



To create a new 4d string, the user must select one of the two Append options on the **Append** walk-right of the **4d Edit** menu which has options to simply create points or to create points and text at the points.

The **Append** and **Append + text** options are used to add points and text (in a plan view) to either end of an existing string, or in the case of a new string, places the 1st point and text and then begins appending points and text to the 1st point.

For all 4d strings, a cross is then drawn in each plan view that the string's model is on, and the cross follows the cursor around the screen. If the string's model is not added to any plan view, the model is automatically added to all plan views.

After the cross is on the screen (moving with the cursor), clicking LB and accepting with MB selects the 1st vertex of the string (using the appropriate snaps).

If **Append + text** was selected, the user will then be prompted for the text to be placed at that points.

The string and text is then drawn from the 1st vertex to the cursor position, which represents the second vertex of the string. Clicking LB and accepting with MB selects the second string vertex and the process repeats for subsequent string vertices.

Now that the string is created, all the edit option on the string's **4d Edit** menu are usable.

The options in the **4d Edit** menu are not only used for placing the initial vertices of the string, but for editing the string once it is created. Since the **Append** and other options in the **4d edit** menu are identical to the options used when editing an existing string, they will be discussed in detail in the string **Editor** section.

For full information in the 4d Editor, go to the section 14.20.5 4d Edit - Old

14.20.1.5 Create - Pipe (Pre V8)

Position of option on menu: **Strings =>Create =>Old =>Pipe**

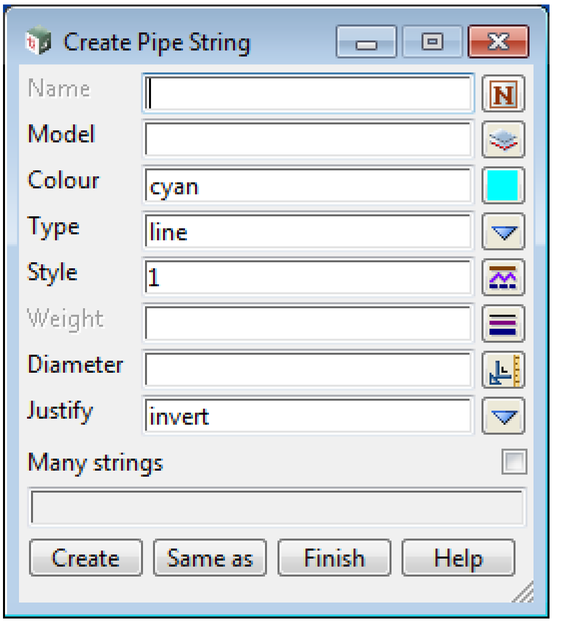
A pipe string is the same as a 3d string except that the pipe string also has a diameter hence the options for creating and editing a pipe string are almost the same as for a 3d string.

The main difference is that there are no options to open and close a pipe string and there is an additional option for modifying the pipe diameter.

The Many strings tick-box is used when more than one string of the same type is to be created, If many strings is set to *tick*, when the current string creation is **Finished** or **Quit**, a new **Create Pipe String** panel is placed on the screen with the same information in it as the string just created. If any of the information needs to be modified for the new string, simply change it in the **Create Pipe String** panel fields before selecting the **Create** button for the new string. Hence a new string of the same type can be created without going back to the **Create** menu.

The Same as button is used to obtain information from and existing string (not necessarily of the same type) and pipe it into the name, colour, model, style breakline type and height field of the **Create Pipe String** panel.

On selecting the **Pipe string** option, the **Create Pipe String** panel is displayed.



To **create** a new pipe string, the name, colour, model, point-line type and diameter of the new string are entered into the appropriate fields and the **create** button selected.

The new fields and buttons used in the **create pipe string** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input		
the name of the new string.			
Model	input		available models
name of the model that the new string is in.			
Colour	input	default colour	available colours
the colour of the new string.			
Style	input	1	available line styles
line style of the string.			

Weight input 0

thickness of the string.

Diameter input 0

diameter of the pipe

Justify input invert invert, centre, overt

justification of the pipe with respect to the co-ordinates given for the pipe string.

Many strings tick

if ticked then after the current string is finished, a new create panel is placed on the screen with all the same values for the panel fields as the current string.

Create button

After the **create** *button is chosen, the* **pipe edit** *menu and* **pipe edit info** *panel are displayed.*

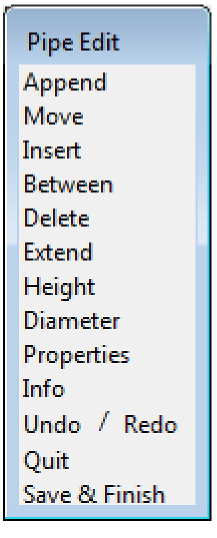
Same as button

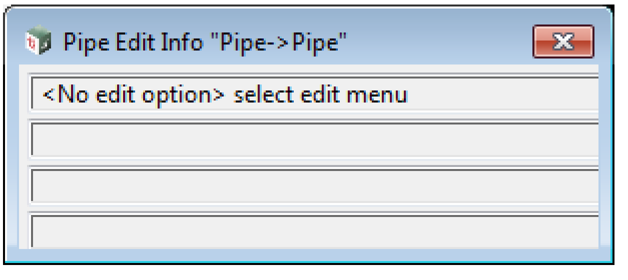
After the **same as** *button is chosen, another string is selected and information about it is used for the fields in this panel.*

14.20.1.5.1 Pipe Edit

On selecting the **Create** button in the **Create Pipe String** panel the **Pipe Edit** menu and **Pipe Edit info** panel are placed on the screen.

The **Pipe Edit** menu and panel for a pipe string are





To create a new pipe string, select the **Append** option from the **Pipe Edit** menu.

The **Append** option is used to add points to either end of an existing string, or in the case of a new string, places the 1st point and then begins appending points to the 1st point.

For all pipe strings, a cross is then drawn in each plan view that the string's model is on, and the cross follows the cursor around the screen. If the string's model is not added to any plan view, the model is automatically added to all plan views.

After the cross is on the screen (moving with the cursor), clicking LB and accepting with MB selects the 1st vertex of the string (using the appropriate snaps).

The string is then drawn from the 1st vertex to the cursor position, which represents the second vertex of the string. Clicking LB and accepting with MB selects the second string vertex and the process repeats for subsequent string vertices.

Now that the string is created, all the edit option on the string's **Pipe Edit** menu are usable.

The options in the **Pipe Edit** menu are not only used for placing the initial vertices of the string, but for editing the string once it is created. Since the **Append** and other options in the **Pipe Edit** menu are identical to the options used when editing an existing string, they will be discussed in detail in the string **Editor** section.

For full information in the Pipe Editor, go to the section 14.20.6 Pipe Edit - Old

14.20.1.6 Create - Polyline (Pre V8)

Position of option on menu: **Strings =>Create =>Old =>Polyline**

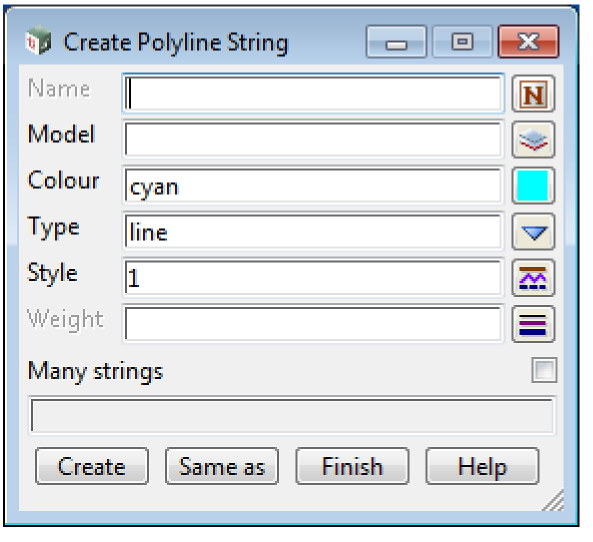
A polyline string is similar to a 3d string except that it can have either straight lines or arcs joining the (x,y,z) points of the string. The arcs are plan arcs with possibly a different z at either end and the z values are linearly interpolated between the end points. Hence in a long section, the end points are joined by a straight lines for both line and arc segments of the polyline.

Creating and editing a polyline string is very similar to a 3d string. The only major difference is that a radius is required at each string segment (a radius of 0 means no arc, just a straight line).

The Many strings tick-box is used when more than one string of the same type is to be created, If many strings is set to *tick*, when the current string creation is **Finished** or **Quit**, a new **Create Polyline String** panel is placed on the screen with the same information in it as the string just created. If any of the information needs to be modified for the new string, simply change it in the **Create Polyline String** panel fields before selecting the **Create** button for the new string. Hence a new string of the same type can be created without going back to the **Create** menu.

The Same as button is used to obtain information from and existing string (not necessarily of the same type) and pipe it into the name, colour, model, style breakline type and height field of the **Create Polyline String** panel.

On selecting the **polyline string** option, the **create polyline string** panel is displayed.



To **create** a new polyline string, the name, colour, model, point-line type and linestyle of the new string are entered into the appropriate fields and the **create** button selected.

The new fields and buttons used in the **create pipe string** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input		
the name of the new string.			
Model	input		available models
name of the model that the new string is in.			
Colour	input	default colour	available colours
the colour of the new string.			
Style	input	1	available line styles
line style of the string.			
Weight	input	0	
thickness of the string.			

Many strings tick

if ticked then after the current string is finished, a new create panel is placed on the screen with all the same values for the panel fields as the current string.

Create button

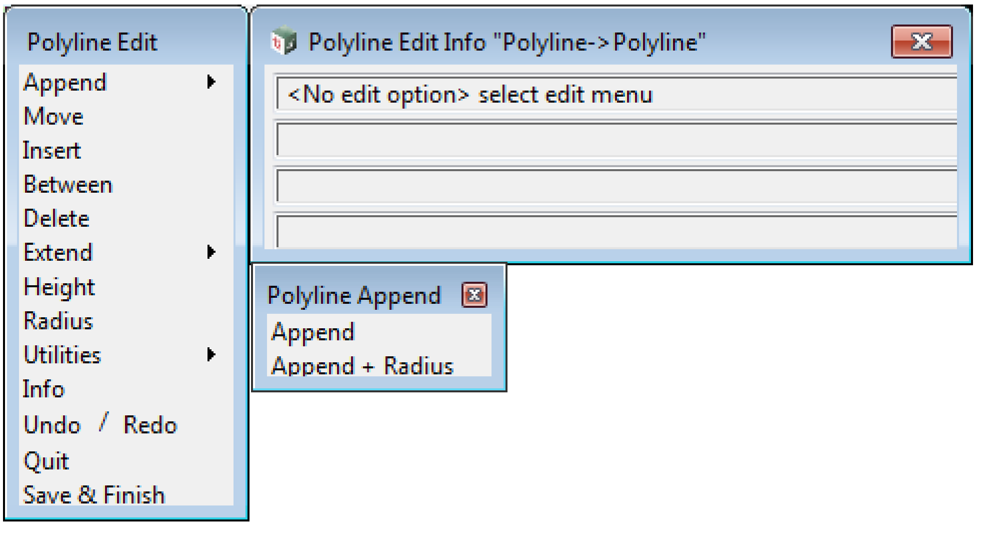
*After the **create** button is chosen, the **polyline edit** menu and **polyline edit info** panel are displayed.*

Same as button

*After the **same as** button is chosen, another string is selected and information about it is used for the fields in this panel.*

14.20.1.6.1 Polyline Edit

Selecting the **create** button in the **create polyline string** panel the **polyline edit** menu and **polyline edit info** panels are placed on the screen.



To create a new polyline string, the user must select one of the two append options (**Append** or **Append + radius**) on the **Append** walk-right of the **Polyline Edit** menu. The which **Append** option to simply creates points which are joined by a straight segment whereas the **Append + radius** create points with a radius for the segment.

The **Append** options are used to add points to either end of an existing string, or in the case of a new string, places the 1st point and then begins appending points to the 1st point.

For all polyline strings, a cross is then drawn in each plan view that the string's model is on, and the cross follows the cursor around the screen. If the string's model is not added to any plan view, the model is automatically added to all plan views.

After the cross is on the screen (moving with the cursor), clicking LB and accepting with MB selects the 1st vertex of the string (using the appropriate snaps).

If **Append + radius** was selected, the user will then be prompted for the radius of the segment being placed.

The string is then drawn from the 1st vertex to the cursor position, which represents the second vertex of the string. Clicking LB and accepting with MB selects the second string vertex and the process repeats for subsequent string vertices.

Now that the string is created, all the edit option on the string's **Polyline Edit** menu are usable.

The options in the **Polyline Edit** menu are not only used for placing the initial vertices of the string, but for editing the string once it is created. Since the **Append** and other options in the **Polyline Edit** menu are identical to the options used when editing an existing string, they will be discussed in detail in the string **Editor** section.

For full information in the Polyline Editor, go to the section 14.20.7 Polyline Edit - Old

14.20.2 Alignment Edit

Position of option on menu: **Strings =>Editor**

An alignment string is defined by specifying the horizontal and vertical geometry as separate operations.

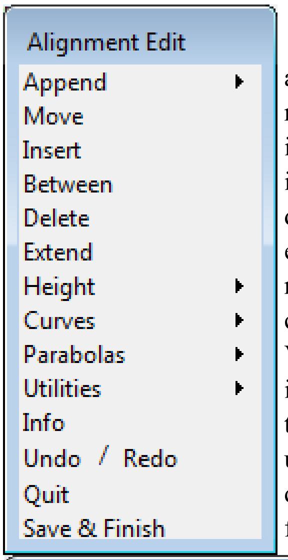
The horizontal geometry consists of a series of (x,y) points (called horizontal intersection points, HIP's) and circular curves and transitions (eg clothoid spiral) applied to the intersection points.

Vertical geometry also consists of a series of points but they are defined with respect to the plan length of the string (chainage) and height. Hence, the vertical geometry is defined by a series of (chainage,height) points called vertical intersection points (VIP's) and parabolic or circular curves applied to the vertical intersection points.

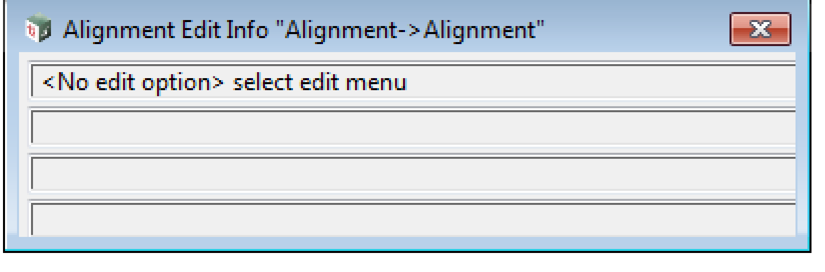
The horizontal geometry is defined in a plan view and the vertical geometry in a section view. Hence the alignment string editor can edit information for the string on both plan and section views.

Since a section view is only defined in relation to a specific string (called the primary string for the section view), the vertical geometry of the picked string can only be edited in a section view if the string is the primary string for that section view. That is, the string can only be edited on a section view if the string being edited is profiled on that section view.

On selecting an alignment string, the **alignment edit** menu and the **alignment edit info** are placed on the screen.



append or prepend an IP
move an IP or TP
insert an IP
insert on IP-IP line
delete an IP
extend an IP
modify the strings z-value
circular curves, transitions (*e.g.* spirals)
VG only, radius or length
interval, start ch, validate, *etc*.
toggle edit info panel
undo/redo
quit the edit session
finish the edit session



Most of the options are applicable to either horizontal or vertical geometry. The meaning of the option depends on whether the string is being edited in a plan or a section view. Hence the description of the option will often need to be split into its separate action on horizontal or vertical geometry.

See the earlier section 14.4.1 Super String Edit - Common Information for general information about editing strings.

Each option will now be discussed in detail.

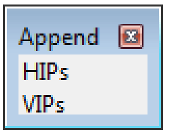
Go to 14.20.2.1 Alignment Append

Go to 14.20.2.2 Alignment Move
Go to 14.20.2.3 Alignment Insert
Go to 14.20.2.4 Alignment Between
Go to 14.20.2.5 Alignment Delete
Go to 14.20.2.6 Alignment Extend
Go to 14.20.2.7 Alignment Height
Go to 14.20.2.8 Alignment Curves
Go to 14.20.2.9 Alignment Parabolas
Go to 14.20.2.10 Alignment Utilities

14.20.2.1 Alignment Append

The **append** operation for an alignment string is similar to the 3d string case except that there are separate append options for the horizontal and vertical geometry of the string.

The **append** walk-right is



14.20.2.1.1 Append HIPs - Horizontal Geometry

The **append=>HIPs** option in a plan view is used to create the 1st horizontal point in a new alignment string, to append a new horizontal intersect point to the end of the string or to prepend a new horizontal intersection point to the beginning of the string. In this option, both appending and prepending will be referred to as appending.

14.20.2.1.1.1 Existing Alignment String

Appending a point is a two step process.

Step (a) - selecting the end to append the point to.

After picking **append=>HIPs**, the end of the string to append the point to is selected. Once the string end is selected, the new intersection point is assumed to be at the current cursor position. As the cursor is moved, the string is redrawn reflecting the changing position of the appended intersection point

message area 1 <Append Points>
message area 2 IP point number, x, y, z co-ords of IP
message area 3 bearing-in, bearing out, bearing difference for an IP
message area 4 select final position - after pick
Screen message area
<Select string end to append to> [picks][][menu]
<Select string end to append to> [picks][accepts][menu]

Step (b) - selecting the position for the new appended point.

A cross indicates where the cursor currently is. The position of the new appended point is set to the current cursor position by picking (LB) and accepting (MB).

message area 1 <Append Points>
message area 2 IP point number, x, y, z co-ords of IP
message area 3 bearing-in, bearing out, bearing difference for closest IP
message area 4 select final position - before pick
Screen message area
<Select final position of point> [picks][][menu]
<Select final position of point> [picks][accepts][menu]

Once an intersection point has been appended to the string, the appended intersection point is considered to be the selected string end and a new append cycle begins. That is, stage (a) is already set up. The current cursor position indicates the new position of the next appended intersection point.

Hence a series of string points is easily entered by first selecting the string end that the new points are to be appended to (step (a)) and then moving the cursor to the position of each new point in turn and selecting them in turn.

Typed input can be used in either step.

The **append** option is terminated by either bringing up the **pick ops** menu and selecting **cancel** or by selecting a new option from the **alignment edit** menu.

If, after bringing up the **pick ops** menu, it is decided to continue with the append option, simply select the **restart** option from the **pick ops** menu and the **pick ops** menu will disappear leaving the append option still current.

14.20.2.1.1.2 New Alignment String

When the **append**=>**HIPs** option is selected, a cross will appear on any plan views that have the alignment string's model on them.

The first horizontal intersection point is then selected by the standard pick (LB) and accept (MB).

The option then continues as if appending to an existing alignment string where the end point has already been accepted.

14.20.2.1.2 Append VIPs - Vertical Geometry

The **append**=>**VIPs** option is used to create and edit the vertical geometry of the picked string if the string is a primary string on any section view. This can be achieved by using either the **VG edit** or the **profile** option from the section view **view**=>**utilities** menu.

Vertical geometry can only be defined for a string with existing horizontal geometry.

On a section view, **Append**=>**VIPs** is used to create the first vertical intersection point in an existing string, to append a new vertical intersect point to the end of the string or to prepend a new vertical intersection point to the beginning of the string. In this option, both appending and prepending will be referred to as appending.

14.20.2.1.2.1 Existing Vertical Geometry

Appending a vertical point is the same as for a horizontal intersection point except that the co-ordinate system is (chainage,height) rather than (x,y) values and the information messages displayed in the **alignment edit info** panel show chainage, height and percentage grade rather than x,y,z and bearings.

For example -

message area 1	<Append Points>	
message area 2	VIP point number, chainage,height of VIP	
message area 3	curve length, %grade-in, %grade-out, grade difference for an VIP	
message area 4	edit finished - before pick	select final position - after pick

14.20.2.1.2.2 New Vertical Geometry

Creating the **1st vertical intersection point** (VIP) in a string is similar to the 1st horizontal IP.

When the **append**=>**VIPs** option is selected, a cross will appear on any section views that have the alignment string profiled on them.

The 1st vertical intersection point is then selected by the standard pick (LB) and accept (MB).

The option then continues as if appending to an existing vertical geometry where the end point has already been accepted.

Typed input can be used at any stage of the append.

See the next section 14.20.2.2 Alignment Move or return to 14.20.2 Alignment Edit.

14.20.2.2 Alignment Move

The **move** option allows the user to move

(a) individual horizontal intersection points (HIPs)

(b) individual tangent points (HTPs) if their is no transition on the HIP.

(c) individual vertical intersection points or tangent points (VIPs or VTPs).

move acts on the horizontal geometry if the point to move is selected in a plan view, or the vertical geometry if the point to move is selected in a section view.

14.20.2.2.1 Move - Horizontal Geometry

The **move** option in a plan view is for moving individual horizontal intersection points (HIP's) or horizontal tangent points (HTPs) of the string.

The move cycle consists of two steps:

(a) selecting the point to be moved

(b) selecting the new position for the point.

Step (a)

First the horizontal intersection or tangent point to be moved is selected. The selected point will then move around the view as the cursor is moved.

If a horizontal intersection point is selected, the string will be redrawn with the intersection point moved to the cursor position and the circular curve radius and transition lengths left constant.

A horizontal tangent point only occurs on a line connecting two adjacent horizontal intersection points (the HIP-HIP line). Hence, a horizontal tangent point can only move along its HIP-HIP line. Moving a horizontal tangent point actually means that the radius of the circular curve at the tangent point is modified so that the new position of the point is still a tangent point on the HIP-HIP line for the new circular curve.

To guarantee that the new position for the tangent point is on the HIP-HIP line, the cursor position is automatically projected perpendicularly onto the HIP-HIP line to give the new tangent point.

message area 1	<Move Point>
message area 2	IP point number, x, y co-ords of IP, radius, start spiral, end spiral
message area 3	bearing-in, bearing out, bearing difference for the inserted IP
message area 4	select final position - after pick
Screen message area	

<Select point to move> [picks][][menu]
<Select point to move> [picks][accepts][menu]

Step (b)

The current cursor position (or its projection for a HTP) is selected as the new position for the point by picking (LB) and accepting (MB). The point being moved is then anchored at the cursor position for the HIP (or at the projected point for a HTP), and the string redrawn.

message area 1	<Move Point>
message area 2	IP point number, x, y co-ords of IP, radius, start spiral, end spiral
message area 3	bearing-in, bearing out, bearing difference for the inserted IP
message area 4	select final position - before pick, edit finished - after pick
Screen message area	

<Select final position of point> [picks][][menu]
<Select final position of point> [picks][accepts][menu]

Typed input can be used in either step

Once the move cycle is completed and the point moved, the move option is still current and can be repeated for other points without having to re-select the move option.

move is terminated by selecting **cancel** from the **pick ops** menu or by selecting a new **alignment edit** option.

14.20.2.2.2 Move - Vertical Geometry

The **move** option is used in a section view to move individual vertical intersection points (VIP's) or vertical tangent points (VTP's).

The move cycle consists of two steps:

(a) selecting the point to be moved

(b) selecting the new position for the point.

Step (a)

First the vertical intersection or tangent point to be moved is selected. The selected point will then move around the view as the cursor is moved.

If a vertical intersection point (VIP) is selected, the string will be redrawn with the vertical intersection point moved to the cursor position and the parabolic curve length or the circular curve radius left constant.

A vertical tangent point (VTP) only occurs on a line connecting two adjacent vertical intersection points (the VIP-VIP line). Hence, a vertical tangent point can only move along its VIP-VIP line. Moving a vertical tangent point actually means that the length of the parabolic curve at the vertical tangent point is modified so that the new position of the point is still a vertical tangent point on the VIP-VIP line for the new parabolic curve.

To guarantee that the new position for the vertical tangent point is on the VIP-VIP line, the cursor position is automatically projected perpendicularly onto the VIP-VIP line to give the new vertical tangent point.

message area 1 <Move Point>
message area 2 VIP point number, ch, ht co-ords of VIP
message area 3 VC length, %grade-in, %grade-out, grade diff for the moved VIP
message area 4 select final position - after pick
Screen message area
<Select point to move> [picks][][menu]
<Select point to move> [picks][accepts][menu]

Step (b)

The current cursor position (or its projection for a VTP) is selected as the new position for the point by selecting (LB) and accepting (MB). The point being moved is then anchored at the cursor position for an VIP (or at the projected point for a VTP), and the string redrawn.

message area 1 <Move Point>
message area 2 VIP point number, ch, ht co-ords of VIP
message area 3 VC length, %grade-in, %grade-out, grade diff for the moved VIP
message area 4 select final position - before pick, edit finished - after pick
Screen message area
<Select final position of point> [picks][][menu]
<Select final position of point> [picks][accepts][menu]

Typed input can be used in either step.

Once the move cycle is completed and the point moved, the **move** option is still current and can be repeated for other points without having to re-select the move option.

The **move** option is terminated by selecting **cancel** from the **pick ops** menu or by selecting a new **alignment edit** option.

See the next section 14.20.2.3 Alignment Insert or return to 14.20.2 Alignment Edit.

14.20.2.3 Alignment Insert

The **insert** option is designed to place a new intersection point in a string between two adjacent intersection points. The inserted point does not have to be on the line joining the two intersection points - the between option is used to guarantee that the IP is on the IP-IP line.

Inserting a point, like moving a point, is a two step process.

Step (a) - selecting the IP's to be on either side of the new intersection point

The two adjacent intersection points are chosen by selecting the line connecting the two intersection points. Once the line is selected, the new IP is assumed to be at the current cursor position. As the cursor is moved, the string is redrawn reflecting the changing position of the inserted IP.

Like the **move** option, the **insert** works on the horizontal or the vertical geometry depending on whether a plan or section view is used to select the connecting line.

For horizontal inserts

message area 1 <Insert Point>
message area 2 IP point number, x, y co-ords of IP, radius, start spiral, end spiral
message area 3 bearing-in, bearing out, bearing difference for the inserted IP
message area 4 select final position - after pick
Screen message area
<Select line to insert on> [picks][][menu]
<Select line to insert on> [picks][accepts][menu]

For vertical inserts

message area 1 <Insert Point>
message area 2 VIP point number, Ch, Ht co-ords of VIP
message area 3 VC length, %grade-in, %grade-out, grade diff for the moved VIP
message area 4 select final position - after pick
Screen message area
<Select line to insert on> [picks][][menu]

<Select line to insert on> [picks][accepts][menu]

Step (b) - selecting the position for the new intersection point

The position of the new intersection point is set to the current cursor position by picking (LB) and accepting (MB).

horizontal

message area 1	<Insert Point>
message area 2	IP point number, x, y co-ords of IP, radius, start spiral, end spiral
message area 3	bearing-in, bearing out, bearing difference for the inserted IP
message area 4	select final position - before pick, edit finished - after pick

Screen message area

<Select final position of point> [picks][][menu]
<Select final position of point> [picks][accepts][menu]

vertical

message area 1	<Insert Point>
message area 2	VIP point number, Ch, Ht co-ords of VIP
message area 3	VC length, %grade-in, %grade-out, grade diff for the moved VIP
message area 4	select final position - before pick, edit finished - after pick

Screen message area

<Select final position of point> [picks][][menu]
<Select final position of point> [picks][accepts][menu]

Once the insert cycle is completed and the point inserted, the insert option is still current and can be repeated for other insertions without having to re-select the insert option.

The **insert** option is terminated by selecting **cancel** from the **pick ops** menu or by selecting a new option from the **alignment edit** menu.

Typed input can be used in either step.

See the next section 14.20.2.4 Alignment Between or return to 14.20.2 Alignment Edit.

14.20.2.4 Alignment Between

The **between** option is similar to the insert option except the inserted point does have to be on the line joining the two intersection points. To accomplish this, the cursor position is automatically projected onto the IP-IP line to give the new IP point position.

The **between** option is terminated by selecting **cancel** from the **pick ops** menu or by selecting a new option from the **alignment edit** menu.

See the next section 14.20.2.5 Alignment Delete or return to 14.20.2 Alignment Edit.

14.20.2.5 Alignment Delete

The **delete** option is used to delete intersection points from the string.

After picking the delete option, any selected intersection point in the string is deleted. The string, minus the deleted point, is redrawn after each deletion.

Once an intersection point has been deleted, another intersection point in the selected string can selected for deletion. Hence any number of the string points can be deleted one after another.

A horizontal intersection point is deleted if the point is chosen from a plan view and a vertical intersection point deleted if the point is selected from a vertical view.

For horizontal deletes

message area 1	<Delete Points>
message area 2	IP point number, x, y co-ords of IP, radius, start spiral, end spiral
message area 3	bearing-in, bearing out, bearing difference for the inserted IP

Screen message area

<Select point to delete> [picks][][menu]
<Select point to delete> [picks][accepts][menu]

For vertical deletes

message area 2 VIP point number, Ch, Ht co-ords of VIP
message area 3 VC length, %grade-in, %grade-out, grade diff for the moved VIP
Screen message area
<Select point to delete> [picks][][menu]
<Select point to delete> [picks][accepts][menu]

The delete option is terminated by selecting **cancel** from the **pick ops** menu or by selecting a new option from the **alignment edit** menu.

Typed input can be used to select a point for deletion.

See the next section 14.20.2.6 Alignment Extend or return to 14.20.2 Alignment Edit.

14.20.2.6 Alignment Extend

The **extend** option is used to move an intersection point along the line joining the intersection point to its neighbouring intersection point.

That is, in a plan view, the bearing of the HIP-HIP line is kept constant and the intersection point is moved along that line either towards or away from its neighbouring intersection point on the HIP-HIP line.

In a section view, the grade of the VIP-VIP line is kept constant and the vertical intersection point is moved along that line either towards or away from its neighbouring intersection point on the VIP-VIP line.

Extending, like moving a point, is a two step process.

Step (a) - selecting the IP-IP line and the IP to be moved along that line

The IP-IP line and the intersection point to be moved are chosen in the one operation by selecting a co-ordinate point near the IP-IP line and close to the intersection point to be moved along that line.

Once the line and IP are selected, the new position of the selected IP is assumed to be at the current cursor position projected perpendicularly onto the IP-IP line. As the cursor is moved, the string is redrawn reflecting the changing position of the moved IP.

For horizontal extends

message area 1 <Extend Point>
message area 2 IP point number, x, y co-ords of IP, radius, start spiral, end spiral
message area 3 bearing-in, bearing out, bearing difference for the inserted IP
message area 4 select final position - after pick
Screen message area
<Select line to extend> [picks][][menu]
<Select line to extend> [picks][accepts][menu]

For vertical extends

message area 1 <Extend Point>
message area 2 VIP point number, Ch, Ht co-ords of VIP
message area 3 VC length, %grade-in, %grade-out, grade diff for the moved VIP
message area 4 select final position - after pick
Screen message area
<Select line to extend> [picks][][menu]
<Select line to extend> [picks][accepts][menu]

Step (b) - selecting the final position for the intersection point

The final position for the intersection point is set to the projection of the current cursor position onto the IP-IP line.

horizontal -

message area 1 <Extend Points>
message area 2 IP point number, x, y co-ords of IP, radius, start spiral, end spiral
message area 3 bearing-in, bearing out, bearing difference for the inserted IP
message area 4 select final position - before pick, edit finished- after pick

Screen message area

<Select final position of point> [picks][][menu]

<Select final position of point> [picks][accepts][menu]

vertical

message area 1	<Extend Points>
message area 2	VIP point number, Ch, Ht co-ords of VIP
message area 3	VC length, %grade-in, %grade-out, grade diff for the moved VIP
message area 4	select final position - before pick, edit finished- after pick

Screen message area

<Select final position of point> [picks][][menu]

<Select final position of point> [picks][accepts][menu]

Once the extend is completed, the extend option is still current and can be repeated without re-selecting the extend option.

The **extend** option is terminated by selecting **cancel** from the **pick ops** menu or by selecting a new option from the **alignment edit** menu.

Typed input can be used in either step.

Note - **Extend** can be used on the end points of the string.

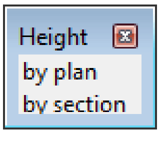
See the next section 14.20.2.7 Alignment Height or return to 14.20.2 Alignment Edit.

14.20.2.7 Alignment Height

The **height** option is used to insert and/or modify the height of a vertical intersection point in a plan view, or to modify the height of a vertical intersection point in a section view.

Hence the height option is unique - it only affects the vertical geometry of the string but can be used in a plan or section view.

The **height** walk-right is



14.20.2.7.1 By Plan - Height in a Plan View

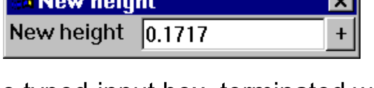
In a plan view, the **height=>by plan** option is used to create a **vertical intersection point** of a given height, or modify the height of an existing vertical point, at a selected plan position on the string.

After the **height** option is chosen, the user selects a position on the string in a plan view. The use of point and line snap will important for selecting the correct position.

If a vertical intersection point already exists at the chainage of the selected position, it will be selected for modification. Otherwise a new vertical intersection point will be inserted into the vertical geometry at the chainage of the selected position.

A **new height** typed-input box is then displayed on the screen with the vertical intersection point's current height (z value).

The **new height** typed-input box looks like:



The height is entered into the typed-input box, terminated with <Enter>. The entered value is taken as the new height of the vertical intersection point in the alignment.

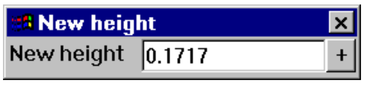
The typed-input box then disappears.

14.20.2.7.2 By Section - Height in a Section View

In a section view, the **height=>by section** option can only be used to modify the height of an **existing** vertical intersection point.

After the **height** option is chosen, the user selects the vertical intersection point that will have its height modified from a section view. After selecting the VIP, a **new height** typed-input box is displayed on the screen with the point's current height (z value).

The **new height** typed-input box looks like:



The height is entered into the typed-input box, terminated with <enter>. The entered value is taken as the new height of the vertical intersection point in the alignment string and the string redrawn with the new height at that point.

The typed-input box then disappears.

For either **height=>by plan** or **height=>by section**, the **height** option is terminated on selecting **cancel** from the **pick ops** menu or by selecting a new option from the **alignment edit** menu.

See the next section 14.20.2.8 Alignment Curves or return to 14.20.2 Alignment Edit.

14.20.2.8 Alignment Curves

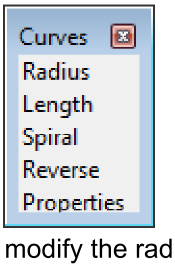
The **Curves** option allows the user to

(a) add a circular curve of a specified radius or curve length to a horizontal intersection point

(b) add start and end transition (eg spiral) to an intersection point with and existing circular curve

(c) add a circular curve of a specified radius or curve length to a vertical intersection point

The **Curves** walk-right menu is



The **radius** option is used to create or modify the radius of a circular curve at a horizontal or vertical intersection point. **radius** acts on the horizontal geometry if the point is selected in a plan view, or the vertical geometry if the point is selected in a section view.

Similarly the **length** option is used to set a circular curve by giving its total curve length rather than the radius. **length** acts on the horizontal geometry if the point is selected in a plan view, or the vertical geometry if the point is selected in a section view.

The **spiral** option is for adding transitions to a horizontal intersection point with a curve already on it.

14.20.2.8.1 Radius

In a plan view, the **radius** option is used to add a circular curve to an intersection point with no curve or to modify the radius of the circular curve if one already exists.

In a section view, the **radius** option is used to add a circular curve to an intersection point with no curve or to modify the radius of a circular curve if one already exists.

Radius is a two step process.

Step (a) - selecting the intersection point

First the intersection point whose curve is to be modified is selected.

For horizontal radius

message area 1	<Radius>
message area 2	IP point number, x, y co-ords of IP, radius, start spiral, end spiral
message area 3	bearing-in, bearing out, bearing difference for the inserted IP

For vertical radius

message area 1	<Radius>
message area 2	VIP point number, Ch, Ht co-ords of VIP
message area 3	radius, %grade-in, %grade-out, grade diff for the moved VIP
message area 4	enter value- after pick

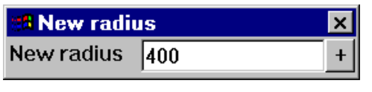
Screen message area

<Select point to change radius> [picks][][menu]
<Select point to change radius> [picks][accepts][menu]

Step (b) - entering the new radius

after an intersection point is chosen, a new radius typed-input box is displayed on the screen with either the intersection point's current curve radius or, if no curve exists, zero.

The new radius typed-input box looks like:



The radius is entered into the typed-input box, terminated with <enter>. The entered value is taken as the radius of the curve at that intersection point and the string redrawn with the new curve. The typed-input box then disappears.

If an existing curve is to be removed from an intersection point, enter the value zero into the new radius typed-input box.

horizontal

message area 1	<Radius>
message area 2	IP point number, x, y co-ords of IP, radius, start spiral, end spiral
message area 3	bearing-in, bearing out, bearing diff for the inserted IP
message area 4	enter value - before value entered edit finished- after value entered

vertical

message area 1	<Radius>
message area 2	VIP point number, Ch, Ht co-ords of VIP
message area 3	radius, %grade-in, %grade-out, grade diff for the moved VIP
message area 4	enter value - before value entered edit finished- after value entered

Like most of the other **alignment edit** options, after a radius has been modified the option is still current and another intersection point can be chosen to have its curve radius modified.

The **radius** option is terminated by selecting a different **alignment edit** option.

14.20.2.8.2 Length

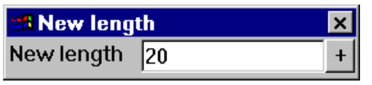
Although the circular curve on an intersection point is stored as a radius, it is possible to define the radius indirectly by giving the total curve length by using the **length** option.

The option works in plan or section views.

To use **length**, first select the intersection point to add a new circular curve to, or to modify the existing curve.

A new length typed-input box is then displayed on the screen with either the current curve length, or if no curve exists, zero.

The new length typed-input box looks like:



The total curve length is entered into the typed-input box, terminated with <enter>. The entered value is taken as the new curve length. The typed-input box then disappears.

If an existing circular curve is to be removed, zero is entered into the new length typed-input box.

Like most of the other **alignment edit** options, after the **length** has been modified the option is still current and another IP can be chosen to have its curve modified.

The **length** option is terminated by selecting a different **alignment edit** option.

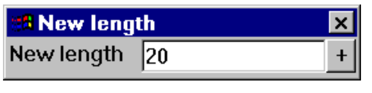
WARNING - when **length** is used, the equivalent radius is computed and stored with the curve. If the IP is moved, the **radius** is kept **constant** and the total curve **length** is **modified**.

14.20.2.8.3 Spiral

In a plan view, the **spiral** option is used to add a start or end transitions (eg spirals) to a curve on an intersection point, or to modify an existing transition. Spiral does not work on a section view.

After picking **spiral**, select the end of the curve to add a new transition to, or the existing transition to be modified. A new length **typed-input box** is displayed on the screen with either the current transition length, or zero if no transition exists.

The new length typed-input box looks like:



The transition length is entered into the typed-input box, terminated with <enter>. The entered value is taken as the new transition length. The typed-input box then disappears.

If an existing transition is to be removed, zero is entered into the new length typed-input box.

Like most of the other **alignment edit** options, after the **spiral** has been modified the option is still current and another curve can be chosen to have its transitions modified.

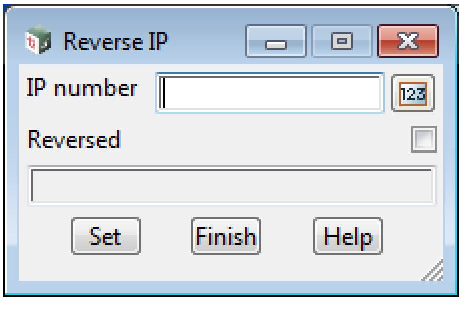
The **spiral** option is terminated by selecting a different **alignment edit** option.

14.20.2.8.4 Reverse

12d Model normally creates a curve on the side of the IP that has the smallest angle. The sign of the radius (positive or negative) is automatically determined by *12d Model*.

The **Reverse** option creates a circular curve that is the part left over from the standard curve.

On selecting **Reverse**, the **Reverse IP** panel is displayed.



The HIP is chosen by either typing in the HIP number in the IP number field, or by clicking on the **123** button and then selecting the HIP.

To change the curve from reversed/not reversed, simply tick on/off the Reversed tick box and then click on **Set**.

See the next section 14.20.2.9 Alignment Parabolas or return to 14.20.2 Alignment Edit.

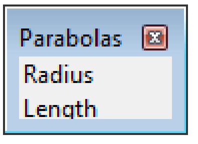
14.20.2.9 Alignment Parabolas

The **parabolas** option is for adding, removing or modifying parabolic curves to the vertical geometry. It allows the user to

(a) add a parabolic curve of a given curve length to a vertical intersection point

(b) add a parabolic curve of a given effective radius to a vertical intersection point

The **parabolas** walk-right menu is



The **radius** option is used to set a parabolic curve of given effective radius to a vertical intersection point.

The **length** option is used to set a parabolic curve of a given curve length to a vertical intersection point.

Note - the option only works on the vertical geometry if the VIP is selected in a section view. Points cannot be selected in any other view.

14.20.2.9.1 Radius

In a section view, the **radius** option is used to add a parabolic curve to an intersection point with no curve or to modify the radius of the parabolic curve if one already exists.

Radius is a two step process.

Step (a) - selecting the intersection point

First the intersection point whose parabola is to be modified is selected.

message area 1	<Radius>
message area 2	VIP point number, Ch, Ht co-ords of VIP
message area 3	VC length, %grade-in, %grade-out, grade diff for the moved VIP
message area 4	enter value- after pick

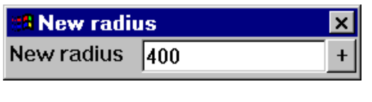
Screen message area

<Select point to change radius> [picks][][menu]
<Select point to change radius> [picks][accepts][menu]

Step (b) - entering the new radius

After an intersection point is chosen, a new radius typed-input box is displayed on the screen with either the intersection point's current parabolic radius or, if no parabola exists, the last value entered into the radius typed-input box.

The new radius typed-input box looks like:



The radius is entered into the typed-input box, terminated with <enter>. The entered value is taken as the radius of the parabola at that intersection point and the string redrawn with the new parabola. The typed-input box then disappears.

If an existing parabola is to be removed from an intersection point, enter the value zero into the radius typed-input box.

message area 1 <Radius>
message area 2 VIP point number, Ch, Ht co-ords of VIP
message area 3 VC length, %grade-in, %grade-out, grade diff for the moved VIP
message area 4 enter value - before value entered edit finished- after value entered

Like most of the other **alignment edit** options, after the radius has been modified the option is still current and another intersection point can be chosen to have its parabolic radius modified.

The **radius** option is terminated by selecting a different **alignment edit** option.

WARNING - when **radius** is used, the equivalent parabolic length is computed and stored with the curve. If the VIP is moved, the **parabolic length** is kept **constant** and the **effective radius** is **modified**.

14.20.2.9.2 Length

The **length** option is used to add a parabolic curve to a vertical intersection point with no curve or to modify the length of the parabolic curve if one already exists.

Adding or modifying curve length is a two step process.

Step (a) - selecting the vertical intersection point

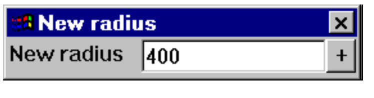
First the vertical intersection point whose parabola is to be modified is selected.

message area 1 <Length>
message area 2 VIP point number, Ch, Ht co-ords of VIP
message area 3 VC length, %grade-in, %grade-out, grade diff for the moved VIP
message area 4 edit finished - before pick enter value - after pick
Screen message area
<Select point to change length> [picks][][menu]
<Select point to change length> [picks][accepts][menu]

Step (b) - entering the new curve length

After a VIP is selected, a new length typed-input box is displayed on the screen with the point's current parabolic curve length.

The new length typed-input box looks like



The curve length is entered into the typed-input box, terminated with <enter>. The entered value is taken as the length of the parabolic curve at that vertical intersection point and the string redrawn with the new curve. The typed-input box then disappears.

If an existing parabola is to be removed from a vertical intersection point, enter the value zero into the curve length typed-input box.

message area 1 <Length>
message area 2 VIP point number, Ch, Ht co-ords of VIP
message area 3 VC length, %grade-in, %grade-out, grade diff for the moved VIP
message area 4 enter value- before value entered edit finished- after value entered

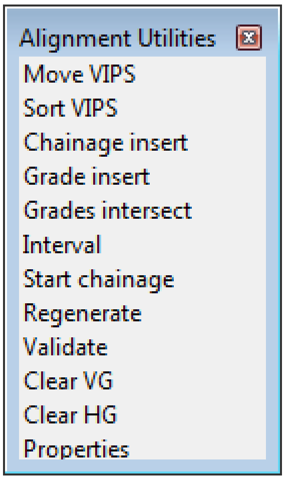
Like most of the other **alignment edit** options, after the parabolic curve length has been modified the option is still current and another vertical intersection point can be chosen to have its parabola modified.

The **length** option is terminated by selecting a different **alignment edit** option.

See the next section 14.20.2.10 Alignment Utilities or return to 14.20.2 Alignment Edit.

14.20.2.10 Alignment Utilities

The **Utilities** walk-right menu contains a number of useful miscellaneous option for the alignment string. The menu is

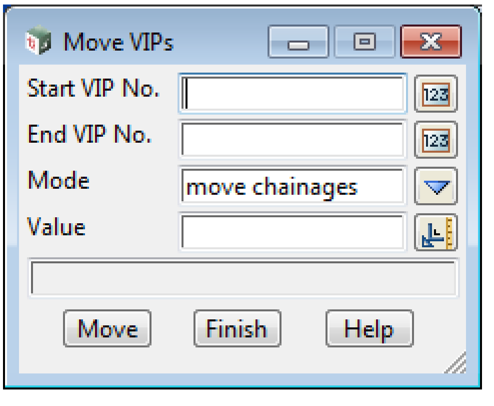


Each of the options will now be discussed.

14.20.2.10.1 Move VIPs

The **Move VIPs** option is for moving a range of vertical intersection points (VIPs) by a chainage or an elevation increment.

After selecting the option, the **Move VIPs** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Start VIP No.	input		

the VIP number of the 1st VIP in the range to be moved. If blank, then 1.

End VIP No.	input		

the VIP number of the last VIP in the range to be moved.If blank, then the last VIP.

Mode	input	move chainage	move chainage move hts

if ***move chainage****, the chainage of all the points in the specified range are incremented by the given value. If* ***move hts****, the heights (elevations) of all the points in the specified range are incremented by the given value.*

Value	input	0	

value to increment either the chainage or the height for all the VIPs in the specified range.

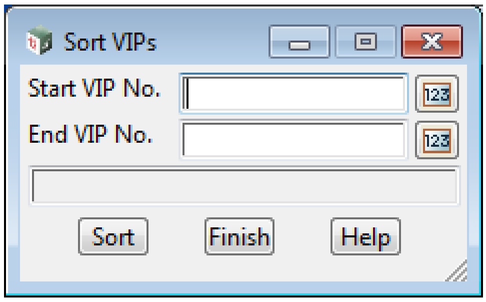
Move button

move the VIPs in the range given by the start and end ip field by the chainage or height given in the value field.

14.20.2.10.2 Sort VIPs

The **sort VIPs** option is for sorting VIP's into increasing chainage order.

After selecting the option, the **sort VIPs** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Start VIP No. input

if not blank, the VIP number of the 1st VIP in the range to be sorted. If blank, then 1.

End VIP No. input

the VIP number of the last VIP in the range to be sorted. If blank, then the last VIP.

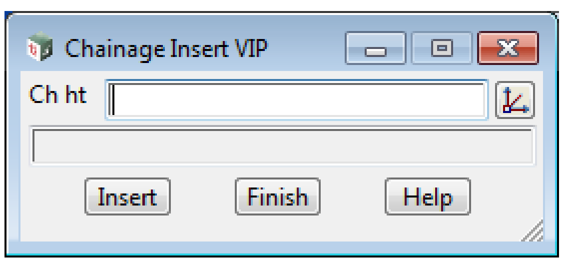
Sort button

sort the VIPs in the range given by the start and end ip field into chainage order.

14.20.2.10.3 Chainage Insert

The **chainage insert** option inserts a vertical intersection point (VIP) at a given chainage and height.

After selecting the option, the **chainage insert VIP** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Ch ht input

the chainage and height of the VIP to insert - separate values by spaces.

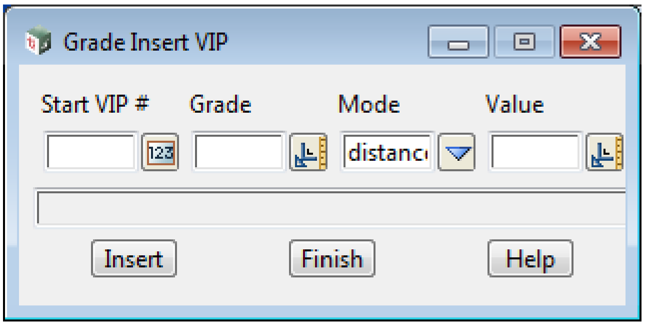
Insert button

insert the VIP point given in the Ch ht *field into the alignment string.*

14.20.2.10.4 Grade Insert

The **grade insert** option inserts a VIP at a given grade from an existing VIP. The new VIP position is given by either a distance from the picked VIP or at given chainage.

After selecting the option, the **grade insert VIP** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Start VIP #	input		

the VIP number whose position is used to start the grade insert from.

Grade	input		

percent grade to be used.

Note ***positive** is **up** when going in the direction of **increasing chainage** and **negative** is **up** when going in the direction of **decreasing chainage**.*

Mode	choice	distance	distance, chainage

*if **distance**, the new VIP is inserted at the given grade and distance (given in the* value *field) from the start VIP.*
*if **chainage**, the new VIP is inserted at the chainage (given in the* value *field but having the given grade from the start VIP.*

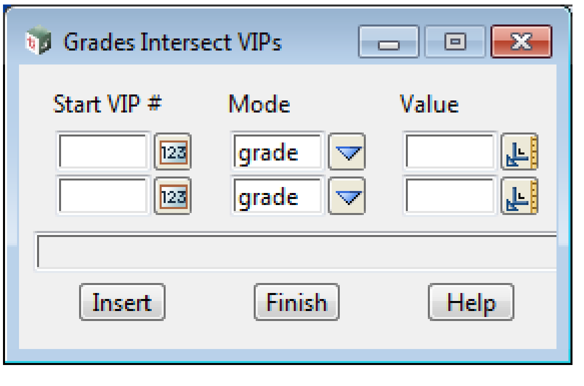
Insert	button		

insert the new VIP into the alignment string.

14.20.2.10.5 Grades Intersect

The **grades intersect** option inserts a VIP which is given by intersecting lines of given grades from two existing VIPs. The grades are either typed in or calculated by giving another VIP that the line goes through.

After selecting the option, the **Grades Intersect VIPs** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Start VIP #	input		

the VIP number which is used in defining a line.

Mode	choice	grade	grade, VIP #

if ***grade****, the field* value *is a percent grade and a line is defined as going through the start VIP and with the given grade.*
if ***VIP #****, the field* value *is the number of a VIP and a line is defined as going through the start VIP and this VIP.*
Note for grades ***positive*** *is* ***up*** *when going in the direction of* ***increasing chainage*** *and* ***negative*** *is* ***up*** *when going in the direction of* ***decreasing chainage****.*

Value	input		

if mode *is grade, value is a percent grade.*
if mode *is VIP #, value is the number of a VIP.*

Insert	button		

calculate the point which is the intersection of the two lines and insert it as a new VIP into the alignment string.

14.20.2.10.6 Interval

Each alignment string has a chainage interval that is used to define regular points along the string. The chainage interval is used for approximation the alignment in operations such as triangulating and applying templates to an alignment string.

After selecting the option, an **enter value** typed-input box is displayed on the screen with the string's current chainage interval placed in it.

The new chainage interval is entered into the typed-input box, terminated with <enter>. The typed-input box then disappears.

The **interval** option terminates after use.

14.20.2.10.7 Start Chainage

Each alignment string has a start chainage which can be positive, negative or zero.

After selecting the option, an **enter value** typed-input box is displayed on the screen with the string's current start chainage in it.

The new start chainage is entered into the typed-input box, terminated with <enter>. The typed-input box then disappears.

When a new start chainage is entered, the vertical intersection points are automatically moved so that they retain their same relative chainage with respect to the start point on the alignment.

The **start chainage** option terminates after use.

14.20.2.10.8 Regenerate

If the horizontal geometry of an alignment string is modified, then the profile through any triangulations would also change. Hence, if the alignment string being edited was profiled on any section views, then the profile would be need to be recalculated after any horizontal geometry changes.

On selecting the **regenerate** option, **any** section view that has the alignment string being edited as its primary string (that is, as the string defining the chainage for the section view) is automatically re-profiled.

Hence the **regenerate** option is equivalent to running the **regenerate** option for each section view with the edited alignment string as its primary string.

14.20.2.10.9 Validate

When constructing horizontal and vertical geometry, it is possible to end of with invalid constructs such as overlapping tangent points.

On selecting the **validate** option, the alignment string will be checked for

horizontal geometry checks

s at least two horizontal intersection points

s no co-incident horizontal intersection points

s no overlapping horizontal tangent points

vertical geometry checks

s at least two vertical intersection points

s no co-incident vertical intersection points

s no overlapping vertical tangent point

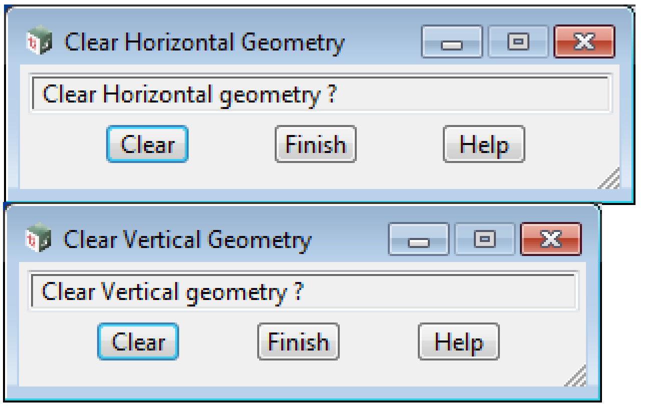
s vertical intersection points chainages are in ascending order

If any errors occur, an **errors** menu is placed on the screen.

The **errors** menu remains on the screen and only disappears when either the **[X]** is picked or one of the error messages is picked with LB.

Note - the **validate** option is automatically run when the edit is finished.

14.20.2.10.10 Clear VG and Clear HG



The **Clear VG** and **Clear HG** options are used to delete all the horizontal and/or vertical intersection points in the string.

The **Clear VG** option removes all the vertical geometry from the string.

The horizontal geometry still exists plus all the other string properties (model, name, colour and type).

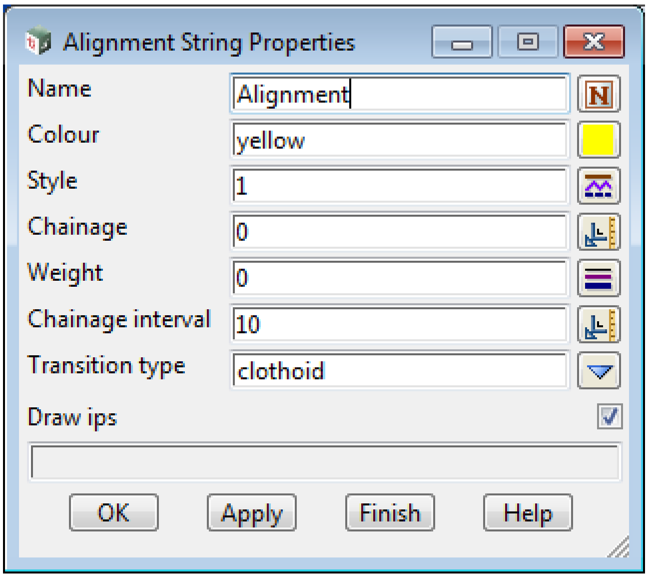
The **Clear HG** option removes all the horizontal and vertical geometry from the string.

The string still exists and keeps its other properties such as model, name, colour and type.

After a **Clear**, the horizontal or vertical geometry can be re- entered using the edit options.

14.20.2.10.11 Properties

Selecting Properties brings up the **Alignment String Properties** panel which is used to modify the string's header information.



The fields in this panel are similar to those in the **Create Alignment String** panel and the alignment string editor options. The only new field is

Field Description	Type	Defaults	Pop-Up
Draw ips	tick	tick	

if ***ticked****, the horizontal and vertical intersection points are draw on plan and section views. If* ***not ticked****, the ips are not draw on the plan and section views.*

OK/Apply	button		

for the string being edited, **OK** *sets the string with the values in the panel fields and removes the panel.* **Apply** *sets the string with the values in the panel fields and leaves the panel on the screen.*

See the next section 14.20.2.11 Finish or return to 14.20.2 Alignment Edit.

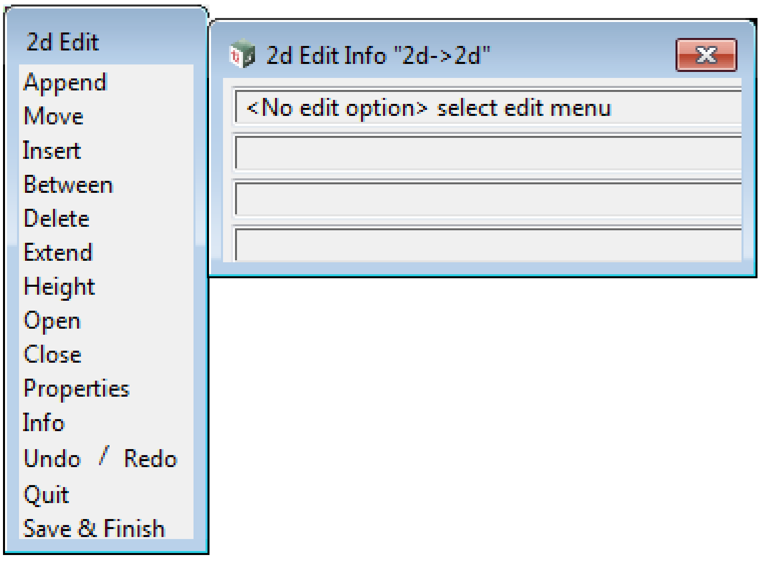
14.20.2.11 Finish

The **Finish** option is used to terminate the alignment string edits. After selecting the **Finish** option, the string is checked for overlapping horizontal or vertical tangent points and any errors reported.

14.20.3 2d Edit - Old

Position of option on menu: **Strings =>Editor**

On picking a 2d string, the **2d edit** menu and the **2d edit info** panel are placed on the screen.



See the earlier section 14.4.1 Super String Edit - Common Information for general information about editing strings.

Each option in the **2d edit** menu will now be described.

14.20.3.1 Append

The **append** option is used to create the 1st point in a new string, to append a new intersect point to the end of the string or to prepend a new intersection point to the beginning of the string. In this option, both appending and prepending will be referred to as appending.

14.20.3.1.1 Existing 2d Strings

Appending a point is a two step process.

Step (a) - selecting the end to append the point to.

After the **append** option has been selected, the end of the string to append the point to is selected. Once the string end is selected, the new intersection point is assumed to be at the current cursor position. As the cursor is moved, the string is redrawn reflecting the changing position of the appended intersection point

message area 1	<Append Points>
message area 2	IP point number, x, y, z co-ords of IP
message area 3	bearing-in, bearing out, bearing difference for an IP
message area 4	select final position- after pick

Screen message area

<Select string end to append to> [picks][][menu]
<Select string end to append to> [picks][accepts][menu]

Step (b) - selecting the position for the new appended point.

A cross indicates where the cursor currently is. The position of the new appended point is set to the current cursor position by picking (LB) and accepting (MB).

message area 1	<Append Points>
message area 2	IP point number, x, y, z co-ords of IP
message area 3	bearing-in, bearing out, bearing difference for closest IP
message area 4	select final position- before pick

Screen message area

<Select final position of point> [picks][][menu]
<Select final position of point> [picks][accepts][menu]

Once a point has been appended to the string, the appended point is considered to be the selected string end and a new append cycle begins. That is, stage (a) is already set up. The current cursor position indicates the new position for the next appended point.

Hence a series of string points is easily entered by first selecting the string end that the new points are to be appended to (step (a)) and then moving the cursor to the position of each new point and selecting them in turn.

Typed input can be used in either step.

The **append** option is terminated by either bringing up the **pick ops** menu and selecting **cancel** or by selecting a new option from the **2d edit** menu.

If, after bringing up the **pick ops** menu, it is decided to continue with the append option, simply select the **restart** option from the **pick ops** menu and the **pick ops** menu will disappear leaving the append option still current.

14.20.3.1.2 New 2d String

For creating a new string, the cursor is used to select the 1st point of the string. The option then continues as if appending to an existing 2d string where the end point has already been selected.

See the next section 14.20.3.2 Move or return to 14.20.3 2d Edit - Old.

14.20.3.2 Move

The **move** option is for moving individual points (intersection points - IP's) of the string.

The move cycle consists of two steps:

(a) selecting the point to be moved

(b) selecting the new position for the point.

Step (a)

First the point to be moved is selected. The selected point will then move around the view and the string redrawn to show the change as the cursor is moved.

message area 1	<Move Point>
message area 2	IP point number, x, y, z co-ords of IP
message area 3	bearing-in, bearing out, bearing difference for the closest IP
message area 4	select final position- after pick

Screen message area

<Select point to move> [picks][][menu]
<Select point to move> [picks][accepts][menu]

Step (b)

The current cursor position is selected as the new position for the point by selecting (LB) and accepting (MB). The point being moved is then anchored at the cursor position for the IP, and the string redrawn.

message area 1	<Move Point>
message area 2	IP point number, x, y, z co-ords of IP
message area 3	bearing-in, bearing out, bearing difference for the moved IP
message area 4	select final position- before pick, edit finished- after pick

Screen message area

<Select final position of point> [picks][][menu]
<Select final position of point> [picks][accepts][menu]

Typed input can be used in either step.

Once the move cycle is completed and the point moved, the move option is still current and can be repeated for other points without having to re-select the **move** option.

The **move** option is terminated by selecting **cancel** from the **pick ops** menu or by selecting a new **2d edit** option.

See the next section 14.20.3.3 Insert or return to 14.20.3 2d Edit - Old.

14.20.3.3 Insert

The **Insert** option is designed to place a new intersection point in a string between two adjacent intersection points (note that the inserted point does not have to be on the line joining the two intersection points).

Inserting a point, like moving a point, is a two step process.

Step (a) - selecting the IP's to be on either side of the new intersection point

The two adjacent intersection points are chosen by selecting the line connecting the two intersection points. Once the line is selected, the new IP is assumed to be at the current cursor position. As the cursor is moved, the string is redrawn reflecting the changing position of the inserted IP.

message area 1	<Insert Point>
message area 2	IP point number, x, y, z co-ords of IP
message area 3	bearing-in, bearing out, bearing difference for closest IP
message area 4	select final position- after pick

Screen message area

<Select line to insert on> [picks][][menu]
<Select line to insert on> [picks][accepts][menu]

Step (b) - selecting the position for the new intersection point

The position of the new intersection point is selecting

message area 1	<Insert Point>
message area 2	IP point number, x, y, z co-ords of IP
message area 3	bearing-in, bearing out, bearing difference for the inserted IP
message area 4	select final position - before pick, edit finished - after pick

Screen message area

<Select final position of point> [picks][][menu]
<Select final position of point> [picks][accepts][menu]

Once the insert cycle is completed and the point inserted, the insert option is still current and can be repeated for other insertions without having to re-select the **insert** option.

The **insert** option is terminated by selecting **cancel** from the **pick ops** menu or by selecting a new option from the **2d edit** menu.

Typed input can be used in either step.

See the next section 14.20.3.4 Between or return to 14.20.3 2d Edit - Old.

14.20.3.4 Between

The **between** option is similar to the **insert** option except that the inserted point does have to be on the line joining the two intersection points. To accomplish this, the cursor position is projected onto the IP-IP line to give the new IP point position.

The **between** option is terminated by selecting **cancel** from the **pick ops** menu or by selecting a new option from the **2d edit.**

See the next section 14.20.3.5 Delete or return to 14.20.3 2d Edit - Old.

14.20.3.5 Delete

The **delete** option is used to delete selected intersection points from the string.

The point to be deleted is picked (LB) and accepted (MB). When the point is accepted, it is deleted. The string, minus the deleted point, is then redrawn.

Once a point has been deleted, another point in the string can then be selected and deleted. Hence any number of points from the string can be deleted one after another.

message area 1 <Delete Points>
message area 2 IP point number, x, y, z co-ords of IP
message area 3 bearing-in, bearing out, bearing difference for closest IP
message area 5
Screen message area
<Select point to delete> [picks][][menu]
<Select point to delete> [picks][accepts][menu]

The **delete** option is terminated by selecting **cancel** from the **pick ops** menu or by selecting a new option from the **2d edit** menu.

Typed input can be used to select a point for deletion.

See the next section 14.20.3.6 Extend or return to 14.20.3 2d Edit - Old.

14.20.3.6 Extend

The **extend** option is used to move an intersection point along the line joining the intersection point to its neighbouring intersection point.

That is, the bearing of the IP-IP line is kept constant and the intersection point is moved along that line either towards or away from its neighbouring intersection point on the IP-IP line.

Extending, like moving a point, is a two step process.

Step (a) - selecting the IP-IP line and the IP to be moved along that line

The IP-IP line and the intersection point to be moved are chosen in the one operation by picking (LB) and accepting (MB) a co- ordinate point near the IP-IP line and close to the intersection point to be moved along that line.

Once the line and intersection point (IP) are selected, the new position of the selected IP is assumed to be at the current cursor position projected perpendicularly onto the IP-IP line.

As the cursor is moved, the string is redrawn reflecting the changing position of the moved IP.

message area 1 <Extend Point>
message area 2 IP point number, x, y, z co-ords of IP
message area 3 bearing-in, bearing out, bearing difference for closest IP
message area 4 select final position- after pick
Screen message area
<Select line to extend> [picks][][menu]
<Select line to extend> [picks][accepts][menu]

Step (b) - selecting the final position for the intersection point

The final position for the intersection point is set to the projection of the selected cursor position onto the IP-IP line.

message area 1 <Extend Points>
message area 2 IP point number, x, y, z co-ords of IP
message area 3 bearing-in, bearing out, bearing difference for closest IP
message area 4 select final position - before pick, edit finished - after pick
Screen message area
<Select final position of point> [picks][][menu]
<Select final position of point> [picks][accepts][menu]

Once the extend is completed, the extend option is still current and can be repeated without re-selecting the extend option.

The **extend** option is terminated by selecting **cancel** from the **pick ops** menu or by selecting a new option from the **2d edit** menu.

Typed input can be used in either step.

Note - Extend can be used on the end points of the string.

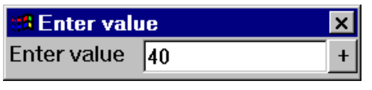
See the next section 14.20.3.7 Height or return to 14.20.3 2d Edit - Old.

14.20.3.7 Height

The **height** option is used to modify the height (z value) of the 2d string.

After the height option is chosen, an enter value typed-input box is displayed on the screen with the string's current height (z value).

The enter value typed-input box looks like:



The height is entered into the typed-input box, terminated by <enter>. The entered value is taken as the height of the 2dstring and the string redrawn with the new height. The typed-input box then disappears.

The height option automatically terminates and a new option is selected from the **2d edit**.

Note - all the points in a 2d string have the same height.

See the next section 14.20.3.8 Open or return to 14.20.3 2d Edit - Old.

14.20.3.8 Open

If the string is closed (that is, the end points have the same x and y values), selecting the **open** option removes the last point of the string.

If the string is not closed, the **open** option does nothing.

See the next section 14.20.3.9 Close or return to 14.20.3 2d Edit - Old.

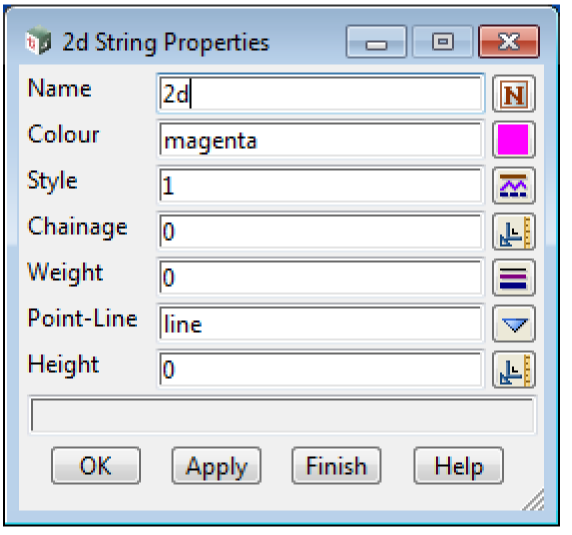
14.20.3.9 Close

Selecting the **close** option adds a point to the end of the string with the same co-ordinate values as the 1st point in the string.

See the next section 14.20.3.10 Properties or return to 14.20.3 2d Edit - Old.

14.20.3.10 Properties

Selecting Properties brings up the **2d String Properties** panel which is used to modify the string's header information.



The fields in this panel are similar to those in the **Create 2d String** panel and the 2d string editor options. The only new field is

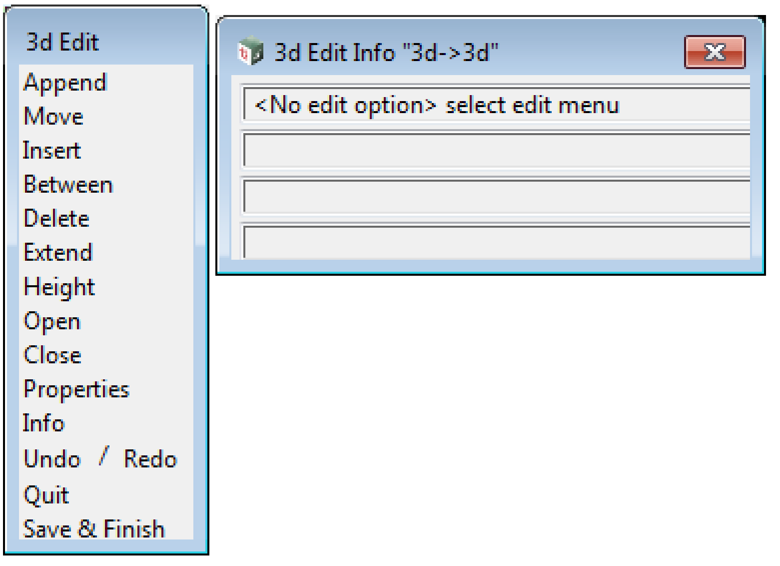
Field Description	Type	Defaults	Pop-Up
Set	button		

for the 2d string being edited, set all the items in the **2d String Properties** *panel to the values given in the panel.*

14.20.4 3d Edit - Old

Position of option on menu: **Strings =>Editor**

On picking a 3d string, the **3d edit** menu and 3d edit info panel are placed on the screen.



The difference between a 2d and a 3d string is that all points in a 2d string have the same z-value, whereas for a 3d string, each point can have a different z-value.

Hence most of the options in the **3d edit** menu are similar to the **2d edit** options of the same name, and only the differences for each option will be discussed. The *2d Edit* options are given in the section 14.4.2 Edit 2d

See the earlier section 14.4.1 Super String Edit - Common Information for general information about editing strings.

14.20.4.1 Append

The **Append** operation for a 3d string is similar to that for a 2d string except that a height (z-value) is required for each new point.

Since in most cases, it would be tiresome to ask for a height every time a point is added, the entry of a new height is controlled by the **height** toggle in the snaps menu. If **height** is toggled to on, then every time a point is placed or moved an enter height typed-input box is displayed on the screen.

The enter height typed-input box looks like:

The height is entered into the typed-input box, terminated with <enter>. The entered value is taken as the height of the 3d string point and the string redrawn with the new height at the point. The typed-input box then disappears.

When the enter height box is placed on the screen, it will already have a value in it depending on the circumstances preceding the operation.

For example, if a point or line was snapped to, the height at that point or line will be displayed in the box.

See the next section 14.20.3.7 Height or return to 14.20.4 3d Edit - Old.

14.20.4.2 Height

The **height** option is used to modify the height (z value) of any point in the string.

After the height option is chosen, the user must select which point is going to have its height modified.

After the point has been selected, an New height typed-input box is displayed on the screen with the point's current height (z value).

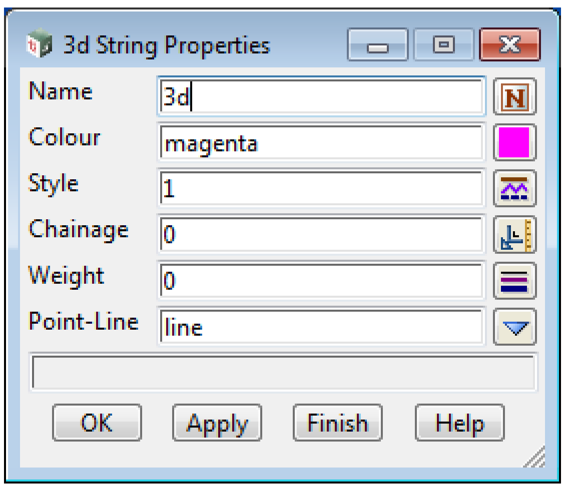
The height is entered into the typed-input box, terminated with <enter>. The entered value is taken as the height of the point in the 3d string and the string redrawn with the new height at that point. The typed-input box then disappears.

The **height** option is terminated on selecting **cancel** from the **pick ops** menu or by selecting a new option from the **3d edit** menu.

See the next section 14.20.3.10 Properties or return to 14.20.4 3d Edit - Old.

14.20.4.3 Properties

Selecting Properties brings up the **3d String Properties** panel which is used to modify the string's header information.



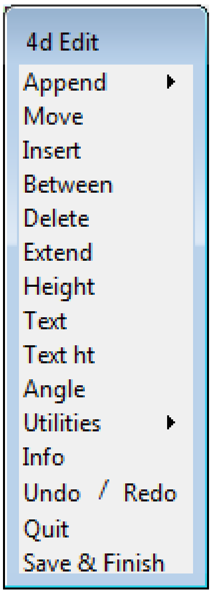
The fields in this panel are similar to those in the **create 3d string** panel and the 3dd string editor options. The only new field is

Field Description	Type	Defaults	Pop-Up
OK/Apply	button		

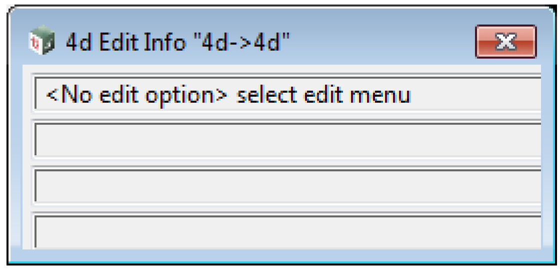
for the string being edited, **OK** *sets the string with the values in the panel fields and removes the panel.* **Apply** *sets the 3d string with the values in the panel fields and leaves the panel on the screen.*

14.20.5 4d Edit - Old

Position of option on menu: **Strings =>Editor**
On picking a 4d string, the **4d edit** menu and **4d edit info** panel are placed on the screen.



append/prepend an IP and text
move an IP
insert an IP
insert IP on line between IP's
delete an IP
extend an IP
modify the strings z-value
modify the text labels
set height of text labels
set angle of text labels
open/close and Properties
toggle info panel
undo/redo
quit the edit session
finish the edit session



Only the options append, text, size and angle will be described in detail since all the other options are similar to the equivalent 3d string option (see the section 14.4.3 Edit 3d).

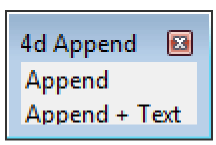
See the earlier section 14.4.1 Super String Edit - Common Information for general information about editing strings.

14.20.5.1 Append

The **append** operation for a 4d string is similar to the 3d string case except that a height and an optional text label is required for each new point.

Since in many cases a text label is not required at every point of a 4d string, there is an append option that asks for text at every point and another that does not ask.

The **4d append** walk-right is



The height question for each point is toggled on or off in the snaps menu just as it was for a 3d string.

The **append** option is terminated on selecting **cancel** from the **pick ops** menu or by selecting a new option from the **4d edit** menu.

See the next section 14.20.5.2 Text or return to 14.20.5 4d Edit - Old.

14.20.5.2 Text

The **text** option is used to modify the text label of any point in the 4d string.

After the **text** option is chosen, the user must select which point is going to have its text label modified. Once the point has been selected, an enter text typed-input box is displayed on the screen with the point's current text label placed in it.

The new text label is entered into the typed-input box, terminated with <enter>. The entered value is taken as the text label of the point in the 4d string and the string redrawn with the new text at that point. The typed-input box then disappears.

The **text** option is terminated on selecting **cancel** from the **pick ops** menu or by selecting a new option from the **4d edit** menu.

See the next section 14.20.5.3 Text ht or return to 14.20.5 4d Edit - Old.

14.20.5.3 Text ht

All the text labels in the 4d string have the same height (given in pixel or world units). The **text ht** option is used to modify this text label height.

After selecting the option, an enter value typed-input box is displayed on the screen with the string's current text label height placed in it.

The new text label height is entered into the typed-input box, terminated with <enter>.

The entered value is taken as the text label height for all the points in the 4d string and the string redrawn using the new height. The typed-input box then disappears.

The **text ht** option automatically terminates after use.

See the next section 14.20.5.4 Angle or return to 14.20.5 4d Edit - Old.

14.20.5.4 Angle

All the text labels in the 4d string are drawn rotated about their defining string point with the same rotation angle. The angle, in degrees, is measured in a counter-clockwise direction about the horizontal axis.

The **angle** option is used to modify the text rotation angle.

After selecting the option, an **enter value** typed-input box is displayed on the screen with the string's current text rotation angle placed in it.

The new text rotation angle is entered into the typed-input box, terminated with <enter>.

The entered value is taken as the text rotation angle for all the points in the 4d string and the string redrawn using the new angle. The typed-input box then disappears.

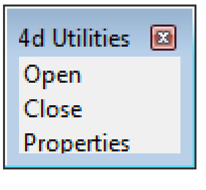
The **angle** option automatically terminates after use.

See the next section 14.20.5.5 Utilities or return to 14.20.5 4d Edit - Old.

14.20.5.5 Utilities

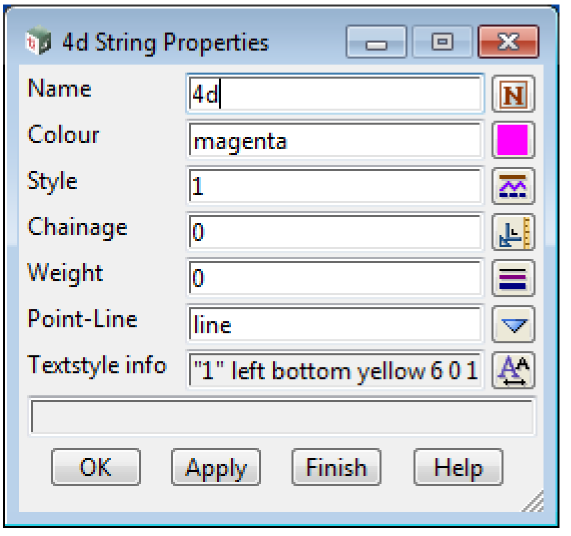
Position of option on menu: **Strings =>Editor**

The **4d utilities** walk-right is



14.20.5.5.1 Properties

Selecting Properties brings up the **4d String Properties** panel which is used to modify the string’s header information.



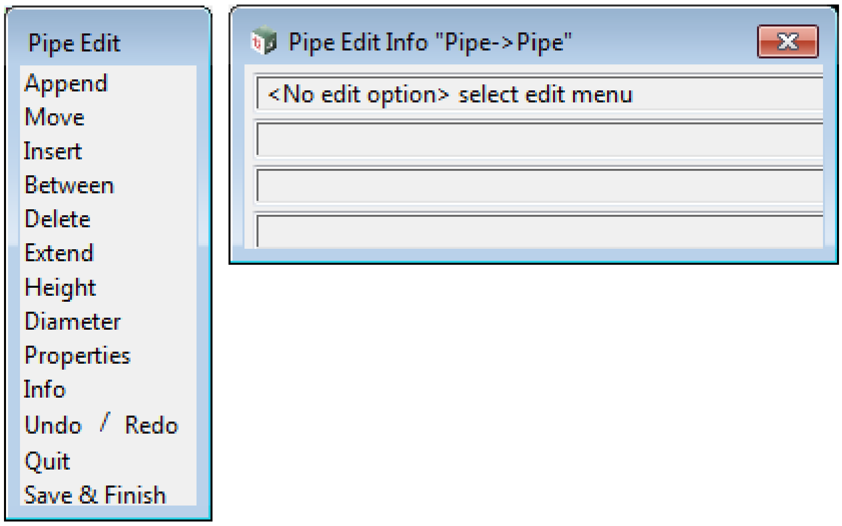
The fields in this panel are similar to those in the **create 4d string** panel and the 4d string editor options. The only new field is

Field Description	Type	Defaults	Pop-Up
OK/Apply	button		

for the string being edited, **OK** *sets the string with the values in the panel fields and removes the panel.* **Apply** *sets the string with the values in the panel fields and leaves the panel on the screen.*

14.20.6 Pipe Edit - Old

On picking a pipe string, the pipe edit menu and panel are placed on the screen.



The only difference between a 3d and a pipe string is that the pipe string has a diameter.

Hence most of the options in the **pipe edit** menu are similar to the **3d edit** options of the same name, and only the **diameter** and **Properties** options need to be discussed. See the section 14.4.3 Edit 3dfor information on the 3d string editor.

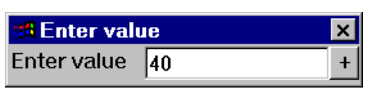
See the earlier section 14.4.1 Super String Edit - Common Information for general information about editing strings.

14.20.6.1 Diameter

The **diameter** option is used to modify the diameter of the pipe string.

After the **diameter** option is chosen, an enter value typed-input box is displayed on the screen with the string's current diameter.

The enter value typed-input box looks like:



The diameter is entered into the typed-input box, terminated with <enter>. The entered value is taken as the diameter of the pipe string. The typed-input box then disappears.

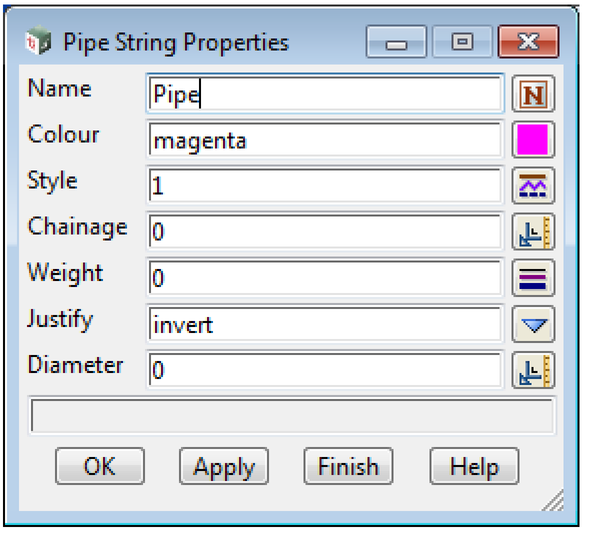
The **diameter** option automatically terminates and a new option is selected from the **pipe edit** menu.

Note - the pipe string has only one diameter for the entire string.

See the next section 14.20.6.2 Properties or return to 14.20.6 Pipe Edit - Old.

14.20.6.2 Properties

Selecting **Properties** brings up the **Pipe String Properties** panel which is used to modify the string's header information.



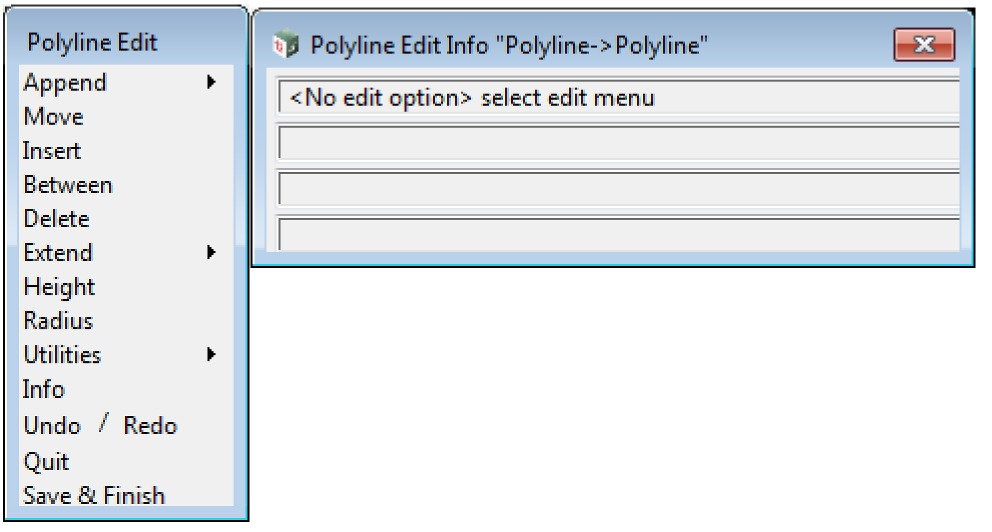
The fields in this panel are similar to those in the **create pipe string** panel and the 3d string editor options. The only new field is

Field Description	Type	Defaults	Pop-Up
OK/Apply	button		

for the string being edited, **OK** *sets the string with the values in the panel fields and removes the panel.* **Apply** *sets the string with the values in the panel fields and leaves the panel on the screen.*

14.20.7 Polyline Edit - Old

On picking a polyline string, the **polyline edit** menu and **polyline edit info** panel are placed on the screen.



The major difference between a polyline and a 3d string is that a polyline string can have an arc instead of a line joining adjacent string points. Hence most of the options in the **polyline edit** menu are similar to the **3d edit** options of the same name, and only the differences for each option will be discussed. See the section 14.4.3 Edit 3d for information on the 3d string editor.

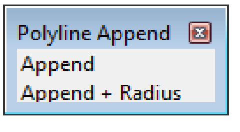
See the earlier section 14.4.1 Super String Edit - Common Information for general information about editing strings.

14.20.7.1 Append

The **append** operation for a polyline string is similar to the 3d string case except that an arc radius can be given for each new line segment.

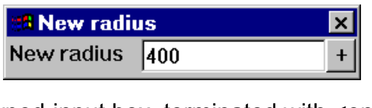
Since in many cases an arc radius is not required at every segment of a polyline string, there is an append option that asks for the radius at every point and another that does not ask.

The **append** walk-right is



If the **append + radius** option selected, then before each point is appended, an enter radius typed-input box is placed on the screen.

The enter radius typed-input box looks like



The radius is entered into the typed-input box, terminated with <enter>. The entered value is taken as the radius of the arc to the next polyline string point and the arc will be drawn correctly as the cursor is moved to the next point.

A radius value of 0 is taken to mean no arc.

The height question for each point is toggled on/off in the snaps menu just as it was for a 3d string.

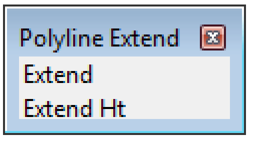
The **append** option is terminated on selecting **cancel** from the **pick ops** menu or by selecting a new option from the **polyline edit** menu.

See the next section 14.20.7.2 Extend or return to 14.20.7 Polyline Edit - Old.

14.20.7.2 Extend

There are two **extend** options for a polyline string - **extend=>extend** that is identical to the 3d string case where the z-value of the point being extended is kept constant, and a second option, **extend=>extend ht** where the z-value of the point being extended is linearly interpolated by the extension distance.

The **Extend** walk-right is



14.20.7.2.1 Extend

extend=>extend that is identical to the 3d string case where the z-value of the point being extended is kept constant**.**

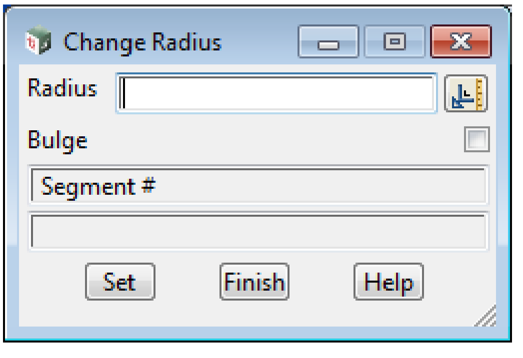
14.20.7.2.2 Extend by ht

extend=>extend ht not only moves the point but also linearly interpolates the z-value of the point being extended.

See the next section 14.20.7.3 Radius or return to 14.20.7 Polyline Edit - Old.

14.20.7.3 Radius

Selecting **radius** brings up the **change radius** panel which is used to modify the radius of any arc/ line joining adjacent polyline points.



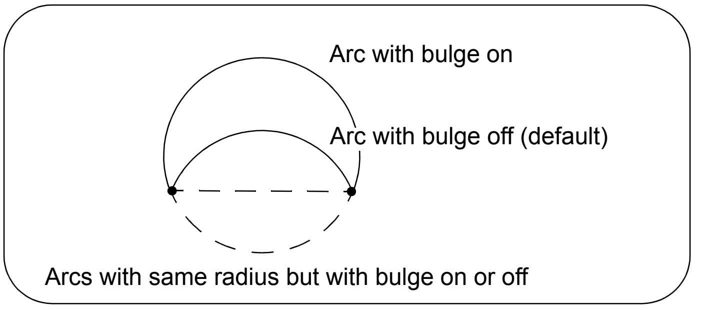
After selecting the **radius** option, the user selects the arc/straight to be modified and the current arc radius and bulge setting will be displayed in the **change radius** panel.

New values can then be entered and the arc modified by selecting the **set** button.

If the radius is positive, the arc is drawn from the start point to the next point on the polyline in a clockwise direction. If the radius is negative, the arc is drawn from the start point to the next point on the polyline in a counter-clockwise direction.

For a given radius (positive or negative), there are two possible cases for the arc- one where the arc is less than a semi-circle, the other when the arc is greater than a semi-circle.

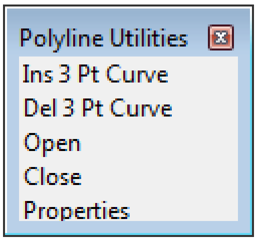
If bulge is turned on, the larger arc is used. The default is bulge turned off.



See the next section 14.20.7.4 Utilities or return to 14.20.7 Polyline Edit - Old.

14.20.7.4 Utilities

The **Utilities** walk-right menu contains a number of useful miscellaneous option for the polyline string. The menu is



Each of the new options will now be discussed.

14.20.7.4.1 Ins 3 Pt Curve

The **Ins 3 Pt Curve** option is used to insert a curve through three adjacent polyline points.

After selecting the option, the middle IP of the three adjacent polyline points is selected.

When the IP is accepted, the radius required to fit a curve through the IP and the two adjacent IP's is calculated, and this radius is then applied to the segments joining the adjacent IP's.

14.20.7.4.2 Del 3 Pt Curve

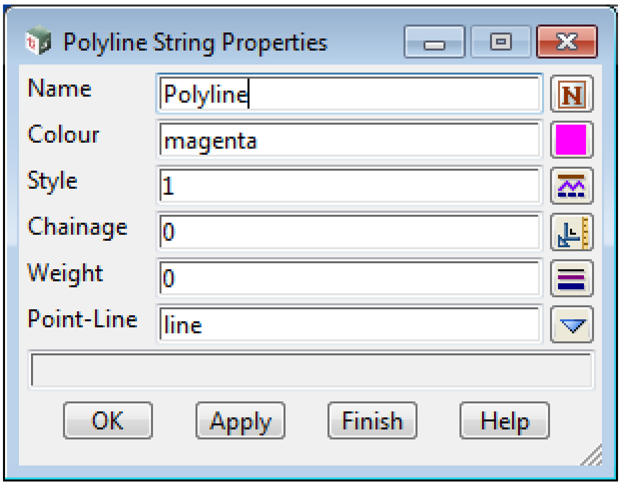
The **del 3 pt curve** option is used to delete the curves on either side of a polyline point.

After selecting the option, an IP is selected and when the IP is accepted, the radii of the segments on either side are set to zero.

Hence the curves on either side of the IP are effectively removed.

14.20.7.4.3 Properties

Selecting **Properties** brings up the **Polyline String Properties** panel which is used to modify the string's header information.



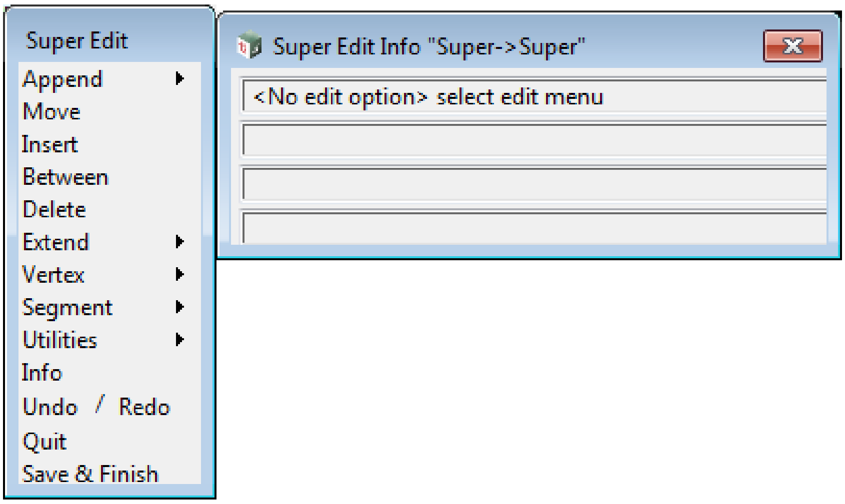
The fields in this panel are similar to those in the **create polyline string** panel and the polyline string editor options. The only new field is

Field Description	Type	Defaults	Pop-Up
OK/Apply	button		

for the string being edited, **OK** *sets the string with the values in the panel fields and removes the panel.* **Apply** *sets the string with the values in the panel fields and leaves the panel on the screen.*

14.20.8 Super Edit - Old

On picking a super string, the **Super Edit** menu and **Super Edit Info** panel are placed on the screen.



The super string is similar to a polyline string in that it can have an arc instead of a line joining adjacent string points. Hence most of the options in the **super edit** menu are similar to the **polyline edit** options of the same name, and only the differences for each option will be discussed.

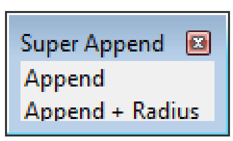
See the earlier section 14.4.1 Super String Edit - Common Information for general information about editing strings.

14.20.8.1 Append

The **Append** operation for a super is the same as the polyline case and an arc radius can be given for each new line segment.

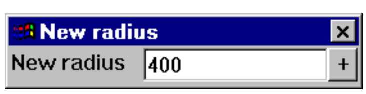
Since in many cases an arc radius is not required at every segment of a super string, there is an append option that asks for the radius at every point and another that does not ask.

The **Append** walk-right is



If the **Append + Radius** option selected, then before each vertex is appended, an enter radius typed-input box is placed on the screen.

The enter radius typed-input box looks like



The radius is entered into the typed-input box, terminated with <enter>. The entered value is taken as the radius of the arc to the next super string vertex and the arc will be drawn correctly as the cursor is moved to the next vertex.

A radius value of 0 is taken to mean *no arc*.

The height question for each vertex is toggled on/off in the snaps menu just as it was for a polyline string.

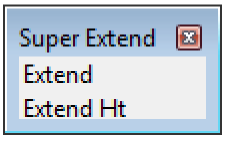
The **Append** option is terminated on selecting **Cancel** from the **Pick Ops** menu or by selecting a new option from the **Super Edit** menu

See the next section 14.20.8.2 Extend or return to 14.20.8 Super Edit - Old.

14.20.8.2 Extend

There are two **extend** options for a polyline string - **Extend=>Extend** that is identical to the 3d string case where the z-value of the point being extended is kept constant, and a second option, **Extend=>Extend Ht** where the z-value of the point being extended is linearly interpolated by the extension distance.

The **Extend** walk-right is



14.20.8.2.1 Extend

Extend=>Extend that is identical to the 3d string case where the z-value of the point being extended is kept constant.

14.20.8.2.2 Extend by Ht

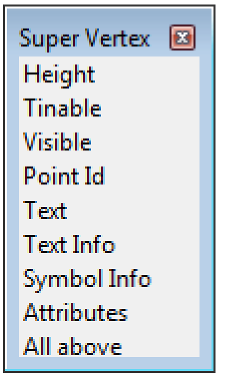
Extend=>Extend Ht not only moves the point but also linearly interpolates the z-value of the point being extended.

See the next section 14.20.8.3 Vertex or return to 14.20.8 Super Edit - Old.

14.20.8.3 Vertex

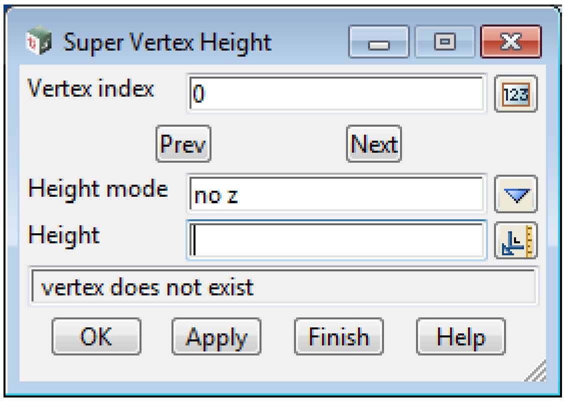
The **Vertex** menu contains options to modify information at any vertex of the super string.

The **Vertex** walk-right is



14.20.8.3.1 Height

Selecting **Vertex=>Height** brings up the **Super Vertex Height** panel which is used to set the height value for vertices.



As soon as **Height** is chosen, a *<Select vertex> [Picks][][Menu]* message is written to the Status Bar and vertices can be selected.

When the vertex to modify is selected, its *vertex number*, *height mode* and *height* are written to the appropriate panel fields. The values and modes can be changed and either **OK** or **Apply** selected to change the values of the vertex.

Another vertex can then be selected or the **Prev** and **Next** buttons used to move to adjacent vertices.

The fields and buttons used in the **Super Vertex Height** panel have the following functions.

Field Description Type Defaults Pop-Up

Vertex no. input selected vertex

if a vertex is selected, then its vertex number is displayed in this field. A number can also be typed in and any information in the panel will then be applied to that vertex if **OK** *or* **Apply** *is selected.*

Prev button

move to the previous vertex (predecessor). The information for the previous vertex is displayed in the panel fields.

Next button

move to the next vertex (successor). The information for the next vertex is displayed in the panel fields.

Height mode input no z, entire string, each vertex

if ***no z****, there is no z value for the vertex.*
if ***entire string****, then the string has the same z value for each vertex.*
if ***each vertex****, then each vertex has a separate z value.*

Height input height of vertex/string

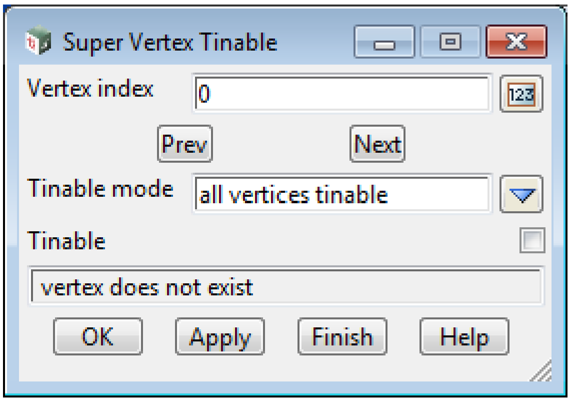
the height used for the vertex or for the entire string.

OK/Apply button

for the vertex being edited, **OK** *sets the vertex/string with the values in the panel fields and removes the panel.* **Apply** *sets the vertex/string with the values in the panel fields and leaves the panel on the screen.*

14.20.8.3.2 Tinable

Selecting **Vertex=>Tinable** brings up the **Super Vertex Tinable** panel which is used to set the tinable flag for vertices.



As soon as **Tinable** is chosen, a *<Select vertex> [Picks][][Menu]* message is written to the Status Bar and vertices can be selected.

When the vertex to modify is selected, its *vertex number* and *tinable flag* are displayed in the panel. The tinable flag can be changed and either **OK** or **Apply** selected to change the tinable flag of the vertex.

Another vertex can then be selected or the **Prev** and **Next** buttons used to move to adjacent vertices.

The fields and buttons used in the **Super Vertex Tinable** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Vertex no.	input	selected vertex	

if a vertex is selected, then its vertex number is displayed in this field. A number can also be typed in and any information in the panel will then be applied to that vertex if **OK** *or* **Apply** *is selected.*

Prev button

move to the previous vertex (predecessor). The information for the previous vertex is displayed in the panel fields.

Next button

move to the next vertex (successor). The information for the next vertex is displayed in the panel fields.

Tinable tick

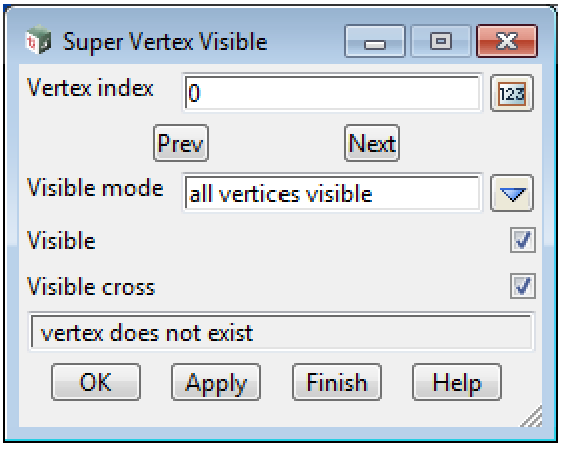
*if **ticked**, the vertex is included in tins.*
*if **not ticked**, then the vertex is ignored when triangulating.*

OK/Apply button

for the vertex being edited, **OK** *sets the vertex/string with the values in the panel fields and removes the panel.* **Apply** *sets the vertex/string with the values in the panel fields and leaves the panel on the screen.*

14.20.8.3.3 Visible

Selecting **Vertex=>Visible** brings up the **Super Vertex Visible** panel which is used to set the visibility flag for vertices.



As soon as **Visible** is chosen, a *<Select vertex> [Picks][][Menu]* message is written to the Status Bar and vertices can be selected.

When the vertex to modify is selected, its *vertex number* and *visibility flag* are displayed in the panel. The visibility flag can be changed and either **OK** or **Apply** selected to change the visibility flag of the vertex.

Another vertex can then be selected or the **Prev** and **Next** buttons used to move to adjacent vertices.

The fields and buttons used in the **Super Vertex Visible** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Vertex no.	input	selected vertex	

if a vertex is selected, then its vertex number is displayed in this field. A number can also be typed in and any information in the panel will then be applied to that vertex if **OK** *or* **Apply** *is selected.*

Prev button

move to the previous vertex (predecessor). The information for the previous vertex is displayed in the panel fields.

Next button

move to the next vertex (successor). The information for the next vertex is displayed in the panel fields.

Visible tick

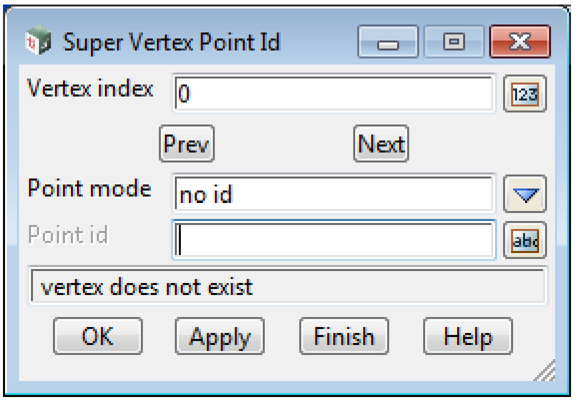
if ***ticked****, the vertex is visible.*
if ***not ticked****, then the vertex is invisible.*

OK/Apply button

for the vertex being edited, **OK** *sets the vertex/string with the values in the panel fields and removes the panel.* **Apply** *sets the vertex/string with the values in the panel fields and leaves the panel on the screen.*

14.20.8.3.4 Point no

Selecting **Vertex=>Point no** brings up the **Super Vertex Point Number** panel which is used to set the point numbers for vertices.



As soon as **Point no** is chosen, a *<Select vertex> [Picks][][Menu]* message is written to the Status Bar and vertices can be selected.

When the vertex to modify is selected, its *vertex number* and *point number* are displayed in the panel. The point number can be changed and either **OK** or **Apply** selected to change the point number of the vertex.

Another vertex can then be selected or the **Prev** and **Next** buttons used to move to adjacent vertices.

The fields and buttons used in the **Super Vertex Visible** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Vertex no.	input	selected vertex	

if a vertex is selected, then its vertex number is displayed in this field. A number can also be typed in and any information in the panel will then be applied to that vertex if **OK** *or* **Apply** *is selected.*

Prev button

move to the previous vertex (predecessor). The information for the previous vertex is displayed in the panel fields.

Next button

move to the next vertex (successor). The information for the next vertex is displayed in the panel fields.

Point no. input point no of vertex

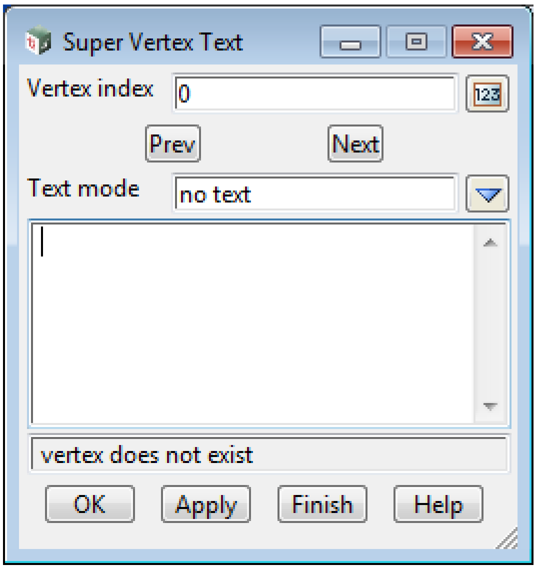
the point number used for the vertex.

OK/Apply button

for the vertex being edited, **OK** *sets the vertex/string with the values in the panel fields and removes the panel.* **Apply** *sets the vertex/string with the values in the panel fields and leaves the panel on the screen.*

14.20.8.3.5 Text

Selecting **Vertex=>Text** brings up the **Super Vertex Text** panel which is used to set the text for vertices.



As soon as **Text** is chosen, a *<Select vertex> [Picks][][Menu]* message is written to the Status Bar and vertices can be selected.

When the vertex to modify is selected, its *vertex number*, *text mode* and *text* are written to the appropriate panel fields. The values and modes can be changed and either **OK** or **Apply** selected to change the values of the vertex.

Another vertex can then be selected or the **Prev** and **Next** buttons used to move to adjacent vertices.

The fields and buttons used in the **Super Vertex Text** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Vertex no.	input	selected vertex	

if a vertex is selected, then its vertex number is displayed in this field. A number can also be typed in and any information in the panel will then be applied to that vertex if **OK** *or* **Apply** *is selected.*

Prev button

move to the previous vertex (predecessor). The information for the previous vertex is displayed in the panel fields.

Next button

move to the next vertex (successor). The information for the next vertex is displayed in the panel fields.

Text mode input no text, entire string, each vertex

*if **no text**, there is no text for the vertex.*
*if **entire string**, then the string has the same text for each vertex.*
*if **each vertex**, then each vertex has a separate text value.*

Text input text of vertex/string

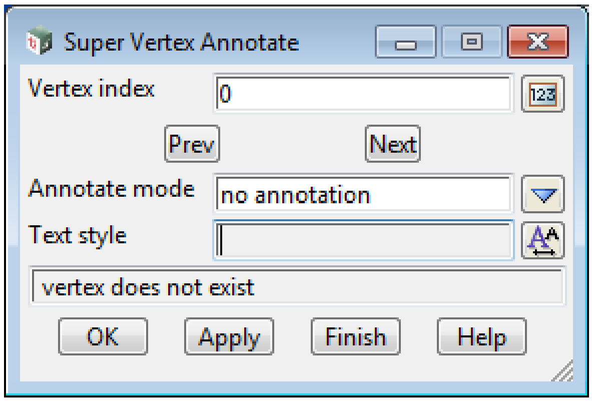
the text used for the vertex or for the entire string.

OK/Apply button

for the vertex being edited, **OK** *sets the vertex/string with the values in the panel fields and removes the panel.* **Apply** *sets the vertex/string with the values in the panel fields and leaves the panel on the screen.*

14.20.8.3.6 Text Info

Selecting **Vertex=>Text info** brings up the **Super Vertex Annotate** panel which is used to set the annotation styles for the text at vertices.



As soon as **Text info** is chosen, a *<Select vertex> [Picks][][Menu]* message is written to the Status Bar and vertices can be selected.

When the vertex to modify is selected, its *vertex nunber*, *annotate mode* and *annotation information* are written to the appropriate panel fields. The values and modes can be changed and either **OK** or **Apply** selected to change the values of the vertex.

Another vertex can then be selected or the **Prev** and **Next** buttons used to move to adjacent vertices.

The fields and buttons used in the **Super Vertex Annotate** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Vertex no.	input	selected vertex	

if a vertex is selected, then its vertex number is displayed in this field. A number can also be typed in and any information in the panel will then be applied to that vertex if **OK** *or* **Apply** *is selected.*

Prev button

move to the previous vertex (predecessor). The information for the previous vertex is displayed in the panel fields.

Next button

move to the next vertex (successor). The information for the next vertex is displayed in the panel fields.

Annotate mode input no annotation, entire string, each vertex

*if **no annotation**, then the text at the vertex is not displayed.*
*if **entire string**, then the same annotation settings are used for each vertex.*
*if **each vertex**, then each vertex has separate annotations settings.*

Text style input 1 available text styles

text style for the text at the vertex.

Text units input pixels pixels, world

units for the height of the text.

Height (u) input

height of the text (in text units).

X factor input 1

x factor of the text.

Offset (u) input 0

distance (in text units) to offset the text from its (x,y) placement position.

Raise (u) input 0

distance (in text units) to raise the text above the line for its (x,y) placement position.

Justify input bottom-left bot-left/cent/right, mid-left/cent/right top-left/cent/right

text justification (about the offset position).

Angle input

angle of the text.

Slant input 0

slant, in degrees, of the text.

Colour input available colours

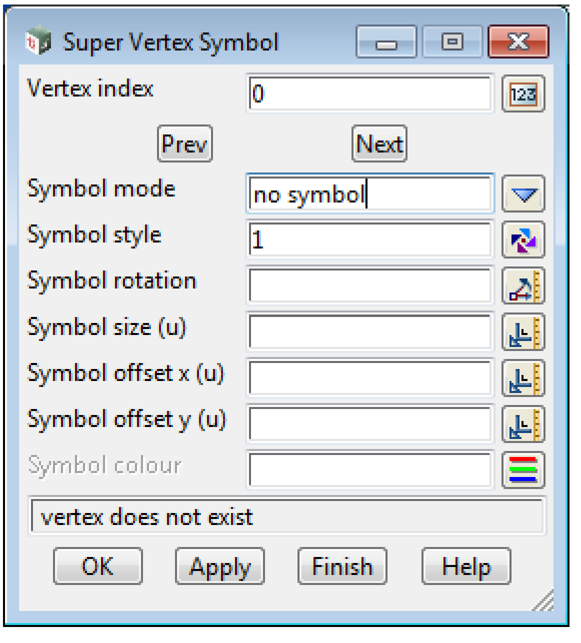
colour of the text.

OK/Apply button

for the vertex being edited, **OK** *sets the vertex/string with the values in the panel fields and removes the panel.* **Apply** *sets the vertex/string with the values in the panel fields and leaves the panel on the screen.*

14.20.8.3.7 Symbol Info

Selecting **Vertex=>Symbol info** brings up the **Super Vertex Symbol** panel which is used to set symbols and their display parameters at vertices.



As soon as **Symbol info** is chosen, a *<Select vertex> [Picks][][Menu]* message is written to the Status Bar and vertices can be selected.

When the vertex to modify is selected, its *vertex nunber*, *symbol mode* and *symbol information* are written to the appropriate panel fields. The values and modes can be changed and either **OK** or **Apply** selected to change the values for the vertex.

Another vertex can then be selected or the **Prev** and **Next** buttons used to move to adjacent vertices.

The fields and buttons used in the **Super Vertex Symbol** panel have the following functions.

Field Description Type Defaults Pop-Up

Vertex no. input selected vertex

if a vertex is selected, then its vertex number is displayed in this field. A number can also be *typed in and any information in the panel will then be applied to that vertex if* **OK** *or* **Apply** *is selected.*

Prev button

move to the previous vertex (predecessor). The information for the previous vertex is displayed in the panel fields.

Next button

move to the next vertex (successor). The information for the next vertex is displayed in the panel fields.

Symbol mode input no symbol, entire string, each vertex

if ***no symbol****, then there is no symbol at the vertex.*
if ***entire string****, then the same symbol and settings are used for each vertex.*
if ***each vertex****, then each vertex has separate symbols and settings.*

Symbol style input 1 available line styles

line style for the symbol at the vertex.

Symbol rotation input

rotation angle of the symbol.

Symbol size (u) input

size of the symbol (in xxx units).

Symbol offset (u) input 0

distance (in xxx units) to offset the symbol from its (x,y) placement position.

Symbol raise (u) input 0

distance (in xxx units) to raise the symbol above the line for its (x,y) placement position.

Symbol colour input available colours

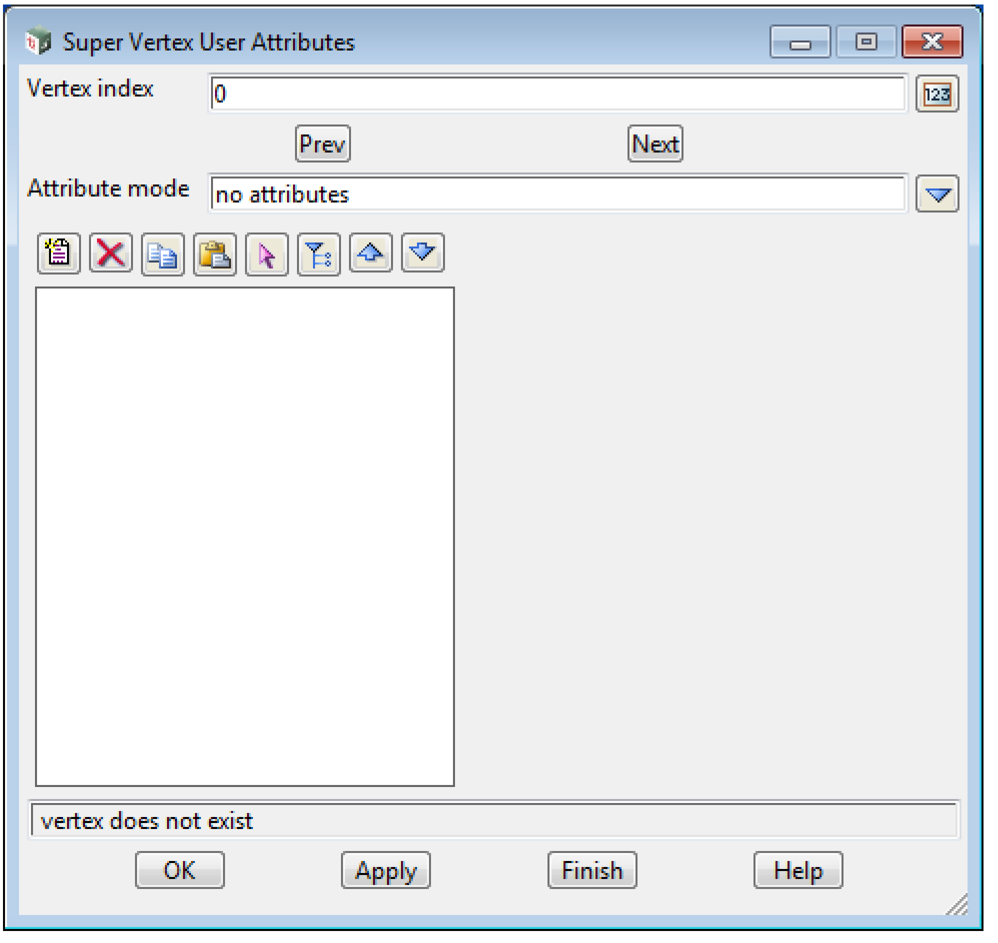
colour of the symbol if none is defined in the symbol definition

OK/Apply button

for the vertex being edited, **OK** *sets the vertex/string with the values in the panel fields and removes the panel.* **Apply** *sets the vertex/string with the values in the panel fields and leaves the panel on the screen.*

14.20.8.3.8 Attributes

Selecting **Vertex=>Attributes** brings up the **Super Vertex User Attributes** panel which is used to display and edit user defined attributes at vertices of the super string.



The **Prev** and **Next** buttons are used to cycle through and display the attributes for each vertex in the super string in the *Name/Type/Data* grid.

The data in the *Name/Type/Data* grid can be deleted, modified or added to and then updated for the vertex using the **OK** or **Apply** button.

The fields and buttons used in the **Super Vertex User Attributes** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Vertex no.	input	selected vertex	

when the option starts, the user attributes for the first vertex is displayed. The **Next** *and* **Prev** *buttons will move onto other vertices. Also a number can be typed into the field and any information in the panel will then be applied to that vertex if* **OK** *or* **Apply** *is selected. Typing <Enter> after entering a number will go to that vertex number and display the attributes.*

Prev button

move to the previous vertex (predecessor). The information for the previous vertex is displayed in the panel fields.

Next button

move to the next vertex (successor). The information for the next vertex is displayed in the panel fields.

Attribute mode choice box no attributes, each vertex

if ***no attributes****, then no vertices have user attributes.*
if ***each vertex****, then each vertex can have user attributes.*

Name/Type/Data Grid

Name input

name for the user attribute. This must be unique for all attributes at this vertex.

Type choice box integer, real, text

type of the attribute.

Data input

value for the attribute.

OK/Apply button

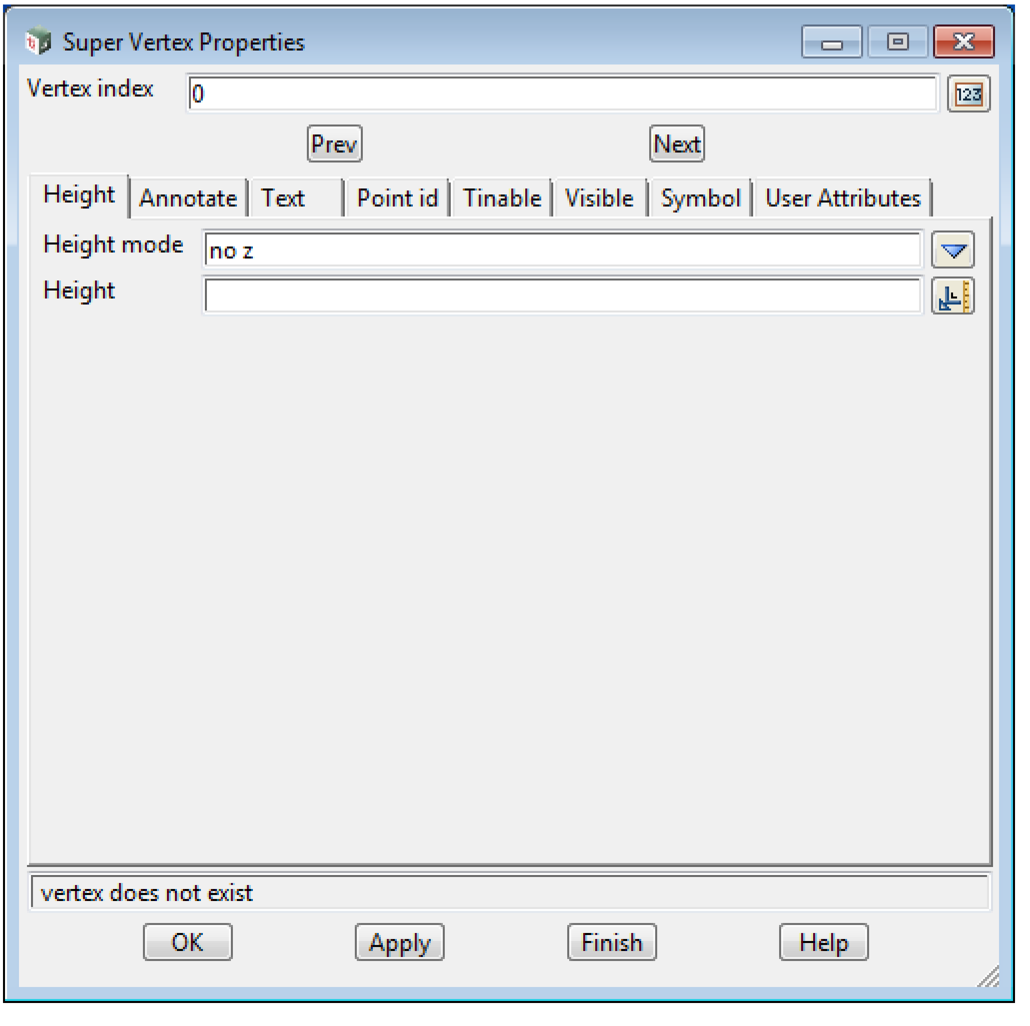
for the vertex being edited, **OK** *sets the vertex with the values in the panel fields and removes the panel.* **Apply** *sets the vertex with the values in the panel fields and leaves the panel on the screen.*

14.20.8.3.9 Vertex - All above

Selecting **Vertex=>All above** brings up the **Super Vertex Properties** panel which is used to display all the properties for a vertex.

This option is also available from the *Strings* menu

Position of option on menu: **Strings =>Properties =>Vertex (all)**



As soon as **All above** is chosen, a *<Select vertex> [Picks][][Menu]* message is written to the Status Bar and vertices can be selected.

When the vertex to modify is selected, its *vertex nunber* and all other *information* are written to the appropriate panel fields. The values and modes can be changed and either **OK** or **Apply** selected to change the values for the vertex.

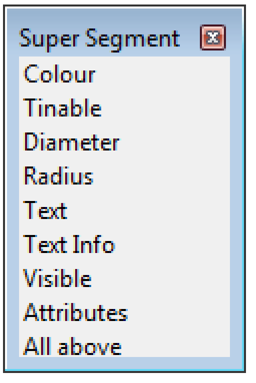
The fields in the **Super Vertex Properties** panel have already been described in the other Vertex options and so will not be described again.

Please continue to the next section 14.4.10.5 Segment Toolbar.

14.20.8.4 Segment

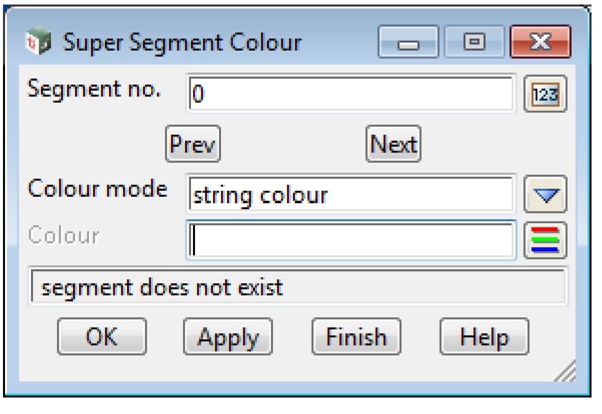
The **Segment** menu contains options to modify information at any segment of the super string.

The **Segment** walk-right is



14.20.8.4.1 Colour

Selecting **Segment=>Colour** brings up the **Super Segment Colour** panel which is used to set the colour of the string segments.



As soon as **Colour** is chosen, a *<Select segment> [Picks][][Menu]* message is written to the Status Bar and segments can be selected.

When the segment to modify is selected, its *segment number*, *colour mode* and *colour* are written to the appropriate panel fields. The values and modes can be changed and either **OK** or **Apply** selected to change the values of the segment.

Another segment can then be selected or the **Prev** and **Next** buttons used to move to adjacent segments. The appropriate segment highlights when the **Prev** and **Next** buttons are used.

The fields and buttons used in the **Super Segment Colour** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Segment no.	input	selected vertex	

if a segment is selected, then its segment number is displayed in this field. A number can also be typed in and any information in the panel will then be applied to that segment if **OK** *or*

Apply *is selected.*

Prev button

move to the previous segment (predecessor). The information for the previous segment is displayed in the panel fields.

Next button

move to the next segment (successor). The information for the next segment is displayed in the panel fields.

Colour mode input string colour, each segment

if ***no z****, there is no z value for the vertex.*
if ***string colour****, then all the segments in the string have the same colour.*
if ***each segment****, then each segment has a separate colour.*

Colour input colour of segment/stringavailable colours

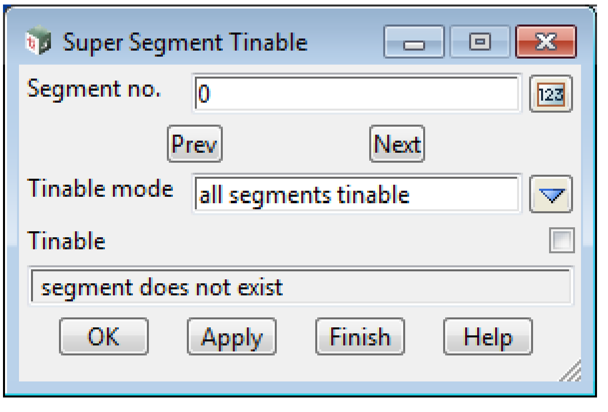
the colour used for the segment or for the entire string.

OK/Apply button

for the segment being edited, **OK** *sets the segment/string with the values in the panel fields and removes the panel.* **Apply** *sets the segment/string with the values in the panel fields and leaves the panel on the screen.*

14.20.8.4.2 Tinable

Selecting **Segment=>Tinable** brings up the **Super Segment Tinable** panel which is used to set the tinable flag for segments.



As soon as **Tinable** is chosen, a *<Select segment> [Picks][][Menu]* message is written to the Status Bar and vertices can be selected.

When the segment to modify is selected, its *segment number* and *tinable flag* are displayed in the panel. The tinable flag can be changed and either **OK** or **Apply** selected to change the tinable flag of the segment.

Another segment can then be selected or the **Prev** and **Next** buttons used to move to adjacent segments.

The fields and buttons used in the **Super Segment Tinable** panel have the following functions.

Field Description Type Defaults Pop-Up

Segment no. input selected segment

if a segment is selected, then its segment number is displayed in this field. A number can also be typed in and any information in the panel will then be applied to that segment if **OK** *or* **Apply** *is selected.*

Prev button

move to the previous segment (predecessor). The information for the previous segment is displayed in the panel fields.

Next button

move to the next segment (successor). The information for the next segment is displayed in the panel fields.

Tinable tick

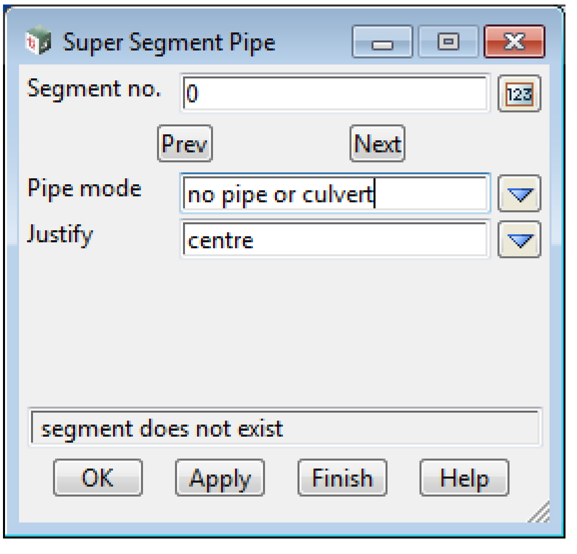
if ***ticked****, the triangulation process tries to preserve the segment as a side of a triangle in the tin.*

OK/Apply button

for the segment being edited, **OK** *sets the segment/string with the values in the panel fields and removes the panel.* **Apply** *sets the segment/string with the values in the panel fields and leaves the panel on the screen.*

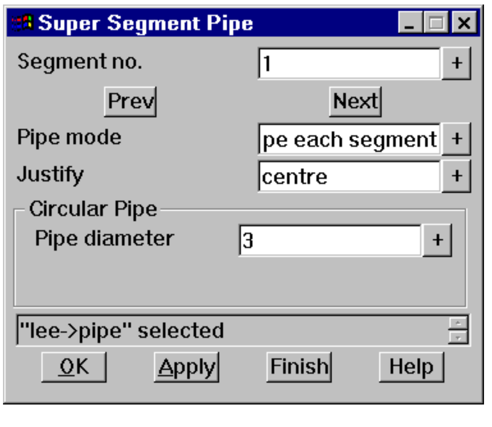
14.20.8.4.3 Diameter

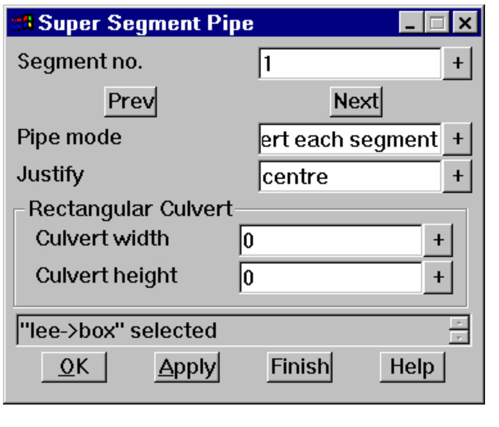
The segments of a super string can have either a pipe or box cross section, or none.
Selecting **Segment=>Diameter** brings up the **Super Segment Pipe** panel which is used to set the pipe mode and size for the string segments.



As soon as **Diameter** is chosen, a *<Select segment> [Picks][][Menu]* message is written to the Status Bar and segments can be selected.

When the segment to modify is selected, its *segment number*, *pipe mode* and *information* are written to the appropriate panel fields and the **Super Segment Pipe** panel will change depending on whether it was a pip or box cross-section for the string.





After any panel fields are modified, selecting either **OK** or **Apply** will change the pipe information for the segment.

Another segment can then be selected or the **Prev** and **Next** buttons used to move to adjacent segments. The appropriate segment highlights when the **Prev** and **Next** buttons are used.

The fields and buttons used in the **Super Segment Pipe** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Segment no.	input	selected vertex	

if a segment is selected, then its segment number is displayed in this field. A number can also be typed in and any information in the panel will then be applied to that segment if **OK** *or* **Apply** *is selected.*

Prev button

move to the previous segment (predecessor). The information for the previous segment is displayed in the panel fields.

Next button

move to the next segment (successor). The information for the next segment is displayed in the panel fields.

Pipe mode input — no pipe or box, pipe entire string, pipe each segment, box entire string, box each segment

if ***no pipe or box****, there is no z value for the vertex.*
if ***string colour****, then all the segments in the string have the same colour.*
if ***each segment****, then each segment has a separate colour.*

Justify input — invert, centre, overt

justification of the pipe/box with respect to the co-ordinates given for the vertices of the super string.
rotation angle of the symbol.

Pipe diameter input

diameter of the pipe in world units.

Box width input

width of the box section in world units.

Box height input

height of the box section in world units.

OK/Apply button

for the segment being edited, **OK** *sets the segment/string with the values in the panel fields and removes the panel.* **Apply** *sets the segment/string with the values in the panel fields and leaves the panel on the screen.*

14.20.8.4.4 Radius

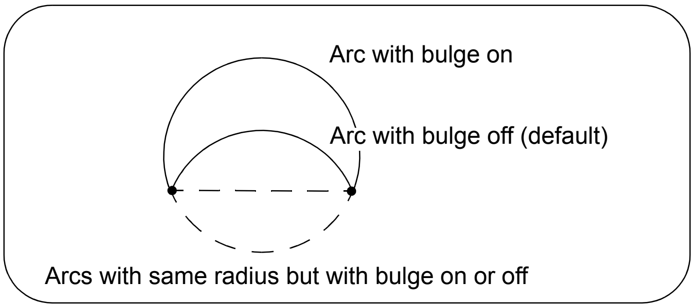
When viewed in plan, the segments of a super string can be joined by string lines or arcs.
If the radius is positive, the arc is drawn from the start vertex to the end vertex of the segment in a clockwise direction. If the radius is negative, the arc is drawn from the start vertex to the end vertex on the segment in a counter-clockwise direction.

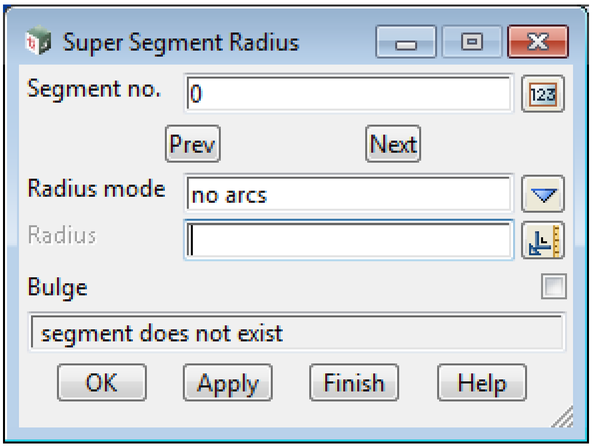
For a given radius (positive or negative), there are two possible cases for the arc- one where the arc is less than a semi-circle, the other when the arc is greater than a semi-circle.

If bulge is turned *on*, the larger arc is used. The default is bulge turned off.

A zero radius is interpreted to be a just a straight line segment with no arc.

Selecting **Segment=>Radius** brings up the **Super Segment Radius** panel which is used to set the plan radius of the segment.





As soon as **Radius** is chosen, a <*Select segment*> *[Picks][][Menu]* message is written to the Status Bar and segments can be selected.

When the segment to modify is selected, its *segment number*, *radius* and *bulge* are written to the appropriate panel fields. The values and modes can be changed and either **OK** or **Apply** selected to change the values of the segment.

Another segment can then be selected or the **Prev** and **Next** buttons used to move to adjacent segments. The appropriate segment highlights when the **Prev** and **Next** buttons are used.

The fields and buttons used in the **Super Segment Radius** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Segment no.	input	selected vertex	

if a segment is selected, then its segment number is displayed in this field. A number can also be typed in and any information in the panel will then be applied to that segment if **OK** *or* **Apply** *is selected.*

Prev	button		

move to the previous segment (predecessor). The information for the previous segment is displayed in the panel fields.

Next	button		

move to the next segment (successor). The information for the next segment is displayed in the panel fields.

Colour mode	input		string colour, each segment

if ***no z****, there is no z value for the vertex.*
if ***string colour****, then all the segments in the string have the same colour.*
if ***each segment****, then each segment has a separate colour.*

Radius	tick		

if ***ticked****, the larger arc is used.*
if not-ticked, the smaller arc is used.

Bulge input 0

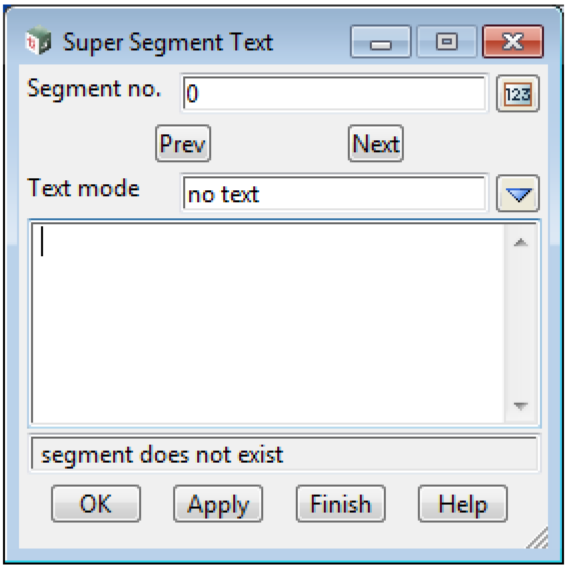
radius of the segment arc. A radius of 0 mean no arc.

OK/Apply button

for the segment being edited, **OK** *sets the segment/string with the values in the panel fields and removes the panel.* **Apply** *sets the segment/string with the values in the panel fields and leaves the panel on the screen.*

14.20.8.4.5 Text

Selecting **Segment=>Text** brings up the **Super Segment Text** panel which is used to set the text for segments.



As soon as **Text** is chosen, a *<Select segment> [Picks][][Menu]* message is written to the Status Bar and vertices can be selected.

When the segment to modify is selected, its *segment number*, *text mode* and *text* are written to the appropriate panel fields. The values and modes can be changed and either **OK** or **Apply** selected to change the values of the segment.

Another segment can then be selected or the **Prev** and **Next** buttons used to move to adjacent segments.

The fields and buttons used in the **Super Segment Text** panel have the following functions.

Field Description Type Defaults Pop-Up

Segment no. input selected vertex

if a segment is selected, then its segment number is displayed in this field. A number can also be typed in and any information in the panel will then be applied to that segment if **OK** *or* **Apply** *is selected.*

Prev button

move to the previous segment (predecessor). The information for the previous segment is displayed in the panel fields.

Next button

move to the next segment (successor). The information for the next segment is displayed in the panel fields.

Text mode input no text, entire string, each segment

if ***no text****, there is no text for the segment.*
if ***entire string****, then the string has the same text for each segment.*
if ***each segment****, then each segment has a separate text value.*

Text input text of segment/string

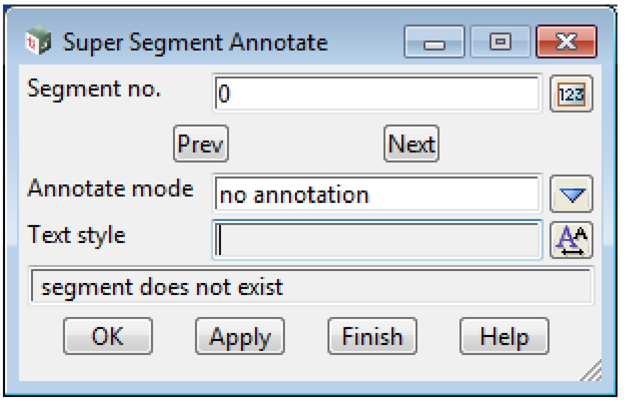
the text used for the segment or for the entire string.

OK/Apply button

for the segment being edited, **OK** *sets the segment/string with the values in the panel fields and removes the panel.* **Apply** *sets the segment/string with the values in the panel fields and leaves the panel on the screen.*

14.20.8.4.6 Text Info

Selecting **Segment=>Text info** brings up the **Super Segment Annotate** panel which is used to set the annotation styles for the text at segments.



As soon as **Text info** is chosen, a *<Select segment> [Picks][][Menu]* message is written to the Status Bar and segments can be selected.

When the segment to modify is selected, its *segment nunber*, *annotate mode* and *annotation information* are written to the appropriate panel fields. The values and modes can be changed and either **OK** or **Apply** selected to change the values of the segment.

Another segment can then be selected or the **Prev** and **Next** buttons used to move to adjacent segments.

The fields and buttons used in the **Super Segment Annotate** panel have the following functions.

Field Description Type Defaults Pop-Up

Segment no. input selected vertex

if a segment is selected, then its segment number is displayed in this field. A number can also be typed in and any information in the panel will then be applied to that segment if **OK** *or* **Apply** *is selected.*

Prev button

move to the previous segment (predecessor). The information for the previous segment is displayed in the panel fields.

Next button

move to the next segment (successor). The information for the next segment is displayed in the panel fields.

Annotate mode input no annotation, entire string, each segment

if ***no annotation****, then the text at the segment is not displayed.*
if ***entire string****, then the same annotation settings are used for each segment.*
if ***each vertex****, then each segment has separate annotations settings.*

Textstyle info input

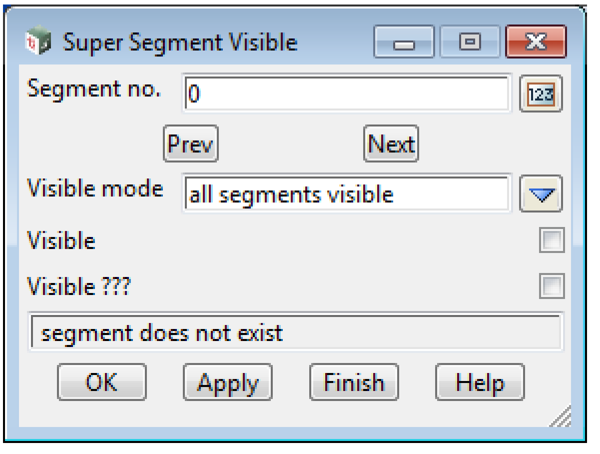
textstyle information.

OK/Apply button

for the segment being edited, **OK** *sets the segment/string with the values in the panel fields and removes the panel.* **Apply** *sets the segment/string with the values in the panel fields and leaves the panel on the screen.*

14.20.8.4.7 Visible

Selecting **Segment=>Visible** brings up the **Super Segment Visible** panel which is used to set the visibility flag for segments.



As soon as **Visible** is chosen, a <*Select segment*> *[Picks][][Menu]* message is written to the Status Bar and segments can be selected.

When the segment to modify is selected, its *segment number* and *visibility flag* are displayed in the panel. The visibility flag can be changed and either **OK** or **Apply** selected to change the visibility flag of the segment.

Another segment can then be selected or the **Prev** and **Next** buttons used to move to adjacent segments.

The fields and buttons used in the **Super Segment Visible** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Segment no.	input	selected vertex	

if a segment is selected, then its segment number is displayed in this field. A number can also be typed in and any information in the panel will then be applied to that segment if **OK** *or* **Apply** *is selected.*

Prev button

move to the previous segment (predecessor). The information for the previous segment is displayed in the panel fields.

Next button

move to the next segment (successor). The information for the next segment is displayed in the panel fields.

Visible tick

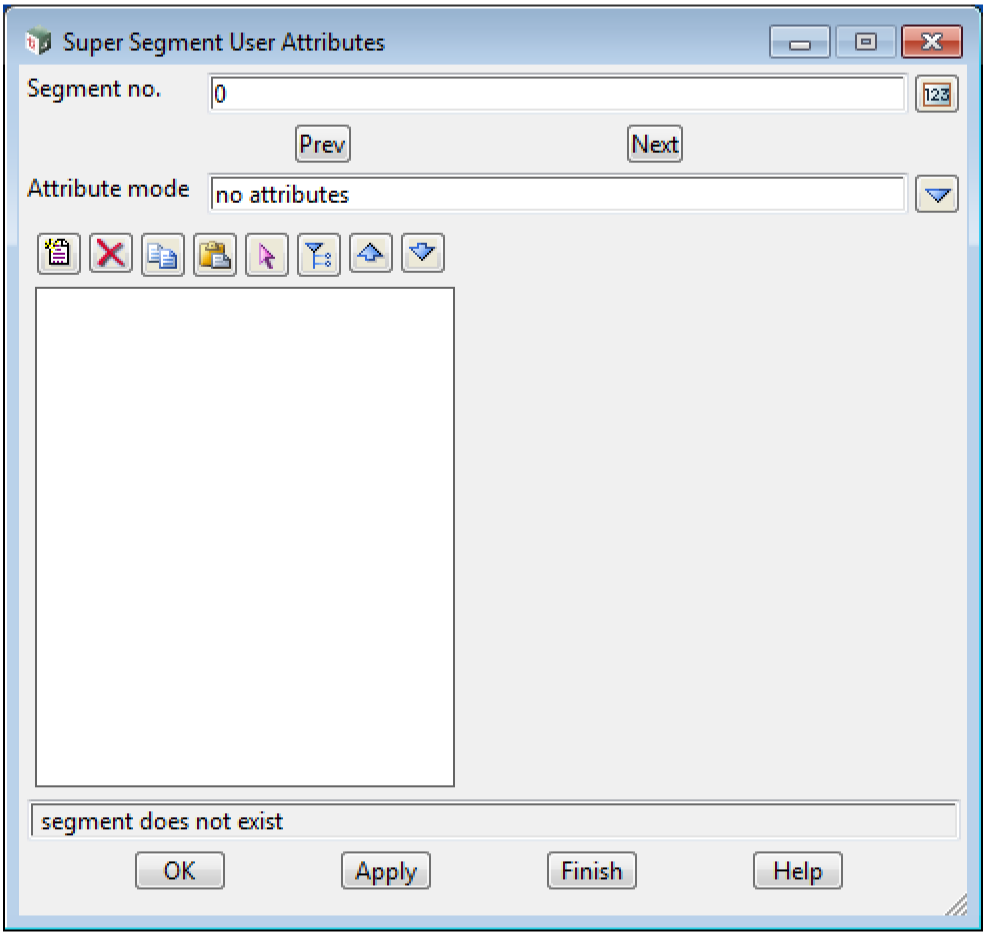
*if **ticked**, the segment is visible.*
*if **not ticked**, then the segment is invisible.*

OK/Apply button

for the segment being edited, **OK** *sets the segment/string with the values in the panel fields and removes the panel.* **Apply** *sets the segment/string with the values in the panel fields and leaves the panel on the screen.*

14.20.8.4.8 Attributes

Selecting **Segment=>Attributes** brings up the **Super Segment User Attributes** panel which is used to display and edit user defined attributes at segments of the super string.



The **Prev** and **Next** buttons are used to cycle through and display the attributes for each segment in the super string in the *Name/Type/Data* grid.

The data in the *Name/Type/Data* grid can be deleted, modified or added to and then updated for the segment using the **OK** or **Apply** button.

The fields and buttons used in the **Super Segment User Attributes** panel have the following functions.

Field Description Type Defaults Pop-Up

Segment no.

when the option starts, the user attributes for the first segment is displayed. The **Next** *and* **Prev** *buttons will move onto other segments. Also a number can be typed into the field and any information in the panel will then be applied to that segment if* **OK** *or* **Apply** *is selected. Typing <Enter> after entering a number will go to that segment number and display the attributes.*

Prev button

move to the previous segment (predecessor). The information for the previous segment is displayed in the panel fields.

Next button

move to the next segment (successor). The information for the next segment is displayed in the panel fields.

Attribute mode choice box no attributes, each segment

if ***no attributes****, then no segments have user attributes.*

*if **each segment**, then each segment can have user attributes.*

Name/Type/Data Grid

Name input

name for the user attribute. This must be unique for all attributes at this segment.

Type choice box integer, real, text

type of the attribute.

Data input

value for the attribute.

OK/Apply button

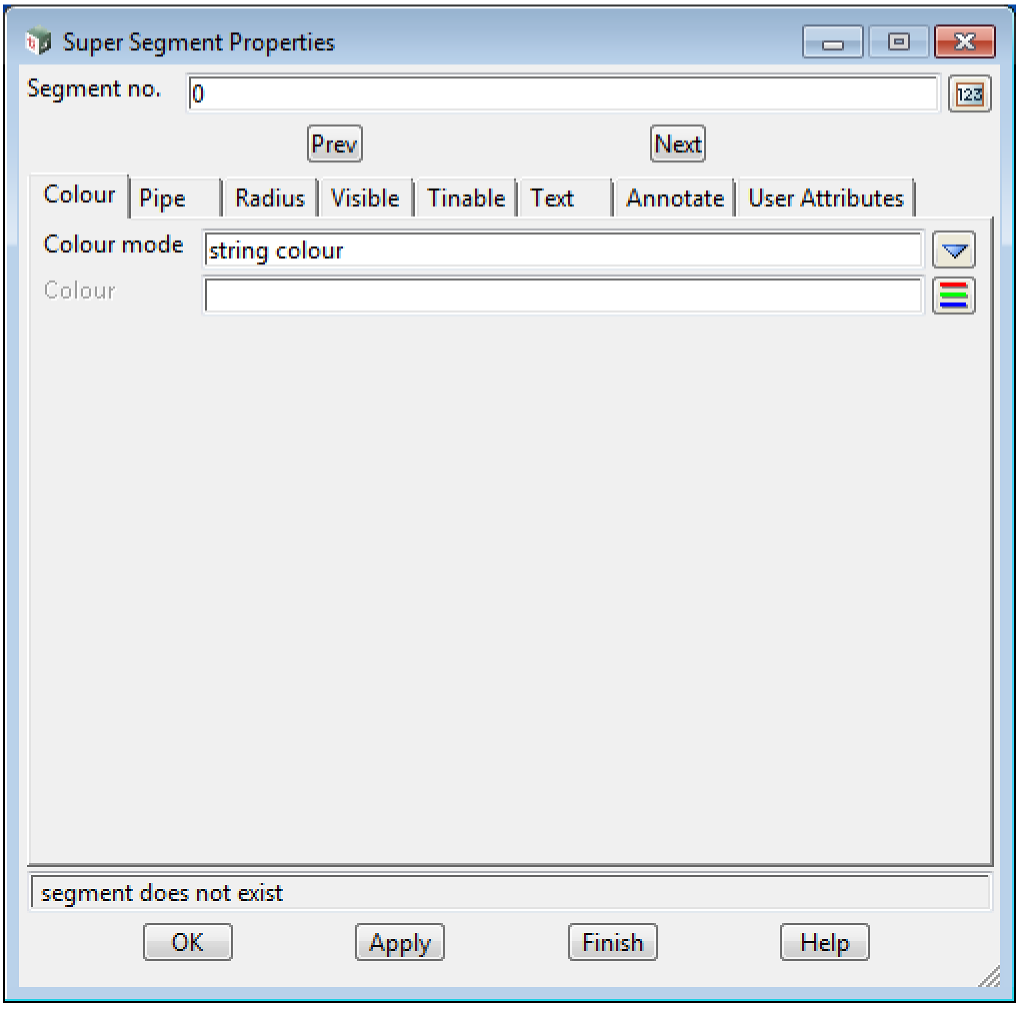
for the segment being edited, **OK** *sets the segment with the values in the panel fields and removes the panel.* **Apply** *sets the segment with the values in the panel fields and leaves the panel on the screen.*

14.20.8.4.9 Segment - All above

Selecting **Segment=>All above** brings up the **Super Segment Properties** panel which is used to display all the properties of a segment.

This option is also available from the *Strings* menu

Position of option on menu: **Strings =>Properties =>Segment (all)**



As soon as **All above** is chosen, a *<Select segment> [Picks][][Menu]* message is written to the Status Bar and segments can be selected.

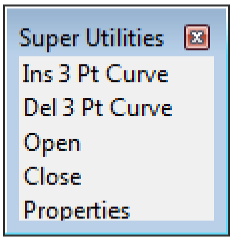
When the segment to modify is selected, its *segment nunber* and all other *information* are written to the appropriate panel fields. The values and modes can be changed and either **OK** or **Apply** selected to change the values for the segment.

The fields in the **Super Segment Properties** panel have already been described in the other Segment options and so will not be described again.

See the next section 14.20.8.5 Utilities or return to 14.20.8 Super Edit - Old.

14.20.8.5 Utilities

The **Utilities** walk-right menu contains a number of useful miscellaneous option for the super string. The menu is



Each of the new options will now be discussed.

14.20.8.5.1 Ins 3 Pt Curve

The **Ins 3 Pt Curve** option is used to insert a curve through three adjacent super string vertices.

After selecting the option, the middle vertex of the three adjacent super string vertices is selected.

When the vertex is accepted, the radius required to fit a curve through the vertex and the two adjacent vertices is calculated, and this radius is then applied to the segments joining the adjacent vertices.

14.20.8.5.2 Del 3 Pt Curve

The **del 3 pt curve** option is used to delete the curves on either side of a super string vertex.

After selecting the option, a vertex is selected and when the vertex is accepted, the radii of the segments on either side are set to zero.

Hence the curves on either side of the vertex are effectively removed.

14.20.8.5.3 Open

If the string is closed, selecting the **open** option removes the segment between the 1st and last vertex of the super string.

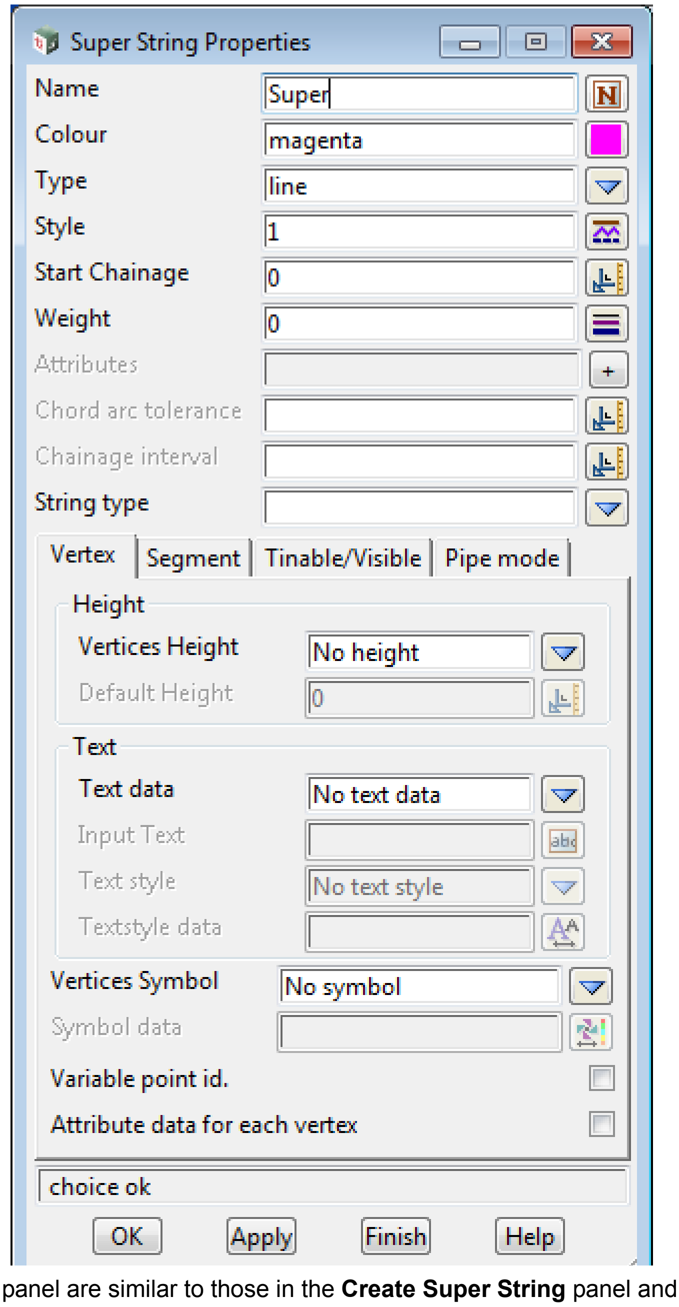
If the string is not closed, the **open** option does nothing.

14.20.8.5.4 Close

Selecting the **close** option adds a segment between the 1st and the last vertices of the super string. Note that unlike other strings, no extra vertices are added.

14.20.8.5.5 Properties

Selecting **Properties** brings up the **Super String Properties** panel which is used to modify the string's header information.



The fields in this panel are similar to those in the **Create Super String** panel and the Super String Editor options. The only new field is

Field Description	Type	Defaults	Pop-Up
OK/Apply	button		

for the string being edited, **OK** *sets the string with the values in the panel fields and removes the panel.* **Apply** *sets the string with the values in the panel fields and leaves the panel on the screen.*

15 CAD

In ***12d Model*** there are CAD options which are available under both the **CAD** menu and flyouts on the *CAD Toolbar* which by default, is on the left hand side of the ***12d Model*** screen.

The **CAD** options create and edit strings using a variety of methods and are fully described in the section 15.2 CAD Options.

The *strings* created by the *CAD menu* or *CAD toolbars* use values in the three *controlbars* (**Cad Controlbar**, **Symbol Controlbar**, and **Text Controlbar**) to define string attributes such as name, model and colour.

So the controlbars will be described in full in the section 15.1 Controlbars before the CAD options.

For basic information on toolbars and ControlBars, see the section 4.3.5 Toolbars and Controlbars.

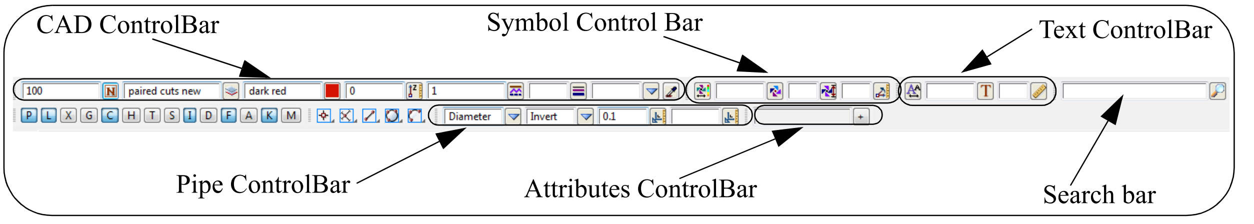
The **CAD** options create various elements using a number of methods. These options make use of **Tool bars** and **Control bars**. Tool bars just have icons on them but Control bars have icons and also controls such as a model box on them. The method groupings are shown on the toolbars (*e.g.* Points, Lines).

For details on each of the walk rights menus on the CAD menu, go to the section 15.2 CAD Options

For details on the Controlbars go to the section 15.1 Controlbars

15.1 Controlbars

This default position of the *ControlBars* is on the top left hand side of the screen under the main menu.



See

15.1.1 CAD Controlbar

15.1.2 Symbol Controlbar

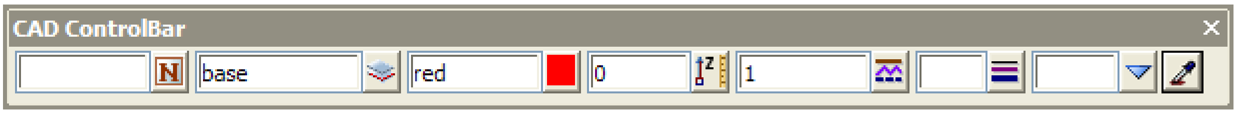
15.1.3 Text Controlbar

15.1.4 Pipe Controlbar

15.1.5 Attributes Controlbar

15.1.6 Search Bar

15.1.1 CAD Controlbar



The fields and buttons used in CAD controlbar have the following functions.

Field Description | Type | Defaults | Pop-Up

name box | | names.4d names

name of string. If a valid name already exists in names.4d, the + button can be used to bring up a choice box of available names. On selection of a valid name, the rest of the values in the control bar will be filled out. e.g. colour, linetype etc.

model box | cogo | existing models

this field can be recognised by the model icon button on the right hand side of the field. The user can select an existing model by selecting the model icon. If a new model is to be used, the user simply types the model name into the field.

colour box | red | standard 12d colours

this field can be recognised by the colour icon button on the right hand side of the field. The user can select a12d standard colour model by selecting the colour icon

input | | height measures menu

this field allows a height or z value to be assigned to the created elements. If a valid value exists, this value will be applied to the created element. This is regardless if the z value was specified in an XYZ box. If blank, the null value is used.
If no value is specified, the level will be interpolated where possible. A value of null can be entered into the height field as well so that created vertices will be given a null height value.

linetype box | 1 | valid linestyles

this field can be recognised by the linestyle icon button on the right hand side of the field. The user can select a valid linestyle by selecting the linestyle icon.

weight box

this field can be recognised by the weight icon button on the right hand side of the field. The user can type in the required weight (millimetres on the plot). If blank, no weight is assigned.

tinability box | | no, yes, points

the choices in this field set the tinability for the string.
*If **no**, no vertices or segments in the string are tinable.*
*If **yes**, all vertices and segments in the string are tinable.*
*If **point**, all vertices in the string are tinable but not the segments.*

button

the eye dropper allows the user to select an existing element which will define the cad control bar values.

Continue to the next section 15.1.2 Symbol Controlbar or return to 15.1 Controlbars.

15.1.2 Symbol Controlbar

The fields and buttons used in Symbol controlbar have the following functions.

Field Description | Type | Defaults | Pop-Up

The  symbols data box allows a symbols favourite to be selected and fill the other boxes. The symbols data box will be described after the other controls.

 symbols box | valid symbols

this field can be recognised by the symbol icon button on the right hand side of the field. The user can select a valid symbol by selecting the symbol icon.

 input | size measures menu

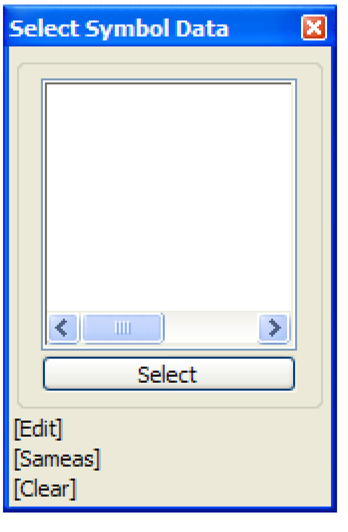
this field gives the size of the created symbols. If a valid value exists, this value will be applied to the created symbol.

 input | angle measures menu

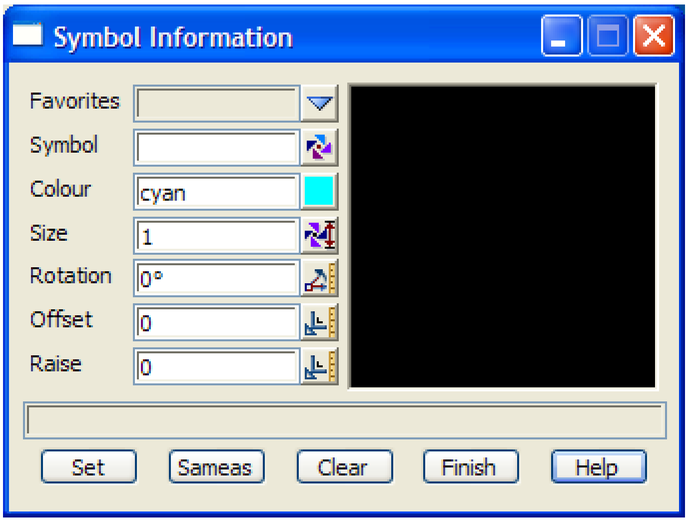
this field gives the angle of the created symbols. If a valid value exists, this value will be applied to the created symbol.

 Symbols data box

on pressing the button a list of available symbols data predefined names read from the ***symbols_names.4d*** *file are displayed. If a symbols data is selected, the values are used for the other Symbol Controlbar fields.*



If no names exist, the user can edit the current settings by selecting the edit button and bring up the **Symbols Information** *panel. This allows for definition of symbol colour, sized, rotation, offset and raise.*



Continue to the next section 15.1.3 Text Controlbar or return to 15.1 Controlbars.

15.1.3 Text Controlbar

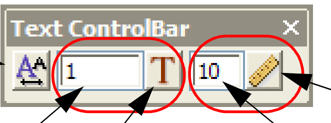
Text can occur as a text string, or on vertices and segments of a super string.

Each type of text has a vertex (these are displayed when Vertices are toggle on in a plan view), a justification point, a rotation, an offset and raise value.

The vertex and justification point only coincide if the offset and raise values are both zero. What parts of the text on a super string vertex or segment that can be independently modified depends on the settings for the super string.

For the CAD text options, the created elements will have attributes as defined by the **Text Controlbar**. The default position of the Text Controlbar is in top middle of the screen under the main menu.

The **Text Controlbar** is



select textstyle data

a textstyle favourite can be selected which then defines the text style and height values.

textstyle select textstyle

the user can select an existing textstyle by selecting the textstyle icon or entering a value into the input box to the left of the button.

text height

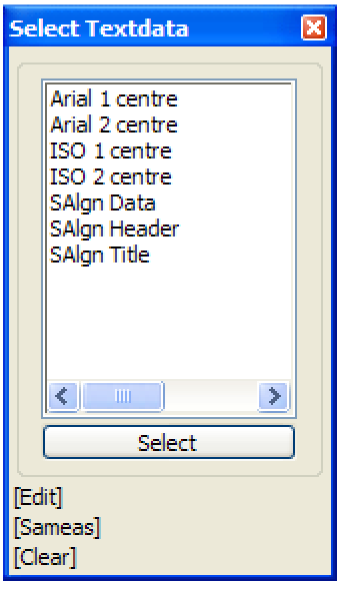
measure for height

the user can measure a height by selecting the text height icon or entering a value into the input box to the left of the button. The value units are defaulted to world units. This can be changed in the textstyle data box

The operation of the Textstyle data box is as follows:

Textstyle data box

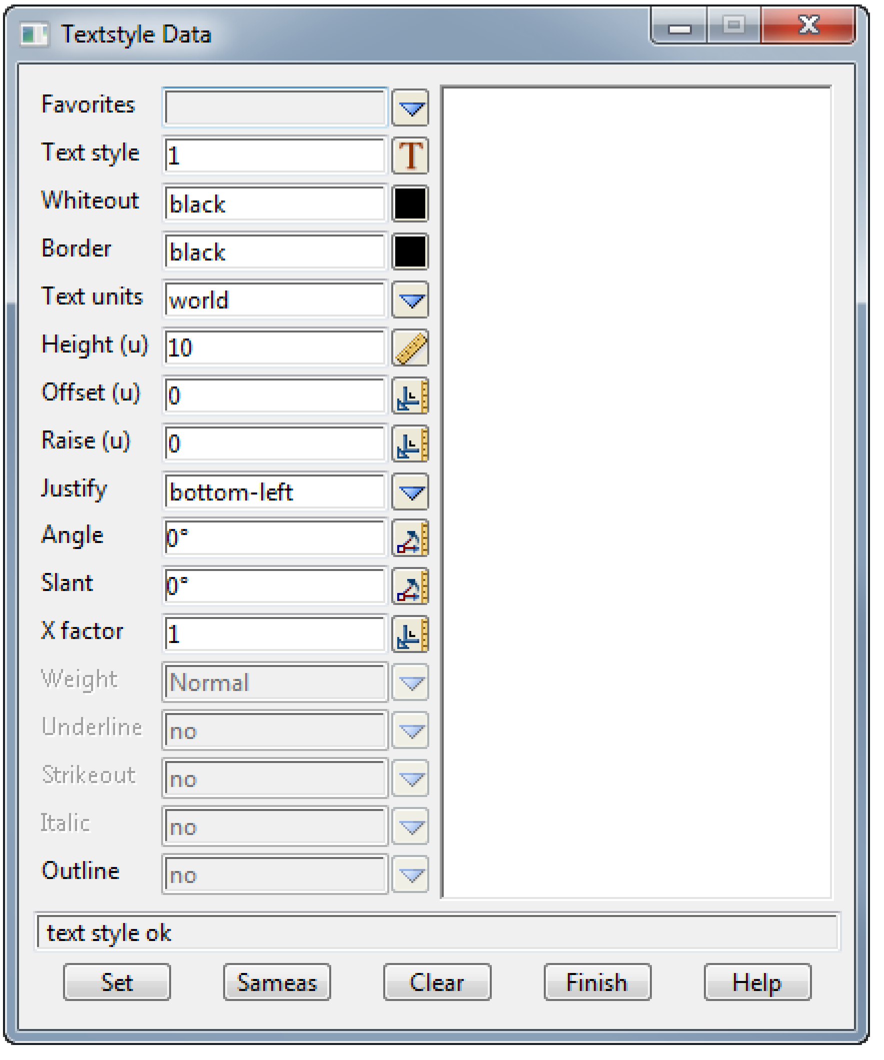
on pressing the button a list of available textdata predefined names read from the ***texstyle_names.4d*** *file are displayed.*



If no names exist, the user can edit the current settings by selecting the edit button and bring up the

Textstyle Data *panel. This allows for definition of textstyle, units, height offset raise etc.*

.



Continue to the next section 15.1.4 Pipe Controlbar or return to 15.1 Controlbars.

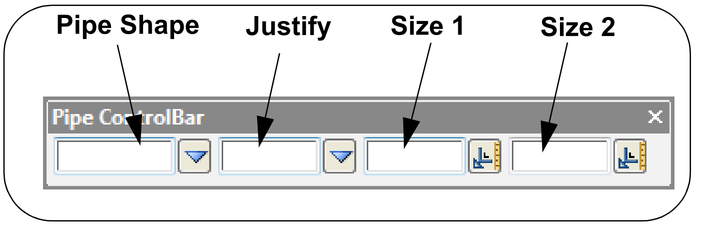
15.1.4 Pipe Controlbar

The Pipe ControlBar is used to set the Pipe dimension when a super string is create with CAD.

The Pipe ControlBar can set the entire super string to be a round pipe or a box culvert pipe.

The **Pipe Controlbar** is

There is a ***Pipe ControlBar*** and it is used to make a round or culvert string whenever a new string is created by the CAD options.



When a new string is created with the CAD options and the value in the **Shape** field of the **Pipe ControlBar** is **Diameter** or **Culvert**, then the **Justify**, **Size 1** and **Size 2** fields from the **Pipe ControlBar** are given to the created string.

To clear **all** the fields in the Pipe ControlBar, clear the **Pipe Shape** field and press <Enter>. This clears all the other fields of the **Pipe ControlBar**.

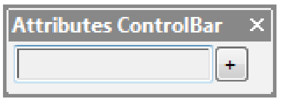
Note

The *Names.4d* now has a **Pipe** section so that the **Pipe ControlBar** can be automatically filled in when a name is selected in the **CAD ControlBar** that is in the **Pipe** section. See 7.9.1.6 Pipe Node in 7.9.1 Name Mappings - Names.4d File.

Continue to the next section 15.1.5 Attributes Controlbar or return to 15.1 Controlbars.

15.1.5 Attributes Controlbar

There is a ***Attributes ControlBar*** and it is used to apply **string** attributes whenever a new super string is created by the CAD options, the attributes in the Attributes ControlBar are given to the string as string attributes.



When a new string is created with the CAD options, the string is given the attributes in the **Attributes ControlBar** as string attributes.

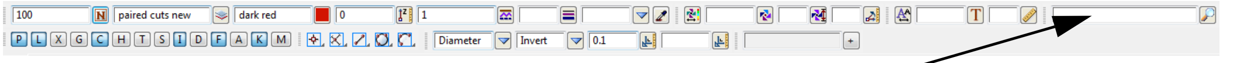
Note

The *Names.4d* now has an **Attribute Data** section so that the ***Attributes*** **ControlBar** can be automatically filled in when a name is selected in the **CAD ControlBar** that is in the ***Attributes*** section. See 7.9.1.7 Attributes Data Node in 7.9.1 Name Mappings - Names.4d File.

Continue to the next section 15.1.6 Search Bar or return to 15.1 Controlbars.

15.1.6 Search Bar

The **Search Bar** allows quick access to any option in ***12d Model***.

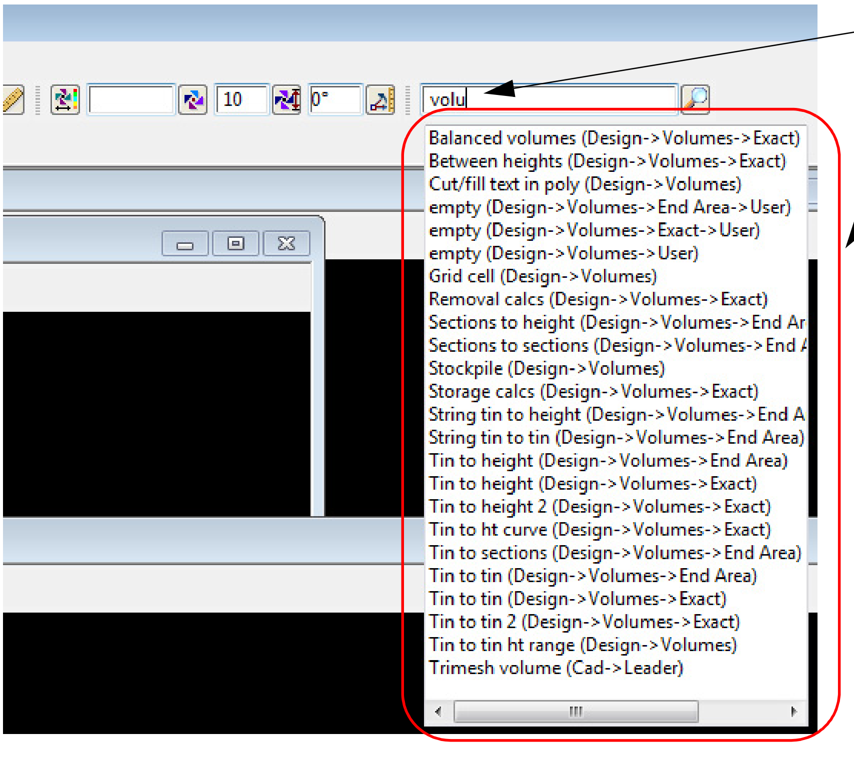


Search bar

By simply typing text into the **Search Bar,** the option searches for matches of the typed text amongst the **full path names** of all the **options on the menus**, and then lists the menu items and the position of the menu that contains the menu item.

For example, typing in **volu** will bring up the list shown below.

Typing **volu** in the *Search Bar*



List of matches for **volu** in the full path to the name of the option on a menu, and the position in brackets of the menu that contains the text **volu**.

Clicking on any item in the list brings up the panel for the option

Double clicking on an item in the list brings up the panel for that item.

Note that case is ignored when searching for matches.

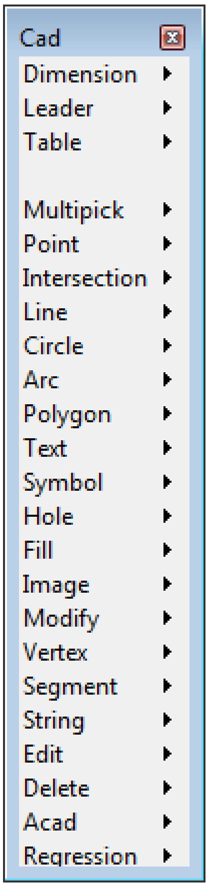
Return to 15.1 Controlbars.

15.2 CAD Options

Position of option on menu: **CAD**

The **CAD** options are available from the **CAD** menu or from the **CAD toolbar**. The **CAD toolbar** is a flyout toolbar.

The **CAD** walk-right menu is



and the **CAD Toolbar** is

The string creation process is similar for each string type and for editing strings as well. On selecting the appropriate option, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

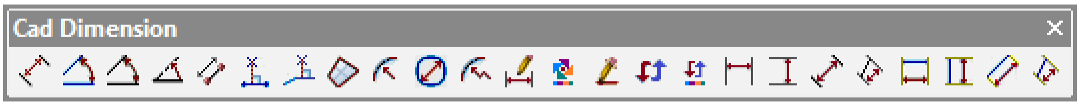
For the options see

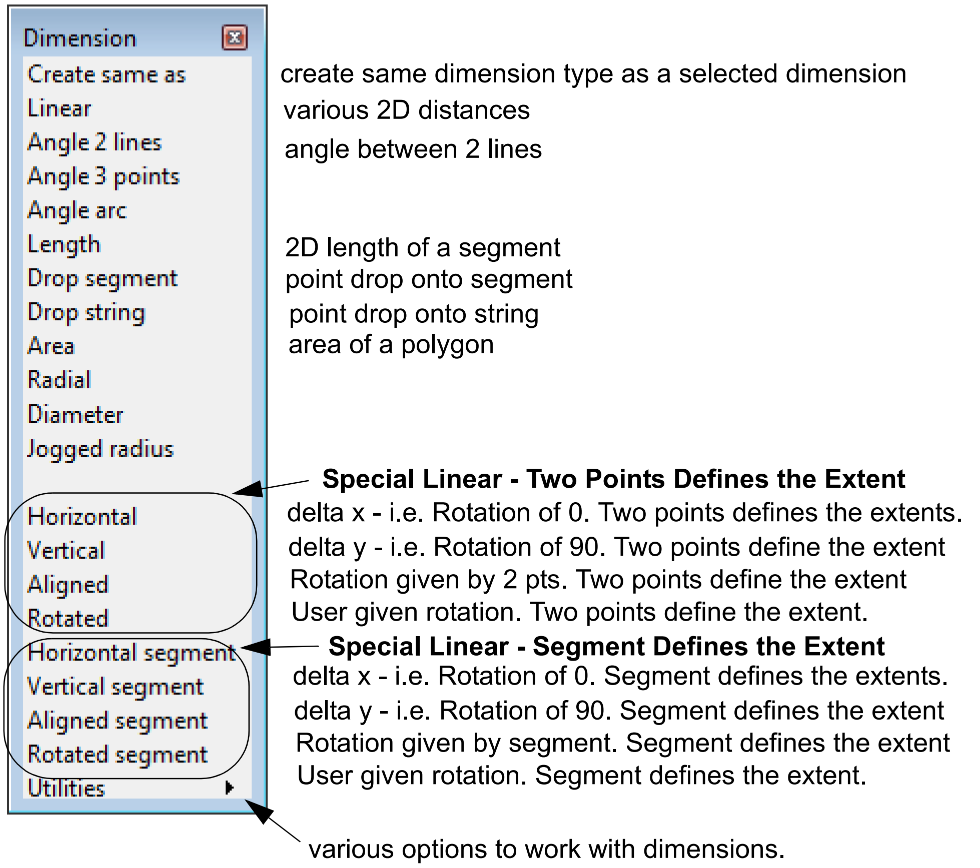
Dimension	15.3 CAD Dimension
Leader, go to	15.4 CAD Leader
Table, go to	15.6 CAD Table
Multipick, go to	15.8 CAD Multipick
Point	15.9 CAD Point
Intersection	15.10 CAD Intersection
Line	15.11 CAD Line

15.3 CAD Dimension

Position of option on menu: **CAD =>Dimension**

Various types of **Dimensions** can be created with the **Dimension** options.





Each type of **Dimension** remembers the items it was created from (Association). In most cases when the items are modified, the **Dimensions** and the associated values on the Dimension dynamically change. See 15.3.1 Dimension Association.

For information about the meaning of the linear dimension commands Linear, Horizontal, Vertical, Aligned, Rotated, Horizontal segment, Vertical segment, Aligned segment and Rotated segment see

For information about the meaning of the linear dimension commands Linear, Horizontal, Vertical, Aligned, Rotated, Horizontal segment, Vertical segment, Aligned segment and Rotated segment see 15.3.6 Linear Dimension Commands.

For more information on each of the dimension commands, see

Create same as - create new dimension of same type as selected dimension15.3.5 Dimension - Create Same As

Linear - various 2D distances. *See* 15.3.6.1 Linear

Angle 2 lines - angle between two selected lines. *See* 15.3.14 Angle 2 Lines

Angle 3 points - angle between two lines defined by 3 points.*See* 15.3.15 Angle 3 points

Angle Arc - angle subtended by existing arc or a segment with a radius. *See* 15.3.16 Angle Arc

Length - the 2D length of selected segments *See* 15.3.7 Length

Drop segment - point dropped onto segment. *See* 15.3.8 Drop Segment

Drop string - point dropped onto a string. *See* 15.3.8 Drop Segment

Area - area of selected strings. *See* 15.3.10 Area

Radius - radius of a selected arc. *See* 15.3.13 Radius

Diameter - diameter of an arc 15.3.12 Diameter.

Jogged radius - radius of a selected arc with jog. *See* 15.3.11 Jogged radius

Arc Length - Arc by Centre, 2 Points - length of an arc with arc defined by selecting a centre and two points on the arc. *See* 15.3.17 Arc Length by Centre, 2 Points

Horizontal - the horizontal distance (delta x) between two selected points. *See* 15.3.6.3 Horizontal

Vertical - the vertical distance (delta y) between two selected points. *See* 15.3.6.4 Vertical

Aligned - 2D distance between two selected points. *See* 15.3.6.2 Aligned

Rotated - the 2D between a selected point and another selected point dropped onto the line going through the first point with a user given angle. *See* 15.3.6.5 Rotated

Horizontal segment - the horizontal distance (delta x) between the end vertices of selected segments. *See* 15.3.6.7 Horizontal Segment

Vertical segment - the vertical distance (delta y) between the vertical points of the selected segment. *See* 15.3.6.8 Vertical Segment

Aligned segment - the 2D distance between the end vertices of selected segments. *See* 15.3.6.6 Aligned Segment

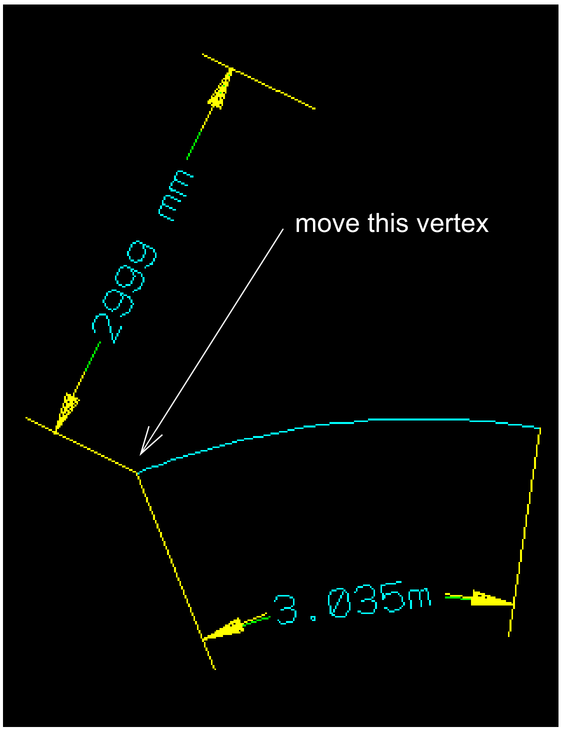
Rotated segment -the 2D between the first vertex of a selected segment and the end vertex of the segment dropped onto the line going through the first vertex with a user given angle. *See* 15.3.6.9 Rotated Segment

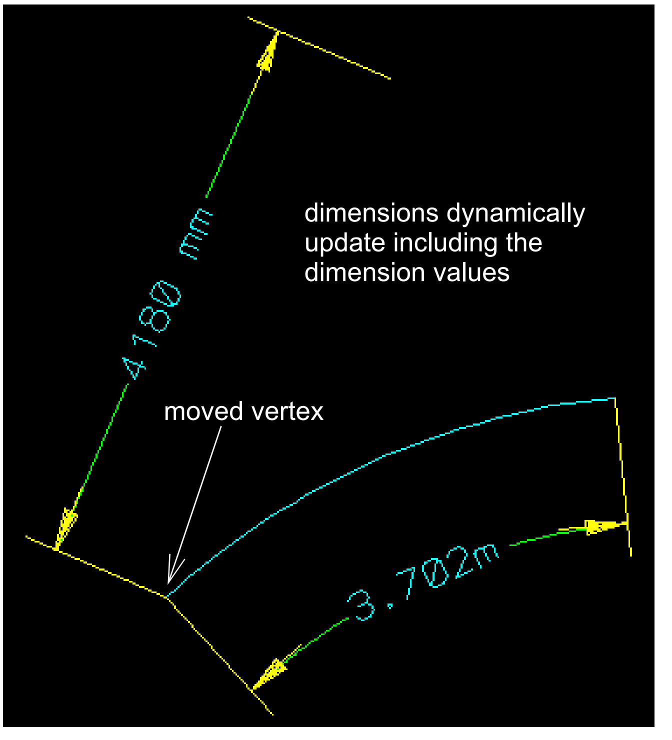
Utilities - various dimension utilities. *See* 15.3.18 Dimension Utilities

15.3.1 Dimension Association

When *dimensions* are created, the dimension can remember the items that were used to create them (**Association**).

In most cases when items are moved, the dimensions associated with the items dynamically change.



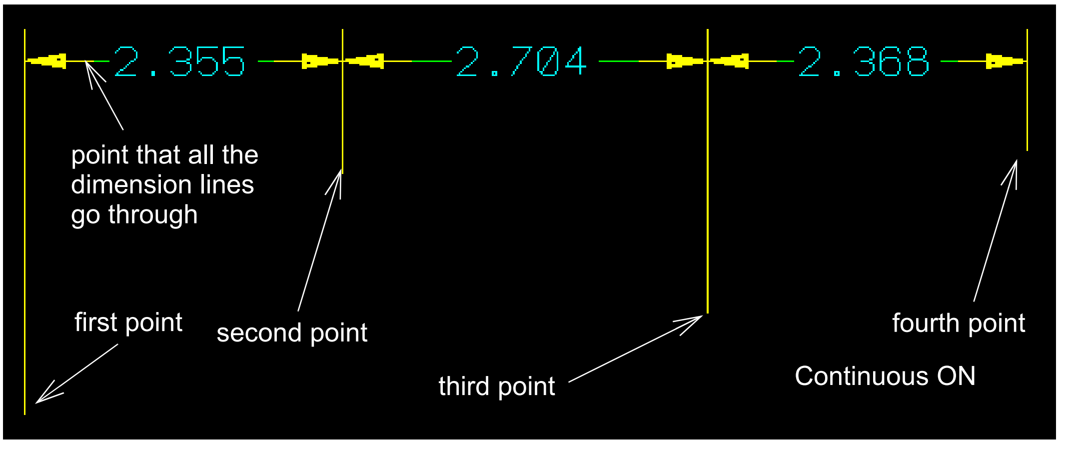


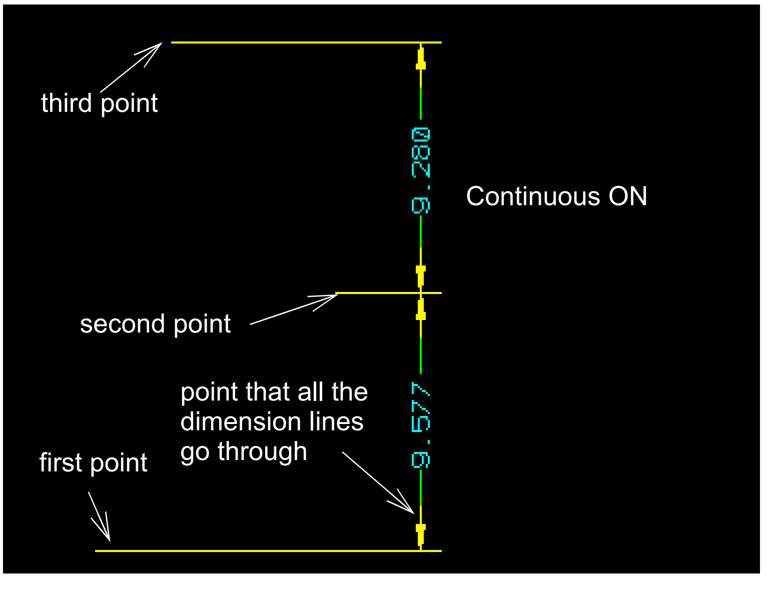
In cases where it is not automatic, there is a **Recalc** and **Recalc all** for dimensions and leaders. In the dimension options, typing **a** or **A** toggles Association **ON** and **OFF**.

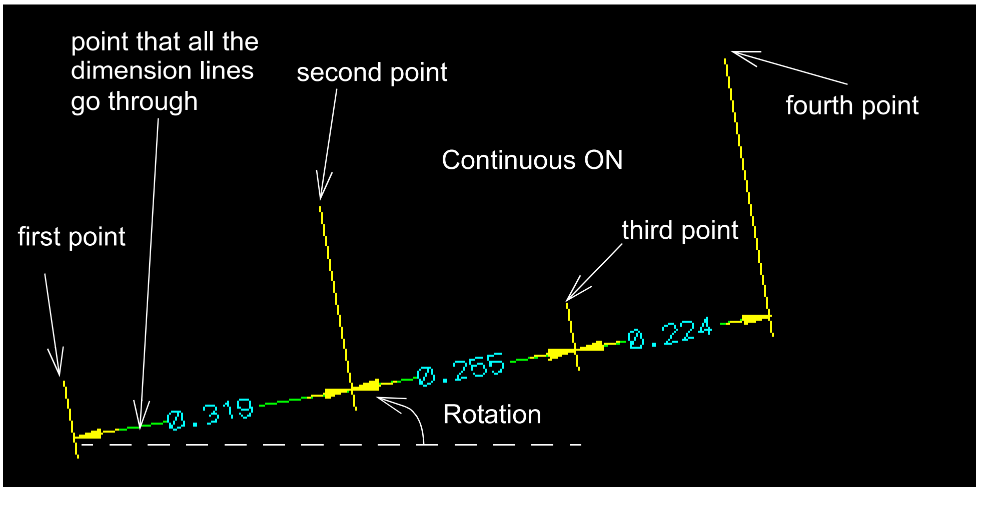
15.3.2 Continuous Mode

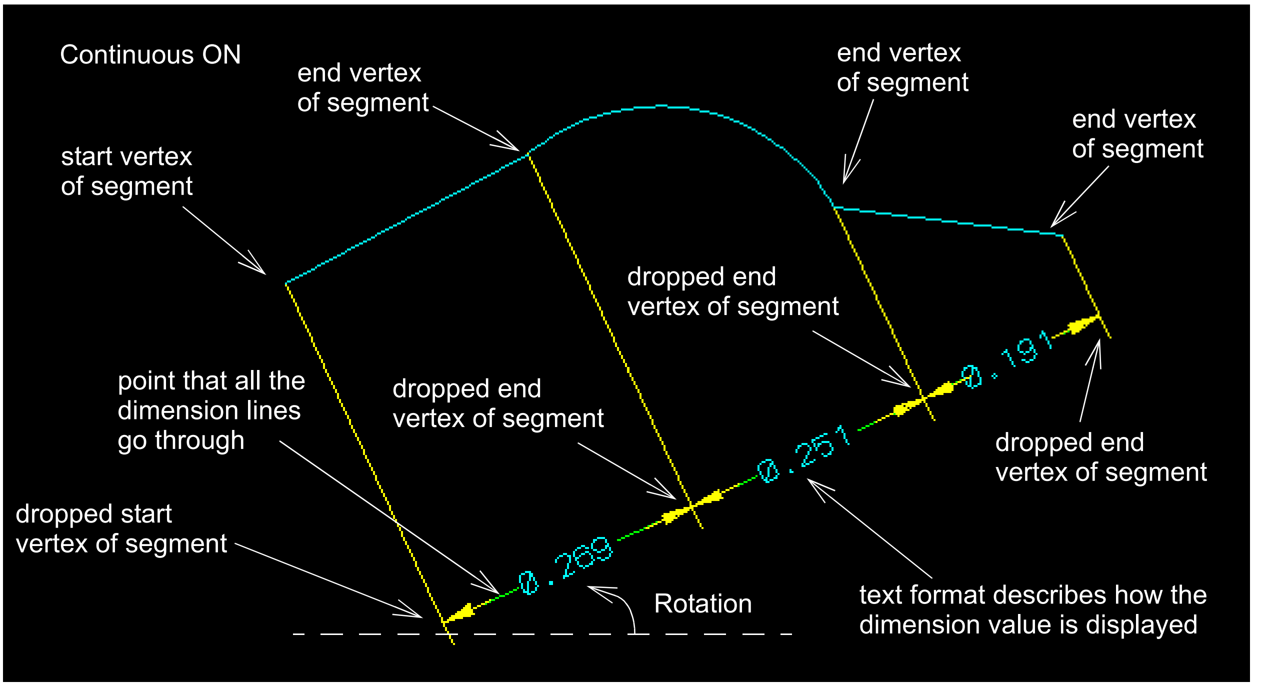
For all the **Linear** dimension options except **Aligned**, there is a **Continuous** mode.

When Continues mode is **on**, all the dimensions lines are on the same line as the dimension line of the first dimension done in the sequence.









In the dimension options which support Continuous, typing **c** or **C** toggles Continuous **ON** and **OFF**.

15.3.3 Baseline Mode

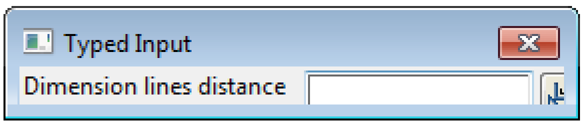
For all the dimension options *Horizontal*, *Vertical* and *Rotated*, there is a **Baseline** mode.

When **Baseline** mode is **ON**, the distances are all measured from the first picked point, the first dimension line goes through the picked dimension point, and for each subsequent dimension line, the distance is measured from the first point, and the dimension line is moved a give distance offset to the previous dimension line so that the dimension lines for not overlap.

In the dimension options which support *Baseline*, **(b)aseline** appears in the message in the screen message area:

<[(a)ssociation ON] [(c)ontinuous OFF] [(b)aseline OFF] Pick start point or choose (s)tyle[d1] > [picks][accepts][Menu]

Typing **b** or **B** toggles **Baseline ON** and **OFF** and when **Baseline** is toggle **ON** for the first time, the *Dimension lines distance* **Typed Input** box is place on the screen.



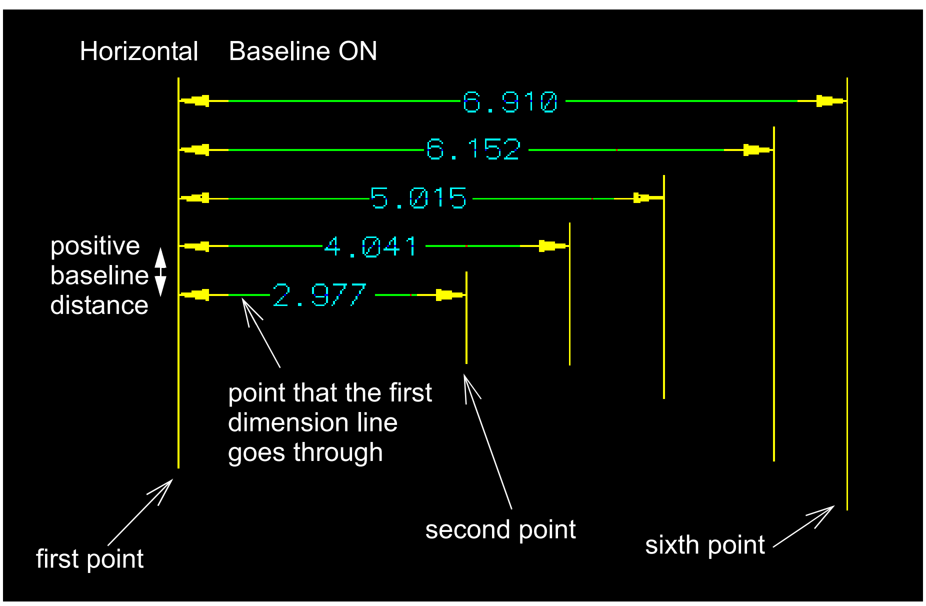
The new distance is typed into the box and <Enter> pressed.

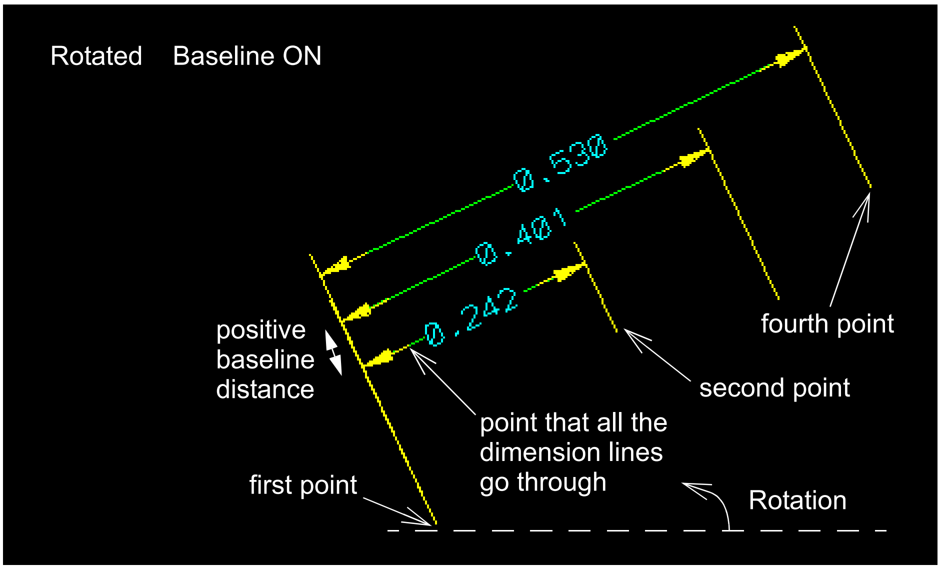
Whenever *Baseline* is toggled **ON**, (**d)istance** appears in the message in the screen message area:

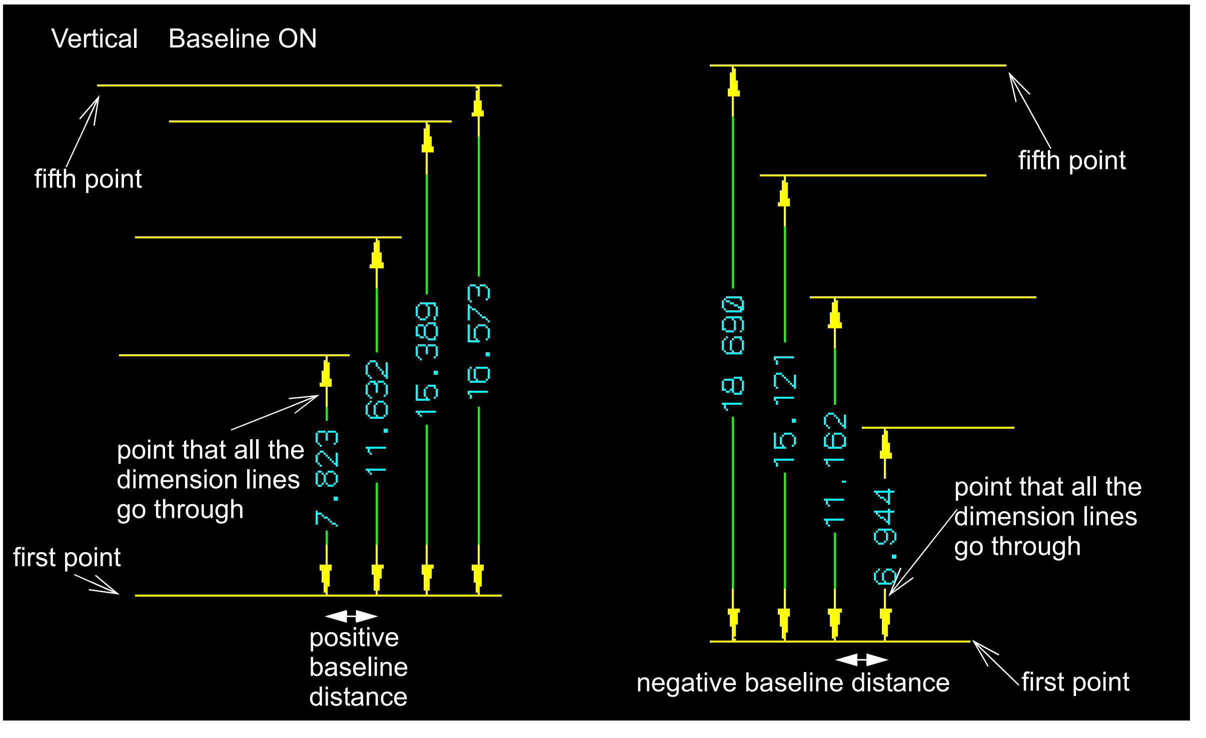
<[(a)ssociation ON] [(c)ontinuous OFF] [(b)aseline ON (d)istance[0.500]]Pick start point or choose (s)tyle[d1] > [picks]

Typing **d** or **D** toggles brings up the *Dimension lines distance* **Typed Input** box and a new distance can be entered.

Whenever **Baseline** is **ON**, **Continuous** is turned **OFF**.




Rotated Baseline ON
0.530
0.401
0.242
positive
baseline
distance
fourth point
second point
point that all the
dimension lines
go through
first point
Rotation


Vertical Baseline ON
fifth point
point that all the
dimension lines
go through
first point
7.823
11.632
15.389
16.573
positive
baseline
distance
18.690
15.121
11.162
6.944
fifth point
point that all the
dimension lines
go through
negative baseline distance
first point

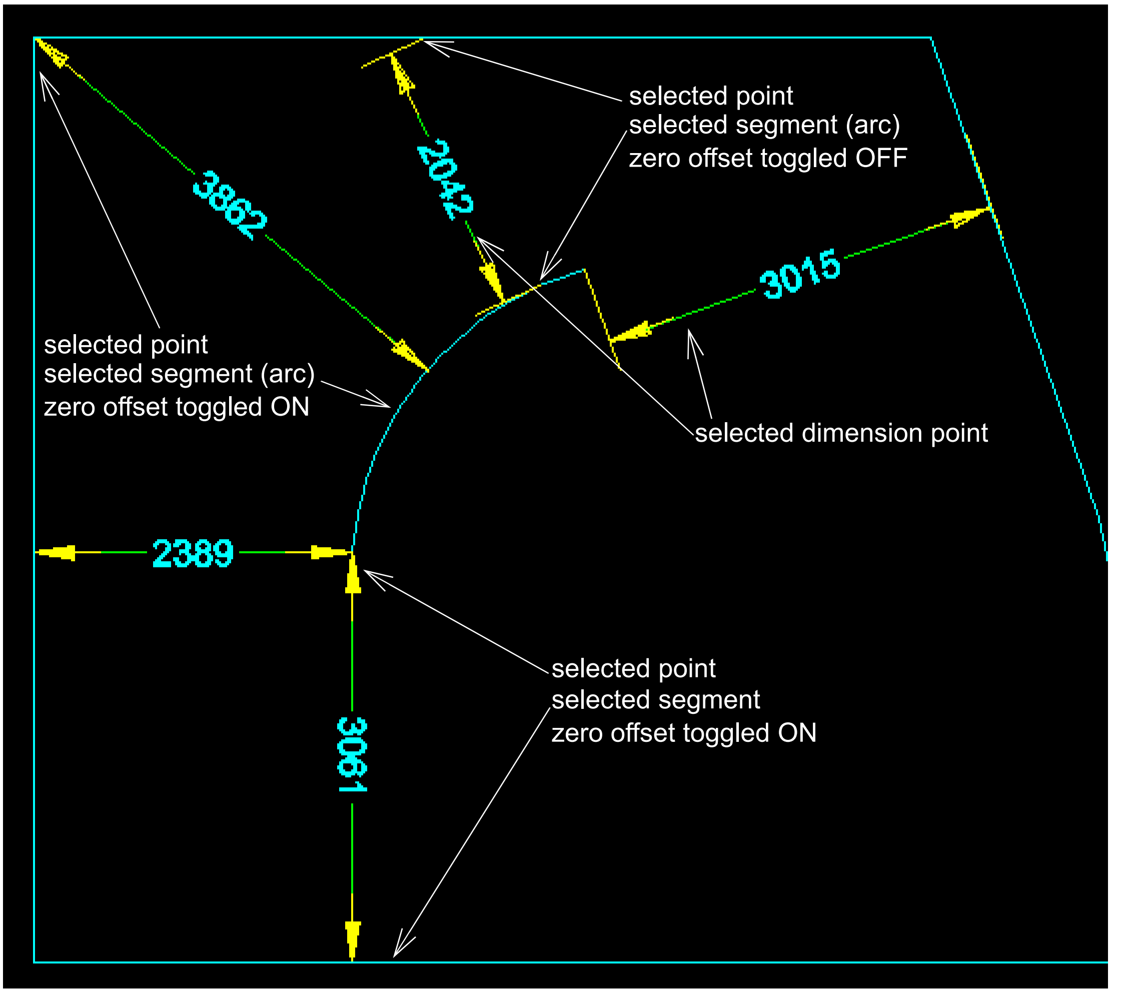
15.3.4 Zero Offset Mode

For some dimension options there is a **Zero Offset** mode.

When **Zero Offset** mode is **ON**, then the user is not asked to select a dimension point.

The dimension line is automatically placed on the line between the two points

When **Zero Offset** is available, typing **z** or **Z** toggles **Zero offset ON** and **OFF**.



15.3.5 Dimension - Create Same As

Create same as creates a new Dimension of the same type as a selected Dimension.

Step 1.

Select **Create same as** and the following message is written to the screen message area

<[Pick dimension]> [picks][fast][Menu]

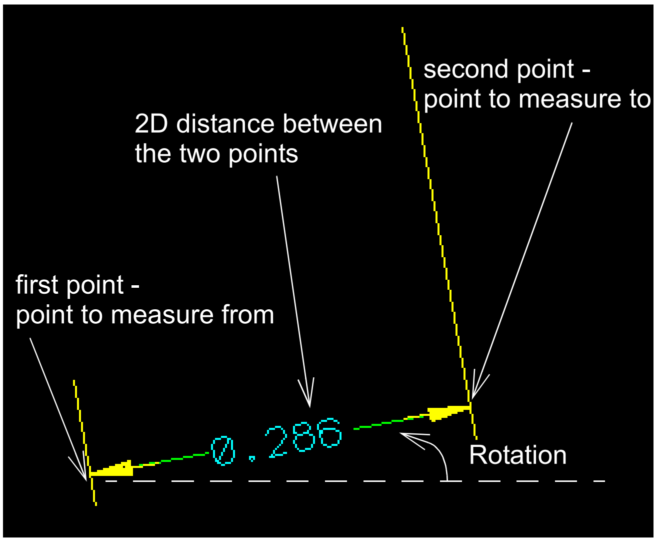
Step 2.Select an existing Dimension

When an existing **Dimension** is selected, a new **create Dimension** of the same Dimension type as the selected Dimension, is started.

15.3.6 Linear Dimension Commands

The **Linear Dimension** commands measure the **2D** (plan) **distance between two points**.

The two points to measure between define a line and that line has a **Rotation** angle with respect to the x-axis (Rotation is measured in an anticlockwise direction from the positive x-axis).



Note that the two points may be selected individually or can be the end points of a selected segment (if the segment is an arc or a transition, only the end points of the segment are used).

Because most drawings to be dimensioned are very regular and often drawn so that the major line work is parallel to the x and y axes, it is an advantage to have special forms of the **Linear Dimension** so that the intent is clearer and less information has to be provided by the user.

The **special Linear Dimensions** are best classified by their **Rotation** angle.

(a) **Horizontal** - **rotation angle of 0**

This is the difference between the two x coordinates of the points or segment (delta x).

The Rotation is known (0) so once the first point is selected, the software automatically drops the second point onto the line with Rotation 0 that goes through the first point and uses the x value from the dropped point.

Horizontal and selecting **two points**. *See* 15.3.6.3 Horizontal
Horizontal and selecting a **segment**. *See* 15.3.6.7 Horizontal Segment

(b) **Vertical** - **rotation angle of 90**

This is the difference between the two y coordinates (delta y).

The Rotation is known (90) so once the first point is selected, the software automatically drops the second point onto the line with Rotation 90 that goes through the first point and uses the y value from the dropped point.

Vertical and selecting **two points**. *See* 15.3.6.4 Vertical
Vertical and selecting a **segment** *See* 15.3.6.8 Vertical Segment

(c) **Rotated - a user given rotation value**.

The Rotation is known so once the first point is selected, the software automatically drops the second point onto the line with the given Rotation that goes through the first point and uses the dropped point to measure the 2D distance to.

Rotated and selecting **two points**. *See* 15.3.6.5 Rotated
Rotated and selecting a **segment** *See* 15.3.6.9 Rotated Segment

(d) **Aligned** - **rotation defined by selecting two points or a segment**

This is a more general case where the two points have to accurately selected and are used to defined the Rotation. This is the only case when the Rotation is not known before selecting anything.

Aligned and selecting two points. *See* 15.3.6.2 Aligned

Aligned and selecting a **segment** *See* 15.3.6.6 Aligned Segment

Hence in ***12d Model*** there are **eight separate commands** for defining a Linear dimension with a special rotation, and **one** general **Linear** command that covers all **eight Linear command**.

See

Linear - all eight cases. *See* 15.3.6.1 Linear

15.3.6.1 Linear

The **Linear** option allows the use any of the eight modes of defining a Linear dimension.

The dimension is added to the model given in the CAD toolbar.

a or **A** toggles Association **ON** and **OFF**. See 15.3.1 Dimension Association
c or **C** toggles Continuous **ON** and **OFF**. See 15.3.2 Continuous Mode
t or **T** brings up the dimension text format. See 15.5 Text Format for Dimensions and Leaders
s or **S** brings up the dimension style list. 15.3.18.2 Dimension Styles

Step 1.

Select **Linear** and the following message is written to the screen message area

<[Pick start poi(n)t] [(h)orizontal (v)(l)(r)] [(c)ontinuous OFF] [(b)aseline OFF] (a)ssociative (s)tyle[Arial Aligned] forma(t) > [picks][fast][Menu]

Step 2.Settings

Typed one characters commands are now required and the first ones to set are those that don't actually start a dimension but are set ups - **n**, **a**, **c**, **s**.

Some of the commands, **n**, **a**, **c**, **s**

n or **N** toggles between picking two poi**n**ts or picking a segme**n**t.

<[Pick start poi(n)t] [(h)orizontal (v)(l)(r)] [(c)ontinuous OFF ... (s)tyle[Arial Aligned] forma(t) > [picks][fast][Menu]
<[Pick segme(n)t] [(h)orizontal (v)(l)(r)] [(c)ontinuous O... Aligned] forma(t) > [picks][fast][Menu]

clicking **n** toggles between poi**n**t and segme**n**t

a or **A** toggles association **ON** and **OFF**.
c or **C** toggles continuous **ON** and **OFF**.
s or **S** brings up the dimension style list.

Step 3.Dimension Commands

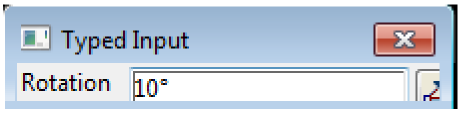
The dimension commands are **h**orizontal, **v**ertical, **a**ligned or **r**otated and the current **active** dimension command has square brackets around it.

h makes **h**orizontal the active dimension command.
v makes **v**ertical the active dimension command.
l makes a**l**igned the active dimension command.
r makes **r**otated the active dimension command.

Note - When **rotated** is the active dimension command, the current value for the rotation angle is displayed in the message area.

[(h)(v)(l) (r)otated] an(g)le[0°] rotation angle

Whenever the angle is displayed, typing **g** or **G** will bring up a **Typed Input** box to change the rotation angle.



Step 4.Creating a Dimension - picking first point or a segment.

What type of linear dimension is to be created has been set (h, v, l, r).
Whether it requires two points or a segment **has** been set.

Whether it will be **a**ssociated or not, or **c**ontinuous or not **has** been set.

Depending on the point/segment mode, the start of the message indicates whether picking a point or picking a segment is required. So now pick the first point or a segment.

How the **Linear** option operates is now identical to one of eight specialised linear dimension options so it will not be documented now but done in each of the specialised cases.

If **Horizontal** and **point**, see 15.3.6.3 Horizontal
If **Horizontal** and **segment**, see 15.3.6.7 Horizontal Segment

If **Vertical** and **point**, see 15.3.6.4 Vertical
If **Vertical** and **segment**, see 15.3.6.8 Vertical Segment

If **Aligned** and **point**, see 15.3.6.2 Aligned
If **Aligned** and **segment**, see 15.3.6.6 Aligned Segment

With Rotated, the angle is already set so you won't be asked for that again.
If **Rotated** and **point**, see 15.3.6.5 Rotated
If **Rotated** and **segment**, see 15.3.6.9 Rotated Segment

Step 5.

After the dimension has been created, the **Linear** option then starts again. That is, goes back to **Step 1**. You can continue with the current dimension type or switch to another dimension type by clicking on **h**, **v**, **l** or **r** (or **H**, **V**, **L** or **R**).

For information on how the dimension object appears, see 15.3.18.2 Dimension Styles.

15.3.6.2 Aligned

Aligned labels the 2D distance between two selected points.

The dimension is added to the model given in the CAD toolbar.

a or **A** toggles Association **ON** and **OFF**. See 15.3.1 Dimension Association
t or **T** brings up the dimension text format. See 15.5 Text Format for Dimensions and Leaders
s or **S** brings up the dimension style list. 15.3.18.2 Dimension Styles
z or **Z** toggles *Zero Offset* mode **ON** and **OFF**. See 15.3.4 Zero Offset Mode

Step 1. Pick the start point

Select **Aligned** and the following message is written to the screen message area

```
<[Pick start point] (a)ssociative (s)tyle[default] forma(t) [(z)ero offset OFF] > [picks][fast][Menu]
```

If Association is **ON** then the start of the dimension is associated with the selected start point.

Step 2.Pick the end point

The following message is written to the screen message area.

```
<[Pick end point] (a)ssociative > [picks][fast][Menu]
```

If Association is **ON** then the end of dimension is associated with the selected end point.

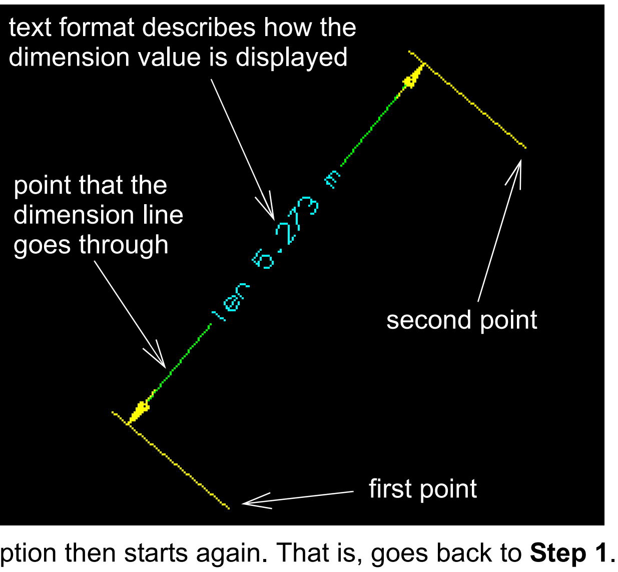
The **2D distance** is calculated between these two points.

Step 3.Pick the point that the dimension line will go through (dimension point)

The following message is written to the screen message area.

```
<Pick dimension point> [picks][fast][Menu]
```

The dimension line is then drawn parallel to the line between the two selected points and going through the dimension point.



The **Aligned** option then starts again. That is, goes back to **Step 1**.

For information on how the dimension object appears, see 15.3.18.2 Dimension Styles.

15.3.6.3 Horizontal

Horizontal labels the horizontal distance (delta x) between two selected points.

The dimension is added to the model given in the CAD toolbar.

a or **A** toggles Association **ON** and **OFF**. See 15.3.1 Dimension Association
c or **C** toggles Continuous **ON** and **OFF**. See 15.3.2 Continuous Mode
t or **T** brings up the dimension text format. See 15.5 Text Format for Dimensions and Leaders
s or **S** brings up the dimension style list. 15.3.18.2 Dimension Styles

Step 1.Pick the start point

Select **Horizontal** and the following message is written to the screen message area

```
<[Pick start point] [(c)ontinuous OFF] [(b)aseline OFF] (a)ssociative (s)tyle[default] forma(t) > [picks][fast][Menu]
```

If Association is **ON** then the start of the dimension is associated with the selected first point.

Step 2.Pick the end point

The following message is written to the screen message area.

```
<[Pick end point] (a)ssociative > [picks][fast][Menu]
```

If Association is **ON** then the end of dimension is associated with the selected end point.

The horizontal distance (delta x) is calculated between these two points.

Step 3.Pick the point that the dimension line will go through (dimension point)

The following message is then written to the left hand side of the screen message area.

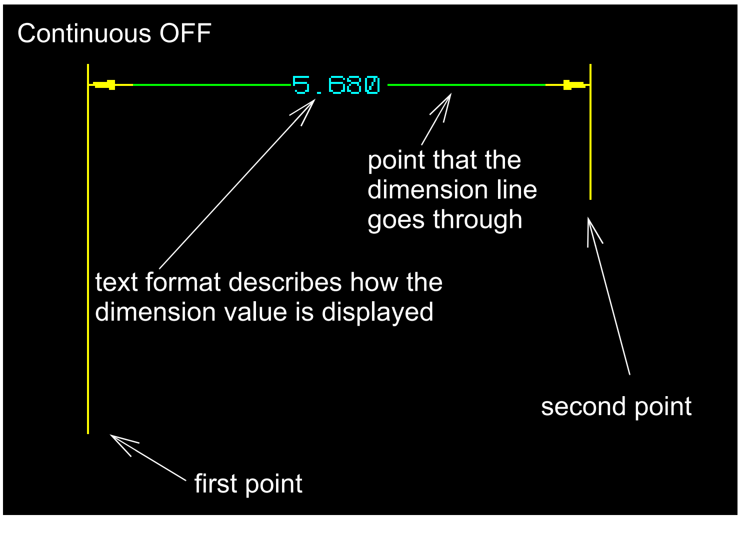
```
<Pick dimension point> [picks][fast][Menu]
```

The horizontal dimension line is then drawn parallel to the x-axis going through the dimension point.

Step 4.I

If Continuous is **OFF**

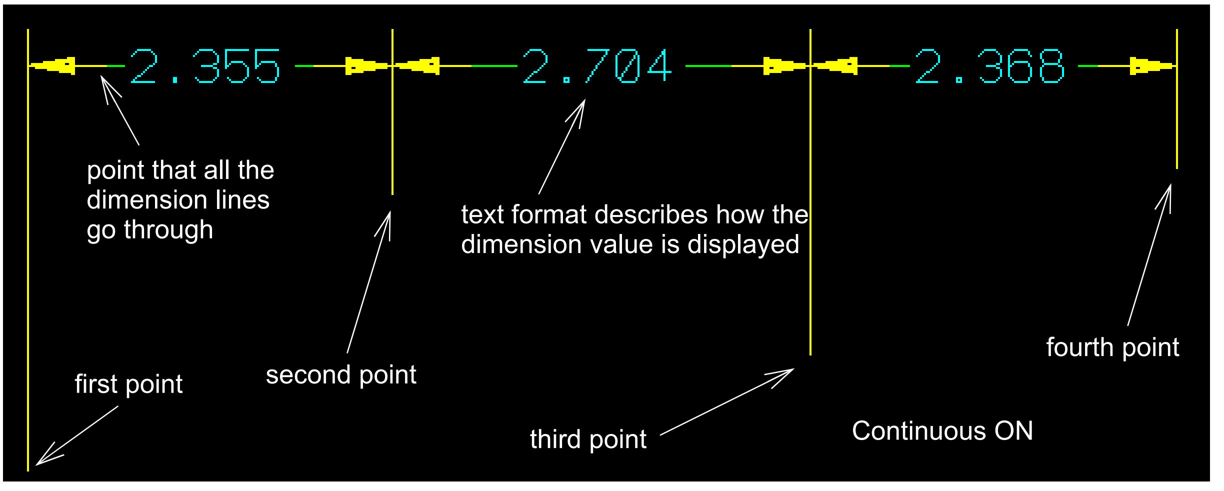
The **Horizontal** option then starts again. That is, goes back to **Step 1**.



If Continuous is **ON**

If Continuous is **ON** then you are asked to pick another end point and another Horizontal dimension is created with the start of the horizontal dimension being the previous dimension end, and the dimension point is the same as the previous dimension point.

This repeats until Continuous is toggle **OFF** and then the next selected end point is the last horizontal dimension in the sequence.



For information on how the dimension object appears, see 15.3.18.2 Dimension Styles.

15.3.6.4 Vertical

Vertical labels the vertical distance (delta y) between two selected points.

The dimension is added to the model given in the CAD toolbar.

a or **A** toggles Association **ON** and **OFF**. See 15.3.1 Dimension Association
c or **C** toggles Continuous **ON** and **OFF**. See 15.3.2 Continuous Mode
t or **T** brings up the dimension text format. See 15.5 Text Format for Dimensions and Leaders
s or **S** brings up the dimension style list. 15.3.18.2 Dimension Styles

Step 1.Pick the start point

Select **Vertical** and the following message is written to the screen message area

<[Pick start point] [(c)ontinuous OFF] [(b)aseline OFF] (a)ssociative (s)tyle[default] forma(t) > [picks][fast][Menu]

If Association is **ON** then the start of the dimension is associated with the selected first point.

Step 2.Pick the end point

The following message is written to the screen message area.

<[Pick end point] (a)ssociative > [picks][fast][Menu]

If Association is **ON** then the end of dimension is associated with the selected end point.

The vertical length (delta y) is calculated between these two points.

Step 3.Pick the point that the dimension line will go through (dimension point)

The following message is then written to the left hand side of the screen message area.

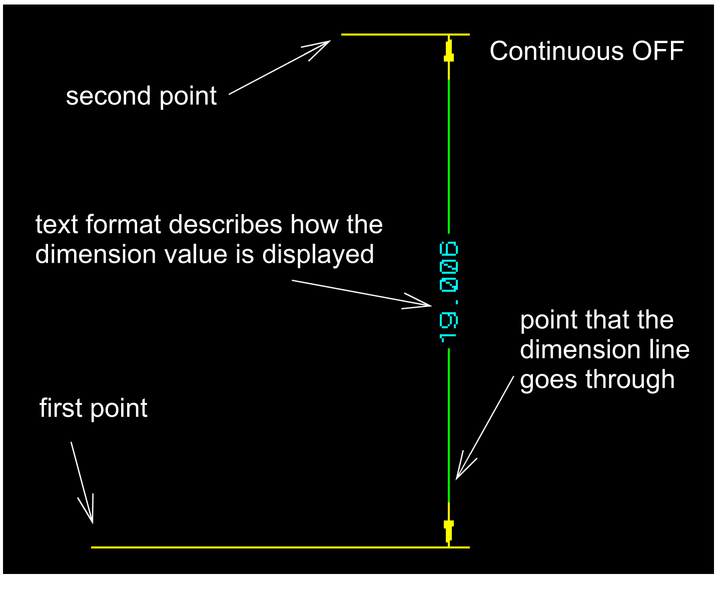
<Pick dimension point> [picks][fast][Menu]

The vertical dimension line is then drawn parallel to the y-axis going through the dimension point.

Step 4.I

If Continuous is **OFF**

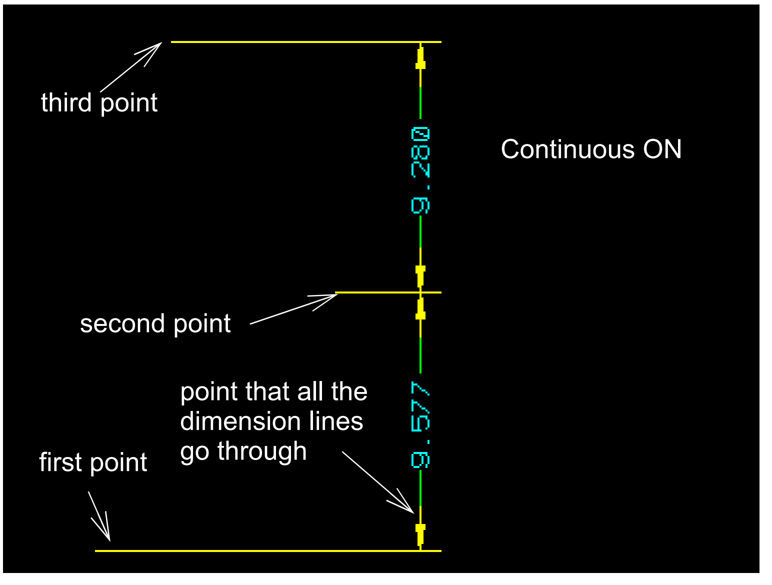
The **Vertical** option then starts again. That is, goes back to **Step 1**.



If Continuous is **ON**

If Continuous is ON then you are asked to pick another end point and another Vertical dimension is created with the start of the vertical dimension being the previous dimension end, and the dimension point is the same as the previous dimension point.

This repeats until Continuous is toggle **OFF** and then the next selected end point is the last vertical dimension in the sequence.



For information on how the dimension object appears, see 15.3.18.2 Dimension Styles.

15.3.6.5 Rotated

Rotated labels the **2D distance** between the first selected point and the second selected point **dropped** onto the line with a user given angle (Rotation).and going through the first selected point.

The dimension is added to the model given in the CAD toolbar.

a or **A** toggles Association **ON** and **OFF**. See 15.3.1 Dimension Association
c or **C** toggles Continuous **ON** and **OFF**. See 15.3.2 Continuous Mode
t or **T** brings up the dimension text format. See 15.5 Text Format for Dimensions and Leaders
s or **S** brings up the dimension style list. 15.3.18.2 Dimension Styles

Step 1.Pick the start point

Select **Rotated** and the following message is written to the screen message area

<[Pick start point] [(c)ontinuous OFF] [(b)aseline OFF] (a)ssociative (s)tyle[default] forma(t) > [picks][fast][Menu]

If Association is **ON** then the start of the dimension is associated with the selected first point.

Step 2.Pick the end point

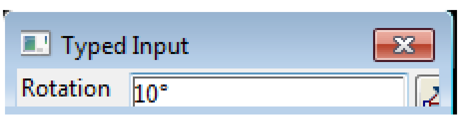
The following message is written to the screen message area.

<[Pick end point] (a)ssociative > [picks][fast][Menu]

If Association is **ON** then the end of dimension is associated with the selected end point.

Step 3.Type in the Rotation value

The following message is written to the screen message area.



The Rotation is in degrees in hp notation and is measured in an anticlockwise direction from the positive x axis.

Type in the Rotation value and <Enter>

When in **Continuous** mode the Rotation is only asked for when placing the first **Rotated** dimensions otherwise it is asked for each time.

Note: for the **Linear** command, the **Rotation** value is not asked for because it will have already been set.

Step 4.Pick the point that the dimension line will go through (dimension point)

The following message is then written to the left hand side of the screen message area.

<Pick dimension point> [picks][fast][Menu]

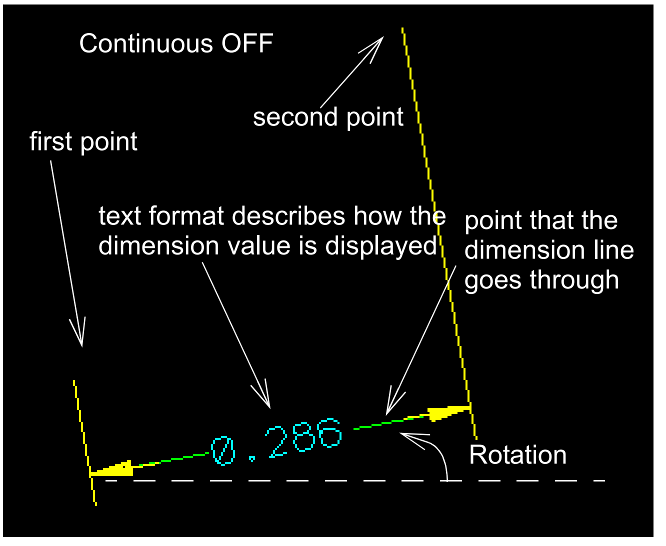
The **2D (plan) distance** is calculated between the first selected point and the second selected point dropped onto the line going through the first selected and with the user given angle (Rotation).

The dimension line is then drawn with the angle of the dimension line equal to the entered Rotation.

Step 5.I

If Continuous is **OFF**

The **Rotated** option then starts again. That is, goes back to **Step 1.**

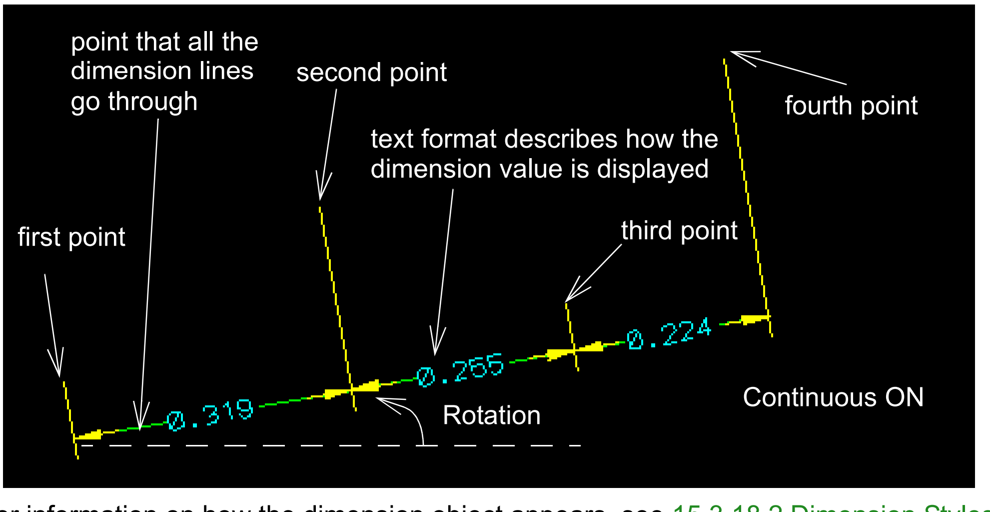


If Continuous is **ON**

If Continuous is ON then you are asked to pick another end point and another Rotated dimension is created with the start of the Rotated dimension being the previous dimension end, and the dimension point is the same as the previous dimension point.

The **2D (plan) distance** is calculated between the previous dropped selected point and the new selected point dropped onto the line with the user given angle (Rotation) going through the first selected point.

This repeats until Continuous is toggle OFF and then the next selected end point is the last Rotated dimension in the sequence.



For information on how the dimension object appears, see 15.3.18.2 Dimension Styles.

15.3.6.6 Aligned Segment

Aligned segment labels the **2D distance** between the start and end vertices of a selected segment. The segment can be straight, arc or a transition.

The dimension is added to the model given in the CAD toolbar.

Note that this is **not** the same as the length of a segment for an arc, a transition or offset transition.

a or **A** toggles Association **ON** and **OFF**. See 15.3.1 Dimension Association
t or **T** brings up the dimension text format. See 15.5 Text Format for Dimensions and Leaders
s or **S** brings up the dimension style list. 15.3.18.2 Dimension Styles

Step 1.Pick the segment

Select **Aligned segment** and the following message is written to the screen message area

<[Pick segment] (a)ssociative (s)tyle[default] forma(t) > [picks][fast][Menu]

If Association is **ON** then the dimension is associated with the selected segment.

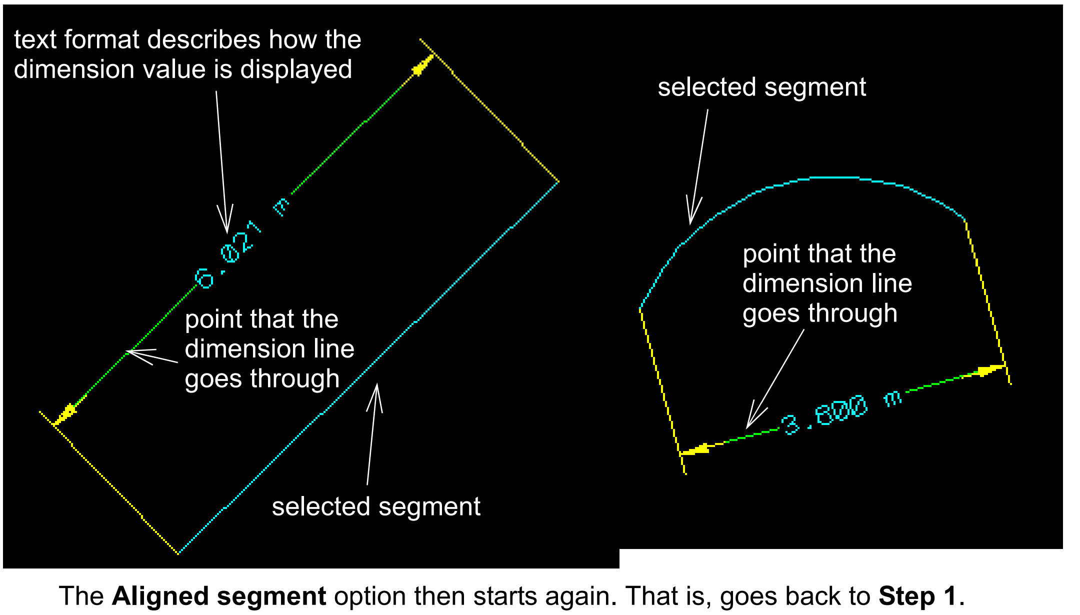
The 2D distance is calculated between the start and end vertices of the selected segment.

Step 2.Pick the point that the dimension line will go through (dimension point)

The following message is then written to the left hand side of the screen message area.

<Pick dimension point> [picks][fast][Menu]

The dimension line is then drawn parallel to the line from the start to the end vertices of the segment and going through the dimension point.



The **Aligned segment** option then starts again. That is, goes back to **Step 1**.

For information on how the dimension object appears, see 15.3.18.2 Dimension Styles.

15.3.6.7 Horizontal Segment

Horizontal segment labels the horizontal distance (delta x) between the start and end vertices of a selected segment. The segment can be straight, arc or a transition.

The dimension is added to the model given in the CAD toolbar.

a or **A** toggles Association **ON** and **OFF**. See 15.3.1 Dimension Association
c or **C** toggles Continuous **ON** and **OFF**. See 15.3.2 Continuous Mode
t or **T** brings up the dimension text format. See 15.5 Text Format for Dimensions and Leaders
s or **S** brings up the dimension style list. 15.3.18.2 Dimension Styles

Step 1.Pick the segment

Select Horizontal segment and the following message is written to the screen message area

<[Pick segment] (a)ssociative (s)tyle[default] forma(t) [(c)ontinuous OFF] > [picks][fast][Menu]

If Association is **ON** then the dimension is associated with the selected segment.

The horizontal distance (delta x) is calculated between the start and end vertices of the selected segment.

Step 2.Pick the point that the dimension line will go through (dimension point)

The following message is then written to the screen message area.

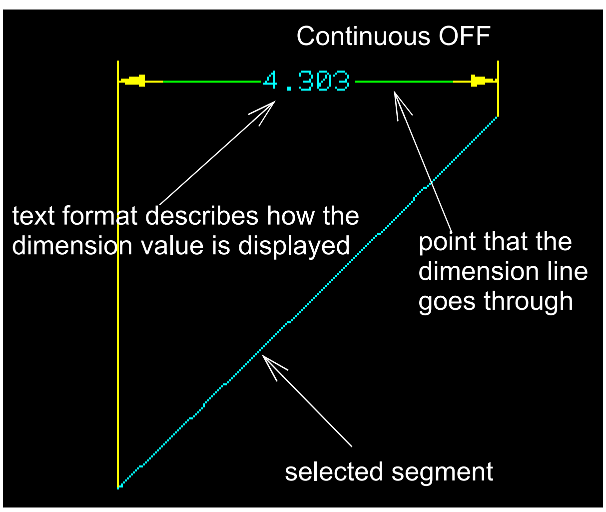
<Pick dimension point> [picks][fast][Menu]

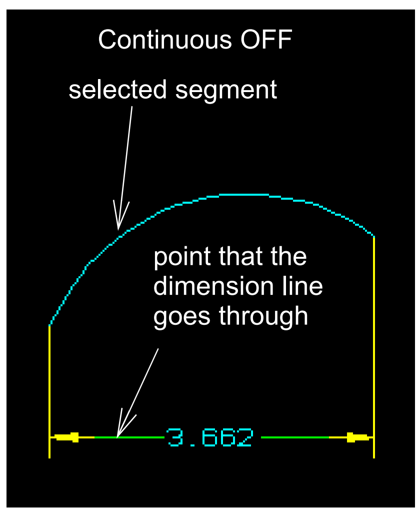
The horizontal dimension line is then drawn parallel to the x-axis going through the dimension point.

Step 3.I

If Continuous is **OFF**

The **Horizontal segment** option then starts again. That is, goes back to **Step 1.**

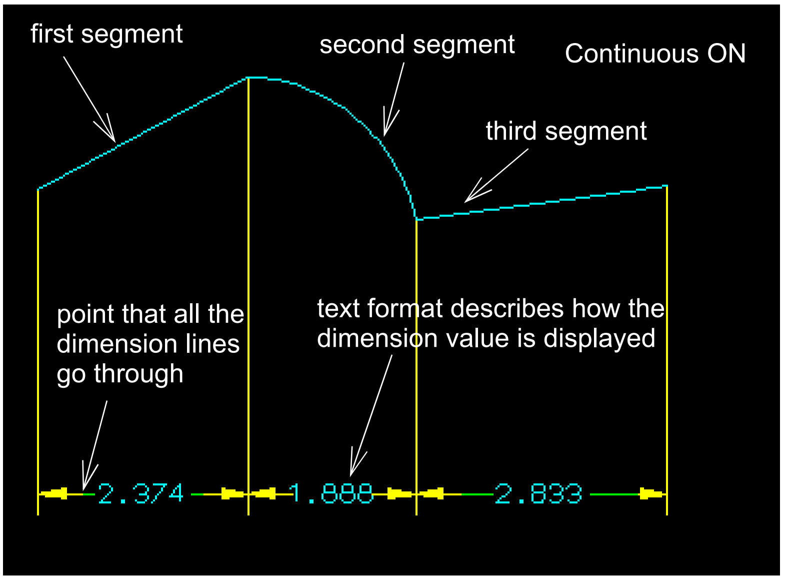




If Continuous is **ON**

If Continuous is **ON** then you are asked to pick another segment and another Horizontal segment dimension is created for the next segment and the dimension point is the same as the previous dimension point.

This repeats until Continuous is toggle **OFF** and then the next selected segment is the last horizontal dimension in the sequence.



For information on how the dimension object appears, see 15.3.18.2 Dimension Styles.

15.3.6.8 Vertical Segment

Vertical segment labels the vertical distance (delta y) between end vertices of a selected segment.The segment can be straight, arc or a transition.

The dimension is added to the model given in the CAD toolbar.

a or **A** toggles Association **ON** and **OFF**. See 15.3.1 Dimension Association
c or **C** toggles Continuous **ON** and **OFF**. See 15.3.2 Continuous Mode
t or **T** brings up the dimension text format. See 15.5 Text Format for Dimensions and Leaders
s or **S** brings up the dimension style list. 15.3.18.2 Dimension Styles

Step 1.Pick the segment

Select Vertical segment and the following message is written to the screen message area

```
<[Pick segment] (a)ssociative (s)tyle[default] forma(t) [(c)ontinuous OFF] > [picks][fast][Menu]
```

If Association is **ON** then the dimension is associated with the selected segment.

The vertical distance (delta y) is calculated between the start and end vertices of the selected segment.

Step 2.Pick the point that the dimension line will go through (dimension point)

The following message is then written to the screen message area.

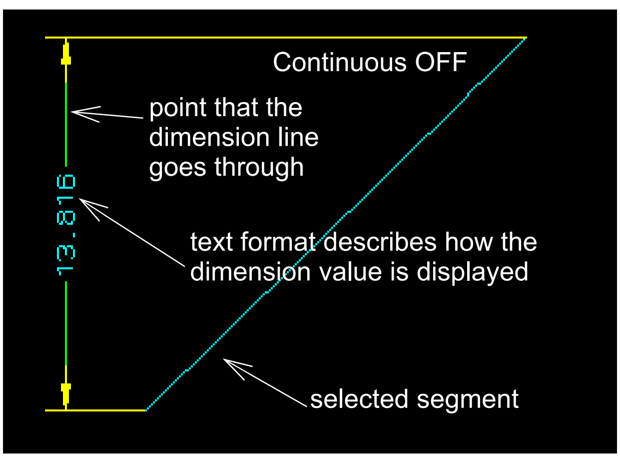
```
<Pick dimension point> [picks][fast][Menu]
```

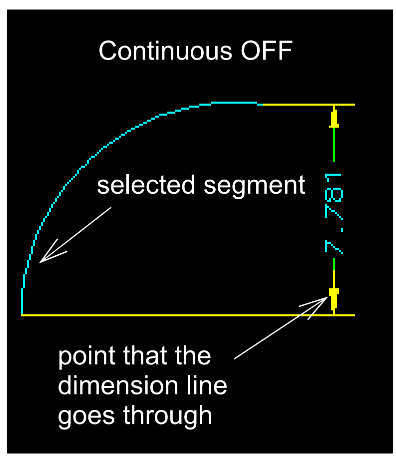
The vertical dimension line is then drawn parallel to the y-axis going through the dimension point.

Step 3.I

If Continuous is **OFF**

The **Vertical segment** option then starts again. That is, goes back to **Step 1**.

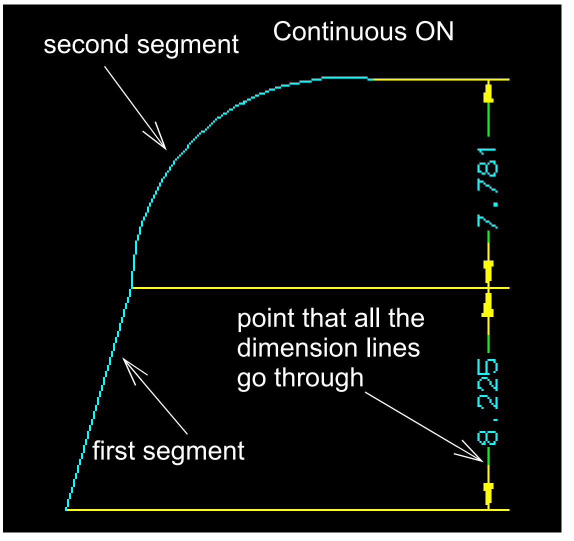




If Continuous is **ON**

If Continuous is **ON** then you are asked to pick another segment and another Vertical segment dimension is created for the next segment and the dimension point is the same as the previous dimension point.

This repeats until Continuous is toggle **OFF** and then the next selected segment is the last vertical dimension in the sequence.



For information on how the dimension object appears, see 15.3.18.2 Dimension Styles.

15.3.6.9 Rotated Segment

Rotated segment labels the 2D distance between the start vertex of a segment, and the end vertex of the segment dropped onto the line with a user given angle (Rotation) and going through the first vertex.

The dimension is added to the model given in the CAD toolbar.

a or **A** toggles Association **ON** and **OFF**. See 15.3.1 Dimension Association
c or **C** toggles Continuous **ON** and **OFF**. See 15.3.2 Continuous Mode
t or **T** brings up the dimension text format. See 15.5 Text Format for Dimensions and Leaders
s or **S** brings up the dimension style list. 15.3.18.2 Dimension Styles

Step 1.

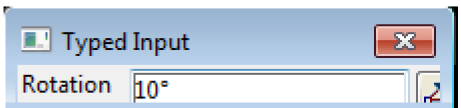
Select **Rotated segment** and the following message is written to the screen message area

<[Pick segment] (a)ssociative (s)tyle[default] forma(t) [(c)ontinuous OFF] an(g)le[10°] > [picks][fast][Menu]

If Association is **ON** then the dimension is associated with the selected segment.

Step 2.Type g to Change the Rotation Value

If the Rotation is to be changed, type **g** or **G** and the **Rotation** *Typed Input* box is then displayed.



Type in the Rotation value and press <Enter>.

The Rotation is in degrees in hp notation and is measured in an anticlockwise direction from the positive x axis.

Step 3.Pick the segment

The following message is written to the screen message area.

<[Pick segment] (a)ssociative (s)tyle[default] forma(t) [(c)ontinuous OFF] an(g)le[10°] > [picks][fast][Menu]

If Association is **ON** then the dimension is associated with the selected segment.

Step 4.Pick the point that the dimension line will go through (dimension point)

The following message is then written to the screen message area.

<Pick dimension point> [picks][fast][Menu]

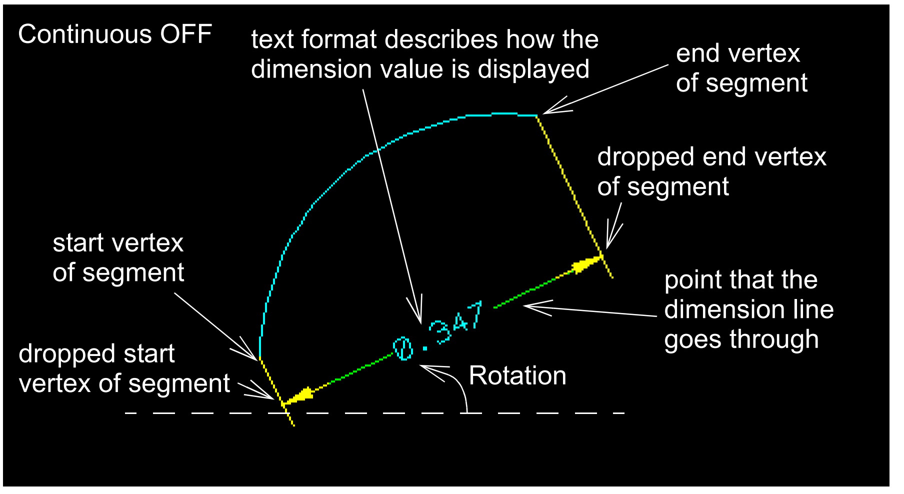
The **2D (plan) distance** is calculated between the start of the segment and the end of the segment dropped onto the line with the user given angle (Rotation) going through the start of the selected segment.

The dimension line is then drawn with the angle of the dimension line equal to the entered **Rotation**.

Step 5.

If Continuous is **OFF**

The **Rotated** option then starts again. That is, goes back to **Step 1.**



If Continuous is **ON**

If Continuous is ON then you are asked to pick another segment. The Rotation angle can also be changed.

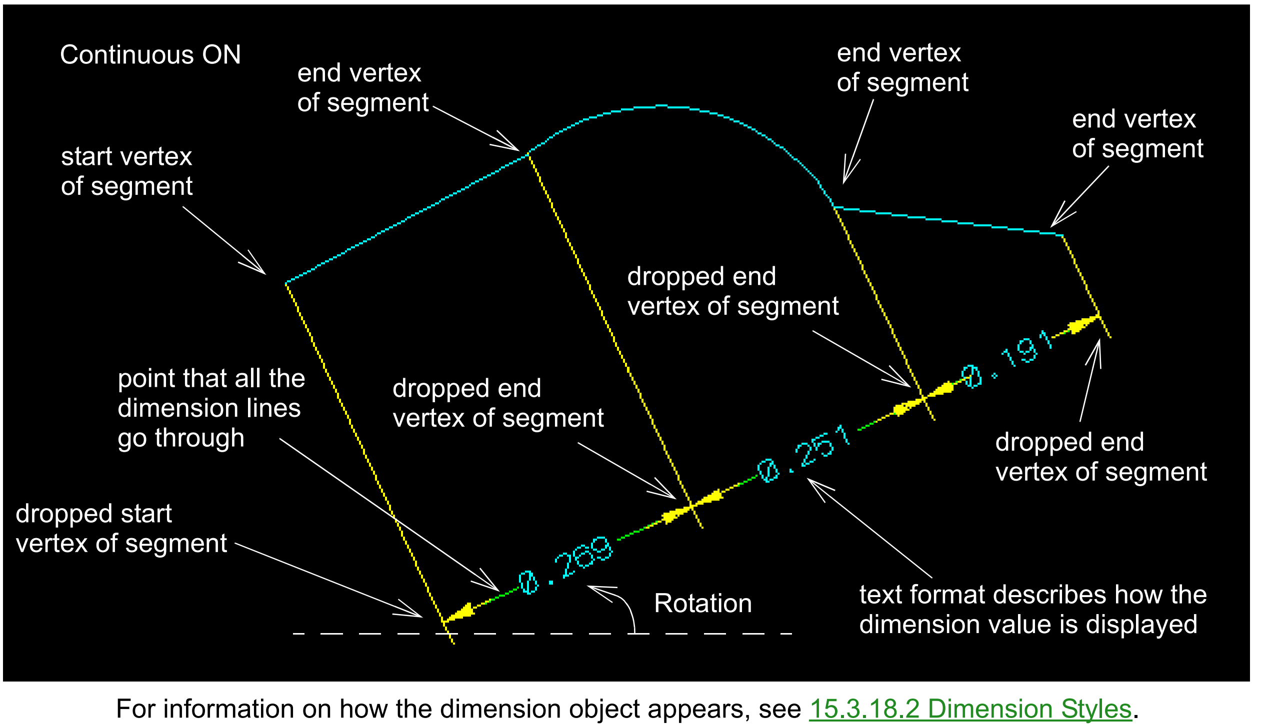
```
<[Pick segment] (a)ssociative (s)tyle[default] forma(t) [(c)ontinuous ON] an(g)le[10°] > [picks][fast][Menu]
```

When another segment is selected, the start and end vertices are dropped onto the line with the user given angle (Rotation) going through the current dimension point.

The **2D (plan) distance** is calculated between the dropped ends of the segment.

This repeats until Continuous is toggle **OFF** and then the next selected segment is the last **Rotated segment** dimension in the sequence.

The **Rotated segment** option then starts again. That is, goes back to **Step 1**.



For information on how the dimension object appears, see 15.3.18.2 Dimension Styles.

15.3.7 Length

Length labels the **2D length** of a selected **segment**. The segment may be a straight, arc, transition or offset transition.

The dimension is added to the model given in the CAD toolbar.

a or **A** toggles association **ON** and **OFF**. See 15.3.1 Dimension Association
c or **C** toggles Continuous **ON** and **OFF**. See 15.3.2 Continuous Mode
o or **O** toggles offset **ON** and **OFF**. See An Important Note about the Offset Toggle
s or **S** brings up the dimension style list. 15.3.18.2 Dimension Styles

Step 1.Pick the segment

Select **Length** and the following message is written to the screen message area

<[Pick segment] (a)ssociative (s)tyle[default] forma(t) [(o)ffset ON] [(c)ontinuous OFF] > [picks][fast][Menu]

If **association** is **ON** then the dimension is associated with the selected segment.

Note: Continuous mode is only available if **offset** is **ON**.

The 2D length of the selected segment is calculated.

Step 2.Pick the dimension point

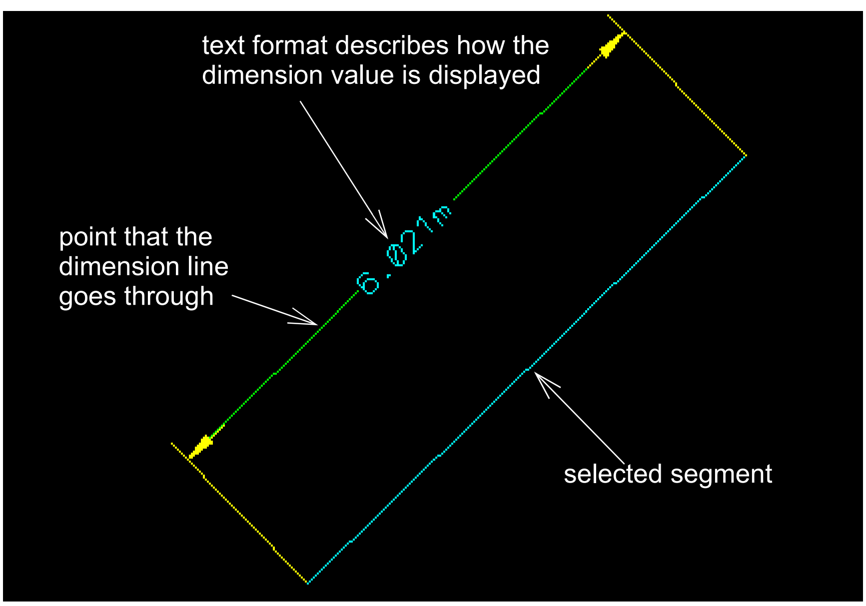
The following message is then written to the screen message area.

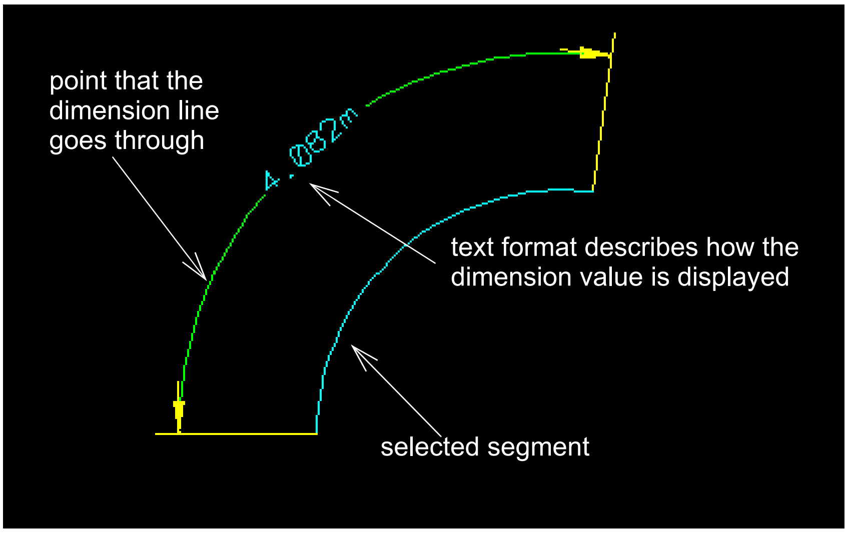
<Pick dimension point> [picks][fast][Menu]

The dimension line is then drawn parallel to the segment and going through the dimension point.

If **offset** is **ON** then the distance between the dimension point and the segment (the offset distance) is remembered with the dimension.

If **offset** is **OFF** then the dimension point itself is remembered with the dimension.





Step 3.I

If **Continuous** is **OFF:**

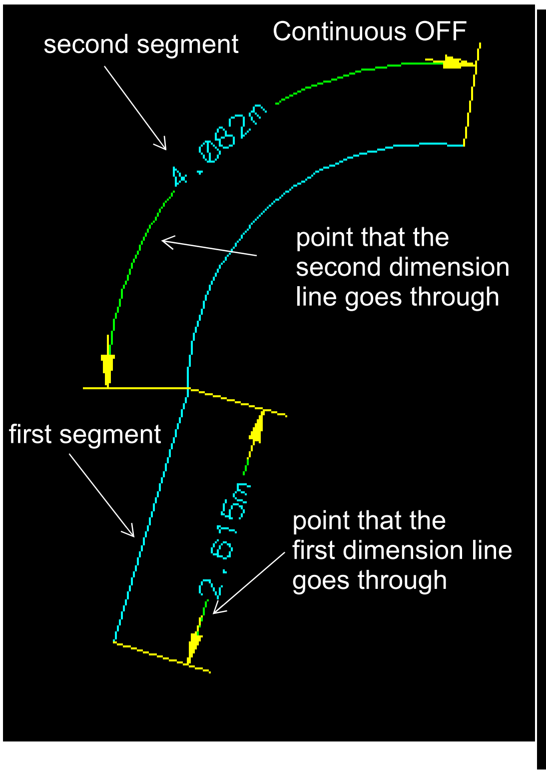
The **Length** option then starts again. That is, goes back to **Step 1**.

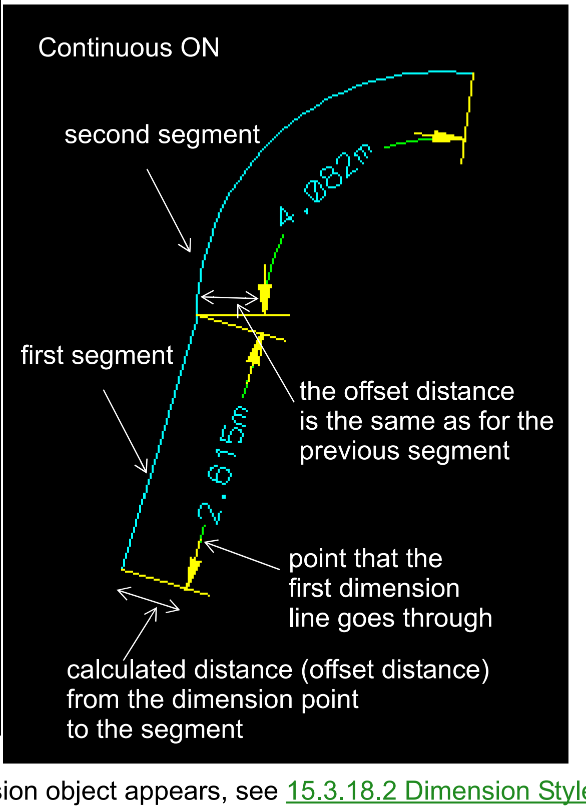
Continuous is **ON** (and for that **Offset** must also be ON):

If Continuous is **ON** then you are asked to pick another segment and another Length dimension is created for the next segment. The distance the dimension line is from the segment (offset distance) is the same as it was for the previous segment.

This repeats until Continuous is toggle **OFF** and then the next selected segment is the last Length dimension in the sequence.

The **Length** option then starts again. That is, goes back to **Step 1**.





For information on how the dimension object appears, see 15.3.18.2 Dimension Styles.

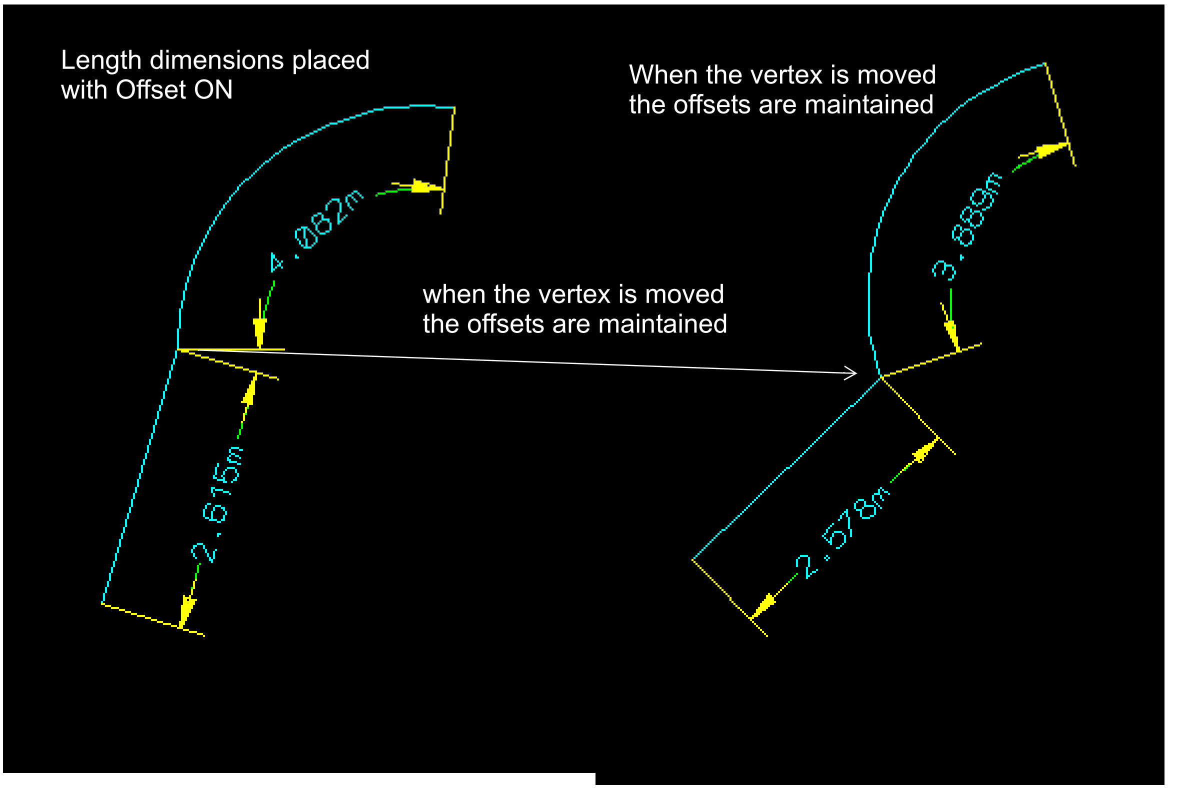
An Important Note about the Offset Toggle

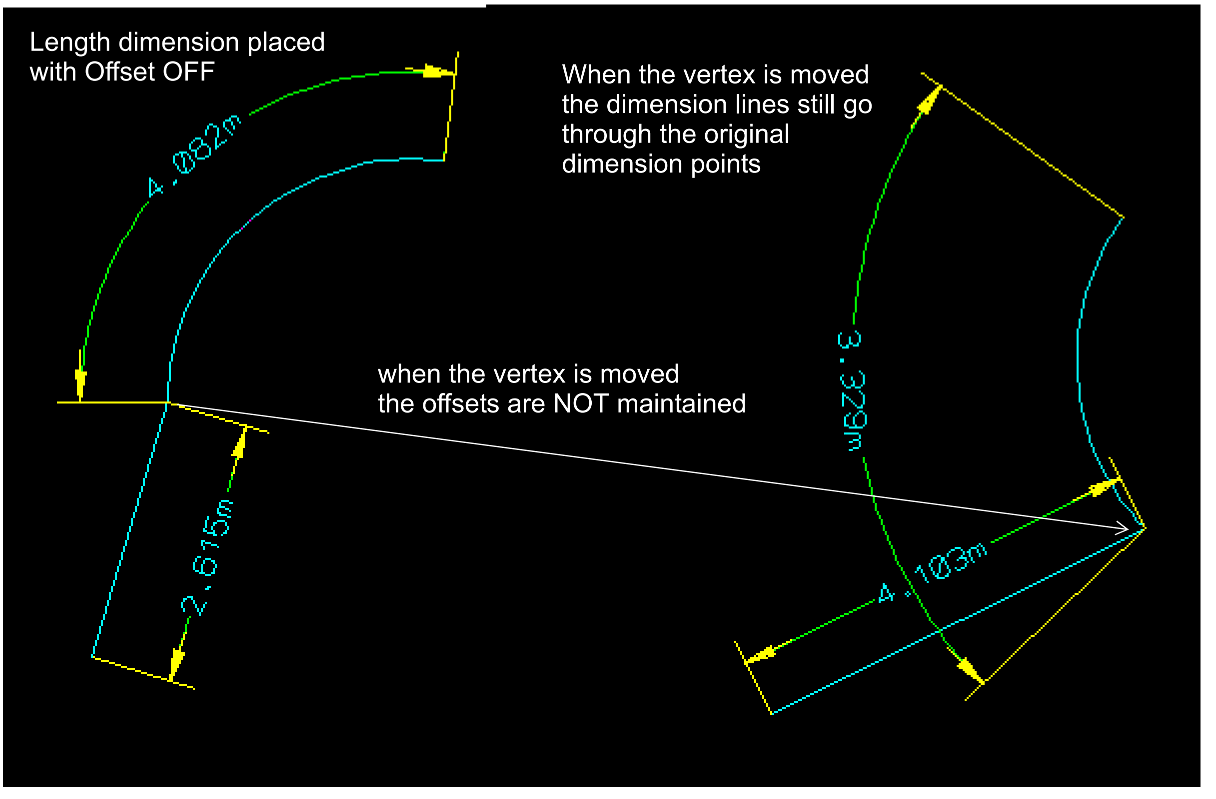
When the dimension line is first drawn, the difference between **Offset ON** and **Offset OFF** is not apparent.

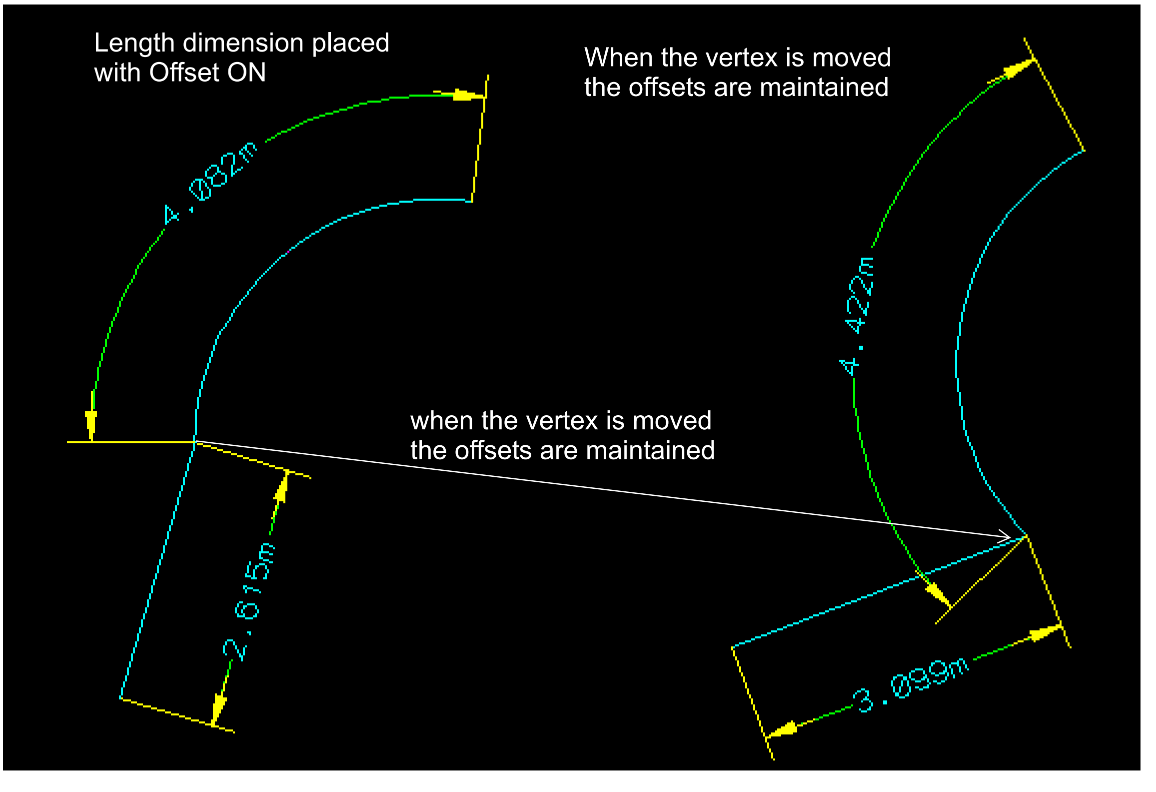
But when you move a vertex and the associated **Length** dimensions go with it, the difference is move obvious.

For the **Length** dimensions that are created with **Offset ON**, the **original offsets** are **maintained**. So the new dimension lines will be the same parallel distance from the segments as when they were originally dimensioned.

For the **Length** dimensions that are created with **Offset OFF**, the **original dimensions points** are **maintained**.So the new dimension lines will go through the original dimension points and so their parallel distance from the segments will be totally different from before




Length dimension placed
with Offset OFF
4.082m
2.615m
When the vertex is moved
the dimension lines still go
through the original
dimension points
3.329m
when the vertex is moved
the offsets are NOT maintained
4.103m


Length dimension placed
with Offset ON
4.082m
2.615m
When the vertex is moved
the offsets are maintained
4.422m
when the vertex is moved
the offsets are maintained
3.099m

15.3.8 Drop Segment

Drop segment labels the 2D distance between a selected point, and the point is dropped perpendicularly onto a selected base segment.

If the base segment is a line then the drop may occur on the extended segment.

If the base segment is an arc then it will drop to the outside or inside of the visible part of the arc.

If the base segment is a circle then the drop will go to the inside or outside of the circle depending on whether the point is inside or outside the circle.

The dimension is added to the model given in the CAD toolbar.

a or **A** toggles Association **ON** and **OFF**. See 15.3.1 Dimension Association
t or **T** brings up the dimension text format. See 15.5 Text Format for Dimensions and Leaders
s or **S** brings up the dimension style list. 15.3.18.2 Dimension Styles
z or **Z** toggles *Zero Offset* mode **ON** and **OFF**. See 15.3.4 Zero Offset Mode

Step 1.

Select **Drop segment** and the following message is written to the screen message area

```
<[(a)ssociation ON] Pick start point or change format (t)ext (s)tyle[d1] [(z)ero offset ON] > [picks][fast][Menu]
```

Pick the start point.

If Association is **ON** then the start of the dimension is associated with the selected start point.

Step 2.

The following message is written to the screen message area asking for a segment to be selected. The segment can be a straight, an arc, a transition or an offset transition.

```
<[(a)ssociation ON] Pick base segment> [picks][fast][Menu]
```

Pick the segment.

If Association is **ON** then the end of the dimension is associated with the selected segment.

The picked point is dropped perpendicularly onto the selected segment (or the extended line segment) and the 2D distance is calculated between these two points.

Step 3.

If **Zero Offset** is **OFF**, the following message is then written to the screen message area.

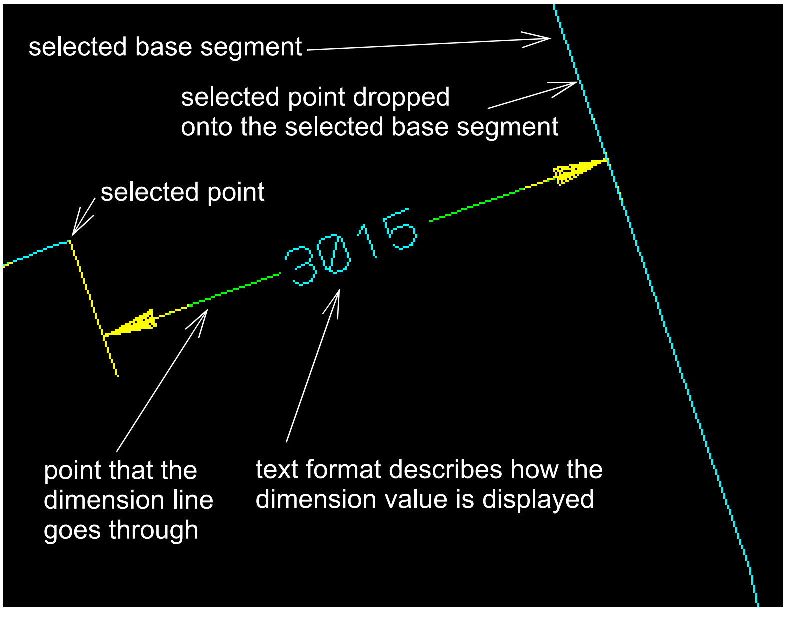
```
<Pick dimension point> [picks][fast][Menu]
```

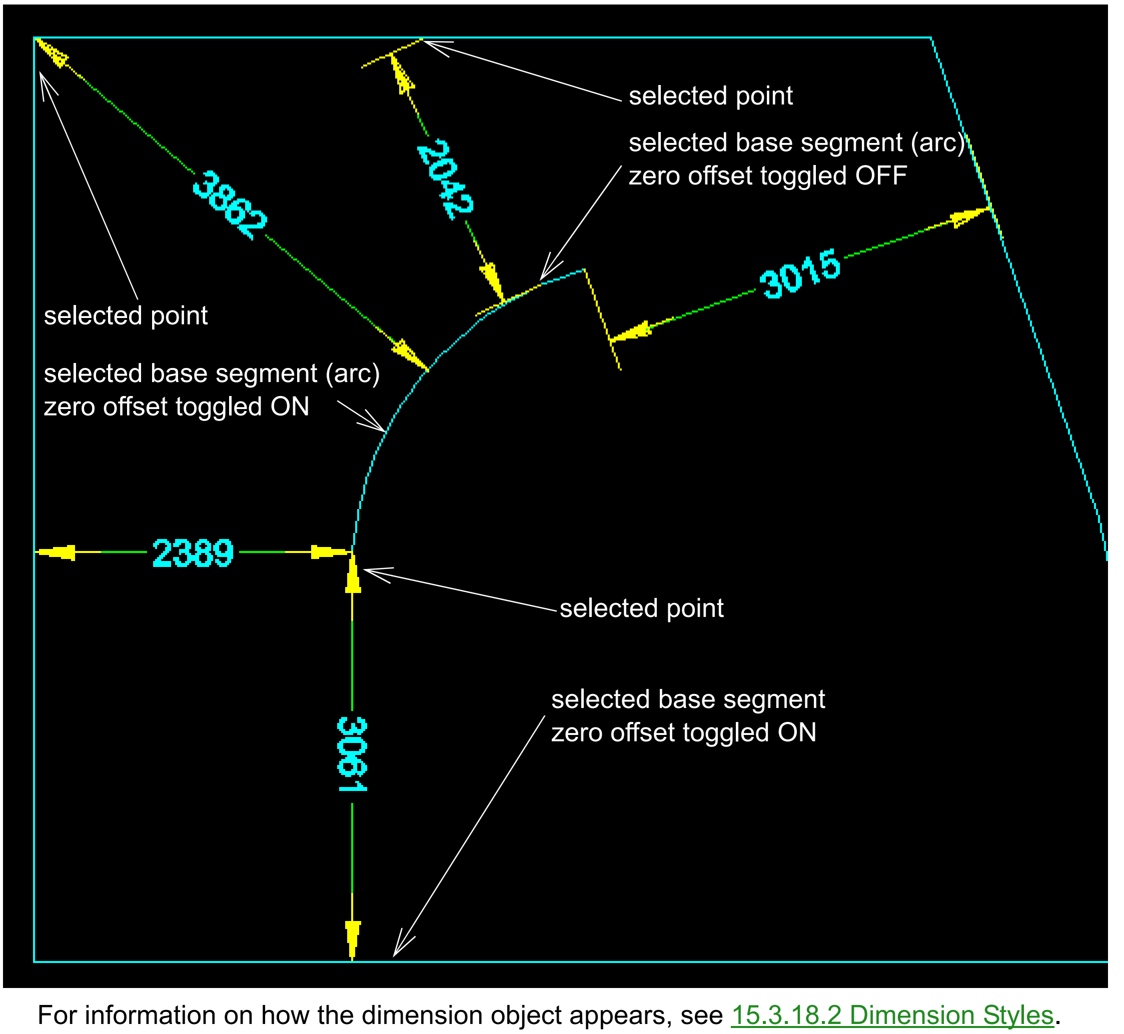
Pick the point that the dimension line will go through (dimension point).

The dimension line is then drawn parallel to the line between the selected point and the selected point dropped perpendicularly onto the selected segment, and going through the dimension point.

If **Zero Offset** is **ON**, then no dimension point is picked because the dimension point is taken to be on the line joining the picked point and the dropped point.

For both cases of **Zero Offset**, the **Drop segment** option then starts again. That is, goes back to **Step 1**.



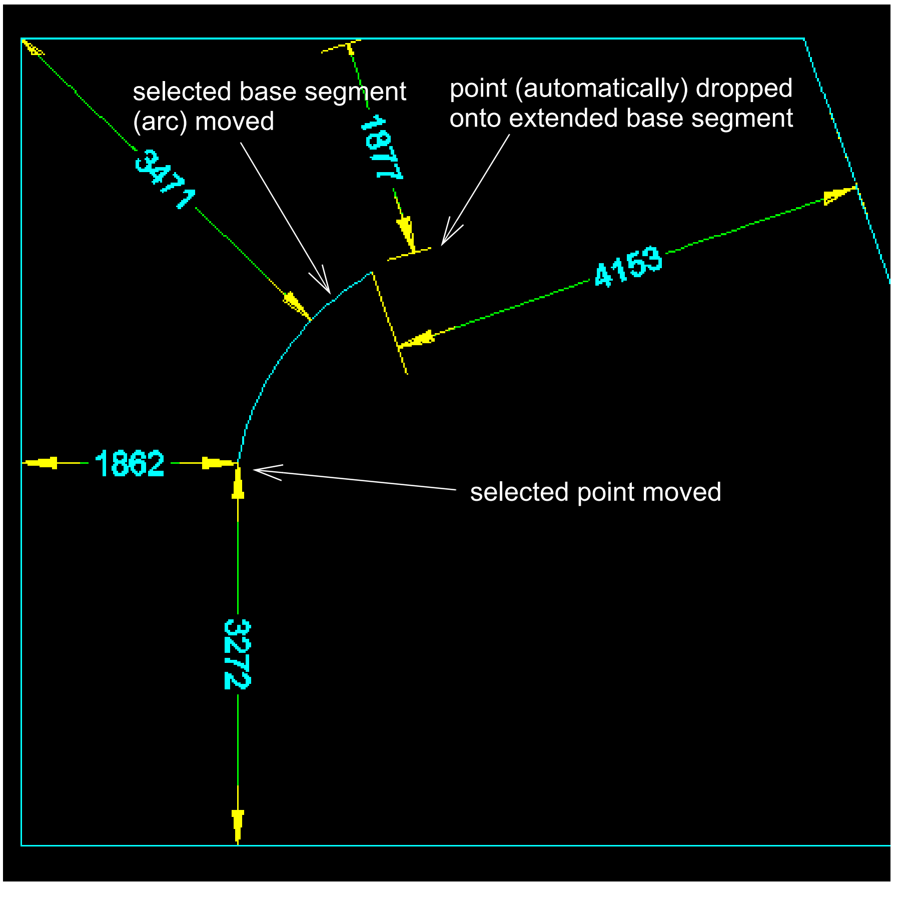


For information on how the dimension object appears, see 15.3.18.2 Dimension Styles.

Note on Automatic Recalculation of Drop Segment Dimension When the Point or Segment Moves

When **Association** in turned **ON**, the picked point and the picked base segment are associated with the *Drop segment* dimension. So if either the picked point or the picked base segment is moved, the drop dimensions automatically recalcs.

Also note that the drop onto the base segment may involve the extended base segment because the drop point may not be between the end points of the selected base segment.



15.3.9 Drop String

Drop string labels the 2D distance between a selected point and the point dropped onto a selected base string.

By Dropped onto a string we mean that the point is first checked to see that it can be dropped perpendicularly onto a segment of the selected string (not an extended segment), or if there is no segment that the point can drop perpendicularly onto then it will drop onto the **nearest vertex** of the string.

If the base string is an arc then it will drop to the outside or inside of the visible part of the arc.

If the base string is a circle then the drop will go to the inside or outside of the circle depending on whether the point is inside or outside the circle.

Hence **Drop string** will always end on the base string.

The dimension is added to the model given in the CAD toolbar.

a or **A** toggles Association **ON** and **OFF**. See 15.3.1 Dimension Association
t or **T** brings up the dimension text format. See 15.5 Text Format for Dimensions and Leaders
s or **S** brings up the dimension style list. 15.3.18.2 Dimension Styles

Step 1.Pick the start point

Select **Drop string** and the following message is written to the screen message area

```
<[Pick start point] (a)ssociative (s)tyle[default] forma(t) > [picks][fast][Menu]
```

If Association is **ON** then the start of the dimension is associated with the selected start point.

Step 2.Pick the base string

The following message is written to the screen message area asking for a base string to be selected.

```
<(a)ssociative Pick base string> [picks][fast][Menu]
```

If Association is **ON** then the end of the dimension is associated with the selected base string.

The picked point is dropped onto the selected base string and the 2D distance is calculated between these two points.

The dimension line is then drawn between the selected point and the selected point dropped onto the selected base string.

The **Drop string** option then starts again. That is, goes back to **Step 1**.

15.3.10 Area

Area labels the plan area of a selected string.

The dimension is added to the model given in the CAD toolbar.

a or **A** toggles Association **ON** and **OFF**. See 15.3.1 Dimension Association
t or **T** brings up the dimension text format. See 15.5 Text Format for Dimensions and Leaders
s or **S** brings up the dimension style list. 15.3.18.2 Dimension Styles

Step 1.Pick the polygon to dimension

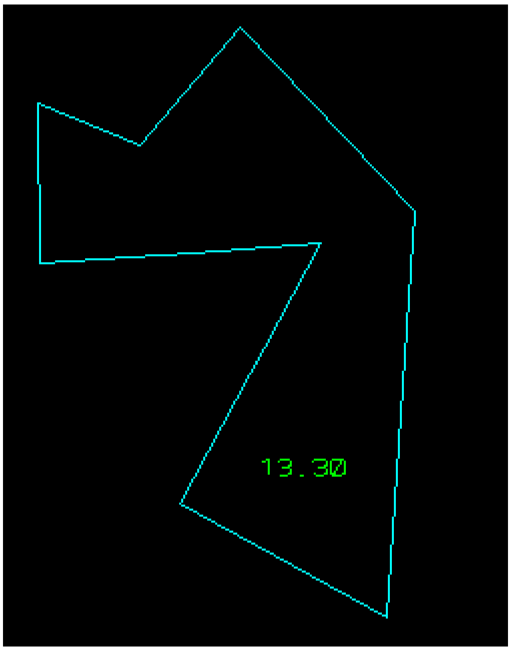
Select **Area** and the following message is written to the screen message area

<[Pick polygon] (a)ssociative (s)tyle[default] forma(t) > [picks][fast][Menu]

If Association is **ON** then the dimension is associated with the polygon.

If the selected string is not closed, the area is calculated for the polygon formed by joining the first and last vertices of the string.

The plan area is calculated for the polygon and placed inside the polygon.



The **Area** option then starts again. That is, goes back to **Step 1**.

For information on how the dimension object appears, see 15.3.18.2 Dimension Styles.

15.3.11 Jogged radius

This option has not yet been completed

- radius of a selected arc with jog

  Pick an arc or a segment with a radius.

  Then pick the start point of the jog and then the dimension point.

  A dimension line is draw with a start symbol at the start and then a job between the dimension point and the start point of the jog.

The dimension is added to the model given in the CAD toolbar.

15.3.12 Diameter

This option has not yet been completed

Diameter labels the diameter of an arc or a segment with a non zero radius.

The dimension is added to the model given in the CAD toolbar.

a or **A** toggles Association **ON** and **OFF**. See 15.3.1 Dimension Association
t or **T** brings up the dimension text format. See 15.5 Text Format for Dimensions and Leaders
s or **S** brings up the dimension style list. 15.3.18.2 Dimension Styles

Step 1.

Select Diameter and the following message is written to the screen message area

<[(A)ssociation ON] Pick arc or choose (s)tyle[diags - yes fixed len dim line] > [picks][fast][Menu]

Pick the arc or segment with non zero radius.

If Association is **ON** then the dimension is associated with the selected arc or segment.

The segment must have a radius. That is, it can't be a line segment.

The diameter is calculated for the arc or segment.

Step 2.

The following message is then written to the left hand side of the screen message area.

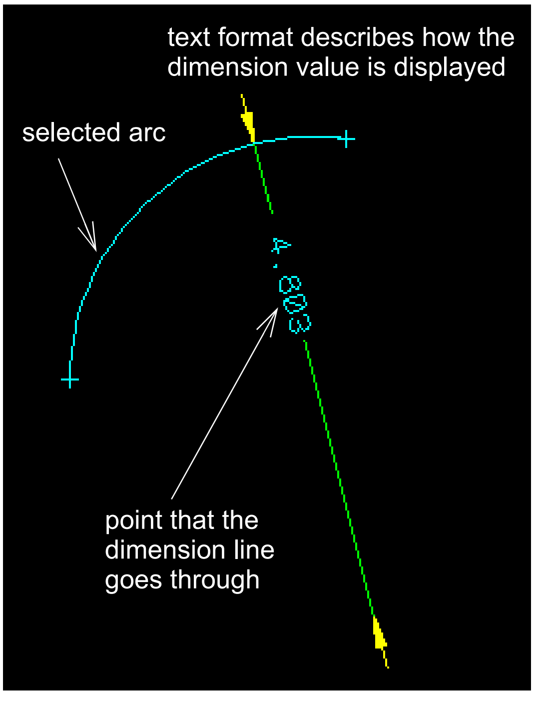
<Pick dimension point> [picks][fast][Menu]

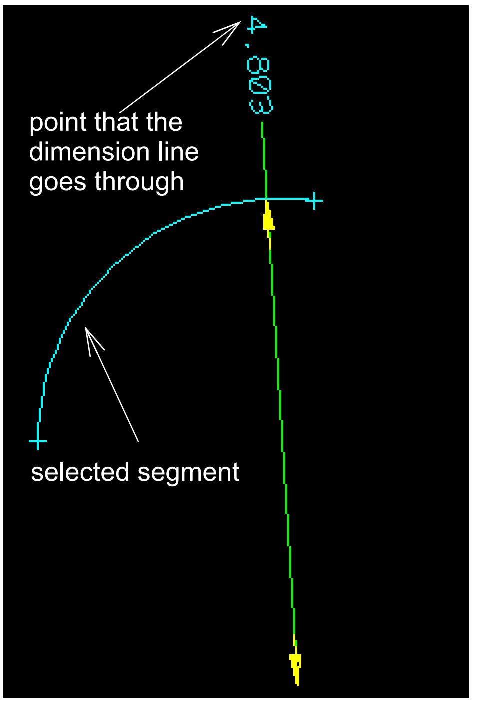
t or **T** brings up the dimension text format. See 15.5 Text Format for Dimensions and Leaders

Pick the point that the dimension line will go through (dimension point).

The dimension line is then drawn from the centre of the arc and through the dimension point and onto the arc. The text is placed at the dimension point.

If the dimension point is outside the arc, the dimension line continues through the arc.





The **Diameter** option then starts again. That is, goes back to Step 1.

For information on how the dimension object appears, see 15.3.18.2 Dimension Styles.

15.3.13 Radius

This option has not yet been completed

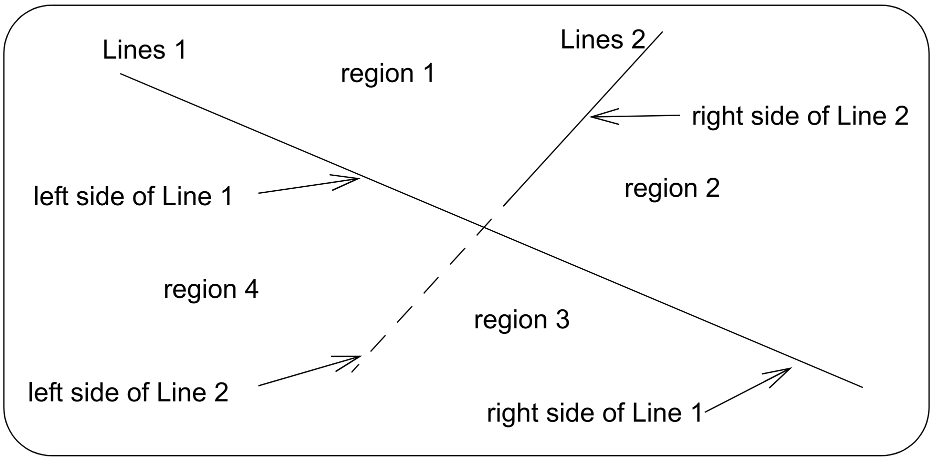
Radius of an arc or of a segment with a radius.

The dimension is added to the model given in the CAD toolbar.

15.3.14 Angle 2 Lines

Angle 2 lines labels the angle between two selected lines.

To define this dimension, the intersection point of the two extended lines divides the extended lines into two sides and the plane into four regions.



Although the lines can only be initially picked by the non extended part of the lines, the dimension can go between any of the halves of the extended lines.

The dimension is added to the model given in the CAD toolbar.

a or **A** toggles Association **ON** and **OFF**. See 15.3.1 Dimension Association
t or **T** brings up the dimension text format. See 15.5 Text Format for Dimensions and Leaders.
s or **S** brings up the dimension style list. 15.3.18.2 Dimension Styles

Step 1.Pick the side to start the dimension from

Select **Angle 2 lines** and the following message is written to the screen message area

<[Pick start side] (a)ssociative (s)tyle[default] forma(t) > [picks][fast][Menu]

If Association is **ON** then the start of the dimension is associated with the selected start line.

Step 2.Pick the side to end the dimension on

The following message is written to the screen message area.

<[Pick end side] (a)ssociative > [picks][fast][Menu]

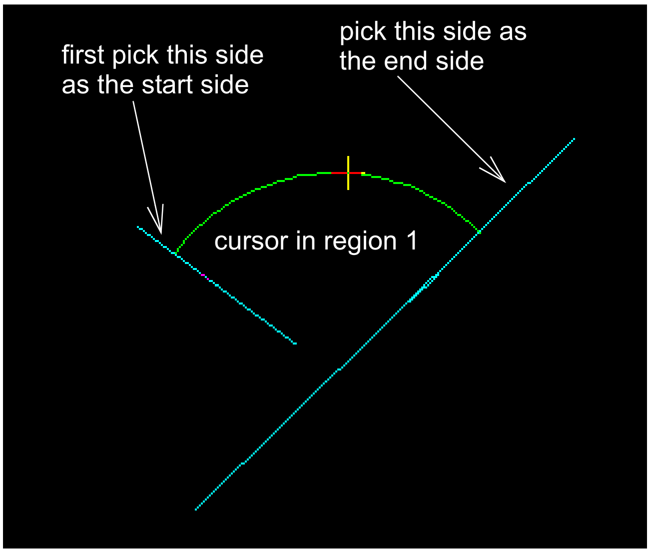
If Association is **ON** then the end of dimension is associated with the selected end line.

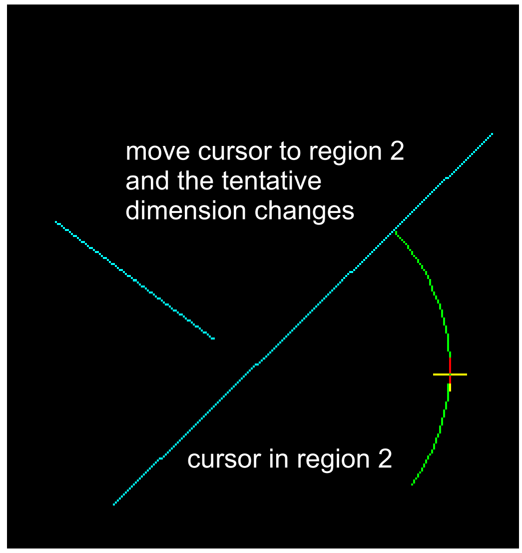
Step 3.

The following message is then written to the left hand side of the screen message area.

<Pick dimension point or reverse (d)irection (s)tart (e)nd line extension> [picks][fast][Menu]

As you move your cursor between the four regions, you will see the dimension line display as if the cursor was the selected dimension point.



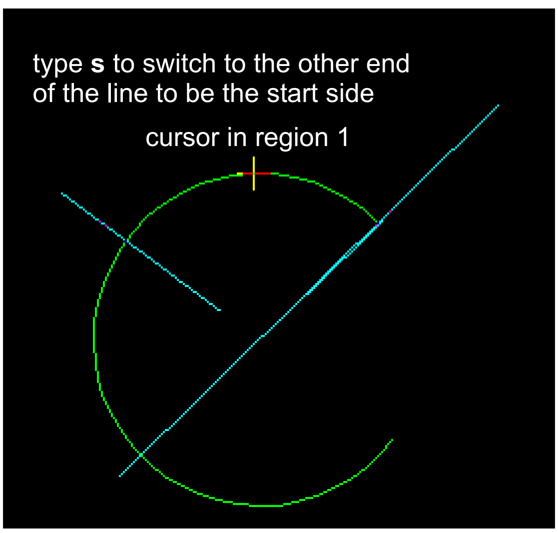


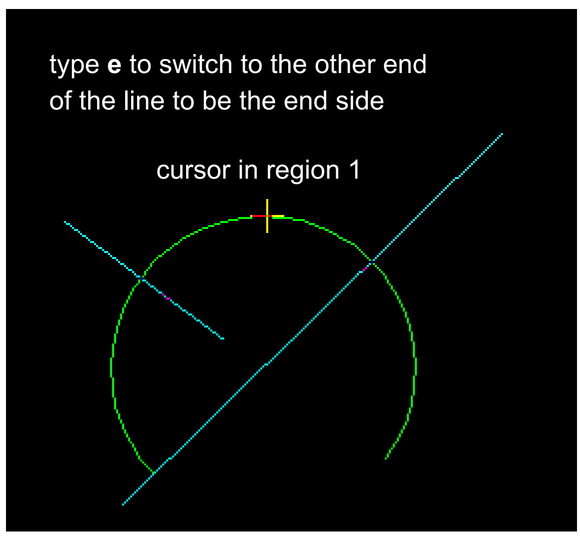
s or **S** takes the other end of the first line to start the dimension on.

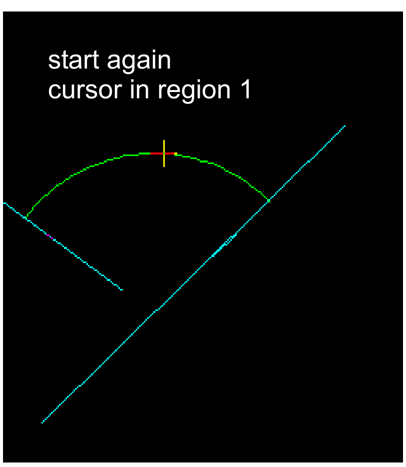
e or **E** takes the other end of the second line to end the dimension on.

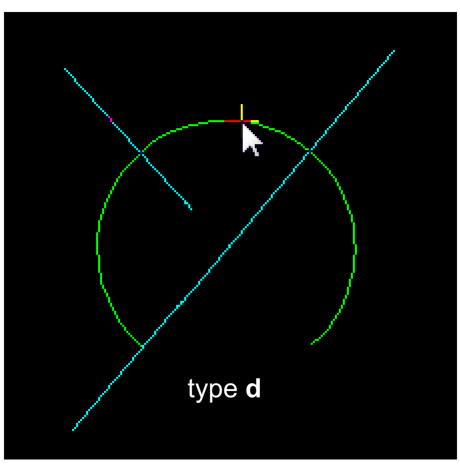
d or **D** toggles between a clockwise and counter clockwise angle for the dimension arc.

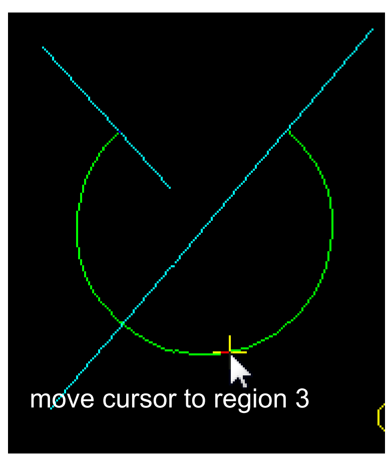
t or **T** can be typed to change the text format. See 15.5 Text Format for Dimensions and Leaders.





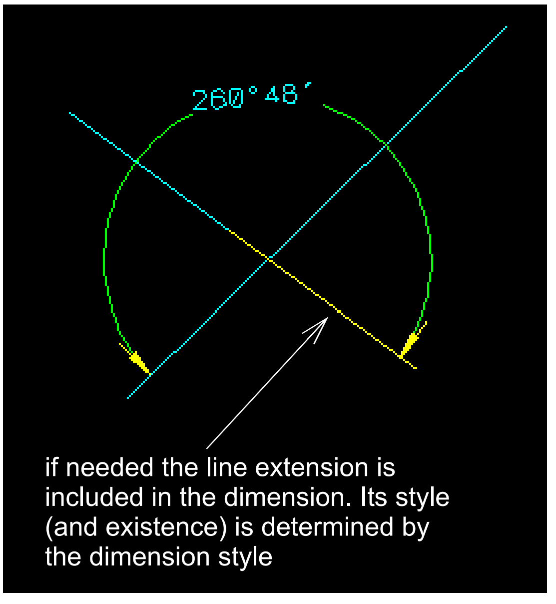


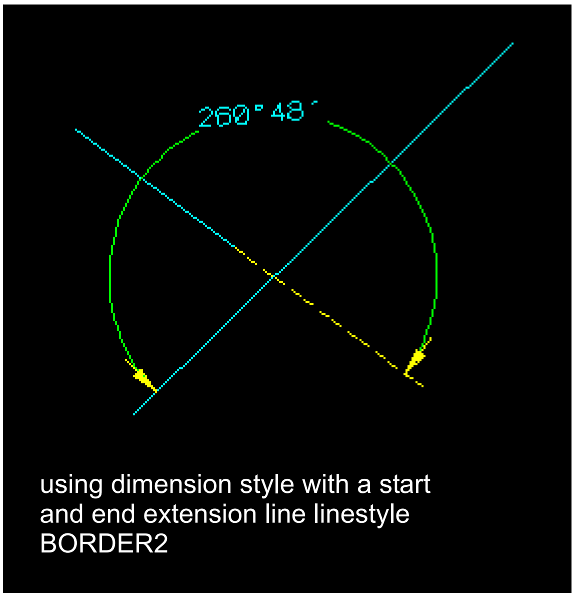




Pick the point that the dimension line will go through (dimension point).

The dimension line is then drawn from the start side to the end side and going through the dimension point.





The **Angle 2 lines** option then starts again. That is, goes back to **Step 1.**

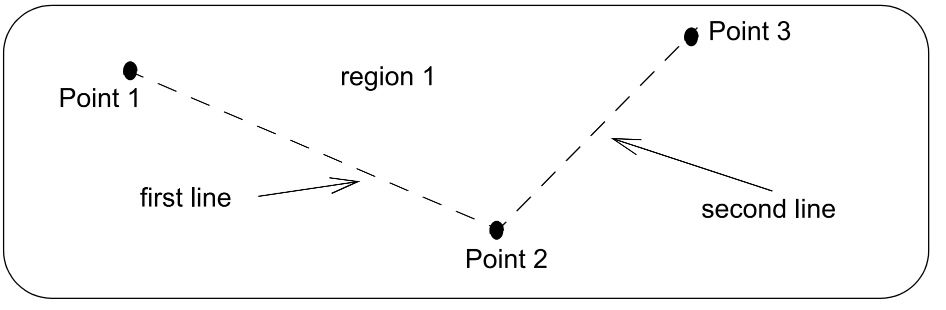
For information on how the dimension object appears, see 15.3.18.2 Dimension Styles.

15.3.15 Angle 3 points

Angle 3 points labels the angle between two temporary lines defined by 3 points. That is, it labels the angle between two lines that are constructed from the three points.

The first point is the start of the first line, the second point is the end of the first line and the start of the second line and the third point in the end of the second line.

The dimension is added to the model given in the CAD toolbar.



a or **A** toggles Association **ON** and **OFF**. See 15.3.1 Dimension Association
t or **T** brings up the dimension text format. See 15.5 Text Format for Dimensions and Leaders.
s or **S** brings up the dimension style list. 15.3.18.2 Dimension Styles

Step 1.Pick the first point of the first line

Select **Angle 3 points** and the following message is written to the screen message area

<[Pick start point] (a)ssociative (s)tyle[default] forma(t) > [picks][fast][Menu]

If Association is **ON** then the start of the dimension is associated with the selected point.

Step 2.Pick the second point that is the end of the first line and the start of the second line

The following message is written to the screen message area.

<[Pick vertex point] (a)ssociative > [picks][fast][Menu]

If Association is **ON** then the vertex of the dimension is associated with the selected point.

Step 3.Pick the third point that is the end of the second line

The following message is written to the screen message area.

<[Pick end point] (a)ssociative [(d)irection counter-clockwise] > [picks][fast][Menu]

If Association is **ON** then the end of the dimension is associated with the selected point.

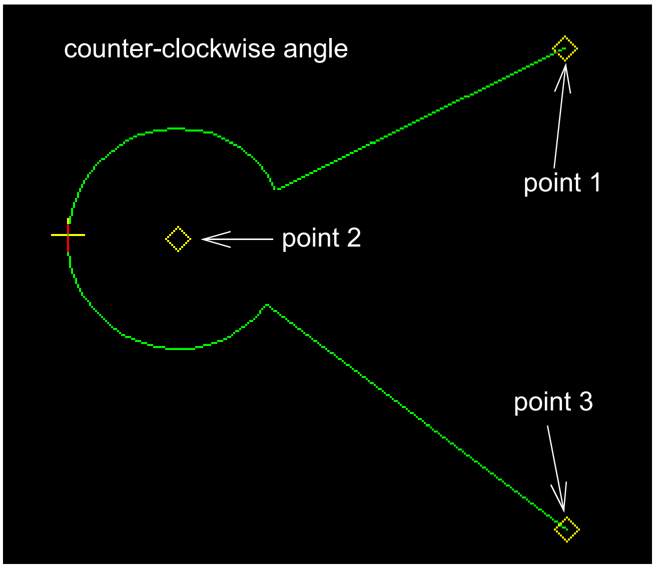
Step 4.Pick the point that the dimension arc will go through (dimension point)

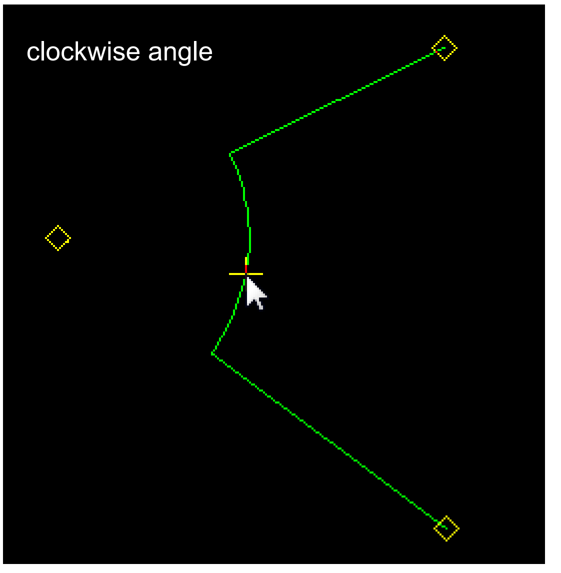
The following message is then written to the screen message area.

<[Pick dimension point] [(d)irection clockwise] > [picks][fast][Menu]

As you move your cursor you see how large the line of dimensions arc will be.

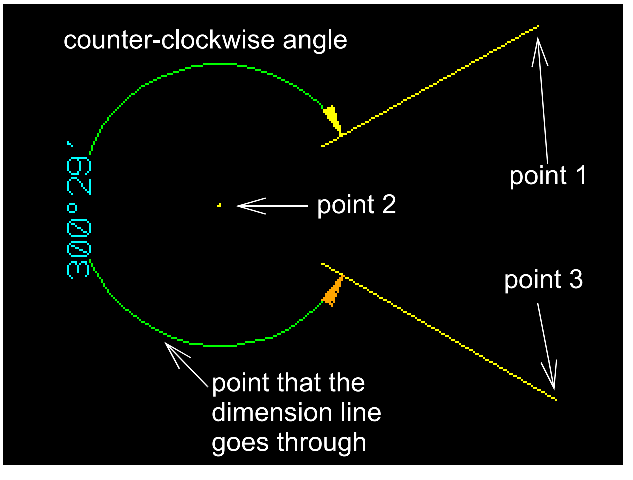
d or **D** toggles between a counter-clockwise and clockwise angle for the dimension arc.

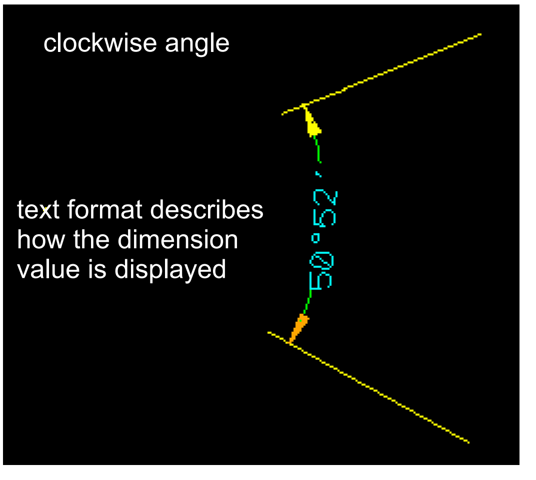




When the dimension point is selected, the dimension arc is drawn from the start side to the end side and going through the dimension point.

The **Angle 3 points** option then starts again. That is, goes back to **Step 1**.





For information on how the dimension object appears, see 15.3.18.2 Dimension Styles.

15.3.16 Angle Arc

Angle arc labels the angle subtended by an existing arc, or of a segment with a radius.

The dimension is added to the model given in the CAD toolbar.

a or **A** toggles Association **ON** and **OFF**. See 15.3.1 Dimension Association
t or **T** brings up the dimension text format. See 15.5 Text Format for Dimensions and Leaders
s or **S** brings up the dimension style list. 15.3.18.2 Dimension Styles

Step 1.Pick the arc of arc segment to dimension

Select **Angle arc** and the following message is written to the screen message area

<[Pick segment] (a)ssociative (s)tyle[default] forma(t) > [picks][fast][Menu]

If Association is **ON** then the dimension is associated with the polygon.

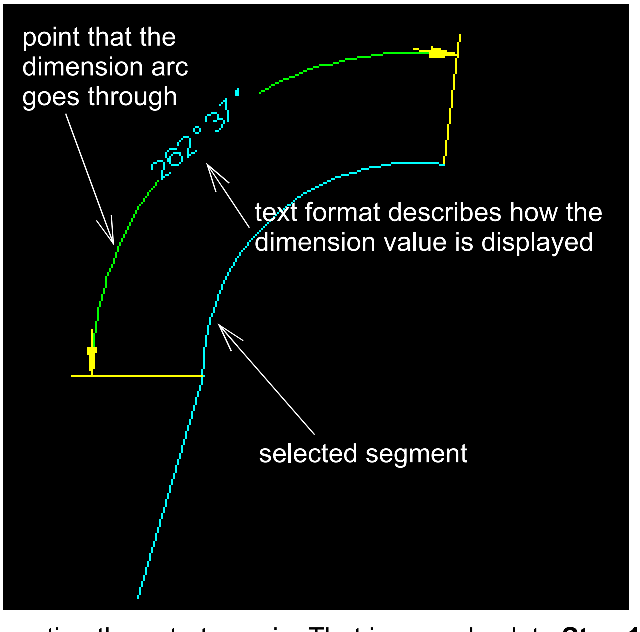
The angle of the arc can then be calculated.

Step 2.Pick the point that the dimension arc will go through (dimension point

The following message is then written to the left hand side of the screen message area.

<Pick dimension point> [picks][fast][Menu]

The dimension arc is then drawn parallel to the arc or segment and goes through the dimension point.



The **Angle arc** option then starts again. That is, goes back to **Step 1**.

For information on how the dimension object appears, see 15.3.18.2 Dimension Styles.

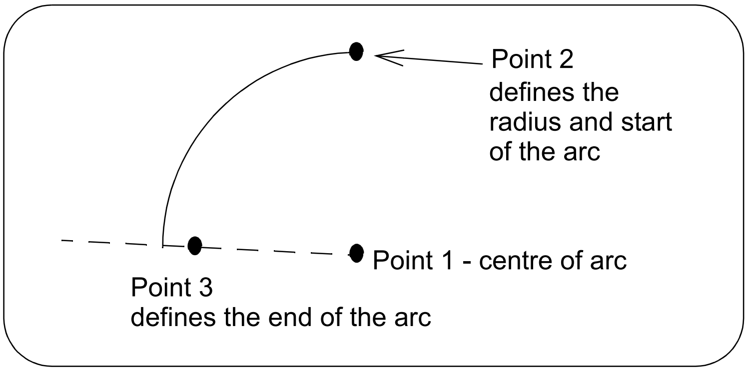
15.3.17 Arc Length by Centre, 2 Points

NOT currently on the menu.

Arc labels the length of an arc where the arc is temporarily defined by picking a centre point and two points on the arc.

Pick a position that will be centre point of the arc, and then the position that will be the first point of arc. This defines the centre, radius and start point of the arc.

A third position is then selected and it is dropped onto the arc defined by the first two points to give the end of the arc. This fully defines a temporary arc.



a or **A** toggles Association **ON** and **OFF**. See 15.3.1 Dimension Association
t or **T** brings up the dimension text format. See 15.5 Text Format for Dimensions and Leaders.
s or **S** brings up the dimension style list. 15.3.18.2 Dimension Styles

Step 1.

Select Arc and the following message is written to the screen message area

```
<[(a)ssociation ON] Pick centre point or choose (s)tyle[paper no clip orang end] c
```

Pick the position to be the centre point of the arc

If Association is **ON** then the start of the dimension is associated with the selected point.

Step 2.

The following message is written to the screen message area.

```
<[(a)ssociation ON] Pick start point> [picks][fast][Menu]
```

Pick the second point is the start of the arc

If Association is **ON** then the vertex of the dimension is associated with the selected point.

Step 3.

The following message is written to the screen message area.

```
<[(a)ssociation ON] Pick end point or change arc (d)irection [counter-clockwise] > [picks][fast][Menu]
```

d or **D** toggles between a counter-clockwise and clockwise angle for the dimension arc.

Pick the third point that is the end of the arc.

If Association is **ON** then the end of the dimension is associated with the selected point.

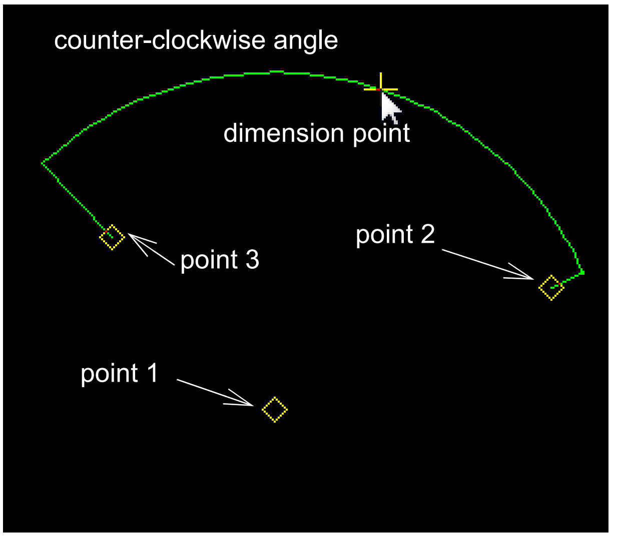
Step 4.

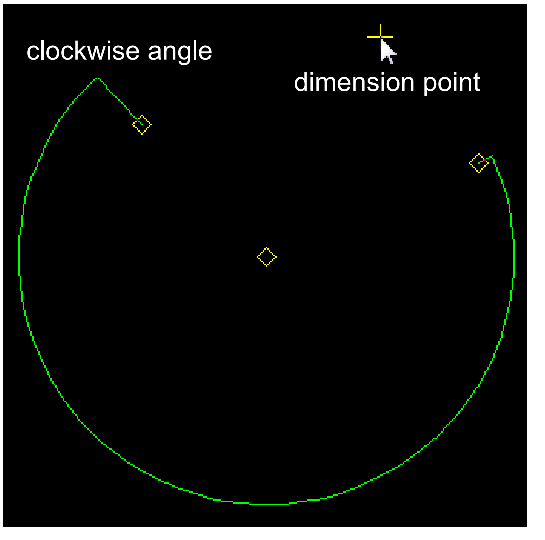
The following message is then written to the left hand side of the screen message area.

<[Pick dimension point] [(d)irection clockwise] > [picks][fast][Menu]

As you move your cursor you see how large the line of dimensions arc will be.

d or **D** toggles between a counter-clockwise and clockwise angle for the dimension arc.

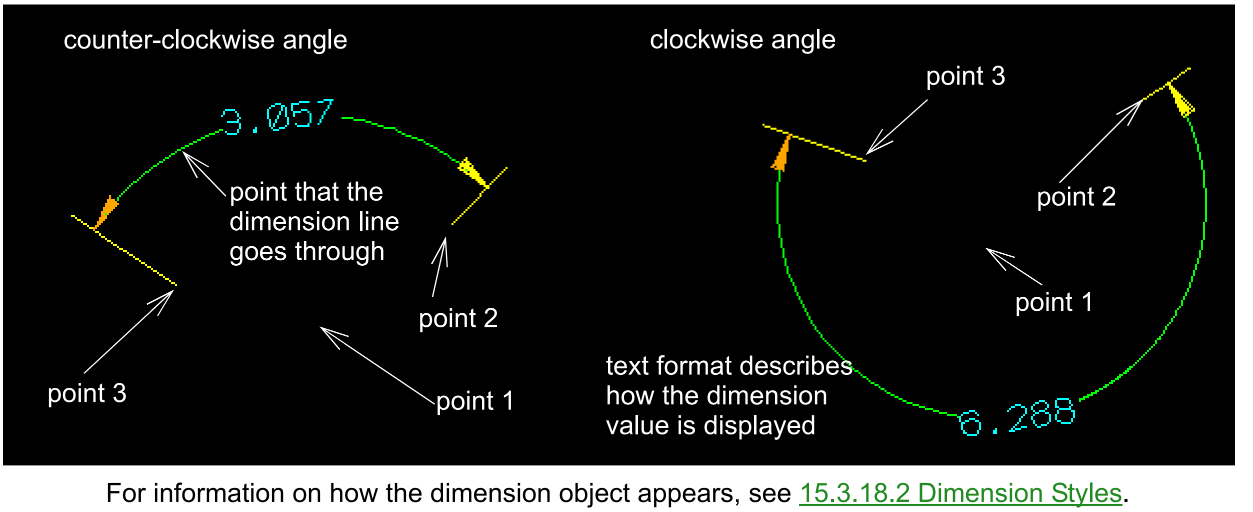




Pick the point that the dimension arc will go through (dimension point).

The dimension arc is then drawn from the start side to the end side and going through the dimension point and labelled with the subtended angle.

The **Arc** option then starts again. That is, goes back to Step 1.

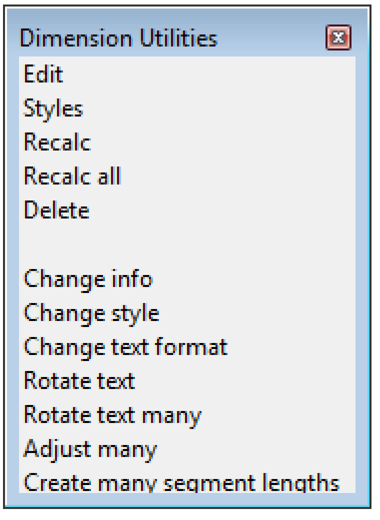


For information on how the dimension object appears, see 15.3.18.2 Dimension Styles.

15.3.18 Dimension Utilities

Position of menu: **Cad =>Dimension =>Utilities**

The **Dimension Utilities** menu is:



edit a dimension
define styles for dimensions

See

15.3.18.1 Dimension Edit
15.3.18.2 Dimension Styles
Recalc - select a dimension to recalc.
Recalc all - recalcs all dimensions.
Delete - select a dimension to delete.
15.3.18.3 Change Style of Dimension
15.3.18.4 Change Text Format
15.3.18.5 Create Many Segment Lengths

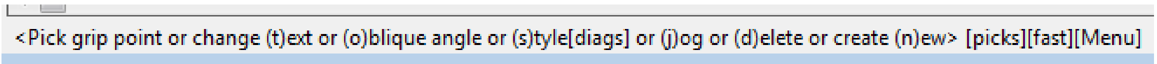
15.3.18.1 Dimension Edit

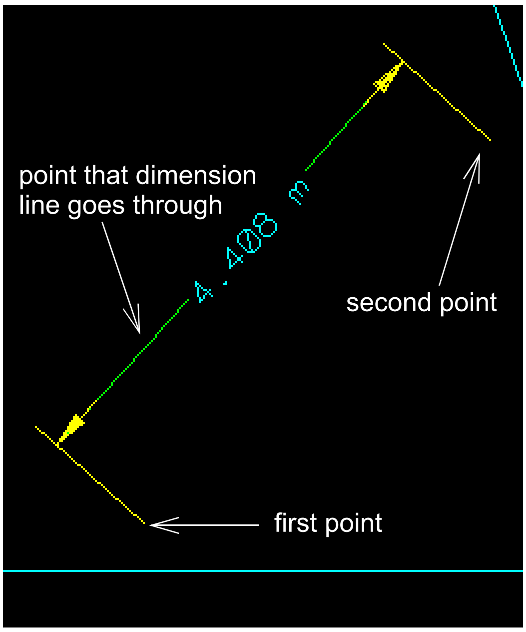
Picking an existing Dimension will bring up the Dimension editor.

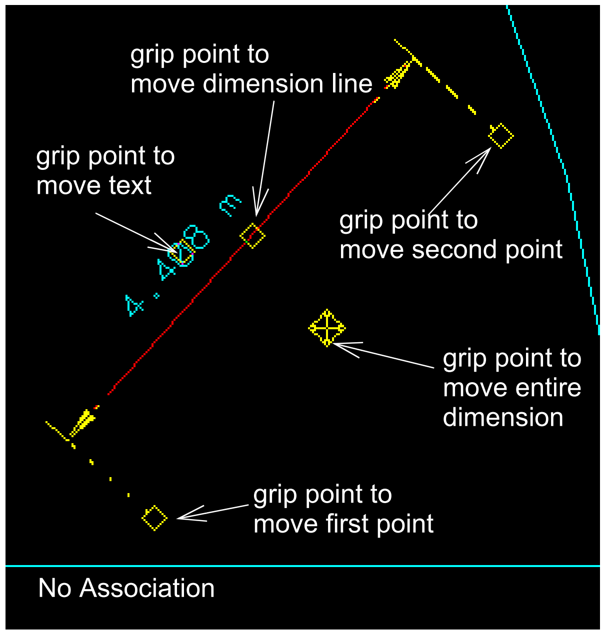
In the Dimension editor, grip points relevant to the type of dimension will be displayed.

If there is **no Association** because either the Association is OFF, or Association is ON but there is nothing currently associated with the dimension, then you won't get a grip point to move the entire dimension because the dimension is locked to its associated items.

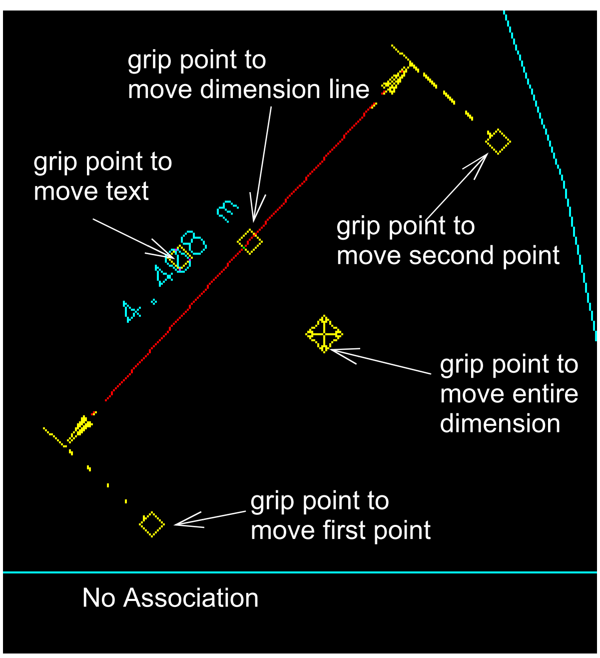
A list of typed options is also displayed in the screen message area.

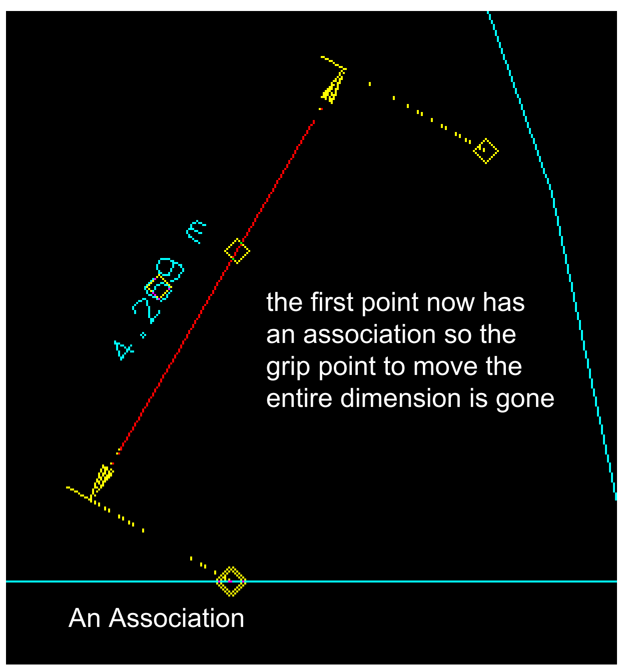






Picking and accepting a grip point will move the grip point and when it is placed again, the association is recalced and the new value for the dimension displayed.





For the typed options, typing

t or **T** changes the dimension text format. See 15.5 Text Format for Dimensions and Leaders

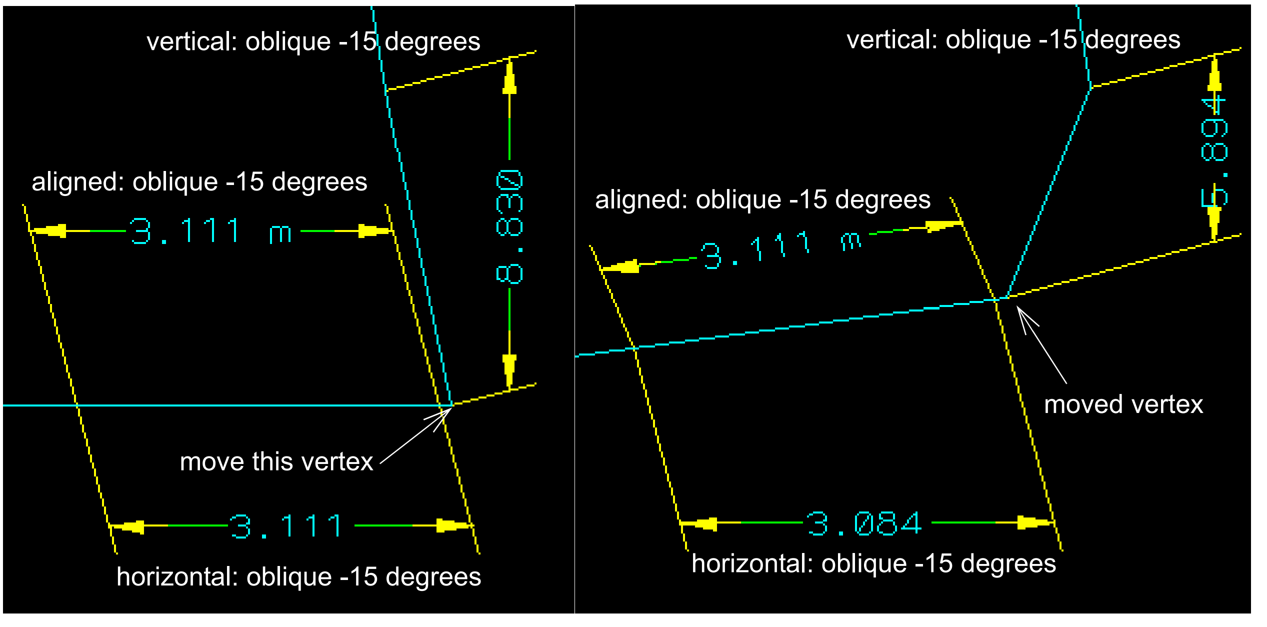
o or **O** changes the angle of the extension lines. The angle is measured in a counter clockwise direction and for horizontal dimensions it is measured from the positive y axis, vertical dimensions it is measured from the positive x axis and for aligned dimensions it is the measured from the perpendicular to the two points defining the dimension.

s or **S** changes the dimension style

j or **J** allows one **jog** to be inserted in a linear dimension

d or **D** deletes the dimension

n or **N** pick a new dimension to edit.



Also

Typing **<Esc>** will exit the editing of the current *dimension* and ask for a new drafting element (Leader or Dimension) to be selected for editing.

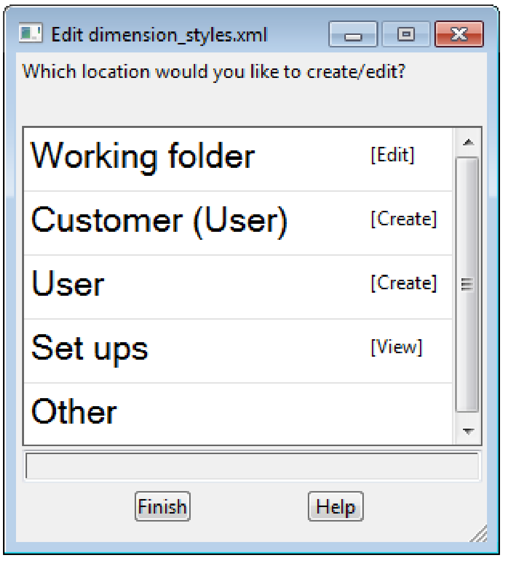
Similarly clicking **RB** and selecting **Cancel** from the **Pick Ops** menu will exit the editing of the current *dimension* and ask for a new drafting element (Leader or Dimension) to be selected for editing.

When back in the Drafting Editor, typing **<Esc>** or clicking **RB** and selecting **Cancel** from the **Pick Ops** menu, will exit the Drafting Editor

15.3.18.2 Dimension Styles

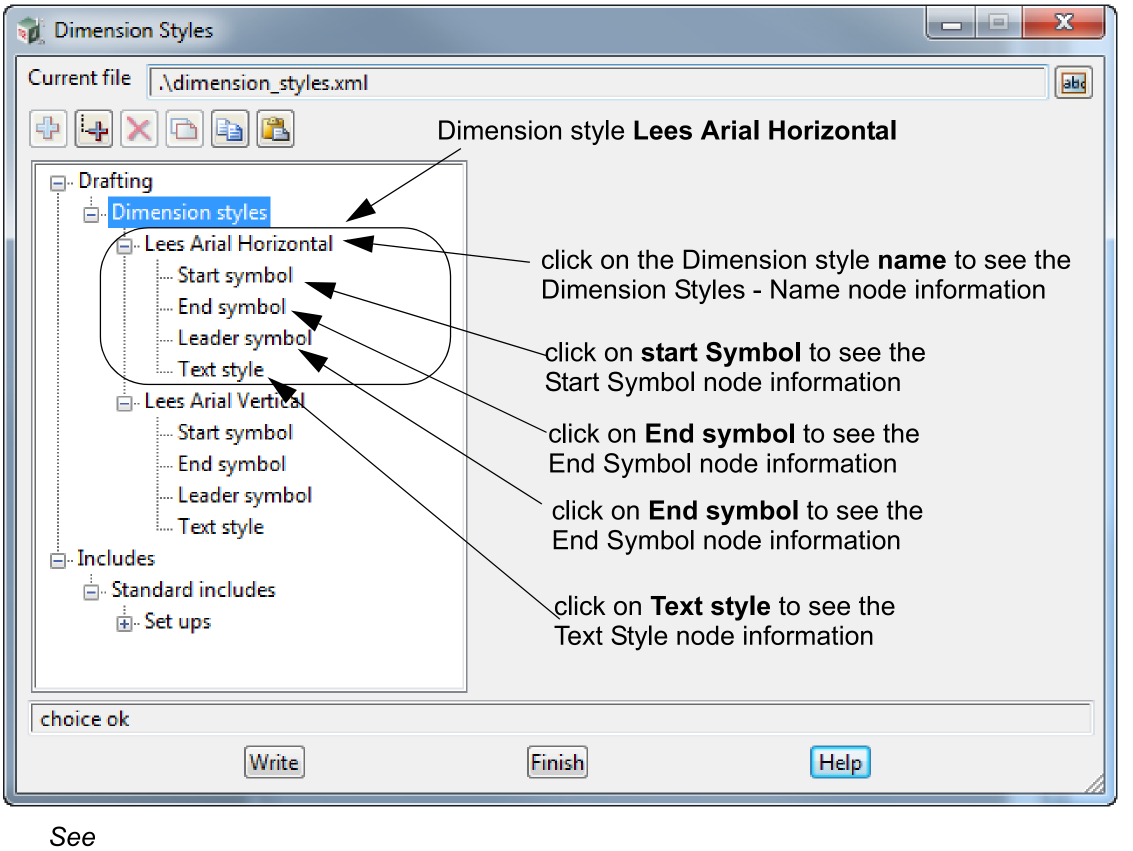
Position of option on menu: **CAD =>Dimension =>Styles**

When you click on the option an **Edit dimension_styles.xml** panel is brought up and the panel shows the standard areas for looking for the table_styles.XML files - Working folder, Customer (User), User, Set Ups and Other.



For information about where how to find and create dimension_styles.xml files, see 15.7 Style XML Files for Dimensions, Leaders & Tables.

Once a dimension_styles.xml file has been opened, the **Dimension Styles** panel is displayed.



See

15.3.18.2.1 Dimension Styles - Name Node

15.3.18.2.1 Dimension Styles - Name Node

This is on the right hand side of the panel when you click on the ***Dimension style*** name.

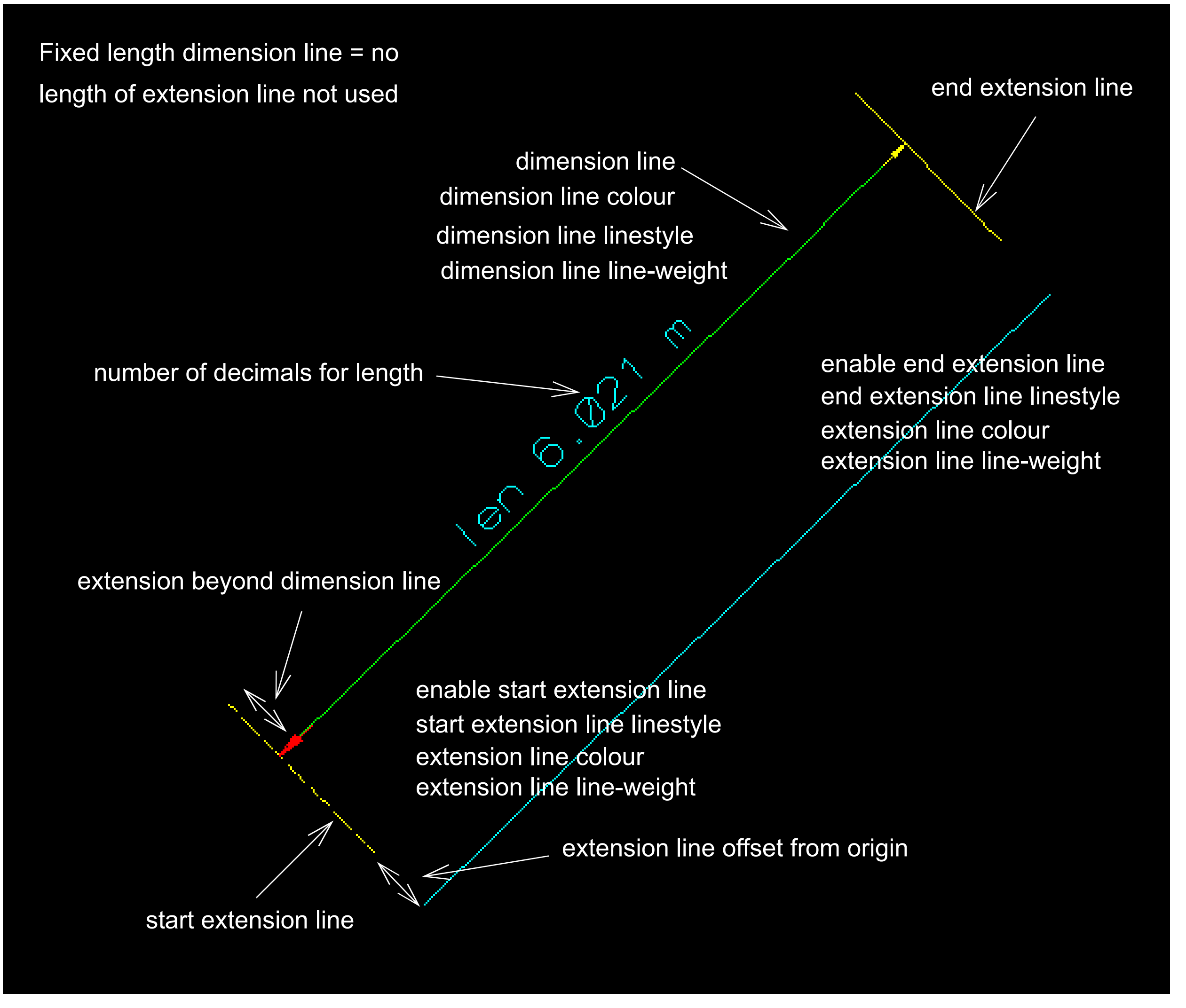
Name	d3
Description	test
Dimension line colour	green
Dimension line linestyle	THICK
Dimension line line-weight	0
Extension line colour	blue
Start extension line linestyle	BORDER
End extension line linestyle	BORDER
Extension line line-weight	0
Enable start extension line	yes
Enable end extension line	yes
Extension beyond dimension line	0.2
Extension line offset from origin	0.05
Fixed length extension line	no
Length of extension line	1
Centre mark style	centre_mark
Centre mark size	0.1
Radius jog angle	0.888
Radius jog factor	1.6
Text alignment	text_aligned
Forced text up	yes
Angle for reading text	0
Do not clip dimension line for text	no
Use box around text	yes
Use rounded box	yes
Text box colour	blue
Text box linestyle	THIN
Number of decimals for length	3
Number of decimals for area	2
Number of decimal for angle	2
Dimension line extension size	0
Baseline spacing	1
Enable start dimension line	yes
Enable end dimension line	yes
Dimension line break	0
Length symbol style	length_preceding
Text fit	fit_either
Text offset	0
Number of decimals for volume	2
Angle format style	show_angle

only used with some dimensions

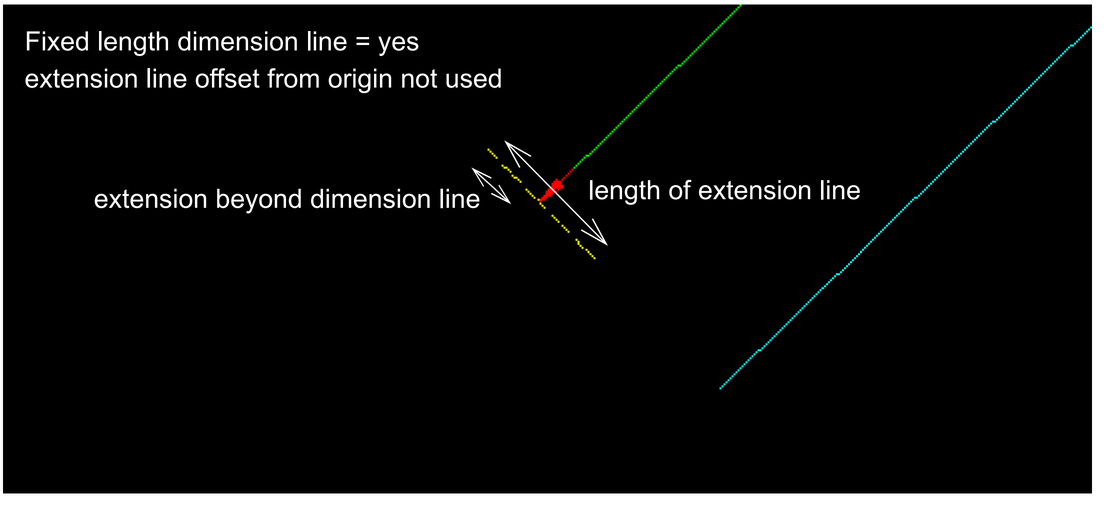
if *Do not clip dimension line for text* is **yes** or Textstyle Type is not world then these are not used

if *Do not clip dimension line for text* is **no** and Textstyle Type is world, then these are used

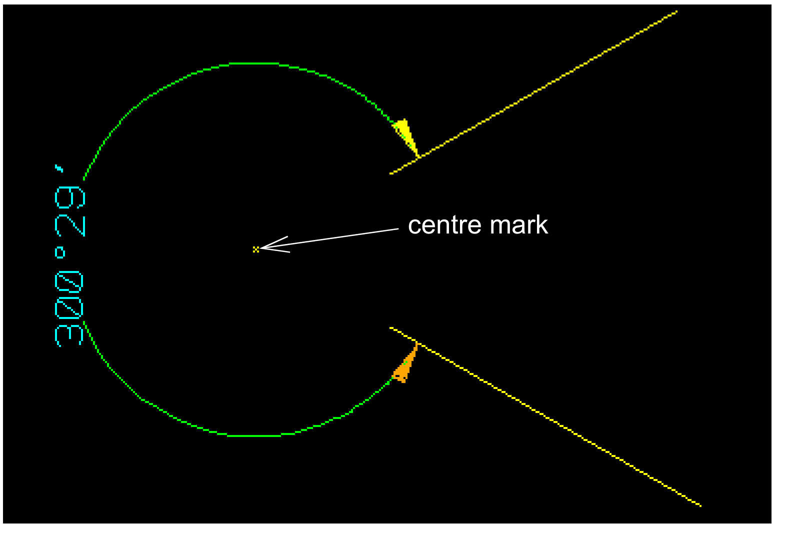
these settings are not currently used



Fixed length dimension line = no
length of extension line not used
end extension line
dimension line
dimension line colour
dimension line linestyle
dimension line line-weight
number of decimals for length
len 6.021 m
enable end extension line
end extension line linestyle
extension line colour
extension line line-weight
extension beyond dimension line
enable start extension line
start extension line linestyle
extension line colour
extension line line-weight
extension line offset from origin
start extension line



Fixed length dimension line = yes
extension line offset from origin not used
extension beyond dimension line
length of extension line



If **Don't clip dimension line for tex**t is set to **no,** or the **Type** from the **Textstyle** tab is not **world,** then the following are not used.

Use box around text
Use rounded box
Text box colour
Text box linestyle

The following items that are at the bottom of the tab have not yet been implemented.

Dimension line extension size
Baseline spacing
Enable start dimension line
Enable end dimension line
Dimension line break
Length symbol style
Text fit
Text offset
Number of decimals for volumes
Angle format style

15.3.18.2.2 Dimension Styles - Start Symbol Node

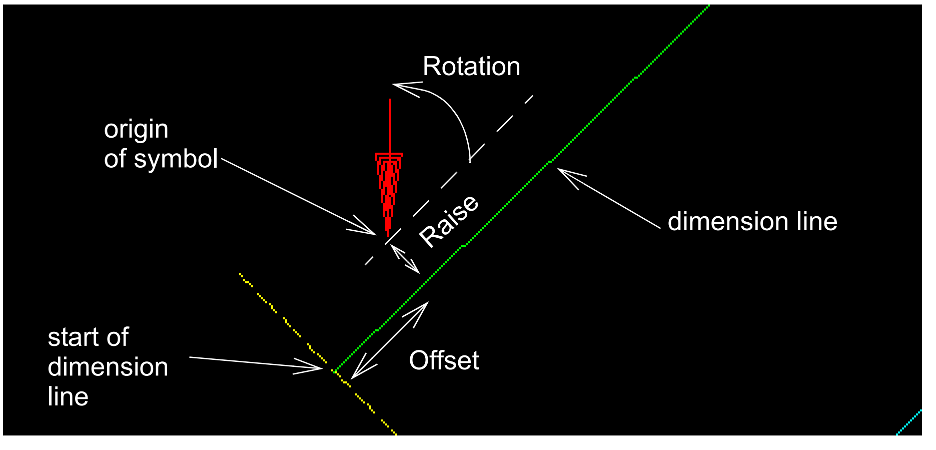
This is on the right hand side of the panel when you click on the ***Start symbol*** node.

Style	Arrow
Colour	red
Size	0.5
Rotation	45
Offset	0.2
Raise	0.5

For the **Start symbol**, the origin of the symbol is displaced from the **start** of the dimension line by the **Offset** and **Raise** values.

With a **Rotation** value of zero, the symbol is given the same rotation as the dimension line.

For a non zero **Rotation**, the value is **added** to the angle of the dimension line.



15.3.18.2.3 Dimension Styles - End Symbol Node

This is on the right hand side of the panel when you click on the ***End symbol*** node.

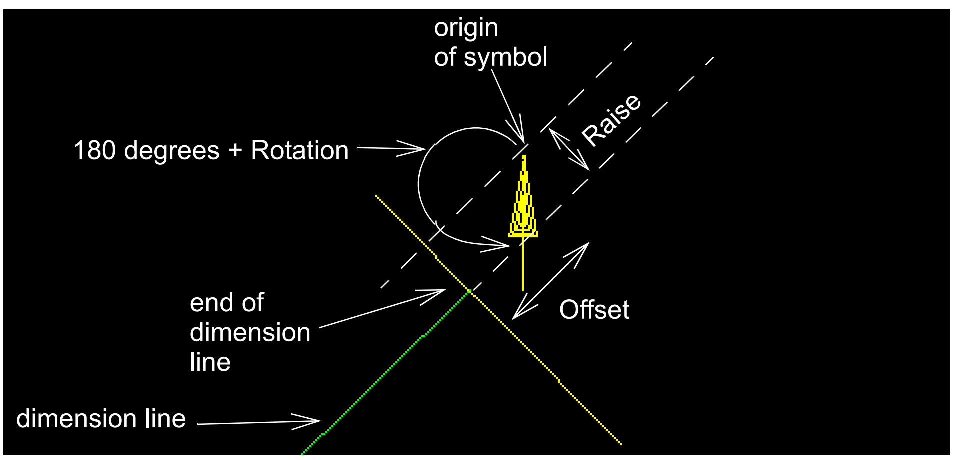
Style	ARROW
Colour	yellow
Size	0.5
Rotation	45
Offset	0.2
Raise	0.5

For the **End symbol**, the origin of the symbol is displaced from the **end** of the dimension line by the **Offset** and **Raise** values.

With a **Rotation** value of zero, the symbol is given the angle of the dimension line **plus 180 degrees**.

For a non zero **Rotation**, the value **plus 180 degrees** is **added** to the angle of the dimension line.

The addition of 180 degrees for the angle of the end symbol is so that the same symbol can be used for the start and the end of the dimension line.



15.3.18.2.4 Dimension Styles - Text style Node

This is on the right hand side of the panel when you click on the ***Text style*** node.

Textstyle	Arial
Name	
Type	paper
Size	0.2
Angle	0°
X-factor	1
Slant	0
Offset	0
Raise	0
Colour	cyan
Whiteout	no colour
Border	no colour
Border type	
Justify	bottom-left
TTF Underline	no
TTF Strikeout	no
TTF Italic	no
TTF Outline	no
TTF Weight	400

This is expected to be modified shortly so that it uses a *Textstyle Data* and all the features from *Textstyle Data* are automatically available. For the moment with paper text, only **Whiteout** is supported.

When the *Textstyle Data* is implemented, it can have its own *Border type* but there is no linestyle for the border in *Textstyle Data*. The only way to get a border with a linestyle is to use World text and set the border parameters in the *General* tab (15.3.18.2.1 Dimension Styles - Name Node).

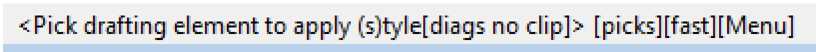
If **Type** is not ***world***, then **Don't clip dimension line for tex**t is set to **no** and these items from the General tab are not used.

Use box around text
Use rounded box
Text box colour
Text box linestyle

See 15.3.18.2.1 Dimension Styles - Name Node

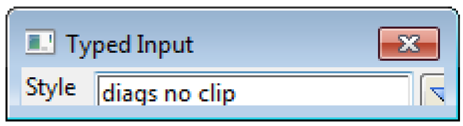
15.3.18.3 Change Style of Dimension

Click on **Change style of dimension** and the following message is written to the screen message area



Pick a dimension and its style will be changed to that shown in the square brackets ([]). Then pick another dimension to change its style.

Typing **s** or **S** brings up the *Style* **Typed Input** box to select a new dimension style.



The new style is picked from the pop up list, or typed into the box and <Enter> pressed.

The box is then removed and the message again written to the screen message area.

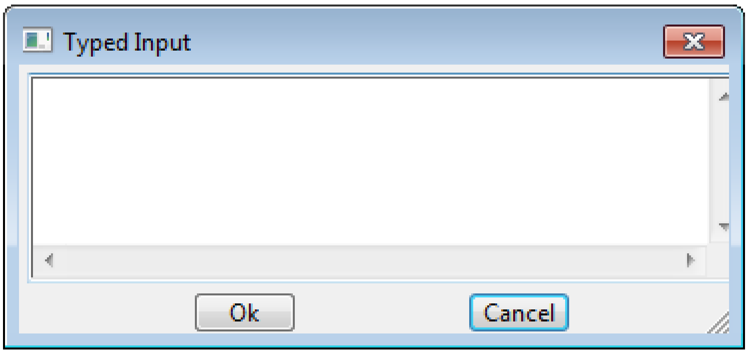
All subsequent selected dimensions will have their styles changed to this new style until either the option is terminated or a new dimension style is selected

To terminate the option, press **<Esc>**, select **Cancel** from the **Pick Ops** menu, or select another CAD option.

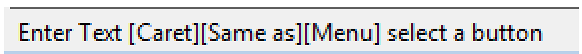
15.3.18.4 Change Text Format

The **Change text format** option is used the change the text format for **Dimensions** or **Leaders**.

Click on **Change text format** and a **Typed Input** box comes up

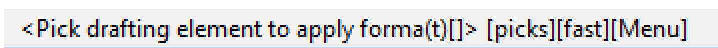


and whilst you are in the **Typed Input** box the following message is written to the screen message area.



The new text format is typed into the **Typed Input** box and **OK** clicked.

The following message is written to the screen message area



Pick a **Dimension** or **Leader** and its text format will be changed to that shown in the square brackets ([]). Then pick another **Dimension** or **Leader** to change its text format

Typing **t** or **T** will bring the **Typed Input** box to type in a a new text format. See 15.5 Text Format for Dimensions and Leaders

All subsequent selected **Dimensions** or **Leaders** will have their text formats changed to this new text format until either the option is terminated or a new text format is selected

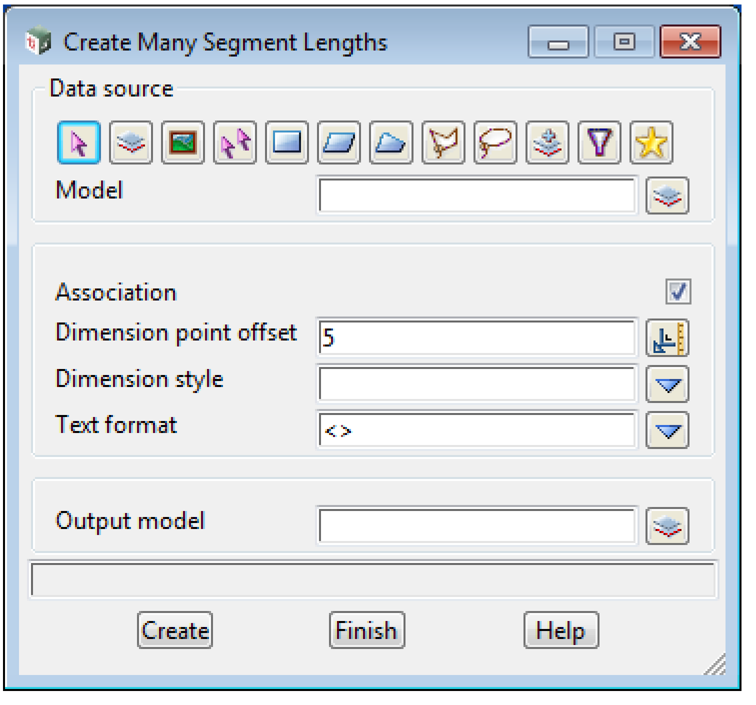
To terminate the option, press **<Esc>**, select **Cancel** from the **Pick Ops** menu, or select another CAD option.

15.3.18.5 Create Many Segment Lengths

This option creates Length dimensions for all the segments of the strings in the Data Source.

The created **Length** Dimensions all have the given **Offset** and **Text format**.

Selecting the **Create many segment lengths** brings up the **Create Many Segment Lengths** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source *in the chapter* 4 Tools and Concepts.

Data source	input		

source of data to have labelled with Length dimensions.

Association	tick box	tick	

*if **ticked**, the Length dimensions are created with Association turned ON.*
*If **not ticked,** the Length dimensions are created with Association turned OFF.*

Dimension point offset	real box	5	

offset to use with the Length dimension. This will override the value from the **Style**.

Dimension Style	dimension style box		

the dimension style to use for the Length dimensions.

Text format	choice		Dim Length text format choice

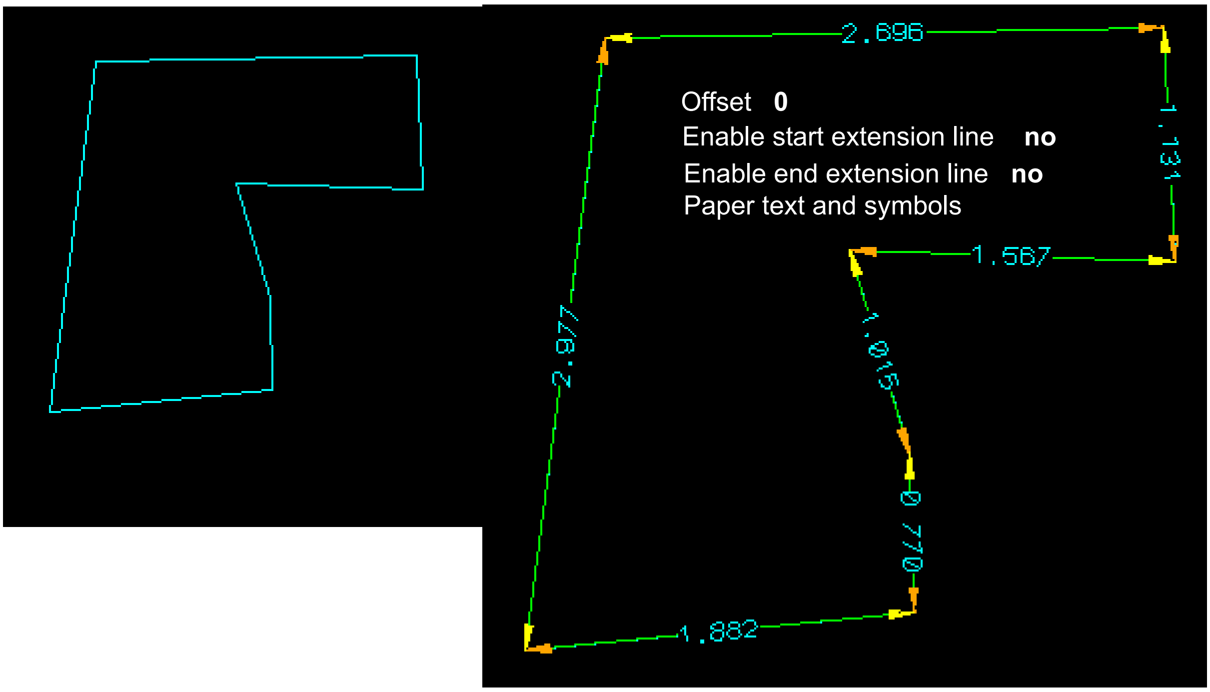
the text format to use for the Length dimensions.

Output model	model box		available models

model to add all the created Length dimensions to.

Create	button		

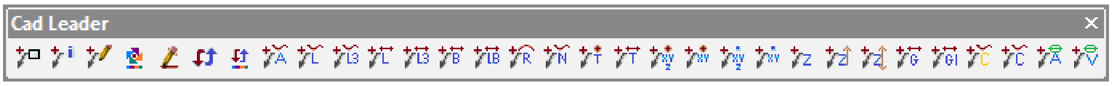
create Length dimensions for all the segments in the data source.

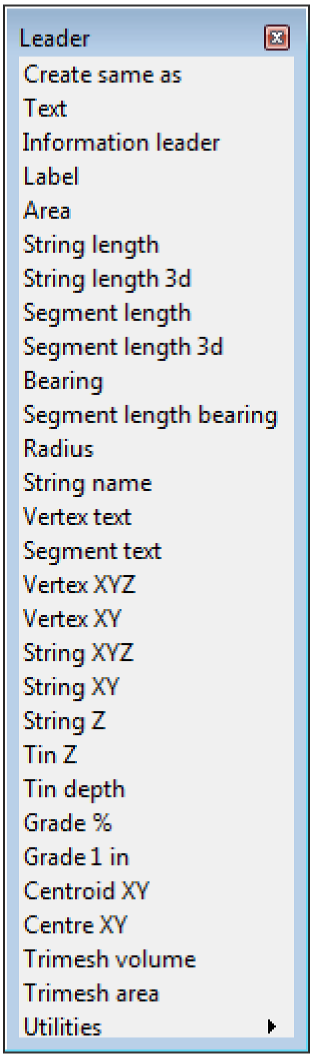

2.696
Offset 0
Enable start extension line no
Enable end extension line no
Paper text and symbols
1.131
1.567
2.877
1.015
0.770
1.882

15.4 CAD Leader

Position of option on menu: **CAD =>Leader**

Various types of **Leaders** can be created with the **Leader** options





Apart from the Trimesh Volume and Trimesh area leaders, each type of Leader can remember the items it was created from (Association). In most cases when the items are modified, the Leader and the associated values from the items dynamically change. See 15.4.1 Leader Association

All of the **Leaders** on the menu except for **Trimesh Volume** and **Trimesh Area** are also special cases of the **Information Leader**. That is, when in the **Information Leader** option, all but the Trimesh Leaders can be selected as types of the **Information Leader**.

For more information on each of the **Leader** options and menus, see

Create same as - create new information leader of same type as selected Information leader 15.4.3 Leader - Create Same As

Text -simple leader that only displays text. *See* 15.4.4 Text Leader

Information Leader - leader with many choices of values. *See* 15.4.5 Information Leader

Label -a case of Information Leader that displays text. *See* 15.4.5.1 Label

Area - 2D area of a selected string. *See* 15.4.5.2 Area

String length - 2D length of a selected string. *See* 15.4.5.3 String Length

String length 3d - 3D length of a selected string. *See* 15.4.5.4 String Length 3D

Segment length - 2D length of a selected segment. *See* 15.4.5.5 Segment Length

Segment length 3d - 3D length of a selected segment. *See* 15.4.5.6 Segment Length 3D

Bearing - bearing of a selected string. *See* 15.4.5.7 Segment Bearing

Bearing & segment length - bearing at string pick point and length of the segment. *See* 15.4.5.8 Bearing & Segment Length

Radius - radius of a selected string. *See* 15.4.5.9 Segment Radius

String name - name of a selected string. *See* 15.4.5.10 String name

Vertex text - vertex text of a selected vertex. *See* 15.4.5.11 Vertex Text

Segment text - segment text of a selected segment. *See* 15.4.5.12 Segment Text

Vertex XYZ - (x,y,z) coordinates of a vertex. *See* 15.4.5.13 Vertex XYZ

Vertex XY - (x,y) coordinates of a vertex. *See* 15.4.5.14 Vertex XY

String XYZ - (x,y,z) coordinates of a position on a string. *See* 15.4.5.15 String XYZ

String XY - (x,y) coordinates of a position on a string *See* 15.4.5.16 String XY

String Z - z coordinates (level) of a position on a string *See* 15.4.5.17 String Z

Tin Z - z value of tin at a selected (x,y) position. *See* 15.4.5.18 Tin Z

Tin depth - depth from a string position to a tin. *See* 15.4.5.19 Tin Depth

Grade % - grade in percent at a point on a string. *See* 15.4.5.20 Grade (%)

Grade 1 in - grade as 1 in (ie slope) at a point on a string. *See* 15.4.5.21 Grade 1 n - Slope

Centroid XY - (x,y) coordinates of selected string. *See* 15.4.5.22 Centroid XY

Centre XY - (x,y) coordinates of medial centre of selected string. *See* 15.4.5.23 Medial Centre XY

Trimesh volume - volume of a closed trimesh. *See* 15.4.5.24 Trimesh Volume

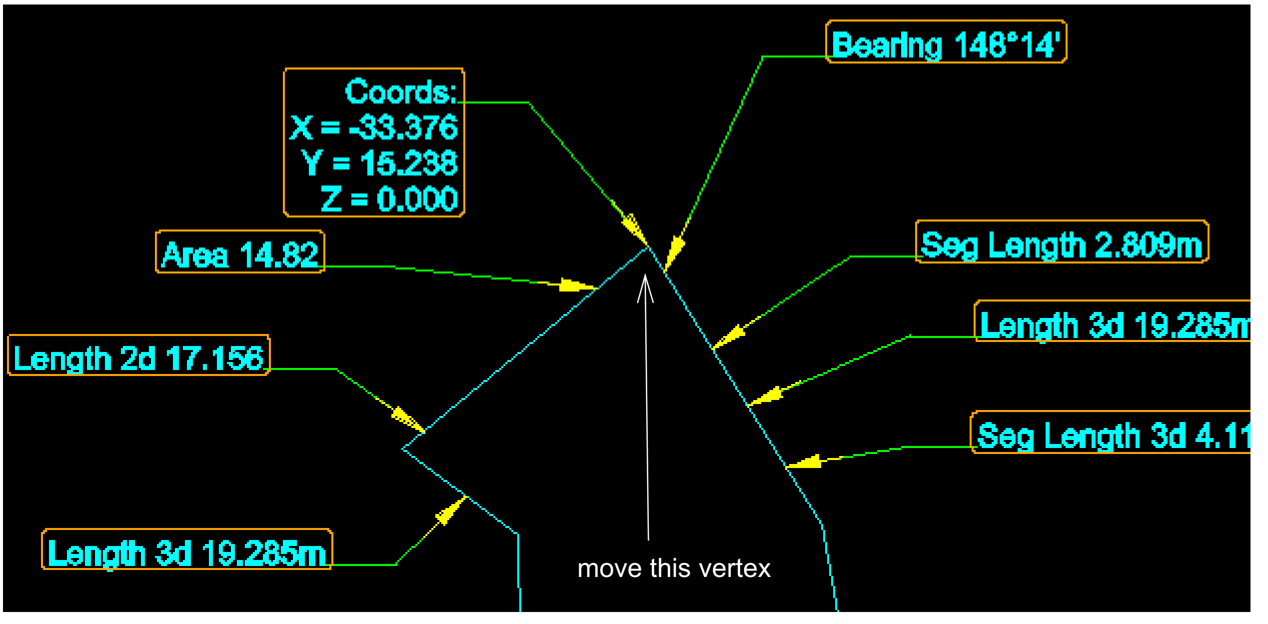
Trimesh area - surface area of one side of a trimesh. *See* 15.4.5.25 Trimesh Surface Area

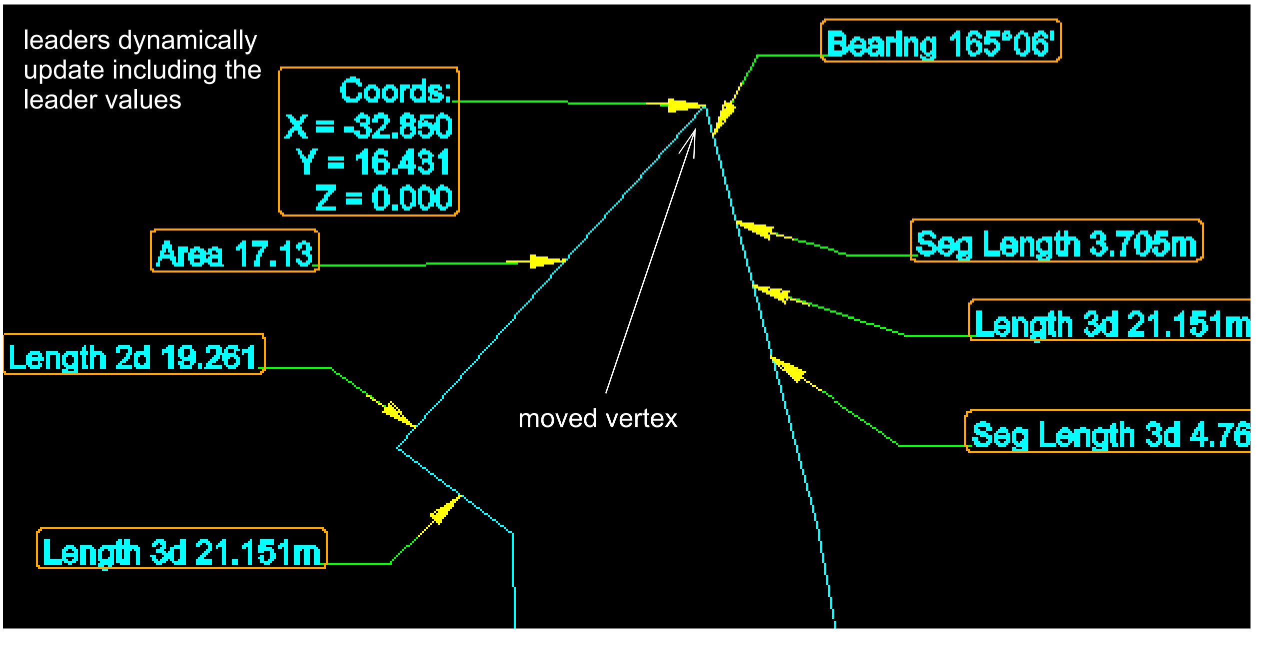
Utilities - various leader utilities. *See* 15.4.6 Leader Utilities

15.4.1 Leader Association

When *leaders* are created, the leader can remember the items that were used to create them (**Association**).

In most cases when items are moved, the leaders associated with the items dynamically change.



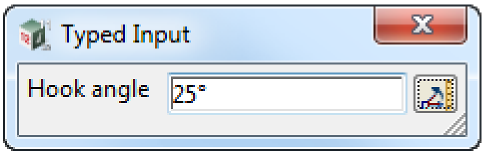


In cases where it is not automatic, there is a **Recalc** and **Recalc all** for leaders and dimensions.

In the leader options, typing **a** or **A** toggles Association **ON** and **OFF**.

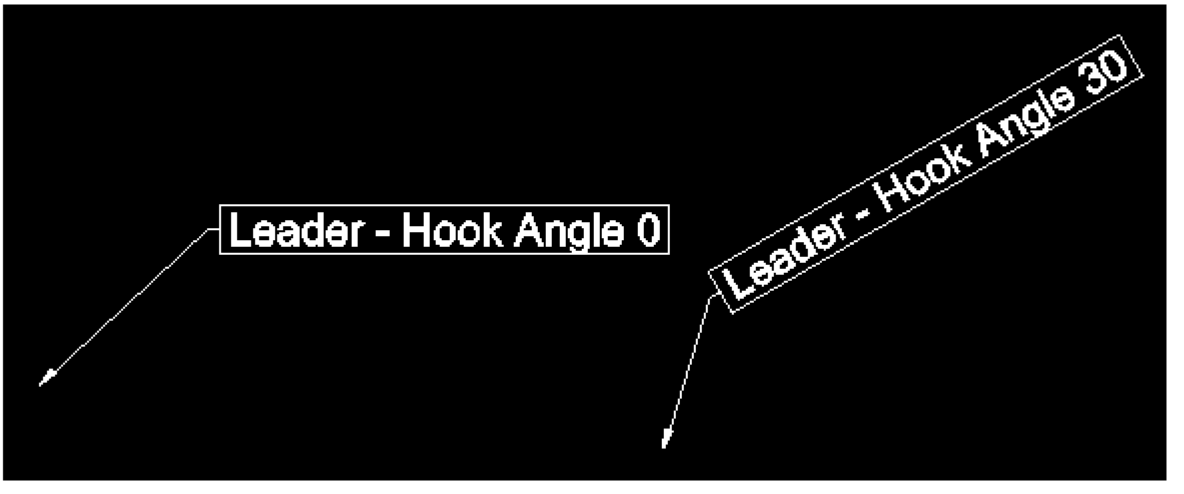
15.4.2 Hook Angle

Typing **h** or **H** allows you to change the **Hook Angle** and brings up the *Hook angle* **Typed Input** box.



The new hook angle is typed into the box and **<Enter>** pressed.

The units for hook angle is degrees entered in hp notation, and is measure in a counter clockwise direction from the positive x-axis.



15.4.3 Leader - Create Same As

Create same as creates a new Leader of the same type as a selected Leader.

Step 1.

Select **Create same as** and the following message is written to the screen message area

<[Pick leader]> [picks][fast][Menu]

Step 2.Select an existing Leader

When an existing **Leader** is selected, a new **create Leader** of the same Leader type as the selected Leader, is started.

15.4.4 Text Leader

This is simplest type of Leader and only creates a leader with user defined text.

Text Leader creates a leader line from a selected point (the arrow point) to a selected hook point, with user given text drawn at the end of the leader line.

a or **A** toggles Association **ON** and **OFF**. See 15.4.1 Leader Association
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle

Step 1.Pick the arrow point (the point to start the Leader from)

Select Leader and the following message is written to the screen message area

<[Pick arrow point] (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] > [picks][fast][I

If Association is **ON** then the start of the Leader is associated with the selected arrow point.

Step 2.Pick the hook point

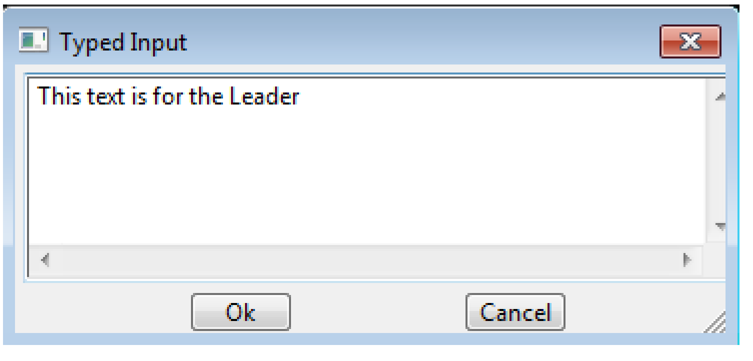
The following message is written to the screen message area.

<Pick hook point> [picks][fast][Menu]

If Association is **ON** then the hook point of the leader is associated with the selected hook point.

Step 3.

The text **Typed Input** box then comes up for the text for the leader to be typed into.



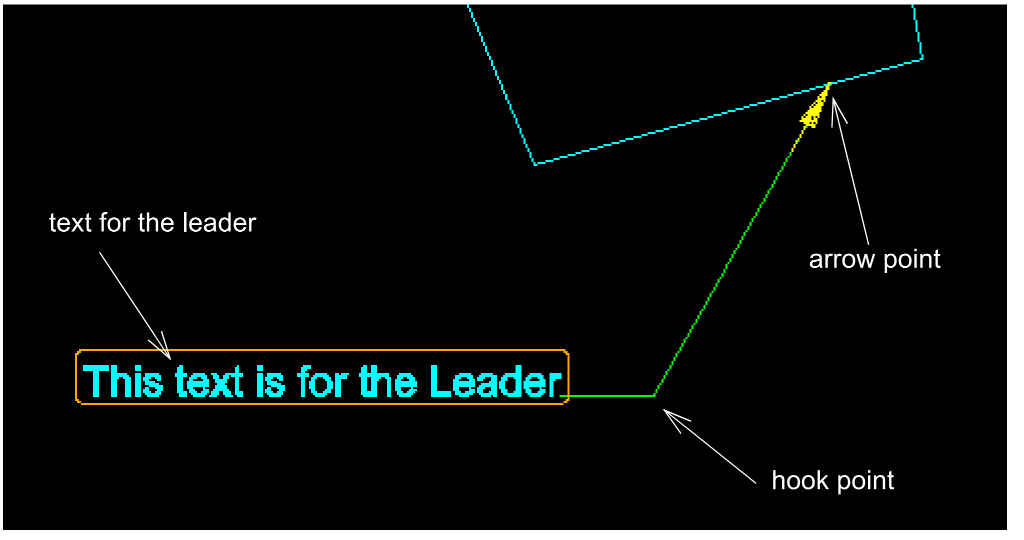
Whilst you are in the **Typed Input** box the following message is written to the screen message area.

Enter Text [Caret][Same as][Menu] select a button

The text for the Leader is typed into the **Typed Input** box and **OK** clicked.

The *Leader* is then created with the current hook angle.

The **Leader** option then starts again. That is, goes back to **Step 1**.



For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

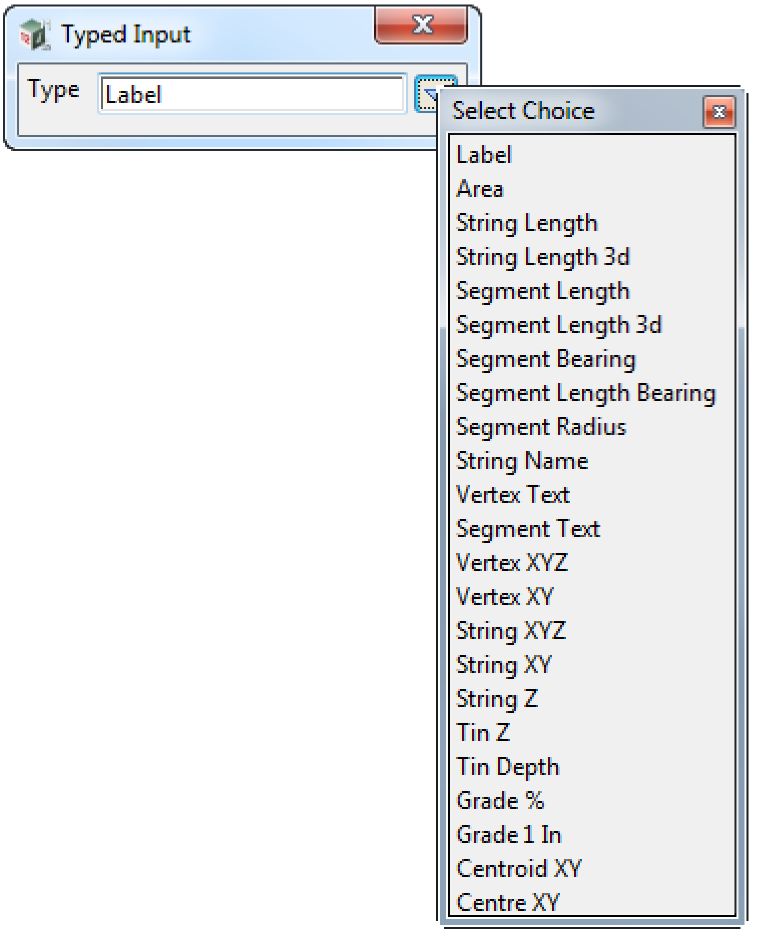
15.4.5 Information Leader

Information Leader creates a leader line from a selected point (the **arrow point)** to a selected **hook point**, with a user selected choice of what information is shown at the end of the leader line.

When **Information Leader** is selected, the following message is written to the screen message area.

<(a)ssociative (t)ype[Label] Pick arrow location (s)tyle[Arial Border] (h)ook-angle[25°] > [picks][fast][Menu]

Typing **t** or **T** brings up the **Information Leader** *Type* **Typed Input** box which has a choice pop up for selecting what information is drawn at the end of the leader.



After a selection is made, the message changes to indicate the current type of **Information Leader** being created. For example, for the choice **Area**

type of Information Leader being created

< (t)ype[Area] Pick string (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] fixed-a(r)row> [picks][fast][Menu]

For more information on each of the types for Information Leader see

Label -simple leader that only displays text. *See* 15.4.5.1 Label

Area - 2D area of a selected string. *See* 15.4.5.2 Area

String length - 2D length of a selected string. *See* 15.4.5.3 String Length

String length 3d - 3D length of a selected string. *See* 15.4.5.4 String Length 3D

Segment length - 2D length of a selected segment. *See* 15.4.5.5 Segment Length

Segment length 3d - 3D length of a selected segment. *See* 15.4.5.6 Segment Length 3D

Bearing - bearing of a selected string. *See* 15.4.5.7 Segment Bearing

Bearing & segment length - bearing at string pick point and length of the segment. *See* 15.4.5.8 Bearing & Segment Length

Radius - radius of a selected string. *See* 15.4.5.9 Segment Radius

String name - name of a selected string. *See* 15.4.5.10 String name

Vertex text - vertex text of a selected vertex. *See* 15.4.5.11 Vertex Text

Segment text - segment text of a selected segment. *See* 15.4.5.12 Segment Text

Vertex XYZ- (x,y,z) coordinates of a vertex. *See* 15.4.5.13 Vertex XYZ

Vertex XY - (x,y) coordinates of a vertex. *See* 15.4.5.14 Vertex XY

String XYZ - (x,y,z) coordinates of a position on a string. *See* 15.4.5.15 String XYZ

String XY - (x,y) coordinates of a position on a string *See* 15.4.5.16 String XY

String Z - z coordinates (level) of a position on a string *See* 15.4.5.17 String Z

Tin Z - z value of tin at a selected (x,y) position. *See* 15.4.5.18 Tin Z

Tin depth - depth from a string position to a tin. *See* 15.4.5.19 Tin Depth

Grade % - grade in percent at a point on a string. *See* 15.4.5.20 Grade (%)

Grade 1 in - grade as 1 in (ie slope) at a point on a string. *See* 15.4.5.21 Grade 1 n - Slope

Centroid XY - (x,y) coordinates of selected string. *See* 15.4.5.22 Centroid XY

Centre XY - (x,y) coordinates of medial centre of selected string. *See* 15.4.5.23 Medial Centre XY

15.4.5.1 Label

Information Leader choice **Label** displays in the leader box, user given text.

Label creates a leader line from a selected point (the pick point) to a selected hook point, with the given text drawn at the end of the leader line.

The leader is added to the model given in the CAD toolbar.

Note: Unlike the **Text Leader** which is also used for placing text and brings up a Typed Input box for entering multiline text (see 15.4.4 Text Leader), **Label** is a single line format which needs new line characters if a new line is needed.

The case for using **Label** instead of the **Text** leader is when you have the same text to place multiple times. The **customise (t)ext** command uses the last specified text, and also remembers the last twelve of them.

Step 1.Pick the point to start the Leader from

Select **Leader** and the following message is written to the screen message area

```
< (t)ype[Label] Pick arrow location (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] > [picks][fast][Menu]
```

a or **A** toggles Association **ON** and **OFF**. See 15.4.1 Leader Association
t or **T** changes the type of Information Leader. See 15.4.5 Information Leader
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle

If Association is **ON** then the start of the Leader is associated with the selected pick point.

Step 2.Pick the hook point

The following message is written to the screen message area.

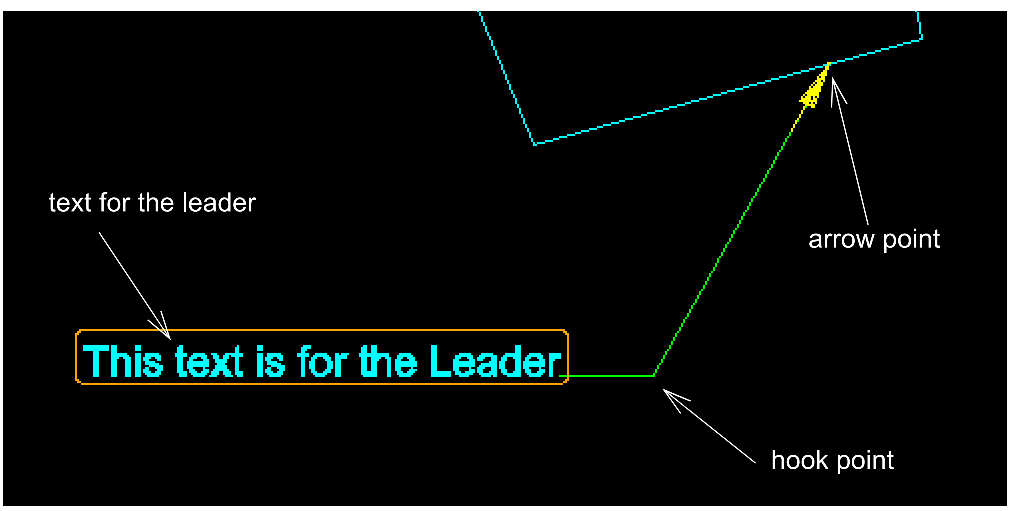
```
<Pick hook point or customise (t)ext> [picks][fast][Menu]
```

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

Unlike the **Text Leader** which brings up a Typed Input box for entering multiline text (see 15.4.4 Text Leader), this is a single line format which needs new line characters if a new line is needed.

When the hook point is selected, the *Leader* is created.

The **Label Information Leader** then starts again. That is, it goes back to **Step 1**.



For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.5.2 Area

Information Leader choice **Area** displays in the leader box, the 2D (plan) area of a selected string. If the selected string is not closed, the area is calculated for the polygon formed by joining the first and the last vertices.

A leader line is drawn from either selected position on a string (string pick point) or the string centre, to a selected hook point, and then to the end of the hook itself.

Using the text format, the value of the area is displayed at the end of the hook line.

The leader is added to the model given in the CAD toolbar.

Step 1.Pick the string to label with its 2D (plan) area

When **Area** is the choice, the following message is written to the screen message area

```
< (t)ype[Area] Pick string (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] fixed-a(r)row> [picks][fast][Menu]
```

a or **A** toggles Association **ON** and **OFF**. See 15.4.1 Leader Association
t or **T** changes the type of Information Leader. See 15.4.5 Information Leader
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle
r or **R** toggles between the arrow starting at the pick point (**free arrow**) or at the polygon centre (**fixed arrow**)

If Association is **ON** then the start of the Leader is associated with the string pick point or the polygon centre (depending on **r**).

Step 2.Pick the hook point

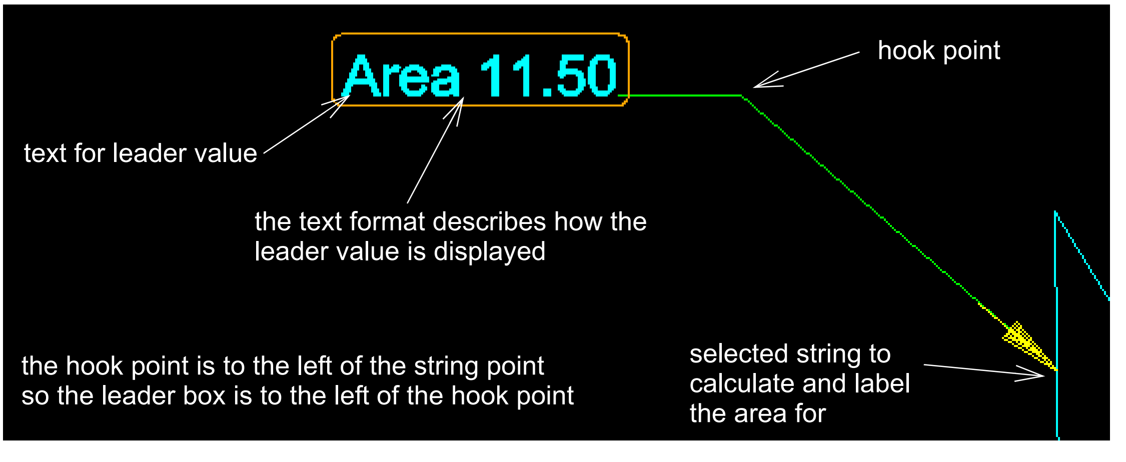
The following message is written to the screen message area.

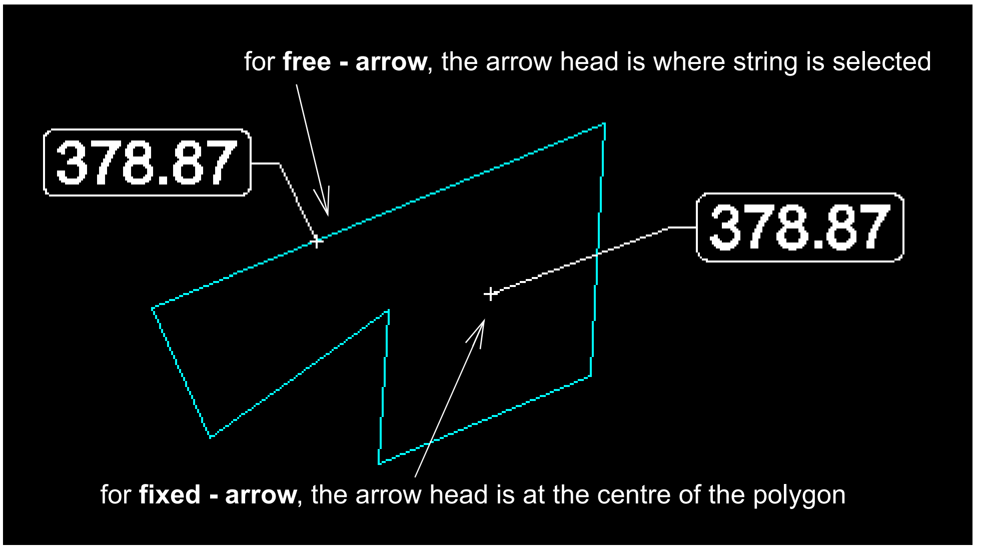
```
<Pick hook point or customise (t)ext> [picks][fast][Menu]
```

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

The *Information Leader* is then created.

The **Area Information Leader** then starts again. That is, it goes back to **Step 1**.





Note on Position of Leader Box With Respect to the Hook Point

As well as determining the length of the leader line, the hook point also determines whether the leader box goes to the left or the right of the hook point.

If the hook point is to the left of the string pick point then the leader box is drawn to the left of the hook point. If the hook point is to the right of the string pick point then the leader box is drawn to the right of the hook point.

For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.5.3 String Length

Information Leader choice **String length** displays in the leader box, the 2D (plan) length of a selected string.

A leader line is drawn from a selected position on a string (string pick point) to a selected hook point, and then to the end of the hook itself.

Using the text format, the value of the string length is displayed at the end of the hook line.

The leader is added to the model given in the CAD toolbar.

Step 1.Pick the string to label with its 2D length

When **String Length** is the choice, the following message is written to the screen message area

```
< (t)ype[String Length] Pick string (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] > [picks][fast][Menu]
```

a or **A** toggles Association **ON** and **OFF**. See 15.4.1 Leader Association
t or **T** changes the type of Information Leader. See 15.4.5 Information Leader
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle

If Association is **ON** then the start of the Leader is associated with the string pick point.

Step 2.Pick the hook point

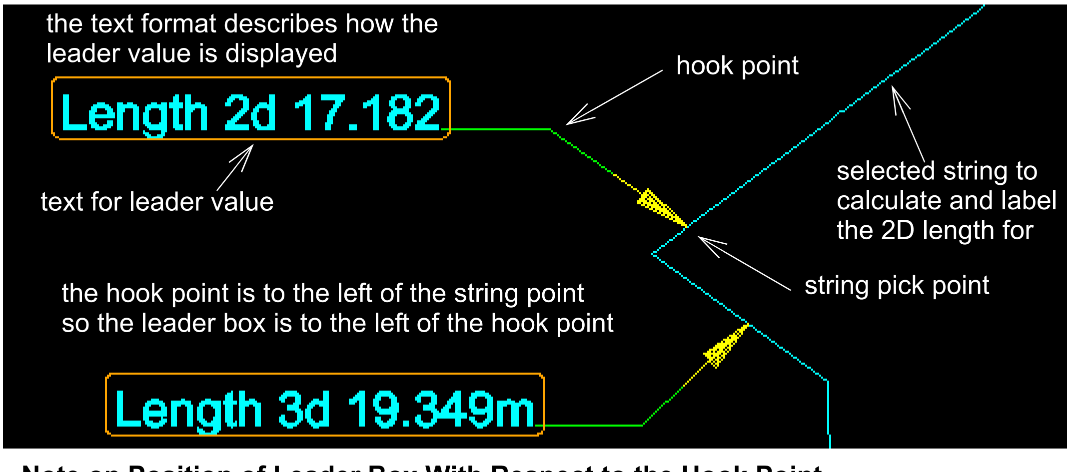
The following message is written to the screen message area.

```
<Pick hook point or customise (t)ext> [picks][fast][Menu]
```

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

The *Information Leader* is then created.

The **String Length Information Leader** then starts again. That is, it goes back to **Step 1**.



Note on Position of Leader Box With Respect to the Hook Point

As well as determining the length of the leader line, the hook point also determines whether the leader box goes to the left or the right of the hook point.

If the hook point is to the left of the string pick point then the leader box is drawn to the left of the hook point. If the hook point is to the right of the string pick point then the leader box is drawn to the right of the hook point.

For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.5.4 String Length 3D

Information Leader choice **String length 3d** displays in the leader box, the 3D length of a selected string.

A leader line is drawn from a selected position on a string (string pick point) to a selected hook point, and then to the end of the hook itself.

Using the text format, the value of the length is displayed at the end of the hook line.

The leader is added to the model given in the CAD toolbar.

Step 1.Pick the string to label with its 3D length

When **String length 3d** is the choice, the following message is written to the screen message area

```
< (t)ype[String Length 3d] Pick string (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] > [picks][fast][Menu]
```

a or **A** toggles Association **ON** and **OFF**. See 15.4.1 Leader Association
t or **T** changes the type of Information Leader. See 15.4.5 Information Leader
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle

If Association is **ON** then the start of the Leader is associated with the string pick point.

Step 2.Pick the hook point

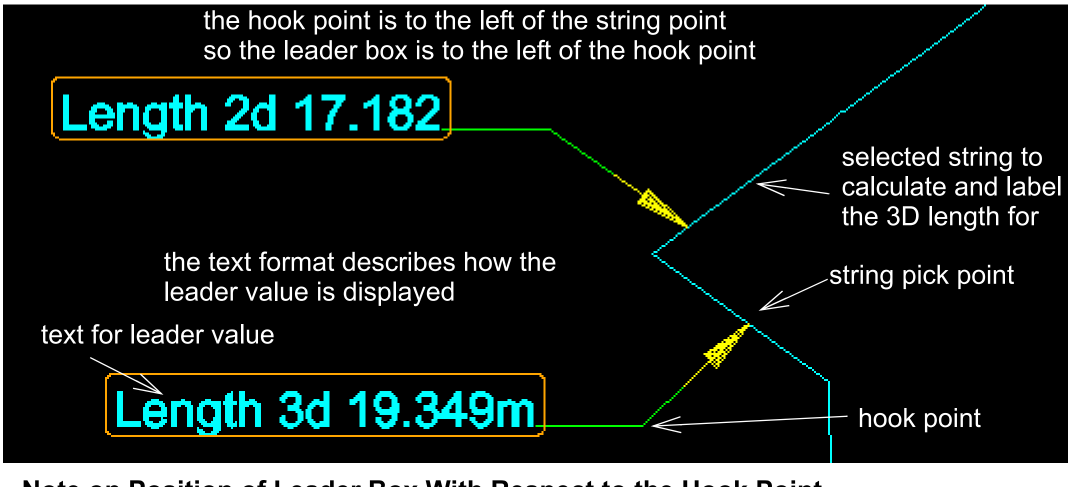
The following message is written to the screen message area.

```
<Pick hook point or customise (t)ext> [picks][fast][Menu]
```

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

The *Information Leader* is then created.

The **String Length 3D Information Leader** then starts again. That is, it goes back to **Step 1**.



Note on Position of Leader Box With Respect to the Hook Point

As well as determining the length of the leader line, the hook point also determines whether the leader box goes to the left or the right of the hook point.

If the hook point is to the left of the string pick point then the leader box is drawn to the left of the hook point. If the hook point is to the right of the string pick point then the leader box is drawn to the right of the hook point.

For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.5.5 Segment Length

Information Leader choice **Segment length** displays in the leader box, the 2D (plan) length of a selected **segment**.

A leader line is drawn from a selected position on a segment (segment pick point) to a selected hook point, and then to the end of the hook itself.

Using the text format, the value of the segment length is displayed at the end of the hook line.

The leader is added to the model given in the CAD toolbar.

Step 1.Pick the segment to label with its 2D length

When **Segment Length** is the choice, the following message is written to the screen message area

< (t)ype[Segment Length] Pick segment (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] > [picks][fast][Menu]

a or **A** toggles Association **ON** and **OFF**. See 15.4.1 Leader Association
t or **T** changes the type of Information Leader. See 15.4.5 Information Leader
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle

If Association is **ON** then the start of the Leader is associated with the segment pick point.

Step 2.Pick the hook point

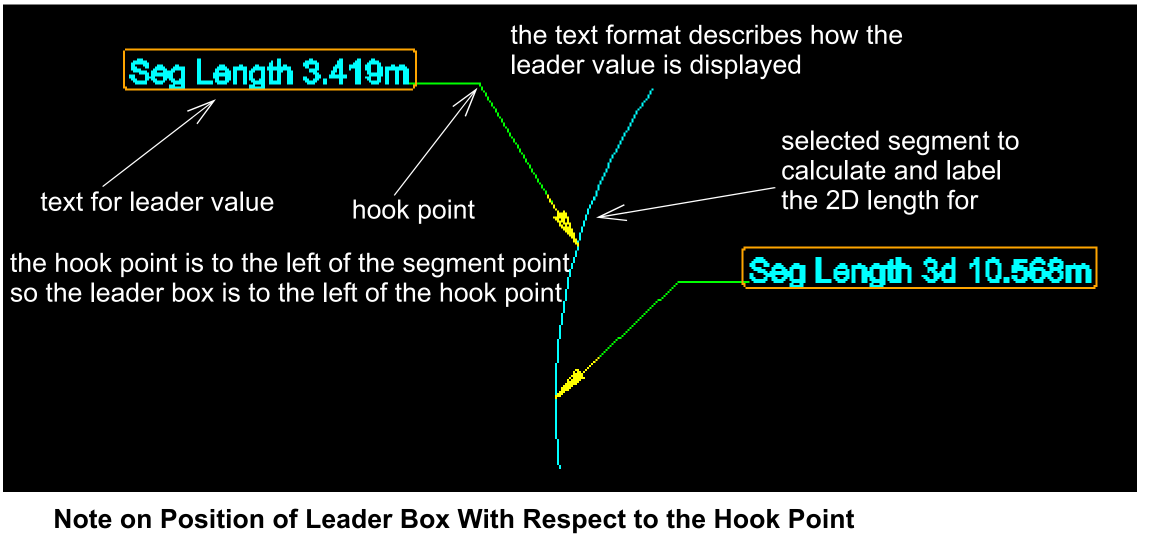
The following message is written to the screen message area.

<Pick hook point or customise (t)ext> [picks][fast][Menu]

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

The *Information Leader* is then created.

The **Segment Length Information Leader** then starts again. That is, it goes back to **Step 1**.



Note on Position of Leader Box With Respect to the Hook Point

As well as determining the length of the leader line, the hook point also determines whether the leader box goes to the left or the right of the hook point.

If the hook point is to the left of the segment pick point then the leader box is drawn to the left of the hook point. If the hook point is to the right of the segment pick point then the leader box is drawn to the right of the hook point.

For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.5.6 Segment Length 3D

Information Leader choice **Segment length 3d** displays in the leader box, the 3D length of a selected segment.

A leader line is drawn from the pick position on a selected segment (segment pick point) to a selected hook point, and then to the end of the hook itself.

Using the text format, the value of the 3d length of the segment is displayed at the end of the hook line.

The leader is added to the model given in the CAD toolbar.

Step 1.Pick the segment to label with its 3D length

When **Segment Length 3d** is the choice, the following message is written to the screen message area

< (t)ype[Segment Length 3d] Pick segment (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] > [picks][fast][Menu]

a or **A** toggles Association **ON** and **OFF**. See 15.4.1 Leader Association
t or **T** changes the type of Information Leader. See 15.4.5 Information Leader
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle

If Association is **ON** then the start of the Leader is associated with the segment pick point

Step 2.Pick the hook point

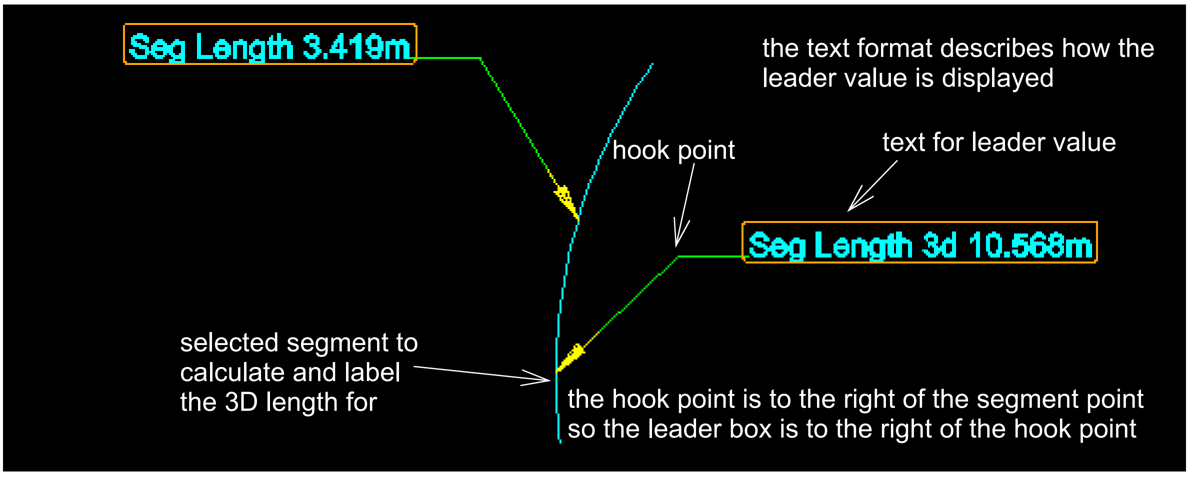
The following message is written to the screen message area.

<Pick hook point or customise (t)ext> [picks][fast][Menu]

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

The *Information Leader* is then created.

The **Segment Length 3D Information Leader** then starts again. That is, it goes back to **Step 1**.



Note on Position of Leader Box With Respect to the Hook Point

As well as determining the length of the leader line, the hook point also determines whether the leader box goes to the left or the right of the hook point.

If the hook point is to the left of the segment pick point then the leader box is drawn to the left of the hook point. If the hook point is to the right of the segment pick point then the leader box is drawn to the right of the hook point.

For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.5.7 Segment Bearing

Information Leader choice **Segment Bearing** displays in the leader box, the bearing of a selected position on string (string pick point).

A leader line is drawn from the string pick point to a selected hook point, and then to the end of the hook itself.

Using the text format, the value of the bearing at the string pick point is displayed at the end of the hook line.

The leader is added to the model given in the CAD toolbar.

Step 1.Pick the segment to label with its bearing

When **Segment Bearing** is the choice, the following message is written to the screen message area

```
< (t)ype[Segment Bearing] Pick point on segment (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] > [picks][fast][Menu]
```

a or **A** toggles Association **ON** and **OFF**. See 15.4.1 Leader Association
t or **T** changes the type of Information Leader. See 15.4.5 Information Leader
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle

If Association is **ON** then the start of the Leader is associated with the string pick point.

Step 2.Pick the hook point

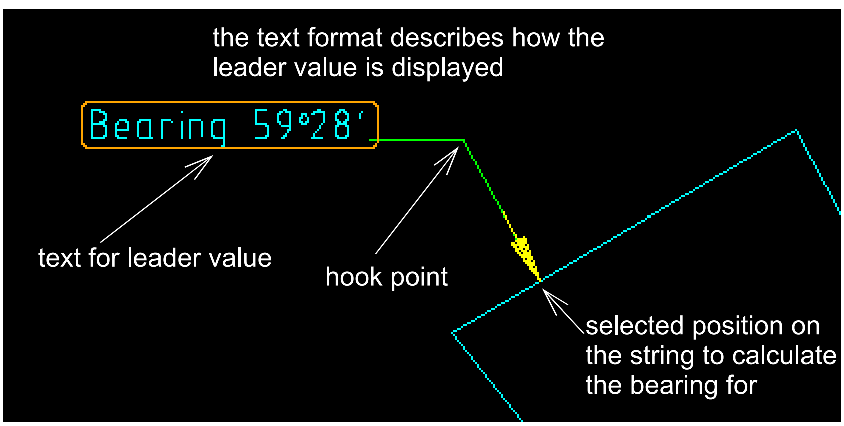
The following message is written to the screen message area.

```
<Pick hook point or customise (t)ext> [picks][fast][Menu]
```

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

The *Information Leader* is then created.

The **Segment Bearing Information Leader** then starts again. That is, it goes back to **Step 1**.



Note on Position of Leader Box With Respect to the Hook Point

As well as determining the length of the leader line, the hook point also determines whether the leader box goes to the left or the right of the hook point.

If the hook point is to the left of the string pick point then the leader box is drawn to the left of the hook point. If the hook point is to the right of the string pick point then the leader box is drawn to the right of the hook point.

For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.5.8 Bearing & Segment Length

Information Leader choice **Bearing & segment length** displays in the leader box, the bearing at the string pick point and the length of the segment it is on.

A leader line is drawn from a selected position on a string (string pick point) to a selected hook point, and then to the end of the hook itself.

Using the text format, the value of the bearing and length of the pick point and segment is displayed at the end of the hook line.

The leader is added to the model given in the CAD toolbar.

Step 1.Pick the string to label with its bearing & length

When **Segment Length & Bearing** is the choice, the following message is written to the screen message area

```
< (t)ype[Segment Length Bearing] Pick point on segment (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] > [picks][fast][Menu]
```

a or **A** toggles Association **ON** and **OFF**. See 15.4.1 Leader Association
t or **T** changes the type of Information Leader. See 15.4.5 Information Leader
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle

If Association is **ON** then the start of the Leader is associated with the segment pick point.

Step 2.Pick the hook point.

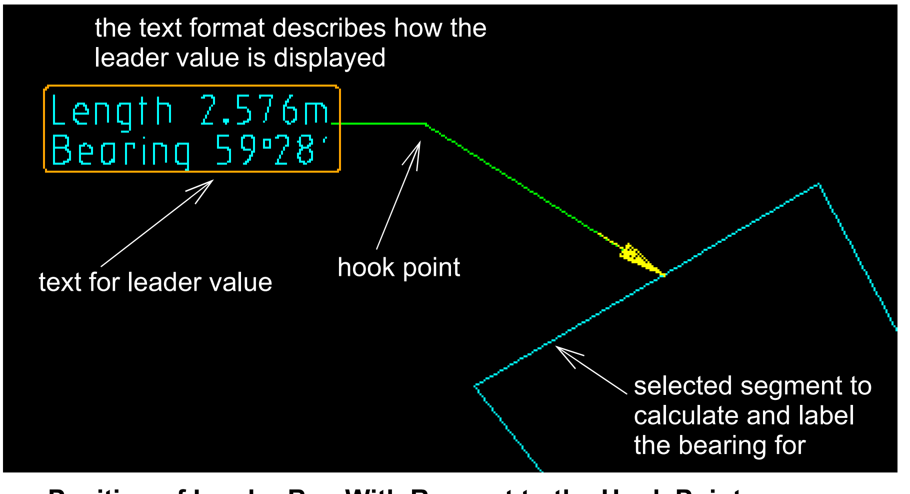
The following message is written to the screen message area.

```
<Pick hook point or customise (t)ext> [picks][fast][Menu]
```

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

The *Information Leader* is then created.

The **Segment Length & Bearing Information Leader** then starts again. That is, it goes back to **Step 1**.



Note on Position of Leader Box With Respect to the Hook Point

As well as determining the length of the leader line, the hook point also determines whether the leader box goes to the left or the right of the hook point.

If the hook point is to the left of the string pick point then the leader box is drawn to the left of the hook point. If the hook point is to the right of the string pick point then the leader box is drawn to the right of the hook point.

For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.5.9 Segment Radius

Information Leader choice **Segment Radius** displays in the leader box the radius of a selected segment. If the segment is a straight then the radius is displayed as 0. If the segment is a transition, then the displayed radius is the radius of the transition at the picked position.

A leader line is drawn from a string pick point to a selected hook point, and then to the end of the hook itself.

Using the text format, the value of the radius of the string pick point is displayed at the end of the hook line.

The leader is added to the model given in the CAD toolbar.

Step 1.Pick the segment to label with its radius

When **Segment Radius** is the choice, the following message is written to the screen message area

< (t)ype[Segment Radius] Pick point on segment (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] > [picks][fast][Menu]

a or **A** toggles Association **ON** and **OFF**. See 15.4.1 Leader Association
t or **T** changes the type of Information Leader. See 15.4.5 Information Leader
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle

If Association is **ON** then the start of the Leader is associated with the string pick point.

Step 2.Pick the hook point

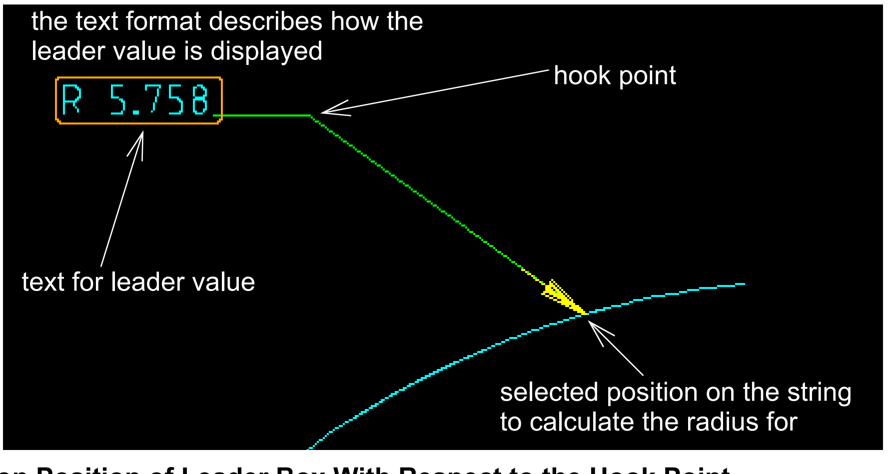
The following message is written to the screen message area.

<Pick hook point or customise (t)ext> [picks][fast][Menu]

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

The *Information Leader* is then created.

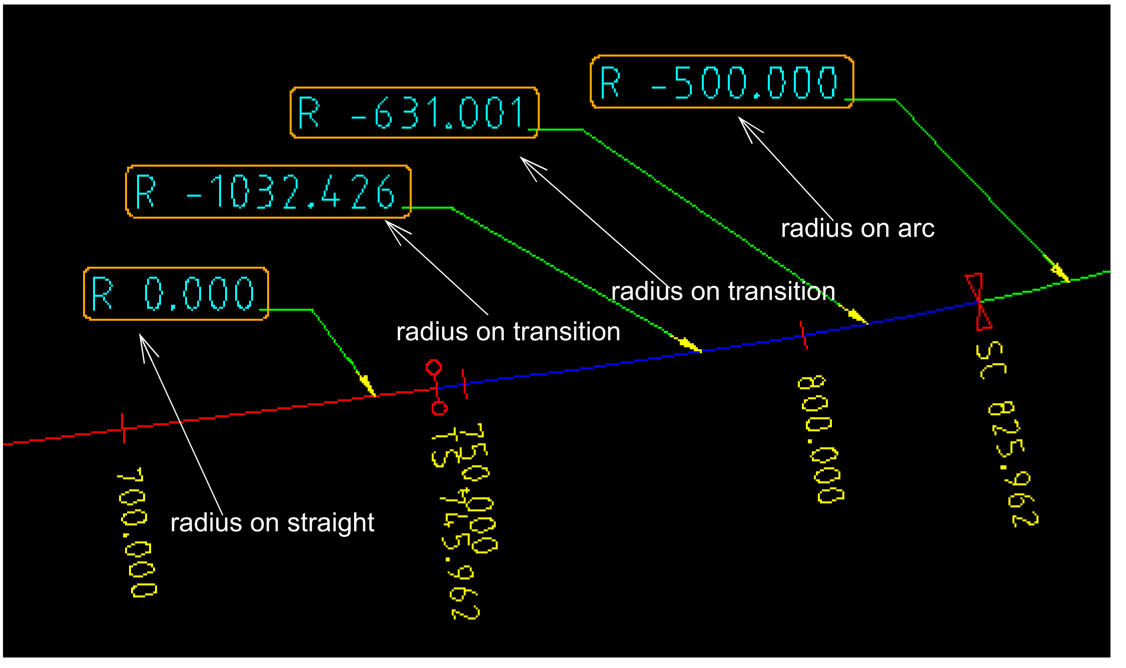
The **Segment Radius Information Leader** then starts again. That is, it goes back to **Step 1**.



Note on Position of Leader Box With Respect to the Hook Point

As well as determining the length of the leader line, the hook point also determines whether the leader box goes to the left or the right of the hook point.

If the hook point is to the left of the string pick point then the leader box is drawn to the left of the hook point. If the hook point is to the right of the string pick point then the leader box is drawn to the right of the hook point.



For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.5.10 String name

Information Leader choice **String name** displays in the leader box, the name of a selected string.

A leader line is drawn from a selected position on a string (string pick point) to a selected hook point, and then to the end of the hook itself.

Using the text format, the name of the string is displayed at the end of the hook line.

The leader is added to the model given in the CAD toolbar.

Step 1.Pick the string to label with its name

When **String Name** is the choice, the following message is written to the screen message area

```
< (t)ype[String Name] Pick string (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] > [picks][fast][Menu]
```

a or **A** toggles Association **ON** and **OFF**. See 15.4.1 Leader Association
t or **T** changes the type of Information Leader. See 15.4.5 Information Leader
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle

If Association is **ON** then the start of the Leader is associated with the segment pick point.

Step 2.Pick the hook point

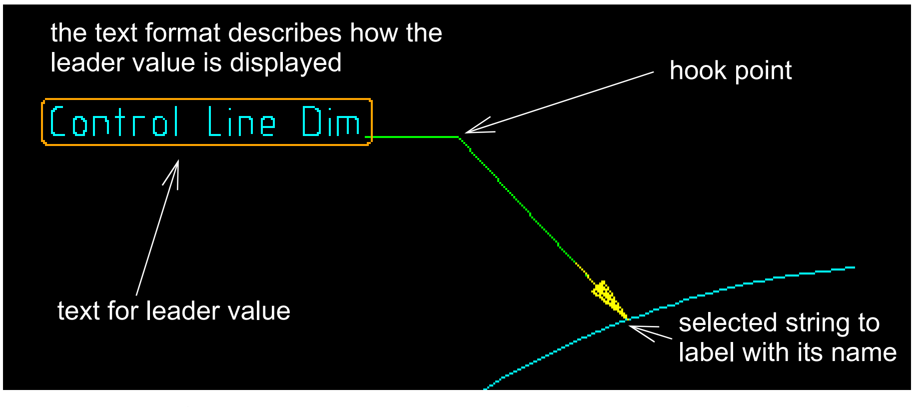
The following message is written to the screen message area.

```
<Pick hook point or customise (t)ext> [picks][fast][Menu]
```

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

The *Information Leader* is then created.

The **String Name Information Leader** then starts again. That is, it goes back to **Step 1**.



Note on Position of Leader Box With Respect to the Hook Point

As well as determining the length of the leader line, the hook point also determines whether the leader box goes to the left or the right of the hook point.

If the hook point is to the left of the string pick point then the leader box is drawn to the left of the hook point. If the hook point is to the right of the string pick point then the leader box is drawn to the right of the hook point.

For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.5.11 Vertex Text

Information Leader choice **Vertex text** displays in the leader box, the vertex text of a selected string vertex.

A leader line is drawn from a selected vertex on a string (string pick point) to a selected hook point, and then to the end of the hook itself.

Using the text format, the vertex text for the selected vertex of the string is displayed at the end of the hook line.

The leader is added to the model given in the CAD toolbar.

Step 1.Pick the string vertex to label with its vertex text

When **Vertex Text** is the choice, the following message is written to the screen message area

```
< (t)ype[Vertex Text] Pick vertex (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] > [picks][fast][Menu]
```

a or **A** toggles Association **ON** and **OFF**. See 15.4.1 Leader Association
t or **T** changes the type of Information Leader. See 15.4.5 Information Leader
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle

If Association is **ON** then the start of the Leader is associated with the segment pick point.

Step 2.Pick the hook point

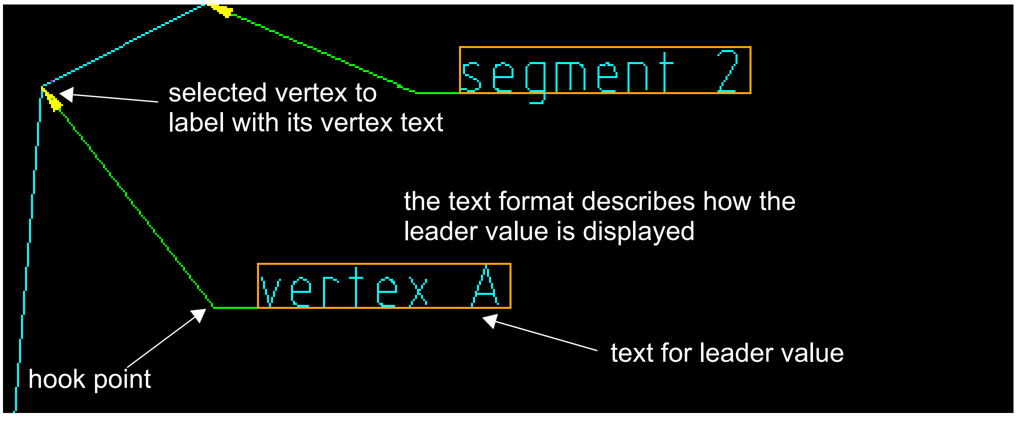
The following message is written to the screen message area.

```
<Pick hook point or customise (t)ext> [picks][fast][Menu]
```

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

The *Information Leader* is then created.

The **Vertex Text Information Leader** then starts again. That is, it goes back to **Step 1**.



Note on Position of Leader Box With Respect to the Hook Point

As well as determining the length of the leader line, the hook point also determines whether the leader box goes to the left or the right of the hook point.

If the hook point is to the left of the string pick point then the leader box is drawn to the left of the hook point. If the hook point is to the right of the string pick point then the leader box is drawn to the right of the hook point.

For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.5.12 Segment Text

Information Leader choice **Segment text** displays in the leader box, the segment text of a selected string segment.

A leader line is drawn from a selected position on a string (string pick point) to a selected hook point, and then to the end of the hook itself.

Using the text format, the segment text for the selected segment of the string is displayed at the end of the hook line.

The leader is added to the model given in the CAD toolbar.

Step 1.Pick the string segment to label with its segment text

When **Segment Text** is the choice, the following message is written to the screen message area

```
< (t)ype[Segment Text] Pick segment (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] > [picks][fast][Menu]
```

a or **A** toggles Association **ON** and **OFF**. See 15.4.1 Leader Association
t or **T** changes the type of Information Leader. See 15.4.5 Information Leader
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle

If Association is **ON** then the start of the Leader is associated with the segment pick point.

Step 2.Pick the hook point

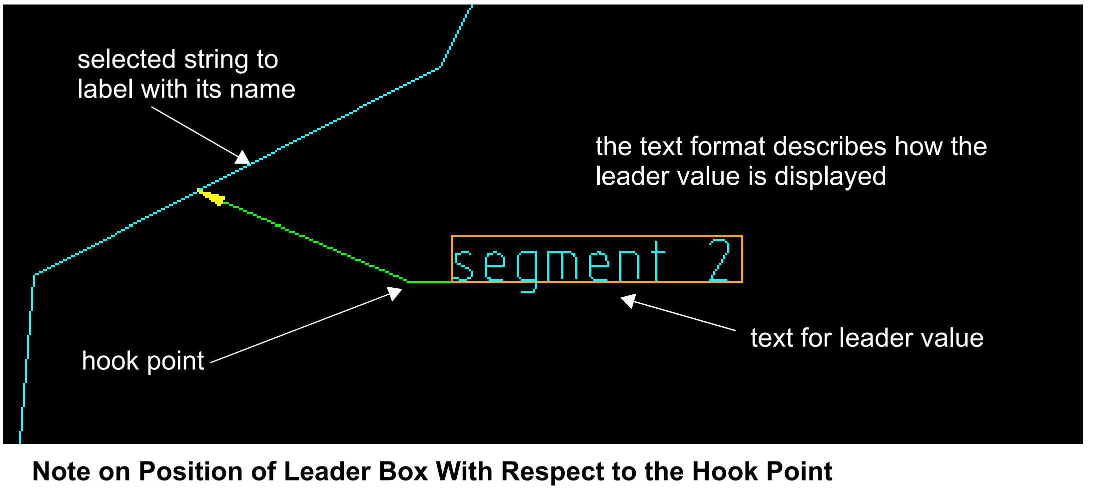
The following message is written to the screen message area.

```
<Pick hook point or customise (t)ext> [picks][fast][Menu]
```

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

The *Information Leader* is then created.

The **Segment Text Information Leader** then starts again. That is, it goes back to **Step 1**.



Note on Position of Leader Box With Respect to the Hook Point

As well as determining the length of the leader line, the hook point also determines whether the leader box goes to the left or the right of the hook point.

If the hook point is to the left of the string pick point then the leader box is drawn to the left of the hook point. If the hook point is to the right of the string pick point then the leader box is drawn to the right of the hook point.

For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.5.13 Vertex XYZ

Information Leader choice **Vertex XYZ** displays in the leader box, the x, y and z coordinates of a selected vertex.

A leader line is drawn from a selected vertex on a string to a selected hook point, and then to the end of the hook itself.

Using the text format, the x, y and z coordinates of a selected vertex are displayed at the end of the hook line.

The leader is added to the model given in the CAD toolbar.

Step 1.Pick the vertex to label with its x, y and z coordinates

When **Vertex XYZ** is the choice, the following message is written to the screen message area

< (t)ype[Vertex XYZ] Pick vertex (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] > [picks][fast][Menu]

a or **A** toggles Association **ON** and **OFF**. See 15.4.1 Leader Association
t or **T** changes the type of Information Leader. See 15.4.5 Information Leader
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle

If Association is **ON** then the start of the Leader is associated with the vertex.

Step 2.Pick the hook point

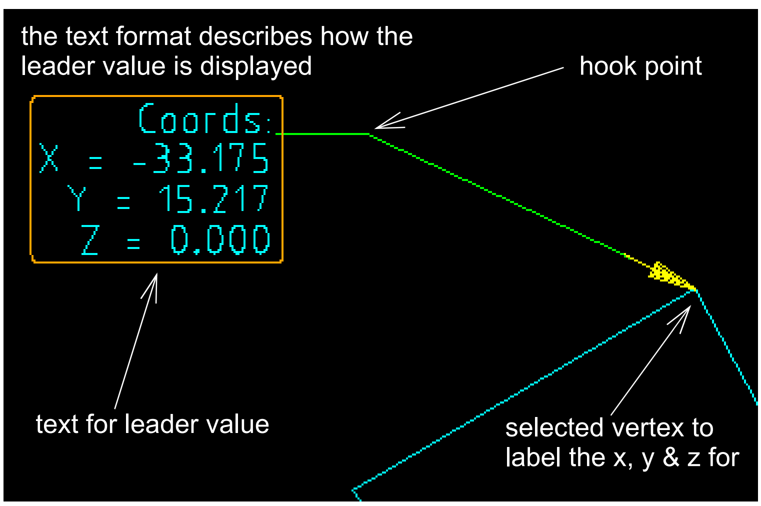
The following message is written to the screen message area.

<Pick hook point or customise (t)ext> [picks][fast][Menu]

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

The *Information Leader* is then created.

The **Vertex XYZ Information Leader** then starts again. That is, it goes back to **Step 1**.



Note on Position of Leader Box With Respect to the Hook Point

As well as determining the length of the leader line, the hook point also determines whether the leader box goes to the left or the right of the hook point.

If the hook point is to the left of the segment pick point then the leader box is drawn to the left of the hook point. If the hook point is to the right of the segment pick point then the leader box is drawn to the right of the hook point.

For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.5.14 Vertex XY

Information Leader choice **Vertex XY** displays in the leader box, the x and y coordinates of a selected vertex.

A leader line is drawn from a selected vertex on a string to a selected hook point, and then to the end of the hook itself.

Using the text format, the x and y coordinates of a selected vertex are displayed at the end of the hook line.

The leader is added to the model given in the CAD toolbar.

Step 1.Pick the vertex to label with its x and y coordinates

When **Vertex XY** is the choice, the following message is written to the screen message area

```
< (t)ype[Vertex XY] Pick vertex (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] > [picks][fast][Menu]
```

a or **A** toggles Association **ON** and **OFF**. See 15.4.1 Leader Association
t or **T** changes the type of Information Leader. See 15.4.5 Information Leader
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle

If Association is **ON** then the start of the Leader is associated with the vertex.

Step 2.Pick the hook point

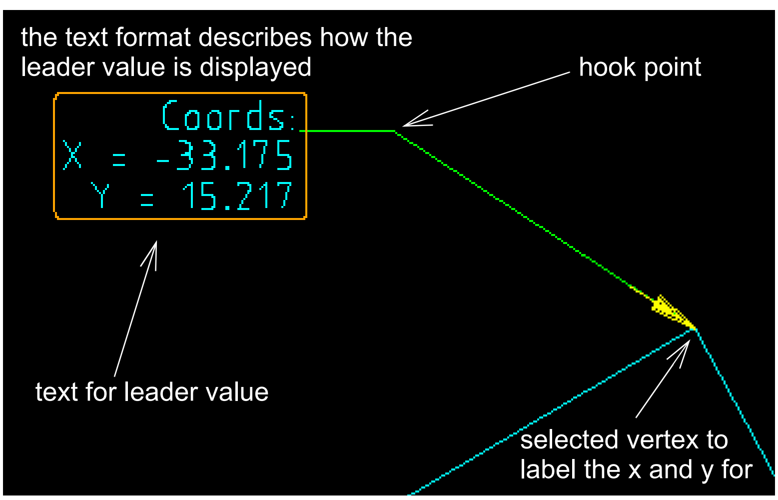
The following message is written to the screen message area.

```
<Pick hook point or customise (t)ext> [picks][fast][Menu]
```

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

The *Information Leader* is then created.

The **Vertex XY Information Leader** then starts again. That is, it goes back to **Step 1**.



Note on Position of Leader Box With Respect to the Hook Point

As well as determining the length of the leader line, the hook point also determines whether the leader box goes to the left or the right of the hook point.

If the hook point is to the left of the segment pick point then the leader box is drawn to the left of the hook point. If the hook point is to the right of the segment pick point then the leader box is drawn to the right of the hook point.

For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.5.15 String XYZ

Information Leader choice **String XYZ** displays in the leader box, the x, y and z coordinates of any selected position on a string.

A leader line is drawn from a selected position on a string (string pick point) to a selected hook point, and then to the end of the hook itself.

Using the text format, the x, y and z coordinates of the selected string position are displayed at the end of the hook line.

The leader is added to the model given in the CAD toolbar.

Step 1.Pick the string position to label with its x, y and z coordinates

When **String XYZ** is the choice, the following message is written to the screen message area

```
< (t)ype[String XYZ] Pick point (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] > [picks][fast][Menu]
```

a or **A** toggles Association **ON** and **OFF**. See 15.4.1 Leader Association
t or **T** changes the type of Information Leader. See 15.4.5 Information Leader
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle

If Association is **ON** then the start of the Leader is associated with the string pick point.

Step 2.Pick the hook point

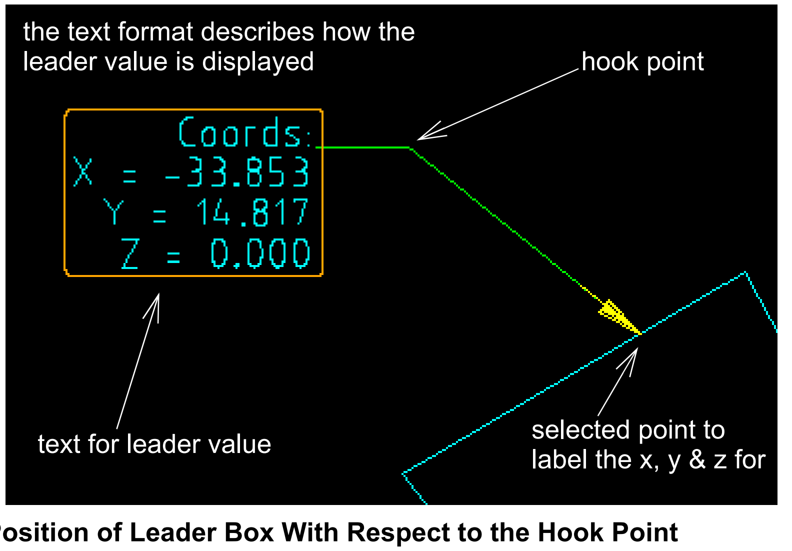
The following message is written to the screen message area.

```
<Pick hook point or customise (t)ext> [picks][fast][Menu]
```

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

The *Information Leader* is then created.

The **String XYZ Information Leader** then starts again. That is, it goes back to **Step 1**.



Note on Position of Leader Box With Respect to the Hook Point

As well as determining the length of the leader line, the hook point also determines whether the leader box goes to the left or the right of the hook point.

If the hook point is to the left of the segment pick point then the leader box is drawn to the left of the hook point. If the hook point is to the right of the segment pick point then the leader box is drawn to the right of the hook point.

For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.5.16 String XY

Information Leader choice **String XY** displays in the leader box, the x and y coordinates of any selected position on a string.

A leader line is drawn from a selected position on a string (string pick point) to a selected hook point, and then to the end of the hook itself.

Using the text format, the x and y coordinates of the selected string position are displayed at the end of the hook line.

The leader is added to the model given in the CAD toolbar.

Step 1.Pick the string position to label with its x and y coordinates

When **String XY** is the choice, the following message is written to the screen message area

< (t)ype[String XY] Pick point (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] > [picks][fast][Menu]

a or **A** toggles Association **ON** and **OFF**. See 15.4.1 Leader Association
t or **T** changes the type of Information Leader. See 15.4.5 Information Leader
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle

If Association is **ON** then the start of the Leader is associated with the string pick point.

Step 2.Pick the hook point

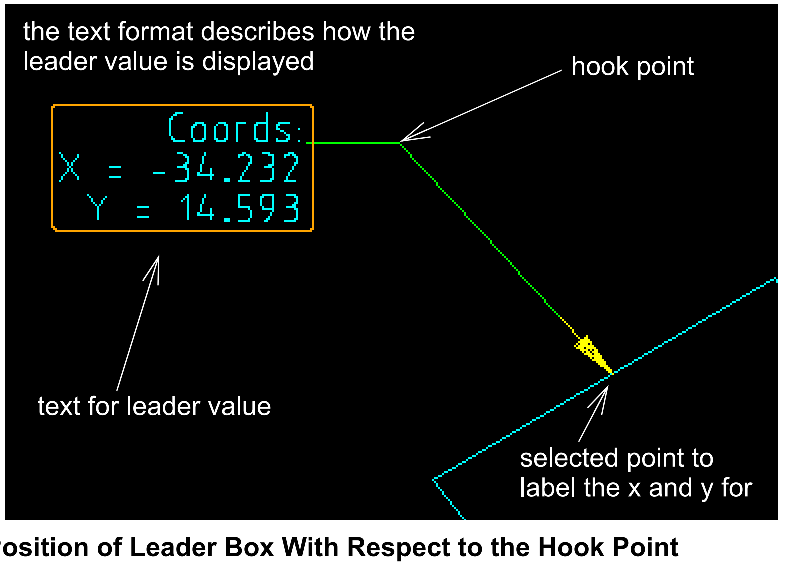
The following message is written to the screen message area.

<Pick hook point or customise (t)ext> [picks][fast][Menu]

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

The *Information Leader* is then created.

The **String XY Information Leader** then starts again. That is, it goes back to **Step 1**.



Note on Position of Leader Box With Respect to the Hook Point

As well as determining the length of the leader line, the hook point also determines whether the leader box goes to the left or the right of the hook point.

If the hook point is to the left of the segment pick point then the leader box is drawn to the left of the hook point. If the hook point is to the right of the segment pick point then the leader box is drawn to the right of the hook point.

For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.5.17 String Z

Information Leader choice **String Z** displays in the leader box, the z coordinate (level) of any selected position on a string.

A leader line is drawn from a selected position on a string (string pick point) to a selected hook point, and then to the end of the hook itself.

Using the text format, the z coordinate (level) of the selected string position are displayed at the end of the hook line.

The leader is added to the model given in the CAD toolbar.

Step 1.Pick the string position to label with level (z coordinate)

When **String Z** is the choice, the following message is written to the screen message area

< (t)ype[String Z] Pick point (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] > [picks][fast][Menu]

a or **A** toggles Association **ON** and **OFF**. See 15.4.1 Leader Association
t or **T** changes the type of Information Leader. See 15.4.5 Information Leader
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle

If Association is **ON** then the start of the Leader is associated with the string pick point.

Step 2.Pick the hook point

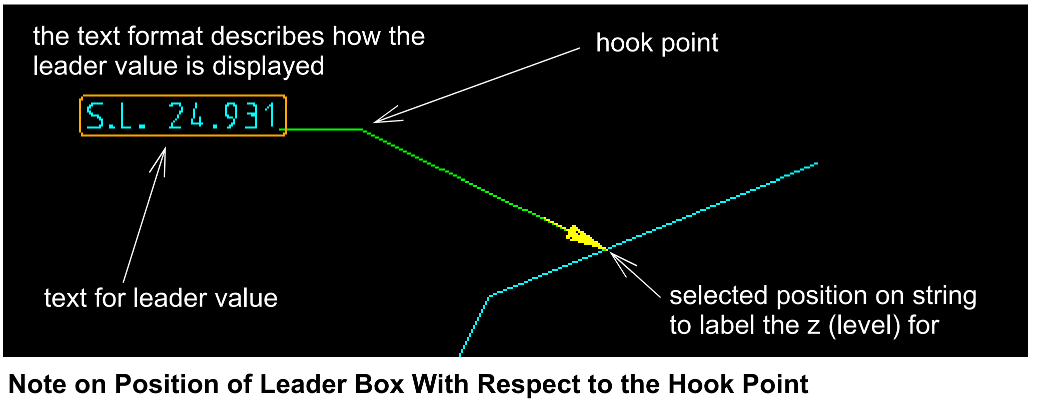
The following message is written to the screen message area.

<Pick hook point or customise (t)ext> [picks][fast][Menu]

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

The *Information Leader* is then created.

The **String Z Information Leader** then starts again. That is, it goes back to **Step 1**.



Note on Position of Leader Box With Respect to the Hook Point

As well as determining the length of the leader line, the hook point also determines whether the leader box goes to the left or the right of the hook point.

If the hook point is to the left of the segment pick point then the leader box is drawn to the left of the hook point. If the hook point is to the right of the segment pick point then the leader box is drawn to the right of the hook point.

For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.5.18 Tin Z

Information Leader choice **Tin Z** displays in the leader box, the z value (level) from the tin at the (x,y) coordinates of a selected position.

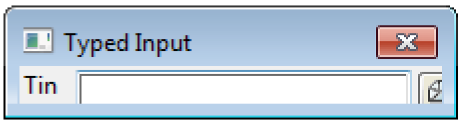
A leader line is drawn from the selected position (pick point) to a selected hook point, and then to the end of the hook itself.

Using the text format, the value of the level from the tin is displayed at the end of the hook line.

The leader is added to the model given in the CAD toolbar.

Step 1.Pick the position that is to be labelled with the level (z value) from the tin

When **Tin Z** is the choice, and if it is the first time the **Tin z** option has been used in this session, then the *Tin* **Typed Input** box is placed on the screen.



The tin to take the levels from is picked from the pop up list, or typed into the box and **<Enter>** pressed. The box is then removed.

The following message is written to the screen message area.

```
< (t)ype[Tin Z] Pick point on tin (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] > [picks][fast][Menu]
```

a or **A** toggles Association **ON** and **OFF**. See 15.4.1 Leader Association
t or **T** changes the type of Information Leader. See 15.4.5 Information Leader
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle
i or **I** brings up the *Tin* **Typed Input** box to select a new tin.

If Association is **ON** then the start of the Leader is associated with the pick point

Step 2.

The following message is written to the screen message area.

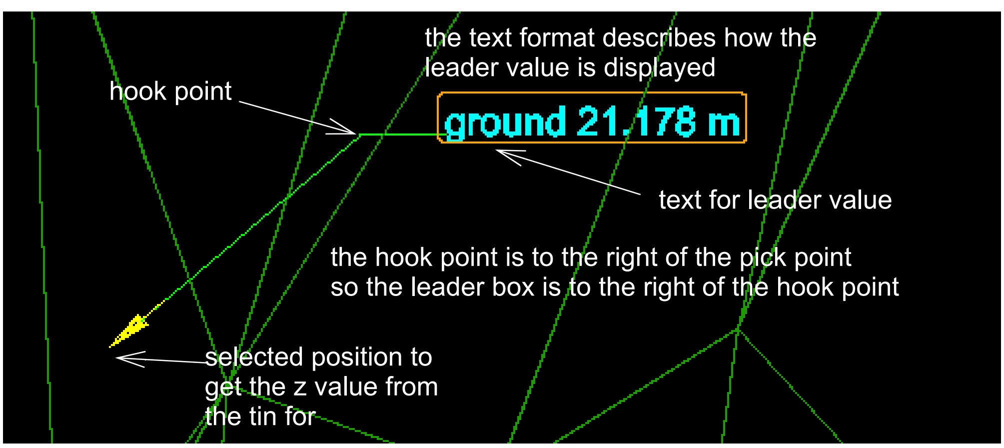
```
<Pick hook point or customise (t)ext> [picks][fast][Menu]
```

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

Pick the hook point.

The *Information Leader* is then created.

The **Tin Z Information Leader** then starts again. That is, it goes back to **Step 1**.



Note - the tin itself does not have to be on any view.

Note on Position of Leader Box With Respect to the Hook Point

As well as determining the length of the leader line, the hook point also determines whether the leader box goes to the left or the right of the hook point.

If the hook point is to the left of the pick point then the leader box is drawn to the left of the hook point. If the hook point is to the right of the pick point then the leader box is drawn to the right of the hook point.

For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.5.19 Tin Depth

Information Leader choice **Tin Depth** displays in the leader box, the depth at a selected position on a string to a tin.

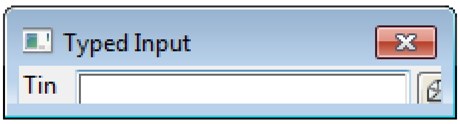
A leader line is drawn from the selected string position (pick point) to a selected hook point, and then to the end of the hook itself.

Using the text format, the value of the depth form the pick point to the tin is displayed at the end of the hook line.

The leader is added to the model given in the CAD toolbar.

Step 1.Pick the point that is to be labelled with its depth to the tin

When **Tin Depth** is the choice, and if it is the first time the **Tin z** option has been used in this session, then the *Tin* **Typed Input** box is placed on the screen.



The tin to take the levels from is picked from the pop up list, or typed into the box and **<Enter>** pressed. The box is then removed

The following message is written to the screen message area.

```
< (t)ype[Tin Depth] Pick point on tin (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] > [picks][fast][Menu]
```

a or **A** toggles Association **ON** and **OFF**. See 15.4.1 Leader Association
t or **T** changes the type of Information Leader. See 15.4.5 Information Leader
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle
i or **I** brings up the *Tin* **Typed Input** box to select a new tin.

If Association is **ON** then the start of the Leader is associated with the pick point

Step 2.Pick the hook point

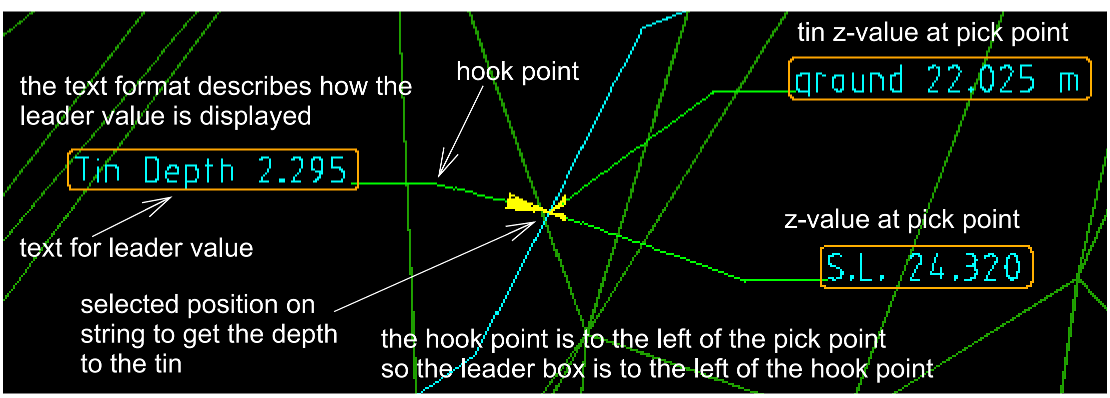
The following message is written to the screen message area.

```
<Pick hook point or customise (t)ext> [picks][fast][Menu]
```

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

The *Information Leader* is then created.

The **Tin Depth Information Leader** then starts again. That is, it goes back to **Step 1**.



Note - the tin itself does not have to be on any view.

Note on Position of Leader Box With Respect to the Hook Point

As well as determining the length of the leader line, the hook point also determines whether the leader box goes to the left or the right of the hook point.

If the hook point is to the left of the pick point then the leader box is drawn to the left of the hook point. If the hook point is to the right of the pick point then the leader box is drawn to the right of the hook point.

For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.5.20 Grade (%)

Information Leader choice **Grade (%)** displays in the leader box, the percent grade of a selected position on string (string pick point).

A leader line is drawn from the string pick point to a selected hook point, and then to the end of the hook itself.

Using the text format, the value of the percent grade at the string pick point is displayed at the end of the hook line.

The leader is added to the model given in the CAD toolbar.

Step 1.Pick the position on a string to label with per cent grade

When **Grade per cent** is the choice, the following message is written to the screen message area

```
< (t)ype[Grade %] Pick point on string (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] > [picks][fast][Menu]
```

a or **A** toggles Association **ON** and **OFF**. See 15.4.1 Leader Association
t or **T** changes the type of Information Leader. See 15.4.5 Information Leader
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle

If Association is **ON** then the start of the Leader is associated with the string pick point.

Step 2.Pick the hook point

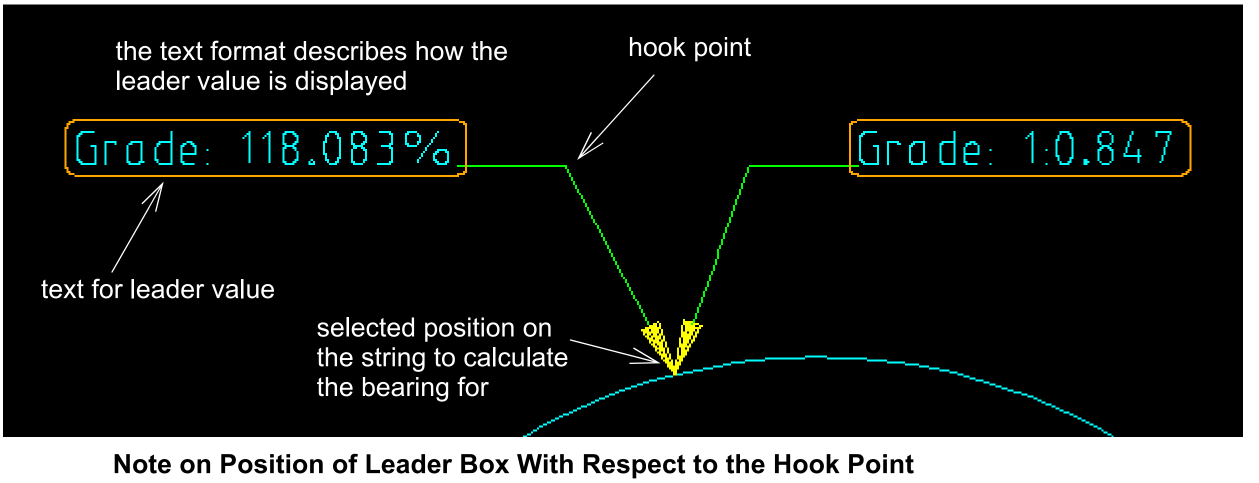
The following message is written to the screen message area.

```
<Pick hook point or customise (t)ext> [picks][fast][Menu]
```

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

The *Information Leader* is then created.

The **Grade per cent Information Leader** then starts again. That is, it goes back to **Step 1**.



Note on Position of Leader Box With Respect to the Hook Point

As well as determining the length of the leader line, the hook point also determines whether the leader box goes to the left or the right of the hook point.

If the hook point is to the left of the string pick point then the leader box is drawn to the left of the hook point. If the hook point is to the right of the string pick point then the leader box is drawn to the right of the hook point.

For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.5.21 Grade 1 n - Slope

Information Leader choice **Grade 1 in** displays in the leader box, the grade as a 1 in (slope) of a selected position on string (string pick point).

A leader line is drawn from the string pick point to a selected hook point, and then to the end of the hook itself.

Using the text format, the value of the grade as a 1 in (slope) at the string pick point is displayed at the end of the hook line.

The leader is added to the model given in the CAD toolbar.

Step 1.Pick the position on a string to label with its grade as 1:in (slope)

When **Grade 1 in** is the choice, the following message is written to the screen message area

```
< (t)ype[Grade 1 In] Pick point on string (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] > [picks][fast][Menu]
```

a or **A** toggles Association **ON** and **OFF**. See 15.4.1 Leader Association
t or **T** changes the type of Information Leader. See 15.4.5 Information Leader
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle

If Association is **ON** then the start of the Leader is associated with the string pick point.

Step 2.Pick the hook point

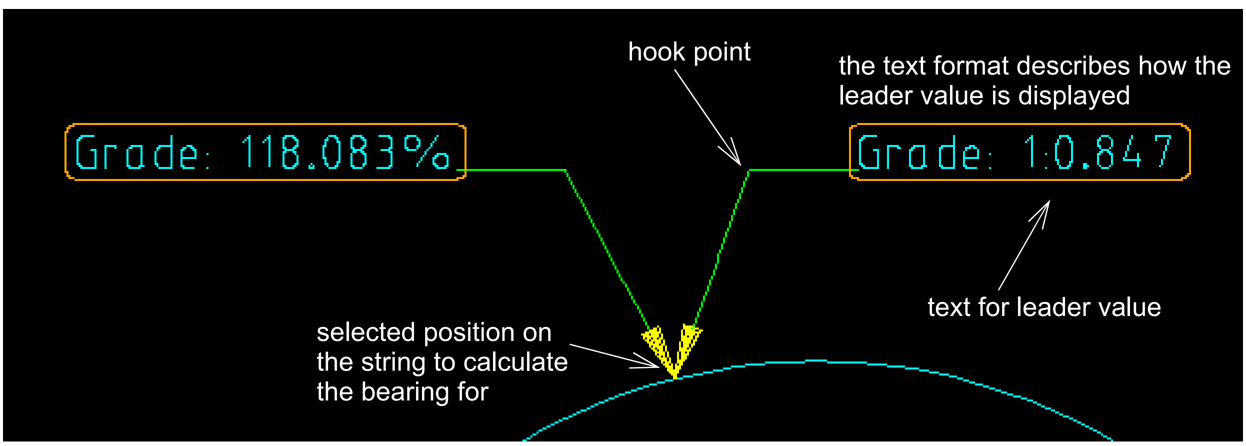
The following message is written to the screen message area.

```
<Pick hook point or customise (t)ext> [picks][fast][Menu]
```

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

The *Information Leader* is then created.

The **Grade 1 in Information Leader** then starts again. That is, it goes back to **Step 1**.



Note on Position of Leader Box With Respect to the Hook Point

As well as determining the length of the leader line, the hook point also determines whether the leader box goes to the left or the right of the hook point.

If the hook point is to the left of the string pick point then the leader box is drawn to the left of the hook point. If the hook point is to the right of the string pick point then the leader box is drawn to the right of the hook point.

For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.5.22 Centroid XY

Information Leader choice **Centroid XY** displays in the leader box, the (x,y) coordinates of the centroid of a selected string. If the string is not a closed string (a polygon) then it is temporarily closed for calculating the Centroid. Note that the centroid may be outside the closed string - for a point that is inside the string, see 15.4.5.23 Medial Centre XY.

A string is selected (string pick point) and a position for the hook and leader line is drawn from either the string pick point or the **centroid** of the string, to a selected hook point, and then to the end of the hook itself.

Using the text format, the (x,y) coordinates of the centroid of the selected string are displayed at the end of the hook line.

The leader is added to the model given in the CAD toolbar.

Step 1.Pick the string to label with the (x,y) coordinates of its centroid

When **Centroid XY** is the choice, the following message is written to the screen message area

< (t)ype[Centroid XY] Pick string (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] fixed-a(r)row> [picks][fast][Menu]

a or **A** toggles Association **ON** and **OFF**. See 15.4.1 Leader Association
t or **T** changes the type of Information Leader. See 15.4.5 Information Leader
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle
r or **R** toggles between the arrow starting at the pick point (**free arrow**) or at the polygon centre (**fixed arrow**)

If Association is **ON** then the start of the Leader is associated with the pick point or the centroid of the string.

Step 2.Pick the hook point

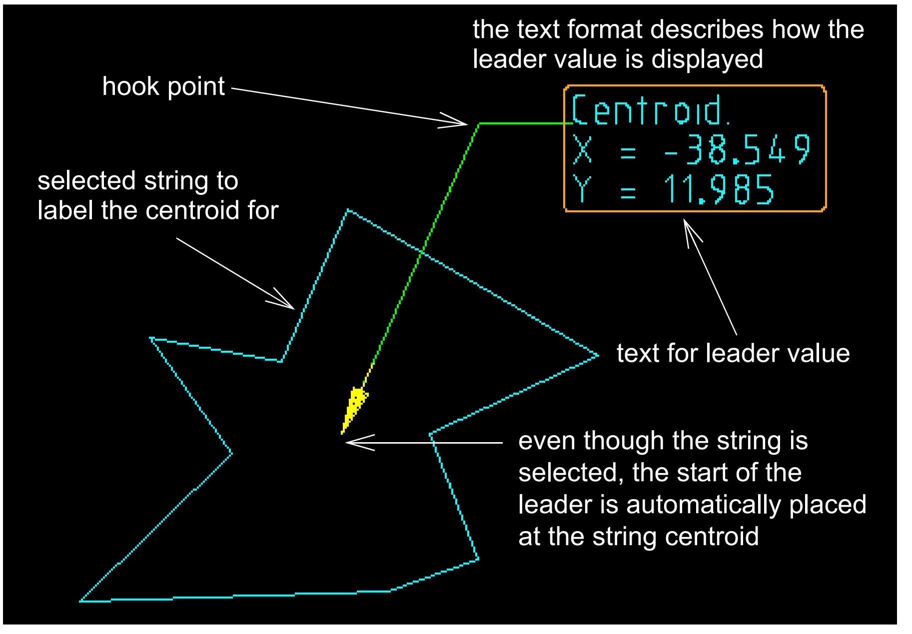
The following message is written to the screen message area.

<Pick hook point or customise (t)ext> [picks][fast][Menu]

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

The *Information Leader* is then created.

The **Centroid XY Information Leader** then starts again. That is, it goes back to **Step 1**.



Note on Position of Leader Box With Respect to the Hook Point

As well as determining the length of the leader line, the hook point also determines whether the leader box goes to the left or the right of the hook point.

If the hook point is to the left of the centroid of the string then the leader box is drawn to the left of the hook point. If the hook point is to the right of the centroid of the string then the leader box is drawn to the right of the hook point.

For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.5.23 Medial Centre XY

Information Leader choice **Centre XY** displays in the leader box, the (x,y) coordinates of the medial axis centre of a selected string. If the string is not a closed string (polygon) then it is temporarily closed for calculating the Medial centre.

Note - the Medial centre is a pointed calculated by 12d to use as the **"**centre**"** of a polygon and unlike a centroid, it is always **inside** the polygon. WARNING - this currently does not take arcs into consideration.

A string is selected (string pick point) and a position for the hook and leader line is drawn from either the string pick point or the **medial axis centre** of the string, to a selected hook point, and then to the end of the hook itself.

Using the text format, the (x,y) coordinates of the medial centre of the selected string are displayed at the end of the hook line.

The leader is added to the model given in the CAD toolbar.

Step 1.Pick the string to label with the (x,y) coordinates of its (medial) centre

When **Centre XY** is the choice, the following message is written to the screen message area

```
< (t)ype[Centroid XY] Pick string (a)ssociative (s)tyle[Arial Border] (h)ook-angle[0°] fixed-a(r)row> [picks][fast][Menu]
```

a or **A** toggles Association **ON** and **OFF**. See 15.4.1 Leader Association
t or **T** changes the type of Information Leader. See 15.4.5 Information Leader
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle
r or **R** toggles between the arrow starting at the pick point (**free arrow**) or at the polygon centre (**fixed arrow**)

If Association is **ON** then the start of the Leader is associated with the centre of the string.

Step 2.Pick the hook point

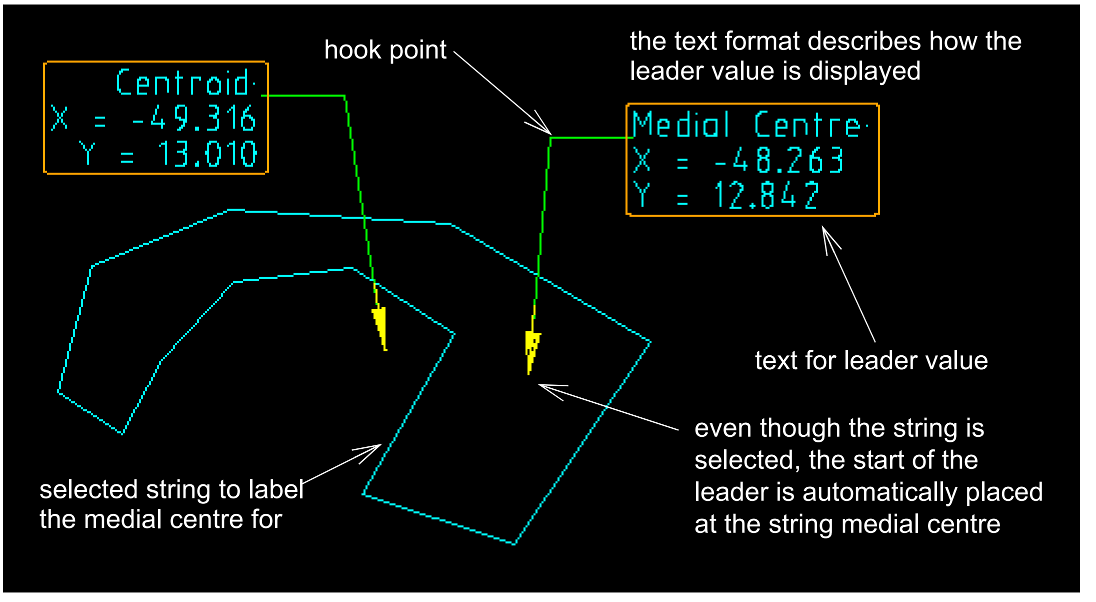
The following message is written to the screen message area.

```
<Pick hook point or customise (t)ext> [picks][fast][Menu]
```

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

The *Information Leader* is then created.

The **Centre XY Information Leader** then starts again. That is, it goes back to **Step 1**.



Note on Position of Leader Box With Respect to the Hook Point

As well as determining the length of the leader line, the hook point also determines whether the leader box goes to the left or the right of the hook point.

If the hook point is to the left of the medial centre of the string then the leader box is drawn to the left of the hook point. If the hook point is to the right of the medial centre of the string then the leader box is drawn to the right of the hook point.

For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.5.24 Trimesh Volume

The **Leader** choice **Trimesh volume** displays in the leader box, the volume of a selected closed trimesh.

A trimesh is selected (trimesh pick point) and a position for the hook and leader line is drawn from the trimesh pick point to a selected hook point, and then to the end of the hook itself.

Using the text format, the volume of the selected trimesh is displayed at the end of the hook line.

The leader is added to the model given in the CAD toolbar.

Step 1.Pick the trimesh to label with its volume

When **Trimesh Volume** is the choice, the following message is written to the screen message area

<Pick trimesh (s)tyle[default] (h)ook-angle[0°] > [picks][fast][Menu]

s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle

Step 2.Pick the hook point

The following message is written to the screen message area.

<Pick hook point or customise (t)ext> [picks][fast][Menu]

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

The *Trimesh Volume Leader* is then created.

The **Trimesh Volume Leader** then starts again. That is, it goes back to **Step 1**.

Note on Position of Leader Box With Respect to the Hook Point

As well as determining the length of the leader line, the hook point also determines whether the leader box goes to the left or the right of the hook point.

If the hook point is to the left of the medial centre of the string then the leader box is drawn to the left of the hook point. If the hook point is to the right of the medial centre of the string then the leader box is drawn to the right of the hook point.

For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.5.25 Trimesh Surface Area

The **Leader** choice **Trimesh area** displays in the leader box, the surface area of a selected trimesh.

A trimesh is selected (trimesh pick point) and a position for the hook and leader line is drawn from the trimesh pick point to a selected hook point, and then to the end of the hook itself.

Using the text format, the volume of the selected trimesh is displayed at the end of the hook line.

The leader is added to the model given in the CAD toolbar.

Step 1.Pick the trimesh to label with its surface area

When **Trimesh area** is the choice, the following message is written to the screen message area

```
<Pick trimesh (s)tyle[default] (h)ook-angle[0°] > [picks][fast][Menu]
```

s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
h or **H** brings up the Hook Angle 15.4.2 Hook Angle

Step 2.Pick the hook point

The following message is written to the screen message area.

```
<Pick hook point or customise (t)ext> [picks][fast][Menu]
```

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.

The *Trimesh Area Leader* is then created.

The **Trimesh Area Leader** then starts again. That is, it goes back to **Step 1**.

Note on Position of Leader Box With Respect to the Hook Point

As well as determining the length of the leader line, the hook point also determines whether the leader box goes to the left or the right of the hook point.

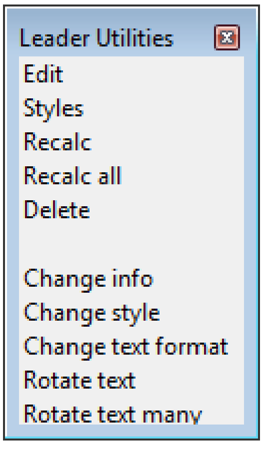
If the hook point is to the left of the medial centre of the string then the leader box is drawn to the left of the hook point. If the hook point is to the right of the medial centre of the string then the leader box is drawn to the right of the hook point.

For information on how the Leader object appears, see 15.4.6.2 Leader Styles.

15.4.6 Leader Utilities

Position of menu: **Cad =>Leader =>Utilities**

The **Leader Utilities** menu is:



edit a leader
define styles for leaders

See

15.4.6.1 Leader Edit
15.4.6.2 Leader Styles
Recalc - select a leader to recalc.
Recalc all - recalcs all leaders.
Delete - select a leader to delete.
15.4.6.3 Change Style of Leader
15.3.18.4 Change Text Format (same option for Dimensions and Leaders)

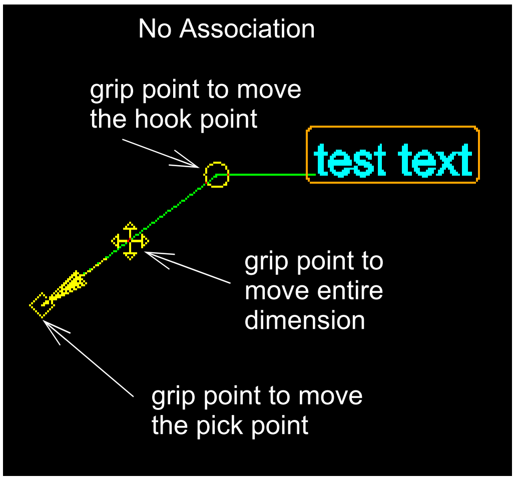
15.4.6.1 Leader Edit

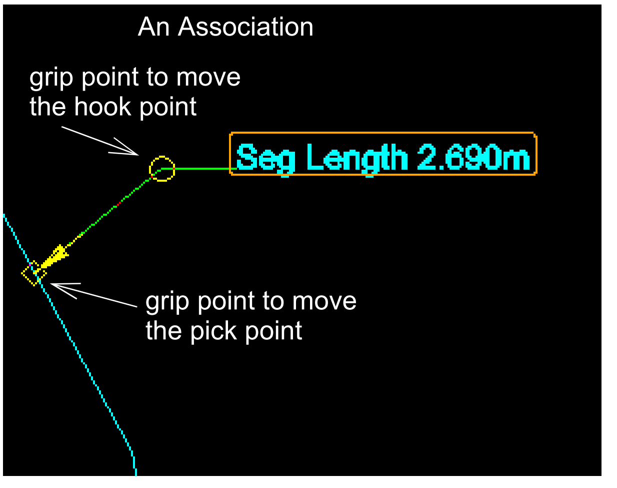
Picking an existing Leader will bring up the Leader editor and the grip points for the pick point and the hook point are displayed.

If there is **no Association** because either the Association is OFF, or Association is ON but there is nothing currently associated with the leader, then you won't get a grip point to move the entire leader because the leader is locked to its associated items.

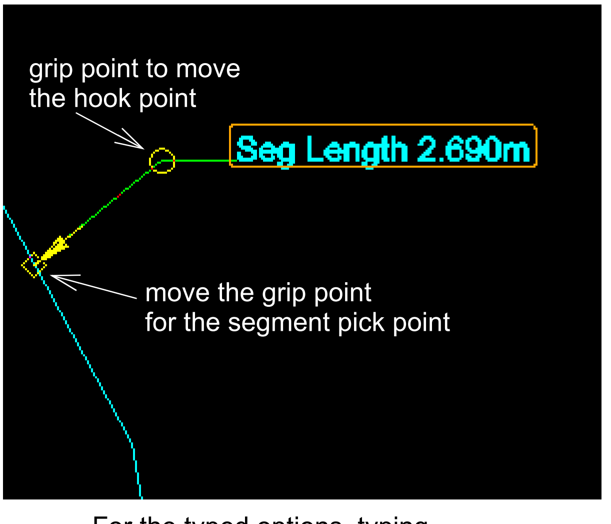
A list of typed options is also displayed in the screen message area.

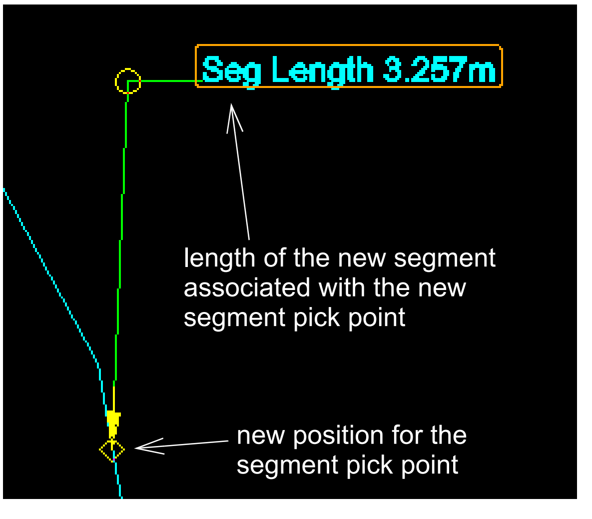
<[Pick grip point] (t)ext (s)tyle[default] (h)ook-angle[0°] (f)lip (d)elete a(n)other> [picks][fast][Menu]





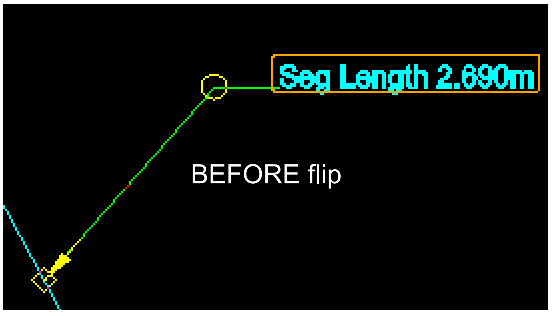
Picking and accepting a grip point will move the grip point and when it is placed again, the association is recalced and the new value for the leader displayed.

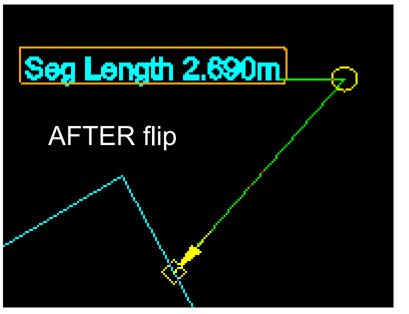




For the typed options, typing

t or **T** brings up the leader text format. See 15.5 Text Format for Dimensions and Leaders.
s or **S** brings up the leader style list. 15.4.6.2 Leader Styles
f or **F** to flip the side that the hook goes to.
d or **D** deletes the leader
n or **N** pick a new leader to edit.





Also

Pressing **<Esc>** will exit the editing of the current *leader* and ask for a new drafting element (Leader or Dimension) to be selected for editing.

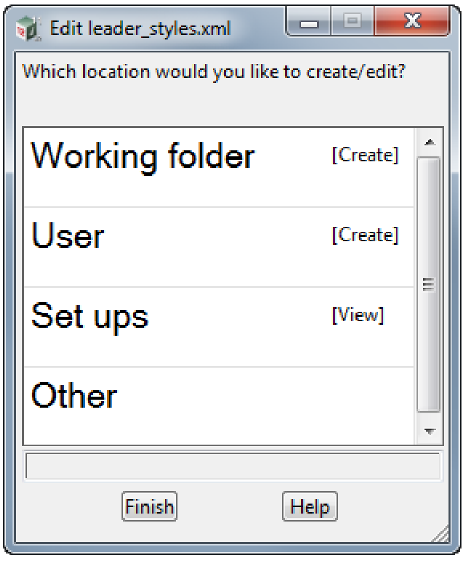
Similarly clicking **RB** and selecting **Cancel** from the **Pick Ops** menu will exit the editing of the current *leader* and ask for a new drafting element (Leader or Dimension) to be selected for editing.

When back in the Drafting Editor, pressing **<Esc>** or clicking **RB** and selecting **Cancel** from the **Pick Ops** menu, will exit the Drafting Editor

15.4.6.2 Leader Styles

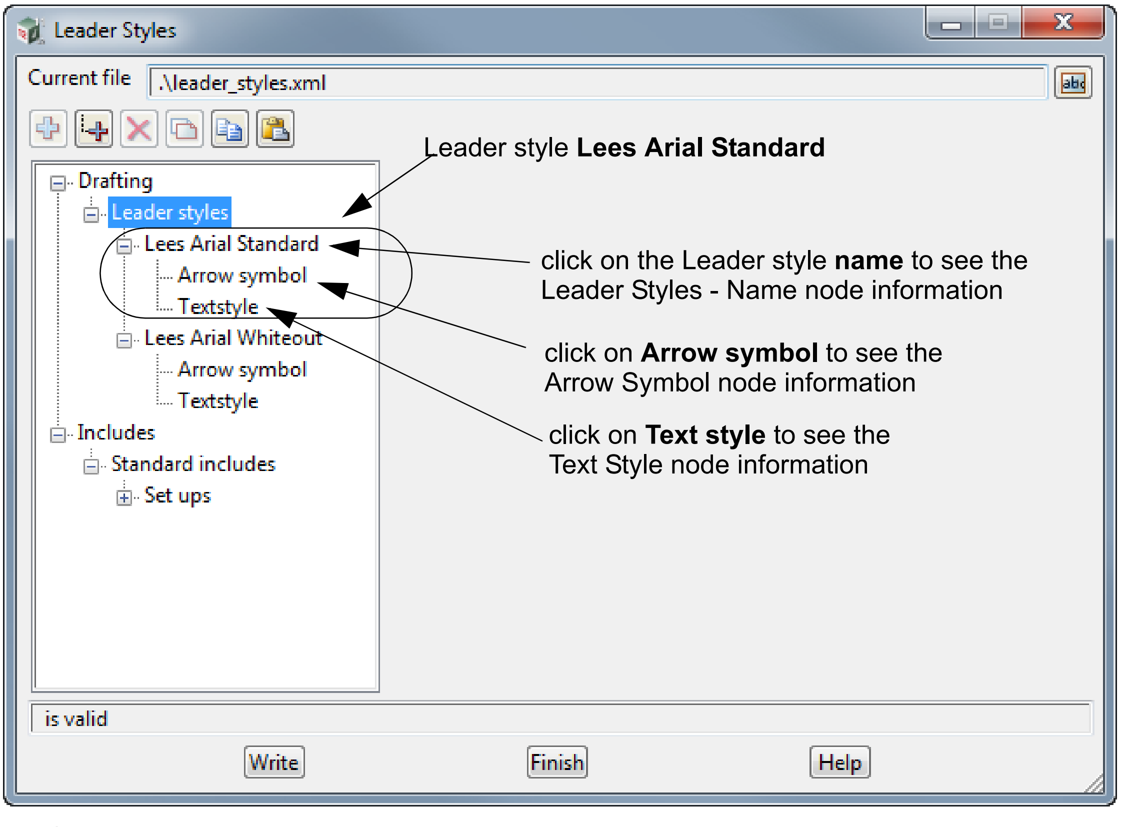
Position of option on menu: **CAD =>Leader =>Styles**

When you click on the option an **Edit leader_styles.xml** panel is brought up and the panel shows the standard areas for looking for the leader_styles.XML files - Working folder, Customer (User), User, Set Ups and Other.



For information about where how to find and create dimension_styles.xml files, see 15.7 Style XML Files for Dimensions, Leaders & Tables.

Once a leader_styles.xml file has been opened, the **Leader Styles** panel is displayed.



See

15.4.6.2.1 Leader Styles - Name Node

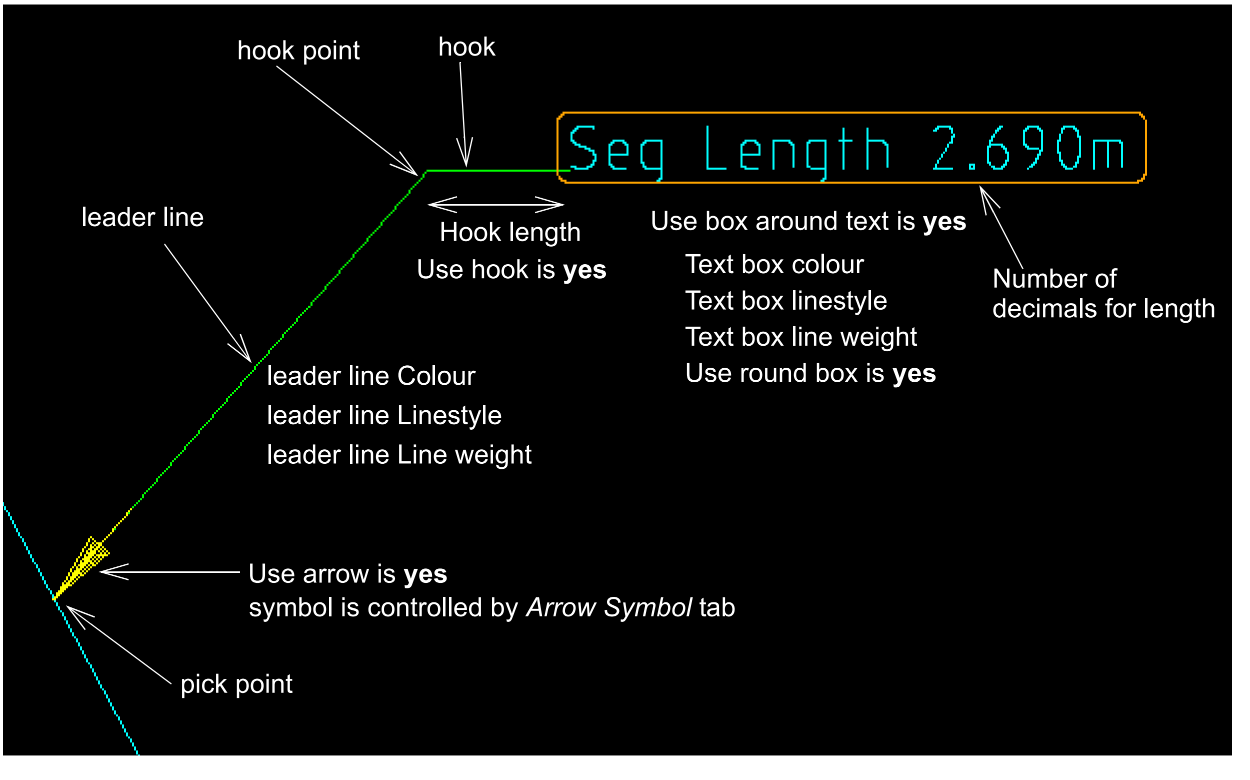
15.4.6.2.1 Leader Styles - Name Node

This is on the right hand side of the panel when you click on the ***Leader style*** name.

Name	Paper
Group	no hook
Colour	green
Linestyle	THIN
Lineweight	1
Use arrow	yes
Use hook	yes
Hook length	0.5
Use box around text	yes
Use rounded box	yes
Text box colour	orange
Text box linestyle	1
Text box lineweight	0
Number of decimals for length	3
Number of decimals for area	2
Use dynamic size	no
Hook angle	0
Angle format style	show_angle
Number of decimal for angle	2
Number of decimals for volume	2

Hook length has same units as Textstyle. So i the Textstyle is paper, then **Hook length** s in millimetres.

these settings are not currently used



If the **Type** from the **Textstyle** tab is **not world,** then the following are not used.

Use text box - should be *Use box around text* like dimension.

Text box rounded - should be *Use rounded box* like dimension?.
Text box colour
Text box linestyle

Text box line weight

The following items that are at the bottom of the tab have not yet been implemented.

Hook angle
Use dynamic size
Number of decimals for volumes
Angle format style

15.4.6.2.2 Leader Styles - Arrow Symbol Node

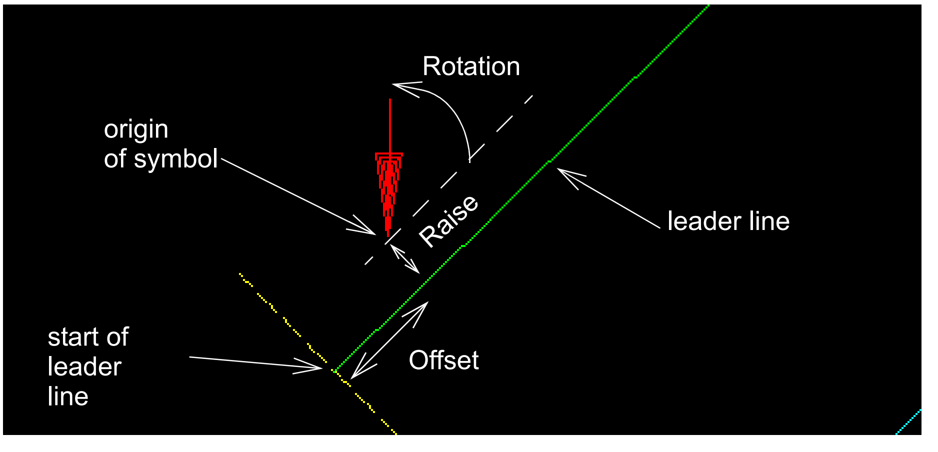
This is on the right hand side of the panel when you click on the ***Arrow symbol*** node.

Style	Arrow
Colour	red
Size	0.5
Rotation	45
Offset	0.2
Raise	0.5

The origin of the symbol is displaced from the **start** of the leader line by the **Offset** and **Raise** values.

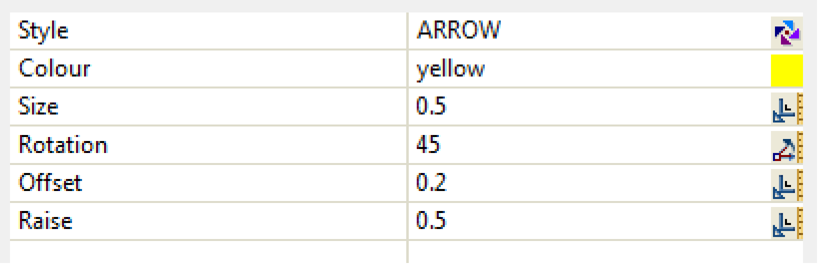
With a **Rotation** value of zero, the symbol is given the same rotation as the leader line.

For a non zero **Rotation**, the value is **added** to the angle of the leader line.



15.4.6.2.3 Leader Styles - Hook Symbol Tab

Not implemented at this stage.



Style	ARROW
Colour	yellow
Size	0.5
Rotation	45
Offset	0.2
Raise	0.5

15.4.6.2.4 Leader Styles - Textstyle Node

This is on the right hand side of the panel when you click on the ***Textstyle*** node.

Textstyle	Arial
Name	
Type	paper
Size	0.2
Angle	0°
X-factor	1
Slant	0
Offset	0
Raise	0
Colour	cyan
Whiteout	no colour
Border	no colour
Border type	
Justify	bottom-left
TTF Underline	no
TTF Strikeout	no
TTF Italic	no
TTF Outline	no
TTF Weight	400

This is expected to be modified shortly so that it uses a *Textstyle Data* and all the features from *Textstyle Data* are automatically available. For the moment with paper text, only **Whiteout** is supported.

When the *Textstyle Data* is implemented, it can have its own *Border type* but there is no linestyle for the border in *Textstyle Data*. The only way to get a border with a linestyle is to use World text and set the border parameters in the *General* tab (15.4.6.2.1 Leader Styles - Name Node).

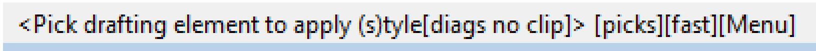
If **Type** is not ***world***, then **Don't clip dimension line for tex**t is set to **no** and these items from the General tab are not used.

Use box around text
Use rounded box
Text box colour
Text box linestyle

See 15.4.6.2.1 Leader Styles - Name Node

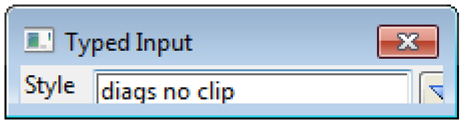
15.4.6.3 Change Style of Leader

Click on **Change style of style** and the following message is written to the screen message area



Pick a leader and its style will be changed to that shown in the square brackets ([]). Then pick another leader to change its style.

Typing **s** or **S** brings up the *Style* **Typed Input** box to select a new leaders style.



The new style is picked from the pop up list, or typed into the box and <Enter> pressed.

The box is then removed and the message again written to the screen message area.

All subsequent selected leaders will have their styles changed to this new style until either the option is terminated or a new leader style is selected

To terminate the option, press **<Esc>**, select **Cancel** from the **Pick Ops** menu, or select another CAD option.

15.5 Text Format for Dimensions and Leaders

For *Dimensions* and *Leaders*, there is one, and sometimes more than one, value that is to be displayed in the *Dimension* or *Leader* as text. For example, the length of a segment or the x an y coordinates of a point.

The user rarely want to display all the different dimes ion and leader values in the same way. For example, some values may be displayed in metres, others in millimetres. Or you may want to display both metres and yards.

In ***12d Model***, there are special **Text Formats** to allow the user to tailor how the values of the *Dimensions* and *Leaders* are displayed.

So a ***Text Format*** is used to describe how a single value for a dimension or leader is displayed.

In general is has the form

*pre_text***Expression***post_text*

where **Expression** can be either

(a) a set of angle brackets enclosing a *value_format* which describes how the value is modified and how many decimal places are used in the final text display of the value

<*value_format*>

of

(b) and expression involving an **IF** statement (known as a ***dynamic text format***)

The full structure of the **Text Format**, the *value_format* and dynamic text format is described in the following sections:

15.5.1 Angle Brackets <>
15.5.2 Pre and Post Text
15.5.3 Units Factor
15.5.4 Round Off
15.5.5 Number of Decimals
15.5.6 Dynamic Text Format
15.5.7 Multiple Text Formats
15.5.8 Typing t or T to Change the Text Format
15.5.9 Defaults File for Text Formats

15.5.1 Angle Brackets <>

In the Text Formats, angle brackets **<>** are used to indicate the position of the value of the *Dimension* or *Leader*.

There can be special value formatting instructions inside the angle brackets to indicate if and how the value is to be modified, and how many decimals to use when finally displayed as text.

<*value_format*>

What is available for *value_format* will be described in the following sections.

In some cases, the angle brackets may not appear at all in the Text Format and then the text does not include the actual value itself.

If the *Dimension* or *Leader* has only one value, then just <> is required.

If the *Dimension* or *Leader* has more than one value, then extra commands are needed to indicate which value is being talked about in the Text Format.

<x> is used for the x value of a coordinate

<y> is used for the y value of a coordinate

<z> is used for the z value of a coordinate

15.5.2 Pre and Post Text

The text displayed for a dimension or leader value can have user defined *pre* and *post* text around the value and that is indicated by the expression

*pre_text***<***value_format***>***post_text*

where **pre_text** is written before the dimension or leader value, **post_text** is written after the value.

Typing **\\n** into **pre_text** or **post_text** will be interpreted as a new line.

The angle brackets **<>** indicate where you want the value in relation to the **pre_text** and **post_text**. The angle brackets may even be missing.

For example, to show just the word “varies”, use the following with no angle brackets.

varies

15.5.3 Units Factor

The value to be displayed can be multiplied by a ***units_factor.*** The units_factor is written inside the angle brackets <>. That is

<*units_factor*>

So the expression

*pre_text***<***units_factor***>***post_text*

means that **pre_text** is written before the dimension or leader value, **post_text** is written after the value, **<>** indicates that you want the value and the value is multiplied by the **units_factor** before being displayed.

If **units_factor** is left blank then it is defaulted to 1.

If **<units_factor>** is left out, then there is no value in the dimension or leader text but you still get the **pre_text** and **post_text**.

Note

pre_text<units_factor>post_text is considered to be **one** block of text. So raise, offset, justification etc apply to the whole block, not to the *pre_text* or *post_text* individually.

As an example of *pre_text<units_factor>post_text*, you can show millimetres instead of metres by using:

<1000>

or, if you want the text “ mm” after the value

<1000> mm

15.5.4 Round Off

Inside the angle brackets <>, the type of Rounding Off can be specified by a key character **O** (down), **M** (middle) or **U** (up) followed immediately (with no spaces) by the ***rounding_unit***.

O*rounding_unit* means to round down to the closest rounding_unit

M*rounding_unit* means to round to the closest rounding_unit

U*rounding_unit* means to round up to the closest rounding_unit

For example

O1 means that 10.2 would round down to 10.

O0.1 means that 10.2 would round down to 10.2

D0.25 means that 10.2 would round up to 10.

U0.25 means that 10.2 would round up to 10.25

M0.25 means that 10.2 would round up to 10.25 rather than down to 10.

Note: the *units_factor* is applied first and then the *Round Off.*

15.5.5 Number of Decimals

Inside the angle brackets <>, you can give the number of decimal places to use when writing out the real value by using the key character **D** followed immediately (with no spaces) by the ***number of decimal places***.

D*n* where **n** is the number of decimal places.

*pre_text***<***units_factorDn***>***post_text*

The *units_factor* is applied first and then the number of decimal places used on the resulting value.

For example

<1000D1>

says to multiply the real by 1000 and only use one decimal place in the text for the value.

Note: the *units_factor* is applied first, followed by the *Round Off*, and finally the *number of decimal places* used in the text to display the resulting value.

So <100D2U0.25> is the same as <100U0.25D2> - multiply by 100, round Up using the *rounding_unit* of 0.25, and then writing the result with two decimal places.

15.5.6 Dynamic Text Format

It is also possible to change the format for a value depending on the size of the value by using an **Expression** with an **IF** test instead of the simple angle brackets **< >**.

The **Expression** has key words _IF_,_THEN_, _ELSE_ and _ELSEIF_ and can be

_IF_ op1 *comparison_value_1* **_THEN_** *single_value_format_1* **_ELSEIF_ op2** *comparison_value_2* **_THEN_** *single_value_format_2***_ELSE_IF_ op3** *comparison_value_3* **_THEN_** *single_value_format_3***…_ELSE_** *single_value_format_n*

where **op1** and **op2** are one of

=	for *equals*
>	for *greater than*
>=	for *greater than or equal to*
<	for *less than*
<=	for *less than or equal to*

and

single_value_format_n is a **Text Format** for a single value and **not** an Expression.

The comparison is always being made between the *dimension/leader value* and the *comparison_value* so we have left out the *dimension/leader value*.

That is, **_IF_** *>comparison_value* means **if** (*dimension value* is greater than *comparison_value*).

For example if you wanted to display kilometres to three decimal place, for anything greater than or equal to 1,000, metres to three decimal places for values greater or equal to 1, otherwise millimetres with no decimal places, then use

_IF_>=1000_THEN_<0.001D3> km _ELSEIF_>=1_THEN_<D3> m_ELSE_<1000D0>mm

Important Notes

1. The *Dynamic Text Format* always evaluates to only one piece of text.
2. The underscore is part of **_IF_**, **_THEN_**, **_ELSEIF_** and **_ELSE_**.
3. **_IF_ op** *comparison_value* must be followed by a **_THEN_** *single_value_format*.
4. **_ELSEIF_ op** *comparison_value* must be followed by a **_THEN_** *single_value_format*.
5. It is compulsory to have an **_ELSE_** at the end of the **Expression**.
6. An **Expression** is only for **<>** values and **not** <x>, <y> etc.

15.5.7 Multiple Text Formats

You can have just the one Text Format or multiple text formats in the dimension or leader.

That is, you can have

*pre_text***<***value_format_1***>***text_1<value_format_2>***…***text_n<value_format_n>*

where **pre_text** is written before the value with value_format_1, **text_n** is written after the n'th <> and the value in the n'th <> has the format *value_format_n*.

For example, to show metres and International feet, use

<> m or <3.28083895> ft

or to show metres and US Survey feet, use

<> m or <3.280833333> ft

15.5.8 Typing t or T to Change the Text Format

When using many of the **Dimension** or **Leader** commands, a list of typed options is often displayed in the screen message area and this may include **forma(t)** or **customise (t)ext.**

example for Dimensions

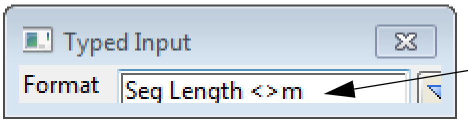
<[Pick start point] (a)ssociative (s)tyle[default] forma(t) [(z)ero offset OFF] > [picks][fast][Menu]

for Information Leaders

<Pick hook point or customise (t)ext> [picks][fast][Menu]

When this is the case, typing **t** or **T** is for changing the Text Format for the dimension or leader. The default value is the last one used.

After typing **t** or **T**, the *Format* **Typed Input** box is displayed showing the current text format of the selected dimension or leader.



format for a **Segment length** leader

Clicking on the Choice icon will bring up a list of *Text Formats* for the type of *dimension* or *leader* being worked on. This includes the **last six** used which is usefull when using the same format multiple times. See 15.5.9 Defaults File for Text Formats.

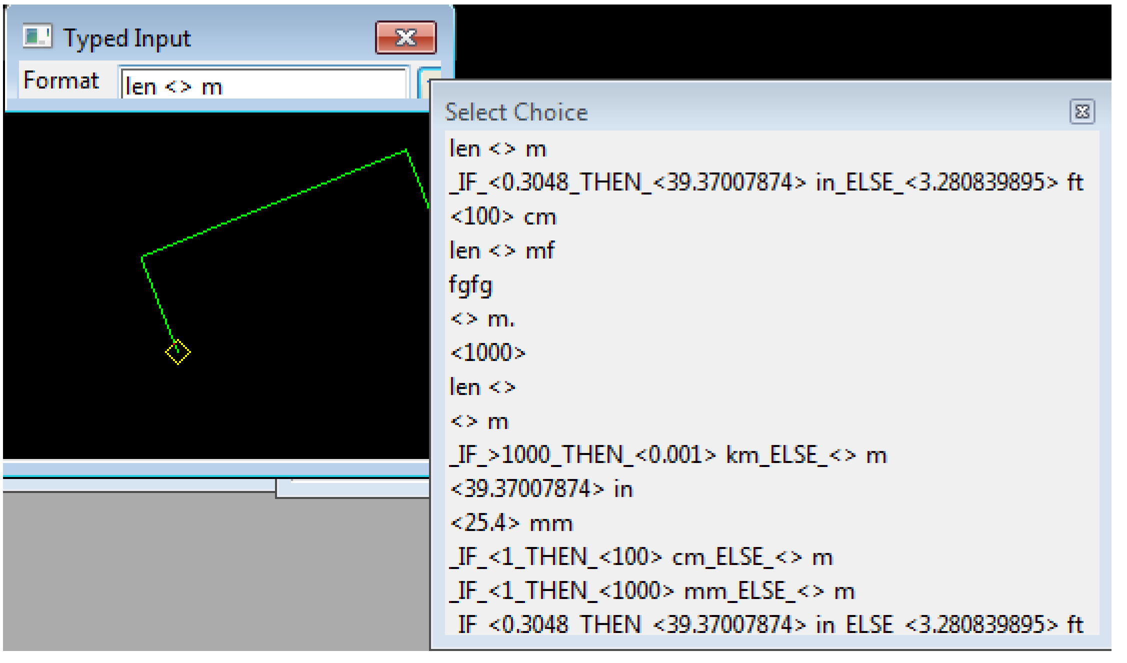
The new *Text Format* is picked from the pop up list, or typed into the box and **<Enter>** pressed.

The *Format* **Typed Input** box is then removed.

15.5.9 Defaults File for Text Formats

There is an XML file called ***draft_texts.xml*** which contains up to fifteen text formats for **each** *Dimension* type, and up to fifteen text formats for **each** type of *Information Leader*.

When you type **t** or **T** after seeing **(t)ext** included in the message in the screen message area, a *Format* **Typed Input** box is placed on the screen. The pop on the box up brings up a list of up to fifteen entries which come from the appropriate section of the ***draft_texts.xml*** file for the dimension or leader type being created or edited.



The first **six** of the fifteen in the pop up are the last six ones used in the current session of ***12d Model***, and the rest come from the *drafts_texts.xml* file.

So in a new session, all the entries in the pop up come from the ***drafts_texts.xml*** file. Each time one is selected from the list, or a new text format typed in, the selected or typed in text format is placed on the top of the list. If there is already six new entries on the top of the list, the sixth one is deleted and the first five are shuffled down.

As each of the six new entries are added to the top of the list, the bottom six that have come from the ***draft_texts.xml*** file are dropped off the list.

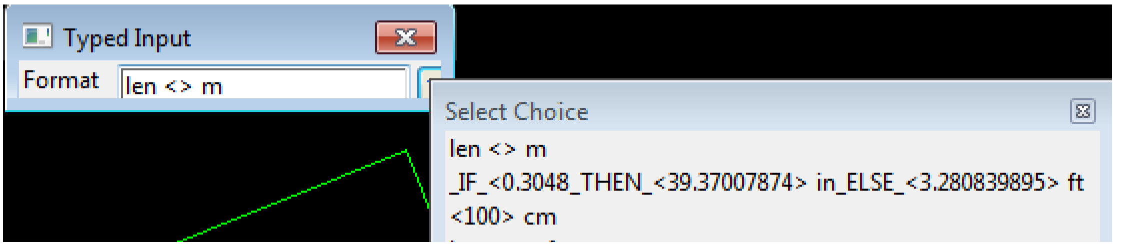
The ***draft_texts.xml*** file is read each time you open a ***12d Model*** project.

15.5.9.1 An Excerpt from a draft_text.xml File

The ***draft_text.xml*** file consists of a <meta_data> block at the top and then a <drafting> block with <Leader_Text> and <Dimension_Text> sections.

Note that because **<** and **>** are part of the XML syntax, in an XML file, less than (<) is written as **<** and greater than (>) is written as **>**.

An example of part of the *Dimension_Text* and *Leader_Text* sections of the ***drafts_texts.xml*** file, which includes the Aligned_Type dimension text formats for the first three entries in the pop up list.



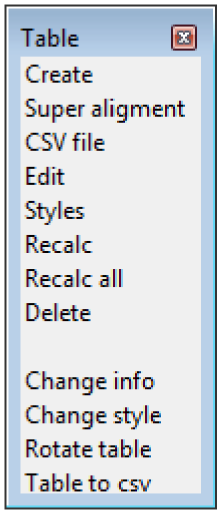
is:

```
<Dimension_Text>
    <Aligned_Type0>len &lt;&gt; m</Aligned_Type0>
    <Angular_Type0>&lt;&gt;</Angular_Type0>
    <Arc_Ang_Type0>&lt;&gt;</Arc_Ang_Type0>
    <Aligned_Type1>_IF_&lt;0.3048_THEN_&lt;39.37007874&gt; in_ELSE_&lt;3.280839895&gt; ft</Aligned_Type1>
    <Angular_Type1>&lt;&gt;</Angular_Type1>
    <Arc_Ang_Type1>&lt;&gt;</Arc_Ang_Type1>
    <Aligned_Type2>&lt;100&gt; cm</Aligned_Type2>
</Dimension_Text>
<Leader_Text>
    <Unknown_Type0>sadfsdfa\\nLine2</Unknown_Type0>
    <Area_Type0>Area &lt;&gt;</Area_Type0>
    <String_Length_3d_Type0>Length 3d &lt;&gt;m</String_Length_3d_Type0>
    <Unknown_Type1>sadfsdfa</Unknown_Type1>
    <Area_Type1>Area &lt;&gt;</Area_Type1>
    <String_Length_3d_Type1>Length 3d &lt;&gt;m</String_Length_3d_Type1>
    <Unknown_Type2>sadf</Unknown_Type2>
    <Area_Type2>Area &lt;&gt;</Area_Type2>
    <String_Length_3d_Type2>Length 3d &lt;&gt;m</String_Length_3d_Type2>
</Leader_Text>
```

15.6 CAD Table

Position of option on menu: **CAD =>Table**

The **CAD Table** options create and edit tables.



create a table
create a table for a super alignment
create a table for a CSV file
edit a table
create/edit table styles
recalc a selected table
recalc all tables
delete a table

change style of a table
rotate a table
convert a table to a CSV file

A user table can be created but also tables from super alignments or CSV files that can automatically update with changes in the super alignment or CSV file.

See

15.6.1 Create Table
15.6.2 Create Table from a Super Alignment
15.6.3 Create Table from a CSV File
15.6.4 Table Edit
15.6.5 Table Styles

Recalc - select a table to recalc.

Recalc all - recalcs all tables.

Delete - select a table to delete.

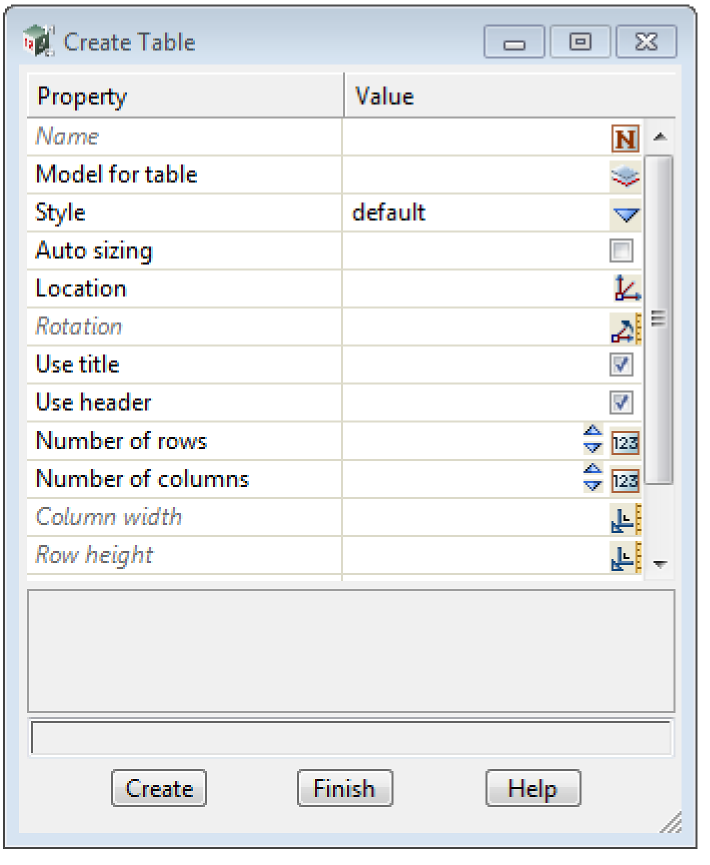
15.6.6 Table to CSV

15.6.1 Create Table

Position of option on menu: **CAD =>Table =>Create**

Important Note - all the text in a table will be treated as World Units.

Selecting **Create** brings up the **Create Table** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	name box		available names

the name for the table. This must not be blank.

Model for table	model box		available models

the model for the table.

Style	table style box		available table styles

the table style to use for this table.

Auto sizing	tick box	not ticked	

*if **ticked**, the table automatically resizes the column widths and heights to fit the text into the columns. If **not ticked**, the table will not resize the column widths and heights to fit the text into the column.*

Location	x y box	0	

*the (x,y) coordinates of the **top left hand corner** of the table.*

Rotation	angle box	0	

the rotation of the table - entered in hp notation. It is measured in a counter clockwise direction from the positive x axis.

Use title tick box ticked

*if **ticked**, the table has a title as the first row and it only consists of one column which goes from the left side of the table to the right side of the table. The textstyle information for text in the title is given in the* **Title style** *node of the Drafting Styles for Tables.*
*If **not ticked**, the table does not have a title at the top.*

Use header tick box ticked

*if **ticked**, each column has as its first entry a header. The textstyle information for text in the row of headers is given in the* **Header style** *node of the Drafting Styles for Tables*
*If **not ticked**, the table does not have a title at the top.*

Number of rows integer box

the number of rows in the table.

Number of columns integer box

the number of columns in the table.

Column width real box

if **Auto sizing** *is **no**, the width of each column.*

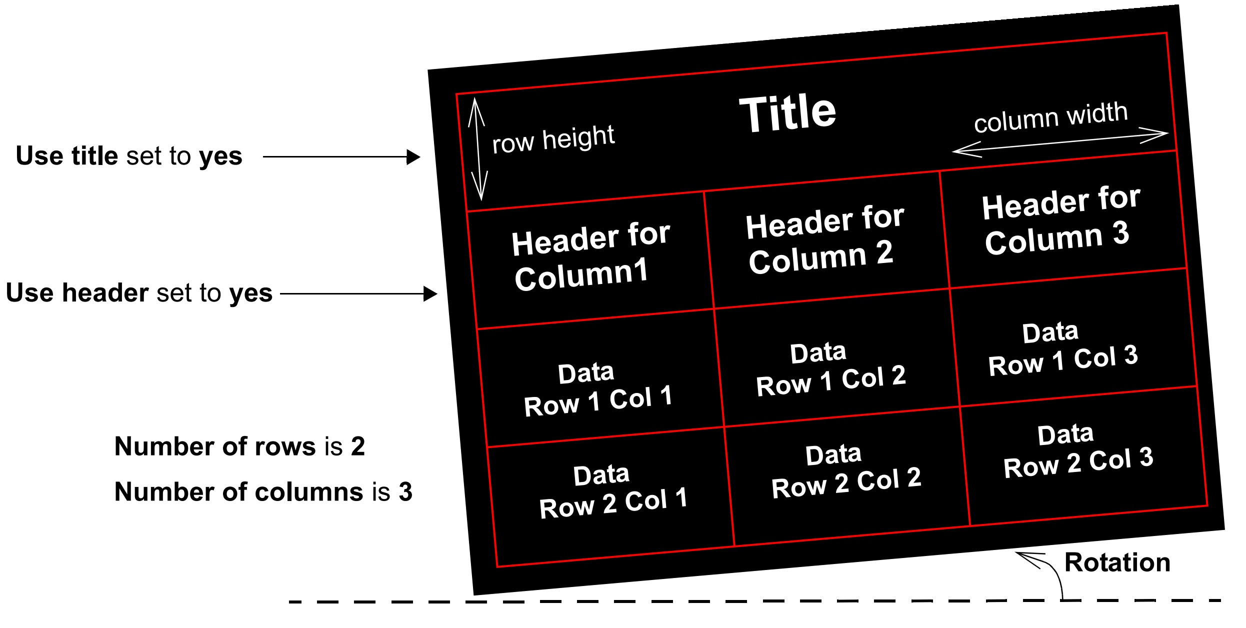
Row height real box

if **Auto sizing** *is **no**, the height of each column.*

Buttons at bottom

Create button

create the table with the properties in this panel at the location given in **Location**.



Continue to the next section 15.6.2 Create Table from a Super Alignment or return to 15.6 CAD Table.

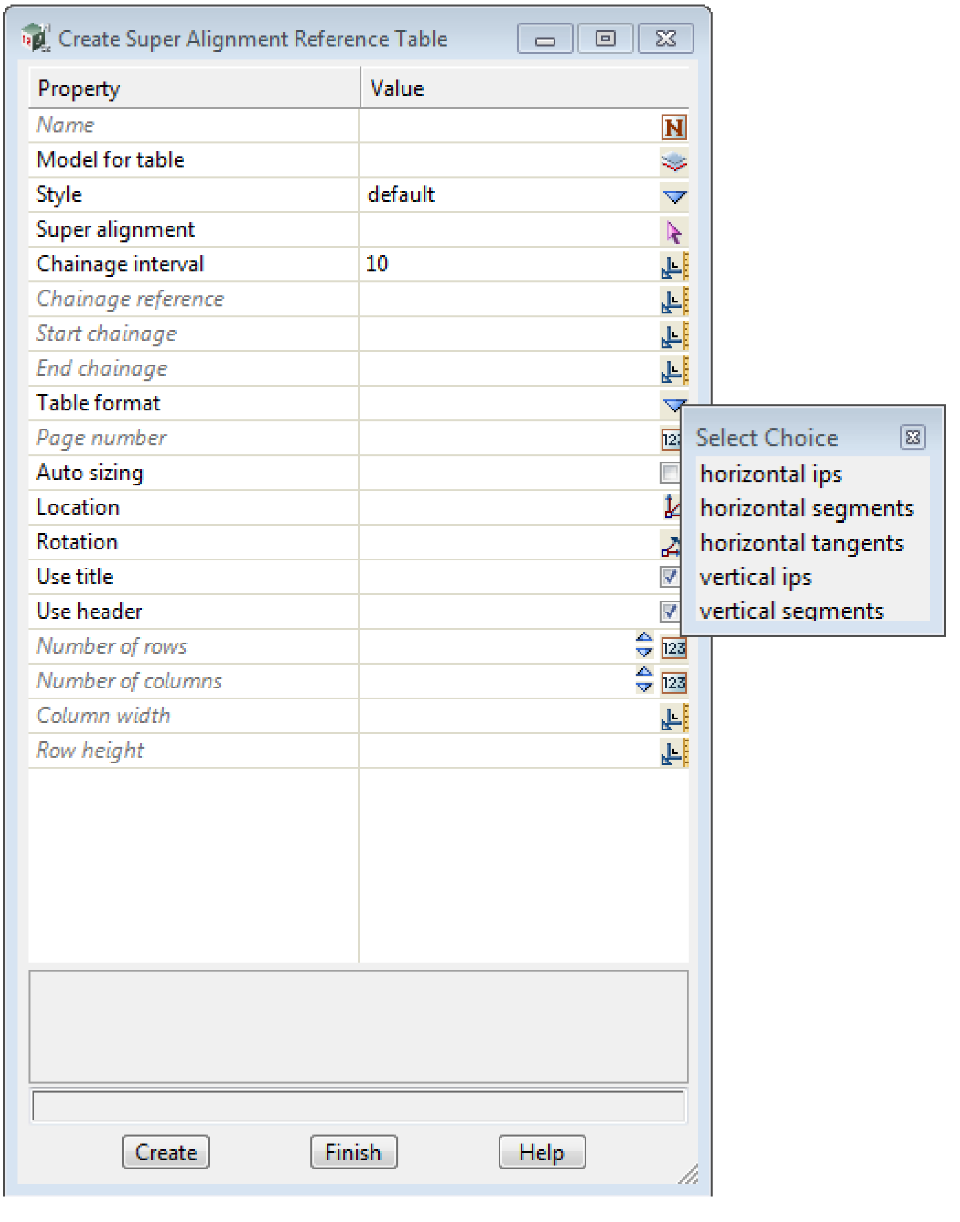
15.6.2 Create Table from a Super Alignment

Position of option on menu: **CAD =>Table =>Super alignment**

This table option is for creating a table of information about a selected ***super alignment***. The table is associated with the super alignment so that the table can be updated when the super alignment is modified.

Important Note - all the text in a table will be treated as World Units.

Selecting **Super alignment** brings up the **Create Super Alignment Reference Table** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	name box		available names

the name for the table. This must not be blank.

Field Description	Type	Defaults	Pop-Up
Model for table	model box		available models

the model for the table.

Style table style box available table styles

the table style to use for this table.

Super alignment string select box

select the super alignment to provide the information to generate the table.

Chainage interval real box

the interval to use if reporting information at chainage intervals down the super alignment.

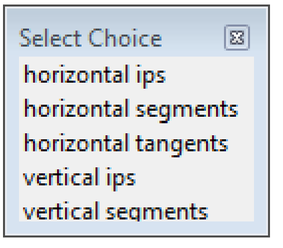
Chainage reference real box

the intervals for reporting are incremented about the Chainage reference value.

Start/End chainage choice box

*if **not blank**, the chainage to use for the Start/End chainages reported in the table.*
*If **blank**, the Start/End chainage of the super alignment is used.*

Table format choice box



*if **horizontal ips,** the table has information about the horizontal IP's for the super alignment, and includes columns for the horizontal IP number, the x, y, z coordinates of the HIP and the radius of any arc on the HIP.*

*If **horizontal segments**, the table has information on the vertices at the beginning of each segment along the horizontal geometry for the super alignment, and includes columns for the type of the segment that starts at that vertex, the x, y, z (level) coordinates of the vertex, the bearing of the segment coming into and leaving the vertex, and the radius of the segment coming into and leaving the vertex.*

*If **horizontal tangents**,*

*If **vertical ips**, the table has information about the vertical IP's for the super alignment, and includes columns for the vertical IP number, the chainage of the VIP and the height of the VIP.*

*If **vertical segments**, the table has information on the vertices at the beginning of each segment along the vertical geometry of the super alignment, and includes columns for the type of the segment that starts at that vertex, the chainage and z (level) coordinates of the vertex, the grade of the segment coming into and leaving the vertex, and the type of the segment coming into and leaving the vertex.*

Page number integer box

Auto sizing tick box not ticked

*if **ticked**, the table automatically resizes the column widths and heights to fit the text into the columns.*
*If **not ticked**, the table will not resize the column widths and heights to fit the text into the column.*

Location x y box 0

*the (x,y) coordinates of the **top left hand corner** of the table.*

Rotation angle box 0

the rotation of the table - entered in hp notation. It is measured in a counter clockwise direction from the positive x axis.

Use title tick box ticked

*if **ticked**, the table has a title as the first row and it only consists of one column which goes from the left side of the table to the right side of the table. The textstyle information for text in the title is given in the* **Title style** *node of the Drafting Styles for Tables.*

*If **not ticked**, the table does not have a title at the top.*

Use header tick box ticked

*if **ticked**, each column has as its first entry a header. The textstyle information for text in the row of headers is given in the* **Header style** *node of the Drafting Styles for Tables*
*If **not ticked**, the table does not have a title at the top.*

Number of rows integer box

the number of rows in the table.

Not used - the Table format and super alignment determines how many rows there are.

Number of columns integer box

the number of columns in the table.

Not used - the Table format and super alignment determines how many columns there are.

Column width real box

if **Auto sizing** *is **no**, the width of each column.*

Row height real box

if **Auto sizing** *is **no**, the height of each column.*

Buttons at bottom

Create button

create the table with the properties in this panel at the location given in **Location**.

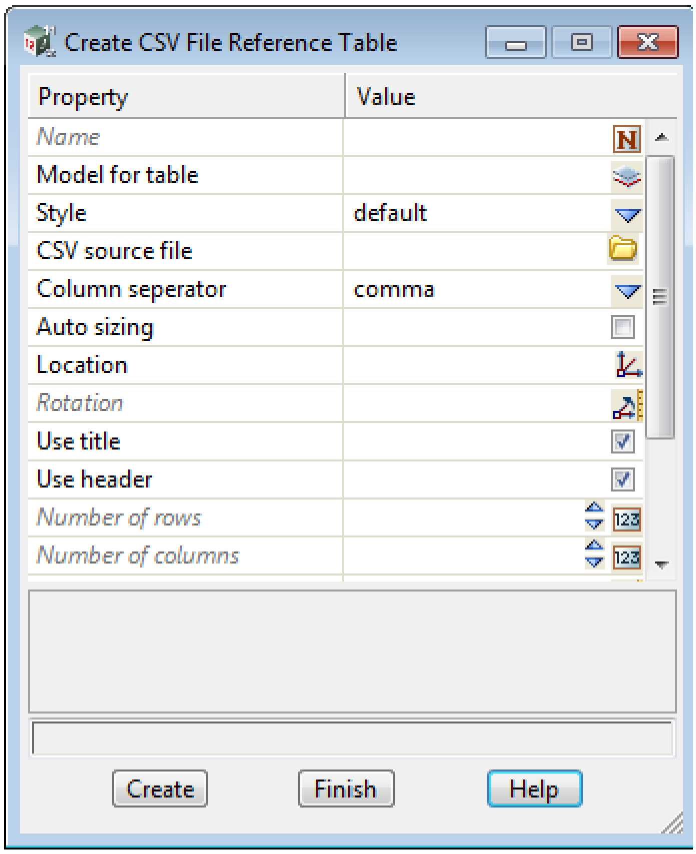
Continue to the next section 15.6.3 Create Table from a CSV File or return to 15.6 CAD Table.

15.6.3 Create Table from a CSV File

This table option is for creating a table of information from the information in a CSV file. The table is associated with the CSV file so that the table can be updated when the CSV file is modified.

Important Note - all the text in a table will be treated as World Units.

Selecting **CSV file** brings up the **Create CSV File Reference Table** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	name box		available names

the name for the table. This must not be blank.

Model for table	model box		available models

the model for the table.

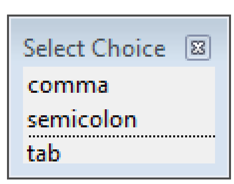
Style	table style box		available table styles

the table style to use for this table.

CSV source file	file box		

name of the CSV file to associate with the table.

Column separator	choice box		



*if **comma**, in the CSV file a comma (**,**) is used the separate the information for each column.*

*If **semicolon**, in the CSV file a semicolon (**;**) is used the separate the information for each column.*
*If **tab**, in the CSV file a tab is used the separate the information for each column.*

Auto sizing tick box not ticked

*if **ticked**, the table automatically resizes the column widths and heights to fit the text into the columns.*
*If **not ticked**, the table will not resize the column widths and heights to fit the text into the column.*

Location x y box 0

*the (x,y) coordinates of the **top left hand corner** of the table.*

Rotation angle box 0

the rotation of the table - entered in hp notation. It is measured in a counter clockwise direction from the positive x axis.

Use title tick box

*if **ticked** then the first line of the CSV file is taken as the titles for each column in the table. The textstyle information for text in the title is given in the* **Title style** *node of the Drafting Styles for Tables.*

*If **not ticked**, the table does not have a title.*

Use header tick box

*if **ticked** then the next line of the CSV file is taken to be the Headers for each column of the table. The textstyle information for text in the row of headers is given in the* **Header style** *node of the Drafting Styles for Tables*

*If **not ticked**, the table does not have a header.*

Number of rows integer box

the number of rows in the table.

Not used - the Table format and super alignment determines how many rows there are.

Number of columns integer box

the number of columns in the table.

Not used - the Table format and super alignment determines how many columns there are.

Column width real box

if **Auto sizing** *is **no**, the width of each column.*

Row height real box

if **Auto sizing** *is **no**, the height of each column.*

Buttons at bottom

Create button

create the table with the properties in this panel at the location given in **Location**.

Important Notes

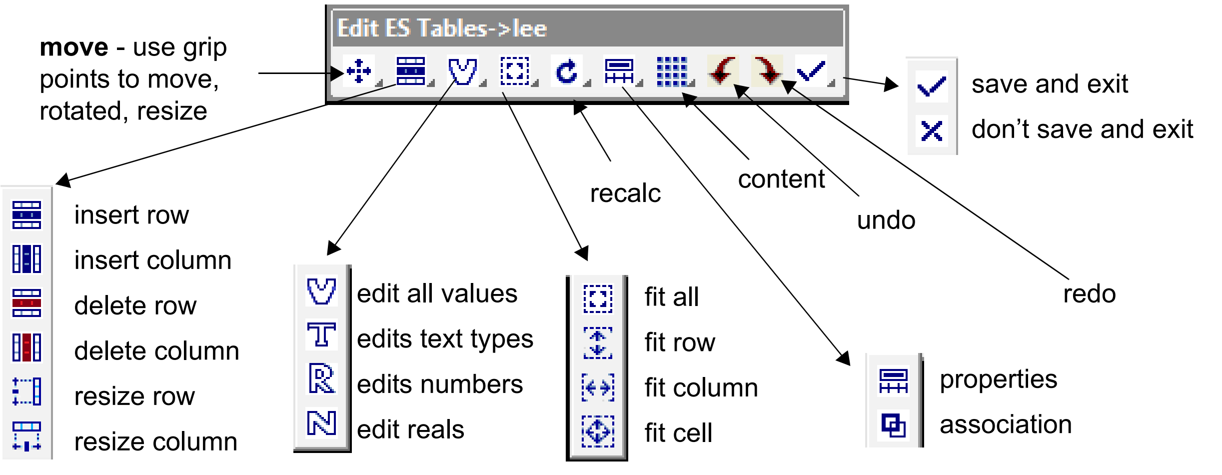
1. When a **Recalc** is done, the CSV file is reread and the information n the table updated.
2. **Auto size** in the table Properties must be ticked on for the new table to be automatically resized.
3. If you don't want the CSV file to be reread on a **Recalc**, you need to set **Type** back to **no reference** in the **Table Association** panel.

Continue to the next section 15.6.4 Table Edit or return to 15.6 CAD Table.

15.6.4 Table Edit

Picking an existing table will bring up the **Table** editor, and at the same time the grip points and point are displayed on the plan views that the table is on (*see* 15.6.4.1 Move).

Important Note - all the text in a table will be treated as World Units.



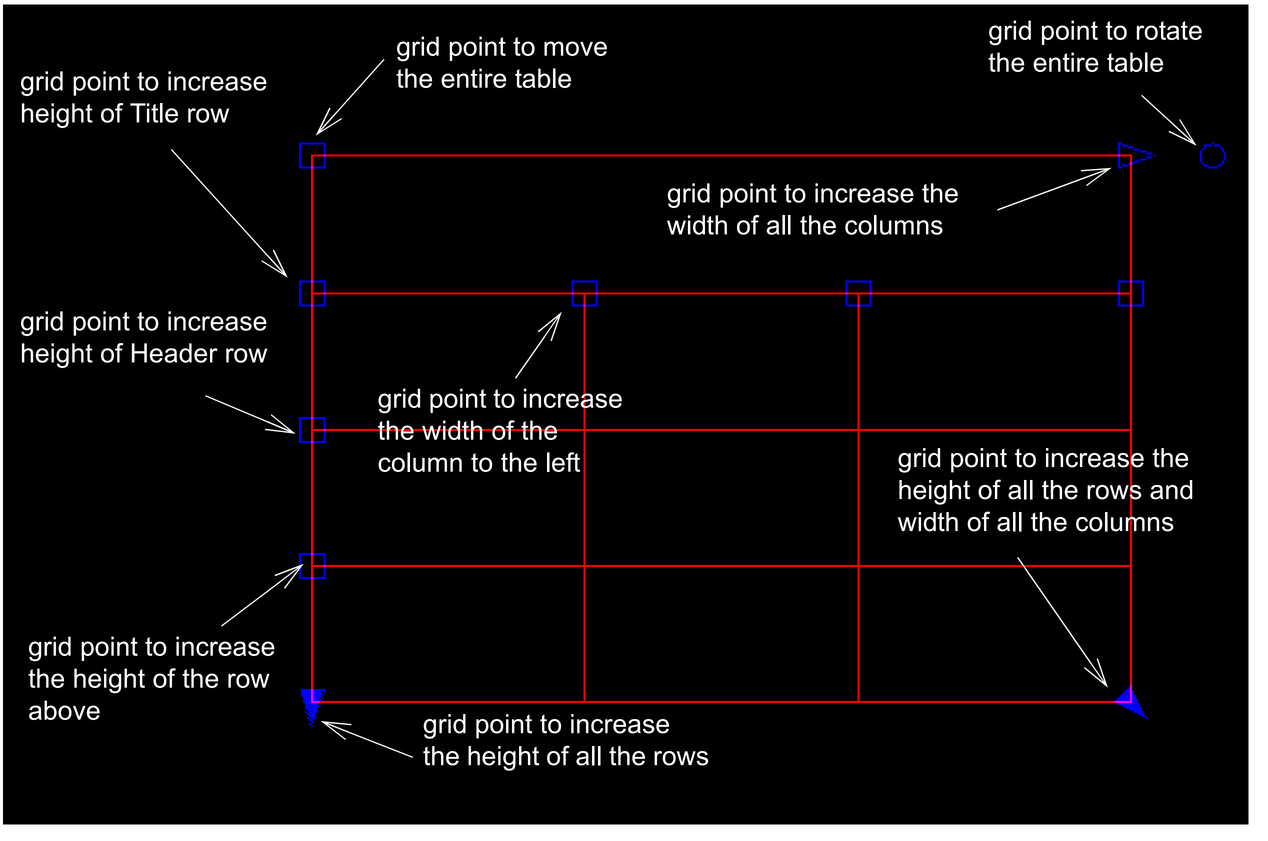
See

- 15.6.4.1 Move
- 15.6.4.2 Column and Row Edits
- 15.6.4.3 Text and Values Edit
- 15.6.4.4.1 Fit All
- 15.6.4.5 Recalc
- 15.6.4.6 Properties and Associations
- 15.6.4.7 Content

15.6.4.1 Move

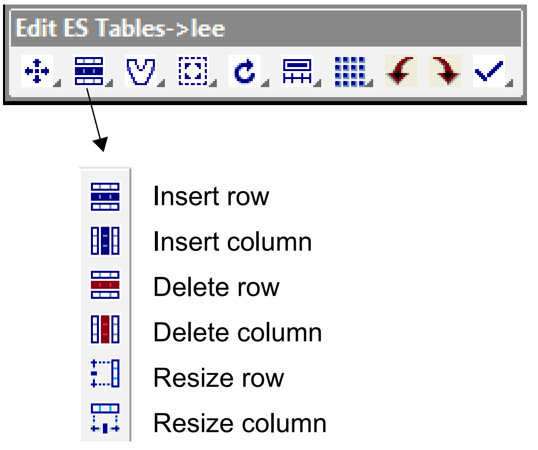
When the **Table Editor** is running, the grips points are displayed on any plan view that the table is on.

The use a grip point, you must first click on the **Move** icon on the **Table Editor** toolbar and then click and accept on the **grip** that you wish to use.



Continue to the next section 15.6.4.2 Column and Row Edits or return to 15.6.1 Create Table or to 15.6 CAD Table.

15.6.4.2 Column and Row Edits



See

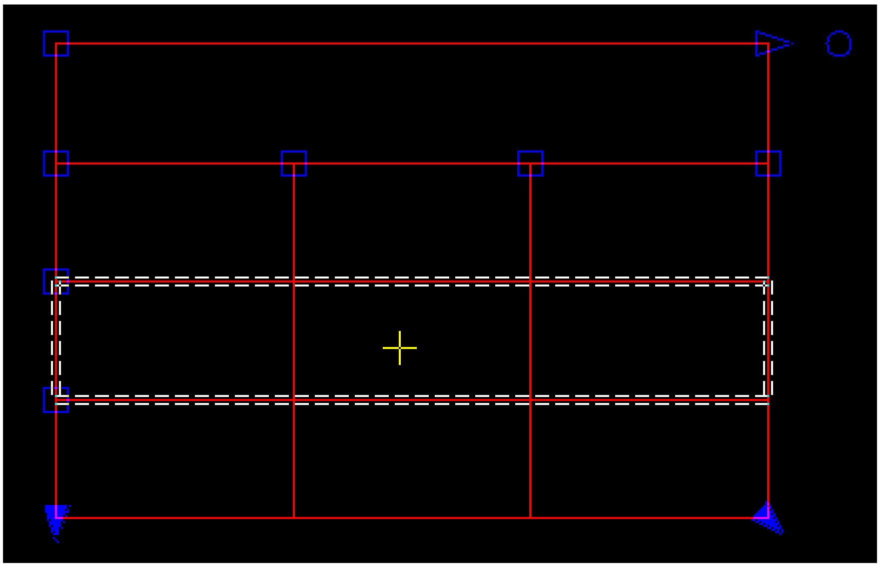
[15.6.4.2.1 Insert Row](#)
[15.6.4.2.2 Insert Column](#)
[15.6.4.2.3 Delete Row](#)
[15.6.4.2.4 Delete Column](#)
[15.6.4.2.5 Resize Row](#)
[15.6.4.2.6 Resize Column](#)

15.6.4.2.1 Insert Row

Click on the **Insert row** icon and the following message will appear in the screen message area.

<select row to insert (a)bove, (b)elow or a(p)pend> [picks][fast][Menu]

Then move over the table until the row you want to insert a row above or below highlights.

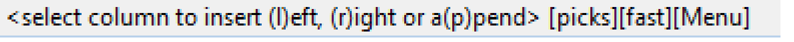


Press **a** or **A** to insert above, or **b** or **B** to insert below, the highlighted row.

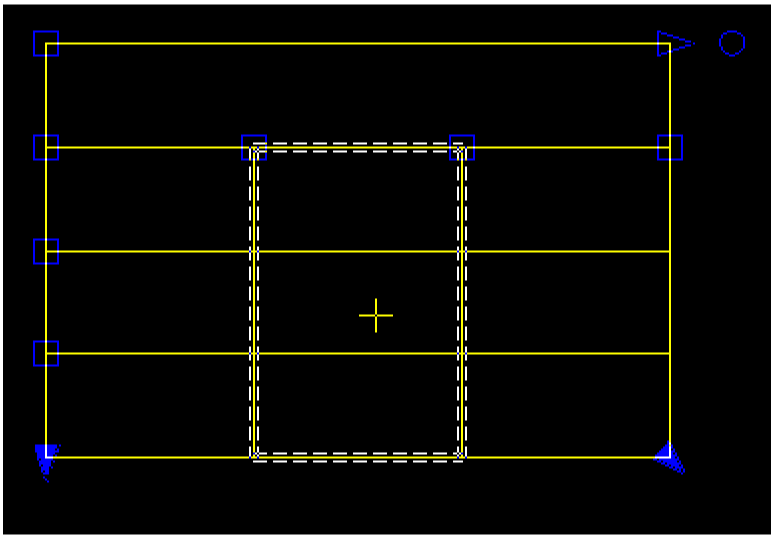
Continue to the next section 15.6.4.2.2 Insert Column or return to 15.6.4.2 Column and Row Edits or 15.6.1 Create Table or 15.6 CAD Table.

15.6.4.2.2 Insert Column

Click on the **Insert column** icon and the following message will appear in the screen message area.



Then move over the table until the column you want to insert a column either to the left or right of highlights.



Press **l** or **L**, to insert to the left, or **r** or R to insert to the right, of the highlighted column.

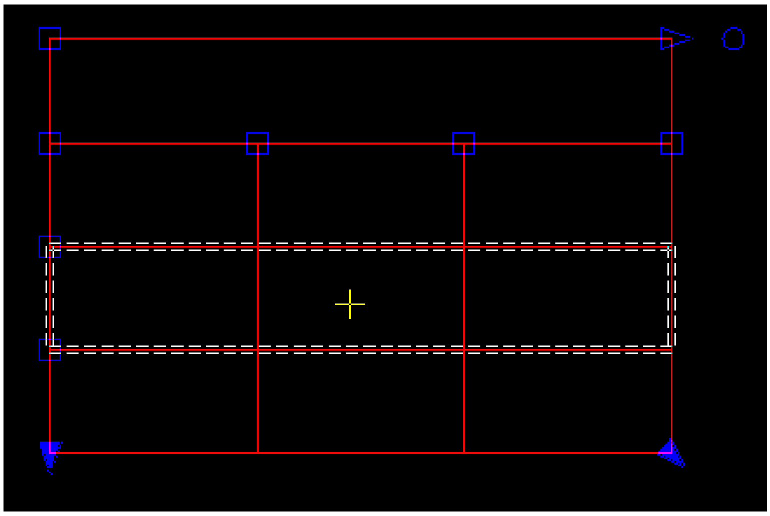
Continue to the next section 15.6.4.2.3 Delete Row or return to 15.6.4.2 Column and Row Edits or 15.6.1 Create Table or 15.6 CAD Table.

15.6.4.2.3 Delete Row

Click on the **Delete row** icon and the following message will appear in the screen message area.

<select row to (d)elete> [picks][accepts][Menu] Cursor-position

Then move over the table until the row you want to delete highlights



Press **d** or **D**, or pick and accept, to delete the highlighted row.

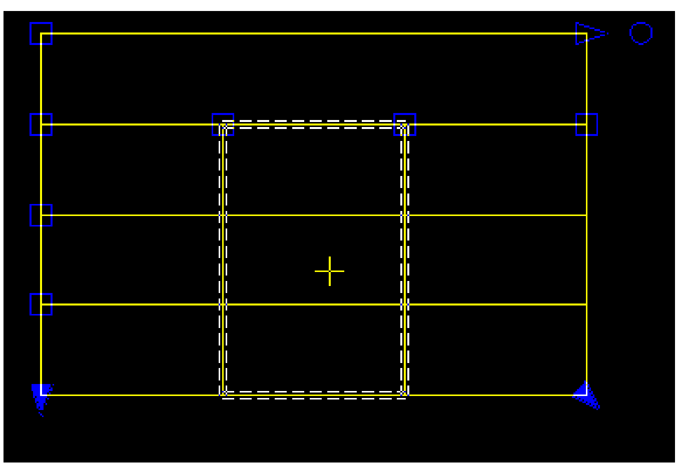
Continue to the next section 15.6.4.2.4 Delete Column or return to 15.6.4.2 Column and Row Edits or 15.6.1 Create Table or 15.6 CAD Table.

15.6.4.2.4 Delete Column

Click on the **Delete column** icon and the following message will appear in the screen message area.

<select column to (d)elete> [picks][accepts][Menu] Cursor-position

Then move over the table until the column you want to delete highlights

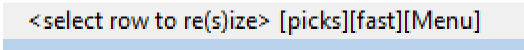


Press **d** or **D**, or pick and accept, to delete the highlighted column.

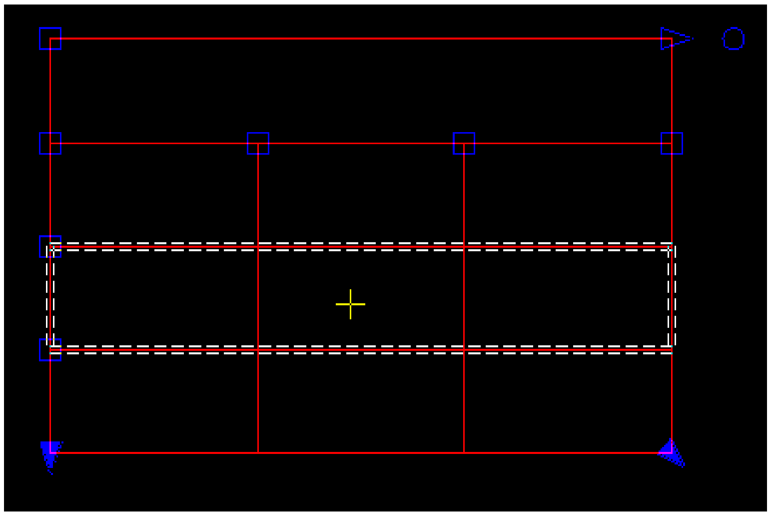
Continue to the next section 15.6.4.2.5 Resize Row or return to 15.6.4.2 Column and Row Edits or 15.6.1 Create Table or 15.6 CAD Table.

15.6.4.2.5 Resize Row

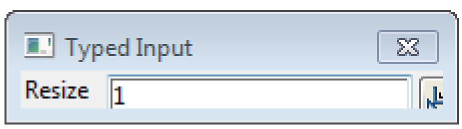
Click on the **Resize row** icon and the following message will appear in the screen message area.



Then move over the table until the row you want to resize highlights.



Press **s** or **S**, or pick and accept, and the *Resize* **Typed Input** box will appear with the current size of the row in it.

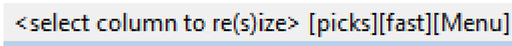


Type the new size into the box and then press <Enter>. The box will disappear and the row given the new size value.

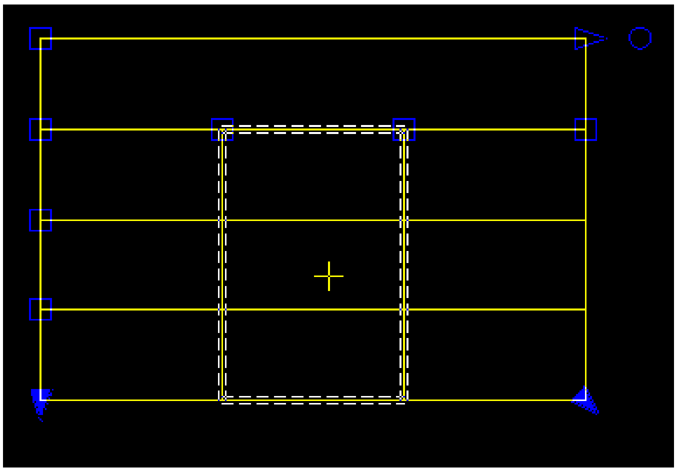
Continue to the next section 15.6.4.2.6 Resize Column or return to 15.6.4.2 Column and Row Edits or 15.6.1 Create Table or 15.6 CAD Table.

15.6.4.2.6 Resize Column

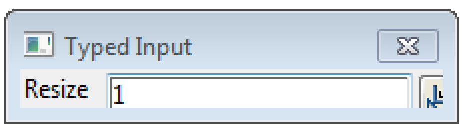
Click on the **Resize column** icon and the following message will appear in the screen message area.



Then move over the table until the column you want to resize highlights.



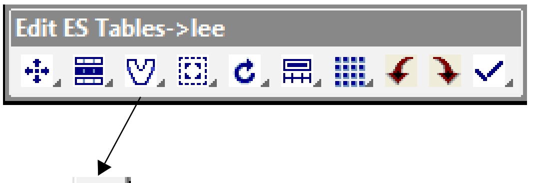
Press **s** or **S**, or pick and accept, and the *Resize* **Typed Input** box will appear with the current size of the selected column in it.

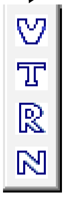


Type the new size into the box and then press <Enter>. The box will disappear and the column given the new size value.

Return to 15.6.4.2 Column and Row Edits or 15.6.1 Create Table or 15.6 CAD Table.

15.6.4.3 Text and Values Edit





create & edit all types of values

create and edit text

create and edit reals

create and edit numbers

See

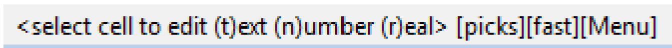
[15.6.4.3.1 Value Create and Edit]
[15.6.4.3.2 Text Create and Edit]
[15.6.4.3.3 Real Create and Edit]
[15.6.4.3.3 Real Create and Edit]

15.6.4.3.1 Value Create and Edit

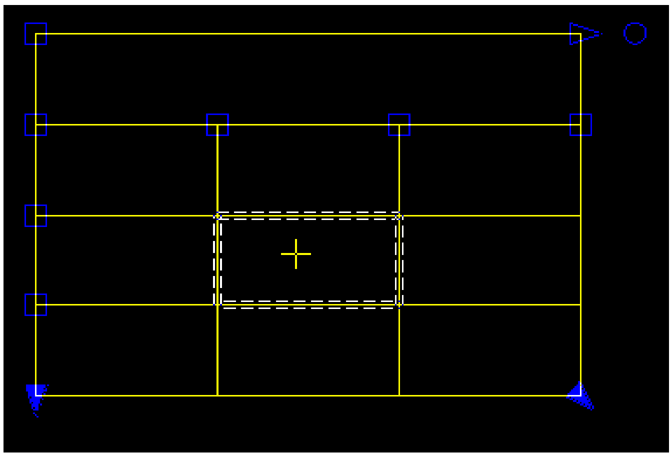
For an empty cell, **Value** can create either a text, number or real value.

If used on a non empty cell, the cell contents are displayed in a Typed input box or type Text, Number of Real depending on the type of the cell contents.

Click on the **Value** icon and the following message appears in the screen message area.

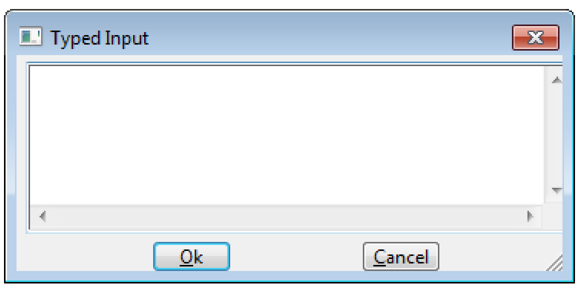


Move over the table until the cell you want to edit or create a value, highlights.

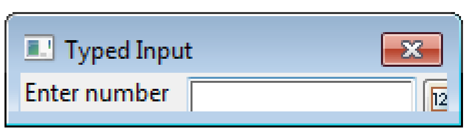


If the cell is **empty**,

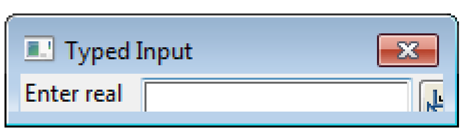
Type **t** or **T**, or pick and accept inside the cell, and the *Enter text* **Typed Input** box will appear.



Type **n** or **N**, and the *Enter number* **Typed Input** box will appear.



Type **r** or **R**, and the *Enter real* **Typed Input** box will appear.

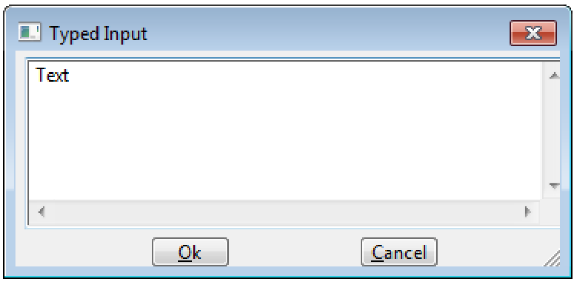


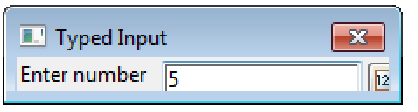
Type the new text, number or real into the box and press <Enter>.

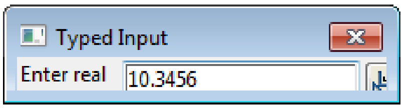
The box will disappear and the new text, number or real placed in the cell, and the cell given the appropriate type.

Note: If the type is **Real** then the actual typed in value is kept in the cell BUT it is displayed with the number of decimal places given in the Table style.

If the cell is **not** empty, then the cell value is displayed in a **Typed Input** box of the **same type** as the **cell contents**.







Type the new **Text**, **Number** or **Real** into the **Typed Input** box and press <Enter>.

The box will disappear and the new value placed in the cell.

Note: If the type is **Real** then the actual typed in value is kept in the cell BUT it is displayed with the number of decimal places given in the Table style.

Continue to the next section 15.6.4.3.2 Text Create and Edit or return to 15.6.4.3 Text and Values Edit or 15.6.1 Create Table or 15.6 CAD Table.

15.6.4.3.2 Text Create and Edit

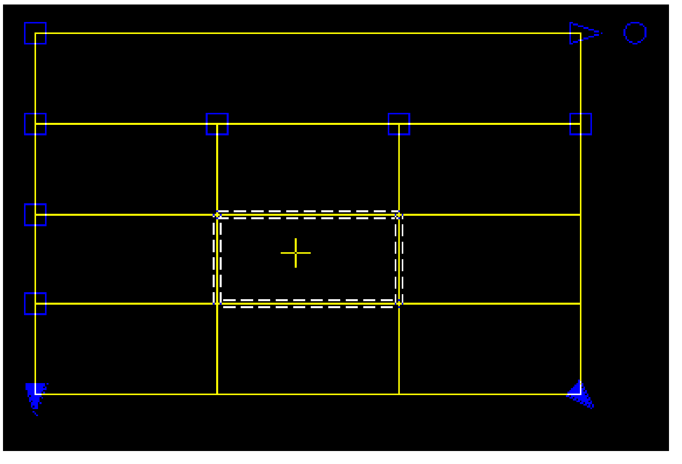
Important Note - all the text in a table will be treated as World Units.

For an empty cell, **Text** creates and displays one or more lines of text.

If used on a non empty cell, the cell contents are displayed in a Text box, and when saved, is written into the cell as a **Text** type

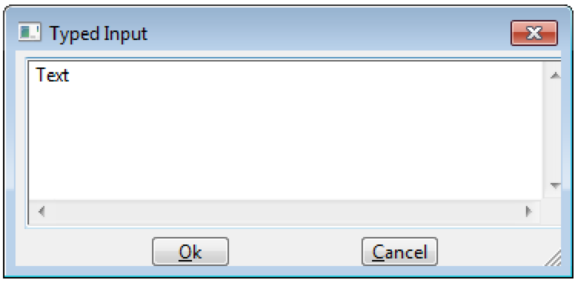
Click on the **Text** icon and the following message will appear in the screen message area.

Move over the table until the cell you want to edit or create a Text, highlights.



Type **t** or **T**, or pick and accept inside the cell, and the *Enter text* **Typed Input** box will appear.

If the cell was empty then nothing is in the *Enter text* box but if the cell is not empty, the contents of the cell are displayed as text.



Type the new text into the *Enter text* box and press <Enter>.

The box will disappear and the new text placed in the cell, and the cell given type **Text**.

Continue to the next section 15.6.4.3.3 Real Create and Edit or return to 15.6.4.3 Text and Values Edit or 15.6.1 Create Table or 15.6 CAD Table.

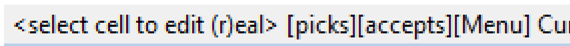
15.6.4.3.3 Real Create and Edit

For an empty cell, **Real** creates and displays a real with the number of decimal places used in the display is taken from the Table style.

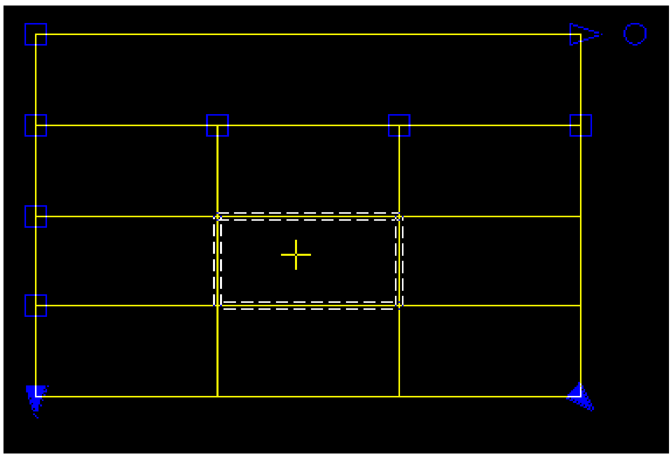
If used on a non empty cell, the cell contents are converted into a real and displayed in a Real box for editing, and when saved, is written into the cell as a **Real** type.

It the cell contains text that can't be converted to a real then the Real box is left empty.

Click on the **Real** icon and the following message will appear in the screen message area.

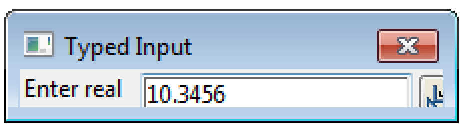


Move over the table until the cell you want to edit or create a Number, highlights.



Type **r** or **R**, or pick and accept inside the cell, and the *Enter real* **Typed Input** box will appear.

If the cell was empty then nothing is in the *Enter real* box but if the cell is not empty, the contents of the cell are displayed a Real, or nothing if the contents can't convert to a real.



Type the new real into the *Enter real* box and press <Enter>.

The box will disappear and the new real placed in the cell and displayed with the number of decimal places given in the Table style. The cell given type **Real**.

Note: If the type is **Real** then the actual typed in value is kept in the cell BUT it is displayed with the number of decimal places given in the Table style.

Continue to the next section 15.6.4.3.4 Number Create and Edit or return to 15.6.4.3 Text and Values Edit or 15.6.1 Create Table or 15.6 CAD Table.

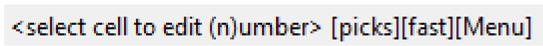
15.6.4.3.4 Number Create and Edit

For an empty cell, **Number** creates and displays a number (a positive or negative integer).

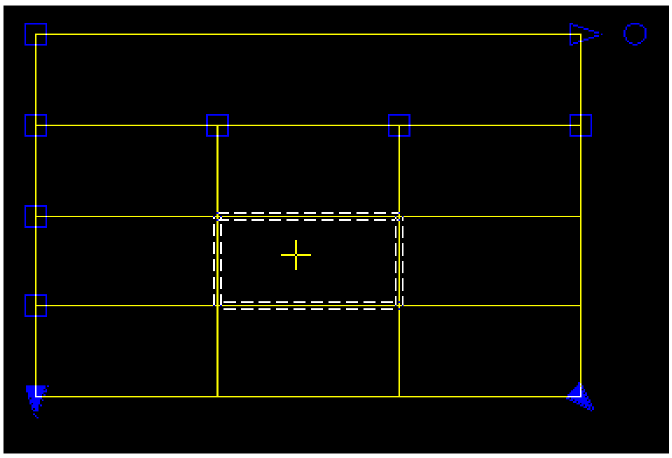
If used on a non empty cell, the cell contents are converted into a number and displayed in a number box for editing, and when saved, is written into the cell as a **Number** type.

It the cell contains text that can’t be converted to a number then the Number box is left empty.

Click on the **Number** icon and the following message will appear in the screen message area.

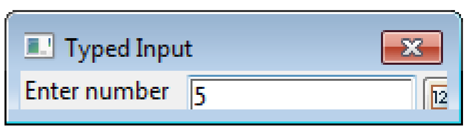


Move over the table until the cell you want to edit or create a Number, highlights.



Type **n** or **N**, or pick and accept inside the cell, and the *Enter number* **Typed Input** box will appear.

If the cell was empty then nothing is in the *Enter number* box but if the cell is not empty, the contents of the cell are displayed a Number, or nothing if the contents can’t convert to a number.

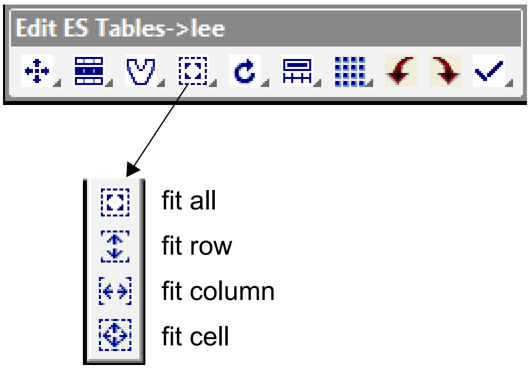


Type the new number into the *Enter number* box and press <Enter>.

The box will disappear and the new number placed in the cell, and the cell given type **Number**.

Return to 15.6.4.3 Text and Values Edit or 15.6.1 Create Table or 15.6 CAD Table.

15.6.4.4 Fit



See

- 15.6.4.4.1 Fit All
- 15.6.4.4.2 Fit Row
- 15.6.4.4.3 Fit Column
- 15.6.4.4.4 Fit Cell

15.6.4.4.1 Fit All

Fit goes though each cell and resizes it both vertically and horizontally so that the text fits inside the cell.

It then looks at each row and makes all the cells in the same row as high as the highest one in the row.

It then looks at each column and makes all the cells in the same column as wide as the widest one in the column.

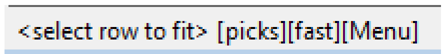
Fit runs even if **Auto sizing** is set to **no**.

Continue to the next section 15.6.4.4.2 Fit Row or return to 15.6.4.4 Fit or 15.6.1 Create Table or 15.6 CAD Table.

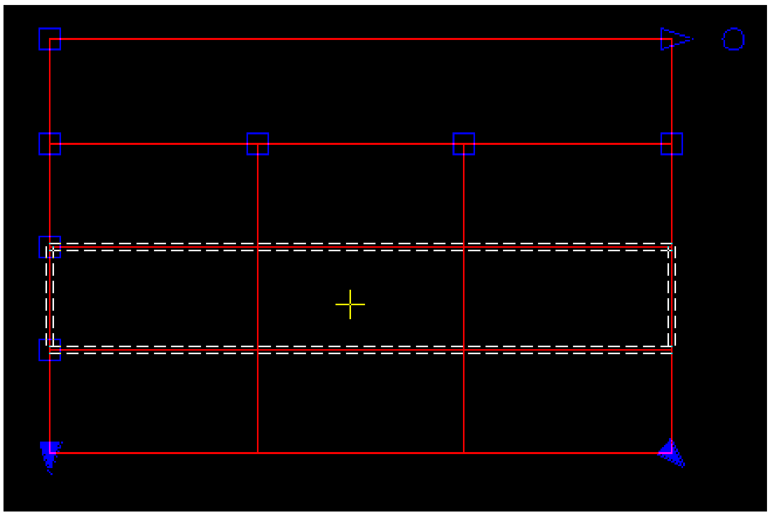
15.6.4.4.2 Fit Row

Fit row goes though each cell in the selected row to find the maximum height required to fit the text vertically in the cell, and then makes the height of the row the maximum of all the heights required for each cell. That is, it makes the row high enough to fit the text vertically in every cell in the row.

Click on the **Fit row** icon and the following message will appear in the screen message area.



Then move over the table until the row you want to fit highlights.



Accepting makes the row high enough to fit any text in the row.

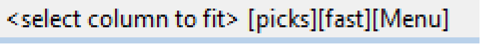
Fit row runs even if **Auto sizing** is set to **no**.

Continue to the next section 15.6.4.4.3 Fit Column or return to 15.6.4.4 Fit or 15.6.1 Create Table or 15.6 CAD Table.

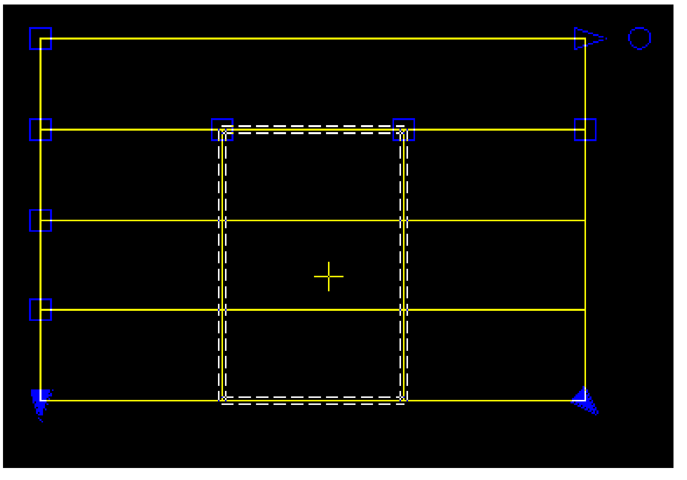
15.6.4.4.3 Fit Column

Fit column goes though each cell in the selected column to find the maximum width required to fit the text horizontally in the cell, and then makes the width of the column the maximum of all the widths required for each cell. That is, it makes the column wide enough to fit the text horizontally in every cell in the column.

Click on the **Fit column** icon and the following message will appear in the screen message area.



Then move over the table until the column you want to fit highlights.



Accepting makes the column wide enough to fit any text in the column.

Fit row runs even if **Auto sizing** is set to **no**.

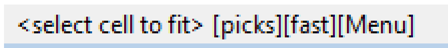
Continue to the next section 15.6.4.4.4 Fit Cell or return to 15.6.4.4 Fit or 15.6.1 Create Table or 15.6 CAD Table.

15.6.4.4.4 Fit Cell

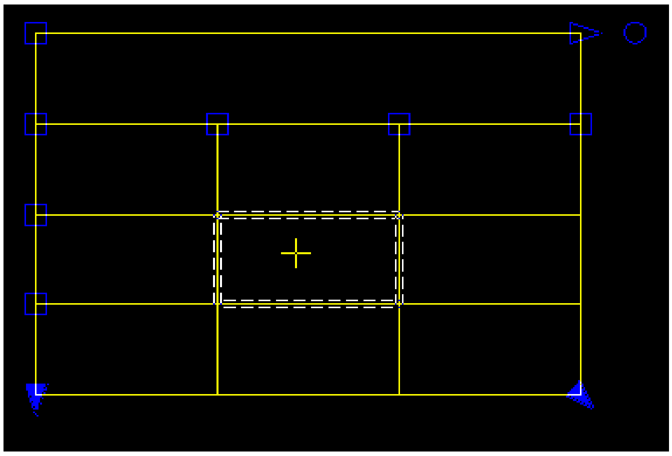
Fit cell increases the row height enough to fit the text vertically in the cell and also increases the column width enough to fit the text horizontally in the cell. That is, it makes the row and column wide enough to fit the text in the selected cell.

Note that the row height and column width will not be decreased, only increased

Click on the **Fit cell** icon and the following message will appear in the screen message area.



Then move over the table until the cell you want to fit highlights.

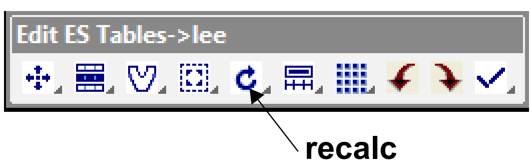


Accepting makes the column wide enough to fit any text in the column.

Fit cell runs if **Auto sizing** is set to **no**.

Return to 15.6.4.4 Fit or 15.6.1 Create Table or 15.6 CAD Table.

15.6.4.5 Recalc

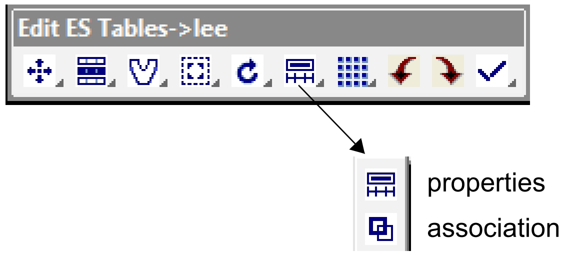


Sometime when the object is **associated** with the **Table** is modified, the **Table** does not update.

Recalc forces an update of all the information connected with the **Association** object.

Return to 15.6.1 Create Table or 15.6 CAD Table.

15.6.4.6 Properties and Associations

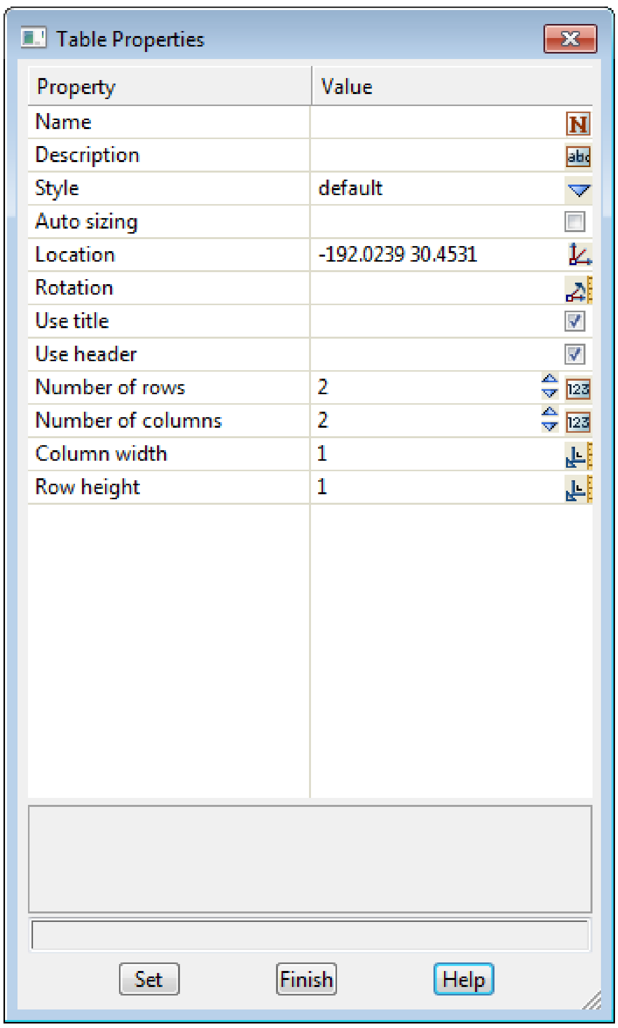


See

[15.6.4.6.1 Properties](#)
[15.6.4.6.2 Table Association](#)

15.6.4.6.1 Properties

Click on the **Properties** icon and the **Table Properties** panel comes up with the values for the current table in it.



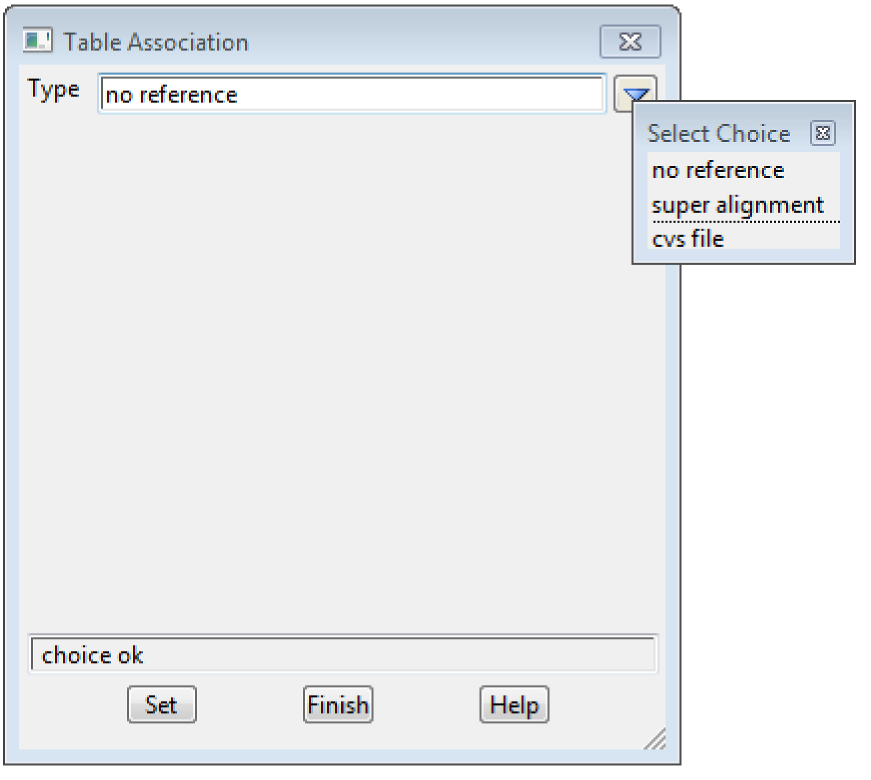
Change values in the panel and then click **Set** for them to become the tables new values.

Continue to the next section 15.6.4.6.2 Table Association or return to 15.6.4.6 Properties and Associations or 15.6.1 Create Table or 15.6 CAD Table.

15.6.4.6.2 Table Association

Click on the **Association** icon and the **Table Association** panel comes up.

If there is no Association for the table, one can be added with this option.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Type	choice box		no reference, super alignment csv file

if ***no reference****, then the table is not associated with anything.*

If ***super alignment****, read in and associate a CSV file with the table. See*[15.6.4.6.2.1 Associate a Super Alignment](#)

If ***CSV file****, read in and associate a CSV file with the table. See*[15.6.4.6.2.2 Associate a CSV File](#)

Set	button		

set the association for the table.

See

[15.6.4.6.2.1 Associate a Super Alignment](#)

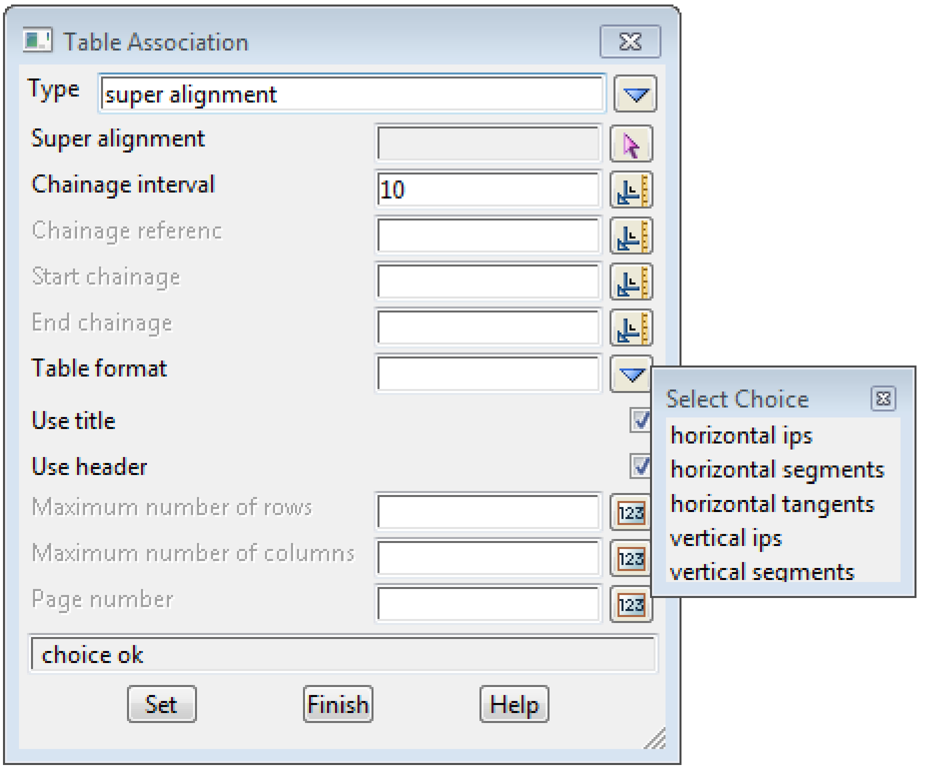
[15.6.4.6.2.2 Associate a CSV File](#)

Return to [15.6.1 Create Table](#) or [15.6 CAD Table](#).

15.6.4.6.2.1 Associate a Super Alignment

Choice ***super alignment*** is selected when a super alignment is to be associated with the table.

When ***super alignment*** is selected, the panel changes to display fields to enter information about the super alignment and how it is to be associated with the table.

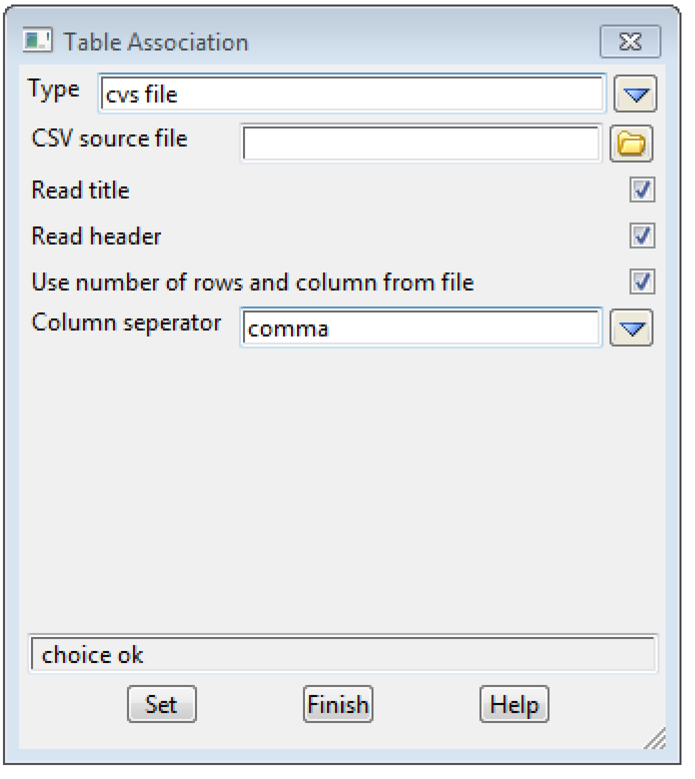


For information about the fields in this panel, see 15.6.2 Create Table from a Super Alignment

15.6.4.6.2.2 Associate a CSV File

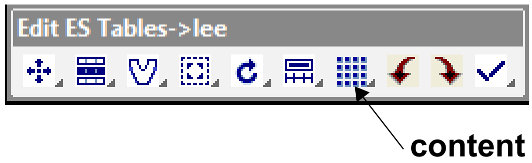
Choice ***CSV file*** is selected when a CSV file is to be associated with the table.

When ***CSV file*** is selected, the panel changes to display fields to enter for information about the CSV file and how it is to be associated with the table.

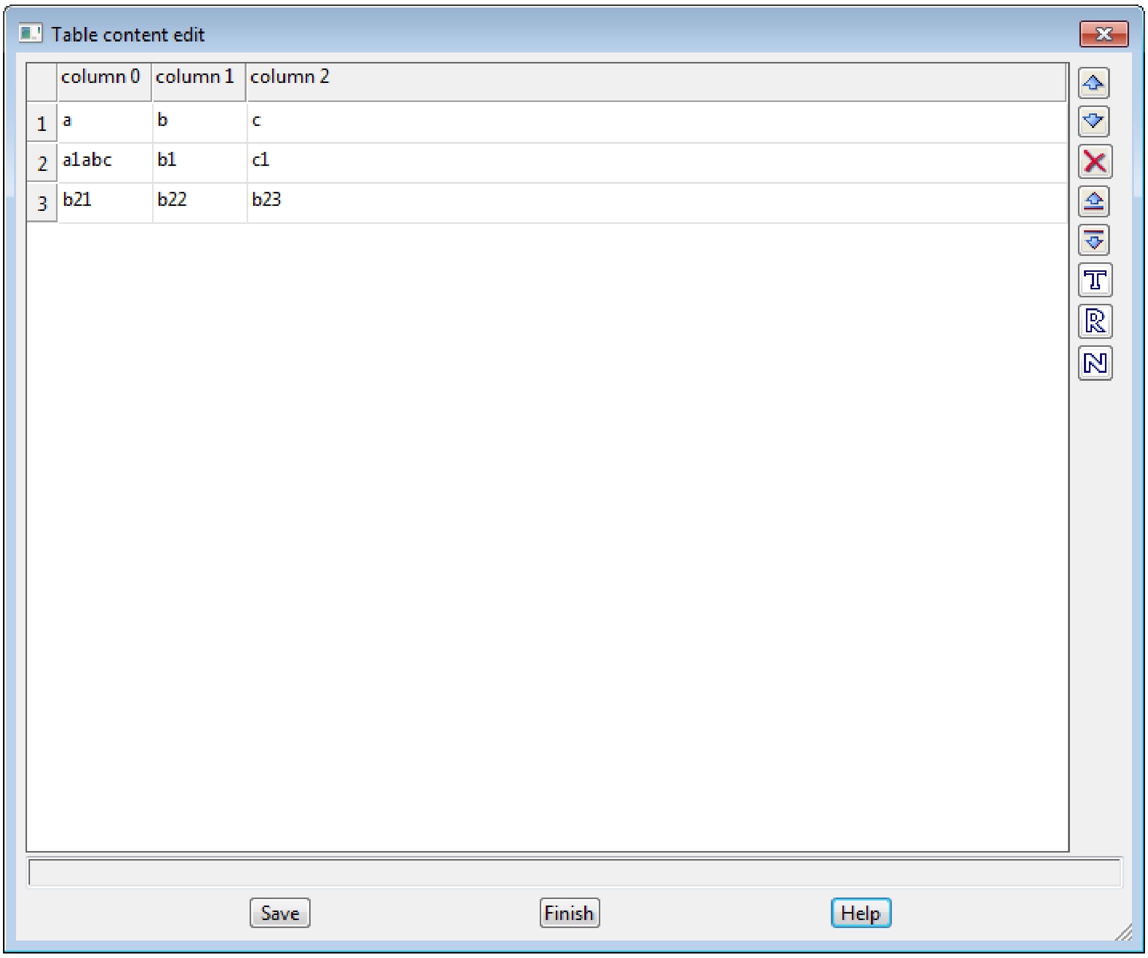


For information about the fields in this panel, see 15.6.3 Create Table from a CSV File

15.6.4.7 Content



Click on the **Content** icon and the **Table Content Edit** panel comes up with the values for the current table in it.



In the panel you can edit the values in each cell, add new values, add or delete rows or move rows up or down.

Finally click **Save** for the contents of the panel to become the new values in the table.

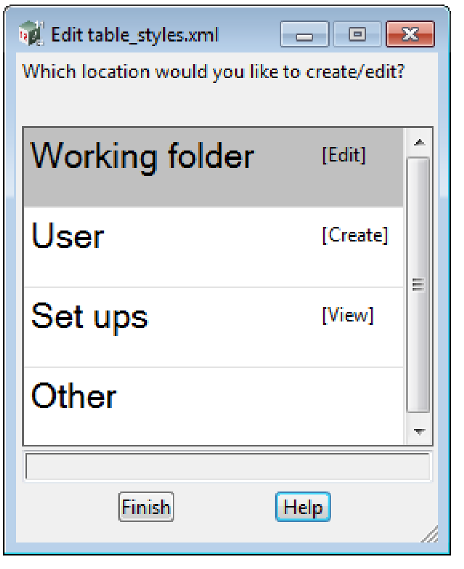
Return to 15.6.1 Create Table or 15.6 CAD Table.

15.6.5 Table Styles

Position of option on menu: **CAD =>Table =>Styles**

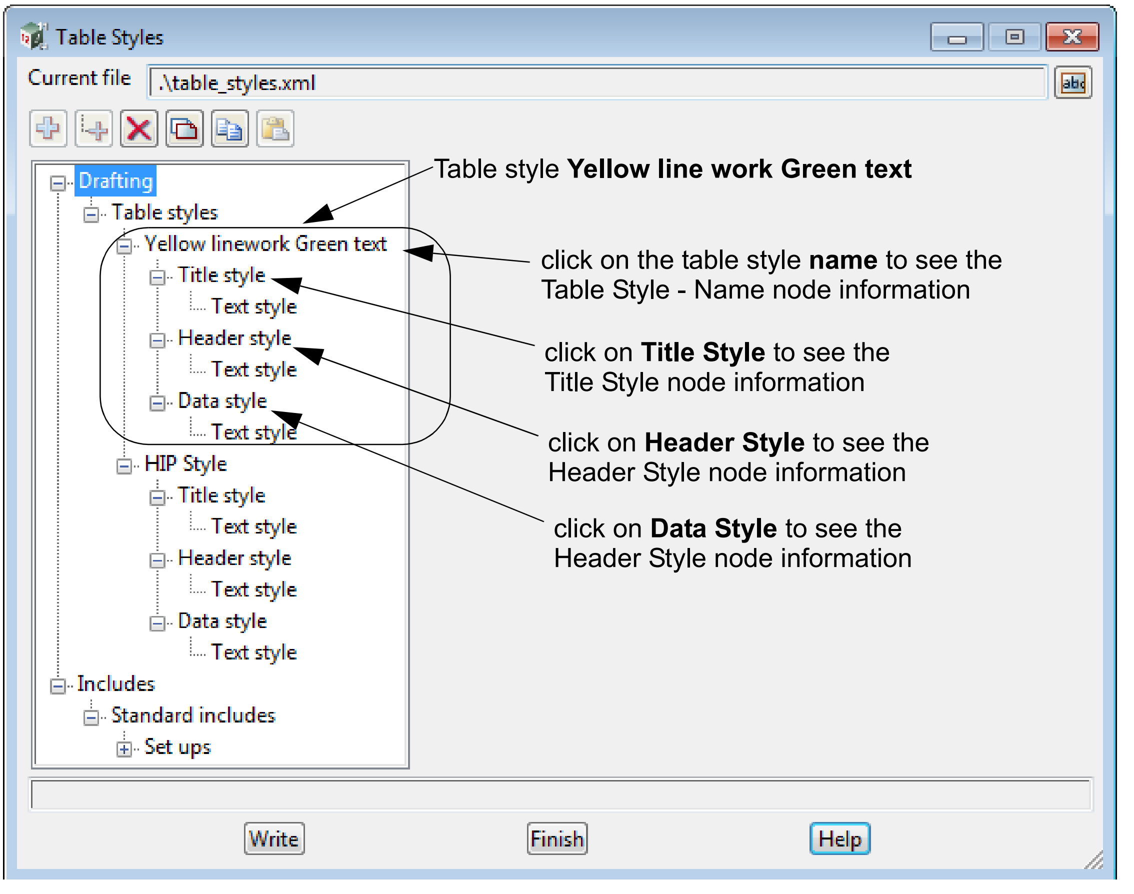
Important Note - all the text in a table will be treated as World Units.

When you click on the option an **Edit table_styles.xml** panel is brought up and the panel shows the standard areas for looking for the table_styles.XML files - Working folder, Customer (User), User, Set Ups and Other.



For information about where how to find and create table_styles.xml files, see 15.7 Style XML Files for Dimensions, Leaders & Tables.

Once a table_styles.xml has been opened, the **Table Styles** panel is displayed.

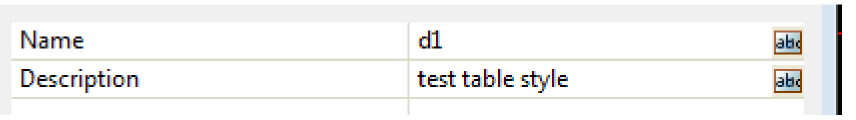


For the description of each of the nodes, see

15.6.5.1 Table Styles - Name Node
15.6.5.2 Table Styles - Title Style Node
15.6.5.3 Table Styles - Header Style Node
15.6.5.4 Table Styles - Data Style Node

15.6.5.1 Table Styles - Name Node

This is on the right hand side of the panel when you click on the ***Table Styles*** name.



Name text box

name of the table style.

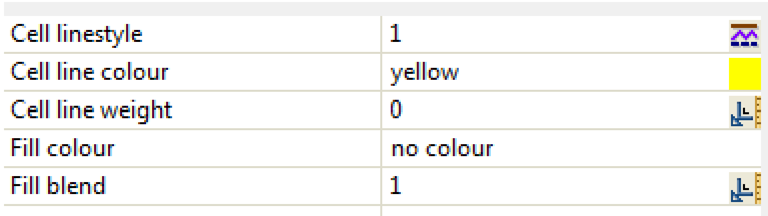
Description text box

description of the table style.

Continue to the next section 15.6.5.3.1 Table Styles - Header Style >Text Style Node or return to 15.6.5 Table Styles or 15.6.1 Create Table or 15.6 CAD Table.

15.6.5.2 Table Styles - Title Style Node

This is on the right hand side of the panel when you click on the ***Title style*** node.



Cell linestyle linestyle box available linestyles

linestyle for drawing the line work around the title row in the table.

Cell line colour colour box available colours

colour of the line work drawn around the title row.

Cell line weight real box

weight of the line work drawn around the title row.

Fill colour colour box available colours

colour to fill the title row.

Fill blend real box

blend for the fill colour for the title row.

To control the **text** in the Title row, see 15.6.5.3.1 Table Styles - Header Style >Text Style Node

15.6.5.2.1 Table Styles - Title Style > Text Style Node

Important Note - all the text in a table will be treated as World Units.

This is on the RHS of the panel when you click on the ***Table style*** **>Title style >Text style** node.

If **Use title** is **yes** in the tables **Properties** then this node controls the drawing of the text in the Title at the top of the table.

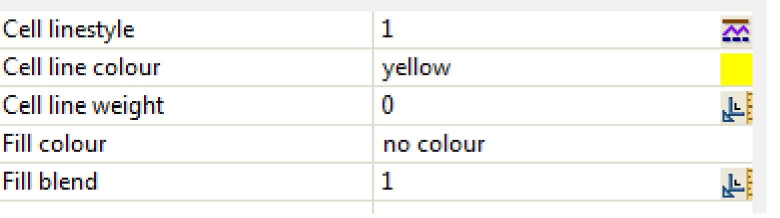
Textstyle	Arial
Name	
Type	world
Size	0.5
Angle	0°00'
X-factor	1.0
Slant	0°00'
Offset	0.0
Raise	0.0
Colour	green
Whiteout	
Border	
Border type	
Justify	middle-centre
TTF Underline	no
TTF Strikeout	no
TTF Italic	no
TTF Outline	no
TTF Weight	400.0

These are the standard settings for defining text. See 4.6 Text Definitions.

Continue to the next section 15.6.5.3 Table Styles - Header Style Node or return to 15.6.5 Table Styles or 15.6.1 Create Table or 15.6 CAD Table.

15.6.5.3 Table Styles - Header Style Node

This is on the RHS of the panel when you click on the **Header style** node.



Cell linestyle	linestyle box	available linestyles

linestyle for drawing the line work around the header row in the table.

Cell line colour	text box	

colour of the line work drawn around the header row.

Cell line weight	real box	

weight of the line work drawn around the header row.

Fill colour	colour box	available colours

colour to fill the header row cells.

Fill blend	real box	

blend for the fill colour for the header row cells.

To control **text** in the Header row, see 15.6.5.3.1 Table Styles - Header Style >Text Style Node

15.6.5.3.1 Table Styles - Header Style >Text Style Node

Important Note - all the text in a table will be treated as World Units.

This is on the RHS of the panel when you click on the **Header style >Text style** node.

If **Use header** is **yes** in the tables **Properties** then this node controls the drawing of the text in the **Header** at the top of each column in the table.

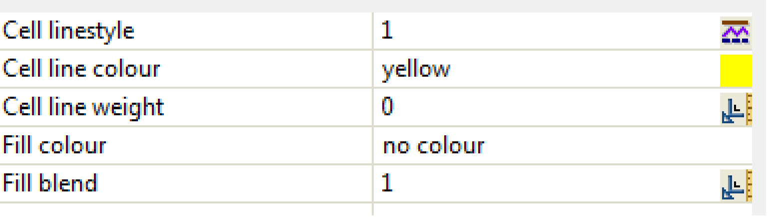
Textstyle	Arial
Name	
Type	world
Size	0.5
Angle	0°00'
X-factor	1.0
Slant	0°00'
Offset	0.0
Raise	0.0
Colour	green
Whiteout	
Border	
Border type	
Justify	middle-centre
TTF Underline	no
TTF Strikeout	no
TTF Italic	no
TTF Outline	no
TTF Weight	400.0

These are the standard setting for defining text. See 4.6 Text Definitions.

Continue to the next section 15.6.5.4 Table Styles - Data Style Node or return to 15.6.5 Table Styles or 15.6.1 Create Table or 15.6 CAD Table.

15.6.5.4 Table Styles - Data Style Node

This is on the RHS of the panel when you click on the **Data style** node.



Cell linestyle linestyle box available linestyles

linestyle for drawing the line work around the data rows in the table.

Cell line colour text box

colour of the line work drawn around the data rows.

Cell line weight real box

weight of the line work drawn around the data rows.

Fill colour colour box available colours

colour to fill the data row cells.

Fill blend real box

blend for the fill colour for the data row cells.

To control the text in the Data rows, see 15.6.5.3.1 Table Styles - Header Style >Text Style Node

15.6.5.4.1 Table Styles - Data Style >Text Style Node

Important Note - all the text in a table will be treated as World Units.

This is on the RHS of the panel when you click on the **Data style >Text style** node.

This node controls the drawing of the text in the **Data** rows in the table.

Textstyle	Arial
Name	
Type	world
Size	0.5
Angle	0°00'
X-factor	1.0
Slant	0°00'
Offset	0.0
Raise	0.0
Colour	green
Whiteout	
Border	
Border type	
Justify	middle-centre
TTF Underline	no
TTF Strikeout	no
TTF Italic	no
TTF Outline	no
TTF Weight	400.0

These are the standard settings for defining text. See 4.6 Text Definitions.

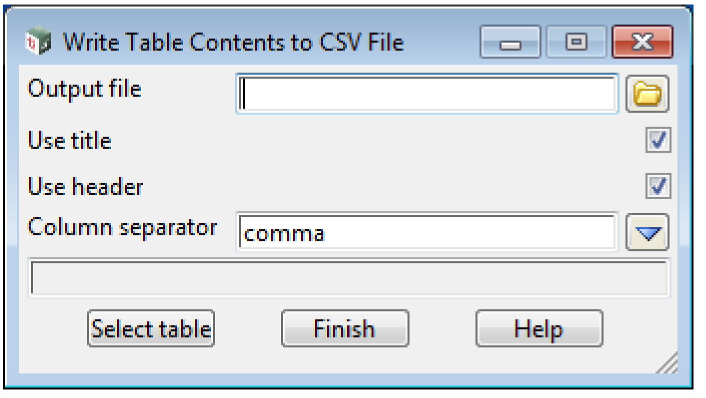
Return to 15.6.5 Table Styles or 15.6.1 Create Table or 15.6 CAD Table.

15.6.6 Table to CSV

Position of option on menu: **CAD =>Table =>Table to csv**

This option creates a CSV file for a table that is associated with a CSV file.

Selecting the **Table to csv** brings up the **Write Table Contents to CSV File** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Output file	file box		*.csv files
name of the CSV file to write to the table out to.			
Use title	tick box		
*if **ticked**, the first line of the CSV file is the title of the table.*			
Use header	tick box		
*if **ticked**, the next line of the CSV file is the Headers for each column of the table.*			
Column separator	choice box	commas	comma, semicolon, tab
the column separator used when writing out the CSV file.			
Select table	button		
select the table AND then write out the contents of the table to a CSV file.			

Return to 15.6 CAD Table.

15.7 Style XML Files for Dimensions, Leaders & Tables

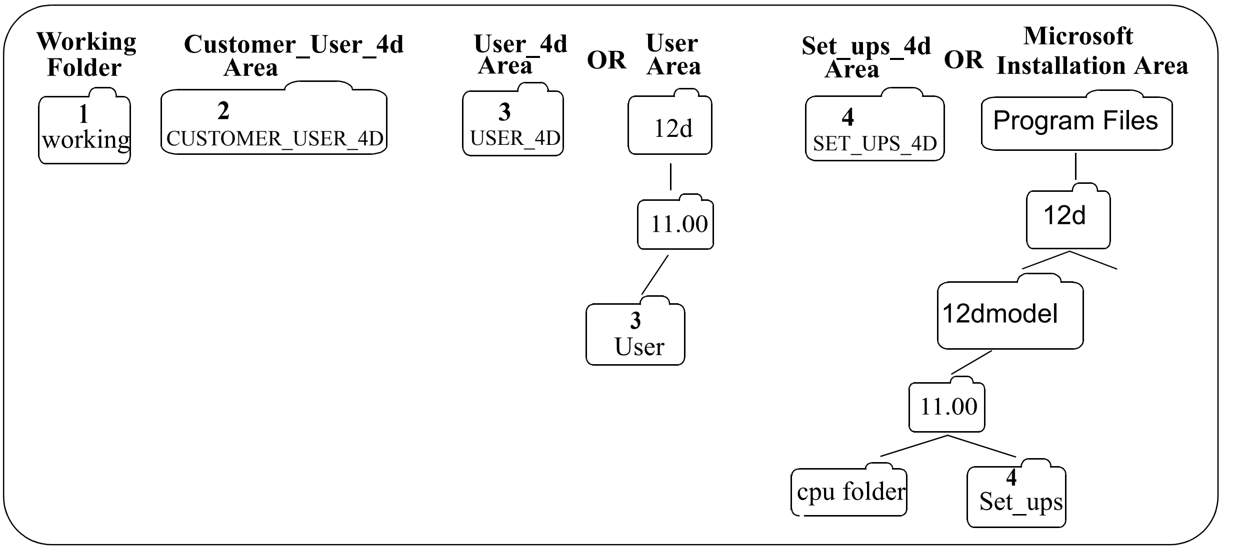
There are XML files to define the styes used for Dimensions, Leaders and Tables:

(a) for **Dimensions**, the file is **dimensions_styles.xml**

(b) for **Leaders**, the file is **leader_styles.xml**.

(c) for **Tables**, the file is **table_styles.xml.**

These XML files are all **Set Up** files like *colours.4d* and folders are searched is a specific order to find the files.



HOWEVER, unlike *colours.4d*, it doesn't matter if the styles xml file is found in one folder, the search **continues** and **all** the **folders** that **contain the file** are found.

And, again unlike *colours.4d*, the styles used are the **merger** of **all** the **found files**.

The search order is still important because if the **same name** exists in more than one file in the found folders, the **first** occurrence in the files in the search order is the one that **is used**.

For example, if you had a dimensions style called **d1** in the file *dimensions_styles.xml* in your working folder, and one called **d1** in the file *dimensions_styles.xml* in Programs files\12d\12dmodle\11.0\set_ups, then the **d1** is your **working folder** is the one that will be used and seen in any dimensions styles pop ups in ***12d Model***.

See

15.7.1 Create & Editing Dimensions, Leader & Table Styles

15.7.1 Create & Editing Dimensions, Leader & Table Styles

Position of option on menu: **CAD =>Dimension =>Styles**
Position of option on menu: **CAD =>Leader =>Styles**
Position of option on menu: **CAD =>Table =>Styles**

When you click on the options, a **Edit Styles** panel is brought up for the appropriate style and it shows the standard areas for looking for the Styles XML files - Working folder, Customer (User), User, Set Ups and Other - and displays whether the appropriate xml file:

(a) exists and available for editing (**Edit**)

If the xml file exists in the folder and you have access to edit it, then the folder name is written in the panel and **[Edit]** is written on the right hand side of the line.

(b) does not exist but can be created (**Create**) by the editor

If the xml file does not exist in the folder and you have access to create it, then the folder name is written in the panel and **[Create]** is written on the right hand side of the line.

In this case a new file is created and written out to the given folder.

(c) is in a folder that has no access for editing and so can only be viewed (**View**)

If there is no access the folder to create/edit the file, then the folder name is written in the panel and **[View]** is written on the right hand side of the line. Although the styles can't be edited, they can be **copied**.

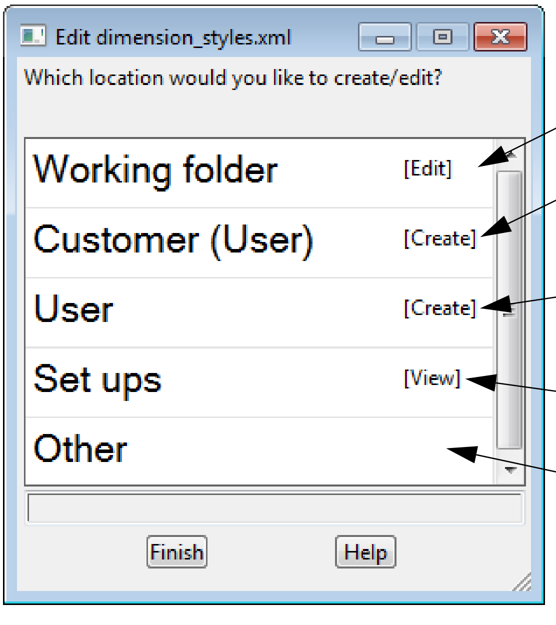
This usually applies to the **Set Ups** folder which normally needs Admin privileges to write to.

(d) is in **Customer (User)**.

This only appears if a *Customer User* area has been defined.

(e) is defined separately by the user and then **Other** is displayed.

In this case, the full path name of the XML file is given by the user.



[Edit] - there is an xml file and it can be edited

[Create] - there is no xml file but one can be created. **Customer (User)** only appears if you have defined a Customer User area

[Create] - there is no xml file but one can be created.

[View] - no xml file can be created because there is no access, just viewed.

nothing means that the user has not defined any additional files to use

See

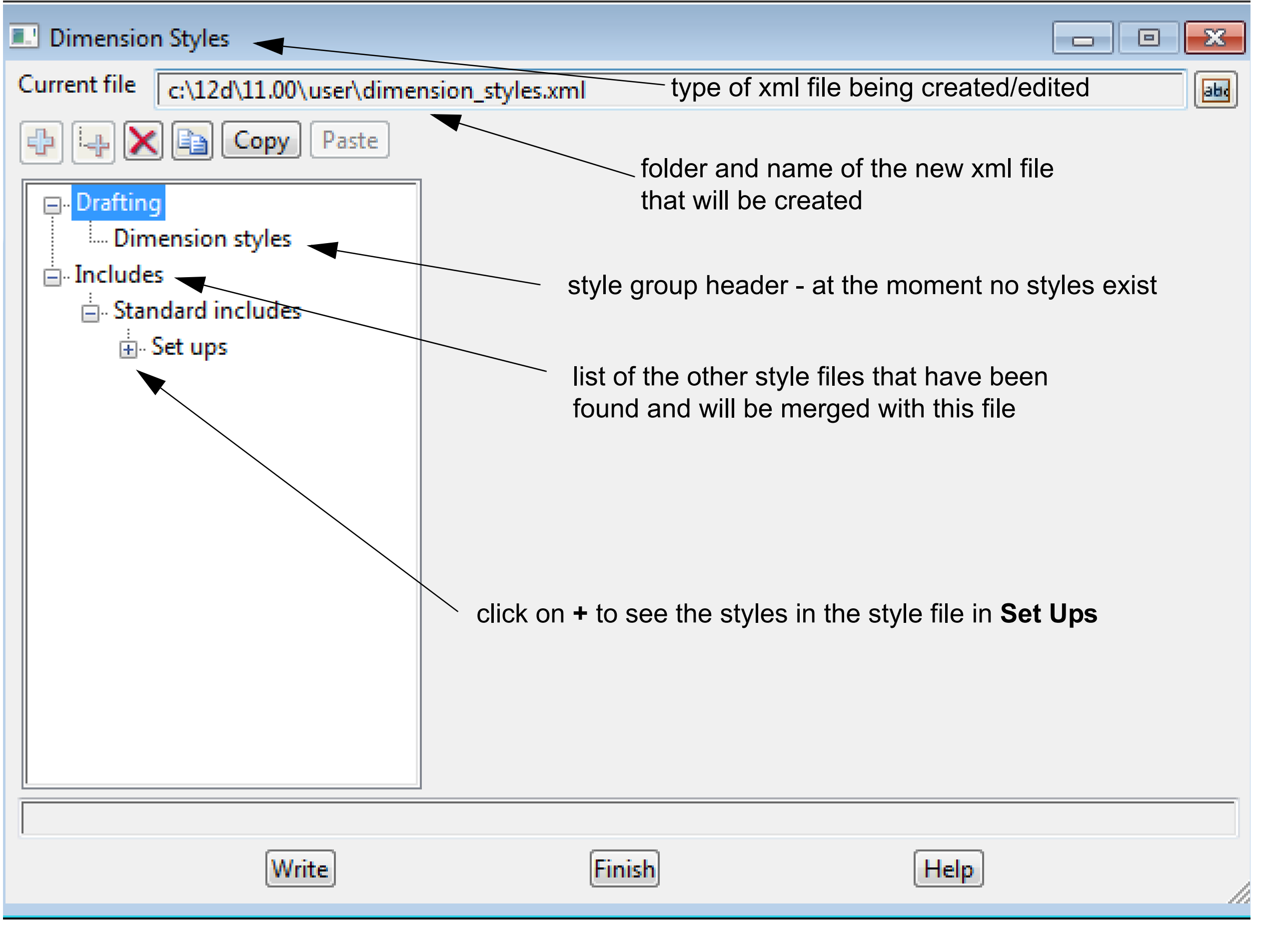
15.7.1.1 [Create] for Dimension, Leader and Table Styles
15.7.1.2 [Edit] for Dimension, Leader and Table Styles

15.7.1.1 [Create] for Dimension, Leader and Table Styles

When **[Create]** is shown, a new xml file can be created in that folder.

Click on the line with **[Create]** and the **Editor** for that Style starts up with a **style group header** but no styles in it.

All the other style files that are found are listed in the **Includes** section displayed at the bottom of the tree.



The included files can not be edited whilst create/editing this file BUT you can copy items from them and paste them into items in this file.

New styles can now be created (see 15.7.1.2.1 Creating New Styles) and a new XML created by clicking on the **Write** button.

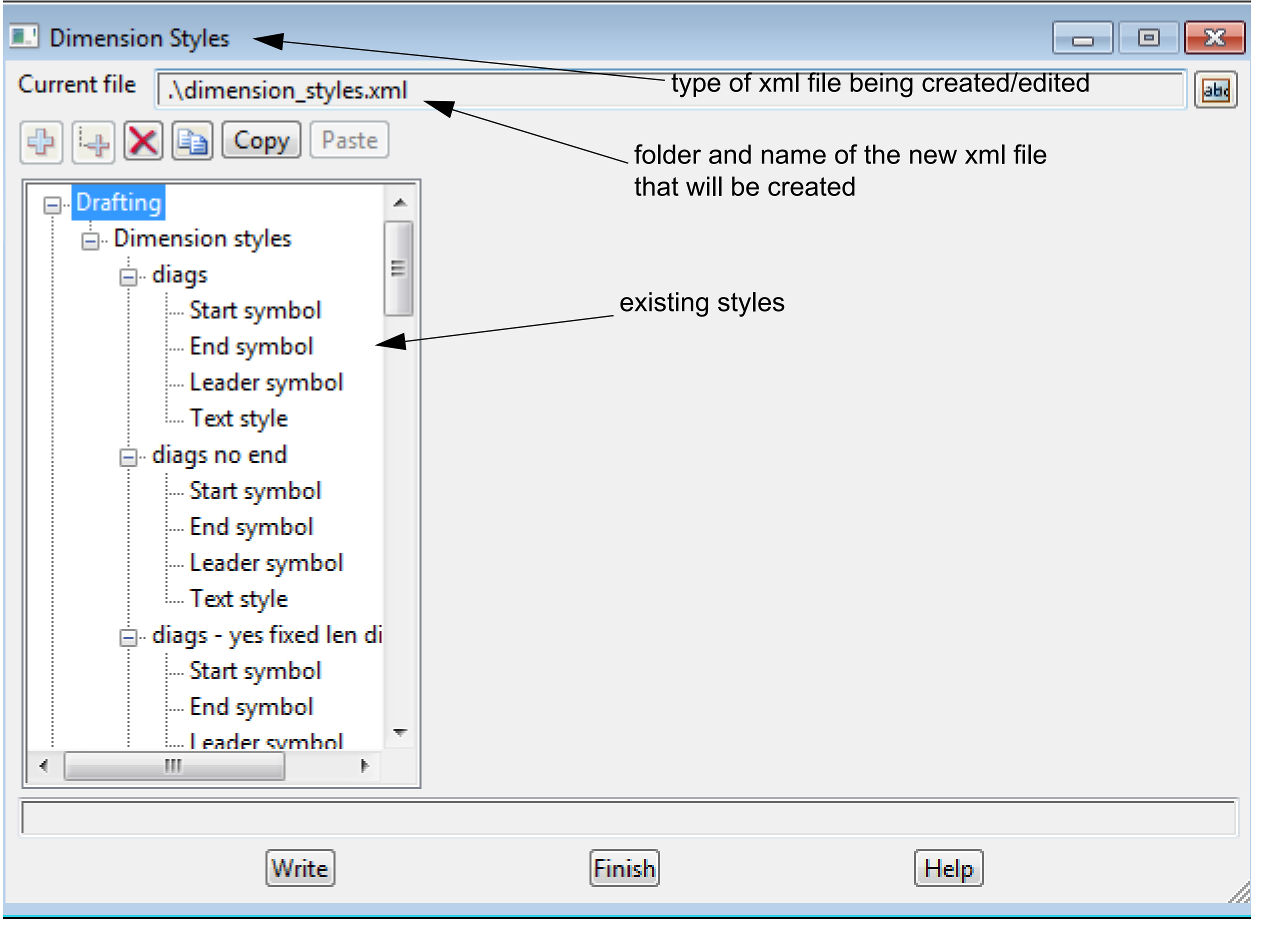
Creating and editing of styles is the same as for **[Edit]** and so is documented in that section. See 15.7.1.2 [Edit] for Dimension, Leader and Table Styles.

Write - writes out the information in the panel to the files name given in **Current file**.

15.7.1.2 [Edit] for Dimension, Leader and Table Styles

When **[Edit]** is shown, an xml file already exists and can be edited.

Click on the line with **[Edit]** and the **Editor** for that *Style* starts up with all the existing styles in that file displayed.



After completing any editing, the XML is saved by clicking on the **Write** button.

As in the **[Create]** case, all the other appropriate files that are found are listed in the Includes section are displayed at the bottom of the tree (see 15.7.1.1 [Create] for Dimension, Leader and Table Styles).

The included files can not be edited whilst create/editing this file BUT you and copy items from them and paste them into items in this file.

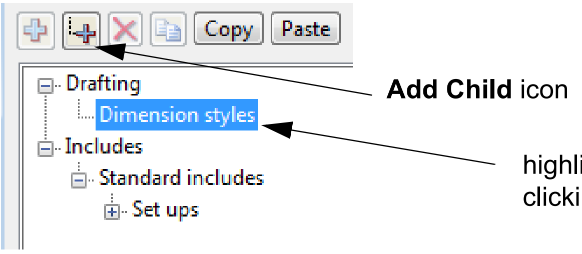
For editing in the panel, see

- 15.7.1.2.1 Creating New Styles
- 15.7.1.2.2 Copy and Paste
- 15.7.1.2.3 Deleting a Style
- 15.7.1.2.4 Duplicating a Style
- 15.7.1.2.5 Editing a Style

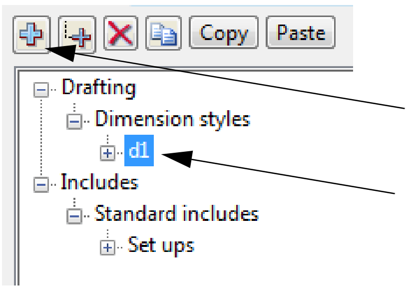
15.7.1.2.1 Creating New Styles

To create a **new style** from scratch when there are **no styles** already, highlight the **style group header** (Dimension style**s**, Leader style**s** or Table style**s**) and click on the **Add Child** icon.

Note - the **Add Child** icon adds a node as a subnode of the highlighted node.

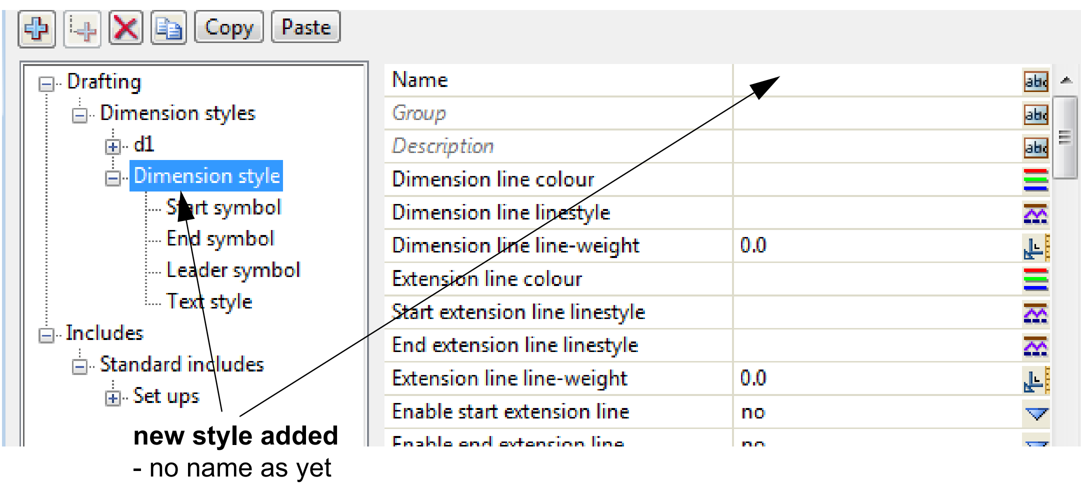


Too create a **new style** from scratch when **some styles already exist**, you can highlight the **style name** (name of the style or Dimension style, Leader style or Table style if it has not been given a name) and click on the **Add** (Add Sibling) icon.

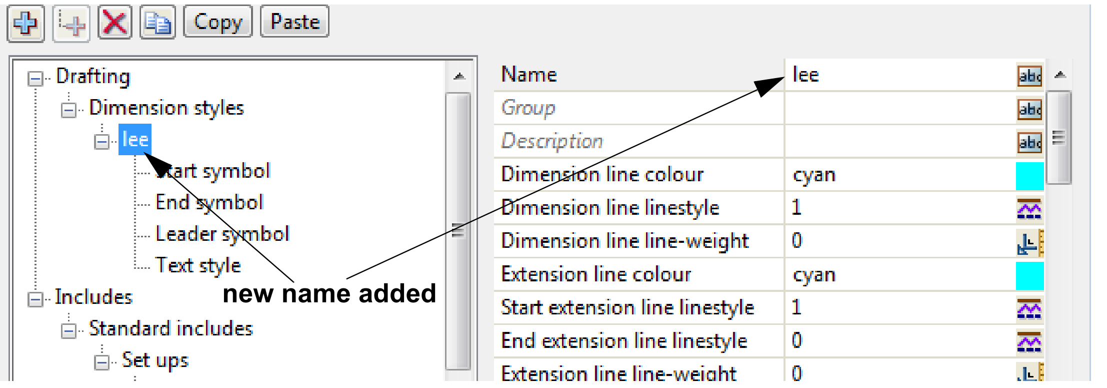


The **Add** icon adds a new item at **the same level** (a sibling) as the highlighted item and that is why a style must already exist so that one can be selected.

In both cases a new style is created with no name and so the general style name (Dimension style, Leader style or Table style) is written in the style list until the style is given a name.



Once the style is given a name then that name will appear in the tree.



Once the new style is created then it can be edited by going into each field of the style and making changes.

Quicker Methods for Creating New Styles

When you click on a node to edit that section of the tree, you can not leave the edited node until all the required values are filled in. Similarly you won't be able to save the style unless all the required values are filled in. And this may involve filling in a lot of values.

So rather than starting from scratch, it is often easier to use **Duplicate** (15.7.1.2.4 Duplicating a Style) if there are already existing styles in the file, or **Copy** and **Paste** (15.7.1.2.2 Copy and Paste).

Copy and **Paste** is particularly useful when there are **no** styles in the file you are creating styles but there are styles in the other files listed in the **Includes** node. Although these styles can't be edited, they can be copied.

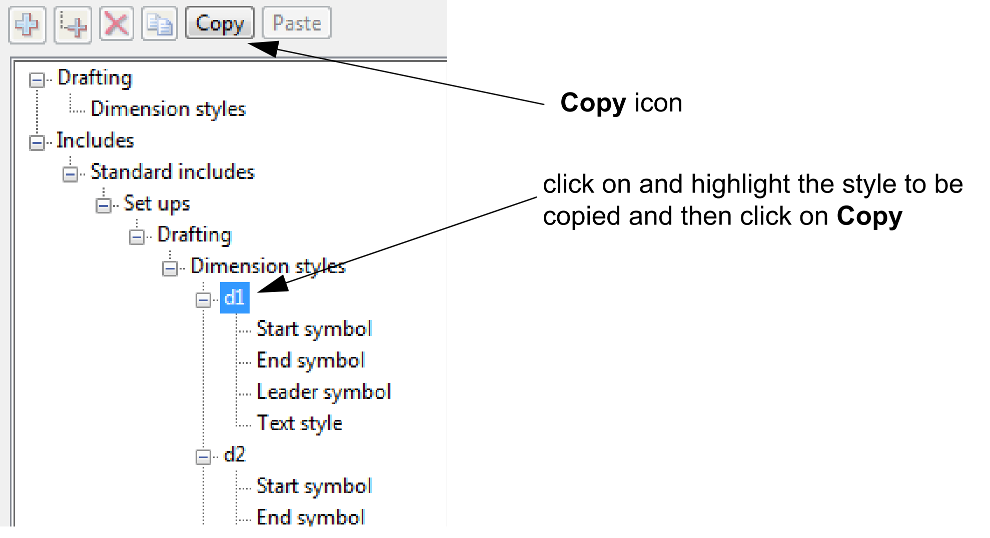
Continue to the next section 15.7.1.2.2 Copy and Paste or return to 15.7.1.2 [Edit] for Dimension, Leader and Table Styles or 15.7 Style XML Files for Dimensions, Leaders & Tables.

15.7.1.2.2 Copy and Paste

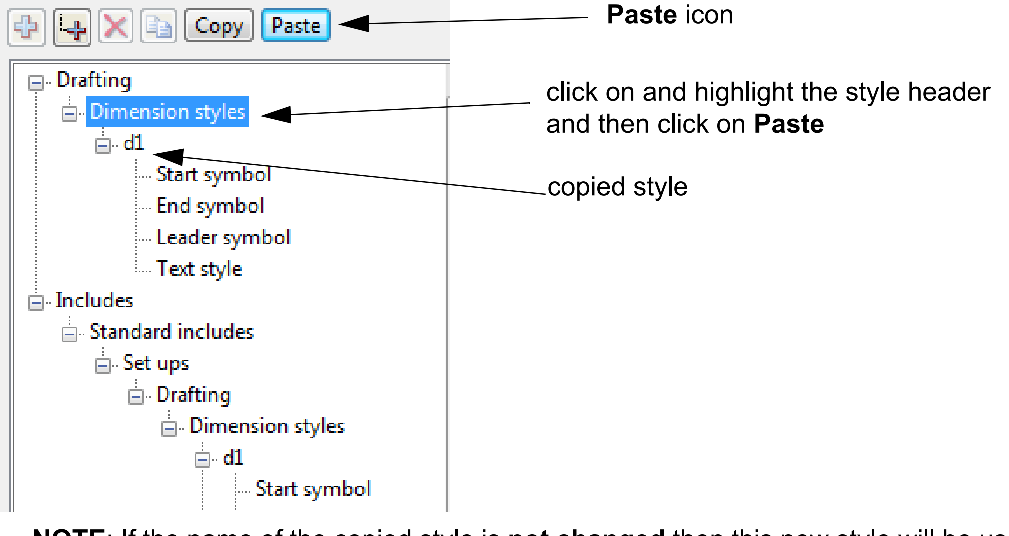
Rather than creating a new style from scratch, it is often easier to use **Copy** and **Paste** to create a copy of an existing style, and then edit the copied style.

Although there may be no styles in the folder you are creating styles for, there will usually be some styles in the XML file in **Set Ups** that can be copied.

In the tree in the **Editor**, click on the **+** in front of **Set Ups** to expand the tree and then click on the item in the tree that you want to copy. Then click on the **Copy** icon.



Next click on and highlight the style header (Dimension styles, Leader styles or Table styles) and click on **Paste**. The style will then appear under the style header with the same name as the original style. The copied style can then be edited.

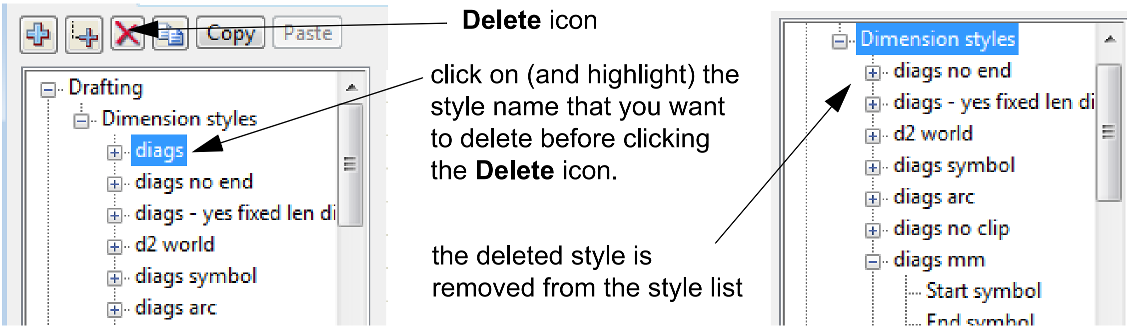


NOTE: If the name of the copied style is **not changed** then this new style will be used instead of the one in **Set Ups**.

Continue to the next section 15.7.1.2.3 Deleting a Style or return to 15.7.1.2 [Edit] for Dimension, Leader and Table Styles or 15.7 Style XML Files for Dimensions, Leaders & Tables.

15.7.1.2.3 Deleting a Style

To **delete** a style, simply click on and highlight the **style name,** and then click on the **Delete** icon.

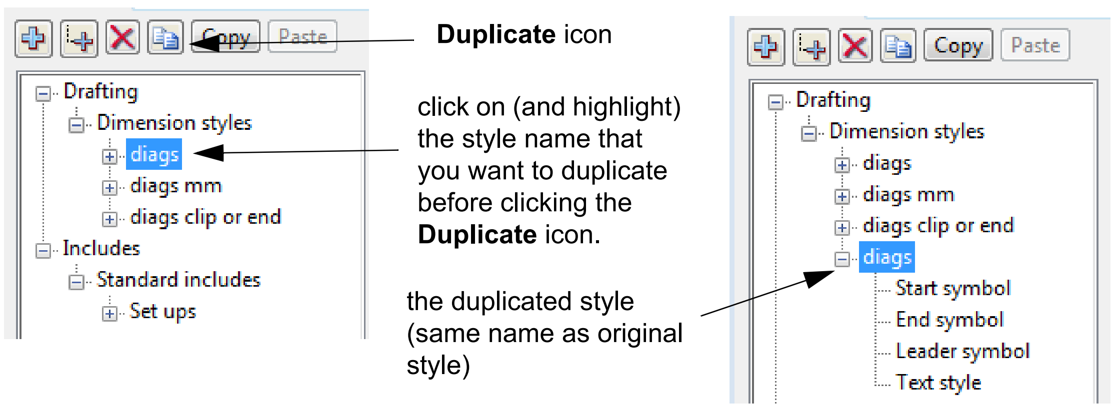


Continue to the next section 15.7.1.2.4 Duplicating a Style or return to 15.7.1.2 [Edit] for Dimension, Leader and Table Styles or 15.7 Style XML Files for Dimensions, Leaders & Tables.

15.7.1.2.4 Duplicating a Style

To **duplicate** a style, simply click on and highlight the **style name,** and then click on the **Duplicate** icon.

A copy of the style, with exactly the same name, is created and added to the end of the list of styles.

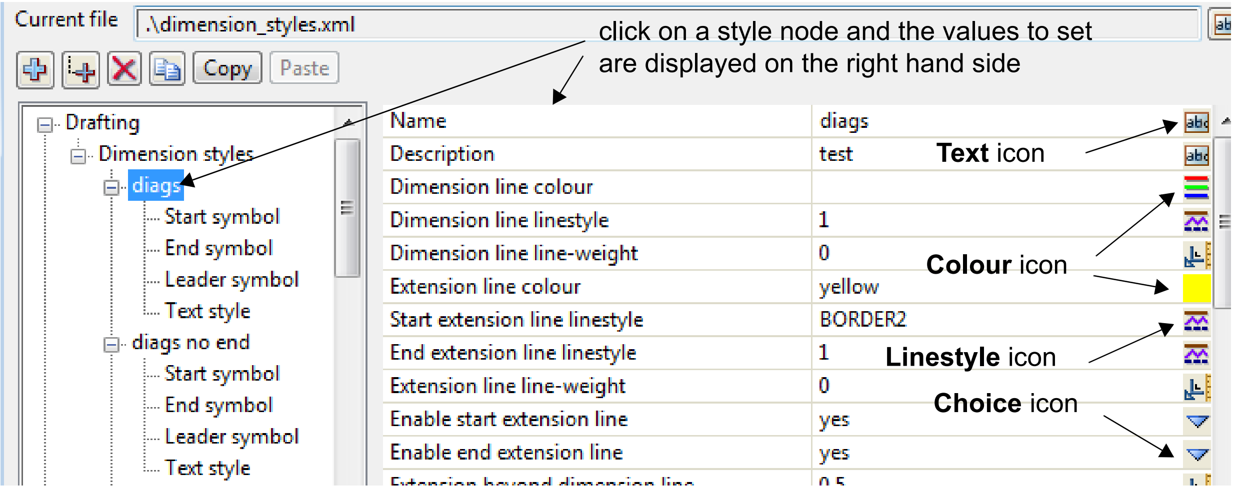


The name of the style needs to be changed and any other edits done that are required.

Continue to the next section 15.7.1.2.5 Editing a Style or return to 15.7.1.2 [Edit] for Dimension, Leader and Table Styles or 15.7 Style XML Files for Dimensions, Leaders & Tables.

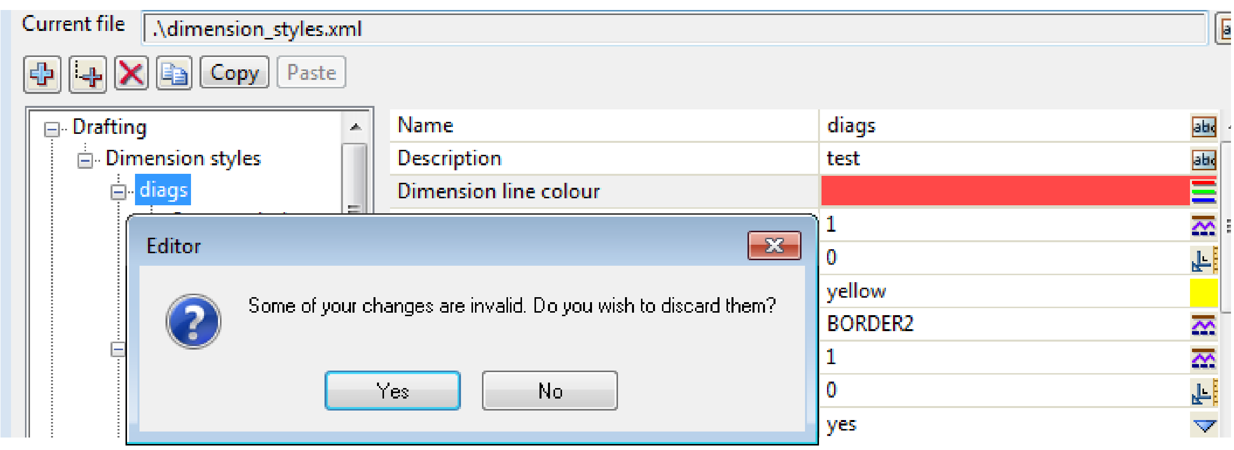
15.7.1.2.5 Editing a Style

To edit a **style**, you click on the style name, or expand the style and click on any of the sub nodes, and all the values for that node will be displayed on the right hand side of the panel.



The panel fields on the right hand side of the panel have the same icons as in other ***12d Model*** panels and the standard pop ups are all available. For example, text, colour and choice icons.

When clicking to go to another node, all the fields currently on the right hand side are validated and if there are any problems, the first line with an error is coloured red and an **Editor error** box is displayed.



If **Yes** is selected, then all the **invalid field** values are **discarded**, the **Editor error** panel removed, and the data for the new selected node displayed on the right hand side of the panel.

If **No** is selected, then the right hand side of the panel does not change and can undergo further edits.

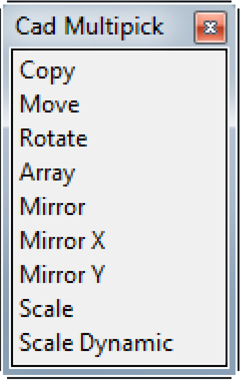
Return to 15.7 Style XML Files for Dimensions, Leaders & Tables.

15.8 CAD Multipick

Position of option on menu: **CAD =>Multipick**

This section of documentation is a work in progress and will be updated in subsequent releases.

The **Cad Multipick** walk-right menu is



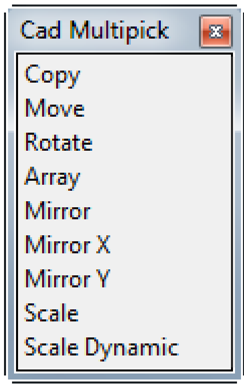
For selecting the data, see 15.8.1 Selecting with Multipick

For the options, see

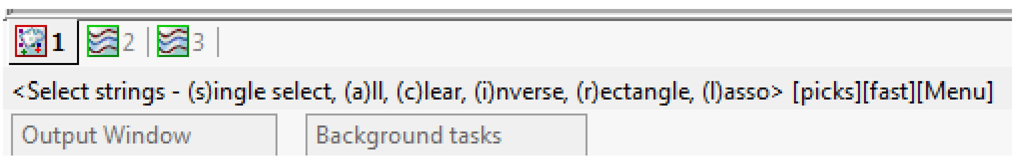
Copy	15.8.2 CAD Multipick Copy
Move	15.8.3 CAD Multipick Move
Rotate	15.8.4 CAD Multipick Rotate
Array	15.8.5 CAD Multipick Array
Mirror	15.8.6 CAD Multipick Mirror
Mirror X	15.8.7 CAD Multipick Mirror X
Mirror Y	15.8.8 CAD Multipick Mirror Y
Scale	15.8.10 CAD Multipick Scale
Scale Dynamic	15.8.11 CAD Multipick Scale Dynamic

15.8.1 Selecting with Multipick

Upon selecting any of the Multipick options from the **CAD Multipick** menu



the *Screen Message Area* prompts are shown for selecting selection options.

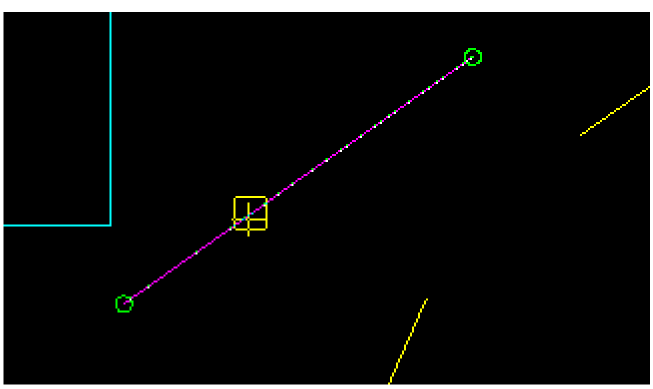


The default option is <**m**> for multipick select.

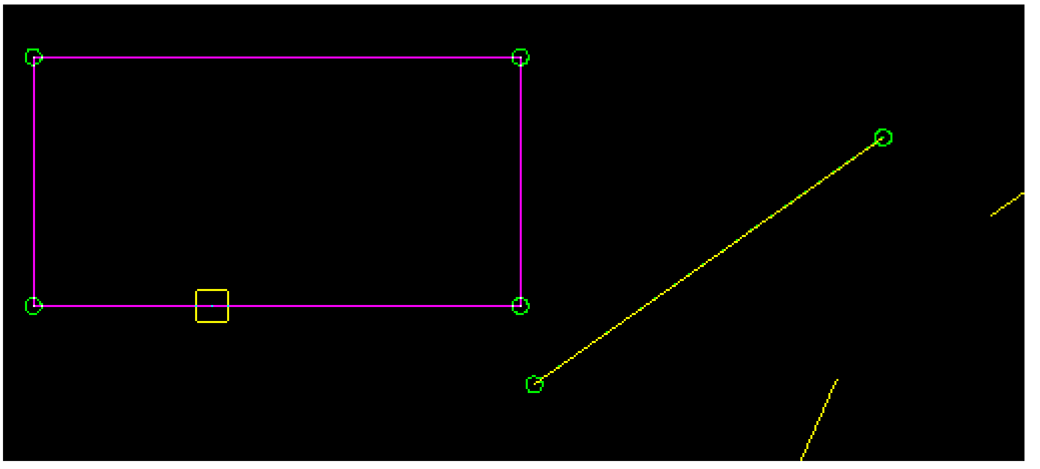
For <***m***> go to	Multipick
<***s***>	Single
<***a***>	All
<***c***>	Clear
<***i***>	Inverse
<***r***>	Rectangle
<***l***>	Lasso

Multipick

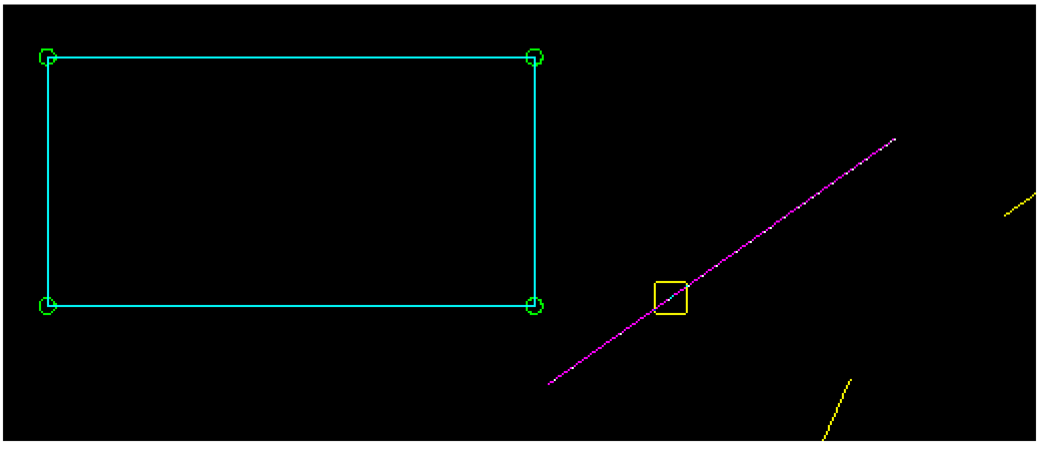
For <**m**>, strings are selected by clicking LB on the string, but unlike a normal select, ***the string is not accepted***. After LB is clicked, the string is highlighted and circles placed at the string vertices to indicate that the string is part of the ***Multipick set***.



Another string is added to the *Multipick set* by clicking LB on the new string and again ***not accepting***. This string will also be drawn with circles at each of its vertices to indicate that it is part of the *Multipick set*.



A string is ***removed*** from a *Multipick set* by simply clicking LB on the string (and not accepting). The string is then deselected and the circles are removed from the vertices of the string to indicate that it is no longer part of the *Multipick set*.



When all the required strings of the *Multipick set* have been selected, the Multipick selection is terminated by clicking MB (accepting).

The selected strings in the *Multipick set* are then passed on to the CAD Multipick option for processing. For example, if the Multipick was part of **Multipick Copy**, then after Accepting, the ***Copy*** part of the **Multipick Copy** takes over.

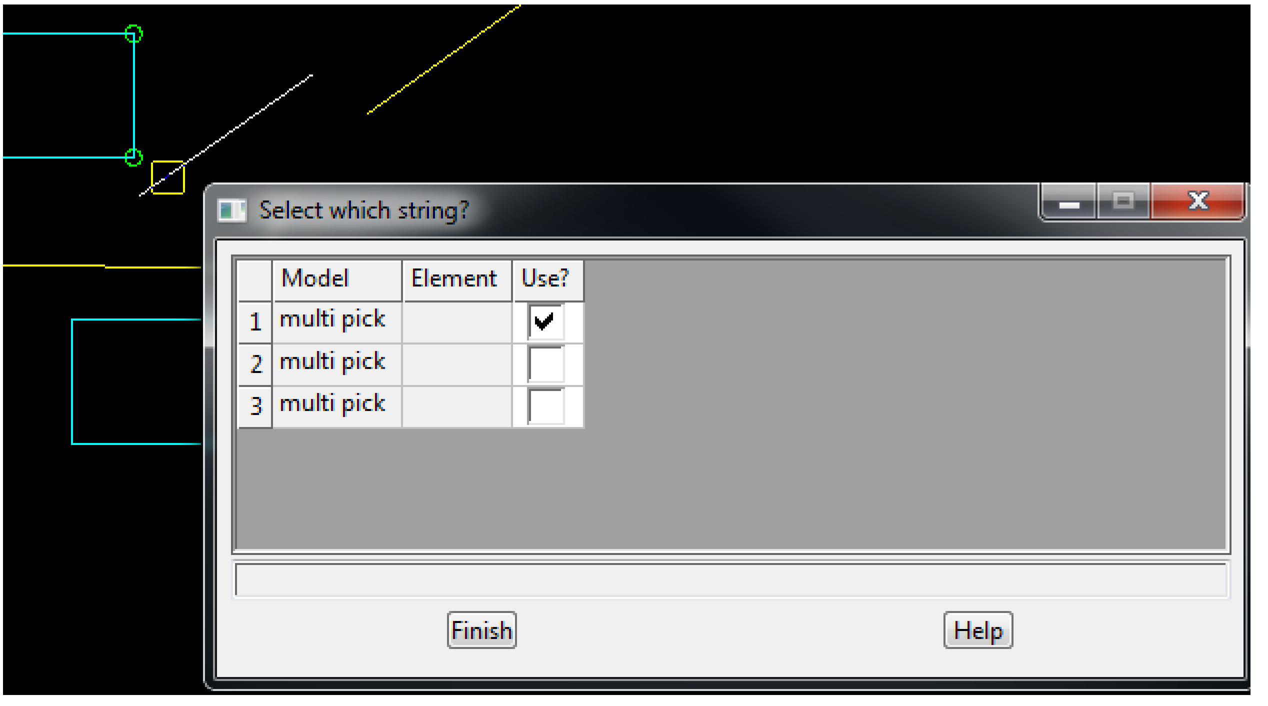
Once the Multipick option (*e.g.* **Multipick Copy**) is completed, the ***Multipick set*** is cleared and the Multipick CAD option begins again, and the process of building up a new ***Multipick set*** begins again for that same option.

To terminate the *Multipick option*, simply press <Esc>.

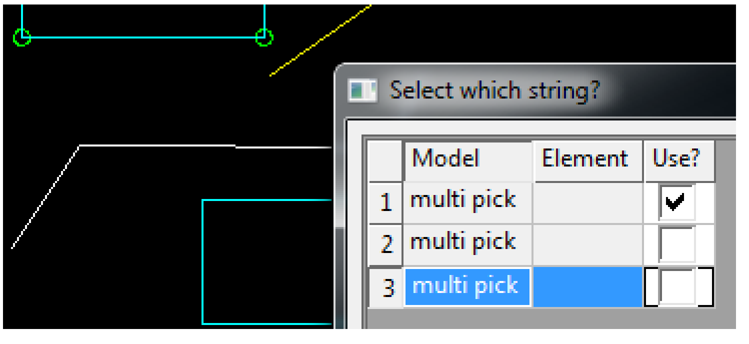
Note:

Because LB is used to keep adding new strings to the ***Multipick set***, it can't be used for the normal cycling through a selection set when more than one string satisfies the picking tolerance. So when more than one string satisfies the picking criteria a **Select which string?** panel is placed on the screen with the list of strings satisfying the LB picking criteria.

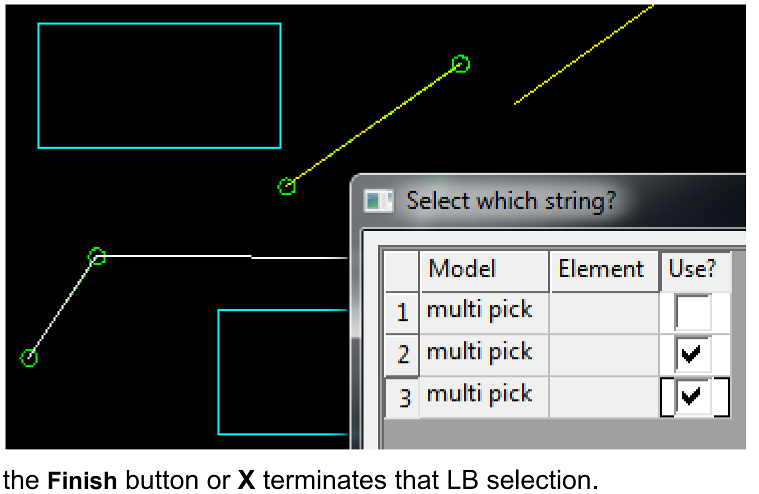
If the string is already in the *Multipick set*, there will be a *tick* in the **Use?** column.



Clicking on a line in the grid highlights the string.



And clicking in the **Use?** column to make a ***tick*** will add the string to the *Multipick set* and place circles on each of its vertices, or ***turning off a tick*** will deselect the string from the *Multipick set* and remove the circles from each of its vertices.



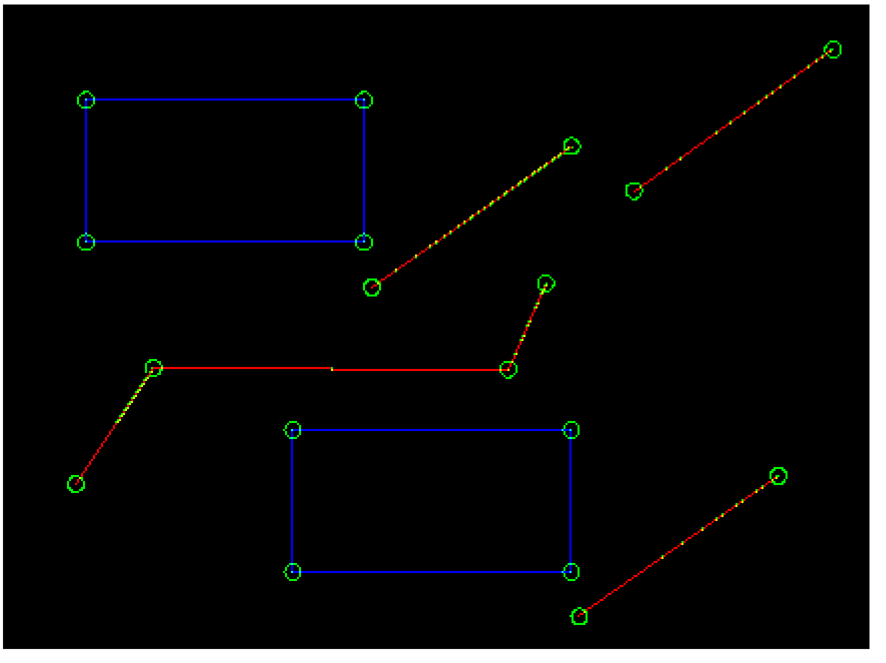
Clicking on the **Finish** button or **X** terminates that LB selection.

Single

<**s**> places the option back into the mode of selecting a single string. That is, Multipick is not used.

All

<**a**> will add all the strings in the models on the current view to the Multipick set. Note that the strings do not have to be visible on the view to be selected by <a>.

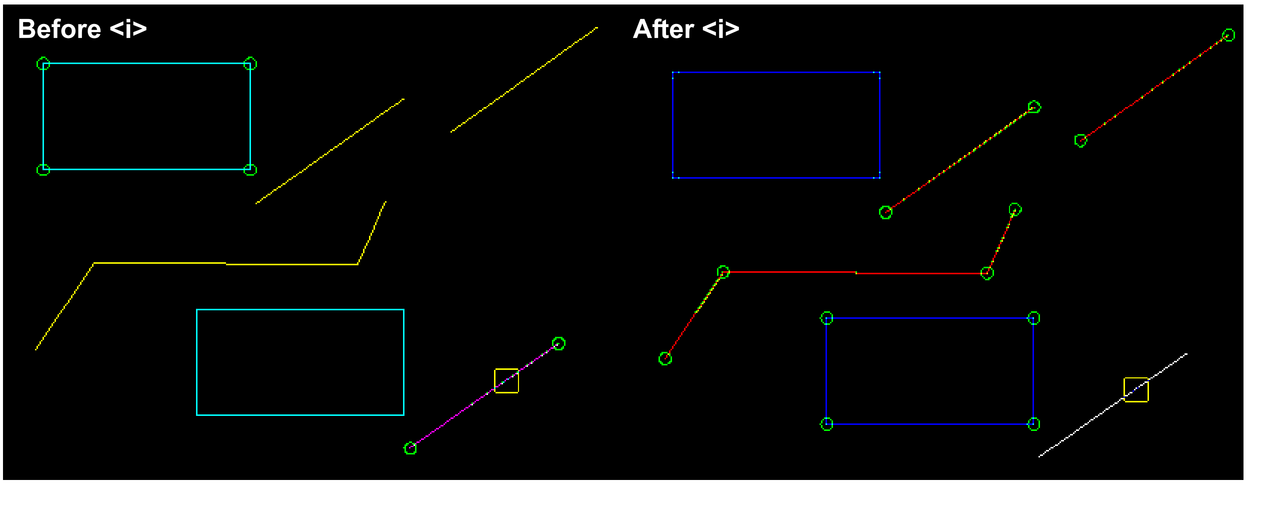


Clear

<**c**> will clear the current Multipick set. That is, all selected strings are deselected.

Inverse

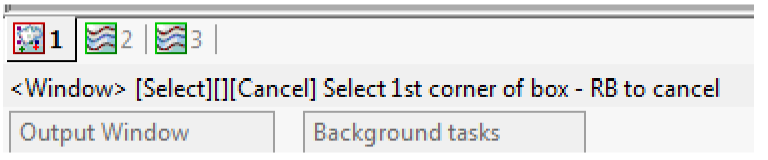
<**i**> will take the inverse of the current *Multipick set*. That is, all strings in the models on the current view that are currently in the *Multipick set* are deselected, and all strings in the models on the current view that are not n the *Multipick set* are added to the *Multipick set*. Note that the strings do not have to be visible on the view to be selected/deselected by <i>.

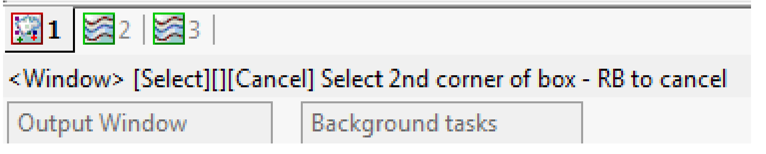


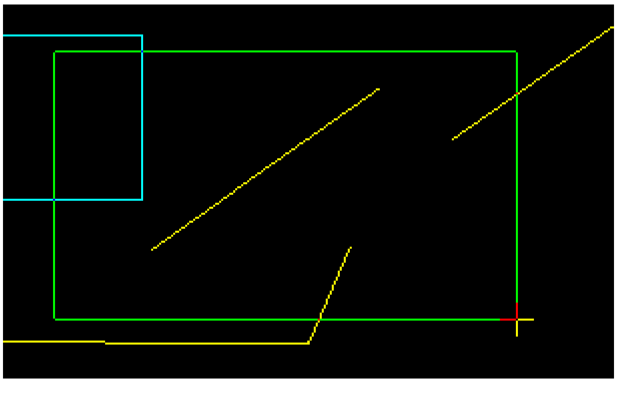
Rectangle

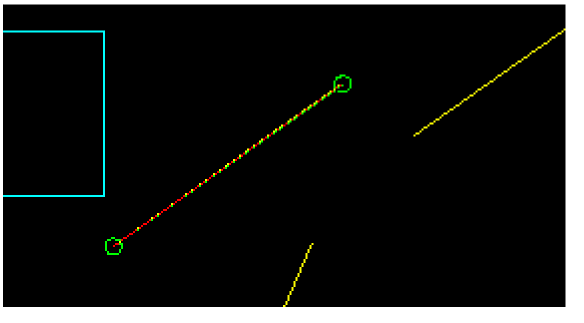
<**r**> requires the user to create a rectangle and all strings totally enclosed by that rectangle will be added to the *Multipick set* if they are not already in the *Multipick set*, or **deselected** if they are already in the *Multipick set*.

After pressing <r>, the first corner of the rectangle is selected by pressing and holding down LB, and the cursor is then dragged to the second corner of the rectangle and LB released.







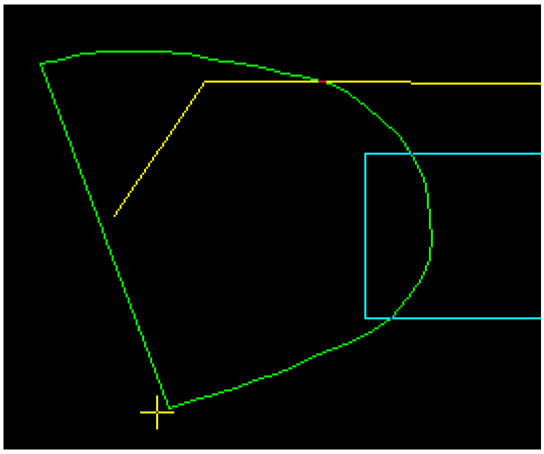


The strings totally enclosed by the rectangle are then added to/removed from the *Multipick set*.

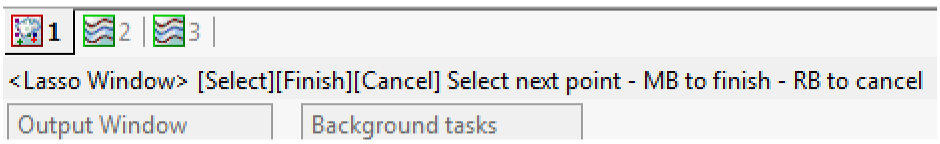
Lasso

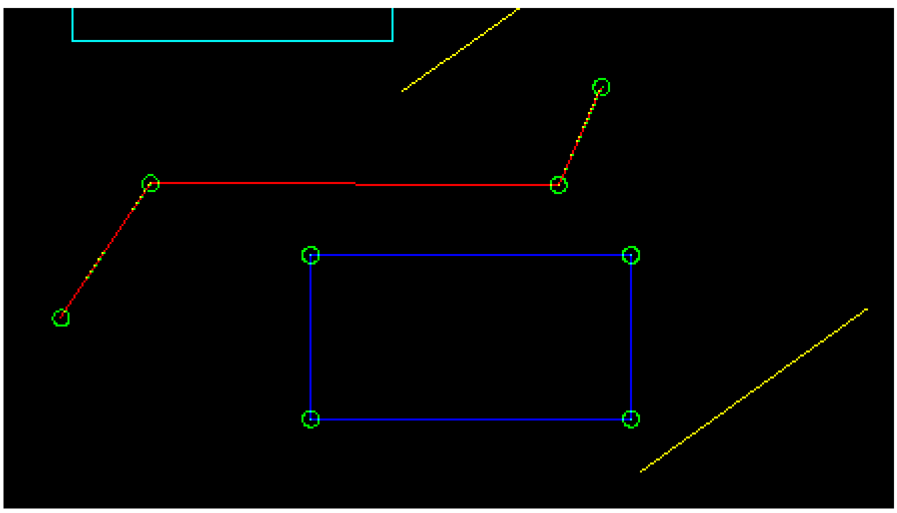
<**l**> requires the user to create a lasso, and all strings enclosed by or crossing the lasso will be added to the Multipick set if they are not in the Multipick set, or deselected if they are already in the Multipick set.

After pressing <l>, the lasso start point is selected by pressing and holding down LB and the cursor is then dragged to trace out the lasso. LB is then released to stop adding to the lasso.



MB is pressed and released to accept the lasso. If LB is pressed down before accepting the lasso, further points are added to the lasso until LB is released.





The strings totally enclosed by, or crossing, the lasso are then added/removed from the *Multipick set*.

15.8.2 CAD Multipick Copy

For selecting the data, see 15.8.1 Selecting with Multipick

After data has been selected, a base point is selected and the Copy part of the option is the same as the 15.20.2 CAD Copy command.

15.8.3 CAD Multipick Move

For selecting the data, see 15.8.1 Selecting with Multipick

After data has been selected, a base point is selected and the Move part of the option is the same as the 15.20.1 CAD Move command.

15.8.4 CAD Multipick Rotate

For selecting the data, see 15.8.1 Selecting with Multipick

After data has been selected, a base point is selected and the Rotate part of the option is the same as the 15.20.4 CAD Rotate command.

15.8.5 CAD Multipick Array

For selecting the data, see 15.8.1 Selecting with Multipick

After data has been selected, a base point is selected and the Array part of the option is the same as the 15.20.3 CAD Array command.

15.8.6 CAD Multipick Mirror

For selecting the data, see 15.8.1 Selecting with Multipick

After data has been selected, a base point is selected and the Mirror part of the option is the same as the 15.20.5 CAD Mirror command.

15.8.7 CAD Multipick Mirror X

For selecting the data, see 15.8.1 Selecting with Multipick

After data has been selected, a base point is selected and the Mirror X part of the option is the same as the 15.20.6 CAD Mirror X Axis command.

15.8.8 CAD Multipick Mirror Y

For selecting the data, see 15.8.1 Selecting with Multipick

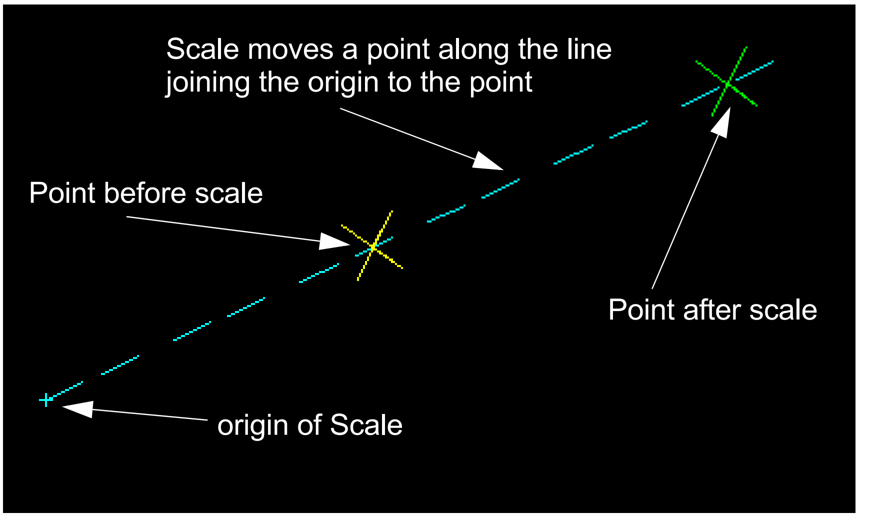
After data has been selected, a base point is selected and the Mirror Y part of the option is the same as the 15.20.7 CAD Mirror Y Axis command.

15.8.9 Scale About an Origin

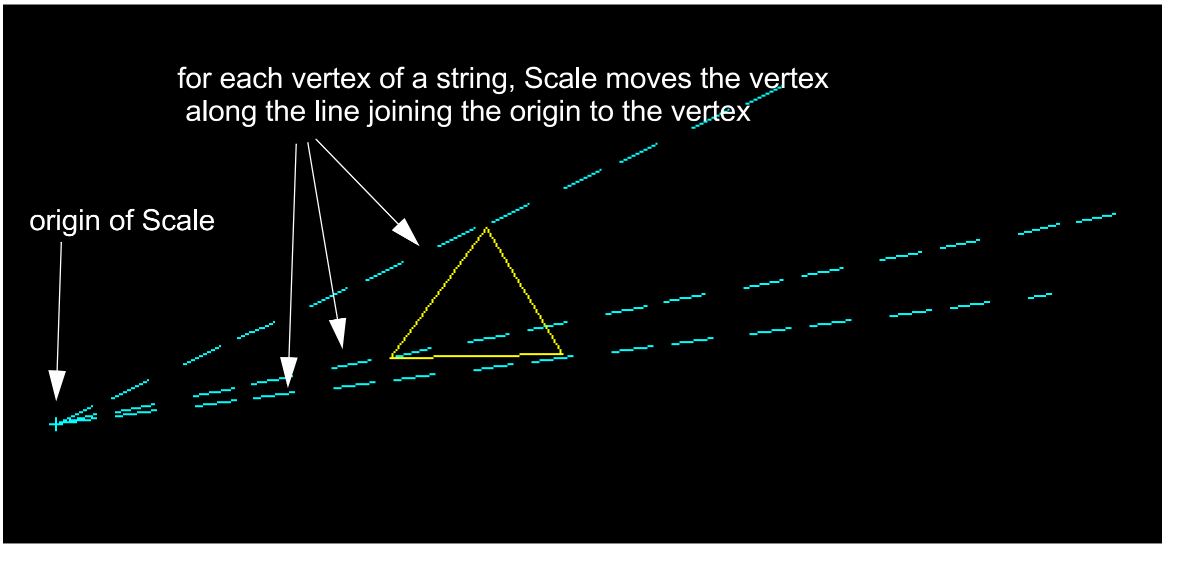
A number of options use the concept of **Scaling** about an **Origin**.

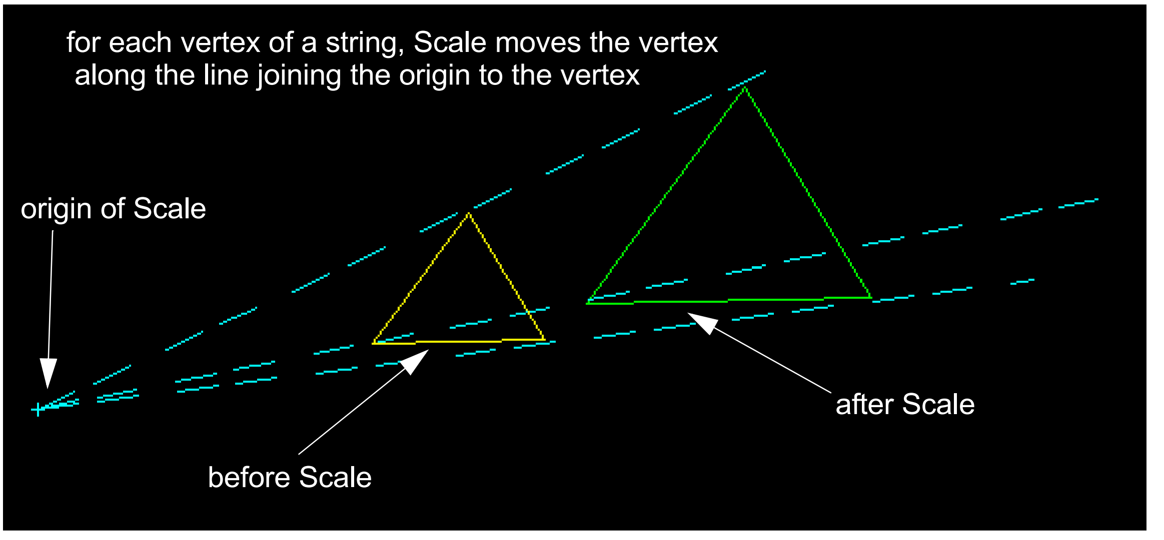
To scale a point about an Origin by a given scale factor, the point is moved along the line joining the point to the origin so that

distance between the scaled point and the origin = scale factor * distance from point to origin



A **string** is scaled about an origin by a given scale factor by scaling each vertex in the string about the origin by the scale factor.



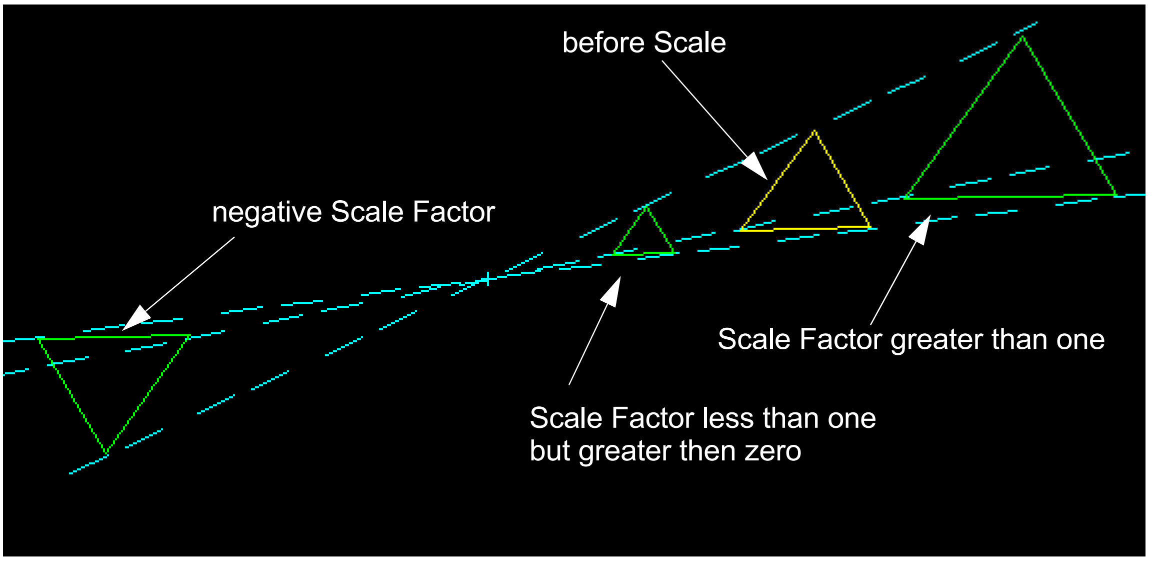


If the Scale Factor is greater than one, then the scaled string is enlarged and is on the same side of the origin as the original string but further from the origin than the original string.

If the Scale Factor is less than one but greater than zero, then the scaled string is reduced and is between the origin and the original string.

If the Scale Factor is negative then the scaled string is on the opposite side of the origin to the original string.

A Scale Factor of zero would put all the vertices of the string on top of the origin.



15.8.10 CAD Multipick Scale

For selecting the data, see 15.8.1 Selecting with Multipick

For the definition of scaling about an origin, see 15.8.9 Scale About an Origin

After data has been selected, an Origin point is selected and the Scale part of the option is the same as the 15.20.8 CAD Scale command.

15.8.11 CAD Multipick Scale Dynamic

For selecting the data, see 15.8.1 Selecting with Multipick

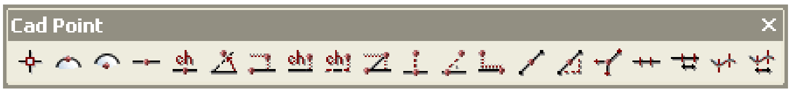
For the definition of scaling about an origin, see 15.8.9 Scale About an Origin

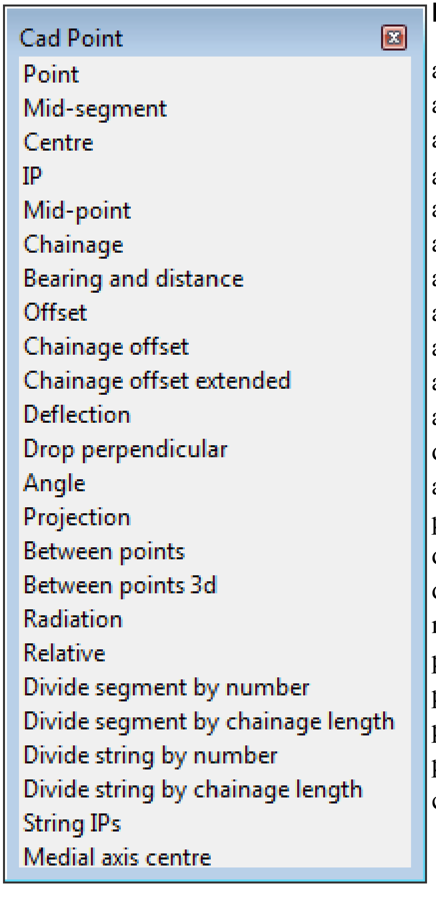
After data has been selected, an Origin point and Start Scale position is selected and the Scale Dynamic part of the option is then similar to the 15.20.9 CAD Scale Dynamic command.

15.9 CAD Point

Position of option on menu: **CAD =>Point**

The **Point** walk-right menu is





Menu of Options to Create New One Point Strings

at a user selected position
at middle of a selected segment
at the centre point of an arc
at the IP of a selected arc or transition segment
at the middle of the two selected points
at the position of a chainage along a selected string
a given bearing and distance from a selected position
at the offset position of a point on a selected string
at the offset position of a chainage along a selected string
at the offset position of a chainage along a selected string extended
a given distance at right angles to a selected string
dropped perpendicularly onto a selected string
a given distance at a given angle to a selected string
projected a chainage distance along a selected string
distance between two selected positions
distance between two selected positions with interpolated z
many points at given bearings and distances from a selected position
points which divide selected seg into a number of equal segs
points which divide selected seg into segs of given length
points which divide selected string into a number of equal pieces
points which divide selected string into pieces of given length
creates new points for each vertex of a string

For the option *Point*, go to	15.9.1 Point
Mid-segment	15.9.2 Mid Segment
Centre	15.9.3 Centre
IP	15.9.4 IP
Mid-point	15.9.5 Mid point
Chainage	15.9.6 Chainage
Bearing and distance	15.9.7 Bearing and distance
Offset	15.9.8 Offset
Chainage offset	15.9.9 Chainage offset
Chainage offset extended	15.9.10 Chainage offset extended
Deflection	15.9.11 Deflection
Drop perpendicular	15.9.12 Drop Perpendicular
Angle	15.9.13 Angle
Projection	15.9.14 Projection
Between points	15.9.15 Between Points

15.9.1 Point

Position of option on menu: **CAD =>Point =>Point**

or by selection of appropriate icon from the toolbar.

This option creates a single vertex (point) string.

On selecting **Point**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

The user can select a position with the mouse and on accepting that point (Middle mouse button or enter) the point is created at the selected position. The model, colour, height, *etc.* are defined in the **Cad Control Bar**.

The snap mode will influence the mouse selection. For example if cursor snap is on, the user can choose a position not yet defined. If point snap is on and the selection snaps to an existing point, the option will place another point at that location.

The user can also activate the selection menu used with the mouse (right button) that allows various positioning options.

Specification of a position can also be done by the direct input of the xyz coordinate of the point by pressing the space bar to bring up the enter XYZ panel or by typing of the value to bring up the XYZ panel. **NOTE:** The z value will default to the value entered into the **Cad Control Bar** whether or not it is specified in the XYZ box. If no height value exists in the **Cad Control Bar** then a value will be interpolated if possible, otherwise a null value will be assigned.

15.9.2 Mid Segment

Position of option on menu: **CAD =>Point =>Mid segment**

or by selection of appropriate icon from the toolbar. 

This option creates a point in the middle of a selected line or arc segment.

On selecting **Mid segment**, the user is prompted to select a segment and a vertex is created at the mid point of the segment. The model, colour, height, *etc.* are defined in the **Cad Control Bar**.

15.9.3 Centre

Position of option on menu: **CAD =>Point =>Centre**

or by selection of appropriate icon from the toolbar. 

This option creates a point at the centre of a selected arc or circle.

On selecting **Centre** the user is prompted to select an arc and a vertex is created at the centre of the arc. The model, colour, height, *etc.* are defined in the **Cad Control Bar**.

15.9.4 IP

Position of option on menu: **CAD =>Point =>IP**

or by selection of appropriate icon from the toolbar. 

This option creates a point at the IP of a selected arc or transition segment. That is, it creates a point at the intersection of the tangent from the start of the segment and the tangent at the end of the segment. Picking a straight segment will give an error.

On selecting **IP** the user is prompted to select an arc or transition segment and a new point is created at the IP of the selected segment. The model, colour, height, *etc.* are defined in the **Cad Control Bar**.

15.9.5 Mid point

Position of option on menu: **CAD =>Point =>Mid point**

or by selection of appropriate icon from the toolbar.

This option creates a point in the middle of two selected positions.

On selecting **Mid point**, the user is prompted to select the first position and then the second position. A vertex is created at the mid point of the two selected positions. The model, colour, height, *etc.* are defined in the **Cad Control Bar**.

15.9.6 Chainage

Position of option on menu: **CAD =>Point =>Chainage**

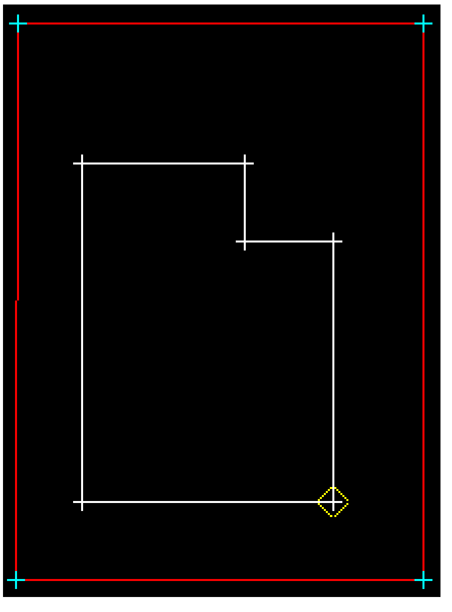
or by selection of appropriate icon from the toolbar. 

This option creates a new point at the position of a user specified chainage of an existing string.

On selecting **Chainage**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

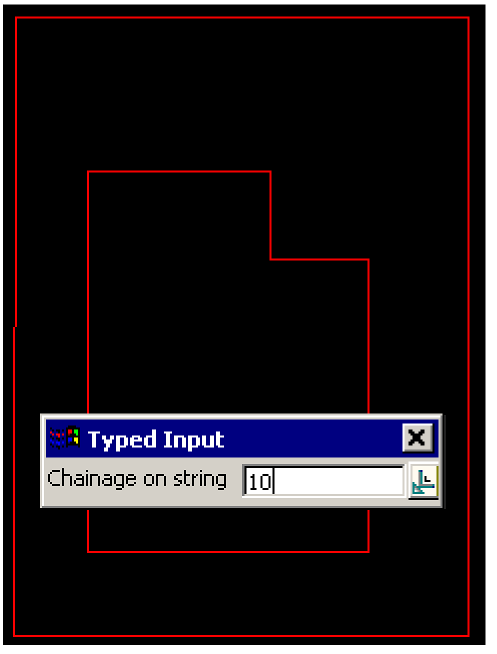
STEP 1:

The user selects a valid string (left mouse button) and accepts that string (Middle mouse button or <Enter>).



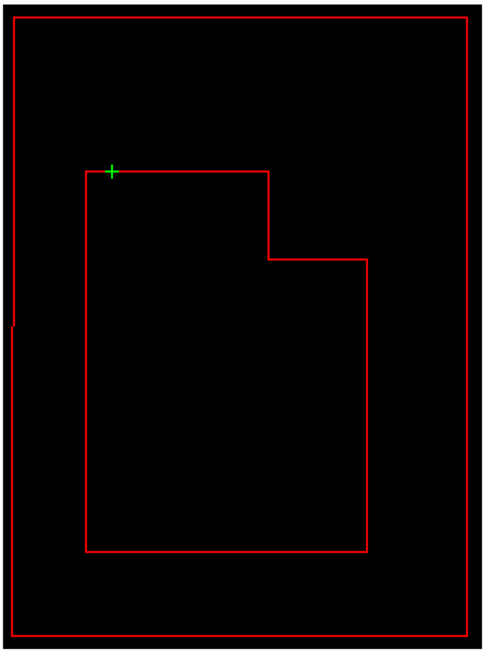
STEP 2:

The user enters a chainage value where a point should be placed followed by the <Enter> key. This chainage is with respect to the selected string.



STEP 3:

A new one point string is created at the positioned on the string at the given chainage.



To create another point, the user is prompted for the string as per **STEP 1**.

15.9.7 Bearing and distance

Position of option on menu: **CAD =>Point =>Bearing and distance**

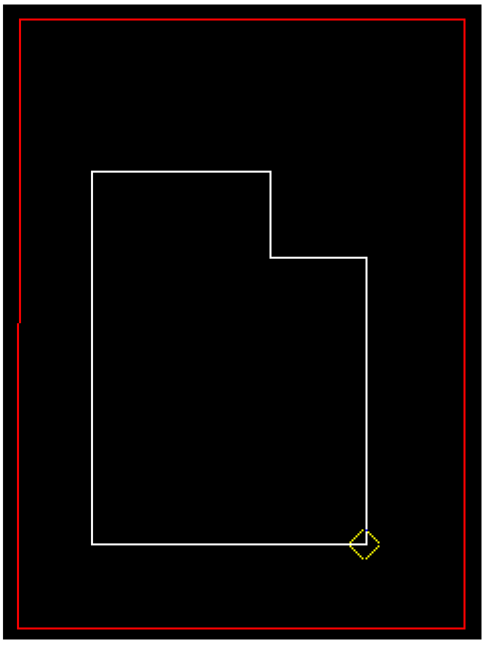
or by selection of appropriate icon from the toolbar. 

This option calculates a point string that is located a given bearing and distance from a start position.

On selecting **Bearing and distance**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

A start position is selected and accepted.

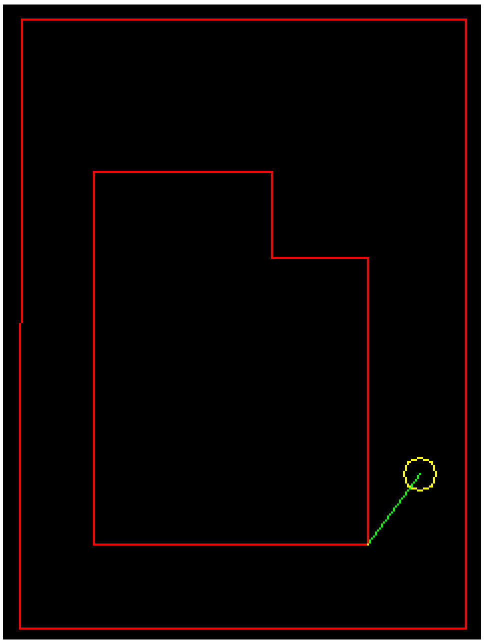


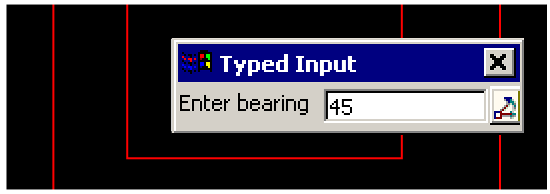
STEP 2:

A bearing is selected with the mouse or entered in via the keyboard. For typed entry, simply start typing or press the space bar to bring up the bearing entry box. The value is entered into the input box followed by the enter key.

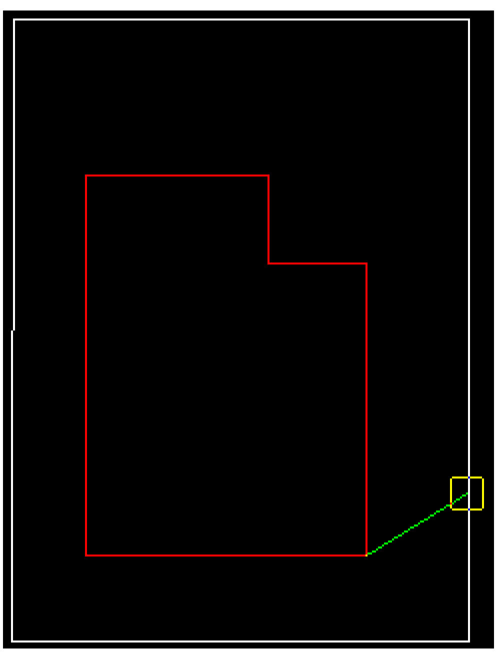
The line drawn represents the bearing value and changes with movement of the mouse. If the user wants to see what the current value of the bearing is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the point or the input box can be closed and the rubber banding (graphically changing) of the bearing continued.

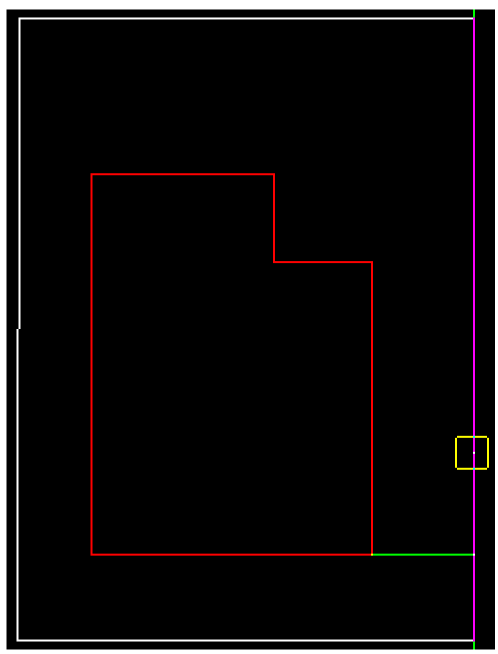
Note: **The Page up and page down keys can be used when the input angle box comes up to add or subtract intervals of 90 degrees.**





This option also allows the definition of the bearing by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next. The example shown below is the perpendicular case.

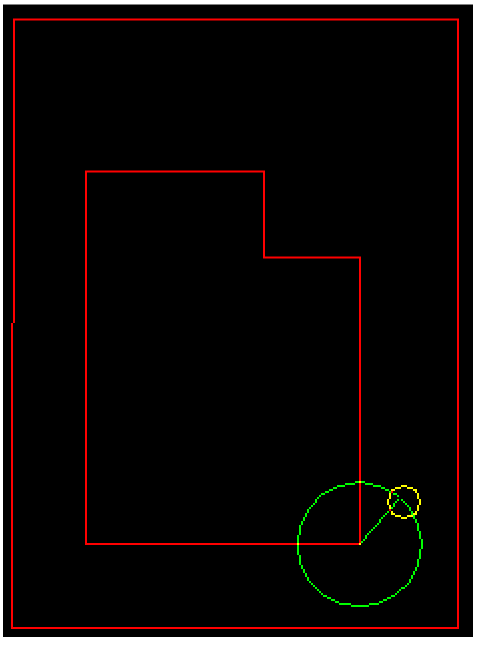


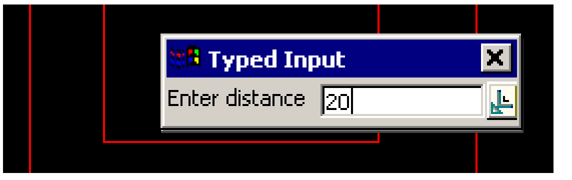


STEP 3:

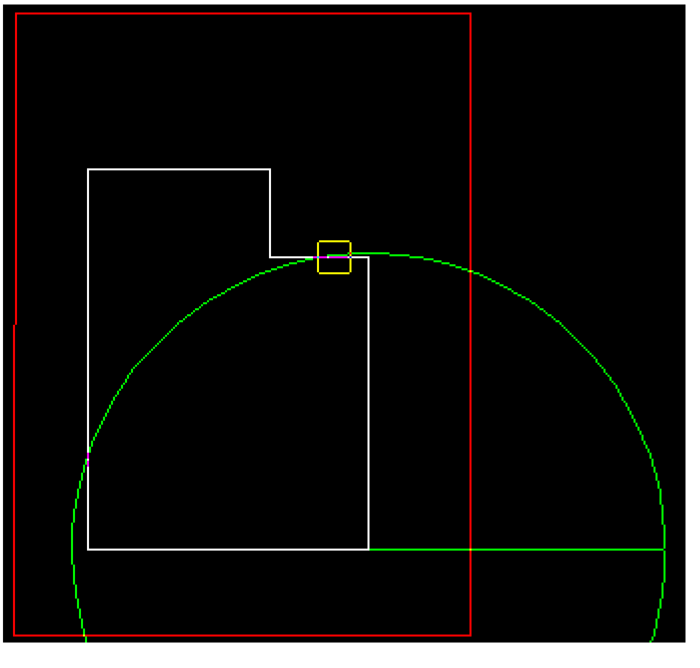
A distance is selected and accepted with the mouse or entered in via the keyboard. For typed entry, simply start typing or press the space bar to bring up the distance entry box. The value is entered into the input box followed by the Enter key.

The circle drawn represents the distance value and changes with movement of the mouse. If the user wants to see what the current value of the distance is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the point or the input box can be closed and the rubber banding (graphically changing) of the circle continued.

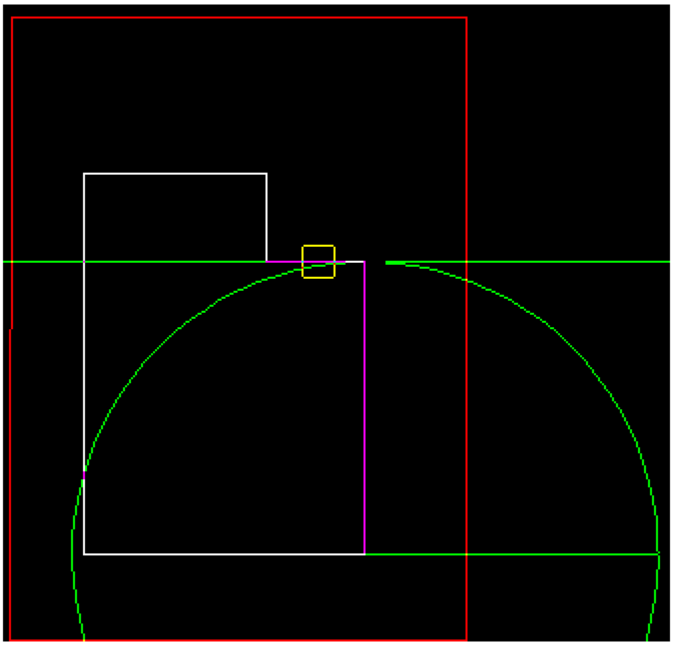




This option also allows the definition of the distance by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next. The example shown below is the perpendicular case.

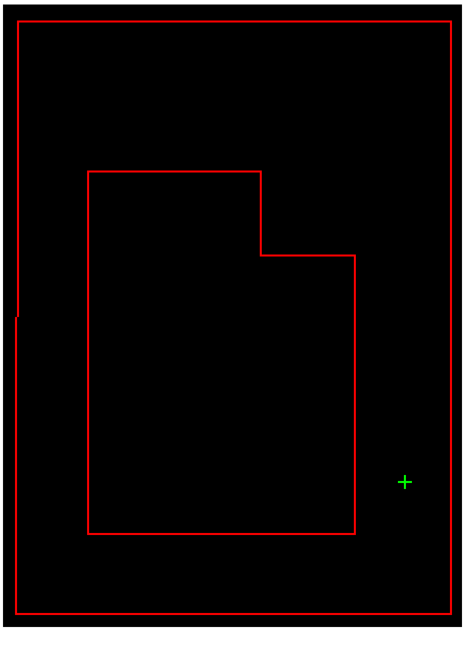


The perpendicular distance shown below by the purple line will be use at the defined bearing to create the point.

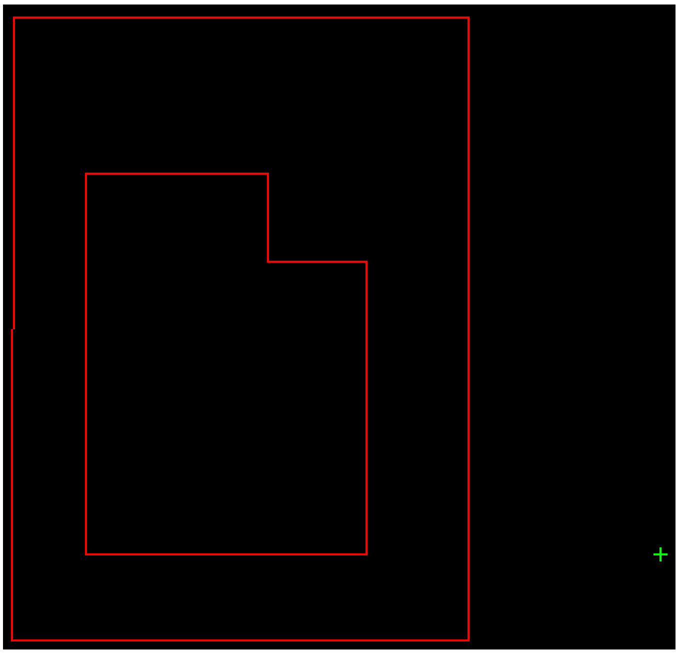


STEP 4:

A point string is created using the information supplied. By entry into the input boxes:



Or by using the perpendicular/tangential tools:



15.9.8 Offset

Position of option on menu: **CAD =>Point =>Offset**

or by selection of appropriate icon from the toolbar.

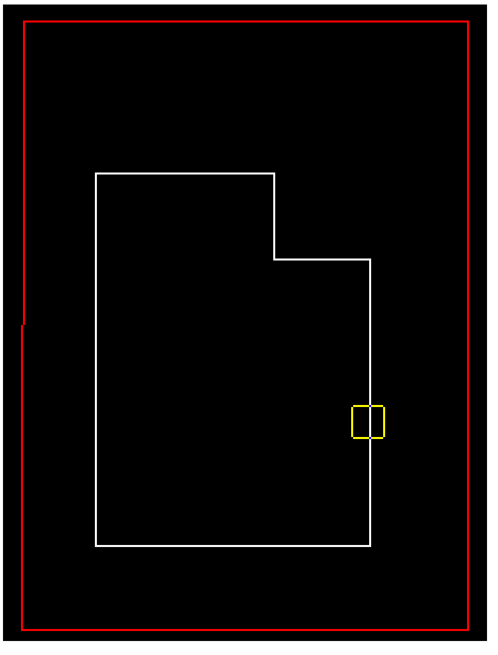
This option creates a one point string that is located by reference to a string, a control point, a chainage distance along the string from the control point and an offset to the selected string.

After selection and acceptance of a string, a control point is selected and accepted. This point is dropped perpendicular onto the string. The distance along the string is measured from this dropped point. Positive distances are in the direction that the string was picked. Finally a offset to the string can be specified for the placement of the new point string.

On selecting **Offset**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

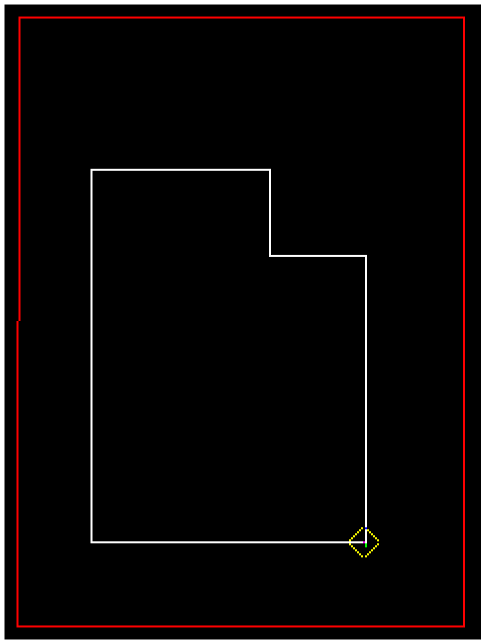
STEP 1:

The user selects and accepts a string with direction. This defines what side the offset applies to.



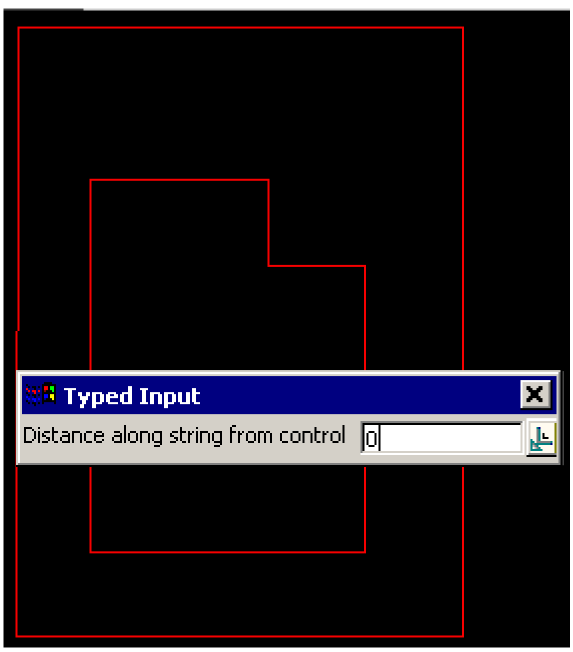
STEP 2:

The user picks and accepts a control point to be dropped onto the selected string.



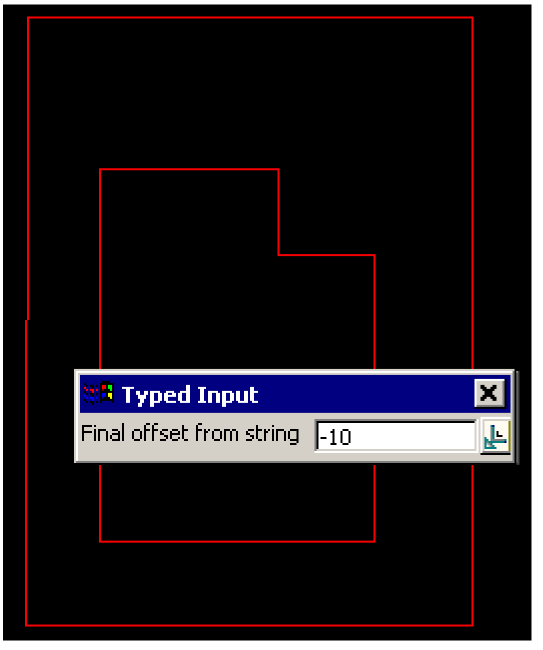
STEP 3:

A distance along from the dropped point is specified. Positive distances are in the direction of the string selection pick. The value is entered into the input box followed by the Enter key.



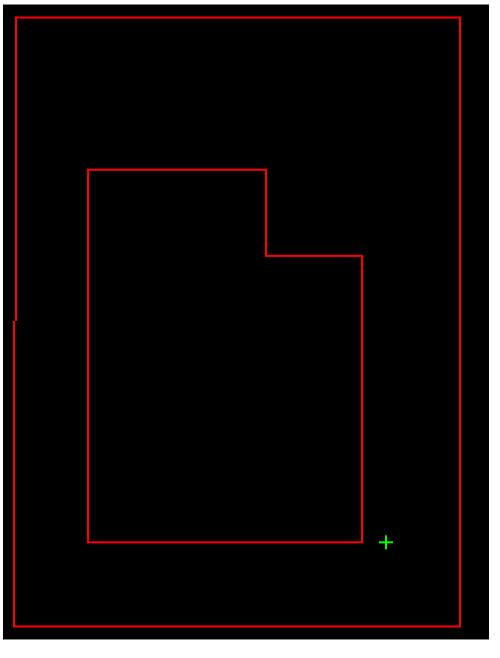
STEP 4:

An offset relative to the selected string (and direction) is specified in the input box followed by the enter key.



STEP 5:

A point string is created using the information supplied.



15.9.9 Chainage offset

This option creates a one point string that is located perpendicular to the reference string with a defined offset.

STEP 1:

The user selects the reference string with direction.

STEP 2:

The user specifies a chainage on the reference string.

STEP 3:

The user specifies an offset distance from the reference string.

A one point string is created using the information supplied.

15.9.10 Chainage offset extended

Is similar to chainage offset except the specified chainage can be extended beyond the start and end chainages of the selected string.

15.9.11 Deflection

Position of option on menu: **CAD =>Point =>Deflection**

or by selection of appropriate icon from the toolbar. 

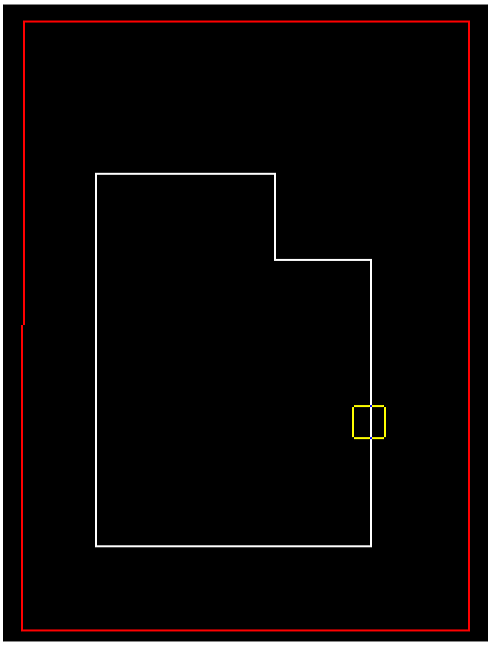
This option creates a one point string that is located by reference to a string, a control point, a distance along the string from the control point, a deflection angle and deflection distance.

After selection of a string, a control point is selected. This point is dropped perpendicular onto the string. A distance along the string can be entered to move the measure point. Positive distances are in the direction that the string was picked. A deflection angle is specified which is a clockwise angle from the measure point. The deflection distance is the distance from the measure point to the point which is to be created.

On selecting **Deflection**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

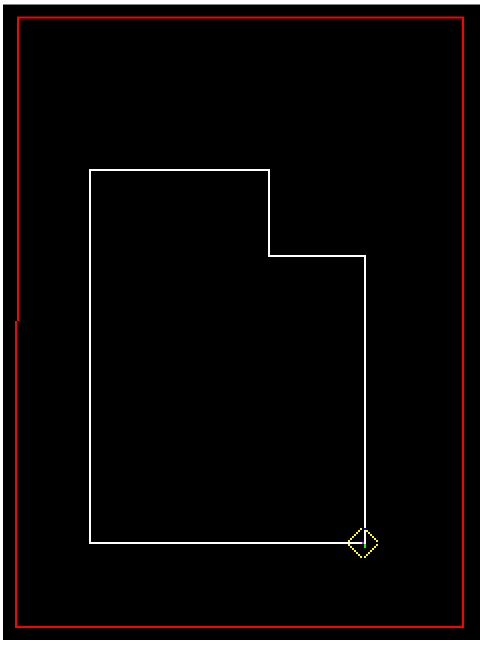
STEP 1:

The user selects and accepts a string with direction.



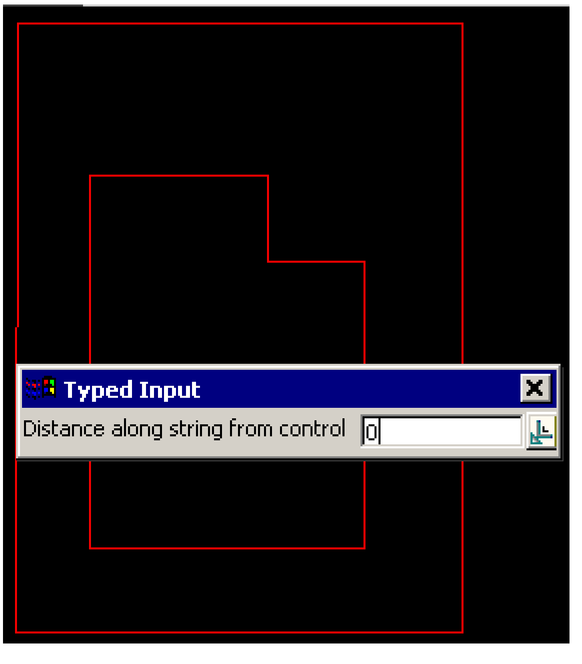
STEP 2:

The user picks and accepts a control point to be dropped onto the selected string.



STEP 3:

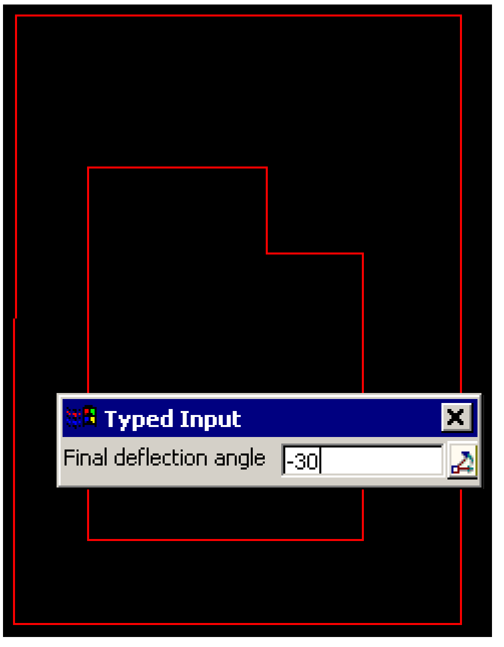
A distance along from the dropped point is specified. Positive distances are in the direction of the string selection pick. The value is entered into the input box followed by the enter key.



STEP 4:

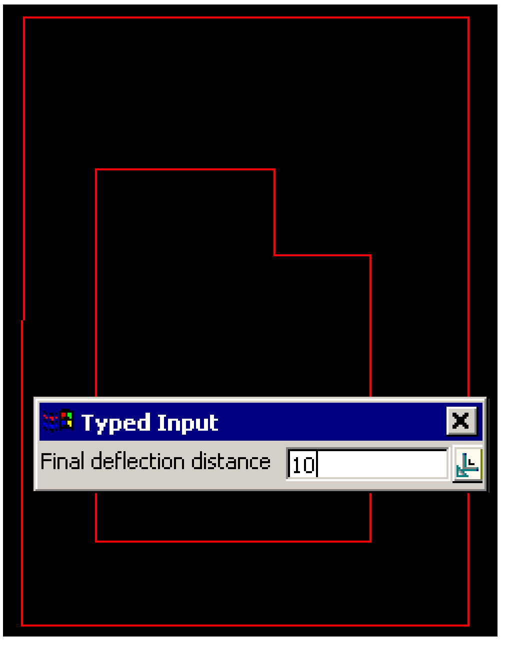
A deflection angle is specified. This angle is clockwise, relative to the direction of the string selection pick. The value is entered into the input box followed by the enter key.

Note: **The Page up and page down keys can be used when the input angle box comes up to add or subtract intervals of 90 degrees.**



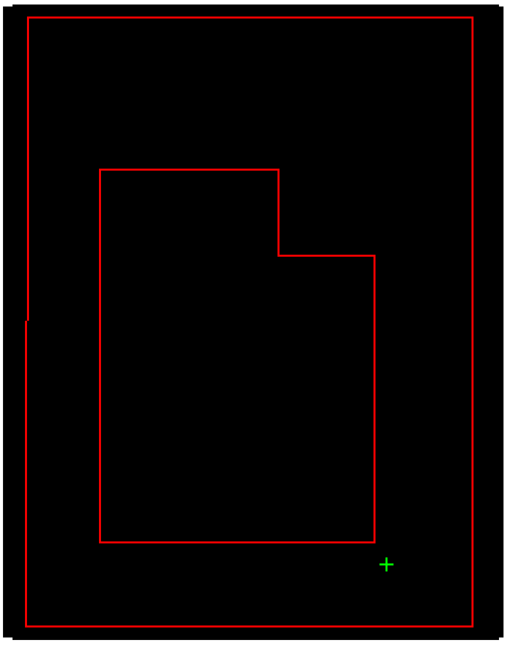
STEP 5:

A deflection distance is supplied. Positive is in the direction of the string selection pick. The value is entered into the input box followed by the enter key.



STEP 6:

A point string is created using the information supplied.



15.9.12 Drop Perpendicular

Position of option on menu: **CAD =>Point => Drop perpendicular**

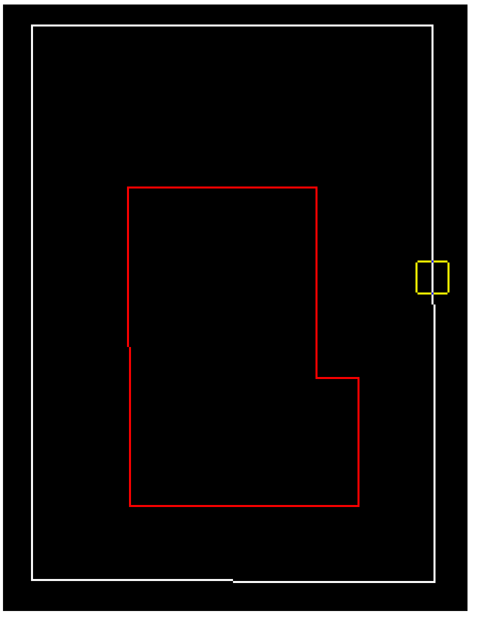
or by selection of appropriate icon from the toolbar. 

This option creates a one point string by dropping from a user selected position perpendicularly onto a user selected string.

On selecting **Drop perpendicular**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

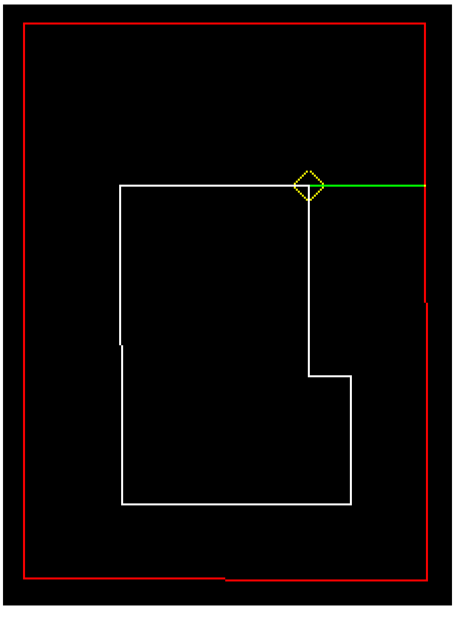
STEP 1:

The user selects a string and accepts (with direction) to have the point dropped onto.



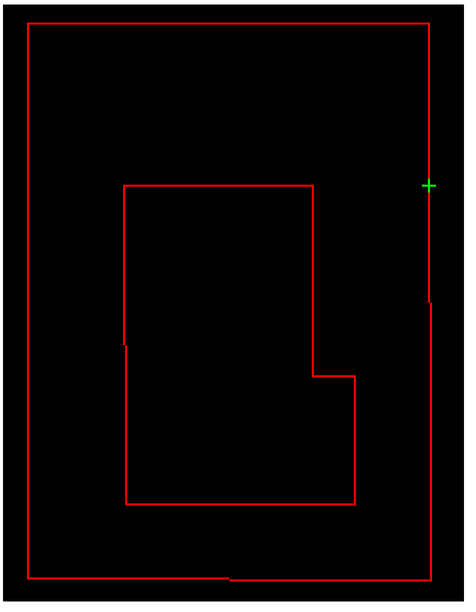
STEP 2:

The user picks and accepts a position to drop onto the nominated string



STEP 3:

A point string is created at the perpendicular drop point.



15.9.13 Angle

This option creates a one point string that is located by projecting a selected position back to a string by a specified angle.

STEP 1:

User selects and accepts a string to have the point projected onto.

STEP 2:

User specifies an angle for the projection.

STEP 3:

User selects a position on the screen to be projected back to the string.

STEP 4:

A one point string is created, which form the nominated angle with the selected position on **STEP 3.**

15.9.14 Projection

Position of option on menu: **CAD =>Point =>Projection**

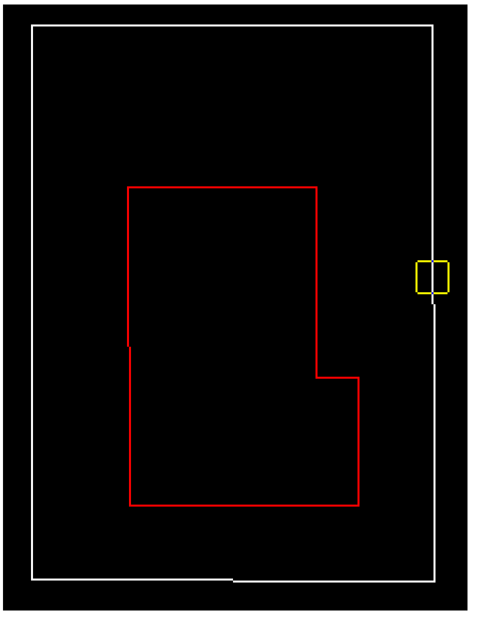
or by selection of appropriate icon from the toolbar. 

This option creates a one point string that is located by firstly dropping a point onto a string and then giving a distance along the string from the dropped point.

On selecting **Projection**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

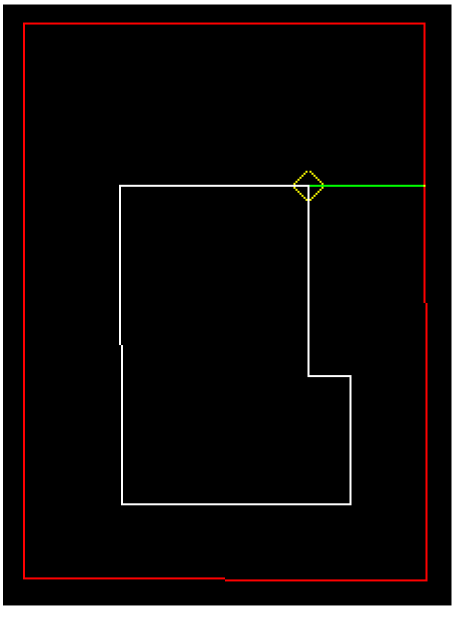
STEP 1:

The user selects and accepts a string (with direction) to have the point dropped onto



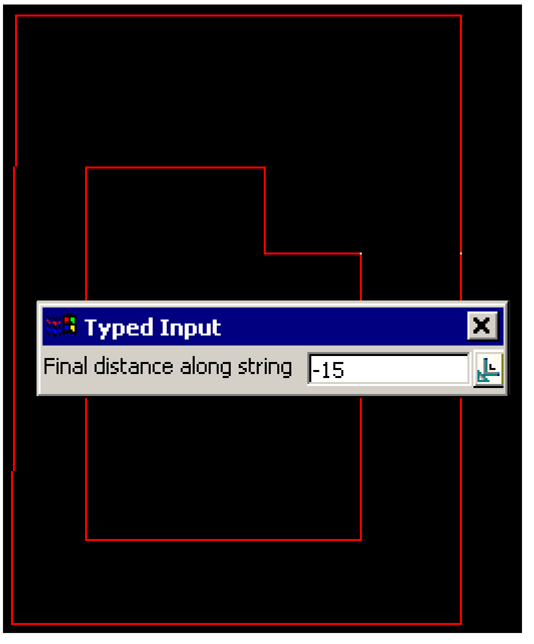
STEP 2:

The user picks and accepts a position to drop onto the nominated string



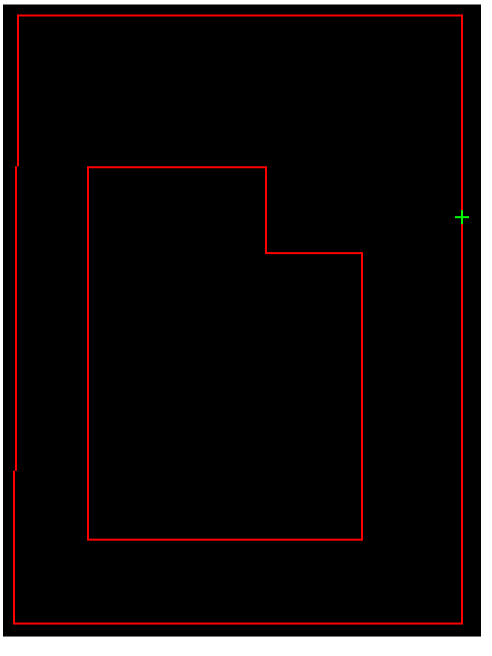
STEP 3:

The user is prompted for the distance along the string. Positive distances are in the direction that the string was picked. A value is entered into the input box followed by the enter key.



STEP 4:

The point string is created the nominated distance along the string from the dropped point.



15.9.15 Between Points

Position of option on menu: **CAD =>Point =>Between points**

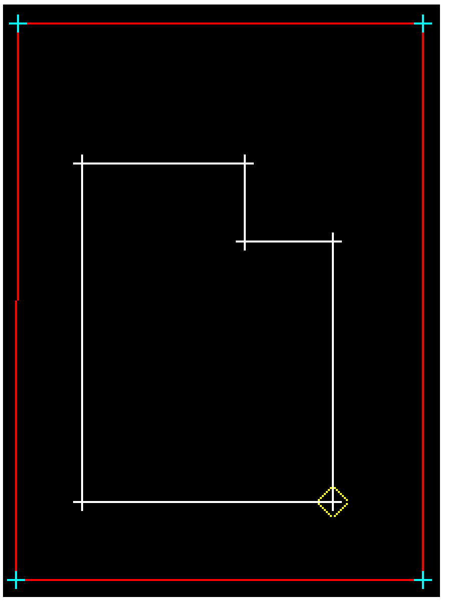
or by selection of appropriate icon from the toolbar. 

This option creates a one point string that is on the line between two selected positions and a given distance from the first point.

On selecting **Between points**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

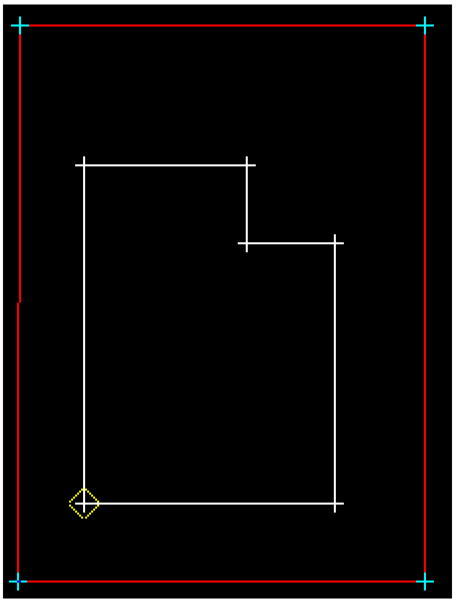
STEP 1:

The user selects a start point. Specification of a position can be done by the direct input of the xyz coordinate of the point by pressing the space bar to bring up the enter XYZ panel or by typing of the value to bring up the XYZ panel. The user can also select a point with the mouse and accepts that point (Middle mouse button or enter).



STEP 2:

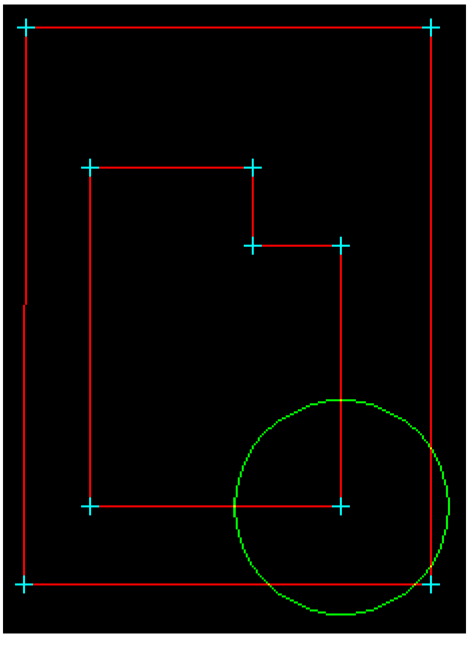
The user picks a 2nd point and accepts that point (Middle mouse button or enter) to define the reference line.

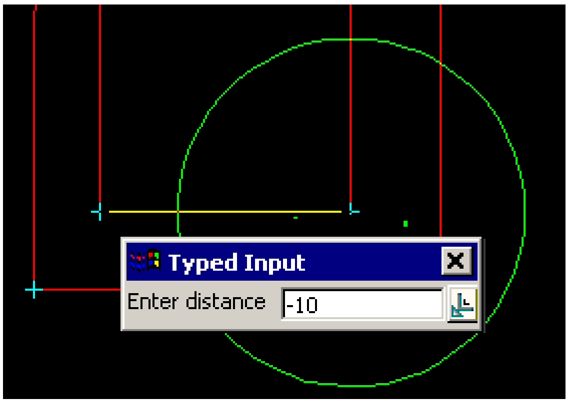


STEP 3:

A distance from the 1st point to create the point is given either by selection with the mouse or by typing a value. To type a value, simply start typing and the input box for the distance will appear. Alternatively you can press the space bar to bring up the input box. Enter the value and then the enter key.

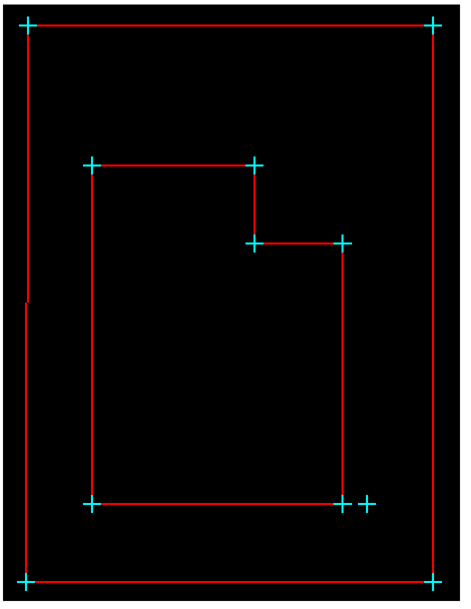
The circle drawn represents the distance value and changes with movement of the mouse. If the user wants to see what the current value of the distance is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the point or the input box can be closed and the rubber banding (graphically changing) of the circle continued.





STEP 4:

A point is created on the segment specified by the selection of the 1st and 2nd points, at the nominated distance. Negative distances can be entered as in this example.



15.9.16 Between points 3d

Position of option on menu: **CAD =>Point =>Between points 3d**

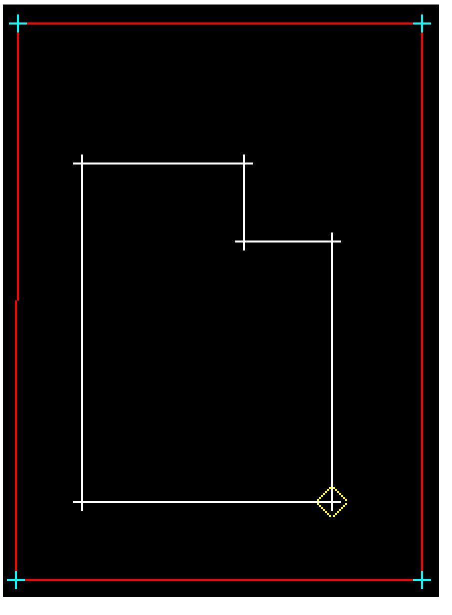
or by selection of appropriate icon from the toolbar.

This option creates a one point string that is on the line between two selected positions and a given distance from the first point. The z-value of the string is interpolated from the two selected positions.

On selecting **Between points 3d**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

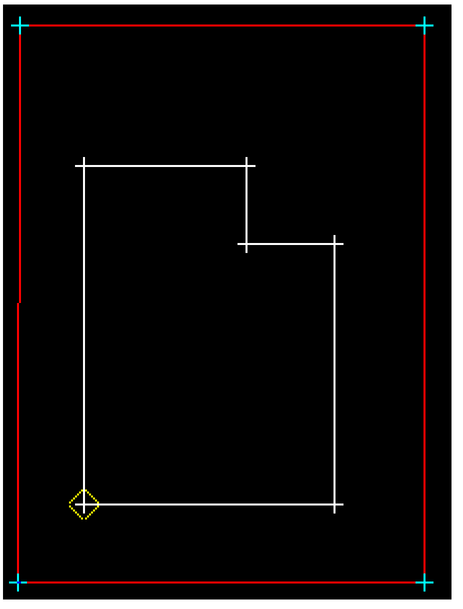
STEP 1:

The user selects a start point. Specification of a position can be done by the direct input of the xyz coordinate of the point by pressing the space bar to bring up the enter XYZ panel or by typing of the value to bring up the XYZ panel. The user can also select a point with the mouse and accepts that point (Middle mouse button or enter).



STEP 2:

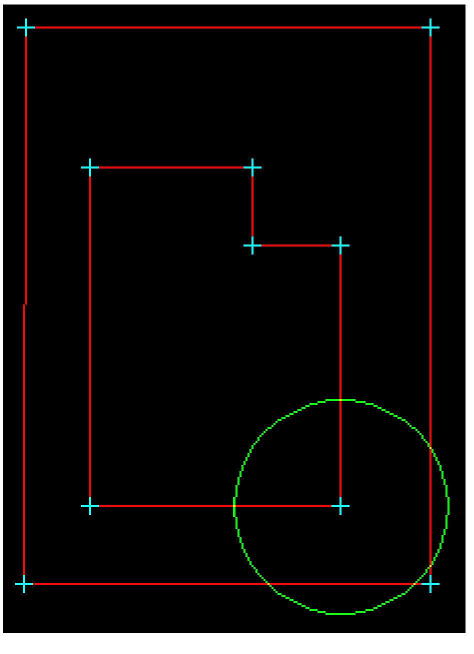
The user picks a 2nd point and accepts that point (Middle mouse button or enter) to define the reference line.

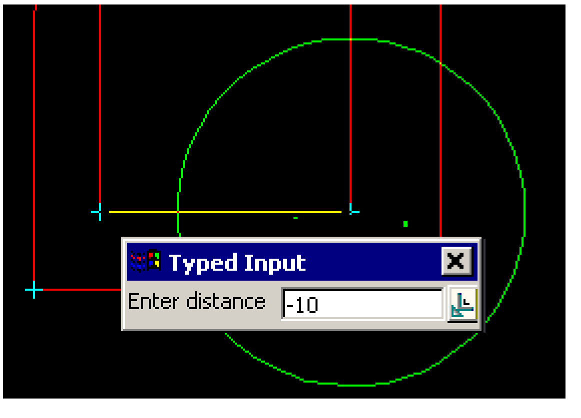


STEP 3:

A distance from the 1st point to create the point is given either by selection with the mouse or by typing a value. To type a value, simply start typing and the input box for the distance will appear. Alternatively you can press the space bar to bring up the input box. Enter the value and then the enter key.

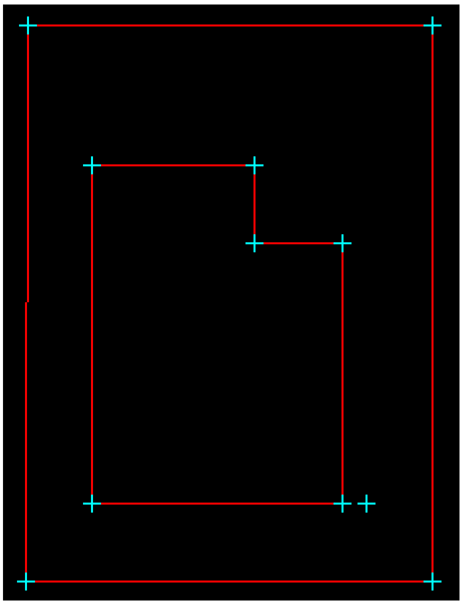
The circle drawn represents the distance value and changes with movement of the mouse. If the user wants to see what the current value of the distance is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the point or the input box can be closed and the rubber banding (graphically changing) of the circle continued.





STEP 4:

A point is created on the segment specified by the selection of the 1st and 2nd points, at the nominated distance. Negative distances can be entered as in this example.



15.9.17 Radiation

Position of option on menu: **CAD =>Point =>Radiation**

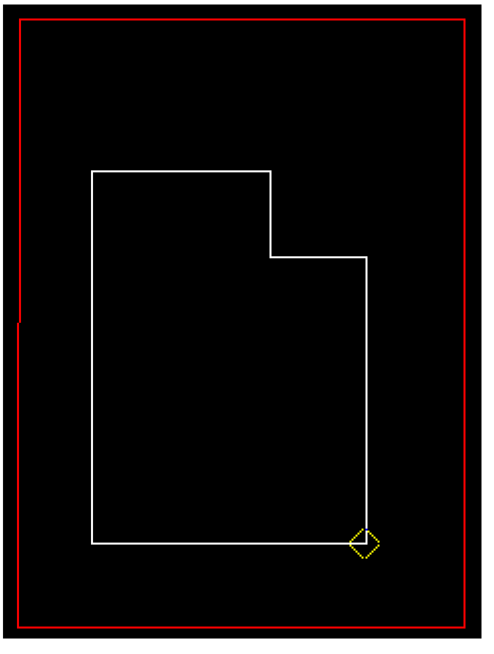
or by selection of appropriate icon from the toolbar. 

This option creates a series of one point strings that are located by given bearings and distances from a chosen start position.

On selecting **Radiation**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

A start position is selected and accepted.

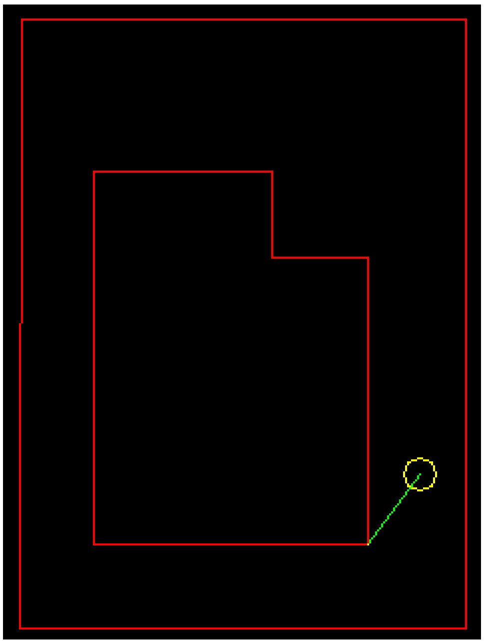


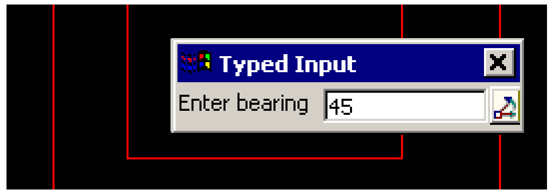
STEP 2:

A bearing is selected with the mouse or entered in via the keyboard. For typed entry, simply start typing or press the space bar to bring up the bearing entry box. The value is entered into the input box followed by the enter key.

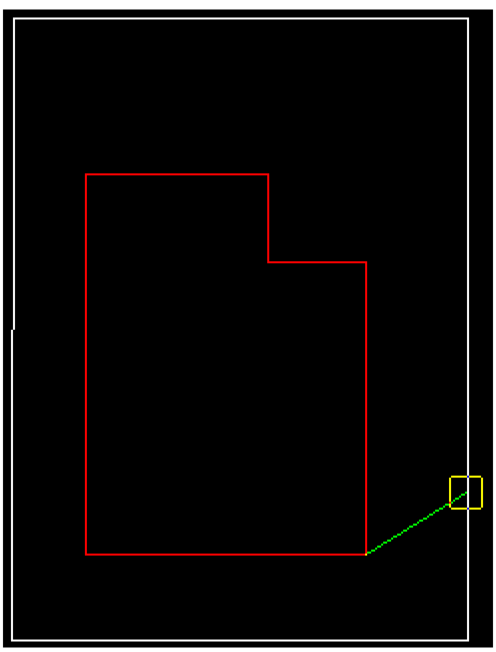
The line drawn represents the bearing value and changes with movement of the mouse. If the user wants to see what the current value of the bearing is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the point or the input box can be closed and the rubber banding (graphically changing) of the bearing continued.

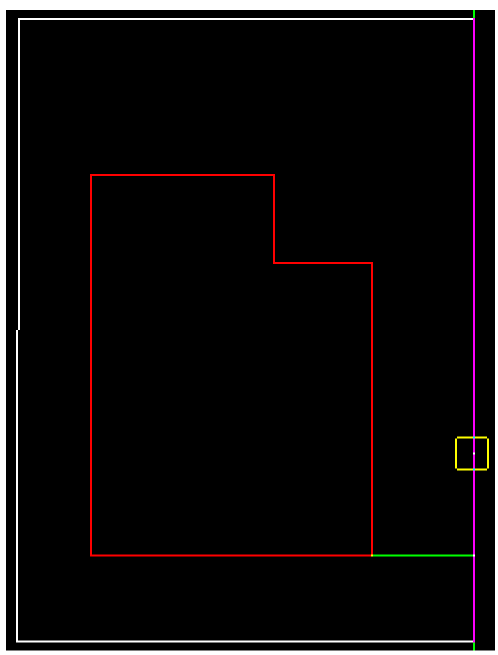
Note: **The Page up and page down keys can be used when the input angle box comes up to add or subtract intervals of 90 degrees.**





This option also allows the definition of the bearing by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next. The example shown below is the perpendicular case.

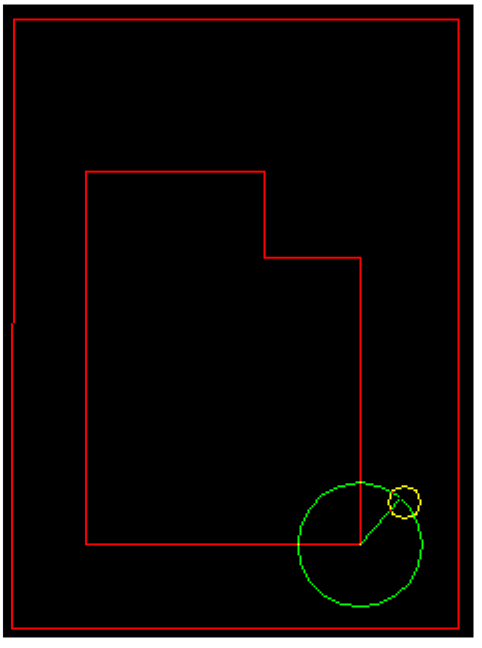


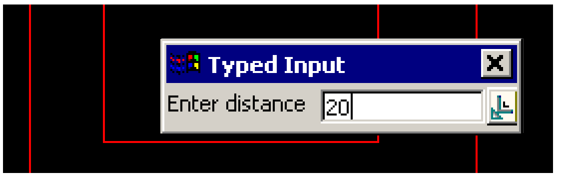


STEP 3:

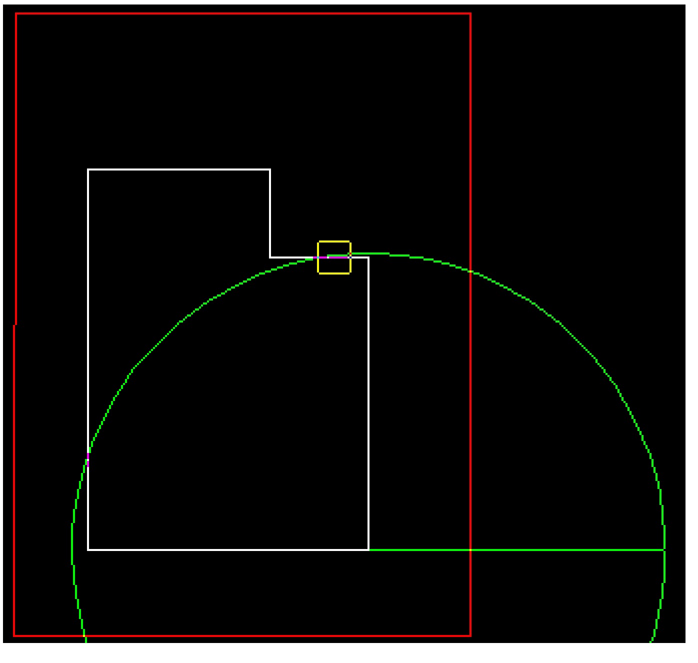
A distance from the start point to create the point is given either by selection with the mouse or by typing a value. To type a value, simply start typing and the input box for the distance will appear. Alternatively you can press the space bar to bring up the input box. The value is entered into the input box followed by the enter key.

The circle drawn represents the distance value and changes with movement of the mouse. If the user wants to see what the current value of the distance is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the point or the input box can be closed and the rubber banding (graphically changing) of the circle continued.

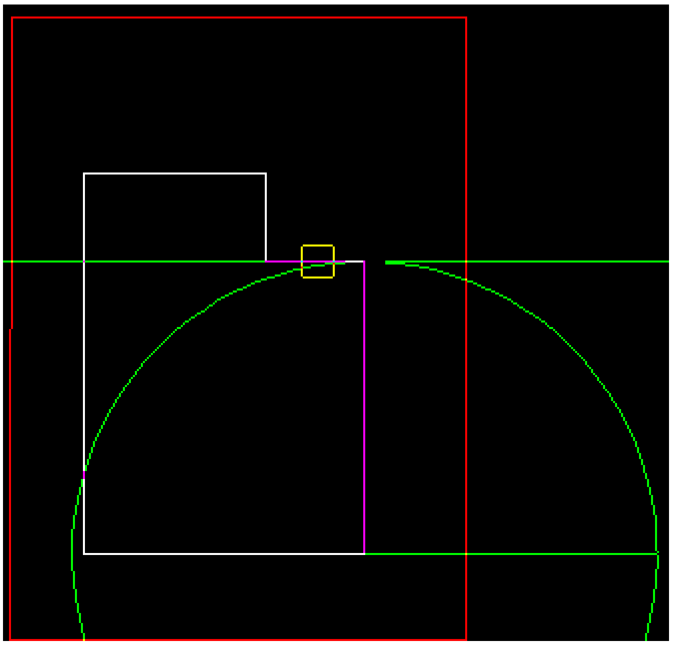




This option also allows the definition of the distance by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next. The example shown below is the perpendicular case.

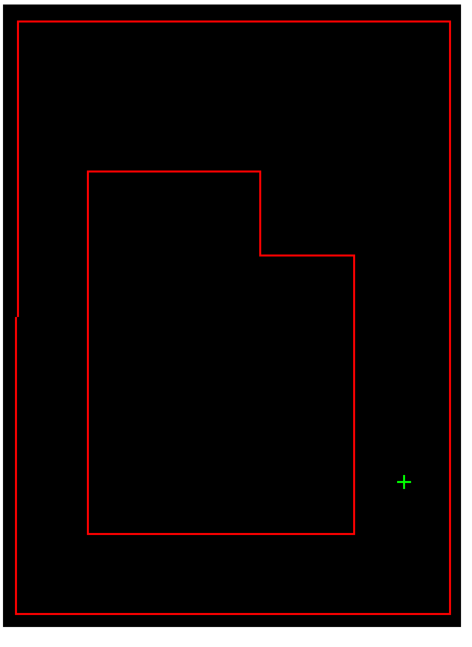


The perpendicular distance shown below by the purple line will be use at the defined bearing to create the point.



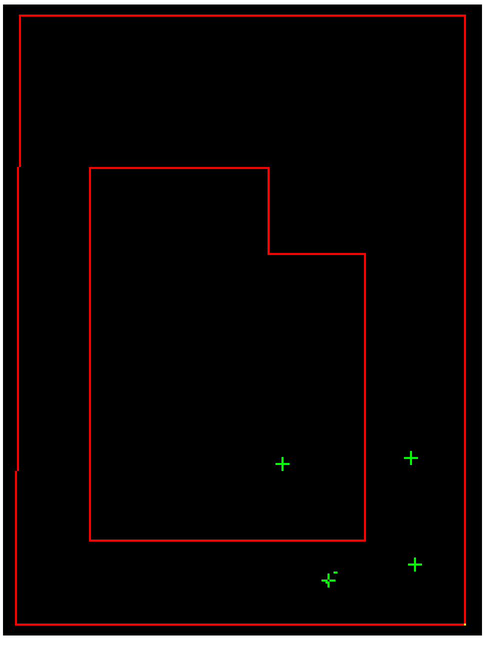
STEP 4:

A point string is created using the information supplied.



STEP 5

The next bearing and distance is prompted for as per step 3 and 4. Consequent points are created from the original start point selected.



15.9.18 Divide segment by number

Position of option on menu: **CAD =>Point =>Divide segment by number**

or by selection of appropriate icon from the toolbar.

This option creates one point strings which will break up a super string ***segment*** into equal pieces based on a user given number of intervals. The first and last points are not created since they are taken to be the end points of the selected segment. So for five intervals, four new one point strings are created. The option works for line or arc segments of a super string.

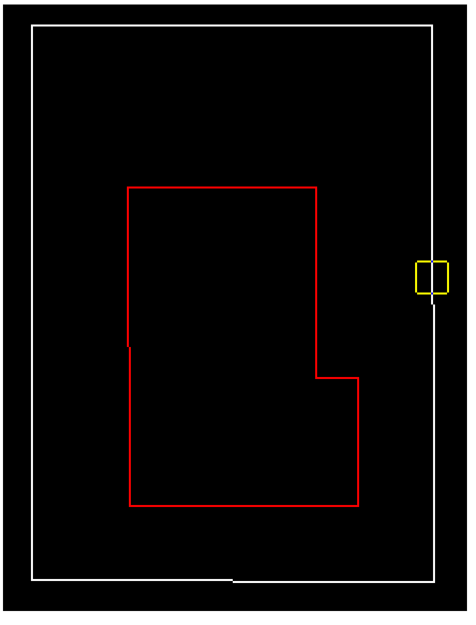
NOTES:

1. No vertices are inserted into the selected super string - the option **Strings =>Strings edits =>Segment strings** does that.
2. This option is for super strings only.

On selecting **Divide segment by number**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

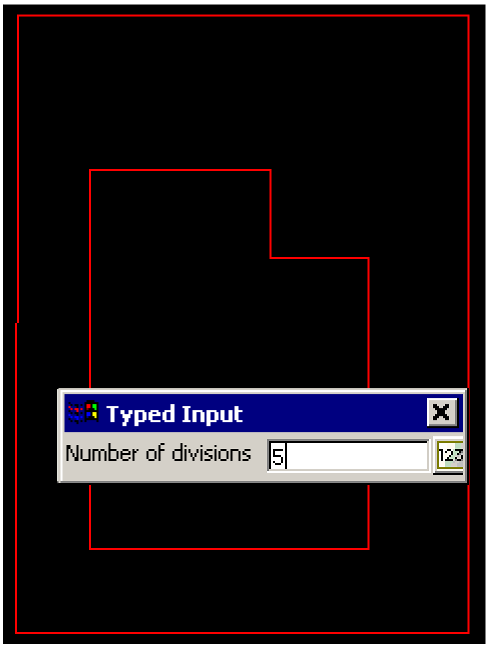
STEP 1:

A super string segment is selected and accepted.



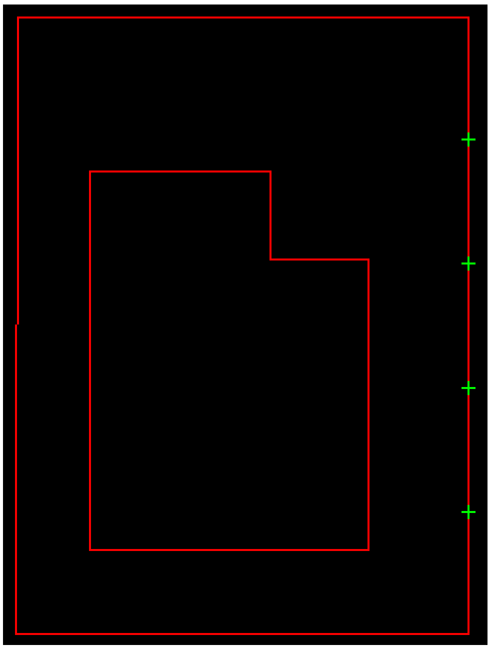
STEP 2:

The user is prompted for the number of divisions. The value is entered into the input box followed by the enter key.



STEP 3:

Given the above information, the new points are created.



15.9.19 Divide segment by chainage length

Position of option on menu: **CAD =>Point =>Divide segment by chainage length**

or by selection of appropriate icon from the toolbar.

This option creates one point strings which will break up a super string ***segment*** into pieces based on a user given chainage length. The process starts from the 1st point of the segment and continues until no more pieces can be created without going past the last point of the segment (the last piece may not be the correct length). This options works for line or arc segments of a super string.

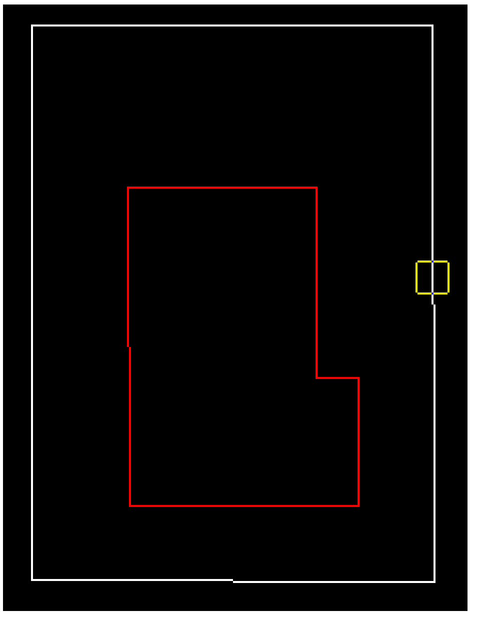
NOTES:

1. No vertices are inserted into the selected super string - the option **Strings =>Strings edits =>Segment strings** does that.
2. This option is for super strings only.

On selecting **Divide segment by chainage length**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

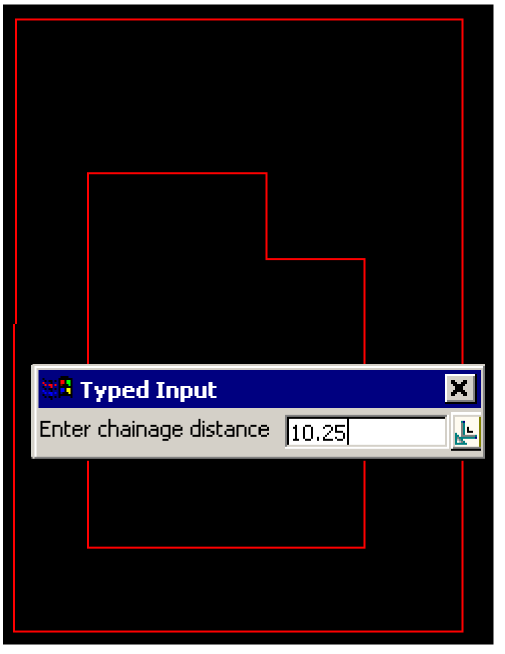
STEP 1:

A super string segment is selected and accepted.



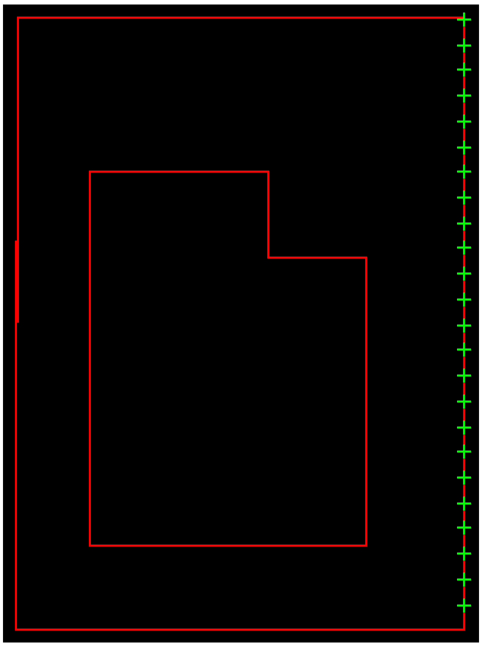
STEP 2:

The user is prompted for the chainage distance. The value is entered into the input box followed by the enter key.



STEP 3:

Given the above information, the new points are created.



15.9.20 Divide string by number

Position of option on menu: **CAD =>Point =>Divide string by number**

or by selection of appropriate icon from the toolbar. 

This option creates one point strings which will break up a super string into equal pieces based on a user given number of divisions. The first and last points are not created since they are taken to be the end points of the selected string. So for five intervals, four new one point strings are created.

NOTES:

1. No vertices are inserted into the selected super string - the option **Strings =>Strings edits =>Segment strings** does that.
2. This option is for super strings only.

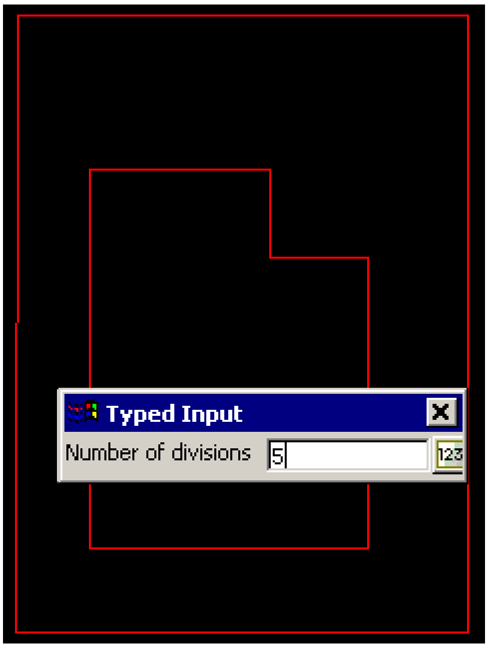
On selecting **Divide string by number**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

A super string is selected and accepted.

STEP 2:

The user is prompted for the number of divisions. The value is entered into the input box followed by the enter key.



STEP 3:

Given the above information, the new points are created.

15.9.21 Divide string by chainage length

Position of option on menu: **CAD =>Point =>Divide string by chainage length**

or by selection of appropriate icon from the toolbar. 

This option creates one point strings which will break up a *super string* into pieces based on a user given chainage length. The process starts from the 1st vertex of the strings and continues until no more pieces can be created without going past the last vertex of the string (the last piece may not be the correct length).

NOTES:

1. No vertices are inserted into the selected super string - the option **Strings =>Strings edits =>Segment strings** does that.
2. This option is for super strings only.

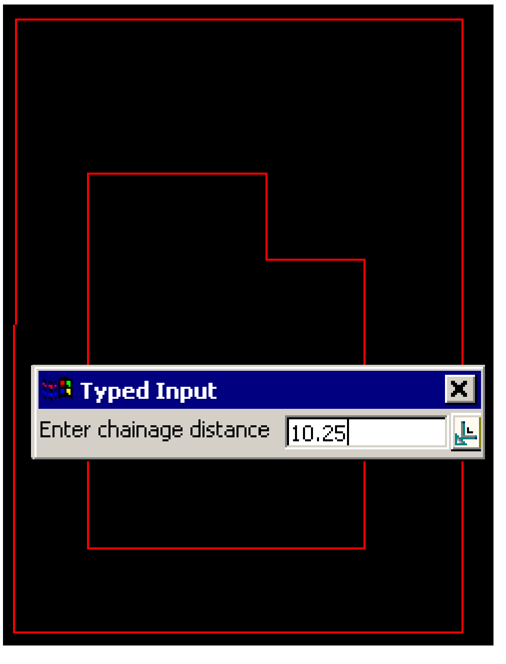
On selecting **Divide string by chainage length**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

A super string is selected and accepted.

STEP 2:

The user is prompted for the chainage distance. The value is entered into the input box followed by the <Enter> key.



STEP 3:

Given the above information, the new points are created.

15.9.22 String IPs

Position of option on menu: **CAD =>Point =>String IPs**

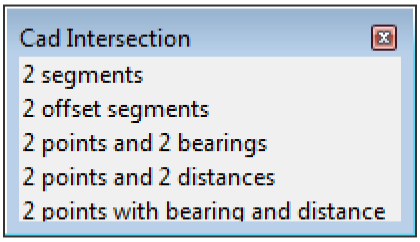
This option creates a separate point at each vertex of a selected string.

On selecting **String IPs** the user is prompted to select a string and a new point (a one vertex super string) is created for each vertex on the selected string. The model, colour, height, *etc.* for the new points are defined in the **Cad Control Bar**.

15.10 CAD Intersection

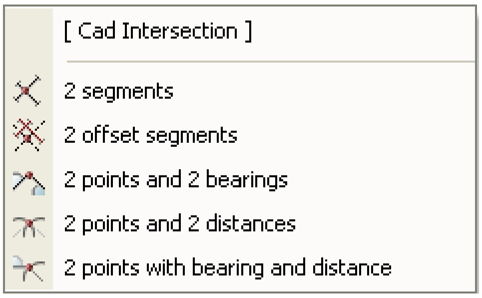
Position of option on menu: **CAD =>Intersection**

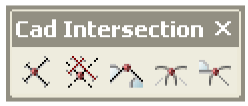
The **Intersection** walk-right menu is



Menu of Options to Create One Point Strings by Intersections

by intersecting two selected segments
by intersecting two offset selected segments
by given bearings from two selected positions
create a point with given distances from two selected positions
by selecting posn and bearing and another selected posn and dist





For the option 2 *segments*, go to 15.10.1 2 segments
2 offset segments 15.10.2 2 offset segments
2 points and 2 bearings 15.10.3 2 points and 2 bearings
2 points and 2 distances 15.10.4 2 points and 2 distances
2 points with bearing and distance 15.10.5 2 points with bearing and distance

15.10.1 2 segments

Position of option on menu: **CAD =>Intersection=> 2 segments**

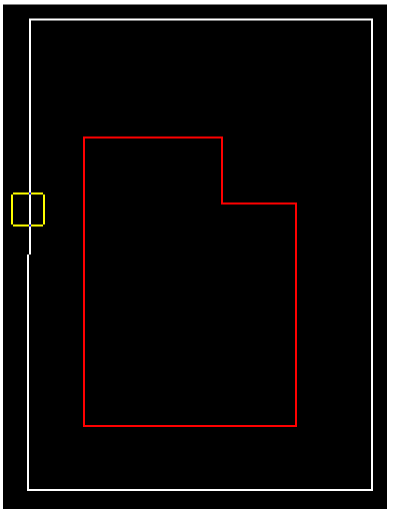
or by selection of appropriate icon from the toolbar.

This option creates a point at the intersection of the projections of two line or arc segments.

On selecting **2 segments**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

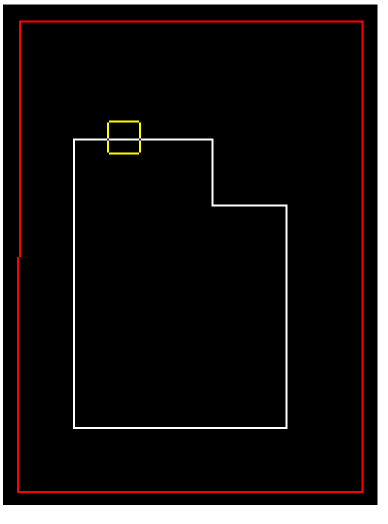
STEP 1:

The 1st segment is selected and accepted.



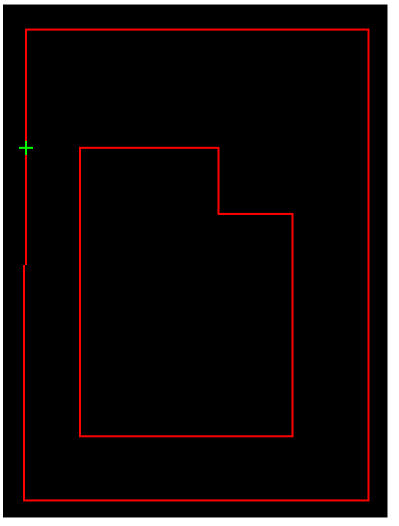
STEP 2:

The 2nd segment is selected and accepted.



STEP 3:

A point is created at the intersection of the two segments (if a solution exists). Note that the 2nd segment in this case has been projected to enable a solution to be calculated.



15.10.2 2 offset segments

Position of option on menu: CAD =>Intersection=> 2 offset segments

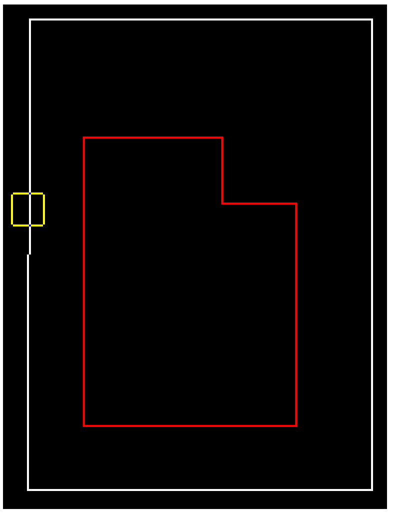
or by selection of appropriate icon from the toolbar.

This option creates a point at the intersection of the offsets of two selected segments.

On selecting **2 offset segments**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

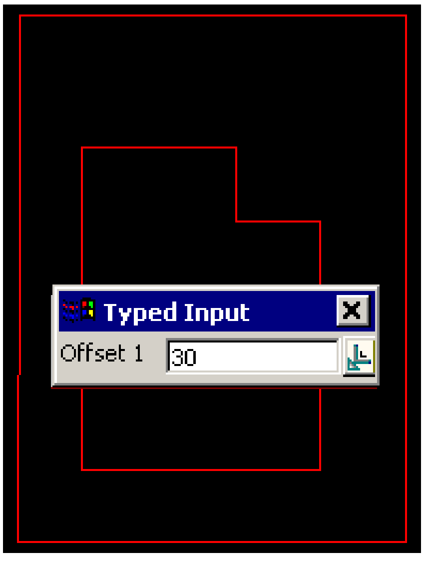
STEP 1:

The 1st segment is selected and accepted with direction. This sets the positive direction of the offset to the right of the direction of pick.



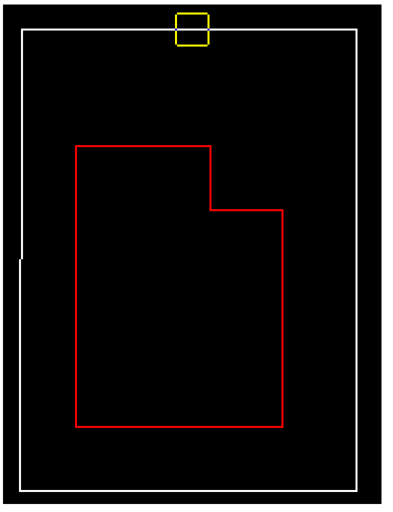
STEP 2:

The offset is given by into an offset input box. The positive direction is at 90 degrees to the direction of pick for the segment. The value is entered into the input box followed by the enter key.



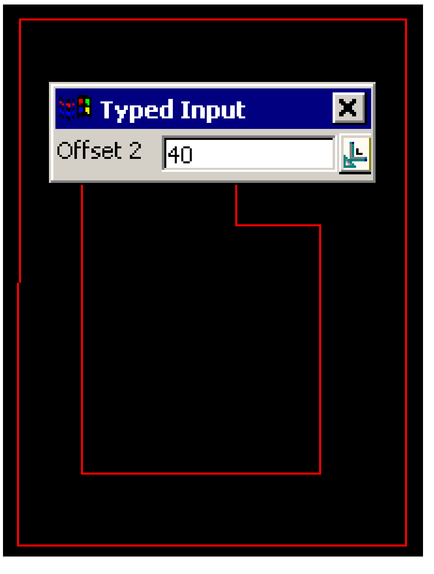
STEP 3:

The 2nd segment is selected with direction and accepted. This sets the positive direction of the offset to the right of the direction of pick.



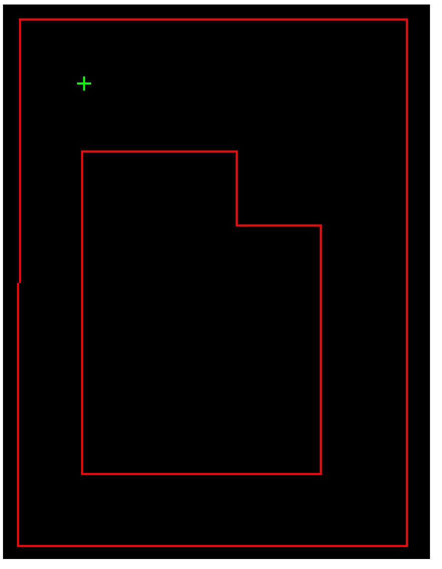
STEP 4:

The offset is given by into an offset input box. The positive direction is at 90 degrees to the direction of pick for the segment. The value is entered into the input box followed by the enter key.



STEP 5:

A point is created at the intersection of the projected lines offset to the segments (if a solution exists).



15.10.3 2 points and 2 bearings

Position of option on menu: **CAD =>Intersection=>2 points and 2 bearings**

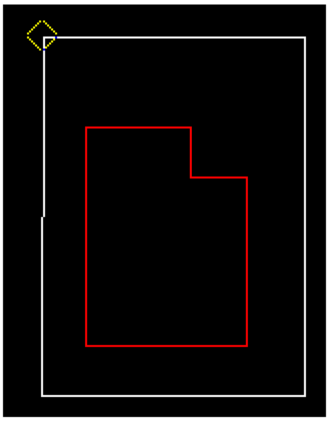
or by selection of appropriate icon from the toolbar.

This option creates a point by using two points and two bearings.

On selecting **2 points and 2 bearings**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

The 1st point is selected and accepted.

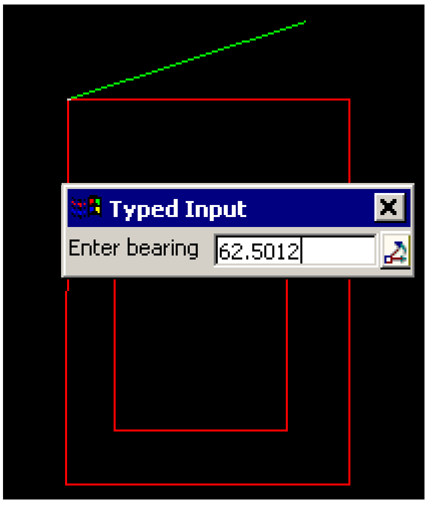


STEP 2:

A bearing from the 1st point to create the point is given either by selection with the mouse or by typing a value. To type a value, simply start typing and the input box for the bearing will appear. Alternatively you can press the space bar to bring up the input box. The value is entered into the input box followed by the enter key.

The line drawn represents the bearing value and changes with movement of the mouse. If the user wants to see what the current value of the bearing is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the point or the input box can be closed and the rubber banding (graphically changing) of the line continued.

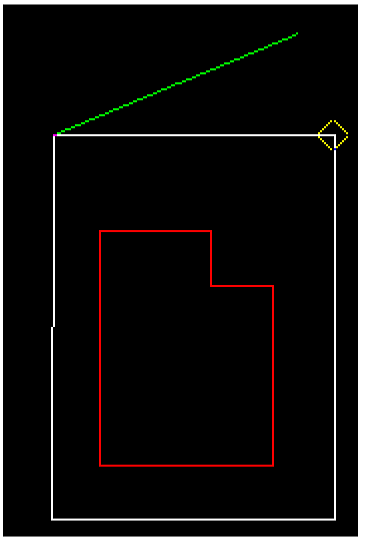
Note: **The Page up and page down keys can be used when the input angle box comes up to add or subtract intervals of 90 degrees.**



This option also allows the definition of the bearing by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next.

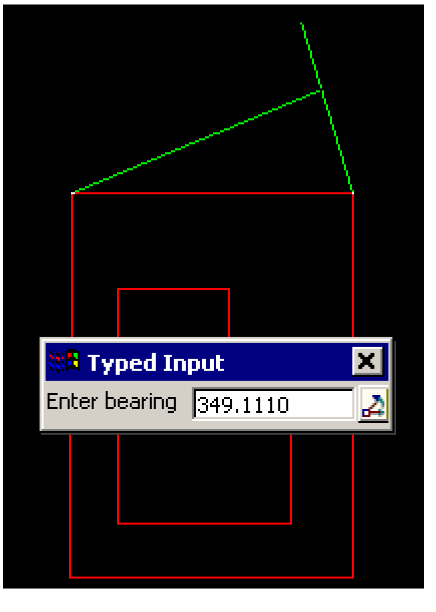
STEP 3:

The 2nd point to create the point from is selected and accepted.



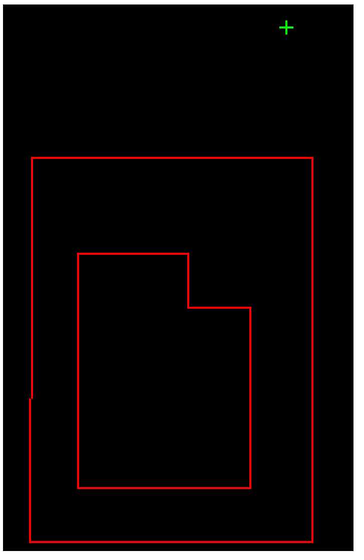
STEP 4:

A bearing from the 2nd point to create the point is given using the optional outlined in **STEP 2** above.



STEP 5:

The point is created if there is a valid solution.



15.10.4 2 points and 2 distances

Position of option on menu: **CAD =>Intersection=>2 points and 2 distances**

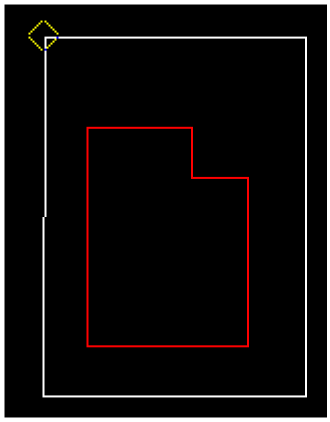
or by selection of appropriate icon from the toolbar.

This option creates a point using two points and two distances.

On selecting **2 points and 2 distances**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

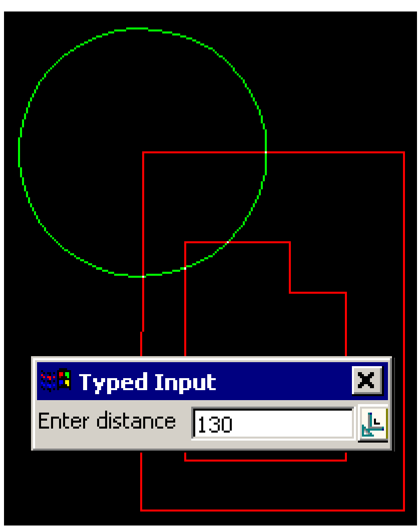
The 1st point is selected and accepted.



STEP 2:

A distance from the 1st point to create the point is given either by selection with the mouse or by typing a value. To type a value, simply start typing and the input box for the distance will appear. Alternatively you can press the space bar to bring up the input box. The value is entered into the input box followed by the enter key.

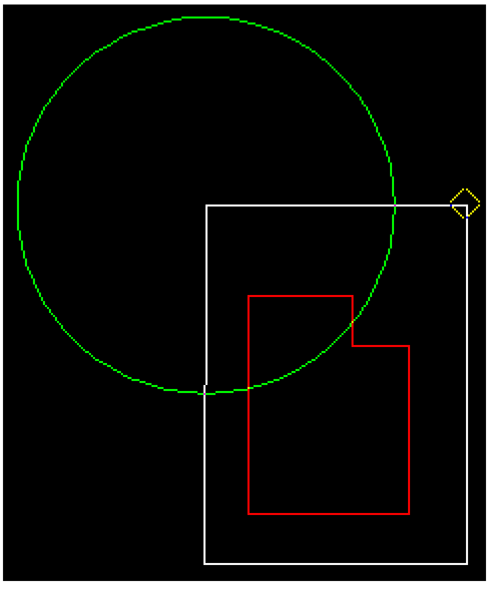
The circle drawn represents the distance value and changes with movement of the mouse. If the user wants to see what the current value of the distance is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the point or the input box can be closed and the rubber banding (graphically changing) of the circle continued.



This option also allows the definition of the distance by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next. The example shown below is the perpendicular case.

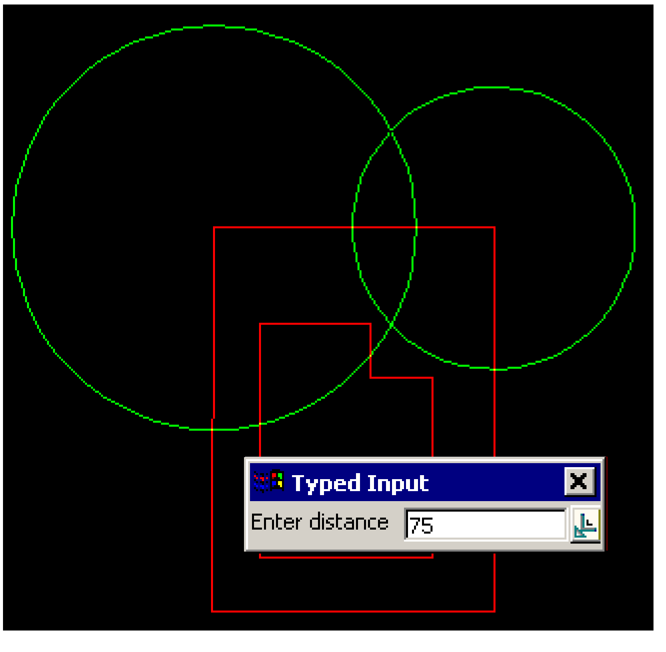
STEP 3:

The 2nd point to create the point from is selected.



STEP 4:

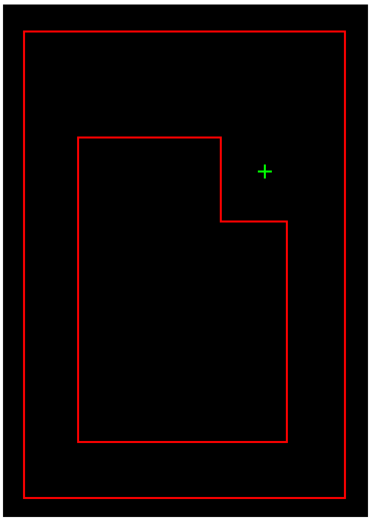
A distance from the 2nd point to create the point is given using the optional outlined in **STEP 2** above.



STEP 5:

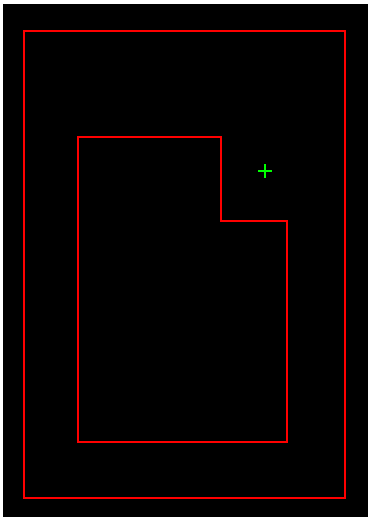
As there is two solutions, the user can select the correct one depending on the method of construction. This can be done by the direct entry of distances or by use of the mouse.

1. Distance entry. After the entry of the 1st distance, the 2nd point is selected. Following the selection of the 2nd point, the 2nd radius is shown (rubber banding). The user can select one of the two solutions by choosing with a **LB** mouse click over the approximate position of the required solution. The solution chosen is the closest solution to the selected point. The final radius can then be entered via the keyboard by simply starting typing which brings up the radius entry panel automatically. This panel can also be activated by pressing the space bar.
2. Use of the mouse. The 1st point is selected and the radius entered by using the mouse or by direct entry from the keyboard. The 2nd point is then selected and the 2nd radius is displayed (rubber banding). A solution can be chosen by selecting with a **LB** mouse click over the required solution. The final solution will be the closest one to the selection. **MB** to accept the intersection and create the point.



STEP 6:

The point is created if there is a valid solution.



15.10.5 2 points with bearing and distance

Position of option on menu: **CAD =>Intersection=>2 points with bearing and distance**

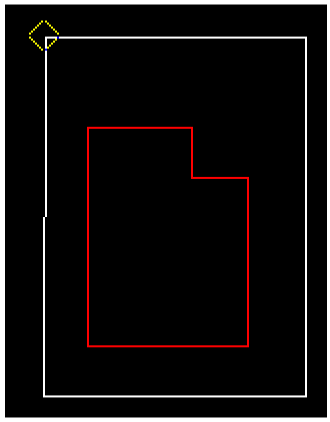
or by selection of appropriate icon from the toolbar. 

This option creates a point from a given point and a bearing, and a second point and a distance.

On selecting **2 points with bearing and distance**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

The 1st point is selected and accepted.

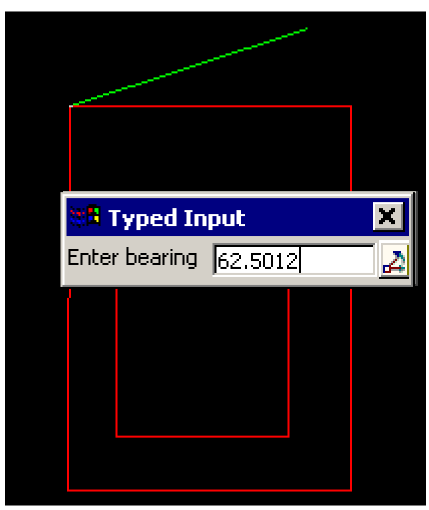


STEP 2:

A bearing from the 1st point to create the point is given either by selection with the mouse or by typing a value. To type a value, simply start typing and the input box for the bearing will appear. Alternatively you can press the space bar to bring up the input box. The value is entered into the input box followed by the enter key.

The line drawn represents the bearing value and changes with movement of the mouse. If the user wants to see what the current value of the bearing is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the point or the input box can be closed and the rubber banding (graphically changing) of the line continued.

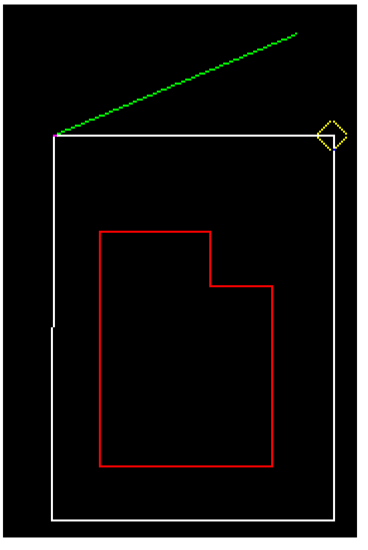
Note: **The Page up and page down keys can be used when the input angle box comes up to add or subtract intervals of 90 degrees.**



This option also allows the definition of the bearing by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next.

STEP 3:

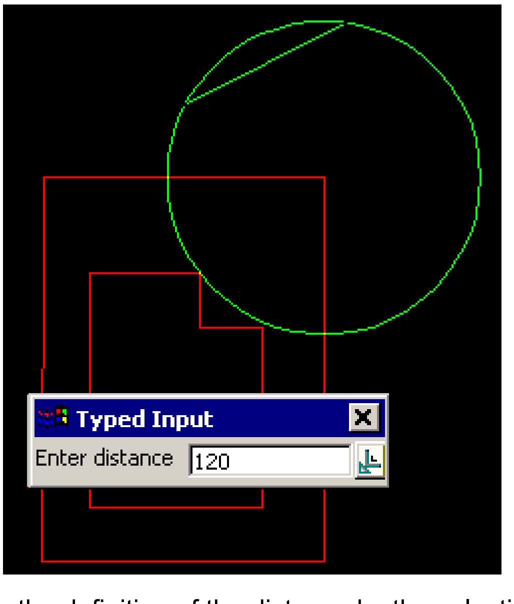
The 2nd point to create the point from is selected and accepted.



STEP 4:

A distance from the 2nd point to create the point is given either by selection with the mouse or by typing a value. To type a value, simply start typing and the input box for the distance will appear. Alternatively you can press the space bar to bring up the input box. The value is entered into the input box followed by the enter key.

The circle drawn represents the distance value and changes with movement of the mouse. If the user wants to see what the current value of the distance is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the point or the input box can be closed and the rubber banding (graphically changing) of the circle continued.

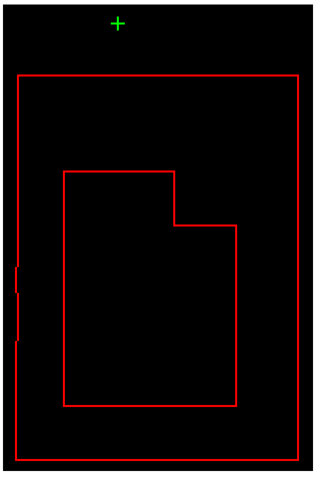


This option also allows the definition of the distance by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next. The example shown below is the perpendicular case.

A line is drawn between the two possible solutions.

STEP 5:

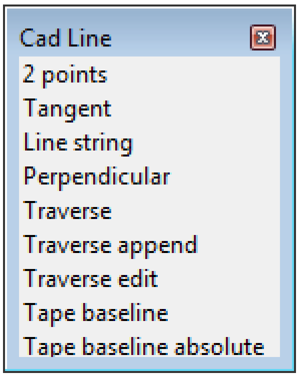
As there is two solutions, the user can select the correct one with the mouse.



15.11 CAD Line

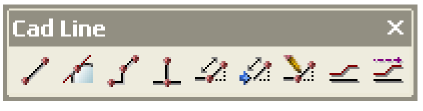
Position of option on menu: **CAD =>Line**

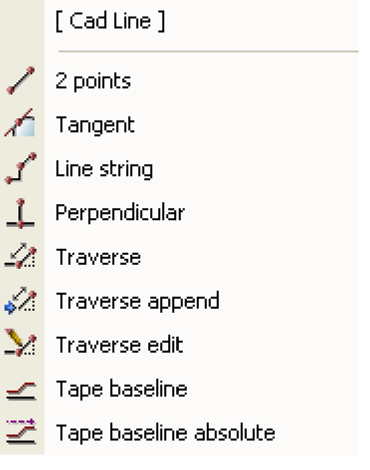
The **Line** walk-right menu is



Menu of Options to Create Strings

create a two vertex (point) super string
create tangent to two selected items (vertices and/or arcs)
create a many vertex super string
create a two points string perpendicular to a reference string
create many vertex super string by bearing distances from previous vertex
append to a super string using bearing distance entry
modify the bear/distance of a selected segment
create string with vertices a dist along and dist offset to a line
create string with vertices a dist along and dist offset absolute to a line





For the option *2 points*, go to 15.11.1 2 Points
Tangent 15.11.2 Tangent
Line string 15.11.3 Line string
Perpendicular 15.11.4 Perpendicular
Traverse 15.11.5 Traverse
Traverse append 15.11.6 Traverse append
Traverse edit 15.11.7 Traverse edit
Tape baseline 15.11.8 Tape baseline
Tape baseline absolute 15.11.9 Tape baseline absolute

15.11.1 2 Points

Position of option on menu: **CAD =>Line =>2 points**

or by selection of appropriate icon from the toolbar. 

This option creates two vertex super strings with a line between the two vertices.

On selecting **2 points**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

The user can select a position with the mouse and on accepting that point (Middle mouse button or enter) the point is created at the selected position. The model, colour, height etc. are defined in the **Cad Control Bar**.

The snap mode will influence the mouse selection. For example if cursor snap is on, the user can choose a position not yet defined. If point snap is on and the selection snaps to an existing point, the option will place another point at that location.

The user can also activate the selection menu used with the mouse (right button) that allows various positioning options.

Specification of a position can also be done by the direct input of the xyz coordinate of the point by pressing the space bar to bring up the enter XYZ panel. **NOTE:** The z value will default to the value entered into the **Cad Control Bar** whether or not it is specified in the XYZ box. If no height value exists in the **Cad Control Bar**

The 2nd point is selected in the same way as the 1st point. The line is created after successful selection and acceptance of the 2nd point. The option remains current so that a number of lines can be created.

15.11.2 Tangent

Position of option on menu: **CAD =>Line =>Tangent**

or by selection of appropriate icon from the toolbar. 

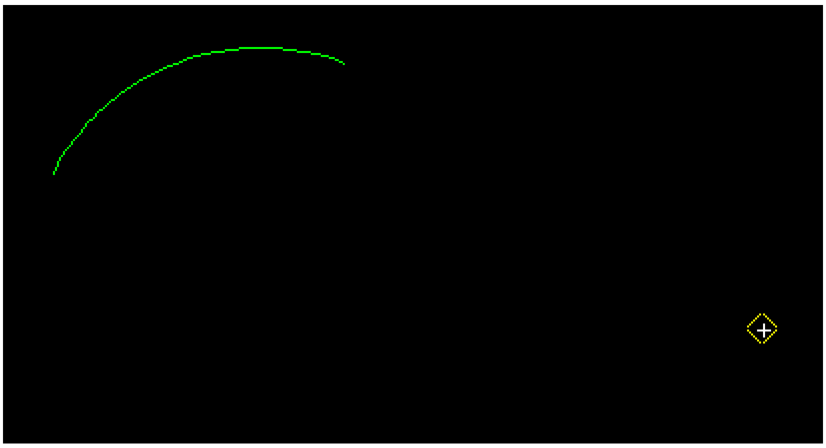
This option creates the tangential line between two elements.

NOTE: This option is for super strings only. When Selecting an arc or circle, the selection must be a line snap with direction. If a point is selected on the arc/circle the line will be draw between the selected points and not the tangent.

On selecting **Tangent**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

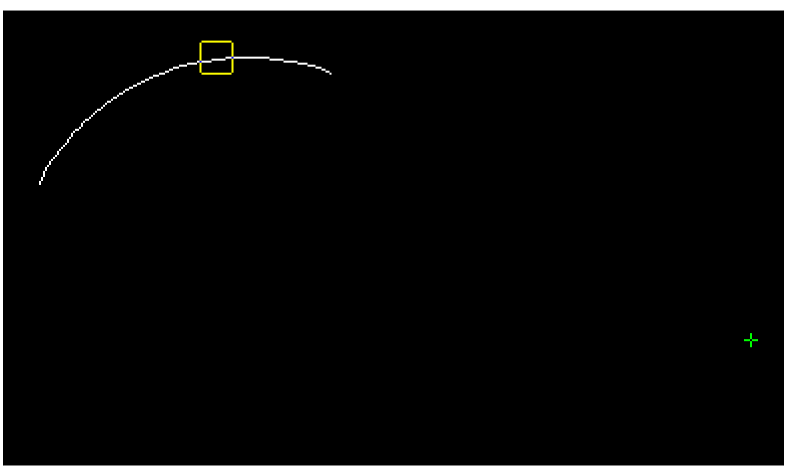
STEP 1:

A start position of the reference line is selected and accepted.



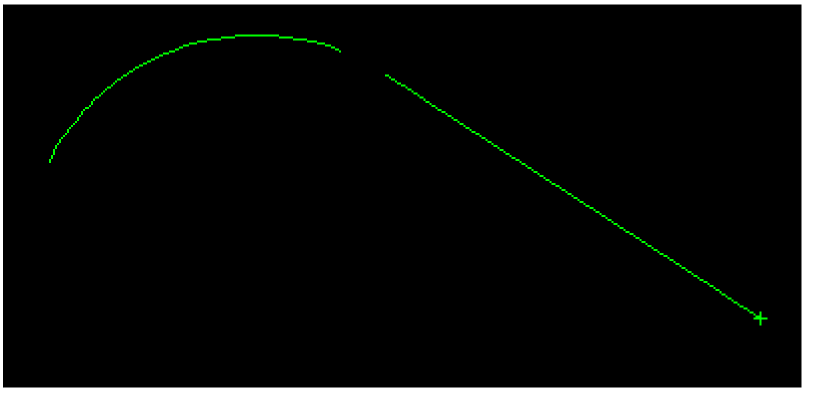
STEP 2:

The user selects and accepts the arc/circle segment with direction. The direction is required because there are two possible solutions. In this case, the direction was anti clockwise.



STEP 3:

After accepting the segment, a line is draw from the 1st selected point to the tangent point. Note that in this example, the arc is produced around so that a solution can be found.



15.11.3 Line string

Position of option on menu: **CAD =>Line =>Line string**

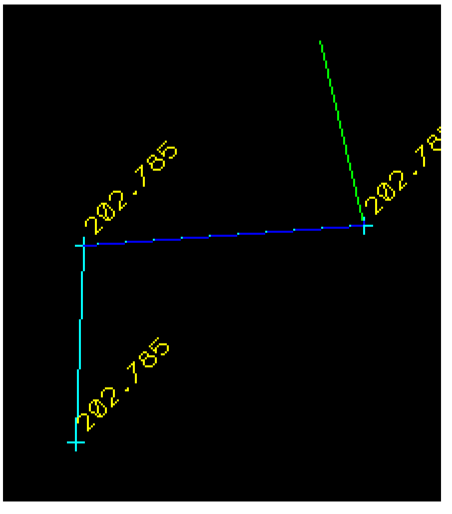
or by selection of appropriate icon from the toolbar. 

This option creates a string of many points.

On selecting **Line string**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

The position of each vertex of the string is selected and accepted in order from the first to the last vertex.



15.11.4 Perpendicular

This option creates a string by selecting a reference string and a reference point the create a line from the reference point perpendicular to the reference string.

On selecting **Perpendicular**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

A reference string is selected and accepted.

STEP 2:

A line is drawn from the current mouse position perpendicular to the reference string

STEP 3:

After the final position is accepted, a line string is created which start from the selected position and perpendicular to the reference string.

15.11.5 Traverse

Position of option on menu: **CAD =>Line=>Traverse**

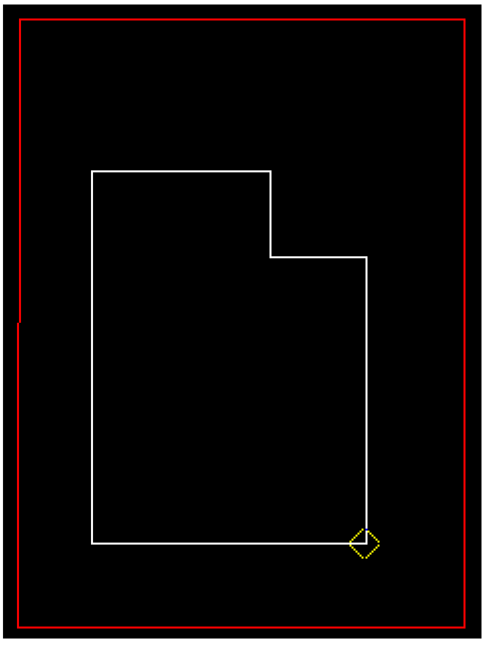
or by selection of appropriate icon from the toolbar.

This option creates a string by giving the bearing and distance of each vertex from the previous vertex. The position of the first vertex is the start of the string and then the bearing and distance is given to define the position of the next vertex. The created vertex then become the position to take the next bearing and distance from.

On selecting **Traverse**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

A start position is selected and accepted. This becomes the first vertex of the string.

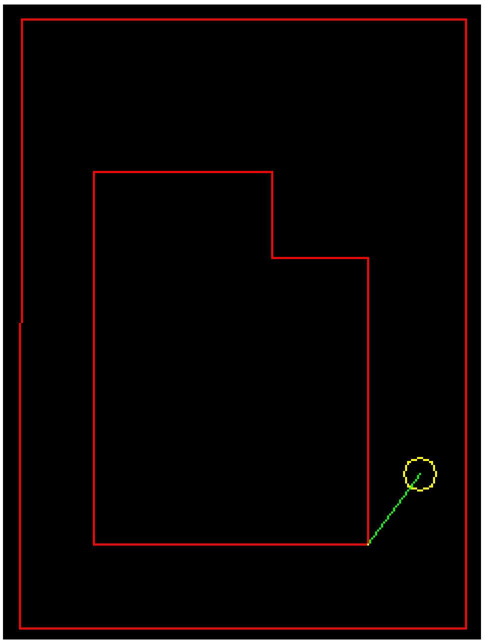


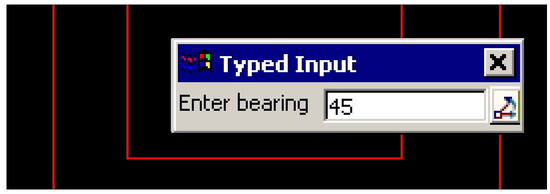
STEP 2:

A line is drawn from the vertex to the current mouse position. If the user wants to see what the current value of the bearing is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the point or the input box can be closed and the rubber banding (graphically changing) of the bearing continued.

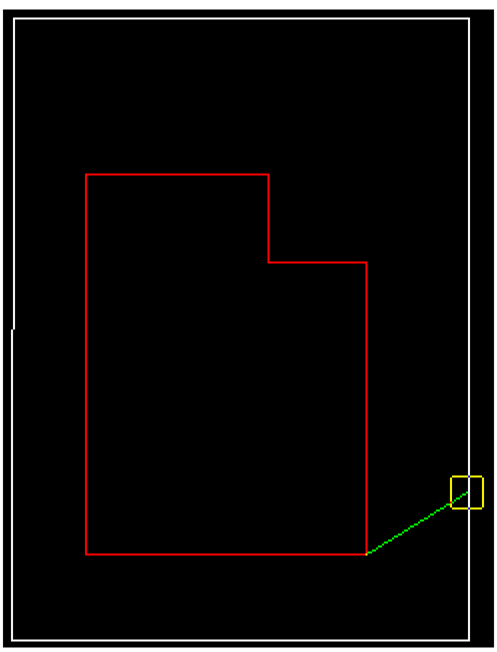
For typed entry, simply start typing or press the space bar to bring up the bearing entry box. The value is entered into the input box followed by the enter key.

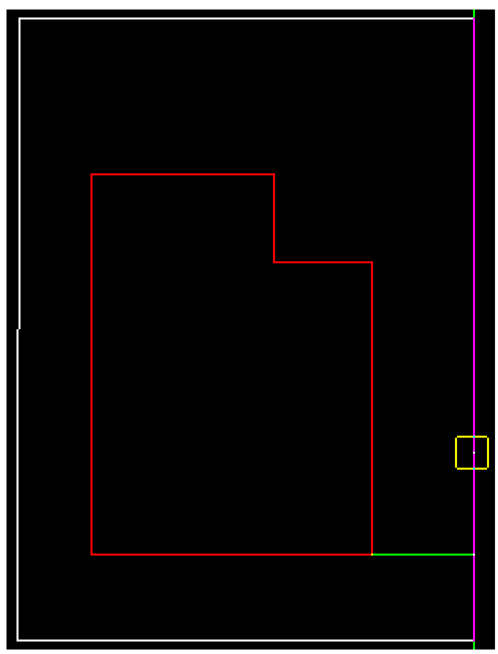
Note: **The Page up and page down keys can be used when the input angle box comes up to add or subtract intervals of 90 degrees.**





This option also allows the definition of the bearing by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next. The example shown below is the perpendicular case.

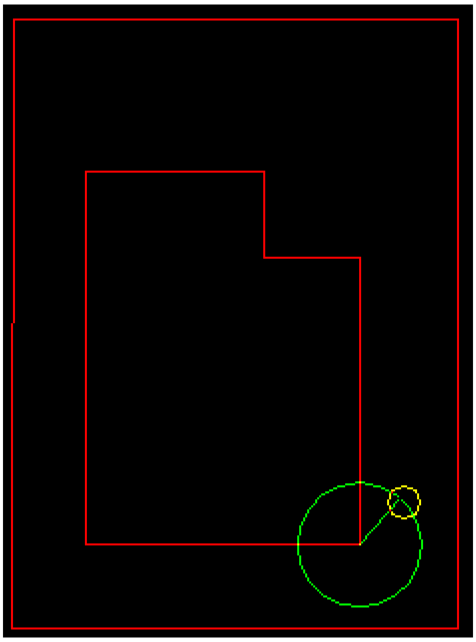


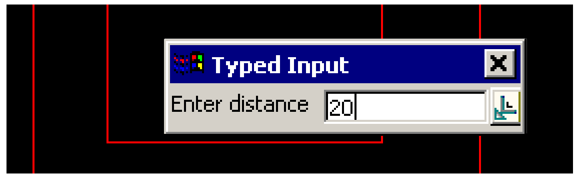


STEP 3:

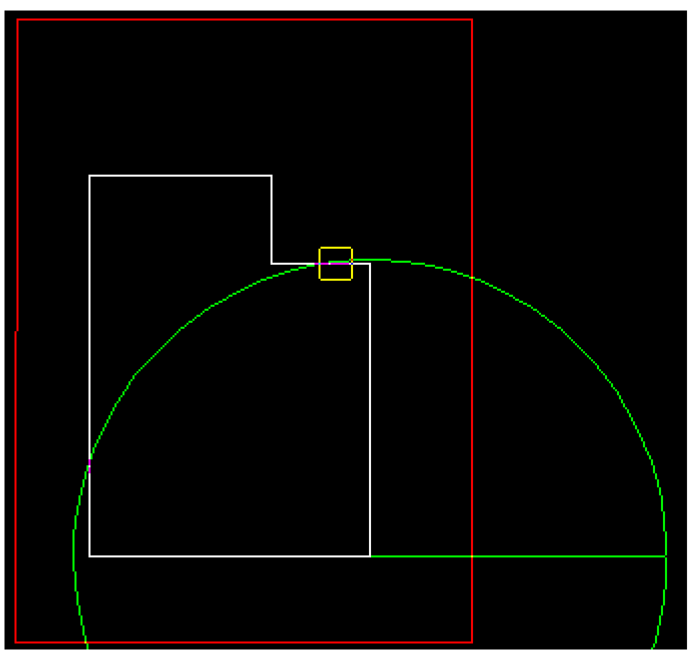
A distance from the start point to create the point is given either by selection with the mouse or by typing a value. To type a value, simply start typing and the input box for the distance will appear. Alternatively you can press the space bar to bring up the input box. The value is entered into the input box followed by the enter key.

The circle drawn represents the distance value and changes with movement of the mouse. If the user wants to see what the current value of the distance is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the point or the input box can be closed and the rubber banding (graphically changing) of the circle continued.

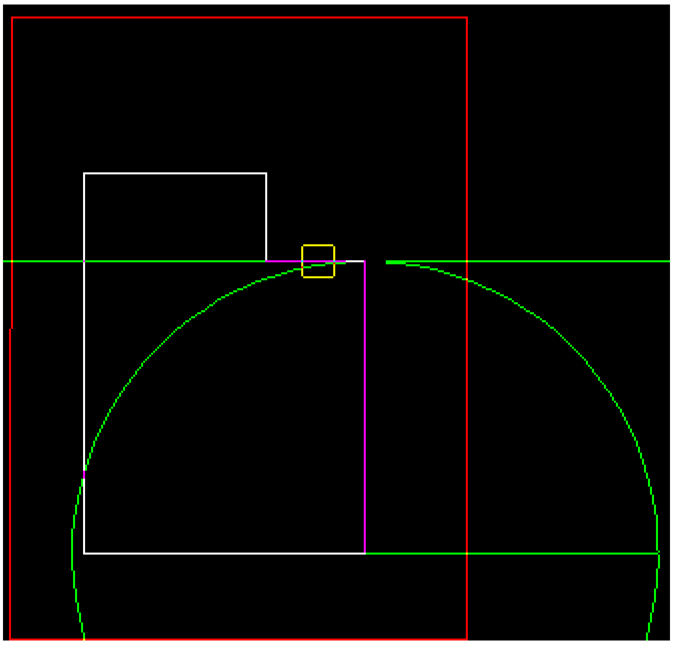




This option also allows the definition of the distance by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next. The example shown below is the perpendicular case.

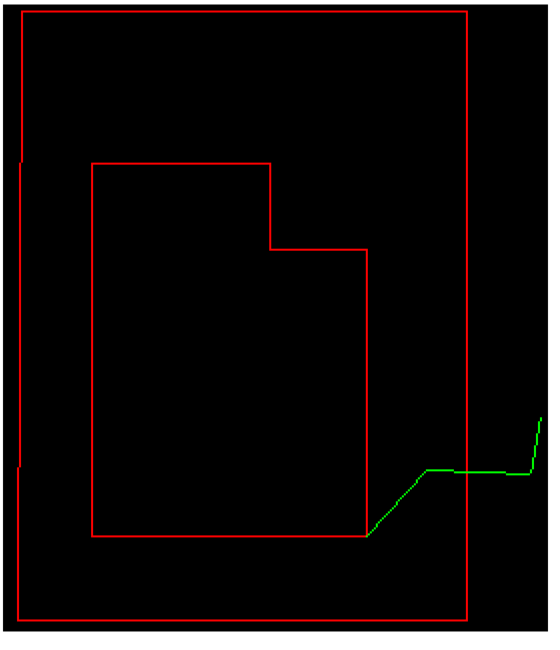


The perpendicular distance shown below by the purple line will be use at the defined bearing to create the point.



STEPS 4-5:

The next bearing and distance is prompted for as per step 4 and 5. The start point is now the last created point. A number of segments can be entered in this way.



15.11.6 Traverse append

Position of option on menu: **CAD =>Line =>Traverse append**

or by selection of appropriate icon from the toolbar.

This option adds vertices to the ***end*** of a super string by giving the bearing and distance of each vertex from the previous vertex. When a string is selected, the end of the string becomes the position that the bearing distance is taken from. The created vertex then become the position to take the next bearing and distance from.

On selecting **Traverse append**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

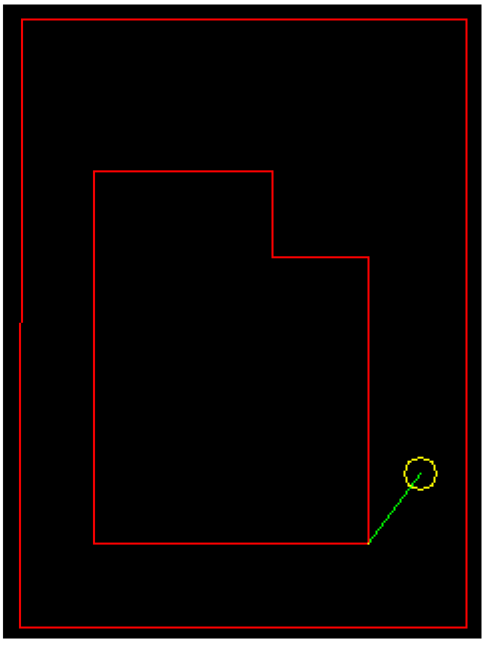
A string to append vertices to is selected and accepted. The end of this string becomes the position to define the first bearing distance from.

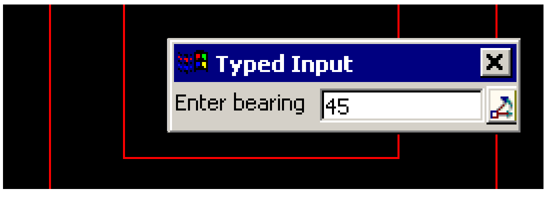
STEP 2:

A line is drawn from the last vertex to the current mouse position. If the user wants to see what the current value of the bearing is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the point or the input box can be closed and the rubber banding (graphically changing) of the bearing continued.

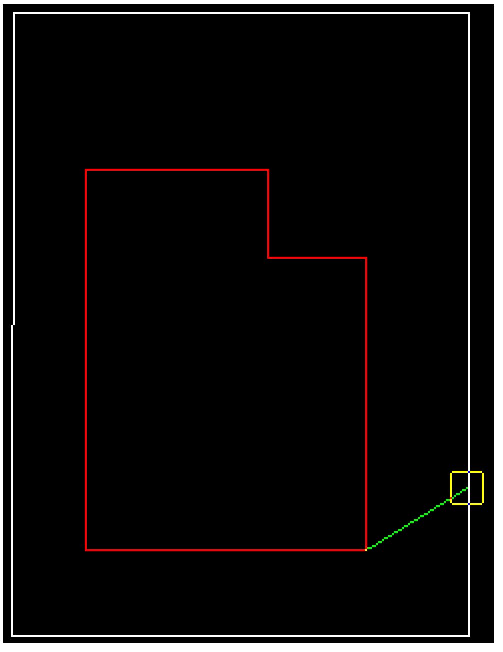
For typed entry, simply start typing or press the space bar to bring up the bearing entry box. The value is entered into the input box followed by the enter key.

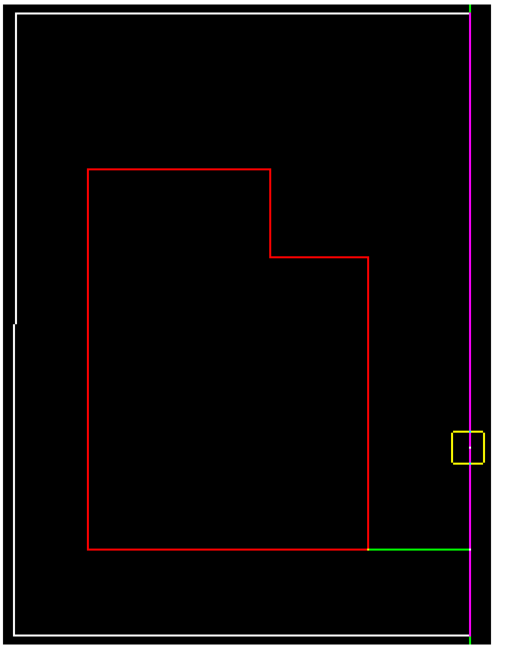
Note: **The Page up and page down keys can be used when the input angle box comes up to add or subtract intervals of 90 degrees.**





This option also allows the definition of the bearing by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next. The example shown below is the perpendicular case.

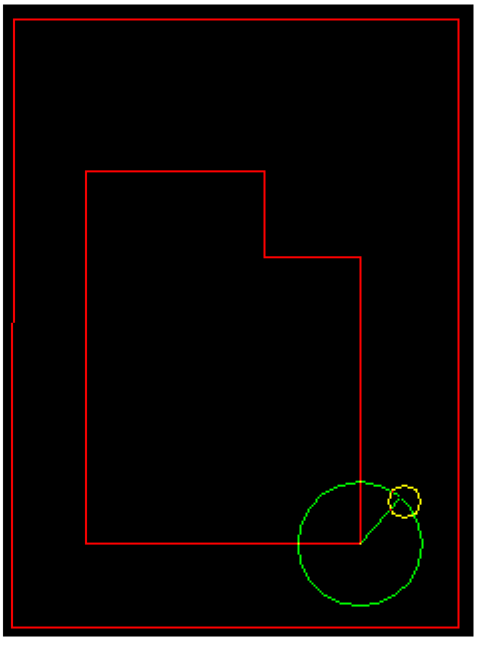


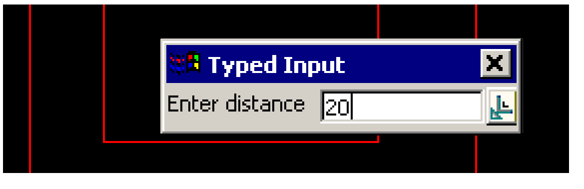


STEP 3:

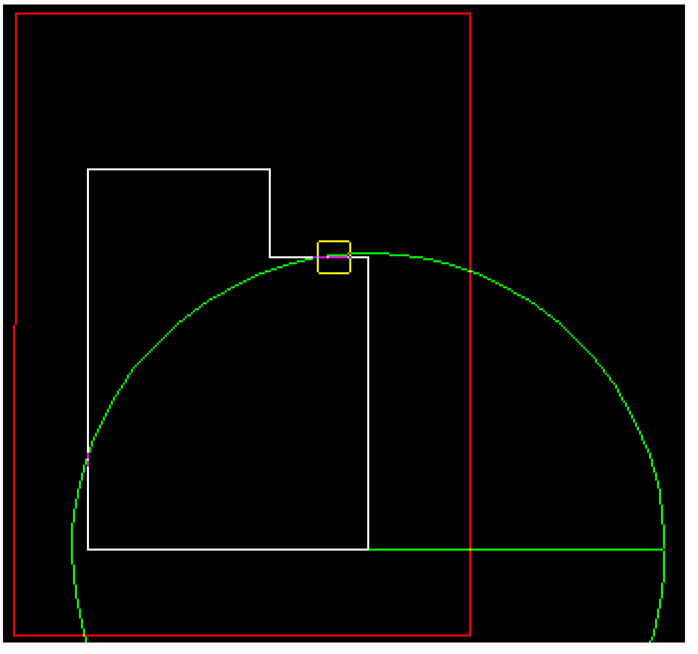
A distance from the start point to create the point is given either by selection with the mouse or by typing a value. To type a value, simply start typing and the input box for the distance will appear. Alternatively you can press the space bar to bring up the input box. The value is entered into the input box followed by the enter key.

The circle drawn represents the distance value and changes with movement of the mouse. If the user wants to see what the current value of the distance is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the point or the input box can be closed and the rubber banding (graphically changing) of the circle continued.

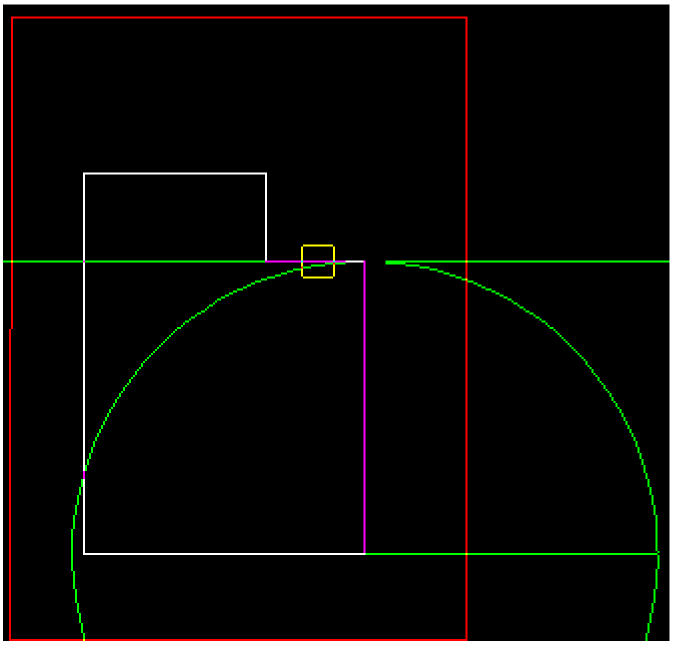




This option also allows the definition of the distance by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next. The example shown below is the perpendicular case.

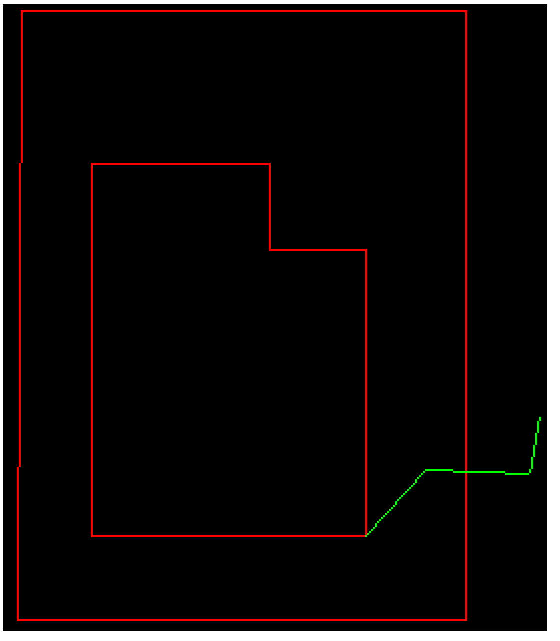


The perpendicular distance shown below by the purple line will be use at the defined bearing to create the point.



STEPS 4-5:

The next bearing and distance is prompted for as per step 4 and 5. The start point is now the last created point. A number of segments can be entered in this way.



15.11.7 Traverse edit

Position of option on menu: **CAD =>Line =>Traverse edit**

or by selection of appropriate icon from the toolbar.

This option changes the bearing and length of a selected line segment of a super string.

On selecting **Traverse edit**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

A line segment of a string is selected and accepted. This segment is the one to have its bearing distance redefined.

STEP 2:

A **Bearing** and then the **Distance** typed input boxes are presented with the current values for the line segment displayed. After new values are entered (or just <enter> if the original value is to be maintained) then the new bearing and distance values are applied to the start vertex of the segment to produce a new end vertex. The rest of the string keeps the same bearing/distances are in the original string.

Note: **The Page up and page down keys can be used when the input bearing box comes up to add or subtract intervals of 90 degrees.**

15.11.8 Tape baseline

Position of option on menu: **CAD =>Line =>Tape baseline**

or by selection of appropriate icon from the toolbar.

This option creates a line string by entering distances between points and the offset distance to a user defined reference line string.

On selecting **Tape baseline**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

A start position of the reference line is selected and accepted.

Note: Originating at the start point, the reference line will extend to infinity after intercepting the 2nd point. The offset values are made relative to this reference line. The reference line will be removed once the option has been terminated.

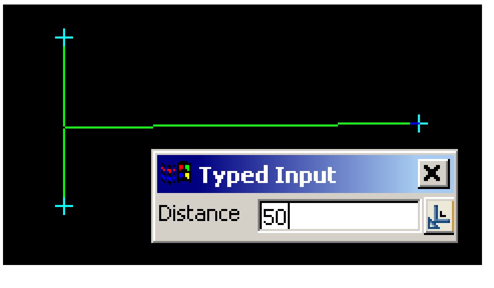
STEP 2:

The interception point for the reference line is selected and accepted.

STEP 3:

After the start and interception points are accepted, the reference line is constructed and the Distance Input box appears. A distance value is to be entered into the Input box, followed by the enter key. For the first point, the distance entered is relative to the start point of the reference line. A positive value means a distance towards the intercept point (from the start point).

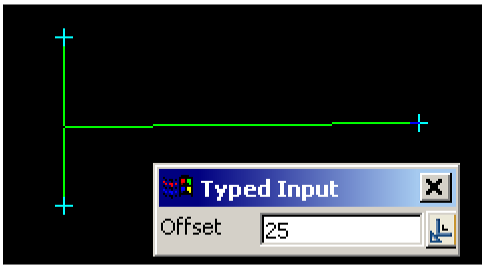
Note: The distance is always relative to the previously entered vertex (positive is from start point to intercept point along the reference line).



STEP 4:

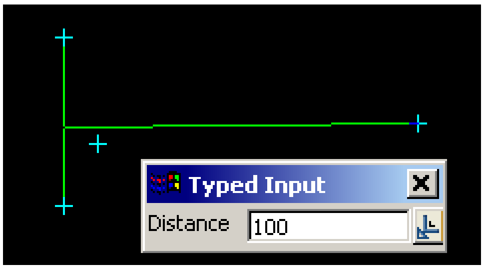
After the distance value has been entered, the Offset Input box will appear. Type the required offset value into the Input box and press the enter key.

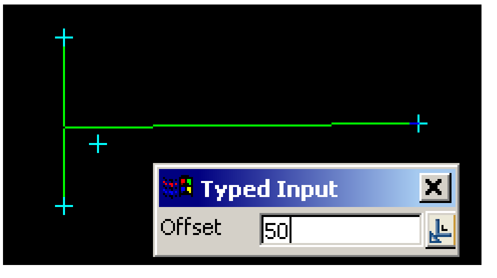
Note: The offset distance is always relative to the reference line (negative is to the left).



STEP 5:

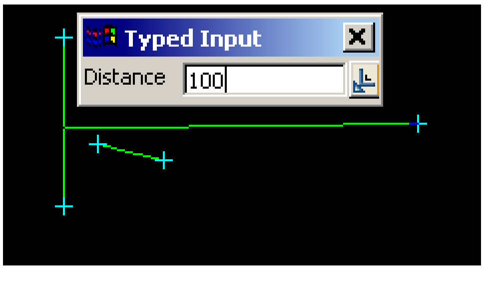
Steps 3 and **4** should be repeated until the required number of vertices have been created in the line string.

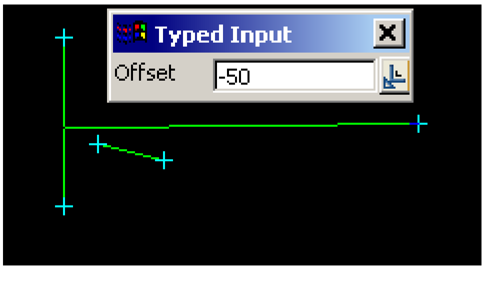


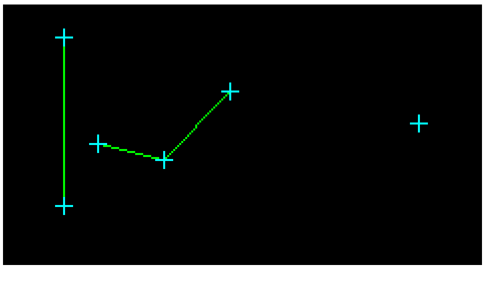


STEP 6:

The option can be terminated by click on the close button on the Input box, followed by the Esc key.







15.11.9 Tape baseline absolute

Position of option on menu: **CAD =>Line =>Tape baseline absolute**

or by selection of appropriate icon from the toolbar.

This option creates a line string by entering distances from the start point and the offset distance to a user defined reference line string.

On selecting **Tape baseline absolute**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

A start position of the reference line is selected and accepted.

Note: Originating at the start point, the reference line will extend to infinity after intercepting the 2nd point. The offset values are made relative to this reference line. The reference line will be removed once the option has been terminated.

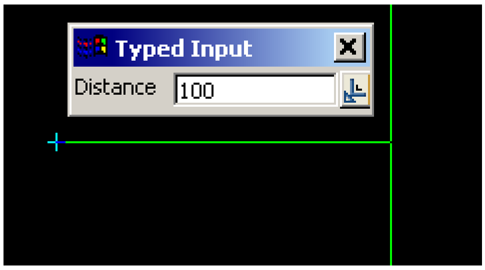
STEP 2:

The interception point for the reference line is selected and accepted.

STEP 3:

After the start and interception points are accepted, the reference line is constructed and the Distance Input box appears. A distance value is to be entered into the Input box, followed by the enter key. For the absolute option, all distances entered are relative to the start point of the reference line. A positive value means a distance towards the intercept point (from the start point).

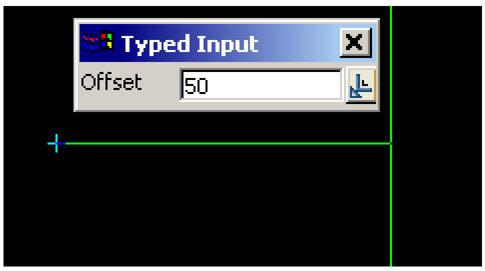
Note: For the absolute option, distances are always relative to the start point of the reference line (positive is from start point to intercept point along the reference line).



STEP 4:

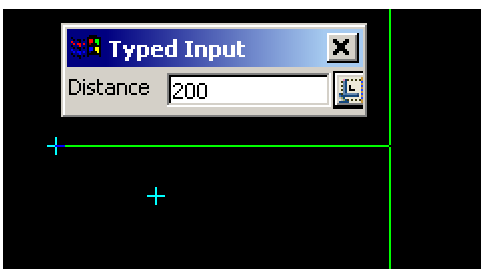
After the distance value has been entered, the Offset Input box will appear. Type the required offset value into the Input box and press the enter key.

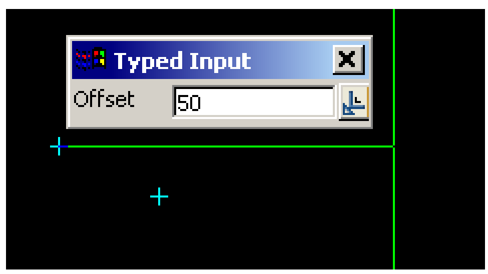
Note: The offset distance is always relative to the reference line (negative is to the left).



STEP 5:

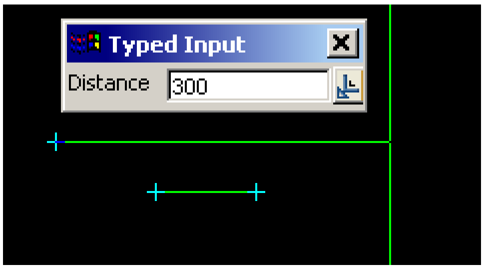
Steps 3 and 4 should be repeated until the required number of vertices have been created in the line string.

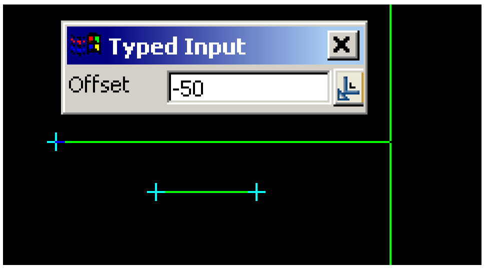


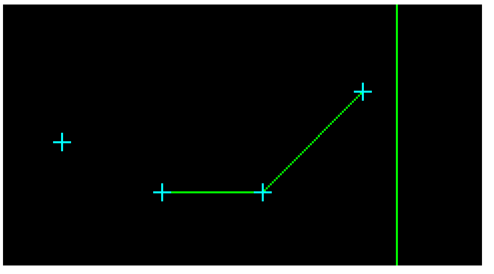


STEP 6:

The option can be terminated by click on the close button  on the Input box, followed by the Esc key.


Typed Input
Distance
300

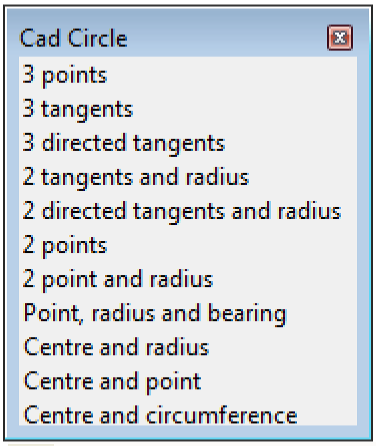

Typed Input
Offset
-50



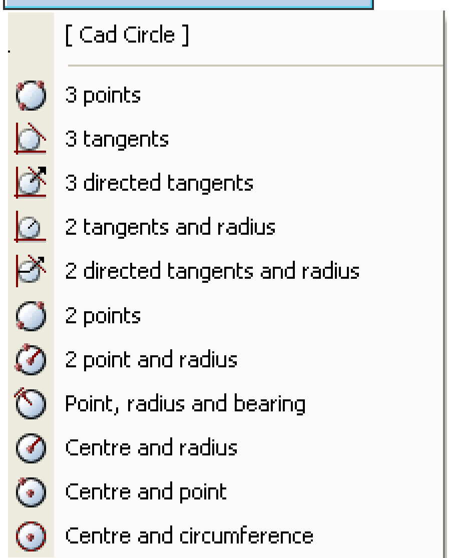
15.12 CAD Circle

Position of option on menu: **CAD =>Circle**

The **Circle** walk-right menu is



from 3 selected points
tangential to 3 tangents (including points)
tangential to 3 directed tangents (including points)
known radius and tangential to 2 tangents
known radius and tangential to 2 directed tangents
from 2 points on diameter
through 2 points and known radius
through point with known radius and a bearing at point
centre position and specified radius
centre position and point on circle



For the option *3 points*, go to 15.12.1 3 Points
3 tangents 15.12.2 3 tangents
3 directed tangents 15.12.3 3 directed tangents
2 tangents and radius 15.12.4 2 tangents and radius
2 directed tangents and radius. 15.12.5 2 directed tangents and radius
2 points 15.12.6 2 points
2 points and radius 15.12.7 2 point and radius
Point, radius and bearing 15.12.8 Point, radius and bearing
Centre and radius 15.12.9 Centre and radius
Centre and point 15.12.10 Centre and point
Centre and circumference 15.12.11 Centre and circumference

15.12.1 3 Points

Position of option on menu: CAD =>Circle =>3 points

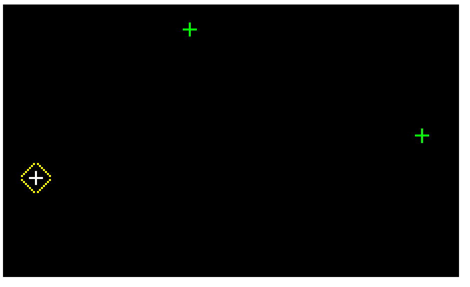
or by selection of appropriate icon from the toolbar.

This option creates the circle through three selected points.

On selecting **3 points**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

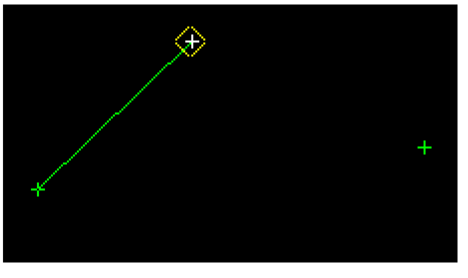
The 1st point is selected with the mouse or entered in via the keyboard. To specify the 1st point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter the 1st point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.



STEP 2:

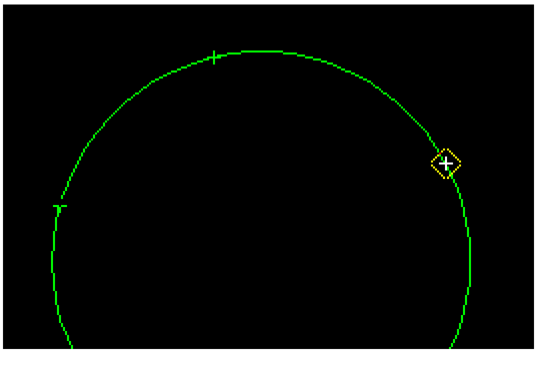
The 2nd point is selected and accepted.

After the 2nd point is accepted a circle will be displayed ‘rubber banding’ to the various solutions according to the position of the cursor. This will continue until the 3rd point is selected and accepted.



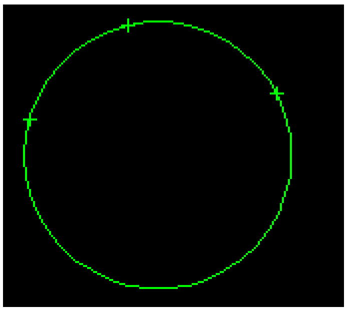
STEP 3:

The 3rd point is selected and accepted.



STEP 4:

A circle is constructed through the three selected points



15.12.2 3 tangents

Position of option on menu: **CAD =>Circle=>3 tangents**

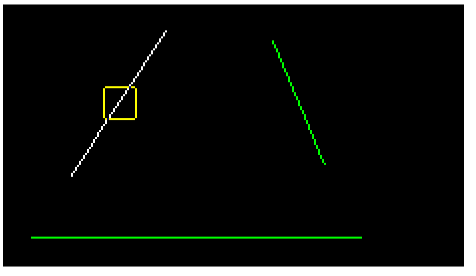
or by selection of appropriate icon from the toolbar. 

This option creates a circle that is tangential to three selected segments.

On selecting **3 tangents**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

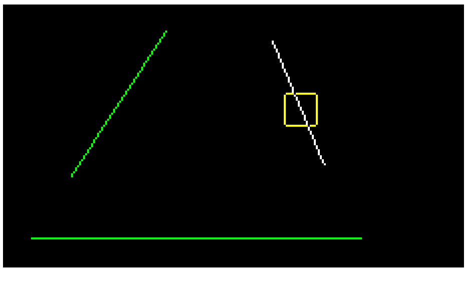
STEP 1:

The 1st tangent is selected with the mouse (Left Button) and accepted (Middle Button).



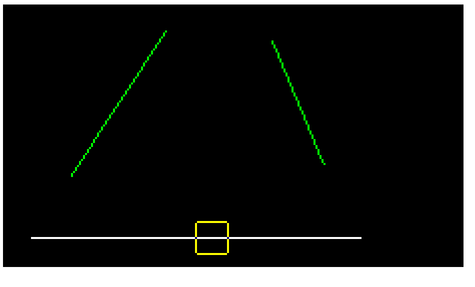
STEP 2:

The 2nd tangent is selected and accepted.



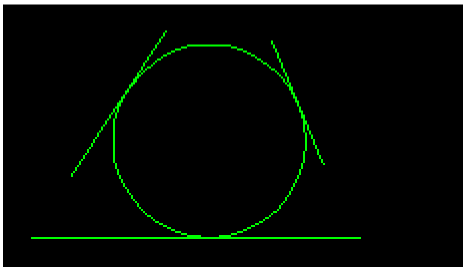
STEP 3:

The 3rd tangent is selected and accepted.



STEP 4:

A circle is constructed that touches each of the three selected tangents.



15.12.3 3 directed tangents

Position of option on menu: **CAD =>Circle =>3 directed tangents**

or by selection of appropriate icon from the toolbar. 

This option creates a circle that is tangential to three selected segments. The segments are selected in order and with direction and the circle is to the right of the direction of the selected segments.

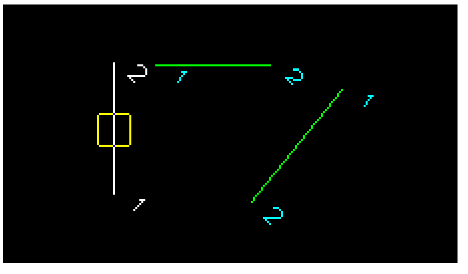
On selecting **3 directed tangents** the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

Select and accept the 1st tangent.

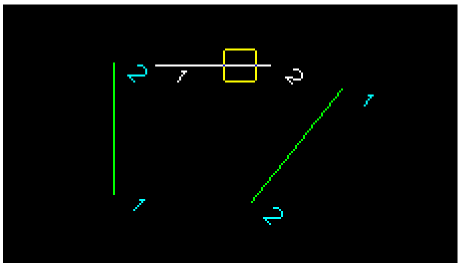
Note: For this option the direction of the selected tangents is important. The circle will be constructed to the right of a tangent. A user may reverse the direction of a tangent by selecting the tangent *with direction*. For further notes on picking tangents with direction, see Picking with Direction.

Note: The Vertex indices can be displayed by toggling the option on the Toggle Menu.



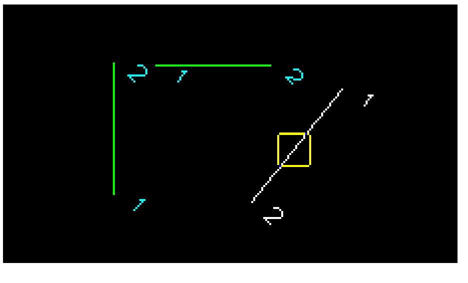
STEP 2:

The 2nd tangent is selected and accepted.



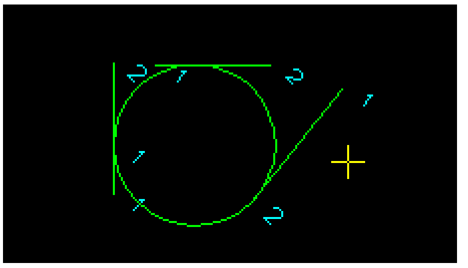
STEP 3:

The 3rd tangent is selected and accepted.



STEP 4:

If a solution exists, a circle is constructed using the given information.



15.12.4 2 tangents and radius

Position of option on menu: **CAD =>Circles=>2 tangents and radius**

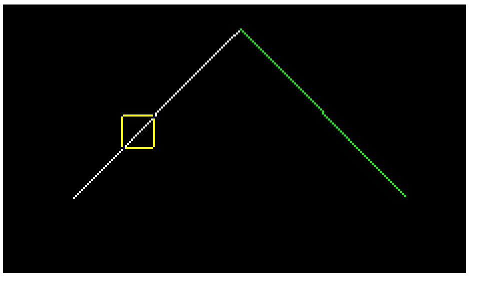
or by selection of appropriate icon from the toolbar.

This option creates a circle with a given radius that is tangential to two selecting segments.

On selecting **2 tangents and radius**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

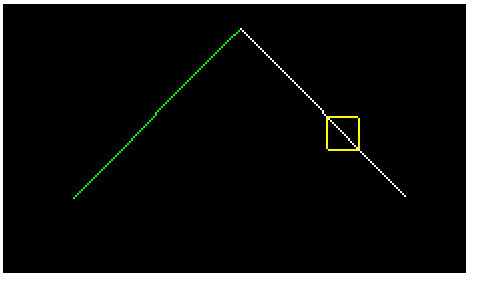
STEP 1:

Select the 1st tangent with the mouse (Left Button) and accept it (Middle Button).



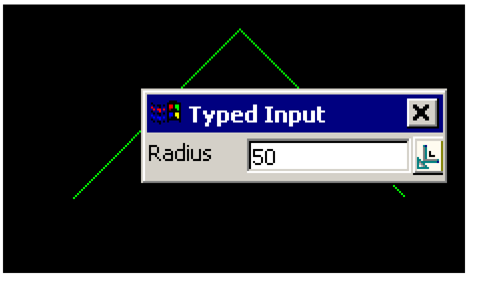
STEP 2:

Select the 2nd tangent and accept it.



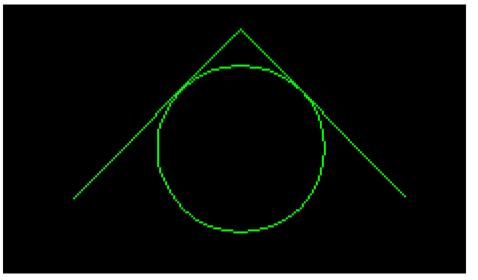
STEP 3:

After the 2nd tangent is accepted, the Radius Input box will appear. Type the radius value into the Input box and press the enter key. The browse button on the Input box can be used to define the radius by measuring existing elements.



STEP 4:

If a solution exists, a circle is fitted touching the two selected tangents using the given radius.



15.12.5 2 directed tangents and radius

Position of option on menu: CAD =>Circle=>2 directed tangents and radius

or by selection of appropriate icon from the toolbar.

This option creates a circle with a given radius that is tangential to two selected segments that are picked with direction.

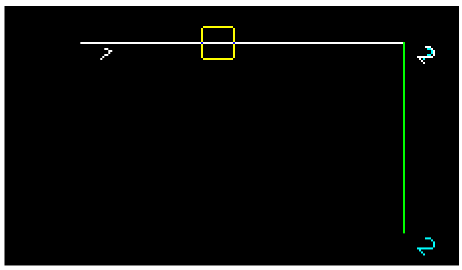
On selecting **2 directed tangents and radius**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

Select and accept the 1st tangent.

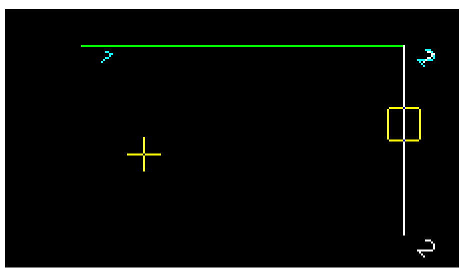
Note: For this option the direction of the selected tangents is important. The circle will be constructed to the right of a tangent. A user may reverse the direction of a tangent by selecting the tangent *with direction*. For further notes on picking tangents with direction, see Picking with Direction.

Note: The Vertex indices can be displayed by toggling the option on the Toggle Menu.



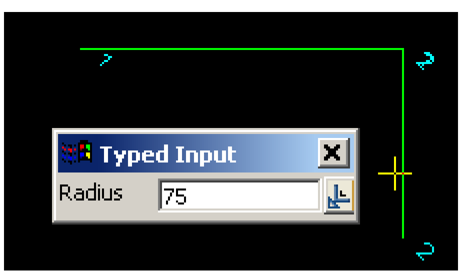
STEP 2:

The 2nd tangent is selected and accepted.



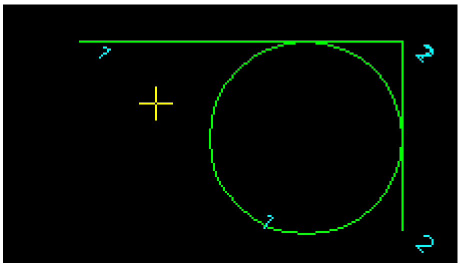
STEP 3:

After the 2nd tangent is accepted, the Radius Input box will appear. Type the radius value into the Input box and press the Enter key. The browse button on the Input box can be used to define the arc radius by measuring existing elements.



STEP 4:

If a solution exists, a circle is constructed using the given information.



15.12.6 2 points

Position of option on menu: **CAD =>Circle =>2 points**

or by selection of appropriate icon from the toolbar.

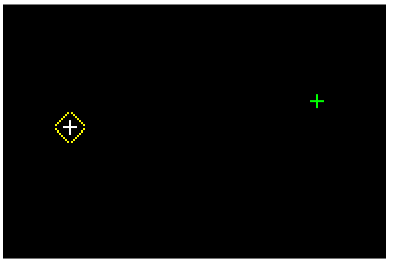
This option creates a circle by selecting two points that define the diameter of the circle.

On selecting **2 points**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

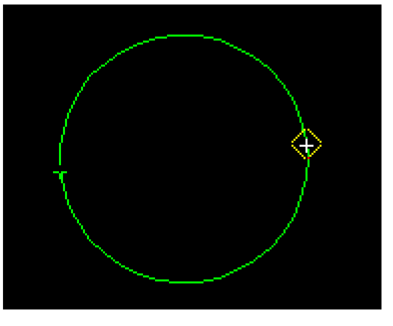
The 1st point is selected with the mouse or entered in via the keyboard. To specify the 1st point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter the 1st point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.

After the 1st point is accepted a circle will be displayed 'rubber banding' to the various solutions according to the position of the cursor (cursor position taken as the other end of a diameter). This will continue until the 2nd point is selected and accepted.



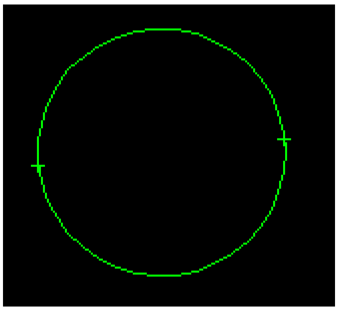
STEP 2:

The 2nd point is selected and accepted.



STEP 3:

The circle is constructed through the two selected points. The two points define the diameter.



15.12.7 2 point and radius

Position of option on menu: **CAD =>Circle =>2 point and radius**

or by selection of appropriate icon from the toolbar. 

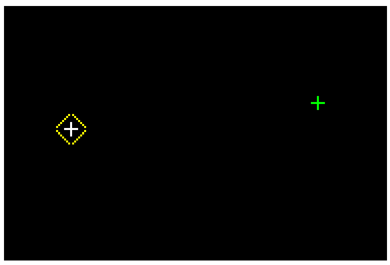
This option creates a circle of a given radius that goes through two selected points.

On selecting **2 point and radius**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

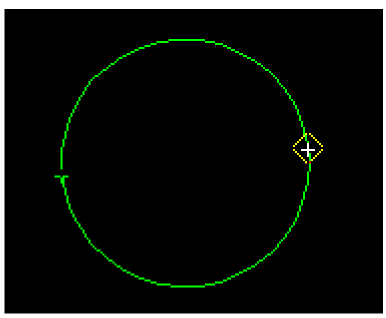
The 1st point is selected with the mouse or entered in via the keyboard. To specify the 1st point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter the 1st point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.

After the 1st point is accepted a circle will be displayed 'rubber banding' to the various solutions according to the position of the cursor. This will continue until the 2nd point is selected and accepted.



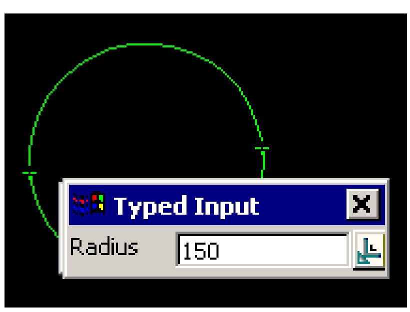
STEP 2:

The 2nd point is selected and accepted.



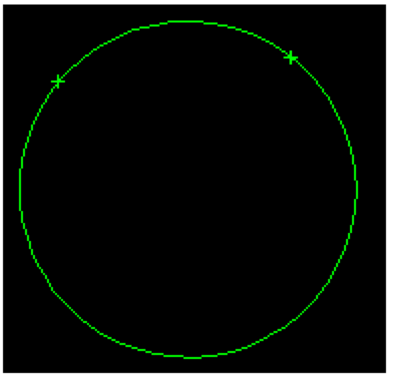
STEP 3:

After the 2nd point is accepted, the Radius Input box will appear. The radius value is entered into the input box followed by the enter key.



STEP 4:

If a solution exists, the circle is fitted through the two selected points using the given radius.



15.12.8 Point, radius and bearing

Position of option on menu: **CAD =>Circle=>Point, radius and bearing**

or by selection of appropriate icon from the toolbar.

This option creates a circle of a given radius, a selected point on the circle and the bearing of the tangent to the circle at that point.

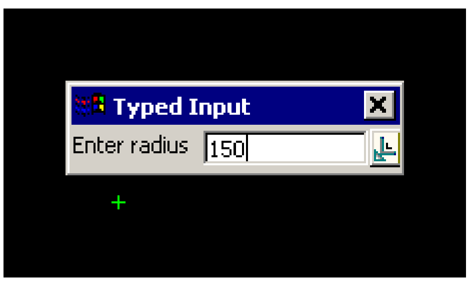
On selecting **Point, radius and bearing**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

The 1st point is selected with the mouse or entered in via the keyboard. To specify the 1st point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter a 1st point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the Enter key.

STEP 2:

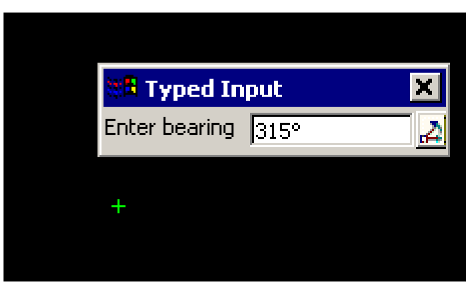
After the 1st point is accepted, the **Enter radius** input box will appear. Type the radius value into the Input box and press the enter key. The browse button on the Input box can be used to define the arc radius by measuring existing elements.



STEP 3:

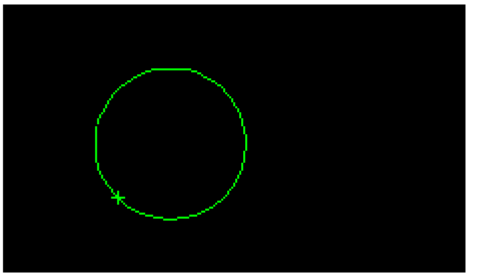
After the radius has been entered, the **Enter bearing** input box will appear. Type the bearing into the Input box and press the enter key. The browse button on the Input box can be used to define the bearing by measuring existing elements.

Note: **The Page Up and Page Down keys can be used when the Enter Bearing input box comes up to add or subtract intervals of 90 degrees.**



STEP 4:

If a solution exists, a circle is fitted from the given bearing at the selected point using the selected radius.



15.12.9 Centre and radius

Position of option on menu: **CAD =>Circle =>Centre and radius**

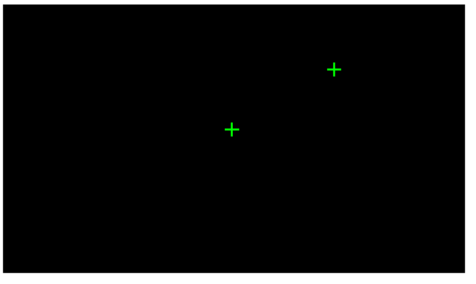
or by selection of appropriate icon from the toolbar. 

This option creates a circle of a given radius and a selected centre point

On selecting **Centre and radius**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

A centre point is selected with the mouse or entered in via the keyboard. To specify a centre point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter a centre point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.



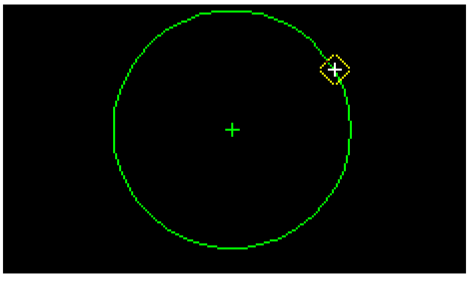
STEP 2:

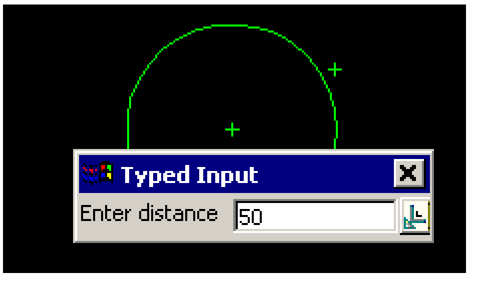
After the centre point is accepted a circle will be displayed ‘rubber banding’ to the various solutions according to the position of the cursor. This will continue until a radius value is entered.

A radius value is selected with the mouse or entered in via the keyboard.

To specify a radius value with the mouse, a point must be selected (Left Button) and accepted (Middle Button). This option also allows the definition of the radius by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next.

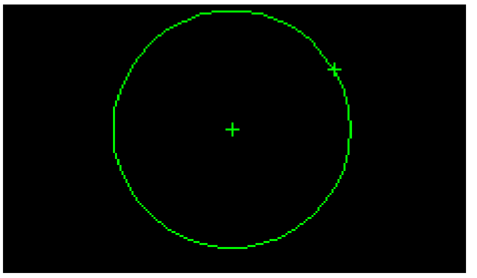
To enter a radius value with the keyboard, simply start typing or press the space bar to bring up the Radius Input box. Type the radius value into the Radius Input box and press the enter key. The browse button  on the Input box can be used to define the arc radius by measuring existing elements.





STEP 3:

The circle is constructed through the centre point using the specified radius.



15.12.10 Centre and point

Position of option on menu: **CAD =>Circle =>Centre and point**

or by selection of appropriate icon from the toolbar. 

This option creates a circle by selecting a centre point and then using the cursor to select a second point that in on the circle.

On selecting **Centre and point**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

A centre point is selected with the mouse or entered in via the keyboard. To specify a centre point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter a centre point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.

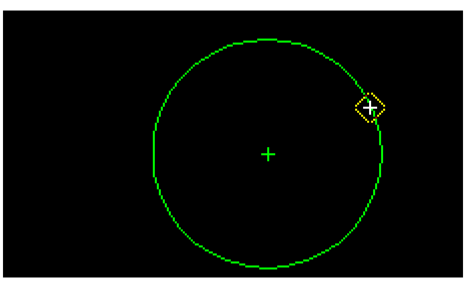
STEP 2:

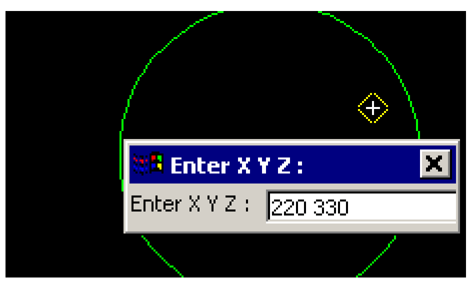
After the centre point is accepted a circle will be displayed ‘rubber banding’ to the various solutions according to the position of the cursor. This will continue until a radius value is entered.

A radius value is selected with the mouse or entered in via the keyboard.

To specify a radius value with the mouse, a point must be selected (Left Button) and accepted (Middle Button). This option also allows the definition of the radius by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next.

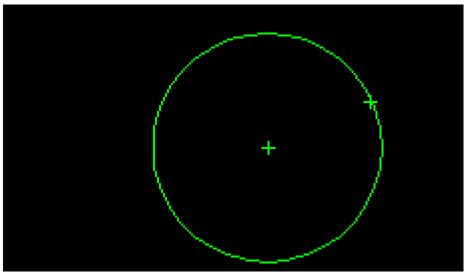
To enter a radius value with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the point into the Input box and press the enter key.





STEP 3:

A circle is constructed using the two points defined by the user.



15.12.11 Centre and circumference

Position of option on menu: **CAD =>Circle=>Centre and circumference**

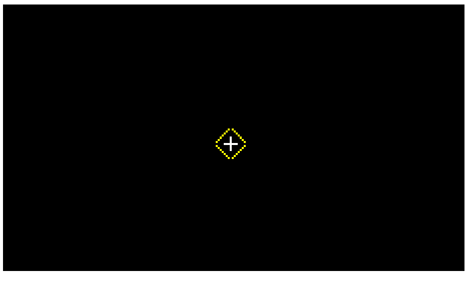
or by selection of appropriate icon from the toolbar.

This option creates a circle by selecting the centre point and giving a circumference value.

On selecting **Centre and circumference**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

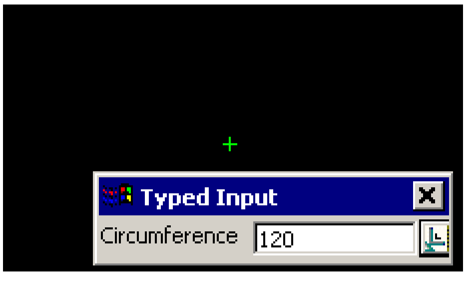
STEP 1:

A centre point is selected with the mouse or entered in via the keyboard. To specify a centre point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter a centre point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.



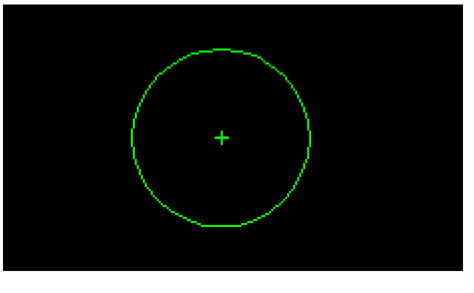
STEP 2:

After the centre point is accepted, the **Circumference** Input box will appear. Type the circumference length into the Input box and press the enter key. The browse button on the Input box can be used to define the arc radius by measuring existing elements.



STEP 3:

A circle is constructed using the centre point and the given circumference.

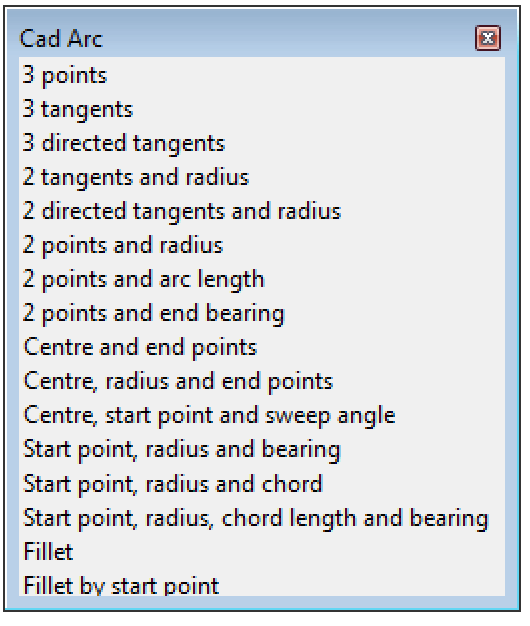


15.13 CAD Arc

Position of option on menu: **CAD =>Arc**

The arc options creates ***super string arcs***, not arc strings. The options **Strings =>Create =>Arc** create arc strings.

The **Arc** walk-right menu is



from 3 selected points
tangential to 3 tangents (including points)
tangential to 3 directed tangents (including points)
known radius and tangential to 2 tangents
known radius and tangential to 2 directed tangents
through 2 points and known radius
2 points and known arc length
2 points and known bearing at first point
centre and end points
centre, radius and end points
centre, start point and sweep angle
through point with known radius and a bearing at point
start point, radius and chord length
start point, radius and chord length and bearing
fillet between 2 segments and given radius
fillet between 2 segments with selected start point

For the option *3 points*, go to 15.13.1 3 points

3 tangents 15.13.2 3 tangents

3 directed tangents 15.13.3 3 directed tangents

2 tangents and radius 15.13.4 2 tangents and radius

2 directed tangents and radius 15.13.5 2 directed tangents and radius

2 points and radius 15.13.6 2 points and radius

2 points and arc length 15.13.7 2 points and arc length

2 points and end bearing 15.13.8 2 points and end bearing

Centre and end points 15.13.9 Centre and end points

Centre, radius and end points 15.13.10 Centre, radius and end points

Centre, start point and sweep angle 15.13.11 Centre, start point and sweep angle

Start point, radius, and bearing 15.13.12 Start point, radius and bearing

Start point, radius and chord 15.13.13 Start point, radius and chord

Start point, radius, chord length and bearing 15.13.14 Start point, radius, chord length and bearing

Fillet 15.13.15 Fillet

Fillet by start point 15.13.16 Fillet by start point

15.13.1 3 points

Position of option on menu: **CAD =>Arc =>3 points**

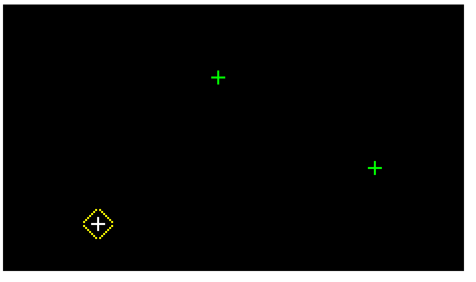
or by selection of appropriate icon from the toolbar. 

This option creates the arc through three selected points and the first and third points are the start and end of the arc.

On selecting **3 points**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

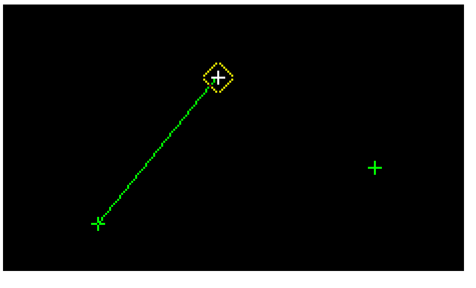
The 1st point is selected with the mouse or entered in via the keyboard. To specify the 1st point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter the 1st point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the Enter key.



STEP 2:

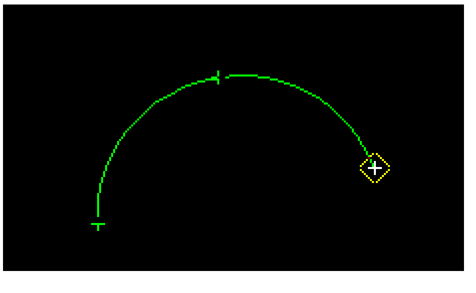
The 2nd point is selected and accepted.

After the 2nd point is accepted an arc will be displayed ‘rubber banding’ to the various solutions according to the position of the cursor. This will continue until the 3rd point is selected and accepted.



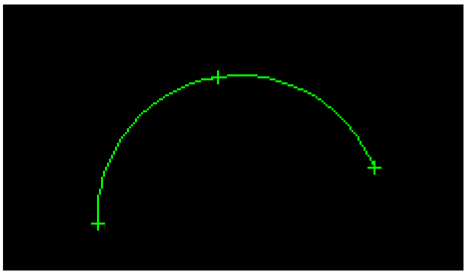
STEP 3:

The 3rd point is selected and accepted.



STEP 4:

An arc is constructed through the three selected points.



15.13.2 3 tangents

Position of option on menu: **CAD =>Arc =>3 tangents**

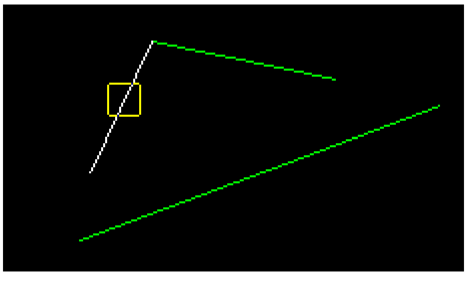
or by selection of appropriate icon from the toolbar. 

This option creates an arc that is tangential to three selected segments. The first and third segments are the start and end of the arc.

On selecting **3 tangents**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

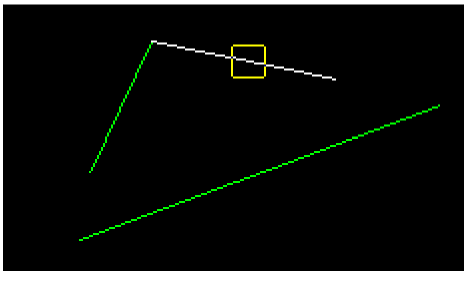
STEP 1:

The 1st tangent is selected with the mouse (Left Button) and accepted (Middle Button).



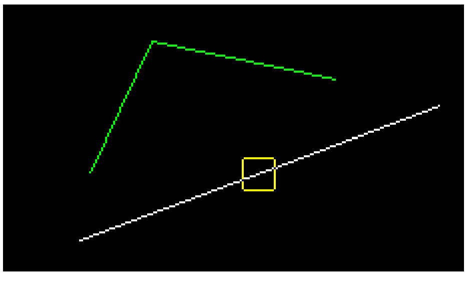
STEP 2:

The 2nd tangent is selected and accepted.



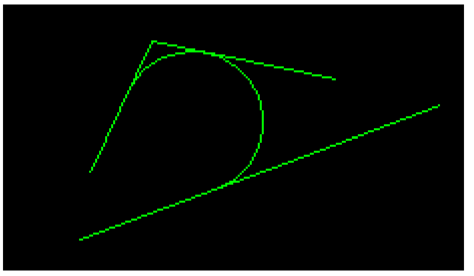
STEP 3:

The 3rd tangent is selected and accepted.



STEP 4:

The arc is constructed touching the three selected tangents.



15.13.3 3 directed tangents

Position of option on menu: **CAD =>Arc =>3 directed tangents**

or by selection of appropriate icon from the toolbar. 

This option creates an arc that is tangential to three selected segments. The segments are selected in order and with direction and the arc is to the right of the direction of the selected segments. The first and third segments are the start and end of the arc.

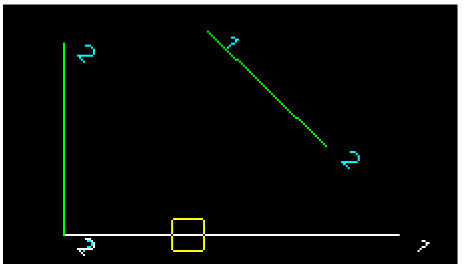
On selecting **3 directed tangents**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

Select and accept the 1st tangent.

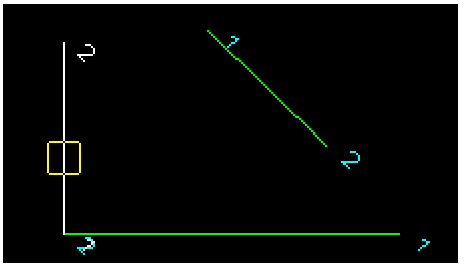
Note: For this option the direction of the selected tangents is important. The arc will be constructed to the right of the tangent. A user may reverse the direction of the tangent by selecting a tangent *with direction*. For further notes on picking tangents with direction, see Picking with Direction.

Note: The Vertex indices can be displayed by toggling the option on the Toggle Menu.



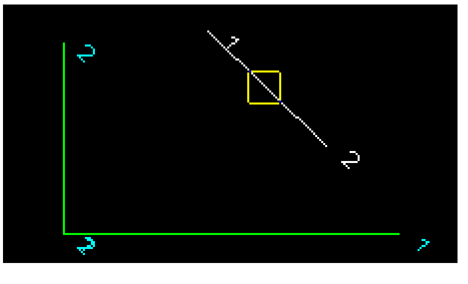
STEP 2:

The 2nd tangent is selected and accepted.



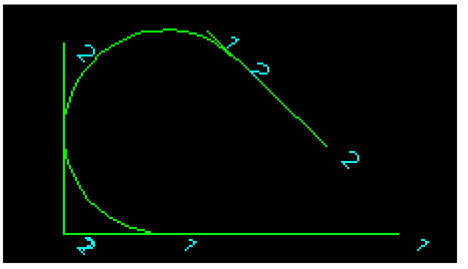
STEP 3:

The 3rd tangent is selected and accepted.



STEP 4:

If a solution exists, an arc is constructed using the given information.



15.13.4 2 tangents and radius

Position of option on menu: **CAD =>Arc => 2 tangents and radius**

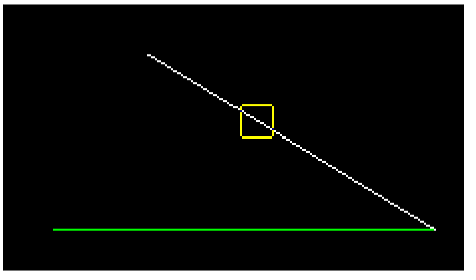
or by selection of appropriate icon from the toolbar.

This option creates an arc with a given radius that is tangential to two selecting segments. This is the same as a fillet.

On selecting **2 tangents and radius**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

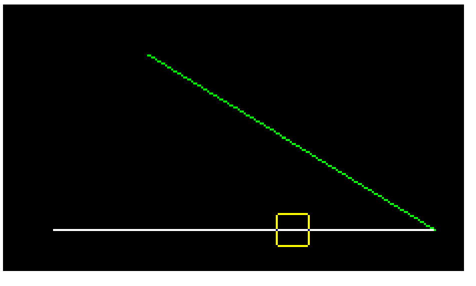
STEP 1:

The 1st tangent is selected with the mouse (Left Button) and accepted (Middle Button).



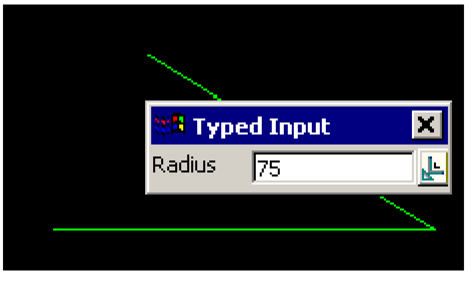
STEP 2:

The 2nd tangent is selected and accepted.



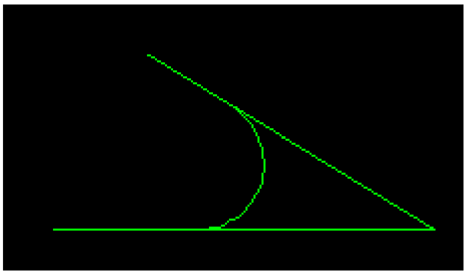
STEP 3:

After the start point is accepted, the Radius Input box will appear. Type the radius value into the Input box and press the enter key. The browse button on the Input box can be used to define the arc radius by measuring existing elements.



STEP 4:

If a solution exists, the arc is fitted through the two selected tangents using the given radius.



15.13.5 2 directed tangents and radius

Position of option on menu: **CAD =>Arc =>2 directed tangents and radius**

or by selection of appropriate icon from the toolbar.

This option creates an arc with a given radius that is tangential to two selected segments that are picked with direction.

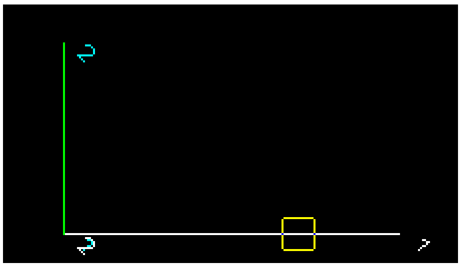
On selecting **2 directed tangents and radius**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

Select and accept the 1st tangent.

Note: For this option the direction of the selected tangents is important. The arc will be constructed to the right of a tangent. A user may reverse the direction of a tangent by selecting the tangent *with direction*. For further notes on picking tangents with direction, see Picking with Direction.

Note: The Vertex indices can be displayed by toggling the option on the Toggle Menu.

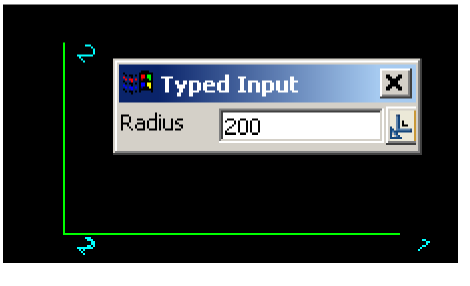


STEP 2:

The 2nd tangent is selected and accepted.

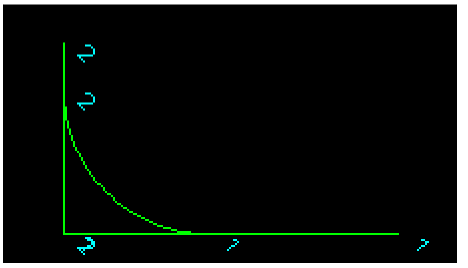
STEP 3:

After the start point is accepted, the Radius Input box will appear. Type the radius value into the Input box and press the enter key. The browse button on the Input box can be used to define the arc radius by measuring existing elements.



STEP 4:

If a solution exists, the arc is fitted through the two selected tangents using the given radius.



15.13.6 2 points and radius

Position of option on menu: **CAD =>Arc =>2 points and radius**

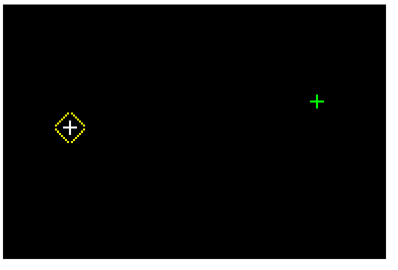
or by selection of appropriate icon from the toolbar.

This option creates an arc of a given radius that starts and ends on two selected points.

On selecting **2 points and radius**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

The 1st point is selected with the mouse or entered in via the keyboard. To specify the 1st point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter the 1st point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.



STEP 2:

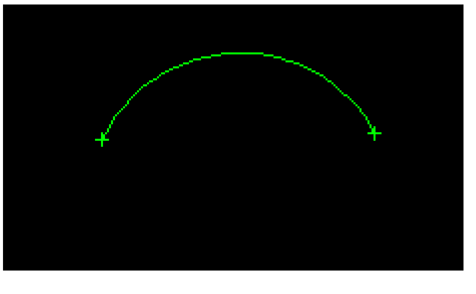
After the start point is accepted, the Radius Input box will appear. Type the radius value into the Input box and press the enter key. The browse button on the Input box can be used to define the arc radius by measuring existing elements.

STEP 3:

The 2nd point is selected and accepted.

STEP 4:

If a solution exists, an arc is constructed using the given information.



15.13.7 2 points and arc length

Position of option on menu: **CAD =>Arc =>2 points and arc length**

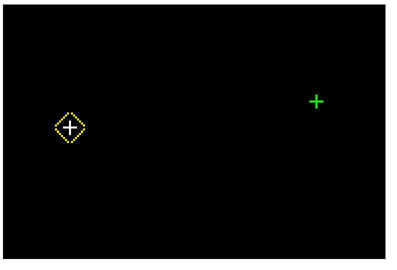
or by selection of appropriate icon from the toolbar 

This option creates an arc of a given arc length that starts and ends on two selected points.

On selecting **2 points and arc length**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

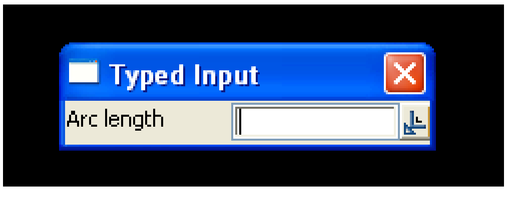
STEP 1:12d

The 1st point is selected with the mouse or entered in via the keyboard. To specify the 1st point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter the 1st point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.



STEP 2:

After the start point is accepted, the Arc length Input box will appear. Type the arc length value into the Input box and press the enter key. The browse button  on the Input box can be used to define the arc length by measuring existing elements.

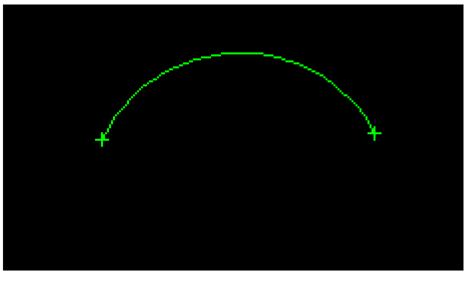


STEP 3:

The 2nd point is selected and accepted.

STEP 4:

If a solution exists, an arc is constructed using the given information.



15.13.8 2 points and end bearing

Position of option on menu: **CAD =>Arc =>2 points and end bearing**

or by selection of appropriate icon from the toolbar.

This option creates an arc that starts and end on two selected points and has a given bearing of the tangent at the end point (the end bearing).

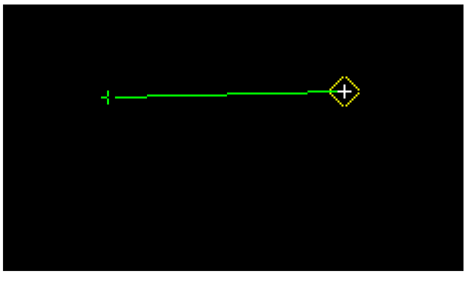
On selecting **2 points and end bearing**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

The 1st point is selected with the mouse or entered in via the keyboard. To specify the 1st point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter the 1st point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.

STEP 2:

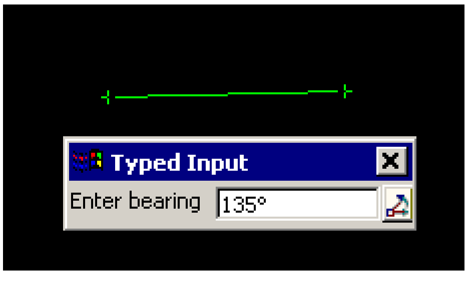
The 2nd point is selected and accepted.



STEP 3:

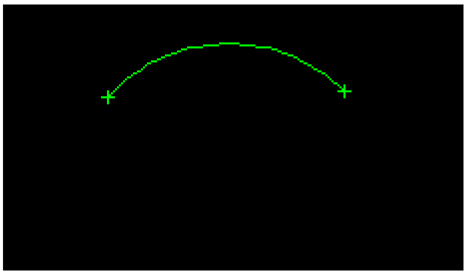
After the 2nd point is accepted, the Enter bearing input box will appear. Type the bearing into the Input box and press the enter key. The browse button on the input box can be used to define the bearing by measuring existing elements.

Note: **The Page Up and Page Down keys can be used when the Enter Bearing input box comes up to add or subtract intervals of 90 degrees.**



STEP 4:

If a solution exists, the arc is fitted through the two selected points with the end point of the given bearing.



15.13.9 Centre and end points

Position of option on menu: **CAD =>Arc =>Centre and end points**

or by selection of appropriate icon from the toolbar. 

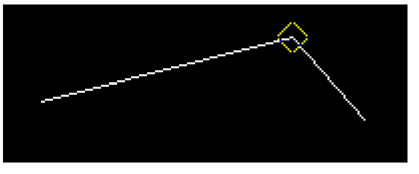
This option creates an arc by selecting in order, the centre point and the start and end points. The radius of the arc is the distance between the centre and the start point.

On selecting **Centre and end points**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

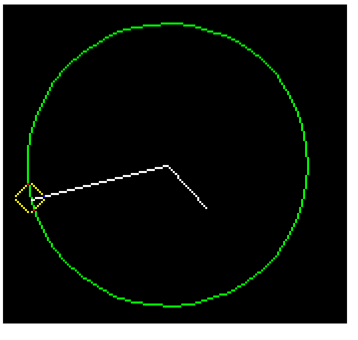
A centre point is selected with the mouse or entered in via the keyboard. To specify a centre point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter a centre point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.

After the centre point is accepted a circle will be displayed 'rubber banding' to the various solutions according to the position of the cursor. This will continue until the start point is selected and accepted.



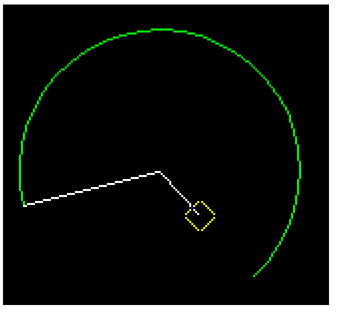
STEP 2:

Select and accept the start point. This point defines the radius and the start of the arc.



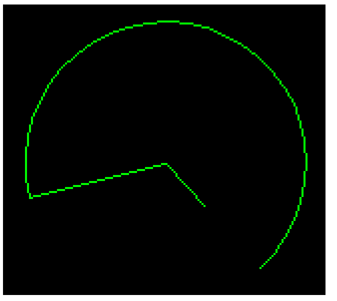
STEP 3:

The end point of the arc is specified.



STEP 4:

The arc is constructed using the given information.



15.13.10 Centre, radius and end points

Position of option on menu: **CAD =>Arc =>Centre, radius and end points**

or by selection of appropriate icon from the toolbar.

This option allows the creation of an arc given a centre point, radius, start and end points.

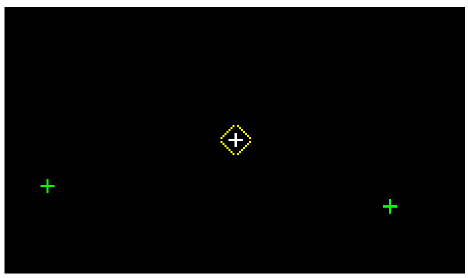
This option creates an arc of a given radius by selecting in the centre point and the start and end positions.

On selecting **Centre, radius and end points** the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

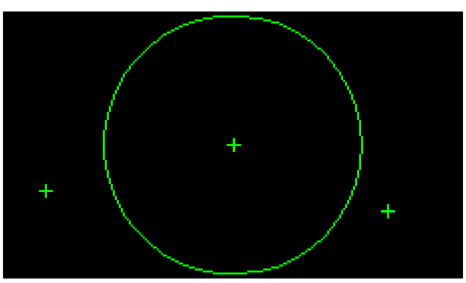
A centre point is selected with the mouse or entered in via the keyboard. To specify a centre point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter a centre point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.

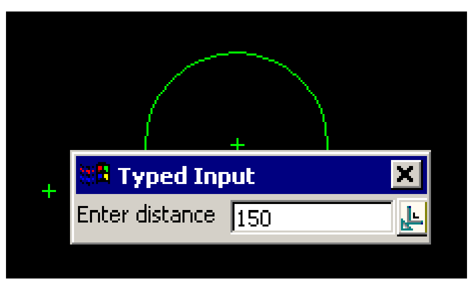
After the centre point is accepted a circle will be displayed ‘rubber banding’ to the various solutions according to the position of the cursor. This will continue until the start point is selected and accepted.



STEP 2:

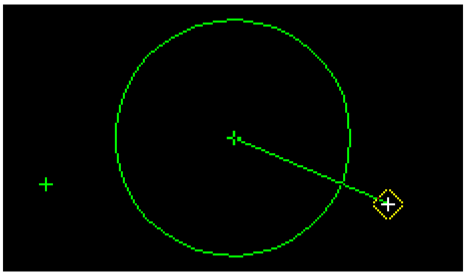
A radius value is selected with the mouse or entered in via the keyboard. To specify a radius value with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter radius value with the keyboard, simply start typing or press the space bar to bring up the Radius Input box. Type the radius value into the Radius Input box and press the Enter key. The browse button on the Input box can be used to define the arc distance by measuring existing elements.

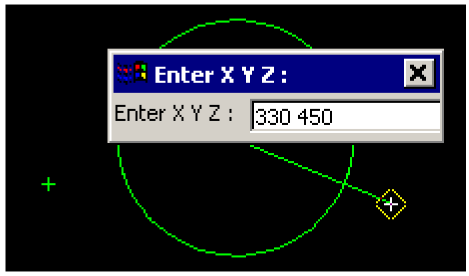




STEP 3:

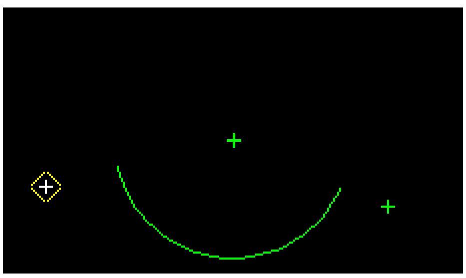
A start point defines the start of the sweep angle to define where to start the arc. It can be selected with the mouse or entered in via the keyboard. To specify a start point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter a start point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the Enter key.

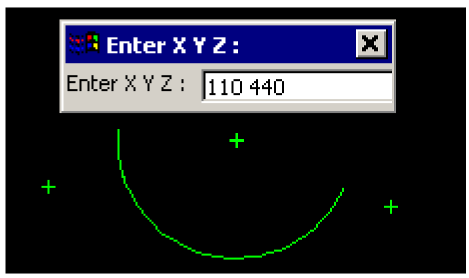




STEP 4:

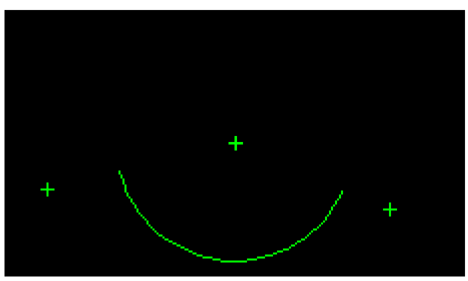
The end point of the arc is specified. The start and end points define the sweep angle to define the arc. The rubber banding of the arc will use the cursor position as the end position until a end point is selected and accepted. It can be selected with the mouse or entered in via the keyboard. To specify a start point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter a start point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key. The end point does not have to be on the arc itself. It is used to define the sweep angle.





STEP 5:

The arc is constructed using the given information.



15.13.11 Centre, start point and sweep angle

Position of option on menu: **CAD =>Arc =>Centre, start point and sweep angle**

or by selection of appropriate icon from the toolbar.

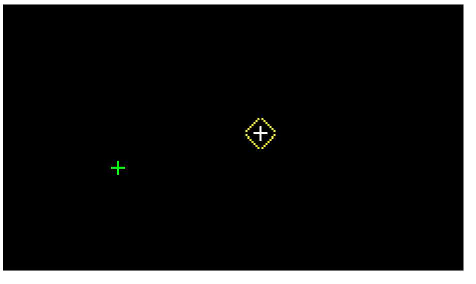
This option creates an arc by selecting a centre point and a start points and giving a sweep angle. The radius of the arc is the distance between the centre and start point.

On selecting **Centre, start point and sweep angle**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

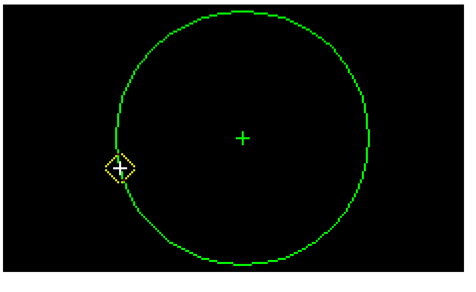
A centre point is selected with the mouse or entered in via the keyboard. To specify a centre point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter a centre point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.

After the 1st point is accepted a circle will be displayed ‘rubber banding’ to the various solutions according to the position of the cursor. This will continue until the 2nd point is selected and accepted.



STEP 2:

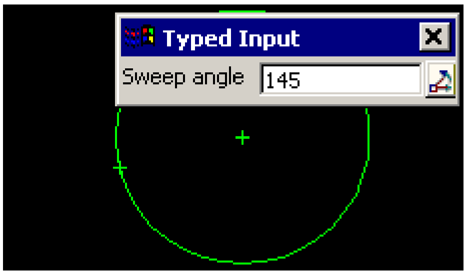
The 2nd point is selected with the mouse or entered in via the keyboard. This point defines the radius as well as the start point of the arc.To specify the 2nd point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter the 2nd point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.



STEP 3:

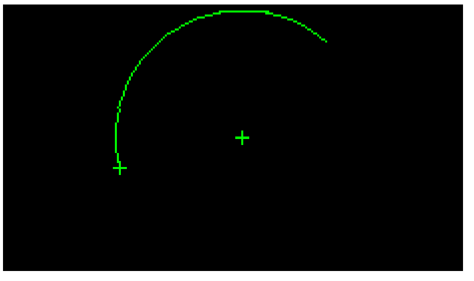
After the 2nd point is accepted, the **Sweep Angle** input box will appear. Type the sweep angle value into the input box and press the enter key. The browse button on the Input box can be used to define the arc radius by measuring existing elements.

Note: **The Page up and page down keys can be used when the input angle box comes up to add or subtract intervals of 90 degrees.**



STEP 4:

The arc is created using the information supplied



15.13.12 Start point, radius and bearing

Position of option on menu: **CAD =>Arc =>Start point, radius and bearing**

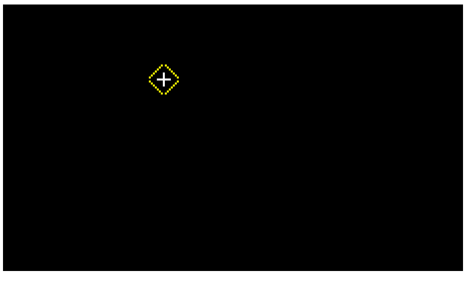
or by selection of appropriate icon from the toolbar.

This option creates an arc of a given radius, starting at a selected point and bearing of the tangent at the start point and going for a given arc length.

On selecting the **Start point, radius and bearing** option, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

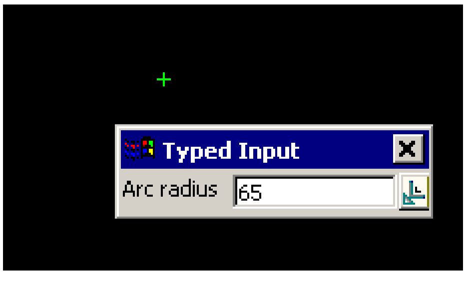
STEP 1:

A start point is selected with the mouse or entered in via the keyboard. To specify a start point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter a start point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.



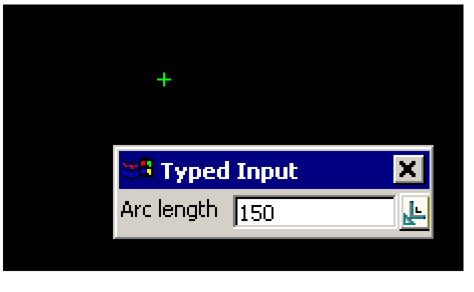
STEP 2:

After the start point is accepted, the Arc Radius Input box will appear. Type the radius value into the Input box and press the enter key. The browse button on the Input box can be used to define the arc radius by measuring existing elements.



STEP 3:

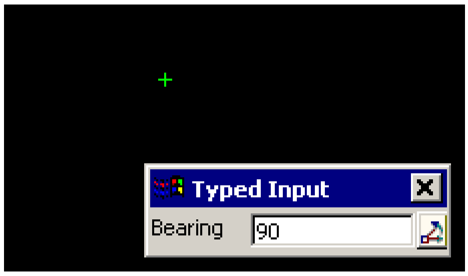
After the radius has been entered, the **Arc Length** Input box will appear. Type the arc length into the Input box and press the enter key. The browse button on the Input box can be used to define the arc distance by measuring existing elements.



STEP 4:

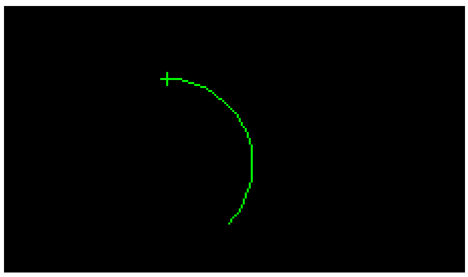
After the arc length is accepted, the Bearing Input box will appear. Type the bearing into the Input box and press the enter key. The browse button on the Input box can be used to define the bearing by measuring existing elements.

Note: **The Page Up and Page Down keys can be used when the Bearing Input box comes up to add or subtract intervals of 90 degrees.**



STEP5:

The arc is created with the given information.



15.13.13 Start point, radius and chord

Position of option on menu: **CAD =>Arc =>Start point, radius and chord**

or by selection of appropriate icon from the toolbar.

This option creates an arc of a given radius, starting from a selected point and with a given chord bearing at the start point and a given arc length.

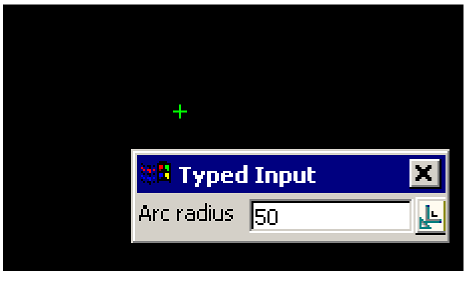
On selecting **Start point, radius and chord**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

The start point is selected with the mouse or entered in via the keyboard. To specify the start point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter the start point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.

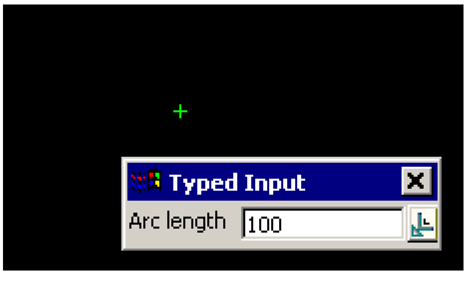
STEP 2:

After the start point is accepted, the **Arc Radius** Input box will appear. Type the radius value into the Input box and press the enter key. The browse button on the Input box can be used to define the arc radius by measuring existing elements.



STEP 3:

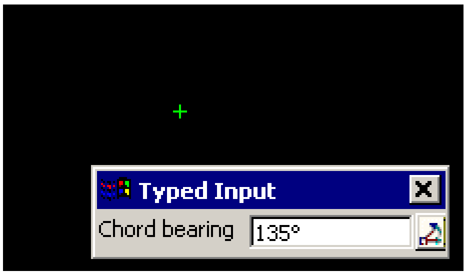
After the arc radius is entered, the **Arc Length** Input box will appear. Type the arc length into the Input box and press the enter key. The browse button on the Input box can be used to define the arc distance by measuring existing elements.



STEP 4:

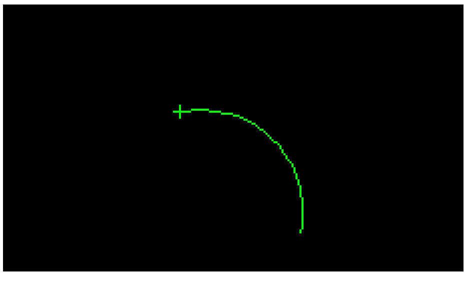
After the arc length is entered, the **Chord Bearing** Input box will appear. Type the bearing into the Input box and press the enter key. The browse button on the Input box can be used to define the bearing by measuring existing elements.

Note: **The Page Up and Page Down keys can be used when the Chord Bearing Input box comes up to add or subtract intervals of 90 degrees.**



STEP 5:

An arc is created with the given information.



15.13.14 Start point, radius, chord length and bearing

Position of option on menu: **CAD =>Arc =>Start point, radius, chord length and bearing**

or by selection of appropriate icon from the toolbar.

This option creates an arc of a given radius, starting from a selected point, with a given bearing of the tangent at the start point and a given chord length from the start point.

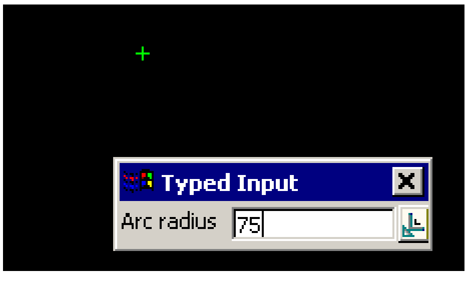
On selecting **Start point, radius, chord length and bearing**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

A start point is selected with the mouse or entered in via the keyboard. To specify a start point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter a start point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.

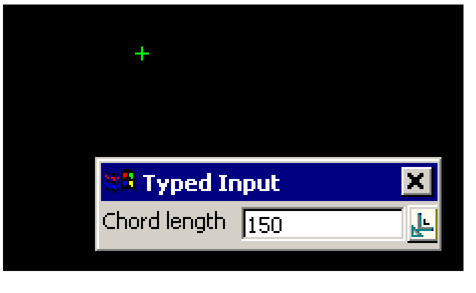
STEP 2:

After the start point is accepted, the **Arc Radius** Input box will appear. Type the radius value into the Input box and press the enter key. The browse button on the Input box can be used to define the arc radius by measuring existing elements.



STEP 3:

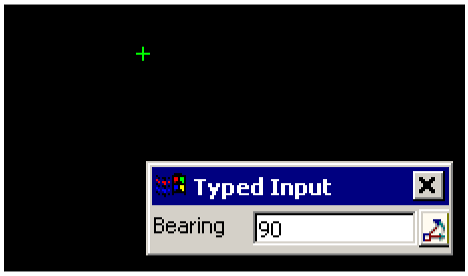
After the radius has been entered, the **Chord Length** Input box will appear. Type the chord length into the Input box and press the enter key. The browse button on the Input box can be used to define the chord length by measuring existing elements.



STEP 4:

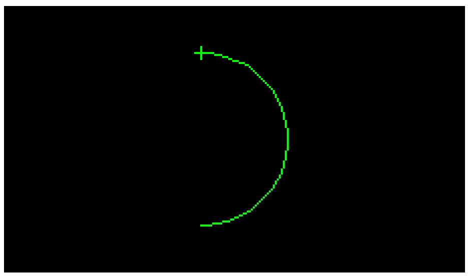
After the chord length has been entered, the **Bearing** Input box will appear. Type the bearing of the start tangent into the Input box and press the enter key. The browse button on the Input box can be used to define the bearing by measuring existing elements.

Note: **The Page Up and Page Down keys can be used when the Bearing Input box comes up to add or subtract intervals of 90 degrees.**



STEP 5:

The arc is created with the given information.



15.13.15 Fillet

Position of option on menu: **CAD =>Arc =>Fillet**

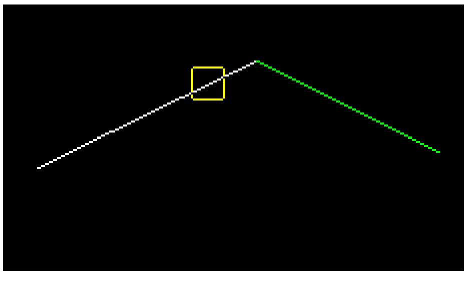
or by selection of appropriate icon from the toolbar.

This option creates an arc given a fillet radius

On selecting **Fillet**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

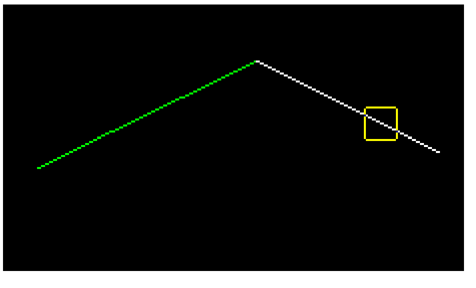
STEP 1:

The 1st segment where the fillet is to depart from is selected and accepted. This should be done by a pick with direction. The direction will influence the calculations as there are a number of solutions for intersecting segments. In the case shown, the direction was in a north-east direction.



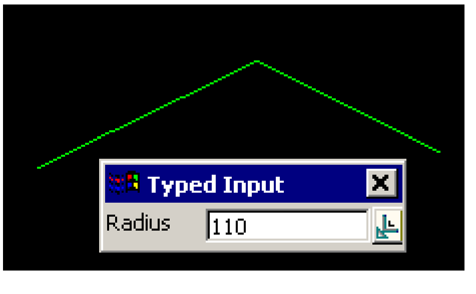
STEP 2:

The 2nd segment where the fillet is to arrive from is selected and accepted. This should be done by a pick with direction. The direction will again influence the calculations as there are a number of solutions for intersecting segments. In the case shown the direction was in a south-east direction



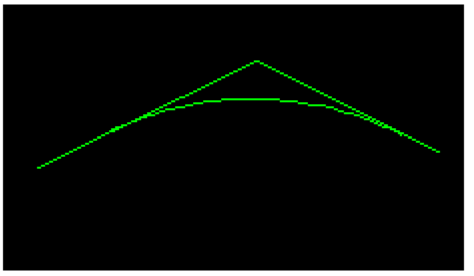
STEP 3:

After the 2nd segment is accepted, the Arc Radius Input box will appear. Type the radius of the fillet in the Input box (positive radius for curves curving to the right) and press the enter key. The browse button on the Input box can be used to define the arc radius by measuring existing elements.



STEP 4:

The fillet arc is placed between the selected segments.



15.13.16 Fillet by start point

Position of option on menu: **CAD =>Arc =>Fillet by start point**

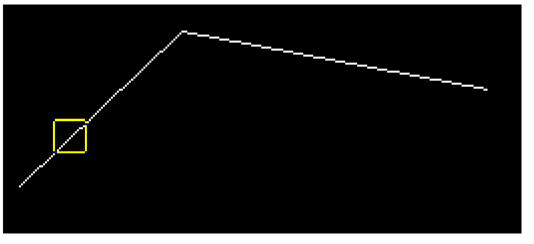
or by selection of appropriate icon from the toolbar. 

This option creates an arc that is a fillet between two selected sections and with a selected fillet start point (which will be a tangent point of the created arc).

On selecting **Fillet by start point**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

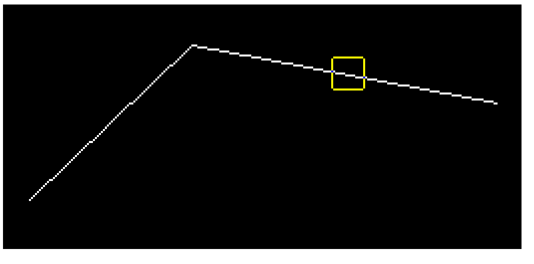
STEP 1:

The 1st segment where the fillet is to depart from is selected and accepted. This should be done by a pick with direction. The direction will influence the calculations as there are a number of solutions for intersecting segments. In the case shown the direction was in a north-east direction.



STEP 2:

The 2nd segment where the fillet is to arrive from is selected and accepted. This should be done by a pick with direction. The direction will again influence the calculations as there are a number of solutions for intersecting segments. In the case shown the direction was in a south-east direction

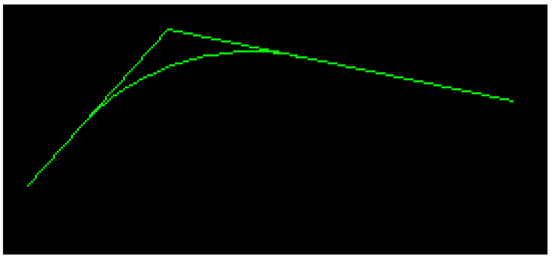


STEP 3:

The tangent start point of the fillet is selected and accepted.

STEP 4:

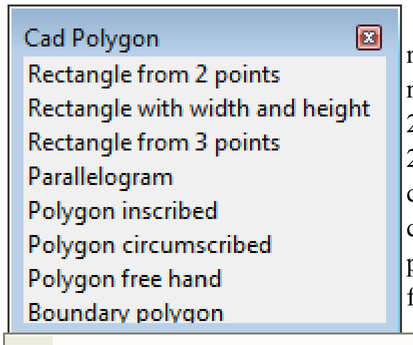
The fillet arc is placed between the selected segments.



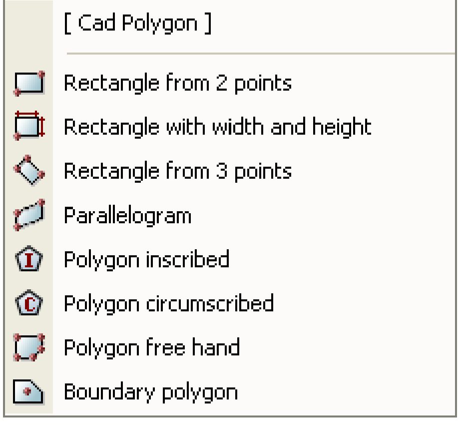
15.14 CAD Polygon

Position of option on menu: **CAD =>Polygon**

The **Cad polygon** walk-right menu is



rectangle (fillet/chamfer) parallel to axes by picking 2 diagonal corners
rectangle parallel to axes by picking BL point, width and height
2 points defining base, 3rd point defining height
2 points defining base, 3rd point defining corner
centre point, number of sides and point on circumference
centre point, number of sides and point on circumference
pick in polygon vertices (in order)
find polygon from lines and arcs, surrounding selected point



For the option *Rectangle from 2 points*, go to 15.14.1 Rectangle from 2 points
Rectangle with width and height 15.14.2 Rectangle with width and height
Rectangle from 3 points 15.14.3 Rectangle from 3 points
Parallelogram 15.14.4 Parallelogram
Polygon inscribed 15.14.5 Polygon inscribed (in a circle)
Polygon circumscribed 15.14.6 Polygon Circumscribed About a Circle
Polygon free hand 15.14.7 Free Hand Polygon
Boundary polygon 15.14.8 Create Boundary Polygon

15.14.1 Rectangle from 2 points

Position of option on menu: **CAD =>Polygon =>Rectangle from 2 Points**

or by selection of appropriate icon from the toolbar.

This option creates a rectangle parallel to the x and y axis by selecting two diagonal corners of the rectangle.

On selecting **Rectangle from 2 Points**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

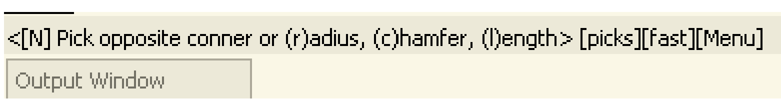
STEP 1:

The 1st point is selected with the mouse or entered in via the keyboard. To specify the 1st point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter the 1st point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.

After the 1st point is accepted a rectangle will be displayed 'rubber banding' to form a rectangle with the first point and the cursor position being diagonal corners. This will continue until the 2nd point is selected and accepted.

STEP 2:

The following message is displayed in the screen message box:



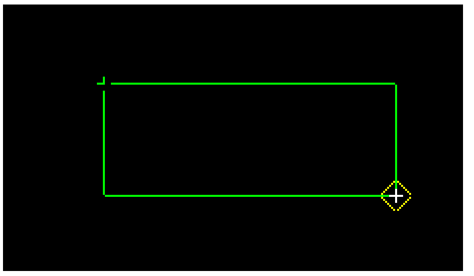
If filleted corners are required, type **r** and enter the *corner radius*.

If chamfered corners are required, type **c** and enter the *chamfer length*.

If a fillet or chamfer has been selected, type **n** to turn it off.

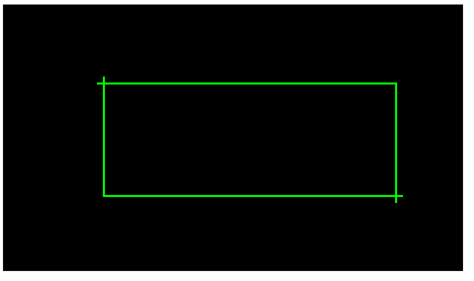
STEP 3:

The 2nd point is selected with the mouse or entered in via the keyboard. To specify the 2nd point with the mouse, a point must be selected and accepted. To enter the 2nd point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.



STEP 4:

The rectangle is created with the given information.



15.14.2 Rectangle with width and height

Position of option on menu: **CAD =>Polygon =>Rectangle with width and height**

or by selection of appropriate icon from the toolbar.

This option creates a rectangle parallel to the x and y axis by selecting a start point and giving the width and height of the rectangle.

On selecting **Rectangle with width and height**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

The start point is selected with the mouse or entered in via the keyboard. To specify the start point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter the start point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.

This point defines the bottom left point of the rectangle.

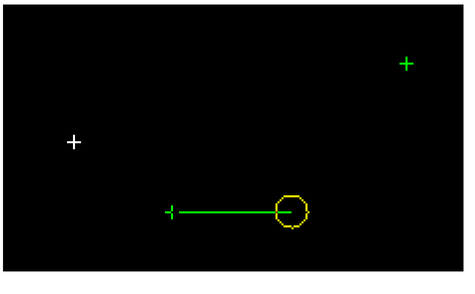
After the start point is accepted, the width will be displayed ‘rubber banding’ to right of the start point (positive distance). The width is equal to the distance from the start point to the position of the cursor. This will continue until the width is selected and accepted.

STEP 2:

The width is selected with the mouse or entered in via the keyboard. To specify the width with the mouse, a point must be selected and accepted. To enter the width with the keyboard, simply start typing or press the space bar to bring up the Distance Input box. Type the distance into the Distance Input box and press the enter key. The browse button on the Input box can be used to define the distance by measuring existing elements.

A negative distance (to the left of the 1st point) can be entered via the Distance Input box in two ways. Either type a negative distance, or use the browse button to select a distance, then place a minus "-" sign in front of the distance value.

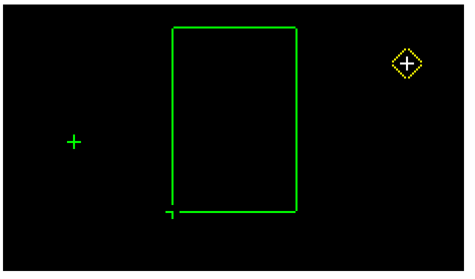
After the width is accepted, the width of the rectangle is confirmed and the height will be displayed ‘rubber banding’ upwards on the screen. The height is equal to the distance from the found width point to the position of the cursor. The rectangle will be created once the height is selected and accepted.



STEP 3:

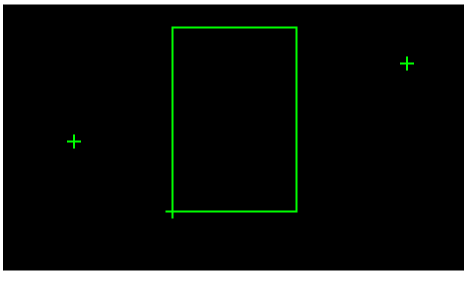
The height is selected with the mouse or entered in via the keyboard. To specify the height with the mouse, a point must be selected and accepted. To enter the height with the keyboard, simply start typing or press the space bar to bring up the Distance Input box. Type the distance into the Distance Input box and press the enter key. The browse button on the Input box can be used to define the distance by measuring existing elements.

A negative distance (below of the 2nd point) can be entered via the Distance Input box in two ways. Either type a negative distance, or use the browse button to select a distance, then place a minus "-" sign in front of the distance value.



STEP 4:

The rectangle is created with the given information.



15.14.3 Rectangle from 3 points

Position of option on menu: **CAD =>Polygon =>Rectangle from 3 points**

or by selection of appropriate icon from the toolbar.

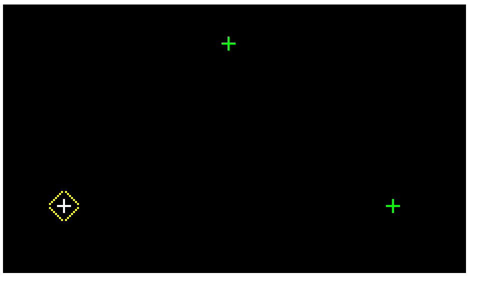
This option creates a rectangle by selecting three points. The first and second point define the base of the rectangle, which can be at any angle, and the third point is on the opposite side of the rectangle. The rectangle can be filleted or chamfered.

On selecting **Rectangle from 3 points**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

The 1st point is selected with the mouse or entered in via the keyboard. To specify the 1st point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter the 1st point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.

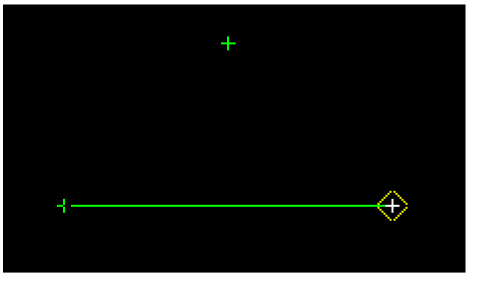
After the 1st point is accepted, the base of the rectangle will be displayed ‘rubber banding’ on the screen. The baseline length is equal to the distance from the 1st point to the position of the cursor. This will continue until the 2nd point is selected and accepted.



STEP 2:

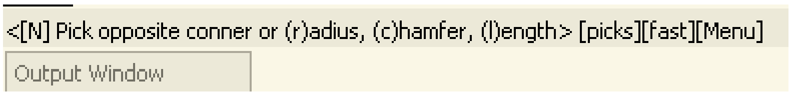
The 2nd point is selected with the mouse or entered in via the keyboard. To specify the 2nd point with the mouse, a point must be selected and accepted. To enter the 2nd point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key. The browse button on the Input box can be used to define the distance by measuring existing elements.

After the 2nd point is accepted, the baseline of the rectangle is confirmed and the height will be displayed ‘rubber banding’ on the screen. The height is equal to the distance from the 2nd point to the position of the cursor. The rectangle will be created once the height is selected and accepted.



STEP 3:

The following message is displayed in the screen message box:



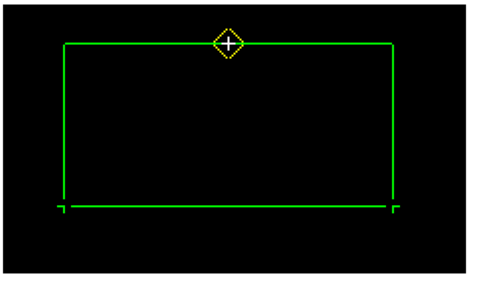
If filleted corners are required, type **r** and enter the *corner radius*.

If chamfered corners are required, type **c** and enter the *chamfer length*.

If a fillet or chamfer has been selected, type **n** to turn it off.

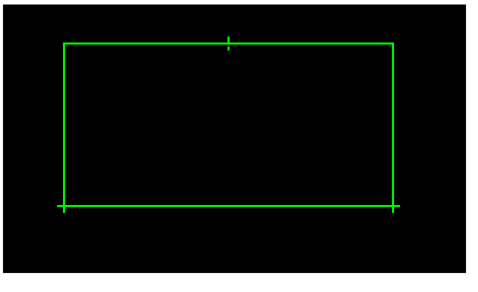
STEP 4:

The height is selected with the mouse or entered in via the keyboard. To specify the height with the mouse, a point must be selected and accepted. To enter the 3rd point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key. The browse button on the Input box can be used to define the distance by measuring existing elements.



STEP 5:

The rectangle is created with the given information.



15.14.4 Parallelogram

Position of option on menu: **CAD =>Polygon =>Parallelogram**

or by selection of appropriate icon from the toolbar.

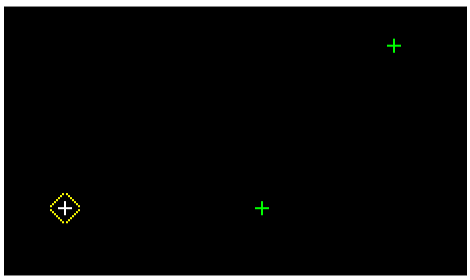
This option creates a parallelogram by selecting three points. The first and second point define the base of the parallelogram, which can be at any angle, and the third point is the end point of the opposite side of the parallelogram.

On selecting **Parallelogram**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

The 1st point is selected with the mouse or entered in via the keyboard. To specify the 1st point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter the 1st point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.

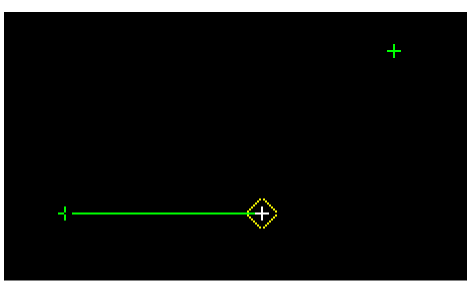
After the 1st point is accepted, the base of the rectangle will be displayed 'rubber banding' on the screen. The baseline length is equal to the distance from the 1st point to the position of the cursor. This will continue until the 2nd point is selected and accepted.



STEP 2:

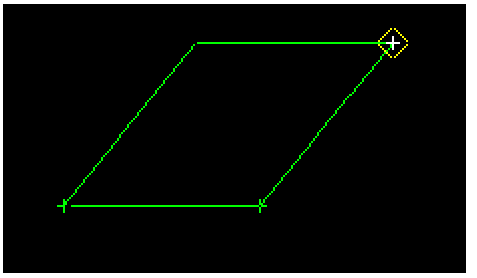
The 2nd point is selected with the mouse or entered in via the keyboard. To specify the 2nd point with the mouse, a point must be selected and accepted. To enter the 2nd point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key. The browse button on the Input box can be used to define the distance by measuring existing elements.

After the 2nd point is accepted, the baseline of the parallelogram is confirmed and the height will be displayed 'rubber banding' on the screen. The height is equal to the distance from the 2nd point to the position of the cursor. The parallelogram will be created once the height is selected and accepted.



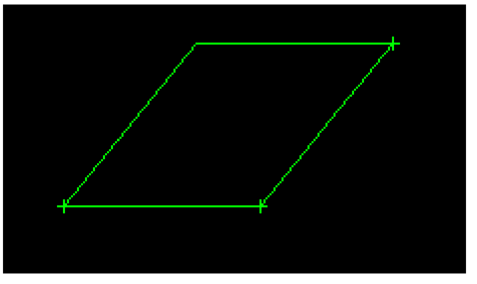
STEP 3:

The height is selected with the mouse or entered in via the keyboard. To specify the height with the mouse, a point must be selected and accepted. To enter the 3rd point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key. The browse button on the Input box can be used to define the distance by measuring existing elements.



STEP 4:

The parallelogram is created with the given information.



15.14.5 Polygon inscribed (in a circle)

Position of option on menu: **CAD =>Polygon =>polygon inscribed**

or by selection of appropriate icon from the toolbar. 

This option creates a polygon with a user defined number of sides on the inside of a a circle.

The polygon is constructed inside a circle, thus each corner of the polygon touches the circumference of the circle defined by the user. Conversely using the **Polygon circumscribed** option, the midpoint of each side of the polygon touches the circumference of the circle creating a polygon outside the circle.

On selecting **polygon inscribed**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

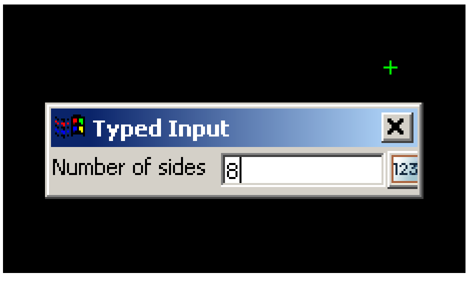
STEP 1:

The centre point is selected with the mouse or entered in via the keyboard. To specify the centre point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter the centre point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.

STEP 2:

After the centre point is accepted, the Number of Sides Input box will appear. Type in the desired number of sides for the polygon into the Input box and press the enter key.

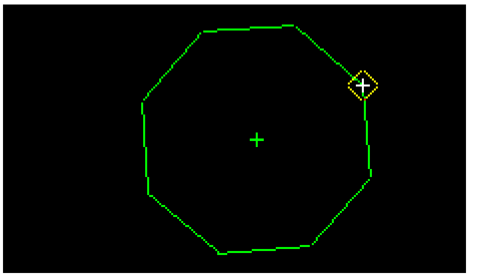
After the number of sides have been selected, the polygon will be displayed 'rubber banding' on the screen. The polygon will be created once the radius is selected and accepted.



STEP 3:

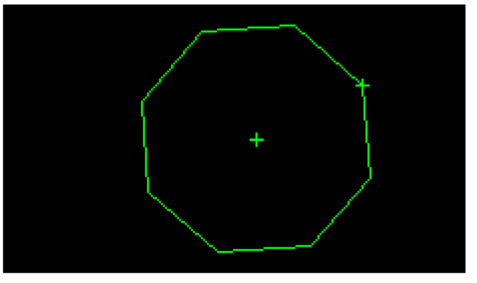
Before a polygon can be created inside the circle, its radius must be defined by the user. To specify the radius with the mouse, a point must be selected and accepted. To enter the point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.

Note: One of the polygon's corners will be located on the point selected to defined the circle's radius



STEP 4:

A polygon is created inside a circle with a radius defined by the user.



15.14.6 Polygon Circumscribed About a Circle

Position of option on menu: **CAD =>Polygon =>Polygon circumscribed**

or by selection of appropriate icon from the toolbar.

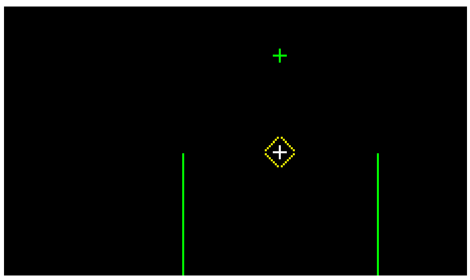
This option creates a polygon with a user defined number of sides on the outside a circle.

The polygon is constructed outside a circle, thus the midpoint of each side of the polygon touches the circumference of the circle defined by the user. Conversely using the **Polygon inscribed** option, each corner of the polygon touches the circumference of the circle creating a polygon inside the circle.

On selecting **Polygon circumscribed**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

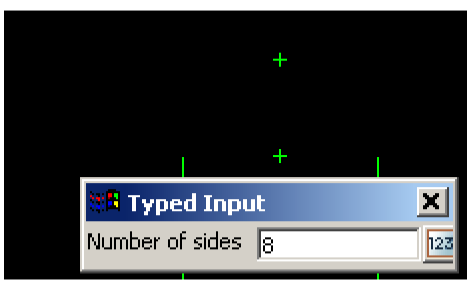
The centre point is selected with the mouse or entered in via the keyboard. To specify the centre point with the mouse, a point must be selected (Left Button) and accepted (Middle Button). To enter the centre point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.



STEP 2:

After the centre point is accepted, the Number of Sides Input box will appear. Type in the desired number of sides for the polygon into the Input box and press the enter key.

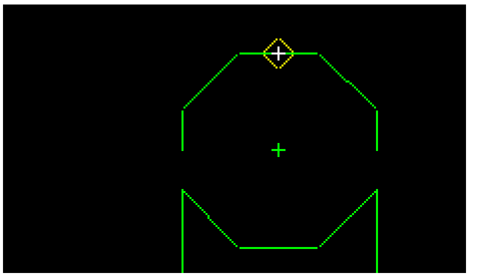
After the number of sides have been selected, the polygon will be displayed 'rubber banding' on the screen. The polygon will be created once the radius is selected and accepted.



STEP 3:

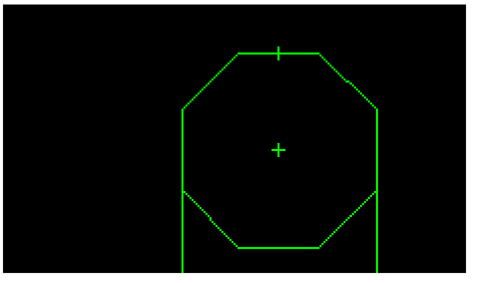
Before a polygon can be created outside the circle, its radius must be defined by the user. To specify the radius with the mouse, a point must be selected and accepted. To enter the point with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the enter key.

Note: One midpoint of the polygon's side will be located on the point selected to defined the circle's radius



STEP 4:

A polygon is created outside the circle with a radius defined by the user.



15.14.7 Free Hand Polygon

Position of option on menu: **CAD =>Polygon =>Polygon free hand**

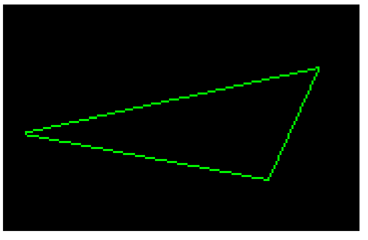
or by selection of appropriate icon from the toolbar. 

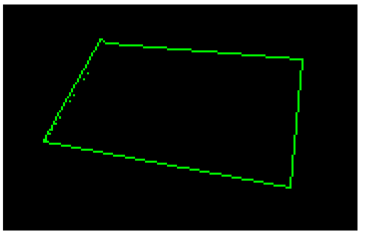
This option creates a closed polygon from selected vertices.

On selecting **Polygon free hand**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

Pick and accept the position to form the vertices of the polygon. As each vertex is accepted, the polygon is automatically closed.





STEP 2:

The polygon is finished by pressing the <Esc> key.

15.14.8 Create Boundary Polygon

Position of option on menu: **CAD =>Polygon =>Boundary polygon**

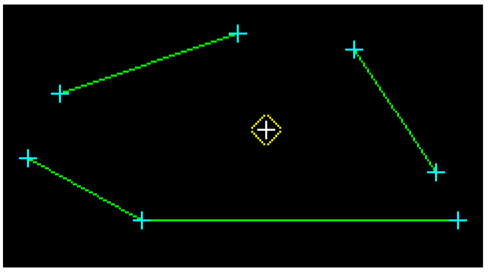
or by selection of appropriate icon from the toolbar.

This option tries to form a boundary from a selected set of segments.

On selecting **Boundary polygon**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

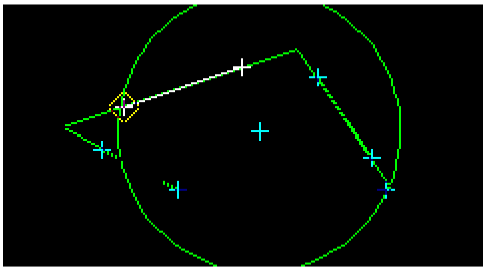
STEP 1:

Pick a position inside the lines, arcs and strings to form a bounding polygon from.



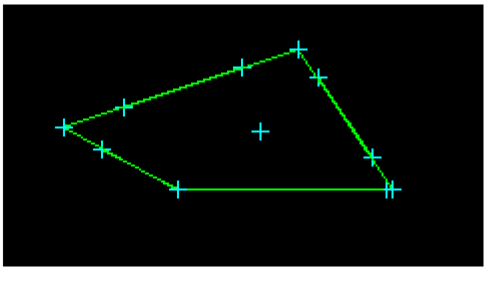
STEP 2:

A circle is then drawn centred on the selected position with circumference going through the position of the cursor. Drag the cursor out until it encloses all the data to be used to create the closest bounding polygon around the centre.



STEP 3:

On accepting the circle position, an attempt is made to create the inner polygon using all the strings inside the circle.



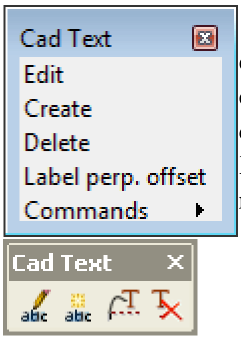
15.15 CAD Text

Position of option on menu: **CAD =>Text**

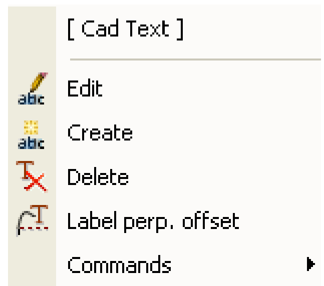
Position of option on menu: **Various toolbars**

Text can occur as a text string, on vertices and segments of a super string or on vertices of a 4d string. But the display position of text is not as straight forward as simply placing a point. For more information about text, go to the section 4.6 Text Definitions.

The **Text** walk-right menu is



edit text string using typed commands and grips
create text string using typed commands and grips
delete text
label perpendicular offset distance
menu of individual Text options



For the option *Edit*, go to	15.15.2 Edit Text
Create	15.15.1 Create Text
Delete	15.15.20 Text Delete
Label perp. offset	15.15.18 Label Perpendicular Offset
Commands	15.15.3 CAD Text Commands

15.15.1 Create Text

Position of option on menu: **CAD =>Text =>Create**

or by selection of appropriate icon from the toolbar.

For information about text and its definitions, go to the section 4.6 Text Definitions.

12d Model has two CAD text options ***Create*** and ***Edit***. However both of these run the same option which starts up in different modes - ***Create*** is in the mode to create text as straight text, or text on a super string vertex or on a super string segment and ***Edit*** which edits existing text.

Once either CAD text option is started it is possible to change between the ***Create*** and ***Edit*** modes at any time.

The ***Create*** option creates text strings, text at super string vertices and text on super string segments, and then displays a list key commands in the screen message area, and grips on screen, to quickly allow editing of the text, text size, colour, rotation, style, angle, raise, justification point etc.

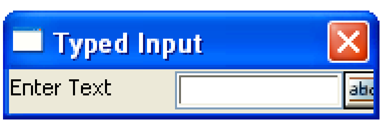
After selecting the option, the ***Pick position*** message is displayed in the screen message area.

To go from **Create** to **Edit** mode, type **e** - go to To Go from Create to Edit Mode - Typing e
To create a text string, type **c** - go to Creating a Text String - Typing c
To create vertex text, type **v** - go to Creating Text at a Super String Vertex- Typing v
To create segment text, type **s** - go to Creating Text at a Super String Segment- Typing s

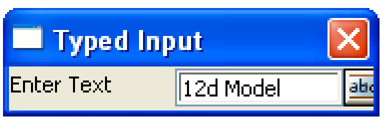
Creating a Text String - Typing c

After typing **c**, the ***Pick position*** message is again displayed in the screen message area and the position of the text is then selected with the cursor.

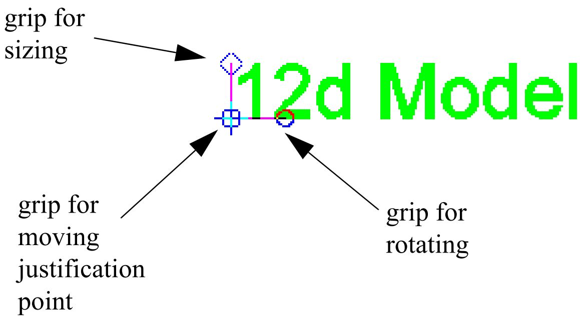
When the position is accepted an **Enter text** typed input box is then displayed on the screen.



The text is typed into the box and the <enter> key pressed.

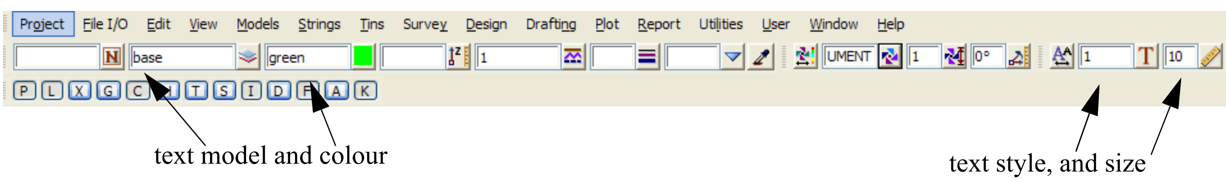


The text vertex and text justification point is placed at that position and the direction of the text is parallel to the x-axis. The angle and raise are both zero and the text justification is (left x, bottom y).

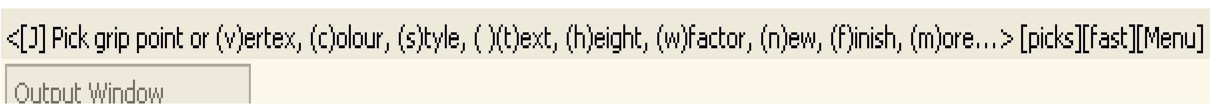


When the text is displayed, grips for quick editing of size, rotation and moving of the justification point are also shown. The appropriate grip can then selected for quick editing. For details on using the grips, go to the section 4.14 Text Grips.

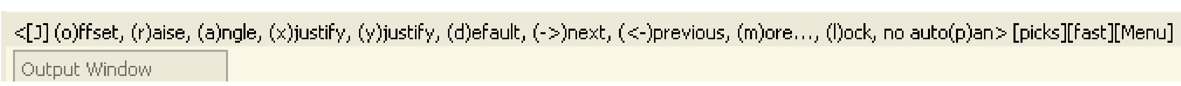
For the new text, the style and height are taken from the ***Text controlbar*** and the model and colour from the ***CAD controlbar*** (see 15.1 Controlbars,15.1.2 Symbol Controlbar and 15.1.1 CAD Controlbar).



After the text is created, a list of typed options is then displayed in the screen message area.



and if **m** is typed, the rest of the options are shown

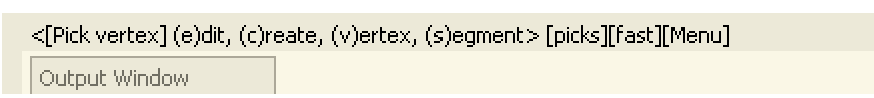


These option are the same as when editing existing text, and will be described in detail in the section 15.15.2 Edit Text.

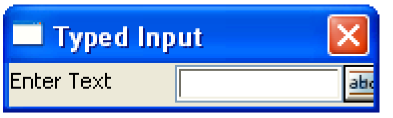
c is the default if nothing is typed.

Creating Text at a Super String Vertex- Typing v

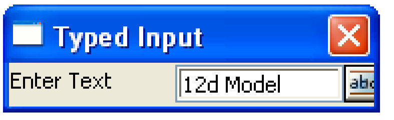
After typing **v**, the ***Pick vertex*** message is displayed in the screen message area and the vertex of the super string to place the text at is then selected with the cursor.



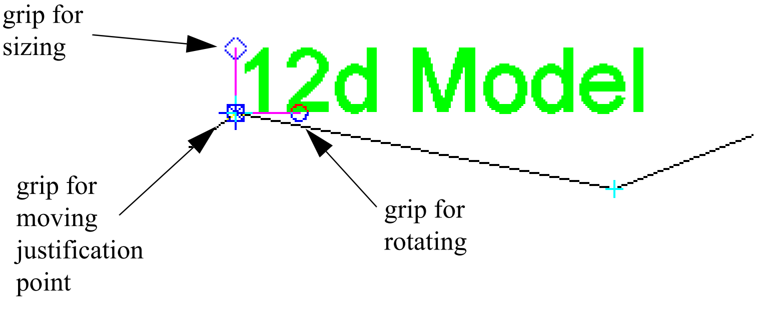
When the vertex is accepted an **Enter text** typed input box is then displayed on the screen.



The text is typed into the box and the <enter> key pressed.

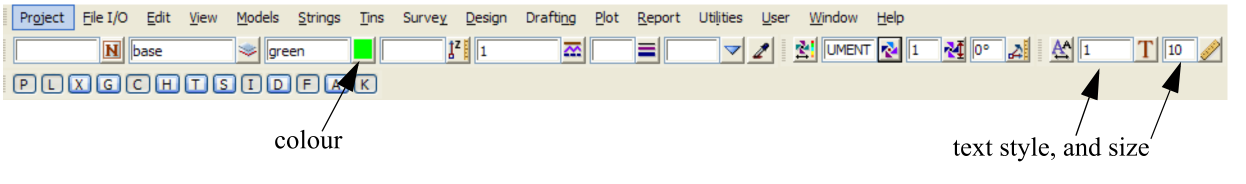


The text justification point is placed at the selected string vertex and the direction of the text is parallel to the x-axis. The angle and raise are both zero and the text justification is (left x, bottom y).

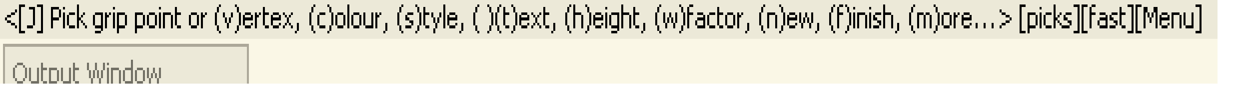


When the text is displayed, grips for quick editing of size, rotation, moving of the justification point and string vertex are also shown. The appropriate grip can then selected for quick editing. For details on using the grips, go to the section 4.14 Text Grips.

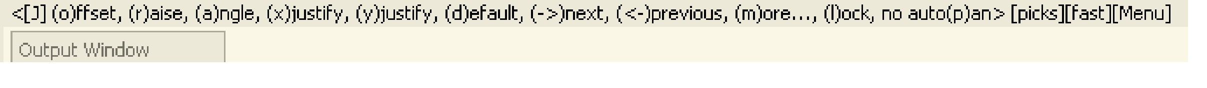
For the new text, the style and height are taken from the ***Text controlbar*** and colour from the ***CAD controlbar*** (see 15.1 Controlbars,15.1.2 Symbol Controlbar and 15.1.1 CAD Controlbar). The created text will be part of the super string so no model is required.



After the vertex text is created, a list of typed options is then displayed in the screen message area.



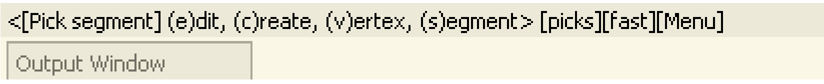
and if **m** is typed, the rest of the options are shown



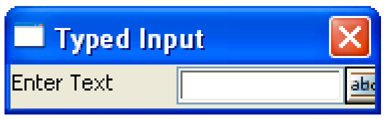
These option are the same as when editing existing text, and will be described in detail in the section 15.15.2 Edit Text.

Creating Text at a Super String Segment- Typing s

After typing **s**, the ***Pick segment*** message is displayed in the screen message area and the segment of the super string to place the text on is then selected with the cursor.

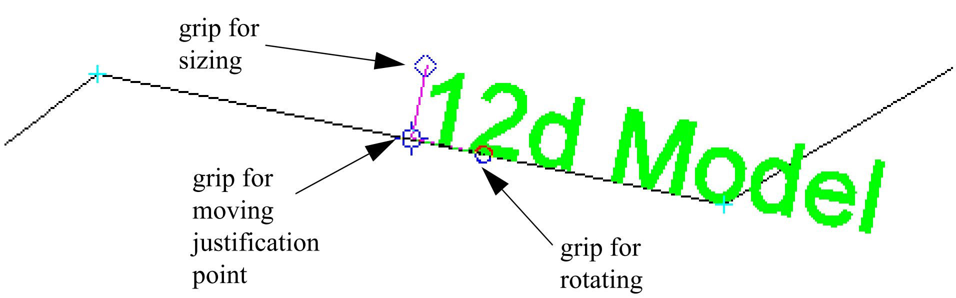


When the segment is accepted an **Enter text** typed input box is then displayed on the screen.



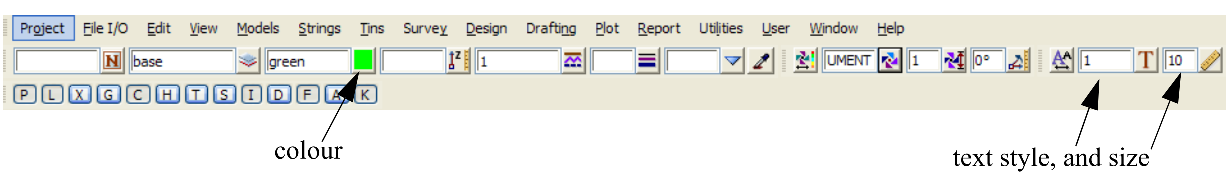
The text is typed into the box and the <enter> key pressed.

The text justification point is placed at the centre of the segment and the direction of the text is parallel to the segment. The angle and raise are both zero and the text justification is (left x, bottom y).

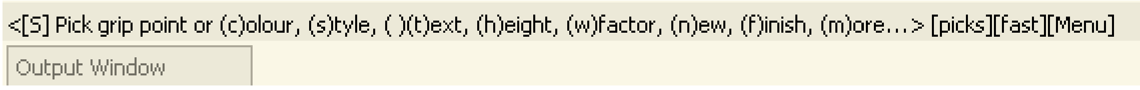


When the text is displayed, grips for quick editing of size, rotation and moving of the justification point are also shown. The appropriate grip can then selected for quick editing. For details on using the grips, go to the section 4.14 Text Grips.

For the new text, the style and height are taken from the ***Text controlbar*** and colour from the ***CAD controlbar*** (see 15.1 Controlbars,15.1.2 Symbol Controlbar and 15.1.1 CAD Controlbar). The created text will be part of the super string so no model is required.



After the segment text is created, a list of typed options is then displayed in the screen message area.

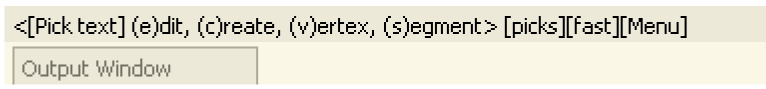


and if **m** is typed, the rest of the options are shown

These option are the same as when editing existing text, and will be described in detail in the section 15.15.2 Edit Text.

To Go from Create to Edit Mode - Typing e

After typing **e**, the ***Pick text*** message is displayed in the screen message area.



This is the same as when selecting the Edit option from the CAD Text toolbar and will be described in detail in the section 15.15.2 Edit Text.

15.15.2 Edit Text

Position of option on menu: **CAD =>Text =>Edit**

or by selection of appropriate icon from the toolbar. 

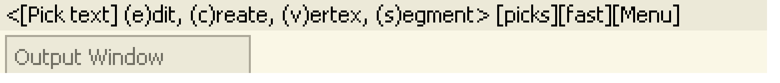
For information about text and its definitions, go to the section 4.6 Text Definitions.

12d Model has two CAD text options ***Create*** and ***Edit***. However both of these run the same option which starts up in different modes - ***Create*** is in the mode to create text as straight text, or text on a super string vertex or on a super string segment and ***Edit*** which edits existing text.

Once either CAD text option is started it is possible to change between the Create and Edit modes at any time.

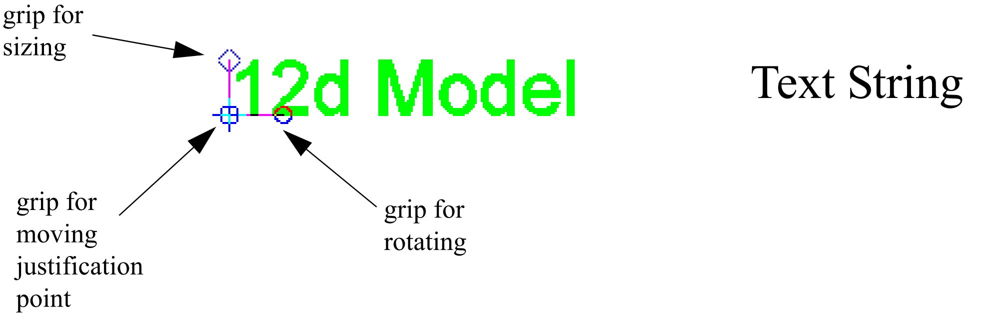
The ***Edit*** option edits text as either text strings, text at super string vertices or text on super string segments.

After typing **e**, the ***Pick text*** message is displayed in the screen message area

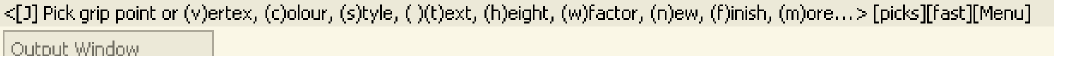


and text either as a text string, or text on a super string vertex or segment, is then selected by the mouse for editing.

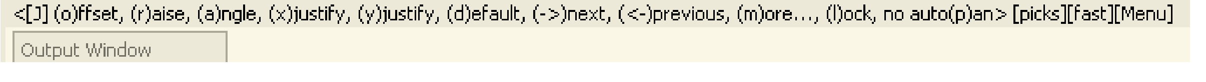
The selected text with the text edit grips is shown on screen, and a list of available typed commands in the screen message area to quickly allow editing of the text, text size, colour, rotation, style, angle, raise, justification point, vertex, etc.



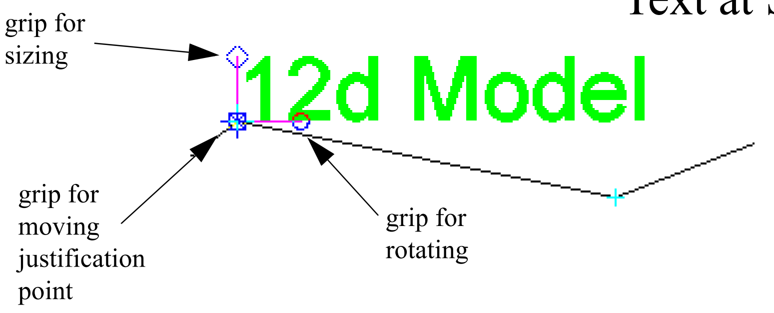
Options displayed after selecting text



and by typing **m**



Text at Super String Vertex



Options displayed after selecting text

<[J] Pick grip point or (v)ertex, (c)olour, (s)tyle, ( )(t)ext, (h)eight, (w)factor, (n)ew, (f)inish, (m)ore...> [picks][fast][Menu]

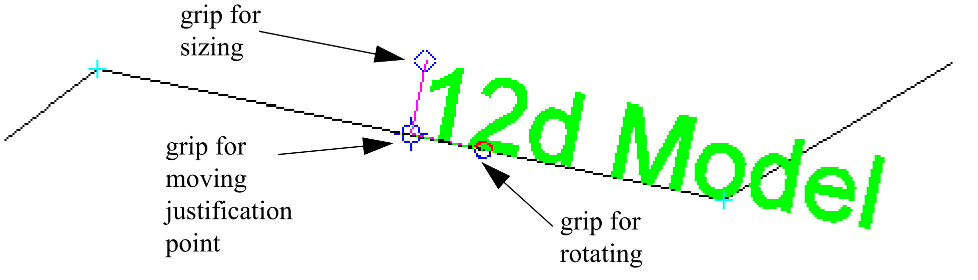
Output Window

and by typing **m**

<[J] (o)ffset, (r)aise, (a)ngle, (x)justify, (y)justify, (d)efault, (->)next, (<-)previous, (m)ore..., (l)ock, no auto(p)an> [picks][fast][Menu]

Output Window

Text on Super String Segment



Options displayed after selecting text

<[S] Pick grip point or (c)olour, (s)tyle, ( )(t)ext, (h)eight, (w)factor, (n)ew, (f)inish, (m)ore...> [picks][fast][Menu]

Output Window

and by typing **m**

<[S] (o)ffset, (r)aise, (a)ngle, (x)justify, (y)justify, (d)efault, (->)next, (<-)previous, (m)ore..., no auto(p)an, <-previous, ->next> [picks][fast]

Output Window

To change the text colour, type **c** - go to Change Text Colour - Typing c

To change the text style, type **s** - go to Change Text Style - Typing s

To change the actual text, type **t** or press **space bar** - go to Change Text - Pressing space bar or Typing t

To change the text height, type **h** - go to Change Text Height - Typing h

To change the text width factor, type **w** - go to Change Text Width Factor - Typing w

To create/edit a new text, type **n** - go to Create/Edit New Text - Typing n

To end the create/edit option, type **f** - go to Finish the Create/Edit - Typing f

To show the other line of options, type **m** - go to Show Other Line of Options - Typing m

To change the text offset, type **o** - go to Change Text Offset - Typing o

To change the text raise, type **r** - go to Change Text Raise - Typing r

To change the text angle, type **a** - go to Change Text Angle - Typing a

To change the text x justification, type **x** - go to Change Text X Justification - Typing x

To change the text y justification, type **y** - go to Change Text Y Justification - Typing y

To reset some text values to their defaults, type **d** - go to Reset Text Parameters to Defaults - Typing d

To change focus to the next text, press **->** - go to Change Focus to the Next Text in String - Pressing right arrow key (->)

To change focus to the previous text, press **<-** - go to Change Focus to the Previous Text in String - Pressing left arrow key (<-)

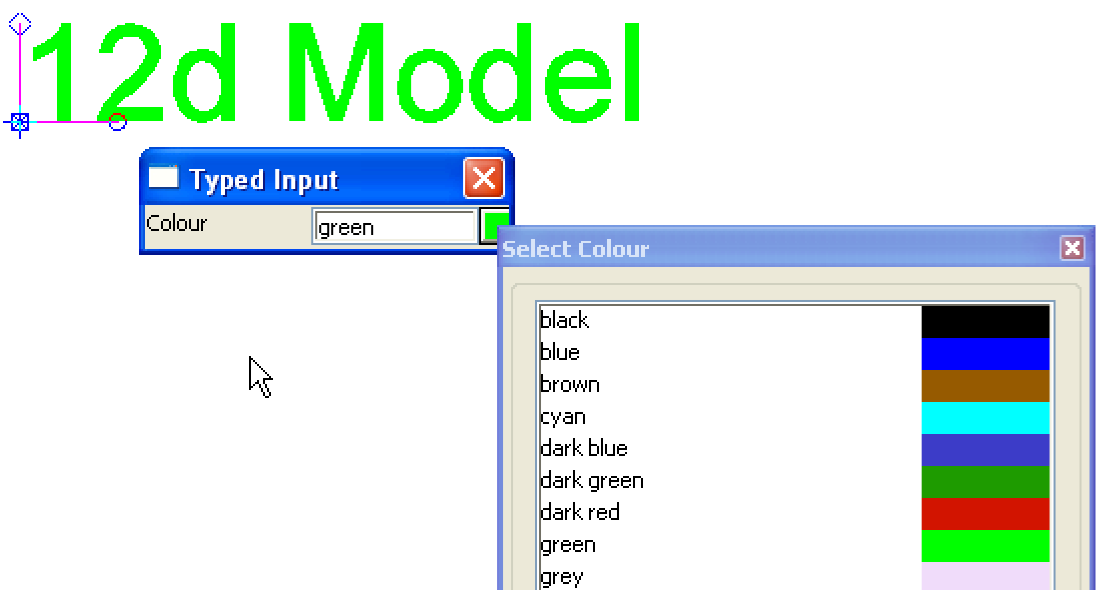
To show the other line of options, type **m** - go to Show Other Line of Options - Typing m

To toggle vertex lock on and off, type **l** - go to Toggle Lock - Typing l

To toggle autopan on and off, type **p** - go to Toggle Autopan - Typing p

Change Text Colour - Typing c

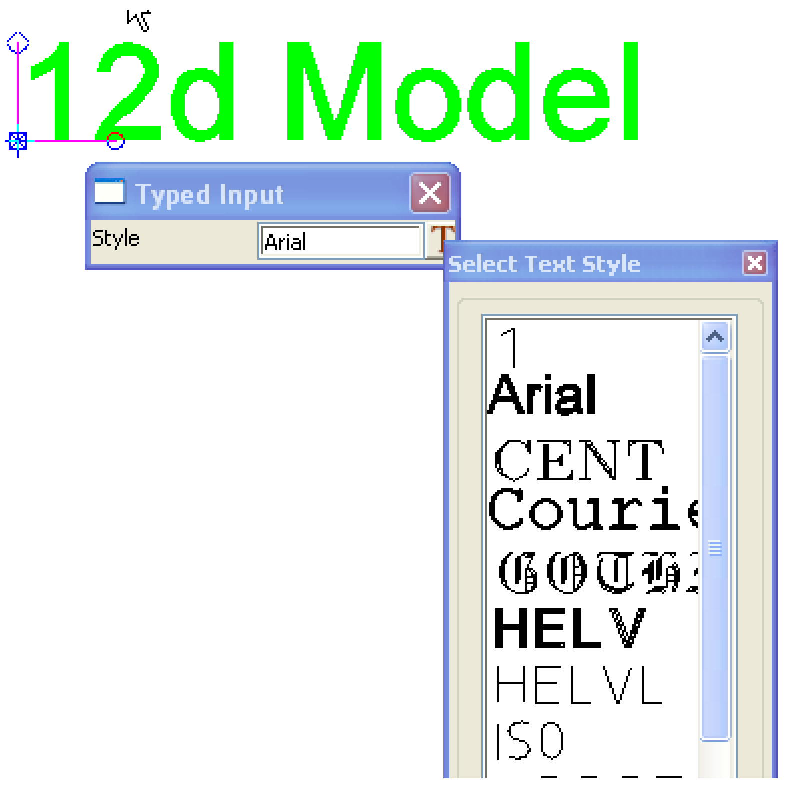
After typing **c**, a Colour **Typed input** box appears with the current text colour displayed.



The new colour for the text is typed into the Colour input box followed by the <enter> key, or by selecting a colour from the **Select Colour** box which is brought up by clicking B1 on the colour button.

Change Text Style - Typing s

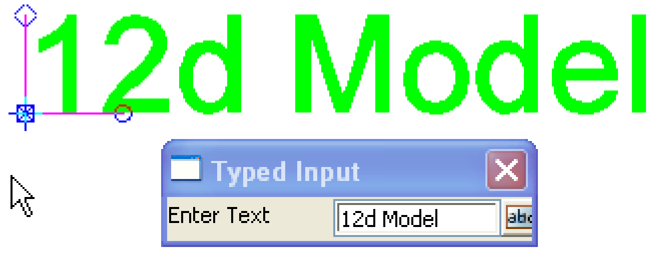
After typing **s**, a Style **Typed input** box appears with the current text style displayed.



The new style for the text is typed into the Style input box followed by the <enter> key, or by selecting a style from the **Select Text Style** box which is brought up by clicking B1 on the style button.

Change Text - Pressing space bar or Typing t

After typing **t** or pressing the space bar, an Enter Text **Typed input** box appears with the current text in it.



The new text is typed into the Enter Text input box followed by the <enter> key.

Change Text Height - Typing h

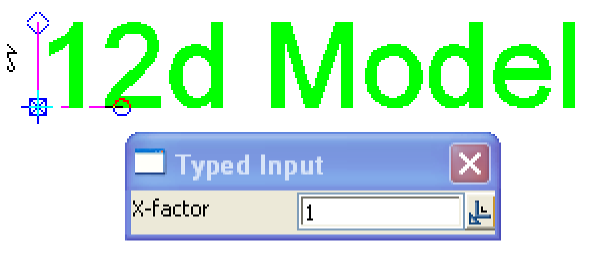
After typing **h**, a Size **Typed input** box appears with the current text height in it.

The new height is typed into the Size input box followed by the <enter> key.

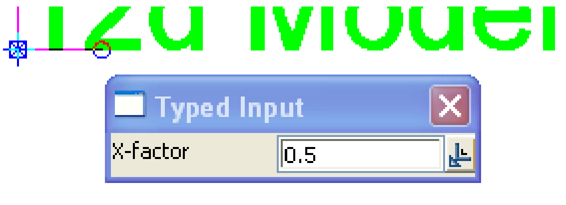
Note - Many arithmetic expressions are supported in the **Size** typed input box (*e.g.* 10+27). See the section 4.19.1 Expressions in Panel Fields.

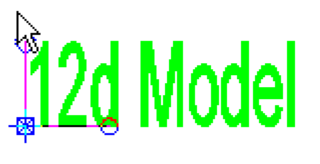
Change Text Width Factor - Typing w

After typing **w**, a X-Factor **Typed input** box appears with the current width (x-factor) in it.



The new width (x-factor) is typed into the X-factor input box followed by the <enter> key.



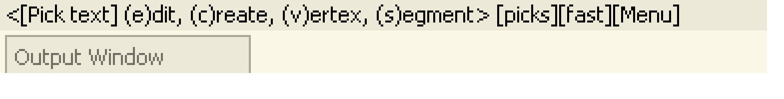


width (x-factor) = 0.5

Note - Many arithmetic expressions are supported in the **X-factor** typed input box (e.g. 0.5*2.2). See the section 4.19.1 Expressions in Panel Fields.

Create/Edit New Text - Typing n

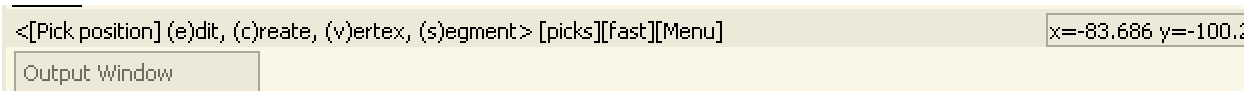
If the option was in ***Edit*** mode and **n** is typed, a new **Edit** is started (**e** mode). The ***Pick text*** message is displayed in the screen message area and another text is selected to edit.



Or a ***new*** text can be created by typing **c**, **v** or **s**. See the section 15.15.1 Create Text for more information on **c**, **v** and **s**.

If the option was in ***Create*** mode and **n** is typed, a new **Create** is started in the **c** mode. The

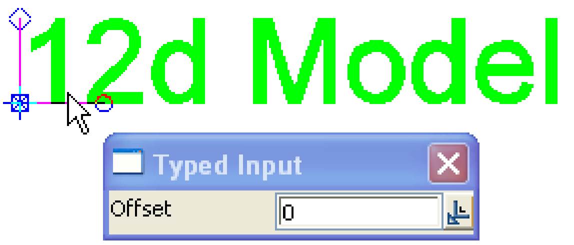
Pick position message is displayed in the screen message area and a new text is created.

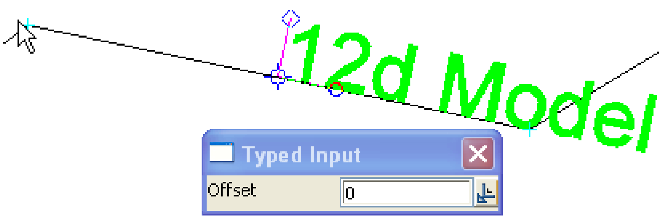


Or a ***new*** segment or vertex text can be created by typing **v** or **s** or the Edit mode can be entered by typing **e**. See the section 15.15.1 Create Text for more information on **c**, **v** and **s**.

Change Text Offset - Typing o

After typing **o**, an Offset **Typed input** box appears with the current text offset in it.



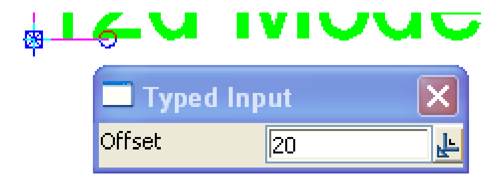


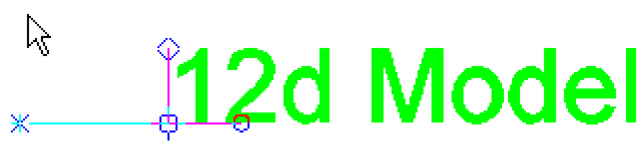
Vertex text - offset = 0

Segment text - offset = 0

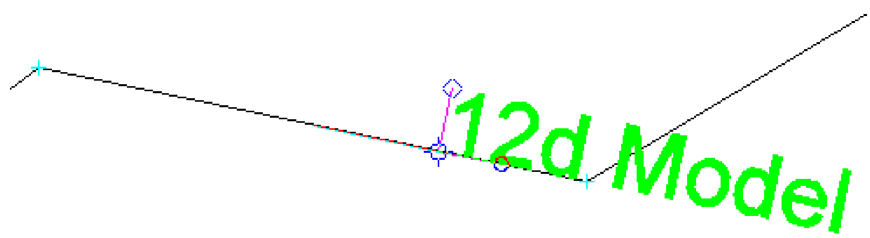
The new offset is typed into the Offset input box followed by the <enter> key.

The text justification point will be offset by the new amount





Vertex text - offset = 20



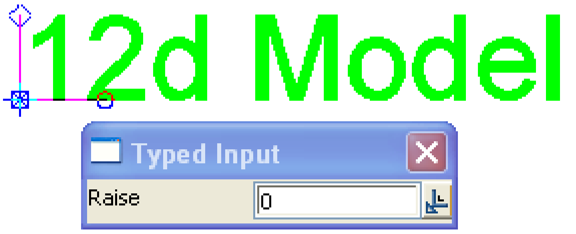
Segment text - offset = 20

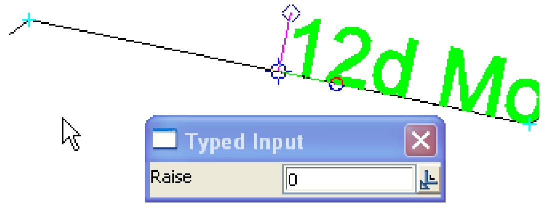
For information about text and the definition of **offset**, go to the section 4.6 Text Definitions.

Note - Many arithmetic expressions are supported in the **Offset** typed input box (*e.g.* 10+27). See the section 4.19.1 Expressions in Panel Fields.

Change Text Raise - Typing r

After typing **r**, a Raise **Typed input** box appears with the current text raise in it.

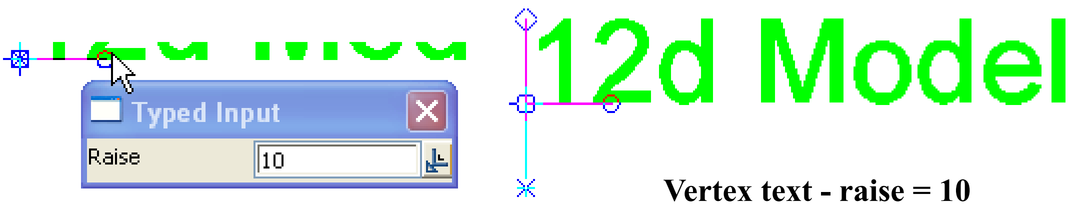




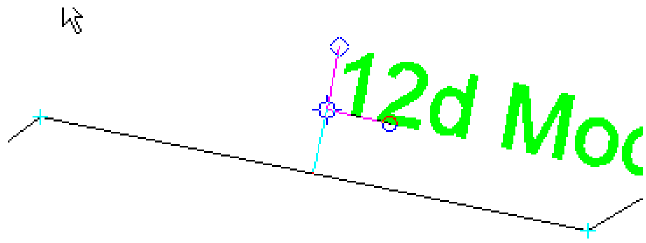
Vertex text - raise = 0 **Segment text - raise = 0**

The new raise is typed into the Raise input box followed by the <enter> key.

The text justification point will be raised by the new amount



Vertex text - raise = 10



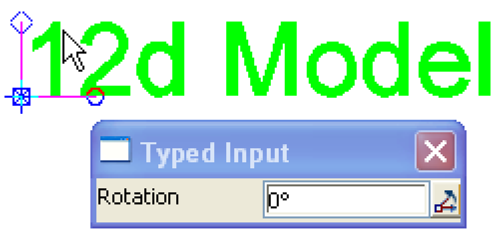
Segment text - raise = 10

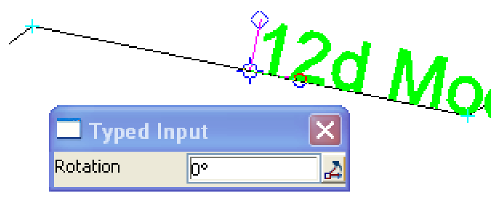
For information about text and the definition of **raise**, go to the section 4.6 Text Definitions.

Note - Many arithmetic expressions are supported in the **Raise** typed input box (*e.g.* 10+27). See the section 4.19.1 Expressions in Panel Fields.

Change Text Angle - Typing a

After typing **a**, a Rotation **Typed input** box appears with the current text angle in it.



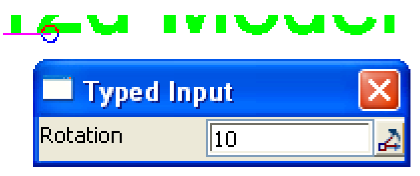


Vertex text - angle = 0 **Segment text - angle = 0**

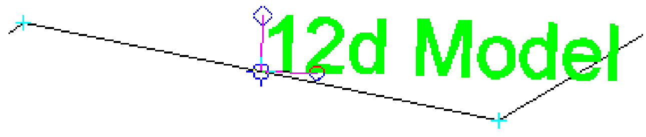
The new angle is typed into the Rotation input box followed by the <enter> key.

The direction of the text point will be rotated by the new amount.

For information about text and the definition of **angle**, go to the section 4.6 Text Definitions.



Vertex text - angle = 10

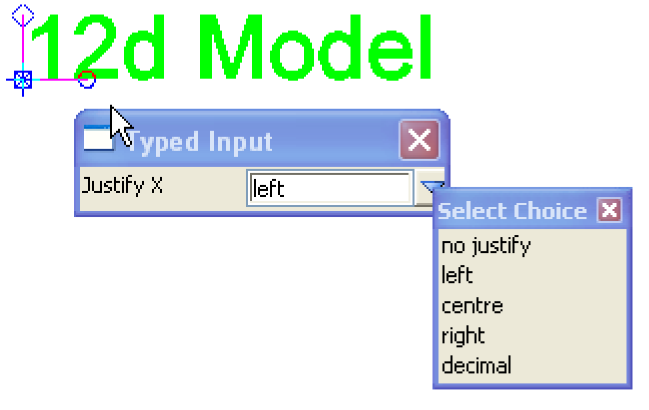


Segment text - angle = 10

Note - Some arithmetic expressions are supported in the **Rotation** typed input box (*e.g.* 10+90). See the section 4.19.1.2 Expressions in Bearing, Angle Panel Fields for what is allowed in angle/ bearing panel fields.

Change Text X Justification - Typing x

After typing **x**, a Justify X **Typed input** box appears with the current text x justification in it.

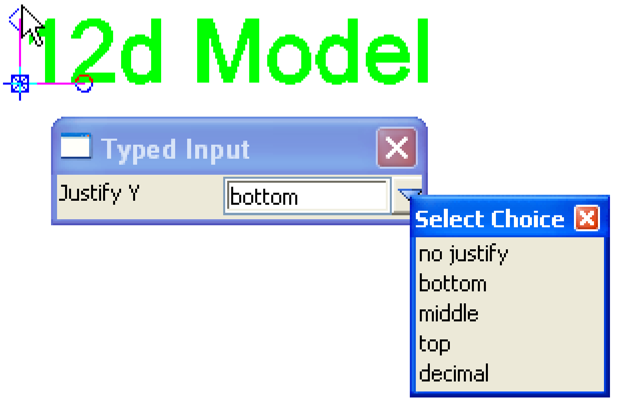


The new x justification for the text is typed into the Justify X input box followed by the <enter> key, or by selecting a justification from the **Select Choice** box which is brought up by clicking B1 on the choice button.

For information about text and the definition of **justification**, go to the section 4.6 Text Definitions.

Change Text Y Justification - Typing y

After typing **y**, a Justify Y **Typed input** box appears with the current text y justifcation in it.



The new y justification for the text is typed into the Justify Y input box followed by the <enter> key, or by selecting a justification from the **Select Choice** box which is brought up by clicking B1 on the choice button.

For information about text and the definition of **justification**, go to the section 4.6 Text Definitions.

Reset Text Parameters to Defaults - Typing d

After typing **d**, some of the text parameters are set back to their default values.

That is for a:

Text String, Vertex Text and Segment Text:

Text style is set to the Text style in the Cad Text Controlbar

Colour is reset to the colour in the Cad Controlbar

Size is set to the size in the Cad Text Controlbar

Angle is set to 0r

Offset is set to 0

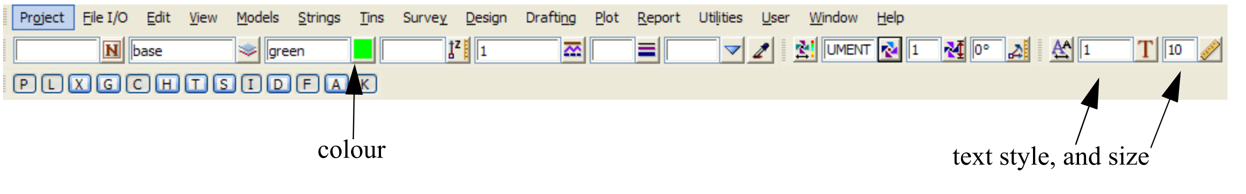
Raise is set to 0

X Factor is set to 1

X Justification is set to left

Y Justification is set to bottom

Model stays the same



For information about text definitions, go to the section 4.6 Text Definitions.

Change Focus to the Next Text in String - Pressing right arrow key (->)

For vertex text on a super string, pressing **->,** moves the focus of the edit commands to the vertex text on the next vertex of the super string. If the text is on the last vertex, pressing **->** does nothing.

For segment text on a super string, pressing **->**, moves the focus of the edit commands to the segment text on the next segment of the super string. If the text is on the last segment, pressing **->** does nothing.

For information about text definitions, go to the section 4.6 Text Definitions.

Change Focus to the Previous Text in String - Pressing left arrow key (<-)

When editing vertex text on a super string, pressing **<-,** moves the focus of the edit commands to the vertex text on the **previous** vertex of the super string. If the text is on the first vertex, pressing **<-** does nothing. If Autopan is on, the selected vertex text is made the centre of the active plan view.

When editing segment text on a super string, pressing **<-,** moves the focus of the edit commands to the segment text on the **previous** segment of the super string. If the text is on the first segment, pressing **<- d**oes nothing. If Autopan is on, the selected segment text is made the centre of the active plan view.

For information about text definitions, go to the section 4.6 Text Definitions.

Toggle Lock - Typing l

Typing **l** toggles locking the vertex on and off.

When vertex lock is on, the vertex grip can not be selected and used to move the position of the string vertex.

When the screen message area reads (l) lock (l)ock, , this means that vertex lock is off (unlocked) and typing **l** locks it.

When the screen message area reads un(l) lock un(l)ock, , this means that the vertex is locked and typing **l** unlocks it.

For information about the vertex text grip, go to the section 4.14 Text Grips.

For information about text definitions, go to the section 4.6 Text Definitions.

Toggle Autopan - Typing p

Typing **p** toggles autopan on and off.

When editing vertex text on a super string and **Autopan** is **on**, pressing **->** or **<-**, moves the focus of the edit commands to the vertex text on the next/previous vertex of the super string, and the selected vertex text is made the centre of the active plan view.

When editing segment text on a super string and **Autopan** is **on**, pressing **->** or **<-**, moves the focus of the edit commands to the vertex text on the next/previous segment of the super string, and the selected segment text is made the centre of the active plan view.

When vertex lock is on, the vertex grip can not be selected and used to move the position of the string vertex.

When the screen message area reads auto(p)an 

, this means that autopan is off and typing **p** turns autopan on.

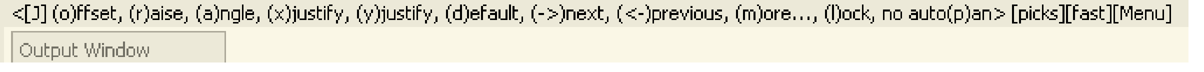
When the screen message area reads no auto(p)an 

, this means that autopan is on and typing **p** turns it off.

For information about text definitions, go to the section 4.6 Text Definitions.

Show Other Line of Options - Typing m

There are too many *Edit* options to fit onto one line of the screen message and two lines of options are required. Typing **m** toggles between the two lines of options.



Finish the Create/Edit - Typing f

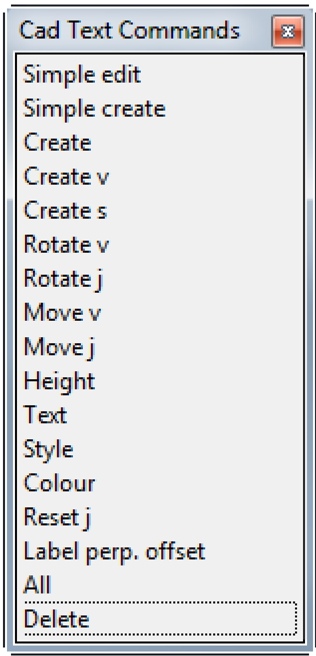
Typing **f** exits the create/edit text option.

15.15.3 CAD Text Commands

The *Cad Text Commands* do just one text modification at a time but once one selected text has been modified, another text can then be selected to have the same type of modification.

So the CAD Text Commands perform the same modification on progressively selected texts whereas the CAD Text Create and CAD Text Edit options perform any of the modifications on the one selected text.

The **Cad text commands** walk-right menu is



Menu of Individual Options to Create and Manipulate Text

earlier version of CAD Text Edit - edit text using grips
earlier version of CAD Text Create - create text string and edit using grips
create text string at a selected position
create text at a user selected super string vertex
create text at a user selected super string segment
rotate text about its vertex
rotate text about its justification points (which may be offset from the vertex
move text by its vertex
move text by its justification point (the vertex does not move)
change the height of text
change the characters of the text
change the style of text
change the colour of text
reset the text justification point so that it is on top of the vertex posn
label perpendicular offset distance
menu to make most of the above changes to one selected text
delete text

For the option *Simple edit*, go to	15.15.4 Simple Text Edit
Simple create	15.15.5 Simple Text Create
Create	15.15.6 Text Create
Create v	15.15.7 Create Text on Vertex of Super String
Centre s	15.15.8 Text Create on Segment of Super Strings
Rotate v	15.15.9 Rotate Text around Vertex
Rotate j	15.15.10 Rotate Text about the Justification Point
Move v	15.15.12 Move the Text Vertex Point
Move j	15.15.11 Move the Text Justification Point
Height	15.15.13 Height of Text
Text	15.15.14 Change Text
Style	15.15.15 Style of Text
Colour	15.15.16 Text Colour
Reset j	15.15.17 Reset Text Justification Point
Label perp. offset	15.15.18 Label Perpendicular Offset
All	15.15.19 All Text Edits
Delete	15.15.20 Text Delete

15.15.4 Simple Text Edit

Position of option on menu: CAD =>Text =>Commands =>Simple edit

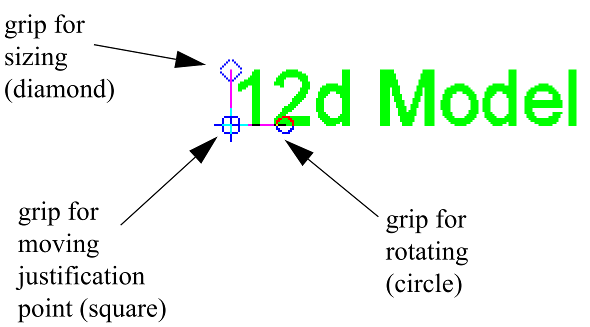
or by selection of appropriate icon from the toolbar. 

This option edits text strings, and text on super string vertices or segments, using grips to quickly define the mode of editing (sizing, rotating or moving the justification point).

On selecting **Simple edit**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

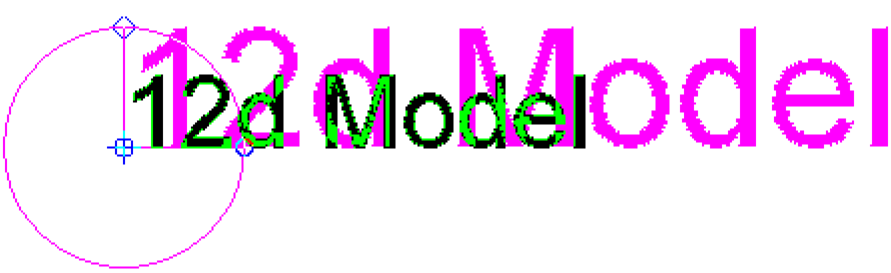
When the text to be edited is selected, grips for size, rotating and moving the justification point are displayed. The appropriate grip is then selected for quick editing.



Size Grip (diamond):

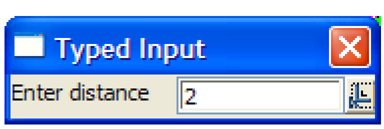
If the **Size grip** is selected, the text size is dynamically adjusted by moving the cursor. The size of the text is displayed as part of the *Enter height* prompt in the message area.

The text size is set when a position is accepted.



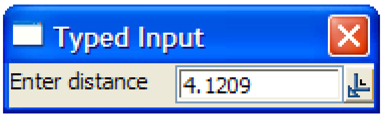
An ***exact*** text size can be entered by hitting the <space bar> or by starting to type a size. This brings up the **Enter distance** typed input box. The size is typed in and the <enter> key pressed

.



To return to dynamic sizing without entering a distance, simply select the **X** on the top of the input box. The box will disappear and dynamic sizing will resume.

If the text is being dynamically sized, the ***current*** size of the text can be displayed by pressing the **d** key which brings up the **Enter distance** typed input box with the current text size in it.



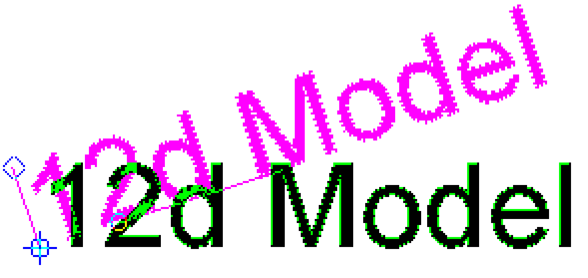
The size can be modified and the value used by pressing the <enter> key.

To return to dynamic sizing without entering a distance, simply select the **X** on the top of the input box.

Rotate Grip (circle):

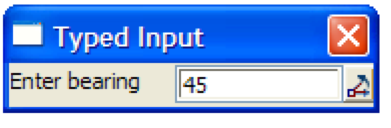
If the **Rotate grip** is selected, the text bearing is dynamically adjusted by moving the cursor. The bearing of the text is displayed as part of the *Enter bearing* prompt in the message area.

The text bearing is set when a position is accepted.



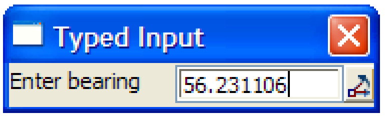
An ***exact*** bearing can be entered by hitting the <space bar> or by starting to type a bearing. This brings up the **Enter bearing** typed input box. The *bearing* is typed in and the <enter> key pressed

.



To return to dynamic rotating without entering a bearing, simply select the **X** on the top of the input box. The box will disappear and dynamic sizing will resume.

If the text is being dynamically rotated, the ***current*** bearing of the text can be displayed by pressing the **d** key which brings up the **Enter bearing** typed input box with the *current* text bearing in it.



The *bearing* can be modified and the value used by pressing the <enter> key.

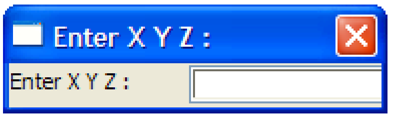
To return to dynamic rotation without entering a bearing, simply select the **X** on the top of the input box.

Justification Grip (square):

If the **Justification grip** is selected, the text justification point is dynamically adjusted by moving the cursor. The (x,y) position of the text is displayed in the message area.

The text justification point is set when a position is accepted.

An ***exact*** coordinate can be entered by hitting the <space bar> or by starting to type a coordinate. This brings up the **Enter X Y Z** typed input box. The *x and y coordinates* are typed in, separated by a space, and the <enter> key pressed.



To return to dynamic moving without entering a coordinate, simply select the **X** on the top of the input box. The box will disappear and dynamic moving will resume.

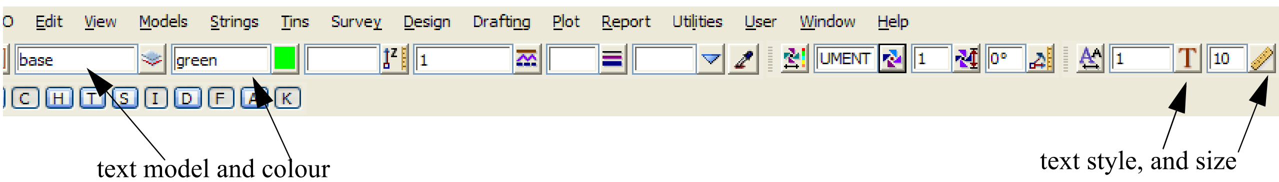
15.15.5 Simple Text Create

Position of option on menu: **CAD =>Text =>Simple create**

or by selection of appropriate icon from the toolbar. 

This option creates text strings and then displays grips to quickly allow editing of text size, rotation and justification point.

The text style and height are taken from the **Text controlbar** and the model and colour from the **CAD controlbar** (see 15.1 Controlbars,15.1.2 Symbol Controlbar and 15.1.1 CAD Controlbar).



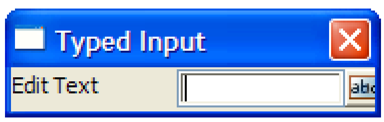
The text will be placed with the vertex and justification point at the selected position.

After defining one piece of text, the option restarts so that another of text can be defined. Picking Cancel from the Pick Ops menu, hitting <esc> or selecting another option terminates the option.

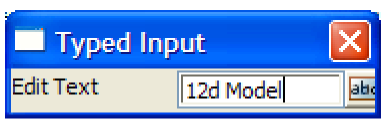
On selecting **Simple create**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

An insertion point for the text is selected and accepted and an **Enter text** typed input box is then displayed on the screen.

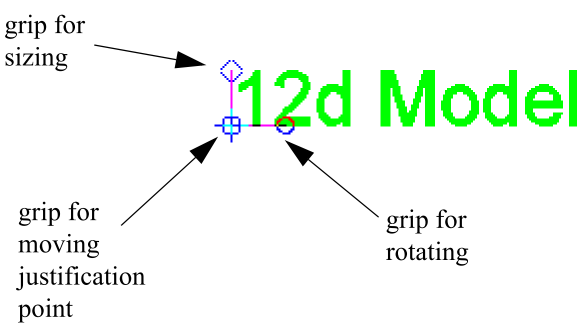


The text is typed into the box and the <enter> key pressed.



The text will be placed with the vertex and justification point at the selected position and using values such as colour, model, text height and text style given in the **CAD controlbar** and **Text controlbar** (See 15.1 Controlbars)

When the text is displayed, grips for quick editing of size, rotation and moving of the justification point are also shown. The appropriate grip can then selected for quick editing. For details on using the grips, go to the section 15.15.5 Simple Text Create.



15.15.6 Text Create

Position of option on menu: **CAD =>Text =>Create**

or by selection of appropriate icon from the toolbar.

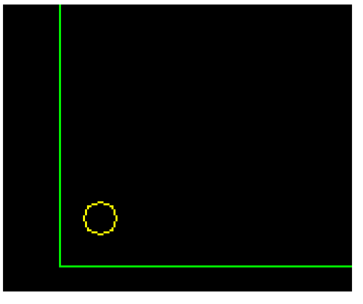
This option creates text strings at a selected insertion point (a cursor snap is valid).

After defining one piece of text, the option restarts so that another piece of text can be defined. Picking Cancel from the Pick Ops menu, hitting <esc> or selecting another option terminates the option.

On selecting **Create**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

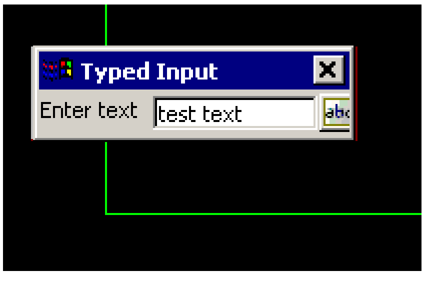
STEP 1:

An insertion point is selected and accepted. The text will be placed relative to the insertion point given other parameters such as justification, as defined in the textstyle data panel



STEP 2:

The actual text is entered into the text input box followed by the enter key.



STEP 3:

The text is created using the specified settings.

15.15.7 Create Text on Vertex of Super String

Position of option on menu: **CAD =>Text =>Create v**

or by selection of appropriate icon from the toolbar.

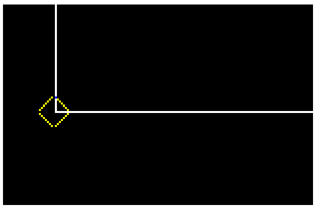
This option creates text at a vertex of a super string. One piece of text is allowed per super string vertex, so if this option is used on existing vertex text, the existing value is displayed in the text input box.

After defining one piece of text, the option restarts so that another piece of text can be defined. Picking Cancel from the Pick Ops menu, hitting <esc> or selecting another option terminates the option.

On selecting **Create v**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

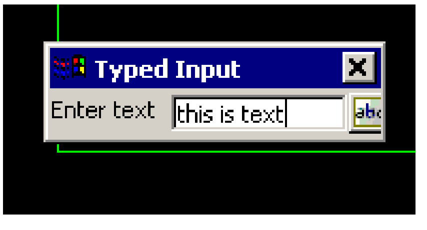
STEP 1:

An insertion point is selected and accepted. The point snap is forced on so that a vertex can be selected. The text will be placed relative to the insertion point given other parameters such as justification, as defined in the textstyle data panel



STEP 2:

The actual text is entered into the text input box followed by the Enter key.



STEP 3:

The text is created using the specified settings.

Picking Cancel from the Pick Ops menu, hitting <esc> or selecting another option terminates the option.

15.15.8 Text Create on Segment of Super Strings

Position of option on menu: **CAD =>Text =>Create s**

or by selection of appropriate icon from the toolbar.

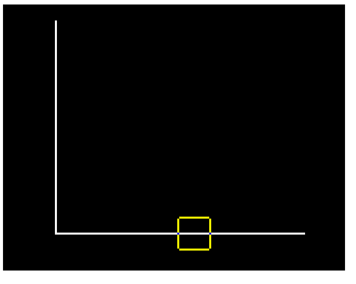
This option creates text at the mid point of a selected super string segment. One piece of text is allowed per super string segment, so if this option is used on existing segment text, the existing value is displayed in the text input box.

After defining one piece of text, the option restarts so that another piece of text can be defined. Picking Cancel from the Pick Ops menu, hitting <Esc> or selecting another option terminates the option.

On selecting **Create s**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

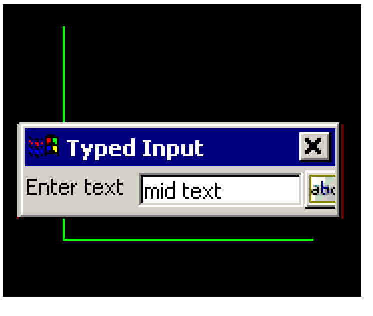
STEP 1:

A super string segment is selected and accepted. The line snap is forced on so that a segment can be selected. The text will be placed relative to the insertion point which is the midpoint of the segment given other parameters such as justification, as defined in the textstyle data panel



STEP 2:

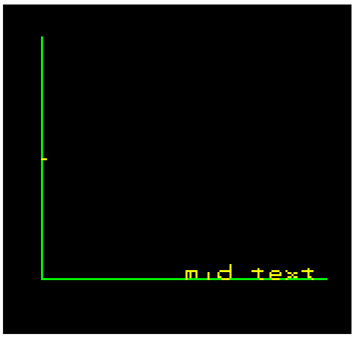
The actual text is entered into the text input box followed by the Enter key.



STEP 3:

The text is created using the specified settings.

Picking Cancel from the Pick Ops menu, hitting <Esc> or selecting another option terminates the option.


mid text

15.15.9 Rotate Text around Vertex

Position of option on menu: **CAD =>Text =>Rotate v**

or by selection of appropriate icon from the toolbar.

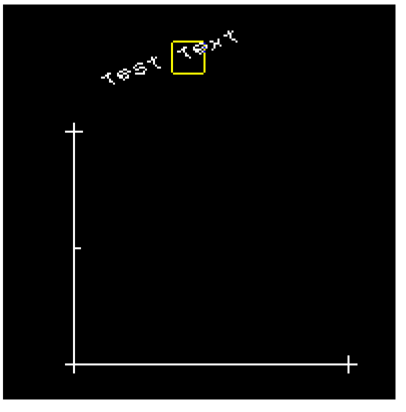
This option rotates text about its vertex. This applies to text strings, super string segment or vertex text and 4d string vertex text. For super string segment text, the vertex is taken to be the mid point of the segment.

After defining one piece of text, the option restarts so that another piece of text can be defined. Picking **Cancel** from the **Pick Ops** menu, hitting <Esc> or selecting another option terminates the option.

On selecting **Rotate v**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

A text string to be rotated is selected and accepted.



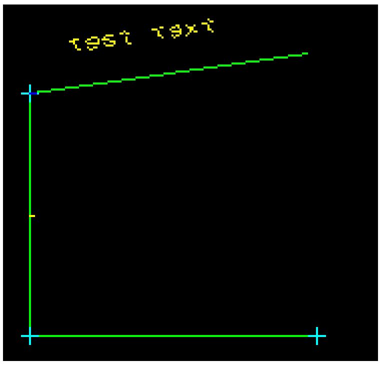
STEP 2:

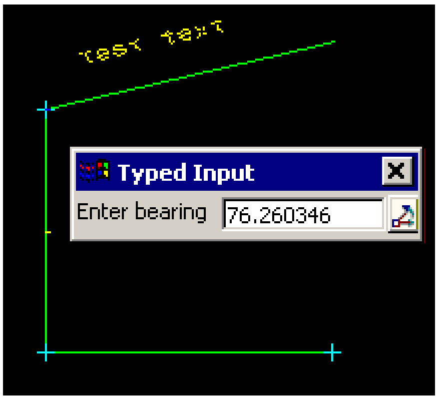
The rotation bearing (the final bearing of the text) is specified by selection and acceptance of a point with the mouse or by entry into an input box followed by the enter key. To bring up the input box start typing or press the space bar. The value is entered into the input box followed by the enter key.

Note: The typed bearing is not relative but absolute. i.e. the value given will not rotate the existing text by that amount, rather it will reposition the text at that bearing.

The line drawn represents the bearing value and changes with movement of the mouse. If the user wants to see what the current value of the bearing is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to specify the bearing or the input box can be closed and the rubber banding (graphically changing) of the angle continued.

Note: **The Page up and page down keys can be used when the input angle box comes up to add or subtract intervals of 90 degrees.**



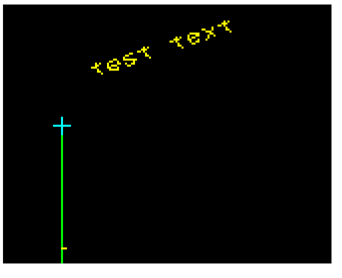


This option also allows the definition of the bearing by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next. The example shown below is the perpendicular case.

STEP 3:

The text is rotated to the specified bearing.

Picking Cancel from the Pick Ops menu, hitting <Esc> or selecting another option terminates the option.



15.15.10 Rotate Text about the Justification Point

Position of option on menu: **CAD =>Text =>Rotate j**

or by selection of appropriate icon from the toolbar. 

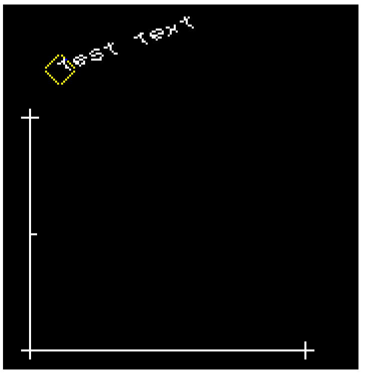
This option rotates text about its justification point. Note that the vertex point and the justification point do not coincide when either the raise or offset for the text is not zero.

Note: The typed bearing is not relative but absolute. i.e. the value given will not rotate the existing text by that amount, rather it will reposition the text at that bearing.

On selecting **Rotate j**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

The text to be rotated is selected and accepted.

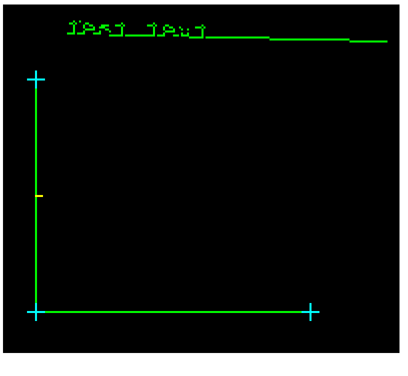


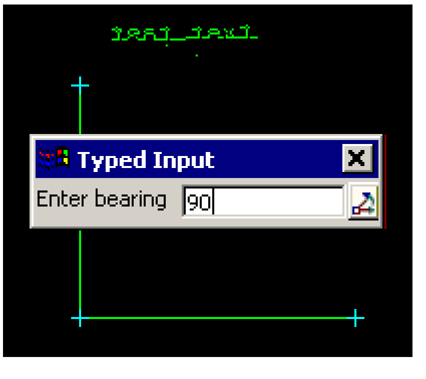
STEP 2:

The rotation bearing (the final bearing of the text) is specified by selection and acceptance of a point with the mouse or by entry into an angle box followed by the enter key. To bring up the angle box start typing or press the space bar. The value is entered into the input box followed by the enter key.

The line drawn represents the bearing value and changes with movement of the mouse. If the user wants to see what the current value of the bearing is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to specify the bearing or the input box can be closed and the rubber banding (graphically changing) of the bearing continued.

Note: **The Page up and page down keys can be used when the bearing input box comes up to add or subtract intervals of 90 degrees.**



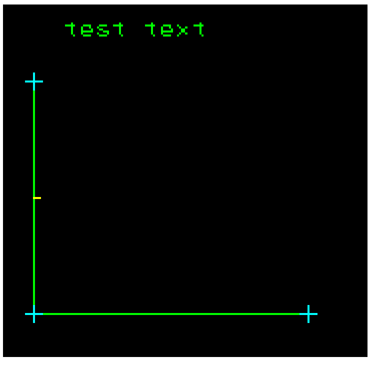


This option also allows the definition of the bearing by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next. The example shown below is the perpendicular case.

STEP 3:

The text is rotated to the specified bearing.

Picking Cancel from the Pick Ops menu, hitting <Esc> or selecting another option terminates the option.



15.15.11 Move the Text Justification Point

Position of option on menu: **CAD =>Text =>Move j**

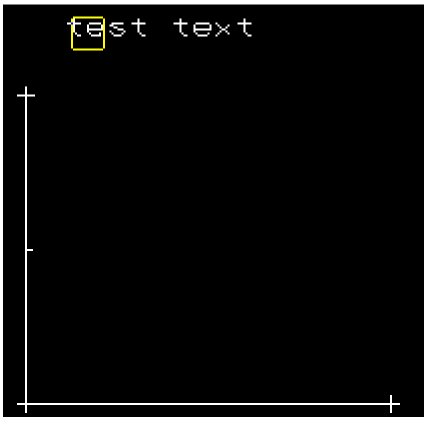
or by selection of appropriate icon from the toolbar.

This option moves the text justification point, and hence the text with it.

On selecting **Move j**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

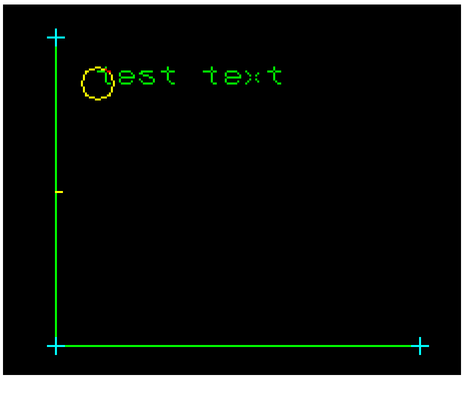
STEP 1:

The text to be moved is selected and accepted.



STEP 2:

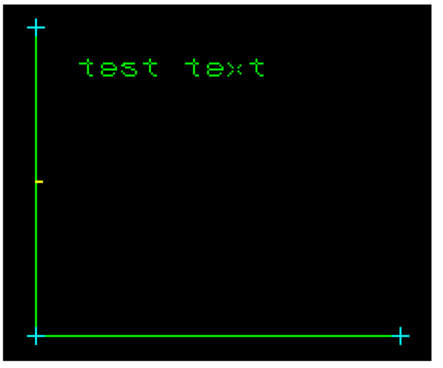
The new position for the text is selected and accepted.



STEP 3:

The text is moved to the new position.

Picking Cancel from the Pick Ops menu, hitting <Esc> or selecting another option terminates the option.


test text

15.15.12 Move the Text Vertex Point

Position of option on menu: **CAD =>Text =>Move v**

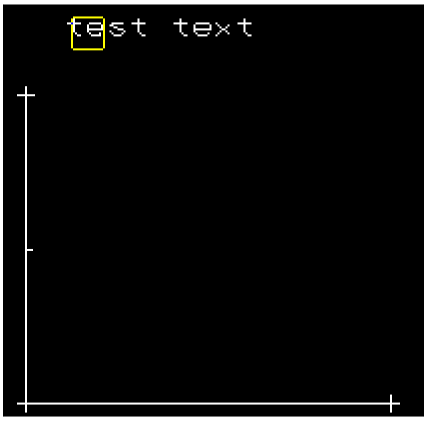
or by selection of appropriate icon from the toolbar. 

This option moves the vertex that vertex text is attached to, and hence moves the text with it. The option works for text strings, vertex text for super strings and 4d string but will ***not*** work for segment text where there is no actual vertex at the mid segment position where the segment text is initially placed.

On selecting **Move v**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

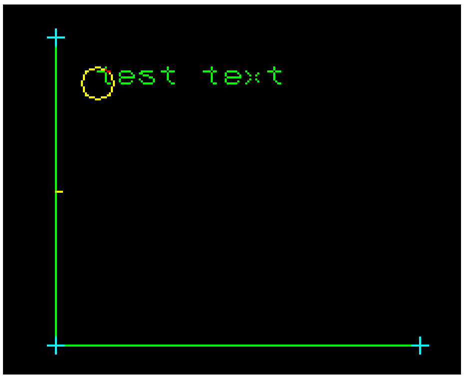
STEP 1:

The vertex text to be moved is selected and accepted.



STEP 2:

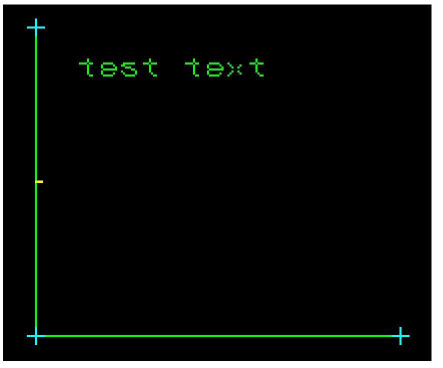
The new position for the vertex and the text is selected and accepted.



STEP 3:

The vertex and the vertex text is moved to the new position.

Picking **Cancel** from the **Pick Ops** menu, hitting <Esc> or selecting another option terminates the option.


test text

15.15.13 Height of Text

Position of option on menu: **CAD =>Text =>Height**

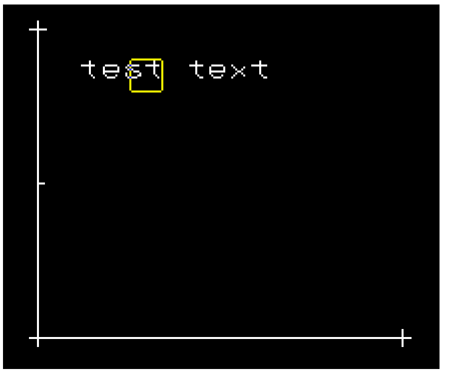
or by selection of appropriate icon from the toolbar. 

This option changes the height of text.

On selecting **Height**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

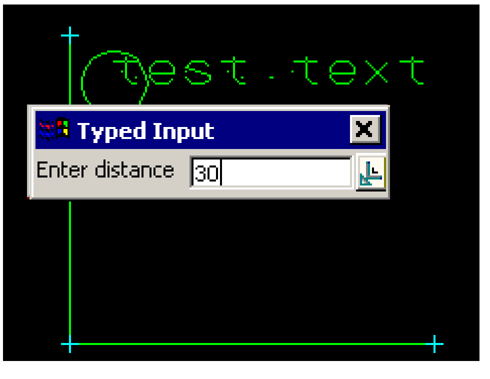
The text of which the height is to be changed is selected and accepted.



STEP 2:

A height for the text is given either by selection with the mouse or by typing a value. To type a value, simply start typing and the input box for the height will appear. Alternatively you can press the space bar to bring up the input box. The value is entered into the input box followed by the enter key.

The circle drawn represents the height value and changes with movement of the mouse. If the user wants to see what the current value of the height is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the point or the input box can be closed and the rubber banding (graphically changing) of the circle continued.

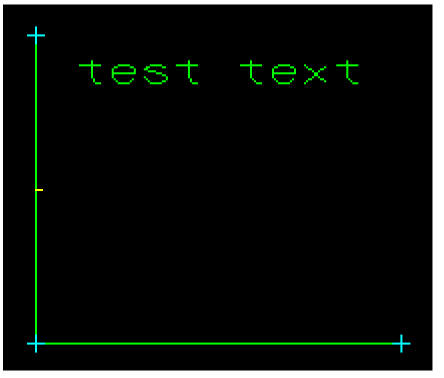


This option also allows the definition of the distance by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next.

STEP 3:

The text height is changed.

Picking Cancel from the Pick Ops menu, hitting <Esc> or selecting another option terminates the option.



15.15.14 Change Text

Position of option on menu: CAD =>Text =>Text

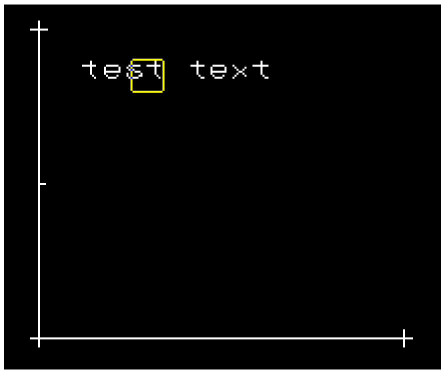
or by selection of appropriate icon from the toolbar.

This option edits the characters of selected text.

On selecting **Text**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

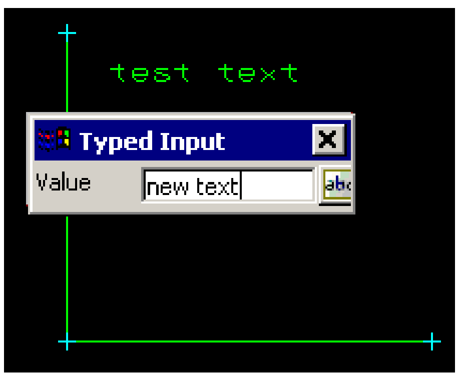
STEP 1:

The text to be changed is selected and accepted.



STEP 2:

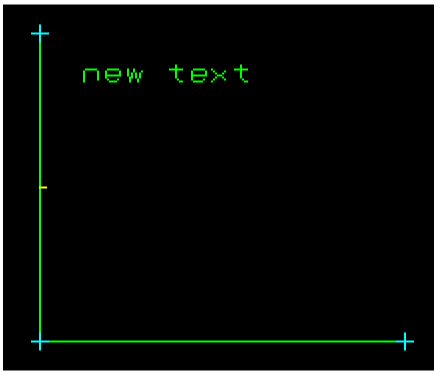
The new text is entered into the text input box followed by the Enter key.



STEP 3:

The text is changed.

Picking Cancel from the Pick Ops menu, hitting <Esc> or selecting another option terminates the option.


new text

15.15.15 Style of Text

Position of option on menu: CAD =>Text =>Style

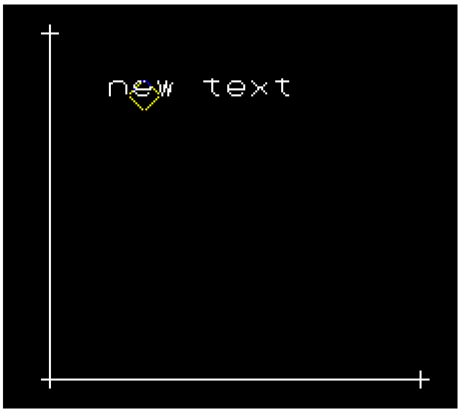
or by selection of appropriate icon from the toolbar.

This option changes the textstyle of selected text.

On selecting **Style**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

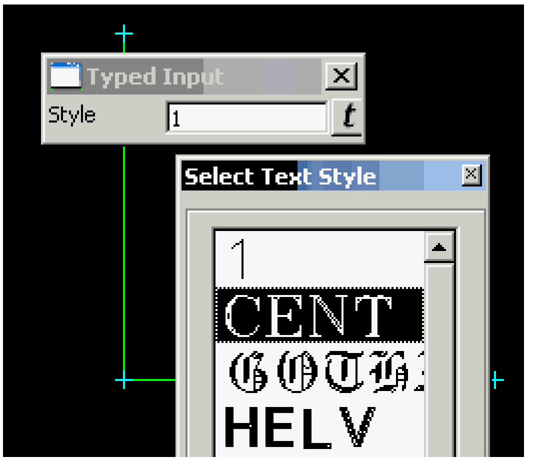
STEP 1:

The text to be changed is selected and accepted.



STEP 2:

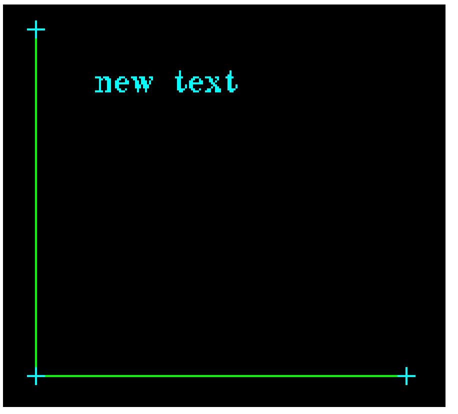
The new textstyle is selected from the style pop-up list.



STEP 3:

The textstyle is changed.

Picking Cancel from the Pick Ops menu, hitting <Esc> or selecting another option terminates the option.


new text

15.15.16 Text Colour

Position of option on menu: **CAD =>Text =>Colour**

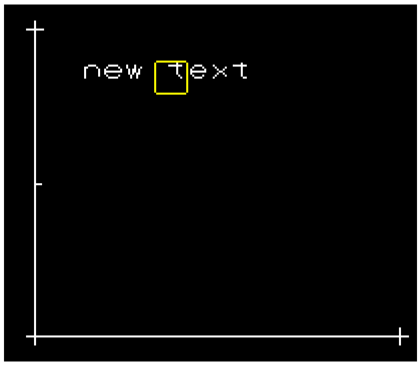
or by selection of appropriate icon from the toolbar.

This option changes the colour of selected text.

On selecting **Colour**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

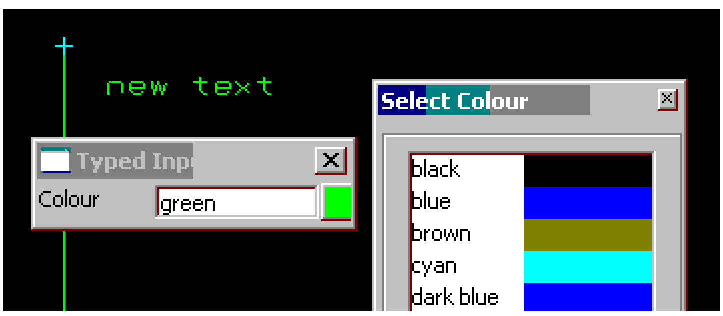
STEP 1:

The text to be changed is selected and accepted.



STEP 2:

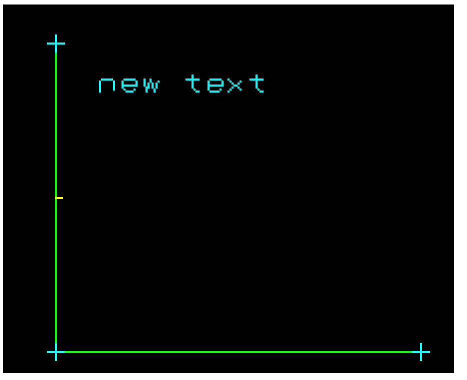
The new colour for the text is entered into the text input box or by selecting the colour square on the input box to bring up the select colour choice box followed by the enter key.



STEP 3:

The text colour is changed.

Picking Cancel from the Pick Ops menu, hitting <Esc> or selecting another option terminates the option.


new text

15.15.17 Reset Text Justification Point

Position of option on menu: **CAD =>Text =>Reset j**

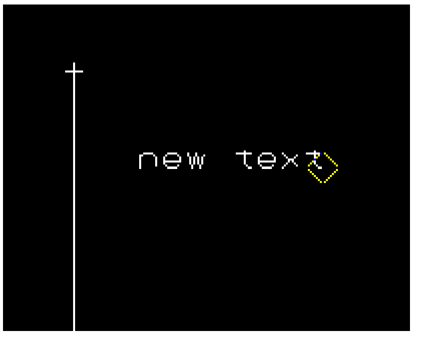
or by selection of appropriate icon from the toolbar. 

This option resets the text justification point back to being the same as the vertex for vertex text, and to the segment mid point for segment text. That is, the offset and raise for the selected text are both set to zero.

On selecting **Reset j**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

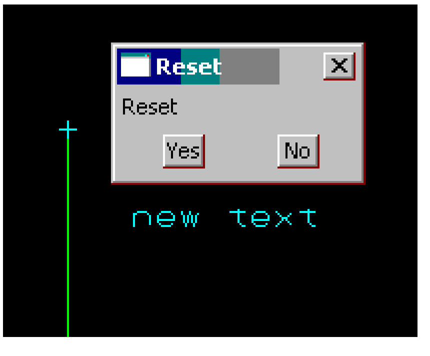
STEP 1:

The text to be changed is selected and accepted.



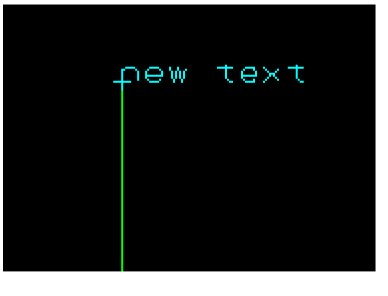
STEP 2:

A warning message is shown giving the user the option of resetting the justification point or not.



STEP 3:

The text is reset if accepted.



15.15.18 Label Perpendicular Offset

Position of option on menu: **CAD =>Text =>Label perp. Offset**

or by selection of appropriate icon from the toolbar. 

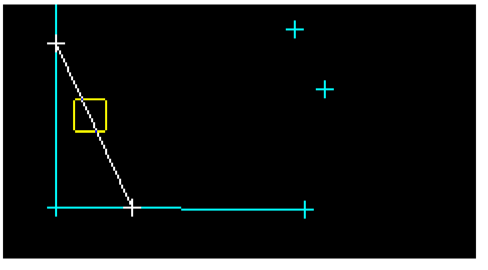
This option creates a perpendicular line from a selected segment to a selected position, and creates text with the value of the perpendicular distance between the segment and the selected position, on the line.

On selecting **Label perp. Offset**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

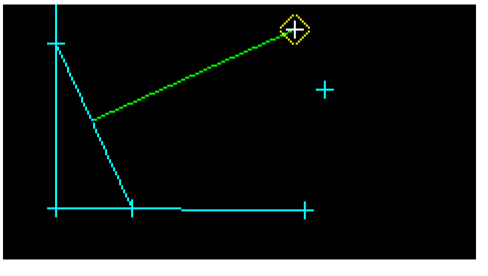
A string is selected and accepted.

After the string is accepted a perpendicular line will be displayed 'rubber banding' to the various solutions according to the position of the cursor. This will continue until a 2nd point is selected and accepted.



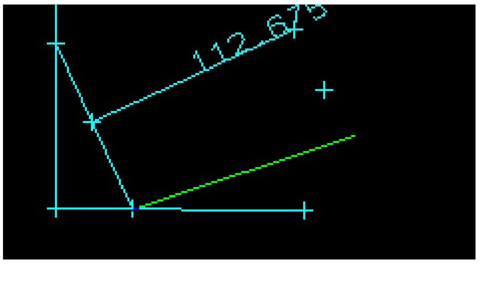
STEP 2:

Select a 2nd point and accept it.



STEP 3:

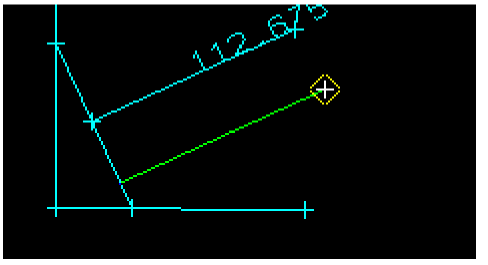
A perpendicular line is constructed between the accepted string and point. The length value of this line will be displayed in text. This text can be modified or changed using the other CAD Text options.



STEP 4:

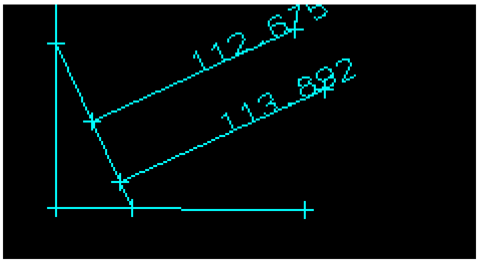
After the labelled perpendicular line is constructed, another perpendicular *rubber banding* line will be displayed. This will continue until the option is terminated (via the Esc key).

To create another labelled perpendicular line, select another point and accept it.



STEP 5:

Another labelled perpendicular line is constructed. Another perpendicular *rubber banding* line will be displayed. This will continue until the option is terminated (via the Esc key).



15.15.19 All Text Edits

Position of option on menu: **CAD =>Text =>All**

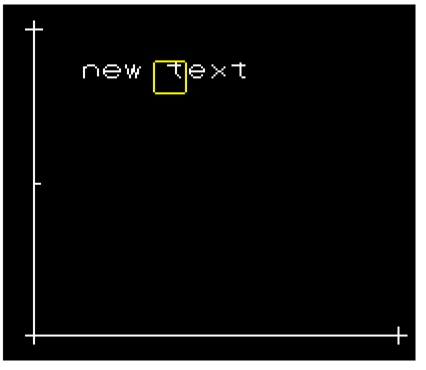
or by selection of appropriate icon from the toolbar. 

This option positions text using a menu made up of most of text edits already described.

On selecting **All**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

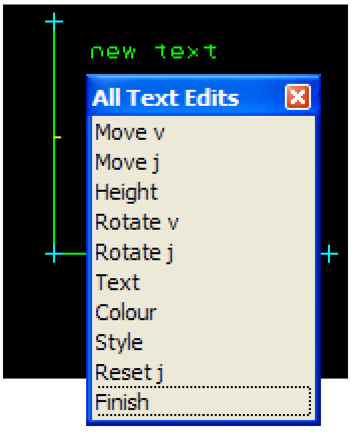
STEP 1:

The text to be changed is selected and accepted.



STEP 2:

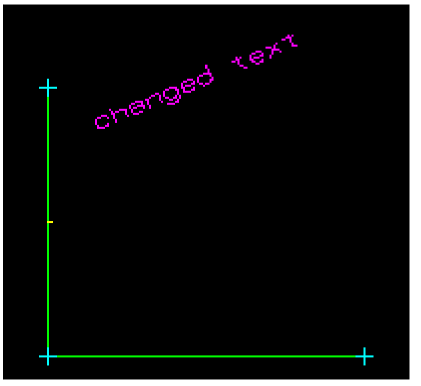
After the selection and acceptance of a piece of text, the positioning menu is shown. This also allows various other text editing functions such as colour, height and text.



STEP 3:

The user can select the appropriate option from the menu and make the changes. The menu remains active allowing a number of operations to be made is succession.

Picking Cancel from the Pick Ops menu, hitting <Esc> or selecting another option terminates the option.


changed text

15.15.20 Text Delete

Position of option on menu: CAD =>Text =>Delete

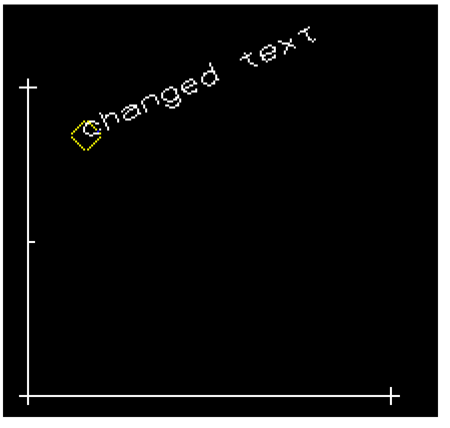
or by selection of appropriate icon from the toolbar.

This option deletes selected text.

On selecting **Delete**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

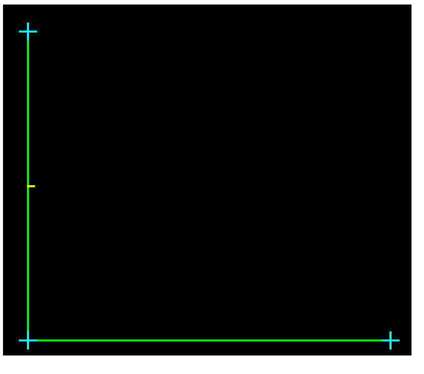
The text to be deleted is selected and accepted.



STEP 2:

The selected text is deleted.

Picking **Cancel** from the **Pick Ops** menu, hitting <Esc> or selecting another option terminates the option.



15.16 CAD Symbol

Position of option on menu: **CAD =>Symbol**

Position of option on menu: **Various toolbars**

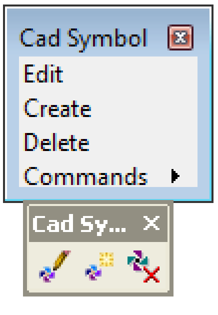
12d Model symbols are world styles at a vertex of a super string. The symbol is attached to a vertex (these are displayed when ***Vertices*** are toggle on in a plan view) and has a justification point, a rotation, a delta x and a delta y. The vertex and justification point only coincide if the delta x and delta y values are both zero.

In some symbol options, the symbol *type*, *size* and *angle* are defined by the **Symbol Controlbar** and *model* and *colour* from the **CAD Controlbar** (for more information, go to the section 15.1.2 Symbol Controlbar).

The *symbols* are defined in the ***symbols.4d*** file.

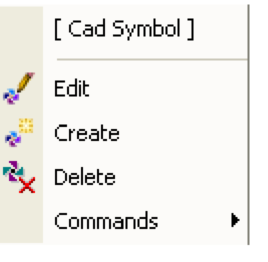
For more information on the control bars, go to the sections 15.1.2 Symbol Controlbar and 15.1.1 CAD Controlbar**.**

The **Cad Symbol** walk-right menu is



Menu of Options to Create and Manipulate Symbols

edit symbols using grips
create symbols and edit using grips
delete a symbol



For the option *Edit*, go to	15.16.2 Edit Symbol
Create	15.16.1 Create Symbol
Delete	15.16.3 Symbol Delete
Commands	15.16.3.1 Cad Symbol Commands

15.16.1 Create Symbol

Position of option on menu: **CAD =>Symbol =>Create**

or by selection of appropriate icon from the toolbar. 

For information about Symbol and its definitions, go to the section 4.7 Symbol Definitions.

12d Model has two CAD symbol options **Create** and **Edit**. However both of these run the same option which starts up in different modes - **Create** is in the mode to create symbols on a super string vertex and **Edit** which edits existing symbols.

Once either CAD symbol option is started it is possible to change between the Create and Edit modes at any time.

The ***Create*** option creates a new super string with a symbol on the vertex, or create a symbol on an existing super string vertex which does not already have a symbol, and then displays a list key commands in the screen message area, and grips on screen, to quickly allow editing of the symbol type, size, colour, rotation, angle, x and y offset etc.

After selecting the option, the ***Pick position*** message is displayed in the screen message area.

<[Pick position] (e)dit, (c)reate, (v)ertex> [picks][fast][Menu]

To go from Create to Edit mode, type **e** - go to To Go from Create to Edit Mode - Typing e

To create a symbol and a new super string, type **c** - go to Creating a Symbol - Typing c

To create vertex symbol, type **v** - go to Creating Text at a Super String Vertex- Typing v

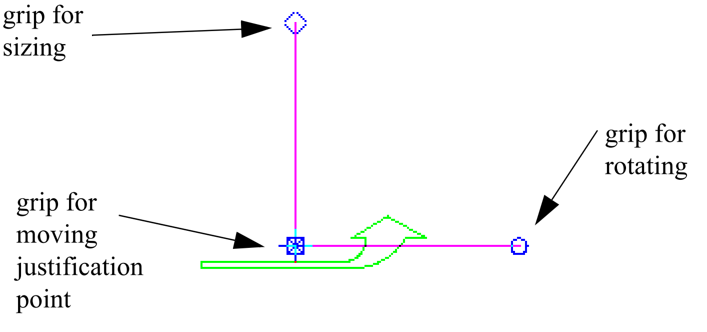
Creating a Symbol - Typing c

After typing **c**, the ***Pick position*** message is again displayed in the screen message area and the position of the symbol is then selected with the cursor.

<[Pick position] (e)dit, (c)reate, (v)ertex> [picks][fast][Menu]

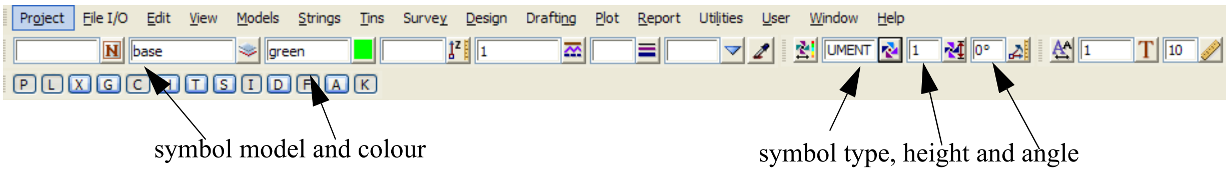
When the position is accepted a new one vertex super string is created with the symbol on it.

The super string vertex and symbol justification point is placed at that position. The x and y offset both zero.



When the symbol is displayed, grips for quick editing of height, rotation and moving of the justification point are also shown. The appropriate grip can then selected for quick editing. For details on using the grips, go to the section 4.15 Symbol Grips.

For the new symbol, the symbol name, height and angle are taken from the ***Symbol controlbar*** and the model and colour from the ***CAD controlbar*** (see 15.1 Controlbars,15.1.2 Symbol Controlbar and 15.1.1 CAD Controlbar).



After the symbol is created, a list of typed options is then displayed in the screen message area.

```
<[J] Pick grip point or (v)ertex, (c)olour, (s)ymbol, (h)eight, (n)ew, (f)inish, (m)ore...> [picks][fast][Menu]
```

and if **m** is typed, the rest of the options are shown

```
<[J] offset (x), offset (y), rot(a)tion, (d)efault, (m)ore..., (l)ock, no auto(p)an> [picks][fast][Menu]
```

These option are the same as when editing existing symbols, and will be described in detail in the section 15.16.2 Edit Symbol.

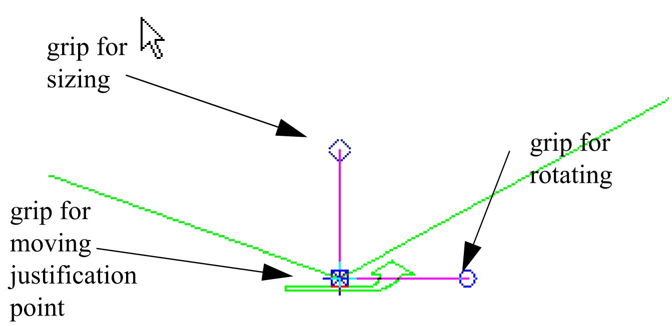
c is the default if nothing is typed.

Creating a Symbol at a Super String Vertex- Typing v

After typing **v**, the ***Pick grip point*** message is displayed in the screen message area and the vertex of the super string to place the symbol on is then selected with the cursor.

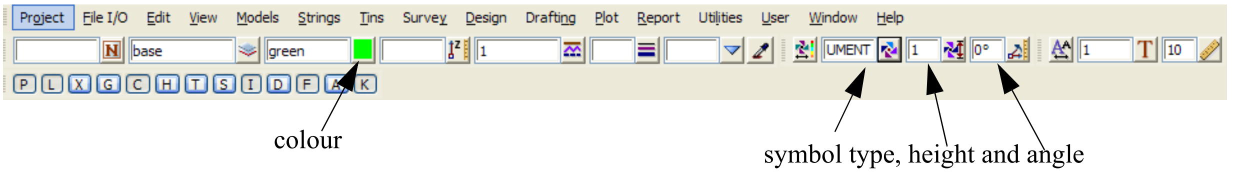
```
<[V] Pick grip point or (j)ustification, (c)olour, (s)ymbol, (h)eight, (n)ew, (f)inish, (m)ore...> [picks][fast][Menu]
```

The symbol and symbol justification point are placed at that vertex. The x and y offset both zero.



When the symbol is displayed, grips for quick editing of size, rotation, moving of the justification point and string vertex are also shown. The appropriate grip can then selected for quick editing. For details on using the grips, go to the section 4.15 Symbol Grips.

For the new symbol, the symbol name, height and angle are taken from the ***Symbol controlbar*** and colour from the ***CAD controlbar*** (see 15.1 Controlbars,15.1.2 Symbol Controlbar and 15.1.1 CAD Controlbar). The created symbol will be part of the super string so no model is required.



After the vertex symbol is created, a list of typed options is then displayed in the screen message area.

<[V] Pick grip point or (j)ustification, (c)olour, (s)ymbol, (h)eight, (n)ew, (f)inish, (m)ore...> [picks][fast][Menu]

and if **m** is typed, the rest of the options are shown

<[V] offset (x), offset (y), rot(a)tion, (d)efault, (m)ore..., (l)ock, no auto(p)an, <-previous, ->next> [picks][fast][Menu]

These option are the same as when editing existing symbols, and will be described in detail in the section 15.16.2 Edit Symbol.

To Go from Create to Edit Mode - Typing e

After typing **e**, the ***Pick Symbol*** message is displayed in the screen message area.

<[Pick symbol] (e)dit, (c)reate, (v)ertex> [picks][fast][Menu]

This is the same as when selecting the Edit option from the CAD Symbol toolbar and will be described in detail in the section 15.16.2 Edit Symbol.

15.16.2 Edit Symbol

Position of option on menu: **CAD =>Symbol =>Edit**

or by selection of appropriate icon from the toolbar. 

For information about Symbol and its definitions, go to the section 4.7 Symbol Definitions.

12d Model has two CAD symbol options **Create** and **Edit**. However both of these run the same option which starts up in different modes - **Create** is in the mode to create a new symbol, or a symbol on a super string vertex and **Edit** which edits existing symbols.

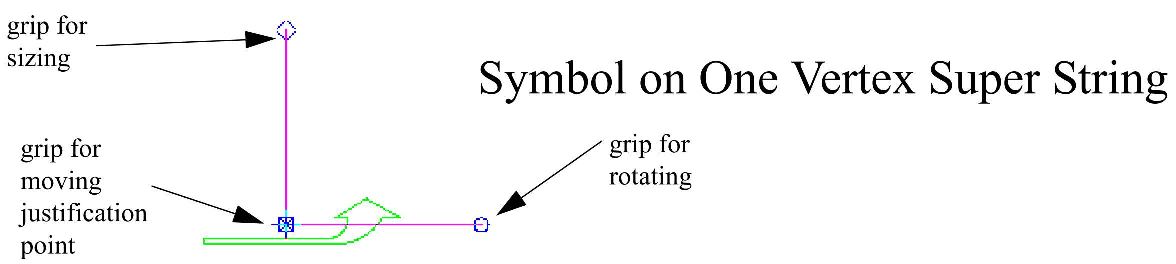
Once either CAD symbol option is started it is possible to change between the Create and Edit modes at any time.

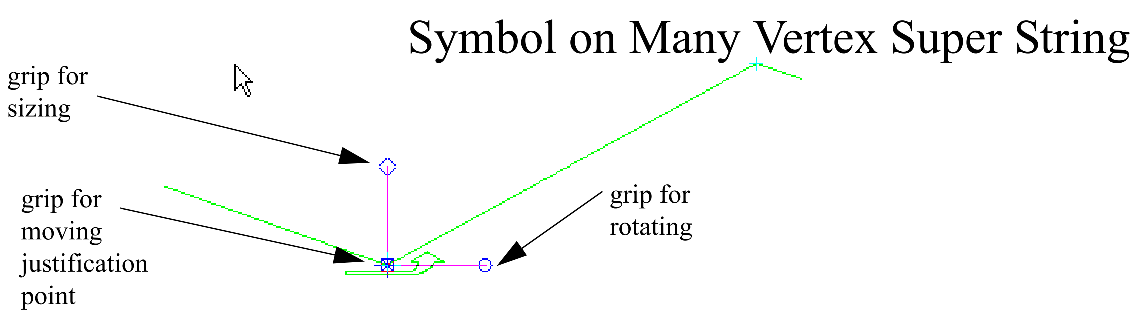
After typing **e**, the ***Pick symbol*** message is displayed in the screen message area

<[Pick symbol] (e)dit, (c)reate, (v)ertex> [picks][fast][Menu]

and a symbol s then selected by the mouse for editing.

The selected symbol with the symbol edit grips is shown on screen, and a list of available typed commands in the screen message area to quickly allow editing of the symbol name, symbol size, colour, rotation, angle, x and y offset, justification point, vertex, etc.





Options displayed after selecting symbol

<[J] Pick grip point or (v)ertex, (c)olour, (s)ymbol, (h)eight, (n)ew, (f)inish, (m)ore...> [picks][fast][Menu]

and by typing **m**

<[J] offset (x), offset (y), rot(a)tion, (d)efault, (m)ore..., (l)ock, no auto(p)an> [picks][fast][Menu]

To change the symbol colour, type **C** - go to Change Symbol Colour - Typing c

To change the symbol name, type **s** - go to Change Symbol Name - Typing s

To change the symbol height, type **h** - go to Change Symbol Height - Typing h

To create/edit a new symbol, type **n** - go to Create/Edit a New Symbol - Typing n

To end the create/edit option, type **f** - go to Finish the Create/Edit - Typing f

To show the other line of options, type **m** - go to Show Other Line of Options - Typing m

To change the x offset, type **x** - go to Change Offset x - Typing x

To change the y offset, type **y** - go to Change Offset y - Typing y

To change the symbol angle, type **a** - go to Change Symbol rotation - Typing a

To reset some symbol values to their defaults, type **d** - go to Reset Symbol Parameters to Defaults - Typing d

To show the other line of options, type **m** - go to Show Other Line of Options - Typing m

To toggle vertex lock on and off, type **l** - go to Toggle Lock - Typing l

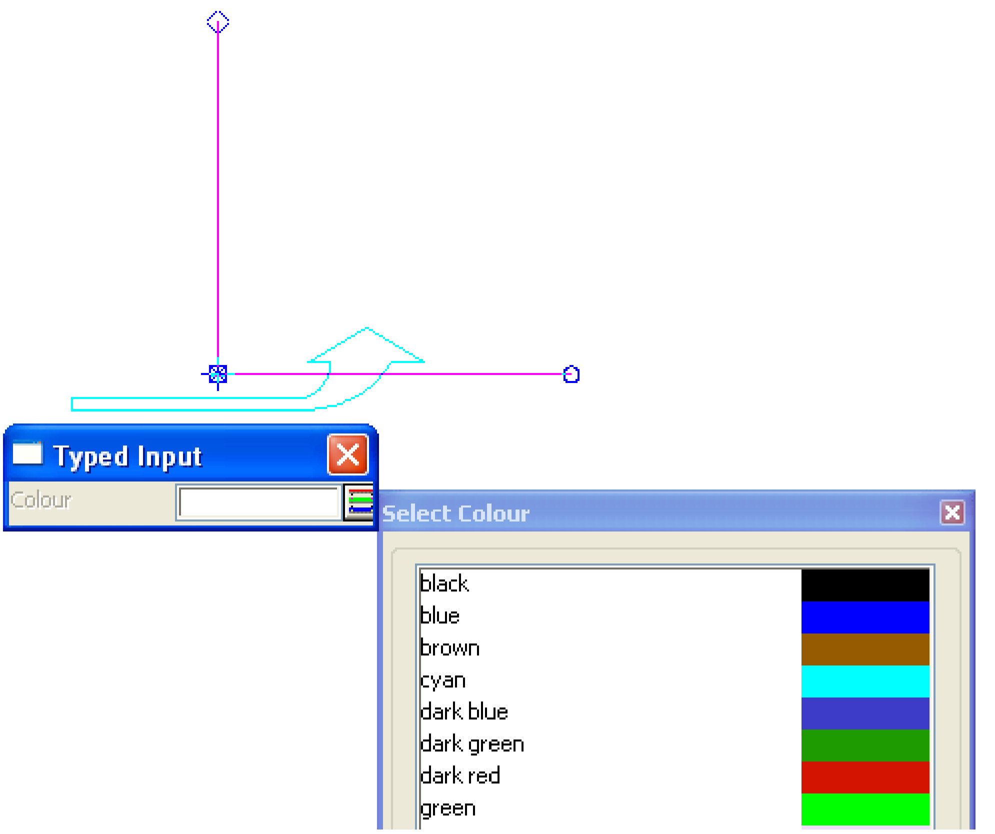
To toggle autopan on and off, type **p** - go to Toggle Autopan - Typing p

To change focus to the next symbol press **->** - go to Change Focus to the Next Vertex in String - Pressing right arrow key (->)

To change focus to the previous symbol, press **<-** - go to Change Focus to the Previous Vertex in String - Pressing left arrow key (<-)

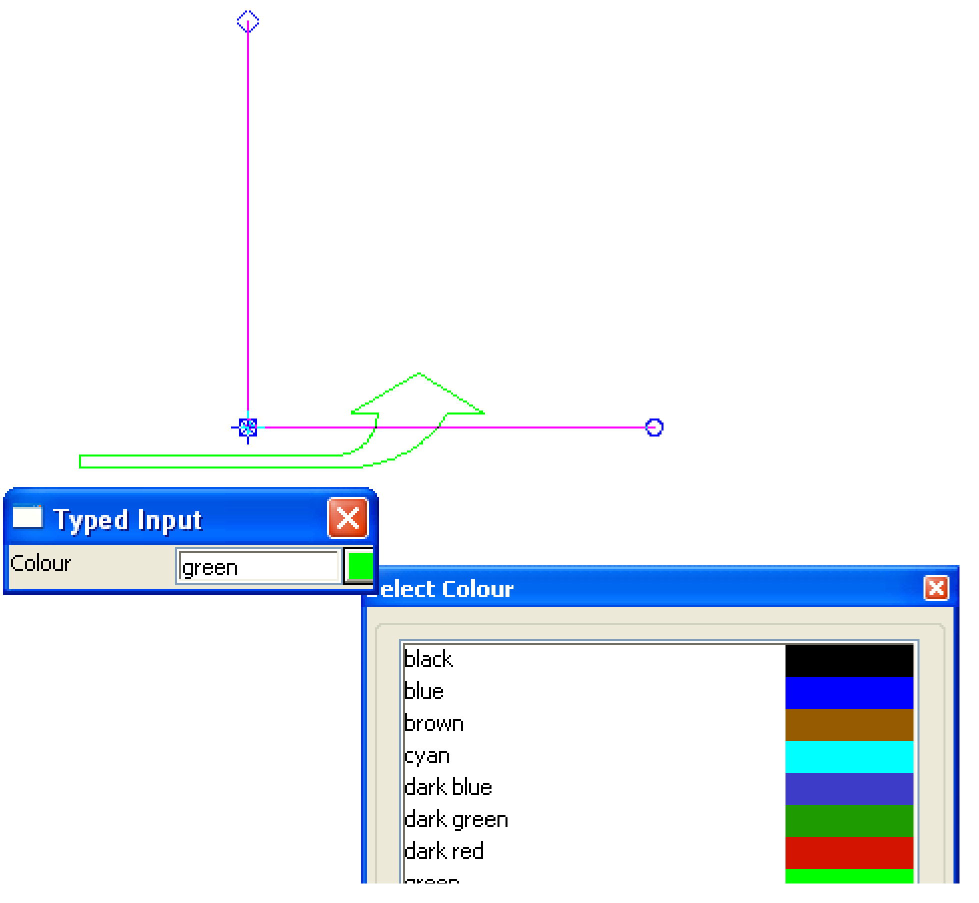
Change Symbol Colour - Typing c

After typing **c**, a Colour **Typed input** box appears on the screen.



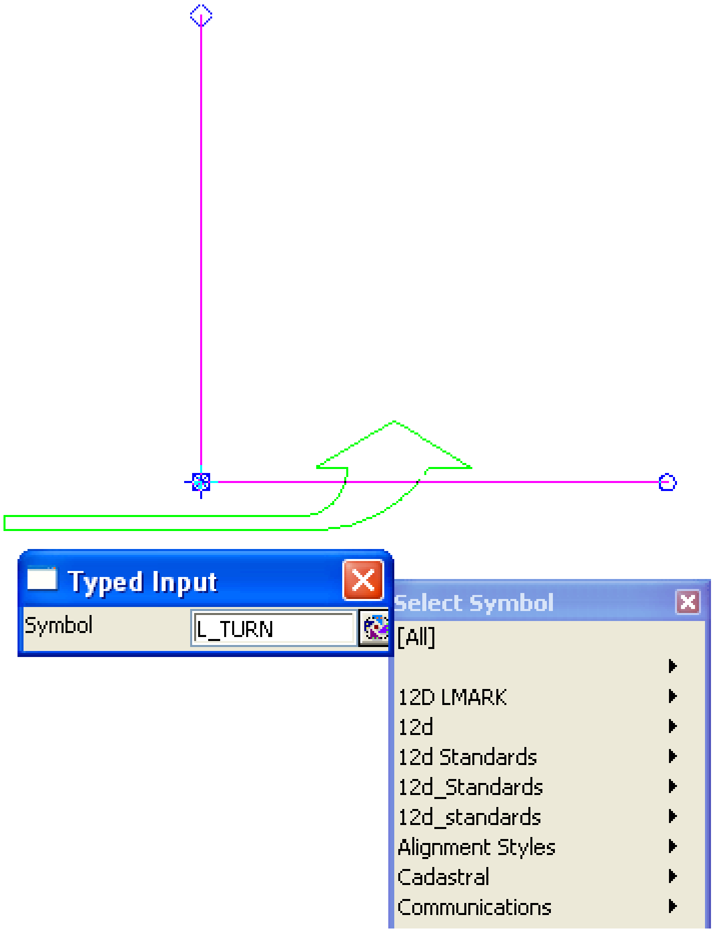
The new colour for the symbol is typed into the Typed input box followed by the <enter> key, or by selecting a colour from the **Select Colour** box which is brought up by clicking on the colour button.

If the symbol has an existing colour, other then the default colour, typing **c** will display the Typed Input box with the current colour displayed.



Change Symbol Name - Typing s

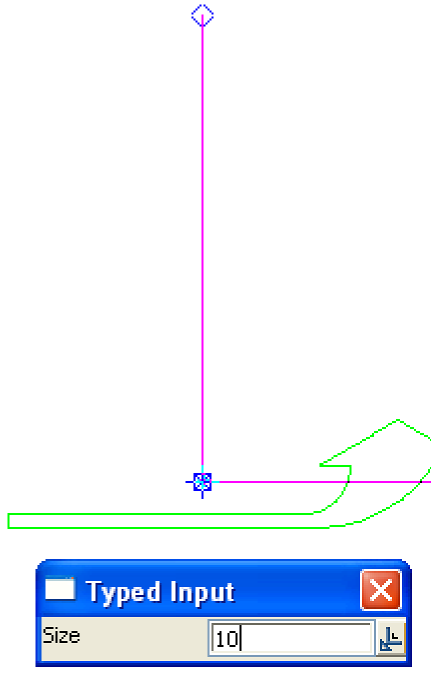
After typing **s**, a Symbol **Typed input** box appears with the current symbol style displayed.



The new style for the symbol is typed into the Style input box followed by the <Enter> key, or by selecting a style from the **Select Symbol** box which is brought up by clicking on the style button.

Change Symbol Height - Typing h

After typing **h**, a Size **Typed input** box appears with the current symbol height in it.



The new height is typed into the Size **Typed Input** box followed by the <Enter> key.

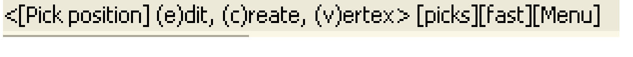
Note - Many arithmetic expressions are supported in the **Size** typed input box (*e.g.* 10+27). See the section 4.19.1 Expressions in Panel Fields.

Create/Edit a New Symbol - Typing n

If the option was in ***Edit*** mode and **n** is typed, a new **Edit** is started (**e** mode). The ***Pick symbol*** message is displayed in the screen message area and another symbol is selected to edit.

Or a ***new*** symbol can be created by typing **c** or **v**. See the section 15.16.1 Create Symbol for more information on **c** and **v**.

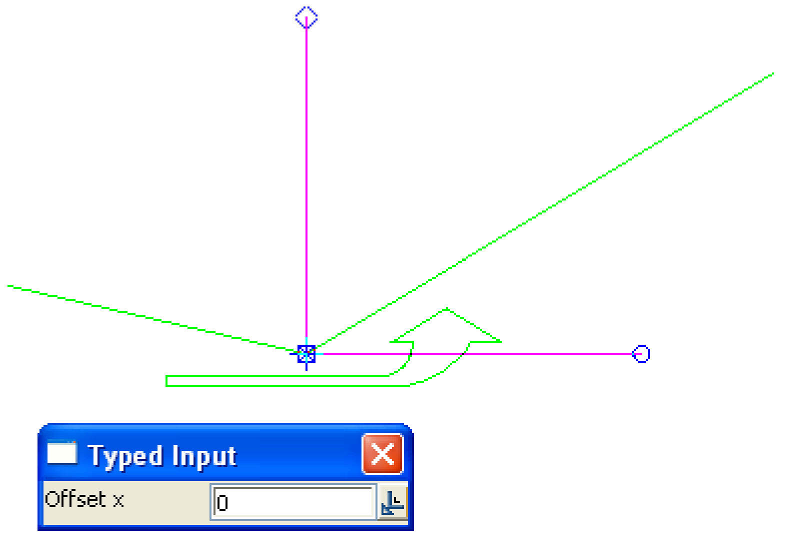
If the option was in ***Create*** mode and **n** is typed, a new **Create** is started in the **c** mode. The ***Pick position*** message is displayed in the screen message area and a new symbol is created.



Or a ***new*** vertex symbol can be created by typing **v,** or the **Edit** mode can be entered by typing **e**. See the section 15.16.1 Create Symbol for more information on **c** and **v**.

Change Offset x - Typing x

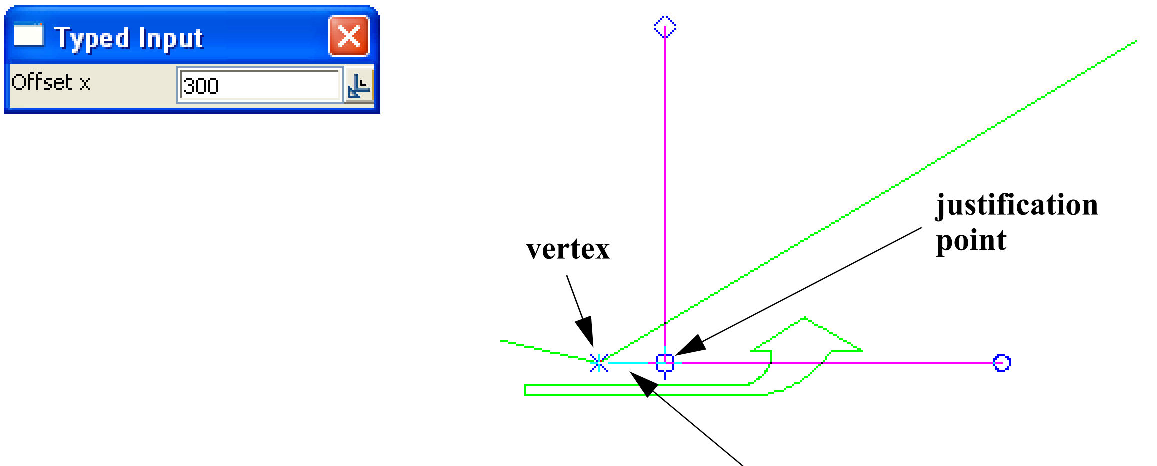
After typing **x**, an Offset x **Typed input** box appears with the current symbol x offset in it.



Symbol - offset x = 0

The new offset x is typed into the Offset x input box followed by the <Enter> key.

The symbol justification point will be offset x by the new amount



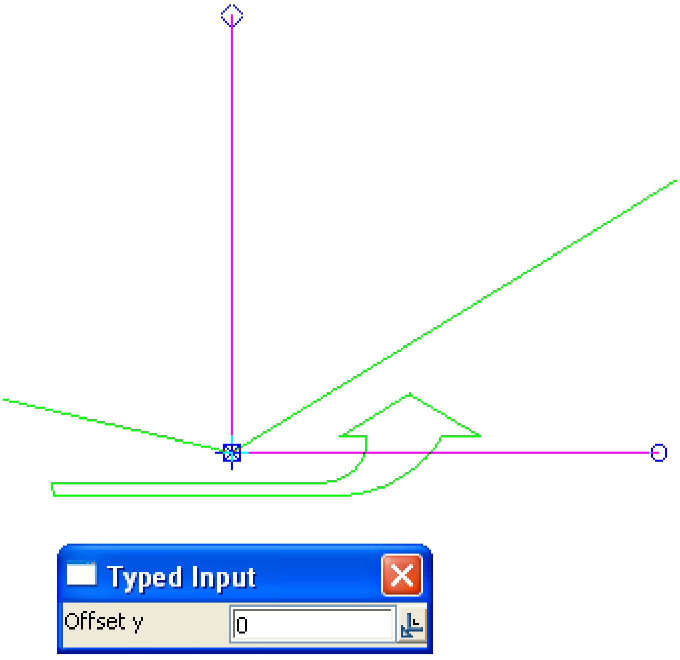
Symbol - offset x = 300

For information about Symbol and the definition of **offset**, go to the section 4.7 Symbol Definitions.

Note - Many arithmetic expressions are supported in the **Offset** typed input box (*e.g.* 10+27). See the section 4.19.1 Expressions in Panel Fields.

Change Offset y - Typing y

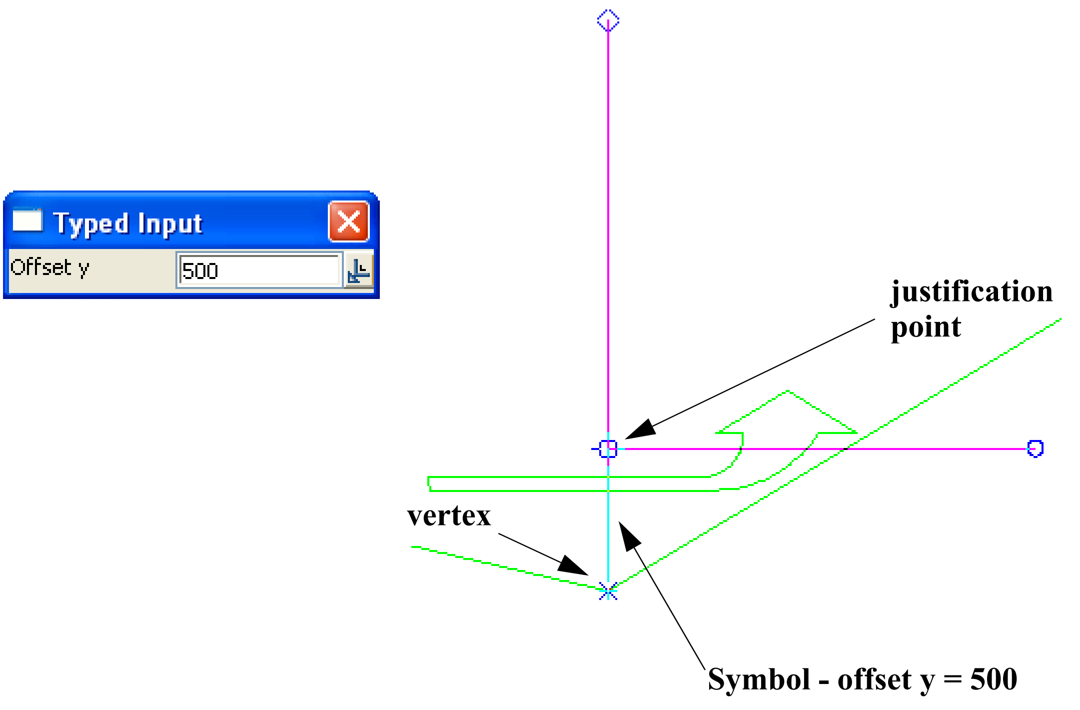
After typing **o**, an Offset y **Typed input** box appears with the current symbol offset in it.



Symbol - offset y = 0

The new offset is typed into the Offset y **Typed Input** box followed by the <Enter> key.

The symbol justification point will be offset y by the new amount

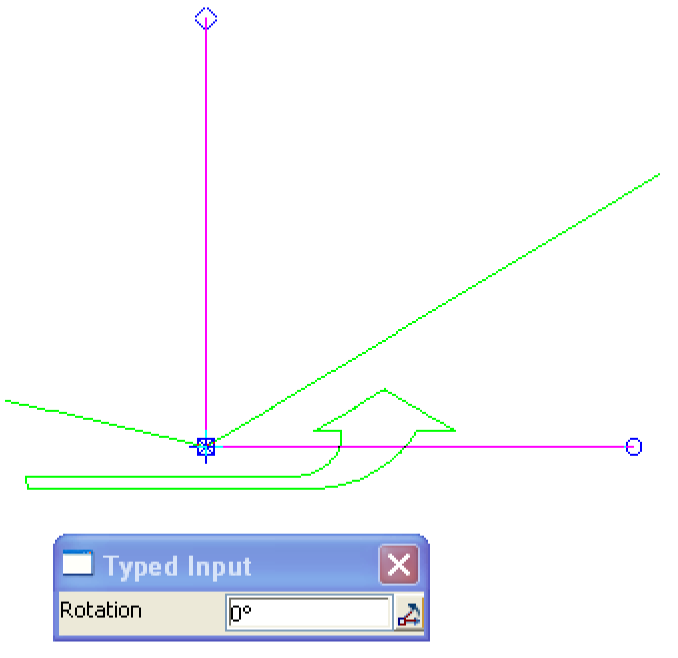


For information about symbol and the definition of **offset**, go to the section 4.7 Symbol Definitions.

Note - Many arithmetic expressions are supported in the **Offset** typed input box (*e.g.* 10+27). See the section 4.19.1 Expressions in Panel Fields.

Change Symbol rotation - Typing a

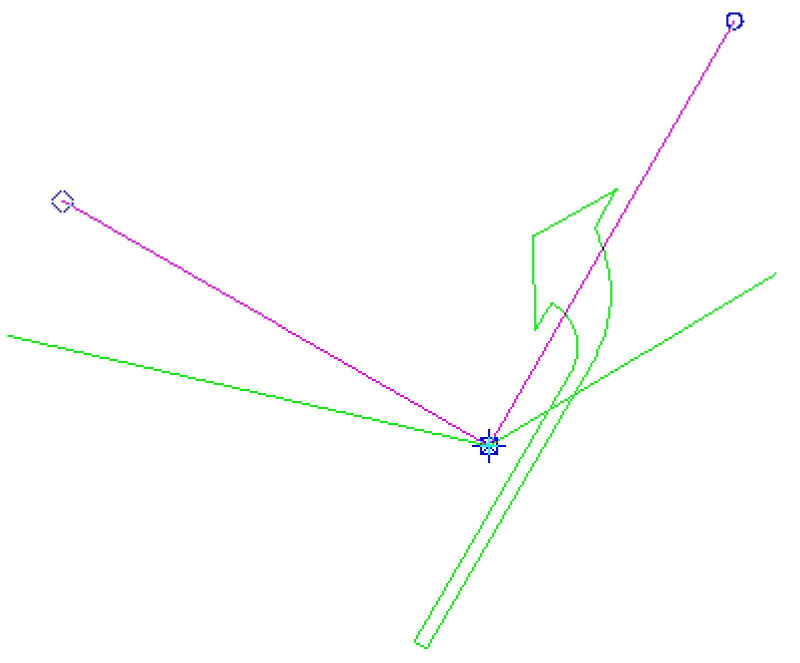
After typing **a**, a Rotation **Typed input** box appears with the current symbol angle in it.



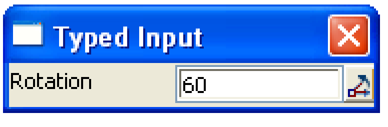
Symbol - angle rotation = 0

The new angle is typed into the Rotation **Typed input** box followed by the <Enter> key.

The direction of the symbol point will be rotated by the new amount.



Symbol - angle rotation= 60



For information about symbol and the definition of **angle rotation**, go to the section 4.7 Symbol Definitions.

Note - Some arithmetic expressions are supported in the **Rotation** typed input box (*e.g.* 10+90). See the section 4.19.1.2 Expressions in Bearing, Angle Panel Fields for what is allowed in angle/ bearing panel fields.

Reset Symbol Parameters to Defaults - Typing d

After typing **d**, some of the symbol parameters are set back to their default values.

That is:

Symbol type is set to the symbol in the Cad Symbol Controlbar

Height is set to the height in the Cad Symbol Controlbar

Angle is set to the angle in the Cad Symbol Controlbar

Model is left as it is

Colour is rest to the string colour.

symbol name, height and angle

The default symbol *type*, *height* and *angle* are defined by the **Symbol Controlbar** and *model* and *colour* from the **CAD Controlbar** (for more information, go to the section 15.1.2 Symbol Control-bar).

For information about Symbol definitions, go to the section 4.7 Symbol Definitions.

Change Focus to the Next Vertex in String - Pressing right arrow key (->)

For vertex symbol on a super string, pressing **->**, moves the focus of the edit commands to the vertex symbol on the next vertex of the super string. If the symbol is on the last vertex, pressing **->** does nothing.

For information about symbol definitions, go to the section 4.7 Symbol Definitions.

Change Focus to the Previous Vertex in String - Pressing left arrow key (<-)

When editing vertex symbol on a super string, pressing **<-,** moves the focus of the edit commands to the vertex symbol on the **previous** vertex of the super string. If the symbol is on the first vertex, pressing **<-** does nothing. If Autopan is on, the selected vertex symbol is made the centre of the active plan view.

For information about symbol definitions, go to the section 4.7 Symbol Definitions.

Toggle Lock - Typing l

Typing **l** toggles locking the vertex on and off.

When vertex lock is on, the vertex grip can not be selected and used to move the position of the string vertex.

When the screen message area reads (l) lock (l)ock, , this means that vertex lock is off (unlocked) and typing **l** locks it.

When the screen message area reads un(l) lock un(l)ock, , this means that the vertex is locked and typing **l** unlocks it.

For information about the vertex symbol grip, go to the section 4.15 Symbol Grips.

For information about symbol definitions, go to the section 4.7 Symbol Definitions.

Toggle Autopan - Typing p

Typing **p** toggles autopan on and off.

When editing vertex symbol on a super string and **Autopan** is **on**, pressing **->** or **<-**, moves the focus of the edit commands to the vertex symbol on the next/previous vertex of the super string, and the selected vertex symbol is made the centre of the active plan view.

When vertex lock is on, the vertex grip can not be selected and used to move the position of the string vertex.

When the screen message area reads auto(p)an auto(p)an, , this means that autopan is off and typing **p** turns autopan on.

When the screen message area reads no auto(p)an no auto(p)an, , this means that autopan is on and typing **p** turns it off.

For information about symbol definitions, go to the section 4.7 Symbol Definitions.

Show Other Line of Options - Typing m

There are too many *Edit* options to fit onto one line of the screen message and two lines of options are required. Typing **m** toggles between the two lines of options.

<[J] Pick grip point or (v)ertex, (c)olour, (s)ymbol, (h)eight, (n)ew, (f)inish, (m)ore...> [picks][fast][Menu]

<[J] offset (x), offset (y), rot(a)tion, (d)efault, (m)ore..., (l)ock, no auto(p)an> [picks][fast][Menu]

Finish the Create/Edit - Typing f

Typing **f** exits the create/edit symbol option.

15.16.3 Symbol Delete

Position of option on menu: **CAD =>Symbol =>Delete**

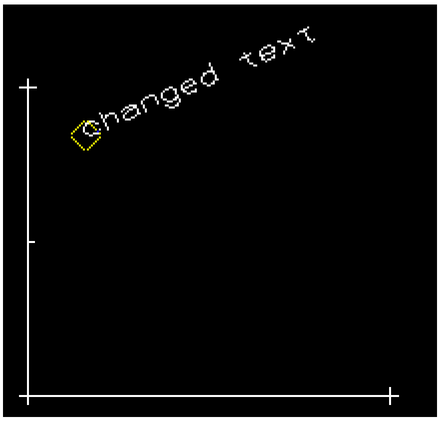
or by selection of appropriate icon from the toolbar. 

This option deletes symbols.

On selecting **Delete**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

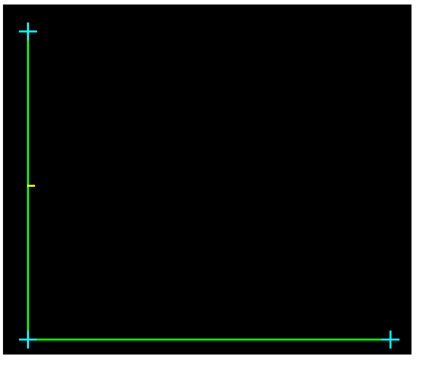
The Symbol to be deleted is selected and accepted.



STEP 2:

The selected Symbol is deleted.

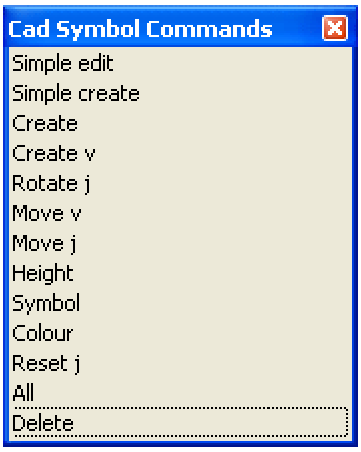
Picking Cancel from the Pick Ops menu, hitting <Esc> or selecting another option terminates the option.



15.16.3.1 Cad Symbol Commands

Position of option on menu: **CAD =>Symbol =>Commands**

The **Cad Symbol Commands** walk-right menu is



Menu of Options to Create and Manipulate Symbols

edit symbols using grips
create symbols and edit using grips
create a symbol at a selected position - a one point string is created
create a symbol at a user selected super string vertex
rotate symbol about its justification pt (which may be offset from the vertex)
move symbol by its vertex
move symbol by its justification point (the vertex does not move)
change the height of a symbol
change the symbol
change the colour of a symbol
reset the symbol justification point so that it is on top of the vertex posn

For the option *Simple edit*, go to	15.16.4 Simple Symbol Edit
Simple create	15.16.5 Simple Symbol Create
Create	15.16.6 Create Symbol
Create v	15.16.7 Create Symbol at the Vertex of a Super String
Rotate j	15.16.8 Rotate Symbol about the Justification Point
Move v	15.16.10 Move the Symbol Vertex Point
Move j	15.16.9 Move the Symbol Justification Point
Height	15.16.11 Height of Symbol
Symbol	15.16.12 Change Symbol
Colour	15.16.13 Symbol Colour
Reset j	15.16.14 Reset the Symbol Justification Point
All	15.16.15 All Symbol Edits
Delete	15.16.3 Symbol Delete

15.16.4 Simple Symbol Edit

Position of option on menu: **CAD =>Symbol =>Commands=>Simple edit**

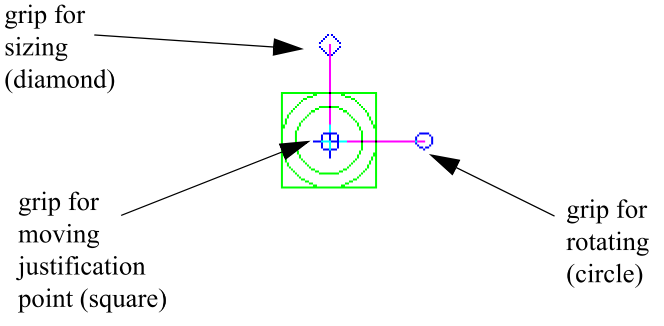
or by selection of appropriate icon from the toolbar. 

This option edits symbols using grips to quickly define the mode of editing (sizing, rotating or moving the justification point).

On selecting **Simple edit**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

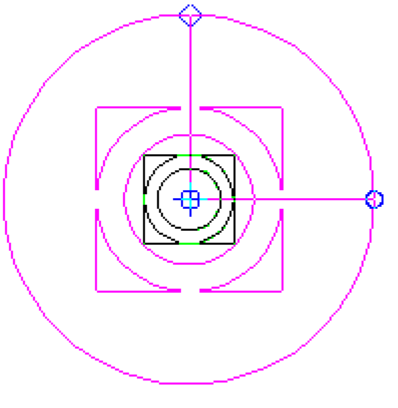
When the symbol to be edited is selected, grips for size, rotating and moving the justification point are displayed. The appropriate grip is then selected for quick editing.



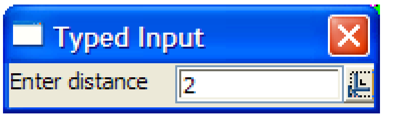
Size Grip (diamond):

If the **Size grip** is selected, the symbol size is dynamically adjusted by moving the cursor. The size of the symbol is displayed as part of the *Enter height* prompt in the message area.

The symbol size is set when a position is accepted.

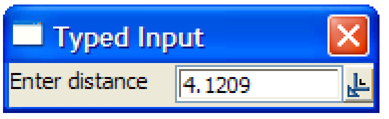


An ***exact*** symbol size can be entered by hitting the <space bar> or by starting to type a size. This brings up the **Enter distance** typed input box. The size is typed in and the <enter> key pressed.



To return to dynamic sizing without entering a distance, simply select the **X** on the top of the input box. The box will disappear and dynamic sizing will resume.

If the symbol is being dynamically sized, the ***current*** size of the symbol can be displayed by pressing the **d** key which brings up the **Enter distance** typed input box with the current dynamic symbol size in it.



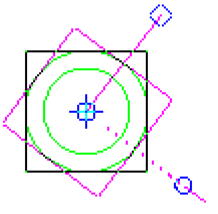
The size can be modified and the value used by pressing the <Enter> key.

To return to dynamic sizing without entering a distance, simply select the **X** on the top of the input box.

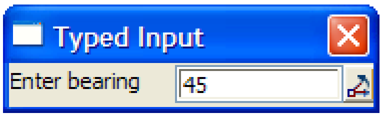
Rotate Grip (circle):

If the **Rotate grip** is selected, the symbol bearing is dynamically adjusted by moving the cursor. The bearing of the symbol is displayed as part of the *Enter bearing* prompt in the message area.

The symbol bearing is set when a position is accepted.

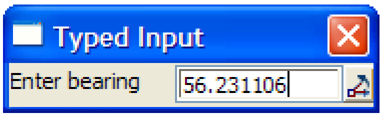


An ***exact*** bearing can be entered by hitting the <space bar> or by starting to type a bearing. This brings up the **Enter bearing** typed input box. The *bearing* is typed in and the <Enter> key pressed.



To return to dynamic rotating without entering a bearing, simply select the **X** on the top of the input box. The box will disappear and dynamic sizing will resume.

If the symbol is being dynamically rotated, the ***current*** bearing of the symbol can be displayed by pressing the **d** key which brings up the **Enter bearing** typed input box with the *current* symbol bearing in it.



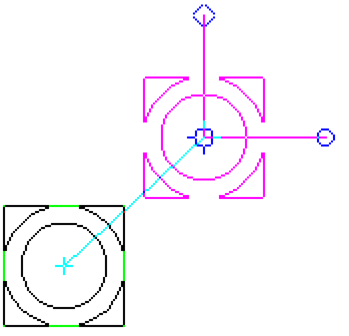
The *bearing* can be modified and the value used by pressing the <Enter> key.

To return to dynamic rotation without entering a bearing, simply select the **X** on the top of the input box.

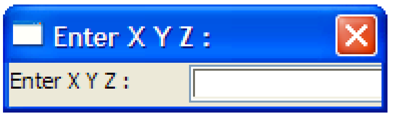
Justification Grip (square):

If the **Justification grip** is selected, the symbol justification point is dynamically adjusted by moving the cursor. The (x,y) position of the symbol is displayed in the message area.

The symbol justification point is set when a position is accepted.



An ***exact*** coordinate can be entered by hitting the <space bar> or by starting to type a coordinate. This brings up the **Enter X Y Z** typed input box. The *x and y coordinates* are typed in, separated by a space, and the <Enter> key pressed.



To return to dynamic moving without entering a coordinate, simply select the **X** on the top of the input box. The box will disappear and dynamic moving will resume.

15.16.5 Simple Symbol Create

Position of option on menu: **CAD =>Symbol =>Commands=>Simple create**

or by selection of appropriate icon from the toolbar. 

This option creates symbols and then displays grips to allow quick editing of symbol size, rotation and justification point.

The symbol and symbol height and symbol angle are taken from the **Symbol controlbar** (See 15.1 Controlbars) and the model and colour (if the symbol does not have an inbuilt colour) from the **CAD controlbar** (see 15.1 Controlbars,15.1.2 Symbol Controlbar and 15.1.1 CAD Controlbar).

The symbol will be placed with the vertex and justification point at the selected position.

After defining one symbol, the option restarts so that another symbol can be defined. Picking **Cancel** from the **Pick Ops** menu, hitting <Esc> or selecting another option terminates the option.

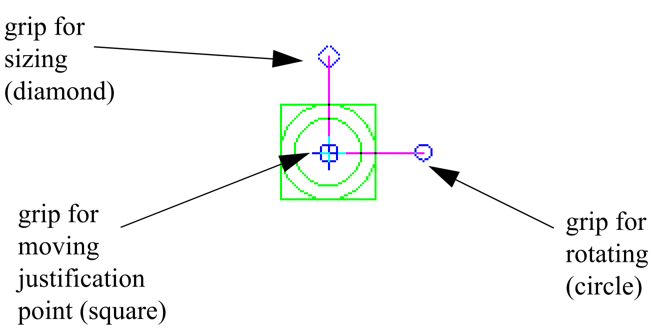
On selecting **Simple create**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

An insertion point for the symbol is selected and accepted, the symbol is displayed on the screen.

***Warning* -** if no symbol appears, then there is probably no symbol selected in the **Symbol controlbar**.

When the symbol is displayed, grips for quick editing of size, rotation and moving of the justification point are also shown. The appropriate grip can then selected for quick editing. For details on using the grips, go to the section 15.16.4 Simple Symbol Edit.



15.16.6 Create Symbol

Position of option on menu: **CAD =>Symbol =>Commands=>Create**

or by selection of appropriate icon from the toolbar.

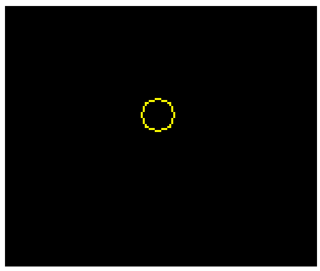
This option creates a symbol (as a one vertex super string) at a selected point (a cursor snap is valid).

After defining a symbol, the option restarts so that another symbol can be defined. Picking **Cancel** from the **Pick Ops** menu, hitting <esc> or selecting another option terminates the option.

On selecting **Create**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

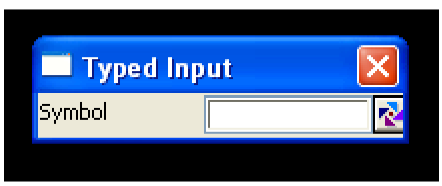
STEP 1:

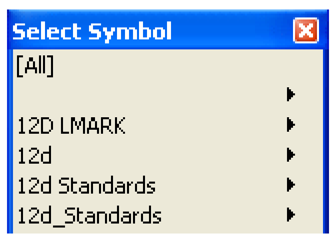
An insertion point is selected and accepted. A one vertex super string is created and the symbol placed at that vertex. The symbol is placed relative to the insertion point given in the symbol definition (See symbols.4d file).



STEP 2:

The symbol style (name) is entered into the input box or selected from the select symbol choice box. The choice box is opened by pressing the symbol icon on the right of the input box. A valid symbol type can be found by walking right on the symbol menu



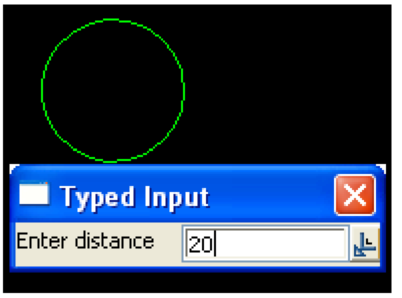


STEP 3:

A symbol size is given either by selection with the mouse or by typing a value. To type a value, simply start typing and the input box for the size will appear. The user can also press the space bar to bring up the input box. The value is entered into the input box followed by the enter key.

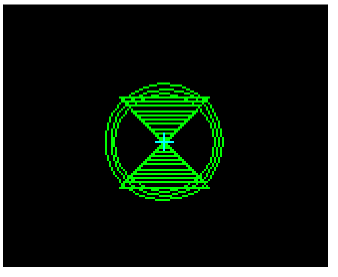
The circle drawn represents the size value and changes with movement of the mouse. If the user wants to see what the current value of the height is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the symbol or the input box can be closed and the rubber banding (graphically changing) of the circle continued.

This option also allows the definition of the size by the selection of a point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next.



STEP 4:

The symbol is created together with a point super string at the selected point.



15.16.7 Create Symbol at the Vertex of a Super String

Position of option on menu: **CAD =>Symbol =>Commands=>Create v**

or by selection of appropriate icon from the toolbar.

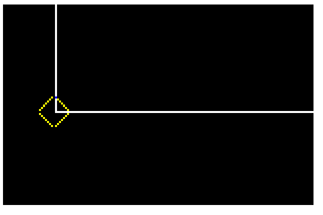
This option creates a symbol at a vertex of a super string. One symbol is allowed per super string vertex, so if this option is used on existing vertex symbol, the existing value is displayed in the symbol input box.

After defining a symbol, the option restarts so that another symbol can be defined. Picking Cancel from the Pick Ops menu, hitting <esc> or selecting another option terminates the option.

On selecting **Create v**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

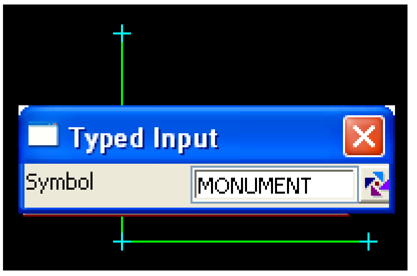
STEP 1:

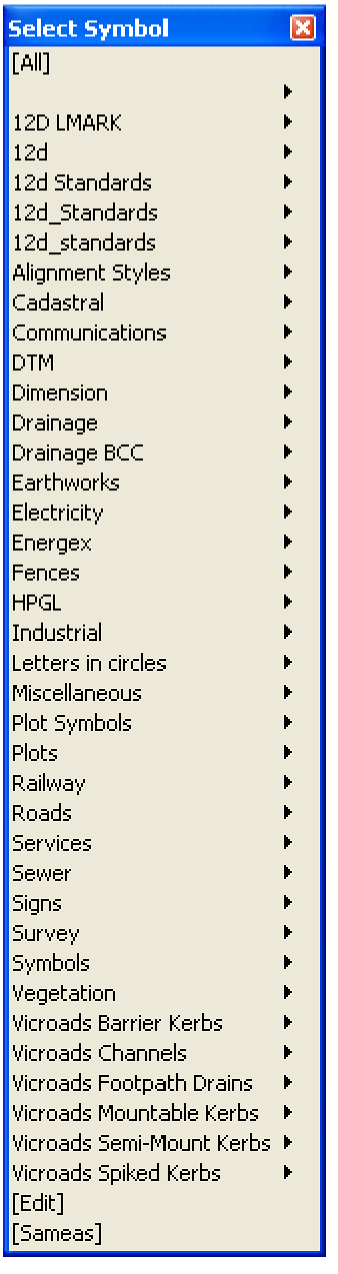
An insertion point is selected and accepted. The point snap is forced on so that a vertex can be selected. The Symbol will be placed relative to the insertion point given other parameters such as justification, as defined in the symbol definition.



STEP 2:

The symbol style is entered into the input box or selected from the select symbol choice box. The choice box is opened by pressing the symbol icon on the right of the input box. A valid symbol type can be found by walking right on the symbol menu



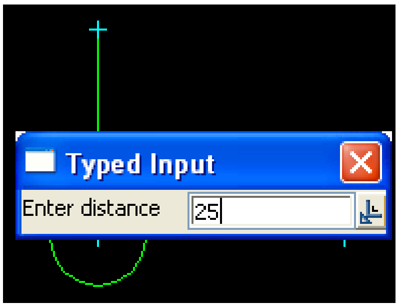


STEP 3:

A symbol size is given either by selection with the mouse or by typing a value. To type a value, simply start typing and the input box for the size will appear. The user can also press the space bar to bring up the input box. The value is entered into the input box followed by the enter key.

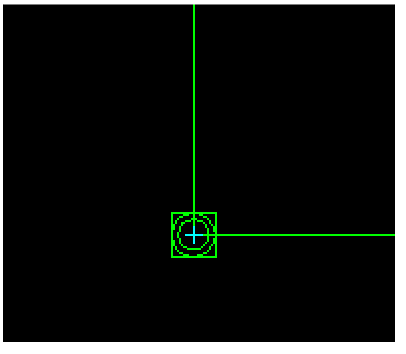
The circle drawn represents the size value and changes with movement of the mouse. If the user wants to see what the current value of the height is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the symbol or the input box can be closed and the rubber banding (graphically changing) of the circle continued.

This option also allows the definition of the size by the selection of a point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next.



STEP 4:

The symbol is created at the vertex of the super string.



15.16.8 Rotate Symbol about the Justification Point

Position of option on menu: **CAD =>Symbol =>Commands=>Rotate j**

or by selection of appropriate icon from the toolbar.

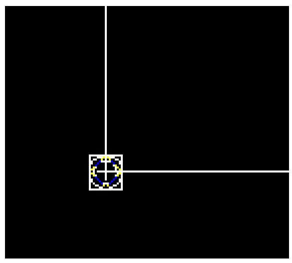
This option rotates a symbol about its justification point. The vertex point and the justification point often do not coincide. This occurs when non zero value is specified for either the x or y offset.

Note: The typed bearing is not relative but absolute. i.e. the value given will not rotate the existing symbol by that amount, rather it will reposition the symbol at that bearing.

On selecting **Rotate j**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

The Symbol to be rotated is selected and accepted.

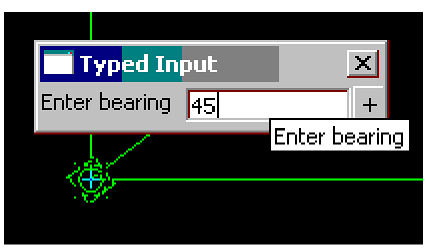


STEP 2:

The rotation bearing (the final bearing of the symbol) is specified by selection and acceptance of a point with the mouse or by entry into an angle box followed by the enter key. To bring up the angle box start typing or press the space bar. The value is entered into the input box followed by the enter key.

The line drawn represents the bearing value and changes with movement of the mouse. If the user wants to see what the current value of the bearing is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to specify the bearing or the input box can be closed and the rubber banding (graphically changing) of the bearing continued.

Note: **The Page up and page down keys can be used when the bearing input box comes up to add or subtract intervals of 90 degrees.**

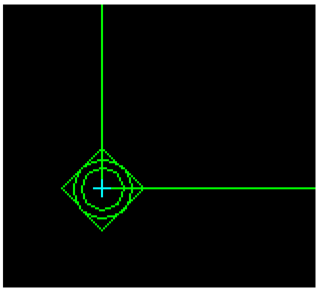


This option also allows the definition of the bearing by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next. The example shown below is the perpendicular case.

STEP 3:

The Symbol is rotated to the specified bearing.

Picking Cancel from the Pick Ops menu, hitting <esc> or selecting another option terminates the option.



15.16.9 Move the Symbol Justification Point

Position of option on menu: **CAD =>Symbol =>Commands=>Move j**

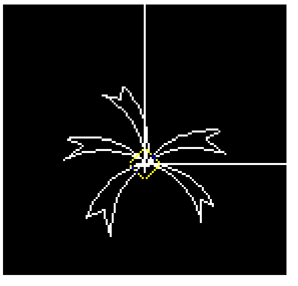
or by selection of appropriate icon from the toolbar. 

This option moves a symbol by moving its justification point.

On selecting **Move j**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

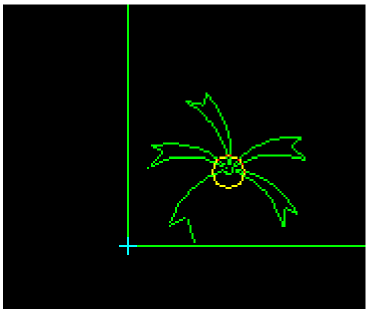
STEP 1:

The Symbol to be moved is selected and accepted.



STEP 2:

The new position for the Symbol is selected and accepted.



STEP 3:

The Symbol moves to the new position.

Picking Cancel from the Pick Ops menu, hitting <esc> or selecting another option terminates the option.

15.16.10 Move the Symbol Vertex Point

Position of option on menu: **CAD =>Symbol =>Commands=>Move v**

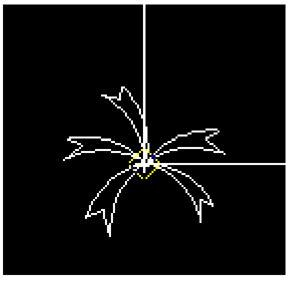
or by selection of appropriate icon from the toolbar. 

This option moves a symbol by moving the vertex that the symbol is attached to. Hence the symbol and the justification point for the symbol will move.

On selecting **Move V**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

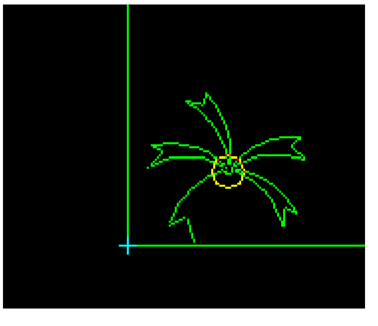
STEP 1:

The Symbol to be moved is selected and accepted.



STEP 2:

The new position for the vertex of the Symbol is selected and accepted.



STEP 3:

The Symbol moves to the new position.

Picking Cancel from the Pick Ops menu, hitting <esc> or selecting another option terminates the option.

15.16.11 Height of Symbol

Position of option on menu: **CAD =>Symbol =>Commands=>Height**

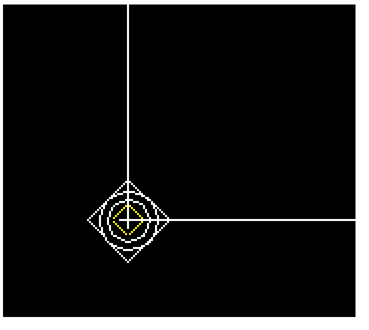
or by selection of appropriate icon from the toolbar. 

This option changes the height of a symbol.

On selecting **Height**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

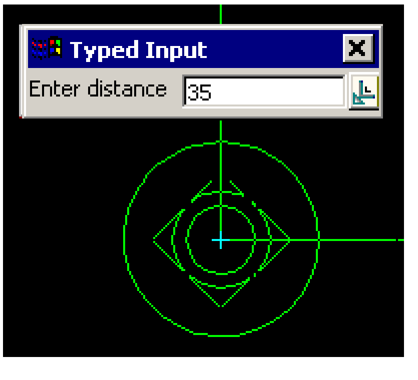
The Symbol of which the height is to be changed is selected and accepted.



STEP 2:

A height for the Symbol is given either by selection with the mouse or by typing a value. To type a value, simply start typing and the input box for the height will appear. Alternatively you can press the space bar to bring up the input box. The value is entered into the input box followed by the enter key.

The circle drawn represents the height value and changes with movement of the mouse. If the user wants to see what the current value of the height is, simply press the **D** key (dynamic value). This puts the value into the input box where it can be accepted to create the point or the input box can be closed and the rubber banding (graphically changing) of the circle continued.

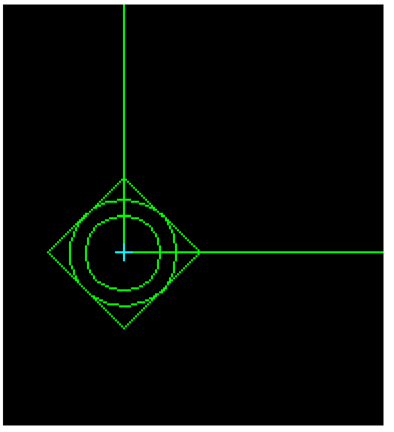


This option also allows the definition of the distance by the selection of the 2nd point perpendicular or tangential to a selected segment. For this, the line snap should be on. The user selects the segment (line or arc) and then by pressing **P** for perpendicular or **T** for tangential a solution is shown. As there is often two solutions with respect to arcs, the user can move the mouse to change from one solution to the next.

STEP 3:

The Symbol height is changed.

Picking Cancel from the Pick Ops menu, hitting <esc> or selecting another option terminates the option.



15.16.12 Change Symbol

Position of option on menu: CAD =>Symbol =>Commands=>Symbol

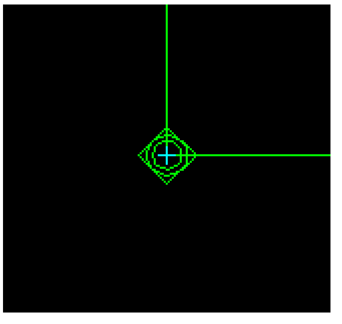
or by selection of appropriate icon from the toolbar. 

This option changes the actual symbol defined at a super string vertex.

On selecting **Symbol**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

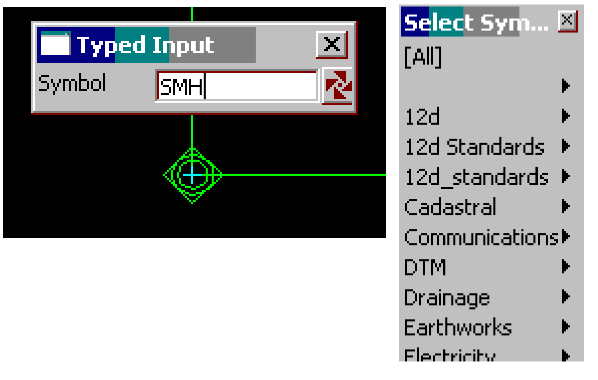
STEP 1:

The Symbol to be changed is selected and accepted.



STEP 2:

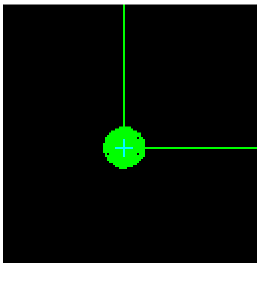
The symbol style is entered into the input box or selected from the select symbol choice box. The choice box is opened by pressing the symbol icon on the right of the input box. A valid symbol type can be found by walking right on the symbol menu



STEP 3:

The Symbol is changed.

Picking Cancel from the Pick Ops menu, hitting <esc> or selecting another option terminates the option.



15.16.13 Symbol Colour

Position of option on menu: **CAD =>Symbol =>Commands=>Colour**

or by selection of appropriate icon from the toolbar.

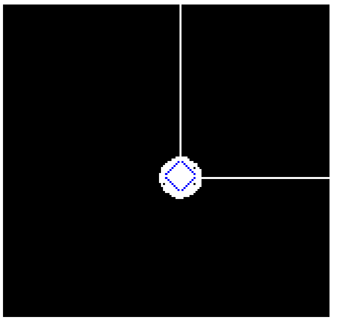
This option changes the default colour of a symbol.

Important Note: if the symbol has inbuilt colours, changing the default colour will not change the symbols displayed colours. For this case, the definition of the symbol itself will need to be changed.

On selecting **Colour**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

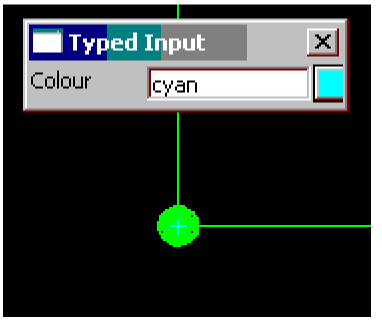
STEP 1:

The symbol to have the default colour changed is selected and accepted.



STEP 2:

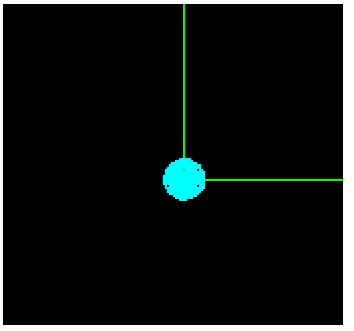
The new colour for the Symbol is entered into the Symbol input box or by selecting the colour square on the input box to bring up the select colour choice box followed by the enter key.



STEP 3:

The Symbol colour is changed.

Picking Cancel from the Pick Ops menu, hitting <esc> or selecting another option terminates the option.



15.16.14 Reset the Symbol Justification Point

Position of option on menu: **CAD =>Symbol =>Commands=>Reset j**

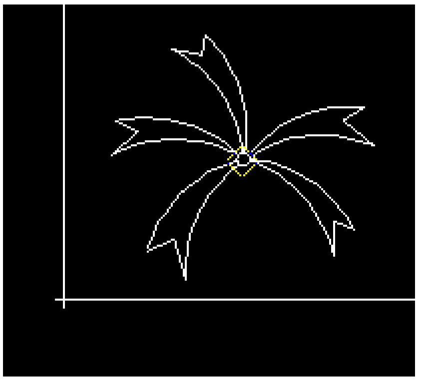
or by selection of appropriate icon from the toolbar.

This option resets justification point of a selected symbol. That is, the x and y offsets for the justification point are set to zero.

On selecting **Reset j**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

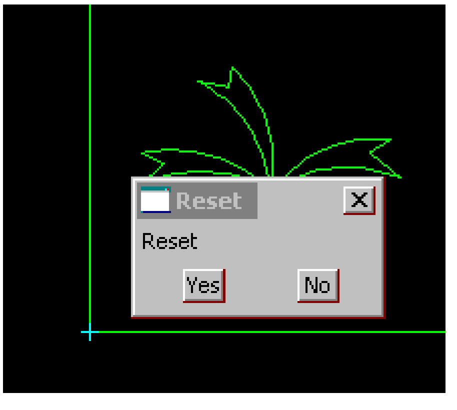
STEP 1:

The Symbol to be changed is selected and accepted.



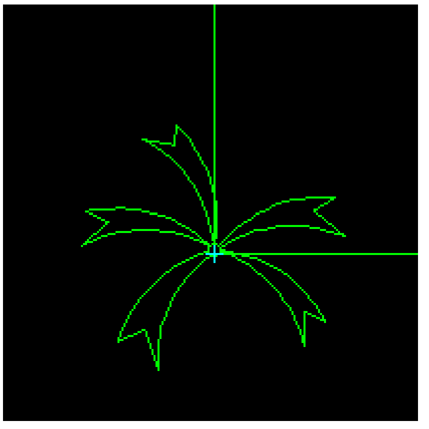
STEP 2:

A warning message is shown giving the user the option of resetting the justification point or not.



STEP 3:

The symbol is reset if accepted.



15.16.15 All Symbol Edits

Position of option on menu: **CAD =>Symbol =>Commands=>All**

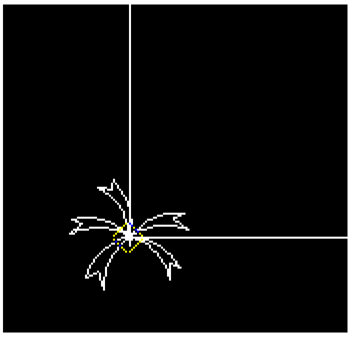
or by selection of appropriate icon from the toolbar. 

This option allows a number of symbol editing options to be done in succession using a menu made up of many of the options defined above.

On selecting **All**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

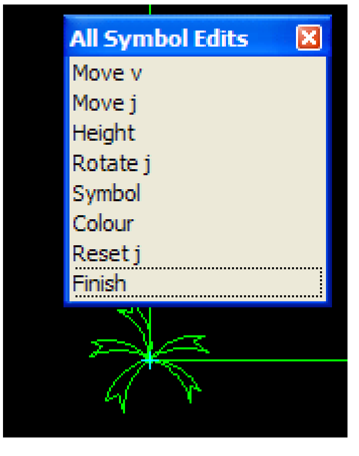
STEP 1:

The Symbol to be changed is selected and accepted.



STEP 2:

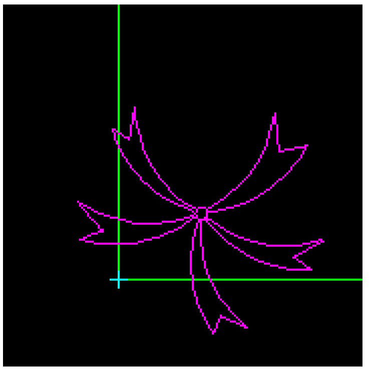
After the selection and acceptance of a symbol, the positioning menu is shown. This also allows various other symbol editing functions such as colour, height and rotate.



STEP 3:

The user can select the appropriate option from the menu and make the changes. The menu remains active allowing a number of operations to be made is succession.

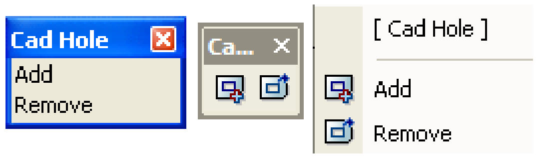
Picking Cancel from the Pick Ops menu, hitting <esc> or selecting another option terminates the option.



15.17 CAD Hole

Position of option on menu: **CAD =>Hole**

The C**ad Hole** walk-right menu is



For the option *Add*, go to [15.17.1 Add Holes](#)
Remove [15.17.2 Remove Holes](#)

15.17.1 Add Holes

Position of option on menu: **CAD =>Hole =>Add**

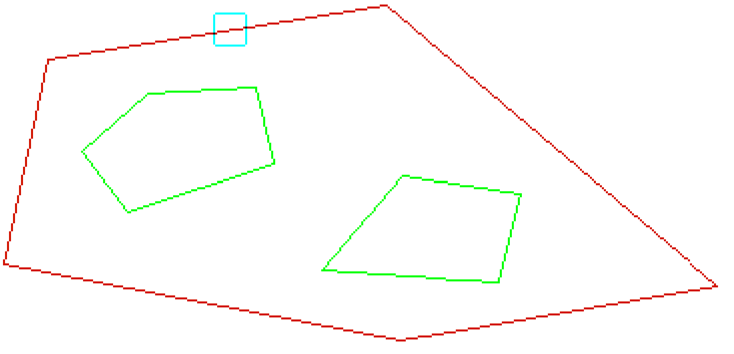
or by selection of appropriate icon from the toolbar.

This option adds holes to a polygon.

On selecting **Add holes**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

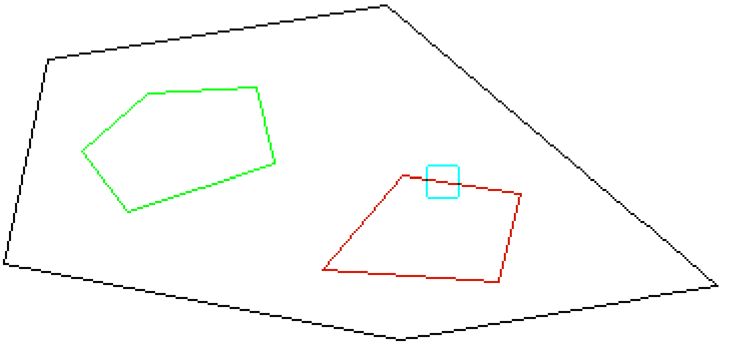
First select the parent string. That is the polygon that is to have holes added to it.



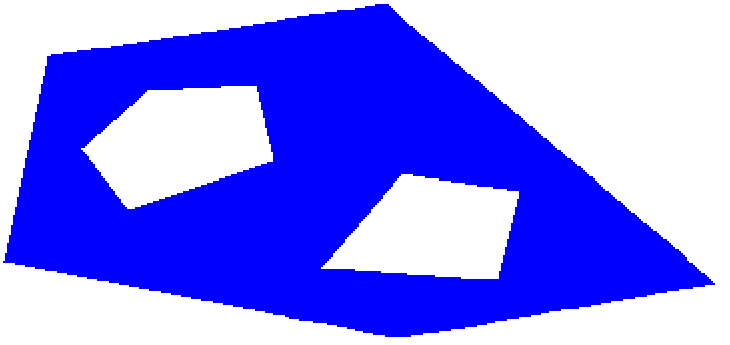
STEP 2:

Then select the polygons that are to be the holes in the parent polygon (the child strings).

The selection of holes is terminated by pressing the <esc> key.



It is easier to see the result by colour filling the parent polygon.



15.17.2 Remove Holes

Position of option on menu: CAD =>Hole =>Remove

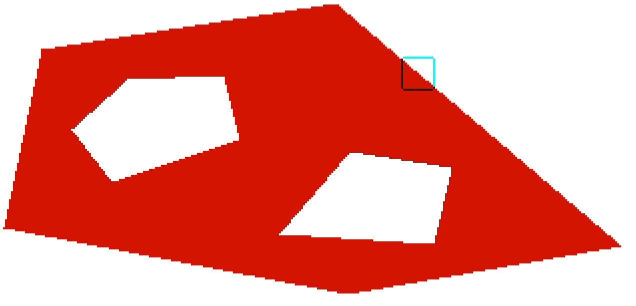
or by selection of appropriate icon from the toolbar.

This option removes all holes from a polygon with holes.

On selecting **Remove holes**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

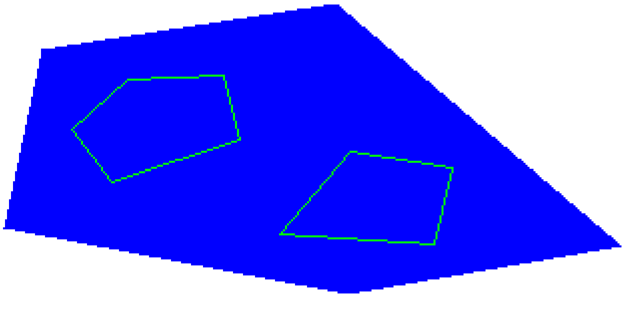
STEP 1:

Select the parent string. That is the polygon that contains holes.



STEP 2:

On selecting and accepting the string, all holes are removed from the string. Super strings are created for each of the hole boundaries.



15.18 CAD Fill

Position of option on menu: **CAD =>Fill**

The **Fill** walk-right menu is

For the option *Solid*, go to 15.18.1 Solid Fill
Solid many 15.18.2 Solid Fill Many Strings

15.18.1 Solid Fill

Position of option on menu: **CAD =>Fill =>Solid**

or by selection of appropriate icon from the toolbar.

This option fills a super string with a user defined colour and blend.

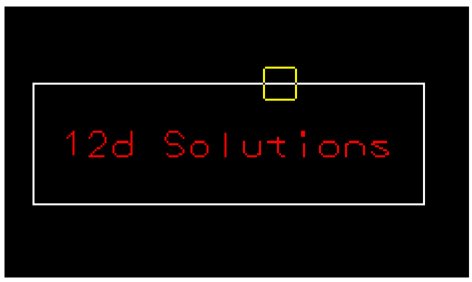
On selecting **Solid Fill**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

Select and accept a string.

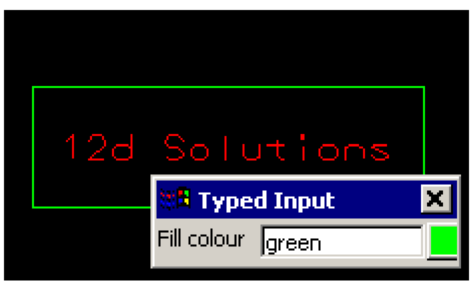
If the string selected is an enclosed shape, then the filled area is simply the closed shape. If the string isn't closed (such as two perpendicular lines), for the purposes of this option, the string will be temporarily closed and closed area coloured.

Note: If the string selected is a straight line, then its area is zero. When the colour fill of a line is completed, the line will simply disappear. This is because there is no area to colour.



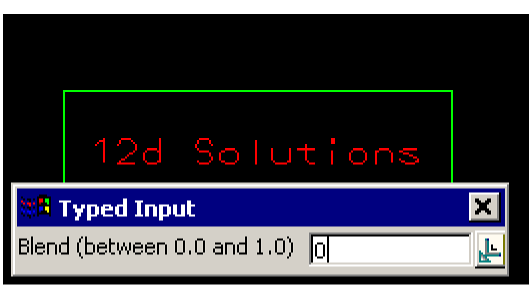
STEP 2:

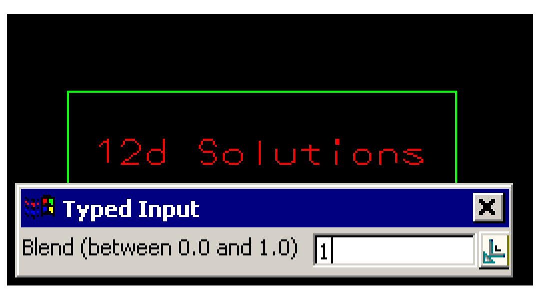
After the string is accepted, the **Fill Colour** Input box will appear. Type the name of the required colour into the Input box and press the enter key. The browse button on the Input box can be used to define the blend by measuring existing elements.



STEP 3:

After the colour is accepted, the Blend Input box will appear. This selection determines the transparency of the filled colour. A value between 0 (totally opaque) and 1(totally transparent) must be selected.





STEP 4:

The selected super string is then filled.

Blend 0

Blend 1

15.18.2 Solid Fill Many Strings

Position of option on menu: **CAD =>Fill =>Solid many**

or by selection of appropriate icon from the toolbar.

This option fills many selected super strings with the same user defined colour and blend.

On selecting **Solid fill many**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

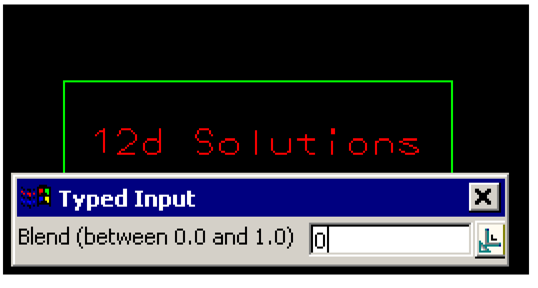
STEP 1:

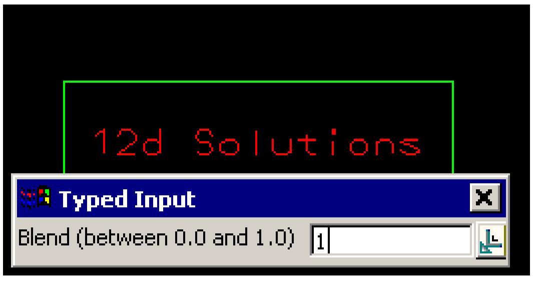
After selecting the option, the **Fill Colour** Input box is displayed.

Type the name of the required colour into the Input box and press the enter key. The browse button on the Input box can be used to define the blend by measuring existing elements.

STEP 2:

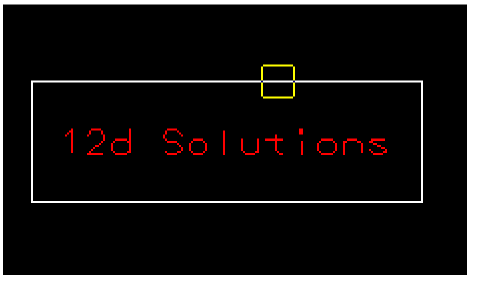
After the colour is accepted, the **Blend** Input box will appear. This selection determines the transparency of the filled colour. A value between 0 (totally opaque) and 1(totally transparent) must be selected.





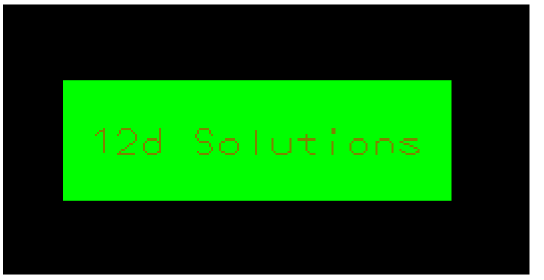
STEP 3:

Select and accept a string and the string is coloured filled.



The selected super string is then filled.

Blend 0



Blend 1

Then select and accept the next string to fill and the same colour and blend is used to colour fill the next string

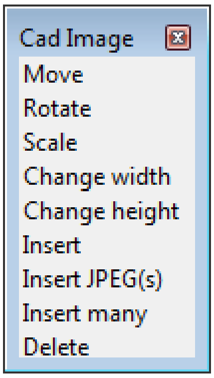
If the string selected is an enclosed shape, then the filled area is simply the closed shape. If the string isn't closed (such as two perpendicular lines), for the purposes of this option, the string will be temporarily closed and closed area coloured.

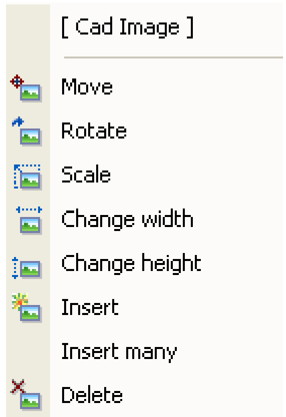
Note: If the string selected is a straight line, then its area is zero. When the colour fill of a line is completed, the line will simply disappear. This is because there is no area to colour.

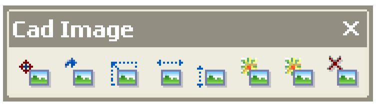
15.19 CAD Image

Position of option on menu: **CAD =>Image**

The **Cad Image** walk-right menu is







For the option *Move*, go to 15.19.1 Move Image
Rotate 15.19.2 Rotate Image
Scale 15.19.3 Scale Image
Change width 15.19.4 Change Width
Change height 15.19.5 Change Height
Insert 15.19.6 Insert Image
Insert JPEG(s) 15.19.7 Insert JPEG(s)
Insert many 15.19.8 Insert Many
Delete 15.19.9 Delete Image

15.19.1 Move Image

Position of option on menu: **CAD =>Image =>Move**

or by selection of appropriate icon from the toolbar.

This option moves an inserted image by modifying the x and y offset for the image. The vertex that the image is attached to does not move.

On selecting **Move Image**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

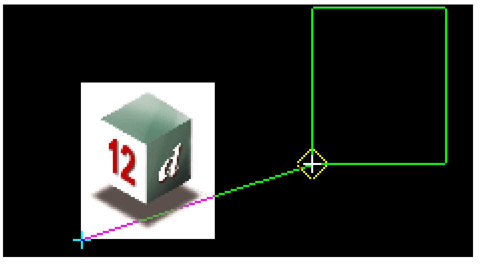
STEP 1:

Select the vertex that the image is attached to.

Note: The position of the image is defined **relative to the vertex.** To select the image, the **vertex** that the image is attached to **must be selected**. The image itself can not be picked.

STEP 2:

After a vertex of an image is accepted, an outline of the image is displayed and moves with the position of the cursor. This will continue until the new position is selected and accepted.



Select the new position and accept it.

The cursor position can be selected with the mouse or entered via the keyboard. To enter a position with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the <enter> key.

STEP 3:

The image is then displayed at its new position.

Note: The ***CAD Move Image*** option only moves **the image offset relative to the vertex.** Use the option ***CAD Move*** to **move the vertex** (and the corresponding image moves with it).

15.19.2 Rotate Image

Position of option on menu: **CAD =>Image =>Rotate**

or by selection of appropriate icon from the toolbar.

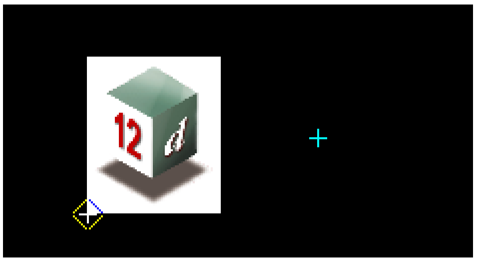
This option rotates an image around its justification point.

On selecting **Rotate Image**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

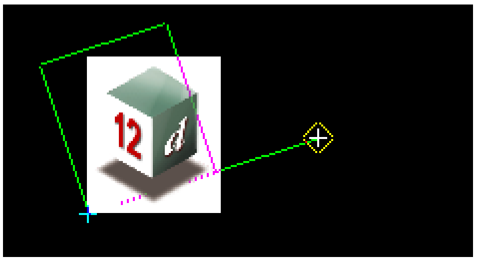
Select the vertex that the image is attached to.

Note: The position of the image is defined **relative to the vertex.** To select the image, the **vertex** that the image is attached to **must be selected**. The image itself can not be picked.



STEP 2:

After the vertex of an image is accepted, an outline of the image is displayed. The outline is rotated dynamically as the position of the cursor changes. This will continue until a final position is selected and accepted.

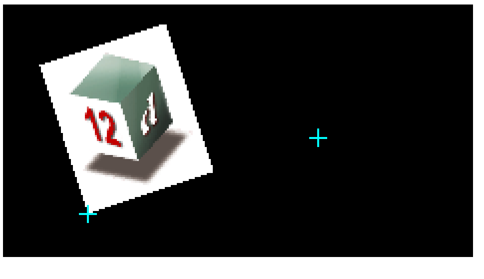


Select the new rotated position and accept it.

The cursor position can be selected with the mouse or entered via the keyboard. To enter a position with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the <enter> key.

STEP 3:

The rotated image is then displayed.



15.19.3 Scale Image

Position of option on menu: **CAD =>Image =>Scale**

or by selection of appropriate icon from the toolbar.

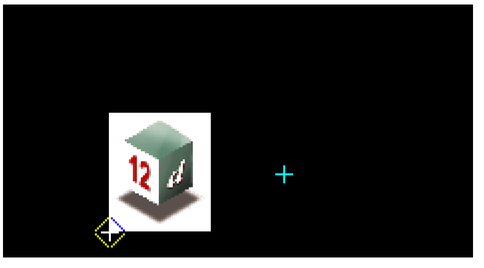
This option changes the size of an image.

On selecting **Scale Image**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

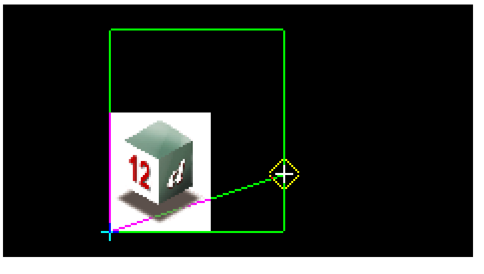
Select the **vertex** that the image is attached to.

Note: The position of the image is defined **relative to the vertex.** To select the image, the **vertex** that the image is attached to **must be selected**. The image itself can not be picked.



STEP 2:

After the vertex of an image is accepted, an outline of the image is displayed. The size of the outline is determined by the position of the cursor. Both the height and the width of the image increase/decrease in the same ratio so that the image is not distorted.

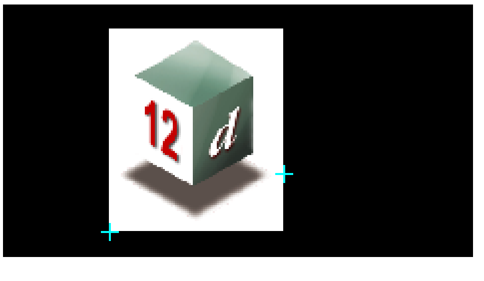


Select the new size as given by the cursor position, and accept it.

The cursor position can be selected with the mouse or entered via the keyboard. To enter a position with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the <enter> key.

STEP 3:

The scaled image is then displayed.



15.19.4 Change Width

Position of option on menu: **CAD =>Image => Change width**

or by selection of appropriate icon from the toolbar. 

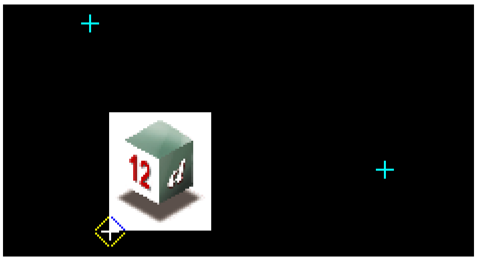
This option is user to increase or decrease the width of an image. The height of the image is not changed so the images will be stretched.

On selecting **Width Image**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

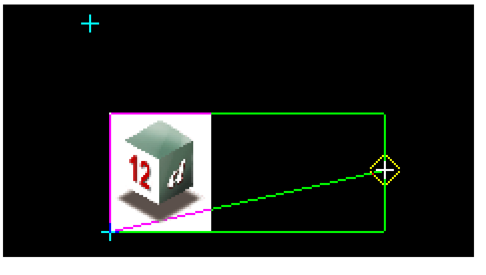
Select the **vertex** that the image is attached to.

Note: The position of the image is defined **relative to the vertex.** To select the image, the **vertex** that the image is attached to **must be selected**. The image itself can not be picked.



STEP 2:

After the vertex of an image is accepted, the outline of the image is displayed. The width of the outline is determined by the position of the cursor. Whilst the height remains constant, the width of the outline increases and decreases dynamically as the position of the cursor changes.



Select the new width as given by the cursor position, and accept it.

The cursor position can be selected with the mouse or entered in via the keyboard. To enter a position with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the <enter> key.

STEP 3:

The image with the new width is then displayed.

15.19.5 Change Height

Position of option on menu: **CAD =>Image => Change height**

or by selection of appropriate icon from the toolbar.

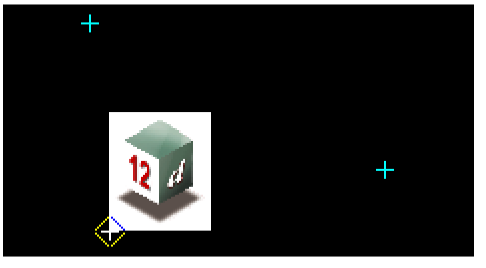
This option allows the user to increase or decrease the height of an image. The width of the image is not changed so the images will be stretched.

On selecting **Height Image**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

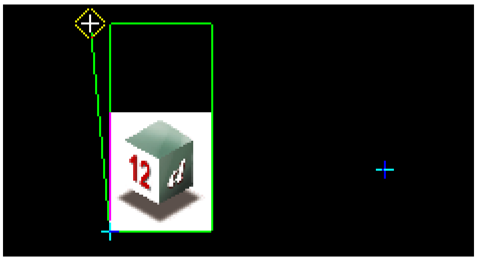
Select the **vertex** that the image is attached to.

Note: The position of the image is defined **relative to the vertex.** To select the image, the **vertex** that the image is attached to **must be selected**. The image itself can not be picked.



STEP 2:

After the vertex of an image is accepted, the outline of the image is displayed. The height of the outline is determined by the position of the cursor. Whilst the width remains constant, the height of the outline increases and decreases dynamically as the position of the cursor changes.



Select the new height as given by the cursor position, and accept it.

The cursor position can be selected with the mouse or entered in via the keyboard. To enter a position with the keyboard, simply start typing or press the space bar to bring up the XYZ Input box. Type the coordinates into the XYZ Input box and press the <enter> key.

STEP 3:

The image with the new height is then displayed.

15.19.6 Insert Image

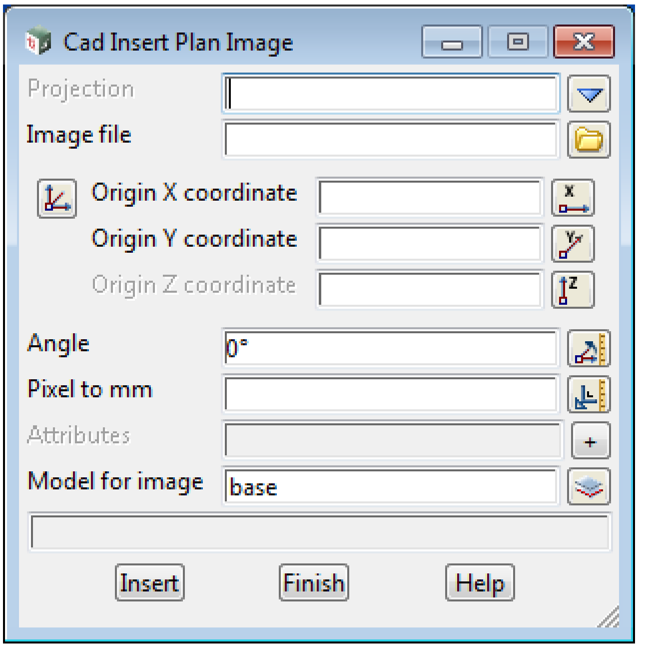
Position of option on menu: **CAD =>Image =>Insert**

or by selection of appropriate icon from the toolbar.

This option takes a user selected image file and creates a super string vertex with the image inserted at that vertex.

If the file type is JPEG and it includes the EXIF and GPS information, the information is loaded into attributes for the image.

On selecting **Insert**, the **Cad Insert Plan Image** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Format	input box		BMP, DIB, ECW, GIF, JPEG JPEG 2000, PNG, TGA TIFF, 12D

select the type of image file to insert. If the file type is JPEG and it includes the EXIF and GPS information, the information is loaded into attributes for the image and they can be examined by clicking '+' next to the Attributes *field.*

Image file input box

Select the image filename to be inserted into the project.

X/Y co-ordinate XY Pick button

select the x/y co-ordinate for origin (bottom left corner) of the image.

Origin X coordinate input box

enter the x coordinate for the bottom left of the image.

Origin Y coordinate input box

enter the y coordinate for the bottom left of the image.

Pixel to mm input box

the units for images are pixels (width and height). The pixel width and height of the image are multiplied by the Pixel to mm *(mm being millimetres) value to give width and height size in world unit* *(metres) which is needed when inserting the image.*

Attributes

when JPEG is selected for the Format *and the JPEG file includes the EXIF and GPS information, the information is loaded into attributes for the image. The attributes can be looked at by clicking on the '+'.*

Model for image input box

select the model for the image to be inserted into.

Insert button

*when all the fields have been entered, the **Insert** button creates an image from the selected file.*

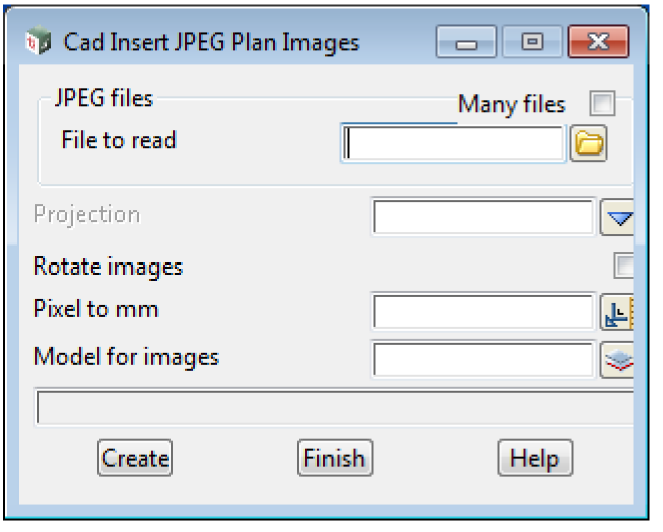
15.19.7 Insert JPEG(s)

Cad_Insert_JPEG_Plan_Images**Position of option on menu:** **CAD =>Image =>Insert JPEG(s)**

This option is for reading in JPEG files which must include EXIF and GPS information. This information is then used to position the image.

If a Projection is selected by the user, then it is used to convert the lat/long of the GPS values to the XYZ co-ordinates. If no Projection is given, then the co-ordinates are left as lat/long.

On selecting **Insert JPEG(s)**, the **Cad Insert JPEG Plan Images** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

Advanced tick box not ticked

clicking Advanced *brings up a grid to allow the user to enter many JPEG files.*

File to read file box available *.jpg files

the JPEG files to read in and create images from. The JPEG file must include EXIF and GPS information, which is used to position the image. If Projection *is not blank, then it is used to convert the lat/long of the GPS values to the XYZ co-ordinates. If* Projection *is blank, then the co-ordinates are left as lat/long.*

Projection choice box

if a Project Projection is set, then it is placed in the Projection *field. This can be changed to any other projection. The projection is used to convert the lat/long of the GPS values to the XYZ co-ordinates.*

If no projection is set, then the co-ordinates are left as lat/long.

Rotate images tick box not ticked

*if **ticked**, the rotate value from the EXIF will be used to rotate the image.*

Pixel to mm input box

the units for images are pixels (width and height). The pixel width and height of the image are multiplied by the Pixel to mm *(mm being millimetres) value to give width and height size in world unit (metres) which is needed when inserting the image.*

Model for images input box

the model for the image to be inserted into.

Create button

*when all the fields have been entered, the **Create** button creates images from the selected files.*

15.19.8 Insert Many

Position of option on menu: CAD =>Image =>Insert Many

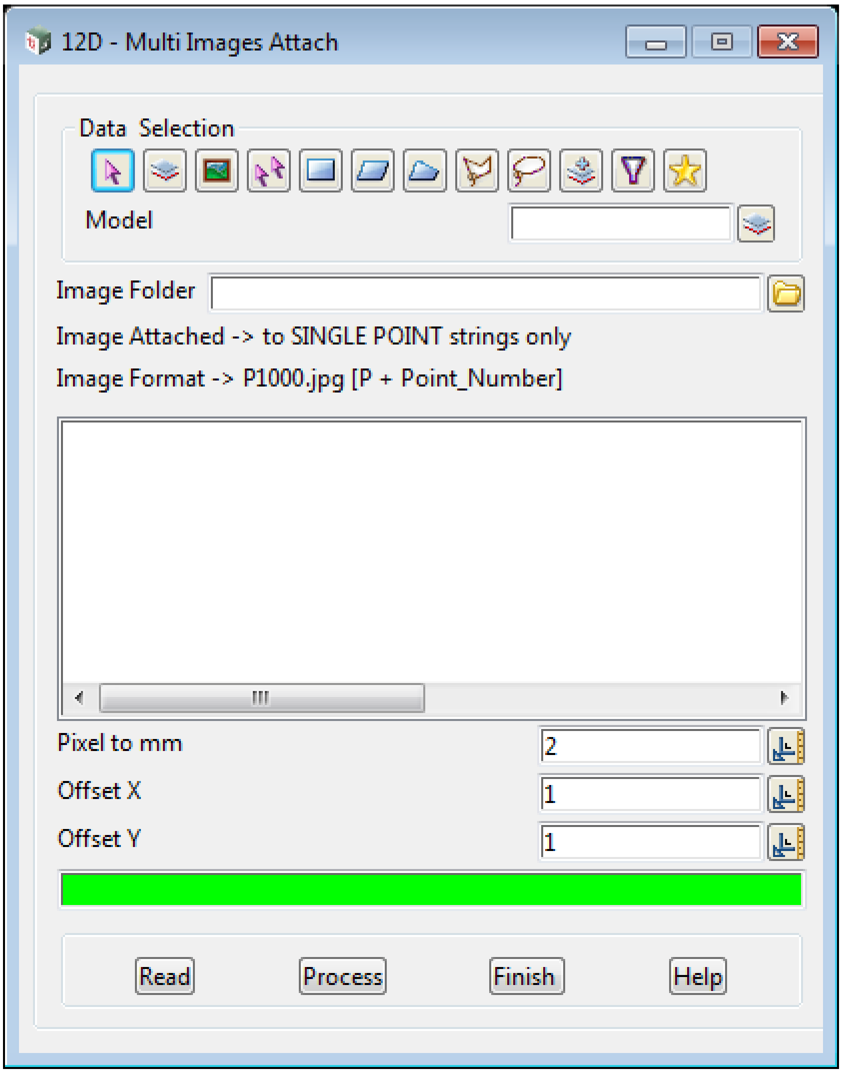
or by selecting the icon from the Cad Images **toolbar.** 

This option allows the attachment of an image to a selection of super point strings.

As shown on the panel, the images are to be stored in a directory folder and must be in the following format.

Image Format: P1000.jpg [P + point number]

Note: Attachment to Single Point super strings only



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Image Folder	directory		select folder

browse for a folder containing the images (folder only)

Pixel to mm	measure box	At Point, Point to Point, String from Point, String to Point

scale factor for image (check pixel size first)

Offset X	measure box	At Point, Point to Point, String from Point, String to Point

image offset in the x direction from point

Offset Y	measure box	At Point, Point to Point, String from Point, String to Point

image offset in the y direction from point

Read button

searches the image folder for the jpg images and displays them in the list box above

Process button

attaches the image to appropriate point

15.19.9 Delete Image

Position of option on menu: **CAD =>Image =>Delete**

or by selection of appropriate icon from the toolbar.

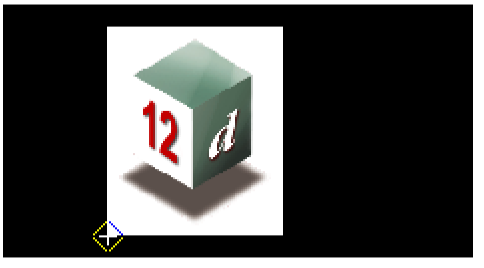
This option deletes an image.

On selecting **Delete Image**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

Select the **vertex** that the image is attached to.

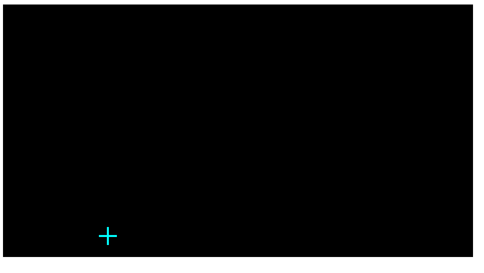
Note: The position of the image is defined **relative to the vertex.** To select the image, the **vertex** that the image is attached to **must be selected**. The image itself can not be picked.



STEP 2:

When the selected image is accepted, it is then deleted.

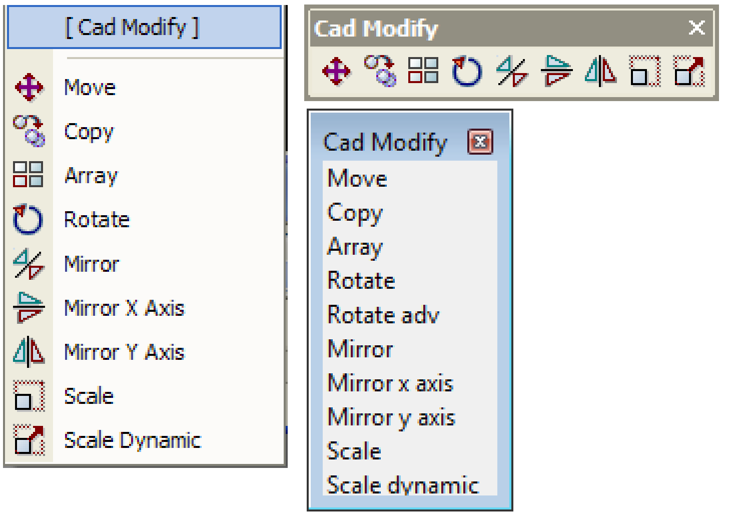
Note that the **vertex** that the image was attached to is not deleted.



15.20 CAD Modify

Position of option on menu: **CAD =>Modify**

The **Modify** walk-right menu is



For the option *Move*, go to	15.20.1 CAD Move
Copy	15.20.2 CAD Copy
Array	15.20.3 CAD Array
Rotate	15.20.4 CAD Rotate
Rotate adv	
Mirror	15.20.5 CAD Mirror
Mirror x axis	15.20.6 CAD Mirror X Axis
Mirror y axis	15.20.7 CAD Mirror Y Axis
Scale	15.20.8 CAD Scale
Scale dynamic	15.20.9 CAD Scale Dynamic

15.20.1 CAD Move

Position of option on menu: **CAD =>Modify =>Move**

or by selection of appropriate icon from the toolbar. 

This option moves a selected string.

On selecting **Move**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the *12d Model* application window.

For **CAD Multipick Move**, go straight to Step 2.

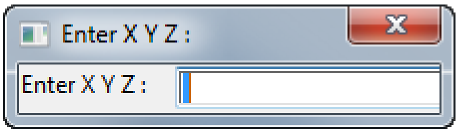
STEP 1:

A super string or circle is picked.

STEP 2:

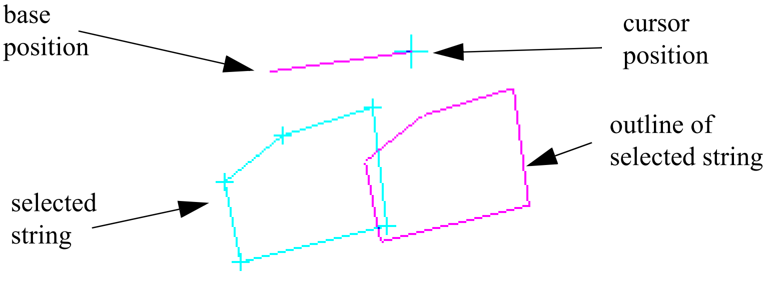
A base position is picked and then an outline of the selected string will be drawn and moved with the cursor, plus a line will be drawn from the base position to the current cursor position.

NOTE: If the <Spacebar> or any other key is pressed, an Enter X Y Z typed input box is displayed on the screen.



X, Y and Z are entered, separated by a space, and <Enter> accepts the values and removes the Enter X Y Z box from the screen. If values are not going to be typed in, clicking **X** on the top right-hand corner of the Enter X Y Z box removes the box from the screen and the option returns to selecting the base position by moving the cursor.

Note: The base position is not required to lie on the string itself.



STEP 3:

A final cursor position is selected which defines a distance and direction from the base position to the final position.

STEP 4:

On accepting the final position, the selected super string is moved by the determined distance and direction.

15.20.2 CAD Copy

Position of option on menu: **CAD =>Modify =>Copy**

or by selection of appropriate icon from the toolbar. 

This option copies a super string.

On selecting **Copy**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

For **CAD Multipick Copy**, go straight to Step 2.

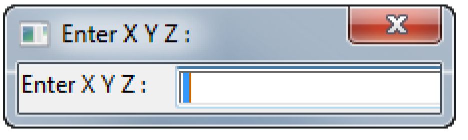
STEP 1:

Select a super string.

STEP 2:

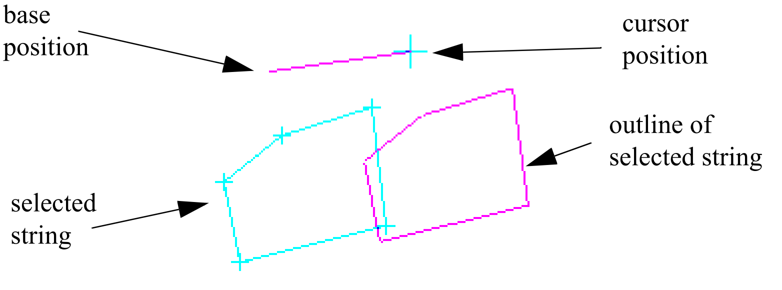
A base position is picked and then an outline of the selected string will be drawn and moved with the cursor, plus a line will be drawn from the base position to the current cursor position.

NOTE: If the <Spacebar> or any other key is pressed, an Enter X Y Z typed input box is displayed on the screen.



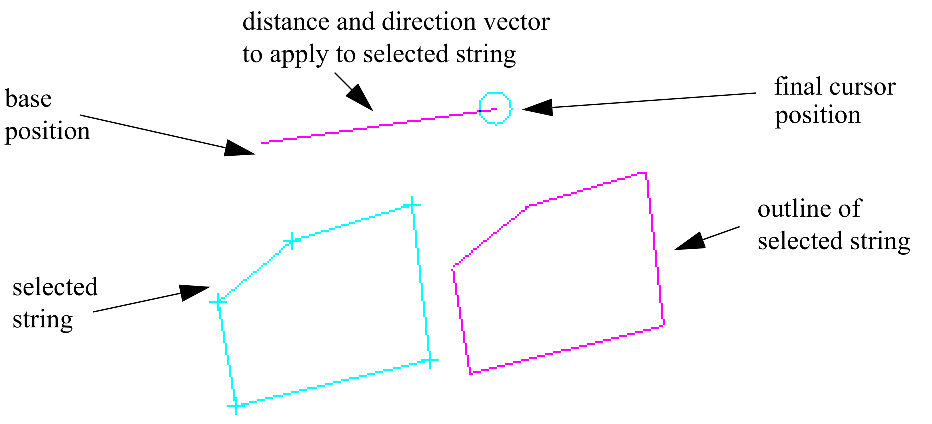
X, Y and Z are entered, separated by a space, and <Enter> accepts the values and removes the Enter X Y Z box from the screen. If values are not going to be typed in, clicking on **X** on the top right-hand corner of the Enter X Y Z box removes the box from the screen and the option returns to selecting the base position by moving the cursor.

Note: The base position is not required to lie on the string itself.



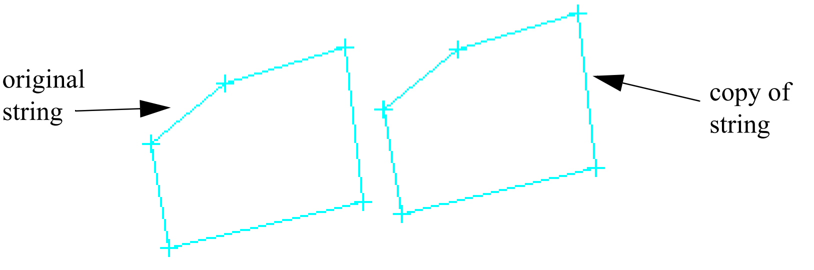
STEP 3:

A final cursor position is selected which defines a distance and direction from the base position to the final position.



STEP 4:

On accepting the final position, a copy of the selected super string is moved by the determined distance and direction.



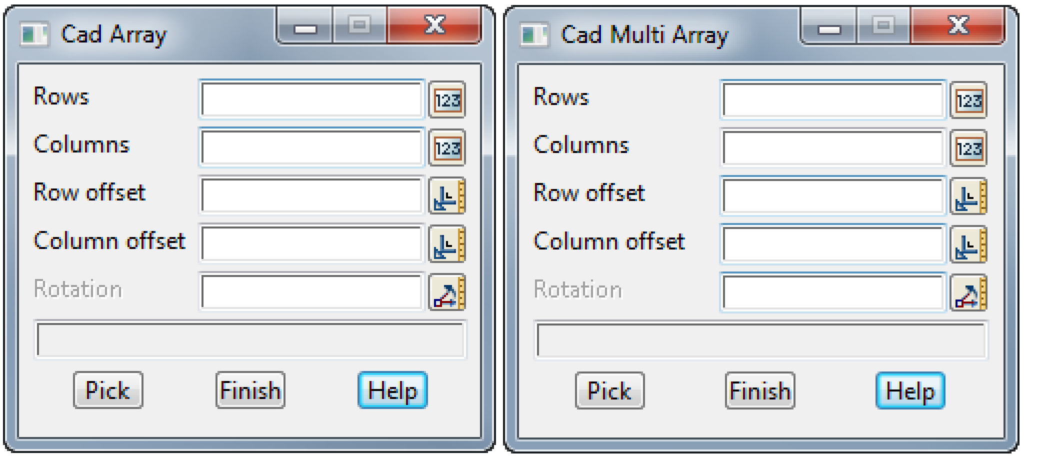
15.20.3 CAD Array

Position of option on menu: **CAD =>Modify =>Array**

or by selection of appropriate icon from the toolbar.

This option copies a super string a number of times. The Multipick Array option works in the same way once the *Multipick set* is selected.

On selecting **Array**, either the **Cad Array** panel (for CAD Array) or the **CAD Multi Array** panel (for CAD Multipick Array) is displayed.

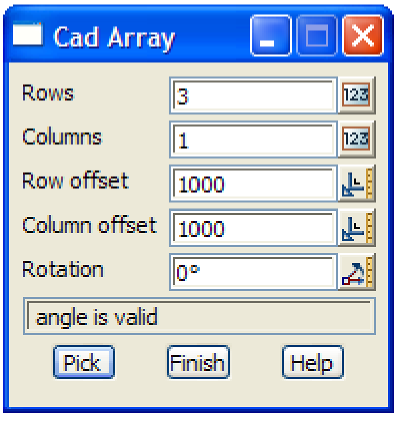


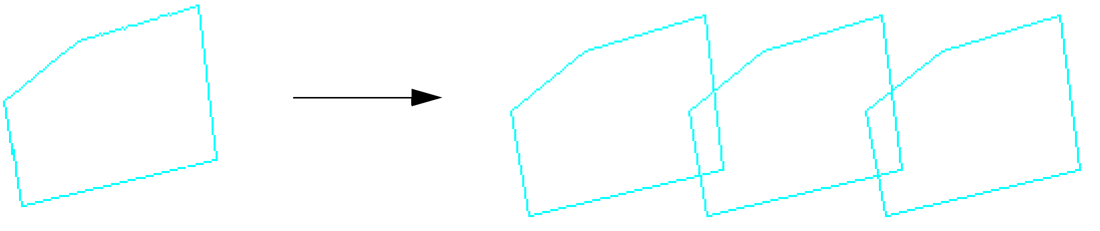
The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Rows	Input box		
number of rows			
Columns	Input box		
number of columns			
Row offset	Input box		
distance between the rows			
Column offset	Input box		
distance between the columns			
Rotation	angle box		
the angle to use for the rows and columns of copies			
Pick	button		
pick the super string to copy and when it is accepted the super string is copied			

STEPS:

Fill in the fields in the **Cad Array** panel, click **Pick** and select the super string to copy.


Cad Array
Rows 3
Columns 1
Row offset 1000
Column offset 1000
Rotation 0°
angle is valid
Pick
Finish
Help



15.20.4 CAD Rotate

Position of option on menu: **CAD =>Modify =>Rotate**

or by selection of appropriate icon from the toolbar. 

This option rotates a super string.

On selecting **Rotate**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

For **CAD Multipick Rotate**, go straight to Step 2.

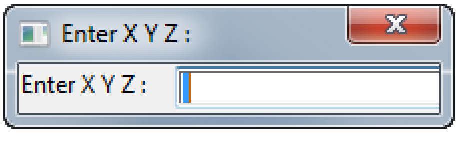
STEP 1:

Select the super string to rotate.

STEP 2:

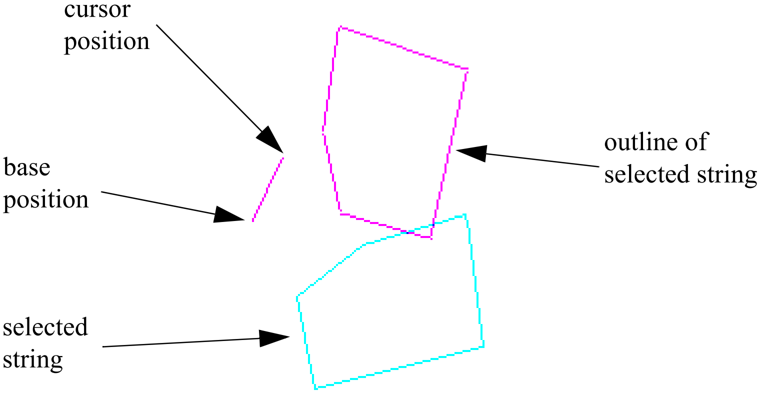
A base position is picked and then an outline of the selected string is drawn and rotated around the base position with the cursor, plus a line will be drawn from the base position to the current cursor position.

NOTE: If the <Spacebar> or any other key is pressed, an Enter X Y Z typed input box is displayed on the screen.



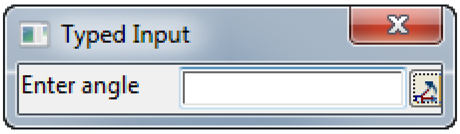
X, Y and Z are entered, separated by a space, and <Enter> accepts the values and removes the Enter X Y Z box from the screen. If values are not going to be typed in, clicking on **X** on the top right-hand corner of the Enter X Y Z box removes the box from the screen and the option returns to selecting the base position by moving the cursor.

Note: The base position is not required to lie on the string itself.



When the line has an angle of 0 degrees (bearing of 90 degrees), the outline will be on top of the selected string (*i.e.* there is no rotation). As the line rotates around the base position, the outline rotates around the base position. This rotation will continue until the final point is selected and accepted.

NOTE: If the <Spacebar> or any other key is pressed, an ***Enter Angle*** Typed Input box is displayed on the screen.

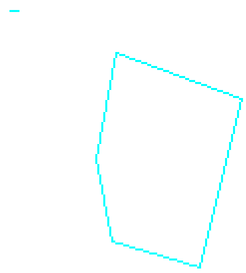


An angle (measured counter-clockwise from the positive X-axis and in 4.17.1 HP Notation) can be typed in and <Enter> will accept the value and remove the Typed Input box from the screen. If a value is not going to be typed in, clicking on **X** on the top right-hand corner of the Typed Input box removes the box from the screen and the option returns to selecting the rotation by moving the cursor.

STEP 3:

A final cursor position is selected which defines the rotation about the base position.

On accepting the final position, or after an angle has been entered, the selected string is rotated around the base position through the defined angle.



15.20.5 CAD Mirror

Position of option on menu: **CAD =>Modify =>Mirror**

or by selection of appropriate icon from the toolbar. 

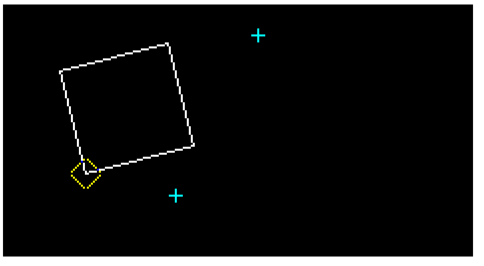
This option creates a mirrored image of a selected string.

On selecting **Mirror**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

For **CAD Multipick Mirror**, go straight to Step 2.

STEP 1:

Select a super string to mirror.

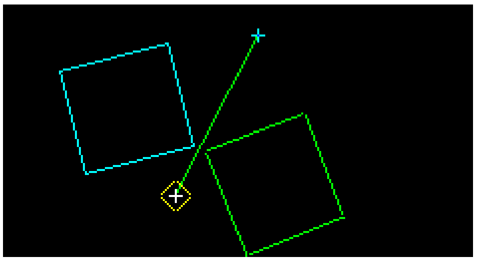


STEP 2:

Select the 1st point of the mirror axis.

The 1st point is selected with the mouse or entered via the keyboard. To enter a point with the keyboard, simply start typing or press the space bar to bring up the X Y Z Input box. Type the coordinates into the XYZ Input box and press the <Enter> key.

After the 1st point is accepted a line will be drawn from the 1st point to the position of the cursor. This line forms the mirroring axis. The outline of the reflected image will be dynamically displayed according to the changing cursor position.



STEP 3:

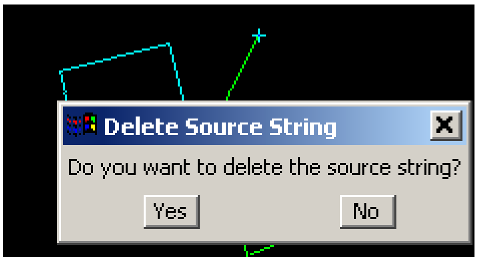
Select and accept the 2nd point of the mirror axis.

STEP 4:

For **CAD Mirror**, after the base point is accepted, the **Delete Source String** box will appear.

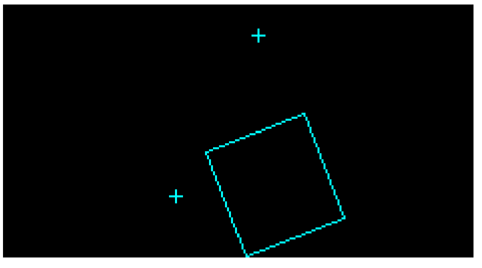
Select **Yes** if you want the original super string to be deleted when the new mirrored string is constructed.

Select **No** if you want the original super string to remain when the new mirrored string is constructed.

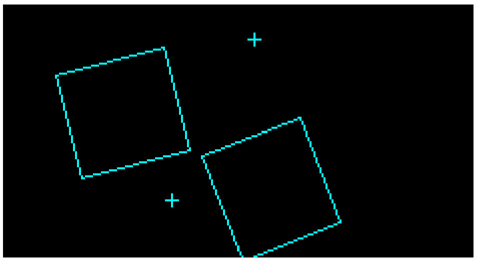


STEP 5:

The selected super string will be mirrored using the information given.



Delete the source string



Don't delete the source string

For **CAD Multipick Mirror**, the original strings are always deleted.

15.20.6 CAD Mirror X Axis

Position of option on menu: **CAD =>Modify =>Mirror X axis**

or by selection of appropriate icon from the toolbar.

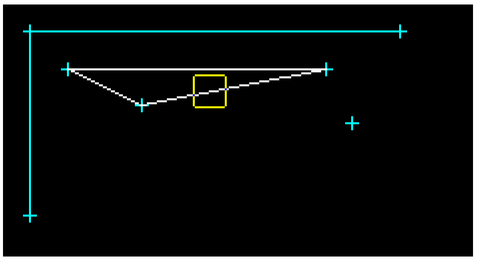
This option mirrors a string through the X axis.

On selecting **Mirror X Axis,** the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

For **CAD Multipick Mirror X axis**, go straight to Step 2.

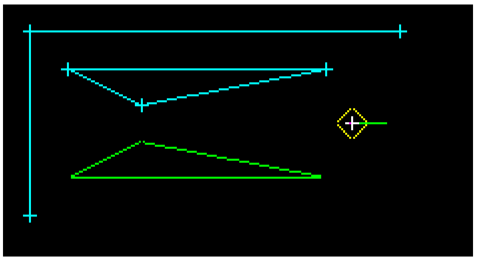
STEP 1: Picking a string

Select a super string to mirror though the X axis.



STEP 2:

After the string is accepted the position of the cursor on the Y axis determines the position of the mirror (which runs parallel to the X axis). As the cursor moves along the Y axis, a reflection of the selected string will be dynamically displayed. This will continue until a final position is selected, thus determining the mirror axis.



Select the final Y-position of the mirror X-axis.

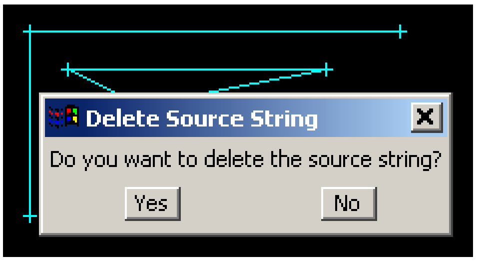
The final position is selected with the mouse or entered via the keyboard. To enter a base point with the keyboard, simply start typing or press the <Spacebar> to bring up the Enter X Y Z box. Type the coordinates into the Enter X Y Z box and press the <Enter> key.

STEP 3:

For **CAD Mirror X axis**, after the final position is accepted, a **Delete Source String** box will appear.

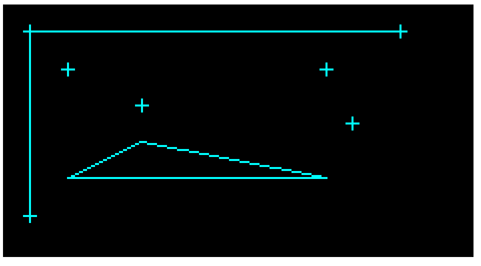
Select **Yes** if you want the original super string to be deleted when the new mirrored string is constructed.

Select **No** if you want the original super string to remain when the new mirrored string is constructed.

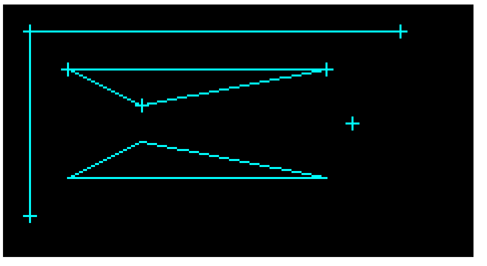


STEP 4:

The selected super string will be mirrored using the information given.



Delete the source string



Don't delete the source string

For **CAD Multipick Mirror X axis**, the original strings are always deleted.

15.20.7 CAD Mirror Y Axis

Position of option on menu: **CAD =>Modify =>Mirror Y Axis**

or by selection of appropriate icon from the toolbar.

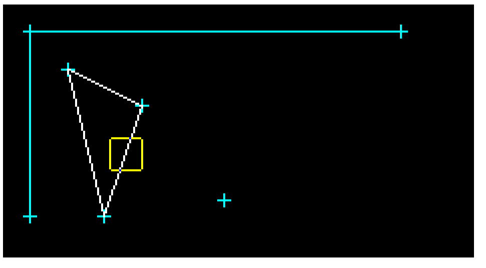
This option mirrors a string through the Y axis.

On selecting **Mirror Y Axis,** the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

For **CAD Multipick Mirror Y Axis**, go straight to Step 2.

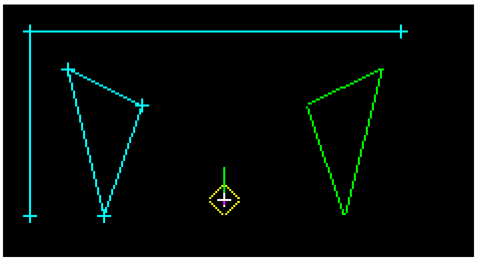
STEP 1:

Select a super string to mirror though the Y axis.



STEP 2:

After the string is accepted the position of the cursor on the X axis determines the position of the mirror (which runs parallel to the Y axis). As the cursor moves along the X axis, a reflection of the selected string will be dynamically displayed. This will continue until a final position is selected, thus determining the mirror axis.



Select the final X-position of the mirror Y-axis.

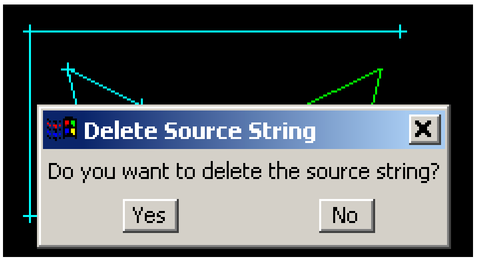
The final position is selected with the mouse or entered via the keyboard. To enter a base point with the keyboard, simply start typing or press the <Spacebar> to bring up the Enter X Y Z box. Type the coordinates into the Enter X Y Z box and press the <Enter> key.

STEP 3:

After the final position is accepted, a **Delete Source String** box will appear.

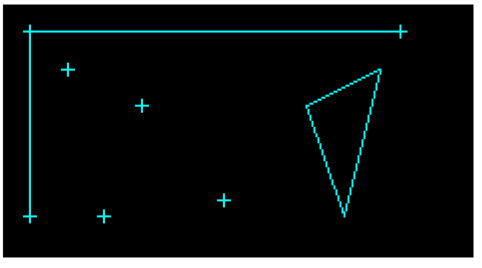
Select **Yes** if you want the original super string to be deleted when the new mirrored string is constructed.

Select **No** if you want the original super string to remain when the new mirrored string is constructed.

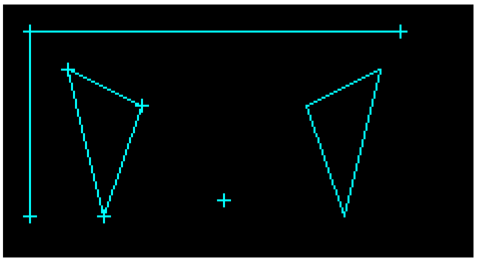


STEP 4:

The selected super string will be mirrored using the information given.



Delete the source string



Don't delete the source string

15.20.8 CAD Scale

Position of option on menu: **CAD =>Modify =>Scale**

or by selection of appropriate icon from the toolbar.

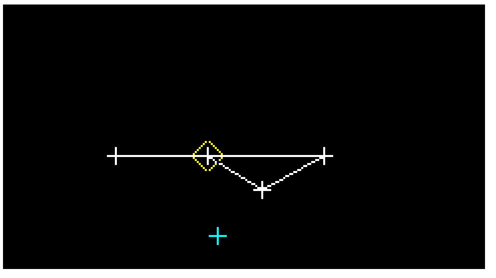
This option increases or decreases the size of a string by a given scale about a user selected origin. Go to 15.8.9 Scale About an Origin for the definition of scaling.

On selecting **Scale**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

For **CAD Multipick Scale**, go straight to Step 2.

STEP 1:

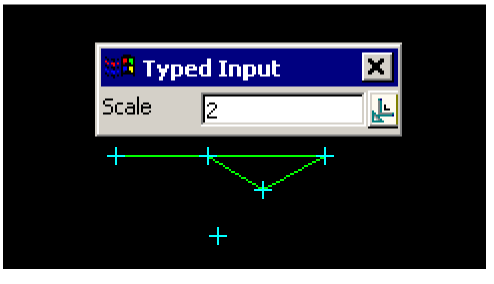
Select a super string to scale.

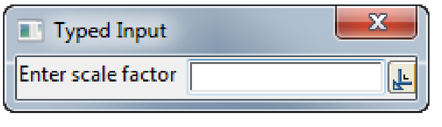


STEP 2:

After a string is accepted, the **Scale** Typed Input box will appear. Type the desired scale factor into the input box and press the <Enter> key. The browse button on the Input box can be used to define the scale by measuring existing elements.

Note: the Scale/Scale Factor can be positive or negative.





STEP 3:

After a scale value has been entered the user is required to select the position of the origin.

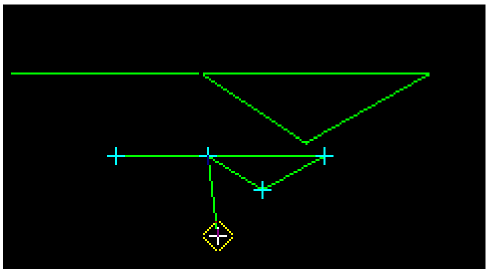
An origin position can be selected with the mouse or entered via the keyboard.

To enter an origin position with the keyboard, simply start typing or press the space bar to bring up the Enter X Y Z box. Type the coordinates into the Enter X Y Z box and press the <Enter> key.

If the position of the origin is given by the cursor, the position of the scaled highlighted string moves dynamically as the position of the origin changes as the cursor moves. The final selected origin position establishes the location of the scaled string relative to the original selected string.

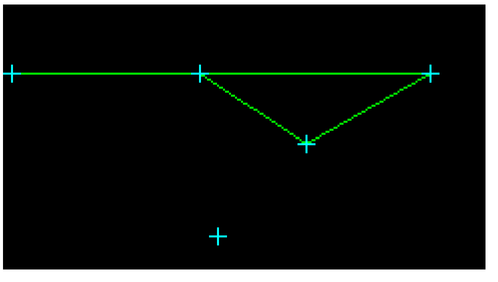
The dimensions of the highlighted string are defined by the Scale value.

For example, if you selected a scale of **2**, the distance between the cursor and a point on the highlighted string would be double the distance between the cursor and the corresponding point on the original super string.



STEP 4:

After the final position has been accepted, the original string will be removed and the newly scaled highlighted string constructed.



15.20.9 CAD Scale Dynamic

Position of option on menu: **CAD =>Modify =>Scale Dynamic**

or by selection of appropriate icon from the toolbar.

This option dynamically increases or decreases the size of a string.

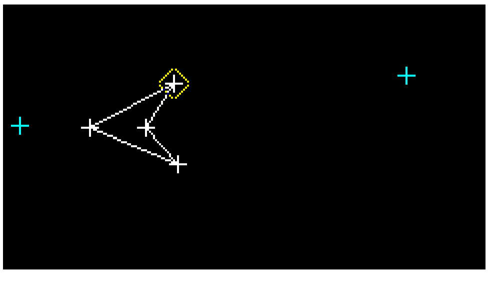
This option dynamically increases or decreases the size of a string about a user selected origin by selecting a point to define the scale factor. Go to 15.8.9 Scale About an Origin for the definition of scaling.

On selecting **Scale Dynamic**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

For **CAD Multipick Scale Dynamic**, go straight to Step 2.

STEP 1:

Select a super string to scale.

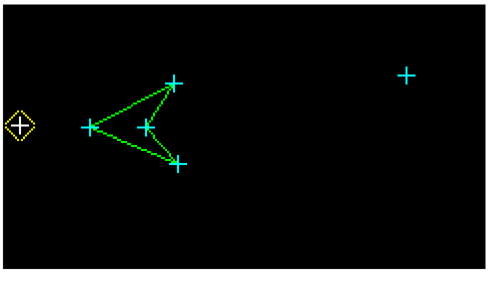


STEP 2:

An Origin point is required to be defined. An Origin point can be selected with the mouse or entered in via the keyboard.

To enter an Origin point with the keyboard, simply start typing or press the space bar to bring up the Enter X Y Z box. Type the coordinates into the Enter X Y Z box and press the <Enter> key.

To enter the origin by the cursor, simply select and accept the required origin position.



STEP 3:

For **CAD Scale Dynamic**:

After an Origin point has been accepted, the cursor then defines the Projection point.

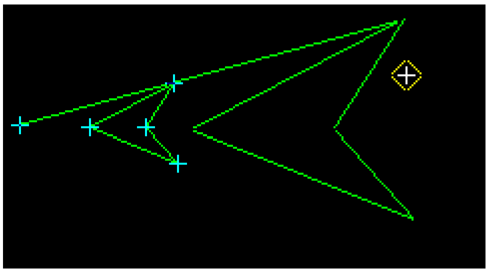
Note: the Projection Point is the cursor position dropped perpendicularly onto the line from the origin through the original point selected on the string.

The scale factor is taken to be the ratio of the distance from the projection point to the origin and the distance from the original point selected on the string to the origin.

That is,

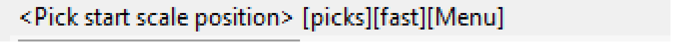
Scale factor = (the distance from the projection point to the origin) divided by (the distance from the original point selected on the string to the origin)

The highlighted string is projected from the Origin point to the Projection point (which is being defined by the cursor).

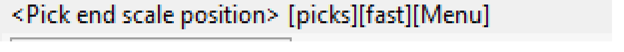


For **CAD Multipick Scale Dynamic**:

After an Origin point has been accepted, a Start Scale Position is selected.



After that the cursor then dynamically defines the End Scale position.



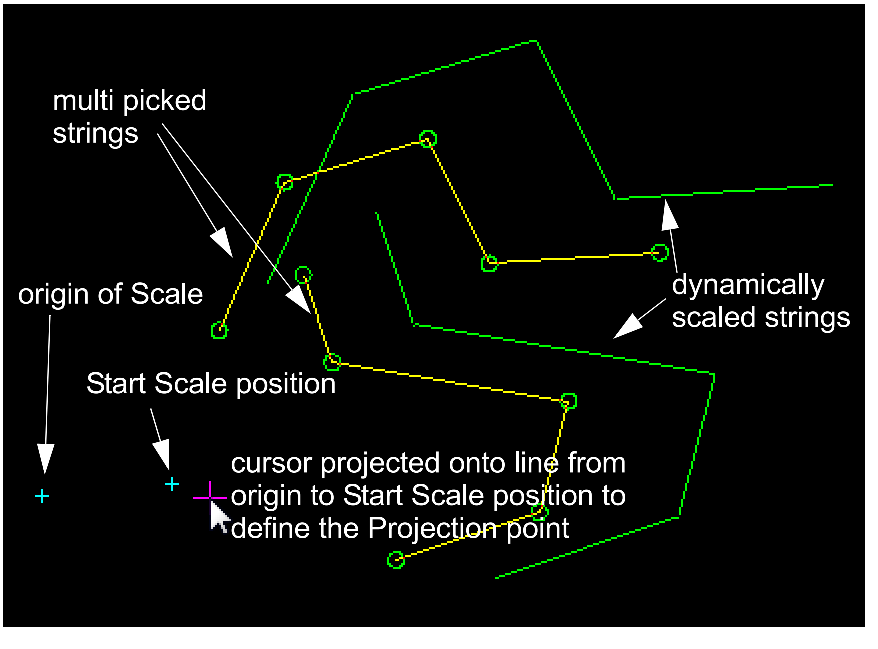
For Multipick, the Projection Point is the End Scale position dropped perpendicularly onto the line from the origin through the Start Scale position.

The scale factor is taken to be the ratio of the distance from the Projection Point to the origin and the distance from the Start Scale position to the origin.

That is,

Scale factor = (the distance from the Projection Point to the origin) divided by (the distance from the Start Scale position to the origin)

The highlighted strings are projected from the Origin point using the scale factor dynamically defined by the cursor (and hence the Projection point).



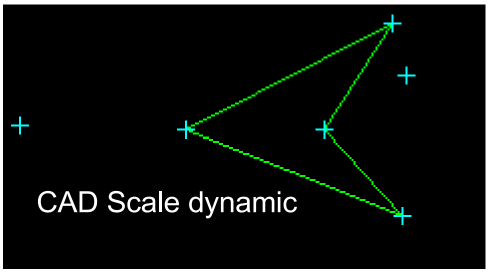
Hence for both the CAD Scale Dynamic and the CAD Multipick Scale Dynamic, the size and position of the highlighted string will move dynamically as the position of the Projection point moves with the cursor.

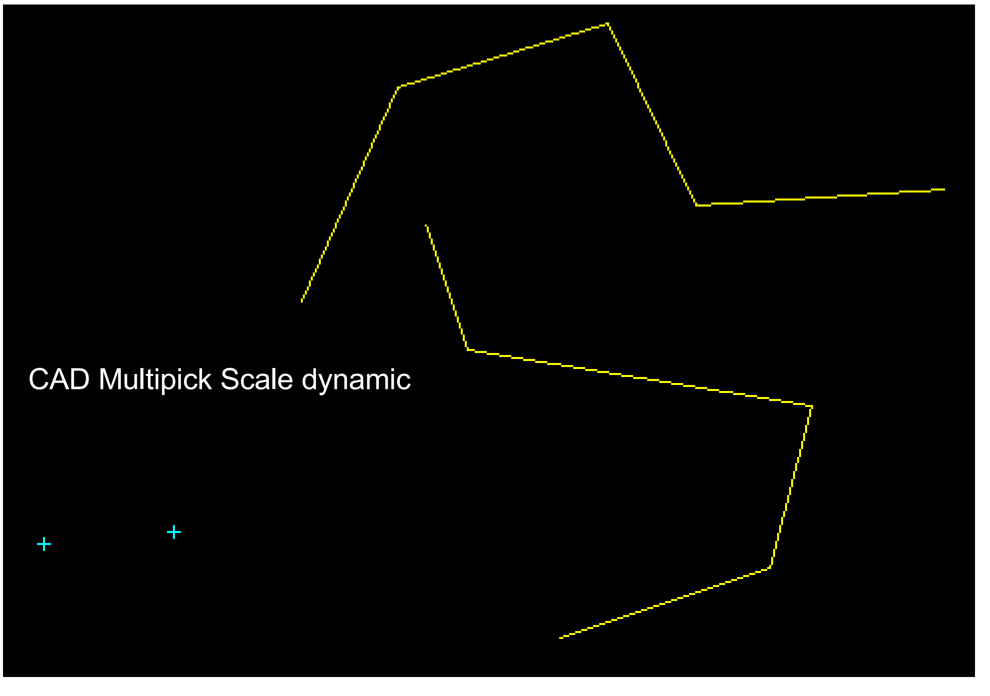
The highlighted string will increase in size if the cursor moves the Projection point further from the Origin than the original string point/Start Scale position, or decrease in size if the cursor moves the Projection point closer to the Origin than the original string point/Start Scale position.

This will continue until a Projection point is accepted.

STEP 4:

After the Projection point has been accepted, the new scaled strings are created and the original strings deleted.



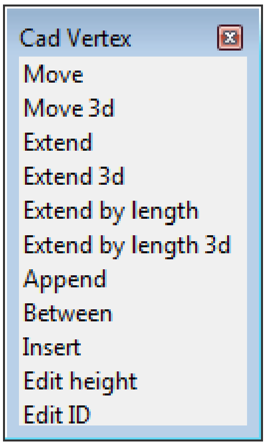


15.21 CAD Vertex

Position of option on menu: CAD =>Vertex

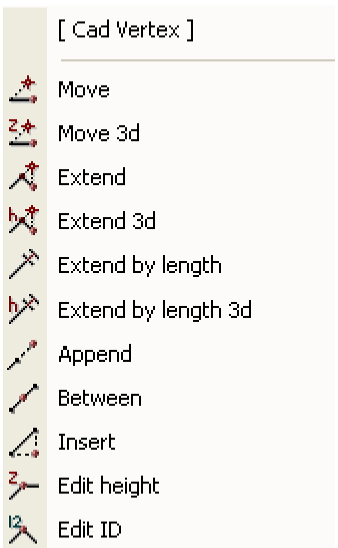
This section of documentation is a work in progress and will be updated in subsequent releases.

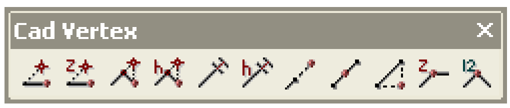
The **Cad Vertex** walk-right menu is



Menu of Options to Move Vertices

move a vertex
move a vertex and modify its height
move a vertex along its segment - keep the same height
move a vertex along its segment - interpolate/extrapolate the height
move a vertex by a given chainage distance along a segment - keep height
move a vertex a given chainage dist along a segment - extrapolate height
append a vertex to an existing string
create a vertex on the segment joining two adjacent vertices of a string
insert a vertex between two adjacent vertices of a string
edit the height of a vertex
edit the point (vertex) ID





These options allow the editing of vertices on super strings.

For the option *Move*, go to 15.21.1 Move a Vertex while Keeping its Height
Move 3d 15.21.2 Move a Vertex and Modify its Height
Extend 15.21.3 Extend a Vertex while Keeping its Height
Extend 3d 15.21.4 Extend Segment and Extrapolating Heights
Extend by length 15.21.5 Extend Segment by Length and Keeping Height
Extend by length 3d 15.21.6 Extend Segment by Length and Extrapolating Heights
Append 15.21.7 Append a Vertex
Between 15.21.8 New Vertex on a Segment (Between Two Vertices)

Insert	15.21.9 Insert a Vertex
Edit height	15.21.10 Change Height of a Vertex
Edit ID	15.21.11 Edit ID

15.21.1 Move a Vertex while Keeping its Height

Position of option on menu: **CAD =>Vertex =>Move**

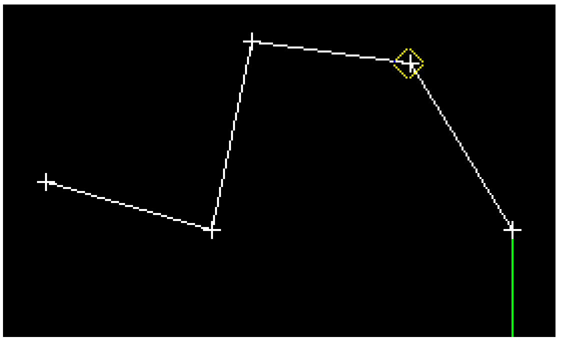
or by selection of appropriate icon from the toolbar. 

This option moves the vertex of a super string. The height of the vertex is not modified.

On selecting **Move**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

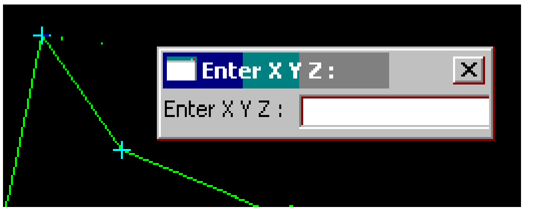
STEP 1:

A vertex on a super string is selected and accepted.



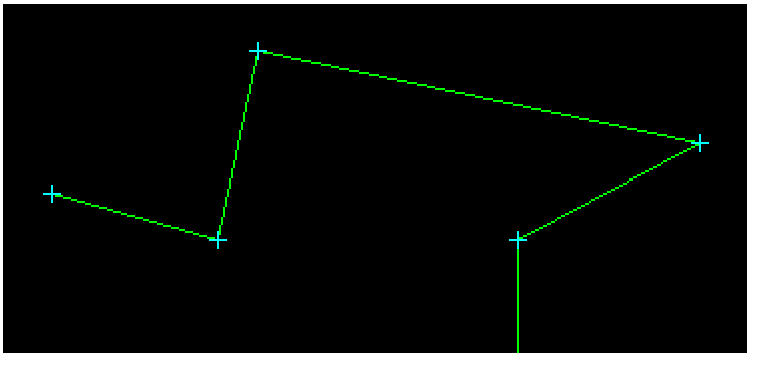
STEP 2:

The position of the selected vertex moves with the cursor until the final position of the vertex is selected and accepted. Alternatively, the user can type in the final coordinate of the vertex by the using the **Enter XYZ** box. This box is brought up by simply typing a value or by pressing the space bar.



STEP 3:

The selected vertex is moved to the new position.



15.21.2 Move a Vertex and Modify its Height

Position of option on menu: **CAD =>Vertex =>Move 3d**

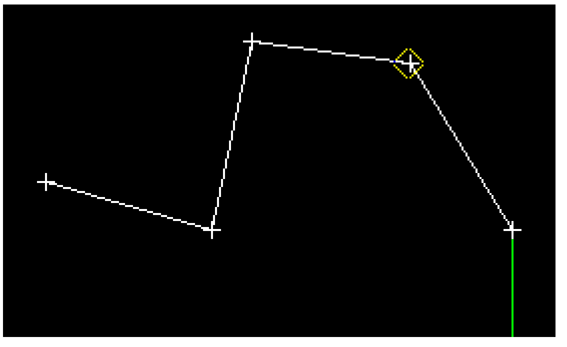
or by selection of appropriate icon from the toolbar.

This option moves the vertex of a super string and the height of the vertex is taken from the height of the final selected position.

On selecting **Move 3d**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

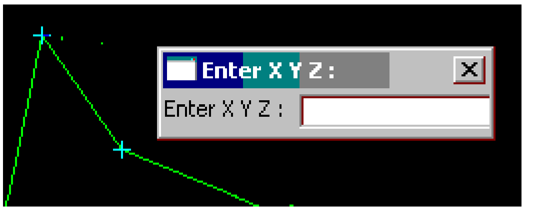
STEP 1:

A vertex on a super string is selected and accepted.



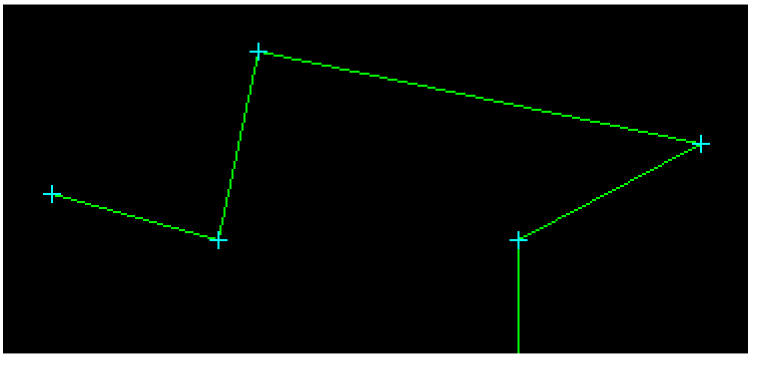
STEP 2:

The position of the selected vertex moves with the cursor until the final position of the vertex is selected and accepted. Alternatively, the user can type in the final coordinate of the vertex by the using the **Enter XYZ** box. This box is brought up by simply typing a value or by pressing the space bar.



STEP 3:

The selected vertex is moved to the new position and takes on the height of the string it is snapped to, or the Z value entered into the **Enter XYZ** box.



15.21.3 Extend a Vertex while Keeping its Height

Position of option on menu: **CAD =>Vertex =>Extend**

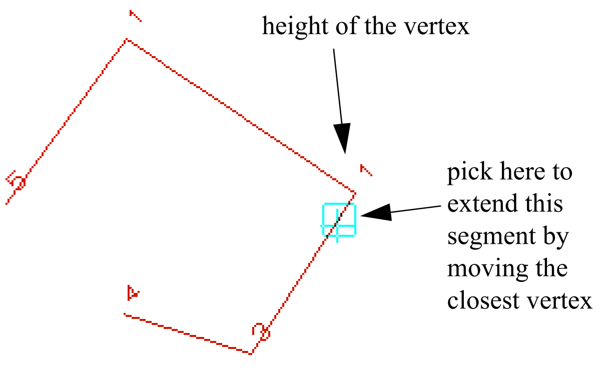
or by selection of appropriate icon from the toolbar.

This option extends (or shortens) a segment by moving a vertex along the segment joining the vertex to its neighbouring vertex. The height of the vertex is not modified.

On selecting **Extend**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

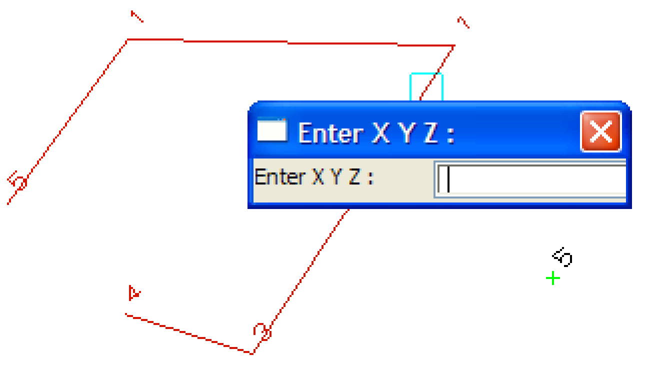
STEP 1:

Most vertices are on two segments so it is necessary to indicate which segment is to be extended, This is done by selecting **not the vertex**, but by selecting **on the segment** that is to be extended, close to the vertex to be moved.



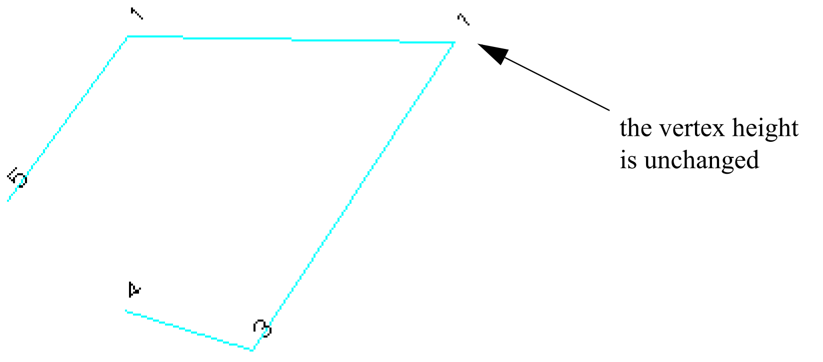
STEP 2:

As the cursor is moved, the string is redrawn reflecting the changing position of the moved vertex. The final position of the vertex is selected and accepted. Alternatively, the user can type in the final coordinate of the moved vertex into the **Enter XYZ** box. This box is brought up by simply typing a value or by pressing the space bar.



STEP 3:

The selected vertex is moved to the new position. The height of the vertex is not changed.


the vertex height
is unchanged

15.21.4 Extend Segment and Extrapolating Heights

Position of option on menu: CAD =>Vertex =>Extend 3d

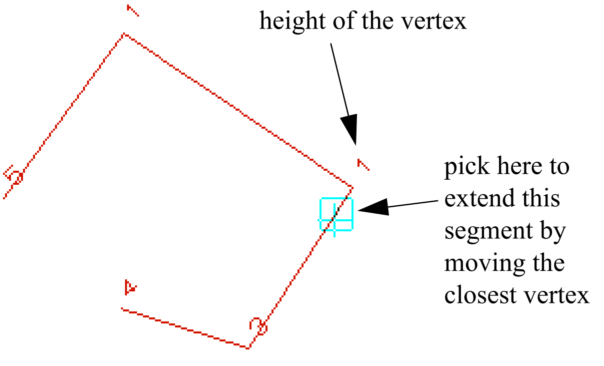
or by selection of appropriate icon from the toolbar.

This option extends (or shortens) a segment by moving a vertex along the segment joining the vertex to its neighbouring vertex. The height of the vertex is extrapolated.

On selecting **Extend 3d**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

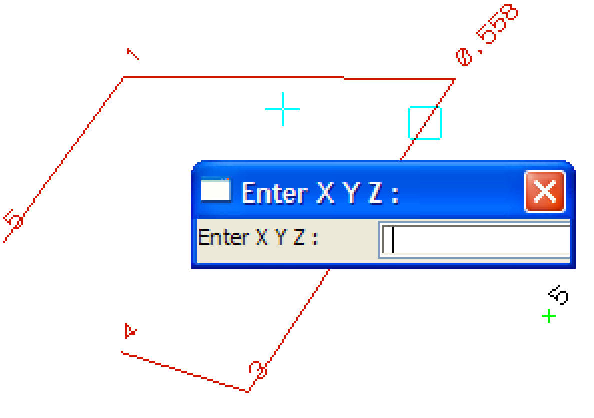
Most vertices are on two segments so it is necessary to indicate which segment is to be extended, This is done by selecting **not the vertex**, but by selecting **on the segment** that is to be extended, close to the vertex to be moved.



STEP 2:

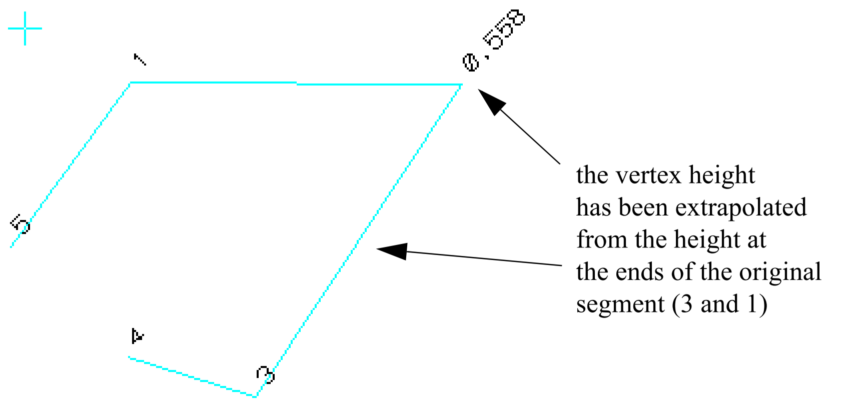
As the cursor is moved, the string is redrawn reflecting the changing position of the moved vertex. The final position of the vertex is selected and accepted. Alternatively, the user can type in the final coordinate of the moved vertex into the **Enter XYZ** box. This box is brought up by simply typing a value or by pressing the space bar.

The point selected is moved to the cursor position in a rubber banding fashion. The final position of the point is selected and accepted. Alternatively, the user can enter the final coordinate value of the moved point by the Enter XYZ box. This box can be brought into view by simply typing a value or by pressing the space bar.



STEP 3:

The selected vertex is moved to the new position. The height of the vertex has been extrapolated from the heights at either end of the original segment.



15.21.5 Extend Segment by Length and Keeping Height

Position of option on menu: **CAD =>Vertex =>Extend by length**

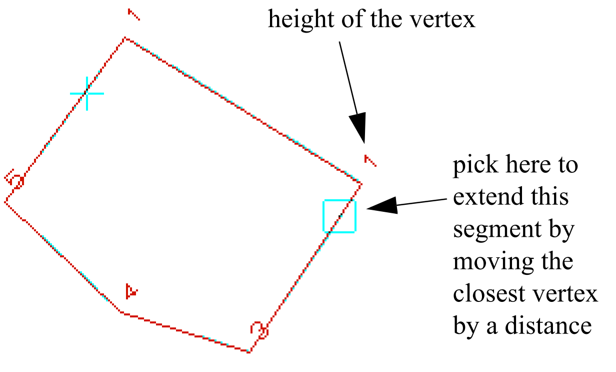
or by selection of appropriate icon from the toolbar.

This option extends (or shortens) a segment by moving a vertex along the segment joining the vertex to its neighbouring vertex by a user defined distance. The height of the vertex is not modified.

On selecting **Extend by length**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

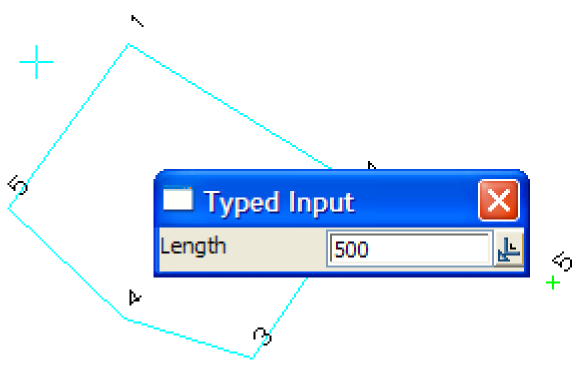
STEP 1:

Most vertices are on two segments so it is necessary to indicate which segment is to be extended, This is done by selecting **not the vertex**, but by selecting **on the segment** that is to be extended, close to the vertex to be moved.



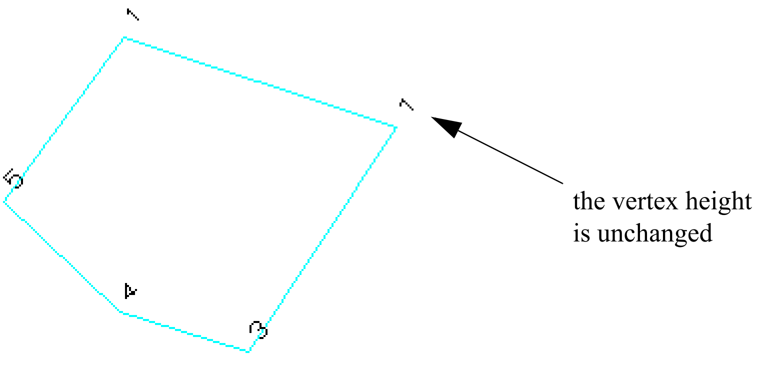
STEP 2:

The user enters the distance to extend the segment. This value is positive in the direction that the string was created.



STEP 3:

The point selected is moved to the new position and the string topology maintained.


the vertex height
is unchanged

15.21.6 Extend Segment by Length and Extrapolating Heights

Position of option on menu: **CAD =>Vertex =>Extend by length 3d**

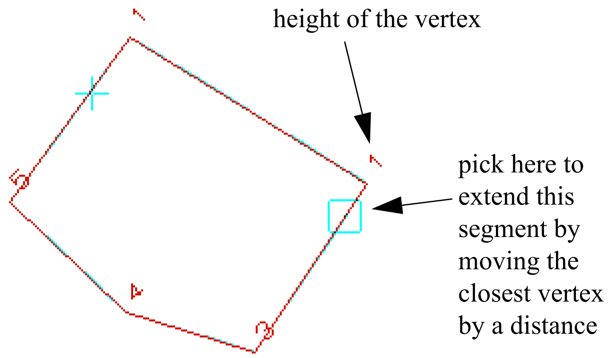
or by selection of appropriate icon from the toolbar.

This option extends (or shortens) a segment by moving a vertex along the segment joining the vertex to its neighbouring vertex by a user defined distance. The height of the vertex is extrapolated modified.

On selecting **Extend by length 3d**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

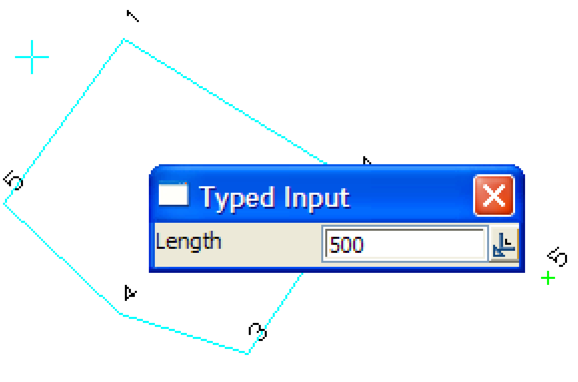
STEP 1:

Most vertices are on two segments so it is necessary to indicate which segment is to be extended, This is done by selecting **not the vertex**, but by selecting **on the segment** that is to be extended, close to the vertex to be moved.



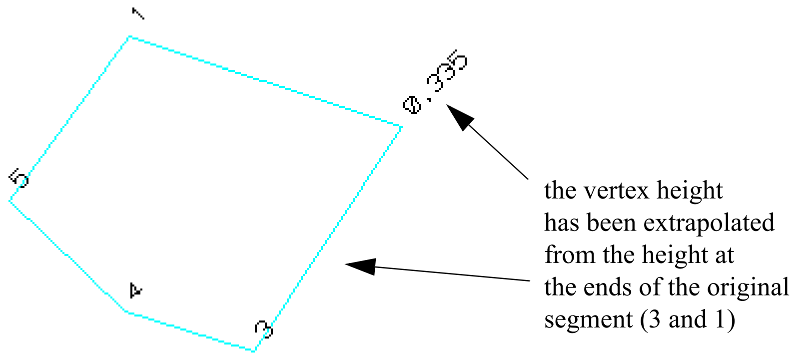
STEP 2:

The user enters the length value for which the point is to be extended by. This value is positive in the direction that the string was created.



STEP 3:

The point selected is moved to the new position and the height adjusted according to the grade of the extended segment.


1
0.335
5
4
3
the vertical height
has been extrapolated
from the height at
the ends of the original
segment (3 and 1)

15.21.7 Append a Vertex

Position of option on menu: **CAD =>Vertex =>Append**

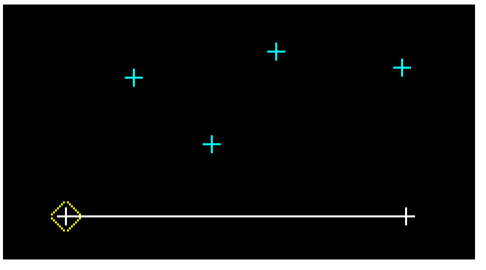
or by selection of appropriate icon from the toolbar. 

This option adds vertices to an existing super string.

On selecting **Append**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

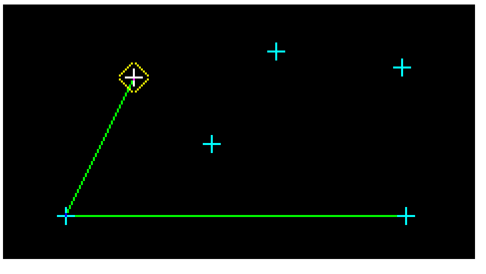
STEP 1:

Select a string by picking near the end of the string that vertices are to be added to (added to the beginning of the string - prepend, or added to the end of the string - append).



STEP 2:

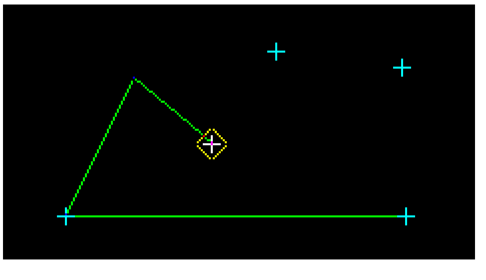
After the string is accepted, a line is drawn from the end of the string to the current cursor position.



Select the position for the new vertex.

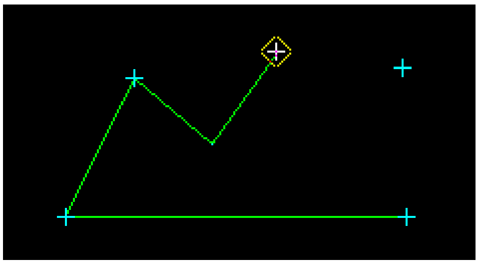
STEP 3:

After the new vertex is accepted, a line is drawn from that vertex to the current cursor position. That is, the process of adding points is repeated.

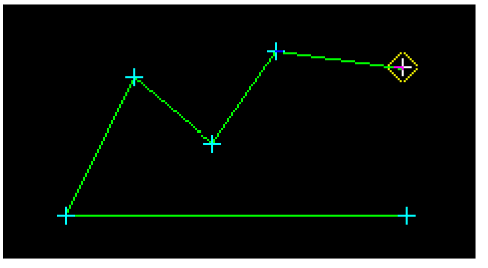


STEP 4:

This step should be repeated until the required number of vertices has been added.



STEP 5:



STEP 6:

After the required number of vertices have been added, the option is terminated for this string by pressing the <Esc> key.

15.21.8 New Vertex on a Segment (Between Two Vertices)

Position of option on menu: **CAD =>Vertex =>Between**

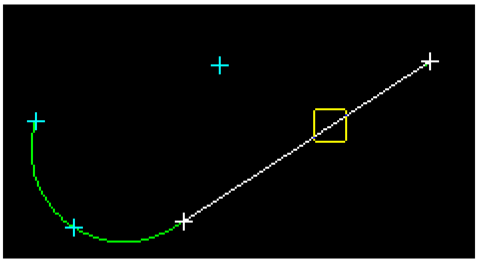
or by selection of appropriate icon from the toolbar.

This option adds a vertex to a string but it is constrained to be on an existing segment of the string. The height of the vertex is interpolated from the heights at the end of the segment.

On selecting **Between**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

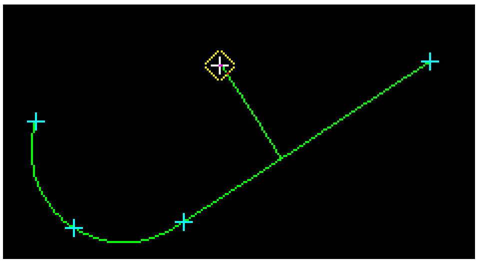
STEP 1:

Select the string on the segment of the string that is to have a vertex added to it. The new vertex will be constrained to be on the segment.



STEP 2:

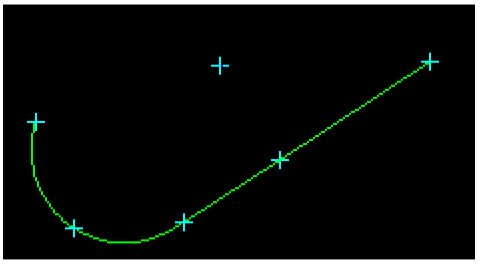
After the segment is accepted, a perpendicular line is drawn from the selected segment of the string to the position of the cursor.



STEP 3:

A position is selected with the mouse or typed in via the keyboard and the new vertex is created by **dropping** the position **perpendicularly** onto the segment.

Note - to enter via the keyboard, simply start typing or press the space bar to bring up the **Enter XYZ** Input box. Type the coordinates into the XYZ Input box and press the enter key.



15.21.9 Insert a Vertex

Position of option on menu: **CAD =>Vertex =>Insert**

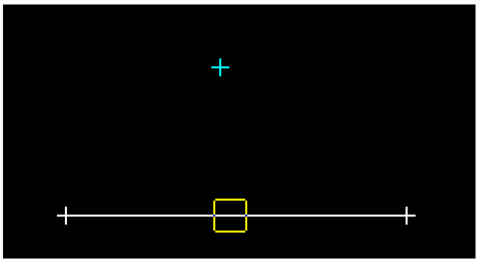
or by selection of appropriate icon from the toolbar.

This option inserts a vertex into to a string between two adjacent vertices but it is constrained to be on the existing segment joining the vertices. The height of the vertex is taken from the final selected position.

On selecting **Insert**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

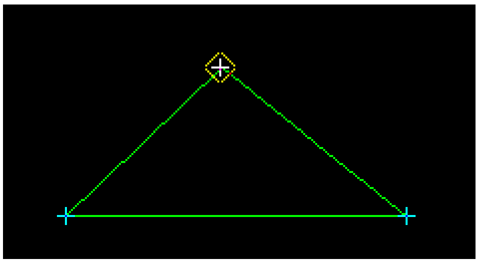
STEP 1:

Select the string on the segment of the string that is to have a vertex added to it. The new vertex is ***not*** constrained to be on the segment.



STEP 2:

After a string is accepted, two lines will be drawn from the cursor position to either end of the selected segment.

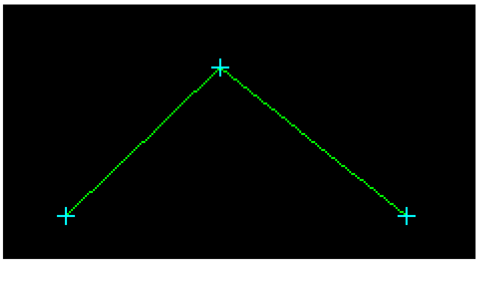


STEP 3:

A final position is selected with the mouse or typed in via the keyboard and the new vertex is created at the final position.

Once the new vertex is created the original segment is no longer drawn.

Note - to enter via the keyboard, simply start typing or press the space bar to bring up the **Enter XYZ** Input box. Type the coordinates into the XYZ Input box and press the enter key.



15.21.10 Change Height of a Vertex

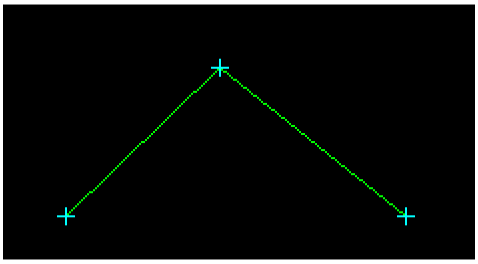
Position of option on menu: **CAD =>Vertex =>Edit height**

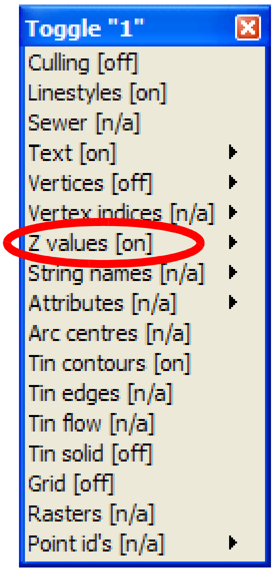
or by selection of appropriate icon from the toolbar.

This option changes the height (z value) of a vertex.

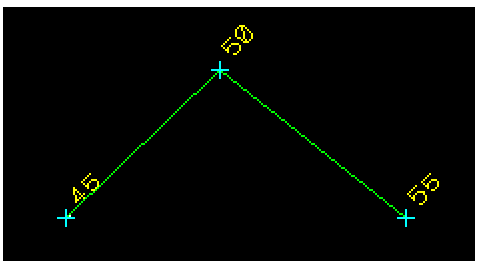
On selecting **Edit height**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:





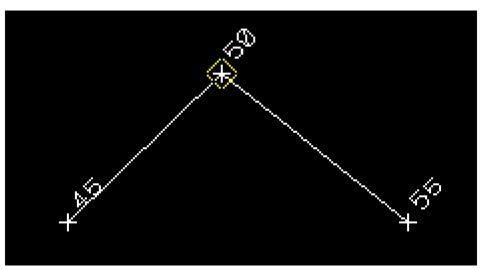
To see the effects of this option, use the **Toggle** menu to toggle on **Z values**. The height of each vertex (z value) is then displayed on the plan view so any modification to the height will be visible.



STEP 2:

Select a vertex with either the mouse or via the keyboard.

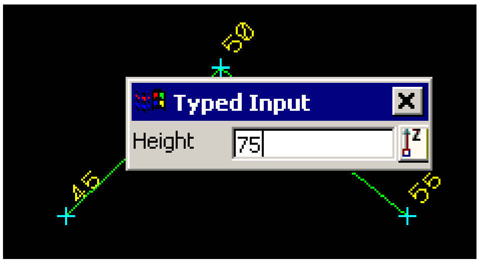
Note - to enter via the keyboard, simply start typing or press the space bar to bring up the **Enter XYZ** Input box. Type the coordinates into the **Enter XYZ** Input box and press the <Enter> key.



STEP 3:

After a point is accepted, the **Height** input box will appear with the current height displayed in it.

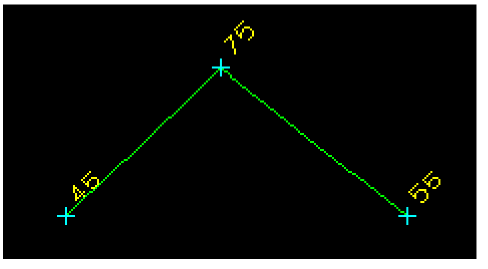
Type the new height (z value) of the vertex into the **Height** input box and press the <Enter> key. The browse button on the Input box can also be used to define the point's height.



STEP 4:

The height (z value) of selected point will have changed to the new value.

If z values have been turned on, the new value will be displayed in the view.



15.21.11 Edit ID

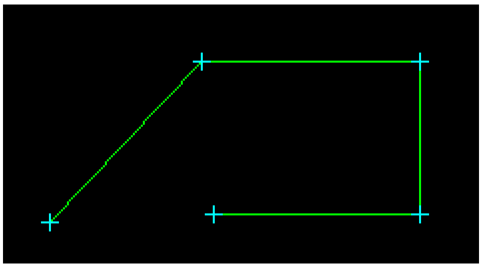
Position of option on menu: **CAD =>Vertex =>Edit ID**

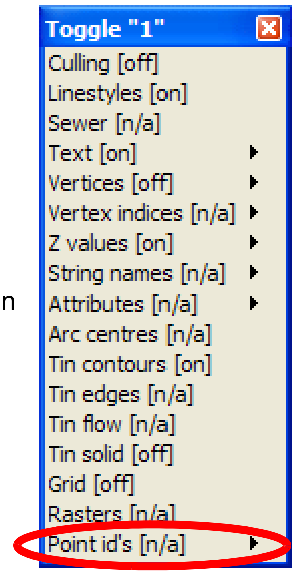
or by selection of appropriate icon from the toolbar. 

This option changes the point id of the vertex of a super string.

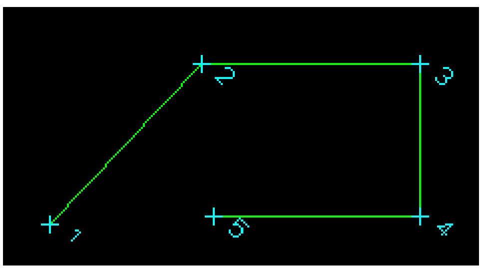
On selecting **Edit ID**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:





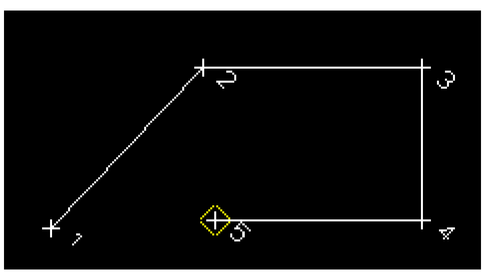
To see the effects of this option, use the **Toggle** menu to toggle on **Point id's**. The Point id of each vertex is then displayed on the plan view so any modification to the point id will be visible.



STEP 2:

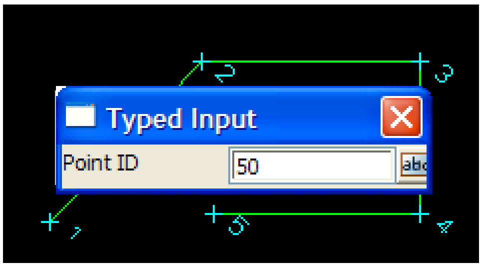
Select a vertex with either the mouse or via the keyboard.

Note - to enter via the keyboard, simply start typing or press the space bar to bring up the **Enter XYZ** Input box. Type the coordinates into the **Enter XYZ** Input box and press the Enter key.



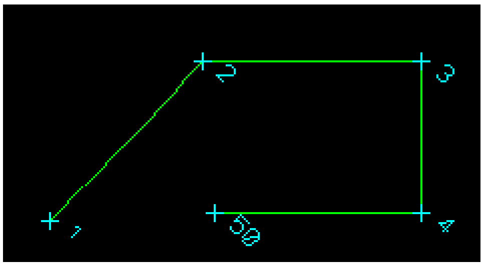
STEP 3:

After the vertex is accepted, the **Point ID** input box will appear. Type the point id into the input box and press the Enter key.



STEP 4:

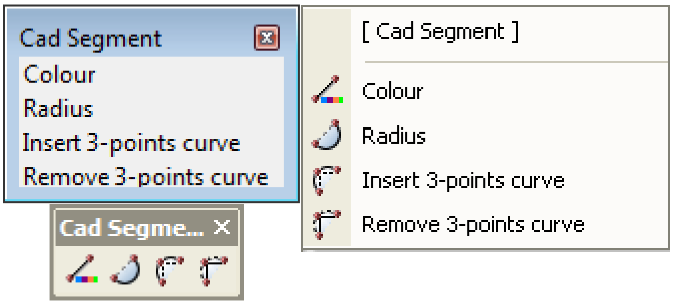
The point id of selected vertex is changed to the new value. If Point id's have been turned on, the new value will be displayed in the view.



15.22 CAD Segment

Position of option on menu: **CAD =>Segment**

The **Cad Segment** walk-right menu is



For the option *Colour*, go to 15.22.1 Colour
Radius 15.22.2 Radius
Insert 3-points curve 15.22.3 Insert 3-points curve
Remove 3-points curve 15.22.4 Remove 3-points curve

15.22.1 Colour

Position of option on menu: **CAD =>Cad Segment =>Colour**

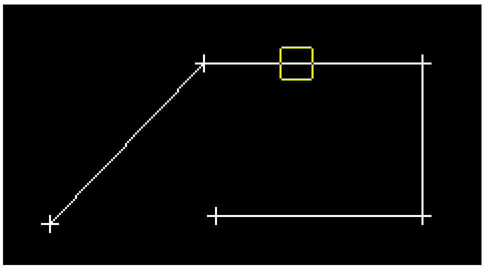
or by selection of appropriate icon from the toolbar. 

This option changes the colour of a selected segment.

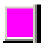
On selecting **Segment colour**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

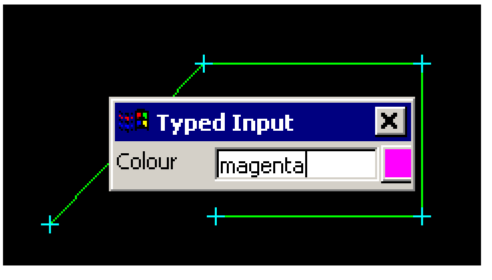
STEP 1:

Select the segment of the super string that is to have a colour change.



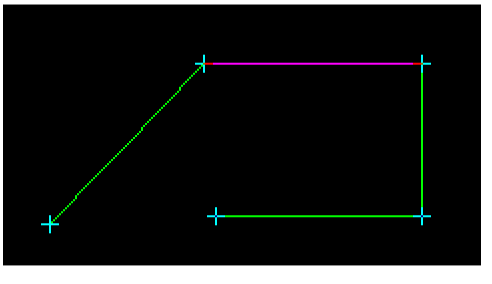
STEP 2:

After the segment is accepted, the **Colour** Input box will appear. Type the desired colour into the Input box and press the enter key. Alternatively you can select a colour from a list by clicking the browse button  on the Input box.



STEP 3:

The colour of the selected segment is drawn in the new colour.



15.22.2 Radius

Position of option on menu: **CAD =>Cad Segment =>Radius**

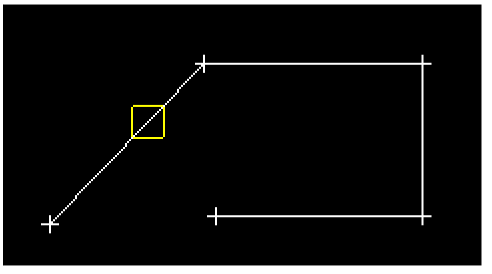
or by selection of appropriate icon from the toolbar. 

This option changes the radius of a segment.

On selecting **Segment radius**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

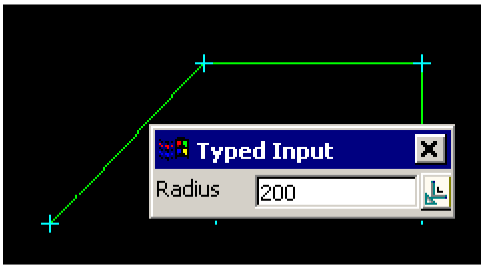
STEP 1:

Select the segment of the super string that is to have a change of radius.



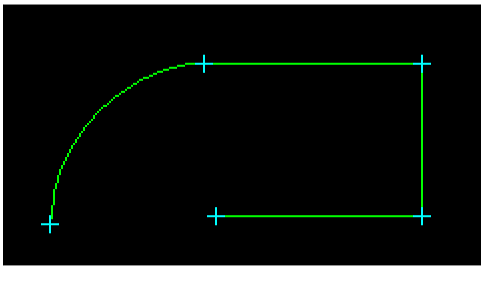
STEP 2:

After the segment is accepted, the **Radius** box will appear. Type the desired radius into the Input box and press the enter key. The browse button  on the Input box can be used to define the radius by measuring existing elements.



STEP 3:

The selected segment is then drawn with the new radius.



15.22.3 Insert 3-points curve

Position of option on menu: **CAD =>Cad Segment =>Insert 3-points curve**

or by selection of appropriate icon from the toolbar.

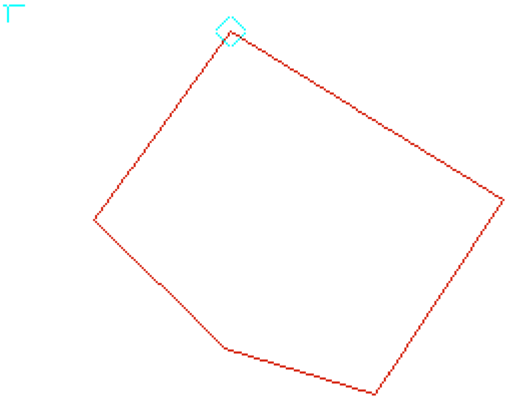
This option allows the creation of a 3 point curve using 2 existing segments.

On selecting **Insert 3pt curve**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

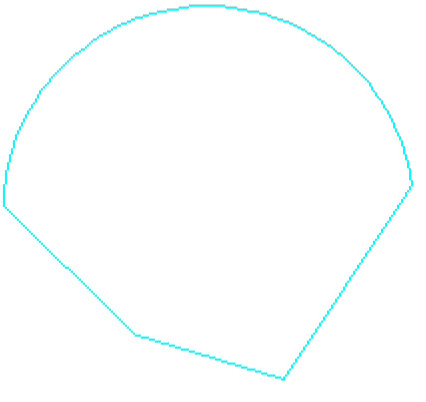
Select a vertex of a super string a vertex with a segment on either side, or segment of a super string.

If a **vertex** is selected, that vertex will be the middle vertex of the constructed 3 point curve. If a **segment** is selected, the closest vertex will become the **middle** vertex of the 3 point curve.



STEP 2:

After the vertex or segment is selected and accepted, an arc is through the three vertices is calculated and its radius used for the tow segments on either side of the vertex.



15.22.4 Remove 3-points curve

Position of option on menu: CAD =>Cad Segment =>Remove 3-points curve

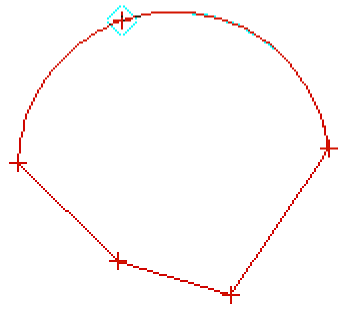
or by selection of appropriate icon from the toolbar. 

This option removes the arcs from either side of a selected vertex. If a segment is selected, the closest vertex is used and the arcs removed from either side of the vertex,

On selecting **Remove 3pt curve**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

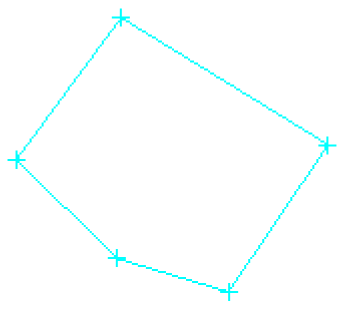
STEP 1:

Select a vertex of a super string with arcs on one or two sides of the vertex.



STEP 2:

After the vertex is selected and accepted, arc on either side will removed (the radius is set to zero). Two line segments are then drawn.

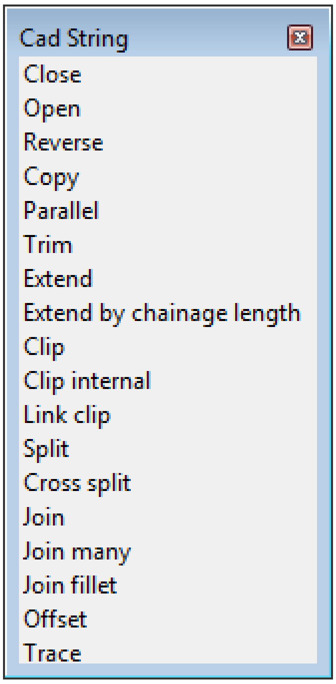


15.23 CAD Edit Strings

Position of option on menu: **CAD =>String**

These options are currently under development.

The **Cad String** walk-right menu is



Menu of Options to Edit Strings

close a string
open a string
reverse the direction of a string
copy a string
parallel a string
trim a string back to a selected string
extend a string until it cuts a selected string
extend a string by a typed chainage length
clip a string and keep the outer strings
clip a string and keep the internal string
deletes the selected link from the string (making two strings)
split a string
splits two strings where they first intersect each other
join two strings
join many strings
fillet and join
string parallel and more
trace over parts of strings to form a new string

For the option *Close*, go to 15.23.1 Close String
Open 15.23.2 Open String
Reverse 15.23.3 Reverse a String
Copy 15.23.4 Copy a String
Parallel 15.23.5 Parallel a String
Trim 15.23.6 Trim a String
Extend 15.23.7 Extend a String
Extend by chainage length 15.23.8 Extend String by Chainage Length
Clip 15.23.9 Clip String
Clip internal 15.23.10 Delete an Internal Section of a String
Link clip 15.23.11 Delete a Segment
Split 15.23.12 Split String
Cross split 15.23.13 Split Crossing Strings
Join 15.23.14 Join String
Join many 15.23.15 Join String Many
Join fillet 15.23.16 Join Fillet
Offset 15.23.17 Offset
Trace 15.23.18 Trace

15.23.1 Close String

Position of option on menu: **CAD =>String=>Close**

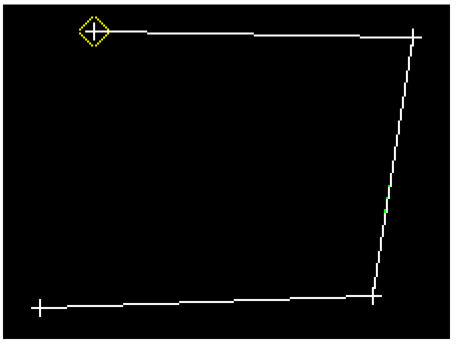
or by selection of appropriate icon from the toolbar. 

This option closes an open string.

On selecting **Close**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

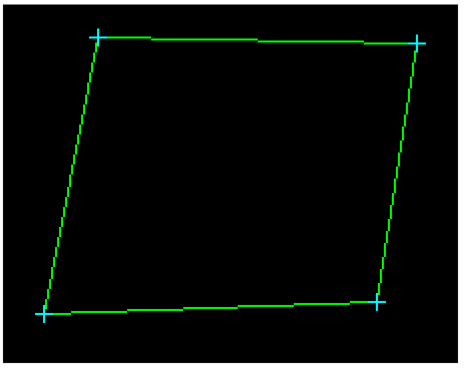
An open string is selected and accepted.



STEP 2:

The string is closed

After closing of a string, the option restarts so that another string can be selected. Picking Cancel from the Pick Ops menu, hitting <esc> or selecting another option terminates the option.



15.23.2 Open String

Position of option on menu: **CAD =>String=>Open**

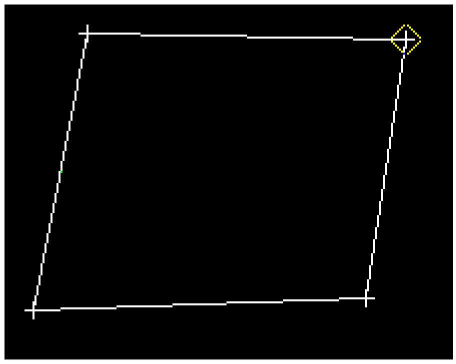
or by selection of appropriate icon from the toolbar. 

This option opens a closed string.

On selecting **Open**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

A closed string is selected and accepted.

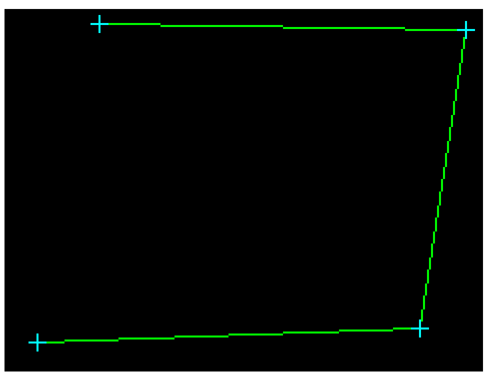


STEP 2:

The string is opened by deleting the last segment of the closed string.

Note - this may not be where the string was selected.

After opening of a string, the option restarts so that another string can be selected. Picking Cancel from the Pick Ops menu, hitting <esc> or selecting another option terminates the option.



15.23.3 Reverse a String

Position of option on menu: **CAD =>String=>Reverse**

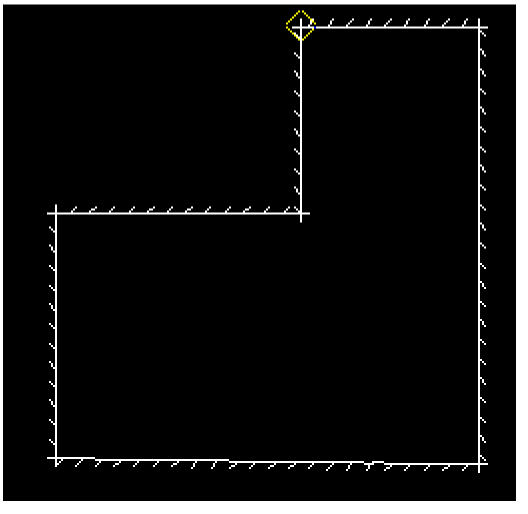
or by selection of appropriate icon from the toolbar. 

This option reverses the direction of a string

On selecting **Reverse**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

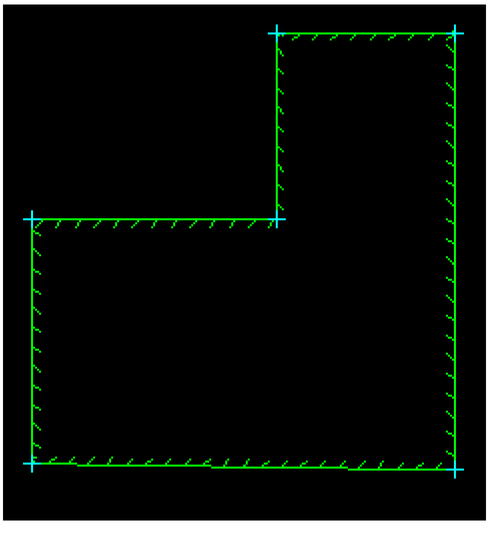
STEP 1:

A string is selected and accepted.



STEP 2:

The string is reversed.



After reversing of a string, the option restarts so that another string can be selected. Picking **Cancel** from the **Pick Ops** menu, hitting <esc> or selecting another option terminates the option.

15.23.4 Copy a String

Position of option on menu: **CAD =>String=>Copy**

or by selection of appropriate icon from the toolbar. 

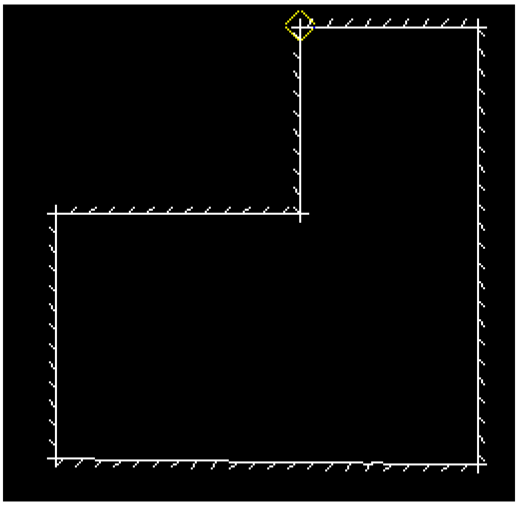
This option copies a string. The copied strings takes its attributes from the **Cad Control Bar.**

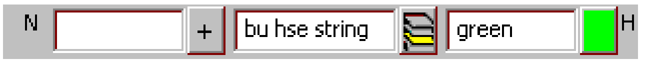
Copy is particularly helpful in backing up strings.

On selecting **Copy**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

A string is selected and accepted. The copied string will take on the values in the control bar at the time of accepting.





STEP 2:

A copy of the original string will be saved into the model **bu hse string**.

15.23.5 Parallel a String

Position of option on menu: **CAD =>String=>Parallel**

or by selection of appropriate icon from the toolbar.

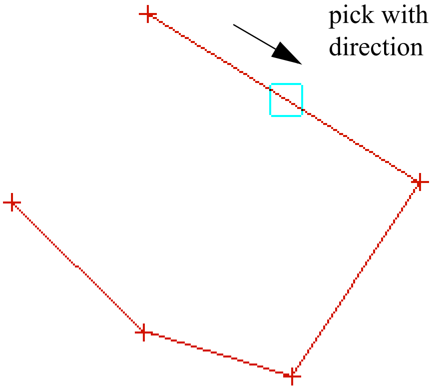
This option parallel a string by a given to a selected string.

Note that if there are arcs in the string, the string can not be paralleled by a distance that would collapse the arcs.

On selecting **Parallel**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

The string to parallel is picked with direction. The direction determines what it means to parallel to the right or left of the string.

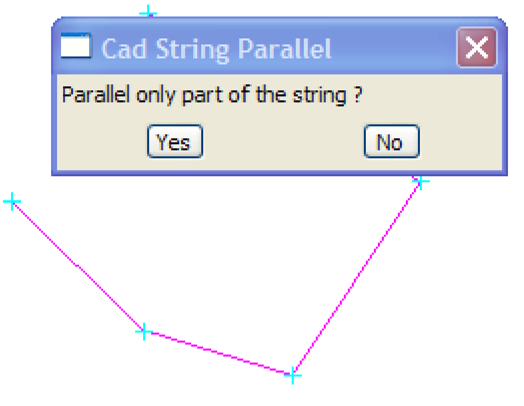


STEP 2:

The parallel option can be applied to **all** of the selected string or only **part** of the string.

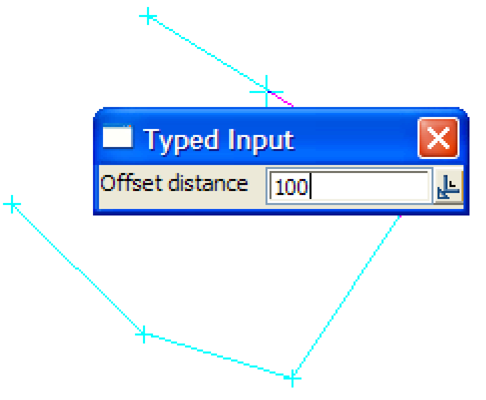
After the string is selected, a panel is displayed asking if only part of the string is to be paralleled.

If only part of the string is to be paralleled (i.e. **yes** selected) the user picks the start and end positions of where the string is to be paralleled. This does not necessarily have to be at the end of the segments as the user can select a position anywhere on the segment to define these values. If the whole string is to be paralleled the user should select **no**.



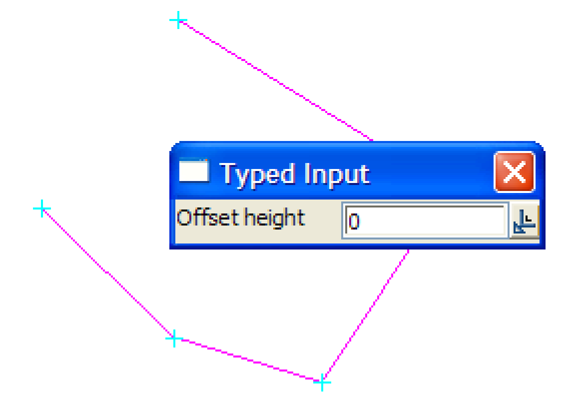
STEP 3:

The user is prompted for the parallel distance. Positive distance is to the right of the selected string. Negative to the left.



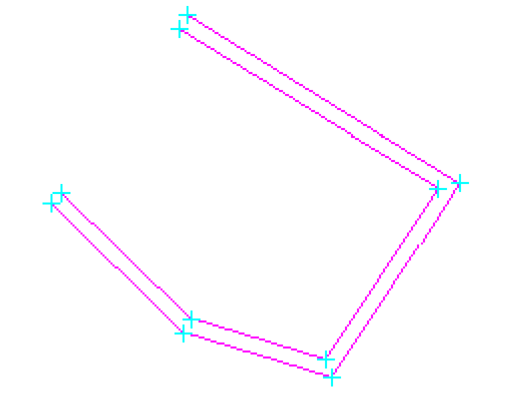
STEP 4:

The user is prompted for the offset height. This height value will be added to the selected string values to produce the heights for the paralleled string.



STEP 5:

The paralleled string is then generated using the supplied information.



15.23.6 Trim a String

Position of option on menu: **CAD =>String=>Trim**

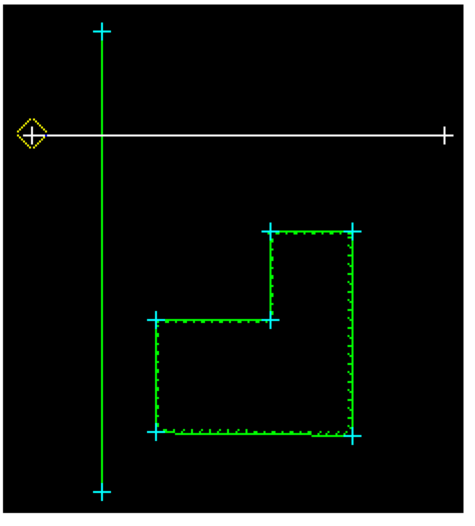
or by selection of appropriate icon from the toolbar. 

This option trims a string to another nominated cutting string.

On selecting **Trim**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

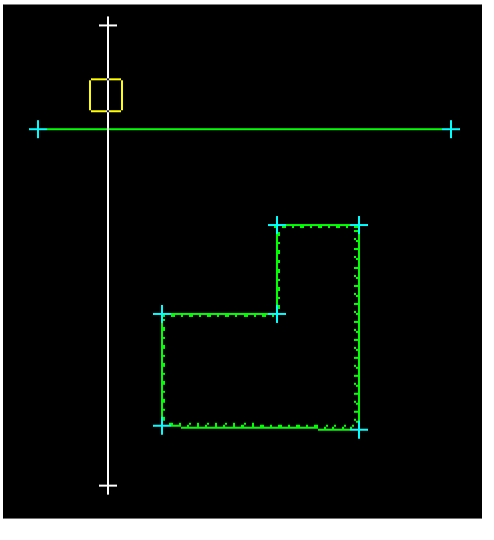
STEP 1:

The cutting string which will be used to define the trim extent is selected and accepted.



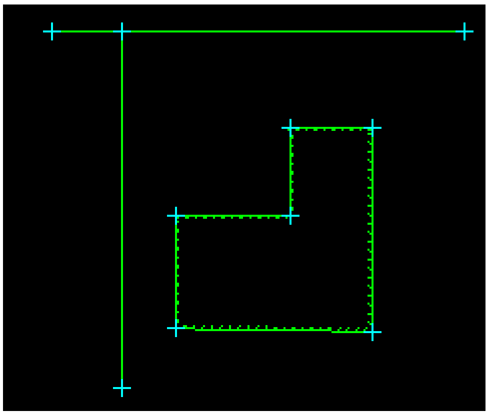
STEP 2:

The string to trim is selected. The position of the pick relative to the cutting string will dictate which part of the string will be trimmed. In this case, the selection is above the cutting string and so the trim will be of the top part of the string.



STEP 3:

The string is trimmed.



15.23.7 Extend a String

Position of option on menu: **CAD =>String=>Extend**

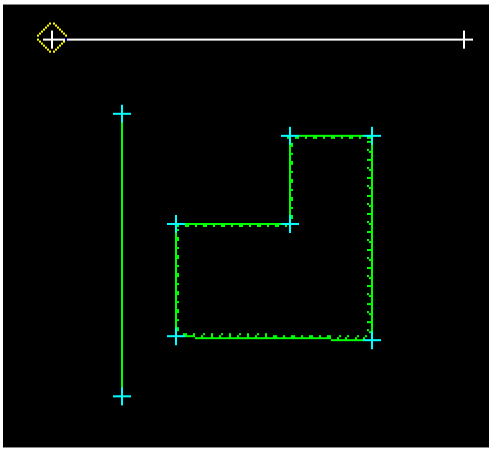
or by selection of appropriate icon from the toolbar.

This option extends a string to a another nominated string.

On selecting **Extend**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

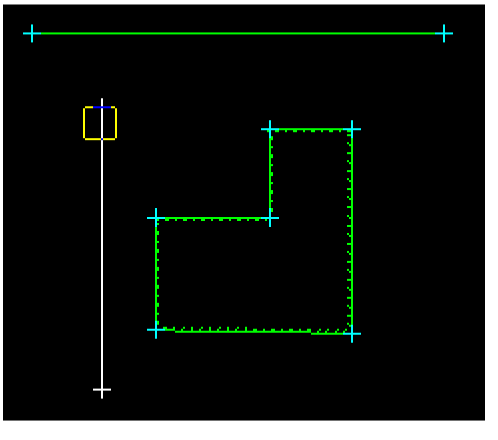
STEP 1:

The string to extend to is selected and accepted.



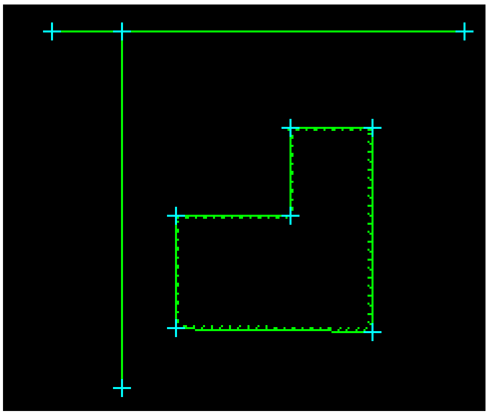
STEP 2:

The string to extend **to** selected and accepted



STEP 3:

The string is extended.



15.23.8 Extend String by Chainage Length

Position of option on menu: **CAD =>String=>Extend by chainage length**

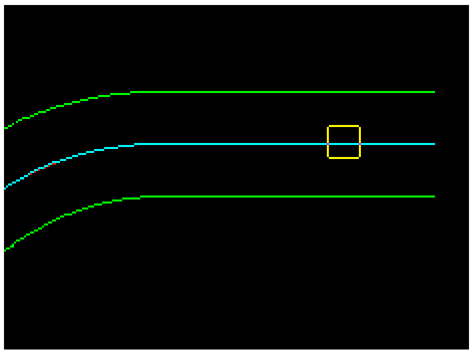
or by selection of appropriate icon from the toolbar.

This option extends a string by a given chainage value.

On selecting **Extend by chainage length**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

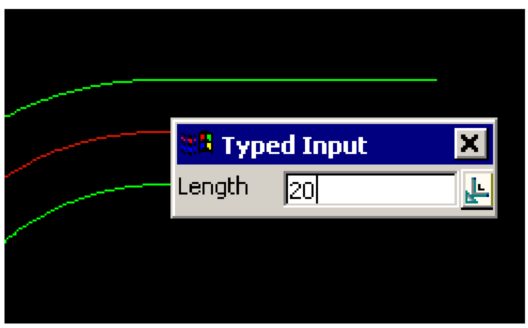
STEP 1:

The string to extend is selected and accepted.



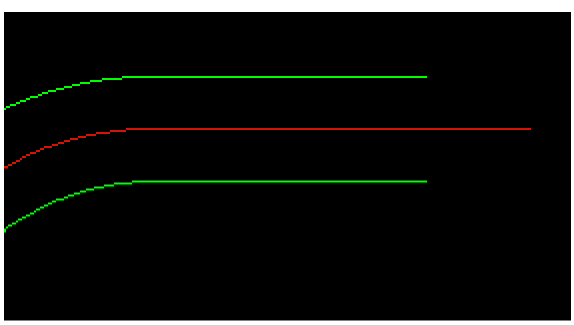
STEP 2:

The chainage length to extend the string is entered into the **Length** input Box. Positive is in the direction of the string. Negative in the opposite direction of the string.



STEP 3:

The string is extended by the given chainage amount.



15.23.9 Clip String

Position of option on menu: **CAD =>String=>Clip**

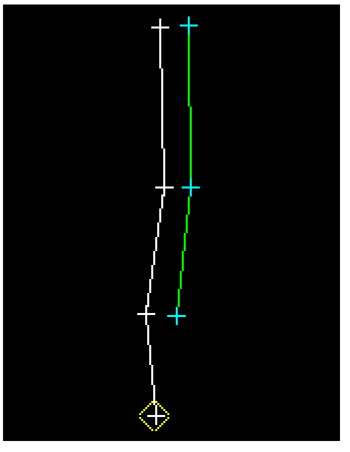
or by selection of appropriate icon from the toolbar.

This option deletes part of a string.

On selecting **Clip**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

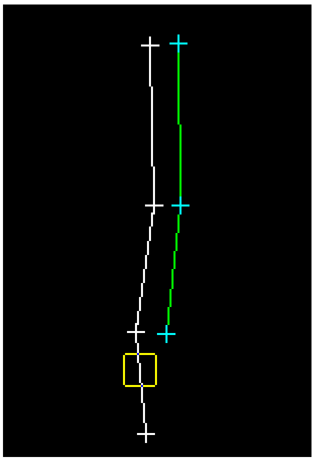
The string to be clipped is selected and accepted.

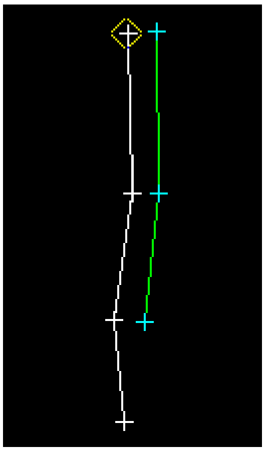


STEP 2:

The first and second positions defining what part of the string is to **remain** are selected and accepted.

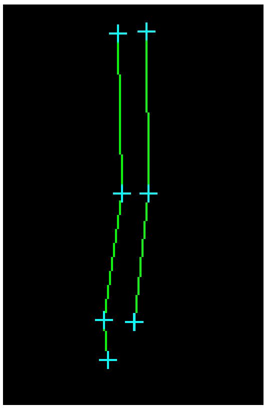
The positions do not have to be at vertices but can be anywhere on segments





STEP 3:

The string is clipped.



15.23.10 Delete an Internal Section of a String

Position of option on menu: **CAD =>String=>Clip internal**

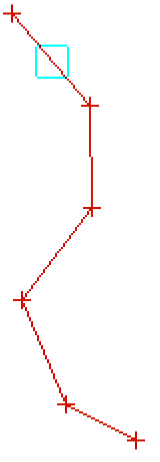
or by selection of appropriate icon from the toolbar. 

This option deletes an internal part of a string and leaves the two end sections.

On selecting **Clip internal**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

The string to be clipped is selected and accepted.

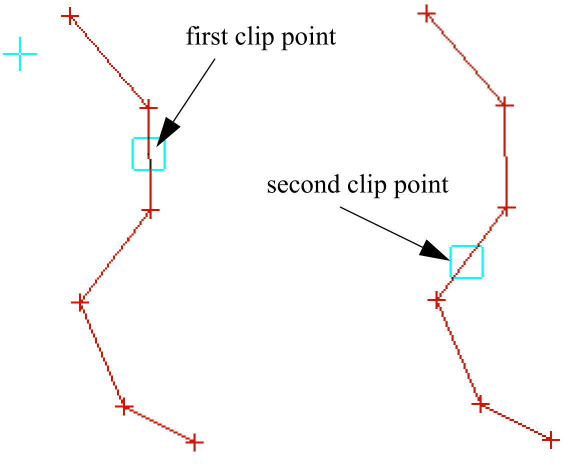


STEP 2:

The first and second position defining what part of the string is to **remain** is selected and accepted. This does not necessarily have to be at the end of the segments as the user can select a position anywhere on the segment to define these values.

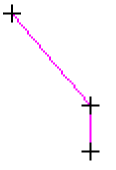
The first and second positions defining what part of the string is to be **deleted** are selected and accepted.

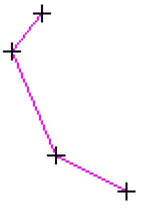
The positions do not have to be at vertices but can be anywhere on the segments



STEP 3:

The part of the string between the two selected clip points is deleted leaving two strings.





15.23.11 Delete a Segment

Position of option on menu: **CAD =>String=>Link clip**

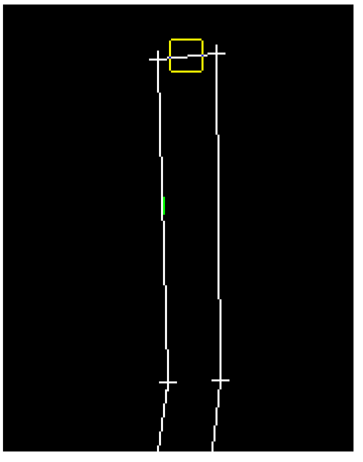
or by selection of appropriate icon from the toolbar.

This option deletes a segment from a string.

On selecting **Link clip**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

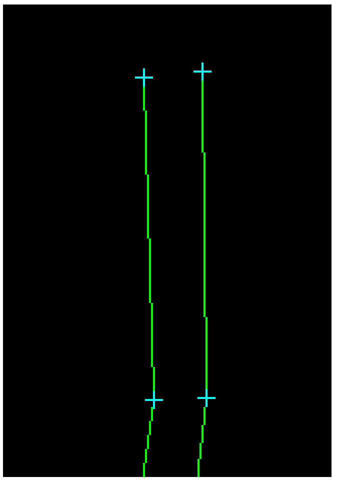
The **segment** to be deleted is selected and accepted.



STEP 2:

The selected segment is deleted.

Note - unless the selected string was a closed string, two strings are produced.



15.23.12 Split String

Position of option on menu: **CAD =>String=>Split**

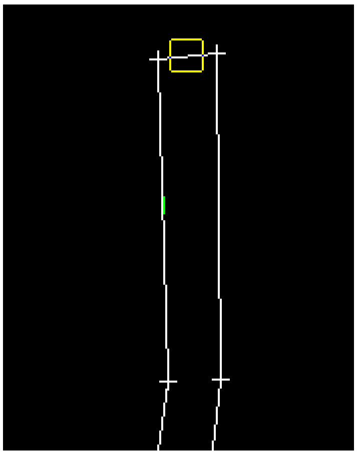
or by selection of appropriate icon from the toolbar.

This option splits one string into two strings at a given split position.

On selecting **Split**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

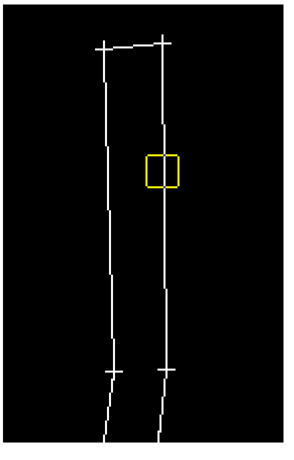
STEP 1:

The string to be split is selected and accepted.



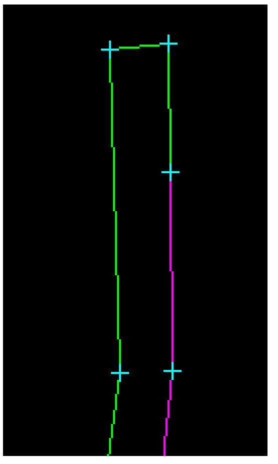
STEP 2:

The split position is selected and accepted.



STEP 3:

The split is made at the position selected. If no vertex exists at that position, a vertex is created.



15.23.13 Split Crossing Strings

Position of option on menu: CAD =>String=> Cross split

or by selection of appropriate icon from the toolbar.

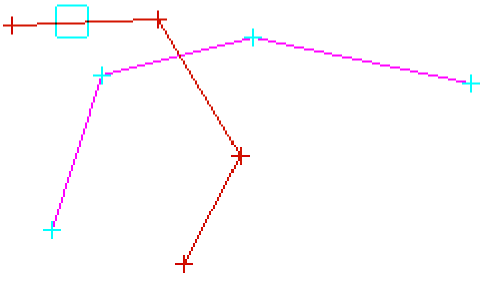
This option splits two crossing strings at the position of their crossing.

If there are multiple intersections between the two strings, then it splits the strings at the intersection that is the shortest distance to a pick position.

On selecting **Cross split**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

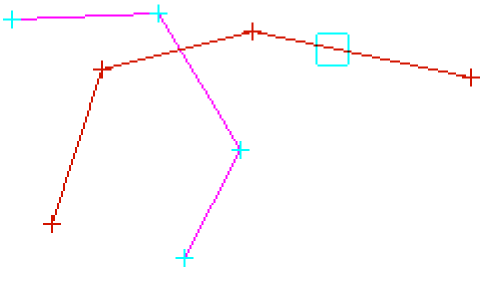
STEP 1:

Pick the first string.



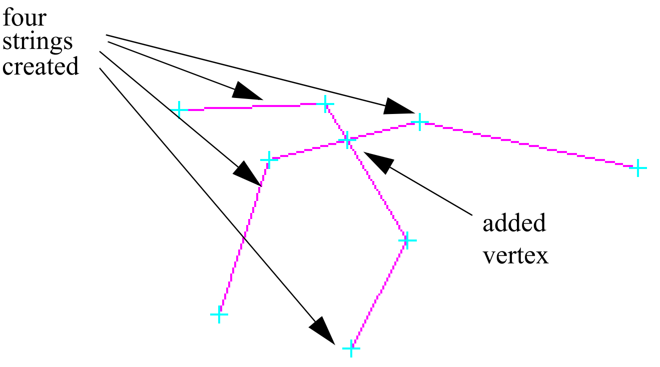
STEP 2:

Pick the second string.



STEP 3:

The two strings are then split where the strings crossed (at the intersection that is the shortest distance to a pick position). This will create four strings.


four
strings
created
added
vertex

15.23.14 Join String

Position of option on menu: **CAD =>String=>Join**

or by selection of appropriate icon from the toolbar. 

This option joins two super strings together at their end points.

There are four possible ways of joining the strings and the required case is determined by picking the strings with direction.

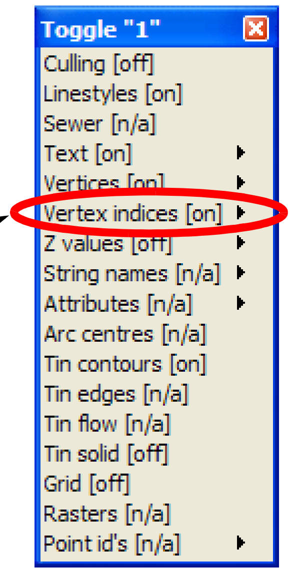
On selecting the **Join** option, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

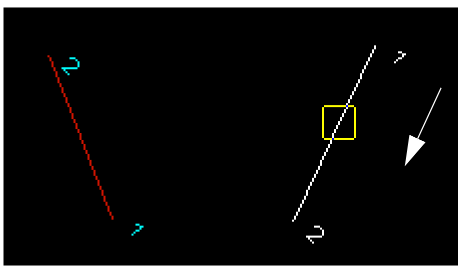
STEP 1:

Select and accept a super string *with direction*.

For notes on picking strings with direction, see Picking with Direction.

Note: *Vertex indices* can be displayed by toggling the option on the Toggle Menu.



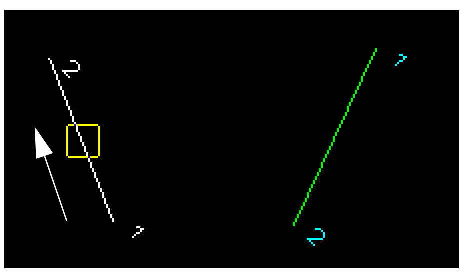


pick with direction

STEP 2:

Select and accept another super string *with direction*.

pick with direction



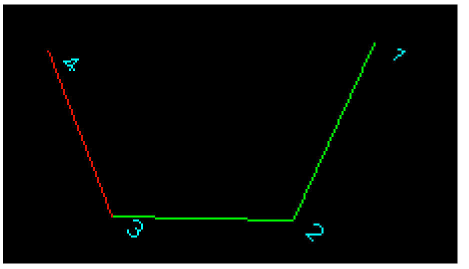
STEP 3:

The head of the first string (as determined by the picking direction) is then joined to the tail of the second string (as determined by the picking direction).

After the strings are joined, the vertex indices will change since the order of the vertices in the string is now different.

The colour of the constructed line is adopted from first selected super string. The colour of the second string will remain unchanged.

Note: The constructed line and selected strings are stored in the model of the first selected string.


4
1
3
2

15.23.15 Join String Many

Position of option on menu: **CAD =>String=>Join many**

or by selection of appropriate icon from the toolbar. 

This option joins many strings together. Picking with direction is used to specify which ends of the strings are joined.

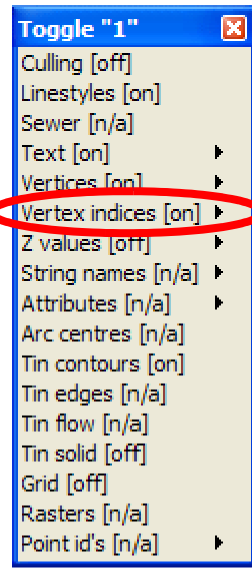
On selecting the **Join many** option, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

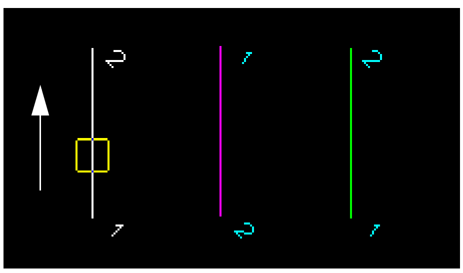
Pick a super string *with direction*.

For notes on picking strings with direction, see Picking with Direction.

Note: The ***Vertex indices*** can be displayed by toggling the option on the Toggle Menu.



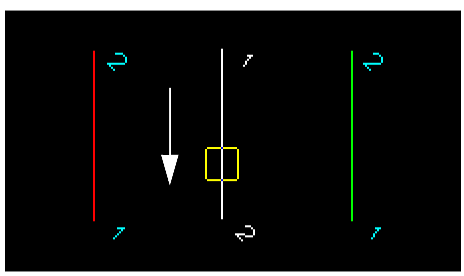
pick with direction



STEP 2:

Select and accept another super string, again picking *with direction*.

pick with direction

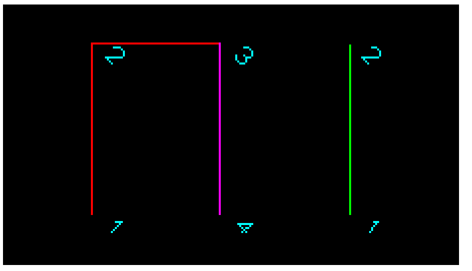


STEP 3:

The **head** of the first string is then joined to the **tail** of the second string. What is the head or tail of the strings is determined by the *picking direction*.

After the strings are joined, the vertex indices to reflect the new order of the vertices in the string. The colour of the constructed segment will be the same as that of the first selected super string. The colour of the second string will remain unchanged.

Note: The constructed segment and selected strings are stored in the model which contains the 1st selected string.

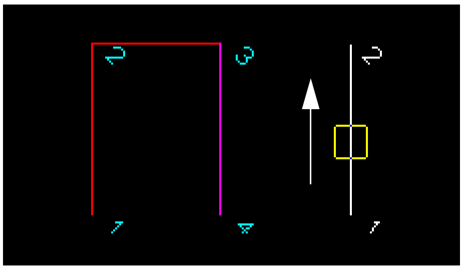


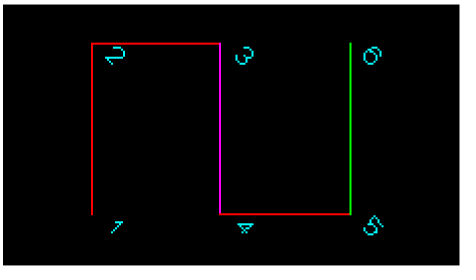
STEP 4:

The *Join Many* option joins extra strings to the already selected (and joined) strings. The user may continue selecting additional strings to join together by select and accept additional super strings by picking with direction.

This option can be terminated by the user with the **Esc** key.

Note: Each further construction line adopts the same colour as the 1st selected string. Both the constructed line and selected strings are stored in the model which contains the first selected string.





15.23.16 Join Fillet

Position of option on menu: **CAD =>String=>Join fillet**

or by selection of appropriate icon from the toolbar. 

This option fillets, trims and joins strings.

There are eight possible ways of filleting the strings and the required case is determined by picking the strings with direction and by the sign of the radius.

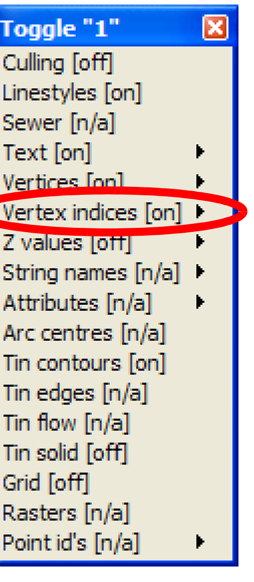
On selecting the **Join strings** option, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

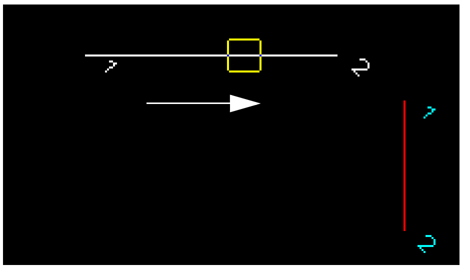
Pick a super string *with direction*.

For notes on picking strings with direction, see Picking with Direction.

Note: The Vertex indices can be displayed by toggling the option on the Toggle Menu.



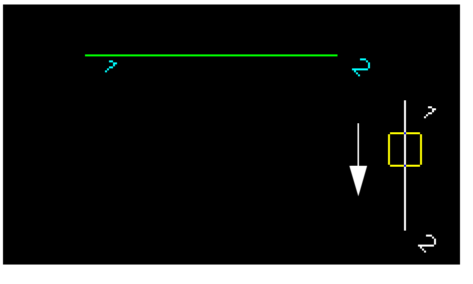
pick with direction



STEP 2:

Select and accept another super string, again picking *with direction*.

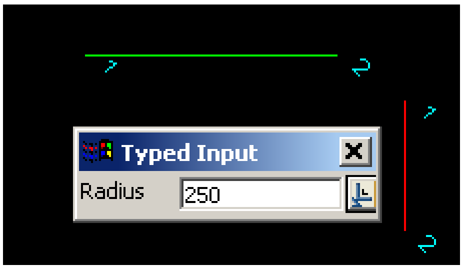
pick with direction



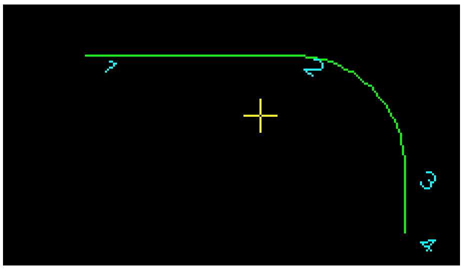
STEP 3:

After the second super string is accepted, the **Radius** input box will appear. Type the *fillet radius* into the input box and press the Enter key. The browse button  on the input box can be used to define the fillet radius by measurement of existing elements.

Note: A **positive** fillet radius value will curve to the right, where as a **negative** fillet radius will curve to the left.



A fillet arc with the radius is constructed and joined to the two strings and the ends of the two strings are automatically trimmed back to the arc.



The vertex indices will change to accommodate the new vertices in the string. The colour of the constructed line and 2nd selected string adopts the colour of the 1st selected super string.

Note: The constructed arc and selected strings are stored in the model of the first selected string.

15.23.17 Offset

Position of option on menu: **CAD =>String =>Offset**

or by selection of appropriate icon from the toolbar. 

This option allows the user to create "parallel" strings, *i.e.* creating a string that is offset by a given distance from a selected string.

There are five possible ways of offsetting the strings as well as options for offsetting full strings or partial strings.

The string to offset is selected by picking with direction, and the offset distance is negative to offset to the left of the pick direction and positive to offset to the right of the pick direction.

On selecting the **Offset** option, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

e.g.

<[N][J] Pick string to offset or type (n)ormal, (p)artial, (j)oin, (i)ntersect, (f)illet, (d)ual, (c)lip> [picks][fast][Menu]

The letters in the square brackets indicate which offset modes are currently selected.

For example, [N][J] at the front indicates that (n)ormal and (j)oin are the selected options.

For the option *N for (n)ormal*, go to	Typing N for (n)ormal
P for (p)artial	Typing P for (p)artial
J for (j)oin	Typing J for (j)oin
I for (i)ntersect	Typing I for (i)ntersect
F for (f)illet	Typing F for (f)illet
D for (d)ual	Typing D for (d)ual
C for (c)lip	Typing C for (c)lip

Typing N for (n)ormal

Typing **N** or **n** for (n)ormal causes the **whole string to be offset**.

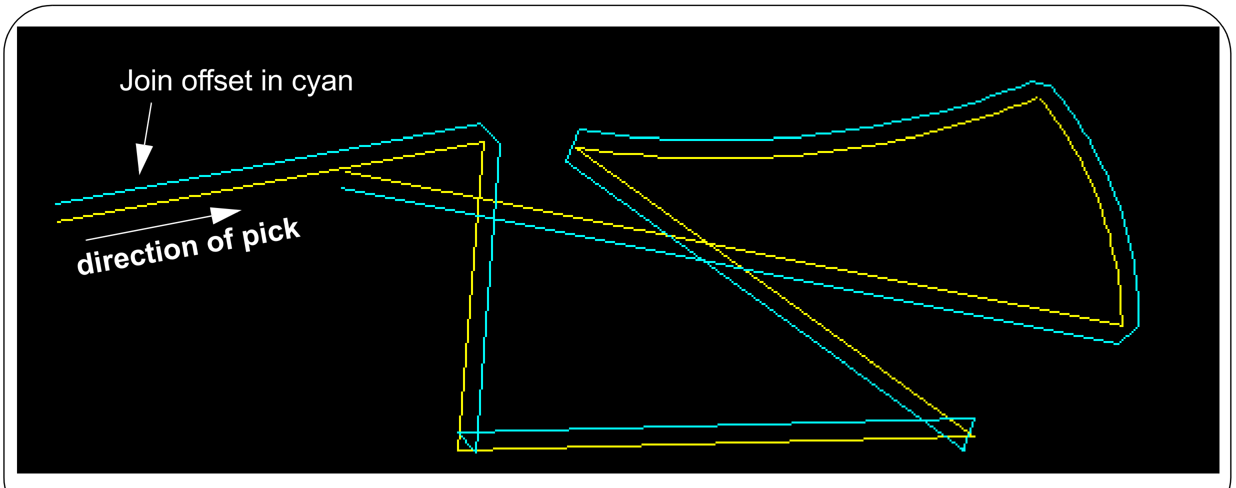
Note - the other choice is Partial when only part of a string is offset.

When **N** is typed, the first character in the square brackets changes to [N].

Continue to the next mode Typing P for (p)artial or to the beginning of the Offset documentation 15.23.17 Offset.

Typing J for (j)oin

For **Join**, each segment of the selected string is offset by the given distance and the end points of adjacent offset segments are joined together.

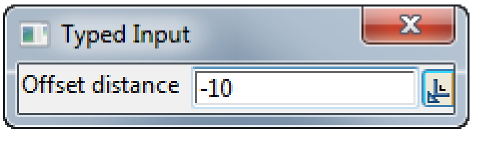


Offset Join with Offset Distance of -10

So when **J** or **j** is typed, the second character in the square brackets changes to [J] and the string to offset is then selected by picking with direction.

<[N][J] Pick string to offset or type (n)ormal, (p)artial, (j)oin, (i)ntersect, (f)illet, (d)ual, (c)lip> [picks][fast][Menu]

After the string is selected, the **Offset distance** typed input box is placed on the screen and the distance to offset is typed into the box followed by the <Enter> key.



NOTE: The offset distance is negative to offset to the left of the pick direction and positive to offset to the right of the pick direction.

The selected string is then offset.

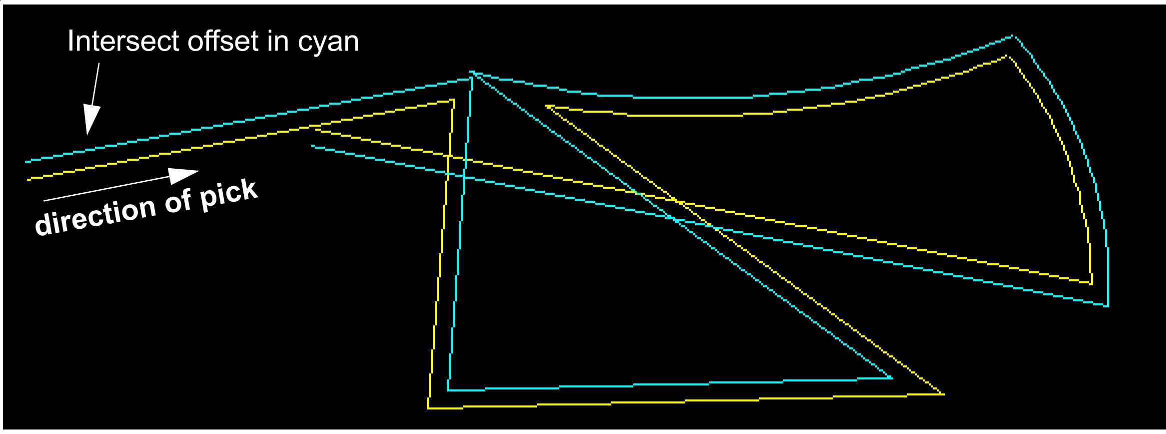
After the string is offset, the mode can be changed or left the same and another string selected to offset using the new mode(s).

The option is terminated by pressing the <Esc> key or by clicking **X** on the top right corner of the **Offset distance** typed input box.

Continue to the next mode [Typing I for (i)ntersect] or to the beginning of the Offset documentation [15.23.17 Offset].

Typing I for (i)ntersect

For **Intersect**, each segment of the selected string is offset by the given distance and the ends of adjacent offset segments are extended or contracted so that they intersect.

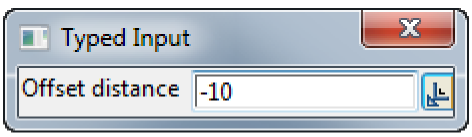


Offset Intersect with Offset Distance of -10

So when **I** or **i** is typed, the second character in the square brackets changes to **[I]** and the string to offset is then selected by picking with direction.

<[N][I] Pick string to offset or type (n)ormal, (p)artial, (j)oin, (i)ntersect, (f)illet, (d)ual, (c)lip> [picks][fast][Menu]

After the string is selected, the **Offset distance** typed input box is placed on the screen and the distance to offset is typed into the box followed by the <Enter> key.



NOTE: The offset distance is negative to offset to the left of the pick direction and positive to offset to the right of the pick direction.

The selected string is then offset.

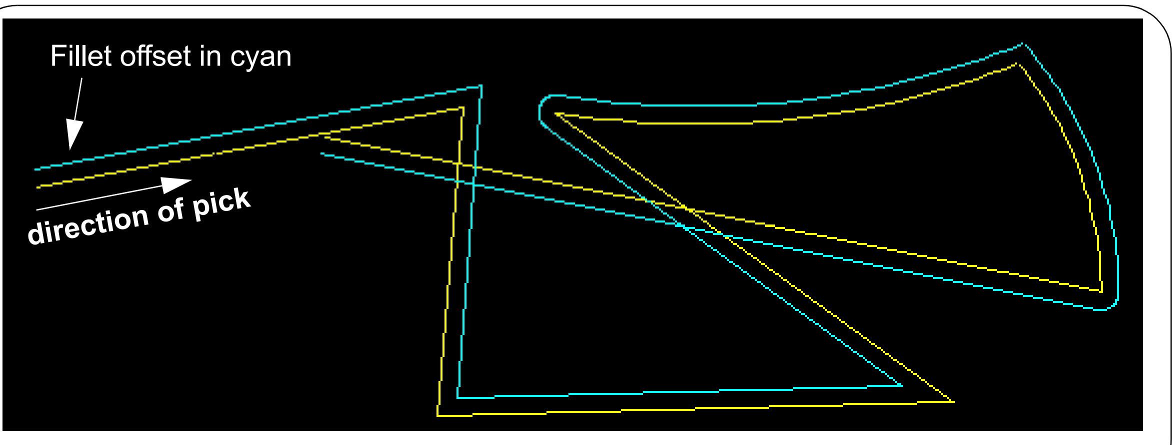
After the string is offset, the mode can be changed or left the same and another string selected to offset using the new mode(s).

The option is terminated by pressing the <Esc> key or by clicking **X** on the top right corner of the **Offset distance** typed input box.

Continue to the next mode Typing F for (f)illet or to the beginning of the Offset documentation 15.23.17 Offset.

Typing F for (f)illet

For **Fillet**, each segment of the selected string is offset by the given distance and the ends of adjacent offset segments are extended or contracted so they intersect. If the change of direction between adjacent segments is greater than 180 degrees, an arc is used instead of an intersect of the two segments.

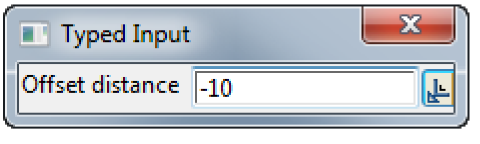


Offset Fillet with Offset Distance of -10

So when **F** or **f** is typed, the second character in the square brackets changes to [F] and the string to offset is then selected by picking with direction.

<[N][F] Pick string to offset or type (n)ormal, (p)artial, (j)oin, (i)ntersect, (f)illet, (d)ual, (c)lip> [picks][fast][Menu]

After the string is selected, the **Offset distance** typed input box is placed on the screen and the distance to offset is typed into the box followed by the <Enter> key.



NOTE: The offset distance is negative to offset to the left of the pick direction and positive to offset to the right of the pick direction.

The selected string is then offset.

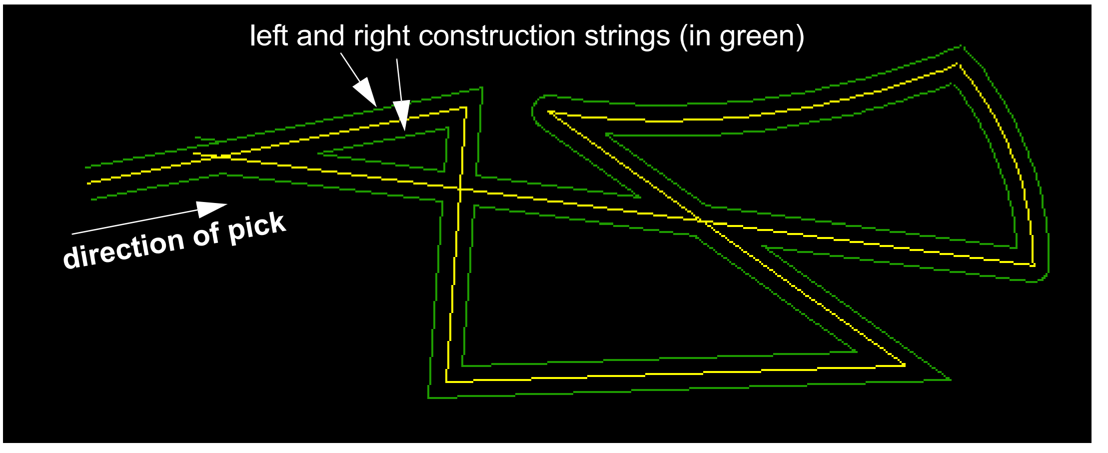
After the string is offset, the mode can be changed or left the same and another string selected to offset using the new mode(s).

The option is terminated by pressing the <Esc> key or by clicking **X** on the top right corner of the **Offset distance** typed input box.

Continue to the next mode [Typing D for (d)ual] or to the beginning of the Offset documentation [15.23.17 Offset].

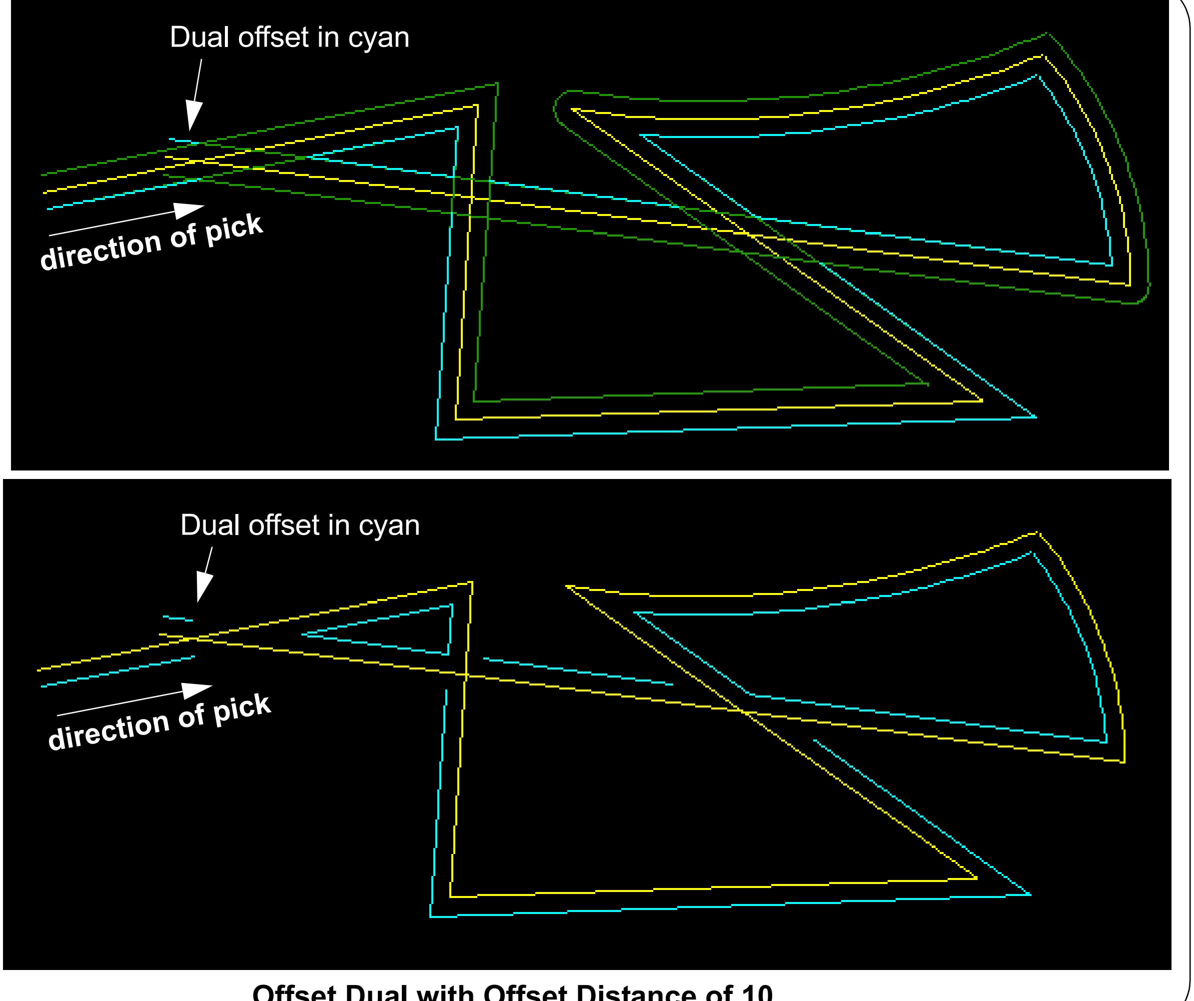
Typing D for (d)ual

For **Dual**, *construction* strings are created by offset fillets to both the left and right with the given offset distance.



Construction Strings for Offset Dual with Offset Distance of 10

Then the **Offset Dual** is created for the given offset distance but the created string **does not cut** either of the two left and right offset construction strings.



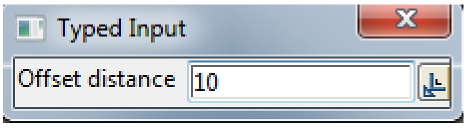
Offset Dual with Offset Distance of 10

More than one string may be created by the **Dual** option.

So when **D** or **d** is typed, the second character in the square brackets changes to [D] and the string to offset is then selected by picking with direction.

<[N][D] Pick string to offset or type (n)ormal, (p)artial, (j)oin, (i)ntersect, (f)illet, (d)ual, (c)lip> [picks][fast][Menu]

After the string is selected, the **Offset distance** typed input box is placed on the screen and the distance to offset is typed into the box followed by the <Enter> key.



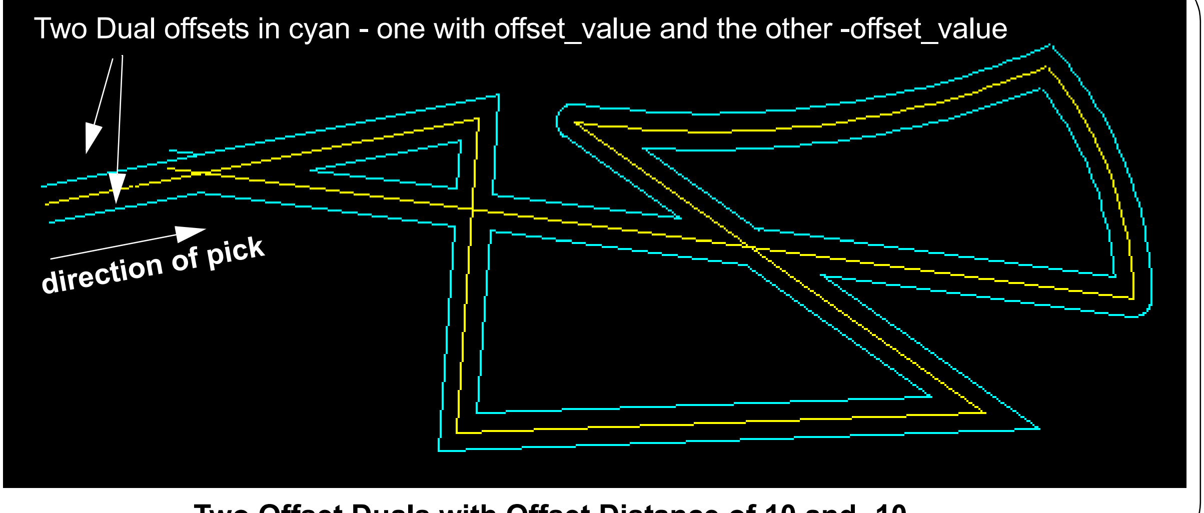
NOTE: The offset distance is negative to offset to the left of the pick direction and positive to offset to the right of the pick direction.

The selected string is then offset.

After the string is offset, the mode can be changed or left the same and another string selected to offset using the new mode(s).

The option is terminated by pressing the <Esc> key or by clicking **X** on the top right corner of the **Offset distance** typed input box.

NOTE: Running the Dual offset command twice, one with an offset and then again with the negative of the offset (-offset)

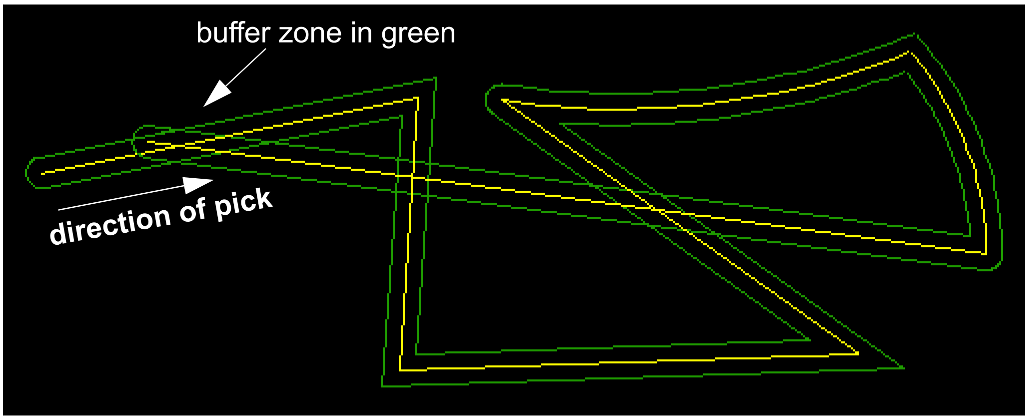


Two Offset Duals with Offset Distance of 10 and -10

Continue to the next mode [Typing C for (c)lip] or to the beginning of the Offset documentation [15.23.17 Offset].

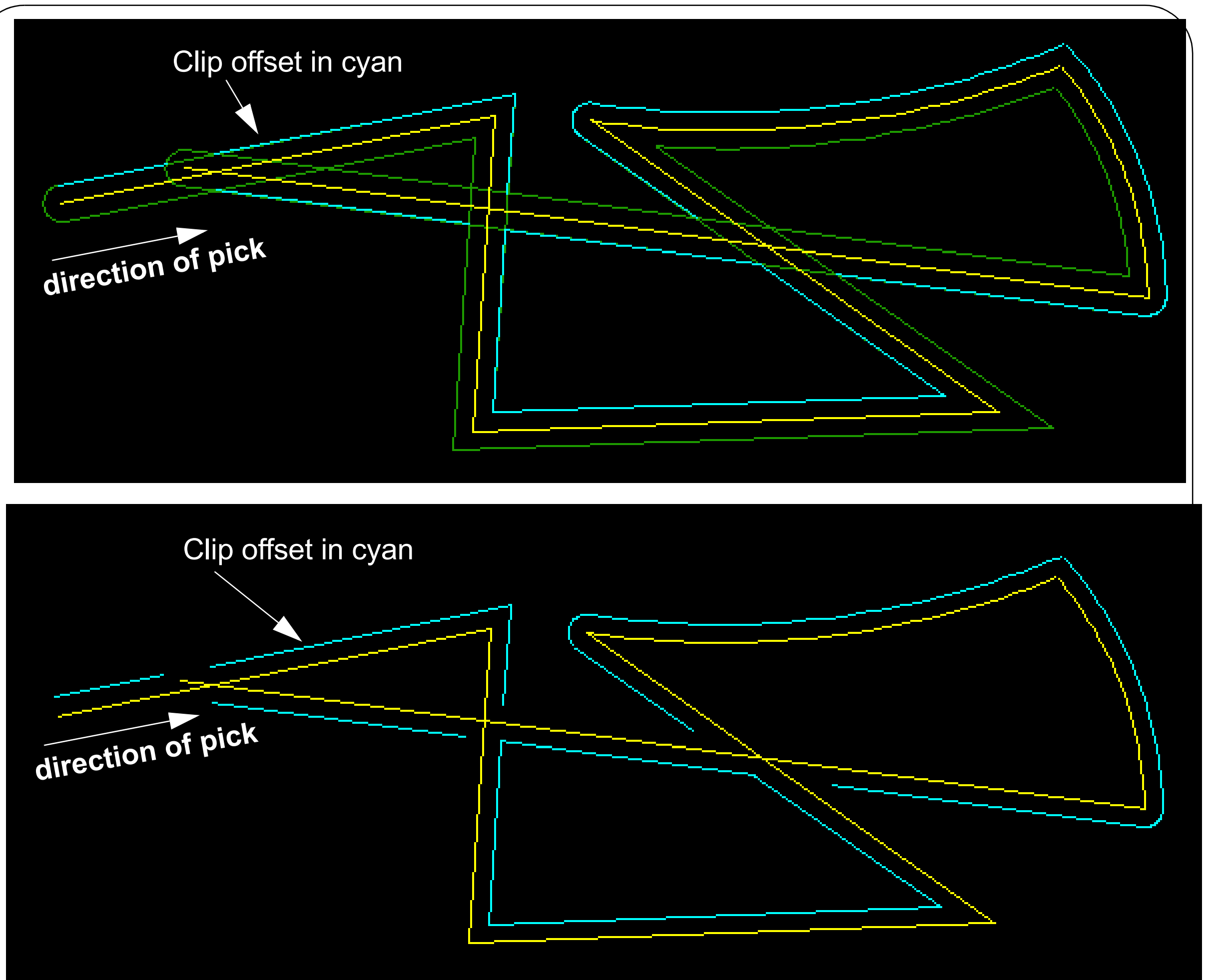
Typing C for (c)lip

For **Clip**, *construction* strings are created as for the Dual option by offsetting the selected string to both the left and right by the given offset distance as an offset fillet and a buffer zone defined by the construction strings and end caps with radius equal to the offset distance at each end of the selected string.



Buffer Zone for Offset Clip with Offset Distance of -10

Then the **Offset Clip** is created for the given offset distance, but it **does not cut** the buffer zone for the selected string.



Offset Clip with Offset Distance of -10

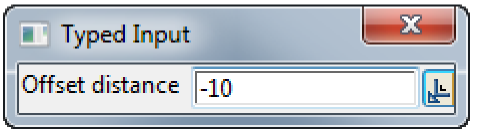
More than one string may be created by the **Clip** option.

So a Clip is like the Dual option except everything is removed from the buffer zone.

So when **C** or **c** is typed, the second character in the square brackets changes to [C] and the string to offset is then selected by picking with direction.

<[N][C] Pick string to offset or type (n)ormal, (p)artial, (j)oin, (i)ntersect, (f)illet, (d)ual, (c)lip> [picks][fast][Menu]

After the string is selected, the **Offset distance** typed input box is placed on the screen and the distance to offset is typed into the box followed by the <Enter> key.



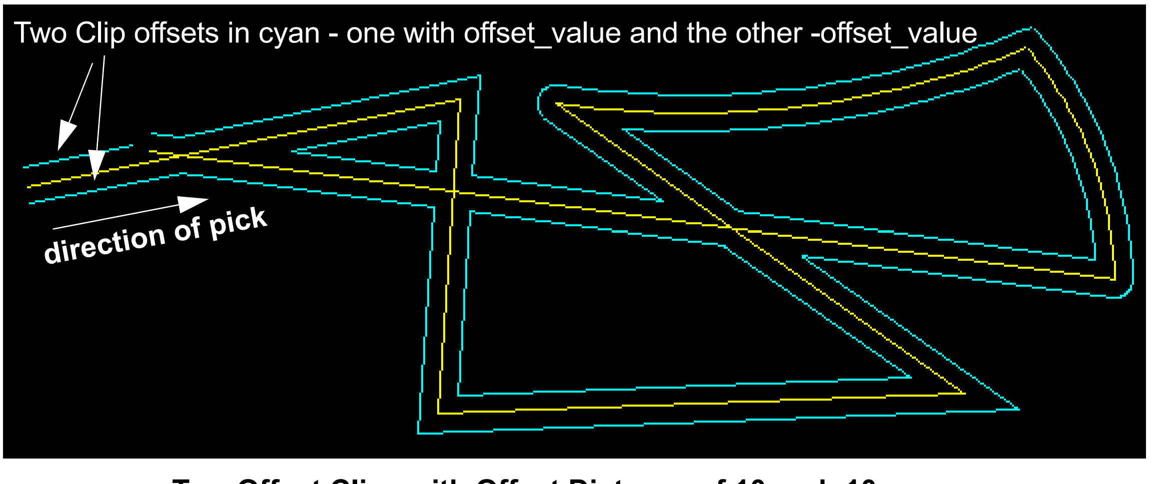
NOTE: The offset distance is negative to offset to the left of the pick direction and positive to offset to the right of the pick direction.

The selected string is then offset.

After the string is offset, the mode can be changed or left the same and another string selected to offset using the new mode(s).

The option is terminated by pressing the <Esc> key or by clicking **X** on the top right corner of the **Offset distance** typed input box.

NOTE: Running the Clip offset command twice, one with an offset and then again with the negative of the offset (-offset)



Two Offset Clips with Offset Distance of 10 and -10

Continue to the first mode Typing N for (n)ormal or to the beginning of the Offset documentation 15.23.17 Offset.

Typing P for (p)artial

Selecting **P** for (p)artial allows for a **part** of the string to be offset.

When **P** or **p** is typed, the first character in the square brackets changes to [P].

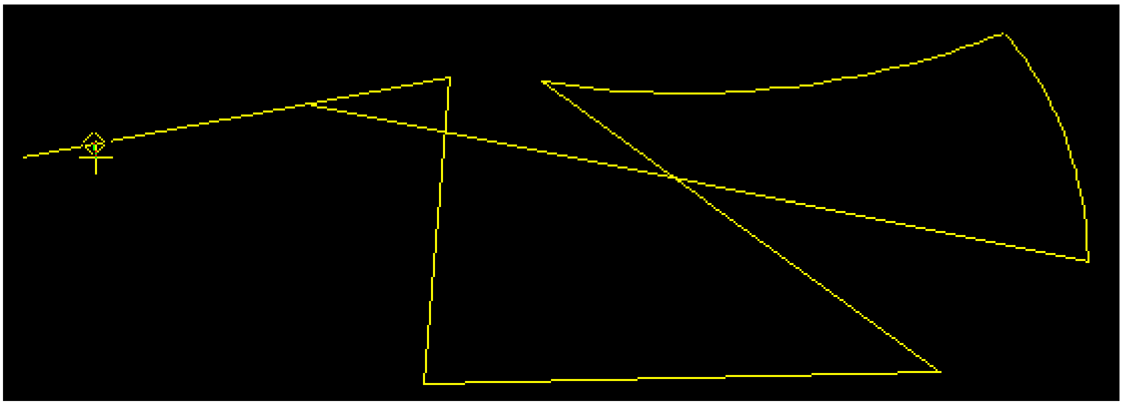
e.g.

<[P][I] Pick string to offset or type (n)ormal, (p)artial, (j)oin, (i)ntersect, (f)illet, (d)ual, (c)lip> [picks][fast][Menu]

After **P** is typed, the two positions on the string are selected to indicate the subsection of the string that is to be offset.

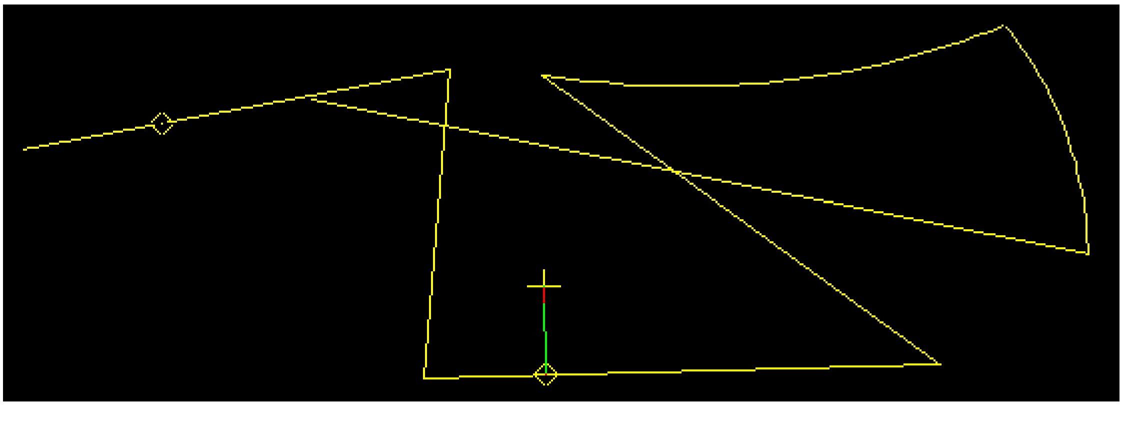
STEP 1:

Pick your first position. This does not have to be on the string to offset because it will be dropped perpendicular onto the string to give the first point of the partial offset.



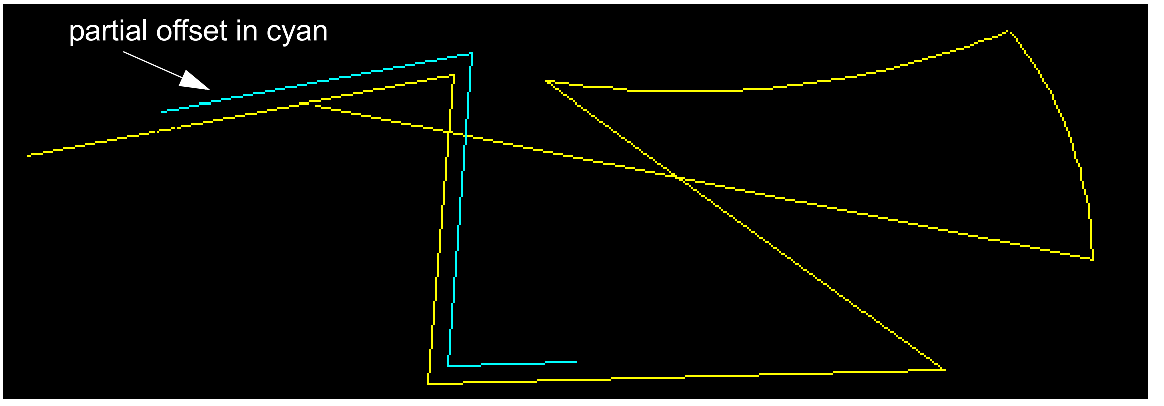
STEP 2:

Pick your second position. This does not have to be on the string to offset because it will be dropped perpendicular onto the string to give the second point of the partial offset.



STEP 3:

After accepting the second position, the part of the string between the two dropped positions is offset.



After the string is offset, the mode can be changed or left the same and another string selected to offset using the new mode(s).

The option is terminated by pressing the <Esc> key or by clicking **X** on the top right corner of the **Offset distance** typed input box.

Continue to the next mode Typing J for (j)oin or to the beginning of the Offset documentation 15.23.17 Offset.

15.23.18 Trace

Position of option on menu: **CAD =>String =>Trace**

or by selection of appropriate icon from the toolbar. 

This option allows the user to create a new string by

(a) creating points as in a normal string

and/or

(b) selecting parts of existing string to copy and make part of the new string (i.e. tracing over part of an existing string)

and/or

(c) closing the new string

On selecting the **Trace** option, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** screen message area at the bottom of the window.

e.g.

<[P] Pick point to append or (t)race, (c)lose> [picks][fast][Menu]

The letter in the square brackets indicate which trace mode is currently selected.

For example, [T] at the front indicates that (t)race mode is selected

For the option *P for (p)ick* go to Typing N for (n)ormal
T for (t)race Typing P for (p)artial
C for (c)lose Typing J for (j)oin

Typing P for (p)ick

Typing P or p for pick means that new vertices of the string will be created in the normal way for a string editor, and the created vertices are appended to the end of the new string being created.

When **P** or **p** is typed, the first character in the square bracket changes to [P].

e.g.

<[P] Pick point to append or (t)race, (c)lose> [picks][fast][Menu]

Points are then created for the new string by the normal pick and accept.

The option is terminated by pressing the <Esc> key.

Continue to the next mode Typing T for (t)race or to the beginning of the Trace documentation 15.23.18 Trace.

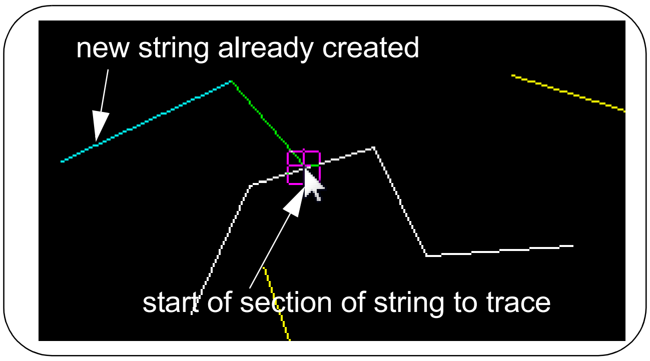
Typing T for (t)race

For **Trace**, an part of an existing string is selected to copy and make part of the new string. That is, part of an existing string is traced over to form part of the new string.

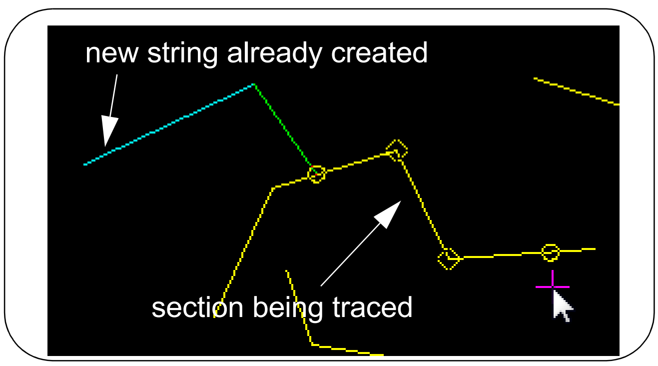
When **T** or **t** is typed, the second character in the square brackets changes to [T]

<[T] Select start point to trace or (p)ick, (c)lose> [picks][fast][Menu]

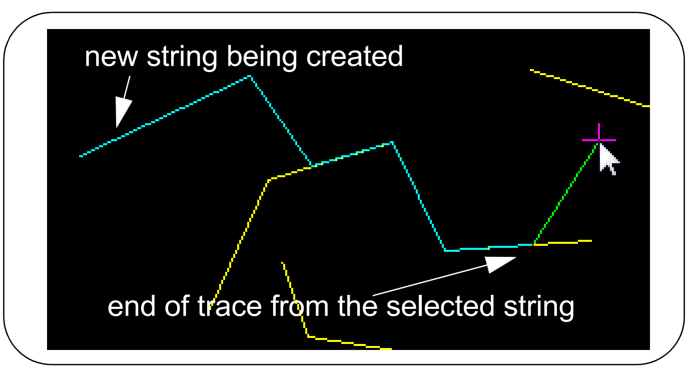
and the string to trace over is then selected by picking the string at the start position of the section of the string to trace over.



The cursor is then moved along the sting in the direction of the trace. Vertices on the traced section are displayed as diamonds and a non-vertex start on the string of the trace is shown as a circle.



The final position of the trace on the string is then selected and the traced portion is copied to the string being created.



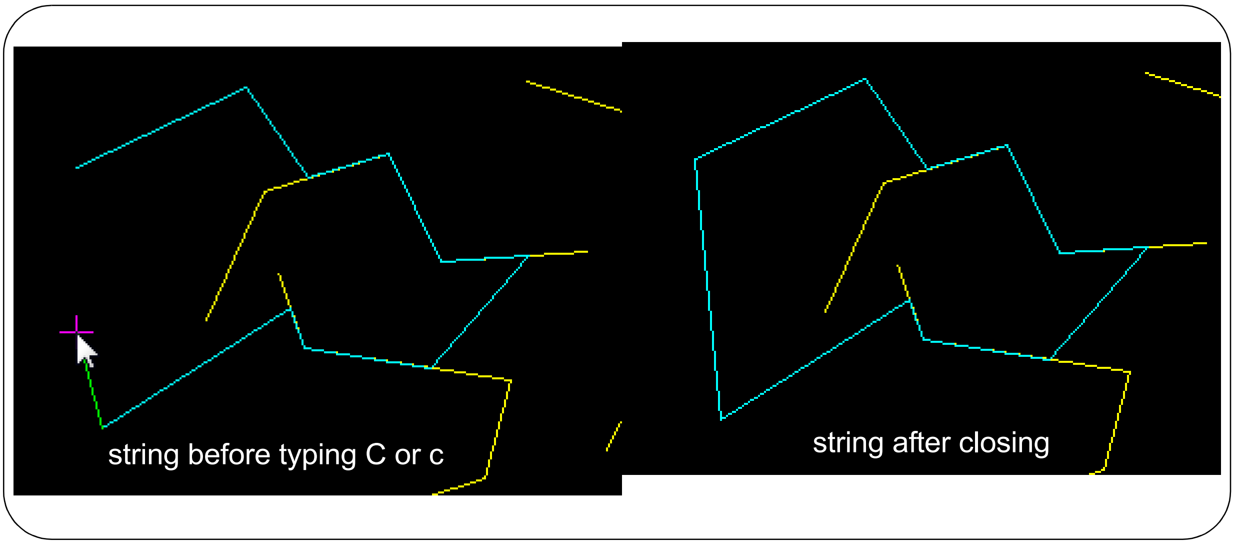
Another string can then be selected to trace over, or P or C typed to change modes.

The option is terminated by pressing the <Esc> key.

Continue to the next mode Typing C for (c)lose or to the beginning of the Trace documentation 15.23.18 Trace.

Typing C for (c)lose

Typing **C** or **c** will **Close** the string and ends the creation of the new string.

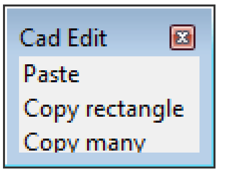


A new Trace option then begins.

The option is terminated by pressing the <Esc> key.

Go to the beginning of the Trace documentation 15.23.18 Trace.

15.24 CAD Edit



For the option *Paste*, please go to 15.24.1 Paste
Copy rectangle 15.24.2 12d Copy Rectangle
Copy many 15.24.3 12d Copy Many

15.24.1 Paste

Pastes the 12da data from the clipboard into the current view in 12d.

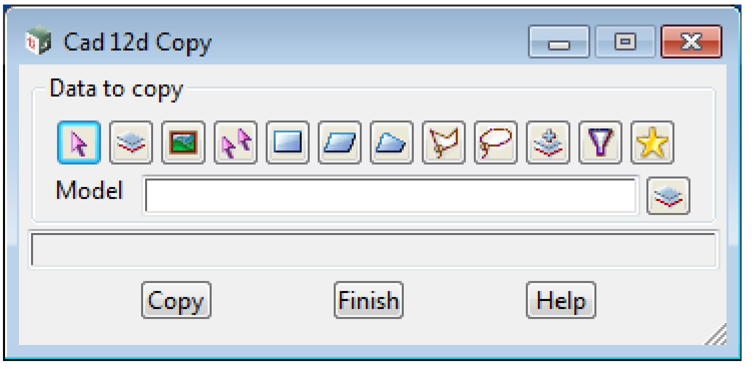
15.24.2 12d Copy Rectangle

In the chosen view select the first point and then drag the cursor to visualise your desired rectangle. When the second point is accepted, the data inside the rectangle is copied to the clipboard in 12da format, ready for pasting into an open 12d project. The data is copied into models of the same model names as the original data.

15.24.3 12d Copy Many

Data is copied to the clipboard by selecting data from the Data to Copy field and then selecting **Copy**.

On selecting the **Copy many** option, a **Cad 12d Copy** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

Data to Copy

data selection type - for a full description go to 4.19.3 Data Source.

Model model box

select the data from the specified model.

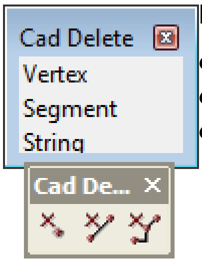
Copy button

Copies the selected data to the clipboard in 12da format.

15.25 CAD Delete

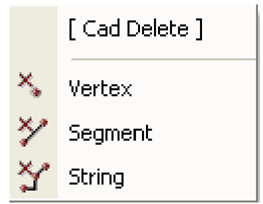
Position of option on menu: **CAD =>Delete**

The **Cad Delete** walk-right menu is



Menu of Options to Delete Vertices, Segments and Strings

delete a vertex
delete a segment
delete a string



For the option *Vertex*, go to 15.25.1 Delete Vertices
Segment 15.25.2 Delete Segments
String 15.25.3 Delete Strings

15.25.1 Delete Vertices

Position of option on menu: **CAD =>Delete =>Points**

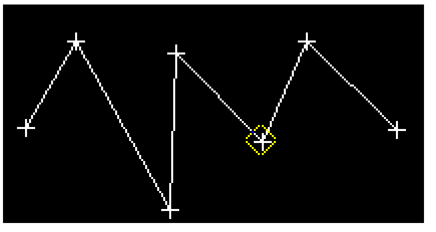
or by selection of appropriate icon from the toolbar. 

This option deletes vertices of super strings.

On selecting **Points**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

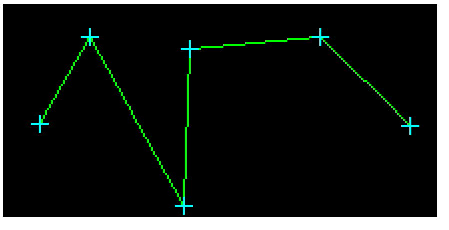
STEP 1:

The vertex to be deleted is selected and accepted (This must be a super string).



STEP 2:

The vertex point is deleted and the string redrawn.



15.25.2 Delete Segments

Position of option on menu: **CAD =>Delete =>Segments**

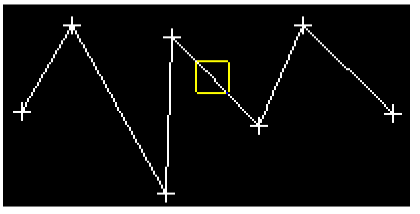
or by selection of appropriate icon from the toolbar. 

This option deletes segments of super strings.

On selecting **Segments**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

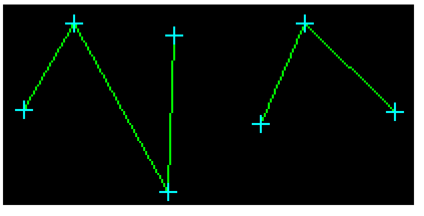
STEP 1:

The segment to be deleted is selected and accepted.



STEP 2:

The segment is deleted leaving two separate strings in this case.



15.25.3 Delete Strings

Position of option on menu: **CAD =>Delete =>Strings**

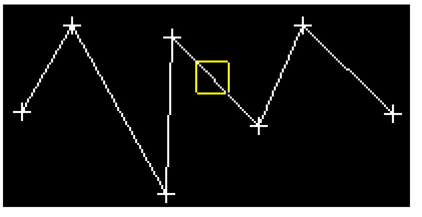
or by selection of appropriate icon from the toolbar. 

This option deletes selected strings. The string type does not specifically have to be a super string.

On selecting **Strings**, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

STEP 1:

The string to be deleted is selected and accepted.



STEP 2:

The string is deleted.

15.25.4 Delete Selection of Strings

Position of option on menu: **CAD =>Delete =>Selection**

or by selection of appropriate icon from the toolbar. 

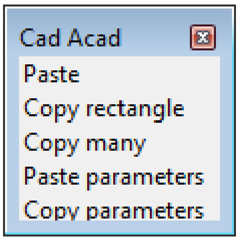
This option allows the deleting of selected objects specified in a source box. For help on this option see 28.9.14 Delete.

On selecting the **Delete** option, the user is prompted for the relevant data in the screen message box located at the bottom left hand corner of the ***12d Model*** application window.

15.26 Cad Acad

Position of option on menu: **CAD =>Acad**

The Cad Acad walk-right menu is



For the option *Paste*, go to 15.26.1 Paste
Copy rectangle 15.26.2 Copy rectangle
Copy many 15.26.3 Copy many
Paste parameters 15.26.4 Paste parameters
Copy parameters 15.26.5 Copy parameters

15.26.1 Paste

Pastes the Dwg/Dxf data from the clipboard into the current view in 12d.

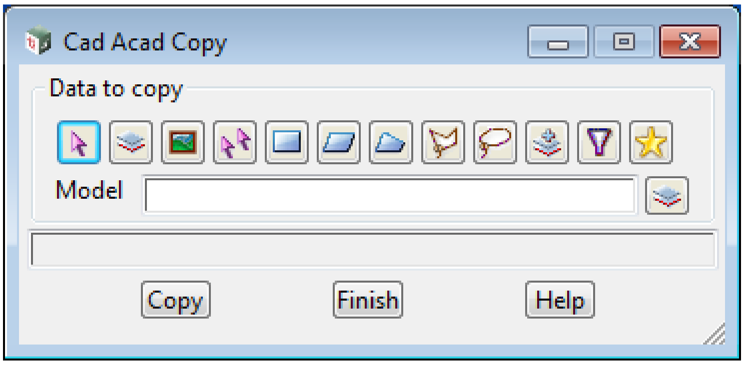
15.26.2 Copy rectangle

In the chosen view select the first point and then drag the cursor to visualise your desired rectangle. When the second point is accepted, the data inside the rectangle is copied to the clipboard for pasting.

15.26.3 Copy many

Data is copied to the clipboard by selecting data from the Data to Copy field and then selecting **Copy**.

Selecting the **Copy many** option, brings up the **Cad Acad Copy panel**.



The fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

Data to Copy

data selection type - for a full description go to 4.19.3 Data Source.

Model model box

select the data from the specified model.

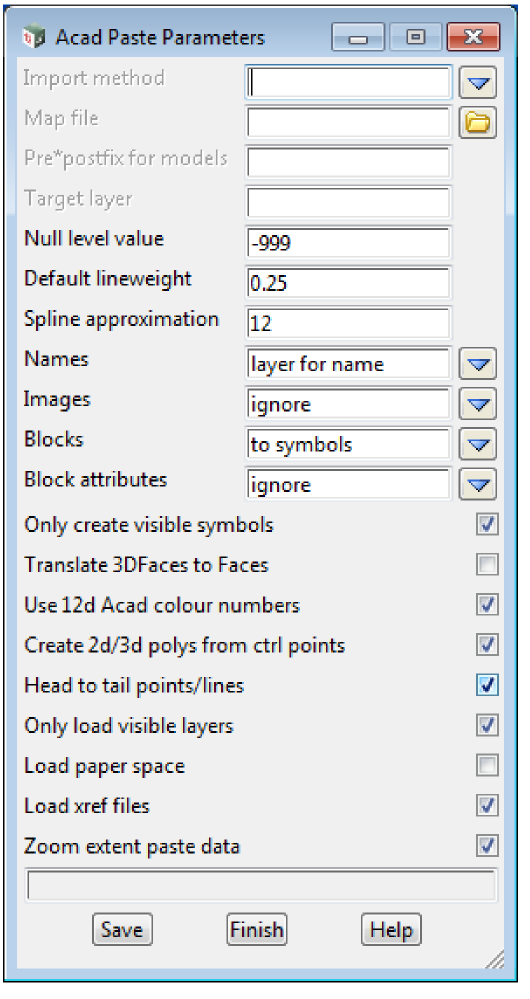
Copy button

Copies the selected data to the clipboard.

15.26.4 Paste parameters

The parameters for this panel are the same as parameters in the **Read DWG/DXF Data** panel.

Selecting **Paste parameters** brings up the **Acad Paste Parameters** panel:

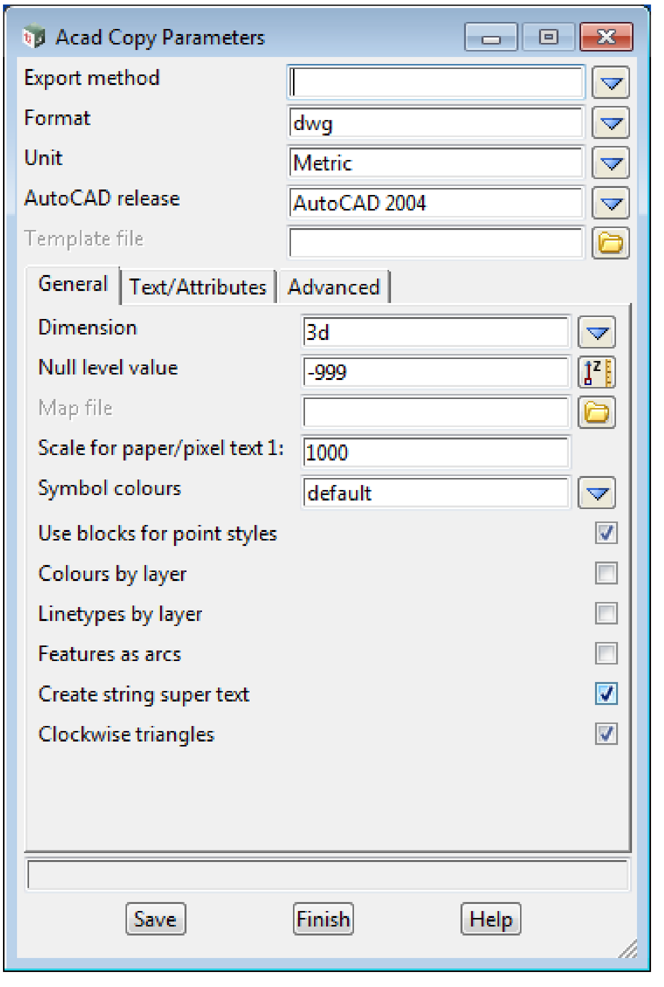


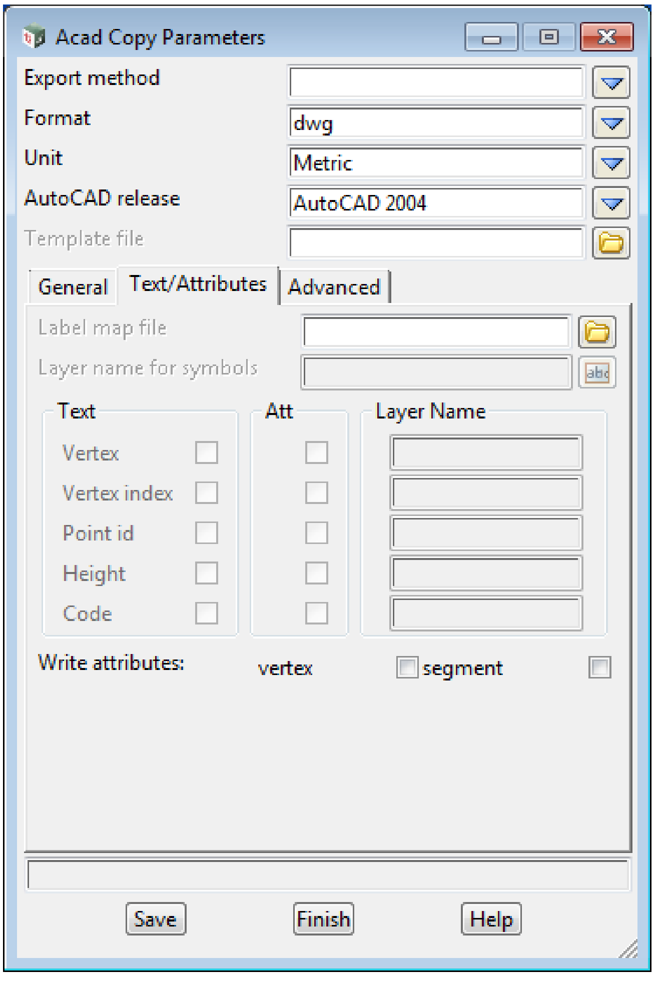
For more information please go to 8.1.10 DWG/DXF Input.

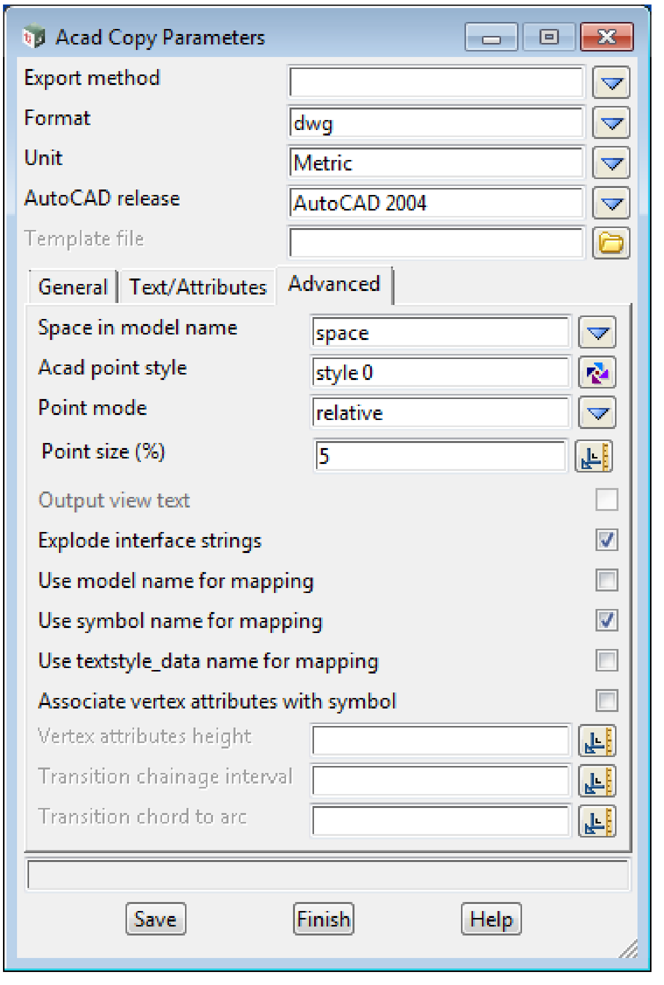
15.26.5 Copy parameters

The parameters for this panel are the same as parameters in the **Write DWG/DXF file for** panel.

Selecting **Copy parameters** brings up the **Acad Copy Parameters** panel:




Acad Copy Parameters
Export method
Format
dwg
Unit
Metric
AutoCAD release
AutoCAD 2004
Template file
General
Text/Attributes
Advanced
Label map file
Layer name for symbols
Text
Att
Layer Name
Vertex
Vertex index
Point id
Height
Code
Write attributes:
vertex
segment
Save
Finish
Help



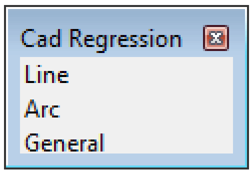
For more information please go to 8.2.7 Output DWG/DFX/DXB Files.

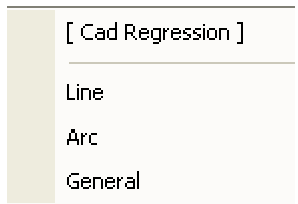
15.27 CAD Regression

Position of option on menu: **CAD =>Regression**

This section of documentation is a work in progress and will be updated in subsequent releases.

The **Cad Regression** walk-right menu is





For the option *Line*, go to 15.27.1 Line
Arc 15.27.2 Arc
General 15.27.3 General

15.27.1 Line

This section of documentation is a work in progress and will be updated in subsequent releases.

15.27.2 Arc

This section of documentation is a work in progress and will be updated in subsequent releases.

15.27.3 General

This section of documentation is a work in progress and will be updated in subsequent releases.

16 Tins

Position of menu: **Tins**

A **Tin** (triangulated irregular network) is an accurate method of representing 2.5D surfaces, especially those described by ground points or design strings. See 4.5.4 Tins.

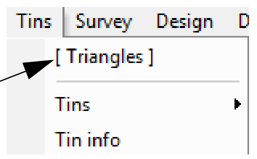
12d Model uses tins in most operations involving surfaces. For example, contouring, interfacing and volume calculations.

Unlike strings, tins can be in more than one model, or even no model at all. However, to be displayed in a view or used for profiling on a section view, tins need to be in at least one model. It is suggested that each tin be in its own model called “tin tin_name”. This makes it easy to know the model a tin is in, and also to see which models contain tins and obtain lists of all tins.

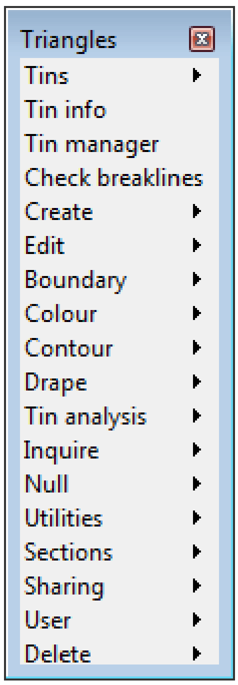
The **Triangles** walk-right menu is

on Main menu

create floating menu **Triangles**



on 12d Model menu and floating Triangles menu



list of tins
information on project tins
check crossing breaklines
triangulate a model or view, Super Tin
modify model list, *etc.* for tin
construct tin boundary
colour of tin, colour in polygon
contouring hts, depths, labelling
drape strings onto tin
slope, aspect, viewshed etc.
tin values at cursor position
remove unwanted triangles
various tin options
creating x-sections, mesh, long sec
tin sharing
Triangles User menus
delete tins from disk

Each of these options will now be discussed.

For the option *Tins*, go to 16.1 Tins
Tins info 16.2 Tin Info
Check breaklines 16.3 Check Breaklines, Duplicate Vertices, Identicals
Create 16.5 Create
Edit 16.6 Edit
Boundary 16.7 Boundary
Colour 16.8 Colour
Contour 16.9 Contour
Drape 16.10 Drape
Tin analysis 16.11 Tin Analysis
Inquire 16.12 Inquire
Null 16.13 Null

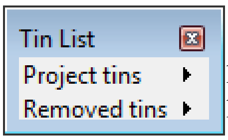
Utilities	16.14 Utilities
Sections	16.15 Sections
Sharing	16.16 Sharing
Delete	16.17 Delete

16.1 Tins

Position of menu: **Tins=>Tins**

The **Tins** walk-right menu provides options to list all the tins in the project (project tins) and all the tins in the project area but not in the project (removed tins).

The **Tin List** walk-right menu is



list of tins in project
list of removed tins in project area

For the option Project tins, go to 16.1.1 Project Tins
Removed tins 16.1.2 Removed Tins

16.1.1 Project Tins

Position of option on menu: **Tins =>Tins => Project Tins**

The **project tins** walk-right menu provides a list of all the tins in the project and there is a walk-right on each tin name showing what models the tin is in. If a tin name is selected from the list of tins, the **tin information** panel is fired up with the selected tin name already in the **tin** field.

16.1.2 Removed Tins

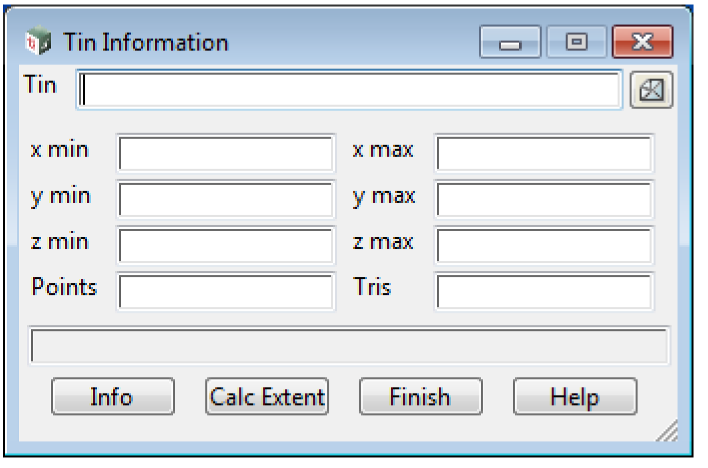
Position of option on menu: **Tins =>Tins => Removed Tins**

The **removed tins** walk-right menu provides a list of all the tins in the project area that have been removed from the project (using the **removed from project** option).

16.2 Tin Info

Position of option on menu: **Tins =>Tin info**

Selecting **Tin info** fires up the **Tin information** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
Tin	tin box		available tins

input the name of the tin to get information on

xmin, xmax	output
ymin, ymax	output
zmin, zmax	output

returns the tin x, y, z limits

Points output

returns the number of points in the tin

Tris output

returns the number of triangles in the tin

Info button

get the information for the tin given in the **tin** *field.*

Calc Extent button

recalculate the x, y, z bounding box for the tin given in the model field.

How to Use the Panel

The information for the tin given in the **tin** field is retrieved and placed in the appropriate panel fields when the tin name is piped into the **tin** field from the pop-ups, or an <enter> is entered after typing the tin name into the **tin** field, or on selecting the **info** button.

16.3 Check Breaklines, Duplicate Vertices, Identicals

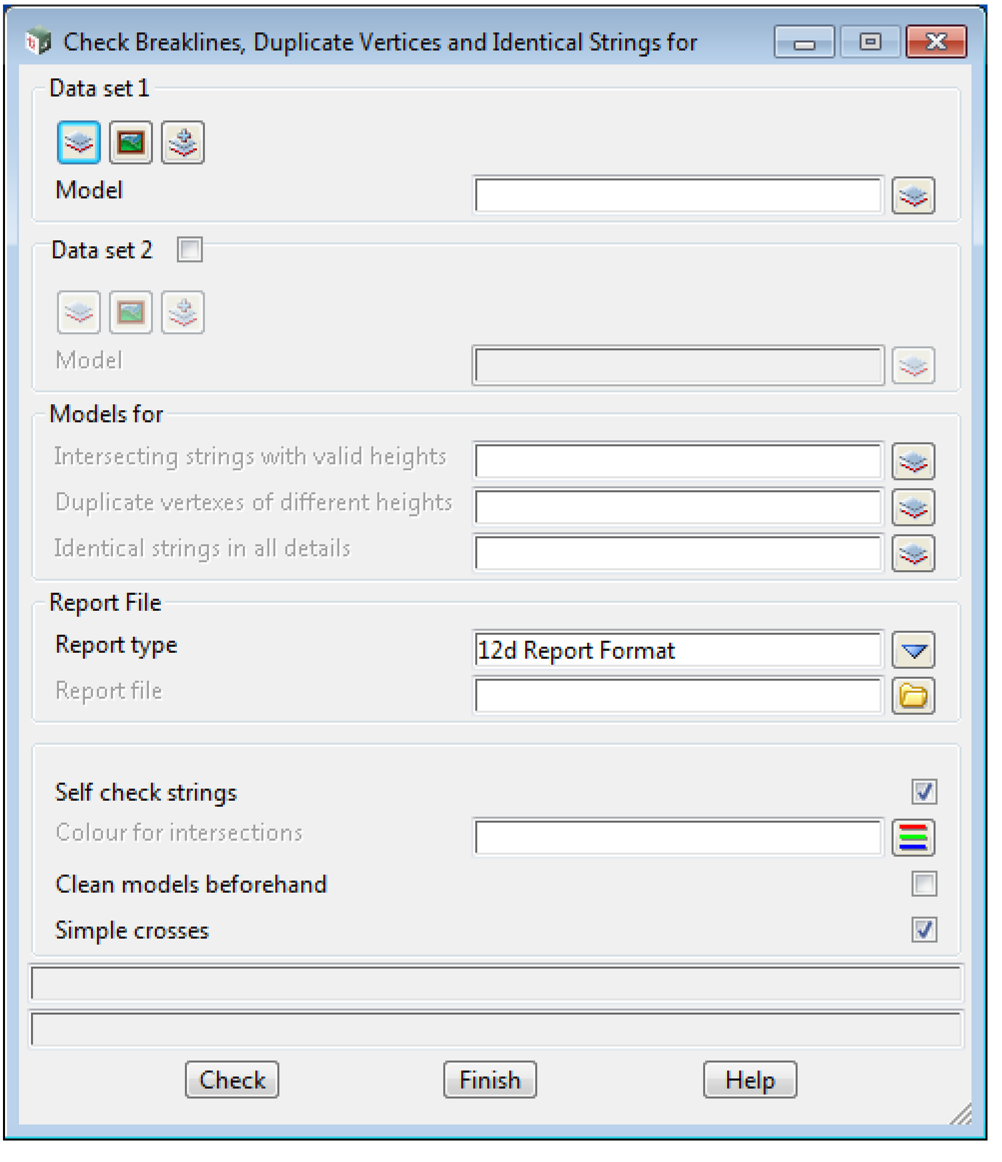
Position of option on menu: **Tins =>Check breaklines**
Utilities =>A-G =>Check/clash =>Check breaklines

The **Check breaklines** option is used to test whether any of the line segments from any line strings in a model or view intersect (cross) the line segments from any other string in the model/view. This includes self intersections of strings.

The option also checks for any points with the same x and y coordinates (same plan position) but different z-values, and for totally duplicated strings (identicals).

Intelligent log lines are written to the output window for showing any crossing breaklines, duplicate vertices, *etc.*

Selecting **Check breaklines** displays the **Check Breaklines, Duplicate Vertices and Identical Strings for** panel.



The fields and buttons in the panel have the following functions.

Field Description Type Defaults Pop-Up

Data set 1

*if **Data set 2** is ticked **off**, then all the selected strings in Data set 1 are checked against all the other selected strings from Data set 1.*

*If **Data set 2** is ticked **on**, then all the selected strings in Data set 1 are checked against all the selected strings in Data set 2, BUT the strings in Data set 1 are not checked against each other and the strings in Data set 2 are not checked against each other.*

Data set 1 source type Model

data selection type - for a full description go to 4.19.3 Data Source.

Data set 1 source input

data to be processed.

Data set 2

*if **Data set 2** is ticked **off**, then all the selected strings in Data set 1 are checked against all the other selected strings from Data set 1.*

*If **Data set 2** is ticked **on**, then all the selected strings in Data set 1 are checked against all the selected strings in Data set 2, BUT the strings in Data set 1 are not checked against each other and the strings in Data set 2 are not checked against each other.*

Data set 2 source type Model

data selection type - for a full description go to 4.19.3 Data Source.

Data set 2 source input

data to be processed.

Intersecting strings with valid heights model box available models

if non-blank, check for any crossing line segments and place either a copy of the crossing segments into the model given in this field or a diamond depending on the state of Simple crosses.

Duplicate vertices of different heights model box available models

*if non-blank, **copies** of any strings that are completely duplicated are placed in the model given in this field and circles are placed in this model at any duplicate points that aren't from an entire string.*

Identical strings in all details model box available models

*if non-blank, the duplicates of any strings are **moved** to model given in this field. That is, if any strings are identical in all ways, then the second and subsequent identical strings are moved to the this model. This is especially for the case when a second copy of some data has been supplied.*

Self check strings tick box tick

if ticked, a string is not check against itself for crossing breaklines etc.
If not ticked, a string is not checked against itself for crossing breaklines. This speeds up processing.

Colour for intersections colour box available colours

colour for the copies of the crossing segments.

Clean models beforehand tick box tick

*if **ticked**, the models for intersecting string, duplicate points, identicals are cleaned before the option is run.*

Report type choice box 12d Report Format, original xml

if 12d Report Format, the xml format is converted to the 12d Report Format for check breaklines

if original xml, report is in xml format

Report file file box

if non-blank, a report file of this name is created giving details of all the crossing breaklines and *duplicate plan points with different z-values.*

Simple crosses tick box tick

*if **ticked**, create diamonds at the position where strings cross, otherwise create a string in the shape of a cross with parts of the crossing strings.*

Check button

after selecting the **check** *button, all the strings in the model/view are tested for any crossing line segments or duplicate points. If requested, a report is generated.*
<esc> can be used to abort the checking option.

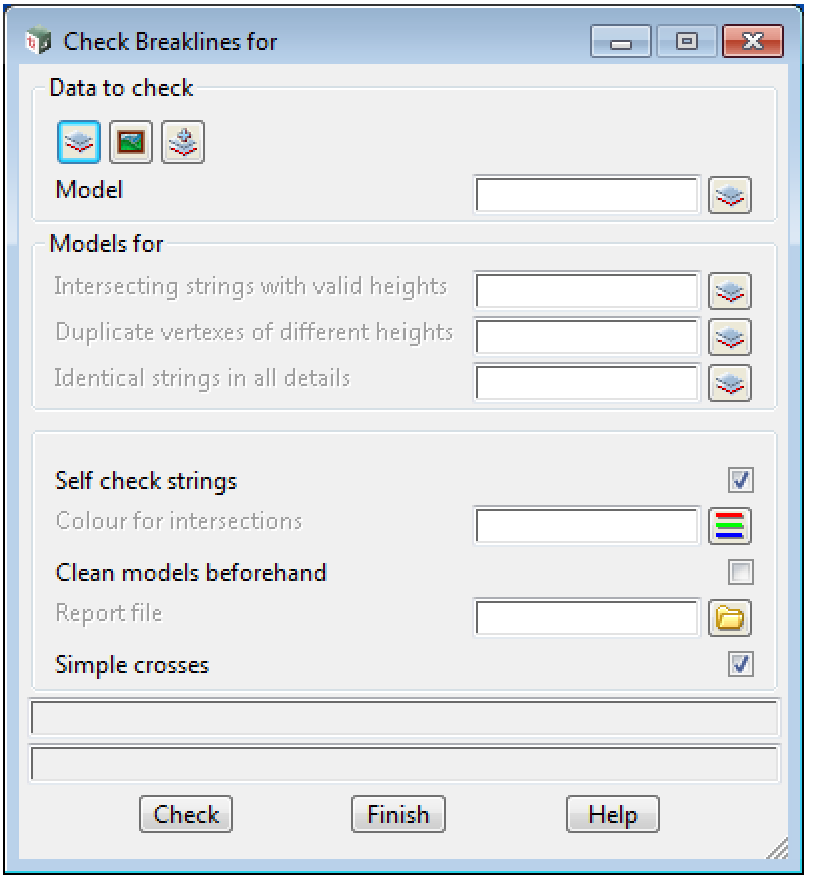
16.4 Check Breaklines - Old

Position of option on menu: **Utilities =>Old =>Check breaklines**

The **check breakline** option is used to test whether any of the line segments from any line strings in a model or view intersect (cross) the line segments from any other string in the model/view. This includes self intersections of strings.

The option also checks for any points with the same x and y co- ordinates (same plan position) but different z-values, and for totally duplicated strings.

On selecting **Check breaklines** option and then the appropriate **Data Source** in the panel, the **Check Breaklines for** panel is displayed.



The fields and buttons in the panel have the following functions.

Field Description   Type   Defaults   Pop-Up

Data source type   Model

data selection type - for a full description go to 4.19.3 Data Source.

Data source   input

data to be processed.

Intersecting strings with valid heights   model box   available models

if non-blank, check for any crossing line segments and place either a copy of the crossing segments into the model given in this field or a diamond depending on the state of Simple crosses.

Duplicate vertices of different heights   model box   available models

*if non-blank, **copies** of any strings that are completely duplicated are placed in the model given in this field and circles are placed in this model at any duplicate points that aren't from an entire string.*

Identical strings in all details model box available models

*if non-blank, the duplicates of any strings are **moved** to model given in this field. That is, if any strings are identical in all ways, then the second and subsequent identical strings are moved to the this model. This is especially for the case when a second copy of some data has been supplied.*

Self check strings tick box tick

if ticked, a string is not check against itself for crossing breaklines etc.
If not ticked, a string is not checked against itself for crossing breaklines. This speeds up processing.

Colour for intersections colour box available colours

colour for the copies of the crossing segments.

Clean models beforehand tick box tick

*if **ticked**, the models for intersecting string, duplicate points, identicals are cleaned before the option is run.*

Report file file box

if non-blank, a report file of this name is created giving details of all the crossing breaklines and duplicate plan points with different z-values.

Simple crosses tick box tick

*if **ticked**, create diamonds at the position where strings cross, otherwise create a string in the shape of a cross with parts of the crossing strings.*

Check button

after selecting the **check** *button, all the strings in the model/view are tested for any crossing line segments or duplicate points. If requested, a report is generated.*

<esc> can be used to abort the checking option.

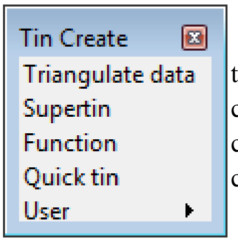
16.5 Create

Position of menu: **Tins =>Create**

The input data for a triangulation is either all the data in a selected model or all the data from all the models on a selected view or a list of models. The options for each method of selecting data are on the **Create** walk-right menu.

A Super Tin™ is a list of Tins and a Tin function creates a re-calc function name for an existing Tin. Each of these options are also on the **Create** walk-right menu.

The **Create** walk-right menu is



triangulate a model or view of data
create a supertin
create a triangulate function
create a quick tin of selected strings

For the option	*Triangulate data*, go to	16.5.1 Triangulate Data
	Supertin	16.5.2 Create Super Tin™
	Function	16.5.3 Function
	Quick tin	16.5.4 Quick Tin

16.5.1 Triangulate Data

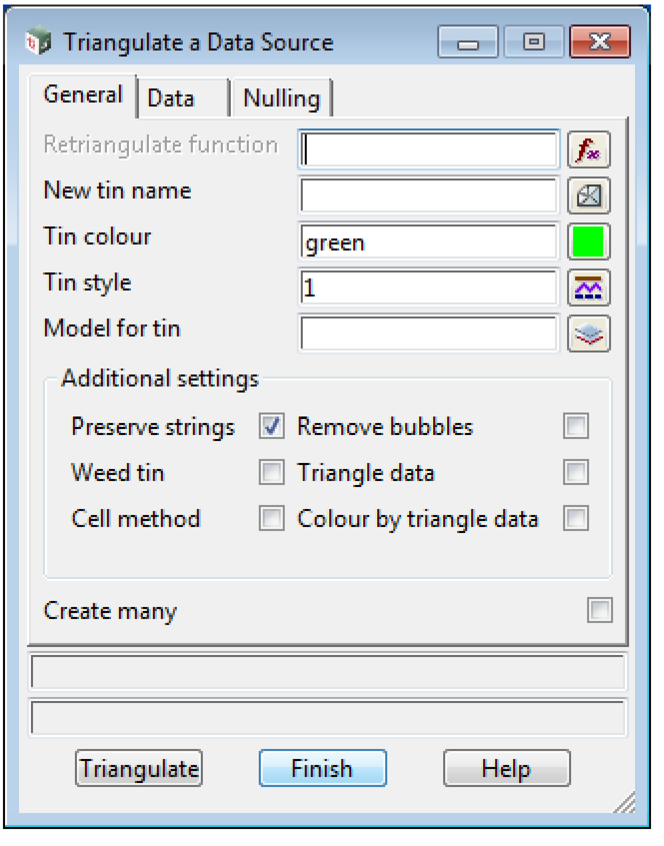
Position of option on menu: **Tins =>Create =>Triangulate data**

For simple jobs, the data to triangulate is all in the one model. However, it often occurs that the data one needs to triangulate is not in one model but is in the models displayed on a particular view.

The **Triangulate data source** option allows use to either select the data to triangulate from a single model, a list of models or to it uses the data points from all the models attached to a particular view, rather than the data from a particular model.

In each case, the models used in creating the triangulation are recorded so that the models can be easily retriangulated if any of the data changes.

Selecting **Triangulate data** displays the **Triangulate a Data Source** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

General tab

Retriangulate function function box box

if non blank, name of the tin function. The function can be recalced to recreate a new tin from the current data in the models used to construct the original tin. This is optional.

New tin name input

name of the tin created from the triangulation of the model/view.If <enter> is type after the name, the **model for tin** *field is filled out with the name* tin new_tin_nam***e***.

Tin colour input default tin colour available colours

colour to draw the triangles whenever they are displayed in a view. This is called the ***base tin colour****. It is possible to change the colour of individual triangles so they are different to base tin colour.*

Tin style input 1 available line styles

line style used for drawing sections through the tin on the section view.

Model for tin input available models

if non-blank, the created tin will be added to the model given in this field.
if blank, the name tin new_tin_name *will be placed in the field when the* **triangulate** *button is selected.*

Preserve strings tick box tick

*if **ticked**, all strings with breakline type **line** or segment type **tinable** will be preserved as sides of triangles. Otherwise, all line strings or segments of type tinable will be treated as point strings for the triangulation.*

Remove bubbles tick box

*if **ticked**, post-processing occurs to try and stop all the points of a triangle coming from the same string. This helps prevents triangles with all three points coming from the one contour string and hence forming a flat triangle.*

Weed tin tick box

*if **ticked**, all duplicate points all removed from the tin database. This helps to reduce the size of the tin files if there was a large amount of duplicate data. This flag is automatically set to tick for triangle data.*

Triangle data tick box

*if **ticked**, then it is assumed the original data was triangles and 12d will ensure that the triangles produced by the triangulation exactly match (including null regions) the original triangle data.* **Weed tin** *is automatically set to tick if* **face data** *is set to tick.*

Cell method tick box

*if **ticked**, try to create the triangles in cells - an alternate method of ordering the data which is often faster when the data comes in lines rather than randomly.*

Colour by triangle data tick box

*if **ticked** and* **Triangle data** *is ticked, the colour of each triangle in the created tin is made the same as the triangle from the triangle data.*

Create many tick box

*if **ticked**, once the tin has been triangulated, the panel will stay up so you can create another tin.*
*If **not ticked**, after the tin has been created, the panel is replaced by a Retriangulate Tin panel so if there are any mods to be made, no time is wasted in needed to select the panel.*

Data tab

Data source type Model

data selection type - for a full description go to 4.19.3 Data Source.

For this option, only data types involving models can be used - model, view, model list and the models from a favourite.

Data source input

source of data is to be triangulated - only Model or View allowed for this option.

Data polygon string-select

if a string is selected as a data polygon, only string points that are inside the data polygon are used in the triangulation.
If no string is selected, then all the points in all the model/view are used in the triangulation.

Nulling tab

Apply nulling tick box

if ticked, null the resultant tin using Angle*,* Length*,* Combined angle and Combined length*.*

Angle angle box 5

if a triangle has an external side (that is not a breakline) with an angle on it less than **Angle***, then the triangle is nulled. The default value works most of the time.*

If blank then no triangles are nulled by this test.

Length real value 100

if a triangle has an external side (that is not a breakline) greater than **Length***, the triangle is nulled.*

If blank then no triangles are nulled by this test.

Combined angle angle box 60
Combined length real box 20

For the Combined case, a triangle is nulled if it:

has an external side (that is not a breakline) and the ***sum*** *of the two angles on it is* ***less*** *than* **Combined angle** *(the default value works most of the time)*

and

has an external side (that is not a breakline) whose length is ***greater*** *than* **Combined length***. A suggested value is one third to one half of Length.*

If either **Combined angle** *or* **Combined length** *is blank, then no triangles are nulled by this test.*

Null polygon string-select

if a string is selected as a null polygon, any triangle whose centroid is outside the null polygon is automatically nulled.

Button at bottom

Triangulate button

Triangulate the model/view given in the **model/view to triangulate** *field. The created tin will be stored with the name given in the* **new tin name** *field. The tin will be the colour given in the tin colour field. If the* **model/view for tin** *field is non-blank, the created tin will be added to the model given in that field - if the model does not exist, it will be created.*

Panel Messages

Progress messages - sent to the panel message area

no of points in model/view = *no. of points*
no. points pts *no. break lines* break lines
no. points pts *no. of secs* s *no.* pts/s *no.* dup
break lines finished. time taken = *secs*

Completion message - sent to the panel message area

average no of pts per sec = *no.*

<esc> can be used to abort the triangulation option.

16.5.2 Create Super Tin™

Position of option on menu: **Tins =>Create =>Supertin**

The **SuperTin**™ option is designed to create a super triangulation from other triangulations.

With Super Tins, existing triangulations can be combined to form the equivalent of a merged triangulation without having to completely re-build a merged model and retriangulating it.

This means there are less steps in creating a Super Tin merged triangulation so a lot less time is required and there is less opportunity for errors. Also, if any of the component triangulations of the Super Tin are modified, then the Super Tin is also instantly modified.

The Super Tin is defined by a list of triangulations in increasing number order. Wherever two triangulations overlap, the triangulation of higher number takes precedence and is used to defined z-values and sections.

Hence, for any (x,y) point, the z-value on the Super Tin is defined to be the z-value from the triangulation of the highest number which is under the (x,y) point.

A section through a Super Tin is made up of the sections through the tin of highest number at each point along the section and so may involve more than one tin.

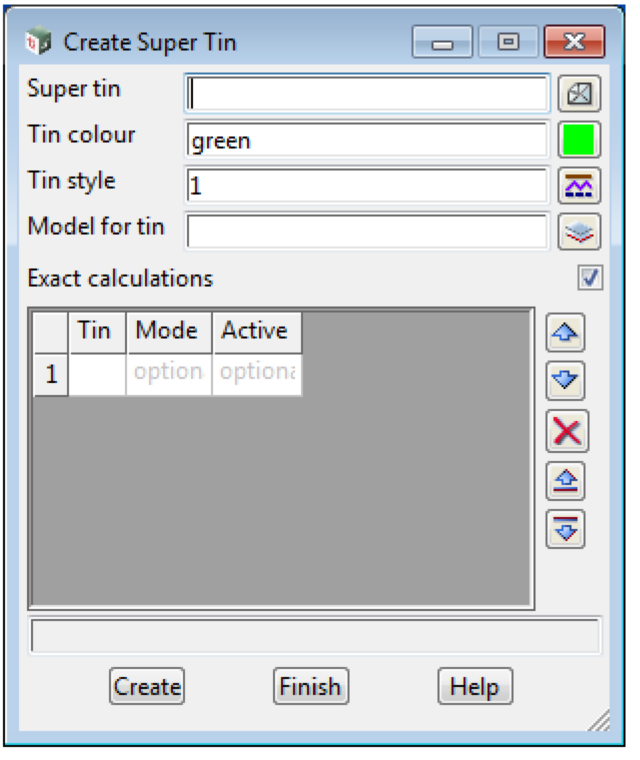
All Super Tins support Sections, Apply MTFs, Site Lines and volumes by end area calculations.

For a Super Tin, there is also an **Exact calculations flag** and if this is set to on, extra calculations are done when creating a Super Tin (and every time a tin included in a Super Tin is modified) so that the Super Tin can be used for Exact Volumes and Visualisation.

So Super Tins with the **Exact calculations** flag set support sections, Apply MTFs, site lines, and volumes by end area methods, as well as volumes by exact methods, and can also be used in visualisations.

Note - the reason that **Exact calculations** may be turned off is that for some tins, it may add significant processing time whenever a tin in the Super Tin is changed. So if that is slowing down your project and you don't need exact volumes or visualisations all the time, turn off **Exact calculations** for the Super Tin. **Exact calculations** can always be turned back on when you need them.

Selecting **Supertin** brings up the **Create Super Tin** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Super tin	super tin box		

name of the tin created from the triangulation of the model/view.If <enter> is type after the name, the **model for tin** *field is filled out with the name* tin new_tin_name*.*

Tin colour	colour box	default tin colour	available colours

the **tin colour** *is used as the section colour when sectioning through the Super Tin.When displayed on plan or section views, the each tin making up the Super Tin is drawn in its own colour.*

Tin style	linestyle box	1	available line styles

line style used for drawing sections through the Super Tin on the section view.

Model for tin	model box		available models

if non-blank, the created Super Tin will be added to the model given in this field.
if blank, the name tin new_tin_name *will be placed in the field when the* **Create** *button is selected.*

Exact calculations	tick box		

if ticked, when a Super Tin is created or a tin included in a Super Tin is modified, extra calculations are done for the Super Tin so that is can be used for Exact Volumes and in visualisations.

If ***not ticked****, no extra calculations are done for the Super Tin and it can only be used in options that use sections. For example, end area volumes and site lines.*

Tin Grid

Tin	table		available tins

name of the tin to be included in the Super Tin. The order is critical since the higher number tins take precedence over the lower number tins.

There can be up to 127 tins in a Super Tin.

Mode			replace, subtract

Replace

where ever this tin exists, use it instead of any tins of a lower number in the Tin column of the grid.

Subtract

where ever this tin exists, all parts of the tins of a lower number in the Tin column, are removed. So where ever the tin exists, it creates holes in all the tins of a lower number.

An example would be a tin that is the position of many gully pits. Using this tin in subtract *mode will punch holes in all the tins of a lower number in the Tin column. So in effect, it does a nulling that also nulls parts of triangles from the tins of a lower number.*

Active

if yes*, the tin is included in the Super Tin.*

If no, the tin is not included in the Super Tin. So the tin can be taken out of the Super Tin without having to remove it from the Tin list.

Create button

create a Super Tin from the tins in the **Tins to include** *table. The created Super Tin will be stored with the name given in the* **New tin name** *field.*

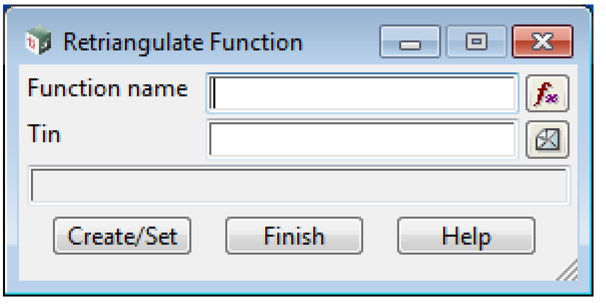
If there are no errors in creating the Super Tin, the panel then turns into a 16.6.2 Edit Super Tin *panel to allow any other edits to be made before finishing.*

16.5.3 Function

Position of option on menu: **Tins =>Create =>Function**

The **function** option is used to construct a function which when recalced, will run a retriangulate on the tin. Since a function name can be given when a tin is first created, this option is rarely needed.

Selecting **function** displays the **Retriangulate Function** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function name	input		available functions

the name to give to the retriangulate function.

Tin	input		available tins

name of the tin to be retriangulated when the function is recalced.

Create/Set	button		

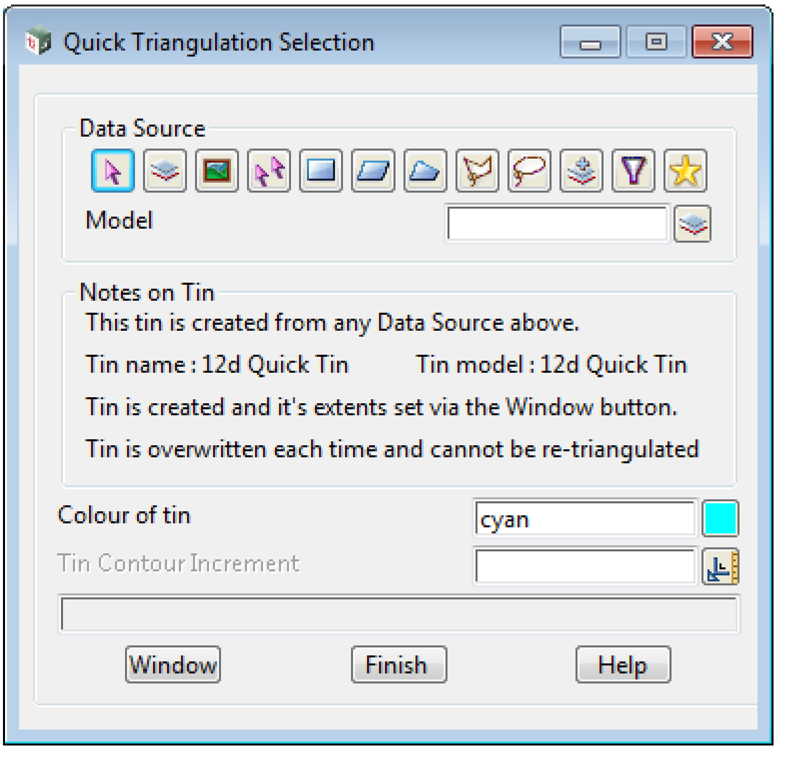
create or modify the retriangulate function.

16.5.4 Quick Tin

Position of option on menu: **Tins =>Create =>Quick tin**

The **Quick tin** option create a temporary tin from strings selected in a data source, and restricted to a user selected rectangular window. Contours of a user given increment can also be generated. Each time the option is run, is creates a tin called ***12d Quick Tin*** in the model ***12d Quick Tin*** so the tin is over written each time the option is run.

Selecting **Quick tin** displays the **Quick Triangulation Selection** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data is to be triangulated - only Model or View allowed for this option.

Colour of tin	colour box		available colours

the colour or the tin, and any contours.

Tin contour increment	real value box		

in non blank, contours are created at the given increment and displayed on the view where the Window is created. Fast contours are turned on for the view so contours are displayed for the tin.

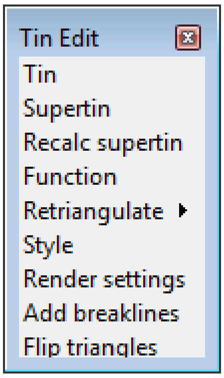
Window	button		

after clicking on Window, two corners to define a window are selected in a view. The tin for the selected data restricted to the Window is then created with the name 12d Quick Tin*, added to the model* 12d Quick Tin, *and then the model added to the view that the Window was defined. Tin contours is also turned of for the view.*

16.6 Edit

Position of menu: **Tins =>Edit**

Once a tin is created, it can be edited in a variety of ways. The **Edit** walk-right menu is



edit a tin definition and retriangulate
edit Super Tin definitions
recalculate a Super Tin
modify a retriangulate function
retriangulate an existing tin
set tin style
settings for rendering V6.1
add breakline
swap triangle edges

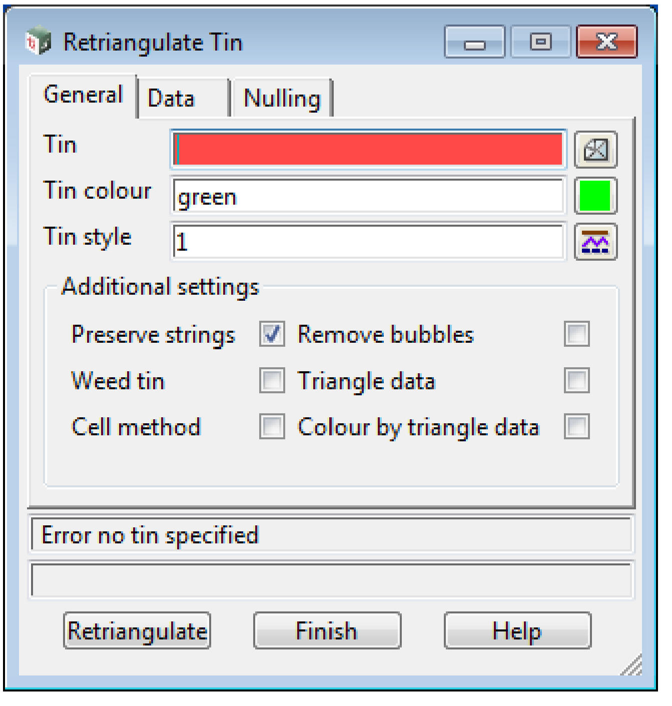
For the option *Tin*, go to	16.6.1 Edit a Tin
Supertin	16.6.2 Edit Super Tin
Recalc supertin	16.6.3 Recalc Super Tin
Function	16.6.4 Edit Tin Function
Retriangulate	16.6.5 Retriangulate
Style	16.6.6 Style
Render settings	16.6.7 Render Settings
Add breaklines	16.6.8 Add Breaklines
Flip triangles	16.6.9 Flip Triangles

16.6.1 Edit a Tin

Position of option on menu: **Tins =>Edit =>Tin**

When a tin is created, the settings and models used in the triangulation are recorded with the tin data. The **Edit** => **Tin** option can be used to modify the models and setting used to define the tin and then recreate the tin from the new models and settings.

Selecting **Edit** => **Tin** displays the **Retriangulate Tin** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

General tab

Tin	tin box	original tin name	

name of the tin to be retriangulated. When the tin name is given, the list of models used for the tin and other settings, are displayed.

Tin colour	colour box	original colour	available colours

original colour of the tin - this can be modified. Note that this only modifies the base tin colour and any triangles that have had their colour modified, will keep the modified colour.

Tin style	line style box	original line style	available line styles

original line style of the tin - this can be modified.

Preserve strings	tick box	original setting	

*if **ticked**, all strings with breakline type **line** will be preserved as sides of triangles. Otherwise, all line strings will be treated as point strings for the triangulation.*

Remove bubbles	tick box	original setting	

*if **ticked**, post-processing occurs to try and stop all the points of a triangle coming from the same string.*

Weed tin	tick box	original setting	

*if **ticked**, all duplicate points all removed from the tin database.*

Triangle data tick box original setting

*if **ticked**, only triangles in the tin that are under the centroid of a string representing a triangle in the model/view are considered valid, all others are set to null. That is, any triangles not under the centroid of a string from the model/view will be set to null. This is extremely useful when the original string data represents triangles and it will ensure that the triangles produced by the triangulation will exactly match, including null regions, the original triangle data.* **Weed tin** *is automatically set to tick if* **face data** *is set to tick.*

Cell method tick box original setting

*if **ticked**, try to create the triangles in cells - an alternate method of ordering the data for use when the triangulation takes too long by the non-cell method.*

Data tab

Model List models in the tin

list of models used to create the original tins. The models and the order can be modified.

Data polygon string-select original data polygon

if a string is selected as a data polygon*, only string points that are inside the* data polygon *are used in the triangulation.*
If no string is selected, then all the points in all the models are used in the triangulation.

Nulling tab

Apply nulling tick box original setting

if ticked, null the resultant tin using Angle*,* Length*,* Combined angle and Combined length*.*

Angle angle box original setting

if a triangle has an external side (that is not a breakline) with an angle on it less than **Angle***, then the triangle is nulled. The default value works most of the time.*

If blank then no triangles are nulled by this test.

Length real value original setting

if a triangle has an external side (that is not a breakline) greater than **Length***, the triangle is nulled.*

If blank then no triangles are nulled by this test.

Combined angle angle box original setting
Combined length real box original setting

For the Combined case, a triangle is nulled if it:

*has an external side (that is not a breakline) and the **sum** of the two angles on it is **less** than* **Combined angle** *(the default value works most of the time)*
and

*has an external side (that is not a breakline) whose length is **greater** than* **Combined length***. A suggested value is one third to one half of Length.*

If either **Combined angle** *or* **Combined length** *is blank, then no triangles are nulled by this test.*

Null polygon string-select original null polygon

if a string is selected as a null polygon*, any triangle whose centroid is outside the* null polygon *is automatically nulled.*

Retriangulate button

The original tin is deleted and the data in the **models to include** *table used to create a new triangulation which is given the original tin name. The settings in this panel are used for the retriangulation.*

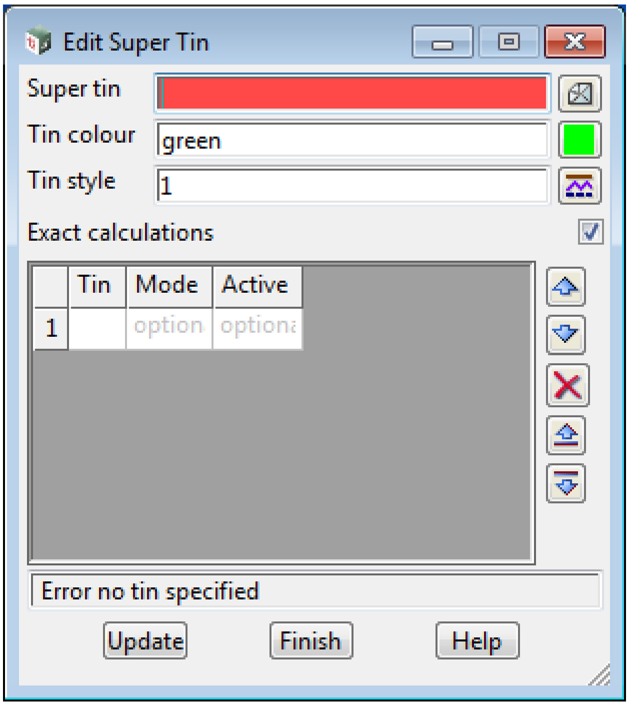
16.6.2 Edit Super Tin

Position of option on menu: **Tins =>Edit =>Supertin**

When a Super Tin is created, the tins and settings used to create the Super Tin are recorded.

The **Edit=>Supertin** option modifies the tins and setting used to define the Super Tin.

On selecting the **edit=>supertin** option, the **edit super tin** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Super tin name	sup tin box	original super tin name	existing Super Tins

name of the Super Tin to be modified.

Tin colour	colour box	existing super tin colour	available colours

original colour for drawing sections through the Super Tin - this can be modified.

Tin style	line style box	existing Super Tin style	available line styles

original linestyle for drawing sections through the Super Tin - this can be modified.

Exact calculations	tick box		

if ticked, when a Super Tin is created or a tin included in a Super Tin is modified, extra calculations are done for the Super Tin so that is can be used for Exact Volumes and in visualisations.

*If **not ticked**, no extra calculations are done for the Super Tin and it can only be used in options that use sections. For example, end area volumes and site lines.*

Tin Grid

Tin	table		available tins

name of the tin to be included in the Super Tin. The order is critical since the higher number tins take precedence over the lower number tins.

There can be up to 127 tins in a Super Tin.

Mode replace, subtract

Replace

where ever this tin exists, use it instead of any tins of a lower number in the Tin column of the grid.

Subtract

where ever this tin exists, all parts of the tins of a lower number in the Tin column, are removed. So where ever the tin exists, it creates holes in all the tins of a lower number.

An example would be a tin that is the position of many gully pits. Using this tin in subtract *mode will punch holes in all the tins of a lower number in the Tin column. So in effect, it does a nulling that also nulls parts of triangles from the tins of a lower number.*

Active

if yes*, the tin is included in the Super Tin.*

If no, the tin is not included in the Super Tin. So the tin can be taken out of the Super Tin without having to remove it from the Tin list.

Update button

update the Super Tin definition with the information in the above panel fields.

16.6.3 Recalc Super Tin

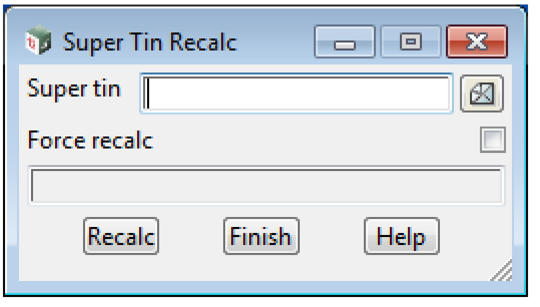
Position of option on menu: **Tins =>Edit =>Recalc supertin**

This option gets a super tin that has **Exact calculations** ticked on, to recalculate itself.

That is, the super tin updates all the interfaces with its component tins if there were any changes to the component tins.

The option can be used in a Chain.

Selecting **Edit=> Recalc supertin** displays the **Super Tin Recalc** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Super tin	super tin box		existing Super Tins

name of the Super Tin to be recalculated.

Force recalc	tick box		

if ticked and **Exact calculations** *is ticked on for the super tin, a recalc of the super tin is done.*
If not ticked and **Exact calculations** *is ticked on for the super tin, a recalc of the super tin is only done if the super tin believes it is necessary.*

If **Exact calculations** *is not ticked on for the super tin, then no recalc occurs.*

Recalc	button		

recalculates the super tin.

16.6.4 Edit Tin Function

Position of option on menu: **Tins =>Edit =>Function**

The **edit=>function** option is used to edit a tin function.

On selecting the **function** option, the **retriangulate function** panel is displayed which is the same as for the option **Tins =>Create =>Function** (see 16.5.3 Function).

16.6.5 Retriangulate

Position of option on menu: **Tins =>Edit =>Retriangulate**

When a tin is created, the settings and models used in the triangulation are recorded with the tin data. The **retriangulate** option is used to recreate the tin from the same models and settings.

The **retriangulate** menu item operates two ways.

(a) The **retriangulate** walk-right brings up the **tins** menu which is a list of all the existing tins.

By choosing one of the tins from the **tins** menu, the tin is retriangulated.

(b) If **retriangulate** itself is activated (by clicking LB when **retriangulate** is highlighted), the **retriangulate tin** panel appears. The **retriangulate tin** panel can be used to modify the tin definition and retriangulate the tin.

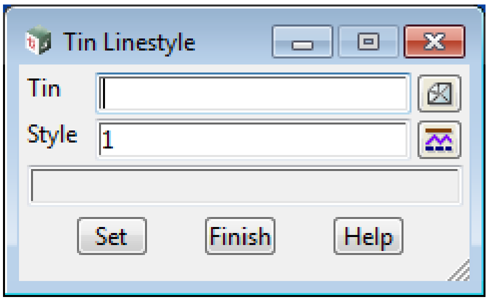
16.6.6 Style

Position of option on menu: **Tins =>Edit =>Style**

When a tin is created, it is given a linestyle (the tin linestyle) which is used for drawing the tin whenever it is profiled on a section view or plotted in long and cross section plots.

The **style** option is used to modify the tin linestyle.

On selecting the **style** option, the **tin linestyle** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin to modify the linestyle of.

Style	linestyle box	1	available linestyles

linestyle for the tin.

Set	button		

set the tin linestyle to that given in the **style** *panel field.*

16.6.7 Render Settings

Position of option on menu: **Tins =>Edit =>Render settings**
Position of option on menu: **Tins =>Visualiation =>Tin render settings**

This option sets blending, texture mappings, raster draping, one/two sides, draping of polygons, text, billboards and images for triangles in a tin. It has already been documented as

Tins =>Visualiation =>Tin render settings

in the section 12.13.1 Tin Render Settings.

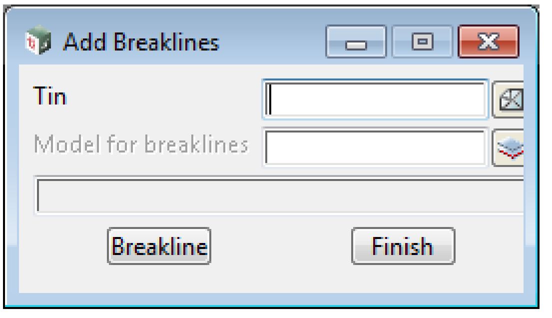
IMPORTANT NOTE: the ***Visualisation*** module is required for the *Tin Render Settings* to be used on a Perspective OpenGL view.

16.6.8 Add Breaklines

Position of option on menu: **Tins =>Edit =>Add breaklines**

This option allows extra breaklines joining existing vertices in the tin to be created. The new breaklines can be added to a model so that the results can be reproduced on a retriangulation.

Selecting **Add breakline** displays the **Add Breaklines** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin to add a breakline to.

Field Description	Type	Defaults	Pop-Up
Model for breaklines	model box	1	available models

if non-blank, then as breaklines are created with this option, then the breakline is automatically added to this model so that the breakline modifications will try to be preserved on a retriangulation (this may not be possible with crossing breaklines).

Field Description	Type	Defaults	Pop-Up
Breakline	button		

add a breakline by clicking near the two vertices of the tin that are to be connected by a breakline (cursor snap will do). As each breakline is added, the triangles are rearranged. If fast contours are turned on, the effect will be immediate. Note that you can have fast contours and tin edges toggled on at the same time.

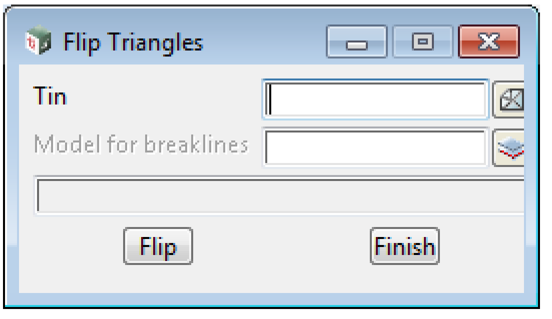
16.6.9 Flip Triangles

Position of option on menu: **Tins =>Edit =>Flip triangles**

This option works on a triangle side and if it is possible, it finds the two triangles with this common side, removes the common side and replaces it with a new common side which joins the non common triangle points in the original triangle. Note that this can't be done when the new side goes outside the two triangles.

A new breaklines can also be automatically added to a model so that the results can be reproduced on a retriangulation.

Selecting **Flip triangles** displays the **Flip Triangles** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin to add a breakline to.

Model for breaklines	model box	1	available models

if non-blank, then as triangles are selected and flipped, then breaklines are automatically added to this model so that the flip modifications will try to be preserved on a retriangulation (this may not be possible with crossing breaklines).

Flip	button		

flip a triangle by clicking near the common edge (cursor snap will do). As each flip is done, the triangles are rearranged. If fast contours are turned on, the effect will be immediate. Note that you can have fast contours and tin edges toggled on at the same time.

16.7 Boundary

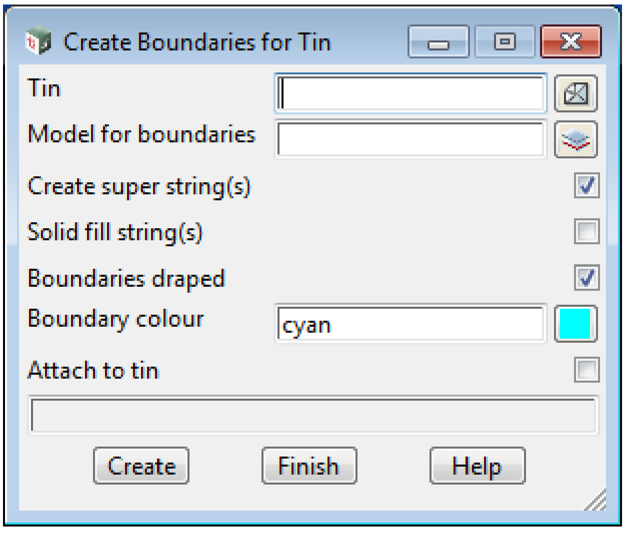
Position of option on menu: **Tins =>Boundary**

The **Boundary** option is used to construct the boundary strings for a tin, i.e. the strings going around the edges between the null and non-null triangles in the tin.

If the tin has holes (internal null regions), then there will be more than one boundary string created.

Often the boundary strings is used as a first step in creating a polygon to be used to null out further triangles not required in the tin.

Selecting **Boundary** displays the **Create Boundaries for Tin** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin (not a super tin) to calculate the boundary strings for.

Model for boundaries	model box		available models

the model to place the boundary strings in.

Create super string(s)	tick box	ticked	

*if **ticked**, the boundary strings are super strings.*

*If **not ticked**, the boundary strings are 3d strings.*

Solid fill string(s)	tick box	not ticked	

*if **ticked**, each of the boundaries is solid filled with the tin colour.*

Boundaries draped	tick box	ticked	

*if **ticked**, the points of the boundary have the z-values from the tin.*

*If **not ticked**, the z-value is set to 1.*

Boundary colour	input	cyan	available colours

the colour for the boundary strings.

Attach to tin	tick box	not ticked	

*if **ticked**, the boundaries are remembered with the tin and when you have the 250M version of*

12d Model and have Model Density turned on, if the density is such that the tin would only be drawn as a red rectangle, the boundaries are drawn instead of the red rectangle.

Create button

create the boundary polygons for the tin.

16.8 Colour

Position of menu: **Tins =>Colour**

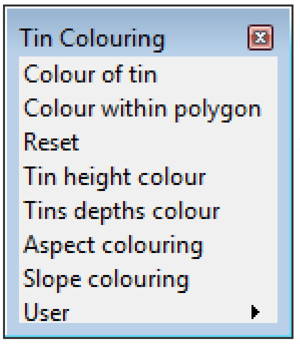
When a tin is created, all the triangles have the same colour (called the tin's **base** colour). This is the colour given by the user when creating the tin with the **Triangulate a Data Source** panel. The ***base*** colour can be changed at any time using the **colour of tin** option.

However, it often necessary to colour specific triangles in some other colour than the base colour. The option **Colour within polygon** is designed to do just that.

When the **Triangles** menu is torn off the main menu, the **Colour** menu item on it operates in two ways:

First, clicking on **Colour** without walking right brings up the **Colour of Tin** panel (see the next section). Secondly, the **Colour** walk-right brings up the **Tin Colouring** menu.

The **Tin Colouring** walk-right menu is



change base colour of tin
colour triangles within a polygon
reset all triangles to base colour
colour height ranges
colour depth ranges between two tins
colour aspect ranges
colour slope ranges

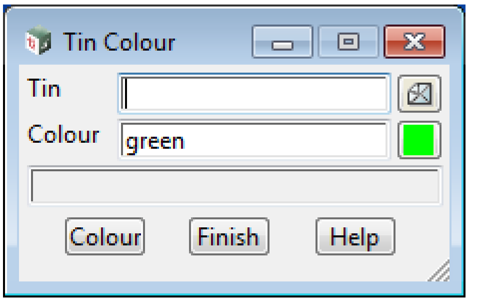
For the option *Colour of tin*, go to	16.8.1 Colour of Tin.
Colour within polygon	16.8.2 Colour Within Polygon
Reset	16.8.3 Reset
Tin height colour	16.8.4 Tin Height Colour
Tins depths colour	16.8.5 Tins Depths Colour.
Aspect colouring	16.8.6 Aspect Colouring.
Slope colouring,	16.8.7 Slope Colouring.

16.8.1 Colour of Tin

Position of option on menu: **Tins =>Colour =>Colour of tin**

A tin is given a base colour when it is originally constructed. The **colour of tin** option allows the user to change the base colour of the tin.

On selecting the **Colour of tin** option, the **Tin Colour** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin to have its base colour modified. The tin must exist in 12d Model. When the tin name is entered, the existing colour of the tin is displayed in the **Colour** *panel field.*

Colour	colour box	default tin colour	available colours

the new base colour for the tin given.

Colour	button		

colour all the base triangles with the colour given in the **Colour** *field.*

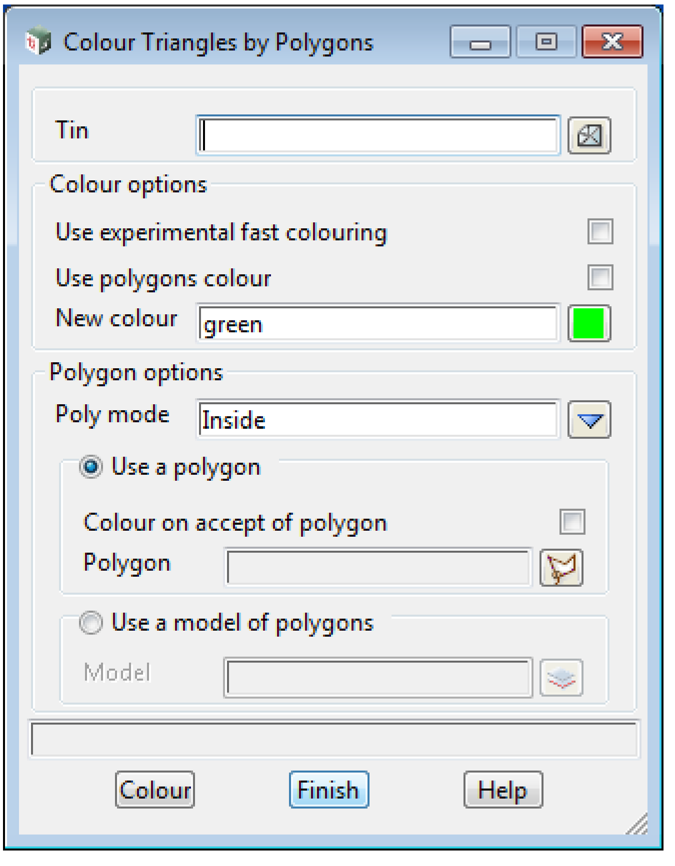
16.8.2 Colour Within Polygon

Position of option on menu: **Tins =>Colour =>colour within polygon**

Although a tin is given a base colour when constructed, it is useful to be able to change the colour of selected triangles. For example, in a design tin with a road and terrain, the road and the terrain could be displayed in different colours. This is particularly effective in shades.

The **colour within polygon** option allows the user to colour all the triangles whose centroids lie inside (or outside) a selected polygon, or for all the polygons in a given model, with either a user specified colour or the colour of each polygon. If a model of polygons is used, the polygons are processed in the order that they occur in the model. The option can be applied any number of times to a tin.

On selecting the **colour within polygon** option, the **colour triangles by polygons** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin to have some of its triangles coloured. The tin must exist in 12d Model.

Use experimental fast colouring tick box

*if **ticked**, an experimental fast colouring algorithm is used.*

Use polygons colour tick box

*if **ticked**, the colour of the polygons is used as the colour for the triangles.*

New colour	input	default tin colour	available colours

if **Use polygons colour** *is **not** set to tick, this is the colour to set all the triangles within/out a polygon to.*

Poly mode choice inside inside, outside

*mode to select whether the triangles with centroids **inside** or **outside** the selected polygons are to be coloured.*

Use a polygon tick box

*if **ticked**, then individual polygons are selected and the tin coloured within them.*

Colour on accept of polygon tick box

*if **ticked**, then the colouring is done as soon as a polygon is accepted.*

Polygon polygon select string pop-up

this is used to
(a) select an individual string to use as a polygon for colouring
(b) dynamically define a rectangle to use as a polygon for colouring
(c) dynamically define a parallelogram to use as a polygon for colouring
(d) dynamically define a lasso to use as a polygon for colouring
(e) edit a string already selected with the polygon select
(f) get information on a string already selected with the polygon select
(g) delete a string already selected with the polygon select
(h) clear any selection already made by the polygon select
(i) toggle on/off the checks made the string selected by the polygon select

Use a model of polygons tick box

*if **ticked**, then all the polygons in the given model are used for colouring.*

Model model box available models

model of polygons which are used for selecting triangles to be coloured. Colour happens when the **Colour** *button is selected.*

Colour button

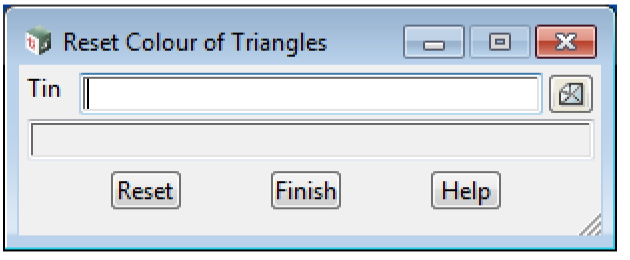
for a model of polygons, each string in the **Model of polygons** *is used to colour the triangles in the tin given in the* **tin** *field. If* **Colour on accept of polygon** *is off, colour within the selected string.*

16.8.3 Reset

Position of option on menu: **Tins =>Colour =>Reset**

The **reset** option is used to change all the non base-coloured triangles in a tin back to the base colour.

On selecting the **reset** option, the **reset colour of triangles** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin to have any non base-coloured triangles reset to the base colour. The tin must exist in 12d Model.

Reset	button		

change any selectively coloured triangles back to the base colour.

16.8.4 Tin Height Colour

Position of option on menu: **Tins =>Colour =>Tin height colour**

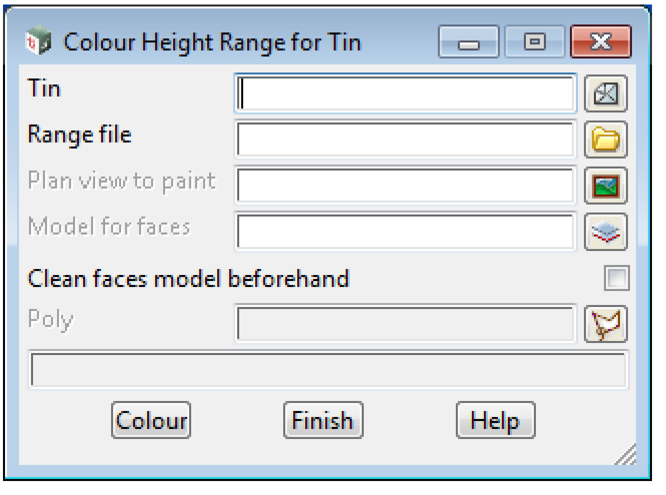
The **tin height colour** option is used to calculate height bands for the tin and colour a view and/or create faces for the height bands. The calculations can be restricted to within a user selected polygon, or if no polygon is selected, the entire tin is used.

The calculated heights and colours are given in a user supplied height range file. A plan view can be temporarily coloured on the height basis using the **range** file. The temporary colours will disappear next time the view is refreshed.

For permanency, the option can also create coloured faces with the appropriate range colour. The faces can be displayed on any plan view using a solid fill colour or a hatch pattern.

For more information on the height range file, see 8.9.6 Height Range File.

Selecting **Tin height colour** displays the **Colour Height Range for Tin** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin to calculate the height ranges of the triangles and colour a view.

Range file	height range file box		*.hrf

the user supplied range file is used to give the colour ranges for heights of the tin. See 8.9.6 Height Range File.

Plan view to paint	view box		available views

if non-blank, the given plan view will be painted according to the range colours given in the range file.

Model for faces	model box		available models

if non-blank, faces will be created with colours according to the range colours given in the range file.

Clean faces model beforehand tick box

if ticked, the model of faces is cleaned out before the option runs.

Poly polygon-select

if selected, this string is used as the bounding polygon for the height calculations.

Colour button

On selecting this button, the heights of the triangles of the tin within the selected bounding polygon are *calculated.*

<esc> can be used to terminate the option during height calculations.

16.8.5 Tins Depths Colour

Position of option on menu: **Tins =>Colour =>Tin depths colour**

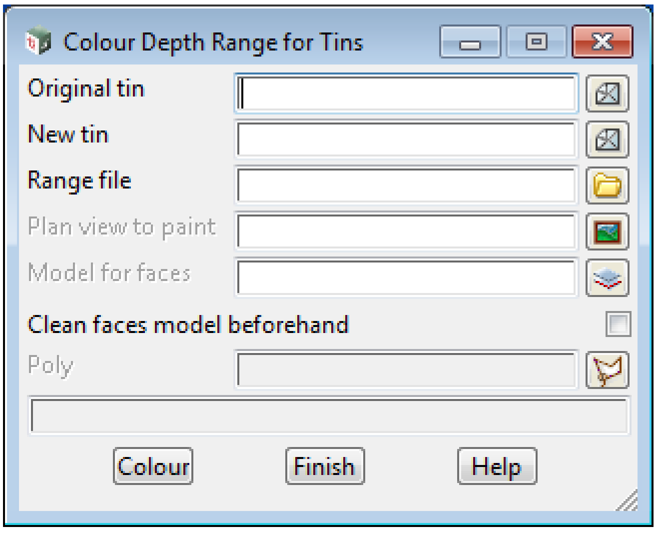
The **Tins depths colour** option calculates colour bands for the ***depths*** between two tins, and colours a view and/or create faces. The calculations can be restricted to within a user selected polygon, or if no polygon is selected, the common portions of the tins are used.

The calculated depths and colours are given in a user supplied depth range file. A plan view can be temporarily coloured on the depth basis using the **range** file. The temporary colours will disappear next time the view is refreshed.

For permanency, the option can also create coloured faces with the appropriate range colour. The faces can be displayed on any plan view using a solid fill colour or a hatch pattern.

For more information on the depth range file, see 8.9.4 Depth Range File.

On selecting the **Tins depths colour** option, the **Colour Depth Range for Tins** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Original/New tin	tin box		available tins

name of the original/new tin to colour depths between. Cut is when the new tin is below the original tin.

Range file	depth range file box		*.drf

the user supplied depth range file is used to give the colour ranges for depths. See 8.9.4 Depth Range File.

Plan view to paint	input		available views

if non-blank, the given plan view will be painted according to the range colours given in the range file.

Model for faces	input		available models

if non-blank, faces will be created with colours according to the range colours given in the range file.

Clean faces model beforehand tick box

if tick, the model of faces is cleaned out before the option runs.

Poly polygon-select

if selected, this string is used as the bounding polygon for the depth calculations.

Colour button

colour the depths between the tins within the selected bounding polygon.

<esc> can be used to terminate the option during depth calculations.

16.8.6 Aspect Colouring

Position of option on menu: **Tins =>Colour =>Aspect colouring**

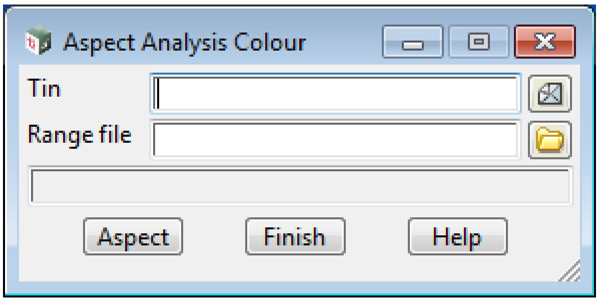
The **aspect colouring** option is used to colour the triangles according to the aspect (direction) of the triangles in the tin. An *aspect range file* is used to define the colour ranges.

Note that the actual colours of the triangles are changed - not a face model produced.

For more information on the aspect range file, see 8.9.5 Aspect Range File.

For each range in the file, the boundary strings for areas of common range can be created and also coloured faces that can be displayed on any plan view.

On selecting the **Aspect colouring** option, the **Aspect Analysis Colour** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin for which the aspects of the triangles will be calculated and then the tin coloured.

Range file	aspect range file box		*.arf

the user supplied range file is used to give the colour ranges for aspects to use to colour the triangles. See 8.9.5 Aspect Range File.

Aspect	button		

On selecting this button, the aspects of the triangles of the tin are calculated and the triangle coloured according to the range file.

<esc> can be used to terminate the option during aspect calculations.

16.8.7 Slope Colouring

Position of option on menu: **Tins =>Colour =>Slope colouring**

The **Slope colouring** option is used to colour the triangles in the tin according to the slope of the triangles. A slope range file is used to define the colour ranges.

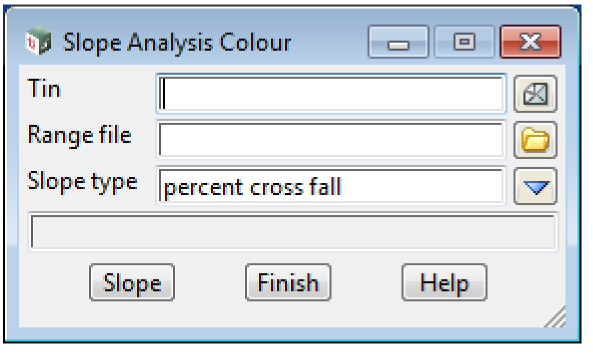
Note that the actual colours of the triangles are changed - not a face model produced.

For more information on the slope range file, see 4.24 Defaults.

The values can be percent cross fall, “1v in” slope or degrees (in 12d Model dms format).

The range_colour is used to colour all triangles in the tin satisfying the range.

On selecting the **Slope colouring** option, the **Slope Analysis Colour** panel is displayed.



The fields and buttons used in this panel have the following functions.

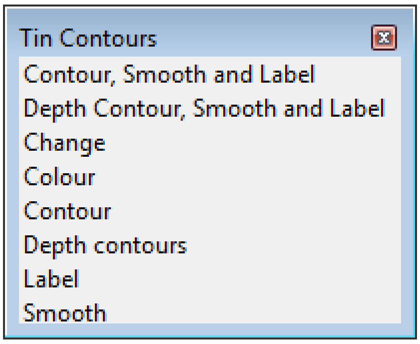
Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins
name of the tin for which the slope of the triangles will be calculated and then the tin coloured.			
Range file	slope range file		*.srf
the user supplied range file is used to give the colour ranges for slopes to use to colour the triangles. See 4.24 Defaults.			
Slope type	input	percent cross fall	percent cross fall, degrees, 1v in
the units used for slope in the range file.			
Slope	button		
On selecting this button, the slopes of the triangles of the tin are calculated and coloured according to the slope range file.			

<esc> can be used to terminate the option during aspect calculations.

16.9 Contour

Position of menu: **Tins =>Contour**

The **Tin Contours** walk-right menu is



create, smooth and label contours
create depth contours and also label and smooth them
change colour/model
colour contours using a height range file
create contours
create depth contours
label contours
smooth contours

For the option *Contour, smooth and label*, go to	16.9.2 Contour, Smooth and Label
Depth Contour, Smooth and Label	16.9.3 Depth, Contour and Label Function
Change	16.9.4 Change Contours
Colour	16.9.5 Colour
Contour	16.9.6 Contour
Depth contours	16.9.7 Depth Contours
Label	16.9.8 Label Contours
Smooth	16.9.9 Smooth

For information on creating contours, go to the section 16.9.1 Creating, Smoothing and Labelling Contours.

16.9.1 Creating, Smoothing and Labelling Contours

Creating Contours

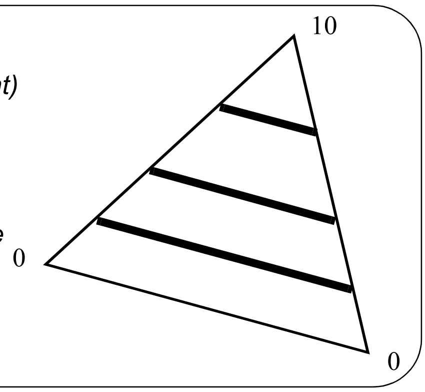
Contours (or level lines) for a surface are the strings of constant height. Contours are often produced on maps and labelled with the height of the contours.

A 12d Model tin represents a surface in three dimensions and there is an option in 12d Model for producing the contours for the tin.

For a triangle in three dimensions, a string of constant height is actually a straight line across the triangle and the contour lines of different heights for a triangle are parallel lines.

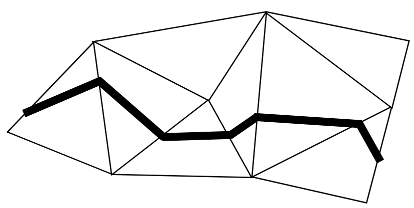
Contours (lines of equal height) on the triangle with heights 0, 0 and 10 at the triangle vertices.

The contours for a the triangle are parallel lines.



Because triangles in a tin have common sides, a contour of a given height for one triangle will connect to the contour of the same height in an adjacent triangle.

So the contour lines on adjacent triangles join up to form continuous contour strings, each segment of which is straight across a triangle.



Contour lines on adjacent triangles in a tin join up to form a continuous contour string.

The contours for a tin lie on the surface of the tin.

The options to create contours in 12d Model has five parameters to define which contours are calculated. They are the

minimum contour value
maximum contour value
contour increment
contour reference
bold increment

The elevation range over which the contours are to be produced is defined by the minimum and maximum contour values. If the minimum (maximum) value is not specified, the tin minimum (maximum) z-value is used.

The increment between successive contours is given by the contour increment.

For example, if contours are required every two metres from 100.0 metres to 200.0 metres, the parameters needed are

contour minimum = 100.0
contour maximum = 200.0
contour increment = 2.0

This would be sufficient information to produce the contours

100.0, 102.0, 104.0... 198.0, 200.0.

However, 12d Model also provides a parameter to allow the following contours to be produced

101.0, 103.0, 105.0 .... 197.0, 199.0

For this and similar cases, the ***contour reference*** value is used.

When a contour reference value is specified, all the contour increments are taken from the contour reference value.

In the first example, the contour reference value is the default value of 0.0. In the second example, the contour reference value is 1.0.

If **major (**or **bold** or **index**) contours are required, the major contour increment is given a non-zero value. Major contours will be drawn at the integer multiples of the major increment instead of the standard contours.

For information on **labelling contours**, go to the section Labelling Contours.

For information on **smoothing contours**, go to the section Smoothing Contours.

Smoothing Contours

If the triangles in a tin, or sections of a tin, are large then the contours for that section of the tin may appear angular. Consequently some people like to **smooth** the contours to take out the angularity.

Be careful of smoothing because smoothing of contours produces strings that no longer sit on the triangulation.

12d Model smooths contours by apply a smoothing algorithm to each of the contour strings that were produced from the triangulation.

Two smoothing algorithms are available

(a) Preserve string points - this method ensures that the smoothed contour goes through all the vertices from the original contour string.

(b) Don’t preserve string points - this method does not have to include the vertices from the original contours in the smoothed contour.

For information on **creating contours**, go to the section Creating Contours.

For information on **labelling contours**, go to the section Labelling Contours.

Labelling Contours

12d Model labels contours by placing text labels at user controlled distances along the contour string.

The position and frequency of the contour labels along a contour string is controlled by two parameters, the *start distance* and the *separation distance*.

The first contour label is placed at a chainage distance of *start distance* from the beginning on the contour and the labels are repeated at a chainage distance of *separation distance* from the previous contour label.

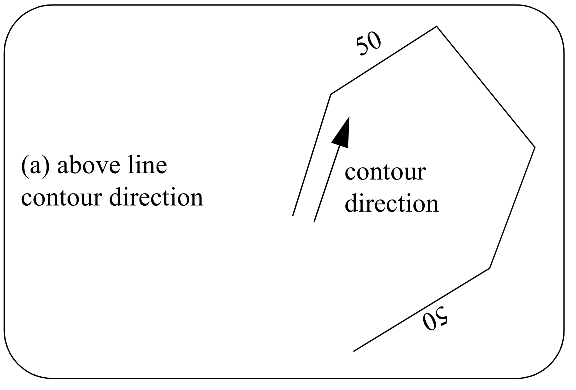
There is also a parameter to specify that contours are to be labelled at the start and the end regardless of the *start distance* and *separation distance*.

The number of decimal places used for the contour label is also user specified.

12d Model has seven methods for positioning the contour labels. They are the

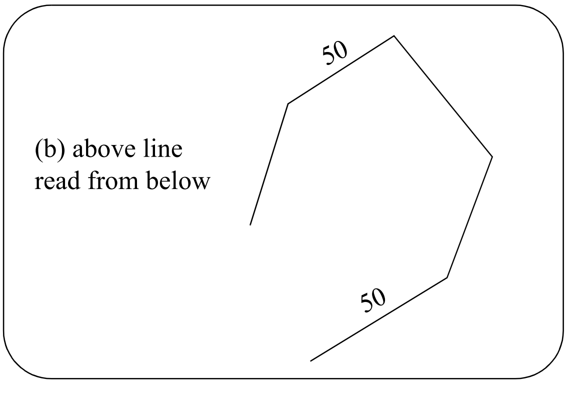
(a) above line contour direction

contour labels are created with the text parallel to the contours line and in the same direction as the contour string. The text is raised slightly above the contour string.



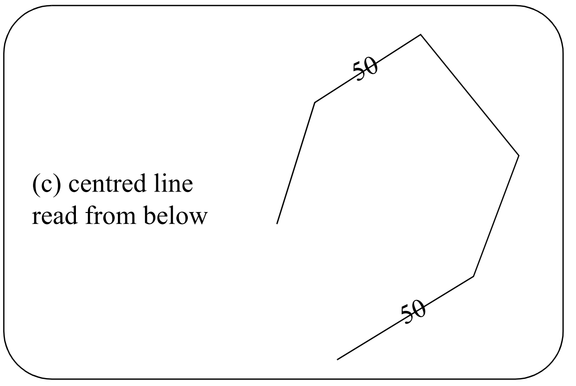
(b) above line read from below

contour labels are created with the text parallel to the contour line and in the same direction as the contour string but at an angle between -90 and +90 degrees. The text is raised slightly above the contour string.



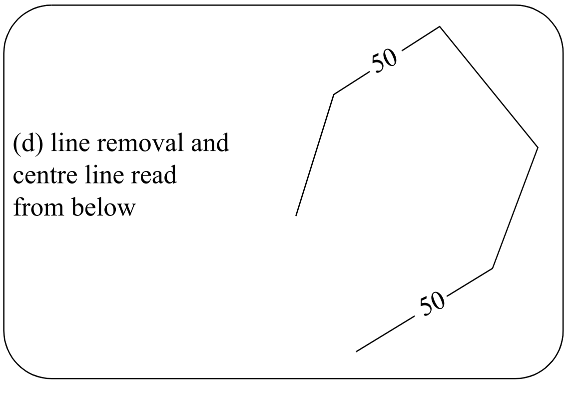
(c) centre line read from below

contour labels are created with the text parallel to the contour line and in the same direction as the contour string but at an angle between -90 and +90 degrees. The text is centred on the contour string so the contour string goes through the text.



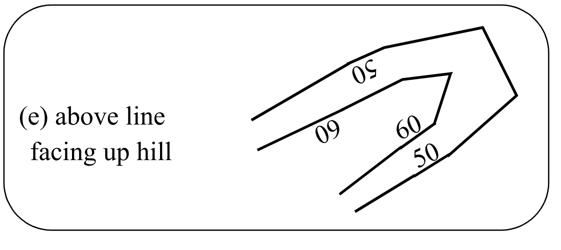
(d) line removal & centred line read from below

contour labels are created with the text parallel to the contour line and in the same direction as the contour string but at an angle between -90 and +90 degrees. The text is centred on the contour string so the contour string goes through the text. Also the contours are cut so that gaps are left for the labels.



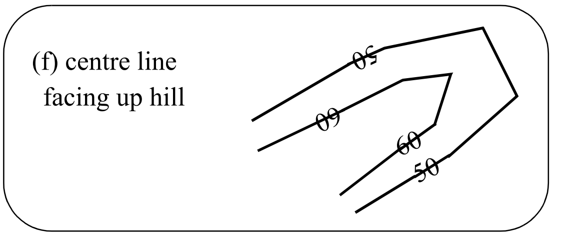
(e) above line facing up hill

contour labels are created with the text parallel to the contours line in the same direction as the contour string. The text is orientated so that if faces uphill. The text is raised slightly above the contour string.



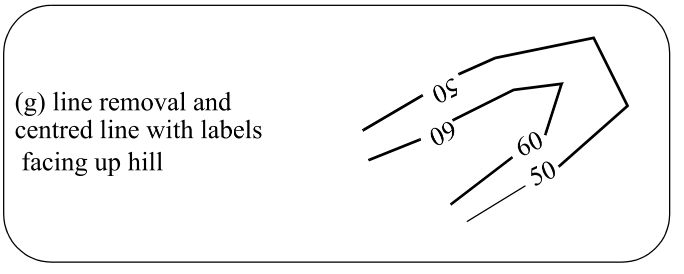
(f) centre line facing up hill

contour labels are created with the text parallel to the contours line in the same direction as the contour string. The text is orientated so that if faces uphill. The text is centred on the contour string so the contour string goes through the text.



(g) line removal & centred line facing up hill

contour labels are created with the text parallel to the contour line. The text is orientated so that if faces uphill. The text is centred on the contour string so the contour string goes through the text. Also the contours are cut so that gaps are left for the labels.



For information on **creating contours**, go to the section Creating Contours.

For information on **smoothing contours**, go to the section Smoothing Contours.

16.9.2 Contour, Smooth and Label

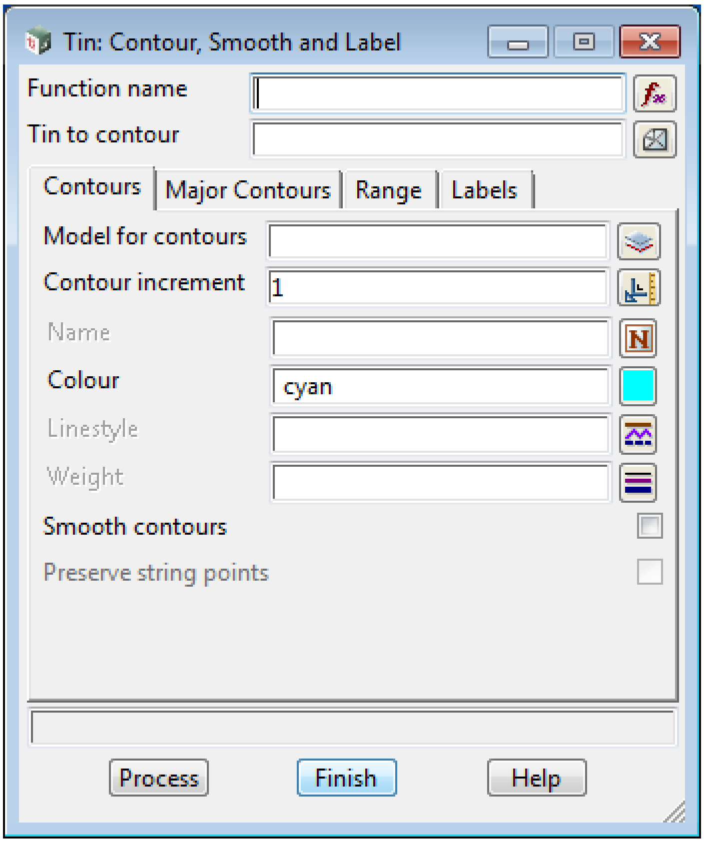
Position of option on menu: **Tins =>Contour =>Contour, smooth and label**

One method of visualizing tin data is with contours. In 12d Model, contour strings can be produced over any user defined elevation range and at any interval.

This option creates contours and can also smooth and label them. It is a function and so if a function is re-run it will remove any data from its last run.

For more information on contouring, smoothing and labelling in 12d Model, go to the section Creating Contours

Selecting **Contour, smooth and label** displays the **Contour, Smooth and Label** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function name	function box		available contour fns

name of the contour, smooth and label function.

Tin to contour	input		available tins

name of the tin to be contoured.

Contours tab

Model for contours	model box		available models

name of the model to place the contours in.

Contour increment	input	1.0	

increment between contoured values.

Name input

if non-blank, name to give the contour strings.

Colour input cyan available colours

colour of the contours

Linestyle input 1 available linestyles

linestyles for the contour strings.

Weight input

if non-blank, weight to give the contour strings.

Smooth contours tick box

if ticked, the contours are smoothed.

Preserve string points tick box

if ticked, then the smoothed string goes through the original vertices of the non smoothed contour strings.
If not ticked, then the smoothed strings do not have to include the vertices from the non-smoothed contours.

Major Contours tab

Create major contours tick box

if ticked, then major contours can be given a different name, colour, style, weight and model.

Model for major contours model box available models

if non blank, the name of the model to place the major contours in. If blank, the same model is used as for the ordinary contours.

Major contour increment input 5.0

increment for the major contours.
If blank or zero, then no bold contours are drawn. If non-zero, it must be an integer multiple of the contour increment.

Name input

if non-blank, name to give the major contour strings. If blank, the same name is used as for the ordinary contours.

Colour input off yellow available colours

colour of the major contours. If blank, the same colour is used as for the ordinary contours.

Linestyle input 1 available linestyles

if non blank, linestyles for the major contour strings. If blank, the same linestyle is used as for the ordinary contours.

Weight input

if non-blank, weight to give the major contour strings. If blank, the same weight is used as for the ordinary contours.

Range tab

Contour minimum input

minimum value of the z range to be contoured. If blank, the tin's minimum z-value is used.

Contour maximum input

maximum value of the z range to be contoured. If blank, the tin's maximum z-value is used.

Contour reference input 0

reference value for the contour increments.

Colour by range tick box

if ticked, then the contours are coloured according to a height range file rather than bold colour. See 8.9.6 Height Range File.

Height range file height range file box *.hrf

the range file used to colour the contours.

Interpolate colours tick box

*if **ticked**, the colour of the contour will be interpolated between the colour for the range and the colour for the next range in the height range file.*
*If **not ticked**, all contours in the same range in the height range file will have the same colour*

Colour text labels tick box

if ticked, then any contour height text is also coloured the same colour as the contour.

Labels tab

Label contours tick box

if ticked, then the contours are labelled.

Label major contours only tick box

if ticked, then only the major contours are labelled.

Model for labels model box available models

if non blank, the name of the model to place the contour labels in. If blank, the same model is used as for the contours.

Contour method choice box above line contour direction
above line read from below
centred line read from below
line removal and centred line read from below
above line facing uphill
centred line facing uphill
line removal & centred facing uphill

For information on each of the methods of labelling contours, go to the section Labelling Contours.

Decimal places

number of decimal places used in the contour label.

Textstyle data 1 textstyle favourites

textstyle information to use for the contour labels.

Start dist (w) input 0

chainage distance to the first label on the string - world units

Separation (w) input 250

chainage interval between the labels - world units

Model of label lines model box available models

if non blank then instead of using the **start distance** *and* **separation** *parameters to define where the contour labels are placed, the* **Model of label lines** *is used. In this case, a label is created where ever s string in the* **Model of label lines** *cuts a contour.*

Label start and end tick box tick

if ticked, the start and end of the contours are labelled regardless of the values of start distance and separation.

Process button

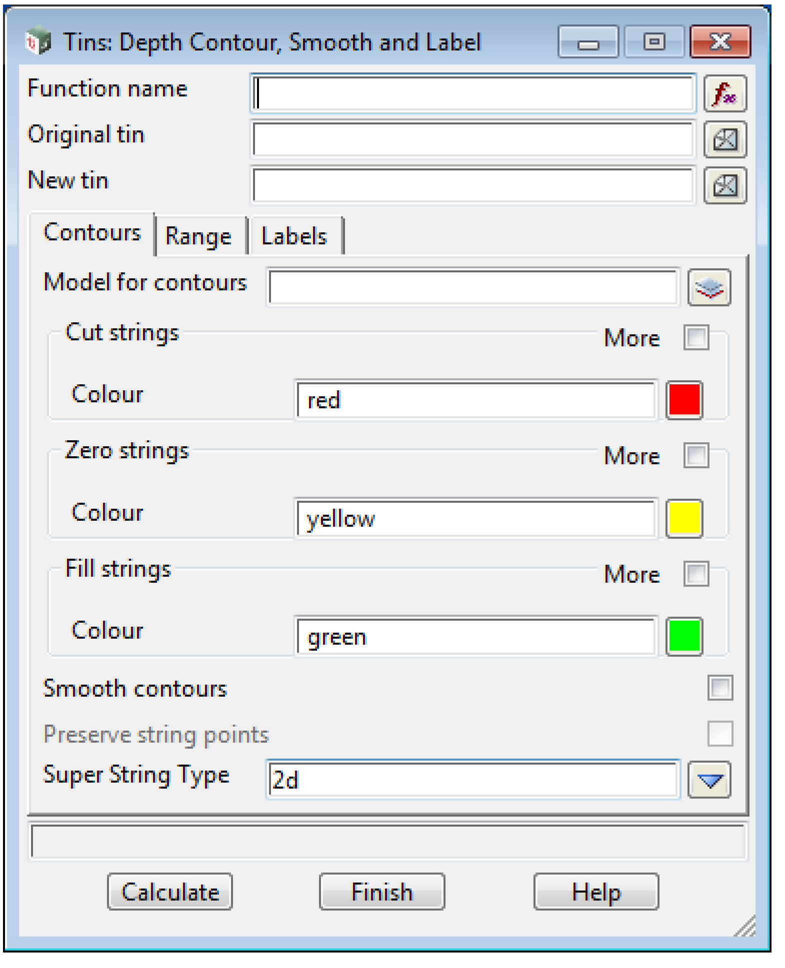
contour, smooth and label the tin given in the **tin** *field and place the contours in the model given in the model field.*

16.9.3 Depth, Contour and Label Function

This function allows the user to create depth contours and also label and smooth them.

Position of option on menu: Tins =>Contour =>Depth contour, smooth and label

Selecting **Depth contour, smooth and label** displays the **Tins: Depth Contour, Smooth and Label** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function name	function box		available depth contour fns

name of the depth contour, smooth and label function.

Original/New tin	tin box		available tins

name of the original/new tin to contour between. Cut is when the new tin is below the original tin.

Contours tab

Cut/Zero/Fill strings sections

More	tick box	not ticked	

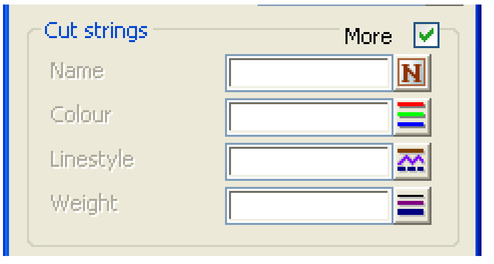
*if **not tick** then only the Colour panel field is displayed.*

*If **not ticked** then the extra fields **Name**, **Colour**, **Linestyle** are **Weight** are shown.*

Colour	colour box	cyan	available colours

*if **non-blank**, colour for the strings*

*If **More** is ticked then Name, Colour, Linestyle and Weight fields are displayed*



Name	name box		available names

*if **not blank**, name to give the cut/zero/fill strings.*

Colour	colour box	cyan	available colours

*if **not-blank**, colour for the cut/zero/fill strings*

Linestyle	linestyle box	1	available linestyles

*if **not blank**, linestyles for the cut/zero/fill strings.*

Weight weight box

*if **not blank**, weight to give the cut/zero/fill strings.*

Smooth contours tick box

*if **ticked**, the contours are smoothed.*

Preserve string points tick box

*if **ticked**, then the smoothed string goes through the original vertices of the non smoothed contour strings.*
*If **not ticked,** then the smoothed strings do not have to include the vertices from the non-smoothed contours.*

Super string type	choice box	2d	2d, 3d original, 3d new

*If **2d**, the created strings are 2d super strings with a z-value equal to the depth.*
*If **3d original**, the depth strings are draped over the original tin to form 3d super strings*
*If **3d new**, the depth strings are draped over the new tin to form 3d super strings.*

Range tab

Start level	real box	-10

the minimum depth to start the depth contours at.

End level	real box	10

the maximum depth to calculate the depth contours to.

Interval	real box	1

the interval between the depth contours.

Colour by range tick box

if ticked, then the contours are coloured according to a depth range file rather than cut and fill colours. See XX.

Depth range file	depth range file box		*.drf

the range file used to colour the depth contours.

Interpolate colours tick box

*if **ticked**, the colour of the contour will be interpolated between the colour for the range and the colour for the next range in the height range file.*
*If **not ticked**, all contours in the same range in the height range file will have the same colour*

Colour text labels tick box

if ticked, then any contour value text is also coloured the same colour as the contour.

Labels tab

Label contours tick box

if ticked, then the depth contours are labelled.

The text is given the z value of XX

Model for labels model box available models

if non blank, the name of the model to place the contour labels in. If blank, the same model is used as for the contours.

Label method choice box above line contour direction
above line read from below
centred line read from below
line removal and centred line read from below

For information on each of the methods of labelling contours, go to the section XX.

Decimal places

number of decimal places used in the contour label.

Textstyle data 1 textstyle favourites

textstyle information to use for the contour labels.

Start dist (w) real box 0

chainage distance to the first label on the string - world units

Separation (w) real box 250

chainage interval between the labels - world units

Model of label lines model box available models

if non blank then instead of using the **start distance** *and* **separation** *parameters to define where the contour labels are placed, the* **Model of label lines** *is used. In this case, a label is created where ever s string in the* **Model of label lines** *cuts a contour.*

Label start and end tick box tick

if ticked, the start and end of the contours are labelled regardless of the values of start distance and separation.

Button at bottom

Calculate button

generate, smooth and label the depth contours between the two tins

16.9.4 Change Contours

Position of option on menu: **Tins =>Contour =>Change**

The **Change** option is for changing the model and/or colour of contours with a given height interval.

The main purpose of the option is to separate contours to be used as index contours from other contours. For example, when contours are supplied by a third party and all the contours are in the same model with the same colour, **change contour colour** can move contours to be used as index contours to a different model and give them a different colour.

For this option the selection process for a Data source is extended.

After the data is selected as defined by the Data source, the two extra parameters, **interval** and **reference**, are used to further specify which contours are to be moved and/or coloured.

Only contour strings whose heights satisfy

height = reference + integer * interval

are selected.

For example, if the contours 5, 10, 15 etc. are required, the reference and interval would be

reference = 0

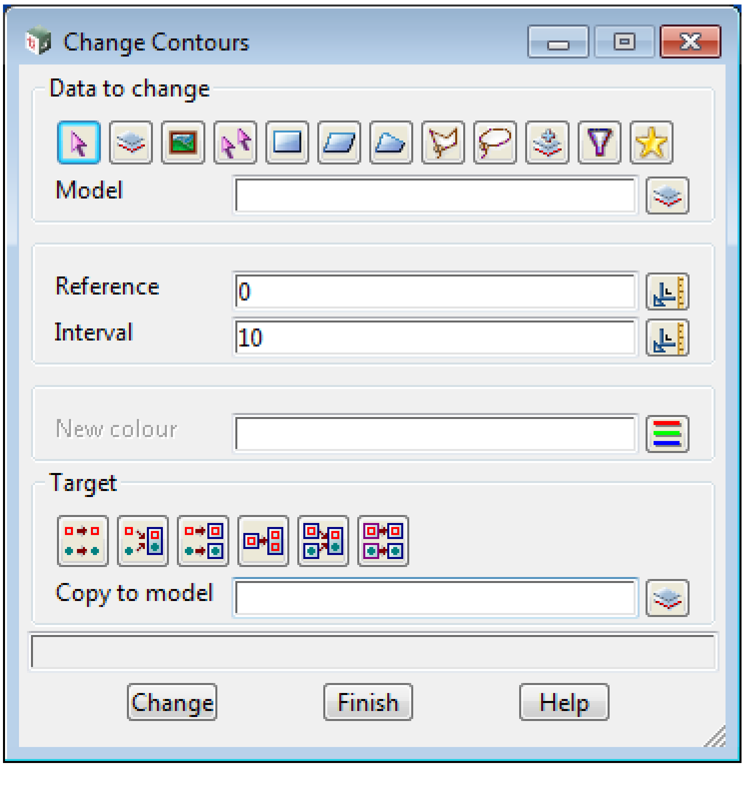
interval = 5

However, to modify the contours 6, 11, 16 etc., the required parameters are

reference = 1

interval = 5

Selecting **Change** displays the **Change Contours** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Reference	input	0	

the reference value to use to select contours.

Interval	input	10	

the interval to use to select contours.

New colour	input		available colours

if non-blank, then the colour of the selected strings will be changed to the colour given in the **New colour** *field.*

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target.

Target info	input		

extra information required for the target.

Change	button		

process the selected strings

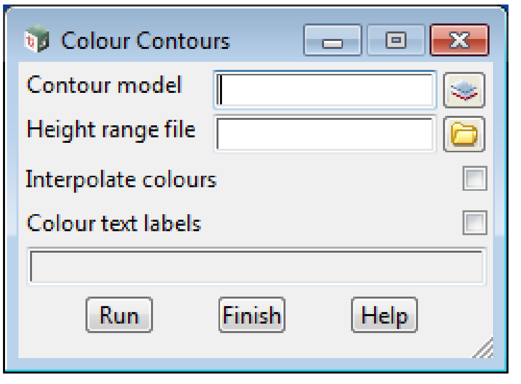
16.9.5 Colour

Position of option on menu: **Tins =>Contour =>Colour**

The **Colour** option is used to colour contours in a model according to a selected height range file.

For more information on the height range file, see 8.9.6 Height Range File.

On selecting **Colour**, the **Colour Contours** panel is displayed.



The fields and buttons in this panel are used as follows

Field Description	Type	Defaults	Pop-Up
Contour model	model box		available models

model of contours to colour

Height range file	height range file box		.hrf files

height range file to specify the colour of the contour. See 8.9.6 Height Range File.

Interpolate colours	tick box		

*if **ticked**, the colour of the contour will be interpolated between the colour for the range and the colour for the next range in the height range file.*
*If **not ticked**, all contours in the same range in the height range file will have the same colour*

Colour text labels	tick box	tick	

*if **ticked**, any text labels in the model will also be coloured.*

Run	button		

colour the contours and text according to the height range file.

16.9.6 Contour

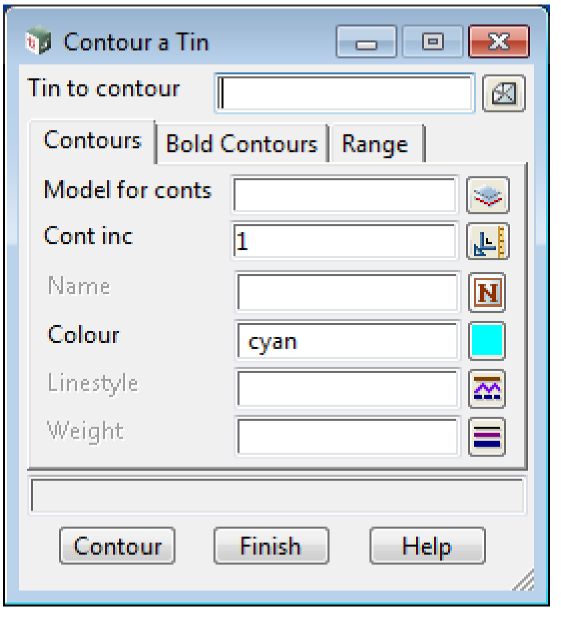
Position of option on menu: **Tins =>Contour =>Contours**

One method of visualizing tin data is with contours. In 12d Model, contour strings can be produced over any user defined elevation range and at any interval.

For more information on contouring in 12d Model, go to the section Creating Contours

The **contour** option can be applied to a tin any number of times. This may be necessary if the contour parameters vary over the total range to be contoured.

On selecting the **contour** option, the **contour a tin** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin to contour	tin box		available tins

name of the tin to be contoured.

Contours tab

Field Description	Type	Defaults	Pop-Up
Model for conts	model box		available models

name of the model to place the contours in.

Cont inc	input	1.0	

increment between contoured values.

Name	input		

if non-blank, name to give the contour strings.

Colour	colour box	cyan	available colours

colour of the contours

Linestyle	linestyle box	1	available linestyles

linestyles for the contour strings.

Weight	weight box		

if non-blank, weight to give the contour strings.

Bold Contours tab

Model for bolds model box available models

name of the model to place the bold contours in. If blank, the same model is used as for the ordinary contours.

Bold inc input 5.0

increment for the bold contours.
If blank or zero, then no bold contours are drawn. If non-zero, it must be an integer multiple of the contour increment

Name input

if non-blank, name to give the bold contour strings.

Colour colour box cyan available colours

colour of the bold contours

Linestyle input 1 available linestyles

linestyles for the bold contour strings.

Weight input

if non-blank, weight to give the bold contour strings.

Range tab

Cont min input

minimum value of the z range to be contoured. If blank, the tin's minimum z-value is used.

Cont max input

maximum value of the z range to be contoured. If blank, the tin's maximum z-value is used.

Cont ref input 0.0

reference value for the contour increments.

Contour button

contour the tin given in the **tin** *field and place the contours in the model given in the model field. The contour range, contour increment and bold contour increment are used specify which contours are calculated.*

Panel Messages

Progress messages - sent to the panel message area

start contouring
contouring level *z-value*

Completion message - sent to the panel message area

finished contouring - *no.* contours

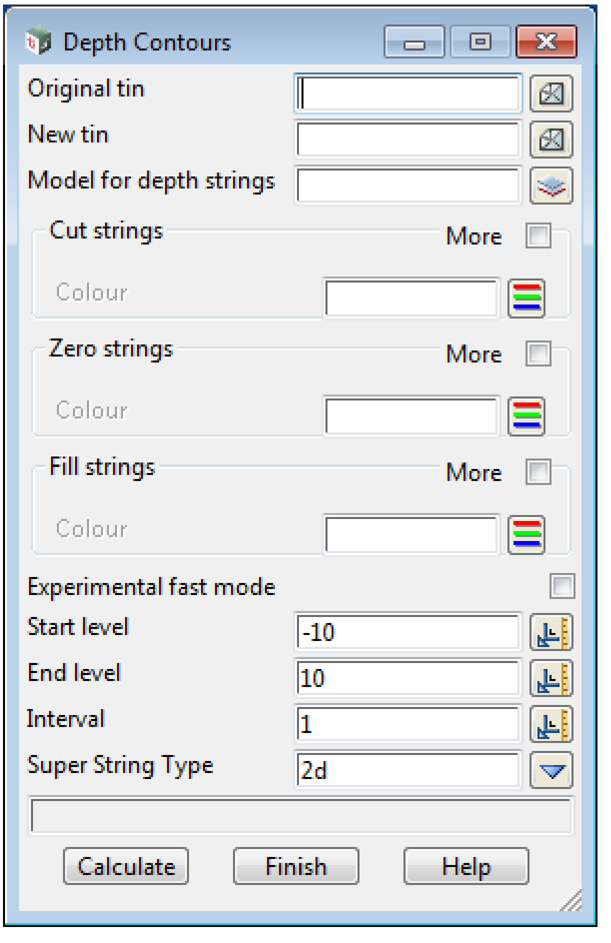
<esc> can be used to abort the calculation of contours.

16.9.7 Depth Contours

Position of option on menu: **Tins =>Contour =>Depth contours**

The **depth contours** option finds the contours lines for the difference between two tins. That is, it calculates the isopachs between the two tins.

Selecting **Depth contours** displays the **Depth Contours** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Original/New tin	tin box		available tins

name of the original/new tin to contour between. Cut is when the new tin is below the original tin.

Model for depth strings	model box		available models

name of the model to contain the depth strings.

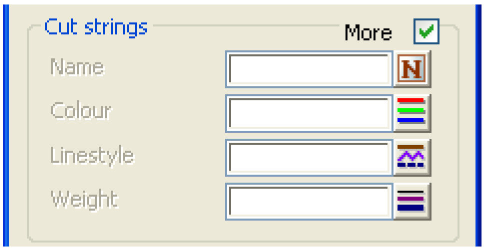
Cut/Zero/Fill strings section **More** tick box

if no tick then only the Colour panel field is displayed

Colour	colour box	cyan	available colours

if non-blank, colour for the strings

if tick then Name, Colour, Linestyle and Weight fields are displayed



Name input

if non-blank, name to give the strings.

Colour colour box cyan available colours

if non-blank, colour for the strings

Linestyle linestyle box 1 available linestyles

linestyles for the strings.

Weight weight box

if non-blank, weight to give the strings.

Start level input -10

the minimum depth to start the depth contours at.

End level input 10

the maximum depth to calculate the depth contours to.

Interval input 1

the interval between the depth contours.

Super string type choice box 2d 2d, 3d original, 3d new

*If **2d**, the created strings are 2d strings with a z-value equal to the depth.*
*If **3d original**, the depth strings are draped over the original tin to form 3d strings.*
*If **3d new**, the depth strings are draped over the new tin to form 3d strings.*

Calculate button

calculate the depth contours between two tins given in the panel fields. The resulting strings are added to the model given in the **Model for depth strings** *field.*

16.9.8 Label Contours

Position of option on menu: **Tins =>Contour =>Label**

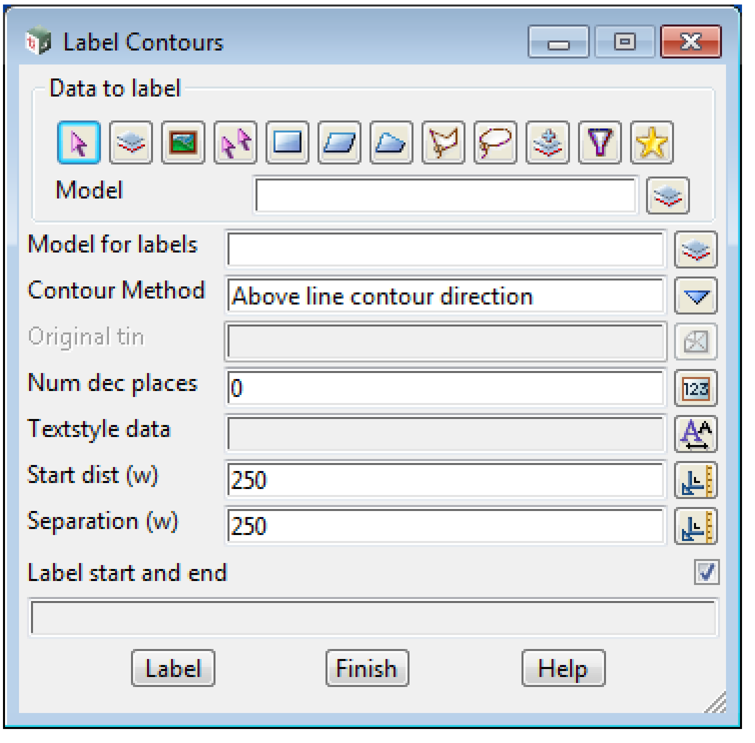
The **label contours** option can be used to label individual contour (2d) strings, or all the contour strings in a model.

The user has control over the label size, colour, number of decimal places, the distance between the labels and the distance to the first label on a string (start distance).

Note - this option cannot label fast contours. Contours strings must already exist or be generated using **Tins => Contour=> Contour, Smooth and Label** or **Tins => Contour=> Contour**

For more information on labelling contours, go to the section Labelling Contours.

On selecting the **label contours** option, the **label contours** panel is displayed.



When the **Label** button is selected, all the contour strings given by the data source field will be labelled according to the parameters in the panel fields.

The fields and buttons used in the **label contours** panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Field Description	Type	Defaults	Pop-Up
Data source	input		

source of data is to be triangulated - only Model or View allowed for this option.

Field Description	Type	Defaults	Pop-Up
Model for labels	model box		available models

name of the model to place the contour labels in.

Field Description	Type	Defaults	Pop-Up
Label method	choice box		above line contour direction above line read from below centred line read from below line removal and centred line read from below

above line facing uphill
centred line facing uphill
line removal & centred facing uphill

in all cases, contour labels are created and added to the model Model for contours.

For information on each of the methods of labelling contours, go to the section Labelling Contours.

Original tin	tin box		available tins

only needed for the "facing uphill" methods - name of the tin used to generate the contours. This is needed to calculate which direction is uphill.

Num dec places	input	1	0,1,2,3,4,5

number of decimal places used in the contour label.

Textstyle data	textstyle data box	1	textstyle favourites

textstyle information to use for the contour labels.

Start dist (w)	input	0

chainage distance to the first label on the string - world units

Separation (w)	input	250

chainage interval between the labels - world units

Pick button

if a number of individual contour strings need to be labelled, the pick button is used. After **pick** *is chosen, any strings selected are then labelled using the information in the panel fields. After labelling one string, another pick and label cycle automatically begins.*

The cycle is terminated by clicking RB to raise the **pick ops** *menu and selecting cancel from it.*

Label button

label all the contour strings selected by the "Data to label" data source.

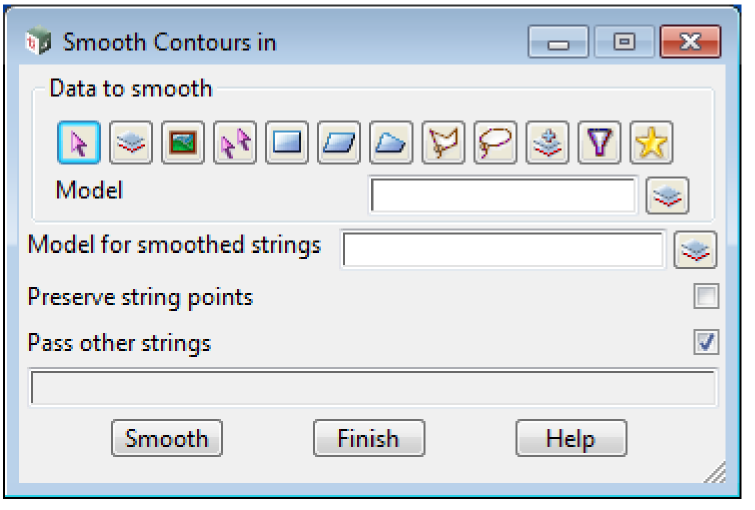
16.9.9 Smooth

Position of option on menu: **Tins =>Contour =>Smooth**

The **smooth** options are used to add extra points into a contour string (2d strings) to make a smoother string.

For more information on labelling contours, go to the section Smoothing Contours.

Selecting **Smooth** displays the **Smooth Contours in** panel.



The fields and buttons in this panel are used as follows

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

data source to be smoothed.

Model for smoothed strings	model box		available models

model for the smoothed strings to go to

Preserve string points	tick box		

*if **ticked**, the smoothed contour will still contain all the original points.*
*if **not ticked**, the smoothed contour may deviate from the original string points*

Pass other strings	tick box	tick	

*if **ticked**, any non-contour string is copied and added to the **smoothed strings** model.*

Smooth	button		

smooth the selected contour (2d) strings.

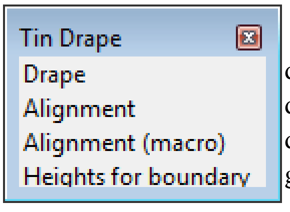
16.10 Drape

Position of menu: **Tins =>Drape**

The **drape** option sections along a string and creates a new 3d-string which sits over the original string in plan view (with line approximations for arcs and spirals) but with z-values taken from a triangulated surface (a tin).

The drape alignment option takes an alignment string and creates a new alignment string with the identical horizontal geometry and vertical intersection points with z-values taken from a tin.

The **tin drape** walk-right menu is



drape strings and produce 3d strings
drape an alignment string and produce an alignment string
drape alignment from V7V1f
give tin heights to a tin boundary

The options in the tin contours menu will now be discussed.

For the option *Drape*, go to 16.10.1 Drape
Alignment 16.10.2 Drape Alignment
Alignment (macro) 16.10.3 Drape Alignment (Macro)
Heights for boundary 16.10.4 Heights for Tin Boundary or Drape Using Closest Tin Node

16.10.1 Drape

Position of option on menu: **Tins =>Drape =>Drape**

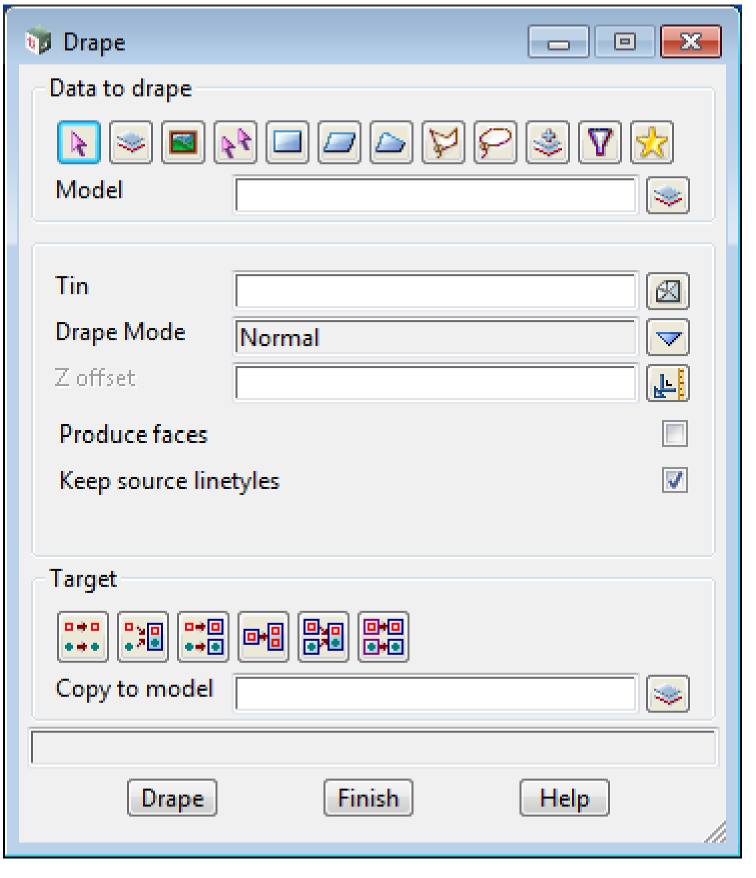
The **drape** operation is simply sectioning along a string.

Draping is a very useful method for attaching a z-value for strings defined in plan only. For example, cadastral information often has only (x,y) co-ordinates. The drape operation could be used to apply z-values to the cadastral information, for example, z-values from the natural surface tin. The draped cadastral information would then sit on the natural surface and could be used effectively in perspective views as well as plan views to delineate properties etc.

The draped strings can also have a z-offset applied so that the draped strings are a constant distance above or below the tin.

Instead of simply producing a section along a string, **drape** can also be used to produce a face for every segment of the string by draping the segment onto the tin and forming a face by perpendicularly connecting the end points of the segment and the draped segment. For example, if the strings represented the tops of buildings, the created faces would be the sides of the buildings down to the terrain.

On selecting **Drape**, the **Drape** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

data to drape.

Tin tin box available tins

name of the tin to drape the strings over.

Drape mode choice box Normal Normal, Vertices only

*if **Normal**, the section through the tin is created.*
*If **Vertices only**, only the z-value from the tin is calculated at the vertices of the strings.*

Z offset real value box

if non-blank, the z-value of all the draped strings is increased by this value.

Produce faces tick-box

*if **ticked**, drape each segment of the string to form faces.*
*If **not ticked**, simply drape the string onto the tin.*

Keep source linestyles tick box

*if **ticked**, the linestyles of the draped strings are the same as the original string that was draped.*

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target.

Target info input

models where the draped strings are placed.

Drape button

drape the strings selected by the "data to drape" data source over the tin given in the tin field.

Note: if faces rather than sections are to be produced, tick on the **produce faces**.

<esc> can be used to abort the drape option.

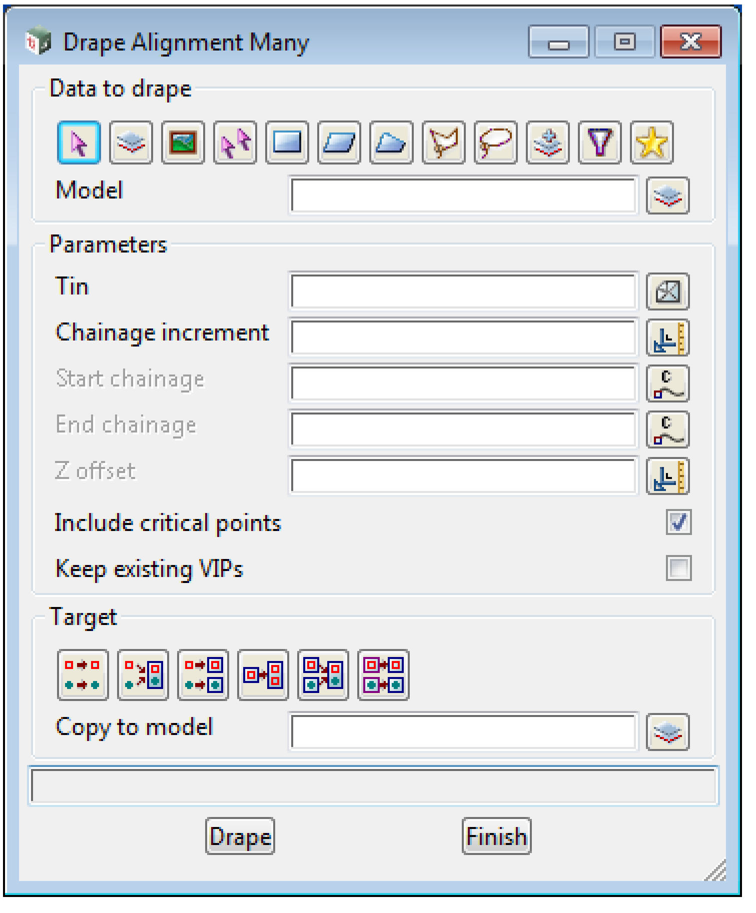
16.10.2 Drape Alignment

Position of option on menu: **Tins =>Drape =>Alignment**

The **Drape alignment** operation takes an alignment/super alignment string and creates a new alignment/super alignment string with identical horizontal geometry and the z-value for vertical intersection points taken from a tin. The z-values are taken over a user specified chainage interval and chainage increment. Vertical intersection points outside the chainage range can be kept.

The draped alignments can also have a z-offset applied so that the draped alignments are a constant distance above or below the tin.

On selecting **Alignment**, the **Drape Alignment Many** panel is displayed.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up

Data source type Model

data selection type - for a full description go to 4.19.3 Data Source.

Data source input

source of data of alignment/super alignment strings to drape.

Tin tin box

triangulation to take z-values from.

Chainage increment input

the chainage interval to create VIP points for. Each created VIP point is given the z-value from the tin and the (x,y) position of the VIP.

Start/End chainage input

*if **non blank**, the chainage range to create VIP's over.*

*If **blank**, use the start/end chainage of the selected string.*

Z offset input

*if **non blank**, the value is added to the z-values from the tin.*

Include critical points tick box tick

if ticked, VIP's are created at the horizontal tangent points.

Keep existing VIPs tick box

if ticked, any existing VIP's outside the given chainage range are kept.

If not ticked, don't create any VIP points outside the chainage range.

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info input

extra information required for the target.

Drape button

create new alignment/super alignment strings with the same horizontal geometry as the selected strings and the z-value for VIP points taken from the given tin.

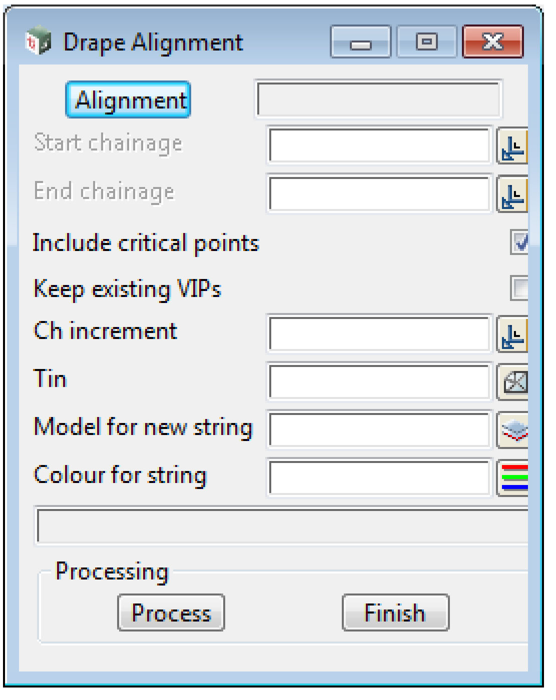
16.10.3 Drape Alignment (Macro)

Position of option on menu: **Tins =>Drape =>Alignment (macro)**

This is the **Drape =>Alignment** option from V7C1f. It has now been superseded.

The **Drape alignment (macro)** operation takes an alignment string and creates a new alignment string with identical horizontal geometry and the z-value for vertical intersection points taken from a tin. The z-values are taken over a user specified chainage interval and chainage increment. Vertical intersection points outside the chainage range can be kept.

On selecting **Alignment (macro)**, the **drape alignment** panel is displayed.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up

Alignment string select

alignment string to drape.

Start/End chainage input

the chainage range to create VIP's over. if blank, use the start/end chainage of the selected string.

Include critical points tick box tick

if ticked, VIP's are created at the horizontal tangent points.

Keep existing VIPs tick box

if ticked, any existing VIP's outside the given chainage range are kept.

If not ticked, don't create any VIP points outside the chainage range.

Ch increment input

the chainage interval to create VIP points for. Each created VIP point is given the z-value from the tin and the (x,y) position of the VIP.

Tin tin box

triangulation to take z-values from.

Model for new string model box

model for the draped string.

Colour for string colour box

Colour for the draped string.

Process button

create a new alignment string with the same horizontal geometry as the selected string and the z-value for VIP points taken from the given tin.

16.10.4 Heights for Tin Boundary or Drape Using Closest Tin Node

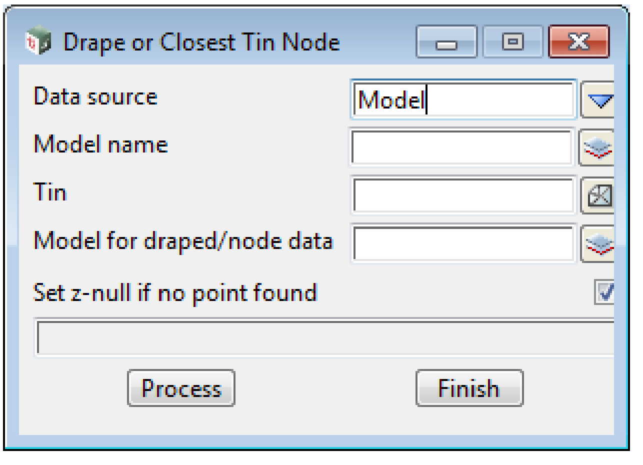
Position of option on menu: **Tins =>Drape =>Heights for boundary**

This option takes strings and for each vertex of each string, replaces the z-value by the z-value at the same (x,y) position from a given tin, or if there is no tin at the (x,y) position, takes the z-value from the closest non-null tin node to the (x,y) position.

The option is useful for taking a tin boundary with no z-values and creating z-values for each vertex from the tin.

Since V8 the option **Tins =>Boundary** option has a *Boundaries draped* flag which performs the same calculations as this option.

On selecting **Heights from boundary,** the **Update Z-values from Tin** panel is displayed.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Data source	choice box	model	model, view, string

type of the data source

Model/View/String name

source of the data to drape vertices for.

Tin tin box

tin to drape the vertices onto.

Model for draped/node data model box

model for the draped strings.

Set z-null if no point found tick box tick

if ticked, the vertex z-value is set to null if no tin exists at the vertex.

If not ticked and the tin does not exist at the vertex, the z-value for the vertex is taken from the closest non-null tin node.

Process button

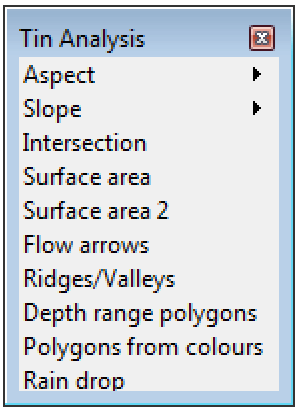
run the option.

16.11 Tin Analysis

Position of menu: **Tins =>Tin analysis**

The **Tin analysis** options are used to calculate the direction and slope of triangles, the intersection between tins, surface areas, flow arrows, ridge and valley lines and viewsheds.

The **Tin analysis** walk-right menu is



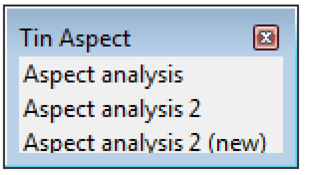
colour and report for ranges of aspects
colour and report for ranges of slopes
lines of intersection between two tins
report surface area within a polygon
report surface area for a model of polygons
create flow arrows for a tin.
create ridge and valley lines for a tin
create polygons for depth ranges
creates polygons around coloured regions of tin
creates flow lines from a selected point

For the options *Aspect*, go to	16.11.1 Aspect
Slope	16.11.2 Slope
Intersection	16.11.3 Intersection
Surface area	16.11.4 Surface Area
Surface area 2	16.11.5 Surface Area 2
Flow arrows	16.11.6 Flow Arrows
Ridge/Valleys	16.11.7 Ridge/Valleys
Depth range polygons	16.11.8 Depth Range Polygons
Polygons from colours	16.11.9 Polygons from Tin Colours
Rain drop	29.9.2 Raindrop/Teardrop

16.11.1 Aspect

Position of menu: **Tins =>Tin analysis =>Aspect**

The **Tin aspect** options calculate and report on the directions of the triangles. The **Tin aspect** walk-right menu is



The options in this menu will now be discussed.

For the option Aspect analysis, go to 16.11.1.1 Aspect Analysis
Aspect analysis 2 16.11.1.2 Aspect Analysis 2 and 2 (new)
Aspect analysis 2 (new) 16.11.1.2 Aspect Analysis 2 and 2 (new)

16.11.1.1 Aspect Analysis

Position of option on menu: **Tins =>Tin analysis =>Aspect => Aspect Analysis**

This panel is used to calculate the aspect or direction that the triangles of a tin face. The calculations can be restricted to within a user selected polygon, or if no polygon is selected, the entire tin is used.

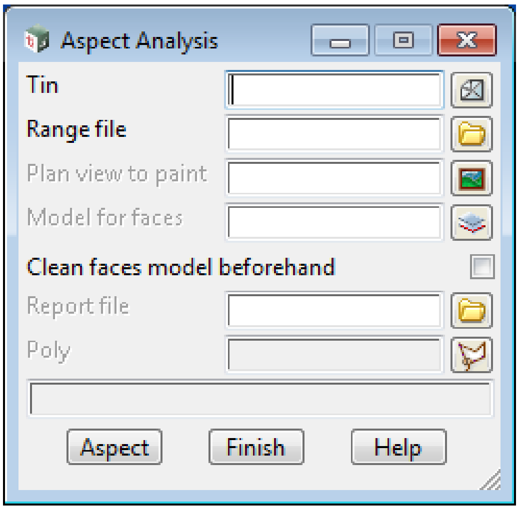
The calculated aspects can be produced and reported over user supplied bearing ranges. Similarly, a plan view can be temporarily coloured on an aspect basis using the same **range** file. The temporary colours will disappear next time the view is refreshed.

For permanency, the option can also create coloured faces with the appropriate range colour. The faces can be displayed on any plan view using a solid fill colour or a hatch pattern.

For more information on the aspect range file, see 8.9.5 Aspect Range File.

For each range in the file, the plan and slope areas of the triangles in the bearing range will be reported on, and the range_colour can be used to colour all areas on a plan view satisfying the range.

On selecting the **aspect analysis** option, the **Aspect Analysis** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Tin tin box available tins

name of the tin for which the aspects of the triangles will be calculated.

Range file aspect range file box *.arf

the user supplied range file is used to split up the aspect report and define the range colours used for painting a plan view. See 8.9.5 Aspect Range File.

Plan view to paint view box available views

if non-blank, the given plan view will be painted according to the range colours given in the range file.

Model for faces model box available models

if non-blank, faces will be created with colours according to the range colours given in the range file.

The z-value for a face is the line number in the range file of the range that the face satisfies.

Clean faces model beforehand tick box

if ticked, the model of faces is cleaned out before the option runs.

Report file file box *.rpt

name of the file to contain the aspect report. If the file already exists, the report will be appended to the file. If no name is given, no report is produced.

Poly string-select

if selected, this string is used as the bounding polygon for the aspect calculations.

Aspect button

the aspects of the triangles of the tin within the selected bounding polygon are calculated.

<esc> can be used to terminate the option during aspect calculations.

16.11.1.1.1 Example of an Aspect Range File

```
//   aspect range file
// format:   lower_bearing     upper_bearing     colour_for_range

0   45   red   // colour red the triangles with bearing greater than 0 and less than or equal to 45
45  90   green   // colour green the triangles with bearing > 45, and <= 90

90  123.30  "dark green"  // colour dark green triangles with bearing > 90 and <=l to 123 deg 30'
```

16.11.1.2 Aspect Analysis 2 and 2 (new)

Position of option on menu: **Tins =>Tin analysis =>Aspect => Aspect Analysis 2**

Position of option on menu: **Tins =>Tin analysis =>Aspect => Aspect Analysis 2 (new)**

As with the **aspect analysis** option, the **aspect analysis 2** option is used to calculate the aspect or direction that the triangles in the tin face.

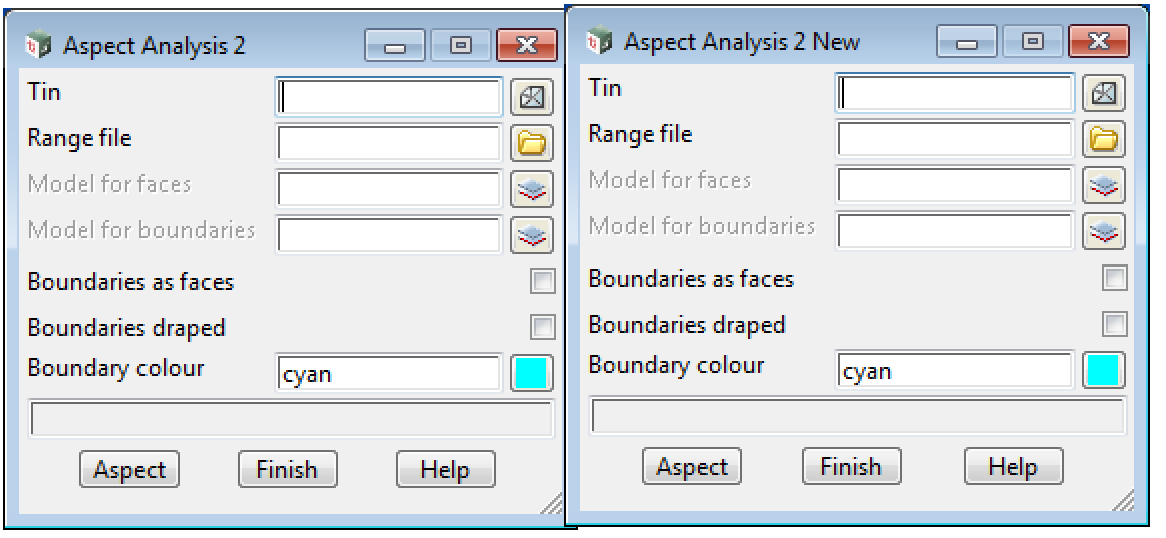
However, this option can also create the strings that bound the triangles in the same aspect range. The **aspect analysis 2 (new)** option is the same as aspect analysis 2 except the boundary polygons have slightly different names.

One restriction for this option is that the calculations can not be restricted to be within a polygon and the entire tin is used.

For more information on the aspect range file, see 8.9.5 Aspect Range File.

For each range in the file, the boundary strings for areas of common range can be created and also coloured faces that can be displayed on any plan view.

On selecting the **Aspect analysis 2** option, the **Aspect Analysis 2** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin for which the aspects of the triangles will be calculated.

Range file	aspect range file box		*.arf

the user supplied range file is used to split the triangles into groups of common aspects so that boundary strings can be created and the range colours used for faces and the boundary strings. See 8.9.5 Aspect Range File.

Model for faces	model box		available models

if non-blank, a faces will be created (with the colour given in the range file) for each triangle.

The z-value for a face is the line number in the range file of the range that the face satisfies.

Model for boundaries	model box		available models

if non-blank, for each range of aspects in the range file, boundary strings will be created for regions of common aspect. The boundary strings are given the name "aspect boundaries" plus the number of the entry in the range file. Many of the boundaries will exist twice since they are the upper boundary for one range and the lower boundary for the next range but will have a different colour and name. For **aspect analysis 2 (new)**, *the boundary name also has the addition of "face" and a number to more clearly differentiate the boundaries of the same range.*

Boundaries as faces	tick box		

if ticked, output the regions of common aspect as faces.

Boundaries draped	tick box		

if ticked, the z-values for the boundary are taken from the tin.

If not ticked, the z-value for a boundary is the line number in the range file of the range that the boundary satisfies.

Boundary colour	colour box	cyan	available colours

not used - was the colour for the aspect boundaries.

Aspect	button		

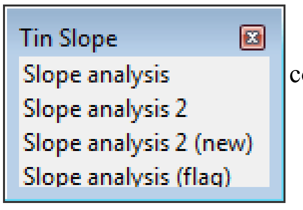
On selecting this button, the aspects of the triangles of the tin are calculated and the appropriate boundaries and faces created.

<esc> can be used to terminate the option during aspect calculations.

16.11.2 Slope

Position of menu: **Tins =>Tin analysis =>Slope**

The **tin slope** options calculate and report on the slopes of the triangles. The **tin slope** walk-right menu is



colour and report for ranges of slopes

The options in this menu will now be discussed.

For the option Slope analysis, go to	16.11.2.1 Slope Analysis
Slope analysis 2	16.11.2.2 Slope Analysis 2 and 2 (new)
Slope analysis 2 (new)	16.11.2.2 Slope Analysis 2 and 2 (new)
Slope analysis (flag)	16.11.2.3 Slope Analysis - Flag Slopes

16.11.2.1 Slope Analysis

Position of option on menu: **Tins =>Tin analysis =>Slope =>Slope analysis**

This panel is used to calculate the slopes of triangles in the tin. The calculations can be restricted to within a user selected polygon, or if no polygon is selected, the entire tin is used.

The calculated slopes can be produced and reported over user supplied slope ranges. Similarly, a plan view can be temporarily coloured on a slope basis using the same **range** file. The temporary colours will disappear next time the view is refreshed.

For permanency, the option can also create coloured faces with the appropriate range colour. The faces can be displayed on any plan view using a solid fill colour or a hatch pattern.

For more information on the slope range file, see 4.24 Defaults.

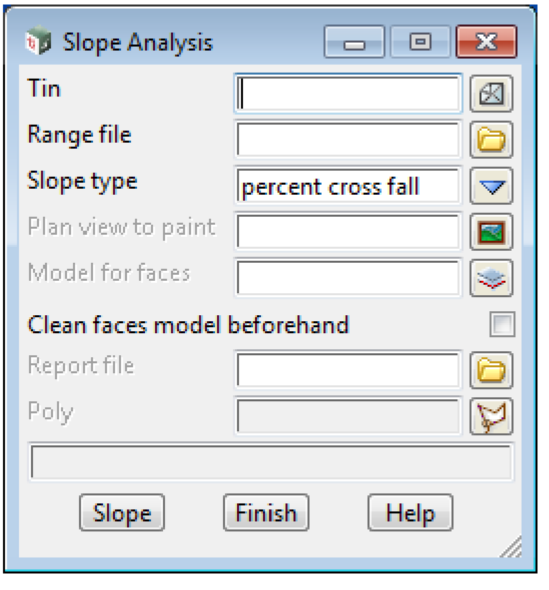
The slopes in the file can be percent cross fall, “1v in” slope or degrees (in 12d Model dms format) but they must all be of the same type in the file.

Note - if ***1v in*** is used, the range is converted to:

smaller_slope_as_percent_cross_fall < slope <= larger_slope_as_percent_crossfall

For each range in the file, the plan and slope areas of the triangles in the range will be reported on, and the range_colour used to colour all areas on a plan view satisfying the range.

On selecting the **Slope analysis** option, the **Slope Analysis** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin for which the slopes of the triangles will be calculated.

Field Description	Type	Defaults	Pop-Up
Range file	slope range file		*.srf

the user supplied range file is used to split up the slope report and define the range colours used for painting a view. See 4.24 Defaults.

Field Description	Type	Defaults	Pop-Up
Slope type	choice box		percent cross fall, degrees, 1v in

the units used for slope in the range file.

Field Description	Type	Defaults	Pop-Up
Plan view to paint	view box		available views

if non-blank, the given plan view will be painted according to the range colours given in the range file.

Clean faces model beforehand tick box

if ticked, the model of faces is cleaned out before the option runs.

Model for faces model box available models

if non-blank, faces will be created with colours according to the range colours given in the range file.

The z-value for a face is the line number in the range file of the range that the face satisfies.

Report file file box *.rpt

name of the file to contain the slope report. If the file already exists, the report will be appended to the file. If no name is given, no report is produced.

Poly string-select

if selected, this string is used as the bounding polygon for the slope calculations.

Slope button

On selecting this button, the slopes of the triangles of the tin within the selected bounding polygon are calculated.

<esc> can be used to terminate the option during slope calculations.

16.11.2.1.1 Example of a Slope Range File

```
//   slope range file in percent cross fall
// format:    lower_slope     upper_slope     colour_for_range

0    5      red      // colour red the triangles with slope greater than 0% and less than
                     // or equal to 5% x-fall
5   10     green    // colour green the triangles with slope > 5%,  and <= 10% x-fall
10   300  "dark green"   // colour dark green triangles with slope greater than 10%
                         // and less than or equal to 300% x-fall
```

// slope range file in ***1v in*** slopes

```
// format:    lower_slope     upper_slope     colour_for_range

  0    20      red      // colour red the triangles with slope greater than flat and
                        // less than or equal to 1:20
  20   10     green    // colour green the triangles with slope between 1:20 and equal to 1:10
  10   1     blue    // colour blue triangles with slope between 1:10 and equal to 1:1
```

16.11.2.2 Slope Analysis 2 and 2 (new)

Position of option on menu: **Tins =>Tin analysis =>Slope =>Slope analysis 2**

Position of option on menu: **Tins =>Tin analysis =>Slope =>Slope analysis 2 (new)**

As with the **Slope analysis** option, the **slope analysis 2** option is used to calculate the slope of triangles in the tin. However, this option can also create the strings that bound the triangles in the same slope range.

The **Slope analysis 2 (new)** option is the same as *Slope analysis 2* except the boundary polygons have slightly different names.

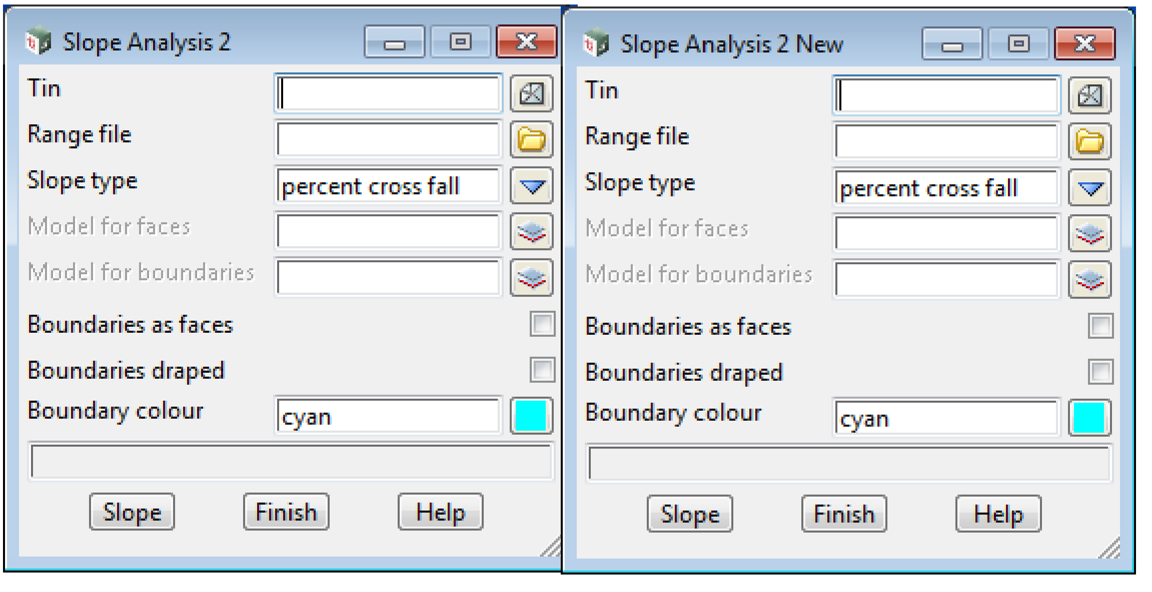
One restriction for this option is that the calculations can not be restricted to be within a polygon and the entire tin is used.

The range file for **Slope analysis 2** has the same format as for **Slope analysis.**

For more information on the slope range file, see 4.24 Defaults.

For each range in the file, the boundary strings for areas of common range can be created and also coloured faces that can be displayed on any plan view.

On selecting the **Slope analysis 2** option, the **Slope Analysis 2 new** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	input		available tins

name of the tin for which the slopes of the triangles will be calculated.

Range file	input		*.srf

the user supplied range file is used to split the triangles into groups of common slopes so that boundary strings can be created and the range colours used for faces. See 4.24 Defaults.

Slope type	input		percent cross fall, degrees, 1v in

the units used for slope in the range file.

Model for faces	input		available models

if non-blank, a faces will be created (with the colour given in the range file) for each triangle.

The z-value for a face is the line number in the range file of the range that the face satisfies.

Model for boundaries input available models

if non-blank, for each range of slopes in the range file, boundary strings will be created for regions of common slope. The boundary strings are given the name "slope boundaries" plus the number of the entry in the range file. Many of the boundaries will exist twice since they are the upper boundary for one range and the lower boundary for the next range but they will have a different colour and name. For **slope analysis 2 (new)***, the boundary name also has the addition of "face" and a number to more clearly differentiate the boundaries of the same range.*

Boundaries as faces tick box

if ticked, output the regions of common slope as faces.

Boundaries draped tick box

if ticked, the z-values for the boundary are taken from the tin.

If not ticked, the z-value for a boundary is the line number in the range file of the range that the boundary satisfies.

Boundary colour input cyan available colours

not used - was the colour for the slope boundaries.

Slope button

On selecting this button, the slopes of the triangles of the tin are calculated and the appropriate boundaries and faces created.

<esc> can be used to terminate the option during slope calculations.

16.11.2.3 Slope Analysis - Flag Slopes

Position of option on menu: **Tins =>Tin analysis =>Slope =>Slope analysis (flag)**

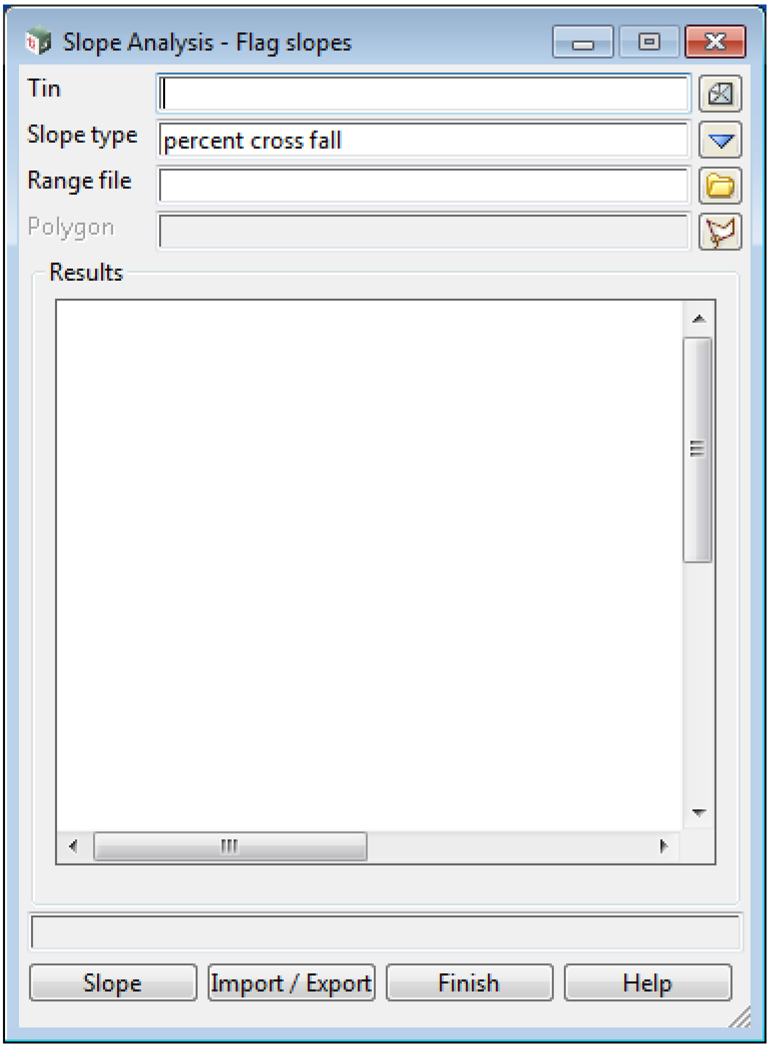
This panel analyses the slope of triangles in a tin and reports on triangles that fall within the supplied range file.

For more information on the slope range file, see 4.24 Defaults.

The results are listed in the results box.

Selecting a result line will pan the last selected view to display that triangle, as well as highlighting it on screen.

Selecting **Slope analysis (flag)** displays the **Slope Analysis - Flag Slopes** panel on the screen.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin			
the tin to analyse			
Slope type	choice box		percent cross file, 1v in, degrees minute seconds
the type of range file			

Polygon

an optional polygon to constraint the analysis within

Results

the list of results

Slope button

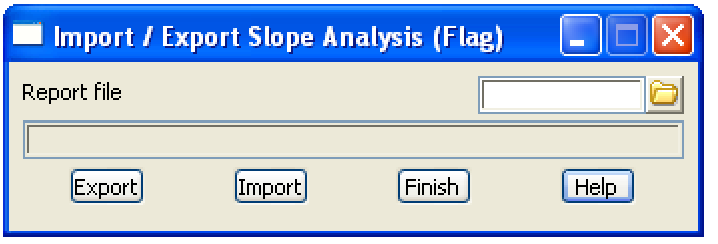
performs the analysis

Import/Export button

opens the 16.11.2.3.1 Import/Export Slope Analysis (Flag) *panel*

16.11.2.3.1 Import/Export Slope Analysis (Flag)

This panel imports and exports the results of a slope analysis.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Report file	file		

the file to read / write from

Export	button		

exports the results to the supplied file

Import	button		

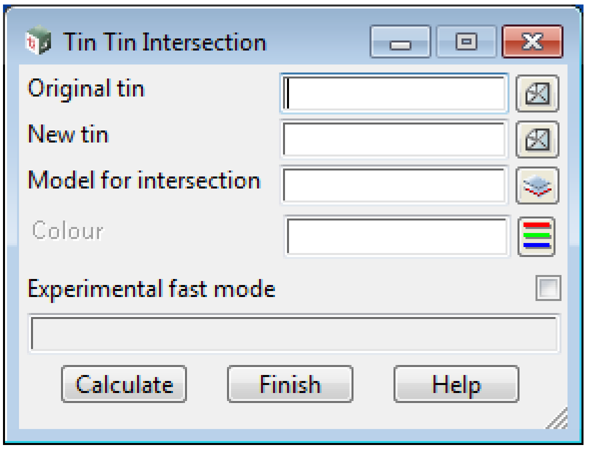
imports the results from the supplied file

16.11.3 Intersection

Position of option on menu: **Tins =>Tin analysis =>Intersection**

The **intersection** option finds the lines of intersection between two tins.

Selecting **intersection** displays the **Tin Tin Intersection** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Original tin	tin box		available tins

name of one of the tins to be intersected.

New tin	tin box		available tins

name of the other tin to be intersected.

Model for intersection	model box		available models

name of the model to contain the intersection strings.

Colour for intersection	colour box	default colour	available colours

colour to make the intersection strings

Experimental fast mode	tick box		

if ticked, a faster method is used to calculate the intersection strings

Calculate	button		

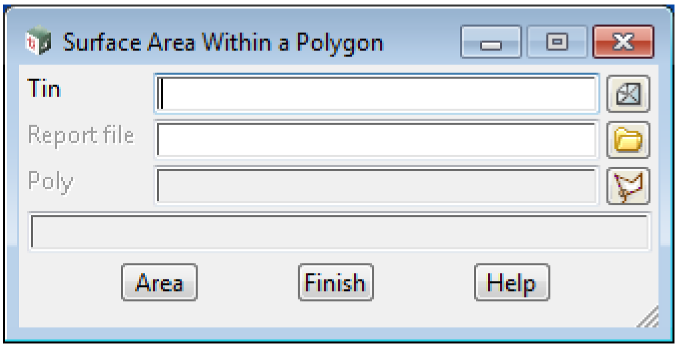
calculate the intersection between two tins given in the panel fields. The resulting intersection strings are added to the model given in the **model for intersection** *field.*

16.11.4 Surface Area

Position of option on menu: **Tins =>Tin analysis =>Surface area**

The **surface area** option is used to calculate the surface area of a tin within a user selected string.

Selecting **surface area** displays the **surface area within a polygon** panel.



The fields and buttons have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

the tin to calculate the surface area on.

Report file	file box		*.rpt files

if non-blank, the name of the file to write the surface area within the selected polygon to.

Poly	poly-select		

select the string to find the slope area within.

Area	button		

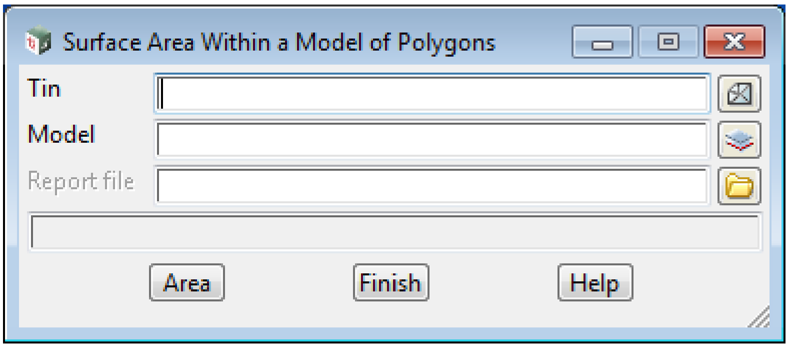
calculate the surface area of the tin within the selected polygon.

16.11.5 Surface Area 2

Position of option on menu: **Tins =>Tin analysis =>Surface area 2**

The **surface area 2** option is used to calculate the surface area of a tin within each polygon in a model.

Selecting **surface area 2** displays the **Surface Area Within a Model of Polygons** panel.



The fields and buttons have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

the tin to calculate the surface area on.

Model	model box		available models

the model of polygons for calculating surface area within.

Report file	file box		*.rpt files

if non-blank, the name of the file to write the surface area within each polygons in the model, plus the total of the surface area.

Area	button		

calculate the surface area of the tin within each polygons in the model, and sum the areas.

16.11.6 Flow Arrows

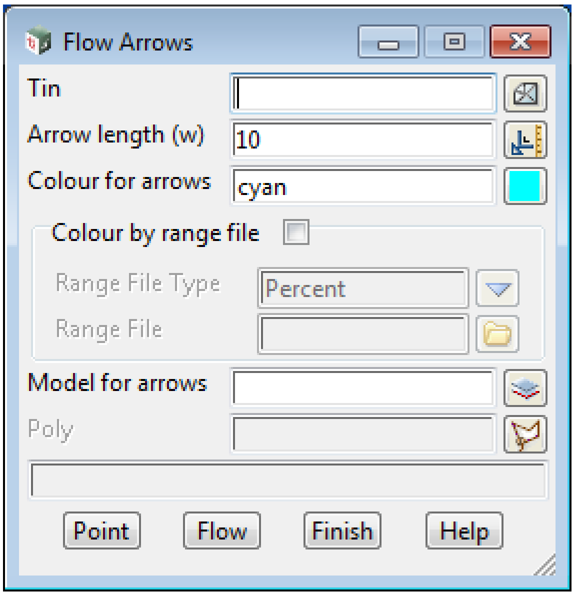
Position of option on menu: **Tins =>Tin analysis =>Flow arrows**

The flow arrows option draws arrows indicating the flow direction across triangles in a tin.

The flow arrows can be drawn for selected points, for all the triangles within a polygon, or if no polygon is selected, for all triangles in the tin.

The arrows are drawn at the centroid of the triangle and have a fixed user given length.

On selecting the **flow arrows** option, the **flow arrows** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin to calculate flow arrows for.

Arrow length (w)	real value box		

length in world units to draw the flow arrows.

Colour for arrows	colour box	cyan	available colours

colour for the arrows.

Colour by range file	tick box		

if ticked, the flow arrows are coloured by the their slope using a slope range file. See 4.24 Defaults.

Range file type	choice box	Percent	Percent, 1v in

the units used for slopes in the range file
***Percent** - percent cross fall*
***1v in** - one vertical in ? horizontal*

Range file	slope range file box		.srf files
Model for arrows	model box		available models

model to put flow arrows in. Must be non-blank.

Poly	polygon select		

if a polygon is selected, then the spot heights will be restricted to within the polygon.

If no polygon is selected, the spot heights will be calculated over the entire tin.

Point button

after selecting **Point***, a flow arrow is drawn at each selected point. This continues until cancel is selected from the* **pick ups** *menu.*

Flow button

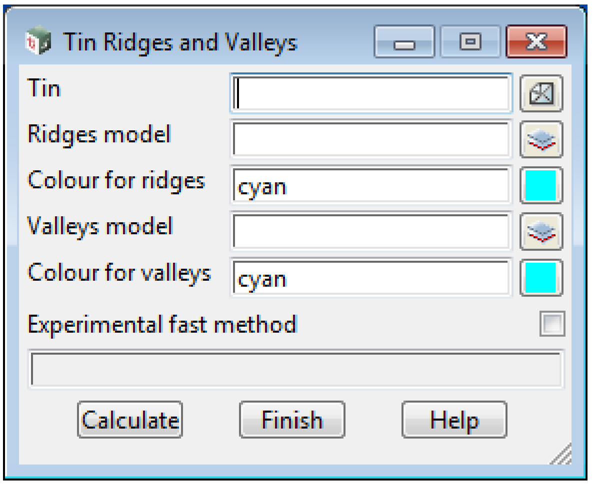
calculate the flow arrows for all the triangles in the tin, or if a polygon has been selected, for the triangles whose centroid is inside the polygon.

16.11.7 Ridge/Valleys

Position of option on menu: **Tins =>Tin analysis =>Ridges/Valleys**

In the **ridge/valleys** option, ridge and valley lines are calculated for the given tin.

On selecting the **ridge/valley** option, the **tin ridges and valleys** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin to calculate ridge and valley lines for.

Ridges model	model box		available models

if non-blank, the name of the model to contain the ridge lines. If blank, the ridge lines are not created.

Colour for ridges	colour box	default colour	available colours

colour for the ridge lines

Valleys model	model box		available models

if non-blank, the name of the model to contain the valley lines. If blank, the valley lines are not created.

Colour for valleys	colour box	default colour	available colours

colour for the valley lines

Calculate	button		

ridge and valley lines will be calculated for the given tin and placed in the ridge and valley models.

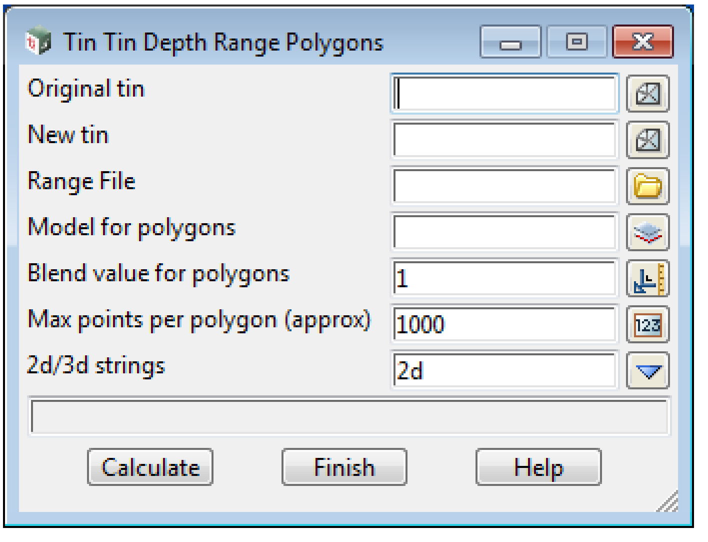
16.11.8 Depth Range Polygons

Position of option on menu: **Tins =>Tin analysis =>Depth range polygons**

In the **Depth range polygons** option, polygons are created around the regions given by a depth range file.

For more information on the depth range file, see 8.9.4 Depth Range File.

On selecting the **Depth range polygons** option, the **Tin Tin Depth Range Polygons** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Original/New Tin	tin box		available tins

name of the original/new tin for determining depths between.

Range file	input		*.drf

the user supplied range file for splitting up the depths and colouring the polygons. See 8.9.4 Depth Range File.

Model for polygons	model box		available models

model to contain the polygons.

Blend value for polygons	real value	1	

The value of blending is between 0 and 1.

0 means the polygon is totally transparent (and hence invisible) and 1 means that the polygon is opaque (non-translucent) and can't be seen through at all.

If blank, then the value is taken as 1 and the polygon is opaque (non-translucent).

Max points per polygon (approx)	real value	1000	

when a polygon is created and the number of vertices in it is greater than this value, the polygon will be split into a number of polygons.

2d/3d strings	choice box	2d	2d, 3d original, 3d new

*If **2d**, the created strings are 2d strings with a z-value equal to the depth.*
*If **3d original**/new, the depth strings are draped over the original/new tin to form 3d strings.*

Calculate	button		

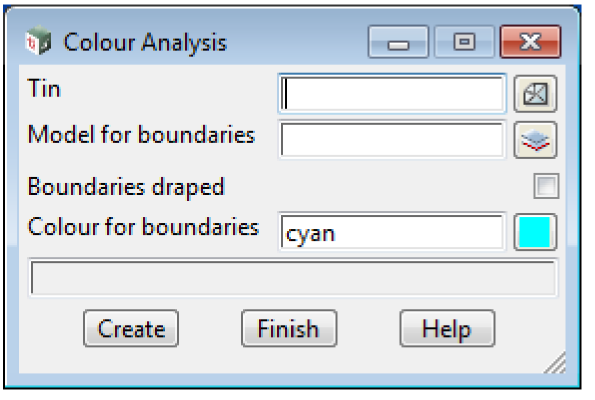
create the polygons surrounding regions of different heights between the tins.

16.11.9 Polygons from Tin Colours

Position of option on menu: **Tins =>Tin analysis =>Polygons from colours**

In the **Polygons from colours** option, polygons are created around the different coloured regions of the tin.

On selecting the **Polygons from colours** option, the **Colour Analysis** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin to calculate polygons around the coloured regions.

Model for boundaries	model box		available models

model to contain the polygons.

Boundaries draped	tick box		

if ticked. the z-values for the vertices for the boundary strings come from the vertices of the coloured triangles.

If not ticked, the boundaries have constant z-values.

Colour for boundaries	colour box	default colour	available colours

colour for the polygons

Create	button		

create the polygons surrounding regions of different colours on the tin.

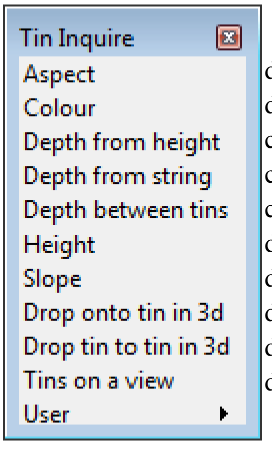
16.12 Inquire

Position of menu: **Tins =>Inquire**

The **inquire** options are used to display at the current cursor position the height (z-value) on a tin, the colour of the triangle, the depth from a fixed height to a tin, and the depth between two tins,.

Once the user specifies which tin or tins are to be analysed, then as the cursor moves around in any plan view, the triangles below the cursor are examined and the appropriate information displayed.

The **tin inquire** walk-right menu is



display aspect of triangles
display colour of tin
calculate depth from height to tin
calculate depth from string to a tin
calculate depth between tins
display height on tin
display slope of triangles
drop onto a tin in 3d
display perpendicular height between two tins
dynamically list all tins under the cursor

The options in the **Tin inquire** menu will now be discussed.

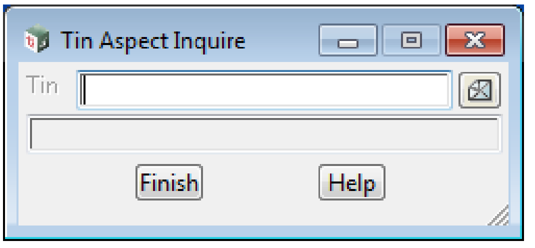
For the options *Aspect*, go to	16.12.1 Aspect Inquire.
Colour	16.12.2 Colour Inquire
Depth from height	16.12.3 Depth from Height
Depth from string	16.12.4 Depth from String
Depth between tins	16.12.5 Depth Between Tins
Height	16.12.6 Height Inquire
Slope	16.12.7 Slope Inquire
Drop onto tin 3d	16.12.8 Tin Drop Point 3d
Drop tin to tin in 3d	16.12.9 Tin Tin Drop Point 3d
Tins on a view	16.12.10 Tins on View Inquire

16.12.1 Aspect Inquire

Position of option on menu: **Tins =>Inquire =>Aspect**

The **aspect inquire** option calculates and displays the aspect (direction) of the triangles under the current plan view cursor position. The aspect is displayed in bearings in degrees, minutes and seconds.

Selecting **tin aspect inquire** displays the **tin aspect inquire** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

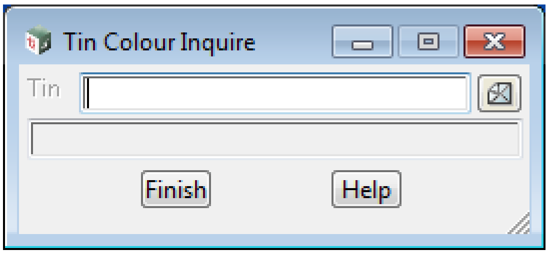
name of the tin for which the aspect of the triangle that the cursor is above will be displayed in the panel's message area.

16.12.2 Colour Inquire

Position of option on menu: **Tins =>Inquire =>Colour**

colour inquire displays the colour of the triangles under the current plan view cursor position.

Selecting **tin colour inquire** displays the **tin colour inquire** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

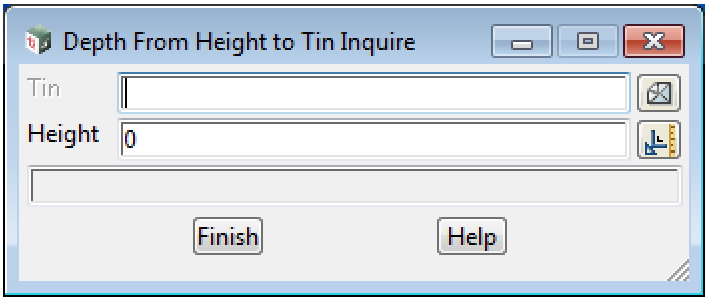
name of the tin for which the colour of the triangle at the current cursor position is displayed in the panel's message area.

16.12.3 Depth from Height

Position of option on menu: **Tins =>Inquire =>Depth from height**

The **depth from height** option calculates and displays the difference between the z-value of a tin and a given height for the current plan view cursor position.

Selecting **depth from height** displays **depth from height to tin inquire** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin used for calculating the distance from the tin to the given height at the cursor position.

Height	input	0	

the difference between the tin value and the value given in this field is displayed in the panel's message area.

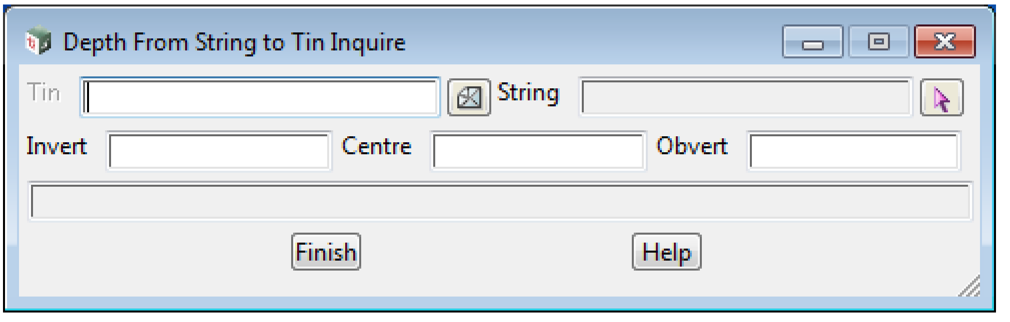
16.12.4 Depth from String

Position of option on menu: **Tins =>Inquire =>Depth from string**

The **depth from string** option calculates and displays the difference between the z-value of a tin and the height of a selected string at the current plan or section view cursor position projected back onto the selected string.

The obvert (top of the string) value is also known as the cover above a string to a tin.

Selecting **Depth from string** displays the **Depth From String to Tin Inquire** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin used for calculating the z-value at the cursor position.

String	string-select		

if a string is selected, the difference between the tin value and the height of the selected string (to the invert, centre and obvert of the string) at the current plan or section view cursor position, projected back onto the selected string, is displayed in the panel's message areas.

Invert/Centre/Obvert	output		

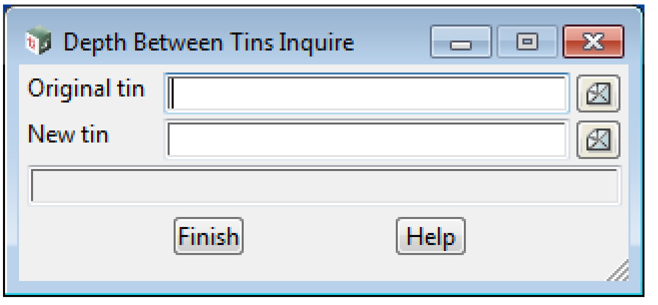
displays the distance between the tin and the invert (bottom), centre (axial) and obvert (top) of the selected string

16.12.5 Depth Between Tins

Position of option on menu: **Tins =>Inquire =>Depth between tins**

The **depth between tins** option calculates and displays the difference between the z-values of two given tins at the current plan view cursor position.

Selecting **depth between tins** displays the **depth between tins inquire** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Original/new tin	tin box		available tins

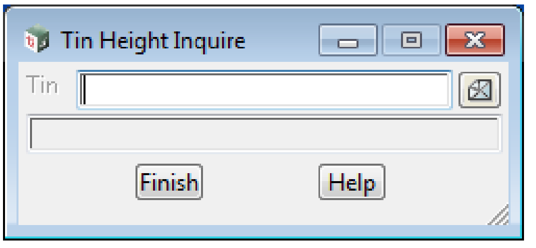
name of the two tins or which the z-value is to be calculated at the current plan view cursor position and the difference between the z-values (z_new - z_original) is displayed in the panel's message area.

16.12.6 Height Inquire

Position of option on menu: **Tins =>Inquire =>Height**

The **height inquire** option calculates and displays the height (z-value) of triangles from a tin at the current plan view cursor position.

Selecting **tin height inquire** displays the **tin height inquire** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

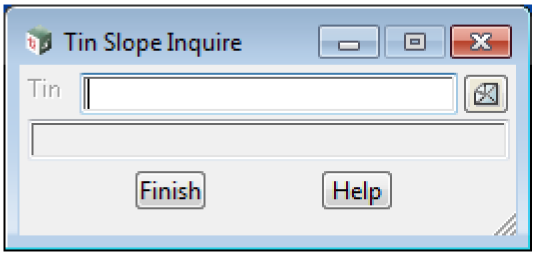
name of the tin for which the z-value at the current plan view cursor position is displayed in the panel's message area.

16.12.7 Slope Inquire

Position of option on menu: **Tins =>Inquire =>Slope**

The **slope inquire** option calculates and displays the slope of triangles from a tin at the current plan view cursor position. The slope is displayed as a percent cross fall, a "1v in "slope and as an angle in degrees, minutes and seconds.

Selecting **tin slope inquire** displays the **tin slope inquire** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

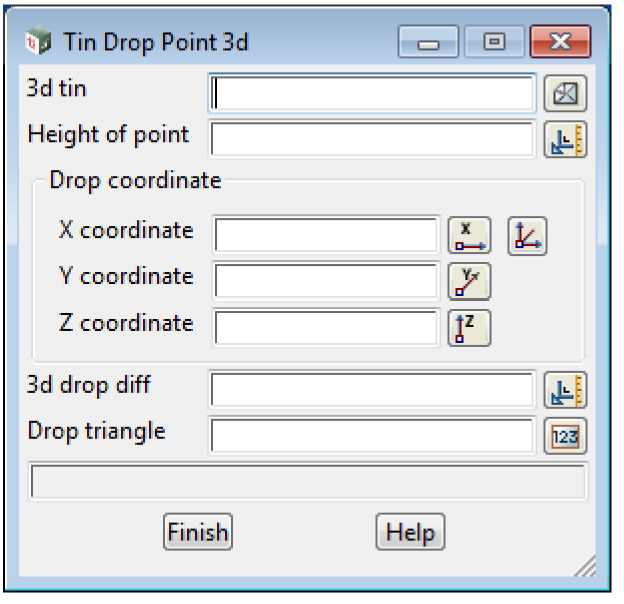
name of the tin for which the slope of the triangle that the cursor is above is displayed in the panel's message area.

16.12.8 Tin Drop Point 3d

Position of option on menu: **Tins =>Inquire =>Drop onto tin in 3d**

The **Drop onto tin in 3d** option takes a cursor position and a user defined height of the point and drops perpendicularly onto the closest triangle.

Selecting **Drop onto tin in 3d** displays the **Tin Drop Point 3d** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
3d tin	tin box		available tins

name of the tin to drop the point perpendicularly onto.This cannot be a Super Tin.

Height of point	real value		

the height to use at the (x,y) cursor position.

Drop coordinates

X, Y, Z coordinate of the point dropped perpendicularly to the tin from the (x,y) cursor position and height from **Height of point** *field*

3d drop diff	real		

the perpendicular distance to the tin.
positive if the point being dropped is above the tin
negative if the point being dropped is below the tin

Drop triangle	number box		

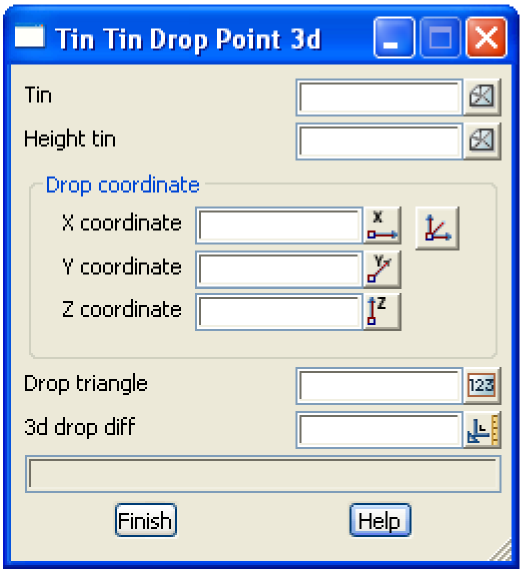
the number of the triangle the point was dropped to.

16.12.9 Tin Tin Drop Point 3d

Position of option on menu: **Tins =>Inquire =>Drop tin to tin in 3d**

The **Tin Tin Drop Point 3d** panel allows the 3d, (perpendicular) height difference between 2 tins to be displayed at any position. Once both tins are selected and a valid 3d drop can occur the difference is displayed and the triangle dropped to highlighted.

Selecting **Drop tin to tin in 3d** brings up the **Tin Tin Drop Point 3d** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

3d Tin

the tin to drop the point to perpendicularly in 3d, this tin cannot be a Super Tin.

Height tin

the tin to get a z value for the current position to use in the 3d drop, this tin can be a Super Tin.

Drop coordinate

*the x, y, z positions of the drop on the **3d Tin**.*

3d drop diff

*the 3d, (perpendicular) difference between the 2 tins, a positive value means the **3d Tin** is below the **Height tin** at the current position, a negative value the **3d Tin** is above the **Height tin**.*

Drop triangle

*the index of the triangle dropped to in the **3d Tin**.*

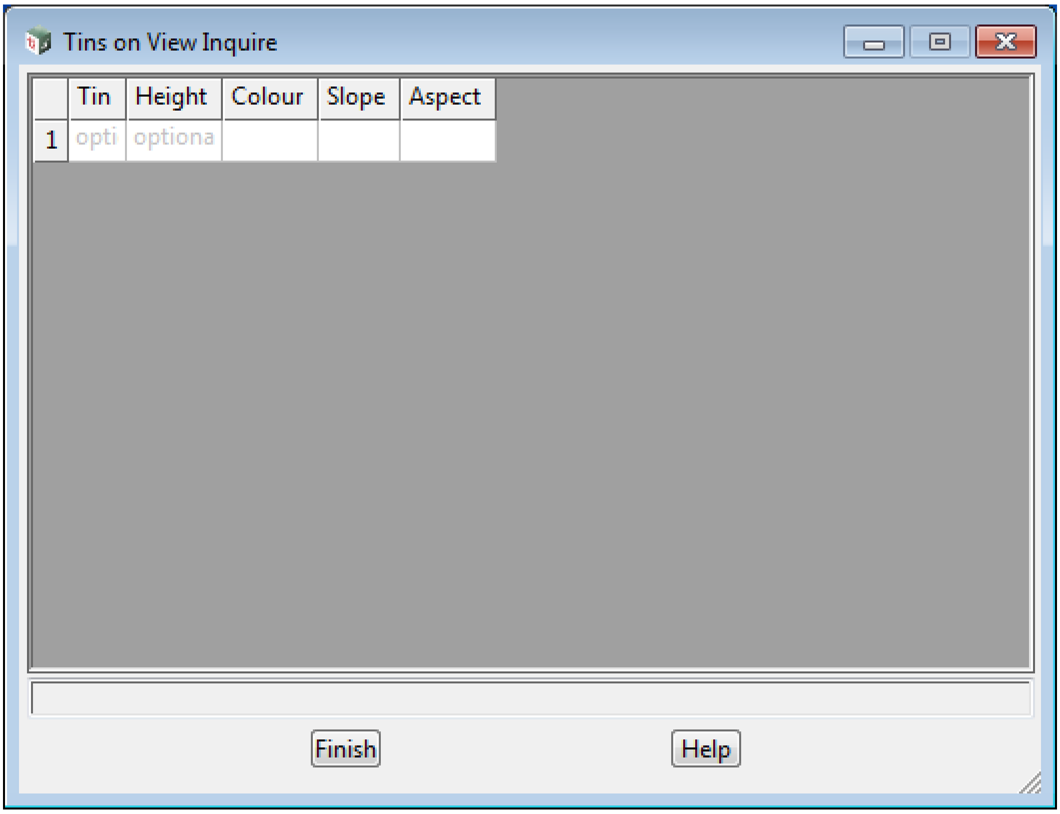
16.12.10 Tins on View Inquire

Position of option on menu: **Tins =>Inquire =>Tins on a view**

As the cursor is moved over any plan view, the **Tins on a view** option lists all the tins and super tins on that view, and at the (x,y) location of the cursor, dynamically displays the z-value of each tin or super tin at that (x,y) location, and the colour, slope and aspect of the triangle from the tins or super tins currently under the cursor.

For a super tin, it also displays the name of the tin being used from the super tin at that cursor location.

Selecting **Tins on a view** brings up the **Tins on View Inquire** panel.



The fields and buttons used in the this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	output		

when the cursor is over a plan view, the **Tin** *column list all the tins and super tins in the view.*

When the cursor is actually over a tin or super tin, on the row in the grid for that tin or super tin, the height of the tin or super tin at the cursor (x,y) location, and the colour, slope and aspect of the triangle under the cursor is displayed in the rest of the columns of the grid.

If it is a super tin and the cursor is over the super tine, then the name of the super tin is given and then a "/" and then the name of the tin from the super tin that is being used to give the height, colour, slope and aspect. For example, in the Tin column the name "combined/final" means that the tin the cursor is over is a super tin called "combined" and it is using the tin called "final" from the super tin "combined" at that location.

Height output

when in a plan view and the cursor is actually over the tin given in the Tin column, this is the height *from the tin/super tin at the cursor location.*

Colour/Slope/Aspect output

when in a plan view and the cursor is actually over the tin or super tin given in the Tin column, this is the colour/slope/aspect of the triangle from the tin or super tin at that cursor location.

16.13 Null

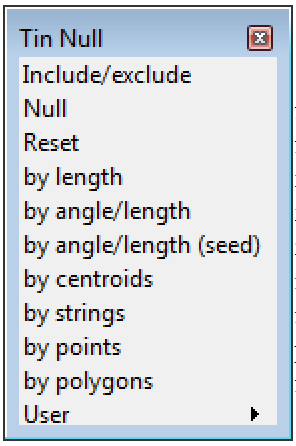
Position of menu: **Tins =>Null**

The triangulation process forms triangles throughout the convex polygon enclosing the data set. This means that triangles may cross regions where there is very little data and may produce strange results in the ill-defined regions.

In 12d Model, it is possible to make triangles invisible (called nulling triangles) so that they are not used in any options.

Note - nulled triangles are not deleted and can be made visible again at any time.

The **null** walk-right menu is



set a polygon to include/exclude all triangles inside the polygon
null triangles outside a polygon
reset all null triangles
null triangles by length of sides
null triangles by length of sides and angle
null by angle and length starting at a seed point
null/reset triangle under the centroid of strings
null triangles that are crossed by strings
pick triangles to null/reset
null/reset triangles inside/outside polygons

The three options in the tin colouring menu will now be discussed.

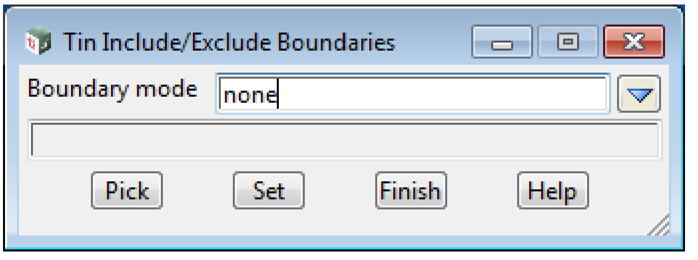
For the options *Include/exclude*, go to 16.13.1 Include/Exclude Boundaries.
Null 16.13.2 Null
Reset 16.13.3 Reset
by length 16.13.4 Null by Length
by angle/length 16.13.5 Null by Angle and Length
by angle/length (seed) 16.13.6 Null by Angle and Length (Seed)
by centroids 16.13.7 Null by Centroids
by string 16.13.8 Null by Strings
by points 16.13.9 Null by Points
by polygons 16.13.10 Null by polygons

16.13.1 Include/Exclude Boundaries

Position of option on menu: **Tins =>Null =>Include/excludes**

The **Include/exclude** option is used to tag a polygon that is part of the data set for a Tin so that all the triangles inside the polygon are either set to null or non-null.

On selecting the **Include/exclude** option, the **Tin Include/Exclude Boundaries** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Boundary mode	choice box	none	none, include, exclude

mode to set the polygon to.

If ***exclude****, then any triangles inside the polygon are set to null.*

If ***include****, then any triangles inside the polygon are reset so that they aren't null.*

If ***none****, then the polygon is not used for automatically nulling/resetting triangles.*

Pick	button		

select the polygon to define a boundary mode for.

Set	button		

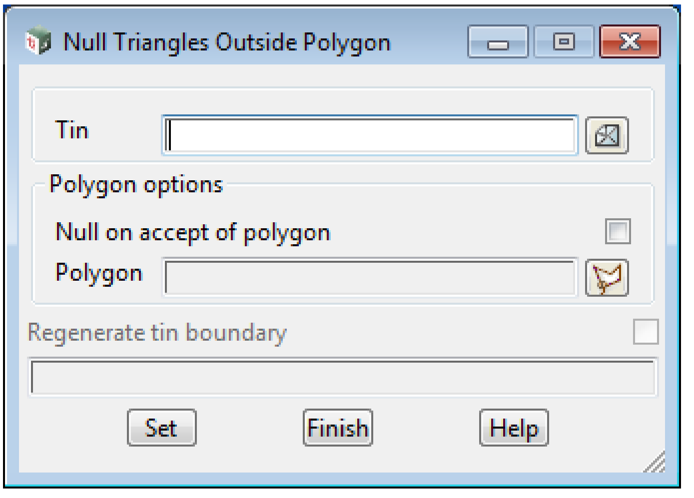
set the boundary mode for the selected polygon.

16.13.2 Null

Position of option on menu: **Tins =>Null =>Null**

The **null** option is used to null any triangle whose *centroid* is outside a user selected polygon.

Selecting **Null** displays the **Null Triangles Outside Polygon** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin to null triangles in.

Null on accept of polygon tick box

if ticked, the triangles will be nulled as soon as the string is accepted.

If not ticked, the triangles will not be nulled until the **Set** *button is selected.*

Polygon polygon select

the string to be used as the polygon is selected after choosing this button. If **Null on accept of polygon** *is set, then the polygon is processed immediately.*

Set button

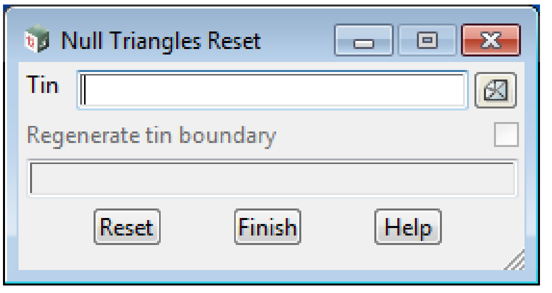
process the selected string/. Any triangle whose centroid is outside the selected boundary polygon is set to null.

16.13.3 Reset

Position of option on menu: **Tins =>Null =>Reset**

The **reset** option is used to change all the nulled triangles in a tin back to the visible triangles.

Selecting **reset** displays the **Null Triangles Reset** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin to have any nulled triangles reset to being visible triangles.

Reset	button		

the **reset** *button resets* ***all*** *null triangles in the tin to the base colour.*

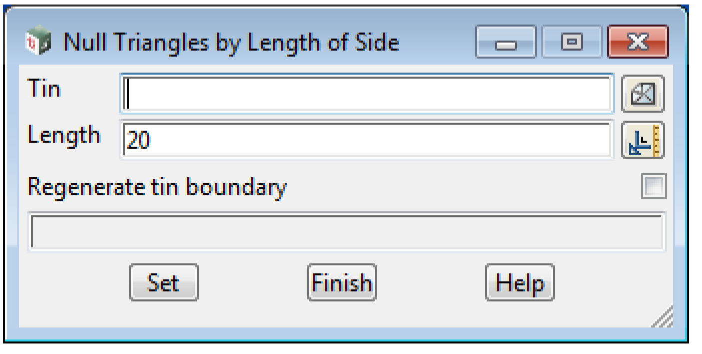
16.13.4 Null by Length

Position of option on menu: **Tins =>Null =>by length**

The **null by length** option is used to null any triangle with a side of length greater than a user specified length.

Warning - this option will null out *internal* triangles. The next option, Null by angle/length will only null triangles on the outer shell.

On selecting the **by length** option, the **null triangles by length of side** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin to null triangles in.

Length	input		

length to check triangle sides against.

Set	button		

after selecting the **set** *button, the tin will be processed and any triangle with a side of length greater than the value given in the* **length** *panel field, will be nulled.*

16.13.5 Null by Angle and Length

Position of option on menu: **Tins =>Null =>by angle/length**

The **null by angle/length** option is a more powerful nulling option which tries to remove most of the external triangles that one expects should be removed. For example, for a T-intersection, all the long, thin outer triangles should be removed. The option has two sets of tests - one to remove long thin triangles and the other which uses a combined test to remove squat triangles.

Unlike the *null by length* option, *null by angle/length* only works on triangles which have an outer null triangle on one or two sides (it does not work on internal holes). These triangles are called *external triangles* and sides with an outer null triangle on them are called *external sides*.

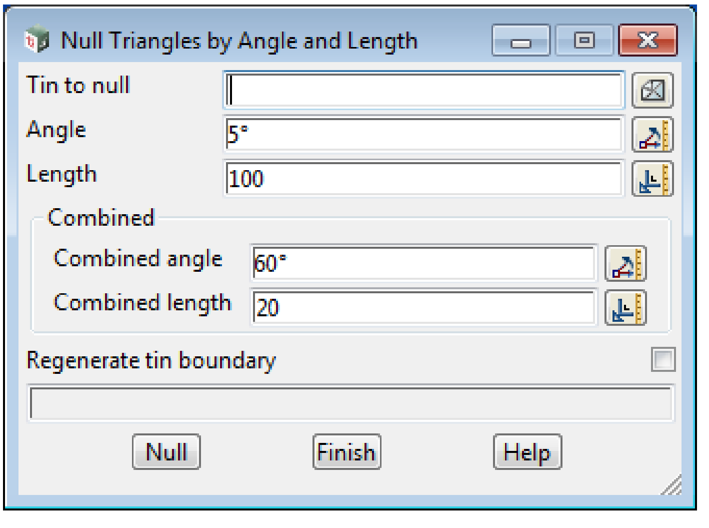
The *null by angle and length* option *does not* remove triangles whose external side is a breakline. Hence, as expected, breaklines will stop the nulling process.

Note that *null by angle and length* keeps processing the remaining non-nulled triangles until no more can be nulled or are stopped by breaklines. Hence the nulling works inwards as outer triangles are nulled revealing new external triangles to be tested. The nulling process is like peeling off layers of an onion.

Important note - the setting for this option are ***stored*** by the triangulation and the option is automatically re-run if the triangulation is re-run. To turn off the tests, just set the panel fields to blank.

Warning - this option should be the first nulling option used because all null triangles are first reset to valid triangles before this option is applied.

On selecting the null **by angle/length** option, the **Null Triangles by Angle and Length** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin to null	tin box		available tins

name of the tin to have triangles nulled. If this option has been previously run on the tin, the previous panel values are written into the panel fields.

Angle	angle box	5	

if a triangle has an external side (that is not a breakline) with an angle on it less than **Angle**, *then the triangle is nulled. The default value works most of the time.*

If blank then no triangles are nulled by this test.

Length input 100

if a triangle has an external side (that is not a breakline) greater than **Length***, the triangle is nulled. If blank then no triangles are nulled by this test.*

Combined angle angle box 60
Combined length input 20

For the Combined case, a triangle is nulled if it:

has an external side (that is not a breakline) and the ***sum*** *of the two angles on it is* ***less*** *than* **Combined angle** *(the default value works most of the time)*
and

has an external side (that is not a breakline) whose length is ***greater*** *than* **Combined length***. A suggested value is one third to one half of Length.*

If either **Combined angle** *or* **Combined length** *is blank, then no triangles are nulled by this test.*

Null button

after selecting the **Null** *button, all nulled triangles are reset to be valid triangles and the tin then processed and triangles nulled.*

16.13.6 Null by Angle and Length (Seed)

Position of option on menu: **Tins =>Null =>by angle/length (seed)**

The **null by angle/length (seed)** option is similar to the powerful null by angle/length option (see 16.13.5 Null by Angle and Length) except it has the additional ***start point***.

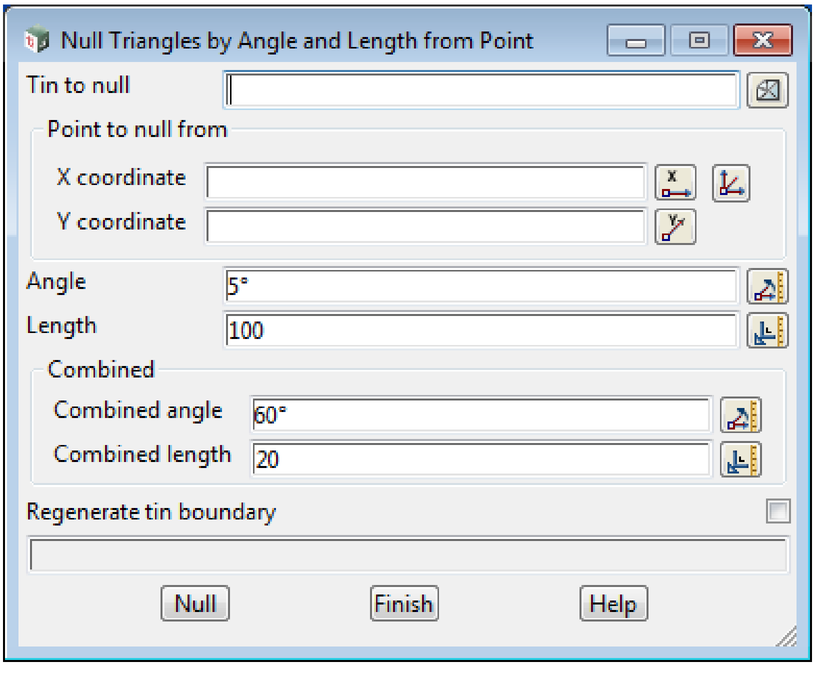
That is, a start triangle is selected by the cursor and the nulling starts from the triangle under the cursor.

The *null by angle/ length (seed)* option then starts nulling triangles from the start triangle using the angle/length parameters until stopped by breaklines.

Hence the nulling works ***outwards*** from the start triangle revealing new triangles to be tested until stopped by breaklines.

This option is particularly useful for nulling internal areas of a tin.

Selecting null **by angle/length (seed)** displays the **Null Triangles by Angle and Length from Point** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin to null	tin box		available tins

name of the tin to have triangles nulled. If this option has been previously run on the tin, the previous panel values are written into the panel fields.

Point to null from

the X/Y co-ordinates of a position to start nulling from.

Angle	angle box	5	

if a triangle has an external side (that is not a breakline) with an angle on it less than **Angle***, then the triangle is nulled. The default value works most of the time.*

If blank then no triangles are nulled by this test.

Length	input	100	

if a triangle has an external side (that is not a breakline) greater than **Length***, the triangle is nulled.*

If blank then no triangles are nulled by this test.

Combined angle input 60
Combined length input 20

For the Combined case, a triangle is nulled if it:

*has an external side (that is not a breakline) and the **sum** of the two angles on it is **less** than* **Combined angle** *(the default value works most of the time)*
and

*has an external side (that is not a breakline) whose length is **greater** than* **Combined length**. *A suggested value is one third to one half of Length.*

If either **Combined angle** *or* **Combined length** *is blank, then no triangles are nulled by this test.*

Null button

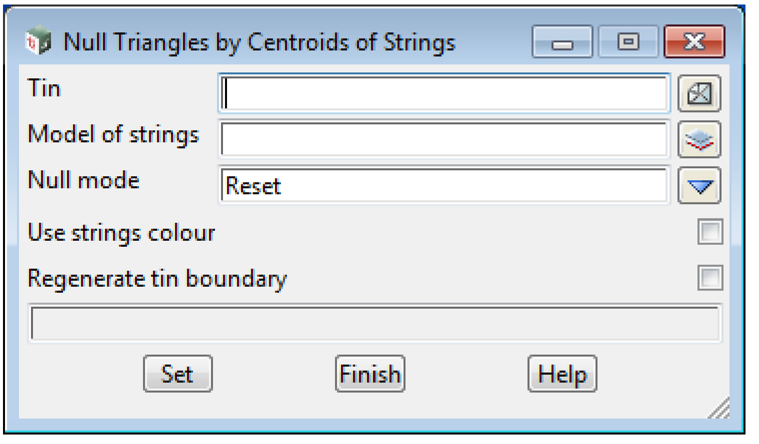
after selecting the **Null** *button, the triangle under the (X,Y) position is determined and null processing starts from that triangle.*

16.13.7 Null by Centroids

Position of option on menu: **Tins =>Null =>by centroids**

null by centroids is used to null/reset any triangle that lies under the centroid of a string.

On selecting the **by centroids** option, the **null triangles by centroids of strings** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin to null/reset triangles in.

Model of strings	model box		available models

model of strings to be used to select triangles that are under a centroid of one of the model's strings.

Null mode	choice box	Reset	Reset, Null

*if **null**, any triangle under the centroid of a string from the* **model of strings***, will be nulled.*
reset**, any triangle under the centroid of a string from the* **model of strings, will be made visible again.*

Set	button		

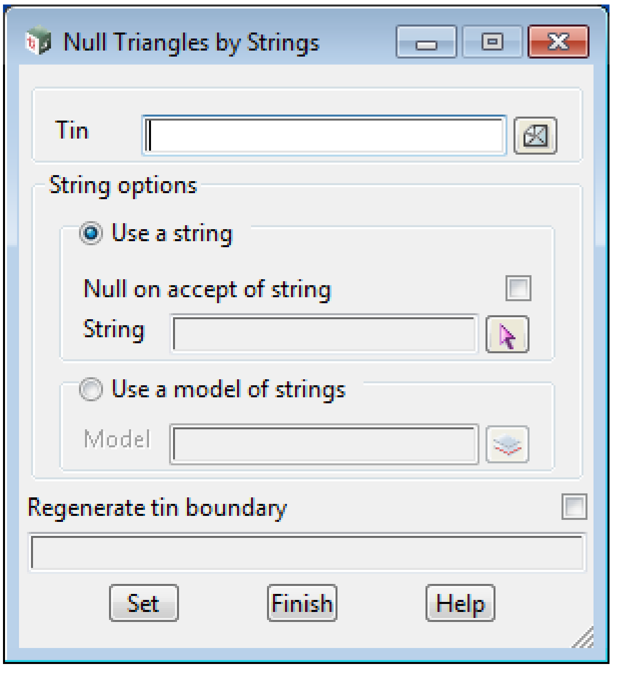
after selecting the **set** *button, the tin will be processed and any triangle under the centroid of a string will be set according to the* **null mode** *panel field.*

16.13.8 Null by Strings

Position of option on menu: **Tins =>Null =>by strings**

The **null by strings** option is used to null any triangle that is cut by a string.

On selecting the **by strings** option, the **null triangles by strings** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin to null triangles in.

Use a string	radio button		

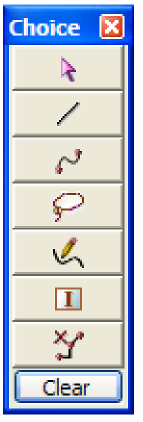
if set, strings will be selected and used to null any triangle that the string passes through.

Null on accept of string	tick box		

if ticked, the triangles will be nulled as soon as the string is accepted.

If not ticked, the triangles will not be nulled until the **Set** *button is selected.*

String	string select	existing string	string pop-up



if string, a strings will be selected and used to null any triangle that the string passes through.

If line, a temporary two point line is drawn and used to null any triangle that the line passes through

If polyline, a temporary polyline is drawn and used to null any triangle that the string passes through

If closed polyline, a temporary closed polyline is drawn and used to null any triangle that the string passes through

Use a model of strings radio button

if set, a model of strings is used and a triangle is nulled if it is cut by any one of the strings.

Model model box available models

model of strings to cut the triangles.

Set button

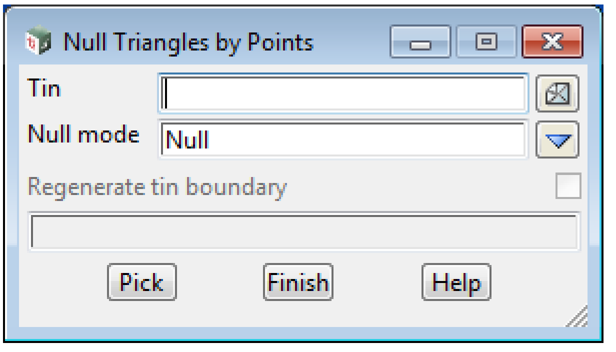
after selecting the **set** *button, the tin will be processed and any triangle that is cut by a string from the* **Model***, or a selected string for the* **User a string** *case, will be nulled.*

16.13.9 Null by Points

Position of option on menu: **Tins =>Null =>by points**

The **null by points** null or reset any triangle that the user clicks LB whilst over the triangle in a plan view.

On selecting the **by points** option, the **null triangles by points** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin to null/reset triangles in.

Field Description	Type	Defaults	Pop-Up
Null mode	choice box	Null	Reset, Null

*if **null**, any triangles under a cursor pick will be nulled.*
* **reset**, any nulled triangles under a cursor pick will be turned back on.*

Field Description	Type	Defaults	Pop-Up
Pick	button		

any triangle under the cursor select will be processing according to the **null mode** *panel field.*

16.13.10 Null by polygons

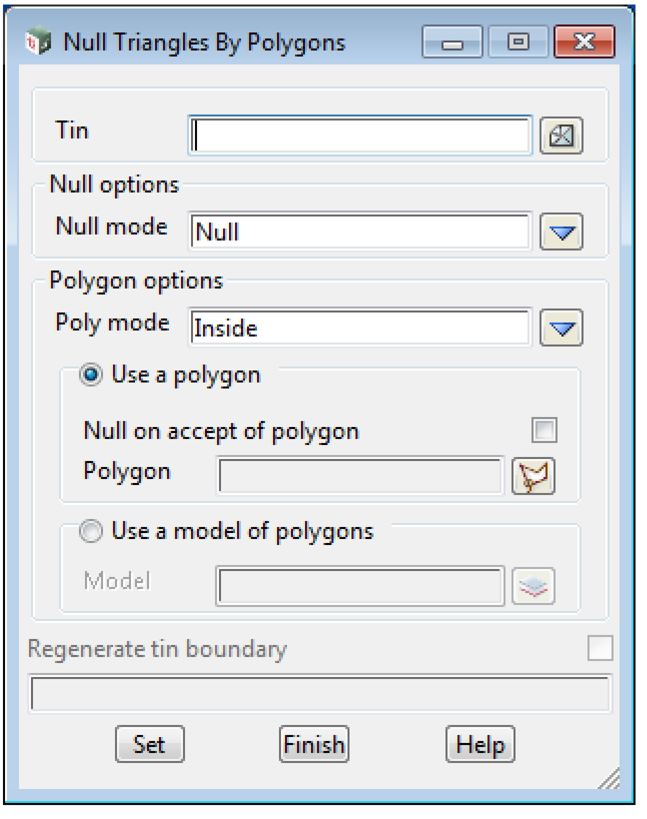
Position of option on menu: **Tins =>Null =>by polygons**

The **null by polygons** option is used to null/reset any triangles that are inside or outside a user selected polygon, or model of polygons.

If a string is selected to use and it is not closed, then a polygon is formed by joining the first and the last points of the string.

On selecting the **by polygons** option, the **null triangles by polygons** panel is displayed

.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin to null/reset triangles in.

Null mode	choice box	Null	Reset, Null

*if **null**, any triangles selected will be nulled.*
__reset__, any nulled triangles selected will be turned back on.

Poly mode	choice box	inside	inside, outside

*mode to select whether the triangles with centroids **inside** or **outside** the selected polygons are to be nulled.*

Use a polygon	radio button		

if set, strings will be selected and used as the polygon to null/reset all triangles with centroids inside/ outside the polygon.

Null on accept of polygontick box

if ticked, the triangles will be nulled/reset as soon as the polygon (string) is accepted.

If not ticked, the triangles will not be nulled/reset until the **Set** *button is selected.*

Null on accept of string tick box

if ticked, the triangles will be nulled as soon as the string is accepted.

If not ticked, the triangles will not be nulled until the **Set** *button is selected.*

Polygon polygon select polygon pop-up

polygon to null or rest triangles when the centroid is inside/outside the polygon

Use a model of polygons radio button

if set, a model of strings will be used as the polygons to null/reset triangles whose centroids are inside/ outside one of the polygons.

Model model box available models

model of strings to provide the polygons for nulling/resetting triangles.

Set button

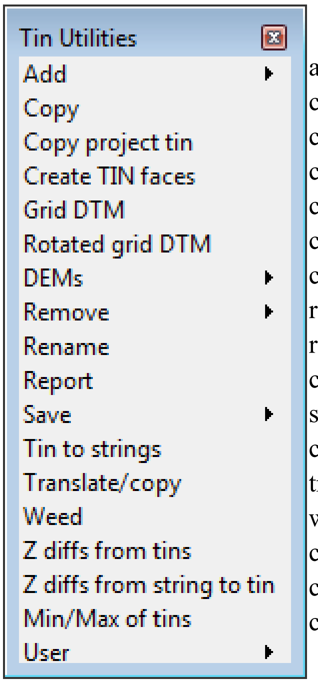
each string in the **model of polygons** *is used to null/reset the triangles in the tin given in the* **tin** *field according to the* **null mode** *and* **poly mode** *panel field.*

16.14 Utilities

Position of option on menu: **Tins =>Utilities =>Tin Utilities**

The **triangles=>utilities** menu contains miscellaneous options involving tins.

The **Utilities** walk-right menu is



add tins to project and models
create a copy of a tin
copy a tin from another project
create faces from a tin
create regular grid of points on a tin
create a rotated grid of points on a tin
create Digital Elevation Models
remove tins from project and models
rename a tin
create report of a tin
save tins to disk
create points and breaklines from a tin
translate/copy a tin
weed duplicate points from the tin database
create vertices with z-difference between two tins
create vertices with z-difference between string and a tin
calculate a grid of min and max values for the tin

Each option will now be described.

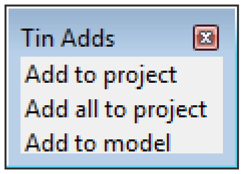
For the options *Add*, go to 16.14.1 Add
Copy 16.14.2 Copy
Copy project tin 16.14.3 Copy Project Tin
Create Tin faces 16.14.4 Create Tin Faces
Grid DTM 16.14.5 Grid DTM
Rotated grid DTM 16.14.6 Rotated Grid
DEMs 16.14.7 DEMs
Remove 16.14.8 Remove
Rename 16.14.9 Rename
Report 16.14.10 Report
Save 16.14.11 Save
Tin to strings 16.14.12 Tin to Strings
Translate/copy 16.14.13 Translate/Copy
Weed 16.14.14 Weed
Z diffs from tins 16.14.15 Z Differences from Tins
Z diffs from string to tin 16.14.16 Z Differences from String to Tin
Min/ Max of tins 16.14.17 Grid of Min / Max of Tins

16.14.1 Add

Position of menu: **Tins =>Utilities =>Add**

Tins can be added to the project and to models.

The **tin adds** walk-right menu is



The options in this menu will now be described.

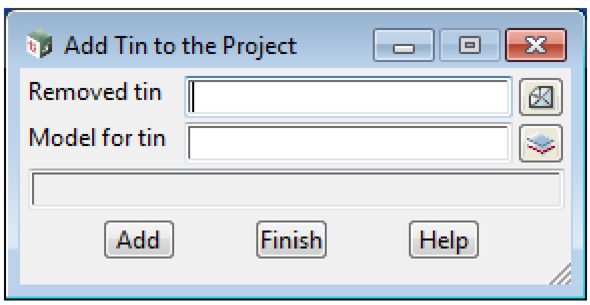
For the option Add to project, go to 16.14.1.1 Add to Project
Add all to project 16.14.1.2 Add All To Project
Add to model 16.14.1.3 Add to Model

16.14.1.1 Add to Project

Position of option on menu: **Tins =>Utilities =>Add =>Add to project**

The **add to project** option is used to add a removed tin back into the project.

On selecting the **add tin to project** option, the **add tin to the project** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Removed tin	tin box		available tins

name of the tin to be added to the working project. The tin must exist on the disk in the project area and not already be in the project.

Model for tin	model box		available models

if non-blank, the loaded tin will be added to the model given in this field.

Add	button		

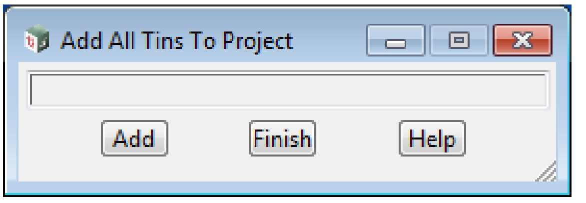
Add the tin given in the Tin field to the working project.
If the **model for tin** *field is non-blank, the tin will be added to the model given in that field - if the model does not exist, it will be created.*

16.14.1.2 Add All To Project

Position of option on menu: **Tins =>Utilities =>Add =>Add all to project**

The **add all tins** option is used to add all the removed tins back into the project.

On selecting the **add all to project** option, the **add all tins to project** panel is displayed.



The fields and buttons used in this panel have the following functions.

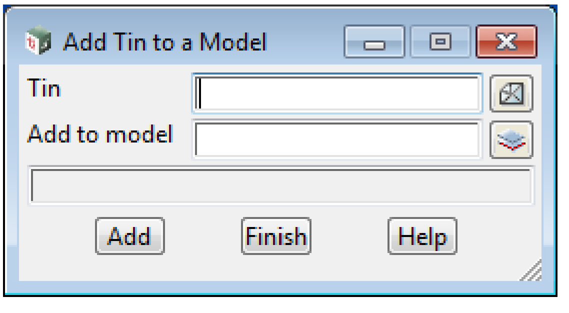
Field Description	Type	Defaults	Pop-Up
Add	button		

after selecting this button, all removed tins in the working project will be added to the project.

16.14.1.3 Add to Model

Position of option on menu: **Tins =>Utilities =>Add =>Add to model**

On selecting the **add to model** output option, the **add tin to a model** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin to be added to a model.

Field Description	Type	Defaults	Pop-Up
Add to model	model box		available models

name of the model to which the tin given in the **tin** *field will be added.*

Field Description	Type	Defaults	Pop-Up
Add	button		

add the tin given in the **tin** *field to the model given by the* **add to model** *field.*

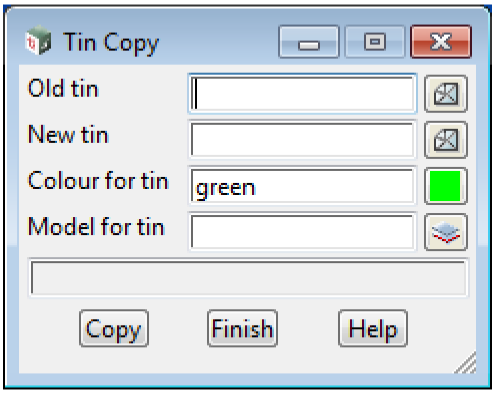
16.14.2 Copy

Position of option on menu: **Tins =>Utilities =>Copy**

A copy of an existing tin can be made using the **copy** option.

A Super Tin can also be copied and what is created is a new super tin with a new name but with exactly the same tins making it up as the original tin.

On selecting **copy**, the **tin copy** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Old tin	tin box		available tins

name of the tin to be copied.

New tin	tin box		available tins

name of the copy of the tin - this can't be the same as any existing tin in the project.

Colour of tin	colour box		available colours

if non-blank, the colour of the new tin. If blank, use the old tin colour

Model for tin	model box		available models

if non-blank, the name of the model for the new tin. If blank, the tin is not put in any model.

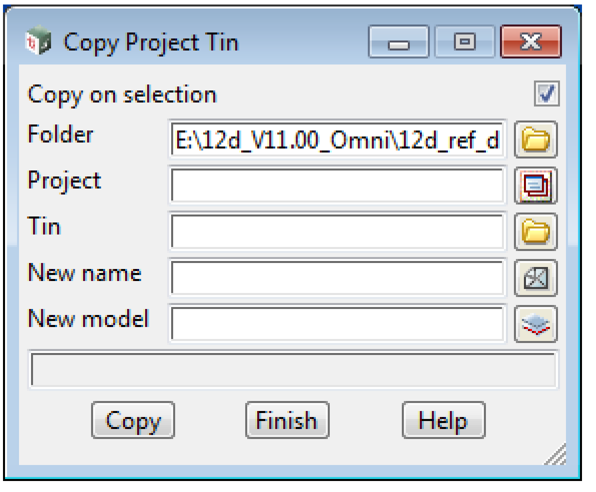
Copy	button		

after selecting this button, the tin given in the **old tin** *field will be copied and the copy given the name in the* **new tin** *field and placed in the model given in the* **model for tin** *field.*

16.14.3 Copy Project Tin

Position of option on menu: **Tins =>Utilities =>Copy project tin**

On selecting the **copy project tin** option, the **copy project tin** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Copy on selection	tick box		

if ticked, the project is copied as soon as it is selected. The same tin name and default model containing the tin is used.

Folder	folder box		working folder

name of the working folder containing the project from which the tin should be copied

Project	project box		current projects

*name of the project in the folder given in the **Folder** field, from which the tin should be copied.*

Tin	tin box		available tins

name of the tin to be copied.

New name	tin box		

if non-blank, the new name to be given to the copied tin which must be different to all the exiting tins names in the working project.
If blank, keep the name of the original tin (as long as it is different to the existing tins).

New model	model box		available models

if non-blank, the tin will be placed in this model.

Copy	button		

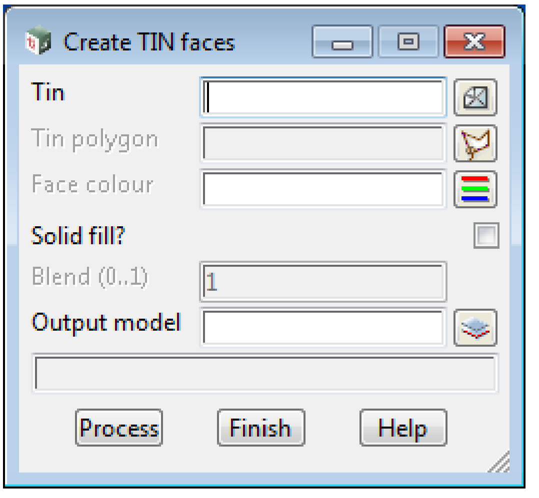
copy to this project, the tin given in the **Tin** *field from the project given in the* **Project** *field.*

16.14.4 Create Tin Faces

Position of option on menu: **Tins =>Utilities =>Create tin faces**

This options creates a separate face for each triangle in a tin.

Selecting **Create tin faces** displays the **Create Tin Faces** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin to have faces created for each triangle.

Tin polygon	polygon select		

if selected, faces are created fro triangles whose centroid is inside the selected polygon.

Face colour	colour box		

if blank, the tin triangle colour is used for the face colour.
If not blank, this colour is used for all faces.

Solid fill ?	tick box		

if ticked, the faces are solid filled with a blend value

Blend (0 ... 1)	real value		

The value of blending is between 0 and 1.

0 means the face is totally transparent (and hence invisible) and 1 means that the face is opaque (non-translucent) and can't be seen through at all.

If blank, then the value is taken as 1 and the polygon is opaque (non-translucent).

Output model	model box		available models

model for the faces

Process	button		

create faces for selected triangles.

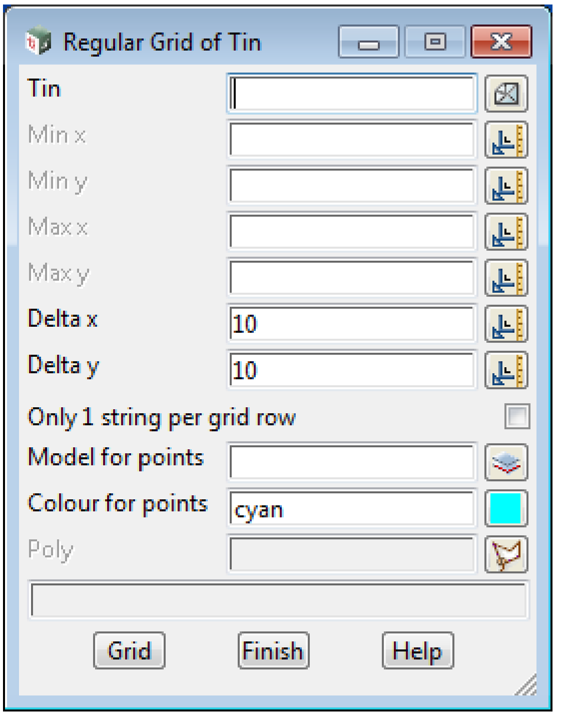
16.14.5 Grid DTM

Position of option on menu: **Tins =>Utilities =>Grid**

The grid option calculates heights at points at regular x and y increments between given minimum and maximum x and y values. If a polygon is given, a z-value is only calculated for points inside the polygon.

Also see the 16.14.7 DEMs options for creating a file of a regular grid to different formats (Arc View, Quantm, Sokkia).

On selecting the **Grid** option, the **Regular Grid of Tin** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin on which the spot heights will be calculated.

Min x/y	input		

minimum x/y values to calculate spot heights for.

Max x/y	input		

maximum x/y values to calculate spot heights for.

Delta x/y	input		

difference between the x/y-values for the grid.

Only 1 string per grid row	tick box		

if ticked, all the grid points for a row are joined in a point string.
If not ticked, individual one vertex strings are created for each grid point.

Model for points	model box		available models

if blank, no point strings will be created.
If non-blank, points strings of the spots heights will be created and stored in this model.

Colour for points colour box default colour available colours

colour to make the point strings.

Poly poly-select

if a polygon is selected, then the spot heights will be restricted to within the polygon.
If no polygon is selected, the spot heights will be calculated over the entire tin.

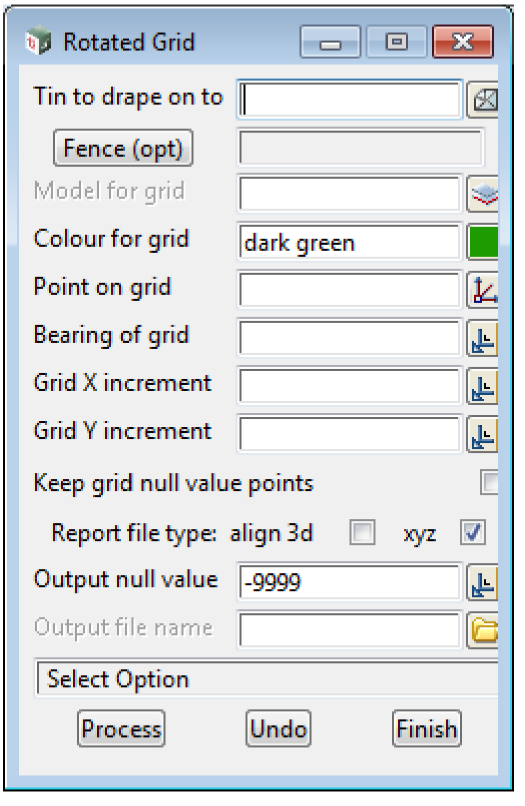
Grid button

calculate the spot heights at regular points over the area given by the min and maximum x and y values and restricted to be inside the polygon.

16.14.6 Rotated Grid

Position of option on menu: **Tins =>Utilities =>Rotated Grid**

This panel is used to create a grid of points with z-values from a given tin. The grid can be rotated and a polygon can be used to restrict the created points. A report of grid points can also be calculated in either simple x y z (one point per line) form or if no polygon is used, in a format accepted by Quantm (formerly Align 3d).



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Tin to drape on to	tin box		available tins

Tin to get the z-values from.

Fence (opt)	string select		

Optional – if selected, the grid is restricted to being inside the polygon.

Model for grid	model box		available models

Model for the created grid.

Colour for grid	colour box	dark green	available colours

Colour for the created grid

Point on grid	point select box		

The selected point is on the grid. The other grid points are integer multiples of the "Grid X increment" and "Grid Y increment" (with the bearing "Bearing of grid") from this point.

Bearing of grid	bearing box		

Bearing of the side of the grid.

Grid X increment input box

The grid is made up of points of integer multiple of "Delta X" in the X direction.

Grid Y increment input box

The grid is made up of points of integer multiple of "Delta Y" in the Y direction.

Keep grid null value points tick box

If ticked, the "Output null value" is written out for any null z-values.

Report file type radio button align 3d/xyz xyz

If xyz – the grid is written out as x y z with one point per line. The" Output null value" is used for any null z-values in the grid.

If align 3d – the grid is written out in a format suitable for align 3d (Quantm). The" Output null value" is used for any null z-values in the grid.

Output null value input box -9999

The z-value to write out if any grid z-value is null.

Output file name file box *.rpt files

Name of the output file for the grid data.

Process button

Run the option

Undo button

Undo the last grid created whilst the panel is up.

16.14.7 DEMs

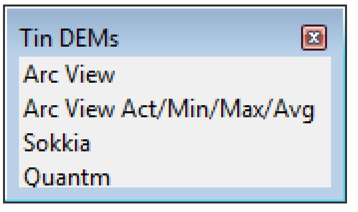
Position of menu: **Tins =>Utilities =>DEMs**

Position of menu: **File I/O =>Data output =>DEMs**

DEMS (Digital Elevation Models) can be created from a tin. DEMS are only regular arrays of z-values, rather than the random values and break lines supported by Tins.

Unfortunately there is not just one formats for DEMs.

The **DEMS** walk-right menu is



For the option *Arc View*, go to 16.14.7.1 Write Arc View DEM
Arc View Act/Min/Max/Avg 16.14.7.2 Write Arc View DEM (Actual, Min, Max, Av)
Sokkia 16.14.7.4 Sokkia
Quantm 16.14.7.3 Quantm

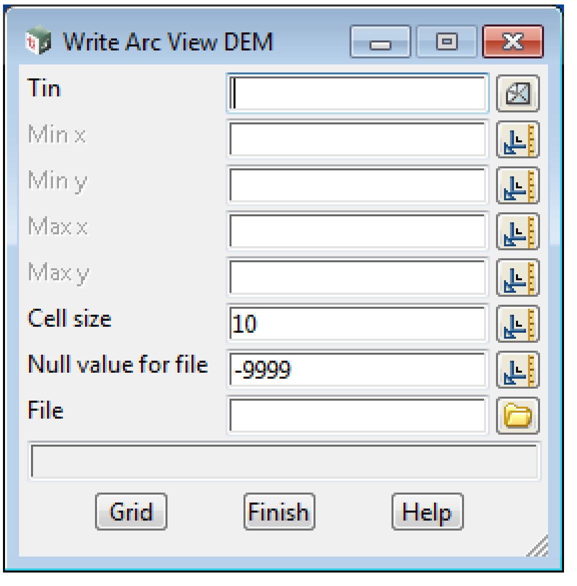
16.14.7.1 Write Arc View DEM

Position of option on menu: **Tins =>Utilities =>DEMs =>Arc View**

Position of menu: **File I/O =>Data output =>DEMs =>Arc View**

The **Arc View** option calculates heights at points at regular x and y increments (cell size) between given minimum and maximum x and y values and writes out the data in the Arc View DEM format.

On selecting the **Arc View** option, the **Write Arc View DEM** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin on which the DEM heights will be calculated.

Min x/y	input		

minimum x/y values to calculate spot heights for.

Max x/y	input		

maximum x/y values to calculate spot heights for.

Cell size	input		

size of the delta x and delta y - cell is a square.

Null value for file	input	-9999	

value to use when writing null z-values out to the file.

File	file box		*.grd

the file name for the spot height values in Arc View DEM format.

Grid	button		

calculate the spot heights at regular points over the area given by the min and maximum x and y values and write out the file.

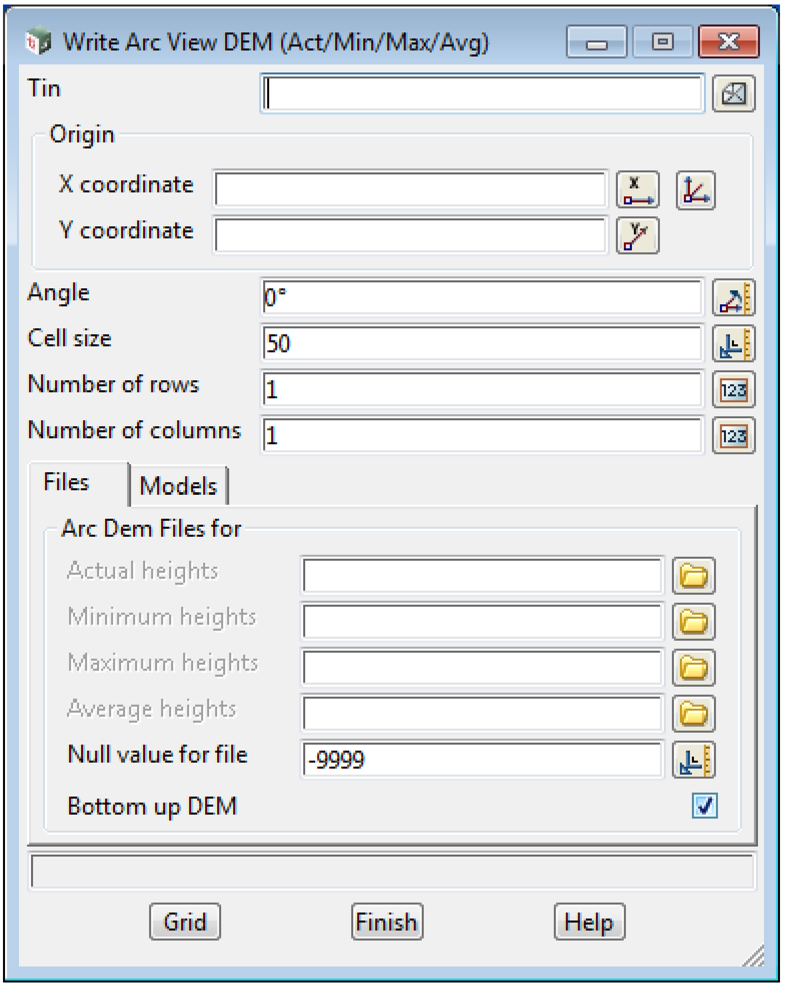
16.14.7.2 Write Arc View DEM (Actual, Min, Max, Av)

Position of option on menu: **Tins =>Utilities =>DEMs =>Arc View Act/Min/Max/Avg**

Position of menu: **File I/O =>Data output =>DEMs =>Arc View Act/Min/Max/Avg**

The **Arc View Act/Min/Max/Agg** option calculates heights at points at regular x and y increments (cell size) between given minimum and maximum x and y values, and also the minimum, maximum and average values for the cell. The data is written out in the Arc View DEM format.

On selecting the **Arc View Act/Min/Max/Avg** option, the **Write Arc View DEM (Act/Min/Max/Avg)** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin on which the DEM heights will be calculated.

X/Y coordinate	input		

X/Y coordinates for the origin of the DEM

Angle	angle box		

angle of rotation of the DEM

Cell size	input		

size of the delta x and delta y - cell is a square.

Number of rows/columns input 1

number of rows/columns in the DEM

Files tab - Arc DEM files for

Actual heights file box

if non blank, write the z-values for the tin at the middle of each cell, out to a dem file

Minimum heights file box

if non blank, write the minimum z-values for the tin for the each cell, out to a dem file

Maximum heights file box

if non blank, write the maximum z-values for the tin for the each cell, out to a dem file

Average heights file box

if non blank, write the average z-values for the tin for the each cell, out to a dem file

Null value for file input -9999

value to use when writing null z-values out to the DEM files.

Bottom up DEM tick box

if ticked, the DEM files are written from the bottom row of the grid and work up.
If not ticked, the DEM files are written from to top row of the grid and work down

Models tab - Models for

Actual heights model box

if non blank, create points of z-values for the tin at the middle of each cell

Minimum heights model box

if non blank, create points of minimum z-values for the tin for the each cell

Maximum heights model box

if non blank, create points of maximum z-values for the tin for the each cell

Average heights model box

if non blank, create points of average z-values for the tin for the each cell

Colour for points colour box

colour for the point strings

Grid button

calculate the spot heights at regular points over the area given by the min and maximum x and y values and write out the file.

16.14.7.3 Quantm

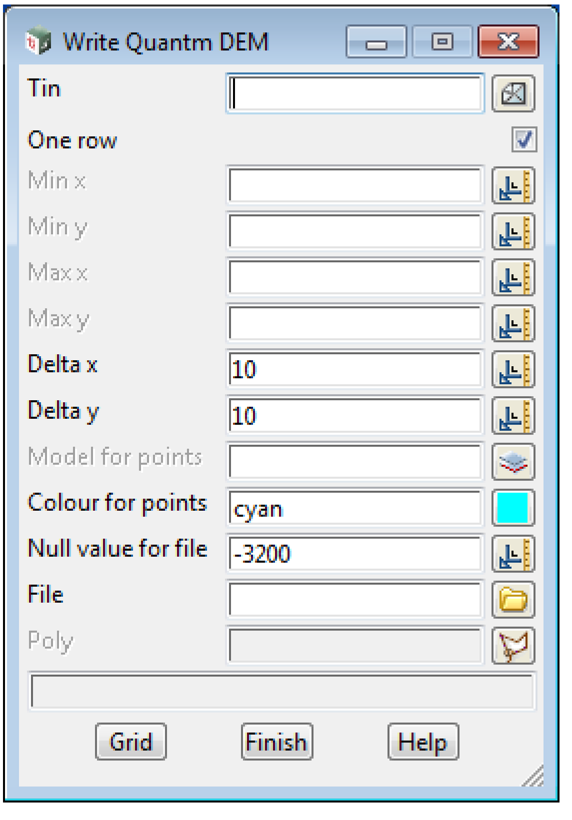
Position of option on menu: **Tins =>Utilities =>DEMs =>Quantm**

Position of menu: **File I/O =>Data output =>DEMs =>Quantm**

The **Quantm** option calculates heights at points at regular x and y increments between given minimum and maximum x and y values. If a polygon is given, a z-value is only calculated for points inside the polygon.

Actual point strings and/or a file of heights at the points in Quantm (formerly Align 3D) format can be generated. For Quantm no polygon should be used and if there is no z-value (no tin) at a point, then a null z-value is written to the output file.

On selecting the **Quantm** option, the **Write Quantm DEM** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin on which the spot heights will be calculated.

One row tick box

*If **ticked**, null points are added in for any missing (x,y) so that there is only one row of data (one string) for each x-value.*

Min x/y input

minimum x/y values to calculate spot heights for.

Max x/y input

maximum x/y values to calculate spot heights for.

Delta x/y input

difference between the x/y-values.

Model for points	model box		available models

if blank, no point strings will be created.
If non-blank, points strings of the spots heights will be created and stored in this model.

Colour for points	colour box	default colour	available colours

colour to make the point strings.

Null value for file	input	-3200	

value to use when writing null z-values out to the file.

File	file box		*.grd

If non-blank, a file of the spot height values in Quantm format is created.

Poly	poly-select

if a polygon is selected, then the spot heights will be restricted to within the polygon.
If no polygon is selected, the spot heights will be calculated over the entire tin.

Grid	button

calculate the spot heights at regular points over the area given by the min and maximum x and y values and restricted to be inside the polygon.

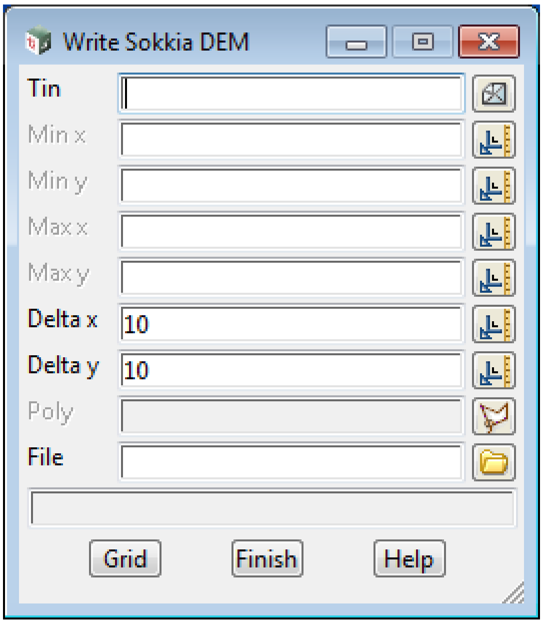
16.14.7.4 Sokkia

Position of option on menu: **Tins =>Utilities =>DEMs =>Sokkia**

Position of menu: **File I/O =>Data output =>DEMs =>Sokkia**

The **Sokkia** option calculates heights at points at regular x and y increments between given minimum and maximum x and y values. If a polygon is given, a z-value is only calculated for points inside the polygon. A file of heights at the points in Sokkia format is generated.

On selecting the **Sokkia** option, the **Write Sokkia DEM** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin on which the spot heights will be calculated.

Min x/y	input		

minimum x/y values to calculate spot heights for.

Max x/y	input		

maximum x/y values to calculate spot heights for.

Delta x/y	input		

difference between the x/y-values.

Poly	poly-select		

if a polygon is selected, then the spot heights will be restricted to within the polygon. If no polygon is selected, the spot heights will be calculated over the entire tin.

File	file box		*.grd

If non-blank, a file of the spot height values in Sokkia format is created.

Grid	button		

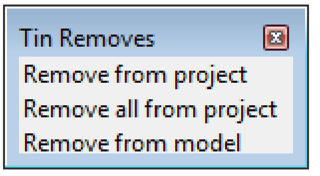
calculate and write a file of the spot heights at regular points over the area given by the min and maximum x and y values and restricted to be inside the polygon.

16.14.8 Remove

Position of menu: **Tins =>Utilities =>Remove**

Tins can be removed from models and from the project. Remove tins are not deleted but simply taken off the tin list in the project. Removed tins are left in the project area and can be added in again using the **add** option.

The **tin removes** walk-right menu is



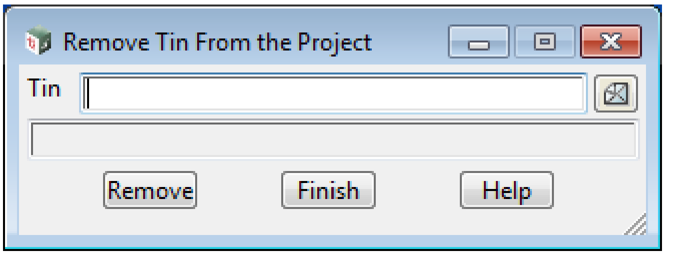
The options in the menu will now be described.

For the option Remove from project, go to	16.14.8.1 Remove From Project
Remove all from project	16.14.8.2 Remove All From Project
Remove from model	16.14.8.3 Remove From Model

16.14.8.1 Remove From Project

Position of option on menu: **Tins =>Utilities =>Remove =>Remove from project**

On selecting the **remove from project** option, the **remove tin from the project** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin to be removed from the working project.

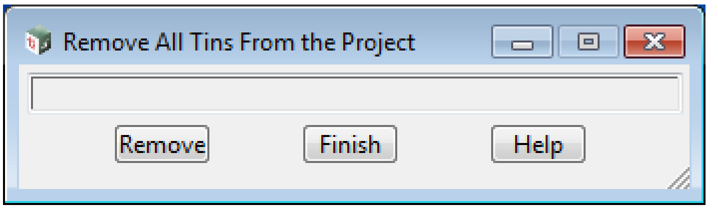
Remove	button		

remove the tin given in the **tin** *field from the project.*

16.14.8.2 Remove All From Project

Position of option on menu: **Tins =>Utilities =>Remove =>Remove all from project**

On selecting the **Remove all from project** option, the **remove all tins from project** panel is displayed.



The fields and buttons used in this panel have the following functions.

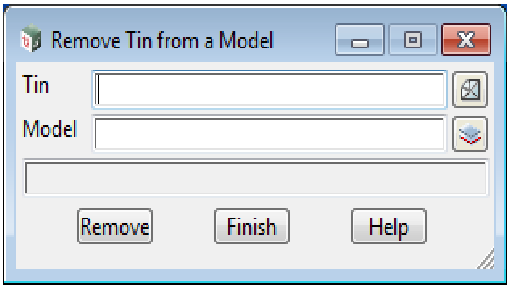
Field Description	Type	Defaults	Pop-Up
Remove	button		

remove all the tins from the working project. If no errors occur, the panel will be removed after the tins have been removed.

16.14.8.3 Remove From Model

Position of option on menu: **Tins =>Utilities =>Remove =>Remove from model**

On selecting the **remove from model** option, the **remove tin from a model** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin to be removed from a model.

Model	model box		available models

name of the model from which the tin given in the **tin** *field will be removed.*

Remove	button		

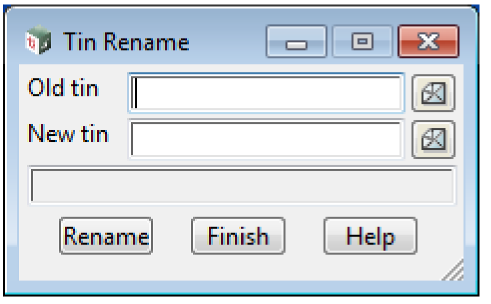
remove the tin given in the **tin** *field from the model given in the* **model** *field.*

16.14.9 Rename

Position of option on menu: **Tins =>Utilities =>Rename**

On selecting the **rename** option, the **tin rename** panel is displayed.

This panel can be used to change the names of existing tins



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Old tin	tin box		available tins

name of the tin to be renamed.

New tin	tin box		

new name for the tin

Rename	button		

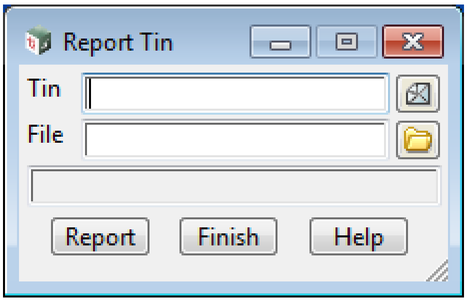
change the name of the tin in the **old tin** *field to the name given in the* **new tin** *field.*

16.14.10 Report

Position of option on menu: **Tins =>Utilities =>Report**

This panel produces a report on the tin including the models making up the tin and all the settings used when creating the tin.

On selecting the **Tin report** option, the **Report Tin** panel is displayed.



The fields and buttons used in this panel have the following functions.

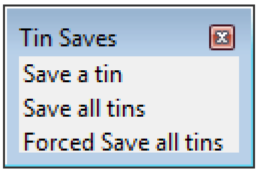
Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins or super tins
name of the tin to create report for.			
File	file box		*.rpt files
name of the report file			
Report	button		
create the tin report.			

16.14.11 Save

Position of menu: **Tins =>Utilities =>Save**

Tins are automatically saved on disk when they are first created but various operations can then be applied to tins which may not be immediately updated on disk.

The **tin saves** walk-right menu is

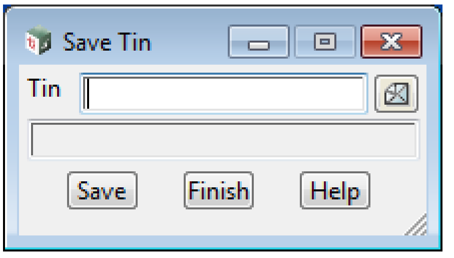


The options in the menu will now be described.

For the option Save a tin, go to	16.14.11.1 Save a Tin
Save all tins	16.14.11.2 Save/Forced Save All Tins
Forced Save all tins	16.14.11.2 Save/Forced Save All Tins

16.14.11.1 Save a Tin

On selecting the **save a tin** option, the **save tin** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin to be saved to disk.

Save	button		

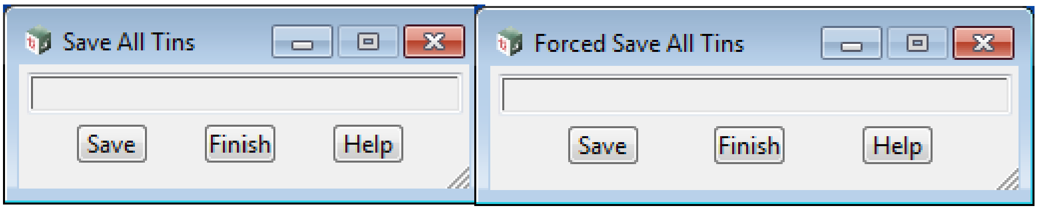
after selecting this button, the tin given in the **tin** *field will be saved to disk.*

16.14.11.2 Save/Forced Save All Tins

Position of option on menu: **Tins =>Utilities =>Save =>Save all tins**

Position of option on menu: **Tins =>Utilities =>Save =>Forced Save all tins**

On selecting the **save/force save all tins** option, the **save/forced save all tins** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Save	button		

for save all tins: after selecting this button, all tins in the working project that have been modified since their last save, will be saved to disk. Unless an error occurs, the panel will be removed after the saving is completed.

for force save all tins: after selecting this button, all tins in the working project will be saved to disk. Unless an error occurs, the panel will be removed after the saving is completed.

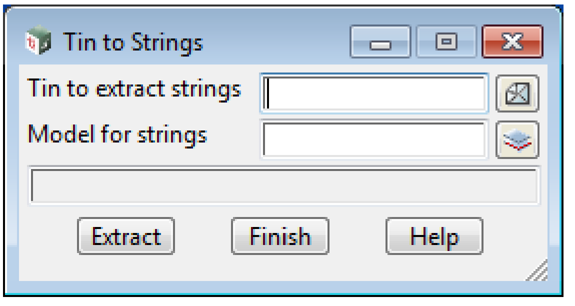
16.14.12 Tin to Strings

Position of option on menu: **Tins =>Utilities =>Tin to strings**

The points and breaklines of an existing tin can be extracted using the **Tin to strings** option.

Note that only the final points and breaklines used in creating the tin are extracted. Original string name, colours etc. are not obtainable.

Selecting **Tin to strings** displays the **Tin to Strings** panel:



The fields and buttons used in this panel have the following functions.

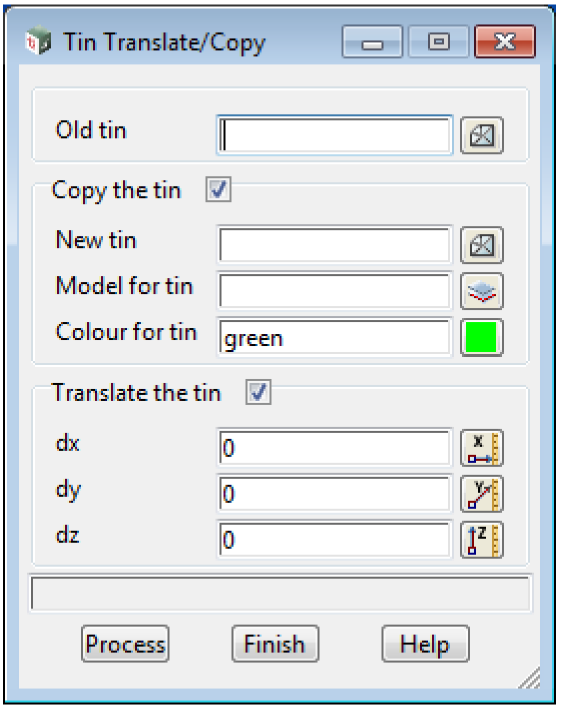
Field Description	Type	Defaults	Pop-Up
Tin to extract strings	tin box		available tins
name of the tin to extract points and breaklines from.			
Model for strings	input		available models
model to hold the extracted point and breakline strings.			
Extract	button		
extract the points and breaklines that make up a tin.			

16.14.13 Translate/Copy

Position of option on menu: **Tins =>Utilities =>Translate/copy**

An existing tin can be translated and/or copied using the **Translate/copy** option.

Selecting **Translate/copy**, displays the **Tin Translate/Copy** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Old tin	tin box		available tins

name of the tin to be copied/translated.

Copy the tin	tick box	tick	

if ticked, the **New tin**, **Model for Tin** *and* **Colour for tin** *fields are activated and supply the information for the new tin.*

New tin	tin box		available tins

name of the copy of the tin - this can't be the same as any existing tin in the project.

Model for tin	model box		available models

if non-blank, the name of the model for the new tin. If blank, the tin is not put in any model.

Colour for tin	colour box		available colours

if non-blank, the colour of the new tin. If blank, use the old tin colour

Translate the tin	tick box	tick	

if ticked, the **dx**, **dy** *and* **dz** *fields are activated and supply the information for translating the tin*

dx/dy/dz	real box	0	

the delta amounts to translate the tine by.

Process	button		

translate/copy the tin.

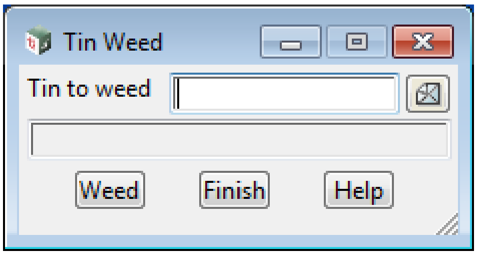
16.14.14 Weed

Position of option on menu: **Tins =>Utilities =>Weed**

When a tin is created, a copy of all the points and lines used in creating the tin is saved with the tin, including any duplicate points and lines.

The **weed** option will remove all the duplicate points and lines from the tin database.

On selecting the **Weed** option, the **tin weed** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin to weed	tin box		available tins

name of the tin to weed.

Weed	button		

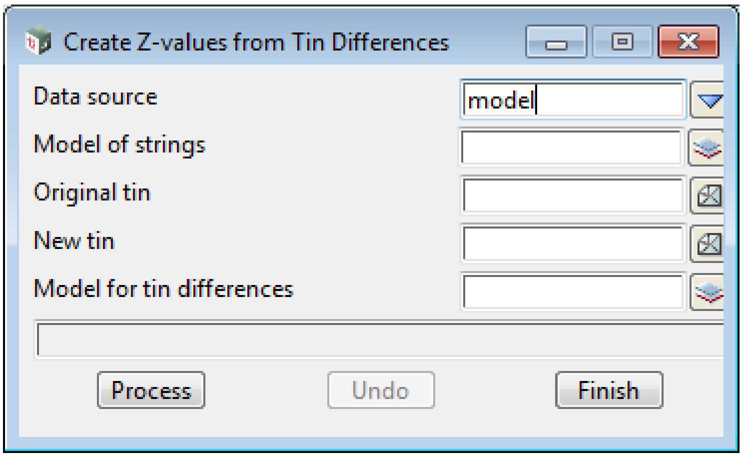
remove all duplicate points and lines from the tin database.

16.14.15 Z Differences from Tins

Position of option on menu: **Tins =>Utilities =>Z diffs from tins**

This option takes a data source and for each point in the data source, creates a new point whose z-value is the difference in z-values from two given tins at the plan position of the data source point.

On selecting the **Z diffs from Tins** option, the **Create Z-Values from Tin Differences** panel is displayed.



The fields and buttons used in this panel have the following functions:

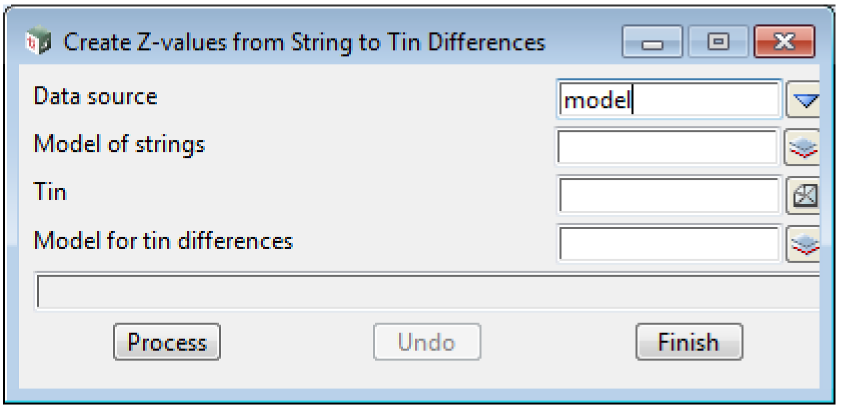
Field Description	Type	Defaults	Pop-Up
Data to settle			
Data source type	choice box	Model	model, string, view
data selection type			
Data source	input		
data to be processed.			
Original tin	tin box		available tins
name of the tin to get the z-value from.			
New tin	tin box		available tins
name of the tin to get the z-value from.			
Model for tin differences	model box		available models
model to place the strings containing the z values with the difference between the tins.			
Process	button		
for all strings in the data source, calculate the difference in the z-values from the two tins at the string vertices.			

16.14.16 Z Differences from String to Tin

Position of option on menu: **Tins =>Utilities =>Z diffs from string to tin**

This option takes a data source and for each string in the data source, creates a plan copy of the string and for each vertex in the new string, the z-value at a vertex is the difference in z-values from the original string vertex and the z-value of the tin at the (x,y) position of the vertex.

On selecting the **Z diffs from string to tin** option, the **Create Z-Values from String to Tin Differences** panel is displayed.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Data source type	choice box	Model	model, string, view

data selection type

Data source	input		

data to be processed.

Tin	tin box		available tins

name of the tin to get the z-value from.

Model for tin differences	model box		available models

model to place the strings containing the z values with the difference between the original string and the tin.

Process	button		

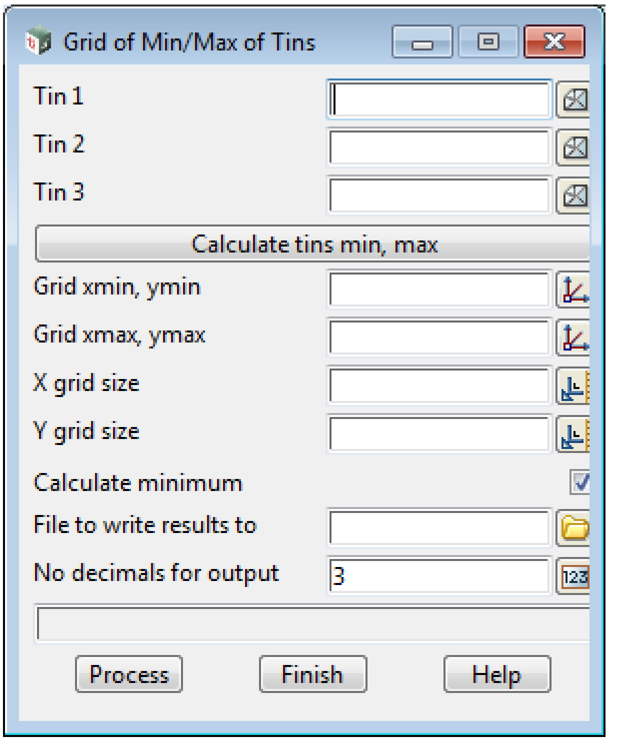
for all strings in the data source, calculate the difference in the z-values from the string to the tin.

16.14.17 Grid of Min / Max of Tins

Position of option on menu: **Tins =>Utilities => Min/ Max of Tins**

This option takes one to three tins and creates a regular grid of points where the z-value at each point is the minimum of the tins at that point.

Selecting **Min/ Max of Tins** brings up the **Grid of Min / Max of Tins** panel.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Tin 1, Tin 2, Tin 3	tin box		available tins

if non blank, this tin is used to use in the calculation to find the minimum of the tins at each grid point.

Calculate tins min, max button

when pressed, the minimum and maximum x and y values for the union of the three tins is calculated and the values written to the Grid xmin, ymin, Grid xmax, ymax fields.

Grid xmin, ymin input

the minimum x value, then one or more spaces, and then the minimum y values, to be used for creating the grid.

Grid xmax, ymax input

the maximum x value, then one or more spaces, and then the maximum y values, to be used for creating the grid.

x grid size input

the x coordinates for the grid points are created starting at the Grid xmin and then incremented by "x grid size" for as long as the x value is less than or equal to Grid xmax

y grid size input

the y coordinates for the grid points are created starting at the Grid ymin and then incremented by "y *grid size" for as long as the y value is less than or equal to Grid ymax*

Calculate minimum tick box ticked

if ticked, then grid points are created with the z-value being the minimum of tins Tin1, Tin 2, Tin3.

If not ticked, then grid points are created with the z-value being the maximum of tins Tin1, Tin 2, Tin3.

File to write results to File

name of the file to write the grid points out to. The format of the file is x, y and z, separated by spaces and with one point per line. The "File to write results to" must be non-blank.

No decimals for output input 3

number of decimal places to use for the x, y and z values

Process button

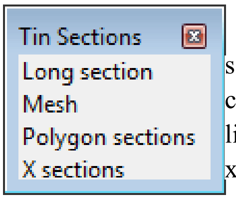
when this button is pressed, the grid of either the tin minimums or maximums is created and written to the file

16.15 Sections

Position of menu: **Tins =>Sections**

The **triangles=>sections** menu contains options involving sections through the tin.

The **sections** walk-right menu is



section along a string
create sets of sections through tin at right angles
limit sections to a polygon
x-sections along a string

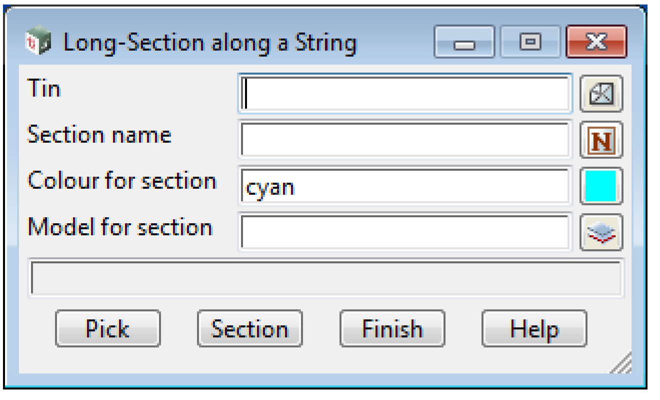
For the option *Long section*, go to 16.15.1 Long Sections.
Mesh, 16.15.2 Mesh
Polygon sections 16.15.3 Polygon Sections
X-sections 16.15.4 X-Sections

16.15.1 Long Sections

Position of option on menu: **Tins =>Sections =>Long Section**

The long section option calculates the section through a tin along a particular string. This option is rarely used because tins are automatically profiled through on the section view and in long section and x-section plots.

On selecting the **long section** option, the **long section along a string** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin that the string will be sectioned against.

Section name	input		

name to give the long section string

Colour for section	colour box	default colour	available colours

colour to make the long section string

Model for section	model box		available models

name of the model to contain the long section string.

Pick	string-select		

select he string to be sectioned along.

Section	button		

calculate the long section along the string selected by the **pick** *button, against the tin given in the tin field. The resulting section string is added to the model given in the* **Model for Section** *field.*

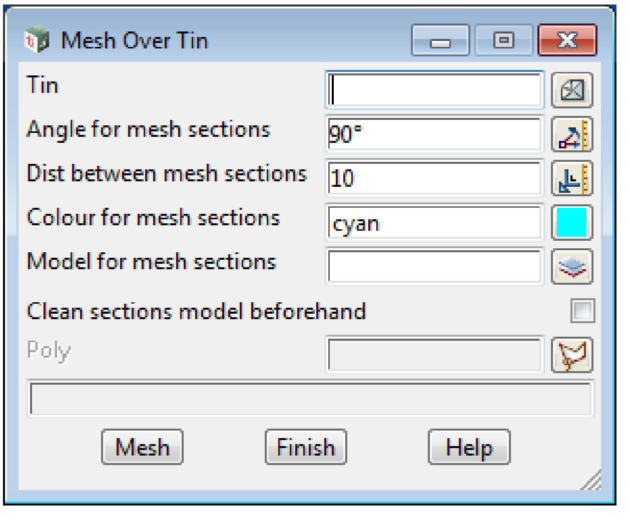
16.15.2 Mesh

Position of option on menu: **Tins =>Sections =>Mesh**

A 12d Model mesh is simply a series of lines cutting through the tin (sections) at a given angle and spacing, and at right angles to the given angle with the same spacing.

A mesh is an effective method of visualizing a triangulation when displayed in a perspective view and for hidden views.

On selecting the **mesh** option, the **mesh over tin** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin that the mesh sections will be calculated through.

Angle for mesh sections	angle box	90.0	0,45,90

angle (in degrees) of the lines to section along. The sections are also created at rights angles to this angle as well.

Dist between mesh sections	input	10.0	1,10,100

distance between the lines to section along

Colour for mesh sections	colour box	default colour	available colours

colour to make the calculated sections

Model for mesh sections	model box		available models

name of the model to contain the calculated sections.

Clean sections model beforehand tick box

if ticked, the model of sections is cleaned out before the option runs.

Poly polygon select

if a polygon is selected, then the sections are restricted to remaining within the polygon.

Mesh button

two sets of sections through the tin given in the **tin** *field are created within the selected polygon, or if no polygon is selected, the bounding polygon for the tin. One set of sections are calculated along straight lines at the angle given by the* **angle** *field and at a separation given by the* **dist** *field and the second set at right angles to the first set.*

<esc> can be used to abort the mesh option.

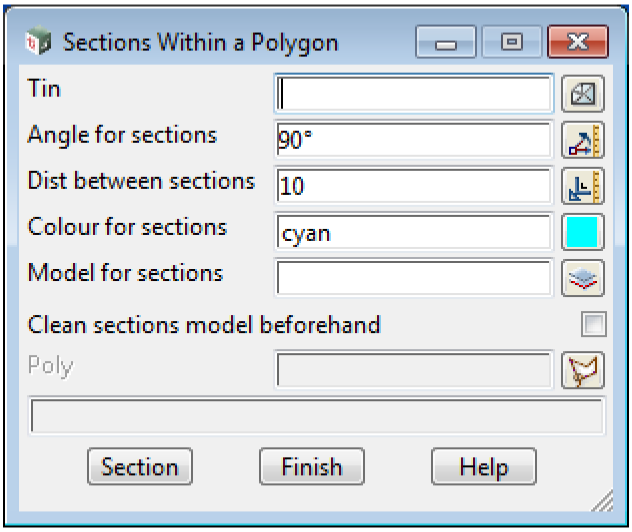
16.15.3 Polygon Sections

Position of option on menu: **Tins =>Sections =>Polygon sections**

In the **polygon sections** option, sections are calculated for a series of parallel lines covering the polygon, or if no polygon is given, the tin. The angle of the parallel lines and the distance between them are set by the user.

This option provides a quick method for producing a one directional set of sections covering a polygonal region on a tin.

On selecting the **polygon sections** option, the **sections within a polygon** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins
name of the tin that the sections will be calculated through			
Angle for sections	angle box	90.0	0,45,90
angle (in degrees) of the lines to section along			
Dist between sections	input	10.0	1,10,100
distance between the lines to section along			
Colour for sections	colour box	default colour	available colours
colour to make the calculated sections			
Model for section	model box		available models
name of the model to contain the calculated sections.			
Clean sections model beforehand	tick box		
if ticked, the model of sections is cleaned out before the option runs.			
Poly	polygon-select		
*A polygon is selected from a view. If **no polygon** is selected, sections are created over the entire tin.*			
Section	button		

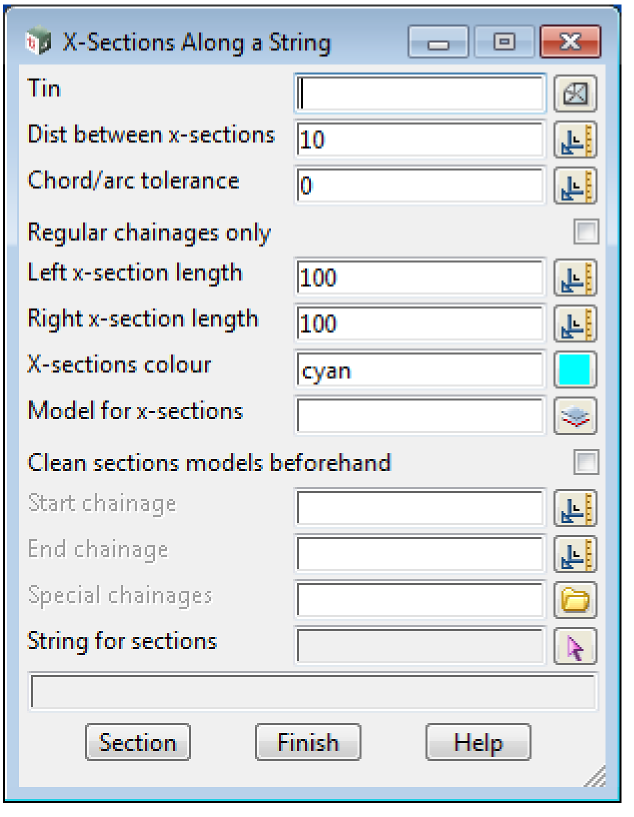
Sections within the polygon selected by **poly** *will be calculated along parallel straight lines at the angle given by the* **angle** *field and at a separation given by the* **dist** *field. The sections are made against the tin given by the Tin field. The sections colour is that given in the* **colour** *field and the section strings* *are placed in the model given in the* **model** *field.*

16.15.4 X-Sections

Position of option on menu: **Tins =>Sections =>X Sections**

The X-sections (cross sections) option calculates cross sections at a given interval along a string.

On selecting the **x-section** option, the **x-sections along a string** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin that the x-sections will be calculated through.

Dist between x-sections	input	10	

the distance between points on the selected string where the x-sections are calculated.

Regular chainages only	tick box		

if ***ticked****, sections are only calculated at points at the given chainage distance apart.*
If ***not ticked****, extra sections are created at horizontal tangent points and horizontal IPs.*

Left x-section length	input	100	

the plan distance for the left side of the x-section - that is, the distance that the x-section extends to the left of the selected string.

Right x-section length	input	100	

the plan distance for the right side of the x-section - that is, the distance that the x-section extends to the right of the selected string.

X-sections colour	colour box	default colour	available colours

colour to make the x- sections

Model for x-sections model box available models

name of the model to contain the x-sections.

Start/End chainage input

the string chainage to start/end the x-sections. If blank, the **start/end chainage** *is taken to be the chainage at the beginning/finish of the string.*

Special chainages file box *.spc files

if non blank, a file of special chainages to also create x-sections at

String for sections string-select

select the string to calculate the x-sections along.

Section button

calculate the x-sections along the string selected by the **pick** *button, against the tin given in the tin field. The resulting x-section strings are added to the model given in the* **model for x-sections** *field.*

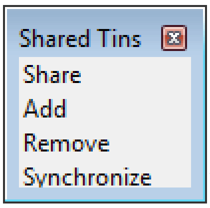
16.16 Sharing

Position of menu: **Tins =>Sharing**

Sharing allows tins from a project (the server project) to be added to other projects (client projects).

Before any tins can be added to a client project, they must first be tagged in the server project as allowed to be shared.

The **Sharing** walk-right menu is



allow tins in this project to used by other projects
add a shared tin from another project
remove a tin shared from another project
updated tins shared from other projects

For the option *Share*, go to 16.16.1 Share Tins
Add 16.16.2 Add Shared Tins
Remove 16.16.3 Remove Shared Tins
Synchronize 16.16.4 Synchronize Shared Tins

16.16.1 Share Tins

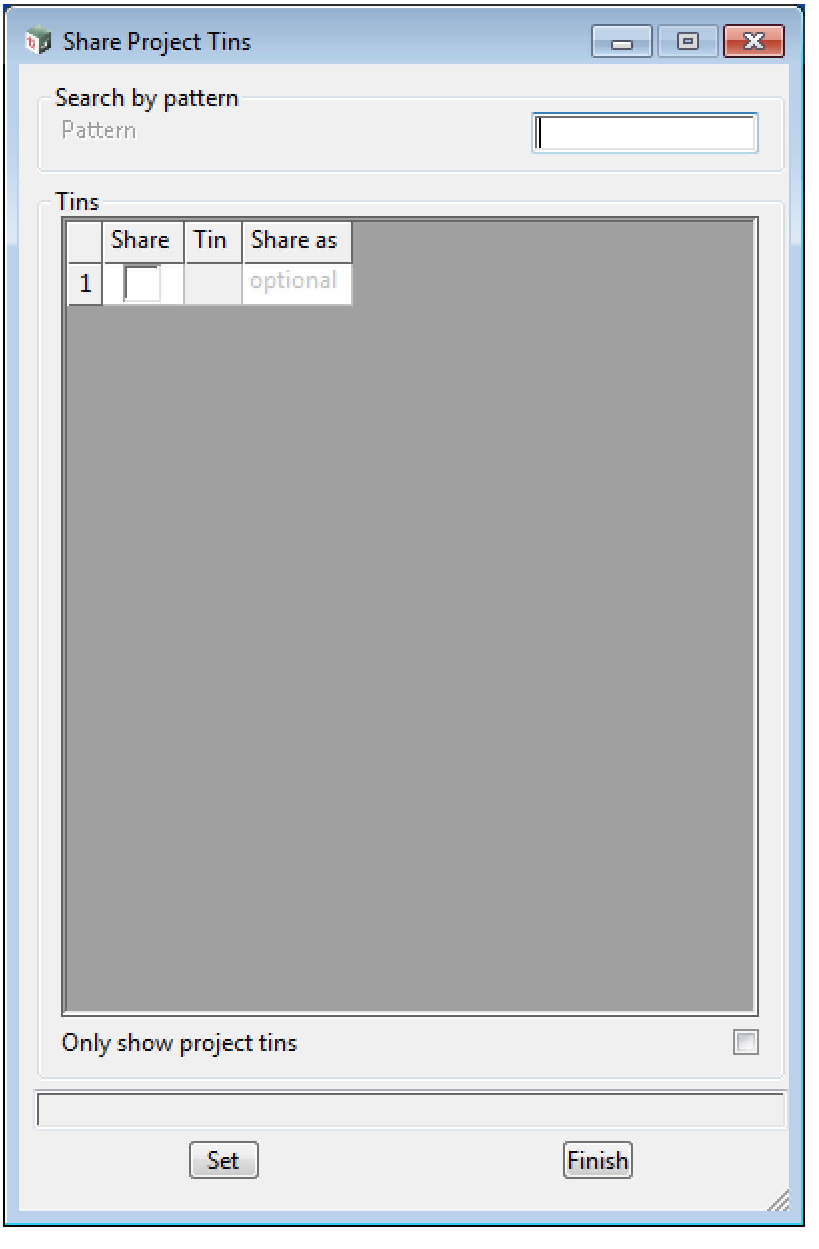
Position of option on menu: **Tins =>Sharing =>Share**

Before tins in a server project can be added to client projects, the tins must be made available for sharing in the server project by using the **Share** option.

All tins that have been made available for sharing are displayed in tin lists in a colour defined by an environment variable. The default colour is a yellow (RGB of (255,166,0)).

This option also can reverse the process. That is, remove the availability of a tin for sharing. If this is done then clients who have shared this tin will be warned.when starting up their project and/or synchronizing occurs.

Selecting **Share** displays the **Share Project Tins** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Default Pop-Up

All the models in the project are listed in the **Share Project Tins** *panel and if a tin has been tagged for Sharing, a ticked will be displayed in the* ***Share*** *column of the grid.*

Pattern

if a pattern is typed then all the tins matching the pattern will have a tick placed in the ***Share*** *column. If * is typed then all tin are ticked for sharing.*

Share grid column

tick if the tin is to be made available for sharing when the **Set** *button is pressed.*

All tins that have been made available for sharing are displayed in tin lists in a colour defined by an environment variable. The default colour is a yellow (RGB of (255,166,0)).

Clicking RB on ***Share*** *at the top of the column brings up a menu to* ***Toggle*** *the ticks,* ***Set*** *all the ticks on,* ***Clear*** *to turn all the ticks off.*

Tin grid column

this column lists all the tins in the project

Share as grid column

an optional name with which the tin will be shared out to clients. Leave it blank to share under the original name.

Only show project tins tick box

if ticked, only tins originating from the current project are shown.

Set button

clicking ***Set*** *marks all the tins with a tick in the* ***Tick*** *column as being available for sharing. Those tins without a tick will not be available for sharing.*

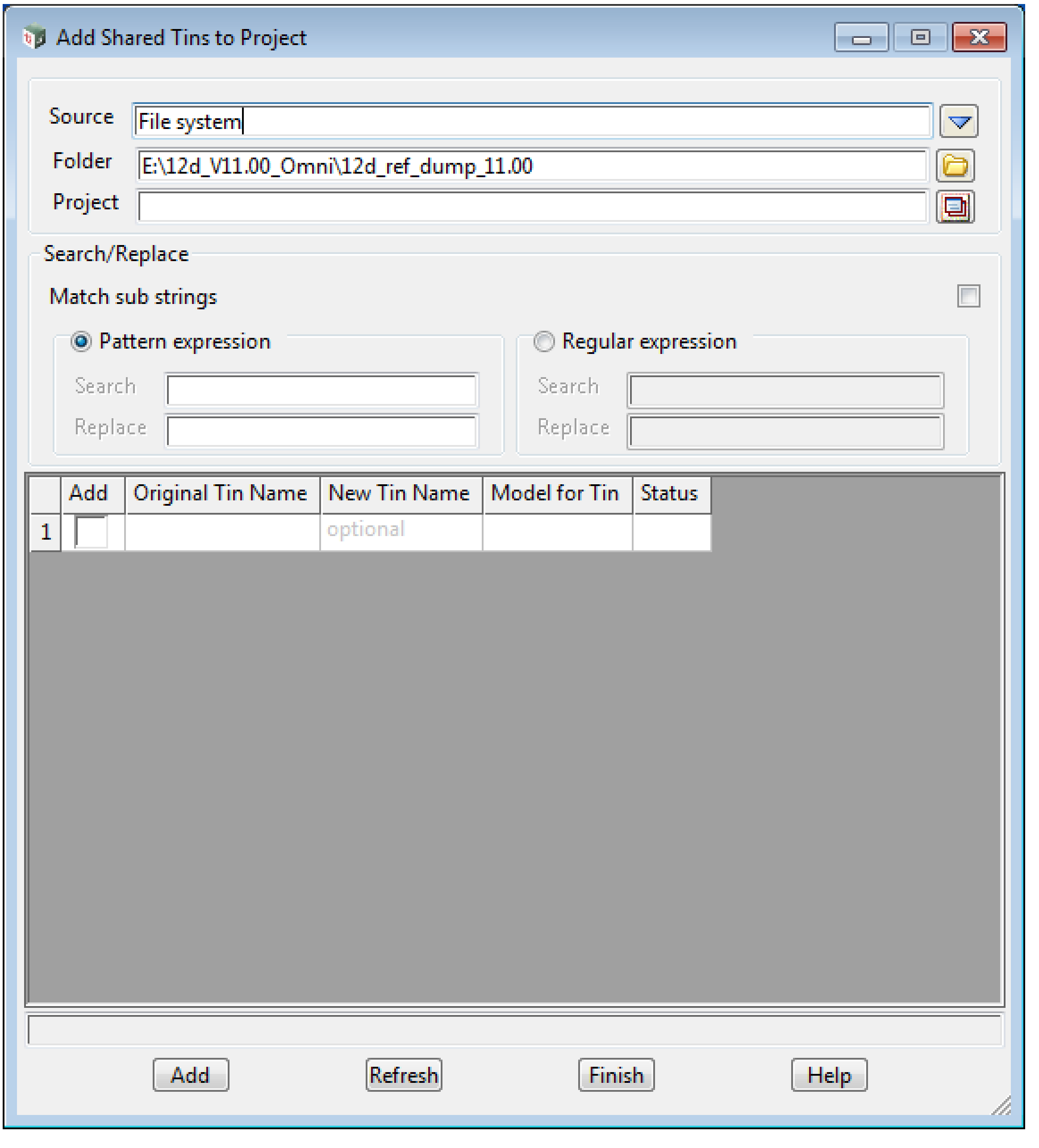
16.16.2 Add Shared Tins

Position of option on menu: **Tins =>Sharing =>Add**

The **Add** option is used to add shared tins from a server project, to this project (a client project).

All tins that have been added as shared tins are displayed in tin lists in a colour defined by an environment variable. The default colour is blue.

Selecting **Add** displays the **Add Shared Tins to Project** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Folder	folder box		select folder

*folder to look for **12d Model** projects. When a folder is selected, all the **12d Model** projects in the folder will be displayed in a **Projects** pop-up list.*

Project folder box select folder

name of the ***12d Model*** *project in the folder given in the* **Folder** *field, to search for tins marked for sharing. Once a project is selected, all the tins marked for sharing from that project will be listed in the* Original Tin Name *column.*

Search/Replace

section for renaming tins from the selected server project

Match sub strings tick box

if ***ticked****, the* **Search** *expression is used to match against part of each tin name.*

If ***not ticked****, the* **Search** *expression is used to match against the entire tin name.*

Pattern expression radio button

if set on, then ***Pattern*** *expressions given in the* **Search** *and* **Replace** *fields are used to modify tin names. Pattern expressions include the standard wild card * and wild character !.*

Search input

pattern to search for in the tin names. For example " exist" will select all tins with a name ending with " exist"*

Replace input

replacement for the search pattern found in the tin name. For example, "exist " in the Replace *field, takes the matched part of the tin name and adds "exist " to the front of it.*

Hence the Search pattern " exist" and Replace pattern "exist *" finds all tin with names ending in " exist" and renames them with the name starting with "exist " (and the " exist" at the end of the name is dropped off).*

Regular expression radio button

if set on, then ***Regular*** *expressions given in the* **Search** *and* **Replace** *fields are used to modify tin names.*

Search input

regular expression to search for in the tin names.

Replace input

replacement for the search expression found in the tin name.

Tin Names Grid

The tins available for sharing in the selected project are shown in the Original Tin Name *column. Any renaming by the Search and Replace expressions are shown in the* New Tin Name *column. Note that if the tin already exists in the current project, then the cell for that model in* Original Tin Name *column will be displayed in yellow.*

Add tick boxes in grid column

if ticked, the tin will be added to the project when the ***Add*** *button is pressed.*
If not ticked, the tin will not be added to this project as a shared tin.

Clicking RB on ***Add*** *at the top of the column brings up a menu to* ***Clear*** *which turns all the ticks off.*

Note - a shared tin that has been previously added can only be removed by using the **Tins =>Sharing =>Remove** *option.*

Original Tin Name column

this column lists all the tins in the server project that are available for adding to this client project.

New Tin Name column

if ***non blank****, the name to use instead of the* Original Tin Name *when the tin is added t the project.*

*If **blank**, the* Original Tin Name *is used for the tin when it is added to the project.*

The New Tin Names *can be from applying the* **Search** *and* **Replace,** *or by just typing them in.*

Model for Tin grid column

*if **non blank**, the model name to use in the client project for this tin added to the project*

If blank, the model name is the same as that in the server project.

Status output column

displays if their has been a match *or* no match *for the search/replace for renaming tins*

Add button

add to this project (a client project) as a shared tin, the ticked tins given in the Original Tin Name *field from the project given in the* **Project** *field, and add them to the model in the* Model for Tin field*.*

Refresh button

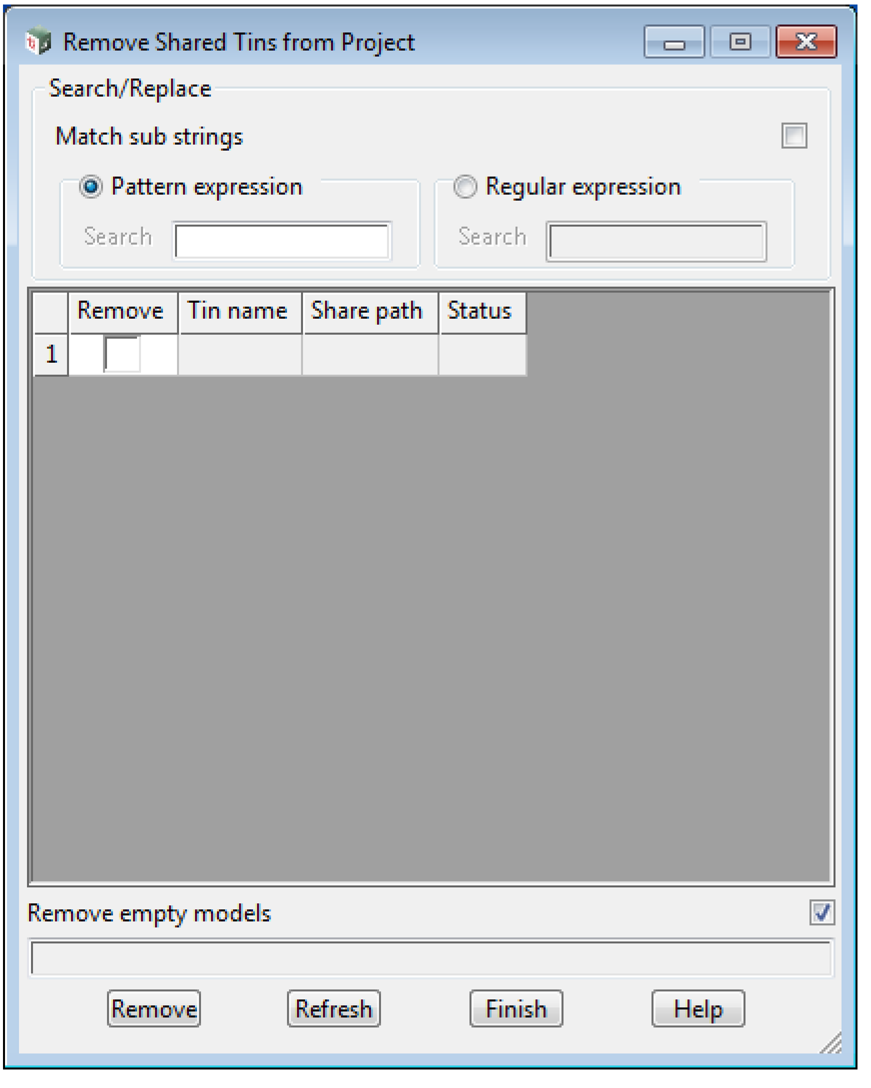
clicking Refresh *refreshes the list of all tins available for sharing in the selected* ***12d Model*** *project (server project).*

16.16.3 Remove Shared Tins

Position of option on menu: **Tins =>Sharing =>Remove**

The **Remove** option is used to remove shared tins from the.project. The shared tins would have been previously added to the project with the **Tins =>Sharing =>Add** option.

Selecting **Remove** displays the **Remove Shared Tins from Project** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Search/Replace

section for selecting by expressions, tins shared from other projects

Match sub strings tick box

*if **ticked**, the* **Search** *expression is used to match against part of each tin name.*

*If **not ticked**, the* **Search** *expression is used to match against the entire tin name.*

Pattern expression radio button

*if set on, then **Pattern** expressions given in the* **Search** *field are used to select tin names. Pattern expressions include the standard wild card * and wild character !.*

Search input

pattern to search for in the tin names. For example " exist" will select all tins with a name ending with " exist"*

Regular expression radio button

*if set on, then **Regular** expressions given in the* **Search** *field are used to select shared tin names.*

Search input

regular expression to search for in the shared tin names.

Tin Names Grid

list of all shared tins added to the project.

Remove tick boxes in grid column

*if ticked, the tin will be removed from this project as a shared tin when the **Remove** button is pressed.*

*Clicking RB on **Remove** at the top of the column brings up a menu to **Toggle** the ticks, **Set** all the ticks on, **Clear** to turn all the ticks off.*

Tin Name grid column

this column lists all the tins in this project added as shared tins from other server projects

Share path

this column lists the path name to the server project and the original tin name in the server project

Status grid column

displays if their has been a match *or* no match *for the* Search *for selecting shared tins*

Remove button

*clicking **Remove** removes as shared tins form this project, all the tins with a tick.*

Refresh button

*clicking **Refresh** refreshes the list of all shared tins previously added to the project*

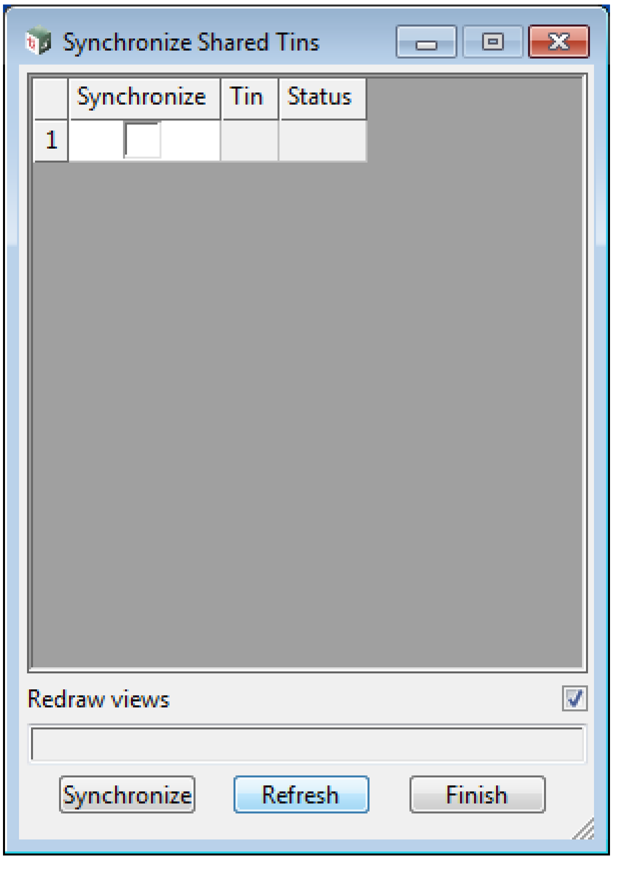
16.16.4 Synchronize Shared Tins

Position of option on menu: **Tins =>Sharing =>Synchronize**

The **Synchronize** option is used to updated any added shared tins. The shared tins would have been previously added to the project with the **Tins =>Sharing =>Add** option.

Three environment variables control the synchronization of updates of added shared models and tins. For more information on synchronizing, go to the section 7.6.13.4 Project Share Settings.

Selecting **Synchronize** displays the **Synchronize Shared Tins** panel.



The fields and buttons used in this panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Synchronize tick boxes in grid column

*if ticked and the tin has changed in the server project, then it will be re-copied from the server project when the **Synchronize** button is pressed.*

*Clicking RB on **Copy** at the top of the column brings up a menu to **Toggle** the ticks, **Set** all the ticks on, **Clear** to turn all the ticks off.*

Tin grid column

this column lists all the tins in this project that have been added from other (server) projects

Synchronize button

*clicking **Synchronize** re-copies any tins with a tick from the server projects*

Status

*Status will either be **Ok** or **No Longer Shared** to reflect the status of the tin on that row.*

Redraw views tick box

Whether or not to redraw all the views any of the synchronized tins are on.

Refresh button

*clicking **Refresh** refreshes the list of all shared tins previously added to the project*

16.17 Delete

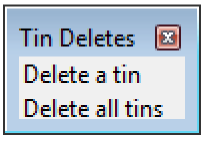
Position of menu: **Tins =>Delete**

The **delete** option is used to delete tins from the project and from the computer disk so that they no longer can be accessed or take up disk space.

To help protect against disasters, a **yes-no** pop-up menu is used to confirm that the user did intend deleting the tin. If deletion is confirmed, the selected tin is removed from the project (if its in the project) and deleted from the disk.

If a **12d** ***Model*** *trash bin* is being used, the deleted tins may be moved to the trash bin rather than deleted from disk (see 7.6.15 Trash Bin).

The **tin Deletes** walk-right menu is



For the option *Delete a tin*, go to 16.17.1 Delete a Tin
Delete all tins 16.17.2 Delete All Tins

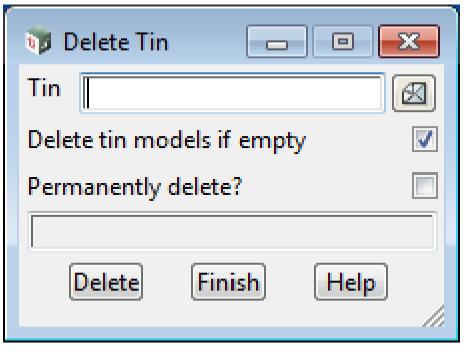
16.17.1 Delete a Tin

Position of option on menu: **Tins =>Delete =>Delete a tin**

The **delete a tin** option can be used to delete a tin in the project.

If a **12d** ***Model*** *trash bin* is being used, the deleted tins may be moved to the trash bin rather than deleted from disk (see 7.6.15 Trash Bin).

On selecting the **delete a tin** option, the **delete tin** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	input		project tins

name of the tin to be deleted from the computer disk.

Delete tin models if empty tick box

if ticked then the models containing the tin are also deleted if they are empty after the tin is deleted.

Permanently delete tick box

if tick, the deleted tins will not go to the trash bin but will be permanently deleted from disk.

Delete button

after selecting this button, the tin given in the tin field will be deleted from the computer disk. A **yes-no** *pop-up is used to confirm that deletion is required.*

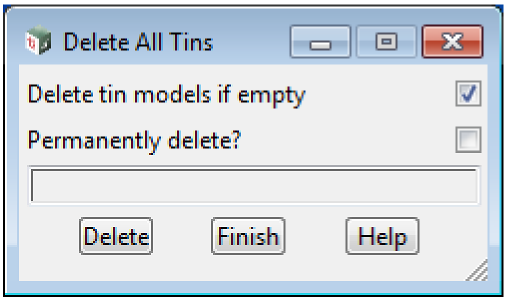
16.17.2 Delete All Tins

Position of option on menu: **Tins =>Delete =>Delete all tins**

The **delete all** option will delete all tins in the working project. It does not delete tins that are in the working project area but not yet added to the project.

If a **12d** ***Model*** *trash bin* is being used, the deleted tins may be moved to the trash bin rather than deleted from disk (see 7.6.15 Trash Bin).

On selecting the **delete all** option, the **delete all tins** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Delete tin models if empty tick box

if ticked then the models containing the tins are also deleted if they are empty after the tins are deleted.

Permanently delete tick box

if ticked, the deleted tins will not go to the trash bin but will be permanently deleted from disk.

Delete button

after selecting this button, a **yes-no** *pop-up is used to confirm that deletion is required. If it is, all tins in the working project will be deleted from disk. Unless an error occurs, the panel will be removed.*

17 Survey

Position of menu: **Survey**

The **Survey** walk-right menu is laid out to reflect the normal sequence of operations when handling field survey data.

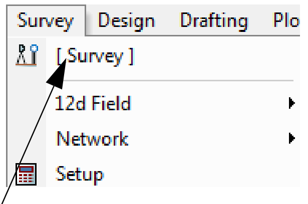
The description of the options on the **Survey** menu assumes that the reader is familiar with surveying, the 12d Model method of field coding for their particular data collector, data collector definitions and the 12d Model field file.

A description of the 12d Model field coding and data collector definitions is given in the Appendix 36 12d Survey Guide.

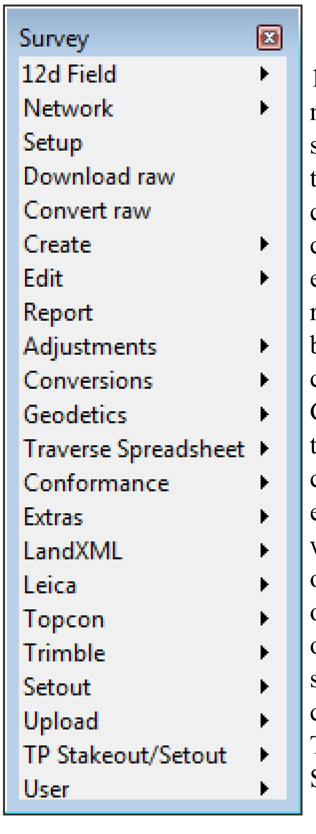
.

on Main menu

on 12d Model menu and floating Survey menu



create floating **Survey** menu



12d Field for Total stations and GNSS
networks incuding lsa, ePlan
select data collector (DC)
transfer data from DC to 12d field file
convert raw DC data to 12d field file
create 12d Survey function
edit 12d Survey function - text & graphically
report on 12d Survey function
bowditch, least squares etc.
conversion functions
Geodetic measure functions
traverse spreadsheet functions
conformance reporting options
extra survey functions - bd entry, etc.
write out x-sections in special LandXML format
outputs for Leica instruments
outputs for Topcon instruments
outputs for Trimble instruments
setout macros
create upload file & upload to data collectors
TP Stakeout/Setout macros
Survey User menu

See

18.2 12d Field - this is a separate chapter on ***12d Field***.

17.2 Network
17.3 Setup
17.4 Download Raw
17.5 Convert Raw
17.6 Create Survey Function
17.7 Edit Survey Function
17.8 Report
17.9 Adjustments

17.1 12d Field

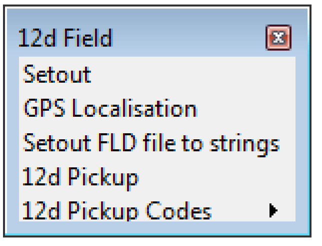
Position of menu: **Survey =>12d Field**

The **12d Field** module is for communicating directory with survey grade GPS units and Total Station theodolites through radio, blue tooth, Wi-Fi and cable. Survey data is displayed live on screen in conjunction with full road project design and existing surface data.

Because of its specialist nature, the documentation of the *12d Field* module is in a separate chapter and also in the ***12d Field Training*** manual.

The **12d Field** walk-right menu is:

.

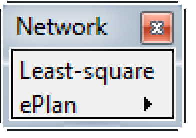


See 18 12d Field .- this is a separate chapter on ***12d Field***.

17.2 Network

Position of option on menu: **Survey =>Network**

Selecting **Network** brings up the **Network** menu



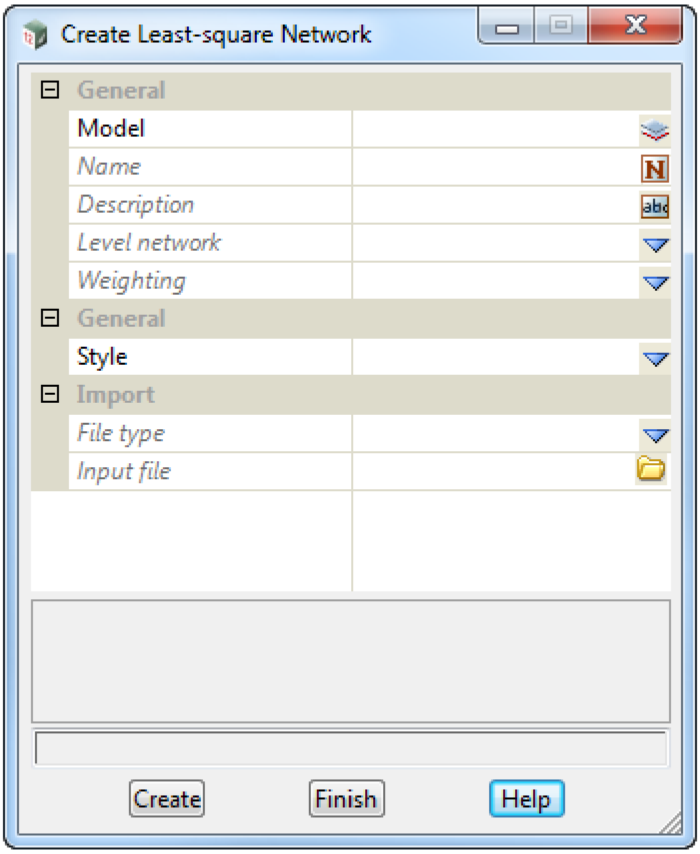
See

17.2.1 Creating a Least-Square Network
17.2.2 ePlan - this refers to a separate chapter on ***ePlan***.

17.2.1 Creating a Least-Square Network

Position of menu: **Survey =>Network =>Least-square**

Selecting **Least-square** brings up the **Create Least-square Network** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model	model box		available models
the model			
Name	name box		
Description	input		
Level network	choice box		
Weighting	choice box		
Style	choice box		
File type	choice box		
Input file	file box		
Create	button		

17.2.2 ePlan

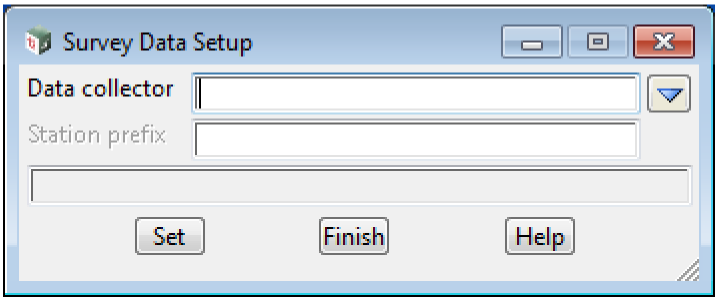
Position of menu: **Survey =>Network =>ePlan**

See 19 ePlan .- this is a separate chapter on **ePlan**.

17.3 Setup

Position of option on menu: **Survey =>Setup**

Selecting **Setup** brings up the **Survey data setup** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data collector	input		available data collectors

name of the data collector definition used to define how to read and interpret the raw survey data file. The list of available data collectors definitions is given in the file pointed to by the environment variable DATA_COLLECTORS_4D.

A data collector definition can be created/edited by
*(a) clicking on the choice button at the end of the **Data collector** panel field and selecting [Edit] at the bottom of the pop-up list of defined data collectors to bring up the Survey.4d Edit/Create panel.*

*(b) using **Project=>Browse=>Survey data collectors** the create/edit the Survey.4d file.*
Please see the 36.9 Data Collector Definitions *of Appendix* 36 12d Survey Guide *for more information on setting up a data collector definition.*

Station prefix	input		

*if **non-blank**, the prefix to be used for any text given for new instrument stations.*

Set	button		

store the selected data collector as being the currently selected one.

Note: Selecting *[Edit]* at the bottom of the pop-up list of data collectors will bring up the **Survey.4d Create/Edit** panel.

17.4 Download Raw

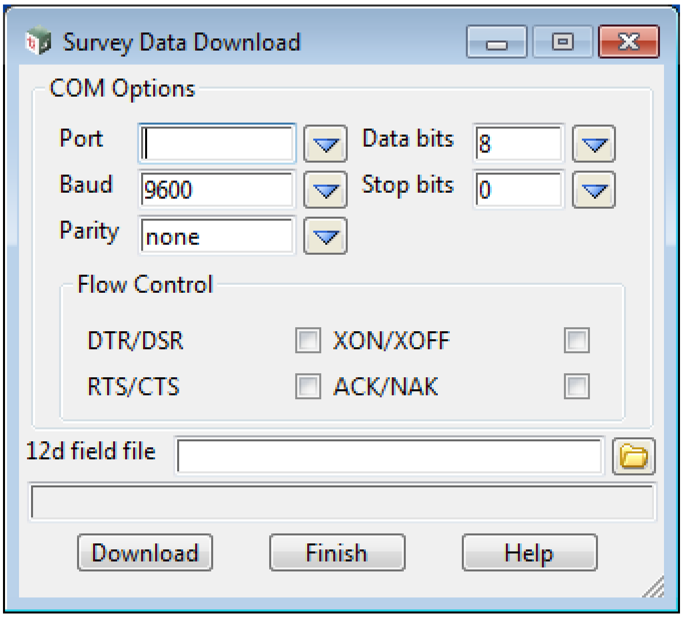
Position of option on menu: **Survey =>Download raw**

The **download raw** option will read data from the data collector connected to the computer's serial port, save it in a file (the raw data file) and once the download is completed, automatically convert the raw data file into a 12d field file using the currently defined data collector.

Hence the **download** option downloads data from the data recorder and creates a data collector raw file and an equivalent 12d field file.

For more detailed information on the survey reduction process in ***12d Model***, go to the Appendix 36 12d Survey Guide.

Selecting **Download raw** brings up the **Survey Data Download** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Port	input	from configuration file	

name of the computer serial port that the data collector is connected to. The default port is specified in the data collector configuration file.

Data bits	input	from configuration file	5,6,7,8

number of bits

Baud rate	input	from configuration file	

speed of the serial port

Stop bits	input	from configuration file	

number of stop bits

Parity	input	from configuration file	

parity

DTR/DSR	tick	from configuration file	

if ticked, use DTR/DSR flow control

RTS/CTS	tick	from configuration file	

*if **ticked**, use RTS/CTS flow control*

XON/XOFF	tick	from configuration file

*if **ticked**, use Xon/Xoff*

ACK/NAK	tick	from configuration file

*if **ticked**, use ACK/NAK*

12d Field file	input	*.fld files

name of the 12d field file that the raw file is to be converted to. The raw file is given the same name but with the extension specified in the configuration file, e.g., ".gre".

Download	button

On clicking the download button, the software will read a data stream from the serial port and store the raw data into a file. The extension of this file, will be specified in the configuration file. The name will be the same as the field file, with any ".fld" removed.

On completion of the download, the raw data file is automatically converted to a 12d field file of name given in the **field file** *field.*

17.5 Convert Raw

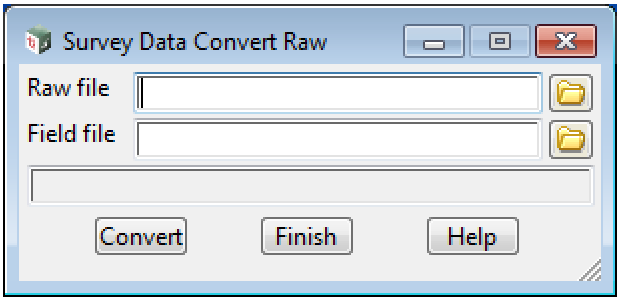
Position of option on menu: **Survey =>Convert raw**

The **convert raw** option is used to convert an existing raw data file of the type given by the data collector setup, into a 12d field file.

The raw data file may have been previously down loaded by 12d Model or obtained via another mechanism.

For more detailed information on the survey reduction process in ***12d Model***, go to the Appendix 36 12d Survey Guide .

Selecting **Convert raw** brings up the **Survey Data Convert Raw** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Raw file	input		available raw files

name of the raw data file to be converted to a 12d field file. The popup will contain all files with the extension specified in the configuration file for raw data files, e.g. ".gre".

Field file	input		*.fld files

name of the 12d field file that the raw file is to be converted to. When the raw file name is given, the file of the same name but with the extension .fld is automatically piped into the **field file** *field.*

Convert	button		

convert the raw data given in the **raw file** *field into the 12d field file format and save it in the file given in the* **field file** *field.*

17.6 Create Survey Function

Position of menu: **Survey =>Create**

The **create** walk-right menu contains options to create a ***12d Model*** *Survey* function from either a 12d field file or from scratch using *Survey Reduction edit commands*.

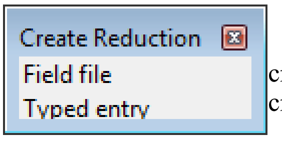
Basically, a **Survey** function keeps track of the field data, the information involved in the survey reduction and all the strings and models created by the *Survey* function. The field data for the *Survey* function can be edited and the reduction re-run and all the old reduced strings automatically deleted and replaced by the updated reduced strings.

For more detailed information on the survey reduction process in ***12d Model***, go to the Appendix 36 12d Survey Guide .

The field data can be read in from a *12d field file* using the **Field file** option, or the **Typed entry** option can be used to enter all the data by hand.

The **create** walk-right menu is:

.



create survey reduction function from a 12d field file - reduces data
create survey reduction function by typed entry into 12d field file.

For the option *Field file*, go to 17.6.1 Field File
Typed entry 17.6.2 Typed Entry

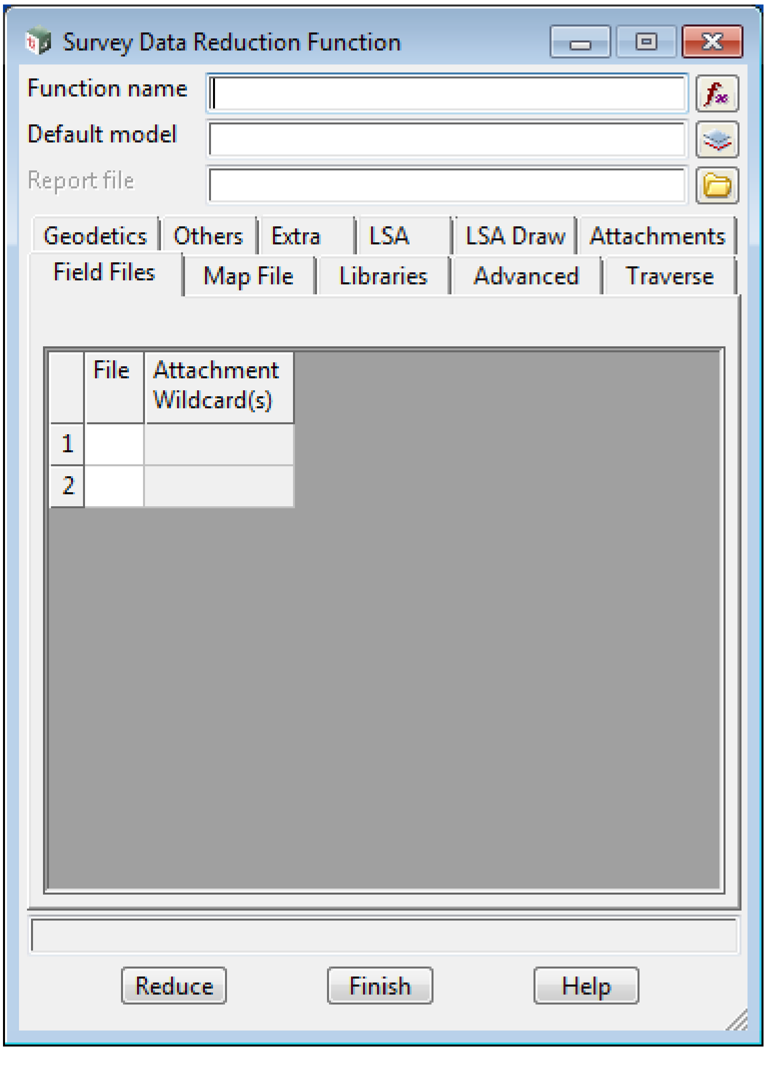
17.6.1 Field File

Position of option on menu: **Survey =>Create =>Field file**

The **field file** option creates a Survey *function* from a 12d field file.

For more detailed information on the survey reduction process in ***12d Model***, go to the Appendix 36 12d Survey Guide .

Selecting **field file** brings up the **survey data reduction function** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function name	input		available Survey functions

name of the 12d Survey function. If the Survey function exists, then it is loaded into the panel. If the Survey function does not exist, then a new Survey function is created.

Default model	input		available models

if a feature code is not found in the map file, or no map file is selected, the strings and points of that feature code will be placed in the **default model***. The colour used will be the default line or point colour for* ***12d Model****. The* **default model** *field is compulsory. Check measurements will be always placed in the default model if a check model is not given (on the Advanced tab).*

Report file input *.rpt

if non-blank, a log of the reduction steps, including new instrument stations, new target heights, scale factors, backsights and check measurements is created. Any errors are also logged.

Button at Bottom

Reduce button

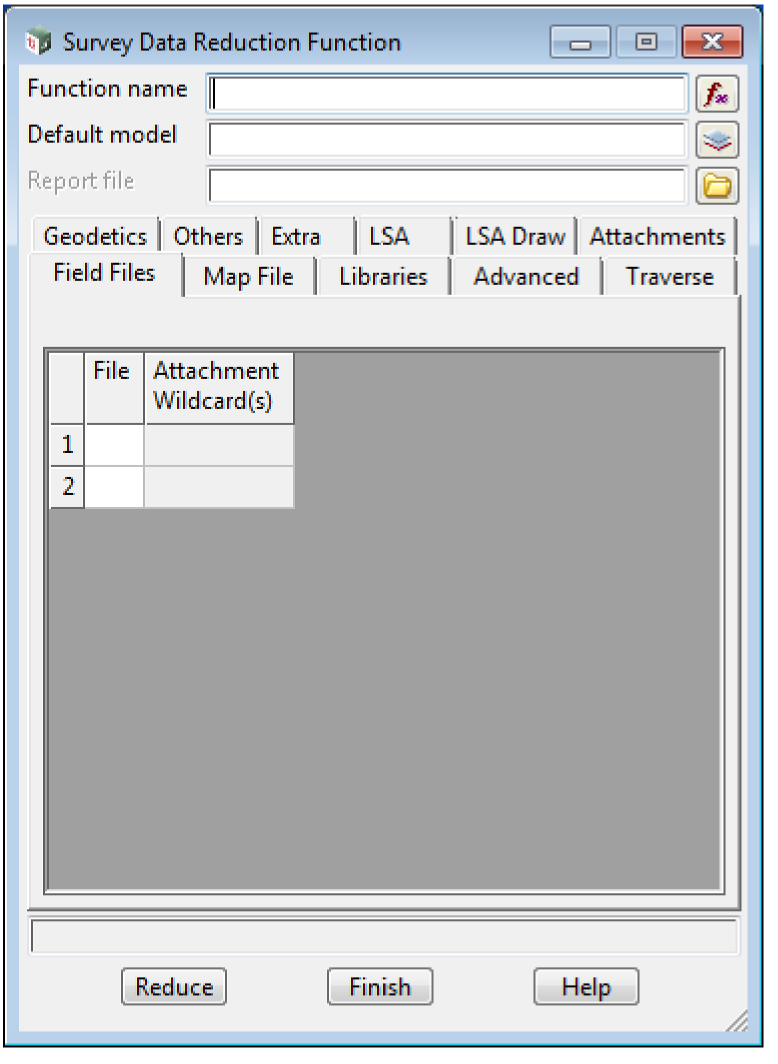
On clicking the reduce button, the software converts the field measurements in the selected 12d field file, using coordinates from the control model if necessary, and produces super strings. The super strings will be assigned names and models using the name library and map file.

For descriptions of each of the tabs on the panel, go to:

Field Files tab
Map File tab
Libraries tab
Advanced tab
Traverse tab
Geodetics tab
Others tab
Attachments tab

Field Files tab

list of field files to read in



The fields and buttons used in this panel have the following functions.

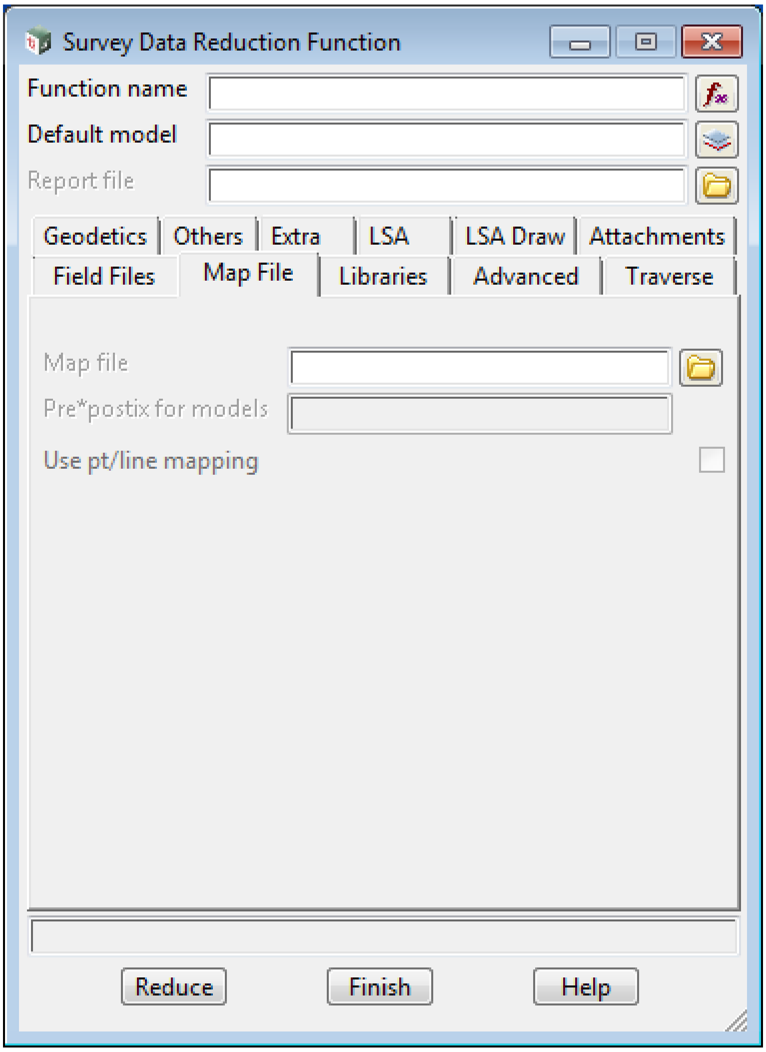
Field Description	Type	Defaults	Pop-Up
Field file	input		*.fld files

name of the 12d field file to be read into the function as its initial field data, and then reduced. The field data in the Survey function can be edited either by command or graphically.

Attachment wildcards	input		

not yet used

Map File tab



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Map file	input		*.mf files

*if **non-blank**, the **map file** is used to map feature codes to string names, models, colours, line styles, etc. In general, strings will be created as line strings by assigning a non-zero string number in the field, and strings will be created as point strings by assigning a zero string number in the field. This behaviour may be overridden with field codes 92, 93 and 94.*

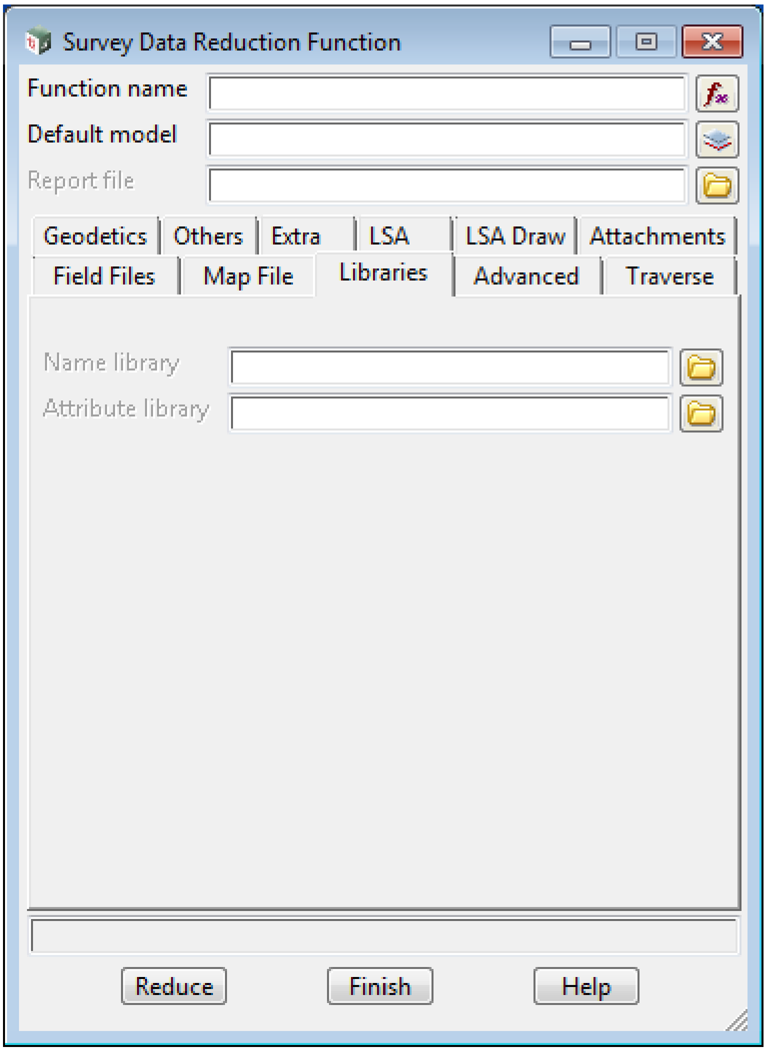
Field Description	Type	Defaults	Pop-Up
Pre*postfix for models	input		available models

*the **pre*postfix for models** text is applied to all model names in the map file.*

Field Description	Type	Defaults	Pop-Up
Use pt/line mapping	tick		

*if **ticked** and a **Map File** is used, the **column** in the map file that specifies the point/line type of the string is be used to set the point/line type of the string.*
*If **not ticked** and a **Map File** is used, the **column** is ignored (not used for setting point/line type of the string).*

Libraries tab



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name library	input		*.nl files

*if **non-blank**, the* **name library** *can be used to automatically create vertex text for the string if no vertex text is given in the field. A name library file contains two columns separated by one or more spaces. The text in the first column is matched against the feature code (wild cards * and ? can be used). The second column contains the vertex text to use if a match occurs.*

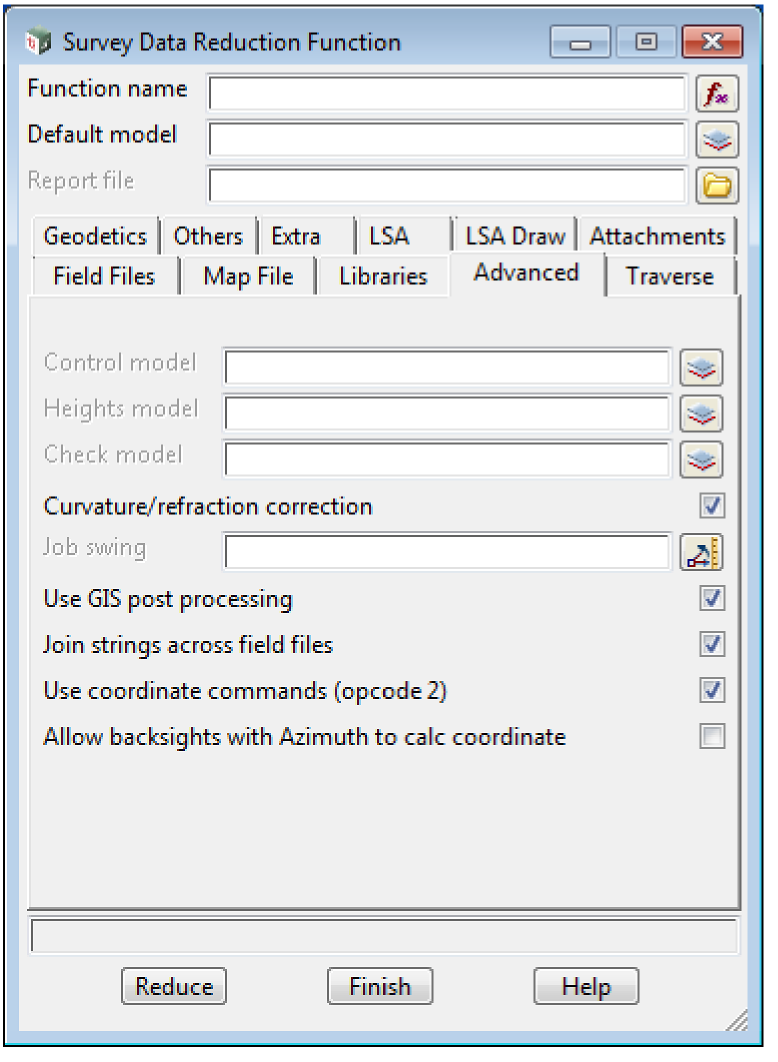
Note that if vertex text is given in the field then it is used instead of the text in the name library.

*Pre-super strings: if **non-blank**, the* **name library** *will be used to automatically create 4d strings and to assign text to 4d strings. Name library files will contain two columns. The first column in the name library will be matched against the feature code. Note that if a match is found, a 4d string is always created, as opposed to another string type. The second column will contain the text for the 4d string.*

Attribute library	input		*.al files

*if **non-blank**, the* **attribute library** *is used to give names to any unnamed attributes. The first word of each line in the file is the code to define the attribute for (can include wild cards) and the next n words are the names for each unnamed attributes in order. If the attribute library is mission, then the unnamed attributes will be given the names **unnamed attribute i** for 1=1, ...*

Advanced tab



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Control model	input		available models

if not blank, name of the model containing instrument and backsight stations that can be referenced during the reduction. The name of the string is taken as the station name and if the string contains more than one point, the first point in the string is used for the co-ordinates of the station.

Heights model input available models

in not blank, name of the model containing points that are used for the heights of points during reduction. The points in the heights model have the point id of required point and a height which is used whenever that point id is found in the reduction. The (x,y) co-ordinates of the point are ignored.

Check model input available models

if non-blank, then all check shots will go to this model.
If blank then check shots go to the **default model**.

Curvature/refraction correction tick box tick

*if **ticked**, during the reduction, each EDM tacheometry measurement (field code 7) has a correction for earth curvature and refraction applied to the measurement.*

Job swing

if non-blank, then the job is swung through this value. The rotation is in a clockwise direction and the *value is in hp dms format.*

Use GIS post processing tick box tick

if ticked, run the given 12d macro to process attribute blocks

Join strings across field files tick box tick

*if **ticked**, points with the same code and string number will be included in the same string in any of the field files. The order that the points are placed in the string is the order of the field files in the **Field File** tab.*
*if **not ticked**, any strings will terminate at the end of a field file. Even if the string has the same name and string number is any following field file, a new string is started in the following field files.*

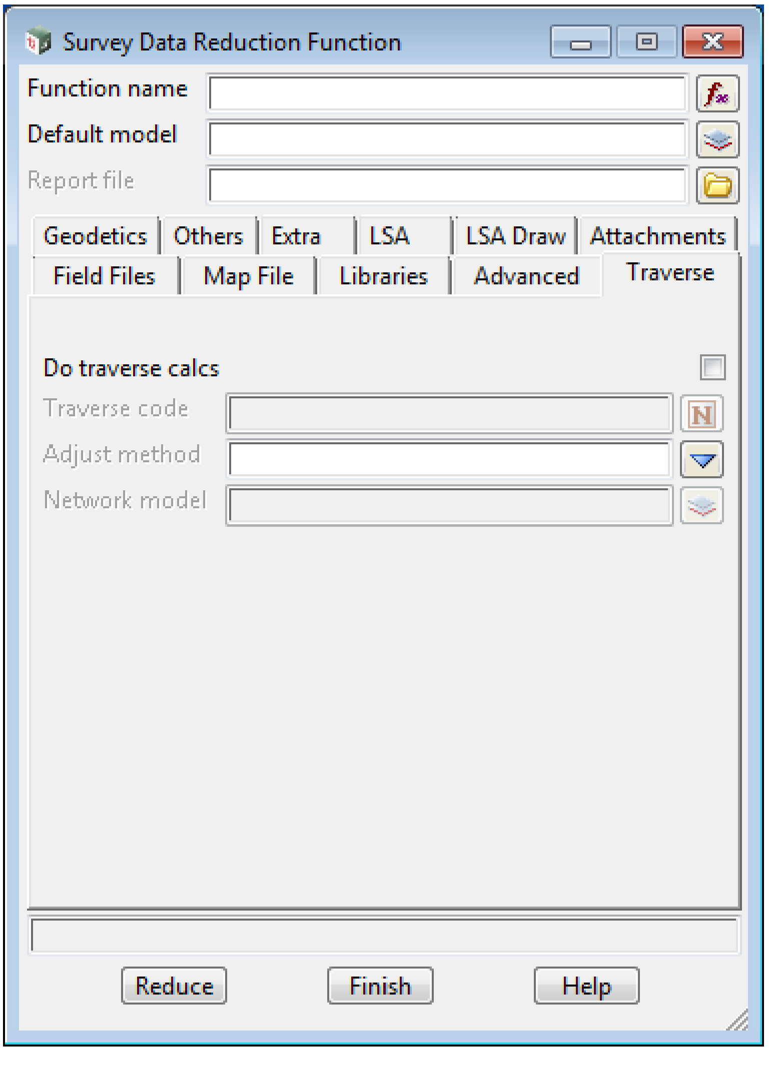
Use coordinate commands (opcode 2) tick box tick

if ticked,

Allow backsights with Azimuth to calc coordinate tick box

If ticked, for backsight commands that reference unknown backsight points, and the backsight has the horizontal angle, vertical angle, slope distance, and Azimuth defined, the backsight point will be created from that information.
This option was only included to allow reduction from legacy external formats, and should not be used as it is not good survey practise.

Traverse tab



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Do traverse calcs	tick box	not tick	

*if **ticked**, the traverse coded points are placed into a network model, and an adjustment done at the time of reduction.*

Traverse code	input		

if valid, this code will be used to identify traverse data. This will effectively be the same as stringing features in the field except the user has the option of isolating the model, adjusting the strings etc.

E.g. TL. The reduction function will look at the field file and extract any observations involving field codes with the TL code. Only backsights and measurements with the point name field populated will be processed.

Each traverse will have a separate string number. i.e. The main traverse with a feature code/string number of e.g. TL7 and a side traverse having a feature code/string number of e.g. TL3. Note the string numbers do not have to be in any particular order. The reduction will determine the order on which the separate traverses will be processed (by dependencies).

See 36.6 Traverse coding *in Appendix* 36 12d Survey Guide .

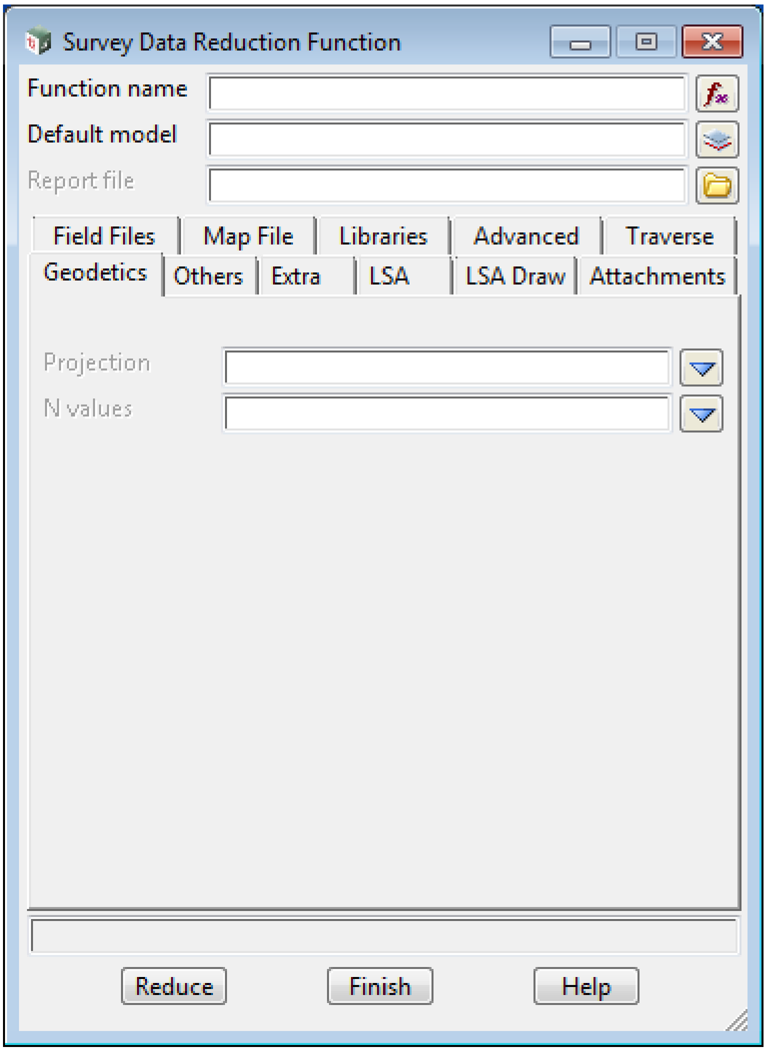
Adjustment method	choice		bowditch, compass, transit, least square, none

*method of adjustment. The adjustment will be made to the extracted traverse strings from known point to known point. **If a traverse string goes through a number of known points (known coordinates), the adjustment will made from known point to known point rather than the first known to the last known point.***

Network model	model box		

if non blank, any valid traverse strings are placed in this model

Geodetics tab



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Projection	projection box	current projection	available projections

the projection of the data to be reduced. If a valid projection is specified, the reduction will be done taking into account the projection scale factors. ***If this method is used it is paramount that the known coordinates(e.g station setups) are in terms of the projection coordinates and are not truncated (i.e. full coordinate values)****. These coordinates will allow the calculation of the relative longitude and latitude values which are used to compute coordinates from observations from the setup points. For more information about how to setup different projections see the section* 7.6.6 Projections .

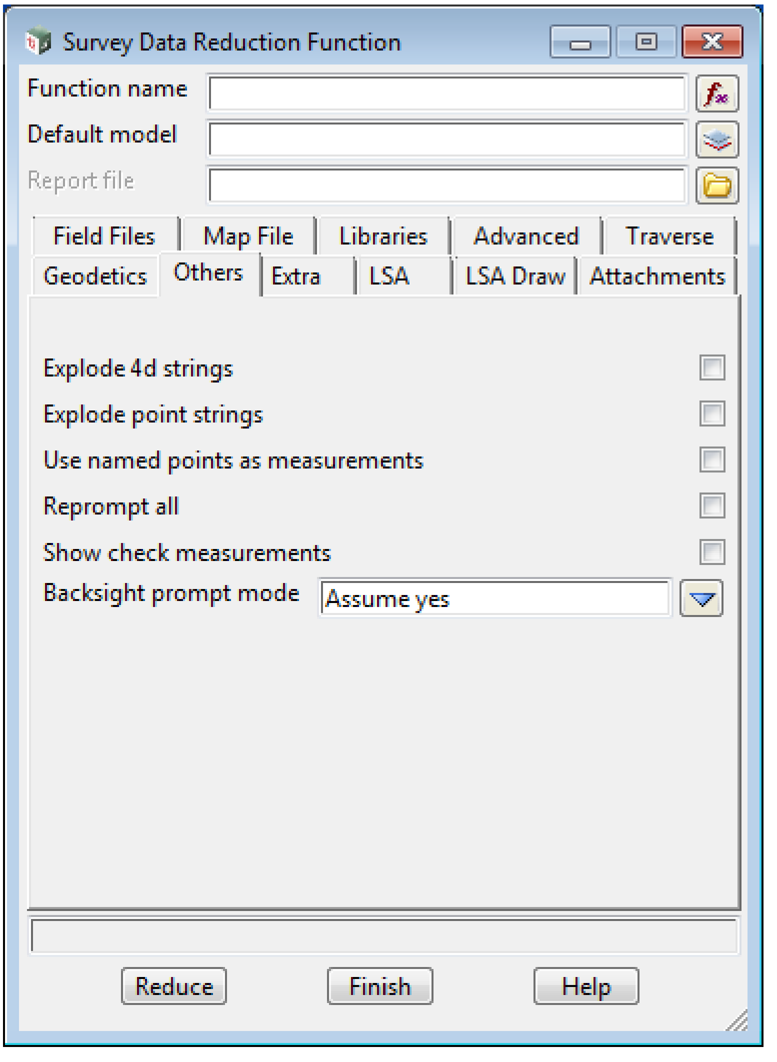
Field Description	Type	Defaults	Pop-Up
N values	choice box	currently set method	Available n value methods

the N value method allows the conversion of non-ellipsoid heights to ellipsoid. The methods are defined in the project n value settings. For more information on the n value settings see the section 7.6.7 N values

The N value will be used to convert a geoid height (e.g. AHD) into an ellipsoid height. Ellipsoid height = geoid height + N value. The conversion is used for the calculations only. The original z value for the point will remain unchanged

For more information about geodetic terminology including projections and N values, see the Appendix 38 Geodetics Summary .

Others tab



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Explode 4d strings tick box

*if **ticked**, any 4d strings created during the reduction process will have only one vertex (point), i.e., no stringing of 4d strings will occur.*

Explode point strings tick box

*if **ticked**, point strings are created as one vertex (point) strings i.e., no stringing of any point strings will occur.*

Use named points as measurements tick box

*if **ticked**, named points are also created as measurements.*

Reprompt all

*if **ticked**, all check measurements are redisplayed on recalcs. Also if* **Backsight prompt mode** *is set to prompt, then backsights are redisplayed on recalcs.*
*if **not ticked**, check measurements are not redisplayed on a recalc and backsights depend on the backsight prompt mode.*

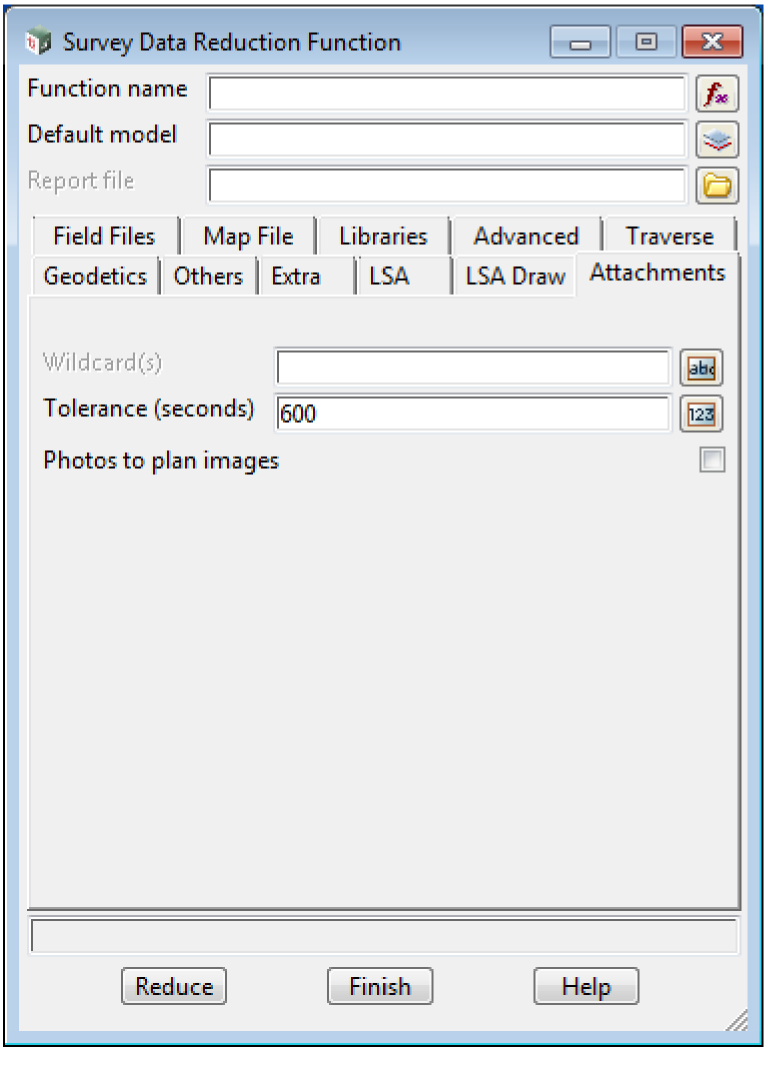
Show check measurements tick box

*if **ticked**, when a check measurement is processed a panel is displayed showing the calculated (x,y,z)* *for the check measurement.*

Backsight prompt mode choice box prompt, assume yes
assume no

*if **prompt**, then the backsight panel is brought up in a command has not been answered adequately previously or if something has changed that warrants a redisplay of the panel.*
*If **assume yes**, then **yes** is assumes for all the backsights and the panel is not brought up.*
*If **assume no**, then **no** is assumes for all the backsights and the panel is not brought up.*

Attachments tab



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Wildcard(s)

space delimited list of wildcards for attaching files to shots. For example, photo images taken in the field.

Tolerance (seconds) input 600

the shot must be taken first and then any files with time less than this number of seconds after the shot, but before any subsequent shots, are attached to the shot

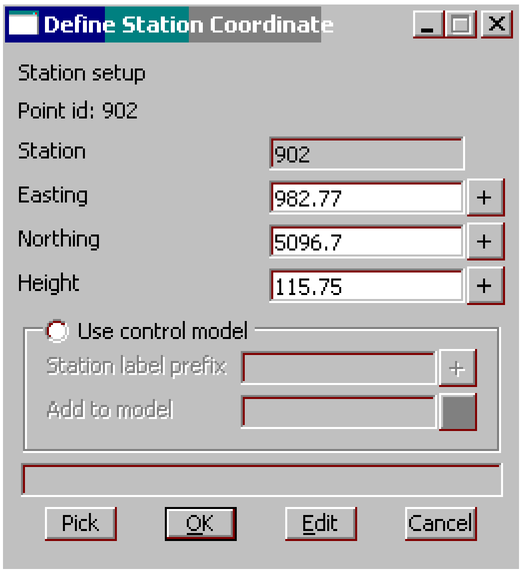
Photos to plan images tick box

*if **ticked**, 12d plan images are created for ay attached image in jpeg, tiff, bmp or png format*

During the reduction, the coordinates of a point may need to be supplied. This occurs when:

- an undefined point is named as the new instrument point (field code 3)
- backsight measurements are made to an undefined point (field code 4)
- check measurements are made to an undefined point (field code 6)

In this situation, a **define station coordinate** panel will pop up:



The coordinates may be typed in, or selected from existing strings. The station may also be added to a model to allow it to be used as a station later.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Station name	output		

the station name

Use control model radio button

if ticked, the point is added to the control model.
if not ticked, the point is created but is not placed in the control model.

Easting/Northing/Height input — Pop-Up: xyz ops menu

the easting (x)/northing (y)/height (z) co-ordinates for the station.

Station prefix input

if non-blank, the created station name will be prefixed by this value.

Add to model input — Defaults: control model — Pop-Up: available models

if not blank, a 4d string is created using the **Easting/Northing/Height** *values as co-ordinates, and the station name (with a possible prefix) as the text and name of the 4d string. The station can then be referred to later in the reduction.*

Pick button

pick a point whose co-ordinates are piped into the **Easting/Northing/Height** *fields.*

OK button

on selecting **ok***, the coordinates entered in the* **station xyz field** *are used as the station co-ordinates* *and if the* **add to model field** *is not blank, a 4d string created.*

Edit button

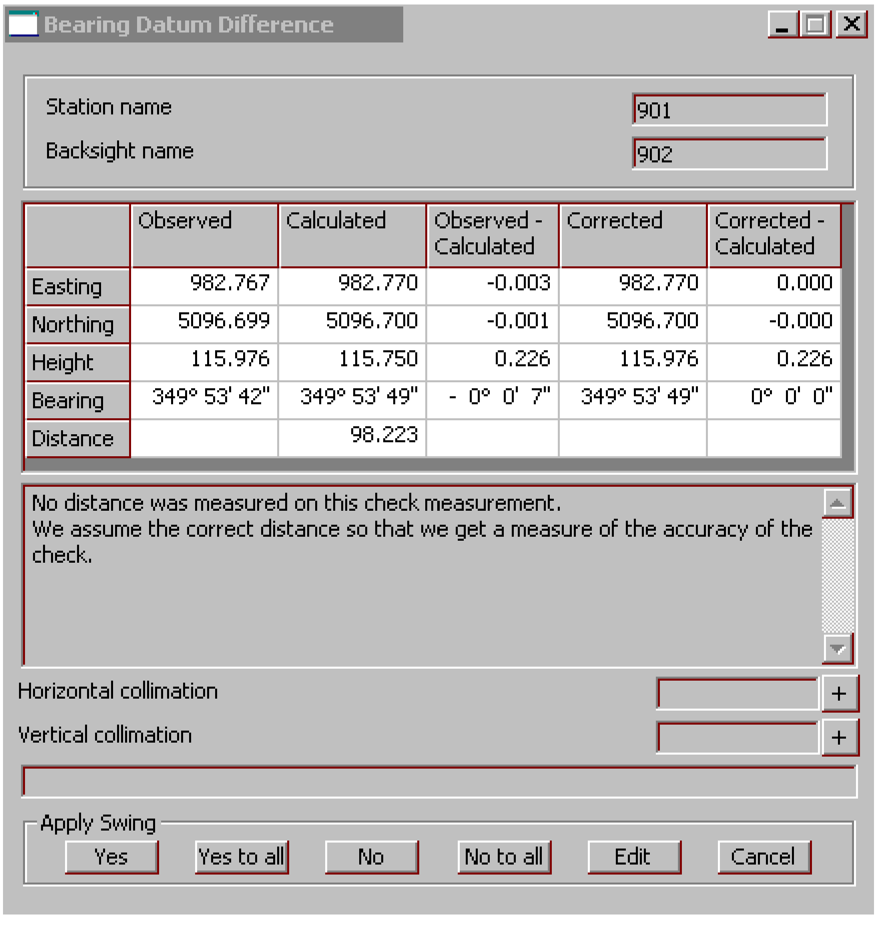
on selecting **Edit***, the next new instrument station entry is shown which allows the station number and associated fields to be edited.*

Cancel button

the cancel button will remove the panel and terminate the reduction.

If the **Backsight prompt mode** is set on, then at each backsight measurement, the **bearing datum difference** panel is displayed

:



If **yes** button is selected, the bearing datum difference is applied to this horizontal circle reading.

If **yes to all** button is selected, the bearing datum difference is applied to all subsequent horizontal circle readings.

The **cancel** button terminates the reduction.

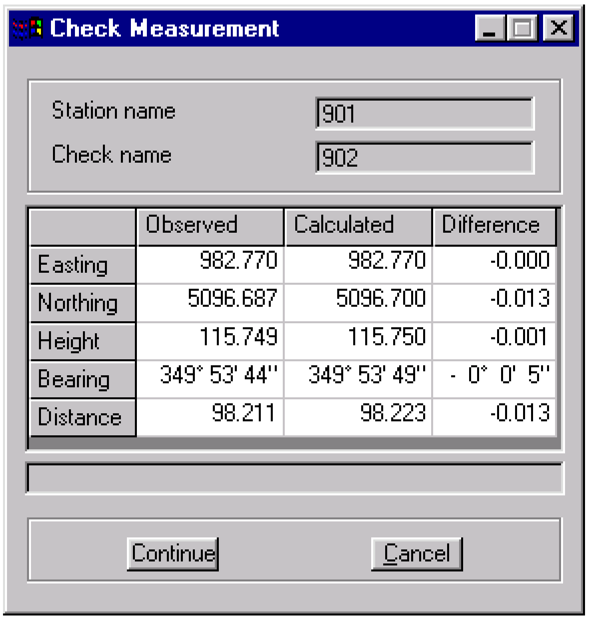
If **no** is selected, this horizontal circle reading is treated as actual bearing.

If **no to all** is selected, all the horizontal circle reading are treated as actual bearings.

This facility allows the horizontal circle to have any orientation during data capture.

At each check measurement, the **check measurement** panel is displayed giving information about the measurement, the point it was a check measurement for and the differences between the two.

:



If the **Continue** button is selected, the reduction continues.

The **Cancel** button terminates the reduction

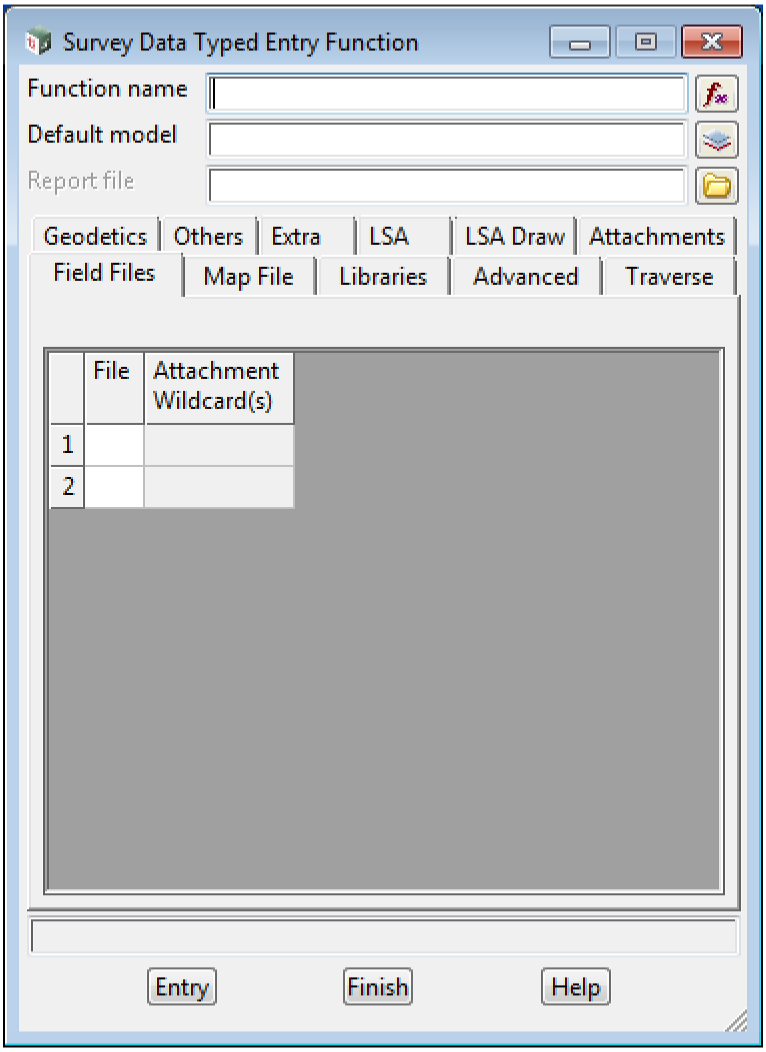
17.6.2 Typed Entry

Position of option on menu: **Survey =>Create =>Typed entry**

The **Typed entry** option creates a *Survey function* from scratch using the edit commands.

For more detailed information on the survey reduction process in ***12d Model***, go to the Appendix 36 12d Survey Guide .

Selecting **Typed entry** brings up the **survey Data Typed Entry Function** panel



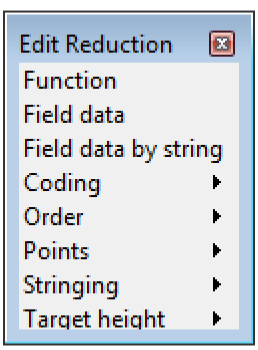
The fields and buttons are identical to those in the **Survey Data Reduction Function** panel in the previous section 17.6.1 Field File .

17.7 Edit Survey Function

Position of menu: **Survey =>Edit =>Edit Reduction**

The **Edit** walk-right menu is

.



edit 12d Survey function
edit field data for a 12d Survey function
edit 12d Survey function by selecting a string
point code correction
re-order points
graphically selected field edits for individual shots
graphically selected field edits for strings
insert and/or modify target heights

The **Edit reduction** walk-right menu contains options to create a ***12d Model*** *Survey function* from either a *12d field file* or from scratch using *Survey Reduction edit commands*.

Basically, a *Survey* function keeps track of all the information involved in the survey data reduction and all the strings and models created by the *Survey* function. The *Survey* function can be re-run and all the old information automatically deleted and replaced by the updated information.

Once the *Survey* function is created, it then reduces the data according to the information supplied and stored with the *Survey* function. As with other ***12d Model*** functions, the *Survey* function information can be modified and the function recalculated.

Many errors in a raw data file are only detected after the reduction has taken place These are then corrected and the function recalculated to re-reduce the data.

Some of the types of problems that regularly occur during electronic field surveys are:

- Instrument height incorrectly entered
- Target height incorrectly entered
- Change of target height not entered
- Backsight station incorrectly named
- Instrument station incorrectly named
- Feature code and string number errors

The *12d Survey* edits allow for all commands in the field data to be modified and new commands added. Selection for some edits can be done graphically (for example incorrect target height) and others through interactive editing of the Survey field data. In all cases, once the edit is completed, the effect is immediately calculated and displayed.

For more detailed information on the survey reduction process in ***12d Model***, go to the Appendix 36 12d Survey Guide.

Each option will now be described in detail.

For the option *Function*, please go to	17.7.1 Function
Field data	17.7.2 Field Data
Field data by string	17.7.3 Field Data by String
Coding	17.7.4 Coding
Order	17.7.7 Order
Points	17.7.8 SDR Point Edits
Stringing	17.7.9 SDR Strings Edit
Target height	17.7.10 Target Height

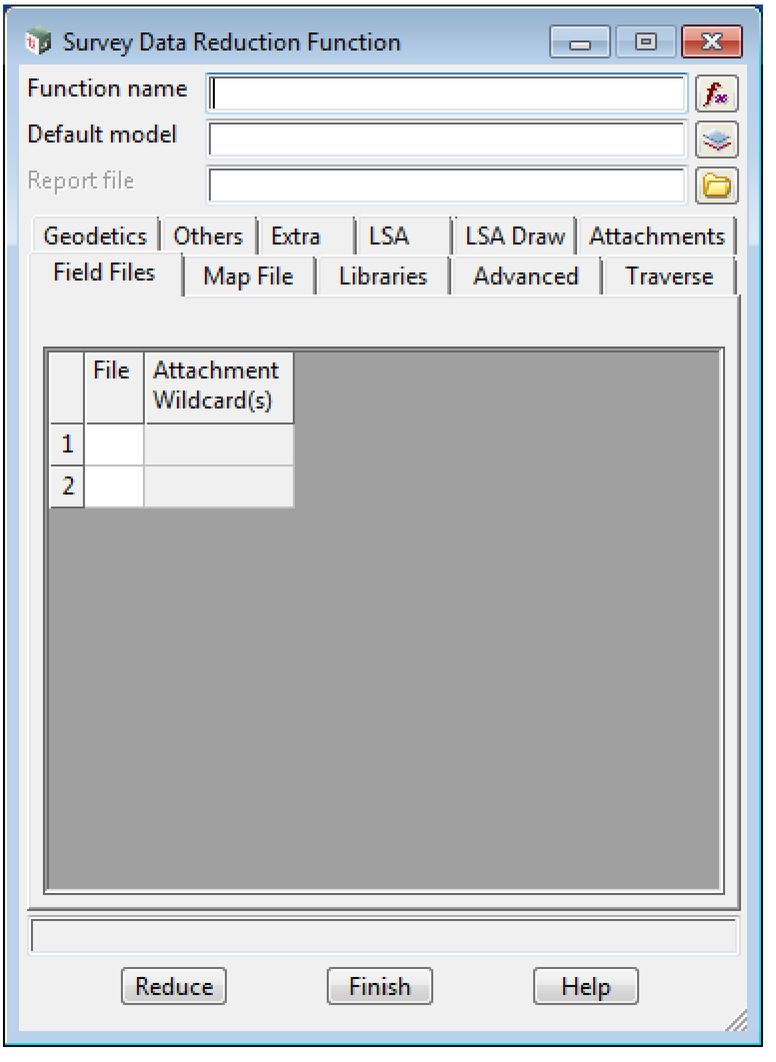
17.7.1 Function

Position of option on menu: Survey =>Edit =>Function

The **function** option is used to edit the set-up panel values for a *Survey* function.

Note - to edit the *field data* rather than the set-up panel, the **Field data** option is used.

Selecting **Function** brings up the **Survey Data Reduction Function** panel (which is the same panel used to create the function in the **Survey =>Create =>Field file** option).



A *Survey* function is selected for editing by choosing the survey function from the **Function name** pop-up list which loads up the panel information for that function.

The panel information is then modified and the new values stored and used by selecting the **Reduce** button. This recalcs the function which re-reduces the data using the new panel fields.

For more information on the fields in the panel, go to the section 17.7 Edit Survey Function .

For information on the *Survey Field Data Editor*, go to the section 17.7.5 Survey Field Data Editor
.

For more detailed information on the survey reduction process in ***12d Model***, go to the Appendix 36 12d Survey Guide .

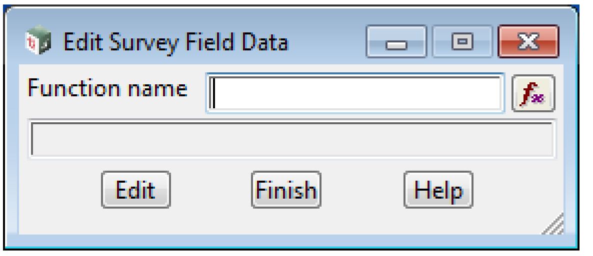
Please continue to the next section 17.7.2 Field Data .

17.7.2 Field Data

Position of option on menu: **Survey =>Edit =>Field data**

The **field data** option is used to edit the *field data* from a Survey function. As the changes are made, the field data is automatically reduced to reflect the changes.

Selecting **Field data** brings up the **Edit Survey Field Data** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function name	function box		all survey functions

name of the Survey function to have the field data modified.

Edit	button		

clicking **edit** *will start the Survey Field Data Editor for the given Survey function. This will be documented in the section Survey Field Data Editor.*

For information on the *Survey Field Data Editor*, go to the section 17.7.5 Survey Field Data Editor
.

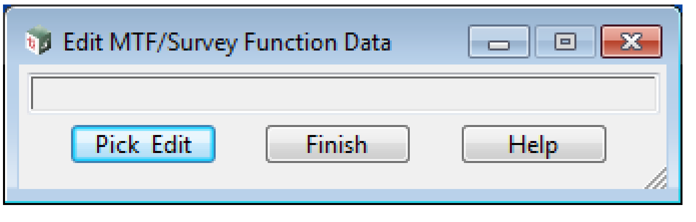
Please continue to the next section 17.7.3 Field Data by String .

17.7.3 Field Data by String

Position of option on menu: **Survey =>Edit =>Field data by string**

The **Field data by string** option is used to edit the *field data* by selecting a string created from a Survey function.

Selecting **Field data by string** brings up the **Edit MTF/Survey Function Data** panel:



After selecting the **Pick Edit** button, a string created by Survey function is selected and the Survey Field Data Editor is started for the given Survey function.

For information on the *Survey Field Data Editor*, go to the section 17.7.5 Survey Field Data Editor
.

Please continue to the next section 17.7.4 Coding .

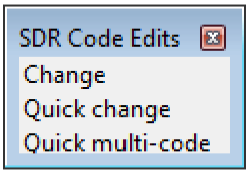
17.7.4 Coding

Position of menu: **Survey =>Edit =>Coding**

The **coding** options are used to modify the feature code and/or string number of selected points. The 12d field file is automatically updated.

The **coding** walk-right menu is

.



change for feature code, string number
change feature code, string number for many point

Each option will now be described in detail.

For the option *Change*, go to		17.7.4.1 Change
	Quick change	17.7.4.2 Quick Change
	Quick multi-code	17.7.4.3 Quick Multi Coding

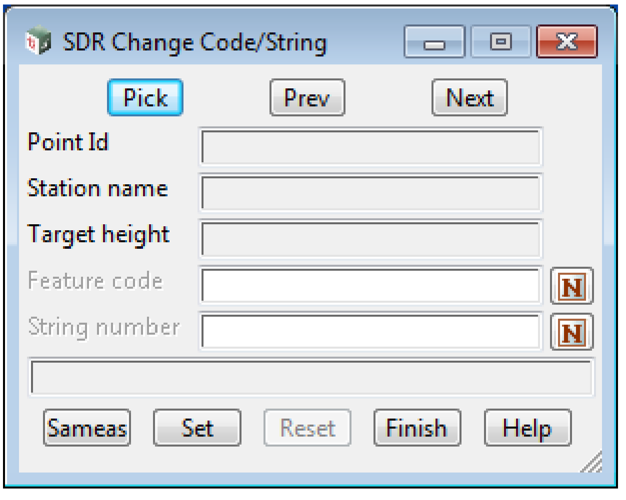
17.7.4.1 Change

Position of option on menu: **Survey =>Edit =>Coding =>Change**

Change is used to modify a points feature code and/or string number. When a point is selected, its point id, station name and target height are displayed as well as feature code and string number.

If either the feature code or string number is then modified in the panel, selecting the set button will give the selected point the new values for feature code and/or string number.

Selecting **Change** brings up the **SDR Change Code/String** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Pick button

select a point whose information will be displayed in the panel fields. A highlighted line joining the point to the station that it was surveyed from is also displayed.

Prev button

move to the previous point in the string.

Next button

move to the next point in the string.

Point id display only

point id of the selected point.

Station name display only

name of the station that the point was surveyed from.

Target height display only

height of the target when the point was surveyed.

Feature code input/output

when a point is selected, its feature code is displayed in this field. The value can then be changed by the user and if **set** *is selected, the point is given the new feature code.*

String number input/output

when a point is selected, its string number is displayed in this field. The value can then be changed by the user and if **set** *is selected, the point is given the new string.*

Same as button

after selecting **same as***, a point is selected and its feature code and string number are piped into the panel fields.*

Set button

selecting **set** *gives the selected point the values in the panel fields.*

Reset button

reset *undoes the changes.*

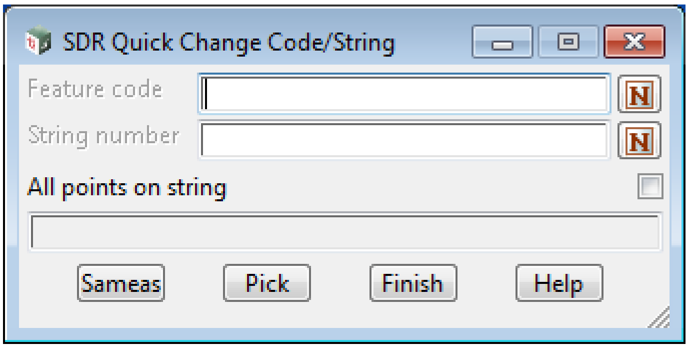
Please continue to the next section 17.7.4.2 Quick Change .

17.7.4.2 Quick Change

Position of option on menu: **Survey =>Edit =>Coding =>Quick change**

The **quick change** option is for more advanced users wishing to modify the same feature code and/or string number for many points.

The new feature code and/or string number is entered into the panel and whenever a point is selected and accepted, it is given the new feature code and/or string number. The picking of points continues until the picking is cancelled from the **pick ops** menu.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Feature code input/output

if not blank, when a point is accepted its feature code is changed to the value in this field.

String number input

if not blank, when a point is accepted its string number is changed to the value in this field.

Same as button

after selecting **same as***, a point is selected and its feature code and string number are piped into the panel fields.*

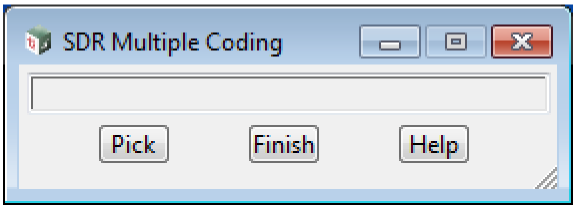
Pick button

selecting **pick** *starts the picking sequence and any accepted point will have its feature code and/or string number modified if the corresponding panel field is not blank. The pick is terminated by selecting* **cancel** *from the* **pick ops** *menu.*

Please continue to the next section 17.7.4.3 Quick Multi Coding .

17.7.4.3 Quick Multi Coding

Position of option on menu: Survey =>Edit =>Coding =>Quick multi-code



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Pick button

after selecting **Pick,** *pick the vertex that you want to multi-code, and then pick a string and the code and string number from the string will be added to the selected vertex as a multi-code.*

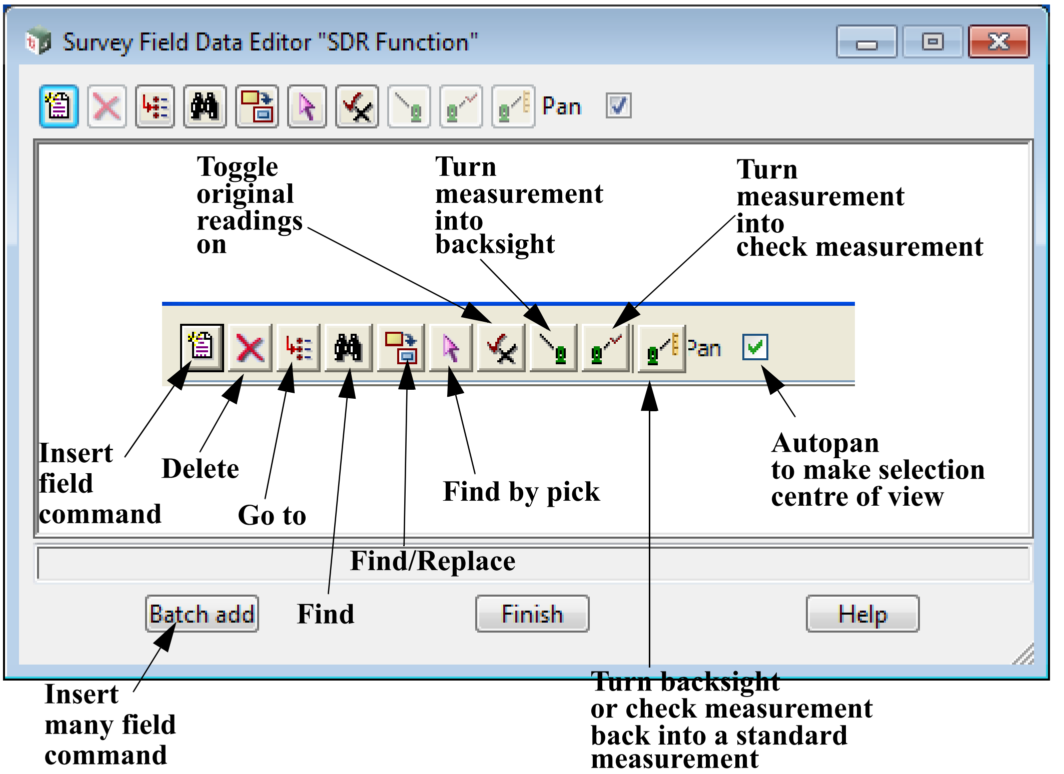
The sequence start again and continues until the pick is terminated by selecting **Cancel** *from the* **Pick ops** *menu.*

Please continue to the next section 17.7.5 Survey Field Data Editor .

17.7.5 Survey Field Data Editor

The **Survey Field Data Editor** is used to interactively edit the field data of a ***Survey*** function. All the commands in the field data can be modified or deleted, or new commands added. When changes are made, the field data is automatically re-reduced so the effects can be immediately checked.

If the Editor is opened using the **Survey=>Create=>Typed Entry** option the editor will appear blank.



For information on using things inside the panel, go to 17.7.5.1 Using the Survey Field Data Panel

For information on the field commands, go to 17.7.6 List of Field Data Commands and Panels

For information on the icons and buttons on the panel:

- *Insert*	*go to*	Insert and Batch Add button
- *Delete*		Delete
- *Go to*		Goto
- *Find*		Find
- *Find*/Replace		Find/Replace
- *Find by Pick*		Find by Pick
- *Toggle*		Toggle
- *Make backsight*		Make Measurement into a Backsight
- *Make check measurement*		Make Measurement into a Check Measurement
- *Change to measurement*		Make into a Measurement
Pan - *Autopan*		Autopan
- *Batch add*		Insert and Batch Add button

17.7.5.1 Using the Survey Field Data Panel

17.7.5.1.1 Colour Coding of the Field Data

Normally data has been loaded into the Survey function by a raw data file being downloaded from a data collector, converted to a 12d field file and then loaded into a Survey function.

When ***12d Model*** converts the raw file to a 12d field file, all the raw data can be included in the 12d field file as comments.

All **comments** lines are displayed in **green**.

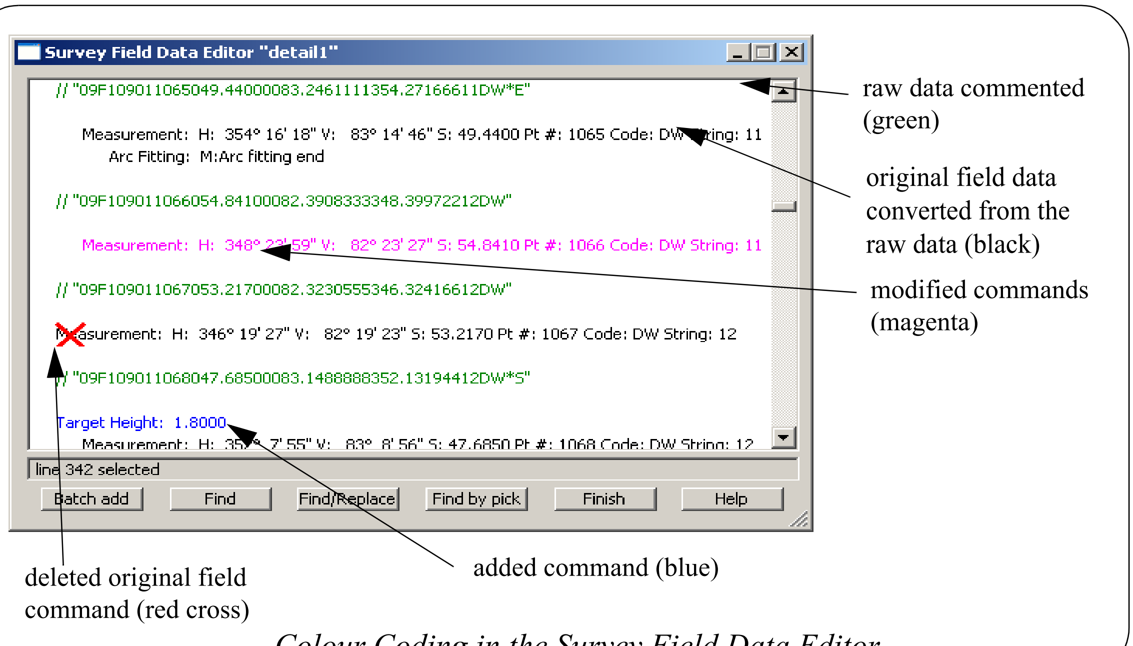
Each line (or lines if necessary) of raw data is converted into one or more equivalent 12d field file commands. These field file commands are placed straight after the commented line containing the raw data lines so it is easy to see what the raw data was converted to.

The **original** 12d **field commands** converted form the raw data are displayed in **black**.

If a field command has been **inserted**, it is coloured in **blue**. If a comment has been inserted, it will still be displayed in green but the word ***: added*** will be appended to the end of the comment.

If a field command has been **modified**, it is displayed in **magenta**.

If an **original** command has been **deleted**, the original command is left in the file but it has a **large red cross** placed on the left hand side of the line to indicate that it has been deleted. Any original command that has been deleted can be undeleted by simply deleting the command again.

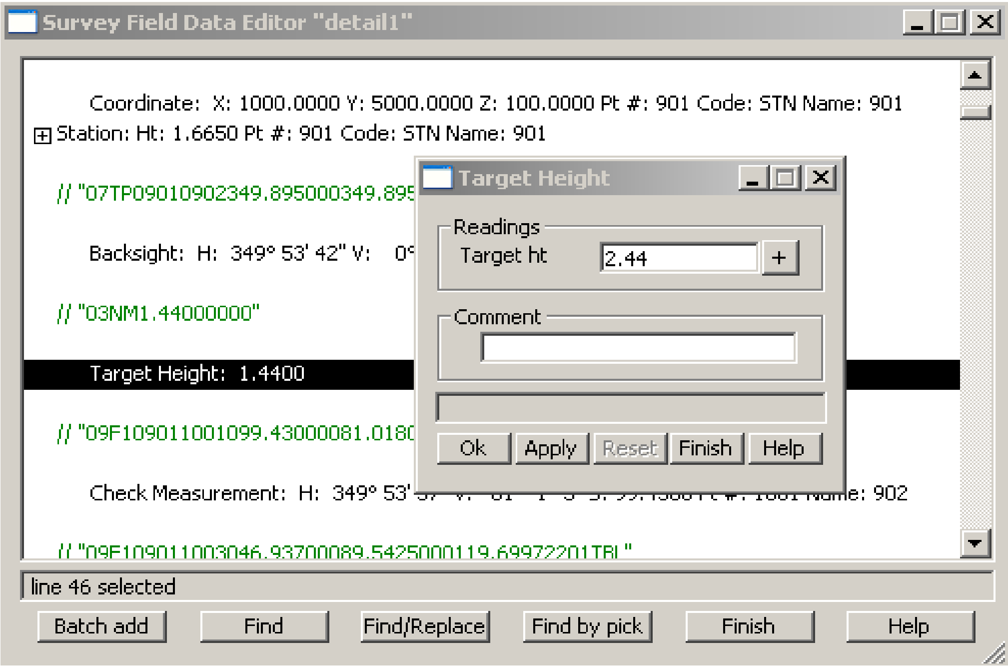


Colour Coding in the Survey Field Data Editor

17.7.5.1.2 Modifying the Field Data

To modify an existing command in the field data, double click on the line to be modified and the appropriate panel for the line of field data is displayed. The data in the panel can then be modified and stored by selecting either the **OK** or **Apply** buttons. The colour of the line of modified field data is then magenta.

For example, double clicking on **target height** command in the field file brings up the **Target Height** panel with the values loaded into the appropriate panel field. The data in the panel can be modified and saved by clicking **OK**.



17.7.5.1.3 Deleting Field Data

To delete a line from the field data, simply highlight the data by clicking in the line and then delete key on the keyboard or select the delete icon from the top of the panel. If the line was an original command, a red cross is placed

If an **original** command has been **deleted**, the original command is left in the file but it has a **large red cross** placed on the left hand side of the line to indicate that it has been deleted. Any original command that has been deleted can be undeleted by simply highlighting the deleting command and pressing the delete icon or delete key again.

If an **inserted** command has been **deleted**, the command is removed from the field data.

More than one line can be selected for deletion by using the standing Windows selection commands to highlight the commands and then press the delete key on the keyboard or select the delete icon from the top of the panel. Similarly to undelete multiple commands.

For information on the field commands, go to 17.7.6 List of Field Data Commands and Panels

For information on the icons and buttons on the Survey Field Data panel, go to 17.7.5.2 Icons and Buttons on the Survey Field Data Editing Panel

17.7.5.2 Icons and Buttons on the Survey Field Data Editing Panel

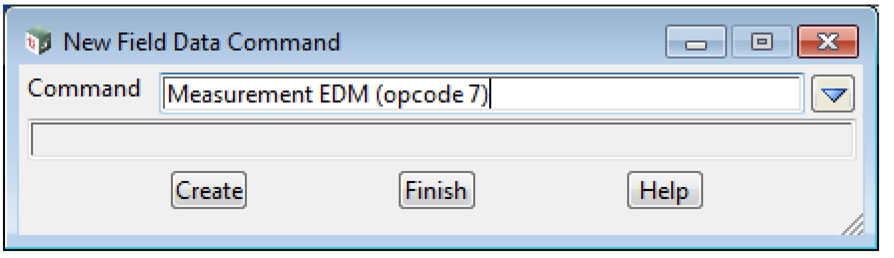
The use of the buttons at the bottom of the survey field data editing panel will now be described.

Insert and *Batch Add* button

To insert a new command into the field data, simply click on the line where the data is to be inserted and press the **Batch Add** button or select the **Insert** icon from the top of the panel.

The difference between the button and the icon, is that the button allows multiple entries whilst the icon allows one entry at a time.

Selecting either the icon or the **Batch Add** button, displays the **New Field Data Command** panel.



By clicking on the **Command** choice box selector, a full list of field commands is displayed and the required command is selected from the list.

Clicking **Create** then brings up the panel for the selected command type. The description of the field data commands panels is given in the section17.7.6 List of Field Data Commands and Panels

Alternatively, if the ***12d Field Data Op Code*** is known, this can be entered into the choice box instead of selecting from the list. For example, to enter a ***Measurement EDM*** field data command, the number ***7*** can be inserted in the choice box field.

When using **Batch Add,** once a command is selected, it will remain as a default for subsequent entries until another type is chosen. This allows a number of the same command to be quickly entered such as in the case of *Measurement EDM* by entering data into the appropriate fields and continuing to press <enter> through the default choices

For a more detailed explanation on each of the fields within the panels and the appropriate field ***12d Field File Op Codes*** see the section 36.10.4 Full Description of 12d Field File Op Codes .

For a summary of just the field file op codes, see the section 36.10.5 Summary of 12d Field File Op Codes

Delete

To delete a line from the field data, highlight the data by clicking in the line and then either use the ***Delete*** key on the keyboard or select the **Delete** icon from the top of the panel.

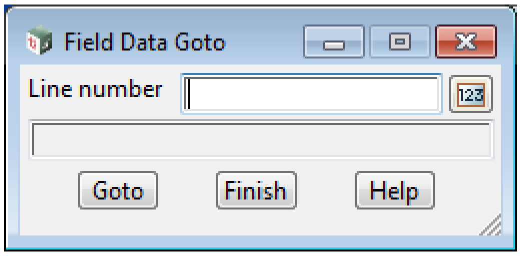
If an **original** command has been **deleted**, the original command is left in the file but has a **large red cross** placed on the left hand side of the line to indicate that it has been deleted. Any original command that has been deleted can be undeleted by simply highlighting the deleted command and pressing the **Delete** icon or ***Delete*** key again.

If an **inserted** command has been **deleted**, the command is removed from the field data.

More than one line can be selected for deletion by using the standing Windows selection commands to highlight the commands and then press the ***Delete*** key on the keyboard or the **Delete** icon from the top of the panel. Similarly to undelete multiple commands.

Goto

Selecting the **Goto** icon brings up the **Field Data Goto** panel.

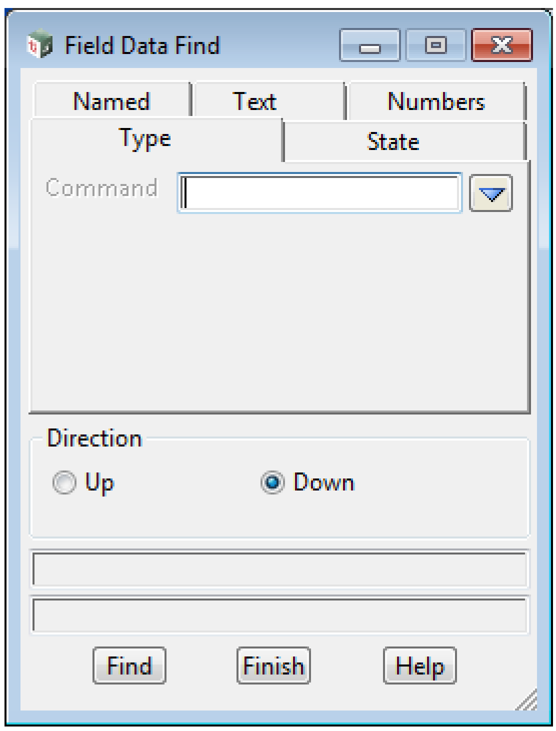


The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Line number			
line number to go to			
Goto	button		
go to the given line number			

Find

Selecting **Find** brings up the **Field Data Find** panel.



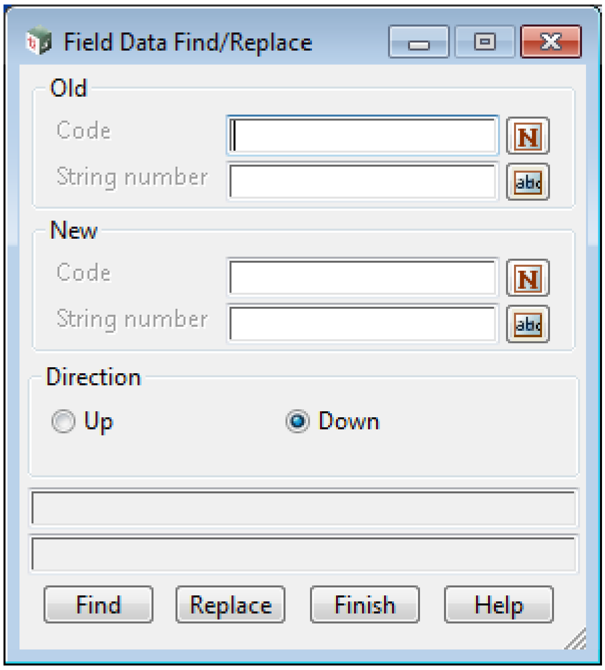
Each of the tabs sets the panel for the required data.

Named find given code, string number, point id etc.
Numbers find numbers between given minimum and maximum.
State find a field command in a given modification state (added, changed, deleted, field)
Text find text containing given text.
Type find a field command of a given type. For example "New String".

The up/down radio button define which direction to search in the field data.

Find/Replace

Selecting the **Find/replace** icon brings up the **Field Data Find/Replace** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

The user specifies the old and new ***Code*** and old and new ***String number*** fields, and the search and replace is done using those parameters.

The **Up/Down** radio button define which direction to search in the field data.

Find by Pick

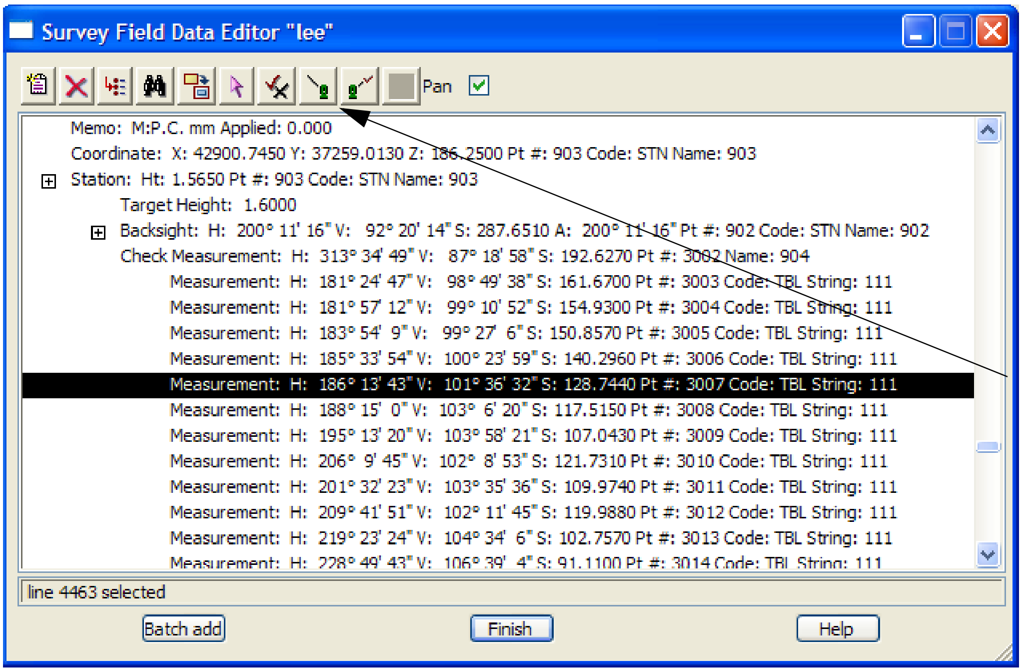
After selecting **Find by pick**, the user selects a point created by the Survey function and the field data editor will scroll to the line of the field data that created the point.

Toggle

The toggle button allows commented information to be shown/ not shown. Simply press the **Toggle** icon to move from one state to another.

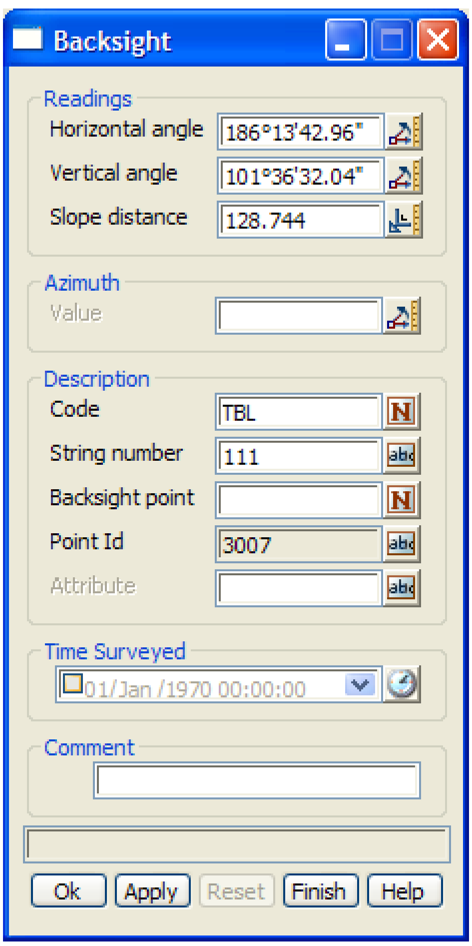
Make Measurement into a Backsight

The **Make Backsight** icon is visible when a ***measurement*** line has been selected (highlighted) in the field data.



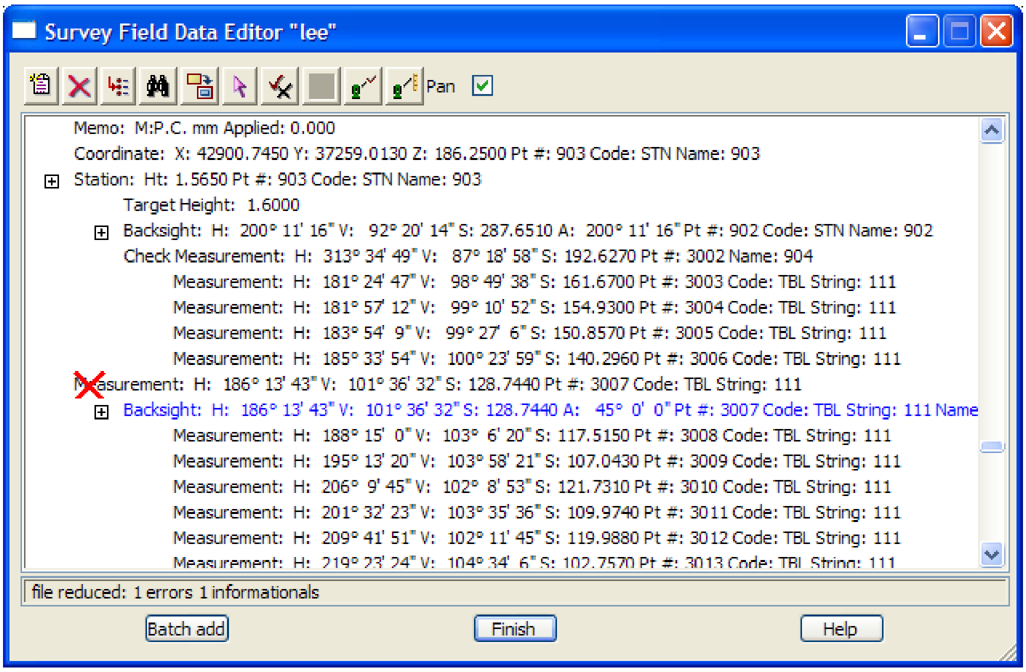
Highlighting a ***Measurement*** then displays the **Make Backsight** and **Make Check Measurement** icons

When the measurement line is highlighted, clicking on the **Make Backsight** icon brings up the **Backsight** panel with the highlighted measurement details filled in.



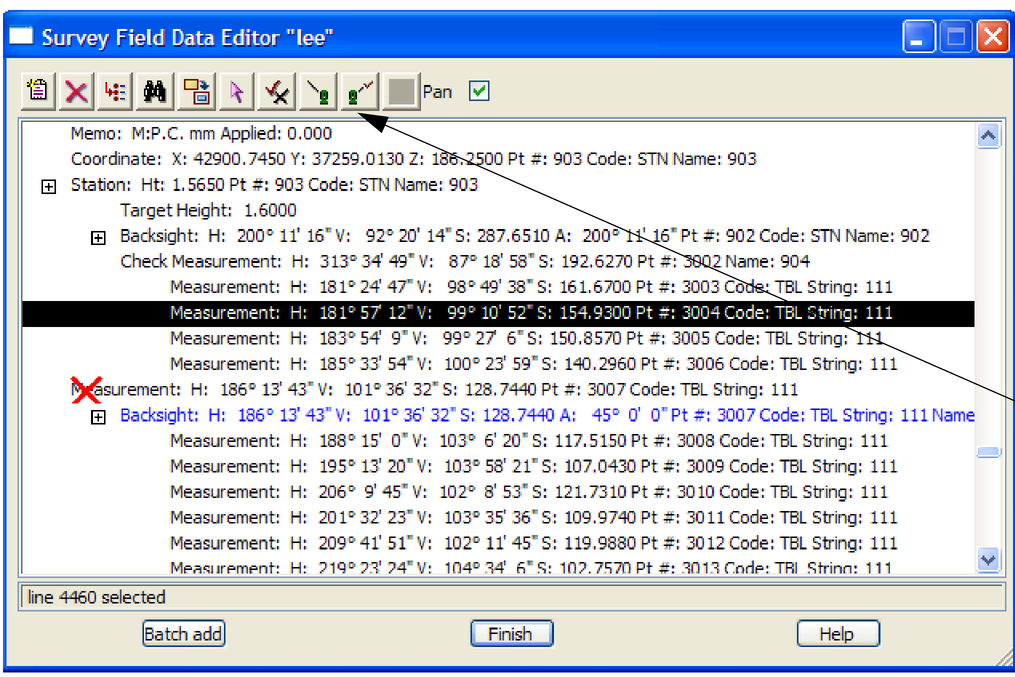
The ***Azimuth*** and ***Backsight point*** fields are filled in and the **OK** or **Apply** button selected.

The original measurement will then be shown as deleted and a new Backsight record created.



Make Measurement into a Check Measurement

The **Make Check Measurement** icon is only visible when a ***measurement*** or ***backsight*** line has been selected (highlighted) in the field data.

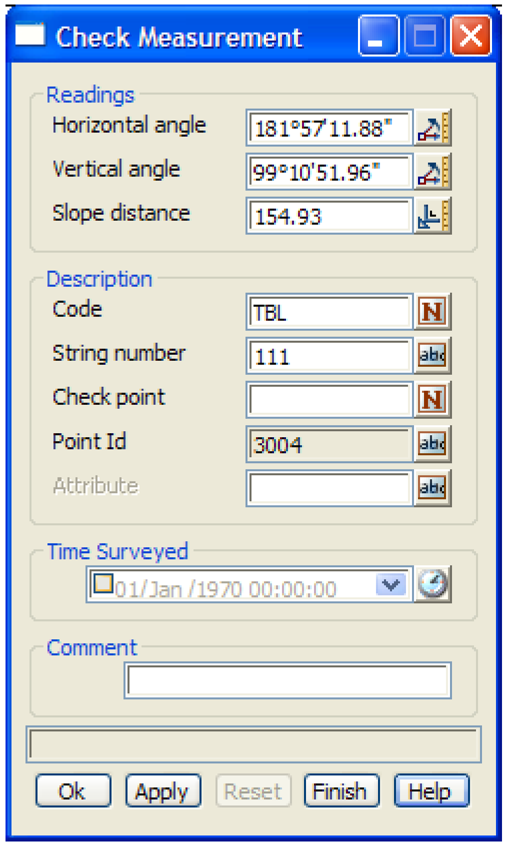


Highlighting a ***Measurement*** or ***Backsight*** then displays the **Make Check Sight**

and

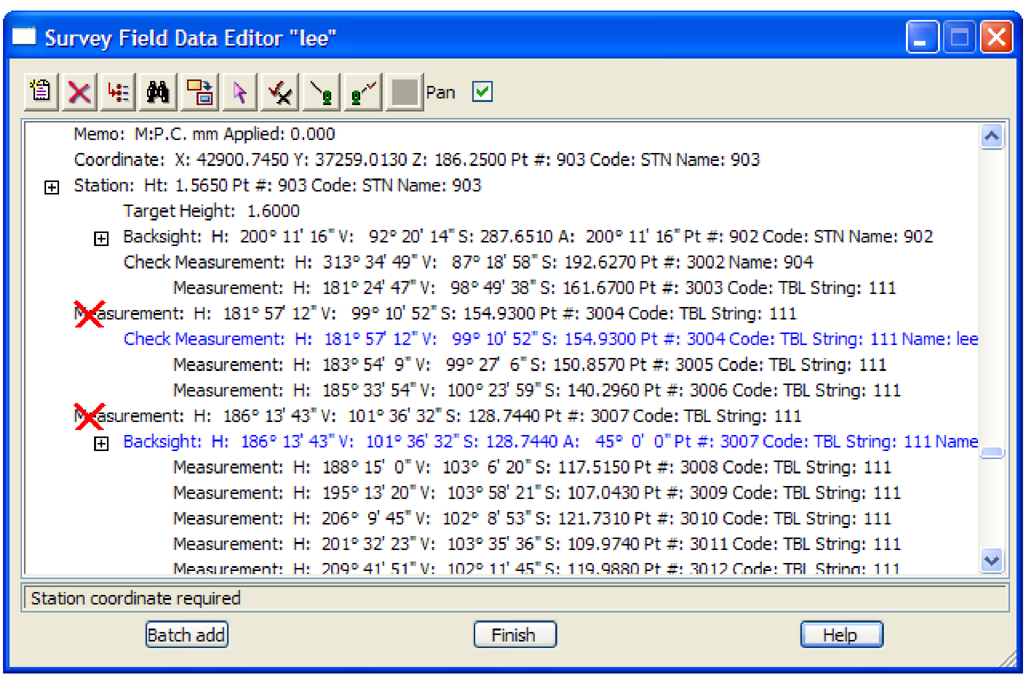
Make Check Measurement

When the measurement line is highlighted, clicking on the **Make Check Measurement** icon brings up the **Check Measurement** panel with the details from the highlighted measurement filled in.

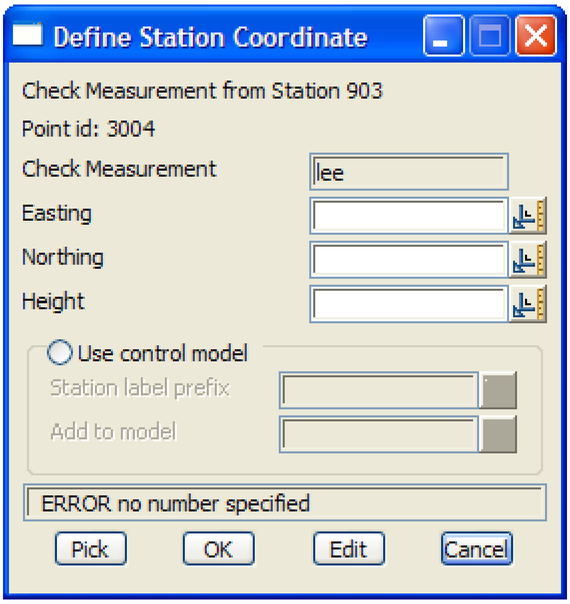


The ***Check point*** is then filled in and the **OK** or **Apply** button selected.

The original measurement will then be shown as deleted and a new ***Check measurement*** record created.

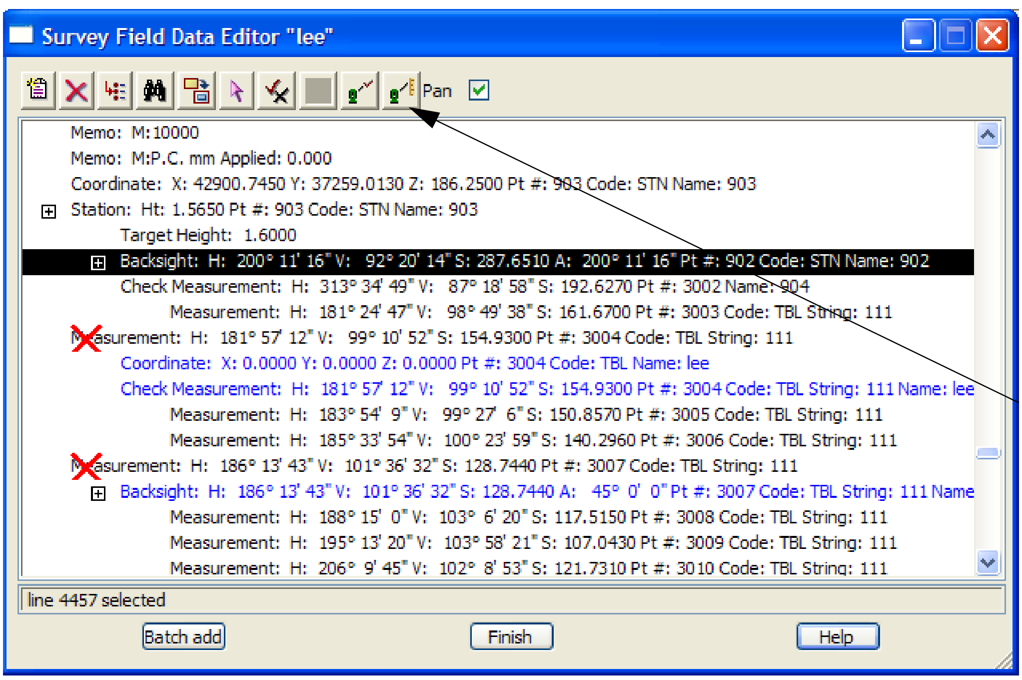


Note - if the ***Check point*** does not exist, a **Define Station Coordinate** panel is displayed for the user to enter the coordinates for the ***Check point***.



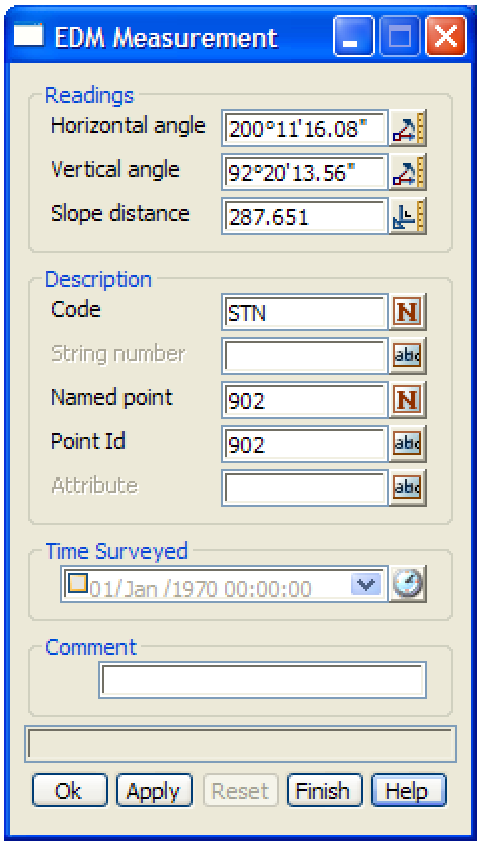
Make into a Measurement

The **Make into a Measurement** icon is visible when a ***backsight*** or ***check measurement*** line has been selected (highlighted) in the field data.



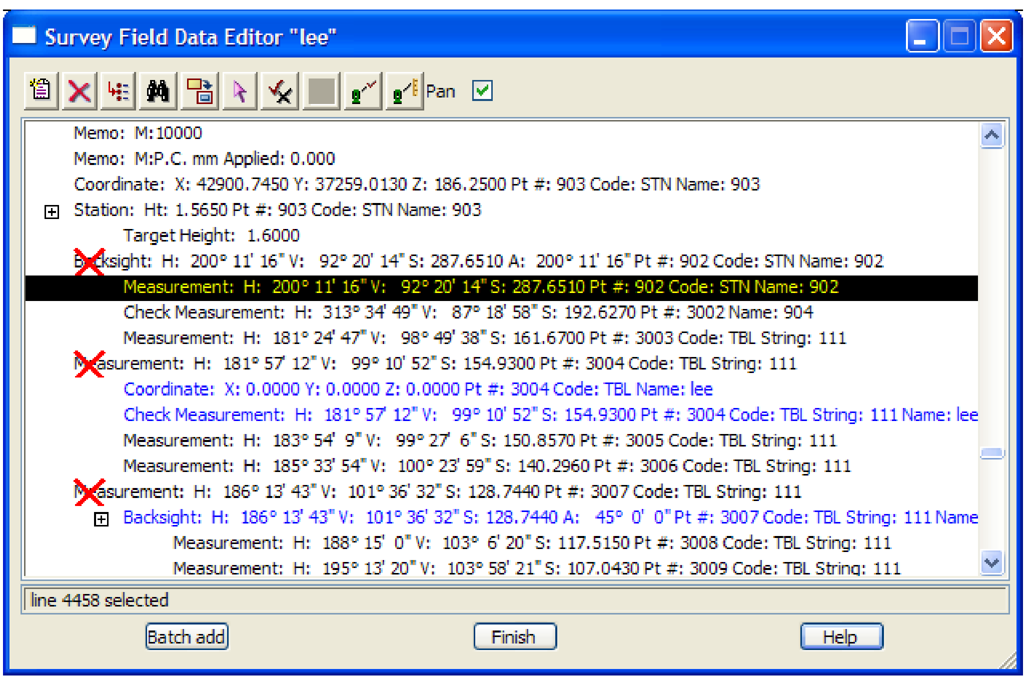
Highlighting a ***Backsight*** or ***Check Measurement*** then displays the **Make Measurement** icon

When the measurement line is highlighted, clicking on the **Make Measurement** icon brings up the **EDM Measurement** panel with the highlighted ***backsight*** or ***check measurement*** details filled in.



The ***String number*** field may need to be filled in and the **OK** or **Apply** button selected.

The original ***backsight*** or ***check measurement*** will then be shown as deleted and a new ***Measurement*** record created.



Autopan

Ticked on - when a line is selected (highlighted) in the survey data editor panel, if the point is not visible in a plan view that the model for the data is on, the view is modified so that the selected point is at the centre of the view (keeping the same scale for the view).

Please continue to the next section 17.7.6 List of Field Data Commands and Panels .

17.7.6 List of Field Data Commands and Panels

Panel buttons

The buttons that appear on the bottom of the panels used in this section have the following functions.

OK

Changes field data and closes the panel

Apply

Changes the field data but keeps the panel in view

Reset

If the record was an original field record (i.e. the command was not inserted manually) pressing the reset button will revert the record back to the original values.

Help and Finish

standard 12d buttons.

Existing point search

Some commands such as ***Backsight***, ***New instrument*** and ***Check*** allow the user to enter existing *point names* (i.e. backsight point, Setup point and Check point) or in the case of measurements from a data collector, both *point name* and *point ids*.

The difference between *point names* and *point ids* is that *point names* are usually specified by the user and should be a unique identifier for a point, whilst for that same physical point a number of measurements (and hence point ids) may be assigned (usually by the data collector). This may be particularly true of control station measurements where measurements are made to a given point name but each measurement is given a different point id by the data collector.

In most instances, a measurement to a point has a point id (from the data collector) and 12d automatically gives it the same point name as it is rare to measure a non-control point more than once (the point name can be over ridden by the user).

The names allow the reduction routine to search for the details of that point (eg coordinates) to allow for the reduction of further measurements. The order in which this searching takes place is as follows:

First search the Control model (if it exists):

1. A search is made of the control model for a string whose name is the same as the specified *point name*. If a string is found, the first point of the string is used for the (x,y,z) co-ordinates.
2. A search is made of the control model for a vertex of a string whose point id is the same as the specified *point name*. If a vertex is found its (x,y,z) co-ordinates are used.
3. If only a *point id* was specified, a search is made of the control model for a vertex of a string whose point id is the same as the specified *point id*. If a vertex is found its (x,y,z) co-ordinates are used.

Next search the already entered directly entered co-ordinates (DEC) in the field file:

4. A search is made of previously entered directly entered co-ordinates in the field file for a directly entered co-ordinate whose *point name* is the same as the specified *point name*. If a

DEC is found, its (x,y,z) co-ordinates are used.

5. A search is made of previously entered directly entered co-ordinates in the field file for a directly entered co-ordinate whose *point id* is the same as the specified *point name*. If a DEC is found, its (x,y,z) co-ordinates are used.
6. A search is made of previously entered directly entered co-ordinates in the field file for a directly entered co-ordinate whose *point id* is the same as the specified *point id*. If a DEC is found, its (x,y,z) co-ordinates are used.

Next search the previous measurements in the field file:

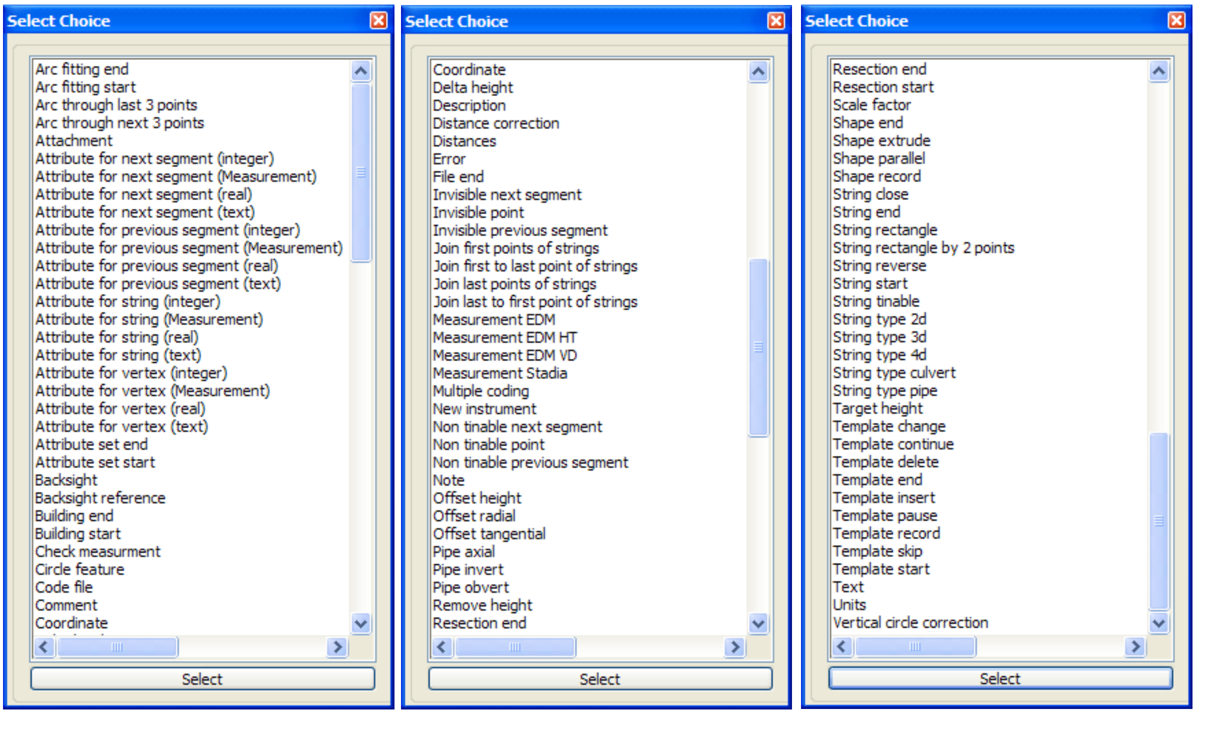
7. A search is made of previous measurements in the field file for a measurement whose *point name* is the same as the specified *point name*. If a measurement is found, its (x,y,z) co-ordinates are used.
8. A search is made of previous measurements in the field file for a measurement whose *point id* is the same as the specified *point name*. If a measurement is found, its (x,y,z) co-ordinates are used.
9. A search is made of previous measurements in the field file for a measurement whose *point id* is the same as the specified *point id*. If a measurement is found, its (x,y,z) co-ordinates are used.

Finally

10. If no match is found, the user will be prompted for the details of the previously undefined point. The user is asked to type in the (x,y,z) co-ordinates in the **Define Station coordinate** panel. If a model is specified in the **Add to model** field of the panel, then a new one point super string is created with the name *point name*, and as the vertex text for the point, the **Station label prefix** field value followed by *point name*

Field Data commands

The list of field commands in the pop-up from **Insert** or **Batch Add** are:



For a summary of the field file op codes see the section 36.10.5 Summary of 12d Field File Op Codes

For details on each option in the pop-up:

*Circle feature (*opcode 18)	Feature (opcode 18)
*Code file (*opcode 119)	Code file (opcode 119)
*Comment (*opcode -2)	Comment (opcode -2)
*Coordinate (*opcode 2)	Coordinate (opcode 2)
*Delta height (*opcode 28)	Height Or Depth (opcode 28)
*Description (*opcode 1)	Job Data (opcode 1)
Distance correction (opcode 127)	Distance correction (opcode 127)
*Distances (*opcode 49)	Distances (opcode 49)
*Error (*opcode -1)	Error (opcode -1)
*File end (*opcode 99)	End File (opcode 99)
*Invisible next segment (*opcode 108)	Invisible (opcodes 107, 108, 109)
*Invisible point (*opcode 109)	Invisible (opcodes 107, 108, 109)
*Invisible previous segment (*opcode 107)	Invisible (opcodes 107, 108, 109)
*Join first points of strings (*opcode 23)	Strings Join (opcodes 21 to 24)
*Join first to last point of strings (*opcode 22)	Strings Join (opcodes 21 to 24)
*Join last points of strings (*opcode 21)	Strings Join (opcodes 21 to 24)
*Join last to first point of strings (*opcode 24)	Strings Join (opcodes 21 to 24)
*Measure EDM (*opcode 7)	EDM Measurement (opcode 7)
*Measure EDM HT (*opcode 11)	EDM Measurement (HA,HD,HT) (opcode 11)
*Measure EDM VD (*opcode 12)	Edm measurement (HA,HD,Diff HT) (opcode 12)
*Measure Stadia (*opcode 10)	Stadia Measurement (opcode 10)
*Multiple coding (*opcode 16)	Multiple Coding (opcode 16)
*New instrument (*opcode 3)	New Instrument (opcode 3)
*Non tinable next segment (*opcode 38)	Tinable (opcodes 38, 39, 40)
*Non tinable point (*opcode 40)	Tinable (opcodes 38, 39, 40)
*Non tinable previous segment (*opcode 39)	Tinable (opcodes 38, 39, 40)
*Note (*opcode 29)	Note (opcode 29)
*Offset height (*opcode 44)	Offset Measurement (opcodes 42, 43, 44)
*Offset radial (*opcode 42)	Offset Measurement (opcodes 42, 43, 44)

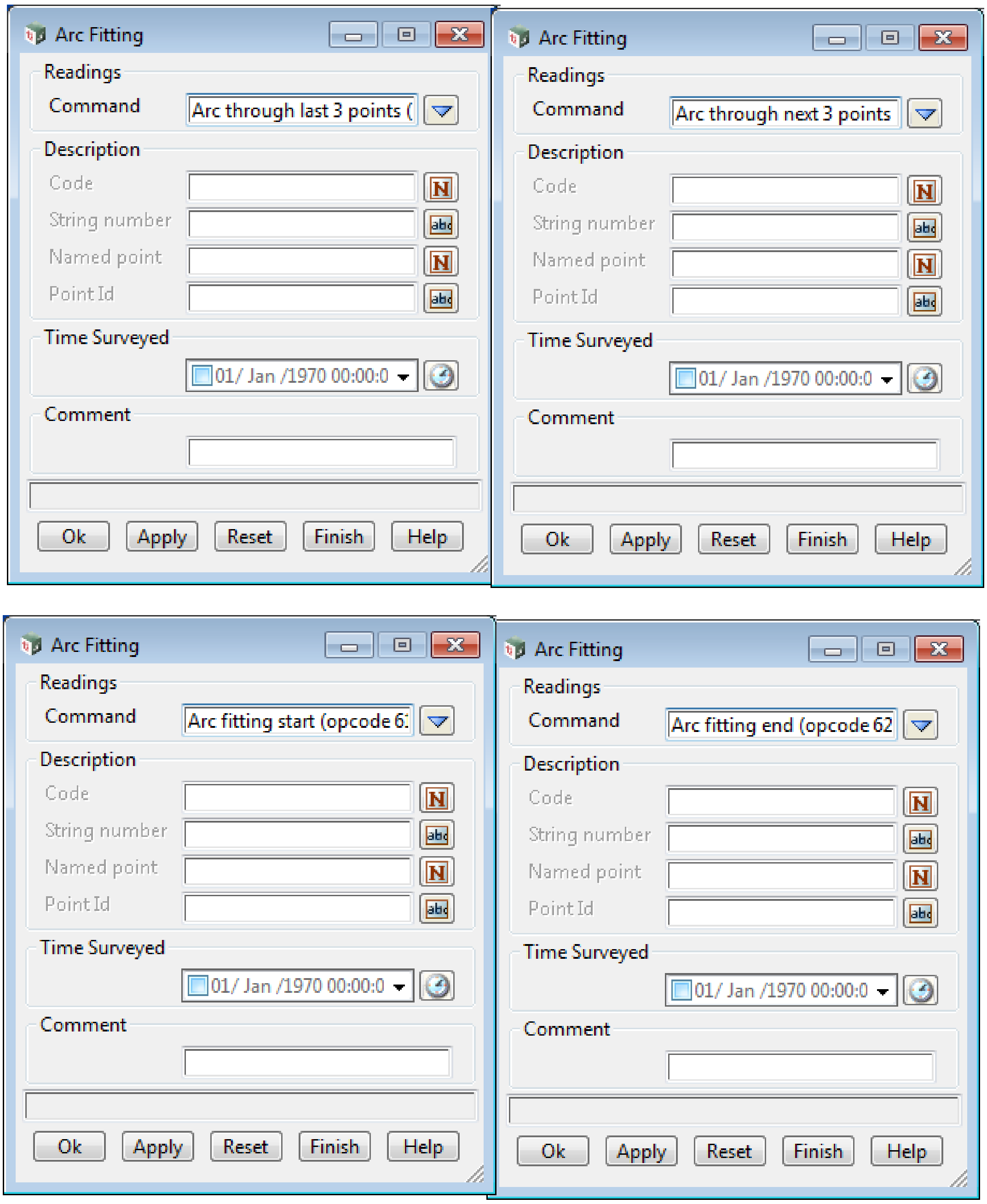
*Offset tangential (*opcode 43)	Offset Measurement (opcodes 42, 43, 44)
PPM Correction (opcode 131)	PPM Correction (opcode 131)
*Pipe axial (*opcode 81)	Pipe Justification (opcodes 80, 81, 82)
*Pipe invert (*opcode 80)	Pipe Justification (opcodes 80, 81, 82)
*Pipe obvert (*opcode 82)	Pipe Justification (opcodes 80, 81, 82)
*Remove height (*opcode 30)	Remove Height (opcode 30)
*Resection end (*opcode 129)	Resection end (opcode 129)
*Resection start (*opcode 128)	Resection Start (opcode 128)
*Scale factor (*opcode 9)	Scale Factor (opcode 9)
*Shape end (*opcode 84)	Shaping (opcodes 83 to 86)
*Shape extrude (*opcode 86)	Shaping (opcodes 83 to 86)
*Shape parallel (*opcode 85)	Shaping (opcodes 83 to 86)
*Shape record (*opcode 83)	Shaping (opcodes 83 to 86)
*String close (*opcode 20)	String Close (opcode 20)
*String end (*opcode 48)	String End (opcode 48)
*String rectangle (*opcode 45)	Rectangle (opcode 45)
String rectangle by 2 points *(*opcode 37)	Rectangle By 2 Points (opcode 37)
*String reverse (*opcode 19)	String Reverse (opcode 19)
*String start (*opcode 47)	String Start (opcode 47)
*String tinable (*opcode 46)	Breakline String (String tinable (opcode 46))
*String type 2d (*opcode 92)	String Type (opcodes 92, 93, 94)
*String type 3d (*opcode 93)	String Type (opcodes 92, 93, 94)
*String type 4d (*opcode 94)	String Type (opcodes 92, 93, 94)
*String type culvert (*opcode 96)	Culvert (opcode 96)
*String type pipe (*opcode 95)	Pipe Diameter (opcode 95)
*Target height (*opcode 5)	Target Height (opcode 5)
*Template change (*opcode 59)	Templating (opcodes 51 to 59)
*Template continue (*opcode 54)	Templating (opcodes 51 to 59)
*Template delete (*opcode 57)	Templating (opcodes 51 to 59)

*Template end (*opcode 52)	Templating (opcodes 51 to 59)
*Template insert (*opcode 58)	Templating (opcodes 51 to 59)
*Template pause (*opcode 53)	Templating (opcodes 51 to 59)
*Template record (*opcode 55)	Templating (opcodes 51 to 59)
*Template skip (*opcode 56)	Templating (opcodes 51 to 59)
*Template start (*opcode 51)	Templating (opcodes 51 to 59)
*Text (*opcode 41)	Additional Text For Point (opcode 41)
*Vertical circle correction (*opcode 15)	Vertical Circle Correction (opcode 15)

Arc Fitting (opcodes 17, 60, 61, 62)

Fitting arcs through measurements.

See 36.3.9 Arcs Through Points in Appendix 36 12d Survey Guide .



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Command	choice box		Arc through last 3 points (opcode 17) Arc through next 3 points (opcode 60) Arc fitting start (opcode 61) Arc fitting end (opcode 62)

for Arc through last 3 pts

Named point is not used.

If no *feature code, string number, named point or point id* is given, then the current measurement point and the two previous points with the same feature code and string number as the current measurement point, are joined by an arc. If there is less than three such points, no arc is fitted.

If the *feature code* and *string number* exist*,* the last three previous measurement points of the same *feature code* and *string number* are joined by an arc. If the current measurement point has that feature code and string number, then it is the third of the three points used. If there is less than three points, no arc is fitted.

If a *point id* exists, then the *feature code* and *string number* are taken from the previous measurement point **with** that point id. That point and two measurement points previous to the predefined point of the same *feature code* and *string number*, are joined by an arc. If there is less than three points, no arc is fitted

for Arc through next 3 pts

Named point is not used.

If no feature *code, string number, named point or point id is* given, an arc is inserted through the current measurement point and the next two measured points with the same feature code and string number as the current measurement point. If there is less than three points, no arc is fitted.

If a feature *code, string number or point id* exists, then either the *feature code* and/or *string number* and/or the *point id* section of the *point description* can be used.

If the *feature code* or *string number* exist, a search is made for a previously defined measurement with the same feature code or string number. An arc is inserted through this previous measurement and the next two measured points following this previous measurement with the same feature code and string number. If the current point has that feature code and string number, then it is the first of the three points. If there is less than three points, no arc fitted.

If the *point id exists,* then the *feature code* and *string number* are taken from the previous measurement point **with** that point id, and an arc is inserted through that point and the next two measurement points with the same *feature code* and *string number.* If there is less than three points, no arc is fitted

for Start arc fitting

Named point is not used.

If no feature *code, string number, named point or point id* is given, arcs are inserted through the following sets of measurement points with the same feature code and string number as the current measurement point. The current measurement point is the first of the points.

The arcs are fitted as follows - the first arc is fitted through points one, two and three, the next arc through points three, four and five etc. If the current point has that feature code and string number, then it is the first of the points. If there is less than three points, then no arc is fitted.

If the *feature code* and *string number exist,* a search is made for a previously defined measurement with the same feature code or string number. An arc is inserted through the following measured points with the same feature code and string number as given. If the current point has that feature code and string number, then it is the first of the points.

If the *point id* exists, then the *feature code* and *string number* are taken from the previous measurement point **with** that point id, and arcs are inserted through that point and the following measured points with the same *feature code* and *string number.*

for End arc fitting

Named point is not used.

If no feature *code, string number, named point or point id* is given, the fitting of arcs through the current string is stopped. The current measurement point is the last of the points used in the arc fitting.

If the *feature code* and *string number* exist, then the fitting of arcs through the points of the previous string with the same *feature code* and *string number* is stopped. If the current measurement point has that feature code and string number, then it is the last point used in the arc fitting.

If the *point id* from the *point description* exists, then the point with that point id is the last point used in the arc fitting.

If ***12d Model*** encounters an *End Arcs (62*) but no *Start Arcs through sets of three points* (61) command for the string, then a *Start Arcs through sets of three points* (61) is assumed to apply at the beginning of the string and hence arc fitting will be applied to the entire string.

Code

feature code. See the description of the panel field **Arc mode** *for usage*

String number

string number. See the description of the panel field **Arc mode** *for usage*

Named point

not used

Point id

point id. See the description of the panel field **Arc mode** *for usage*

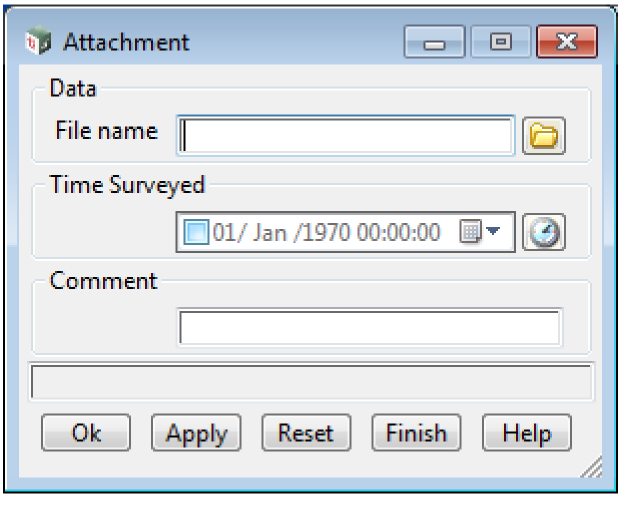
Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Attachment (opcode 126)

Attach a file to the measurement.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

File name

file to attach to the measurement

Time surveyed

time when the command (op code) was created

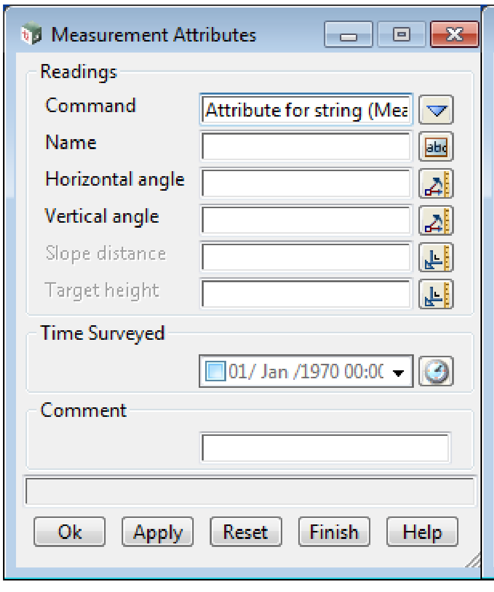
Comment

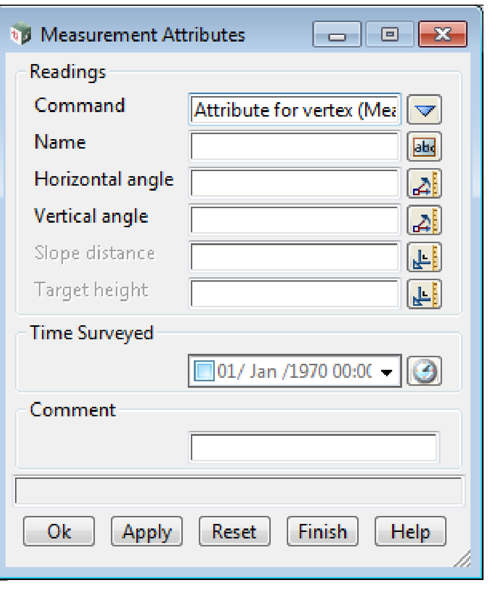
comment for the command within the field file.

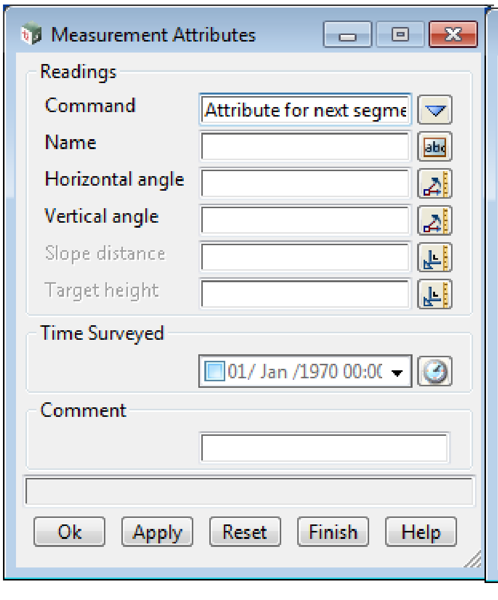
OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

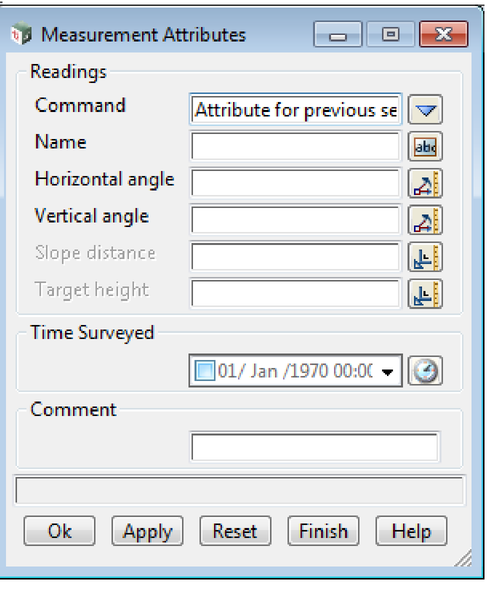
Measurement Attributes (Opcode 120 to 123)

This section of documentation is a work in progress and will be updated in subsequent releases.









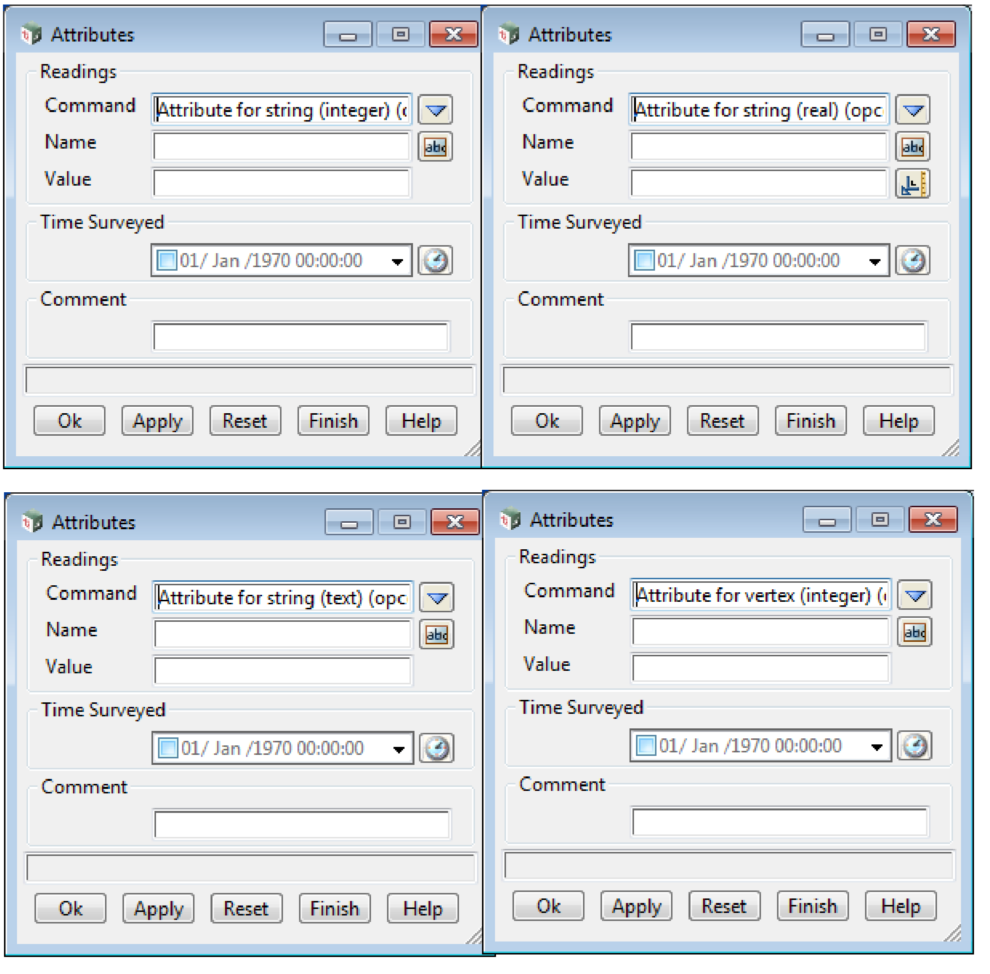
The fields and buttons used in this panel have the following functions.

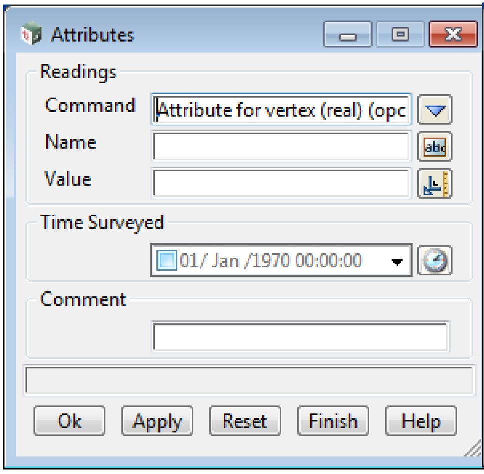
Field Description Type Defaults Pop-Up

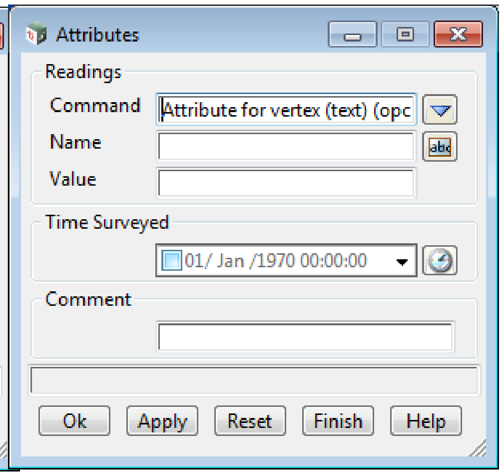
Attribute for string (Measurement) (opcode 120)
Attribute for vertex (Measurement) (opcode 121)
Attribute for next segment (Measurement) (opcode 122)
Attribute for previous segment (Measurement) (opcode 123)

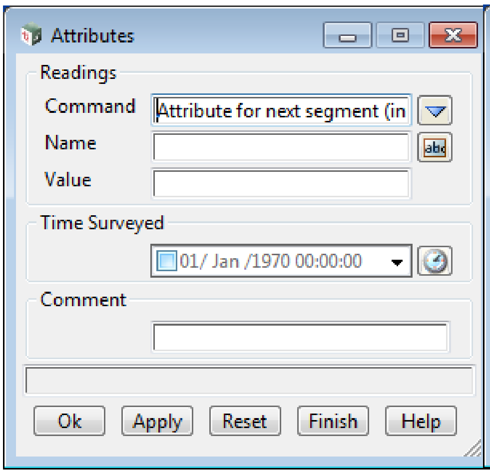
Attributes (opcode 68 to79)

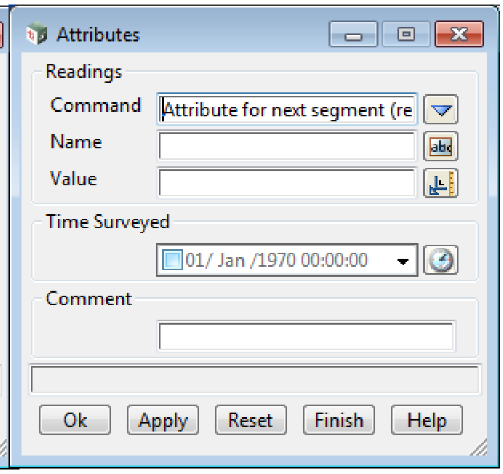
Create attributes of type measurement, integer, real or text, to string, vertices or segments.

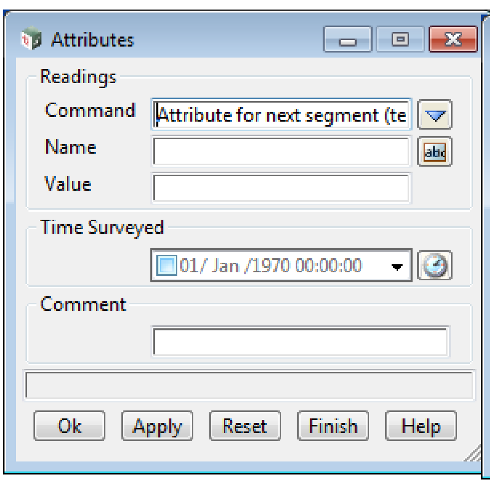


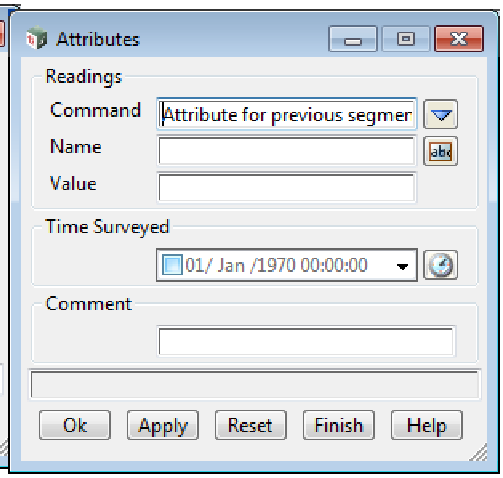

Attributes
Readings
Command
Attribute for vertex (real) (opc
Name
Value
Time Surveyed
01/ Jan /1970 00:00:00
Comment
Ok
Apply
Reset
Finish
Help

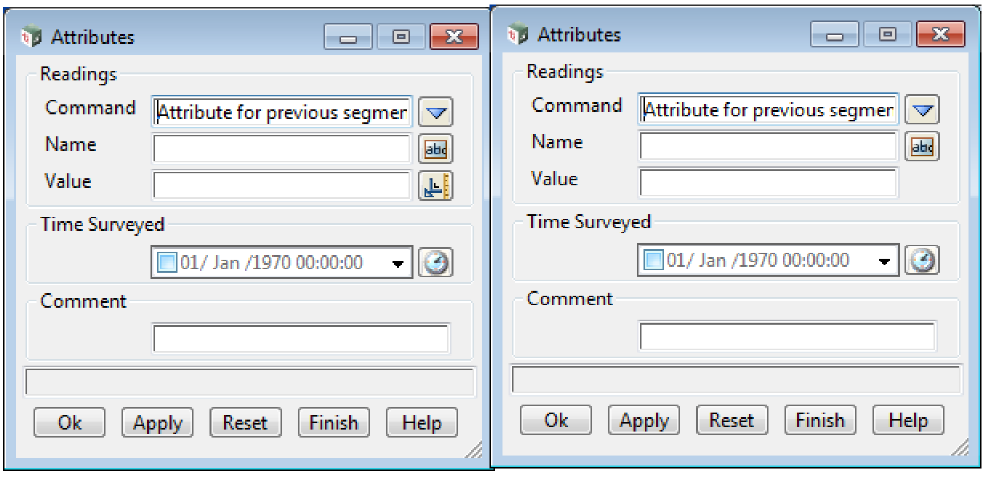

Attributes
Readings
Command
Attribute for vertex (text) (opc
Name
Value
Time Surveyed
01/ Jan /1970 00:00:00
Comment
Ok
Apply
Reset
Finish
Help


Attributes
Readings
Command
Attribute for next segment (in
Name
Value
Time Surveyed
01/ Jan /1970 00:00:00
Comment
Ok
Apply
Reset
Finish
Help


Attributes
Readings
Command
Attribute for next segment (re
Name
Value
Time Surveyed
01/ Jan /1970 00:00:00
Comment
Ok
Apply
Reset
Finish
Help


Attributes
Readings
Command
Attribute for next segment (te
Name
Value
Time Surveyed
01/ Jan /1970 00:00:00
Comment
Ok
Apply
Reset
Finish
Help


Attributes
Readings
Command
Attribute for previous segmer
Name
Value
Time Surveyed
01/ Jan /1970 00:00:00
Comment
Ok
Apply
Reset
Finish
Help



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Command	choice box		Attribute for string (integer) (opcode 68) Attribute for string (real) (opcode 69) Attribute for string (text) (opcode 70) Attribute for vertex (integer) (opcode 71)) Attribute for vertex (real) (opcode 72)) Attribute for vertex (text) (opcode 73) Attribute for next segment (integer) (opcode 74) Attribute for next segment (real) (opcode 75)) Attribute for next segment (text) (opcode 76)) Attribute for previous segment (integer) (opcode 77)) Attribute for previous segment (real) (opcode 78) Attribute for previous segment (text) (opcode 79)

for String (integer)

Add an user defined integer attribute to the current string.

for String (real)

Add a real (floating point) user defined attribute to the current string.

for String (text)

Add a text user defined attribute to the current string.

for Vertex (integer)

Add an integer user defined attribute to the current measurement point.

for Vertex (real)

Add a real (floating point) user defined attribute to the current measurement point.

for Vertex (text)

Add a text user defined attribute to the current measurement point.

for Next segment (integer)

Add an integer user defined attribute to the next segment from the current measurement point.

for Next segment (real)

Add a real (floating point) user defined attribute to the next segment from the current measurement point.

for Next segment (text)

Add a text user defined attribute to the next segment from the current measurement point.

for Previous segment (integer)

Add an integer user defined attribute to the previous segment for the current measurement point.

for Previous segment (real)

Add a real (floating point) user defined attribute to the previous segment for the current measurement point.

for Previous segment (text)

Add a text user defined attribute to the previous segment for the current measurement point.

Name

The name of the attribute.

Value

The attribute data in the form specified by the mode.

Time surveyed

time when the command (op code) was created

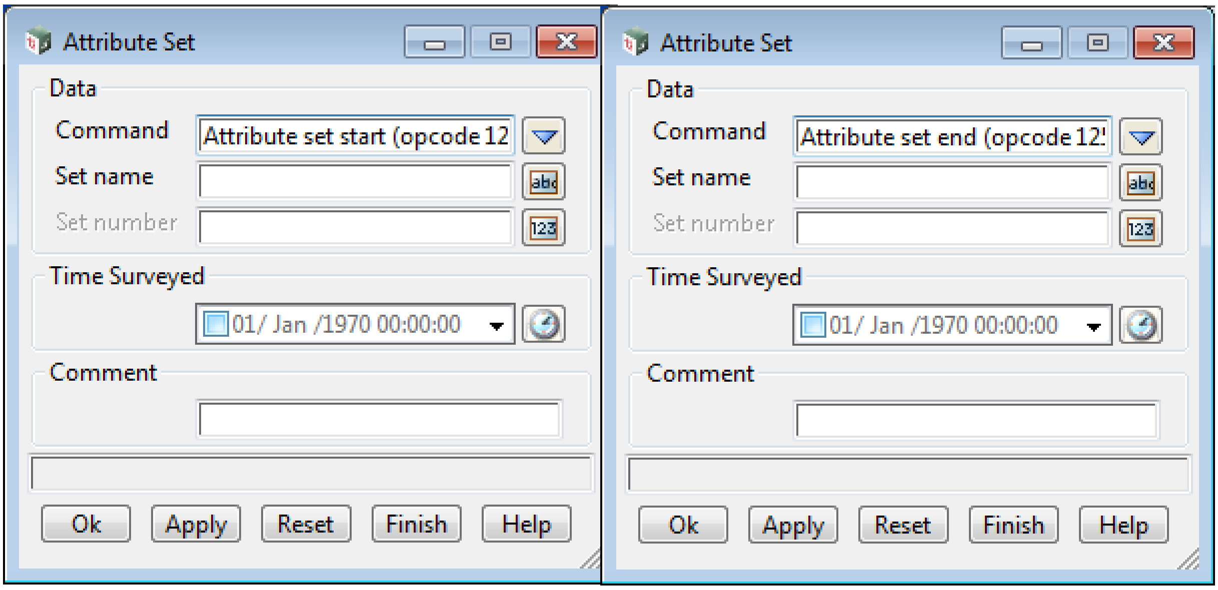
Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Attribute Set (opcode 124 and 125)

This section of documentation is a work in progress and will be updated in subsequent releases.



The fields and buttons used in this panel have the following functions.

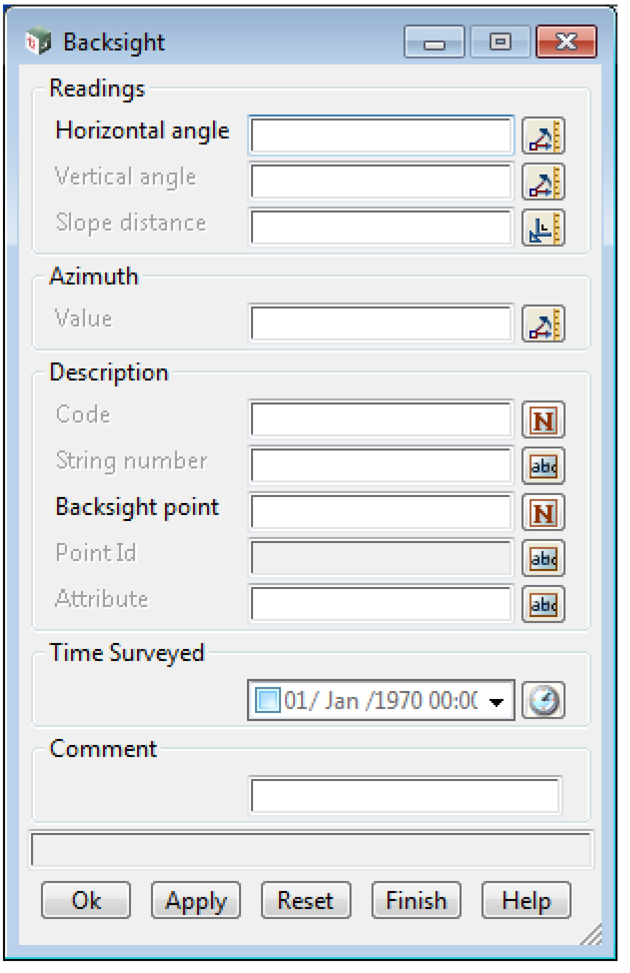
Field Description	Type	Defaults	Pop-Up
			Attribute set start (opcode 124) Attribute set end (opcode 125)

Backsight (opcode 4)

Denote the measurement as a backsight measurement.

Individual backsight measurements can be entered, or they can consist of pairs of Face1/Face2 measurements.

To enter a Face1/Face2 measurement, the backsight command should be entered twice, with the appropriate values entered into the panel each time.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Horizontal angle

the horizontal angle to the backsight in dd.mmss format.

Vertical angle

the vertical angle to the backsight in dd.mmss format. If the value is in the range 0 -180 degrees the measurement is considered a Face 1 measurement, and for measurements in the range 180-360 degrees they are considered Face2.

Slope distance

the slope distance to the measurement to the backsight. If a pair of face1/face2 measurements exist, the mean value of the slope distance is used for reduction purposes.

Value

the azimuth to the backsight in dd.mmss format. This allows backsights to be specified by azimuth only. In the case of a differing azimuth and horizontal angle, a swing will be computed by the subtraction of the azimuth value and the horizontal angle.

Code

the feature code of the measurement.

String number

the string number of the measurement.

Backsight point

point name.of the backsight point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search

Point id

point id.of the backsight point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search*. If a new backsight entry is inserted into the file, this field will be non-editable since only the backsight point is required.*

Attribute

Time surveyed

time when the command (op code) was created

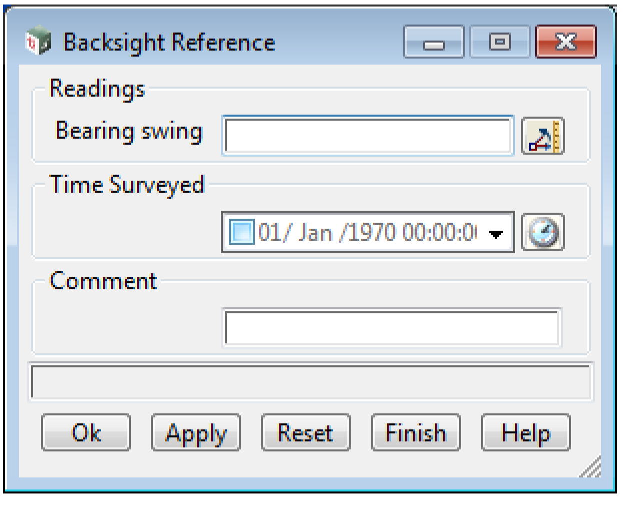
Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Backsight Reference (opcode 50)

Specify an angle (measured positive, clockwise from north) to swing the following measurements by.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Bearing swing

the angle (measured positive, clockwise from north) to swing following measurements. It is equivalent to the bearing datum difference. Note, this swing is only current for measurements following the command and for that particular station setup. It also negates any previous swings calculated from backsight measurements (bearing datum difference) for that station setup. Therefore, measurements after the command will use the new swing value. The direction of the swing is positive in the clockwise direction

Time surveyed

time when the command (op code) was created

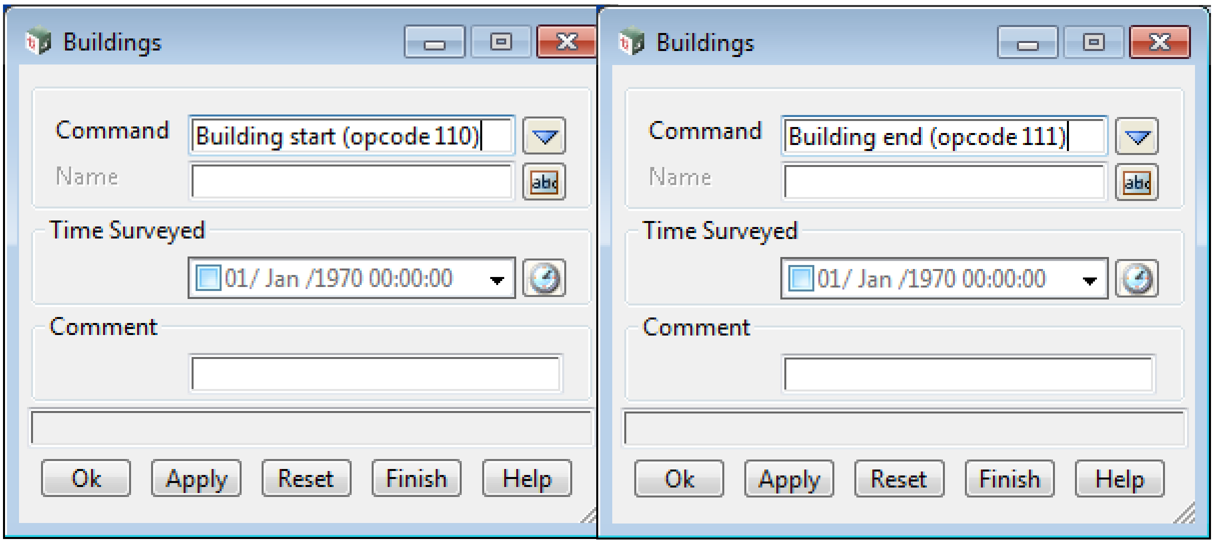
Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Buildings (opcodes 110, 111)

Create a building face.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Command			Building end (opcode 111) Building start (opcode 110)

for Building end

If no *name exist*s, the current building face observation set is finished (including the current measurement point).

for Buildings start

Start recording a building face with the given name. If *name* is non-blank, then the default building face is defined by the *feature code* and *string number.*

The following measurements until a *Finish* code (111), are stored as the building face. There is no limit to the number of points in a building face.

Name

Name has the meaning as outlined above in Building end/start definition.

Time surveyed

time when the command (op code) was created

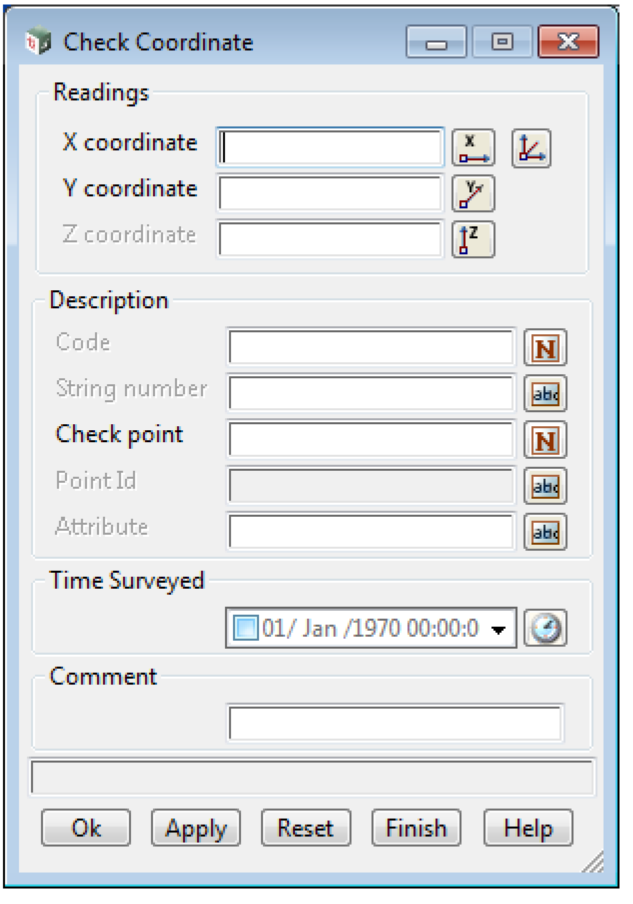
Comment

insert a comment within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Check Coordinate (opcode 14)

This section of documentation is a work in progress and will be updated in subsequent releases

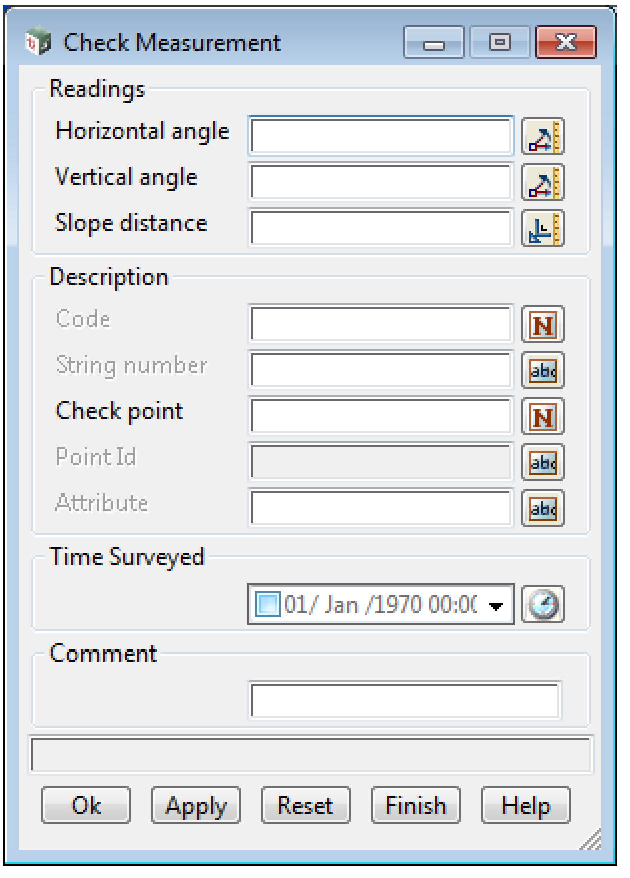


Check Measurement (opcode 6)

Denote the measurement as a check measurement.

Individual check measurements can be entered, or they can consist of pairs of Face1/Face2 measurements.

To enter a Face1/Face2 measurement the check command should be entered twice, with the appropriate values entered into the panel each time.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Horizontal angle

the horizontal angle to the check point in dd.mmss format.

Vertical angle

the vertical angle to the check point in dd.mmss format. If the value is in the range 0 -180 degrees the measurement is considered a Face 1 measurement, and for measurements in the range 180-360 degrees they are considered Face2.

Slope distance

the slope distance to the measurement to the check point. If a pair of face1/face2 measurements exist, the mean value of the slope distance is used for reduction purposes.

Code

the feature code of the measurement.

String number

the string number of the measurement.

Check point

point name.of the check point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search

Point id

point id.of the check point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point

search . *If a new check point entry is inserted into the file, this field will be non-editable since only the check point is required.*

Attribute

Time surveyed

time when the command (op code) was created

Comment

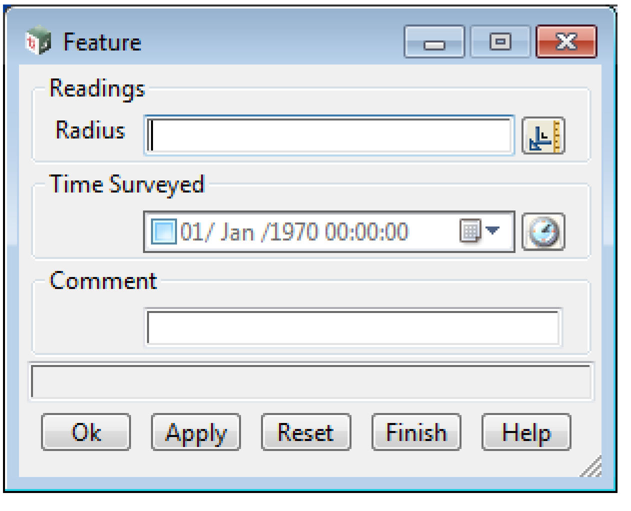
comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Feature (opcode 18)

Create a circle of a given radius around the current measurement point.

See 36.3.7 Feature *in Appendix* 36 12d Survey Guide .



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Radius

the radius of the circle to be drawn around the current measurement point.

Time surveyed

time when the command (op code) was created

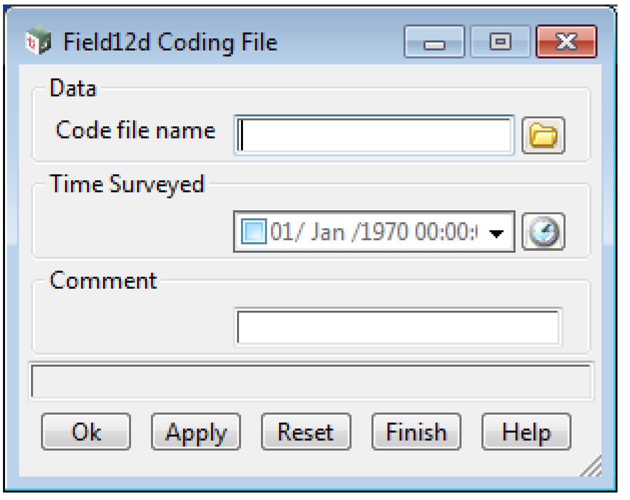
Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

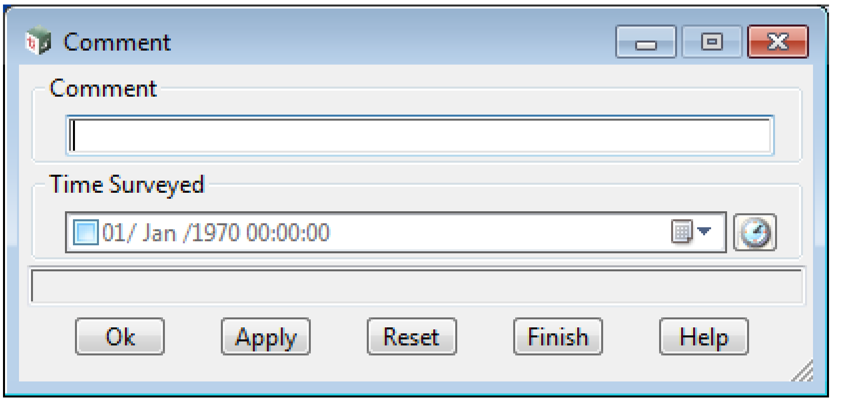
Code file (opcode 119)

This section of documentation is a work in progress and will be updated in subsequent releases.



Comment (opcode -2)

Insert a comment in the field file.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Comment

the comment to insert into the field file.

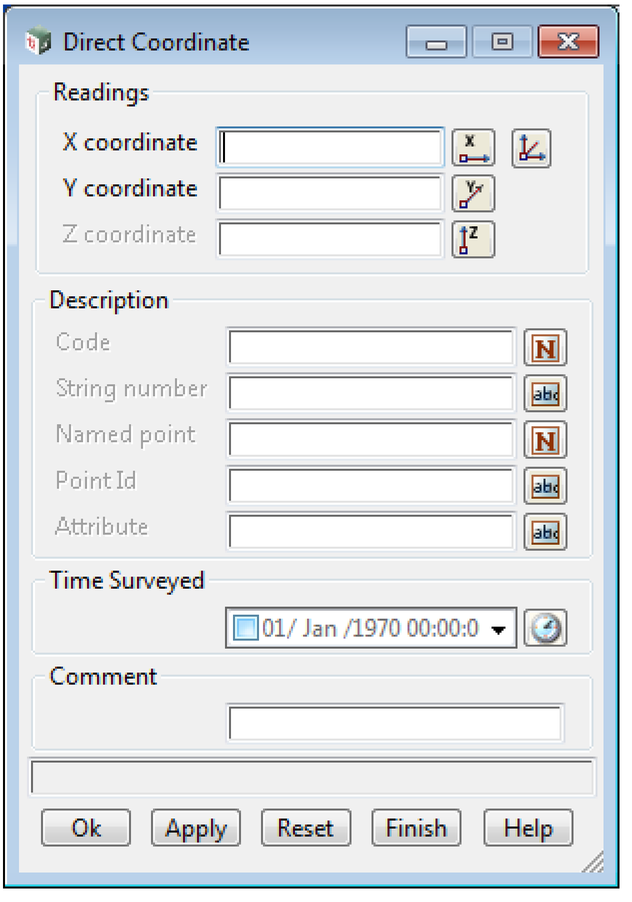
Time surveyed

time when the command (op code) was created

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Coordinate (opcode 2)

Create a point with specified coordinates.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
X coordinate	double box		

the x coordinate value of the nominated point.

Y coordinate double box

the y coordinate value of the nominated point.

Z coordinate double box

the z coordinate value of the nominated point.

Code

the feature code of the nominated point.

String number

the string number of the nominated point.

Named point

point name of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search

Point id

point id.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search. *If a new check point entry is inserted into the file, this field will be non-editable since only the check point is required.*

Attribute

Time surveyed

time when the command (op code) was created

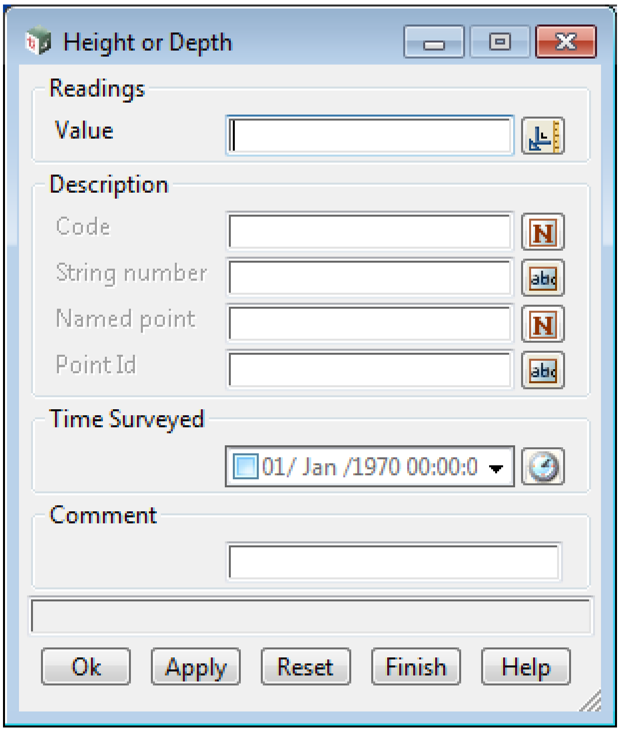
Comment

insert a comment within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Height Or Depth (opcode 28)

Note - this option ***does not alter heights***. It just adds a text comment of the given value.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Value

value is converted to text and added to the string name as specified in the description fields.

Code, String number, Named point

Named point is not used.

If no feature *code, string number, named point or point id* is given, the value is added to all string names with the same code and string number as the current measurement. This applies to the entire field file.

If the *feature code* and *string number* exist, then the value is added to all string names with the same code and string number. This applies to the entire field file.

If the *point id* exists, then the value is added to all string names with the same code and string number as the point defined by the point id. This applies to the entire field file.

Time surveyed

time when the command (op code) was created

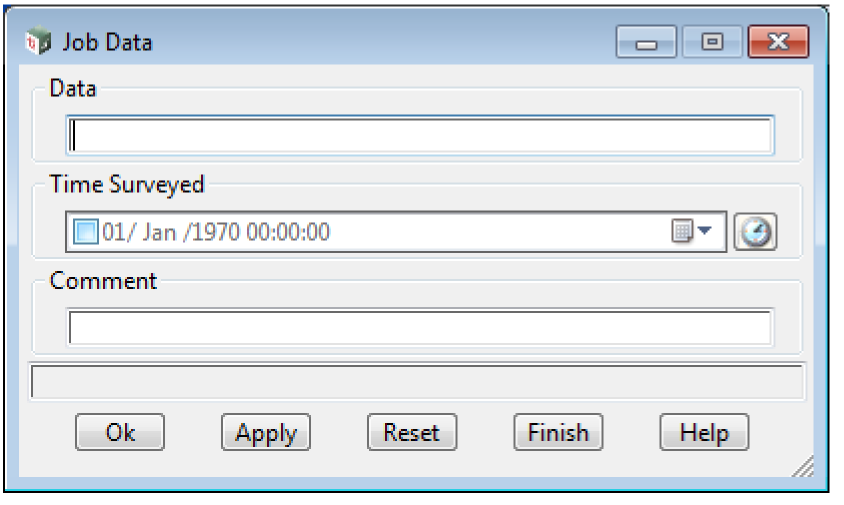
Comment

insert a comment within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Job Data (opcode 1)

Enter text to be added as header information



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Data

extra text to be added as header information

Time surveyed

time when the command (op code) was created

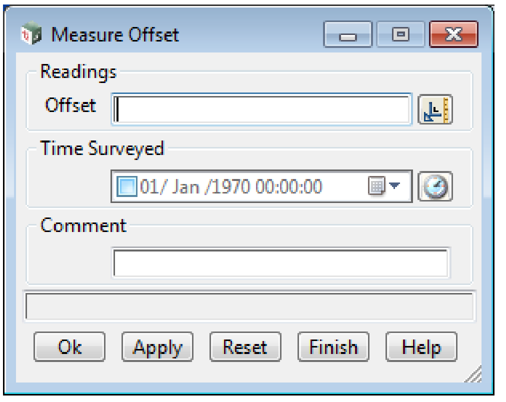
Comment

insert a comment within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

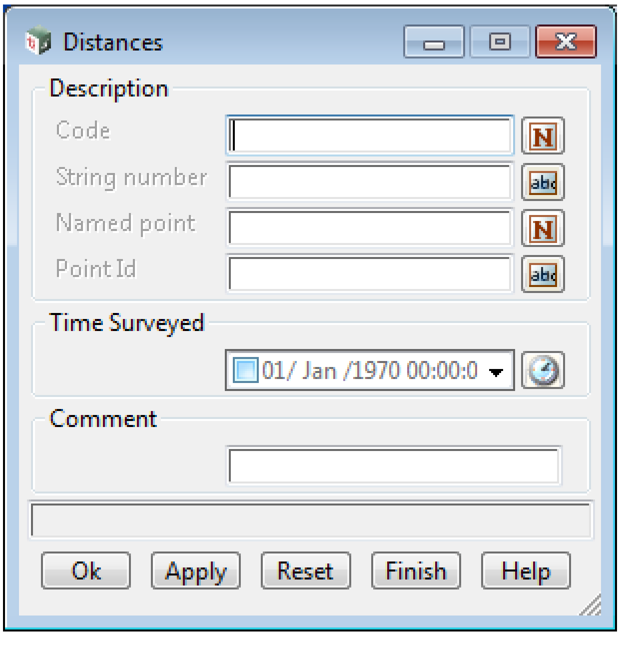
Distance correction (opcode 127)

This section of documentation is a work in progress and will be updated in subsequent releases.



Distances (opcode 49)

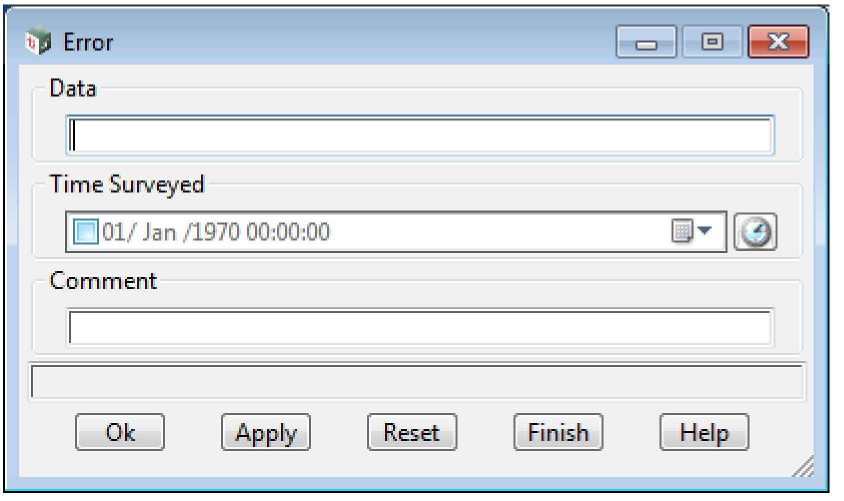
This section of documentation is a work in progress and will be updated in subsequent releases.



Error (opcode -1)

Enter text to be added as error information.

If the field file was reduced from a raw data collector file, any records that can't be parsed correctly will be made into an error comment.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Data

extra text to be added as error information. If the field file was reduced from a raw data collector file, any records that can't be parsed correctly will have an error comment.

Time surveyed

time when the command (op code) was created

Comment

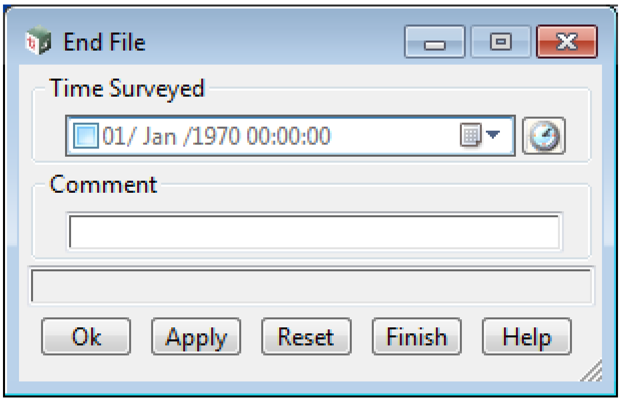
insert a comment within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

End File (opcode 99)

Command to stop the processing of the 12d field file at this line.

This is useful for debugging errors.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Time surveyed

time when the command (op code) was created

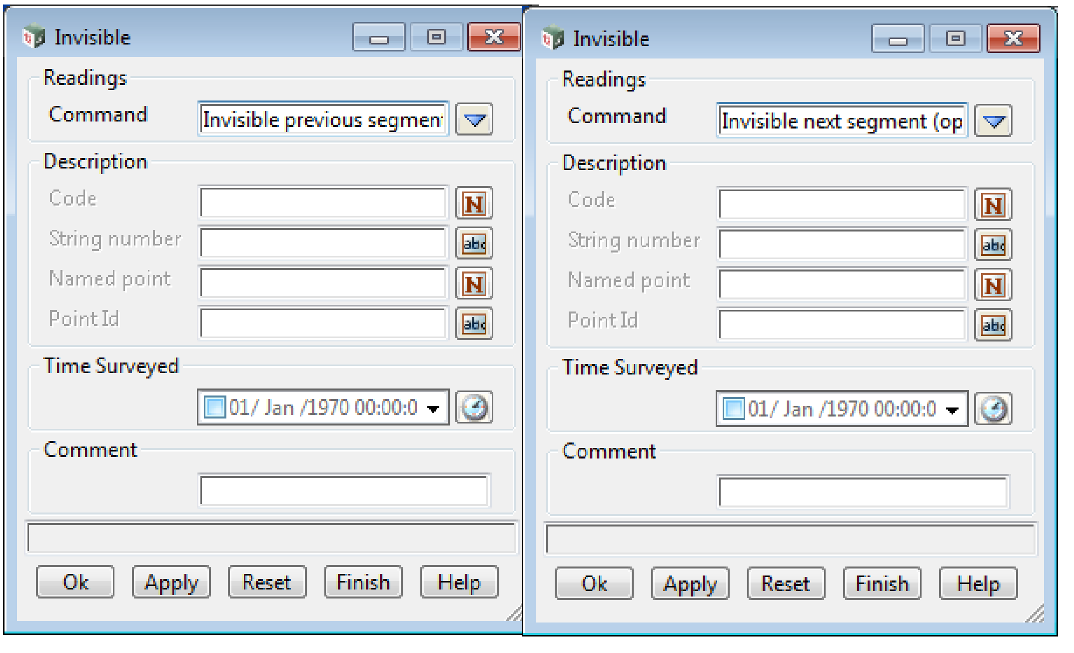
Comment

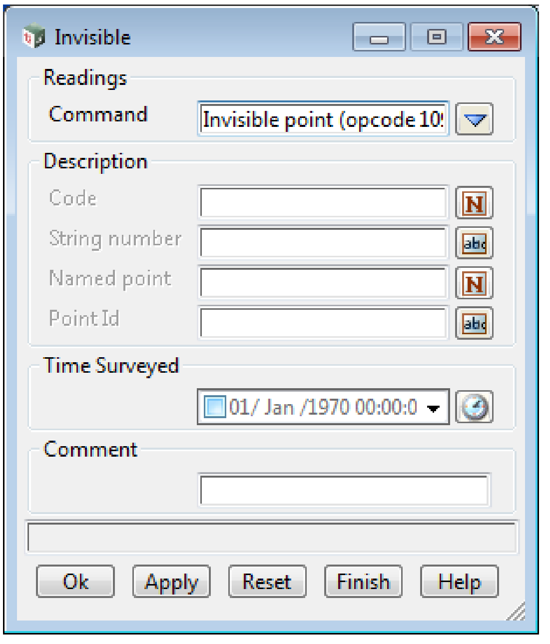
Stop processing the 12d field file at this line. Useful for debugging errors.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons .

Invisible (opcodes 107, 108, 109)

Make segments or vertices invisible.





The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Command	choice box		Invisible previous segment (opcode 107) Invisible next segment (opcode 108) Invisible point (opcode 109)

for Make the previous segment invisible

If no *description* is given, the previous segment containing the current measurement point is set to invisible.

If a *description* exists, then either the *feature code* and *string number* or the *point id* section of the *description* can be used.

If the *feature code* and *string number* exist, then the last segment of the previous string with that *feature code* and *string number* is set to invisible.

If the *point id* exists, then the segment containing the point with that point id as an end point, is set to invisible.

for Make the next segment invisible

If no *description* is given, the next segment containing the current measurement point as a starting point is set to invisible. That is, it will not be treated as a breakline in triangulations.

If a *description* exists, then either the *feature code* and *string number* or the *point id* section of the *description* can be used.

If the *feature code* and *string number* exist, then the segment that is created in the future from the last point of the previous string with that *feature code* and *string number* is set to invisible.

If the *point id* exists, then the segment containing the point with that point id as a start point, is set to invisible.

for Make a point invisible

If no *description* is given, the current measurement point is set to invisible. That is, it will not be included in triangulations.

If a *description* exists, then either the *feature code* and *string number* or the *point id* section of the *description* can be used.

If the *feature code* and *string number* exist, then the last point of the previous string with that *feature code* and *string number* is set to invisible.

If the *point id* exists, then the point with that point id is set to invisible.

Code

the feature code of the nominated point.

String number

the string number of the nominated point.

Named point

point name.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search

Point id

point id.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search *. If a new check point entry is inserted into the file, this field will be non-editable since only the check point is required.*

Time surveyed

time when the command (op code) was created

Comment

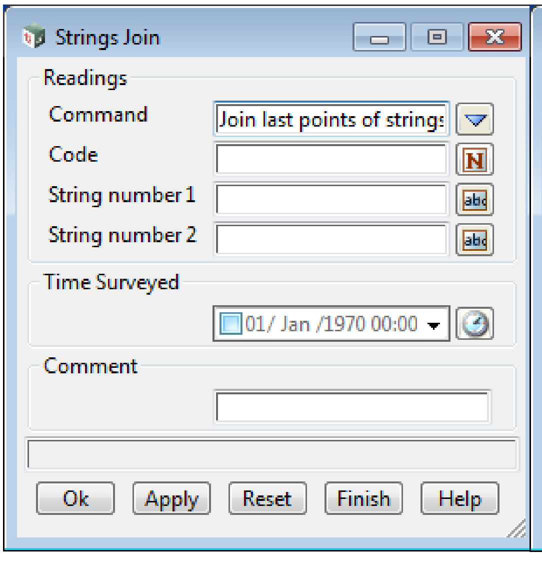
comment for the command within the field file.

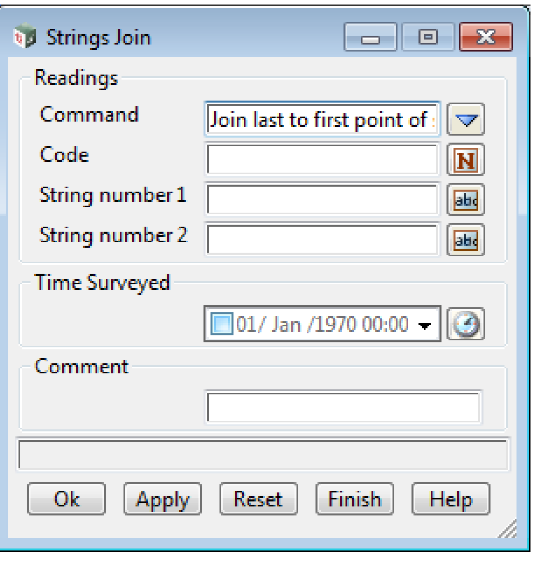
OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

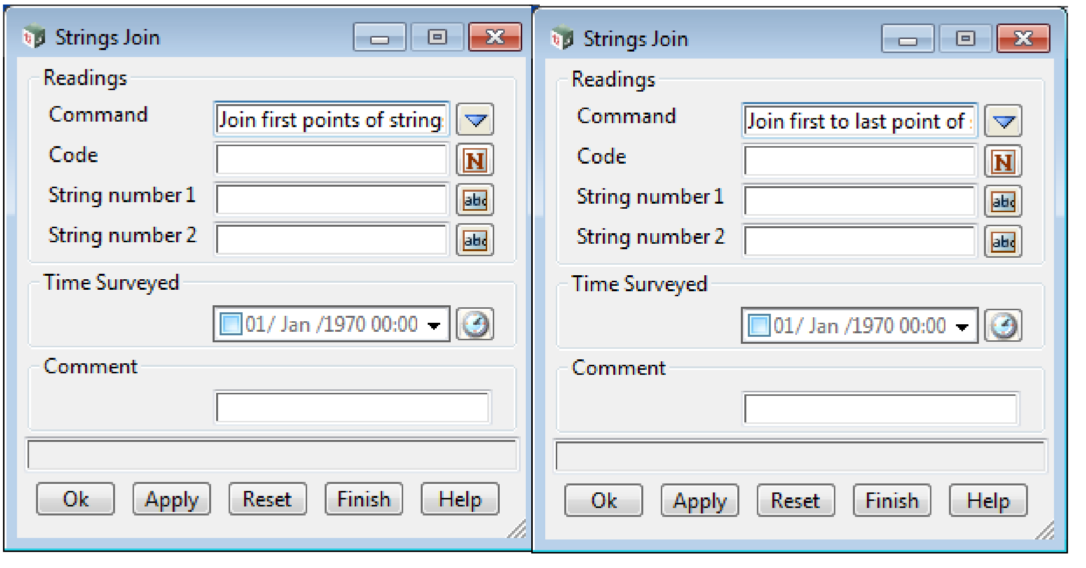
Strings Join (opcodes 21 to 24)

Join two strings together.

See 36.3.8 Joining Strings in Appendix 36 12d Survey Guide .







The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Command	choice box		Join last points of strings (opcode 21) Join first to last point of strings (opcode 22) Join first points of strings (opcode 23) Join last to first point of strings (opcode 24)

for Join last points of strings

In the final reduction, the last point of the string with the given *feature code* and *string_number_1* is joined to the last point of the string with given *feature code* and *string_number_2*. The created string has the given *feature code* (no string number is needed since it is the final phase of reduction when the string numbers are dropped).

for Join first to last point of strings

In the final reduction, the first point of the string with the given *feature code* and *string_number_1* is joined to the last point of the string with given *feature code* and *string_number_2*. The created string has the given *feature code* (no string number is needed since it is the final phase of reduction when the string numbers are dropped).

for Join first points of strings

In the final reduction, the first point of the string with the given *feature code* and *string number 1* is joined to the fist point of the string with given *feature code* and *string number 2*. The created string has the given *feature code* (no string number is needed since it is the final reduction when the string numbers are then dropped).

for Join last to first point of strings

In the final reduction, the last point of the string with the given *feature code* and *string_number_1* is joined to the first point of the string with given *feature code* and *string_number_2*. The created string has the given *feature code* (no string number is needed since it is the final phase of reduction when the string numbers are dropped).

Code

the feature code of the strings to be joined

String number 1

the string number of the first string

String number 2

the string number of the second string

Time surveyed

time when the command (op code) was created

Comment

insert a comment within the field file.

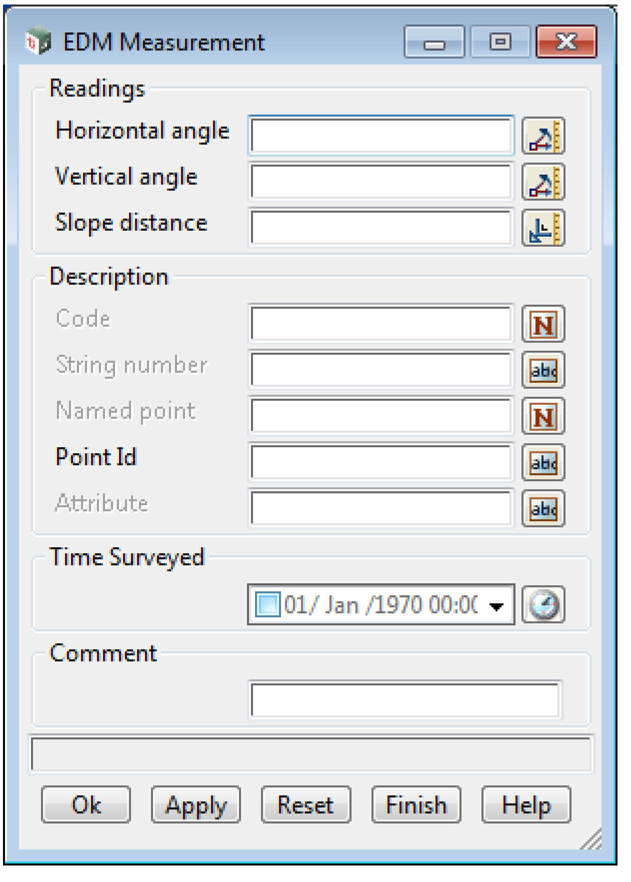
OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

EDM Measurement (opcode 7)

Store a measurement given by Horizontal angle, Vertical angle and Slope distance.

Individual measurements can be entered or they can consist of pairs of Face1/Face2 measurements.

To enter a Face1/Face2 measurement the command should be entered twice, with the appropriate values entered into the panel each time.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Horizontal angle

the horizontal angle to the point in dd.mmss format.

Vertical angle

the vertical angle to the point in dd.mmss format. If the value is in the range 0 -180 degrees the measurement is considered a Face 1 measurement, and for measurements in the range 180-360 degrees they are considered Face2.

Slope distance

the slope distance to the measurement to the point. If a pair of face1/face2 measurements exist, the mean value of the slope distance is used for reduction purposes.

Code

the feature code of the measurement.

String number

the string number of the measurement.

Named point

point name.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search

Point id

point id.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search *. If a new check point entry is inserted into the file, this field will be non-editable since only the check point is required.*

Attribute

Time surveyed

time when the command (op code) was created

Comment

comment for the command within the field file.

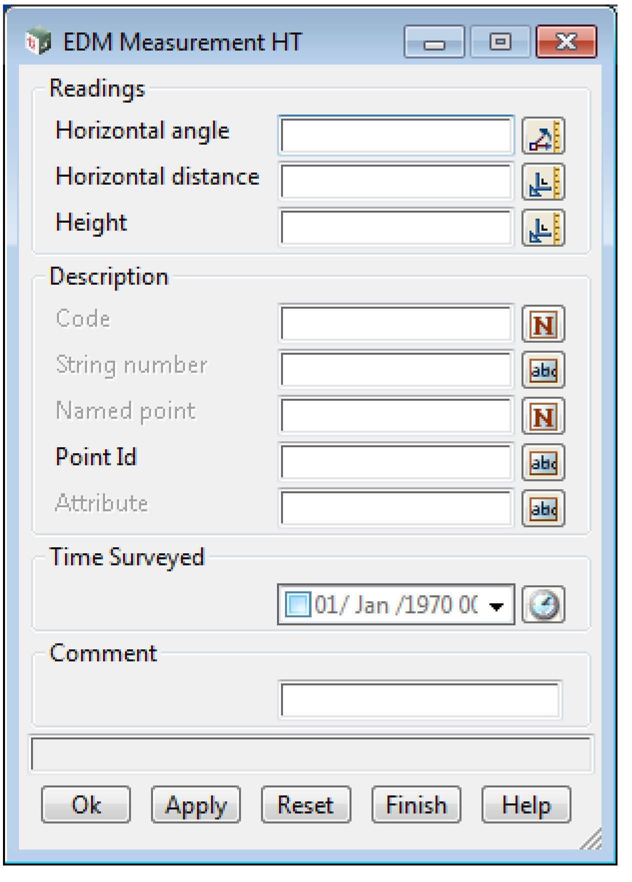
OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

EDM Measurement (HA,HD,HT) (opcode 11)

Store a measurement given by Horizontal angle, Horizontal distance and Height.

Individual measurements can be entered or they can consist of pairs of Face1/Face2 measurements.

To enter a Face1/Face2 measurement the command should be entered twice, with the appropriate values entered into the panel each time.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Horizontal angle

the horizontal angle to the point in dd.mmss format.

Horizontal distance

the reduced horizontal distance.

Height

the height of the observation point.

Code

the feature code of the measurement.

String number

the string number of the measurement.

Named point

point name.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search

Point id

point id.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search *. If a new check point entry is inserted into the file, this field will be non-editable since only the check point is required.*

Attribute

Time surveyed

time when the command (op code) was created

Comment

comment for the command within the field file.

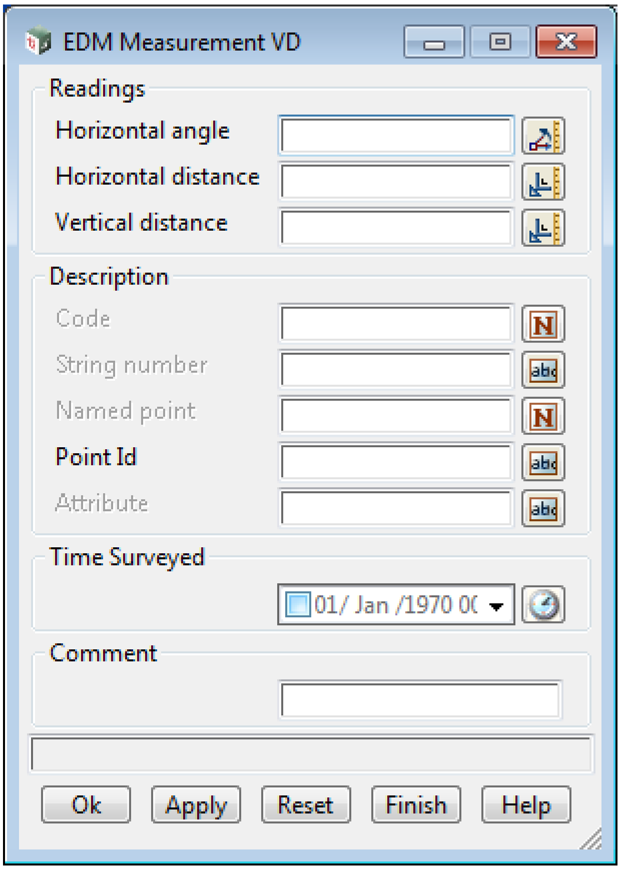
OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Edm measurement (HA,HD,Diff HT) (opcode 12)

Store a measurement given by Horizontal angle, Horizontal distance and Vertical distance.

Individual measurements can be entered or they can consist of pairs of Face1/Face2 measurements.

To enter a Face1/Face2 measurement the command should be entered twice, with the appropriate values entered into the panel each time.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Horizontal angle

the horizontal angle to the point in dd.mmss format.

Horizontal distance

the reduced horizontal distance.

Vertical distance

the change in height as measured from the collimation height of the instrument to the target point (usually centre of target).

Code

the feature code of the measurement.

String number

the string number of the measurement.

Named point

point name.of the point. This can be an integer, real, text or alphanumeric. For a more detailed *description of how the reduction finds/uses the appropriate point see the section* [Existing point search](#)

Point id

point id.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section [Existing point search](#)*. If a new check point entry is inserted into the file, this field will be non-editable since only the check point is required.*

Attribute

Time surveyed

time when the command (op code) was created

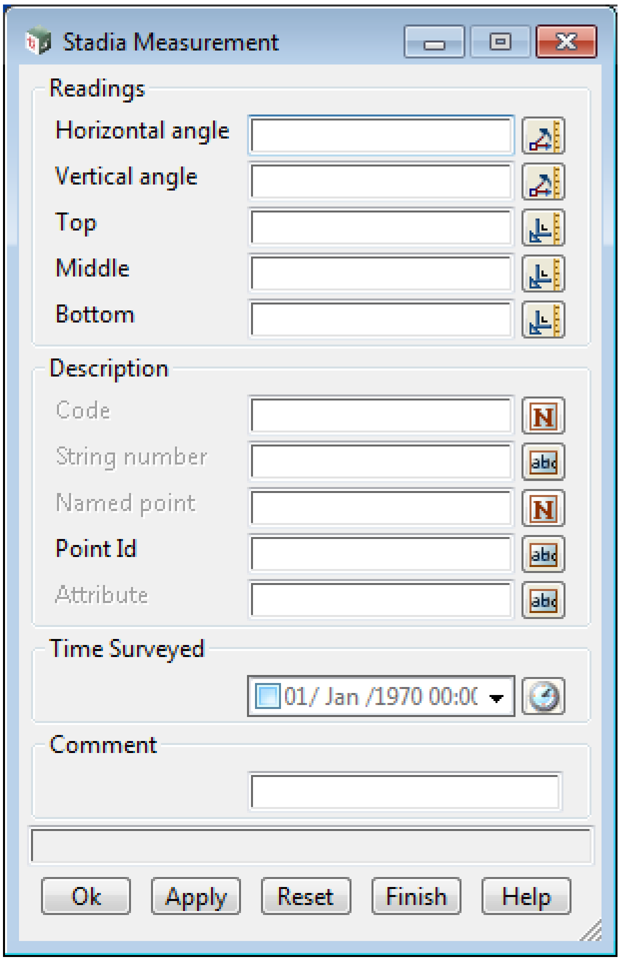
Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section [Panel buttons](#)

Stadia Measurement (opcode 10)

Store a measurement given by Horizontal angle, Horizontal distance and the Top, Middle and Bottom hair readings.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Horizontal angle

the horizontal angle to the point in dd.mmss format.

Vertical angle

the vertical angle to the point in dd.mmss format.

Top

the top hair reading.

Middle

the middle hair reading.

Bottom

the bottom hair reading.

Code

the feature code of the measurement.

String number

the string number of the measurement.

Named point

point name.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search

Point id

point id.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search*. If a new check point entry is inserted into the file, this field will be non-editable since only the check point is required.*

Attribute

Time surveyed

time when the command (op code) was created

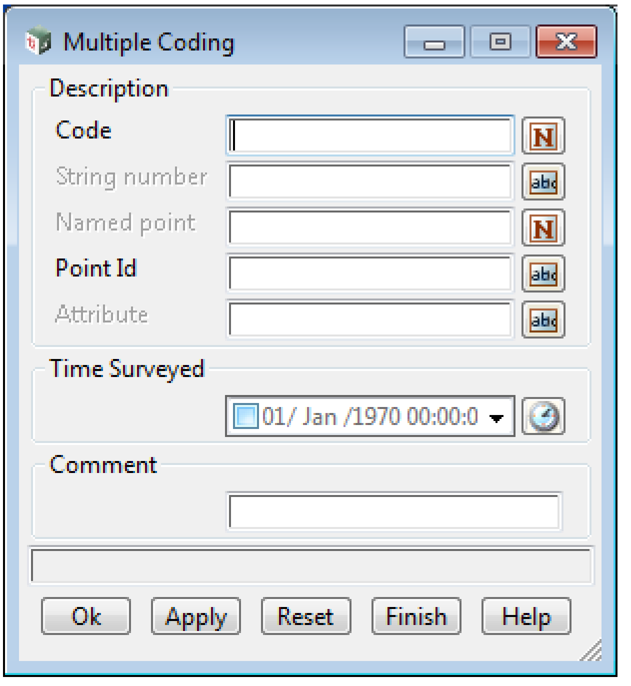
Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Multiple Coding (opcode 16)

Record a new point at the same position as the current measurement point but with possibly a different feature code and string number.



A new point is created at the same position as the current measurement point but with the specified *feature code* and *string number*.

The *point id* and *text* are recorded as the point id and text for that vertex of the super string.

If a *point name* exists, then it is a ***named measurement*** and a 4d point string of name *point name* is created and mapped using the Map File. The 4d text is the station prefix followed by *point name*. The *point name* is added to the internal list of named points for searching for co-ordinates.

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Code

the feature code of the new point.

String number

the string number of the new point.

Named point

point name of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search

Point id

point id of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search *. If a new check point entry is inserted into the file, this field will be non-editable since only the check point is required.*

Attribute

Time surveyed

time when the command (op code) was created

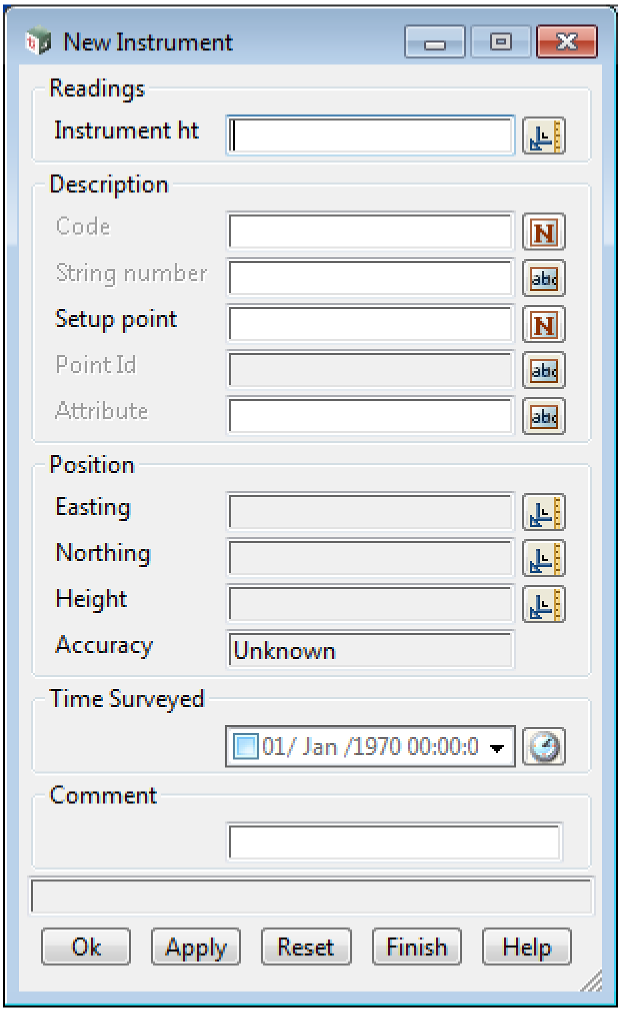
Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

New Instrument (opcode 3)

Define a new instrument setup.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Instrument ht

the height of the instrument setup.

Code

the feature code of the measurement.

String number

the string number of the measurement.

Named point

point name.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search

Point id

point id.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search *. If a new check point entry is inserted into the file, this field will be non-editable since only the check point is required.*

Easting

this field is non-editable and is populated if a valid coordinate exists for the nominated setup point.

Northing

this field is non-editable and is populated if a valid coordinate exists for the nominated setup point.

Height

this field is non-editable and is populated if a valid coordinate exists for the nominated setup point.

Attribute

Time surveyed

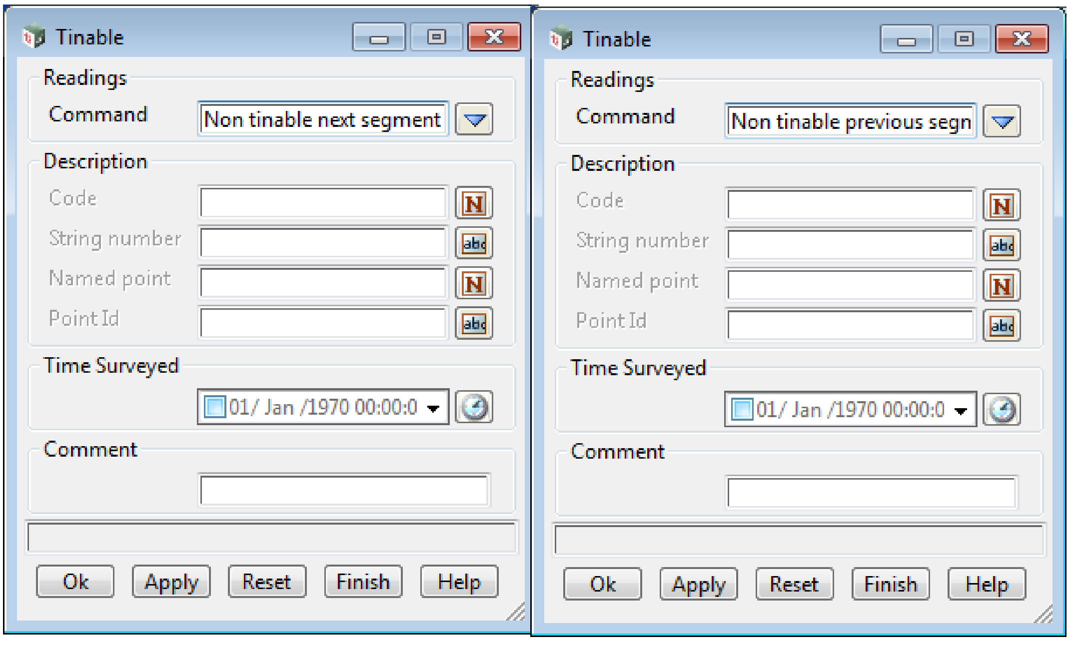
time when the command (op code) was created

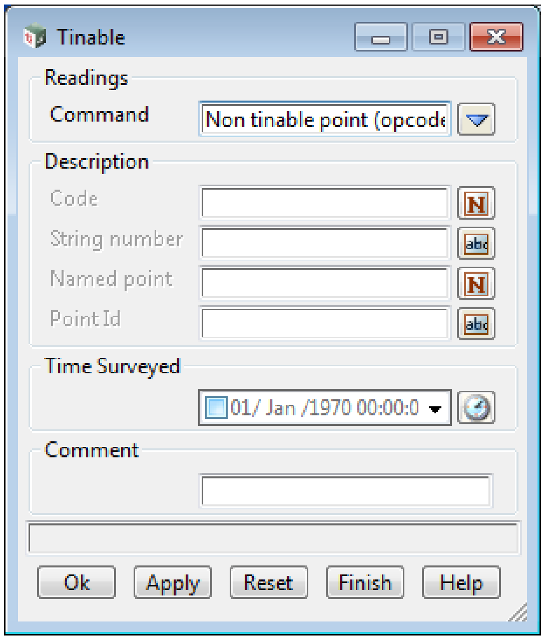
Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Tinable (opcodes 38, 39, 40)





The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Command	choice box		Non tinable previous segment (opcode 39) Non tinable next segment (opcode 38) Non tinable point (opcode 40)

for Make the previous segment non-tinable

If no *description* is given, the previous segment containing the current measurement point is set to non-tinable. That is, it will not be treated as a breakline in triangulations.

If a *description* exists, then either the *feature code* and *string number* or the *point id* section of the *description* can be used.

If the *feature code* and *string number* exist, then the last segment of the previous string with that *feature code* and *string number* is set to non-tinable.

If the *point id* exists, then the segment containing the point with that point id as an end point, is set to non-tinable.

for Make the next segment non-tinable

If no *description* is given, the next segment containing the current measurement point as a starting point is set to non-tinable. That is, it will not be treated as a breakline in triangulations.

If a *description* exists, then either the *feature code* and *string number* or the *point id* section of the *description* can be used.

If the *feature code* and *string number* exist, then the segment that is created in the future from the last point of the previous string with that *feature code* and *string number* is set to non-tinable.

If the *point id* exists, then the segment containing the point with that point id as a start point, is set to non-tinable.

for Make a point not tinable

If no *description* is given, the current measurement point is set to not tinable. That is, it will not be included in triangulations.

If a *description* exists, then either the *feature code* and *string number* or the *point id* section of the *description* can be used.

If the *feature code* and *string number* exist, then the last point of the previous string with that *feature code* and *string number* is set to not tinable.

If the *point id* exists, then the point with that point id is set to not tinable.

Attribute

Time surveyed

time when the command (op code) was created

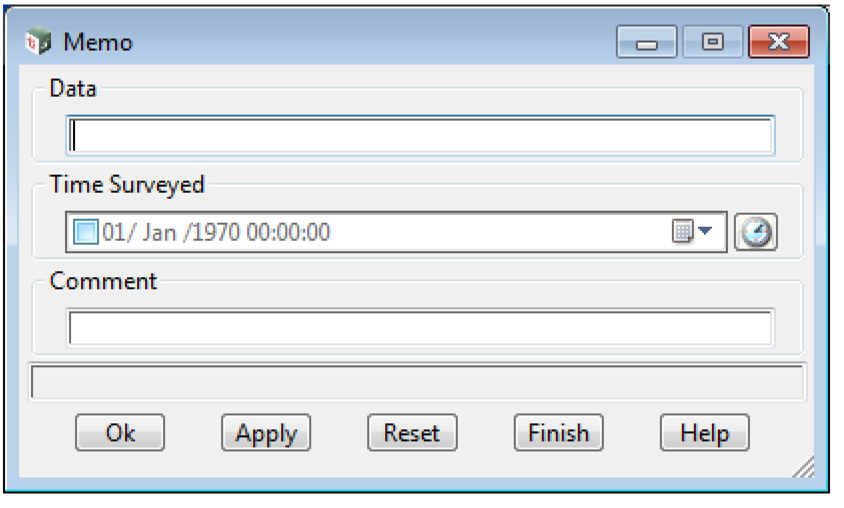
Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Note (opcode 29)

Give extra text to be added as information in the field file.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Data

extra text to be added as information in the field file.

Time surveyed

time when the command (op code) was created

Comment

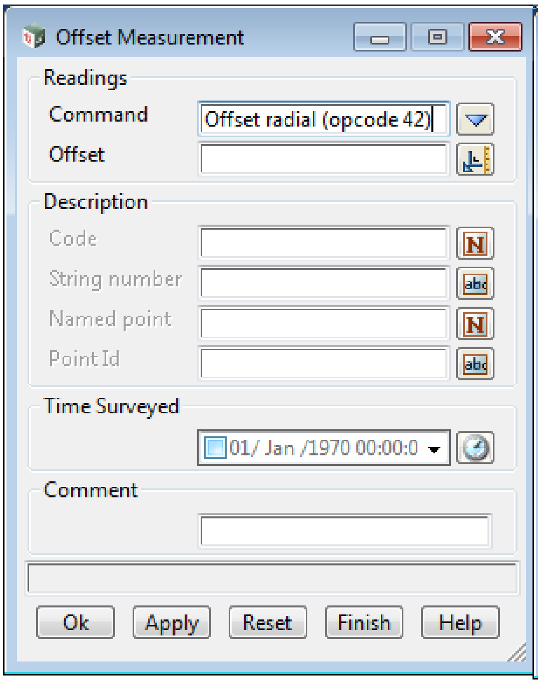
insert a comment within the field file.

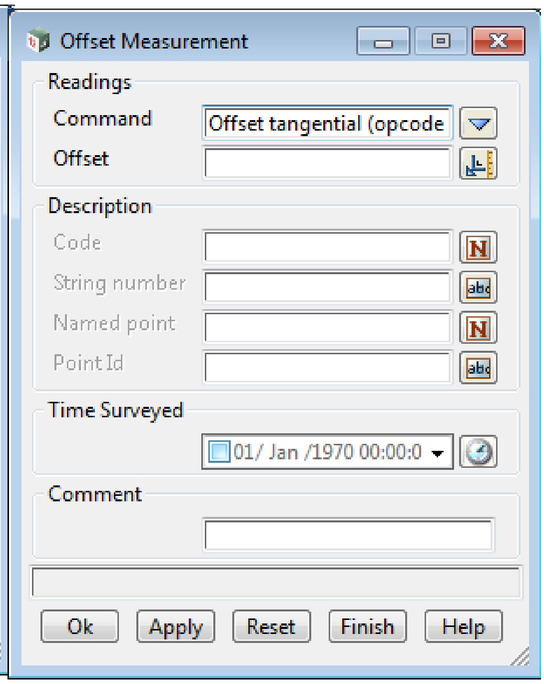
OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

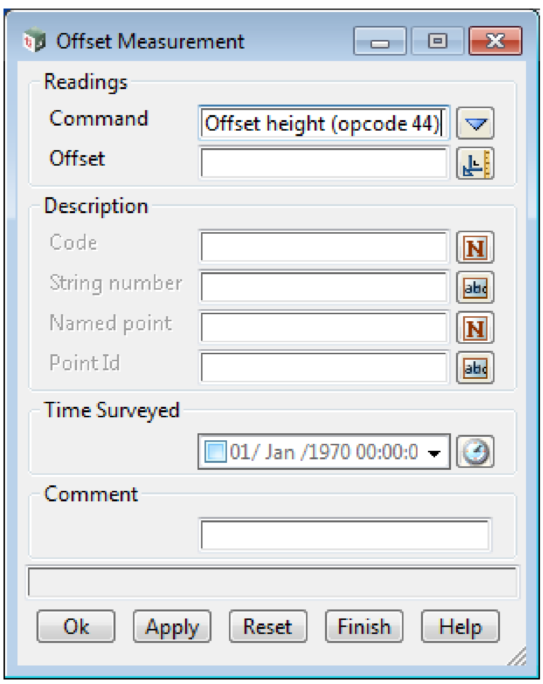
Offset Measurement (opcodes 42, 43, 44)

It is not always possible to measure a point directly but it may be possible to measure a point nearby and then measure an offset to adjust the measured point by and so produce the coordinates of the required point.

See 36.3.2 Offsets in Appendix 36 12d Survey Guide .







The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Command			Offset radial (opcode 42) Offset tangential (opcode 43)

for Radial

The *radial offset* is used to adjust the position of the specified point by a plan distance from the specified points original position, along the plan line joining the current station to the specified point. A positive offset is away from the station, negative is toward the station.

If no *point description* is given, the offset is used to adjust the position of the current measured point.

If a *point description* exists, then either the *feature code* and *string number* or the *point id* section of the *point description* can be used.

If the *feature code* and *string number* exist, then the last point of the previous string with that *feature code* and *string number* is adjusted.

If the *point id* exists, then the point with that point id is adjusted.

for Tangential

The *tangential offset* is used to adjust the position of the specified point by a plan distance from the specified points original position, at rights angles to the plan line joining the current station to the specified point. A negative offset is to the left (looking from the station), and positive is to the right (looking from the station).

If no *point description* is given, the offset is used to adjust the position of the current measured point.

If a *point description* exists, then either the *feature code* and *string number* or the *point id* section of the *point description* can be used.

If the *feature code* and *string number* exist, then the last point of the previous string with that *feature code* and *string number* is adjusted.

If the *point id* exists, then the point with that point id is adjusted.

for Height

If the height of the specified point is not null, then the *height offset* adjusts the height of the point. A positive offset adds to the height, a negative offset reduces the height.

If no *point description* is given, the offset is used to adjust the position of the current measured point.

If a *point description* exists, then either the *feature code* and *string number* or the *point id* section of the *point description* can be used.

If the *feature code* and *string number* exist, then the last point of the previous string with that *feature code* and *string number* is adjusted.

If the *point id* exists, then the point with that point id is adjusted.

Time surveyed

time when the command (op code) was created

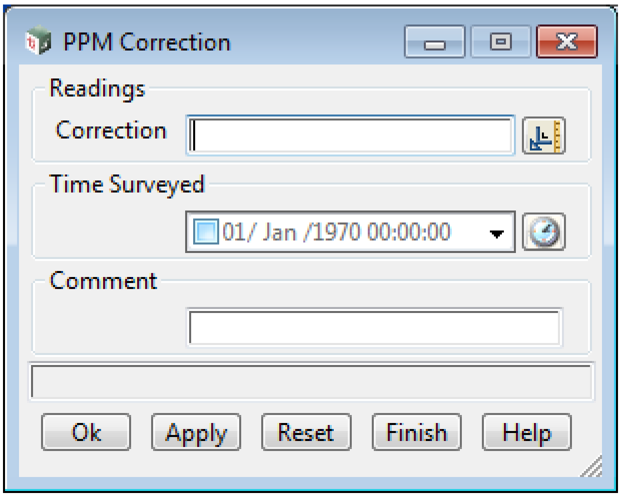
Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

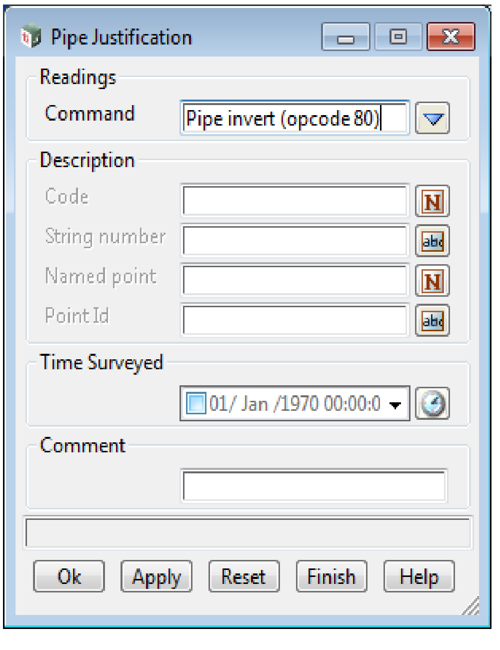
PPM Correction (opcode 131)

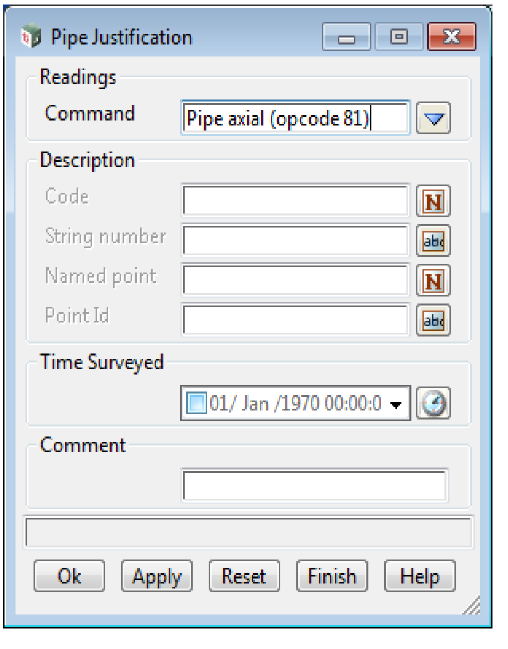
This section of documentation is a work in progress and will be updated in subsequent releases.

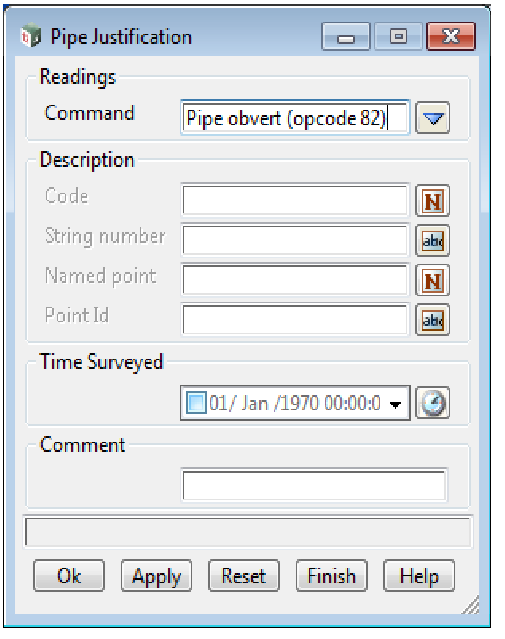


Pipe Justification (opcodes 80, 81, 82)

Specify what type of pipe justification point that the current measurement is.







The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Command			Pipe invert (opcode 80) Pipe axial (opcode 81) Pipe obvert (opcode 82)

for Pipe invert (opcode 80)

If no *point description* is given, the current measurement point is on the invert (bottom) of a pipe. This is the default for measurements to points on pipe strings. If the point is not part of a pipe string, it is ignored.

If the *feature code* and *string number* exist, the last point of the previous string with the same *feature code* and *string number* as given in *point description* is on the invert (bottom) of a pipe. If the point is not part of a pipe string, it is ignored.

If the *point id* exists, then the point with that point id is on the invert (bottom) of a pipe. If the point is not part of a pipe string, it is ignored.

for Pipe axial (opcode 81)

If no *point description* is given, the current measurement point is on the axis (centre) of a pipe. If the point is not part of a pipe string, it is ignored.

If the *feature code* and *string number* exist, the last point of the previous string with the same *feature code* and *string number* as given in *point description* is on the axis (centre) of a pipe. If the point is not part of a pipe string, it is ignored.

If the *point id* exists, then the point with that point id is on the axis (centre) of a pipe. If the point is not part of a pipe string, it is ignored.

for Pipe obvert (opcode 82)

If no *point description* is given, the current measurement point is on the obvert (top) of a pipe. If the point is not part of a pipe string, it is ignored.

If the *feature code* and *string number* exist, the last point of the previous string with the same *feature code* and *string number* as given in *point description* is on the obvert (top) of a pipe. If the point is not part of a pipe string, it is ignored.

If the *point id* exists, then the point with that point id is on the obvert (top) of a pipe. If the point is not part of a pipe string, it is ignored.

Code

the feature code of the measurement.

String number

the string number of the measurement.

Named point

point name.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search

Point id

point id.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search. *If a new check point entry is inserted into the file, this field will be non-editable since only the check point is required.*

Time surveyed

time when the command (op code) was created

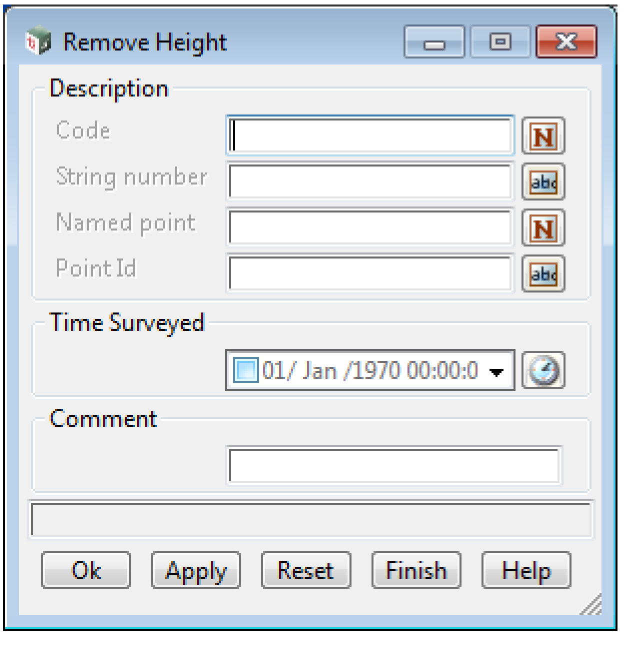
Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Remove Height (opcode 30)

Set the height of a measurement to null.



If no *point description* is given, the height of the current measurement point is set to null.

If the *feature code* and *string number* exist, then the height of the last point of the previous string with that *feature code* and *string number* is set to null.

If the *point id* exists, then the height of the point with that point id is set to null.

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Code

the feature code of the measurement.

String number

the string number of the measurement.

Named point

point name.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search

Point id

point id.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search *. If a new check point entry is inserted into the file, this field will be non-editable since only the check point is required.*

Time surveyed

time when the command (op code) was created

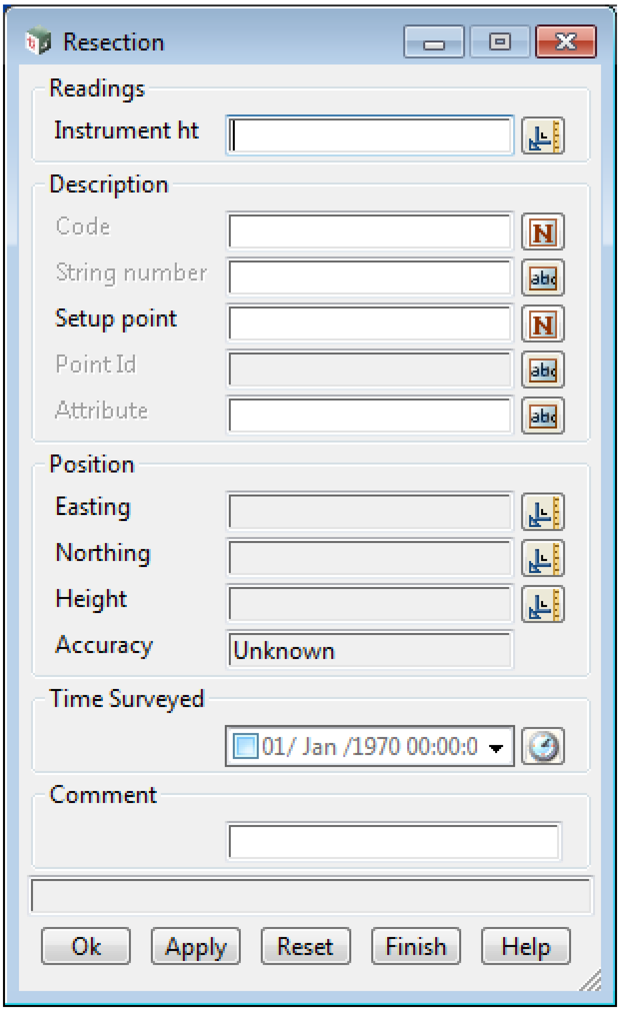
Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Resection Start (opcode 128)

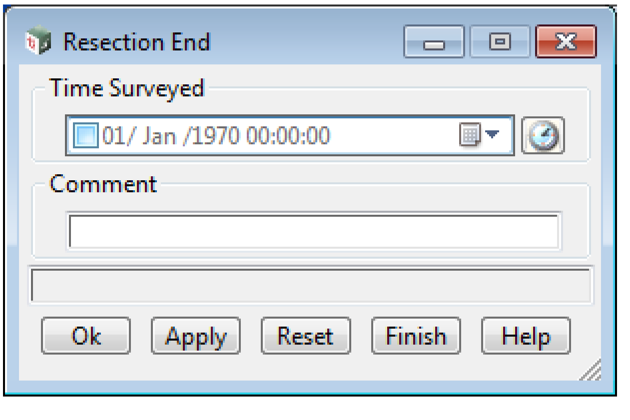
This option is under development.



OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Resection end (opcode 129)

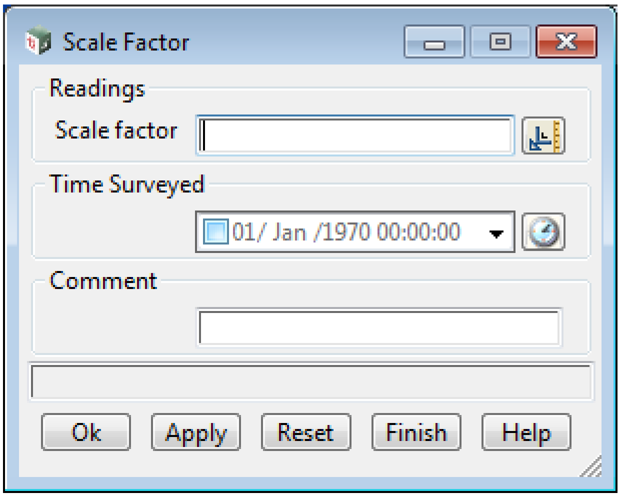
This option is under development.



OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Scale Factor (opcode 9)

Specify the scale factor for subsequent measurements.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Scale factor

the scale factor to be applied to subsequent distance measurements. It is applied by multiplying raw distances by the scale factor to give the corrected distance.

Time surveyed

time when the command (op code) was created

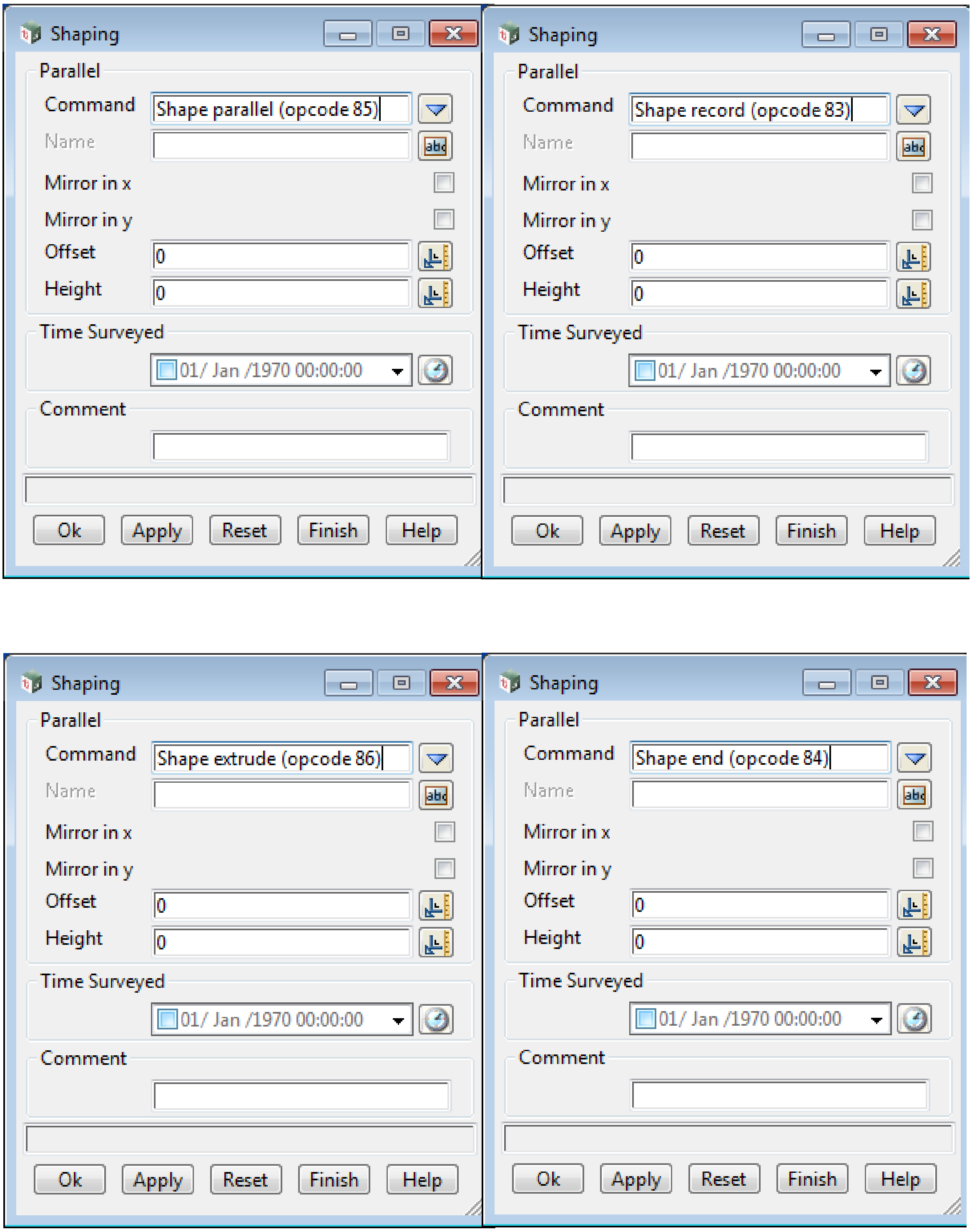
Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Shaping (opcodes 83 to 86)

See 36.5 Shape field coding in Appendix 36 12d Survey Guide .



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Command	choice box		Shape end (opcode 84) Shape record (opcode 83) Shape extrude (opcode 86) Shape parallel (opcode 85)

for Shape end

Stops using the current shape or stops recording a shape.

for Shape record

Start recording a shape with the shape name. If *Shape_name* is not blank, then the default field Shape is defined by the *feature_code* and *string_number* of the following measurements until a *shape end* co. There is no limit to the number of points in a shape.

for Shape extrude

extrude the current shape along the specified super string.

for Shape parallel

parallel the current shape along the specified super string. This creates a number of strings to represent each feature code of the shape record. In the case of shapes which contain curves, a number of strings will be created according to an arc/chord tolerance.

Name

the name of the shape.

Time surveyed

time when the command (op code) was created

Comment

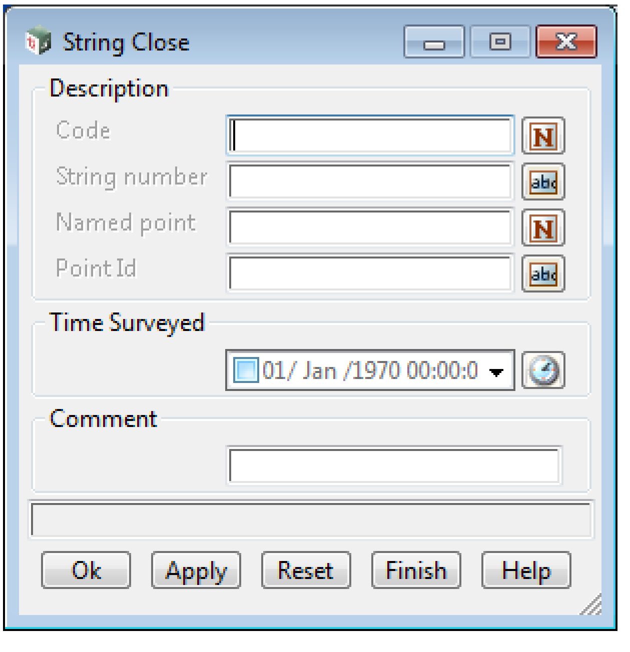
comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

String Close (opcode 20)

Close a string.

See 36.3.4 Close String in Appendix 36 12d Survey Guide .



If no *point description* is given, the current string is closed.

If the *feature code* and *string number* from the *point description* exist, the last previous string with that *feature code* and *string number* is closed.

If the *point id* from the *point description* exists, then the string containing that point id will be closed.

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Code

the feature code of the measurement.

String number

the string number of the measurement.

Named point

point name.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search

Point id

point id.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search. *If a new check point entry is inserted into the file, this field will be non-editable since only the check point is required.*

Time surveyed

time when the command (op code) was created

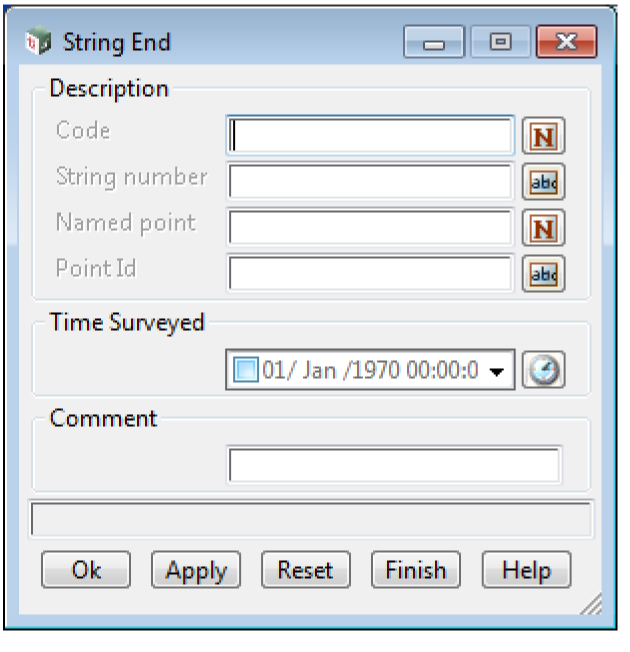
Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

String End (opcode 48)

Stop a string.



If no *Description exist*s, the current string is terminated (including the current measurement point).

If *Code* (feature code) and *String number* exist, then the last point of the previous string with that feature code and string number becomes the last point of that string.

If *Point id* exists, then the previous string containing the point with that point id is terminated *after* the point id point

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Code

the feature code of the measurement.

String number

the string number of the measurement.

Named point

point name.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search

Point id

point id.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search *. If a new check point entry is inserted into the file, this field will be non-editable since only the check point is required.*

Time surveyed

time when the command (op code) was created

Comment

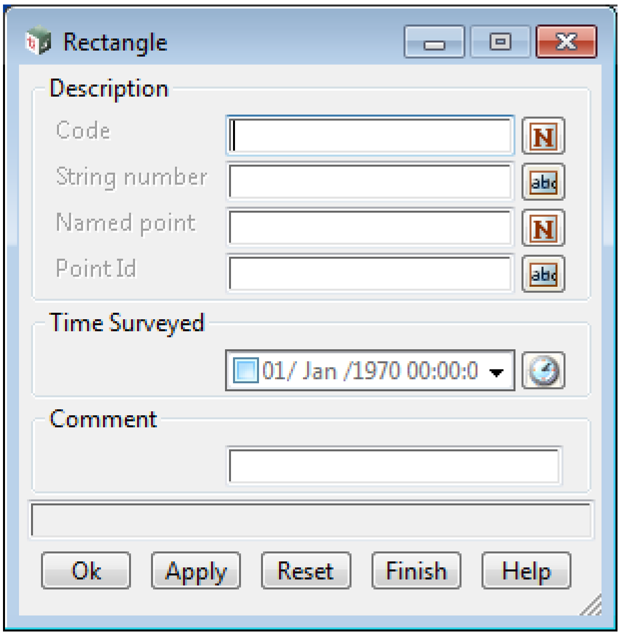
comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Rectangle (opcode 45)

Create a parallelogram (squashed rectangle) from three points.

See 36.3.5 Rectangle in Appendix 36 12d Survey Guide .



If no *Description* is given, the current measurement point and the two previous points from the current string are used and a fourth point is created to form a parallelogram (squashed rectangle) and the height of the fourth point is set to null. The string is then closed.

If the *feature code* and *string number* exist, the last three points with that *feature code* and *string number* are used and a fourth point is created to form a parallelogram (squashed rectangle) and the height of the fourth point is set to null. The string is then closed.

If the *point id* exists, then the *feature code* and *string number* of the point with that point id are used and processed as above. Note that the point with the point id is not necessarily used.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Code

the feature code of the measurement.

String number

the string number of the measurement.

Named point

point name of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search

Point id

point id of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search *. If a new check point entry is inserted into the file, this field will be non-editable since only the check point is required.*

Time surveyed

time when the command (op code) was created

Comment

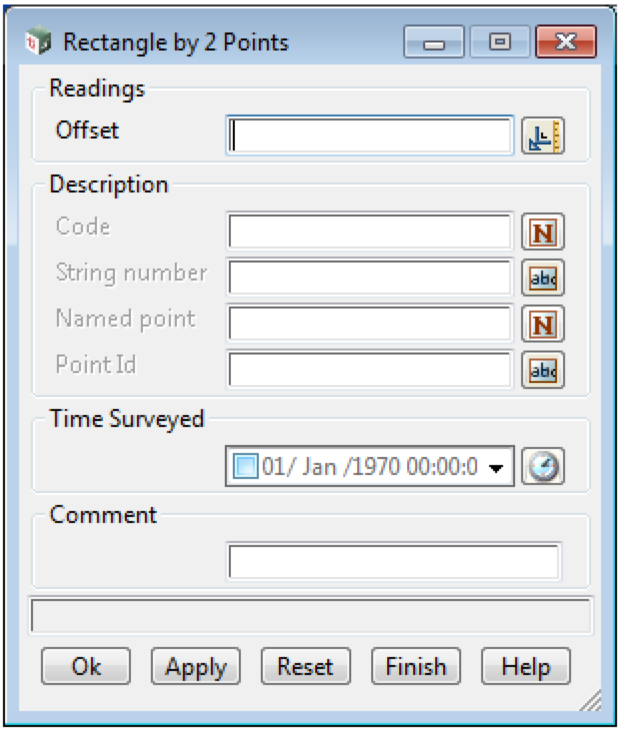
comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Rectangle By 2 Points (opcode 37)

Create a rectangle from two points and a given offset.

See 36.3.6 Rectangle by 2 Points in Appendix 36 12d Survey Guide .



The rectangle is defined by two points (reference side) and a offset.

If a positive offset value is given, two points will be created to the right of the reference side.

If a negative offset value is given, two points will be created to the left of the reference side.

If no *Description* is given, the two new points will be joined to the given points in a closed rectangular string, and will have the same feature code as the points given.

If the *feature code* and *string number* exist, then a search is made for the last occurrence of two points with the same *feature code* and *string number*. If found, then these points are used to define the reference side of the rectangle.

If the *point id* exists, then a search is made for the last occurrence of two points with the same *feature code* and *string number* as the point given by the point id. If found, then these points are used to define the reference side of the rectangle.

Two consecutive rectangles are unable to be defined side by side. In other words if the two points given are part of string of greater than two vertices, the command will only work for sets of two points that are exclusively defined. i.e. For a 5 point string, a rectangle can be defined by points 1 and 2, and 4 and 5.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Code

the feature code of the measurement.

String number

the string number of the measurement.

Named point

point name.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search

Point id

point id.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search *. If a new check point entry is inserted into the file, this field will be non-editable since only the check point is required.*

Time surveyed

time when the command (op code) was created

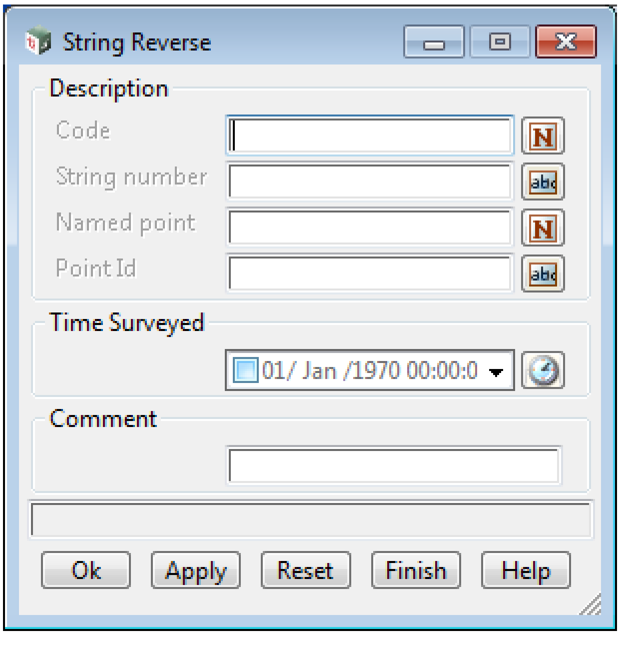
Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

String Reverse (opcode 19)

Reverse the direction of a string.



If no *point description* is given, the current string is reversed.

If the *feature code* and *string number* from the *point description* exist, the last previous string with that *feature code* and *string number* is reversed.

If the *point id* from the *point description* exists, then the string containing that point id will be reversed.

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Code

the feature code of the measurement.

String number

the string number of the measurement.

Named point

point name.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search

Point id

point id.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search *. If a new check point entry is inserted into the file, this field will be non-editable since only the check point is required.*

Time surveyed

time when the command (op code) was created

Comment

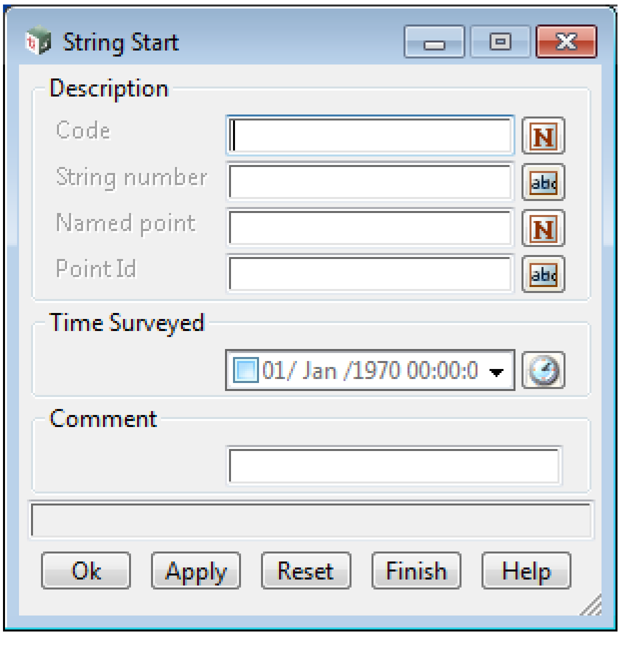
comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

String Start (opcode 47)

Start a new string.

See 36.3.3 Start New String in Appendix 36 12d Survey Guide .



If no *Description* is given, the current string is terminated (without including the current measurement point) and the current measurement point becomes the first point of a new string with the same feature code and string number.

If the *feature code* and *string number* exist, then the last point of the previous string with that feature code and string number becomes the first point of a new string with the same *feature code* and *string number*.

If the *point id* exists, then the previous string containing the point with that point id is terminated *before* the point id point, and the point becomes the first point of a new string with the *same feature code* and *string number*.

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Code

the feature code of the measurement.

String number

the string number of the measurement.

Named point

point name.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search

Point id

point id.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search*. If a new check point entry is inserted into the file, this field will be non-editable since only the check point is required.*

Time surveyed

time when the command (op code) was created

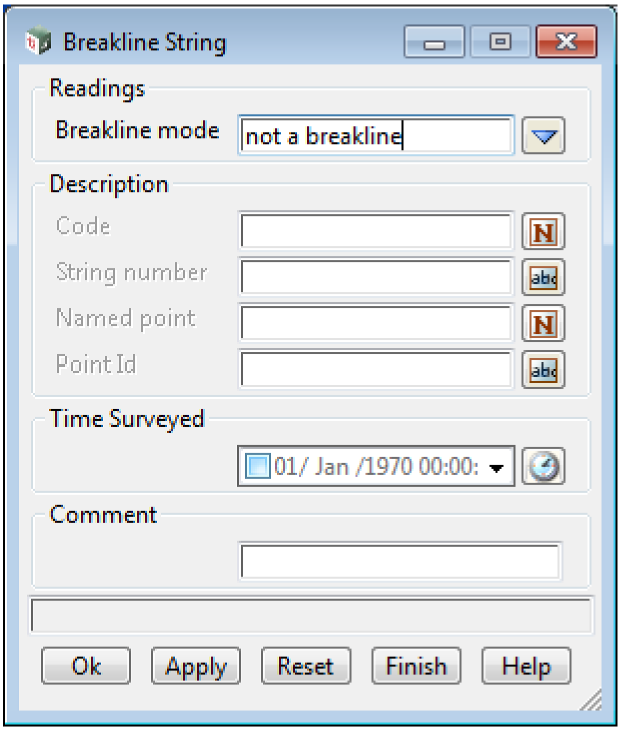
Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Breakline String (String tinable (opcode 46))

Make a string a breakline (*i.e.* all vertices and segments are tinable) or not a breakline.



The *point description* is used to select a string and the *mode* is used specify if the string is a breakline or not.

If no *point description* is given, the current string is selected.

If the *feature code* and *string number* exist, the last string with that *feature code* and *string number* is selected.

If the *point id* exists, then the string containing the point with that point id is selected.

If no *mode* is given, the selected string is set as a point string (that is, not a breakline).

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Breakline mode			not a breakline breakline

for Not a breakline

the selected string is set to a point string and hence is not a breakline (however the points are tinable).

for Breakline

the selected string is set to a line string (all vertices and segments are tinable) and is therefore a breakline.

Code

the feature code of the measurement.

String number

the string number of the measurement.

Named point

point name of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search

Point id

point id of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search *. If a new check point entry is inserted into the file, this field will be non-editable since only the check point is required.*

Time surveyed

time when the command (op code) was created

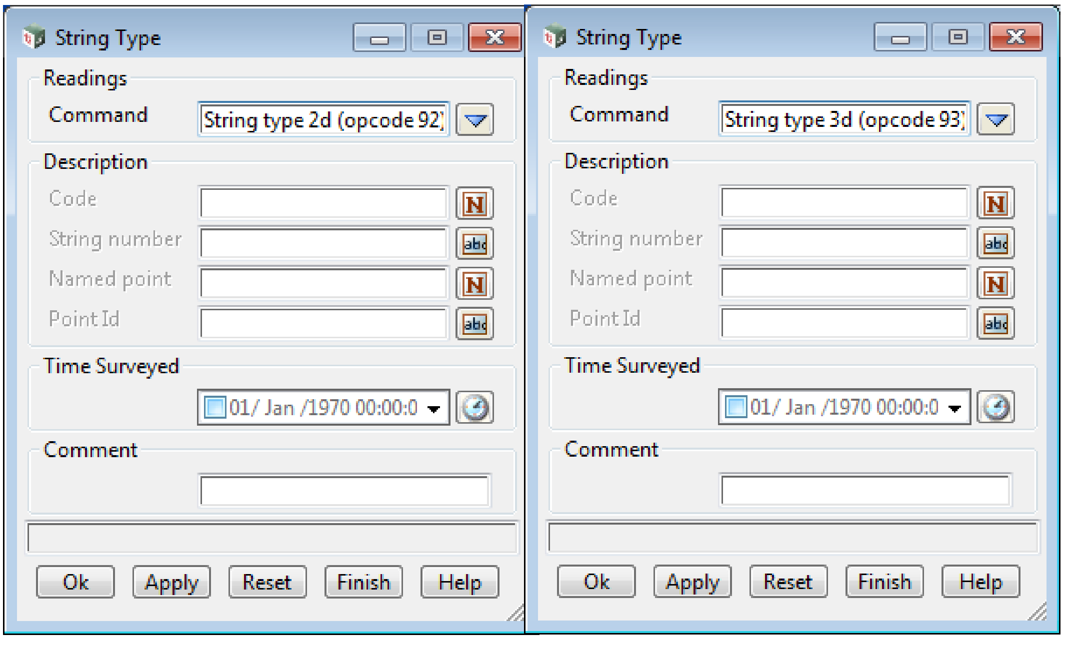
Comment

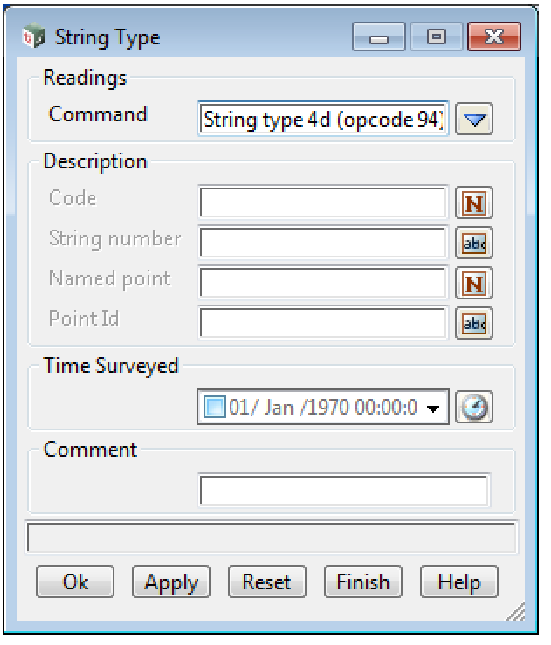
comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

String Type (opcodes 92, 93, 94)

Specify the string to be a 2d, 3d or 4d string.





The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Command	choice box	String type 2d	String type 2d (opcode 92) String type 3d (opcode 93) String type 4d (opcode 94)

*if **2d** - the string has one height for the entire string*
*if **3d** - the string can have different heights at each vertex*
*if **4d** - the string can have different heights and text at each vertex*

Code

the feature code of the measurement.

String number

the string number of the measurement.

Named point

point name.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search

Point id

point id.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search. *If a new check point entry is inserted into the file, this field will be non-editable since only the check point is required.*

Time surveyed

time when the command (op code) was created

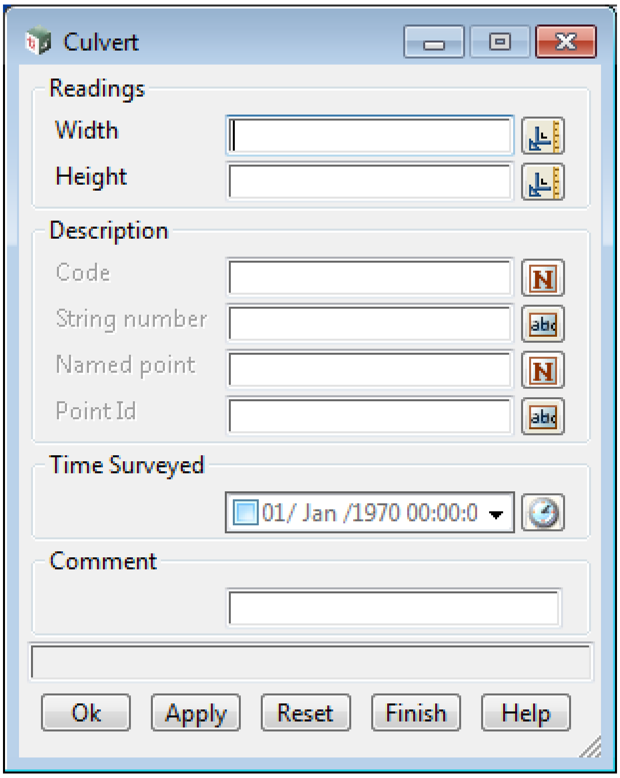
Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons.

Culvert (opcode 96)

Specify the string to be a culvert and give the culvert width and height.



Culvert strings are always line strings and are stored with the justification of the majority of the string points. Individual culvert points are picked up either top (obvert), centre (axial) or bottom (invert) of the culvert using op codes 80, 81 and 82.

If no *point_description* is given, the current string is created as a culvert string with the given width and height.

If a *point_description* exists, then either the *feature code* and *string number* or the *point id* section of the *point_description* can be used.

If the *feature code* and *string number* exist, the last string with the same *feature code* and *string number* is created as a culvert with the given width and height.

If the *point id* exists, then the string containing that point id is created as a culvert string with the given width and height.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Width

the width of the culvert

Height

the height of the culvert

Code

the feature code of the measurement

String number

the string number of the measurement

Named point

point name.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search

Point id

point id.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search *. If a new check point entry is inserted into the file, this field will be non-editable since only the check point is required.*

Time surveyed

time when the command (op code) was created

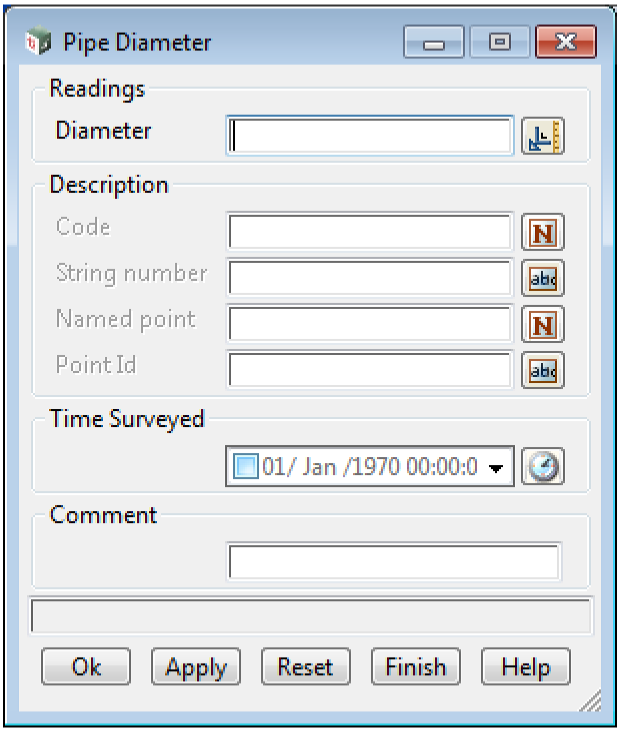
Comment

comment for the command within the field file

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons .

Pipe Diameter (opcode 95)

Specify the string to be a pipe and give the pipe diameter.



Pipe strings are always line strings and are stored with the justification of the majority of the string points. Individual pipe points are picked up either top (obvert), centre (axial) or bottom (invert) of the pipe using op codes 80, 81 and 82.

If no *point_description* is given, the current string is created as a pipe string with the given diameter.

If a *point_description* exists, then either the *feature code* and *string number* or the *point id* section of the *point_description* can be used.

If the *feature code* and *string number* exist, the last string with the same *feature code* and *string number* is created as a pipe with the given diameter.

If the *point id* exists, then the string containing that point id is created as a pipe string with the given diameter.

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Diameter

the diameter of the pipe.

Code

the feature code of the measurement.

String number

the string number of the measurement.

Named point

point name.of the point. This can be an integer, real, text or alphanumeric. For a more detailed *description of how the reduction finds/uses the appropriate point see the section* Existing point search

Point id

point id.of the point. This can be an integer, real, text or alphanumeric. For a more detailed description of how the reduction finds/uses the appropriate point see the section Existing point search *. If a new check point entry is inserted into the file, this field will be non-editable since only the check point is required.*

Time surveyed

time when the command (op code) was created

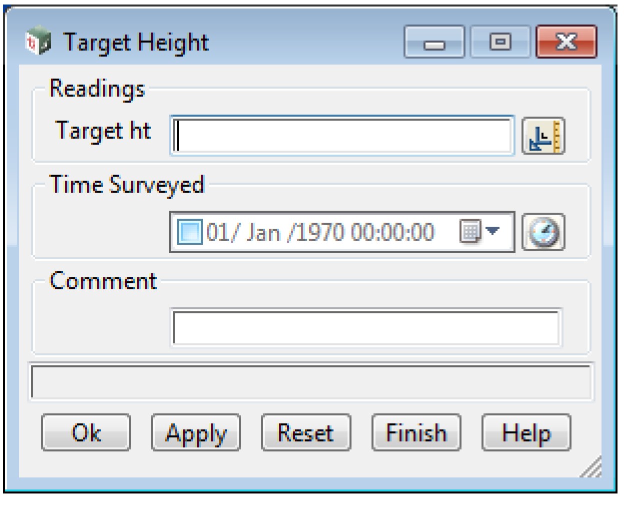
Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Target Height (opcode 5)

Define the target height to be used for following measurements.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Target height

target height of following measurements.

Time surveyed

time when the command (op code) was created

Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Templating (opcodes 51 to 59)

Record and use field templates when picking up string.

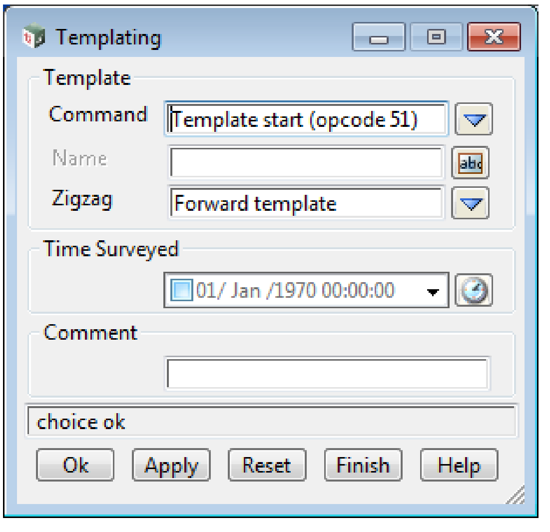
See 36.4 Field Templates in Appendix 36 12d Survey Guide .

For *Template start (opcode 51)*,go to Template start
Template end (opcode 52) Template end
Template pause (opcode 53) Template pause
Template continue (opcode 54) Template continue
Template record (opcode 55) Template record
Template skip (opcode 56) Template skip
Template insert (opcode 58) Template insert
Template delete (opcode 57) Template delete
Template change (opcode 59) Template change

Template start

Selecting *Template start (opcode 51)* brings up the **Templating** panel with the *Command* field set to **Template start**

Template start starts using the field template given in the field ***Name***.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Command		Template start	Template start (opcode 51) Template end (opcode 52) Template pause (opcode 53) Template continue (opcode 54) Template record (opcode 55)
Name	Text box		

Name of the field template to use.
If *Name* is blank, the default field template is used.

Field Description	Type	Defaults	Pop-Up
Zigzag	choice box	Forward template	Forward template Reverse template Start on zig of zigzag Start of zag of zigzag

If *zigzag* is ***forward template***, then the field template is used as a *forward* template.*See* 36.4.1 Forward Direction *in Appendix* 36 12d Survey Guide

reverse template, then the field template is used as a *reverse* template.*See* 36.4.2 Reverse Direction *in Appendix* 36 12d Survey Guide

start on zig, then the field template is used as a zig_zag template and is used in the *forward* definition direction first (that is starts on a zig). *See* 36.4.3 Zig-Zag *in Appendix* 36 12d Survey Guide .

start on zag, then the template is used as a zig_zag template and is used in the *reverse* direction first (that is, starts on a zag).*See* 36.4.3 Zig-Zag *in Appendix* 36 12d Survey Guide .

If *zigzag* is ***blank***, or anything other than ***forward***, ***reverse***, or ***start on zag*** then the field template is used as a zig-zag template starting on a ***zig***.

Time surveyed

time when the command (op code) was created

Comment

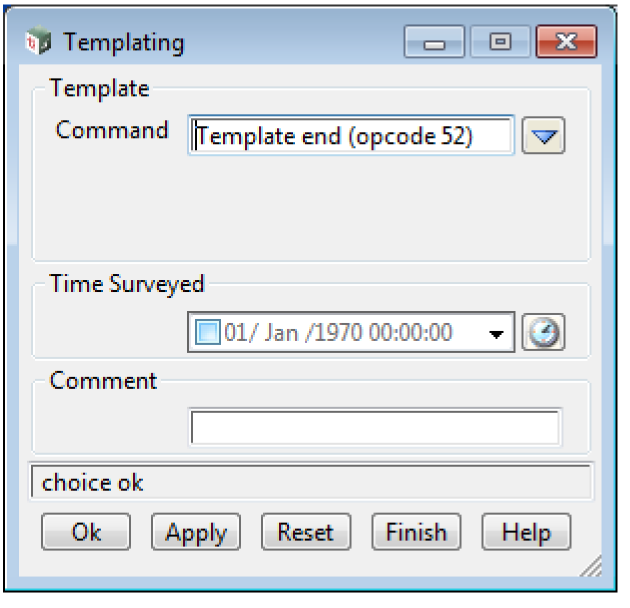
comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Template end

Selecting *Template end (opcode 52)* brings up the **Templating** panel with the *Command* field set to **Template end**

Template end stops using the current field template, or stops recording a field template.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Command		Template end	Template start (opcode 51) Template end (opcode 52) Template pause (opcode 53) Template continue (opcode 54) Template record (opcode 55)

Time surveyed

time when the command (op code) was created

Comment

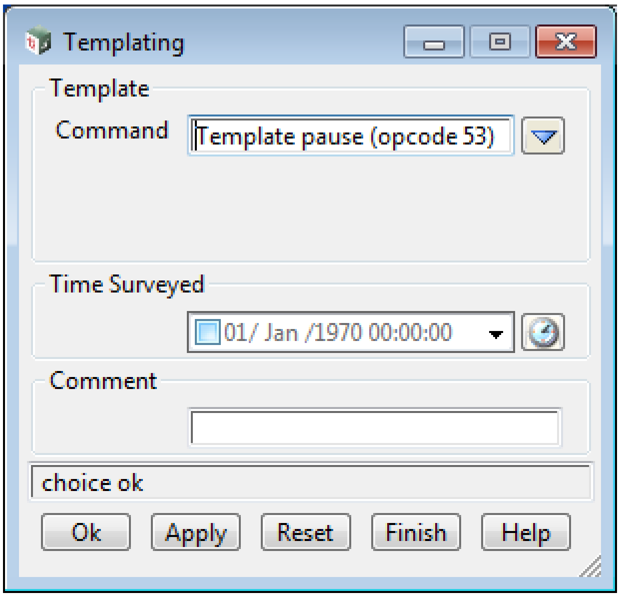
comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Template pause

Selecting *Template pause (opcode 53)* brings up the **Templating** panel with the *Command* field set to **Template pause**

Pause using the current field template or defining a field template, until a *continue field template (54*) or a finish field *template (52) code* is given.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Command		Template pause	Template start (opcode 51) Template end (opcode 52) Template pause (opcode 53) Template continue (opcode 54) Template record (opcode 55)

Time surveyed

time when the command (op code) was created

Comment

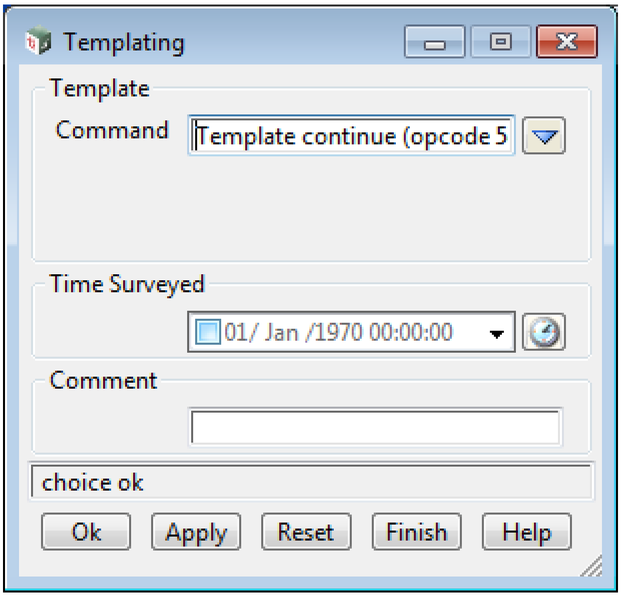
comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Template continue

Selecting *Template continue (opcode 54)* brings up the **Templating** panel with the *Command* field set to **Template continue**

Continue using or defining the current field template, which has been stopped by a *Template pause (opcode 53*). The *Continue* command only needs to be given once and applies to all following measurements until another *Pause* or *Finish* command is given.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Command		Template continue	Template start (opcode 51) Template end (opcode 52) Template pause (opcode 53) Template continue (opcode 54) Template record (opcode 55)

Time surveyed

time when the command (op code) was created

Comment

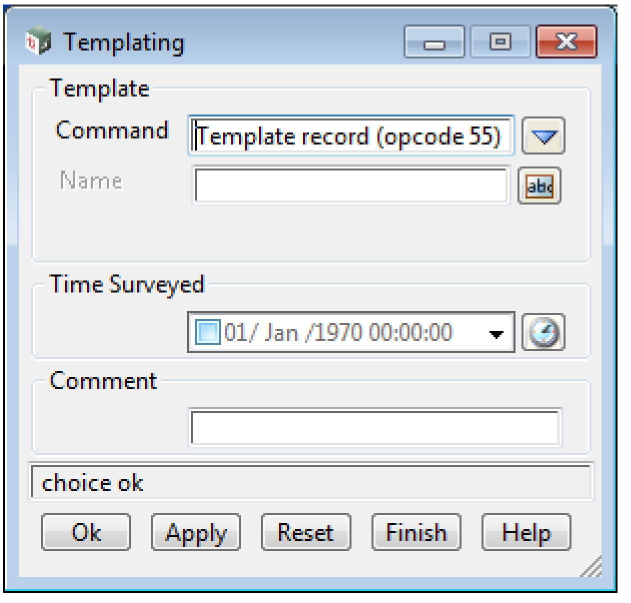
comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Template record

Selecting *Template record (opcode 55)* brings up the **Templating** panel with the Command field set to **Template record**

Template record stores the *feature code* and *string number* of the following measurements as a field template until a *Finish* code (52) is given.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Command		Template record	Template start (opcode 51) Template end (opcode 52) Template pause (opcode 53) Template continue (opcode 54) Template record (opcode 55)

Name

name of the template being created

Start recording a field template with the name **Name**. *If* **Name** *is blank, then it is the default field template that is defined.*

The feature code and string number of the following measurements are stored as the field template until a Finish code (52) is given. There is no limit to the number of feature code and string number pairs that can be stored in a field template.

Time surveyed

time when the command (op code) was created

Comment

comment for the command within the field file.

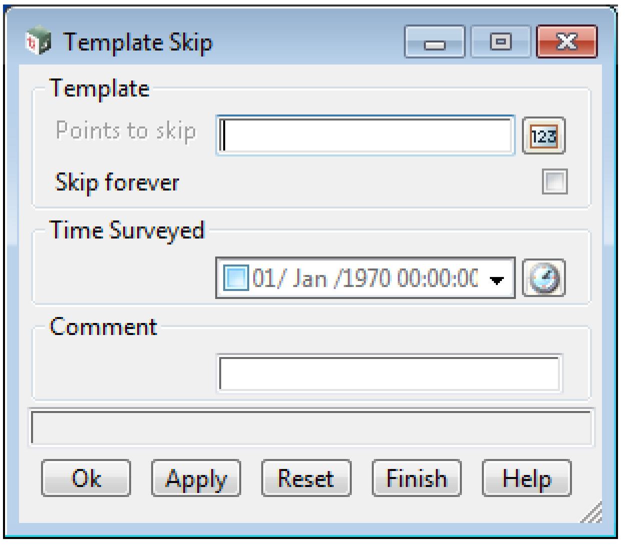
OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Template skip

Selecting *Template skip (opcode 56)* brings up the **Template Skip** panel.

This panel allows the user to skip picking up one or more points (feature code and string number pairs) from the field template currently being used. The next measurement takes the *feature code* and *string number* from the next point after the skipped points, from the field template definition.

See 36.4.4 Skipping Field Template Points in Appendix 36 12d Survey Guide



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Points to skip

the number of points (feature code and string number pairs) of the template to skip.
If **Points to skip** *is blank or zero, then only one point is skipped otherwise* **Points to skip** *points are skipped.*

Skip forever tick box

if ticked, the given number of feature code and string number pairs are not used from then on.
If not ticked, only skip for this one used of the template.

Time surveyed

time when the command (op code) was created

Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

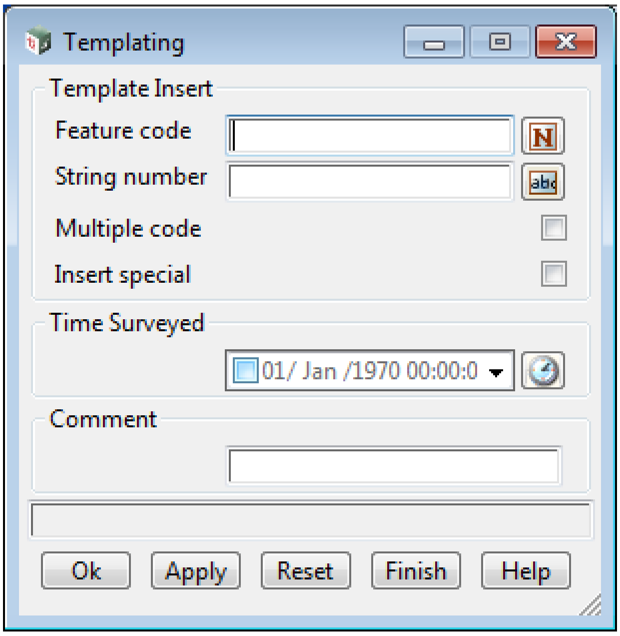
Template insert

Selecting *Template insert (opcode 58)* brings up the **Templating** panel.

This option allows the user to insert new point definitions into the template.

These may be **new points** or to **add a multiple code** to an **existing point** in the template.

See 36.4.5 Insert Template Points or Insert Multiple Codes in Appendix 36 12d Survey Guide



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Feature code

feature code to be inserted (as a new point or a multiple code) with the string number, into the template

String number

string number to be inserted (as a new point or a multiple code) with the feature code, into the template

Multiple code tick box

If ticked off (the default), a new point is inserted into the template with the Feature code and String number given in the panel.

If ticked on, no new point is inserted but the current template point will be made a multiple coded point with the Feature code and String number given in the panel. The multiple coding will be used each time the template point is used.

Insert special tick box

If ticked on, the point will be added to the current template being picked up (that is, to the end of the template).
If not ticked, the point will be added to the next template being picked up (that is, to the beginning of the template).

Time surveyed

time when the command (op code) was created

Comment

comment for the command within the field file.

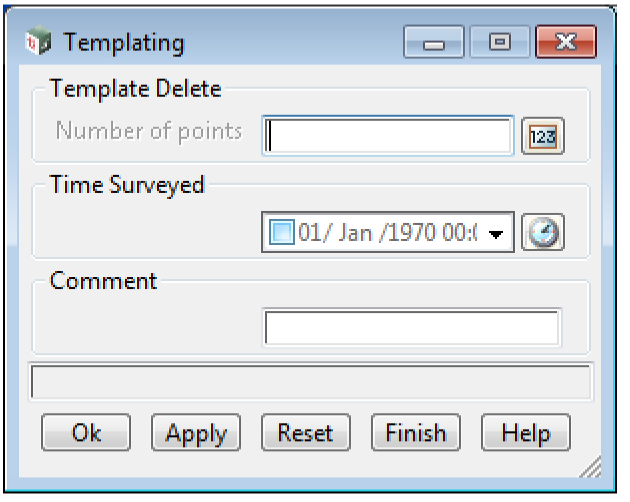
OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Template delete

Selecting *Template delete (opcode 57)* brings up the **Templating** panel.

Allows the user to delete one or more points on the template. Picking up will use the updated template definition.

See 36.4.6 Delete Template Points in Appendix 36 12d Survey Guide



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Number of points

number of points to delete from the template. Any further use of the template will used the updated template definition.

Time surveyed

time when the command (op code) was created

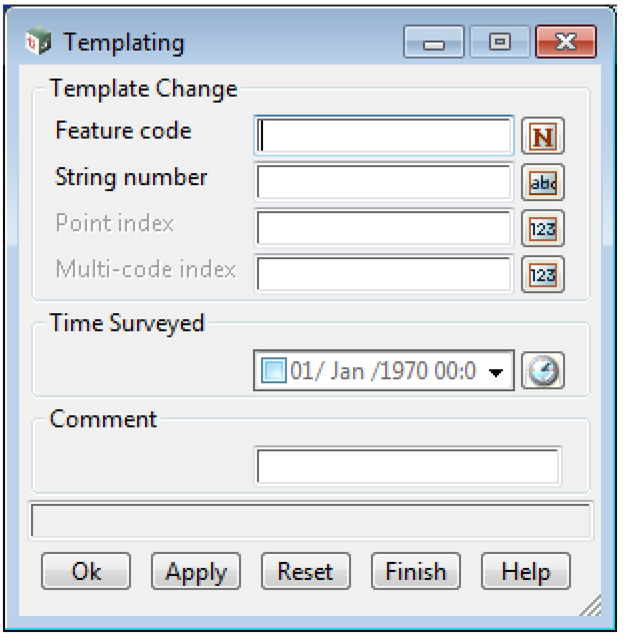
Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Template change

Selecting *Template change (opcode 59)* brings up the **Templating** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Time surveyed

time when the command (op code) was created

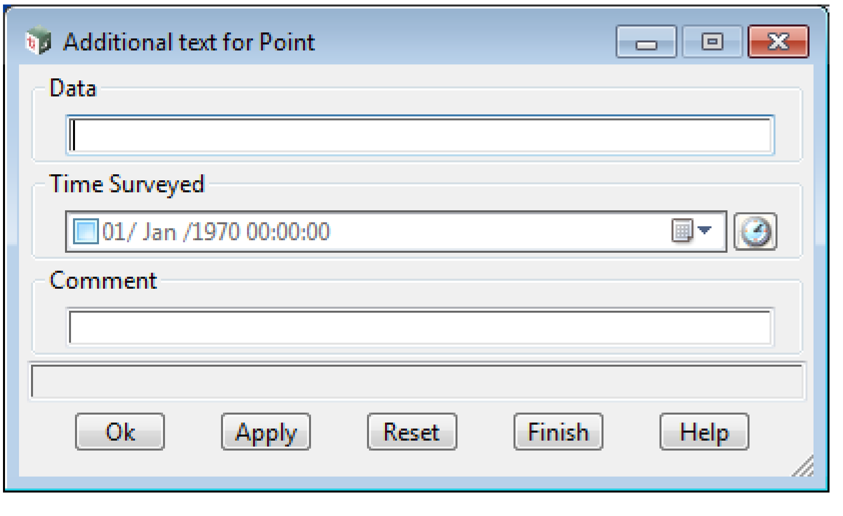
Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Additional Text For Point (opcode 41)

Add extra text to any existing text for the current measurement.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Data

The given text in **Data** *is added to the end of any existing text for the current measurement point.*

Time surveyed

time when the command (op code) was created

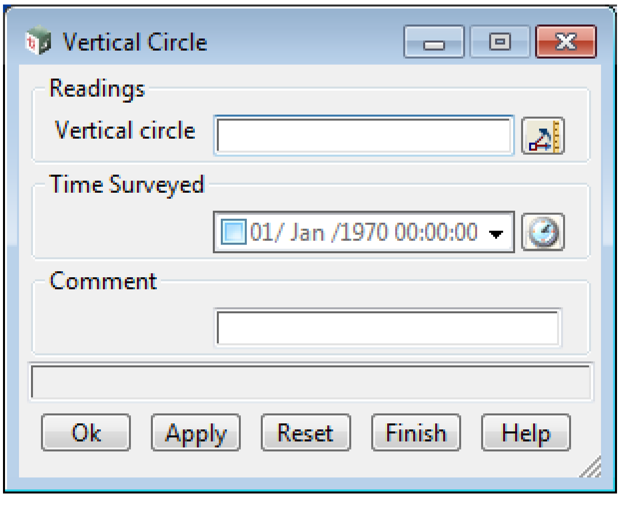
Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

Vertical Circle Correction (opcode 15)

The given value (in decimal degrees) is **subtracted** from the vertical circle value in subsequent measurements.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Vertical circle

the vertical circle (in decimal degrees) is ***subtracted*** *from the vertical circle value in any measurements*

Time surveyed

time when the command (op code) was created

Comment

comment for the command within the field file.

OK, Apply, Reset, Finish, Help See the description for the panel buttons in the section Panel buttons

17.7.7 Order

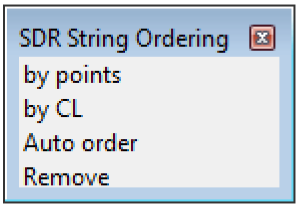
Position of menu: **Survey =>Edit =>Order**

This option is still under development.

The **order** option re-orders points in a string.

The **Order** walk-right menu is

.



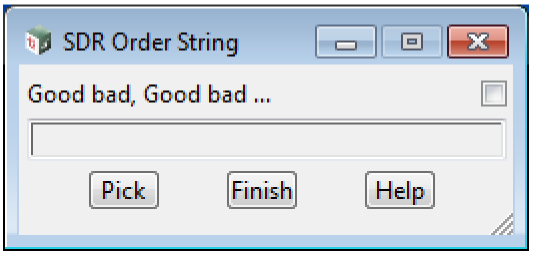
For information on *by points* please go to 17.7.7.1 By Points
by CL 17.7.7.2 By CL
Auto order 17.7.7.3 Auto Order
Remove 17.7.7.4 Remove Order

17.7.7.1 By Points

Position of option on menu: **Survey =>Edit =>Order =>by points**

This option is still under development.

.

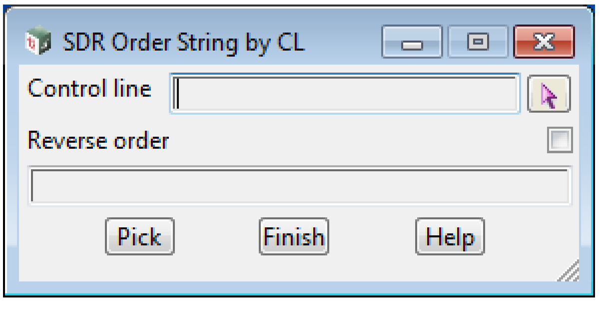


17.7.7.2 By CL

Position of option on menu: **Survey =>Edit =>Order =>by CL**

This option is still under development.

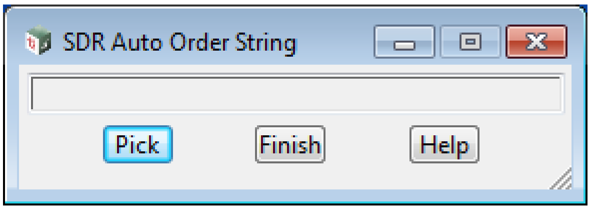
.



17.7.7.3 Auto Order

Position of option on menu: **Survey =>Edit =>Order =>Auto order**

This option is still under development.

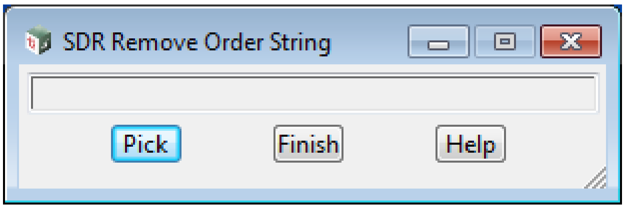


17.7.7.4 Remove Order

Position of option on menu: **Survey =>Edit =>Order =>Remove**

This option is still under development.

.



17.7.8 SDR Point Edits

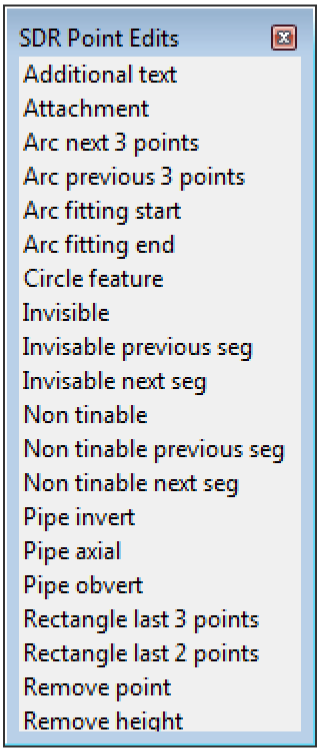
Position of menu: **Survey =>Edit =>Points**

The **Points** walk-right menu contains options that work on individual points (vertices).

Note - a point may be individual point (one vertex strings) or a vertex of a string.

The **Points** walk-right menu is:

.



add text to the vertex
attach an image to the vertex
put an arc through the next 3 points
put an arc through the preceding 3 points
start arc fitting
end arc fitting
make the vertex a feature
make the vertex invisible
make the last segment invisible
make the next segment invisible
make the vertex non tinable
make the last segment non tinable
make the next segment non tinable
the z-value is the invert of a pipe
the z-value is the centre of a pipe
the z-value is the overt of a pipe
make a rectangle of the last 3 points
make a rectangle of the last two points
delete a point
make the height null

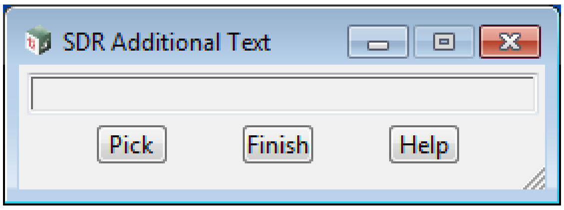
For the option *Additional text*, go to 17.7.8.1 Additional Text
Attachment 17.7.8.2 Attach a File
Arc next 3 points 17.7.8.3 Arc Through Next Three Points
Arc last 3 points 17.7.8.4 Arc Through Last Three Points
Arc fitting start 17.7.8.5 Start Arc Fitting
Arc fitting end 17.7.8.6 End Arc Fitting
Circle feature 17.7.8.7 Create a Feature String
Invisible 17.7.8.8 Make a Vertex Invisible
Invisible last seg 17.7.8.9 Make the Previous Segment Invisible
Invisible next seg 17.7.8.10 Make the Next Segment Invisible
Non tinable 17.7.8.11 Make a Vertex Non Tinable
Non tinable last seg 17.7.8.12 Make the Previous Segment Non Tinable
Non tinable next seg 17.7.8.13 Make the Next Segment Non Tinable
Pipe invert 17.7.8.14 Make a Vertex an Invert Level of a Pipe
Pipe axial 17.7.8.15 Make a Vertex an Axial Level of a Pipe
Pipe obvert 17.7.8.16 Make a Vertex an Obvert Level of a Pipe
Rectangle last 3 points 17.7.8.17 Make a Parallelogram from the Last Three Points
Rectangle last 2 points 17.7.8.18 Make a Rectangle from the Last Two Points
Remove point 17.7.8.19 Delete a Vertex
Remove height 17.7.8.20 Set a Vertex Height to Null

17.7.8.1 Additional Text

Position of option on menu: **Survey =>Edit =>Points =>Additional text**

Additional text adds text to selected vertices. The user selects the vertex to add text to, and then types in the text.

Selecting **Additional text** brings up the **SDR Additional Text** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and vertices can be selected.*

*Each time a vertex is picked and accepted, an **Additional text** typed input panel is displayed to type the text into. A **Text** code is then inserted into the 12d field file.*

The option continues until the **Finish** *button is selected.*

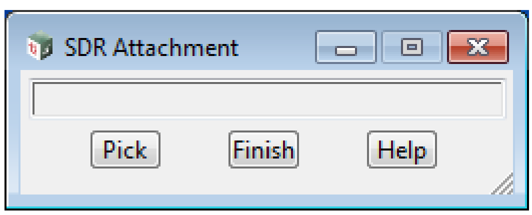
*If the **pick** is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

17.7.8.2 Attach a File

Position of option on menu: **Survey =>Edit =>Points =>Attachment**

Attachment adds files to selected vertices. The user selects the vertex to add an attachment to, and then enters the name of the file to attach.

Selecting **Attachment** brings up the **SDR Attachment** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and vertices can be selected.*

*Each time a vertex is picked and accepted, an **Attachment file name** typed input panel is displayed to enter the file name into. An **Attachment** field code is then inserted into the 12d field file.*

The option continues until the **Finish** *button is selected.*

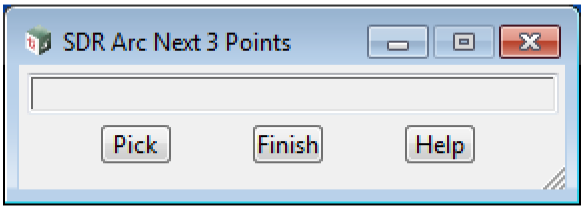
*If the **pick** is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

17.7.8.3 Arc Through Next Three Points

Position of option on menu: **Survey =>Edit =>Points =>Arc next 3 points**

Arc next 3 points adds an arc through the selected vertex and the next two vertices of the same string.

Selecting **Arc next 3 points** brings up the **SDR Arc Next 3 Points** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and vertices can be selected.*

*Each time a vertex is picked and accepted, an arc is placed through the selected vertex and the next two vertices of the same string by inserting an **Arc through next 3 points** field code.*

The option then repeats and continues until the **Finish** *button is selected.*

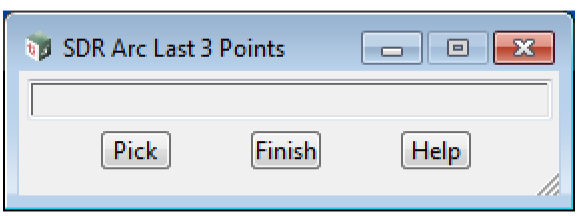
*If the **Pick** is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

17.7.8.4 Arc Through Last Three Points

Position of option on menu: **Survey =>Edit =>Points =>Arc last 3 points**

Arc last 3 points adds an arc through the selected vertex and the previous two vertices of the same string.

Selecting **Arc last 3 points** brings up the **SDR Arc Last 3 Points** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and vertices can be selected.*

*Each time a vertex is picked and accepted, an arc is placed through the selected vertex and the previous two vertices of the same string by inserting an **Arc through last 3 points** field code.*

The option then repeats and continues until the **Finish** *button is selected.*

*If the **Pick** is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the*

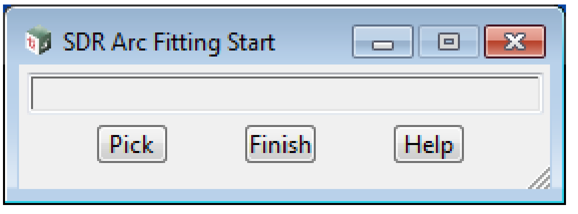
Pick *button can be used to restart the option.*

17.7.8.5 Start Arc Fitting

Position of option on menu: **Survey =>Edit =>Points =>Arc fitting start**

Arc fitting start starts arc fitting through from the selected vertex until the end of the string or an stop arc fitting field code is encountered for that string.

Selecting **Arc fitting start** brings up the **SDR Arc Fitting Start** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and vertices can be selected.*

Each time a vertex is picked and accepted, arc fitting is started at the selected vertex by inserting an ***Arc fitting start*** *field code.*

The option then repeats and continues until the **Finish** *button is selected.*

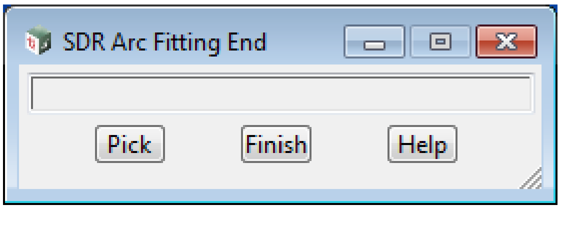
If the ***Pick*** *is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

17.7.8.6 End Arc Fitting

Position of option on menu: **Survey =>Edit =>Points =>Arc fitting end**

Arc fitting end ends any arc fitting for that string at the selected vertex.

Selecting **Arc fitting end** brings up the **SDR Arc Fitting End** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and vertices can be selected.*

Each time a vertex is picked and accepted, any arc fitting for the string is stopped at the selected vertex by inserting an ***Arc fitting end*** *field code.*

The option then repeats and continues until the **Finish** *button is selected.*

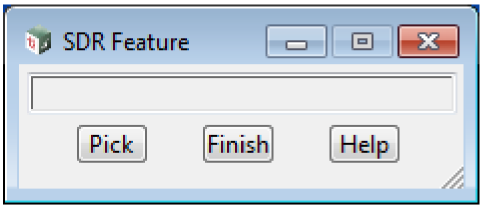
If the ***Pick*** *is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

17.7.8.7 Create a Feature String

Position of option on menu: **Survey =>Edit =>Points =>Circle feature**

Circle feature creates a feature string of a given radius at the selected vertex.

Selecting **Circle feature** brings up the **SDR Feature** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and vertices can be selected.*

Each time a vertex is picked and accepted, an ***Feature radius*** *typed input panel is displayed to type the radius into. A* ***Feature*** *code is then inserted into the 12d field file.*

The option then repeats and continues until the **Finish** *button is selected.*

If the ***Pick*** *is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

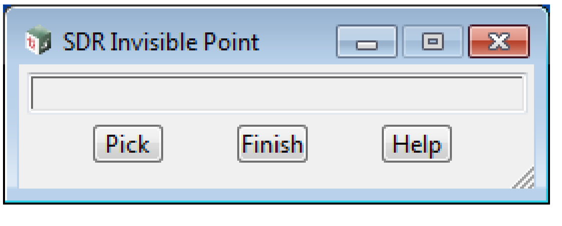
17.7.8.8 Make a Vertex Invisible

Position of option on menu: **Survey =>Edit =>Points =>Invisible**

Invisible turns the selected vertex invisible.

Even if the segments on either side of the invisible vertex ***are*** tagged as visible, the segments can not be drawn because the start/end point of the segment is invisible.

Selecting **Invisible** brings up the **SDR Invisible Point** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Pick button

As soon as the option is selected, the **Pick** *is active and vertices can be selected.*

Each time a vertex is picked and accepted, the vertex is turned invisible by inserting an ***Invisible Point*** *code into the 12d field file.*

The option then repeats and continues until the **Finish** *button is selected.*

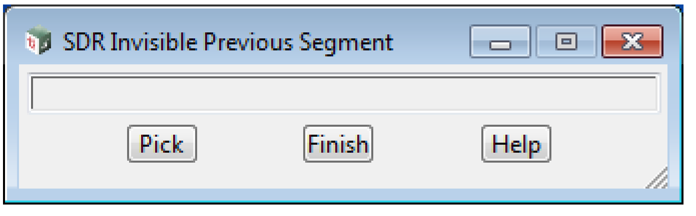
If the ***Pick*** *is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

17.7.8.9 Make the Previous Segment Invisible

Position of option on menu: **Survey =>Edit =>Points =>Invisible previous segment**

Invisible previous segment turns the segment ending on the selected vertex invisible. The vertex itself is not invisible.

Selecting **Invisible previous segment** brings up the **SDR Invisible Previous Segment** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and vertices can be selected.*

Each time a vertex is picked and accepted, the segment ending on the vertex is turned invisible by inserting an ***Invisible Previous Segment*** *code into the 12d field file.*

The option then repeats and continues until the **Finish** *button is selected.*

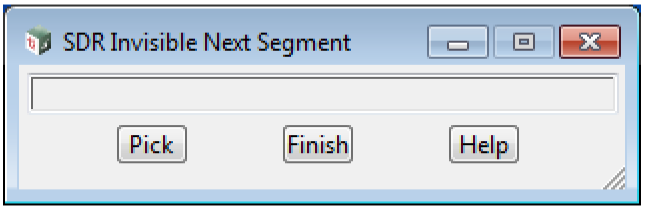
If the ***Pick*** *is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

17.7.8.10 Make the Next Segment Invisible

Position of option on menu: **Survey =>Edit =>Points =>Invisible next segment**

Invisible next segment turns the segment starting on the selected vertex invisible. The vertex itself is not invisible.

Selecting **Invisible next segment** brings up the **SDR Invisible Next Segment** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and vertices can be selected.*

Each time a vertex is picked and accepted, the segment starting on the vertex is turned invisible by inserting an ***Invisible Next Segment*** *code into the 12d field file.*

The option then repeats and continues until the **Finish** *button is selected.*

If the ***Pick*** *is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

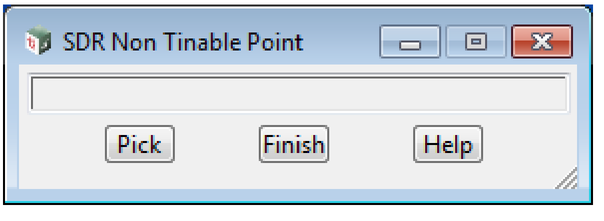
17.7.8.11 Make a Vertex Non Tinable

Position of option on menu: **Survey =>Edit =>Points =>Non tinable**

Non tinable sets the selected vertex to be non tinable (that is, the vertex is not included in any triangulation).

Even if the segments on either side of the non tinable vertex ***are*** tinable, the segments can not be included in any triangulation because the start/end point of the segment is non tinable.

Selecting **Non tinable** brings up the **SDR Non Tinable Point** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and vertices can be selected.*

Each time a vertex is picked and accepted, the vertex is set to non tinable (that is, the vertex is not included in any triangulations) by inserting a ***Non tinable point*** *code into the 12d field file.*

The option then repeats and continues until the **Finish** *button is selected.*

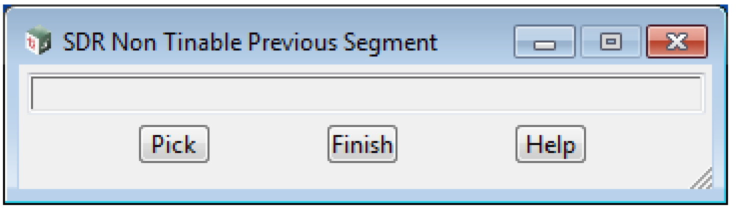
If the ***Pick*** *is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

17.7.8.12 Make the Previous Segment Non Tinable

Position of option on menu: **Survey =>Edit =>Points =>Non tinable previous segment**

Non tinable previous segment sets the segment ending on the selected vertex non tinable. That is, the segment is not included as a breakline in any tins (triangulations) The vertex itself can be tinable.

Selecting **Non tinable previous segment** brings up the **SDR Non Tinable Previous Segment** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and vertices can be selected.*

Each time a vertex is picked and accepted, the segment ending on the vertex is set to non tinable by inserting a ***Non tinable previous segment*** *code into the 12d field file.*

The option then repeats and continues until the **Finish** *button is selected.*

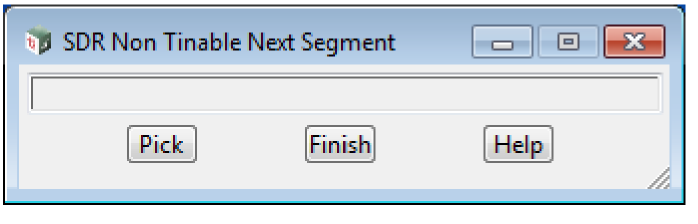
If the ***Pick*** *is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

17.7.8.13 Make the Next Segment Non Tinable

Position of option on menu: **Survey =>Edit =>Points =>Non tinable next segment**

Non tinable next segment sets the segment starting on the selected vertex non tinable. That is, the segment is not included as a breakline in any tins (triangulations) The vertex itself can be tinable.

Selecting **Non tinable next segment** brings up the **SDR Non Tinable Next Segment** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and vertices can be selected.*

Each time a vertex is picked and accepted, the segment starting on the vertex is set to non tinable by inserting an ***Non tinable next segment*** *code into the 12d field file.*

The option then repeats and continues until the **Finish** *button is selected.*

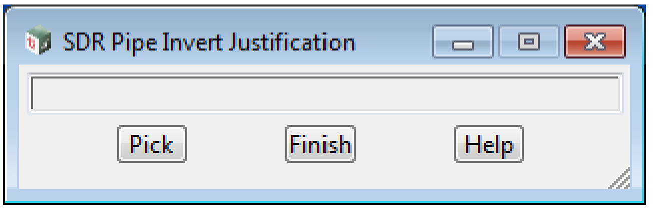
If the ***Pick*** *is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

17.7.8.14 Make a Vertex an Invert Level of a Pipe

Position of option on menu: **Survey =>Edit =>Points =>Pipe invert**

Pipe invert sets the z-value of the selected vertex to be used as an invert level (pipe top) when the vertex is in a pipe string.

Selecting **Pipe invert** brings up the **SDR Pipe Invert Justification** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and vertices can be selected.*

Each time a vertex is picked and accepted, the vertex is set to be an invert level (top of the pipe) for any pipe string it is in by inserting a ***Pipe invert*** *code into the 12d field file.*

The option then repeats and continues until the **Finish** *button is selected.*

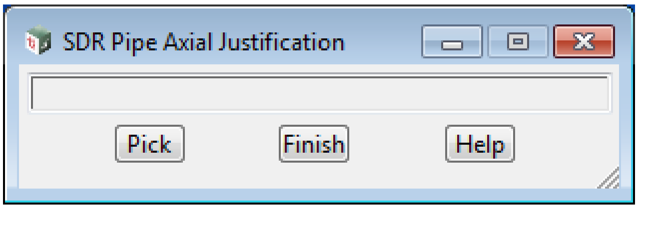
If the ***Pick*** *is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

17.7.8.15 Make a Vertex an Axial Level of a Pipe

Position of option on menu: **Survey =>Edit =>Points =>Pipe axial**

Pipe axial sets the z-value of the selected vertex to be used as an axial level (pipe centre) when the vertex is in a pipe string.

Selecting **Pipe axial** brings up the **SDR Pipe Axial Justification** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and vertices can be selected.*

Each time a vertex is picked and accepted, the vertex is set to be an axial level (centre of the pipe) for any pipe string it is in by inserting a ***Pipe axial*** *code into the 12d field file.*

The option then repeats and continues until the **Finish** *button is selected.*

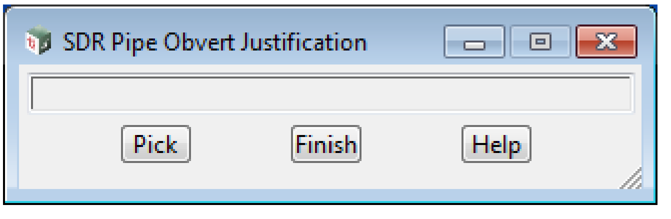
If the ***Pick*** *is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

17.7.8.16 Make a Vertex an Obvert Level of a Pipe

Position of option on menu: **Survey =>Edit =>Points =>Pipe obvert**

Pipe obvert sets the z-value of the selected vertex to be used as an obvert level (bottom of pipe) when the vertex is in a pipe string.

Selecting **Pipe obvert** brings up the **SDR Pipe Obvert Justification** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and vertices can be selected.*

Each time a vertex is picked and accepted, the vertex is set to be an obvert level (bottom of the pipe) for any pipe string it is in by inserting a ***Pipe obvert*** *code into the 12d field file.*

The option then repeats and continues until the **Finish** *button is selected.*

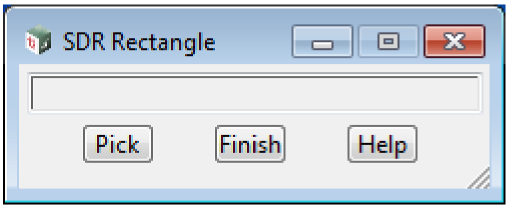
If the ***Pick*** *is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

17.7.8.17 Make a Parallelogram from the Last Three Points

Position of option on menu: **Survey =>Edit =>Points =>Rectangle last 3 points**

Rectangle last 3 points creates a parallelogram (squashed rectangle) from the selected vertex and the previous two vertices in the same string.

Selecting **Rectangle last 3 points** brings up the **SDR Rectangle** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and vertices can be selected.*

Each time a vertex is picked and accepted, a parallelogram (squashed rectangle) is created using the selected vertex and the previous two vertices of the same string (three vertices in total) and creating a four vertex to form the parallelogram of four vertices. The three vertices are removed from the original string and any following vertices are used in a new string.

*A **String rectangle** code is inserted into the 12d field file.*

The option then repeats and continues until the **Finish** *button is selected.*

*If the **Pick** is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

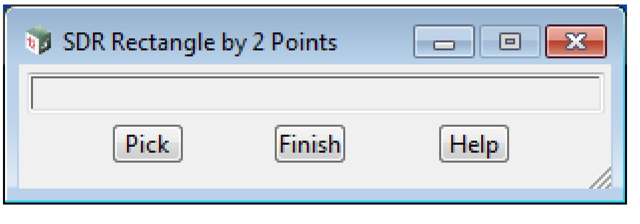
17.7.8.18 Make a Rectangle from the Last Two Points

Position of option on menu: **Survey =>Edit =>Points =>Rectangle last 2 points**

Rectangle last 2 points creates a rectangle from the selected vertex and the previous vertex in the same string, and a width supplied by the user.

The rectangle is formed by using the two vertices as the base of the rectangle and using the given width as the length of the other side of the rectangle. Two new vertices are created to from the rectangle.

Selecting **Rectangle last 2 points** brings up the **SDR Rectangle by 2 Points** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and vertices can be selected.*

*Each time a vertex is picked and accepted, a **Rectangle width** type input box is displayed and filled in.*

The rectangle is formed by using the two vertices as the base of the rectangle and using the given width as the length of the other side of the rectangle. Two new vertices are created to from the rectangle.

The two vertices are removed from the original string and any following vertices are used in a new string.

*A **String rectangle by 2 points** code is inserted into the 12d field file.*

The option then repeats and continues until the **Finish** *button is selected.*

*If the **Pick** is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

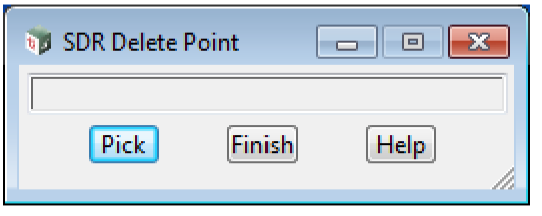
17.7.8.19 Delete a Vertex

Position of option on menu: **Survey =>Edit =>Points =>Remove point**

Remove point deleted the selected vertex.

The adjacent vertices in the string containing the deleted vertex will then be joined together.

Selecting **Remove point** brings up the **SDR Delete Point** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and vertices can be selected.*

Each time a vertex is picked and accepted, the vertex deleted from the field file.

The option then repeats and continues until the **Finish** *button is selected.*

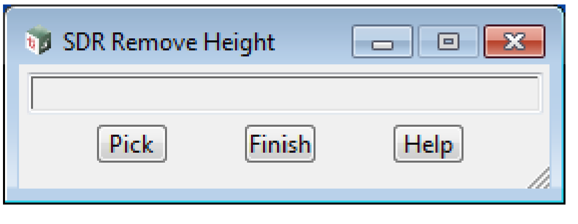
*If the **Pick** is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

17.7.8.20 Set a Vertex Height to Null

Position of option on menu: **Survey =>Edit =>Points =>Remove height**

Remove height sets the height of the selected vertex to null.

Selecting **Remove height** brings up the **SDR Remove Height** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and vertices can be selected.*

*Each time a vertex is picked and accepted, the height of the selected vertex is set to null by inserting a **Remove height** code into the 12d field file.*

The option then repeats and continues until the **Finish** *button is selected.*

If the **Pick** *is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

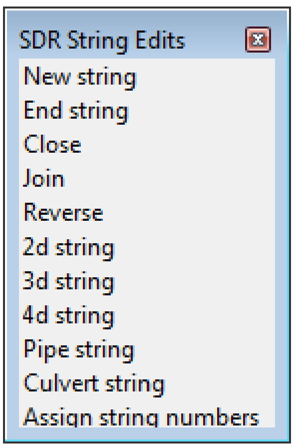
17.7.9 SDR Strings Edit

Position of menu: **Survey =>Edit =>Stringing**

The **Stringing** walk-right menu contains options that work on strings rather than individual points.

The **Stringing** walk-right menu is:

.



start a new string
end a string
close a string
join two strings
reverse a string
make a string 2d
make a string 3d
make a string 4d
make a string a pipe
make a string a culvert
assign string numbers

For the option *New string*, go to 17.7.9.1 New String
End string 17.7.9.2 End String
Close 17.7.9.3 Close String
Join 17.7.9.4 Join Two Strings of Same Feature Code
Reverse 17.7.9.5 Reverse String Direction
2d string 17.7.9.6 Make a 2d String
3d string 17.7.9.7 Make a 3d String
4d string 17.7.9.8 Make a 4d String
Pipe string 17.7.9.9 Make a Pipe String
Culvert string 17.7.9.10 Make a Culvert String
Assign string numbers 17.7.9.11 Assign String Numbers

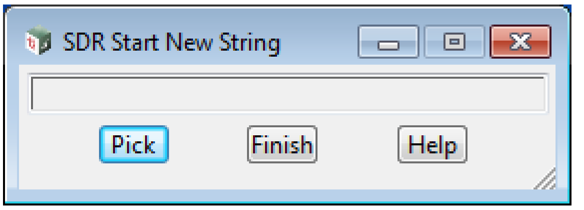
17.7.9.1 New String

Position of option on menu: **Survey =>Edit =>Stringing =>New string**

New string corrects the problem of not changing string numbers which creates the error of the last point of one string being joined to the first point of what should have been the next string. Hence there is a link between the two points that should not exist.

With **new string**, the user selects the incorrect link and a ***new string*** command is inserted into the 12d field file which removes the link by starting a new string at the end point of the link.

Selecting **New string** brings up the **SDR Start New String** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and string links can be selected. If the pick is cancelled for some reason, the* **Pick** *button can be used to restart the option.*

Each time a string segment is picked and accepted, a ***string start*** *command is inserted into the 12d field file which removes the link by starting a new string at the end point of the link.*

The option then repeats and continues until the **Finish** *button is selected.*

If the **Pick** *is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

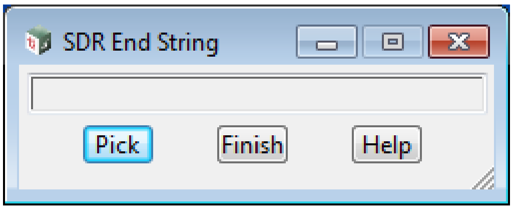
17.7.9.2 End String

Position of option on menu: **Survey =>Edit =>Stringing =>End string**

End string corrects the problem of not changing string numbers which creates the error of the last point of one string being joined to the first point of what should have been the next string. Hence there is a link between the two points that should not exist.

With **end string**, the user selects the incorrect link and a ***end string*** command is inserted into the 12d field file which removes the link by starting a new string at the end point of the link.

Selecting **End string** brings up the **SDR End String** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and string links can be selected. If the pick is cancelled for some reason, the* **Pick** *button can be used to restart the option.*

Each time a string segment is picked and accepted, an ***string end*** *command is inserted into the 12d field file which removes the link by starting a new string at the end point of the link.*

The option then repeats and continues until the **Finish** *button is selected.*

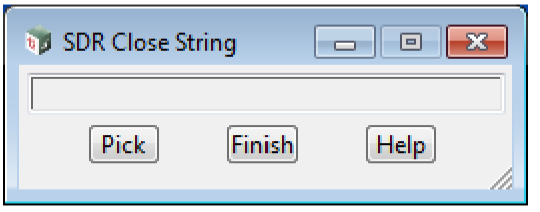
If the **Pick** *is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

17.7.9.3 Close String

Position of option on menu: **Survey =>Edit =>Stringing =>Close**

With **Close**, the user selects any part of the string to be closed and a ***close string*** command is inserted into the *12d field file*.

Selecting **Close** brings up the **SDR Close String** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and string can be selected. If the pick is cancelled for some reason, the* **Pick** *button can be used to restart the option.*

Each time a string is picked and accepted, a ***string close*** *command is inserted in the 12d field file which closes the selected string.*

The option then repeats and continues until the **Finish** *button is selected.*

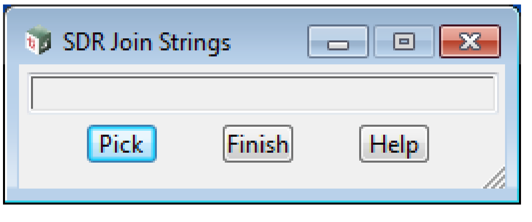
If the ***Pick*** *is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

17.7.9.4 Join Two Strings of Same Feature Code

Position of option on menu: **Survey =>Edit =>Stringing =>Join**

With **Join**, the two strings of the same name (code) are selected with direction and the strings are joined by inserting the appropriate ***join string*** command in the *12d field file.*

Selecting **Join** brings up the **SDR Join Strings** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and string can be selected. If the pick is cancelled for some reason, the* **Pick** *button can be used to restart the option.*

*Each time two strings of the same name (code) are picked with direction, the appropriate join string field code (**Join first points of strings**, **Join first to last point of strings**, **Join last points of strings**, **Join last to first point of strings**) is inserted into the 12d field file.*

The option then repeats and continues until the **Finish** *button is selected.*

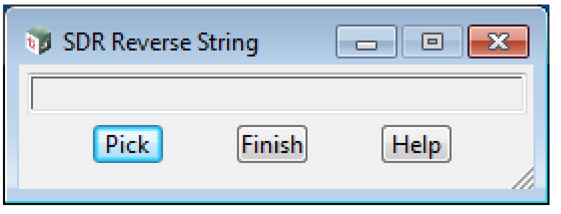
*If the **Pick** is cancelled by hitting the <Esc> key or selecting **Cancel** from the **Pick Ops** menu, the **Pick** button can be used to restart the option.*

17.7.9.5 Reverse String Direction

Position of option on menu: **Survey =>Edit =>Stringing =>Reverse**

With **Reverse**, the selected strings is reversed by inserting the ***String reverse*** command into the *12d field file.*

Selecting **Reverse** brings up the **SDR Reverse String** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

*As soon as the option is selected, the **Pick** is active and string can be selected. If the pick is cancelled for some reason, the **Pick** button can be used to restart the option.*

*Each time a string is selected, a **String reverse** field code is inserted into the 12d field file.*

*The option then repeats and continues until the **Finish** button is selected.*

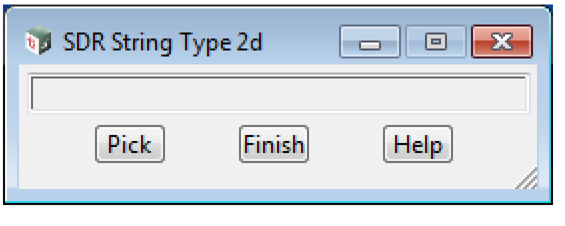
*If the **Pick** is cancelled by hitting the <Esc> key or selecting **Cancel** from the **Pick Ops** menu, the **Pick** button can be used to restart the option.*

17.7.9.6 Make a 2d String

Position of option on menu: **Survey =>Edit =>Stringing =>2d string**

With **2d string** the user selects any vertex of a string and a ***string type 2d*** command is then inserted into the *12d field file*.

Selecting **2d string** brings up the **SDR String Type 2d** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

*As soon as the option is selected, the **Pick** is active and string can be selected. If the pick is cancelled for some reason, the **Pick** button can be used to restart the option.*

Each time a string is picked and accepted, a ***String type 2d*** *command is inserted in the 12d field file.*

The option then repeats and continues until the **Finish** *button is selected.*

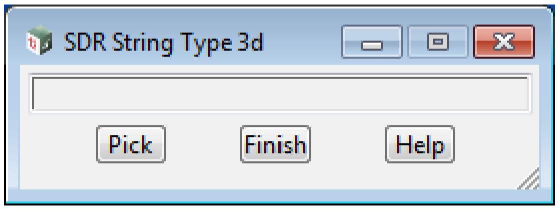
If the ***Pick*** *is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

17.7.9.7 Make a 3d String

Position of option on menu: **Survey =>Edit =>Stringing =>3d string**

With **3d string** the user selects any vertex of a string and a ***string type 3d*** command is then inserted into the *12d field file*.

Selecting **3d string** brings up the **SDR String Type 3d** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and string can be selected. If the pick is cancelled for some reason, the* **Pick** *button can be used to restart the option.*

Each time a string is picked and accepted, a ***String type 3d*** *command is inserted in the 12d field file.*

The option then repeats and continues until the **Finish** *button is selected.*

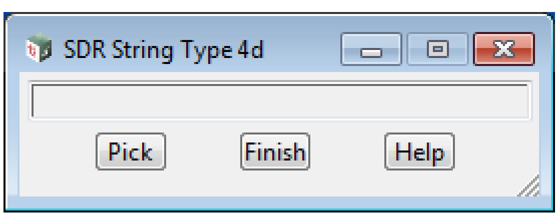
If the ***Pick*** *is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

17.7.9.8 Make a 4d String

Position of option on menu: **Survey =>Edit =>Stringing =>4d string**

With **4d string** the user selects any vertex of a string and a ***string type 4d*** command is then inserted into the *12d field file*.

Selecting **4d string** brings up the **SDR String Type 4d** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Pick button

As soon as the option is selected, the **Pick** *is active and string can be selected. If the pick is cancelled for some reason, the* **Pick** *button can be used to restart the option.*

Each time a string is picked and accepted, a ***String type 4d*** *command is inserted in the 12d field file.*

The option then repeats and continues until the **Finish** *button is selected.*

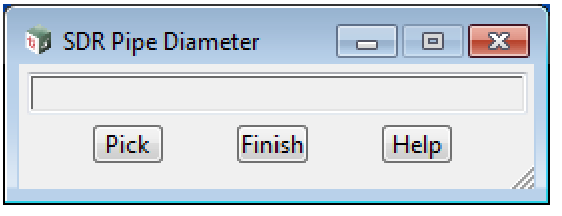
If the ***Pick*** *is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

17.7.9.9 Make a Pipe String

Position of option on menu: **Survey =>Edit =>Stringing =>Pipe string**

With **Pipe string**, the user selects any vertex of a string, gives a pipe diameter and then a ***string type pipe*** command is inserted into the *12d field file*.

Selecting **Pipe string** brings up the **SDR Pipe Diameter** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	button		

As soon as the option is selected, the **Pick** *is active and string can be selected. If the pick is cancelled for some reason, the* **Pick** *button can be used to restart the option.*

Each time a string is picked and accepted, a ***Pipe diameter*** *text input is displayed and after a value is typed in, a* ***String type pipe*** *command is inserted in the 12d field file.*

The option then repeats and continues until the **Finish** *button is selected.*

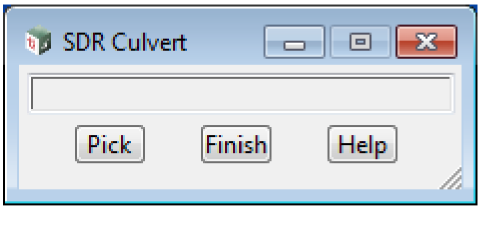
If the ***Pick*** *is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

17.7.9.10 Make a Culvert String

Position of option on menu: **Survey =>Edit =>Stringing =>Culvert string**

With **Culvert string**, the user selects any vertex of a string, gives a culvert width and height and then a ***string type culvert*** command is inserted into the *12d field file*.

Selecting **Culvert string** brings up the **SDR Culvert** panel:



The fields and buttons used in this panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Pick | button

As soon as the option is selected, the **Pick** *is active and string can be selected. If the pick is cancelled for some reason, the* **Pick** *button can be used to restart the option.*

Each time a string is picked and accepted, ***Culvert width*** *and* ***Culvert height*** *text inputs are displayed and after values are typed in, a* ***String type culvert*** *command is inserted in the 12d field file.*

The option then repeats and continues until the **Finish** *button is selected.*

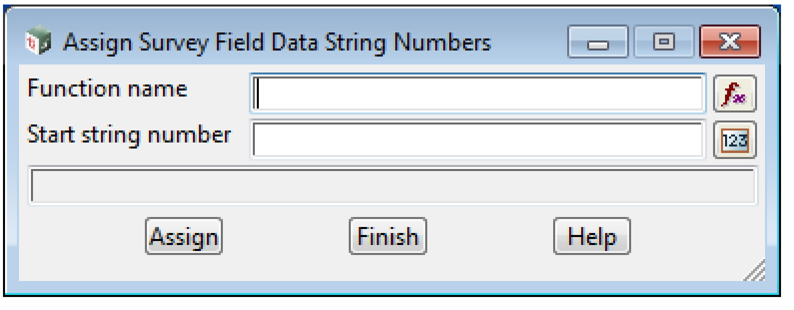
If the **Pick** *is cancelled by hitting the <Esc> key or selecting* **Cancel** *from the* **Pick Ops** *menu, the* **Pick** *button can be used to restart the option.*

17.7.9.11 Assign String Numbers

Position of option on menu: **Survey =>Edit =>Stringing =>Assign string numbers**

Assign string numbers xxx

Selecting **Assign string numbers** brings up the **Assign Survey Field Data String Numbers** panel:



The fields and buttons used in this panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Function name | function box | | SDR functions

name of the SRD function to give string numbers to.

Start string number

string number to start numbering at.

Assign | button

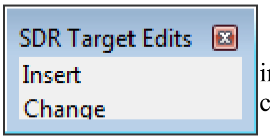
17.7.10 Target Height

Position of menu: **Survey =>Edit =>Target height**

The **Target height** option is used to correct target height errors. This includes giving an incorrect value for a target height or forgetting to change the target height.

The **Target height** walk-right menu is

.



insert an new target height
change existing target height

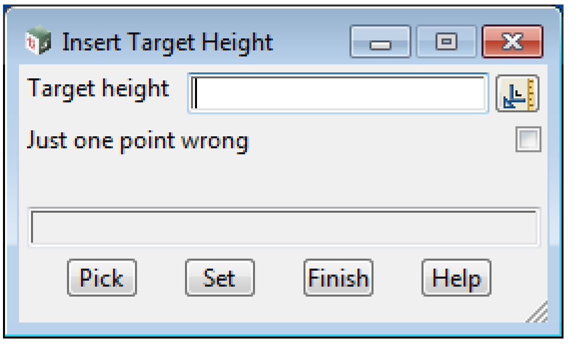
Each option will now be described in detail.

17.7.10.1 Insert Target Height

Position of option on menu: **Survey =>Edit =>Target height =>Insert**

The **insert** option inserts a new target height command into the 12d field file at a user selected point. The user can choose whether the new target height only applied for that one point or for all following points until a new target height is set.

Selecting **insert** brings up the **insert target height** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Target height	input/output		

when a point is selected, the target height for that point is displayed. The value can then be changed by the user and if **set** *is selected, the target height is modified and all affected points updated.*

Just one point tick

if ticked, the new target height only applied for that one point and then reverts pack to the previous target height.
if not tick, the new target height applies for all following points until a new target height is set.

Pick button

pick the point to insert a new target height.

Set button

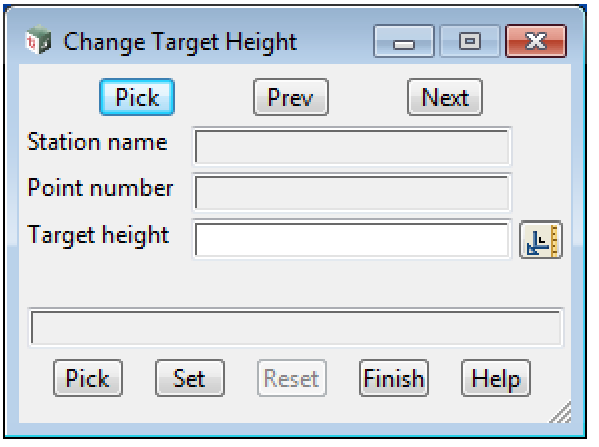
selecting **set** *inserts the new target height command into the 12d field file.*

17.7.10.2 Change Target Height

Position of option on menu: **Survey =>Edit =>Target height =>Change**

The **Change** option changes an existing target height to a new value.

Selecting **Change** brings up the **change target height** panel:



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Pick button

select a point whose information will be displayed in the panel fields. A highlighted line joining the point to the station that it was surveyed from is also displayed.

Prev button

move to the previous point in the string.

Next button

move to the next point in the string.

Station name display only

name of the station that the point was surveyed from.

Point ID display only

point ID of the selected point.

Target height input/output

when a point is selected, the target height for that point is displayed. The value can then be changed by the user and if **set** *is selected, the target height is modified and all affected points updated.*

Pick button

pick the point to change the target height.

Set button

selecting **set** *changes the target height command in the 12d field file and updates affected points.*

Reset button

reset *undoes the changes.*

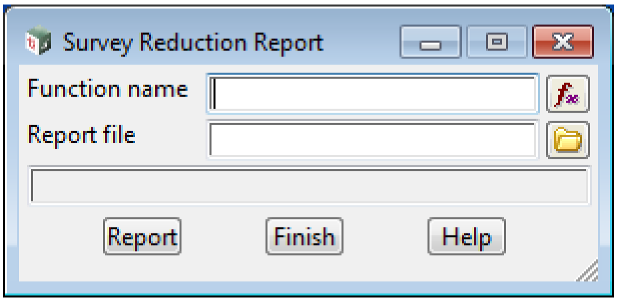
17.8 Report

Position of option on menu: **Survey =>Report**

Report produces a report on the field data from a *Survey* function.

Selecting **report** brings up the **survey reduction report** panel

The fields and buttons used in this panel have the following functions.



Field Description	Type	Defaults	Pop-Up
Function name	input		available Survey functions

name of the 12d Survey function to report on.

Report file	input		*.rpt

name of the file for the Survey function report.

Report	button		

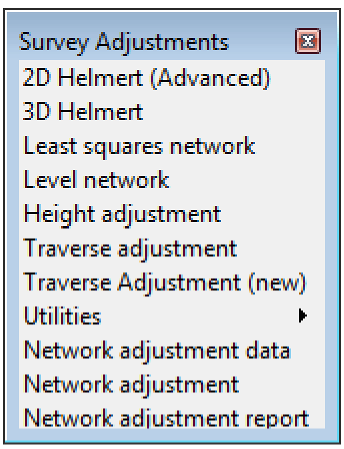
create the Survey report.

17.9 Adjustments

Position of menu: **Survey =>Adjustments**

The **Adjustments** walk-right menu is

.



2D Helmert with fixed scale
3D Helmert
least squares network adjustment
adjustment of level nets
adjust heights of data
traverse adjustment
traverse adjustment (new)
various utilities for massaging Leica files

For *2D Helmert (Advanced)*, go to 28.10.4.1 2D Helmert (Advanced)
3D Helmert 28.10.4.2 3D Helmert
Least squares network 17.9.3 Least Squares Network
Level network 17.9.4 Level Network
Height adjustment 17.9.5 Height Adjustment
Traverse Adjustment 17.9.6 Traverse Adjustment
Traverse Adjustment (new) 17.9.6 Traverse Adjustment
Utilities 17.9.7 Survey Adjustments Utilities

17.9.1 2D Helmert (Advanced)

Position of option on menu: **Survey =>Adjustments =>2D Helmert (Advanced)**

A ***2D Helmert*** transformation is a two dimensional linear transformation consisting of a scaling, rotation and a 2D- translation (shift) of data. Hence there are flour parameters to be specified.

The 2D Helmert (Advanced) option calculated and applied a Helmert transformation, and also allows the user to ***fix the scale*** for the transformation.

For information on the 2D Helmert transformation, please go to the section 28.9.1 Coordinate Transformations - Helmert and Affine

The *2D Helmert (Advanced)* panel is documented in the section 28.10.4.1 2D Helmert (Advanced)

17.9.2 3D Helmert

Position of option on menu: **Survey =>Adjustments =>3D Helmert**

A ***3D Helmert*** transformation is a three dimensional linear transformation consisting of a scaling, three rotations, and a 3D-translation (shift) of data. Hence there are 7 parameters to be specified.

For information on the 3D Helmert transformation, please go to the section 28.9.1 Coordinate Transformations - Helmert and Affine

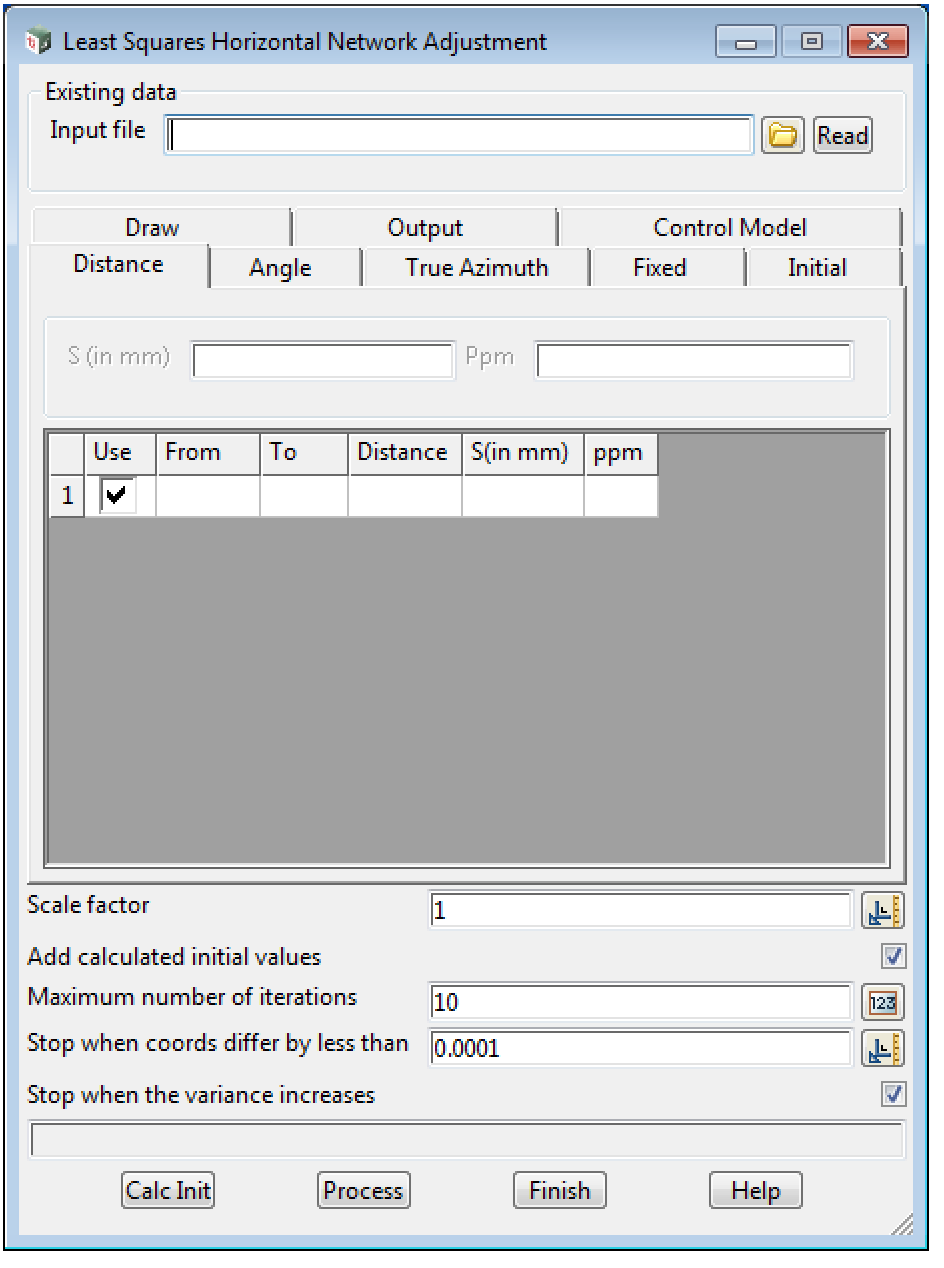
The *3D Helmert* panel is documented in the section 28.10.4.2 3D Helmert

17.9.3 Least Squares Network

Position of option on menu: **Survey =>Adjustments =>Least squares network**

The **Least square network** option performs a least square adjustment on a set of observations that can be manually entered into predefined grid box columns or by reading an existing input file. These observations can be of the form of distances, angles, azimuths or a combination of these.

Selecting**Least squares network** brings up the **Least Squares Horizontal Network Adjustment** panel



The fields and buttons used in this panel have the following functions.

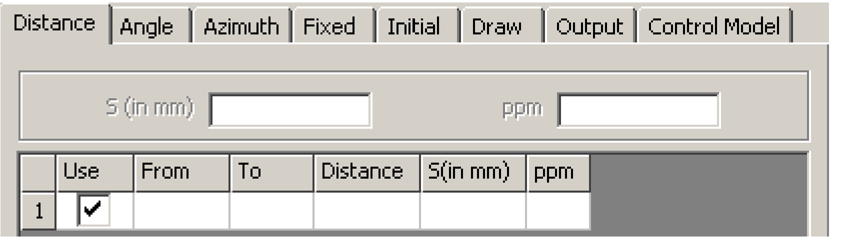
Field Description	Type	Defaults	Pop-Up
Input file	input box		*.in

a pre existing .in input file can be specified. For the format of the .in file, go to Format of the LSA .in File

Field Description	Type	Defaults	Pop-Up
Read	button		

on selection of the read button, if a valid .in input file has been specified, the relevant parameters will be read into the grid controls.

Distance tab



The fields and buttons used in this tab have the following functions.

Field Description	Type	Defaults	Pop-Up
S(in mm)	input box		

standard deviation of a single distance measurement in mm. If entered the value will be used for all lines in the grid. This eliminates the need to put the standard deviation on each line.

ppm button

parts per million error. If entered the value will be used for all lines in the grid. This eliminates the need to put the ppm on each line.

the grid control values can be entered using valid inputs into the various fields:

Use - *if* ***ticked****, the row is used in the adjustment. If* ***not ticked*** *then the row is ignored*

From *point id which represents the point from which the distance was read*

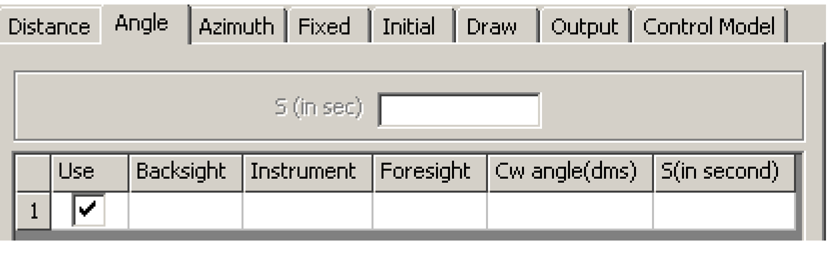
To *point id which represents the point to which the distance was read*

Distance *(horizontal) with all corrections having been made except for that of scale.*

S(in mm) *- if not blank, the standard deviation to use for this single distance measurement in millimetres. If blank, the default value in the panel field* **S(in mm)** *is used.*

ppm *- if not blank, the parts per million (ppm) error to use for this single distance measurement. If blank, the default value in the panel field* **Pmm** *is used.*

Angle tab



The fields and buttons used in this tab have the following functions.

Field Description	Type	Defaults	Pop-Up
S(in sec)	input box		

standard deviation for a single angle measurement in seconds of arc. If entered the value will be used for all lines in the grid. This eliminates the need to put the standard deviation on each line.

the grid control values can be entered using valid inputs into the various fields. Angle input must be in the form of degrees minutes and seconds in 4.17.1 HP Notation *(e.g. 12.1055)*

Use - *if* ***tick****, the row is used in the adjustment. If* ***not ticked*** *then the row is ignored*

Backsight *- the backsight point id is entered.*

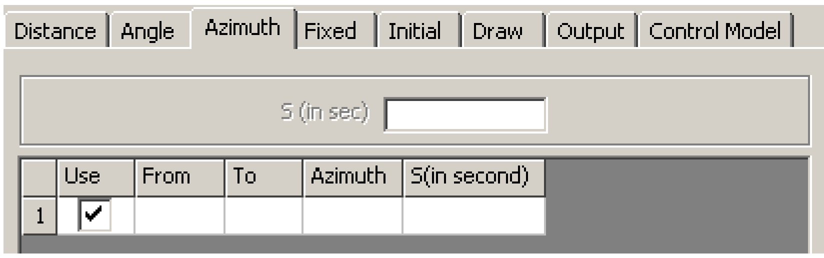
Instrument - *the point id of the instrument or observation station is entered.*

Foresight - *the point id of the foresight station is entered into the* ***Foresight*** *column.*

Angle (dms) -*. the angle defined by the clockwise measurement from the backsight to foresight station as observed from the instrument station.Valid ranges of 0 -360 degrees.*

S (in second) - *if non blank, the standard deviation of the angular measurement in seconds of arc to use for this single angle measurement. If blank, the default value in the panel field* **S (in sec)** *is used.*

Azimuth tab



The fields and buttons used in this tab have the following functions.

Field Description	Type	Defaults	Pop-Up
S(in sec)	input box		

standard deviation for a single azimuth measurement in seconds of arc. If entered the value will be used for all lines in the grid. This eliminates the need to put the standard deviation on each line.

the grid control values can be entered using valid inputs into the various fields.

Use - *if* ***ticked****, the row is used in the adjustment. If* ***not ticked*** *then the row is ignored*

From *- the instrument station point id should be entered.*

To *- the point id of the observed station is entered.*

Azimuth *- the true azimuth to the observed point is entered. Azimuth input should be in the form of degrees minutes and seconds (e.g. 12.1055) Valid ranges of 0 -360 degrees.*

S(in second) - *if non blank, the standard deviation of the azimuth in seconds of arc to use for this single angle measurement. If blank, the default value in the panel field* **S (in sec)** *is used.*

Fixed tab

the grid control values can be entered using valid inputs into the various fields.

Use - *if* ***ticked****, the row is used in the adjustment. If* ***not ticked*** *then the row is ignored*

Point name *- the point id of the fixed stations*

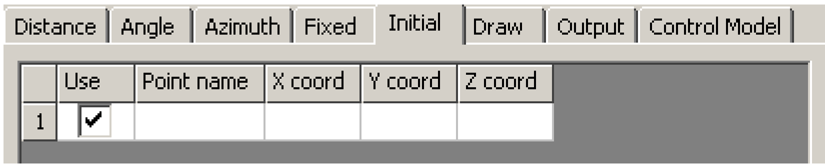
X coord *- the fixed x coordinate value is entered*

Y coord *- the fixed y coordinate value is entered*

Z coord *- the fixed z coordinate value is entered*

Initial tab

12d Model *will try and calculate initial position for each point from the entered measurement but for* *some points this is not possible and then the user must enter an initial position.*



the grid control values can be entered using valid inputs into the various fields.

***Use -** if **tick**, the row is used in the adjustment. If **not ticked** then the row is ignored*

***Point name** - the point id of a free or floating station should be entered.*

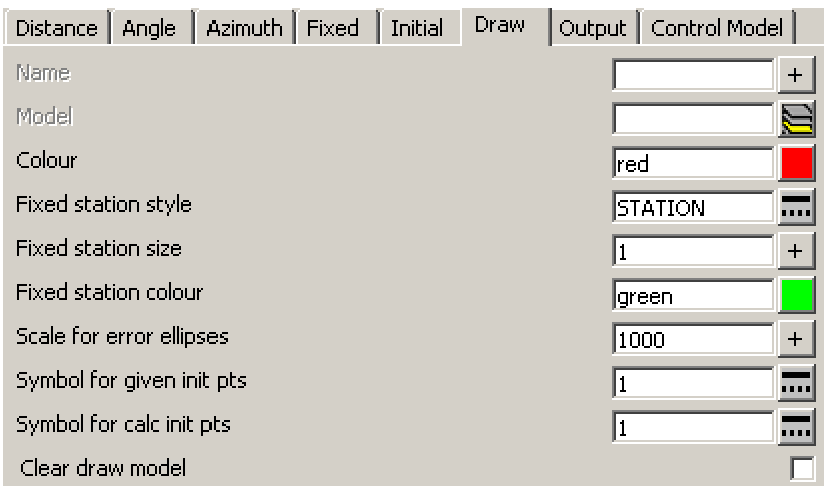
***X coord** - the initial x coordinate estimate is entered.*

***Y coord** - the initial y coordinate estimate is entered.*

***Z coord** - the initial z coordinate estimate is entered.*

NB. It is very important that the initial values are approximately equal to the true values. A solution will still be generated for rubbish values.

Draw tab



the following values can be entered to enable the production of a user defined plot of the adjustment.

Field Description	Type	Defaults	Pop-Up
Name	name box		defined names from names.4d file

if non-blank, the name of the new string

Model	model box		available models

if non-blank, the model of the new string (including error ellipse and station symbol if specified).

Colour	colour box	red	available colours

if non-blank, the colour of the new string. If blank, the original string colour is used.

Fixed station style	input	Station	available line styles

line style of the station symbol. These will be placed at all of the fixed stations if specified.

Fixed station size	input	1	

size of symbol in world units.

Fixed station colour	colour box	green	available colours

if non-blank, the colour of the station symbols

Scale for error ellipses input 1000

this value will be applied to any error ellipses to enable an exaggerated view of the ellipse geometry

Symbol for given init points input 1 available line styles

line style of the given initial point symbols

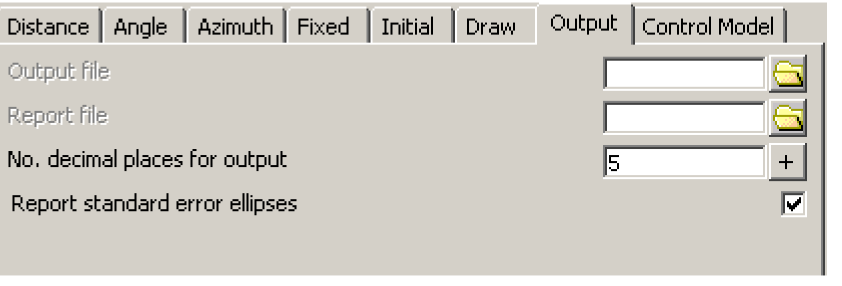
Symbol for calc init points input 1 available line styles

line style of the calculated initial point symbols

Clear draw model beforehand tick box

if ticked, the draw model will be cleaned before the new strings are created

Output tab



The fields in this panel define the output parameters.

Field Description Type Defaults Pop-Up

Output file input box *.in

if non blank, an output file is produced that can be read in the future.

Report file input box *.rpt

if non blank, a report file of this name is creating giving the calculation specific parameters and result.

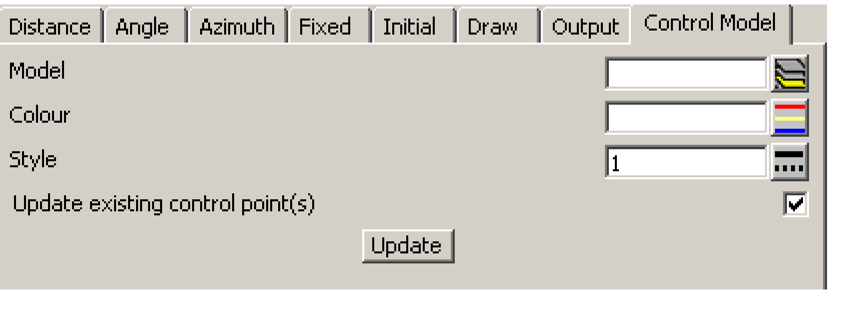
No. decimal places for output input box 5

the number of decimal place to use in the report

Report standard error ellipse tick box ticked

*if **ticked**, error ellipses for each new point are created*

Control Model tab



Model model box

a new model or an existing model containing some control points an be entered.

If a new model, the calculated points will placed into that control model.
*If an existing model, if there is a point in the model with the same point id as a point being calculated, then that rather than a new point being created, the coordinates of the existing point can be updated after the reduction is complete (depends on the **Update existing controls points** tick box).*

Colour model box

colour used for control station symbol

Style linestyle box

style for control stations.

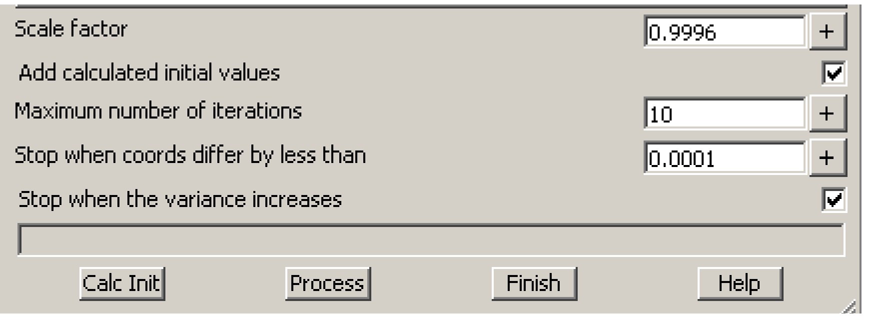
Update existing control points tick box ticked

if ticked, any existing coordinate positions from the adjustment will be updated in the given control model.

Update button

if pressed, the updating of changed level values in the given control model will be carried out.

The remaining fields refer to those located below the grid control



Scale factor input box project central meridian scale factor

a scale factor can be defined which will be applied to any entered distances. Final distances used in calculations are derived by multiplying the entered distance by the scale factor.

Add calculated initial values tick box tick

if this box is checked, then the initial values will be included in the report

Maximum number of iterations input box 10

the calculation of the adjustment can be aborted if a solution is not found after the specified number of iterations

Stop when coords differ... input box 0.0001

the calculation of the adjustment is stopped when the difference between successive calculations meet the defined tolerance

Stop when the variance increases tick box tick

*if **ticked**, the calculation of the adjustment is stopped when the variance between successive calculations increases*

Process button

run the adjustment

Format of the LSA .in File

The data in each line of the *.in* file is space delimited. That is, each piece of data on the line is separated by one or more spaces.

Line 1:

The layout of the data on the first line is as follows:

Position	Description	Variable	Type
1	Number of distance observations	N-Dist	Integer
2	Number of angle observations	N-Ang	Integer
3	Number of true azimuth observations	N-Azi	Integer
4	Number of control (fixed) stations	N-Fix	Integer
5	Number of stations (including control stations)	N-Stn	Integer
6	Distance standard deviation (positive if used)	D-Std	Double
7	Distance part per million (non-negative if used)	D-Ppm	Double
8	Angle standard deviation (positive if used)	A-Std	Double
9	Azimuth standard deviation (positive if used)	Z-Std	Double
10	Scale factor		Double

Note: D-Std, is in millimetres, A-Std and Z-Std are in seconds.

Example:

63 71 0 16 35 5.00000 5.00000 15.00000 5.00000 1.00000

Next ***N-Stn*** lines:

The next ***N-Fix*** lines are *control station* records, followed by (***N-Stn - N-Fix***) *station* records.

The layout of the line of a *station record* is as follows:

Position	Description	Type
1	Flag (1 if the station is used, otherwise 0)	Integer
2	Point number	Text
3	X-coord	Double
4	Y-coord	Double

Example:

1 9338 3889.23700 5341.54800

….

….

0 9712a 4007.45700 5168.68700

….

Next ***N-Dist*** lines

The next ***N-Dist*** lines are *distance observation* records.

The layout of the line of a *distance observation record* is as follows:

Position	Description	Type
1	Flag (1 if the observation is used, otherwise 0)	Integer
2	Start point number	Text
3	End point number	Text
4	Distance	Double
5	Standard deviation (in meter, negative if not used)	Double
6	Part per million*	Double

Note: * Part per million is different to D-Ppm by a factor of 1000. That is, if you want a ppm value of 5, the number in the file should be 0.005

Example:

1 9542 9702 100.20200 0.00300 0.00200

1 9542 9703 142.98200 -1.00000 -1.00000

0 9700 9506 72.10700 -1.00000 -1.00000

….

Next *N-Ang* lines

The next ***N-Ang*** lines are *angle observation* records.

The layout of the line of a *angle observation record* is as follows:

Position	Description	Type
1	Flag (1 if the observation is used, otherwise 0)	Integer
2	Backsight point number	Text
3	Instrument point number	Text
4	Foresight point number	Text
5	Degree	Integer
6	Minute	Integer
7	Second	Double
8	Standard deviation (in mm, negative if not used)	Double

Example:

1 19206 9506 9701 174 29 20.00000 -1000.00000

….

Next *N-Azi* lines

The next ***N-Azi*** lines are *true azimuth observation* records.

The layout of the line of a *true azimuth observation record* is as follows:

Position	Description	Type
1	Flag (1 if the observation is used, otherwise 0)	Integer
2	Backsight point number	Text
3	Instrument point number	Text
4	Foresight point number	Text
5	Degree	Integer
6	Minute	Integer
7	Second	Double
8	Standard deviation (in mm, negative if not used)	Double

Example:

1 19206 9506 174 29 20.00000 3.00000

….

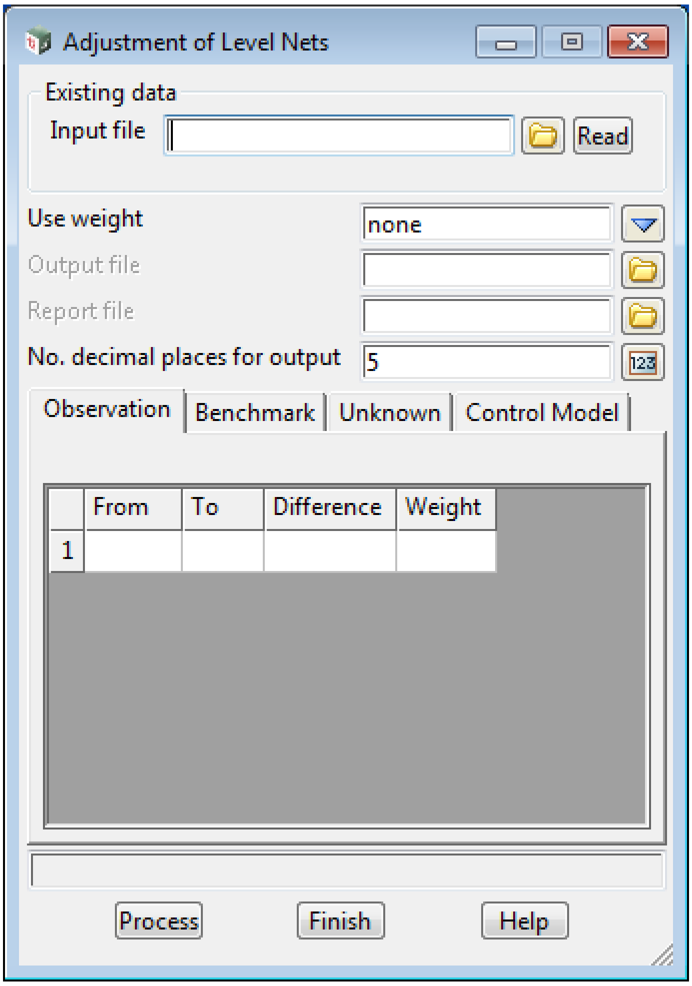
That is the end of data in the file.

17.9.4 Level Network

Position of option on menu: **Survey =>Adjustments =>Level network**

The **level network** option performs a least square adjustment on a set of level observations that can be manually entered into predefined grid box columns or by reading an existing input file.

Selecting **Level network** brings up the **Adjustment of Level Nets** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Input file	input box		*.in

a pre existing input file can be specified.

Read	button		

on selection of the read button, if a valid input file has been specified, the relevant parameters will be read into the grid controls.

Use weight	choice box	none	none, distance, standard deviation

*a weighting per observation can be made according to the **none**, **distance**, and **standard deviation** choices*
***None -** no weighting will be applied.*

***Distance** - if the distance of the level network legs are to be used. The horizontal distance is entered into the weight field. eg 101.23*

***Standard deviation** - to define a standard deviation. e.g. 0.005*

***Note:** The method of defining the weight should be kept consistent. i.e. one of the 3 options.*

Output file	input box		*.in

an output file can be defined which will allow for the reading of the file in the future.

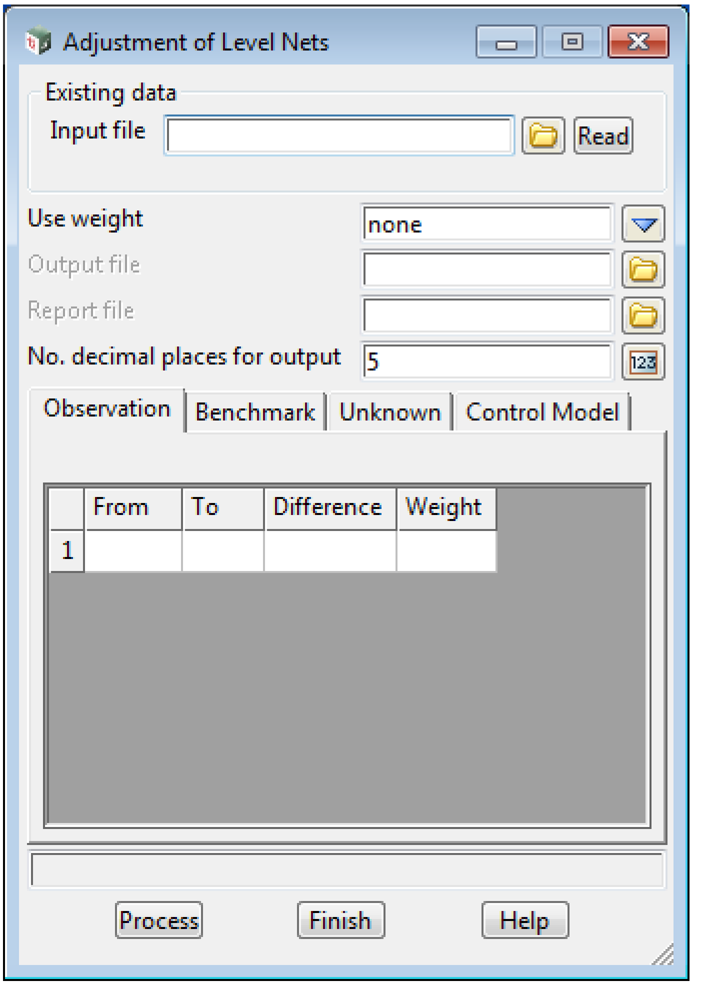
Report file			*.rpt files

a report using the nominated file name is created.

No. decimal places for out,,,input box 5

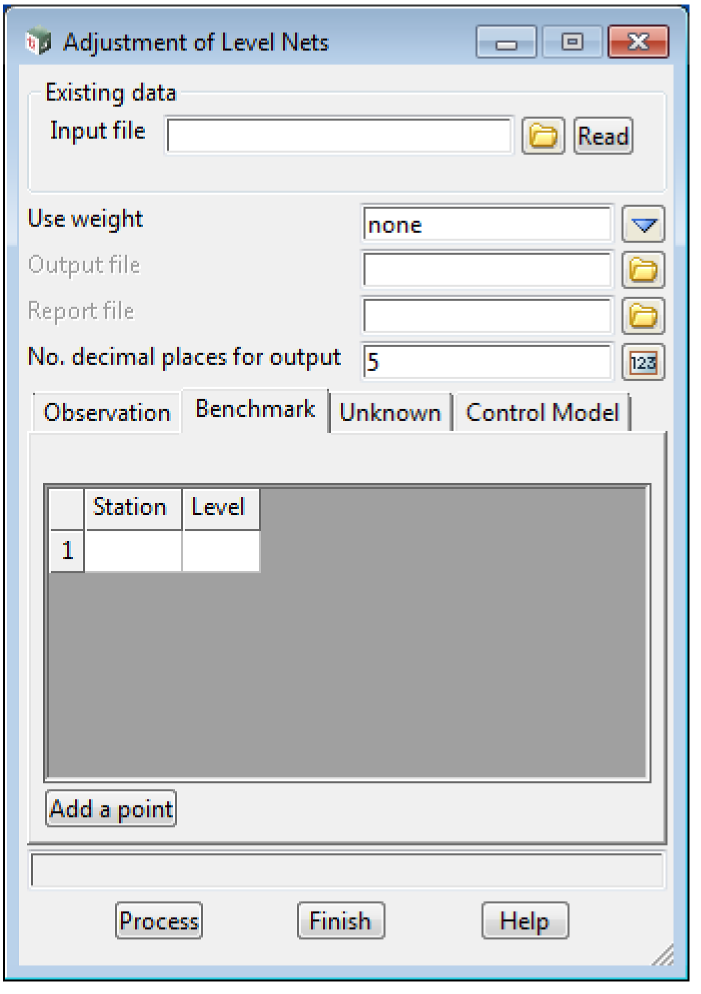
the number of decimal places for the new levels can be defined.

Observation tab



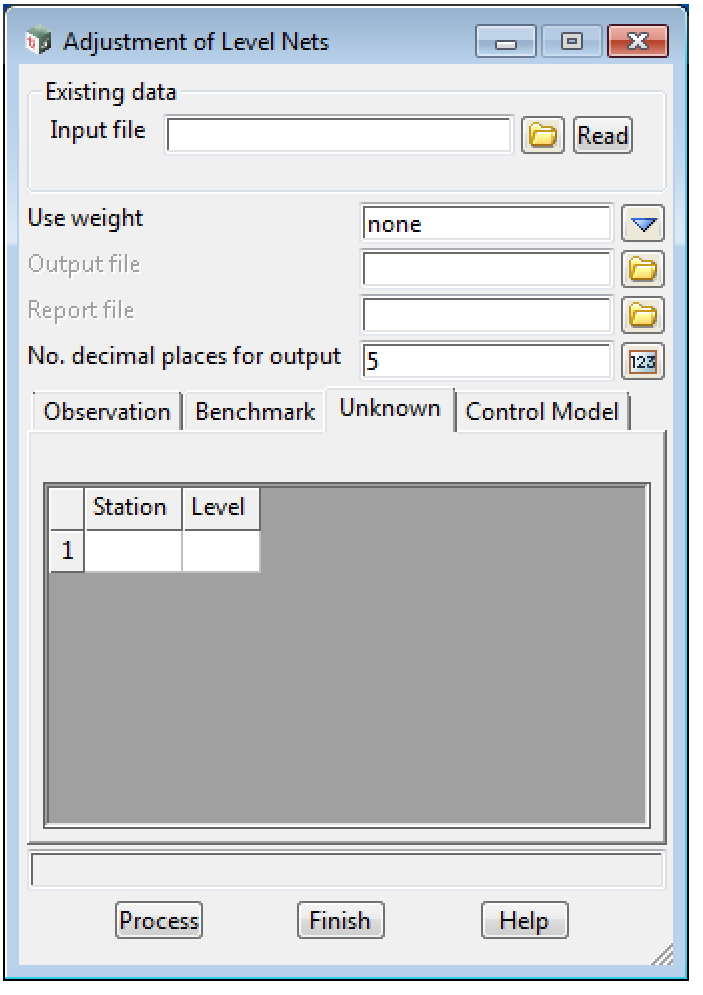
the grid control values can be entered using valid inputs into the various fields:
From *point id which represents the first point of the level difference*
To *point id which represents the second point to which the level difference refers*
Difference *the difference between the levels defined by the* ***From*** *point and the* ***To*** *point. i.e. the difference = (to point) level - (from point) level*

Benchmark tab



the grid control values can be entered using valid inputs into the various fields.
Station*. The point id of a fixed level station should be entered.*
Level*. The corresponding level value of the fixed station.*

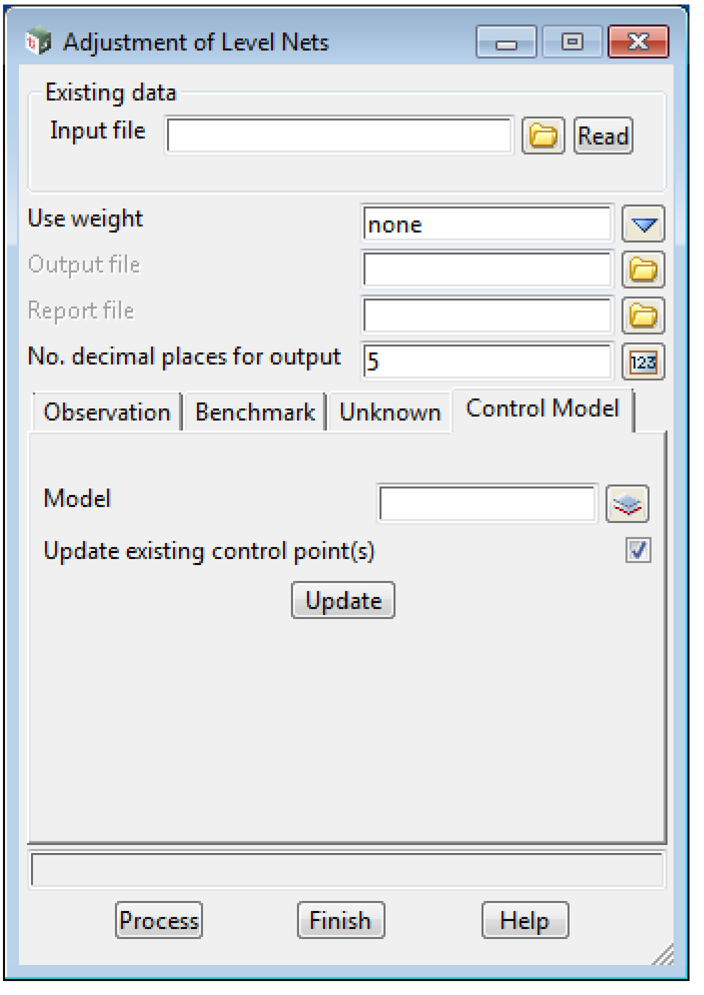
Unknown tab



the grid control values can be entered using valid inputs into the various fields.
Station*. The point id of any free or floating level stations should be entered.*
Level*. The corresponding approximate level value of free or floating level station.*

a report using the nominated file name is created.

Control Model tab



Model model box

an existing control model can be entered so that changes to level values can be updated.

Update existing control points tick box ticked

if ticked, any new level values from the adjustment will be updated in the given control model.

Update button

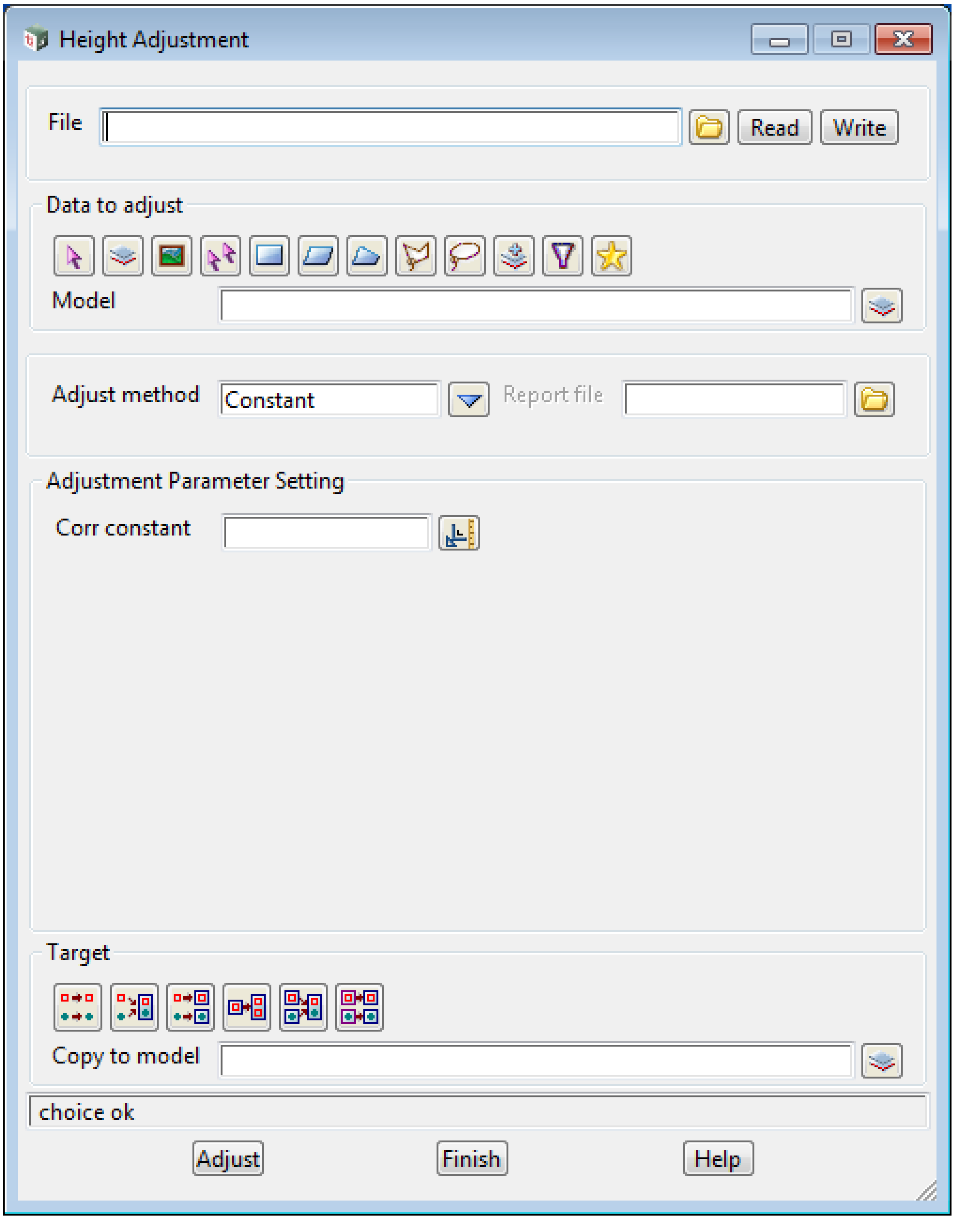
if pressed, the updating of changed level values in the given control model will be carried out.

17.9.5 Height Adjustment

Position of option on menu: **Survey =>Adjustments =>Height adjustment**

The **Height adjustment** option adjusts the selected data using either a constant z value, a plane defined by parameters, a plane defined by points (least square fit) or a tin of height differences.

Selecting **Height adjustment** brings up the **Height Adjustment** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
File	file box		*.haf files

a filename can be specified for reading or writing a file.

Read	button		

if a valid file exists, the file contents can be loaded into the panel.

Write	button		

if a valid name is specified, the user can write the input data to a file.

Data to adjust	input		

the source of the data to be adjusted is selected using the data source box. For more information on the data source box see 4.19.3 Data Source

Adjustment method choice box Constant Constant
Plane parameters
Plane by points
Difference tin

the adjustment method should be selected from the list. A description of each method is given below.

Report file *.rpt files

a report using the nominated file name is created.

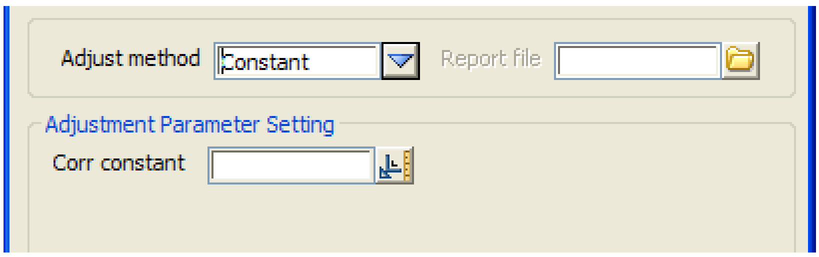
Target target

the target for the adjusted data should be specified. For more information on how to use a data target box see the section 4.19.4 Data Target

Transform button

perform the transformation.

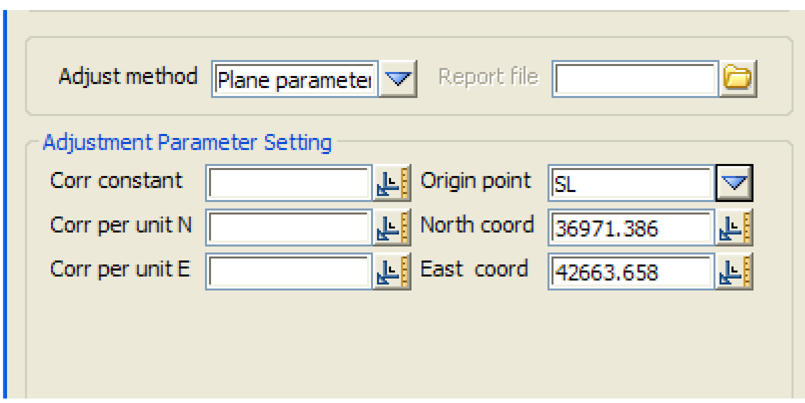
Constant adjustment method



Corr constant input box

the value to be added to the selected data.This method is the same as a translation in the z values only.

Plane parameters adjustment method



Corr constant input box

the value to be added to the selected data.

Corr per unit N input box

the z value correction per unit nothing.

Corr per unit E input box

the z value correction per unit easting.

Origin point string select [same as]

the name of the string of the origin pt.

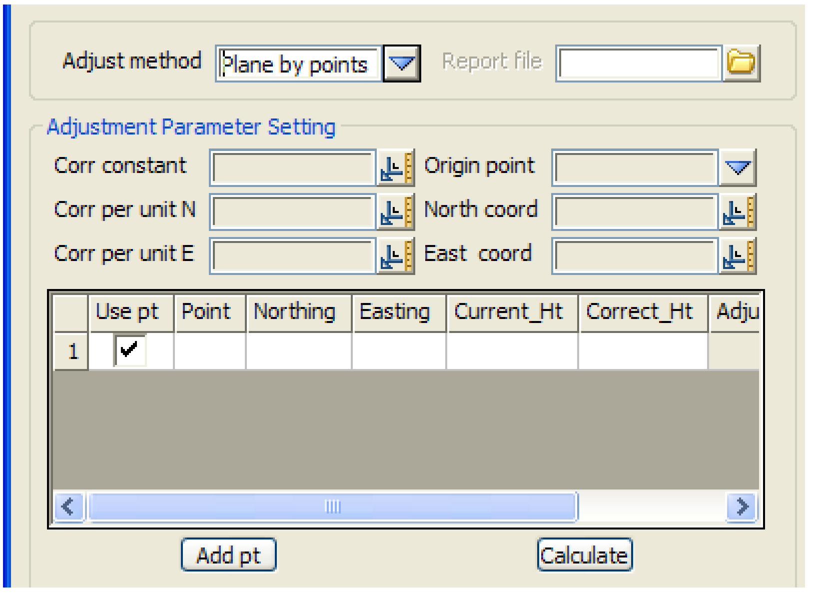
North coord input box

the northing value of the origin point.

East coord input box

the easting value of the origin point.

Plane by points adjustment method



the grid control values can be entered to build up the bench mark points from which the plane will be fitted to.

***Use pt**. the point (bench mark point) can be included in the derivation of the parameters by selecting the **Use pt** option. If the point is not to be included, then uncheck the box.*

***Point**. The point id of the bench mark*

***Northing**. The northing value of the bench mark*

***Easting**. The Easting value of the bench mark*

***Current_Ht**. The current level value of the bench mark*

***Correct_Ht**. The correct height in the adjusted level plane.*

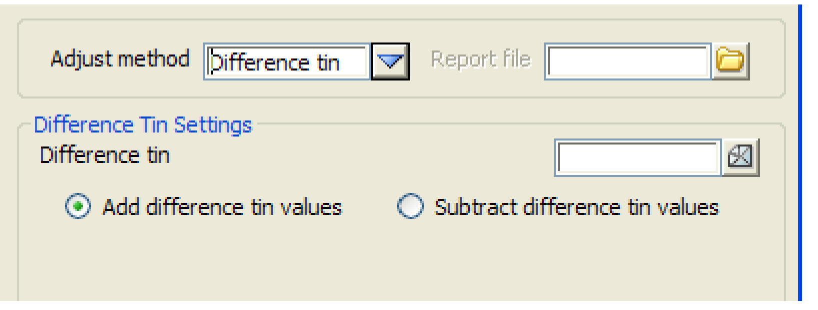
***Adjusted_Ht**. This field shows the adjusted height after the adjustment is performed.*

***Residual**. This field shows the difference between the correct height and the adjusted height. This value is shown after the adjustment is performed.*

The **Add pt** button used in this panel can be used to add points into the grid control from selecting points from a current view. After the button is selected, consequent selection and acceptance of points from the view will add the value into the grid. The **Correct_Ht** parameter will then have to be filled out by the user.

The **Calculate** button can be used to calculate the adjustment parameters and residuals so that they can be inspected before the adjustment is made.

Difference tin adjustment method



This option allows a difference tin to be specified as the adjustment reference. The selected points to be adjusted will be projected onto the difference tin to establish an interpolated adjustment value. This value is then applied to the data according to the add or subtract options. The fields used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Difference tin	tin box		available tins

the difference tin to be used to interpolate adjustment values.

Add difference tin values radio button

if the button **Add difference tin values** *is checked the interpolated value from the difference tin will be added to the existing levels of the adjusted points.*

Subtract difference tin values radio button

if the button ***Subtract difference tin values*** *is checked the interpolated value from the difference tin will be subtracted from the existing levels of the adjusted points.*

if *Constant, a given value is subtracted from each z-value*

Corr constant

the value to be subtracted from all z-values

Origin points

the value to be added to all z-values

if *Plan parameters, a 3d-plane is defined and the z-value of the plane subtracted from any points*

Corr constant

z-correction at the original point

Corr per unit N

z-correction per unit is the North (y) direction abut the origin point

Corr per unit E

z-correction per unit is the East (x) direction

Origin point 2d select box

pick box for selected a pint to use for North and East coordinates about the origin point

North coordinate

East coordinate

if *Constant, a given value is subtracted from each z-value*

Corr constant

the value to be subtracted from all z-values

Origin points

the value to be added to all z-values

if *Plan by points a 3d-plane is fitted using least squares adjustment to selecting points. The z-value of the plane is then subtracted from any points.*

Corr constant

z-correction at the original point

Corr per unit N

z-correction per unit is the North (y) direction abut the origin point

Corr per unit E

z-correction per unit is the East (x) direction

Origin point 2d select box

pick box for selected a pint to use for North and East coordinates about the origin point

North coordinate

East coordinate

if *Difference tin, the z-value from a triangulation is added/subtracted from any points*

Difference tin tin box available tins

tin of values to be added/subtracted from the z-values of points

Add difference tin values radio button

*if **on**, the values of the tin are **added** to the selected points*

Subtract difference tin values radio button

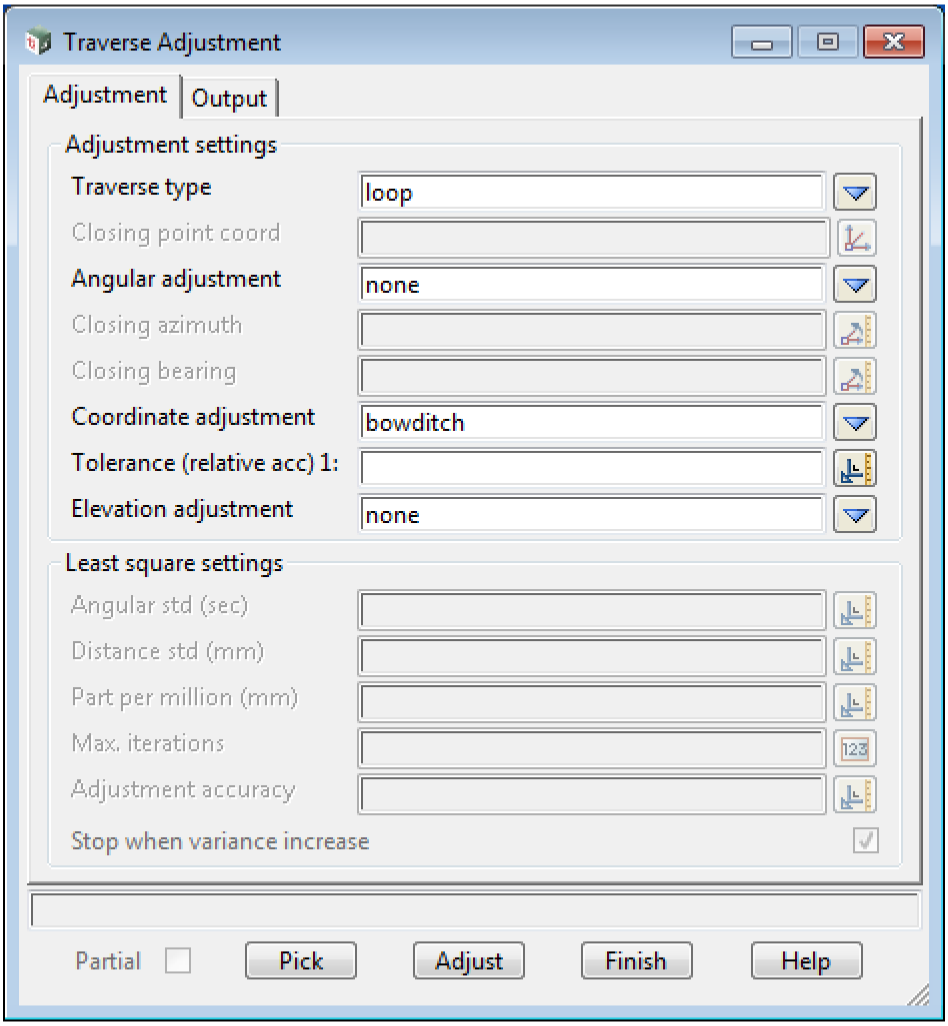
*if **on**, the values of the tin are **subtracted** to the selected points*

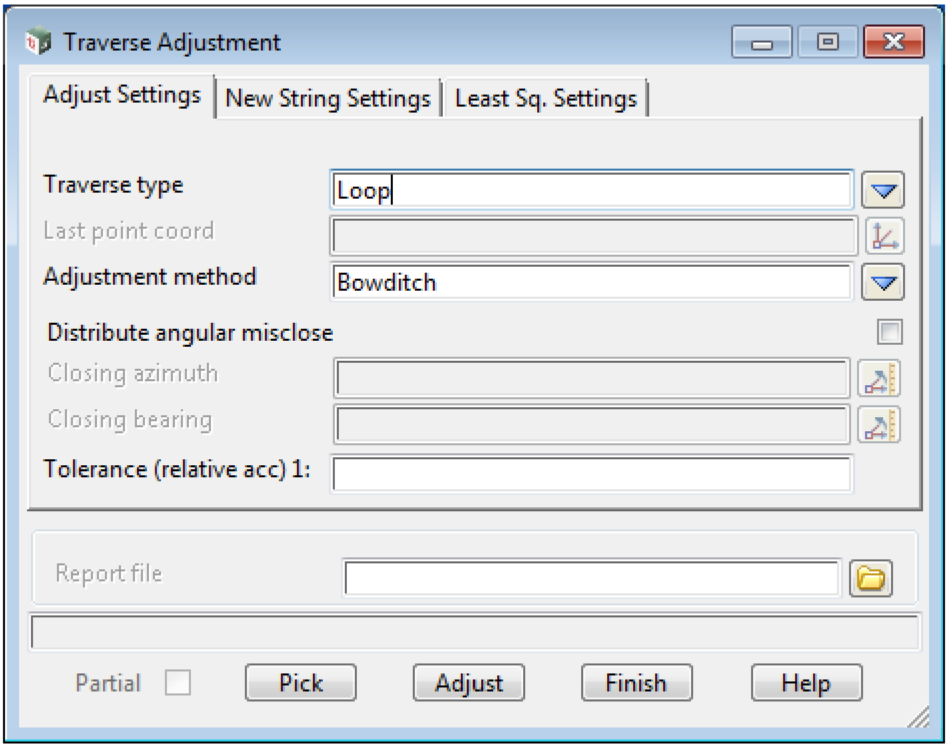
17.9.6 Traverse Adjustment

Position of option on menu: **Survey =>Adjustments =>Traverse adjustment**

The **Traverse adjustment** option performs a Bowditch, Transit, Compass or Least Square adjustment on a selected string. The adjustment can be for a closed string (loop) or an open string (non-loop).

Selecting **Traverse adjustment** brings up the **Traverse Adjustment** panel.


Traverse Adjustment
Adjustment
Output
Adjustment settings
Traverse type
loop
Closing point coord
Angular adjustment
none
Closing azimuth
Closing bearing
Coordinate adjustment
bowditch
Tolerance (relative acc) 1:
Elevation adjustment
none
Least square settings
Angular std (sec)
Distance std (mm)
Part per million (mm)
Max. iterations
Adjustment accuracy
Stop when variance increase
Partial
Pick
Adjust
Finish
Help



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Adjust Settings tab

Traverse type	choice box	loop	loop, non-loop

the traverse can be a closed loop or a non loop.

Last point coord	xyz		

co-ordinates of the last point of the traverse. Only applicable for a non-loop traverse.

Adjustment method	choice box	transit	Transit, Bowditch, Compass, Least Square

method of adjustment.

Distribute angular misclose	tick box	unticked	

if ticked, the closing azimuth and bearing fields are enabled, allowing the entry of the values so that a calculation of an angular misclose can be made. i.e. angular misclose = closing azimuth - closing bearing.

This requires a closing observation to be recorded separately to the adjustment string chosen.

This angular misclose can be distributed around the traverse before adjustment. The distribution follows standard survey practice, calculating a misclose value per setup station and then adding this constant to every traverse leg in a accumulative manner.

*i.e. The 1st leg angle adjusted bearing = orig bearing 1st leg + 1*misclose/setup stn.*

*The 2nd leg angle adjusted bearing = orig bearing 1st leg + 2*misclose/setup stn. and so forth.*

The closing observation is not taken from the traverse string to be adjusted, but by direct entry into the closing azimuth and bearing fields.

Closing azimuth	angle box		

The closing azimuth is the bearing as calculated from the last point (known coordinate; equal to the *start point if a loop traverse) to the closing station (known coordinate or bearing).*

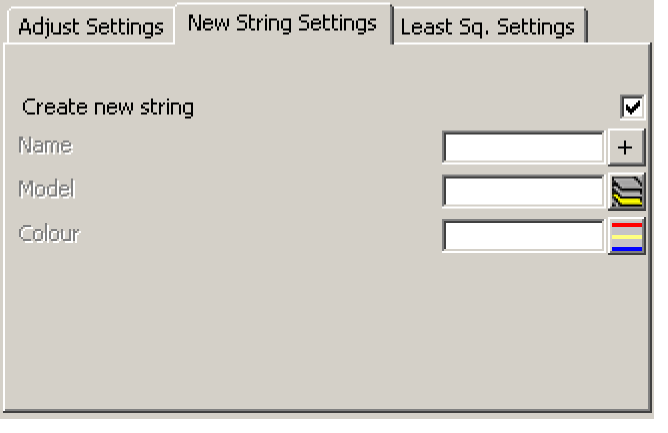
Closing bearing angle box

The closing bearing is the observation from the last traverse point to the closing station (known coordinate or bearing)

Tolerance (relative acc) 1:

The accuracy can be set which will be used to assess the misclose value before the adjustment is made. If the misclose is out of tolerance, a message box will appear notifying the user that the linear tolerance has not been met and prompting for further action. This may force the adjustment to continue or the exit of the process.

New String Settings tab



Create new string tick box tick

if ticked, a new adjusted string is created. If not tick, the selected string is adjusted.

Name name box

if non-blank, the name of the new string. If blank, the original string name is used.

Model model box available models

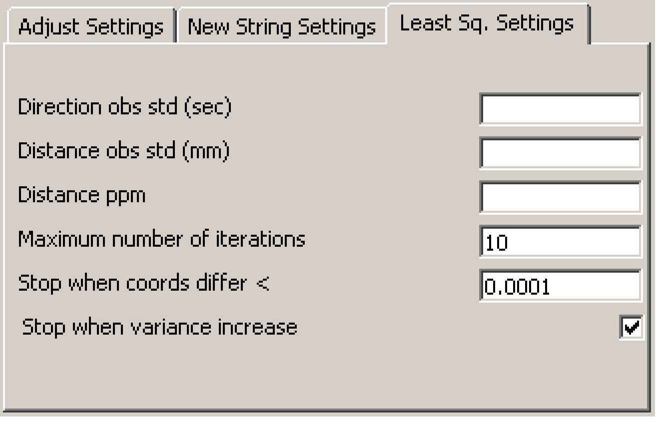
if non-blank, the model of the new string. If blank, the original string model is used.

Colour colour box available colours

if non-blank, the colour of the new string. If blank, the original string colour is used.

if the least square method is chosen the following parameters can be set in the **Least Sq. Settings** *tab. If a different method is chosen the least square parameters are shown* ***greyed out*** *indicating that they are not used in the adjustment.*

Least Sq. Settings tab



Direction obs std (sec)

standard deviation of a single angular measurement in seconds of arc.

Distance obs std (mm)

standard deviation of a single distance measurement.

Distance ppm

distance error in parts per million.

This value is defined by the edm scale error which is dependant on the length of line measured (e.g. 5ppm = 5mm error over a 1km distance)

Iterations input box 10

the calculation of the adjustment can be aborted if a solution is not found after the specified number of iterations.

Coords differ input box 0.0001

the calculation of the adjustment can stopped when the difference between successive calculations meet the defined tolerance.

Variance increase tick box ticked

the calculation of the adjustment can stopped when the variance between successive calculations increases

The remaining values are as follows:

Report file *.rpt files

if non-blank, a report for the adjustment is created with this name. If non-blank, no report is created.

Partial tick box

if ticked, only part of the selected string is adjusted.

Pick button

select the string to adjust.

Adjust button

perform the adjustment.

17.9.7 Survey Adjustments Utilities

Position of option on menu: **Survey =>Adjustments =>Utilities**

The options on the **Survey Adjustment Utilities** menu are designed to get data from external sources into the Least Squares Network and Level Network formats, (.IN .LIN files respectively) for processing by 12d.

The data conversion is a multistep process as it is designed to bring in several different formats into the 12d .IN/.LIN formats.

.SOA files

The SOA files are an intermediate text file generated onboard a Leica 1200/Viva TPS/GPS instrument.

The format files to generate these files from a Leica DBX are shipped in the library directory.

- **ANG_Code_Office.FRT** is the format file used when the Station number is put in as a code
- **ANG_ID_Office.FRT** is the format file used when the Station number is put in as a Point ID

The use of format files is out of the scope of this manual and described in the Leica documentation.

The SOA file format is intermediary and not documented.

.ANG files

ANG files are a text file containing rounds of angles, they are generated by the **Leica DBX to ANG** panel or can be generated in TP-Setout or TP-Stakeout.

The format of an **.ANG** is such

```
<Backsight ID> <Backsight Height> <Station ID> <Station Height>
<Foresight ID> <Foresight Height> <Backsight bearing> <Backsight Ver-
tical angle> <Backsight slope distance> <Foresight bearing> <Fore-
sight Vertical angle> <Foresight slope distance>
```

Bearing and angles are in radians and station Ids must be quoted if they have spaces.

```
"34748A" 1.300 "34872A" 1.660 "34978B" 0.000 5.49580322 1.58360594
159.477 2.83371620 1.53209318 117.170
```

.LSF files

LSF files are a TP-Setout/ TP-Stakeout format for a least squares network adjustment.

The format consists of several sections, each section is terminated by an **END**.

The first a list of station coordinates and weighting definitions.

```
<Station name> <Easting> <Northing> <Weight> <Level> <Weight>
GB117 9808.1480 168635.6450 1000 4.0314 1000
```

A weight of 1000 means fixed, 1 completely free.

The second section is the horizontal observations.

```
<Backsight ID> <Station ID> <Foresight ID> <Backsight bearing> <Fore-
sight bearing>
KHABB KHP14 KHP13 0.0000 3.15157367
```

Bearing and angles are in radians and station Ids must be quoted if they have spaces.

The 3rd section is the horizontal distances.

```
<Station name> < Station name > < Horizontal distance>
KHABB GB117 68.909
```

The 4th section is the height difference information.

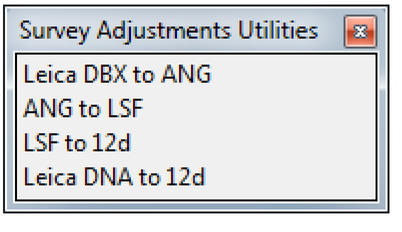
```
<Station name> < Station name > < Vertical distance>
KHABB GB117 -11.138
```

A complete sample:

```
GB117 9808.1480 168635.6450 1000 4.0314 1000
KHP17 9839.3686 168651.5200 1 18.2912 1000
KHABB 9813.0536 168704.3997 1 15.1675 1
END
GB117 KHABB KHP17 0.0000 5.74994583
KHABB KHP17 KHP16 0.0000 3.13833713
END
KHABB KHP17 59.053
KHABB GB117 68.909
KHP17 KHP16 71.045
END
KHABB KHP17 3.119
KHABB GB117 -11.138
KHP17 KHP16 3.757
END
```

The **Utilities** walk-right menu is

.

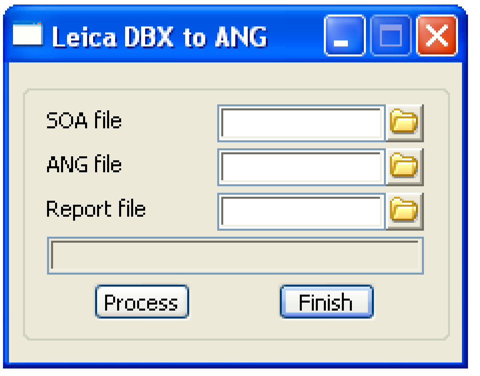


For the option Leica DBX to ANG go to 17.9.7.1 Leica DBX to ANG
ANG to LSF 17.9.7.2 ANG to LSF
LSF to 12d 17.9.7.3 LSF to 12d
Leica DNA to 12d 17.9.7.4 Leica DNA to 12d

17.9.7.1 Leica DBX to ANG

Position of option on menu: **Survey =>Adjustments =>Utilities => Leica DBX to ANG**
This option converts the **.SOA** file produced on the Leica instrument into a rounds of angles **.ANG** file.

Selecting **Leica DBX to ANG brings up the Leica DBX to ANG** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
SOA file	file box		

select the SOA file produced from a Leica DBX to process.

ANG file file box

enter the ANG to be created, if existing you will be warned it will be overwritten.

Report file file box

a report file is generated to allow scrutiny of the conversion process to detect gross errors.

if it looks as below containing details of each collated measurement, this report should be checked before further processing of the created ANG fie.

Report for conversion of "6404-CONTROL.SOA" to "QQ.ANG"

Overall stations 50
Overall measures 1816

17 meas's from SSM139008 to target PM42949
Hz: mean 49°54'55.7" std dev 0°00'03.0"
Va: mean 87°34'50.4" std dev 0°00'05.0"
Sd: mean 385.938 std dev 0.0003
TH: mean 1.784 std dev 0.0000

Process button

process the SOA file and produce the ANG file.

17.9.7.2 ANG to LSF

Position of option on menu: **Survey =>Adjustments =>Utilities => ANG to LSF**
The panel converts a round of angles **ANG file** into the **LSF** least squares network format.

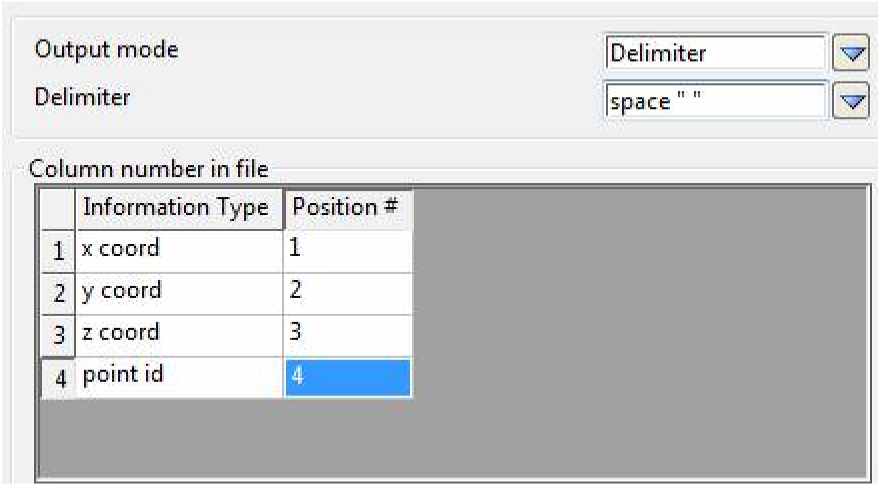
As a prerequisite to using this panel a **PTA file** of the fixed control points must be created.

The PTA format is a space delimited **<Easting> <Northing> <Level> <"ID">**

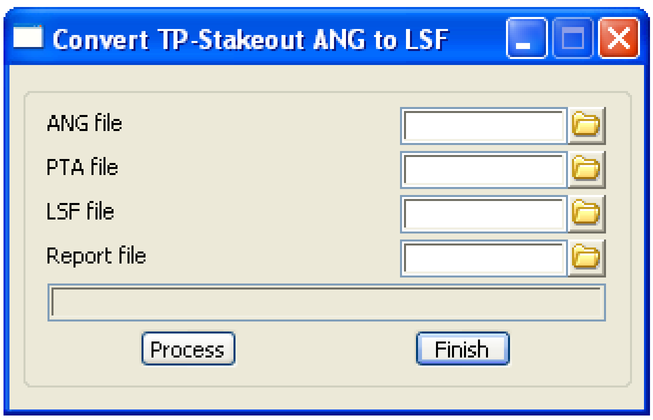
```
12343.345 234435.236 500.127 "STN1"
12654.877 234435.345 501.457 "STN2"
12456.345 234543.866 499.345 "STN3"
```

The **PTA file** can come in from **TP-Setout/TP-Stakeout** or it can be written out using the **File I/O =>Data output->x y z general** option. See 8.2.5.4 User X Y Z and Attributes Output .

Set the following fields in the xyz output for creating a **PTA** format file.



Selecting **ANG to LSF brings up the Convert TP- Stakeout ANG to LSF** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
ANG file	file box		

the ANG file to be converted.

PTA file	file box		

the PTA file containing the fixed control points.

LSF file	file box		

the LSF file to be created.

Report file	file box		

a report file is generated to allow scrutiny of the conversion process to detect gross errors. It has simple statistics on the angle, distances and height differences.

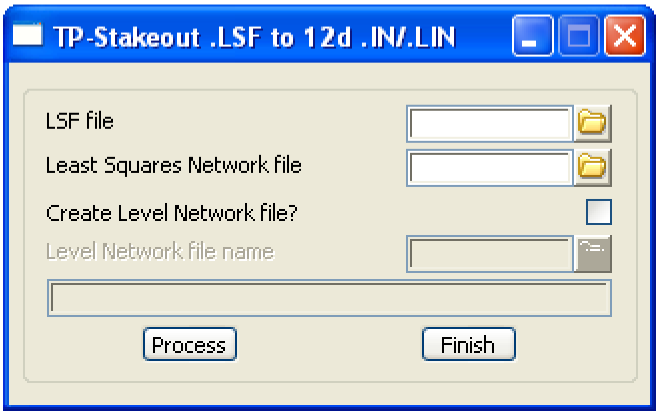
Process	button		

create a LSF file from the ANG file.

17.9.7.3 LSF to 12d

Position of option on menu: **Survey =>Adjustments =>Utilities => LSF to 12d**
This panel converts a **LSF** format file into the **12d** Least squares network and Level network file formats.

Selecting **LSF to 12d brings up the TP- Stakeout .LSF to 12d .IN/.LIN** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

LSF file file box

the LSF file to be processed.

12d Least squares network file file box

the 12d .IN file to be created.

Create Level Network file? tick box

if ticked enter the Level Network .LIN file will be created.

Level Network file name

the Level Network .LIN file to be created.

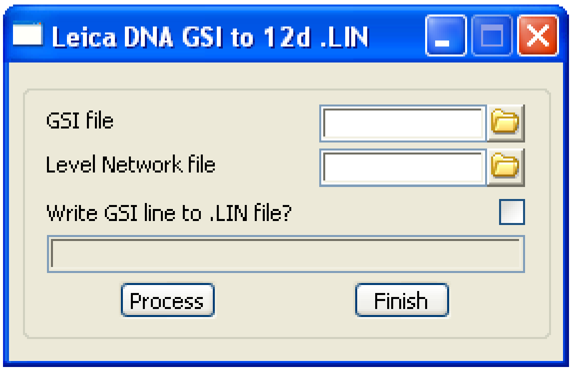
Process button

create the .IN and optional .LIN file form the LSF file.

17.9.7.4 Leica DNA to 12d

Position of option on menu: **Survey =>Adjustments =>Utilities => Leica DNA to 12d**
This panel converts a **GSI** format file from a Leica digital level into the 12d Level network file format.

Selecting **Leica DNA to 12d brings up the Leica DNA GSI to 12d .LIN** panel.



The fields and buttons used in this panel have the following functions.

Field Description | Type | Defaults | Pop-Up

GSI file file box

the GSI file to process

Level Network file file box

the .LIN file to create

Write GSI line to .LIN file? tick box

if ticked the raw data from the GSI file will be written as a comment line into the .LIN file.

Process button

create the level network LIN file.

17.10 Conversions

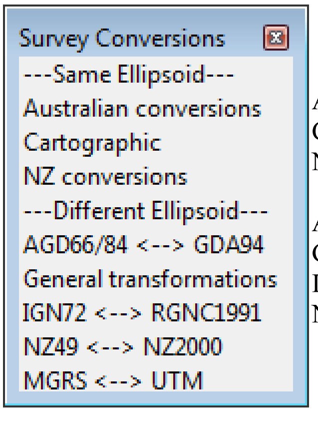
Position of menu: **Survey =>Conversions**

The **Conversions** walk-right menu contains survey transformations for converting between most projections (or longitude, latitude) including Transverse Mercator, UTM etc.conversion options.

Conversions can involve the same ellipsoid or different ellipsoid. When using different ellipsoids, either a 7-parameter transformation or NTVT2 grids can be used to convert between the ellipsoids.

The **Conversion** walk-right menu is

.



Australian conversions (eg zone to zone) fro the same datum
Cartographic conversion from one projection to another (same datum)
NZ conversions (eg circuit to circuit) for the same datum

AGD 66/84 <--> GDA94 conversions (AMG <-> MGA
General conversions with ellipsoid change
IGN72 <-->RGN1991 conversions
NZ49 <--> NZ2000 conversions

For converting between projections using the same ellipsoid:

For the option *Australian conversions*, go to 17.10.1 Australian Conversions .

Cartographic, 17.10.2 Cartographic

NZ conversions, 17.10.3 NZ Conversions

For converting between projections with different ellipsoids:

For the option *AGD66/84<-->GDA94*, go to 17.10.4 AGD66/84 <---> GDA94 .

General transformations 17.10.6 General Transformations

IGN72<->RGNC1991, 17.10.7 IGN72 <---> RGNC1991

NZ49<->NZ2000, 17.10.5 NZ49 <---> NZ2000

MGRS<->UTM, 17.10.8 MGRS <---> UTM

For more information about terminology used in these sections, see the Appendix 38 Geodetics Summary .

17.10.1 Australian Conversions

Position of option on menu: **Survey =>Conversions =>Australian conversions**

The **Australian conversions** option converts data from one form (AMG/MGA, Long Lat, Global XYZ) to another for the same common datum (AGD66/84 or GDA94). The option also allows for the conversion from one zone to another for a common datum. (e.g. from zone 54 to 55 in MGA).

For converting ISG co-ordinates in the same datum, use the option

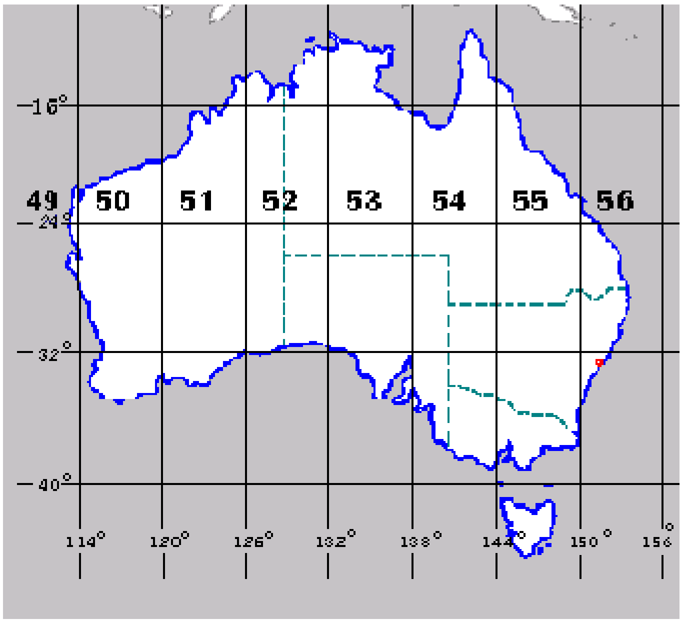
Survey =>Conversions =>Cartographic (see 17.10.2 Cartographic).

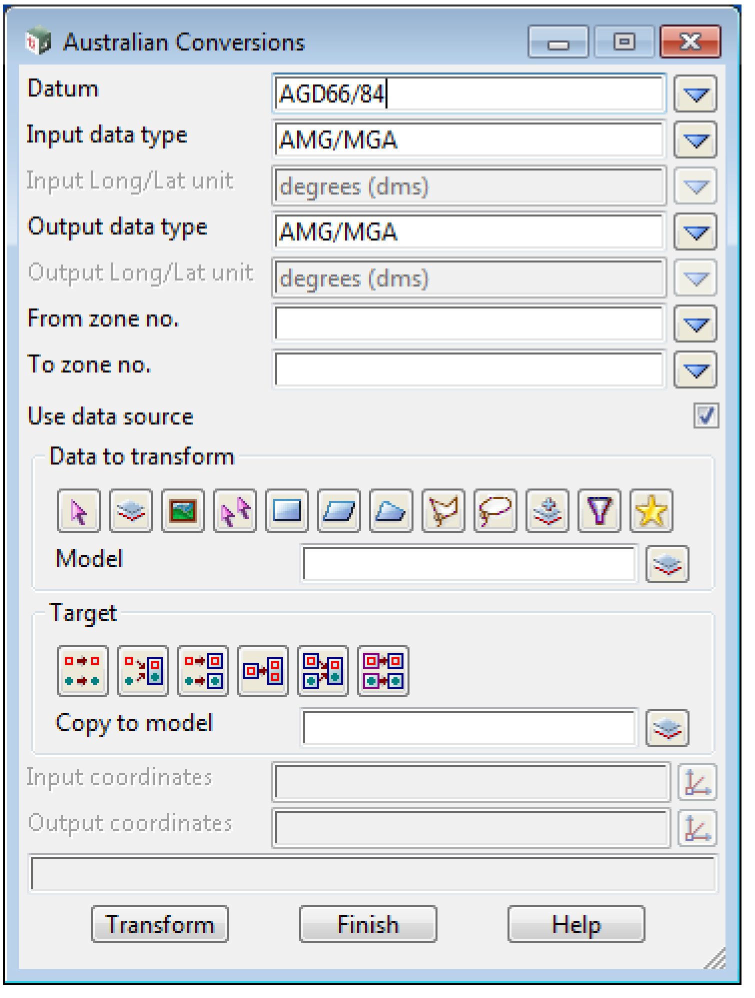
For converting between the ***different*** datums AGD66/84 and GDA94, use the option

Survey =>Conversions =>AGD66/84 <-> GDA94 (see 17.10.4 AGD66/84 <---> GDA94).

For more information, go to the section 38.5 Converting Between AMG, ISG and MGA in the Appendix 38 Geodetics Summary .

Selecting **Australian conversions** brings up the **Australian conversions** panel





The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Datum	choice box	AGD66/84	AGD66/84, GDA94
either AGD66/84 or MGA94 is selected.			
Input data type	choice box	AMG/MGA	AMG/MGA, Long Lat Global XYZ
type of the input data type.			
Input Lat/Long unit	*choice box*	degrees	radians, degrees, decimal degrees
if **Lat/Long** *was selected, the Lat-Long coords have the selected units.*			
Output data type	choice box	AMG/MGA	AMG/MGA, Long Lat Global XYZ
type of the input data type.			
Output Lat/Long unit	*choice box*	degrees	radians, degrees, decimal degrees
if **Lat/Long** *was selected, the Lat-Long coords have the selected units.*			
From zone no.			49 -> 59
if **AMG/MGA** *was selected for the input data type, the AMG/MGA zone is given is this field.*			
To zone no.			49 -> 59

if **AMG/MGA** *was selected for the output data type, the AMG/MGA zone is given is this field.*

Use data source tick box tick

if ticked, a data source is used to select the data to be transformed. If not ticked, a given co-ordinate is transformed.

Data source input

if ***use data source*** *is ticked, the source of data to transform. The source of the data to be adjusted is selected using the data source box. For more information on the data source box see* 4.19.3 Data Source

Target target

if ***use data source*** *in the data source is ticked, the target of the transformed data should be specified. The target for the data is selected using the data target box. For more information on the data target box see* 4.19.4 Data Target

Input coordinates XYZ box

if ***use data source*** *is not ticked, the co-ordinates in this field is transformed.*

Output coordinates XYZ box

if ***use data source*** *is not ticked, the transformed co-ordinates are displayed in this field.*

Transform button

perform the transformation.

17.10.2 Cartographic

Position of option on menu: **Survey =>Conversions =>Cartographic**

Position of option on menu: **Utilities =>A-G =>Cartographic**

The **Cartographic** option is used to transform data based on the same datum between

(a) two different cartographic projections (based on the same datum)

(b) longitude and latitude and a cartographic projection (based on the same datum)

(c) a cartographic projection and longitude and latitude (based on the same datum).

The datum, relates to the ellipsoid model used. This should be the same for both the to and from fields. i.e. a transform between data based on the ANS ellipsoid, to data based on the WGS84 ellipsoid would be incorrect.

(a) Transforming Between Two Cartographic Projections

For the first case, the data starts in one cartographic projection and is to be transformed into another cartographic projection.

This is achieved by converting the data in (x,y) co-ordinates in the original projection, to (longitude, latitude) and then converting from (longitude, latitude) to (x,y) in the new projection.

original projection → (longitude, latitude) → new projection

For example, to transform from MGA zone 50 to MGA zone 51, the original data starts in MGA zone 50, is transformed to (longitude, latitude) and then transformed from (longitude, latitude) to MGA zone 51.

Note that to make the transformation, the two cartographic projection must already be defined. That is in the example above, the two projections MGA zone 50 and MGA zone 51 must already be defined.

In 12d Model, a *Cartographic projection* is defined by the option

Project => Projections =>Create/edit

and is documented in the section 7.6.6.2 Create/Edit Projection .

The two cartographic projection definitions for the conversion are given in the **Cartographic** panel.

(b) Transform Between Longitude-Latitude and a Cartographic Projection

For the second case, the data starts in (longitude, latitude) is to be transformed into (x,y) co-ordinates in a cartographic projection.

(longitude, latitude) $\longrightarrow$ cartographic projection

For example, to transform (longitude, latitude) to MGA zone 51.

To make the transformation, the required cartographic projection must already be defined by the option

Project => Projections =>Create/edit

which is documented in the section 7.6.6.2 Create/Edit Projection .

Notes

1. In the southern hemisphere, the latitude values are **negative**. If the latitude is given as positive rather than negative, then the option **Factor** can be used to multiply the latitude by -1.
2. If the information is given in (latitude, longitude) rather than the required (longitude, latitude), then the option **Swap XY** can be used to swap the order of the co-ordinates.

(c) Transform Between a Cartographic Projection and Longitude-Latitude

For the third case, the data starts in (x,y) co-ordinates in a cartographic projection and need to be transformed to (longitude, latitude).

cartographic projection $\longrightarrow$ (longitude, latitude)

For example, to transform from MGA zone 51 to (longitude, latitude).

To make the transformation, the required cartographic projection must already be defined by the option

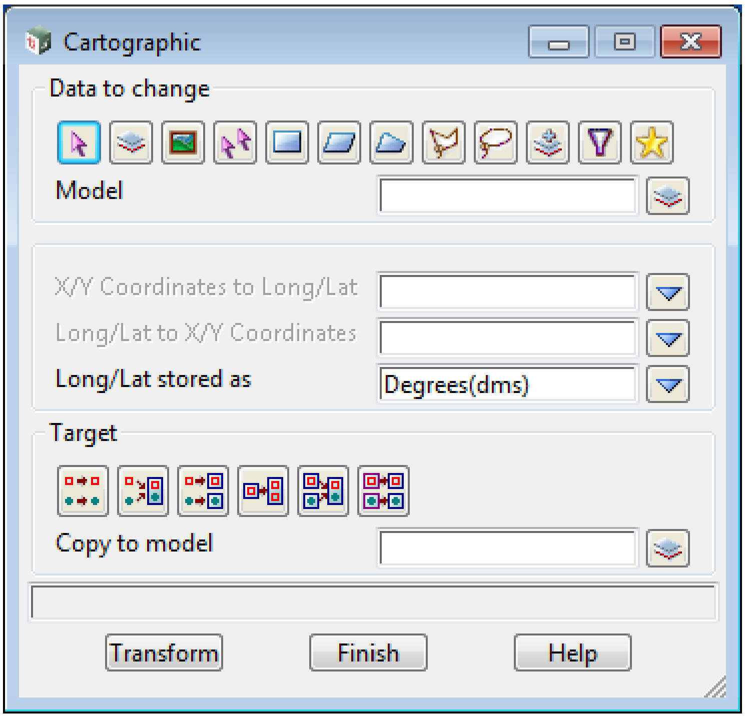
Project => Projections =>Create/edit

which is documented in the section 7.6.6.2 Create/Edit Projection .

Notes

1. In the southern hemisphere, the latitude values are **negative**. If a positive value for latitude is required, then the option **Factor** can be used to multiply the latitude by -1.
2. If the information is required to be given in (latitude, longitude) rather than the (longitude, latitude) produced by the transformation, then the option **Swap XY** can be used to swap the order of the co-ordinates.

On selecting the **Cartographic** option, the **Cartographic** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type			

data selection type - for a full description go to 4.19.3 Data Source .

Data source	input		

source of data to be processed.

X/Y Coordinates to Long/Lat		choice box	available projections

if ***non-blank****, the name of the cartographic projection to map the selected string co-ordinates into longitude-latitude values.*
If ***blank****, the initial co-ordinates are not transformed from (x,y) to (longitude, latitude). Hence the initial co-ordinates are then assumed to be in (longitude, latitude) form ready to be transformed to (x,y) by the transformation given in the* **Long/Lat to x/y coordinates** *field. Note that in the southern hemisphere, the latitude values must be given as* ***negative****.*

Long/Lat to X/Y Coordinates		choice box	available projections

if ***non-blank****, the cartographic projection to apply to the longitude-latitude values.*
If ***blank****, the co-ordinates are not transformed from (longitude, latitude) to (x,y). Hence the initial (x,y) co-ordinates are transformed to (longitude, latitude) by the transformation given in the* **x/y co-ordinates to Long/Lat** *field and then left in (longitude, latitude). Note that in the southern hemisphere, the latitude values are* ***negative****.*

Long/Lat stored as	input	degrees	radians degrees decimal degrees

format for the longitude and latitudes - either radians, degrees (in 4.17.1 HP Notation *for degrees, minutes and seconds) or decimal degrees.*

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info	input		

extra information required for the target.

Transform button

transform the selected strings to longitude-latitude co-ordinates using the **x/y coordinates to long/ lat (radians)** *transformation, and then project those longitude-latitude co-ordinates into the new co-ordinate system using the* **long/lat (radians) to x/y coordinates** *transformation.*

Note

The **Cartographic** option can be used to transform just *x/y coordinates* to *long/lat* or *long/lat* to *x/ y coordinates* by simply leaving the appropriate panel field blank. The units for *longitude* and *latitude* are either degrees (dms), decimal degrees or radians.

17.10.3 NZ Conversions

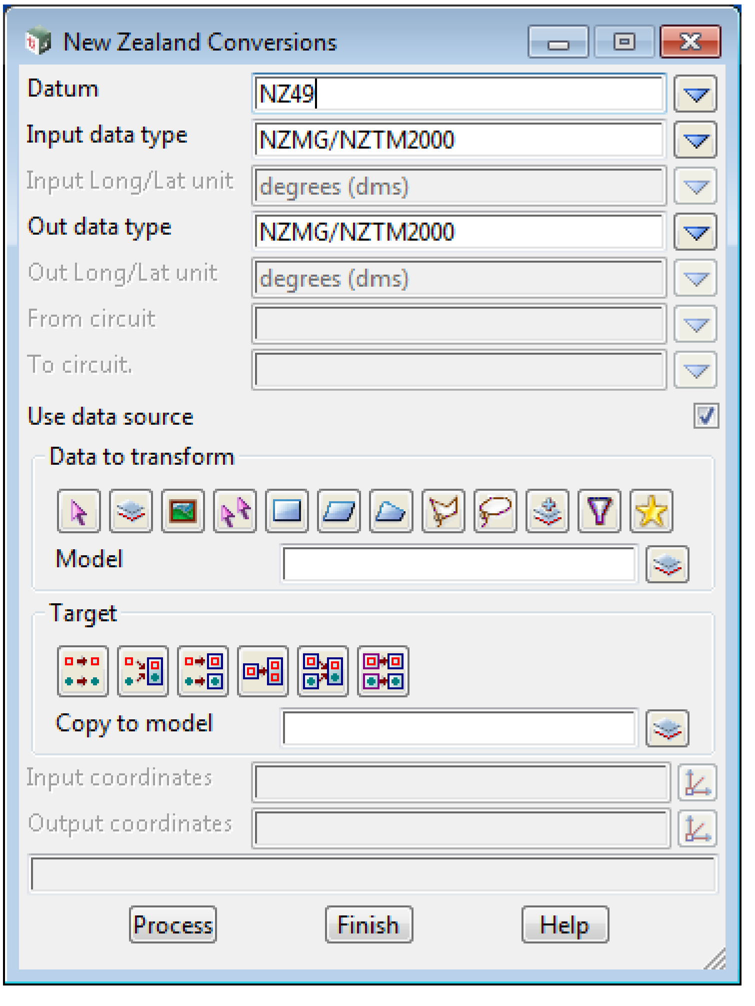
Position of option on menu: **Survey =>Conversions =>NZ conversions**

The **NZ conversions** option converts data from one form (NZMG/NZTM2000, Long Lat) to another for the same ellipsoid (i.e. datum NZ49 or NZ2000).

For converting between the ***different*** datums NZ 49 and NZ 2000, use the option

Survey =>Conversions =>NZ49 <-> NZ2000 (see 17.10.5 NZ49 <---> NZ2000).

Selecting **NZ conversions** brings up the **New Zealand Conversions** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Datum	choice box	NZ49	NZ49, NZ2000

either NZ49 or NZ2000 is selected.

Input data type	choice box	NZMG/NZTM2000NZMG. Circuit, Long Lat	

type of the input data type.

Input Lat/Long unit	*choice box*	degrees	radians, degrees, decimal degrees

if **Lat/Long** *was selected, the Lat-Long coords have the selected units.*

Output data type	choice box	NZMG/NZTM2000NZMG. Circuit, Long Lat	

type of the input data type.

Output Lat/Long unit	*choice box*	degrees	radians, degrees, decimal degrees

if **Lat/Long** *was selected, the Lat-Long coords have the selected units.*

From Circuit

if **Circuit** *was selected for the input data type, the NZ Circuit is given is this field.*

To Circuit

if **Circuit** *was selected for the output data type, the NZ Circuit is given is this field.*

Use data source	tick box	tick	

if ticked, a data source is used to select the data to be transformed. If not ticked, a given co-ordinate is transformed.

Data source type	Model		

if ***use data source*** *is ticked, the data selection type*

Data source	input		

if ***use data source*** *is ticked, the source of data to transform. The source of the data to be adjusted is selected using the data source box. For more information on the data source box see* 4.19.3 Data Source

Target	target		

if ***use data source*** *in the data source is ticked, the target of the transformed data should be specified. The target for the data is selected using the data target box. For more information on the data target box see* 4.19.4 Data Target

Input coordinates	XYZ box		

if ***use data source*** *is not ticked, the co-ordinates in this field is transformed.*

Output coordinates	XYZ box		

if ***use data source*** *is not ticked, the transformed co-ordinates are displayed in this field.*

Transform	button		

perform the transformation.

17.10.4 AGD66/84 <---> GDA94

Position of option on menu: **Survey =>Conversions =>AGD66/84 <-> GDA94**

This option converts data between the two Australian Datums, AGD 66/84 and GDA 94. The transformation between the two datums can be by either a seven parameter similarity transformation for a State, or a NTv2 grid for either a State or Australia.

12d Model is GDA compliant for both the similarity transformations and NTv2 grids.

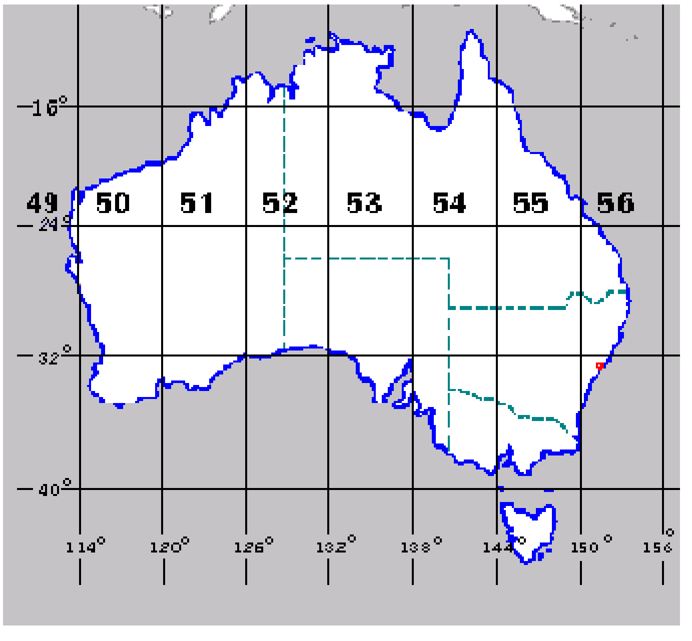
GDA

The AGD 66/84 data can be either longitude and latitude, co-ordinates in an AMG Zone or co-ordinates in an ISG Zone.

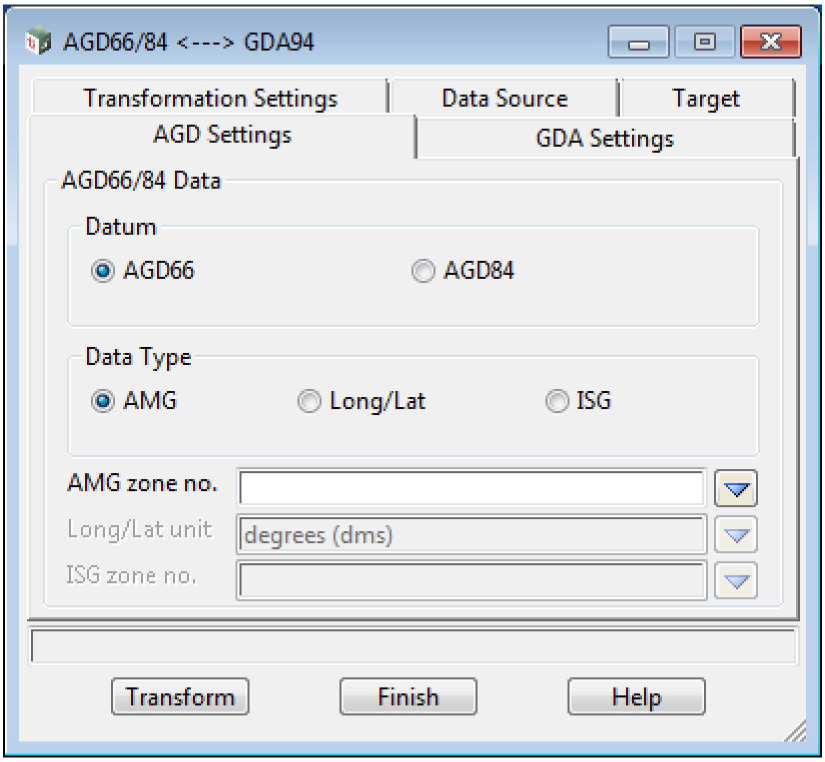
The GDA 94 data can be either longitude and latitude, Global XYZ or co-ordinates in a MGA Zone.

Hence the option will converts data from AMG 66/84 or ISG 66/84 to MGA 94 and vice-versa.

For more information, go to the section 38.5 Converting Between AMG, ISG and MGA in the Appendix 38 Geodetics Summary



Selecting agd66/84 <---> gda94 brings up the agd66/84 <---> gda94 panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

AGD Settings

AGD66 AGD84	radio button		

either AGD66 or AGD84 is selected.(AGD = Australian Geodetic datum)

AMG Lat/Long ISG	radio button		

either AMG, Lat/Long or ISG is selected.(AMG = Australian Map Grid, ISG = Integrated Survey Grid). The ISG relates to the state of New South Wales

AMG Zone no.	*choice box*		49 -> 59

if **AMG** *was selected, the AMG zone is given is this field.*

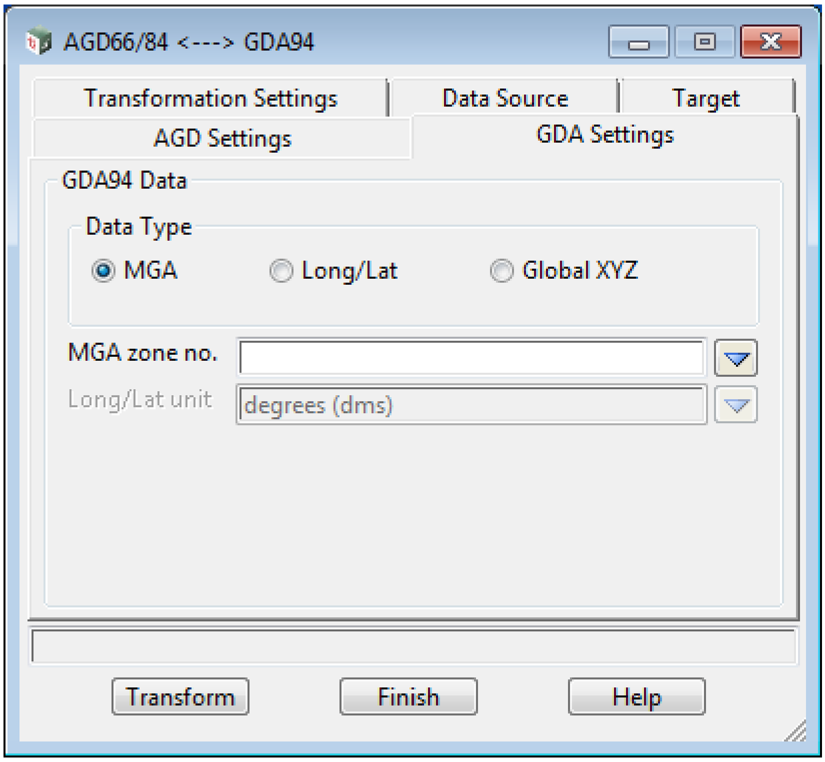
Lat/Long unit	*choice box*	degrees	radians, degrees, decimal degrees

if **Lat/Long** *was selected, the Lat-Long coords have the selected units.*

ISG Zone no.	*choice box*		ISG54/2 -> ISG56/3

if **ISG** *was selected, the ISG zone is given is this field.*

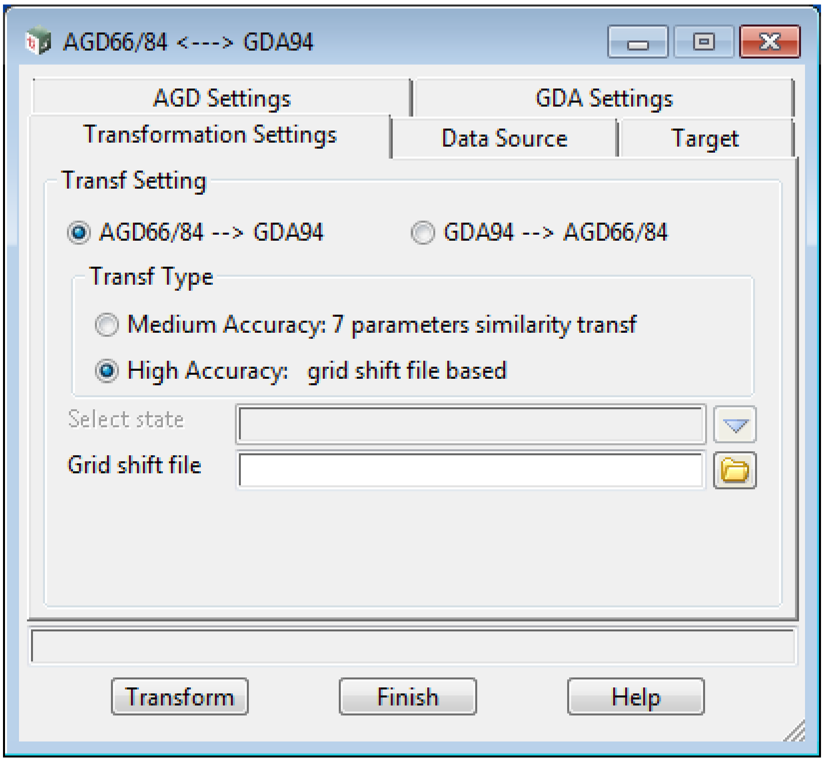
GDA Settings



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
MGA Lat/Long Global XYZ radio button			
one of the three choices is selected.			
MGA Zone no.	*choice box*		49 -> 59
if **MGA** *was selected, the MGA zone is given is this field.*			
Lat/Long unit	*choice box*	degrees	radians, degrees, decimal degrees
if **Lat/Long** *was selected, the Lat-Long coordinates have the selected units.*			

Transformation Settings



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

AGD66/84 --> GDA94 **GDA94 --> AGD66/84** radio button

one of the two choices is selected.

Medium accuracy radio box tick

if ticked, a seven parameter similarity transformation for the selected Australian State is used.

High accuracy radio button

if ticked, the NT V2 grid shift file is used.

Select State *choice box* ACT, NSW,TAS, VIC

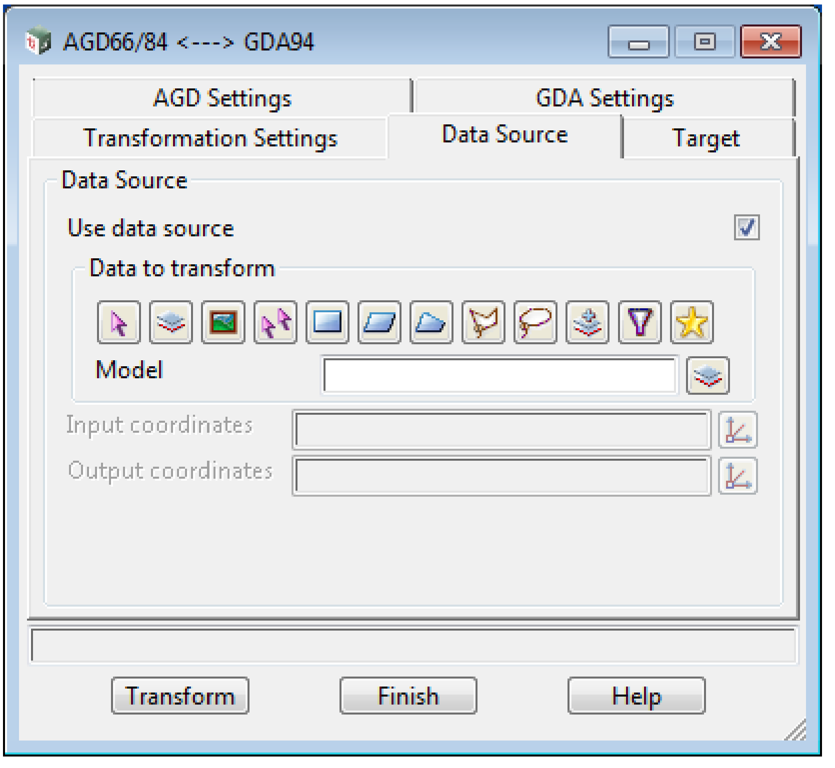
if the AGD66 datum has been selected, and the 7 parameter transformation is to be used, the seven parameter transformation parameters are different for each state. (i.e the states which adopted the AGD66 datum) By selecting the appropriate state, the corresponding values will be used.

If the AGD84 datum has been selected, the parameters used are the australian wide values and therefore the state selection is not appropriate.

Grid shift file *file box* **gsb*

for high accuracy, the NTv2 grid shift files can be used.

Data Source



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use data source	tick box	tick	

if ticked, a data source is used to select the data to be transformed. If not ticked, a given co-ordinate is transformed.

Data source input

*if **use data source** is ticked, the source of data to transform. The source of the data to be adjusted is selected using the data source box. For more information on the data source box see* 4.19.3 Data Source

Input coordinates XYZ box

*if **use data source** is not ticked, the co-ordinates in this field is transformed.*

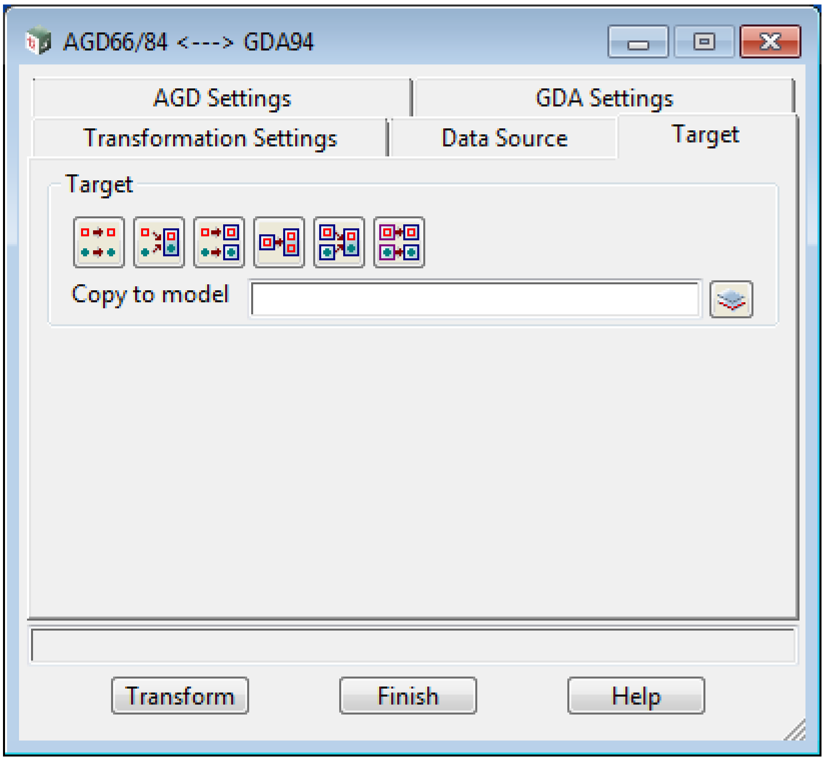
Output coordinates XYZ box

*if **use data source** is not ticked, the transformed co-ordinates are displayed in this field.*

Transform button

perform the transformation.

Target



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Target	target		

*if **use data source** in the data source is ticked, the target of the transformed data should be specified The target for the data is selected using the data target box. For more information on the data target box see* 4.19.4 Data Target

17.10.5 NZ49 <---> NZ2000

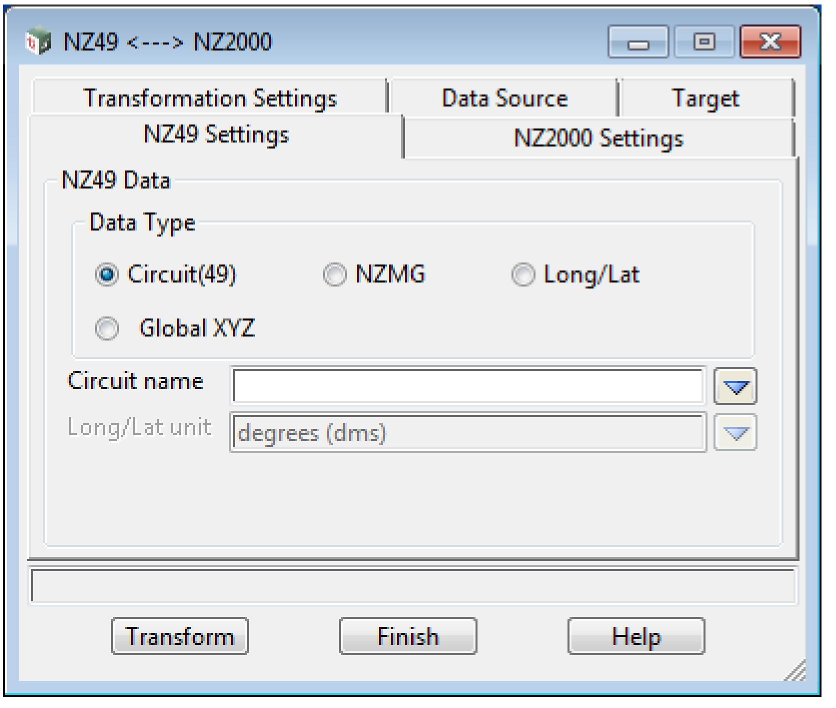
Position of option on menu: **Survey =>Conversions =>NZ49 <-> NZ2000**

This option converts data from the NZ 49 datum to the NZ 2000 datum and vice-versa.

The transformation between the two datums can be by either the NZ seven parameter similarity transformation or a NTv2 grid.

The data to be converted can be longitude and latitude, NZMG, circuits or Global XYZ.

Selecting **NZ49<---> NZ2000** brings up the **NZ49 <---> NZ2000** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

NZ 49 Settings

Circuit (49)	radio button		

if the data in the NZ49 datum is of ***circuit*** *type the circuit radio button is selected.*

NZMG	radio button		

if the data in the NZ49 datum is of NZMG type the radio button is selected.

Lat/Long	radio button		

if the data in the NZ49 datum is of Lat/Long type the radio button is selected.

Global XYZ	radio button		

if the data in the NZ49 datum is of Global XYZ type the radio button is selected.

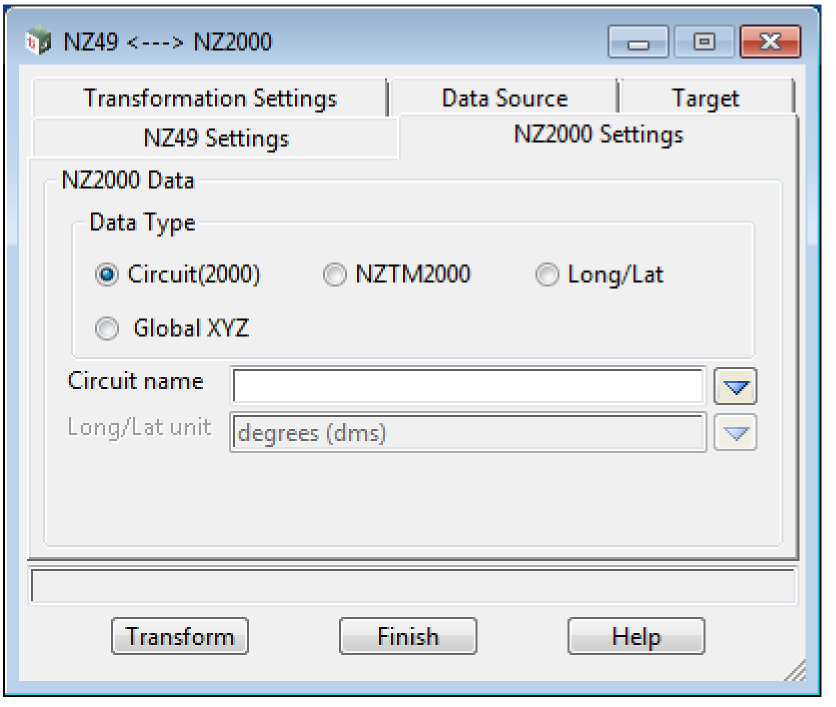
Circuit name	choice box		All NZ 1949 circuits

if **Circuit (49)** *was selected, the specific NZ Circuit should be selected.*

Lat/Long unit	*choice box*	degrees	radians, degrees, decimal degrees

if **Lat/Long** *was selected, the Lat-Long coords have the selected units.*

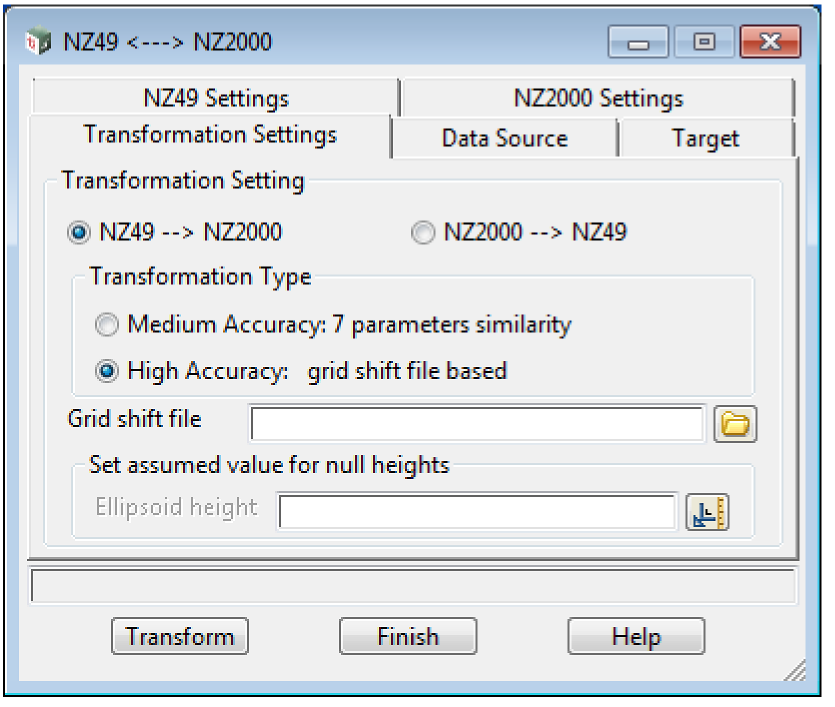
NZ 2000 Settings



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Circuit (2000)	radio button		
if the data in the NZ2000 datum is of ***circuit*** *type the circuit radio button is selected.*			
NZTM2000	radio button		
if the data in the NZ2000 datum is of NZTM2000 type the radio button is selected.			
Lat/Long	radio button		
if the data in the NZ2000 datum is of Lat/Long type the radio button is selected.			
Global XYZ	radio button		
if the data in the NZ2000 datum is of Global XYZ type the radio button is selected.			
Circuit name	choice box		All NZ 1949 circuits
if **Circuit (2000)** *was selected, the specific NZ Circuit should be selected.*			
Lat/Long unit	*choice box*	degrees	radians, degrees, decimal degrees
if **Lat/Long** *was selected, the Lat-Long coords have the selected units.*			

Transformation Settings



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

NZ49 --> NZ2000 NZ2000 --> NZ49 radio button

the direction of the transformation is specified by the selection of one of the two choices.

Medium accuracy tick box tick

if ticked, a seven parameter similarity transformation for NZ is used.

High accuracy tick box

if ticked, the NT V2 grid shift file is used.

Grid shift file *file box*

if **High accuracy** *is ticked, a NTv2 grid shift should be selected. A single file for New Zealand can be used.*

Ellipsoid Height *input box*

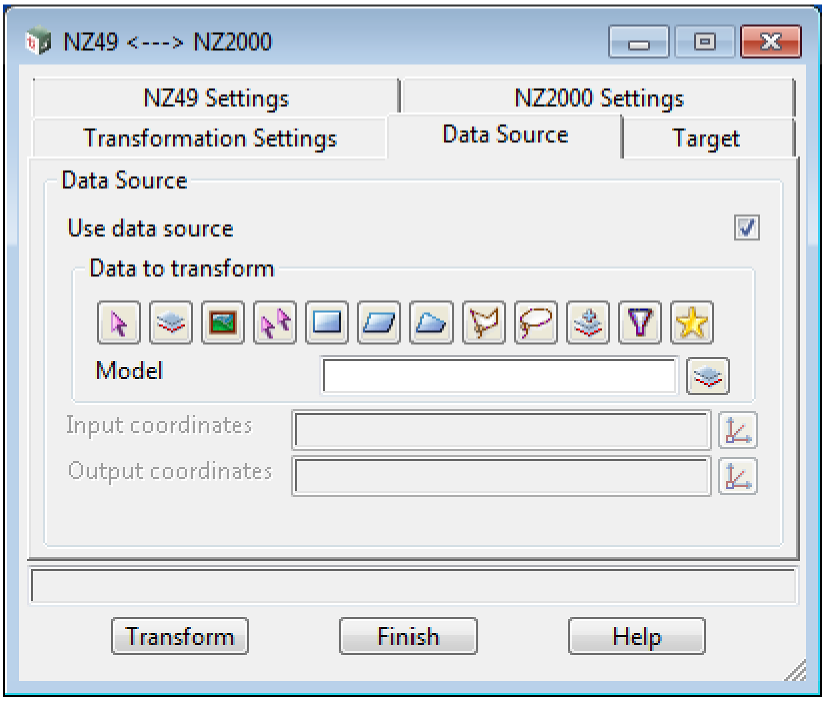
this is only required if the data to be transformed contains null values ***and*** *when*

1) The ***7 parameter similarity transformation*** *is used and/or*

2) When either the input or output types are of ***Global XYZ*** *type*

An approximate ellipsoid level needs only to be +/- 100 m from the true value for most accuracy requirements. (An error in estimation of the ellipsoid height of a few hundred metres will introduce an error in the transformed horizontal position in the order of millimetres only).

Data Source



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use data source	tick box	tick	

if ticked, a data source is used to select the data to be transformed. If not ticked, a given co-ordinate is transformed.

Data source	input		

if ***use data source*** *is ticked, the source of data to transform. The source of the data to be adjusted is selected using the data source box. For more information on the data source box see* 4.19.3 Data Source

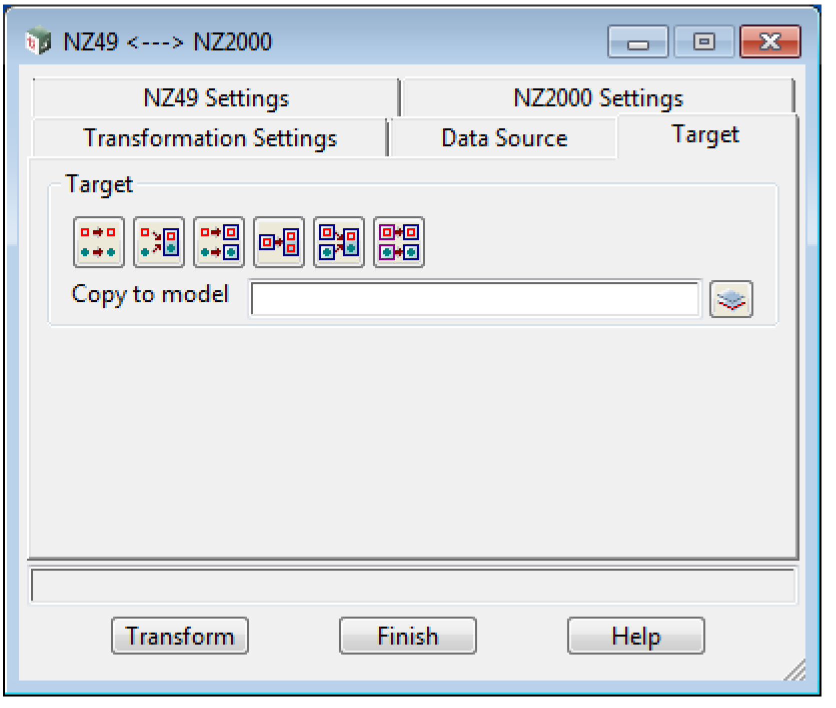
Input coordinates	XYZ box		

if ***use data source*** *is not ticked, the co-ordinates in this field is transformed.*

Output coordinates	XYZ box		

if ***use data source*** *is not ticked, the transformed co-ordinates are displayed in this field.*

Target



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Target	target		

*if **use data source** in the data source is ticked, the target of the transformed data should be specified. The target for the data is selected using the data target box. For more information on the data target box see* 4.19.4 Data Target

Transform	button		

perform the transformation.

17.10.6 General Transformations

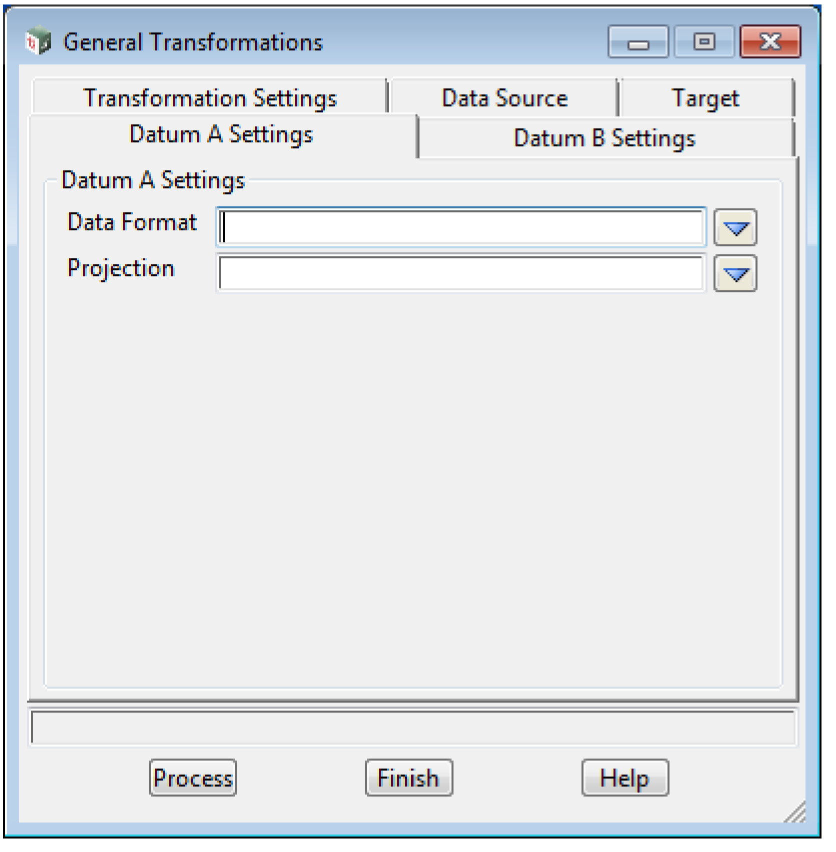
Position of option on menu: **Survey =>Conversions =>General transformations**

This option converts data between two datums (i.e. different ellipsoids).

The transformation between the two datums can be by either a seven parameter similarity transformation or a NTv2 grid.

The data to be converted can be longitude and latitude, a 12d supported projection (eg TM, UTM) or Global XYZ.

Selecting **General transformations** brings up the **General Transformations** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Datum A Settings, Datum B Settings

Data format	choice box		Global XYZ Easting Northing Long Lat - radians Long Lat - degrees (dms) Long Lat - decimal degrees

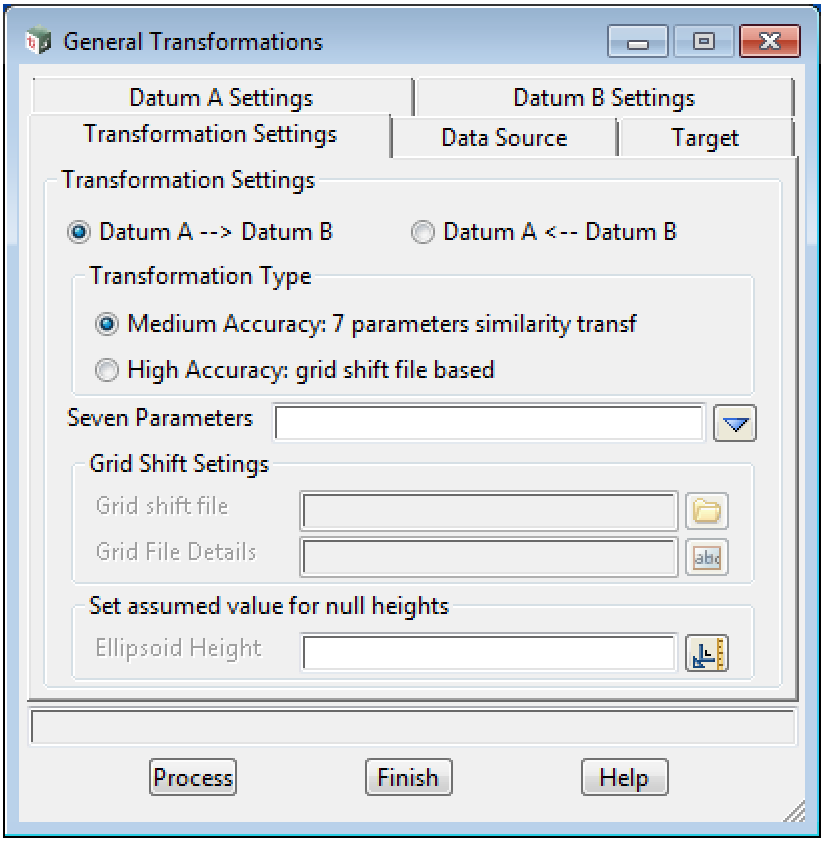
the type of data to be transformed/ transformed to.

*if **Easting, Northing**, the data is co-ordinates in the projection defined by the* **Projection** *field.*
*if **Long Lat - radians**, the data is (longitude, latitude) with the angles given in radians.*
*if **Long Lat - degrees (dms)**, the data is (long, lat) with the angles given in hp format (i.e. ddd.mmssss).*
*if **Long Lat - decimal degrees**, the data is (long, lat) with the angles given decimal degrees.*
*if **Global XYZ**, the data is in the Global XYZ system.*

Projection	projection box		defined projections

the name of the cartographic projection that the data is in.

Transformation Settings



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Datum A --> Datum B **Datum B --> Datum A** radio button

the direction of the transformation is specified by the selection of one of the two choices.

Medium accuracy tick box tick

if ticked, a seven parameter similarity transformation is used.

High accuracy tick box

if ticked, a NT V2 grid shift file is used.

Seven parameters *file box* *available 7 param definitions*

if ***Medium accuracy*** *is ticked, a seven parameter similarity transformation is selected.*

Grid shift file *file box* *available gsb files*

if ***High accuracy*** *is ticked, a NTv2 grid shift is selected.*

Ellipsoid Height *input box*

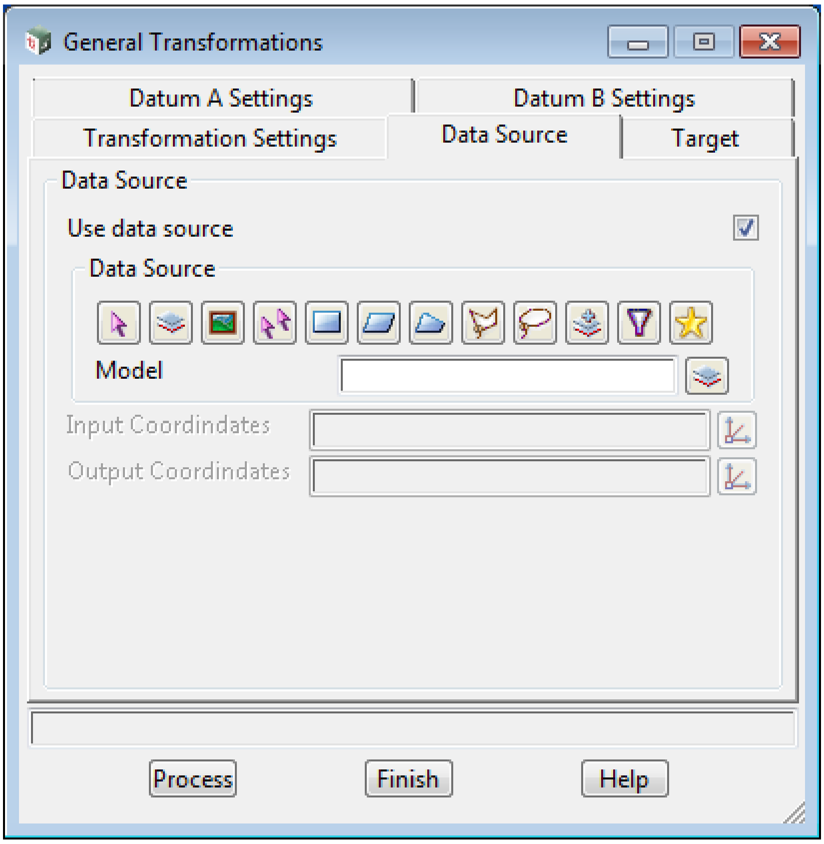
this is only required if the data to be transformed contains null values ***and*** *when*

1) The 7 ***parameter similarity transformation*** *is used and/or*

2) When either the input or output types are of ***Global XYZ*** *type*

An approximate ellipsoid level needs only to be +/- 100 m from the true value for most accuracy requirements. (An error in estimation of the ellipsoid height of a few hundred metres will introduce an error in the transformed horizontal position in the order of millimetres only).

Data Source



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use data source	tick box	tick	

if ticked, a data source is used to select the data to be transformed. If not ticked, a given co-ordinate is transformed.

Data source	input		

*if **use data source** is ticked, the source of data to transform. The source of the data to be adjusted is selected using the data source box. For more information on the data source box see* 4.19.3 Data Source

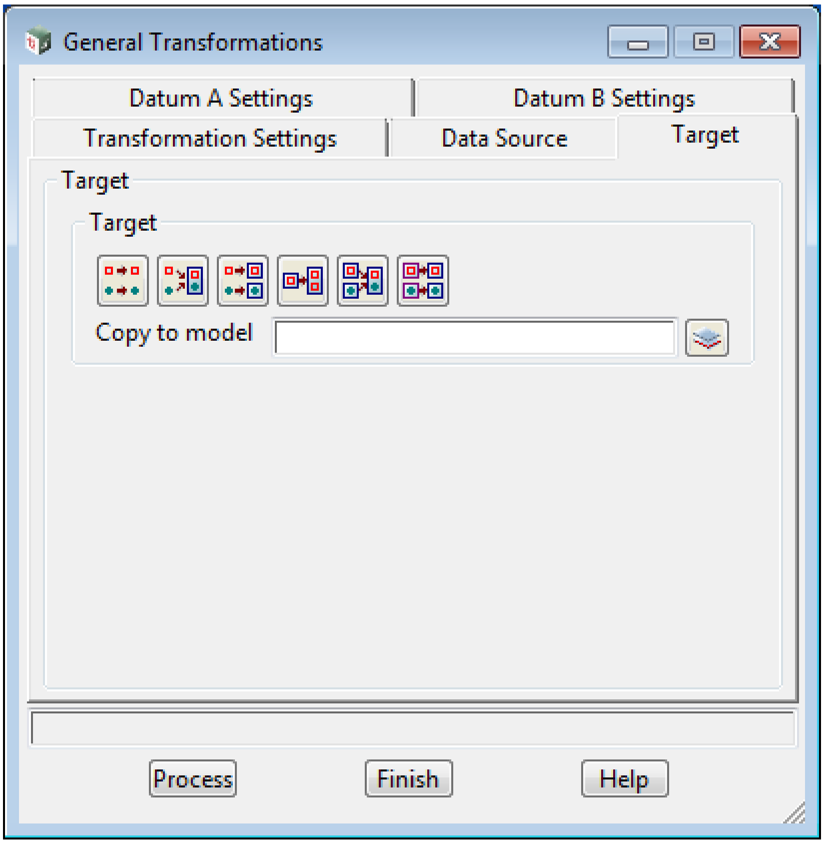
Input coordinates	XYZ box		

*if **use data source** is not ticked, the co-ordinates in this field is transformed.*

Output coordinates	XYZ box		

*if **use data source** is not ticked, the transformed co-ordinates are displayed in this field.*

Target



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Target target

*if **use data source** in the data source is ticked, the target of the transformed data should be specified. The target for the data is selected using the data target box. For more information on the data target box see* 4.19.4 Data Target

Process button

perform the transformation.

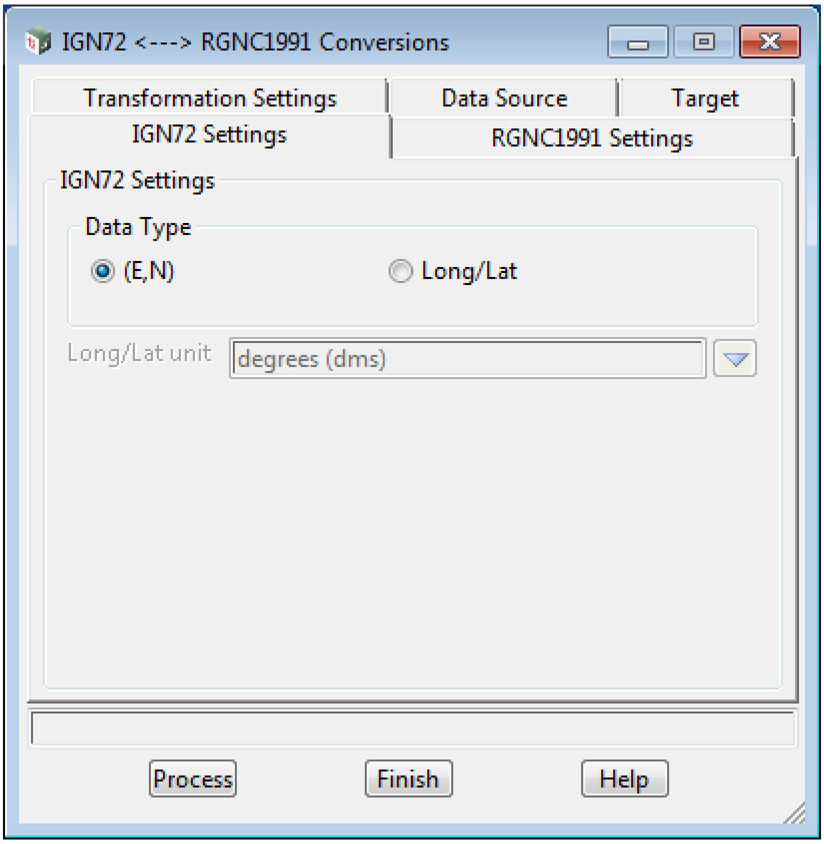
17.10.7 IGN72 <---> RGNC1991

Position of option on menu: **Survey =>Conversions =>IGN72 <->RGNC1991**

This option converts data between the old and new datums for New Caledonia.

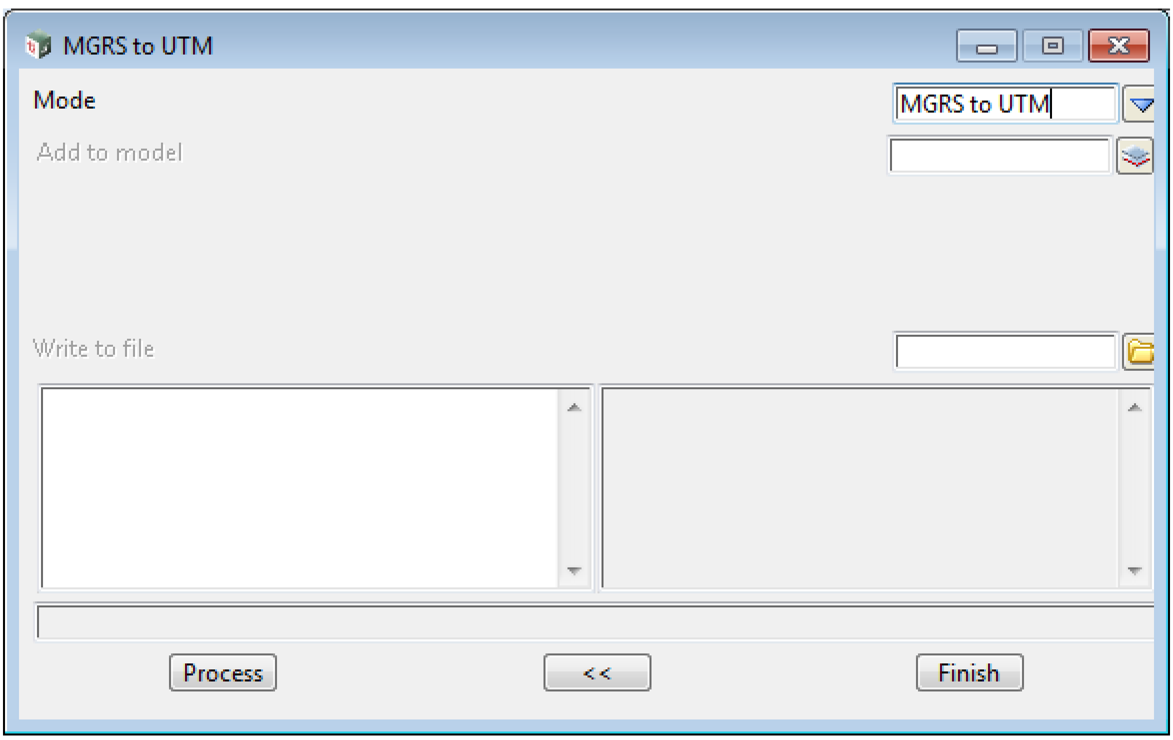
The transformation between the two datums can be by either a seven parameter similarity transformation or a NTv2 grid.

The data to be converted to/from can be (longitude, latitude) or (Easting, Northing).



17.10.8 MGRS <---> UTM

Position of option on menu: **Survey =>Conversions => MGRS<->UTM**

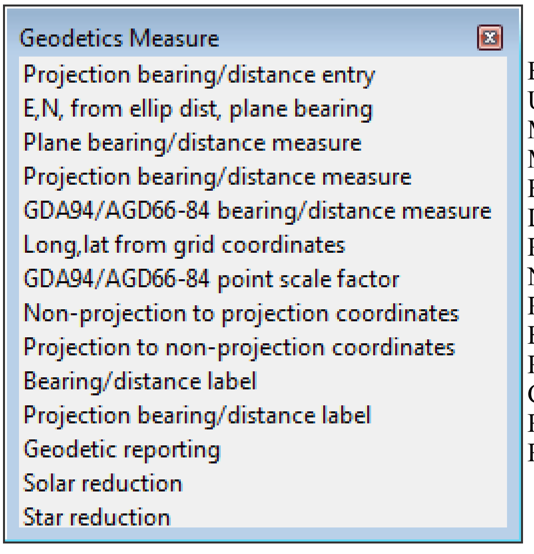


17.11 Geodetic Measures and Entry

Position of menu: **Survey =>Geodetics**

The **Geodetics** walk-right menu is

.



Bearing distance entry for a projection
Use ellipsoid dist, plane brg to create point
Measure bearing, distance, with user input for scale
Measure plane brg, ellip dist, given projection
Bearing dist between 2 pts for GDA/AGD
Long and lat from projection (grid) coords
Point scale factor at a grid point for GDA/AGD
Non-projection to projection coordinate conversion
Projection to non-projection coordinate conversion
Bearing distance label for selected string/segment
Projection brg-dist label for selected string/segment
Create a report based on projection coordinates
Reduce solar observations
Reduce star observations

For Projection bearing /distance entry, go to 17.11.1 Projection Bearing/Distance Entry
E,N calc from ellip dist, plane brg, 17.11.2 E,N from Ellipsoid Distance and Plane Bearing .
Plane bearing /distance measure, 28.6.2 Bearing and Distance
Projection bearing/distance measure, 17.11.4 Projection Bearing and Distance Measure
GDA94/AGD66-84 bearing/distance measure, 17.11.5 GDA94/AGD66-84 Bearing and Distance Measure
Long, lat from grid coordinates, 17.11.6 Longitude, Latitude from Grid Coordinates
GDA94/AGD66-84 Point scale factor, 17.11.7 GDA94/AGD66-84 Point Scale Factor
Non-projection to projection coordinates, 17.11.8 Non-Projection to Projection Coordinates
Projection to non-projection coordinates, 17.11.9 Projection to Non-Projection Coordinates
Bearing/distance label, 17.11.10 Bearing/Distance Label
Projection bearing/distance label, 17.11.11 Projection Bearing/Distance Label
Geodetic reporting, 17.11.12 Geodetic Reporting
Solar reduction, 17.11.13 Solar Reduction
Star reduction, 17.11.14 Star Reduction

For more information about terminology used in these sections, see the Appendix 38 Geodetics Summary .

17.11.1 Projection Bearing/Distance Entry

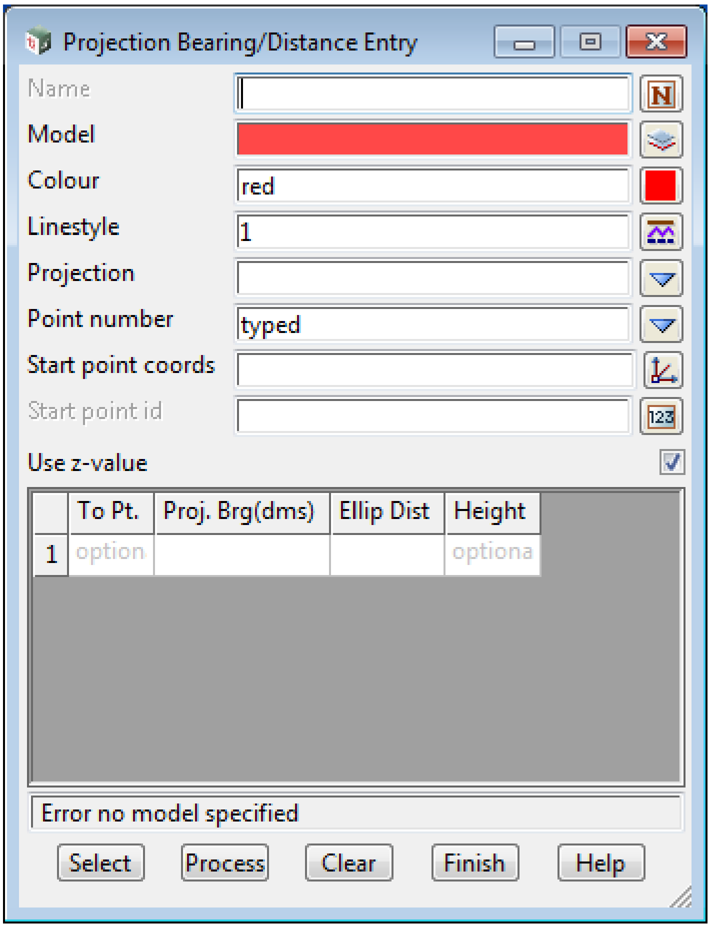
Position of option on menu: **Survey =>Geodetics =>Projection bearing/distance entry**

This section of documentation is a work in progress and will be updated in subsequent releases.

The **Projection bearing/distance entry** option allows the input of a traverse by manual input of projection bearings (the plane bearing) and ellipsoid distances or by selecting an existing string.

For each point, the projection co-ordinates are used with the projection bearing and ellipsoid distance from that point to calculate the projection co-ordinates for the new point taking the map projection into account.

Selecting **Projection bearing/distance entry** brings up the **Projection Bearing/Distance Entry** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	name box		defined names from names.4d file

this field is optional. If non-blank, the name of the new string.

Model	model box		available models

if non-blank, the model of the new string. If blank, the original string model is used.

Colour	colour box	red	available colours

if non-blank, the colour of the new string. If blank, the original string colour is used.

Linestyle	input	1	available line styles

line style of the string.

Projection	input	project projection	available projections

This is the projection that is used for calculating the (Easting, Northing) values.

Point number	choice box		

Start point coords measure box

Use z- value tick box transit

*if **ticked**, the z-values for each point are entered.*
*If **no ticked** then z-values are not entered and are not displayed in the grid.*

Point id input typed none, auto increment, typed

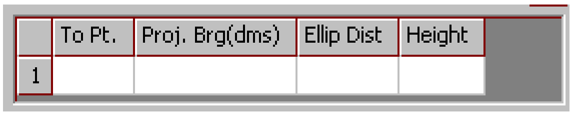
*if **none,** no point ids are entered and the grid control will not show a column for point ids.*

*If **auto increment** then the values of point ids will be incremented by a value of 1 starting from the specified **Start point id**. If no **Start point id** is specified then no point ids will be allocated. No column for point ids is shown in the grid control.*

*if **typed** then the values of point ids will be incremented by a value of 1, starting from the specified **Start point id**. If no **Start point id** is specified then no point ids will be allocated unless a value is entered in the grid control on which time the next value in the grid will have a incremented value. The column for point ids is shown in the grid control.*

VALUES IN GRID grid box

The columns shown in the grid will depend on the selection of the **Use z- value** *and* **Point id** *fields.*



To Pt. input

*if **none** is selected for the* **Point id** *field, this column will not be displayed.*

*if **auto increment** is selected for the* **Point id** *field, this column will not be displayed.*

*if **typed** is selected for the* **Point id** *field, this column will be displayed. In this case, the point id will increment automatically by entering over the field. I f a new value is typed into the To Pt. field, the next line will increment from that number.*

Proj.Brg angle box

The user should enter the projection bearing for the segment into this field. The projection bearing can be defined as the bearing resulting from plane geometry calculations between the two projection coordinates. This is sometimes called the grid bearing in some countries (e.g. New Zealand) and the plane bearing in others such as Australia as defined in the GDA technical manual (ICSM)).

Ellip Dist input box

the user should enter the ellipsoid distance. This distance is the measured horizontal distance that has been reduced onto the ellipsoid taking into account the heights above the ellipsoid at each end of the measured line. This reduction may be by the use of a height scale factor for example. Distances that are measured near mean sea level approximate the ellipsoid distance (since the MSL approximates the ellipsoid in many reference ellipsoids (eg AGD)). The amount of correction will be dependant on the length of line, the heights above the ellipsoid and the reference ellipsoid being used.

Height input box

This column will only be visible in the grid if the **Use z- value** *tickbox has been ticked. The user should enter the height of the point. This value will not be used for calculation of the segment. It will simply be assigned to the newly created vertex.*

Select button

on pressing the select button, a user is able to pick an existing string from the current view. If a non-traverse type string is selected an option to convert it to a traverse string will be given. The grid control will be filled with the relevant information for the traverse string.

If a traverse string has been modified by some other process (e.g. move) an option will be given to adopt the new characteristics of the string as displayed or revert back to the information that originally defined the traverse string. Depending on which option is selected, the grid will be filled with the *relevant information.*

Process button

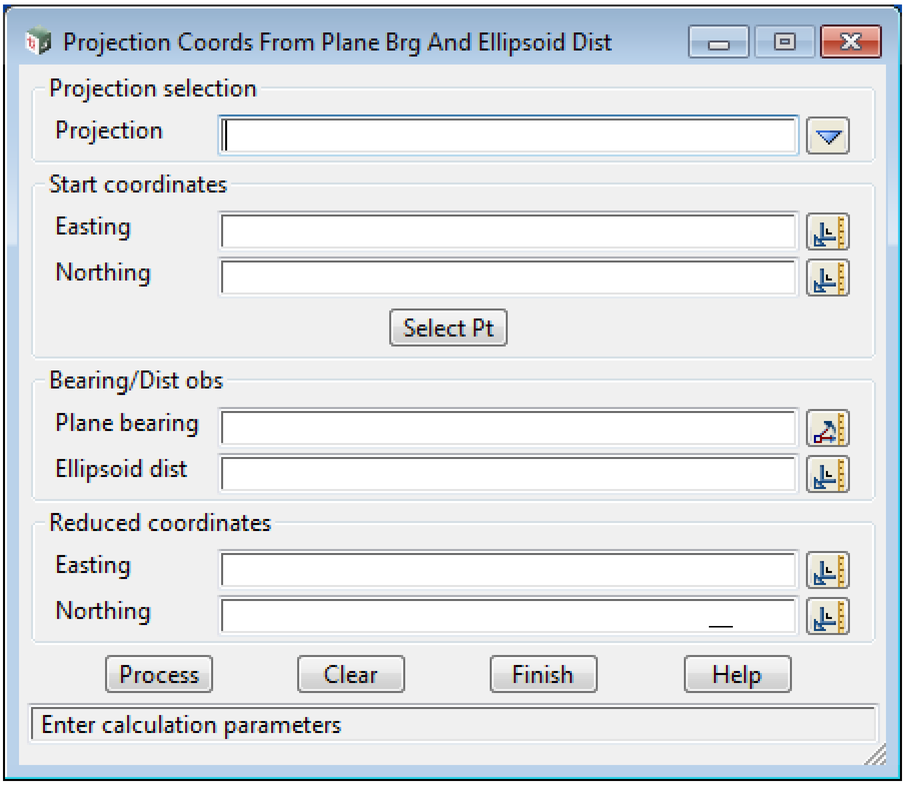
changes to the traverse can be made in the grid control for lines already defined. For example, a distance entry may be incorrectly typed in and edited some time later. By using the process button the traverse string is re-calculated using the current values in the grid control.

17.11.2 E,N from Ellipsoid Distance and Plane Bearing

Position of option on menu: **Survey =>Geodetics =>E,N from ellip dist, plane bearing**

The **E,N calc from ellip dist, plane brg** option allows users to calculate projection coordinates given the projection, start coordinate, plane bearing and ellipsoid distance. The results are given inside the panel.

On selecting the **E,N calc from ellip dist, plane brg** option, the **Projection Coords from plane brg and ellipsoid dist** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Projection selection

Projection choice box current projection available projections

the projection of the data to be used.

Start coordinates

Easting input

the easting value of the start coordinate.

Northing input

the northing value of the start coordinate.

Select Pt button

if the point exists in a view, the coordinate can be entered by selecting the button and then on the required point. The selected points coordinates will be returned to the relevant coordinate boxes.

Bearing/Dist obs

Plane bearing input

the plane bearing (or Projection brg) of the line from the nominated start coordinate. The projection bearing can be defined as the bearing resulting from plane geometry calculations between the two projection coordinates. This is sometimes called the grid bearing in some countries (e.g. New Zealand) and the plane bearing in others (eg Australia as defined in the GDA technical manual (ICSM)).

Ellipsoid distance input

the ellipsoid distance of the line from the nominated start coordinate

Reduced coordinates

Easting input

the calculated easting value of the 2nd point.

Northing input

the calculated northing value of the 2nd point.

Buttons at bottom

Process button

perform the calculation.

Clear button

clear the input fields

17.11.3 Plane Bearing and Distance Measure

Position of option on menu: **Survey =>Geodetics =>Plane bearing/distance measure**

Measures the plane angle and plane distance between two selected points.

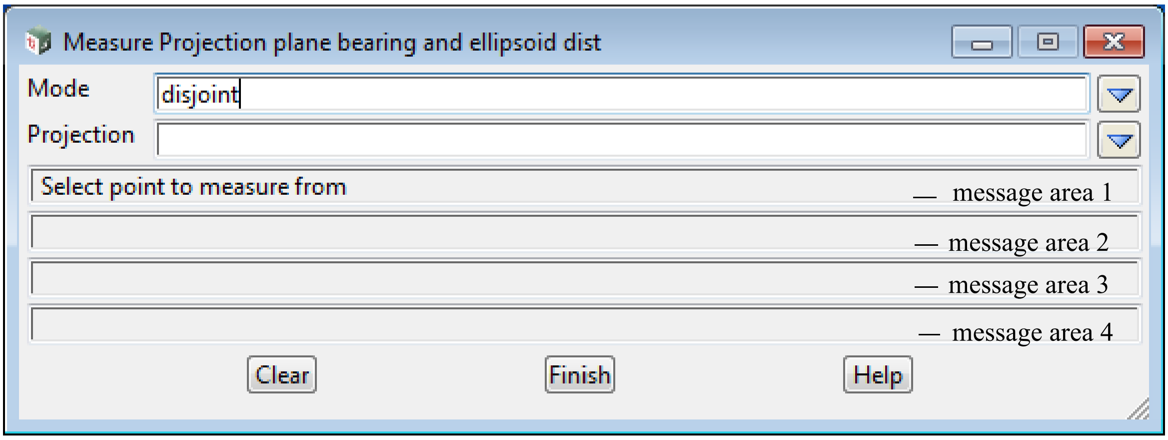
This option has already been documented in the section 28.6.2 Bearing and Distance.

17.11.4 Projection Bearing and Distance Measure

Position of option on menu: **Survey =>Geodetics =>Projection bearing/distance measure**

The **Projection bearing and dist** option allows users to measure projection plane bearing, plane distance, ellipsoid distance, arc-to chord values and a line scale factor.

On selecting the **Projection bearing and dist** option, the **Measure Projection Plane bearing and ellipsoid dist** panel is displayed.



This panel is principally used to display the distances between, and bearing of the line joining, pairs of user selected points.

message area 1: point selection comment

message area 2: dx = dy =

message area 3: plane brg = plane dist = ellipsoid dist =

message area 4: t-T fwd = t-T rev = l.s.f =

where dx is the x coordinate difference between the points, dy is the y coordinate difference between the points

plane brg is the angle measured clockwise from north and is calculated using the coordinate values (Inverse value)

plane dist is the distance calculated using the coordinate values (Inverse value)

ellipsoid dist is the ellipsoid distance calculated using the specified projection and coordinate values.

t-T fwd is the forward arc-to chord correction.

t-T rev is the reverse arc-to chord correction.

l.s.f is the line scale factor. It is calculated by plane distance/ellipsoid distance.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Mode	input	disjoint	disjoint, continuous

In disjoint mode, only the distance between the two points and the bearing (in degrees, minutes and seconds) of the (imaginary) line connecting the two points are displayed.

In continuous mode, after the initial reporting of the bearing/distance, the user is prompted to select the next point(2nd point). In this case the previously selected second point becomes the first point.

Projection	choice	current projection	available projections

the projection of the data to be used.

Clear	button		

when this button is selected, the selection sequence is re-initialised.

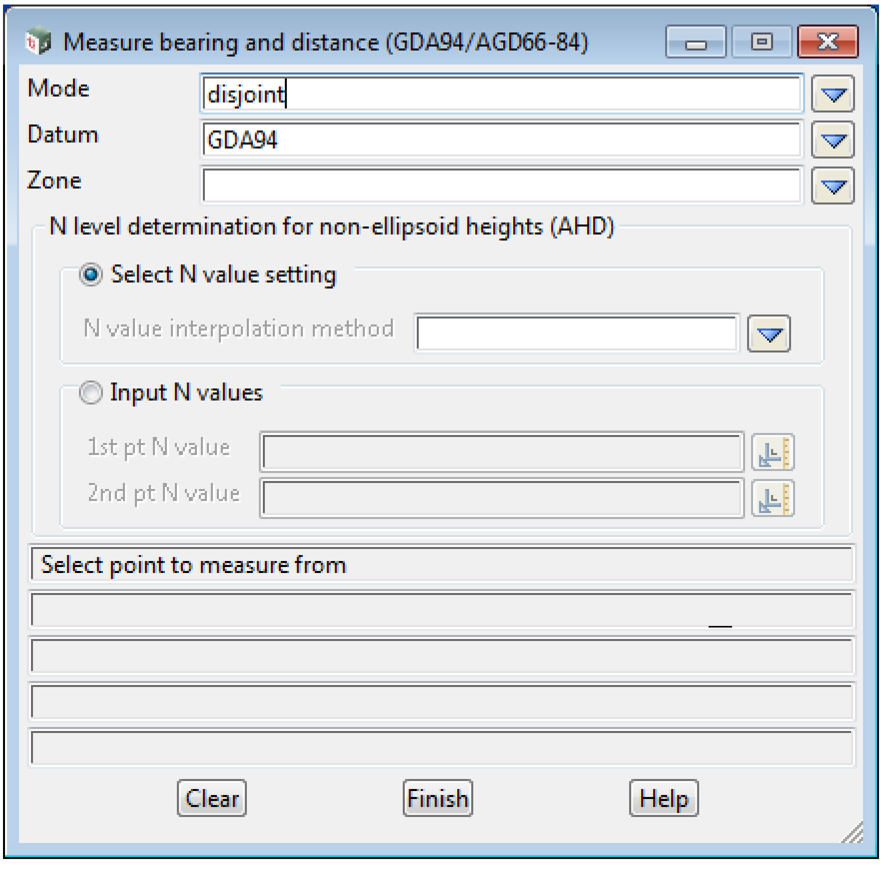
17.11.5 GDA94/AGD66-84 Bearing and Distance Measure

Position of option on menu: Survey =>Geodetics =>GDA94/AGD66-84 bearing/distance

measure

The **Bearing and distance (GDA94/AGD66-84)** is specifically for Australian use. It allows users to select two AMG/MGA coordinates and return the calculated geodetic parameters.

On selecting the **Bearing and distance (GDA94/AGD66-84)** option, the **Bearing and distance (GDA94/AGD66-84)** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Mode	input	disjoint	disjoint, continuous

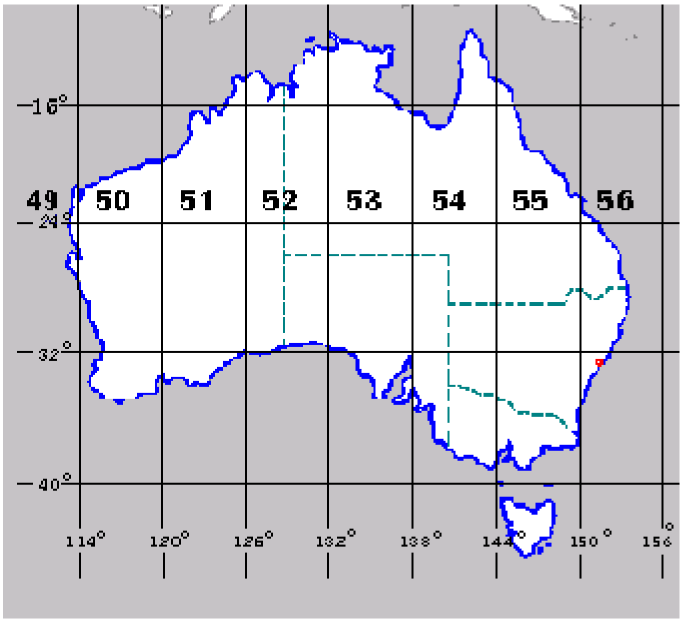
In disjoint mode, the geodetic parameters between the two points are displayed. After the first calculation, further selections must include both the first and second points.

In continuous mode, after the initial reporting of the values, the user is prompted to select the next point(2nd point). In this case the previously selected second point becomes the first point.

Field Description	Type	Defaults	Pop-Up
Datum	choice	current projection	AGD66/84 GDA94

the datum of the data to be used.

I



Zone	choice	current zone	49
			50
			51
			52
			53
			54
			55
			56
			57
			58
			59

the zone of the data to be used.

Select N value setting radio button selected

if the data has non-ellipsoid heights then a conversion to ellipsoid heights is available using a method defined in the **N value interpolation method** *choice box.*

Note: The ellipsoid height is required to calculate the values. However most level datums, such as AHD approximate geoid heights. Therefore, one of the selected points should at least have a non-null value and a means for computing a ellipsoid height. If one point has a valid height and the other does not, an option will be given to assume the other point has the same value.

Various options of converting non-ellipsoid geoid heights to ellipsoid are given by the* N value interpolation method *choice box or the user can enter N values for the first and second points.

N value interpolation method choice box currently set method Available n value methods

the N value method allows the conversion of non-ellipsoid heights to ellipsoid. The methods are define d in the project n value settings. For more information on the n value settings see the section 7.6.7 N values

The N value will be used to convert a geoid height (e.g. AHD) into an ellipsoid height. Ellipsoid height = geoid height + N value. The conversion is used for the calculation only. The original z value for the point will remain unchanged

Input N values radio button unselected

if an N value is known for the area or for the selected points, it will be used to convert a geoid height (e.g. AHD) into an ellipsoid height. Ellipsoid height = geoid height + N value. The conversion is used for the calculation only. The original z value for the point will remain unchanged

1st pt N value input box

if the Input N value option is chosen, an N value is entered for the first point.

2nd pt N value input box

if the Input N value option is chosen, an N value is entered for the second point.

How to Use the Panel and Panel Messages

The results are returned to the panel in the following format:

message area 1	Messages
message area 2	plane brg =, plane dist, dx =, dy =
measure area 3	grid brg =, ellipsoid dist =, line scale factor =
message area 4	Mean height factor =, (Plane dist/level terrain dist) scale factor =
measure area 5	Mean level terrain distance =

Note: Mean height factor is the factor to apply to a measured horizontal distance (usually measured at a height above or below the ellipsoid) to reduce it down to an ellipsoid distance. The other geodetic values and terminology are clearly defined in the GDA technical manual. See http://www.anzlic.org.au/icsm/gdatm/

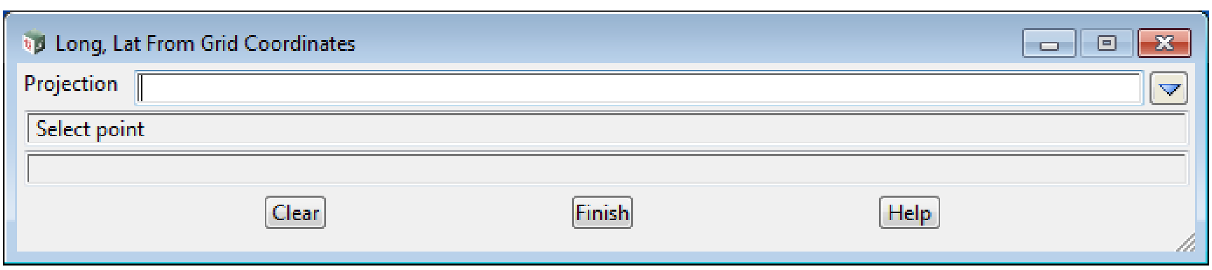
The cycle can then be repeated for another point by simply selecting another point without leaving the option. The **clear** button resets the message areas.

17.11.6 Longitude, Latitude from Grid Coordinates

Position of option on menu: **Survey =>Geodetics =>Long, lat from grid coordinates**

The **Longitude, latitude from grid coordinates** option allows a user to calculate Longitude, latitude and convergence of a point, given a projection and selection of a grid coordinate.

On selecting the **Longitude, latitude from grid coordinates** option, the **Longitude, latitude from grid coordinates** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Projection	choice	current projection	available projections

the projection of the data to be used.

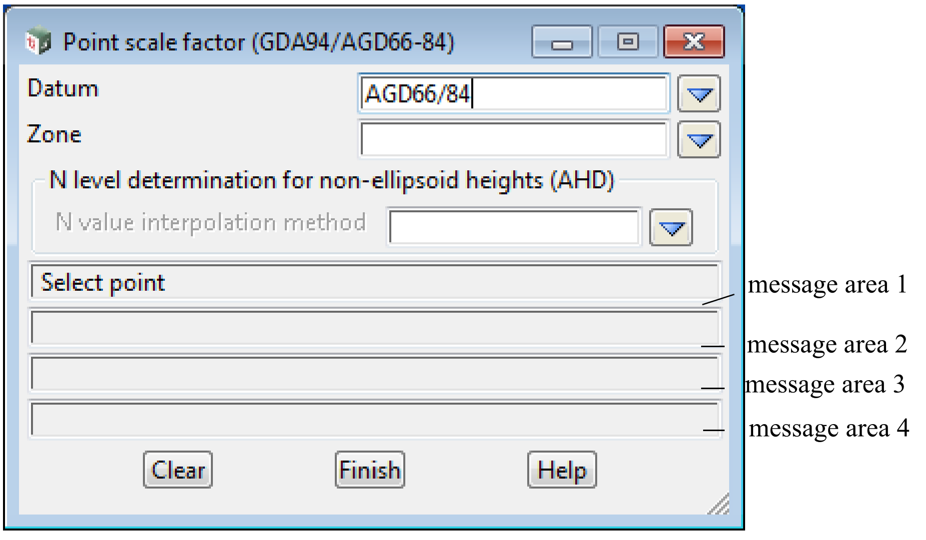
The cycle can then be repeated for another point by simply selecting another point without leaving the option. The **clear** button resets the message areas.

17.11.7 GDA94/AGD66-84 Point Scale Factor

Position of option on menu: **Survey =>Geodetics =>GDA94/AGD66-84 point scale factor**

The **Point scale factor (GDA94/AGD66-84)** is specifically for Australian use. It allows users to select AMG/MGA coordinates and return the point scale factor and latitude and longitude of the point.

On selecting the **Point scale factor (GDA94/AGD66-84)** option, the **Point scale factor (GDA94/AGD66-84)** panel is displayed.

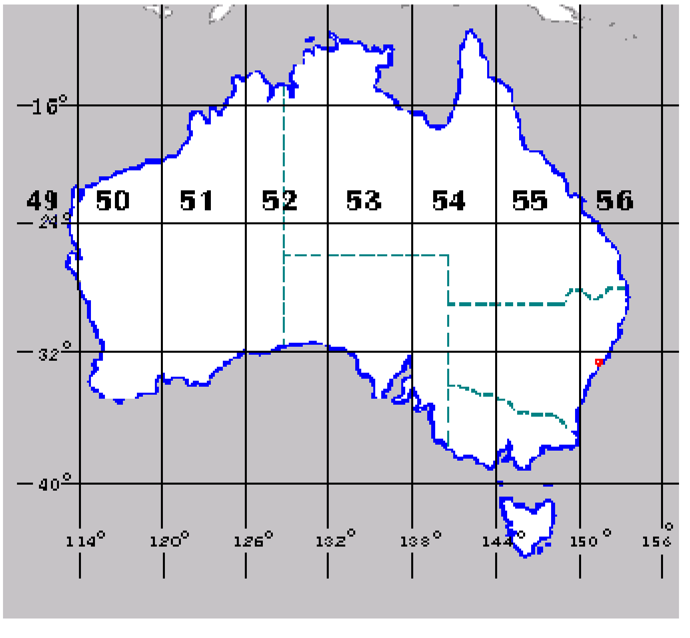


The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Datum	choice	current projection	AGD66/84 GDA94

the datum of the data to be used.

I



Zone	choice	current zone	49

50
51
52
53
54
55
56
57
58
59

the zone of the data to be used.

Note: The ellipsoid height is required to calculate the point scale factor. However most level datums, such as the AHD approximate geoid heights. Therefore, selected points should at least have a non-null value and a means for computing a ellipsoid height. Various options of converting non-ellipsoid geoid heights to ellipsoid are given.

Select N value setting

radio button selected

if the data has non-ellipsoid heights then a conversion to ellipsoid heights is available using a method defined in the **N value interpolation method** ***choice box****.*

N value interpolation method choice box currently set method Available n value methods

the N value method allows the conversion of non-ellipsoid heights to ellipsoid. The methods are define d in the project n value settings. For more information on the n value settings see the section 7.6.7 N values

The N value will be used to convert a geoid height (e.g. AHD) into an ellipsoid height. Ellipsoid height = geoid height + N value. The conversion is used for the calculation only. The original z value for the point will remain unchanged

Input N value radio button unselected

if an N value is known for the area or for the selected point, it will be used to convert a geoid height (e.g. AHD) into an ellipsoid height. Ellipsoid height = geoid height + N value. The conversion is used for the calculation only. The original z value for the point will remain unchanged

Enter N value input box

if the Input N value option is chosen, an N value is entered.

How to Use the Panel and Panel Messages

The results are returned to the panel in the following format:

message area 1	Messages
message area 2	Point scale = , Interpolated or entered N value = (if applicable)
measure area 3	Combined point scale/height factor =
message area 4	Longitude = Latitude =

The cycle can then be repeated for another point by simply selecting another point without leaving the option. The **clear** button resets the message areas.

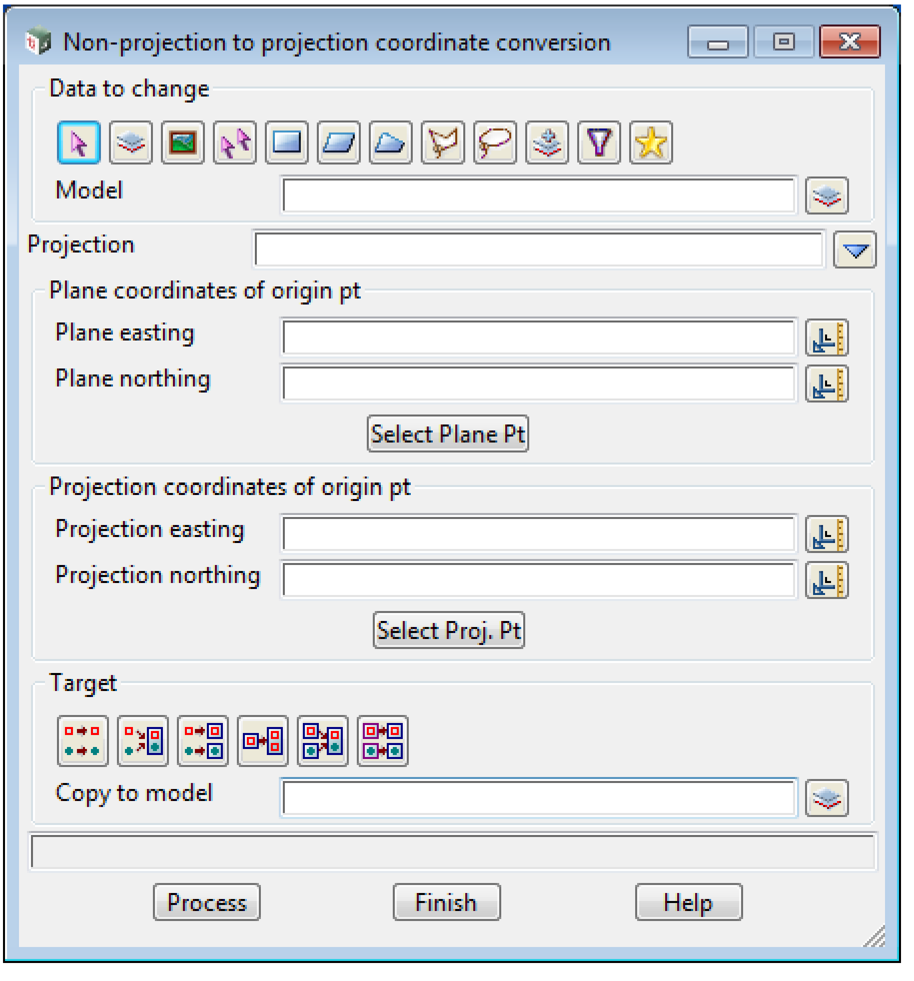
17.11.8 Non-Projection to Projection Coordinates

Position of option on menu: **Survey =>Geodetics =>Non-projection to projection coordinate conversion**

The **Non-projection to projection coordinate conversion** is for the conversion of non-projection coordinates (plane) to a specified projection system. This is done by calculating a series of vectors from an origin point. The origin coordinates in both the plane and projection are known. The plane vector is calculated between the origin and other selected points. Then using the projection coordinates of the origin and the plane vector a projection vector is calculated which allows the projection coordinates of the selected points to be calculated. **The azimuth orientation should be the same in both systems, i.e. no swing should be required.**

The process will convert the vertex of straight segments very effectively. However, care should be taken when trying to convert non-linear segments such as curves. These may have construction entities such as TP’s that no longer have the same relationship with a centre point after conversion for example. This is due to differing scale factors of points due to the projection.

On selecting the **Non-projection to projection coordinates** option, the **Non-projection to projection coordinate conversion** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data to change	source box	model	

the source of objects to be converted are selected using the data source box. For more information on the data source box see 4.19.3 Data Source

Projection	projection box	current projection	available projections

the projection to which the data is to be converted to is specified in the projection box.

Plane easting	input box		

the plane easting value of the origin point.

Plane northing input box

the plane northing value of the origin point.

Select Plane Pt button

selection of the button allows the point to picked from a view. The user selects and accepts the point and then the values are placed into the relevant input boxes.

Projection easting input box

the projection easting value of the origin point.

Projection northing input box

the projection northing value of the origin point.

Select Projection Pt button

selection of the button allows the point to picked from a view. The user selects and accepts the point and then the values are placed into the relevant input boxes.

Target target

the target of the transformed data should be specified. The target for the data is selected using the data target box. For more information on the data target box see 4.19.4 Data Target

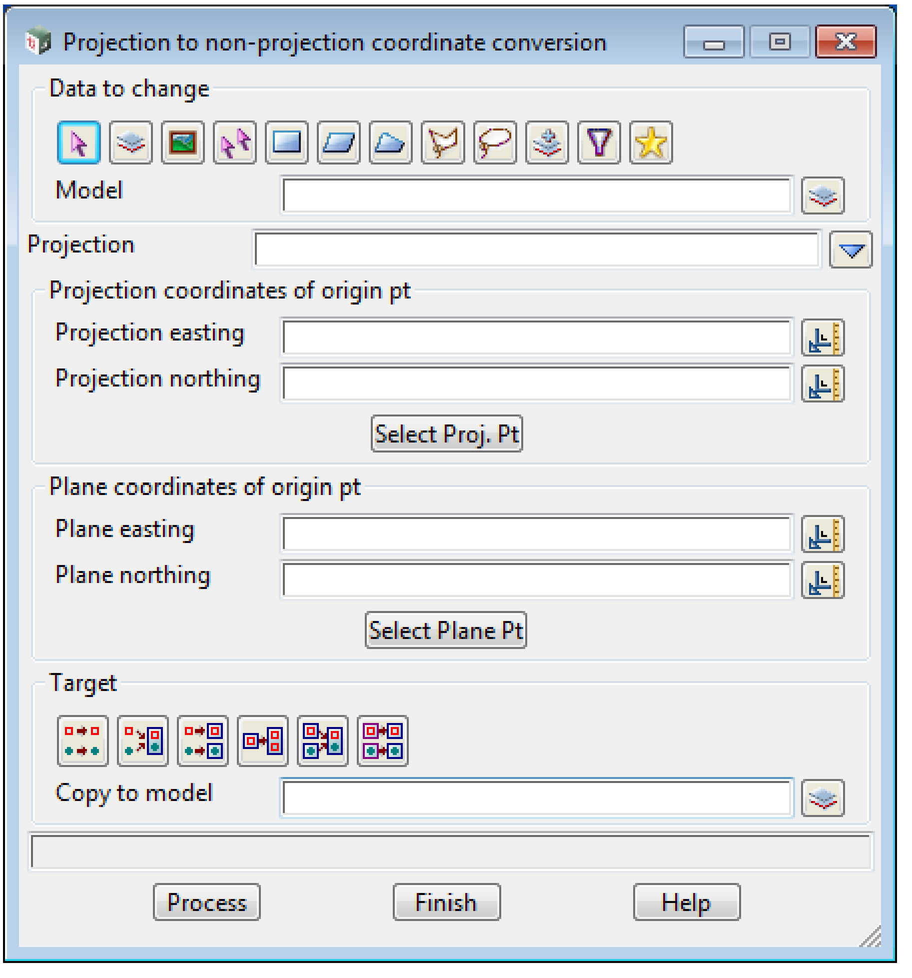
17.11.9 Projection to Non-Projection Coordinates

Position of option on menu: **Survey =>Geodetics => Projection to non-projection coordinate conversion**

The **Projection to non-projection coordinate conversion** is for the conversion of projection coordinates to plane coordinates. This is done by calculating a series of vectors from an origin point. The origin coordinates in both the plane and projection are known. The projection vector is calculated between the origin and other selected points. Then using the plane coordinates of the origin and the projection vector a plane vector is calculated which allows the projection coordinates of the selected points to be calculated. **The azimuth orientation should be the same in both systems, *i.e.* no swing should be required.**

The process will convert the vertex of straight segments very effectively. However, care should be taken when trying to convert non-linear segments such as curves. These may have construction entities such as TP's that no longer have the same relationship with a centre point after conversion for example. This is due to differing scale factors of points due to the projection.

On selecting the **Projection to non-projection coordinates** option, the **Projection to non-projection coordinate conversion** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data to change	source box	model	

the source of objects to be converted are selected using the data source box. For more information on the data source box see 4.19.3 Data Source

Projection	choice box	current projection	available projections

the projection to which the data is to be converted from is specified in the projection box.

Projection easting input box

the projection easting value of the origin point.

Projection northing input box

the projection northing value of the origin point.

Select Proj. Pt button

selection of the button allows the point to picked from a view. The user selects and accepts the point and then the values are placed into the relevant input boxes.

Plane easting input box

the plane easting value of the origin point.

Plane northing input box

the plane northing value of the origin point.

Select Plane Pt button

selection of the button allows the point to picked from a view. The user selects and accepts the point and then the values are placed into the relevant input boxes.

Target target

the target of the transformed data should be specified. The target for the data is selected using the data target box. For more information on the data target box see 4.19.4 Data Target.

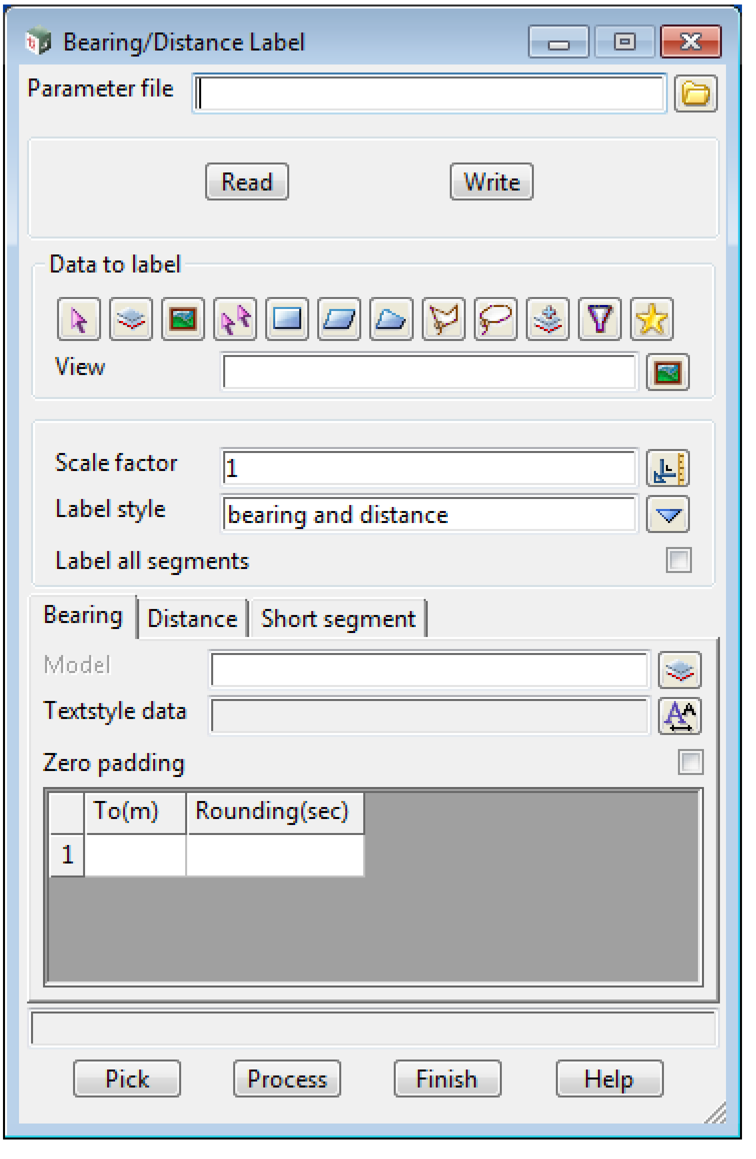
17.11.10 Bearing/Distance Label

Position of option on menu: **Survey =>Geodetics => Bearing/distance label**

The **Bearing/distance label** is for the labelling of a selected string with bearing and distance information. It allows a parameter file to be setup (*.lbf) which can be written and read into the option. This file allows the setting of panel parameters such as rounding, textstyle data etc.

The pick should be with direction. This will effect the bearing that is labelled (+/-180 degrees).

On selecting the **Bearing/distance label** option, the **Bearing/distance label** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Parameter file	file box		*.lbf files

the parameter file can be read in which will fill in the remaining values within the panel. The user can make changes and save the choices as a different file, thus enabling a library of labelling options to be created.

Read	button		

if a valid parameter file is entered into the parameter file field, the user can press the read button to load the information in the file into the panel.

Write	button		

if a valid parameter file name is entered into the parameter file field, the user can press the write button to save the edited panel information into a file so that it can be read in at a later date.

Scale factor	input box	Current Scale factor at Central meridian if set	

if a scale factor is specified, the distance values labelled will use the scale factor to compute the label distance. This scale factor will be applied to the calculated plane distance from coordinates in the following manner:

label distance = plane distance /scale factor.

Label style	choice box	bearing and distance	bearing and distance bearing distance distance bearing

the label style.

Bearing and distance equates to having the bearing and distance justification point at the same point on the midpoint of the segment (the text justification can make the bearing bottom-middle and the distance top-middle for example to show the bearing above the line and the distance below)

Bearing distance equates to having the bearing and distance justification point along side. For example, a label may be a bearing on the left and a distance on the right

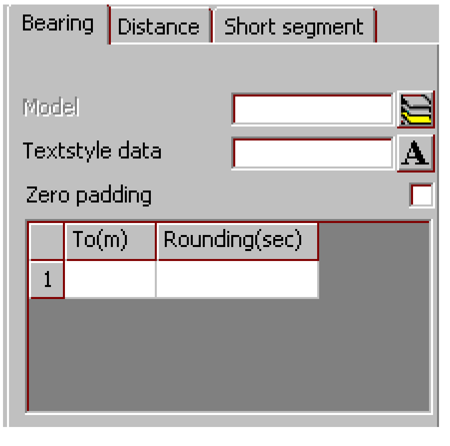
Distance bearing equates to having the distance and bearing justification point along side. For example, a label may be a distance on the left and a bearing on the right

Label all segments	tick box	unticked	

if ticked all the segments of the selected string will be labelled.

Bearing tab

the following fields are located on the bearing tab.

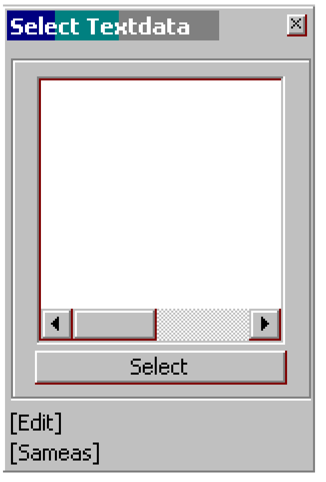


Field Description	Type	Defaults	Pop-Up
Model	model box		current models

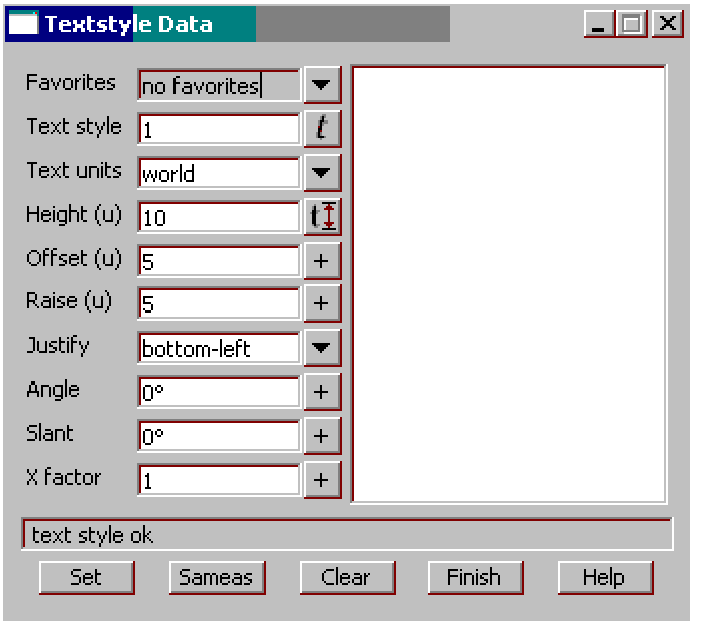
if a valid model name is specified, the bearing text will be created in that model.

Textstyle data textstyle box current textstyle names

on pressing the textstyle data button a list of available textdata predefined names read from the ***texstyles_name.4d*** *file are displayed.*



If no names exist, the user can edit the current settings by selecting the edit button and bring up the **textstyle data** *panel. This allows for definition of textstyle, units, height offset raise etc.*



The textstyle data panel enables the user to define multiple text parameters.

Zero padding tick box unticked

if ticked, the labels will have zero's inserted so that at least 2 characters exist for the minutes and second part of the label. For example, A bearing of 10^o *6' 5" would become* 10^o *06' 05" with padding.*

The values in the grid relate to the rounding of bearings for labelling. Many different ranges can be specified in the grid with each line representing a valid rounding range.

To(m) input box

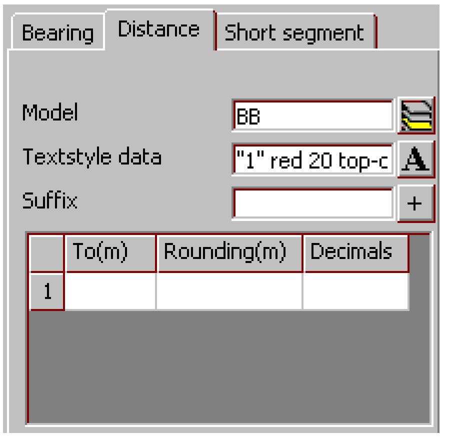
the rounding upper range limit in meters. This value together with the rounding specified, allow the bearings between certain distance ranges to be rounded to a certain value. The first line in the grid is *from 0 to the specified distance range. Any following lines in the grid use the* **To(m)** *value in the previous line for the lower limit to the* **To(m)** *value in that line.*

Rounding (sec) input box

the rounding value given relates to the distance range. Any bearing within the distance range specified will be rounded to the value in seconds. e.g for a rounding value of 60 seconds the bearing will be rounded to the nearest minute.

Distance tab

the following fields are located on the distance tab.

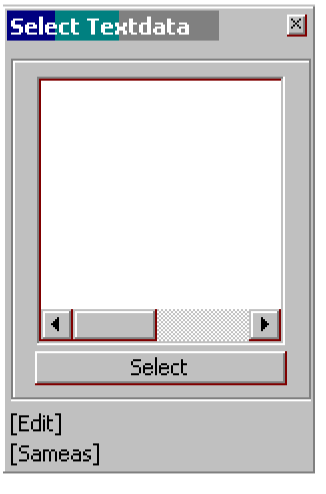


Field Description	Type	Defaults	Pop-Up
Model	model box		current models

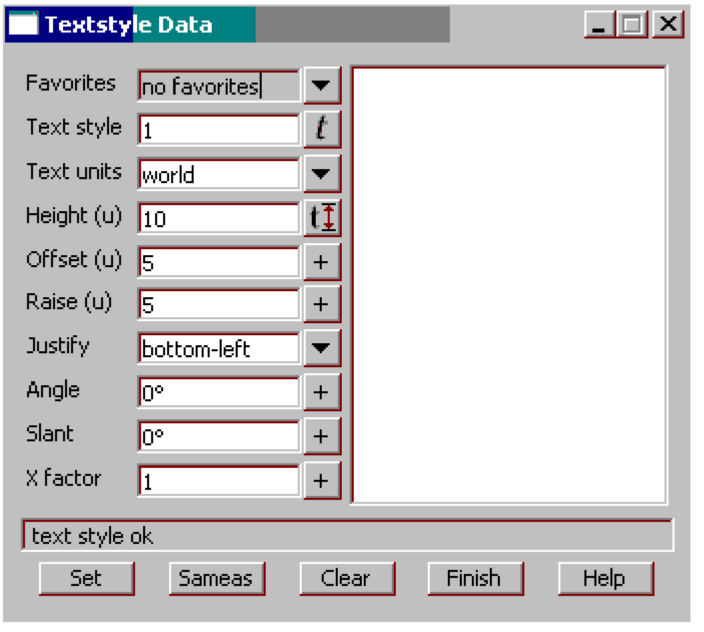
if a valid model name is specified, the distance text will be created in that model.

Textstyle data	textstyle box		current textstyle names

on pressing the textstyle data button a list of available textdata predefined names read from the ***texstyles_name.4d*** *file are displayed.*



If no names exist, the user can edit the current settings by selecting the edit button and bring up the **textstyle data** *panel. This allows for definition of textstyle, units, height offset raise etc.*



The textstyle data panel enables the user to define multiple text parameters.

Suffix input box

if entered, this suffix will be appended to the end of the distance label.

The values in the grid relate to the rounding of distances for labelling. Many different ranges can be specified in the grid with each line representing a valid rounding range.

To(m) input box

the rounding upper range limit in meters. This value together with the rounding and number of decimals specified, allow the distance to be rounded to a certain value. The first line in the grid is from 0 to the specified distance range. Any following lines in the grid use the **To(m)** *value in the previous line for the lower limit to the* **To(m)** *value in that line.*

Rounding (m) input box

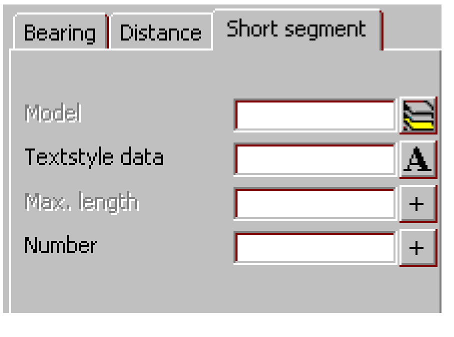
the rounding value given relates to the distance range. Any distance within the distance range specified will be rounded to the value in metres. e.g for a rounding value of 0.05 a distance of 125.261 will be rounded to a value of 125.25

Decimals input box

the number of decimal places can be specified. For a distance of 125.261, rounding of 0.05 and Decimals equal to 3, the label will be 125.250.

Short segment tab

the following fields are located on the short segment tab.

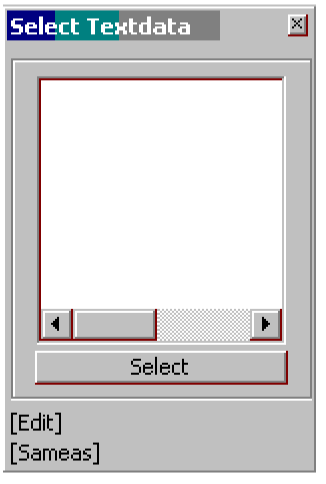


Field Description	Type	Defaults	Pop-Up
Model	model box		current models

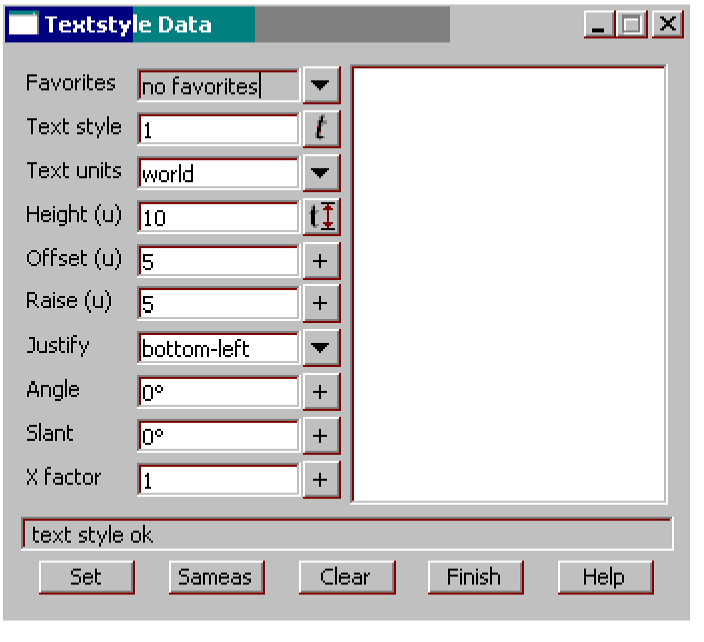
if a valid model name is specified, the short segment text will be created in that model.

Textstyle data	textstyle box		current textstyle names

*on pressing the textstyle data button a list of available textdata predefined names read from the **texstyles_name.4d** file are displayed.*



If no names exist, the user can edit the current settings by selecting the edit button and bring up the **Textstyle data** *panel. This allows for definition of textstyle, units, height offset raise etc.*



The textstyle data panel enables the user to define multiple text parameters.

Max. length	input box		

if entered, this value will be compared with the selected segment distance. If the selected distance is less than this distance, the short line labelling will override any bearing and distance labelling.

Number	input box		

if entered, this value will be used for the short segment text.

Pick button

on selection the user can then select a segment/line from a view with direction. The direction will give the bearing direction which is usually +/- 180 degrees different. eg a segment with a bearing of 24 degrees can be labelled with a bearing of 204 by picking in a south west direction.

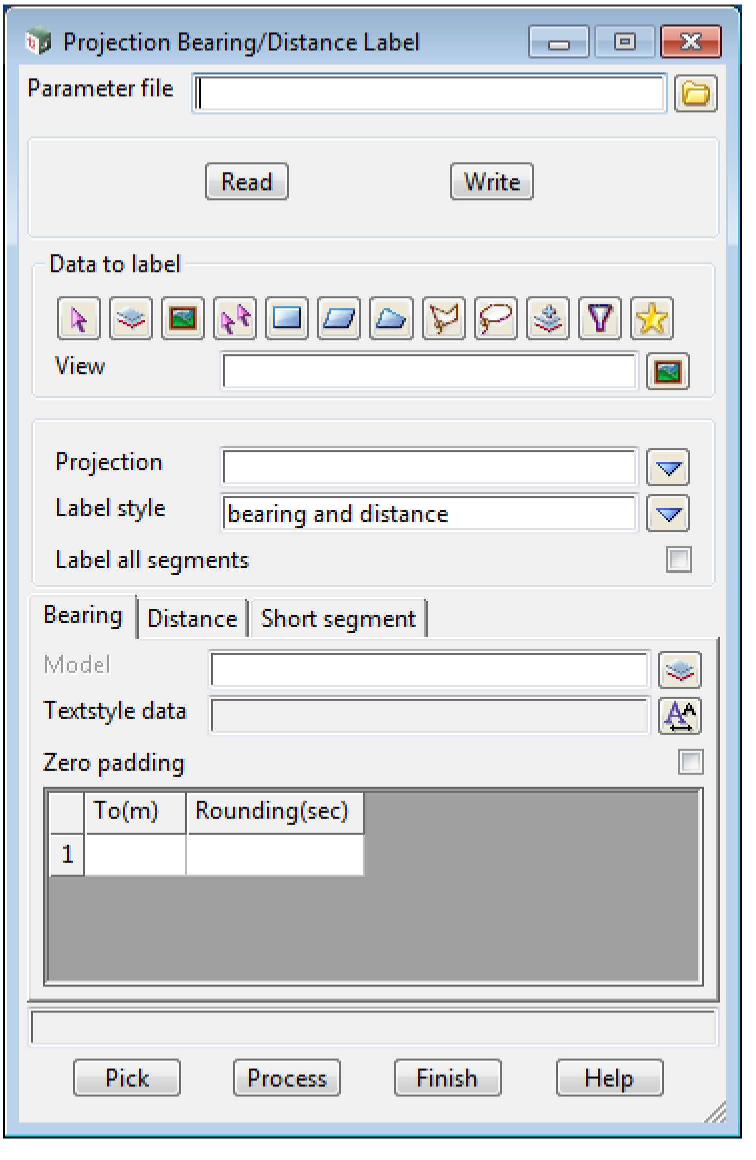
17.11.11 Projection Bearing/Distance Label

Position of option on menu: **Survey =>Geodetics => Projection bearing/distance label**

The **Projection bearing/distance label** is for the labelling of a selected string with plane bearing and ellipsoid distance information that is in terms of a selected projection. It allows a parameter file to be setup (*.lbf) which can be written and read into the option. This file allows the setting of panel parameters such as rounding, textstyle data etc.

The pick should be with direction. This will effect the bearing that is labelled (+/-180 degrees of actual direction).

On selecting the **Projection bearing/distance label** option, the **Projection bearing/distance label** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Parameter file	file box		*.lbf files

the parameter file can be read in which will fill in the remaining values within the panel. The user can make changes and save the choices as a different file, thus enabling a library of labelling options to be created.

Read	button		

if a valid parameter file is entered into the parameter file field, the user can press the read button to load the information in the file into the panel.

Write	button		

if a valid parameter file name is entered into the parameter file field, the user can press the write button to save the edited panel information into a file so that it can be read in at a later date.

Projection	choice	current projection	available projections

the projection of the data to be used for labelling.

Label style	choice box	bearing and distance	bearing and distance, bearing distance, distance bearing

the label style.

Bearing and distance equates to having the bearing and distance justification point at the same point on the midpoint of the segment (the text justification can make the bearing bottom-middle and the distance top-middle for example to show the bearing above the line and the distance below)

Bearing distance equates to having the bearing and distance justification point along side. For example, a label may be a bearing on the left and a distance on the right

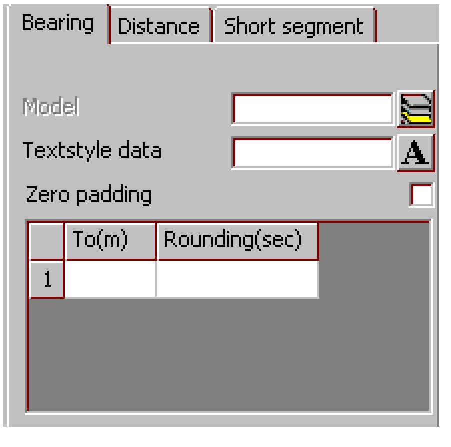
Distance bearing equates to having the distance and bearing justification point along side. For example, a label may be a distance on the left and a bearing on the right

Label all segments	tick box	unticked	

if ticked all the segments of the selected string will be labelled.

Bearing tab

the following fields are located on the bearing tab.

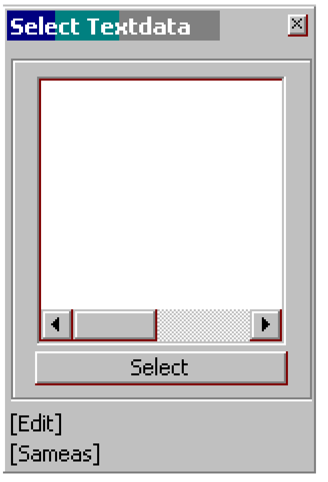


Field Description	Type	Defaults	Pop-Up
Model	model box		current models

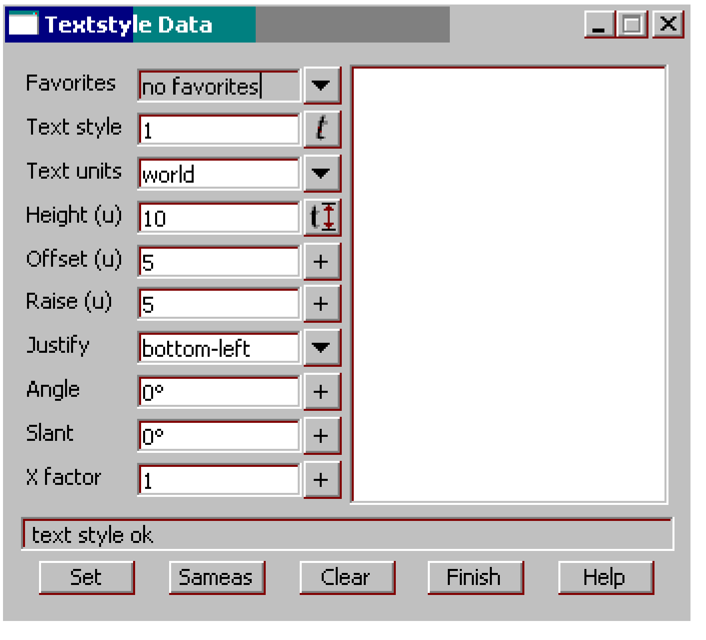
if a valid model name is specified, the bearing text will be created in that model.

Textstyle data	textstyle box		current textstyle names

on pressing the textstyle data button a list of available textdata predefined names read from the ***texstyles_name.4d*** *file are displayed.*



If no names exist, the user can edit the current settings by selecting the edit button and bring up the **textstyle data** *panel. This allows for definition of textstyle, units, height offset raise etc.*



The textstyle data panel enables the user to define multiple text parameters.

Zero padding tick box unticked

if ticked, the labels will have zero's inserted so that at least 2 characters exist for the minutes and second part of the label. For example, A bearing of 10^o *6' 5" would become* 10^o *06' 05" with padding.*

The values in the grid relate to the rounding of bearings for labelling. Many different ranges can be specified in the grid with each line representing a valid rounding range.

To(m) input box

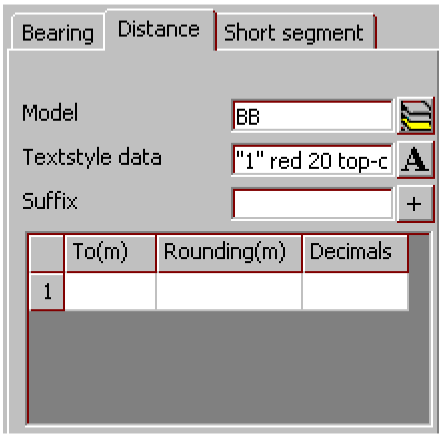
the rounding upper range limit in meters. This value together with the rounding specified, allow the bearings between certain distance ranges to be rounded to a certain value. The first line in the grid is from 0 to the specified distance range. Any following lines in the grid use the **To(m)** *value in the previous line for the lower limit to the* **To(m)** *value in that line.*

Rounding (sec) input box

the rounding value given relates to the distance range. Any bearing within the distance range specified *will be rounded to the value in seconds. e.g for a rounding value of 60 seconds the bearing will be rounded to the nearest minute.*

Distance tab

the following fields are located on the distance tab.

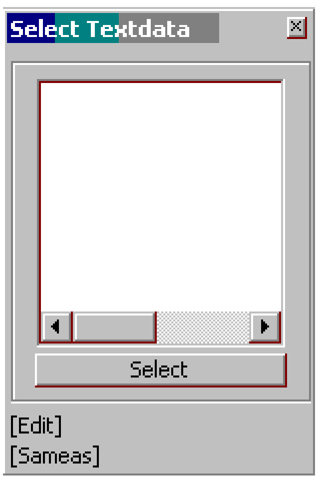


Field Description	Type	Defaults	Pop-Up
Model	model box		current models

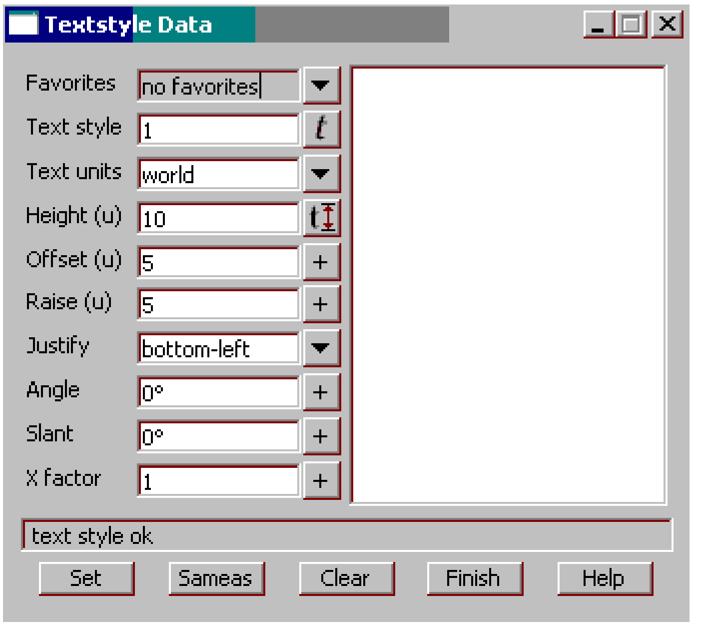
if a valid model name is specified, the distance text will be created in that model.

Field Description	Type	Defaults	Pop-Up
Textstyle data	textstyle box		current textstyle names

*on pressing the textstyle data button a list of available textdata predefined names read from the **texstyles_name.4d** file are displayed.*



If no names exist, the user can edit the current settings by selecting the edit button and bring up the **textstyle data** *panel. This allows for definition of textstyle, units, height offset raise etc.*



The textstyle data panel enables the user to define multiple text parameters.

Suffix input box

if entered, this suffix will be appended to the end of the distance label.

The values in the grid relate to the rounding of distances for labelling. Many different ranges can be specified in the grid with each line representing a valid rounding range.

To(m) input box

the rounding upper range limit in meters. This value together with the rounding and number of decimals specified, allow the distance to be rounded to a certain value. The first line in the grid is from 0 to the specified distance range. Any following lines in the grid use the **To(m)** *value in the previous line for the lower limit to the* **To(m)** *value in that line.*

Rounding (m) input box

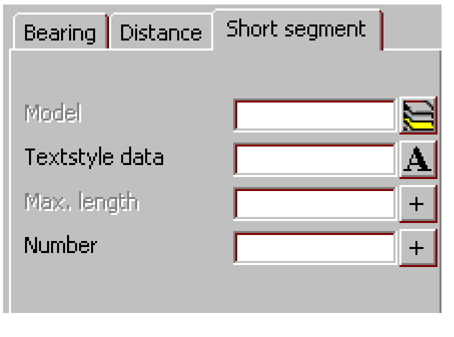
the rounding value given relates to the distance range. Any distance within the distance range specified will be rounded to the value in metres. e.g for a rounding value of 0.05 a distance of 125.261 will be rounded to a value of 125.25

Decimals input box

the number of decimal places can be specified. For a distance of 125.261, rounding of 0.05 and Decimals equal to 3, the label will be 125.250.

Short segment tab

the following fields are located on the short segment tab.

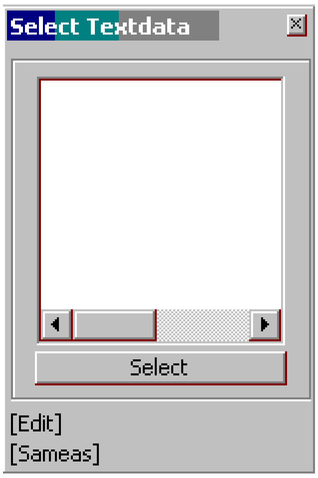


Field Description	Type	Defaults	Pop-Up
Model	model box		current models

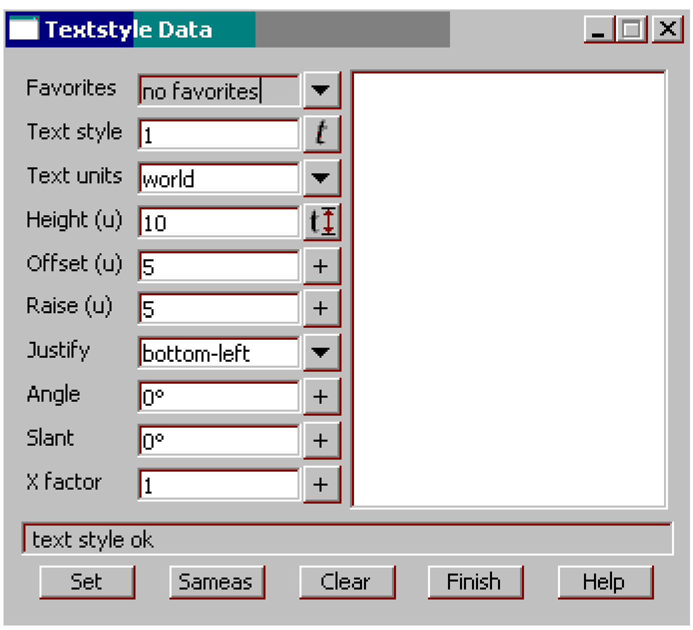
if a valid model name is specified, the short segment text will be created in that model.

Textstyle data	textstyle box		current textstyle names

on pressing the textstyle data button a list of available textdata predefined names read from the ***texstyles_name.4d*** *file are displayed.*



If no names exist, the user can edit the current settings by selecting the edit button and bring up the **textstyle data** *panel. This allows for definition of textstyle, units, height offset raise etc.*



The textstyle data panel enables the user to define multiple text parameters.

Max. length	input box		

if entered, this value will be compared with the selected segment distance. If the selected distance is less than this distance, the short line labelling will override any bearing and distance labelling.

Number	input box		

if entered, this value will be used for the short segment text.

Pick button

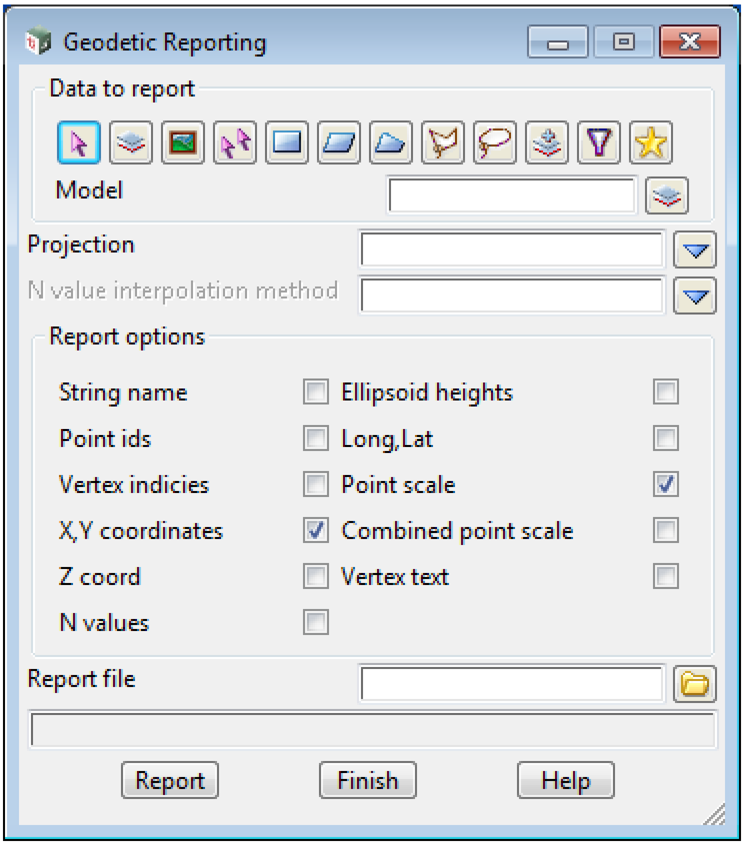
on selection the user can then select a segment/line from a view with direction. The direction will give the bearing direction which is usually +/- 180 degrees different. eg a segment with a bearing of 24 degrees can be labelled with a bearing of 204 by picking in a south west direction.

17.11.12 Geodetic Reporting

Position of option on menu: **Survey =>Geodetics => Geodetic reporting**

The **Geodetic reporting** option is for the reporting of selected data (projection coordinates) in terms of geodetic positions and variables. The user can choose from a number of different reporting options including, point scale factor, Longitude and latitude, point IDs, *etc.*

On selecting the **Geodetic reporting** option, the **Geodetic Reporting** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Data to report source box model

the source of objects to be reported are selected using the data source box. For more information on the data source box see 4.19.3 Data Source *. The selected data must be in terms of the selected map projection.coordinate system.*

Projection choice current projection available projections

the projection of the data to be reported.

N value interpolation method choice box currently set method Available n value methods

the N value method allows the conversion of non-ellipsoid heights to ellipsoid. The methods are define d in the project n value settings. For more information on the n value settings see the section 7.6.7 N values

The N value will be used to convert a geoid height (e.g. AHD) into an ellipsoid height. Ellipsoid height = geoid height + N value. The conversion is used for the reporting only. The original z value for the point will remain unchanged

String name tick box

if selected, any valid string names will be shown in the report.

Point IDs tick box

if selected, any valid point ids will be shown in the report.

Vertex indices tick box

if selected, vertex numbers will be shown in the report.

X,Y coordinates tick box ticked

if ticked, the coordinates of the selected data will be reported.

Z coord tick box

if selected, any valid z values will be shown in the report.

N values tick box

if ticked, the N value calculated for the given point will be reported (if valid).

Ellipsoid heights tick box

if selected, the calculated ellipsoid height will be reported. If no N values are calculated, this will be the same as the Z value.

Long, Lat tick box

if selected, the calculated longitude and latitude will be reported.

Point scale tick box ticked

if selected, the point scale for each point will be reported.

Combined point scale tick box

if selected, the combined point scale for each point will be reported. This combined scale factor is the product of the point scale factor and a height scale factor. The height scale factor is computed from the ellipsoid height of the point.

Text tick box

if selected, any valid text values will be shown in the report.

Report file input *.rpt

name of the file to report to.

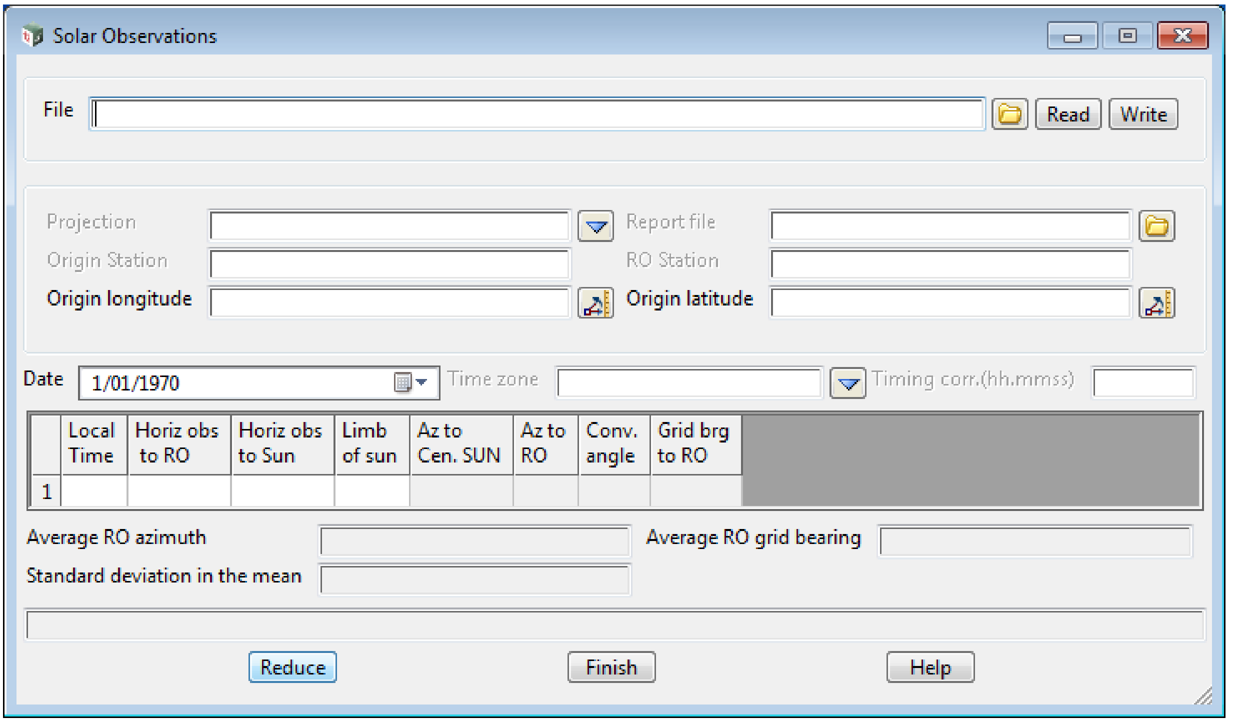
Report button

17.11.13 Solar Reduction

Position of option on menu: **Survey =>Geodetics => Solar reduction**

The **Solar reduction** option allows the reduction of observations to the sun by the hour angle method. The user can specify either limb or the centre of the sun, and a number of observations can be processed together. **The option does not require almanac details as these are processed internally using algorithms supplied by the US naval observatory**. These algorithms will produce accuracy in the order of 2 arc seconds in azimuth for the period AD1800 - 2050

On selecting the **Solar reduction** option, the **Solar Observations** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
File	file box		*.ast files

a filename can be specified for reading or writing a file.

Read	button		

if a valid file exists, the file contents can be loaded into the panel.

Write	button		

if a valid name is specified and the valid values are entered into all of the fields, the user can write the input data to a file.

Projection	choice	current projection	available projections

a projection can be specified so that a grid convergence value can be computed. This value will be used to compute the grid or projection bearing to the RO. If left blank, only the True azimuth will be reported.

Report file	input		*.rpt

name of the file to report to.

Origin station	input		

name of the origin station

RO station	input		

name of the reference object station

Origin longitude	input		

the observation station's longitude should be specified in DMS format.

Origin latitude	input		

the observation station's latitude should be specified in DMS format.

Date	date box	current system date	

the date of the observations as at the observers position should be entered. This can be by direct entry *into the date field or by selecting the day, month, year from a date pop-up panel. The user can scroll through different months by selecting the arrow buttons on the pop-up date panel.*

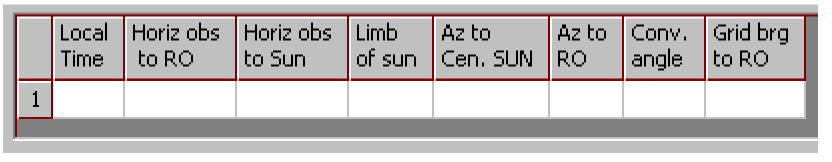
Time zone time zone box major time zones

a valid time zone for the observers position and local time should be selected from the pop-up list.

Timing corr.(hh.mmss) input

the correction to apply for any timing error including stopwatch corrections should be entered. The value should be in a hours, minutes seconds in hp format.

GRID VALUES: input



Local time input

the local time of the observation specified in 24 hour time in HMS. eg. 18.121002

Horiz obs to RO input

the horizontal angle (bearing) observed to the RO for the set of measurements. A set in this case includes a pointing to the RO and sun (at a certain time) in the same face.

Horiz obs to SUN input

the horizontal angle (bearing) observed to the Sun for the set of measurements.(The observation should not be corrected for the suns semi diameter as this is done internally by the reduction process)

If observation s are made on both faces, a separate line should be entered for each face. e.g Face 1 to RO may be 0.0000 and the face 1 to the sun may be 50.0000 which would be on the same line, then the observer may change face to face 2 and observe to the sun again followed by the RO. The face 2 observations should appear on the next line eg 180.0004 230.0110.

Limb of sun choice box Centre
Left limb
Right limb

the limb of the sun observed should be specified. This enables the suns semi diameter correction to be applied. This also allows the observations to be read in any order.

Az to cen. SUN output

the true azimuth to the sun will be displayed in this field on successful reduction of the observations.

Az to RO output

the true azimuth to the RO will be displayed in this field on successful reduction of the observations.

Conv. angle output

the convergence angle at the observation point will be displayed in this field if a valid projection is specified and there is a successful reduction of the observations.

Grid brg to RO output

the grid bearing to the RO will be displayed in this field if a valid convergence angle is calculated and there is a successful reduction of the observations. The grid bearing is calculated by:

Grid brg = True azimuth + convergence

The fields and buttons used in this panel below the grid have the following functions.

Field Description Type Defaults Pop-Up

Average RO azimuth output

if there is as successful reduction, the average RO azimuth value will be reported in this field.

Average RO grid bearing output

if there is as successful reduction, and the projection given, the average RO grid bearing value will be reported in this field.

Standard deviation in the mean output

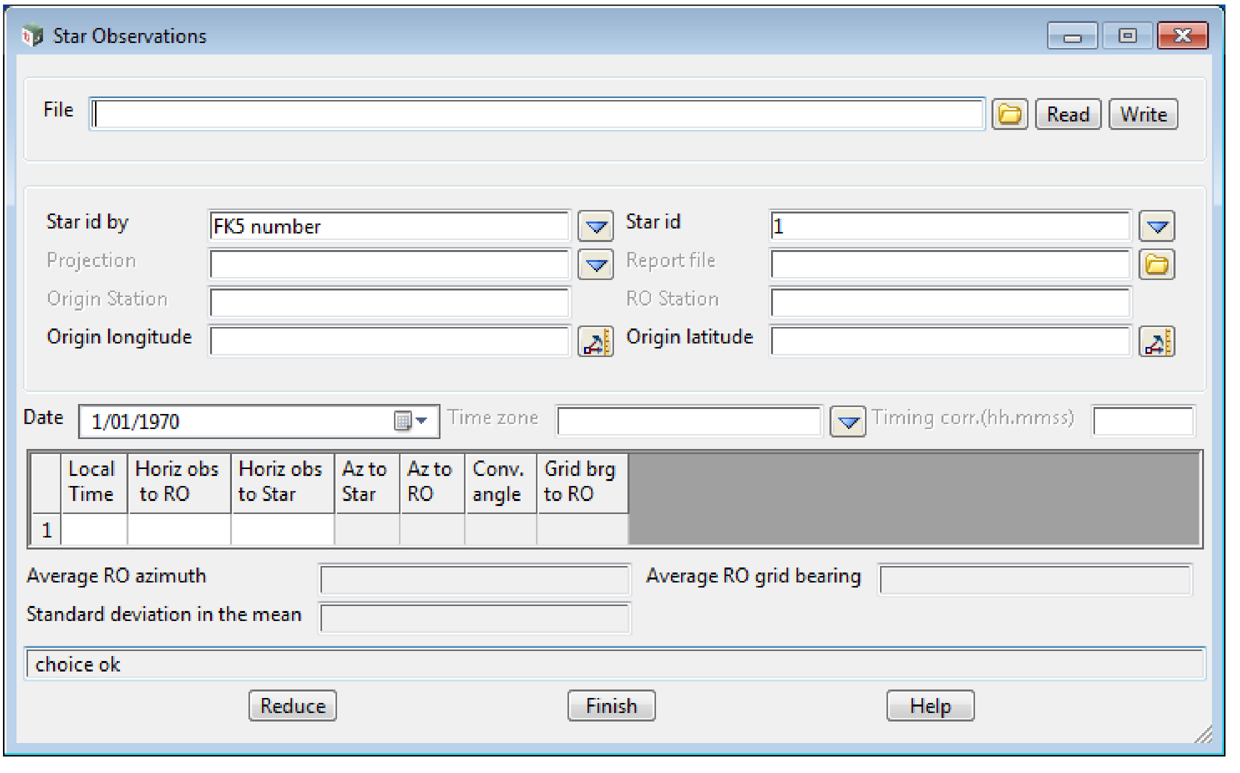
this is a statistical measure of the reduced observations.

17.11.14 Star Reduction

Position of option on menu: **Survey =>Geodetics => Star reduction**

The **Star reduction** option allows the reduction of observations to the nominated star by the hour angle method. A number of observations can be processed together. **The option does not require almanac details as these are processed internally using algorithms supplied by the US naval observatory.** These algorithms will produce accuracy in the order of 2 arc seconds in azimuth for the period AD1800 - 2050

On selecting the **Star reduction** option, the **Star Observations** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
File	file box		*.ast files

a filename can be specified for reading or writing a file.

Read button

if a valid file exists, the file contents can be loaded into the panel.

Write button

if a valid name is specified and the valid values are entered into all of the fields, the user can write the input data to a file.

Star id by	button	FK5 number	General name Constellation Bayer/ Flamsteed number FK5 number SAO number

the star can be identified by a number of different means. The stars available are based on the Fifth Fundamental Catalogue(FK5) catalogue. Cross matching of id's are not possible in some instances. e.g A star may have no known general name.

The selection of an id method will force the changing of the id list so the particular star can be selected.

Star id	choice	1st star id in list	available id's for that type

the star id can be selected from the list.

Report file	input		*.rpt

name of the file to report to.

Origin station input

name of the origin station

RO station input

name of the reference object station

Origin longitude input

the observation station's longitude should be specified in DMS format.

Origin latitude input

the observation station's latitude should be specified in DMS format.

Date date box current system date

the date of the observations as at the observers position should be entered. This can be by direct entry into the date field or by selecting the day, month, year from a date pop-up panel. The user can scroll through different months by selecting the arrow buttons on the pop-up date panel.

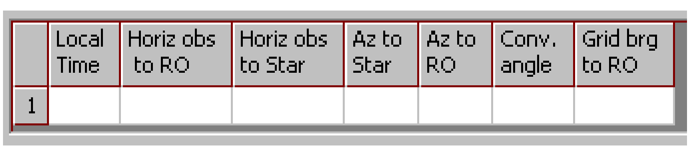
Time zone time zone box major time zones

a valid time zone for the observers position and local time should be selected from the pop-up list.

Timing corr.(hh.mmss) input

the correction to apply for any timing error including stopwatch corrections should be entered. The value should be in a hours, minutes seconds in hp format.

GRID VALUES: input



Local time input

the local time of the observation specified in 24 hour time in HMS. eg. 18.121002

Horiz obs to RO input

the horizontal angle (bearing) observed to the RO for the set of measurements. A set in this case includes a pointing to the RO and star (at a certain time) in the same face.

If observation s are made on both faces, a separate line should be entered for each face. e.g Face 1 to RO may be 0.0000 and the face 1 to the star may be 50.0000 which would be on the same line, then the observer may change face to face 2 and observe to the star again followed by the RO. The face 2 observations should appear on the next line eg 180.0004 230.0110.

Horiz obs to Star input

the horizontal angle (bearing) observed to the Star for the set of measurements.

Az to Star output

the true azimuth to the sun will be displayed in this field on successful reduction of the observations.

Az to RO output

the true azimuth to the RO will be displayed in this field on successful reduction of the observations.

Conv. angle output

the convergence angle at the observation point will be displayed in this field if a valid projection is specified and there is a successful reduction of the observations.

Grid brg to RO output

the grid bearing to the RO will be displayed in this field if a valid convergence angle is calculated and there is a successful reduction of the observations. The grid bearing is calculated by:

Grid brg = True azimuth + convergence

The fields and buttons used in this panel below the grid have the following functions.

Field Description	Type	Defaults	Pop-Up

Average RO azimuth output

if there is as successful reduction, the average RO azimuth value will be reported in this field.

Average RO grid bearing output

if there is as successful reduction, and the projection given, the average RO grid bearing value will be reported in this field.

Standard deviation in the mean output

this is a statistical measure of the reduced observations.

17.12 Traverse Spreadsheet

Position of menu: **Survey =>Traverse spreadsheet**

The **Traverse Spreadsheet** options allow the entry of traverse and radiation blocks from the keyboard, selection of strings from the current view or by reading from a file. Adjustments can then be made including transit, Bowditch, compass or least square adjustment. The adjustment can be for a closed string (loop) or an open string (non-loop). Traverse Spreadsheet will often be abbreviated as TSS.

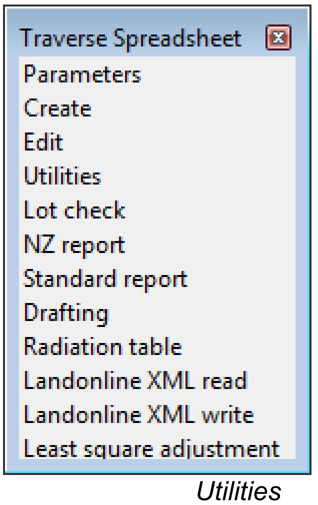
There are two types of traverse spreadsheets (TSS) - Plane and Projection.

For the Plane TSS, entered distances are multiplied by the scale factor to give final distances and hence calculate co-ordinates, whereas for the Projection TSS, ellipsoid distances are entered and ***12d Model*** calculates the correct co-ordinate position using full projection calculations.

The **Traverse spreadsheet** walk-right menu contains the traverse spreadsheet options.

.

For the option *Parameters, go to* 17.12.1 TSS Parameters
Create 17.12.2 TSS Create
Edit 17.12.3 TSS Edit



Setting parameters for the traverse spreadsheet option
Create a TSS - Plane or Projection
Edit an existing TSS
Utilities for use with existing TSSs
Run lot checks on typed bearing/distance data
NZ traverse spreadsheet report
Standard traverse spreadsheet report
Traverse spreadsheet drafting
Creating radiation tables from a TSS
Reading LINZ XML data
Writing LINZ XML data
Least Square adjustment

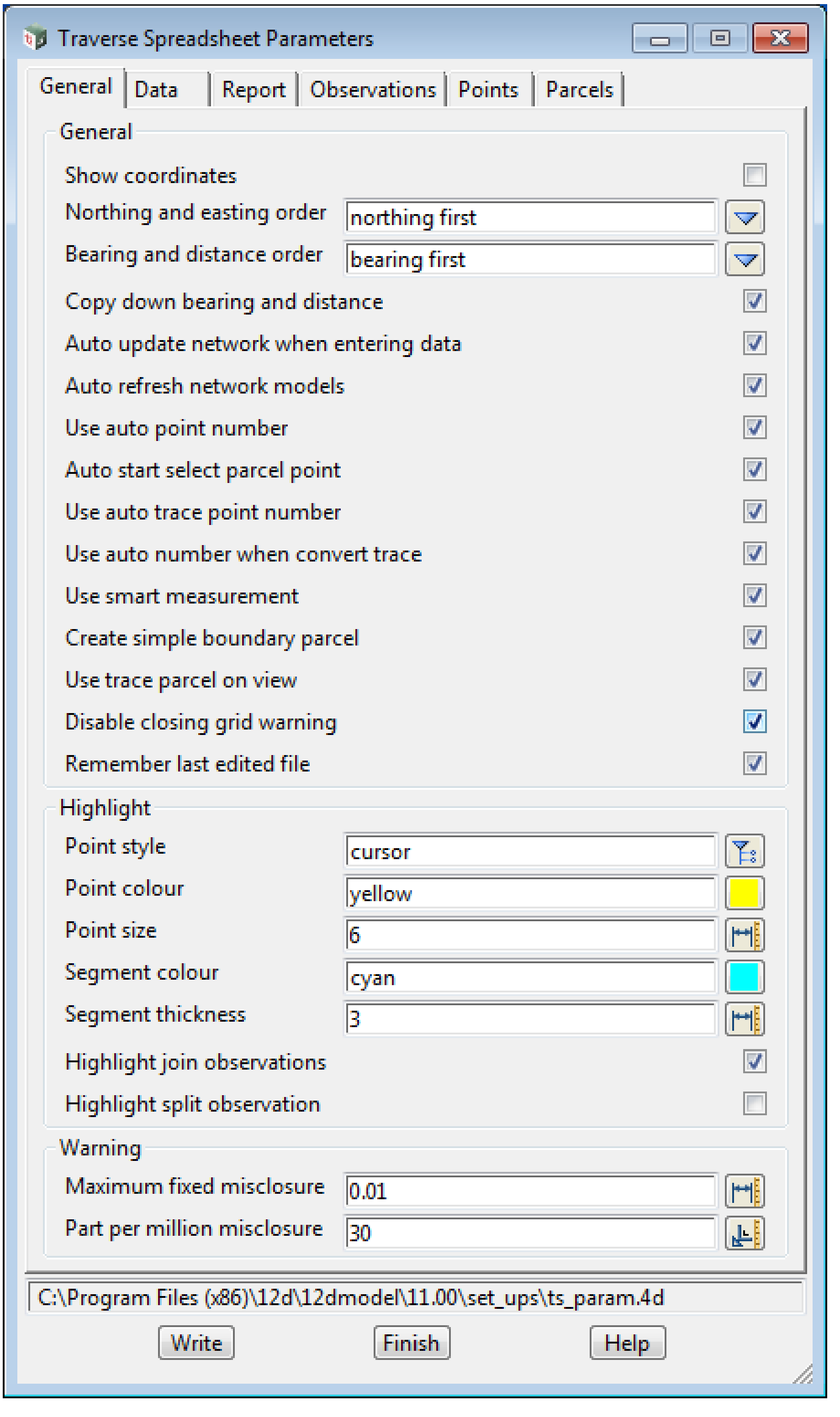
Utilities	17.12.4 TSS Utilities
Lot check	17.12.5 Lot Check
NZ report	17.12.6 NZ TSS Report
Standard report	17.12.7 Standard TSS Report
Drafting	17.12.8 TSS Drafting
Radiation table	17.12.9 TSS Radiation Table Drafting
Landonline XML read	17.12.10 Landonline XML Read
Landonline XML write	17.12.11 Landonline XML Write
Least square adjustment	

17.12.1 TSS Parameters

Position of option on menu: **Survey =>Parameters**

The **Traverse Spreadsheet Parameters** sets the parameters for defining the *look and feel* of the TSS create and edit panels, reports and drafting.

Selecting **Parameters** brings up the **Traverse Spreadsheet Parameters** panel:



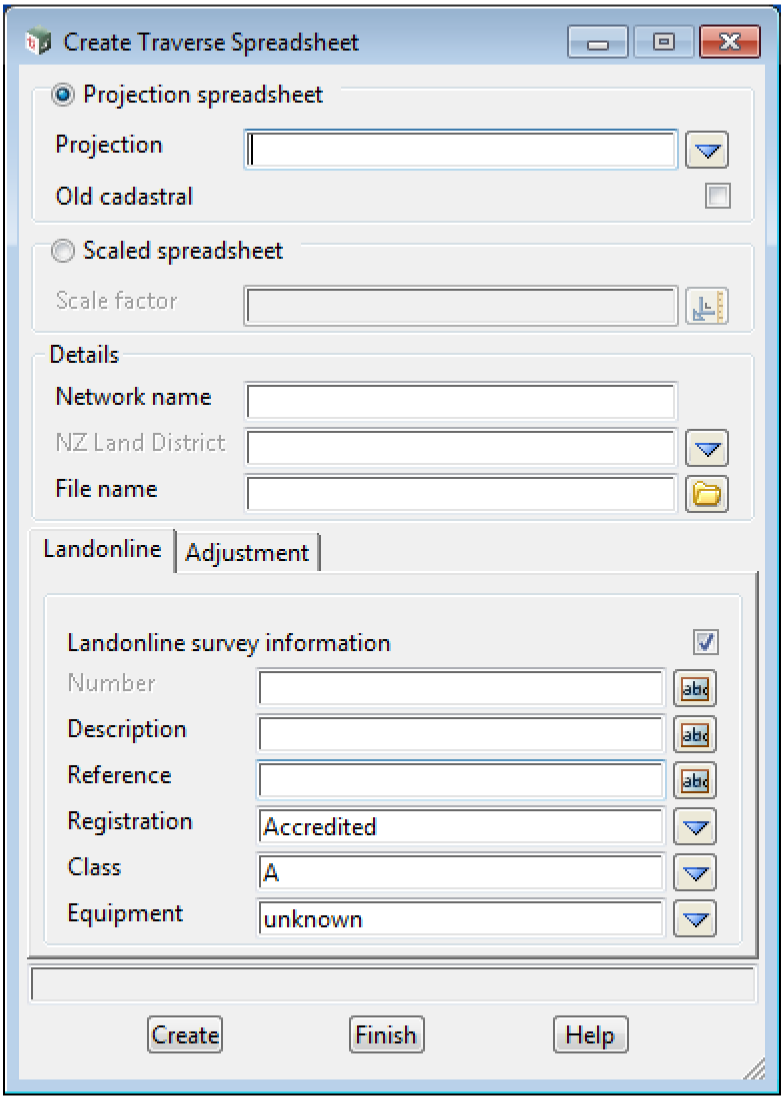
17.12.2 TSS Create

Position of option on menu: **Survey =>Traverse spreadsheet =>Create**

A traverse spreadsheet (TSS) can either use a constant scale factor or a cartographic projection to define the relationship between bearings/distances and the (x,y) co-ordinates.

The **Traverse Spreadsheet Create** option allows the entry of traverse and radiation blocks from the keyboard, selection of strings from the current view or by reading from a file. Adjustments can then be made including transit, Bowditch, compass or least square adjustment. The adjustment can be for a closed string (loop) or an open string (non-loop).

Selecting Traverse Spreadsheet **Create** brings up the **Create Traverse Spreadsheet** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Projection spreadsheet	radio button		

if selected then the **Projection** *field must be filled in.*

Projection	projection box	first projection	project projections

the cartographic projection used in the TSS. For more information on projections go to 7.6.6 Projections *. This is a Projection TSS.*

Scaled spreadsheet	radio button		

if selected then the **Scale factor** *field must be filled in.*

Scale factor	real box		

the constant scale factor used in the TSS. This is a Plane TSS.

Details section

Spreadsheet name text box

the name for the TSS

File name text box

*name of the *.tf file to store all the TSS data in. The TSS can be brought back for editing by giving this filename.*

Create button

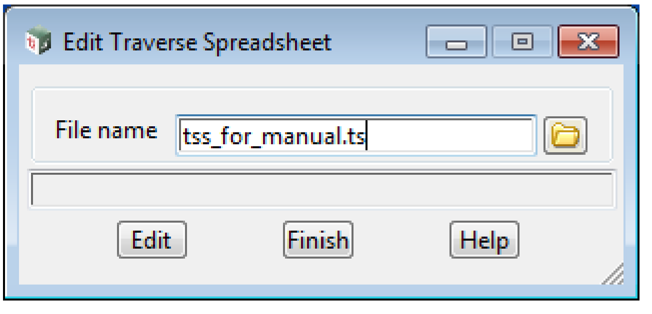
create a new traverse spreadsheet using the parameters entered in the above panel field. The appropriate editor (either for a **Projection Traverse Spreadsheet** *or a* **Plane Traverse Spreadsheet** *panel) is then brought up for the TSS data to be entered and/or edited.*

17.12.3 TSS Edit

Position of option on menu: **Survey =>Traverse spreadsheet =>Edit**

The traverse spreadsheet editor reads in an existing TSS file and loads the information into either the **Projection Traverse Spreadsheet** or the **Plane Traverse Spreadsheet** panel.

Selecting Traverse Spreadsheet **Edit** brings up the **Edit Traverse Spreadsheet** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
File name	file box		*.ts files

name of the file containing the traverse spreadsheet information to be edited.

Edit button

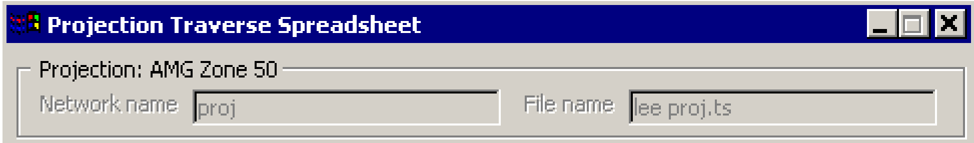
read the information in the traverse spreadsheet file and load it into either a **Projection Traverse Spreadsheet** *or a* **Plane Traverse Spreadsheet** *panel.*

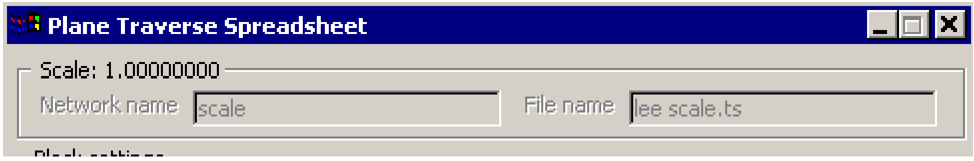
For entering/editing either a Projection TSS or Plane TT, please go to the section 17.12.3.1 Traverse Spreadsheets

17.12.3.1 Traverse Spreadsheets

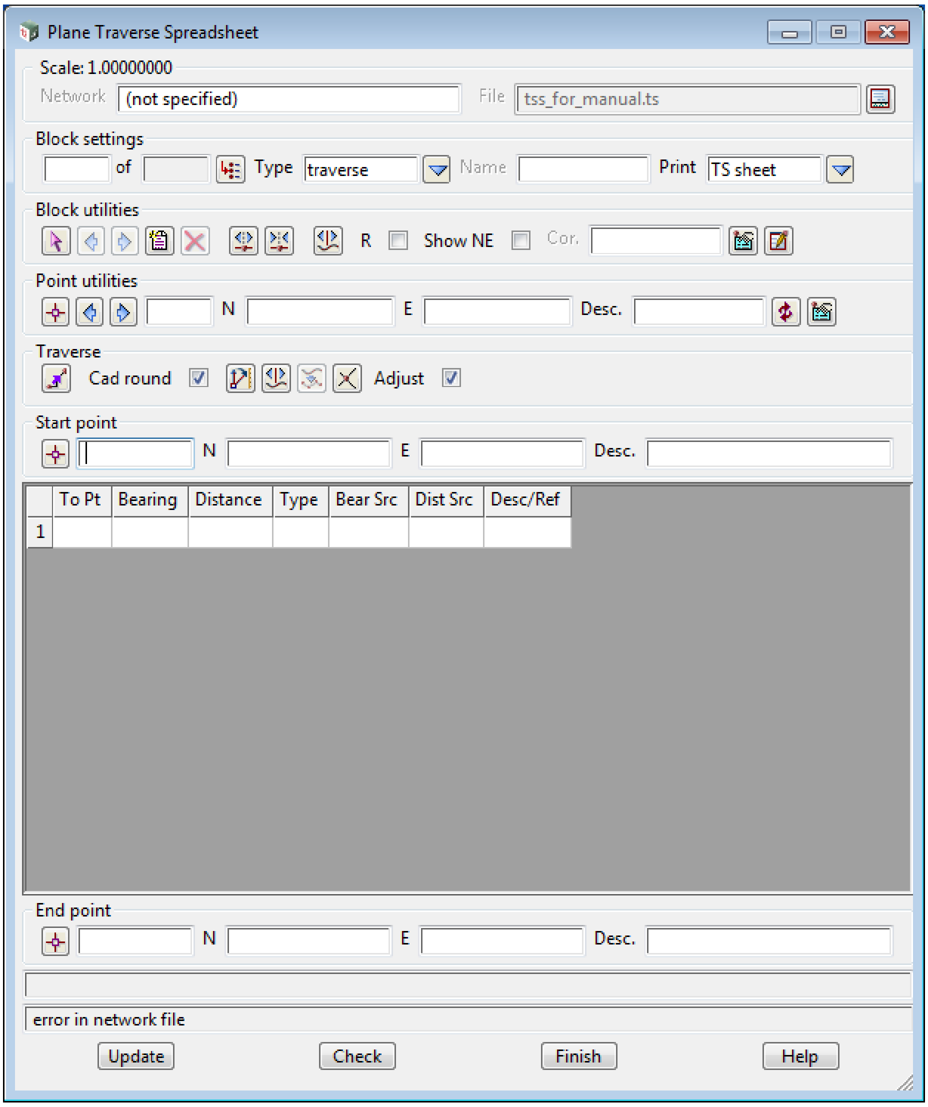
The **Projection** and **Plane TSS** panels allows the entry of traverse and radiation blocks from the keyboard, selection of strings from the current view or by reading from a file. Adjustments can then be made including transit, Bowditch, compass or least square adjustment. The adjustment can be for a closed string (loop) or an open string (non-loop).

The two panels are very similar and only the top section is different. The Projection Traverse Spreadsheet requires a **Projection** rather than a **Scale factor**.





For more information about terminology used in the projection option, see the Appendix 38 Geodetics Summary .



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Network name	display only		

the entire spreadsheet can be given a name for identification purposes. This was defined when the TSS was created and can only be changed using the TSS Utilities option.

File name	display only		

the file name containing the TSS information. This was defined when the TSS was created and can only be changed using the TSS Utilities option.

The only visible difference between the Plane and Projections Traverse Spreadsheet panel is that the Scale or Projection is given at the top of the panel. All the panel fields are the same.

Of course the calculations are different for the two types of spreadsheets. For the Plane TSS, entered distances are multiplied by the scale factor to give final distances and hence calculated co-ordinates, whereas for the Projection TSS, ellipsoid distances are entered and ***12d Model*** *calculates the correct co-ordinate position using full projection calculations.*

Block Settings

Position	input		

*the order of a block within the spreadsheet can be specified by nominating the block position. Numbering within the list starts at **1**. This parameter is used for inserting new blocks into the list. If a user nominates to insert in position 5 and there is only 2 blocks already defined, the block will be placed at the end of the list (i.e. position 3)*

Count	input		

the number of blocks within the spreadsheet.

Type	choice	traverse	traverse, radiation, radiation backsight, parcel, irregular trace, balance

the type of a block within the spreadsheet can be specified by nominating the block type.

*The **traverse** block is for the entry of loop and non-loop traverses starting and ending at points with known coordinate values.*

For a hanging traverse, i.e. a traverse that does not close onto a known point, the data can be entered by entering a 0 (zero) in the point id field of the grid control, as well as the end point - point id field.

*The **radiation** block consists of a number of radiations or **side shots** from a point with known coordinates. This point may be defined in a previous traverse block or it may be from a known point. A valid coordinate must exist for the radiation start point for the radiation block to be used.*

*A **radiation backsight** block is the same a s a radiation block except that the value set to the backsight is included. After adjustment of previous blocks, the coordinates of setup and backsight points often change. This will introduce swing errors into calculations if not accounted for. By entering the backsight value as set in the field, a comparison is made between what was set and the adjusted value. The difference is then applied to the radiations in the block to reflect the corrected angle measurements.*

XXXX - need 12d NZ documentation on TSS

Name	input		

a spreadsheet can consist of a number of different blocks. For each block, a name can be given which can help identify a particular traverse/radiation block within the spreadsheet.

Flag	input		

the adjustment mode of the current block. The mode is either adjusted or unadjust. All blocks are set to *unadjusted on reading of a file, and the user specifies which blocks are to be adjusted.*

Prev button

allows the navigation to the previous block in the list.

Next button

allows the navigation to the next block in the list.

New button

allows for the definition and entry of a new block in the list.

Update button

allows for the updating of entries when changes have been made. For example, if a distance measurement were to be changed, pushing the ***update*** *button will ensure the new value will be used in further calculations and adjustments*

Insert button

allows for the insertion of a block into the list in the position defined by the **Block position** entry.

Append button

allows for the insertion of a block at the end of the list.

Delete button

allows for the deletion of the currently displayed block.

Show EN tick box ticked

if ticked the easting and northing columns will be shown in the grid. If unticked the columns will not be shown.

Copy b dsc tick box ticked

if ticked the bearing and description from the previous line in the grid will automatically be copied down to the next entered line.

Pick string button

allows for the loading of the current block by selecting a string from the current view.

Adjustment method choice bowditch, compass, transit, least square

method of adjustment.

Loop tick box unticked

For loop traverses the tick box should be checked. This assumes that the start and end points are the same.

The fields in this panel define the start and end point details. If the **loop** option is chosen the end point details are assumed to be the same as the start point In this case the end point group is disabled.

Least square tab

Angular std (sec).

*The standard deviation of a single angular measurement in seconds of arc***.**

Linear std (mm)

this value is defined by the standard deviation of a single distance measurement (e.g. 5 mm)

ppm

this value is defined by the edm scale error which is dependant on the length of line measured (e.g. 5ppm = 5mm error over a 1km distance)

Iterations input box 10

the calculation of the adjustment can be aborted if a solution is not found after the specified number of iterations.

Coords differ input box 0.0001

the calculation of the adjustment can stopped when the difference between successive calculations meet the defined tolerance.

variance increase tick box ticked

the calculation of the adjustment can stopped when the variance between successive calculations increases

The fields in start point group specify the start point parameters

Start point button/input

the start point can be selected by selecting the button and then selection and acceptance of a point in an open view. If valid a point id, easting, northing and description of the selected point will be copied into the appropriate fields.

If the start point is to be entered manually, the point id should be entered into the field adjacent to the start point button.

N input

the start point's northing value.

E input

the start point's easting value.

Desc input

the start point's description.

the grid control values can be entered using valid inputs into the various fields.

***To Point**. The point id of the next traverse station.*
***Bearing (dms)**. The bearing of the traverse line.*
***Distance**. The distance of the traverse line.*
***Point Desc.** The description of the traverse point*
***Easting** The easting coordinate of the traverse point. **NB.** This field is only displayed when the calc or adjust buttons are selected.*
***Northing** The northing coordinate of the traverse point. **NB.** This field is only displayed when the calc or adjust buttons are selected.*

The fields in end point group specify the end point parameters

end point button/input

the end point can be selected by selecting the button and then selection and acceptance of a point in an open view. If valid a point id, easting, northing and description of the selected point will be copied into the appropriate fields.

If the end point is to be entered manually, the point id should be entered into the field adjacent to the end point button.

N input

the end point's northing value.

E input

the end point's easting value.

Desc input

the end point's description.

the misclose line near the bottom of the panel have the following functions.

Bearing input box

the closing bearing for the traverses.

Distance input box

the closing distance of the traverse

Delta east input box

the closing vector's delta east component

Delta east input box

the closing vector's delta east component

Misclosure input box

the ratio of the closing distance to the total traverse distance.

The buttons at the bottom of the panel have the following function.

Calc button

the calc button is used to calculate coordinate values based on the information input into the grid control and the points tab. The block has to be inserted or appended into the list ***before*** *this can be done.*

Adjust button

the adjustment of the current block can be done after the calc stage has been reached. This will use the method of adjustment as stipulated by the user.

Adjust All button

the adjustment of all of the blocks in the list can be done after the calc stage has been reached in each of the individual blocks. This will use the method of adjustment as stipulated by the user.

Save button

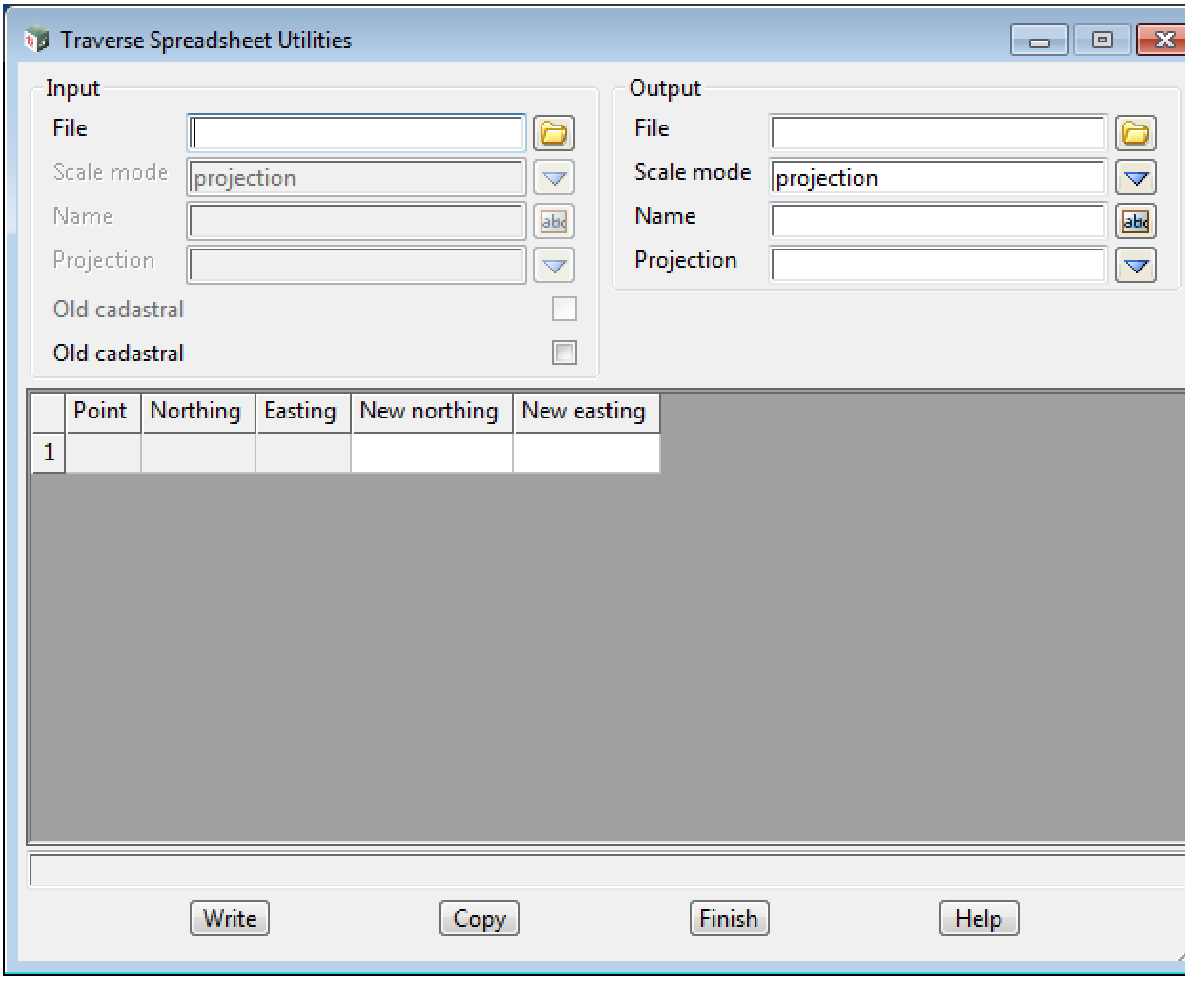
saves the current data in the panel to the TSS file (displayed in the ***file name*** *box).*

17.12.4 TSS Utilities

Position of option on menu: **Survey =>Traverse spreadsheet =>Utilities**

The traverse spreadsheet utilities are used to change the file containing a TSS, change the name (TSS), change between Plane (constant scale) or Projection type,

Selecting Traverse Spreadsheet Utilities brings up the **Traverse Spreadsheet Utilities** panel:

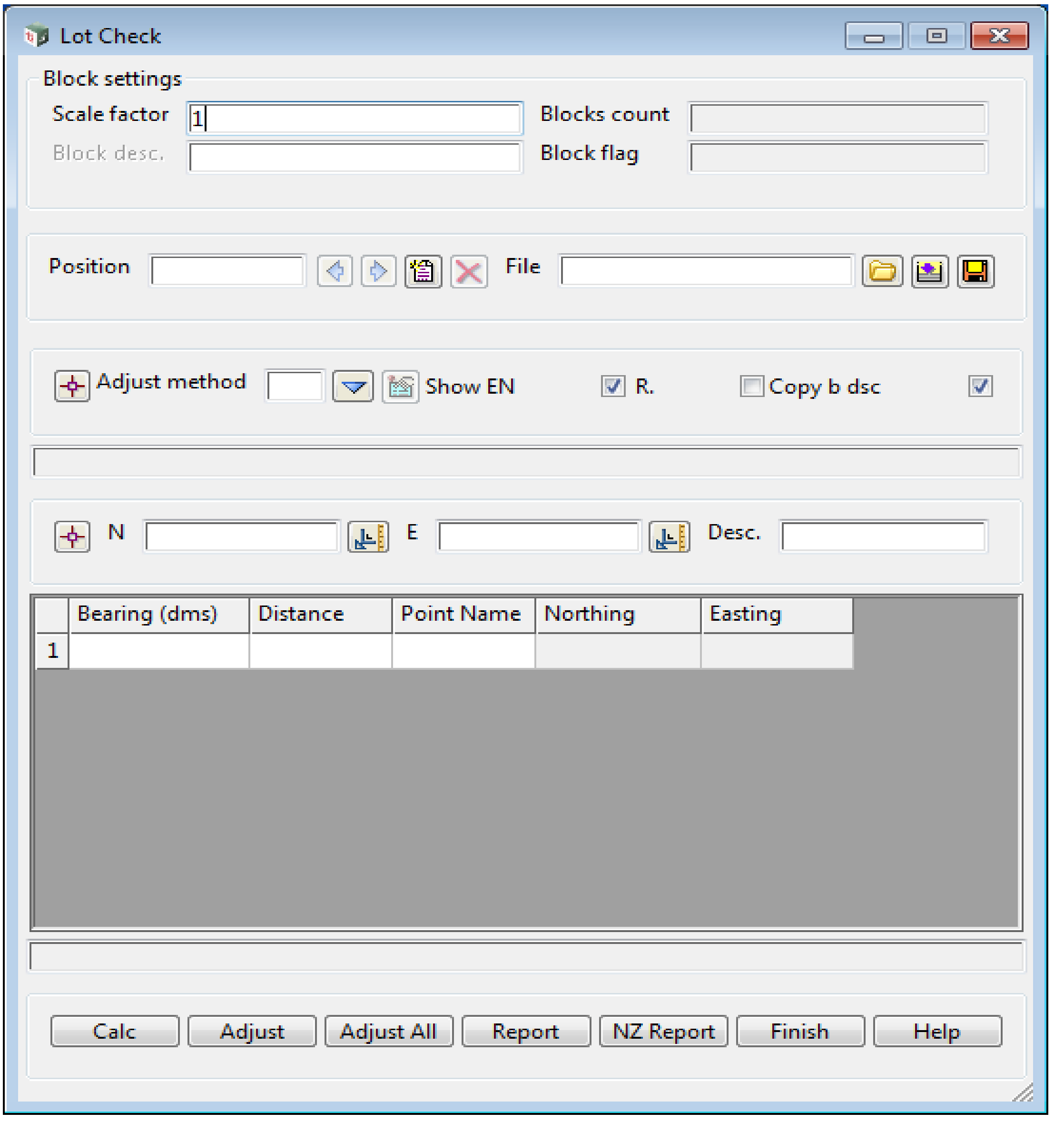


17.12.5 Lot Check

Position of option on menu: **Survey =>Traverse spreadsheet =>Lot check**

The **Lot check** option allows the entry of traverse and radiation blocks from the keyboard, selection of strings from the current view or by reading from a file. Adjustments can then be made including transit, Bowditch, compass or least square adjustment. This option can be used to enter existing plan dimensions to check area calculations as well as ensuring lots are closed to prescribed limits.

Selecting **Lot check** brings up the **Lot Check** panel



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

The scale and datum fields are the only difference between the plane and datum lot check panels.

Scale factor input project scale factor

a scale factor can be defined which will be applied to any entered distances. Final distances used in calculations are derived by multiplying the entered distance by the scale factor.

Block count input

the number of blocks within the spreadsheet.

Desc. input

a spreadsheet can consist of a number of different blocks. For each block, a block description can be given which can help identify a particular traverse/radiation block within the spreadsheet.

Block flag input

the adjustment mode of the current block. The mode is either adjusted or unadjust. All blocks are set to unadjusted on reading of a file, and the user specifies which blocks are to be adjusted.

Prev button

allows the navigation to the previous block in the list.

Next button

allows the navigation to the next block in the list.

New button

allows for the definition and entry of a new block in the list.

Update button

*allows for the updating of entries when changes have been made. For example, if a distance measurement were to be changed, pushing the **update** button will ensure the new value will be used in further calculations and adjustments*

Insert button

allows for the insertion of a block into the list in the position defined by the **Block position** entry.

Append button

allows for the insertion of a block at the end of the list.

Delete button

allows for the deletion of the currently displayed block.

File name input box *.ts

a file name can be defined which will allow for the reading of an existing file and for defining a new file name for reading in the future.

Read button

*the read button allows for the loading of an existing file as specified by the **file name** box.*

Write button

*the write button allows for the saving of the current file as specified in the **file name** box.*

Pick string button

allows for the loading of the current block by selecting a string from the current view.

Adjust method choice bowditch, compass, transit, least square, none

method of adjustment.

Show EN tick box ticked

if ticked the easting and northing columns will be shown in the grid. If unticked the columns will not be shown.

Copy b dsc tick box ticked

if ticked the bearing and description from the previous line in the grid will automatically be copied down to the next entered line.

The fields in this panel define the start and end point details. If the **loop** option is chosen the end point details are assumed to be the same as the start point In this case the end point group is disabled.

Least square tab

The fields in this panel define the values to be used for least square adjustments.

Angular std (sec).

The standard deviation of a single angular measurement in seconds of arc.

Linear std (mm)

this value is defined by the standard deviation of a single distance measurement (e.g. 5 mm)

ppm

this value is defined by the edm scale error which is dependant on the length of line measured (e.g. 5ppm = 5mm error over a 1km distance)

Iterations input box 10

the calculation of the adjustment can be aborted if a solution is not found after the specified number of iterations.

Coords differ input box 0.0001

the calculation of the adjustment can stopped when the difference between successive calculations meet the defined tolerance.

variance increase tick box ticked

the calculation of the adjustment can stopped when the variance between successive calculations increases

The fields in start point group specify the start point parameters

Start point button/input

the start point can be selected by selecting the button and then selection and acceptance of a point in an open view. If valid a point id, easting, northing and description of the selected point will be copied into the appropriate fields.

If the start point is to be entered manually, the point id should be entered into the field adjacent to the start point button.

N input

the start point's northing value.

E input

the start point's easting value.

Desc input

the start point's description.

The grid control values can be entered using valid inputs into the various fields.

To Point. *The point id of the next traverse station.*
Bearing (dms). *The bearing of the traverse line.*
Distance. *The distance of the traverse line.*
Point Desc. *The description of the traverse point*
Easting *The easting coordinate of the traverse point.* ***NB.*** *This field is only displayed when the calc or adjust buttons are selected.*
Northing *The northing coordinate of the traverse point.* ***NB.*** *This field is only displayed when the calc or adjust buttons are selected.*

The fields in end point group specify the end point parameters

end point button/input

the end point can be selected by selecting the button and then selection and acceptance of a point in an open view. If valid a point id, easting, northing and description of the selected point will be copied into the appropriate fields.

If the end point is to be entered manually, the point id should be entered into the field adjacent to the end point button.

N input

the end point's northing value.

E input

the end point's easting value.

Desc input

the end point's description.

The misclose line near the bottom of the panel has the following functions:

Bearing input box

the closing bearing for the traverses.

Distance input box

the closing distance of the traverse

Delta east input box

the closing vector's delta east component

Delta east input box

the closing vector's delta east component

Misclosure input box

the ratio of the closing distance to the total traverse distance.

The buttons at the bottom of the panel have the following functions:

Calc button

*the calc button is used to calculate coordinate values based on the information input into the grid control and the points tab. The block has to be inserted or appended into the list **before** this can be done.*

Adjust button

the adjustment of the current block can be done after the calc stage has been reached. This will use the method of adjustment as stipulated by the user.

Adjust All button

the adjustment of all of the blocks in the list can be done after the calc stage has been reached in each of the individual blocks. This will use the method of adjustment as stipulated by the user.

Report button

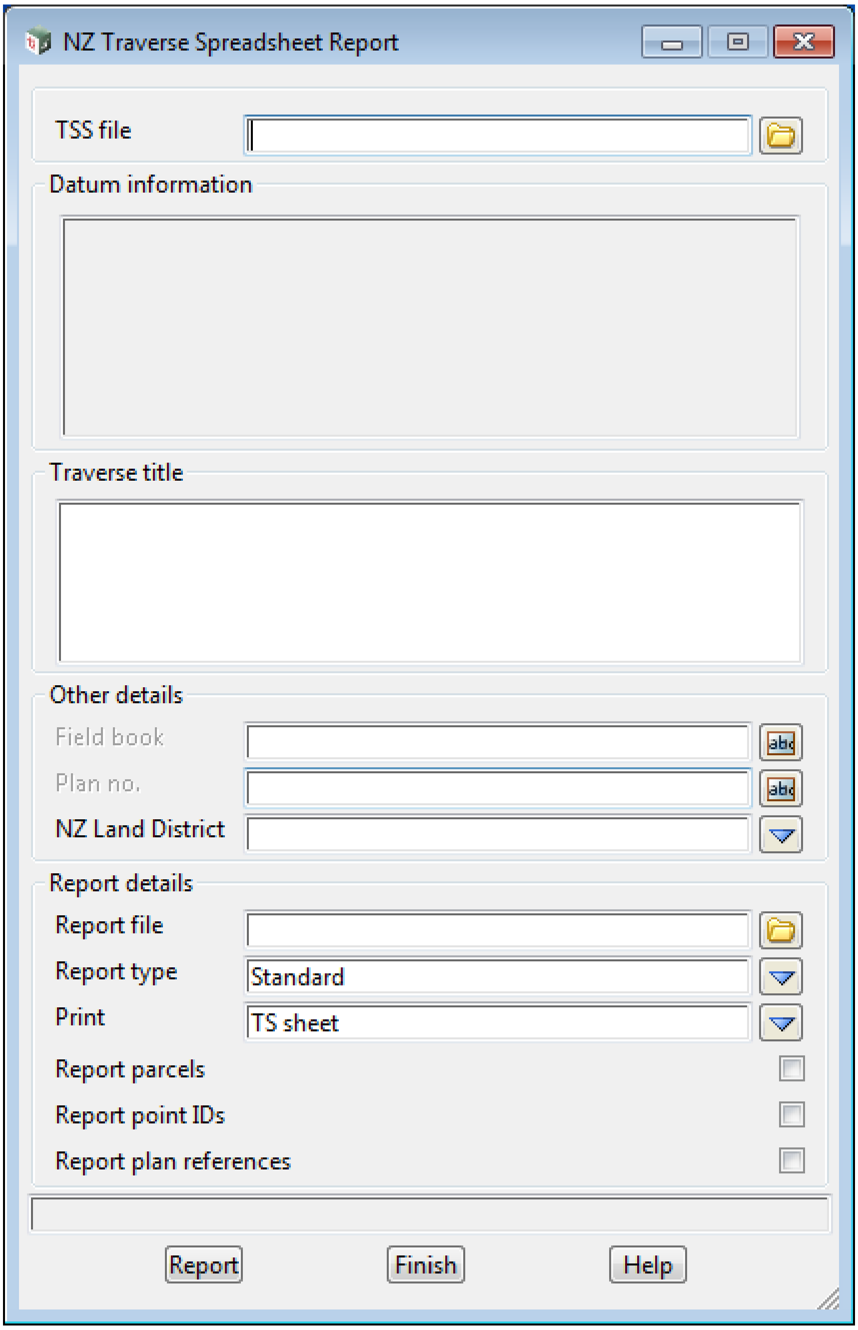
on selecting this button the NZ traverse spreadsheet report panel appears.

17.12.6 NZ TSS Report

Position of option on menu: **Survey =>Traverse spreadsheet =>NZ report**

The **NZ report** option allows the user to define the report header etc. so as to produce a spreadsheet report which can be printed to a pre-printed NZ sheet.

Selecting **NZ report** brings up the **NZ Traverse Spreadsheet Report** panel



The fields and buttons used in this panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Input format | choice box | *.ts | *.ts, *.lch

either a TSS file (.ts) or a lot check file (*.lch) can be reported.*

Input file | input box | | *.ts

name of the TSS or lot check file.

Print | choice box | TS sheet | TS sheet, Calc sheet, Other

either TS sheet.

Report parcels | tick box

If tick, parcel block types are included in the report.

Traverse Title section

Traverses of, line 2, line 3, line 4

*The title of the traverses can be entered into the **traverses of** box as well as the following **line 2**, **line 3** and **line 4** input boxes. Due to the limitation of space on the pre-printed forms supplied by LINZ the number of characters should be limited to 20 characters (inclusive of space characters) for the **traverse of** line and 33 characters (inclusive of space characters) for **lines 2** to **4**.*

*The input into the **traverses for** line is compulsory but for **lines 2** to **4** it is optional.*

Datum Information section

NZ Circuit choice box existing NZ circuits

the circuit in which the current spreadsheet file relates to should be specified.

Projection choice box Geodetic 1949
Geodetic 2000
Old Cadastral

the appropriate NZ geodetic datum should be selected.

Coords in terms of section

these parameters are only required to be entered for the Old cadastral datum case. For other datums, the values are automatically updated.

Point name input box

the name of the origin point should be entered.

Northing input box

the nothing coordinate of the origin point should be entered.

Easting input box

the easting coordinate of the origin point should be entered.

Other Details section

Field book input box

the name of the field book used.

Plan no input box

the plan no to be created if known.

NZ Land District choice box Existing NZ land districts

the appropriate NZ land district should be selected.

Report file files box *.rpt files

file name for the report.

Report file button

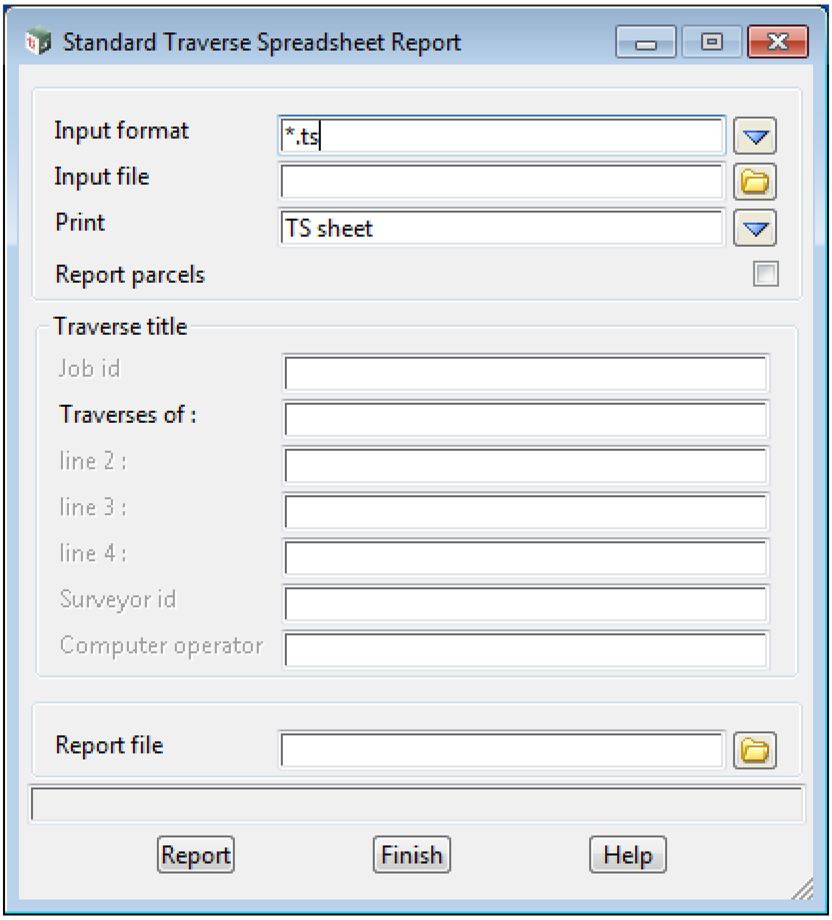
a report using the nominated file name is created.

17.12.7 Standard TSS Report

Position of option on menu: **Survey =>Traverse spreadsheet =>Standard report**

The **Standard report** allows the user to define the report header etc. so as to produce a standard TSS report which can be printed, saved or edited.

Selecting **Standard report** brings up the **Standard Traverse Spreadsheet Report** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Input format	choice box	*.ts	*.ts, *.lch

either a TSS file (.ts) or a lot check file (*.lch) can be reported.*

Input file	input box		*.ts

name of the TSS or lot check file.

Print	choice box	TS sheet	TS sheet, Calc sheet, Other

either TS sheet.

Report parcels	tick box		

If ticked, parcel block types are included in the report.

Traverse Title section

Job id text input box

text to use as the Job id.

Traverses of, line 2, line 3, line 4

*The title of the traverses can be entered into the **traverses of** box as well as the following **line 2**, **line 3** and **line 4** input boxes. Due to the limitation of space on the pre-printed forms supplied by LINZ the number of characters should be limited to 20 characters (inclusive of space characters) for the **traverse of** line and 33 characters (inclusive of space characters) for **lines 2** to **4**.*

*The input into the **traverses for** line is compulsory but for **lines 2** to **4** it is optional.*

Surveyor id text input box

text to use as the Surveyor id.

Computer operator text input box

text to use as the Computer operator.

Report file files box *.rpt files

file name for the report.

Report file button

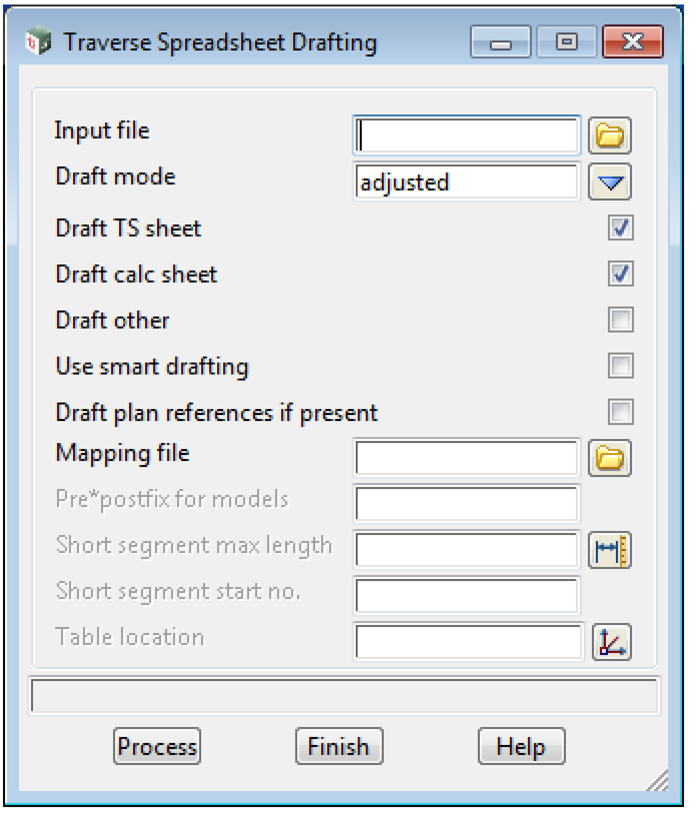
a report using the nominated file name is created.

17.12.8 TSS Drafting

Position of option on menu: **Survey => Traverse spreadsheet=> Drafting**

The Traverse spreadsheet **drafting** produce models, linestyles and colours for data within the TSS.

Selecting **Drafting** brings up the **Traverse Spreadsheet Drafting** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Input file	file box		*.ts

name of the existing spreadsheet file to create drafting for.

Draft mode	choice box	adjusted	adjusted initial

the draft mode will determine if adjusted strings are used for drafting or if the unadjusted values (initial) values are used

Draft TS sheet tick box

if ticked,

Draft calc sheet tick box

if ticked,

Draft other tick box

if ticked,

Map file Map File box *.mapfile, *.mf files

Map File to define the drafting

Pre*postfix for models text input box

pre and post text to add to the model names given in the Map File

Short segment max length real input box

maximum size for a segment to be considered a short segment. Short segments are labelled with a number and the details placed in a table.

Short segment start no. integer input box

start number for any short segments.

Table location xyz box

world position for the short segment table.

Process button

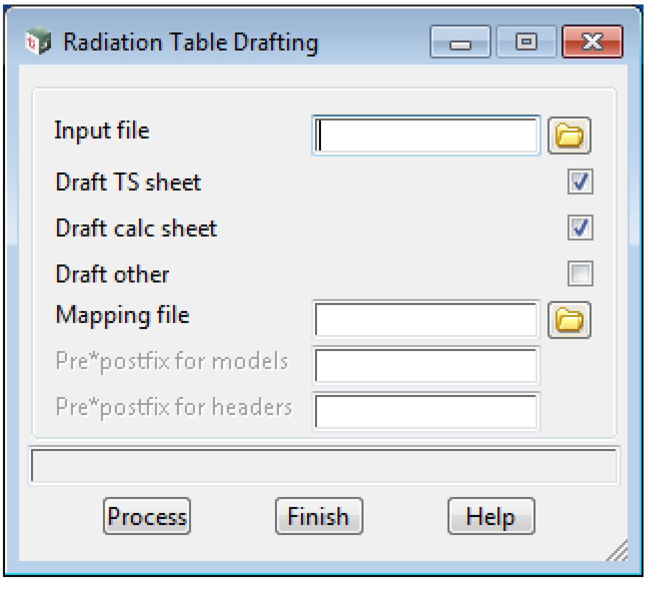
process the traverse spreadsheet and create the drafting and short segment table.

17.12.9 TSS Radiation Table Drafting

Position of option on menu: **Survey => Traverse spreadsheet=> Radiation table**

The Traverse spreadsheet **radiation table** option produces radiation tables for the radiation data in a traverse spreadsheet.

Selecting **Radiation table** brings up the **Radiation Table Drafting** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Input file	file box		*.ts

name of the existing spreadsheet file to create radiation tables for.

Draft TS sheet tick box

if ticked,

Draft calc sheet tick box

if ticked,

Draft other tick box

if ticked,

Map File Map File box *.mapfile, *.mf files

Map File to define the drafting

Pre*postfix for models text input box

pre and post text to add to the model names given in the Map File

Pre*postfix for headers text input box

pre and post text to add to the header names for the radiation tables.

Process button

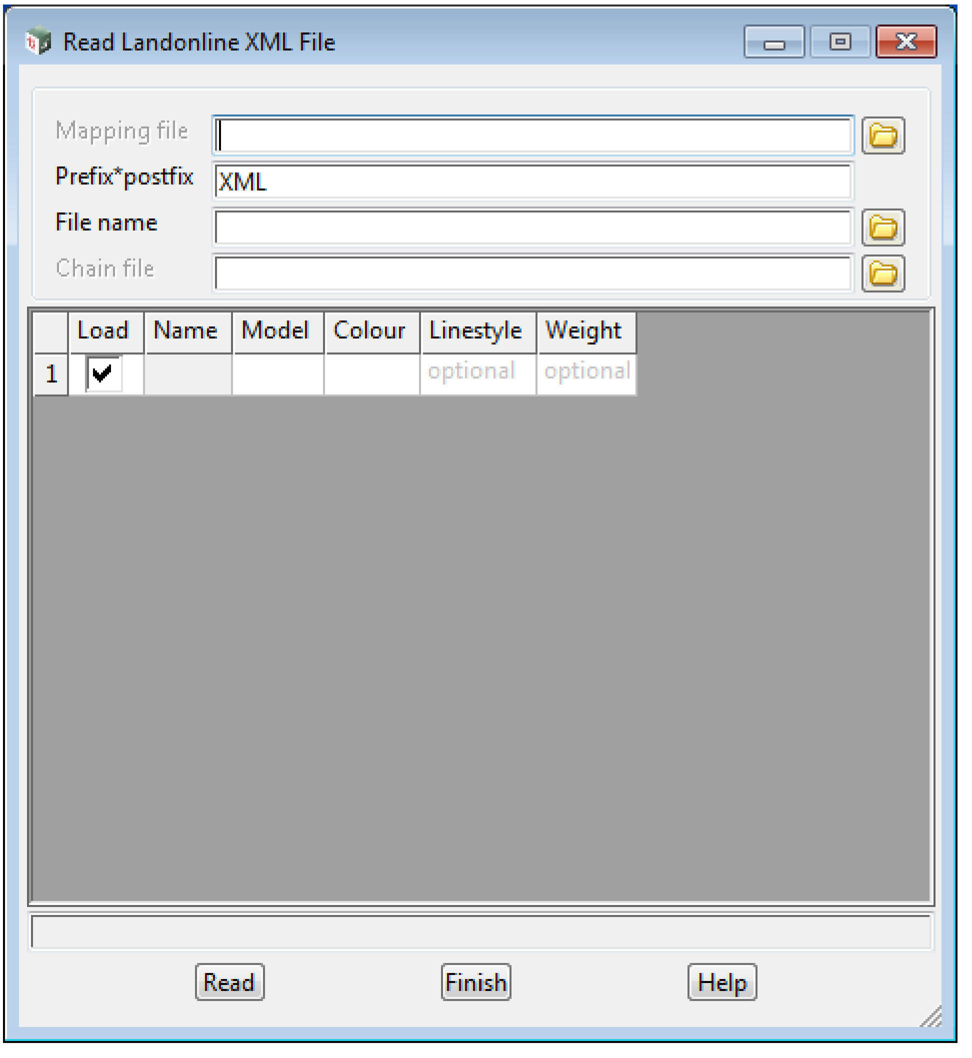
process the traverse spreadsheet and create the radiation tables.

17.12.10 Landonline XML Read

Position of menu: **Survey =>Traverse spreadsheet => Landonline XML read**

The **Landonline XML Read** option reads in data in the XML format for Landonline New Zealand.

Selecting **Landonline XML read** brings up the **Read Landonline XML File** panel



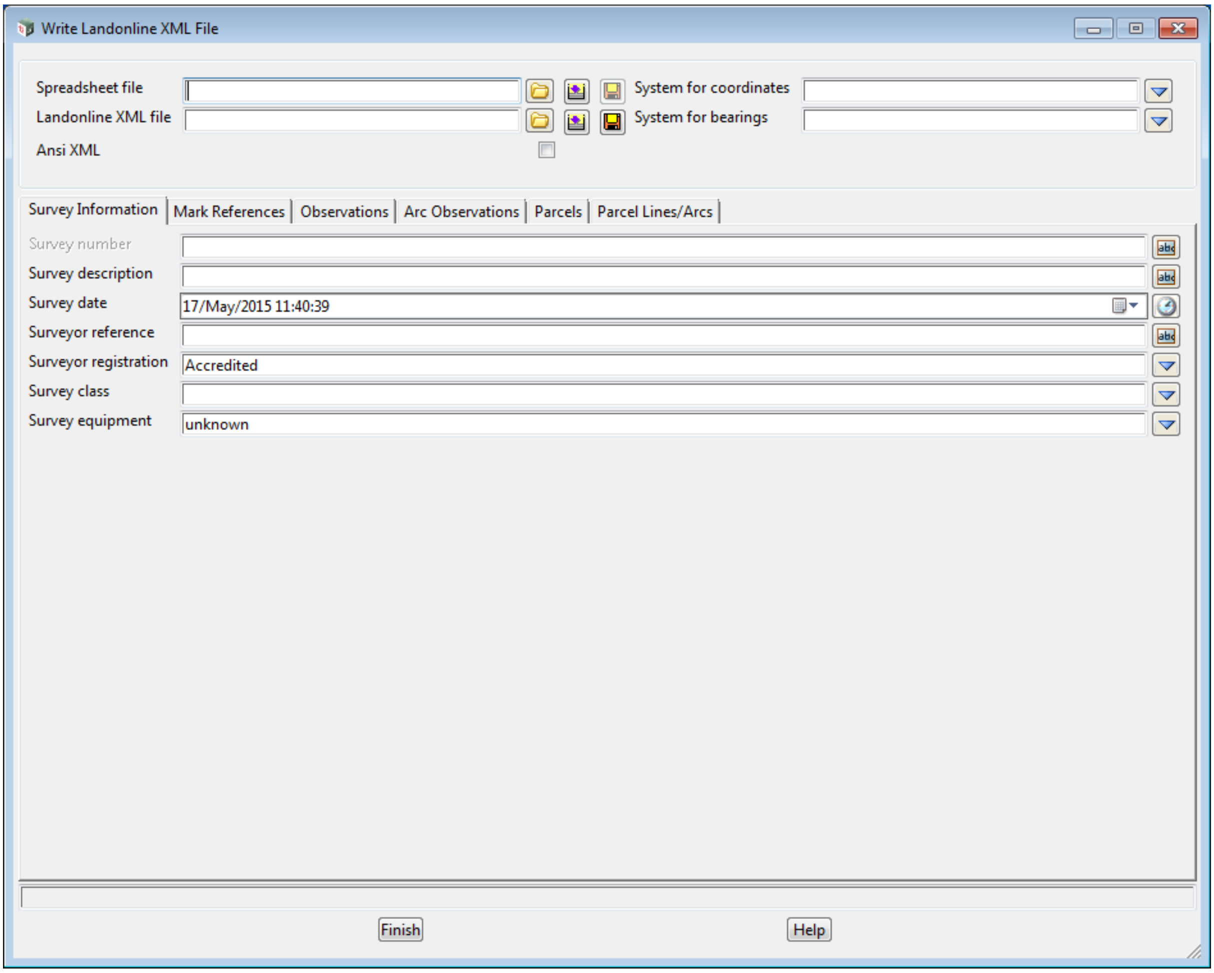
Documentation is available from *12d NZ Ltd*.

17.12.11 Landonline XML Write

Position of menu: **Survey =>Traverse spreadsheet => Landonline XML write**

The **Landonline XML Write** option writes out data in the XML format for Landonline New Zealand.

Selecting **Landonline XML write** brings up the **Write Landonline XML File** panel



Documentation is available from 12d NZ Ltd.

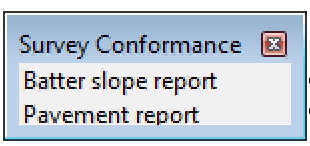
17.13 Conformance

Position of menu: **Survey =>Conformance**

The **Conformance** walk-right menu contains survey options for generating conformance reports and models on surveyed points representing as-built batter slopes and pavements, compared against string data representing designed batter slopes and pavements.

The **Conformance** walk-right menu is

.



Generate batter slope conformance report
Generate pavement conformance report

For the option *Batter slope report*, go to 17.13.1 Batter slope report
Pavement report 17.13.2 Pavement report

17.13.1 Batter slope report

Position of option on menu: Survey =>Conformance =>Batter slope report

The **Batter slope report** option generates a conformance report on surveyed points representing an as-built batter slope, compared against string data representing a designed batter slope.

In addition to the conformance report, an output results model of the surveyed points can be generated, grouping the points by colour into their conformance zones (i.e. *within tolerance*, *above tolerance*, *below tolerance*, and *not tested*). The points in the results model can have z-values and vertex text set to show various combinations of point level, point conformance, and point error.

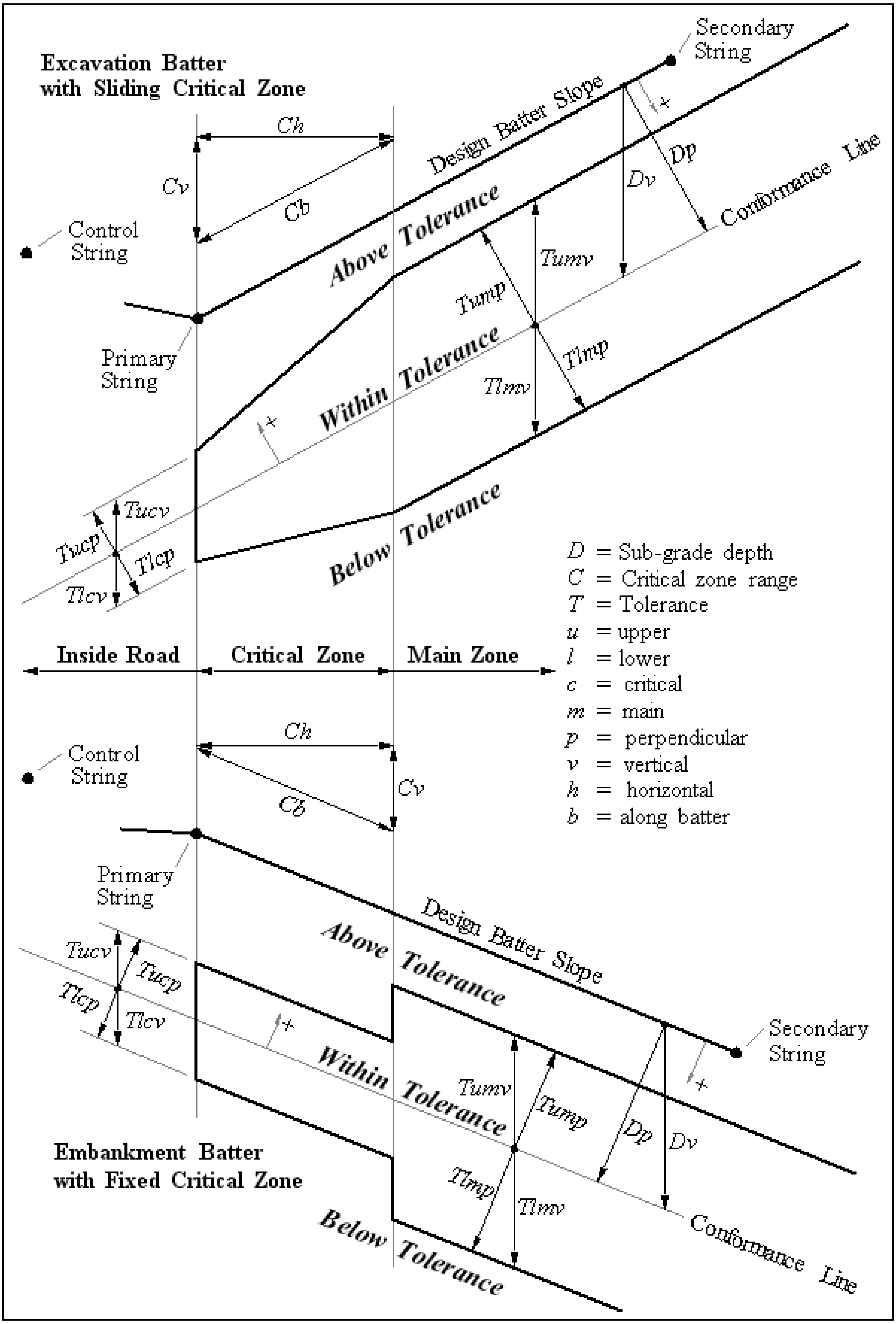
Note1: the term *point conformance* is used here to refer to a point's distance from design (distance from the conformance line on diagram below), while the term *point error* refers to a point's distance out of tolerance. As such, a point that is within tolerance (i.e. a *conformant point*) will, in general, have a non-zero conformance and a zero error, while a point that is out of tolerance (i.e. a *non-conformant point*) will have both a non-zero conformance and a non-zero error.

Note2: the point conformances and errors can be measured either vertically or perpendicular to the conformance line. It is perhaps worthwhile to note, especially in the case of the *sliding critical zone*, that the perpendicular measurements are determined from the vertical measurements, and **not** vice versa. That is, if the slope of the conformance line, measured from horizontal, is denoted by the angle *A*, then a surveyed point *Q*, has a perpendicular conformance *Qcp*, and error *Qep*, determined from the point's vertical conformance *Qcv*, and error *Qev*, via the following relationships:

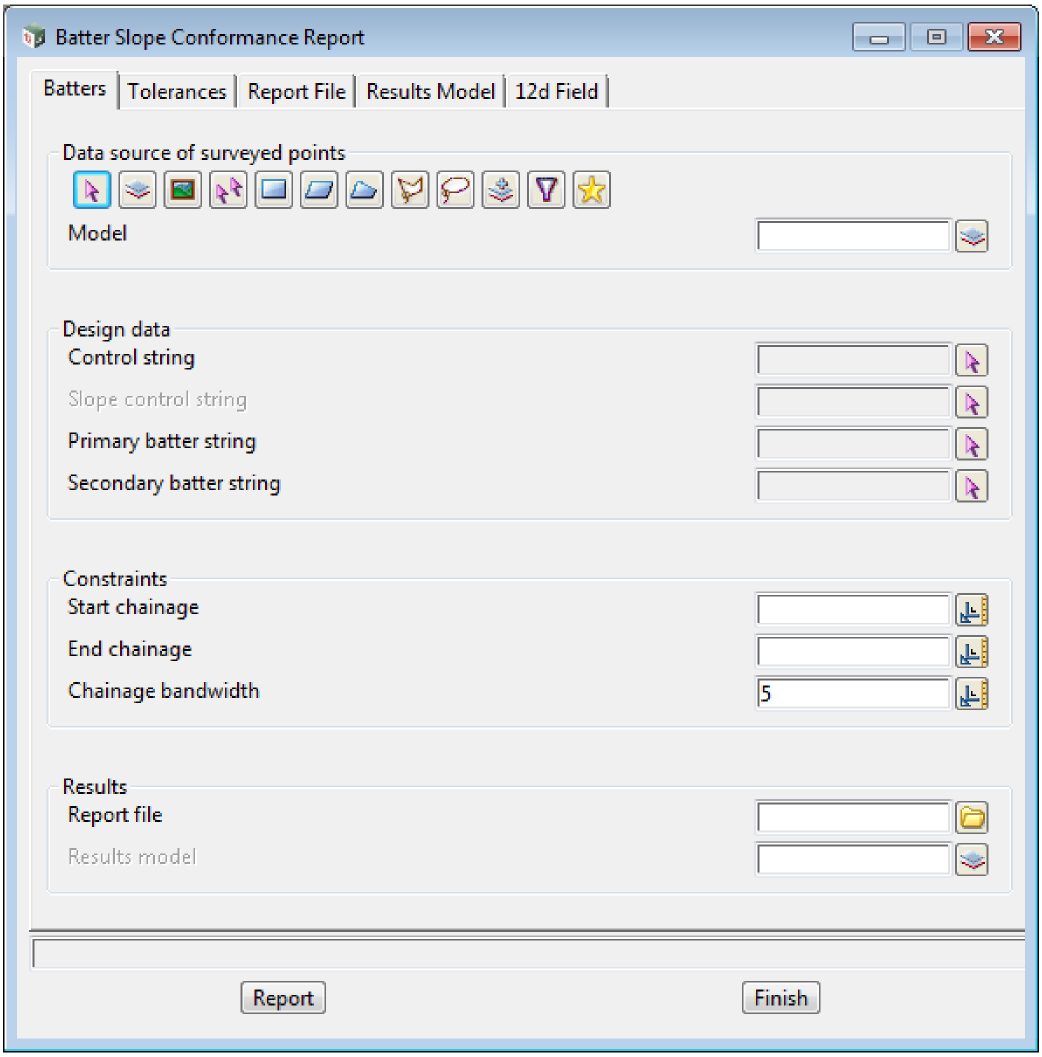
$$Qcp = Qcv.\cos A$$
$$Qep = Qev.\cos A$$

Note3: the conformance line is the line from which all point conformances and tolerances are measured, and is always parallel to the design batter slope line. When testing a sub-grade survey, the conformance line will normally be below the design batter slope line, but when testing a completed-construction survey, the two lines should normally coincide.

Typical sections through excavation and embankment batters are shown in the diagram below, along with schematics showing how the surveyed points are tested for conformance:



Selecting **Batter slope report** brings up the **Batter Slope Conformance Report** panel



Batters Tab:

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Data source of points source box

all points selected with this source box (whether they exist in point-string form or line-string form) will be considered as the set of surveyed points to be tested for conformance.

Note: *the data for the* Point Description *column in the report will be taken from the string names of the selected points. If* ***all*** *the string names are blank, however, the point description data in the report will show point ids representing the sequential order that the points are reported in. The results model (if generated) will always have the string names of each point set to match the reported point description.*

Design data

Control string string box

the string selected with this box should normally represent the road centreline, and will be used to determine the chainage and offset of each surveyed point

Slope control string string box

the Slope control string *is optional (and is not shown on the diagram above). If no string is selected, the* Primary string*, below, will be used as the* Slope control string*. The line that is formed in plan, from a* *surveyed point to the nearest point on the* Slope control string*, defines the vertical plane used to conform that surveyed point. The slope formed between the primary and secondary strings on this vertical plane, is the design slope for that surveyed point.*

Primary string string box

the string selected with this box should represent one edge of the designed batter (normally the edge closer to the road - i.e. the toe string of an excavation batter, or the crest string of an embankment batter).

The Primary string *represents the start of the* Critical Zone*, which continues horizontally in the direction of the* Secondary string*, for a distance specified in the* Critical zone range *field (on the* Tolerances Tab*).*

Secondary string string box

the string selected with this box should represent the other edge of the designed batter (normally the edge further away from the road - i.e. the interface string)

Constraints

Start chainage real box

the start chainage of the surveyed points to be conformed. Any point with a chainage less than the start chainage will not be tested. By default, the start chainage is set when the Control string *is selected, but a different value can be typed in.*

End chainage real box

the end chainage of the surveyed points to be conformed. Any point with a chainage greater than the end chainage will not be tested. By default, the end chainage is set when the Control string *is selected, but a different value can be typed in.*

Chainage bandwidth real box 5.0

if zero or blank, the surveyed points are simply sorted in ascending chainage order. However, if a value greater than zero is entered, the surveyed points will also be sub-sorted into chainage bands, in ascending offset order. Within each chainage band, the difference between the maximum and minimum chainage will be less than the Chainage bandwidth. *This is a useful feature if the surveyed points are set out in rows of roughly equivalent chainage.*

Results

Report file file box *.rpt

the name of the conformance report file to be created. If no extension is given, it will be given an extension of ".rpt".

Results model model box available models

the name of the results model to be created. If blank, the results model is not created.

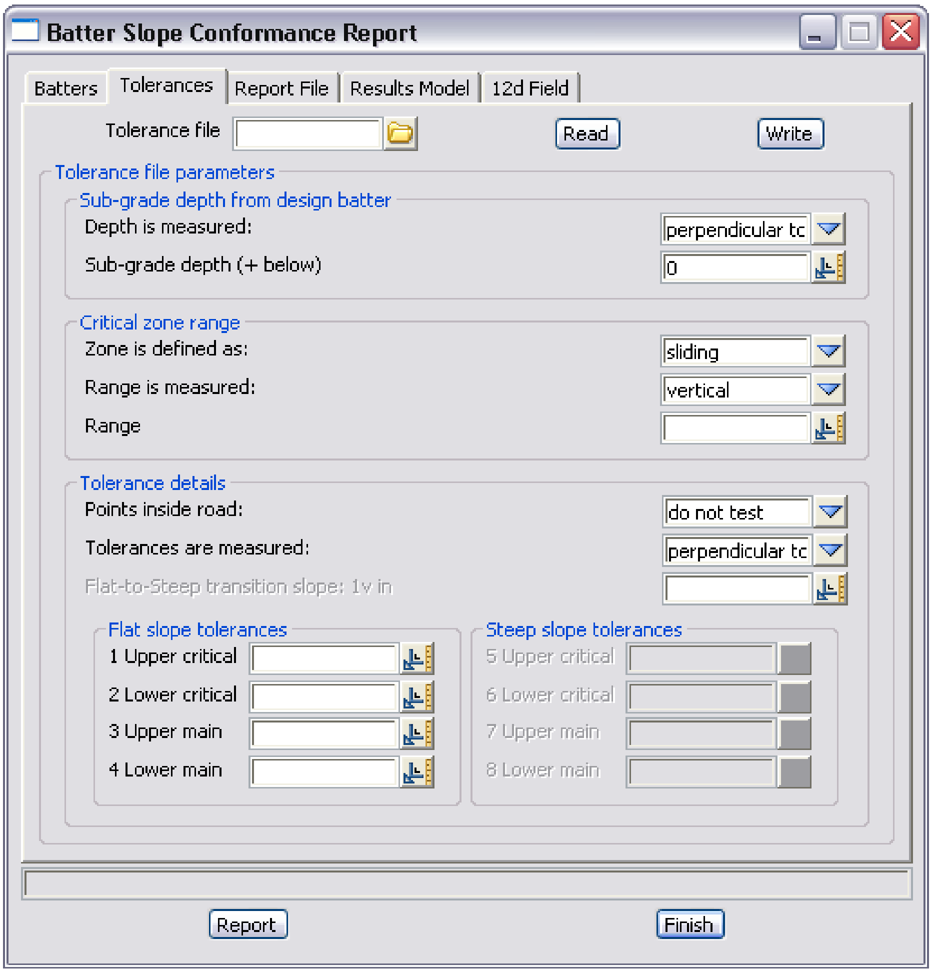
Report button

generates the conformance report file and the results model. This button can be activated regardless of which panel tab is currently active.

Finish button

exits the option and closes the panel. This button can be activated regardless of which panel tab is currently active.

Tolerances Tab:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tolerance file	file box		*.tol

the name of the tolerance file used to load and/or save the details of the conformance. If no extension is given, it will be given an extension of ".tol". *The tolerance file stores the details of every field on the* Tolerances Tab, *the* Report File Tab, *and the* Results Model Tab. *The tolerance file itself is not required to run a conformance test - it is merely provided as a convenience.*

Read button

Pressing this button will populate the panel fields on the last three panel tabs, with the data stored in the tolerance file. If data for a particular field is not found, that field remains unaffected.

Write button

Pressing this button will create a new tolerance file or replace an existing one. The contents of all non-blank fields on the last three panel tabs, are written to the file at this time.

Sub-grade depth from design batter

Field Description	Type	Defaults	Pop-Up
Depth is measured	choice box	perpendicular	perpendicular to batter vertical

whether the sub-grade depth is measured perpendicular to the batter or vertically (refer to parameters Dp and Dv on the diagram above)

Sub-grade depth real box 0.0

the depth of the conformance line from the design batter (refer to parameters Dp and Dv on the diagram above). If testing a sub-grade survey, this value will normally be greater than zero. If testing a completed-construction survey, however, this value will normally be zero, and the conformance line will coincide with the design batter slope line.

Critical zone range

Zone is defined as choice box sliding sliding
fixed

determines whether the Critical Zone is defined as sliding *or* fixed *(refer to diagram above). For excavation batters, the zone is normally* sliding*, whilst for embankment batters, it is normally* fixed*.*

Range is measured choice box vertical along batter
vertical
horizontal

determines how the size of the Critical Zone is measured (refer to parameters Cb, Cv, and Ch on the diagram above)

Range real box

the size of the Critical Zone, measured in the specified direction (refer to parameters Cb, Cv, and Ch on the diagram above). This distance must be zero or greater.

Tolerance details

Points inside road choice box do not test do not test
test with zero tolerance
test with fixed crit. tolerance

determines what to do with any surveyed points found inside the road. If such points are found, and this field is set to **do not test***, the points will be classified as untested, and in the results model, will have their vertex text set to* **INR***.*

Note: *untested points do not appear in the conformance report.*

Tolerances measured choice box perpendicular perpendicular to batter
vertical

*whether the conformance tolerances are measured perpendicular to the batter or vertically (refer to parameters T**p and T**v on the diagram above). This field also determines whether the reported point conformances and point errors are measured perpendiculary or vertically.*

Flat-to-Steep transition slope: 1v in
real box

It is possible to use two different sets of batter slope tolerances, when producing the conformance report. The Flat slope tolerance *set will be used whenever the design slope is equal to the* transition slope *or flatter, and the* Steep slope tolerance *set will be used whenever the design slope is steeper than the* transition slope. *The* transition slope *must be specified as a positive number, and will be interpreted as a slope in the form*
1[v] : transition slope[h]. *Leaving this field blank will disable the* Steep slope tolerance *set, ensuring that only the* Flat slope tolerance *set will ever be used.*

Flat slope tolerances

The following four tolerances are used if the *transition slope* is not specified, or whenever the design slope is equal to the *transition slope* or flatter.

1 Upper critical real box

refer to parameters Tucp and Tucv on the diagram above. This value is normally zero or greater.

2 Lower critical real box

refer to parameters Tlcp and Tlcv on the diagram above. This value is normally zero or less.

3 Upper main real box

the upper allowable distance that a surveyed point, found to be in the Main Zone, may be from the conformance line, in order to be conformant (refer to parameters Tump and Tumv on the diagram above). This value is normally zero or greater.

4 Lower main real box

the lower allowable distance that a surveyed point, found to be in the Main Zone, may be from the conformance line, in order to be conformant (refer to parameters Tlmp and Tlmv on the diagram above). This value is normally zero or less.

Steep slope tolerances

The following four tolerances are used only if the *transition slope* is specified, and the design slope is steeper than the *transition slope*.

5 Upper critical real box

refer to parameters Tucp and Tucv on the diagram above. This value is normally zero or greater.

6 Lower critical real box

refer to parameters Tlcp and Tlcv on the diagram above. This value is normally zero or less.

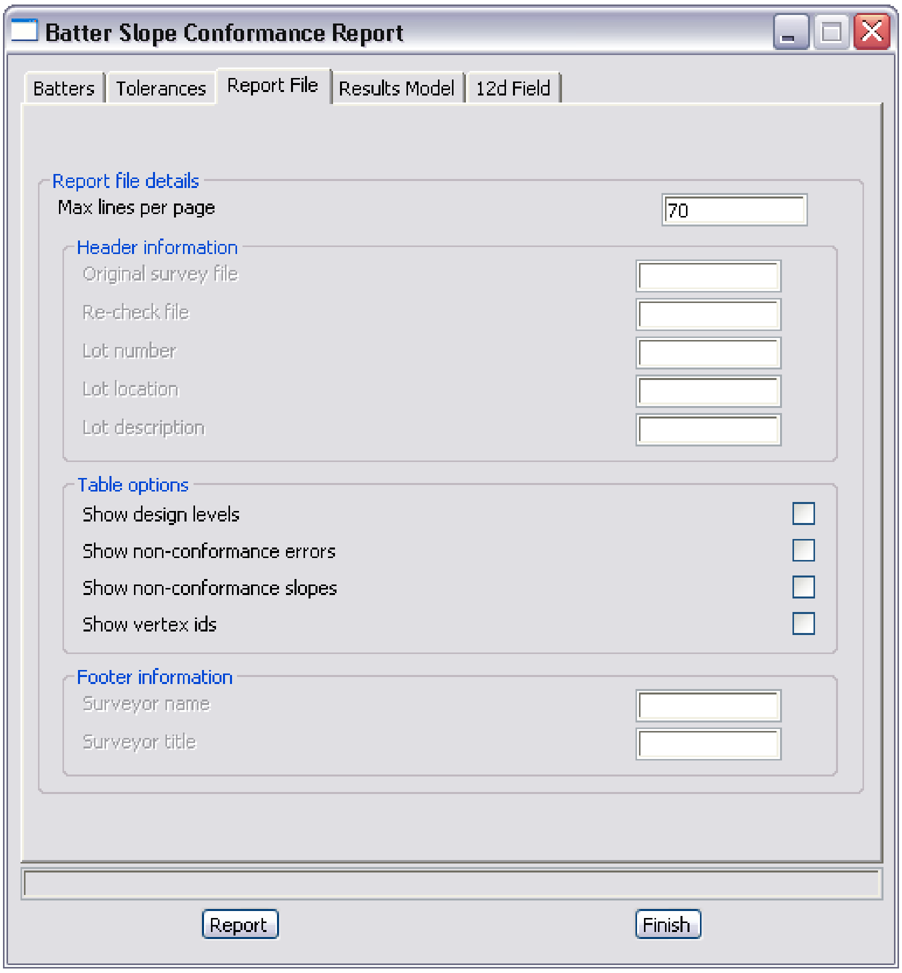
7 Upper main real box

the upper allowable distance that a surveyed point, found to be in the Main Zone, may be from the conformance line, in order to be conformant (refer to parameters Tump and Tumv on the diagram above). This value is normally zero or greater.

8 Lower main real box

the lower allowable distance that a surveyed point, found to be in the Main Zone, may be from the conformance line, in order to be conformant (refer to parameters Tlmp and Tlmv on the diagram above). This value is normally zero or less.

Report File Tab:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Max lines per page	integer box	70	

the maximum number of report lines that can fit on a printed page. This number is needed so that page breaks (form feed characters) can be inserted into the report file at the appropriate places. The number of lines per page will vary depending on the editor/word-processor used for printing, the desired page size and margin widths, the desired page layout (portrait or landscape), and the desired font and font size. As such, the number needs to be tuned for the user's particular report specifications/requirements.

Note1: *you may find that MS Notepad is not a good program for printing out the report file (Notepad doesn't seem to interpret form feed characters).*

Note2: *if* Max lines per page *is set to a large enough number, there will be no page breaks in the report file.*

Header information

Original survey file input box

optional line of text in the report header to identify the original survey data file

Re-check file

optional line of text in the report header to identify the survey data file where the pavement was rechecked.

Lot number input box

optional line of text in the report header to identify the lot number

Lot location input box

optional line of text in the report header to identify the lot location

Lot description input box

optional line of text in the report header to identify the lot description

Table options

Show design levels tick box off

whether to show the design levels *in the column after the* point levels

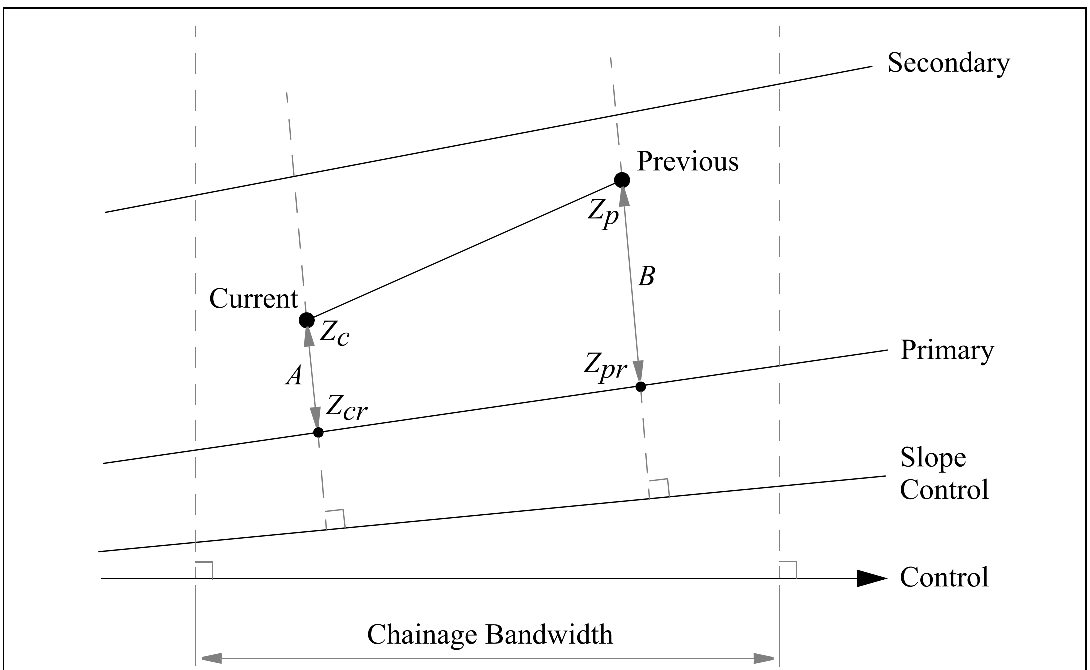
Note: *the* design levels *are always vertically above or below the* point levels*, even if the tolerances are measured perpendicular to the design slope.*

Show non-conformance errors tick box off

whether to show the point errors *for non-conformant points (i.e. the distances the points are out of tolerance). If turned on, the* point errors *will appear in parentheses, (=, after the conformance and tolerance details.*

Show non-conformance slopes tick box off

whether to show the surveyed slopes *at non-conformant points. The* surveyed slope *is the slope between the previous and current point in the report, within a single, offset-sorted chainage band. The slope is calculated as though the previous point lies in the same vertical plane as that of the current point's design slope. As such, the slope is adjusted for any differences in bearing direction between the two design slopes, and for any incline in the road, as shown in the diagram, below:*



$$\text{Surveyed Slope} = 1 \; in \left|\frac{D_h}{D_v}\right| \qquad D_h = A - B \qquad D_v = Z_c - Z_p - Z_{cr} + Z_{pr}$$

If turned on, the surveyed slopes *will appear in square brackets, [=, after the conformance, tolerance and point error details.*

Note: *if the* Chainage bandwidth *on the* Batters Tab *is zero or blank, then the points will only be sorted by chainage, and not sub-sorted by offset within chainage bands. In this case, surveyed slopes will* *never be reported.*

Show vertex ids

if ticked the vertex id of the point rather than the name of the string containing the point will be shown in the report file.

Footer information

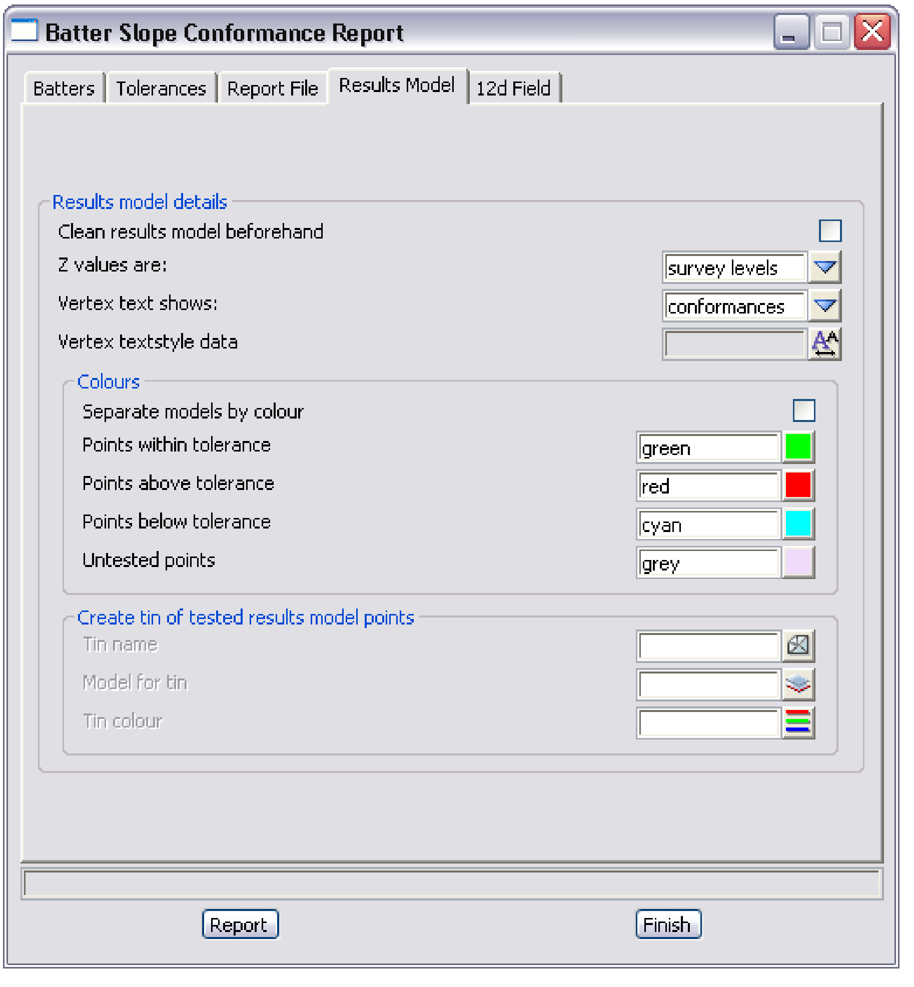
Surveyor name input box

name of the person required to sign the report (optional)

Surveyor title input box

title of the person required to sign the report (optional)

Results Model Tab:



The fields have the following functions:

Field Description	Type	Defaults	Pop-Up
Clean results model beforehand	tick box	off	
Z values are	choice box	levels	levels conformances errors

whether to clean the contents of an existing results model before re-running the option (Clean results model beforehand)

determines what z values to give the points generated in the results model (Z values are)

***Note:** any untested points in the model will have their z values set to their original point levels,* *regardless.*

Vertex text shows	choice box	conformances	levels conformances errors

determines what values to show in the vertex text of the points generated in the results model

***Note1:** any points within tolerance will have a zero error. For these points, if the vertex text is set to* **errors**, *no text will be set.*

***Note2:** any untested points in the model will have the vertex text set to show one of four possible error codes, explaining why the point was not tested. The error codes are* **NUL** *(invalid z-value),* **OCR** *(outside chainage range),* **ERR** *(intersection error or similar), and* **INR** *(inside road).*

Vertex textstyle data	textdata box		user textdata favourites

controls the appearance of the vertex text for the points generated in the results model

Colours

Separate models by colour tick box

if ticked the results model will be several models, each result colour going to a separate model named with a suffix of the colour name.

Points within tolerance	colour box	green	available colours

colour for points within tolerance

Points above tolerance	colour box	red	available colours

colour for points above tolerance

Points below tolerance	colour box	cyan	available colours

colour for points below tolerance

Untested points	colour box	grey	available colours

colour for untested points

***Note:** untested points do not appear in the conformance report.*

Create tin of tested results model points

Tin name	tin box		available tins

name of the tin to create. If the tin already exists, a prompt will ask if it should be replaced. If this field is blank, or if the Results model *field (on the* Batters Tab*) is blank, no tin will be created.*

***Note:** the tin will be created only with the set of tested points created in the* Results model, *and can be used to quickly display contour maps of the results. When creating the tin, an attempt is made to null any triangles outside the **shrink-wrapped** boundary of the tested points. The boundary polygon used for this nulling procedure is determined from the extremity points of each offset-sorted chainage band (see* Chainage bandwidth *on the* Batters Tab*).*

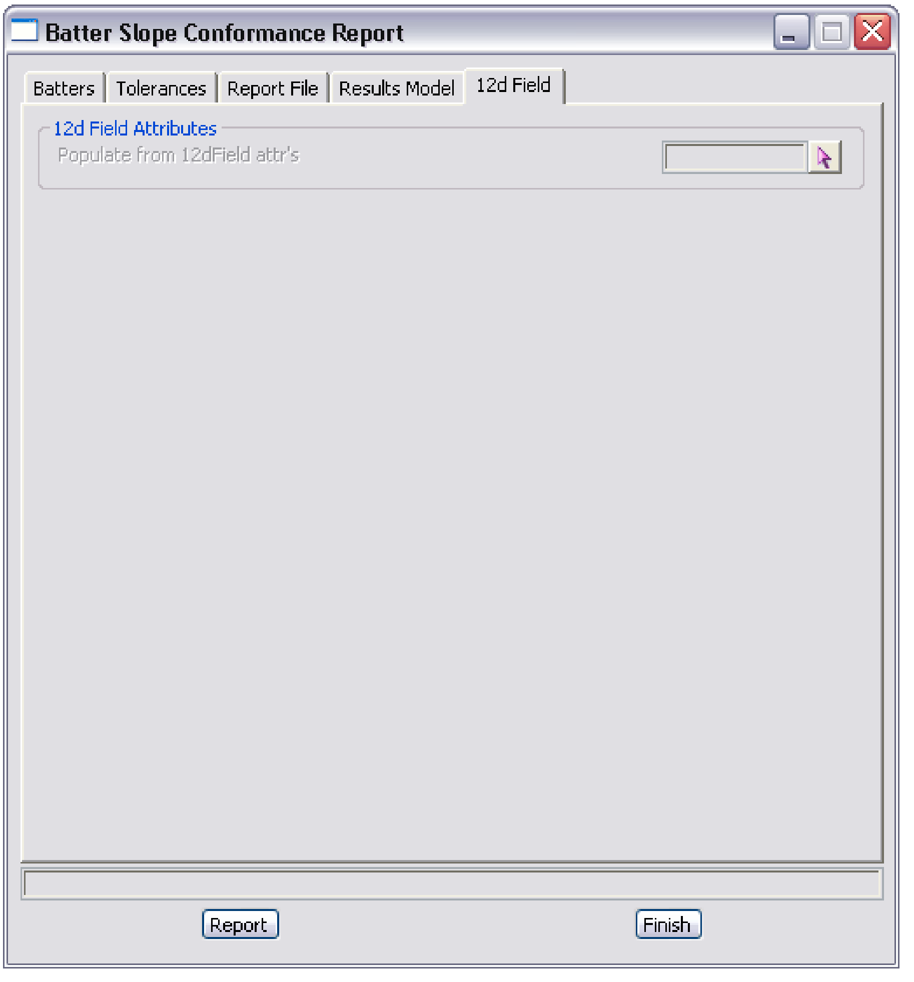
Model for tin	model box		available models

model for tin to be placed in, for viewing purposes. If blank, the tin is not placed into a model.

Tin colour	colour box		available colours

colour for tin. If blank, the tin will adopt the colour selected for Points within tolerance.

12d Field Tab:



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Populate from 12dField attr's

if 12dField was used to pickup the points then the attributes can be used to populate the conformance panel automatically, for example the control lines and primary and secondary edge strings. On selecting the string the panel will be updated.

17.13.2 Pavement report

Position of option on menu: **Survey =>Conformance =>Pavement report**

The **Pavement report** option generates a conformance report on surveyed points representing the top of an as-built pavement, compared against string data representing the top of a designed pavement (level conformance), and compared against tin data representing the bottom of an as-built pavement (thickness conformance).

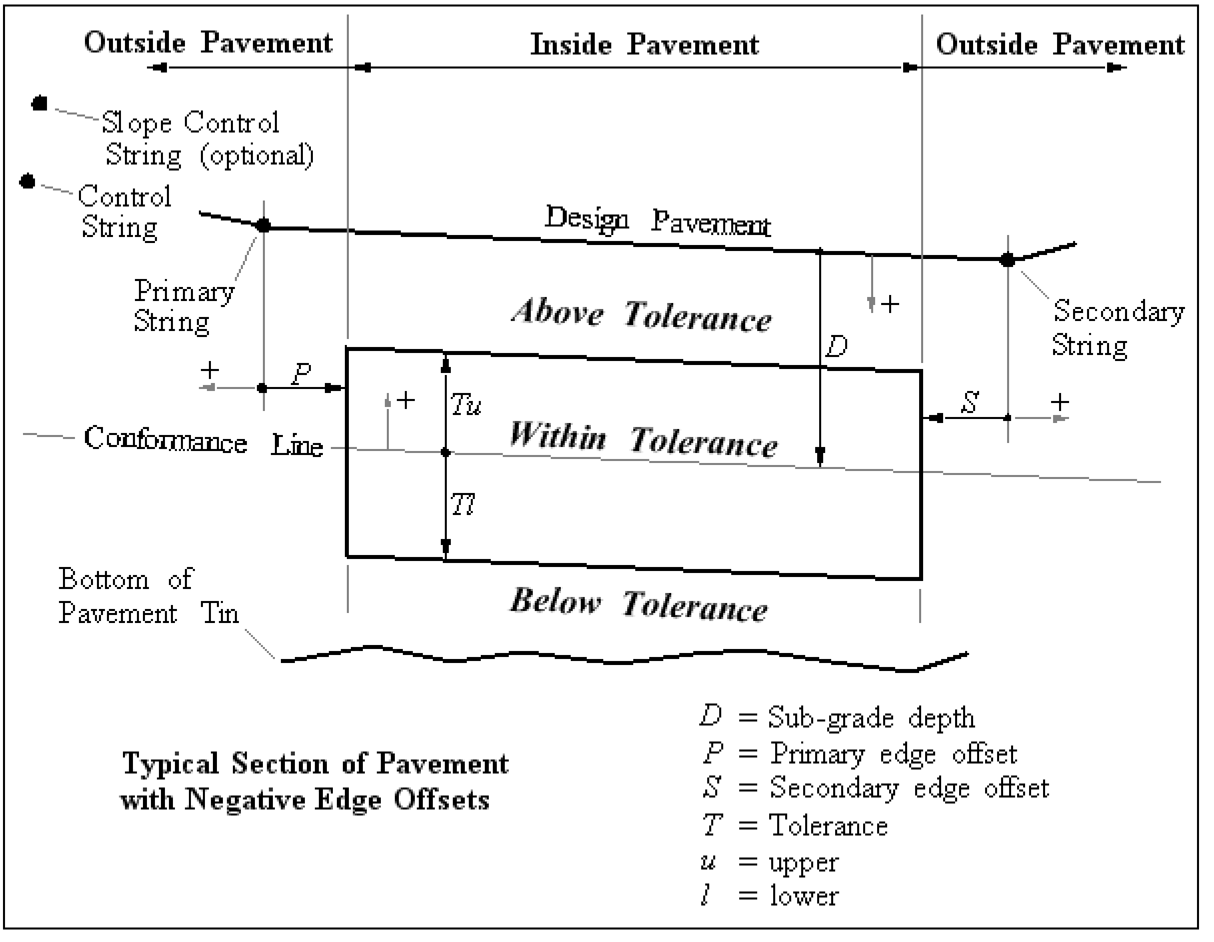
In addition to the conformance report, an output results model of the surveyed points can be generated, grouping the points by colour into their conformance zones (i.e. *within tolerance*, *above tolerance*, *below tolerance*, and *not tested*). The points in the results model can have z-values and vertex text set to show various combinations of point level, point conformance, point error, pavement thickness and pavement thickness error.

Note1: the term *point conformance* is used here to refer to a point's distance from design (distance from the conformance line on diagram below), while the term *point error* refers to a point's distance out of tolerance. As such, a point that is within tolerance (i.e. a *conformant point*) will, in general, have a non-zero conformance and a zero error, while a point that is out of tolerance (i.e. a *non-conformant point*) will have both a non-zero conformance and a non-zero error.

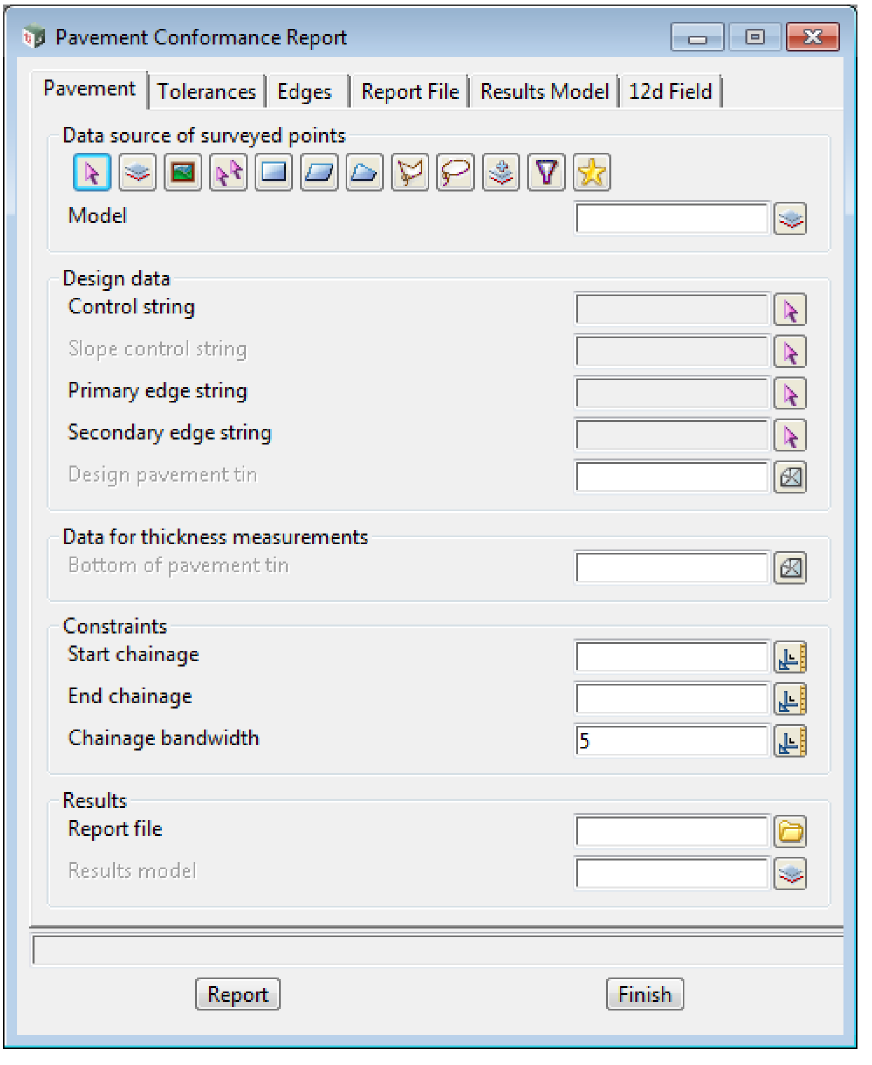
Note2: the point conformances and errors can be measured either vertically or perpendicular to the conformance line. The conformance line is the line from which all point conformances and tolerances are measured, and is always parallel to the design pavement line. When testing a sub-grade survey, the conformance line will normally be below the design pavement line, but when testing a completed-construction survey, the two lines should normally coincide.

Note3: the optional pavement thickness measurements are always made vertically downwards from the surveyed points, to the tin representing the bottom of pavement. Pavement thickness errors will be non-zero wherever the pavement is found to be too thick or too thin. When testing for both level conformance and thickness conformance, both tests must pass for a point to be considered conformant.

A typical section of pavement is shown in the diagram below, along with a schematic showing how the surveyed points are tested for level conformance:



Selecting **Pavement report** brings up the **Pavement Conformance Report** panel



Pavement Tab:

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source of points	source box		

all points selected with this source box (whether they exist in point-string form or line-string form) will be considered as the set of surveyed points to be tested for conformance.

***Note:** the data for the* Point Description *column in the report will be taken from the string names of the selected points. If **all** the string names are blank, however, the point description data in the report will show point ids representing the sequential order that the points are reported in. The results model (if generated) will always have the string names of each point set to match the reported point description.*

Design data

Control string string box

the string selected with this box should normally represent the road centreline, and will be used to determine the chainage and offset of each surveyed point

Slope control string string box

the Slope control string *is optional. If no string is selected, the* Control string*, above, will be used as the*

Slope control string*. The line that is formed in plan, from a surveyed point to the nearest point on the* Slope control string*, defines the vertical plane used to conform that surveyed point. The slope formed between the primary and secondary strings on this vertical plane, is the design slope for that surveyed point.*

Primary string string box

the string selected with this box should represent one edge of the designed pavement (normally the edge that is closer to the Control string*)*

Secondary string string box

the string selected with this box should represent the other edge of the designed pavement (normally the edge that is further away from the Control string*).*

Note: *If the selected string is the same as the* Primary string*, or if no string is selected, then the pavement is defined by one string and is considered to be flat (i.e. zero cross-fall), and to have a width defined by the* Primary *and* Secondary edge offsets *(on the* Tolerances Tab*). This feature can be useful for identifying any surveyed points that are close to an edge line.*

Design pavement tin

if selected this tin will be used for design levels rather than the primary and secondary strings.

Data for thickness measurements

Bottom of pavement tin tin box available tins

the Bottom of pavement tin *is optional. If not selected, the pavement is not tested for thickness conformance. The tin selected with this box should represent the as-built bottom surface of the pavement currently being tested.*

Constraints

Start chainage real box

the start chainage of the surveyed points to be conformed. Any point with a chainage less than the start chainage will not be tested. By default, the start chainage is set when the Control string *is selected, but a different value can be typed in.*

End chainage real box

the end chainage of the surveyed points to be conformed. Any point with a chainage greater than the end chainage will not be tested. By default, the end chainage is set when the Control string *is selected, but a different value can be typed in.*

Chainage bandwidth real box 5.0

if zero or blank, the surveyed points are simply sorted in ascending chainage order. However, if a value greater than zero is entered, the surveyed points will also be sub-sorted into chainage bands, in ascending offset order. Within each chainage band, the difference between the maximum and minimum chainage will be less than the Chainage bandwidth*. This is a useful feature if the surveyed points are set out in rows of roughly equivalent chainage.*

Results

Report file file box *.rpt

the name of the conformance report file to be created. If no extension is given, it will be given an extension of .rpt.

Results model model box available models

the name of the results model to be created. If blank, the results model is not created.

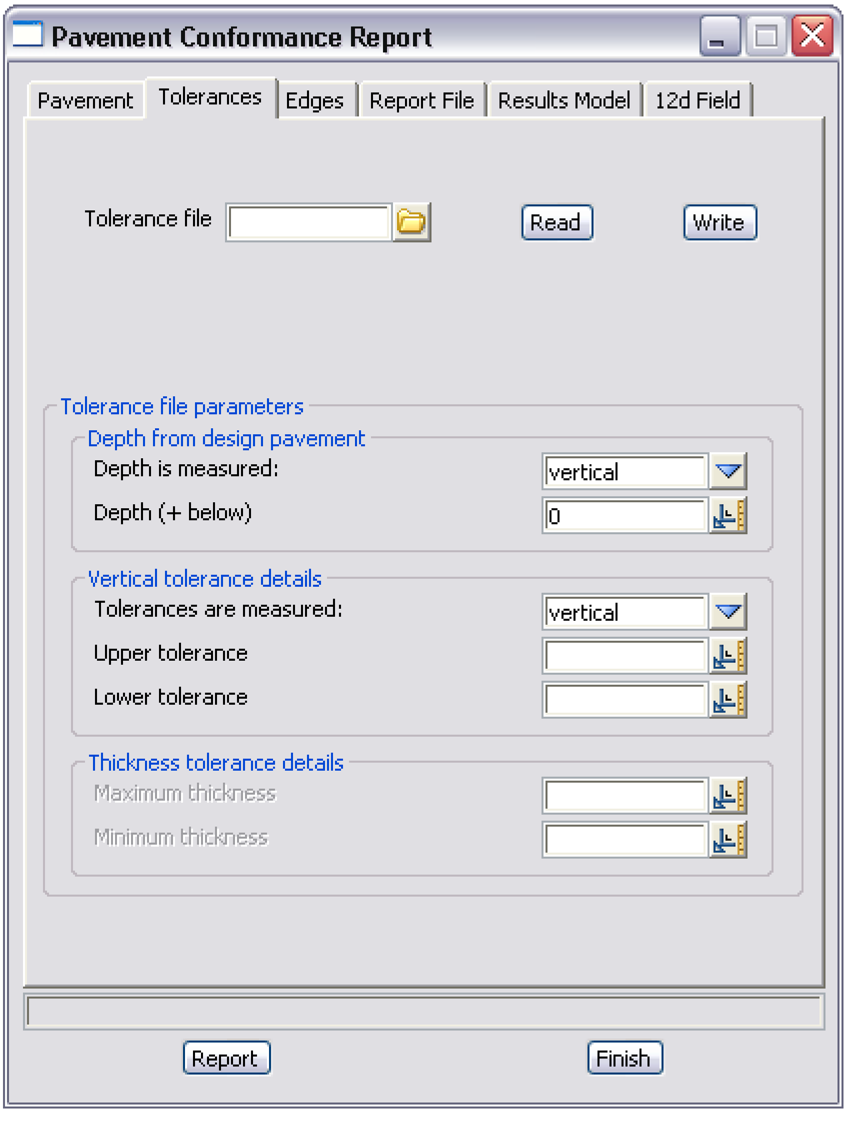
Report button

generates the conformance report file and the results model. This button can be activated regardless of which panel tab is currently active.

Finish button

exits the option and closes the panel. This button can be activated regardless of which panel tab is *currently active.*

Tolerances Tab:



The fields and buttons have the following functions:

Field Description	Type	Defaults	Pop-Up
Tolerance file	file box		*.tol

the name of the tolerance file used to load and/or save the details of the conformance. If no extension is given, it will be given an extension of .tol. *The tolerance file stores the details of every field on the* Tolerances Tab, *the* Report File Tab, *and the* Results Model Tab. *The tolerance file itself is not required to run a conformance test - it is merely provided as a convenience.*

Read button

Pressing this button will populate the panel fields on the last three panel tabs, with the data stored in the tolerance file. If data for a particular field is not found, that field remains unaffected.

Write button

Pressing this button will create a new tolerance file or replace an existing one. The contents of all non-blank fields on the last three panel tabs, are written to the file at this time.

Depth from design pavement

Depth is measured choice box vertical perpendicular to pavement
vertical

whether the sub-grade depth is measured perpendicular to the pavement or vertically (refer to parameter D on the diagram above)

Note: *since pavements typically have cross-fall slopes of less than 7%, the value of this setting should make very little difference to the results.*

Depth (+ below) real box 0.0

the depth of the conformance line from the design pavement (refer to parameter D on the diagram above). If testing a sub-grade survey, this value will normally be greater than zero. If testing a completed-construction survey, however, this value will normally be zero, and the conformance line will coincide with the design pavement line.

Vertical tolerance details

Tolerances are measured choice box vertical perpendicular to pavement
vertical

whether the conformance tolerances are measured perpendicular to the pavement or vertically (refer to parameters Tu and Tl on the diagram above). This field also determines whether the reported point conformances and point errors are measured perpendiculary or vertically.

Note: *since pavements typically have cross-fall slopes of less than 7%, the value of this setting should make very little difference to the results.*

Upper tolerance real box

the upper allowable distance that a surveyed point may be from the conformance line, in order to be conformant (refer to parameter Tu on the diagram above). This value is normally zero or greater.

Lower tolerance real box

the lower allowable distance that a surveyed point may be from the conformance line, in order to be conformant (refer to parameter Tl on the diagram above). This value is normally zero or less.

Thickness tolerance details

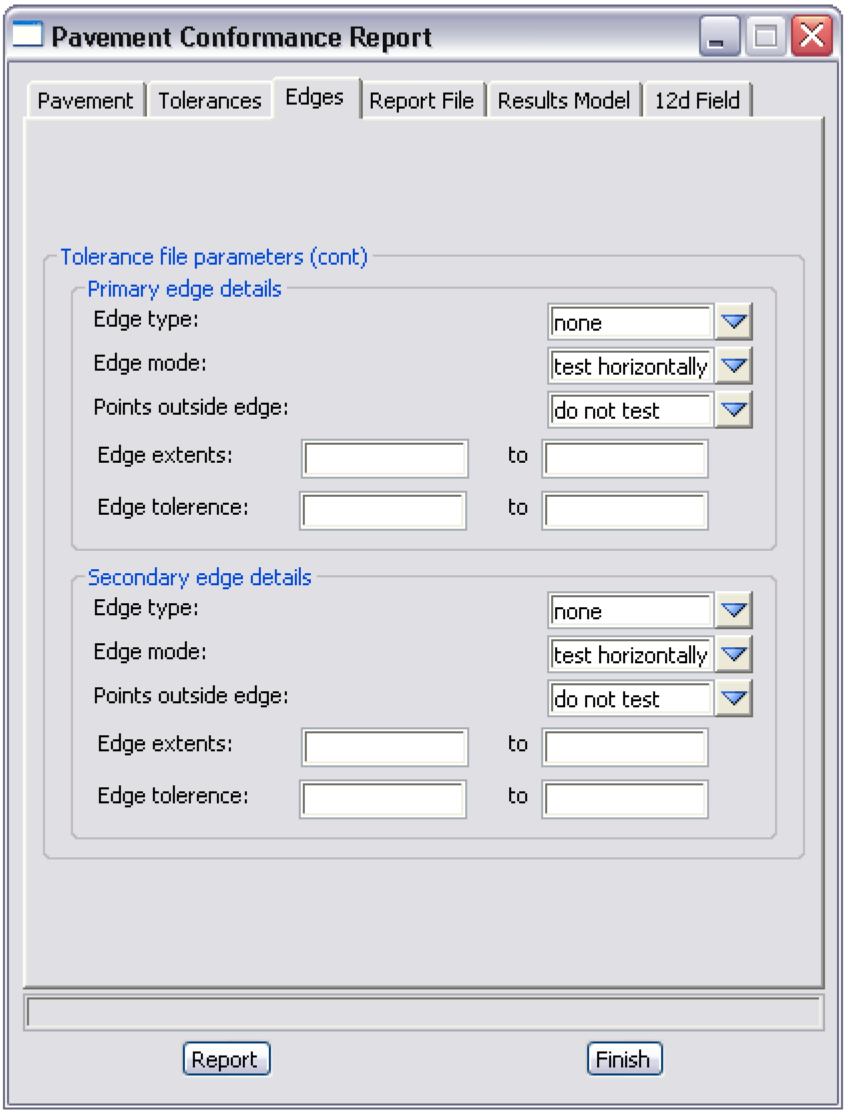
Maximum thickness real box

the maximum allowable vertical distance that a surveyed point may be above the Bottom of pavement tin*, in order to be conformant. If blank, there will be no upper limit to the pavement thickness.*

Minimum thickness real box

the minimum allowable vertical distance that a surveyed point may be above the Bottom of pavement tin*, in order to be conformant. If blank, there will be no lower limit to the pavement thickness.*

Edges Tab:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

The settings for **Primary edge details** and **Secondary edge details** are the same.

Tolerance file parameters (cont).

Primary edge details /Secondary edge details

Edge type	choice box		left, right, join, none

***left**: the string is the left hand edge of the pavement.*
***right:** the string is the right hand edge of the pavement.*
***join**: the string is joint between pavements.*
***none**: the string is not used for edge checks.*

Edge mode	choice box	test horizontally, test vertically & horizontally	

***test horizontally:** the edge is only tested for horizontal conformance*
***test vertically & horizontally:** the edge is only tested for both horizontal and vertical conformance*

Points outside edge	choice box		do not test, test vertically

Determines what to do with any surveyed points found outside the pavement.

do not test: *points outside the edge are not tested.The points will be classified as untested, and in the results model, will have their vertex text set to* EDG. ***Note:*** *untested points do not appear in the conformance report.*

test vertically*: points outside the edge are tested vertically.*

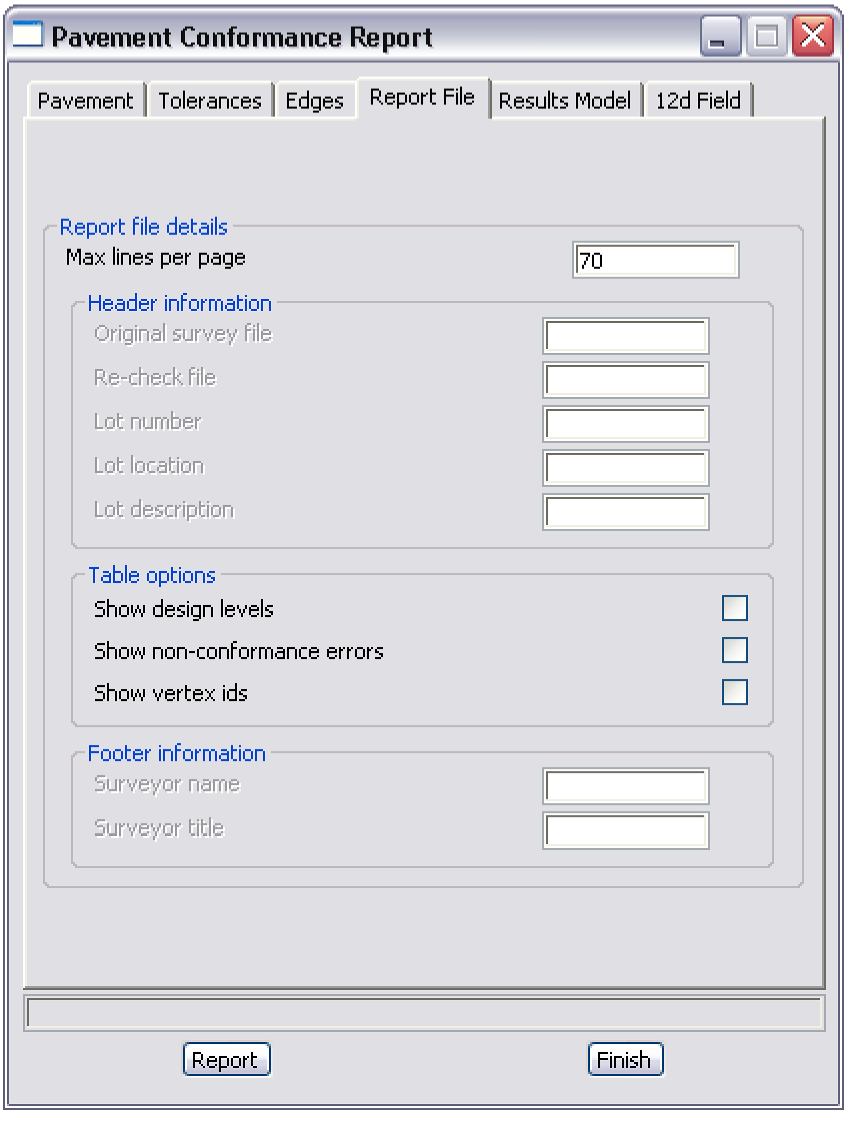
Edge extents

Enter the range around the nominal edge in which a point is considered to be an edge check. This is relative to the 'left/right' setting for the edge, +ve is outside the pavement and -ve into the pavement.

Edge tolerance

Enter the tolerances from the design edge, this is relative to the 'left/right' setting for the edge, +ve is outside the pavement and -ve into the pavement.

Report File Tab:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Max lines per page	integer box	70	

the maximum number of report lines that can fit on a printed page. This number is needed so that page breaks (form feed characters) can be inserted into the report file at the appropriate places. The number *of lines per page will vary depending on the editor/word-processor used for printing, the desired page size and margin widths, the desired page layout (portrait or landscape), and the desired font and font size. As such, the number needs to be tuned for the user's particular report specifications/requirements.*

Note1: *you may find that MS Notepad is not a good program for printing out the report file (Notepad doesn't seem to interpret form feed characters).*

Note2: *if* Max lines per page *is set to a large enough number, there will be no page breaks in the report file.*

Header information

Original survey file input box

optional line of text in the report header to identify the original survey data file

Re-check file

Optional line of text in the report header to identify the survey data file where the pavement was rechecked.

Lot number input box

optional line of text in the report header to identify the lot number

Lot location input box

optional line of text in the report header to identify the lot location

Lot description input box

optional line of text in the report header to identify the lot description

Table options

Show design levels tick box off

whether to show the design levels *in the column after the* point levels

Note: *the* design levels *are always vertically above or below the* point levels*, even if the tolerances are measured perpendicular to the design slope.*

Show non-conformance errors tick box off

whether to show the point errors *and* thickness errors *for non-conformant points (i.e. the distances the points are out of level and thickness conformance, respectively). If turned on, the* point errors *will appear in parentheses,* ( )*, after the conformance and tolerance details, and the* thickness errors *will appear in parentheses,* ( )*, after the thickness and max/min thickness details.*

Show vertex ids tick box

if ticked the vertex id of the point rather than the name of the string containing the point will be shown in the report file.

Footer information

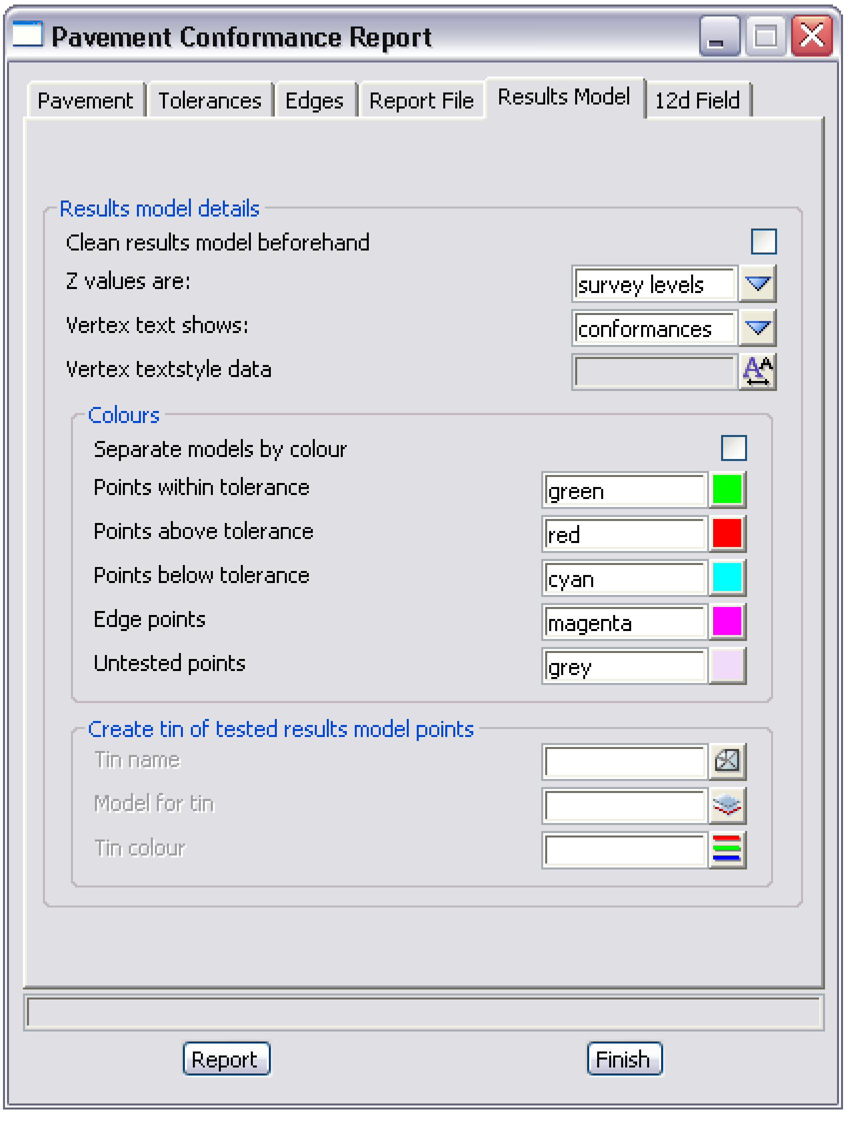
Surveyor name input box

name of the person required to sign the report (optional)

Surveyor title input box

title of the person required to sign the report (optional)

The next tab on the panel is the Results Model Tab:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Clean results model beforehand	tick box	off	

whether to clean the contents of an existing results model before re-running the option

Z values are	choice box	levels	levels conformances errors thicknesses thickness errors

determines what z values to give the points generated in the results model

Note: *any untested points in the model will have their z values set to their original point levels, regardless.*

Vertex text shows	choice box	conformances	levels conformances errors thicknesses thickness errors

determines what values to show in the vertex text of the points generated in the results model

Note1: *any points that pass the level conformance test will have a zero error. For these points, if the vertex text is set to* **errors***, no text will be set. Likewise, any points that pass the thickness conformance test will have a zero thickness error. For these points, if the vertex text is set to* **thickness errors***, no text will be set.*

Note2: *any untested points in the model will have the vertex text set to show one of five possible error codes, explaining why the point was not tested. The error codes are* **NUL** *(invalid z-value),* **OCR** *(outside chainage range),* **ERR** *(intersection error or similar),* **EDG** *(outside pavement edge), and* **NTN** *(bottom of pavement tin not defined at point).*

Vertex textstyle data	textdata box		user textdata favorites

controls the appearance of the vertex text for the points generated in the results model

Colours

Separate models by colour tick box

if ticked the results model will be several models, each result colour going to a separate model named with a suffix of the colour name.

Points within tolerance	colour box	green	available colours

colour for points within tolerance (i.e. points that pass both the level and thickness conformance tests)

Points above tolerance	colour box	red	available colours

colour for points above tolerance (i.e. points that are above the Upper tolerance *or points where the pavement is thicker than the* Maximum thickness*)*

Points below tolerance	colour box	cyan	available colours

colour for points below tolerance (i.e. points that are below the Lower tolerance *or points where the pavement is thinner than the* Minimum thickness*)*

Untested points	colour box	grey	available colours

colour for untested points

Note: *untested points do not appear in the conformance report.*

Create tin of tested results model points

Tin name	tin box		available tins

name of the tin to create. If the tin already exists, a prompt will ask if it should be replaced. If this field is blank, or if the Results model *field (on the* Pavement Tab*) is blank, no tin will be created.*

Note: *the tin will be created only with the set of tested points created in the* Results model*, and (depending on what the points' z-values are set to in the* Results model*) can be used to quickly display contour maps of the results, or to create a* Bottom of pavement tin *for the next pavement layer to be tested. When creating the tin, an attempt is made to null any triangles outside the shrink-wrapped boundary of the tested points. The boundary polygon used for this nulling procedure is determined from the extremity points of each offset-sorted chainage band (see* Chainage bandwidth *on the* Pavement Tab*).*

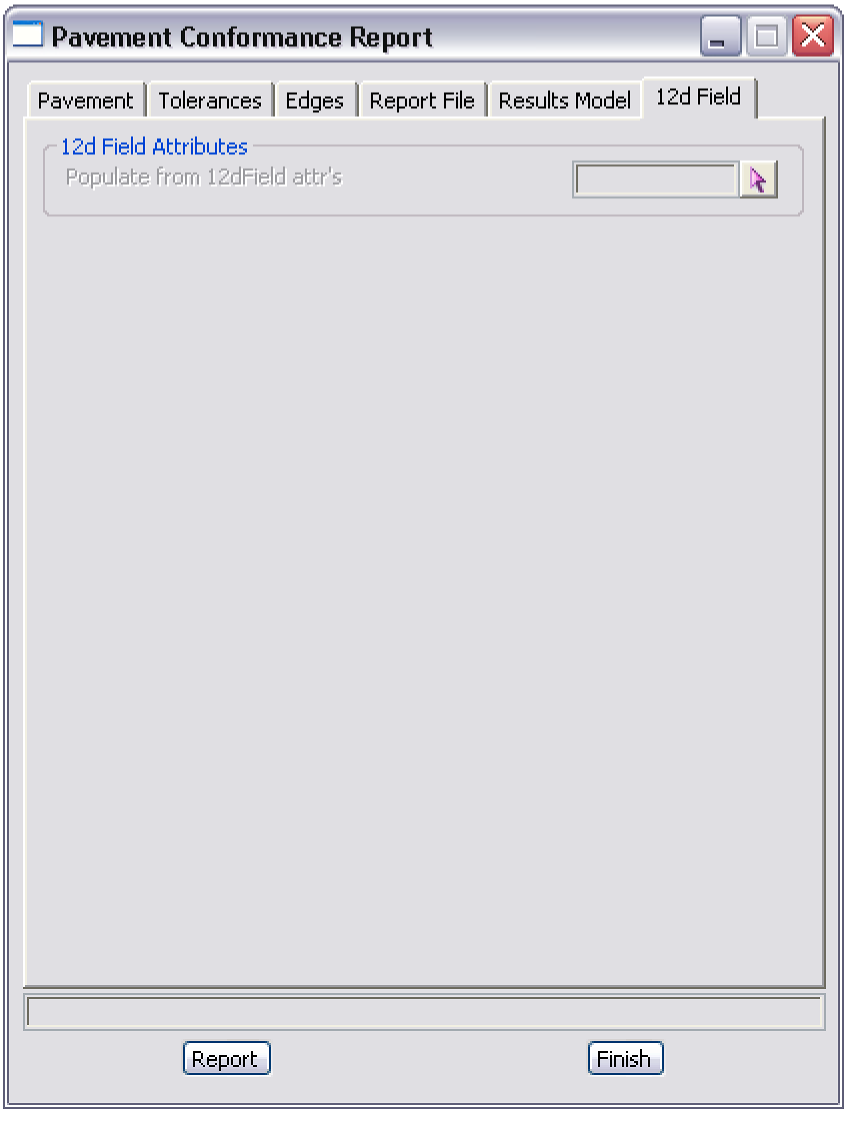
Model for tin	model box		available models

model for tin to be placed in, for viewing purposes. If blank, the tin is not placed into a model.

Tin colour	colour box		available colours

colour for tin. If blank, the tin will adopt the colour selected for Points within tolerance.

12d Field Tab:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Populate from 12dField attr's

if 12dField was used to pickup the points then the attributes can be used to populate the conformance panel automatically, for example the control lines and primary and secondary edge strings. On selecting the string the panel will be updated.

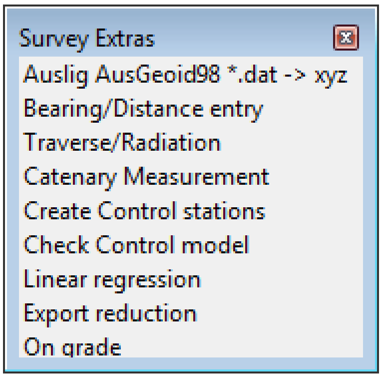
17.14 Extras

Position of menu: **Survey =>Extras**

The **Survey Extras** walk-right menu contains extra survey options, some of which are still being developed.

The **Survey Extras** walk-right menu is

.



Convert AusGeoid98 file into XYZ format
entry and edit by plane bearing and plane distance
bearing distance entry to produce traverse or radiations
create points in a vertical plane
create control stations
check unique point id in control model
Linear regression
write out a field file from a survey function
create an on-grade string

For the option *Auslig AusGeoid98 *.dat ->xyz*, go to 17.14.1 AusGeoid98 *.dat -> xyz
Bearing/Distance entry 17.14.2 Plane Bearing/Distance Entry
Traverse/Radiation 17.14.3 Bearing/Distance Entry for Traverse/ Radiation
Catenary measurement 17.14.4 Catenary Measurement
Create Control Station 17.14.5 Create Control Stations
Check control model 17.14.6 Check Control Model
Linear regression 17.14.7 Linear Regression
Reduction to field file 17.14.8 Survey Function to Field File
On grade 17.14.9 On Grade

17.14.1 AusGeoid98 *.dat -> xyz

Position of option on menu: **Survey =>Extras =>Auslig AusGeoid98 *.dat->xyz**

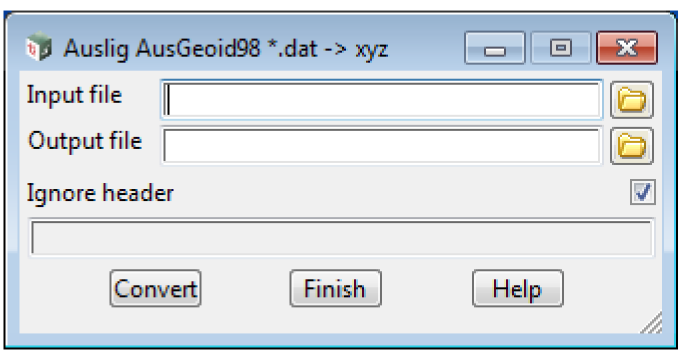
The **Ausgeoid98 *.dat -> xyz** option allows the conversion of a standard AusGeoid98 *.dat file into a XYZ format. This format can then be read into 12d. The values in the XYZ file will have the following meaning

X value will represent the Longitude

Y value will represent the Latitude

Z value will represent the N value

Selecting **Ausgeoid98 *.dat -> xyz** brings up the **Ausgeoid98 *.dat -> xyz** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Input file	file box		*.dat

this filename of the AusGeoid file. The .dat extension is added by default.

Output file file box

this filename of the XYZ file to be produced.

Ignore header tick box ticked

*if ticked, the first line in the *.dat file is ignored (usually the header)*

Convert button

convert the file.

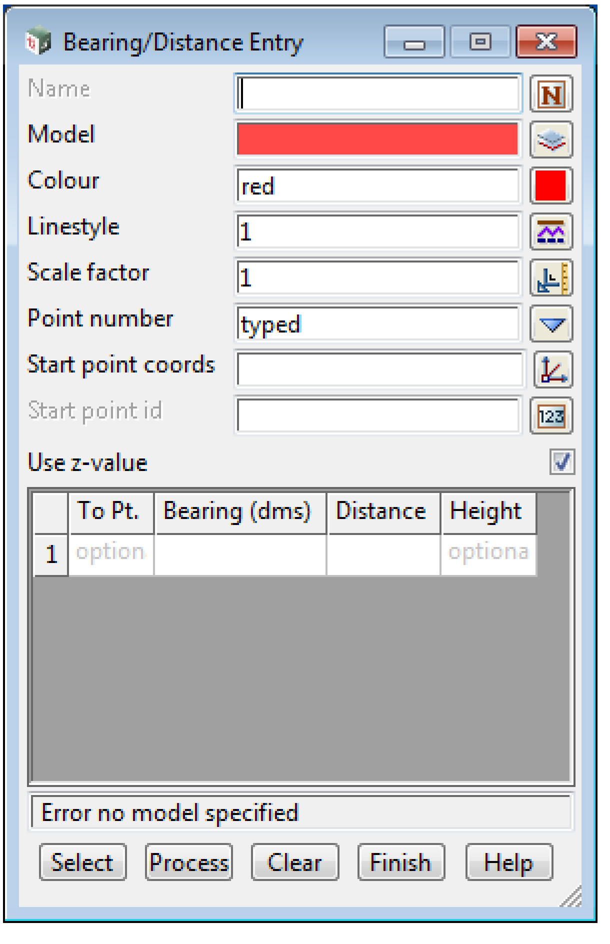
For more information about terminology used in this option, see the Appendix 38 Geodetics Summary .

17.14.2 Plane Bearing/Distance Entry

Position of option on menu: **Survey =>Extras =>Bearing/distance entry**

The **bearing/distance entry** option allows the input of a traverse by manual input of plane bearings and plane distances or by selecting an existing string.

Selecting **Bearing/distance entry** brings up the **Bearing/Distance Entry** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	name box		defined names from names.4d file

this field is optional. If non-blank, the name of the new string.

Model model box available models

if non-blank, the model of the new string. If blank, the original string model is used.

Colour colour box red available colours

if non-blank, the colour of the new string. If blank, the original string colour is used.

Linestyle input 1 available line styles

line style of the string.

Scale factor input projection scale

*This value will be applied to any entered distances. i.e. final distance = entered distance * scale factor*
The scale factor could be a point scale factor, a line scale factor or a combined scale factor.
The final distance will be used for coordinate calculations.

Use z- value tick box transit

*if **ticked**, the z-values for each point can be entered.*
*If **no ticked** then z-values are not entered.*

Point id input typed none, auto increment, typed

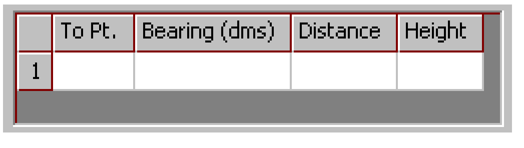
*if **none,** no point ids are entered and the grid control will not show a column for point ids.*

*If **auto increment** then the values of point ids will be incremented by a value of 1 starting from the specified **Start point id**. If no **Start point id** is specified then no point ids will be allocated. No column for point ids is shown in the grid control.*

*if **typed** then the values of point ids will be incremented by a value of 1, starting from the specified **Start point id**. If no **Start point id** is specified then no point ids will be allocated unless a value is entered in the grid control on which time the next value in the grid will have a incremented value. The column for point ids is shown in the grid control.*

VALUES IN GRID grid box

The columns shown in the grid will depend on the selection of the **Use z- value** *and* **Point id** *fields.*



To Pt. input

*if **none** is selected for the* **Point id** *field, this column will not be displayed.*

*if **auto increment** is selected for the* **Point id** *field, this column will not be displayed.*

*if **typed** is selected for the* **Point id** *field, this column will be displayed. In this case, the point id will increment automatically by entering over the field. I f a new value is typed into the To Pt. field, the next line will increment from that number.*

Bearing (dms) angle box

The user should enter the bearing for the segment into this field.

Distance input box

the user should enter the distance of the segment.

Height input box

This column will only be visible in the grid if the **Use z- value** *tickbox has been ticked. The user should enter the height of the point. This value will not be used for calculation of the segment. It will simply be assigned to the newly created vertex.*

Select button

on pressing the select button, a user is able to pick an existing string from the current view. If a non-

traverse type string is selected an option to convert it to a traverse string will be given. The grid control will be filled with the relevant information for the traverse string.

If a traverse string has been modified by some other process (e.g. move) an option will be given to adopt the new characteristics of the string as displayed or revert back to the information that originally defined the traverse string. Depending on which option is selected, the grid will be filled with the relevant information.

Process button

changes to the traverse can be made in the grid control for lines already defined. For example, a distance entry may be incorrectly typed in and edited some time later. By using the process button the traverse string is re-calculated using the current values in the grid control.

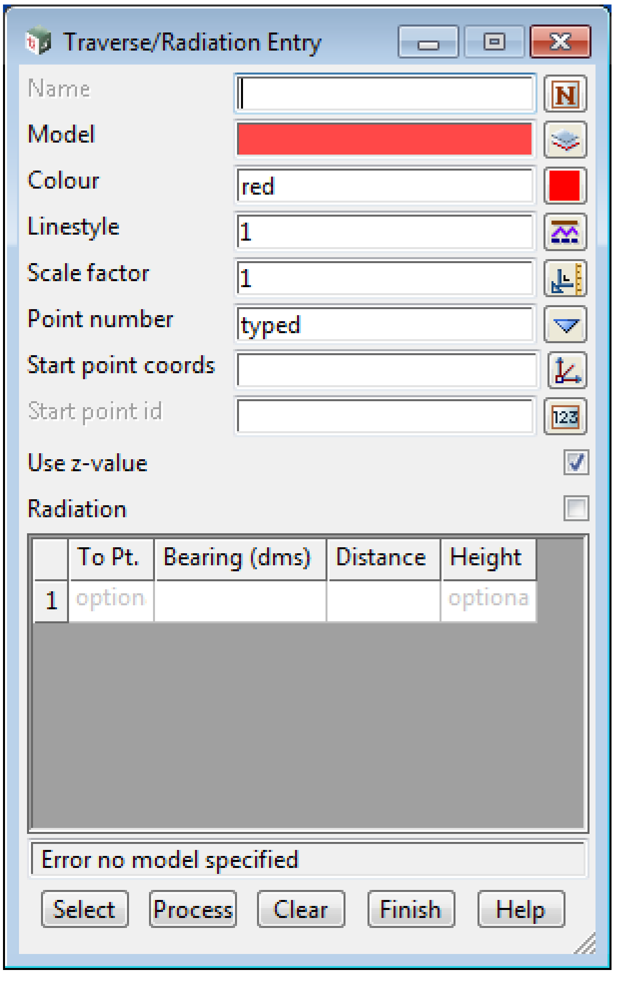
17.14.3 Bearing/Distance Entry for Traverse/Radiation

Position of option on menu: **Survey =>Extras =>Traverse/Radiation**

The **Traverse/radiation** option allows the input of a traverse or radiations by manual input of plane bearings and plane distances or by selecting an existing string.

Bearing/Distance Traverse/Radiation Entry

Selecting **Traverse/Radiation** brings up the **Bearing/Distance Traverse/Radiation Entry** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	name box		defined names from names.4d file

this field is optional. If non-blank, the name of the new string/strings.

Model	model box		available models

if non-blank, the model of the new string. If blank, the original string model is used.

Colour	colour box	red	available colours

if non-blank, the colour of the new string. If blank, the original string colour is used.

Linestyle	input	1	available line styles

line style of the string.

Scale factor	input	projection scale	

*This value will be applied to any entered distances. i.e. final distance = entered distance * scale factor*
The scale factor could be a point scale factor, a line scale factor or a combined scale factor.
The final distance will be used for coordinate calculations.

Start point coords	xyz pick box		

coordinates of the first point. The bearing/distances start from this point.

Start point id	input		

if Point ids is auto increment, point id of the first point.

Point id	input	typed	none, auto increment, typed

*if **none,** no point ids are entered and the grid control will not show a column for point ids.*

*If **auto increment** then the values of point ids will be incremented by a value of 1 starting from the specified **Start point id**. If no **Start point id** is specified then no point ids will be allocated. No column for point ids is shown in the grid control.*

*if **typed** then the values of point ids will be incremented by a value of 1, starting from the specified **Start point id**. If no **Start point id** is specified then no point ids will be allocated unless a value is entered in the grid control on which time the next value in the grid will have a incremented value. The column for point ids is shown in the grid control.*

Use z- value	tick box		

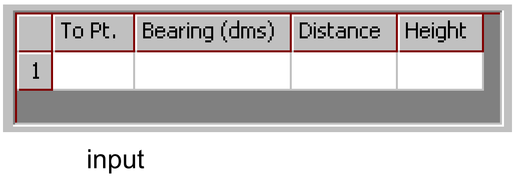
*if **ticked**, the z-values for each point can be entered.*
*If **not ticked** then z-values are not entered.*

Radiation	tick box		

*if **ticked**, the bearing distances are used to create points that are radiations from the start point.*
*If **not ticked** the bearing/distances are used to create a traverse string beginning at the start point.*

Values In Grid	grid box		

The columns shown in the grid will depend on the selection of the **Use z- value** *and* **Point id** *fields.*



To Pt.	input		

*if **none** is selected for the* **Point id** *field, this column will not be displayed.*

*if **auto increment** is selected for the* **Point id** *field, this column will not be displayed.*

*if **typed** is selected for the* **Point id** *field, this column will be displayed. In this case, the point id will increment automatically by entering over the field. I f a new value is typed into the To Pt. field, the next line will increment from that number.*

Bearing (dms) angle box

The user should enter the bearing for the segment into this field.

Distance input box

the user should enter the distance of the segment.

Height input box

This column will only be visible in the grid if the **Use z- value** *tickbox has been ticked. The user should enter the height of the point. This value will not be used for calculation of the segment. It will simply be assigned to the newly created vertex.*

Select button

on pressing the select button, a user is able to pick an existing string from the current view. If a non-traverse type string is selected an option to convert it to a traverse string will be given. The grid control will be filled with the relevant information for the traverse string.

If a traverse string has been modified by some other process (e.g. move) an option will be given to adopt the new characteristics of the string as displayed or revert back to the information that originally defined the traverse string. Depending on which option is selected, the grid will be filled with the relevant information.

Process button

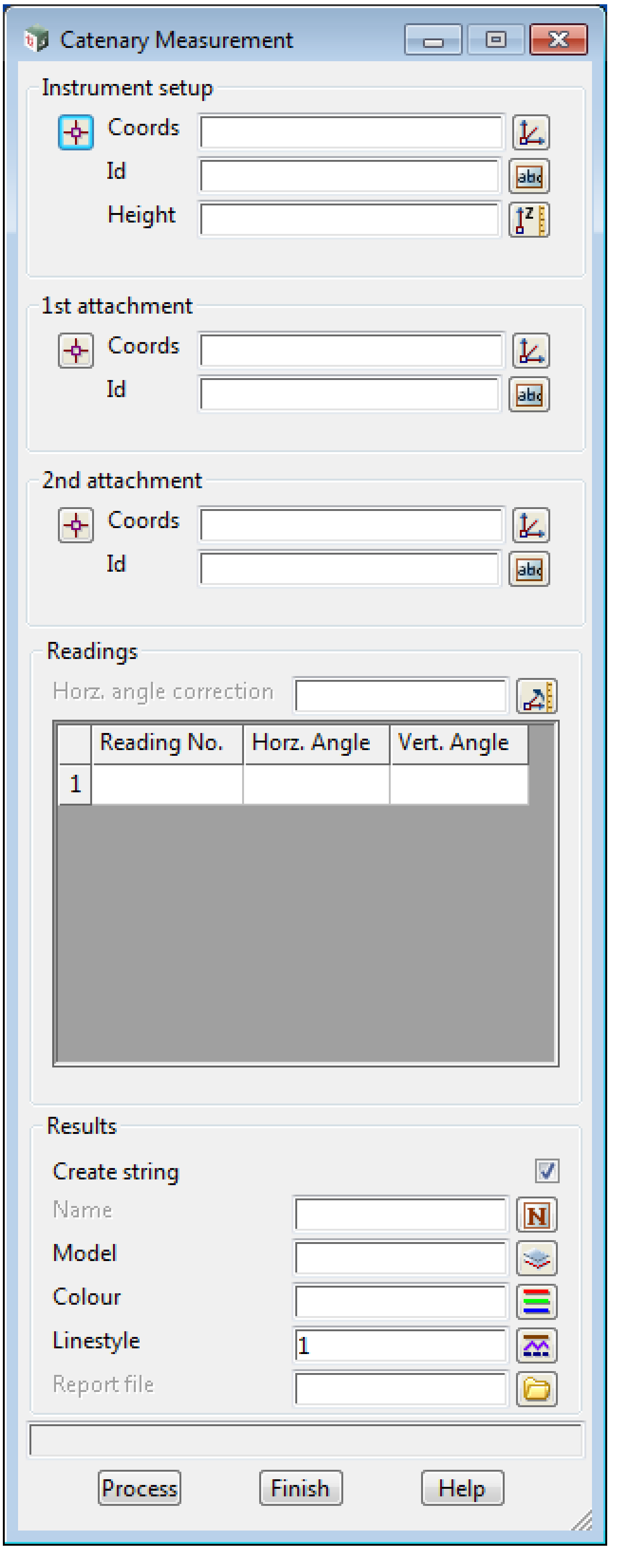
changes to the traverse can be made in the grid control for lines already defined. For example, a distance entry may be incorrectly typed in and edited some time later. By using the process button the traverse string is re-calculated using the current values in the grid control.

17.14.4 Catenary Measurement

Position of option on menu: **Survey =>Extras =>Catenary measurement**

The **Catenary measurements** option allows for the picking up of data in the vertical plane between two known points using only a bearing and a vertical angle. For example, picking up wires between towers where a distance measurement to the cable is not possible.

Selecting **Catenary measurements** brings up the **Catenary Measurements** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Instrument setup

Coords xyz box

the xyz coordinates of the instrument.

Id input box

the vertex id (point id) of the instrument.

Height input box

the height of the instrument.

1st Attachment

Coords xyz box

the xyz coordinates of the attachment point of the cable on the first tower.

Id input box

the vertex id (point id) of the first tower.

2nd Attachment

Coords xyz box

the xyz coordinates of the attachment point of the cable on the second tower.

Id input box

the vertex id (point id) of the second tower.

Reading

Horizontal angle correction angle box

angle to be subtracted from the Horizontal Angle reading to give true bearing.

Reading No input box

number for the reading.

Horz. Angle angle box

Horizontal Angle reading. The Horizontal angle correction is subtracted from this to give the true bearing.

Vert. Angle angle box

Vertical Angle reading.

Results

Create string tick box ticked

if ticked, a string of the observed points is created.

Name input box

name of the created string.

Model model box available models

name of the model for the created string.

Colour colour box red available colours

the colour of the created string.

Linestyle input 1 available linestyles

linestyle of the created string.

Report file file box *.rpt files

if non-blank, the name of the file to write the report to.

Process button

reduce the angle readings.

17.14.5 Create Control Stations

Position of option on menu: **Survey =>Extras =>Create Control stations**

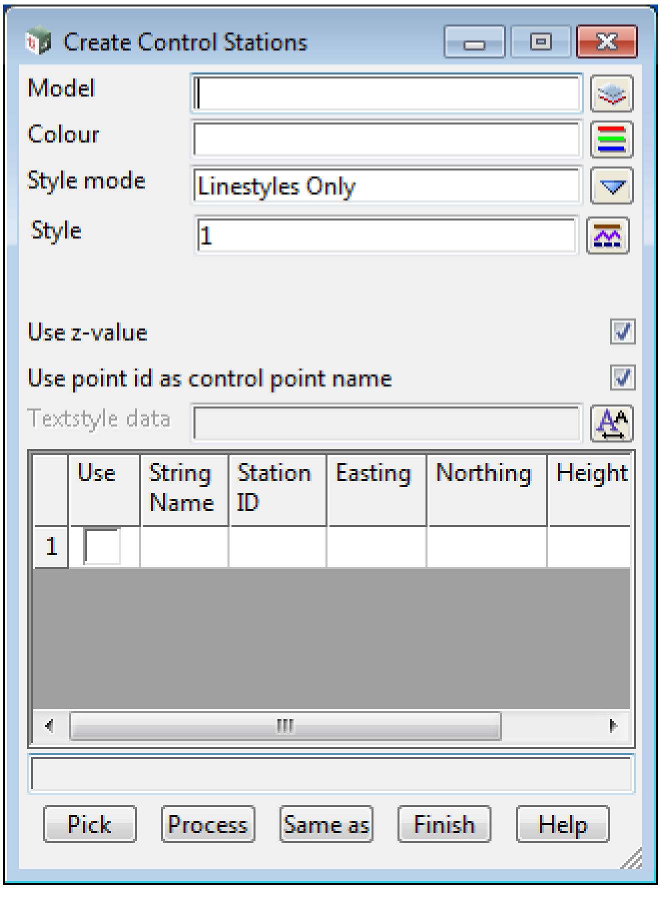
Position of option on menu: **Strings =>Create =>Create Control Stations**

A *control station* (for surveying) consists of a one point string. For V6, the *point name* is the name of the control station. For V5 and before, the *string name* is used as name of the control station.

Control stations are used when reducing 12d field files in the *Survey Reduction* module.

The **Create Control Stations** options allows the user to define control stations in a given model. The option checks that the point ids are unique in the control model.

Selecting **Create Control Stations** brings up the **Create Control Station** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model	model box		available models

name of the model for the control stations.

Colour	colour box	red	available colours

the colour of the new control stations.

Line style	input	1	available line styles

line style of the control station.

Use z-value tick box ticked

if ticked, the height value must be specified for all of the stations.

Use point id as control point name tick box ticked

if ticked, the point names are used as the names of the control points and the ***point names*** *are checked for uniqueness.*
If not ticked, the name of the string is used as the names of the control points and the ***string names*** *are checked for uniqueness. This is only for compatibility in pre* **12d Model** *V6.0.*

Station name Easting Northing Height Remark

the information for stations to be created is entered into the grid. The stations are created by selecting the **Process** *button.*

If a station name is typed in and the <enter> key pressed then the Model is searched and if a control station already exists of the same name (either point name or string name depending on the ***Use point id as control point name*** *flag), the co-ordinates are displayed in the grid.*

The grid control values can be entered/displayed using valid inputs into the various fields.

***Station name**. The associated text that identifies the control point.*
***Easting** The easting coordinate of the control point.*
***Northing** The northing coordinate of the control point.*
***Height** The height of the station. (Only displayed if the use z-value tickbox is selected.)*
***Remark** The description of the control station.*

Process button

After the **Process** *button is chosen, the control stations in the grid are created. If a station of the given name already exists in the control Model, the user is asked if the co-ordinates of the station are to be updated by the values given in the grid.*

Same as button

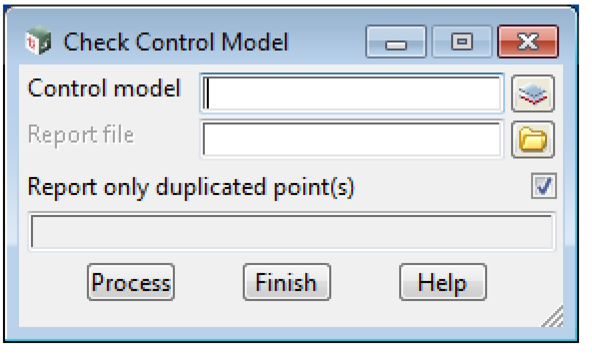
After the **Same as** *button is chosen, another string is selected and information about it is used for the Model, Colour and Point style fields in this panel.*

17.14.6 Check Control Model

Position of option on menu: **Survey =>Extras =>Check Control model**

The **Check control model** option checks that there are no points in the control model with the same point id. A report of all the points in the control models, or only those with clashing point ids, is created.

Selecting **Check control model** brings up the **Check Control Model** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Control model	model box		available models

the model the check for unique point ids.

Report file file box *.rpt files

if non-blank, the name of the file to write the report to.

Report only duplicated point(s) tick box ticked

if ticked, only those points with duplicated point ids are reported.
If not ticked, all points in the control model are written out.

Process button

check the Control model for duplicated point ids.

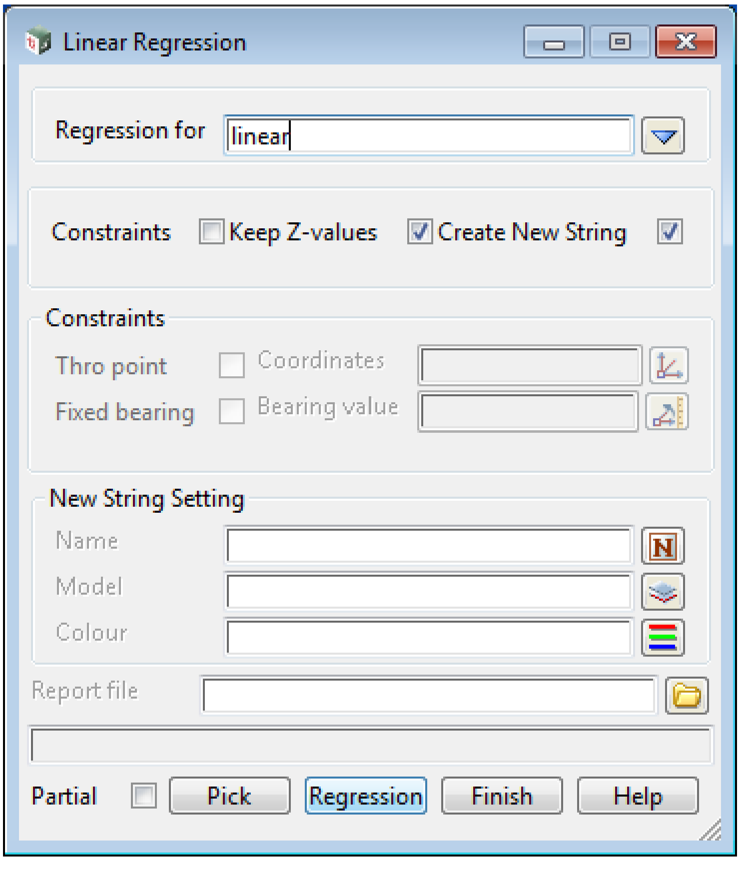
17.14.7 Linear Regression

Position of option on menu: Survey =>Extras =>Linear regression

Position of option on menu: Strings =>Utilities =>Linear regression

The linear regression option will create a line, circle or arc of best fit through points on a string. The line can be constrained to go through a selected point or have a given bearing.

Selecting **linear regression** brings up the **linear regression** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Regression for	choice box	line	line, circle, arc

if line/circle/arc, the line/circle/arc of best fit through the points is created.

Constraints	tick box		

if ticked, the regression line can be restrained to go through a selected point or be given a selected bearing.

Keep z-values	tick box	tick	

if ticked, the z-values of the selected string are used on the regression string.

Create new string	tick box	tick	

if ticked, a new adjusted string is created. If not ticked, the selected string is adjusted.

Constraints

Thro point tick box

if ticked, the regression line is constrained to go through the point with co-ordinates given in the **Coordinates** *field.*

Coordinates

if ***thro point*** *is ticked, the regression line is constrained to go through the point given in this field.*

Fixed bearing tick box

if ticked, the regression line is constrained to have the bearing given in the **Bearing value** *field.*

Bearing value

if ***thro point*** *is ticked, the regression line is constrained to have the bearing given in this field.*

Name name box

if non-blank, the name of the new string. If blank, the original string name is used.

Model model box available models

if non-blank, the model of the new string. If blank, the original string model is used.

Colour colour box available colours

if non-blank, the colour of the new string. If blank, the original string colour is used.

Report file *.rpt files

if non-blank, a report for the adjustment is created with this name. If non-blank, no report is created.

Partial tick box

if ticked, only part of the string is used in the regression.

Pick button

select the string to create a regression line/circle/arc from.

Regression button

perform the regression.

17.14.8 Survey Function to Field File

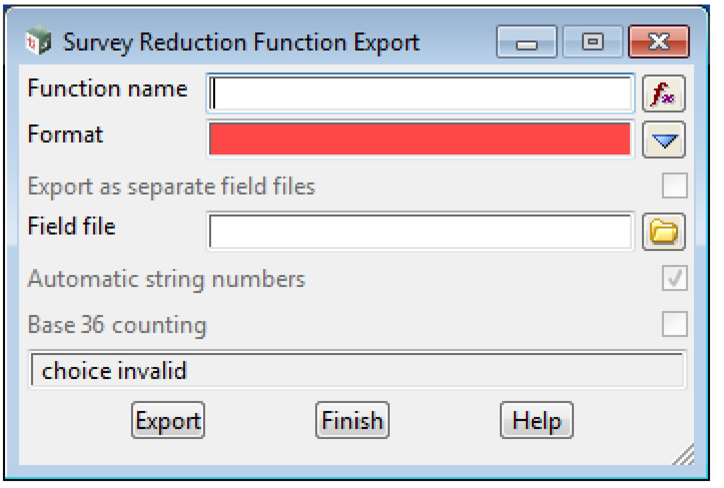
Position of option on menu: **Survey =>Extras =>Reduction to field file**

The **reduction to field file** option writes out the field data from a Survey function to a file in the standard 12d field file format.

Selecting **reduction to field file** brings up the **survey reduction function export** panel

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function name	input		available Survey functions



name of the 12d Survey function to write out a 12d field file for.

Field file input *.fld

name of the 12d field file to write out the Survey field data to.

Export button

create the 12d field file.

17.14.9 On Grade

Position of option on menu: **Survey =>Extras =>On grade**

Position of option on menu: **Strings =>Utilities =>On grade**

The **on grade** option produces a string that is a fixed horizontal distance from a selected string with z-values produced by extrapolating the grade from two strings or from a tin.

Selecting **On grade** brings up the **On Grade String** panel

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Primary string string select

string to produce the on grade string from

Horizontal offset input

horizontal distance to offset the on grade string from the primary string, relative to the direction of the primary/control string, +ve to the right

Vertical offset input

vertical distance to offset the on grade string from the primary string

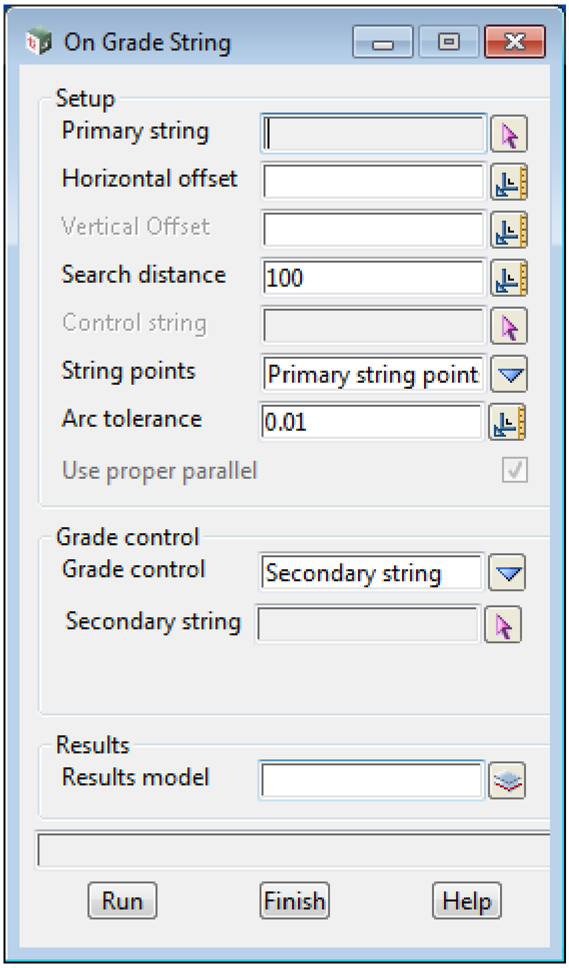
Search distance input

only string cuts inside this distance will be considered

Control string string select

*if **selected**, the horizontal distance is measured at right angles to the control string. In this case, the on-grade may not be parallel to the primary string.*
*If **not selected**, the horizontal distance is measured at right angles to the primary string.*

String points choice box Primary string points
Chainage interval
Special chainage file



*if **Primary string points**, the on-grade string has vertices on it where ever there is a vertex on the primary string. Extra points will also be include for the given arc tolerance.*

Arc tolerance input

*if **non zero**, chord to arc tolerance to use for adding additional vertices*

Use proper parallel tick box

*if **ticked**, the primary string is parallelled to from the on-grade string*

*If **Chainage interval**, vertices are created on the on-grade string corresponding to the given chainage interval along the primary/control string.*

Interval input

the chainage interval the use along the primary/control string for creating on-grade vertices.

*If **Special chainage file**, vertices are created on the on-grade string from chainages on the primary or control string, corresponding to the special chainages file.*

Special chainages file box .spc files

the chainage interval the use along the primary/control string for creating on-grade vertices.

Grade control choice box Secondary string
Tin

*if **Secondary string**, the grade to use to create z-values on the on-grade string is the grade between the secondary string and the primary string.*

Secondary string string select

the grade to use is taken from the secondary to the primary string

*If **Tin**,*

Tin tin box

the tin to use

Offset input

offset from the primary string to the point on the tin to create the grade, relative to the primary string direction, +ve to the right

Results model model box available models

the model for the created on-grade string

Run button

create the on grade string

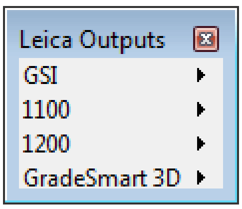
17.15 Leica

Position of menu: **Survey =>Leica**

The **Leica** walk-right menu collects options for working with Leica instruments.

The **Leica** walk-right menu is

.

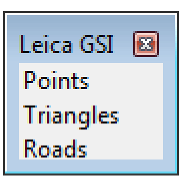


For the option *GSI*, go to [17.15.1 Leica GSI](#)
1100 [17.15.2 Leica 1100](#)
1200 [17.15.6 Leica 1200](#)
GradeSmart 3D [17.15.12 Leica GradeSmart 3D](#)

17.15.1 Leica GSI

The **Leica GSI** walk-right menu collects under the one menu, options that can work with Leica GSI format.

.

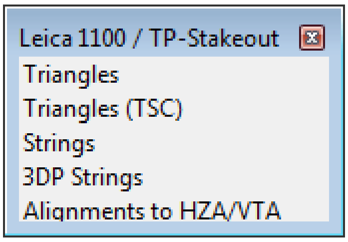


For the option *Points*, go to [17.19.3 Create Points Upload File](#)
Triangles [17.19.2 Create Triangle Upload File](#)
Roads [17.19.4 Create Road Upload File](#)

17.15.2 Leica 1100

The **Leica 1100** walk-right menu collects under the one menu, options that can work with Leica 1100 instruments.

.



For the option *Triangles*, go to 8.2.13 TP Stakeout Triangles Output

Triangles (TSC) 17.15.3 Write TP Stakeout Binary Triangles (TSC Format)

Strings 8.2.14 TP Stakeout Strings Output

3DP Strings 17.15.4 Write TP Stakeout 3DP String

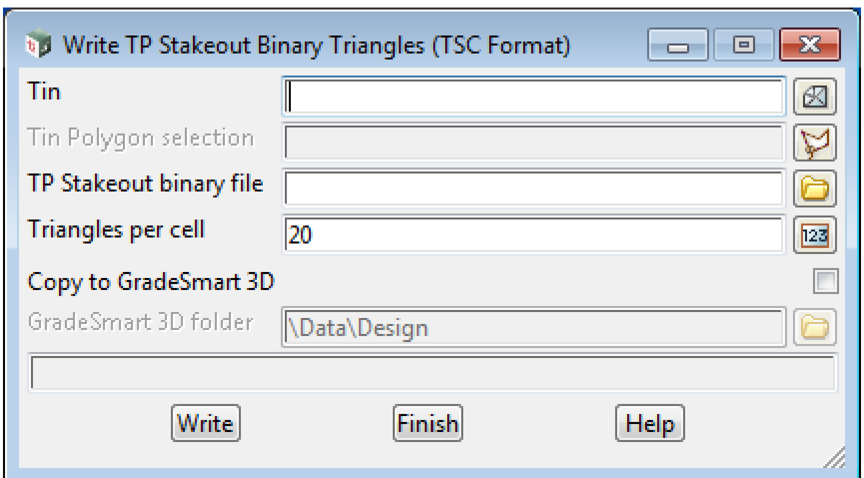
Alignments to HZA/VTA 17.15.5 TP Alignment

17.15.3 Write TP Stakeout Binary Triangles (TSC Format)

Position of option on menu: **Survey => Leica => 1100 =>Triangles (TSC)**

The Write *TP Stakeout Binary triangles TSC* option writes out a tin as a TP Stakeout binary file.

On selecting the **Triangles (TSC)** option, the **Write TP Stakeout Binary Triangles (TSC Format)** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin to write out in TP Stakeout binary format.

Tin polygon selection	poly string-select		

if selected, only triangles with their centroid inside this string are written out

TP Stakeout binary file	input		*.tsc

name of the file for the triangles

Triangles per cell

a TP Stakeout parameter

Copy to GradeSmart 3D tick box

if ticked, copy file to GradeSmart folder

GradeSmart 3D folder folder box

folder for GradeSmart files

Write button

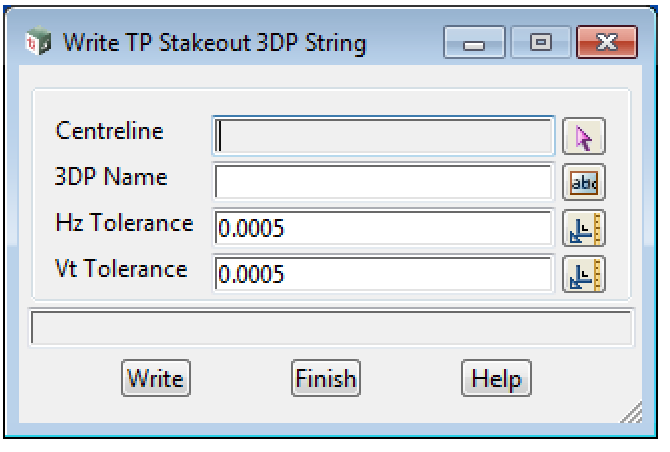
write out the tin in TP Stakeout binary format.

17.15.4 Write TP Stakeout 3DP String

Position of option on menu: **Survey => Leica => 1100 =>3DP Strings**

This panel creates a TP-Stakeout 3DP format file

Selecting **3DP Strings** brings up the **Write TP Stakeout 3DP String** panel**.**



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Centreline string select

select the super alignment or super string to convert, the string must have valid vertical geometry or an error will occur when processing.

3DP Name

the name of the 3DP file to be created, the name of the file must be <= 8 characters.

Hz Tolerance

the arc to chord tolerance of the created horizontal points.

Vt Tolerance

the arc to chord tolerance of the created vertical points.

Write button

write the 3DP file, any errors in processing the selected string will be displayed.

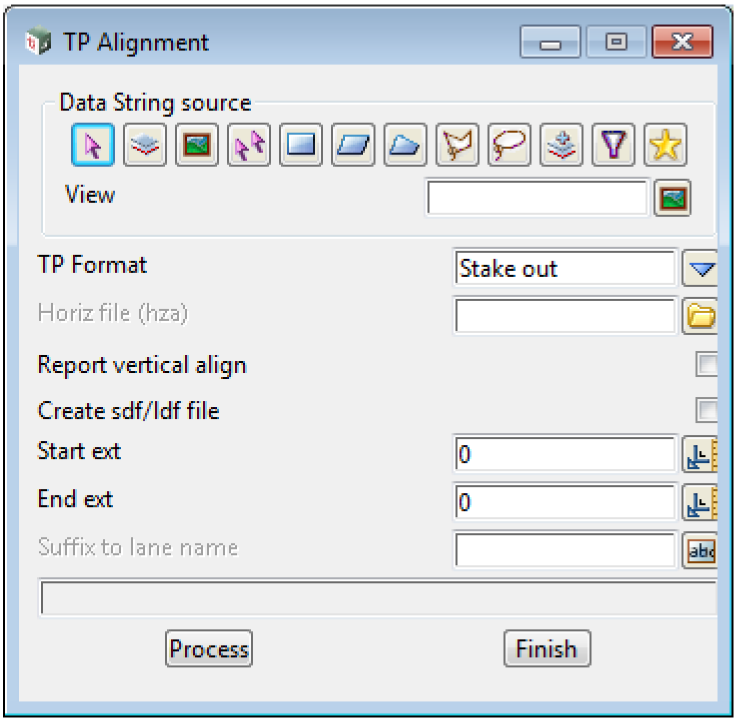
17.15.5 TP Alignment

Position of option on menu: **Survey => Leica => 1100 =>Alignments to HZA/VTA**

This option converts alignments or super alignments into the TP-Setout/TP-Stakeout alignment format. These programs support the clothoid/natural clothoid and cubic parabola transition types. Due to restrictions in the software using these files the file names and alignment names must be

<= 8 characters.

On selecting the **Alignments to HZA/VTA** option, the **TP Alignment** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Data String source

Data source type

data selection type - for a full description go to 4.19.3 Data Source .

Data source model

Source of the alignment/super alignment strings to convert.

TP Format choice box Stake out Stake out, Set out

Stake out - write the data out in TP-Stakeout format.
Set out - write the data out in TP-Setout format.

Horiz file (hza) file

The HZA file to create, the name must be 8 characters or less.

Report vertical align tick box

if ticked the vertical alignment will be written to a .VTA file of the same name as the HZA file.

Create sdf/ldf file tick box

if ticked a surface/lane entry will be created for the alignments being written in the SDF/LDF file of the same name as the HZA file.

Start ext input

The start extension to all alignments being written, +ve extends before the start chainage, -ve invalidates that distance +into the alignment.

End ext input

The end extension to all alignments being written, +ve extends after the end chainage, -ve invalidates that distance back into the alignment.

Suffix to lane name

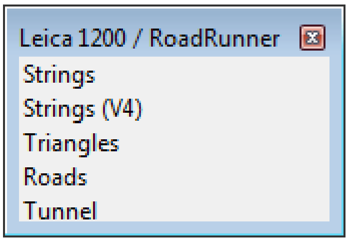
A suffix will be added to the alignment name in the SDF/LDF file entry.

Process button

Create the TP-Stakeout/Setout format files, if errors are encountered during the conversion such as names > 8 characters the appropriate warnings will be displayed.

17.15.6 Leica 1200

The **Leica 1200** walk-right menu collects under the one menu, options that can work with Leica 1200 instruments.



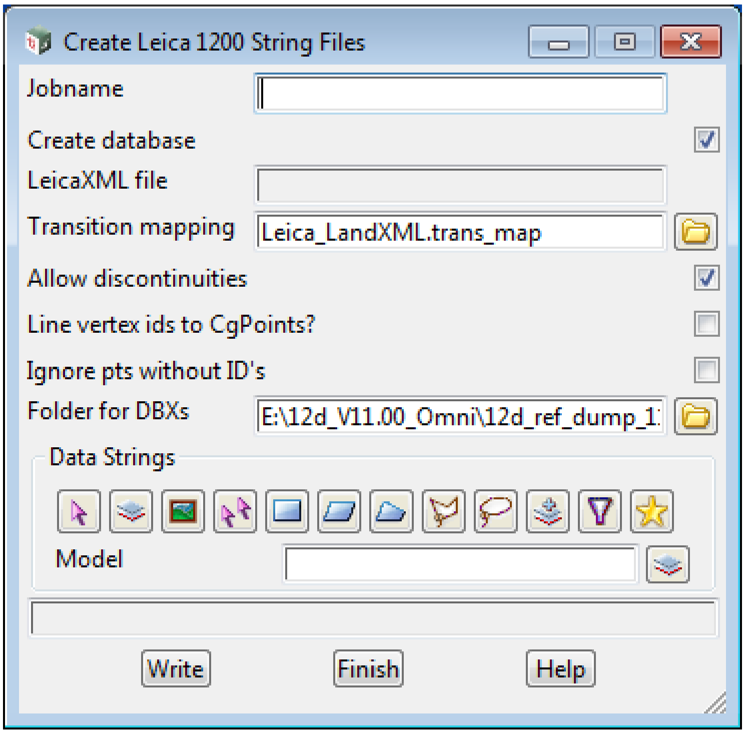
For the option *Strings*, go to [17.15.7 Create Leica 1200 String Files](#)
Strings (V4) [17.15.8 Create Leica 1200 String Files (Old)](#)
Triangles [17.15.9 Create Leica 1200 Triangle Files](#)
Roads [17.15.10 Create Leica 1200 Road Files](#)
Tunnel [17.15.11 Create Leica 1200 Tunnel Files](#)

17.15.7 Create Leica 1200 String Files

Position of menu: **Survey =>Leica => 1200 => Strings**

This panel creates a points and lines database for use on the Leica 1200 series of TPS/GPS instruments. This is done by creating an intermediate LandXML file and then calling a Leica provided converter to create the DBX files.

Selecting **Strings** brings up the **Create Leica 1200 String Files** panel



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Jobname

The internal jobname seen inside the Leica software.

Create database tick box

If ticked will create the 1200 DBX
if not ticked just the LandXML file is created.

LeicaXML file

The name of the Leica LandXML file to create.

Transition mapping file

Due to the inadequate specification of transitions in the LandXML schema users must explicitly map 12d transition types to other vendors types.

Allow discontinuities tick box

If ticked super string discontinuities will not be written to the DBX.

Line vertex ids to CgPoints? tick box

If ticked will write all of the string vertices to CgPoints as well as writing the strings as plan features.

Ignore pts without ID's tick box

If ticked, points without a Point ID will be ignored.

Folder for DBXs file

The folder to create the databases in.

Data Strings

Data source type model

data selection type - for a full description go to 4.19.3 Data Source .

Data source model

The data source to write to the DBX files

Write button

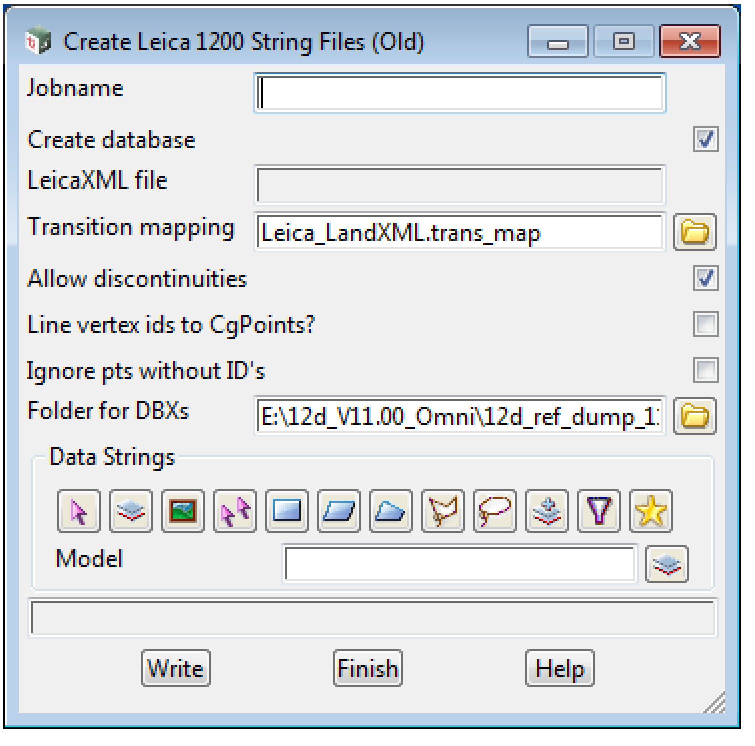
Create the Leica 1200 Strings DBX files.

17.15.8 Create Leica 1200 String Files (Old)

Position of menu: **Survey =>Leica => 1200 => Strings (V4)**

This panel is identical to 17.15.7 Create Leica 1200 String Files but calls an older version of the Leica LandXML/DBX converter.

Selecting **Strings (V4)** brings up the **Create Leica 1200 String Files (Old)** panel

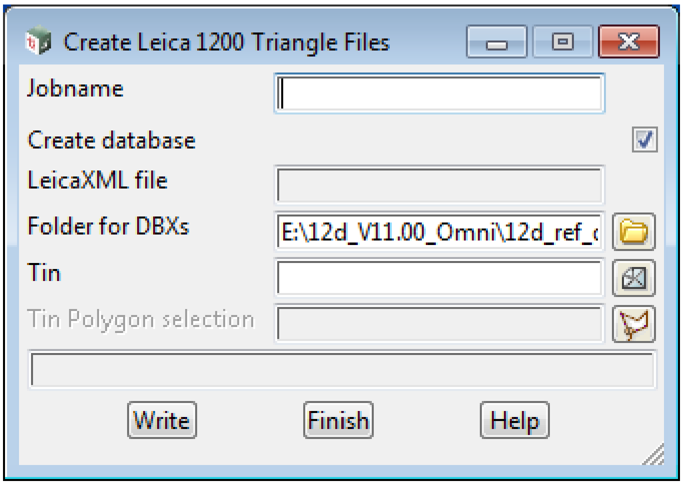


17.15.9 Create Leica 1200 Triangle Files

Position of menu: **Survey =>Leica => 1200 => Triangles**

This panel creates a triangle database for use on the Leica 1200 series of TPS/GPS instruments. This is done by creating an intermediate LandXML file and then calling a Leica provided converter to create the DBX files.

Selecting **Triangles** brings up the **Create Leica 1200 Triangle Files** panel



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Jobname

The internal jobname seen inside the Leica software.

Create database tick box

if ticked will create the 1200 DBX,
if not ticked just the LandXML file is created.

LeicaXML file

The name of the Leica LandXML file to create.

Folder for DBXs file

The folder to create the databases in.

Tin input

The tin to write to the DBX.

Tin Polygon selection

Select a polygon for the region of the tin to write out.

Write button

Create the Leica 1200 Triangle DBX files.

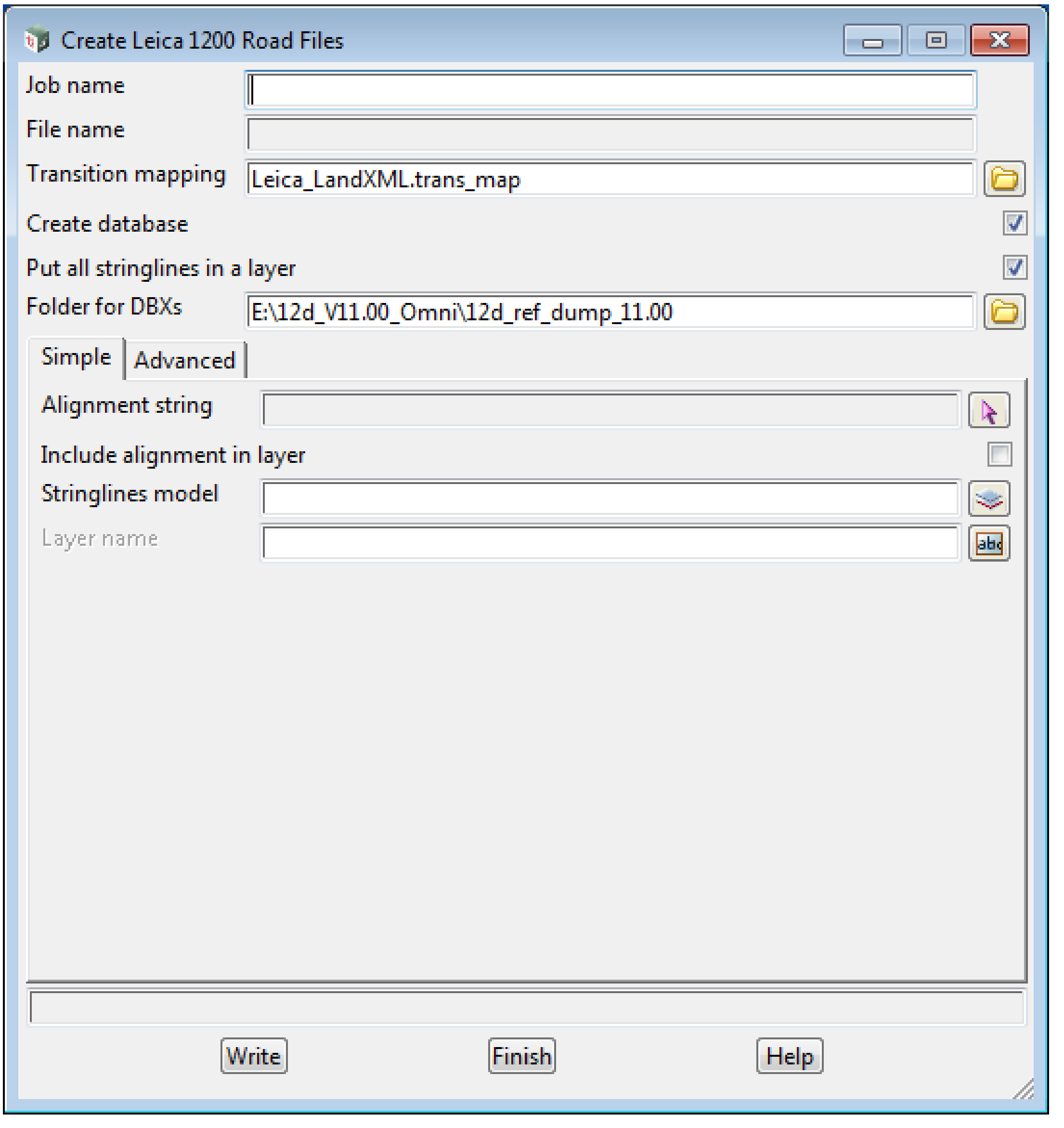
17.15.10 Create Leica 1200 Road Files

Position of menu: **Survey =>Leica => 1200 => Roads**

This panel creates a roads database for use on the Leica 1200 series of TPS/GPS instruments. This is done by creating an intermediate Leica LandXML file and then calling a Leica provided converter to create the DBX files.

This panel allows either a flat group of strings to be written to the DBX or multiple roads and layers in those roads dependent on the users needs.

Selecting **Roads** brings up the **Create Leica 1200 Road Files** panel



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Jobname

The internal jobname seen inside the Leica software.

File name

The name of the Leica LandXML file to create.

Transition mapping file

Due to the inadequate specification of transitions in the LandXML schema users must explicitly map 12d transition types to other vendors types.

Create database tick box

if ticked will create the 1200 DBX,
if not ticked just the LandXML file is created.

Put all stringlines in a layer tick box

if not ticked then the strings in the model selected in the **Stringlines model** *box in the "**Simple**" tab are written to the LandXML file with no layer grouping.*
if ticked there are 2 options.

Folder for DBXs file

The folder to create the databases in.

Simple Tab

The "**Simple**" tab allows a centreline and model of strings to be written to a single layer in the DBX.

Alignment string string select

The centreline for the layer.

Include alignment in layer tick box

if ticked the centreline is included in the layer, is cut as part of the road surface.

Stringlines model model box

The model of strings forming the road surface/layer.

Layer name

The default layer name is the model name, this can be changed here.

Advanced Tab

The "**Advanced**" tab allows multiple centrelines and strings to be written to layers in the DBX. For each alignment string there can be up to 10 layers, e.g. an entire subdivision can be uploaded by nominating the centreline for each road and adding the models for the final surface layers, boxing layers etc.

Alignment string

The centreline for the 10 optional layers.

Alignment in layer 1

If ticked the centreline is included in the 1st layer, is cut as part of the road surface.

Stringlines models 1

The model of strings forming the road surface/layer.

Layer name 1

The default layer name is the model name, this can be changed here.

For the rest of the option layer enter the model and layer name to suit.

To create extra roads/centrelines right click on the row number in the grid box to insert an extra row and fill out to suit.

Write button

Create the Leica 1200 Road DBX files.

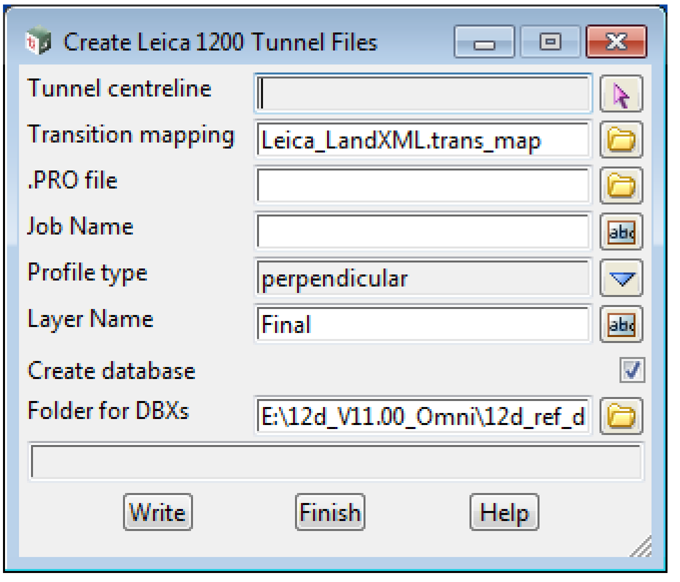
17.15.11 Create Leica 1200 Tunnel Files

Position of menu: **Survey =>Leica => 1200 =>Tunnel**

This panel creates a tunnel database for use on the Leica 1200 series of TPS instruments. This is done by creating an intermediate Leica LandXML file and then calling a Leica provided converter to create the tunnel DBX files.

This panel allows a single centreline and layer to be written to a tunnel DBX, it uses some industry standard tunnel definition files (.PRO,.PRA) to define the shape of profiles and the chainages they are applied at, these files are created via a text editor or other software packages.

Selecting **Tunnel** brings up the **Create Leica 1200 Tunnel Files** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tunnel centreline	string select		

The centreline for the tunnel.

Transition mapping file Leica_LandXML.trans_map

Due to the inadequate specification of transitions in the LandXML schema users must explicitly map 12d transition types to other vendors types.

.PRO file file

The profiles file containing the definition of the tunnel elements.

Create database

If ticked will create the 1200 tunnel DBX
if not ticked, just the Leica LandXML file is created.

Jobname

The internal jobname seen inside the Leica software.

Profile type choice box perpendicular perpendicular, vertical

perpendicular - the profiles are to be interpreted normal to the grade of the tunnel vertical alignment.
vertical - the profiles are not adjusted to the grade of the tunnel vertical alignment.

Layer name Final

The name of the tunnel DBX layer.

Create database tick box

if ticked will create the 1200 DBX
if not ticked just the LandXML file is created.

Folder for DBXs file

The folder to create the databases in.

Write button

Create the tunnel DBX files.

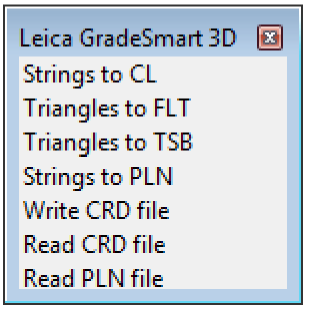
For more information on PRO/PRA tunnel definition files please see 20.12.10 Definition of the PRO and PRA definition files

17.15.12 Leica GradeSmart 3D

The **Leica GradeSmart 3D** walk-right menu collects under the one menu, options that can work with Leica GradeSmart software.

The *GradeSmart* options are all documented in the Leica GradeSmart manual.

.



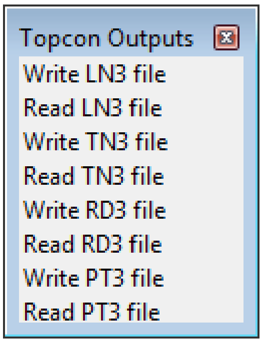
Triangles to TSB

The *GradeSmart* options are all documented in the Leica GradeSmart manual.

17.16 Topcon

The **Topcon** walk-right menu collects under the one menu, options that can work with Topcon instruments.

.



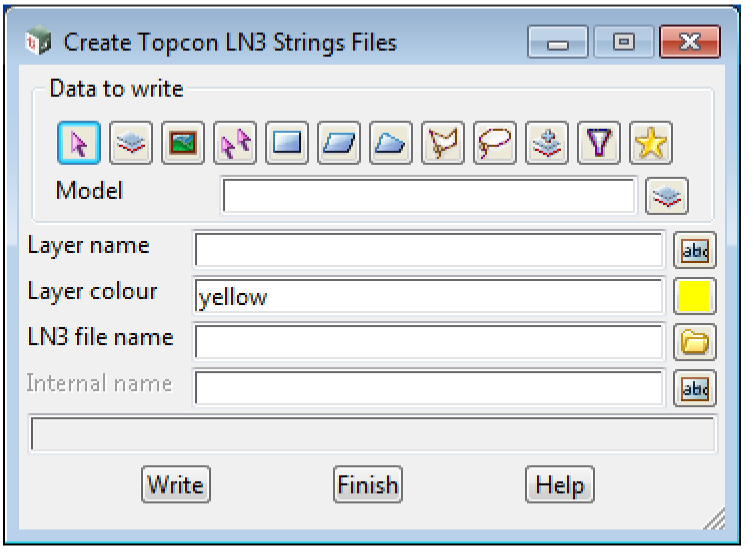
For the option *Write LN3 file* go to	17.16.1 Create Topcon LN3 String Files
Read LN3 file	17.16.2 Read Topcon LN3 Strings Files
Write TN3 file	17.16.3 Create Topcon TN3 Triangle Files
Read TN3 file	17.16.4 Read Topcon TN3 Triangle Files
Write *RD3 file*	17.16.5 Create RD3 Road File
Read RD3 file	17.16.6 Read Topcon RD3 Road File
Write *PT3 file*	17.16.7 Create Topcon PT3 Points Files
Read PT3 file	17.16.8 Read Topcon PT3 Points Files

17.16.1 Create Topcon LN3 String Files

Position of menu: **Survey =>Topcon=> write LN3 file**

This panel writes strings to Topcon LN3 linework files.

Selecting **write LN3 file** brings up the **Create Topcon LN3 String Files** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Data to write

Data source type model

data selection type - for a full description go to [4.19.3 Data Source](#) .

Data source model

The data source to write to the LN3 file.

Layer name

The name of the layer in the LN3 file.

Layer colour colour yellow

The colour of the strings in the Topcon LN3 file.
Please note that the LN3 format colour string by layer and hence individual string colours cannot be mapped to a LN3 file.

LN3 file name

The name of the LN3 file.

Internal name

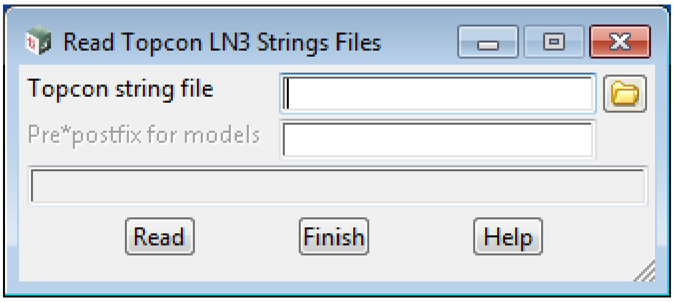
The name of the LN3 file you see inside the Topcon software.

17.16.2 Read Topcon LN3 Strings Files

Position of menu: Survey =>Topcon=> Read LN3 file

This panel reads in strings from the Topcon LN3 linework files.

Selecting **Read LN3 file** brings up the **Read Topcon LN3 String Files** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Topcon string file	file		

The LN3 file you wish to read.

Pre* postfix for models

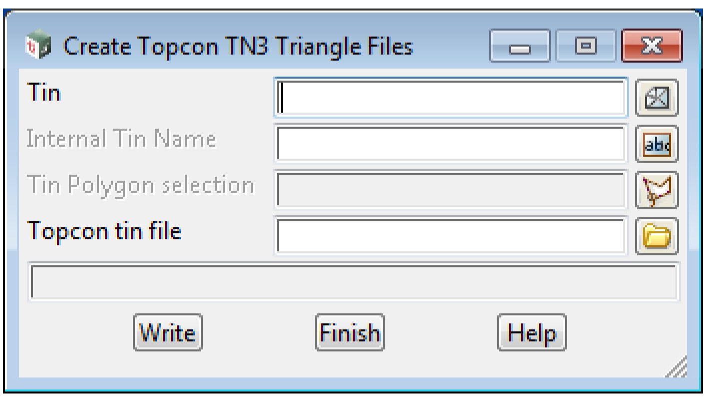
for information please go to 4.19.2 Pre*Postfix in Panel Fields

17.16.3 Create Topcon TN3 Triangle Files

Position of menu: **Survey =>Topcon=> Write TN3 file**

This panel writes triangles to the Topcon TN3 triangulation files.

Selecting **Write TN3 file** brings up the **Create Topcon TN3 Triangle Files** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	input		

The tin to write to the TN3 file.

Internal Tin name

The name of the tin you see inside the Topcon software.

Tin Polygon selection

Select a polygon for the region of the tin to write out.

Topcon tin file	file	

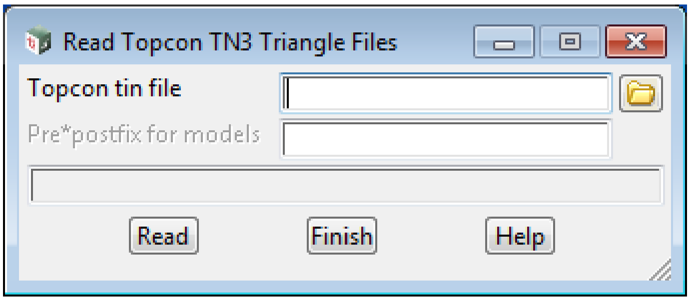
The name of the TN3 file to create.

17.16.4 Read Topcon TN3 Triangle Files

Position of menu: Survey =>Topcon=> Read TN3 file

This panel reads in triangles from the Topcon TN3 triangulation files.

Selecting **Read TN3 file** brings up the **Read Topcon TN3 Triangle Files** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Topcon tin file	file		

The TN3 file you wish to read.

Pre* postfix for models

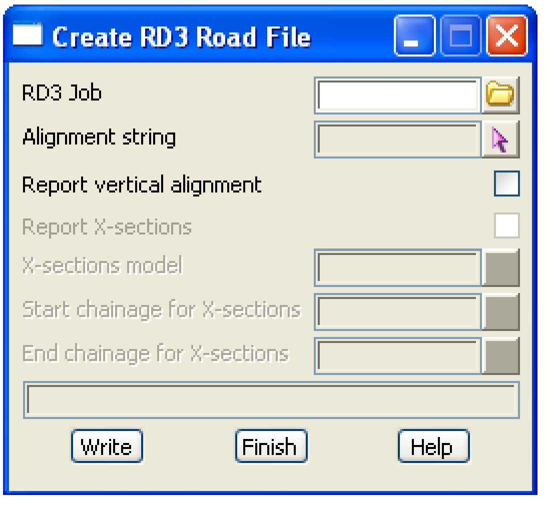
for information please go to 4.19.2 Pre*Postfix in Panel Fields

17.16.5 Create RD3 Road File

Position of menu: Survey =>Topcon=> Write RD3 file

This panel uploads the centreline and cross sections for a Topcon RD3 road definition file. The RD3 file can contain just a centreline, a centreline with vertical alignment or the full definition including cross sections.

Selecting **Write RD3 file** brings up the **Create RD3 Road File** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

RD3 Job file

The RD3 file you wish to create.

Alignment string string select

The centreline of the road.

Report vertical alignment tick box

If ticked the vertical alignment is uploaded to the RD3 file

Report X-sections tick box

If ticked cross sections are uploaded to the RD3 file

X-sections model

The model containing the cut cross section strings to upload to the RD3 file. These must have been cut normal to the nominated alignment string or the upload will fail. Start chainage for X-sections
If non blank the start chainage to upload cross sections from.

End chainage for X-sections

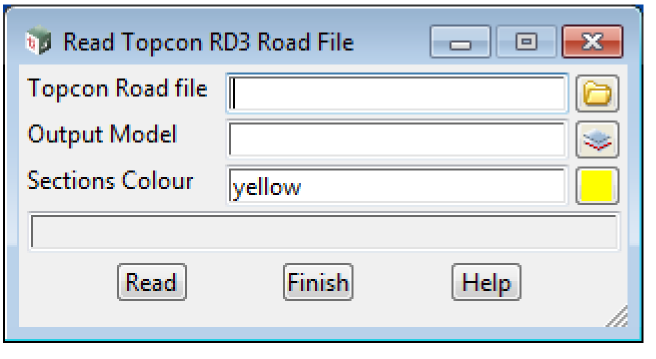
If non blank the end chainage to upload cross sections to.

17.16.6 Read Topcon RD3 Road File

Position of menu: **Survey =>Topcon=> Read RD3 file**

This panel reads in the centreline and cross sections from a Topcon RD3 road definition file.

Selecting **Read RD3 file** brings up the **Read Topcon RD3 Road File** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Topcon Road file

The RD3 file you wish to read.

Output Model model

The model for the centreline and cross sections.

Sections Colour colour box yellow

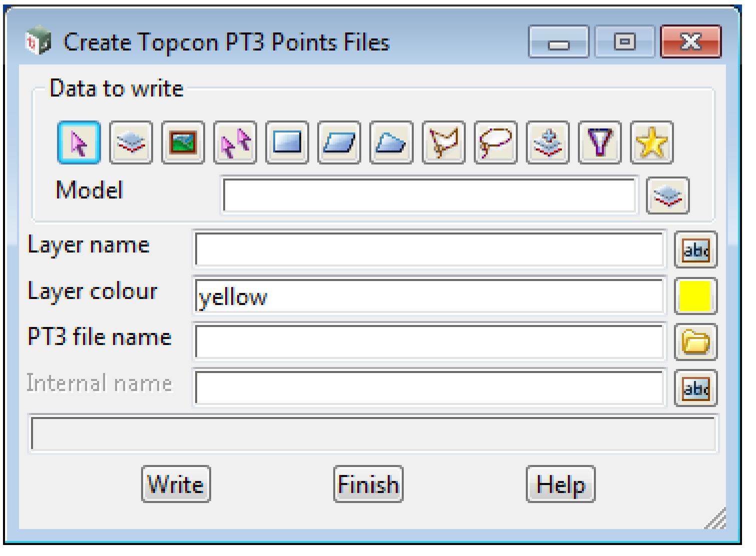
The colour for the cross section strings.

17.16.7 Create Topcon PT3 Points Files

Position of menu: **Survey =>Topcon=> Write PT3 file**

This panel writes strings to Topcon PT3 points files.

Selecting **Write PT3 file** brings up the **Create Topcon PT3 Points Files** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Data to write

The data to write to the PT3 file.

Data source type

data selection type - for a full description go to 4.19.3 Data Source .

Data source model

Layer name

The name of the layer in the PT3 file.

Layer colour select colour yellow available colours

The colour of the points in the Topcon PT3 file. Please note that the PT3 format colours points by layer and hence individual point colours cannot be mapped to a PT3 file.

PT3 file name file box

The name of the PT3 file.

Internal name

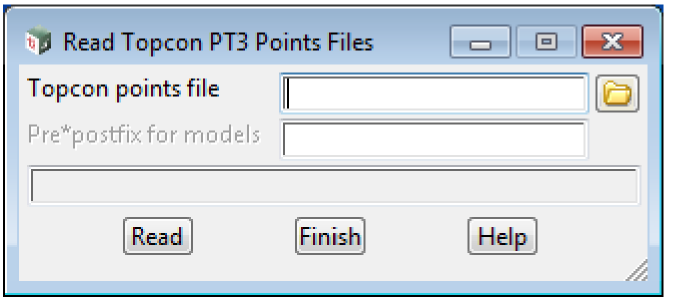
The name of the PT3 file you see inside the Topcon software.

17.16.8 Read Topcon PT3 Points Files

Position of menu: **Survey =>Topcon=> Read PT3 file**

This panel reads in points from the Topcon PT3 points files.

Selecting **Read PT3 file** brings up the **Read Topcon PT3 Points Files** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Topcon points file	file box		

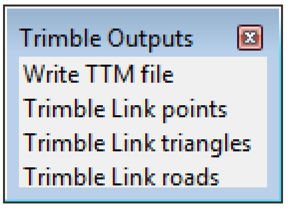
The PT3 file you wish to read.

Pre* postfix for models

for information please go to 4.19.2 Pre*Postfix in Panel Fields

17.17 Trimble

The **Trimble** walk-right menu is:



For the option *Write TTM file*, go to 17.17.1 Write TTM File

Trimble Link points 17.17.2 Trimble Link Points

Trimble Link triangles 17.17.3 Trimble Link Triangles

Trimble Link roads 17.17.4 Trimble Link Roads

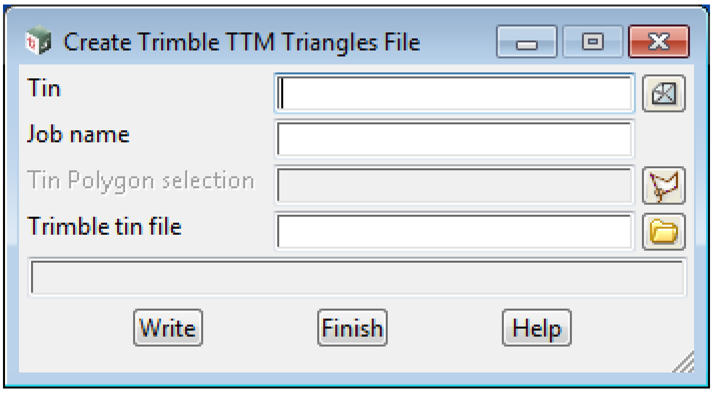
SCS900 CSV to XYZ ID Code 17.17.5 SCS900 CSV to XYZ ID Code

17.17.1 Write TTM File

Triangles from a tin (not a super tin) can be written out as a Trimble TTM file.

Note that the Trimble TTM is binary and so can not be viewed or edited.

Selecting **Write TTM file** brings up the **Create Trimble TTM Triangles File** panel:



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

the tin (not a super tin) to write out to a Trimble TTM file. A super tin can not be written out to a TTM file.

Job name text input

name of the job to write into the Trimble TTM file

Tin Polygon selection polygon box

if a string is selected, then only those triangles whose centroid is inside the polygon are written to the TTM file.

If no string is selected, then all the triangles in the tin are written to the TTM file.

Trimble tin file file box

name of the Trimble TTM file. If it is not present, ".TTM" will be automatically added to the end of the file name.

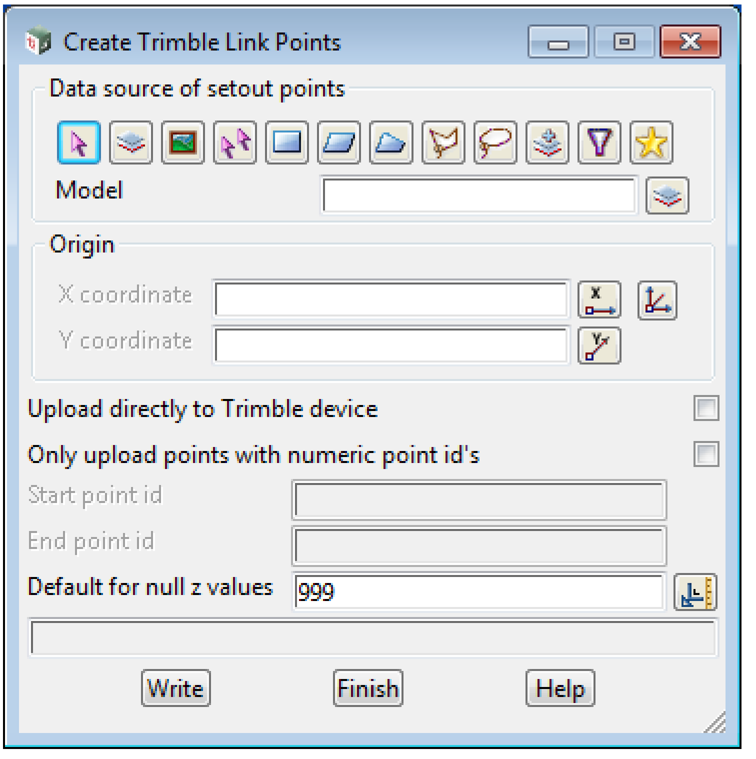
Write button

write out the triangles to the given TTM file.

17.17.2 Trimble Link Points

This option uploads selected vertices with point IDs to a Trimble device or writes the (X, Y, Z, point ID) data to a file.

Selecting **Trimble Link points** brings up the **Create Trimble Link Points** panel:



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Data source of setout points

data selection type - for a full description go to 4.19.3 Data Source .

Data source

data source of setout points to create the upload data from.

Origin X coordinate/Origin Y coordinate

If non-zero, subtract the value from the x/y value before uploading/writing out.

Upload directly to Trimble device tick box not ticked

if ticked, the selected points are uploaded to the Trimble device.
If not ticked, the data is written to a Trimble DC file with the name as the 12d Model project and ending in .DC

Only upload points with numeric point id’s tick box not ticked

if ticked, only vertices with numeric point IDs are uploaded to the Trimble device or written out.
If not ticked, all vertices (whether numeric or alpha-numeric point IDs) are uploaded to the Trimble device or written out.

Start point id input box

point ID to start creating upload data from.

End point id input box

last point ID to create upload data to.
Note: Start and End point IDs may be left blank, and all point IDs are uploaded.

Default for null z values measure box 999 *value to write for null z-values.*

Write button

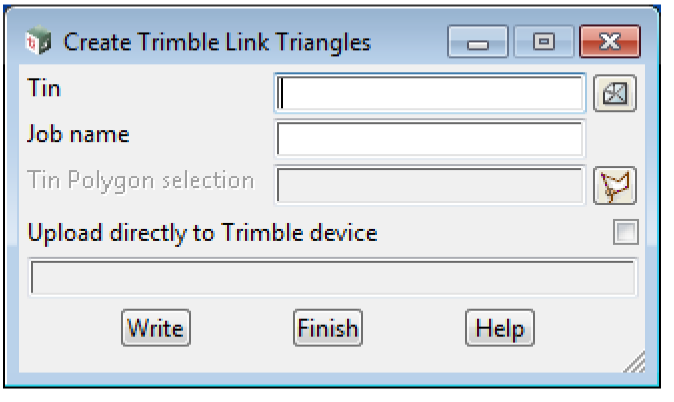
send the vertices directly to the Trimble device or write them to a DC file.

17.17.3 Trimble Link Triangles

Triangles from a tin (not a super tin) can be uploaded to a Trimble device or written out as a Trimble TTM file.

Note that the Trimble TTM is binary and so cannot be viewed or edited.

Selecting **Trimble Link triangles** brings up the **Create Trimble Link Triangles** panel:



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Tin tin box available tins

the tin (not a super tin) to either upload directly to the Trimble device or write out to a Trimble TTM file. A super tin can not be written out to a TTM file.

Job name text input

name of the Trimble job

Tin Polygon selection polygon box

if a string is selected, then only those triangles whose centroid is inside the polygon are processed. If no string is selected, then all the triangles in the tin are processed.

Upload directly to Trimble device tick box not ticked

if ticked, the selected triangles are uploaded to the Trimble device.
If not ticked, the data is written to a Trimble TTM file with the given job name and ending in .TTM

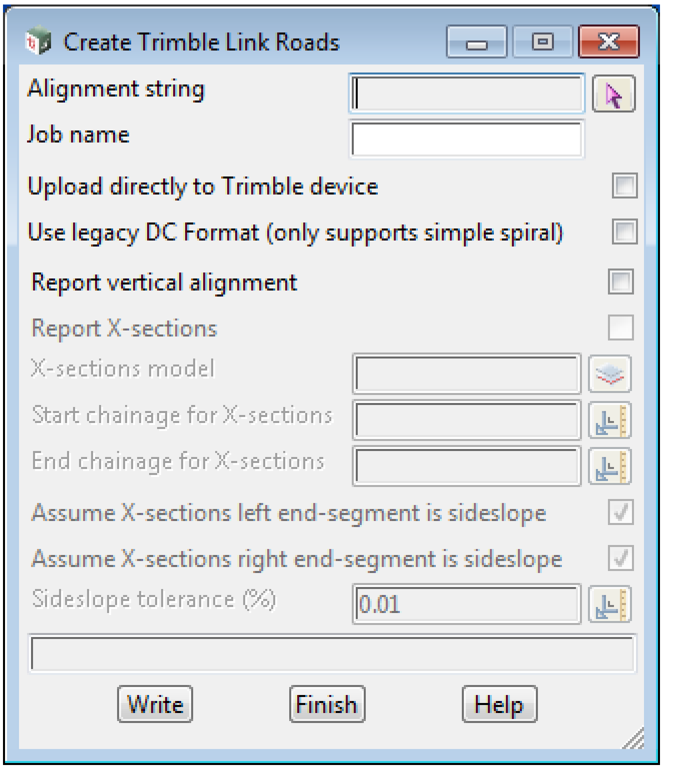
Write button

send the triangles directly to the Trimble device or write them to the given TTM file.

17.17.4 Trimble Link Roads

This option uploads an alignment string and associated x-sections to a Trimble device, or writes the data to a specific roading file.

Selecting **Trimble Link roads** brings up the **Create Trimble Link Roads** panel:



The Trimble upload/DC file has certain limitations and criteria. These include:

1. The alignment string must exist on the X-sections. This allows us to calculate the ***zero pt*** so we can split the template into left and right.
2. The X-sections model should be representative of the alignment string, *i.e.* not generated from a different alignment. Therefore the alignment string should have a 0 offset on the xsections.
3. The alignment string cannot exist above or below the X-sections, *i.e.* it should be on the section as per 1.
4. The number of points on successive section templates (left and right) should be the same, *i.e.* if 5 points are on the LHS template for ch0 then the same number of points should exist for the LHS template for ch20. The instruments using this file, may not handle transitions between different number of points effectively and can give incorrect results. The user should limit the upload data by using the chainage range between areas where the number of points on each side of the X-sections are the same.

The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Alignment string	input		

the alignment string to be uploaded/written to the file.

Job name	input		

name of the Trimble job

Upload directly to Trimble device	tick box	not ticked	

if ticked, the selected data is uploaded to the Trimble device.
If not ticked, the data is written to a Trimble DC file with the given job name and ending in .DC

Use legacy DC Format	tick box	not ticked	
Report vertical alignment	tick box	not ticked	

if ticked, include vertical alignment details in the upload/file.

Report X-sections tick box not ticked

if ticked, the X-sections in **X-section model** *will be uploaded/output to the file.*

X-sections model model box

the model of X-sections to be uploaded/written to the file. This can only be filled in if **Report vertical alignment** *and* **Report X-sections** *are ticked.*

Start chainage for X-sections measure box

if non blank, only X-sections whose chainage is greater than or equal to the **Start chainage for X-sections and** *less than or equal to the* **End chainage for X-sections** *are uploaded/written out.*
If blank, all X-sections that are less than or equal to the **End chainage for x-sections** *are uploaded/written out.*

End chainage for X-sections measure box

if non blank, only X-sections whose chainage is less than or equal to the **End chainage for X-sections and** *greater than or equal to the* **Start chainage for X-sections** *are uploaded/written out.*
If blank, all X-sections that are greater to or equal to the **Start chainage for X-sections** *are uploaded/written out.*

Assume X-sections left end-segment is sideslope tick box ticked

if ticked, the first segment is considered to be a sideslope, and the **Sideslope tolerance (%)** *field is enabled.*

Assume X-sections right end-segment is sideslope tick box ticked

if ticked, the last segment is considered to be a sideslope, and the **Sideslope tolerance (%)** *field is enabled.*

Sideslope tolerance (%) measure box 0.01

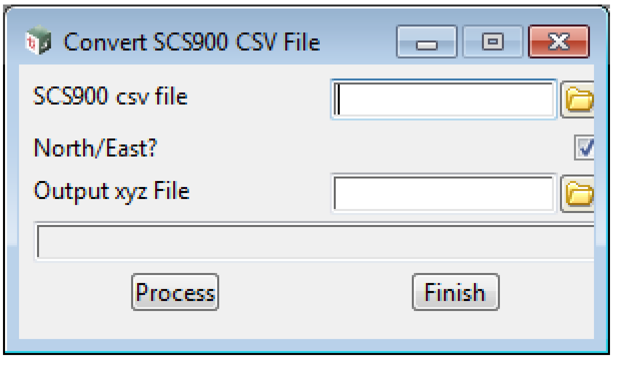
Write button

send the data directly to the Trimble device or write it to the given DC file.

17.17.5 SCS900 CSV to XYZ ID Code

This option converts a Trimble SCS900 CSV file to an xyz file.

Selecting **SCS900 CSV to XYZ ID Code** brings up the **Convert SCS900 CSV File** panel:



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

SCS900 csv file file box *.csv files

name of the SCS900 CSV file

North/East? tick box ticked

if ticked,

Output xyz File file box *.xyz files

name of the xyz file to convert the SCS900 CSV file to

Process button

convert the SCS900 file to a xyz file

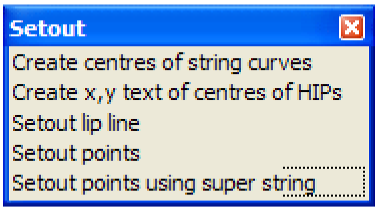
17.18 Setout

Position of menu: **Survey =>Setout**

The **Setout** walk-right menu contains options for creating points numbers and files for uploading to data recorders.

The **Setout** walk-right menu is

.



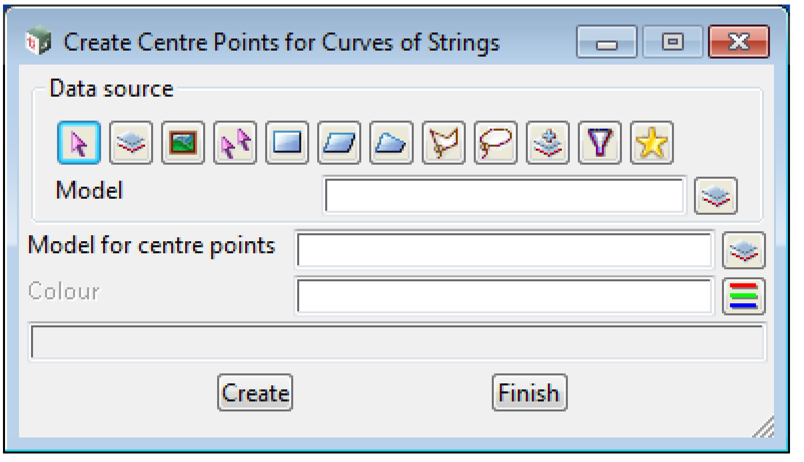
setout centres of string curves
create text at centre of HIP's
setout lip line
create setout points
create setout points using super strings

For *Create centres of string curves*, go to 17.18.1 Create Centre Points for Curves of Strings
Create x,y,z,text of centres of HIPs 17.18.2 Create X Y Text for Centre Points of Alignments
Setout lip line 17.18.3 Setout Lip Line .
Setout points 17.18.4 Create Setout Points .
Setout points using super string 17.18.5 Create Setout Points Using Super String .

17.18.1 Create Centre Points for Curves of Strings

Position of option on menu: **Survey =>Setout =>Create centres of string curves**

This option is used to create points at the centres of arcs in alignment strings, polylines, arcs or circles.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Data source type	Model		

data selection type - for a full description go to 4.19.3 Data Source .

Data source

data source for strings to process

Model for centre points	input box		

model for the created points

Colour	colour box		available colours

colour of centre points

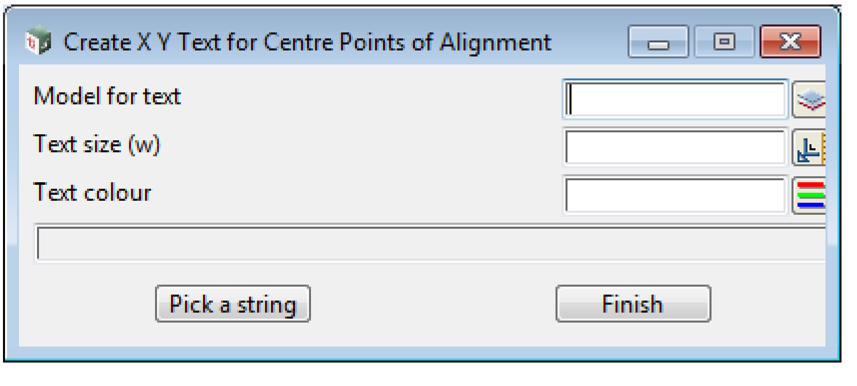
Create	button		

create the points

17.18.2 Create X Y Text for Centre Points of Alignments

Position of option on menu: **Survey =>Setout =>Create x y text of centres of HIPs**

This option creates text for the x and y co-ordinates of the horizontal arcs on an alignment string.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Model for text	input box		

model for the text of the x and y co-ordinates

Text size (w) input box

size of text in world units.

Text colour input box

colour of text

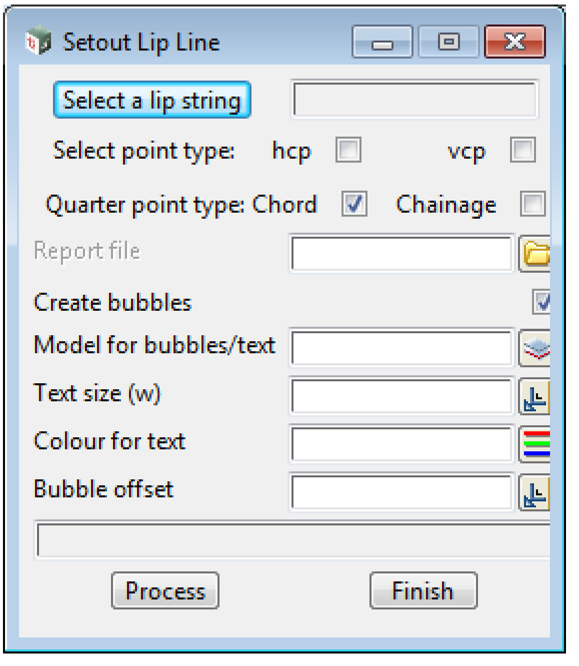
Pick a string button

select string to create text for.

17.18.3 Setout Lip Line

Position of option on menu: **Survey =>Setout =>Setout lip line**

This panel is used to create bubbles and/or a report for the critical horizontal and vertical points and quarter points (by chord or by chainage) for any arcs in an alignment string.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Select a lip string	string select		

pick the alignment string to have bubbles created for.

Select point type:

hcp	tick box		

if ticked, the horizontal tangent points are included.

vcp	tick box		

if ticked, the vertical tangent points are included.

Quarter point type:

Chord	tick box	tick	

if ticked, the quarter points by chord distance are included.

Chainage	tick box		

if ticked, the quarter points by chainage distance are included.

Report file	input box		

name of the report file.

Create bubbles	tick box	tick	

if ticked, bubbles are created with the string name and bubble number inside.

Model for bubbles/text	input box		

model for the bubbles and text.

Text size (w) input box

size (in world units) for the text inside the bubble.

Colour for text input box

Colour for the bubble and bubble text.

Bubble offset input box

offset distance from the alignment string to place the bubble.

Process button

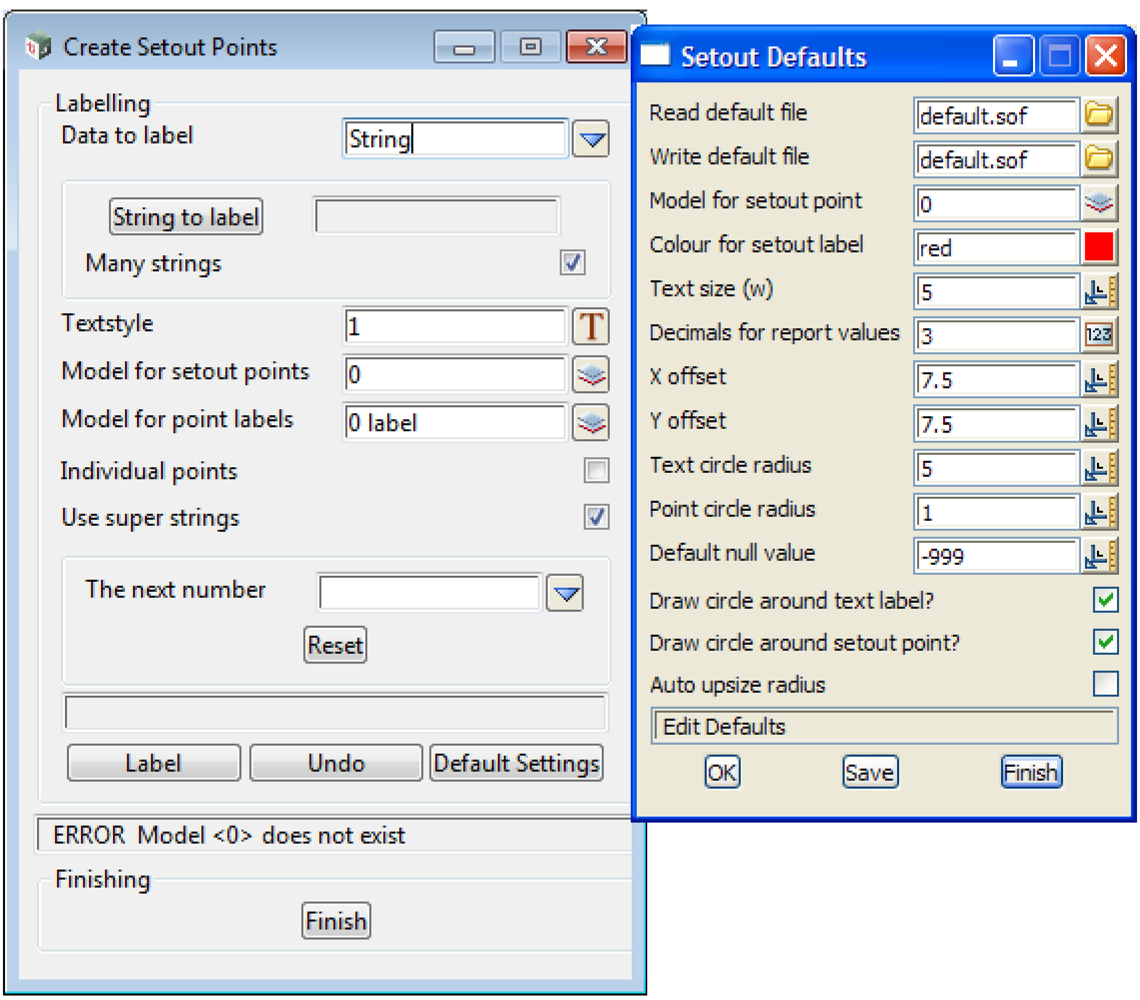
run the option.

17.18.4 Create Setout Points

Position of option on menu: **Survey =>Setout =>Setout points**

This panel is used to create points with numbers for use in setting out data. This option is usually run before the option to create an instrument upload file. The setout point is created as a 4d string with the point at the data point and with the point id as the text for the 4d string, with a text size of zero.

Separate text for the numbers and circles surrounding the data points and the text numbers are also created. By default, a file called "defaults.sof" is read in containing settings required for the option.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Textstyle	input box	1	

textstyle for the created point ids

Model for setout points	input box	0	

model to place the setout points in

Model for points labels	input box	0	

model to place the setout point ids and circles in.

Individual points	tick box		

if ticked, the setout points are created as point strings
If not ticked, individual one point strings are created.

Use super strings	tick box	tick	

if not ticked, super strings are created for the setout points. The super string point id is set to the setout point id.
If not ticked, 4d strings are created for the setout points strings with the 4d text as the setout point id (with text size set to zero).

The next number	input box	1 or highest number in model for setout points	

*next numeric point id to use for creating point ids. If a **Model for setout points** is given, the model is searched for any super strings or 4d strings with numbers as text and the **The next number** is set to one more than the highest numeric point id in the model.*

Reset	button		

*reset the **The next number** to the highest point in the model for setout points.*

Data to label	input box	string	string, model, view, point

type of data source.

Model/View/String/Point to	output box		

data to create point ids for.

Many points/strings	tick box	tick	

if ticked, the each time a string/point is selected, the string/point is processed without having to click on the **Label** *button. After a string/point is processed, another string/point can then be selected.*
If not ticked, the **Label** *button must be clicked before the selected string/point is processed.*

Label	button		

create setout points and circles for the selected data.

Undo	button		

undo the last setout points and s created whilst the panel has been up.

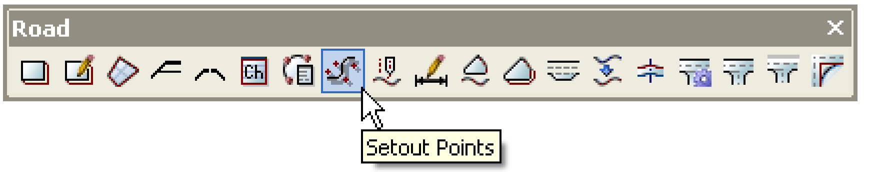
Default settings	button		

default settings for the s. These can be read in from a file.

17.18.5 Create Setout Points Using Super String

Position of option on menu: **Survey =>Setout =>Setout points using super string**

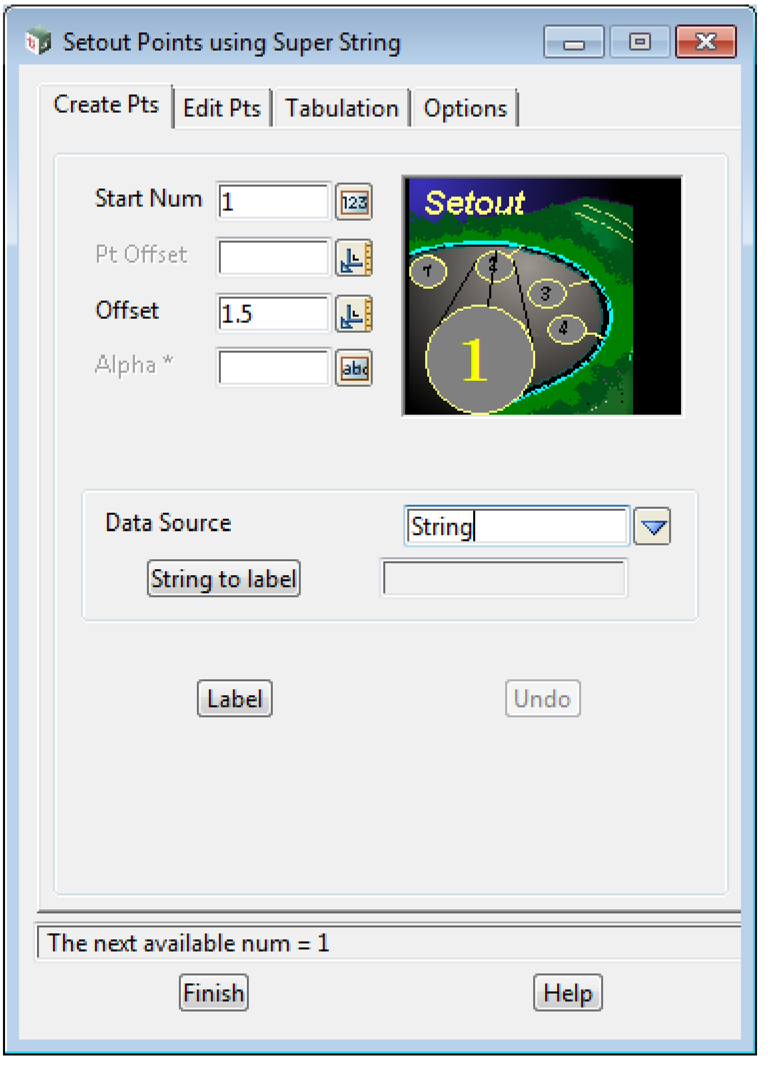
Position of option on menu: **Road Toolbar**



This option creates a super string over the top of the data selected.The super string allows point numbers. The incremental point numbers are displayed at each vertex inside a circle symbol.

The setout numbers are identified on a setout model and incremented each time the panel is activated.

.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Create Pts

Start Num integer 1 (or next available number on subsequent runs of the option)

Type in number and <enter> or accept default

Pt Offset real

Offset of the setout point, typically a distance behind the back of kerb.
(This will be the setout point x,y location but the z value refers to the original string selected e.g. the kerb return)

Offset real

Offset of the setout point text and circle, from the actual setout point.

Alpha* text

If non-blank, Alpha point number prefix is used

Data Source choice box String String, Point, Model, View

String: *Allows the selection of a string (using 12d pick with direction), where the setout number is always placed on the right hand side in relation to the pick direction, perpendicular to the string. Alignment choices: A selection panel is displayed with 3 choices:-*

- by number <divides the string into equal parts>
- by distance <divides the string and may have a remainder>
- by special chainage file

String to label button

Point: *Allows the selection of a point. There is no direction involved.*

Point to label button

Model and View*: Allows the setout of multiple strings and points but again there is no control over the pick direction or the orientation of the setout number in relation to any string.*

Model to label model

existing model selection required

View to label view

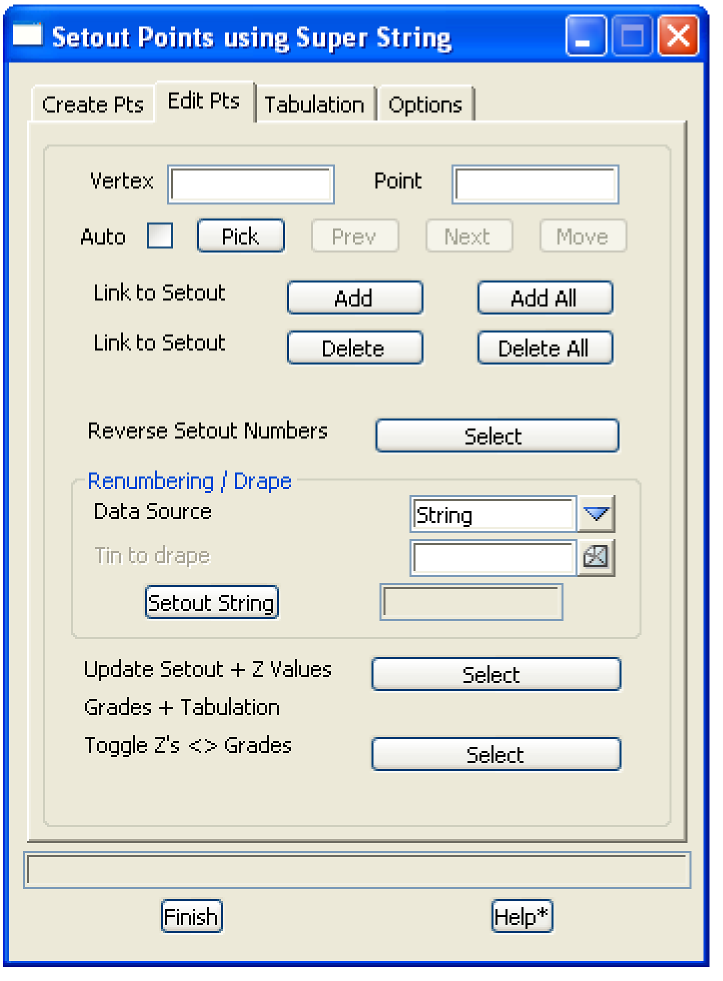
Existing view to be selected

Label button

Runs the option and adds the setout model to the view from which the Data Source was selected

Undo button

Each <Label> creation has an undo



The fields and buttons used in this panel have the following functions:

Field Description | Type | Defaults | Pop-Up

Edit Pts

Vertex Integer

Point Integer

The Vertex and Point fields above are automatically filled in after a setout string has been picked

Auto tick box false

When ticked on, and a setout point is selected using <Pick>, then the <Move> option is automatically activated

Pick button

Select setout point to move text bubble

Prev button

Steps backward and highlights the previous point on setout string

Next button

Steps forward and highlights the next point on setout string

Move button

Activates the move option allowing the setout text bubble to be dynamically relocated on screen, but still remain attached to the actual setout point

Link to Setout:

Add button

Draws a leader line from the actual setout point to the setout text value bubble

Add All button

Draws a leader line from the actual setout point to the setout text value bubble at all vertices on the selected setout string

Link to Setout:

Delete button

Deletes the leader line from the actual setout point to the setout text value bubble, if it exists

Delete All button

Deletes the leader line from the actual setout point to the setout text value bubble at all vertices on the selected setout string

Reverse Setout Numbers:

Select button

This option reverses the numbering order only on a setout string.

Renumbering / Drape:

This option can renumber the start number used on setout data. When a string or model is selected a prompt for a new start number is displayed.

The tin to drape is used to reset the z value of the setout point. A design tin can be used, but care should be taken to have a design point in the tin, close to the setout point, as a "drop onto the tin" is used to calculate the new z value.

Data Source choice box String String, Model

String or model are the choices

Tin to drape tin

if non-blank, setout vertices will be draped to get new z value

Setout String button

*Select the setout string to perform the above **Renumbering / Drape***

Update Setout + Z Values + Grades + Tabulation:

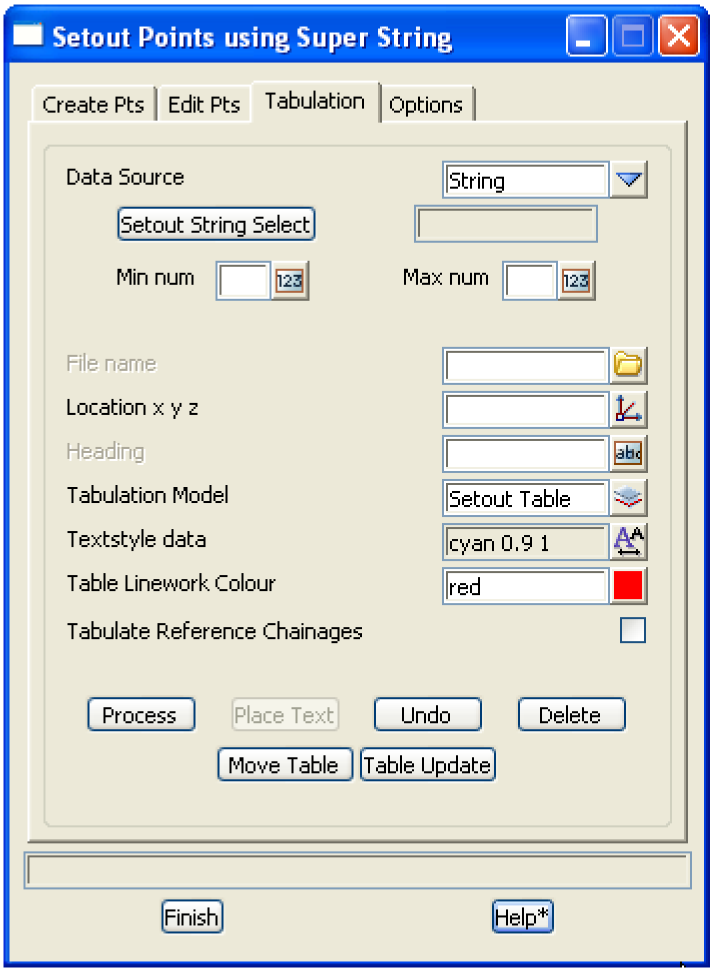
Select button

Setout features can be updated as the original string selected for setout (e.g. Kerb return alignment) has been stored as an attribute on the setout string.Care must be taken to re-select the original string.

Toggle Z's <> Grades:

Select button

Setout features such as labelling Z values and grades on segments are discussed under Options *tab. This option toggles the display between the two*



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up

Tabulation

Data Source — choice box — String — String, Model

String or model are the choices

Setout String Select — button

Select a setout string to tabulate

Model to Tabulate — model

Select existing setout string to tabulate

Min num — integer

Max num — integer

Fields are filled out once setout data is selected, but can be changed manually.

File name — file

If non-blank an xyz file will be created

Location xyz — xyz pick

Select on screen or enter a coord for the top left hand corner of table

Heading — Input

if non-blank entry used as heading for tabulation

Tabulation Model model available models

Enter a model name for the tabulation

Textstyle Data textstyle box select Textdata

Setup text colour, height and width

Table Linework Colour colour box available colours

Colour for any linework in tabulation

Tabulate Reference Chainages tick box false

Chainages can be displayed from the original selected string (e.g. Reference kerb return alignment) in relation to setout vertices

Process button

runs the option

Place Text button

Creates the tabulation on the tabulation model

Undo button

Undo for that operation

Delete button

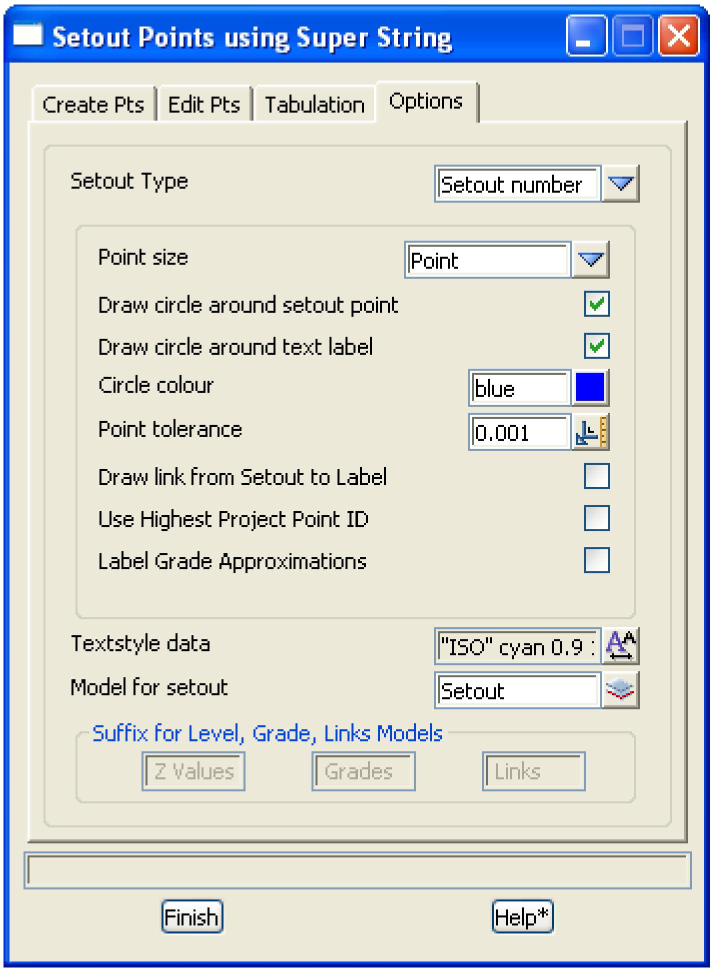
Deletes all the entities on the table selected

Move Table button

Moves the complete table on screen (select table and destination point)

Table Update button

Table can be updated if text parameters are changed or Reference Chainages is ticked on or off. Select the tabulation.



Options

Setout Type choice box Setout number Setout number, Z value

Setout number or Z values are the choices

Setout Number:

Point size choice box Point Point, Point 0.5

Selections are Point and Point 0.5 (refer to the size of the point shown at the setout z, y)

Draw circle around setout point tick true

Refers to the linestyle circle at each setout z, y.

Draw circle around text label tick true

Refers to the circle surrounding each setout point number text value

Circle colour colour box blue

The colour of the circle surrounding each setout point number text value

Point tolerance real

X,Y Tolerance between setout points to handle duplicate or near duplicate points

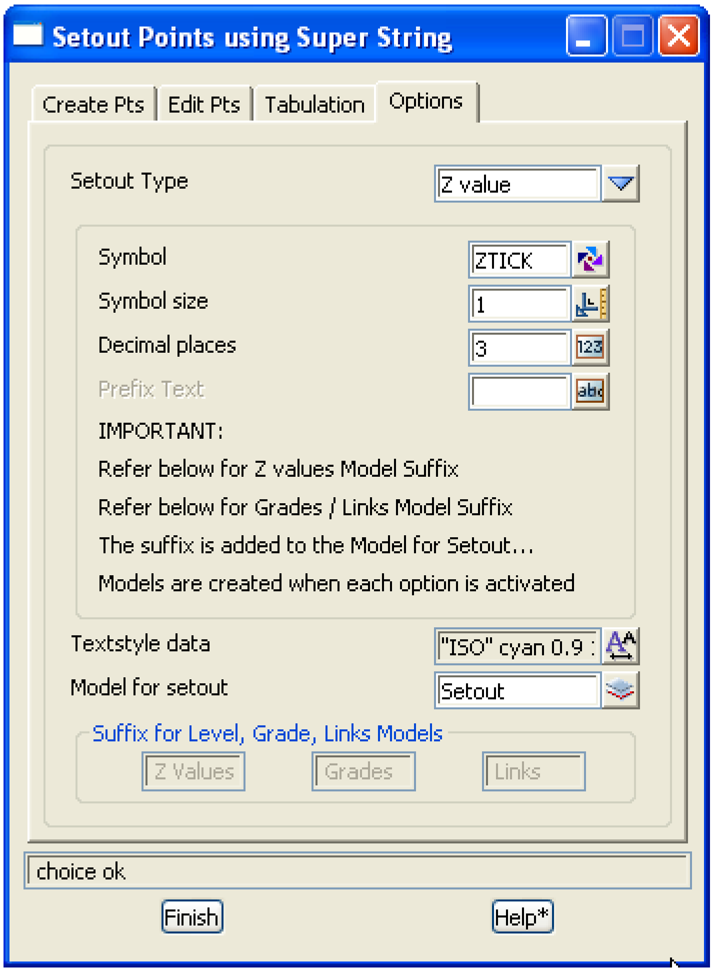
Draw link from Setout to Label tick box false

Use Highest Project Point ID integer

Search for the highest point number available in the project

Label Grade Approximations tick box false

(Grades can be labelled between setout points, but care must be taken as the z value is used from the setout points, the length from the original reference string selected.
The grade is only correct if the points are on grade and or on a parabola. So the grades are reported as approximate only and for a visual check.)



Z Value:

Symbol input ZTICK

Symbol Size real 1

Decimal Places Integer 3

Prefix Text input

if non-blank text value added to text Z value. <e.g. R.L.>

Textstyle Data textstyle box

Setup text style, colour, height and width

Model for setout model box available models

Enter a model name for the setout data

Suffix for Level, Grade, Links Models

Display default values for models that may be created if you choose to draw link lines, label levels or grades. If setout model is "Setout"....then these models would default to "Setout Level" "Setout Link" "Setout Grade"

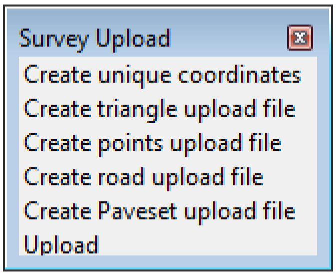
17.19 Upload

Position of menu: **Survey =>Upload**

The **Survey Upload** walk-right menu contains survey options to upload data the most survey instruments. There are options to upload points, triangulations and road geometry.

The **Survey Upload** walk-right menu is

.



Create triangle upload file
Create points upload file
Create roads upload file
Create points upload file (new) - only for *Trimble Link*
Create roads upload file (new) - only for *Leica XML* and *Trimble Link*
Upload data to instrument/device

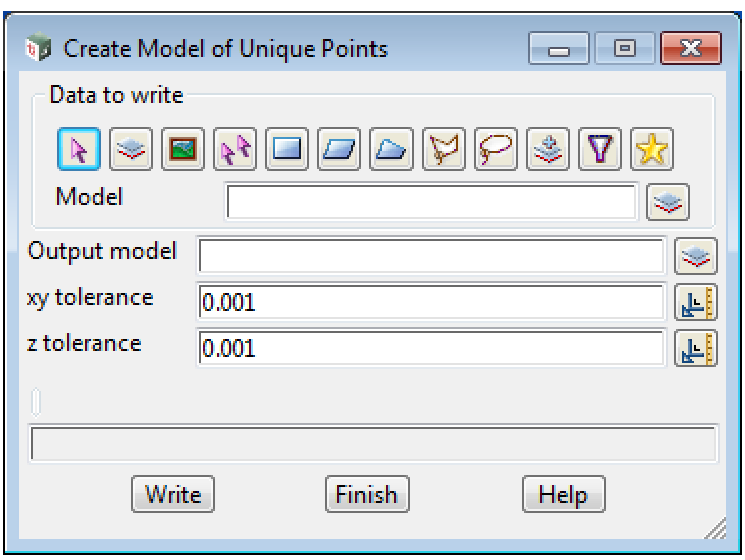
For the option *Create unique coordinates*, go to	17.19.1 Create Unique Coordinates
Create triangle upload file	17.19.2 Create Triangle Upload File
Create points upload file	17.19.3 Create Points Upload File
Create road upload file	17.19.4 Create Road Upload File
Create points upload file (new)	17.19.5 Create Points Upload File (New)
Create road upload file (new)	17.19.6 Create Road Upload File (New)
Create Paveset upload file	17.19.7 Create Paveset Upload File
Upload	17.19.8 Survey Data Upload

17.19.1 Create Unique Coordinates

Position of option on menu: **Survey =>Upload => Create unique coordinates**

This option creates points from the selected data so that there is only one point for all vertices within user defined xy and z tolerances.

Selecting **Create unique coordinates** brings up the **Create Model of Unique Points** panel.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Data source of setout points

data selection type - for a full description go to 4.19.3 Data Source.

Data source

data source of strings to create unique points from

Output model model box

model for the created vertices.

xy tolerance real value box -0.001

z tolerance real value box -0.001

all vertices closer than these tolerances are considered to be the same vertex and only one point will be created for all of them.

Process button

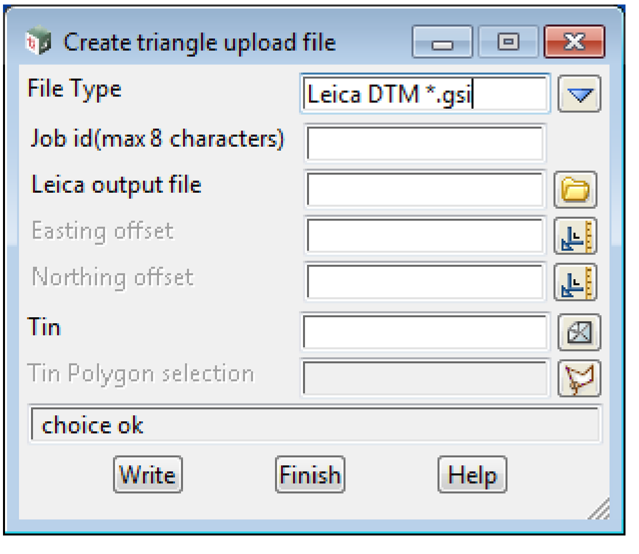
create the distinct points

17.19.2 Create Triangle Upload File

Position of option on menu: **Survey =>Upload => Create triangle upload file**

The **Create triangle upload file** option allows the creation of a triangle file based on a tin. Various file formats can be selected

Selecting **Create triangle upload file** brings up the **Create triangle upload file** panel



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Common parameters at bottom:

Tin input

name of tin from which the triangle file will be based.

Tin polygon selection selection

allows the user to select a specific area of the tin to be exported as a triangle file.

Write button

On clicking the write button, **12d Model** *will write the specified file.*

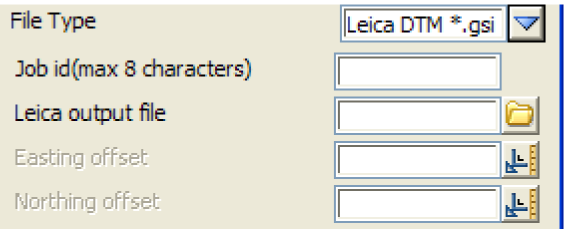
File type choice box

Leica DTM *.gsi
12d xyzs *.dat
Ortho 12d *.T12
Sokkia *.tri
TP Setout/Stakeout *.tsa
LandXML *.xml

file type to be produced.

***For* Leica DTM *.gsi format:**

.



Job id input

job id to be included in file. (max 8 characters)

Leica output file input

name of file to be produced

Easting offset input

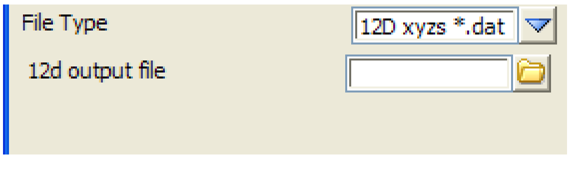
easting offset to be applied to data. This is important where large coordinate values are used, such as in map projections.

Northing offset input

northing offset to be applied to data. This is important where large coordinate values are used, such as in map projections.

***For* 12D xyzs *.dat format:**

.

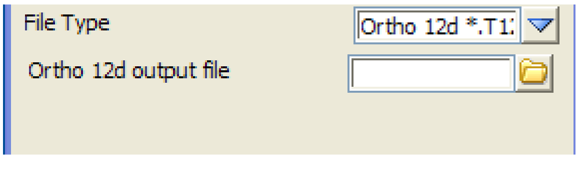


12d output file input

name of file to be produced

***For* Ortho 12d *.T12 format:**

.

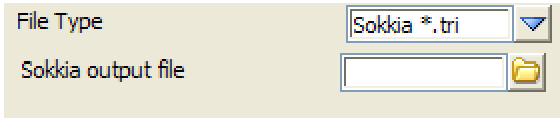


Ortho 12d output file input

name of file to be produced

***For* Sokkia *.tri format:**

.

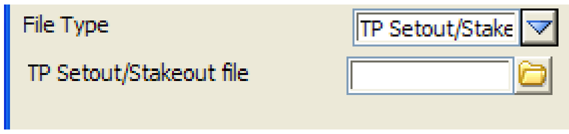


Sokkia output file input

name of file to be produced

***For* TPSetout/Stakeout *.tsa format:**

.

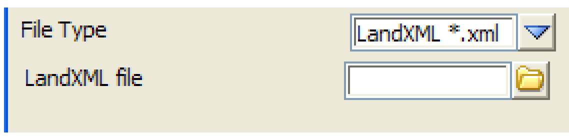


Tp Setout/Stakeout file input

name of file to be produced

***For* TPSetout/Stakeout *.tsa format:**

.



LandXML file input

name of file to be produced

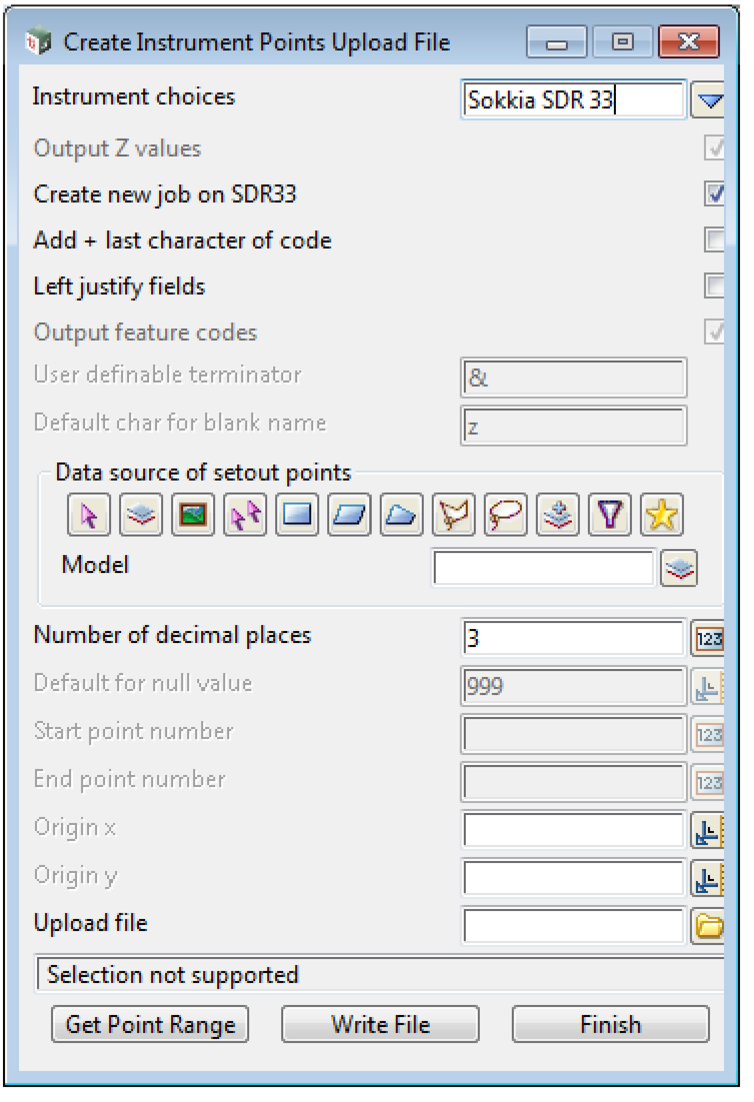
17.19.3 Create Points Upload File

Position of option on menu: **Survey =>Upload => Create points upload file**

The **Create points upload file** option allows the creation of a file of point ids, x, y and z, for string data (super strings and 4d strings only). Various file formats can be selected

Selecting **Create points upload file** brings up the **Create Instrument Points Upload File** panel.

Upload files for points can be created for various Leica, Geodimeter, Sokkia, Topcon, UPL, QuikDraw, Trimble XYZ formats as well as a text file of point id, x,y,z data.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Instrument choices	choice box	report_xyz	available instruments

type of output format

Output z values tick box

if ticked, z-values are output - only used for Geodimeter formats

Create new job on SDR 33 tick box

if ticked, create a new job on SDR33 - only used for Sokkia formats

Left justify fields tick box

if ticked, each field is left justified - only used for Sokkia formats

User definable terminator input box &

terminator to use - only used for Geodimeter formats.

Default char for blank name input box z

character used for blank names - only used for Geodimeter formats.

Data source of setout points

data selection type - for a full description go to 4.19.3 Data Source.

Data source

data source of setout points to create the upload file from.

Number of decimal places input box 3

number of decimal places for x, y and z values.

Default for null value input box -9999

value to write for null z-values.

Start point id input box

point id to start creating upload file from.

End point id input box

last point id to write to upload file.

Note: Start and End point ids may be left blank, and all point ids (whether numeric or alpha-numeric) will be written to the upload file.

Origin x/Origin y input box

If non-zero, subtract the value from the x/y value before writing out.

Upload file file box

name of the upload file to create.

Get Point Range button

pressing this button will populate the Start and End point id fields with the minimum and maximum (numeric) point ids found in the selected source data. If the selected source data contains a mix of numeric and alpha-numeric point ids, the alpha-numeric point ids are considered to have a numeric value of zero (0), but are ignored when determining the range. If the selected source data contains alpha-numeric point ids only, then the Start and End point id fields will both be set to zero (0).

Write File button

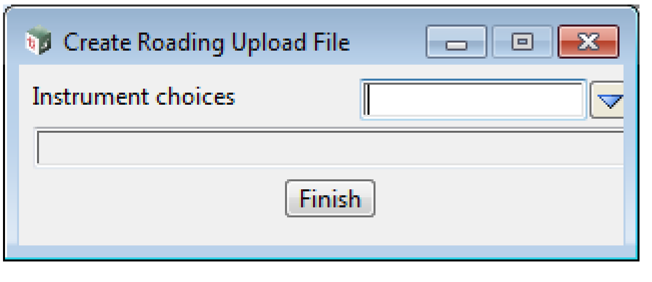
create the upload file

17.19.4 Create Road Upload File

Position of option on menu: **Survey =>Upload => Create roads upload file**

The **Create roads upload file** option allows the creation of a specific roading file. Files can be created for various Leica (8 & 16 formats), Geodimeter, Sokkia, Topcon and Trimble Roading formats.

Selecting **Create road upload file** brings up the **Create Roading Upload File** panel



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Instrument choices	choice box		Geodimeter Roadline 3d Leica Road Plus 8

Leica Road Plus 16
LMGS - D45
Sokkia Roading - Alignment Road
Sokkia Roading - String Road
Topcon MS2000
Topcon GTS-700 Roads
Trimble Roading

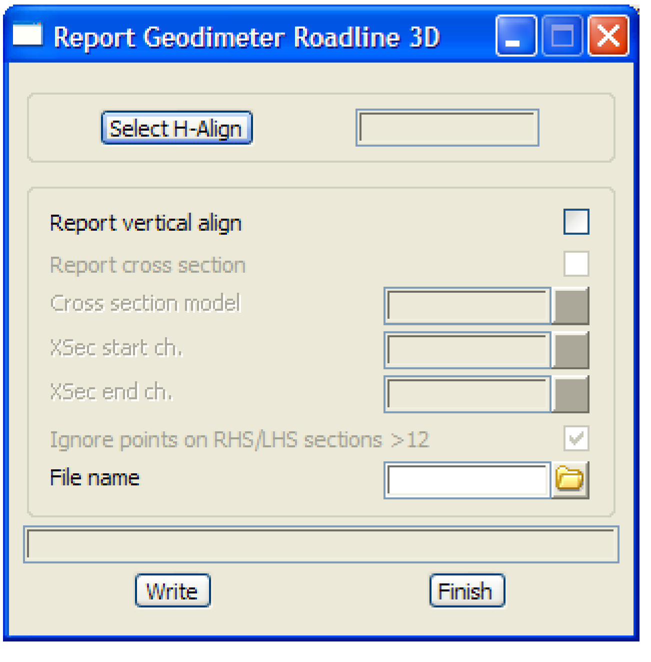
For choice of *Geodimeter Roadline 3d*, go to Geodimeter Roadline 3D file
Leica Road Plus 8 Leica road plus 8 / 16 file
Leica Road Plus 16 Leica road plus 8 / 16 file
LMGS - D45 file LMGS - D45 file
Sokkia Roading - Alignment road Sokkia Roading - Alignment Road file
Sokkia Roading - String road Sokkia Roading - String Road file
Topcon MS2000 Topcon MS2000 Roading file
Topcon GTS-700 Roads Topcon GT700 Roading File
Trimble Roading Trimble Roading File

Geodimeter Roadline 3D file

Position of option on menu: Survey =>Upload => Create roads upload file, Instrument choice = Geodimeter Roadline 3d

The **Geodimeter Roadline 3d** option allows the creation of a specific roading file.

Selecting **Geodimeter Roadline 3d** brings up the **Report Geodimeter Roadline 3D** panel



The Geodimeter (trimble), has certain limitations and criteria associated with the 3d file. These include:

1. Each section template (i.e. right or left) has a max number of 12 points.
2. The first and last segments of the alignment must be a straight (for both horizontal and

vertical).

3. The start and end chainages of the horizontal alignment should be the same for the vertical alignment.
4. The alignment string must exist on the xsec's. This allows the calculation of the ***zero pt*** so we can split the section into left and right templates.
5. The xsec model should be representative of the alignment string used in the macro. i.e. not generated from a different alignment. Therefore the alignment string should have a 0 offset on the xsections.
6. The alignment string cannot exist above or below the xsection. I.e. it should be on the section as per 4)
7. The number of points on consecutive section templates (left and right) should be the same. i.e. if 5 points are on the LHS template for ch0 then the same number of points should exist for the LHS template for ch20. The instruments using this file, cannot handle transitions between different number of points effectively and can give incorrect results. The user should limit the file by using the chainage range between areas where the number of points on each side of the cross sections is the same.

The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Select H-Align	Select box		

select valid alignment string.

Report vertical alignment	tick box	no	

write vertical alignment details to file mode. If yes vertical alignment details written to the specified file.

Report cross sections	tick box	no	

if selected, the cross section model box will be enabled.

Cross section model	model box		

This will ask for the model of cross sections. These sections will be written out to the file.

XSec start ch	input box	min alignment ch	

chainage value to start writing cross sections.

XSec end ch	input box	max alignment ch	

chainage value to end writing cross sections.

Ignore points on RHS/LHS sections >12 tick box ticked

if ticked, where there are more than 12 points in either the RHS or LHS template (derived from the xsection) the outlying points are ignored. e.g. if there was 14 points on the RHS template, only 12 will be written to the file with 2 pts being ignored.

File name	input box		

the name to be used for file creation

Write	button		

write appropriate files.

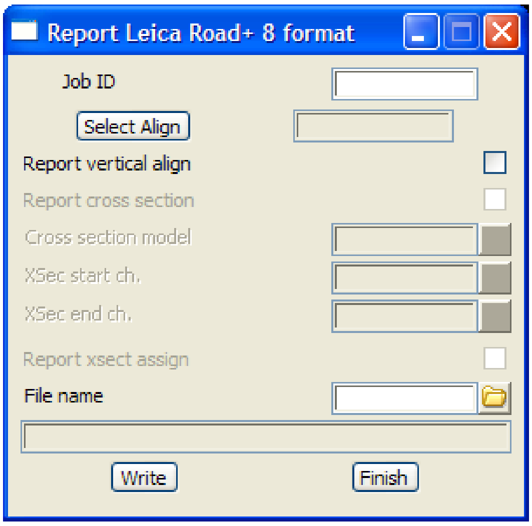
Leica road plus 8 / 16 file

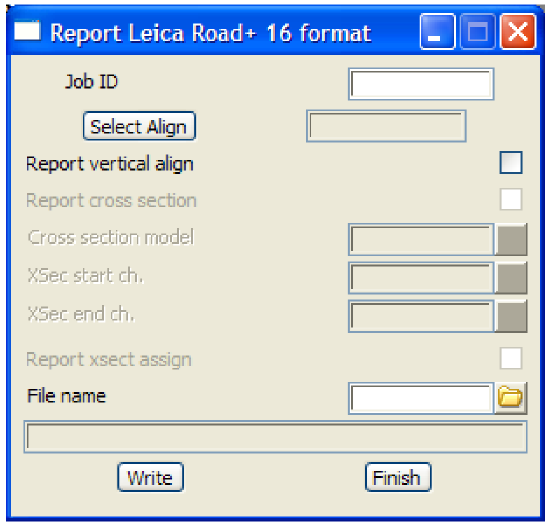
Position of option on menu: Survey =>Upload => Create roads upload file, Instrument choice = Leica roadplus 8 or Leica roadplus 16

The **Leica roadplus 8 and Leica roadplus 16** options allow the creation of specific Leica roading files.

Selecting **Leica roadplus 8** brings up the **Report Leica roadplus 8** panel.

Selecting **Leica roadplus 16** brings up the **Report Leica roadplus 16** panel.





The 8 and 16 formats are different file formats for the leica range of instruments. The user should be aware of the required format for the particular instrument. Please refer to the Leica documentation for specifications

The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Job id	input box		

job id (max 8 characters).

Select alignment	String select box		

select valid alignment string.

Report vertical alignment	tick box	no	

*write vertical alignment details to file mode. If yes vertical alignment details written to appropriate file. i.e. (PRF*****.gsi)*

Report cross sections	tick box	no	

if selected, the cross section model box will be enabled.

Cross section model	model box		

This will ask for the model of cross sections. These sections will be written out to the appropriate file.

*i.e. (CRS*****.gsi)*

XSec start ch	input box	min alignment ch	

chainage value to start writing cross sections.

XSec end ch	input box	max alignment ch	

chainage value to end writing cross sections.

Report cross section assignment	tick box	no	

*If ticked, a cross section assignment file (STA*****.gsi) file will be made.*

File name	input box		

*the name to be used for file creation. **NB depending on what files are specified, the name given will be appended with the correct prefix eg ALN for alignment file. As Leica only takes 8 character file names, the name may be truncated.***

Write	button		

write appropriate files.

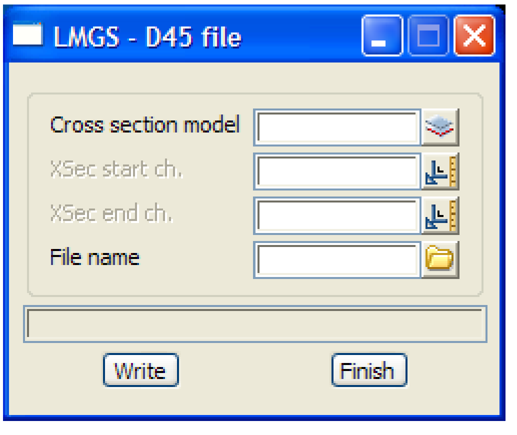
LMGS - D45 file

Position of option on menu: Survey =>Upload => Create roads upload file, Instrument choice = LMGS - D45 file

The **LMGS - D45 file** option allows the creation of a specific file format suitable for upload to Leica'a grader and paver control systems. The format has a number of rules that must be satisfied for correct file creation. These include:

1. There must be an equal number of points on each section
2. Points should not be closer than 5cm in both directions
3. Azimuth changes between cross section points should not be greater than 20gon (18 degrees).

Selecting **LMGS - D45 file** brings up the **LMGS - D45 file** panel



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Cross section model	model box		

the model of cross sections. These sections will be written out to the file.

XSec start ch	input box	min alignment ch	

chainage value to start writing cross sections.

XSec end ch	input box	max alignment ch	

chainage value to end writing cross sections.

File name	input box		

the name to be used for the file creation. The file will automatically be given a .D45 extension.

Write	button		

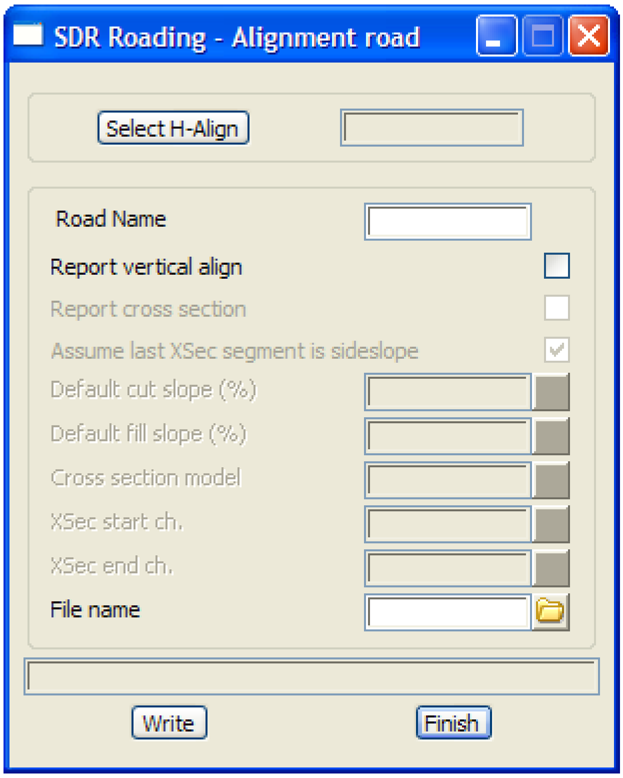
write appropriate file.

Sokkia Roading - Alignment Road file

Position of option on menu: Survey =>Upload => Create roads upload file, Instrument choice = Sokkia Roading - Alignment road

The **Sokkia Roading - Alignment road** option allows the creation of a specific roading file.

Selecting **Sokkia Roading - Alignment road** brings up the **SDR Roading - Alignment Road** panel



The sdr33 roading file has certain limitations and criteria. These include:

1. The alignment string must exist on the xsec's. This allows us to calculate the ***zero pt*** so we can split the template into left and right.
2. The xsec model should be representative of the alignment string used in the macro. i.e. not generated from a different alignment. Therefore the alignment string should have a 0 offset on the xsections.
3. The alignment string cannot exist above or below the xsection. I.e. it should be on the section as per 1).
4. The number of points on adjacent section templates (left and right) should be the same. i.e. if 5 points are on the LHS template for ch0 then the same number of points should exist for the LHS template for ch20. The instruments using this file, may not handle transitions between different number of points effectively and can give incorrect results. The user should limit the file by using the chainage range between areas where the number of points on each side of the cross sections are the same.

The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Select H-Align	Select box		

select valid alignment string.

Road Name	input box	Name of selected alignment string	

input road name. If a valid alignment is selected, the name of that alignment will be the default road name.

Report vertical alignment	tick box	no	

write vertical alignment details to file mode. If yes vertical alignment details written to the specified file.

Assume last XSec segment is sideslope tick box ticked *if box is selected, the last segments left and right of 0.0 offset are considered to be side slopes, and the default cut and fill slope fields are enabled.*

Default cut slope (%) input box

this field is optional. From the section data it is only possible to calculate either the cut or fill sideslope. A default value can be specified to be used for the non calculated value.

When the segment is rising (i.e has a positive slope) it is considered a cut slope. Similarly, if the grade is negative the sideslope is considered a fill. If a fill sideslope is calculated, the cut sideslope definition in the file will be given the value as entered into the ***default cut slope*** *field. If no value is set for the default, the cut slope in this instance will be given a null value. The default value if entered, should be in whole percent e.g. 50.0 for a 1v in 2h slope.*

Default fill slope (%) input box

this field is optional. From the section data it is only possible to calculate either the cut or fill sideslope. A default value can be specified to be used for the non calculated value.

When the segment is rising (i.e has a positive slope) it is considered a cut slope. Similarly, if the grade is negative the sideslope is considered a fill. If a cut sideslope is calculated, the fill sideslope definition in the file will be given the value as entered into the ***default fill slope*** *field. If no value is set for the default, the fill slope in this instance will be given a null value. The default value if entered, should be in whole percent e.g. 50.0 for a 1v in 2h slope.*

Cross section model model box

This will ask for the model of cross sections. These sections will be written out to the file.

XSec start ch input box min alignment ch

chainage value to start writing cross sections

XSec end ch input box max alignment ch

chainage value to end writing cross sections

File name input box

the name to be used for file creation

Write button

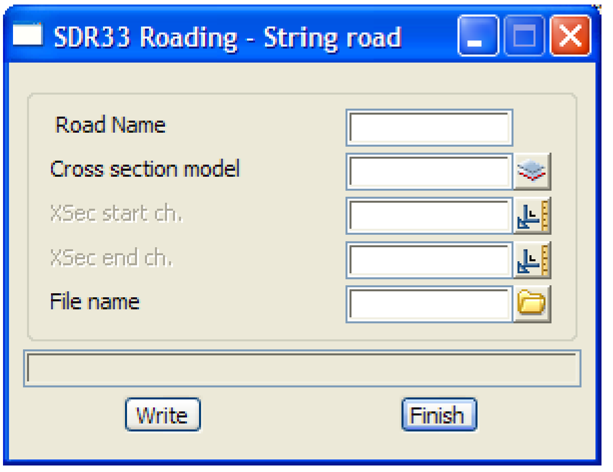
write appropriate file

Sokkia Roading - String Road file

Position of option on menu: **Survey =>Upload => Create roads upload file, Instrument choice = Sokkia Roading - String road**

The **Sokkia Roading - String road** option allows the creation of a specific string roading file.

Selecting **Sokkia Roading - String road** brings up the **SDR Roading - String Road** panel



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Road Name	input box	Name of selected alignment string	

input road name. If a valid alignment is selected, the name of that alignment will be the default road name.

Cross section model	Model box		available models

model of cross sections to write out in SDR33 string format

XSec start ch.	input box	min alignment ch	

chainage of the first cross sections to write out

XSec end ch.	input box	max alignment ch	

chainage of the last cross sections to write out

File name	file box		

the name to be used for file creation

Write	button		

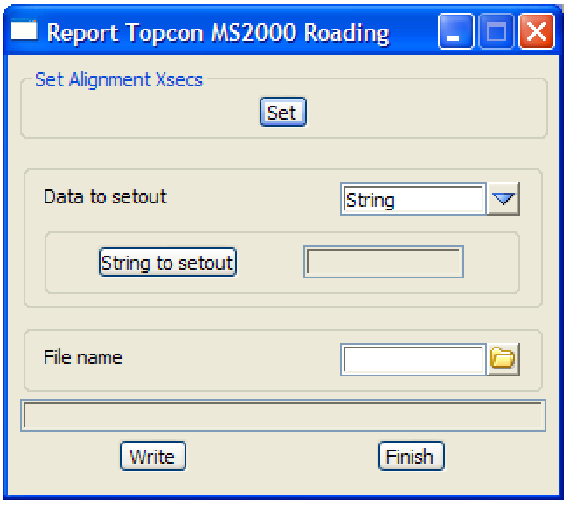
write appropriate file.

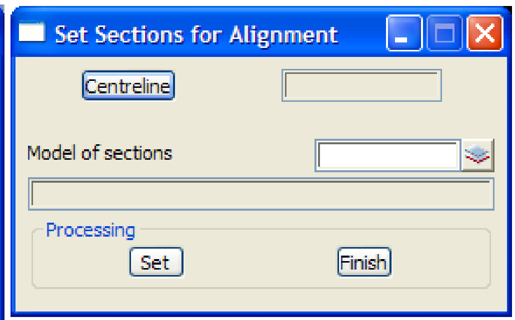
Topcon MS2000 Roading file

Position of option on menu: **Survey =>Upload => Create roads upload file, Instrument choice = Topcon MS2000**

The **Topcon MS2000** option allows the creation of a specific roading file.

Selecting **Topcon MS2000** brings up the **Report topcon MS2000 Roading** panel





The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Set	button		Set section for alignment panel

the set button will bring up the **set sections for alignment panel.** *This panel allows the user to select multiple alignment/xsection combinations for the one file. The alignment is selected (using the* **Centreline** *selection button) followed by the* **Model of sections** *for that alignment. The* **Set** *button is pressed to accept the selection. This can be repeated for a number of alignments. Finish will close the* **set sections for alignment panel** *and return to the original panel*

Data to setout	choice box	string	model view string

the data to be included in the file can be selected by choosing a valid model, view or string. Depending on which data source is selected, the appropriate selection button will appear. The user then selects the button and then the items of interest.

File name input box

the name to be used for file creation

Write button

write appropriate files.

Topcon GT700 Roading File

Position of option on menu: **Survey =>Upload => Create roads upload file, Instrument choice = Topcon GTS-700 Roads**

The **Topcon GTS-700 Roads** option allows the creation of a specific roading file.

Selecting **Topcon GTS-700 Roads** brings up the **Report Topcon GT700 Roading** panel

The fields and buttons used in this panel have the following functions:

Field Description | Type | Defaults | Pop-Up

String to setout select button

after pressing the string to setout button, the user can select a valid string. This string will be used for setting out.

Model of sections model box

the model of cross sections. These sections will be written out to the file.

File for horizontal input box

the name to be used for horizontal file creation. The file will automatically be given a .hg extension.

File for vertical input box

the name to be used for the vertical file creation. The file will automatically be given a .vg extension.

File for sections input box

the name to be used for the section file creation. The file will automatically be given a .xs extension.

Write button

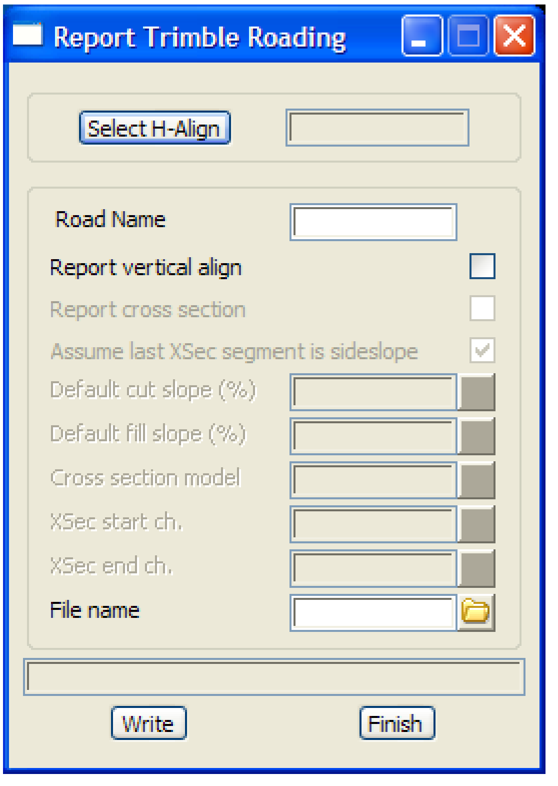
write appropriate files.

Trimble Roading File

Position of option on menu: **Survey =>Upload => Create roads upload file, Instrument choice = Trimble Roading**

The **Trimble Roading** option allows the creation of a specific roading file.

Selecting **Trimble Roading** brings up the **Report Trimble Roading** panel



The Trimble DC file has certain limitations and criteria. These include:

1. The alignment string must exist on the xsec's. This allows us to calculate the ***zero pt*** so we can split the template into left and right.
2. The xsec model should be representative of the alignment string used in the macro. i.e. not generated from a different alignment. Therefore the alignment string should have a 0 offset on the xsections.
3. The alignment string cannot exist above or below the xsection. I.e. it should be on the section as per 1).
4. The number of points on successive section templates (left and right) should be the same. i.e. if 5 points are on the LHS template for ch0 then the same number of points should exist for the LHS template for ch20. The instruments using this file, may not handle transitions between different number of points effectively and can give incorrect results. The user should limit the file by using the chainage range between areas where the number of points on each side of the cross sections are the same.

The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Select H-Align	Select box		

select valid alignment string.

Road Name	input box	Name of selected alignment string	

input road name. If a valid alignment is selected, the name of that alignment will be the default road name.

Report vertical alignment	tick box	no	

write vertical alignment details to file mode. If yes vertical alignment details written to the specified file.

Assume last XSec segment is sideslope tick box ticked *if box is selected, the last segments left and right of 0.0 offset are considered to be side slopes, and the default cut and fill slope fields are enabled.*

Default cut slope (%) input box

this field is optional. From the section data it is only possible to calculate either the cut or fill sideslope. A default value can be specified to be used for the non calculated value.

When the segment is rising (i.e has a positive slope) it is considered a cut slope. Similarly, if the grade is negative the sideslope is considered a fill. If a fill sideslope is calculated, the cut sideslope definition in the file will be given the value as entered into the ***default cut slope*** *field. If no value is set for the default, the cut slope in this instance will be given a null value. The default value if entered, should be in whole percent e.g. 50.0 for a 1v in 2h slope.*

Default fill slope (%) input box

this field is optional. From the section data it is only possible to calculate either the cut or fill sideslope. A default value can be specified to be used for the non calculated value.

When the segment is rising (i.e has a positive slope) it is considered a cut slope. Similarly, if the grade is negative the sideslope is considered a fill. If a cut sideslope is calculated, the fill sideslope definition in the file will be given the value as entered into the ***default fill slope*** *field. If no value is set for the default, the fill slope in this instance will be given a null value. The default value if entered, should be in whole percent e.g. 50.0 for a 1v in 2h slope.*

Cross section model model box

This will ask for the model of cross sections. These sections will be written out to the file.

XSec start ch input box min alignment ch

chainage value to start writing cross sections.

XSec end ch input box max alignment ch

chainage value to end writing cross sections.

File name input box

the name to be used for file creation

Write button

write appropriate files.

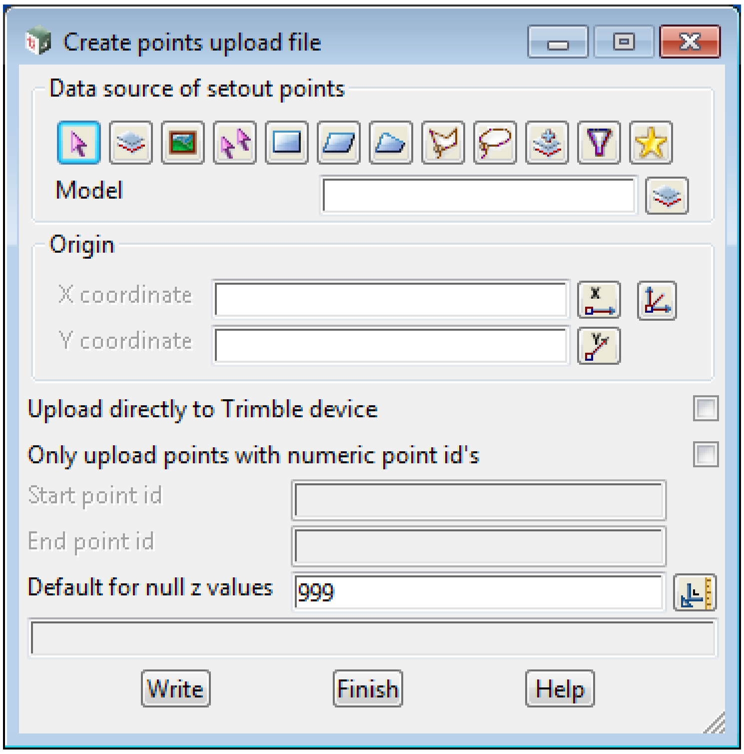
17.19.5 Create Points Upload File (New)

Position of option on menu: **Survey =>Upload => Create points upload file (new)**

The **Create points upload file (new)** option creates or uploads directly to a Trimble instrument a file of point ids, x, y and z, for string data (super strings and 4d strings only).

Only Trimble file formats is supported as this stage - the option **Create points upload file** (17.19.3 Create Points Upload File) is used to create files for most other instruments.

Selecting **Create points upload file (new)** brings up the **Create Points Upload File** panel.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Upload directly to Trimble device tick box

if ticked, the points are uploaded to the attached Trimble instrument
if not ticked, a dc file will be created.

Data source type

data selection type - for a full description go to 4.19.3 Data Source .

Data source

source of data to be processed.

Default for null value input box 999

value to write for null z-values.

Only upload points with numeric point names tick box

if ticked, only points with numeric (integer) point names are used.
if not ticked, alphanumeric point names are used.

Start point id input box

point id to start creating upload file from.

End point id input box

last point id to write to upload file.

Note: Start and End point ids may be left blank, and all point ids (whether numeric or alpha-numeric) will be written to the upload file.

Origin x/Origin y input box

If non-zero, subtract the value from the x/y value before writing out.

Upload File button

create the upload file

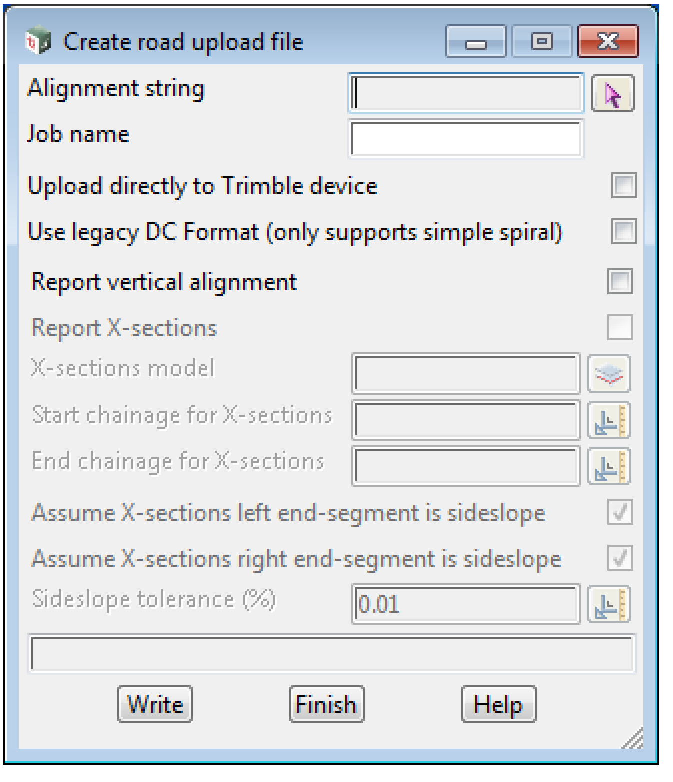
17.19.6 Create Road Upload File (New)

Position of option on menu: **Survey =>Upload => Create road upload file (new)**

The **Create road upload file (new)** option allows the creation of road files or uploads directly to a Trimble instrument and Leica using *Leica XML* format.

Only *Trimble* and *Leica LandXML* file formats are supported as this stage- the option **Create road upload file** (17.19.4 Create Road Upload File) is used to create files for most other instruments.

Selecting **Create road upload file (new)** brings up the **Create Road Upload File** panel



The fields and buttons used in this panel have the following functions:

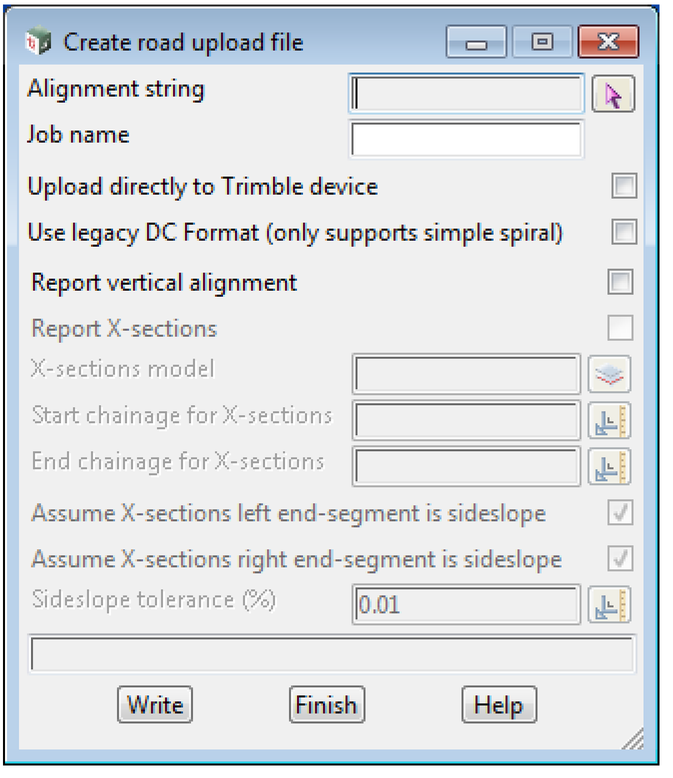
Field Description	Type	Defaults	Pop-Up
File type	choice box		Trimble Link Leica X-sections LandXML

For the choice Trimble Link, go to Trimble Link
Leica X-Sections LandXML Leica X-Sections LandXML

Trimble Link

Position of option on menu: **Survey =>Upload => Create roads upload file (new), File type = Trimble Link**

The **Trimble Link** options creates a file in Trimble dc format or uploads the date directly to the instrument using *Trimble Link*.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Alignment string String select box

select valid alignment string.

Job Name input box

the name for the Trimble job.

Upload directly to Trimble device tick box not ticked

if ticked, the points are uploaded to the attached Trimble instrument
if not ticked, a dc file will be created.

Use legacy DC format (only supports simple spiral) tick box not ticked

if ticked, the old DC format will be used.

Report vertical alignment tick box

If ticked, vertical alignment details written to file.

Report X-sections tick box

if ticked, the cross section model box will be enabled.

X- sections model model box

model of cross sections to be written out.

Start chainage for X-sections

chainage value to start writing cross sections. If blank, start with first x-section.

End chainage for X-sections

chainage value to stop writing cross sections. If blank, end with last x-section.

Assume X-sections left end-segment is sideslope tick box

If ticked, left end segments are tagged as sideslopes.

Assume X-sections right end-segment is sideslope tick box

If ticked, right end segments are tagged as sideslopes.

Sideslope tolerance (%) input 0.01

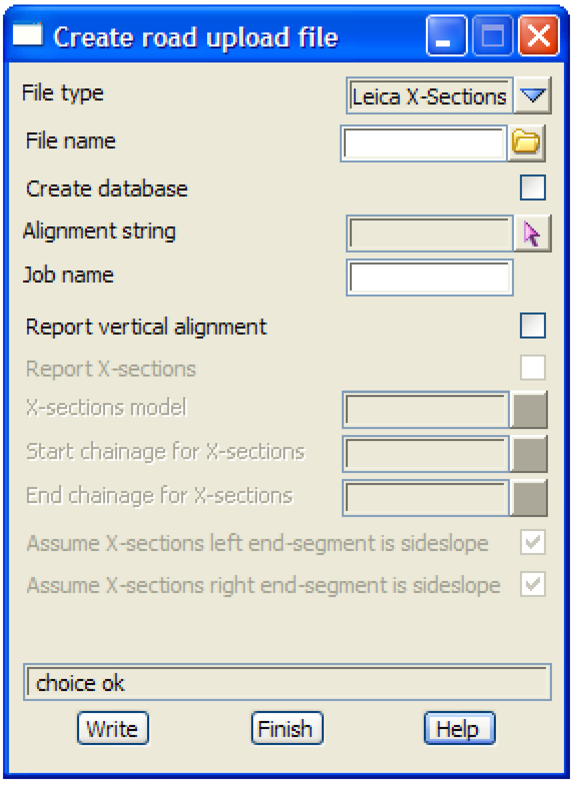
Write button

write out or upload the date.

Leica X-Sections LandXML

Position of option on menu: Survey =>Upload => Create roads upload file (new), File type = Leica X-Sections LandXML

The **Leica X-Sections LandXML** options creates a file in LandXML format suitable for *Leica* instruments



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
File name	input box		*.xml

the name of the Leica LandXML file.

Create database tick box

if ticked, a DBX file is created

Alignment string string select

select valid alignment string.

Job Name input box name of selected alignment string

input job name. If a valid alignment is selected, the name of that alignment will be the default job name.

Report vertical alignment tick box

if ticked, vertical alignment details written to file.

Report cross sections tick box

if ticked, the cross section model box will be enabled.

X- sections model model box available models

model of cross sections to be written out.

Start chainage for X-sections

chainage value to start writing cross sections. If blank, start with first x-section.

End chainage for X-sections

chainage value to stop writing cross sections. If blank, end with last x-section.

Assume X-sections left end-segment is sideslope tick box

If ticked, left end segments are tagged as sideslopes.

Assume X-sections right end-segment is sideslope tick box

If ticked, right end segments are tagged as sideslopes.

Write button

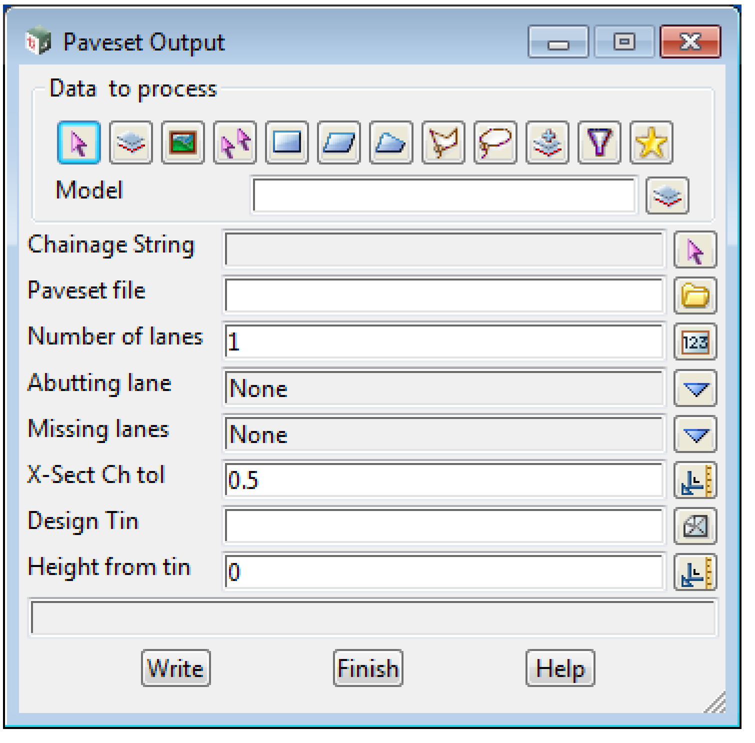
write out file.

17.19.7 Create Paveset Upload File

Position of option on menu: **Survey =>Create Paveset upload file**

The **Paveset Output** panel produces data in a format used by Paveset machine guidance software used in laying asphalt and other materials.

Selecting **Create Paveset upload file** brings up the **Paveset Output** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data to process	Model		

data selection type - for a full description go to 4.19.3 Data Source .

Data source	input		

source of data to be processed.

Chainage String	string select		

The string for chainages

Paveset file	file select		

File to write the output data to

Number of lanes	number box		

The number of lanes for this paving run

Abutting lane	choice box	None	None, Left, Right

If the paving run is abutting an existing lane choose that here

Missing lanes	choice box	None	None, Left, Right

If the paving runs have missing points in some areas choose the side they are missing from here.

None *- the same number of points exist at each cross section, an error will occur in processing if a difference is detected.*

Left/Right *- the output to paveset is padded on the nominated side to match the nominal number of* *lanes, when processing is finished you will display a list of chainages where this occurred*

X- Sect Ch tol input box 0.5

The chainage variation allowed for each cross section

Design Tin input

The design TIN to drop the selected points against for level differences

Height from tin input box

The height from the design TIN
+ve = above
-ve = below

Write button

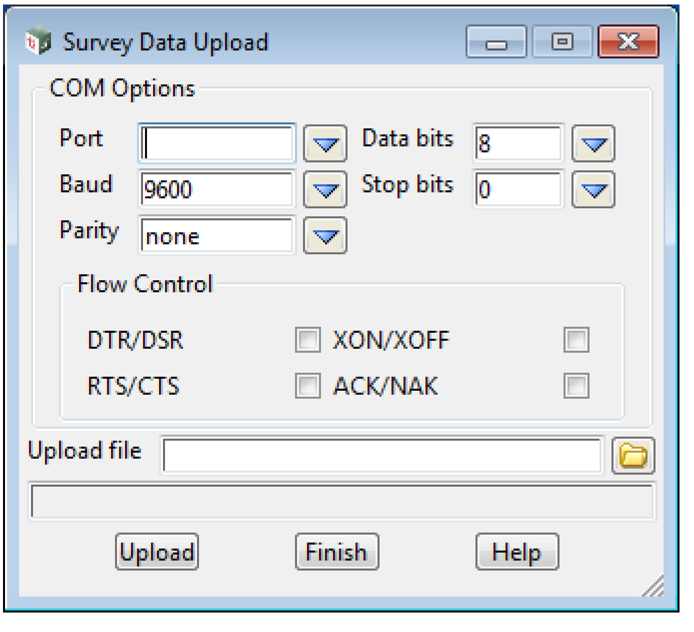
Process the points and write the Paveset file

17.19.8 Survey Data Upload

Position of option on menu: Survey =>Upload

The **upload** option take data from the computer and sends it to the data collector connected to the computer's serial port.

Selecting **Upload** brings up the **Survey Data Upload** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Port	input	from configuration file	

name of the computer serial port that the data collector is connected to. The default port is specified in the data collector configuration file.

Baud rate input from configuration file

speed of the serial port

Data bits input from configuration file

number of data bits

Stop bits input from configuration file *number of stop bits*

Parity input from configuration file

parity

DTR/DSR from configuration file

if ticked, use DTR/DSR flow control

RTS/CTS from configuration file

if ticked, use RTS/CTS flow control

XON/XOFF from configuration file

if ticked, use Xon/Xoff

ACK/NAK from configuration file

if ticked, use ACK/NAK

Field file input *.fld files

name of the 12d field file that the raw file is to be converted to. The raw file is given the same name but with the extension specified in the configuration file, e.g., ".gre".

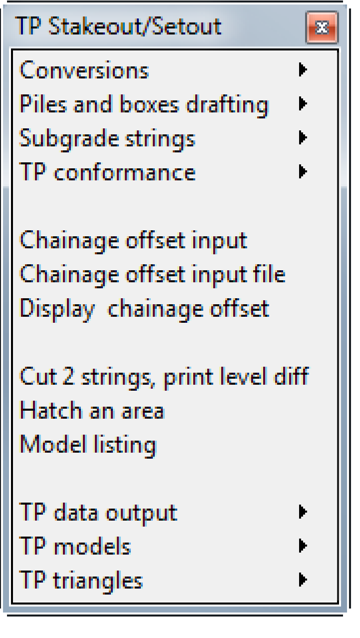
Upload button

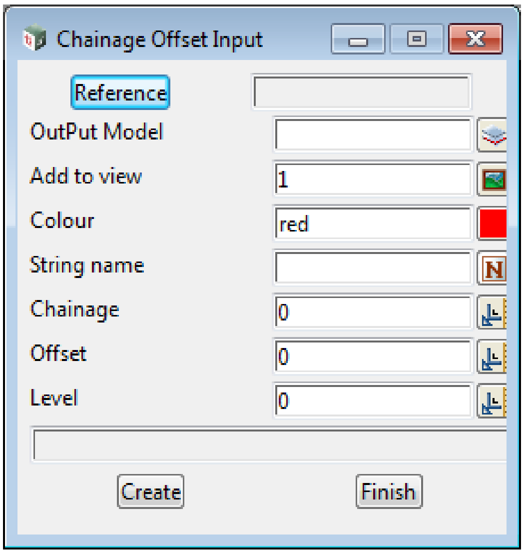
On clicking the upload button, ***12d Model*** *will send the specified file through the serial port and up to the data collector.*

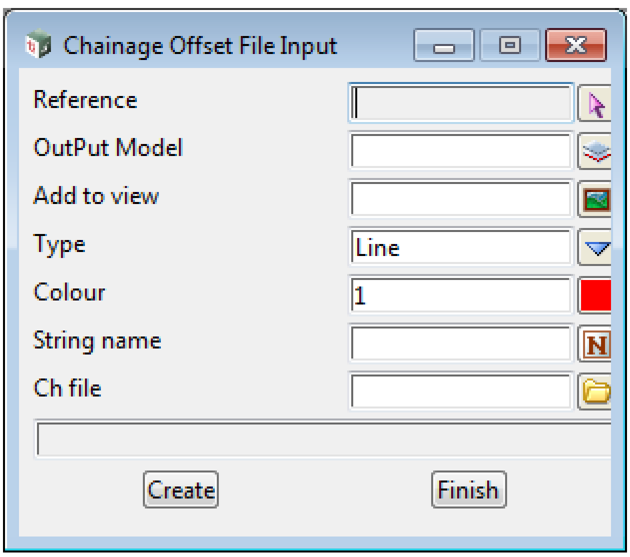
17.20 TP Stakeout/Setout

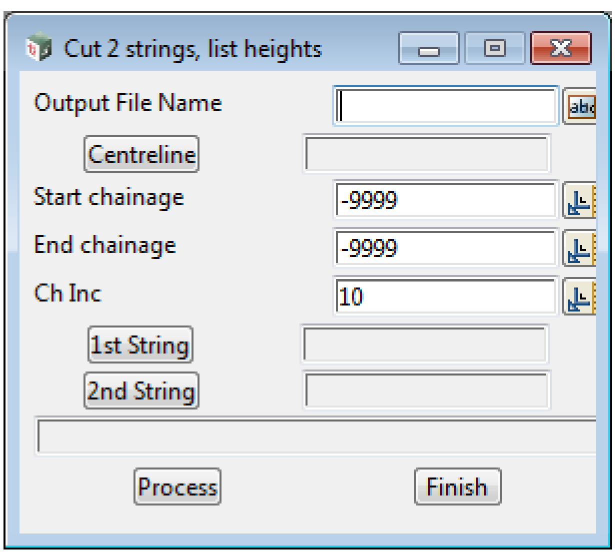
The TP Stakeout /Setout menu contains a variety of options for TP Stakeout and TP Setout users.

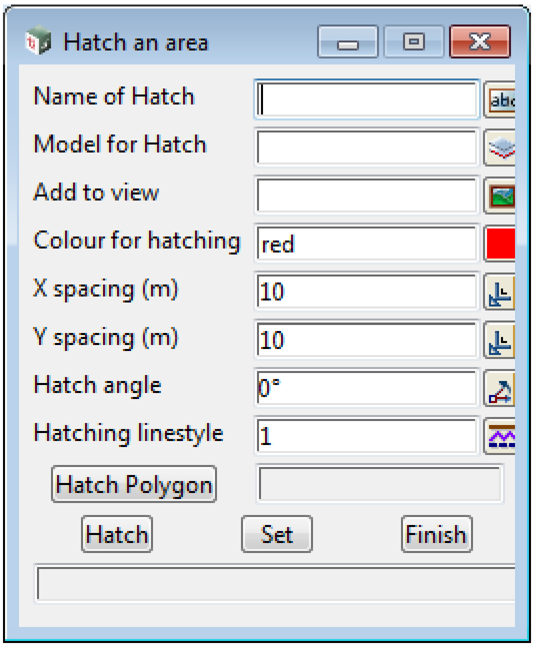
These options are not supplied or supported by 12d Solutions Pty Ltd.

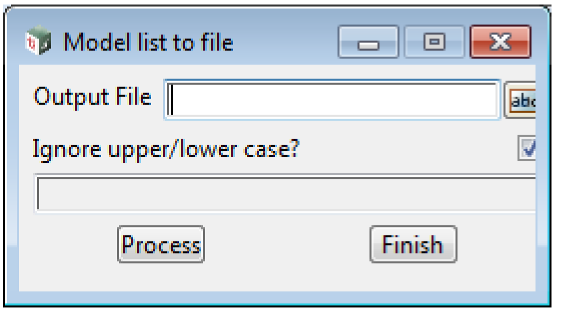



Chainage Offset Input
Reference
OutPut Model
Add to view
1
Colour
red
String name
Chainage
0
Offset
0
Level
0
Create
Finish


Chainage Offset File Input
Reference
OutPut Model
Add to view
Type
Line
Colour
1
String name
Ch file
Create
Finish


Cut 2 strings, list heights
Output File Name
Centreline
Start chainage
-9999
End chainage
-9999
Ch Inc
10
1st String
2nd String
Process
Finish

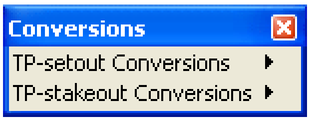




17.20.1 Conversions

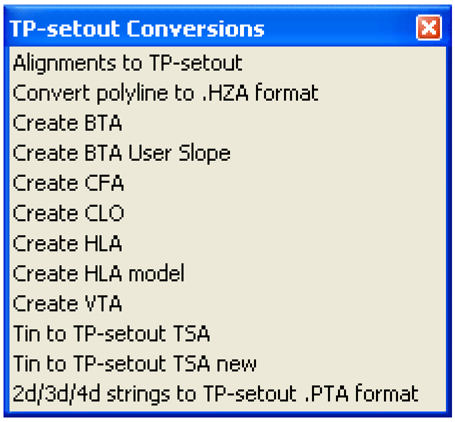
The TP Stakeout /Setout menu contains a variety of options provided for TP Stakeout and TP Setout users.

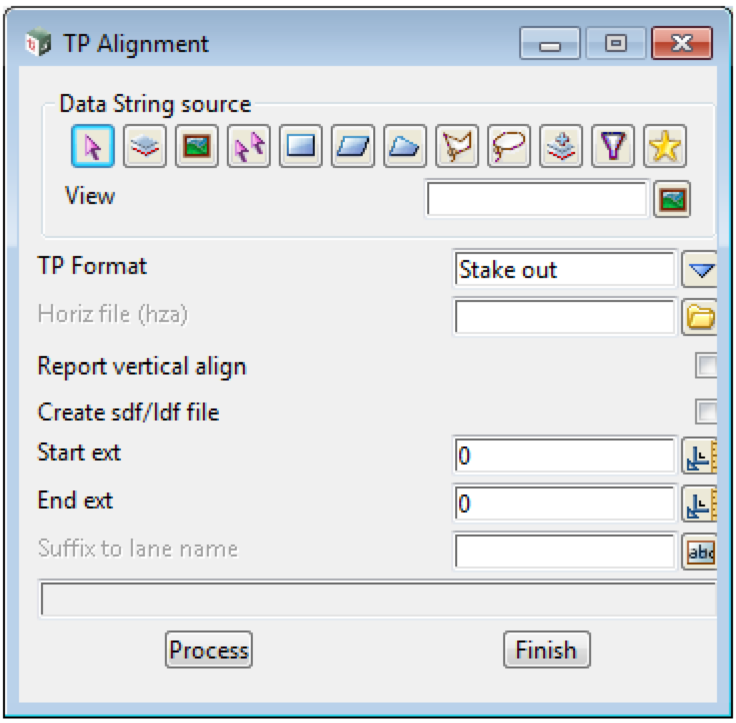
These options are not supplied or supported by 12d Solutions Pty Ltd.

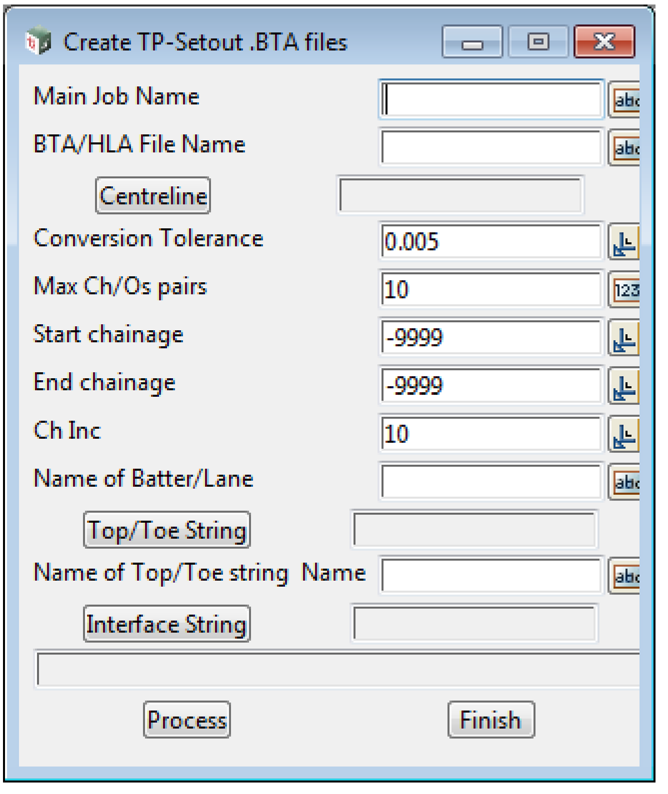


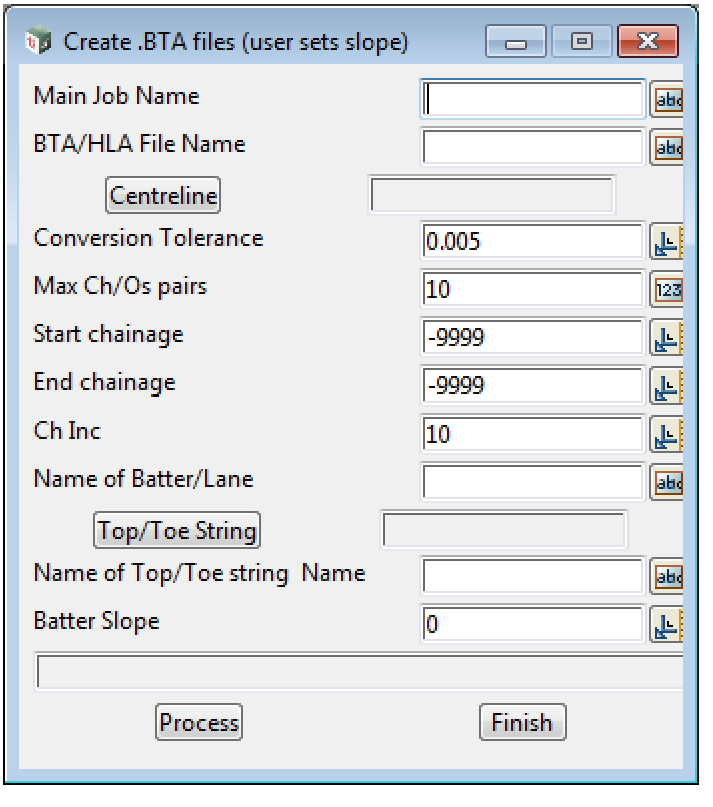
TP Setout Conversions

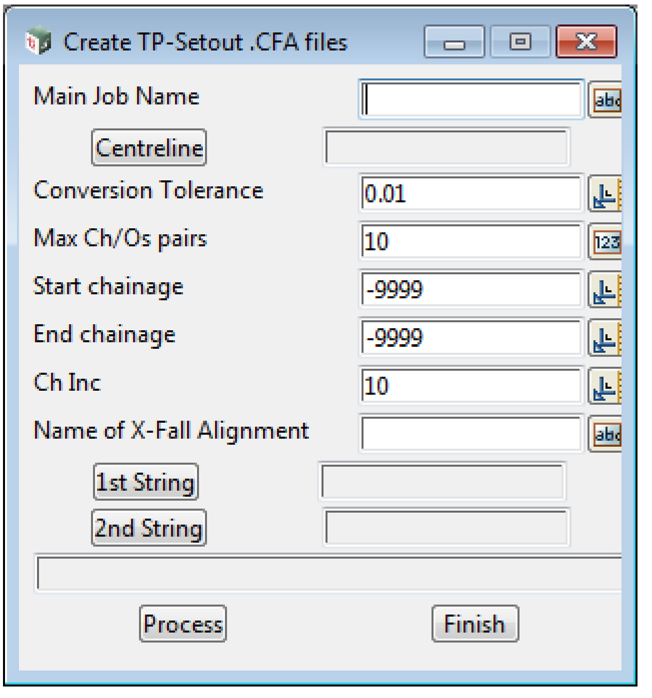
These options are not supplied or supported by 12d Solutions Pty Ltd.

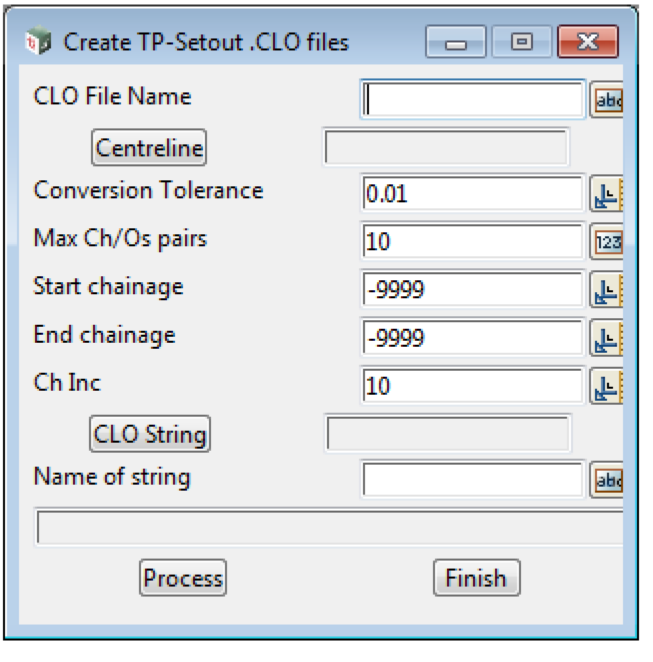

TP-setout Conversions
Alignments to TP-setout
Convert polyline to .HZA format
Create BTA
Create BTA User Slope
Create CFA
Create CLO
Create HLA
Create HLA model
Create VTA
Tin to TP-setout TSA
Tin to TP-setout TSA new
2d/3d/4d strings to TP-setout .PTA format

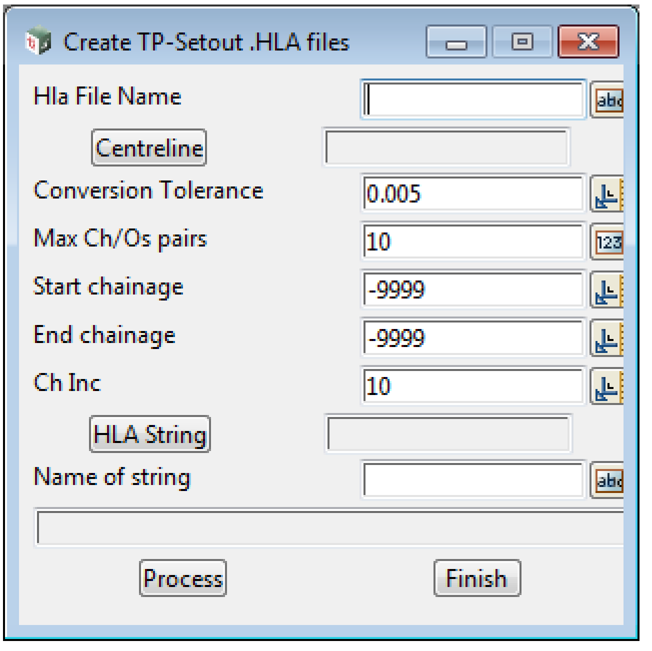

TP Alignment
Data String source
View
TP Format
Stake out
Horiz file (hza)
Report vertical align
Create sdf/ldf file
Start ext
0
End ext
0
Suffix to lane name
Process
Finish

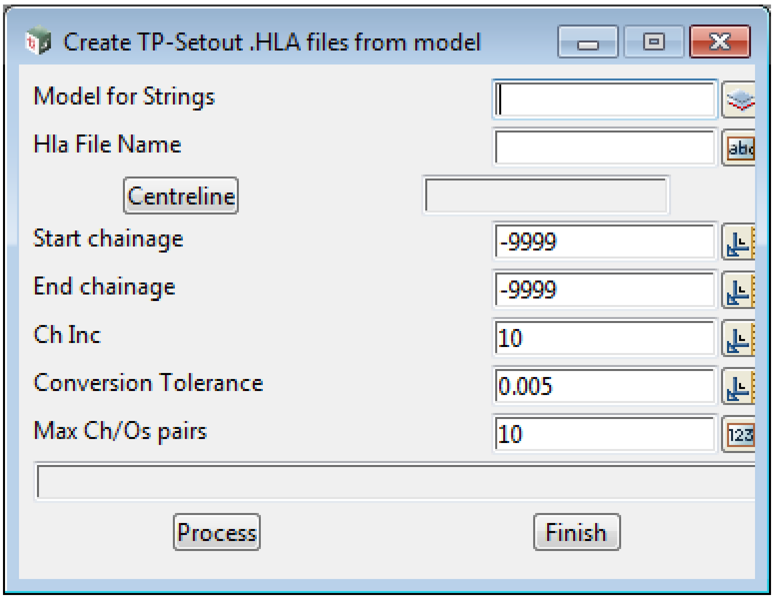

Create TP-Setout .BTA files
Main Job Name
BTA/HLA File Name
Centreline
Conversion Tolerance
0.005
Max Ch/Os pairs
10
Start chainage
-9999
End chainage
-9999
Ch Inc
10
Name of Batter/Lane
Top/Toe String
Name of Top/Toe string Name
Interface String
Process
Finish

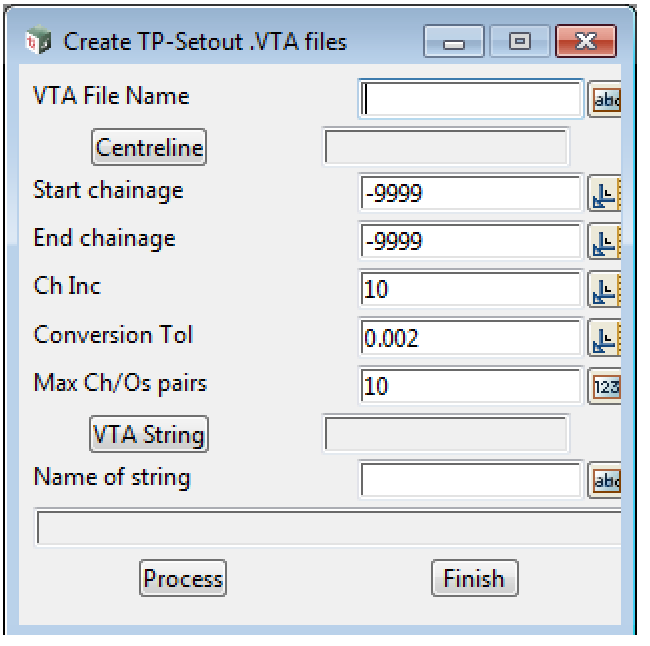

Create .BTA files (user sets slope)
Main Job Name
BTA/HLA File Name
Centreline
Conversion Tolerance
0.005
Max Ch/Os pairs
10
Start chainage
-9999
End chainage
-9999
Ch Inc
10
Name of Batter/Lane
Top/Toe String
Name of Top/Toe string Name
Batter Slope
0
Process
Finish

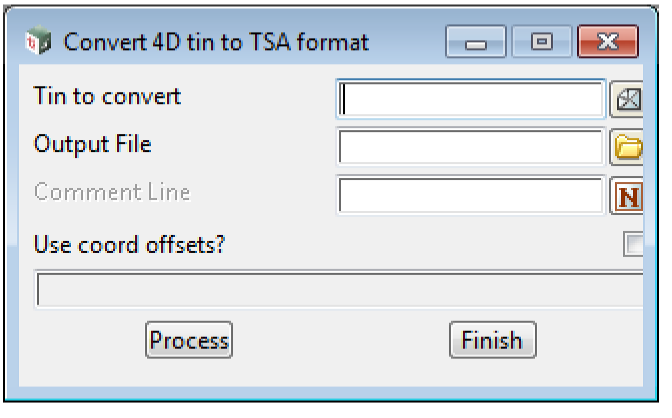

Create TP-Setout .CFA files
Main Job Name
Centreline
Conversion Tolerance
0.01
Max Ch/Os pairs
10
Start chainage
-9999
End chainage
-9999
Ch Inc
10
Name of X-Fall Alignment
1st String
2nd String
Process
Finish


Create TP-Setout .CLO files
CLO File Name
Centreline
Conversion Tolerance
0.01
Max Ch/Os pairs
10
Start chainage
-9999
End chainage
-9999
Ch Inc
10
CLO String
Name of string
Process
Finish


Create TP-Setout .HLA files
Hla File Name
Centreline
Conversion Tolerance
0.005
Max Ch/Os pairs
10
Start chainage
-9999
End chainage
-9999
Ch Inc
10
HLA String
Name of string
Process
Finish


Create TP-Setout .HLA files from model
Model for Strings
Hla File Name
Centreline
Start chainage
-9999
End chainage
-9999
Ch Inc
10
Conversion Tolerance
0.005
Max Ch/Os pairs
10
Process
Finish

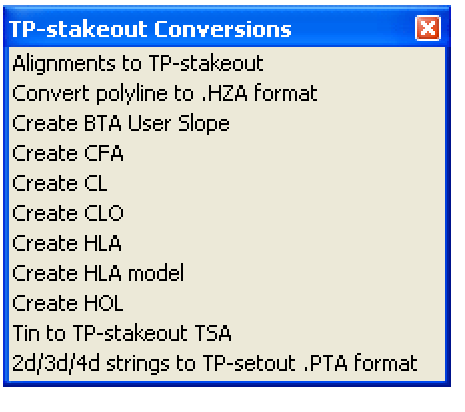


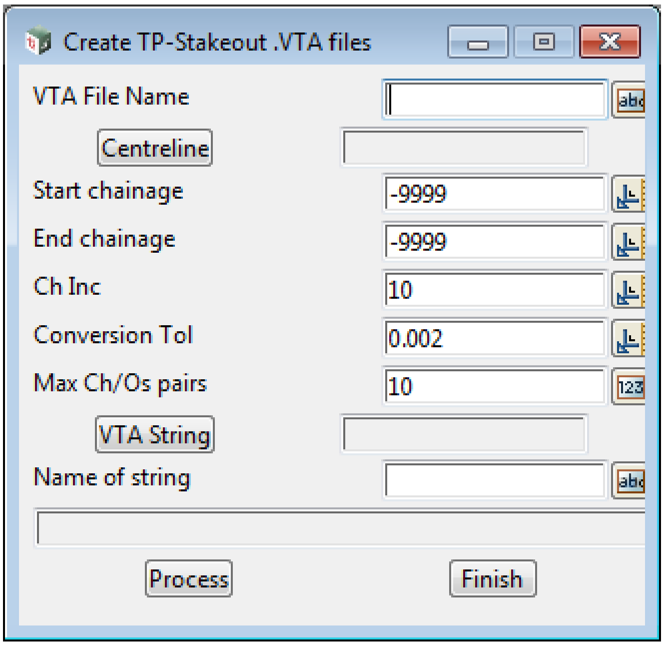


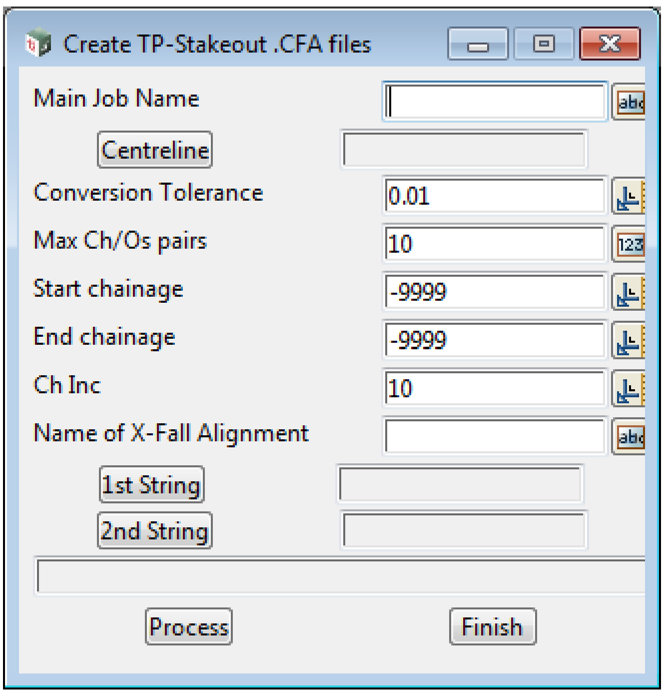
TP Stakeout Conversions

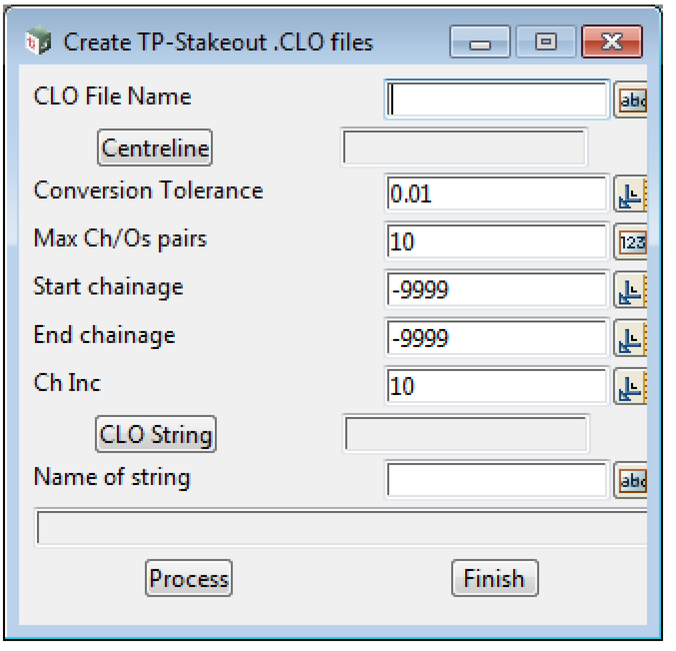
The TP Stakeout /Setout menu contains a variety of options for TP Stakeout and TP Setout users.

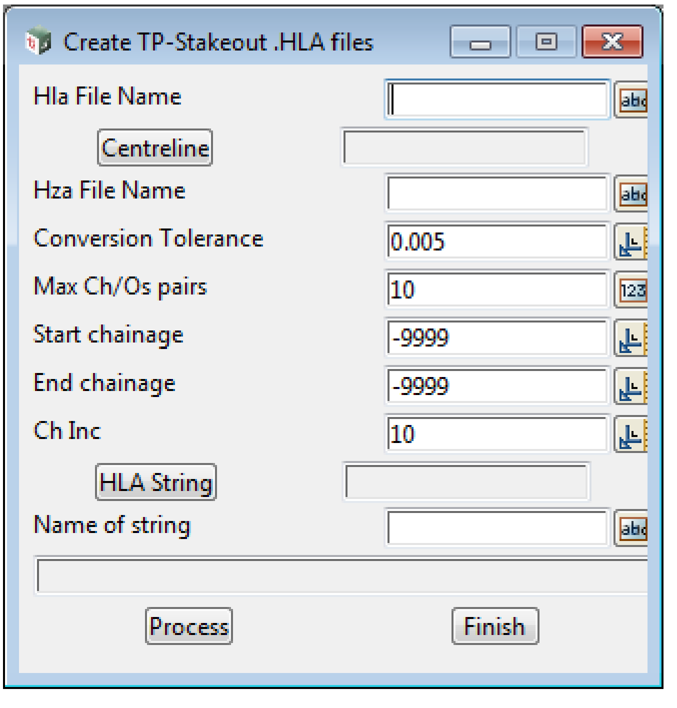
These options are not supplied or supported by 12d Solutions Pty Ltd.

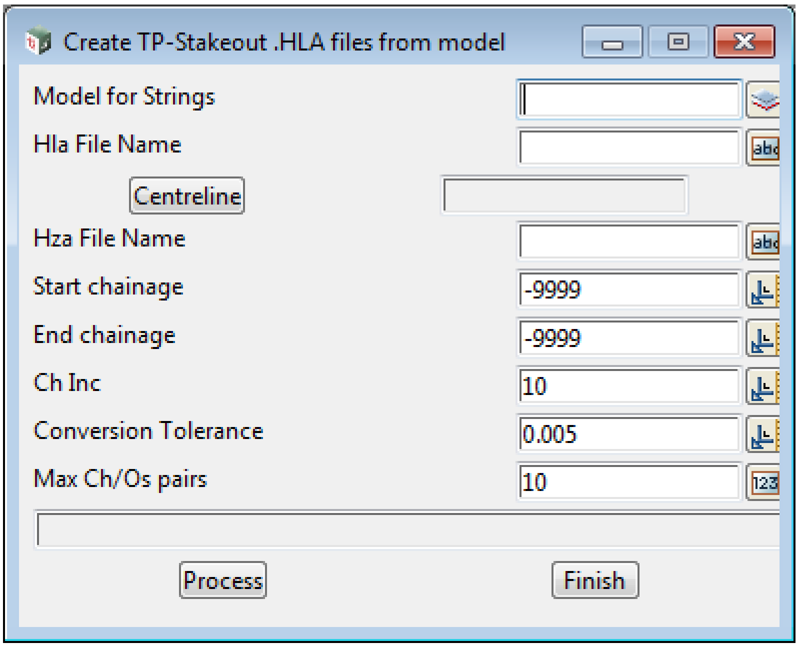


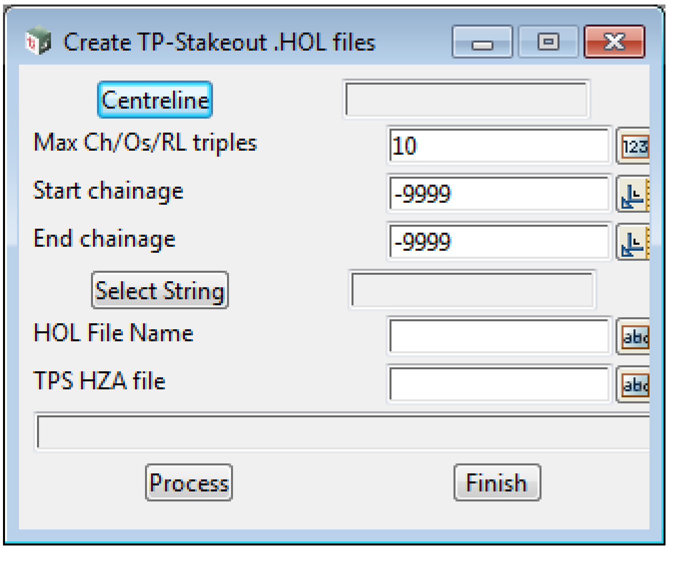

Create TP-Stakeout .VTA files
VTA File Name
Centreline
Start chainage
-9999
End chainage
-9999
Ch Inc
10
Conversion Tol
0.002
Max Ch/Os pairs
10
VTA String
Name of string
Process
Finish


Create TP-Stakeout .CFA files
Main Job Name
Centreline
Conversion Tolerance
0.01
Max Ch/Os pairs
10
Start chainage
-9999
End chainage
-9999
Ch Inc
10
Name of X-Fall Alignment
1st String
2nd String
Process
Finish


Create TP-Stakeout .CLO files
CLO File Name
Centreline
Conversion Tolerance
0.01
Max Ch/Os pairs
10
Start chainage
-9999
End chainage
-9999
Ch Inc
10
CLO String
Name of string
Process
Finish


Create TP-Stakeout .HLA files
Hla File Name
Centreline
Hza File Name
Conversion Tolerance
0.005
Max Ch/Os pairs
10
Start chainage
-9999
End chainage
-9999
Ch Inc
10
HLA String
Name of string
Process
Finish

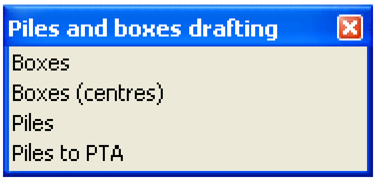

Create TP-Stakeout .HLA files from model
Model for Strings
Hla File Name
Centreline
Hza File Name
Start chainage
-9999
End chainage
-9999
Ch Inc
10
Conversion Tolerance
0.005
Max Ch/Os pairs
10
Process
Finish

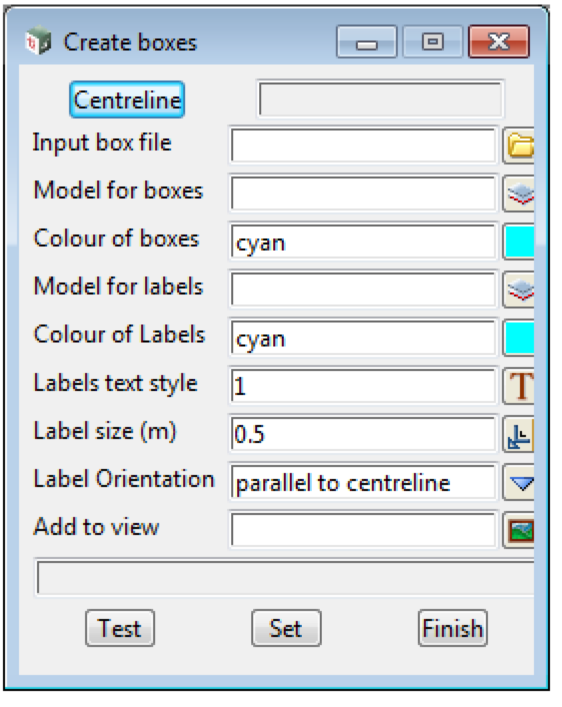

Create TP-Stakeout .HOL files
Centreline
Max Ch/Os/RL triples
10
Start chainage
-9999
End chainage
-9999
Select String
HOL File Name
TPS HZA file
Process
Finish

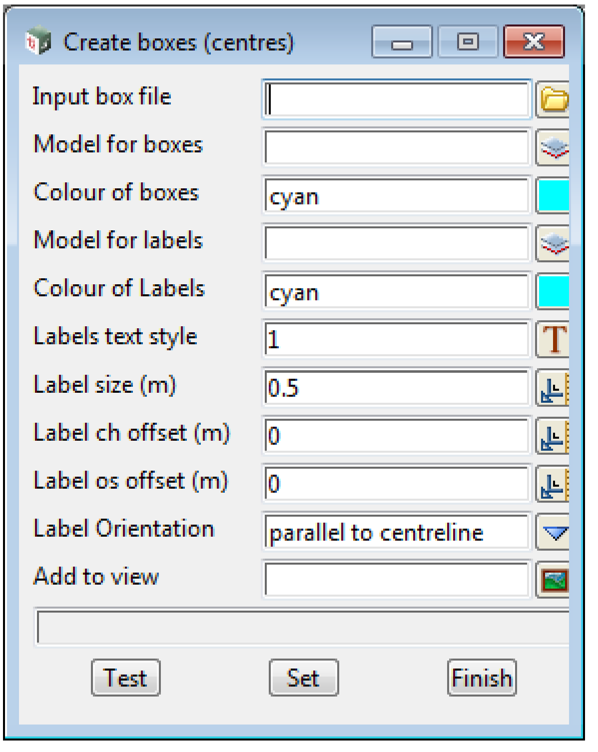
17.20.2 Piles and Boxes Drafting

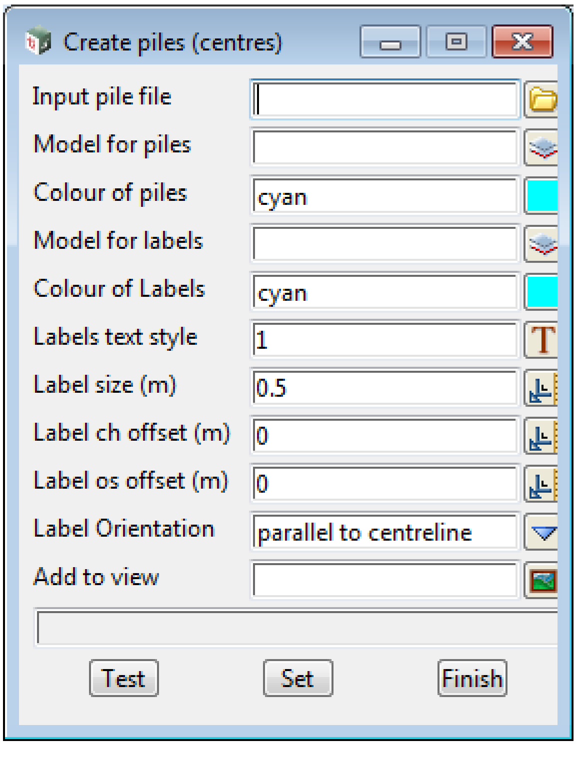
The TP Stakeout /Setout menu contains a variety of options provided for TP Stakeout and TP Setout users.

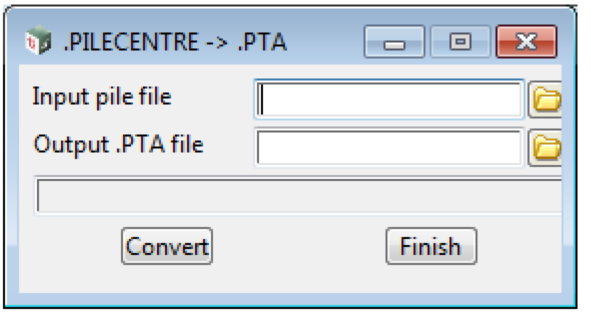
These options are not supplied or supported by 12d Solutions Pty Ltd.





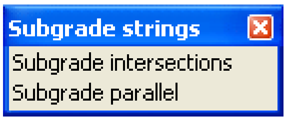

Create boxes (centres)
Input box file
Model for boxes
Colour of boxes
cyan
Model for labels
Colour of Labels
cyan
Labels text style
1
Label size (m)
0.5
Label ch offset (m)
0
Label os offset (m)
0
Label Orientation
parallel to centreline
Add to view
Test
Set
Finish

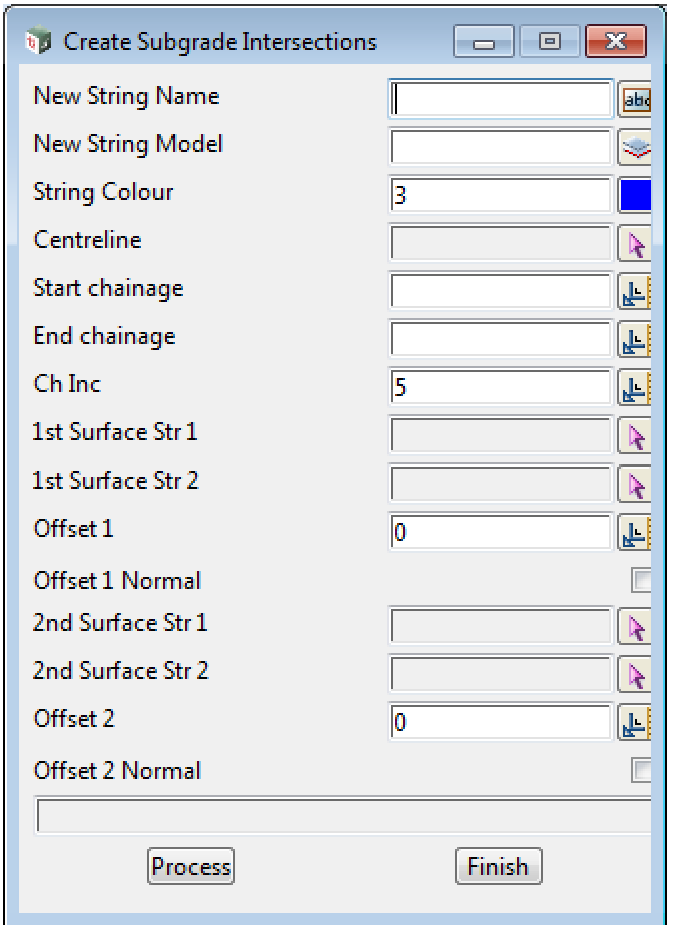

Create piles (centres)
Input pile file
Model for piles
Colour of piles
cyan
Model for labels
Colour of Labels
cyan
Labels text style
1
Label size (m)
0.5
Label ch offset (m)
0
Label os offset (m)
0
Label Orientation
parallel to centreline
Add to view
Test
Set
Finish

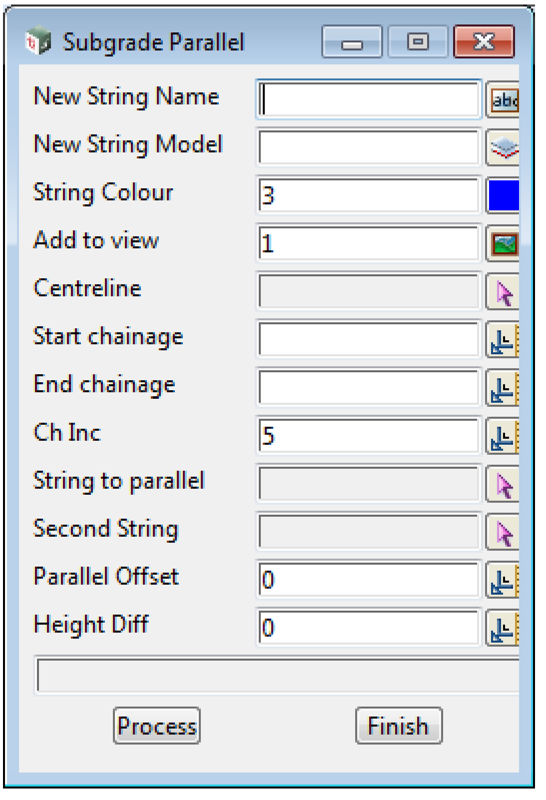

.PILECENTRE -> .PTA
Input pile file
Output .PTA file
Convert
Finish

17.20.3 Subgrade Strings

These options are not supplied or supported by 12d Solutions Pty Ltd.

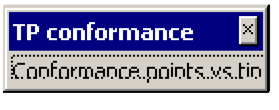


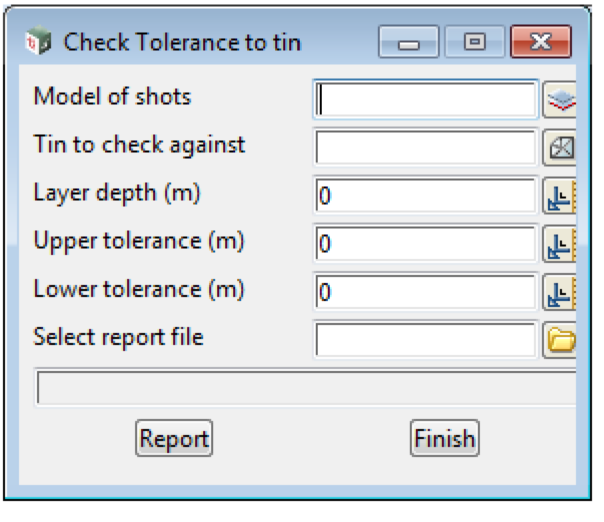




17.20.4 TP Conformance

These options are not supplied or supported by 12d Solutions Pty Ltd.



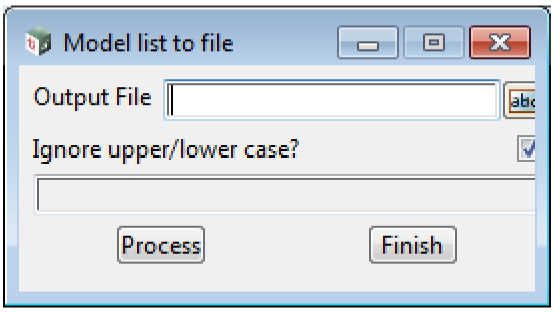


17.20.5 TP Conformance

These options are not supplied or supported by 12d Solutions Pty Ltd.

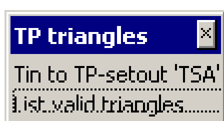
17.20.6 TP Models

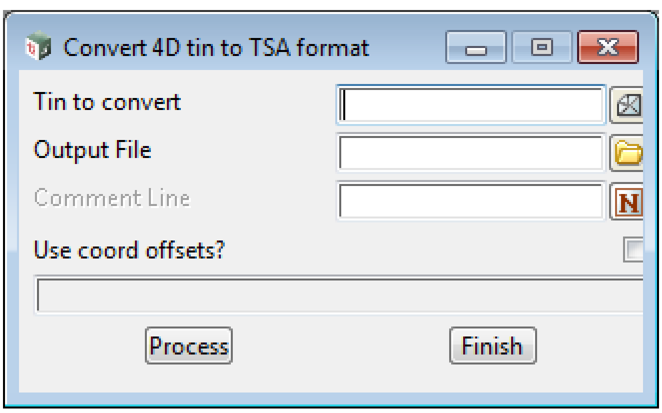
These options are not supplied or supported by 12d Solutions Pty Ltd.

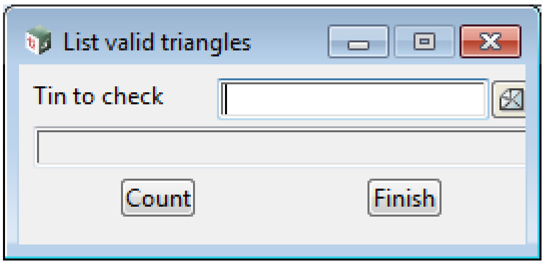


17.20.7 TP Triangles

These options are not supplied or supported by 12d Solutions Pty Ltd.






List valid triangles
Tin to check
Count
Finish

18 12d Field

The **12d Field** allows you to take the full functionality of ***12d Model*** out into the field and control and instrument or GPS for Setout and Pickup.

\

See

18.1 12d Field Overview

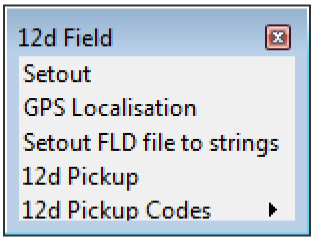
The **12d Field** module is for communicating directory with survey grade GPS units and Total Station theodolites through radio, blue tooth, Wi-Fi and cable. Survey data is displayed live on screen in conjunction with full road project design and existing surface data.

18.2 12d Field

Position of menu: **Survey =>12d Field**

The **12d Field** walk-right menu is:

.



See

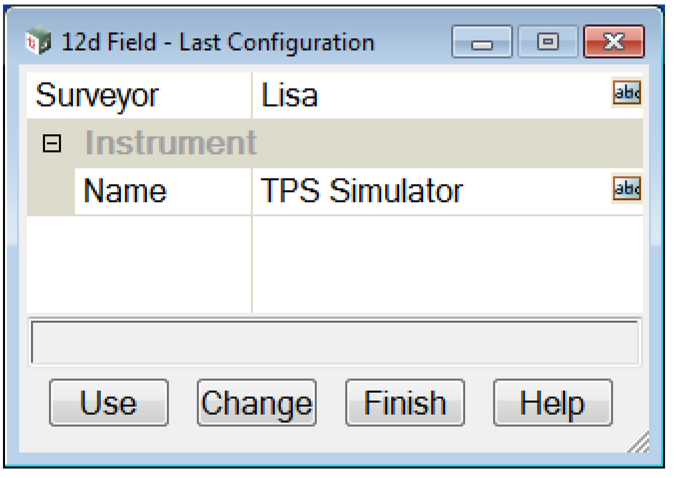
Setout, go to 18.2.1 Setout
GPS Localisation 18.2.2 GPS Localisation
Setout FLD file to strings 18.2.3 12dField Setout FLD File To Strings
12d Pickup 18.2.4 12d Pickup
12d Pickup Codes 18.2.5 12d Pickup Codes

18.2.1 Setout

Position of menu: **Survey =>12d Field=>Setout**

This section of documentation is a work in progress and will be updated in subsequent releases.

This manual will only refer to the **TPS** and **GPS simulators**, for connection details to actual instruments refer to the supplied 12dField documentation.



Continue to next section 18.2.2 GPS Localisation or return to 18.2 12d Field .

18.2.1.1 TPS

This section of documentation is a work in progress and will be updated in subsequent releases.

For the **TPS Simulator** this is the only panel to enter 12dField. Press **Set** and you will enter 12dField proper.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Instrument	choice box		

Select the instrument you wish to use with 12dField, dependent on the instrument selected the panel flow will vary.

Surveyor

Enter the name for the surveyor using 12dField.

After selecting **Set** *the* **12d Field - Setout TPS Simulator** *is displayed on the screen. For more information please go to* 18.2.1.4 12d Field - Setout (TPS Simulator)

Set

Enter 12dField

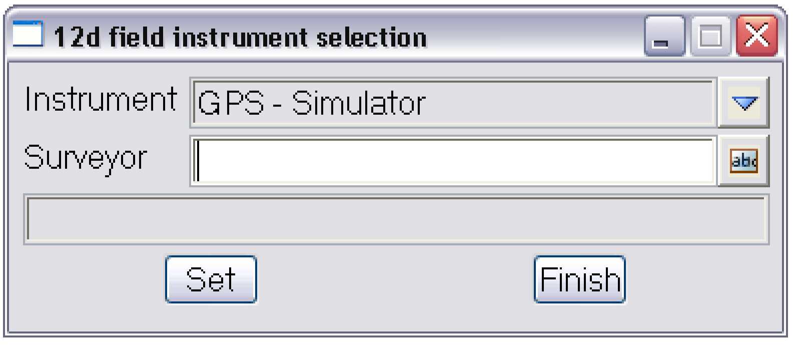
Continue to next section 18.2.1.2 GPS or return to 18.2 12d Field .

18.2.1.2 GPS

This section of documentation is a work in progress and will be updated in subsequent releases.

For the **GPS-Simulator** 2 more panels are required

Select **GPS-Simulator** from the choice box.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Instrument	choice box		

Select the instrument you wish to use with 12dField, dependent on the instrument selected the panel flow will vary.

Surveyor

Enter the name for the surveyor using 12dField.

Set

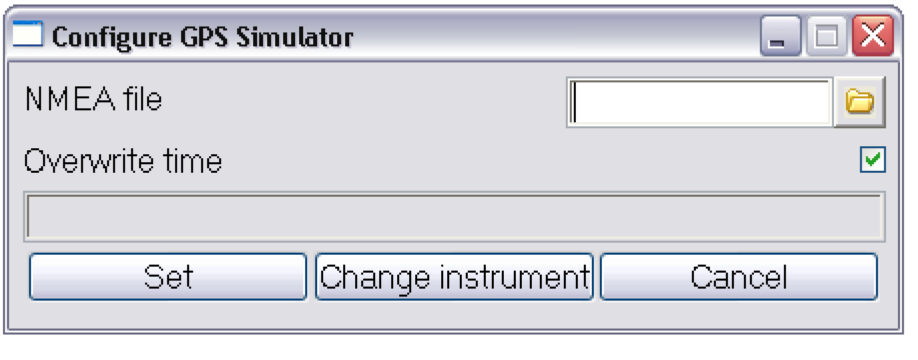
Brings up the **Configure GPS Simulator** *panel.*

For more information on the **Configure GPS Simulator** panel please go to 18.2.1.3 Configure GPS Simulator

Continue to next section 18.2.1.3 Configure GPS Simulator or return to 18.2 12d Field .

18.2.1.3 Configure GPS Simulator

This section of documentation is a work in progress and will be updated in subsequent releases.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

NMEA file folder *.nmea

The GPS simulator needs a NMEA file to run, this can be a file recorded from an actual instrument or constructed from within 12dField itself by 'driving' along an existing string. A default file "Default_LLQ.NMEA" is created automatically for 1st up usage.

Overwrite time tick box

if ticked the time in the NMEA string is ignored and the current computer time used instead.

Set button

proceed to the next panel

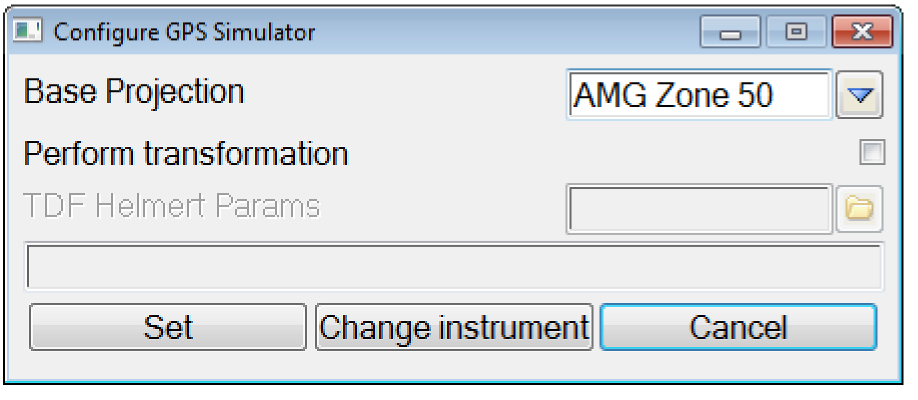
Change instrument button

go back to the instrument selection panel.

Cancel button

exit 12dField

Selecting **Set** displays the next **Configure GPS Simulator** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Base Projection choice box

Select the projection to use, these are located in the carto.4d file. This projection is used to convert the WGS84 lat/long/elevation into cartesian coordinates which can be used directly or have a localisation applied to them.

Perform transformation tick box

if ticked a user defined localisation will be applied to the GPS reading.

12dF Helmert Params

*Select the *.TDF_HEL file containing the transformation details to apply, this file is created with the "GPS Localisation" panel in the main 12DField menu.*

Set button

Enter 12dField

Change instrument button

Go back to the instrument selection panel.

Cancel button

Exit 12dField

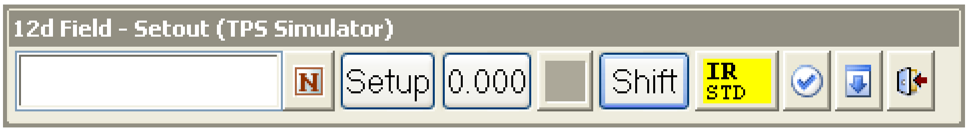
After selecting **Set** *the* **12d Field - Setout GPS Simulator** *is displayed on the screen. For more information please go to*18.2.1.5 12d Field - Setout (GPS Simulator)

Continue to next section 18.2.1.4 12d Field - Setout (TPS Simulator) or return to 18.2 12d Field .

18.2.1.4 12d Field - Setout (TPS Simulator)

This section of documentation is a work in progress and will be updated in subsequent releases.

The control bar has one user definable setting, In the settings panel under the general tab you can toggle on or off whether you want the current string name on the control bar, this is "Str Name on Control Bar", 12dField will need to be restarted for this setting to change.



Names.4d

controls the string details used by 12dField when storing a point, e.g. model/linestyle/weight etc.

Setup

brings up the **Store Point Setup** *panel*.
For more information on **Store Point Setup** *panel*, *please go to* 18.2.1.6.1 Store Point Setup

brings up the **Target Heights Panel**

Show the current number of satellites and status, pressing the button brings up the **GPS Status** *panel. In the settings panel the user defines the thresholds for* ***GPS*** *errors.*

Shift

is used by 12dField panels to change buttons to add extra functionality.

Show the current measuring style, pressing brings up the **GPS/TPS settings** *panel.*

Select button, for tablet PC, allow selection with a press.

Menu

brings up the 12d Field panels menu. For more information please go to 18.2.1.7 Options

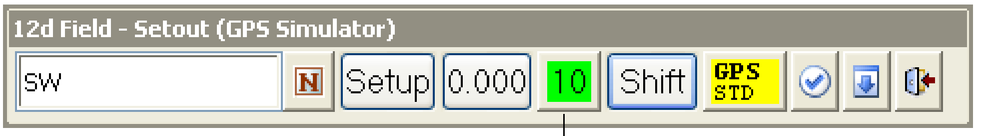
Exit

Exits 12d Field.

Continue to next section 18.2.1.5 12d Field - Setout (GPS Simulator) or return to 18.2 12d Field .

18.2.1.5 12d Field - Setout (GPS Simulator)

This section of documentation is a work in progress and will be updated in subsequent releases.



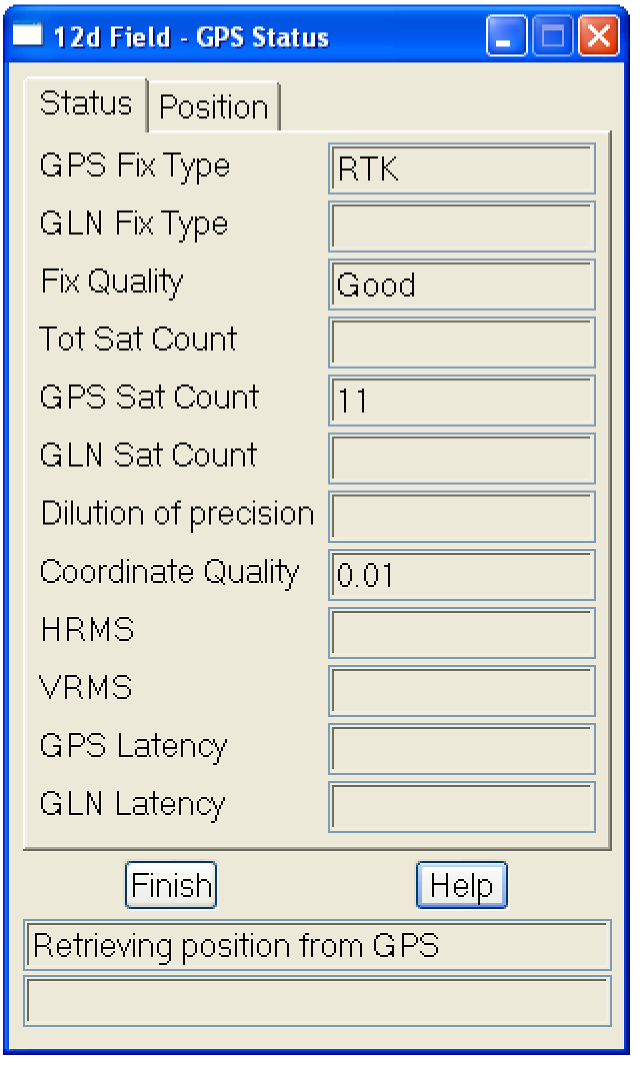
Selecting this option brings up the **12d Field - GPS Status** *panel.*

For information on the **12d Field - GPS Status** panel please go to 18.2.1.6 12d Field - GPS Status

Continue to next section 18.2.1.6 12d Field - GPS Status or return to 18.2 12d Field .

18.2.1.6 12d Field - GPS Status

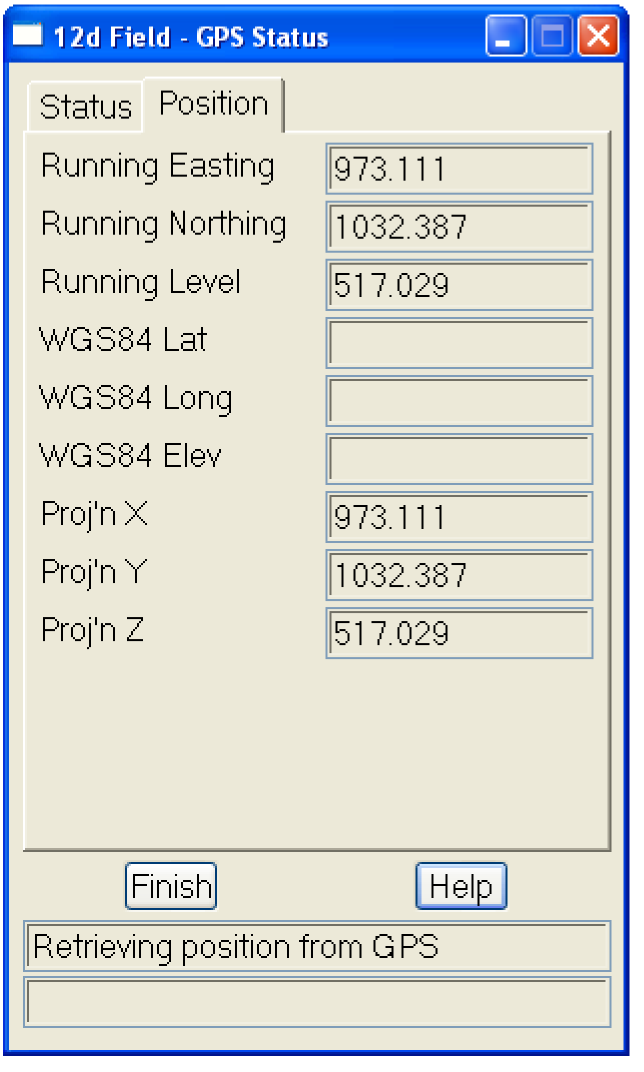
The **12d Field - GPS Status** panel provides information about the current state of the **GPS** instrument such as fix quality and satellites used.



Status Tab

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
GPS Fix Type	12dF input box		
The current fix type of the GPS satellites, e.g RTK			
GLN Fix Type	12dF input box		
The current fix type of the Glonass satellites, e.g RTK			
Fix Quality	12dF input box		
The quality of the GPS, Good, Average, Poor no RTK			
Tot Sat Count	12dF long box		
The number of GPS & Glonass satellites			
GPS Sat Count	12dF long box		
The number of GPS satellites			
GLN Sat Count	12dF long box		
The number of Glonass satellites			
Dilution of precision	12dF double box		
The current dilution of precision			
Coordinate Quality	12dF double box		
Leica specific coordinate quality			
HRMS	12dF double box		
Horizontal Root Mean Square value			
VRMS	12dF double box		
Vertical Root Mean Square value			
GPS Latency	12dF double box		
The latency of the measurements from the GPS satellites.			
GLN Latency	12dF double box		
The latency of the measurements from the Glonass satellites.			



Position Tab

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Running Easting 12dF double box

The current easting of the GPS.

Running Northing 12dF double box

The current northing of the GPS.

Running Level 12dF double box

The current level of the GPS.

WGS84 Lat 12dF angle box

The WGS84 latitude of the running measurement.

WGS84 Long 12dF angle box

The WGS84 longitude of the running measurement.

WGS84 Elev 1 12dF double box

The WGS84 elevation of the running measurement.

Proj'n X 12dF double box

The WGS84 cartesian X of the running measurement.

Proj'n Y 12dF double box

The WGS84 cartesian Y of the running measurement.

Proj'n Z 12dF double box

The WGS84 cartesian Z of the running measurement.

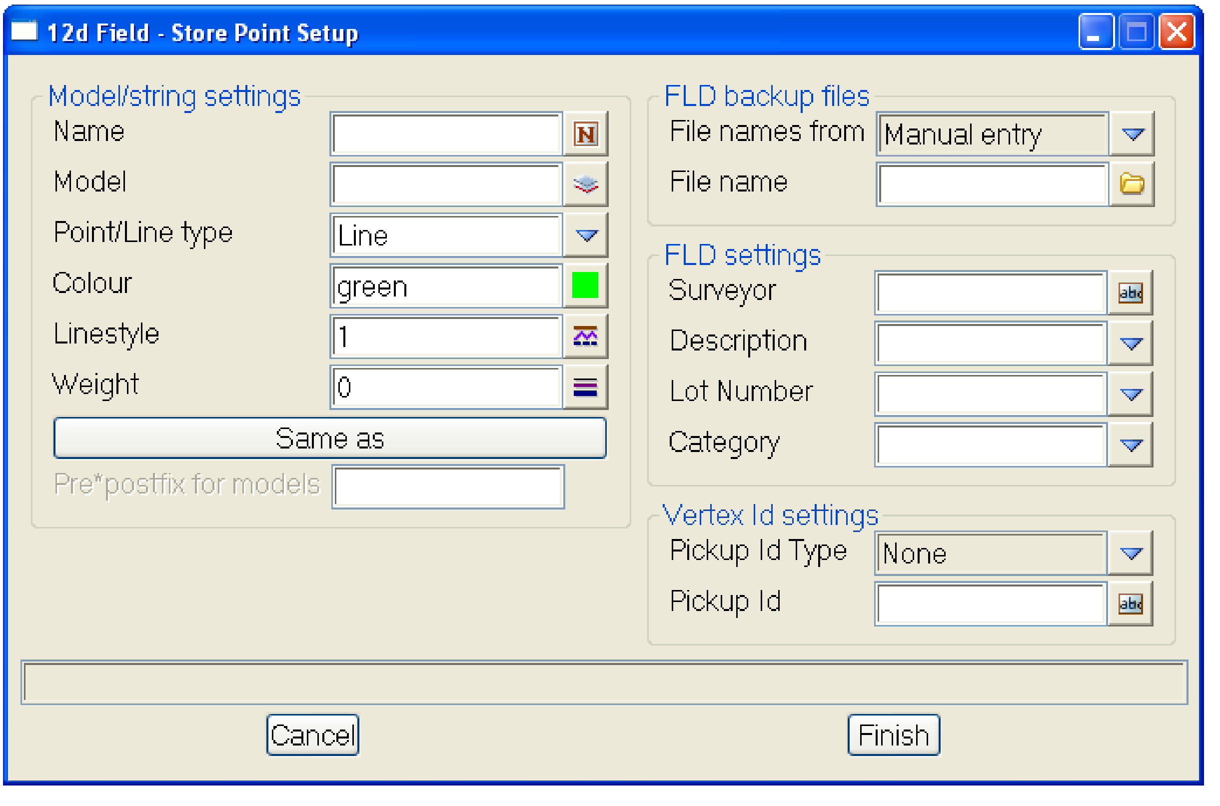
Continue to next section 18.2.1.6.1 Store Point Setup or return to 18.2 12d Field .

18.2.1.6.1 Store Point Setup

The **12d Field - Store Point Setup** panel is called up the 1st time a user attempts to store a point to a model in a 12dField setout panel or if the user presses the **Setup** button on the 12dField control bar.

This panel must be validly completed before a point can be stored.

The panel is grouped into 4 separate areas each described below.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Model/string settings

These settings work the same as your typical 12d settings work, they control the look of the stored string in the model.

Name

enter the name of the string to be stored manually or from the pop up select a predefined name (from names.4d) which will also populate the other settings.

Model model box

the model to store the string in. (This model will automatically be added to your defined 12dField plan view)

Point/Line type choice box Point, Line

whether the string is a point or line string.

Colour colour box

the colour of the string.

Linestyle

linestyle of the string

Weight

the thickness of the string.

Same as button

select an existing point/line and the fields will set to it's properties.

Pre*postfix for models

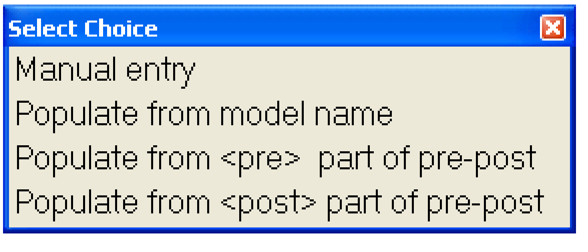
if non blank the pre-postfix will be applied to the nominal model when storing the point.

FLD backup files

All 12dField points are stored to a FLD file as well as a model. This is done for backup reasons as every shot is stored to the FLD file on disk when a shot is taken ensuring no data should be lost in the event of an unexpected shutdown of 12d.

The 12dField files have sufficient attributes in them that they can be read directly back into 12d without going through the survey data reduction functions.

File names from Manual Entry, Populate from model name, Populate from <pre> part of pre post, Populate from <post> part of pre-post



*Manual Entry - enter the name of the file in the **File name** box.*
*Populate from model name - the name of the model in the **model box** is used as the file name.*
Populate from <pre> part of pre-post - the name of the file is the pre part.
Populate from <post> part of pre-post - the name of the file is the post part.

*For example a surveyor might enter a model name of "Asphalt Conformance" in the **model box**.*
*They then for a pre fix enter the year month and day and their initials "20113006MG " and they select "**Populate from model name**" for the FLD file name.*
The model and field file written will be "20113006MG Asphalt Conformance"

FLD Settings

Surveyor

the name of the surveyor

Description

*a description of the survey (Populated from **12dF_JOB_DESCRIPTIONS.4D**)*

Lot number

*a lot number of the survey (Populated from **12dF_JOB_LOT_NUMBERS.4D**)*

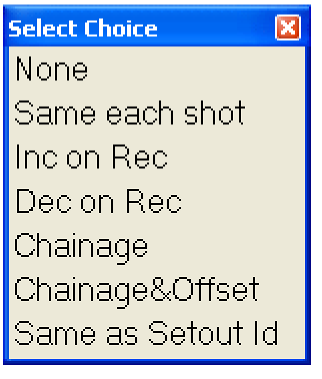
Category

*the category of the survey (Populated from **12dF_JOB_CATEGORIES.4D**)*

Vertex Id settings

This choice box controls the way the id of each vertex in the string is written.

Pickup Id Type None, Same each shot, Inc on Rec, Dec on Rec, Chainage, Chainage&Offset. Same as Setout Id



None - no vertex id is written.
*Same each shot - the value in the **Pickup Id** box will be used for all points stored.*
*Inc on Rec - the value in the **Pickup Id** box will be used for the next point stored then incremented.*
*Dec on Rec - the value in the **Pickup Id** box will be used for the next point stored then decremented.*
Chainage - the chainage of the point being stored will be used as it's vertex id.
Chainage&Offset - the chainage and offset of the point being stored will be used as it's vertex id.
Same as Setout Id - the vertex id of the point being setout will be used.

Notes on the incrementing of the Id.
The id is alphanumeric and the number of characters does not change.The increment/decrement applies to the either the numeric or alpha ending of the id, not to a combination of both.

```
Increment  examples  1->2, 9->0, A1>A2, A9->A0, AA->AB, AZ->BA,
S099->S100, S999->S000
Decrement examples 6->5, 0->9, A2>A1, A0->A9, AB->AA, AA->ZZ,
S100->S099, S000->S999
```

Pickup Id

This is the id of the next vertex to be stored

Cancel button

if cancel is pressed the panel will close but the next time a point is stored it will open again.

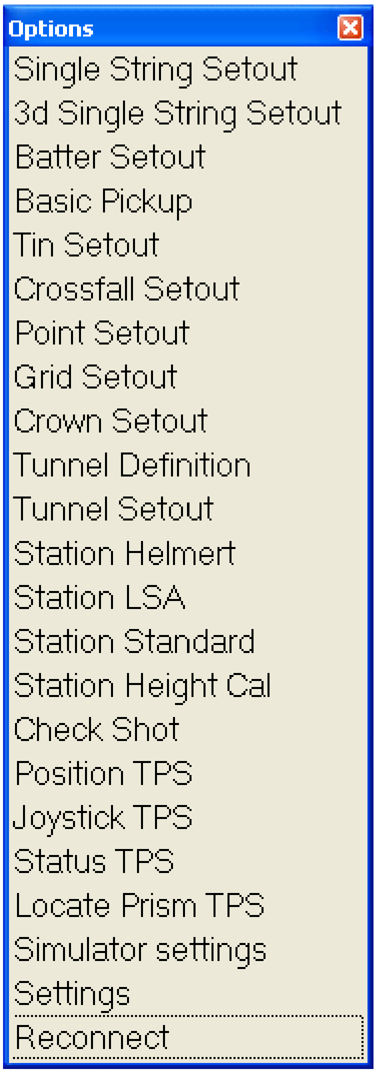
Finish button

if the panel validates correctly it will close and the point will be stored. The panel will only open again if called manually from the control bar or a new setout is started.

Continue to next section 18.2.1.7 Options or return to 18.2 12d Field .

18.2.1.7 Options

This section of documentation is a work in progress and will be updated in subsequent releases.

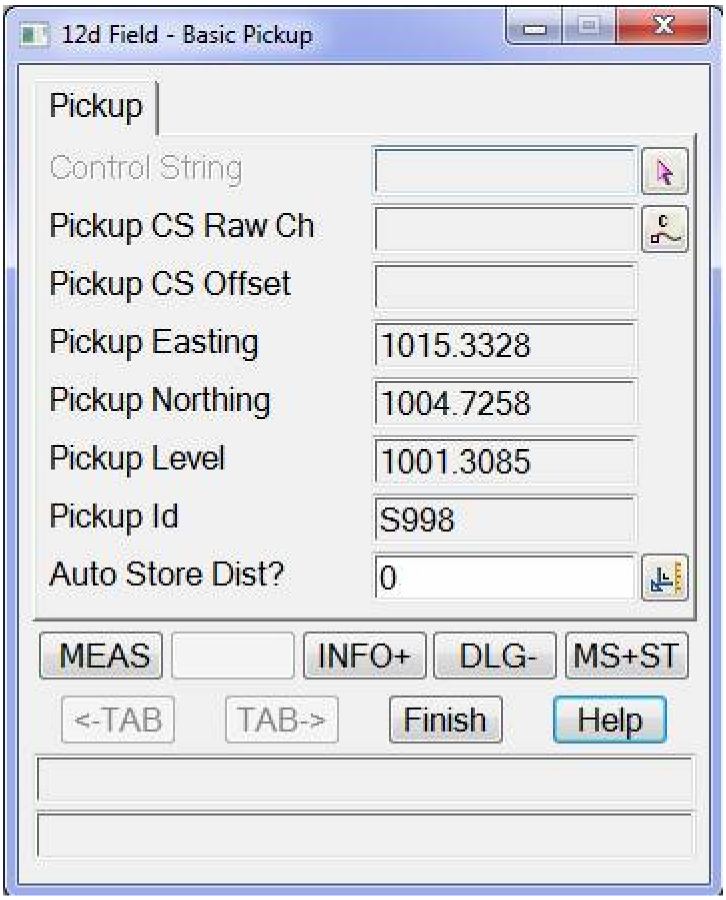


For the option *Single String Setout*, go to 18.2.1.8.1 Single String Setout
Batter Setout 18.2.1.8.2 Batter Setout
Basic Pickup 18.2.1.8.3 12d Field -Basic Pickup
Tin Setout 18.2.1.8.4 Tin Setout
Crossfall Setout 18.2.1.8.5 Crossfall Setout
Point Setout 18.2.1.8.6 Point Setout
Grid Setout 18.2.1.8.7 Grid Setout
Crown Setout 18.2.1.8.8 Crown Setout
Tunnel Definition 18.2.1.8.9 Tunnel Definition
Tunnel Setout 18.2.1.8.10 Tunnel Setout
Station Helmert 18.2.1.8.11 Station Helmert

18.2.1.8 Common Measurement Buttons

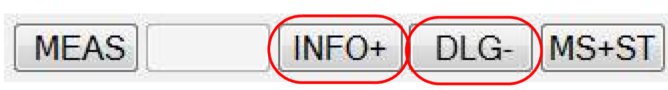
A description of the common measurement buttons on 12dField Setout panels is described below.

The measurement buttons are on a dialog as such:



The INFO and DLG buttons

These buttons do not change in behaviour between all of the different measure modes.

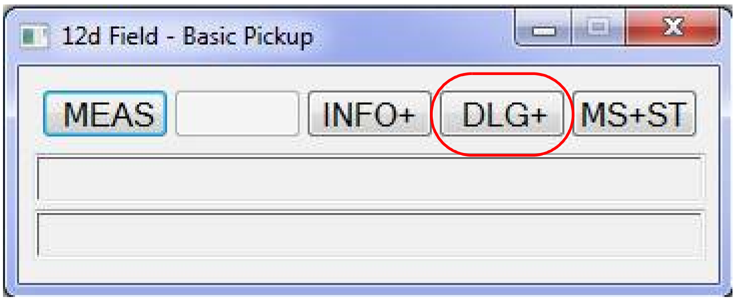


INFO+

*Displays the information panel, this panel contains extra information about the current setout that is not displayed in the standard panel, the contents of the information panel is user configurable via the text file **12dF_INFO_PAGE_CONFIG.4D**. When the information panel is active the button changes to **INFO-**.*

DLG-

*When pressed the dialog minimises to maximise the screen area, the button changes to **DLG+** which when pressed will bring the panel back to full size.*



Single (GPS/TPS) and Multiface (TPS only) measurement modes.

The **MEAS** button.

MEAS

Starts the measurement, on completion the button state changes to this

MEAS

Start another measurement.

STORE

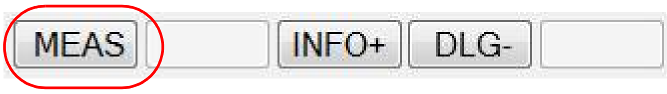
Store the point as per the settings in the 18.2.1.6.1 Store Point Setup *panel, the first store of a point brings this panel up automatically. Once stored the **STORE** button will be blanked out until a new measurement is taken.*

The **MS+ST** button

MS+ST

Starts a measurement and stores the point as per the settings in the 18.2.1.6.1 Store Point Setup *panel. The **STORE** button remains disabled.*

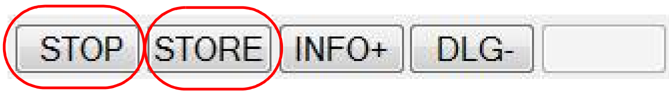
Continuous (GPS/TPS) measurement mode



In continuous mode the **MS+ST** button is disabled.

MEAS

Start continuous measurement



STOP

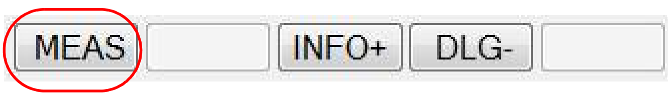
*Stop the continuous measurement, the **STORE** button will remain active until pressed. When the point is stored the button is disabled until a new measurement is started.*

STORE

Store the point as per the settings in the 18.2.1.6.1 Store Point Setup *panel, the button state does not change meaning the user can continue to move storing points at a single press of the button.*

Averaging measurement mode (GPS only)

In averaging mode the **MS+ST** button is disabled.



MEAS

Start the averaging measurement, the details of the averaging, std deviations xy&z are displayed in the 3rd message line of the dialog.

STOP

Stop the averaging measurement.

MEAS

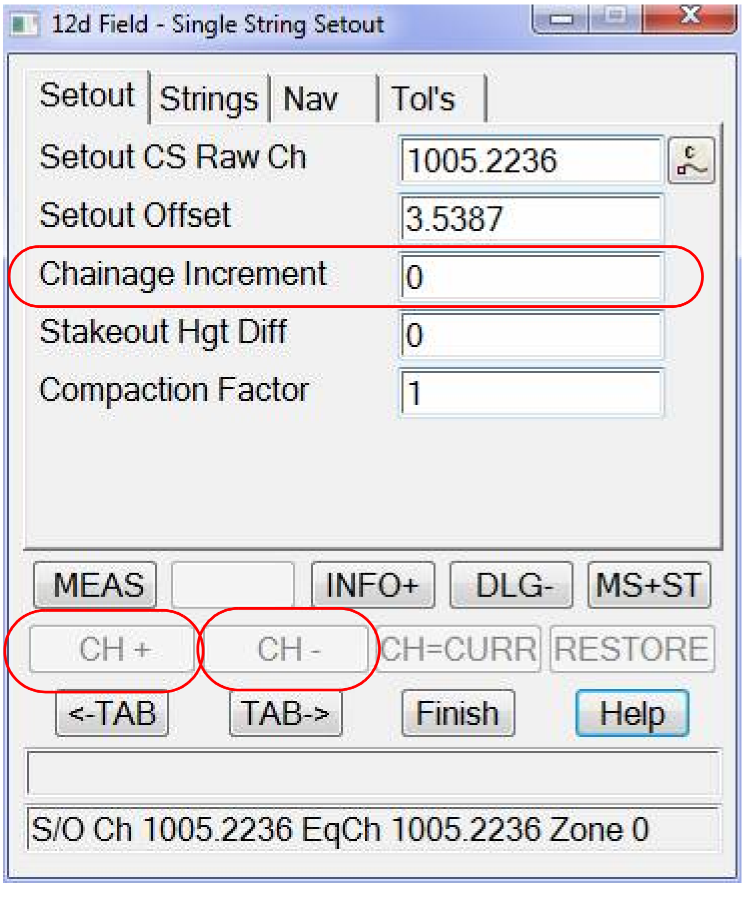
Start a new averaging measurement.

STORE

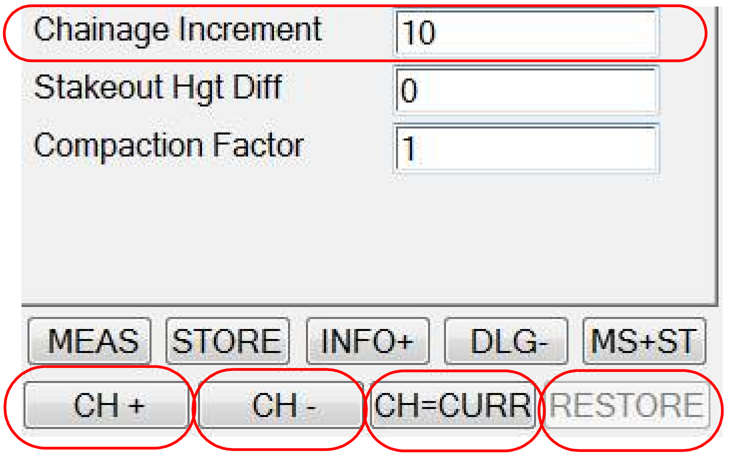
Store the point as per the settings in the 18.2.1.6.1 Store Point Setup *panel.*

The Chainage Buttons

The chainage buttons commonly appear as the middle row of buttons in a 12dField setout panel.



The **CH+** and **CH-** buttons are used to quickly increment to the next chainage to be setout, the buttons are activated when the **Chainage Increment field** in the panel has a value other than 0.



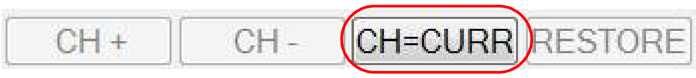
CH+

When pressed the value in the ***Chainage Increment field*** *is added to the current setout chainage. The value in* ***Chainage Increment field*** *can be +ve or -ve.*

CH-

When pressed the value in the ***Chainage Increment field*** *is subtracted from the current setout chainage. The value in* ***Chainage Increment field*** *can be +ve or -ve.*

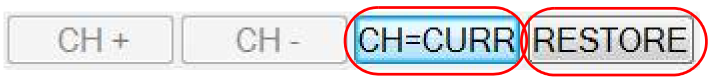
The **CH=CURR** and **RESTORE** buttons are used for where the user has a need to temporarily use the current chainage for setout purposes.



CH=CURR

The button is activated when a measurement has been taken.

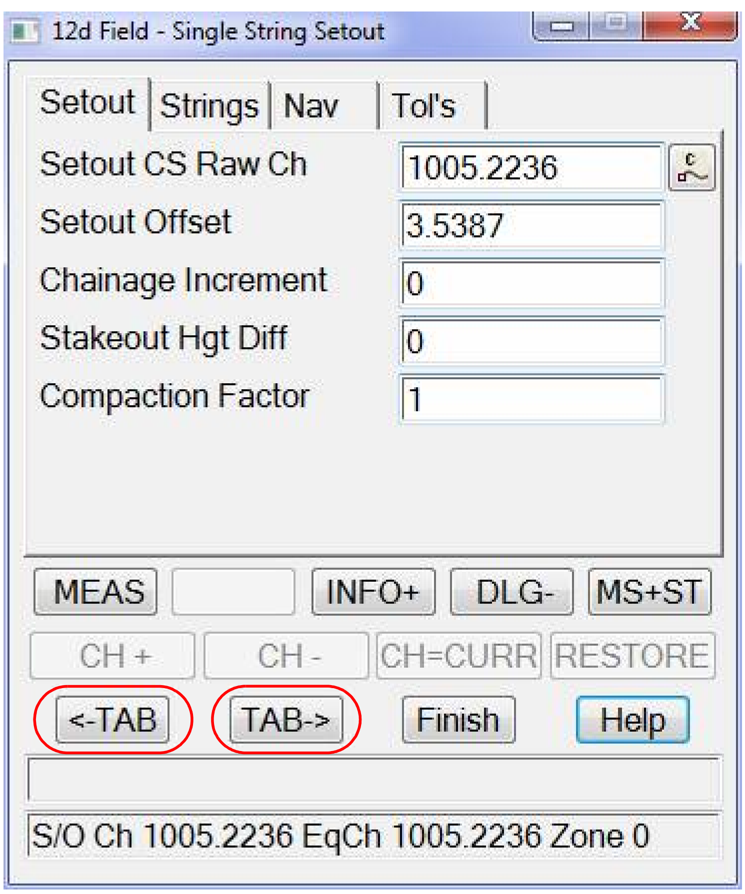
When pressed the setout chainage is set to the current chainage, the **RESTORE** *button is then activated and the* **CH+** and **CH-** *buttons disabled.*



RESTORE

*When pressed the setout chainage is set back to the chainage when **CH=CURR** was pressed and the **CH+ CH-** buttons enabled if possible.*

The Bottom Button Row



<-TAB

changes the active tab in the setout panel to the next left.

TAB->

changes the active tab in the setout panel to the next right.

The **<-TAB** and **TAB->** keys can be overwritten for special user behaviour.

Overwriting the **<-TAB** and **TAB->** keys.

The functionality of the TAB keys can be defined in "**12dF_DLG_USER_KEYS.4D**".

The file allows the user to enter the text for the key and the functionality for the key to take on.

"bt_tab_left" "P2d" "tps_position_hz"

"bt_tab_right" "P XY" "tps_position_xy"

The 1st field is the keyword for the button to override.

The 2nd field is the text for the button to override.

The 3rd field is the action for the button to perform, any of the 12dField hot keys defined in "**12dF_USER_KEYS.4D**" can be assigned.

Continue to next section 18.2.1.8.1 Single String Setout or return to 18.2 12d Field .

18.2.1.8.1 Single String Setout

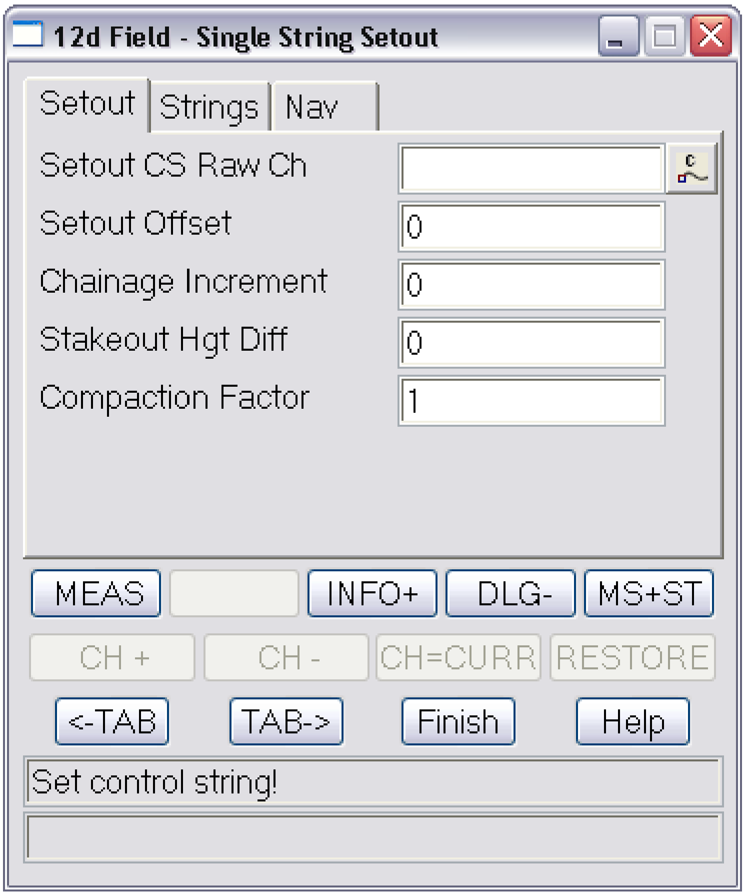
The **12d Field Single String Setout** panel is used when the setout point is relative both horizontally and vertically to one string, for example a traffic island or kerb and gutter.

Single string setout works by dropping a point to a nominated control string, then the setout string is cut normal to the dropped point on the control string.

When setting out the setout string is cut normal from a point at the setout chainage on the control string, then the setout offset is then applied from here to create the setout point.

Note the control string and the setout string could be the same string or all different, this dependant on the task being performed.

Selecting the **Single String Setout** option brings up the **12d Field - Single String Setout** panel.



Setout Tab

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Setout CS Raw Ch 12dF chainage box

The raw, no equalities setout chainage on the control string, start chainage plus distance along string

Setout Offset 12dF double box

The offset from the setout string to setout, +ve is to the right of the string, -ve left

Chainage Increment 12dF double box

The value the setout chainage will be changed by when chainage increment/decrement is called.

Stakeout Hgt Diff 12dF double box

The height diff from the setout surface/string. +ve is above.

Compaction Factor 12dF double box

A compaction factor applied to the delta heights, e.g. if you know say asphalt will compact by 23% and you need to cover this enter the value as 1.23. (Note this widget is optional and only appears if activated in the Settings panel).

Buttons

MEAS button

Start a measurement, the behaviour is determined by the TPS/GPS measurement setting set from the control bar.

INFO+ button

Display the user configurable information panel to view extra information not available on the standard dialog.

DLG- button

Minimises the dialog so only the first 2 rows of buttons are shown.

MS+ST button

Start a measurement and store it on completion.

CH + button

Increment the setout chainage by the value in the chainage increment field.

CH - button

Decrement the setout chainage by the value in the chainage increment field.

CH=CURR button

Set the setout chainage to the chainage of the last measured point.

RESTORE button

*Restore the setout chainage to the chainage prior to the "**CH=CURR**" button being pressed.*

<-TAB button

Go to the previous tab in the dialog.

TAB-> button

Go to the next tab in the dialog.

DLG+ button

Restore a minimised panel to it's full size.

INFO- button

Close the user configurable information panel.

Nav P button

Load a saved navigation page configuration.

Sh M button

Start a measurement with a touch on the screen.

Sh M+S button

Start a measurement with a touch on the screen, store it on completion.

READ button

Load previously saved setout settings for reuse.

SAVE button

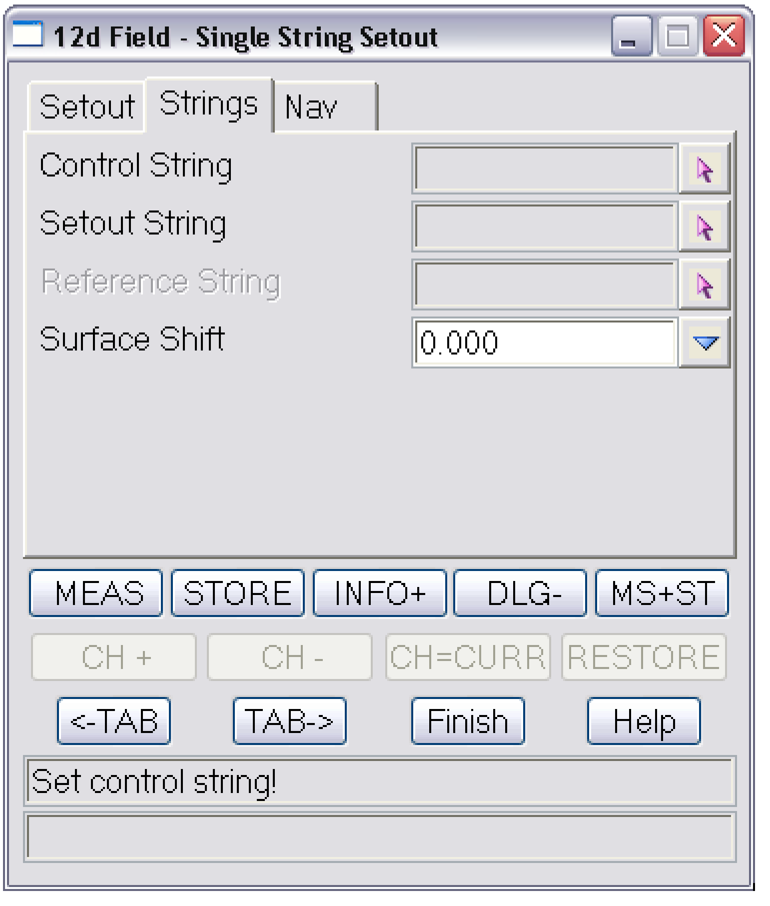
Save the current setout settings for reuse at a later date.

STOP button

Stop the measurement in process.

STORE button

Store the last measured point.



Strings Tab

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Control String	12dF string select box		

Control string, the string to which the other strings are cut normal to for calculations.

Setout String	12dF string select box		

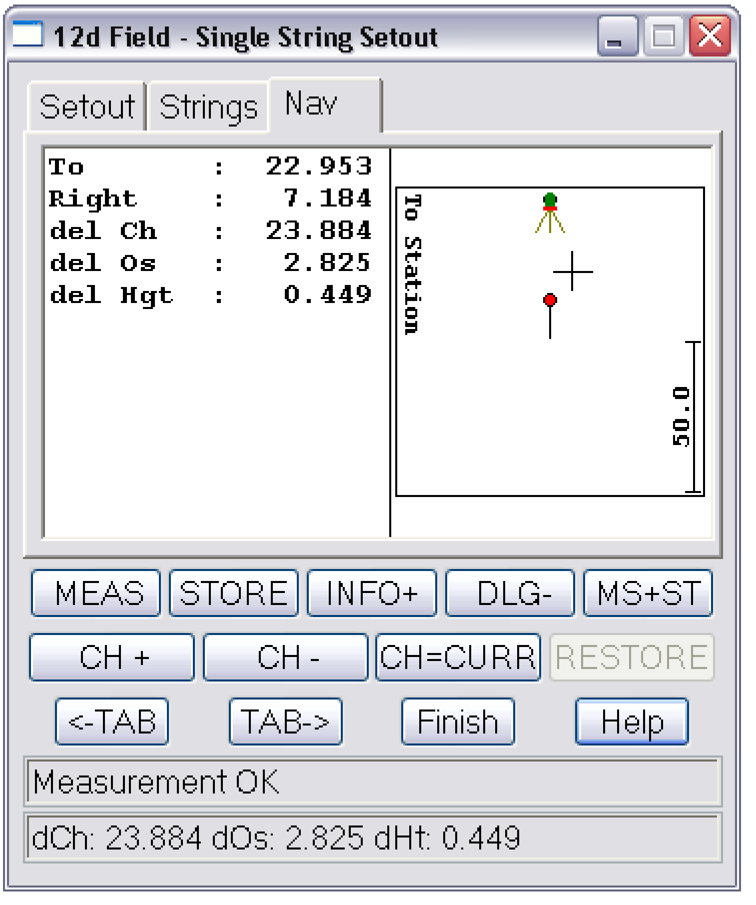
Setout String, setout offset & heights are relative to this string at a point cut normal from the setout chainage on the control string/centreline

Reference String	12dF string select box		

Reference string, a string to which the current point is dropped normally to for information only.

Surface Shift	choice box		-0.0000

*A vertical shift to be applied to the design level, +ve raises the level, can be manually entered or selected from the choice list, (defined in "**TDF_SURFACE_SHIFTS.4D**")*



Nav Tab

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Navigation Box		draw box	

The 12d Field navigation box augments setout by displaying user definable information rows plus a bulls-eye as a visual aid.

Continue to next section 18.2.1.8.2 Batter Setout or return to 18.2 12d Field .

18.2.1.8.2 Batter Setout

The **12d - Field Batter Setout** panel is used to dynamically locate the intersection point of a slope defined by the cut of 2 strings and the natural surface at the users current position.

It is designed around the user wishing to place batter rails in place for guiding the cut/fill.

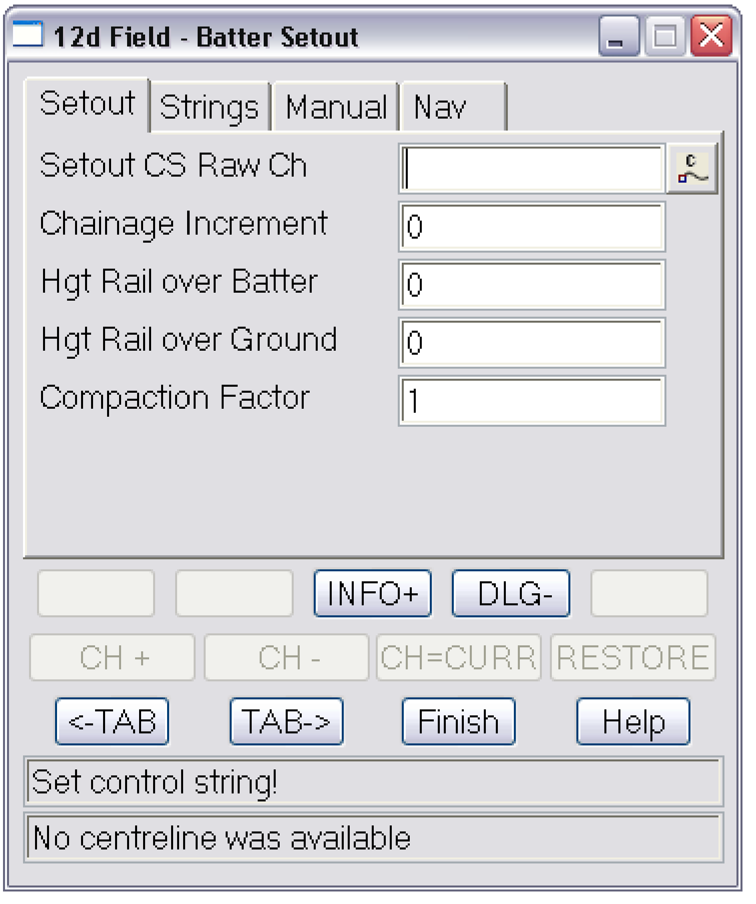
Note, unlike other string setout routines the batter setout does not have a setout string, just the control string and the 2 design strings.

The user is able to set a shift to move the design surface up or down once the strings are cut.

There are manual modes available for setting the design slope when 2 strings are not able to be cut.

Batter setout works by dropping a point to a nominated control string, the 2 strings used to determine the slope are cut normal to the dropped point on the control string, the delta offset for the pole is dependant on the users height rather than a setout string.

Selecting the **Batter Setout** option brings up the **12d - Field Batter Setout** panel.



Setout Tab

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Setout CS Raw Ch	12dF chainage box		

The raw, no equalities setout chainage on the control string, start chainage plus distance along string

Chainage Increment 12dF double box

The value the setout chainage will be changed by when chainage increment/decrement is called.

Hgt Rail over Batter 12dF double box

The height of the batter rail above the design batter, say 0.1 for cut and 1.0 for fill.

Hgt Rail over Ground 12dF double box

The height of the batter rail above the natural ground, typically as large as number as possible to get the rail away from the top/toe of the batter.

Compaction Factor 12dF double box

A compaction factor applied to the delta heights, e.g. if you know say asphalt will compact by 23% and you need to cover this enter the value as 1.23. (Note this widget is optional and only appears if activated the **Settings** *panel.)*

Buttons

MEAS button

*Start a measurement, the behaviour is determined by the **TPS/GPS** measurement setting set from the control bar.*

INFO+ button

Display the user configurable information panel to view extra information not available on the standard dialog

DLG- button

Minimises the dialog so only the first 2 rows of buttons are shown.

MS+ST button

Start a measurement and store it on completion.

CH + button

Increment the setout chainage by the value in the chainage increment field.

CH - button

Decrement the setout chainage by the value in the chainage increment field.

CH=CURR button

Set the setout chainage to the chainage of the last measured point.

RESTORE button

*Restore the setout chainage to the chainage prior to the **"CH=CURR"** button being pressed.*

<-TAB button

Go to the previous tab in the dialog.

TAB-> button

Go to the next tab in the dialog.

STOP button

Stop the measurement in process.

STORE button

Store the last measured point.

Nav P button

Load a saved navigation page configuration.

Sh M button

Start a measurement with a touch on the screen.

Sh M+S button

Start a measurement with a touch on the screen, store it on completion.

READ button

Load previously saved setout settings for reuse.

SAVE button

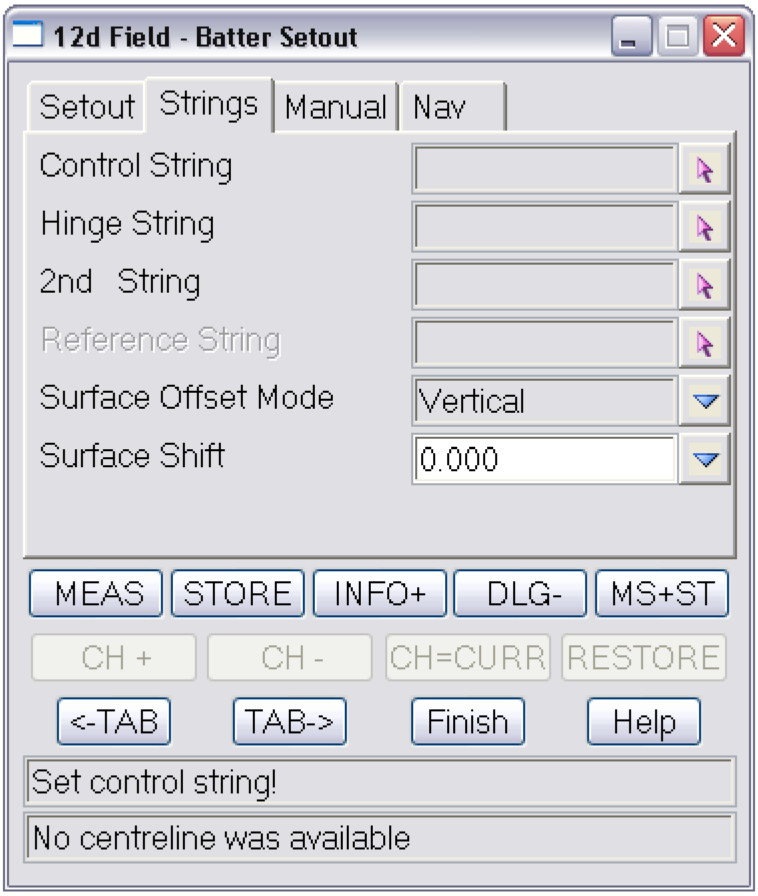
Save the current setout settings for reuse at a later date.

DLG+ button

Restore a minimised panel to it's full size.

INFO- button

Close the user configurable information panel.



Strings Tab

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Control String	12dF string select box		

Control string, the string to which the other strings are cut normal to for calculations.

Hinge String	12dF string select box		

The hinge string is the string the batter is being cut or filled to. Slope distances etc. are given to this string.

2nd String	12dF string select box		

The other string along with the hinge string defining the batter slope.

Reference String	12dF string select box		

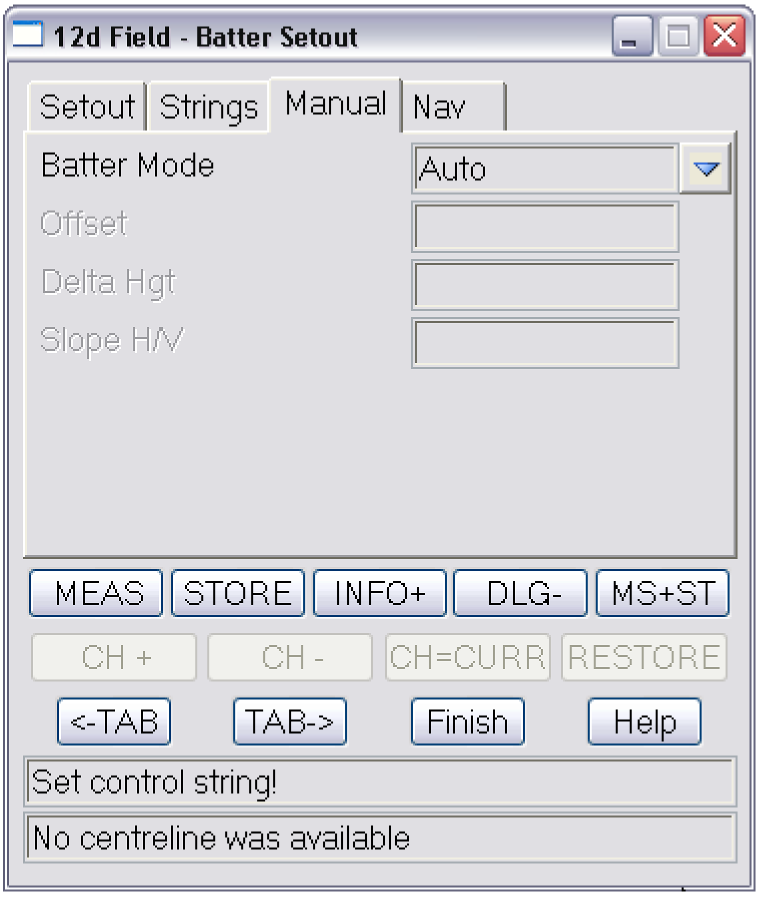
Reference string, a string to which the current point is dropped normally to for information only.

Surface Offset Mode	choice box		vertical, normal

Whether the offset to the surface is normal or vertical
Vertical: *The height offset is applied vertically to the design slope.*
Normal: *The height offset is applied normal/perpendicular to the design slope.*

Surface Shift	choice box		0.000

*A vertical shift to be applied to the design level, +ve raises the level, can be manually entered or selected from the choice list, (defined in "**TDF_SURFACE_SHIFTS.4D**")*



Manual Tab

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Batter Mode	choice box		Auto Centreline, hinge & slope Manual relative (dOS&dHgt), Manual absolute (dOS&RL) Manual, use current slope

Choose the method to generate the batter slope.

***Auto:** The batter slope is determined by cutting the hinge and secondary string.*

***Centreline, hinge & slope:** The batter setout is done by cutting the hinge and applying a manually entered slope from this.*

***Manual relative (dOS&dHgt):** The user manually enters the hinge offset from the control line, the hinge height relative to the control line and the batter slope.*

***Manual absolute (dOS&RL)**: The user manually enters the hinge offset from the control line, the absolute hinge height and the batter slope.*

***Manual, use current slope**: The batter setout is done by cutting the hinge and applying the last measured slope.*

Offset 12dF double box

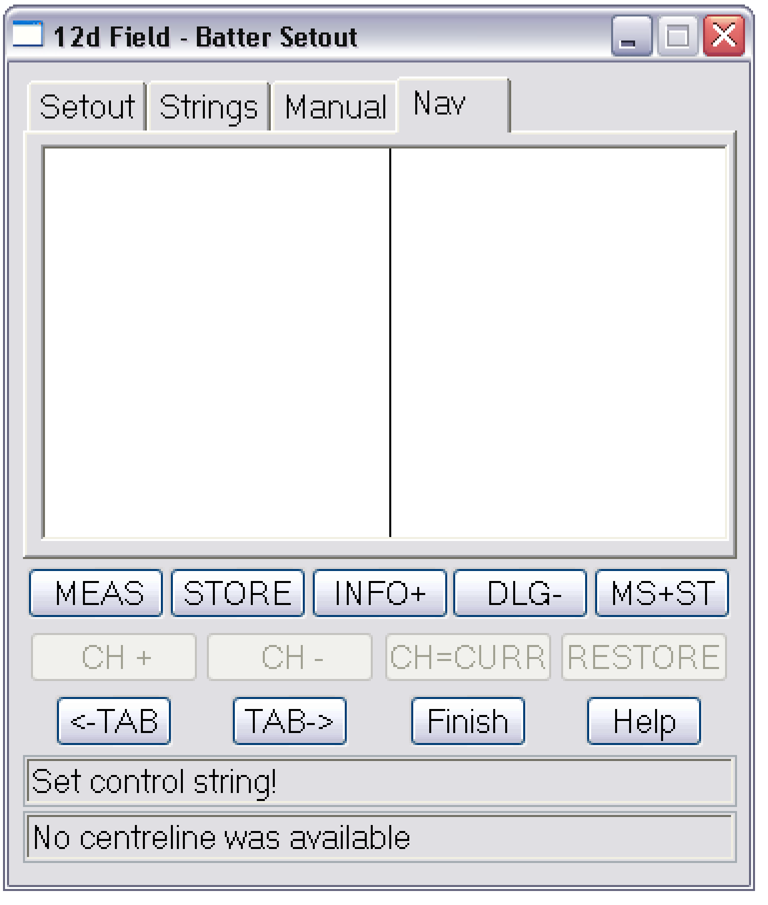
*Dependant on the '**Batter Mode**' setting in use, see description of the options in '**Batter Mode**'.*

Delta Hgt/Absolute Height 12dF double box

*Dependant on the '**Batter Mode**' setting in use, see description of the options in '**Batter Mode**'.*

Slope H/V 12dF double box

The slope of the batter in horizontal/vertical from, e.g. 2:1



Nav Tab

The fields and buttons used in this panel have the following functions.

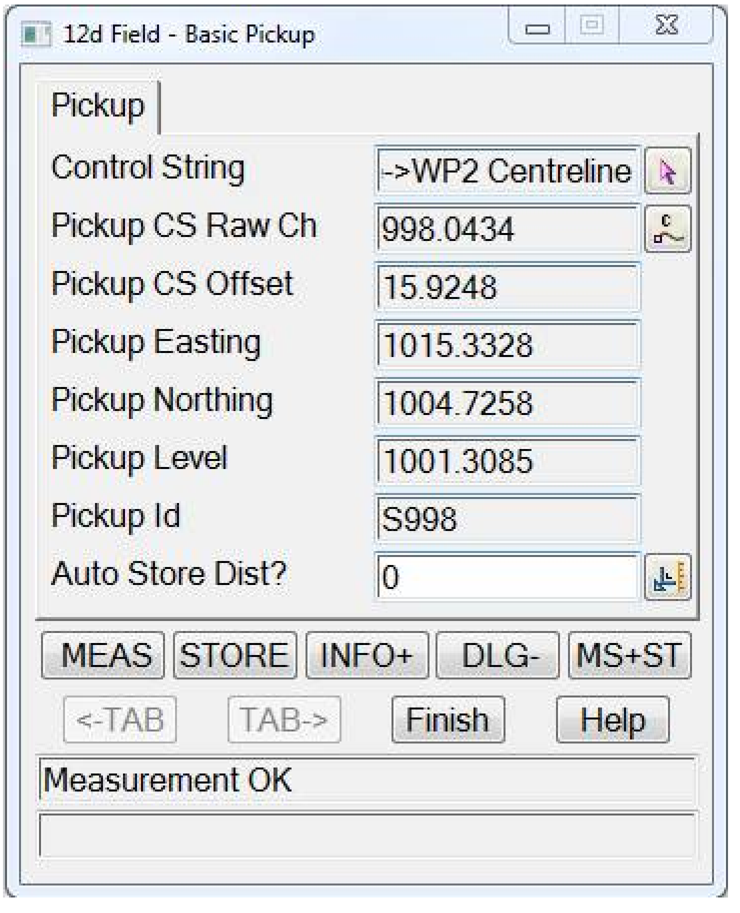
Field Description	Type	Defaults	Pop-Up
Navigation Box	draw box		

The 12d Field navigation box augments setout by displaying user definable information rows plus a bulls-eye as a visual aid.

Continue to next section 18.2.1.8.3 12d Field -Basic Pickup or return to 18.2 12d Field .

18.2.1.8.3 12d Field -Basic Pickup

The **12d Field - Basic Pickup** panel enables the user to pickup and store points. Basic Pickup is only for points and simple lines, it has no edit facilities, recalculations for target height corrections etc.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Control String 12dF string select box

If selected the chainage and offset of the current point on this string will be displayed.

Pickup CS Raw Ch 12dF chainage box

The raw/non equality chainage of the current point on the control string.

Pickup CS Offset 12dF double box

The offset of the current point from the control string, +ve is right

Pickup Easting 12dF double box

The measured easting.

Pickup Northing 12dF double box

The measured northing.

Pickup Level 12dF double box

The measured level.

Pickup Id 12dF input box

The Id of the measured point, the behaviour of the Id is controlled from the 18.2.1.6.1 Store Point Setup *panel.*

Auto Store Dist? 12dF double box

If non zero the distance between shots to automatically store points, the measurement mode must be continuous.

Please see 18.2.1.8 Common Measurement Buttons for information on the buttons.

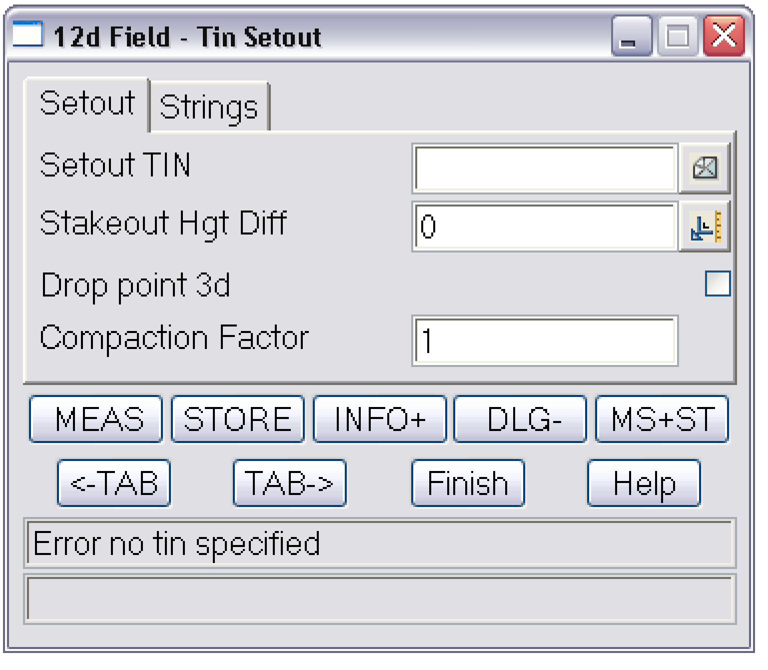
Continue to next section 18.2.1.8.4 Tin Setout or return to 18.2 12d Field .

18.2.1.8.4 Tin Setout

The **12d Field - Tin Setout** panel is used when the user needs to get design levels from a tin.

The point can be dropped either vertically to the tin or 3d/normal to the tin.

Selecting the **Tin Setout** option brings up the **12d Field - Tin Setout** panel.



Setout Tab

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Setout TIN	12dF tin box		

The TIN to use for setting out

Stakeout Hgt Diff 12dF double box

The height diff from the setout surface/string. +ve is above.

Drop point 3d named tick box

When this is toggled on the point is dropped normal to the tin.

In the event the point cannot be dropped to triangle face it will try drop to the edge of a triangle that lies inside the 2d boundary of the tin.

Compaction Factor 12dF double box

A compaction factor applied to the delta heights, e.g. if you know say asphalt will compact by 23% and you need to cover this enter the value as 1.23. (Note this widget is optional and only appears if activated in the Settings panel).

Buttons

MEAS button

Start a measurement, the behaviour is determined by the TPS/GPS measurement setting set from the control bar.

INFO+ button

Display the user configurable information panel to view extra information not available on the standard dialog

DLG- button

Minimises the dialog so only the first 2 rows of buttons are shown.

MS+ST button

Start a measurement and store it on completion.

<-TAB button

Go to the previous tab in the dialog.

TAB-> button

Go to the next tab in the dialog.

STOP button

Stop the measurement in process.

STORE button

Store the last measured point.

Nav P button

Load a saved navigation page configuration.

Sh M button

Start a measurement with a touch on the screen.

Sh M+S button

Start a measurement with a touch on the screen, store it on completion.

READ button

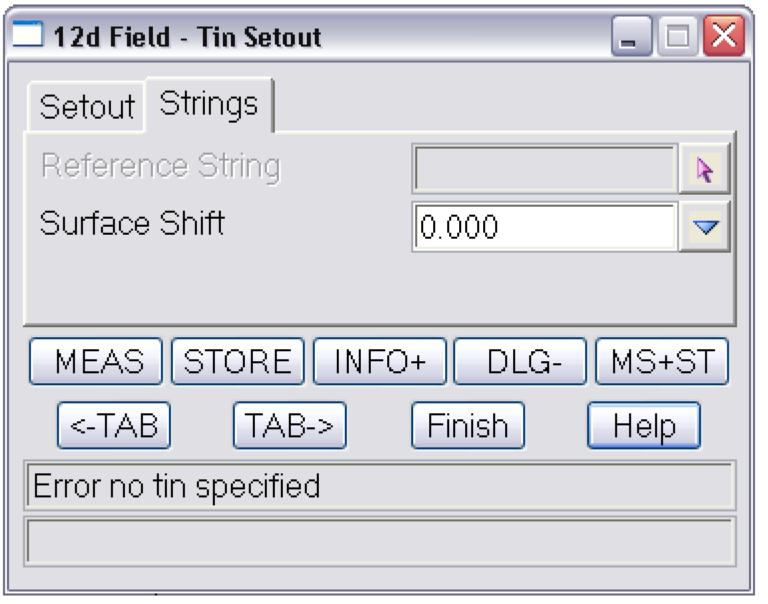
Load previously saved setout settings for reuse.

SAVE button

Save the current setout settings for reuse at a later date.

INFO- button

Close the user configurable information panel.



Strings Tab

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Reference String	12dF string select box		

Reference string, a string to which the current point is dropped normally to for information only.

Surface Shift	choice box		0.000

*A vertical shift to be applied to the design level, +ve raises the level, can be manually entered or selected from the choice list, (defined in "**TDF_SURFACE_SHIFTS.4D**")*

Continue to next section 18.2.1.8.5 Crossfall Setout or return to 18.2 12d Field .

18.2.1.8.5 Crossfall Setout

The **12d Field** - **Crossfall Setout** panel is used when the user wants to generate their design height by cutting 2 strings and projecting the plane of the cuts to their position.

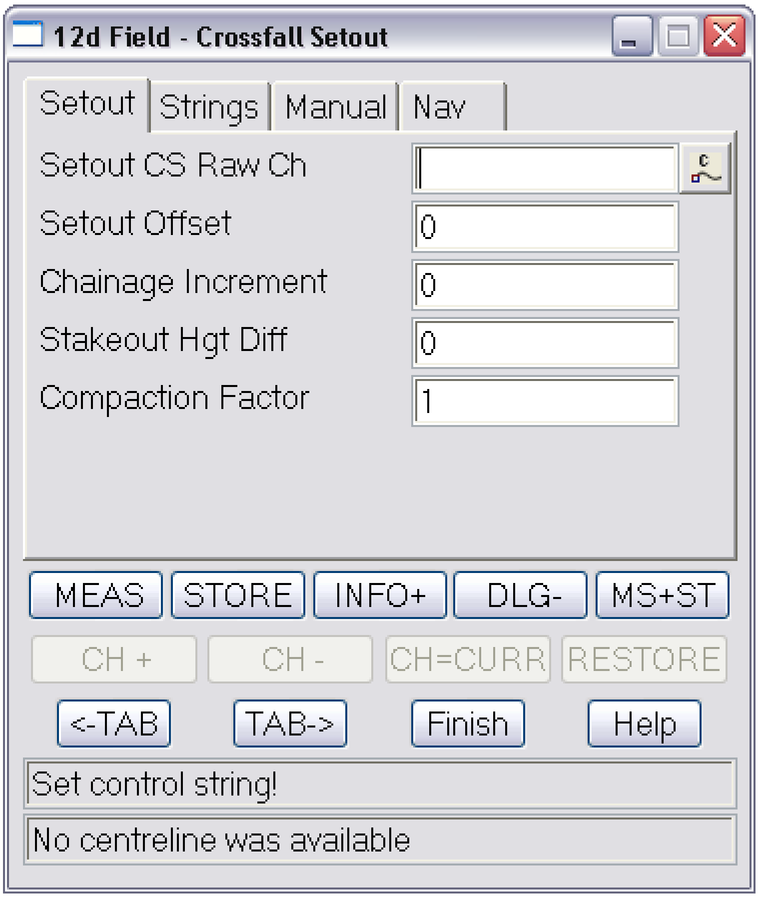
The user is able to set a shift to move the design surface up or down once the strings are cut.

There are manual modes available of setting the design crossfall when 2 strings are not able to be cut.

Crossfall Setout works by dropping a point to a nominated control string, then a setout string and the 2 strings used to determine the crossfall are cut normal to the dropped point on the control string.

Note the control string, the setout string and 1 of the level strings could be the same string or all different, this dependant on the task being performed.

Selecting the **Crossfall Setout** option brings up the **12d Field** - **Crossfall Setout** panel.



Setout Tab

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Setout CS Raw Ch	12dF chainage box		

The raw, no equalities setout chainage on the control string, start chainage plus distance along string

Setout Offset 12dF double box

The offset from the setout string to setout, +ve is to the right of the string, -ve left

Chainage Increment 12dF double box

Chainage Increment

Stakeout Hgt Diff 12dF double box

The height diff from the setout surface/string. +ve is above.

Compaction Factor 12dF double box

*A **Compaction Factor** applied to the delta heights, e.g. if you know say asphalt will compact by 23% and you need to cover this enter the value as 1.23. (Note this widget is optional and only appears if activated in the Settings panel).*

Buttons

MEAS button

Start a measurement, the behaviour is determined by the TPS/GPS measurement setting set from the control bar.

INFO+ button

Display the user configurable information panel to view extra information not available on the standard dialog

DLG- button

Minimises the dialog so only the first 2 rows of buttons are shown.

MS+ST button

Start a measurement and store it on completion.

CH + button

Increment the setout chainage by the value in the chainage increment field.

CH - button

Decrement the setout chainage by the value in the chainage increment field.

CH=CURR button

Set the setout chainage to the chainage of the last measured point.

RESTORE button

*Restore the setout chainage to the chainage prior to the "**CH=CURR**" button being pressed.*

<-TAB button

Go to the previous tab in the dialog.

TAB-> button

Go to the next tab in the dialog.

STORE button

Store the last measured point.

Nav P button

Load a saved navigation page configuration.

Sh M button

Start a measurement with a touch on the screen.

Sh M+S button

Start a measurement with a touch on the screen, store it on completion.

READ button

Load previously saved setout settings for reuse.

SAVE button

Save the current setout settings for reuse at a later date.

DLG+ button

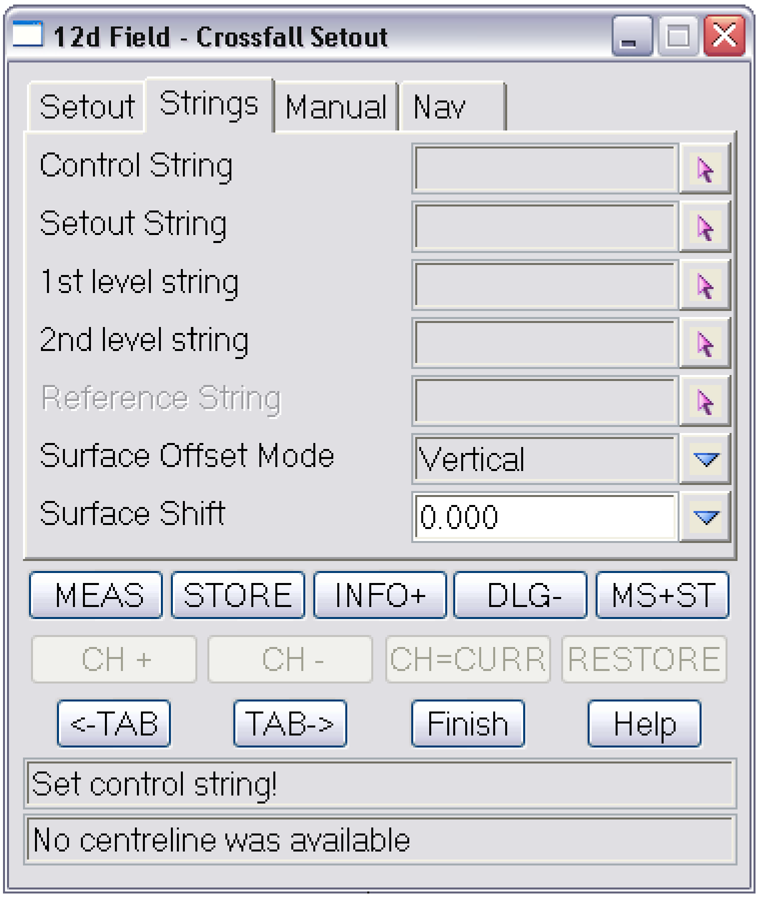
Restore a minimised panel to it's full size.

INFO- button

Close the user configurable information panel.

STOP button

Stop the measurement in process.



Strings Tab

The fields and buttons used in this panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Control String | 12dF string select box

Control string, the string to which the other strings are cut normal to for calculations.

Setout String | 12dF string select box

Setout String, setout offset & heights are relative to this string at a point cut normal from the setout chainage on the control string/centreline

1st level string | 12dF string select box

Level string 1, the string cut for one of the design heights.

2nd level string | 12dF string select box

Level string 2, the string cut for a 2nd design height.

Reference String | 12dF string select box

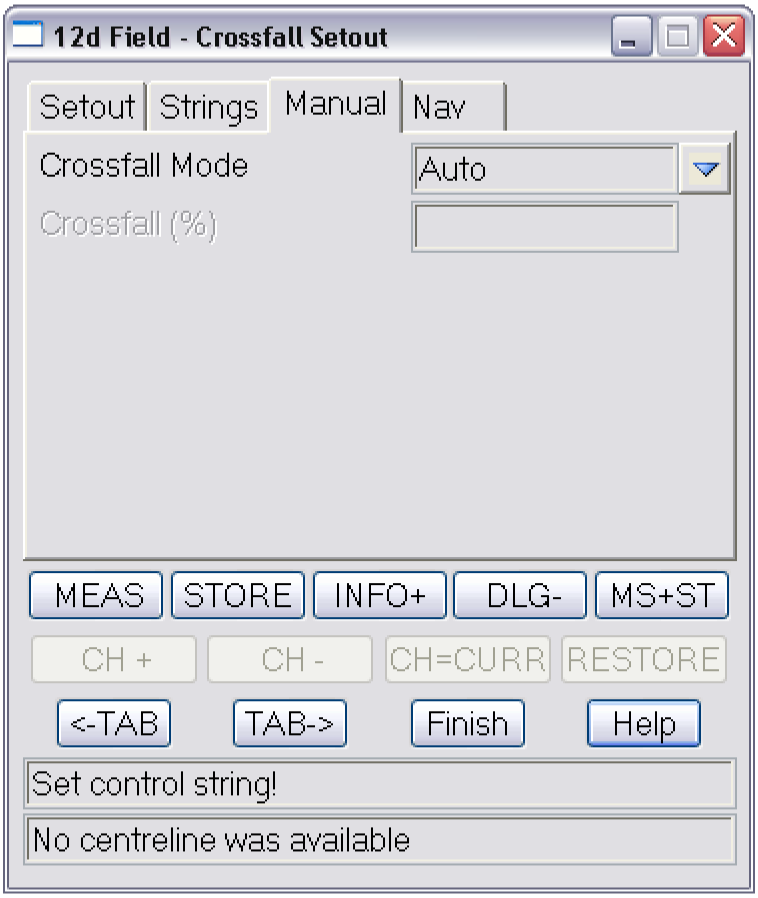
Reference string, a string to which the current point is dropped normally to for information only.

Surface Offset Mode | choice box | Vertical, Normal

Whether the offset to the surface is normal or vertical
***Vertical**: The height offset is applied vertically to the design surface.*
***Normal**: The height offset is applied normal/perpendicular to the design surface.*

Surface Shift | choice box

*A vertical shift to be applied to the design level, +ve raises the level, can be manually entered or selected from the choice list, (defined in "**TDF_SURFACE_SHIFTS.4D**").*



Manual Tab

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Crossfall Mode	choice box		Auto, Centreline, string & x-fall, Manual, use current x-fall

The method used to generate the design crossfall

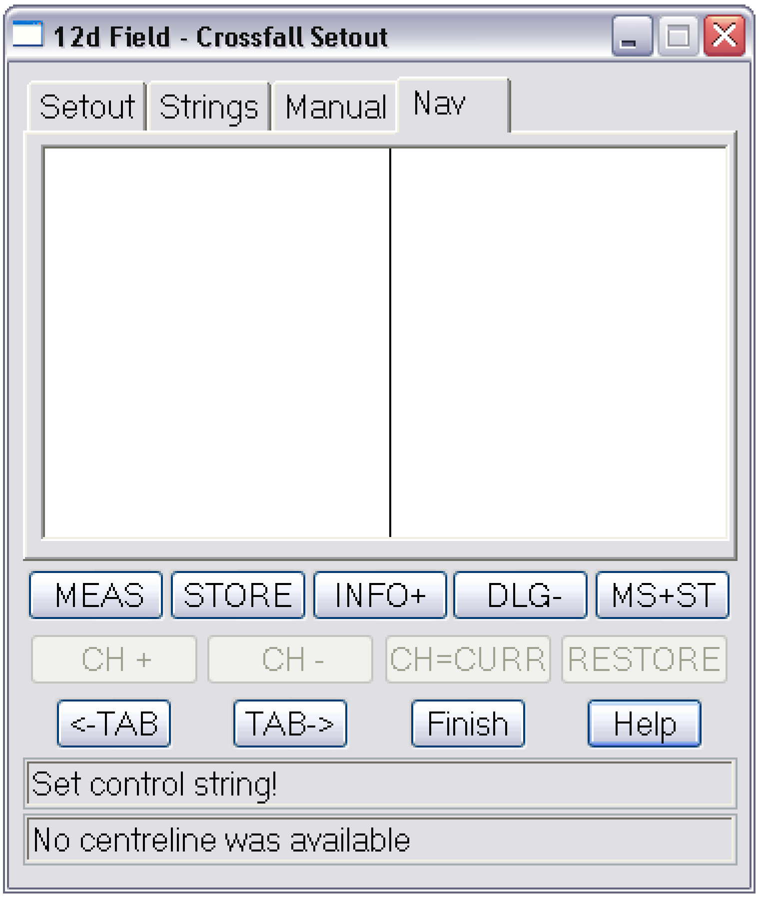
Auto: *The crossfall is calculated by cutting the 1st and 2nd strings.*

Centreline, string & x-fall: *The crossfall is entered manually, only the 1st string is cut and the crossfall projected from this.*

Manual, use current x-fall: *The crossfall is entered manually defaulting to last calculated crossfall, only the 1st string is cut and the crossfall projected from this.*

Crossfall (%)	12dF double box		

The manual crossfall entered by the user in the crossfall setout routine



Nav Tab

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Navigation Box	draw box		

The 12d Field navigation box augments setout by displaying user definable information rows plus a bulls-eye as a visual aid.

Continue to next section 18.2.1.8.6 Point Setout or return to 18.2 12d Field .

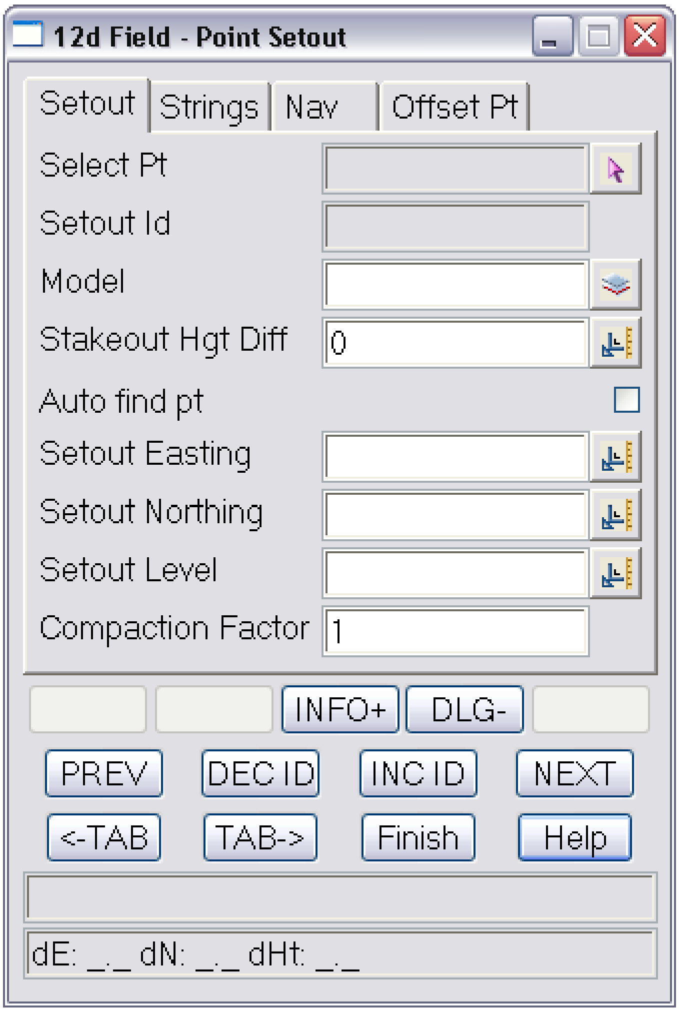
18.2.1.8.6 Point Setout

The **12d Field - Point Setout** panel enables the user to setout individual points, optionally referenced to a control string.

The point can be either selected from a view, be found automatically or be manually entered.

The panel also allows the offsetting of the point by a defined bearing and distance.

Selecting the **Point Setout** option brings up the **12d Field - Point Setout** panel.



Setout Tab

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Point Selection

The point can be picked from a view, in this case the Id and Model boxes will be automatically filled out.

The point id can be manually changed but in this case must be unique in the model.

If auto find is on the point will be set to the closest point in the setout model to the current position in the field

Select Pt 12dF new select box

Select the point to setout, will update the Point Id and Model boxes

Setout Id 12dF input box

The id/name of the setout point.

Model model box

The model containing the setout point.

Stakeout Hgt Diff 12dF double box

The height diff from the setout surface/string. +ve is above.

Auto find pt named tick box

Auto find pt

Setout Easting, Setout Northing, Setout Level

The coordinate boxes are only added to the panel if ticked on under survey in the settings panel.

Setout Easting 12dF double box

The easting to setout.

Setout Northing 12dF double box

The northing to setout.

Setout Level 12dF double box

The height to setout.

Compaction Factor 12dF double box

A compaction factor applied to the delta heights, e.g. if you know say asphalt will compact by 23% and you need to cover this enter the value as 1.23. (Note this widget is optional and only appears if activated in the Settings panel.)

Buttons

MEAS button

Start a measurement, the behaviour is determined by the TPS/GPS measurement setting set from the control bar.

INFO+ button

Display the user configurable information panel to view extra information not available on the standard dialog

DLG- button

Minimises the dialog so only the first 2 rows of buttons are shown.

MS+ST button

Start a measurement and store it on completion.

NEXT button

Make the setout point the next point in the current string.

PREV button

Make the setout point the previous point in the current string.

INC ID button

Increment the point ID and make this the setout point. E.g. BOLT30 -> BOLT31

DEC ID button

Decrement the point ID and make this the setout point. E.g. BOLT31 -> BOLT30

<-TAB button

Go to the previous tab in the dialog.

TAB-> button

Go to the next tab in the dialog.

STOP button

Stop the measurement in process.

STORE button

Store the last measured point.

Nav P button

Load a saved navigation page configuration.

Sh M button

Start a measurement with a touch on the screen.

Sh M+S button

Start a measurement with a touch on the screen, store it on completion.

READ button

Load previously saved setout settings for reuse.

SAVE button

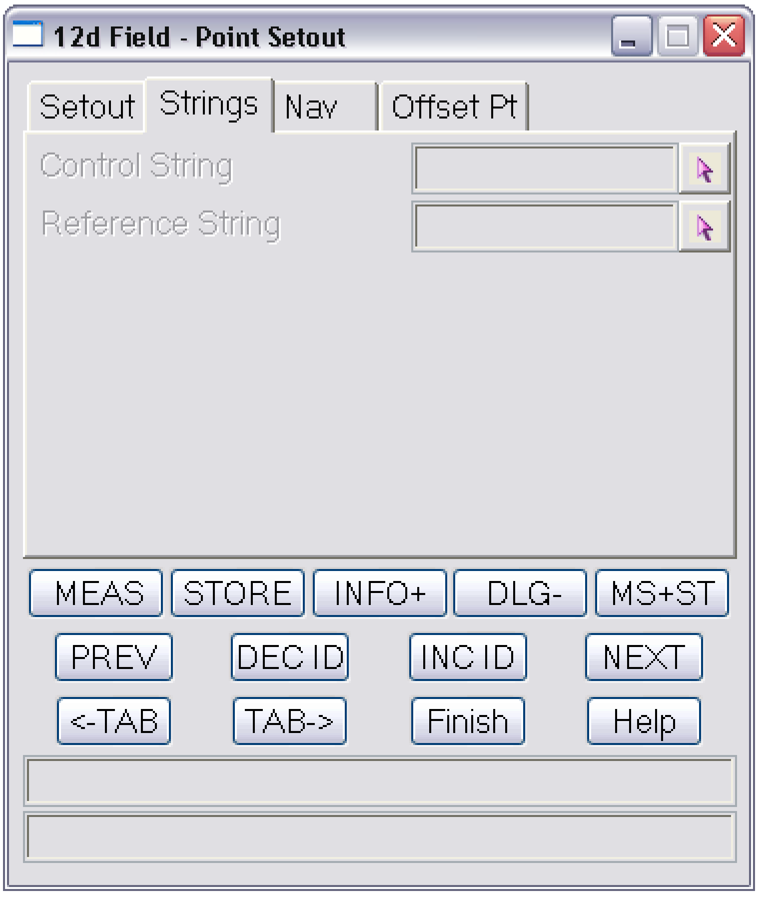
Save the current setout settings for reuse at a later date.

DLG+ button

Restore a minimised panel to it's full size.

INFO- button

Close the user configurable information panel.



Strings Tab

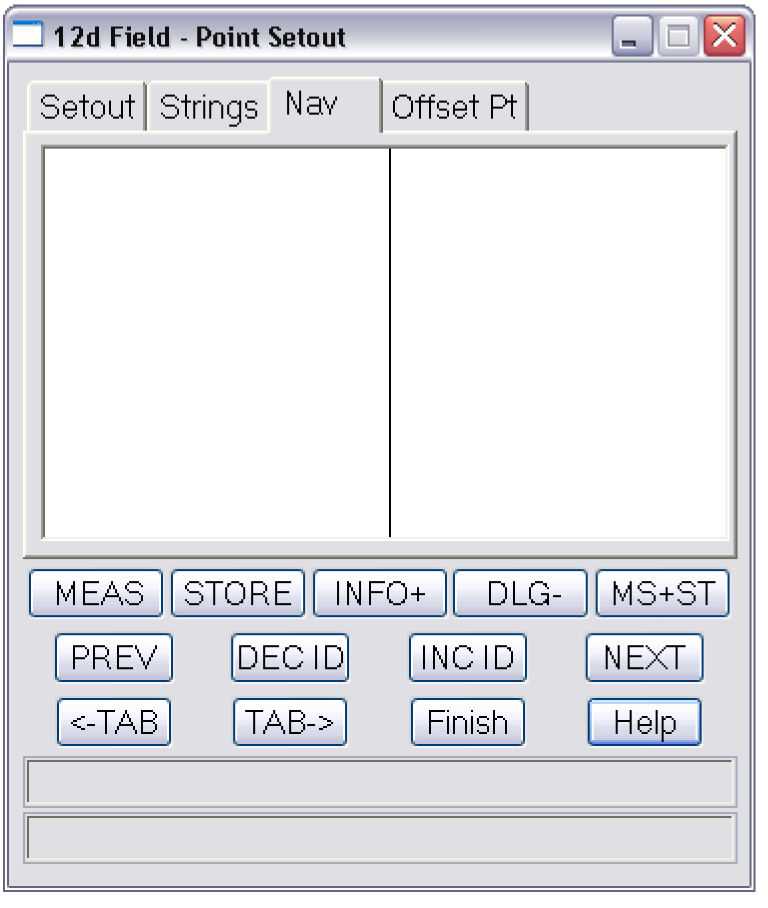
The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Control String	12dF new select box		

Control string, the string to which the other strings are cut normal to for calculations.

Reference String	12dF new select box		

Reference string, a string to which the current point is dropped normally to for information only.

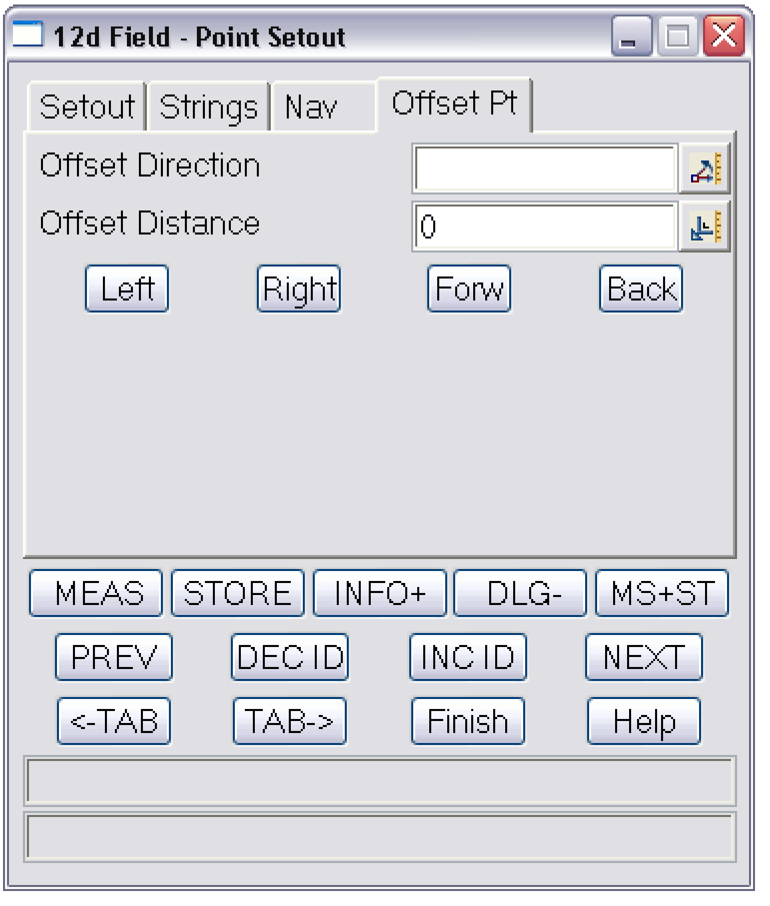


Nav Tab

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Navigation Box	draw box		

The 12d Field navigation box augments setout by displaying user definable information rows plus a bulls-eye as a visual aid.



Offset Pt Tab

The **Offset Point** tab allows the current setout point to be offset in a defined direction and distance.

For example if the offset direction was 45° and the offset distance 5m pressing Left would move the point 5m at 315°.

If a control string is nominated then when the point is originally selected it is dropped to the control string and the bearing at this point is the default for the offset direction.

For example, to place offset pegs to a drainage pit select the road centreline and then select the pit. The bearing is that of the road, enter the offset distance and press Right or Left and the point is updated.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Offset Direction	12dF angle box		

Direction to offset base point in.

Offset Distance	12dF double box		

Distance to offset base point by.

Left	button		

Offset the point by the offset distance to the left of the offset direction.

Right	button		

Offset the point by the offset distance to the right of the offset direction.

Forw button

Offset the point by the offset distance in the offset direction.

Back button

Offset the point by the offset distance against the offset direction.

Continue to next section 18.2.1.8.7 Grid Setout or return to 18.2 12d Field .

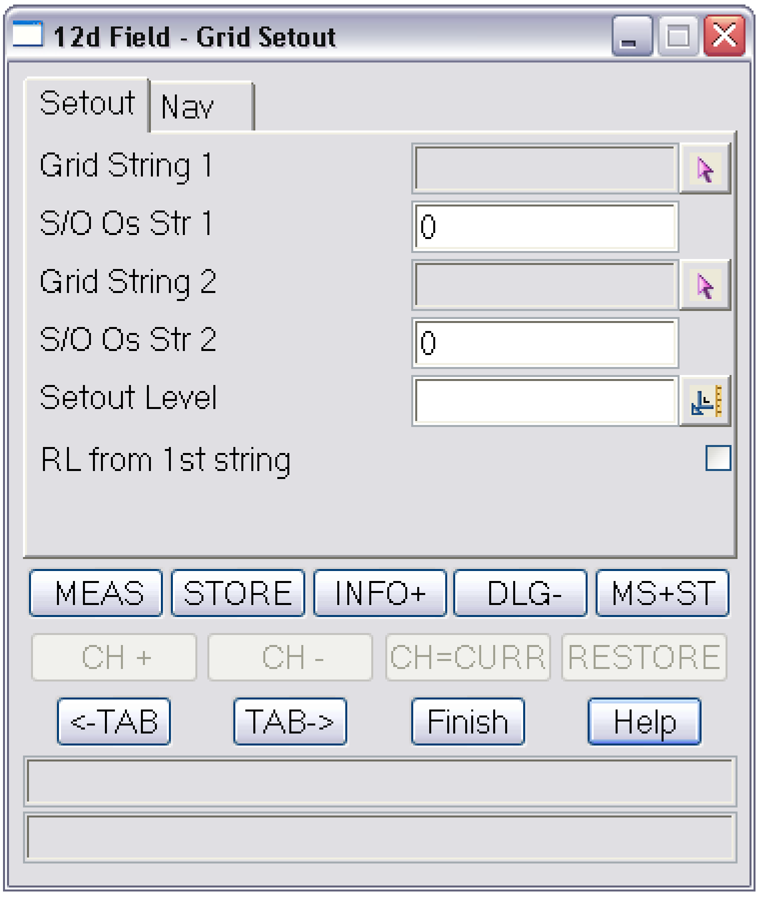
18.2.1.8.7 Grid Setout

This section of documentation is a work in progress and will be updated in subsequent releases.

The **12d Field - Grid Setout** panel creates a setout point at offsets from 2 strings.

Most typically used in buildings where points are dimensioned off setout grids

Selecting the **Grid Setout** option brings up the **12d Field - Grid Setout** panel.



Setout Tab

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Grid String 1 12dF new select box

The first string for grid setout

S/O Os Str 1 12dF double box

The setout offset from the first string

Grid String 2 12dF new select box

The second string for grid setout

S/O Os Str 2 12dF double box

The setout offset from the second string

Setout Level 12dF double box

The height to setout.

RL from 1st string tick box

If ticked then the level of the 1st string selected will be the setout level

Buttons

MEAS button

Start a measurement, the behaviour is determined by the TPS/GPS measurement setting set from the control bar.

INFO+ button

Display the user configurable information panel to view extra information not available on the standard dialog

DLG- button

Minimises the dialog so only the first 2 rows of buttons are shown.

MS+ST button

Start a measurement and store it on completion.

CH + button

increment the setout chainage by the value in the chainage increment field.

CH - button

Decrement the setout chainage by the value in the chainage increment field.

CH=CURR button

Set the setout chainage to the chainage of the last measured point.

RESTORE button

*Restore the setout chainage to the chainage prior to the "**CH=CURR**" button being pressed.*

<-TAB button

Go to the previous tab in the dialog.

TAB-> button

Go to the next tab in the dialog.

STOP button

Stop the measurement in process.

STORE button

Store the last measured point.

Nav P button

Load a saved navigation page configuration.

Sh M button

Start a measurement with a touch on the screen.

Sh M+S button

Start a measurement with a touch on the screen, store it on completion.

READ button

Load previously saved setout settings for reuse.

SAVE button

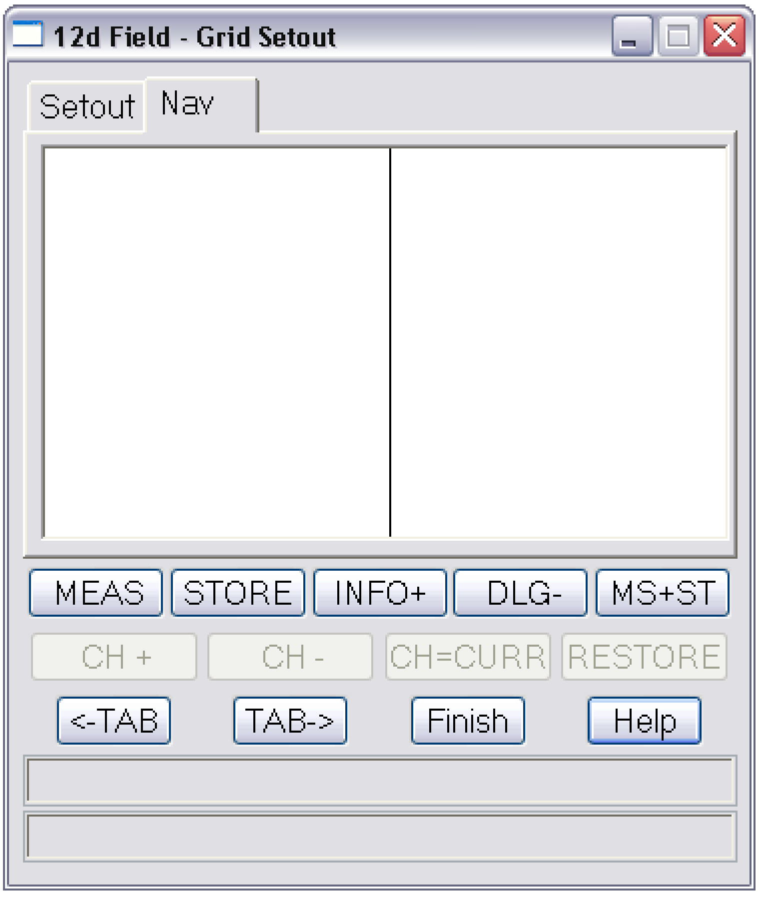
Save the current setout settings for reuse at a later date.

DLG+ button

Restore a minimised panel to it's full size.

INFO- button

Close the user configurable information panel.



Nav Tab

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Navigation Box draw box

The 12d Field navigation box augments setout by displaying user definable information rows plus a bulls-eye as a visual aid.

Continue to next section 18.2.1.8.8 Crown Setout or return to 18.2 12d Field .

18.2.1.8.8 Crown Setout

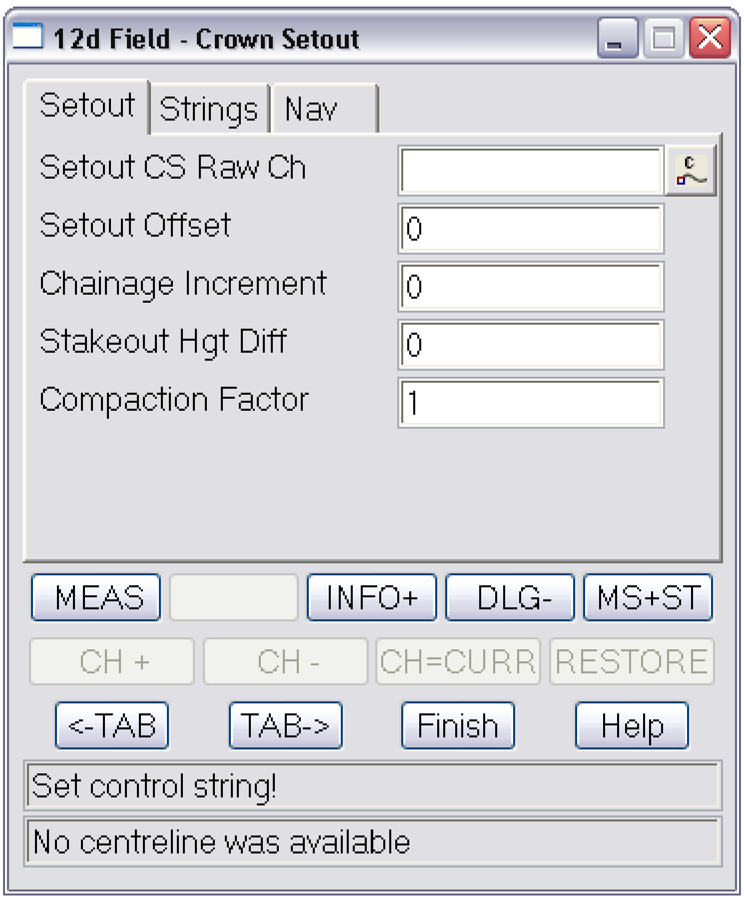
The **12d Field - Crown Setout** panel is effectively 2 **12d Field - Crossfall Setout** panels. It is typically used on rural roads when the user wants to generate design heights to both sides of the crowned road.

Crown setout works by dropping a point to a nominated control string, then the setout string and the 3 strings used to determine the crown are cut normal to the dropped point on the control string.

The user is able to set a shift to move the design surface up or down once the strings are cut.

Note the setout string and 1 of the level strings could be the same string or different, this dependant on the task being performed.

Selecting the **Crown Setout** option brings up the **12d Field - Crown Setout** panel.



Setout Tab

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Setout CS Raw Ch	12dF chainage box		

The raw, no equalities setout chainage on the control string, start chainage plus distance along string

Setout Offset	12dF double box		

The offset from the setout string to setout, +ve is to the right of the string, -ve left

Chainage Increment	12dF double box		

The value the setout chainage will be changed by when chainage increment/decrement is called.

Stakeout Hgt Diff 12dF double box

The height diff from the setout surface/string. +ve is above.

Compaction Factor 12dF double box

A compaction factor applied to the delta heights, e.g. if you know say asphalt will compact by 23% and you need to cover this enter the value as 1.23. (Note this widget is optional and only appears if activated in the Settings panel.)

Buttons

MEAS button

*Start a measurement, the behaviour is determined by the **TPS/GPS** measurement setting set from the control bar.*

INFO+ button

Display the user configurable information panel to view extra information not available on the standard dialog

DLG- button

Minimises the dialog so only the first 2 rows of buttons are shown.

MS+ST button

Start a measurement and store it on completion.

CH + button

Increment the setout chainage by the value in the chainage increment field.

CH - button

Decrement the setout chainage by the value in the chainage increment field.

CH=CURR button

Set the setout chainage to the chainage of the last measured point.

RESTORE button

*Restore the setout chainage to the chainage prior to the "**CH=CURR**" button being pressed.*

<-TAB button

Go to the previous tab in the dialog.

TAB-> button

Go to the next tab in the dialog.

STOP button

Stop the measurement in process.

STORE button

Store the last measured point.

Nav P button

Load a saved navigation page configuration.

Sh M button

Start a measurement with a touch on the screen.

Sh M+S button

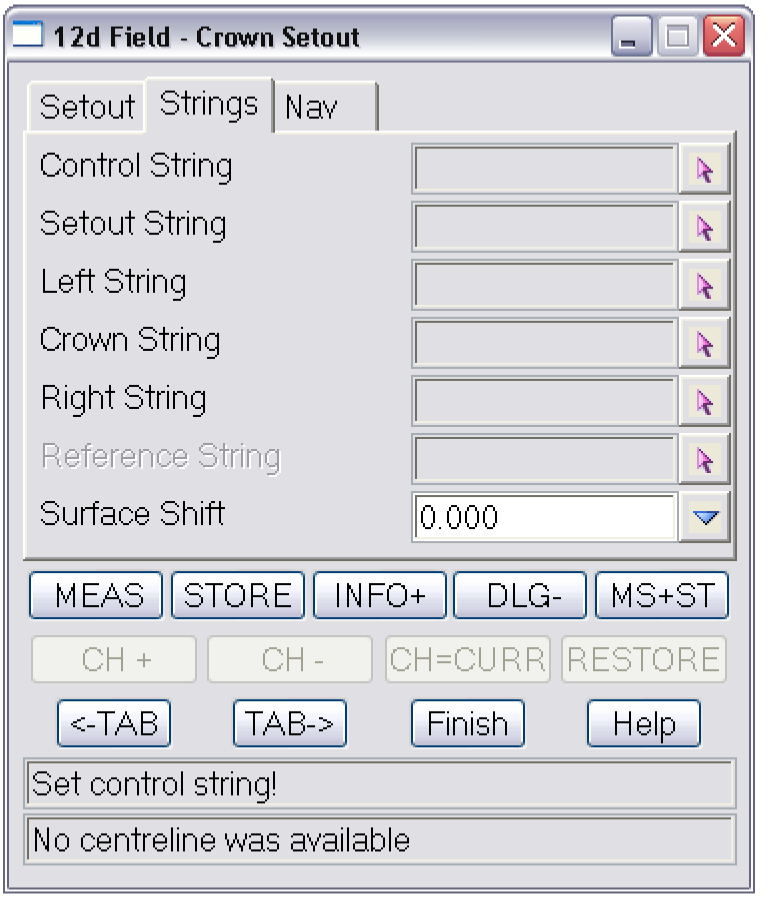
Start a measurement with a touch on the screen, store it on completion.

READ button

Load previously saved setout settings for reuse.

SAVE button

Save the current setout settings for reuse at a later date.



Strings Tab

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Control String 12dF new select box

Control string, the string to which the other strings are cut normal to for calculations.

Setout String 12dF new select box

Setout String, setout offset & heights are relative to this string at a point cut normal from the setout chainage on the control string/centreline

1st Level String 12dF new select box

Level string 1, a string cut for one of the design heights.

Crown String 12dF new select box

Crown string, the middle string for design heights in crown setout

2nd Level String 12dF new select box

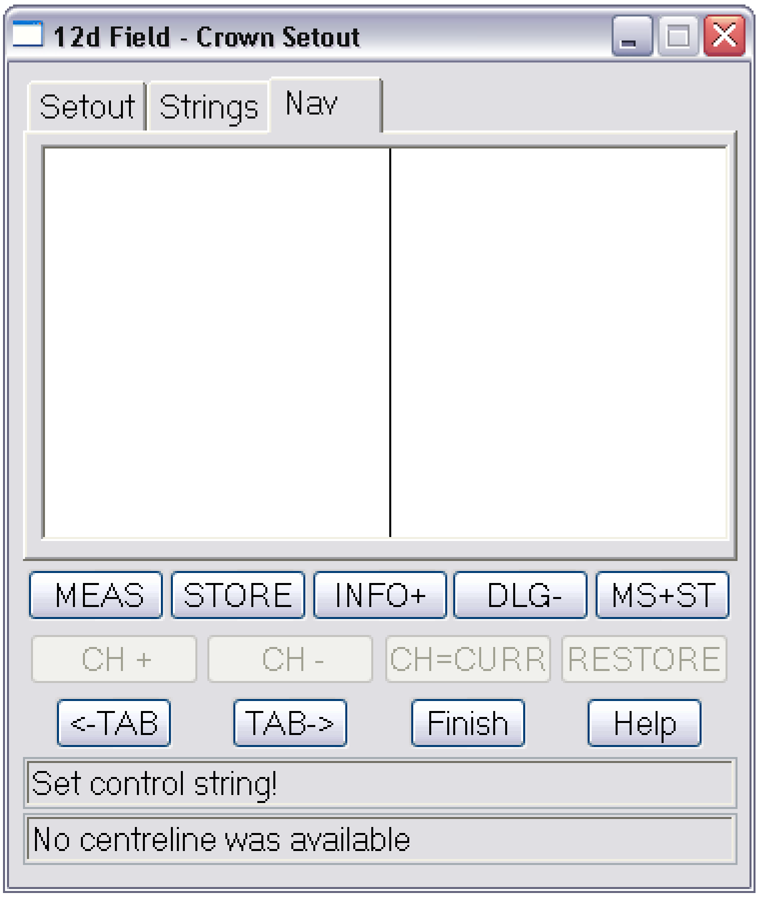
Level string 2, the string cut for a 2nd design height.

Reference String 12dF new select box

Reference string, a string to which the current point is dropped normally to for information only.

Surface Shift choice box 0.0000

*A vertical shift to be applied to the design level, +ve raises the level, can be manually entered or selected from the choice list, (defined in "**TDF_SURFACE_SHIFTS.4D**")*



Nav Tab

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Navigation Box draw box

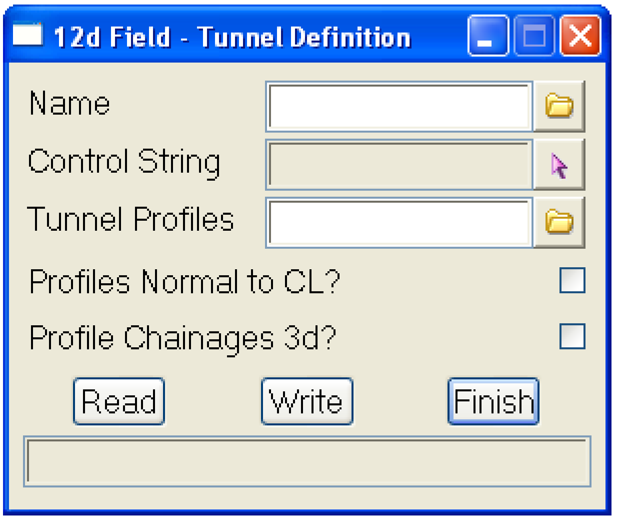
The 12d Field navigation box augments setout by displaying user definable information rows plus a bulls-eye as a visual aid.

Continue to next section 18.2.1.8.9 Tunnel Definition or return to 18.2 12d Field .

18.2.1.8.9 Tunnel Definition

The **Tunnel Definition** panel combines a centreline, the **PRO/PRA** tunnel definition files and information on how the profiles are applied to the centreline into a single file for use by the **Tunnel Setout** panel.

For more information on **PRO/PRA** tunnel definition files please see 20.12.10 Definition of the PRO and PRA definition files



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Name

*The name of the tunnel, this a file with the extension **12dF_TUN_DEF.***

Control string

*The centreline the tunnel definitions are applied to, the centreline must have valid vertical geometry for the chainage ranges in the **PRA** file.*

Tunnel Profiles

*The **PRO/PRA** files containing the tunnel definition. For more information on **PRO/PRA** tunnel definition files please see* 20.12.10 Definition of the PRO and PRA definition files

Profiles Normal to CL? tick box

if ticked the tunnel is calculated perpendicular/normal to the vertical alignment of the centreline resulting in a true 3d model.

if not ticked the tunnel is calculated vertical to the vertical alignment of the centreline, this means an effective loss of clearance on steeper grades.

Profile Chainages 3d? tick box

*if not ticked the chainages in the **PRA** file are taken as plan chainages.*

if ticked the chainages are interpreted as 3d, this is the plan/2d chainage of the 1st point where the horizontal and vertical geometry coincide plus the 3d length along the centreline from there.

Read button

Read in a 12dField tunnel definition file.

Write button

Write to a 12dField tunnel definition file.

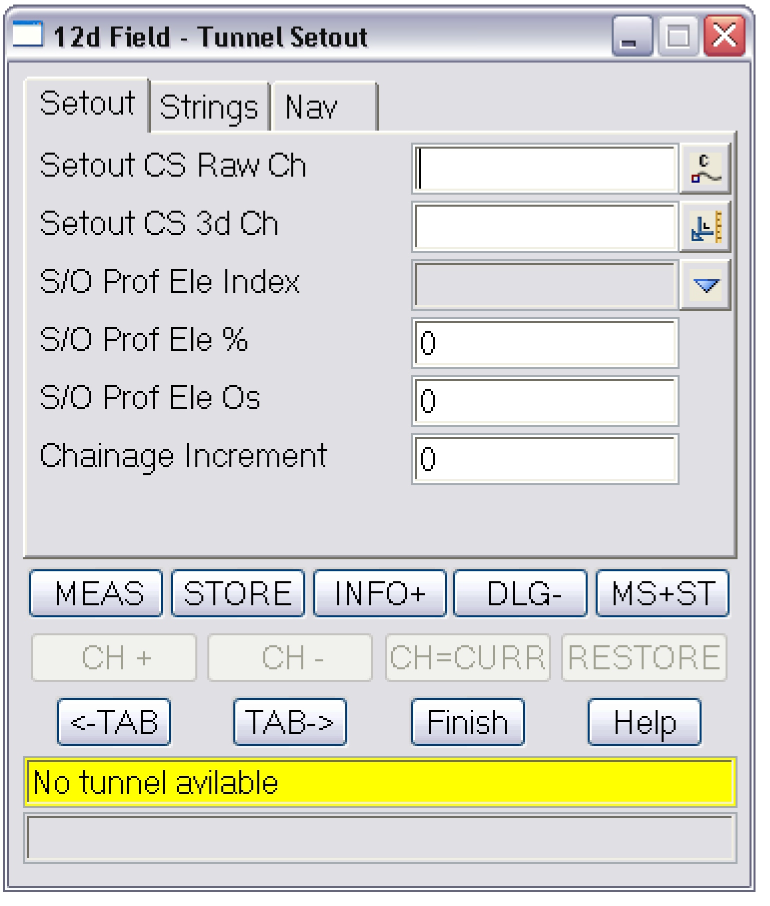
Finish

Exit the panel.

Continue to next section 18.2.1.8.10 Tunnel Setout or return to 18.2 12d Field .

18.2.1.8.10 Tunnel Setout

The **Tunnel Setout** option, (available in TPS only) allows the user to locate and setout points on a tunnel previously defined in the 18.2.1.8.9 Tunnel Definition panel.



Setout Tab

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Setout CS Raw Ch

The plan chainage of the point you wish to setout. If the tunnel chainages are defined as 3d this will be read only and be updated when a 3d chainage is entered.

Setout CS 3d Ch

The 3d chainage of the point you wish to setout. If the tunnel chainages are defined as 2d this will be read only and be updated when a 2d chainage is entered.

S/O Prof Ele Name

The name of the element in the profile to setout.

S/O Prof Ele %

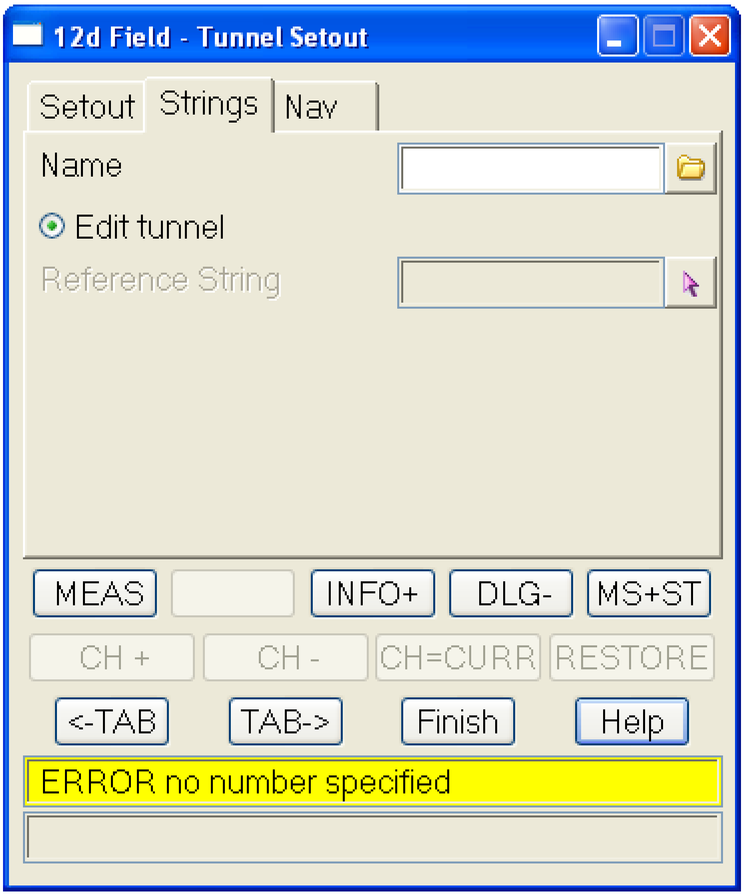
The percentage around the element to setout, 0% means the start of the element, 100% the end of the element.

S/O Prof Ele Os

The offset from the element to setout, +ve is to the right of the element.

Chainage Increment 12dF double box

The value the setout chainage will be changed by when chainage increment/decrement is called.



Strings Tab

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Name

Select the tunnel to setout.

Edit tunnel

If pressed will open up the 18.2.1.8.9 Tunnel Definition *panel with the currently defined tunnel, on closure of the* ***Tunnel Definition*** *panel the definition will be read in again.*

Reference String 12dF new select box

Reference string, a string to which the current point is dropped normally to for information only.

Attributes specific to tunnel setout for display on the information and navigation pages.

pu_tun_ele_name

Name of the tunnel element at the shot taken

pu_tun_ele_idx

Index, (zero based) of the tunnel element in the profile at the shot taken, 0 is the first element

pu_tun_ele_per

Percentage around the tunnel element at the shot taken.

pu_tun_ele_os

Offset from the tunnel element at the shot taken, +ve is to the right hand side.

so_tun_ele_name

Name of the tunnel element at the point setout

so_tun_ele_idx

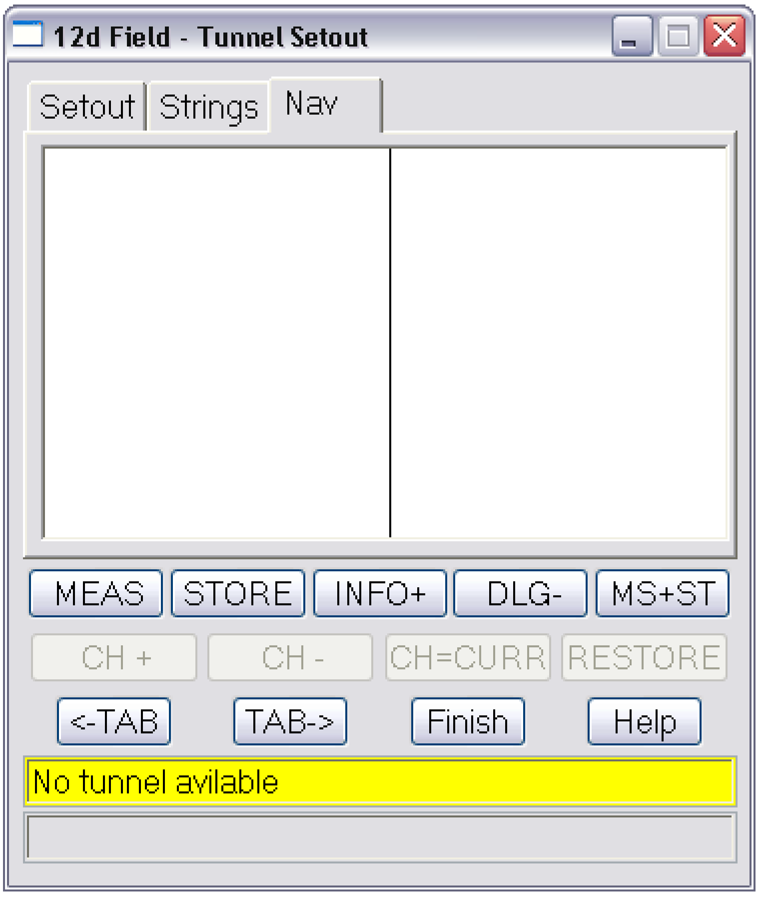
Index, (zero based) of the tunnel element in the profile at the point setout, 0 is the first element

so_tun_ele_per

Percentage around the tunnel element at the point setout.

so_tun_ele_os

Offset from the tunnel element at the point setout, +ve is to the right hand side.



Nav Tab

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Navigation Box	draw box		

The 12d Field navigation box augments setout by displaying user definable information rows plus a bulls-eye as a visual aid.

Continue to next section 18.2.1.8.11 Station Helmert or return to 18.2 12d Field .

18.2.1.8.11 Station Helmert

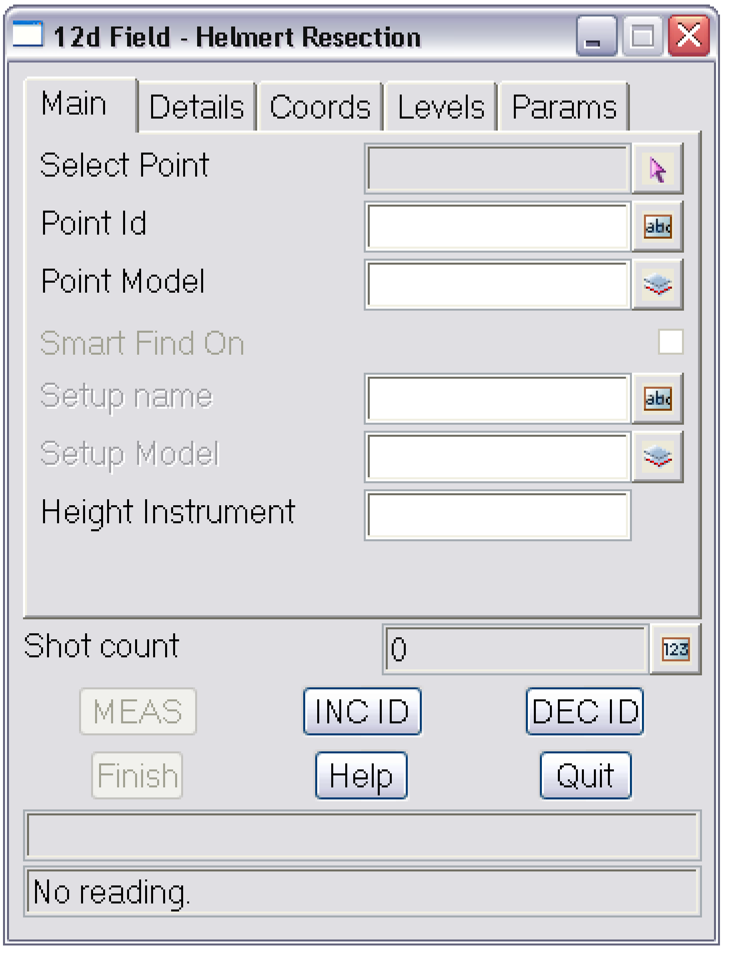
The **12d Field - Helmert Resection** panel can be used to establish a station setup by taking readings to up to 6 known points.

The horizontal position is obtained by a Helmert transformation; translation, rotation and uniform scaling of the readings.

The vertical position is obtained by meaning the z values of the readings, the z value is not weighted on distance measured.

Readings can be used for either horizontal position, vertical position or both.

Selecting **Station Helmert** displays the **12d Field - Helmert Resection** panel on the screen.



Main Tab

A point for use in the resection can either be picked from the screen or manually entered.

If picked from the screen the Id and Model boxes will be filled automatically.

If entered manually the Id must be unique in the point model.

Note in 12dField the Id always refers to the Vertex Id.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Select Point	string select box		

Select the point for the next shot to be used in the resection

Point Id	12dF input box		

The Id/name of the target point.

Point Model model box

The model containing the point for use in the resection.

Smart Find On tick box

Smart find can be used when you can clearly identify one point in a resection, e.g. in a tunnel you can see multiple targets but only identify the one closest to you.
Smart find can be used after you have measured to the identified point.

If ticked on then after a reading to a station smart find will look through the points model to find one that the reading matches, both horizontally and vertically.

The horizontal and vertical tolerances can be set on the parameters tab.

Use smart find to identify all stations measured to automatically.

The surveyor can optionally store the resection point to a model if need be.

Setup name input box

The Id of the setup to be stored.

Setup Model model box

The model to store the setup to.

Height Instrument 12dF double box

The height of the instrument above the setup point.

Buttons

MEAS button

Take a measurement to a station.

INC ID button

Increment the point ID and make this the setout point. E.g. BOLT30 -> BOLT31

DEC ID button

Decrement the point ID and make this the setout point. E.g. BOLT31 -> BOLT30

Shot Count

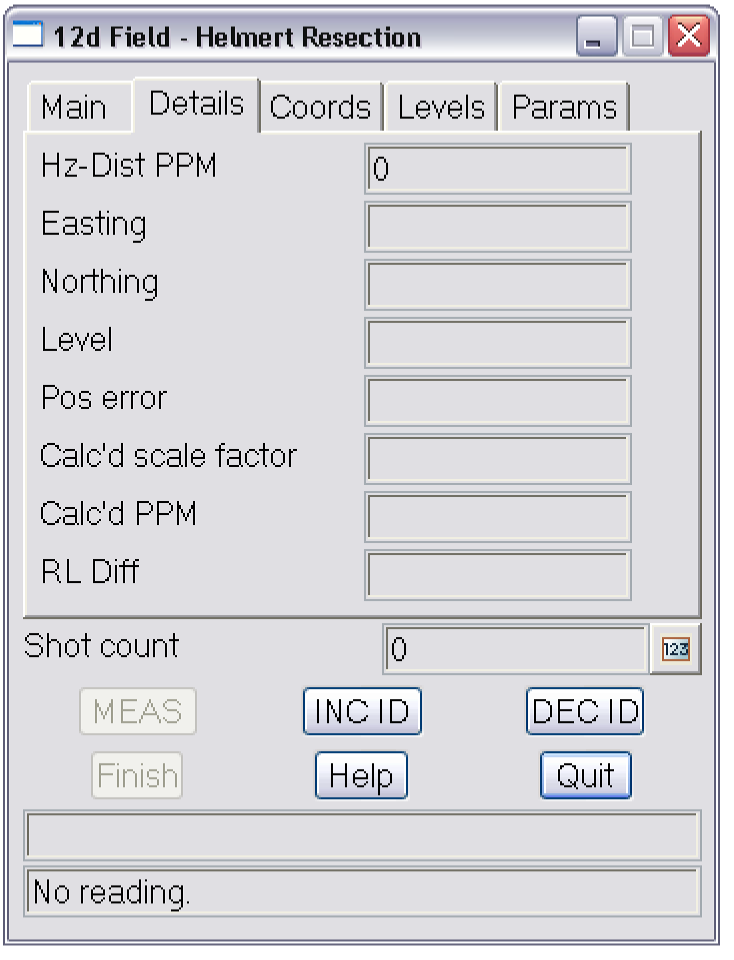
Number of shot used so far in the resection.

Finish button

Leave the panel and make this the new station setup, if any of the position tolerances nominated on the parameters tab are exceeded you will get a warning box allowing you to re-enter or continue.

Quit button

Quit the resection and restore the previous station setup.



Details Tab

The details tab displays the overall details of the current setup.

The level of detail is different for 2 and 3+ shots where redundancies in the helmert calcs are available. For 3+ shot the estimation error can be displayed.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Hz-Dist PPM	12dF long box		

The scale factor to be applied to measured horizontal distances.

Easting	double box		

The calculated easting of the station setup.

Northing	double box		

The calculated northing of the station setup.

Level	double box		

The calculated level of the station setup.

Pos error	double box		

The estimated horizontal error in the station setup.

Calc'd scale factor	double box		

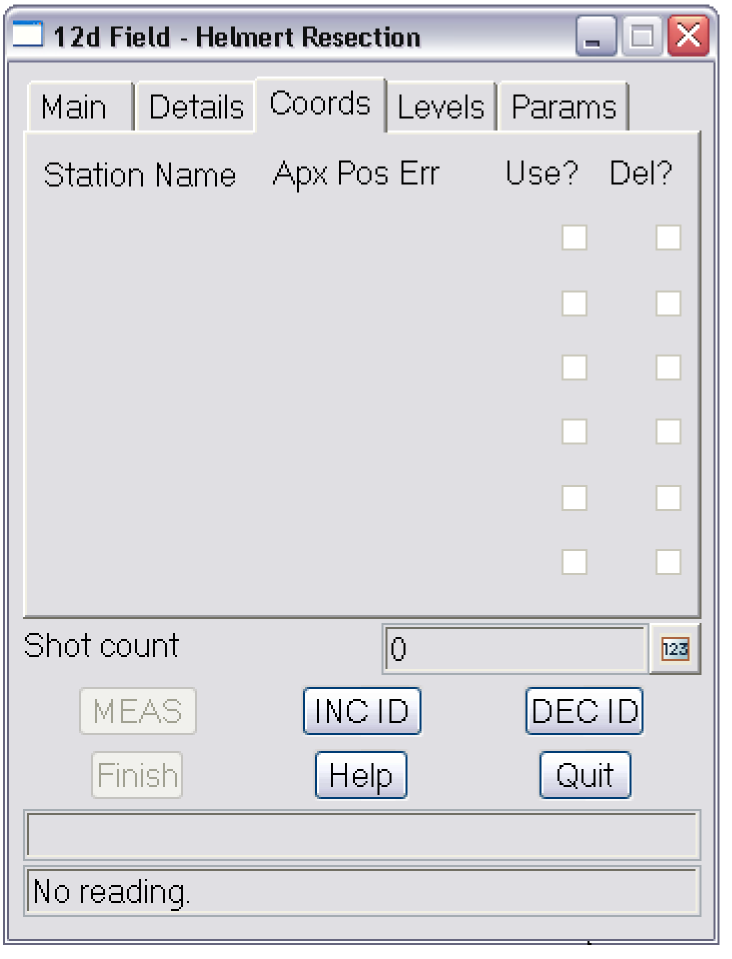
The scale factor applied to the measurements to get the best fit.

Calc'd PPM double box

The scale factor in ppm applied to the measurements to get the best fit.

RL Diff double box

The height difference range between the readings.



Coords Tab

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Station Name

The name of the station this reading refers to.

Apx Pos Err

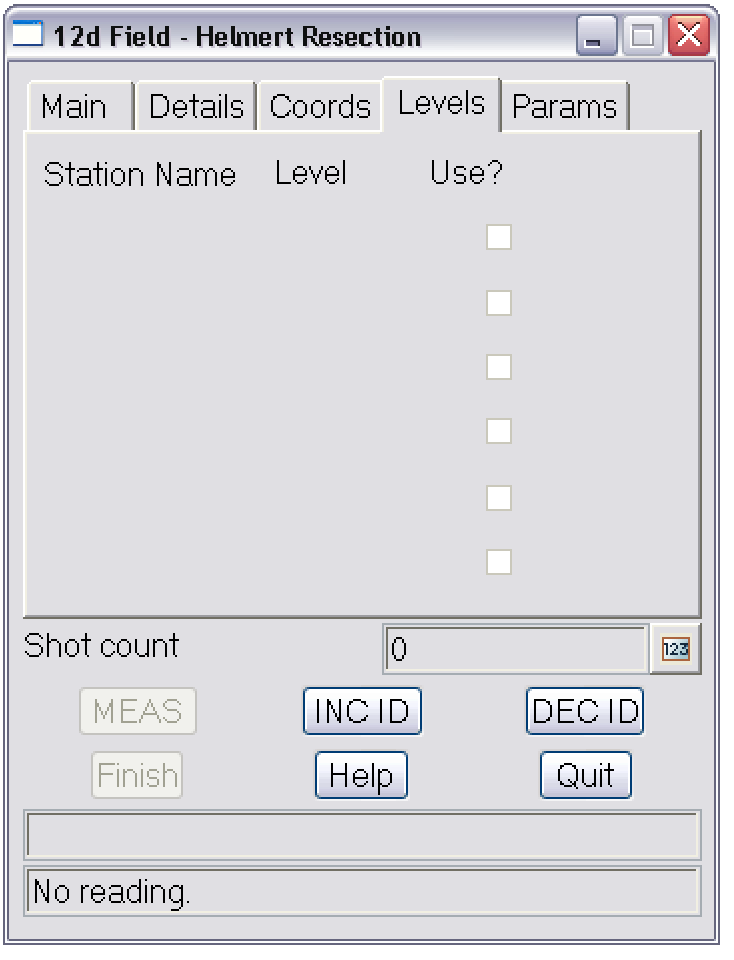
The approximate horizontal position error of this reading calculated in conjunction with all other readings in use. A high error for this reading in comparison to others might indicate a problem with this reading.

Use? tick box

if ticked this reading will be used in the horizontal calculation. Ticking or unticking the box will trigger the recalculation of the horizontal position.

Del? tick box

If ticked a confirmation box will be displayed and the reading removed from the resection. Removal will trigger the recalculation of the horizontal position.



Levels Tab

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Station Name

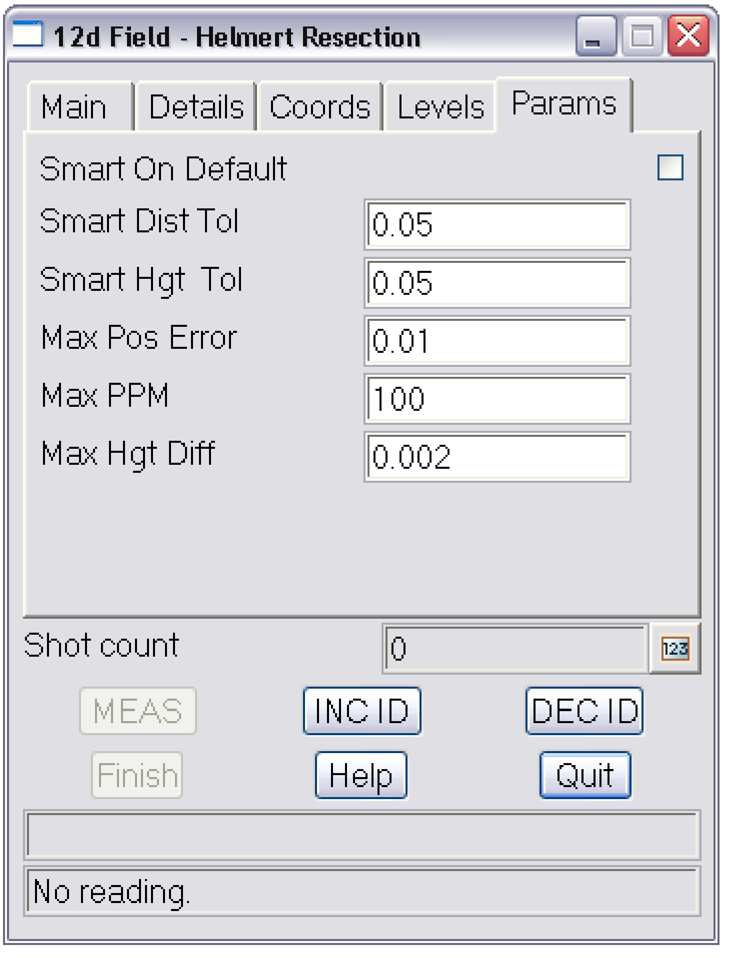
The name of the station this reading refers to.

Level

The level of the station calculated by this reading. A large difference for this reading in comparison to others might indicate a problem with this reading.

Use? tick box

If ticked this reading will be used in the vertical calculation. Ticking or unticking the box will trigger the recalculation of the vertical position.



Params Tab

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Smart On Default tick box

The smart find feature is ticked on by default

Smart Dist Tol 12dF double box

The distance tolerance for a point to be used in the smart find

Smart Hgt Tol 12dF double box

The level tolerance for a point to be used in the smart find

Max Pos Error 12dF double box

The maximum position error allowed, a warning will be shown if exceeded.

Max PPM 12dF double box

The maximum ppm's allowed, a warning will be shown if exceeded.

Max Hgt Diff 12dF double box

The maximum height variation allowed, a warning will be shown if exceeded.

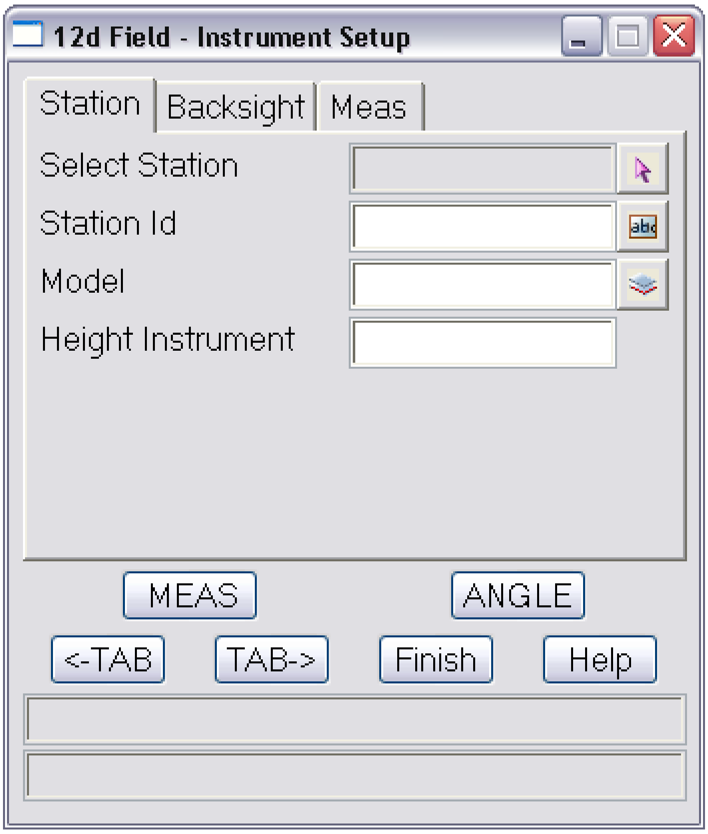
Continue to next section 18.2.1.8.12 Station Standard or return to 18.2 12d Field .

18.2.1.8.12 Station Standard

The **12d Field - Instrument Setup** panel is used to establish a station setup over a known point.

Orientation is established by measuring/sighting a backsight station. It is not necessary to set an orientation on the instrument, 12dField handles the angle difference internally.

Selecting the **Station Standard** option brings up the **12d Field - Instrument Setup** panel.



Station Tab

The point for use in the setup can either be picked from the screen or manually entered.

If picked from the screen the Id and Model boxes will be filled automatically.

If entered manually the Id must be unique in the point model.

Note in 12dField the Id always refers to the Vertex Id.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Select Station	string select box		

Select the control station you are setting up over.

Station Id	12dF input box		

The Id/name of the instrument setup.

Model	model box		

The model containing the instrument station.

Height Instrument	12dF double box		

The height of the instrument above the setup point.

Buttons

MEAS button

Start a full measurement, distance and angles.

ANGLE button

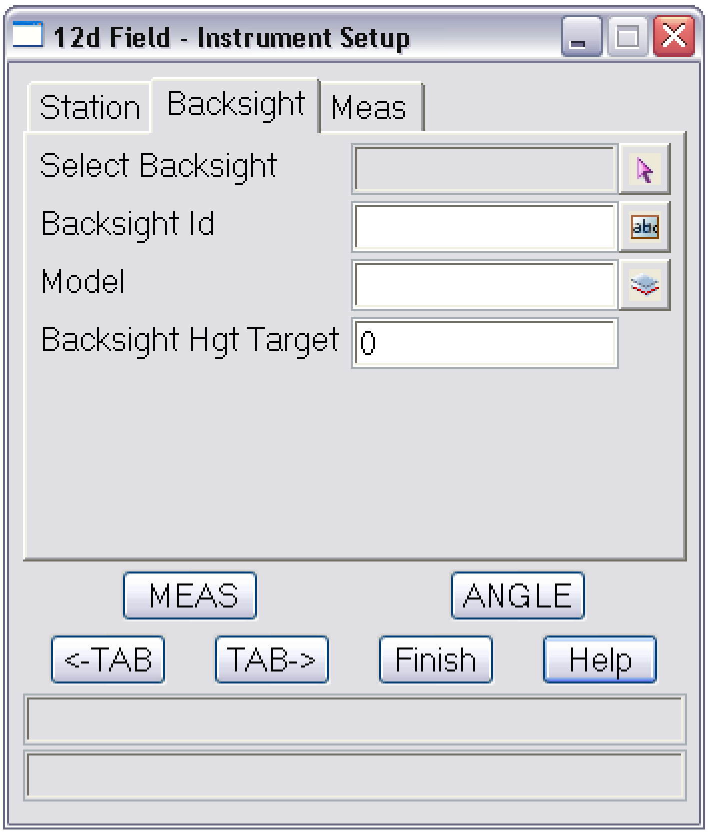
Start a horizontal/vertical angular measurement, no distance.

<-TAB button

Go to the previous tab in the dialog.

TAB-> button

Go to the next tab in the dialog.



Backsight Tab

The backsight for use in the setup can either be picked from the screen or manually entered.

If picked from the screen the Id and Model boxes will be filled automatically.

If entered manually the Id must be unique in the point model.

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Select Backsight string select box

Select the point to be used for the backsight.

Backsight Id 12dF input box

The Id/name of the backsight point.

Backsight Hgt Target 12dF double box

The height of the backsight target above the backsight point.

Meas Tab

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

BkSt Diff Horz Dist 12dF double box

Measured difference in horizontal distance to the backsight point.

BkSt Diff Meas East 12dF double box

Measured difference in easting to the backsight point.

BkSt Diff Meas North 12dF double box

Measured difference in northing to the backsight point.

BkSt Diff Meas Hgt 12dF double box

Measured difference in height to the backsight point.

Bearing Swing 12dF angle box

A bearing swing applied to the instrument reading to get the correct grid bearing.

Max Diff Horz Dist 12dF double box

Difference in horizontal distance that brings up a warning message.

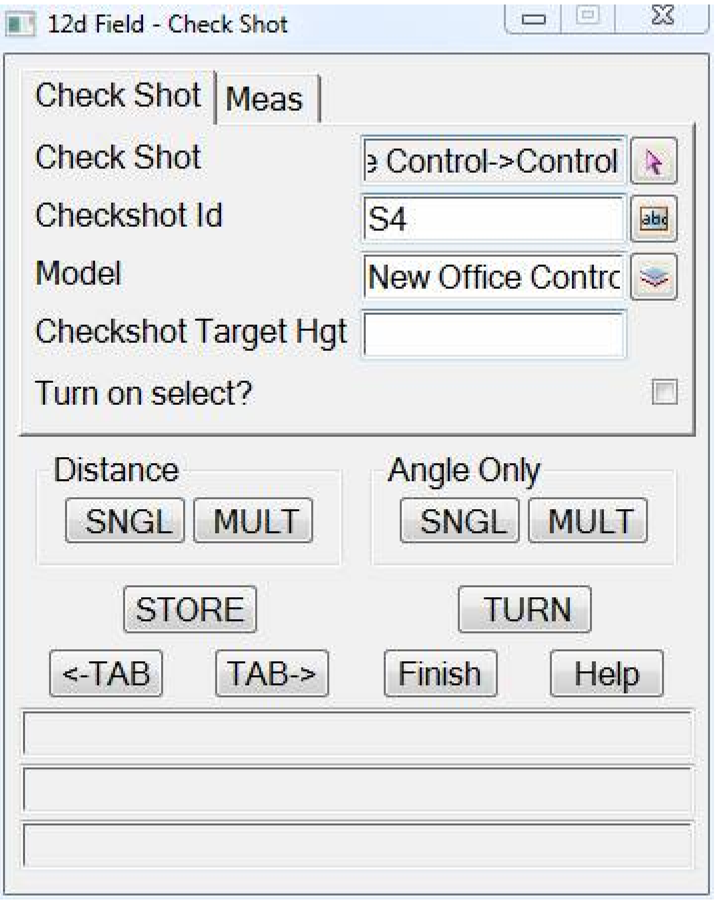
Max Diff Meas Hgt 12dF double box

Difference in height that brings up a warning message.

Continue to next section 18.2.1.8.13 Check Shot or return to 18.2 12d Field .

18.2.1.8.13 Check Shot

The **12d Field - Check Shot** panel allows a measurement to a known point to check the validity of the current setup. The measurement can be a full measurement or angle only.



Check Shot Tab

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Check Shot	string select box		

select a point for the check shot measurement

Checkshot Id

the vertex id of the selected point, this can be populated from the select or manually entered.

Model	model box		

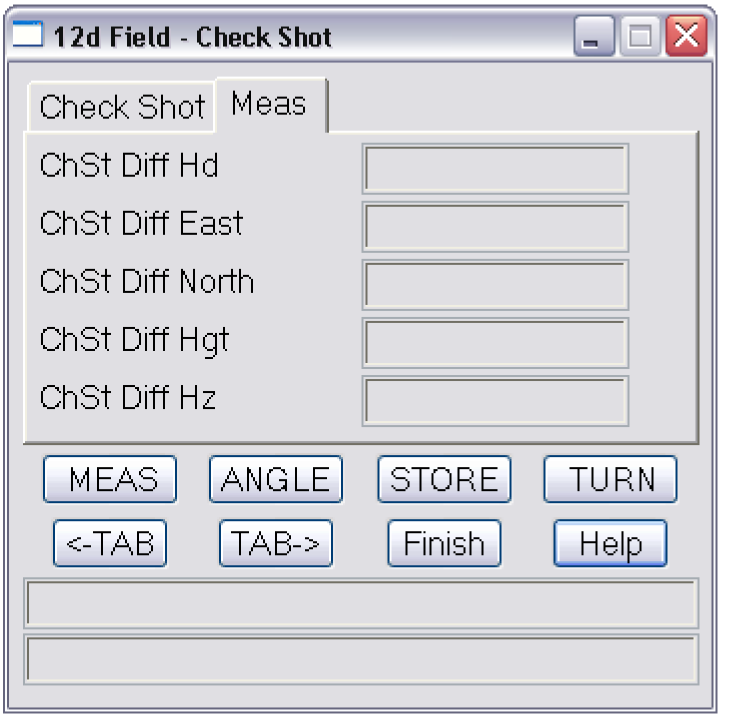
the model containing the checkshot point.

Checkshot Target Hgt

the height of target for the checkshot.

Turn on select?	tick box		

if ticked on selecting a checkshot point the instrument will rotate automatically to that point.



Meas Tab

Once a measurement is completed the focus will change to the **Meas** tab.

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

ChSt Diff Hd

the difference in horizontal distance to the checkshot point, actual - observed

ChSt Diff East

the difference in eastings to the checkshot point, actual - observed

ChSt Diff North

the difference in northings to the checkshot point, actual - observed

ChSt Diff Hgt

the difference in level to the checkshot point, actual - observed

ChSt Diff Hz

the difference in horizontal angle to the checkshot point, actual - observed

Buttons

Distance

SNGL button

Take a single distance measurement to the checkshot dependent on the current TPS measurement settings

MULT button

Take a multiface distance measurement to the checkshot dependent on the current TPS measurement settings

Angle Only

SNGL button

Take a single angular measurement to the checkshot dependent on the current TPS measurement settings

MULT button

Take a multiface angular measurement to the checkshot dependent on the current TPS measurement settings

STORE button

Write the measurement to the checkshot to the current field file (.FLD), this does not store the reading to a model.

TURN button

rotate the instrument to the selected checkshot point.

<-TAB button

Go to the previous tab in the dialog.

TAB-> button

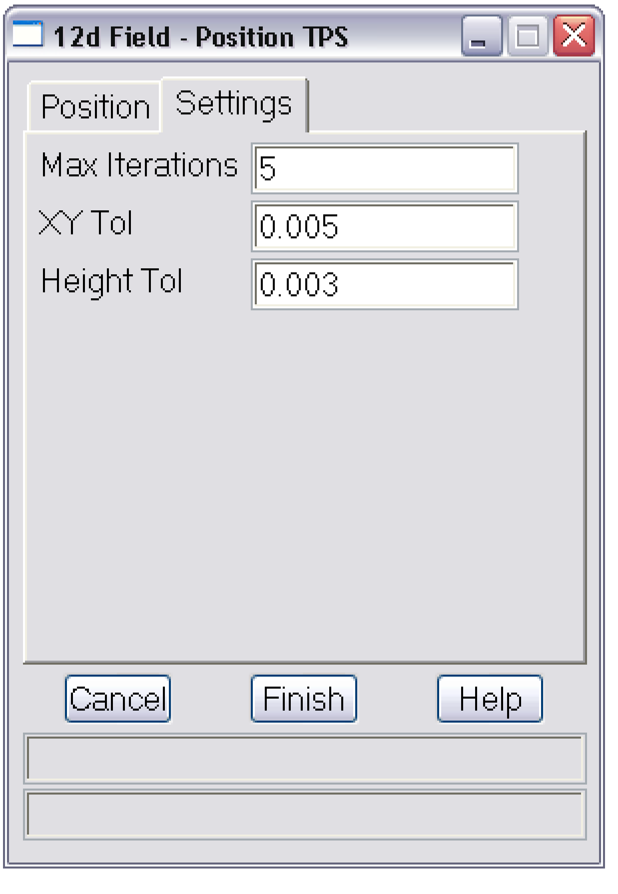
Go to the next tab in the dialog.

Continue to next section 18.2.1.8.14 Position TPS or return to 18.2 12d Field .

18.2.1.8.14 Position TPS

This section of documentation is a work in progress and will be updated in subsequent releases.

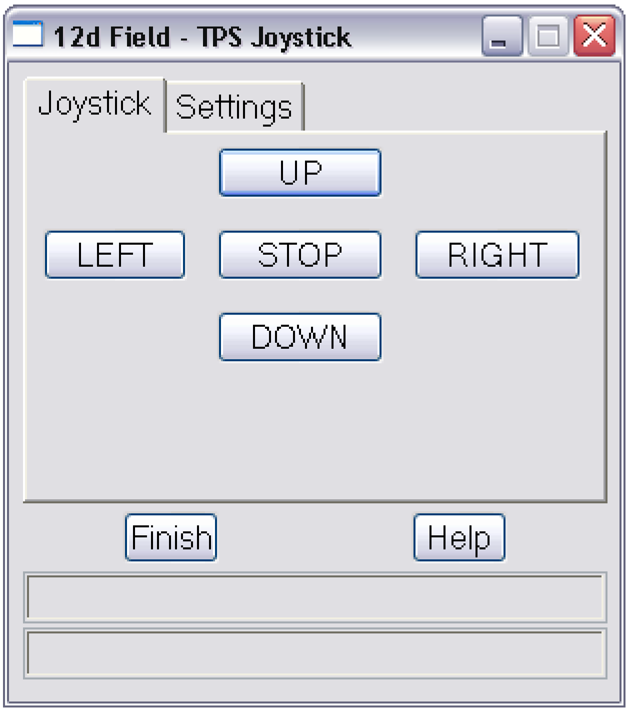

12d Field - Position TPS
Position
Settings
Turn
Hz
Hz 3d
Last
Iterate
XY
Ch
Hgt S/O
Hgt P/U
View Turn
XY
XYZ
Meas
Rep Meas
Cancel
Finish
Help

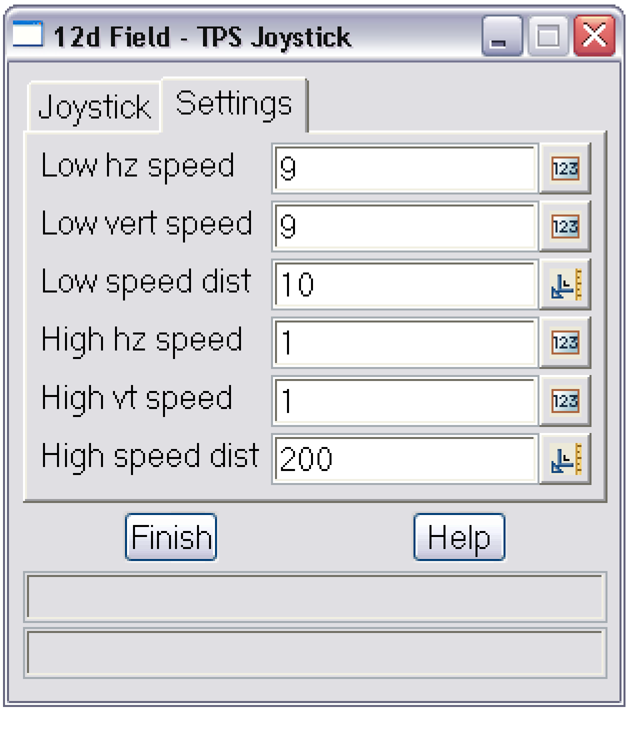

12d Field - Position TPS
Position
Settings
Max Iterations
5
XY Tol
0.005
Height Tol
0.003
Cancel
Finish
Help

Continue to next section 18.2.1.8.15 Joystick TPS or return to 18.2 12d Field .

18.2.1.8.15 Joystick TPS

This section of documentation is a work in progress and will be updated in subsequent releases.

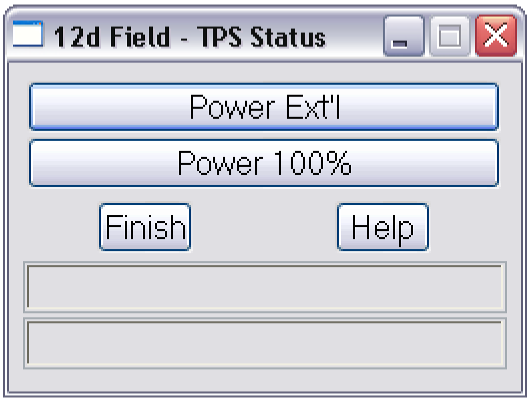




Continue to next section 18.2.1.8.16 Status TPS or return to 18.2 12d Field .

18.2.1.8.16 Status TPS

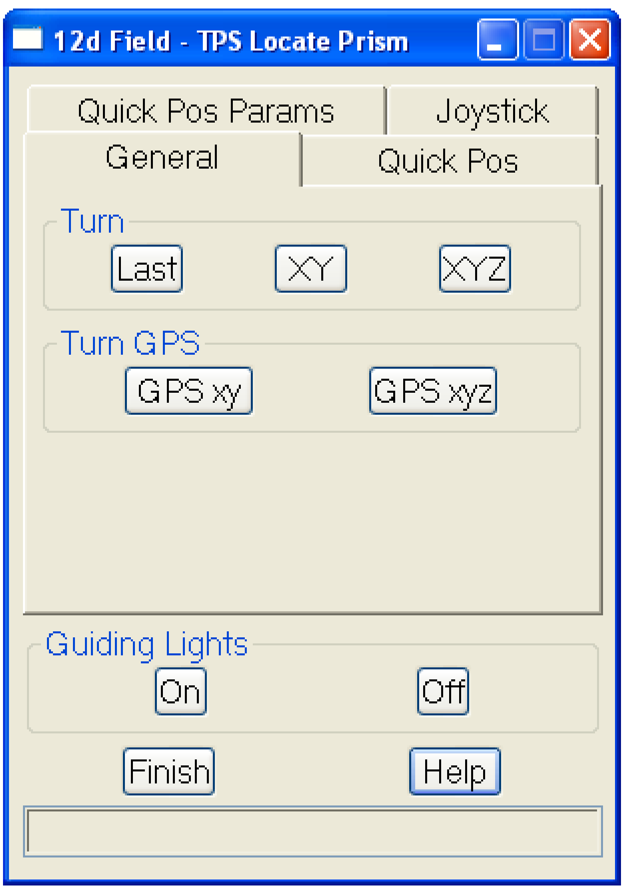
This section of documentation is a work in progress and will be updated in subsequent releases.



Continue to next section 18.2.1.8.17 Locate Prism TPS or return to 18.2 12d Field .

18.2.1.8.17 Locate Prism TPS

The **12d Field - TPS Locate Prism** panel is an aid to remotely position the TPS to the current prism location.



Common fields and buttons

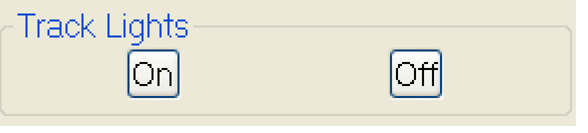
The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

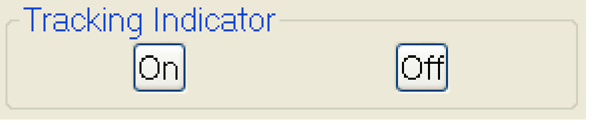
Guiding lights

The guiding lights button group is available if the TPS in use has guide lights. The naming of the guiding lights group depends on the primary instrument.

Trimble SPSx30



Topcon 9000



Leica 1200

On

Activate the instruments guiding lights.

On

Deactivate the instruments guiding lights.

General Tab

Turn

Last

The instrument will turn to the last position a measurement was taken at.

XY

A screen select will be activated and the instrument will turn horizontally only to the selected position.

XYZ

A screen select will be activated and the instrument will turn horizontally and vertically to the selected position.

Turn GPS

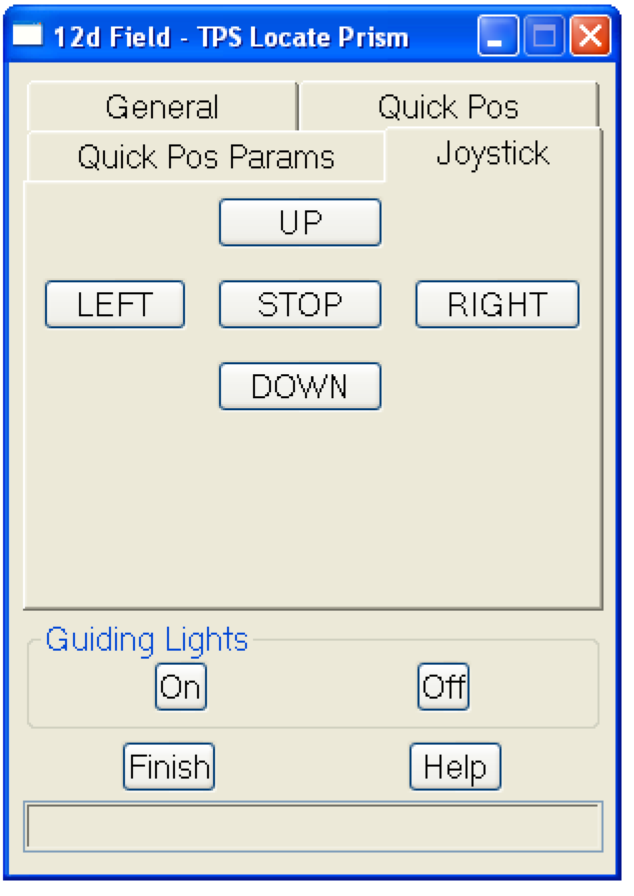
Note these buttons are only available if a GPS instrument is selected as the secondary instrument.

GPS xy

The instrument will turn horizontally only to the current GPS position.

GPS xyz

The instrument will turn horizontally and vertically to the current GPS position.



Joystick Tab

LEFT

Rotate the instrument to the left as viewed from the prism back to the instrument, ***LEFT*** *can be pressed twice more to increase the speed of rotation.*

RIGHT

Rotate the instrument to the right as viewed from the prism back to the instrument, ***RIGHT*** *can be pressed twice more to increase the speed of rotation.*

UP

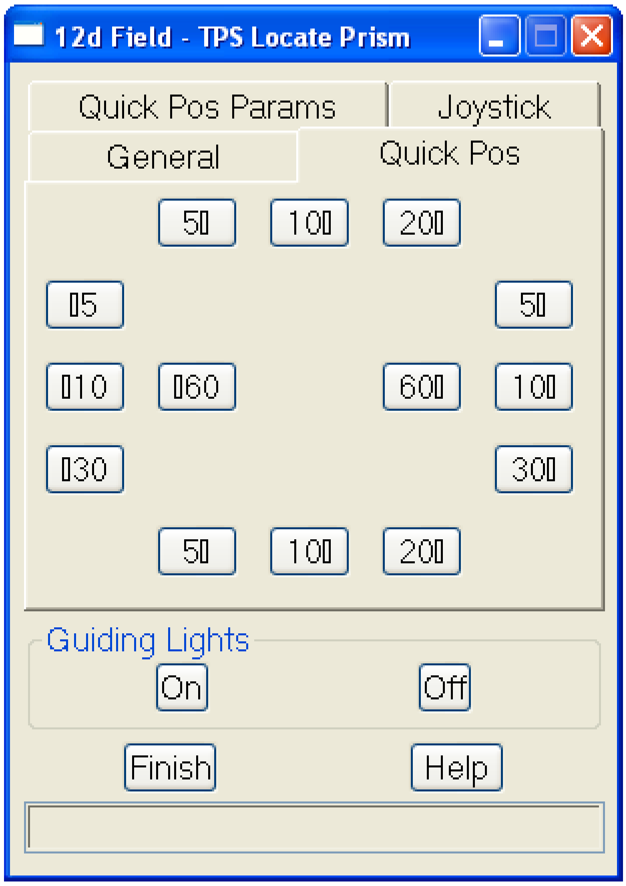
Rotate the Up, ***UP*** *can be pressed twice more to increase the speed of rotation.*

DOWN

Rotate the Down, ***DOWN*** *can be pressed twice more to increase the speed of rotation.*

STOP

Stop the instrument rotating.

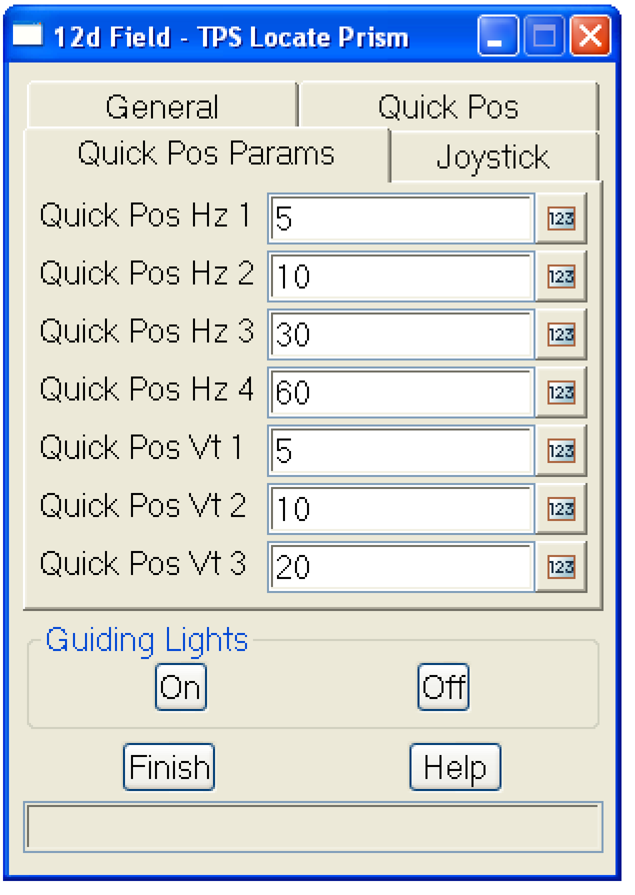


Quick Pos Tab

The **Quick Pos** tab allows the user to manually rotate the instrument in the horizontal and vertical directions by preset amounts.This is sometimes easier to use to orientate the instrument correctly than the joystick controls. There are 4 user definable rotations in the horizontal directions and 3 in the vertical.

The buttons show the direction of rotation and the amount of rotation in degrees, the amount of the rotation is defined on the Quick Pos Params Tab .

When a rotation button is pressed the instrument will rotate by a specified amount and direction, all buttons are disabled until the rotation has completed



Quick Pos Params Tab

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

The **Quick Pos Params** define the rotation available in the Quick Pos Tab .

Quick Pos Hz 1

*the amount of rotation in degrees for the top button in the horizontal groups for **Quick Pos***

Quick Pos Hz 2

*the amount of rotation in degrees for the outer middle button in the horizontal groups for **Quick Pos***

Quick Pos Hz 3

*the amount of rotation in degrees for the lower button in the horizontal groups for **Quick Pos***

Quick Pos Hz 4

*the amount of rotation in degrees for the inner middle button in the horizontal groups for **Quick Pos***

Quick Pos Vt 1

*the amount of rotation in degrees for the left button in the vertical groups for **Quick Pos***

Quick Pos Vt 2

*the amount of rotation in degrees for the middle button in the vertical groups for **Quick Pos***

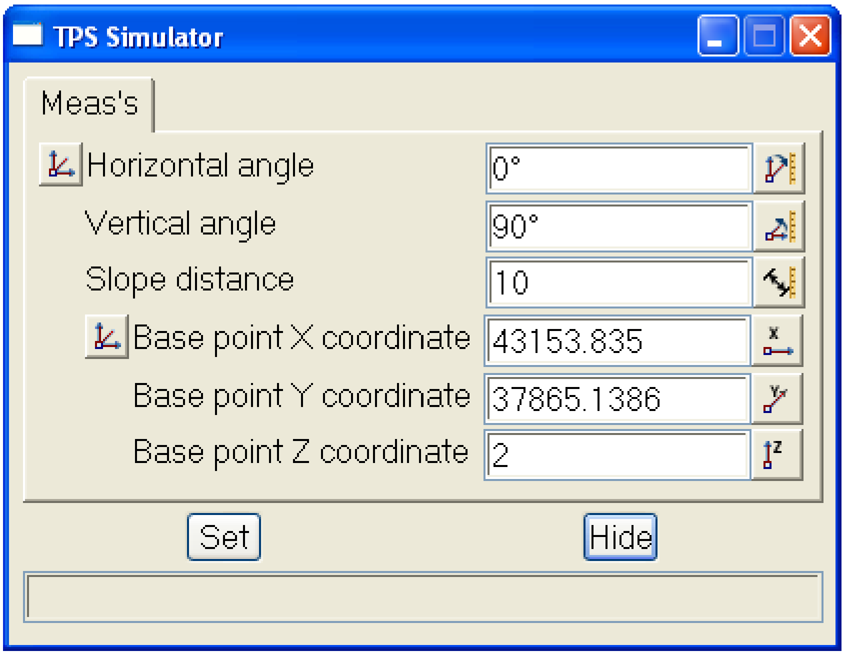
Quick Pos Vt 3

*the amount of rotation in degrees for the right button in the vertical groups for **Quick Pos***

Continue to next section 18.2.1.8.18 Simulator Settings or return to 18.2 12d Field .

18.2.1.8.18 Simulator Settings

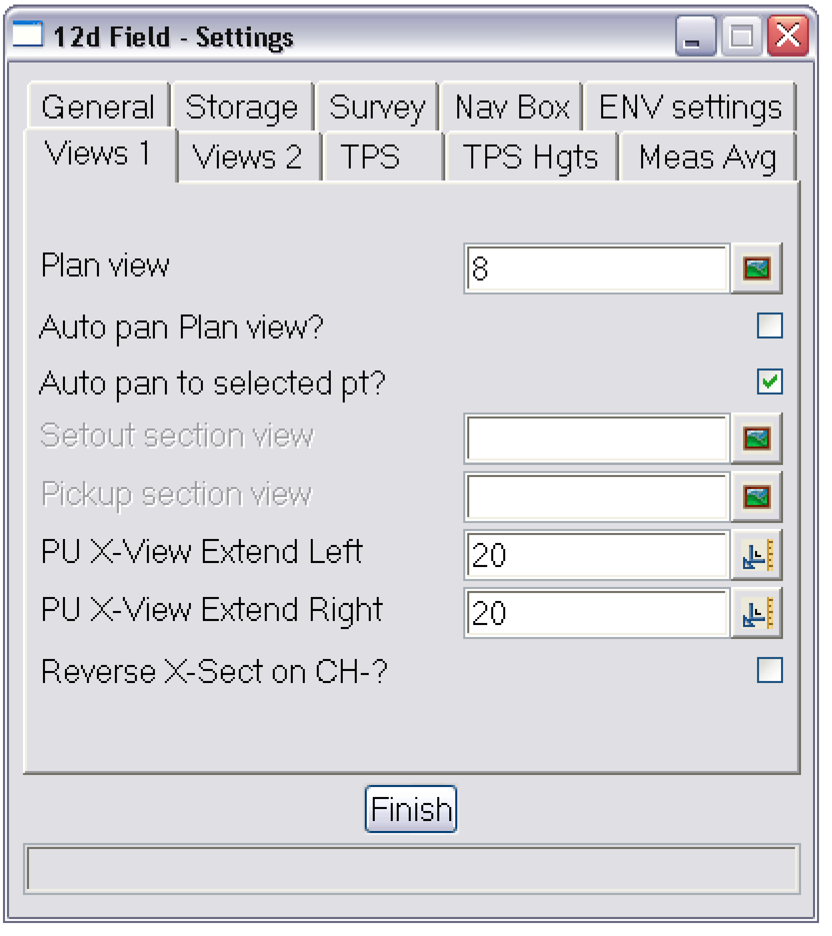
This section of documentation is a work in progress and will be updated in subsequent releases.

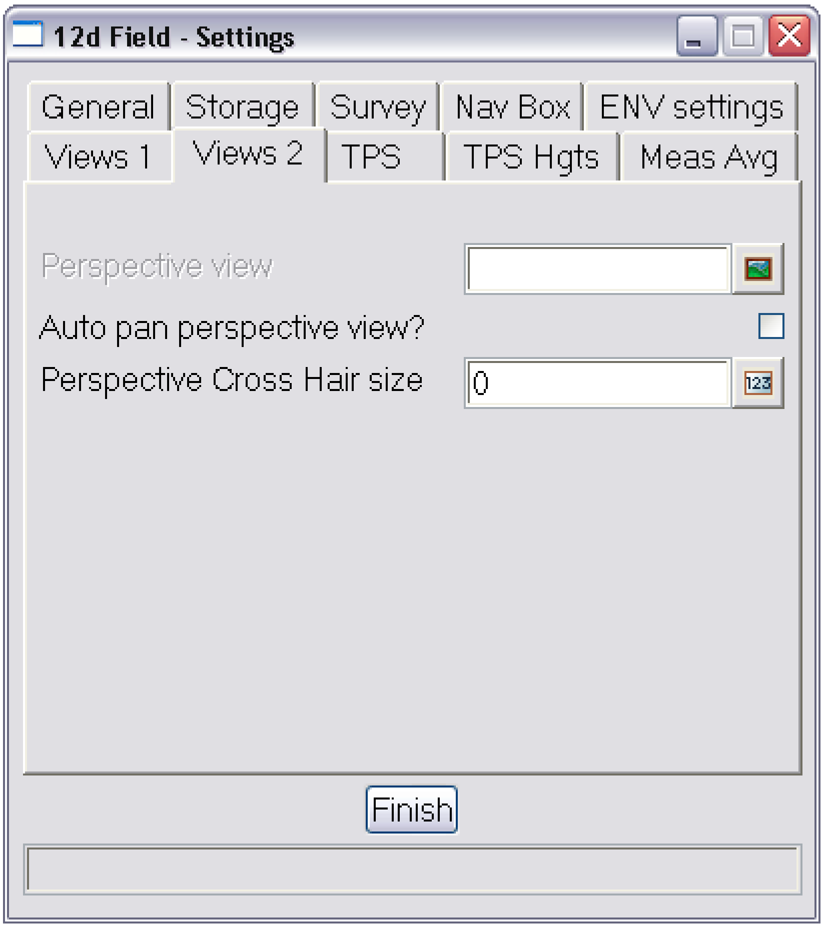


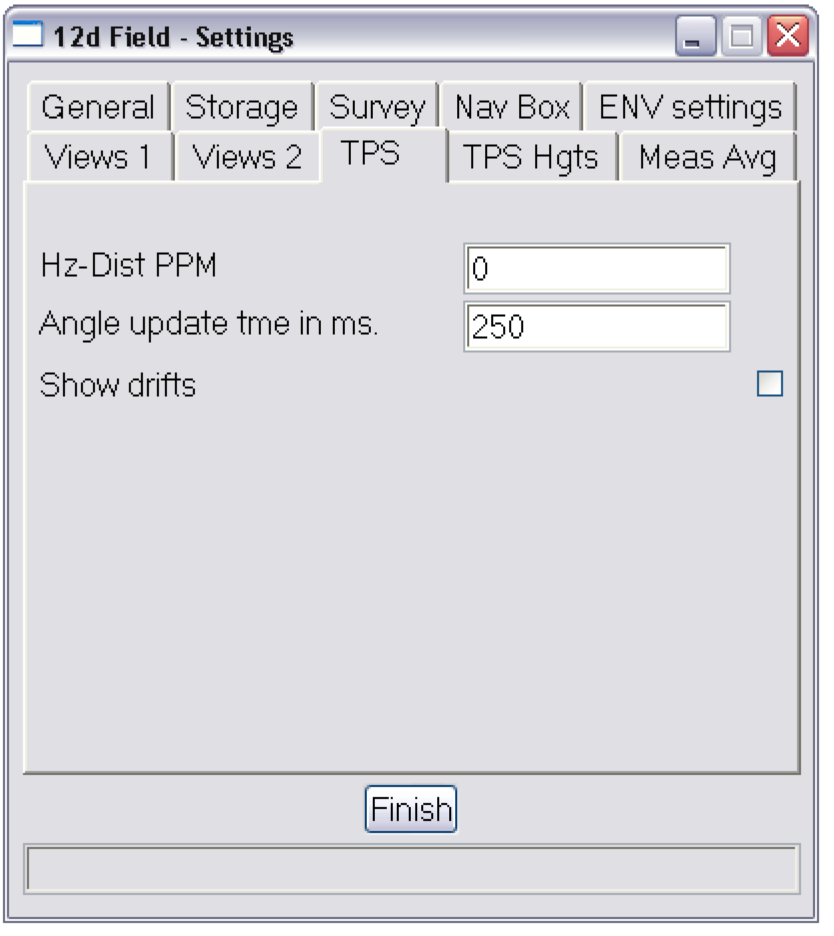
Continue to next section 18.2.1.8.19 Settings or return to 18.2 12d Field .

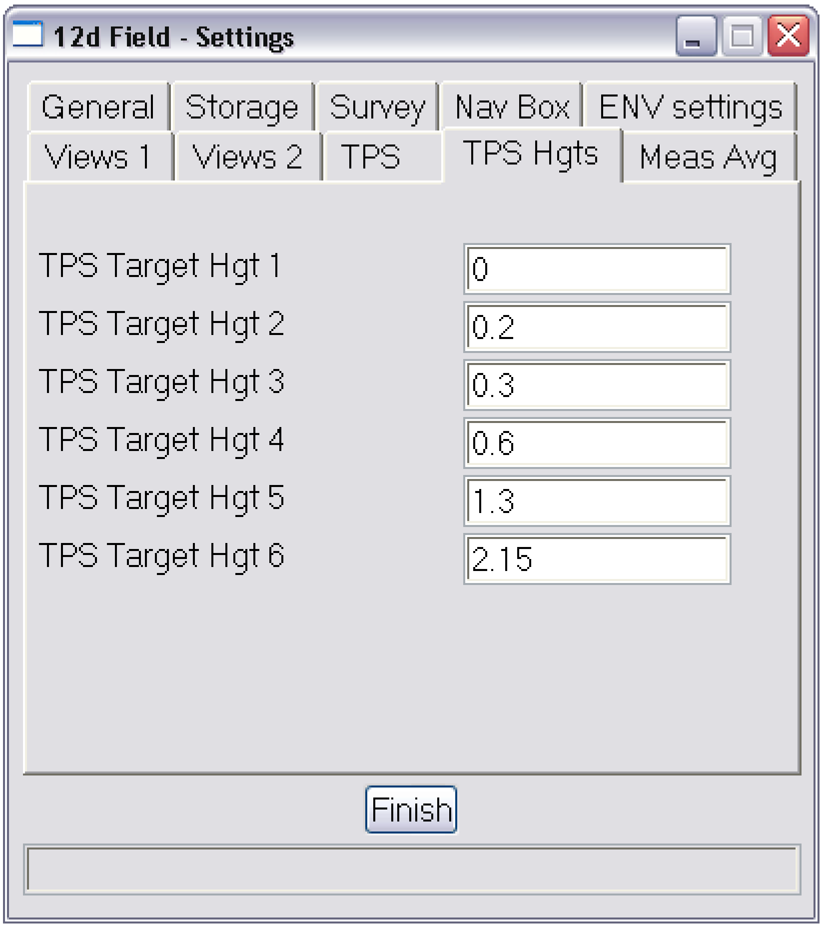
18.2.1.8.19 Settings

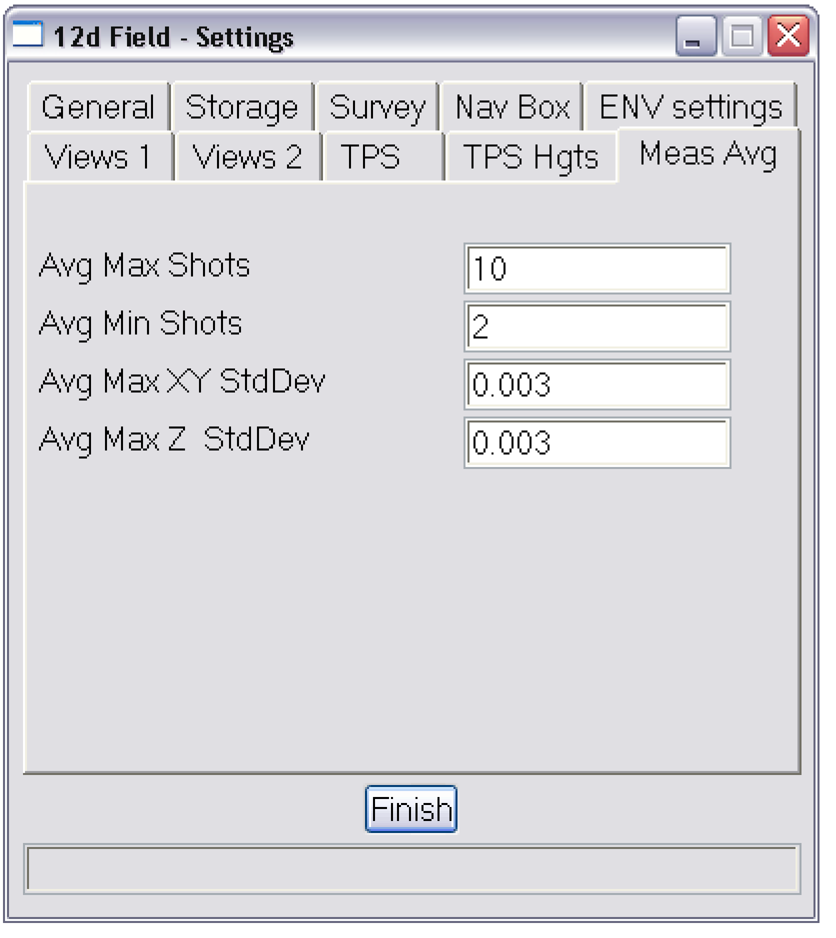
This section of documentation is a work in progress and will be updated in subsequent releases.

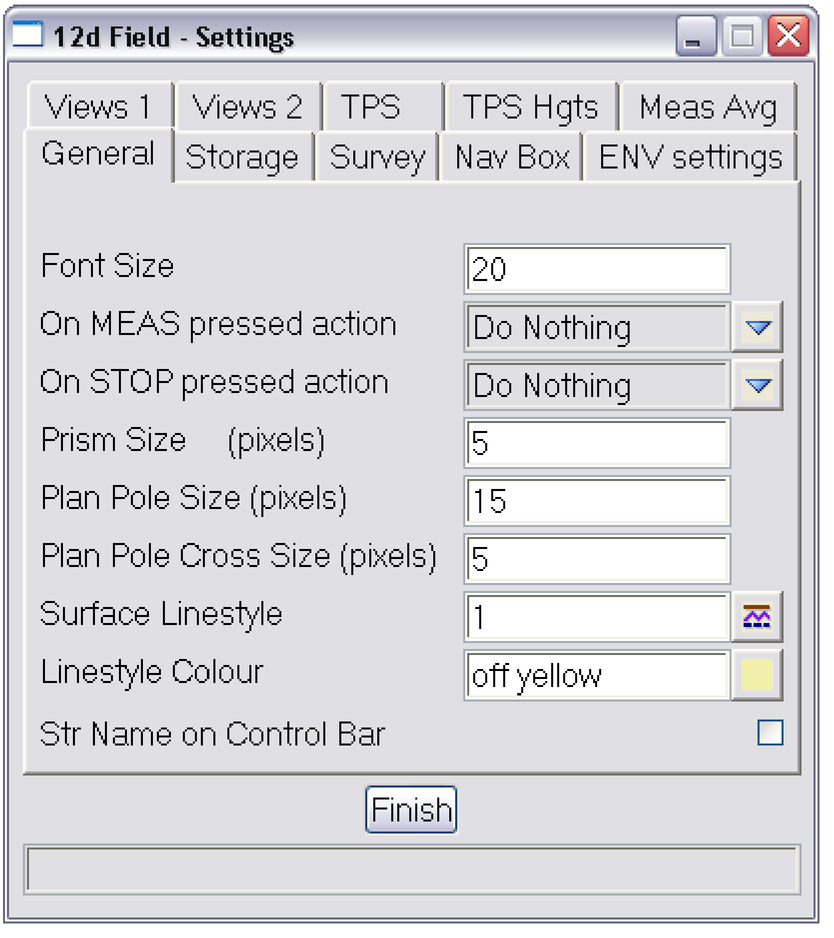

12d Field - Settings
General
Storage
Survey
Nav Box
ENV settings
Views 1
Views 2
TPS
TPS Hgts
Meas Avg
Plan view
8
Auto pan Plan view?
Auto pan to selected pt?
Setout section view
Pickup section view
PU X-View Extend Left
20
PU X-View Extend Right
20
Reverse X-Sect on CH-?
Finish

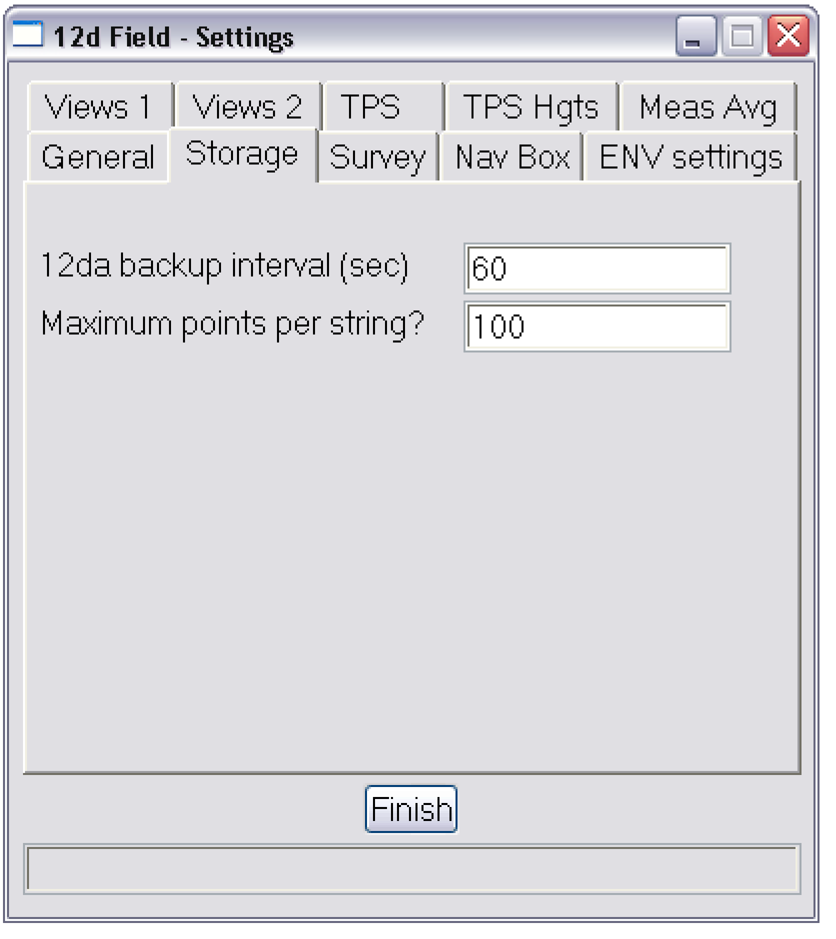

12d Field - Settings
General
Storage
Survey
Nav Box
ENV settings
Views 1
Views 2
TPS
TPS Hgts
Meas Avg
Perspective view
Auto pan perspective view?
Perspective Cross Hair size
0
Finish

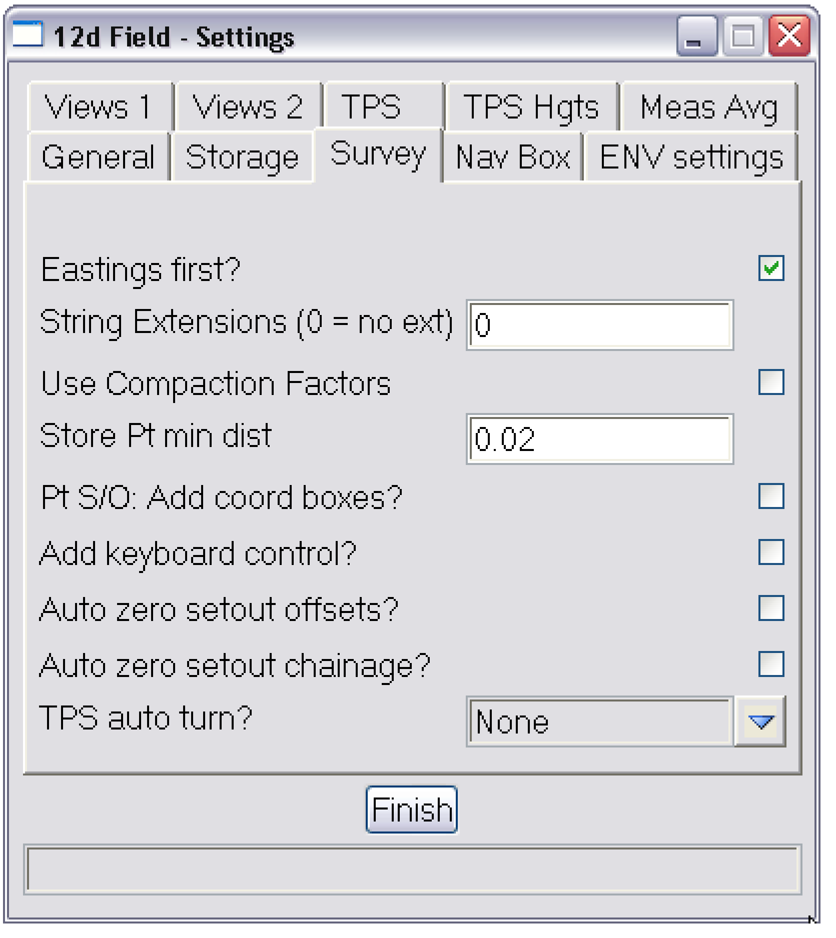

12d Field - Settings
General
Storage
Survey
Nav Box
ENV settings
Views 1
Views 2
TPS
TPS Hgts
Meas Avg
Hz-Dist PPM
0
Angle update tme in ms.
250
Show drifts
Finish

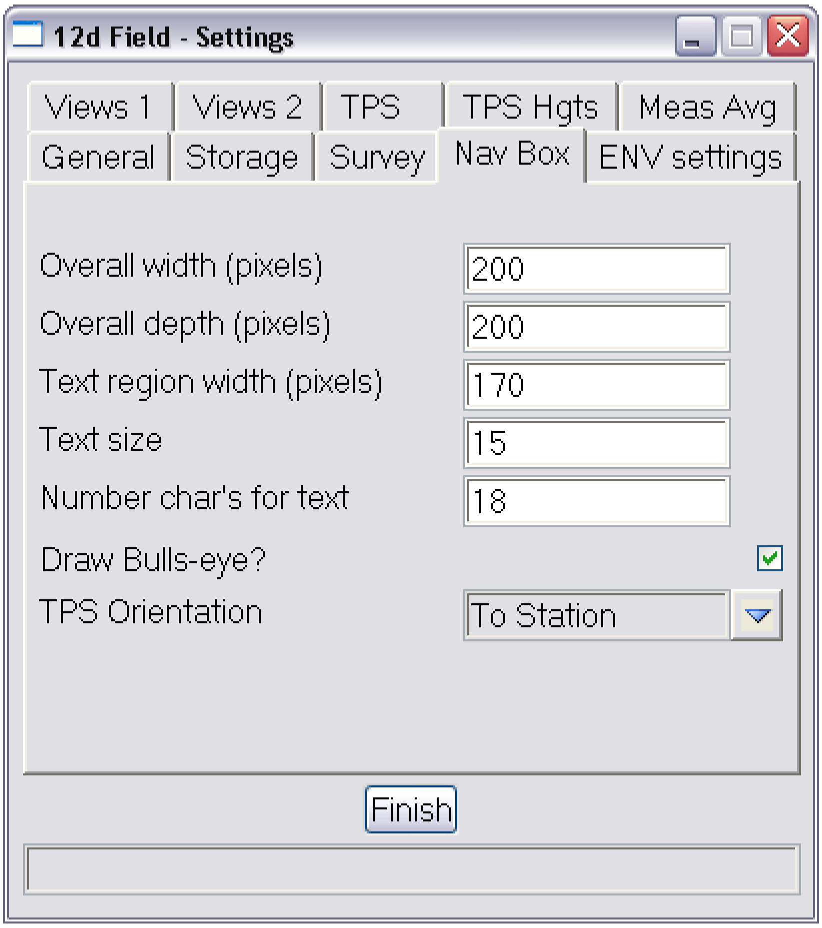

12d Field - Settings
General
Storage
Survey
Nav Box
ENV settings
Views 1
Views 2
TPS
TPS Hgts
Meas Avg
TPS Target Hgt 1
0
TPS Target Hgt 2
0.2
TPS Target Hgt 3
0.3
TPS Target Hgt 4
0.6
TPS Target Hgt 5
1.3
TPS Target Hgt 6
2.15
Finish

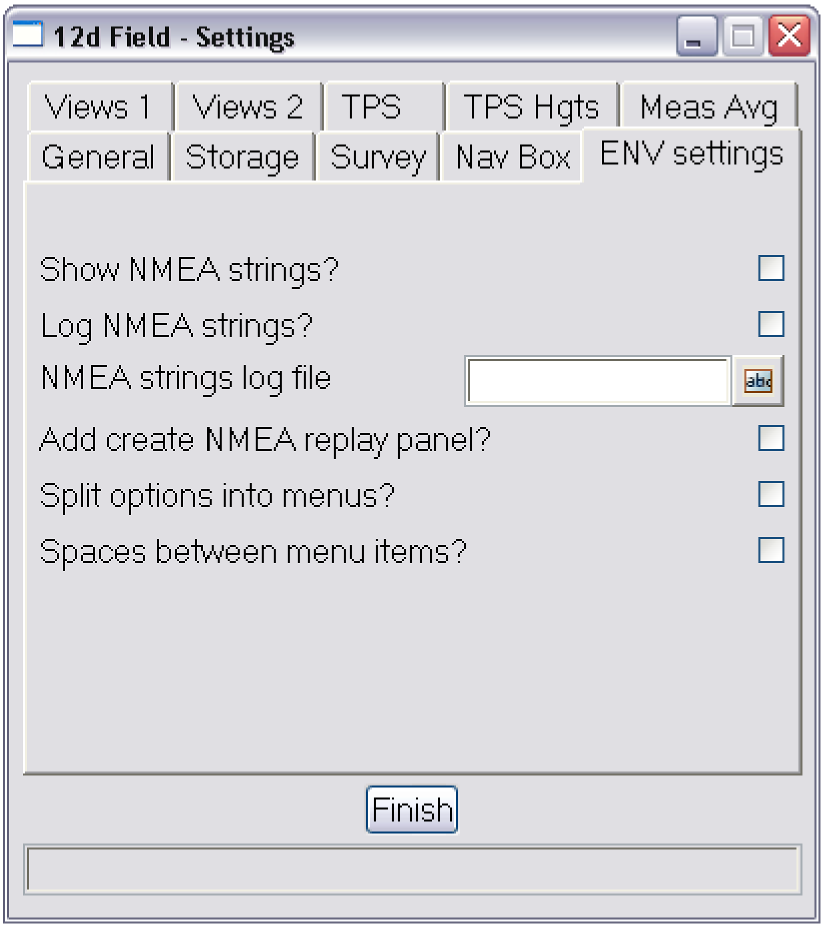

12d Field - Settings
General
Storage
Survey
Nav Box
ENV settings
Views 1
Views 2
TPS
TPS Hgts
Meas Avg
Avg Max Shots
10
Avg Min Shots
2
Avg Max XY StdDev
0.003
Avg Max Z StdDev
0.003
Finish


12d Field - Settings
Views 1
Views 2
TPS
TPS Hgts
Meas Avg
General
Storage
Survey
Nav Box
ENV settings
Font Size
20
On MEAS pressed action
Do Nothing
On STOP pressed action
Do Nothing
Prism Size (pixels)
5
Plan Pole Size (pixels)
15
Plan Pole Cross Size (pixels)
5
Surface Linestyle
1
Linestyle Colour
off yellow
Str Name on Control Bar
Finish


12d Field - Settings
Views 1
Views 2
TPS
TPS Hgts
Meas Avg
General
Storage
Survey
Nav Box
ENV settings
12da backup interval (sec)
60
Maximum points per string?
100
Finish


12d Field - Settings
Views 1
Views 2
TPS
TPS Hgts
Meas Avg
General
Storage
Survey
Nav Box
ENV settings
Eastings first?
String Extensions (0 = no ext)
0
Use Compaction Factors
Store Pt min dist
0.02
Pt S/O: Add coord boxes?
Add keyboard control?
Auto zero setout offsets?
Auto zero setout chainage?
TPS auto turn?
None
Finish


12d Field - Settings
Views 1
Views 2
TPS
TPS Hgts
Meas Avg
General
Storage
Survey
Nav Box
ENV settings
Overall width (pixels)
200
Overall depth (pixels)
200
Text region width (pixels)
170
Text size
15
Number char's for text
18
Draw Bulls-eye?
TPS Orientation
To Station
Finish



Continue to next section 18.2.1.8.20 Reconnect or return to 18.2 12d Field .

18.2.1.8.20 Reconnect

Selecting **Reconnect** will attempt to disconnect from the current instrument and then reconnect. This may be required when the instrument has been restarted.

18.2.2 GPS Localisation

Position of option on menu: **Survey =>Field 12d=>GPS Localisation**

The panel is used to create the localisation parameters used inside 12dField for reducing GPS observations into a local system.

Points collected with 12dField as raw WGS84 cartesian coordinates can be matched with local control points to calculate the parameters to convert GPS readings directly into the local system.

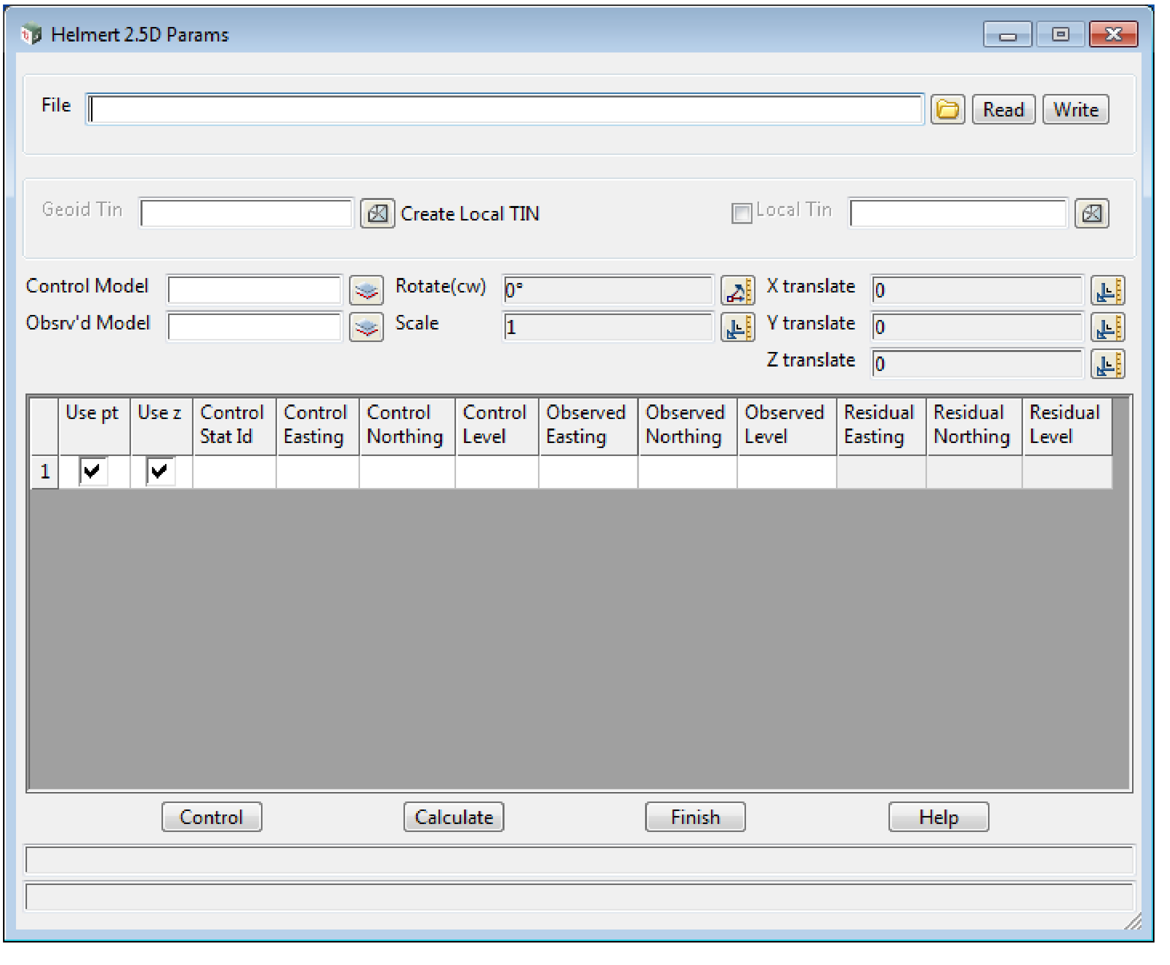
The 12dField localisation treats horizontal and vertical components separately. The horizontal transformation is a Helmert translation, translation, rotation and uniform scaling of the x and y axes.

The vertical translation can either be a simple z translation or be taken from a tin, the tin can be manually created outside of the panel, e.g. from AusGeoid98 or created by the panel via a plane of best fit.

The basic workflow for this panel is to observe known control points in the field with 12dField with just the ellipsoid set, e.g. MGA56.

Then use this panel to match the observed points with the control points to create the localisation.

Selecting **GPS Localisation** brings up the **Helmert 2.5D Params** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
File			*.tdf_hel

The TDF_HEL file to be read in or written to. This is the file used by 12dField to localise GPS reading.

Read button

Read in the TDF_HEL file

Write button

Write out the TDF_HEL file

Geoid Tin

If selected the value of the tin at the local coordinate is added to the height of the raw GPS coordinate.

Create Local TIN tick box

*If unticked and the **Local Tin** box is blank the points are adjusted by the straight mean z difference of the observations.*
*If unticked and the **Local Tin** box is not empty the points are adjusted by the value of the TIN at the xy location.*

*If ticked and the **Local Tin** box is not empty a plane of best fit is used to create the local TIN and the points are adjusted by the value of the TIN at the current xy location. The extent of the local TIN is 1000m outside the control used.*

Local Tin

*The **Local Tin** as described above.*

Control model model box

the model containing the control points.

Obsrv'd model model box

the model containing the observed points.

Rotate (cw) measure box At Point, Point to Point, String from Point, String to Point

The clockwise rotation parameter of the helmert transformation.

Scale measure box At Point, Point to Point, String from Point, String to Point

The scaling parameter of the helmert transformation.

X translate measure box At Point, Point to Point, String from Point, String to Point

The x translation of the helmert transformation.

Y translate measure box At Point, Point to Point, String from Point, String to Point

The y translation of the helmert transformation.

Z translate measure box At Point, Point to Point, String from Point, String to Point

The z translation of the helmert transformation, (note this is 0.0 if a local TIN is being used).

The grid

Please note changing the level of a control station to "null" means the station and corresponding observed point will not be used in the height calculations.

Use pt tick box

if ticked this point is used in the transformation calculations

Use z tick box

If unticked this point is not used to calculate the transformation height parameter.

Control Stat Id

The Id of the control station, will normally match the observed Id.

Control Easting/Northing/Level

the coordinate of the control station.

Observed Easting/Northing/Level

the coordinate of the observed point.

Residual Easting/Northing/Level

the delta of the observed point with the control point after the transformation has been applied.

Buttons

Control button

Start the selection of the control/observed point pairs. Note the environment variable PICK_ORDER_OBSERVED_FIRST_4D can be set to make the selection order 'observed' then 'control'. For ease of use 2 plan views should be used, one with the observed points and one with the control points.

Calculate button

Calculate the transformation and update the residuals in the grid control.

Finish button

Exit the panel, a warning message will appear if the transformation parameters have not yet been written to file.

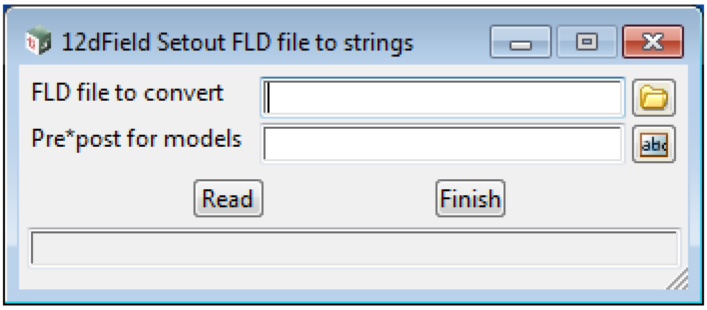
Continue to next section 18.2.3 12dField Setout FLD File To Strings or return to 18.2 12d Field .

18.2.3 12dField Setout FLD File To Strings

Position of option on menu: **Survey =>12d Field =>Setout FLD file to strings**

A field file, (**.FLD**) created by **12dField Setout** is attributed in such a way it can be read directly into ***12d Model*** without going through a Survey Data Reduction function, the string data will be read in with the original name, model, colour, linestyle and weight. Field files created by **12dField Setout** from version **V9C1e** can be read in with this panel.

Selecting **Setout FLD file to strings** displays the**12dField Setout FLD File To Strings** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
FLD file to convert	file		

the field file to be read in.

Pre*post for models

For more information please go to the section 4.19.2 Pre*Postfix in Panel Fields

Read	button		

the FLD file will be read in, if a point in the field file is not correctly attributed a message will be displayed in the output window and no point will be created.

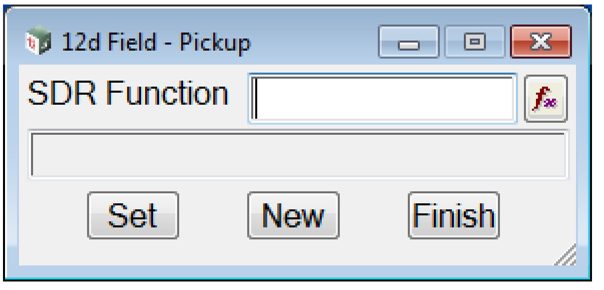
Continue to next section 18.2.4 12d Pickup or return to 18.2 12d Field .

18.2.4 12d Pickup

Position of option on menu: **Survey =>12d Field =>12d Pickup**

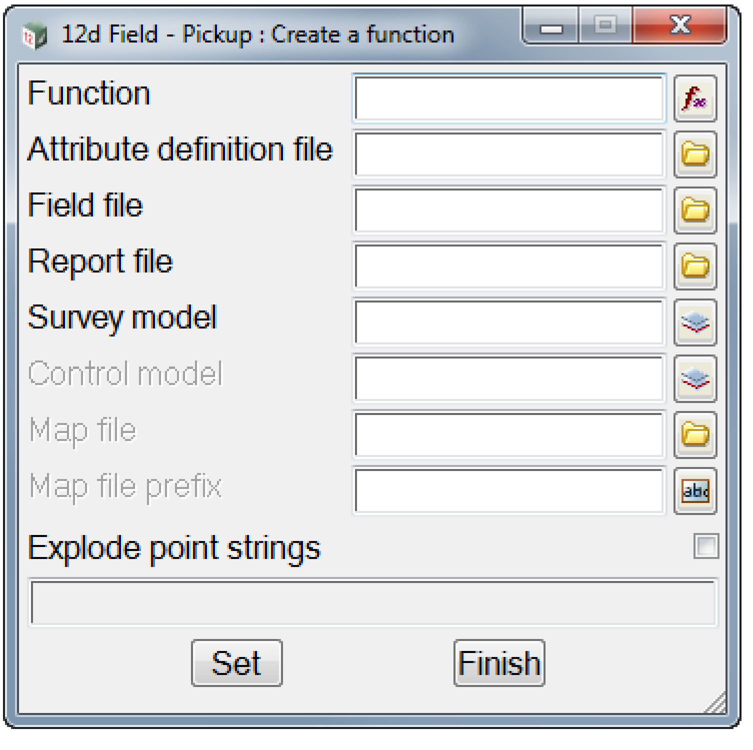
This section of documentation is a work in progress and will be updated in subsequent releases.

Selecting **12d Pickup** brings up the **12d Field - Pickup** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
SDR Function	function box		available functions
Set	button		
New	button		



The fields and buttons used in this panel have the following functions.

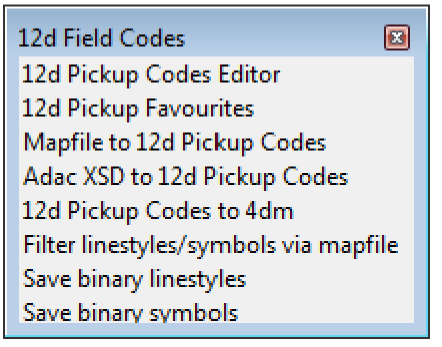
Field Description	Type	Defaults	Pop-Up
Function	function box		available functions

Attribute definition file	file box
Field file	file box
Report file	file box
Survey model	model box
Control model	model box
Map file	file box
Map file prefix	input
Explode point strings	tick box
Set	button

Continue to next section 18.2.5 12d Pickup Codes or return to 18.2 12d Field .

18.2.5 12d Pickup Codes

The **12d Pickup codes** walk right menu is



For information on

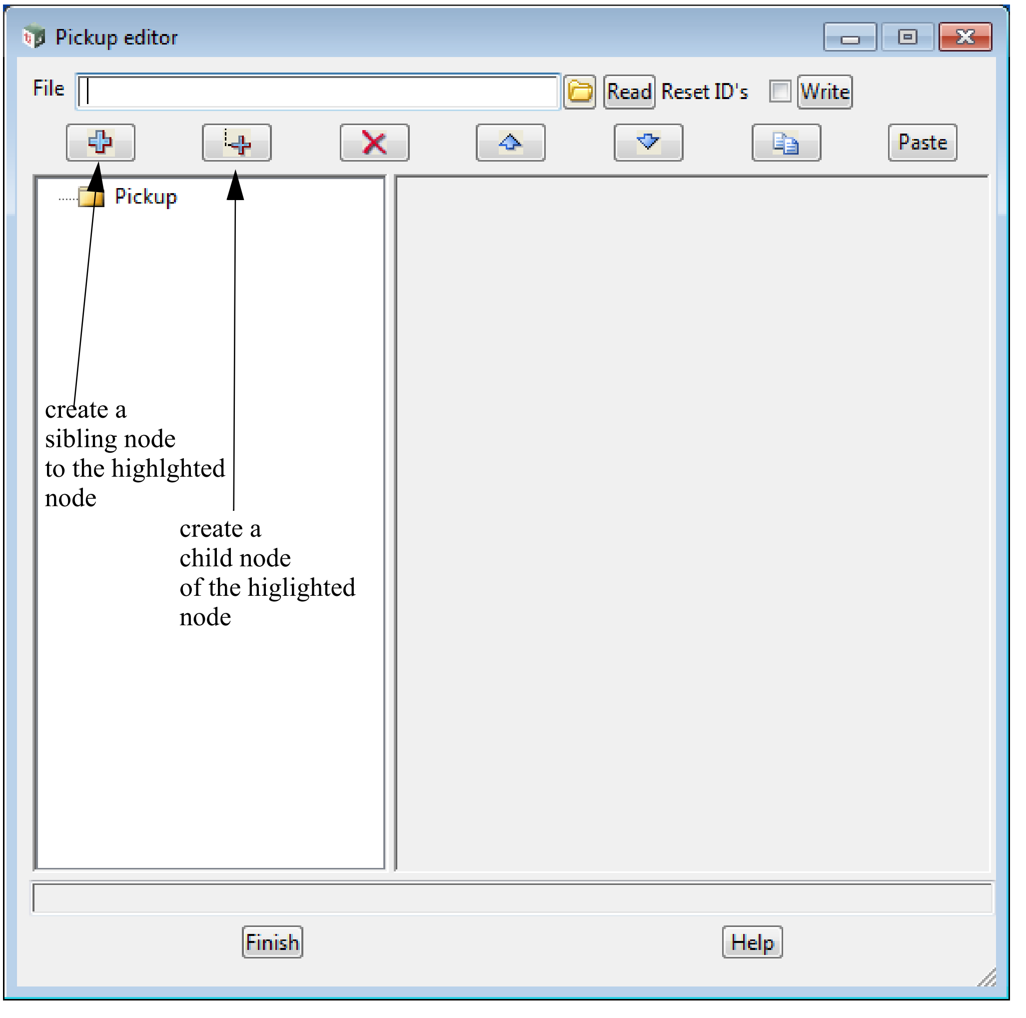
12d Pickup Codes Editor go to	18.2.5.1 Pickup Editor
12d Pickup Favourites	18.2.5.2 Pickup Favourites
Mapfile to 12d Pickup Codes	18.2.5.3 Mapfile to 12d Pickup Codes
12d Pickup Codes to 4dm	18.2.5.5 12d Pickup Codes to Macro
Filter linestyles/symbols via mapfile	18.2.5.6 Filter Linestyles/Symbols via Mapfile
Save binary linestyles	18.2.5.7 Save Binary Linestyles
Save binary symbols	18.2.5.8 Save Binary Symbols

18.2.5.1 Pickup Editor

Position of option on menu: **Survey =>12d Field =>12d Field Pickup Codes =>12d Pickup Codes Editor**

The **Pickup Editor** panel allows you to define and edit feature codes to be used in **12d Field Pickup**.

Selecting **12d Pickup Codes Editor** brings up the **Pickup Editor** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

File

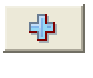
the 12d Field pickup codes file to edit or create

Read

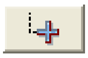
reads the pickup codes file

Write

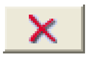
writes the pickup codes file



add a new node at the current level of the highlighted node in the tree - a sibling node. You can't add a sibling node to the top level ***Pickup*** *node.*



add a child to the current highlighted node in the tree.



delete the current highlighted node in the tree

moves the current highlighted node in the tree up

moves the current highlighted node down in the tree

copies to the cache the current highlighted node in the tree

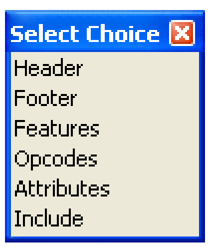
pastes the cached node to the current node in the tree (if it is allowed).

Continue to next section 18.2.5.1.1 Defining Codes or return to 18.2 12d Field .

18.2.5.1.1 Defining Codes

There are a number of different code types that can be added to your pickup codes file, depending on the structure of your tree.

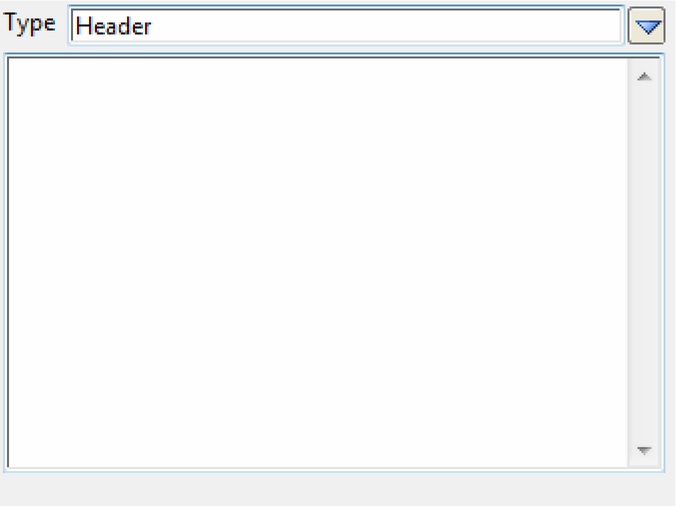
Each code has the following field **Type,** which can be used to define the type:



For information on Header go to 18.2.5.1.1.1 Header
Footer 18.2.5.1.1.2 Footer
Features 18.2.5.1.1.4 Features
Opcodes 18.2.5.1.1.5 Opcodes
Attributes 18.2.5.1.1.6 Attributes
Include 18.2.5.1.1.3 Include

18.2.5.1.1.1 Header

A **Header** to be included in any macro code generation.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

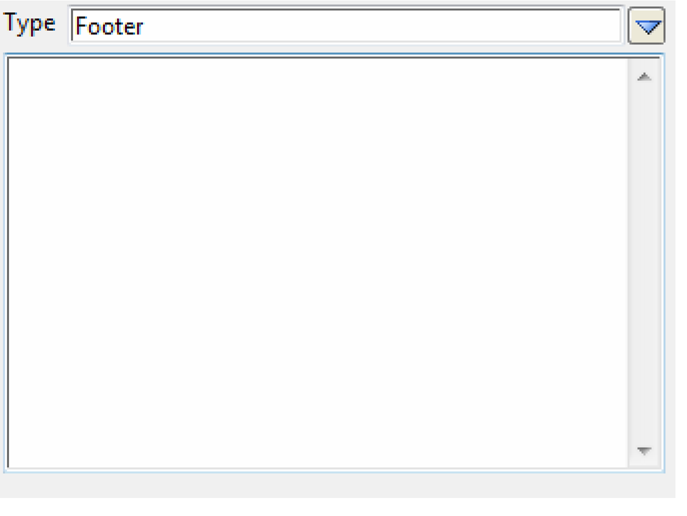
Text field

the macro code to include in the header

Continue to next section 18.2.5.1.1.2 Footer or return to 18.2 12d Field .

18.2.5.1.1.2 Footer

A footer to be included in any macro code generation.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

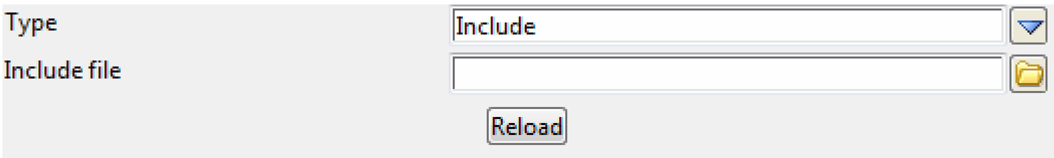
Text field

the macro code to include in the footer

Continue to next section 18.2.5.1.1.3 Include or return to 18.2 12d Field .

18.2.5.1.1.3 Include

Defines another code file to be included.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Include file	file		

the include file

Field Description	Type	Defaults	Pop-Up
Reload	button		

reloads the include file

Note that **include files** can be edited inline, within one editor.

Continue to next section 18.2.5.1.1.4 Features or return to 18.2 12d Field .

18.2.5.1.1.4 Features

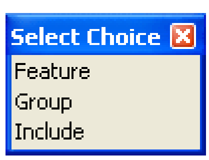
This code type defines a group of features

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Prompt	input		

the prompt name of the features group to appear in 12d Field pickup

Available Children Types

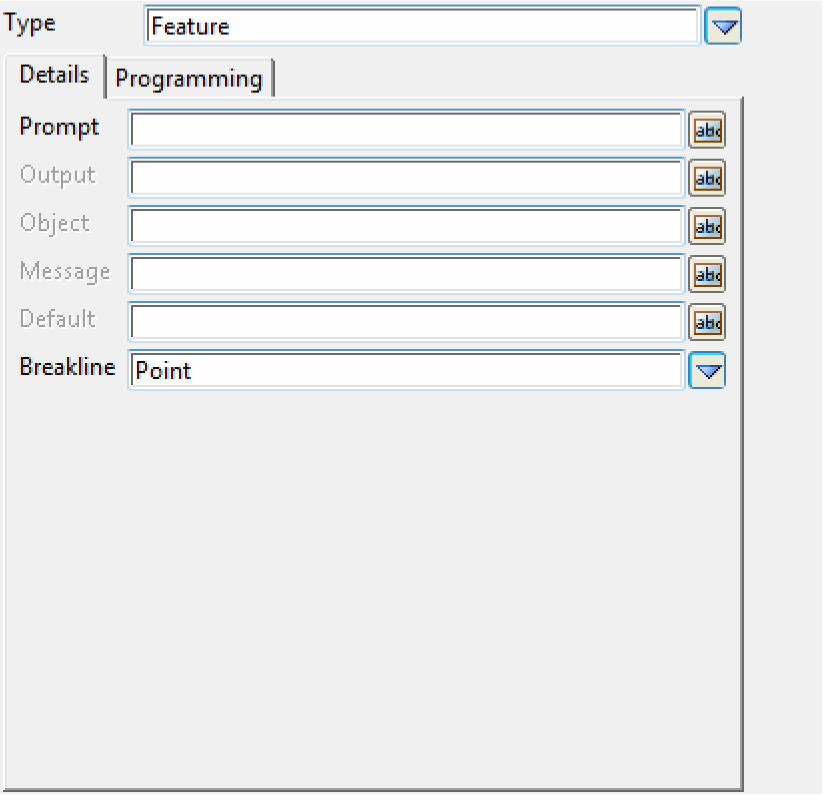


Feature 18.2.5.1.1.4.1 Feature
Group 18.2.5.1.1.4.2 Group (Features)
Include 18.2.5.1.1.3 Include

Continue to next section 18.2.5.1.1.4.1 Feature or return to 18.2 12d Field .

18.2.5.1.1.4.1Feature

This code defines a feature to be picked up in the field



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Details Tab

Prompt input

the cosmetic name to be displayed during 12d field pickup

Output input

the optional value to output to the pickup file - uses prompt if undefined

Object input

the optional value to be used to define a set of attributes - uses output if undefined

Message input

a message to display when picking up the code

Default input

the default value

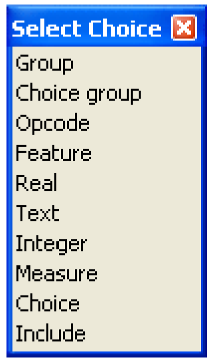
Breakline choice box Both, Point, Line

the type of breakline

Programming Tab

See 18.2.5.1.1.7 Programming

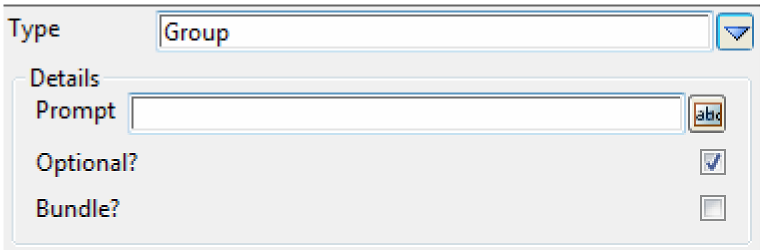
Children types



For information on Group go to	18.2.5.1.1.4.2 Group (Features)
Choice Group	18.2.5.1.1.4.3 Choice Group Attribute
Opcode	18.2.5.1.1.4.6 Opcode Attribute
Feature	18.2.5.1.1.4.7 Feature Attribute
Real	18.2.5.1.1.4.8 Real Attribute
Text	18.2.5.1.1.4.9 Text Attribute
Integer	18.2.5.1.1.4.10 Integer Attribute
Measure	18.2.5.1.1.4.11 Measure Attribute
Choice	18.2.5.1.1.4.12 Choice Attribute
Include	18.2.5.1.1.3 Include

18.2.5.1.1.4.2Group (Features)

Defines a group of other features or feature attributes



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Prompt	input		

the cosmetic name for the group

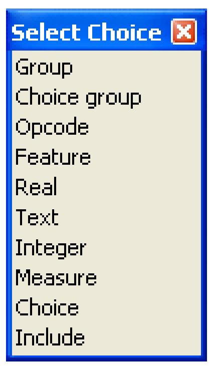
Optional?	tick box		

if ticked, filling out the group is optional
if not ticked, at least one field in the group must be filled out

Bundle?	tick box		

if ticked, this group is treated as a bundle
if not ticked, this group is treated as a normal group

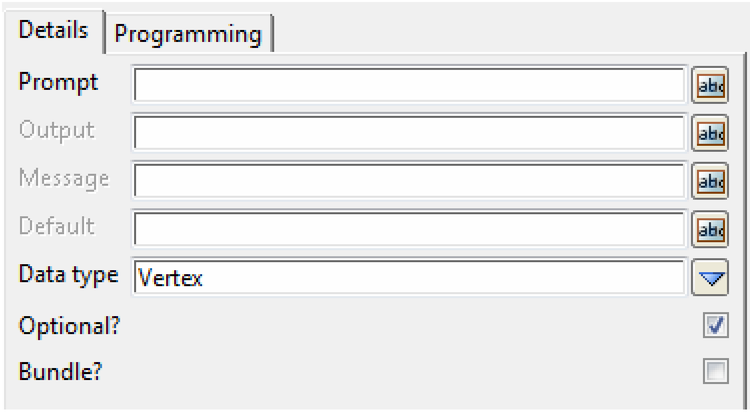
Available Children Types



For information on Group go to 18.2.5.1.1.4.2 Group (Features)
Choice Group 18.2.5.1.1.4.3 Choice Group Attribute
Opcode 18.2.5.1.1.4.6 Opcode Attribute
Feature 18.2.5.1.1.4.7 Feature Attribute
Real 18.2.5.1.1.4.8 Real Attribute
Text 18.2.5.1.1.4.9 Text Attribute
Integer 18.2.5.1.1.4.10 Integer Attribute
Measure 18.2.5.1.1.4.11 Measure Attribute
Choice 18.2.5.1.1.4.12 Choice Attribute
Include 18.2.5.1.1.3 Include

18.2.5.1.1.4.3Choice Group Attribute

This allows the definition of an attribute defined by a set of choices, defined into groups, to attach to a feature.



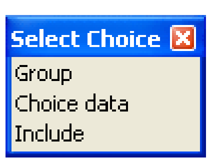
The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Details Tab

Prompt	input		
the cosmetic name to be displayed during 12d field pickup			
Output	input		
the optional value to output to the pickup file - uses prompt if undefined			
Message	input		
a message to display when displaying the choice group			
Default	input		
the default value			
Data type	choice box		Vertex, Next Segment, Prev Segment, String, Prompt, Prompt Segment
how the attribute should be attached			
Optional?	tick box		
if ticked, this attribute is treated as optional *if not ticked, this attribute must be filled out*			
Bundle?	tick box		
if ticked, this attribute is treated as a bundle *if not ticked, this attribute is treated as a normal attribute*			

Available children types



For information on Group go to	18.2.5.1.1.4.4 Group (Choice)
Choice Data	18.2.5.1.1.4.5 Choice Data
Include	18.2.5.1.1.3 Include

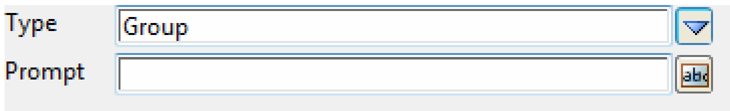
Programming Tab

See 18.2.5.1.1.7 Programming

Continue to next section 18.2.5.1.1.4.4 Group (Choice) or return to 18.2 12d Field .

18.2.5.1.1.4.4Group (Choice)

Allows a group of choice group attribute related data to be defined

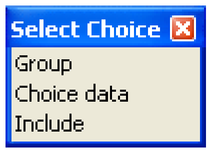


The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Prompt	input		

the cosmetic name of the group to display

Available children types



For information on Group go to [18.2.5.1.1.4.4 Group (Choice)](#)
Choice Data [18.2.5.1.1.4.5 Choice Data](#)
Include [18.2.5.1.1.3 Include](#)

Continue to next section 18.2.5.1.1.4.5 Choice Data or return to 18.2 12d Field .

18.2.5.1.1.4.5Choice Data

The choices to display for a choice attribute.

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Data Items

Grid

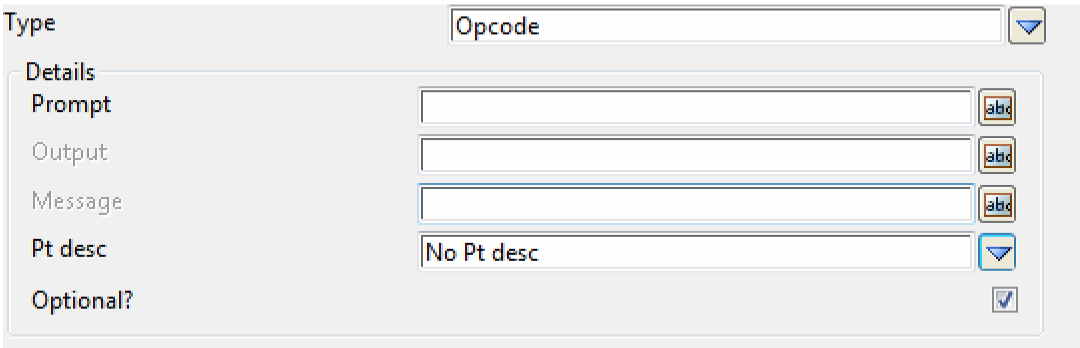
Item

the value for a choice

Continue to next section 18.2.5.1.1.4.6 Opcode Attribute or return to 18.2 12d Field .

18.2.5.1.1.4.6Opcode Attribute

This defines an attribute attached to a feature, which requires the entry of an opcode.



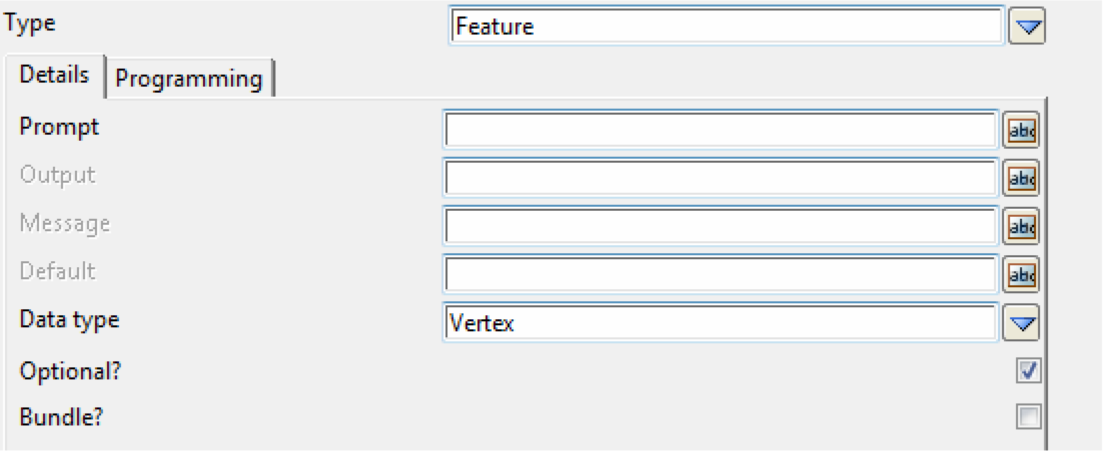
The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Prompt	input		
the cosmetic name for the opcode			
Output	input		
the optional output for the opcode - defaults to prompt if not defined			
Message	input		
an optional message to display when entering the opcode value			
Pt desc	choice box		No pt desc, Pt desc, Null pt desc
the point description			
Optional	tick box		
if ticked, this opcode attribute is optional *if not ticked, this opcode attribute must be entered in the field*			

Continue to next section 18.2.5.1.1.4.7 Feature Attribute or return to 18.2 12d Field .

18.2.5.1.1.4.7Feature Attribute

This defines an attribute attached to a feature, which requires the entry of another feature.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Details Tab

Prompt input

the cosmetic name to be displayed during 12d field pickup

Output input

the optional value to output to the pickup file - uses prompt if undefined

Message input

a message to display when picking up the code

Default input

the default value

Data type choice box — Vertex, Next Segment, Prev Segment, String, Prompt, Prompt Segment

how the attribute should be attached

Optional? tick box

if ticked, this attribute is treated as optional
if not ticked, this attribute must be filled out

Bundle? tick box

if ticked, this attribute is treated as a bundle
if not ticked, this attribute is treated as a normal attribute

Programming Tab

See 18.2.5.1.1.7 Programming

Continue to next section 18.2.5.1.1.4.8 Real Attribute or return to 18.2 12d Field .

18.2.5.1.1.4.8Real Attribute

This defines an attribute attached to a feature, which requires the entry of a real value.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Details Tab

Prompt	input		
the cosmetic name to be displayed during 12d field pickup			
Output	input		
the optional value to output to the pickup file - uses prompt if undefined			
Message	input		
a message to display when picking up the code			
Default	input		
the default value			
Data type	choice box		Vertex, Next Segment, Prev Segment, String, Prompt, Prompt Segment
how the attribute should be attached			
Optional?	tick box		
if ticked, this attribute is treated as optional *if not ticked, this attribute must be filled out*			
Bundle?	tick box		
if ticked, this attribute is treated as a bundle *if not ticked, this attribute is treated as a normal attribute*			

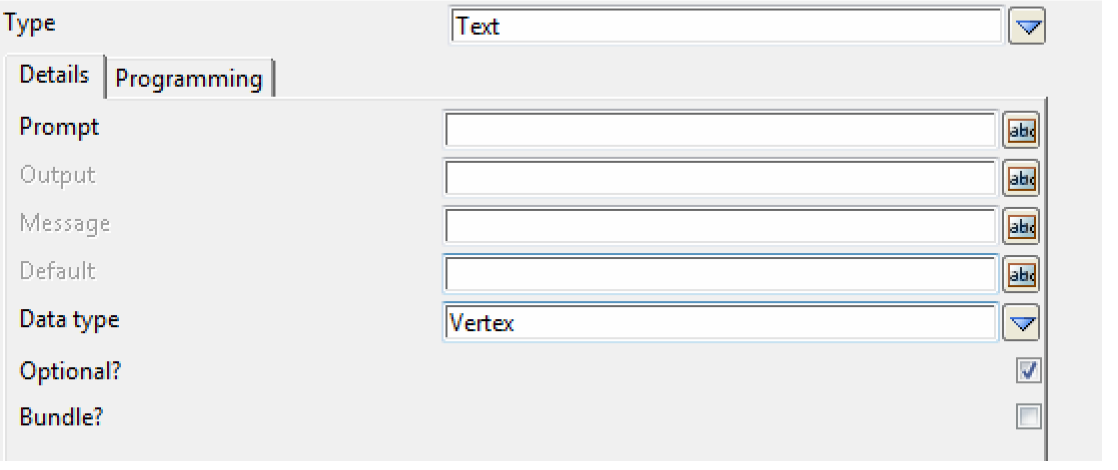
Programming Tab

See 18.2.5.1.1.7 Programming

Continue to next section 18.2.5.1.1.4.9 Text Attribute or return to 18.2 12d Field .

18.2.5.1.1.4.9Text Attribute

This defines an attribute attached to a feature, which requires the entry of a text value.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Details Tab

Prompt input

the cosmetic name to be displayed during 12d field pickup

Output input

the optional value to output to the pickup file - uses prompt if undefined

Message input

a message to display when picking up the code

Default input

the default value

Data type choice box — Vertex, Next Segment, Prev Segment, String, Prompt, Prompt Segment

how the attribute should be attached

Optional? tick box

if ticked, this attribute is treated as optional
if not ticked, this attribute must be filled out

Bundle? tick box

if ticked, this attribute is treated as a bundle
if not ticked, this attribute is treated as a normal attribute

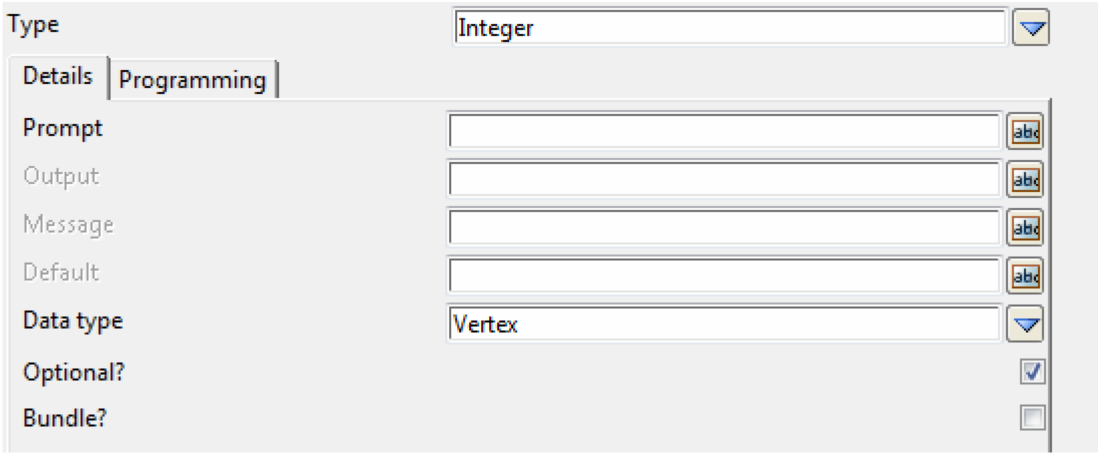
Programming Tab

See 18.2.5.1.1.7 Programming

Continue to next section 18.2.5.1.1.4.10 Integer Attribute or return to 18.2 12d Field .

18.2.5.1.1.4.10 Integer Attribute

This defines an attribute attached to a feature, which requires the entry of an integer value.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Details Tab			
Prompt	input		
the cosmetic name to be displayed during 12d field pickup			
Output	input		
the optional value to output to the pickup file - uses prompt if undefined			
Message	input		
a message to display when picking up the code			
Default	input		
the default value			
Data type	choice box		Vertex, Next Segment, Prev Segment, String, Prompt, Prompt Segment
how the attribute should be attached			
Optional?	tick box		
if ticked, this attribute is treated as optional *if not ticked, this attribute must be filled out*			
Bundle?	tick box		
if ticked, this attribute is treated as a bundle *if not ticked, this attribute is treated as a normal attribute*			

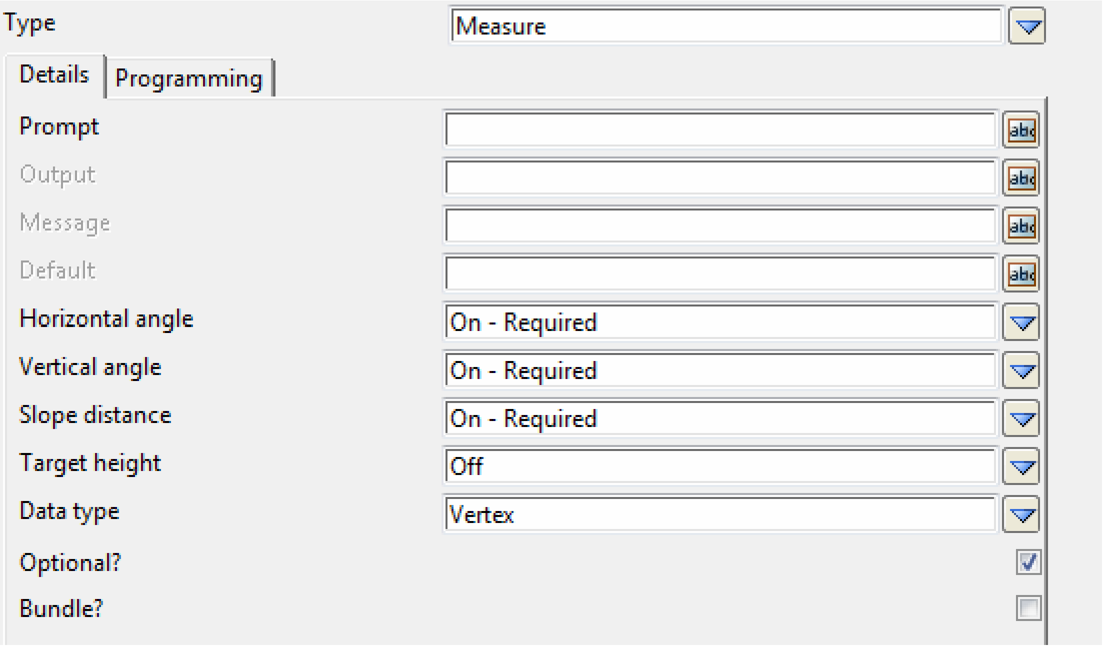
Programming Tab

See 18.2.5.1.1.7 Programming

Continue to next section 18.2.5.1.1.4.11 Measure Attribute or return to 18.2 12d Field .

18.2.5.1.1.4.11 Measure Attribute

This defines an attribute attached to a feature, which requires a physical measurement.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Details Tab

Prompt input

the cosmetic name to be displayed during 12d field pickup

Output input

the optional value to output to the pickup file - uses prompt if undefined

Message input

a message to display when picking up the code

Default input

the default value

Horizontal angle choice box — Off, On - Required, On - Optional

if the horizontal angle is to be captured

Vertical angle choice box — Off, On - Required, On -Optional

if the vertical angle is to be captured

Slope distance choice box — Off, On - Required, On -Optional

if the slope distance is to be captured

Target height choice box — Off, On - Required, On - Optional

if the target height is required

Data type	choice box	Vertex, Next Segment, Prev Segment, String, Prompt, Prompt Segment

how the attribute should be attached

Optional?	tick box

if ticked, this attribute is treated as optional
if not ticked, this attribute must be filled out

Bundle?	tick box

if ticked, this attribute is treated as a bundle
if not ticked, this attribute is treated as a normal attribute

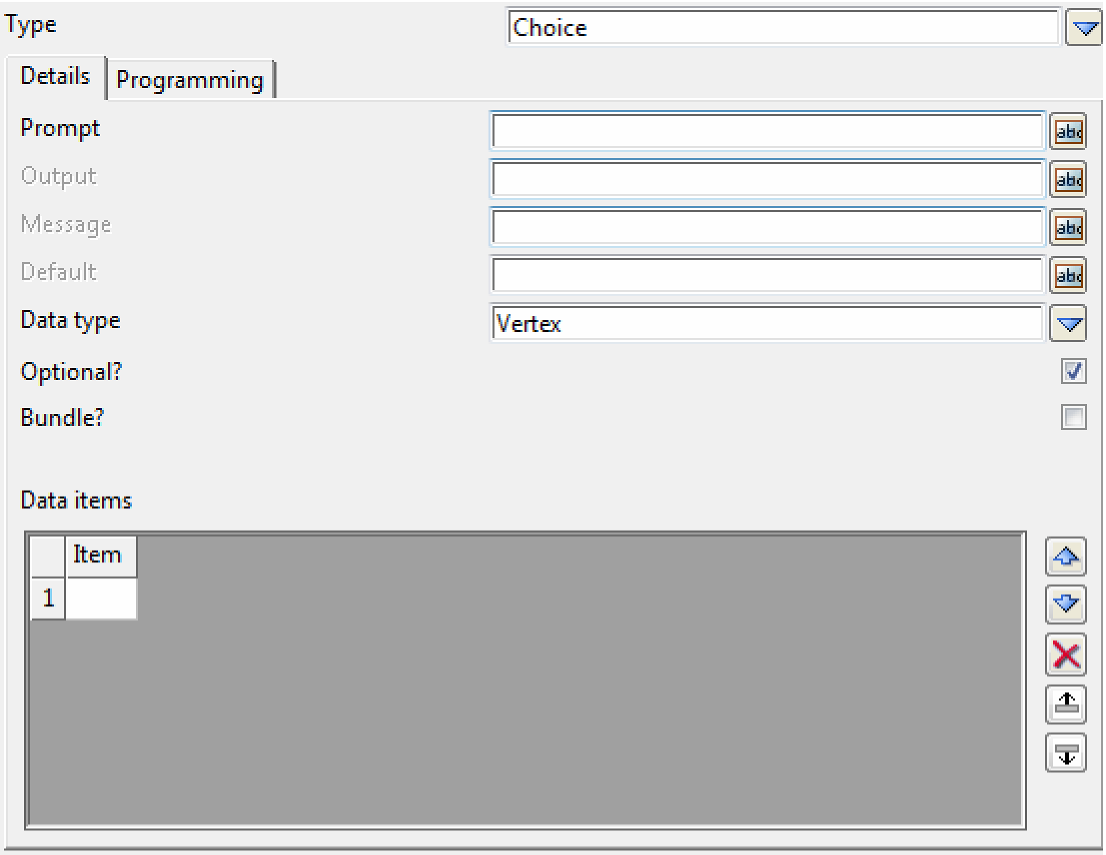
Programming Tab

See 18.2.5.1.1.7 Programming

Continue to next section 18.2.5.1.1.4.12 Choice Attribute or return to 18.2 12d Field .

18.2.5.1.1.4.12Choice Attribute

This defines an attribute attached to a feature, which requires a value to be selected from a list of choices.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Details Tab

Prompt input

the cosmetic name to be displayed during 12d field pickup

Output input

the optional value to output to the pickup file - uses prompt if undefined

Message input

a message to display when picking up the code

Default input

the default value

Data type choice box — Vertex, Next Segment, Prev Segment, String, Prompt, Prompt Segment

how the attribute should be attached

Optional? tick box

if ticked, this attribute is treated as optional
if not ticked, this attribute must be filled out

Bundle? tick box

if ticked, this attribute is treated as a bundle
if not ticked, this attribute is treated as a normal attribute

Data Items grid

Grid

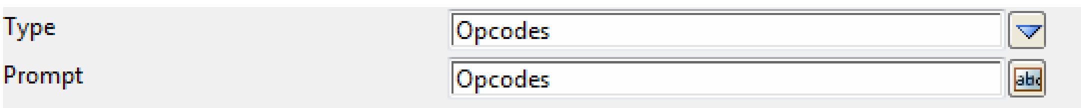
Item grid

a choice item to display

Continue to next section 18.2.5.1.1.5 Opcodes or return to 18.2 12d Field .

18.2.5.1.1.5 Opcodes

This code type defines a group of opcodes.

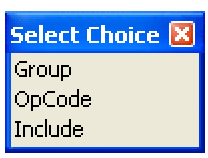


The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Prompt	input		

the cosmetic name of the opcodes group

Available children types

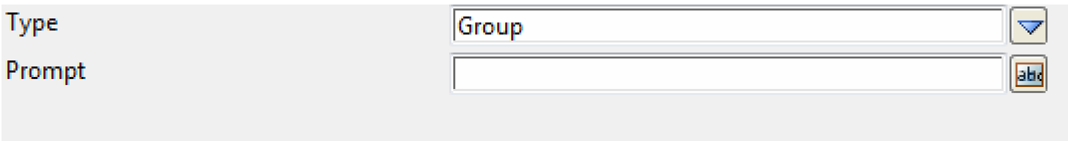


For information on Group go to 18.2.5.1.1.5.1 Group (Opcodes)
OpCode 18.2.5.1.1.5.2 OpCode
Include 18.2.5.1.1.3 Include

Continue to next section 18.2.5.1.1.5.1 Group (Opcodes) or return to 18.2 12d Field .

18.2.5.1.1.5.1Group (Opcodes)

This code defines a group of opcodes

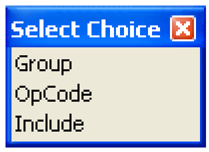


The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Prompt	input		

the cosmetic name of the group of opcodes

Available children types

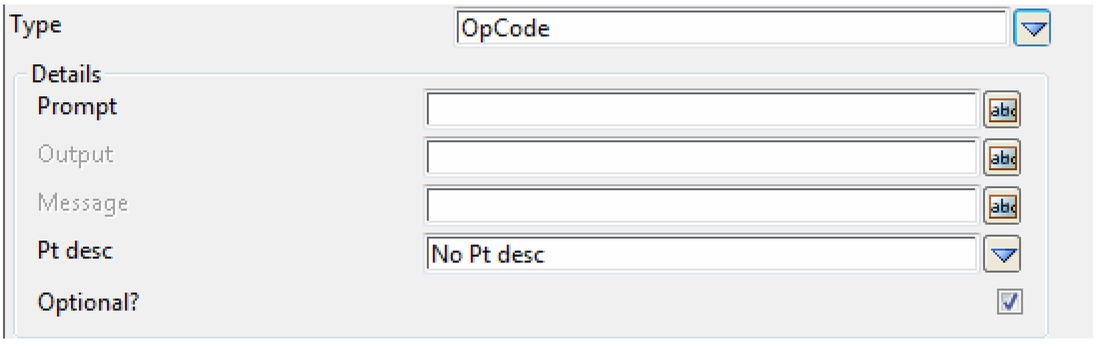


For information on Group go to	18.2.5.1.1.5.1 Group (Opcodes)
OpCode	18.2.5.1.1.5.2 OpCode
Include	18.2.5.1.1.3 Include

Continue to next section 18.2.5.1.1.5.2 OpCode or return to 18.2 12d Field .

18.2.5.1.1.5.2OpCode

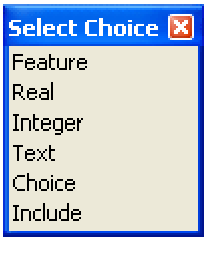
This defines a custom opcode that can be attached to the running pickup function or a picked up feature.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Prompt	input		
the cosmetic name to be displayed during 12d field pickup			
Output	input		
the optional value to output to the pickup file - uses prompt if undefined			
Message	input		
a message to display when entering the opcode			
Pt desc	choice box		No pt desc, Pt desc, Null pt desc
the point description			
Optional?	tick box		
if ticked, this opcode is treated as optional *if not ticked, it is required*			

Available children types

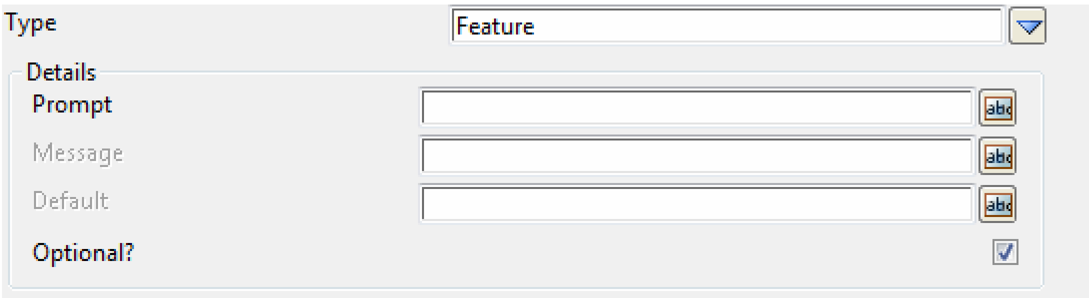


For information on Feature go to 18.2.5.1.1.5.3 Feature Attribute (OpCode)
Real 18.2.5.1.1.5.4 Real Attribute (OpCode)
Integer 18.2.5.1.1.5.5 Integer Attribute (OpCode)
Text 18.2.5.1.1.5.6 Text Attribute (OpCode)
Choice 18.2.5.1.1.5.7 Choice Attribute (OpCode)
Include 18.2.5.1.1.3 Include

Continue to next section 18.2.5.1.1.5.3 Feature Attribute (OpCode) or return to 18.2 12d Field .

18.2.5.1.1.5.3Feature Attribute (OpCode)

This defines an attribute attached to an opcode, which requires the entry of a feature.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Prompt	input		

the cosmetic name to be displayed during 12d field pickup

Message	input		

a message to display when entering the opcode attribute

Default	input		

the default value for the attribute

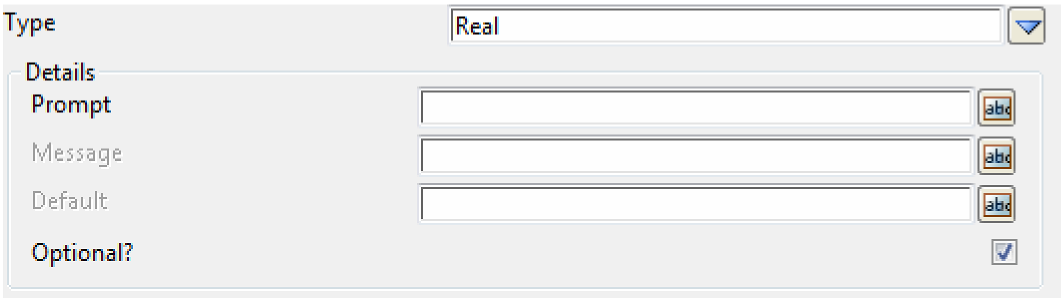
Optional?	tick box		

if ticked, this attribute is treated as optional
if not ticked, it is required

Continue to next section 18.2.5.1.1.5.4 Real Attribute (OpCode) or return to 18.2 12d Field .

18.2.5.1.1.5.4Real Attribute (OpCode)

This defines an attribute attached to an opcode, which requires the entry of a real value.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Prompt	input		

the cosmetic name to be displayed during 12d field pickup

Message	input		

a message to display when entering the opcode attribute

Default	input		

the default value for the attribute

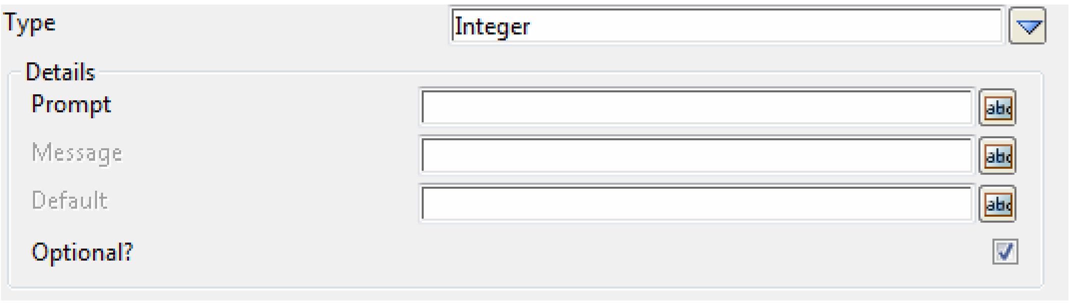
Optional?	tick box		

if ticked, this attribute is treated as optional
if not ticked, it is required

Continue to next section 18.2.5.1.1.5.5 Integer Attribute (OpCode) or return to 18.2 12d Field .

18.2.5.1.1.5.5Integer Attribute (OpCode)

This defines an attribute attached to an opcode, which requires the entry of an integer value.



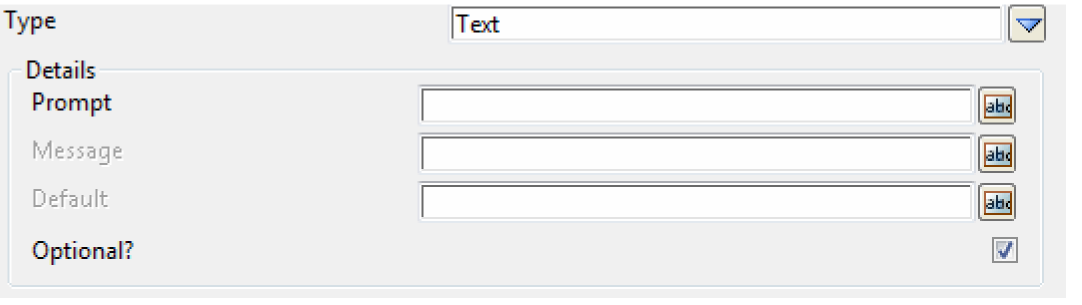
The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Prompt	input		
the cosmetic name to be displayed during 12d field pickup			
Message	input		
a message to display when entering the opcode attribute			
Default	input		
the default value for the attribute			
Optional?	tick box		
if ticked, this attribute is treated as optional *if not ticked, it is required*			

Continue to next section 18.2.5.1.1.5.6 Text Attribute (OpCode) or return to 18.2 12d Field .

18.2.5.1.1.5.6Text Attribute (OpCode)

This defines an attribute attached to an opcode, which requires the entry of a text value.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Prompt	input		

the cosmetic name to be displayed during 12d field pickup

Message	input		

a message to display when entering the opcode attribute

Default	input		

the default value for the attribute

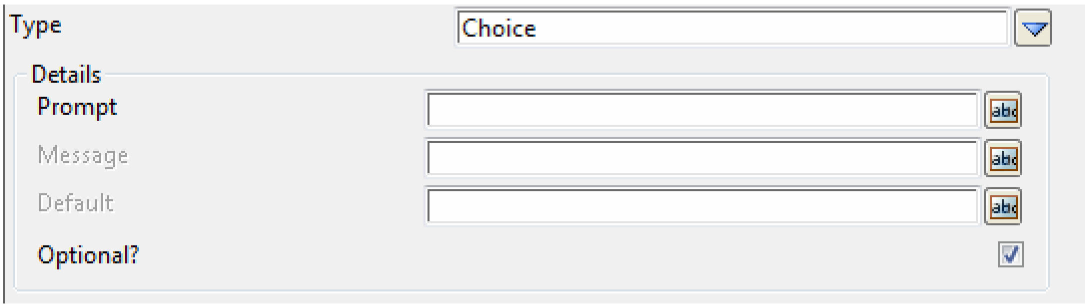
Optional?	tick box		

if ticked, this attribute is treated as optional
if not ticked, it is required

Continue to next section 18.2.5.1.1.5.7 Choice Attribute (OpCode) or return to 18.2 12d Field .

18.2.5.1.1.5.7Choice Attribute (OpCode)

This defines an attribute attached to an opcode, which requires the entry of a value selected from a choice.



The fields and buttons used in this panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Prompt input

the cosmetic name to be displayed during 12d field pickup

Message input

a message to display when entering the opcode attribute

Default input

the default value for the attribute

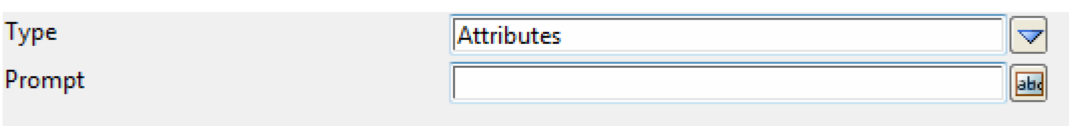
Optional? tick box

if ticked, this attribute is treated as optional
if not ticked, it is required

Continue to next section 18.2.5.1.1.6 Attributes or return to 18.2 12d Field .

18.2.5.1.1.6 Attributes

This defines a top level group of attributes.



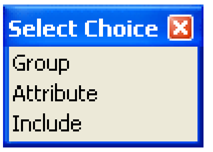
The fields and buttons used in this panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Prompt input

the cosmetic name for the group of attributes.

Available children types

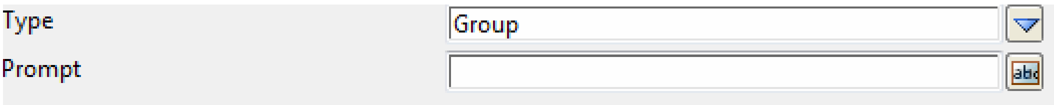


For information on Group go to 18.2.5.1.1.6.1 Group (Attributes)
Attribute 18.2.5.1.1.6.2 Attribute
Include 18.2.5.1.1.3 Include

Continue to next section 18.2.5.1.1.6.1 Group (Attributes) or return to 18.2 12d Field .

18.2.5.1.1.6.1Group (Attributes)

This defines a group of attributes or other groups.

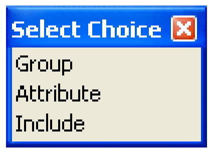


The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Prompt	input		

the cosmetic name for the group that will be displayed

Available children types

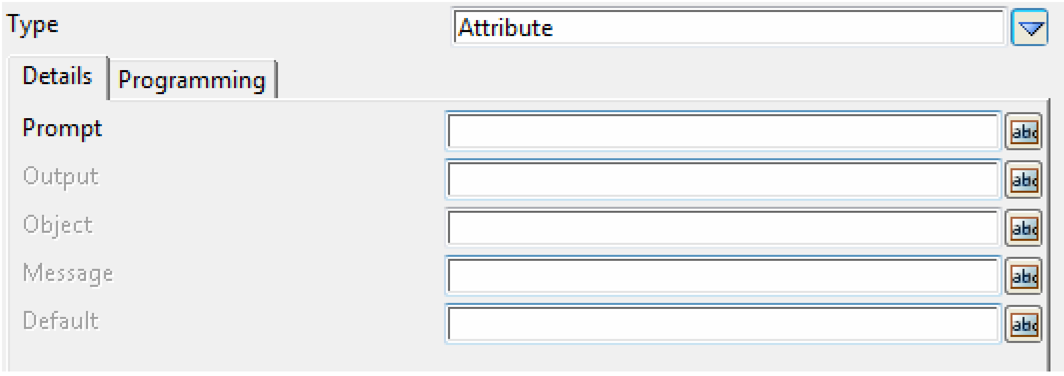


For information on Group go to 18.2.5.1.1.6.1 Group (Attributes)
Attribute 18.2.5.1.1.6.2 Attribute
Include 18.2.5.1.1.3 Include

Continue to next section 18.2.5.1.1.6.2 Attribute or return to 18.2 12d Field .

18.2.5.1.1.6.2Attribute

This defines an attribute that may be attached to a picked up point or string during pickup.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Details Tab

Prompt input

the cosmetic name to be displayed during 12d field pickup

Output input

the optional value to output to the pickup file - uses prompt if undefined

Object input

the optional value to be used to define a set of attributes - uses output if undefined

Message input

a message to display when picking up the code

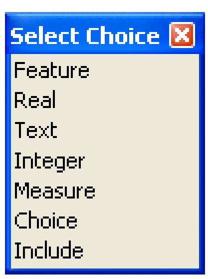
Default input

the default value

Programming Tab

See 18.2.5.1.1.7 Programming

Available children types



For information on Feature go to	18.2.5.1.1.4.7 Feature Attribute
Real	18.2.5.1.1.4.8 Real Attribute
Text	18.2.5.1.1.4.9 Text Attribute
Integer	18.2.5.1.1.4.10 Integer Attribute

Measure	18.2.5.1.1.4.11 Measure Attribute
Choice	18.2.5.1.1.4.12 Choice Attribute
Include	18.2.5.1.1.3 Include

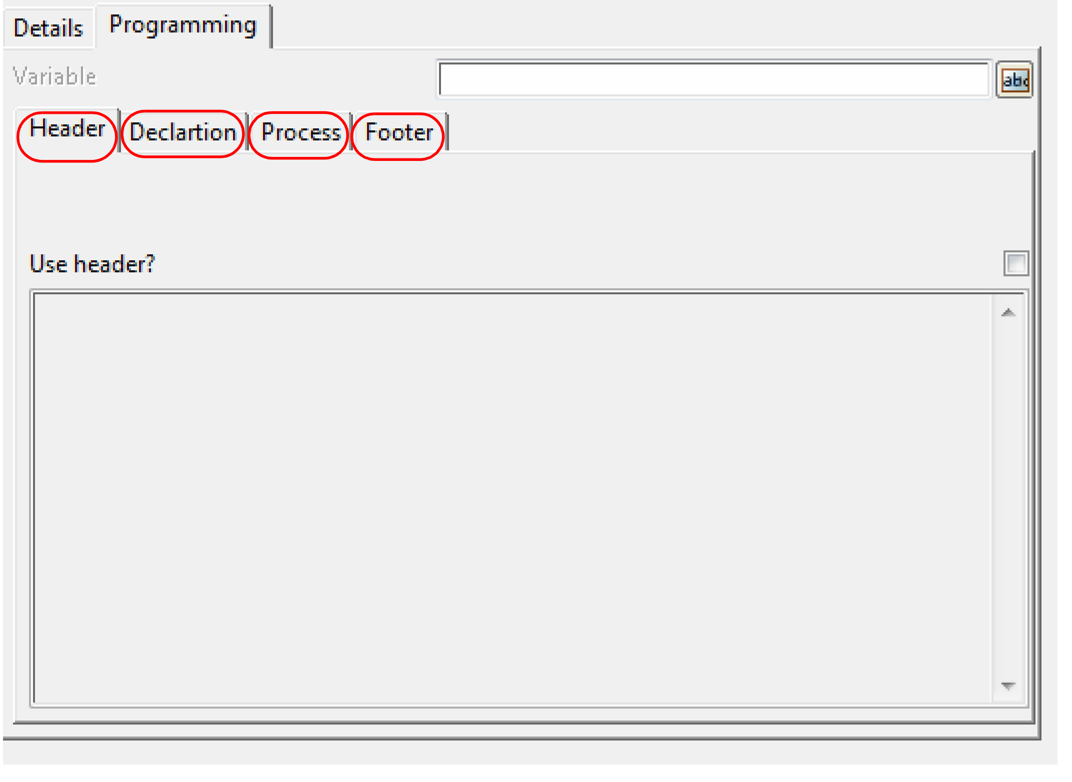
Continue to next section 18.2.5.1.1.7 Programming or return to 18.2 12d Field .

18.2.5.1.1.7 Programming

It is possible to generate macro code for GIS post processing, for any feature or attribute. The **programming tab** is used to assist you in doing so.

See 18.2.5.5 12d Pickup Codes to Macro for more information on how to generate a GIS post processing 4dm file.

To assist you, this is broken into sections: Header Tab , Declaration Tab , Process Tab , Footer Tab .



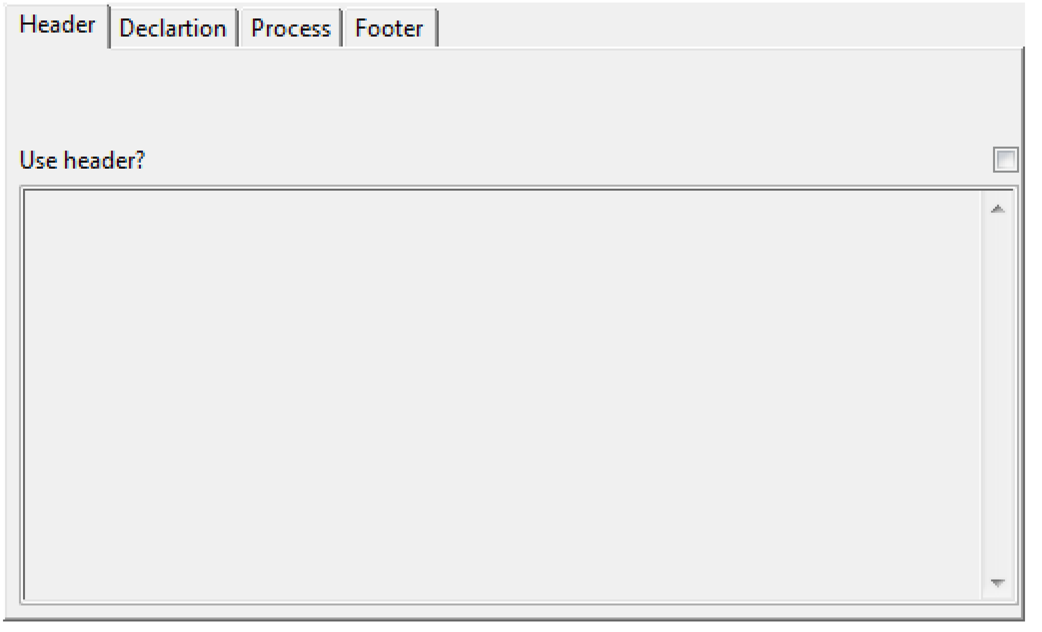
The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Variable	input		

For use with attributes or opcodes, defines the variable that the data should be stored in for use later.

Header Tab

The **Header tab** defines any header that should be output into the macro file for the current item.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use header?	tick box		

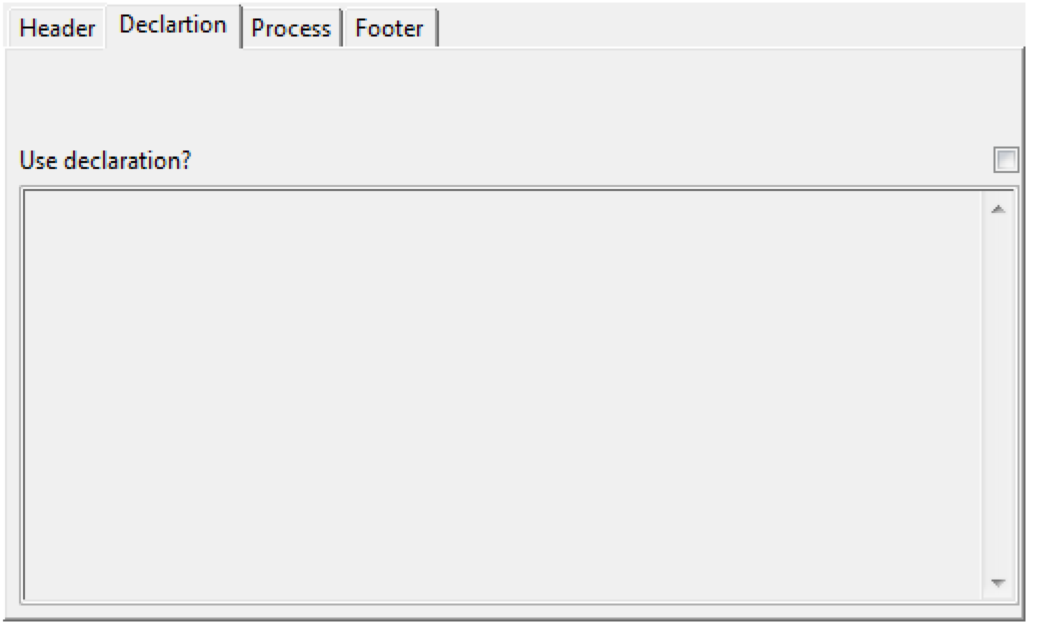
if ticked, the header will output to the generated file
if not ticked, no header will be used

Text field

the macro code to output into the GIS post processing file

Declaration Tab

The **Declaration tab** defines the 'declaration' to be output into the macro, which can be used for defining variables.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use declaration?	tick box		

if ticked, a declaration will be output to the generated file
if not ticked, no declaration will be output

Text field

the macro code to output into the GIS post processing file.

Process Tab

The **Process tab** defines the main processing part of the macro code, which could be used to process a selected feature code and associated attributes, or other items such as opcodes.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use process?	tick box		

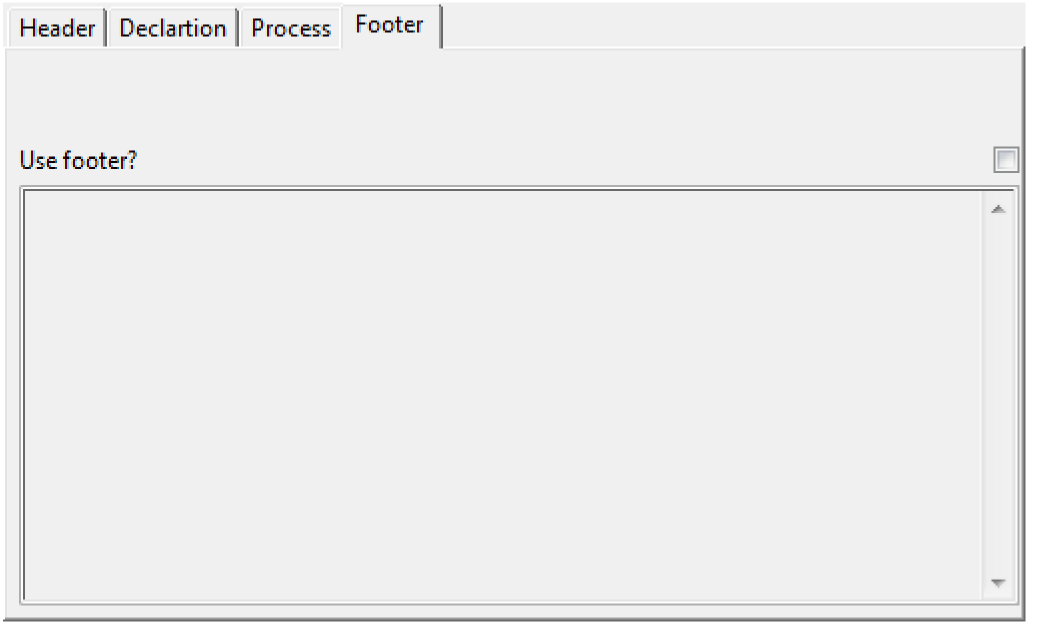
if ticked, the process data will be output into the generated file
if not ticked, no process data will be output.

Text field

the macro code to output into the GIS post processing file

Footer Tab

The **Footer tab** is used to define any macro code that should be added to the footer.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use footer?	tick box		

if ticked, the footer will output to the generated file
if not ticked, no footer will be output

Text field

the macro code to output into the GIS post processing file

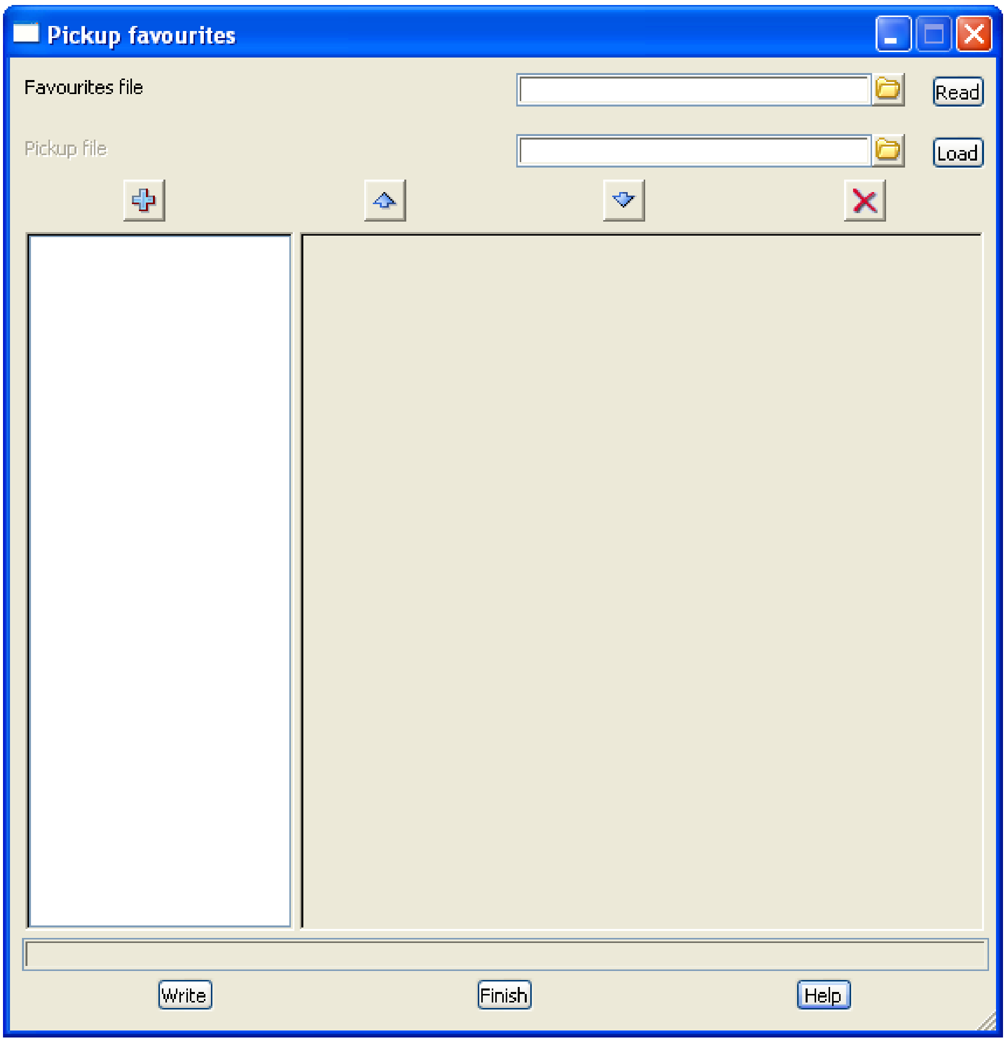
Continue to next section 18.2.5.2 Pickup Favourites or return to 18.2 12d Field .

18.2.5.2 Pickup Favourites

Position of option on menu: Survey =>12d Field =>12d Field Pickup Codes => 12d Pickup Favourites

This panel is used to create and edit 12d field pickup favourites files. They contain a list of feature codes and associated information for use within **12d Field Pickup**.

Selecting **12d Pickup Favourites** brings up the **Pickup Favourites** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Favourites file	file		*.12dfieldfavourites

the favourites file to create or edit

Read	button		

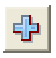
reads the favourites file, if it exists

Pickup file	file		*.12dfieldcodes

the optional 12d field pickup codes file to use as the source of feature codes available as favourites.

Load	button		

loads the optional 12d field pickup codes file for use

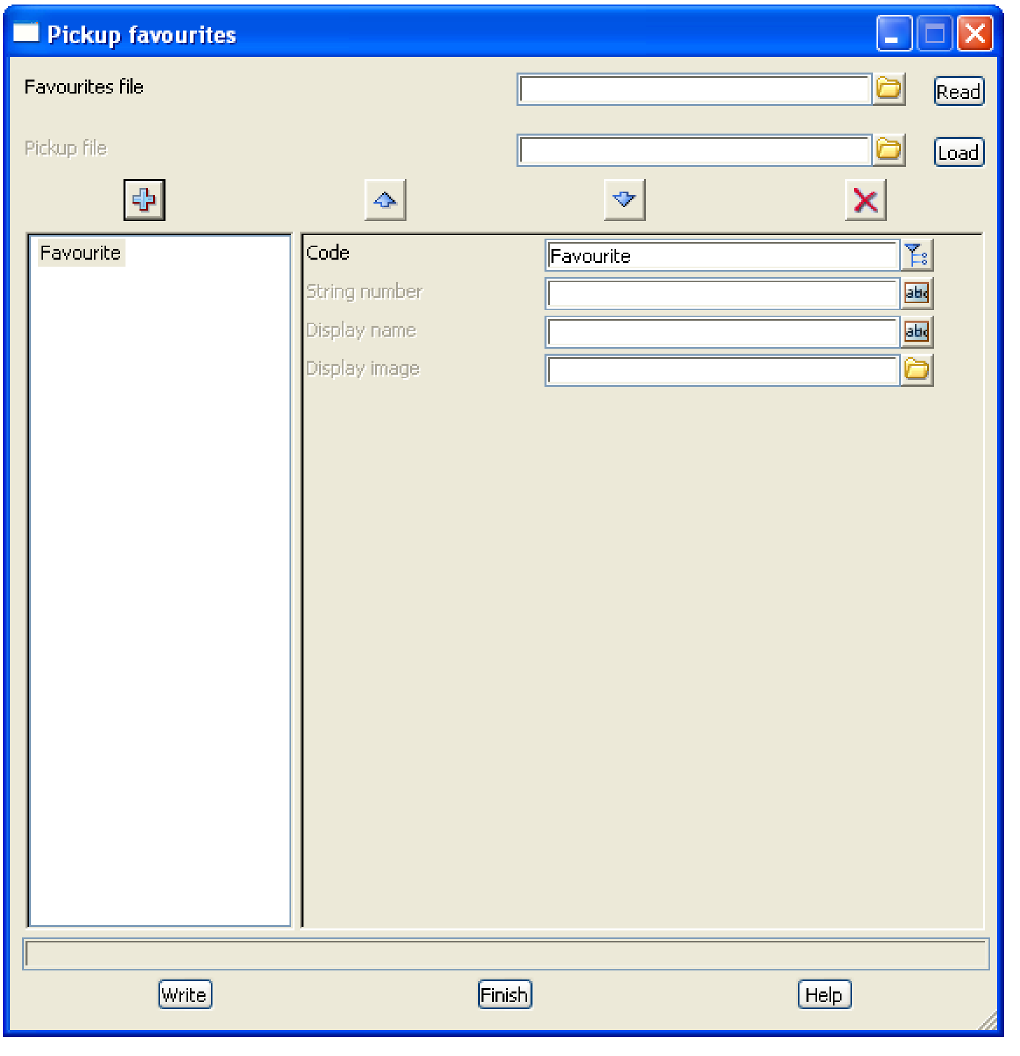


adds a new favourite

moves a favourite up

moves a favourite down

delete a favourite



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Code

the name of the favourite.
if a pickup codes file has been loaded, the list of codes in the pickup codes file will be available as a choice in the browse box.
if no pickup codes file has been loaded, nothing will appear in the browse box.

String number input

the optional string number for the favourite

Display name input

the optional cosmetic name of the favourite to display
If not specified, the favourite will display using the code name

Display image file

the optional cosmetic image of the favourite to display
If not specified, no image will be displayed.

Write button

writes the Pickup favourites to the specified file.

Continue to next section 18.2.5.3 Mapfile to 12d Pickup Codes or return to 18.2 12d Field .

18.2.5.3 Mapfile to 12d Pickup Codes

Position of option on menu: **Survey =>12d Field =>12d Field Pickup Codes => Mapfile to 12d Pickup Codes**

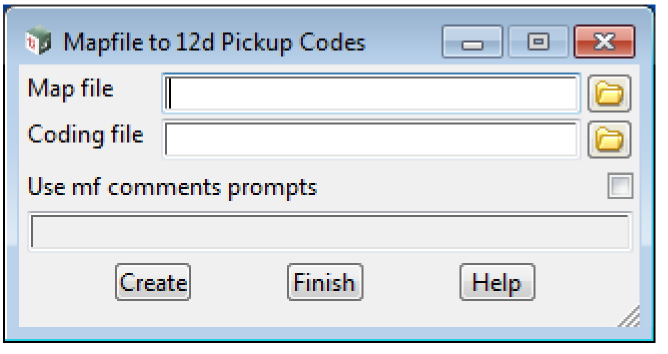
This option is used to create a 12d Field Pickup Coding file from an existing map file. Only the following information from the Basic section of the mapfile is used

Key

Model

Comment

Selecting **Mapfile to 12d Pickup Codes** brings up the **Mapfile to 12d Pickup Codes** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Mapfile	file		*.mapfile *.mf

name of the 12d mapfile to convert from.

Field Description	Type	Defaults	Pop-Up
Coding file	file		*.12dfieldcodes

name of the coding file to convert to.

Use mf comments prompts tick box

if ticked, the Comment field is used as the display prompt that the user sees when selecting the current feature code.
If not ticked, the Key field is used as the display prompt.

Create button

convert the file.

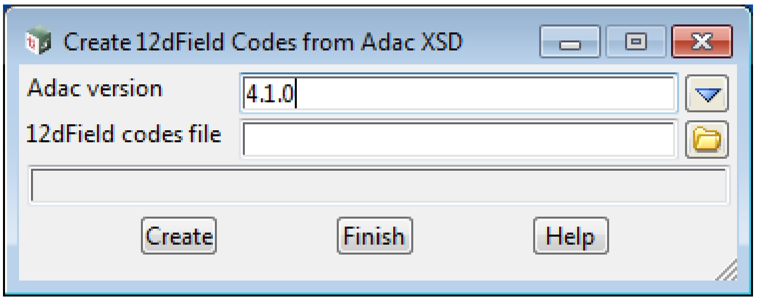
Continue to next section 18.2.5.4 ADAC XSD to 12d Pickup Codes or return to 18.2 12d Field .

18.2.5.4 ADAC XSD to 12d Pickup Codes

Position of option on menu: Survey =>12d Field =>12d Field Pickup Codes => ADAC XSD to 12d Pickup Codes

This section of documentation is a work in progress and will be updated in subsequent releases.

Selecting **ADAC XSD to 12d Pickup Codes** brings up the **Create 12d Field Codes from ADAC XSD** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
ADAC Version	choice box		
the version of ADAC being used			
12dField codes file	file box		
Create	button		

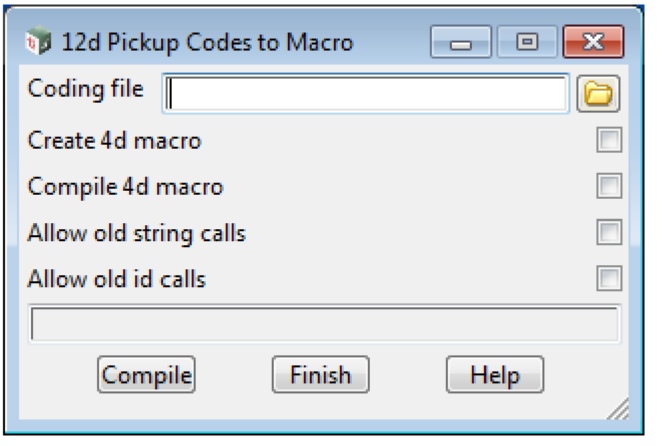
Continue to next section 18.2.5.5 12d Pickup Codes to Macro or return to 18.2 12d Field .

18.2.5.5 12d Pickup Codes to Macro

Position of option on menu: Survey =>12d Field =>12d Field Pickup Codes => 12d Pickup Codes to 4dm

This panel generates a macro based on any programming defined in a 12d Field pickup codes file. It will create a new file of the same name as the 12d Field pickup codes file, with the extension 4dm

Selecting **12d Pickup Codes to 4dm** brings up the **12d Pickup Codes to Macro** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Coding file	file		*.12dfieldcodes

the 12d Field Pickup codes file to read

Create 4d macro tick box

if ticked, it creates the 4dm file from the 12d Field Pickup codes file programming.

Compile 4d macro tick box

if ticked, the created 4dm file will be compiled into a 4do file
if not ticked, no compilation will take place

Allow old string calls tick box

if ticked, old string calls will be allowed by the compiler.
if not ticked, old string calls will not be allowed by the compiler.

Allow old id calls tick box

if ticked, old id calls will be allowed by the compiler.
if not ticked, old id calls will not be allowed by the compiler.

Compile button

creates and/or compiles the 12d Field Pickup codes file into a macro

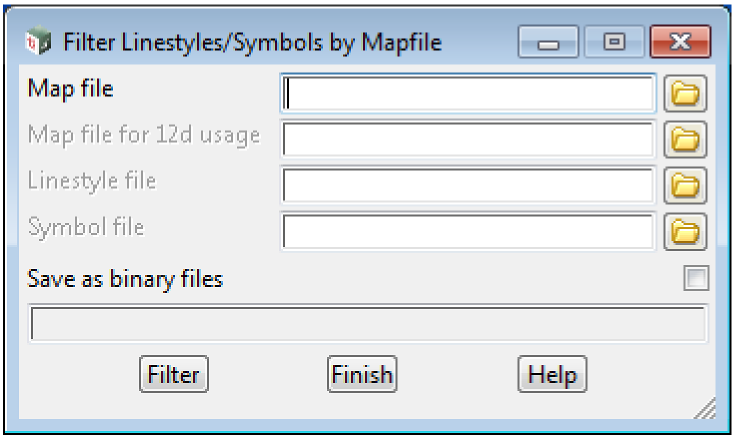
Continue to next section 18.2.5.6 Filter Linestyles/Symbols via Mapfile or return to 18.2 12d Field .

18.2.5.6 Filter Linestyles/Symbols via Mapfile

Position of option on menu: Survey =>12d Field =>12d Field Pickup Codes => Filter linestyles/symbols via mapfile

This section of documentation is a work in progress and will be updated in subsequent releases.

Selecting **Filter linestyles/symbols via mapfile** brings up the **Filter Linestyles/Symbols by Mapfile** panel



The fields and buttons used in this panel have the following functions.

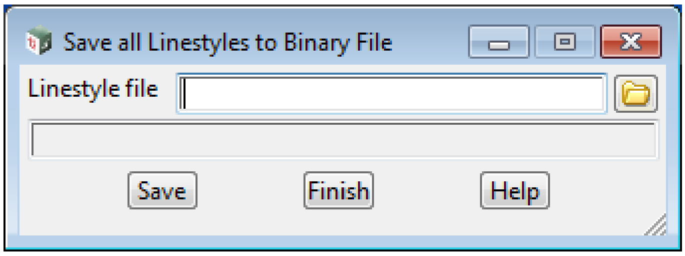
Field Description	Type	Defaults	Pop-Up
Map file	file box		available map files
Map file for 12d usage	file box		available map files
Linestyle file	file box		
Symbol file	file box		
Save as binary files	tick box	not ticked	
Filter	button		

Continue to next section 18.2.5.7 Save Binary Linestyles or return to 18.2 12d Field .

18.2.5.7 Save Binary Linestyles

Position of option on menu: Survey =>12d Field =>12d Field Pickup Codes => Save binary linestyles

Selecting **Save binary linestyles** brings up the **Save all Linestyles to Binary File** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Linestyle file	file box		available *.4d files

the name of the binary form of the linestyle file

Save	button		

saves the linestyle data as a binary file.

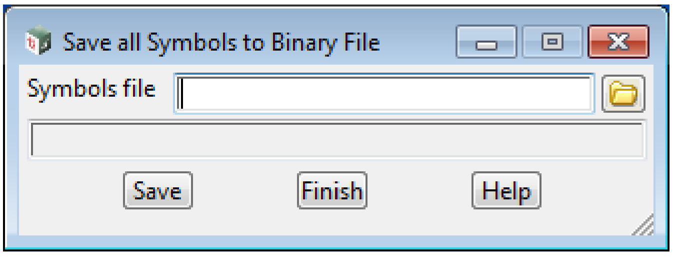
Continue to next section 18.2.5.8 Save Binary Symbols or return to 18.2 12d Field .

18.2.5.8 Save Binary Symbols

Position of option on menu: Survey =>12d Field =>12d Field Pickup Codes => Save binary symbols

This section of documentation is a work in progress and will be updated in subsequent releases.

Selecting **Save binary symbols** brings up the **Save all Symbols to Binary File** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Symbols file	file box		available *.4d files
Save	button		

Return to 18.2 12d Field .

19 ePlan

The **ePlan XML Schema** is a vendor independent XML format introduced by the State Governments in Australia for the submission of Cadastral plans.

So far it has been implemented by NSW, Qld and Victoria with the other States planning to implment it in the fuutre.

Although there are similarities between the States, there are different enumeration files for each State in the ePlan XML Schema. Also each State has its own business rules, and own methods for submitting the ePlan.XML data.

Also althought the various ePlan XML’s are based on Land.XML, what is a valid Land.XML file may not be a valid ePlan.XML for a particular State.

See

19.1 ePlan.XML Overview
19.2 12d Approach to ePlan XML
19.3 12d ePlan Workflow
19.4 12d ePlan Menu

19.1 ePlan.XML Overview

See

19.1.1 ePlan XML Structure

19.1.1 ePlan XML Structure

The ePlan Schema is fully described by the *ePlan XSD* (**X**ML **S**chema **D**efinition), which formally defines the structure and all the elements inside an ePlan XML file. That means that any ePlan.XML can be **validated** against the ePlan Schema XSD to see that it conforms to the Schema.

Return to 19.1 ePlan.XML Overview or 19 ePlan .

19.2 12d Approach to ePlan XML

Creating an ePlan XML file could be approached as a spreadsheet type exercise where you type the values into a document, but this is little more than a manual data collection exercise.

Continue to the next section 19.3 12d ePlan Workflow or return to 19 ePlan .

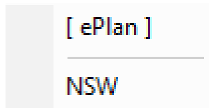
19.3 12d ePlan Workflow

In ***12d Model***, the **source** of the data to write out to an ePlan XML is usually from either

Continue to the next section 19.4 12d ePlan Menu or return to 19 ePlan .

19.4 12d ePlan Menu

Position on menu: **Survey =>Network =>ePlan**



See

19.4.1 Create NSW ePlan Network

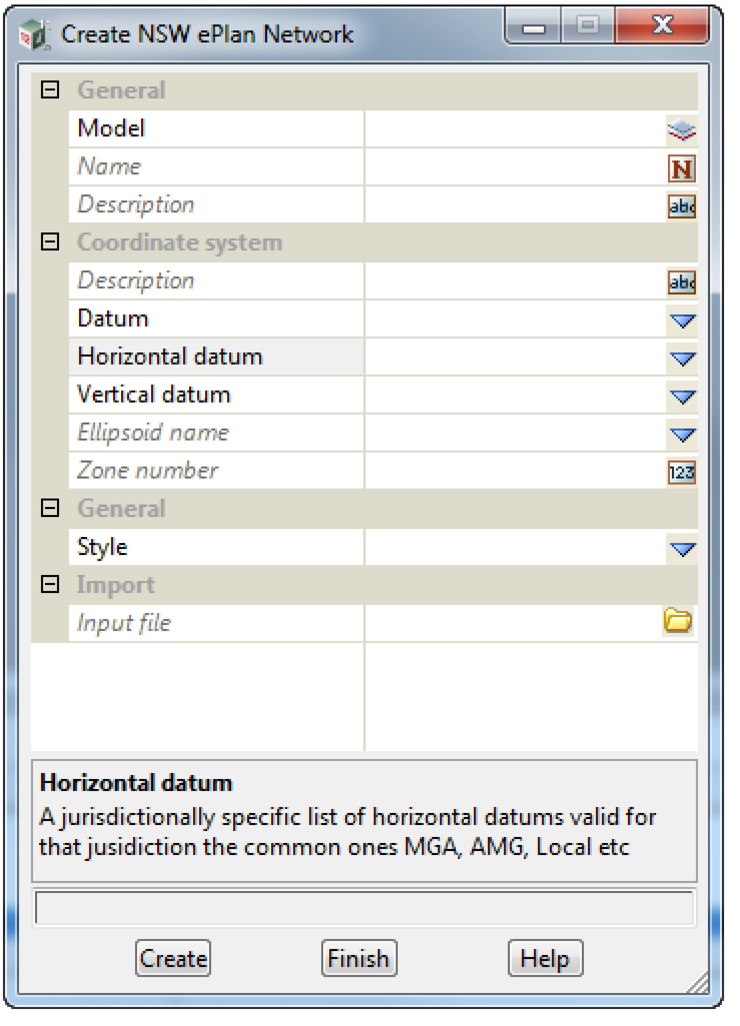
19.4.1 Create NSW ePlan Network

Position of option on menu: **Survey =>Network =>ePlan =>NSW**

To create an ePlan for NSW, an ePlan network needs to be created.

The data to go out as an NSW ePlan.XML file are then added to the ePlan network.

Selecting **NSW** brings up the **Create NSW ePlan Network** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

General node

Model — model box — Pop-Up: available models

the model for the ePlan network.

Name — name box

the name for the ePlan network.

Description — text box

a description for the ePlan network.

Coordindate System node

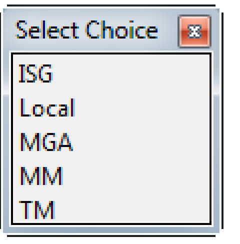
Description — text box

a description for the coordinate system.

Vertical datum choice box AHD

the vertical datum.

Horizontal datum choice box



the horizontal datum.

Ellipsoid name choice box

name of the ellipsoid used (if any).

Zone number text box

zone number if MGA).

General node

Style Network Style box available network styles

the style used for desplaying the ePlan network.

Buttons at bottom

Create button

create an ePlan Network for NSW.

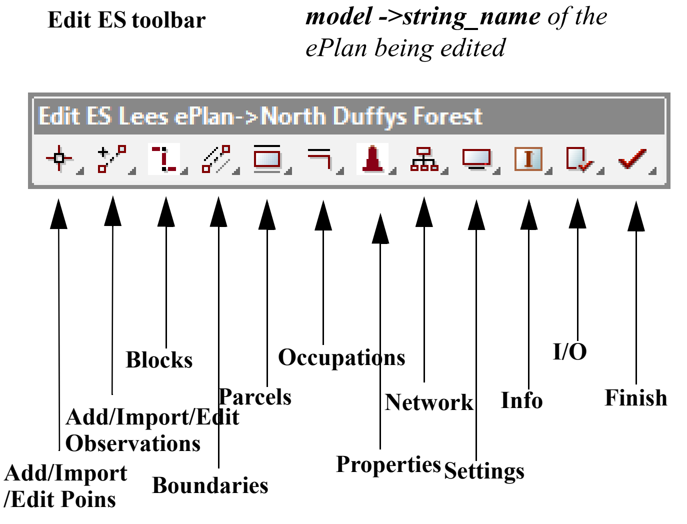
After the **Create** button is pressed, a new NSW ePlan is created and the **Edit ES** toolbar appears. For information on the **Edit ES** toolbar, go to 19.4.2 Edit ES for NSW ePlan Network.

Continue to the next section or return to 19.4 12d ePlan Menu or 19 ePlan.

19.4.2 Edit ES for NSW ePlan Network

Position of option on menu: **Survey =>Network =>ePlan =>Create NSW ePlan Network**

If the Strings Edit is used to seelct an ePlan network, or after the **Create** button is pressed on the **Create NSW ePlan Network** panel and a new NSW ePlan is created, the **Edit ES** toolbar appears to edit the EPlan object.



See

19.4.2.1 Points
19.4.2.2 Observations
19.4.2.3 Blocks
19.4.2.4 Boundaries
19.4.2.5 Parcels
19.4.2.6 Occupations
19.4.2.7 Properties
19.4.2.8 Network
19.4.2.9 Settings
19.4.2.10 Info
19.4.2.11 I/O

19.4.2.1 Points

See

19.4.2.1.1 Adding a Point

19.4.2.1.2 Importing Points

19.4.2.1.3 Editing Points

19.4.2.1.4 Editing CG Points

19.4.2.1.5 Editing Monuments

19.4.2.1.1 Adding a Point

The **Add a Point** option adds a user selected point to the ePlan network.

Selecting **Add a Point** options writes the message to the screen message area

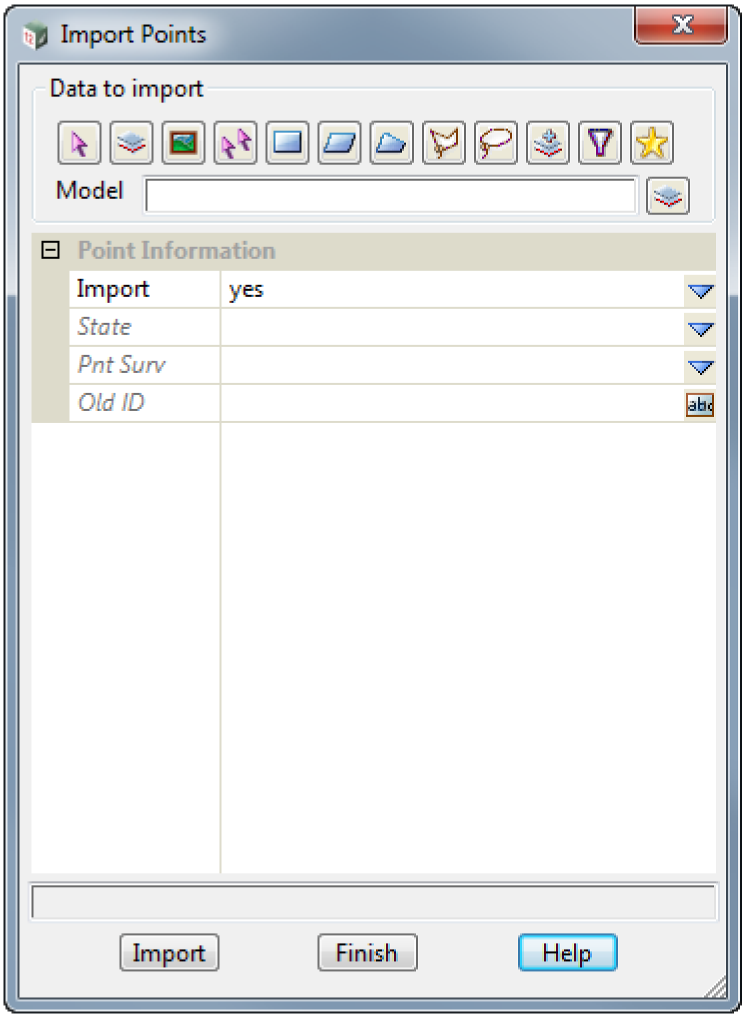
<Pick point or (n)ame, (d)escription> [picks][][Menu]

and a point is selected and if it is outside the tolerance distance of all existing points in the Network, it is added to the network and given the next available **Point id**.

19.4.2.1.2 Importing Points

The **Import Points** option adds a Data Source of user selected points to the ePlan network.

Selecting **Import points** brings up the **Import Points** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source .

Data source

data source for vertices to add to the Network.

Point Information

Import	choice box		yes, no

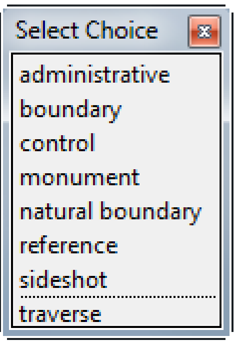
*if **yes**, the selected data is procssed for adding to the network.*

*If **no**.*

State	choice box		existing, proposed

whether the selected vertices are existing survey points or proposed points.

Pnt Surv choice box



the type of the selected vertices.

Old ID input

the Point Id that the CG point used to have. This can be modified.

Import button

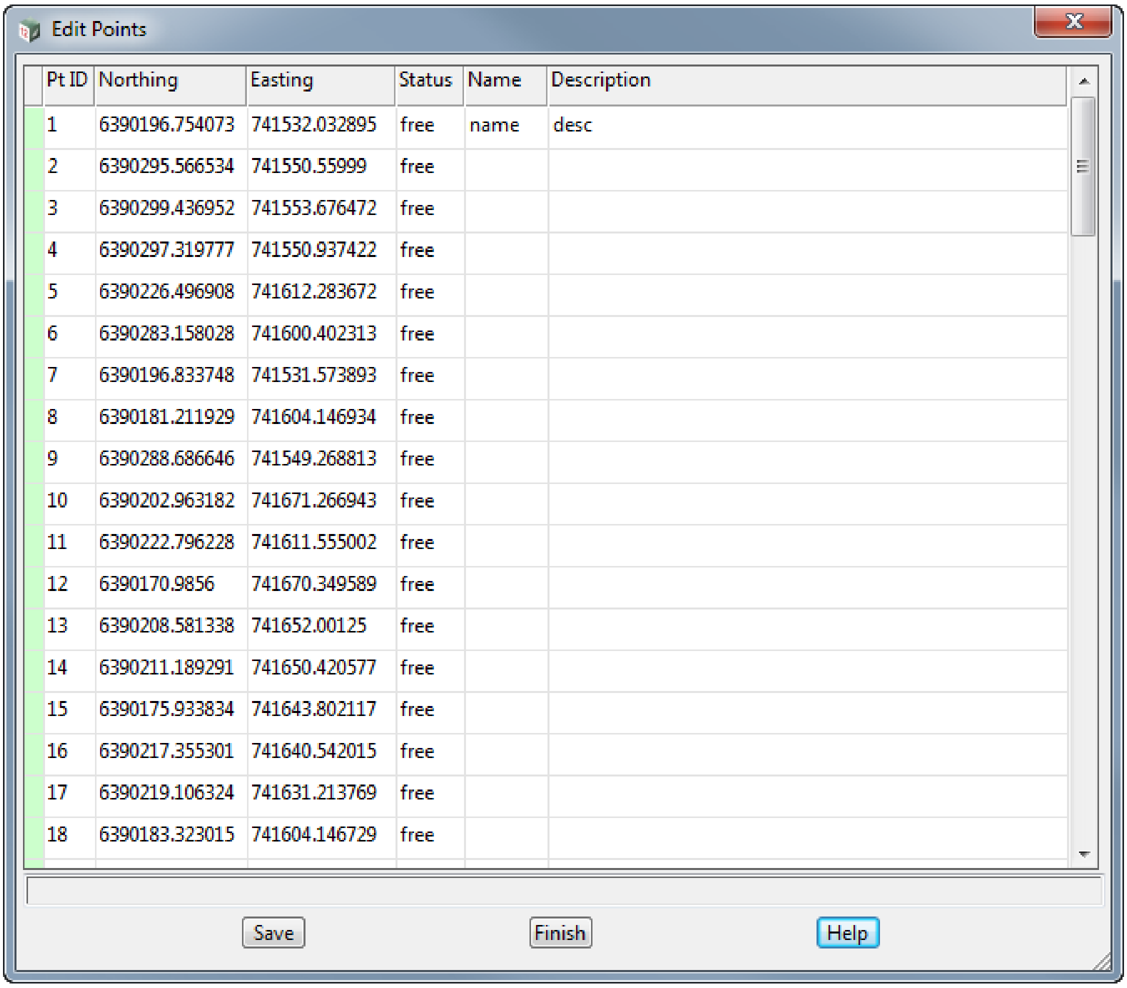
when pressed the selected vertices will be processed. Each vertex is checked in turn and if the vertex is outside the tolerance distance of all the existing point in the Network, it is added to the network and given the next available Point id.

19.4.2.1.3 Editing Points

Selecting **Edit points** brings up the **Edit Points** panel which displays in tabular form all the existing points in the Network.

For each Point it displays the Point ID, Northing, Easting, Status, Name and Description.

Clicking on a line in the grid will highlight and autopan to the point for that line on any plan views that has the Network on them.



Pt ID	Northing	Easting	Status	Name	Description
1	6390196.754073	741532.032895	free	name	desc
2	6390295.566534	741550.55999	free		
3	6390299.436952	741553.676472	free		
4	6390297.319777	741550.937422	free		
5	6390226.496908	741612.283672	free		
6	6390283.158028	741600.402313	free		
7	6390196.833748	741531.573893	free		
8	6390181.211929	741604.146934	free		
9	6390288.686646	741549.268813	free		
10	6390202.963182	741671.266943	free		
11	6390222.796228	741611.555002	free		
12	6390170.9856	741670.349589	free		
13	6390208.581338	741652.00125	free		
14	6390211.189291	741650.420577	free		
15	6390175.933834	741643.802117	free		
16	6390217.355301	741640.542015	free		
17	6390219.106324	741631.213769	free		
18	6390183.323015	741604.146729	free		

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Pt ID display only

unique Point id of the poiont in the Network. This can not be modified.

Northing real box

the Northing of the point with the Point id. This can be edited.

Easting choice box fixed, free

the Easting of the point with the Point id. This can be edited.

Status

the Status is used is a Least Squares Adjustment (LSA).

*If **fixed**, the coordinates of the point will be held constant in a LSA.*

*If **free** then the coordinates of the point will be held modified in a LSA.*

Name

the name of the point. This can be edited.

Description

the description of the point. This can be edited.

Save button

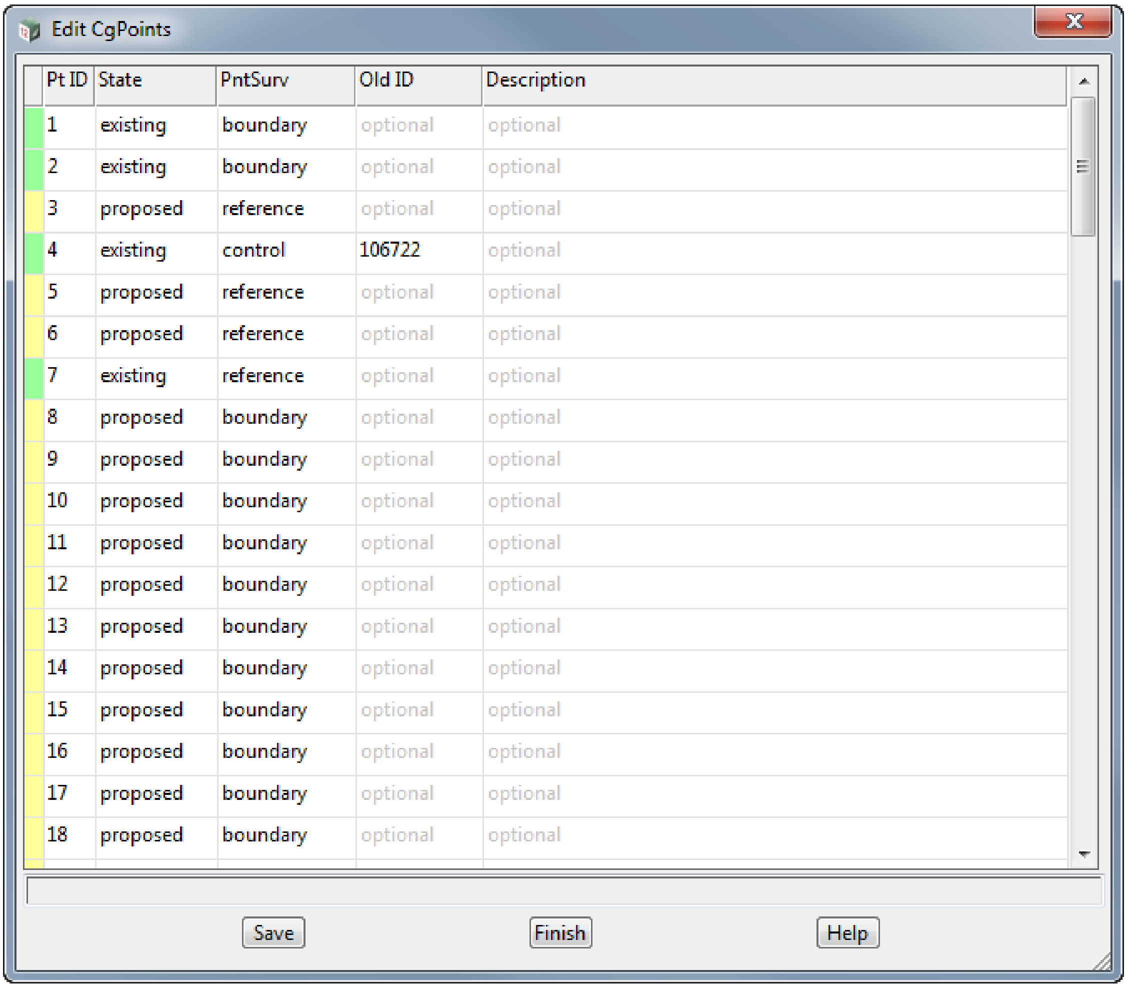
save the data in the panel to the Network object.

19.4.2.1.4 Editing CG Points

Selecting **Edit cgPoints** brings up the **Edit CgPoints** panel which displays in tabular form all the existing CG points in the Network.

For each CG point it displays the Point ID, State, PntSurv, Old ID and Description.

Clicking on a line in the grid will highlight and autopan to the CG point for that line on any plan views that has the Network on them.



Pt ID	State	PntSurv	Old ID	Description
1	existing	boundary	optional	optional
2	existing	boundary	optional	optional
3	proposed	reference	optional	optional
4	existing	control	106722	optional
5	proposed	reference	optional	optional
6	proposed	reference	optional	optional
7	existing	reference	optional	optional
8	proposed	boundary	optional	optional
9	proposed	boundary	optional	optional
10	proposed	boundary	optional	optional
11	proposed	boundary	optional	optional
12	proposed	boundary	optional	optional
13	proposed	boundary	optional	optional
14	proposed	boundary	optional	optional
15	proposed	boundary	optional	optional
16	proposed	boundary	optional	optional
17	proposed	boundary	optional	optional
18	proposed	boundary	optional	optional

The fields and buttons used in this panel have the following functions.

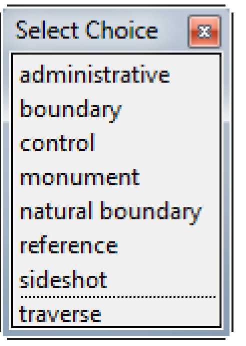
Field Description	Type	Defaults	Pop-Up
Pt ID	display only		

unique Point ID of the CG point in the Network. This can be not be modified

State	choice box		existing, proposed

whether the selected CG point are existing survey points or proposed points.

Pnt Surv choice box



the type of the selected CG point. This can be modified.

Old ID input

the Point Id that the CG point used to have. This can be modified.

Description

the description of the CG point. This can be edited.

Save button

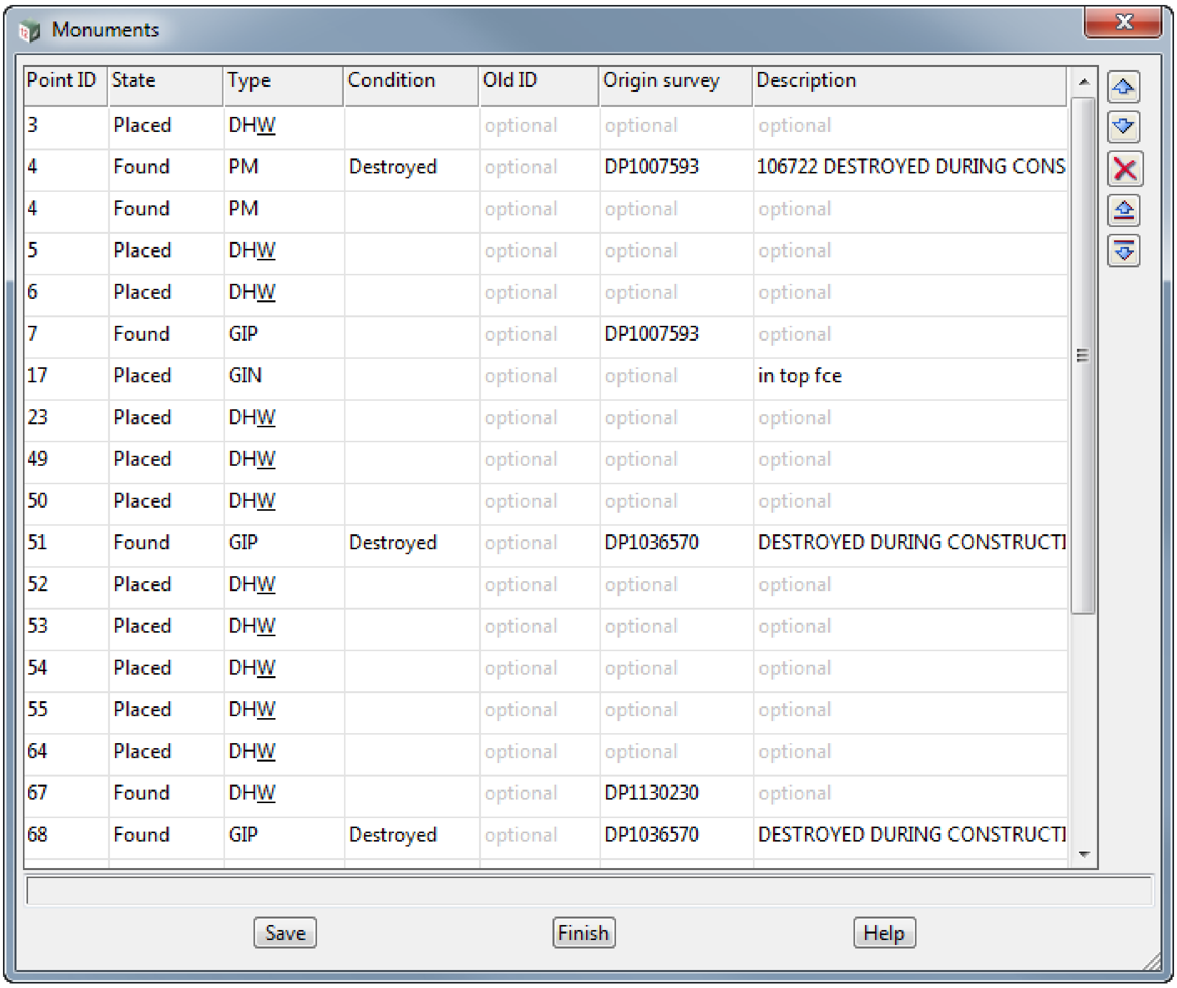
save the data in the panel to the Network object.

19.4.2.1.5 Editing Monuments

Selecting **Edit monuments** brings up the **Monuments** panel which displays in tabular form all the existing Monument points in the Network.

For each Monument Point it displays its Point ID, Type, Condition, Old ID, Origin Survey and Description.

Clicking on a line in the grid will highlight and autopan to the Monument point for that line on any plan views that has the Network on them.



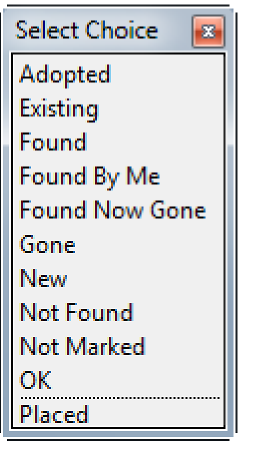
The fields and buttons used in this panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Pt ID display only

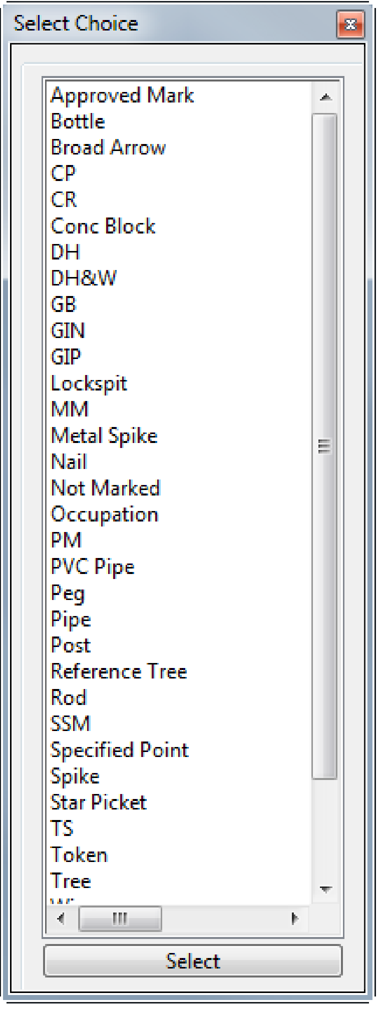
unique Point id of the Monument point in the Network. This can be not be modified.

State choice box



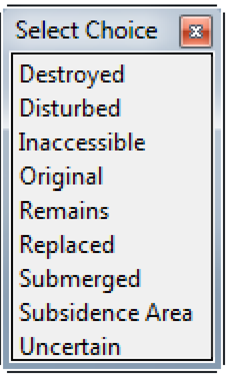
the State of the selected Monument point. This can be modified.

Type choice box



the Type of the selected Monument point. This can be modified.

Condition choice box



the Condition of the selected Monument point (which may be left blank). This can be modified.

Old ID input

the Point Id that the Monument point used to have. This can be modified.

Origin Survey

the Origin Survey for the Monument point. This can be edited.

Description

the description of the Monument point. This can be edited.

Save button

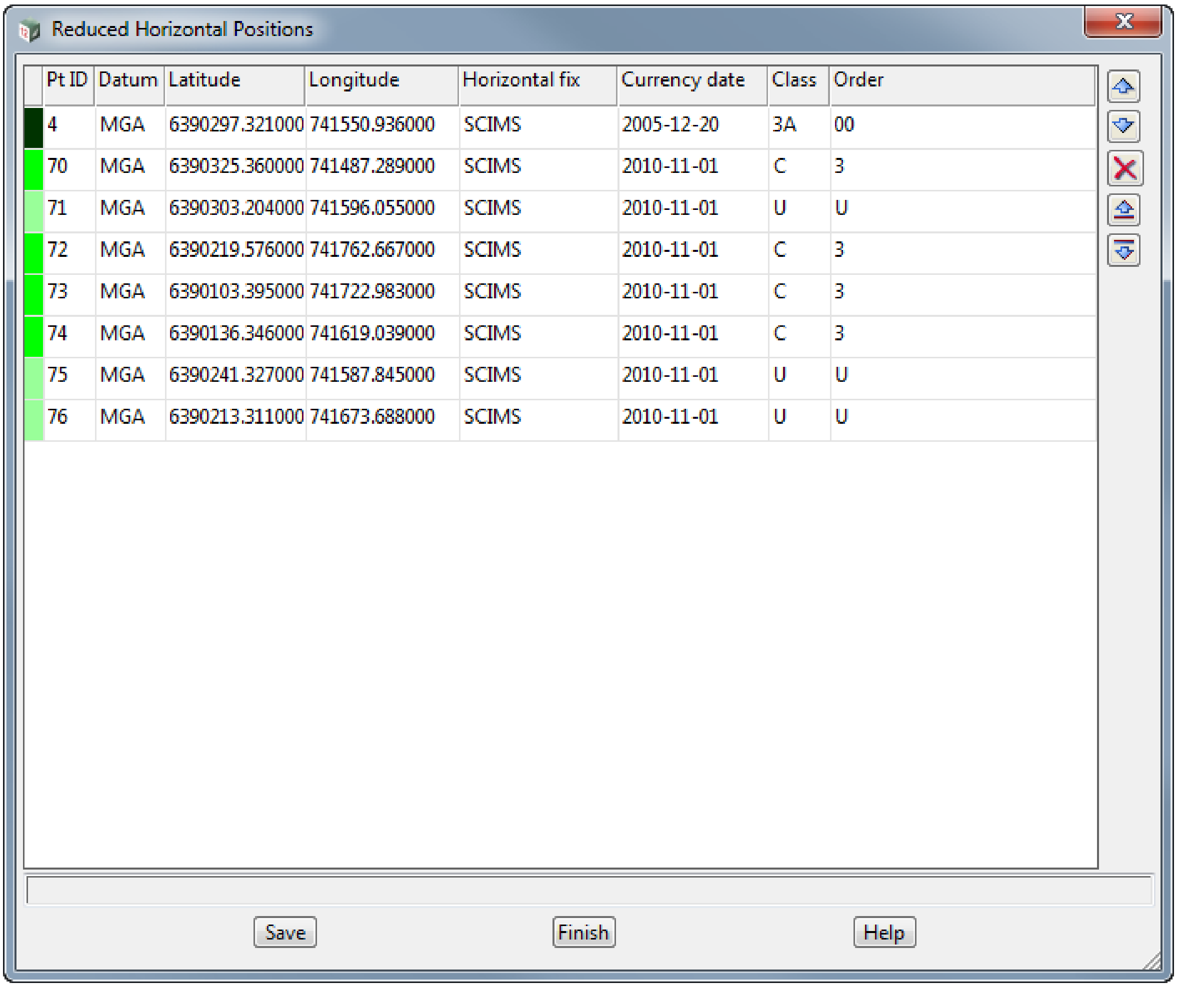
save the data in the panel to the Network object.

19.4.2.1.6 Editing Reduced Horizontal Positions

Selecting **Edit reduced horizontal positions** brings up the **Reduced Horizontal Positions** panel which displays in tabular form all the CG Points in the Network that have **PntSurv** equal to **Control** (the Control points).

For each Control Point it displays the Datum, Latitude (actually Northing) Longitude (actually Easting), Horizontal fix, Currency Date, Class and Order

Clicking on a line in the grid will highlight and autopan to the Control Point for that line on any plan views that has the Network on them.

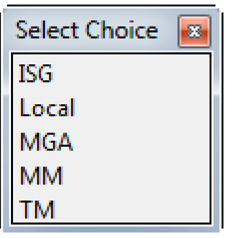


The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pt ID	display only		

unique Point id of the Monument point in the Network. This can be not be modified.

Datum choice box



the horizontal datum that the coordinates for the Control Point is in.

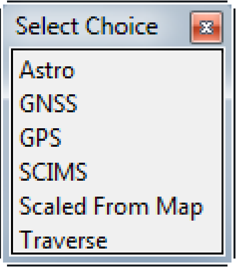
Latitude

the Northing for the Control Point (in the XML file is is stored in the Latitude field).

Longitude

the Easting for the Control Point (in the XML file is is stored in the Longitude field).

Horizontal fix choice box

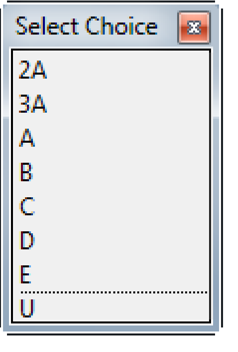


the horizontal fix for the Control Point.

Currency date

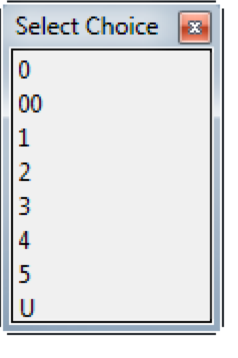
the date that the Control Point coordinates are current.

Class choice box



the class of the Control Point.

Order choice box



the order of the Control Point.

Save button

save the data in the panel to the Network object.

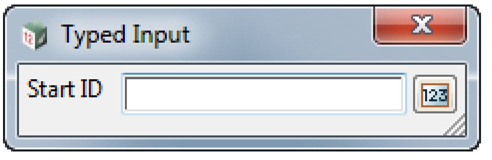
19.4.2.2 Observations

See

19.4.2.2.1 Adding an Observation
19.4.2.2.2 Importing Observations
19.4.2.2.3 Editing Observations

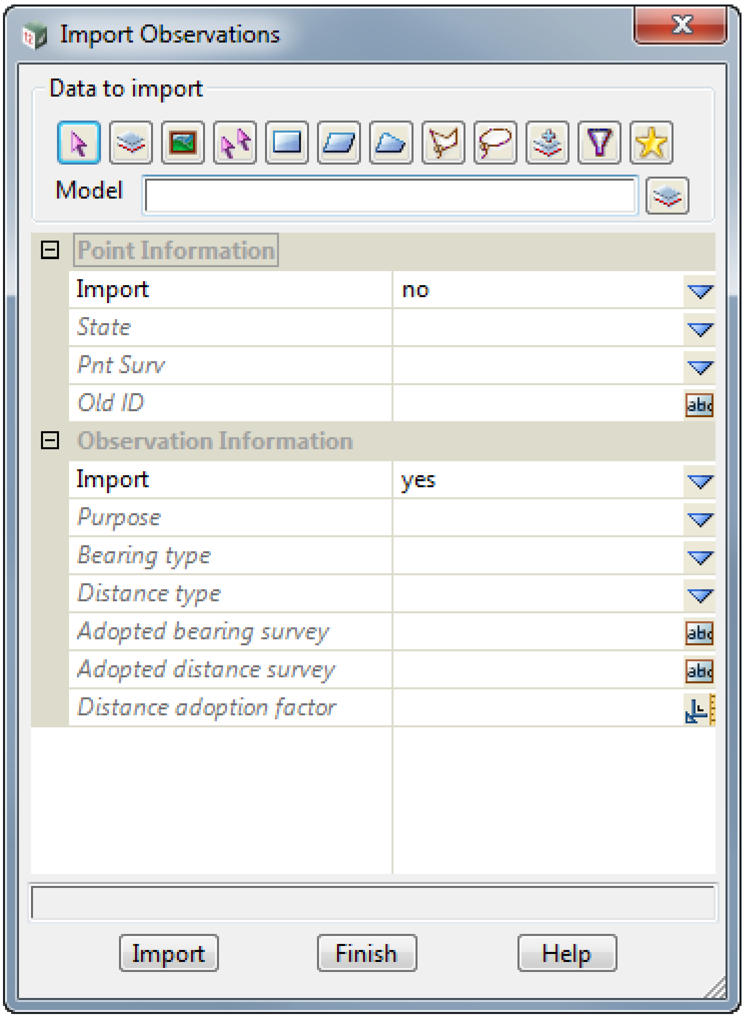
19.4.2.2.1 Adding an Observation

Type a Start ID to set the observation to add.



19.4.2.2.2 Importing Observations

Selecting **Import observations** brings up the **Import Observations** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source .

Data source

data source for segments to add to the Network as observations.

Point Information

Import	choice box		yes, no

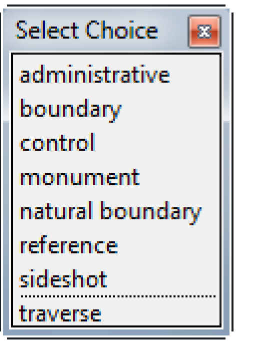
*if **yes,** the selected data is procssed for adding to the network.*

*If **no**.*

State	choice box		existing, proposed

whether the selected observations (segments) are existing survey observations or proposed observations.

Pnt Surv choice box



the type of the selected observations.

Old ID input

the Observation Id that the observation used to have.

Observation Information

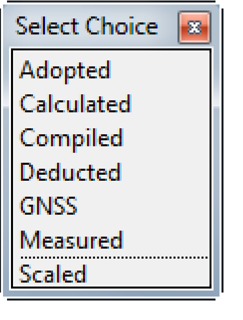
Import choice box yes, no

*if **yes,** the selected data is procssed for adding to the network.*

*If **no**.*

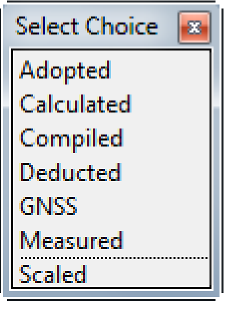
Purpose choice box Boundary, Connection, Road

Bearing type choice box



how the bearing was obtained.

Distance type choice box



how the distance was obtained.

Adopted bearing survey text box

Adopted distance survey text box

Distance adoption factor measure box

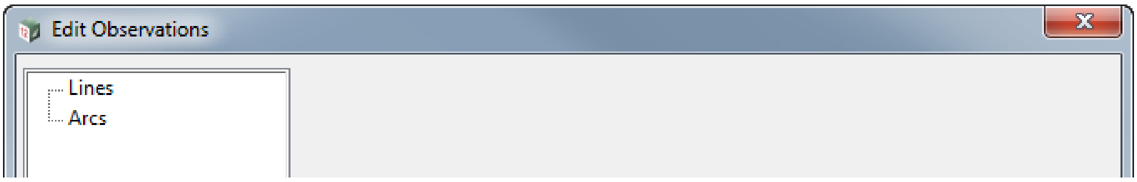
Import button

when pressed the selected segments will be processed. Each segment is checked in turn and if the vertex is outside the tolerance distance of all the existing point in the Network, it is added to the network and given the next available Point id.

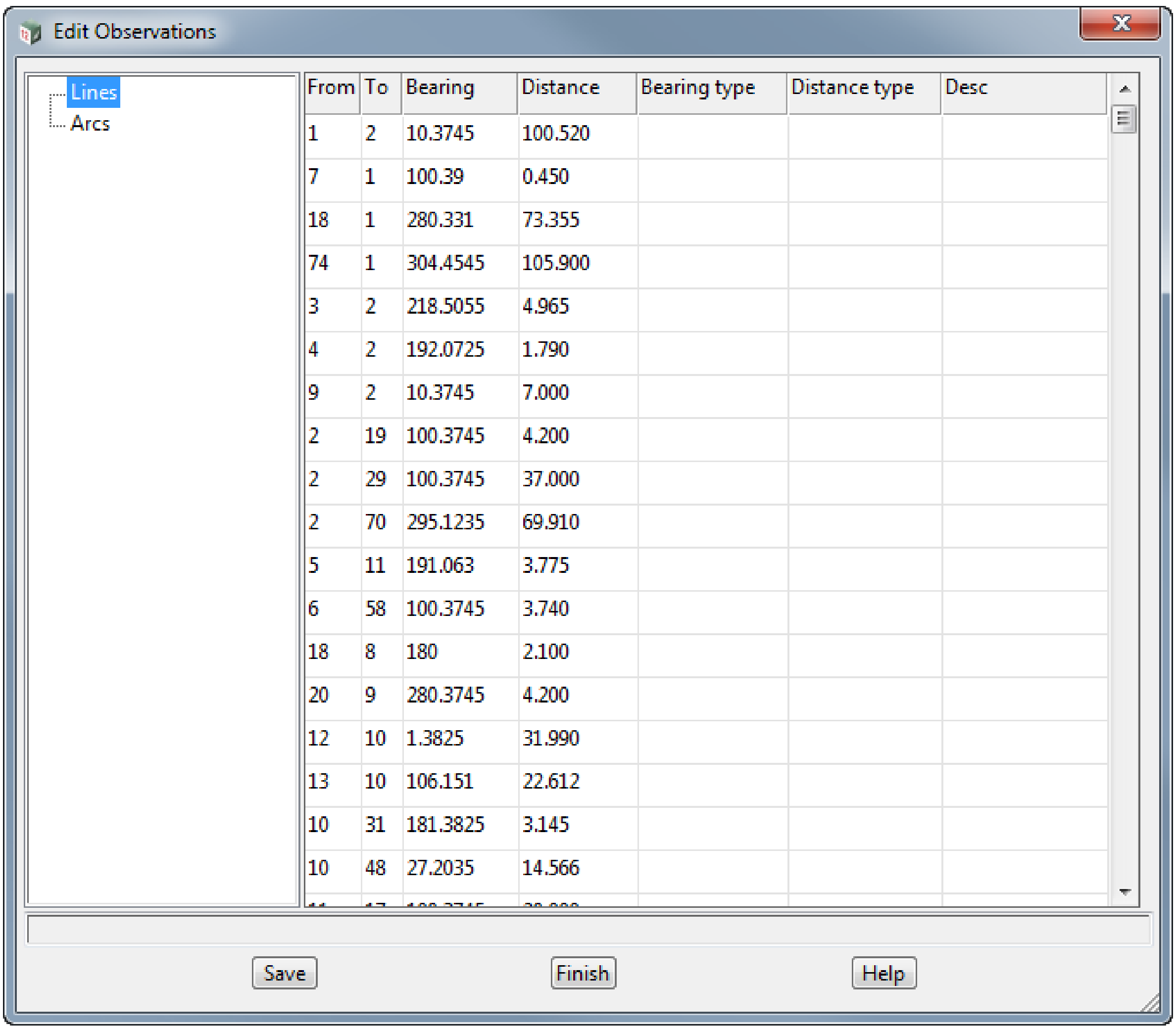
19.4.2.2.3 Editing Observations

Selecting **Edit observations** brings up the **Edit Observations** panel.

Selecting **Edit points** brings up the **Edit Points** panel which displays in tabular form all the existing Lines and Arcs in the Network.



Clicking on the **Lines** node brings up the table of information for each observation of a line.

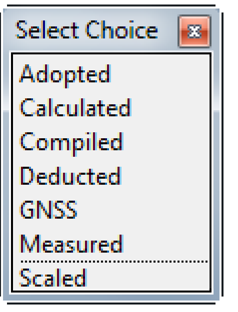


From	To	Bearing	Distance	Bearing type	Distance type	Desc
1	2	10.3745	100.520			
7	1	100.39	0.450			
18	1	280.331	73.355			
74	1	304.4545	105.900			
3	2	218.5055	4.965			
4	2	192.0725	1.790			
9	2	10.3745	7.000			
2	19	100.3745	4.200			
2	29	100.3745	37.000			
2	70	295.1235	69.910			
5	11	191.063	3.775			
6	58	100.3745	3.740			
18	8	180	2.100			
20	9	280.3745	4.200			
12	10	1.3825	31.990			
13	10	106.151	22.612			
10	31	181.3825	3.145			
10	48	27.2035	14.566			

For each Line it displays the **From Point**, **To Point**, **Bearing**, **Distance** for the observation.

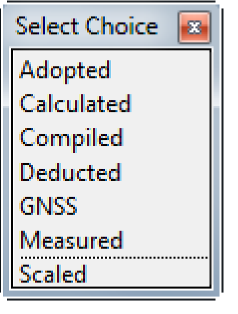
And also the **Bearing Type**, **Distance Type** and **Description** where

Bearing type choice box



how the bearing was obtained.

Distance type choice box



how the distance was obtained.

Desc choice box Boudary, Connection, Road

the description of what the bearing represents.

Save button

save the data in the panel to the Network object.

19.4.2.3 Blocks

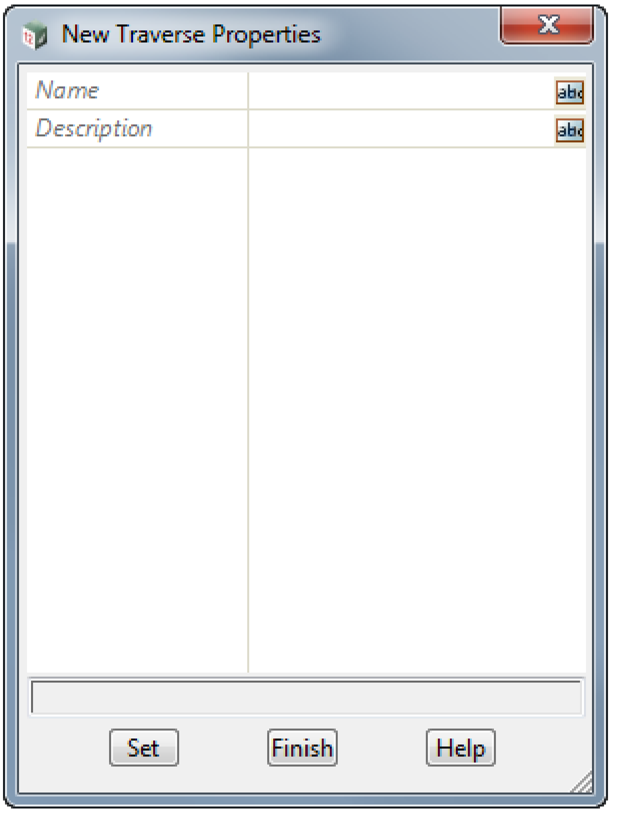
See

19.4.2.3.1 Create Traverse from Points

19.4.2.3.2 Create Radiation from Points

19.4.2.3.1 Create Traverse from Points

Selecting **Create traverse from points** brings up the **New Traverse Properties** panel.

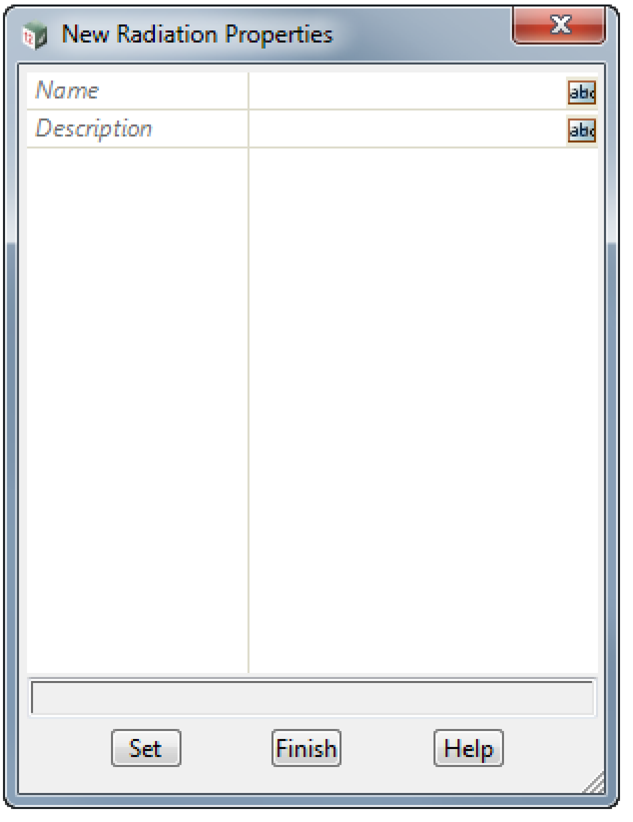


The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input		
Description	input		
Set	button		

19.4.2.3.2 Create Radiation from Points

Selecting **Create radiation from points** brings up the **New Radiation Properties** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input		
Description	input		
Set	button		

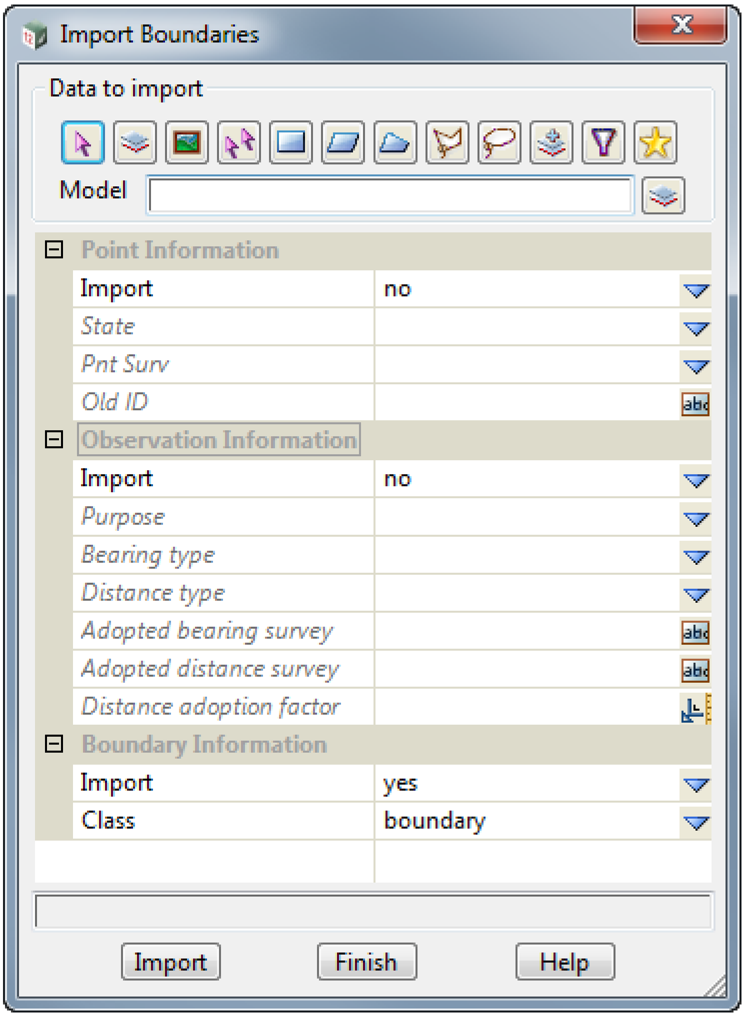
19.4.2.4 Boundaries

See

19.4.2.4.1 Importing Boundaries
19.4.2.4.2 Creating a Boundary
19.4.2.4.3 Creating Multiple Boundaries

19.4.2.4.1 Importing Boundaries

Selecting **Import boundaries** brings up the **Import Boundaries** panel.



The fields and buttons used in this panel have the following functions.

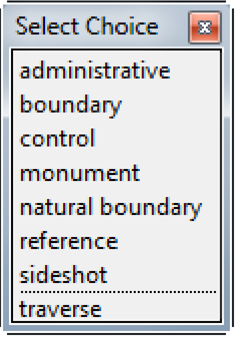
Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source .

Point Information

Import	choice box		yes, no

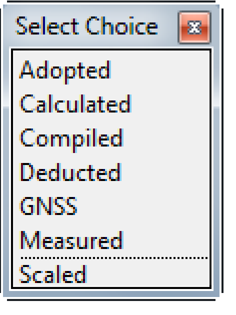
State	choice box	existing, proposed
Pnt Surv	choice box	



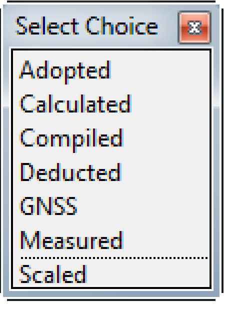
Old ID	input

Observation Information

Import	choice box	yes, no
Purpose	choice box	Boundary, Connection, Road
Bearing type	choice box	



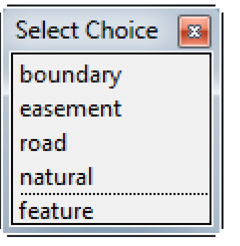
Distance type	choice box



Adopted bearing survey	input
Adopted distance survey	input
Distance adoption factor	measure box

Boundary Information

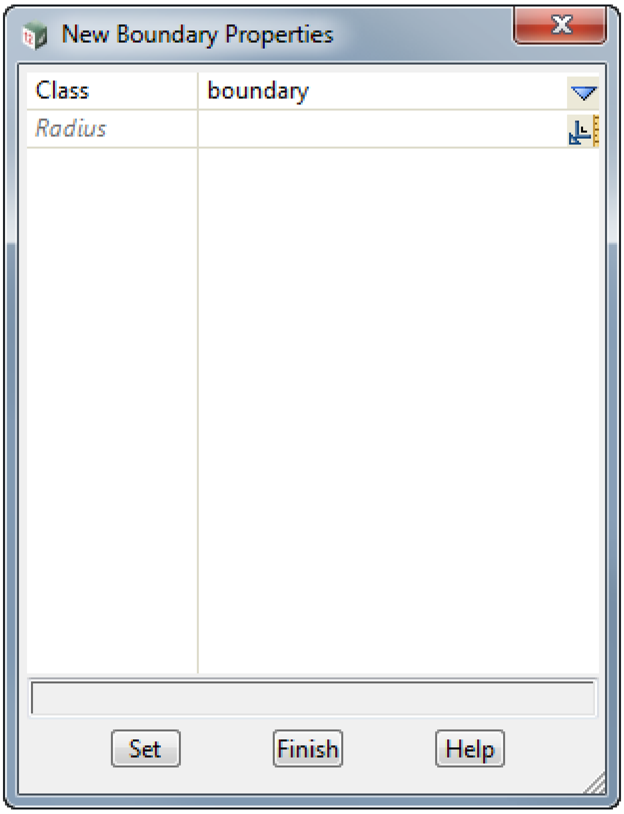
Import	choice box	yes, no
Class	choice box	



Import	button

19.4.2.4.2 Creating a Boundary

Selecting **Create boundary** brings up the **New Boundary Properties** panel.

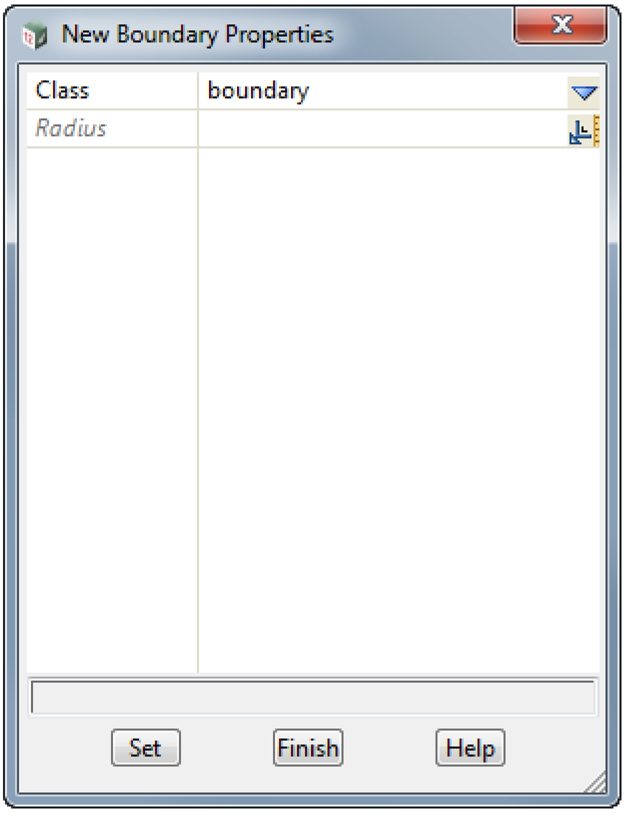


The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Class	choice box		
the			
Radius	measure box		
Set	button		

19.4.2.4.3 Creating Multiple Boundaries

Selecting **Create boundaries** brings up the **New Boundary Properties** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Class	choice box		
the			
Radius	measure box		
Set	button		

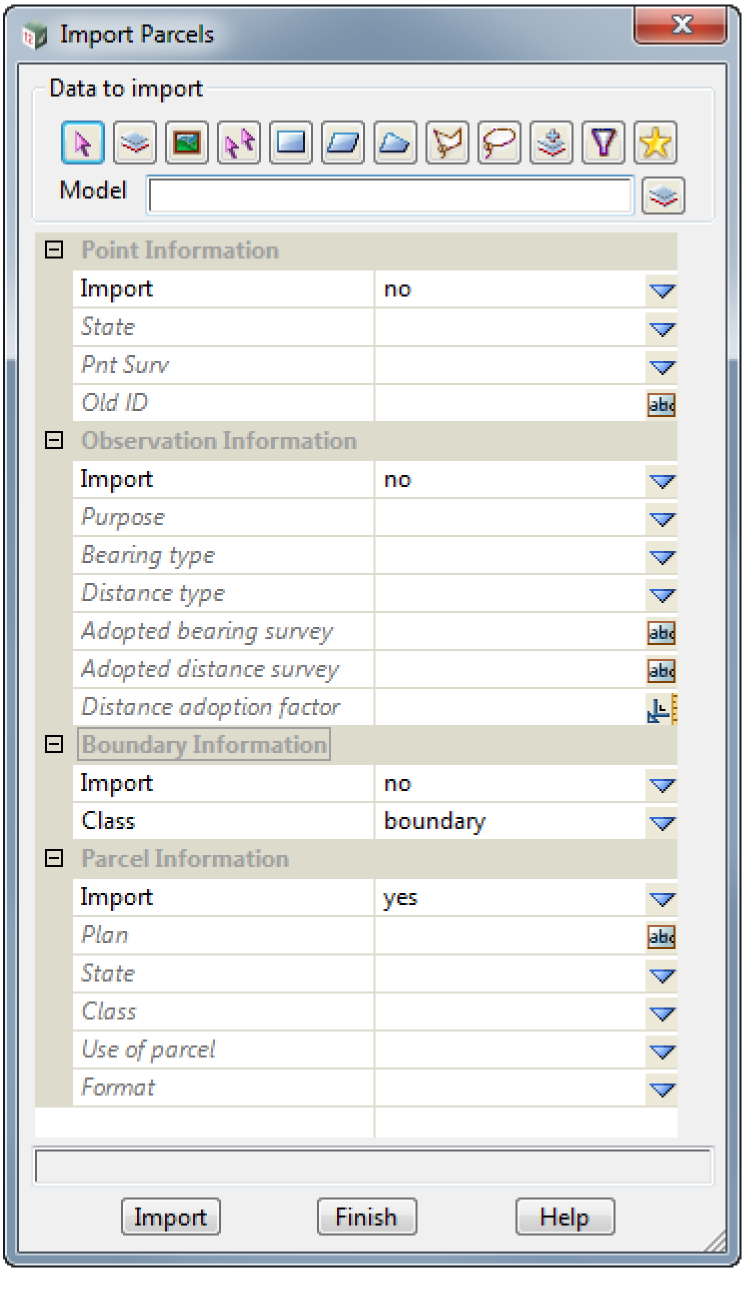
19.4.2.5 Parcels

See

19.4.2.5.1 Importing Parcels
19.4.2.5.2 Creating a Parcel
19.4.2.5.3 Adding Parcels
19.4.2.5.4 Adding a String
19.4.2.5.5 Split Perpendicular

19.4.2.5.1 Importing Parcels

Selecting **Import parcels** brings up the **Import Parcels** panel.

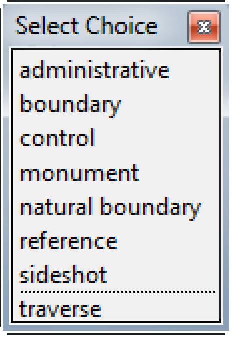


The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source .

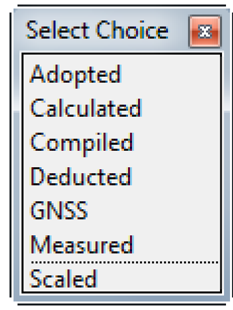
Point Information

Import	choice box		yes, no
State	choice box		existing, proposed
Pnt Surv	choice box		
Old ID	input		

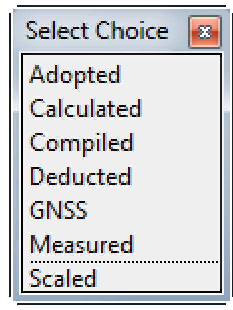
Observation Information

Import	choice box		yes, no

Purpose	choice box	Boundary, Connection, Road
Bearing type	choice box	



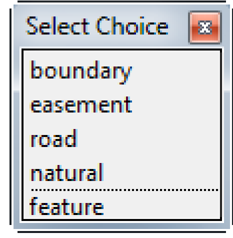
Distance type	choice box	



Adopted bearing survey	input	
Adopted distance survey	input	
Distance adoption factor	measure box	

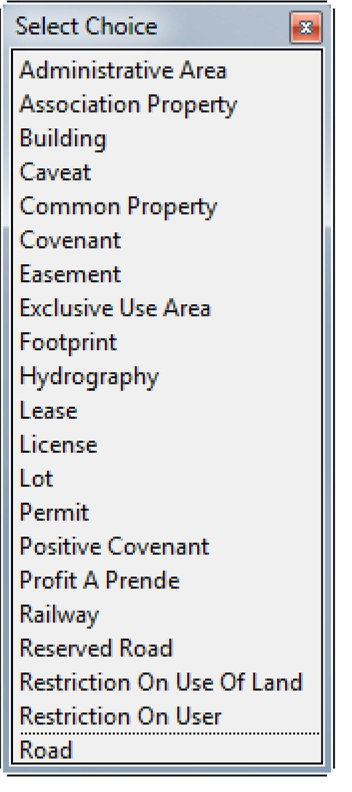
Boundary Information

Import	choice box	yes, no
Class	choice box	



Parcel Information

Import	choice box	yes, no
Plan	input	
State	choice box	adjoining, existing, proposed
Class	choice box	

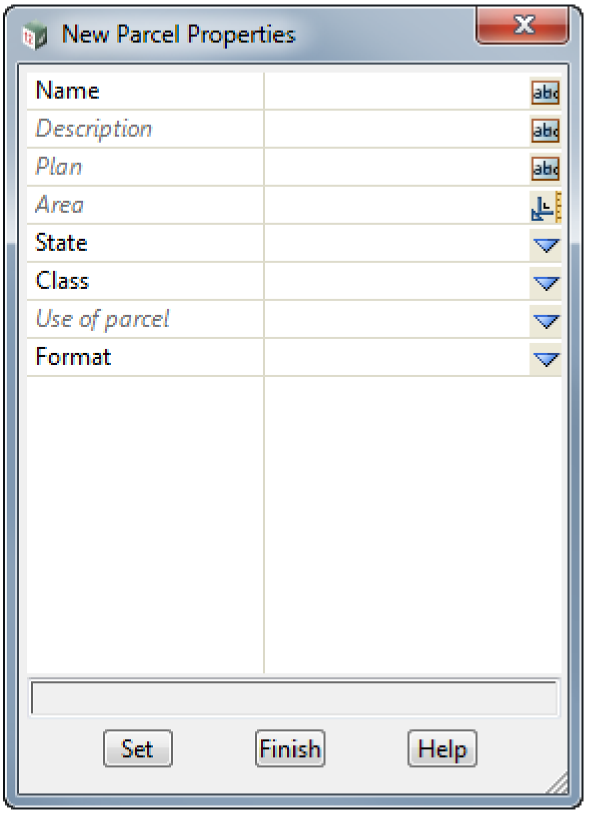


Use of parcel	choice box	
Format	choice box	

Import button

19.4.2.5.2 Creating a Parcel

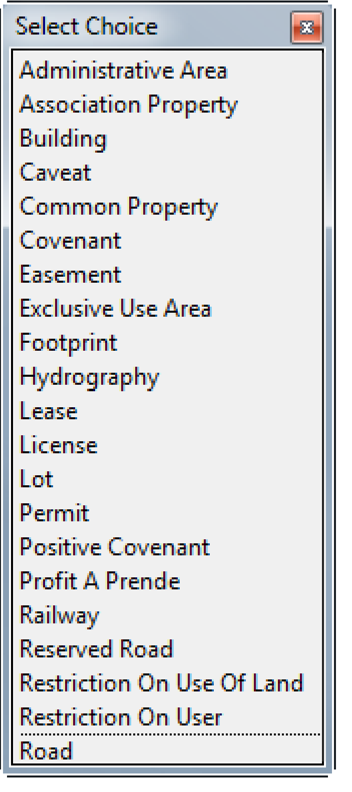
Selecting **Create parcel** brings up the **New Parcel Properties** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input		
the			
Description	input		

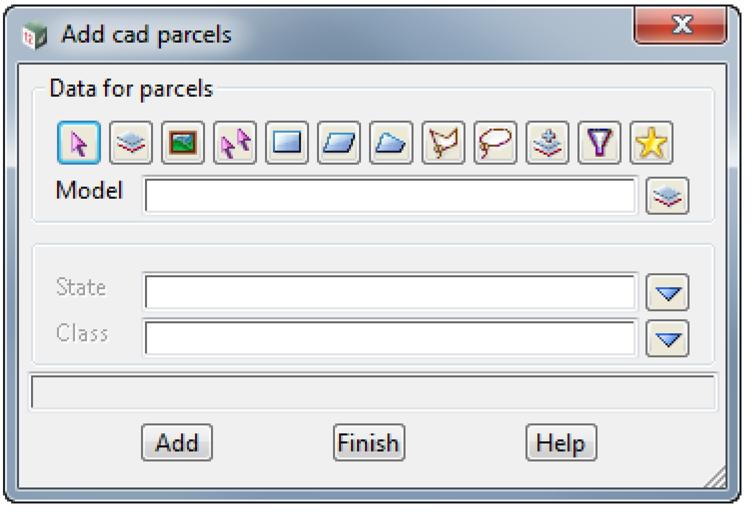
Plan	input	
Area	measure box	
State	choice box	adjoining, existing, proposed
Class	choice box	



Use of parcel	choice box	
Format	choice box	Standard, Strata, Stratum
Set	button	

19.4.2.5.3 Adding Parcels

Selecting **Add parcels** brings up the **Add CAD Parcels** panel.



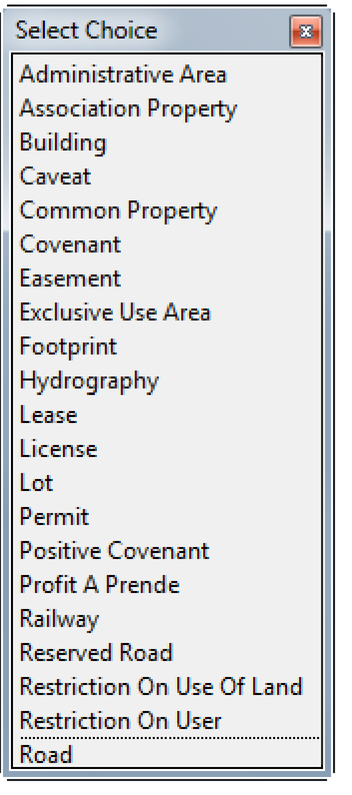
The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source .

State	choice box		adjoining, existing, proposed

Class choice box



Add button

19.4.2.5.4 Adding a String

Select a string to add to the ePlan.

19.4.2.5.5 Split Perpendicular

Select a perpendicular to split.

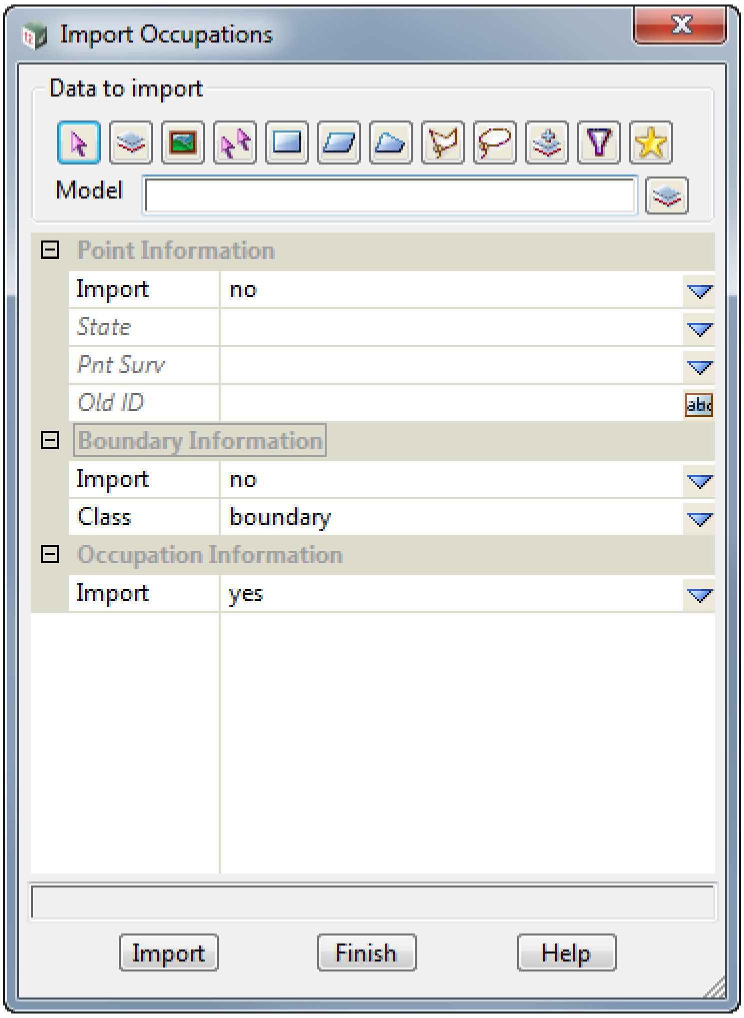
19.4.2.6 Occupations

See

19.4.2.6.1 Importing Occupations

19.4.2.6.1 Importing Occupations

Selecting **Import occupations** brings up the **Import Occupations** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

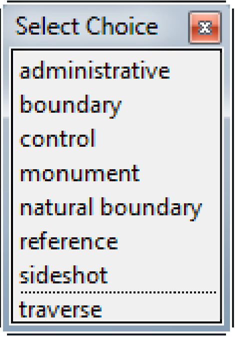
data selection type - for a full description go to 4.19.3 Data Source .

Point Information

Field Description	Type	Defaults	Pop-Up
Import	choice box		yes, no

State choice box existing, proposed

Pnt Surv choice box

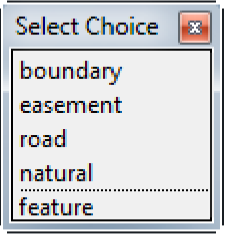


Old ID input

Boundary Information

Import choice box yes, no

Class choice box



Occupation Information

Import choice box yes, no

Import button

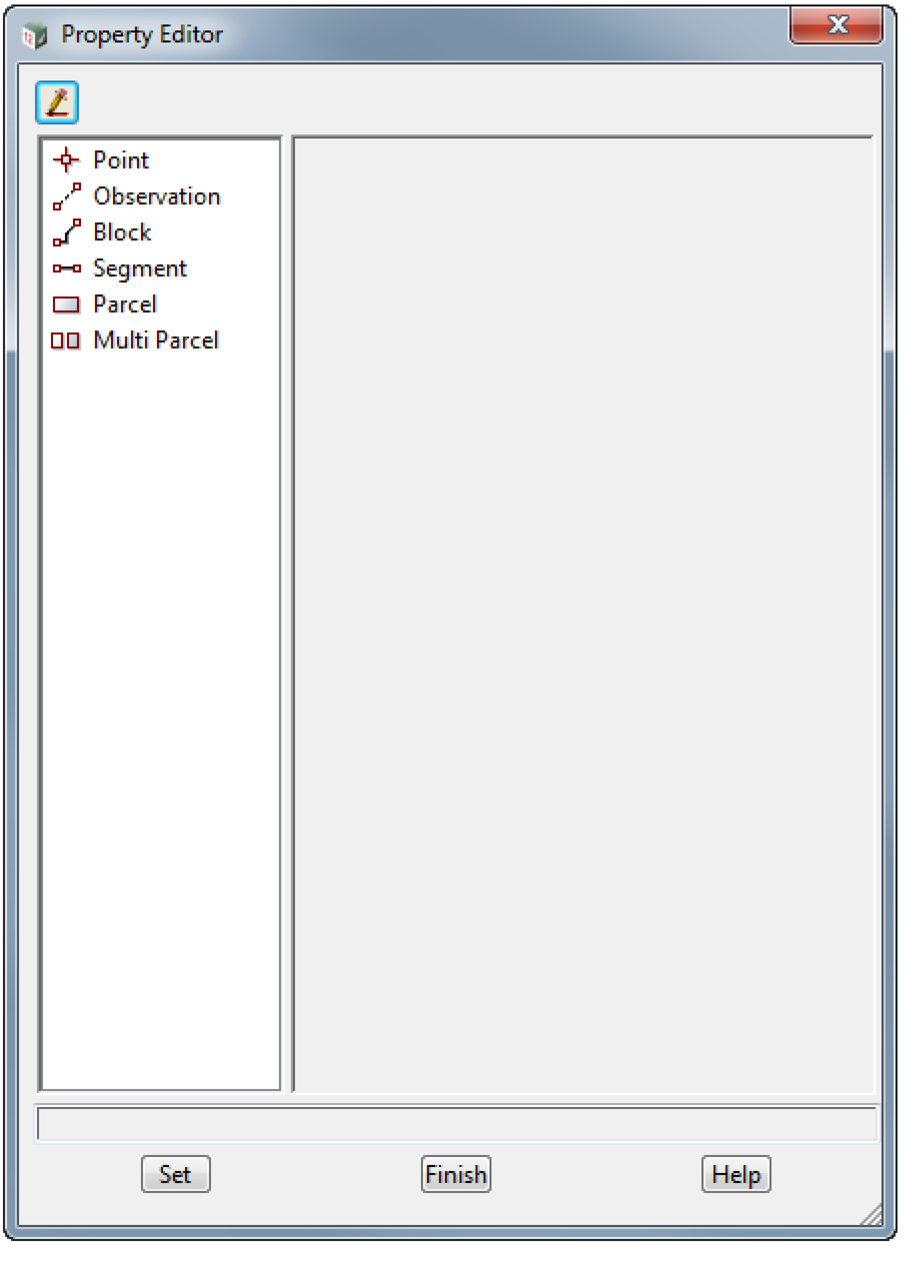
19.4.2.7 Properties

See

19.4.2.7.1 Editing Properties
19.4.2.7.2 Matching Properties

19.4.2.7.1 Editing Properties

Selecting **Edit properties** brings up the **Property Editor** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Point

the

Observation

Block

Segment

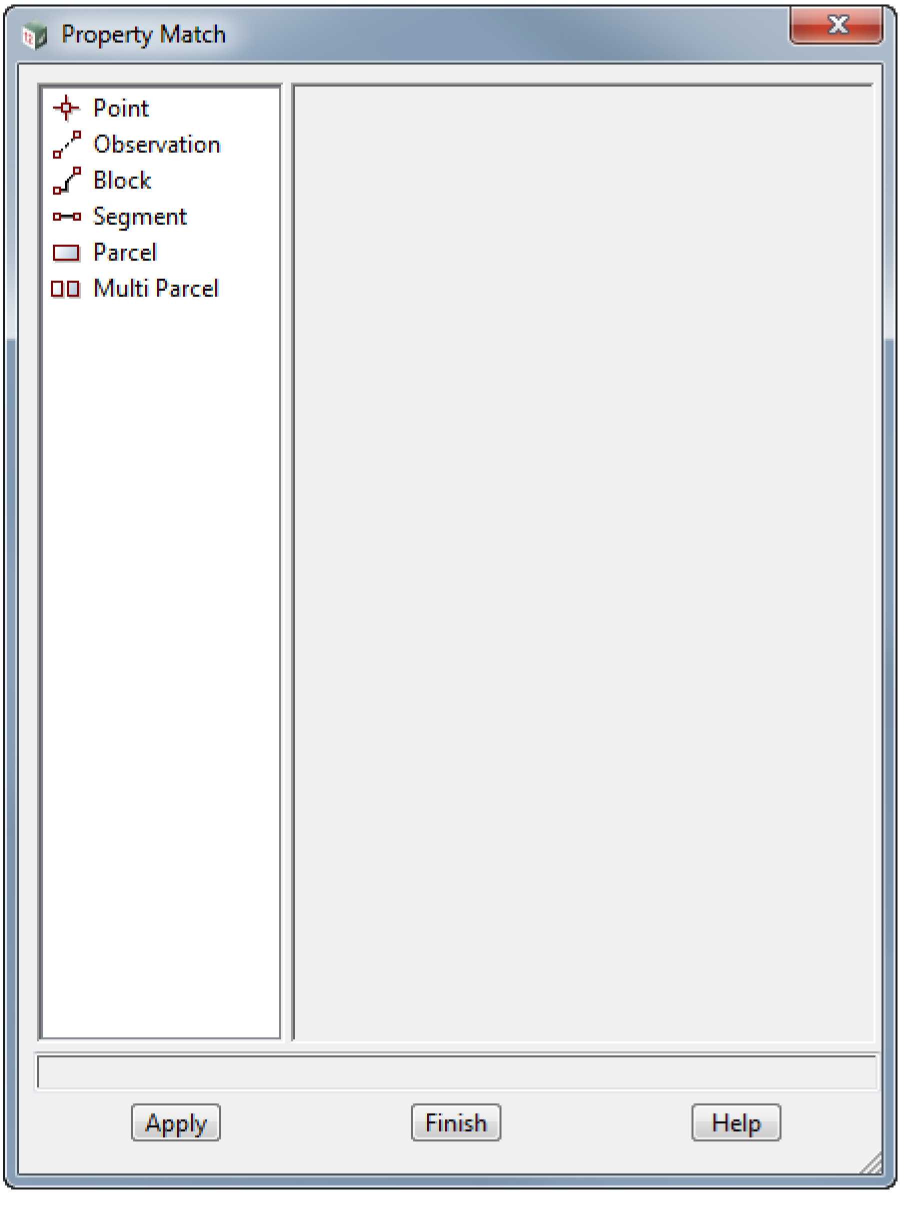
Parcel

Multi Parcel

Set button

19.4.2.7.2 Matching Properties

Selecting **Match properties** brings up the **Property Match** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Point

the

Observation

Block

Segment

Parcel

Multi Parcel

Set	button

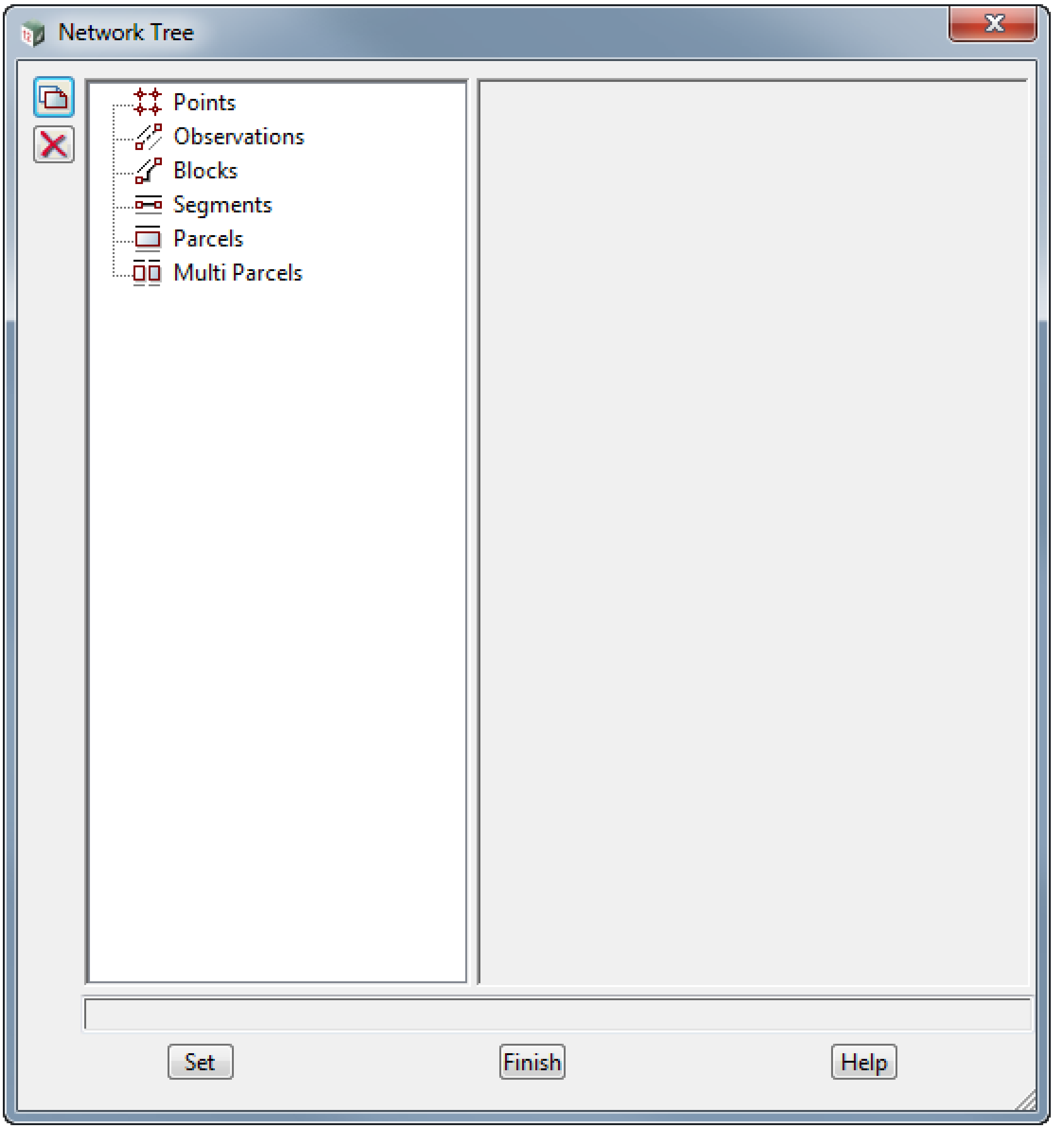
19.4.2.8 Network

See

19.4.2.8.1 Network Tree

19.4.2.8.1 Network Tree

Selecting **Network tree** brings up the **Network Tree** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Points

the

Observations

Blocks

Segments

Parcels

Multi Parcels

Set button

19.4.2.9 Settings

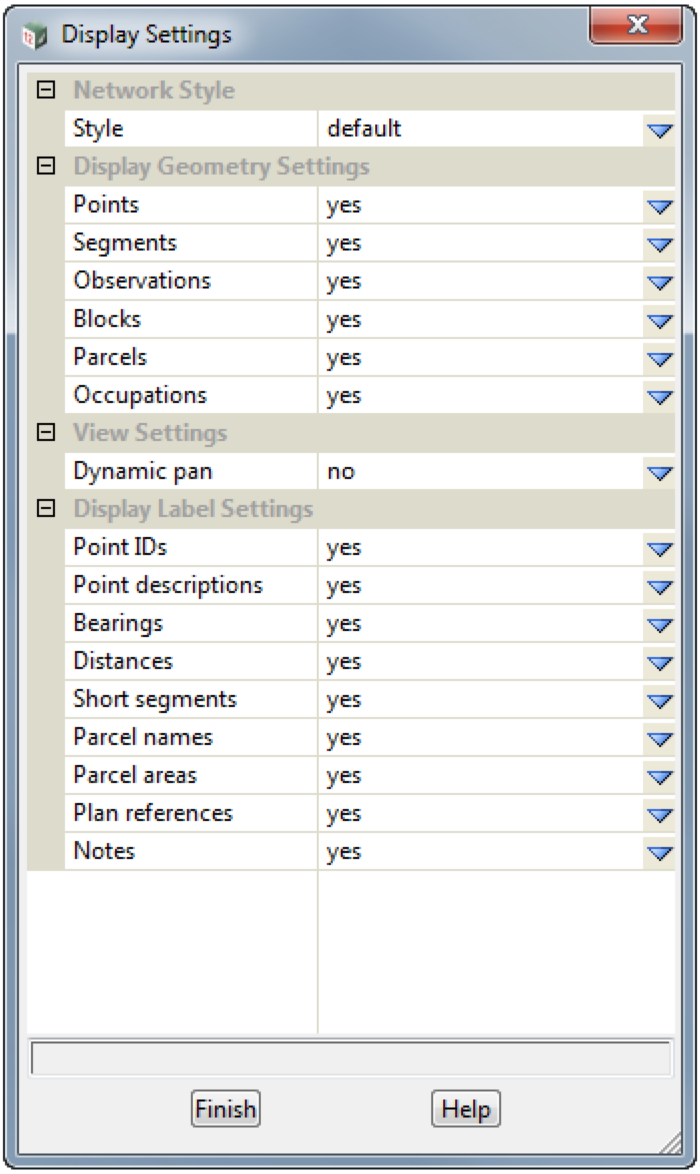
See

19.4.2.9.1 Display Settings

19.4.2.9.2 Survey Header Settings

19.4.2.9.1 Display Settings

Selecting **Display** brings up the **Display Settings** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Network Style

Style	choice box		NSW Standard, default

Display Geometry Settings

Points	choice box		yes, no

Segments	choice box	yes, no
Observations	choice box	yes, no
Blocks	choice box	yes, no
Parcels	choice box	yes, no
Occupations	choice box	yes, no

View Settings

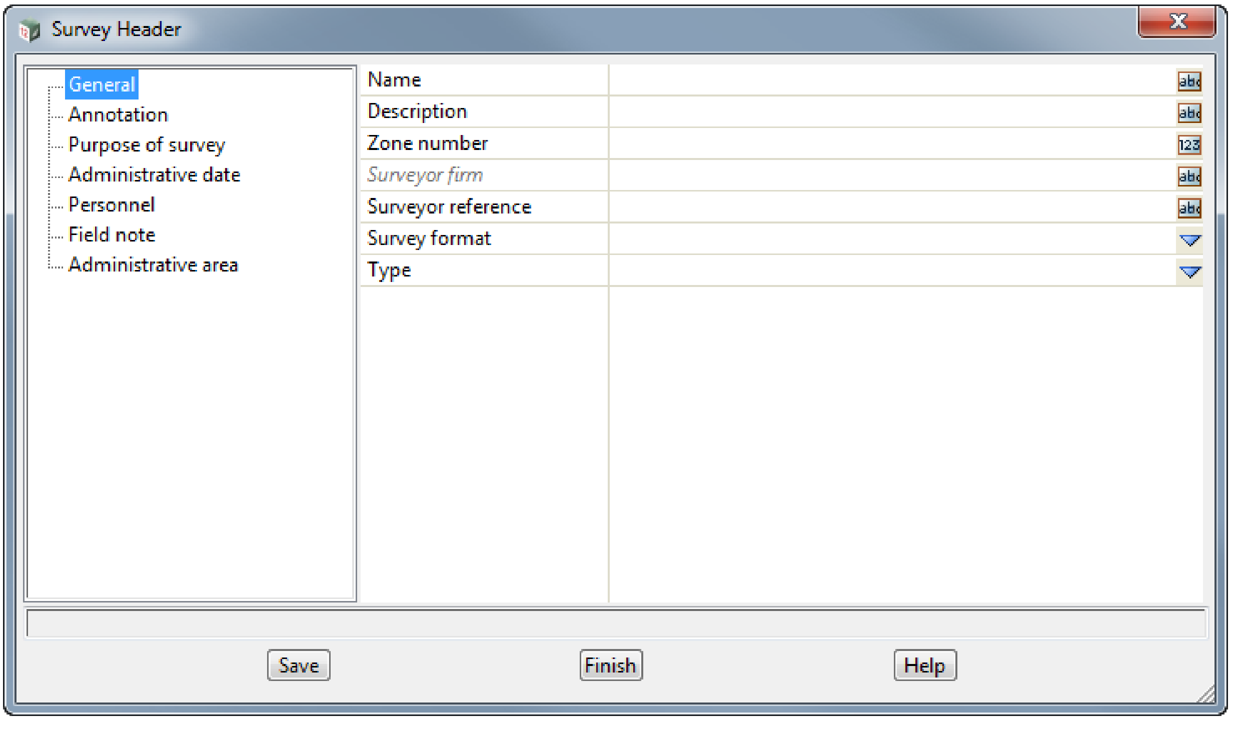
Dynamic pan	choice box	yes, no

Display Label Settings

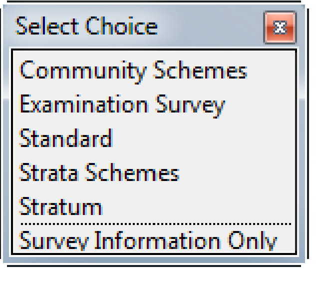
Point IDs	choice box	yes, no
Point descriptions	choice box	yes, no
Bearings	choice box	yes, no
Distances	choice box	yes, no
Short segments	choice box	yes, no
Parcel names	choice box	yes, no
Parcel areas	choice box	yes, no
Plan references	choice box	yes, no
Notes	choice box	yes, no

19.4.2.9.2 Survey Header Settings

Selecting **Survey header** brings up the **Survey Header** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
General node			
Name *the name*	input		
Description	input		
Zone number	number box		
Surveyor firm	input		
Surveyor reference	input		
Survey format	choice box		
Type	choice box		compiled, computed, surveyed

Annotation node

Type

Name

Description

Parcel references

Purpose of survey node

Subdivision

Administrative date node

Type

Date

Personnel node

Name

Role

Registration type

Registration number

Field note node

Note

Administrative area node

Type

Name

Code

Parcel references

Save button

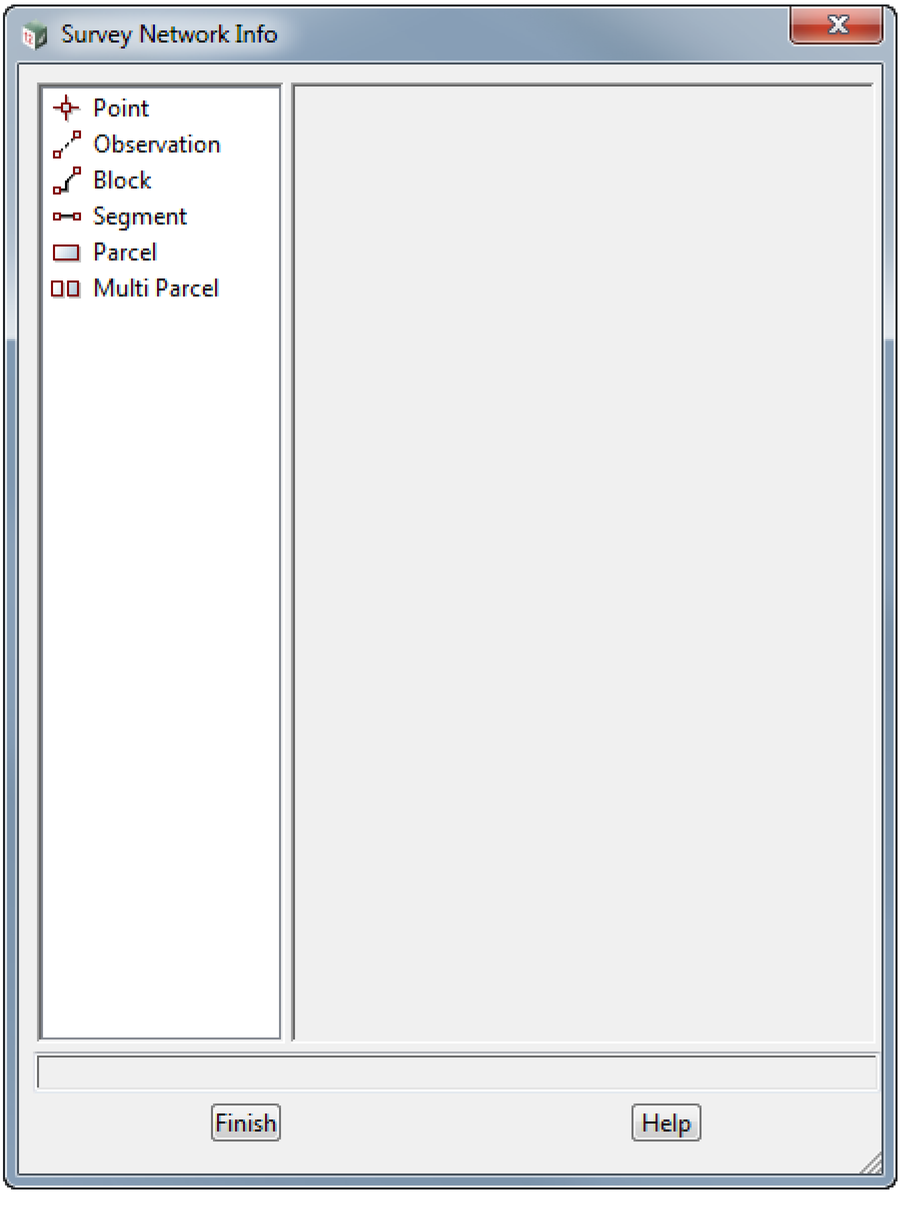
19.4.2.10 Info

See

19.4.2.10.1 Survey Network Info

19.4.2.10.1 Survey Network Info

Selecting **Info** brings up the **Survey Network Info** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Points

the

Observation

Block

Segment

Parcel

Multi Parcel

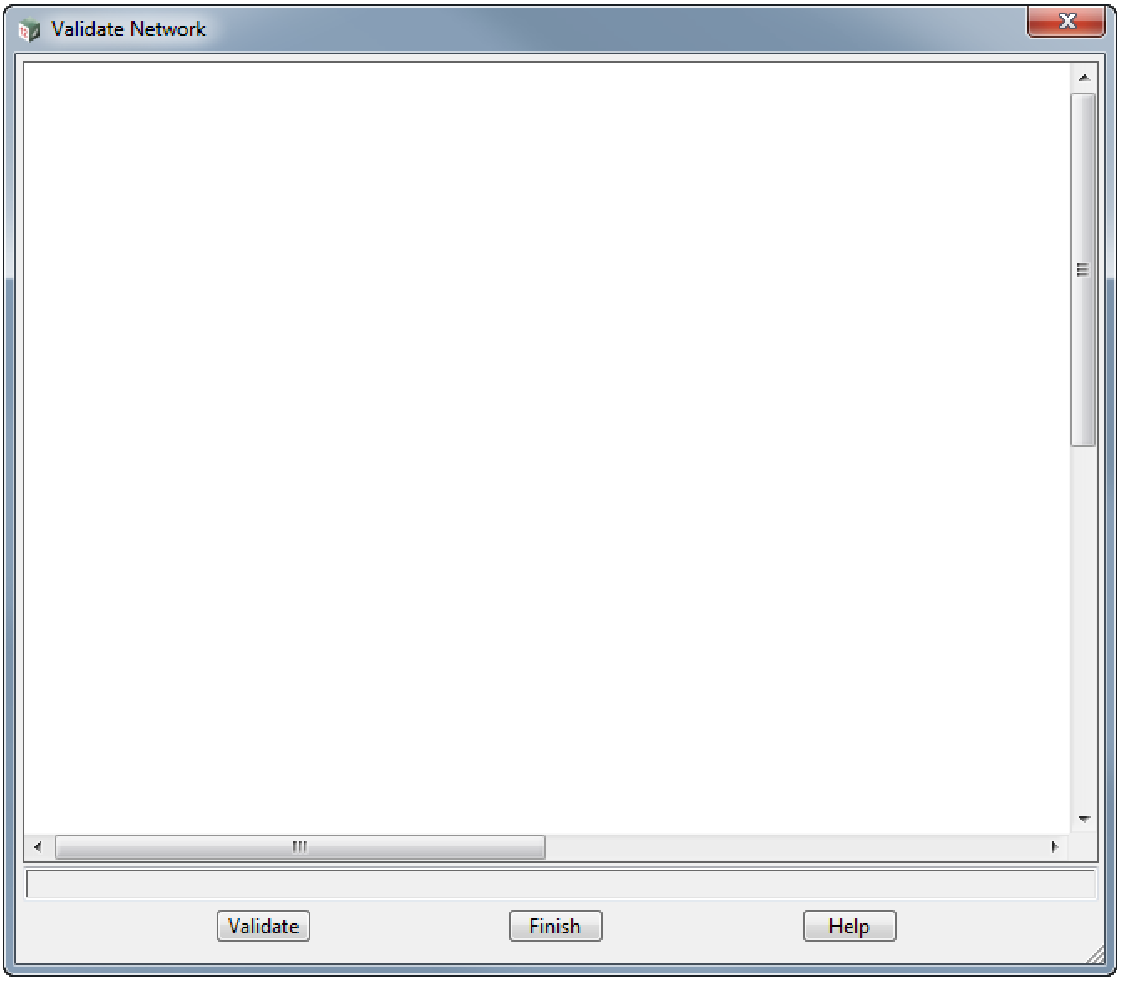
19.4.2.11 I/O

See

19.4.2.11.1 Validate Network
19.4.2.11.2 Export XML

19.4.2.11.1 Validate Network

Selecting **Validate** brings up the **Validate Network** panel.



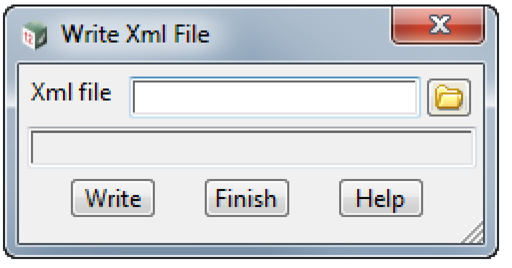
The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Validate	button		

the

19.4.2.11.2 Export XML

Selecting **Export XML** brings up the **Write XML File** panel.



The fields and buttons used in this panel have the following functions.

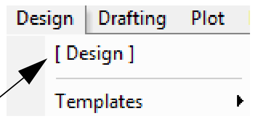
Field Description	Type	Defaults	Pop-Up
XML file	file box		available .xml files
the name of the ePlan XML file			
Write	button		
writes the XML file.			

20 Design

Position of menu: **Design**

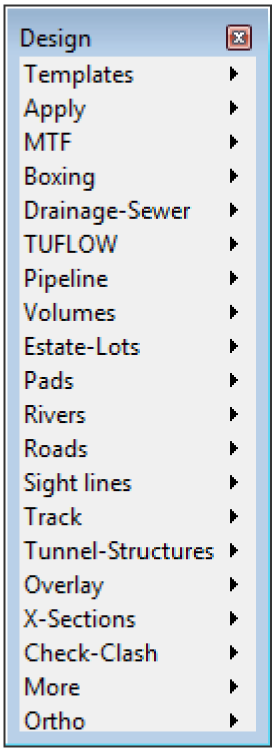
The **Design** walk-right menu is

on Main menu



create floating **Design** menu

on 12d Model menu and floating Design menu



template creation and utilities
apply templates, mtfs, interface, kerb return
edit an mtf file
apply boxing to sections
drainage-sewer options
TUFLOW interface
pipeline option
volumes option
create estate lots
create pads
rivers options
road creation
viewshed, sight distance etc
track options
tunnel and structures
overlay design
cross-sections
clash detection
more design options
Ortho 12d output

For a description of the definition of a template in ***12d Model***, please go to the section 20.1 Templates in 12d Model.

For the option:

Templates, go to	20.2 Templates
Apply	20.3 Apply Functions
MTF	20.4 MTF
Boxing	20.5 Boxing
Drainage-Sewer	22 Drainage and Sewer
Pipeline	22 Pipeline
Volumes	23 Volumes
Estate lots	20.6 Estate Lots
Pads	20.7 Pads
Rivers	21 Rivers
Roads	20.8 Roads
Sight lines	20.10 Sight Lines
Track	20.11 Track
Tunnel-Structures	20.12 Tunnels and Structures

Overlay	20.13 Overlay
X-Sections	20.14 X-Sections
Check/clash	28.9.9.1 Clash Detection
More	20.16 More Design
Ortho	20.17 Ortho 12d

20.1 Templates in 12d Model

In ***12d Model***, Templates provide a quick and easy method for defining design details along a string for use in conceptual designs and simple detailed designs, and visualisations.

For more complex urban design, intersections and roundabouts, Templates are impractical and not an efficient solution. For these cases ***12d Model*** introduced the ***MTF modifiers*** to provide much finer control and provide all the power of string design without losing the ease of Template design in simpler cases.

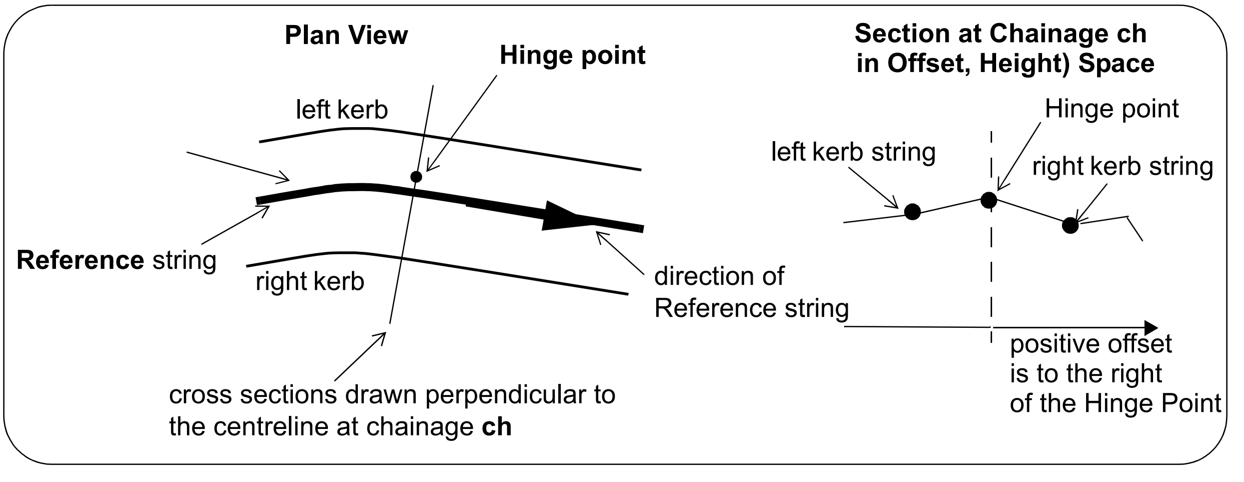
Continue to the next section 20.1.1 Simple Templates.

20.1.1 Simple Templates

But in both the simple **Template** case, and the more powerful **MTF Modifiers** case, the standard civil technique to be able to examining what is happening in a 2D diagram is to look at **cross sections**.

Cross sections are 2D diagrams that display what a section cuts through.

The section is traditionally taken at right angles to a selected **Reference string** (usually an alignment string representing the centreline of a road) and the coordinates of a point in the cross section are (Offset, Height) where Offset is the plan distance along the section of a point in the section from a selected **Hinge Point**, and Height is the height of the point.



A ***12d Model*** **Template** consists of a **left hand side** followed by a **right hand side**.

Each side is made up of a number of **links** with **names** and **colours**, and each link is defined by any two of the three

(a) width of the link

(b) change in z-value from the beginning of the link to the end of the link. that is, the z-value at the end of the link minus the z-value at the start of the link. This is referred to as the **height** of the link.

(c) xfall

and the links are connected sequentially to form a cross section.

Notice that the height is only a **relative** height and **not** a z-value.

On the **left hand side**, the links go from right to left.The vertex (point) at the end of the link is given the same name as the link. The far right of the left hand side is the Hinge Point. So on the left hand side, positive widths go to the left.

On the **right hand side**, the links go from left to right. The vertex (point) at the end of the link is given the same name as the link. The far left of the right hand size is also the Hinge Point. So on the right hand side, positive widths go to the right

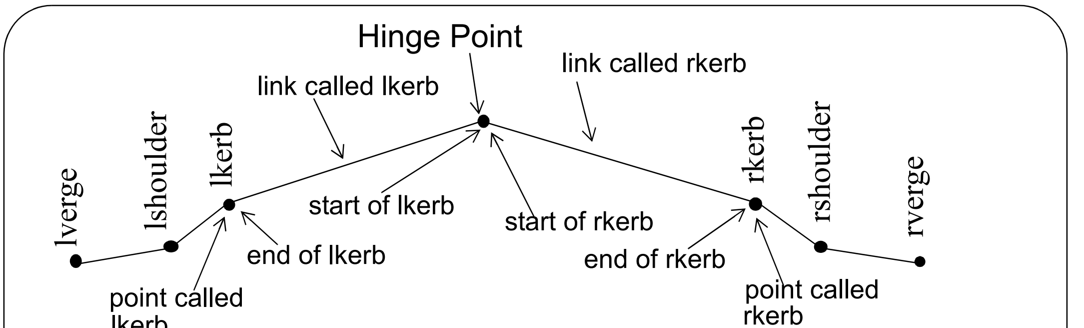
So this definition of the Template is given in terms of **links** and each **link** is defined **relative** to the **previous link** and the left and right hand sides are positioned around a **Hinge Point**.

The **point** at the **end** of the link is given the **name** of the **link** and is called a **Link Point** or **Template Point**.

Hence on the left hand side of the Hinge Point, the Link Point is at the left hand end of the link.

And on the right hand side of the Hinge Point, the Link point is at the right hand end of the link.

For example, a Template consisting of three links called lkerb, lshoulder and lverge on the left hand side, and three links called rkerb, rshoulder, rverge on the right hand side could look like:



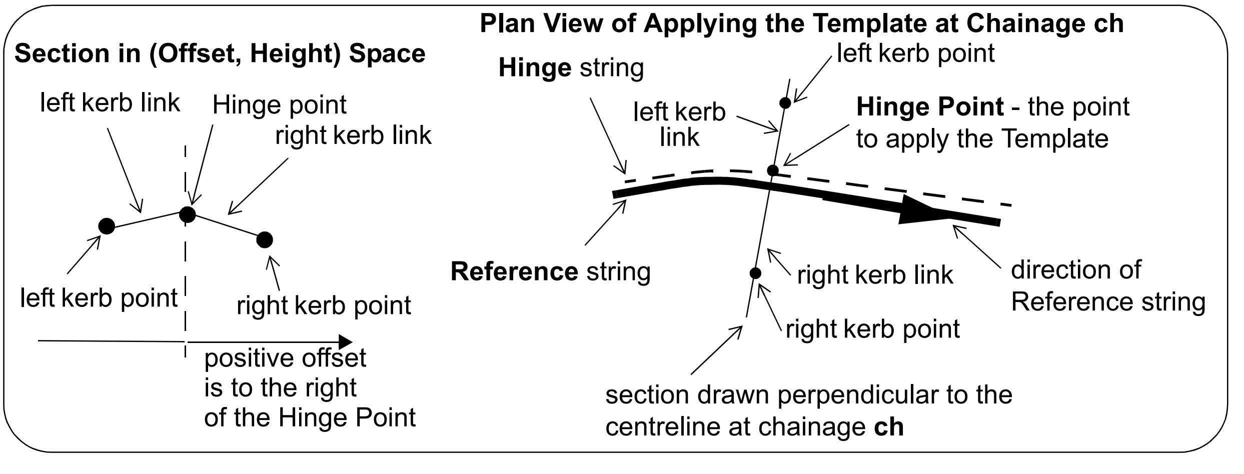
Cross Section View of a Template

Continue to the next section 20.1.1.1 Design X Sections.

20.1.1.1 Design X Sections

A Template is used to generate a cross section at a particular chainage on the Reference string and Hinge string by finding the point where at that chainage the section perpendicular to the **Reference** string **cuts the Hinge string** and the cut point is use as the **Hinge Point** for the Template at that chainage.

So for a simple Template with just the left kerb link on the left hand side and the right kerb link on the right hand side, applying the Template at a chainage **ch** will produce:



This is termed "**applying**" the Template at the chainage **ch**. The created points in the cross section are given the same name as the Link points and joined together in the section in the same order as in the Template

Applying the **Template** to the Reference and Hinge strings means that when the Template is applied at specified chainages down the Reference string.

The sections created by applying a Template are called **design cross sections**, or just **design sections** or **design x sections.**

In ***12d Model***, the design sections are stored as 4d super strings where the (x,y,z) vertices are the coordinates generated applying the Template with the vertex text being the name of the Link point.

Continue to the next section 20.1.1.2 Design Strings.

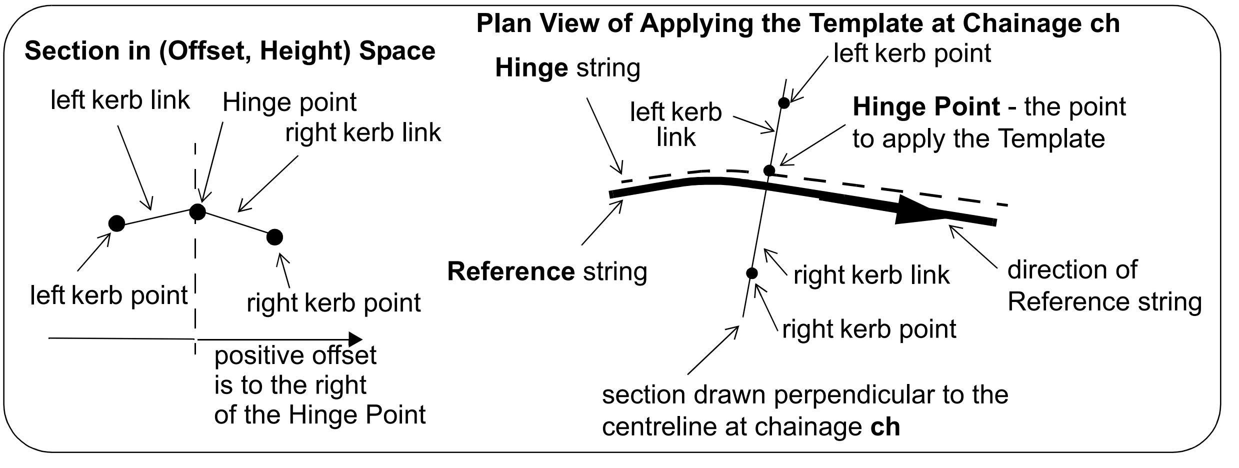
20.1.1.2 Design Strings

For a **Link Point**, for each chainage that the Template is applied there will be a point of the same name on the x section at that chainage.

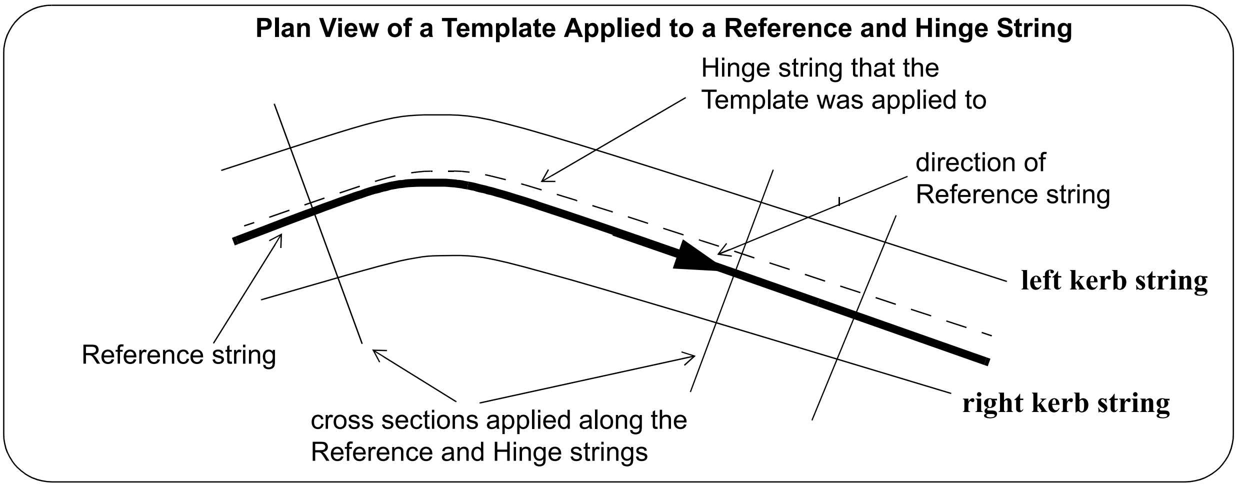
These common points are joined together in chainage order (i.e. longitudinally) to form a ***string*** called the **design string**. The design string is given the **name** of the **Link point** that the string was created from (and hence the same name as the **link**).The default colour of the string is the link colour.

Polygons can also be created for each link by joining the strings created at either end of the link. The polygon is given the link colour.

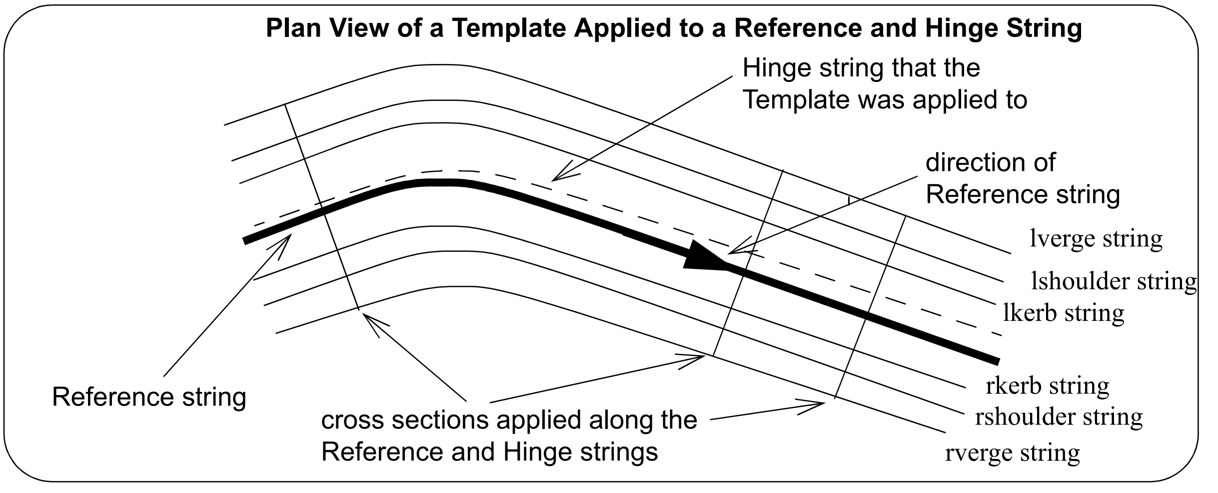
So for the simple two link Template with two **Template Links** *left kerb* and *right kerb:*



will generate two strings *left kerb* and *right kerb*.



And for the case of the more complex six link Template



Continue to the next section 20.1.2 12d Model Templates.

20.1.2 12d Model Templates

12d Model supports Templates with

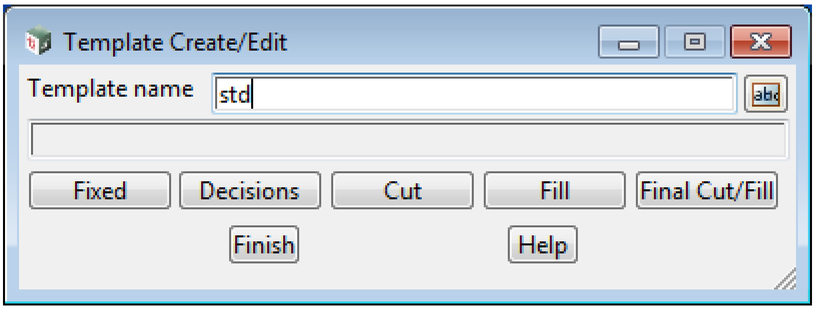
(a) an unlimited number of links that are always created (**fixed links**)

followed by

(b) a decisions table that has special rules to decide when certain links are created

(c) an unlimited number of variable cut links and fill links

(d) a final cut/fill slope to be applied at the end of the last cut or fill template link.



Fixed Template Links

The **fixed template links** are applied to the selected Hinge string and are always used when applying The template regardless of whether the template points are in cut or fill. Each link is defined by two of the three variables, width, height and crossfall, and has a name and colour.

At the end of the fixed links,***12d Model*** checks to see if a **Decisions** table exists.

If a **Decisions** table exists, then it is used and the Decision commands processed.

If a ***Continue Cut/Fill*** command is reached in the decision command processing, then this stops the Decision command processing, and the processing moves onto the **Cut**, **Fill** and **Final Cut/Fill** tables.

If no ***Continue Cut/Fill*** command is reached in the decisions, then the processing stops after the decision commands are exhausted and the **Cut**, **Fill** and **Final Cut/Fill** tables ***ARE NOT USED***.

If **no Decisions** table exists, then the **Cut**, **Fill** and **Final Cut/Fill** tables are used after the **Fixed** table.

Template Decisions

The decisions table is used for

- complicated cut and fill requirements including multiple strata, decisions based on depth below one or more strata or strings, multi-level decisions (i.e. depth decisions followed by fixed links, more depth decision etc.); and
- extended battering including repetitive battering, fixed width batters, and battering relative to a string or strata.

The **decisions** table is documented in the section 21.8 Full Definition of Template Decisions.

Cut and Fill Template Links and Final Cut/Fill

At the end of the fixed links, if there is **no** Decisions table, or if a *Continue Cut Fill* command is reached in the Decisions, the **Cut** and **Fill** tables are used.

(a) If the end of the last link of the fixed template/last decision is in cut, then the Cut links are used.

There can be an unlimited number of cut links and they are used sequentially starting with the first link.

(b) If the end of the fixed template/last decision is in fill, the fill links are used.

There can be an unlimited number of cut and fill links and they are used sequentially starting with the first link.

If the surface tin is intersected when using either the cut links or the fill links, the section is terminated at that intersection point. Otherwise, the entire link is included in the section, and next cut or fill link is then used. This is repeated until the surface is intersected or all the links in the cut or fill table have been used.

If there is still no intersection with the surface after using all of the cut or fill links, then the **Final Cut/Fill** table is used to try and intersect with the tin.

Note: The unlimited cut/fill links are called **variable** template links because at each cross section, a different number of them may be needed. Hence the number of links used varies from cross section to cross section depending on the tin above it.

Final Cut/Fill

If the tin has not been intersected at the end of using the Fixed links, Decisions, and Cut and Fill links, the Final Cut and Fill Slopes are used.

How to create a Template with a fixed link table, a cut and fill tables and the final cut/fill links is discussed further in the option 20.2.2 Create/Edit Templates. The definition of the **Decision Table** is given in the section 21.8 Full Definition of Template Decisions.

For discussion of the Template menu, continue to the next section 20.2 Templates or return to 20.1 Templates in 12d Model.

Note

Around any curves, the Reference string is approximated by chords spaced at the string's chainage interval or, if it is different, the **section separation** interval, or extra points as determined by the horizontal and vertical chord-arc tolerances.

If the chord-to-arc distance is greater than the chord/arc tolerance given in the **Apply** options panels, then extra points are inserted around the curve so that the chord/arc tolerance is met.

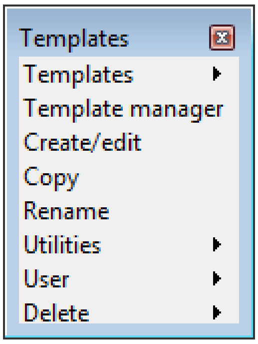
20.2 Templates

Position of menu: **Design =>Templates**

The **templates** walk-right menu has options to create and edit templates, copy, rename and other template utilities.

For a description of templates in ***12d Model***, see section 20.1 Templates in 12d Model.

The **templates** walk- right menu is



list of templates
look settings for templates
create/edit a design template
copy an existing template
rename a template
various template utilities
user defined menu
delete templates from disk

For the option:

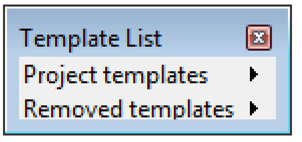
Templates, go to	20.2.1 Templates
Template manager	7.6.12.5 Template Manager
Create/edit	20.2.2 Create/Edit Templates
Copy	20.2.3 Copy Template
Rename	20.2.4 Rename Template
Utilities	20.2.5 Template Utilities
Delete	20.2.6 Delete Templates

20.2.1 Templates

Position of option on menu: **Design =>Templates =>Templates**

The **templates** walk-right menu provides options to list all the templates added to the project templates) and all the templates in the project area but not in the project (removed templates).

The **Templates** walk- right menu is



For the options, see

*Project templates*20.2.1.1 Project Templates

Removed templates20.2.1.2 Removed Templates

20.2.1.1 Project Templates

The **Project templates** walk-right menu provides a list of all the templates in the project.

20.2.1.2 Removed Templates

The **Removed templates** walk-right menu provides a list of all the templates in the project area that have been removed from the project (using the **removed from project** option).

Go to the next section 20.2.2 Create/Edit Templates or return to 20.2 Templates.

20.2.2 Create/Edit Templates

Position of option on menu: **Design =>Templates =>Create/edit**

12d Model supports templates with an unlimited number of fixed links, followed by either

(a) a decisions table

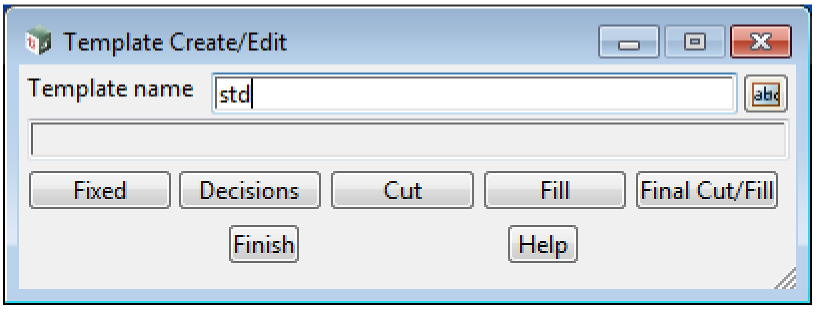
and/or

(b) an unlimited number of cut and fill links and a final cut/fill slope the be applied at the end of the last template link.

To collect the different types of link information needed in the template, the **Template Create/Edit** panel was designed with five buttons **fixed**, **decisions, cut**, **fill** and **final cut/fill** used to bring up and display the panels for creating and editing the fixed, decisions, cut, fill and final cut/fill tables.

For a generate description of Templates in ***12d Model***, see 20.1 Templates in 12d Model.

Selecting **Create/edit** displays the **Template Create/Edit** panel.



The description of the fields and buttons used in the **Template Create/Edit** panel now follows.

The associated panels created by the buttons fixed, cut, fill, final cut/fill will be described in the next four sections and the **Decisions** panel will be described in the section 21.8 Full Definition of Template Decisions.

Field Description	Type	Defaults	Pop-Up
Template name	input		available templates

name of the template being created/edited. If the template already exists, then the associated panels will already contain the information for that template. The buttons fixed, decisions, cut, fill and final cut/fill can only be selected after the **template name** *is given.*

Fixed button

On selecting this button, the **fixed template** *panel is displayed. See* 20.2.2.1 Fixed Template.

Decisions button

On selecting this button, the **Decisions Template** *panel is displayed.*

For the complete definition of decisions, please go to the section 21.8 Full Definition of Template Decisions.

Cut or **Fill** button

The **variable cut** *or* **fill template** *panel is displayed on selecting this button. See* 20.2.2.3 Variable Cut or Fill Template.

Final Cut/Fill button

The **final cut/fill template** *panel is displayed on selecting this button. See* 20.2.2.4 Final Cut/Fill Template.

Note - the buttons can only be selected after the template name has been given in the **Template name** panel field.

How to Use the Panel

(a) A new template is created, or an existing template modified, by first entering the template name into the template field (typed or picked from the pop-up).

(b) To define or edit the fixed sections of the template, select the **Fixed** button and the **Fixed Template** panel will appear.

(c) use the **Decisions** button to define or edit the decisions table

(d) use the **Cut**, **Fill** and **Final Cut/Fill** button to define or edit the variable cut, fill or final cut/fill sections of the template respectively.

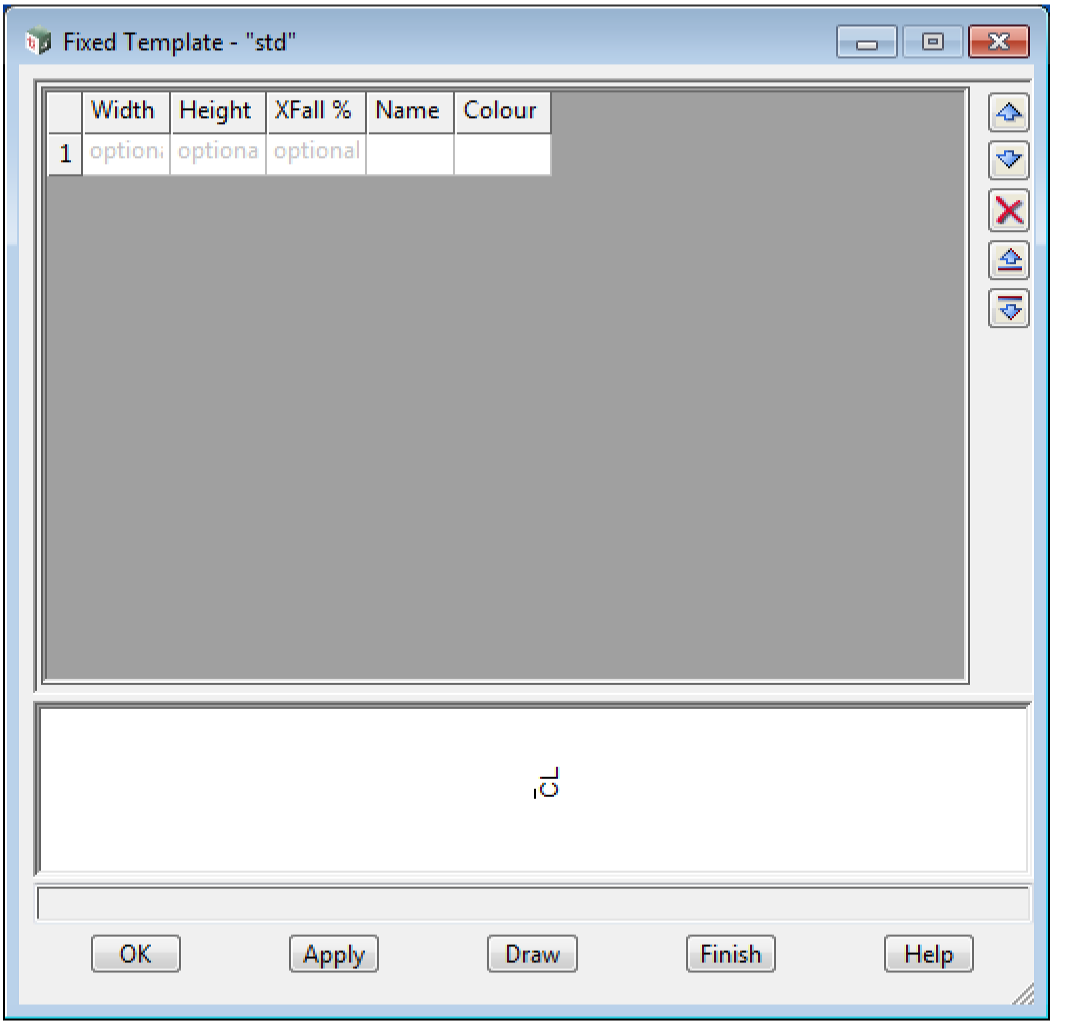
The four panels controlled by the buttons **Fixed**, **Cut**, **Fill** and **Final Cut/Fill** from the **Template Create/Edit** panel will now be described in detail. The **Decisions** button is described in the section 21.8 Full Definition of Template Decisions.

Go to the next section 20.2.2.1 Fixed Template.

20.2.2.1 Fixed Template

If the template already exists, the current fixed template links will be displayed in the panel. If the template is new, the panel will be empty.

The **Fixed Template** panel is



An unlimited number of fixed links can be defined in the grid on the panel. For more information on creating and editing data in a grid, see 4.19.6 Grids in Panels.

For each link, any two of the three values of width, height and percent crossfall (xfall) can be used to defined the link. The colour and name are also defined for the link.

The name of the link is also given to point at the end of the link and also to the string created by the end of the link. The link colour is used as the string colour and when an **Apply Template** or **Apply MTF** is used, the colour of the polygon created by joining the two ends of the link. See 20.1 Templates in 12d Model.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Width	input		

width of the link being defined

Height	input		

the vertical distance between the end point and the start point of the link being defined.
*For a **fixed** link, height is **positive** in the **up** direction. Hence for a fixed link, height is positive up and*

negative down.

NOTE - if height and cross-fall are being used to go down, then define height to be negative and xfall to be negative.

X-fall % input

*percent cross fall of the link being defined - units are percent grade. For **xfall**, **positive** is **up** and negative down.*

NOTE - if height and xfall are being used to go down, then define height to be negative and xfall to be negative.

Colour colour box available colours

colour of the link being defined. If no colour is supplied, the default colour is used.

Name input

name of the link being defined - this is used as the name of the string created by the end point of the link.

OK/Apply button

OK *stores the values in the fields and exits the panel.* **Apply** *stores the values but doesn't exit the panel.*

***Warning** - If the* **OK** or **Apply** *button is **not** selected, then **no** new information will be recorded.*

Draw button

draws the fixed template at the bottom of the panel. Automatically does a fit.

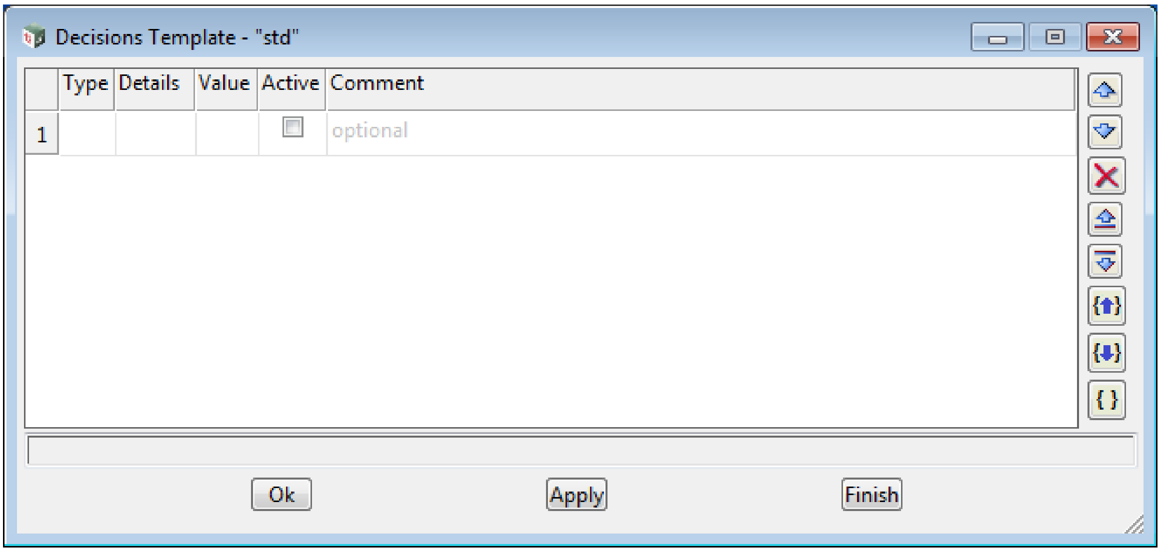
NOTE - if height and xfall are being used to go down, then define height to be negative and xfall to be negative.

Go to the next section 20.2.2.2 Decisions Template or return to 20.2.2 Create/Edit Templates.

20.2.2.2 Decisions Template

If the template already exists, the current Decisions Template links will be displayed in the panel. If the template is new, the panel will be empty.

The **Decisions Template** panel is



The **Decisions Template** panel is documented in the section 21.8 Full Definition of Template Decisions.

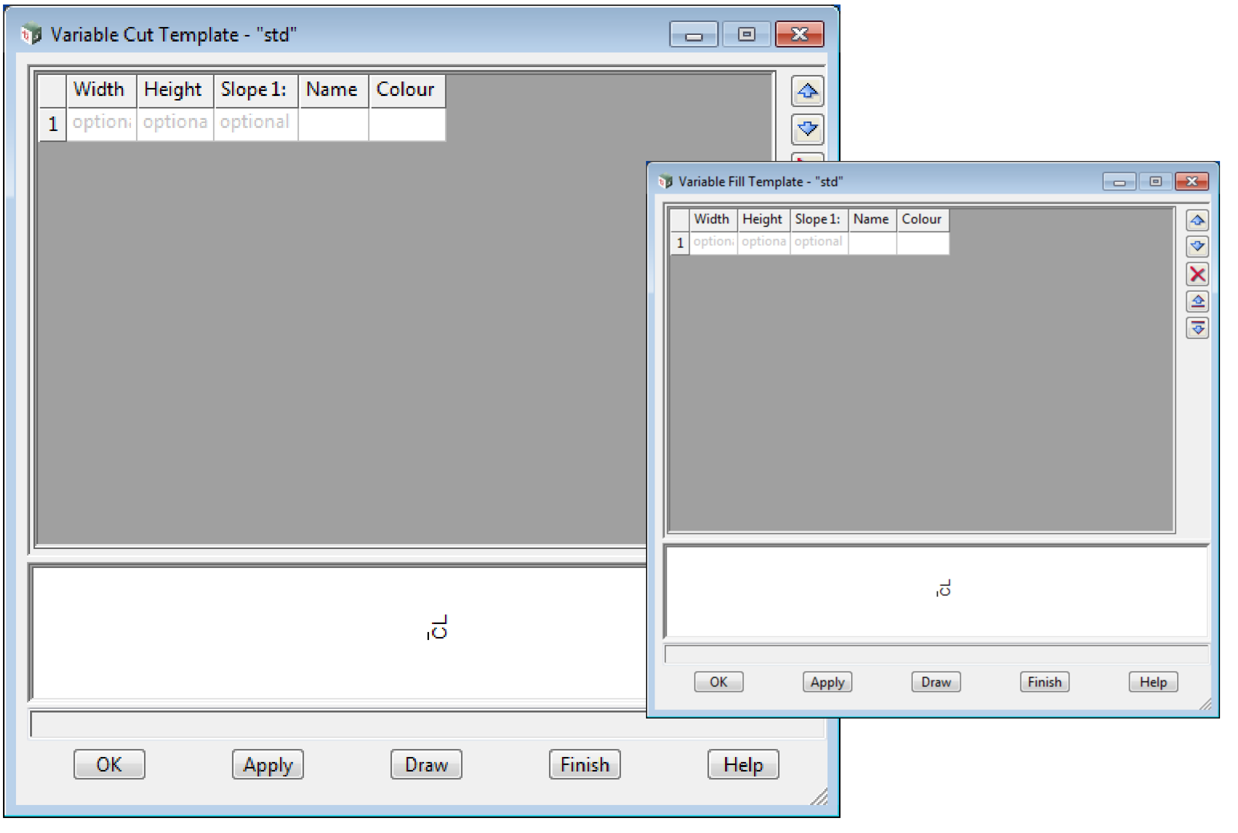
Go to the next section 20.2.2.3 Variable Cut or Fill Template or return to 20.2.2 Create/Edit Templates.

20.2.2.3 Variable Cut or Fill Template

The definition of the cut or fill template links are similar to the fixed template links except the **slope** is given as a “one vertical in a user given horizontal value” rather than a percent crossfall.

If the template already exists, the current cut (fill) template links will be displayed in the panel. If the template is new, the panel will be empty.

The **Variable Cut Template** panel and the **Variable Fill Template** panel are similar and look like



An unlimited number of links can be defined in the variable grid.For more information on creating and editing data in a grid, see 4.19.6 Grids in Panels.

For each link, any two of the three values of width, height and slope can be used to define the link. The colour and name are also defined for the link.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Width	input		

width of the link being defined

Height	input		

the vertical distance between the end point and the start point of the link being defined.

Important note:

*For **cut** link, **height** is **positive** in the **up** direction. That is, for a cut link, height is positive up and negative down.*

*For **fill** link, **height** is **positive** in the **down** direction. That is, for a fill link, height is positive down and negative up.*

This definition for fill links of down being positive for heights may seem strange but it is to fit in with the definition of slope for fill links where a positive slope is down. This was done so that most of the entries in the Slope table did not need a negative sign in front of them.

Slope 1: input

slope of the cut link being defined. The units are "one vertical in the given horizontal value".

Special note*: the value* ***0*** *is used to* ***designate*** *a* ***horizontal*** *slope since a vertical slope is not allowed.*

In the Variable Cut Template, for ***cut slopes****,* ***positive is up*** *and negative down*
In the Variable Fill Template, for ***fill slopes****,* ***positive is down*** *and negative is up.*

In a Variable **Cut** Template, **positive slope** is **up** and **positive height** is **up**.

In a Variable **Fill** Template, **positive slope** is **down** and **positive height** is **down**.

Important note:
This definition of slope for fill links of down being positive may seem strange but it was done so that most of the entries in the Slope table did not need a negative sign in front of them.

NOTE -in cut: *if height and slope are being used to go down, then define height to be negative* ***and*** *slope to be negative.*

In fill: *if height and slope are being used to go up, then define height to be negative* ***and*** *slope to be negative.*

Colour colour box available colours

colour of the link being defined. If no colour is supplied, the default colour is used.

Name input

name of the link being defined - this is used as the name of the string created by the end point of the link.

OK/Apply button

OK *stores the values in the fields and exits the panel.* **Apply** *stores the values but doesn't exit the panel.*

Warning - If the **OK** or **Apply** button is **not** selected, then **no** new information will be recorded**.**

NOTE -in cut: if height and slope are being used to go down, then define height to be negative and slope to be negative.
In fill: if height and slope are being used to go up, then define height to be negative and slope to be negative.

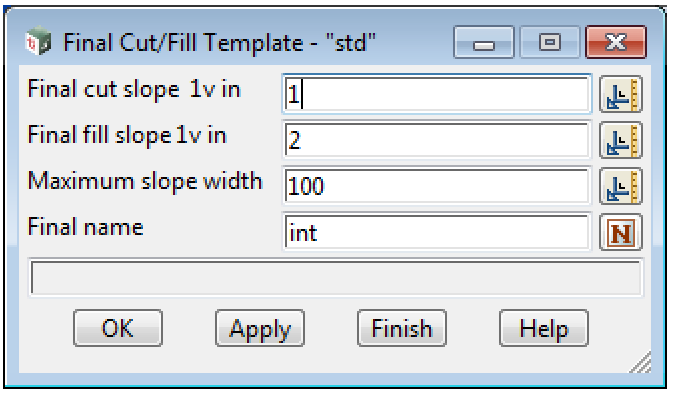
Go to the next section 20.2.2.4 Final Cut/Fill Template or return to 20.2.2 Create/Edit Templates.

20.2.2.4 Final Cut/Fill Template

The Final Cut/Fill Template is used if the links in either the cut or fill templates are exhausted without an intersection with the tin being found.

If the Template already exists, the current final cut/fill values will be displayed in the panel. If the Template is new, the default values are displayed.

The **Final Cut/Fill Template** panel is



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Final cut slope 1 v in	input	0	no slope, 0,1,2,3,4,5,10

the cut slope is used if the end of the last link in the template is in cut. That is, it is below the tin.

*This value is the Cut slope for the interface calculation to be done at the end of the last link of the template. A cut slope of one vertical to the given value of horizontal units is used. The value 0 is used to designate a **horizontal** slope - vertical slopes are not allowed.*

*For **final cut slope**, **positive is up** and negative down*

Field Description	Type	Defaults	Pop-Up
Final fill slope 1 v in	input	0	no slope, 0,1,2,3,4,5,10

the fill slope is used if the end of the last link in the template is in fill. That is, it is above the tin.

*This value is the Fill slope for the interface calculation to be done at the end of the last link of the template. A fill slope of one vertical to the given value of horizontal units is used. The value 0 is used to designate a **horizontal** slope - vertical slopes are not allowed.*

*For **final fill slope**, **positive is down** and negative is up.*
This definition of fill slope being positive when going down is used so that the value in the **Final fill slope 1 v in** *field is normally positive.*

Field Description	Type	Defaults	Pop-Up
Maximum slope width	input	100	

the maximum width for the final slope.

Field Description	Type	Defaults	Pop-Up
Final name	input	int	

name for the string created by this link (normally this string lies on the tin - the interface string).

Field Description	Type	Defaults	Pop-Up
OK/Apply	button		

OK *stores the values in the fields and exits the panel.* **Apply** *stores the values but doesn't exit the panel.*

Warning - If the **OK** or **Apply** button is **not** selected, then **no** new information will be recorded**.**

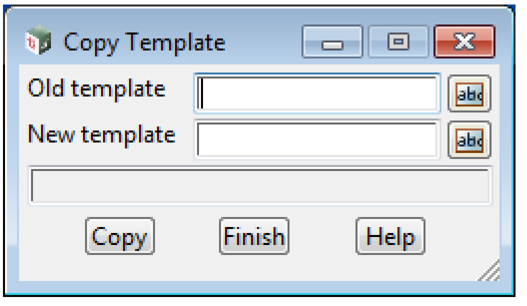
Go to the next section [20.2.3 Copy Template] or return to [20.2.2 Create/Edit Templates] or [20.2 Templates].

20.2.3 Copy Template

Position of option on menu: **Design =>Templates =>Copy**

A copy of an existing template can be made using the **Copy** option. This is often useful when a new template that is similar to an existing template is needed. The existing template can be copied and the copy then edited and modified.

On selecting the **Copy** option, the **Copy Template** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Old template	template box		available templates

name of the template to be copied.

New template	template box		available templates

name of the template copy.

Copy	button		

after selecting this button, the template given in the **Old template** *field will be copied and the copy given the name in the* **New template** *field.*

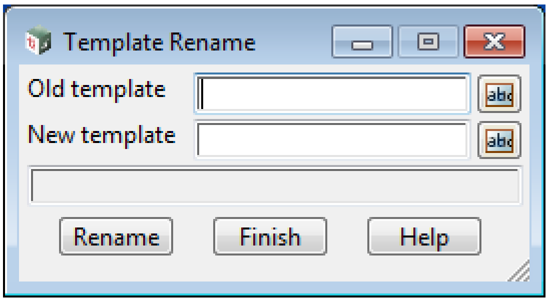
Go to the next section 20.2.4 Rename Template or return to 20.2 Templates.

20.2.4 Rename Template

Position of option on menu: **Design =>Templates =>Rename**

On selecting the **Rename** option, the **Template Rename** panel is displayed.

This panel can be used to change the names of existing templates



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Old template	template box		available templates

name of the template to be renamed.

New template	template box		

new name for the template

Rename	button		

Change the name of the template in the **old template** *field to the name given in the* **new template** *field.*

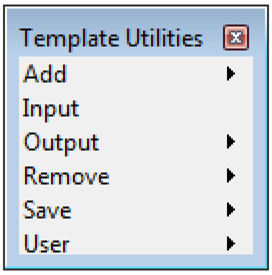
Go to the next section 20.2.5 Template Utilities or return to 20.2 Templates.

20.2.5 Template Utilities

Position of menu: **Design =>Templates =>Utilities**

The **Utilities** menu contains miscellaneous options involving templates.

The **Utilities** walk-right menu is



add templates to project
read in template file
write out templates
remove templates from project
save templates to disk

Each option will now be described.

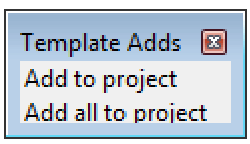
For the options see

Add	20.2.5.1 Add Templates
Input	20.2.5.2 Input Templates
Output	20.2.5.3 Output Templates
Remove	20.2.5.4 Remove Templates
Save	20.2.5.5 Save Templates

20.2.5.1 Add Templates

Position of menu: **Design =>Templates =>Utilities =>Add**

Removed templates can be added back into the project using the options in the **Add** walk-right menu. The **Template Adds** walk-right menu is



For the option *Add to project*, go to 20.2.5.1.1 Add a Template to a project
Add all to project 20.2.5.1.2 Add All Templates to a Project

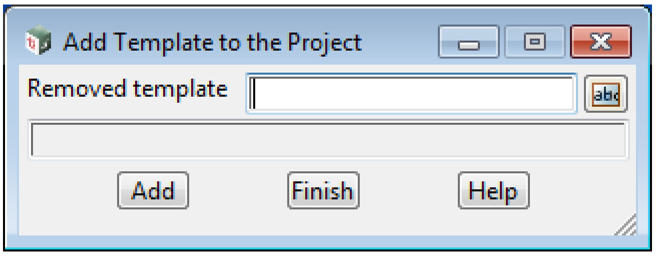
Go to the next section 20.2.5.1 Add Templates, or return to 20.2.5 Template Utilities or 20.2 Templates.

20.2.5.1.1 Add a Template to a project

Position of option on menu: **Design =>Templates =>Utilities =>Add =>Add to project**

Templates in the working project area but not yet in the project can be added to the project using this option.

On selecting the **Add to project** option, the **Add Template to the Project** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Removed template	removed templates box		removed templates

name of the template to be added to the project.

Add	button		

add the template given in the **template** *field to the working project.*

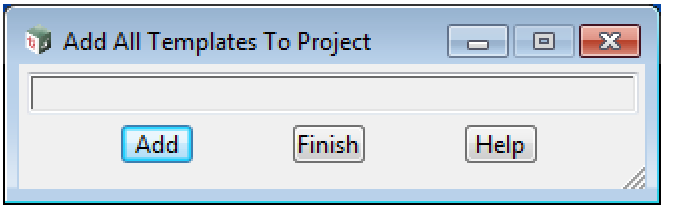
Go to the next section 20.2.5.1.2 Add All Templates to a Project, or return to 20.2.5.1 Add Templates or 20.2.5 Template Utilities.

20.2.5.1.2 Add All Templates to a Project

Position of option on menu: **Design =>Templates =>Utilities =>Add =>Add all to project**

The **Add all to project** option is used to add all the removed templates back into the project.

On selecting the **Add all to project** option, the **Add All Templates To Project** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Add	button		

after selecting this button, all removed templates in the working project will be added to the project.

Return to 20.2.5.1 Add Templates or 20.2.5 Template Utilities.

20.2.5.2 Input Templates

This **input** option is used to read in templates from files in this special ***12d Model*** format.

The option has already been described under **File I/O=>Templates input** (see 8.11.1 Templates Input)

Go to the next section 20.2.5.3 Output Templates, or return to 20.2.5 Template Utilities.

20.2.5.3 Output Templates

The **output** option writes out one or all templates in the ***12d Model*** template format.

The option has already been described under **File I/O=>Templates output (**see 8.11.2 Templates Output).

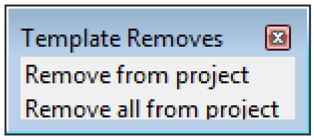
Go to the next section 20.2.5.4 Remove Templates, or return to 20.2.5 Template Utilities.

20.2.5.4 Remove Templates

Position of menu: **Design =>Templates =>Utilities =>Remove**

Templates can be removed from the project using the options in the remove walk-right menu. Removed templates are not deleted but remain in the project area and are no longer accessible in the project.

The **Template Removes** walk-right menu is



For the options see

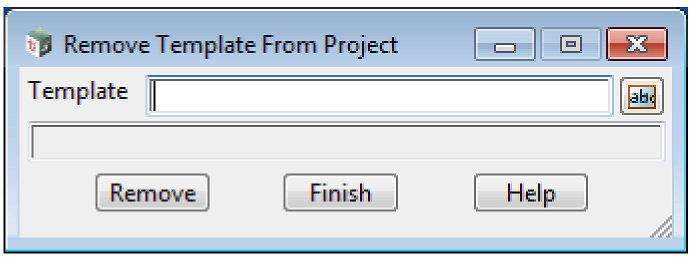
Remove from project 20.2.5.4.1 Remove Template from Project

Remove all from project 20.2.5.4.2 Remove All Templates From Project

20.2.5.4.1 Remove Template from Project

Position of option on menu: **Design =>Templates =>Utilities =>Remove => Remove from project**

Individual templates can be removed from the project using the **Remove from project** option and on selecting the option, the **Remove Template From Project** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Template	template box		available templates

name of the template to be removed from the working project.

Remove	button		

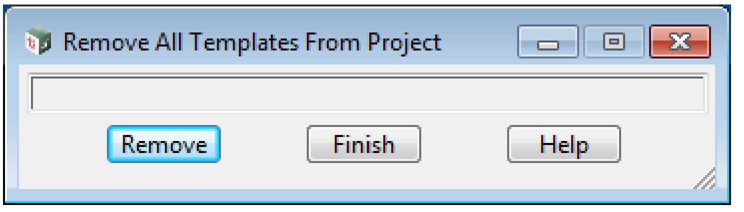
after selecting this button, the template given in the template field will be removed from the working project.

Go to the next section 20.2.5.4.2 Remove All Templates From Project, or return to 20.2.5.4 Remove Templates or 20.2.5 Template Utilities.

20.2.5.4.2 Remove All Templates From Project

Position of option on menu: **Design =>Templates =>Utilities =>Remove => Remove all from project**

All templates can be removed from the project using the **Remove all** option and on selecting the option, the **Remove All Templates From Project** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Remove	button		

after selecting this button, all templates in the working project will be removed.

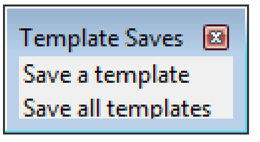
Return to 20.2.5.4 Remove Templates or 20.2.5 Template Utilities.

20.2.5.5 Save Templates

Position of menu: **Design =>Templates =>Utilities =>Save**

Templates can be saved on disk. This is done automatically by the **Save** option on the ***12d Model*** but this option allows just the templates to be saved.

The **Template saves** walk-right menu is



For the options see

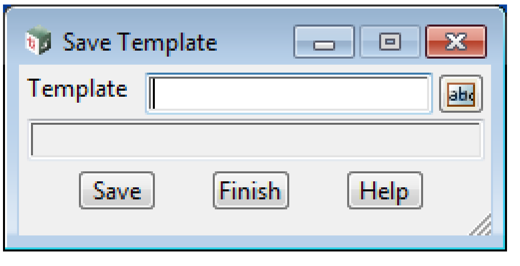
*Save a template*20.2.5.5.1 Save a Template

Save all templates20.2.5.5.2 Save All Templates

20.2.5.5.1 Save a Template

Position of option on menu: **Design =>Templates =>Utilities =>Save =>Save a template**

On selecting the **Save a template** option, the **Save Template** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Template	template box		available templates

name of the template to be saved.

Save	button		

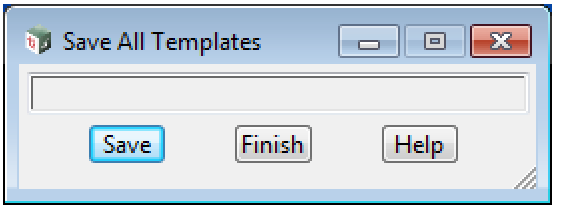
after selecting this button, the template given in the template field will be saved to disk.

Go to the next section 20.2.5.5.2 Save All Templates, or return to 20.2.5.5 Save Templates or 20.2.5 Template Utilities.

20.2.5.5.2 Save All Templates

Position of option on menu: **Design =>Templates =>Utilities =>Save =>Save all templates**

On selecting the **Save all templates** option, the **Save All Templates** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Save	button		

after selecting this button, all templates in the working project will be saved to disk. Unless an error occurs, the panel will be removed after the saving is completed.

Return to 20.2.5.5 Save Templates or 20.2.5 Template Utilities.

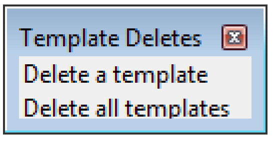
20.2.6 Delete Templates

Position of menu: **Design =>Templates =>Delete**

Using the **Delete** option, templates can be deleted from the project and moved to the Trash Bin, or permanently deleted from the computer disk so that they can no longer be accessed or take up disk space.

To help protect the user against disasters, when a template is selected for deletion, a **Yes-No** pop-up menu is used to confirm that the user did intend deleting the template.

The **Template Deletes** walk-right menu is



For the options see

*Delete a template*20.2.6.1 Delete a Template

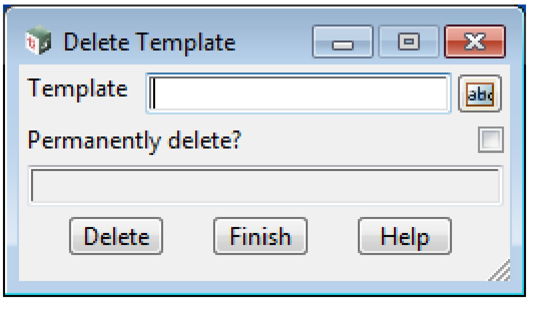
*Delete all templates*20.2.6.2 Delete All Templates

20.2.6.1 Delete a Template

Position of option on menu: **Design =>Templates =>Delete =>Delete a template**

The **delete a template** option is used to delete individual template from the project.

On selecting the **Delete a template** option, the **Delete Template** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Template	template box		project templates

name of the template to be deleted.

Permanently delete?	tick box		

if ticked, the deleted template will not go to the trash bin but will be permanently deleted from disk

Delete	button		

after selecting this button, the template given in the template field will be deleted. A **Yes-No** *pop-up is used to confirm that deletion is required.*

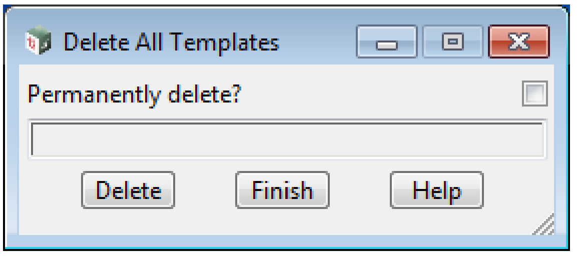
Go to the next section 20.2.6.2 Delete All Templates, or return to 20.2.5.5 Save Templates or 20.2.5 Template Utilities.

20.2.6.2 Delete All Templates

Position of option on menu: **Design =>Templates =>Delete =>Delete all templates**

The **Delete all** option will delete all templates in the working project. It does not delete templates that are in the working project area but not yet added to the project.

On selecting the **Delete all** option, the **Delete All Templates** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Delete	button		

after selecting this button, a **Yes-No** *pop-up is used to confirm that deletion is required. If it is, all templates in the project will be deleted from disk. Unless an error occurs, the panel will be removed.*

Permanently delete tick box

if ticked, the deleted templates will not go to the trash bin but will be permanently deleted from disk.

Return to 20.2.5.5 Save Templates or 20.2.5 Template Utilities.

20.3 Apply Functions

Position of menu: **Design =>Apply**

The **Apply** options are for producing design strings using **Templates** and **Modifiers** to create strings along a Reference or Reference and Hinge string - “Applying the Modifiers to a string”.

This method is used to quickly produce roads, canals, trenches, site batters etc.

When applying the modifiers and templates, design x-sections and strings are automatically produced, plus the cut and fill volumes for the design.

Creating a design in this way is the first job of the **Apply** options.

However, after applying modifiers and templates and examining the results, it is normally necessary to make changes to the Reference or Hinge strings, the modifiers and templates, or even the tin used for interfacing.

In all cases, the strings, sections and volumes created during the **Apply** are no longer valid and need to be replaced by new information.

Although this can be accomplished by deleting the incorrect models and replacing them with new ones by re-applying the template, this is time consuming if repeated over and over again.

Hence in ***12d Model***, the concept of an **Apply function** and **Recalc** was introduced.

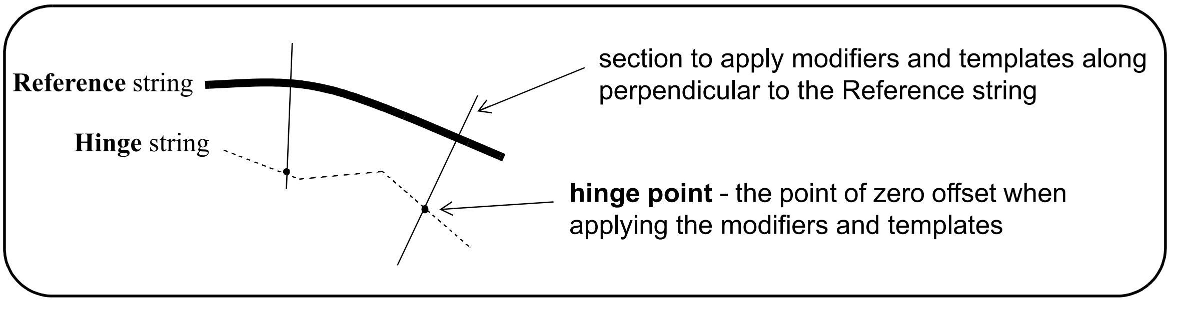
Basically, an **Apply Function** keeps track of all the information involved in applying Modifiers and Templates, and all the strings and models created during the **Apply Template/Apply MTF**.

If either the reference or Hinge strings, the Modifiers or Templates or the Tin are changed, the Apply Functions can be re-run and all the old information automatically deleted and replaced by the updated information.

NOTES ABOUT THE STRING CHOSEN TO APPLY MODIFIERS AND TEMPLATES TO

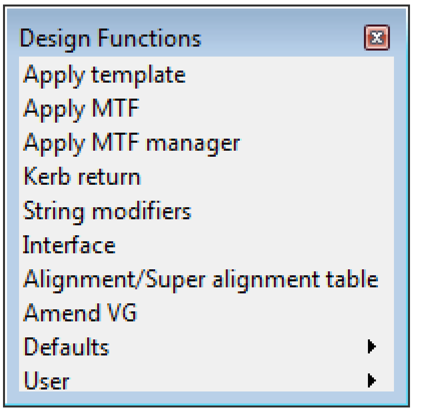
1. The **Apply** options create sections and strings using a **Reference** string to define the chainage and what is meant by perpendicular at each chainage, and a **Hinge** string from where the modifiers and templates links are defined. That is, the Hinge string defines the zero offset. The Modifiers and Templates define a point name and colour for each template link.

   Cross sections can be created with point names corresponding to the template links, and strings formed by joining the same named points from consecutive cross sections.



2. If vertical geometry does not exist for the entire length of the string that the templates are being applied to (either the Hinge string, or if there is no Hinge string then the Reference string), then the modifiers and templates are only applied to the section of the string where vertical geometry exists.
3. If the horizontal or vertical geometry has overlapping tangent points at any section of the string, then the option terminates without any calculations being made.
4. Any curves will be approximated by chords spaced at chainage interval given by the **section separation** value. If the chord-to-arc distance for the chords is greater than the chord/arc tolerance given in the **Apply template/Apply MTF** panel, then extra points are inserted around the curve so that the chord/arc tolerance is met.

The **Apply** walk-right menu is



Each option in this menu will now be described.

For the options:

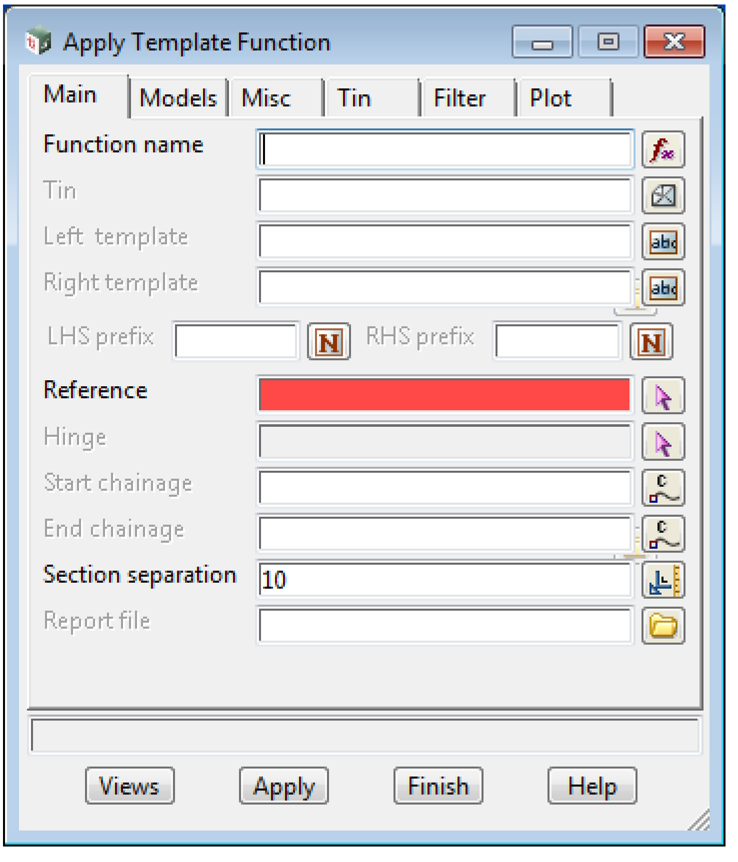
Apply template	20.3.1 Apply Template Function
Apply MTF	20.3.2 Apply MTF Function
Apply MTF manager	20.3.3 Apply MTF Manager - Create/Update
Kerb return	20.3.5 Kerb Return Function
String modifiers	20.3.6 String Modifiers Function
Interface	20.3.7 Interface Function
Alignment/Super alignment table	24.16.14 Alignment & Super Alignment Table 24 Drafting
Amend VG	20.3.8 Amend VG
Defaults	20.3.1 Apply Template Function

20.3.1 Apply Template Function

Position of option on menu: **Design =>Apply =>Apply**

Selecting **Apply** displays the **Apply Template Function** panel.

This panel is used to collect the information for applying a new Apply function, or to change the information for an existing Apply function.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Buttons at bottom

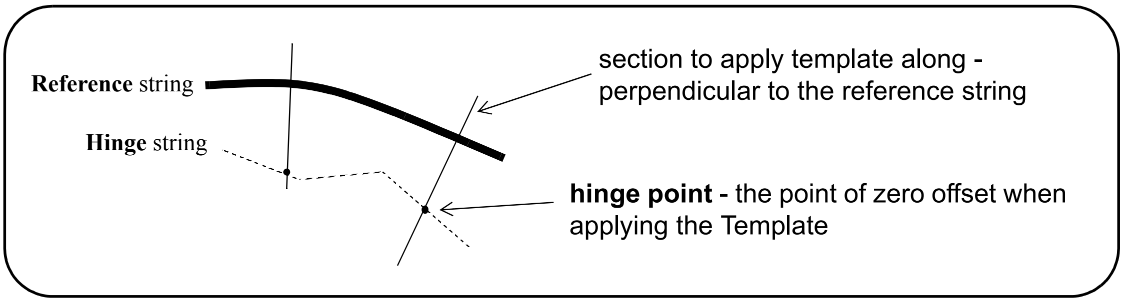
Views button

perspective views can be defined with respect to chainages on the string. This option is described in more detail in the section 20.3.1.1 Views

Apply button

apply the template information to the selected reference and hinge strings, between the start and end chainages.

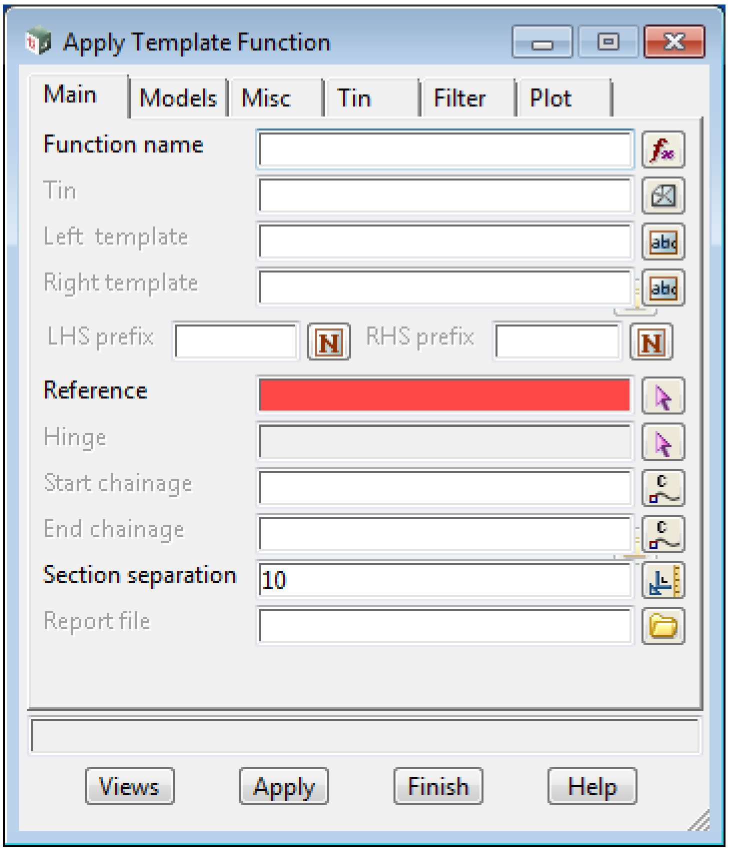
The cut, fill and balance volumes are also calculated and written to the message area.



For the information on each **Apply Template Function** tab, go to

[Apply Main tab]
[Apply Models tab]
[Apply Misc tab]
[Apply Tin tab]
[Apply Filter tab]
[Apply Plot tab]

Apply Main tab



Function name function box available template functions

name of the function to define the **apply** *for. If the function already exists and is picked from a pop-up or an <enter> is given at the end of the name, the information from the existing function will be placed in the appropriate panel fields.*

Tin tin box available tins

if non-blank, the name of the tin to calculate the cut/fill interfaces against at the end of the fixed part of the templates given in the **left** *and* **right template** *fields.*
If blank, then only the fixed part of the templates are used unless a tin is specified in the ***decisions*** *section of the templates.*

Left/right template template box available templates

name of the template to be applied to the left/right of the string. If a template is used on the left/right, the template definitions go from the hinge string out to the left/right.

LHS/RHS prefix input

*prefix/postfix (pre*post) to be applied to the left/right template string names. If pretext only, just give the text. If post text is required, precede it by a *.*

Reference string-select

the selected string is used to defined the meaning of chainage and bearing for each point in the apply. Sections are defined at right angles to the reference string at the appropriate chainage points on the reference string.

Hinge string-select

the reference string defines chainage and bearing but the templates are actually applied to the hinge string. For a given chainage, a line is taken at right angles to the point of that chainage on the reference string and extended until it cuts the hinge string. The templates are applied at that point on the hinge string along the direction of the line.
If no hinge string is selected, the reference string is also the hinge string.

Start/End chainage input

the reference string start/end chainage for applying the template. If blank, the **start/end chainage** *of the reference string is used.*

Section separation input 10.0

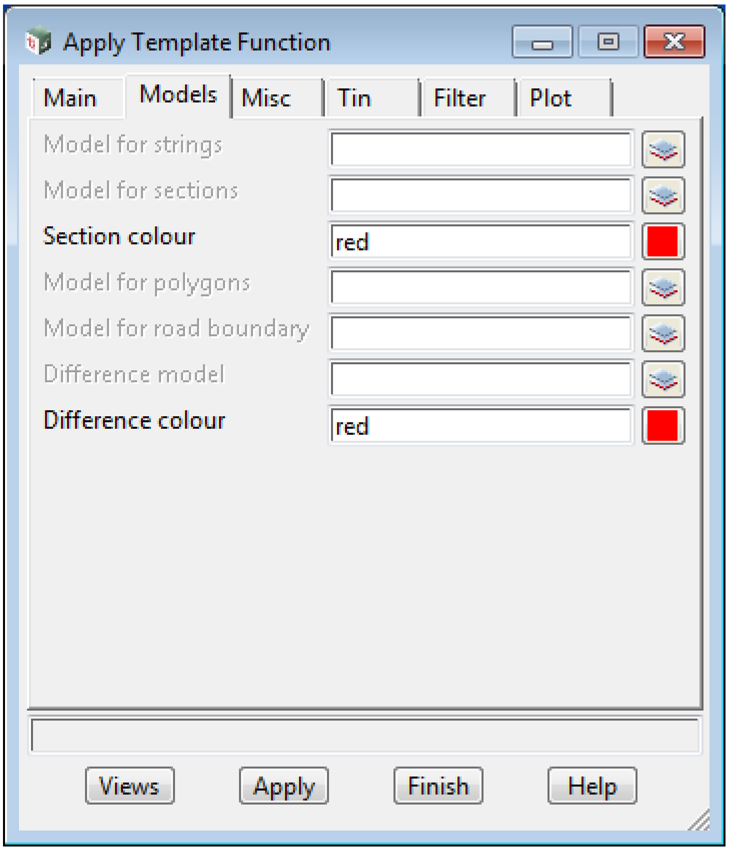
sections are created at right angles to points on the reference string that are the **section separation** *chainage distance apart.*

Report file file box *.rpt

if not blank, the name of the file to contain the volume report for the template calculations. If the file already exists, the report will be appended to the file.

If blank, no report is produced.

Apply Models tab



Model for strings model box available models

if not blank, the name of the model to contain the template and interface strings. The string colour is the template link colour.

If blank, the strings will not be stored.

Model for sections model box available models

if not blank, the name of the model to contain the design sections generated by the templates.
If blank, the sections will not be stored.

Section colour colour box default colour available colours

the colour for the sections strings

Model for polygons model box available models

if not blank, the name of the model to contain the polygons created for each link of the template. The polygon is given the colour of the template link.

If blank, the polygons will not be stored.

Model for road boundary model box available models

if not blank, the name of the model to contain the polygon created by joining the outside links of the road.

If blank, the polygon will not be stored.

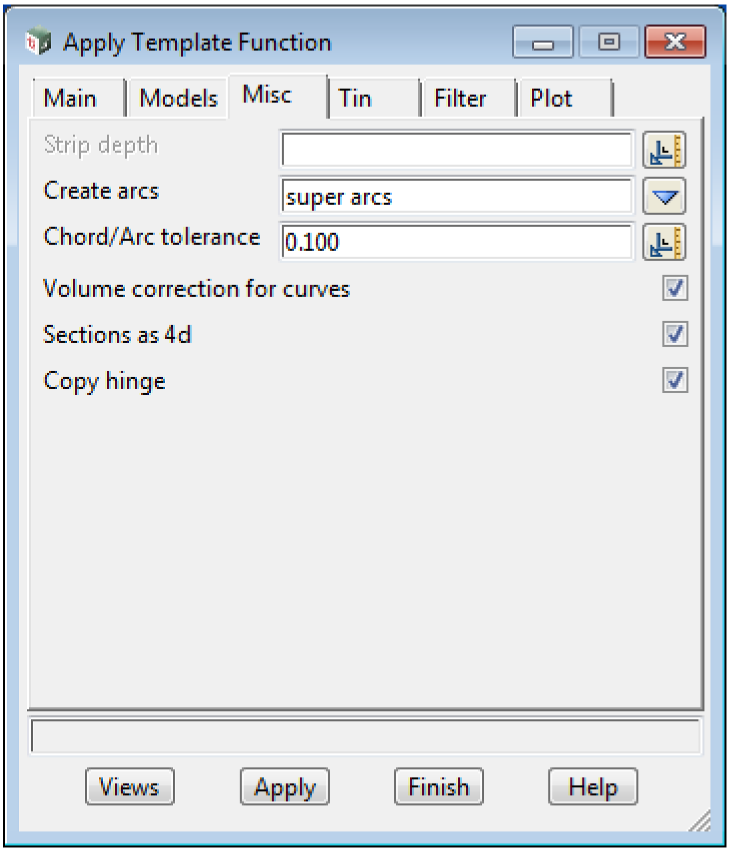
Difference model model box available models

if not blank, the sections which are the difference between the tin sections and the template x-sections are retained and placed in the model given in this field. If blank, the sections are not kept.

Difference colour colour box available colours

colour for the difference sections strings

Apply Misc tab



Strip depth input 0

the stripping depth to be used on the tin before the apply is done.

Create arcs choice box super arcs no arcs, alignment arcs, polyline arcs, super arcs

*if **no arcs**, the strings are created as 3d strings with no arcs.*
***alignment arcs**, the strings are created as alignment strings with arcs*
***polyline arcs**, the strings are created as polyline strings with arcs.*
***super arcs**, the strings are created as super strings with arcs.*

Chord/arc tolerance input default chord/arc tolerance

the chord to arc tolerance to use on the reference string for determining how many sections are created around horizontal curves.

Volume correction for curves tick box

if ticked, volume corrections are made when going around curves.

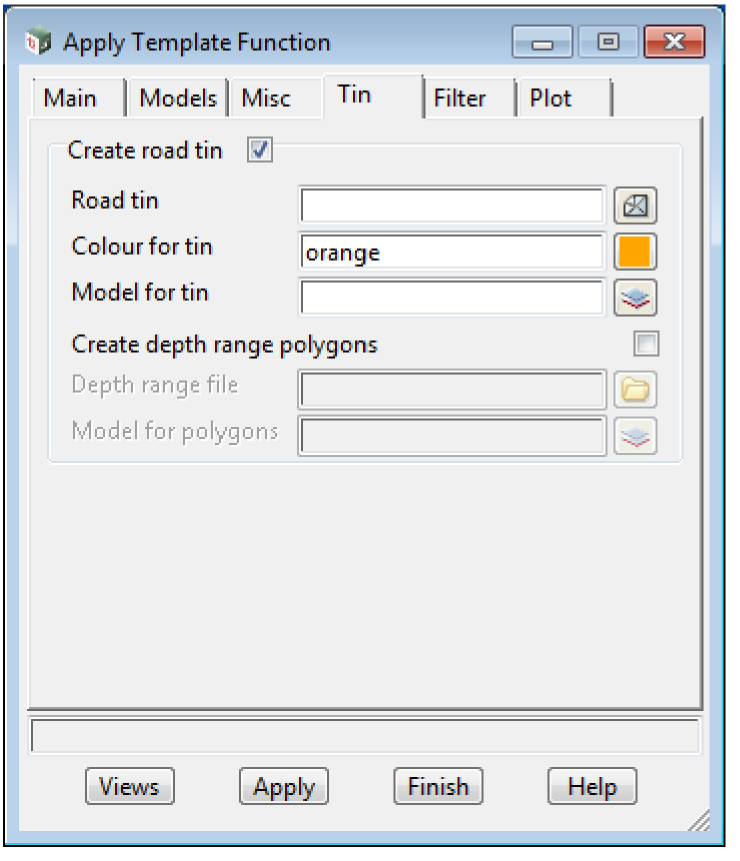
Sections as 4d tick box tick

if ticked, the section strings will be created as 4d super strings with the appropriate template string names as the vertex text at each vertex of the section. These are needed for boxing and some options on x-section plots.

Copy hinge tick box tick

if ticked, a string with points at the apply chainages is created on top of the hinge string.

Apply Tin tab



Create road tin tick box

if ticked, the fields in this tab are used to create a road tin

Road tin tin box available tins

name for the tin created from the design strings and sections

Colour for tin colour box available colours

colour of the road tin

Model for tin model box available models

model for the road tin

Create depth range polygons tick box

*if **ticked**, depth polygons are created*

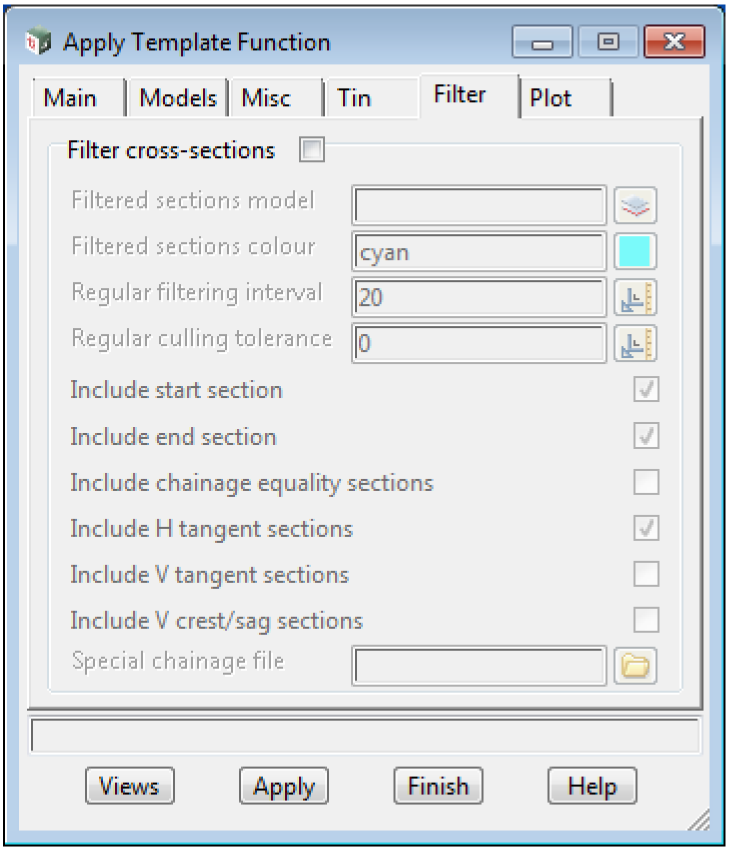
Depth range file file box *.drf files

depth range file used when creating polygons

Model for polygons model box available models

model for the depth polygons

Apply Filter tab



Filter cross-sections tick box

if ticked, the fields in this tab are used to filter cross sections

Filter sections model model box available models

model for the filtered cross sections

Filtered sections colour colour box available colours

colour of the filtered cross sections

Regular filtering interval input

regular interval to use for filtering the cross sections

Regular culling tolerance input 0

tolerance to use when selecting a cross section

Include start sections tick box

if ticked, a section at the start chainage is included even if the start chainage is not at a regular interval

Include end sections tick box

if ticked, a section at the end chainage is included even if the end chainage is not at a regular interval

Include chainage equality sections tick box

if ticked, include sections where there is a chainage equality

Include H/V tangent sections tick box

if ticked, a sections at the horizontal/vertical tangent points are included even if they are not at a regular interval

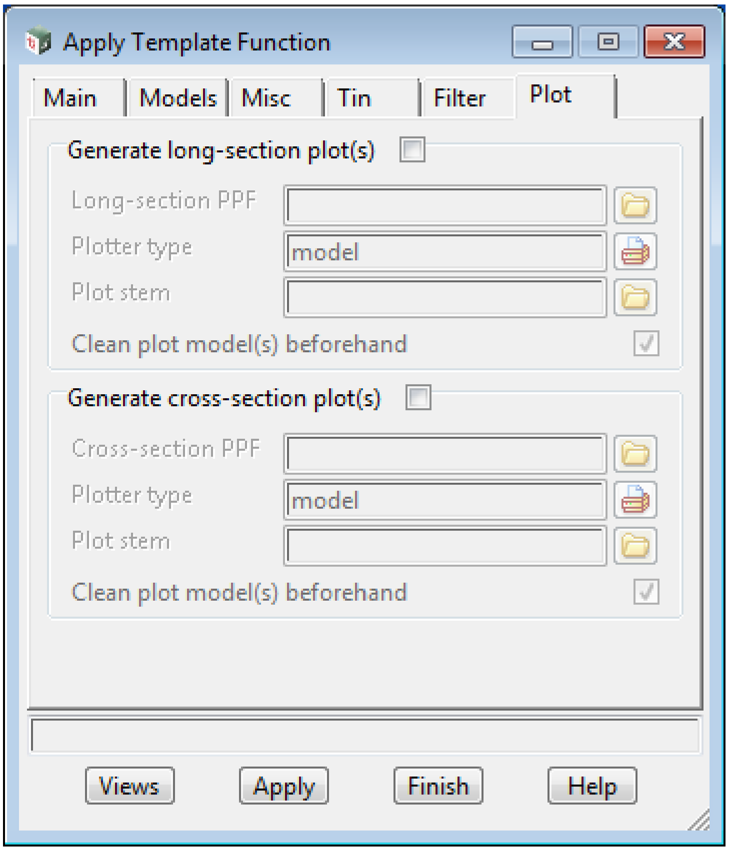
Include V crest/sag sections tick box

if ticked, a sections at the crest and sag points are included even if they are not at a regular interval

Special chainage file file box *.spc files

if non blank, a file of chainages to include sections at even if they are not a regular interval

Apply Plot tab



Generate long section plot(s) tick box

if ticked, long section plots are created

Long section PPF file box *.lplotppf files

binary ppf file to use for the long section plots

Plotter type plotter box model available plotters

type of plotter to use for the long section plots

Plot stem

the name to use for the plots - a number will be added when more than one page is produced

Clean plot model(s) beforehand tick box

*if ticked and the **plotter type is model**, the plot models are cleaned before the plots are created*

Generate cross section plot(s) tick box

if ticked, cross section plots are created

Cross section PPF file box *.lplotppf files *binary ppf file to use for the cross section plots*

Plotter type plotter box model available plotters

type of plotter to use for the cross section plots

Plot stem

the name to use for the plots - a number will be added when more than one page is produced

Clean plot model(s) beforehand tick box

*if ticked and the **plotter type is model**, the plot models are cleaned before the plots are created*

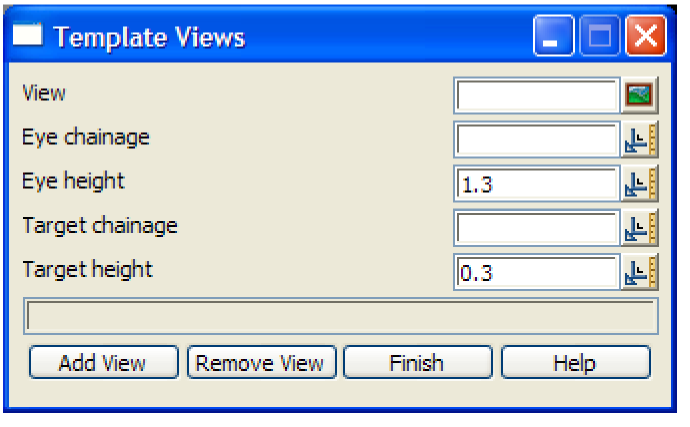
20.3.1.1 Views

This option locks perspective views to the reference string of the **Apply Template/Apply MTF** function,

It defines the eye and target co-ordinates for the perspective views in terms of the chainage and height above an existing reference string (as in the perspective view-ops option, string-walk) rather than entering the (x,y,z) eye and target co-ordinates.

If the string is modified, then the perspective eye and target points will also change.

Selecting **Views** fires up the **Template Views** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box		available views

the perspective view to define the eye and target points for.

Eye chainage input

the string chainage of the eye viewing point.

Eye height input 1.3

height of the eye viewing point above the string.

Target chainage input

the string chainage the target viewing point.

Target height input 0.3

height of the target viewing point above the string.

Add View button

add the view to the template function and then redraw the view using the above eye and target *parameters.*

Remove View button

remove the view from the Apply template/Apply MTF function.

20.3.2 Apply MTF Function

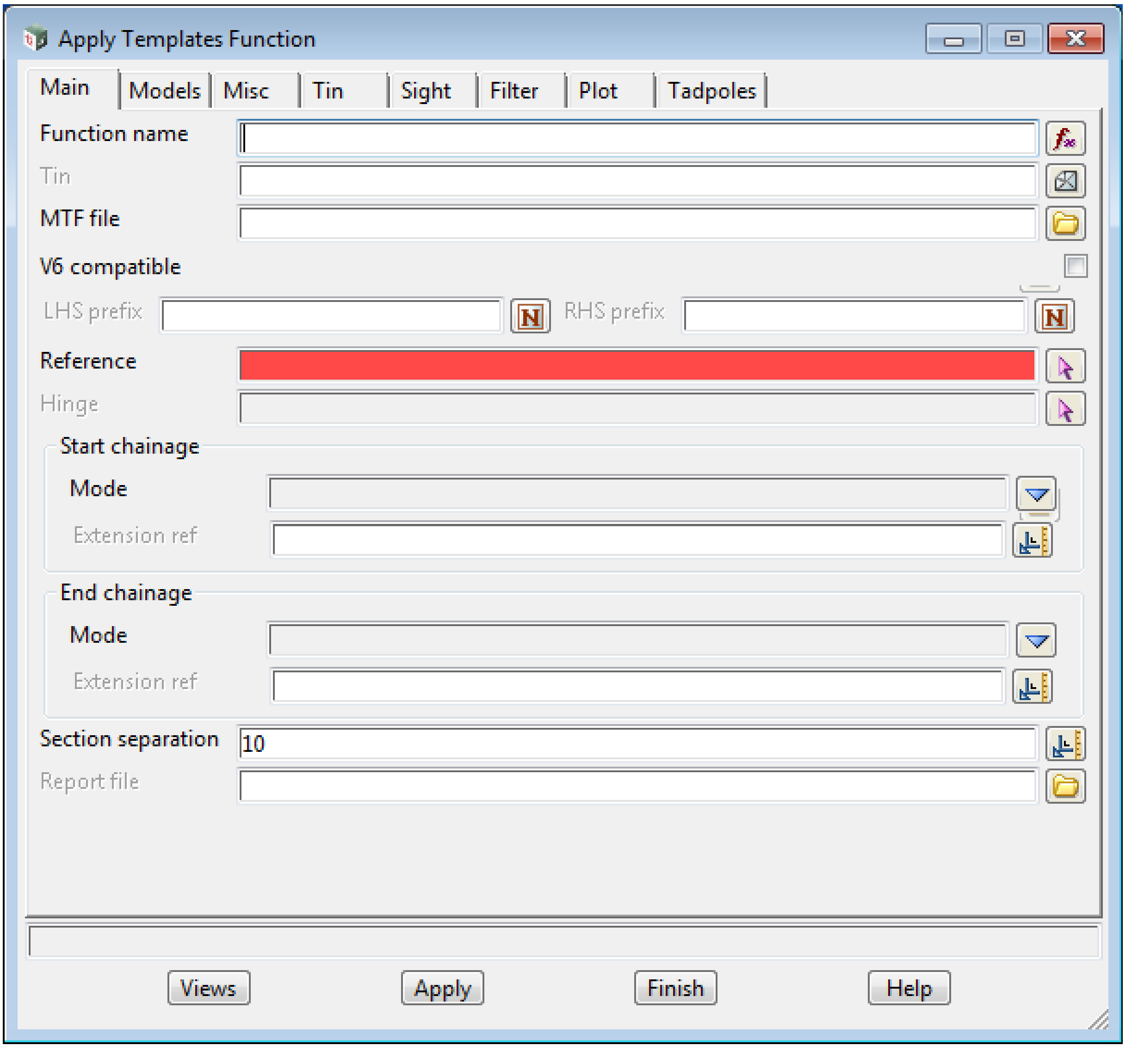
Position of option on menu: **Design =>Apply =>Apply MTF**

The **Apply MTF** option is used for more complex design work such as urban, intersection and roundabout design.

It includes defining different templates over different chainage ranges, modifiers for advanced control of the strings created by the **Apply MTF** strings, sections and trimeshes, stripping depths and boxing, a road tin and depth polygons, sights lines, tadpoles, plus long section and cross section plots.

For **Apply MTF**, the modifiers and templates are supplied in a file called the **Modifiers and Templates File** or ***MTF*** file. The definition of the ***MTF*** will follow after the description of the **Apply MTF** option.

Selecting **Apply MTF** displays the **Apply MTF Function** panel.



This panel can be used to create the information for applying a new template function or to modify the information for an existing template function.

Field Description Type Defaults Pop-Up

Buttons from bottom

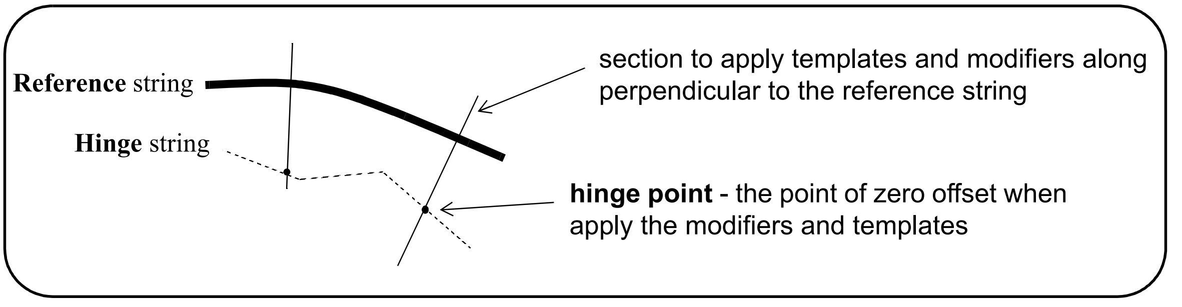
Views button

perspective views can be defined with respect to chainages on the string. This option is described in more detail in the section 20.3.1.1 Views.

Apply button

apply the modifiers and template information to the selected reference and hinge strings, between the start and end chainages.

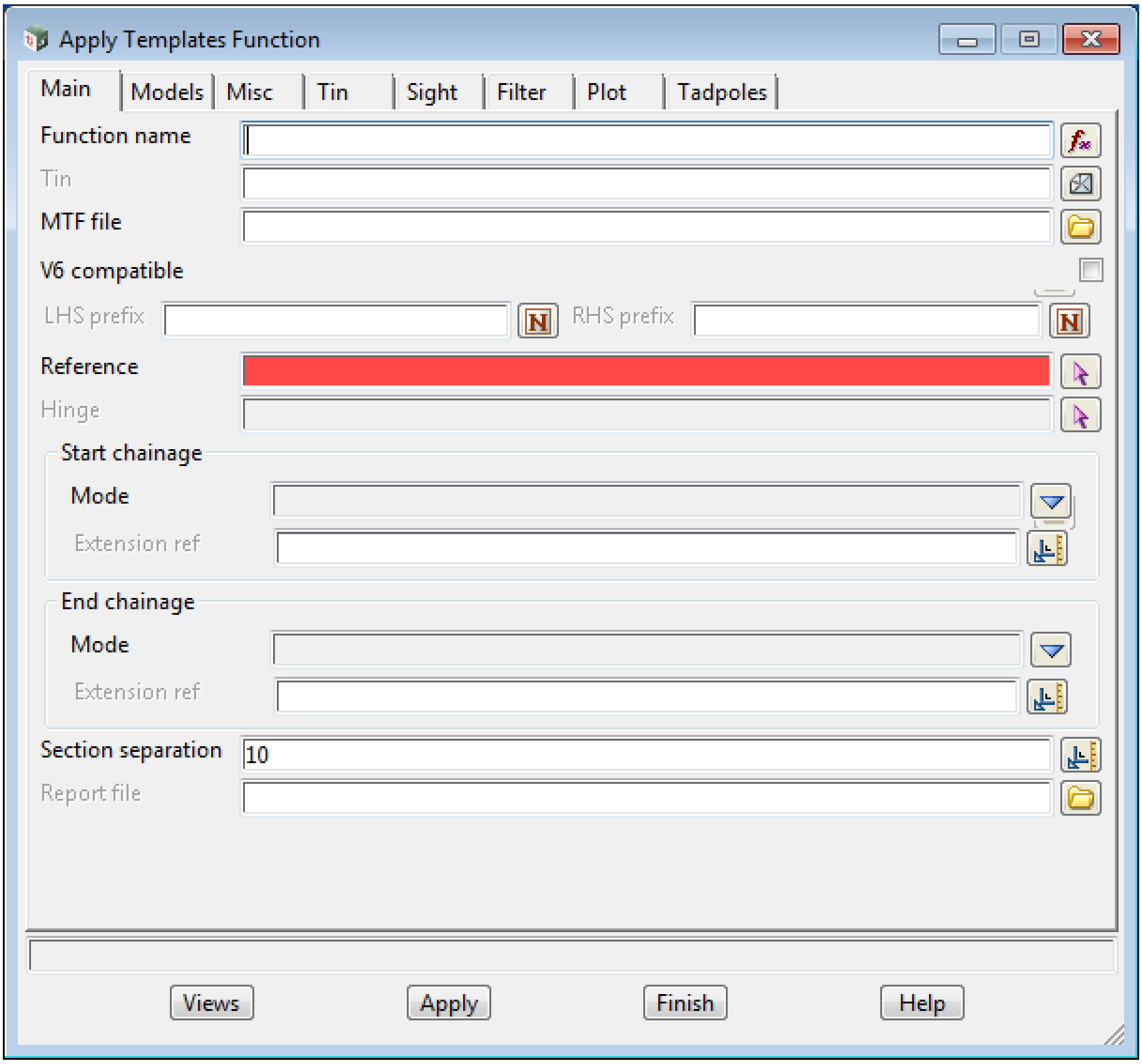
The cut, fill and balance volumes are also calculated and written to the message area and to the Output Window.



For the information on each **Apply MTF Function** tab, go to

- Apply MTF Main tab
- Apply MTF Models tab
- Apply MTF Misc tab
- Apply MTF Tin tab
- Apply MTF Sight tab
- Apply MTF Filter tab
- Apply MTF Plot tab
- Apply MTF Tadpoles tab

Apply MTF Main tab



Function name function box available template functions

name of the function to define the **Apply MTF** *for. If the function already exists and is picked from a pop-up or an <enter> is given at the end of the name, the information from the existing function will be placed in the appropriate panel fields.*

Tin tin box available tins

if non-blank, the name of the tin to calculate the cut/fill interfaces against at the end of the fixed part of the templates given in the template file.
If blank, then only the fixed part of the templates are used unless a tin is specified in the ***decisions*** *section of the templates.*

MTF file file box available mtf's, edit mtf

name of the file containing the definitions of where and how templates are applied and modified along the string, how boxing definitions are applied, stripping depths etc.
For more information on the MTF file, go to the section 21.1 The Modifiers and Templates File - MTF.

V6 compatible tick box

if ticked, the template is used as it was in **12d Model** *version 6*

LHS/RHS prefix input

*prefix/postfix (pre*post) to be applied to the left/right template string names. If pretext only, just give the text. If post text is required, precede it by a *.*

Reference string-select

the selected string is used to defined the meaning of chainage and bearing for each point in the apply. Sections are defined at right angles to the reference string at the appropriate chainage points on the reference string.

Hinge string-select

the Reference string defines chainage and bearing but the zero offset for the modifiers and templates is at the Hinge string.

For a given chainage, a line is taken at right angles to the point of that chainage on the Reference string and extended until it cuts the Hinge string. For the modifiers and templates, this point represents the zero offset. Hence the modifiers and templates are applied at that point on the Hinge string along the direction of the line.

If no Hinge string is selected, the Reference string is also used as the Hinge string.

Start/End chainage input

the Reference string start/end chainage for applying the MTF to. If blank, the **Start/End chainage** *of the Reference string is used.*

Start/End chainage choice box Start (ref)/End (ref)
Mode choice box Smart Chainge modes
Extension ref real box

the **Start/End chainage***,* **Mode** *and* **Extension ref** *are all used to define the start/end chainage on the Reference string for applying the MTF.*

If blank, the **Start/End chainage** *of the reference string is used*

For more information on Start/End Chainage Modes, see 21.2.1 MTF Hinge Modifiers

Section separation input 10.0

sections are created at right angles to points on the Reference string that are the **Section separation** *chainage distance apart.*

Report file input *.rpt

if not blank, the name of the file to contain the volume report for the template calculations. If the file already exists, the report will be automatically over written.

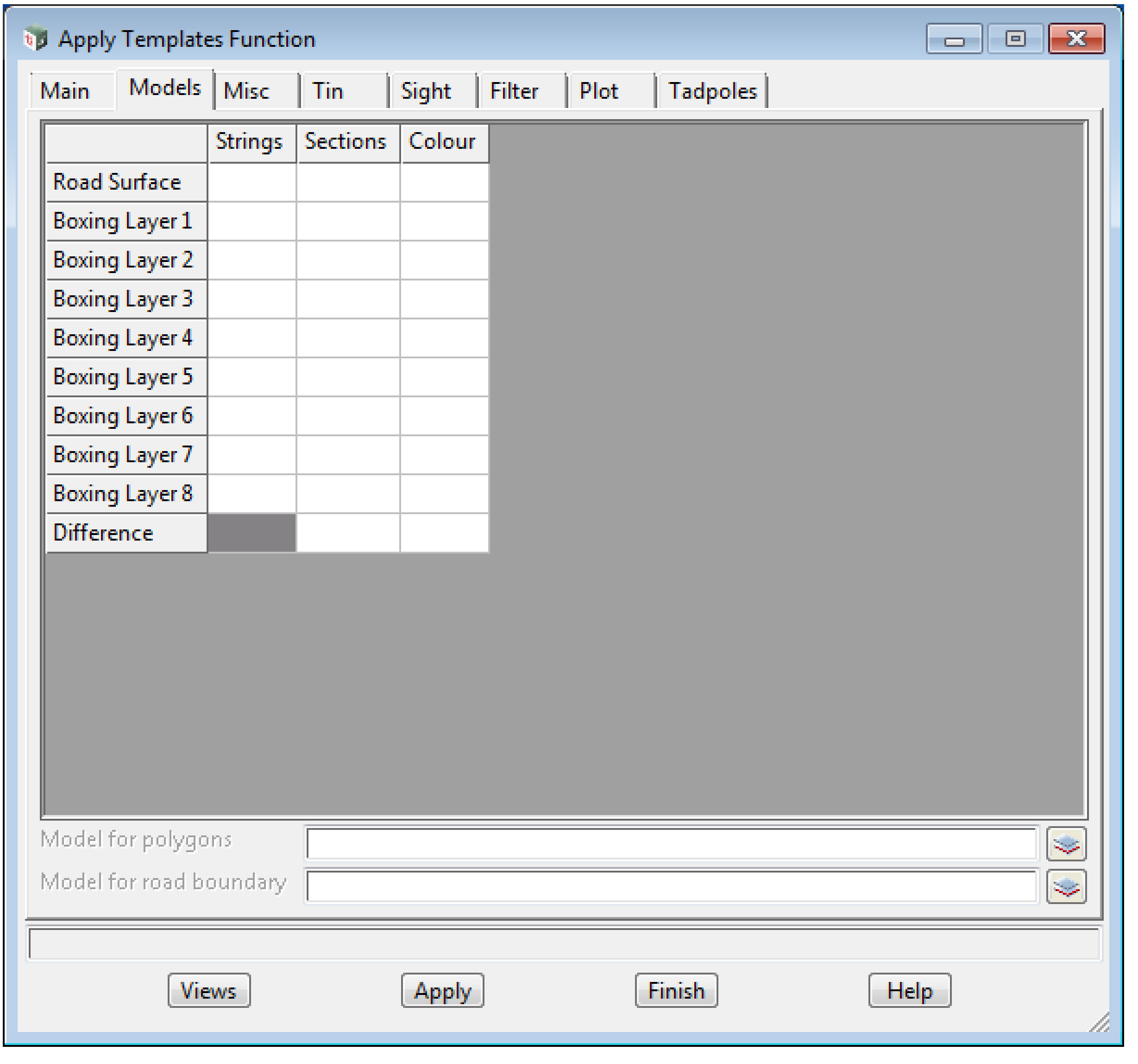
If blank, no report is produced.

Also if boxing is included in the MTF file definition, the volumes in the report can include not only the volumes from the stripped natural surface tin to the design, but also the interboxing layer volumes and the volumes from the last boxing layer to the design and to the stripped natural surface.

Apply MTF Models tab

the Models tab consists of a grid for defining the model for the design strings forming the road surface and design cross section, up to eight layers of boxing and models for the difference sections between the design and the natural surface.

The last boxing layer is also referred to as the ***Subgrade*** *layer.*



Road Surface:

Strings model box available models

if not blank, the name of the model to contain the design and interface strings. The default string colours are the template link colours but that can be overridden by a Map file.

If blank, the strings will not be stored.

Sections model box available models

if not blank, the name of the model to contain the design sections generated by the apply.

If blank, no sections are stored.

Colour colour box available colours

the colour for the design cross sections.

Boxing Layer 1-8:

if not blank, boxing strings and sections will be created for that layer.

*The Boxing Layers are used from **Boxing Layer 1** down until the first blank row.*

*The last boxing layer is also referred to as the **subgrade** layer.*

For more information on boxing, see 21.6 What is Boxing?.

Strings model box available models

if not blank, the name of the model to contain the boxing strings for this layer.

If blank, the strings will not be stored.

Sections model box available models

if not blank, the name of the model to contain the boxing sections generated by the templates and boxing rules. If blank, the sections will not be stored.

Colour colour box available colours

colour for the boxing sections for this layer

Difference

Sections model box available models

if not blank, the sections which are the difference between the tin sections and the template x-sections are retained and placed in the model given in this field.

If blank, the sections are not kept.

Colour colour box available colours

colour for the difference sections strings

Model for polygons model box available models

if not blank, the name of the model to contain the polygons created for each link of the template. The polygon is given the colour of the template link.

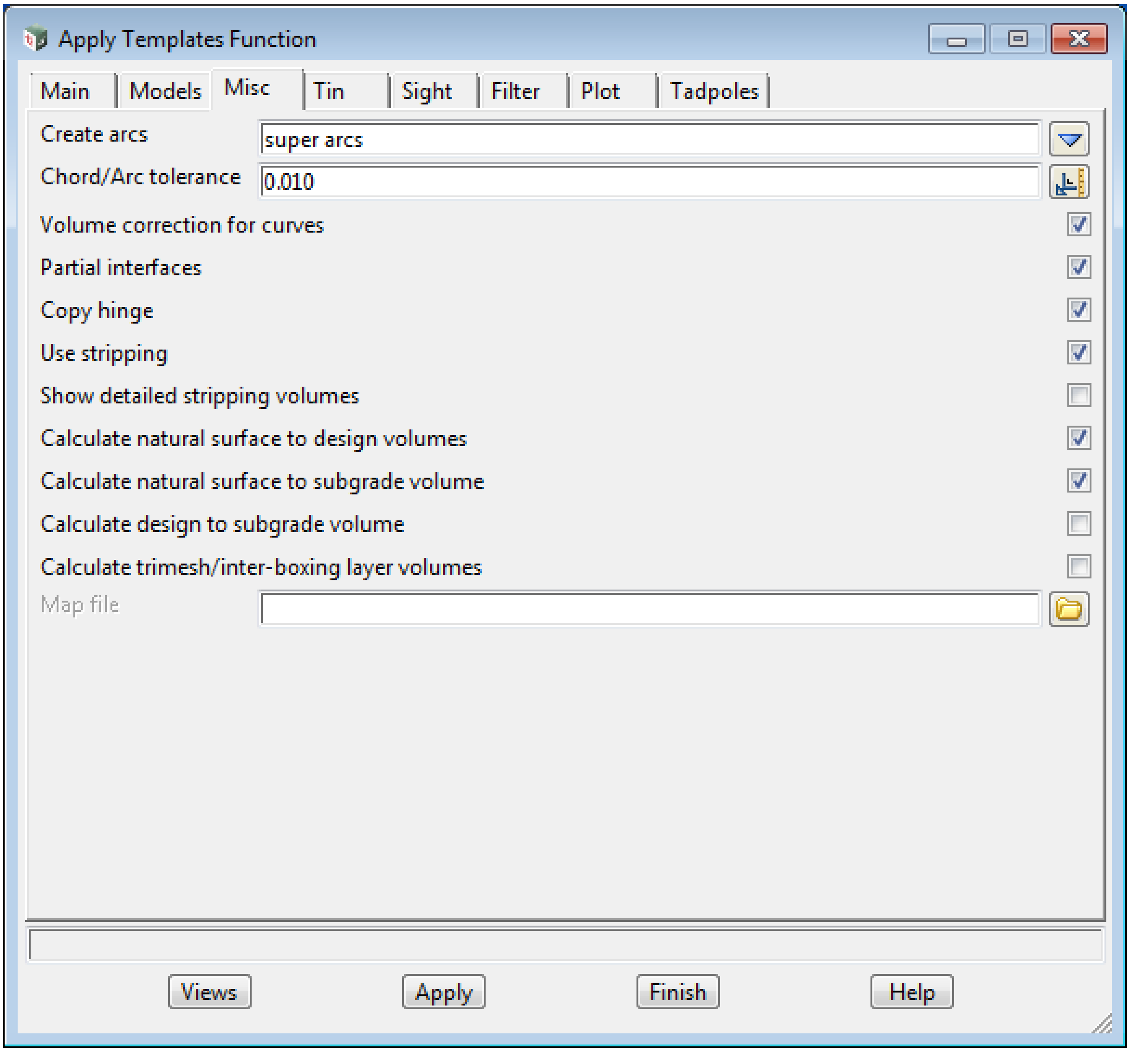
If blank, the polygons will not be stored.

Model for road boundary model box available models

if not blank, the name of the model to contain the polygon created by joining the outside links of the road.

If blank, the polygon will not be stored.

Apply MTF Misc tab



Create arcs choice box super arcs no arcs, alignment arcs, polyline arcs, super arcs

*if **no arcs**, the strings are created as 3d strings with no arcs.*
***alignment arcs**, the strings are created as alignment strings with arcs*
***polyline arcs**, the strings are created as polyline strings with arcs.*
***super arcs**, the strings are created as super strings with arcs.*

Chord/arc tolerance input default chord/arc tolerance

the chord to arc tolerance to use on the reference string for determining how many sections are created around horizontal curves.

Volume correction for curves tick box

if ticked, volume corrections are made when going around curves.

Partial interfaces tick box tick

if ticked, an interface string is not produced when no intersection is made with the tin. The interface may then be broken into a number of interface strings.

If not ticked, a yellow segment is created in the interface string when no intersection is made with the tin. Only one interface string will then be produced.

Copy hinge tick box tick

if ticked, a string with points at the apply chainages is created on top of the hinge string.

Use stripping tick box tick

if ticked, the stripping depths in the mtf file are used.

Show detailed stripping volumes tick box

if ticked, end area stripping volumes are written to the report file

Calculate natural surface to design volumes tick box tick

if ticked, end area volumes between the natural surface and the design strings are written to the report file. If stripping is used, then the cut and fill areas and volumes are calculated from the design to the stripped natural surface.

Calculate natural surface to subgrade volume tick box tick

if ticked, end area volumes between the natural surface to the subgrade (the last boxing layer) are written to the report file. If stripping is used, then the cut and fill areas and volumes are calculated from the last layer of boxing, to the stripped natural surface.

Calculate road to subgrade volumes tick box tick

if ticked, end area volumes between the design strings and the subgrade (the last boxing layer) are written to the report file.

Calculate inter-boxing layer volumes tick box tick

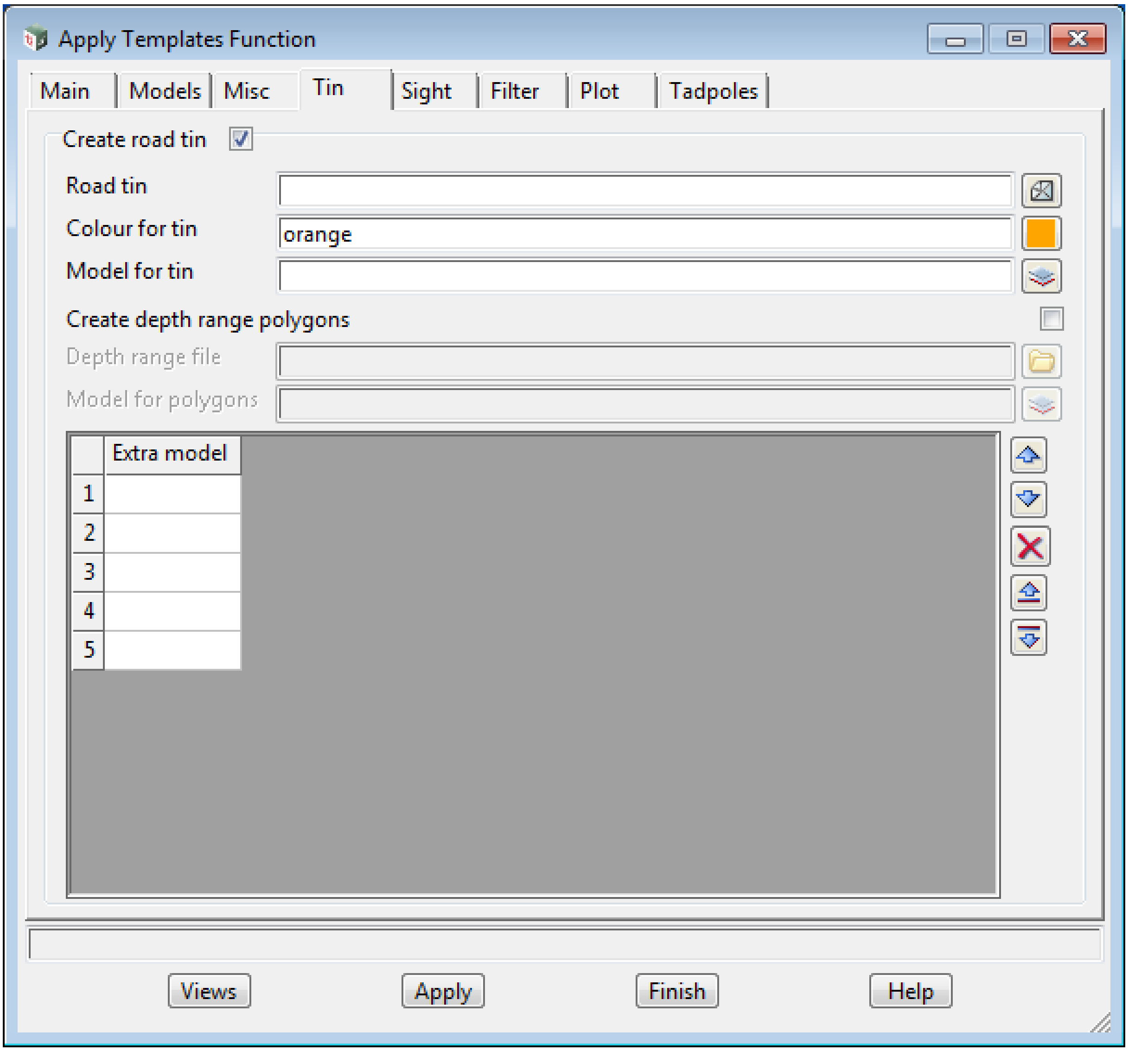
if ticked, end area volumes between each of the boxing layers are written to the report file.

Map file file box *.mapfile files

if ***non blank****, the given map file is applied to all the design strings created by the* ***Apply MTF.*** *The map file overrides any default colours from the templates. The map file can colour strings, apply linestyles, apply hatch patterns to polygons, apply extrudes to super strings etc. For more details on the map file, see* 8.8.1 Create/Edit a Map File.

Apply MTF Tin tab

the ***Tin*** *tab controls the creation of a design tin and depth polygons*



Create road tin tick box

if ticked, the fields in this tab are used to create a road tin

Road tin tin box available tins

name for the tin created from the design strings and sections

Colour for tin colour box available colours

colour of the road tin

Model for tin model box available models

model for the road tin

Create depth range polygons tick box

if ticked, depth polygons are created

Depth range file file box *.drf files

depth range file used when creating polygons

Model for polygons	model box	available models

model for the depth polygons

Extra Models grid

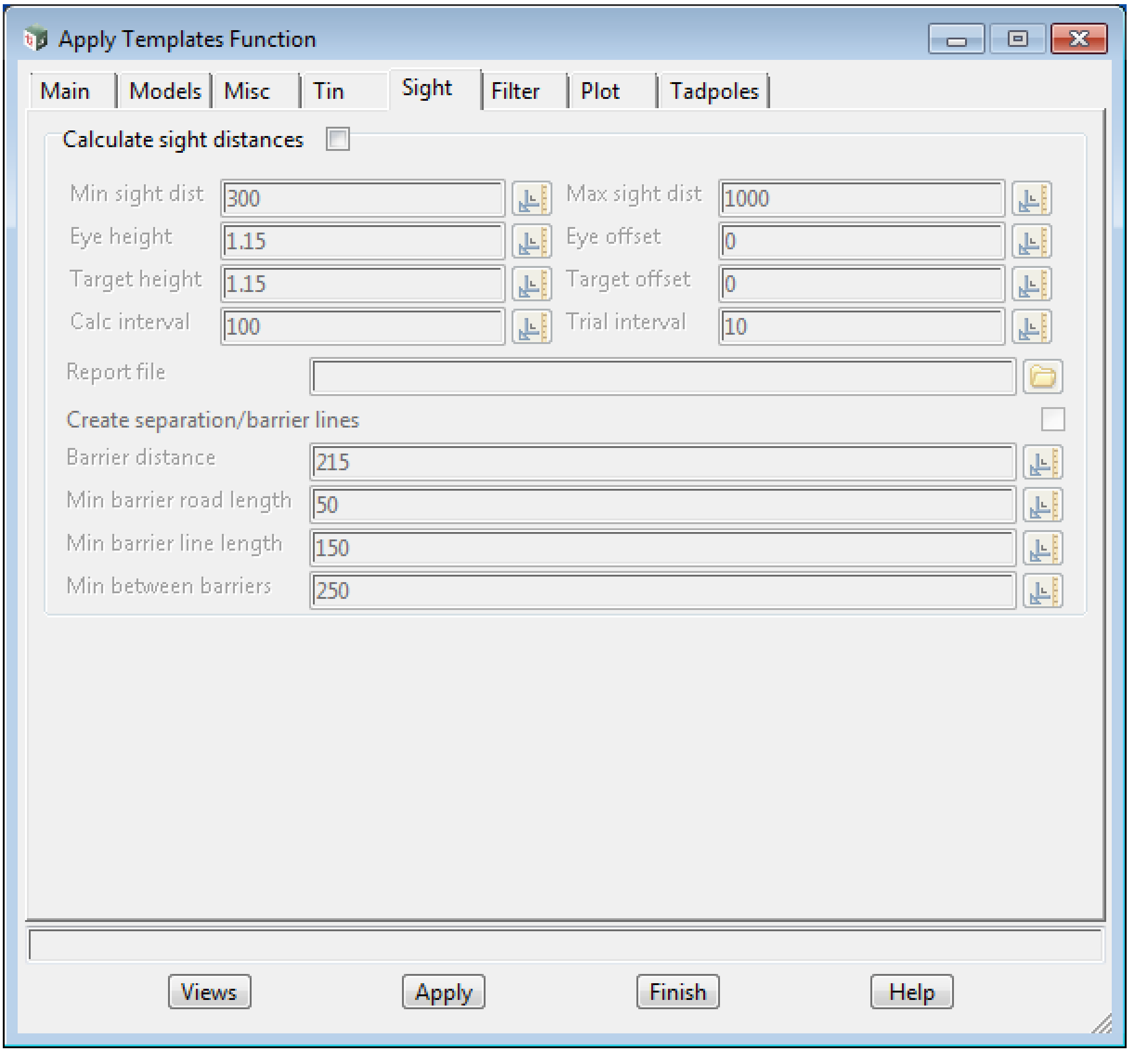
extra models to be included in creating the road tin

Extra Model	model box in column	available models

name of a model to include in the road tin

Apply MTF Sight tab

*the **Sight** tab controls the reporting of sight distance lines between the reference line and a super tin formed from the natural surface tin and the design tin, and barrier lines on the reference line if it is a super alignment. For more information on sight distance calculations and for the option that will also create sight lines/no-lines and take bridge tins into consideration, go to the section* 20.8.3 Stopping Distance.



Calculate sight distance tick box

if ticked, sight distance lines are created and barrier lines calculated and applied to the reference string if it is a super alignment.

Min sight distance input 100

minimum chainage distance to use for placing the test target point.

Max sight distance input 3000

maximum chainage distance to use for placing the test target point.

Eye height input 1.3

height of the eye point above the picked string

Eye offset input 1.3

offset of the eye point from the picked string.

Target height input 0.3

height of the test target point above the picked string

Target offset input 1.3

offset of the target point from the picked string.

Calc interval input 20

chainage increment to move the eye point for the next sight distance calculation.

Trial interval input 10

chainage increment to move to test target point.

Report file file box *.rpt files

if non blank, a sight distance report is produced and written out to this file name.

Create separation/barrier lines tick box

if ticked, and the reference string is a super alignment, barrier and separation lines are created and defined in the super alignment.

Barrier distance input

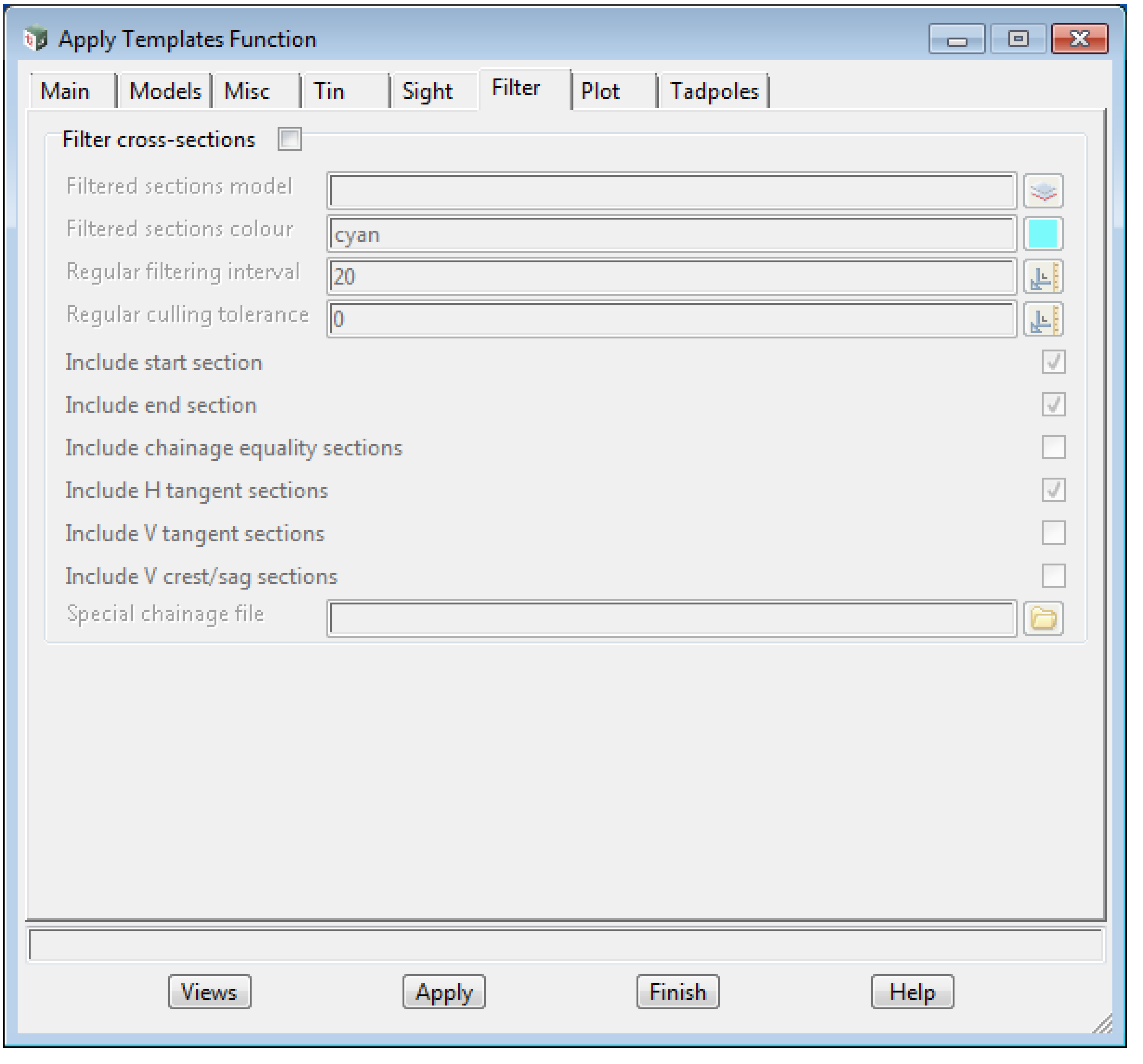
Min barrier road length input

Max barrier road length input

Min between barriers input

Apply MTF Filter tab

*the **Filter** tab controls the filtering of cross sections*



Filter cross-sections tick box

if ticked, the fields in this tab are used to filter cross sections

Filter sections model model box available models

model for the filtered cross sections

Filtered sections colour colour box available colours

colour of the filtered cross sections

Regular filtering interval input

regular interval to use for filtering the cross sections

Regular culling tolerance input 0

tolerance to use when selecting a cross section

Include start sections tick box

if ticked, a section at the start chainage is included even if the start chainage is not at a regular interval

Include end sections tick box

if ticked, a section at the end chainage is included even if the end chainage is not at a regular interval

Include H tangent sections tick box

if ticked, a sections at the horizontal tangent points are included even if they are not at a regular interval

Include V create/sag sections tick box

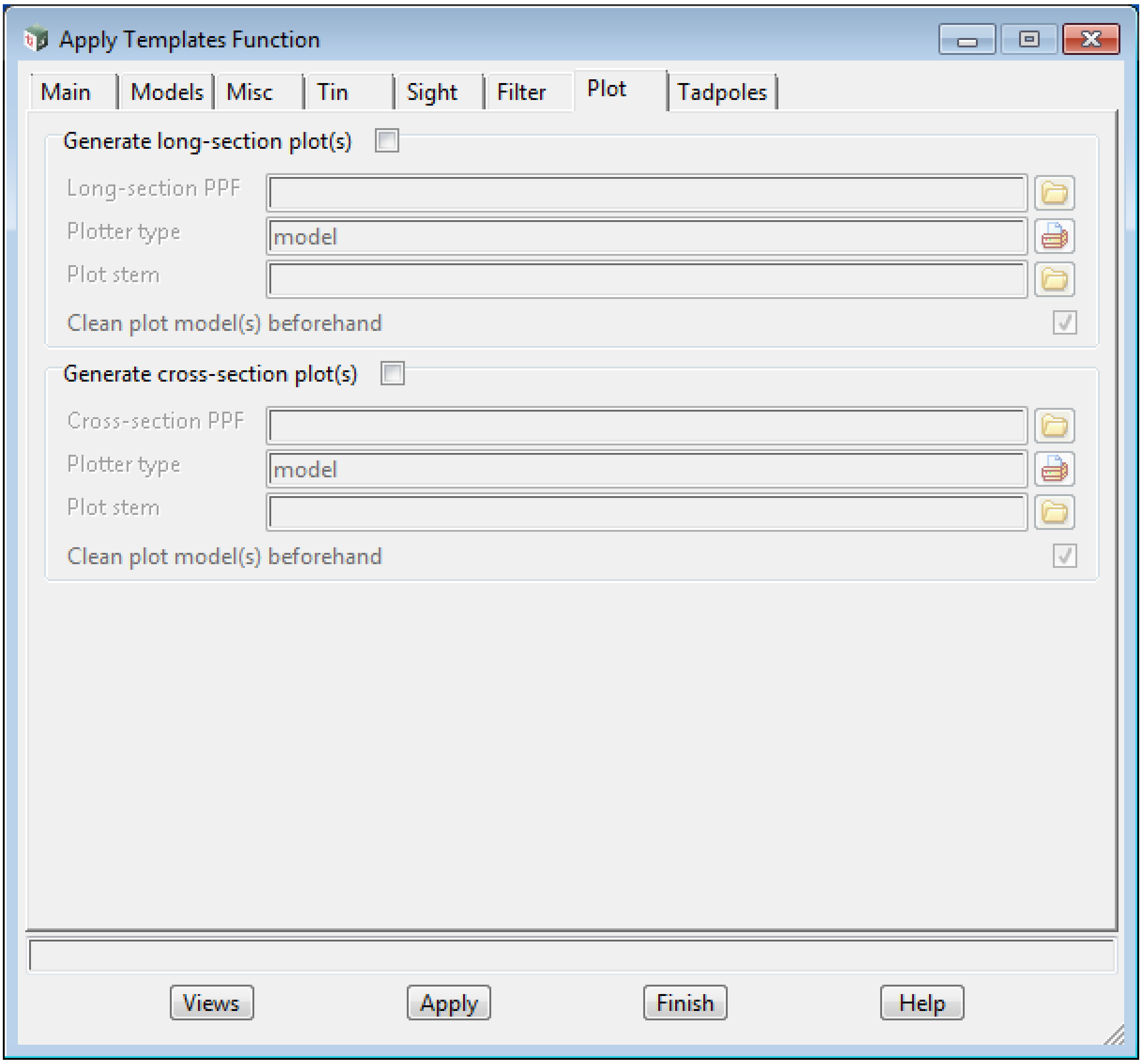
if ticked, a sections at the crest and sag points are included even if they are not at a regular interval

Special chainage file file box *.spc files

if non blank, a file of chainages to include sections at even if they are not at regular intervals

Apply MTF Plot tab

*the **Plot** tab controls the generation of long section and cross section plots*



Generate long section plot(s) tick box

if ticked, long section plots are created

Long section PPF file box *.lplotppf files

binary ppf file to use for the long section plots

Plotter type plotter box model available plotters

type of plotter to use for the long section plots

Plot stem

the name to use for the plots - a number will be added when more than one page is produced

Clean plot model(s) beforehand tick box

*if ticked and the **plotter type is model**, the plot models are cleaned before the plots are created*

Generate cross section plot(s) tick box

if ticked, cross section plots are created

Cross section PPF file box *.lplotppf files

binary ppf file to use for the cross section plots

Plotter type plotter box model available plotters

type of plotter to use for the cross section plots

Plot stem

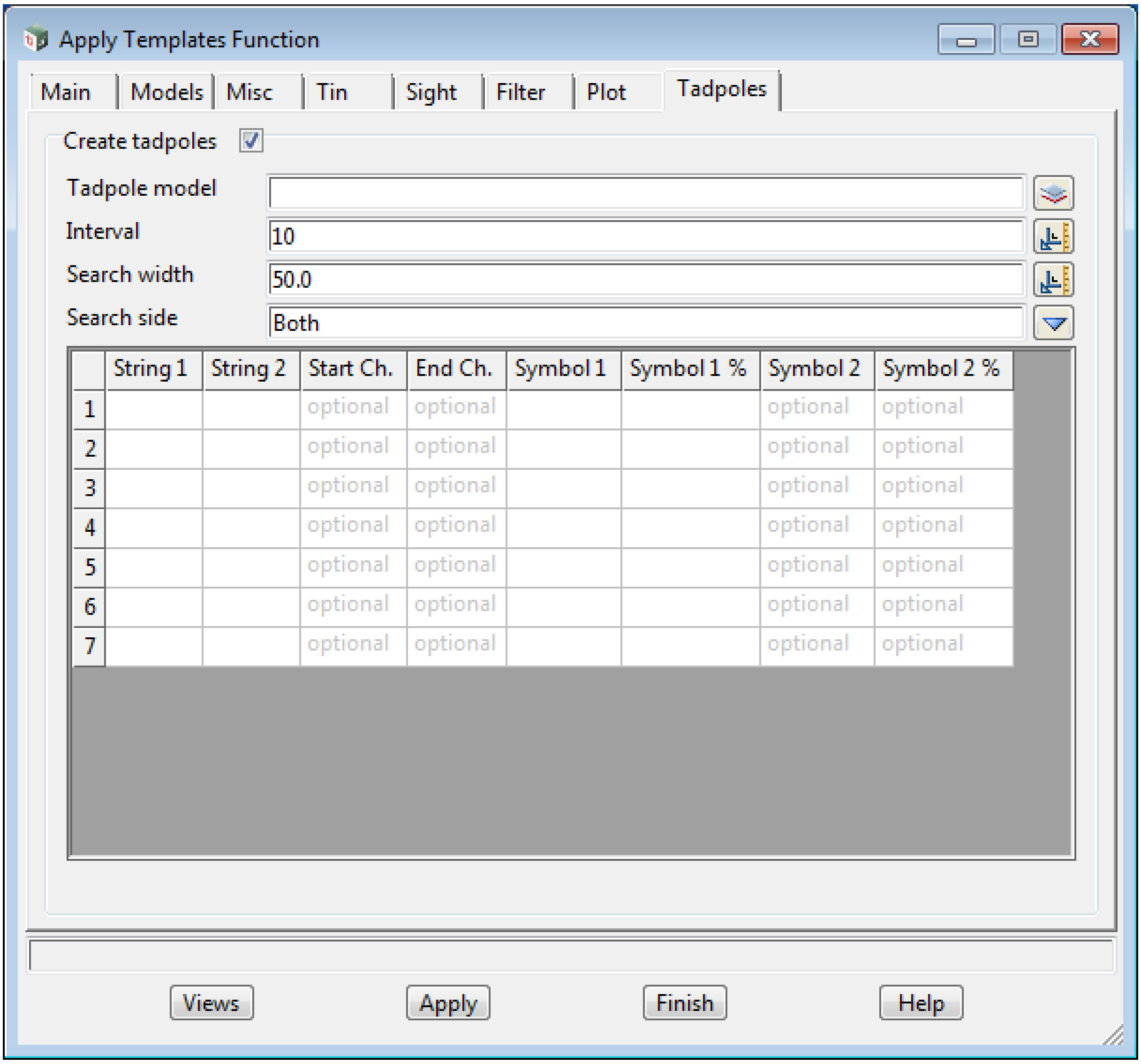
the name to use for the plots - a number will be added when more than one page is produced

Clean plot model(s) beforehand tick box

*if ticked and the **plotter type is model**, the plot models are cleaned before the plots are created*

Apply MTF Tadpoles tab

the ***Tadpole*** *tab controls the generation of batters ticks and tadpoles (hachure notation). For more information see the section* 24.5 Create Cut/Fill Symbols.



Create tadpoles tick box

if ticked, cut/fill symbols are created between given strings

Tadpole model model box available models

model for the created cut-fill symbols

Interval input 10

chainage distance between the symbols

Search width input 50

maximum distance to search to cut the strings for creating symbols. This is important for cases such as a bend in a road where multiple sets of string may then exist even when only searching out on one side of the road

Search side choice box both both, left side only right side only *side of the reference string to search for strings to apply symbols to.*

String 1 input

name of the point on the template to start symbol.

String 2 input

name of the point on the template to stop symbol.

Start Ch. input

if blank, use the start chainage of the Apply MTF.

If non blank, the chainage to start applying this symbol to.

End Ch. input

if blank, use the end chainage of the Apply MTF.

If non blank, the chainage to stop applying this symbol to.

Symbol 1 symbol data

definition of the symbol to create at the given interval, or at twice the given interval if **Symbol 2** *is defined*

Symbol 1 % input

if non-blank, percentage of the distance between String 1 and String 2 to apply the symbol.

If blank, the symbol is applied to the full distance between String 1 and String 2 (i.e. 100%)

Symbol 2 symbol data

definition of a second symbol to create at twice the given interval

Symbol 2 % input

if non-blank, percentage of the distance between String 1 and String 2 to apply the symbol.

If blank, the symbol is applied to the full distance between String 1 and String 2 (i.e. 100%)

20.3.3 Apply MTF Manager - Create/Update

Position of option on menu: **Design =>Apply =>Apply MTF manager**

This option is part of a process to manage your Apply MTF Functions in conjunction with the Apply MTF Defaults.

The Manager is in part an extension of the model naming convention within the defaults.

All models, except ROAD SURFACE STRINGS and ROAD SURFACE SECTIONS, can be toggled on and off when creating an Apply MTF Function,

e.g.

BOXING STRINGS

BOXING SECTIONS

POLYGONS AND BOUNDARY POLYGONS

DESIGN TIN

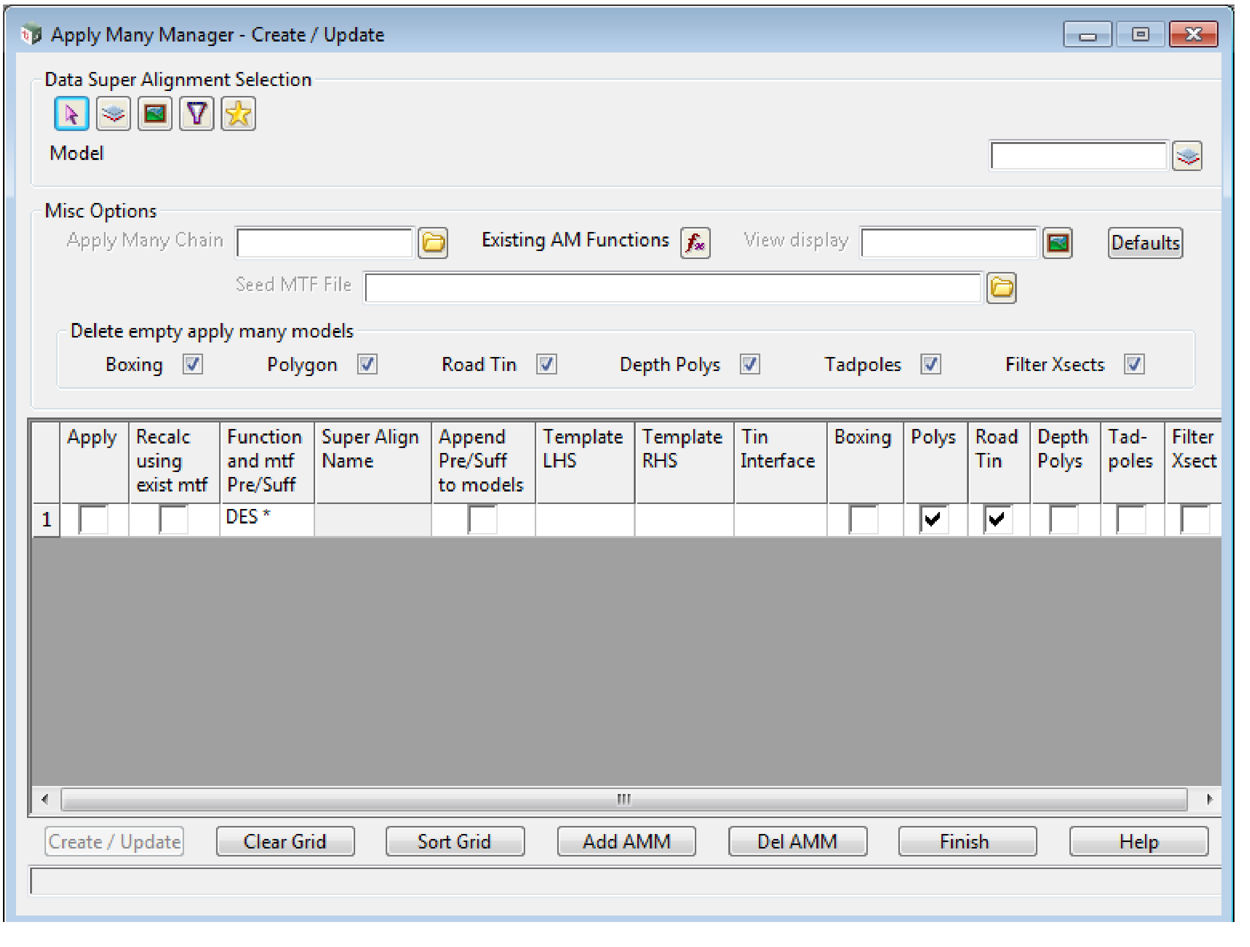
DEPTH POLYGONS

FILTER SECTIONS

TADPOLES

The Manager also allows one project model for ROAD SURFACE STRINGS, ROAD SURFACE SECTIONS, POLYGONS AND BOUNDARY POLYGONS and TADPOLES by using a new syntax in the Apply MTF Defaults.

Selecting the **Apply MTF manager** option displays the **Apply MTF Manager - Create / Update** panel.



The fields and buttons used in this panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Data source type

data selection type - for a full description go to 4.19.3 Data Source.

Misc Options

Apply MTF Chain file box available *.chain files

If non-blank entry used as chain file of all the Apply MTF Functions that are created at the time of process...optional

Existing AM Functions function box available functions

This option will give you a list of all the existing AM Functions in the project, as a quick reference when creating new ones

View Display view box available views

If non-blank entry used as view onto which design strings and sections models will be automatically added, for ease of viewing...optional

Defaults button

*If selected a separate panel will appear that allows you to set up defaults that relate to how the main panel appears. Selecting **Defaults** brings up the **Defaults for startup** panel. For more information on the **Defaults for startup** panel, go to* 20.3.4 Defaults For Startup

Seed MTF File file box

If non-blank, entry used as a seed from which any default names for the application of future Superelevation/Widening will be set

Delete empty Apply MTF models

Boxing tick box ticked

If ticked then any empty boxing models will be deleted when an existing AM is updated

Polygon tick box ticked

If ticked then any empty polygon models will be deleted when an existing AM is updated

Road Tin tick box ticked

If ticked then any empty road tin models will be deleted when an existing AM is updated

Depth Polys tick box ticked

If ticked then any empty depth polys models will be deleted when an existing AM is updated

Tadpoles tick box ticked

If ticked then any empty tadpoles models will be deleted when an existing AM is updated

Filter Xsects tick box ticked

If ticked then any empty filter xsects models will be deleted when an existing AM is updated

Grid

Apply tick box not ticked

Set to ticked once a SA selection string is accepted

*If ticked and the **<Create/Update>** is selected, then the Apply MTF will be created or an existing one recalced*

*If ticked and the **<Del AMM>** button is also selected, then the Apply MTF Manager reference to that alignment will be deleted*

*If not ticked and the **<Create/Update>** is selected, then the Apply MTF will NOT be created or an existing one recalced*

Note: It is up to the user to delete the actual Apply MTF Function and all associated data including the mtf

Recalc using exist mtf tick box

This tick box is an automatic indication of whether or not a Reference Super Alignment has an Apply MTF Function already linked to it

There is no need for the user to use this box when creating or updating an Apply MTF Function

It will be unticked if the reference has no Apply MTF Function attached

In this case templates are required for Left, Right or both

It will be ticked if the reference does have one or more Apply MTF Functions attached

In this case the existing AM will be updated using the existing MTF and reflect any changes in regards to models required

Function & mtf Pre/Suff input DES*

If non-blank, entry used as a prefix or suffix in conjunction with the Apply MTF Reference Super Alignment Name

Example:

Reference SA name:	MC00
Pre/Suff:	DES*
Apply MTF Function Name:	DES MC00
Reference SA name:	MC00
Pre/Suff:	* DES
Apply MTF Function Name:	MC00 DES

If blank, the Apply MTF Reference Super Alignment Name is used only

Apply MTF Function Name: MC00

Super Align Name input

Automatically populated, after the SA reference is selected

The Super Alignment Name entry cannot be changed as it comprises part of the Apply MTF Function name....refer above

Append Pre/Suff to models tick box not ticked

If ticked then the prefix or suffix will be combined with the SA name (refer above) and passed down to any default extensions from the Apply MTF Defaults

*refer **Design=>Apply=>Apply MTF Defaults***

Model Format for Apply MTF Defaults:

Road Surface Strings e.g.

"DESIGN *" used in conjunction with a Pre/Suff of DES * and a SA name MC00, would result in a strings model called "DES DESIGN MC00"

"* DESIGN" used in conjunction with a Pre/Suff of DES * and a SA name MC00, would result in a strings model called "DES MC00 DESIGN"

There is a new addition to the format used by the Apply MTF Manager, that allows one design strings model. That model can be used in all Apply MTF Functions

"DESIGN &" would result in a strings model called just "DESIGN"

Note: The above format can also be used for the following Apply MTF Models:

Model for Cross Sections	"CROSS SECTIONS &"
Model for Polygons	"VIS POLYGONS &"
Model for Road Boundary	"ROAD BOUNDARIES &"
Model for Tadpoles	"TADPOLES &"

Template LHS input

name of the template to be applied to the left side of the reference SA

Template RHS input

name of the template to be applied to the right side of the reference SA

Tin interface input

If non-blank, the name of the tin to calculate the cut/fill interfaces against at the end of the fixed part of the templates given in the left and right template fields

If blank, then only the fixed part of the templates is used unless a tin is specified in the decisions section of the templates

Boxing tick box not ticked

If ticked then all boxing models will be created

Polys tick box ticked

If ticked then both polygon models will be created

Road Tin tick box ticked

If ticked then road tin and model will be created

Depth Polys tick box not ticked

If ticked then depth polygon models will be created

Tadpoles tick box not ticked

If ticked then tadpole model will be created

Filter Xsect tick box not ticked

If ticked then filter xsection model will be created

Buttons at Bottom

Create/Update button

A validation on the panel is carried out along with a check if an Apply MTF Function being created, exists or not

Appropriate warning messages in the output window will be displayed

The Apply MTF Function is created or an existing one is updated

If a new Apply MTF is being created, then the grid on the AMM panel is refreshed and the tick box <Recalc using exist mtf> is set to ticked

Clear Grid button

Clears the selection lines in the grid

Sort Grid button

Sorts the grid lines by placing lines where the <Recalc using exist mtf> box is ticked on, at the top of *the grid*

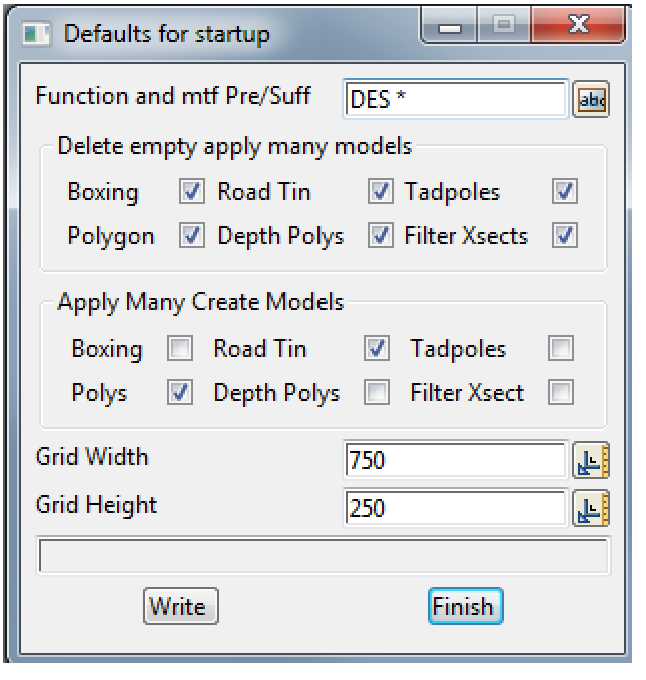
Add AMM button

For clarity...clear the grid first and then select the Super Alignment you wish to add another Apply MTF Function to

Del AMM button

In order to delete (or remove) from the Apply MTF Manager any connection to a Super Alignment, then first clear the grid, select the Super alignment and UNTICK the <Apply> box

20.3.4 Defaults For Startup



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function & mtf Pre/Suff	input	DES*	

If non-blank, entry used as a prefix or suffix in conjunction with the Apply MTF Reference Super Alignment Name

Example:

Reference SA name:	MC00
Pre/Suff:	DES*
Apply MTF Function Name:	DES MC00

Reference SA name:	MC00
Pre/Suff:	* DES
Apply MTF Function Name:	MC00 DES

If blank, the Apply MTF Reference Super Alignment Name is used only

Apply MTF Function Name: MC00

Delete empty Apply MTF models

Boxing tick box ticked

If ticked then any empty boxing models will be deleted when an existing AM is updated

Polygon tick box ticked

If ticked then any empty polygon models will be deleted when an existing AM is updated

Road Tin tick box ticked

If ticked then any empty road tin models will be deleted when an existing AM is updated

Depth Polys tick box ticked

If ticked then any empty depth polys models will be deleted when an existing AM is updated

Tadpoles tick box ticked

If ticked then any empty tadpoles models will be deleted when an existing AM is updated

Filter Xsects tick box ticked

If ticked then any empty filter xsects models will be deleted when an existing AM is updated

Apply MTF Create Models

Boxing tick box not ticked

If ticked then all boxing models will be created

Polys tick box ticked

If ticked then both polygon models will be created

Road Tin tick box ticked

If ticked then road tin and model will be created

Depth Polys tick box not ticked

If ticked then depth polygon models will be created

Tadpoles tick box not ticked

If ticked then tadpole model will be created

Filter Xsect tick box not ticked

If ticked then filter xsection model will be created

Grid Width real 750

If non-blank entry used as width of grid on main panel

Grid Height real 250

If non-blank entry used as height of grid on main panel

Write button

If selected, the defaults file (Create_Apply_Many_Manager_Panel.def) will be written out to the <.project> directory

20.3.5 Kerb Return Function

Position of option on menu: **Design =>Apply =>Kerb return**

The **kerb return** option is used to help create the vertical geometry for an alignment string being used as a kerb return.

The standard situation is that in the plan view, the kerb return joins two kerb lines which already have z-values.

It is then necessary to define vertical geometry for the kerb return, usually taking into consideration the z-values and incoming grades from selected strings (often the centre lines and kerb strings).

A standard first guess at the kerb return vertical geometry is known as the quarter point vertical geometry and it is defined by

(a) break the section along the kerb return into four equal pieces

(b) project the incoming grade from a selected start grade string to intersect the first quarter point line

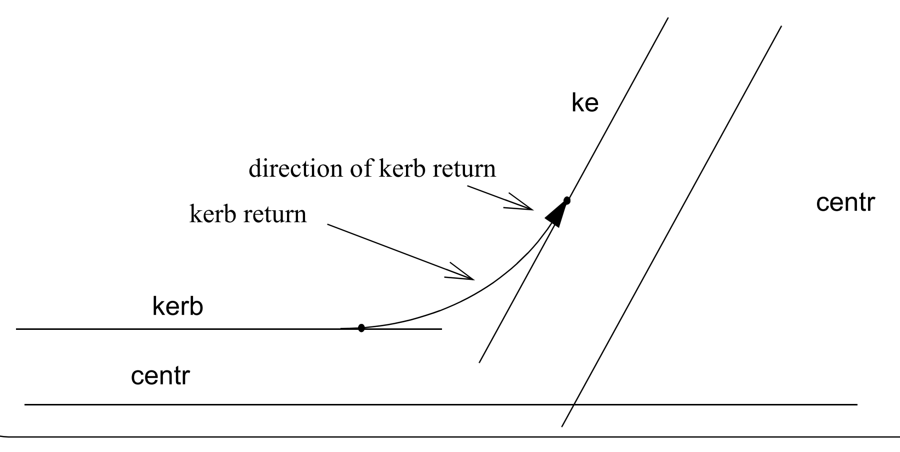
(c) project back the outgoing grade from a selected end grade string to intersect the last quarter point line

(d) join the two projected intersection points

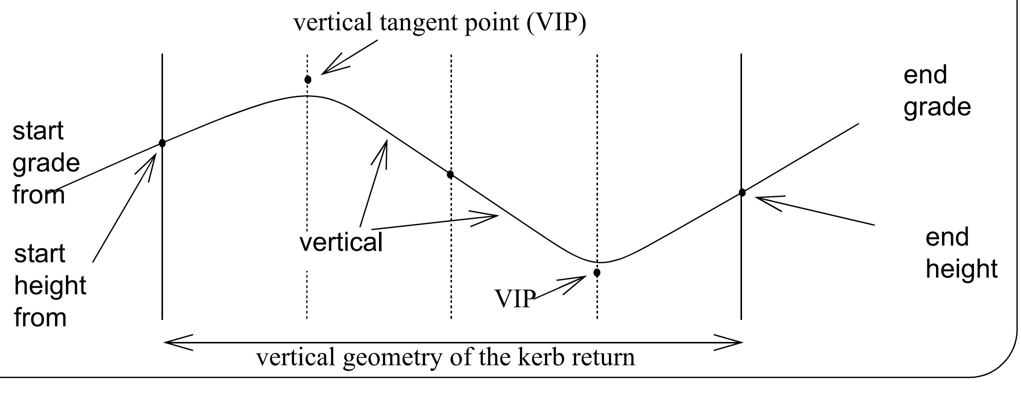
(e) define vertical intersection points (VIP's) at the intersection points

(f) create back to back vertical curves for the VIP's

Plan View of the Horizontal Geometry of the Kerb Return



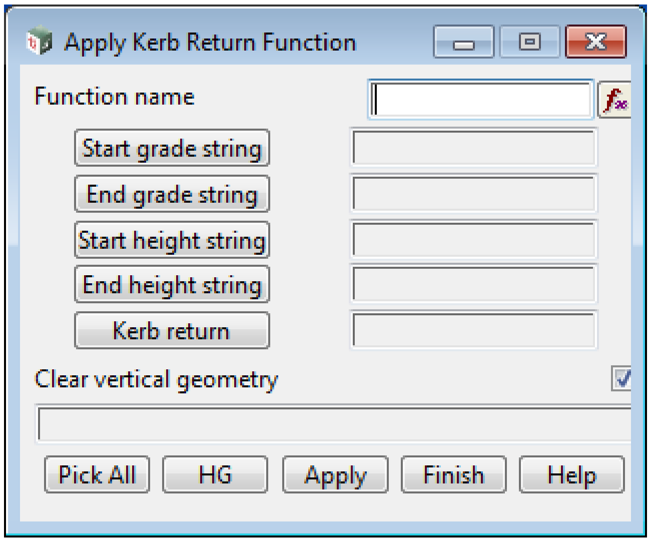
Section View of the Quarter-Point Vertical Geometry for the Kerb Return



The **kerb return** option creates a function which will automatically create the quarter-point vertical geometry for the kerb return and will recalculate it as required.

Note: the actual **direction** of the kerb is important - start grade and start height apply to the **start** of the kerb and end grade and end height apply to the **end of the kerb** where start and end are determined by the **actual direction** of the kerb (**not** the **picking direction**).

On selecting the **kerb return** option, the **apply kerb return function** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function name	function box		available kerb functions

*name of the function to define the **apply** for. If the function already exists and is picked from a pop-up or an <enter> is given at the end of the name, the information from the existing function will be placed in the appropriate panel fields.*

Start grade string string-select

select the string to be used for the in grade at the start of the kerb return.

End grade string string-select

select the string to be used for the out grade at the end of the kerb return.

Start height string string-select

select the string to be used for the height at the start of the kerb return - this is used for the z-value at the start of the kerb return.

End height string string-select

select the string to be used for the height at the end of the kerb return - this is used for the z-value at the end of the kerb return.

Kerb return string-select

select the alignment string that is the kerb return.

Clear vertical geometry tick box

if ticked, the vertical geometry of the kerb return string is cleared as soon as the **apply** *button is selected, or a recalc done.*

Pick all button

if **pick all** *is selected, the user is asked to sequentially select the four strings: the Start grade string, End grade string, Start height string and End height string. The strings are automatically assigned to the appropriate string-select panel fields.*

HG button

brings up the **kerb return function create HG** *panel. This option is currently under development.*

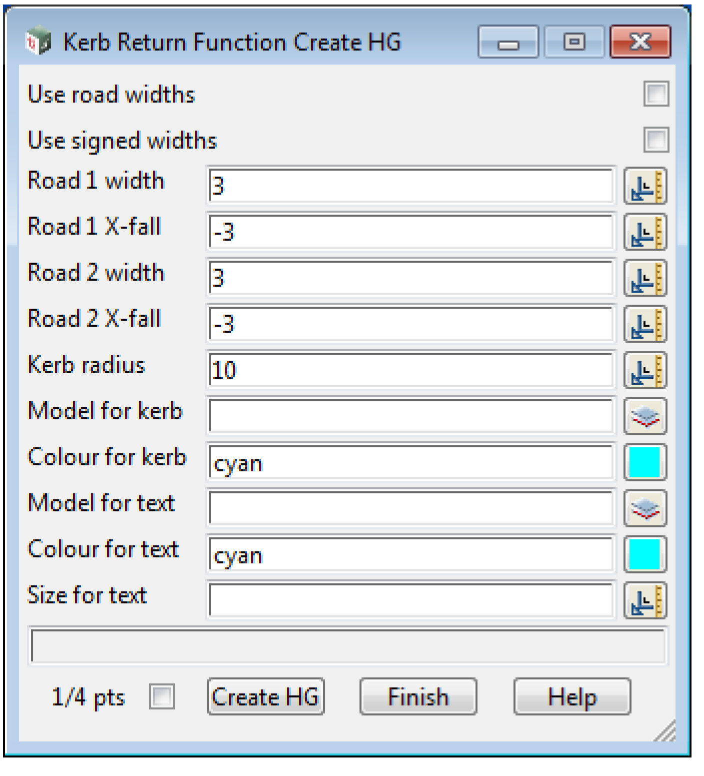
Apply button

if the kerb return has no vertical geometry (it will be automatically be deleted if the **clear vertical geometry** *is set to* ***tick****) or the vertical geometry is the quarter-point geometry for the previous apply, then new quarter-point vertical geometry is created for the kerb return string.*

Otherwize, nothing happens when the **apply** *button is selected. That is, if the kerb return already has vertical geometry which is not the quarter-point geometry for the previous apply, and the* **clear vertical geometry** *is set to* ***no tick****, then* ***nothing happens*** *when the* **apply** *button is selected.*

Note

The experimental **Kerb Return Function Create HG** panel is



20.3.6 String Modifiers Function

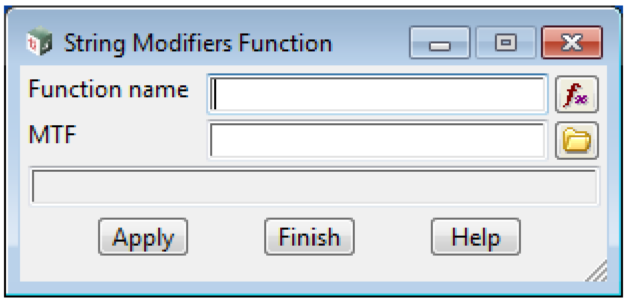
Position of option on menu: **Design =>Apply =>String modifiers**

The **String Modifiers** option is used to create vertical intersection points for an alignment string, or heights for a 2d, 3d, 4d, polyline or super string, using projection of heights from a selected string and user given cross-falls or cross-fall between two selected strings.

The **String Modifier** commands are actually part of an MTF file and are documented in 21 Advanced Design.

This option allows the user to run just the ***string_modifiers*** section of the ***mtf***.

Selecting **String Modifiers** displays the **String Modifiers Function** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function name	function box		available template functions

name of the function to define the **apply** *for. If the function already exists and is picked from a pop-up or an <enter> is given at the end of the name, the information from the existing function will be placed in the appropriate panel fields.*

Field Description	Type	Defaults	Pop-Up
MTF file	file box		available mtf files

the name of the MTF file to use the string_modifiers section from.

Field Description	Type	Defaults	Pop-Up
Apply	button		

apply the string modifiers section of the MTF.

20.3.7 Interface Function

Position of option on menu: **Design =>Apply =>Interface**

An **interface** string is where the design meets the natural terrain - it shows the **extent** of the design. The method used for calculating the **interface string** for a given string, tin and cut and fill slopes, is as follows:

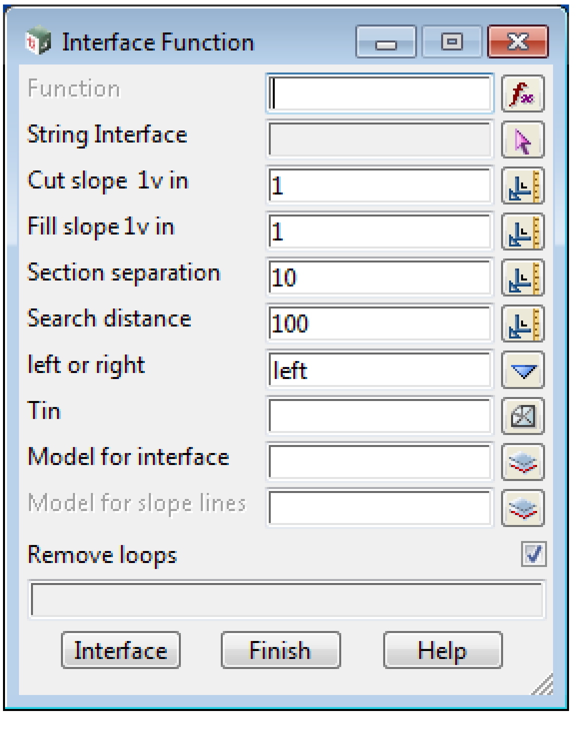
> For a point on a string, an interface point is calculated by going off at right angles to the string along a line of fixed slope until either the tin is cut or a fixed (plan) distance is travelled. The slope of the line is either the cut or fill slope depending on whether the string point is below the tin (cut point) or above the tin (fill point).
>
> The interface string for a string is constructed by calculating the interface points at regular intervals along the string and then joining the interface points together to form the interface string.

After calculating an interface string and examining the results, it is often necessary to make changes to either the original string, the cut and fill slopes used, or even the tin used for interfacing against. The interface string and slope strings created during the **interface** would no longer be valid and need to be replaced by new information.

Although this can be accomplished by deleting the incorrect models and replacing them by new ones by re-applying the interface, this is would be time consuming if repeated over and over again. The **interface function** has been introduced to help reduce the work involved when re-calculating an interface string.

Basically, an **interface function** keeps track of all the information involved in creating an interface string and all the strings and models created during the **interface.** If either the original string, the cut and fill slope or the tin is subsequently modified, the interface functions can be re-run and all the old information automatically deleted and replaced by the updated information.

On selecting **interface**, the **interface function** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function	function box		available interface functions

name of the function to define the **interface** *for. If the function already exists and is picked from a popup or and an <enter> is given at the end of the name, the information from the existing function will be placed in the appropriate panel fields.*

String interface string select box

select the string to calculate the interface from.

Cut slope 1 v in input 1.0

the slope of the interface line if a point is in cut. A cut slope of one vertical to the given value of horizontal units is used. Positive is ***up*** *for a cut slope. A value 0 is used to designate a horizontal line.*

Fill slope 1 v in input 1.0

the slope of the interface line if the point is in fill. A fill slope of one vertical to the given value of horizontal units is used. Positive is ***down*** *for a fill slope. A value 0 is used for a horizontal line.*

Section separation input 10

the distance between the points on the selected string that interface points will be calculated from.

Search distance input 100

the (plan) distance to search along the interface line to see if the tin has been intersected. If the tin has not been intersected then the z-value at the end of the interface line is taken as the interface point z-value.

Left or right choice box left left, right

the interface point is calculated by going out at right angles to the selected string. It is possible to go to either the left or the right depending on the value of this field.

Use super strings tick box

if ticked, the interface and cut and fill slope lines are created as super strings.

Tin tin box available tins

name of the tin that the interface points will be calculated against.

Model for interface model box available models

name of the model to contain the calculated interface string.

Model for slope lines model box available models

if not blank, name of the model to contain the lines connecting the string points with the interface points (the slope lines). If the field is blank, no slope lines are recorded.
If blank, the slope lines are not created.

Remove loops tick box tick

if tick, the strings and sections created by the Interface function are process to try an remove any loops from the strings, and create sections that don't cross each other. That is, remove any places where a string crosses over itself, or sections cross each other.

Loops in the strings and crossing sections often occur on the inside of curves or the inside of adjacent lines.

Interface button

An interface string is calculated using the selected string and the parameters defined in the panel. The interface string points are coloured red if the original string point was in cut and green if the original string point was in fill.

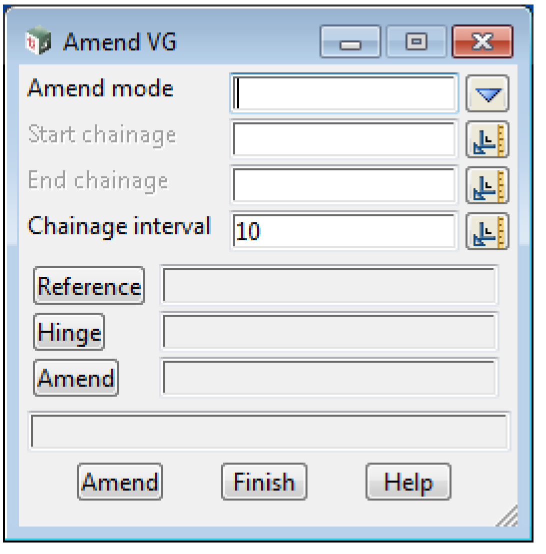
<esc> can be used to abort the interface option.

20.3.8 Amend VG

Position of option on menu: **Design =>Apply =>Amend VG**

Amend VG is used to modify the vertical geometry of an alignment string using a variety of methods including linear interpolation cubic reverse and circular reverse interpolating or by extending crossfall between two given strings.

Selecting **Amend VG**, the **Amend VG** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Amend mode	choice box		linear, cubic reverse, circular reverse, VG linear, Extend Xfall

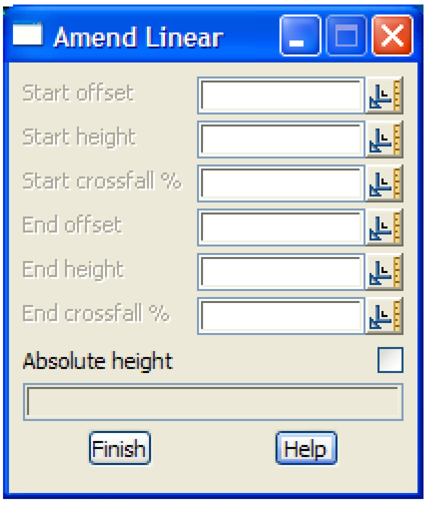
*if **linear**, a panel to input the values for **Start offset/height/crossfall %, End offset/height/crossfall %** and **Absolute height** is displayed and filled in to define how to modifying the vertical geometry of the **Amend** string*

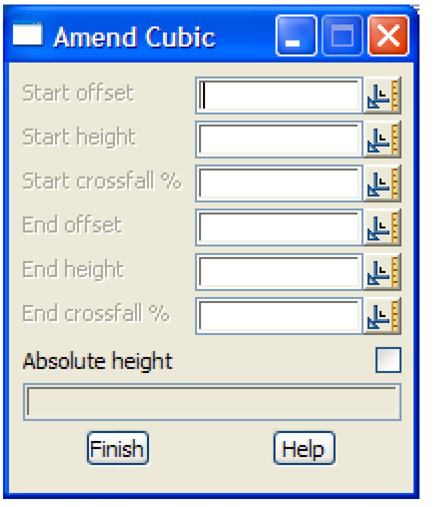
*If **cubic reverse**, a panel to input the values for **Start offset/height/crossfall %, End offset/height/crossfall %** and **Absolute height** is displayed and filled in to define how to modifying the vertical geometry of the **Amend** string*

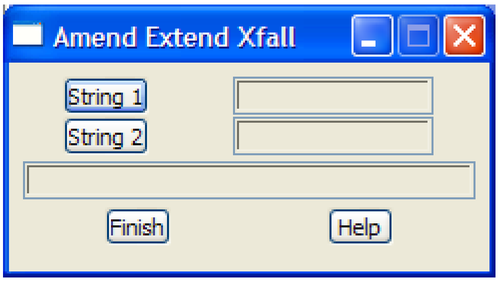
*If **circular reverse**, no extra information is needed to define how to modifying the vertical geometry of the **Amend** string*

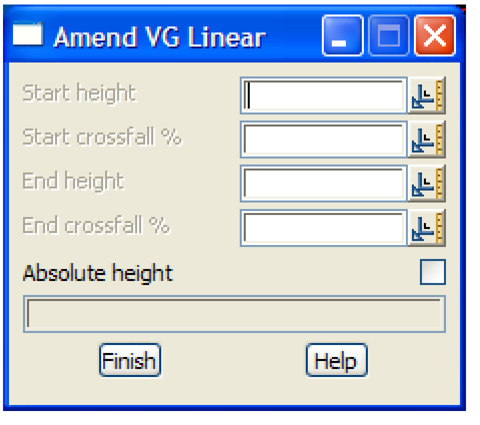
*If **VG linear**, a panel to input the values for **Start height/crossfall %, End height/crossfall %** and **Absolute height** is displayed and filled in to define how to modifying the vertical geometry of the **Amend** string*

*If **Extend Xfall**, a panel to select the two strings to extend the crossfall from is displayed and filled in to define how to modifying the vertical geometry of the **Amend** string*









Start chainage input

chainage to start modifying the vertical geometry of a string

End chainage input

chainage to stop modifying the vertical geometry of a string

Chainage interval input 10

chainage interval to apply the vertical geometry modification

Reference string select box

select the reference string to define chainage and perpendicular. If there is no ***Hinge*** *string, then z-values are only taken from the Reference string.*

chainage interval to apply the vertical geometry modification

Amend string select box

select the reference string to define chainage and perpendicular. If there is no ***Hinge*** *string, then z-values are only taken from the Reference string.*

Hinge string select box

if selected, the string to take z-values from

Amend string select box

the string to have its vertical geometry amended

Amend button

amend the vertical geometry of the Amend string between the start and end chainages

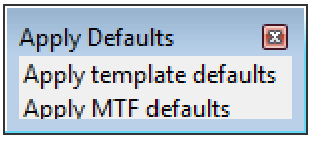
20.3.9 Apply Defaults

Position of menu: **Design =>Apply =>Defaults**

Position of option on menu: **Design =>Apply =>Defaults =>Apply template defaults**

Design =>Apply =>Defaults =>Apply MTF defaults

The **Defaults** walk right menu contains options to set up defaults when first using the **Apply Template Function** and **Apply MTF Function** panels.



The two options **Apply template defaults** and **Apply MTF defaults** will now be described.

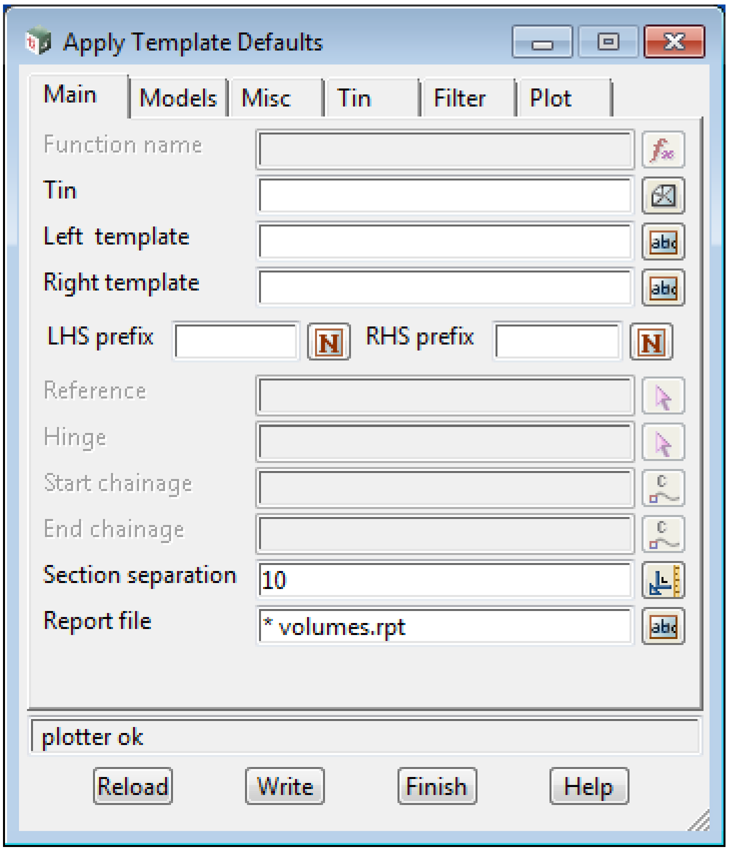
There are two files *apply_defaults.4d* and *apply_many_defaults.4d* which contain the ***pre*** or ***post*** *text* to be added to the name of the function to automatically create many entries in the panel fields for *Apply template* and *Apply MTF* plus default values for all the other panel fields.

After the name of a new function is typed in and <Enter> pressed, all the fields mentioned in the default files are automatically filled in using the function name and the defined ***pre-post text*** as supplied in the file, plus setting any other panel fields from default values supplied in the file. If the files are missing, no panel fields are automatically filled in.

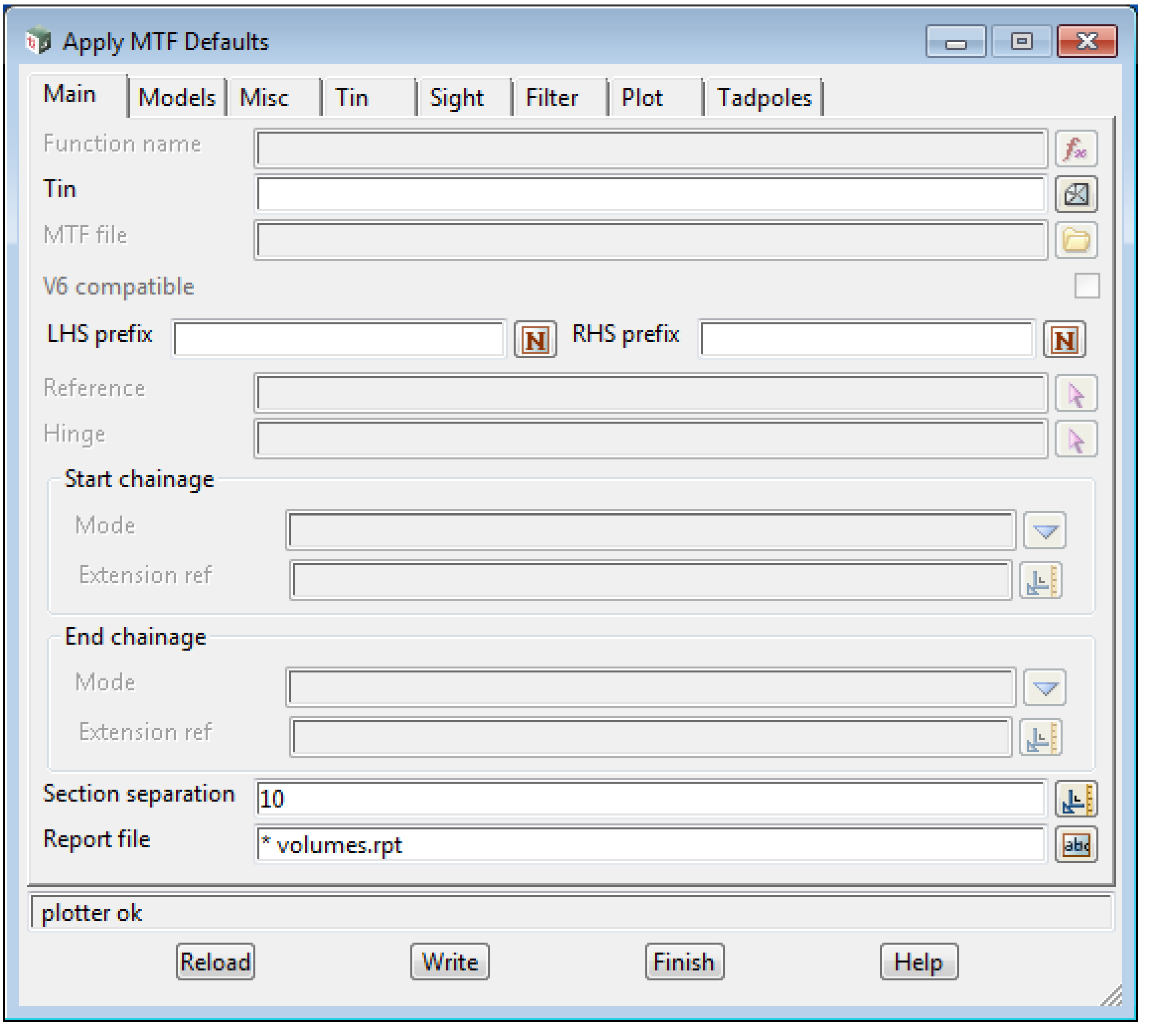
The files can be created and edited using the **Apply template defaults** or **Apply MTF defaults** option which brings up the **Apply Template Defaults** or **Apply Templates Defaults** panels respectively.

Each of the fields in the **Apply Template Defaults** panel matches a field in the **Apply** panel and so for the documentation see 20.3.1 Apply Template Function.

Each of the fields in the **Apply Templates Defaults** panel matches a field in the **Apply Templates Function** panel and so for the documentation see 20.3.2 Apply MTF Function.



Each of the fields in the **Apply Template Defaults** panel matches a field in the **Apply** panel and so for the documentation see 20.3.1 Apply Template Function.



Each of the fields in the **Apply Templates Defaults** panel matches a field in the **Apply Templates Function** panel and so for the documentation see 20.3.2 Apply MTF Function.

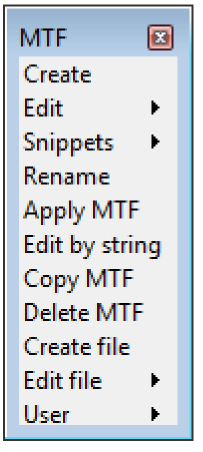
20.4 MTF

Position of menu: **Design =>MTF**

In the **Apply MTF** option, a **MTF file** is used to control the application of templates, modifiers, boxing etc to the Hinge string.

The full definition for the MTF file is given in the section 21.1 The Modifiers and Templates File - MTF.

The **MTF** walk-right menu is



Each option in this menu will now be described.

For the options, see:

Create	20.4.1 Create MTF
Edit	20.4.2 Edit MTF
Snippets	20.4.3 Snippets
Rename	20.4.4 Rename MTF
Apply MTF	20.3.2 Apply MTF Function
Edit by string	20.4.6 Edit by String
Copy MTF	20.4.7 Copy MTF
Delete MTF	20.4.8 Delete MTF
Create file	20.4.9 Create File
Edit file	20.4.4 Rename MTF

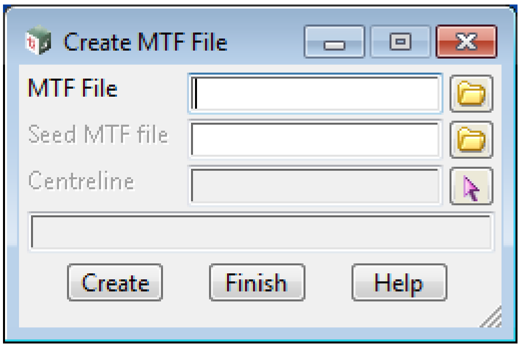
20.4.1 Create MTF

Position of option on menu: **Design =>MTF =>Create**

The **MTF=>Create** option is used to create a new MTF file (*.mtf).

The created MTF file is written to disk and can be edited using the **MTF=>Edit** option.

Selecting **Create** displays the **Create MTF File** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
MTF file	file box		*.mtf files

name of the MTF file to create.

Seed MTF file	file box		*.mtf_seed files

name of an MTF file to load into the new MTF file. This allows standard information to be automatically included in the new MTF file. The Seed MTF file is just a standard MTF file with the extension after the . changed to mtf_seed

Centreline	string select		

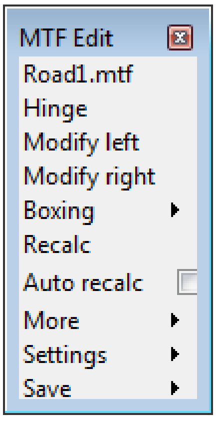
if a centreline string is selected, it is used for selecting chainages in the MTF

Create	button		

create an MTF file with name given by the **MTF file** *panel fields.*
If the file given in the **MTF file** *field does not exist, then the* **MTF Edit** *menu is placed on the screen and is used to created and edit the MTF sections in the MTF file.*
If the file already exists, then nothing will happen on selecting **Create**.

20.4.1.1 MTF Edit

When **Create** is selected from the **Create MTF File** panel, it brings up the **MTF Edit** menu which is used to create and edit the MTF sections to be saved in the MTF file.



The **MTF Edit** menu is described in the section 21.1 The Modifiers and Templates File - MTF.

Go to the next section 20.4.2 Edit MTF or return to 20.4 MTF.

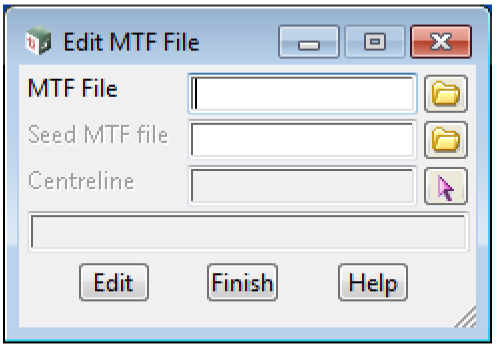
20.4.2 Edit MTF

Position of option on menu: **Design =>MTF =>Edit**

The **MTF=>Edit** option is used to edit mtf files (*.mtf).

The **MTF=>Edit** option has two modes of operation - selecting the **MTF=>Edit** itself, or by activating the **MTF=>Edit** option's walk-right menu, **folder *.mtf**.

Selecting **MTF=>Edit** itself brings up the **Edit MTF File** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
MTF file	file box		*.mtf files

name of the MTF file to edit

Seed MTF file	file box		*.mtf_seed files

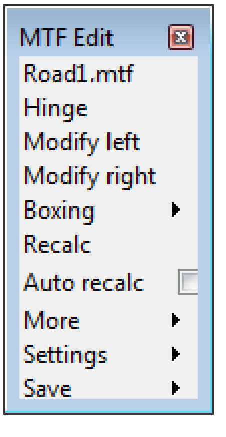
for a new MTF file, the name of an MTF file to load into the new MTF file. This allows standard information to be automatically included in the new MTF file. The Seed MTF file is just a standard MTF file with the extension after the . changed to mtf_seed

Centreline	string select		

if a centreline string is selected, it is used for selecting chainages in the MTF

Edit	button		

edit the file given by **MTF file** *panel field.*
If the file given in the **MTF file** *field exists, then the* **MTF Edit** *menu is brought up to create/edit the MTF sections.*



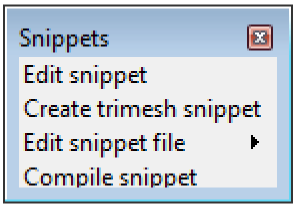
Similarly the **MTF=>Edit** walk-right menu provides a list all the MTF files (files ending in .mtf) in the current folder. When a MTF file is selected from the list, the **Edit MTF File** panel is brought up to create/edit the MTF sections.

The **MTF Edit** menu is described in the section 21.1 The Modifiers and Templates File - MTF.

Go to the next section 20.4.3 Snippets or return to 20.4 MTF.

20.4.3 Snippets

‘



See

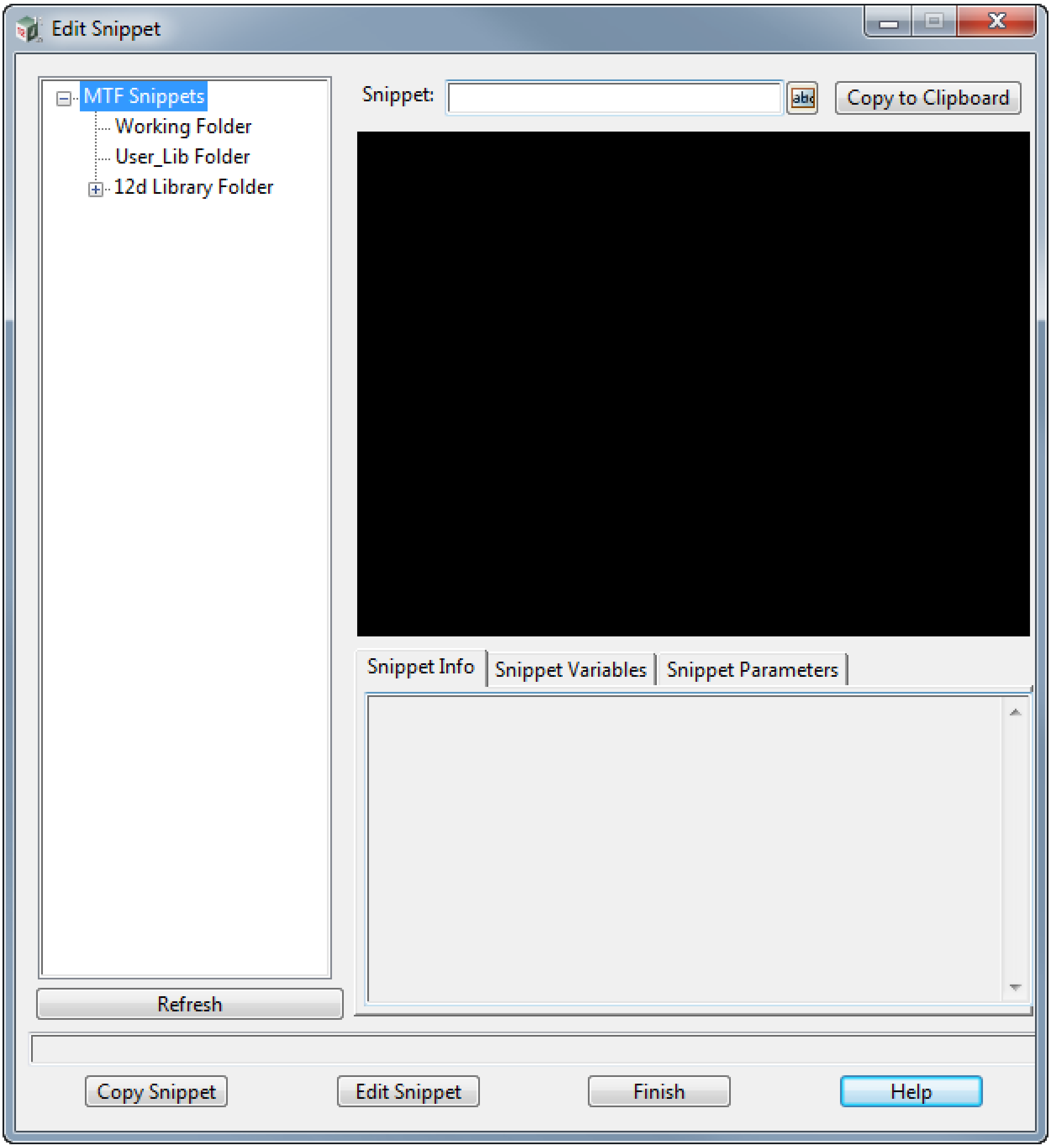
20.4.3.1 Edit Snippet
20.4.3.2 Create Trimesh Snippet
20.4.3.3 Edit Snippet File
20.4.3.4 Compile Snippet
20.4.3.5 12d Supplied Snippets

For information on Snippets, see 21.5 Defining and Using Snippets

20.4.3.1 Edit Snippet

Position of option on menu: **Design =>MTF =>Snippets =>Edit snippet**

Selecting **Edit snippet** brings up the **Edit Snippet** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
MTF Snippets	Tree Box		

lists all of the MTF Snippets available in the following folders:

Working Folder: MTF Snippets located in the current Working Folder
User_Lib Folder: MTF Snippets located in the User Library (USER_LIB_4D)

12d Library Folder: MTF Snippets located in the 12d Model Library (LIB_4D)
(C:\Program Files (x86)\12d\12dmodel\11.00\Library, or
C:\Program Files\12d\12dmodel\11.00\library depending on your setup)

Expanding Working Folder, User_Lib Folder of 12d Library Folder will list all the snippets in that folder. For information on the snippets in 12d Library Folder, see 20.4.3.5 12d Supplied Snippets.

Clicking on a snippet name in the list displays the following information about the selected snippet:

For more information on the inserting and use of snippets, see 21.5 Defining and Using Snippets.

Snippet Input box

displays the file path and name of the selected MTF Snippet.

Copy to Clipboard Button

copies the selected MTF Snippet file path to the clipboard

This may be pasted into the **Modifiers->MTF Snippet** *panel, or useful for creating snippets that call other snippets.*

MTF Snippet Preview Draw box

displays a preview image of the selected MTF Snippet.

Users can create their own preview images using the following guidelines:

- *A preview image must have the same name as its associated MTF Snippet. For example; a snippet named "My_Snippet.MTFSNIPPET", will have a preview image named "My_Snippet.jpg",*
- *The preview image should have the dimensions 400 pixels wide by 300 pixels high,*
- *The preview image must be in jpeg (*.jpg) image format,*
- *The preview image must be located in either your "$USER/images" folder (USER_4D) or your current Working Folder.*

Refresh Button

updates the MTF Snippets tree box.

When clicked, the macro will find and display all MTF Snippets in the current project Working Folder, User Library Folder, and 12d Library Folder.

Note: To fully refresh a list, the relevant tree must also be collapsed and then re-expanded.

Snippet Info Tab

Snippet Info Text Edit Box

displays information about the currently selected MTF Snippet.

Users can display their own custom information for a snippet by using the following guidelines:

- *Open the MTF Snippet in a text editor, by select Edit Snippet,*
- *At the top of the file, prefix your information with:* // INFO

For example; if your snippet file contains:

// INFO Snippet Type: INSERT

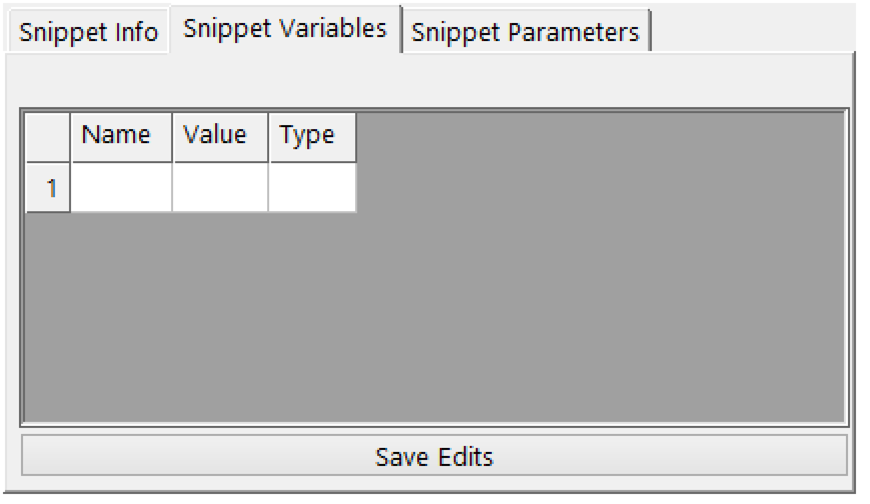
// INFO Description: Kerb & Gutter

The ***Snippet Info*** *text box will display:*

Snippet Type: INSERT

Description: Kerb & Gutter

Snippet Variables Tab



Snippet Variables Grid Control box

lists the currently selected MTF Snippet variables.

Each row represents a variable.

Column 1 - represents the variable name,

Column 2 - represents the variable value,

Column 3 - represents the variable type.

MTF Snippet variable types include: Number, Text

Only variables that define a single value are displayed (i.e.; formulas are ignored).

For example:

#define	*_A*	*1.23*
@ def_tok	*A*	*1.23*
@ def_tok	*STR1*	*KI*

Users can create new variables, edit existing variables and delete existing variables.

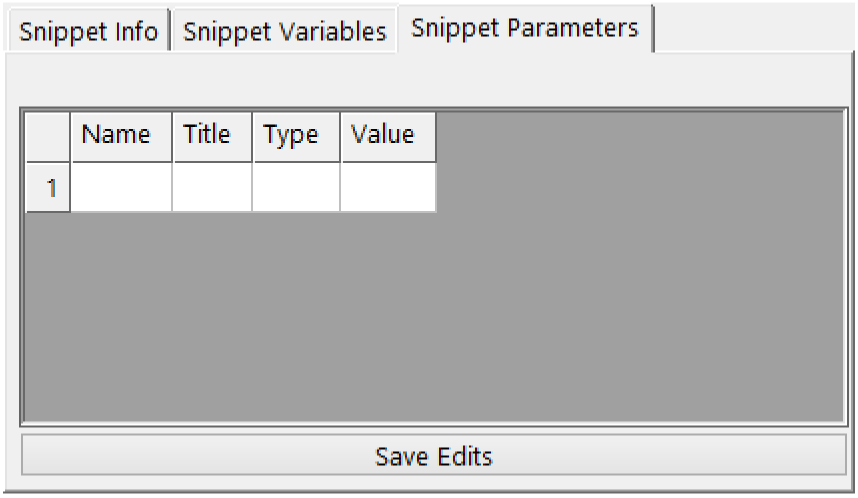
Note: Creating and/or deleting a variable in the Grid Control Box does not change how that variable is used elsewhere in the snippet.

Save Edits button

open the ***Unsaved Changes*** *panel to write out the* ***Snippet Variables*** *(Grid Control Box) to the MTF Snippet file.*

Note: 12d Model Library snippets and User Library snippets cannot be edited via the panel. Therefore this button is disabled when these snippets are selected.

Snippet Parameters Tab



Snippet Parameters Grid Control box

lists the currently selected MTF Snippet parameters.

Each row represents a parameter.

Column 1 - represents the parameter name,

Column 2 - represents the parameter title,

Column 3 - represents the parameter type,

Column 4 - represents the parameter value.

Parameters to be displayed in the Edit Snippet panel must be defined in the snippet_params.4d *file.*

Only parameters of the following types may be displayed and edited: TEXT, REAL, INTEGER

Users can create new parameters, edit existing parameters and delete existing parameters.

Note: Creating and/or deleting a parameter in the Grid Control Box does not change how that parameter is used elsewhere in the snippet.

Save Edits button

opens the ***Unsaved Changes*** *panel to write out the* ***Snippet Parameters*** *(Grid Control Box) to the MTF Snippet file.*

Note: 12d Model Library snippets and User Library snippets cannot be edited via the panel. Therefore this button is disabled when these snippets are selected.

Copy Snippet button

opens the ***Copy Snippet To*** *panel and allows the user to copy the selected snippet to the current Working Folder.*

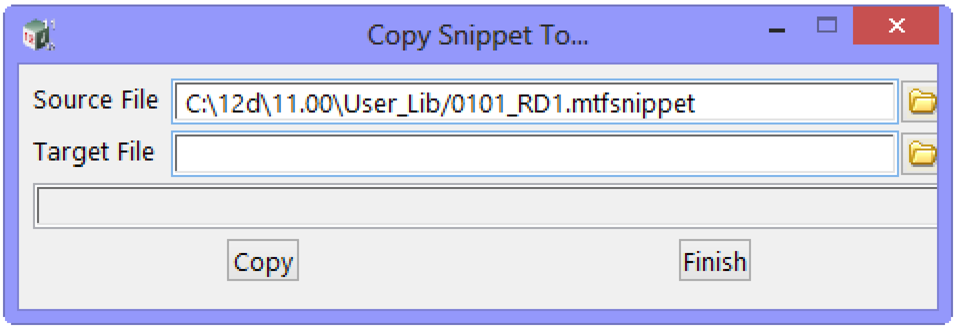
Edit Snippet button

open the selected snippet in a text editor.

*The default text editor is Notepad, unless a text editor is defined in the users env.4d:*EDITOR_4D

Note: The text editors Multiple Instance settings will control whether snippets open for editing in the same window or create a new window.

Selecting the **Copy Snippet** button brings up the **Copy Snippet To...** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Source File File Box

file path and name of the MTF Snippet being copied.

*When the **Copy Snippet To** panel opens, this field is auto-populated with the currently selected MTF Snippet from the MTF Snippets tree.*

Target File File Box

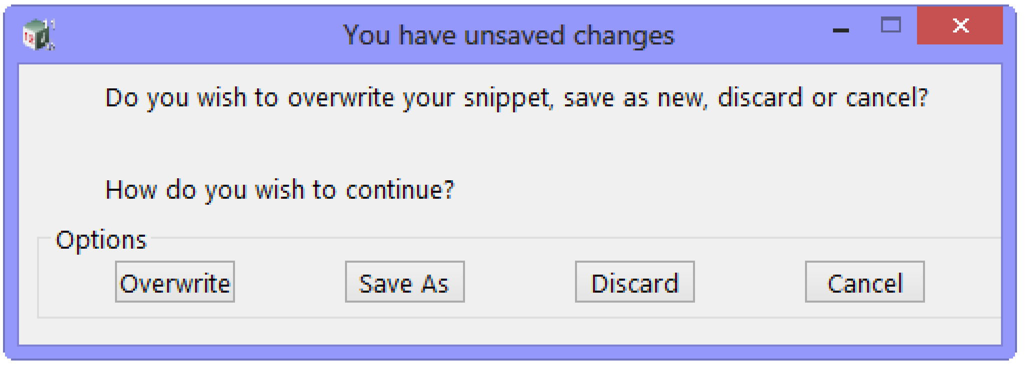
enter a new file name and path for the MTF Snippet being copied.

Entering a file name with no path will copy the MTF Snippet to the current Working Folder.

Copy Button

*when clicked the **Source File** is copied with the **Target File** path and name.*

Selecting the **Save Edits** button brings up the **Unsaved Changes** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Overwrite Button

*writes all variables in the **MTF Variables** Grid Control Box to the currently selected MTF Snippet file.*

Existing variables of the same name will be overwritten.

Save As Button

*save all variables in the **MTF Variables** Grid Control Box to a new MTF Snippet file.*

*Selecting this button brings up the **Copy Snippet To** panel.*

Discard Button

*all edits made to the variables in the **MTF Variables** Grid Control Box are discarded and the currently selected MTF Snippet file is not edited.*

Cancel Button

*closes the **Unsaved Changes** panel.*

How snippet_params.4d Works

snippet_params.4d is a file that allows users to control the visibility of MTF Snippet Parameters in the Snippet Parameters Grid_CtrlBox.

File Format:

- *Plain Text*
- *Each line should contain one MTF Snippet Parameter name. For example: LANE1*
- *MTF Snippet Parameter names are case sensitive*
- *Comments are allowed in the file, and should begin with: //*

Where should snippet_params.4d be located?

By default the file is located in:

C:\Program Files\12d\12dmodel\11.00\set_ups

or

C:\Program Files (x86)\12d\12dmodel\11.00\set_ups

Users can copy this file to their User directory (typically C:\12d\11.00\User) or their current ***12d Model*** project directory. The macro will find and read this file in those directories. The macros search order is:

- *Search the current 12d Model project directory for Snippet_Params.4d. If found; read.*
- *Search the User directory for Snippet_Params.4d. If found; read.*
- *Search the Set_Ups directory for Snippet_Params.4d. If found; read.*

Go to the next section 20.4.3.2 Create Trimesh Snippet or return to 20.4.3 Snippets.

20.4.3.2 Create Trimesh Snippet

Position of option on menu: **Design =>MTF =>Snippets =>Create trimesh snippet**

This section of documentation is a work in progress and will be updated in subsequent releases.

For more information on the inserting and use of snippets, see 21.5 Defining and Using Snippets.

Go to the next section 20.4.3.3 Edit Snippet File or return to 20.4.3 Snippets.

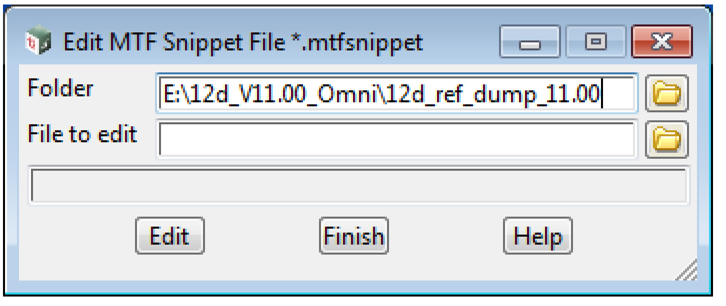
20.4.3.3 Edit Snippet File

Position of option on menu: **Design =>MTF =>Snippet =>Edit snippet file**

The **Edit snippet file** option is used to edit MTF Snippet files (*.mtfsnippet) with the text editor pointed to by the EDIT_4D environment variable.

The **Edit snippet file** option has two modes of operation - clicking on the option **Edit snippet file** itself, or by activating the **MTF =>Snippets =>Edit snippet file** option's walk-right menu, **folder *.mtfsnippet**.

Selecting **Edit snippet file** itself brings up the **Edit MTF File *.mtfsnippet** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Folder	folder box	current folder	

name of the folder for the .mtf snippet file.

File to edit	file box		*.mtfsnippet files

name of the MTF Snippet file, in Folder*, to edit.*

Edit	button		

edit the MTF Snippet file given by the Folder *and* File to edit *panel fields by the text editor pointed to by the* EDIT_4D *environment variable. If the file given in the* File to edit *field does not exist, then a new file is created which already has each of the section headers set up.*

Similarly the **Edit snippet file** walk-right menu provides a list all the MTF Snippet files (files ending in .mtfsnippet) in the current folder. When a file is selected from the list, it is automatically loaded into the text editor.

For more information on the inserting and use of snippets, see 21.5 Defining and Using Snippets.

Go to the next section 20.4.3.4 Compile Snippet or return to 20.4.3 Snippets.

20.4.3.4 Compile Snippet

Position of option on menu: **Design =>MTF =>Snippet =>Compile snippet**

The **Compile snippet** option is used to create a compiled version of a Snippet file.

This can be done to protect any intellectual property (IP) and minimise concerns people may have in developing and sharing snippets.

The compiled snippet file can **not** be read with a text editor.

Compiling a snippet encrypts and encodes the snippet contents into a new file with an extension of **mtfsnippetc** (note the 'c' at the end, denoting a compiled snippet).

A compiled snippet can be run and used by anyone, but the snippet code itself cannot be viewed or edited.

Temporary snippet files, if enabled, are not generated for compiled snippets.

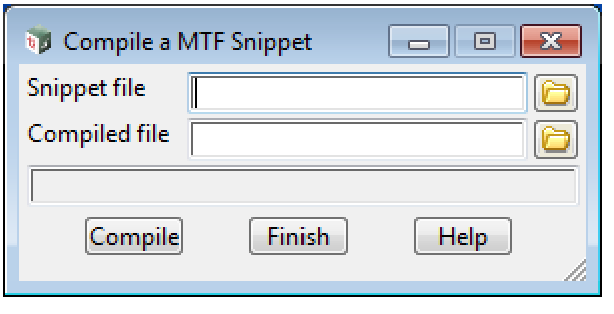
Original Snippet

// INFO Example compiled snippet

Compiled Snippet

nyKgSBlhX6wGqQizY/fRfDcT/YReG2J34wsghywz6Ys+cfVn+Pi+crz/4c9XgNbYI+kmS14F
fIh+l/bNu41RYyD4RtNvIiS9

Selecting **Compile snippet** itself brings up the **Edit MTF File *.mtfsnippet** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Snippet file	file box		*.mtfsnippet files

name of the MTF Snippet file to compile.

Compiled file	file box		*.mtfsnippetc files

name of the compiled MTF Snippet file.

Compile	button		

compile the snippet given in the **Snippet file** *field and write the compiled snippet out with the name given in the* **Compile file** *field.*

For more information on the inserting and use of snippets, see 21.5 Defining and Using Snippets.

Return to 20.4.3 Snippets or 20.4 MTF.

20.4.3.5 12d Supplied Snippets

See

20.4.3.5.1 TRI_1PT_TRENCH_DES_LAY

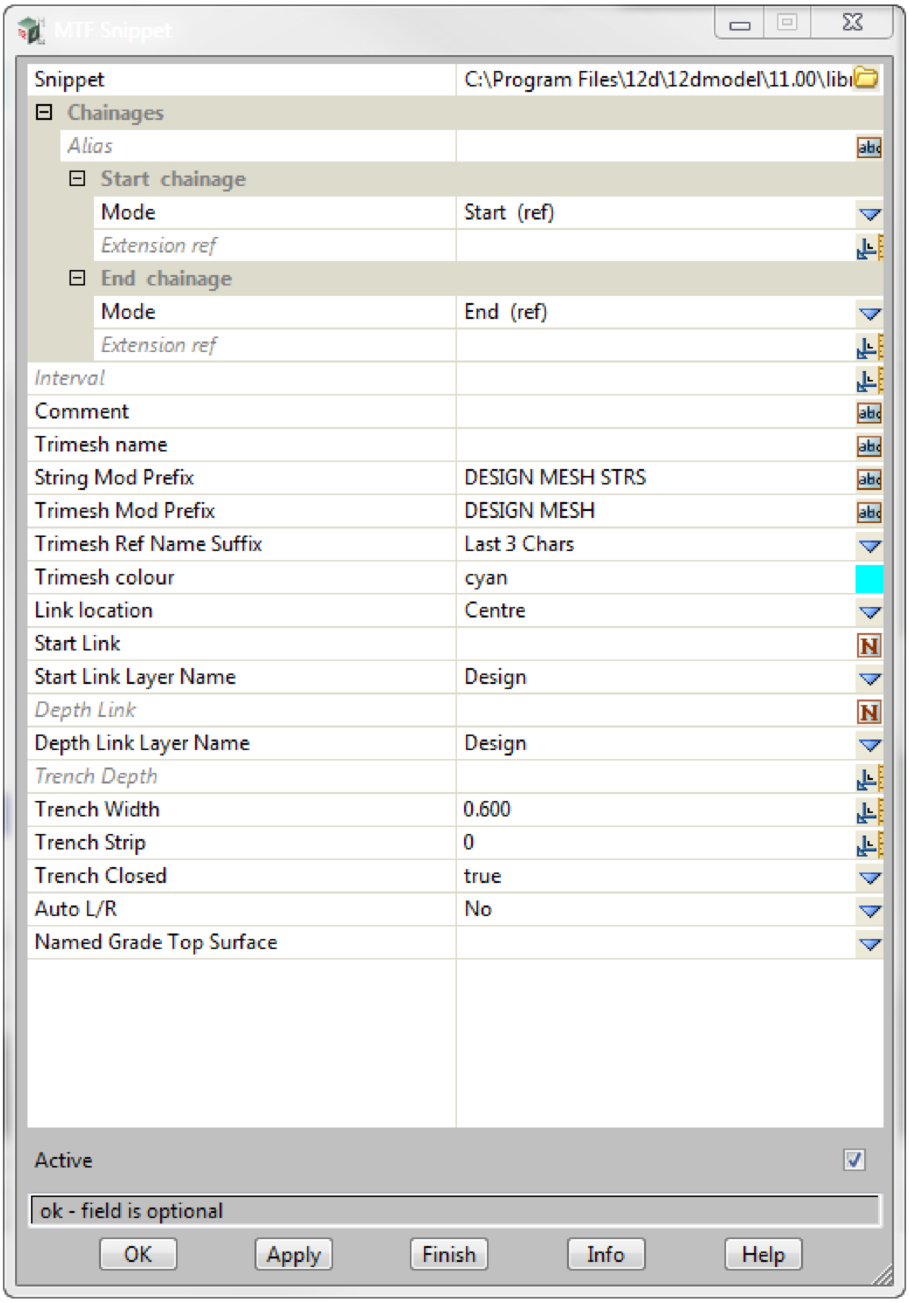
This Snippet is used to create a trench pavement, using <Design> or <Layer> strings created in your MTF.

It is linked to the Start Link at a specified location of left, centre or right.

A depth can be specified either to another link or a set value.

The top surface of the trench is defined by a Named Grade.

The Reference string name can be used as part of the model naming convention.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
MTF file	file box		*.mtf files

name of the MTF file to create.

Snippet	snippet box	*.mtfsnippet,	*.mtfsnippetc files

snippet to run.

Alias, Start Chainage, End Chainage, Interval

defines the start and end chainages to apply a snippet.

For information on these panel fields, see 17.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, Extra start, Extra End, Active, OK, Apply

for information on these panel fields, see 17.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Trimesh Name	input		

trimesh name that is used in model names below

String Mod Prefix	input		DESIGN MESH STRS
Trimesh Mod Prefix	input		DESIGN MESH
Trimesh Ref Name Suffix	choice box	Last 3 Chars	All Chars
			Last Char
			Last 2 Chars
			Last 3 Chars
			Last 4 Chars
			None

(Model syntax: String Mod Prefix   Trimesh Name   Trimesh Ref Name Suffix)

Trimesh Colour	colour box	cyan	various
Link Location	choice box	Centre	Centre
			LHS
			RHS
Start Link	name		
Start Link Layer Name	choice box	Design	various
Depth Link	name		
Depth Link Layer Name	choice box	Design	various

(Another link can be used to control the depth of the trench.

It is optional so if not set then the following Trench depth is used)

Trench Depth	real		

(This is optional so if not set then the Depth Link above is used)

Trench Width	real		0.6
Trench Strip	real		0

(Allows a strip depth measured from the top of the trench)

Trench Closed	choice box	true	true / false
Auto L/R	choice box	No	Yes / No

if set to <No> then the Start and End Links remain unchanged.

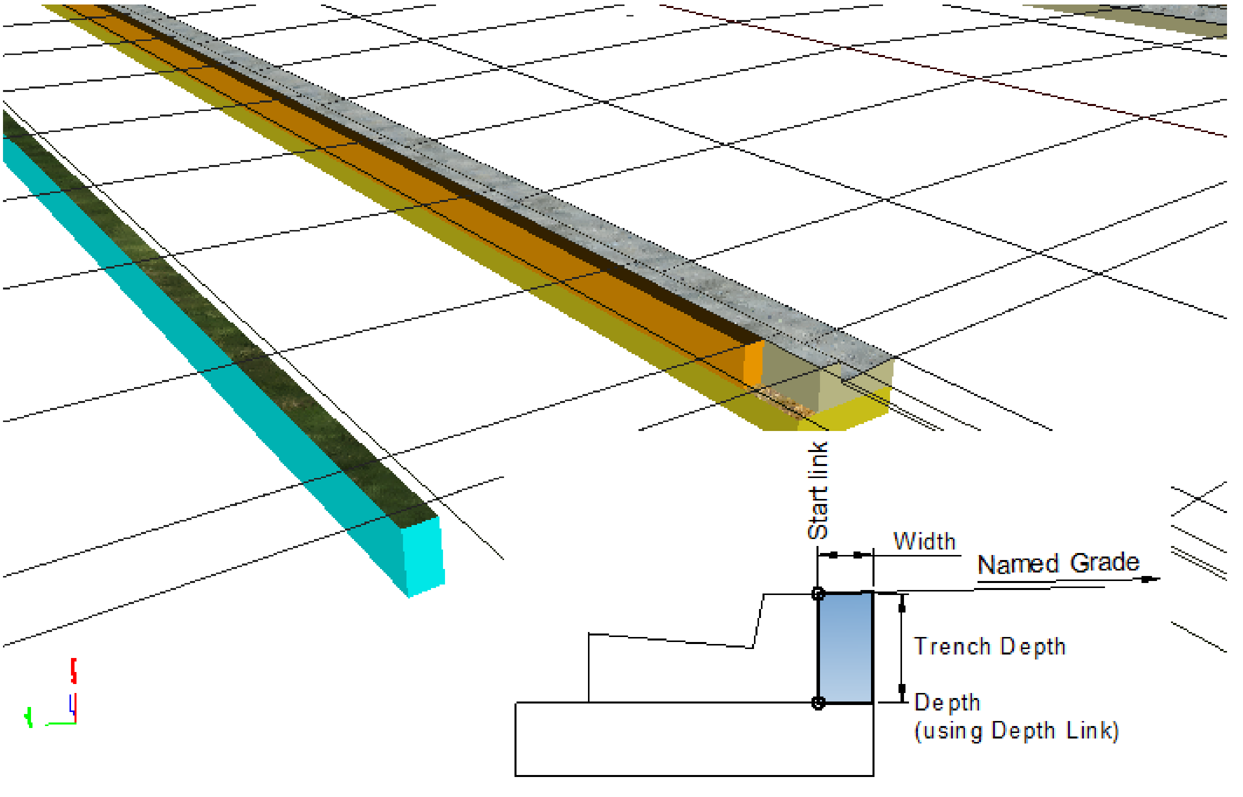
If set to <Yes> then the Start and End Links need the suffix for L/R removed.

*e.g. Any <Design> string names like **ESL** would need the <L> removed*

*e.g. Any <Layer> string names like **PVBALRO1** would need the <LRO1> removed*

Named Grade Top Surface choice box

(The top of the trench will follow this grade while the bottom of the trench is horizontal)



For more information on snippets, see 21.5 Defining and Using Snippets.

Continue to 20.4.3.5.2 TRI_1PT_TRENCH_EXT_STRS or return to 20.4.3.5 12d Supplied Snippets or 20.4.3 Snippets.

20.4.3.5.2 TRI_1PT_TRENCH_EXT_STRS

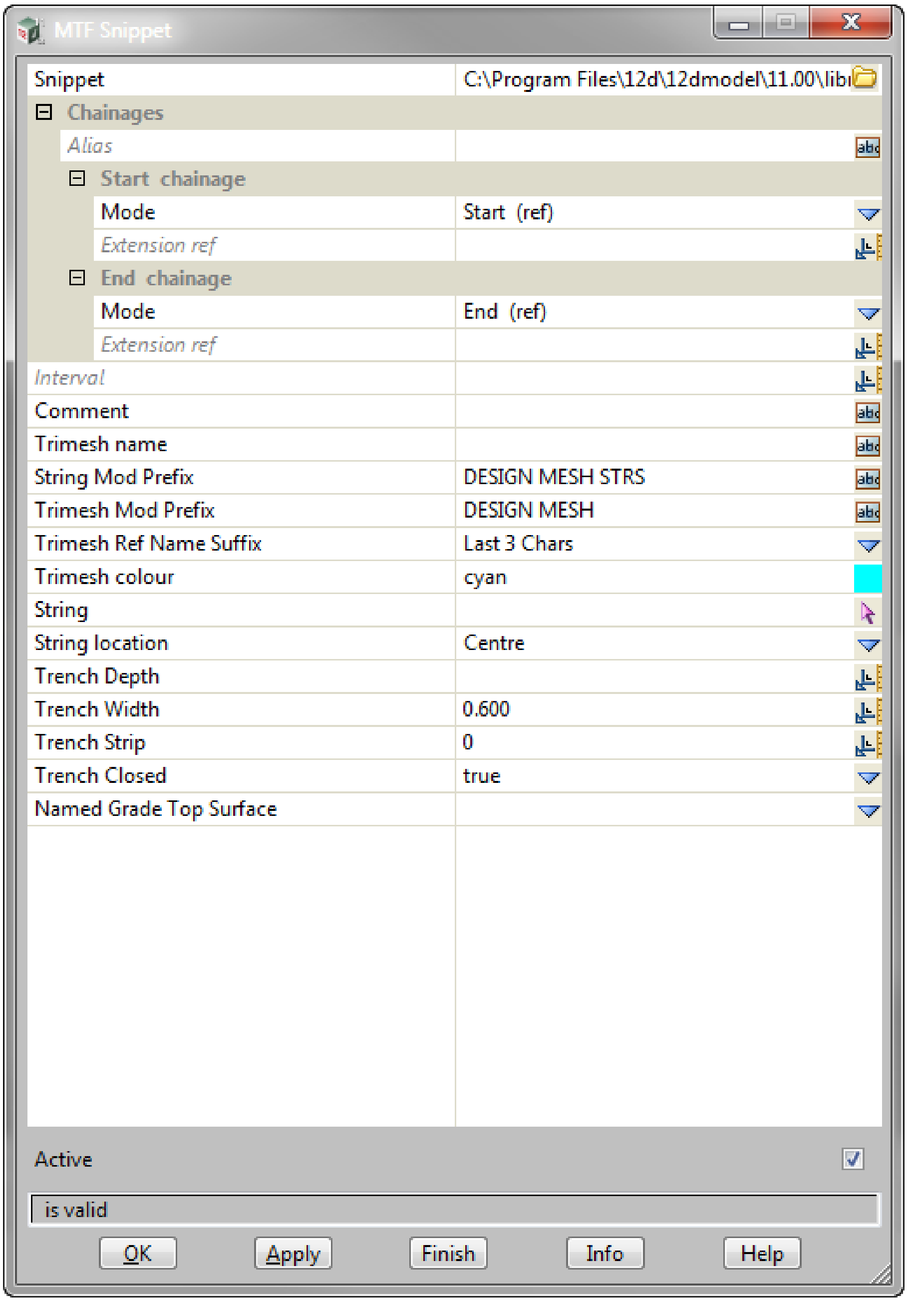
This Snippet is used to create a trench pavement, using External strings NOT created in your MTF.

It is linked to the Start Link at a specified location of left, centre or right.

A depth can be specified either to another link or a set value.

The top surface of the trench is defined by a Named Grade

The Reference string name can be used as part of the model naming convention.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Snippet	snippet box	*.mtfsnippet,	*.mtfsnippetc files

snippet to run.

Alias, Start Chainage, End Chainage, Interval

defines the start and end chainages to apply a snippet.

For information on these panel fields, see 17.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, Extra start, Extra End, Active, OK, Apply

For information on these panel fields, see 17.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Trimesh Name	input		

Trimesh name that is used in model names below

String Mod Prefix	input		DESIGN MESH STRS
Trimesh Mod Prefix	input		DESIGN MESH
Trimesh Ref Name Suffix	choice box	Last 3 Chars	All Chars
			Last Char
			Last 2 Chars
			Last 3 Chars
			Last 4 Chars
			None

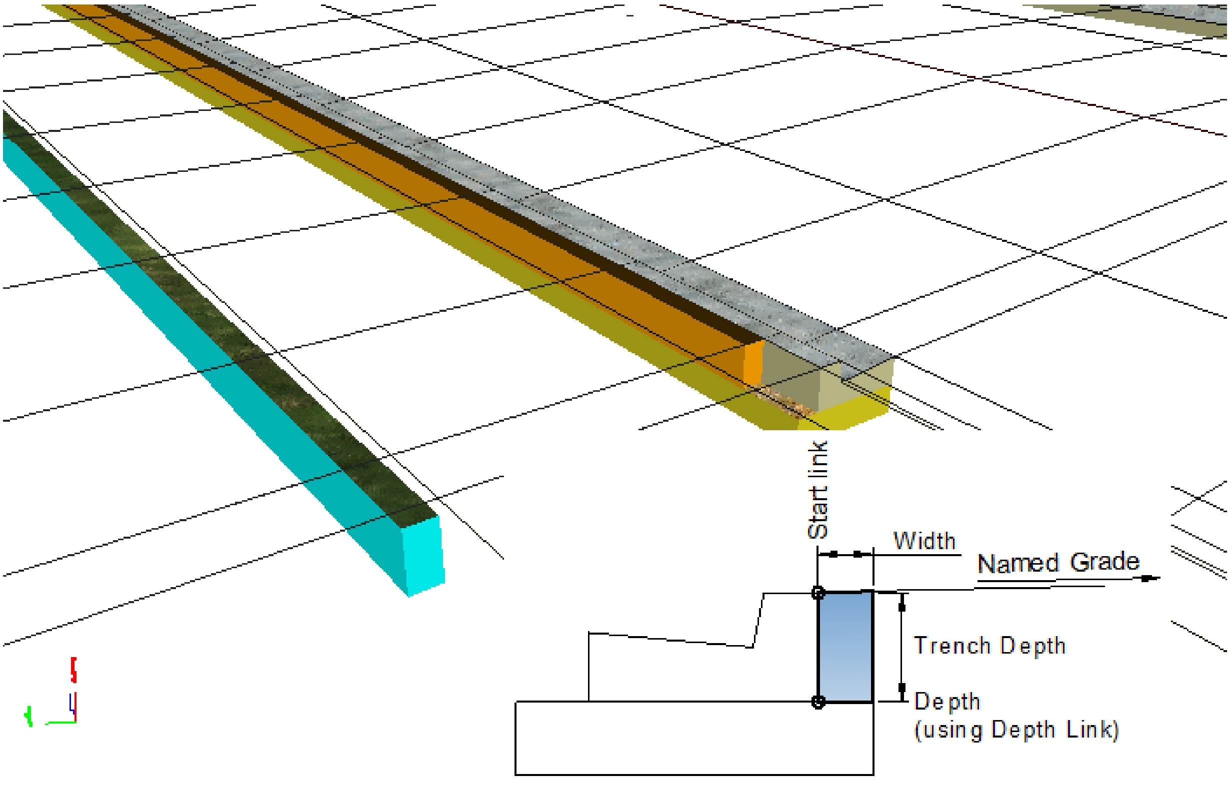
(Model syntax: String Mod Prefix Trimesh Name Trimesh Ref Name Suffix)

Trimesh Colour	colour box	cyan	various
String	select		
String Location	choice box	Centre	Centre
			LHS
			RHS
Trench Depth	real		
Trench Width	real		0.6
Trench Strip	real		0

(Allows a strip depth measured from the top of the trench)

Trench Closed	choice box	true	true / false
Named Grade Top Surface	choice box		

(The top of the trench will follow this grade while the bottom of the trench is horizontal)



For more information on snippets, see 21.5 Defining and Using Snippets.

Continue to 20.4.3.5.3 TRI_2PT_PAV_DES_LAY or return to 20.4.3.5 12d Supplied Snippets or 20.4.3 Snippets.

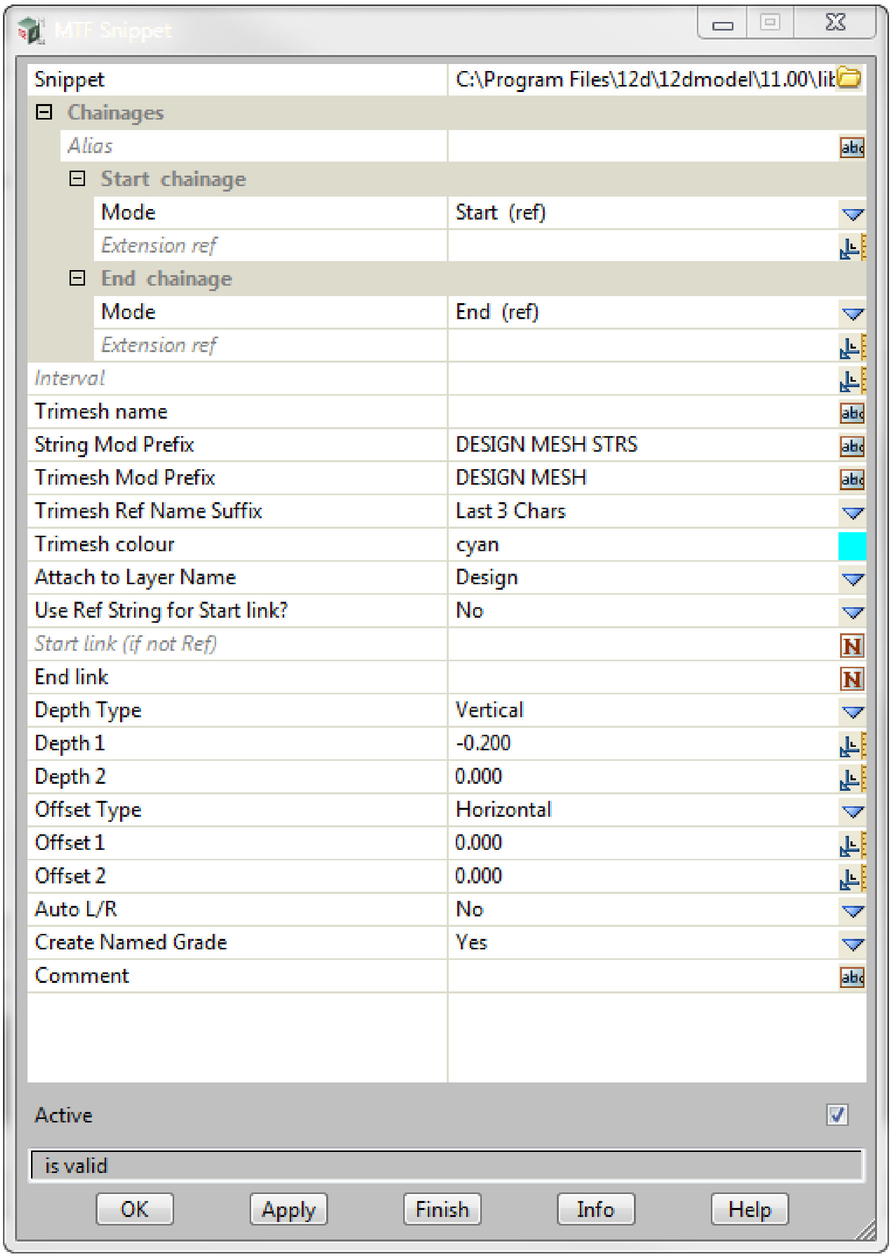
20.4.3.5.3 TRI_2PT_PAV_DES_LAY

This Snippet is used to create typical pavement, using <Design> or <Layer> strings created in your MTF.

It is linked to the Start and End Links at a depth specified.

Offsets from these links can be set and measured Horizontally or down the slope created from the two links.

The Reference string name can be used as part of the model naming convention.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Snippet	snippet box	*.mtfsnippet,	*.mtfsnippetc files

snippet to run.

Alias, Start Chainage, End Chainage, Interval

defines the start and end chainages to apply a snippet.

For information on these panel fields, see 17.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, Extra start, Extra End, Active, OK, Apply

For information on these panel fields, see 17.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Trimesh Name input

Trimesh name that is used in model names below

String Mod Prefix input DESIGN MESH STRS

Trimesh Mod Prefix input DESIGN MESH

Trimesh Ref Name Suffix choice box Last 3 Chars All Chars

Last Char

Last 2 Chars

Last 3 Chars

Last 4 Chars

None

(Model syntax: String Mod Prefix Trimesh Name Trimesh Ref Name Suffix)

Trimesh Colour colour boxcyanvarious

Attach to Layer Name choice box Design various

Use Ref String for Start Link? choice box No Yes / No

Start Link (if not Ref) name

(Optional if use ref string is set to Yes)

End Link name

Depth Type choice box Vertical Vertical

Normal

Top Surface

Bottom Surface

Depth 1 real-0.2

Depth 2 real

(Optional...if not used Depth 1 is set throughout)

Offset Type choice box Horizontal Horizontal

On Slope

Offset 1 real 0

Offset 2 real 0

Auto L/R choice box No Yes / No

if set to <No> then the Start and End Links remain unchanged.

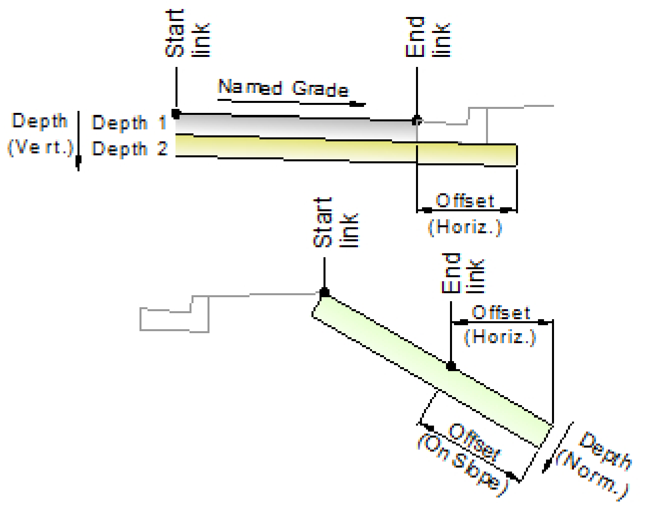
If set to <Yes> then the Start and End Links need the suffix for L/R removed.

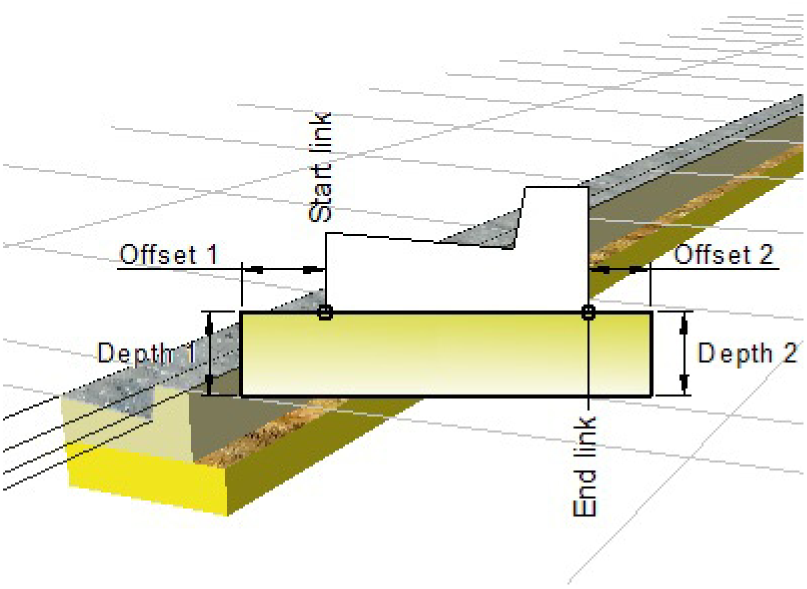
*e.g   Any <Design> string names like **ESL** would need the <L> removed*

*e.g   Any <Layer> string names like **PVBALRO1** would need the <LRO1> removed*

Create Named Grades   choice box   No   Yes / No

if set to <Yes> then a Named Grade in the format <Start Link Name _ End Link Name> is created for use in the MTF by other modifiers





For more information on snippets, see 21.5 Defining and Using Snippets.

Continue to 20.4.3.5.4 TRI_2PT_PAV_EXT_STRS or return to 20.4.3.5 12d Supplied Snippets or 20.4.3 Snippets.

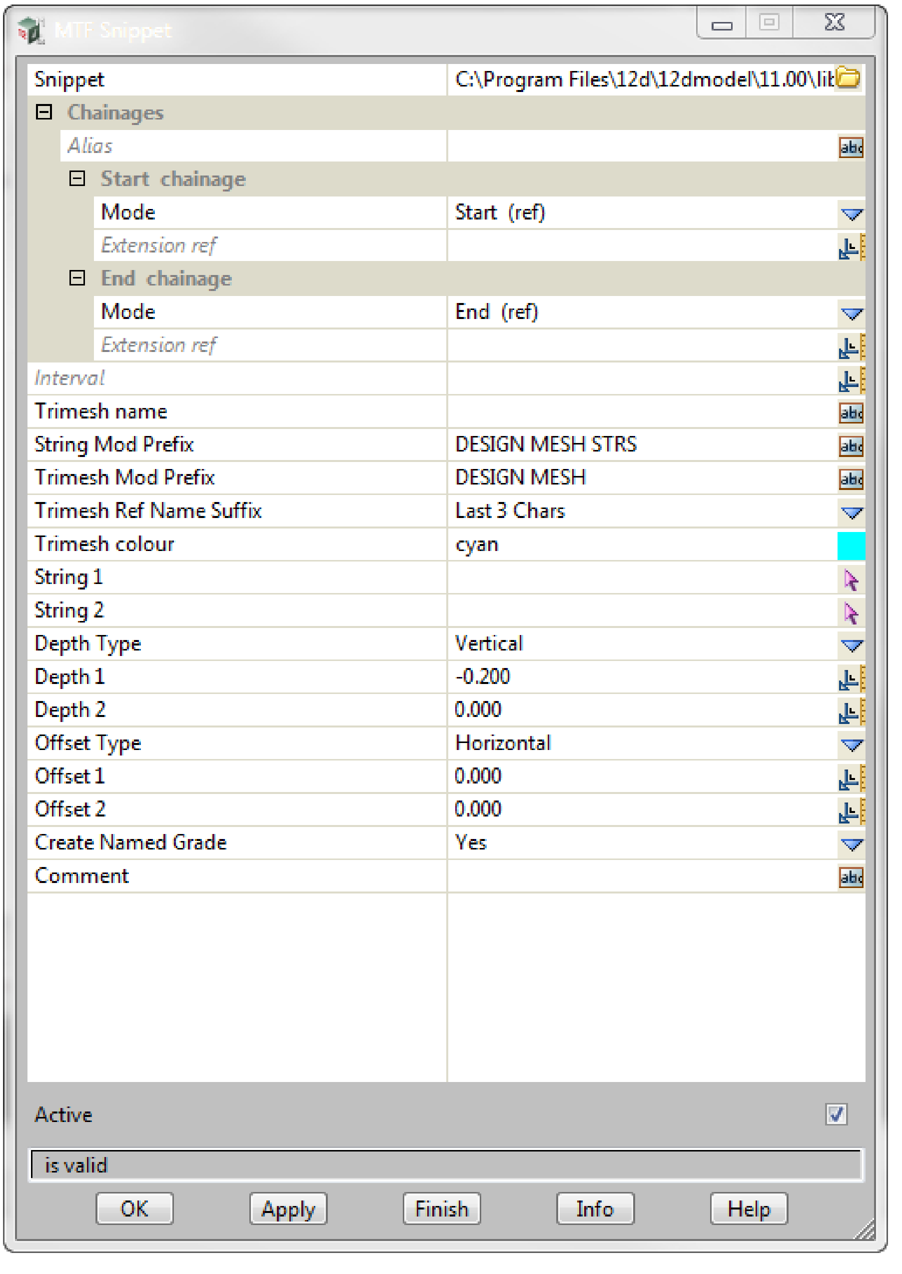
20.4.3.5.4 TRI_2PT_PAV_EXT_STRS

This Snippet is used to create typical pavement, using external strings NOT created in your MTF

It is linked to the String 1 and String 2 at a depth specified.

Offsets from these links can be set and measured Horizontally or down the slope created from the two links.

The Reference string name can be used as part of the model naming convention.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Snippet	snippet box	*.mtfsnippet,	*.mtfsnippetc files

snippet to run.

Alias, Start Chainage, End Chainage, Interval

defines the start and end chainages to apply a snippet.

For information on these panel fields, see 17.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, Extra start, Extra End, Active, OK, Apply

For information on these panel fields, see 17.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Trimesh Name	input		

Trimesh name that is used in model names below

String Mod Prefix	input		DESIGN MESH STRS
Trimesh Mod Prefix	input		DESIGN MESH
Trimesh Ref Name Suffix	choice box	Last 3 Chars	All Chars
			Last Char
			Last 2 Chars
			Last 3 Chars
			Last 4 Chars
			None

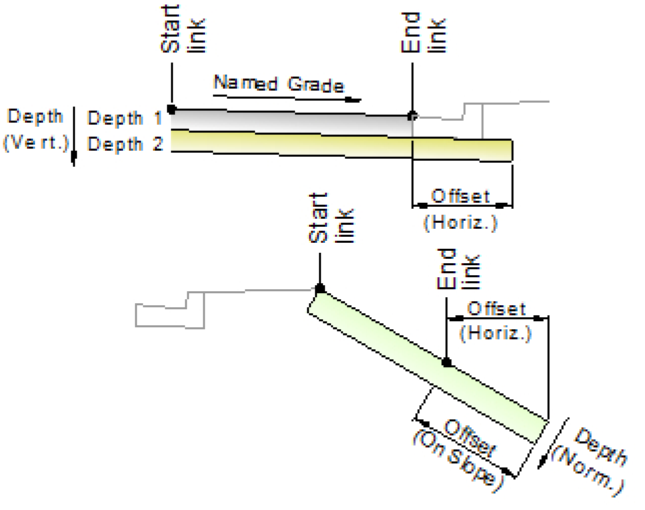
(Model syntax: String Mod Prefix   Trimesh Name   Trimesh Ref Name Suffix)

Trimesh Colour	colour box	cyan	various
Attach to Layer Name	choice box	Design	various
Use Ref String for Start Link?	choice box	No	Yes / No
String 1	select		
String 2	select		
Depth Type	choice box	Vertical	Vertical
			Normal
			Top Surface
			Bottom Surface
Depth 1	real		-0.2
Depth 2	real		

(Optional...if not used Depth 1 is set throughout)

Offset Type	choice box	Horizontal	Horizontal
			On Slope
Offset 1			real0
Offset 2			real0
Create Named Grades	choice box	No	Yes / No

If set to <Yes> then a Named Grade in the format <String 1 _ String 2> is created for use in the MTF by other modifiers.



For more information on snippets, see [21.5 Defining and Using Snippets](#).

Continue to [20.4.3.5.5 TRI_2PT_PAV_NAMED_GRADES](#) or return to [20.4.3.5 12d Supplied Snippets](#) or [20.4.3 Snippets](#).

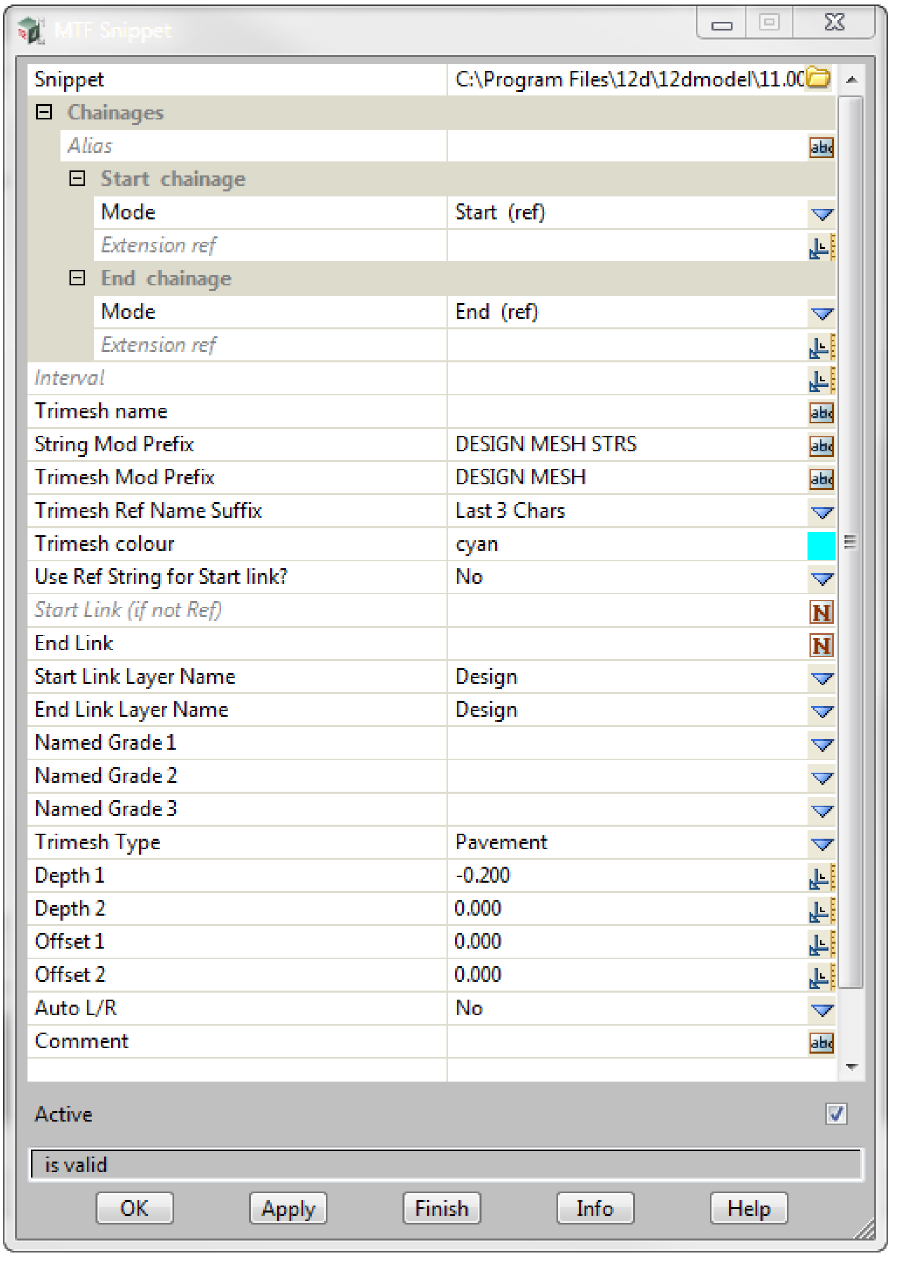
20.4.3.5.5 TRI_2PT_PAV_NAMED_GRADES

This Snippet is used to create typical pavement, using <Design> or <Layer> strings created in your MTF.

It is linked to the Start and End Links at a depth specified.

Offsets from these links can be set and measured Horizontally or down the slope created from the two links

The Reference string name can be used as part of the model naming convention.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Snippet	snippet box	*.mtfsnippet,	*.mtfsnippetc files

snippet to run.

Alias, Start Chainage, End Chainage, Interval

defines the start and end chainages to apply a snippet.

For information on these panel fields, see 17.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, Extra start, Extra End, Active, OK, Apply

For information on these panel fields, see 17.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Trimesh Name input

Trimesh name that is used in model names below

String Mod Prefix	input		DESIGN MESH STRS
Trimesh Mod Prefix	input		DESIGN MESH
Trimesh Ref Name Suffix	choice box	Last 3 Chars	All Chars
			Last Char
			Last 2 Chars
			Last 3 Chars
			Last 4 Chars
			None

(Model syntax: String Mod Prefix Trimesh Name Trimesh Ref Name Suffix)

Trimesh Colour	colour box	cyan	various
Attach to Layer Name	choice box	Design	various
Use Ref String for Start Link?	choice box	No	Yes / No
Start Link (if not Ref)	name		

(Optional if use ref string is set to Yes)

End Link	name		
Start Link Layer Name	choice box	Design	various
End Link Layer Name	choice box	Design	various
Named Grade 1	choice box	Design	various

(To define the top surface)

Named Grade 2	choice box	Design	various

(To define the typical batter slope)

Named Grade 3	choice box	Design	various

(To define the bottom surface)

Trimesh Type	choice box	Pavement	Pavement / Bottom Surface
Depth 1	real		-0.2
Depth 2	real		

(Optional...if used is measure Normal to Named Grade 2)

Offset 1	real		0
Offset 2	real		0

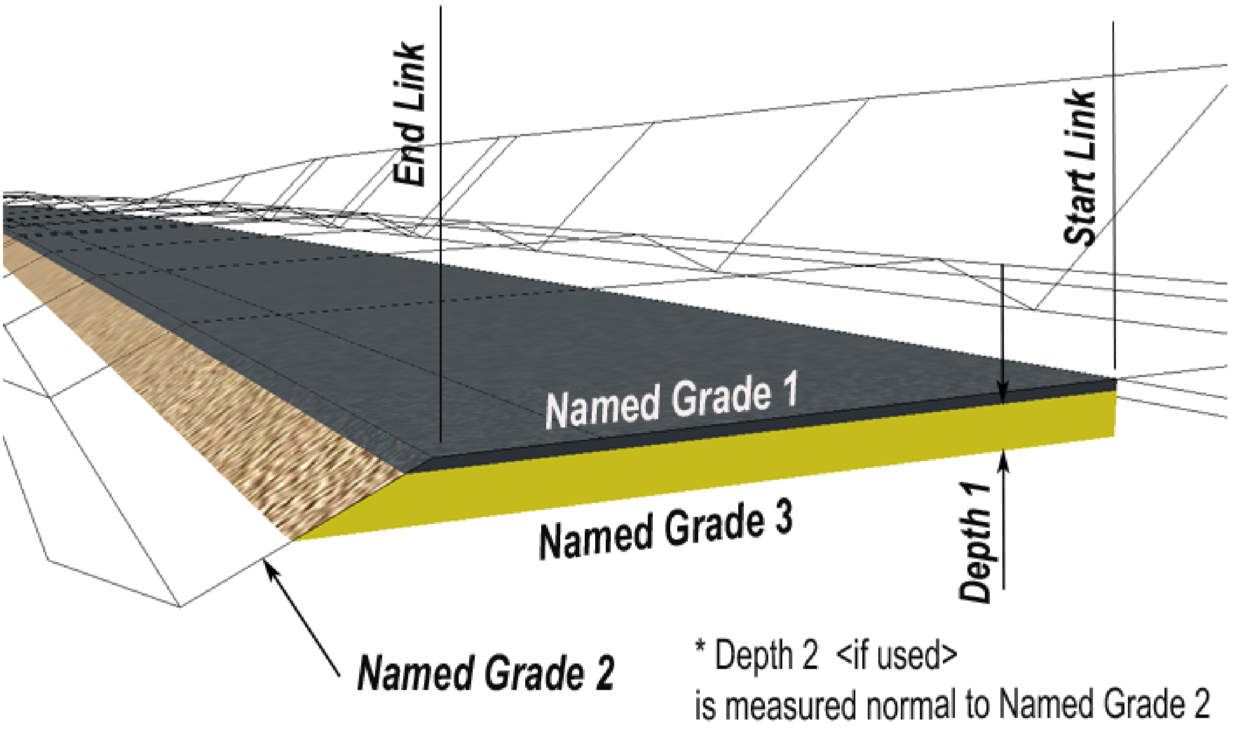
Auto L/R	choice box	No	Yes / No

If set to <No> then the Start and End Links remain unchanged.

If set to <Yes> then the Start and End Links need the suffix for L/R removed.

e.g Any <Design> string names like ***ESL*** *would need the <L> removed*

e.g Any <Layer> string names like ***PVBALRO1*** *would need the <LRO1> removed*



For more information on snippets, see 21.5 Defining and Using Snippets.

Continue to 20.4.3.5.6 TRI_2PT_PAV_NAMED_GRADES_EXT_STRS or return to 20.4.3.5 12d Supplied Snippets or 20.4.3 Snippets.

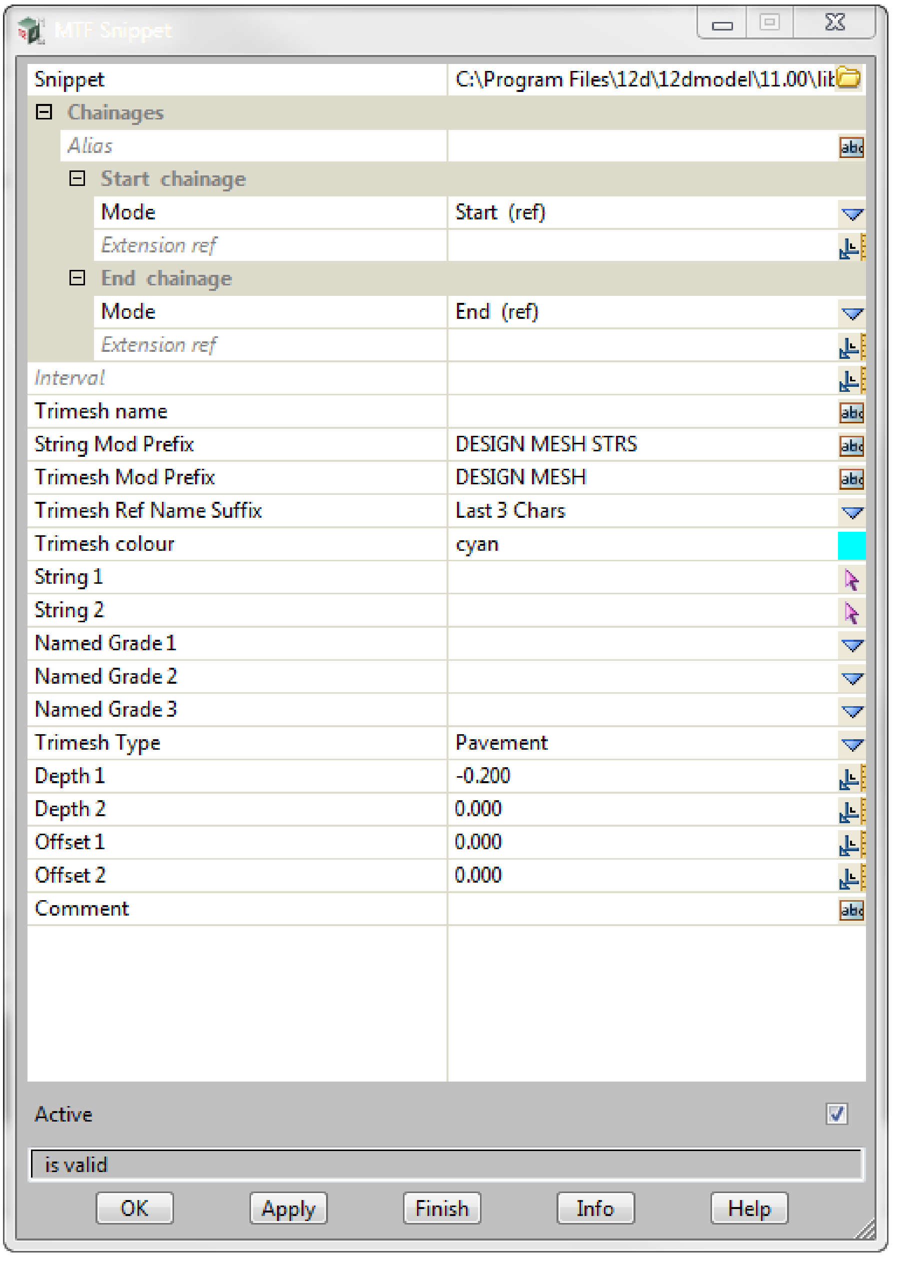
20.4.3.5.6 TRI_2PT_PAV_NAMED_GRADES_EXT_STRS

This Snippet is used to create typical pavement, using External strings NOT created in your MTF.

It is linked to the String 1 and String 2 at a depth specified.

Offsets from these links can be set and measured Horizontally.

The Reference string name can be used as part of the model naming convention.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Snippet	snippet box	*.mtfsnippet,	*.mtfsnippet files

snippet to run.

Alias, Start Chainage, End Chainage, Interval

defines the start and end chainages to apply a snippet.

For information on these panel fields, see 17.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, Extra start, Extra End, Active, OK, Apply

For information on these panel fields, see 17.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Trimesh Name	input		

Trimesh name that is used in model names below

String Mod Prefix	input		DESIGN MESH STRS
Trimesh Mod Prefix	input		DESIGN MESH
Trimesh Ref Name Suffix	choice box	Last 3 Chars	All Chars
			Last Char
			Last 2 Chars
			Last 3 Chars
			Last 4 Chars
			None

(Model syntax: String Mod Prefix Trimesh Name Trimesh Ref Name Suffix)

Trimesh Colour	colour box	cyan	various
String 1	select		
String 2	select		
Named Grade 1	choice box	Design	various

(To define the top surface)

Named Grade 2	choice box	Design	various

(To defined the typical batter slope)

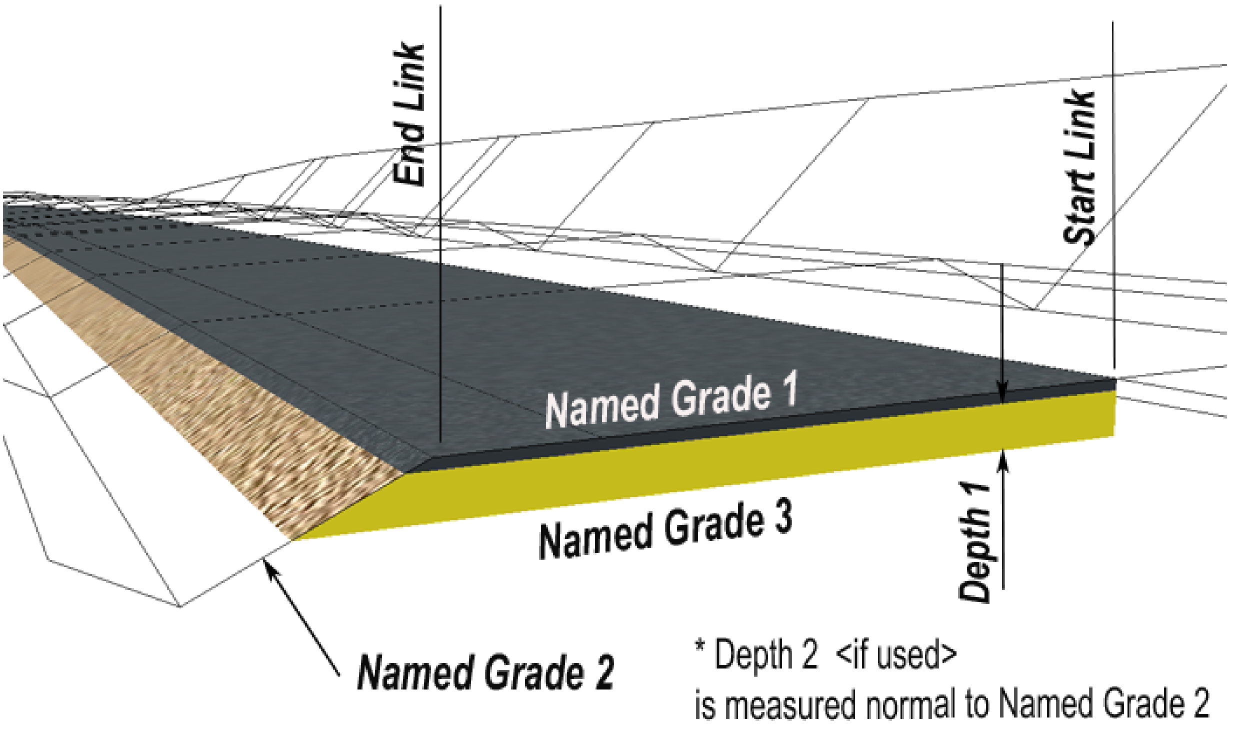
Named Grade 3	choice box	Design	various

(To define the bottom surface)

Trimesh Type	choice box	Pavement	Pavement / Bottom Surface
Depth 1	real		-0.2
Depth 2	real		

(Optional...if used is measure Normal to Named Grade 2)

Offset 1	real		0
Offset 2	real		0



For more information on snippets, see 21.5 Defining and Using Snippets.

Continue to 20.4.3.5.7 TRI_2PT_PAV_STOP_NAMED_GRADES or return to 20.4.3.5 12d Supplied Snippets or 20.4.3 Snippets.

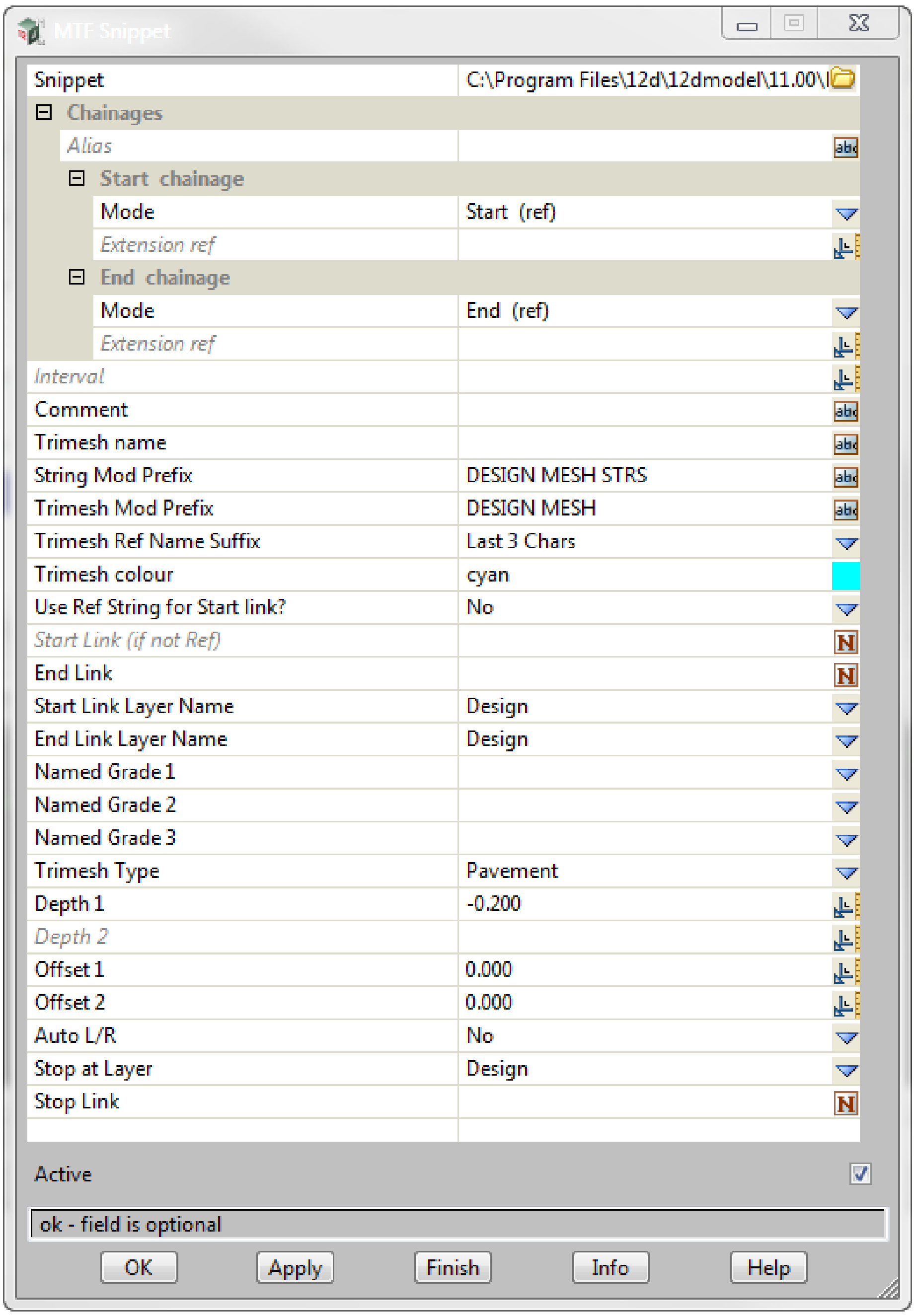
20.4.3.5.7 TRI_2PT_PAV_STOP_NAMED_GRADES

This Snippet is used to create typical pavement, using <Design> or <Layer> strings created in your MTF.

It is linked to the Start and End Links at a depth specified.

Offsets from these links can be set and measured Horizontally or down the slope created from the two links.

The Reference string name can be used as part of the model naming convention.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Snippet	snippet box		*.mtfsnippet, *.mtfsnippetc files

snippet to run.

Alias, Start Chainage, End Chainage, Interval

defines the start and end chainages to apply a snippet.

For information on these panel fields, see 17.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, Extra start, Extra End, Active, OK, Apply

For information on these panel fields, see 17.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Trimesh Name	input		

Trimesh name that is used in model names below

String Mod Prefix	input		DESIGN MESH STRS
Trimesh Mod Prefix	input		DESIGN MESH
Trimesh Ref Name Suffix	choice box	Last 3 Chars	All Chars
			Last Char
			Last 2 Chars
			Last 3 Chars
			Last 4 Chars
			None

(Model syntax: String Mod Prefix Trimesh Name Trimesh Ref Name Suffix)

Trimesh Colour	colour box	cyan	various
Attach to Layer Name	choice box	Design	various
Use Ref String for Start Link?	choice box	No	Yes / No
Start Link (if not Ref)	name		

(Optional if use ref string is set to Yes)

End Link	name		
Start Link Layer Name	choice box	Design	various
End Link Layer Name	choice box	Design	various
Named Grade 1	choice box	Design	various

(To define the top surface)

Named Grade 2	choice box	Design	various

(To defined the typical batter slope)

Named Grade 3	choice box	Design	various

(To define the bottom surface)

Trimesh Type	choice box	Pavement	Pavement / Bottom Surface
Depth 1	real		-0.2
Depth 2	real		

(Optional...if used is measure Normal to Named Grade 2)

Offset 1	real		0

Offset 2	real		0
Auto L/R	choice box	No	Yes / No

If set to <No> then the Start and End Links remain unchanged.

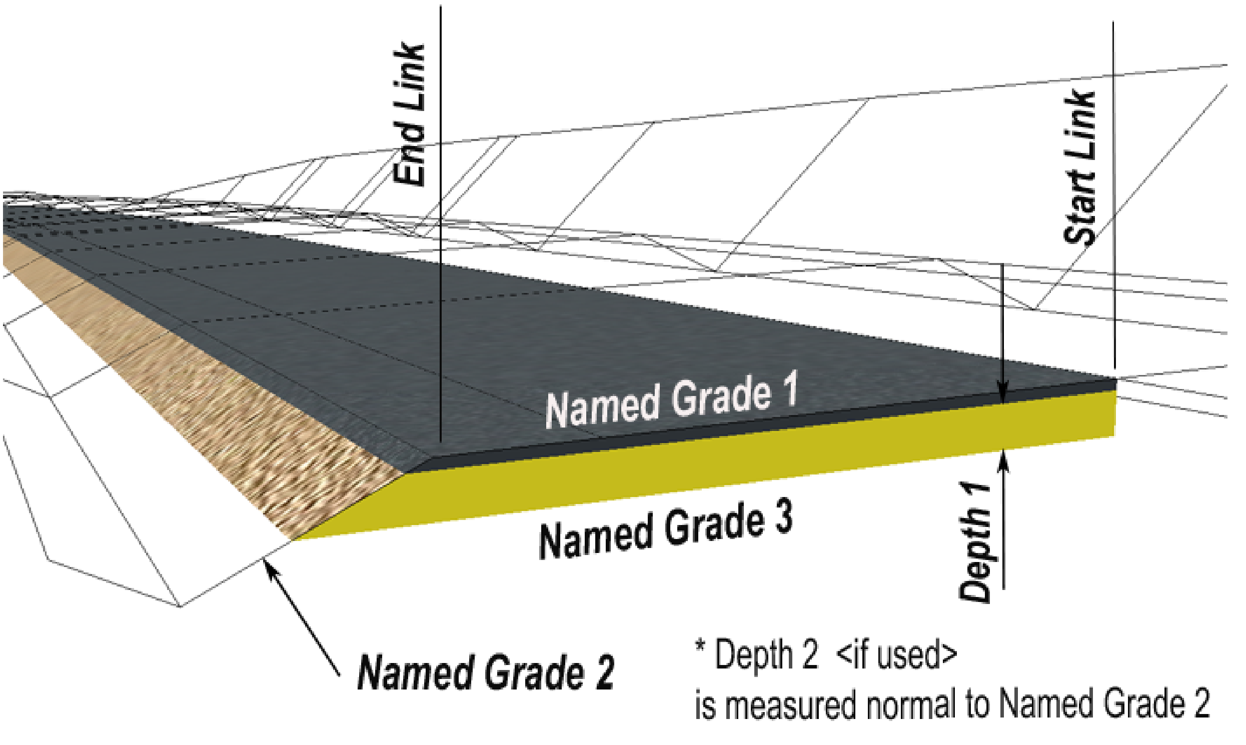
If set to <Yes> then the Start and End Links need the suffix for L/R removed.

*e.g Any <Design> string names like **ESL** would need the <L> removed*

*e.g Any <Layer> string names like **PVBALRO1** would need the <LRO1> removed*

Stop at Layer	choice box	Design	various
Stop Link	name		

(The stop layer and link are to be used to stop any pavement layer under a specific point such as a drain invert or interface point, rather than extending through at "Named Grade 2")



For more information on snippets, see 21.5 Defining and Using Snippets.

Continue to 20.4.3.5.8 TRI_2PT_PAV_STOP_NAMED_GRADES_EXT_STRS or return to 20.4.3.5 12d Supplied Snippets or 20.4.3 Snippets.

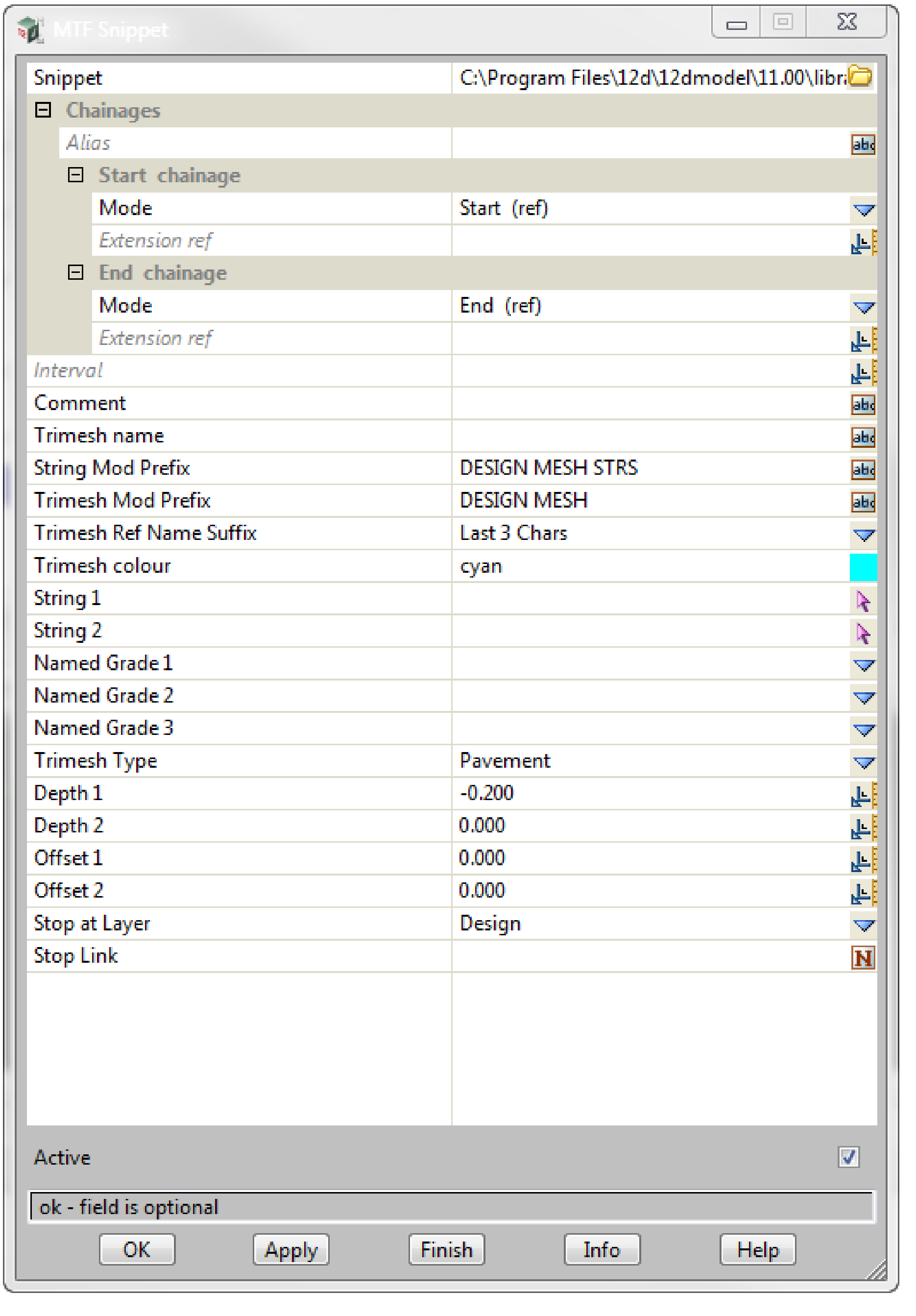
20.4.3.5.8 TRI_2PT_PAV_STOP_NAMED_GRADES_EXT_STRS

This Snippet is used to create typical pavement, using External strings NOT created in your MTF.

It is linked to the String 1 and String 2 at a depth specified.

Offsets from these links can be set and measured Horizontally.

The Reference string name can be used as part of the model naming convention.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Snippet	snippet box	*.mtfsnippet,	*.mtfsnippet files

snippet to run.

Alias, Start Chainage, End Chainage, Interval

defines the start and end chainages to apply a snippet.

For information on these panel fields, see 17.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, Extra start, Extra End, Active, OK, Apply

For information on these panel fields, see 17.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Trimesh Name	input		

Trimesh name that is used in model names below

String Mod Prefix	input		DESIGN MESH STRS
Trimesh Mod Prefix	input		DESIGN MESH
Trimesh Ref Name Suffix	choice box	Last 3 Chars	All Chars
			Last Char
			Last 2 Chars
			Last 3 Chars
			Last 4 Chars
			None

(Model syntax: String Mod Prefix Trimesh Name Trimesh Ref Name Suffix)

Trimesh Colour	colour box	cyan	various
String 1	select		
String 2	select		
Named Grade 1	choice box	Design	various

(To define the top surface)

Named Grade 2	choice box	Design	various

(To defined the typical batter slope)

Named Grade 3	choice box	Design	various

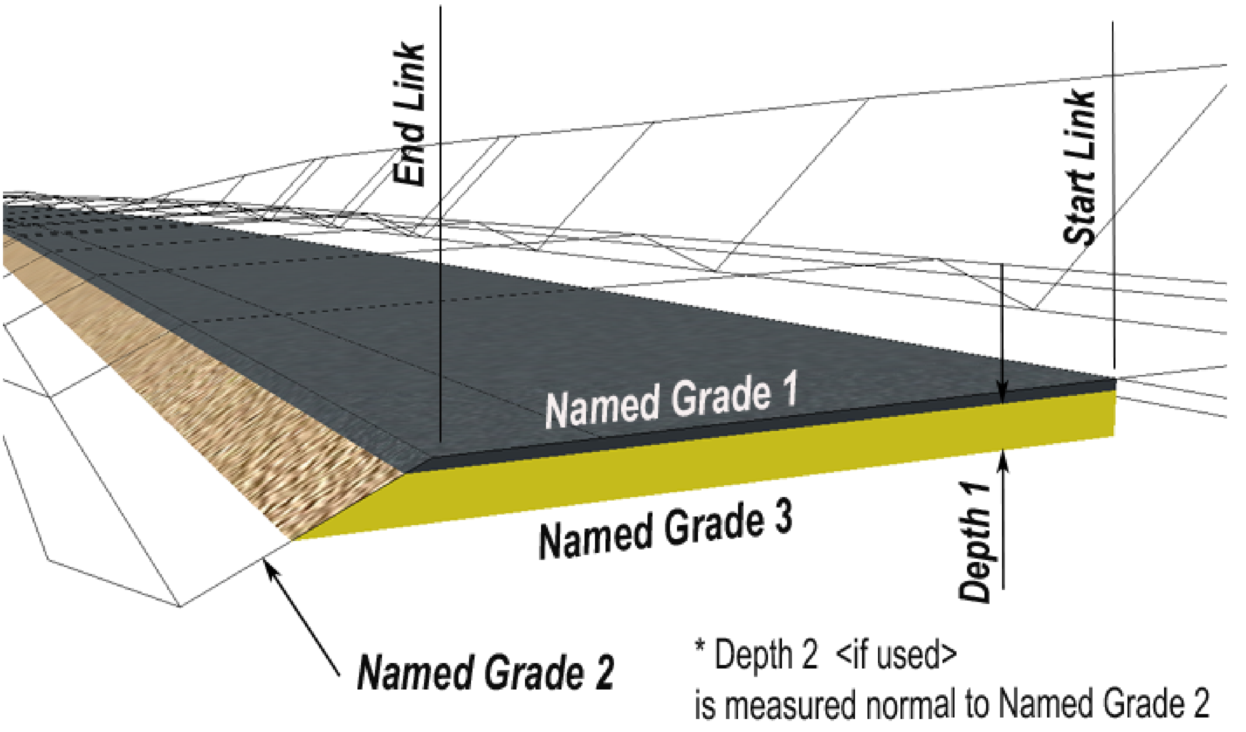
(To define the bottom surface)

Trimesh Type	choice box	Pavement	Pavement / Bottom Surface
Depth 1	real		-0.2
Depth 2	real		

(Optional...if used is measure Normal to Named Grade 2)

Offset 1	real		0
Offset 2	real		0
Stop at Layer	choice box	Design	various
Stop Link	name		

(The stop layer and link are to be used to stop any pavement layer under a specific point such as a drain invert or interface point, rather than extending through at "Named Grade 2")



For more information on snippets, see 21.5 Defining and Using Snippets.

Continue to 20.4.3.5.9 TRI_2PT_STRUCT_FILL or return to 20.4.3.5 12d Supplied Snippets or 20.4.3 Snippets.

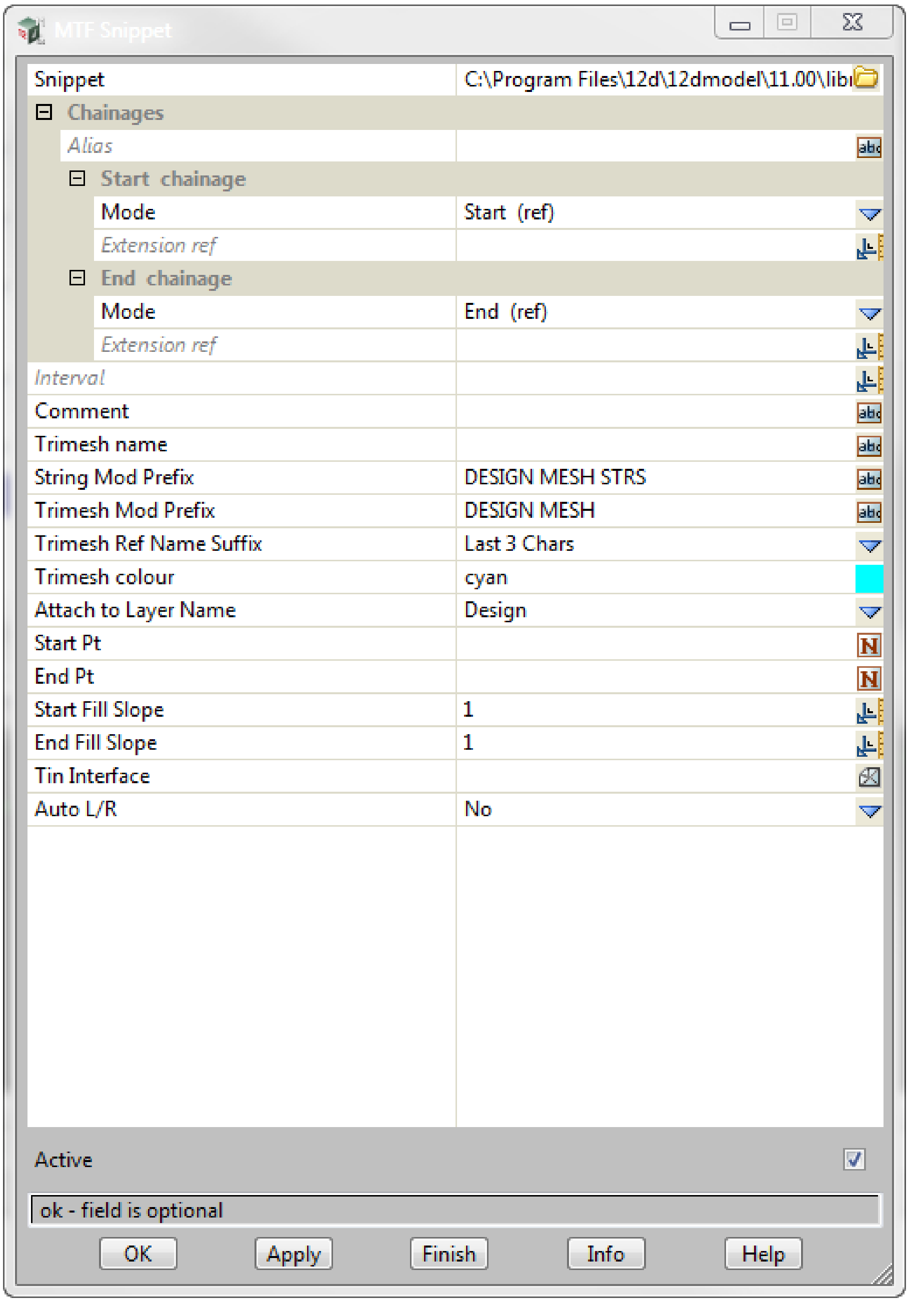
20.4.3.5.9 TRI_2PT_STRUCT_FILL

This Snippet is used to create typical pavement, using <Design> or <Layer> strings created in your MTF.

It is linked to the Start and End Links at a slope specified.

Slopes are for fill, so no sign is required for either slope.

The Reference string name can be used as part of the model naming convention.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Snippet	snippet box	*.mtfsnippet,	*.mtfsnippetc files

snippet to run.

Alias, Start Chainage, End Chainage, Interval

defines the start and end chainages to apply a snippet.

For information on these panel fields, see 17.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, Extra start, Extra End, Active, OK, Apply

For information on these panel fields, see 17.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Trimesh Name	input		

Trimesh name that is used in model names below

String Mod Prefix	input		DESIGN MESH STRS
Trimesh Mod Prefix	input		DESIGN MESH
Trimesh Ref Name Suffix	choice box	Last 3 Chars	All Chars
			Last Char
			Last 2 Chars
			Last 3 Chars
			Last 4 Chars
			None

(Model syntax: String Mod Prefix Trimesh Name Trimesh Ref Name Suffix)

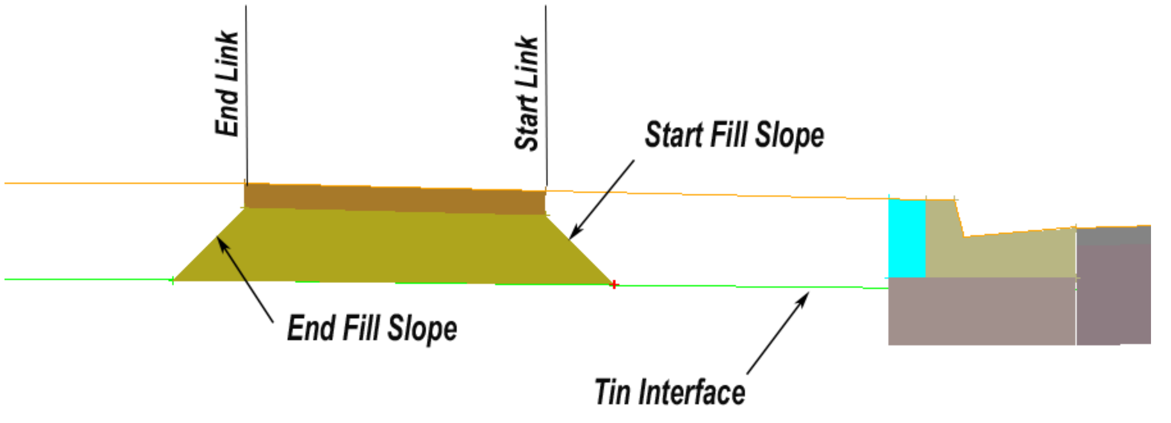
Trimesh Colour	colour box	cyan	various
Attach to Layer Name	choice box	Design	various
Start Link	name		
End Link	name		
Start Fill Slope	real		1
End Fill Slope	real		1
Tin Interface	tin		
Auto L/R	choice box	No	Yes / No

if set to <No> then the Start and End Links remain unchanged.

If set to <Yes> then the Start and End Links need the suffix for L/R removed.

*e.g Any <Design> string names like **ESL** would need the <L> removed*

*e.g Any <Layer> string names like **PVBALRO1** would need the <LRO1> removed*



For more information on snippets, see 21.5 Defining and Using Snippets.

Continue to 20.4.3.5.10 TRI_3PT_PAV_DES_LAY or return to 20.4.3.5 12d Supplied Snippets or 20.4.3 Snippets.

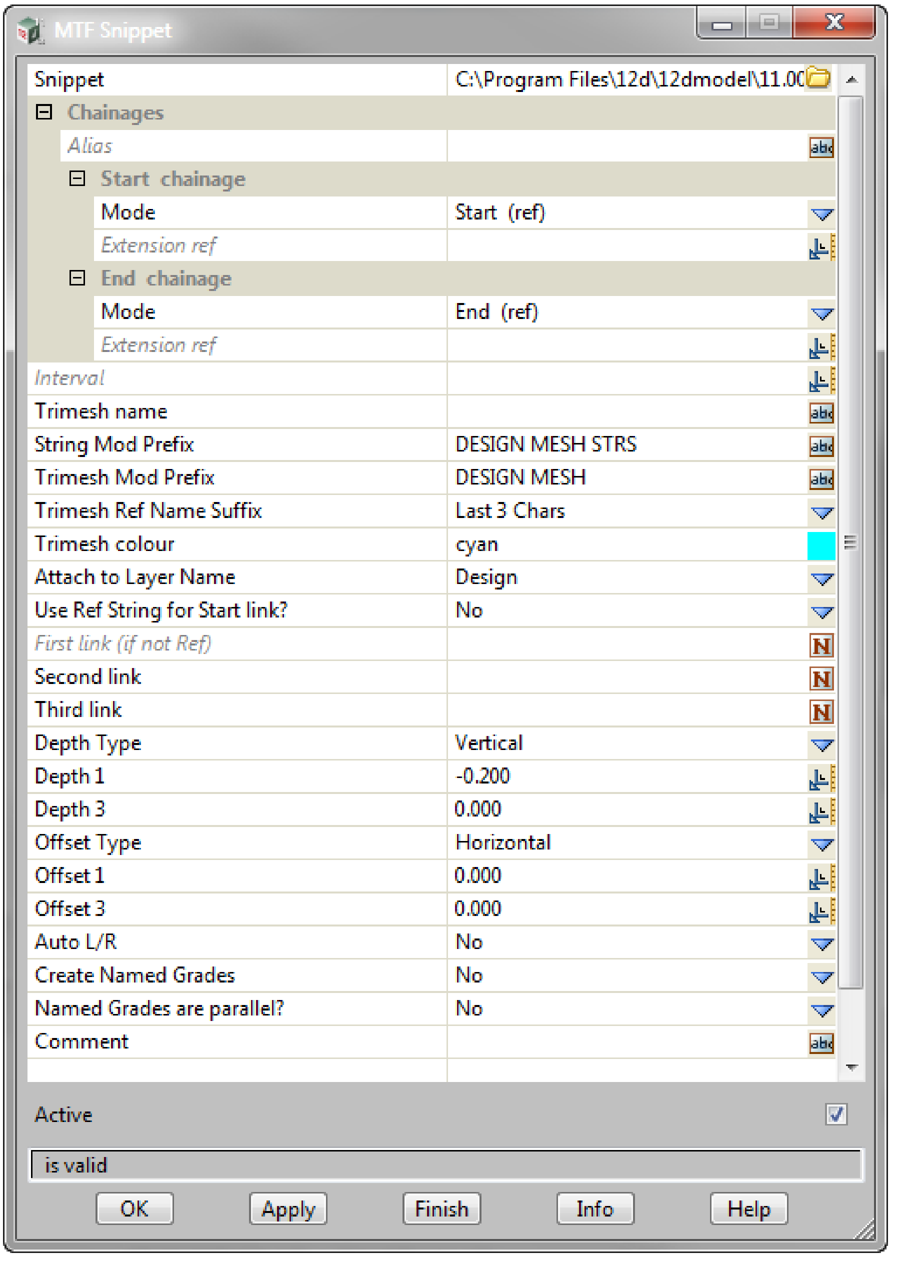
20.4.3.5.10 TRI_3PT_PAV_DES_LAY

This Snippet is used to create typical pavement, using <Design> or <Layer> strings created in your MTF.

It is linked to the First, Second and Third Links at depths specified.

Offsets from these links can be set and measured Horizontally or down the slope created from the links.

The Reference string name can be used as part of the model naming convention.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Snippet	snippet box	*.mtfsnippet,	*.mtfsnippetc files

snippet to run.

Alias, Start Chainage, End Chainage, Interval

defines the start and end chainages to apply a snippet.

For information on these panel fields, see 17.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, Extra start, Extra End, Active, OK, Apply

For information on these panel fields, see 17.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Trimesh Name	input		

Trimesh name that is used in model names below

String Mod Prefix	input		DESIGN MESH STRS
Trimesh Mod Prefix	input		DESIGN MESH
Trimesh Ref Name Suffix	choice box	Last 3 Chars	All Chars
			Last Char
			Last 2 Chars
			Last 3 Chars
			Last 4 Chars
			None

(Model syntax: String Mod Prefix Trimesh Name Trimesh Ref Name Suffix)

Trimesh Colour	colour box	cyan	various
Attach to Layer Name	choice box	Design	various
Use Ref String for Start Link?	choice box	No	Yes / No
First Link (if not Ref)	name		

(Optional if use ref string is set to Yes)

Second Link	name		
Third Link	name		
Depth Type	choice box	Vertical	Vertical
			Normal
			Top Surface
			Bottom Surface
Depth 1	real		-0.2

(Measured from First Link)

Depth 3	real		-0.2

(Measured from Third Link)

Offset Type	choice box	Horizontal	Horizontal
			On Slope
Offset 1	real		0

(Measured from First Link)

Offset 3	real		0

(Measured from Third Link)

Auto L/R choice box No Yes / No

If set to <No> then the Start and End Links remain unchanged.

If set to <Yes> then the Start and End Links need the suffix for L/R removed.

e.g Any <Design> string names like ***ESL*** *would need the <L> removed*

e.g Any <Layer> string names like ***PVBALRO1*** *would need the <LRO1> removed*

Create Named Grades choice box No Yes / No

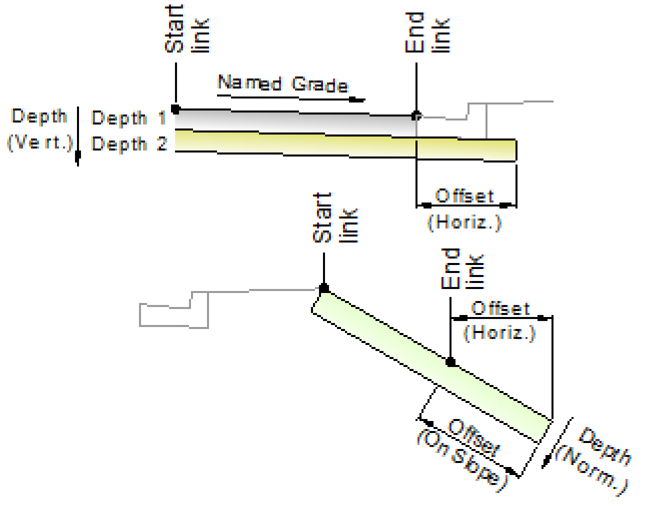
if set to <Yes> then a Named Grade in the format <Start Link Name _ End Link Name> is created for use in the MTF by other modifiers

Named Grades are Parallel? choice box No Yes / No

(This is a decision by the user in case the three links form one grade.

If ***No*** *then the middle point is calculated as an intersection of two grades, using "Depth 1" & "Depth 3".*

If ***Yes*** *then the middle point is calculated from the Second Link at the "Depth 1")*



For more information on snippets, see 21.5 Defining and Using Snippets.

Continue to 20.4.3.5.11 TRI_3PT_PAV_EXT_STRS or return to 20.4.3.5 12d Supplied Snippets or 20.4.3 Snippets.

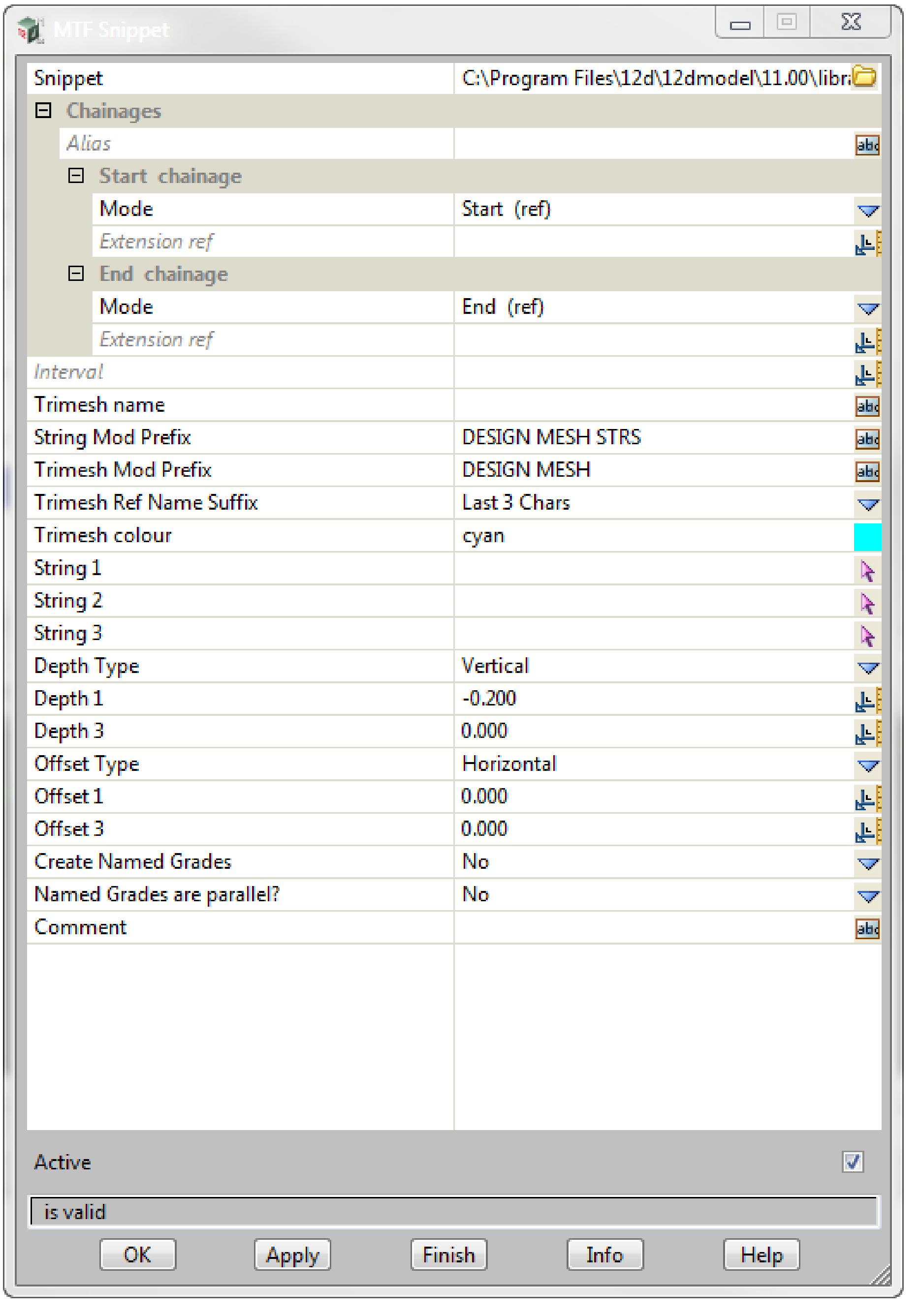
20.4.3.5.11 TRI_3PT_PAV_EXT_STRS

This Snippet is used to create typical pavement, using External strings NOT created in your MTF

It is linked to the three strings at depths specified.

Offsets from these links can be set and measured Horizontally or down the slope created from the links.

The Reference string name can be used as part of the model naming convention.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Snippet	snippet box	*.mtfsnippet,	*.mtfsnippetc files

snippet to run.

Alias, Start Chainage, End Chainage, Interval

defines the start and end chainages to apply a snippet.

For information on these panel fields, see 17.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, Extra start, Extra End, Active, OK, Apply

For information on these panel fields, see 17.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Trimesh Name	input		

Trimesh name that is used in model names below

String Mod Prefix	input		DESIGN MESH STRS
Trimesh Mod Prefix	input		DESIGN MESH
Trimesh Ref Name Suffix	choice box	Last 3 Chars	All Chars
			Last Char
			Last 2 Chars
			Last 3 Chars
			Last 4 Chars
			None

(Model syntax: String Mod Prefix Trimesh Name Trimesh Ref Name Suffix)

Trimesh Colour	colour box	cyan	various
Attach to Layer Name	choice box	Design	various
Use Ref String for Start Link?	choice box	No	Yes / No
String 1	select		
String 2	select		
String 3	select		
Depth Type	choice box	Vertical	Vertical
			Normal
			Top Surface
			Bottom Surface
Depth 1	real		-0.2

(Measured from First Link)

Depth 3	real		-0.2

(Measured from Third Link)

Offset Type	choice box	Horizontal	Horizontal
			On Slope
Offset 1	real		0

(Measured from First Link)

Offset 3	real		0

(Measured from Third Link)

Create Named Grades choice box No Yes / No

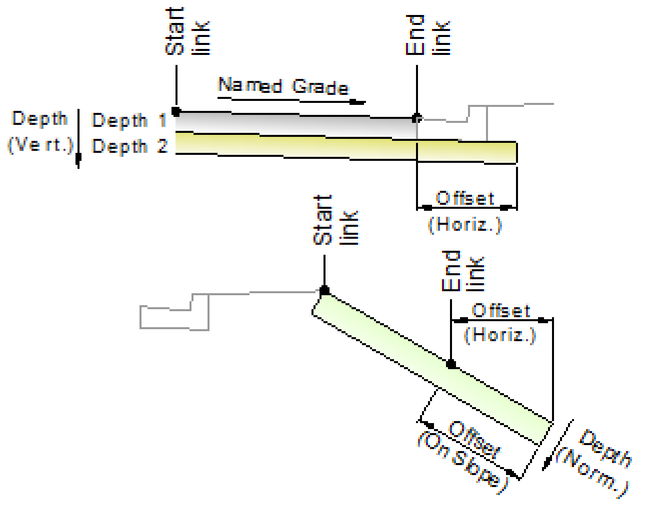
if set to <Yes> then a Named Grade in the format <Start Link Name _ End Link Name> is created for use in the MTF by other modifiers

Named Grades are Parallel? choice box No Yes / No

(This is a decision by the user in case the three links form one grade.

*If **No** then the middle point is calculated as an intersection of two grades, using "Depth 1" & "Depth 3".*

*If **Yes** then the middle point is calculated from the Second Link at the "Depth 1")*



For more information on snippets, see 21.5 Defining and Using Snippets.

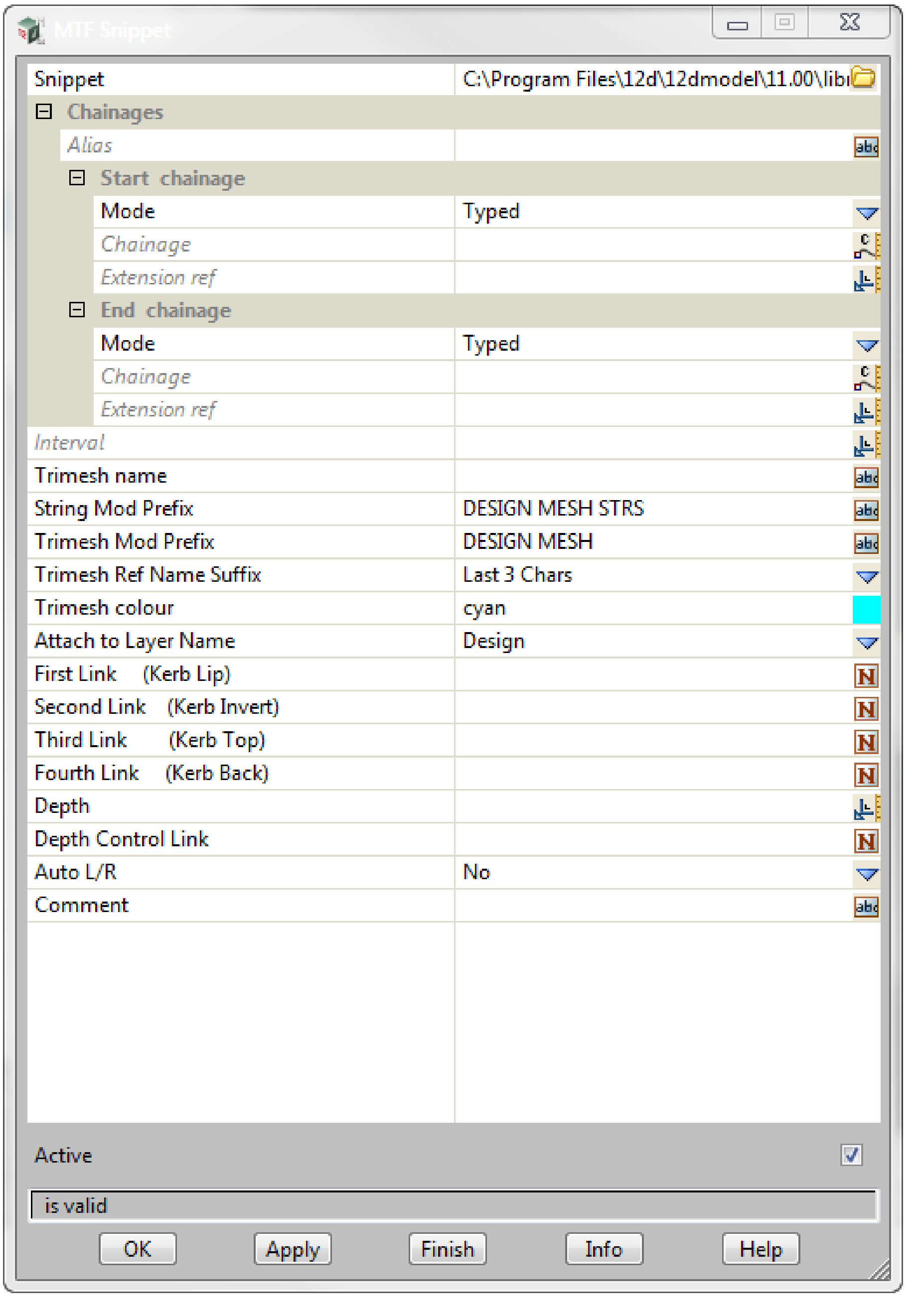
Continue to 20.4.3.5.12 TRI_KERB_PROFILE or return to 20.4.3.5 12d Supplied Snippets or 20.4.3 Snippets.

20.4.3.5.12 TRI_KERB_PROFILE

This Snippet is used to create typical kerb profile, using <Design> or <Layer> strings created in your MTF.

It is defined by four Links at a depth specified by one of those links.

The Reference string name can be used as part of the model naming convention.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Snippet	snippet box	*.mtfsnippet,	*.mtfsnippetc files

snippet to run.

Alias, Start Chainage, End Chainage, Interval

defines the start and end chainages to apply a snippet.

For information on these panel fields, see 17.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, Extra start, Extra End, Active, OK, Apply

For information on these panel fields, see 17.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Trimesh Name	input		

Trimesh name that is used in model names below

String Mod Prefix	input		DESIGN MESH STRS
Trimesh Mod Prefix	input		DESIGN MESH
Trimesh Ref Name Suffix	choice box	Last 3 Chars	All Chars
			Last Char
			Last 2 Chars
			Last 3 Chars
			Last 4 Chars
			None

(Model syntax: String Mod Prefix Trimesh Name Trimesh Ref Name Suffix)

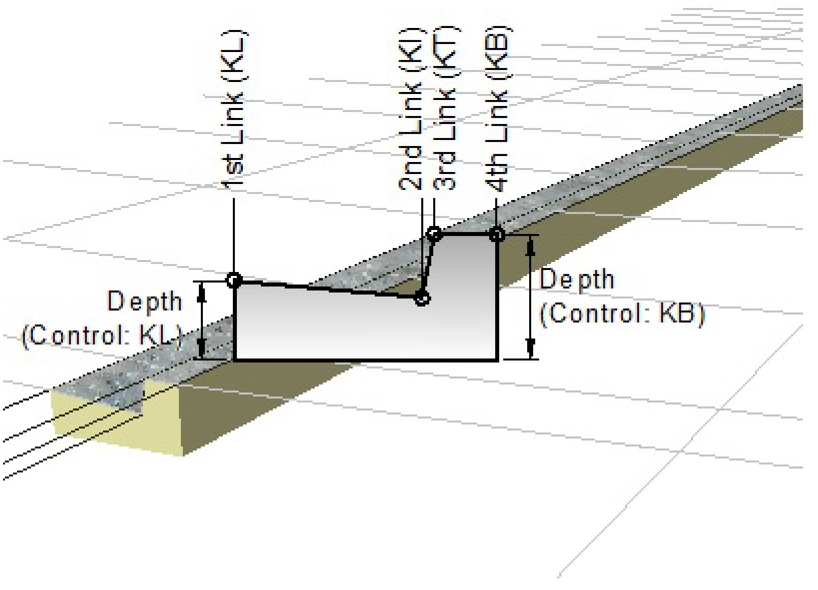
Trimesh Colour	colour box	cyan	various
Attach to Layer Name	choice box	Design	various
First Link	name		
Second Link	name		
Third Link	name		
Fourth Link	name		
Depth	real		
Depth Control Link	name		
Auto L/R	choice box	No	Yes / No

if set to <No> then the Start and End Links remain unchanged.

If set to <Yes> then the Start and End Links need the suffix for L/R removed.

*e.g Any <Design> string names like **ESL** would need the <L> removed*

*e.g Any <Layer> string names like **PVBALRO1** would need the <LRO1> removed*



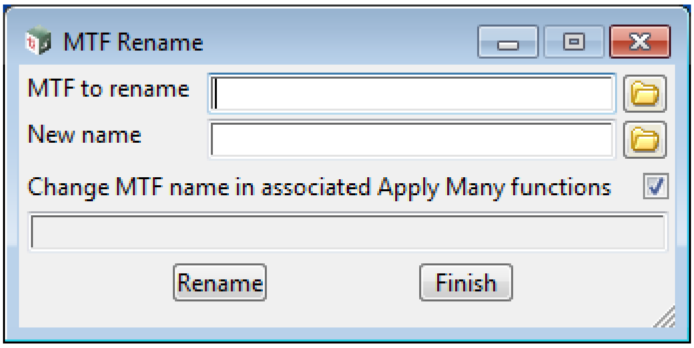
For more information on snippets, see 21.5 Defining and Using Snippets.

Return to 20.4.3.5 12d Supplied Snippets or 20.4.3 Snippets.

20.4.4 Rename MTF

Position of option on menu: **Design =>MTF =>Rename**

Selecting **Rename** brings up the **MTF Rename** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
MTF to rename	file box		*.mtf files

name of the MTF file to rename

New name	file box		*.mtf files

new name for the MTF file

Change MTF name is associated Apply MTF functions tick box

if ticked, rename the MTF in all functions that include it

Rename button

rename the MTF file to the new name.

Go to the next section 20.4.5 Apply MTF or return to 20.4 MTF.

20.4.5 Apply MTF

Position of option on menu: **Design =>MTF =>Apply MTF**

The **Apply MTF** option is used to apply a MTF file to a selected hinge string.

The option has already been described under **Design=>Apply=>Apply MTF.**

For the description of *Apply MTF*, please go to the section 20.3.2 Apply MTF Function.

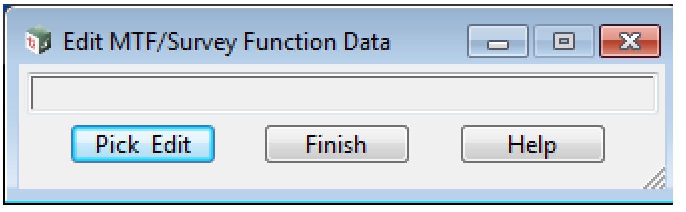
Go to the next section 20.4.6 Edit by String or return to 20.4 MTF.

20.4.6 Edit by String

Position of option on menu: **Design =>MTF =>Edit by string**

The **Edit by string** option is used to edit the ***MTF*** by selecting a string created by the *MTF* in an ***Apply MTF***.

Selecting **Edit by string** brings up the **Edit MTF/Survey Function Data** panel:



After selecting the **Pick Edit** button, a string created by MTF function is selected and the MTF editor is started for the MTF.

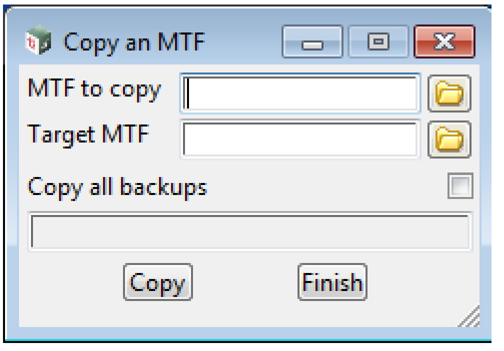
For information on the *MTF Editor*, go to the section 20.4.1.1 MTF Edit.

Go to the next section 20.4.7 Copy MTF or return to 20.4 MTF.

20.4.7 Copy MTF

Position of option on menu: **Design =>MTF =>Copy**

Selecting **Copy** brings up the **Copy an MTF** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
MTF to copy	file box		*.mtf files
name of the MTF file to copy.			
Target MTF	file box		*.mtf files
new name for the MTF file.			
Copy all backups	tick box		
if ticked, copy all the backup files of the MTF as well.			
Copy	button		
copy the MTF file to the new name.			

Go to the next section 20.4.8 Delete MTF or return to 20.4 MTF.

20.4.8 Delete MTF

Position of option on menu: **Design =>MTF =>Delete**

Selecting **Delete** brings up the **Delete an MTF** panel.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
MTF to delete	file box		*.mtf files

name of the MTF file to delete

Field Description	Type	Defaults	Pop-Up
Delete all backups	tick box		

if ticked, delete all the backup files of the MTF as well

Field Description	Type	Defaults	Pop-Up
Delete	button		

delete the MTF file.

Go to the next section 20.4.9 Create File or return to 20.4 MTF.

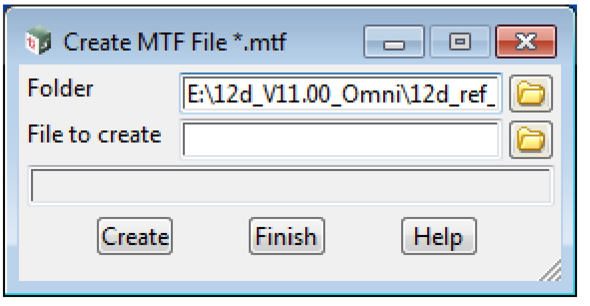
20.4.9 Create File

Position of option on menu: **Design =>MTF =>Create file**

The **MTF=>Create file** option is used to create a new ***Modifiers and Templates Files*** (*.mtf) with the text editor pointed to by the EDIT_4D environment variable.

The created file will already have all the section headers (key words and opening and closing brackets) in it.

Selecting **Create file** brings up the **Create MTF File *.mtf** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Folder	folder box	current folder	

name of the folder for the .mtf file.

File to create	file box		*.mtf files

name of the file to create. The new MTF file already has each of the section headers set up.

Create	button		

create an MTF file given by the **Folder** *and* **File to create** *panel fields. If the file given in the* **File to edit** *field already exists, then an error occurs.*

The text format for the MTF file is described in the section 21.10.1 Text Format of the MTF File.

Go to the next section 20.4.10 Edit MTF File or return to 20.4 MTF.

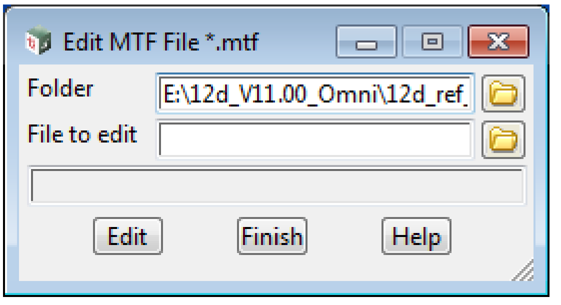
20.4.10 Edit MTF File

Position of option on menu: **Design =>MTF =>Edit file**

The **MTF=>Edit file** option is used to edit MTF files (*.mtf) with the text editor pointed to by the EDIT_4D environment variable.

The **MTF=>Edit file** option has two modes of operation - selecting the **MTF=>Edit file** itself, or by activating the **MTF=>Edit file** option's walk-right menu, **folder *.mtf**.

Selecting **MTF=>Edit file** itself brings up the **Edit a MTF File *.mtf** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Folder	folder box	current folder	

name of the folder for the .mtf file.

File to edit	file box		*.mtf files

name of the MTF file, in **Folder***, to edit.*

Edit	button

edit the MTF file given by the **Folder** *and* **File to edit** *panel fields by the text editor pointed to by the* EDIT_4D *environment variable. If the file given in the* **File to edit** *field does not exist, then a new file is created which already has each of the section headers set up.*

Similarly the **MTF=>Edit file** walk-right menu provides a list all the MTF files (files ending in .mtf) in the current folder. When a file is selected from the list, it is automatically loaded into the text editor.

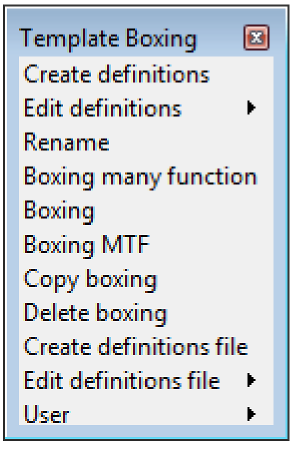
The text format for the MTF file is described in the section 21.10.1 Text Format of the MTF File.

Return to 20.4 MTF.

20.5 Boxing

Position of menu: **Design =>Boxing**

The **Boxing** walk-right menu is



In the **Apply MTF** option, **boxing** can be applied to the generated design sections by special commands in the **MTF**.

However, it is also possible to apply boxing as a post process to a model of x-sections as long as the x-sections were generated in the 4d super string format (this is done by both the Apply and Apply MTF options and the Cuts options - see 28.9.13 Cuts).

For more information on boxing, see 21.6 What is Boxing? and for more information on the various scenarios for generating boxing, see 21.6.2 Applying Boxing.

The rules for calculating different types of boxing are known as **Boxing Definitions** and *Boxing Definition* are created and stored in a **Boxing** file.

The **Boxing**, and the different **Boxing Definitions** contained within the file, are created/edited using the options **Create** and **Edit** from the **Template Boxing** menu.

The options **Boxing many function**, **Boxing many** and **Boxing** use the *Boxing Definitions* to generate boxing.

For the options see:

Create definitions	20.5.1 Create Boxing Definitions
Edit definitions	20.5.2 Edit Boxing Definitions
Rename	20.5.3 Rename Boxing File
Boxing many function	20.5.4 Boxing Many Function
Boxing	20.5.5 Apply Boxing
Boxing many	20.5.6 Apply Boxing Many
Copy boxing	20.5.7 Copy Boxing
Delete boxing	20.5.8 Delete a Boxing File
Create definitions file	20.5.9 Create Definitions Text File
Edit definitions file	20.5.4 Boxing Many Function

20.5.1 Create Boxing Definitions

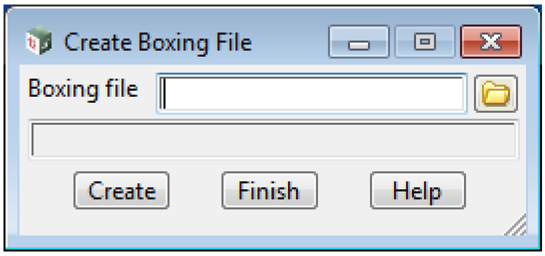
Position of option on menu: **Design =>Boxing =>Create definitions**

The rules for calculating different types of boxing are known as **Boxing Definitions** and *Boxing Definition* are created and stored in a **Boxing** file.

The **Boxing=>Create definitions** option is used to create a new **Boxing** file (*.bf), and then the **Edit Boxing Definitions** panel is opened for the **Boxing** file to create and add new *Boxing Definitions.* The **Edit Boxing Definitions** panel is then used to write the **Boxing** file to disk.

and can be edited using the **Boxing =>Edit definitions** option, or by standard text editor.

Selecting **Create definitions** displays the **Create Boxing File** panel.



The fields and buttons used in this panel have the following functions.

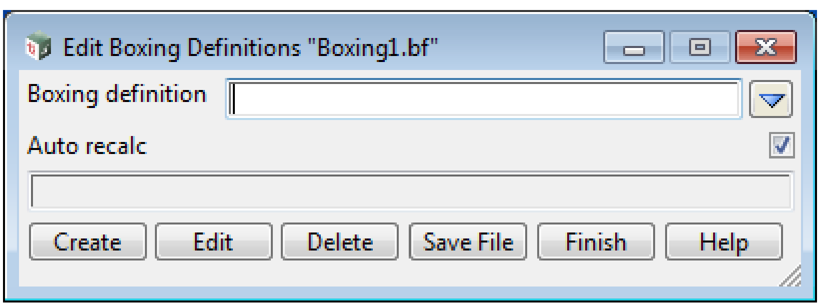
Field Description	Type	Defaults	Pop-Up
Boxing file	file box		*.bf files

name of the boxing file to create.

Create	button		

create a boxing file with name given by the **boxing file** *panel fields.*
If the file given in the **boxing file** *field does not exist, then the* **Edit Boxing Definitions** *panel is placed on the screen and is used to created and edit the boxing definitions for the boxing file.*
If the file already exists, then nothing will happen on selecting **Create**.

When **Create** is selected from the **Create Boxing File** panel, it brings up the **Edit Boxing Definitions** panel which is used to create and edit the boxing definitions, and to saved the boxing definitions to disk in the boxing file.



For information on the **Edit Boxing Definitions** panel, go to 21.7.2 Edit Boxing File.

The **Edit Boxing Definitions** panel is described in the section 21.7.2 Edit Boxing File.

Go to the next section 20.5.2 Edit Boxing Definitions or return to 20.5 Boxing.

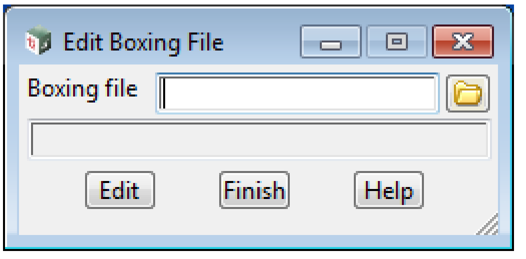
20.5.2 Edit Boxing Definitions

Position of option on menu: **Design =>Boxing =>Edit definitions**

The **Boxing =>Edit definitions** option is used to edit *Boxing* files (*.bf).

The **Boxing =>Edit definitions** option has two modes of operation - selecting the **Boxing =>Edit definitions** itself, or by activating the **Boxing =>Edit definitions** option's walk-right menu, **folder *.bf**.

Clicking on **Boxing=>Edit definitions** itself brings up the **edit boxing file** panel.



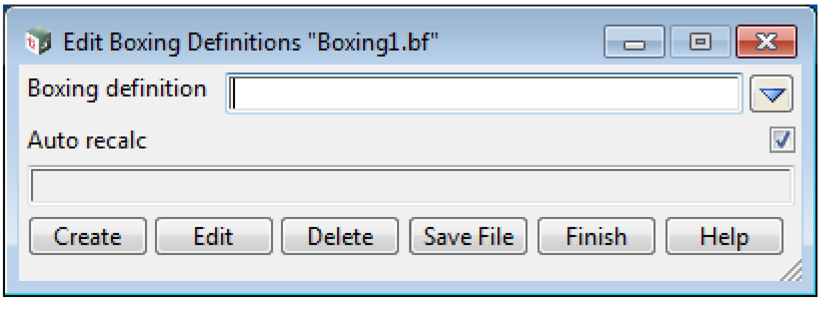
The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Boxing file	input		*.bf files

name of the boxing file to edit.

Edit	button		

edit the file given by **boxing file** *panel field.*
If the file given in the **boxing file** *field exists, then the* **Edit Boxing Definitions** *panel is brought up to create/edit the boxing definitions in the boxing file.*



Similarly, walking right on **Boxing=>Edit** provides a list all the boxing files (files ending in .bf) in the current folder and when a file is selected from the list, the **Edit Boxing Definitions** panel is brought up for the selected boxing file.

The **Edit Boxing Definitions** panel is described in the section 21.7.2 Edit Boxing File.

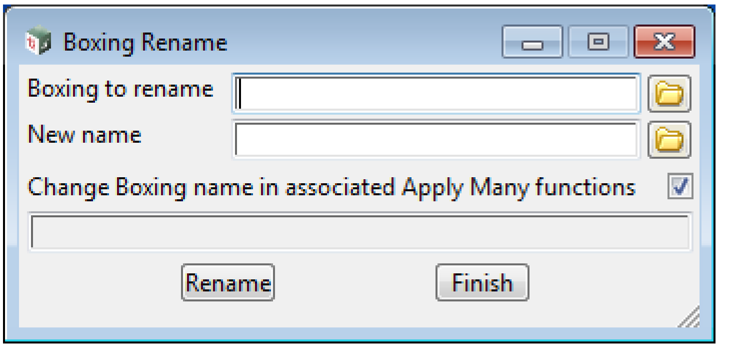
Go to the next section 20.5.3 Rename Boxing File or return to 20.5 Boxing.

20.5.3 Rename Boxing File

Position of option on menu: **Design =>Boxing =>Rename**

On selecting **Rename**, the **Boxing Rename** panel is displayed.

This panel can be used to change the names of existing boxing files.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Boxing to rename	boxing file box		*.bf files

name of the boxing file to be renamed.

New name	boxing file box		*.bf files

new name for the boxing file

Change Boxing name in associated Apply MTF functions tick box

if ticked, rename the bf file in all the Apply MTF functions that include it.

Rename button

Change the name of the boxing file in the **Boxing to rename** *field to the name given in the* **New name** *field.*

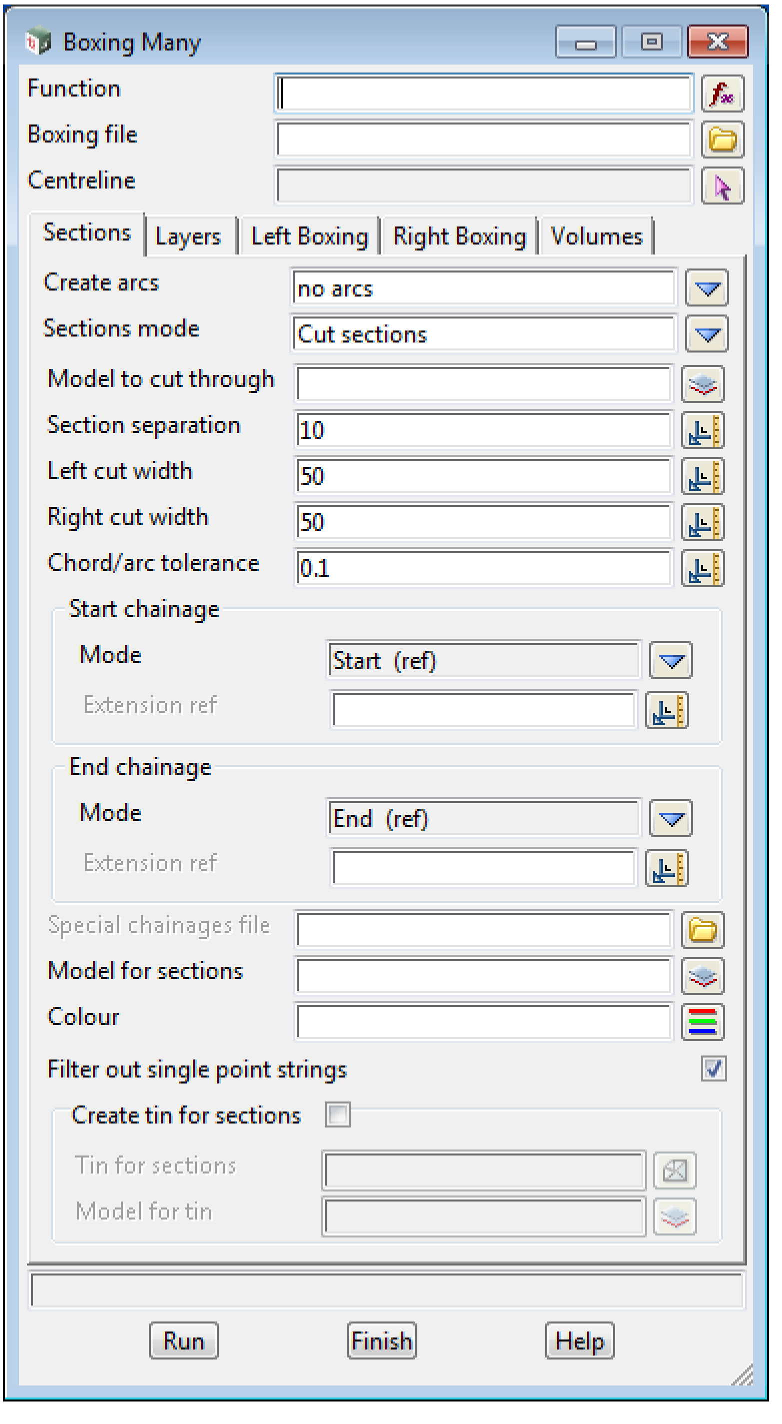
Go to the next section 20.5.4 Boxing Many Function or return to 20.5 Boxing.

20.5.4 Boxing Many Function

Position of option on menu: **Design =>Boxing =>Boxing many function**

This option is used to cut strings to create sections and then apply up to eight layers of boxing to the cut sections.

Selecting **Boxing many function** displays the **Boxing Many** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function	function box		available boxing functions

name of the Boxing function

Boxing file file box *.bf files

the Boxing file that contains the Boxing Definition that are be applied to the cut sections

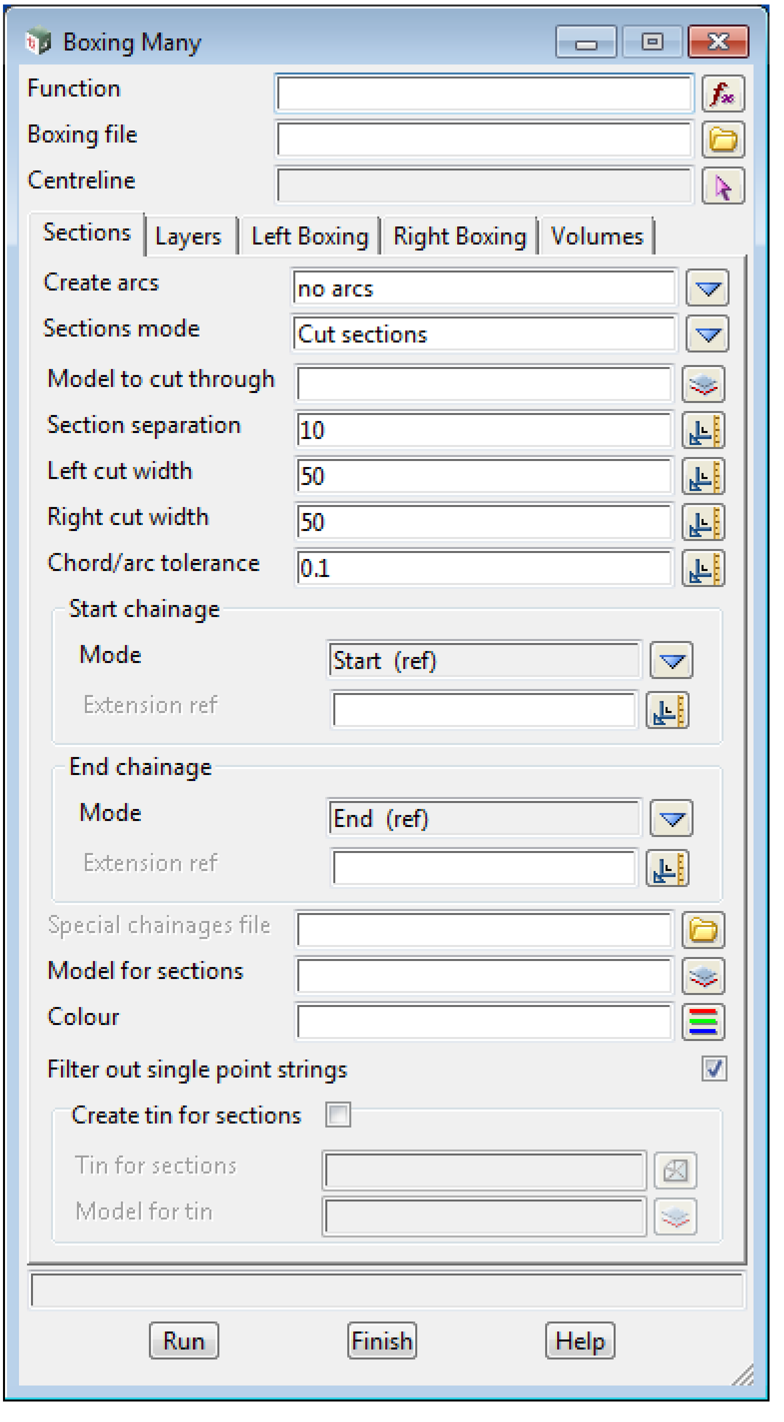
Centreline string select

string to use for chainage, and perpendicular, for cutting cross sections to apply the boxing to

For the information on each **Boxing Many** tab, go to

Sections tab
Layers tab
Left/Right Boxing tabs
Volumes tab

Sections tab



Sections mode choice box Cut sections, Use existing sections

if Cut section, sections are cut through the strings in the **Model to cut through** *and placed in* **Model for sections.**

If User existing sections, no sections are cut and the existing sections in **Model for sections** *are used to apply the boxing to.*

Model to cut through model box available models

model of strings to cut sections through

Section separation input 10

the distance along the selected centre line to generate plan section lines to be used to cut through the *model or view.*

Left/Right cut width input 50

the left/right distance to go out from the centre line for creating a section to cut through the strings.

Chord/arc tolerance input default chord/arc tolerance

the chord to arc tolerance to use on the selected string for determining how many plan sections are created around horizontal curves.

Start/End chainage choice box Start (ref)/End (ref)
Mode choice box Smart Chainge modes
Extension ref real box

the **Start/End chainage***,* **Mode** *and* **Extension ref** *are all used to define the start/end chainage on the selected centreline string.*
The sections for the cuts are restricted to between these start and end chainage.
If blank, the **Start/End chainage** *of the reference string is used*

For more information on Start/End Chainage Modes, see 21.2.1 MTF Hinge Modifiers

Special chainages file file box *.spf files

a file containing chainages, one per line, that are also used as extra chainages to create cross sections at.

Model for sections model box available models

model for the cut cross sections.

Colour colour box available colours

colour for the cut cross sections.

Filter out single point strings tick box

if ticked, don't save any sections that only have a single point in them

Create tin for sections tick box

if ticked, a tin of the cross sections in **Model for sections** *is created.*

Tin for sections tin box available tins

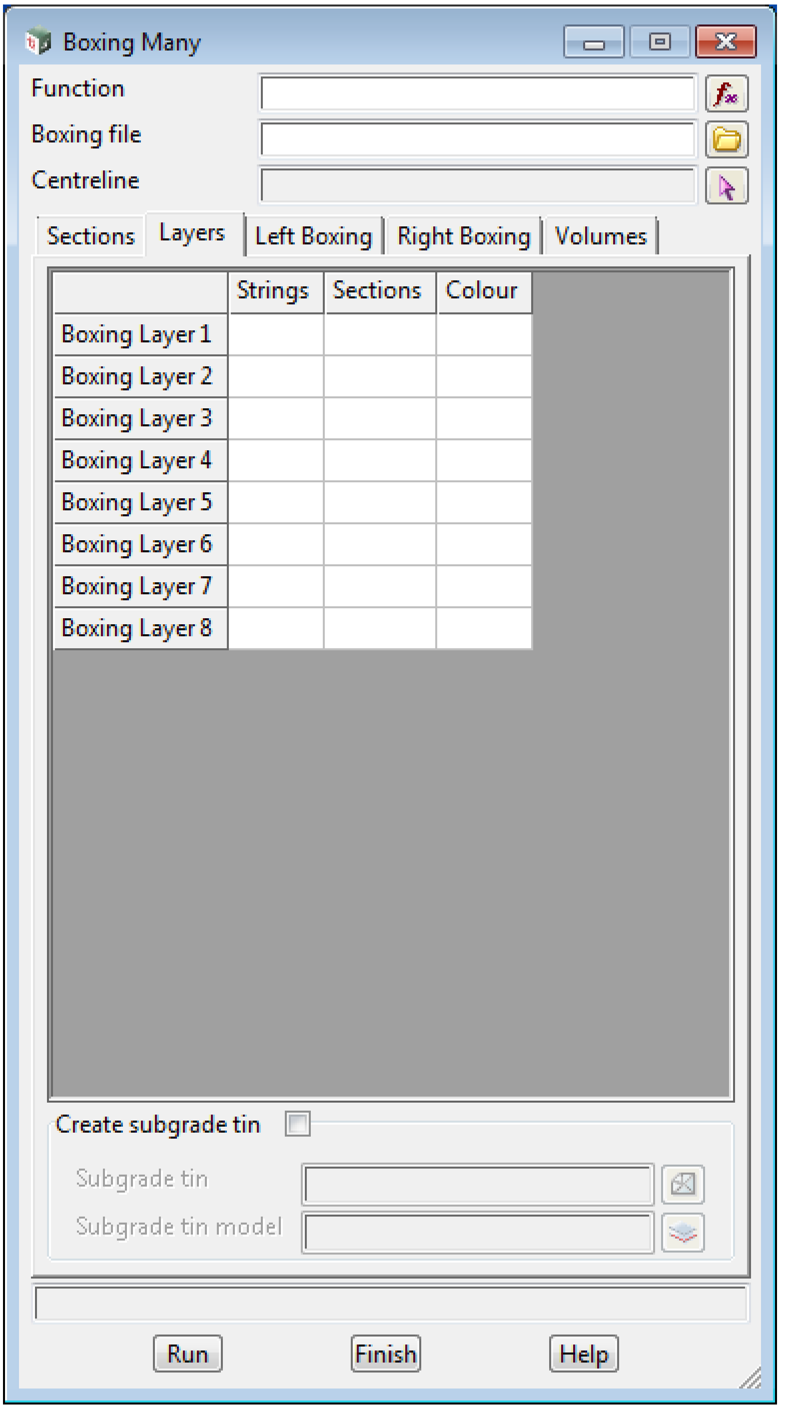
name of the tin of cross sections

Model for tin model box available models

model for the cross sections tin.

Layers tab

the layers tab consists of a grid for defining up to eight layers of boxing strings and sections. The last boxing layer is also referred to as the ***subgrade*** *layer.*



Boxing Layer 1-8:

if the model names are non blank, boxing strings and sections will be created for that layer.

For more information on boxing, see 21.6 What is Boxing?.

Strings input

*if non-blank, pre*post text to use with the function name to create the name of the model for the boxing strings for this layer.*

If blank, the boxing strings will not be stored for this layer.

Sections	input	available models

*if non-blank, pre*post text to use with the function name to create the name of the model for the boxing sections for this layer.*

If blank, the boxing sections will not be stored for this layer.

Colour	colour box	available colours

colour for the boxing sections for this layer

Create subgrade tin	tick box	

if ticked, then a tin of the subgrade strings and sections is created. Note - the subgrade is the last boxing layer.

Subgrade tin	tin box	available tins

name for the subgrade tin

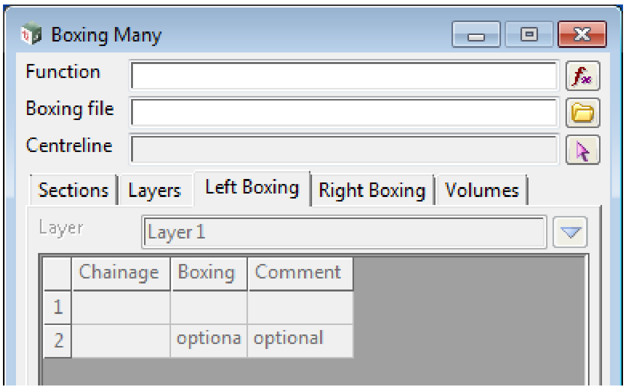
Subgrade tin model	model box	available models

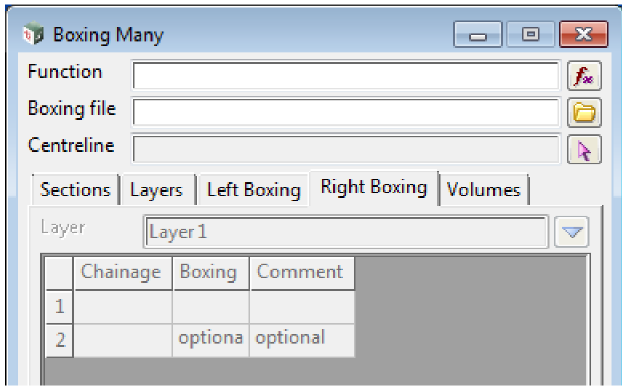
model for the subgrade tin

Left/Right Boxing tabs

tab for applying left boxing - gives the chainage range and boxing definitions to use for each of the eight layers of boxing

For more information on boxing, see 21.6 What is Boxing?.





Layer	choice box	Layer 1, Layer 2 ... Layer 8

as each layer is selected, a grid for defining the chainages and boxing definition that applies to that chainage are given in the grid. The boxing definition goes form the chainge on the row specifying the boxing file until the chainge on the new row.

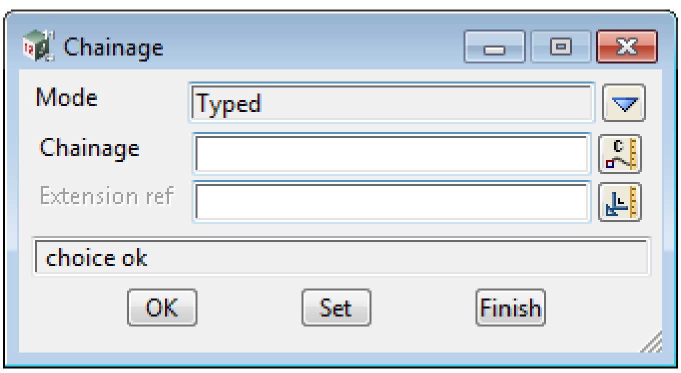
Layer grid:

For each selected layer:

Chainage smart chainge box

For the selected boxing layer, the chainage to apply the boxing definition to. The boxing file starts at this chainge and goes until the chainage on the next row.

*If you right click in the **Chainage** column in the **Left/Right Boxing** tabs, or select **Browse** if the Browse menu comes up) then a* **Chainage** *panel comes up which has the most of the standard MTF choices for **Smart Chainages**.*



The **Chainage Mode** *will determine what other fields are also on the panel (for example* **Extension ref** *for* **Mode** ***Typed****). For more information on Chainage Modes, see* 21.2.1 MTF Hinge Modifiers

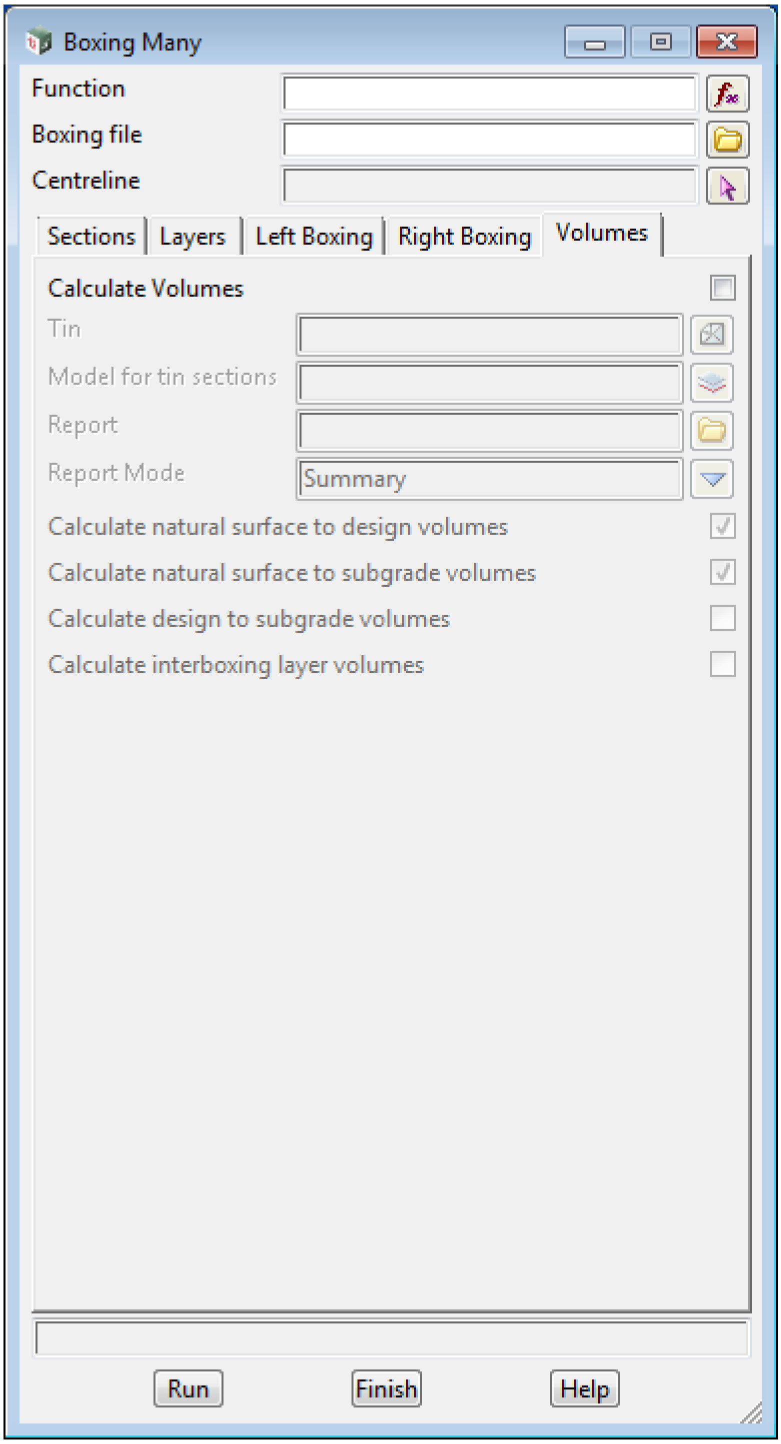
Boxing	boxing definition box	boxing definitions

*the boxing definition to use with the given chainage range. The pop-up lists all the boxing definitions in the **Boxing file***

Comment

record a comment

Volumes tab



Calculate volumes tick box

if ticked, then end area volumes are created

Tin tin box available tins

name of the natural surface tin - required if volumes to the natural surface are calculated

Model for tin sections model box available models

if non blank, model to put the natural surface sections into

Report file file box *.rpt

if non-blank, the name of the file to contain the volume report. If the file already exists, the report will be appended to the file.

If blank, no report is produced.

Report mode choice box Summary, Full

*if **Summary**, the report only contains the final volumes.*

*If **Full**, the report contains sections by section areas and volumes as well as the summary of volumes.*

Calculate natural surface to design volumes tick box tick

if ticked, end area volumes between the natural surface and the design strings are written to the report file

Calculate natural surface to subgrade volume tick box tick

if ticked, end area volumes between the natural surface to the subgrade (the last boxing layer) are written to the report file

Calculate design to subgrade volumes tick box tick

if ticked, end area volumes between the design strings and the subgrade (the last boxing layer) are written to the report file.

Calculate inter-boxing layer volumes tick box tick

if ticked, end area volumes between each of the boxing layers are written to the report file.

Run button

run the option

For more information on boxing, see 21.6 What is Boxing? and for more information on the various scenarios for generating boxing, see 21.6.2 Applying Boxing.

Go to the next section 20.5.5 Apply Boxing or return to 20.5 Boxing.

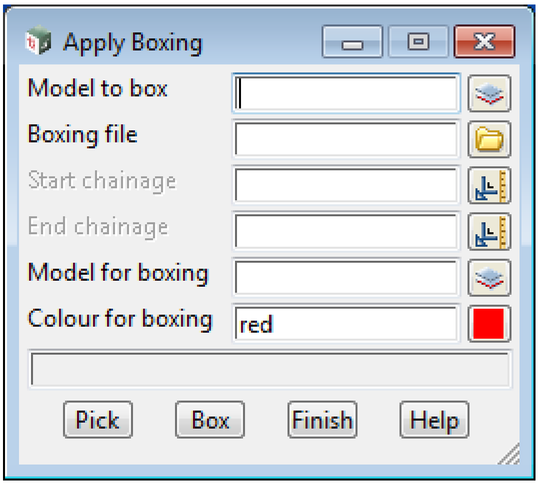
20.5.5 Apply Boxing

Position of option on menu: **Design =>Boxing =>Boxing**

Selecting **boxing** displays the **apply boxing** pane.

This panel is used to create boxing sections by applying the first **Boxing Definition** in a *Boxing* file, to a selected 4d super string or model of sections created as 4d super strings. Only the one *Boxing Definition* is used for all of the sections.

The **Apply Boxing** panel is



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model to box	model box		available models

model of sections (in 4d string format) to apply the boxing definition to.

Boxing file	file box		*.bf files

name of the file containing the boxing definition to be applied to the sections in the **model to box**.

Start/End chainage	input		

the start/end string chainage for applying boxing. If blank, then the **start/end chainage** *is taken to be the chainage at the beginning/end of the picked string.*

Model for boxing	model box		available models

model for the created boxing x-sections

Colour for boxing	colour box		available colours

colour for the boxing x-sections

Pick	button		

after picking the **Pick** *button, any selected strings will have the boxing applied to them.*

Box	button		

apply the first Boxing Definition given in the **Boxing file** *to the sections in the* **model to box** *that are between the given* **Start chainage** *and* **End chainage**.

For more information on boxing, see 21.6 What is Boxing? and for more information on the various scenarios for generating boxing, see 21.6.2 Applying Boxing.

Go to the next section 20.5.6 Apply Boxing Many or return to 20.5 Boxing.

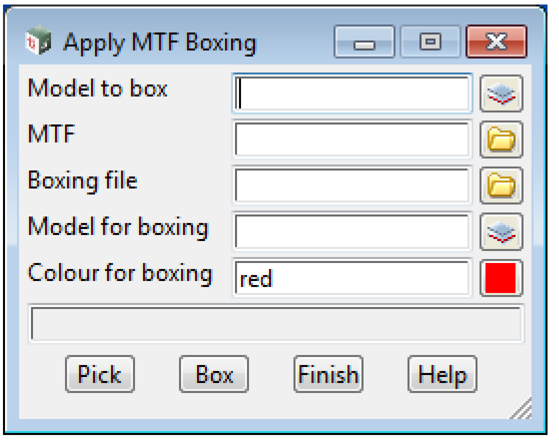
20.5.6 Apply Boxing Many

Position of option on menu: **Design =>Boxing =>Boxing many**

This panel can be used to create boxing sections by applying the **Boxing Definitions** as specified in an MTF file. to a selected 4d super string or to a model of sections.

If an individual 4d super string is picked, the option uses the Boxing defined for the chainage where the section 4d string was created.

Selecting **Boxing Many** displays the **Apply Many Boxing** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model to box	model box		available models

model of sections (in 4d string format) to apply the boxing definition to.

MTF file	file box		*.mtf files

name of the MTF file which contains the application of the Boxing Definitions to the sections given in the **Model to box** *panel field.*

If **Boxing file** *is* ***blank****, use the Boxing file given in the MTF file.*

If **Boxing file** *is* ***non-blank****, then all the Boxing Definitions referred to in the* **MTF file** *are taken from the* **Boxing file.**

Boxing file	file box		*.bf files

If ***non-blank****, then all the Boxing Definitions referred to in the* **MTF file** *are taken from the* **Boxing file***. That is, the Boxing file given in the MTF file is NOT used for the Boxing Definitions.*

If ***blank****, use the Boxing file given in the MTF file.*

Model for boxing	model box		available models

model for the created boxing x-sections

Colour for boxing	colour box		available colours

colour for the generated boxing x-sections

Pick	button		

after picking the **Pick** *button, any selected strings will have the boxing applied to them.*

Box	button		

apply the boxing as given by the **MTF file***, using the boxing definitions is the* **Boxing file***, to the 4d sections in the* **Model for boxing field***. The new boxing sections are added to the* **Model for boxing**.

For more information on boxing, see 21.6 What is Boxing? and for more information on the various scenarios for generating boxing, see 21.6.2 Applying Boxing.

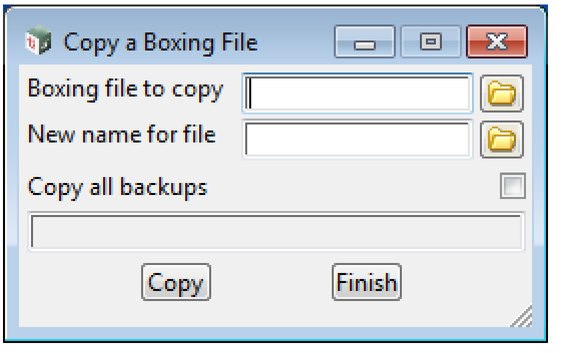
Go to the next section 20.5.7 Copy Boxing or return to 20.5 Boxing.

20.5.7 Copy Boxing

Position of option on menu: **Design =>Boxing =>Copy**

A copy of an existing boxing file can be made using the **Copy** option. This is often useful when a new boxing file that is similar to an existing boxing file is needed. The existing boxing file can be copied, and the copy then edited and modified.

On selecting **Copy** the **Copy a Boxing File** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Boxing file to copy	boxing file box		*.bf files

name of the boxing file to be copied.

New name for file	boxing file box		*.bf files

name of the new boxing file.

Copy all backups tick box

if ticked, copy all the backup files of the boxing file as well.

Copy button

after selecting this button, the boxing file given in the **Boxing file to copy** *field will be copied and the copy given the name in the* **New name for file** *field.*

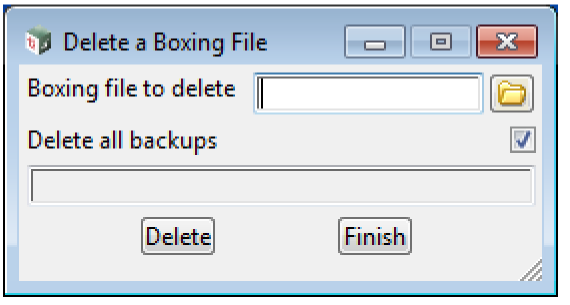
Go to the next section 20.5.8 Delete a Boxing File or return to 20.5 Boxing.

20.5.8 Delete a Boxing File

Position of option on menu: **Design =>Boxing =>Delete**

The **Delete a template** deletes boxing files from the project.

Selecting **Delete a**, displays the **Delete Template** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Boxing file to delete	boxing file box		*.bf files

name of the boxing file to delete.

Delete all backups	tick box		

if ticked, the boxing file and all its backups files will be deleted.

Delete	button		

after selecting this button, the boxing file given in the Boxing file to delete field will be deleted.

Go to the next section 20.5.9 Create Definitions Text File or return to 20.5 Boxing.

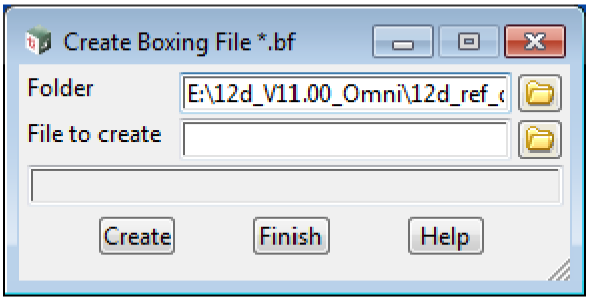
20.5.9 Create Definitions Text File

Position of option on menu: **Design =>Boxing =>Create definitions file**

The **Boxing=>Create definition file** option is used to create a new **Boxing** file (*.bf) with the text editor pointed to by the EDIT_4D environment variable.

The created text file will already have a section header set up for a boxing called “1”(key word and opening and closing brackets).

Selecting **Create definitions file** brings up the **create boxing file *.bf** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Folder	folder box	current folder	

name of the folder for the .bf file.

File to create	file box		*.bf files

name of the file to create.

Create	button		

create a Boxing file given by the **folder** *and* **file to create** *panel fields. If the file given in the* **file to edit** *field does not exist, then a new file is created which already has a section header set up.*

For more information on boxing, see 21.6 What is Boxing? and for more information on the various scenarios for generating boxing, see 21.6.2 Applying Boxing.

The text format for the *Boxing* file is described in the section 21.10.2 Text Format of the Boxing File.

Go to the next section 20.5.10 Edit Definitions Text File or return to 20.5 Boxing.

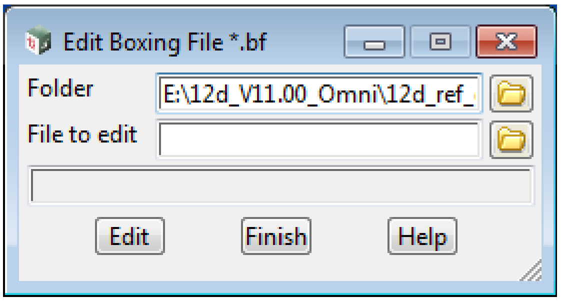
20.5.10 Edit Definitions Text File

Position of option on menu: **Design =>Boxing =>Edit definitions file**

The **Boxing=>Edit definitions file** option is used to edit boxing files (*.bf) with the text editor pointed to by the EDIT_4D environment variable.

The **Boxing=>Edit definitions file** option has two modes of operation - selecting the **Boxing=>Edit definitions file** itself, or by activating the **Boxing=>Edit definitions file** option's walk-right menu, **folder *.bf**.

Selecting **Boxing=>Edit definitions file** itself brings up the **edit boxing file *.bf** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Folder	input	current folder	

name of the folder for the .bf file.

File to edit	input		*.bf files

name of the file, in **folder***, to edit.*

Edit	button		

edit the file given by the **folder** *and* **file to edit** *panel fields. If the file given in the* **file to edit** *field does not exist, then a new file is created which already has a section header set up.*

The **Boxing=>Edit definitions file** walk-right menu provides a list all the boxing files (files ending in .bf) in the current folder. When a file is selected from the list, it is automatically loaded into the text editor.

The format for the boxing file is described in the 21.10.2 Text Format of the Boxing File section.

Return to 20.5 Boxing.

20.6 Estate Lots

Position of menu: **Design =>Estate/Lots**

This module is currently under development.

The Estate Lots module is for creating house lots for an Estate (Subdivision).

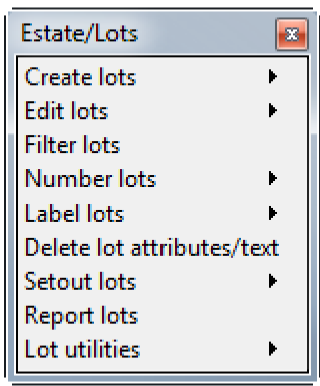
There are options to create and edit lots by a variety of methods. Once the lots are created they can be numbered, given a lot type (see 20.6.1 Lot types) and labelled.

The labelling includes labelling the side of lots with bearing and distances, lots with areas and lot numbers.

Reports of lot areas and types can be created and lots coloured according to areas.

Finally options exist to create point numbers and reports for setting out the lots.

The **Estate/Lots** walk-right menu is:



For the options see

Create lots	20.6.2 Create Lots
Edit lots	20.6.3 Edit Lots
Filter lots	20.6.4 Filter Lots
Number lots	20.6.5 Number Lots
Label lots	20.6.6 Label Lots
Delete lot attributes/text	20.6.7 Delete Lot Attributes and Text
Setout lots	20.6.8 Setout Lots
Report lots	20.6.9 Report Lots
Lot utilities	20.6.10 Lot Utilities

20.6.1 Lot types

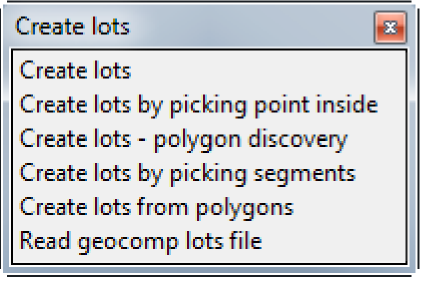
Lots can have a ***type*** which is used in reports. The lot types are defined in a text file called *lottypes.4d* which is searched for in the standard library areas (see 39.3 Library, User Library, Customer Library).

For example of lottypes.4d is

park
road
other
surround
McDonalds

20.6.2 Create Lots

Position of menu: **Design =>Estate/Lots => Create lots**



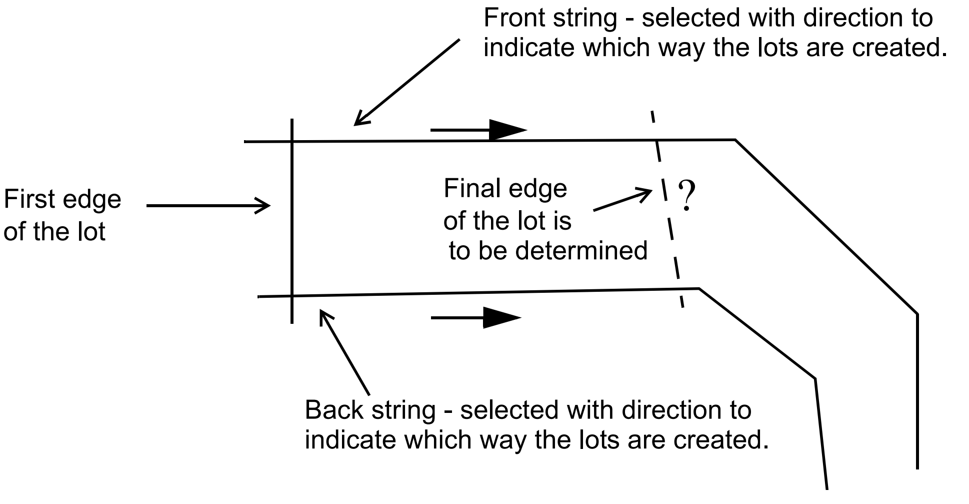
For the options see

Create lots	20.6.2.1 Create Lot
Create lots by picking point inside	20.6.2.2 Create Lot from Picking
Create lots- polygon discovery	28.10.8 Polygon Discovery
Create lots by picking segments	20.6.2.3 Create Lot from Picking Segments
Create lots from polygons	20.6.2.4 Create Lot from Polygon
Read geocomp lots file	20.6.2.5 Read Geocomp Lots File

20.6.2.1 Create Lot

Position of option on menu: **Design =>Estate/Lots => Create lots =>Create lot**

This option creates lots of a user given area using the front and back strings for a number of lots and an initial ***straight line edge*** of a lot. The final edge of the lot then needs to be determined.



The methods of creating the final edge of lot are:

(a) parallel the existing side - *parallel edge.* See Parallel Edge

(b) having the same bearing as a selected line - *parallel pick.* See Parallel Pick

(c) having a given bearing - *parallel bear*. See Parallel Bearing

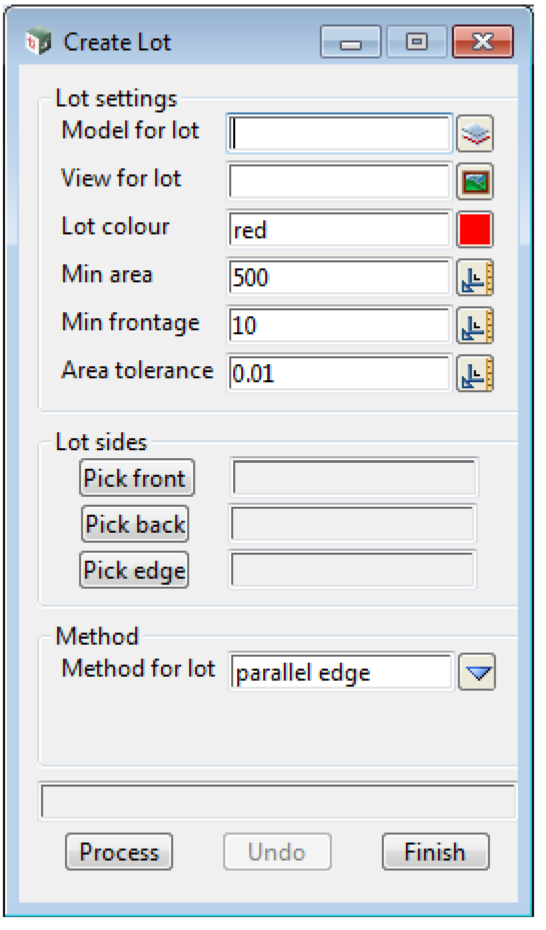
(d) perpendicular to either the front string or the back string - *perpendicular.* See Perpendicular and Perpendicular Pick

(e) perpendicular to either the front string or the back string but starting from a user selected point - *perpen start pt.* See Perpendicular and Perpendicular PickSee Perpendicular and Perpendicular Pick

(f) pivoting about a selected point - *pivot pick.* See Pivot Pick and Pivot Frontage

(g) pivoting about the point at minimum frontage - *pivot frontage.* See Pivot Pick and Pivot Frontage

(h) closing a selected string - *close string.* See Close String



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model for lot	model box		available models

model for the created lot.

View for lot	view box		available views

view to add model of created lot to.

Lot colour	colour box		available colours

colour for the created lot.

Min area double box

required area of the created lot.

Min frontage input box

required minimum frontage of the created lot.

Area tolerance double box

lot to be created can be within this tolerance of the minimum area.required area of the created lot.

Pick front string select

pick the string to form the front of the created lot.

Pick back string select

pick the string to form the back of the created lot.

Pick edge	string select		

*pick the **straight** line to form the edge of the created lot.*

The front, back and edge form three sides of the lot to be created.

Method	choice box	parallel edge	parallel edge, parallel pick parallel bear, perpendicular, perpen start pt, pivot pick, pivot frontage, close string

pick the string to form the back of the created lot.

Process	button		

use the selected front, back and edged and then create the final side of the lot by the selected method.

Continue to the next section 20.6.2.1.1 Creating Lots from a Front and Back String and an Edge for a full description of each method of creating the lots.

20.6.2.1.1 Creating Lots from a Front and Back String and an Edge

Parallel Edge

This method creates a lot of a user specified area using user selected front and back strings and a user selected initial edge. The final edge then needs to be determined to define the lot.

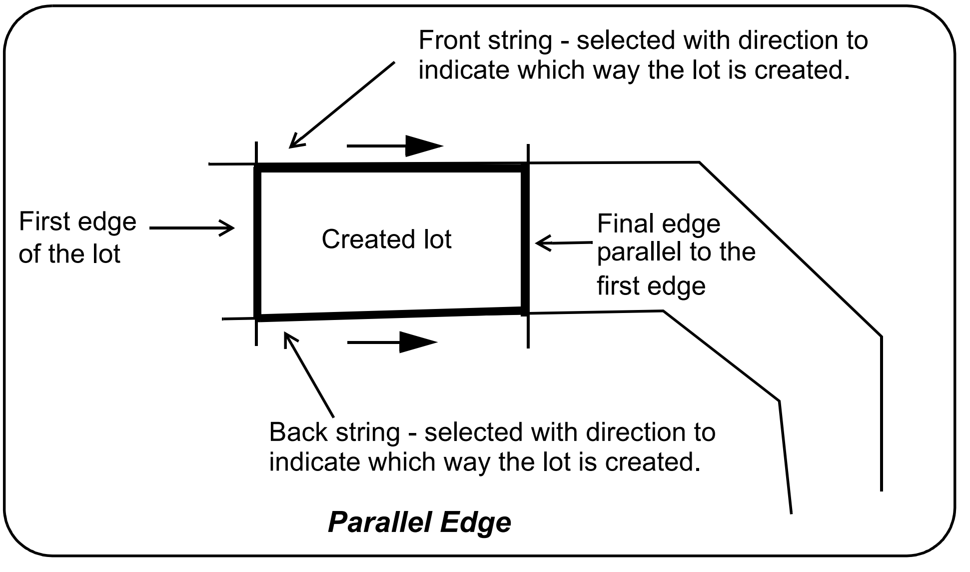
The front and back strings ***must*** be selected ***with direction*** to indicate which direction to move along them to form the lot. The front and back strings can not be closed strings and must be different strings. The front and back strings can have more than one segment and can have straight or arc segments.

The initial edge must cut or be very close to cutting the front and back strings. The ***initial edge*** must be ***one straight line segment***. It can not be an arc.

For *Parallel Edge*, the ***final edge*** is created parallel to the initial edge and is positioned along the front and back strings in their selected direction so that the lot has the user given area (to within the area tolerance).

The lot is only created if the new edge is at least the minimum frontage distance along the front string.

When the lot is created, the final edge is automatically set as the initial edge for the calculation of the next lot.



Parallel Bearing

This method creates a lot of a user specified area using user selected front and back strings and a user selected initial edge. The final edge then needs to be determined to define the lot.

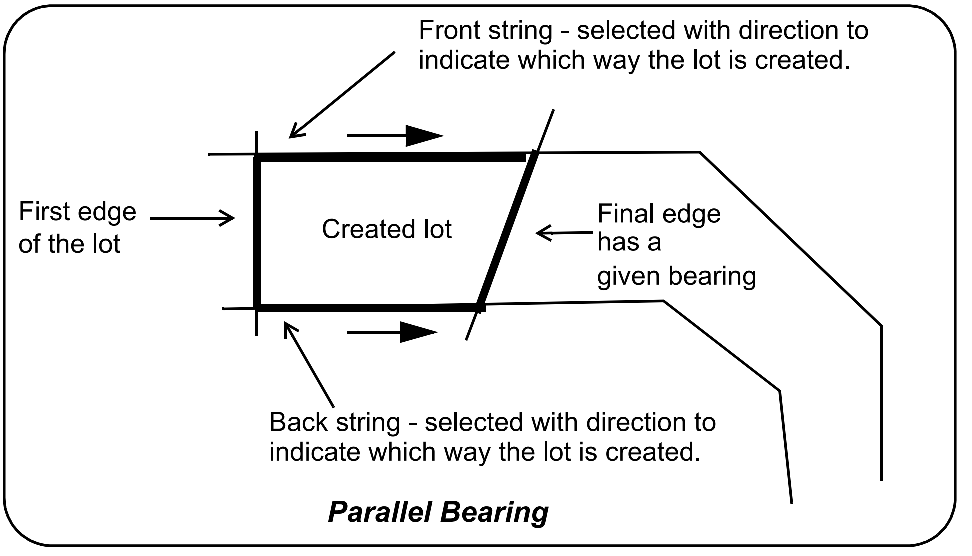
The front and back strings ***must*** be selected ***with direction*** to indicate which direction to move along them to form the lot. The front and back strings can not be closed strings and must be different strings. The front and back strings can have more than one segment and can have straight or arc segments.

The initial edge must cut or be very close to cutting the front and back strings. The ***initial edge*** must be ***one straight line segment***. It can not be an arc.

For *Parallel Bear*, the ***final edge*** is created with a user specified bearing and is positioned along the front and back strings in their selected direction so that the lot has the user given area (to within the area tolerance).

The lot is only created if the new edge is at least the minimum frontage distance along the front string.

When the lot is created, the final edge is automatically set as the initial edge for the calculation of the next lot.



Parallel Pick

This method creates a lot of a user specified area using user selected front and back strings and a user selected initial edge. The final edge then needs to be determined to define the lot.

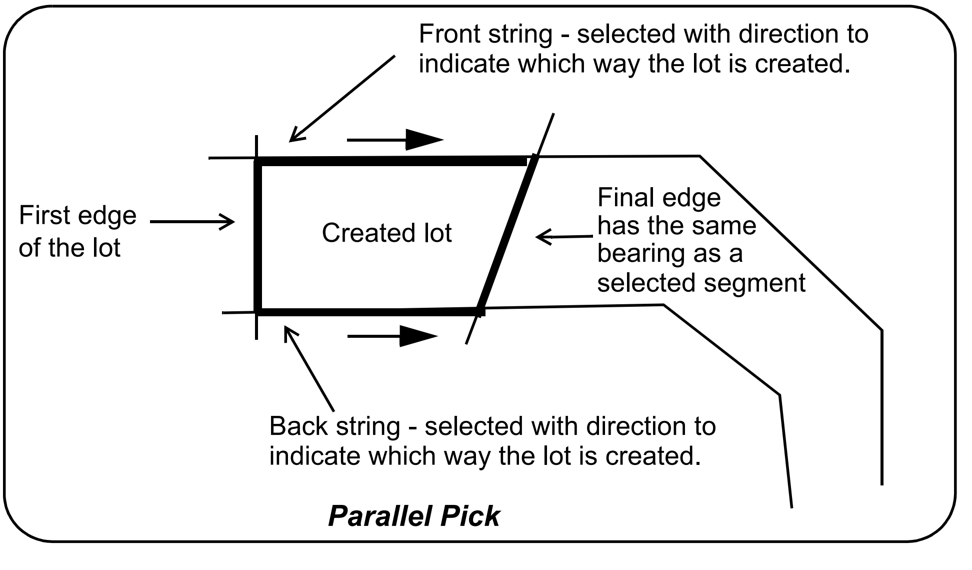
The front and back strings ***must*** be selected ***with direction*** to indicate which direction to move along them to form the lot. The front and back strings can not be closed strings and must be different strings. The front and back strings can have more than one segment and can have straight or arc segments.

The initial edge must cut or be very close to cutting the front and back strings. The ***initial edge*** must be ***one straight line segment***. It can not be an arc.

For *Parallel Pick*, the ***final edge*** is created with the bearing of a user selected segment and is positioned along the front and back strings in their selected direction so that the lot has the user given area (to within the area tolerance).

The lot is only created if the new edge is at least the minimum frontage distance along the front string.

When the lot is created, the final edge is automatically set as the initial edge for the calculation of the next lot.



Perpendicular and Perpendicular Pick

This method creates a lot of a user specified area using user selected front and back strings and a user selected initial edge. The final edge then needs to be determined to define the lot.

The front and back strings ***must*** be selected ***with direction*** to indicate which direction to move along them to form the lot. The front and back strings can not be closed strings and must be different strings. The front and back strings can have more than one segment and can have straight or arc segments.

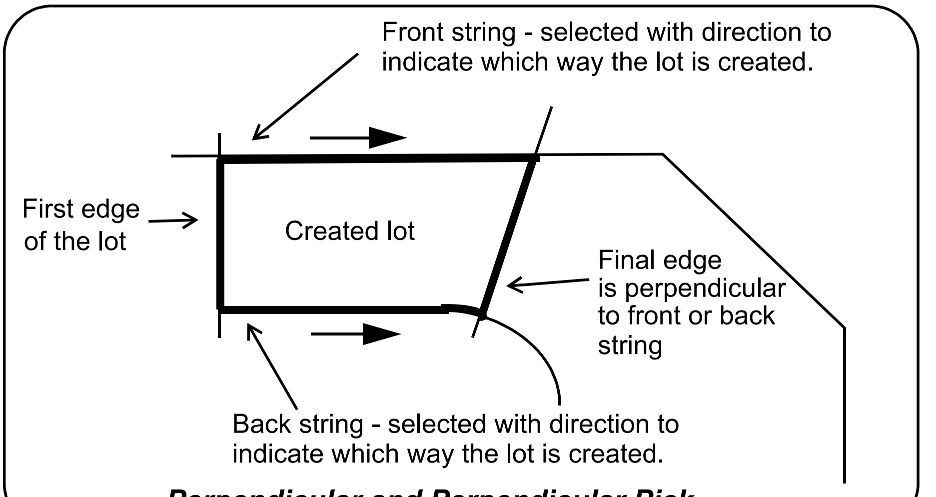
The initial edge must cut or be very close to cutting the front and back strings. The ***initial edge*** must be ***one straight line segment***. It can not be an arc.

For *Perpendicular*, the ***final edge*** is constrained to be perpendicular to either the front or the back string and is positioned along the front and back strings in their selected direction so that the lot has the user given area (to within the area tolerance).

For *Perpendicular Pick*, a start point is selected on the front/back string and the ***final edge*** is created past the start point. This option only needs to be used when *Perpendicular* has problems finding a solution due to sharp changes of bearing along the front/back string.

The lot is only created if the new edge is at least the minimum frontage distance along the front string.

When the lot is created, the final edge is automatically set as the initial edge for the calculation of the next lot.



Perpendicular and Perpendicular Pick

Pivot Pick and Pivot Frontage

This method creates a lot of a user specified area using user selected front and back strings and a user selected initial edge. The final edge then needs to be determined to define the lot.

The front and back strings ***must*** be selected ***with direction*** to indicate which direction to move along them to form the lot. The front and back strings can not be closed strings and must be different strings. The front and back strings can have more than one segment and can have straight or arc segments.

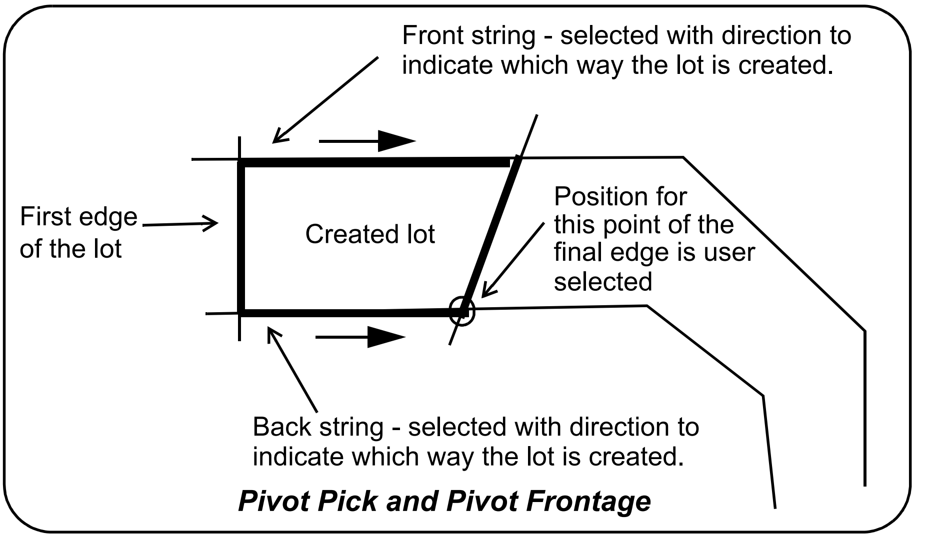
The initial edge must cut or be very close to cutting the front and back strings. The ***initial edge*** must be ***one straight line segment***. It can not be an arc.

For *Pivot Pick*, a position is selected on either the front or back string to be one point of the ***final edge***. The other point of the final edge is determined so that the lot has the required given area (to within the area tolerance).

For *Pivot Frontage*, the pivot point is taken to be the position on the front string that is the minimum frontage distance along the front string from the initial edge.

The lot is only created if the new edge is at least the minimum frontage distance along the front string.

When the lot is created, the final edge is automatically set as the initial edge for the calculation of the next lot.



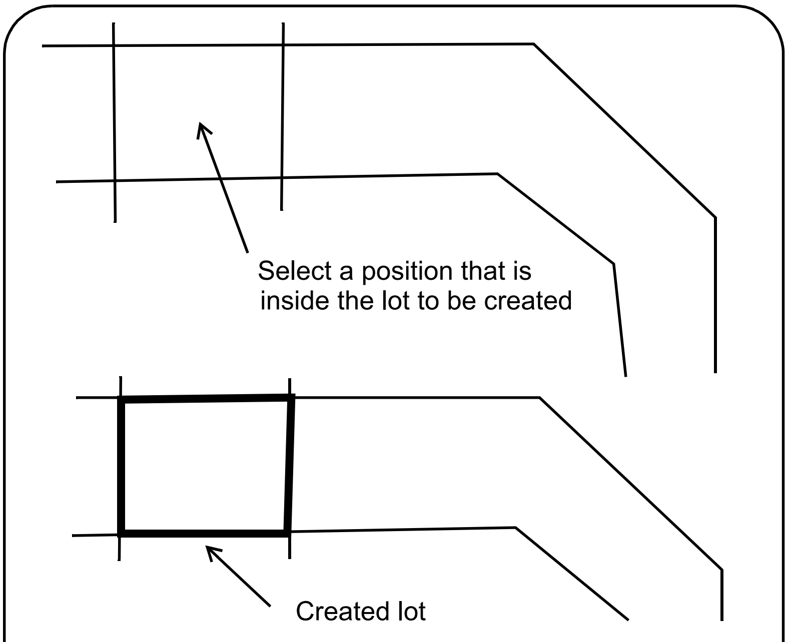
Close String

This method simply closes the string that is selected by the user.

20.6.2.2 Create Lot from Picking

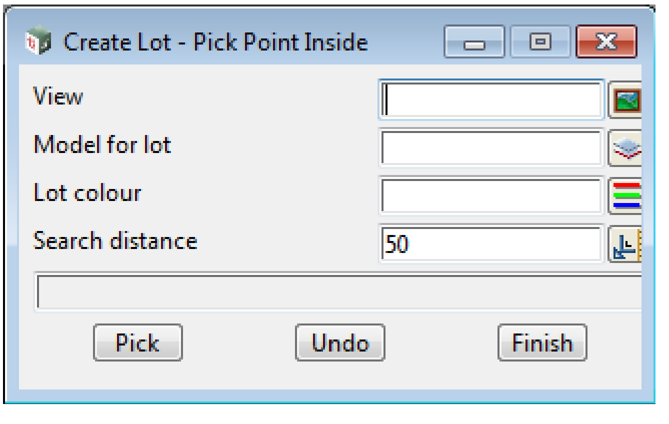
Position of option on menu: **Design =>Estate/Lots => Create lots =>Create lots by picking point inside**

This option creates a lot by a picking inside a collection of strings and the lot is created from the closest strings to the picked position. The picked position must be selected so that all sides of the lot can be "seen" from the picked position. That is, a straight line can be drawn from the picked position to the lot side without cutting any other segment.



Creating a Lot by Picking Inside the Lot

On selecting the **Create lot by picking point inside** option, the **Create Lot - Pick Point Inside** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View	view box		available views

view to add model of created lot to.

Field Description	Type	Defaults	Pop-Up
Model for lot	model box		available models

model for the created lot.

Field Description	Type	Defaults	Pop-Up
Lot colour	colour box		available colours

colour for the created lot.

Search distance double box 20

maximum distance to search from the selected point for sides of the lot

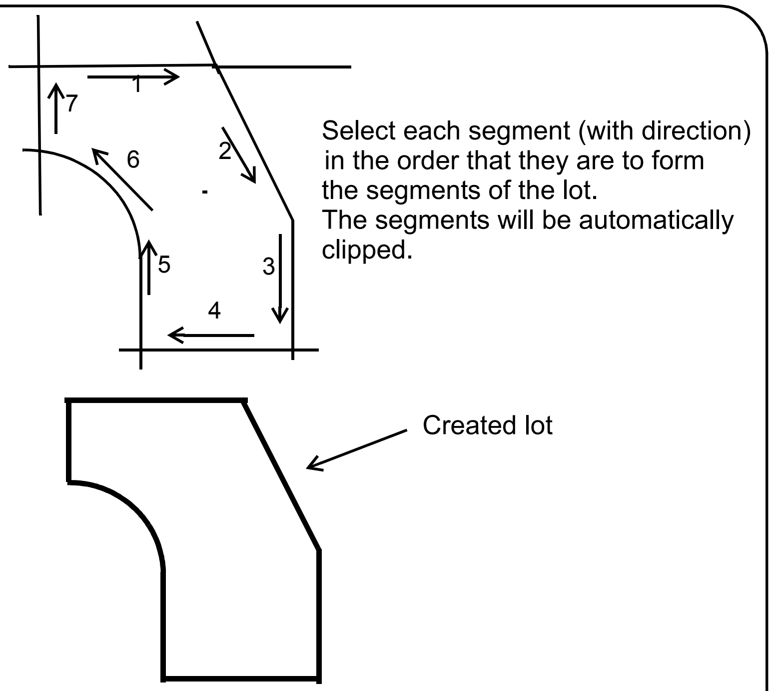
Pick string select

pick the position to try and form a lot around.

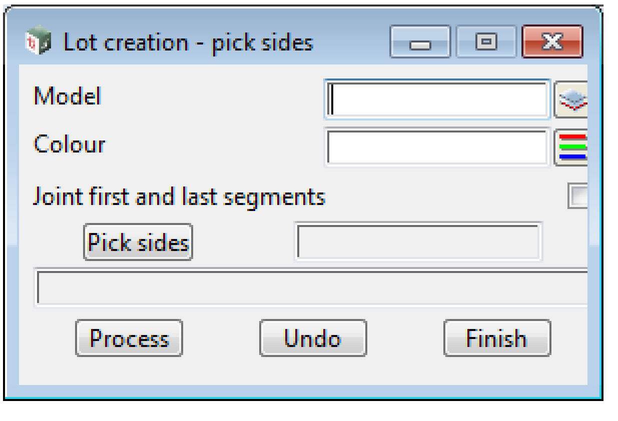
20.6.2.3 Create Lot from Picking Segments

Position of option on menu: **Design =>Estate/Lots => Create lots =>by picking segments**

This option creates a lot by a picking each segment in its order (and with direction) around the lot. Segments will be automatically extended or clipped to form the lot.



Creating a Lot by Picking the Lot Segments



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model for lot	model box		available models

model for the created lot.

Lot colour	colour box		available colours

colour for the created lot.

Join first and last segment	tick box		

if ticked, join the end of the last selected segment to the start of the first selected segment to form the *final side of the lot.*

Pick sides string select

pick, with direction, the segments to be joined together to form the sides of the lot.

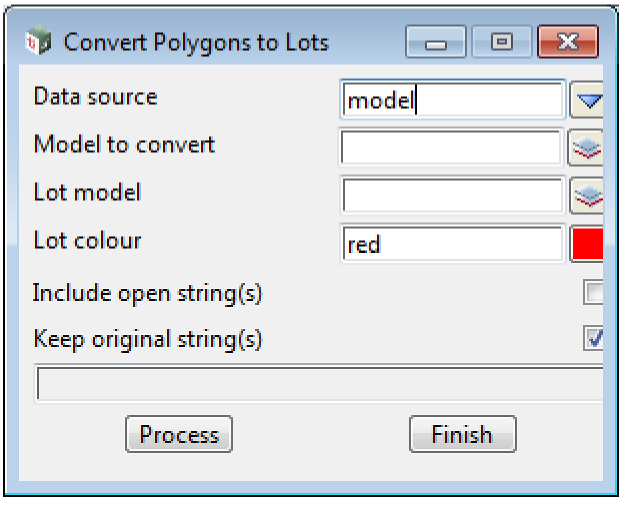
Process button

create the lot from the selected segments.

20.6.2.4 Create Lot from Polygon

Position of option on menu: **Design =>Estate/Lots =>Create lots =>Lots from polygons**

This option converts strings to a lot. The strings can be open.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type			
Data source			
Model for lot	model box		available models
Lot model	model box		available models
Lot colour	colour box		available colours
Include open strings	tick box		
Process	button		

Data source type

data source type.

Data source

data source for strings to convert to lots.

Model for lot model box available models

model for the created lot.

Lot model model box available models

model for the created lots.

Lot colour colour box available colours

colour for the created lots.

Include open strings tick box

if ticked, open strings are used to from lots by making then closed strings. If not ticked, open strings are ignored.

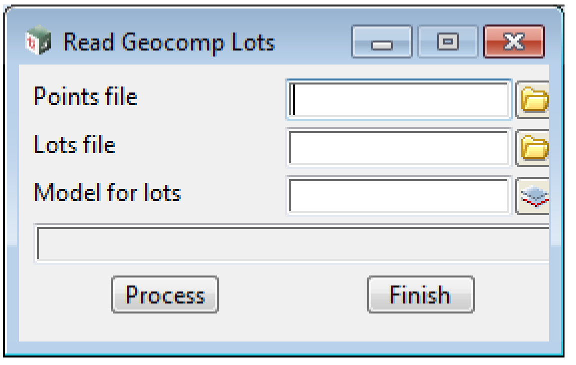
Process button

create lots from the selected strings.

20.6.2.5 Read Geocomp Lots File

Position of option on menu: **Design =>Estate/Lots => Create lots =>Read geocomp lots file**

This option reads a Geocomp lots file and creates ***12d Model*** lots.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Points file			*.pts files

name of the Geocomp points file.

Lots file			*.pts files

name of the Geocomp lots file - is uses points from the given Geocomp points file.

Model for lot	model box		available models

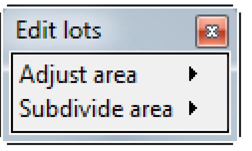
model for the created lot.

Process	button		

read in the given Geocomp points and lots file and created ***12d Model*** *lots.*

20.6.3 Edit Lots

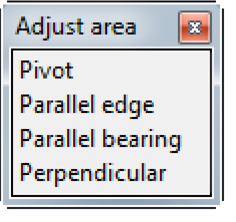
Position of menu: **Design =>Estate/Lots =>Edit lots**



For the option *Adjust area*, go to 20.6.3.1 Adjust Area
Subdivide area 20.6.3.2 Subdivide Area

20.6.3.1 Adjust Area

Position of menu: **Design =>Estate/Lots =>Edit lots =>Adjust area**



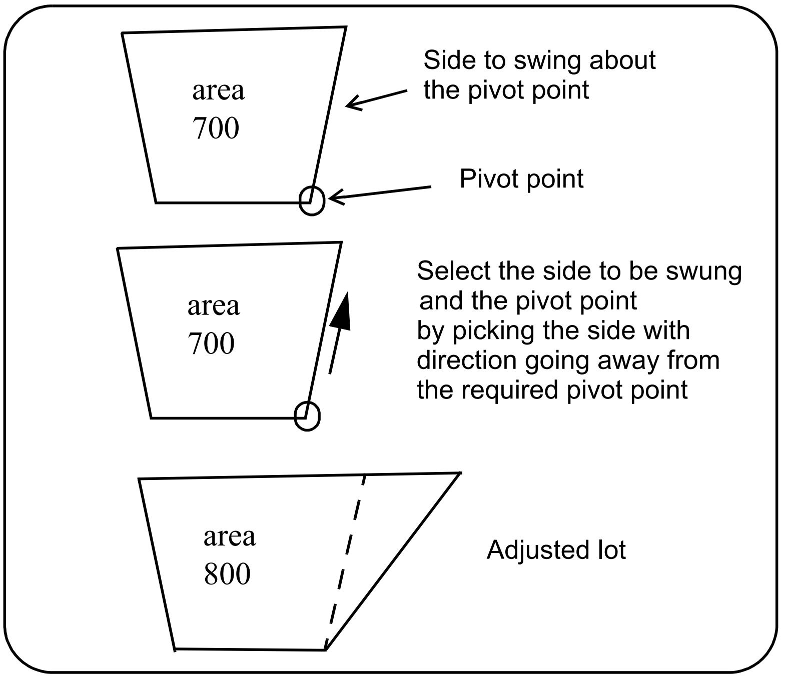
For the options see

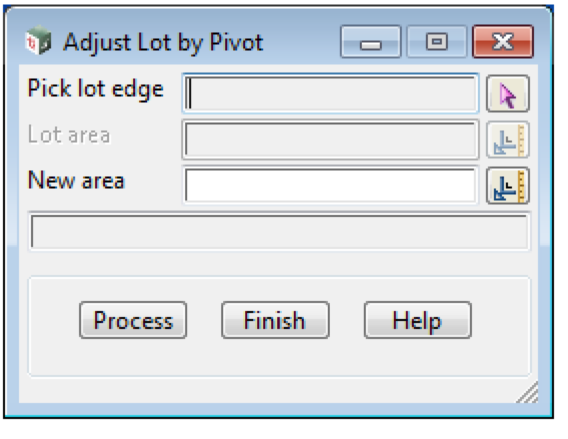
Pivot	20.6.3.1.1 Adjust Lot by Pivot
Parallel edge	20.6.3.1.2 Adjust Lot by Parallel Edge
Parallel bearing	20.6.3.1.3 Adjust Lot by Parallel Bearing
Perpendicular	20.6.3.1.4 Adjust Lot by Perpendicular

20.6.3.1.1 Adjust Lot by Pivot

Position of option on menu: **Design =>Estate/Lots =>Edit lots =>Adjust area =>Pivot**

This option adjusts the size of a lot by pivoting one side about one of its end points until the lot area has a new given area.





The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick lot edge	string select		

pick the lot to adjust by picking a side of the lot with direction. The start of the selected side will be the pivot point.

Lot area	output box		

area of the selected lot.

New area	input box		

required area of the adjusted lot. The area can be larger or smaller than the existing lot area.

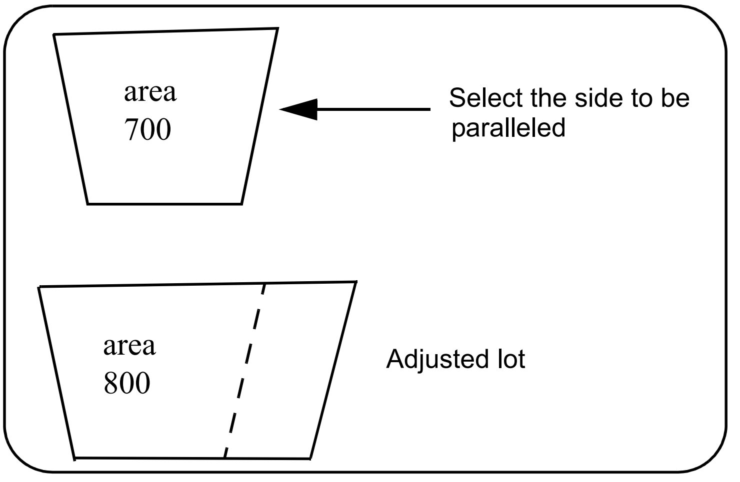
Process	button		

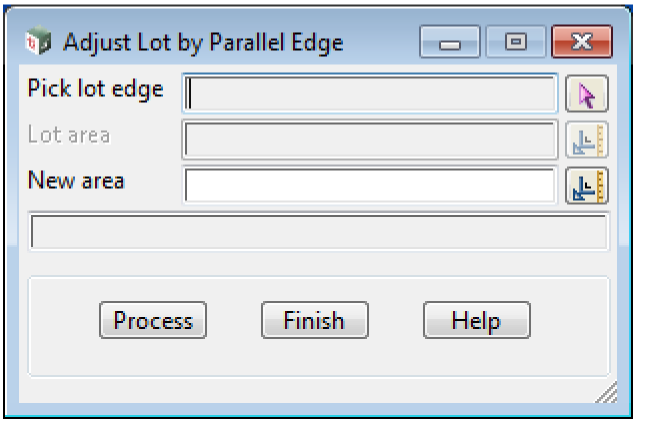
adjust the lot by pivoting the side.

20.6.3.1.2 Adjust Lot by Parallel Edge

Position of option on menu: **Design =>Estate/Lots =>Edit lots =>Adjust area =>Parallel edge**

This option adjusts the size of a lot by moving a side parallel to itself until the lot area has a new given area.





The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Pick lot edge string select

pick the lot to adjust by picking the edge of the lot that is to be paralleled.

Lot area output box

area of the selected lot.

New area input box

required area of the adjusted lot. The area can be larger or smaller than the existing lot area.

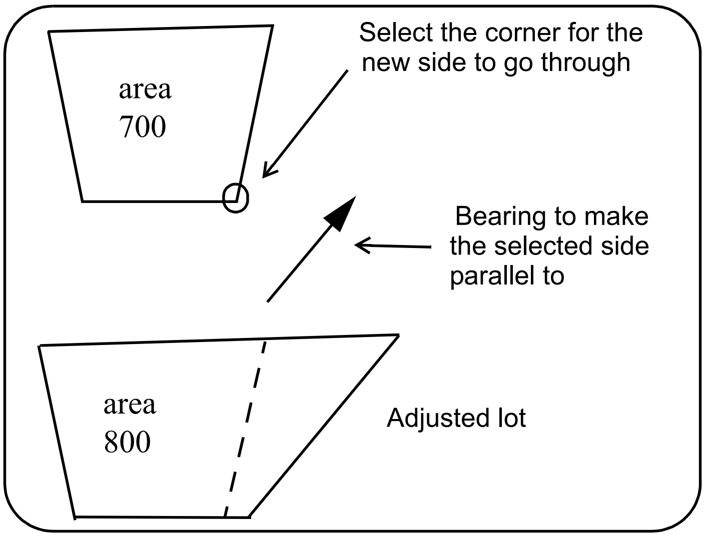
Process button

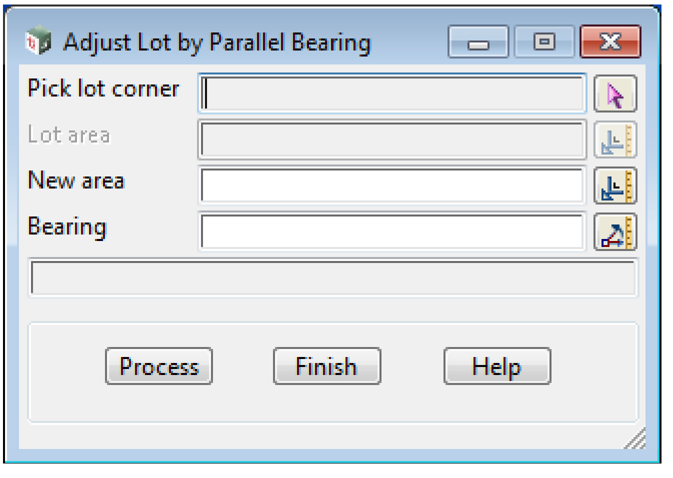
adjust the lot by paralleling the selected edge.

20.6.3.1.3 Adjust Lot by Parallel Bearing

Position of option on menu: **Design =>Estate/Lots =>Edit lots =>Adjust area =>Parallel bearing**

This option adjusts the size of a lot by moving a side parallel to a given bearing until the lot area has a new given area.





The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick lot corner	string select		

pick the lot to adjust by picking the corner of the lot that the side goes through and has the given bearing.

Lot area	output box		

area of the selected lot.

New area	input box		

required area of the adjusted lot. The area can be larger or smaller than the existing lot area.

Bearing	bearing box		

the bearing used for the selected edge.

Process	button		

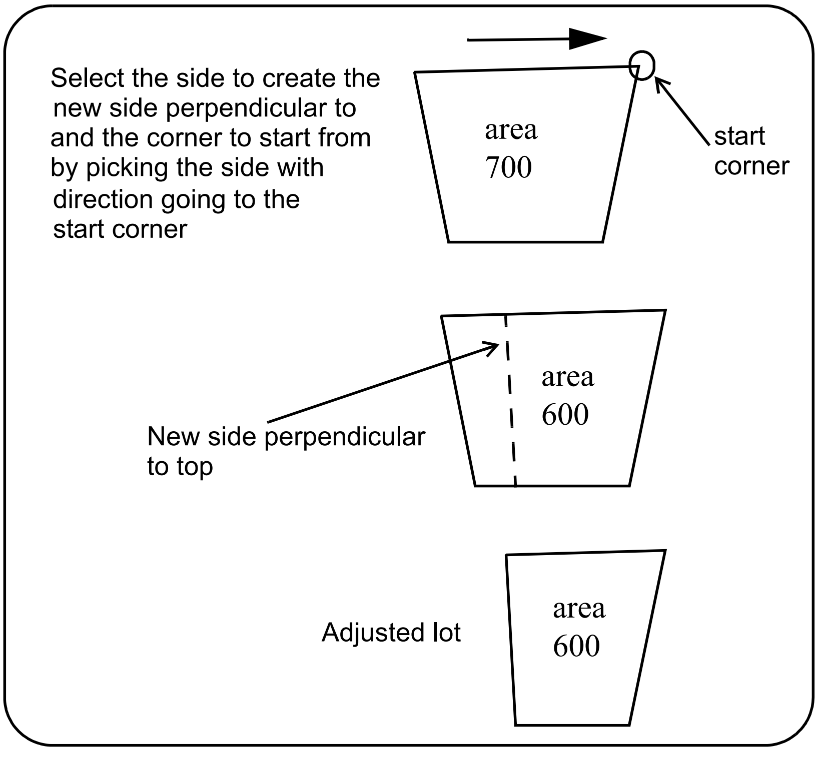
adjust the lot by paralleling the selected edge with the given bearing.

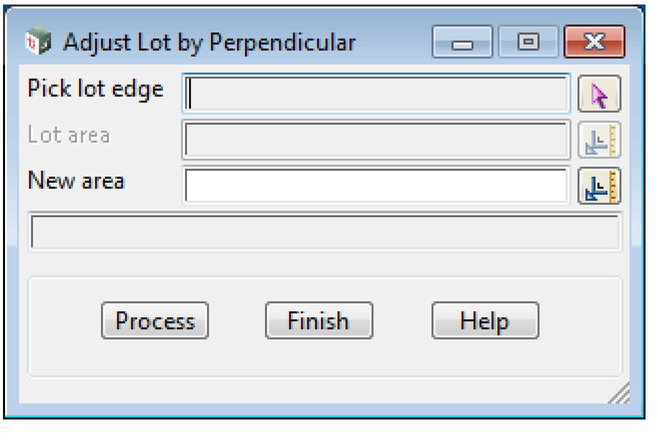
20.6.3.1.4 Adjust Lot by Perpendicular

Position of option on menu: **Design =>Estate/Lots =>Edit lots =>Adjust area =>Perpendicular**

This option adjusts the size of a lot by creating a new side perpendicular to a selected side and then moving the new side until the created lot area has a given area.

Note -the new area must be less than the original area.





The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick lot edge	string select		

pick the side of the lot that a new side will created perpendicular to

Lot area	output box		

area of the selected lot.

New area	input box		

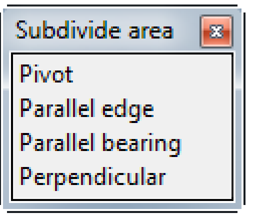
required area of the adjusted lot. The area must be smaller than the existing lot area.

Process button

adjust the lot by moving the side whilst keeping it perpendicular to another side.

20.6.3.2 Subdivide Area

Position of menu: **Design =>Estate/Lots =>Edit lots =>Subdivide area**



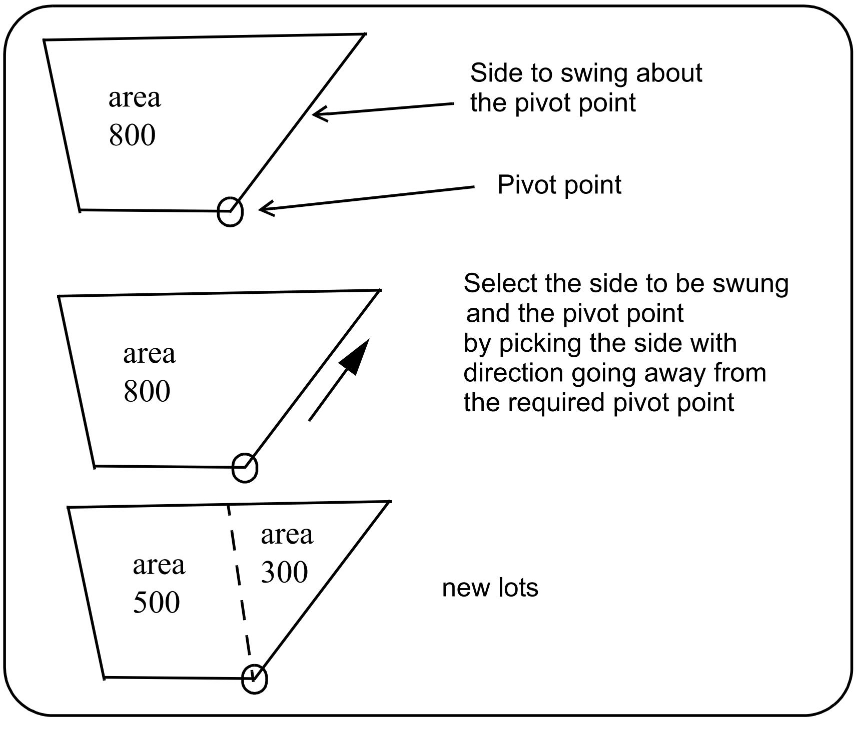
For the options see

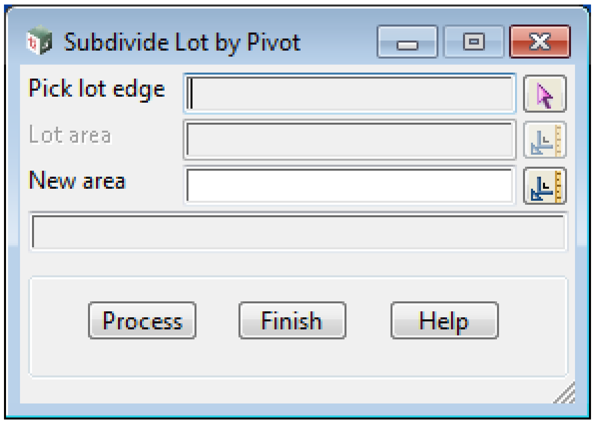
Pivot 20.6.3.2.1 Subdivide Lot by Pivot
Parallel edge 20.6.3.2.2 Subdivide Lot by Parallel Edge
Parallel bearing 20.6.3.2.3 Subdivide Lot by Parallel Bearing
Perpendicular 20.6.3.2.4 Subdivide Lot by Perpendicular

20.6.3.2.1 Subdivide Lot by Pivot

Position of option on menu: **Design =>Estate/Lots =>Edit lots =>Subdivide area =>Pivot**

This option reduces the size of a lot by pivoting one side about one of its end points until the lot area has a new smaller area. A second lot is created from the remainder of the original lot.





The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick lot	string select		

pick the lot to adjust by picking a side of the lot with direction. The start of the selected side will be the pivot point.

Lot area	output box		

area of the selected lot.

New area	input box		

required area of the adjusted lot. The area must be smaller than the existing lot area.

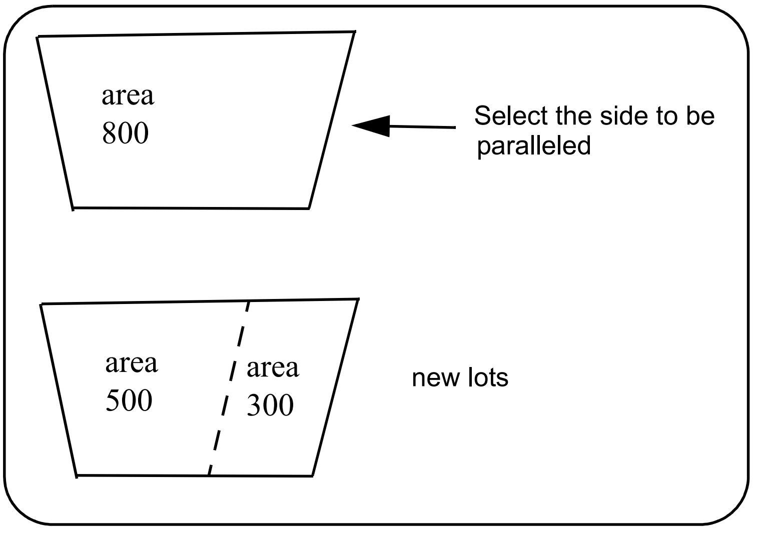
Process	button		

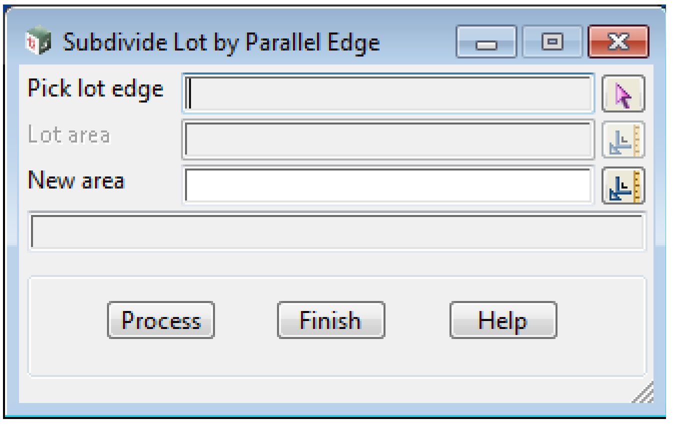
adjust the lot by pivoting the side. A second lot is created from the remainder of the original lot.

20.6.3.2.2 Subdivide Lot by Parallel Edge

Position of option on menu: **Design =>Estate/Lots =>Edit lots =>Subdivide area =>Parallel edge**

This option adjusts the size of a lot by moving a side parallel to itself until the lot area has a new smaller area. A second lot is created from the remainder of the original lot.





The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick lot edge	string select		

pick the lot to adjust by picking the edge of the lot that is to be paralleled.

Lot area	output box		

area of the selected lot.

New area	input box		

required area of the adjusted lot. The area must be than the existing lot area.

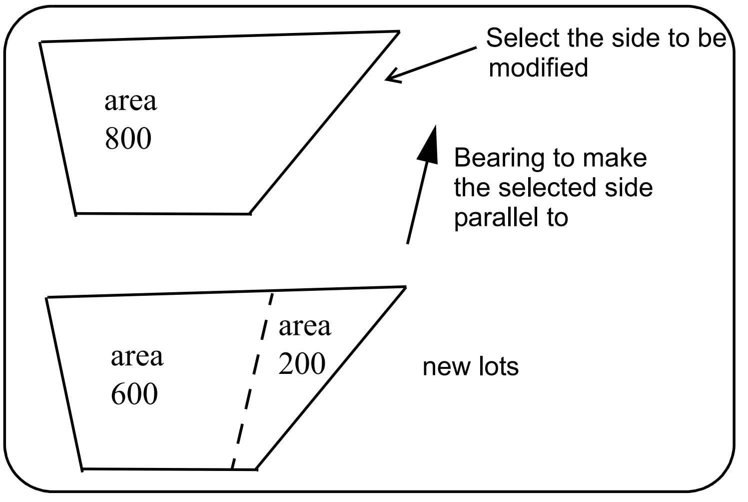
Process	button		

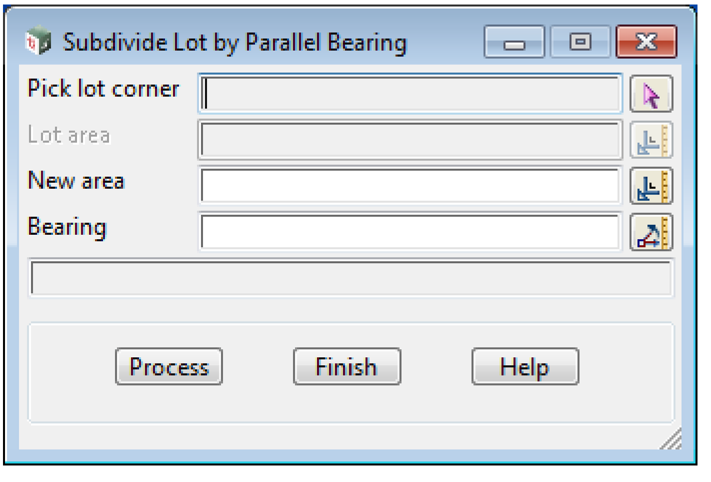
adjust the lot by paralleling the selected edge. A second lot is created from the remainder of the original lot.

20.6.3.2.3 Subdivide Lot by Parallel Bearing

Position of option on menu: **Design =>Estate/Lots =>Edit lots =>Subdivide area =>Parallel bearing**

This option adjusts the size of a lot by moving a side parallel to a given bearing until the lot area has a new smaller area. A second lot is created from the remainder of the original lot.





The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick lot corner	string select		

pick the lot to adjust by picking the edge of the lot that is to be paralleled.

Lot area	output box		

area of the selected lot.

New area	input box		

required area of the adjusted lot. The area must be smaller than the existing lot area.

Bearing	bearing box		

the bearing used for the selected edge.

Process	button		

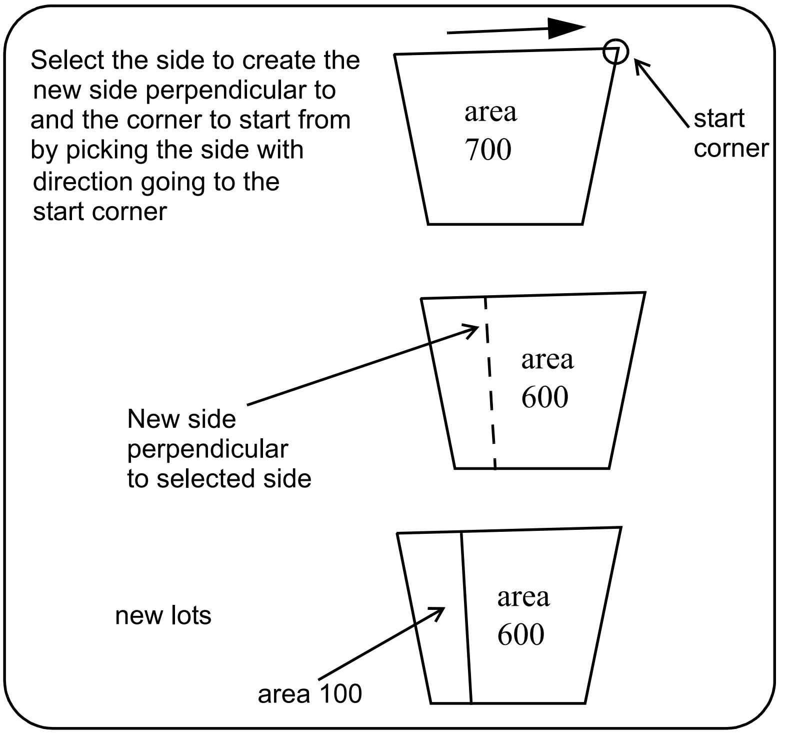
adjust the lot by paralleling the selected edge with the given bearing. A second lot is created from the remainder of the original lot.

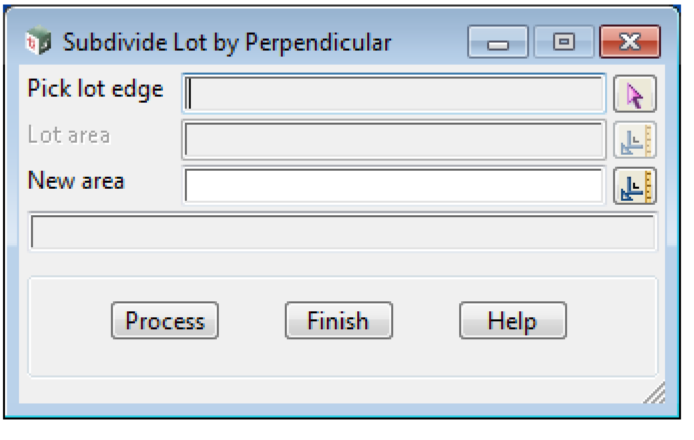
20.6.3.2.4 Subdivide Lot by Perpendicular

Position of option on menu: **Design =>Estate/Lots =>Edit lots =>Subdivide area =>Perpendicular**

This option adjusts the size of a lot by creating a new side perpendicular to a selected side and then moving the new side until the created lot area has a given smaller area. A second lot is created from the remainder of the original lot.

Note -the new area must be less than the original area.





The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick lot edge	string select		

pick the side of the lot that a new side will created perpendicular to

Lot area	output box		

area of the selected lot.

New area	input box		

required area of the adjusted lot. The area must be smaller than the existing lot area.

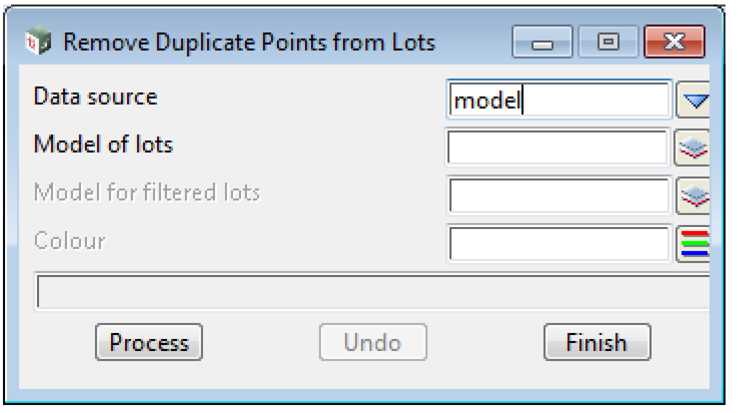
Process button

adjust the lot by moving a side whilst keeping it perpendicular to another side. A second lot is created from the remainder of the original lot.

20.6.4 Filter Lots

Position of option on menu: **Design =>Estate/Lots => Filter lots**

This option is used to remove adjacent duplicate points from lots.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Data source type

data source type.

Data source

data source for strings to convert to lots.

Model for filtered lots model box available models

model for the filtered lots.

Colour colour box available colours

if non-blank, the filtered lots are given this colour.

Otherwise a filtered lot has the same colour as the non-filtered lot.

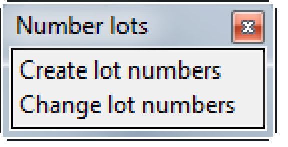
Process button

filter the selected lots.

20.6.5 Number Lots

Position of menu: **Design =>Estate/Lots => Number lots**

After lots are created, they are then numbered. The lot numbers are stored with the lot



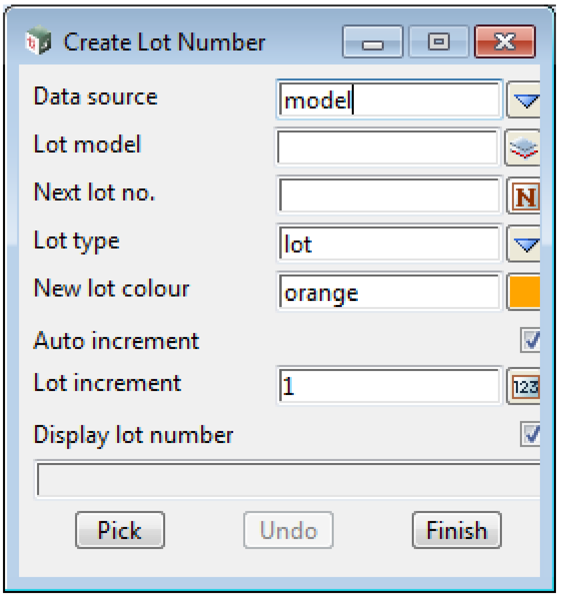
For the option *Create lot numbers*, go to 20.6.5.1 Create Lot Numbers.

Change lot numbers 20.6.5.2 Change Lot Numbers

20.6.5.1 Create Lot Numbers

Position of option on menu: **Design =>Estate/Lots => Number lots =>Create lot numbers**

This option creates lot numbers.



The fields and buttons used in this panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Data source

data source for lots that have been numbered. A new lot number cannot be the same as an existing one in the data source.

Next lot no.

the number to give the next lot that is numbered.

Lot type choice box user defined user defined

type of the lot. The choices are defined by the user in the file. See 20.6.1 Lot types

New lot colour colour box available colours

colour to created the numbered lot. This is done so that numbered and non-numbered lots can be easily distinguished.

Auto increment tick box tick

if ticked, the **Next lot no.** *is incremented after the lot is numbered.*

Lot increment integer box 1

amount to increment the **Next lot no** *if the* **Auto increment** *flag is tick.*

Display lot numbers tick box tick

if ticked, the lot number is temporarily displayed. Permanent lot numbers are created using the Label Lots options.

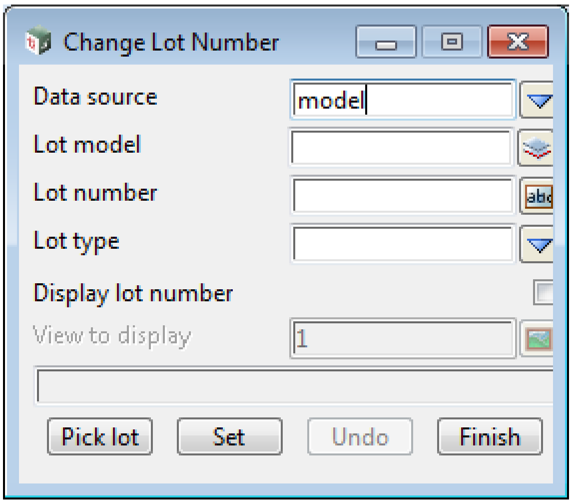
Pick string select

pick the lot the be numbered. After accepting the lot, it is given the **Next lot no.**

20.6.5.2 Change Lot Numbers

Position of option on menu: **Design =>Estate/Lots => Number lots =>Change lot numbers**

This option changes the lot numbers stored with the lot.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Lot number	text box	number of selected lot	

when a lot is selected, its lot number is displayed in this field. This can be changed and the selected lot will get the new **Lot number** *when the* **Set** *button is selected.*

Lot type	choice box	type of selected lot	

when a lot is selected, its lot type is displayed in this field. This can be changed and the selected lot will get the new **Lot type** *when the* **Set** *button is selected. See* 20.6.1 Lot types.

Display lot numbers	tick box		

if ticked, the lot numbers are displayed for the selected lots.

View to display	view box		available views

view to temporarily display the lot numbers.

Pick lot	button		

select the lot to change either the lot number or the lot type.

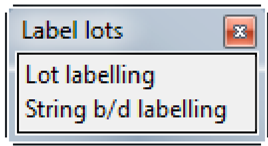
Set	button		

set the lot number and the lot type of the selected lot to the values given in the **Lot number** *and* **Lot type** *fields.*

20.6.6 Label Lots

Position of menu: **Design =>Estate/Lots => Label lots**

After lots are numbered, text for the lot numbers and areas can be displayed.

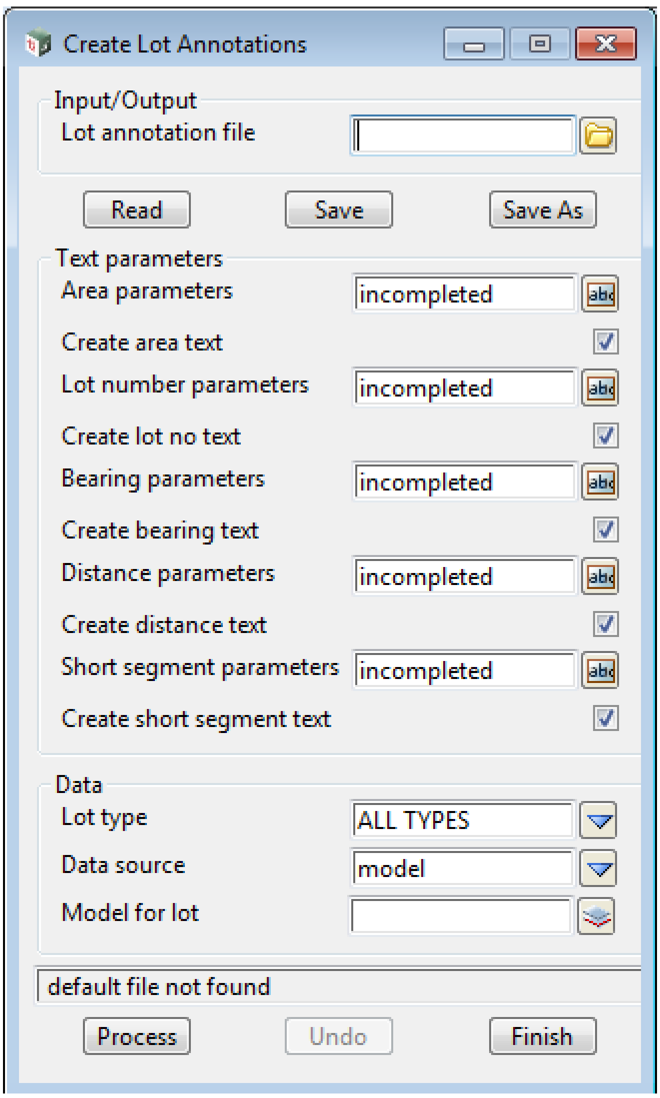


For the option *Lot labelling*, go to 20.6.6.1 Create Lot Annotation.
String b/d labelling 20.6.6.2 String Bearing Distance Labelling

20.6.6.1 Create Lot Annotation

Position of option on menu: **Design =>Estate/Lots => Number lots =>Lot labelling**

This option creates annotation of the lot numbers, lot areas, bearings of the sides and lengths of the sides.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Input/Output parameters

Lot annotation file *.laf files

file of defaults for labelling the lots.

Read button

read in a given lot annotation file.

Save button

write out the current parameters to the given lot annotation file.

Save as button

write out the current parameters to a different lot annotation file.

Text parameters

Area parameters

parameters for labelling the area of a lot.

Create area text tick box tick

if ticked, annotation for the areas is created using the **Area parameters**.

Lot number parameters

parameters for labelling the lot numbers.

Create lot no. text tick box tick

if ticked, annotation for the lot numbers is created using the **Lot number parameters**.

Bearing parameters

parameters for labelling the bearings of the sides of the lots.

Create bearing text tick box tick

if ticked, annotation for the lot bearings is created using the **Bearing parameters**.

Distance parameters

parameters for labelling the lengths of the sides of the lots.

Create distance text tick box tick

if ticked, annotation for the lengths of the sides is created using the **Distance parameters**.

Short segment parameters

parameters for specifying what is a "short Segment" and the parameters for labelling sides of the lots that are short segments.

Create short segment text tick box tick

if ticked, annotation for short segments is created using the **Short segment parameters**.

Data

Data source type

data source type.

Data source

data source for lots to be annotated.

Lot type choice box user defined user defined

of the lots selected by the data source, only lots of this type are labelled. See 20.6.1 Lot types.

Process button

select all the lots given by the **Data source** *and* **Lot type** *and create annotations according the* **Text parameters**.

20.6.6.2 String Bearing Distance Labelling

Position of option on menu: **Design =>Estate/Lots => Label lots =>String b/d labelling**

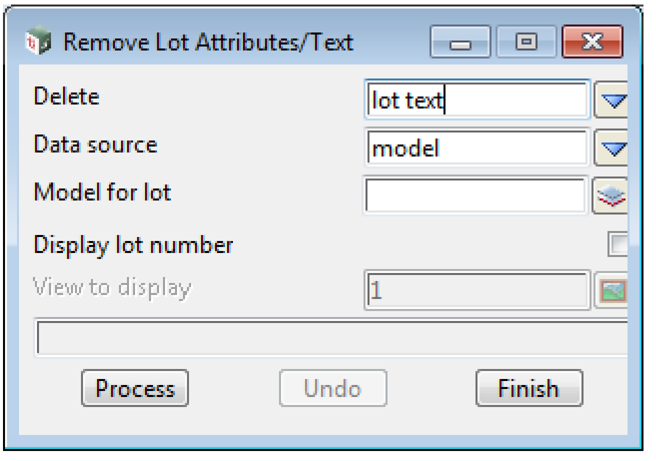
Option to label the bearing and lengths of segments of strings.

This option is described in the section 24.1 Bearing/Distance Labelling

20.6.7 Delete Lot Attributes and Text

Position of option on menu: **Design =>Estate/Lots => Delete lot attributes/text**

This option is used to remove the attributes for a lot which store the lot number, area etc. It can also delete the annotation text that has been created for the lot.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Delete	choice box	lot text	lot text, lot attributes

when a lot is selected, its lot type is displayed in this field. This can be changed and the selected lot will get the new **Lot type** *when the* **Set** *button is selected.*

Data source

data source for lot to delete either the lot text or lot attributes.

Display lot numbers	tick box		

if ticked, the lot numbers are displayed for the selected lots.

View to display	view box		available views

view to temporarily display the lot numbers.

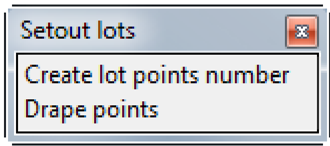
Process	button		

delete the selected attributes from the selected lots.

20.6.8 Setout Lots

Position of option on menu: **Design =>Estate/Lots => Setout lots**

Options to create unique point numbers for the vertices at the ends of each segment of the lots, and to create z-values from a given tin for the vertices of the lots.

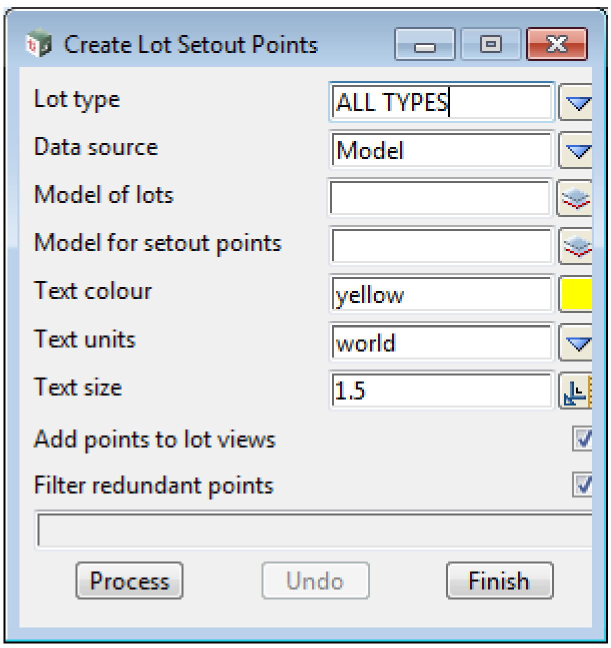


For the option *Create lot point numbers*, go to 20.6.8.1 Create Lot Point Numbers
Drape points 20.6.8.2 Drape Points

20.6.8.1 Create Lot Point Numbers

Position of option on menu: **Design =>Estate/Lots => Setout lots =>Create lot points numbers**

This option creates unique point numbers for the vertices at the ends of each segment of the lots. Duplicate vertices are only given one point number.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Lot type	choice box	user defined	user defined

of the lots selected by the data source, only lots of this type have point numbers created. See 20.6.1 Lot types

Data source

data source for lots to have point numbers created.

Model for setout points	model box		available models

model for the set out points.

Text colour	colour box		available colours

colour of the point numbers.

Text units	choice box	world	screen, paper, world

units for text size.

Text size double box

Size of the text for the point numbers.

Add points to lot views tick box tick

if ticked, add the model of point numbers to the views that the lots are displayed on.

Create short segment text tick box tick

if ticked, annotation for short segments is created using the **Short segment parameters**.

Filter redundant points tick box tick

if ticked, then if two adjacent line have the same bearing then the two segments are replaced by one segment. And if two adjacent arc segments could be the one arc, then they are replaced by one arc.

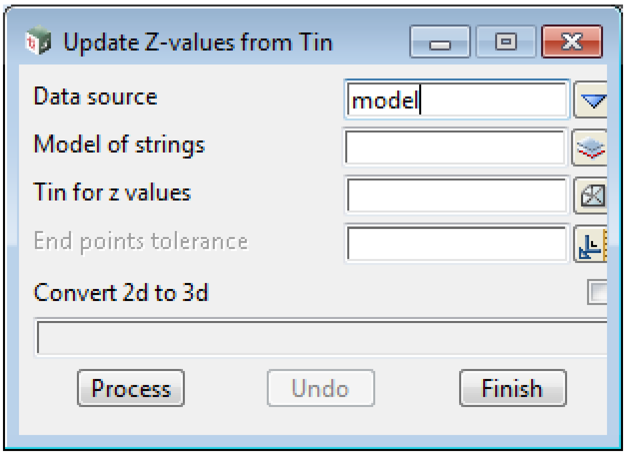
Process button

select all the lots given by the **Data source** *and* **Lot type** *and create annotations according the* **Text parameters**.

20.6.8.2 Drape Points

Position of option on menu: **Design =>Estate/Lots => Setout lots =>Drape points**

Option for setting the z-value of points to the z-value from a specified tin.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source	choice box		model, view, string

type of the data source

Model of strings / View of strings / Pick a String

source of the data

Tin for z values tin box

The tin to take the z values for the vertices from

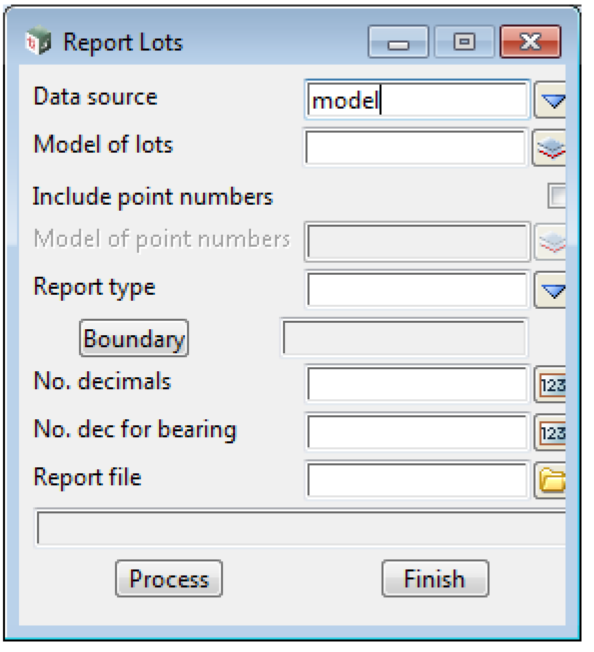
End points tolerance input

Convert 2d to 3d tick box

if ticked, 2d strings will be converted to 3d strings which have a different z value from the tin for each vertex.
if not ticked the 2d string takes its z value from the tin at the first vertex

20.6.9 Report Lots

Position of option on menu: **Design =>Estate/Lots => Report lots**



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source			

data source for lots to be reported.

Include point numbers	tick box		

if ticked, include the point number for each vertex around the lot.

Model of point numbers	model box		available models

model of the point numbers for the vertices of the lots.

Report type	choice box		sort by type, sort by number sort by area, setout

type of report for the selected lots.

Boundary	string select	optional	

if selected, the boundary string for the lots.If selected then the area of the boundary string is calculated and the total area of the lots compared to the area of the boundary string.

No. decimals	integer box		

Number of decimal places for co-ordinates and lengths in the report.

No. dec for bearings	integer box		

Number of decimal places for bearings in the report.

Report file	file box		

Name of the file for the report.

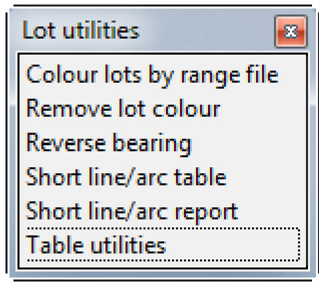
Process	button		

Create the report.

20.6.10 Lot Utilities

Various lot options.

Position of menu: **Design =>Estate/Lots => Lot utilities**

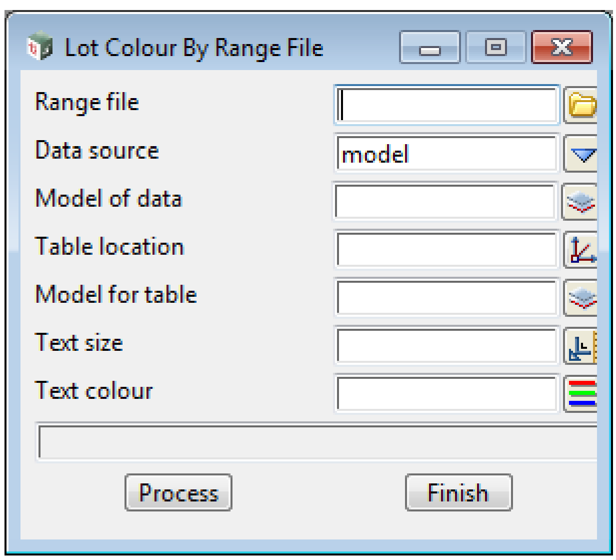


For the option *Colour lots by range file*, go to	20.6.10.1 Colour Lots by Range File
Remove lot colour	20.6.10.2 Remove Lot Colour
Reverse bearing	20.6.10.3 Reverse Bearing
Short line/arc table	20.6.10.4 Short Line/Arc Table
Short line/arc report	20.6.10.5 Short Line/Arc Report
Table utilities	20.6.10.6 Short Segment Utilities

20.6.10.1 Colour Lots by Range File

Position of option on menu: **Design =>Estate/Lots => Lot utilities =>Colour lots by range file**

The option colours lots by area using a lot range file (.lrf) to define the colour for each minimum and maximum area. It also creates a table of the statistics for the area ranges.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Range file	file box		*.lrf

range file for colouring lots.

Data source

data source for lots to be coloured.

Table location position select box

position of the table.

Model for table model box available models

model for the table of statistics on the lot areas.

Text colour colour box available colours

colour of the text.

Text size (w) double box

size of the text for the table.

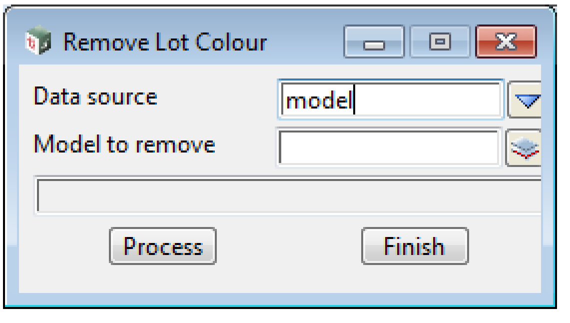
Process button

select all the lots given by the **Data source** *and* **Lot type** *and colour them according to the lot range file.*

20.6.10.2 Remove Lot Colour

Position of option on menu: **Design =>Estate/Lots => Lot utilities =>Remove lot colour**

The option removes the colour of all the selected lots.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Data source

data source for lots to have their colour removed.

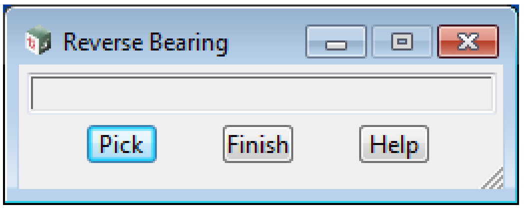
Process button

select all the lots given by the **Data source** *and remove the lot colour.*

20.6.10.3 Reverse Bearing

Position of option on menu: **Design =>Estate/Lots => Lot utilities =>Reverse bearing**

Option to change the value of the bearing text created by the lot labelling option by adding 180 degrees to the bearing.This is then the bearing of a line in the reversed direction to that represented by the original bearing text. The rounding for the reversed bearing is done according to the lot parameters.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Zero padding	tick box	tick	

if ticked, if the minutes and/or seconds are zero then two zeros are written out. If not ticked, if the minutes and/or seconds are zero then they are left out.

Pick	button		

pick the bearing text that is to be modified.

20.6.10.4 Short Line/Arc Table

Position of option on menu: **Design =>Estate/Lots => Lot utilities =>Short line/arc table**

Position of option on menu: **Drafting =>Text and Tables =>Short segment table**

Option to create a table of all the short segments and/or short arcs.

This option is described in the section 24.16.11 Short Segments Table

20.6.10.5 Short Line/Arc Report

Position of option on menu: **Design =>Estate/Lots => Lot utilities =>Short line/arc report**

Position of option on menu: **Drafting =>Text and Tables =>Short segment report**

Option to create a report of all the short segments and/or short arcs.

This option is described in the section 24.16.12 Short Segment Report

20.6.10.6 Short Segment Utilities

Position of option on menu: **Design =>Estate/Lots => Lot utilities =>Table utilities**

Position of option on menu: **Drafting =>Text and Tables =>Short segment table utilities**

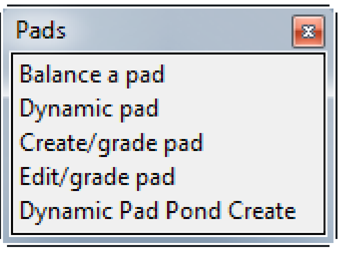
This panel is used modify an existing short segments table.

This option is described in the section 24.16.13 Short Segments Table Utilities

20.7 Pads

Position of menu: **Design =>Pads**

The **Pads** walk-right menu is



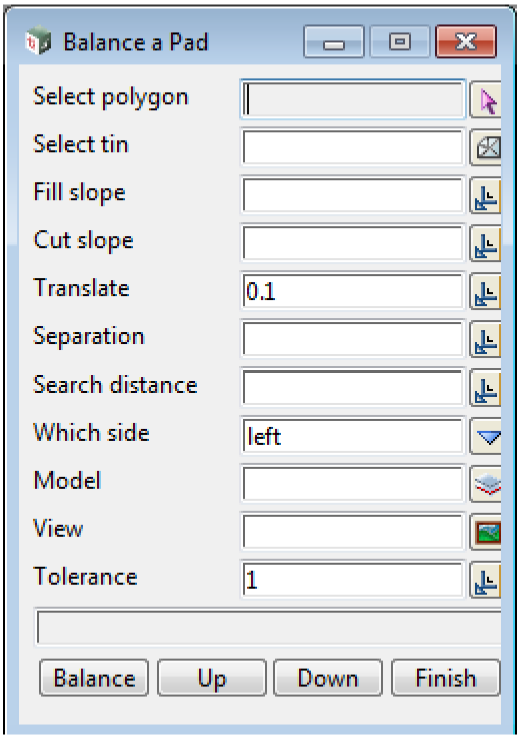
For the options see

Balance a pad	20.7.1 Balance a Pad
Dynamic pad	20.7.2 Dynamic Pad Interface
Create/grade pad	20.7.3 Allotment Pad Create
Edit/grade pad	20.7.4 Allotment Pad Edit
Dynamic Pad Pond Create	20.7.5 Dynamic Pad/Pond Interface

20.7.1 Balance a Pad

Position of option on menu: **Design =>Pads =>Balance a pad**

This option takes a pad formed by a string and user given slopes and batters from the pad to a given tin. The pad is moved up or down until a balance is found.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Select polygon	string select		

The string defining the pad

Select tin	tin select		

The tin to interface the pad to

Fill slope	measures box		At Point, Point to Point, String from Point, String to Point)

The fill slope for the pad

Cut slope	measures box		At Point, Point to Point, String from Point, String to Point)

The cut slope for the pad

Translate	measures box	0.1	At Point, Point to Point, String from Point, String to Point)

The value for translating the pad up and down with the 'Up' and 'Down' keys.

Separation	measures box		At Point, Point to Point, String from Point, String

			to Point)

The interval along the pad string for the volume calculations.

Search distance	measures box		At Point, Point to Point, String from Point, String to Point)

The maximum search distance for intersecting the cut/fill slopes with the pad.

Which side	choice box	left	left, right

Which side of the string is the outside of the pad.

Model	model box		available models

The model for the final strings and tadpoles.

View	view box		select view

View to add temporary strings to.

Tolerance

The tolerance for balancing the cut/fill for the pad.

Balance button

Balance the pad to within tolerance value.

Up button

Move pad up by translation value.

Down button

Move pad down by translation value.

Finish button

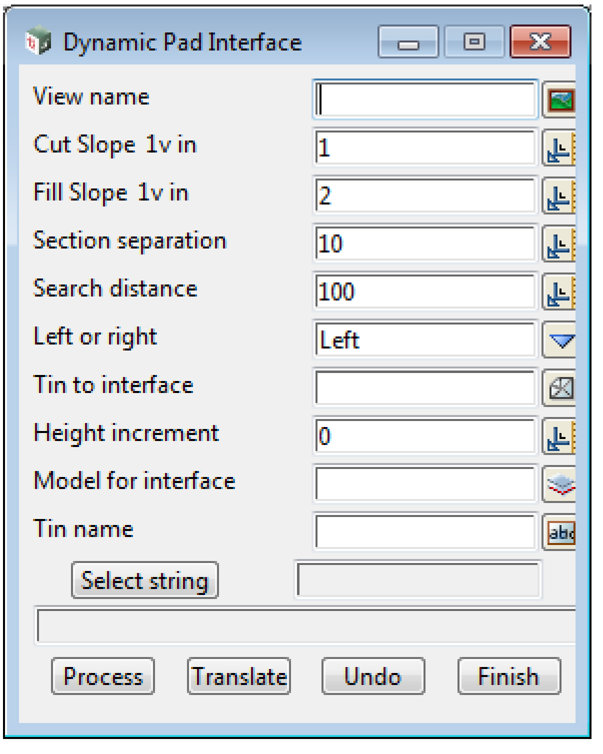
Add strings to permanent model and finish.

20.7.2 Dynamic Pad Interface

Position of option on menu: **Design =>Pads =>Dynamic pad**

This option is to create a pad by interfacing from a selected string and calculated the volume for the pad. The pad can be moved up or down and the new volumes are calculated.

The pad can also be moved in plan and the new interface and volumes are dynamically calculated.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
View name	view box		

name of the view to display the pad interface strings.

Cut slope 1v in		1	

cut slope for interfacing.

Fill slope 1v in		2	

fill slope for interfacing.

Section separation		10	

interval to calculate cut and fill batters.

Search distance		100	

distance for the batter to search to find intersection with the triangles.

Left or right	choice box	left	left, right

batter to the left or right of the selected string.

Tin to interface	tin box		

tin to batter to.

Height increment		0	

when process is selected, the pad height is adjusted by the value in this panel field and the new *interface and volumes are calculated.*

Select string select

select string to batter from

Process button

adjust the height of the pad by the Height increment and recalculate the new interface and volumes.

Translate

when selected the pad will move to the cursor position and dynamically calculate the interface and volumes.

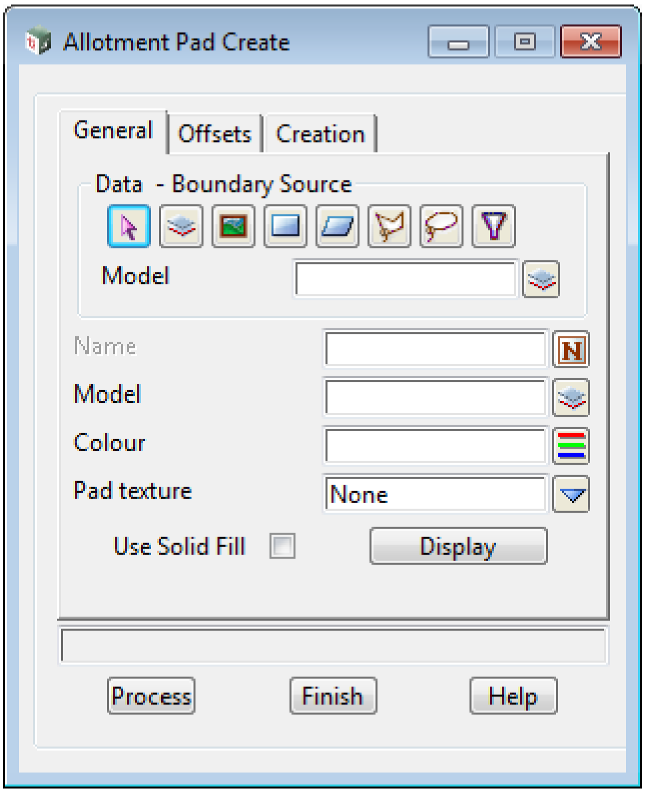
Reset button

reset the pad height to its original value.

20.7.3 Allotment Pad Create

Position of option on menu: **Design =>Pads =>Create/grade pad**

This option and the next **Design =>Pads =>Edit/grade pad** are used to create a pad from surrounding lines, grade the created pad and edit pads already created with this option.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

General tab

Data source type Model

data selection type - for a full description go to 4.19.3 Data Source

Data source input

source of data is used to create a pad.

Name input

the name of the new pad.

Model model box available models

name of the model that the new pad is in.

Colour colour box default colour available colours

the colour of the new pad.

Pad texture choice box None None, some textures

the colour of the new pad.

Use solid fill tick box

if ticked, solid fill the pad

Display button

brings up the **Display Text Defaults** *panel*

Offsets tab

Global offset input

if other offsets are not given, the distance to offset the sides of the pad from the strings used to create the pad

Offsets model model box

model for the offset lines

Offsets type choice box Segment Segment, Point to Point

*if **Segment**, define the offset line by picking a segment.*
*If **Point to Point**, define the offset line by picking two points.*

Select button

select an offset line

Creation tab

Pad level type choice box Typed input, Use tin level Use bdy/ht, Use bdy/slope

*if **typed input**, all vertices of the pad are given the height in the* **Lot level** *field.*
*If **use tin level**, the height of each vertex of the pad is taken from the tin given in the* **Tin** *field.*

If **typed input:**

Pad level input

the height to use for all the vertices of the pad.

If **Use tin level:**

Tin tin box available tins

the tin used to give heights to the vertices of the pad.

If **Bdy/ht:**

Height

the height to be above the selected boundary string

Reference boundary

the selected boundary string for the pad to be a given height above/below

If **Use/slope:**

Slope 1 in

the slope to be above the selected boundary string

Reference boundary

the selected boundary string for the pad to be a given slope above/below

Single/ Along a string radio buttons single

*if **single**, a single pad is created from the strings surrounded the selected point.*
*If **along a string**, pads are created surrounding each vertex of selected string.*

Centroid/search distance string select

*if **single,** a point is selected and then a circle is rubber banded until a second point is selected. The pad is created from segments within the circle.*

Search dist input

*if **along a string,** then for each vertex, all segments within the distance are processed to create the pad.*

Select string select

*if **along a string,** select the string to use. Pads are created surrounding each vertex of this string.*

Process button

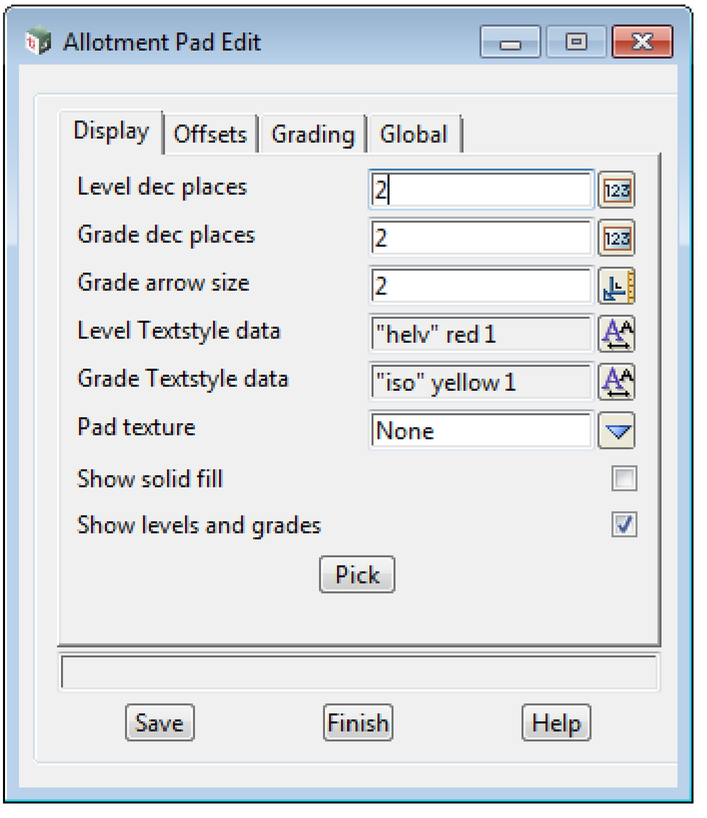
run the option

20.7.4 Allotment Pad Edit

Position of option on menu: **Design =>Pads =>Edit/grade pad**

This option is used to grade a pad created with the previous option, **Design =>Pads =>Create/grade pad.**

Selecting **Edit/grade pad** brings up the **Allotment Pad Edit** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Display tab

Level dec places input

the number of decimal places used to display the height (level) of the pad vertices.

Grade dec places input

the number of decimal places used to display the grade of the pad sides.

Grade arrow size input

size of the grade arrows

Level textstyle data textstyle data box

the textstyle data used for the displaying the level (height).

Grade textstyle data textstyle data box

the textstyle data used for the displaying the grade.

Pad texture choice box None None, some textures

the colour of the new pad.

Show solid fill tick box

if ticked, solid fill the pad

Show levels and grades tick box

if ticked, turn on the level and grades

Pick button

select the pad to edit

Offset tab

Maintain level/Maintain level on Grade radio buttons

*if **Maintain level**, keep the level when a new offset/boundary is selected*
*If **Maintain level on grade**, keep the level on grade when a new offset/boundary is selected*

Select new offset string select

select a new offset string

Select new boundary string select

select a new boundary string

Grading tab

Level and/or grade button

Level entire pad button

Level by str/ht button

Level by str/slope button

Level by seg/ht button

Level by seg/slope button

Batter slope button

Surface grading button

Ht

heights to increment the selected pad by

Up button

move the pad up by the give height

Down button

move the pad down by the give height

Pick (ht) string select

select a pad

Batter slope model model box available models

model for the batter slopes

Global tab

Data source type Model

data selection type - for a full description go to 4.19.3 Data Source

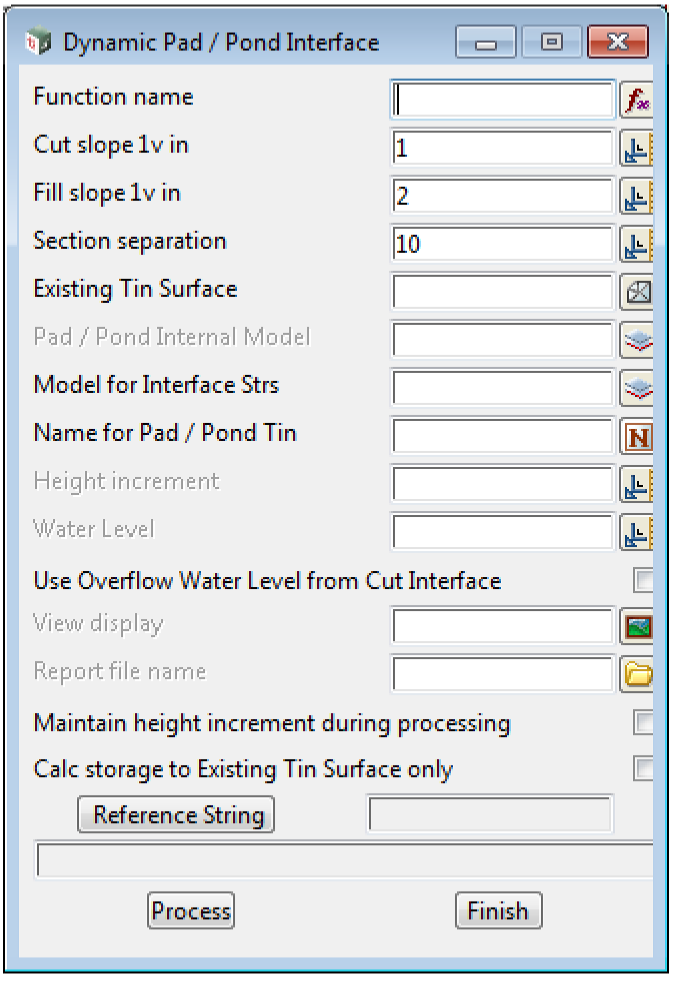
Data source input

source of data is used to create a pad.

20.7.5 Dynamic Pad/Pond Interface

Position of option on menu: **Design =>Pads =>Dynamic Pad Pond Create**

This option is used to interface from a string to a tin plus it can have an included pond model.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function name	function box		available functions

name of the pad/pond function. If the function already exists and is picked from a popup or and an <enter> is given at the end of the name, the information from the existing function will be placed in the appropriate panel fields

Cut slope 1 v in	input	1.0	

*the slope of the interface line if a point is in cut. A cut slope of one vertical to the given value of horizontal units is used. Positive is **up** for a cut slope. A value 0 is used to designate a horizontal line.*

Fill slope 1 v in	input	1.0	

*the slope of the interface line if the point is in fill. A fill slope of one vertical to the given value of horizontal units is used. Positive is **down** for a fill slope. A value 0 is used for a horizontal line.*

Section separation	input	10	

the distance between the points on the selected string that interface points will be calculated from.

Existing tin surface	tin box		available tins

name of the tin to interface to.

Pad /Pond internal model model box available model

if non blank, name of the model to include as part of the interface model

Model for interface strs model box available models

name of the model to contain the calculated interface string.

Name for pad/pond tin tin box available tins

name of the tin to create from the reference string, the interface string and the pad/pond internal model

Height increment input

value to vertically translate the reference string by (cleared after each process unless **Maintain height increment during processing** *is ticked)*

Water level input

if non blank, the height of the water level

Use overflow water level from cut interface tick box

if ticked, the water level to the lowest point on the cut interface is determined and used as the water level height

View display view box available views

if non blank, the view to display the results on

Report file name file box *.rpt files

if non blank, the view to display the results on

Maintain height increment during processing tick box

if ticked, the reference string will be raised by the **Height increment** *each time the function is processed, but only while the panel is open*

Calculate storage to existing tin surface only tick box

if ticked, this will calculate the storage to the **Existing tin surface** *from the* **Water level** *only*

Reference string string select

select the string to interface from.

Process button

run the function

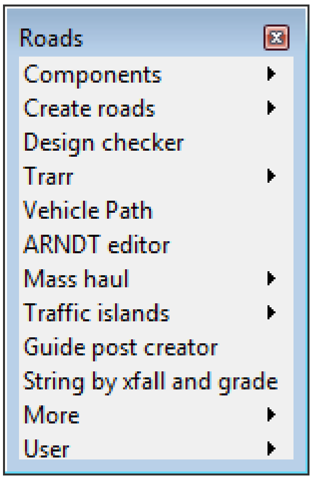
20.8 Roads

Position of menu: **Design =>Roads**

The road options include road network creation options and a number of other options for creating road details.

Some of the options are currently under development.

The **roads** walk-right menu is



generate components
setup and create road network

stopping distance calculations
input/output to TRARR
run TMR VPATH
editor for creating/running TMR ARNDT
generate data for mass haul on long section
create traffic islands
create guide posts to AUSTROAD standard
lots of generating strings by Xfalls
more roads macros

For the options see

Components	20.9 Components
Create roads	20.8.1 Create Roads
Design checker	20.8.2 Design Checker
Stopping distance	20.8.3 Stopping Distance
Trarr	20.8.4 Trarr
Vehicle path	20.8.5 Vehicle Path
ARNDT editor	20.8.6 ARNDT Editor
Mass haul	20.8.7 Mass Haul
Traffic islands	20.8.9 Traffic Islands
Guide post creator	20.8.10 Guide Post Placement
String by xfall and grade	20.8.11 String by Xfall and Grade
More	20.8.12 More Roads

20.8.1 Create Roads

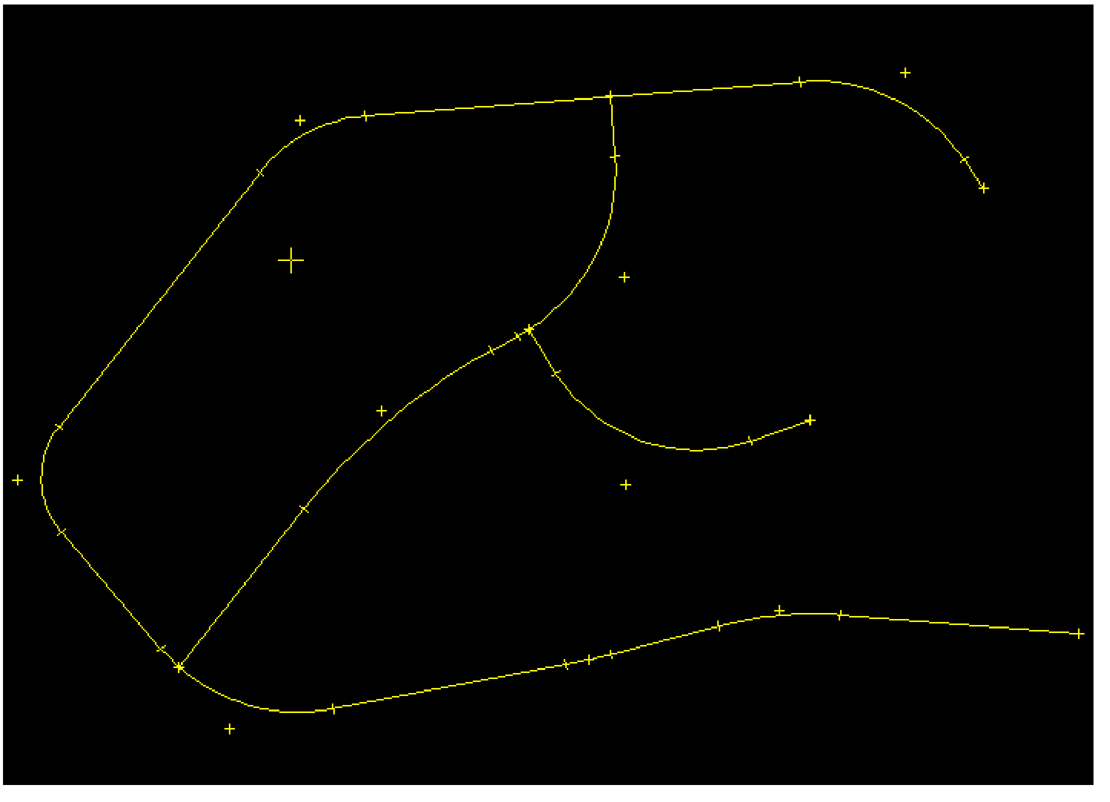
Position of menu: **Design =>Roads =>Create roads**

The **Create roads** functions allow the user to start with a model of road centrelines with vertical geometry, and unique string name.

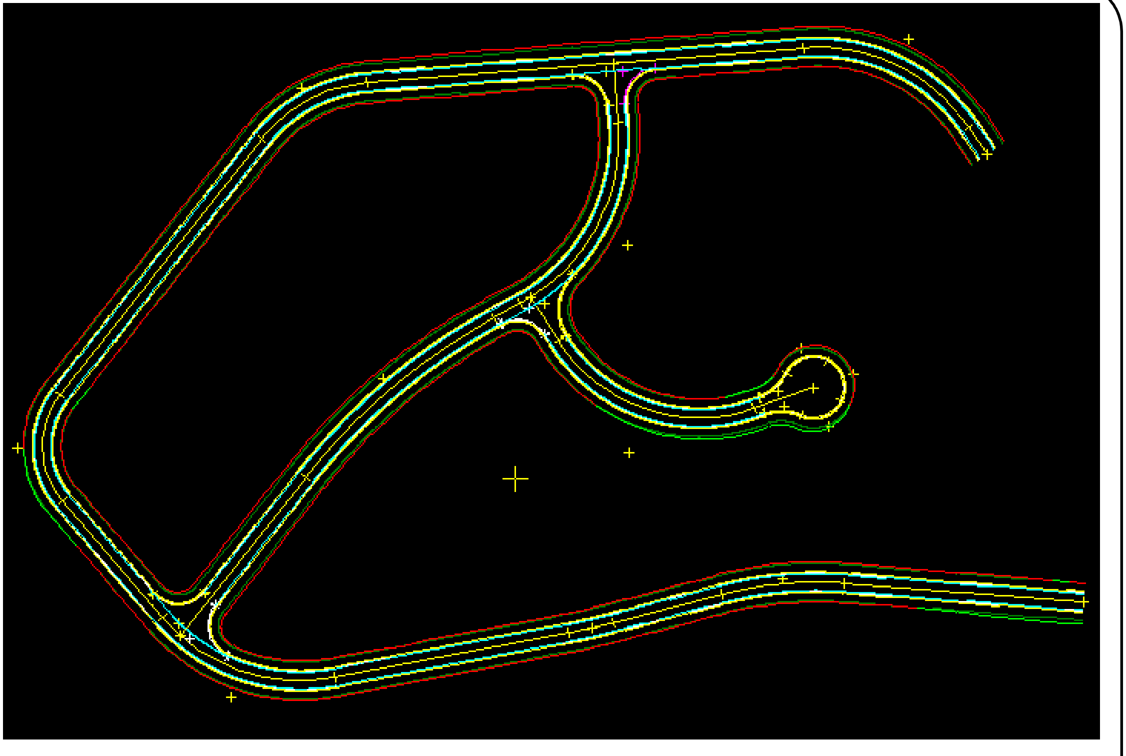
The centrelines can then be tagged with the width of the roads, left and right turn radii for any intersections and the definitions of any culdesacs at the end of a centreline.

The **Create** option will then create the left and right edges of the roads, create kerb returns and any intersections and the appropriate culdesacs.

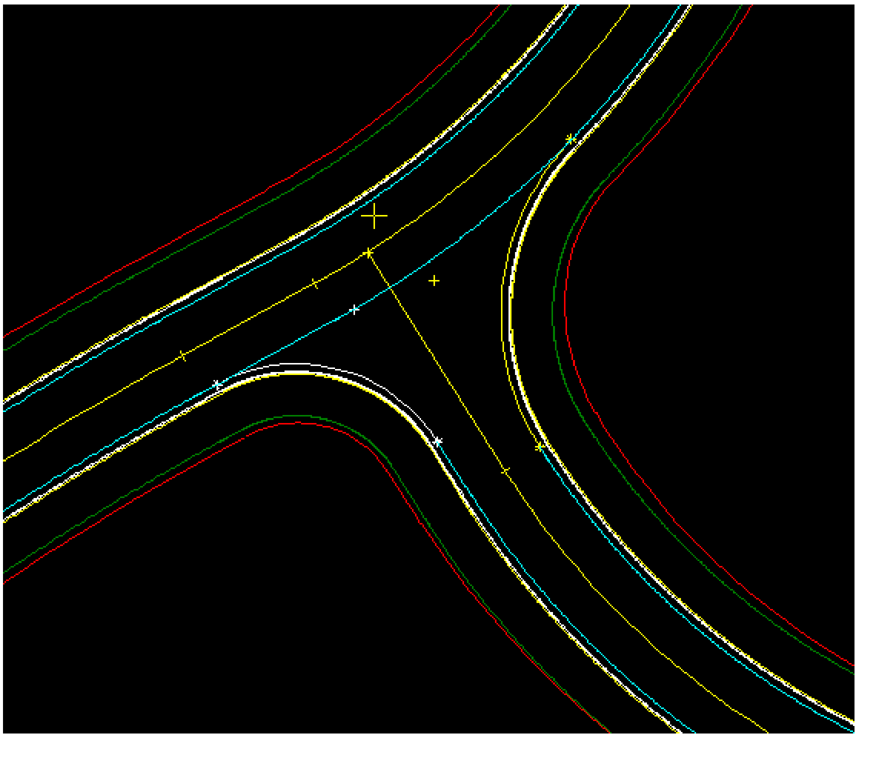
For example



Initial Network of Centrelines

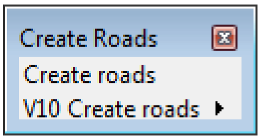


Road Network After Running the Roads Option



Blow Up of an Intersection

The **Create Roads** walk-right menu is



Create Roads
V10 Create Roads - superseded

See

Create Roads	20.8.1.1 Create Roads - Enhanced
V10 Create roads	20.8.1.2 V10 Create Roads

20.8.1.1 Create Roads - Enhanced

Position on menu: **Design =>Roads =>Create =>Create Roads - Enhanced**

The **Create Roads - Enhanced** panel is a total rewrite of ***Create Roads*** with many new features including:

(a) Either *templates* **or** snippets can be used to define the roads, intersections and kerbs.

(b) Wherever possible computators have been used to define kerb returns and culdesacs, and Smart Start and End modes in the Apply MTFs rather than chainage values.

(c) There is now a separate template for the left and right side of the road but unlike **Create Roads**, there is no separate Carriageway template.

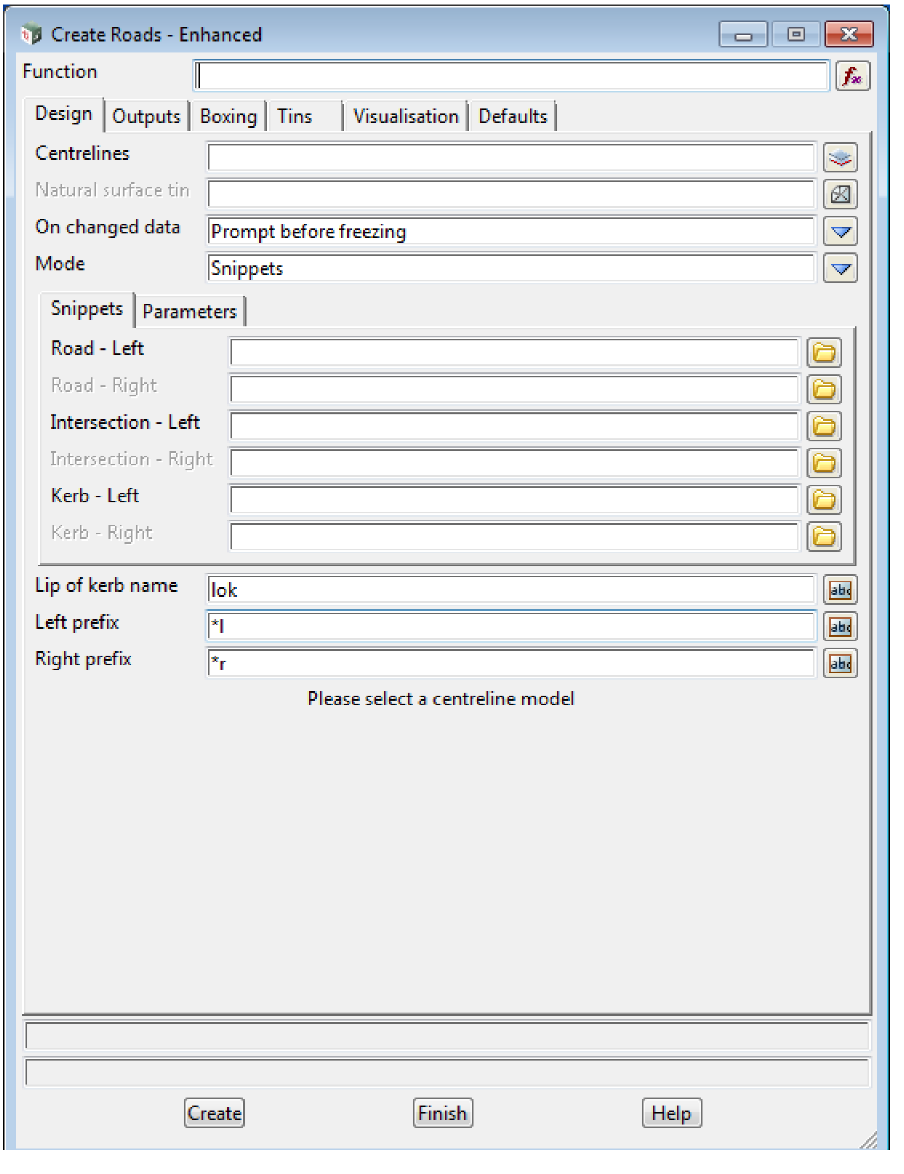
(d) There are default values for culdesacs.

(e) A kerb can be frozen without freezing the roads.

(f) Snippets can use the boxing defined in the **Boxing** tab or snippets can do their own boxing. But if you do both then you are on your own.

(g) Left and Right prefix to use when creating the Apply MTFs.

Selecting **Create Roads - Enhanced** brings up the **Create Roads - Enhanced** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function	function box		available create roads enhancement functions

name of the create roads advanced function

Available tabs

Go to Design tab
Go to Outputs tab
Go to Boxing tab
Go to Tins tab
Go to Visualisation tab
Go to Defaults tab
Go to Buttons at bottom

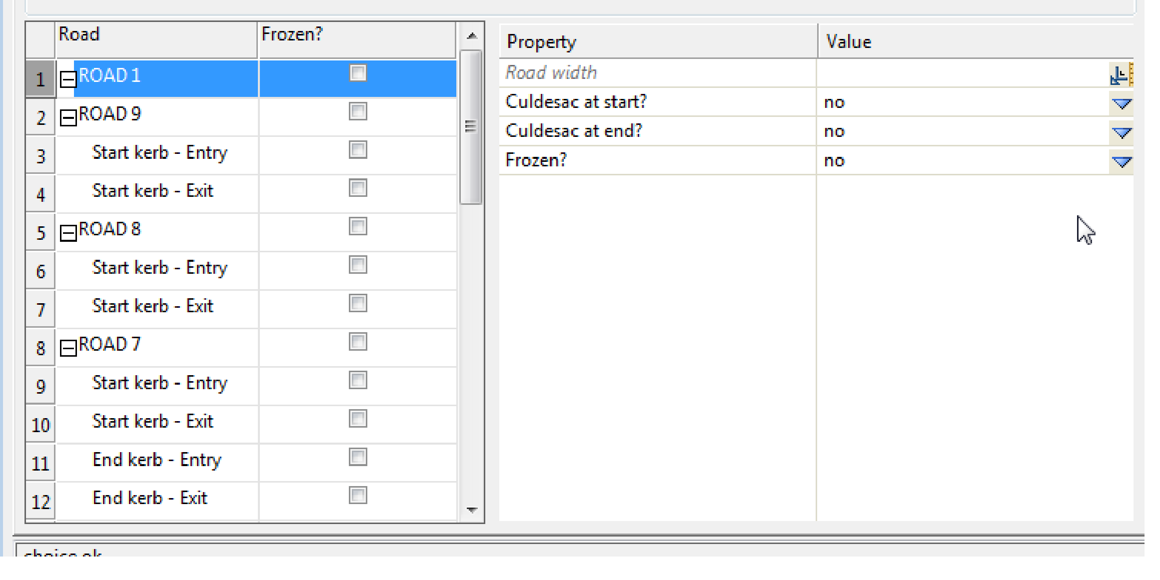
Design tab

Centrelines	model box		available models

model of the centreline lines that are to be processed to create roads, intersections and culdesac. Each centreline must have a unique name and vertical geometry.

After the Centreline model is selected either from the pop-up or by entering a model name and pushing the <Enter> key, the centrelines are analysed to see if left and right kerbs are possible (centreline ending on another serenely), if a culdesac is possible (end of a centreline) etc and then all the centrelines are listed in a Property Sheet on the bottom of the panel.

The entry for each centreline is where you enter values to override the defaults for road width, kerb radius, culdesac parameters etc. This information is then stored ***in the function*** *rather than as attributes on the centreline strings.*



Road grid - list of all the centrelines

Road

centreline name

Frozen	tick box		

if ticked, the centreline is frozen out of the automatic Create Roads process.

*For a frozen centreline, when the **Create Roads** function is run, the Apply MTF for that centreline is deleted and re-created. However any existing Apply MTF for that centreline will be run. Kerb returns are still calculated using the lip line create by running the existing Apply MTF for the frozen centreline. Hence the mtf for a frozen centreline can be modified and the changes will **not** be deleted when the Create Roads function is recalced.*

Natural surface tin	**tin box**	**available tins**

*if **not blank**, the tin to use as the natural surface for battering in the Applies.*
*If **blank**, no battering is done.*

On changed data	choice box	Prompt before freezing Always freeze Always ignore

*if **Prompt before freezing**,*
*If **Always freeze**,*
*If **Always ignore**,*

Mode	**choice box**	**Templates, Snippets**

*if **Templates**, templates are used to define the road.*
*If **Snippets**, snippets are used to define the road.*

*For information on the choice **Template**, go to* If Mode is Templates:
*For information on the choice **Snippets**, go to* If Mode is Snippets:
For the next tab, go to Outputs tab

If **Mode** is T**emplates:**

Templates

Road - left	template box	available templates

*the template to use on the left hand side of the centreline. The width to the lip of kerb name is modified to match the given **Road width**. Through intersection, only the template links out the Lip of kerb name are used so no battering occurs through an intersection.*

Road - right	template box	available templates

*the template to use on the right hand side of the centreline. The width to the lip of kerb name is modified to match the given **Road width**. Through intersection, only the template links out the Lip of kerb name are used so no battering occurs through an intersection.*

Kerb only	template box	available templates

a template with no carriageway part. It is applied to the generated kerb returns

Lip of kerb name	text box

*name of the lip of kerb link that must exist in the left and the right templates. Modifiers are applied to make the width to the lip of kerb to be the **Road width**. The kerb returns are created from the lip of kerb on each road, and the start height and grades are taken from the lip of kerb.*

Left/Right prefix	text box

*prefix/postfix (pre*post) to be used as the LHS/RHS prefix in the created Apply MTFs. These are then applied to the left/right template string names. If pretext only, just give the text. If post text is required, precede it by a* *****

Return to Mode choice box Templates, Snippets *or go to the next tab* Outputs tab

If **Mode** is **Snippets:**

Snippets tab

Road - left snippet box available snippets

*the snippet to use on the left hand side of the centreline. The width to the lip of kerb name is modified to match the given **Road width**. The snippet is stopped at intersections.*

Road - right snippet box available snippet

*the template to use on the right hand side of the centreline. The width to the lip of kerb name is modified to match the given **Road width**. The snippet is stopped at intersections.*

Intersection - left snippet box available snippets

*the snippet to use on the left hand side going through intersections. The width to the lip of kerb name is modified to match the given **Road width**.*

Intersection - right snippet box available snippets

*the snippet to use on the right hand side going through intersections. The width to the lip of kerb name is modified to match the given **Road width**.*

Kerb - left snippet box available snippets

the snippet that is applied to the generated left hand kerbs.

Kerb - right snippet box available snippets

the snippet that is applied to the generated right hand kerbs.

Lip of kerb name text box

*name of the lip of kerb link that must exist in the left and the right snippets. Modifiers are applied to make the width to the lip of kerb to be the **Road width**. The kerb returns are created from the lip of kerb on each road, and the start height and grades are taken from the lip of kerb.*

Natural surface tin **tin box** **available tins**

if non blank, the tin to use as the natural surface for battering in the Applies.
If blank, no battering is done.

For more information on snippets, see 21.5 Defining and Using Snippets.

Return to Mode choice box Templates, Snippets *or go to the next tab* Outputs tab

Outputs tab

Apply MTF functions - Roads

Apply MTF function stem text box APPLY *

*pre*post text to use around the centreline name to create the Apply MTF function names for the roads along each centreline. The function names will be unique because the centrelines names must be unique within the centrelines model.*

Road strings stem text box * STRS

*pre*post text to use with the function name in the created Applies for the design string models for each centreline*

Road sections stem text box * XSECS

*pre*post text to use with the function name and the created Applies for the design cross section models for each centreline*

Road polygons mode choice box Model per centreline Model per centreline

One model
No Polygons

*if **Model for Centreline**, different models are created for the polygons in each Apply MTF.*
*If **One model**, all the polygons created by the Applies are placed in one model.*
*If **No Polygons**, no polygons are created by the Applies.*

If **Road polygons mode** is **Model per centreline:**

Road polygons text box * POLYS

*pre*post text to use with the function name in the created Applies for polygon models for each centreline*

If **Road polygons mode** is **One model:**

Road polygon model model box available models

*when **One model**, the model for all the polygons created by the Applies.*

Kerb Returns

Apply MTF function stem text box KRET APPLY *

*pre*post text to use around the kerb return names to create the Apply MTF function names for the kerb returns. The function names will be unique because the kerb return names are uniquely derived from the centreline names.*

Naming choice box Follow Road Names - LHS driving
Follow Road Names - RHS driving
Use Numbers
Use Letters

method of naming the kerb returns

Kerb returns model model box available models

model for all the kerb returns

Kerb returns Apply MTF strings model model box available models

model for all the strings produced by the kerb return Applies

Kerb returns Apply MTF sections model model box available models

model for all the x-sections produced by the kerb return Applies

Apply MTF map file map file box available map files

*if **not blank**, the name of the map file to use in the created Apply MTFs*

Output chain chain box available chain files

*if **not blank**, a chain is created of all the Apply MTFs that are run by the* **Create Roads - Advanced** *panel **in the order** that they need to be run.*

Go to the next tab Boxing tab *or return to* Available tabs

Boxing tab

boxing can be created for all the road and kerb return strings. The boxing definitions must have a common centreline name in them which will be replaced by the name of each of the road/kerb return centrelines.

Create boxing tick box

if ticked, boxing is applied for all the road and kerb returns. The boxing definitions must have a common centreline name in them which will be replaced by the name of each of the road centrelines or kerb returns.

Default boxing file file box *.blf files

file of boxing file definitions

Default boxing CL delimiter text box

common name in each boxing definition which will be replaced by the name of each road/kerb return centreline

Model stem for boxing strings text box

*pre*post text to use with the road/kerb return function name for boxing string models*

Boxing sections text box

*pre*post text to use with the road/kerb return function name for boxing section models*

Boxing kerb return sections and strings in one model tick box

if ticked, the strings for all the kerb return Applies are place in one model rather than one model for each kerb return. Similarly, the sections for all the kerb return Applies are placed in the one model. If not ticked, the model names are created in the same way as for the road centrelines. That is, the Model stem for boxing strings and sections are used with the kerb return function name and the layer stems to produce the kerb return boxing strings and sections models.

Model name for boxing kerb strings model box available models

*when only one model (per layer) for the box kerb strings is used - the base name for the model of kerb strings created for each boxing layer. The pre*post text in the Layer Stem is applied to this name.*

Model name for boxing kerb sections model box available models

*when only one model (per layer) for the box kerb sections is used - the base name for the model of kerb sections created for each boxing layer. The pre*post text in the Layer Stem is applied to this name.*

Boxing Layer 1-8 in the grid:

if non blank, boxing strings and sections will be created for that layer.

The last boxing layer is also referred to as the ***subgrade*** *layer.*

Layer stem text box

*for this boxing layer, the pre*post text used when naming models for boxing sections and strings.*

Road - LHS text box

for this boxing layer, the boxing definition to use on the left hand side of road centrelines

Road - RHS text box

for this boxing layer, the boxing definition to use on the right hand side of road centrelines

Kerb return text box

for this boxing layer, the boxing definition to use on the kerb returns

Intersection - LHS text box

for this boxing layer, the boxing definition to use on the left hand side of the road though an intersection. The strings and sections are place in the models for the road centreline going through the intersection.

Intersection - RHS text box

for this boxing layer, the boxing definition to use on the right hand side of the road though an intersection. The strings and sections are place in the models for the road centreline going through the intersection.

Go to the next tab Tins tab *or return to* Available tabs

Tins tab

Create tin tick box

if ticked, create a tin for all the roads, kerb returns and culdesacs

Road tin tin box available tins

name for the tin created from the all the strings and sections

Road tin model model box available models

model for the Road tin

Road tin colour colour box available colours

colour of the road tin

Subgrade tin tin box available tins

if non blank, a subgrade tin of this name is created

Subgrade tin model model box available models

model for the Subgrade tin

Subgrade tin colour colour box available colours

colour of the subgrade tin

Nulling angle real box measures pop up

angle for nulling the triangles

Nulling length real box measures pop up

length for nulling the triangles

Nulling Seed points

Seed points are used for nulling triangles in the tins

Seed X/Y two column rid

X/Y coordinate of a nulling seed points.

Add button

after clicking on ***Add****, a seed point is selected from a view and the (X,Y) coordinates are written to the Seed X/Y grid.*

Remove button

*clicking on **Remove** will delete the highlighted row from the Seed X/Y point grid.*

Go to the next tab Visualisation tab *or return to* Available tabs

Visualisation tab

Create visualisation tick box

if ticked, the road tin is processed for visualisation

Apply texture map to tin tick box

if ticked, the texture map is applied to the road tin

Draw road tin as solid on view tick box

if ticked, draw the triangles of a tin as solid colour on the view

Tin colour to use for cut polygons colour box available colours

colour to use for all the cut polygons

Tin colour to use for fill polygons colour box available colours

colour to use for all the fill polygons

Tin colour to use for transition polygons colour box available colours

colour to use for all the polygons in neither cut nor fill

Go to the next tab Defaults tab *or return to* Available tabs

Defaults tab

Roads

Road width real box 3.5 measures pop up

width to be used for the left and right side of the road if one hasn't been defined for any centrelines

Section separation real box 5 measures pop up

chainage distance to use for the Applies created for the roads along the centrelines

Chord/Arc tolerance real box 0.01 measures pop up

chord to arc tolerance to use in the Applies for the centrelines.

Kerb returns

Kerb radius real box 5 measures pop up

radius to use for filleting between any centrelines that don't have a left or right radius defined

Section separation real box 1 measures pop up

chainage distance to use to use for the Applies that are created for the kerb returns

Chord/Arc tolerance real box 0.001 measures pop up

chord to arc tolerance to use for the Applies that are created for the kerb returns

Culdesacs

Bulb radius	real box	9	measures pop up

radius of the bulb of the culdesac

Bulb offset	real box		measures pop up

offset of the centre of the culdesac bulb from the centreline.

Left width	real box	3.5	measures pop up

width of the left hand side of the throat of the culdesac

Right width	real box	3.5	measures pop up

width of the right hand side of the throat of the culdesac

Left radius	real box	15	measures pop up

fillet radius of the curve on the left hand side going from the beginning of the culdesac to either the bulb of the culdesac, or onto the tangent between this curve and the bulb of the culdesac.

Right radius	real box	15	measures pop up

fillet radius of the curve on the right hand side going from the beginning of the culdesac to either the bulb of the culdesac, or onto the tangent between this curve and the bulb of the culdesac.

Left tangent	real box		measures pop up

if non zero, there is a straight line of this length that is the tangent between the fillet curve on the left hand side at the beginning of the culdesac and the bulb of the culdesac. This can not be a negative value.

Right tangent	real box		measures pop up

if non zero, there is a straight line of this length that is the tangent between the fillet curve on the right hand side at the beginning of the culdesac and the bulb of the culdesac. This can not be a negative value.

Projection	real box		measures pop up

*if non zero, the end of the centreline with the culdesac on it is extended by this distance. The centre of the culdesac bulb is then placed at the perpendicular distance of **Bulb offset** from the extended point.*

Go to the next tab Buttons at bottom *or return to* Available tabs

Buttons at bottom

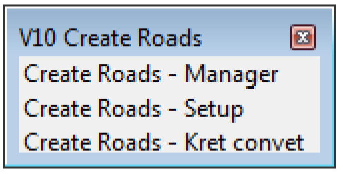
Create	button

run the option and create the road network.

Return to 20.8.1 Create Roads

20.8.1.2 V10 Create Roads

The Create **Roads** walk-right menu is



construct the roads network
setups for road network
convert kerb returns

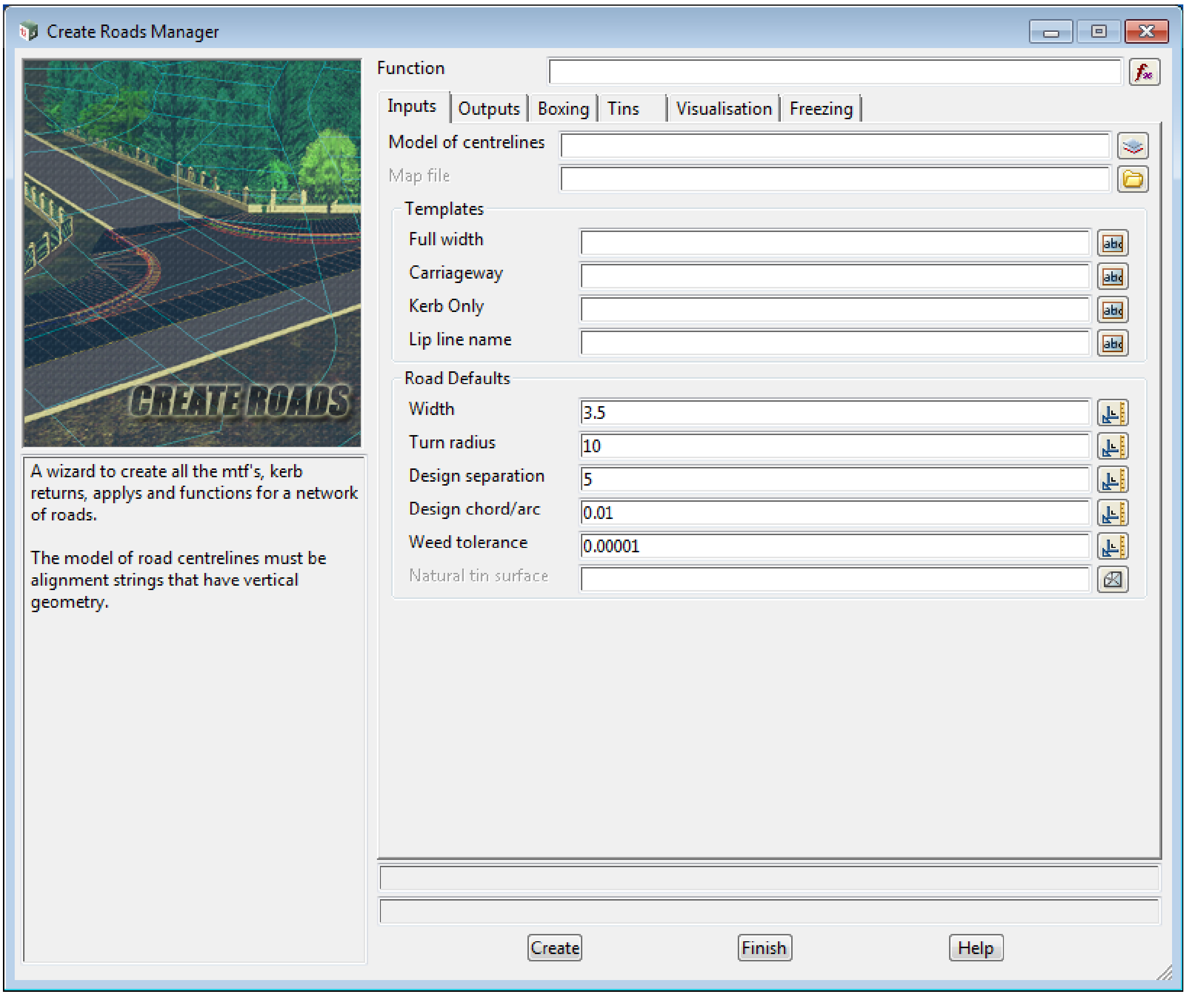
For the options see

Create Roads - Manager, go to 20.8.1.2.1 Create Roads - Manager
Create Roads - Setup 20.8.1.2.2 Create Roads - Setup
Create Roads - Kret convert 20.8.1.2.3 Create Roads - Kreturns Convert

20.8.1.2.1 Create Roads - Manager

Position of option on menu: **Design =>Roads =>Create =>Create Roads - Manager**

This option takes all the centrelines in a model and uses road widths and left and right turn radii to form roads on each centreline.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function	function box		available roads functions

name of the create roads function

Input tab

Model of centrelines	model box		available models

model of the centreline lines that are to processed to create roads, intersections and culdesac. Each centreline must have a unique name and vertical geometry

Templates

Full width	template box		available templates

template with one fixed link and then cut and fill definitions to use for creating the roads. If a centreline has a road width defined, then the template is modified to use those values. The cut/fill section of the template are used to batter as normal.

Carriageway template box available templates

a template with only a fixed part to the end of the carriageway which is applied across any intersections - going from the chainage of the start of the kerb return on one side of the intersection to the chainage at the end of kerb return on the other side of the intersection.

Kerb only template box available templates

a template with no carriageway part. It is applied to any kerb returns

Lip line name

name of the lip line that must exist in each of the full width template and the road template. The kerb returns are created from the lip line on each road and the start height and grades are taken from the lip lines.

Road Defaults

Width

width of the road to be used if one hasn't been defined for any centrelines

Turn radius

radius to use for filleting between any centrelines that don't have a left or right radius defined

Design separation

chainage distance to use in the Applies for creating the roads along the centrelines

Default chord/arc

chord to arc tolerance to use on curves

Natural surface tin **tin box** **available tins**

if non blank, the tin to use as the natural surface for battering in the Applies.
If blank, no battering is done.

Output tab

Model stems

Road strings input

*pre*post text to use with the function name in the created Applies for the design string models for each centreline*

Road sections input

*pre*post text to use with the function name and the created Applies for the design cross section models for each centreline*

Polygons choice box One model One model, No Polygons Model for Centreline

*if **One model**, all the polygons created by the applies are place in one model.*
*If **Model for Centreline**, different models are created for the polygons in each apply.*
*If **No Polygons**, no polygons are created by the applies.*

Model model box available models

*when **One model**, model for all the polygons created by the Applies.*

Road polygons input

*pre*post text to use with the function name in the created Applies for polygon models for each centreline*

Kerb Returns

Turn separation

chord/arc tolerance for use in the kerb return Applies

Turn chord/arc

chord to arc tolerance to use on kerb return

Model for kerb returns model box available models

model for all the kerb returns

Naming choice box Follow Road Names - LHS driving
Follow Road Names - RHS driving
Use Numbers
Use Letters

method of naming the kerb returns

Just create kerb returns tick box

if ticked, only create the kerb returns. Usually only run at the beginning to see if the kerbs look correct.

Just create kerb returns tick box

if ticked, only create the kerb returns. Usually only run at the beginning to see if the kerbs look correct.

Kerb return strings and sections in one model tick box

if ticked, then all the kerb return strings and sections are place in the one model. The model names are given in the next two fields.

Model for kerb return strings model box available models

model for all the kerb return strings

Model for kerb return sections model box available models

model for all the kerb return sections

Delete frozen Apply MTF functions tick box

if ticked, the Apply MTF functions are deleted for all the frozen roads. Normally not used

View for models view box available views

if non blank, view to add all the created models onto.

Error report file box *.rpt files

file for error messages

Boxing tab

boxing can be created for all the road and kerb return strings. The boxing definitions must have a common centreline name in them which will be replaced by the name of each of the road/kerb return centrelines.

Create boxing tick box

if ticked, boxing is applied for all the road and kerb returns. The boxing definitions must have a common centreline name in them which will be replaced by the name of each of the road centrelines or kerb returns.

Default boxing file file box *.blf files

file of boxing file definitions

Default boxing CL delimiter input

common name in each boxing definition which will be replaced by the name of each road/kerb return centreline

Model stem for boxing strings input

*pre*post text to use with the road/kerb return function name for boxing string models*

Boxing sections input

*pre*post text to use with the road/kerb return function name for boxing section models*

Boxing kerb return sections and strings in one model tick box

if ticked, the strings for all the kerb return Applies are place in one model rather than one model for each kerb return. Similarly, the sections for all the kerb return Applies are placed in the one model. If not ticked, the model names are created in the same way as for the road centrelines. That is, the Model stem for boxing strings and sections are used with the kerb return function name and the layer stems to produce the kerb return boxing strings and sections models.

Model name for boxing kerb strings input

*when only one model (per layer) for the box kerb strings is used - the base name for the model of kerb strings created for each boxing layer. The pre*post text in the Layer Stem is applied to this name.*

Model name for boxing kerb sections input

*when only one model (per layer) for the box kerb sections is used - the base name for the model of kerb sections created for each boxing layer. The pre*post text in the Layer Stem is applied to this name.*

Boxing Layer 1-8 in the grid:

if non blank, boxing strings and sections will be created for that layer.

The last boxing layer is also referred to as the ***subgrade*** *layer.*

Layer stem input

*for this boxing layer, the pre*post text used when naming models for boxing sections and strings.*

Road - LHS input

for this boxing layer, the boxing definition to use on the left hand side of road centrelines

Road - RHS input

for this boxing layer, the boxing definition to use on the right hand side of road centrelines

Kerb return input

for this boxing layer, the boxing definition to use on the kerb returns

Intersection - LHS input

for this boxing layer, the boxing definition to use on the left hand side of the road though an intersection. The strings and sections are place in the models for the road centreline going through the intersection.

Intersection - RHS input

for this boxing layer, the boxing definition to use on the right hand side of the road though an intersection. The strings and sections are place in the models for the road centreline going through the intersection.

Tins tab

Create tin tick box

if ticked, create a tin for all the roads, kerb returns and culdesacs

Road tin tin box available tins

name for the tin created from the all the strings and sections

Road tin colour colour box available colours

colour of the road tin

Subgrade tin tin box available tins

if non blank, a subgrade tin of this name is created

Subgrade tin colour colour box available colours

colour of the subgrade tin

Nulling angle input

angle for nulling the triangles

Nulling length input

length for nulling the triangles

Null Seed points:

Seed X/Y input

X/Y coordinate of a nulling seed point

Seed Point button

select a seed point for nulling the tins. The X and Y coordinates are written to the Seed X/Y grid

Visualisation tab

Create visualisation tick box

if ticked, the road tin is processed for visualisation

Apply texture map to tin tick box

if ticked, the texture map is applied to the road tin

Draw road tin as solid on view tick box

Tin colour to use for cut polygons colour box available colours

colour to use for all the cut polygons

Tin colour to use for fill polygons colour box available colours

colour to use for all the fill polygons

Tin colour to use for transition polygons colour box available colours

colour to use for all the polygons in neither cut nor fill

Freezing tab

Colour for the frozen sections colour box available colours

colour to use for the sections of frozen strings. In any view, this gives a quick way of identifying frozen roads.

Road grid - list of all the centrelines

Element available colours

centreline name

Frozen tick box

if ticked, the centreline is frozen out of the automatic Create Roads process.

*For a frozen centreline, when the **Create Roads** function is run, the Apply MTF for that centreline is deleted and re-created. However any existing Apply MTF for that centreline will be run. Kerb returns are still calculated using the lip line create by running the existing Apply MTF for the frozen centreline. Hence the mtf for a frozen centreline can be modified and the changes will **not** be deleted when the Create Roads function is recalced.*

Note - if the two roads used to define a kerb return are both frozen, then the kerb return is automatically frozen and the kerb return is not deleted and recreated when the Create Roads function is recalced. However the existing Apply MTF for the frozen kerb return is run on each recalc.

Buttons at bottom

Create button

run the option and create the road network.

Continue to 20.8.1.2.2 Create Roads - Setup or return to 20.8.1.2 V10 Create Roads.

20.8.1.2.2 Create Roads - Setup

Position of option on menu: **Design =>Roads =>Create =>Create Roads - Setup**

In the **Create Roads - Setup** panel there are default values for the road width and road crossfall and turn radius.

If a different road width or crossfall, or a different left turn radius or right hand radius is required, the **Create Roads - Setup** option is used to define the different information for any centreline. **Create Roads - Setup** is also used to define culdesacs.

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Road Setup tab

Centreline string select

used to set the left and right turn radii, and the width of the road. This option is used twice to set the different radii for each end of a road by picking the centreline close to the end that is to have the left and right turn radii defined. The left and right turn radii can be different for each end of an alignment.

Road width

width of one side of the road for the selected centreline.

LHS turn radius

radius to use for filleting between any centrelines on the left.

RHS turn radius

radius to use for filleting between any centrelines on the right.

Set button

store the given values in the panel with the selected centreline.

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Culdesac Setup tab

Left fillet input

radius of the fillet between the road and the bulb on the left hand side

Bulb radius input

radius of the culdesac bulb

Bulb offset input

offset of the culdesac bulb

Right fillet input

radius of the fillet between the bulb and the right had side of the road

Left straight input

distance between the left fillet and the bulb

Right straight input

distance between the right fillet and the bulb

Set button

store the panel values the selected centreline.

Continue to 20.8.1.2.3 Create Roads - Kreturns Convert or return to 20.8.1.2 V10 Create Roads.

20.8.1.2.3 Create Roads - Kreturns Convert

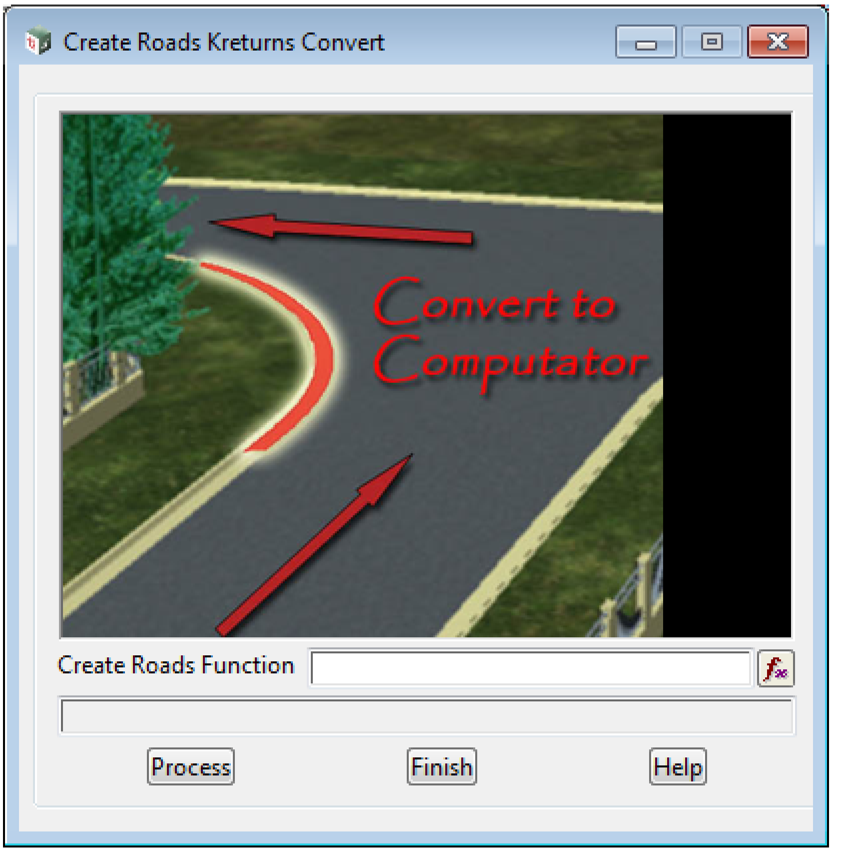
Position of option on menu: **Design =>Roads =>Create =>Kret Convert**

This macro is to be used in conjunction with the **Create Roads.**

Once **Create Roads** has been run, this macro will convert all the SA kerb returns to Element method (in both horizontal and vertical).

The horizontal parts are referenced to the appropriate road centreline, using segment computators and a free arc radius.

The vertical is a combination of a fixed grade at the start and end, with a compound parabola.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Create Roads Function	function		select function
select function from list			
Process	Button		
runs the option			

Return to 20.8.1.2 V10 Create Roads.

20.8.2 Design Checker

Position of option on menu: **Design =>Roads =>Design checker**

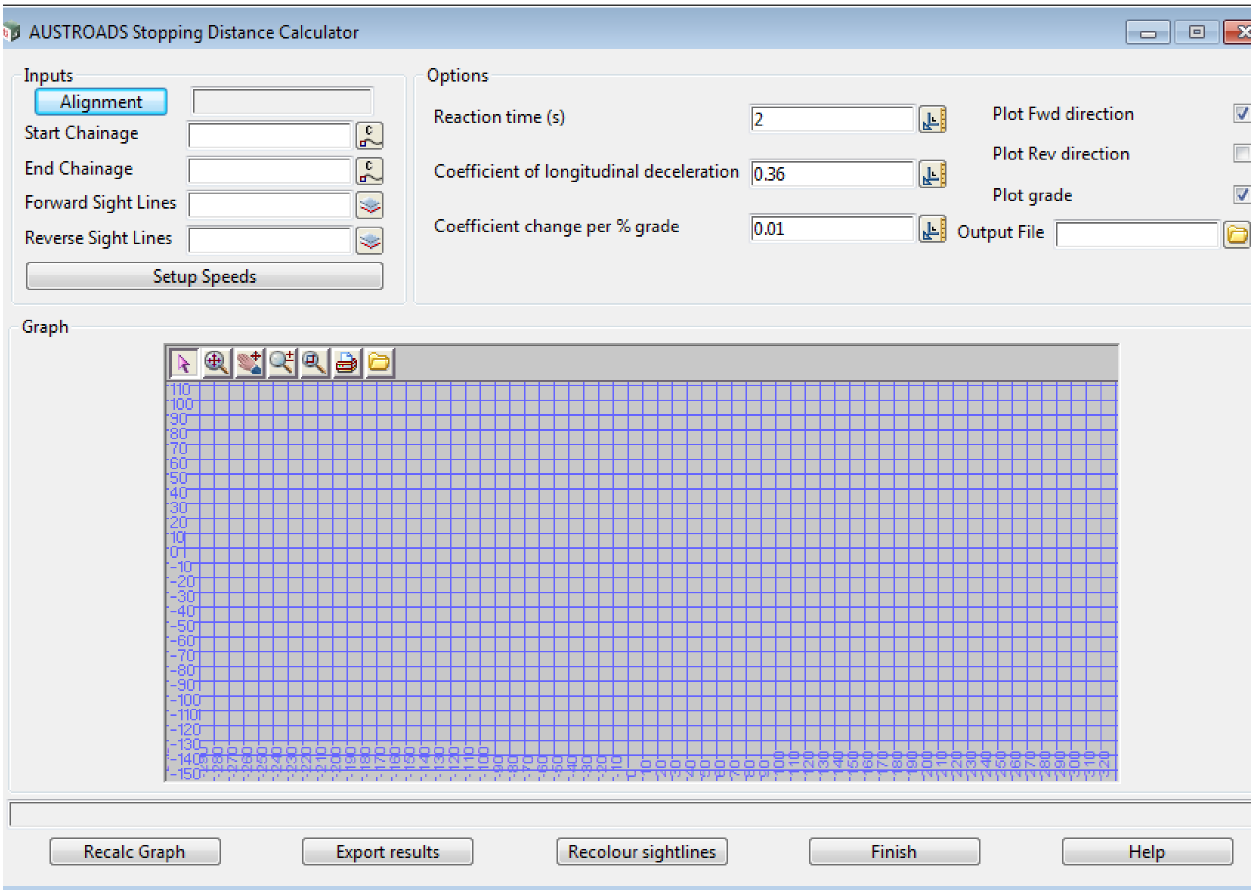
This option is still under development.

Selecting **Design checker** displays the **Design Checker** panel.

20.8.3 Stopping Distance

Position of option on menu: Design =>Roads =>Stopping distance

This section of documentation is a work in progress and will be updated in subsequent releases.

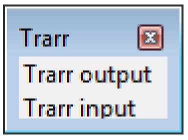


20.8.4 Trarr

Position of menu: **Design => Roads =>Trarr**

The option under **Trarr** read and write Design TRARR files.

The **Trarr** walk-right menu is



write out a Trarr file
read in a Trarr file

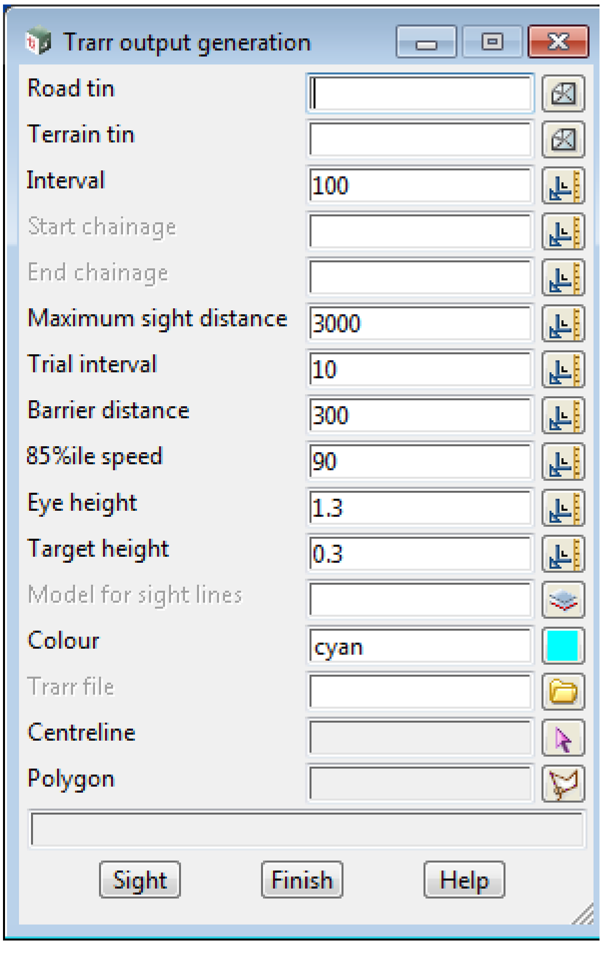
For *Trarr output*, go to 20.8.4.1 Trarr Output
Trarr input 20.8.4.2 Trarr input

20.8.4.1 Trarr Output

Position of option on menu: **Design =>Roads =>Trarr =>Trarr output**

Options to create a file for use with TRARR.

Please contact 12d Solutions Pty Ltd if you wish to use this option.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Road tin	tin box		available tins

tin of the road design

Terrain tin	tin box		available tins

tin of the terrain

Interval input

interval to calculate sight lines

Start chainage

*if **non blank**, start chainage for producing the Trarr file.*
*if **blank**, start at the beginning of the centreline string.*

End chainage

*if **non blank**, end chainage for producing the Trarr file.*
*if **blank**, go to the end of the centreline string.*

Maximum sight distance input

maximum distance to try and calculate sight lines

Trial interval input

interval to move ahead to test sight distance to

Barrier distance input

if sight distance is less than this value then a barrier line is required

85%le speed input

85 percentile speed (to write to the Trarr file)

Eye height input 1.3

height of the eye point above the picked string

Target height input 0.3

height of the target point above the picked string

Model for sight lines	model box		available models

if non blank, the sight lines are kept and placed in this model

Colour	colour box		available colours

colour of the sight lines

Trarr file	file box		*.rpt files

if non blank, a Trarr file of this name is produced

Centreline string-select

string to be used for placing the eye and target points and calculating sight distance, is selected from a view.

Polygon string-select

boundary polygon for the road tin

Sight button

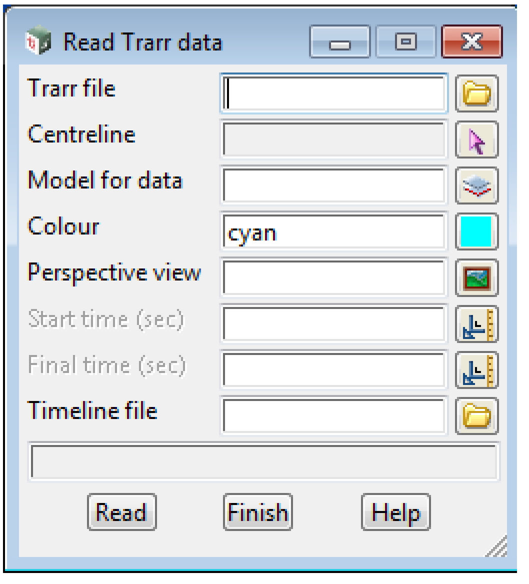
calculate sight lines and write out a Trarr file

20.8.4.2 Trarr input

Position of option on menu: **Design =>Roads =>Trarr =>Trarr input**

Option to read a TRARR file.

Please contact 12d Solutions Pty Ltd if you wish to use this option.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

20.8.5 Vehicle Path

Position of menu: **Design =>Roads =>Vehicle Path**

The **vehicle path** option is used to interface with the vehicle path program **Vpath**, written by the Queensland Department of Transport and Main Roads.**Vpath** can be obtained from the Queensland Department of Transport and Main Roads.

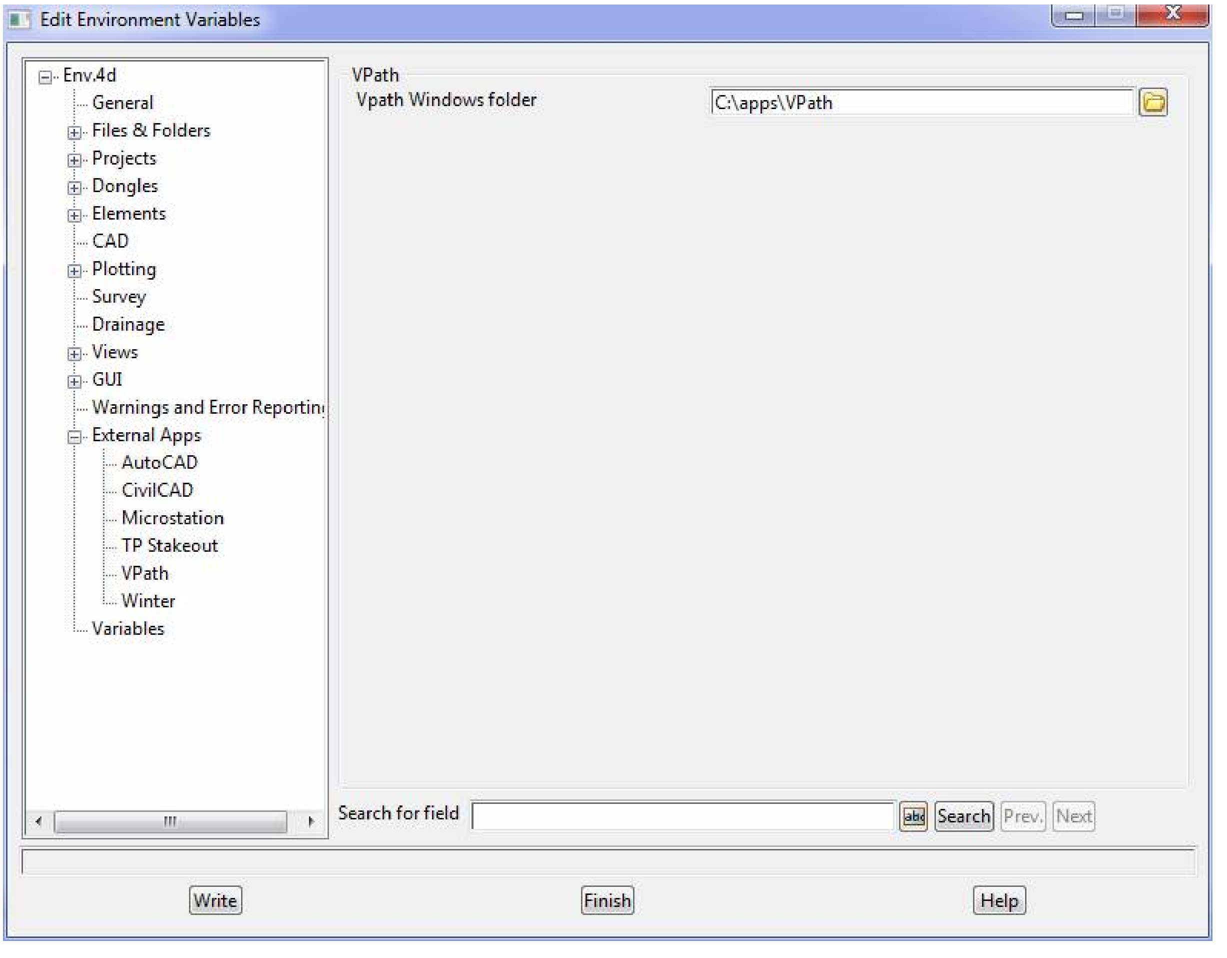
The **Vpath** program is fully described in its own Manual available from Queensland Department of Transport and Main Roads.

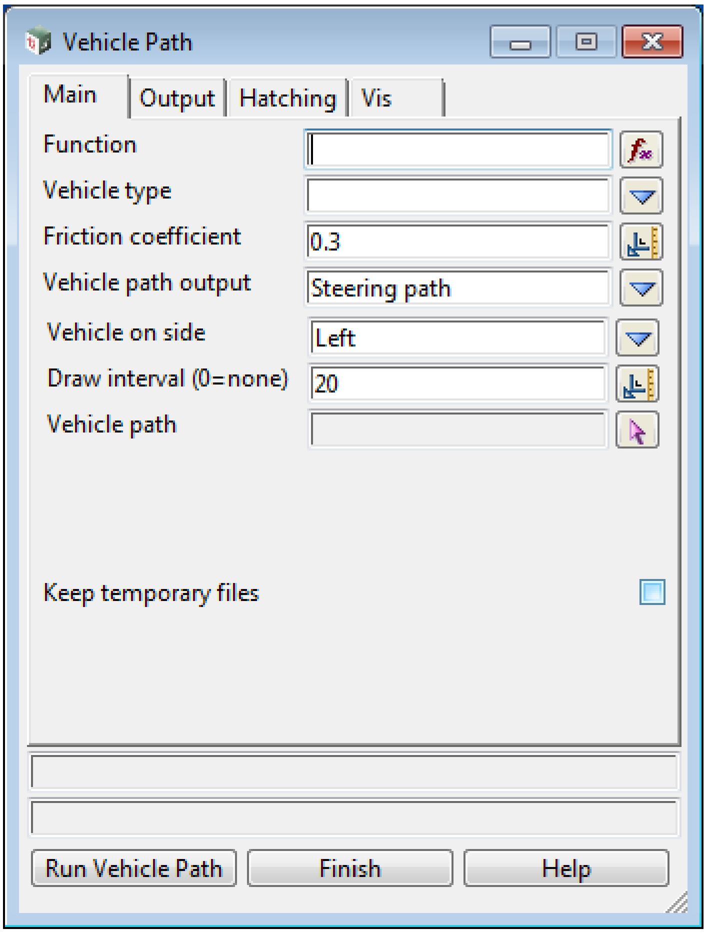
To run **Vpath** from within 12d Model the environment variable **WINDOWS_VEHICLE_PATH_4D** is set to point to the folder where the Windows **Vpath** executable is located.

This can be set manually or done through the Edit Environment Variables panel.

WINDOWS_VEHICLE_PATH_4D *folder-path*

On selecting the **Vehicle Path** option, the **Vehicle Path** panel is displayed. If the environment variable is not set the **Edit Environment Variables** panel will be displayed on the screen.





The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Main Tab

Function function box

*name of the **Steering Path** function. If the function already exists and is picked from a pop-up or an <enter> is given at the end of the name, the information from the existing function will be placed in the appropriate panel fields*

Vehicle type choice box

the vehicle type used for generating the swept path. This is populated from first the vehicles.std and then the vehicles.cus files located in the defined vehicle path directory.

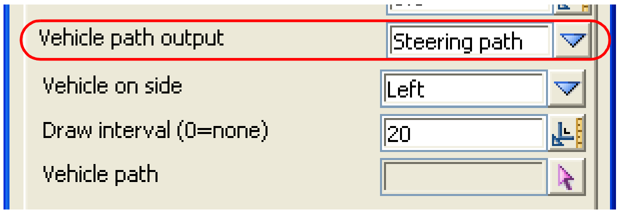
Friction coefficient input

the coefficient of friction to be used for the vehicle.

Vehicle Path Output choice box Steering path, Turning Template

Steering path: the swept path of the vehicle is generated along the selected super alignment/alignment string.

Turning Template: the swept path of the vehicle is generated through 180° at a nominated radius



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Steering path options

Vehicle on side choice box — Pop-Up: Left, Right

the side of the vehicle that the selecting steering path represents

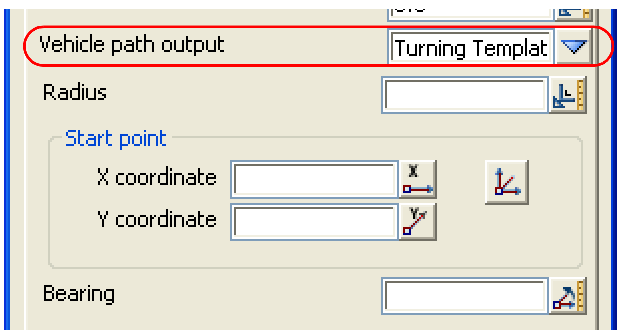
Draw interval (0=none) input

if non-zero, the position of the vehicle on the steering path is, draw the position at this interval

Vehicle Path string select

select the string for generating the swept path.
The string must be a super alignment or alignment string, consecutive straights and transitions curves are not allowed and there can be no more than 20 arcs in the string.

Turning Template Options



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Radius input

enter the radius of the turning template, positive defines a clockwise turn, negative is anti-clockwise.

Start point input

select the coordinates of the start point of the turning template.

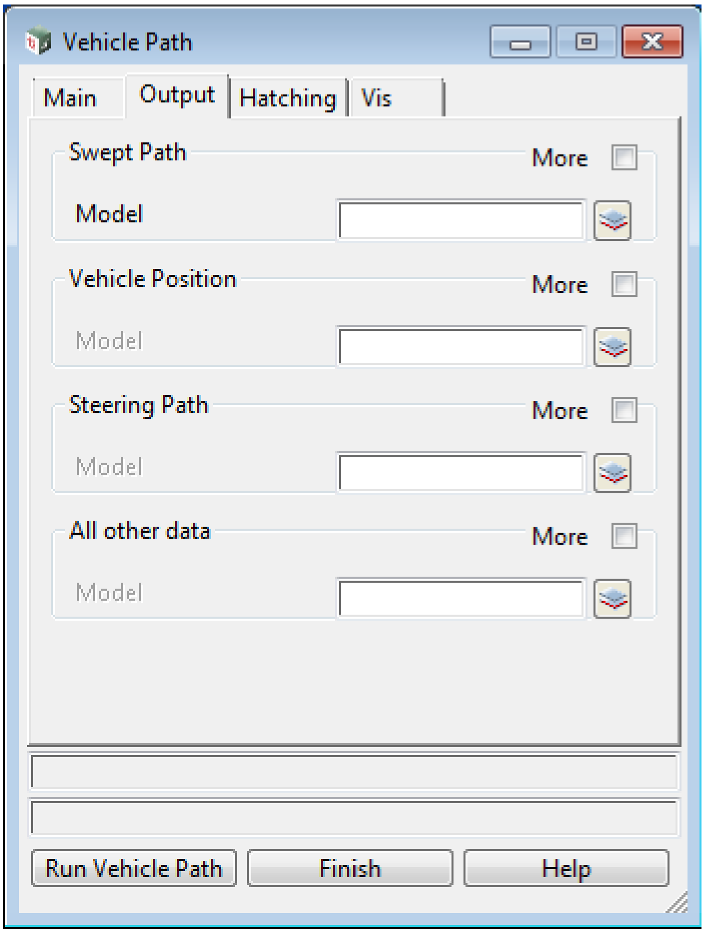
Bearing input

enter the bearing of the start point of the turning template.

Common options

Keep temporary files tick box

*if ticked the input file for **Vpath** and the **dxf** file produced by **Vpath** are kept in the **Vpath** directory.*



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Output Tab

Swept Path

the model for the swept path generated by the extremities of the vehicle

Vehicle Position

if non-blank, the strings tracing out the vehicle position at the nominated interval are placed in this model

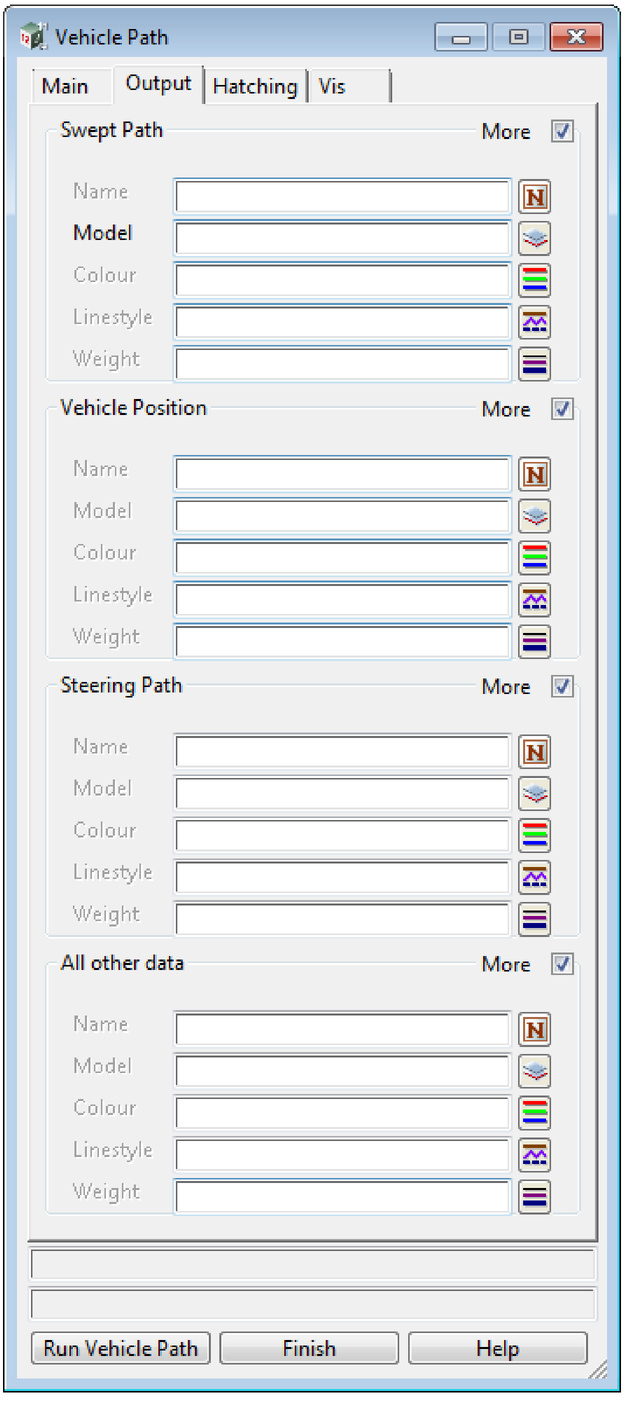
Steering Path

if non-blank, the steering path including the marking of the tangent points and centres of each circular curve are created and placed in this model

All other data

*if non blank all other data produced by **Vpath** including a table of parameters used is placed in this model.*

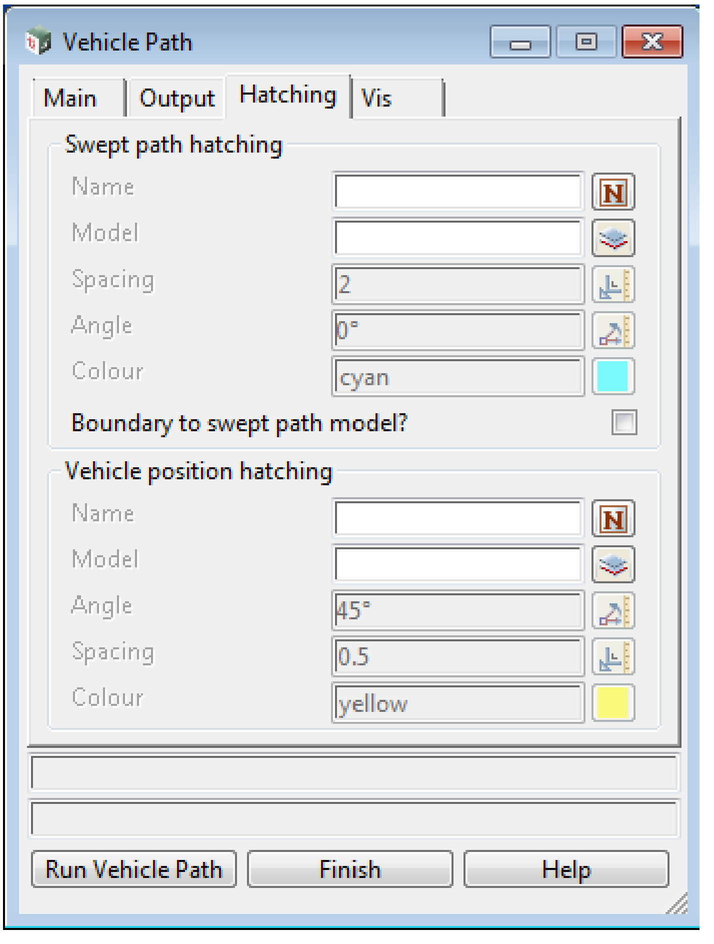
If the **More** tick box is ticked the following fields are displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

With the **More** enabled the **Name**, **Model**, **Colour**, **Linestyle** and **Weight** can be entered for each option.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Hatching Tab

Swept path hatching

The swept path of the vehicle can be hatched, the swept path strings generated by **Vpath** are collated into a single boundary string to produce the hatching.

Warning, this operation can take some time on longer alignments.

Name

name of the swept path hatching.

Model

model for the swept path hatching.

Spacing

spacing of the swept path hatching.

Colour

colour of the swept path hatching.

Boundary to swept path model? tick box

if ticked the collated boundary string is copied to the swept path model.

Vehicle position hatching

the strings tracing out the vehicle position can be hatched.

Name

name of the vehicle position hatching.

Model

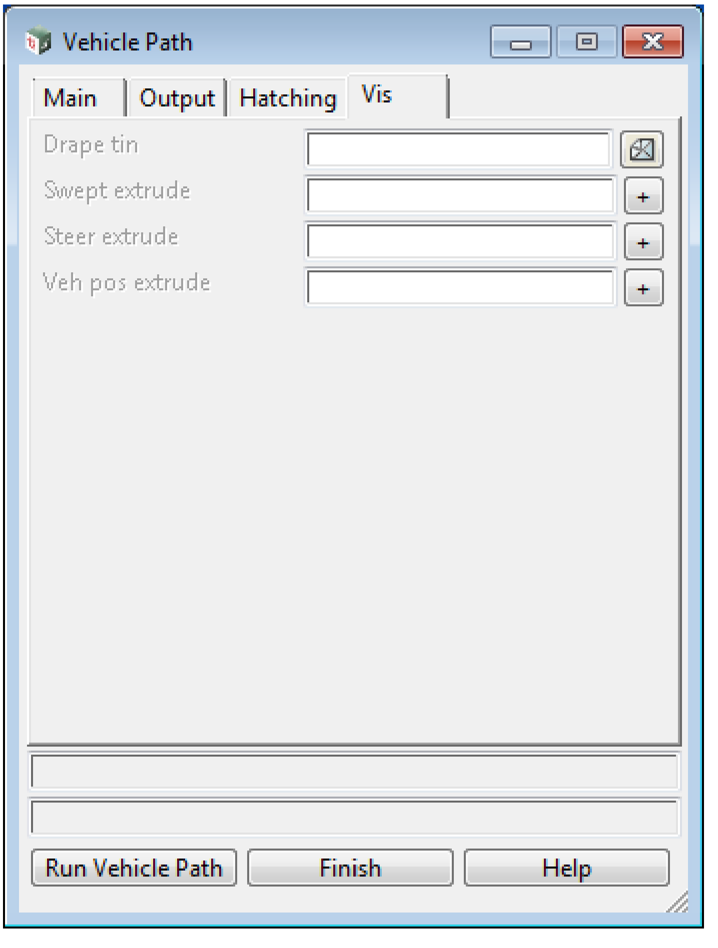
model for the vehicle position hatching.

Spacing

spacing of the vehicle position hatching.

Colour

colour of the vehicle position hatching.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Vis Tab

The **Vis** tab is used to create a visualisation from the generated strings.

Drape tin

the tin to drape the generated string onto to create the visualisation

Swept extrude

the extrusion to use for the swept path strings

Steer extrude

the extrusion to use for the steering path strings

Veh pos extrude

the extrusion to use for the vehicle position strings

Run Vehicle Path button

*run the **Vpath** program and load the results back into the models given in the panel fields.*

***Vehicle Path** supports up to 20 IPs. A warning message will be displayed if this is exceeded and no data will be produced.*

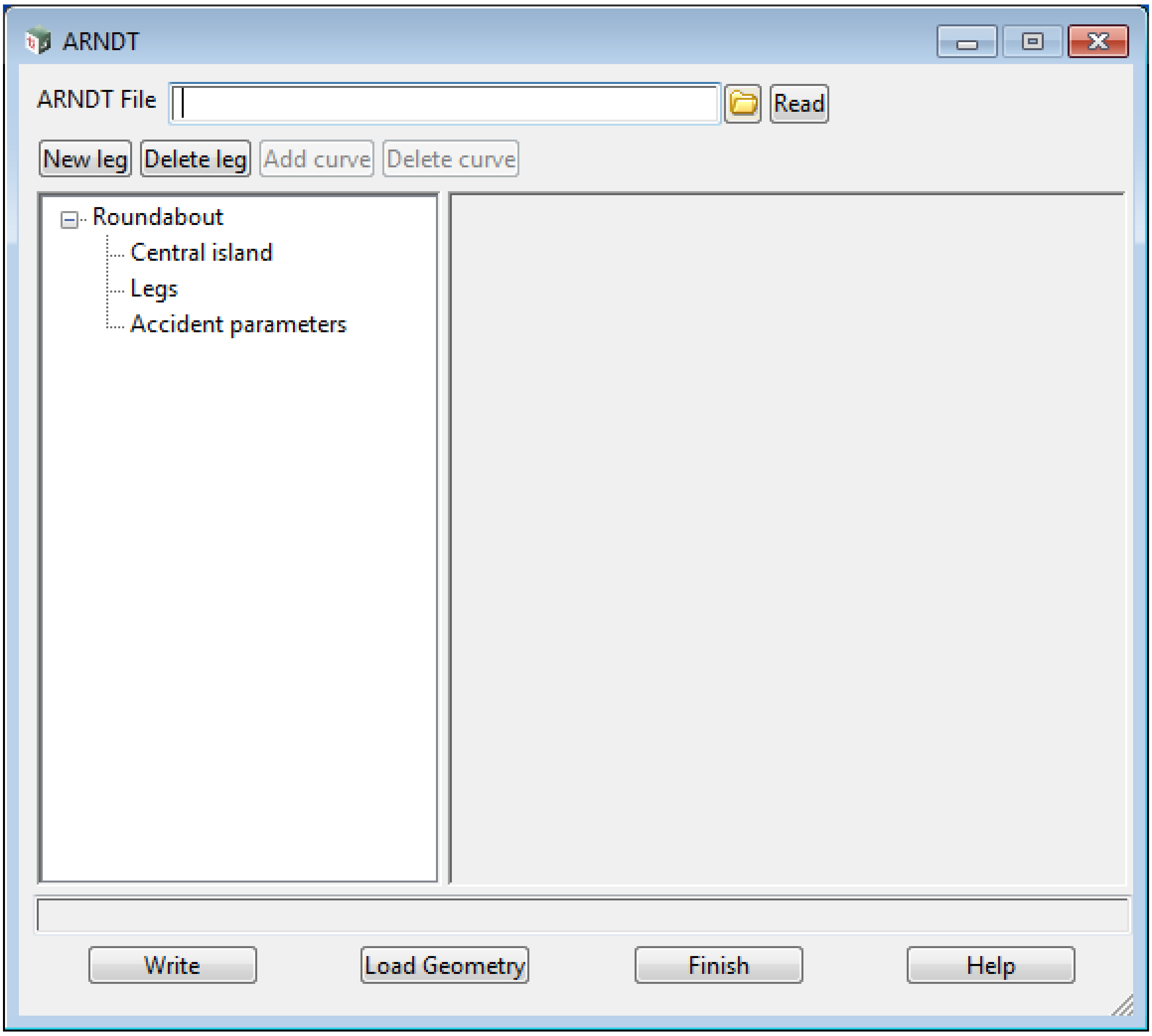
***Vehicle Path** only supports tangential straight/curve and curve/curve geometry. A warning message will be displayed if this criteria is not met and no data will be produced.*

20.8.6 ARNDT Editor

Position of menu: **Design =>Roads =>ARNDT Editor**

This editor allows you to create or edit files for **ARNDT** (**A Roundabout Numerical Design Tool**) files. It can also interface directly with **ARNDT** to create geometry and generate reports.

Selecting **ARNDT Editor** brings up the **ARNDT** panel.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
ARNDT file	file		

*the **ARNDT** file to read or write*

Read	button		

*reads the **ARNDT** file*

New leg	button		

creates a new leg for the roundabout

Delete leg	button		

deletes the currently selected leg

Add curve	button		

adds a new entry curve to the selected leg

Delete curve button

deletes the selected entry curve

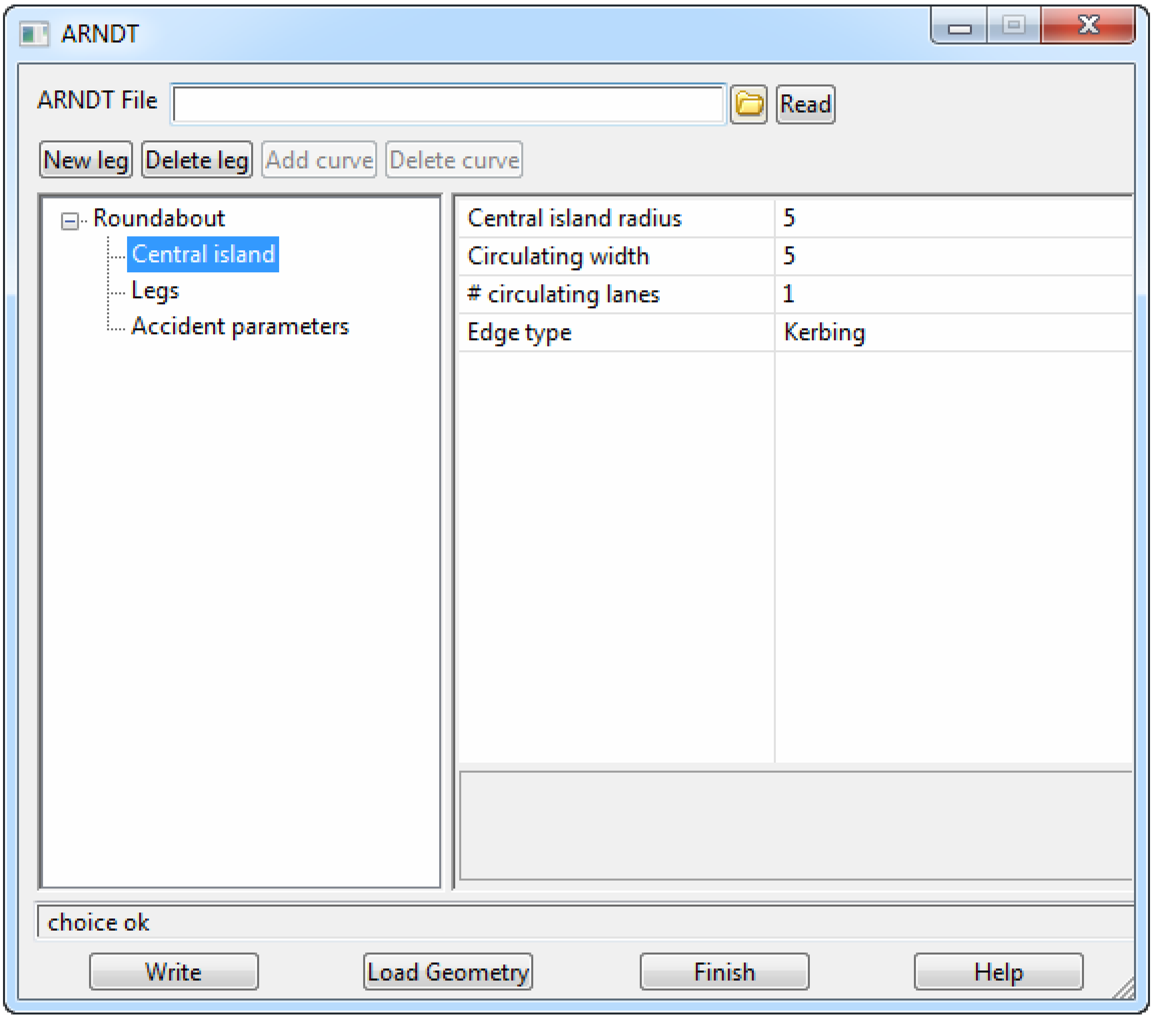
Write button

*writes the **ARNDT** file*

Load geometry button

*options for loading geometry via **ARNDT** and generating report files*

Central Island



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Central island radius

the radius of the central island

Circulating width

the offset from the radius of the circulating string

# of circulating lanes

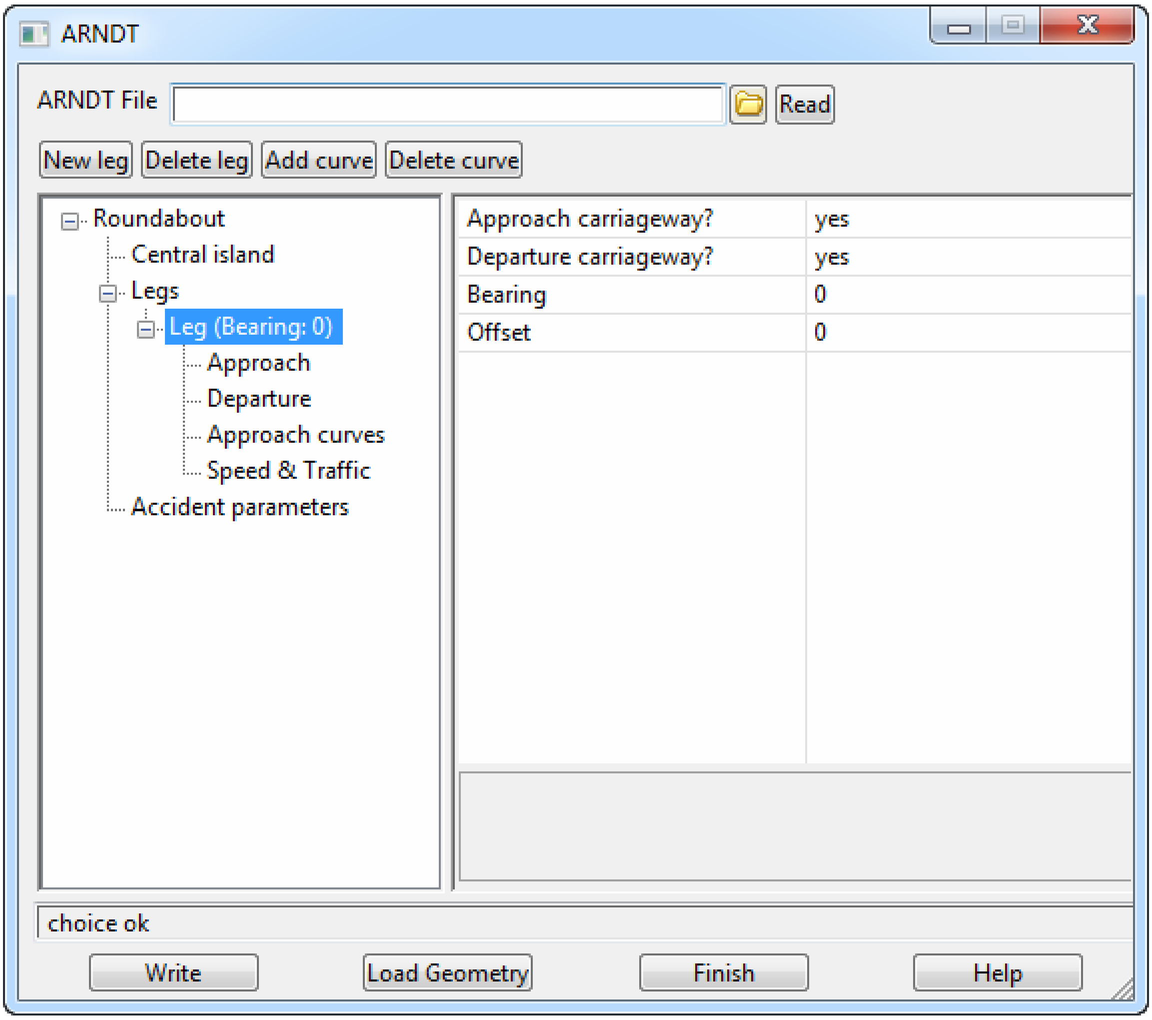
the number of lanes circulating the roundabout

Edge type

the edge type for the central island (Kerbing or Edge Line)

Legs Node

The **Leg node**, for each leg of the roundabout, allows you to modify parameters for that leg.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Approach carriageway?	choice box		yes, no

whether or not there is an approach carriageway (yes or no)

Departure carriageway?	choice box		yes, no

whether or not there is a departure carriageway (yes or no)

Bearing

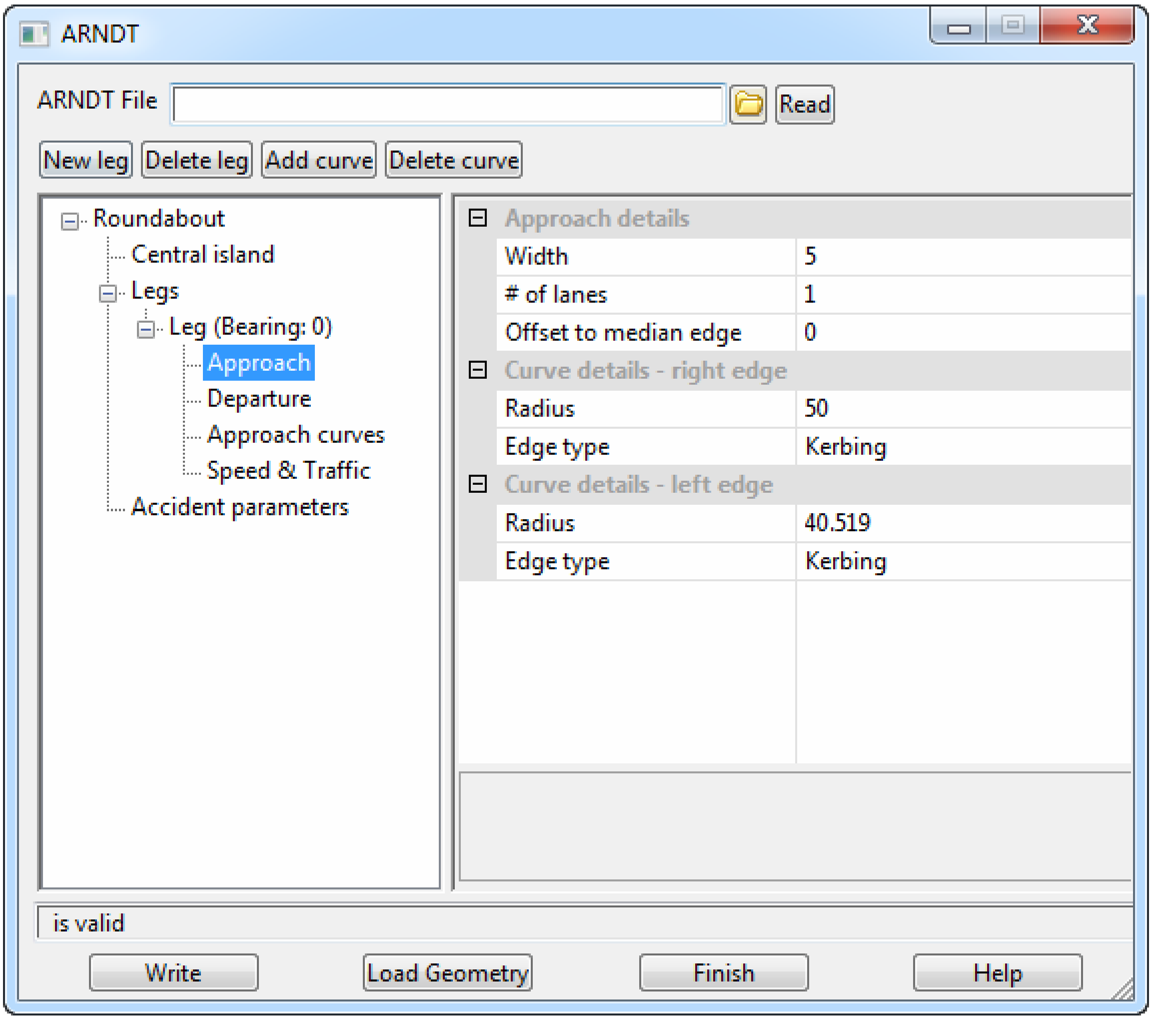
the bearing of the leg

Offset

the offset of the leg from the centre of the roundabout

Leg Approach Node

This defines the properties for the approach carriage of the leg, if there is an approach.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Approach details

Width

the width of the approach carriageway

of lanes

the number of lanes in the approach carriageway

Offset to median edge

the offset of the approach from the median

Curve details - right edge

Radius

the radius of the right edge of the approach

Edge type

the type of edging for the right edge of the approach

Curve details - left edge

Radius

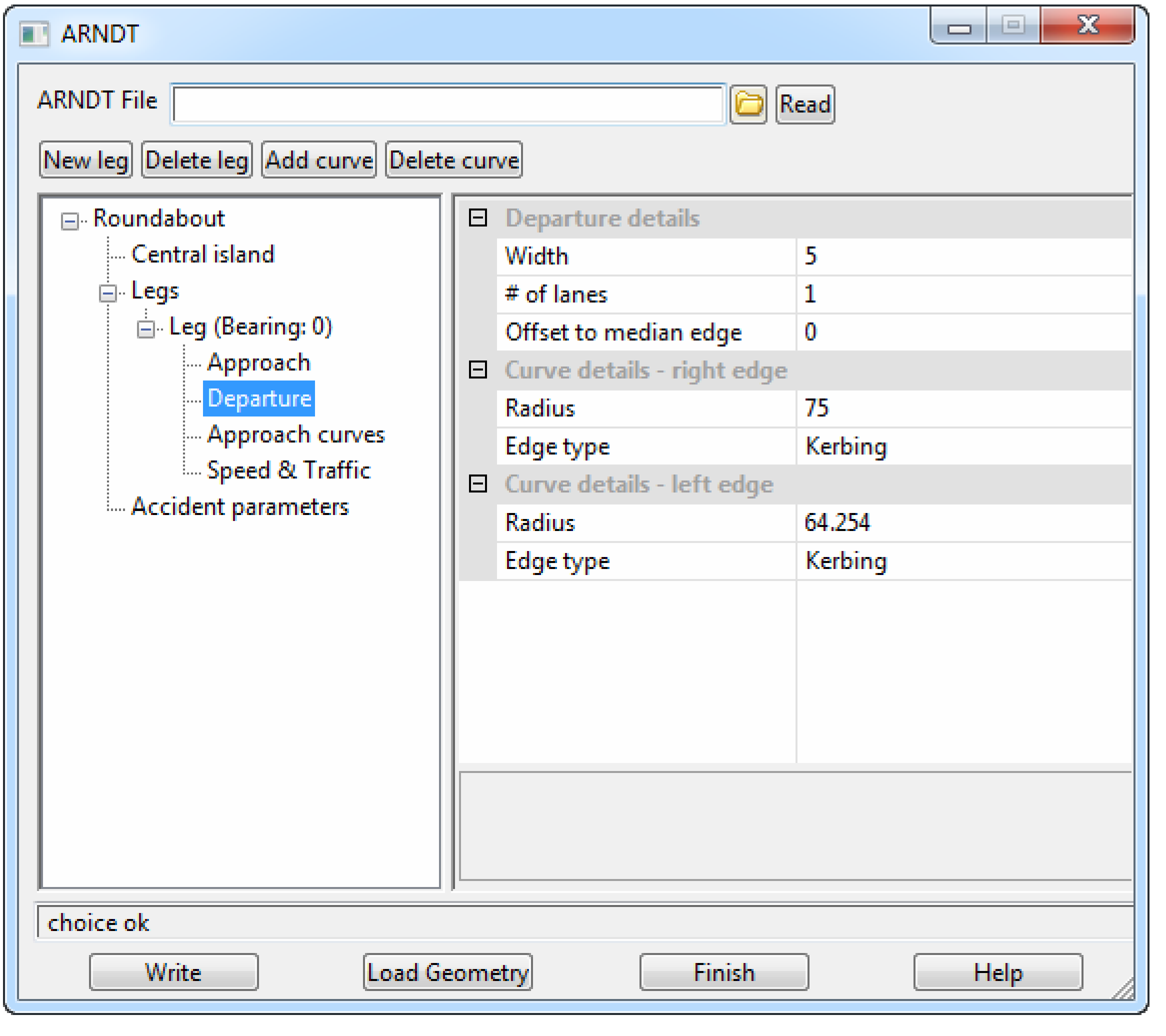
the radius of the left edge of the approach

Edge type

the type of edging for the left edge of the approach

Leg Departure Node

This defines the properties for the departure carriage of the leg, if there is a departure.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Departure details

Width

the width of the departure carriageway

of lanes

the number of lanes in the departure carriageway

Offset to median edge

the offset of the departure from the median

Curve details - right edge

Radius

the radius of the right edge of the departure

Edge type

the type of edging for the right edge of the departure

Curve details - left edge

Radius

the radius of the left edge of the departure

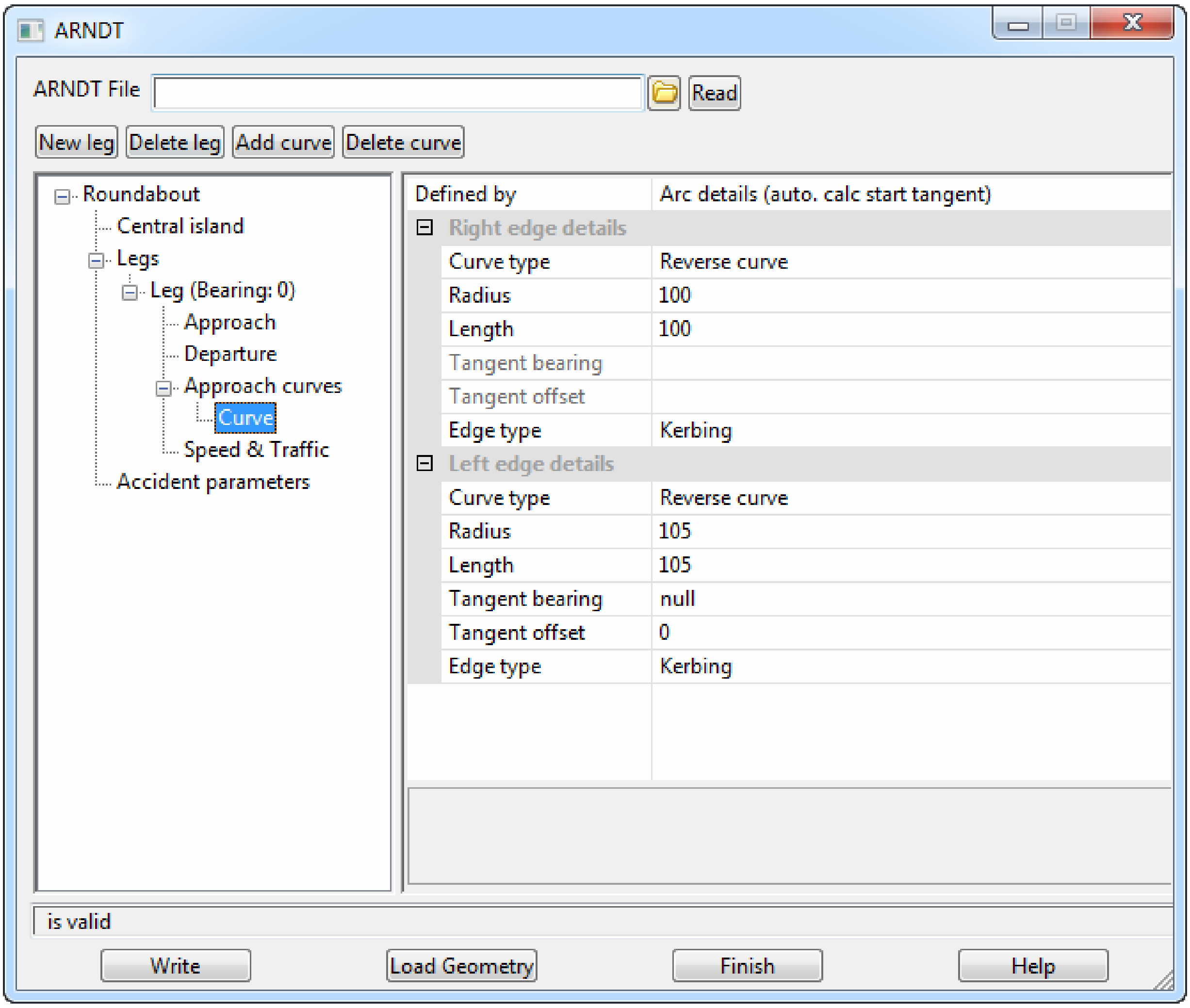
Edge type

the type of edging for the left edge of the departure

Approach Curves Node

This node defines the setting for each approach curve. New approach curves can be added to a leg by clicking the **Add curve** button. There are two methods for defining a curve - either by arc details or start tangent.

Approach curves are only defined by their right hand edge.

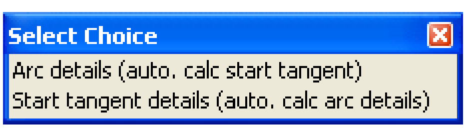


The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Right Edge Details

Defined by choice box



how the curve is defined

Curve type

the type of curve (Reverse or Compound)

Radius

*the radius of the right edge - only available when defined by **Arc details***

Length

*the length of the right edge - only available when defined by **Arc details***

Tangent bearing

*the bearing of the tangent to the curve - only available when defined by **Start tangent details***

Tangent offset

*the offset of the tangent to the roundabout - only available when defined by **Start tangent details***

Edge type

the type of edging for the right edge (Kerbing or Edge Line)

Left Edge Details

Curve type

the type of curve. This field is read only.

Radius

*the radius of the curve, if using **Arc details**. This field is read only.*

Length

*the length of the curve, if using **Arc details**. This field is read only.*

Tangent bearing

*the bearing of the tangent to the curve, if using **Start tangent details**. This field is read only.*

Tangent offset

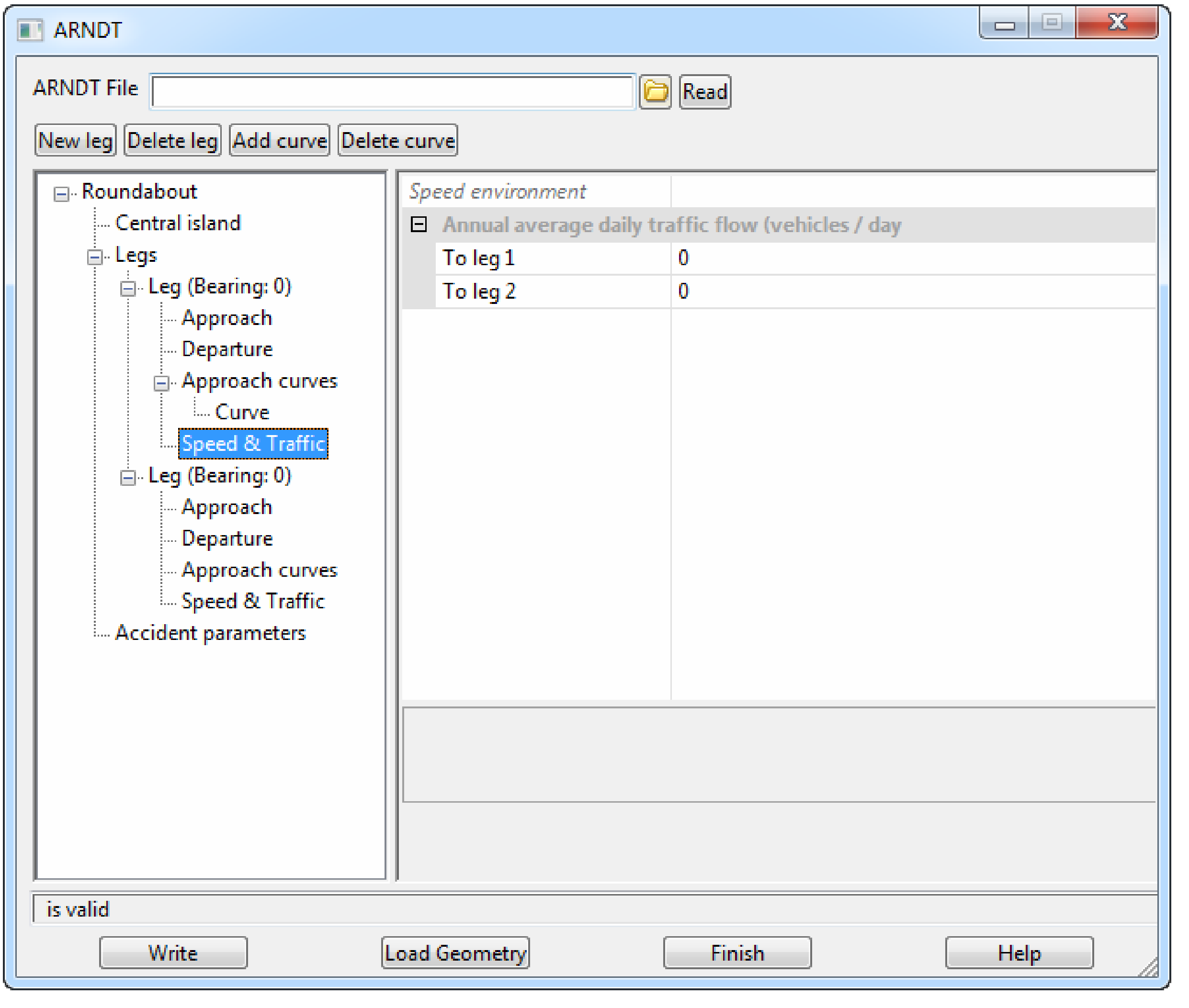
*the offset of the tangent, if using **Start tangent details**. This field is read only.*

Edge type

the type of edging for the right edge (Kerbing or Edge Line)

Leg Speed & Traffic Node

This node defines the speed and traffic details for each leg of the roundabout.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Speed environment

the speed environment of the carriageway

Annual average daily traffic flow

To leg 1

the average daily traffic flow (vehicles per day) leaving leg 1 and returning to leg 1

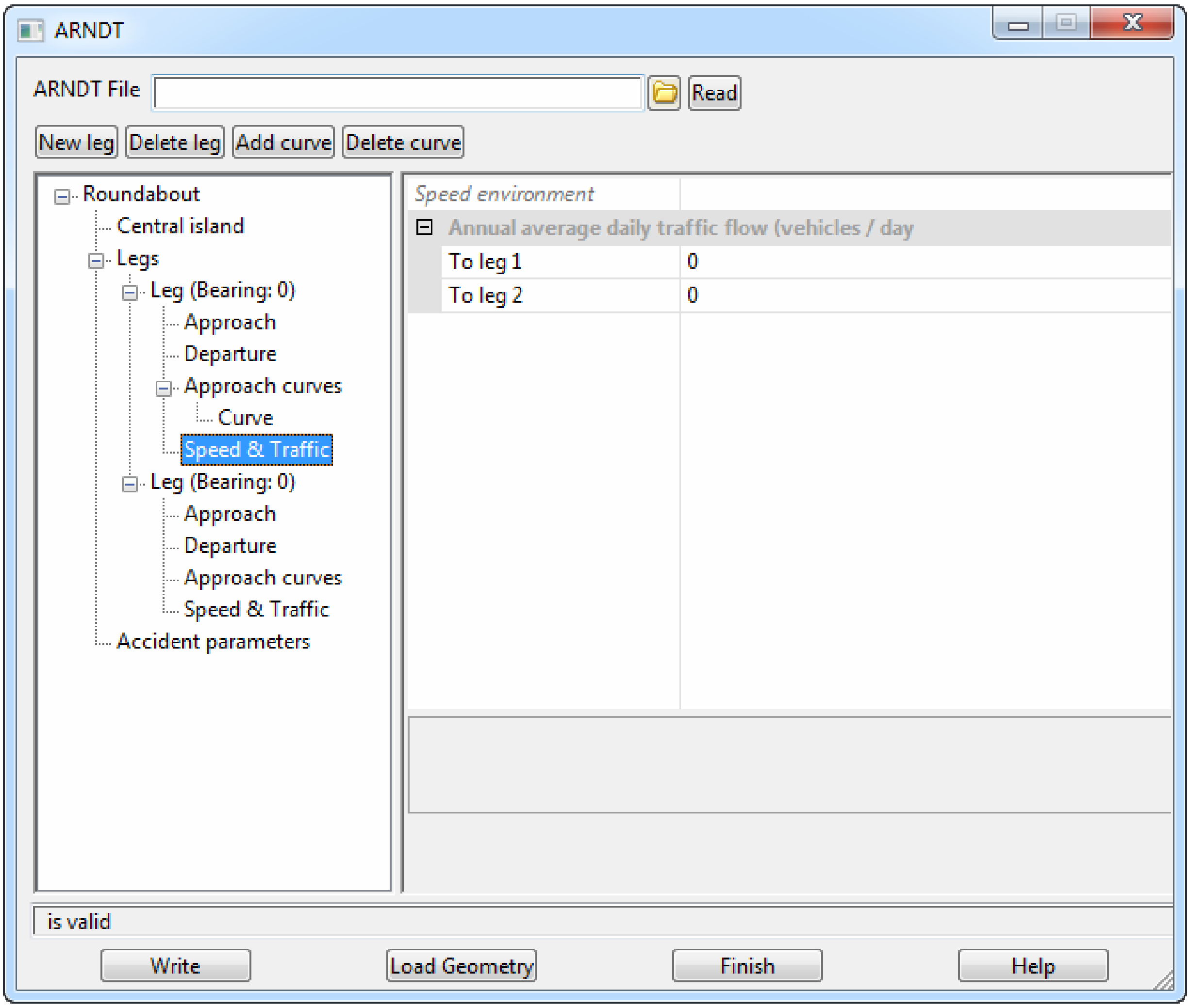
To leg 2

the average daily traffic flow (vehicles per day) leaving leg 1 and exiting via leg 2

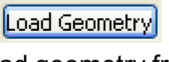
This is repeated for each leg of the roundabout

Accident Parameters

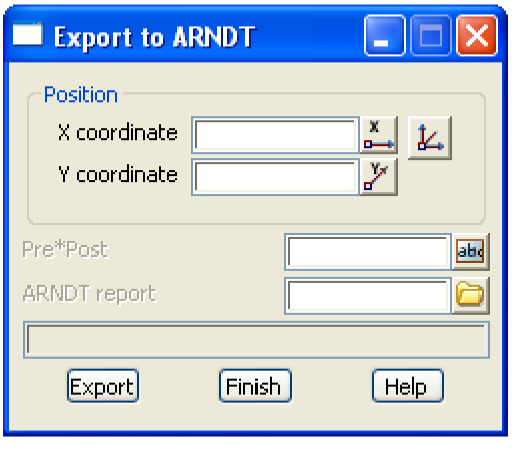
This defines various parameters used by **ARNDT** for calculating risks and costs for the roundabout. These parameters are requested by the **ARNDT** program. Please see the **ARNDT** documentation for more details on these parameters.



Load Geometry / Export to ARNDT

Selecting **Load Geometry**  brings up the **Export to ARNDT** panel.

This panel allows you to load geometry from **ARNDT**, based on the file you are editing, as well as optionally generating reports.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Position

the x and y co-ordinate for the roundabout

Pre*Post

*an optional pre*post applied to any strings read back from **ARNDT***

ARNDT report

*an optional report file of details generated by **ARNDT***

Export

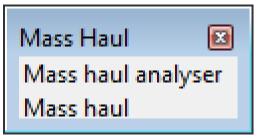
*Exports the current roundabout details to **ARNDT**, imports the results generated by **ARNDT.** Results will include geometry at the nominated position, and a report if specified.*

20.8.7 Mass Haul

Position of menu: **Design =>Roads =>Mass haul**

The option under **Mass haul** reads a volumes report and creates Mass Haul information.

The **Mass haul** walk-right menu is



mass haul analysis from volumes report
create mass haul string and report from volumes report

For *Mass haul analyser*, go to 20.8.7.1 Mass Haul Analyser
Mass haul 20.8.8.3 Mass Haul String and Report

20.8.7.1 Mass Haul Analyser

Position of option on menu: **Design =>Roads =>Mass haul =>Mass haul analyser**

The **Mass Haul Analyser** option calculates the mass haul along an alignment using the volume report files from various 12d functions.

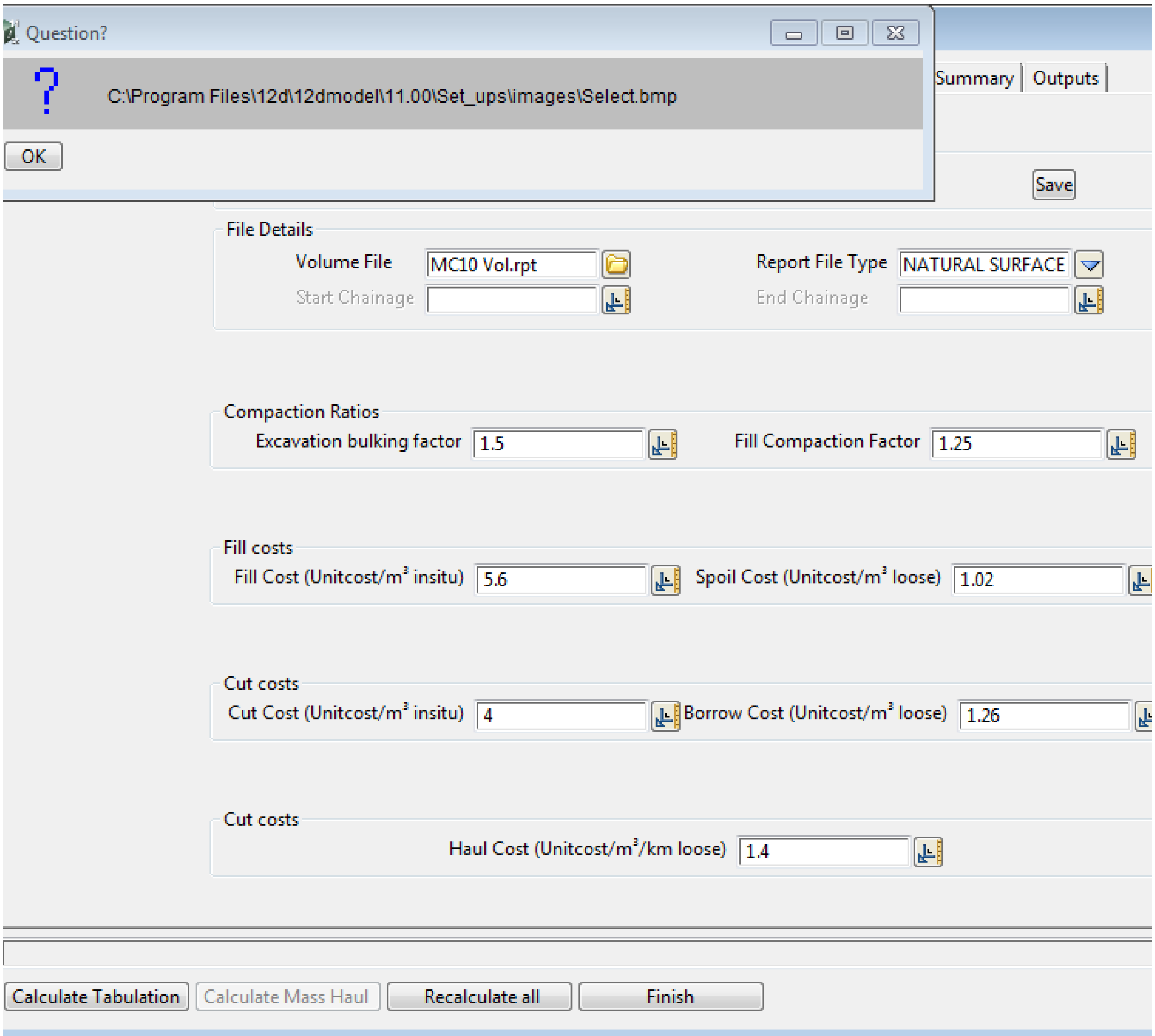
To operate the option a volume report file is selected and the report file type set to match the report type.The **Calculate tabulation** button will then calculate the bulking factors and add the special chainage items to the tabulation. After this the **Calculate Mass Haul** option will push excess fill material forwards along the alignment to a matching cut item with the same accumulated balance and push excess cut material backwards along the alignment to a matching fill item.

Where the cost of Haul cost is greater than the Borrow Plus Spoil cost the option will skip that material.

The option performs this operation several times passing back over the alignment until no more haul material can be found. It then borrows and spoils the remaining material.

The macro is intended to provide an analysis methodology to the designer to enable them to quickly review where material is available in an alignment in order to optimise the alignment with respect to potential cost savings. It is also a useful tool for training in the techniques of mass haul. The final mass haul graph should be exported for detailed analysis using the **Design=>Roads=>Mass haul =>Mass Haul** option.

Selecting the **Mass haul analyser** option brings up the **Mass Haul Analyser** panel.



Configuration Tab

This tab is used to set up the parameters to be used by the macro.

The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Parameter File	file		*.mhp files

name of the file containing the previously used settings for the fields in the panel.

Read Button	button		

read the parameter file in

Write button	button		

write the setting in the panel out to a parameter file.

Volume File	file		*.rpt files

input file generated from another 12d option

Report File Type	choice box	NATURAL SURFACE TO DESIGN VOLUMES see 20.8.8.1 Selecting The Right Report	

File Type

the report file type to be used as an input to the macro

Excavation bulking factor real box 1.5

bulking factor ratio between loose excavated material and insitu compacted material.
number greater than one indicates loose material is bulkier than insitu material.
defaults are arbitrary and must be selected by the user

Fill compaction factor real box 1.25

compaction factor ratio between loose excavated material and insitu compacted material.
number greater than one indicates loose material is bulkier than insitu material.
defaults are arbitrary and must be selected by the user

Fill Cost real box 5.6

cost per cubic metre to place and compact material.
defaults are arbitrary and must be selected by the user

Spoil cost real box 1.02

cost per cubic metre to dispose of excess material (including haul costs) defaults are arbitrary and must be selected by the user

Cut cost real box 4

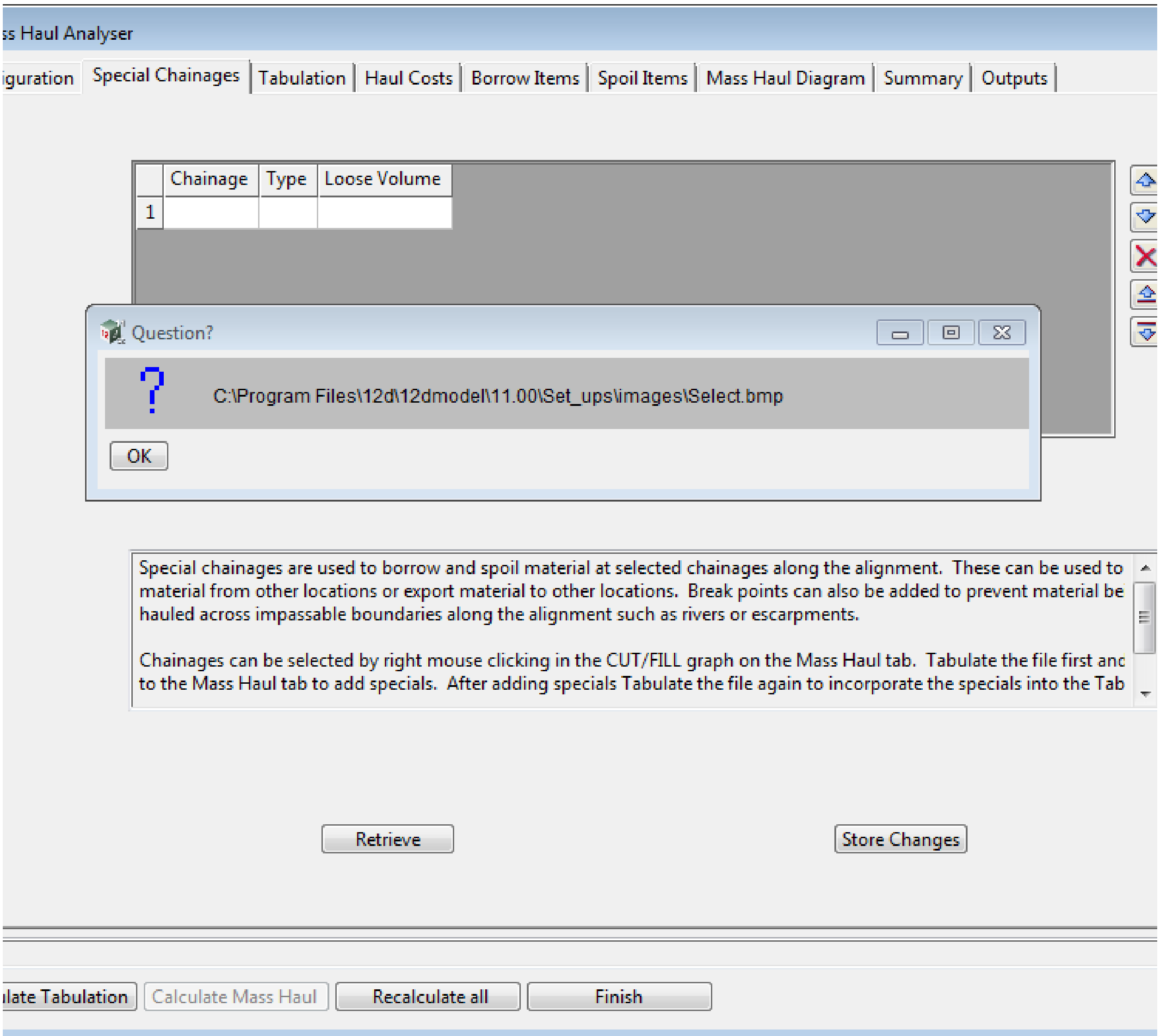
cost per cubic metre to excavate material.
defaults are arbitrary and must be selected by the user

Borrow cost real box 1.26

cost per cubic metre to import additional material (including delivery costs) defaults are arbitrary and must be selected by the user

Haul cost real box 1.4

cost per cubic metre to haul material one kilometre.
defaults are arbitrary and must be selected by the user



Special Chainages Tab

This tab is used to add additional items to the tabulation.

The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Chainage	chainage box		

the chainage that the special item is to be applied

Type	choice box		borrow, spoil, break

the type of special item to be applied

Loose Volume

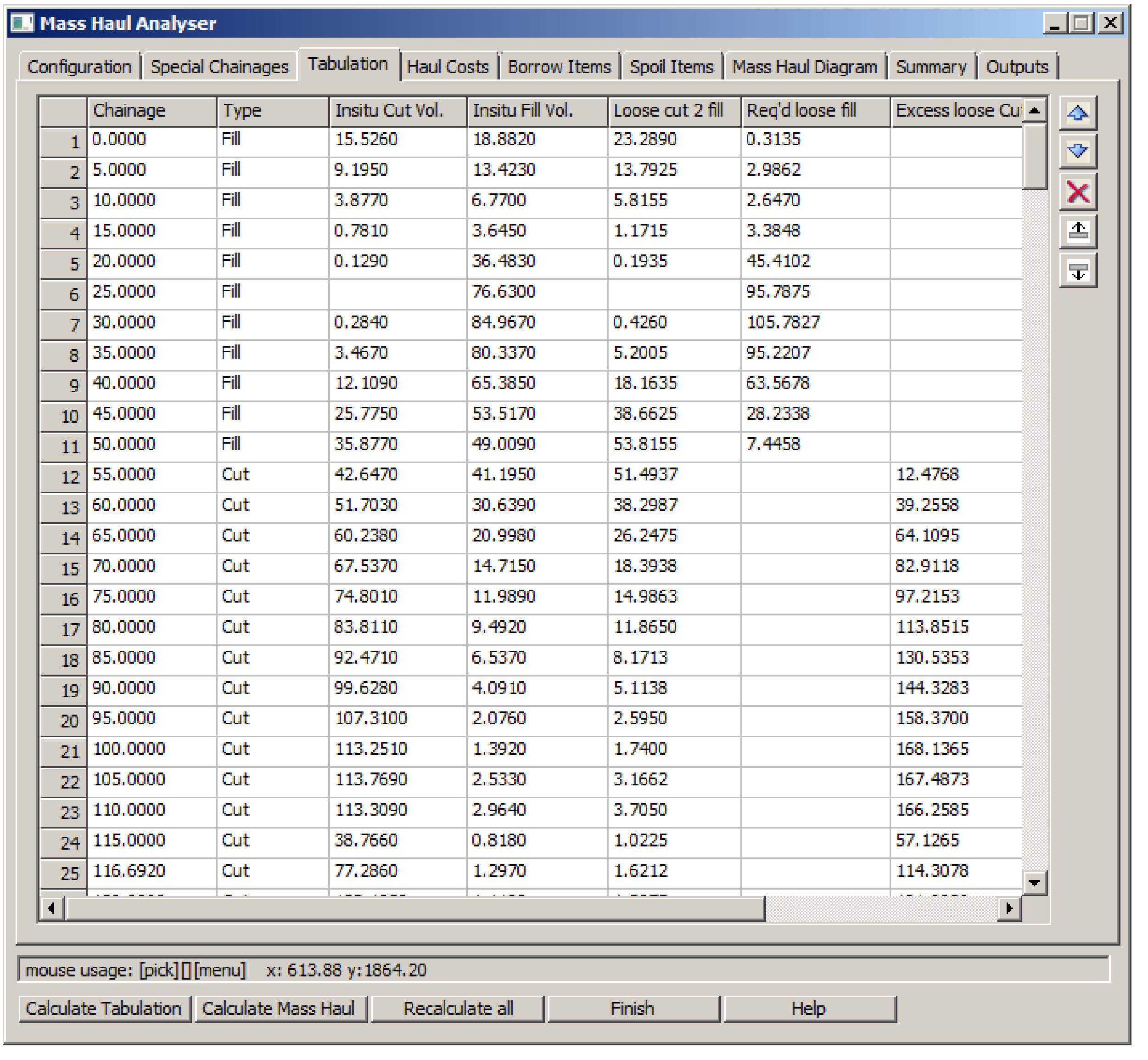
the volume of loose material in the special item

Retrieve button

retrieve special chainages into the grid control

Store Chainages button

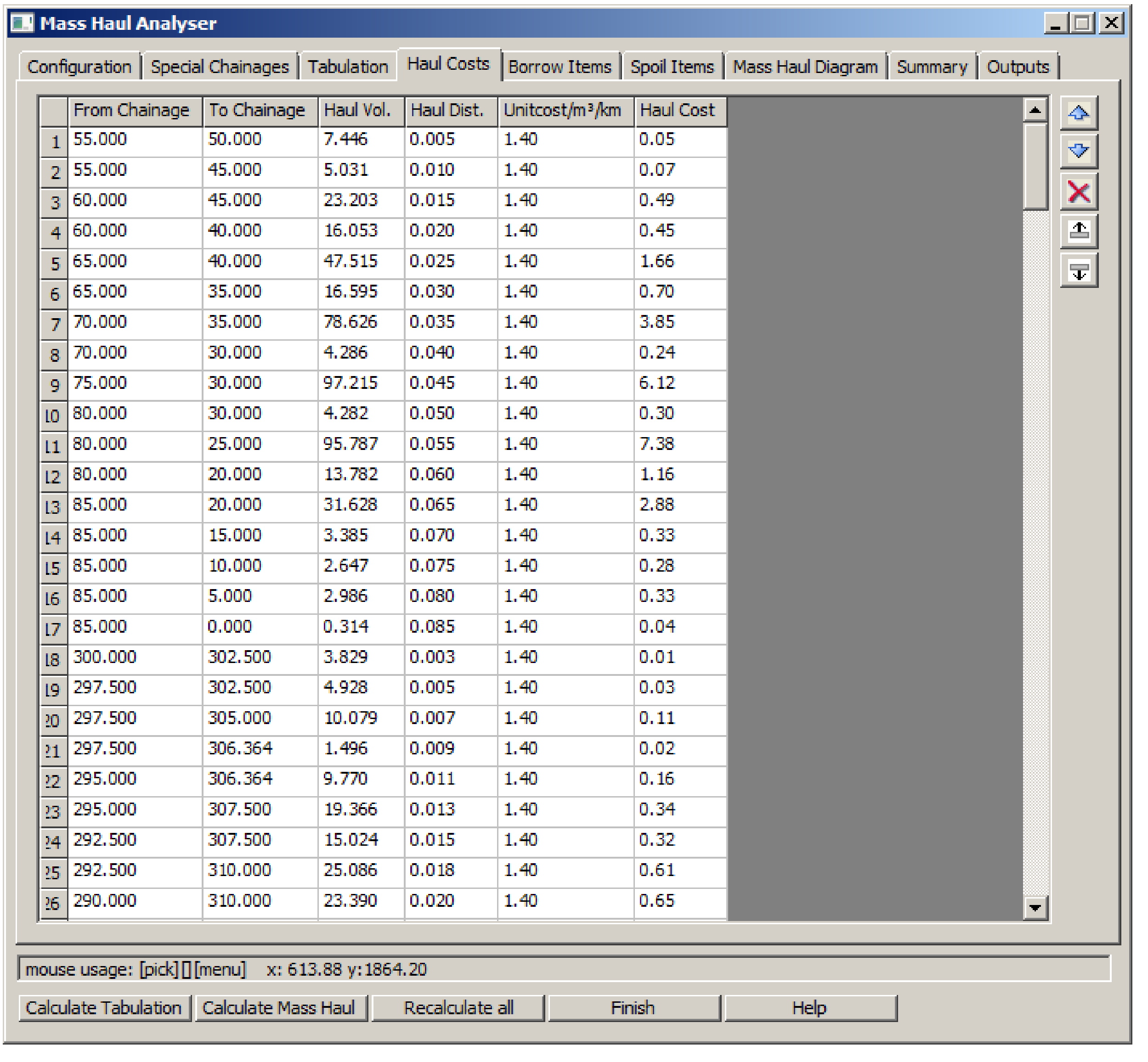
save special chainages from the grid control



	Chainage	Type	Insitu Cut Vol.	Insitu Fill Vol.	Loose cut 2 fill	Req'd loose fill	Excess loose Cu
1	0.0000	Fill	15.5260	18.8820	23.2890	0.3135	
2	5.0000	Fill	9.1950	13.4230	13.7925	2.9862	
3	10.0000	Fill	3.8770	6.7700	5.8155	2.6470	
4	15.0000	Fill	0.7810	3.6450	1.1715	3.3848	
5	20.0000	Fill	0.1290	36.4830	0.1935	45.4102	
6	25.0000	Fill		76.6300		95.7875	
7	30.0000	Fill	0.2840	84.9670	0.4260	105.7827	
8	35.0000	Fill	3.4670	80.3370	5.2005	95.2207	
9	40.0000	Fill	12.1090	65.3850	18.1635	63.5678	
10	45.0000	Fill	25.7750	53.5170	38.6625	28.2338	
11	50.0000	Fill	35.8770	49.0090	53.8155	7.4458	
12	55.0000	Cut	42.6470	41.1950	51.4937		12.4768
13	60.0000	Cut	51.7030	30.6390	38.2987		39.2558
14	65.0000	Cut	60.2380	20.9980	26.2475		64.1095
15	70.0000	Cut	67.5370	14.7150	18.3938		82.9118
16	75.0000	Cut	74.8010	11.9890	14.9863		97.2153
17	80.0000	Cut	83.8110	9.4920	11.8650		113.8515
18	85.0000	Cut	92.4710	6.5370	8.1713		130.5353
19	90.0000	Cut	99.6280	4.0910	5.1138		144.3283
20	95.0000	Cut	107.3100	2.0760	2.5950		158.3700
21	100.0000	Cut	113.2510	1.3920	1.7400		168.1365
22	105.0000	Cut	113.7690	2.5330	3.1662		167.4873
23	110.0000	Cut	113.3090	2.9640	3.7050		166.2585
24	115.0000	Cut	38.7660	0.8180	1.0225		57.1265
25	116.6920	Cut	77.2860	1.2970	1.6212		114.3078

Tabulation Tab

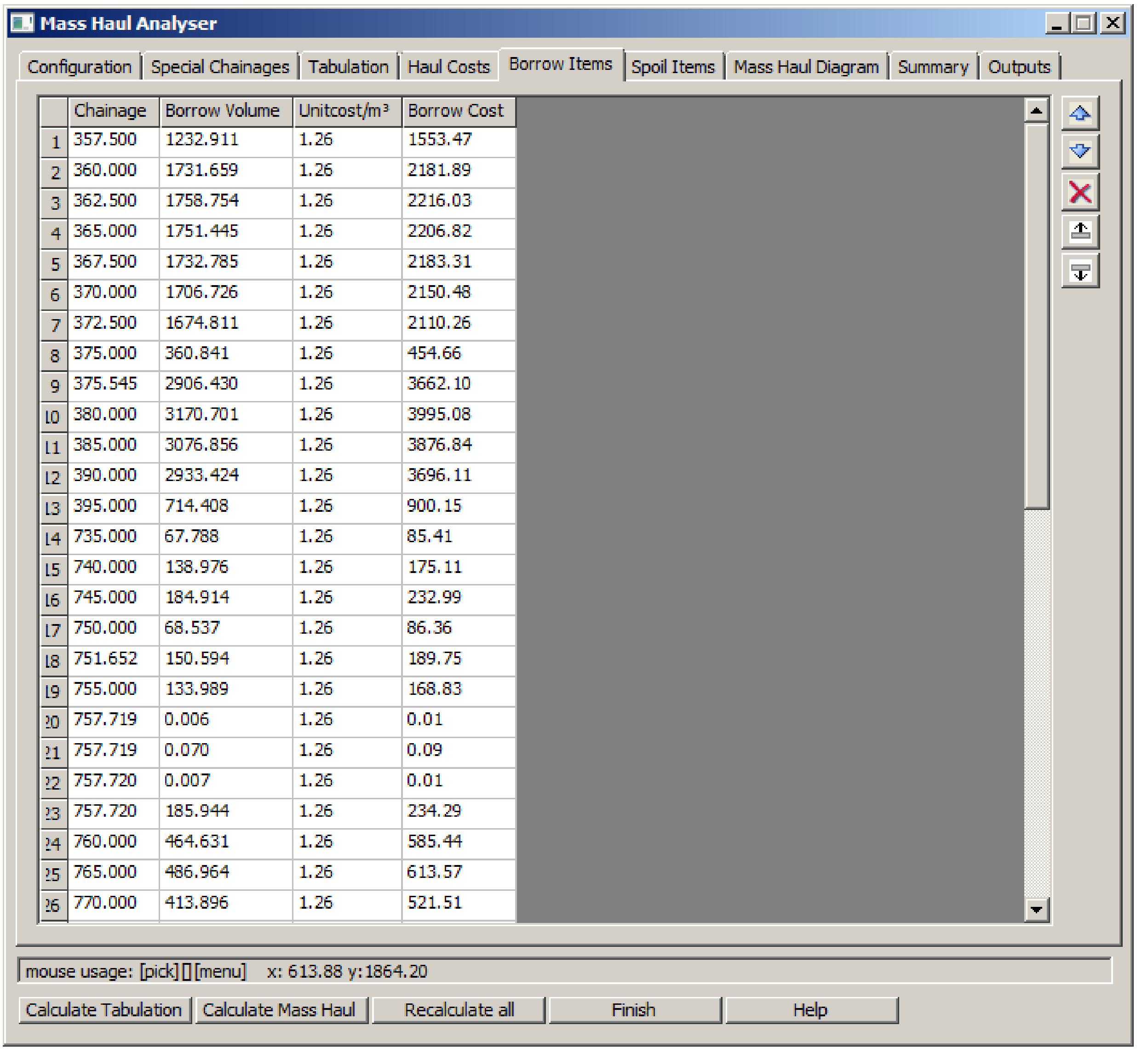
This tab displays the processed cut and fill volumes after bulking factors have been applied.



	From Chainage	To Chainage	Haul Vol.	Haul Dist.	Unitcost/m³/km	Haul Cost
1	55.000	50.000	7.446	0.005	1.40	0.05
2	55.000	45.000	5.031	0.010	1.40	0.07
3	60.000	45.000	23.203	0.015	1.40	0.49
4	60.000	40.000	16.053	0.020	1.40	0.45
5	65.000	40.000	47.515	0.025	1.40	1.66
6	65.000	35.000	16.595	0.030	1.40	0.70
7	70.000	35.000	78.626	0.035	1.40	3.85
8	70.000	30.000	4.286	0.040	1.40	0.24
9	75.000	30.000	97.215	0.045	1.40	6.12
10	80.000	30.000	4.282	0.050	1.40	0.30
11	80.000	25.000	95.787	0.055	1.40	7.38
12	80.000	20.000	13.782	0.060	1.40	1.16
13	85.000	20.000	31.628	0.065	1.40	2.88
14	85.000	15.000	3.385	0.070	1.40	0.33
15	85.000	10.000	2.647	0.075	1.40	0.28
16	85.000	5.000	2.986	0.080	1.40	0.33
17	85.000	0.000	0.314	0.085	1.40	0.04
18	300.000	302.500	3.829	0.003	1.40	0.01
19	297.500	302.500	4.928	0.005	1.40	0.03
20	297.500	305.000	10.079	0.007	1.40	0.11
21	297.500	306.364	1.496	0.009	1.40	0.02
22	295.000	306.364	9.770	0.011	1.40	0.16
23	295.000	307.500	19.366	0.013	1.40	0.34
24	292.500	307.500	15.024	0.015	1.40	0.32
25	292.500	310.000	25.086	0.018	1.40	0.61
26	290.000	310.000	23.390	0.020	1.40	0.65

Haul Costs Tab

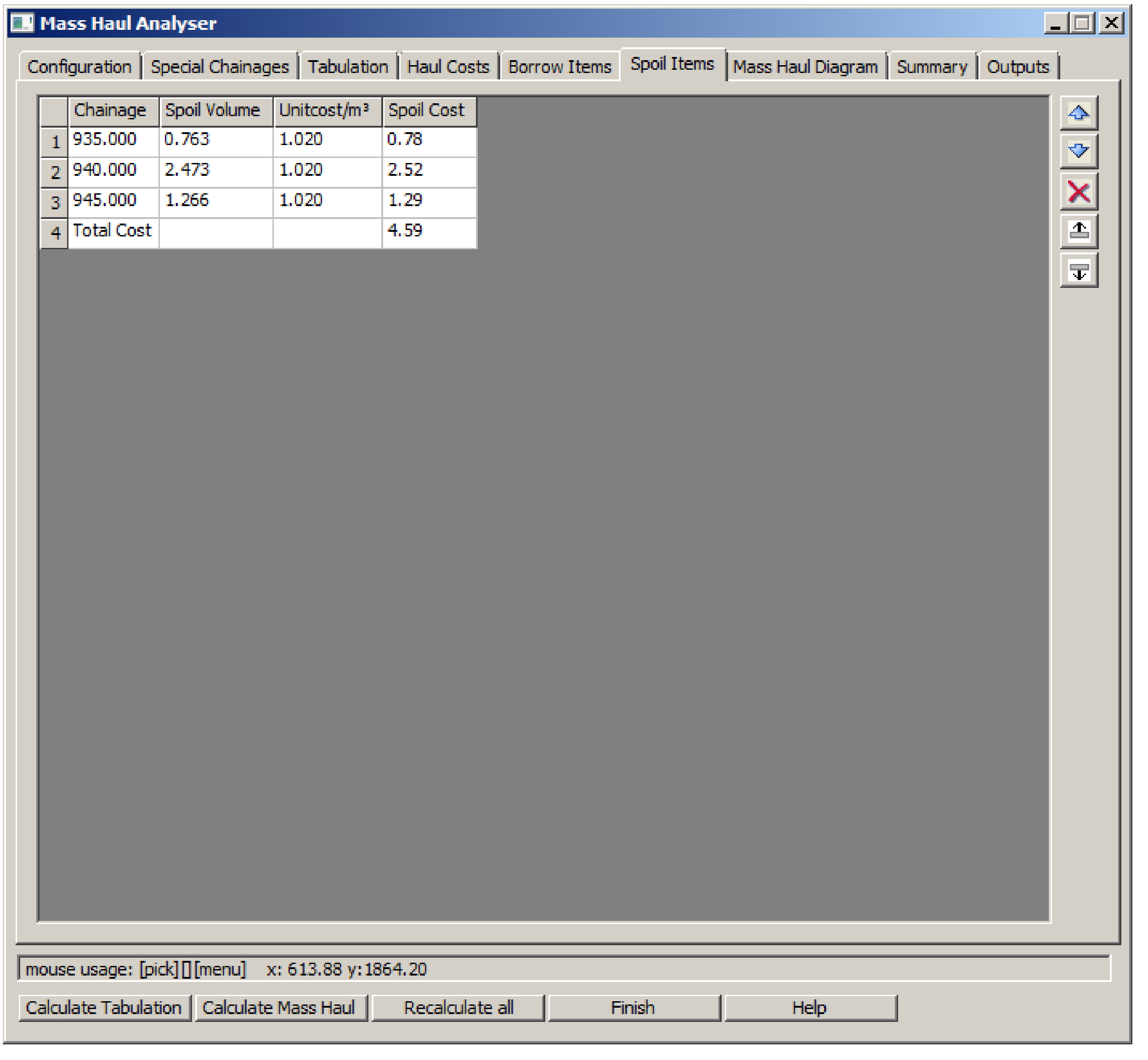
This tab displays the processed haul volumes after mass haul has been applied.



	Chainage	Borrow Volume	Unitcost/m³	Borrow Cost
1	357.500	1232.911	1.26	1553.47
2	360.000	1731.659	1.26	2181.89
3	362.500	1758.754	1.26	2216.03
4	365.000	1751.445	1.26	2206.82
5	367.500	1732.785	1.26	2183.31
6	370.000	1706.726	1.26	2150.48
7	372.500	1674.811	1.26	2110.26
8	375.000	360.841	1.26	454.66
9	375.545	2906.430	1.26	3662.10
10	380.000	3170.701	1.26	3995.08
11	385.000	3076.856	1.26	3876.84
12	390.000	2933.424	1.26	3696.11
13	395.000	714.408	1.26	900.15
14	735.000	67.788	1.26	85.41
15	740.000	138.976	1.26	175.11
16	745.000	184.914	1.26	232.99
17	750.000	68.537	1.26	86.36
18	751.652	150.594	1.26	189.75
19	755.000	133.989	1.26	168.83
20	757.719	0.006	1.26	0.01
21	757.719	0.070	1.26	0.09
22	757.720	0.007	1.26	0.01
23	757.720	185.944	1.26	234.29
24	760.000	464.631	1.26	585.44
25	765.000	486.964	1.26	613.57
26	770.000	413.896	1.26	521.51

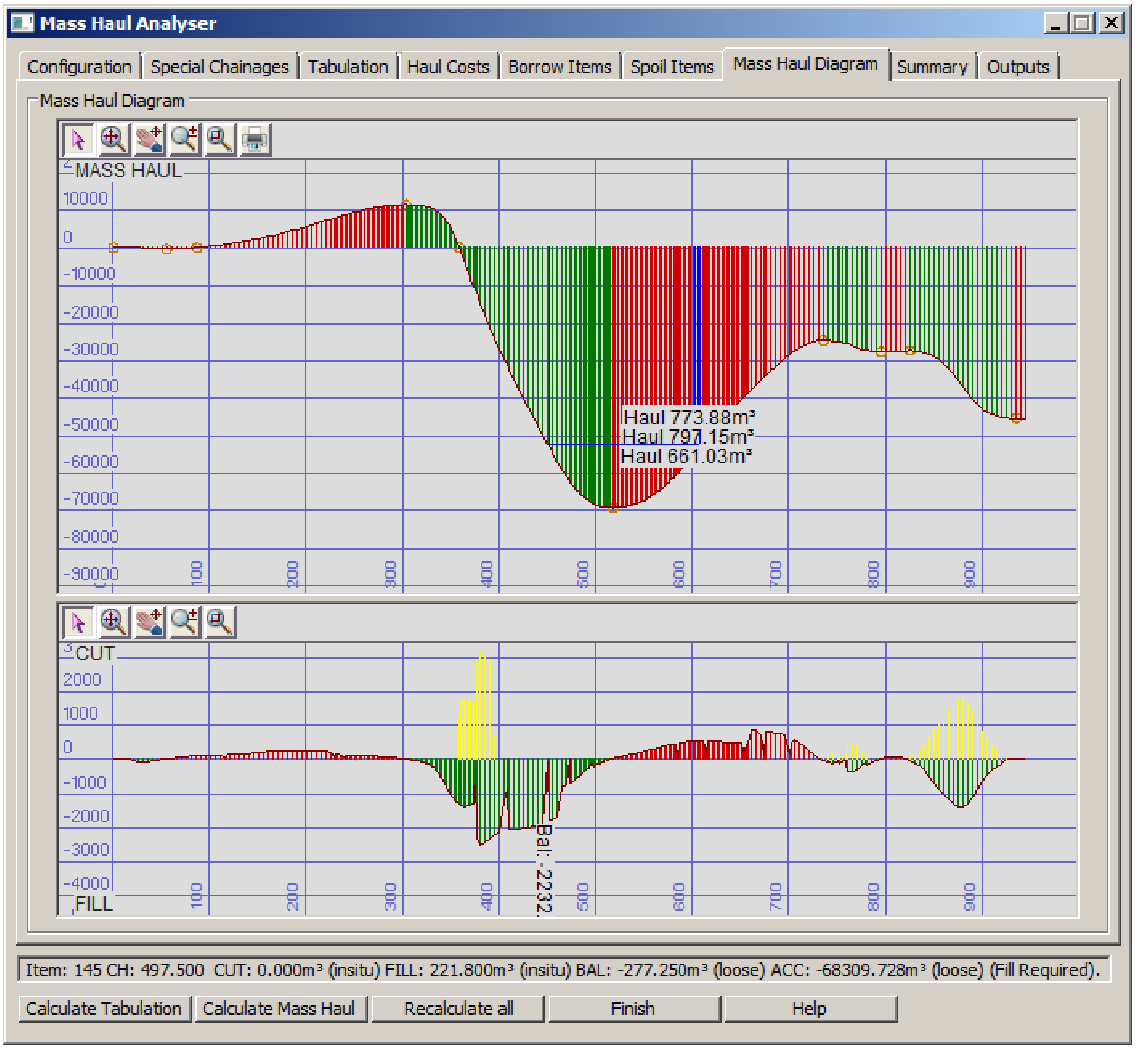
Borrow Items Tab

This tab displays the remaining borrow volumes after mass haul has been applied.



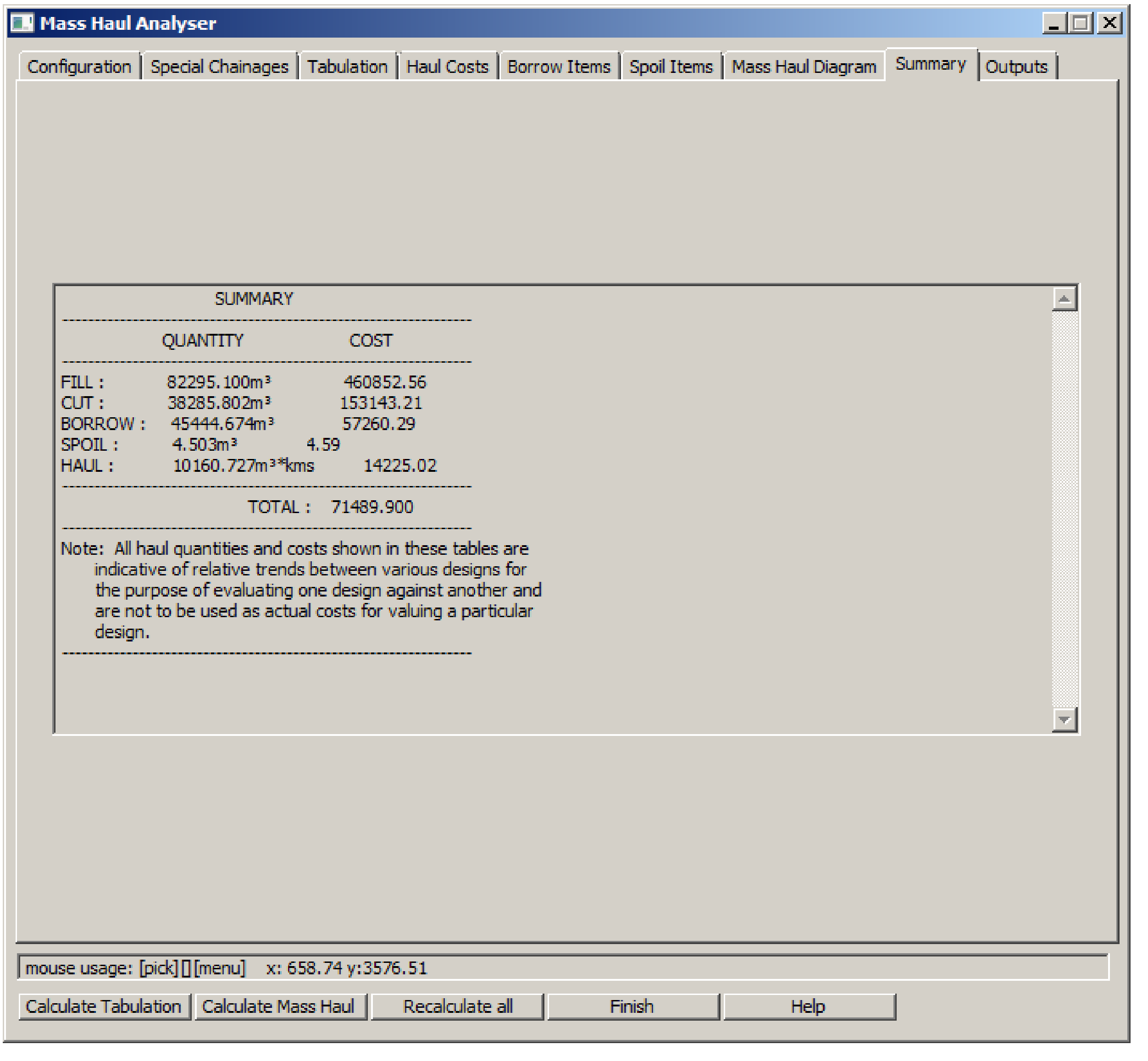
Spoil Items Tab

This tab displays the remaining spoil volumes after mass haul has been applied



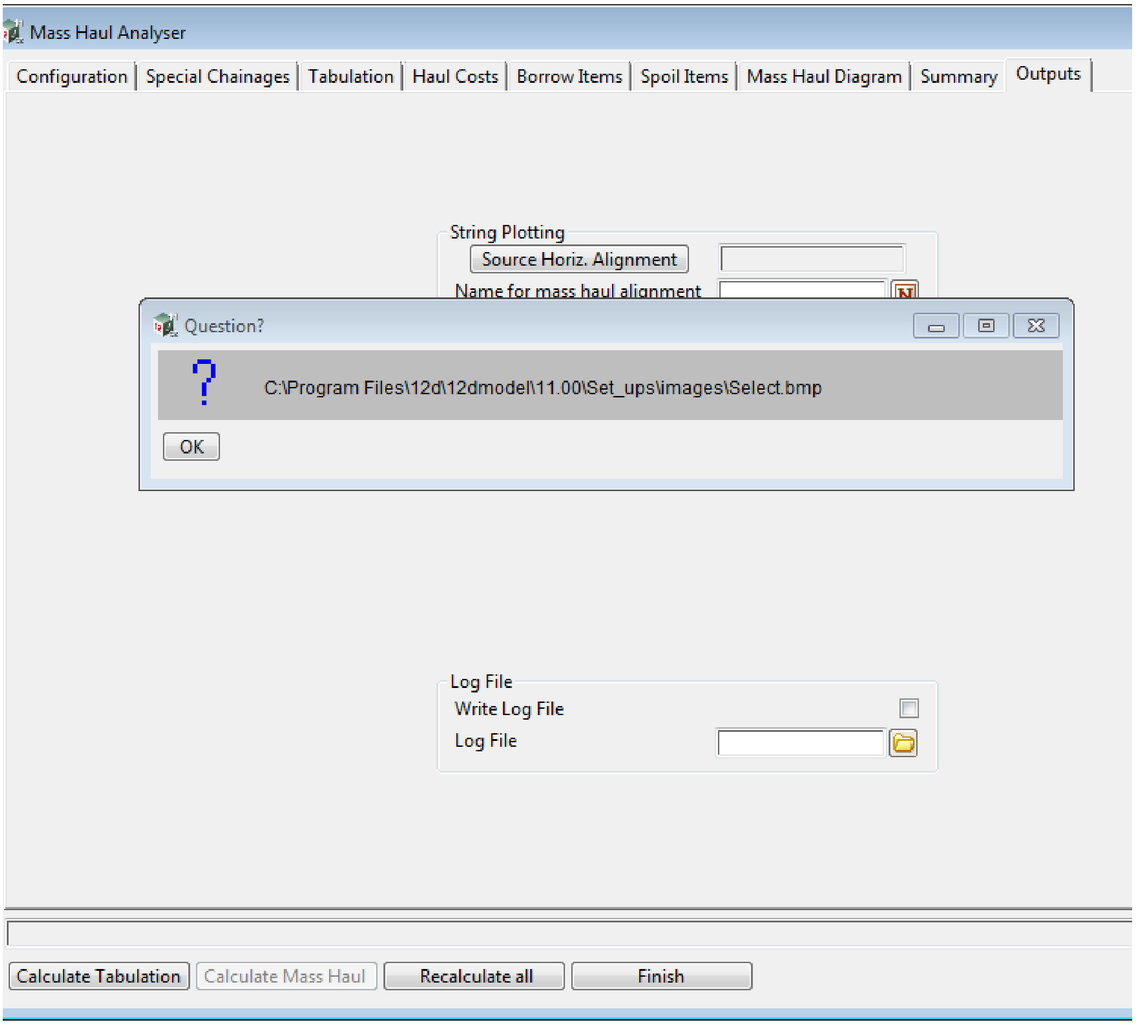
Mass Haul Diagram Tab

This tab displays a graph of the resulting mass haul diagram. For information on several features of the graph please go to 20.8.8.2 Graph Box Operation.



Summary Tab

This tab displays a summary of the costs calculated. These costs can be used to compare trends in the efficiency of the mass haul between different designs. The costs given are not intended to be the final cost of the mass haul as further analysis will be required however they do give an indication as to the efficiency of a design with regard to mass haul.



Outputs Tab

This tab can be used to export data from the option

The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Source Horiz. Alignment select string

selects the seed alignment that will be used for the horizontal alignment of the mass haul string

Name for mass haul alignment name box

sets the name of the plotted mass haul string

Model for mass haul alignment model box

sets the model of the plotted mass haul string

Colour for mass haul alignment colour box red

sets the colour of the plotted mass haul string

Plot Mass Haul Alignment button

duplicates the seed alignment and stores the mass haul accumulated volume balances as chainage *height points in that alignment*

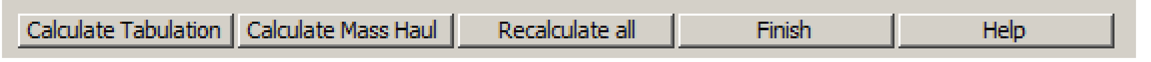
Write log file tick box

choose this option to export a log file

Log file file box

the calculations generated by the mass haul option are stored in this log file

20.8.8 Common Buttons

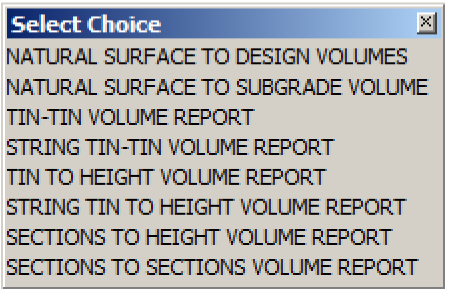


The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Calculate Tabulation	button		
recalculate tabulation			
Calculate Mass Haul	button		
recalculate mass haul			
Recalculate all	button		
recalculate both tabulation and mass haul			
Finish	button		
closes the panel			
Help	button		
displays this help			

20.8.8.1 Selecting The Right Report File Type

The report file type option allows the user to use various report file types generated by 12d.



Each type is selected to cause the option to look for certain key phrases in the report file at the top of each cut/fill/balance tabulation.

Go to NATURAL SURFACE TO DESIGN VOLUMES
NATURAL SURFACE TO SUBGRADE VOLUME
TIN-TIN VOLUME REPORT
STRING TIN-TIN VOLUME REPORT
TIN TO HEIGHT VOLUME REPORT
STRING TIN TO HEIGHT VOLUME REPORT
SECTIONS TO HEIGHT VOLUME REPORT
SECTIONS TO SECTIONS VOLUME REPORT

NATURAL SURFACE TO DESIGN VOLUMES

This choice is selected to use the natural surface to design volumes table generated by the "Apply MTF" function. This function must be run first from the **Design=>Apply=>Apply MTF** menu item.

NATURAL SURFACE TO SUBGRADE VOLUME

This choice is selected to use the natural surface to subgrade volumes table generated by the "Apply MTF" function. This function must be run first from the **Design=>Apply=>Apply MTF** menu item. In order to use this option boxing must be calculated in the Apply MTF function.

TIN-TIN VOLUME REPORT

This choice is selected to use the tin to tin volumes table generated by the Apply MTF function. This function must be run first from the **Design=>Volumes=>End Area=>Tin to tin** menu item.

STRING TIN-TIN VOLUME REPORT

This choice is selected to use the string tin to tin volumes table generated by the "Apply MTF" function. This function must be run first from the **Design=>Volumes=>End Area=>String tin to tin** menu item.

TIN TO HEIGHT VOLUME REPORT

This choice is selected to use the tin to height volumes table generated by the "Apply MTF" function. This function must be run first from the **Design=>Volumes=>End Area=>Tin to height** menu item.

STRING TIN TO HEIGHT VOLUME REPORT

This choice is selected to use the string tin to height volumes table generated by the "Apply MTF" function. This function must be run first from the **Design=>Volumes=>End Area=>String tin to height** menu item.

SECTIONS TO HEIGHT VOLUME REPORT

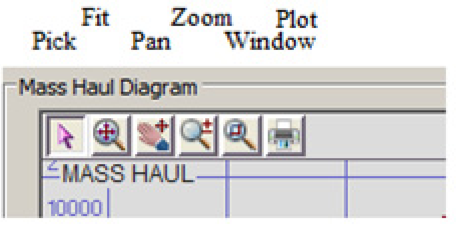
This choice is selected to use the sections to height volumes table generated by the "Apply MTF" function. This function must be run first from the **Design=>Volumes=>End Area=>Sections to height** menu item.

SECTIONS TO SECTIONS VOLUME REPORT

This choice is selected to use the sections to sections volumes table generated by the "Apply MTF" function. This function must be run first from the **Design=>Volumes=>End Area=>Sections to sections** menu item.

20.8.8.2 Graph Box Operation

Each graph box has a number of graph options available as buttons (graph buttons) at the top of the graph. To select one of the graph buttons, click LB whilst the cursor is over the button.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Pick	button		

This option sets the graph box to pick mode. When in this mode the graph box will show various information as the mouse hovers over a chainage in the graph.

The loose material tabulation item number, the available cut and fill within that chainage interval are shown in the message area at the bottom of the panel and the borrow, spoil and haul amounts are graphically shown in the graph highlighting the source chainage and destination chainage.

Fit button

This option sets the graph box to fit mode. This option will calculate the extents of the items plotted in the graph box and zoom out to show them. After the fit is complete the graph will return to pick mode.

Pan button

This option sets the graph box to panning mode. Click once to commence panning, then move the mouse and click a second time to stop panning.

Zoom button

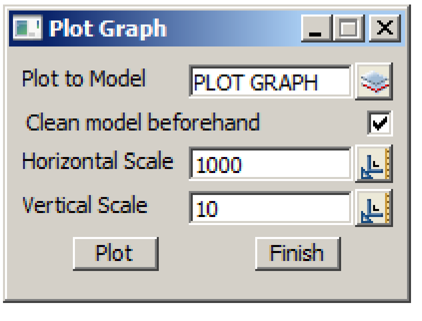
This option sets the graph box to zoom mode. Click once to commence zooming, then move the mouse and click a second time to stop zooming. Moving the mouse left and right will reduce and increase the X scale factor and moving the mouse down and up will reduce or increase the Y scale factor.

Window button

This option sets the graph box to window mode. Click once to select the first corner of a window and click a second time to select the second corner. After the window is complete the graph will return to pick mode.

Plot button

This option enables the contents of the graph box to be plotted to a **12d Model**.
Click on the button and the ***Plot Graph*** *panel is displayed.*



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Plot to Model	model box	PLOT GRAPH	all available models

the name of the model to be created/used

Clean model beforehand check box off

deletes any existing data from the plot model

Horizontal Scale real box 1000

set the horizontal scale to be used to plot the data

Vertical Scale real box 10

set the vertical scale to be used to plot the data

Plot button

plot the data

Finish button

closes the panel

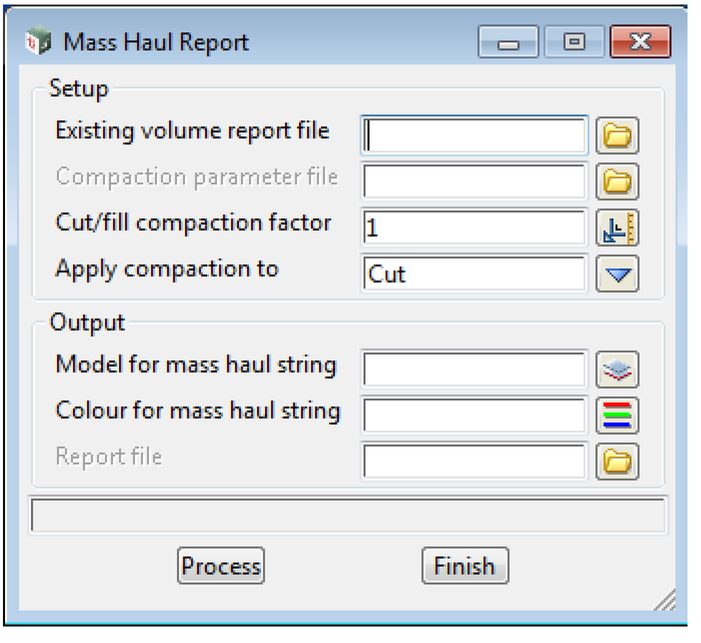
20.8.8.3 Mass Haul String and Report

Position of option on menu: **Design =>Roads =>Mass haul =>Mass haul**

This panel is used to read in a volumes report and optionally a cut/fill compaction parameter file and creates a mass haul string and a new volumes report with the compaction volumes.

The compaction parameter file simply contains the compaction factor for given chainage ranges. Either the cut volumes are multiplied by the compaction factor or the fill volumes are divided by the compaction factor, to give the compacted volumes in the new volumes report.

If the existing volume report contains an alignment string and that string can be found in the project then a new alignment string is created with the same horizontal geometry but with the accumulated volumes as the z-values.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Existing volume report file file box *.rpt files

name of the existing volumes report.

Compaction parameter file input *.mhf files

file containing compaction factors to be used. See 20.8.8.4 Format of the Compaction Parameter File.

Cut/fill compaction factor input 1

cut/fill compaction factor for any chainages not in the compaction parameter file or if there is no file.

Apply compaction factor to choice box cut cut, fill

*if **cut**, the cut volumes are multiplied by the compaction factors.*
*If **fill**, the fill volumes are divided by the compaction factors.*

Model for mass haul string model box available models

model for the created mass haul string.

Colour for mass haul string colour box available colours

colour of the created mass haul string.

Report file input box

if non-blank, a mass haul volumes file is created.

Process button

run the option.

20.8.8.4 Format of the Compaction Parameter File

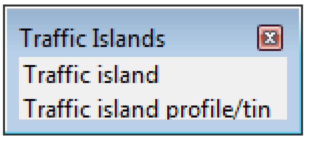
```
Format of the
// Sample cut/fill compaction parameter file - the file ending
is .mhf
// All lines starting with // are comments and blank lines are
ignored
//
// A range is specified by a start and end chainage and a compaction
factor.
// A compaction factor of 1.0 means there is no compaction.
// A compaction factor of 0.9 means that 1.0 cubic metres of cut is
equivalent
// to 0.9 cubic metres of fill due to compaction.

// format is
// Start_ch    End_ch   Compaction_factor

       0         100        0.9
      100        500        1.0
      500        2000       1.1
```

20.8.9 Traffic Islands

Position of menu: **File I/O =>Roads =>Traffic islands**

The options under **Traffic islands** create a traffic island from three travel lines, offsets and nose radii, and also can apply a kerb profile around the island.

The **Traffic islands** walk-right menu is



create a traffic island

run a kerb profile around a traffic island

For

Traffic island 20.8.9.1 Create Traffic Island

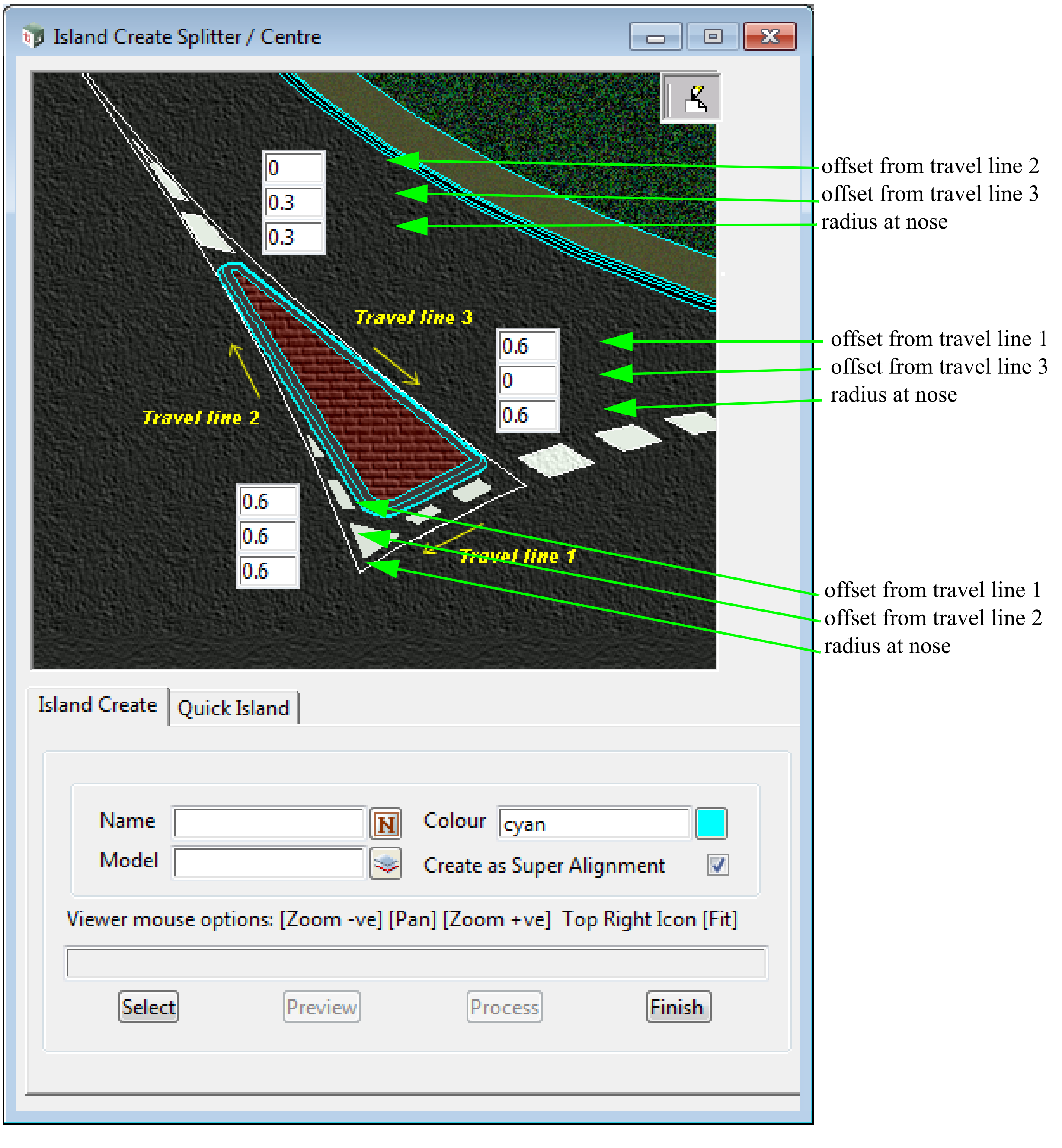
Traffic island profile/tin 20.8.9.2 Traffic Island Profile/Tin

20.8.9.1 Create Traffic Island

Position of option on menu: **Design =>Roads =>Traffic islands =>Traffic Island**

Create an alignment string for a traffic island between user selected strings and with user defined offsets from the strings and nose radii. Drape the island onto a tin to give is z-values and then apply a kerb profile to complete the island.

On selecting the **traffic island** option, the **Island Create Splitter/Centre** panel is displayed.



The position of the panel fields indicate what the values are for. Tool tips appear when the cursor is passed over the panel.fields.

The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up

Buttons at bottom

Select strings select

select in order and with direction, the three strings to use in creating the traffic island. Terminate by clicking RB and selecting cancel from the pick ops menu.

Preview string select

draw the traffic island with the given parameters in the panel draw box.

Process button

create the alignment string

Island Create tab

Offset from travel lines input box

offset distance from the selected travel lines.

Radius at nose input box

radius of the corner of the traffic island between the selected travel lines

Name name box available names

name of the created alignment string.

Model model box available models

model for the created alignment string.

Colour colour box available colours

colour of the created alignment string.

Create as super alignment tick box

if ticked, create the string as a super alignment.
If not ticked, create the string as an alignment.

Quick Island tab

the quick island tab produces a traffic island of fixed sides but with the given nose radii. The traffic island can be created as a super alignment that can then be easily moved around and modified.

Radius at nose input box

radius of the corner of the traffic island

Name name box available names

name of the created alignment string

Model model box available models

model for the created alignment string

Colour colour box available colours

colour of the created alignment string

Create as super alignment tick box

if ticked, create the string as a super alignment.
If not ticked, create the string as an alignment.

Position xyz select box

pick two points to indicate the direction that the island will be created.

Then click on **Process** *to create the Traffic Island.*

Kerb Profile tab

Use island alignment levels instead of Tin tick box

if ticked, use the alignments own z-value instead of draping the alignment onto the tin. If not ticked, drape the alignment onto the tin to get z-values.

Offset kerb face/top/back input box

offset for kerb face/top/back for the selected kerb type.

Height kerb face/top/back input box

height for kerb face/top/back.

Tin tin box available tins

triangulation to drape the traffic island strings onto.

Kerb type input box

type of kerb to apply to the reference string of the traffic island. If type **User** *is selected from the pop-up list, a file which defines the kerb profile can be selected and used.*

File file box *.ukt files

file to read/write the user defined kerb type to.

Read button

read the given **File** *to define a kerb type.*

Save button

save the kerb parameters to the given **File**.

20.8.9.2 Traffic Island Profile/Tin

Position of option on menu: **Design =>Roads =>Traffic islands =>Traffic island profile/tin**

This option applies a kerb profile, either from the **12d** library or from a user defined file.

The template profile is applied to a Super Alignment, creating strings, and a triangulation.

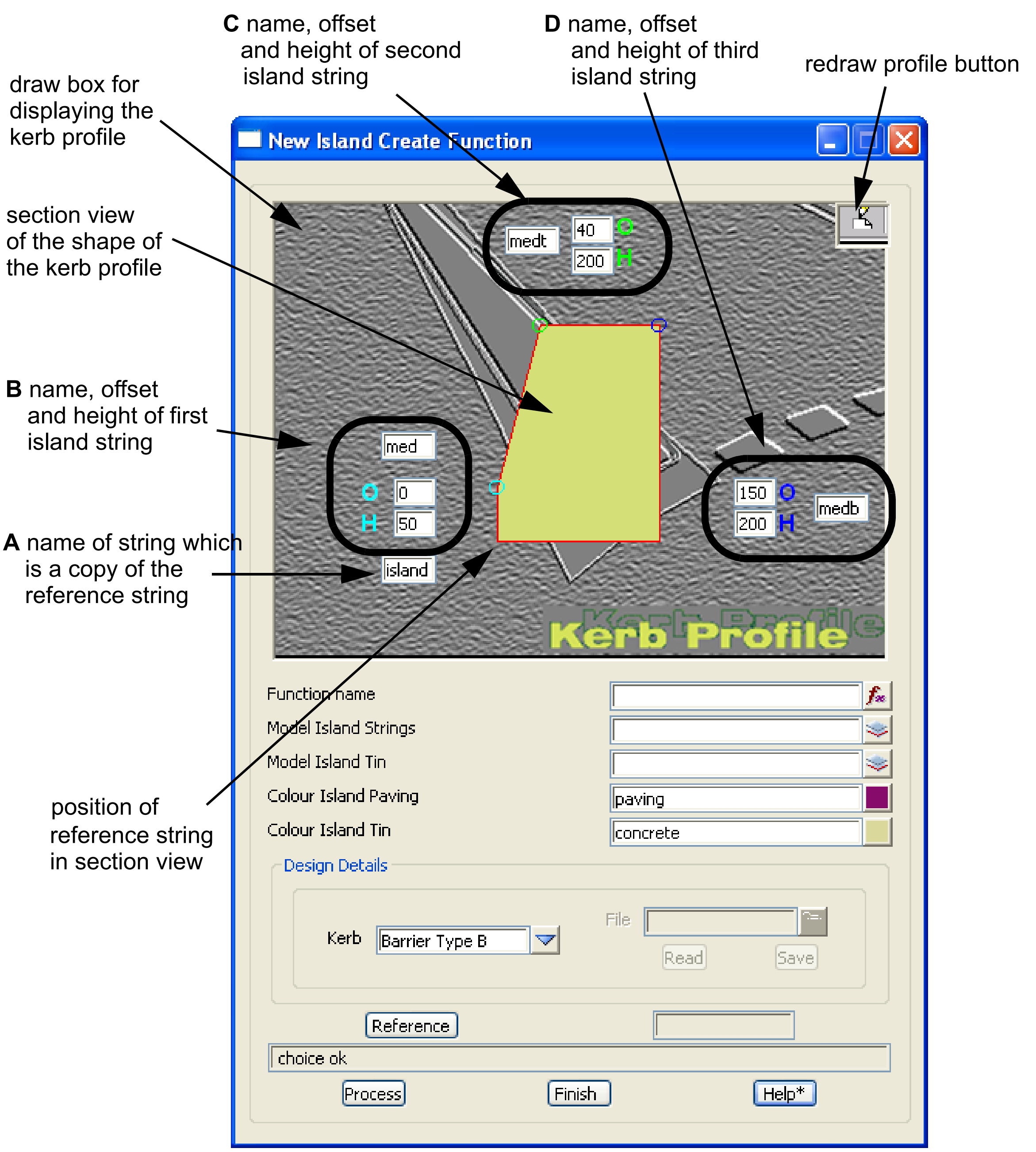
The option is saved as a function within **12d** and as a function can be re-run at any time if changes occur with the Super Alignment.

Some default kerb profiles are available under **Kerb** on the panel.

User defined kerbs are created by entering offset, height and string names in the fields in the draw box at the top of the panel, and then saved as a User Kerb Type file (*.ukt)

User Kerb Type files can then be written out, or read in as required.

On selecting the **Traffic Island profile/tin**, the **New Island Create Function** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
Function name	function		available functions

name of the Island Create function

Model Island Strings	model box		available models

model for the created island strings

Model Island Tin	model box		

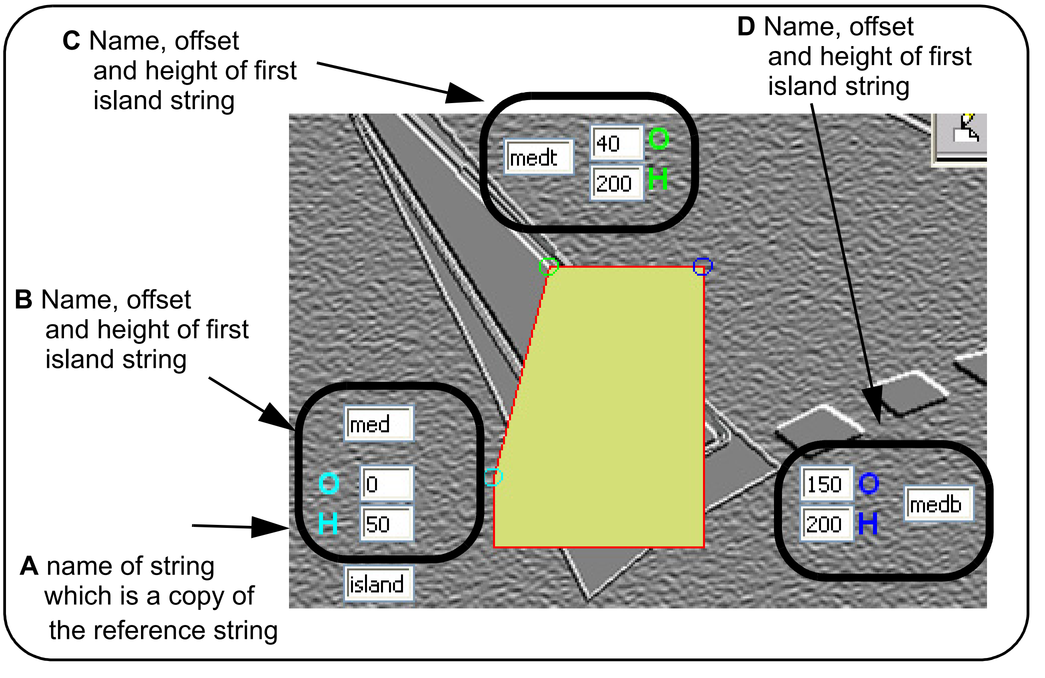
model for the tin of the created island

Colour Island Paving	input	paving	available colours

*if non-blank, then the colour is used to colour the island triangulation within innermost kerb string (**D**)*

Colour Island Tin	input	concrete	available colours

*if non-blank, then the colour is the colour is used to colour the island triangulation between the strings defined by **A** and **D**.*



In the drawing area:

A

a string is created which is a copy of the reference string and given this name. It is added to the **Model Island Strings** *model.*

B

name, offset and height from the reference string, of the first string of the island. It is added to the **Model Island Strings** *model. If offset and height are both zero, then the string is not created.*

C

name, offset and height from the reference string, of the second string of the island. It is added to the **Model Island Strings** *model. If offset and height are both zero, then the string is not created.*

D

name, offset and height from the reference string, of the third string of the island. It is added to the **Model Island Strings** *model. If offset and height are both zero, then the string is not created.*

Design Details

Kerb	choice box	Barrier Type A	Barrier Type A, Barrier Type B, Mountable, Semi - Mountable, User

selected kerb type is displayed in draw box above

If **Kerb** *is* ***User*** *then the* **File** *field and the* **Read** *and* **Save** *buttons are made active, and are used to read in an existing user defined Kerb type, or to create a new one.*

File	file box	files ending in.ukt

name of the user kerb type file (ukt) to read in or create

Read button

Reads in the user defined kerb definition from the ukt file given in the **File** *field and displays the profile and Offsets and Height in the draw box at the top of the Panel*

Save button

Writes out the kerb profile information (O and H values, and string names) as a user defined kerb, into the file given in the **File** *field.*

Reference select pick

Select the reference alignment which will have the kerb profile applied to it to create the island.“pick with direction" in a clockwise manner as the kerb profile is always on the RHS.

Process button

runs the option which created the island string and the coloured island tin

Note:

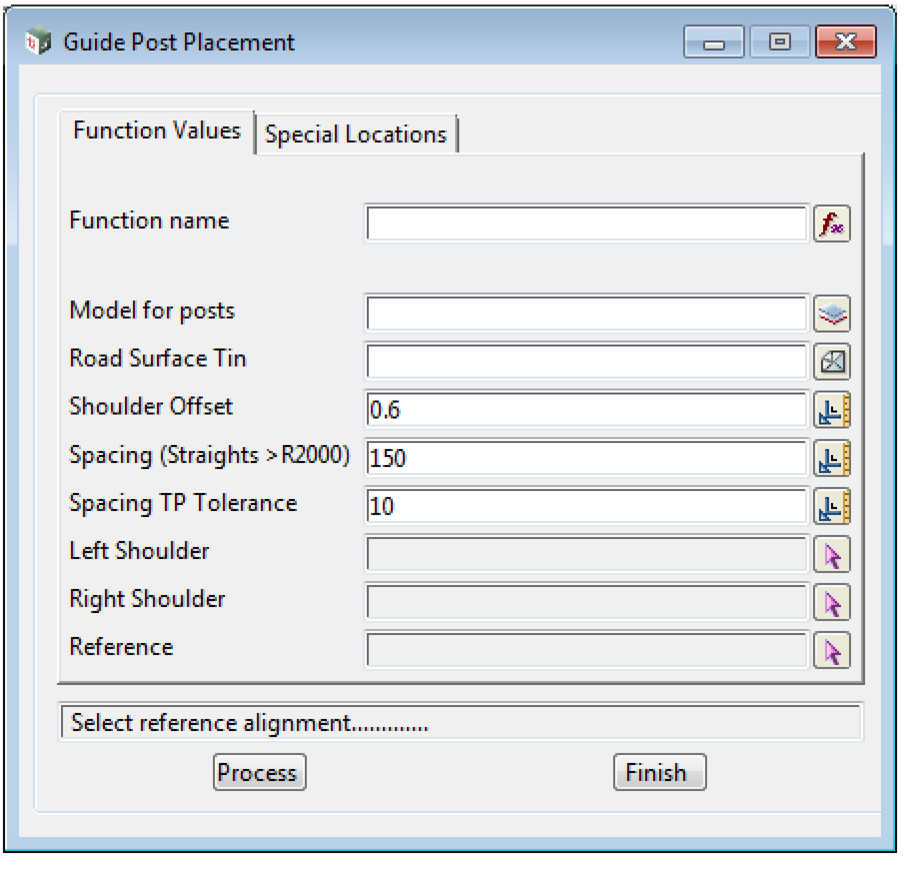
The New Island Create Function reads in the standard kerb profiles as templates and if required, creates a template from the ukt files. The function needs these templates because an apply is used (internally) to create the Island.

20.8.10 Guide Post Placement

Position of option on menu: **Design =>Roads =>Guide post creator**

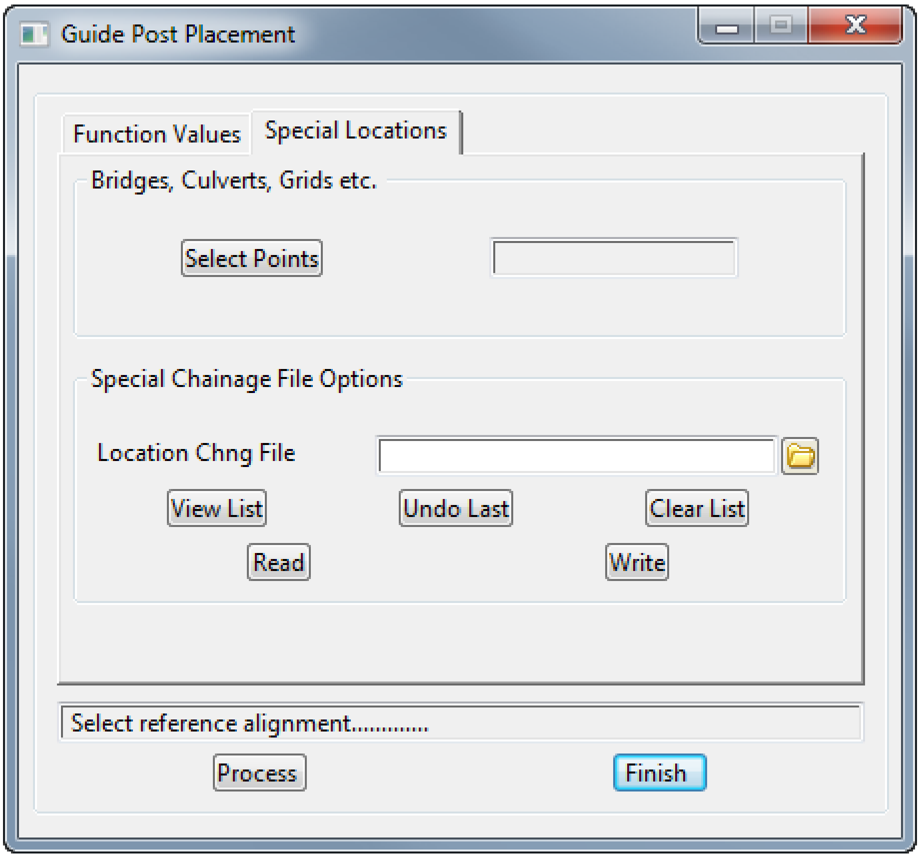
This section of documentation is a work in progress and will be updated in subsequent releases.

Selecting **Guide post creator** brings up the **Guide Post Placement** panel.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Function Values tab			
Function name	function box		available functions
Model for posts	model box		available models
Road Surface Tin	tin box		available tins
Shoulder Offset		0.6	
Spacing (Straights >R2000)		150	
Spacing TP Tolerance		10	
Left Shoulder	choice box		
Right Shoulder	choice box		
Reference	choice box		
Process	button		



The fields and buttons used in this panel have the following functions:

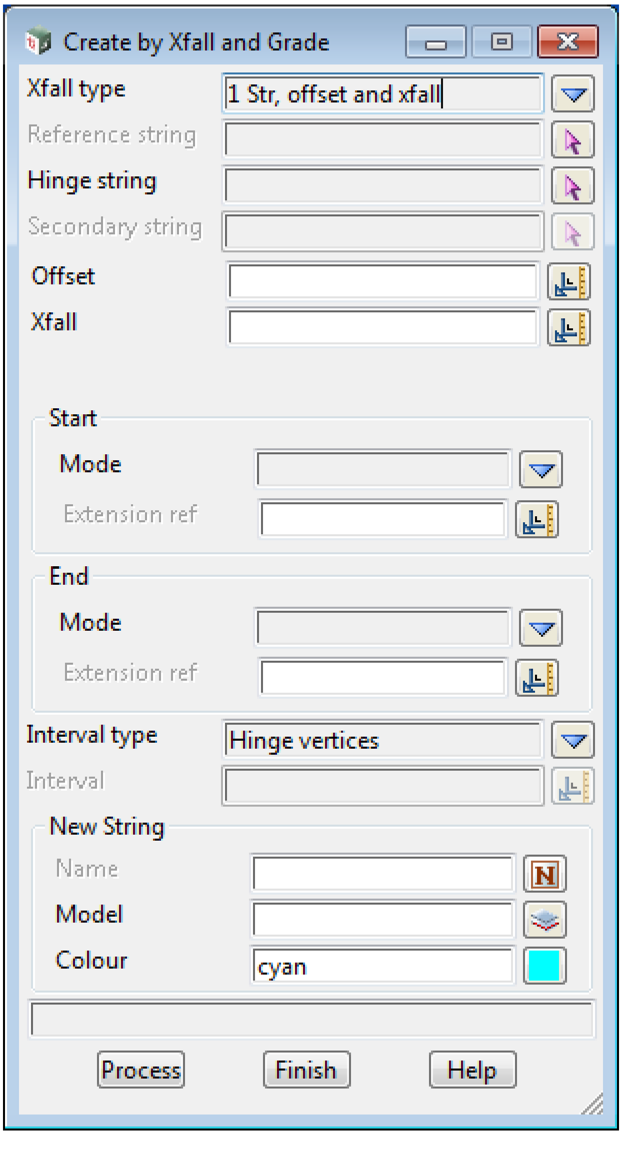
Field Description	Type	Defaults	Pop-Up
Special Locations tab			
Bridges, Culverts, Grids etc.			
Select Points	button		
Special Chainage File Options			
Location Chng File	file		
View List	button		
Undo Last	button		
Clear List	button		
Read	button		
Write	button		
Process	button		

20.8.11 String by Xfall and Grade

Position of option on menu: **Design =>Roads =>String by xfall and grade**

This panel is used to create strings by a variety of methods using xfall from one string, or xfalls from two strings, either using a reference string to define where the xfall is measured, or perpendicular to selected strings. Some of the methods grade the created string.

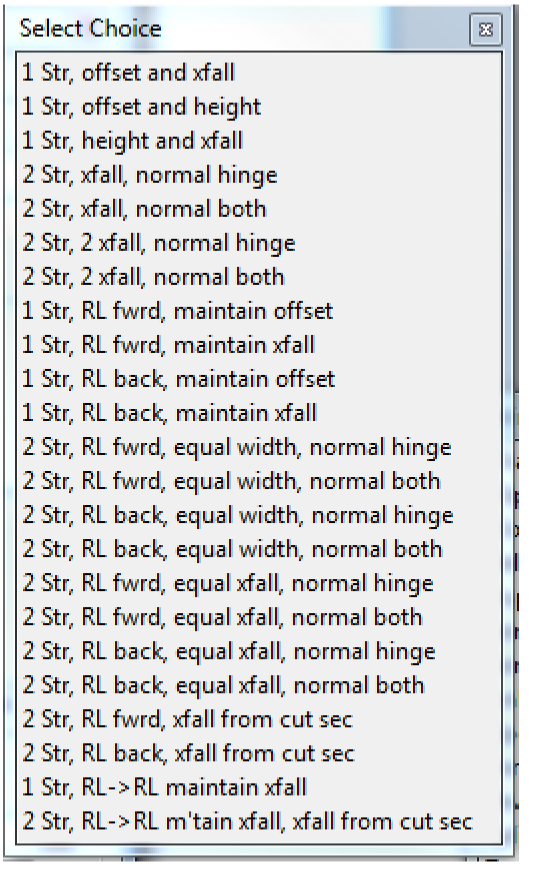
Selecting **String by xfall and grade** brings up the **Create by Xfall and Grade** panel.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Xfall type choice box



methods for creating the new string. For the definition of each choice, go to 20.8.11.1 Xfall Types

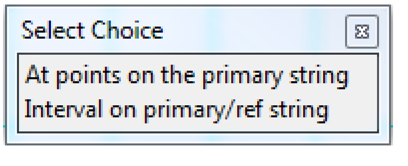
Reference string
Hinge string
Secondary string
Offset, Xfall, Grade etc

which of these fields are present, and what they are for depends on the **Xfall type**.

Start/End mode choice box Start (ref)/End (ref)

defines the start/end chainages for creating the string.
The Start/End modes are the same as for the MTF modifiers. For more information on Start/End mode, see 21.2.1 MTF Hinge Modifiers**.**

Interval type choice box



At vertices on the Hinge string -

Interval on Hinge/ref string -

String name name box names.4d list

the name to use for the created string.

Model for string model box available models

the created string is placed in this mode

String colour colour box green available colours

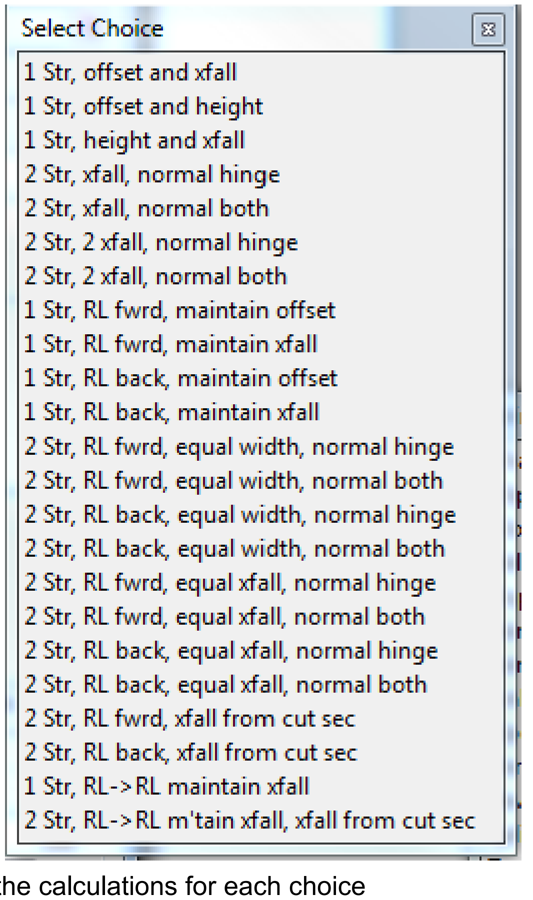
the colour of the created string

Process button

run the option.

Continue to the next section 20.8.11.1 Xfall Types.

20.8.11.1 Xfall Types

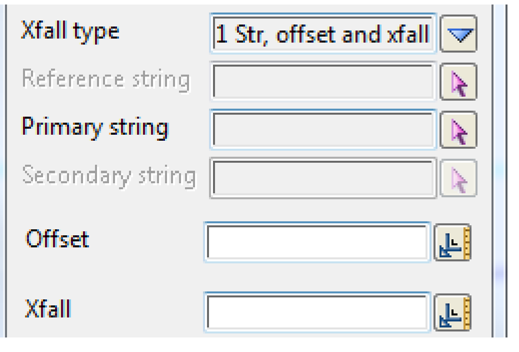


for the definitions of the calculations for each choice

1 Str, offset and xfall go to 20.8.11.1.1 1 String, Offset and Xfall
1 Str, offset and height 20.8.11.1.2 1 String, Offset and Height
1 Str, height and xfall 20.8.11.1.3 1 String, Height and Xfall
2 Str, xfall, normal hinge 20.8.11.1.4 2 Strings, Xfall and Normal to Reference String or Hinge String
2 Str, xfall, normal both 20.8.11.1.5 2 Strings, Xfall and Normal to Both
2 Str, 2 xfall, normal hinge 20.8.11.1.6 2 Strings, 2 Xfalls and Normal to Reference String or Hinge String
2 Str, 2 xfall, normal both 20.8.11.1.7 2 Strings, 2 Xfalls and Normal to Both
1 Str, RL fwrd, maintain offset 20.8.11.1.8 1 String, Start RL and Grade, Fixed Offset
1 Str, RL fwrd, maintain xfall 20.8.11.1.9 1 String, Start RL and Grade, Fixed Xfall
1 Str, RL back maintain offset 20.8.11.1.10 1 String, End RL and Grade, Fixed Offset
1 Str, RL back, maintain xfall 20.8.11.1.11 1 String, End RL and Grade, Fixed Xfall
2 Str, RL fwrd, equal width, normal hinge 20.8.11.1.12 2 Strings, Start RL & Grade, Equal Width, Normal to Reference or Hinge String
2 Str, RL fwrd, equal width, normal both 20.8.11.1.13 2 Strings, Start RL & Grade, Equal Width

For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.

20.8.11.1.1 1 String, Offset and Xfall



This option requires a **Hinge** string, an **Offset** and **Xfall**, and an **optional Reference** string.

Reference String Selected

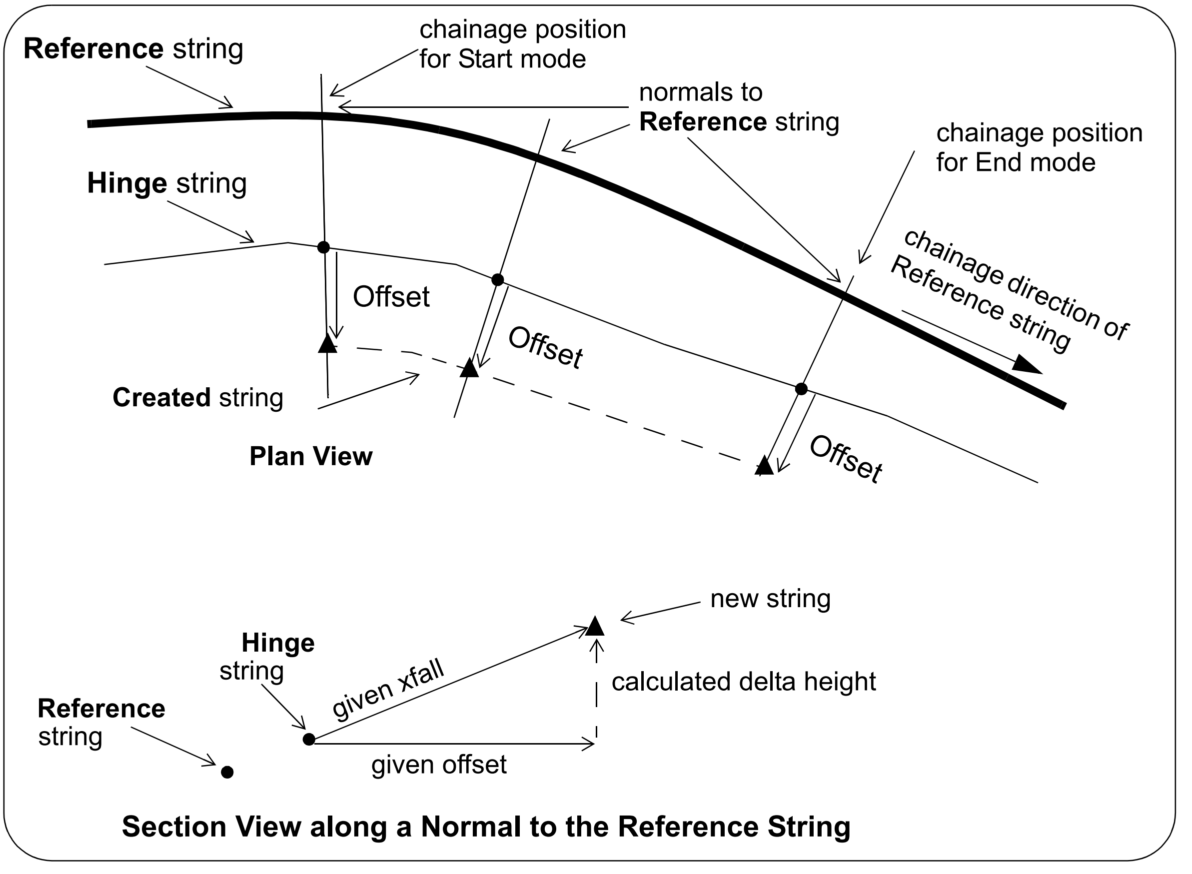
If a **Reference** string is selected then the **Start mode** and **End Mode** refer to **Reference** string and at each chainage ch between the Start mode and the End mode, a section is taken **normal** (perpendicular) to the **Reference string** and cutting the **Hinge** string.

A vertex of a new string is created by going out from the Hinge string along the section at the given **Offset** and **Xfall**.

Positive Offset is defined as going to the right of the **Hinge** string when travelling in the direction of **increasing chainage on the Reference** string. Offset can be positive or negative.

Positive Xfall is in the upward direction. Xfall can be positive or negative.

For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.



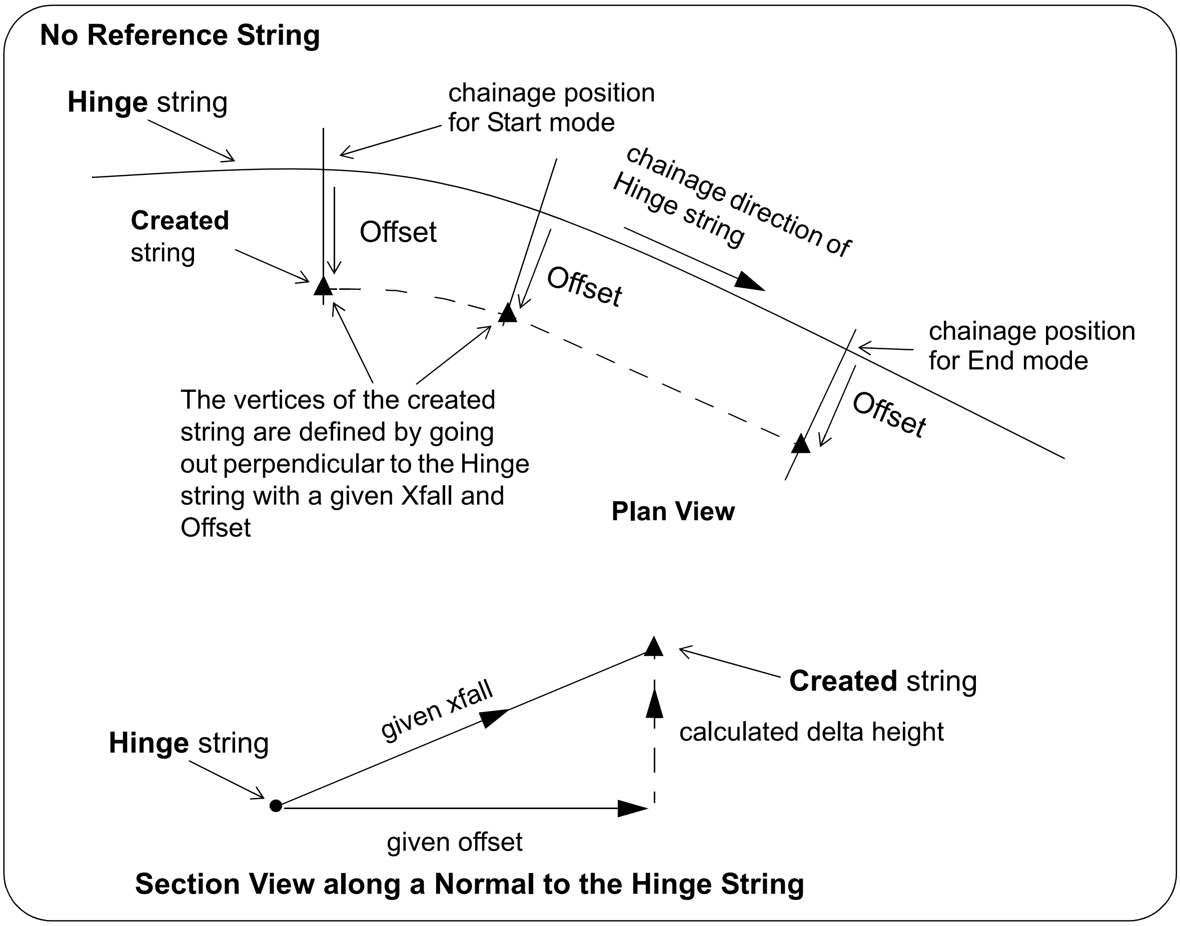
Reference String NOT Selected

If a **Reference** string is **NOT** selected then the **Start mode** and **End Mode** refer to the **Hinge** string and at each chainage ch, a vertex of the new string is created by going out **normal** (perpendicular) to the **Hinge string** for the given **Offset** and **Xfall**.

Positive Offset is defined as going to the right of the Hinge string when travelling in the direction of increasing chainage on the Hinge string. Offset can be positive or negative.

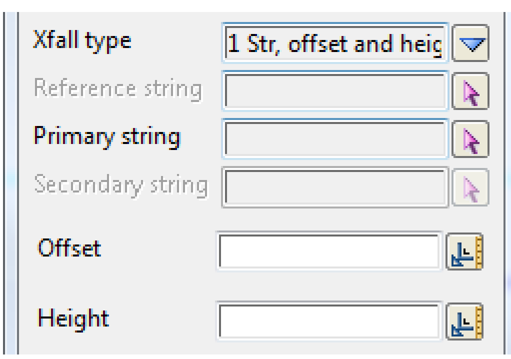
Positive Xfall is in the upward direction. Xfall can be positive or negative.

For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.



Continue to the next choice 20.8.11.1.2 1 String, Offset and Height or return to 20.8.11.1 Xfall Types or 20.8.11 String by Xfall and Grade.

20.8.11.1.2 1 String, Offset and Height



This option requires a **Hinge** string, an **Offset** and **Height**, and an **optional Reference** string.

Reference String Selected

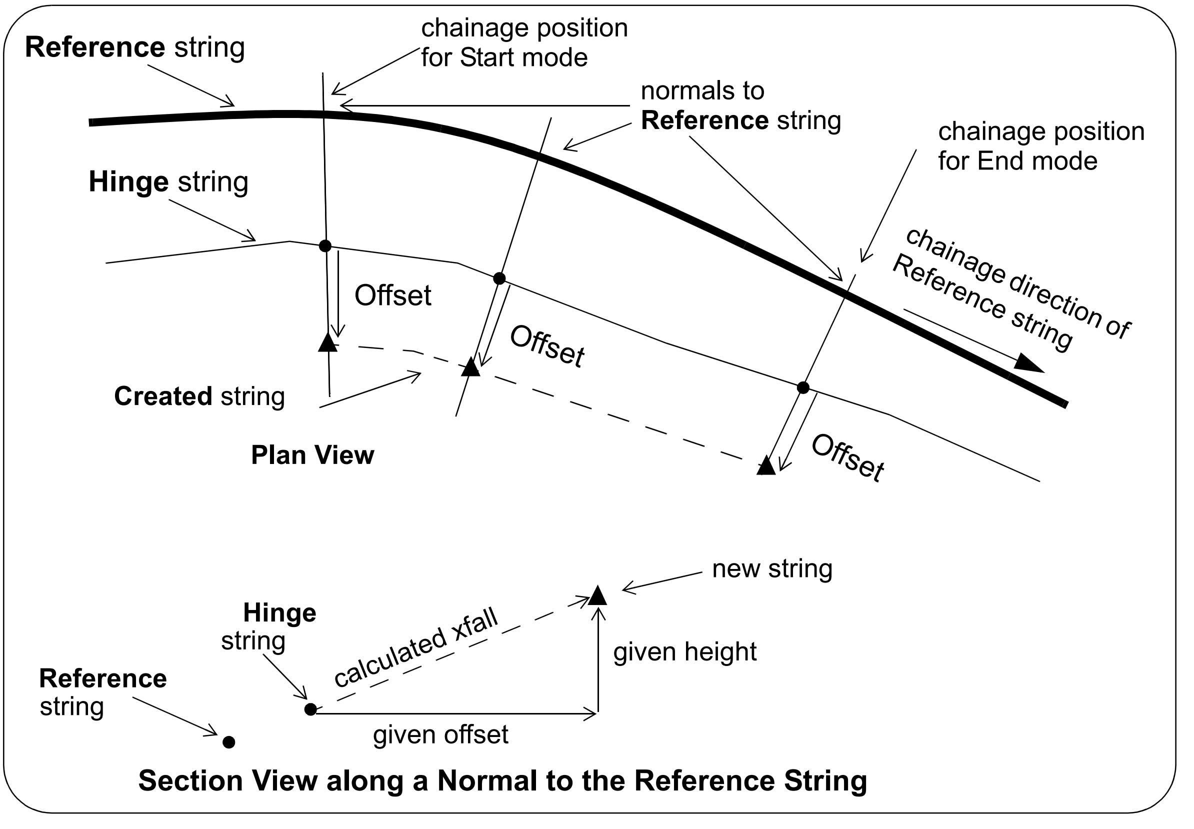
If a **Reference** string is selected then the **Start mode** and **End Mode** refer to **Reference** string and at each chainage ch between the Start mode and the End mode, a section is taken **normal** (perpendicular) to the **Reference string** and cutting the **Hinge** string.

A vertex of a new string is created by going out from the Hinge string along the section at the given **Offset** and **Height**.

Positive Offset is defined as going to the right of the **Hinge** string when travelling in the direction of **increasing chainage on the Reference** string. Offset can be positive or negative.

Height is **added** to the height on the Hinge string. Height can be positive or negative.

For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.



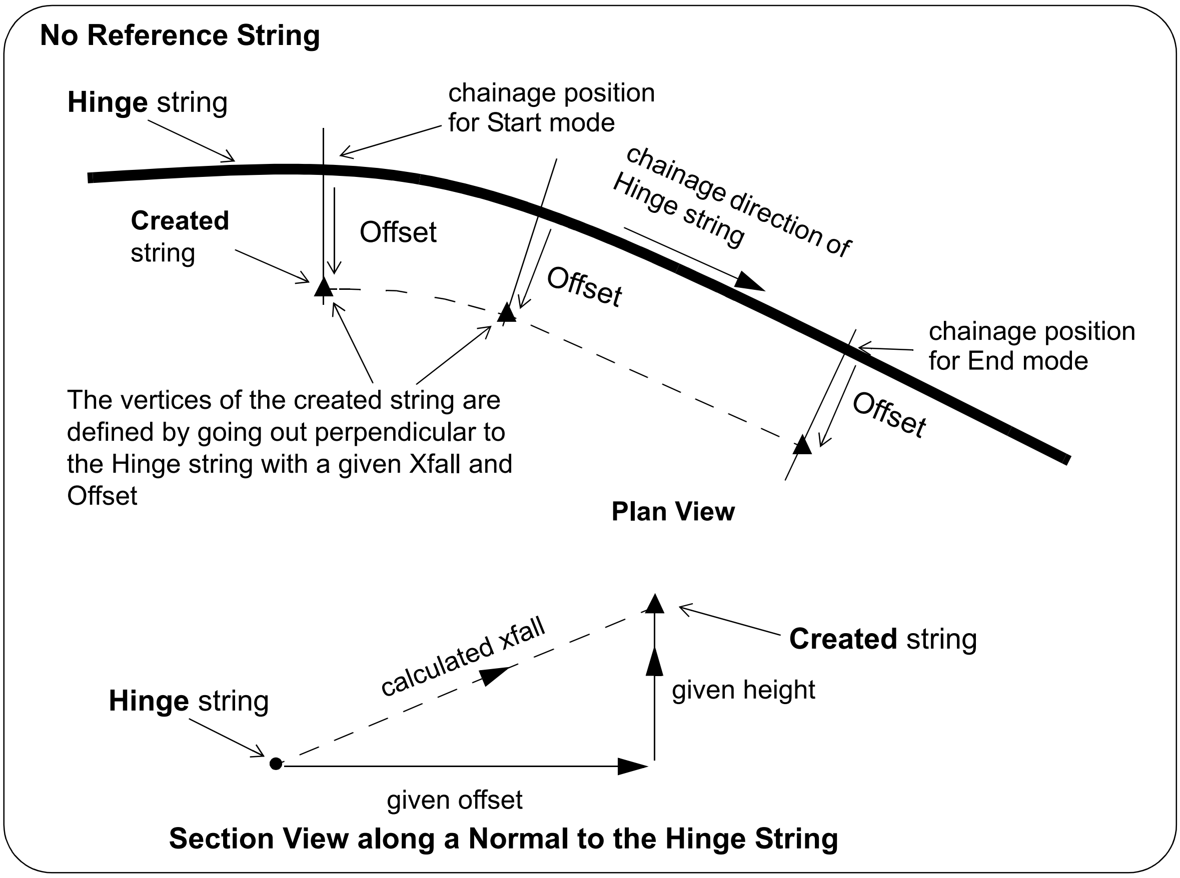
Reference String NOT Selected

If a **Reference** string is **NOT** selected then the **Start mode** and **End Mode** refer to the **Hinge** string and at each chainage ch, a vertex of a new string is created by going out **normal** (perpendicular) to the **Hinge string** for the given **Offset** and **Height**.

Positive Offset is defined as going to the right of the Hinge string when travelling in the direction of increasing chainage on the Hinge string. Offset can be positive or negative.

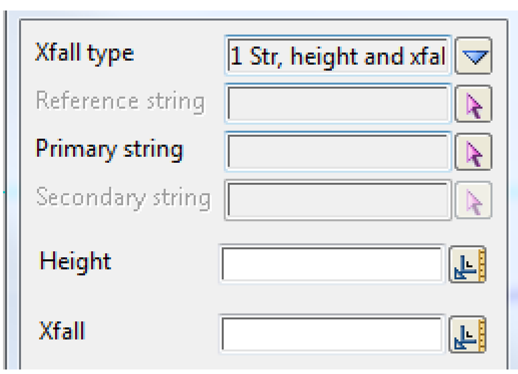
Height is **added** to the height on the Hinge string. Height can be positive or negative.

For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.



Continue to the next choice 20.8.11.1.3 1 String, Height and Xfall or return to 20.8.11.1 Xfall Types or 20.8.11 String by Xfall and Grade.

20.8.11.1.3 1 String, Height and Xfall



This option requires a **Hinge** string, a **Height** and **Xfall**, and an **optional Reference** string.

Reference String Selected

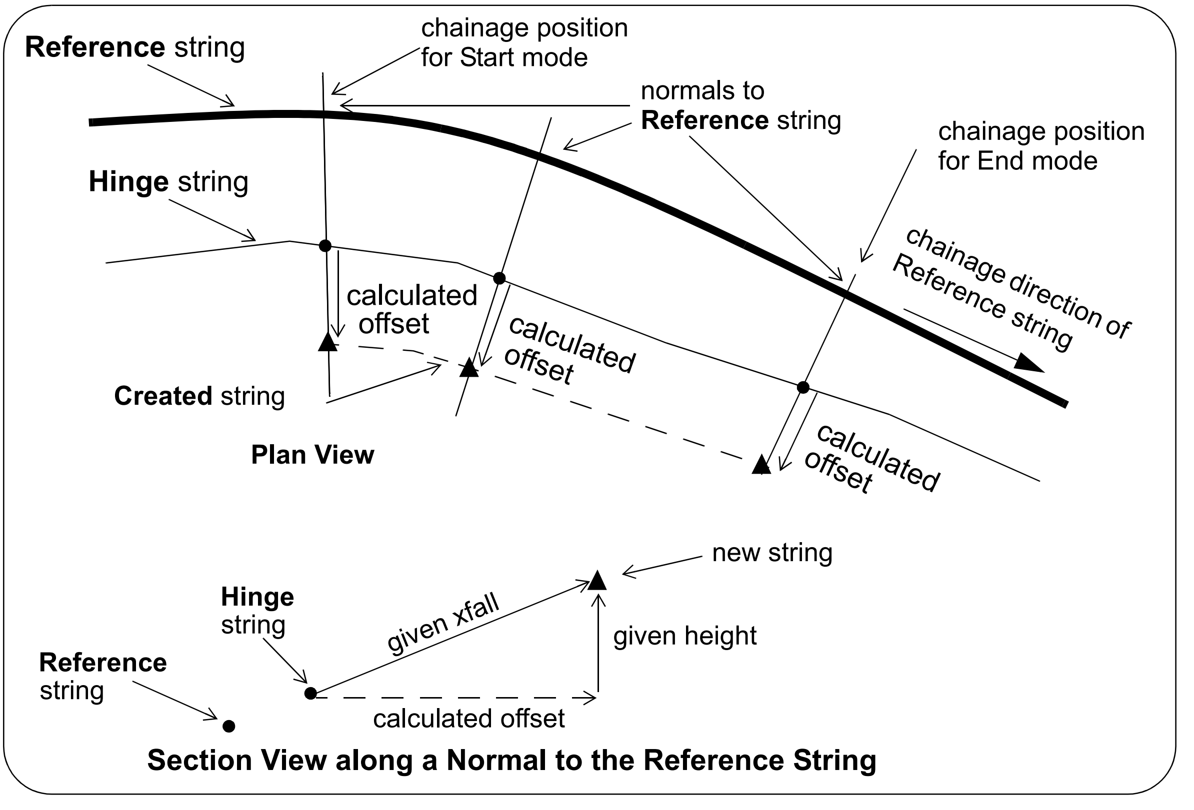
If a **Reference** string is selected then the **Start mode** and **End Mode** refer to **Reference** string and at each chainage ch between the Start mode and the End mode, a section is taken **normal** (perpendicular) to the **Reference string** and cutting the **Hinge** string.

A vertex of a new string is created by going out from the Hinge string along the section at the given **Height** and **Xfall**.

Height is **added** to the height on the Hinge string. Height can be positive or negative.

Positive Xfall is in the upward direction. Xfall can be positive or negative.

For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.



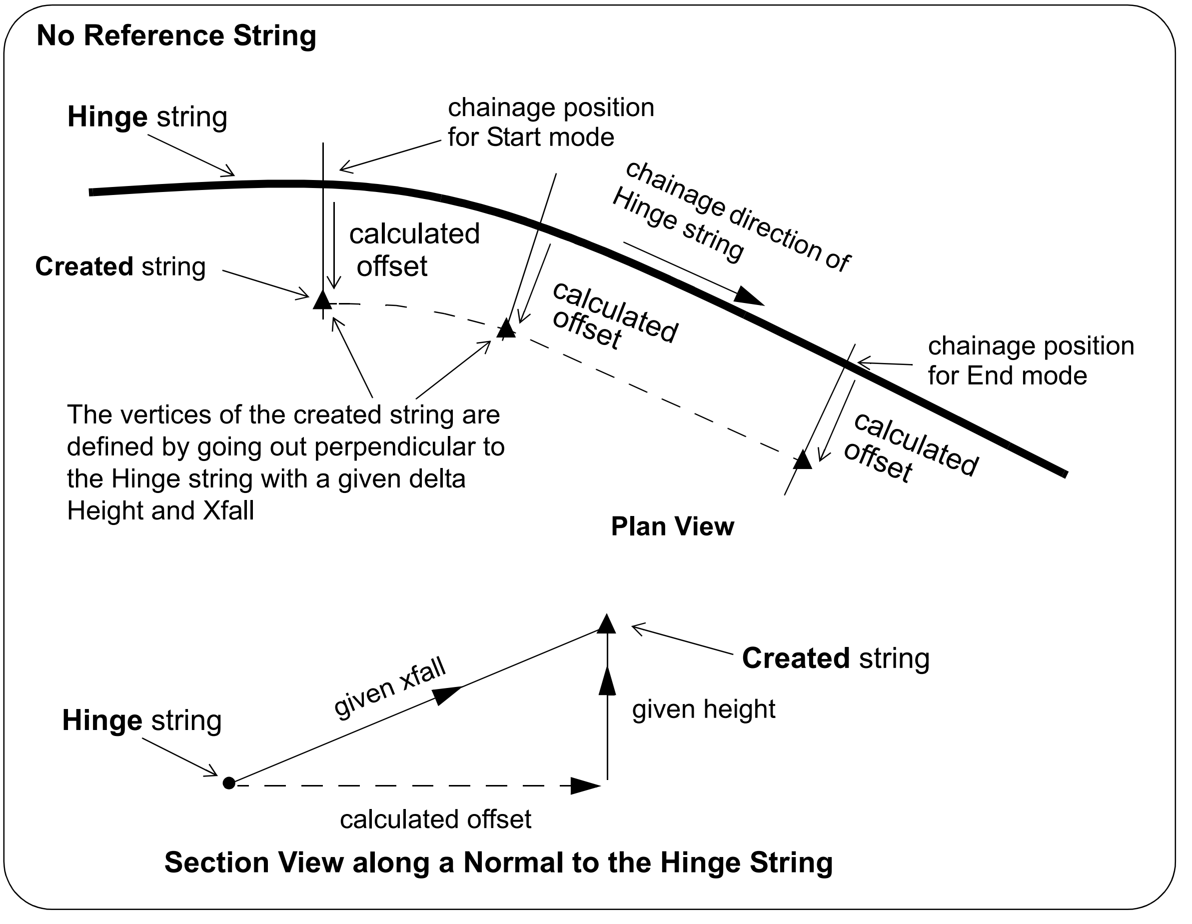
Reference String NOT Selected

If a **Reference** string is **NOT** selected then the **Start mode** and **End Mode** refer to the **Hinge** string and at each chainage ch, a vertex of a new string is created by going out **normal** (perpendicular) to the **Hinge string** for the given **Height** and **Xfall**.

Height is **added** to the height at that chainage on the Hinge string. Height can be positive or negative.

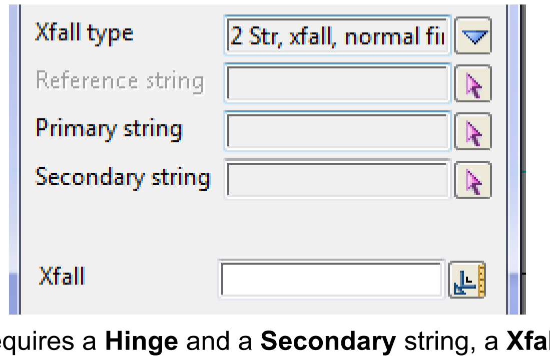
Positive Xfall is in the upward direction. Xfall can be positive or negative.

For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.



Continue to the next choice 20.8.11.1.4 2 Strings, Xfall and Normal to Reference String or Hinge String or return to 20.8.11.1 Xfall Types or 20.8.11 String by Xfall and Grade.

20.8.11.1.4 2 Strings, Xfall and Normal to Reference String or Hinge String



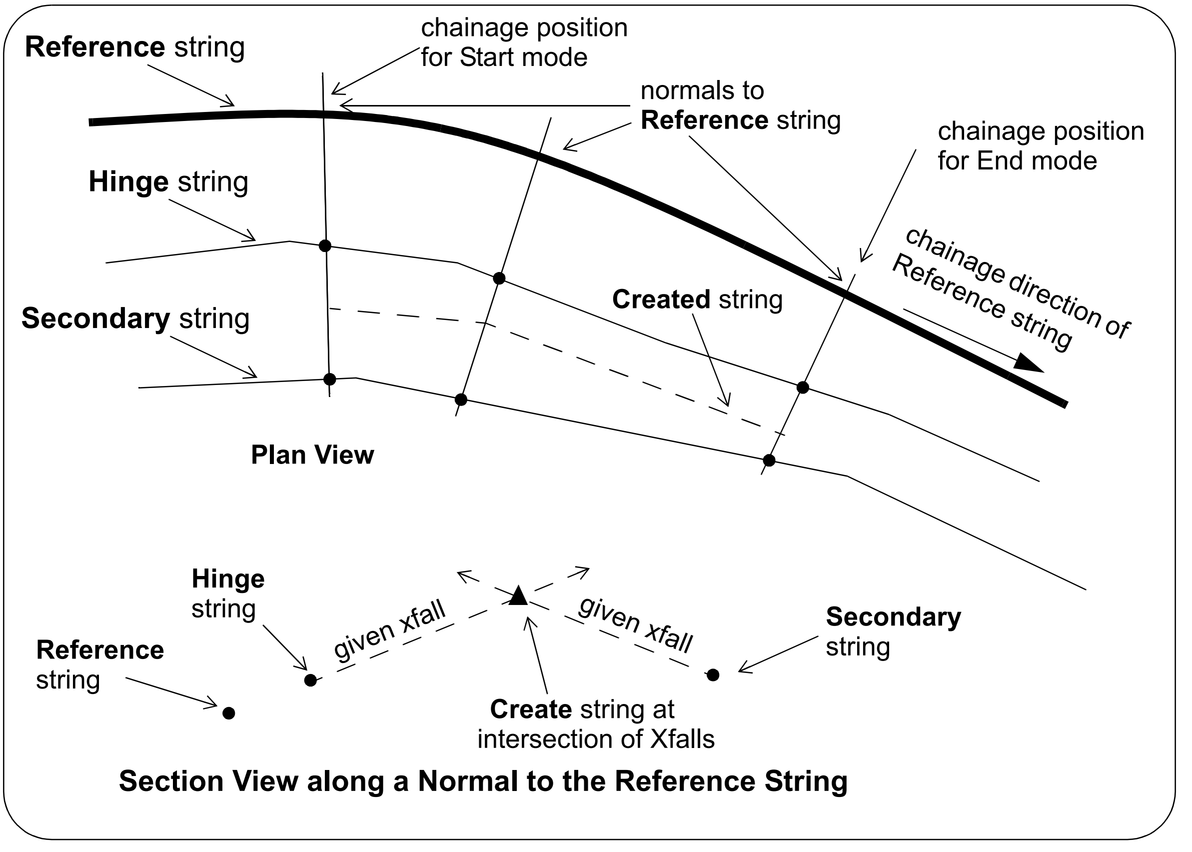
This option requires a **Hinge** and a **Secondary** string, a **Xfall**, and an **option Reference** string.

Reference String Selected

If a **Reference** string is selected then the **Start mode** and **End Mode** refer to **Reference** string and at each chainage (ch) between the Start mode and the End mode, a section is taken **normal** (perpendicular) to the **Reference string** and cutting the **Hinge** string and the **Secondary** strings.

On this section, the **Xfall** is taken from the Hinge and Secondary strings and a vertex of a new string created where the Xfalls intersect.

For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.

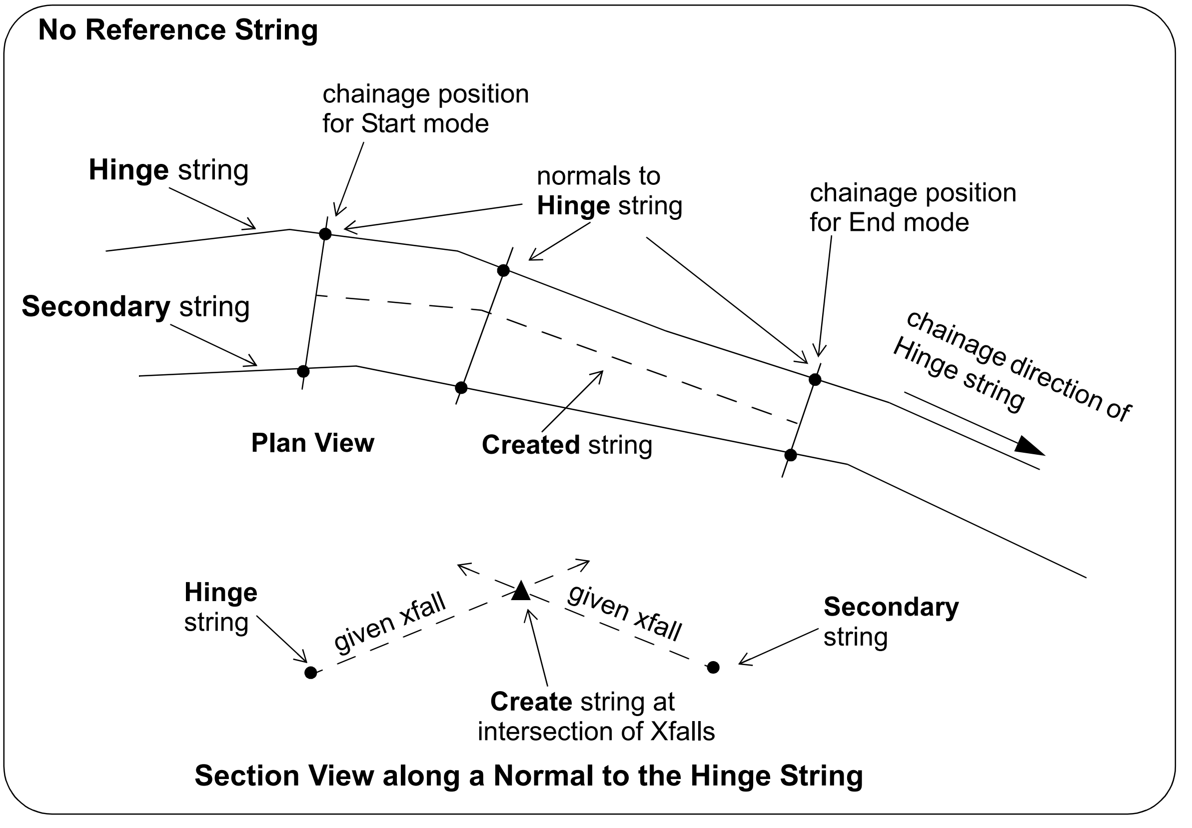


Reference String NOT Selected

If a **Reference** string is **NOT** selected then the **Start mode** and **End Mode** refer to **Hinge** string and at each chainage (ch), a section is taken **normal** (perpendicular) to the **Hinge string** and cutting the **Secondary** strings.

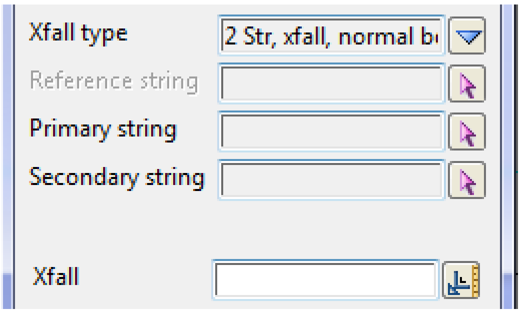
On this section, the **Xfall** is taken from the Hinge and Secondary strings and a vertex of a new string created where the Xfalls intersect.

For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.



Continue to the next choice 20.8.11.1.5 2 Strings, Xfall and Normal to Both or return to 20.8.11.1 Xfall Types or 20.8.11 String by Xfall and Grade.

20.8.11.1.5 2 Strings, Xfall and Normal to Both



This option requires a **Hinge** and a **Secondary** string, a **Xfall**, and an **option Reference** string.

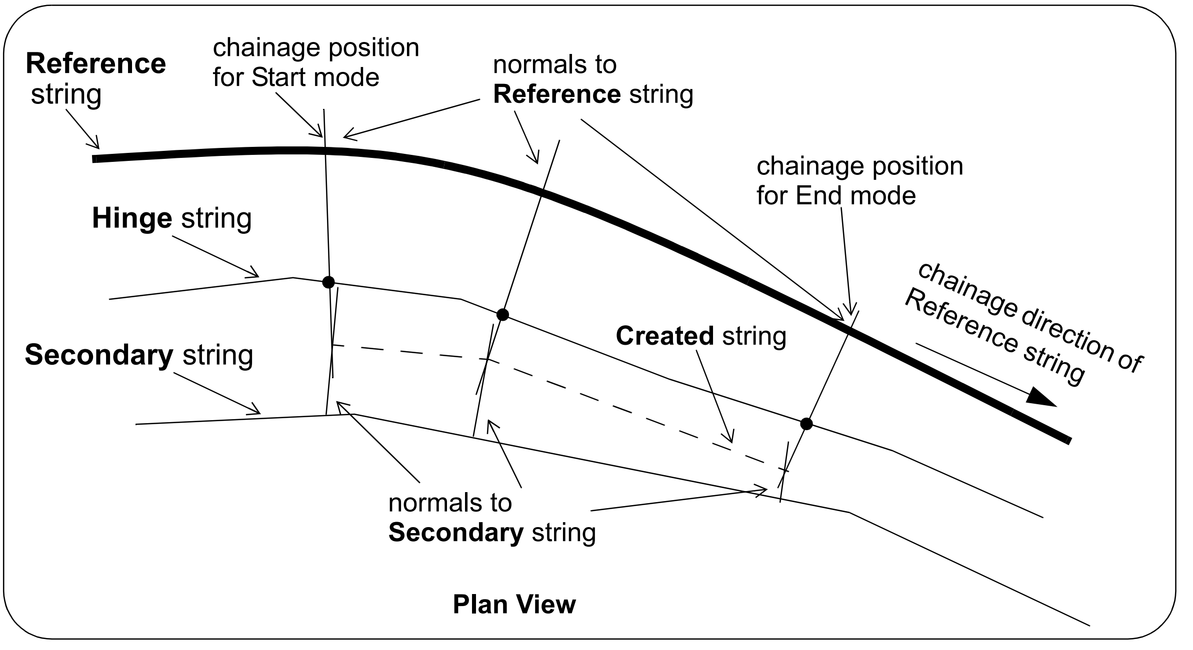
Reference String Selected

If a **Reference** string is selected then the **Start mode** and **End Mode** refer to **Reference** string and at each chainage ch between the Start mode and the End mode, a section is taken **normal** (perpendicular) to the **Reference string** and cutting the **Hinge** string, and the **Xfall** is taken from the cut with the Hinge string and along the section.

This is then intersected with a section **normal** to the **Secondary** string and with the given **Xfall**.

The new string is formed from the intersection points.

Note - it may not be possible to find such an intersection point.



Reference String NOT Selected

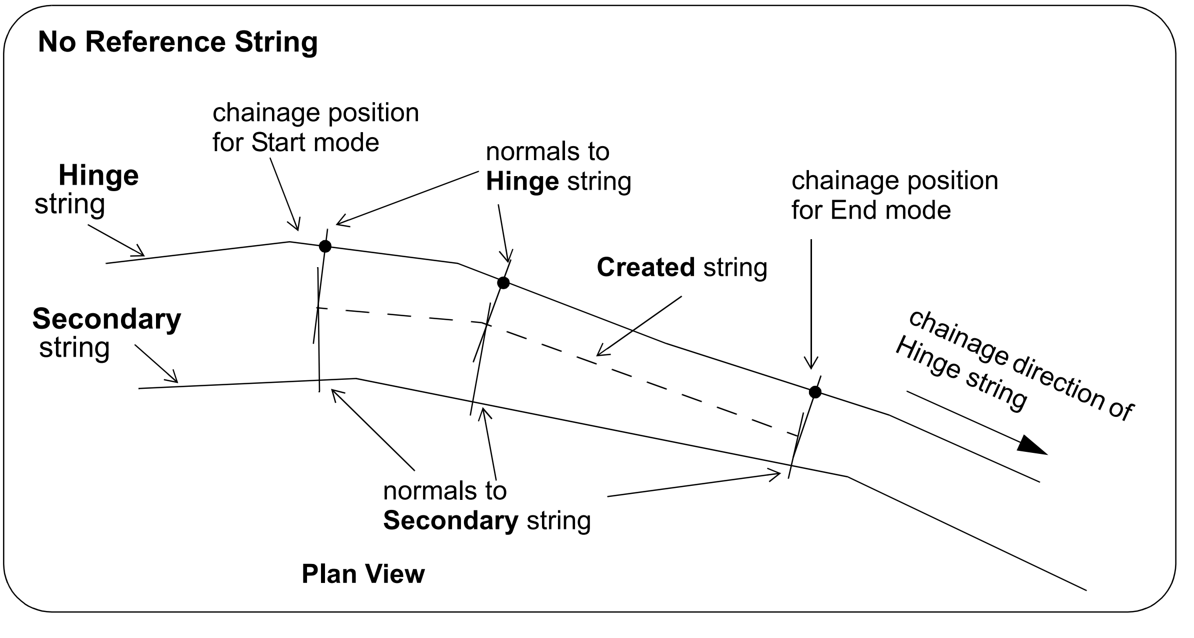
If a **Reference** string is **NOT** selected then the **Start mode** and **End Mode** refer to the **Hinge** string.

At each chainage ch between the Start mode and the End mode, a section is taken **normal** (perpendicular) to the **Hinge string** and the **Xfall** is taken along this section.

This is then intersected with a section **normal** to the **Secondary** string and with the given **Xfall**.

The new string is formed from the intersection points.

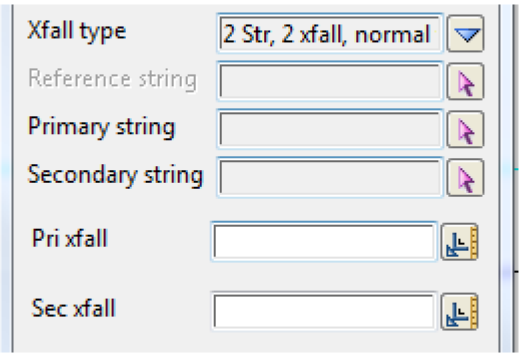
Note - it may not be possible to find such an intersection point.



Positive Xfall is in the upward direction. Xfall can be positive or negative.

Continue to the next choice 20.8.11.1.6 2 Strings, 2 Xfalls and Normal to Reference String or Hinge String or return to 20.8.11.1 Xfall Types or 20.8.11 String by Xfall and Grade.

20.8.11.1.6 2 Strings, 2 Xfalls and Normal to Reference String or Hinge String



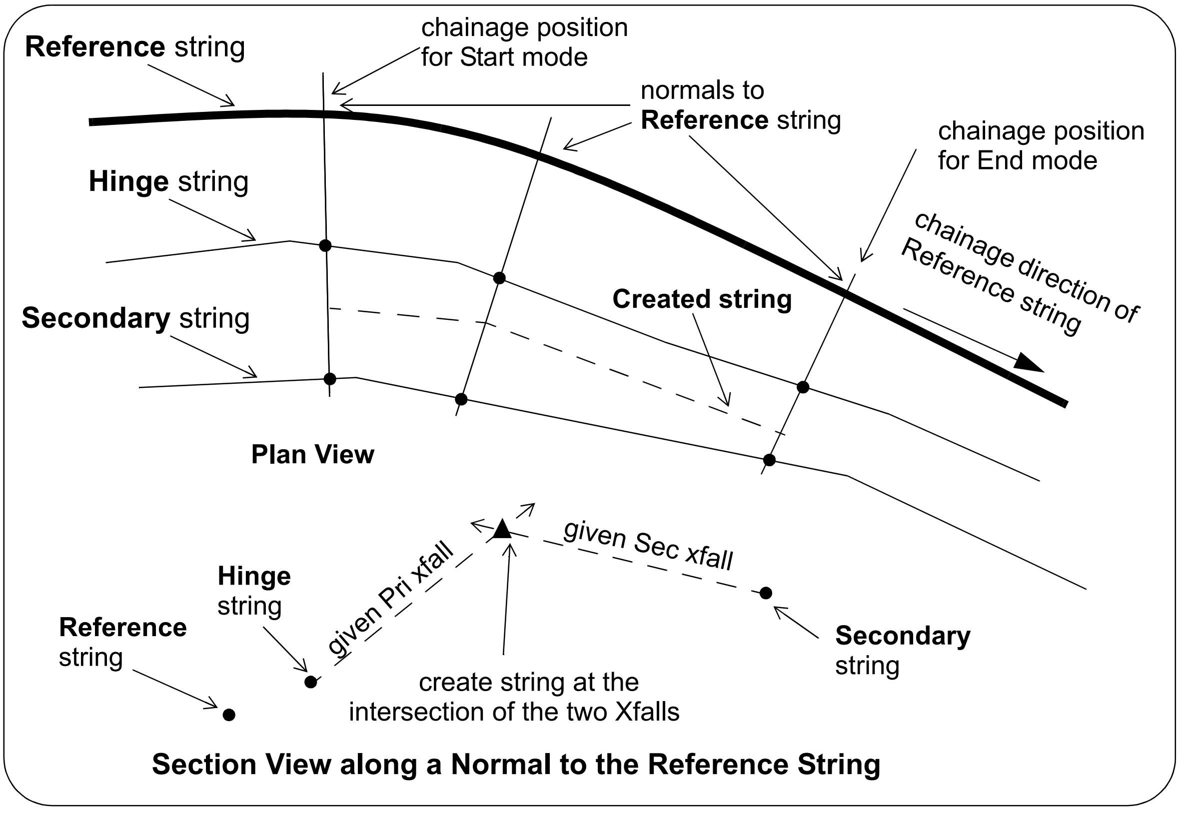
This option requires a **Hinge** and a **Secondary** string, two **Xfalls**, and an **option Reference** string.

Reference String Selected

If a **Reference** string is selected then the **Start mode** and **End Mode** refer to the **Reference** string and at each chainage (ch) between the Start mode and the End mode, a section is taken **normal** (perpendicular) to the **Reference string** and cutting the **Hinge** string and the **Secondary** strings.

On this section, the xfall **Pri xfall** is taken from the Hinge string, and xfall **Sec xfall** from the Secondary string, and a vertex of a new string created where the two xfalls intersect.

For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.

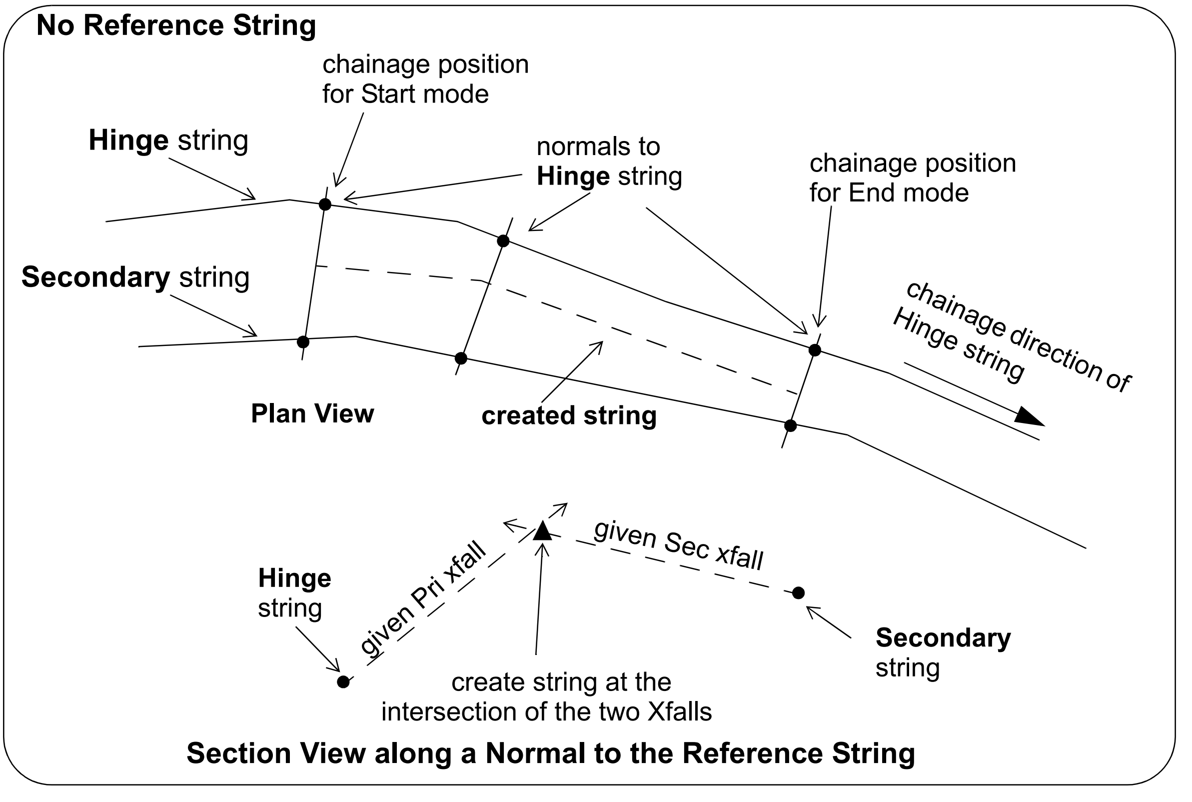


Reference String NOT Selected

If a **Reference** string is **NOT** selected then the **Start mode** and **End Mode** refer to **Hinge** string and at each chainage (ch), a section is taken **normal** (perpendicular) to the **Hinge string** and cutting the **Secondary** strings.

On this section, the xfall **Pri xfall** is taken from the Hinge string, and xfall **Sec xfall** from the Secondary string, and a vertex of a new string created where the two xfalls intersect.

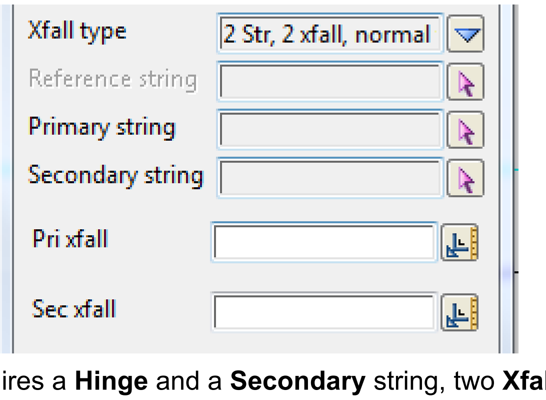
For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.



Positive Xfall is in the upward direction. Xfall can be positive or negative.

Continue to the next choice 20.8.11.1.7 2 Strings, 2 Xfalls and Normal to Both or return to 20.8.11.1 Xfall Types or 20.8.11 String by Xfall and Grade.

20.8.11.1.7 2 Strings, 2 Xfalls and Normal to Both



This option requires a **Hinge** and a **Secondary** string, two **Xfalls**, and an **option Reference** string.

Reference String Selected

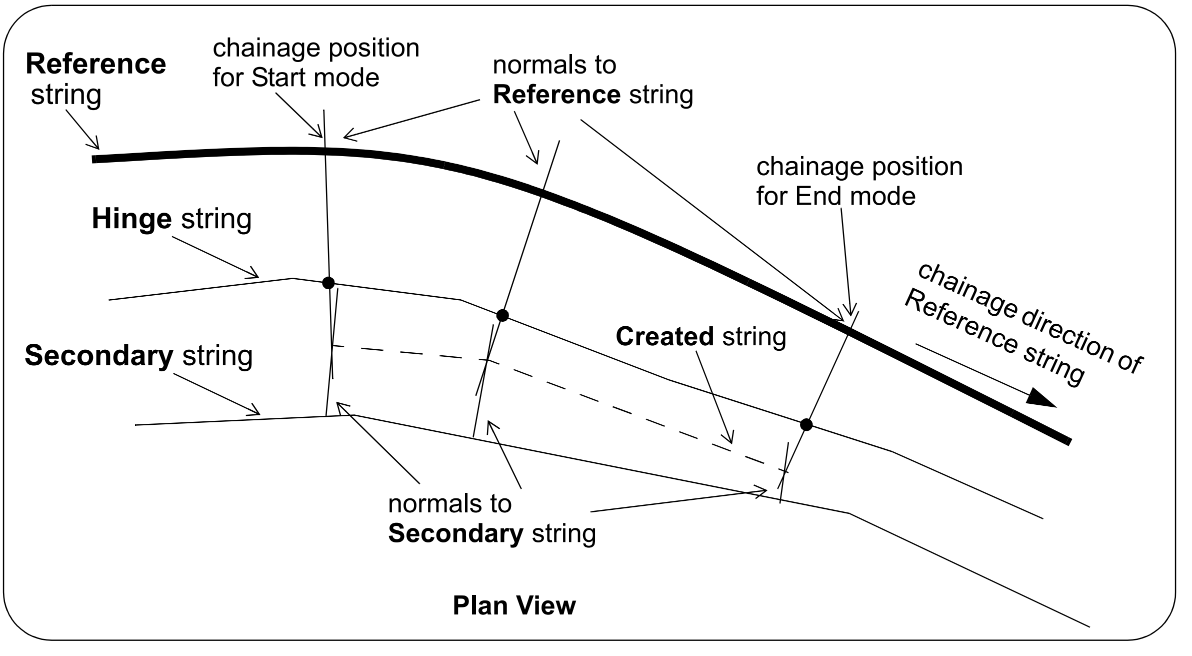
If a **Reference** string is selected then the **Start mode** and **End Mode** refer to **Reference** string and at each chainage ch between the Start mode and the End mode, a section is taken **normal** (perpendicular) to the **Reference string** and cutting the **Hinge** string, and the xfall **Pri xfall** is taken from the cut with the Hinge string and along the section.

This is then intersected with a section **normal** to the **Secondary** string and with the xfall **Sec xfall**.

The new string is formed from the intersection points.

Note - it may not be possible to find such an intersection point.

For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.



Reference String NOT Selected

If a **Reference** string is **NOT** selected then the **Start mode** and **End Mode** refer to the **Hinge** string.

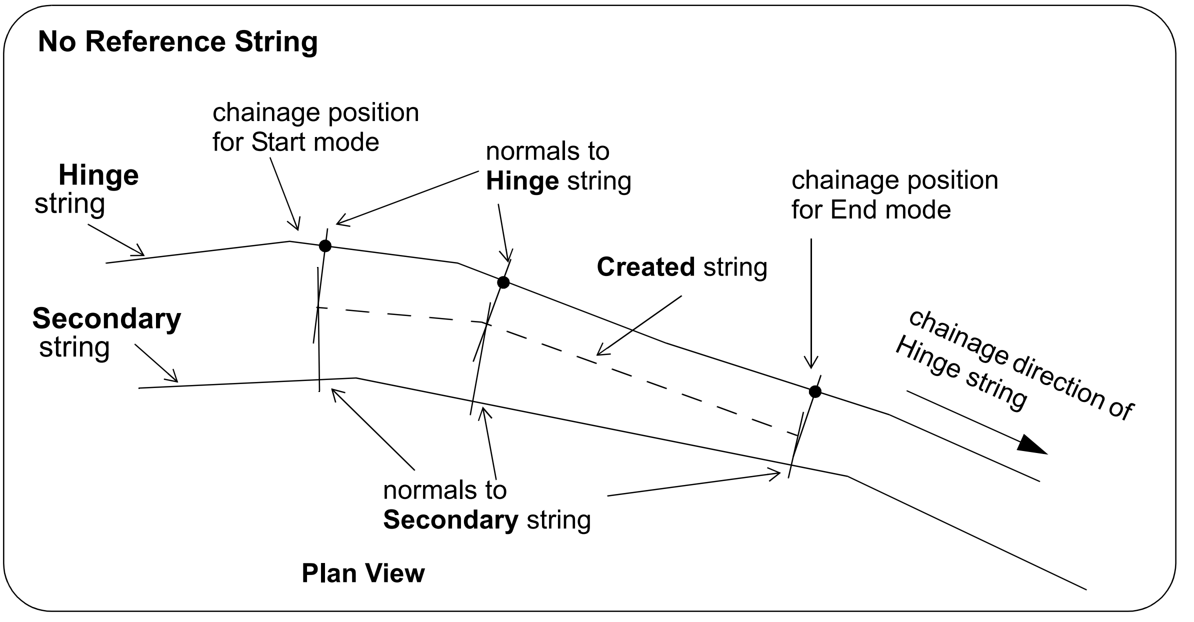
At each chainage ch between the Start mode and the End mode, a section is taken **normal** (perpendicular) to the **Hinge string** and the xfall **Pri xfall** is taken along this section.

This is then intersected with a section **normal** to the **Secondary** string and with the xfall **Sec xfall**.

The new string is formed from the intersection points.

Note - it may not be possible to find such an intersection point.

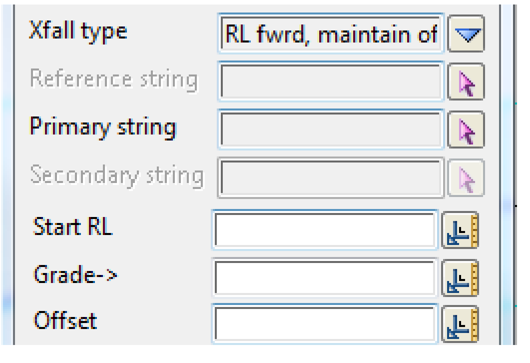
For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.



Positive Xfall is in the upward direction. Xfall can be positive or negative.

Continue to the next choice 20.8.11.1.8 1 String, Start RL and Grade, Fixed Offset or return to 20.8.11.1 Xfall Types or 20.8.11 String by Xfall and Grade.

20.8.11.1.8 1 String, Start RL and Grade, Fixed Offset



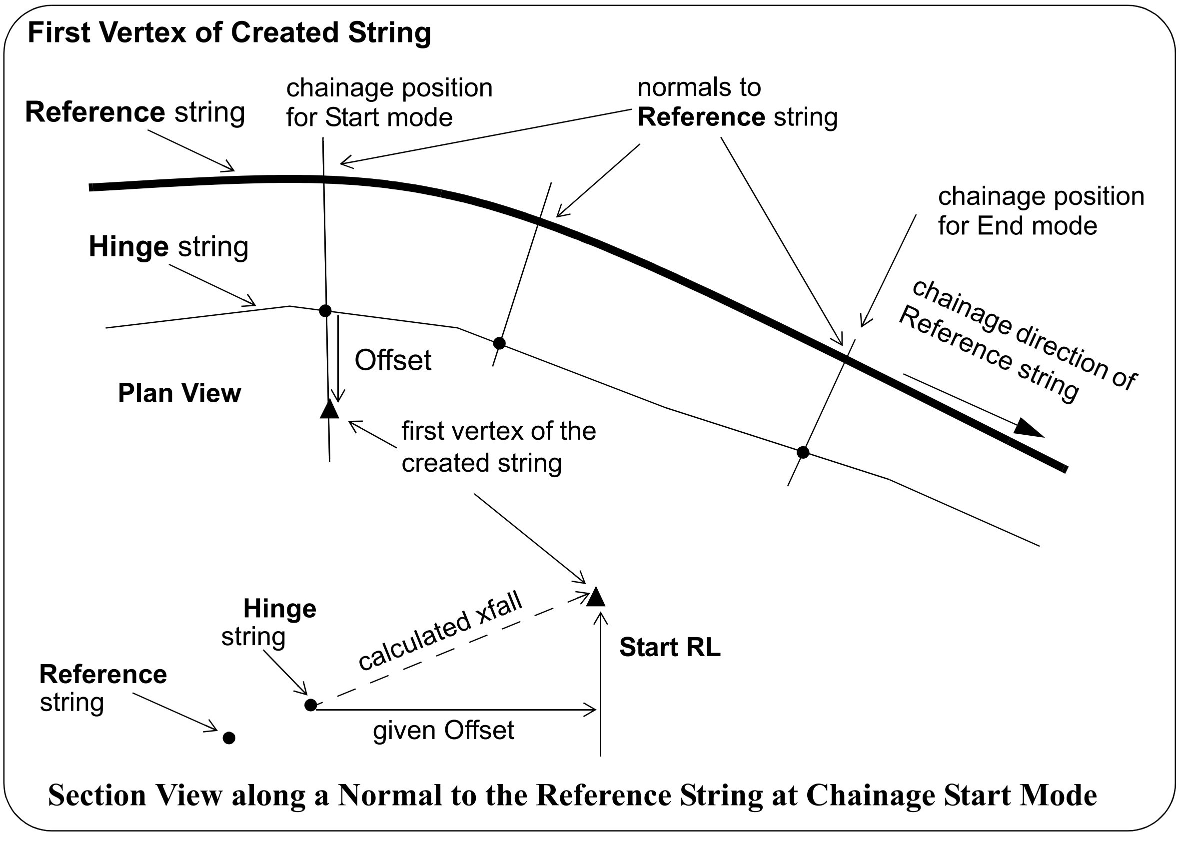
This option requires a **Hinge** string, a **Start RL** and forward **Grade**, an **Offset**, and an **optional Reference** string.

Reference String Selected

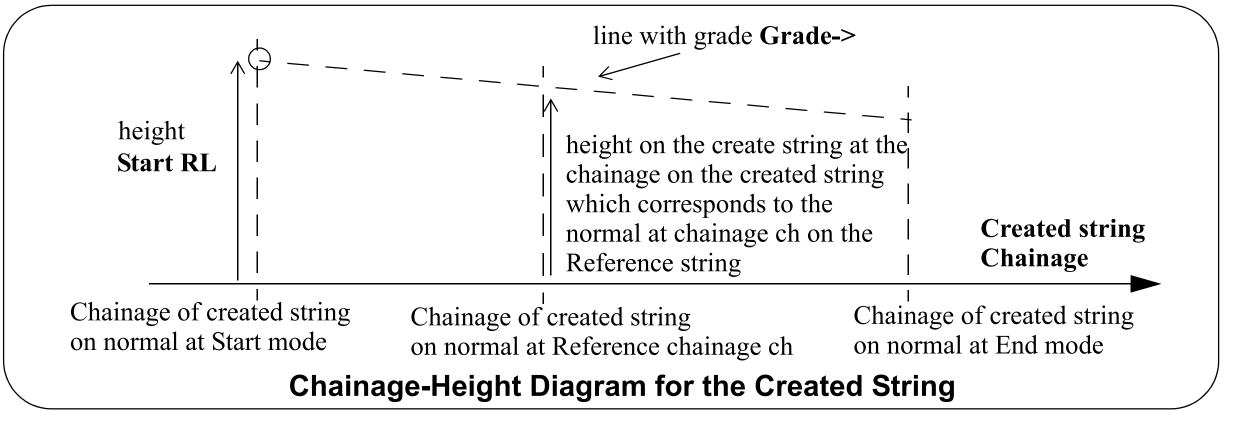
If a **Reference** string is selected then the **Start mode** and **End Mode** refer to the **Reference** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Reference string** and cutting the **Hinge** string. The vertices of the created string lie on these normals.

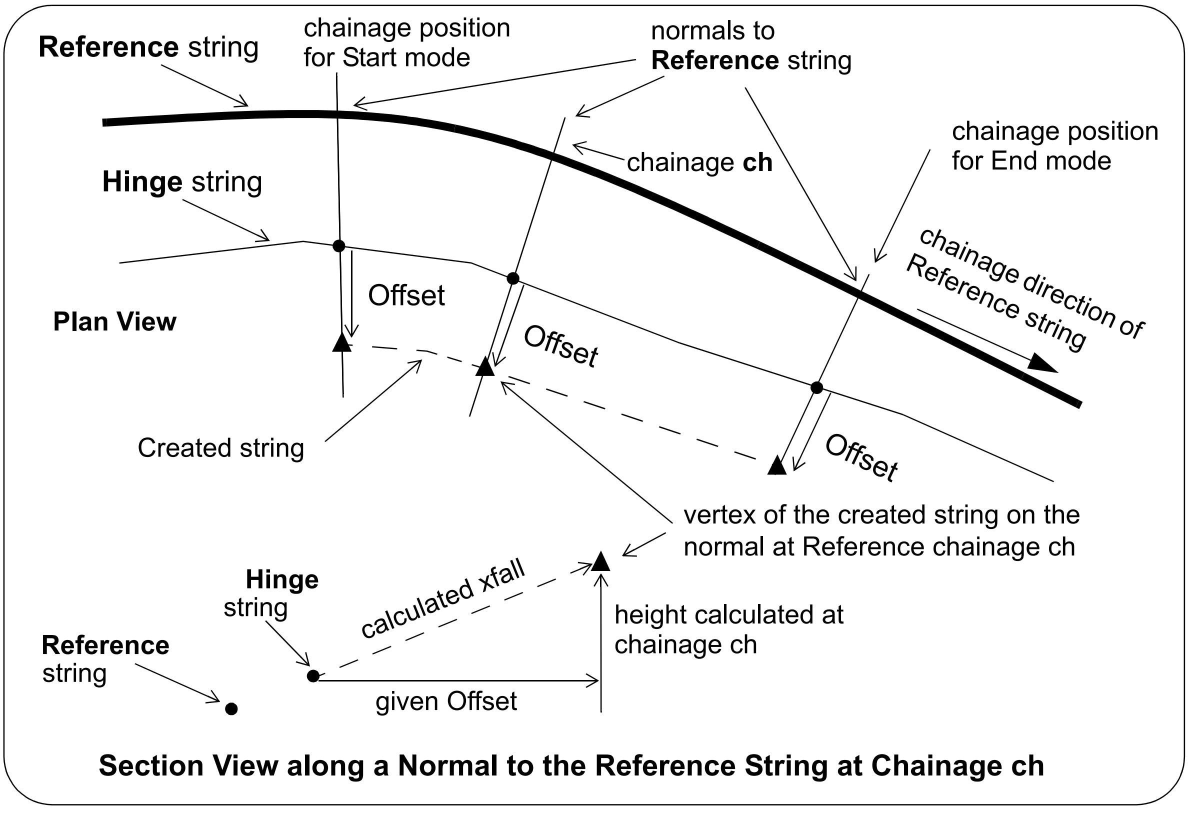
The **plan position** of the created vertices are given by going out from the Hinge string along the normal section for the given **Offset.**

The **first vertex** of the created string is on the normal to the Reference string at chainage **Start mode**, with an offset from the **Hinge** string of **Offset** and with a **height** of **Start RL**.



The **heights** of the subsequent vertices of the created string are defined in (created string chainage, height) space, and lie on the line with grade ***Grade->*** and going through the first vertex.





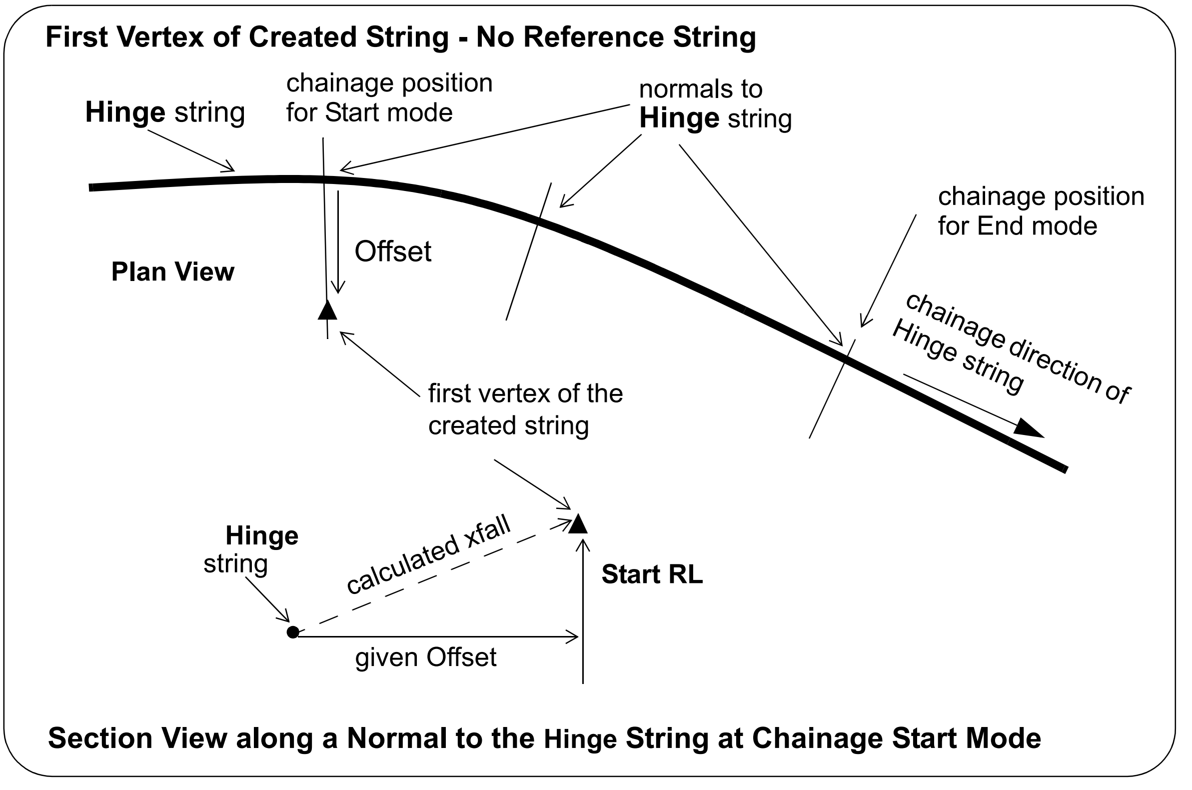
For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.

Reference String NOT Selected

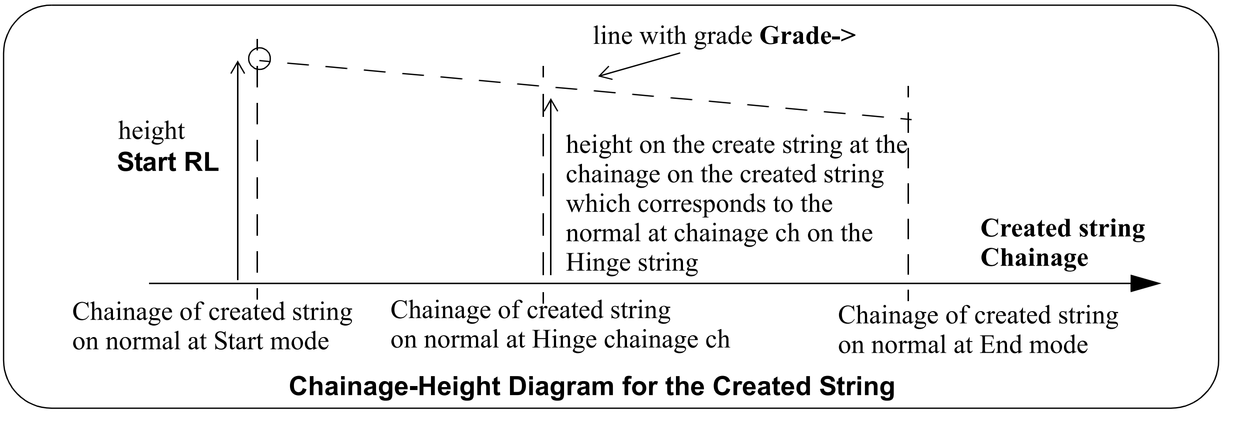
If a **Reference** string is **NOT** selected then the **Start mode** and **End Mode** refer to the **Hinge** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Hinge string**. The vertices of the created string lie on these normals.

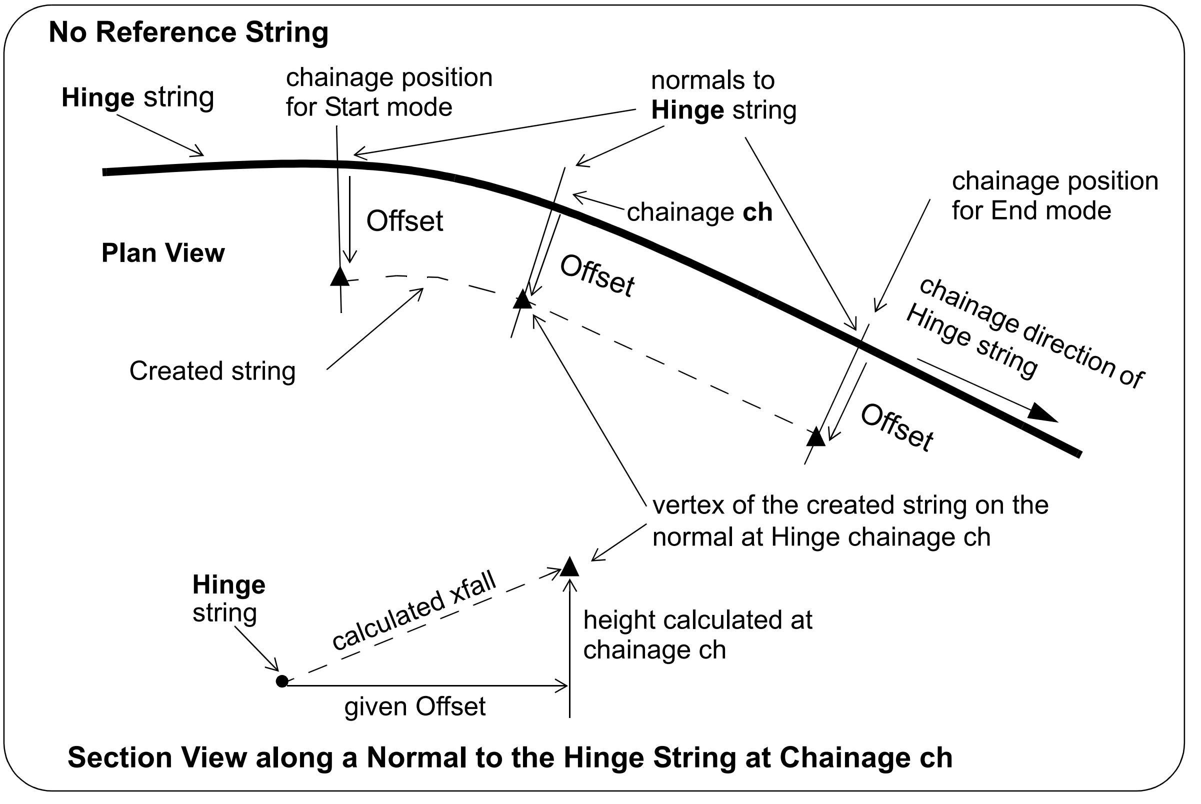
The **plan position** of the created vertices are given by going out from the Hinge string along the normal section for the given **Offset.**

The **first vertex** of the created string is on the normal to the Hinge string at chainage **Start mode**, with an offset from the **Hinge** string of **Offset**, and with a **height** of **Start RL**.



The **heights** of the subsequent vertices of the created string are defined in (created string chainage, height) space, and lie on the line with grade ***Grade->*** and going through the first vertex.

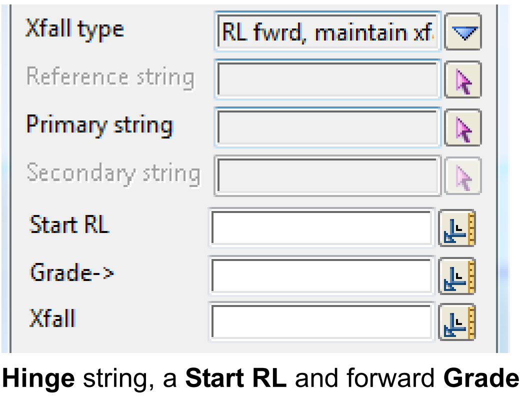




For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.

Continue to the next choice 20.8.11.1.9 1 String, Start RL and Grade, Fixed Xfall or return to 20.8.11.1 Xfall Types or 20.8.11 String by Xfall and Grade.

20.8.11.1.9 1 String, Start RL and Grade, Fixed Xfall



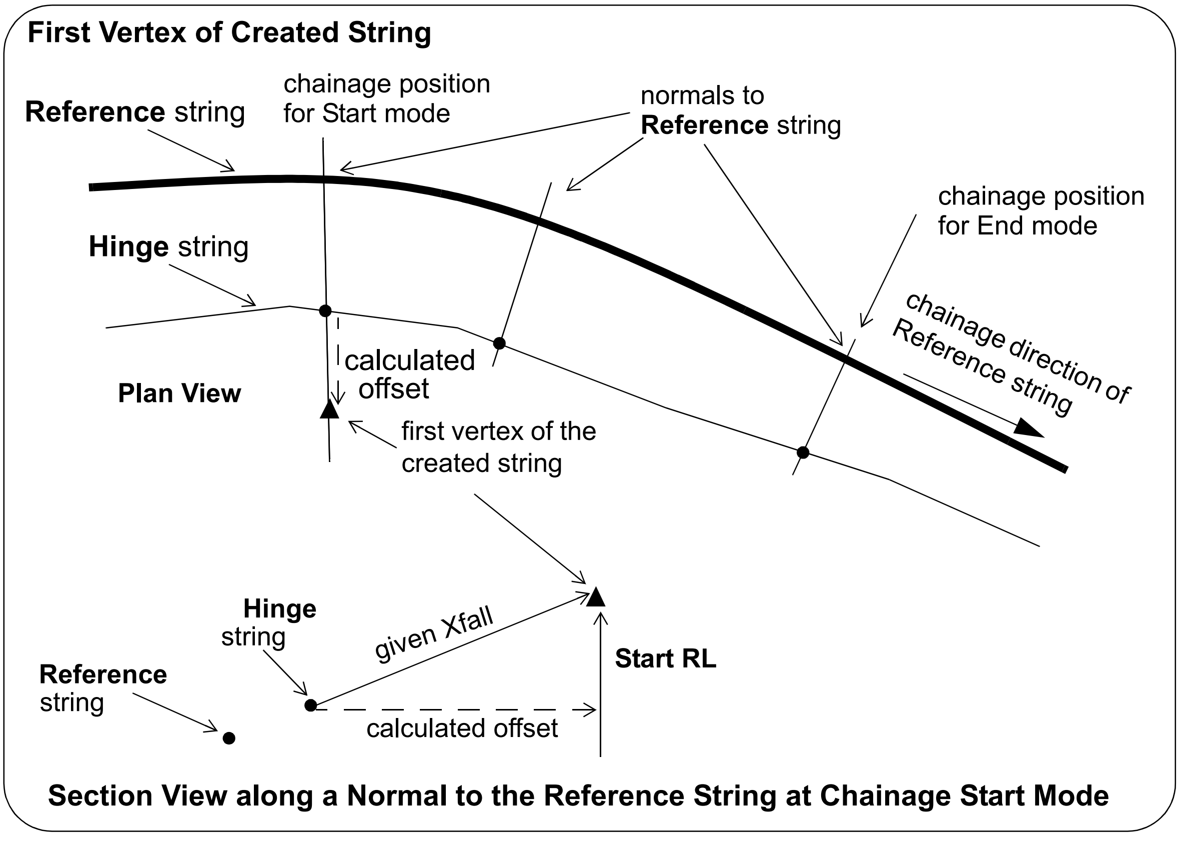
This option requires a **Hinge** string, a **Start RL** and forward **Grade**, a **Xfall**, and an **optional Reference** string.

Reference String Selected

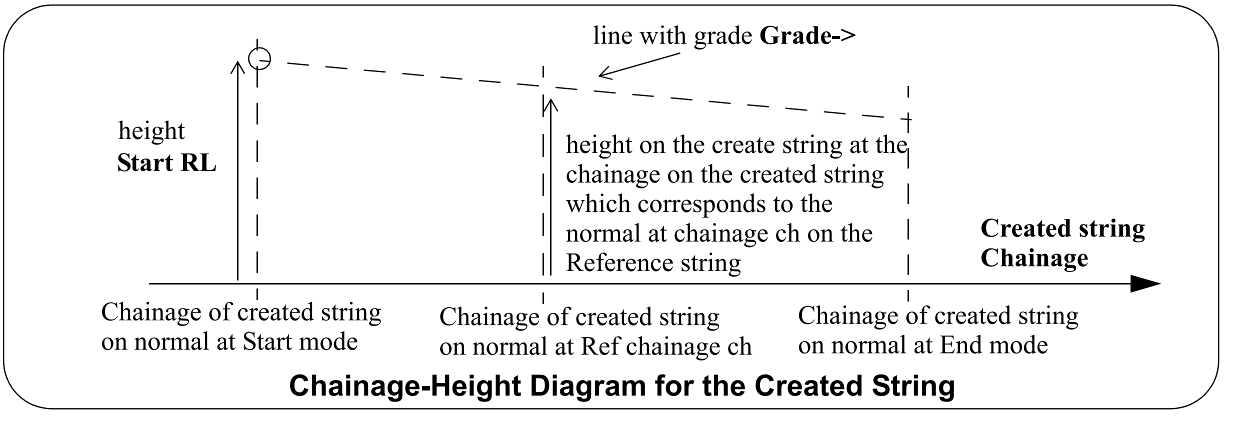
If a **Reference** string is selected then the **Start mode** and **End Mode** refer to the **Reference** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Reference string** and cutting the **Hinge** string. The vertices of the created string lie on these normals.

The **first vertex** of the created string is on the normal to the Reference string and has a **height**

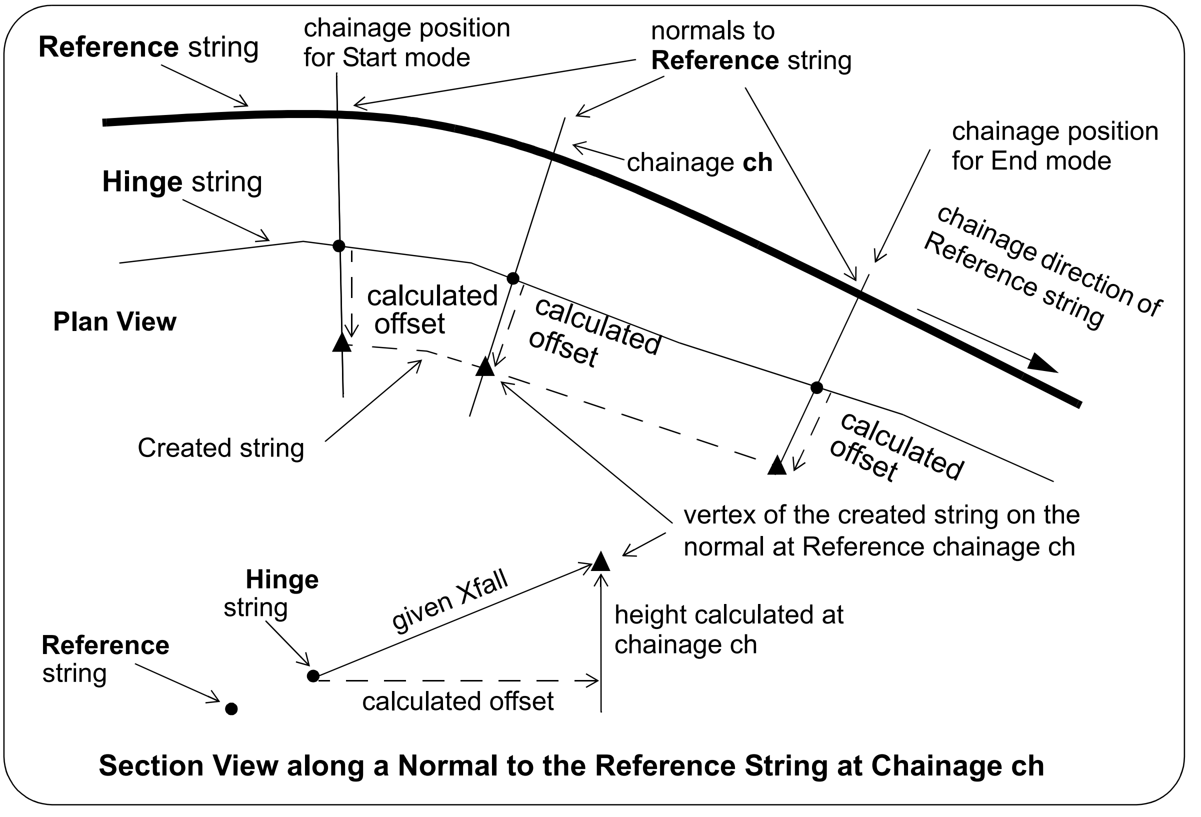
Start RL. The offset for the first vertex is calculated and is such that the given **Xfall** from Hinge string will give the vertex the height **Start RL**.



The **heights** of the subsequent vertices of the created string are defined in (created string chainage, height) space, and lie on the line with grade ***Grade->*** and going through the first vertex.



The offsets for the vertices are calculated and are such that the given **Xfall** from Hinge string will give the vertex the correct **height** for that chainage.

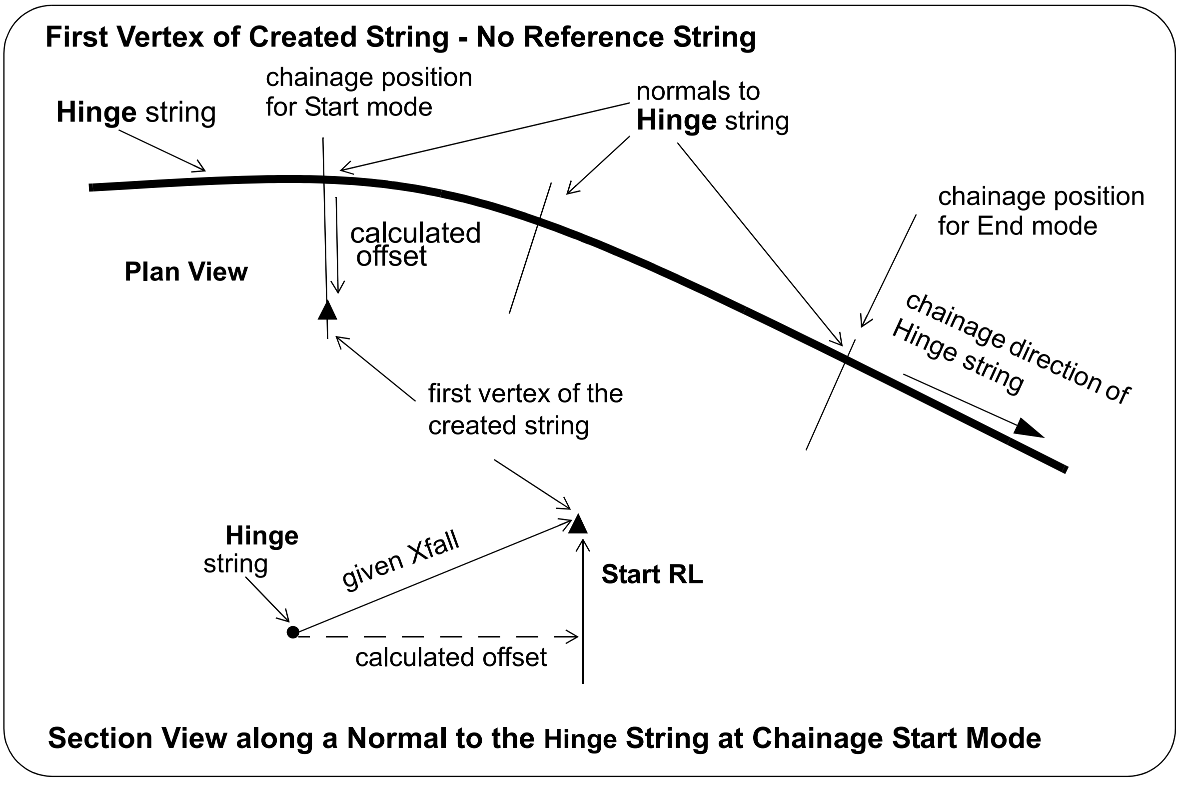


For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.

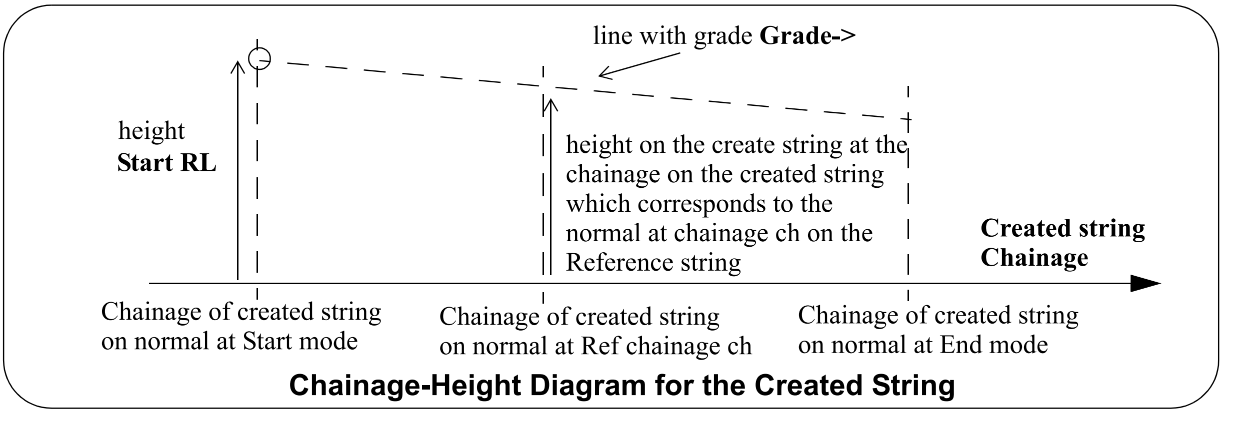
Reference String NOT Selected

If a **Reference** string is **NOT** selected then the **Start mode** and **End Mode** refer to the **Hinge** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Hinge string**. The vertices of the created string lie on these normals.

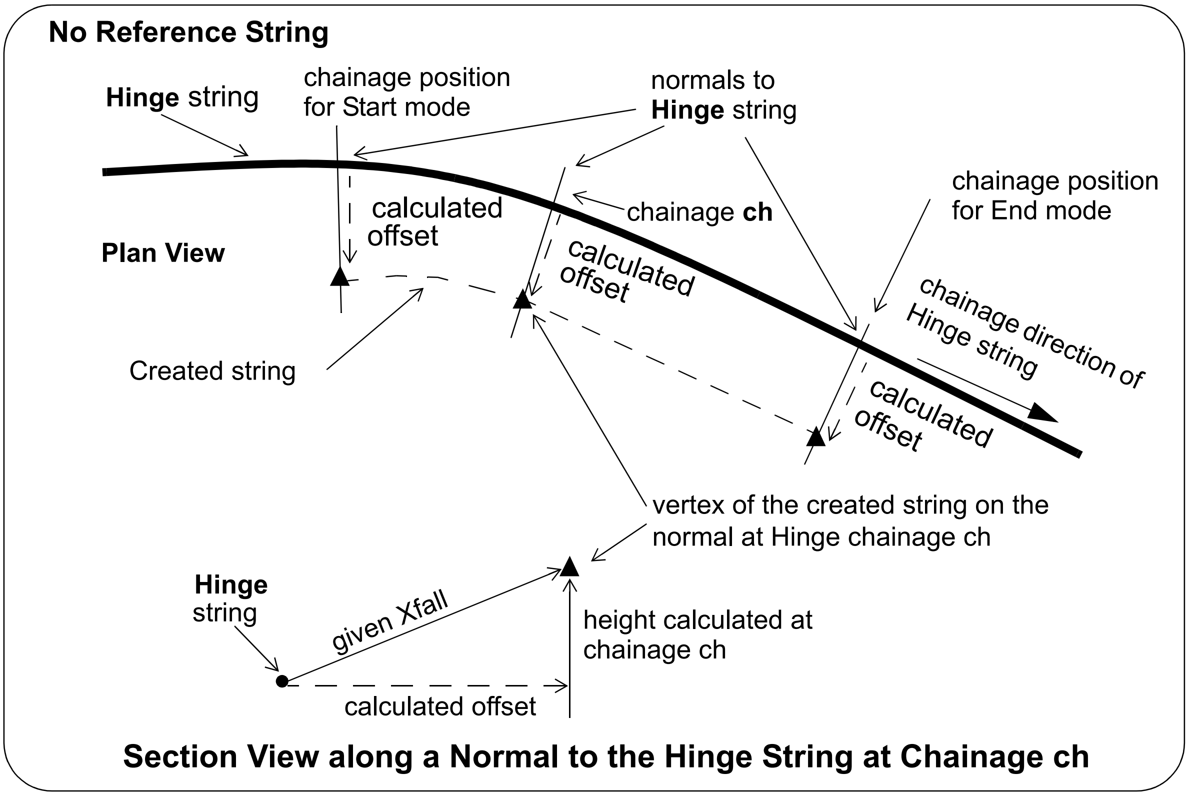
The **first vertex** of the created string is on the normal to the Hinge string and has a **height Start RL**. The offset for the first vertex is calculated and is such that the given **Xfall** from Hinge string will give the vertex the height **Start RL**.



The **heights** of the subsequent vertices of the created string are defined in (created string chainage, height) space, and lie on the line with grade ***Grade->*** and going through the first vertex.



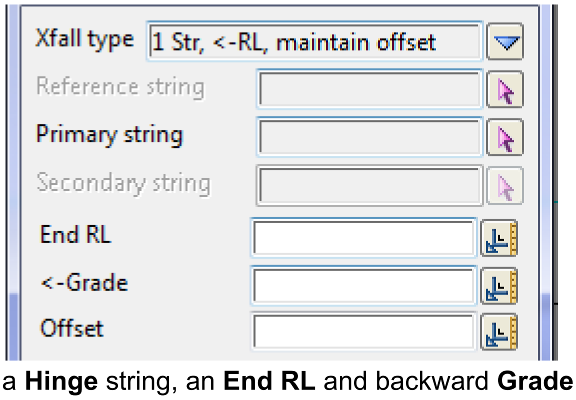
The offsets for the vertices are calculated and are such that the given **Xfall** from Hinge string will give the vertex the correct **height** for that chainage.



For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.

Continue to the next choice 20.8.11.1.10 1 String, End RL and Grade, Fixed Offset or return to 20.8.11.1 Xfall Types or 20.8.11 String by Xfall and Grade.

20.8.11.1.10 1 String, End RL and Grade, Fixed Offset



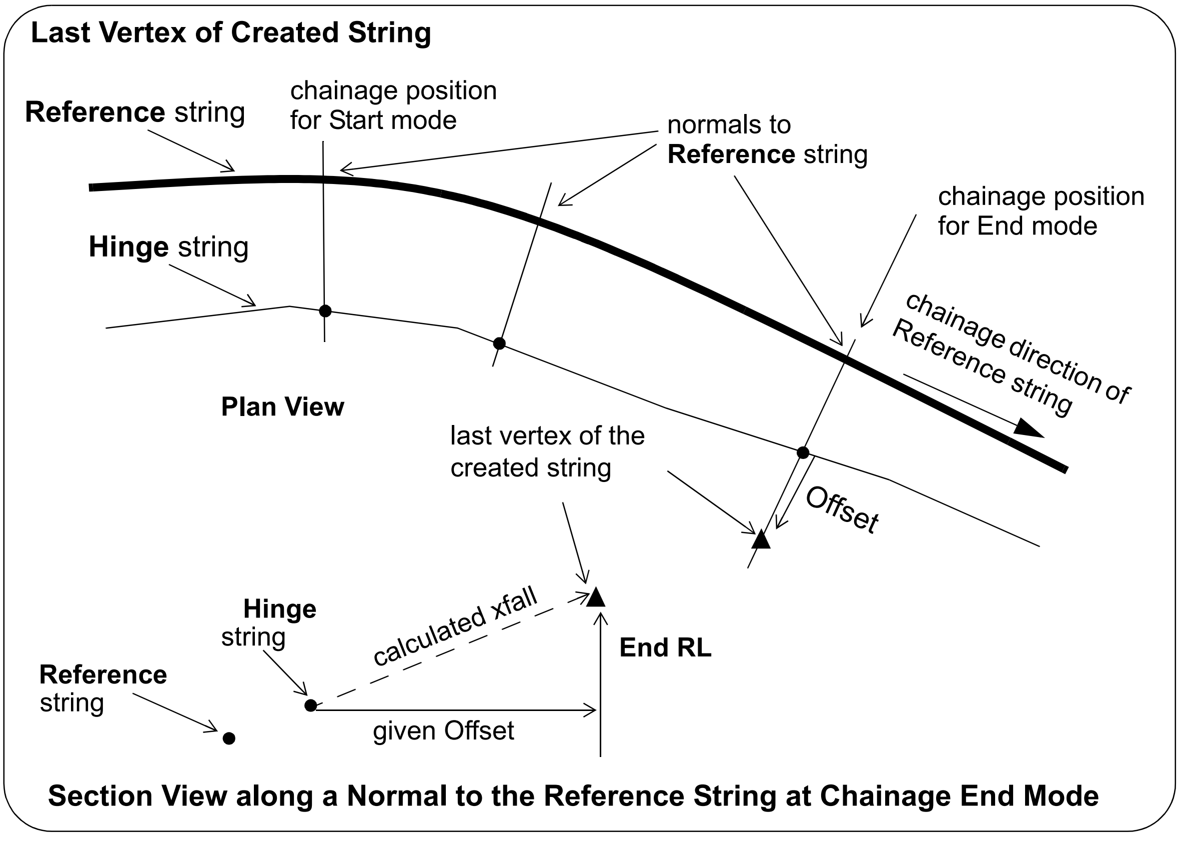
This option requires a **Hinge** string, an **End RL** and backward **Grade**, an **Offset**, and an **optional Reference** string.

Reference String Selected

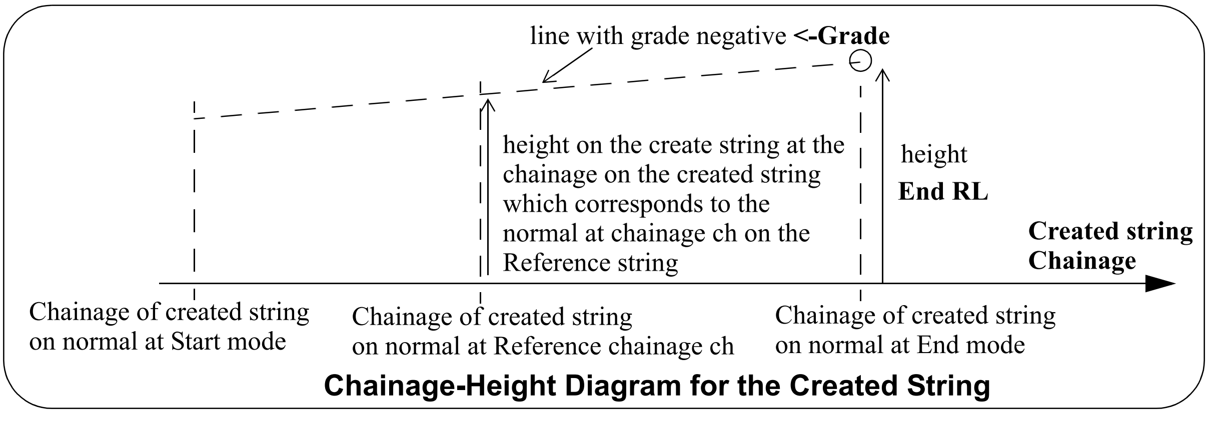
If a **Reference** string is selected then the **Start mode** and **End Mode** refer to the **Reference** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Reference string** and cutting the **Hinge** string. The vertices of the created string lie on these normals.

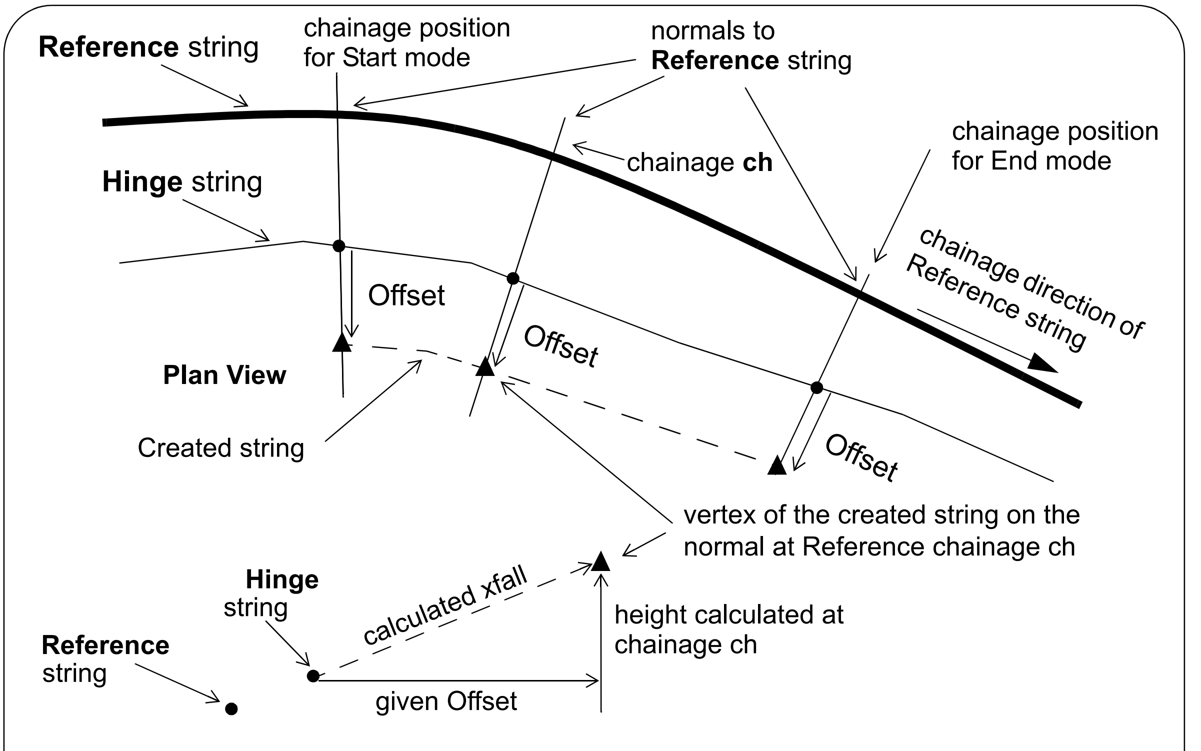
The **plan position** of the created vertices are given by going out from the Hinge string along the normal section for the given **Offset**.

The **last vertex** of the created string is on the normal to the Reference string at chainage **End mode**, with an offset from the **Hinge** string of **Offset** and with a **height** of **End RL**.



The **heights** of the previous vertices of the created string are defined in (created string chainage, height) space, and lie on the line with grade negative ***<-Grade*** and going through the last vertex.





Section View along a Normal to the Reference String at Chainage ch

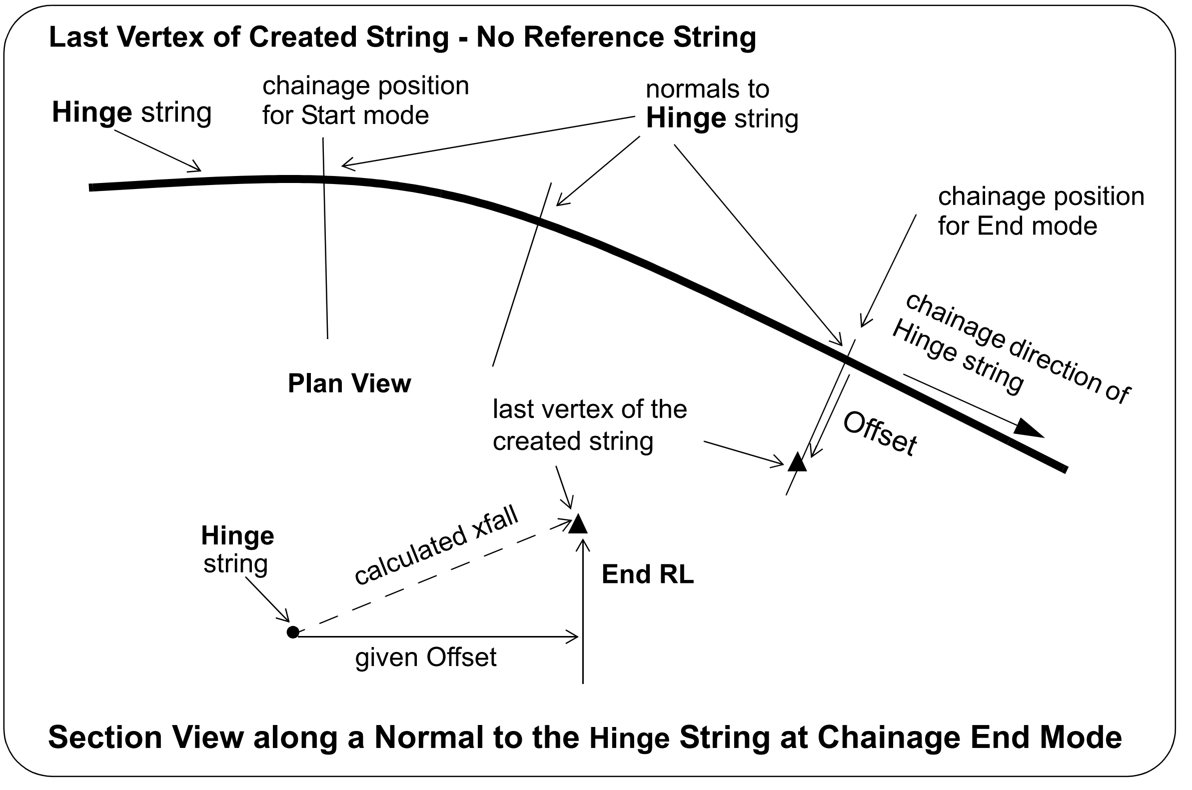
For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.

Reference String NOT Selected

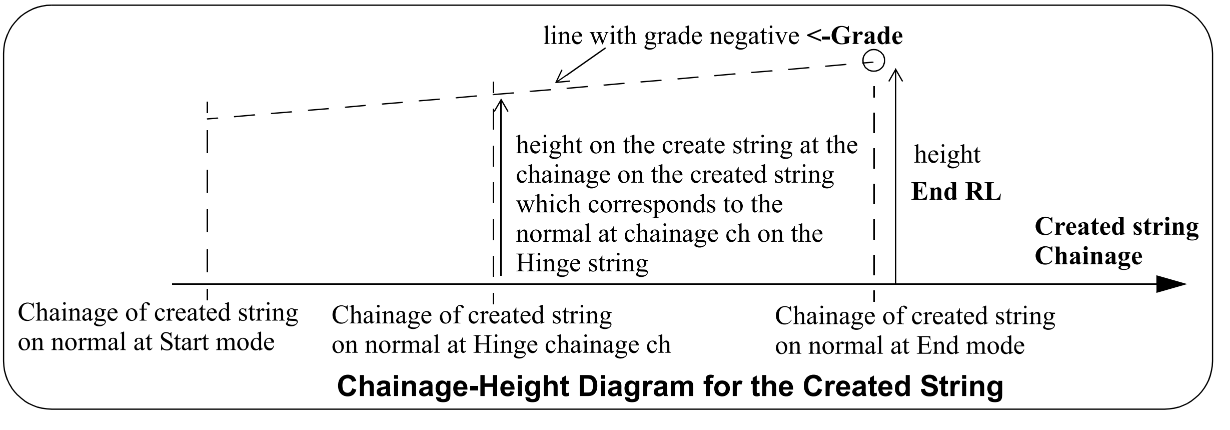
If a **Reference** string is **NOT** selected then the **Start mode** and **End Mode** refer to the **Hinge** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Hinge string**. The vertices of the created string lie on these normals.

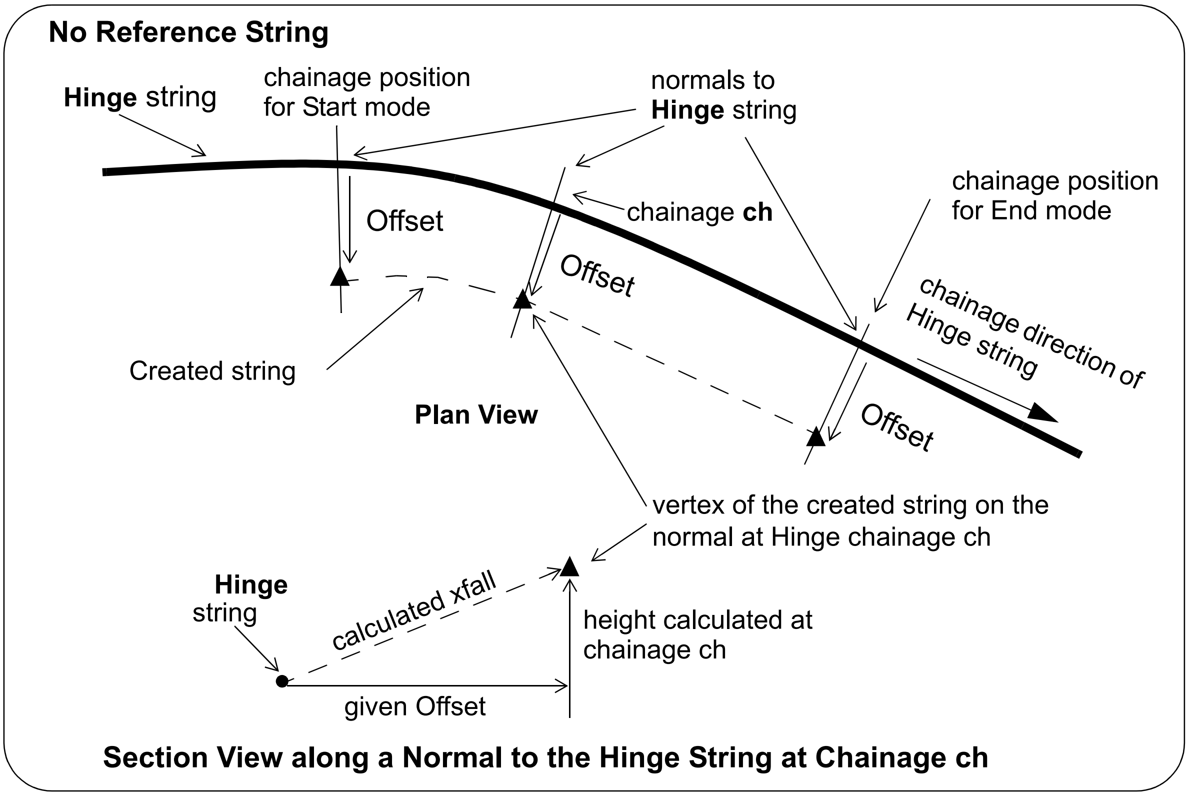
The **plan position** of the created vertices are given by going out from the Hinge string along the normal section for the given **Offset.**

The **last vertex** of the created string is on the normal to the Hinge string at chainage **End mode**, with an offset from the **Hinge** string of **Offset** and with a **height** of **End RL**.



The **heights** of the previous vertices of the created string are defined in (created string chainage, height) space, and lie on the line with grade negative ***<-Grade*** and going through the last vertex.

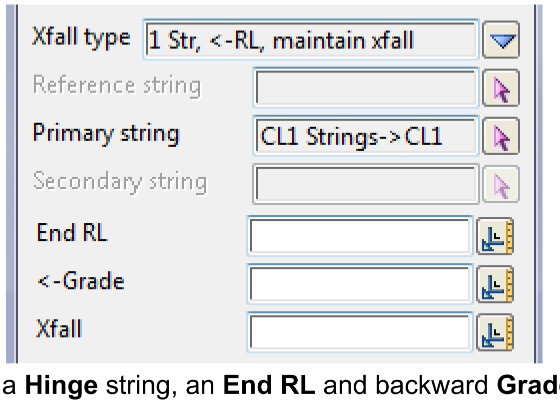




For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.

Continue to the next choice 20.8.11.1.11 1 String, End RL and Grade, Fixed Xfall or return to 20.8.11.1 Xfall Types or 20.8.11 String by Xfall and Grade.

20.8.11.1.11 1 String, End RL and Grade, Fixed Xfall



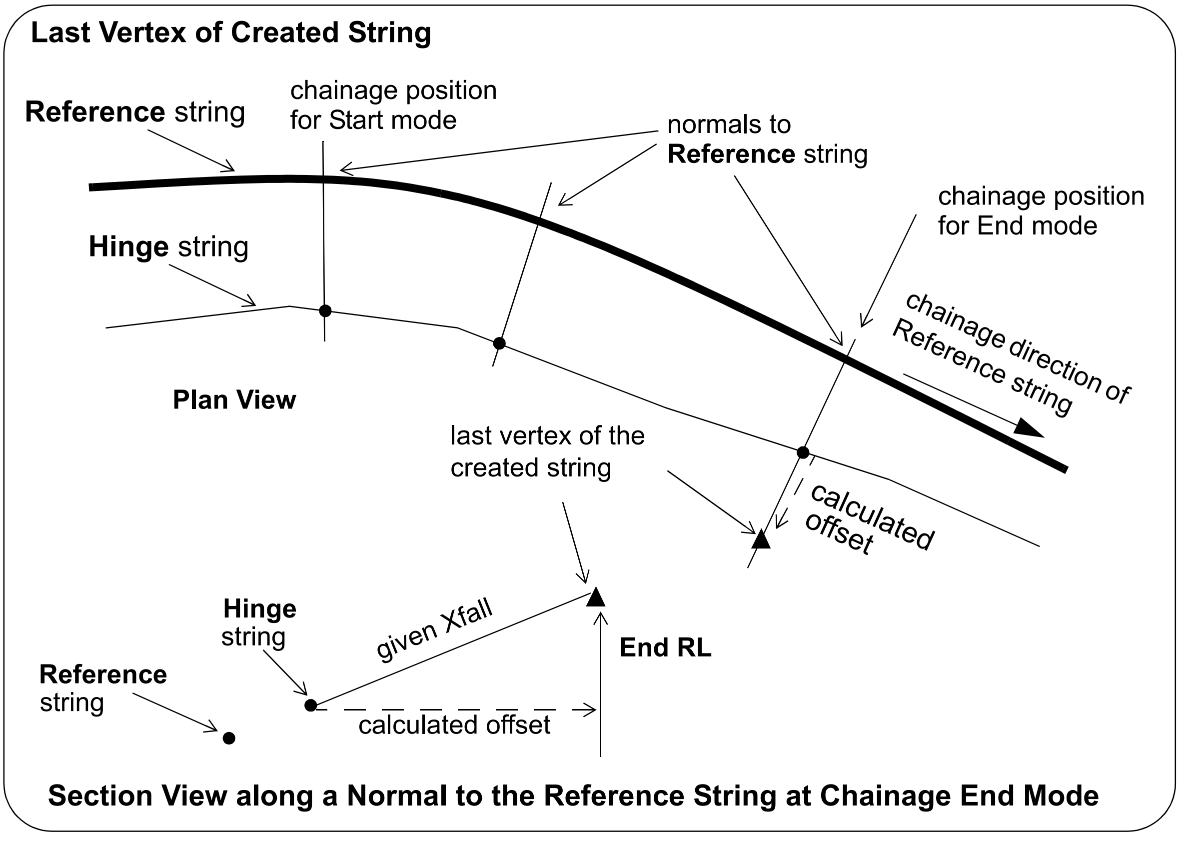
This option requires a **Hinge** string, an **End RL** and backward **Grade**, a **Xfall**, and an **optional Reference** string.

Reference String Selected

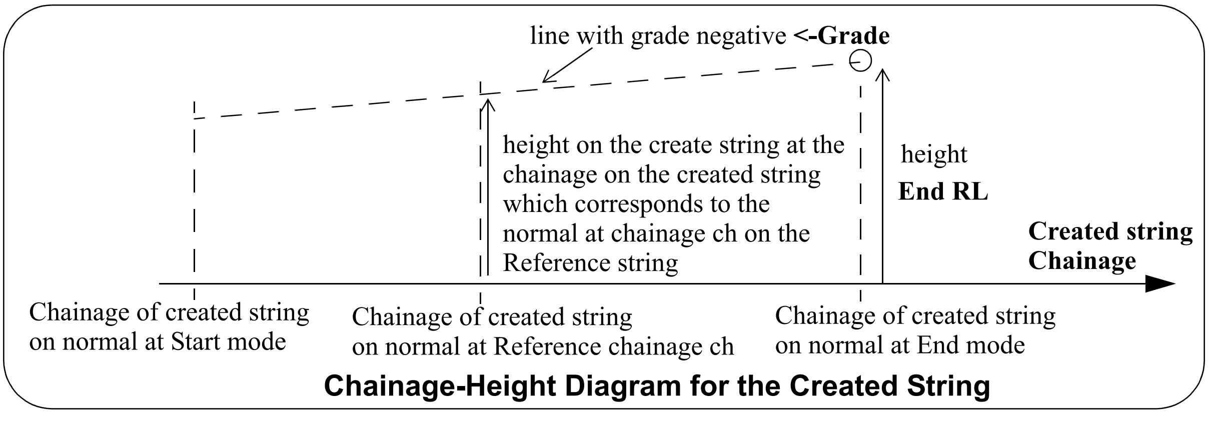
If a **Reference** string is selected then the **Start mode** and **End Mode** refer to the **Reference** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Reference string** and cutting the **Hinge** string. The vertices of the created string lie on these normals.

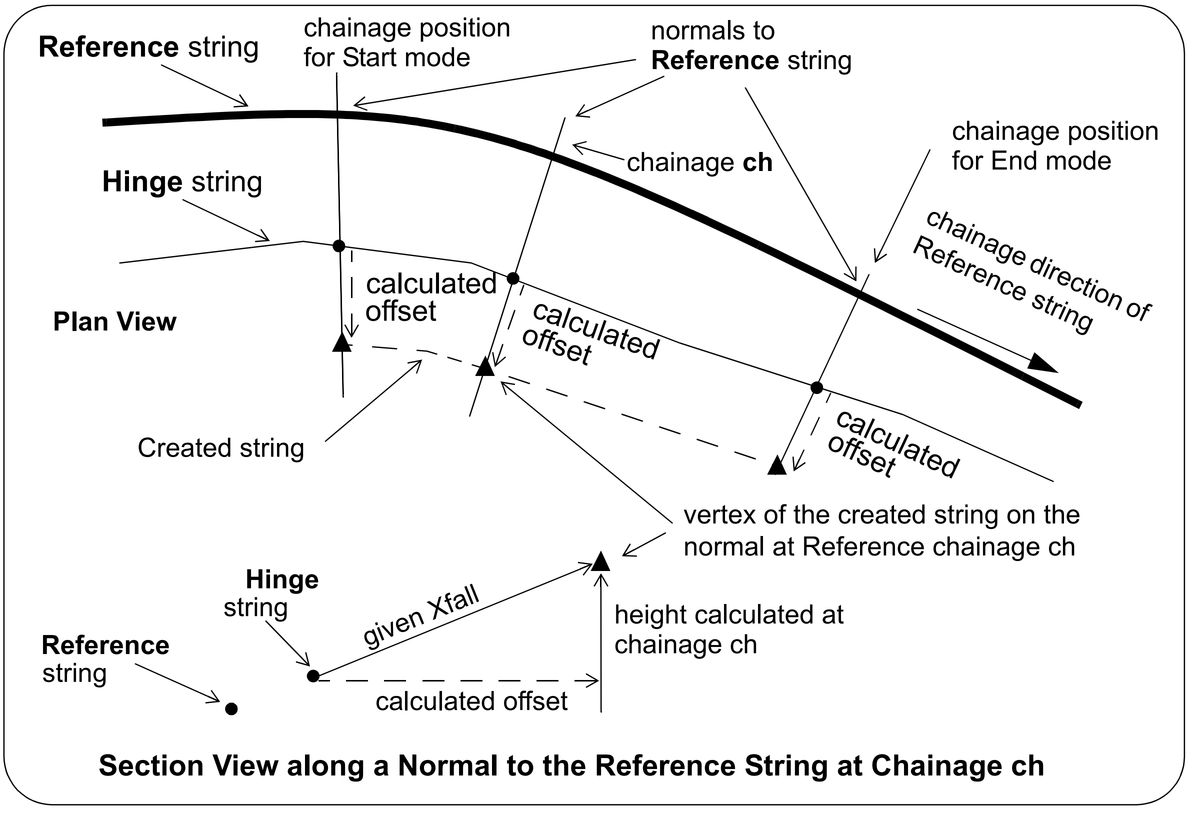
The **last vertex** of the created string is on the normal to the Reference string and has a **height**

End RL. The offset for the last vertex is calculated and is such that the given **Xfall** from Hinge string will give the vertex the height **End RL**.



The **heights** of the previous vertices of the created string are defined in (created string chainage, height) space, and lie on the line with grade negative ***<-Grade*** and going through the last vertex.



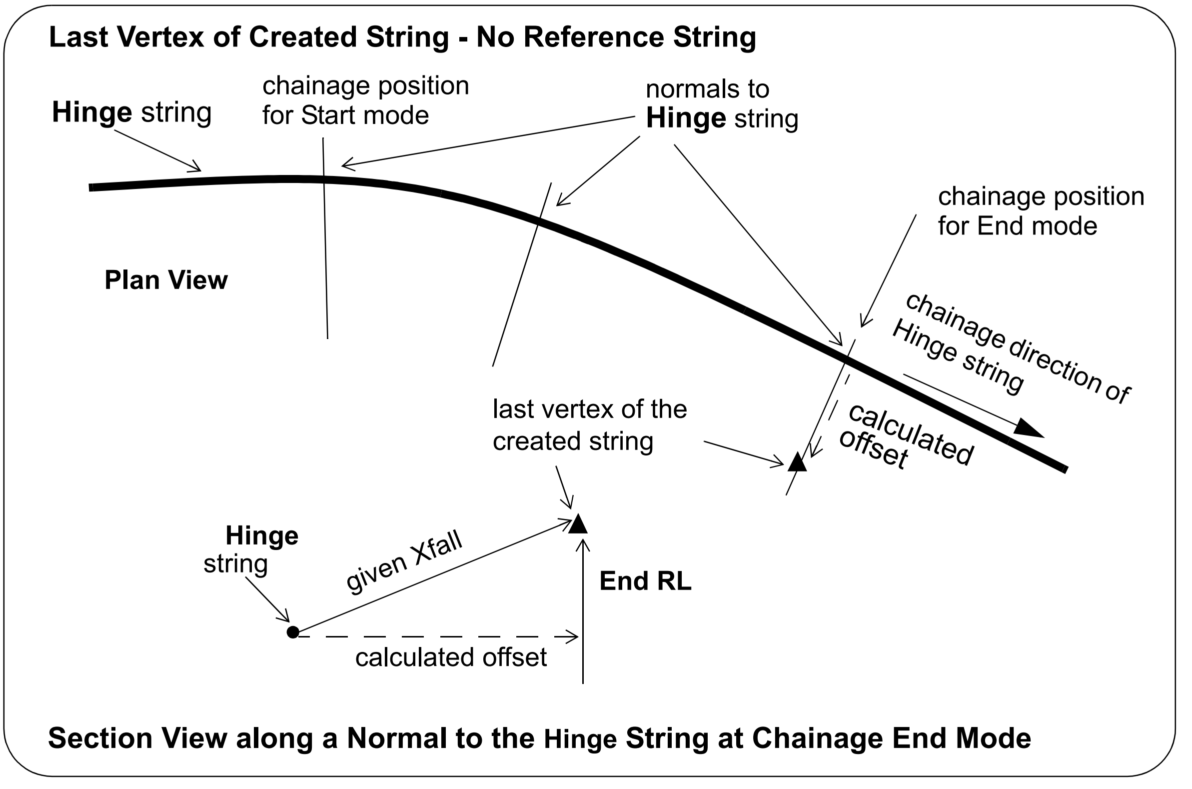


For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.

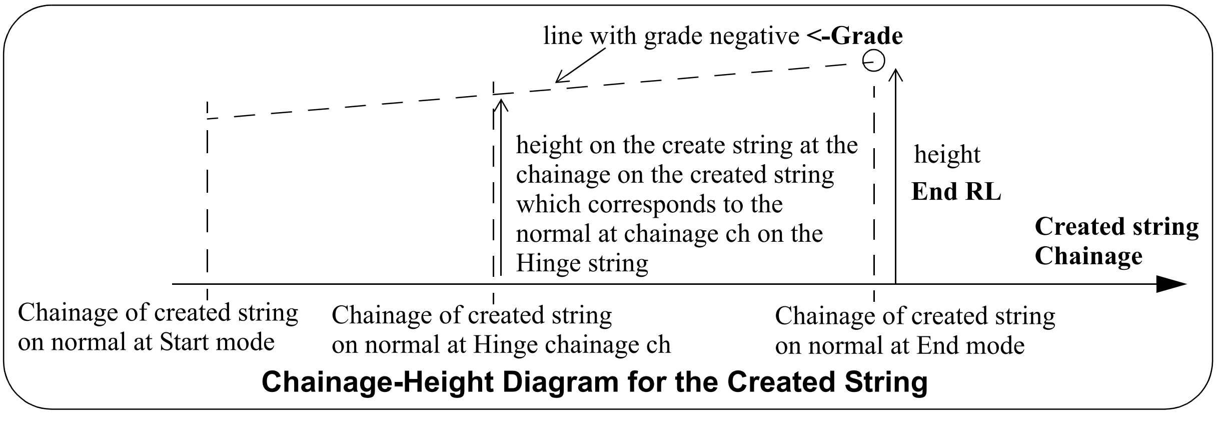
Reference String NOT Selected

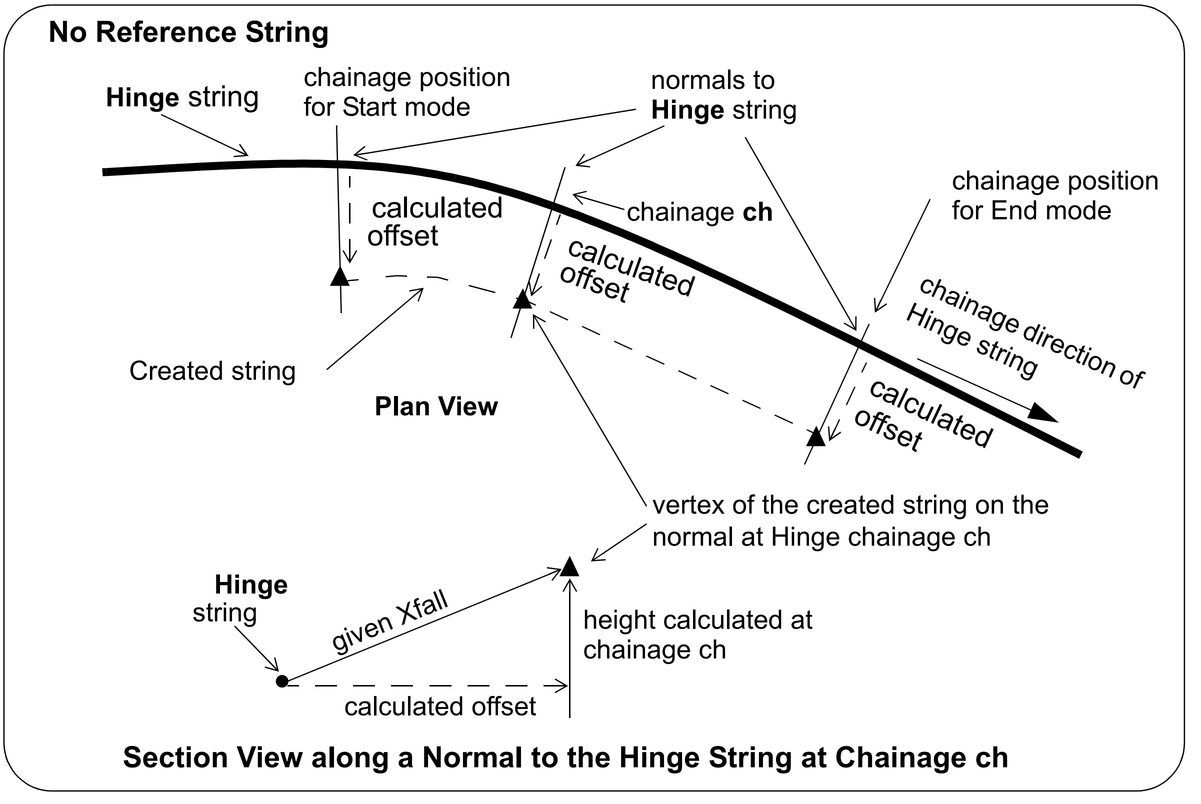
If a **Reference** string is **NOT** selected then the **Start mode** and **End Mode** refer to the **Hinge** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Hinge string**. The vertices of the created string lie on these normals.

The **last vertex** of the created string is on the normal to the Hinge string and has a **height End RL**. The offset for the last vertex is calculated and is such that the given **Xfall** from Hinge string will give the vertex the height **End RL**.



The **heights** of the previous vertices of the created string are defined in (created string chainage, height) space, and lie on the line with grade negative ***<-Grade*** and going through the last vertex.

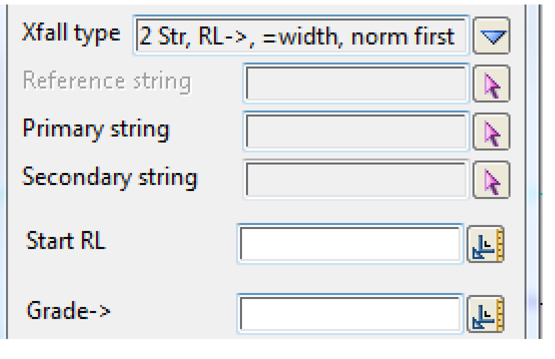




For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.

Continue to the next choice 20.8.11.1.12 2 Strings, Start RL & Grade, Equal Width, Normal to Reference or Hinge String or return to 20.8.11.1 Xfall Types or 20.8.11 String by Xfall and Grade.

20.8.11.1.12 2 Strings, Start RL & Grade, Equal Width, Normal to Reference or Hinge String

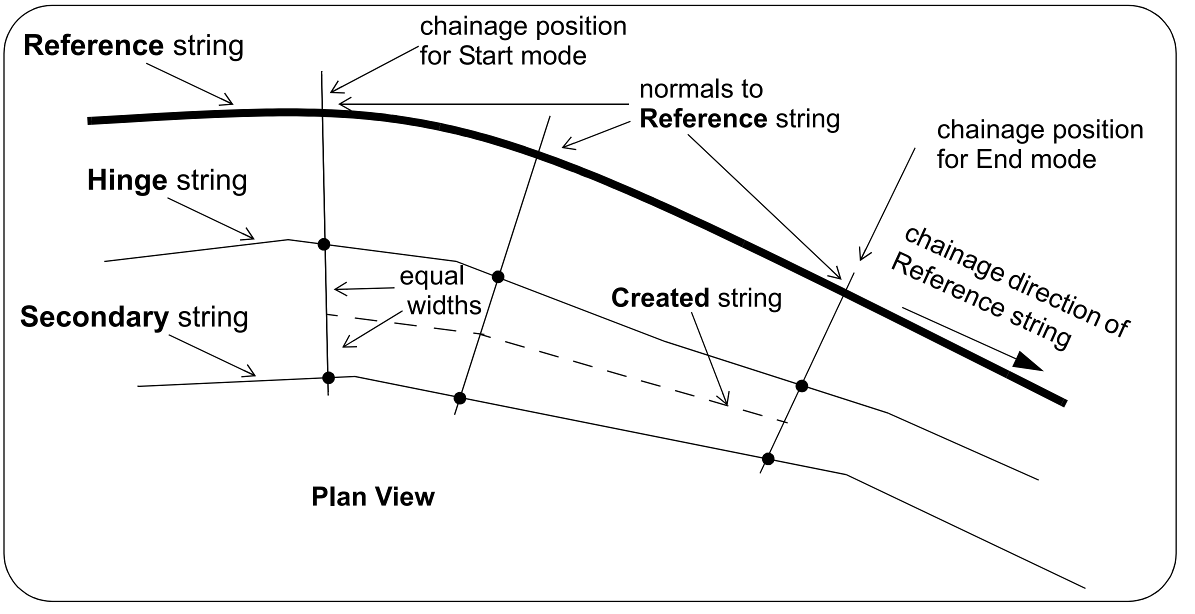


This option requires a **Hinge** and a **Secondary** string, a **Start RL** and forward **Grade**, and an **optional Reference** string.

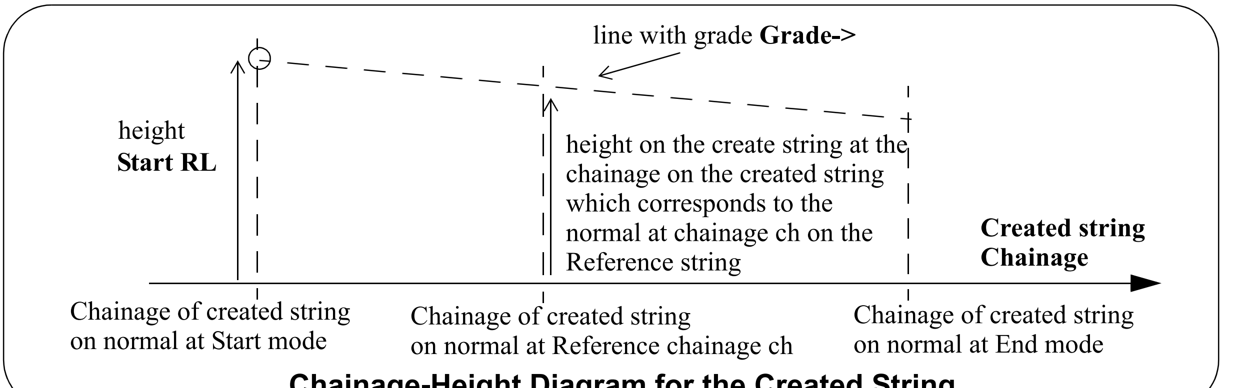
Reference String Selected

If a **Reference** string is selected then the **Start mode** and **End Mode** refer to the **Reference** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Reference string** and cutting the **Hinge** and the **Secondary** strings. The vertices of the created string lie on these normals.

On this section, the plan position of the vertex of the new string is created so that it is halfway between the Hinge string and the Secondary string.



The **height** of the vertex on the section at chainage ch, is defined in (created string chainage, height) space, and is on the line with grade ***Grade->*** and going through the point (normal to Start mode, Start RL).

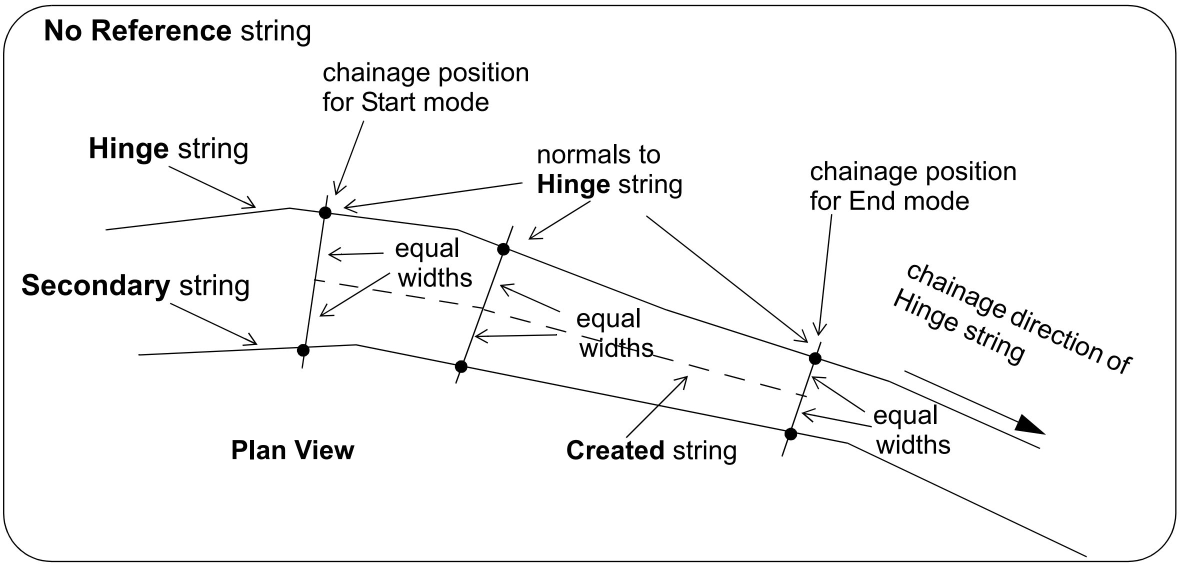


Chainage-Height Diagram for the Created String

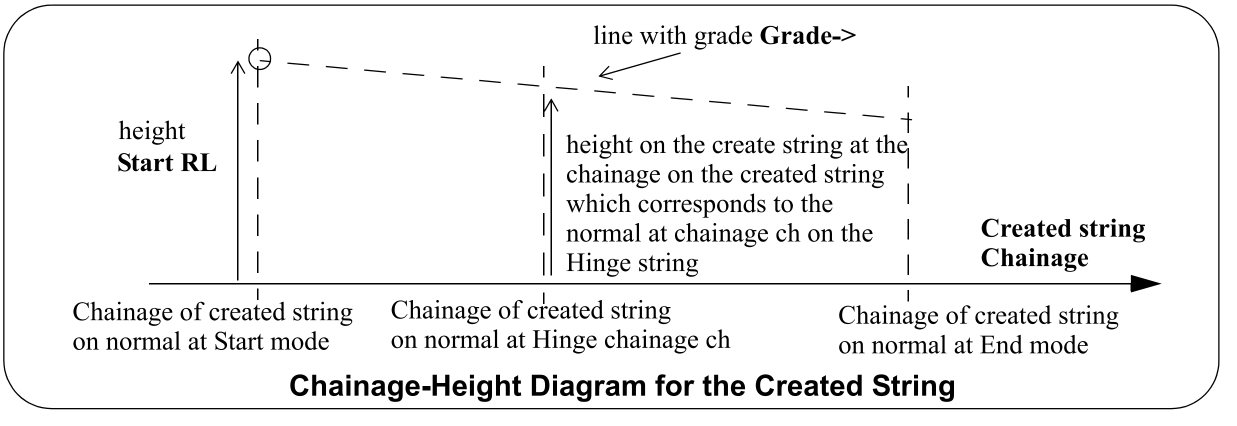
Reference String NOT Selected

If a **Reference** string is **NOT** selected then the **Start mode** and **End Mode** refer to the **Hinge** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Hinge string**. The vertices of the created string lie on these normals.

On this section, the plan position of the vertex of the new string is created so that it is halfway between the Hinge string and the Secondary string.

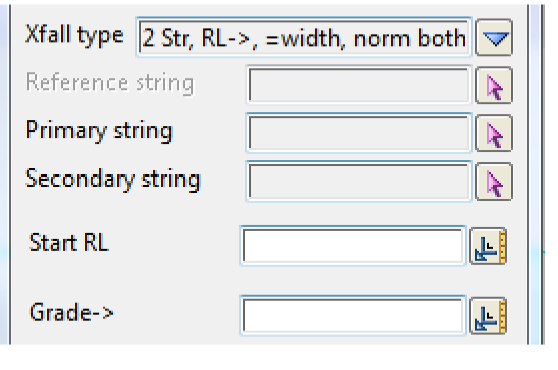


The **height** of the vertex on the section at chainage ch, is defined in (created string chainage, height) space, and is on the line with grade ***Grade->*** and going through the point (normal to Start mode, Start RL).



Continue to the next choice 20.8.11.1.13 2 Strings, Start RL & Grade, Equal Width and Normal to Both or return to 20.8.11.1 Xfall Types or 20.8.11 String by Xfall and Grade.

20.8.11.1.13 2 Strings, Start RL & Grade, Equal Width and Normal to Both



This option requires a **Hinge** and a **Secondary** string, a **Start RL** and forward **Grade**, and an **optional Reference** string.

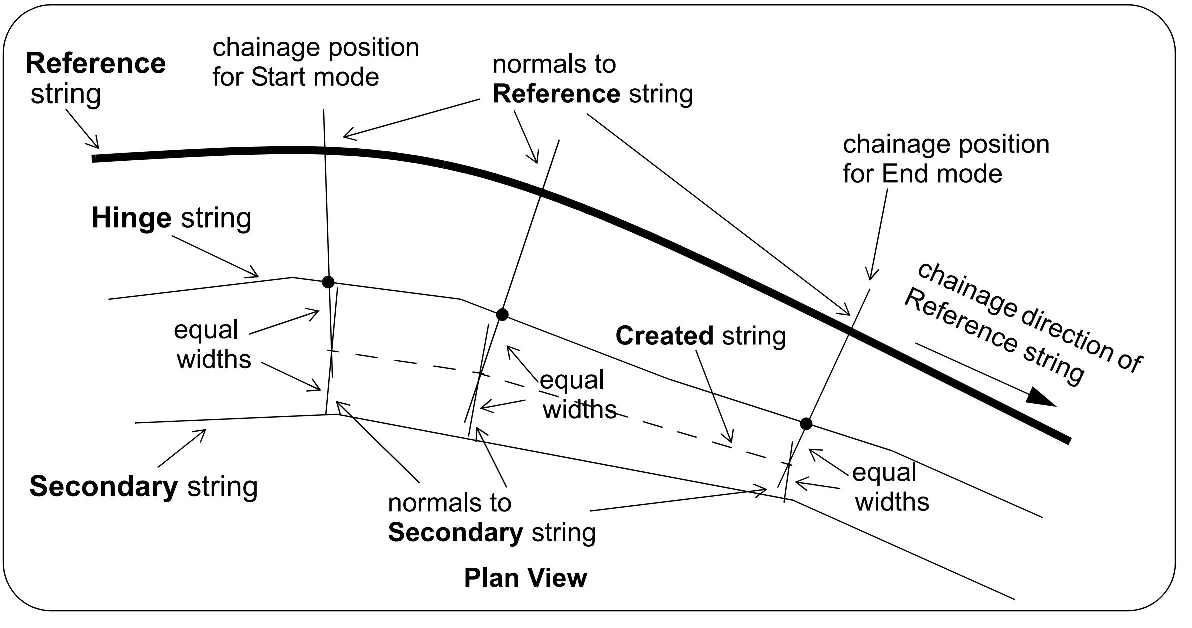
Reference String Selected

If a **Reference** string is selected then the **Start mode** and **End Mode** refer to the **Reference** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Reference string** and cutting the **Hinge** string. The vertices of the created string lie on these normals.

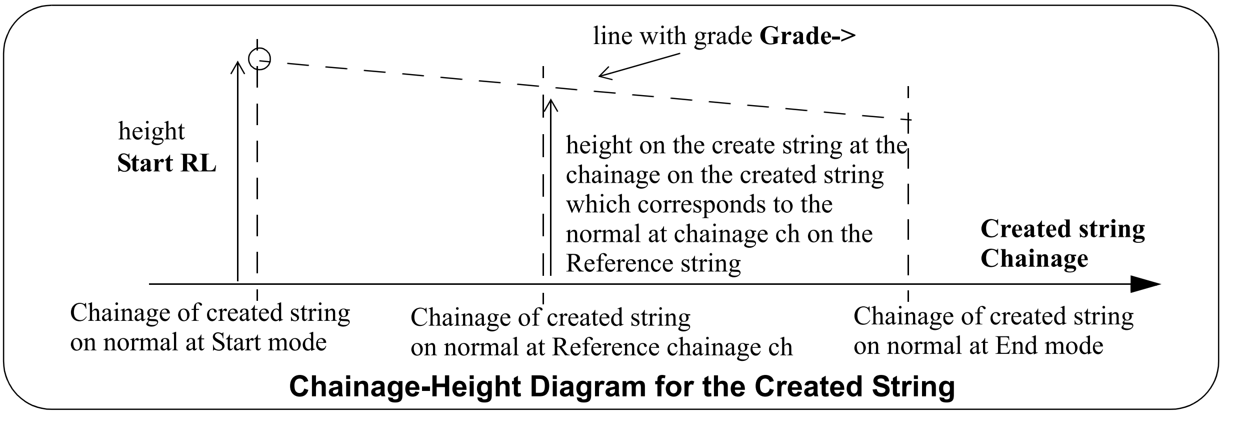
On this section, the plan position of the vertex of the new string is created so that it is halfway between the Hinge string and a normal to the Secondary string.

That is, a normal to the **Secondary** string is created that intersects in plan with the section from the Reference string so that the distance from the Hinge string is the same as the distance from the Secondary string. This gives the plan position of the new vertex of the new string.

Note - it may not be possible to find such an intersection point.



The **height** of the vertex on the section at chainage ch, is defined in (created string chainage, height) space, and is on the line with grade ***Grade->*** and going through the point (normal to Start mode, Start RL).



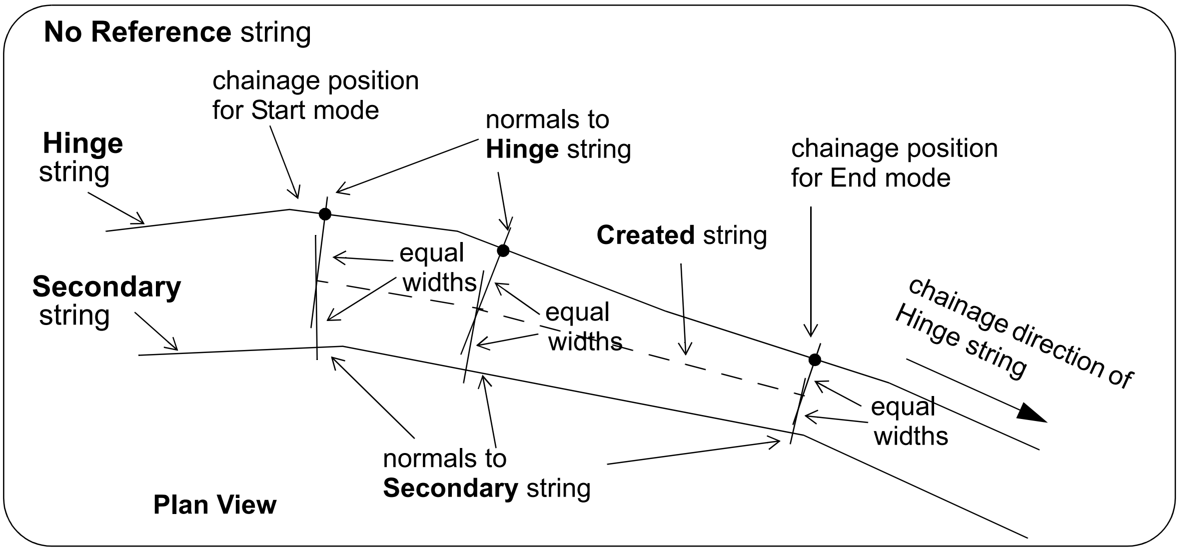
Reference String NOT Selected

If a **Reference** string is **NOT** selected then the **Start mode** and **End Mode** refer to the **Hinge** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Hinge** string. The vertices of the created string lie on these normals.

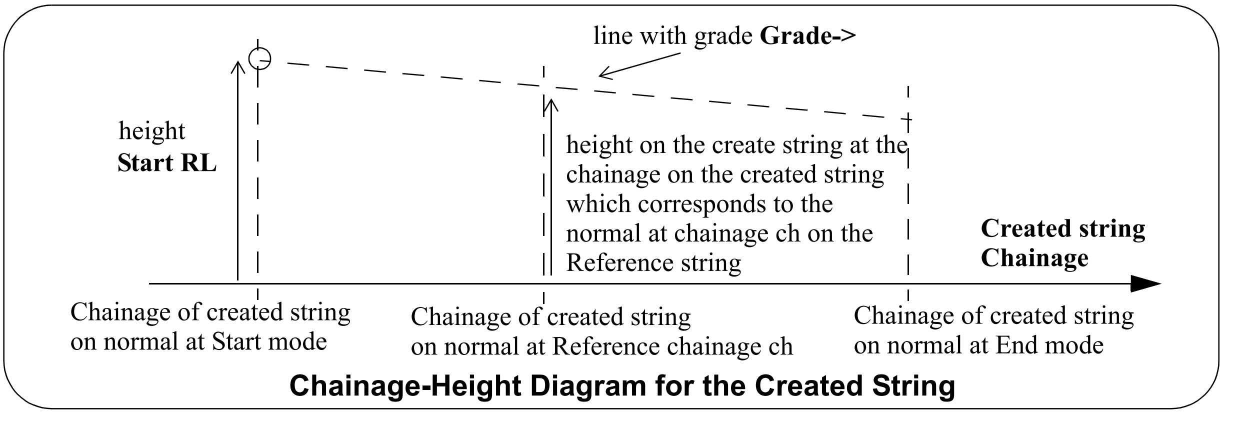
On this section, the plan position of the vertex of the new string is created so that it is halfway between the Hinge string and a **normal** to the Secondary string.

That is, a normal to the **Secondary** string is created that intersects in plan with the section from the Hinge string so that the distance from the Hinge string is the same as the distance from the Secondary string. This gives the plan position of the new vertex of the new string.

Note - it may not be possible to find such an intersection point.

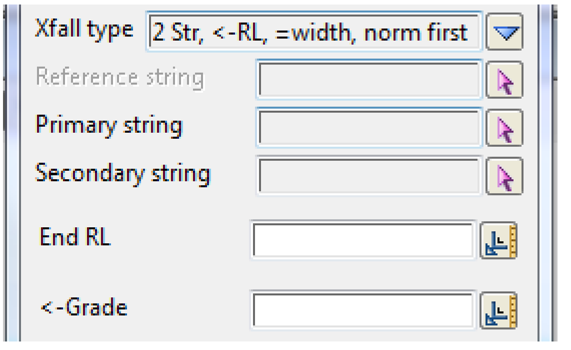


The **height** of the vertex on the section at chainage ch, is defined in (created string chainage, height) space, and is on the line with grade ***Grade->*** and going through the point (normal to Start mode, Start RL).



Continue to the next choice 20.8.11.1.14 2 Strings, End RL & Grade, Equal Width, Normal to Reference or Hinge String or return to 20.8.11.1 Xfall Types or 20.8.11 String by Xfall and Grade.

20.8.11.1.14 2 Strings, End RL & Grade, Equal Width, Normal to Reference or Hinge String

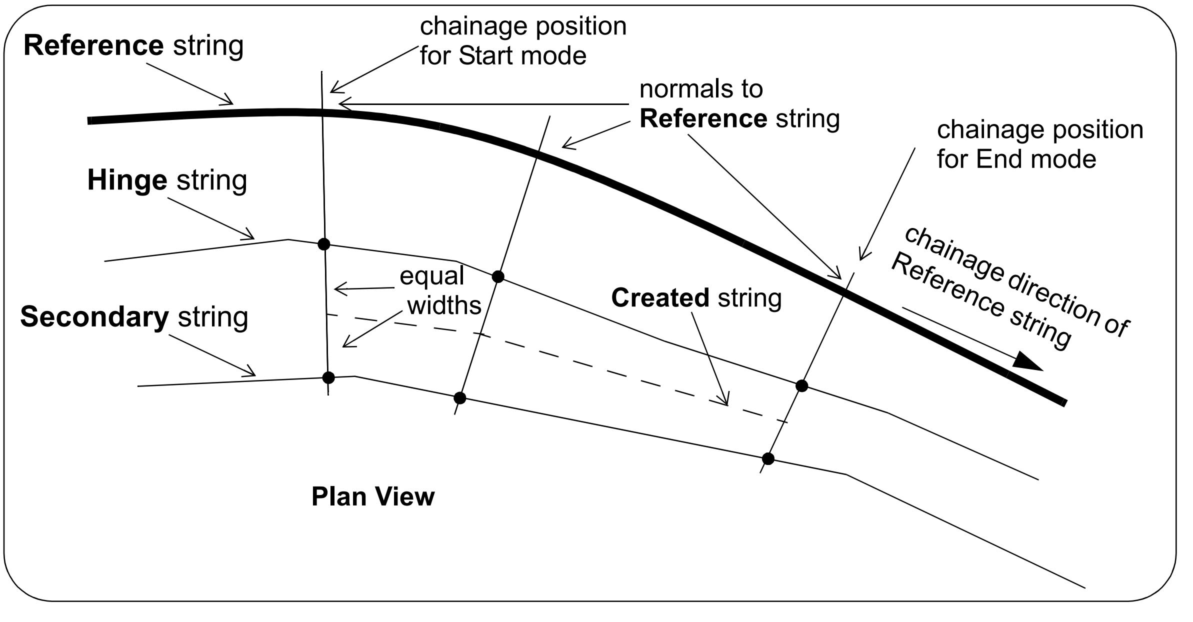


This option requires a **Hinge** and a **Secondary** string, a **End RL** and backward **Grade**, and an **optional Reference** string.

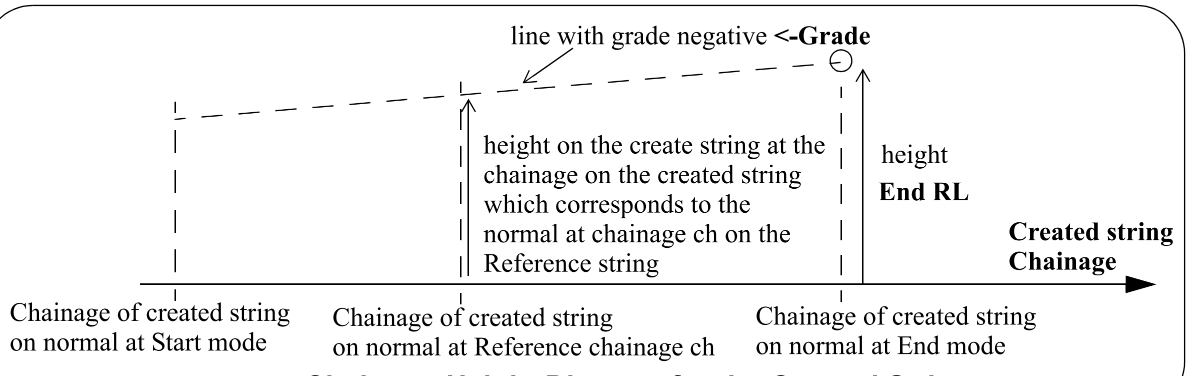
Reference String Selected

If a **Reference** string is selected then the **Start mode** and **End Mode** refer to the **Reference** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Reference string** and cutting the **Hinge** and the **Secondary** strings. The vertices of the created string lie on these normals.

On this section, the plan position of the vertex of the new string is created so that it is halfway between the Hinge string and the Secondary string.



The **height** of the vertex on the section at chainage ch, is defined in (created string chainage, height) space, and is on the line with grade negative **<-Grade** and going through the point (normal to End mode, End RL).



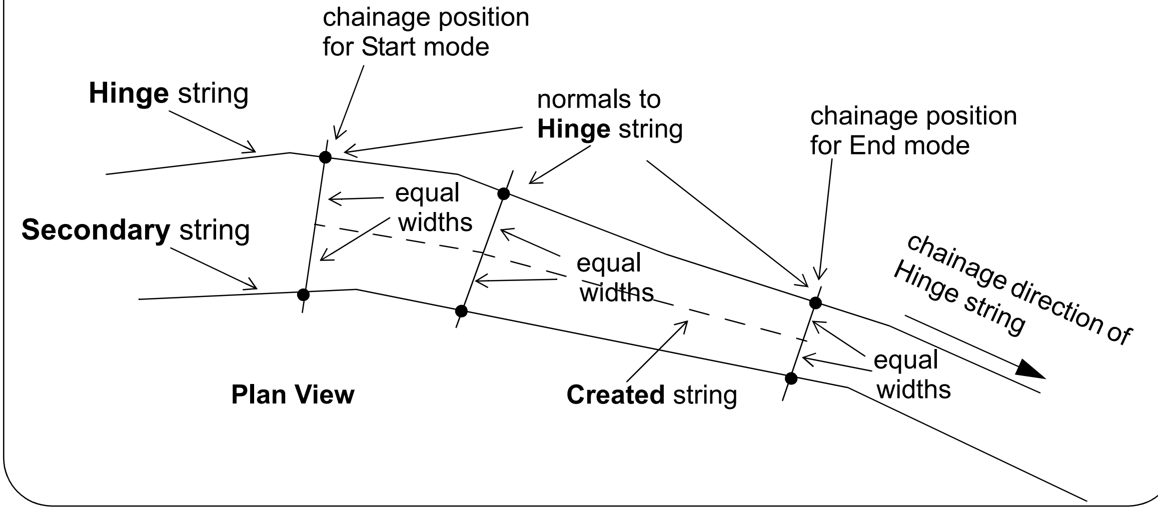
Chainage-Height Diagram for the Created String

Reference String NOT Selected

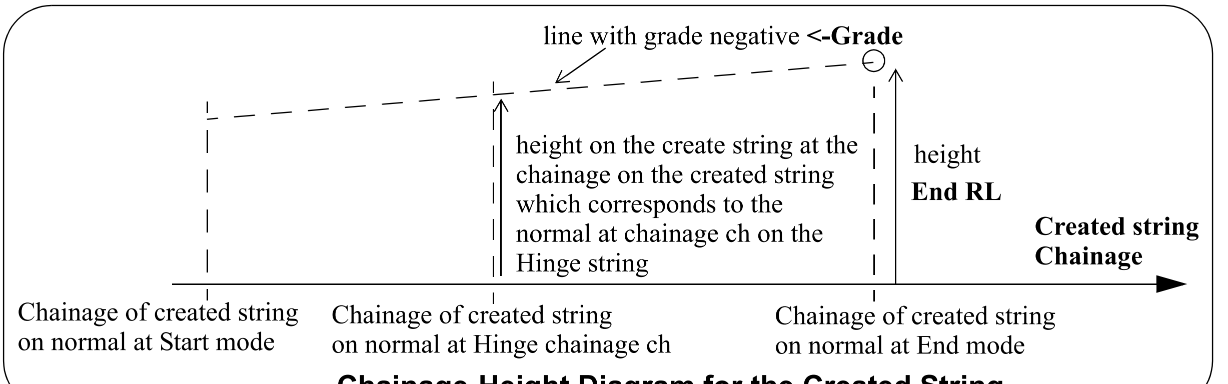
If a **Reference** string is **NOT** selected then the **Start mode** and **End Mode** refer to the **Hinge** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Hinge string** and cutting the **Secondary** string. The vertices of the created string lie on these normals.

On this section, the plan position of the vertex of the new string is created so that it is halfway between the Hinge string and the Secondary string.

No Reference string



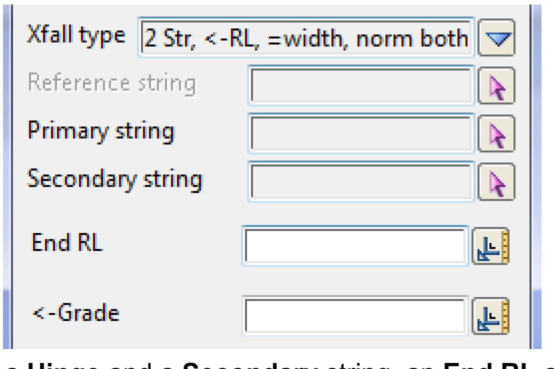
The **height** of the vertex on the section at chainage ch, is defined in (created string chainage, height) space, and is on the line with grade negative **<-Grade** and going through the point (normal to End mode, End RL).



Chainage-Height Diagram for the Created String

Continue to the next choice 20.8.11.1.15 2 Strings, End RL & Grade, Equal Width and Normal to Both or return to 20.8.11.1 Xfall Types or 20.8.11 String by Xfall and Grade.

20.8.11.1.15 2 Strings, End RL & Grade, Equal Width and Normal to Both



This option requires a **Hinge** and a **Secondary** string, an **End RL** and backward **Grade**, and an **optional Reference** string.

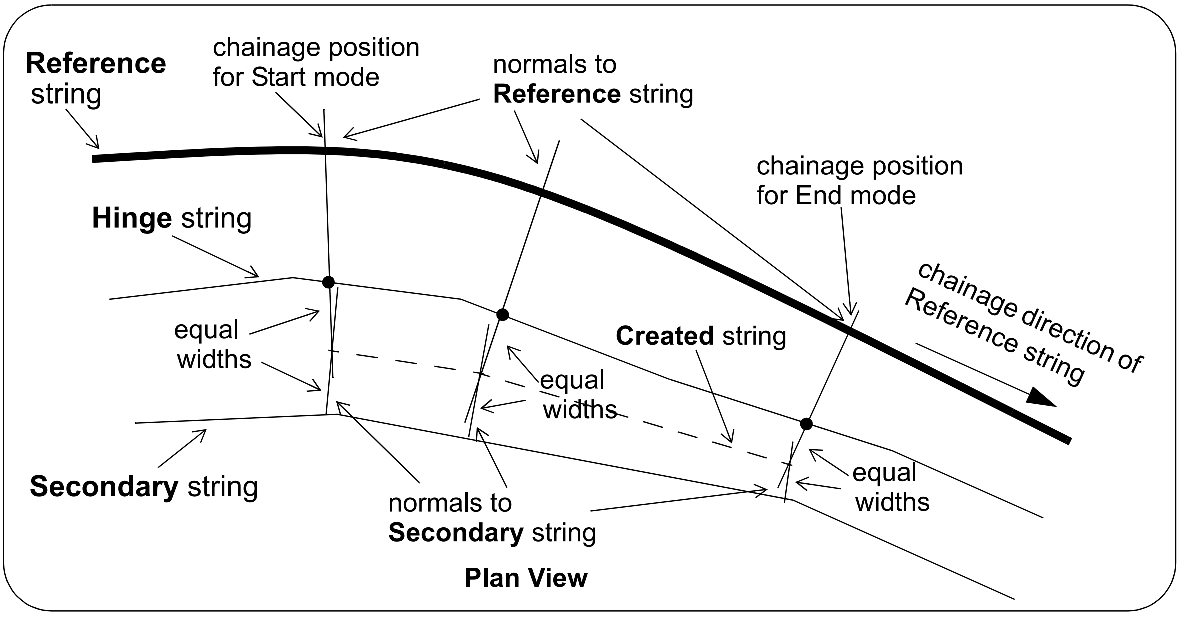
Reference String Selected

If a **Reference** string is selected then the **Start mode** and **End Mode** refer to the **Reference** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Reference string** and cutting the **Hinge** string. The vertices of the created string lie on these normals.

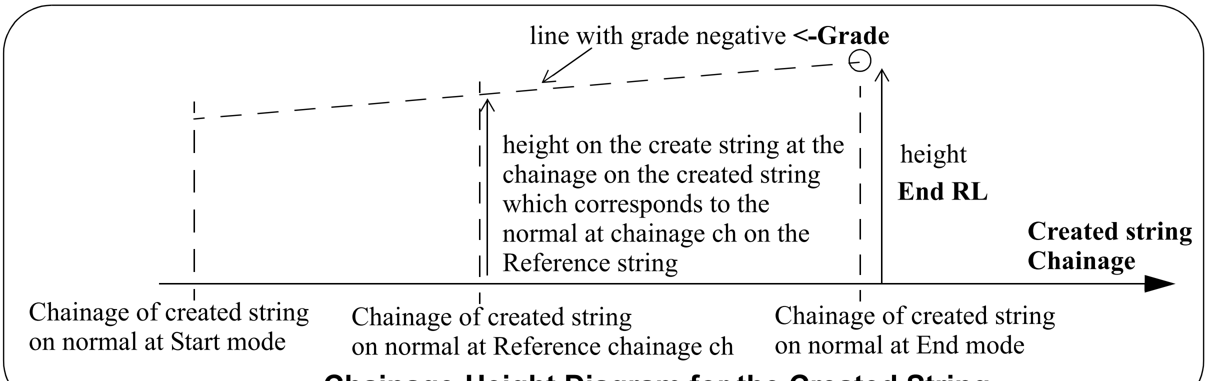
On this section, the plan position of the vertex of the new string is created so that it is halfway between the Hinge string and a normal to the Secondary string.

That is, a normal to the **Secondary** string is created that intersects in plan with the section from the Reference string so that the distance from the Hinge string is the same as the distance from the Secondary string. This gives the plan position of the new vertex of the new string.

Note - it may not be possible to find such an intersection point.



The **height** of the vertex on the section at chainage ch, is defined in (created string chainage, height) space, and is on the line with grade negative **<-Grade** and going through the point (normal to End mode, End RL).



Chainage-Height Diagram for the Created String

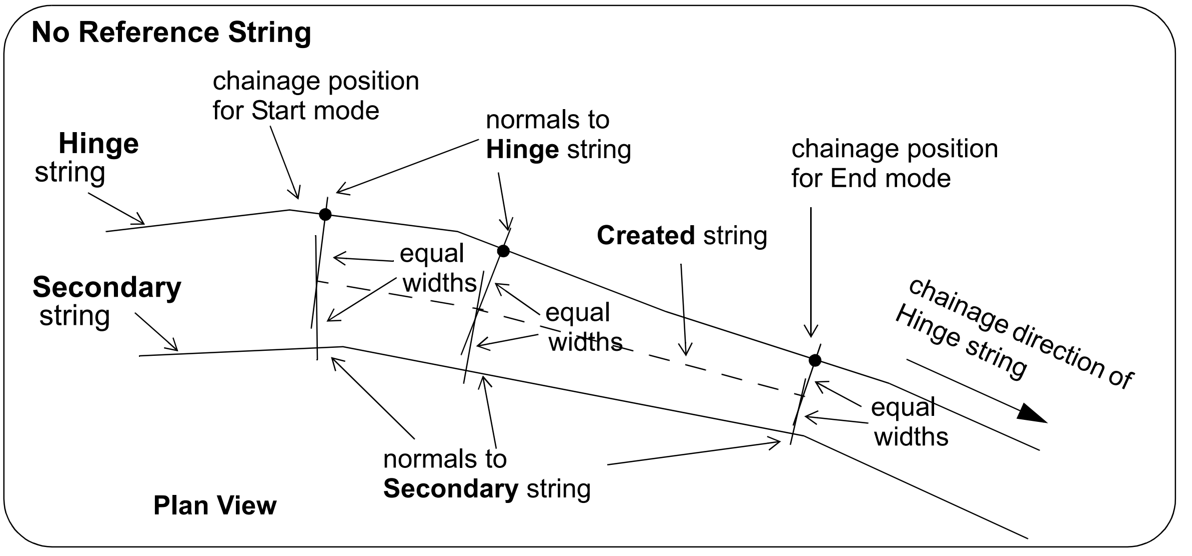
Reference String NOT Selected

If a **Reference** string is **NOT** selected then the **Start mode** and **End Mode** refer to the **Hinge** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Hinge** string. The vertices of the created string lie on these normals.

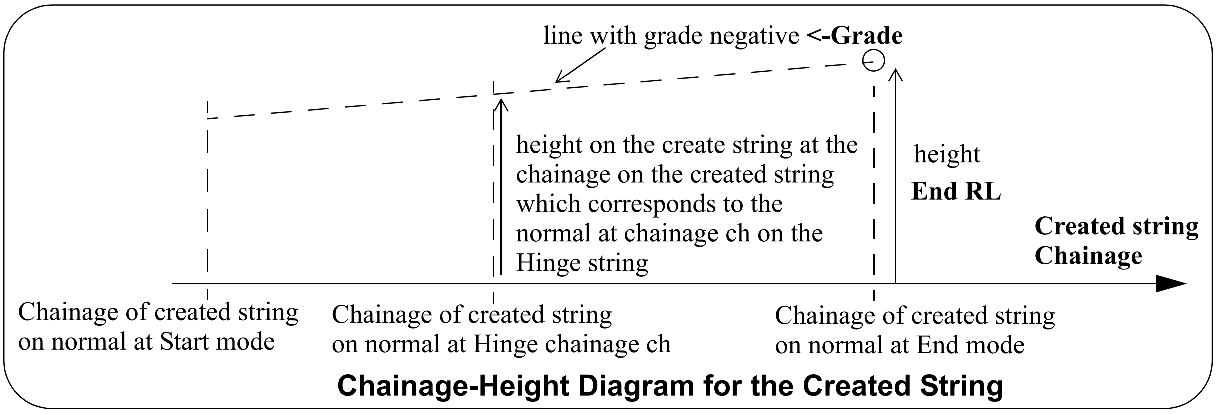
On this section, the plan position of the vertex of the new string is created so that it is halfway between the Hinge string and a **normal** to the Secondary string.

That is, a normal to the **Secondary** string is created that intersects in plan with the section from the Hinge string so that the distance from the Hinge string is the same as the distance from the Secondary string. This gives the plan position of the new vertex of the new string.

Note - it may not be possible to find such an intersection point.



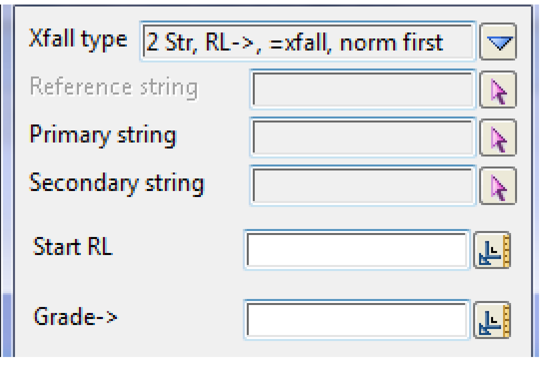
The **height** of the vertex on the section at chainage ch, is defined in (created string chainage, height) space, and is on the line with grade negative **<-Grade** and going through the point (normal to End mode, End RL).



Chainage-Height Diagram for the Created String

Continue to the next choice 20.8.11.1.16 2 Strings, Start RL & Grade, Equal Xfall, Normal to Reference or Hinge String or return to 20.8.11.1 Xfall Types or 20.8.11 String by Xfall and Grade.

20.8.11.1.16 2 Strings, Start RL & Grade, Equal Xfall, Normal to Reference or Hinge String

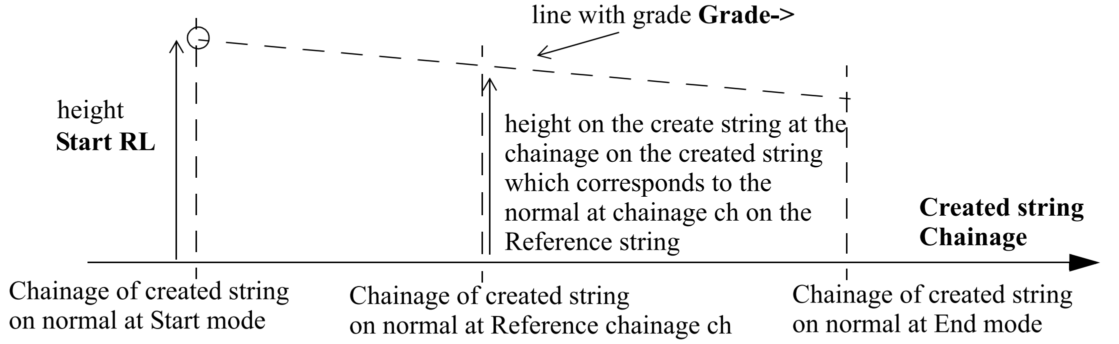


This option requires a **Hinge** and a **Secondary** string, a **Start RL** and forward **Grade**, and an **optional Reference** string.

Reference String Selected

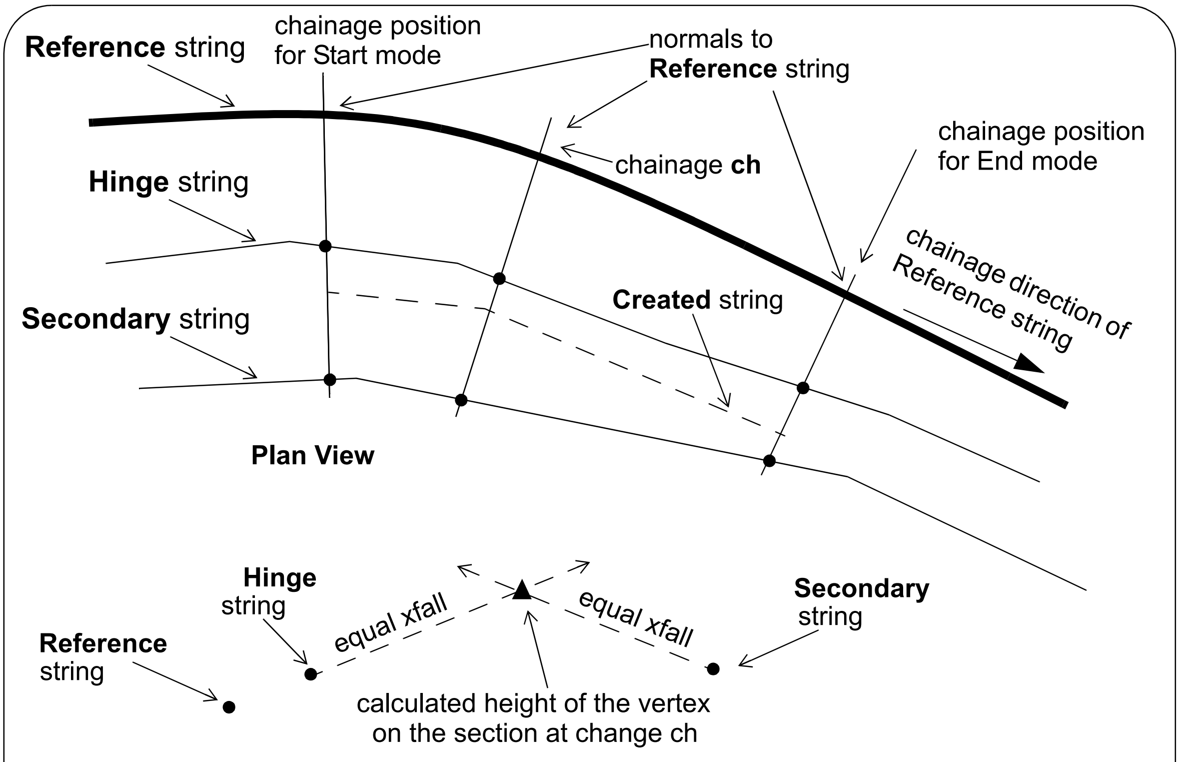
If a **Reference** string is selected then the **Start mode** and **End Mode** refer to the **Reference** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Reference string** and cutting the **Hinge** and the **Secondary** strings. The vertices of the created string lie on these normals.

The height of the vertex of the created string on the section at chainage ch, is defined in (created string chainage, height) space, and is on the line with grade ***Grade->*** and going through the point (normal at Start mode, Start RL).



Chainage-Height Diagram for the Created String

On the section normal to the Reference string at chainage ch, the plan position of the vertex (with the calculated height) is such that the absolute value of the xfall from the Hinge string to the vertex, and the vertex to the Secondary string, is the same.This gives the plan position of the new vertex of the new string.

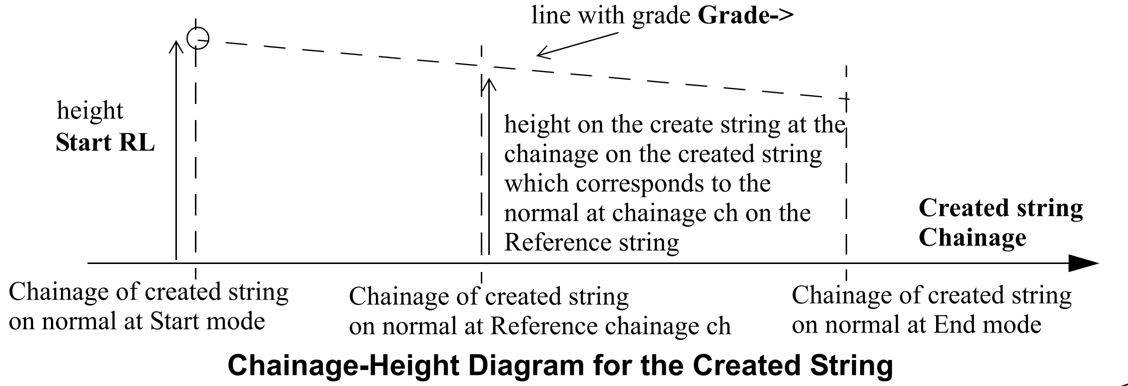


Section View along a Normal to the Reference String at Chainage ch

Reference String NOT Selected

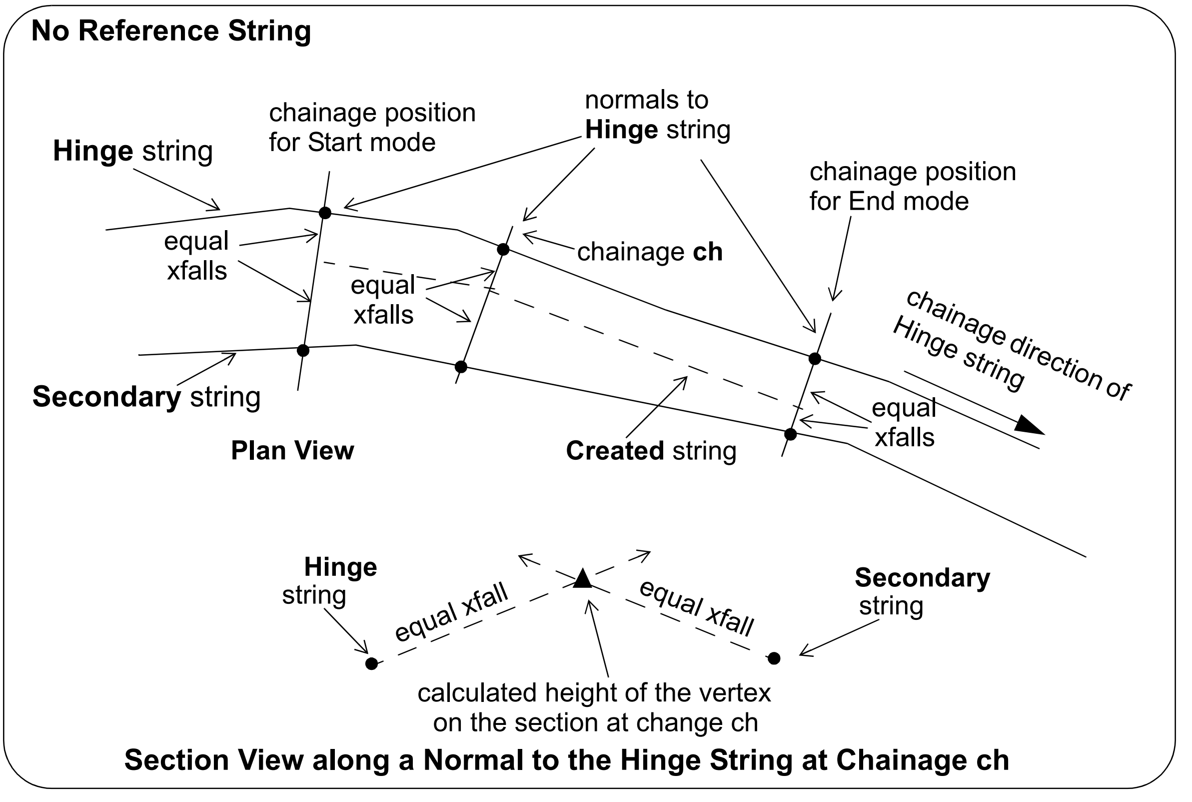
If a **Reference** string is **NOT** selected then the **Start mode** and **End Mode** refer to the **Hinge** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Hinge string** and cutting the **Secondary** string. The vertices of the created string lie on these normals.

The height of the vertex of the created string on the section at chainage ch, is defined in (created string chainage, height) space, and is on the line with grade ***Grade->*** and going through the point (normal at Start mode, Start RL).



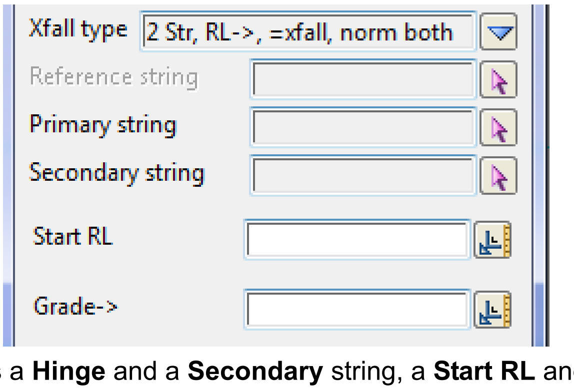
Chainage-Height Diagram for the Created String

On the section normal to the Hinge string at chainage ch, the plan position of the vertex (with the calculated height) is such that the absolute value of the xfall from the Hinge string to the vertex, and the vertex to the Secondary string, is the same.This gives the plan position of the new vertex of the new string.



Continue to the next choice 20.8.11.1.17 2 Strings, Start RL & Grade, Equal Xfall and Normal to Both or return to 20.8.11.1 Xfall Types or 20.8.11 String by Xfall and Grade.

20.8.11.1.17 2 Strings, Start RL & Grade, Equal Xfall and Normal to Both

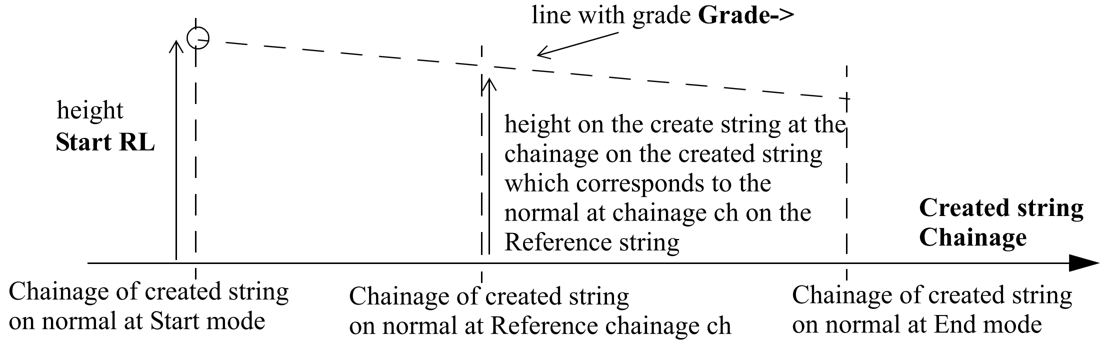


This option requires a **Hinge** and a **Secondary** string, a **Start RL** and forward **Grade**, and an **optional Reference** string.

Reference String Selected

If a **Reference** string is selected then the **Start mode** and **End Mode** refer to the **Reference** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Reference string** and cutting the **Hinge** string. The vertices of the created string lie on these normals.

The height of the vertex of the created string on the section at chainage ch, is defined in (created string chainage, height) space, and is on the line with grade ***Grade->*** and going through the point (normal at Start mode, Start RL).

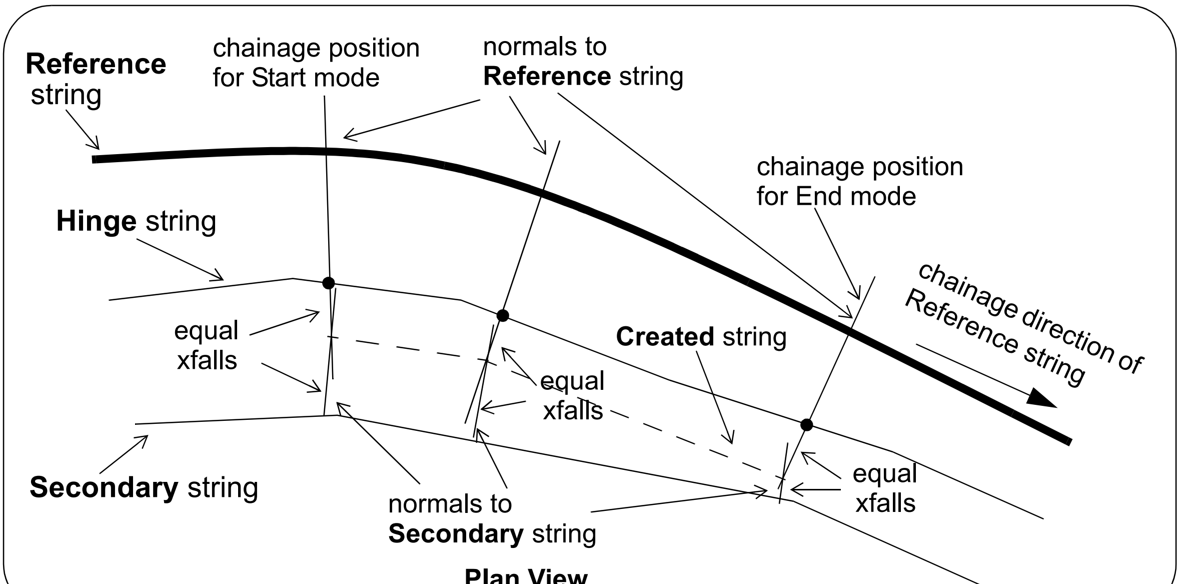


Chainage-Height Diagram for the Created String

On the section at Reference chainage ch, the position of the vertex (with the calculated height) is such that the absolute value of the xfall from the vertex to the Hinge string, and the absolute xfall from the vertex to the **normal to the Secondary** string, is the same.

That is, a normal to the **Secondary** string is created that intersects in plan with the section from the Reference string so that the absolute value of the xfall from the Hinge string to the vertex (with the calculated height), is the same as the absolute grade from the vertex to the Secondary string. This gives the plan position of the vertex of the created string.

Note - it may not be possible to find such an intersection point.

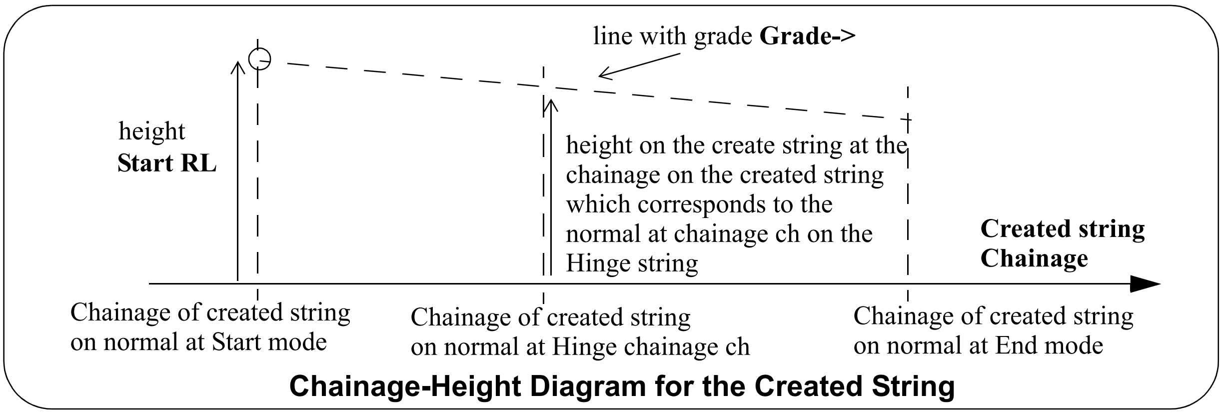


Plan View

Reference String NOT Selected

If a **Reference** string is **NOT** selected then the **Start mode** and **End Mode** refer to the **Hinge** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Hinge string.** The vertices of the created string lie on these normals.

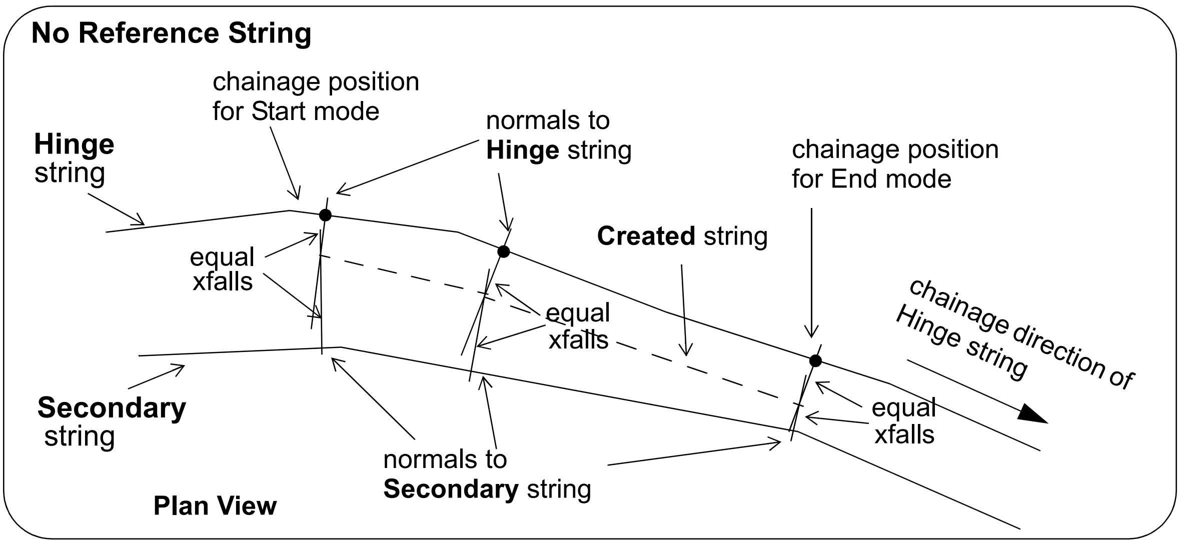
The height of the vertex of the created string on the section at chainage ch, is defined in (created string chainage, height) space, and is on the line with grade ***Grade->*** and going through the point (normal at Start mode, Start RL).



On the section at Hinge chainage ch, the position of the vertex (with the calculated height) is such that the absolute value of the xfall from the vertex to the Hinge string, and the absolute xfall from the vertex to the **normal to the Secondary** string, is the same.

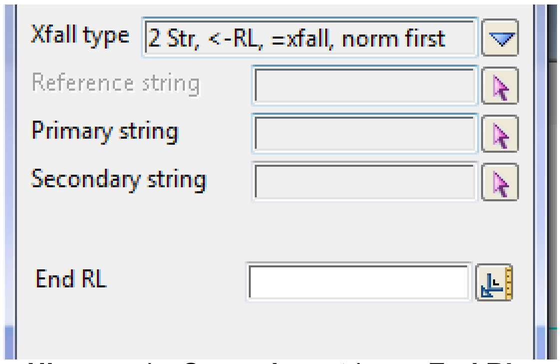
That is, a normal to the **Secondary** string is created that intersects in plan with the section from the Hinge string so that the absolute value of the xfall from the Hinge string to the vertex (with the calculated height), is the same as the absolute grade from the vertex to the Secondary string. This gives the plan position of the vertex of the created string.

Note - it may not be possible to find such an intersection point.



Continue to the next choice 20.8.11.1.18 2 Strings, End RL & Grade, Equal Xfall, Normal to Reference or Hinge String or return to 20.8.11.1 Xfall Types or 20.8.11 String by Xfall and Grade.

20.8.11.1.18 2 Strings, End RL & Grade, Equal Xfall, Normal to Reference or Hinge String

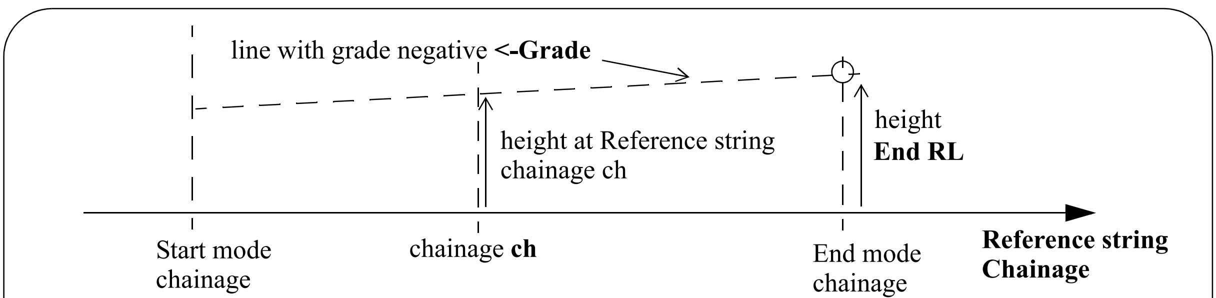


This option requires a **Hinge** and a **Secondary** string, a **End RL** and backward **Grade**, and an **optional Reference** string.

The height of the vertex (of the new string) on the section at chainage ch, is in (Reference string chainage, height) space, on the line with grade of negative **<-Grade** and going through the point (End mode, End RL).

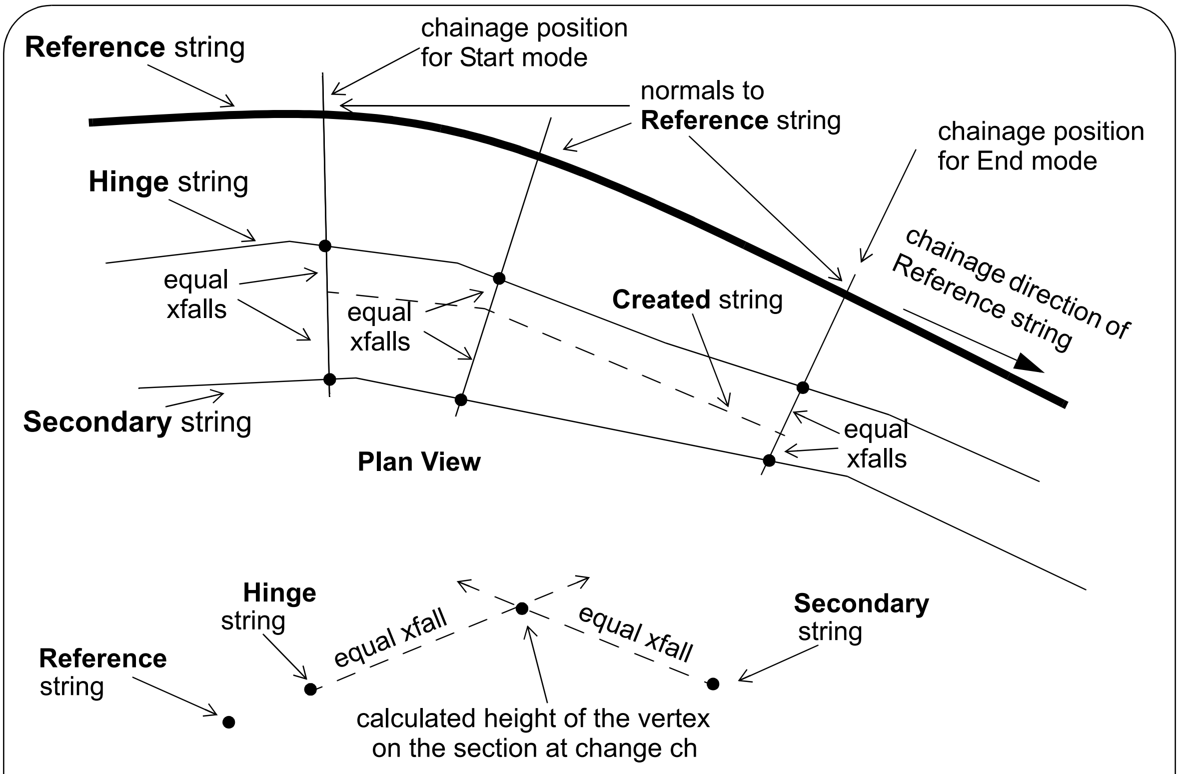
That is,

the "height at ch" = End RL - <-Grade *(End mode - ch)/(End mode - Start mode)/100



Chainage-Height Diagram for End RL and <-Grade

On the section normal to the Reference string at chainage ch, the position of the vertex with the calculated height is such that the absolute value of the xfall from the Hinge string to the vertex, and the absolute value of the xfall from the vertex to the Secondary string, are the same.



Section View along a Normal to the Reference String

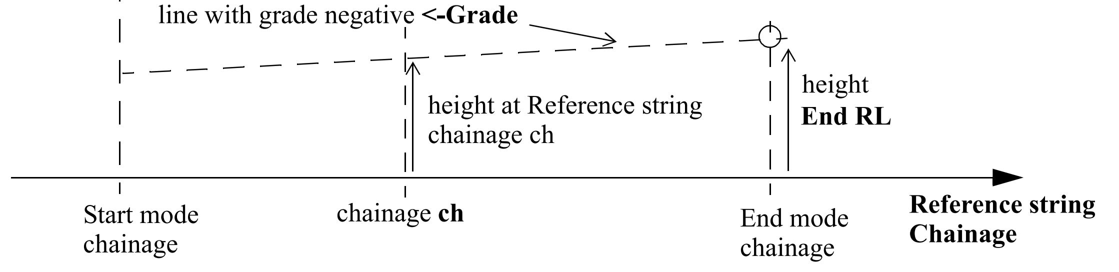
Reference String NOT Selected

If a **Reference** string is **NOT** selected then the **Start mode** and **End Mode** refer to **Hinge** string and at each chainage (ch), a section is taken **normal** (perpendicular) to the **Hinge string** and cutting the **Secondary** strings.

The height of the vertex (of the new string) on the section at chainage ch, is in (Hinge string chainage, height) space, on the line with grade of negative **<-Grade** and going through the point (End mode, End RL).

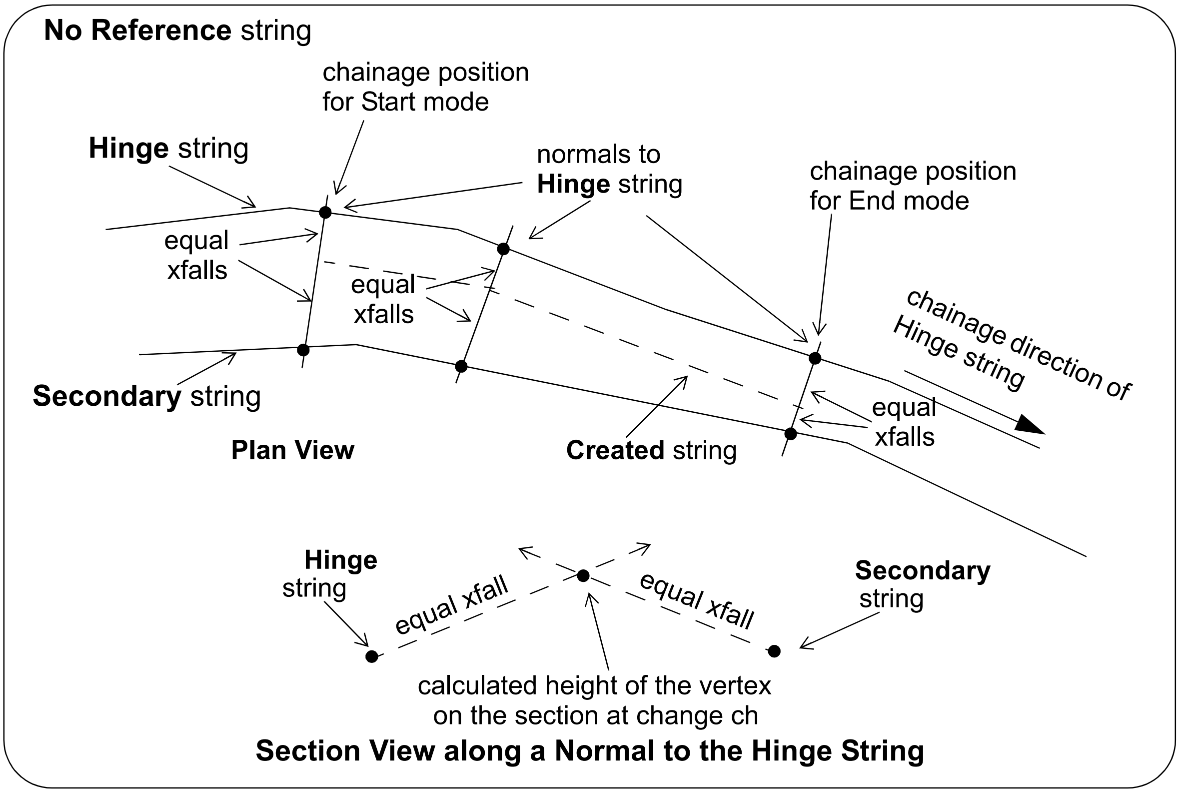
That is,

the "height at ch" = End RL - <-Grade *(End mode - ch)/(End mode - Start mode)/100



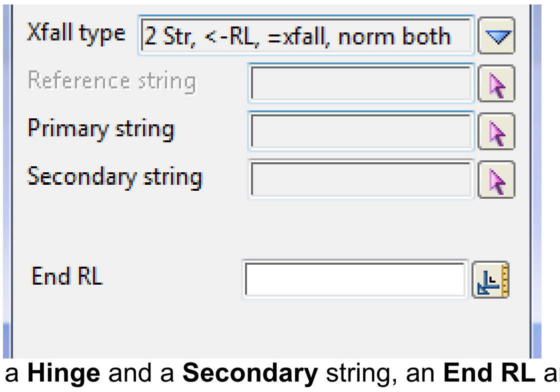
Chainage-Height Diagram for End RL and <-Grade

On the section normal to the Hinge string at chainage ch, the position of the vertex with the calculated height is such that the absolute value of the xfall from the Hinge string to the vertex, and the absolute value of the xfall from the vertex to the Secondary string, are the same.



Continue to the next choice 20.8.11.1.19 2 Strings, End RL & Grade, Equal Xfall and Normal to Both or return to 20.8.11.1 Xfall Types or 20.8.11 String by Xfall and Grade.

20.8.11.1.19 2 Strings, End RL & Grade, Equal Xfall and Normal to Both

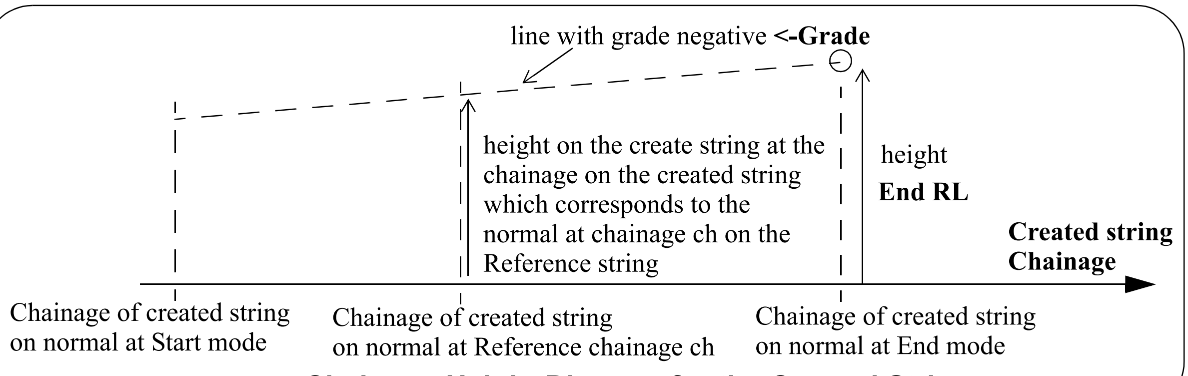


This option requires a **Hinge** and a **Secondary** string, an **End RL** and backward **Grade**, and an **optional Reference** string.

Reference String Selected

If a **Reference** string is selected then the **Start mode** and **End Mode** refer to the **Reference** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Reference string** and cutting the **Hinge** string. The vertices of the created string lie on these normals.

The **height** of the vertex on the section at chainage ch, is defined in (created string chainage, height) space, and is on the line with grade negative **<-Grade** and going through the point (normal to End mode, End RL).

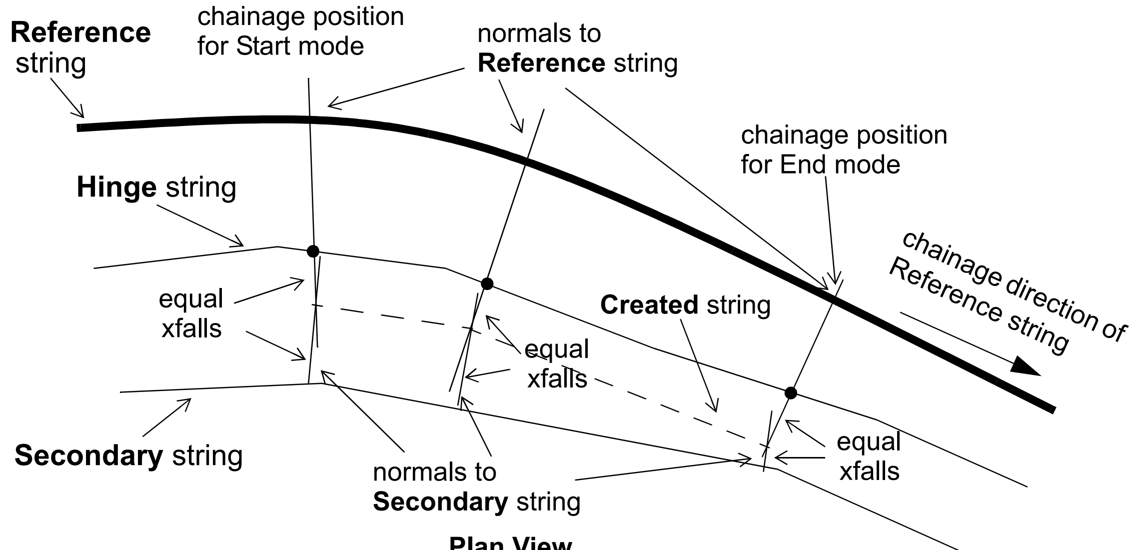


Chainage-Height Diagram for the Created String

On the section at Reference chainage ch, the position of the vertex (with the calculated height) is such that the absolute value of the xfall from the vertex to the Hinge string, and the absolute xfall from the vertex to the **normal to the Secondary** string, is the same.

That is, a normal to the **Secondary** string is created that intersects in plan with the section from the Reference string so that the absolute value of the xfall from the Hinge string to the vertex (with the calculated height), is the same as the absolute grade from the vertex to the Secondary string. This gives the plan position of the vertex of the created string.

Note - it may not be possible to find such an intersection point.

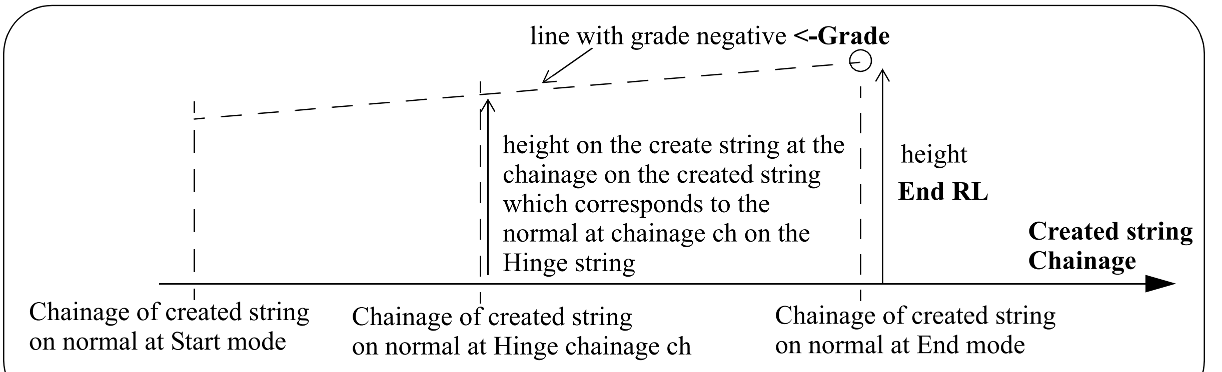


Plan View

Reference String NOT Selected

If a **Reference** string is **NOT** selected then the **Start mode** and **End Mode** refer to the **Hinge** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Hinge string**. The vertices of the created string lie on these normals.

The **height** of the vertex on the section at chainage ch, is defined in (created string chainage, height) space, and is on the line with grade negative **<-Grade** and going through the point (normal to End mode, End RL).

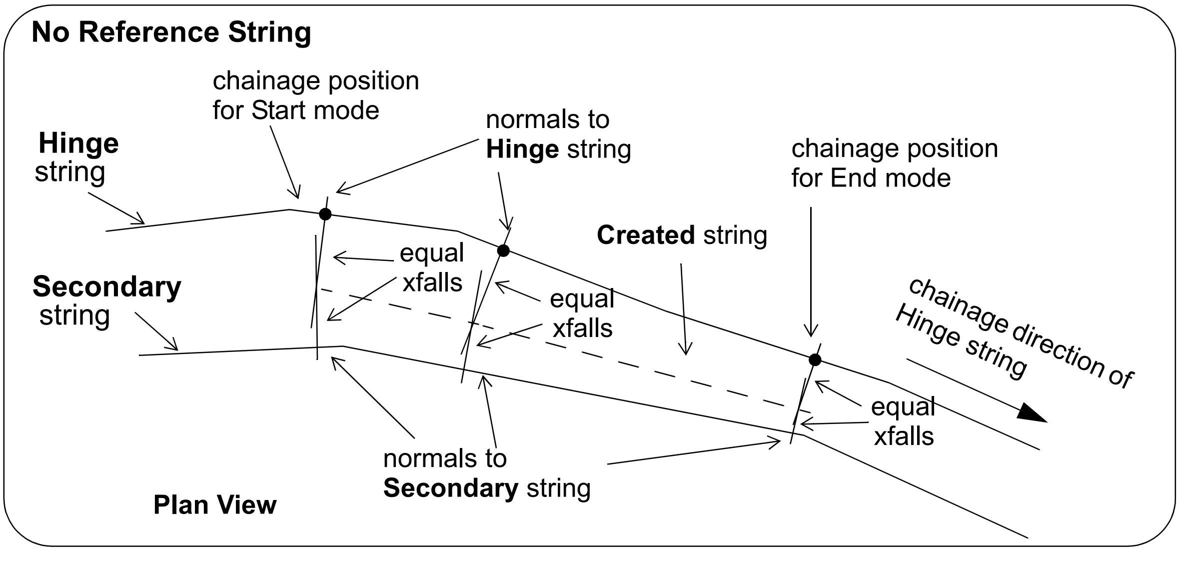


Chainage-Height Diagram for the Created String

On the section at Hinge chainage ch, the position of the vertex (with the calculated height) is such that the absolute value of the xfall from the vertex to the Hinge string, and the absolute xfall from the vertex to the **normal to the Secondary** string, is the same.

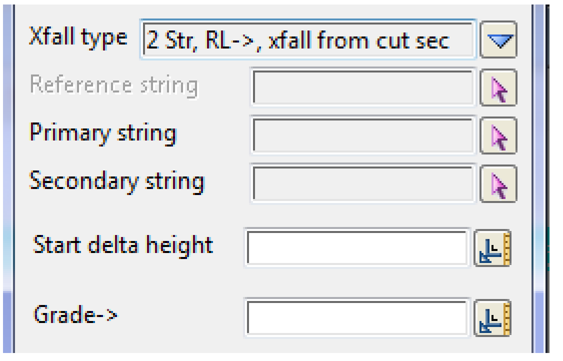
That is, a normal to the **Secondary** string is created that intersects in plan with the section from the Hinge string so that the absolute value of the xfall from the Hinge string to the vertex (with the calculated height), is the same as the absolute grade from the vertex to the Secondary string. This gives the plan position of the vertex of the created string.

Note - it may not be possible to find such an intersection point.



Continue to the next choice 20.8.11.1.20 2 Strings, Start Delta Height & Grade, Xfall from Secondary String, Normal to Reference or Hinge String or return to 20.8.11.1 Xfall Types or 20.8.11 String by Xfall and Grade.

20.8.11.1.20 2 Strings, Start Delta Height & Grade, Xfall from Secondary String, Normal to Reference or Hinge String



This option requires a **Hinge** and a **Secondary** string, a **Start delta height** and forward **Grade**, and an **optional Reference** string.

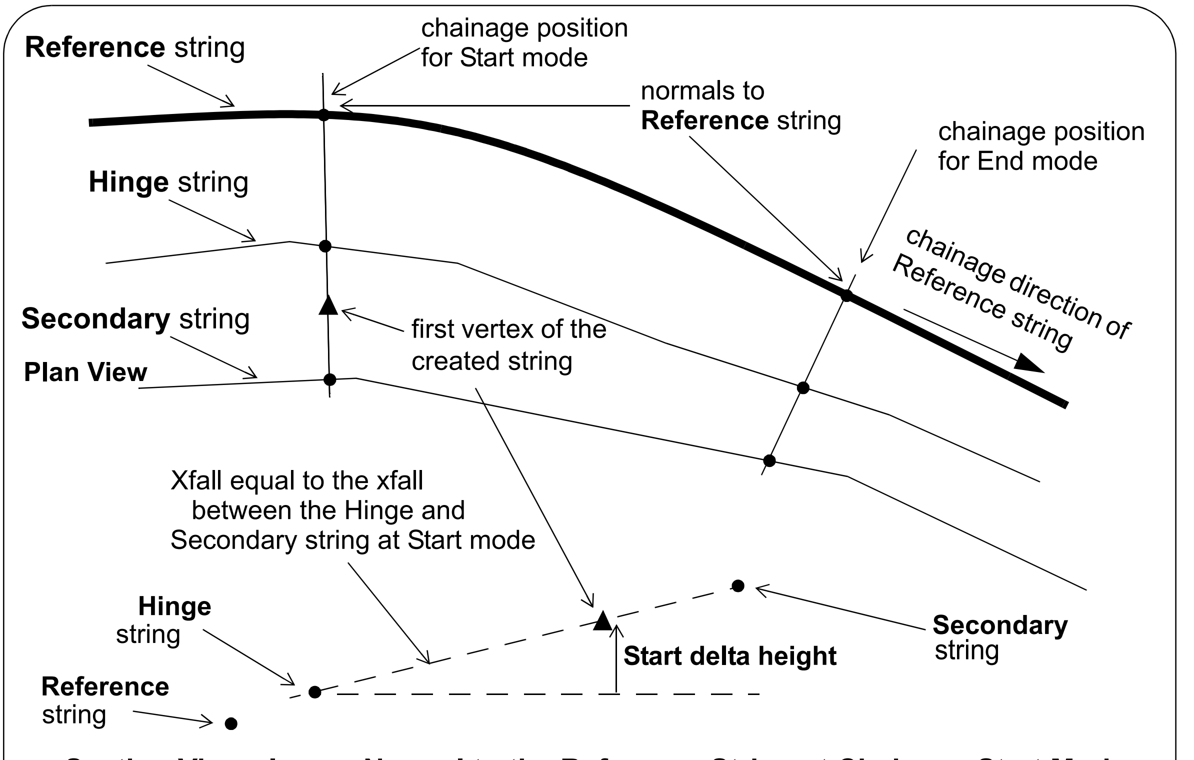
Reference String Selected

If a **Reference** string is selected then the **Start mode** and **End Mode** refer to the **Reference** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Reference string** and cutting the **Hinge** and the **Secondary** strings. The vertices of the created string lie on these normals.

The **first vertex** of the created string is on the normal to the Reference string at chainage **Start mode**, and its **height** is the height of the Hinge string on the Start mode normal, plus Start delta height.

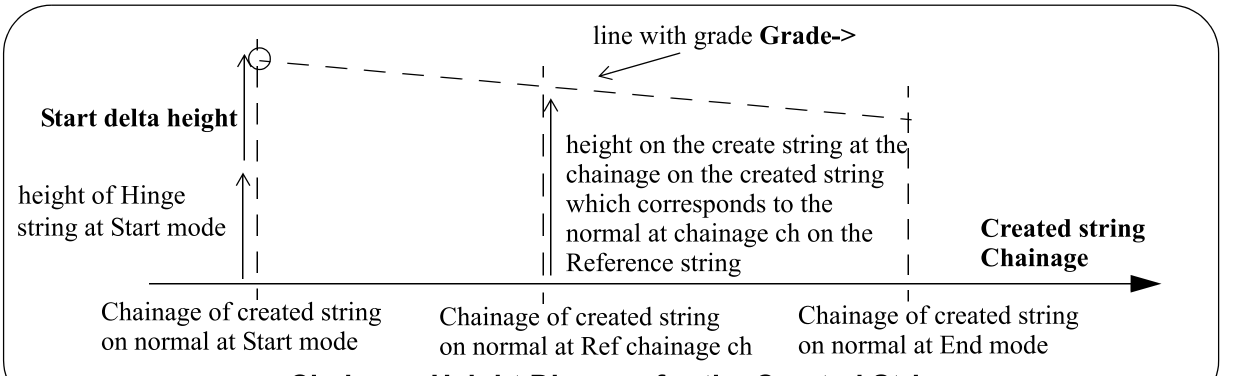
height of first vertex = height of Hinge string on Start mode normal + Start delta height

The plan position of the first vertex is on the Start mode normal, and with an offset from the Hinge string such that the Xfall is equal to the **xfall between the Hinge and Secondary strings**, and the change of height from the Hinge string is **Start delta height**.



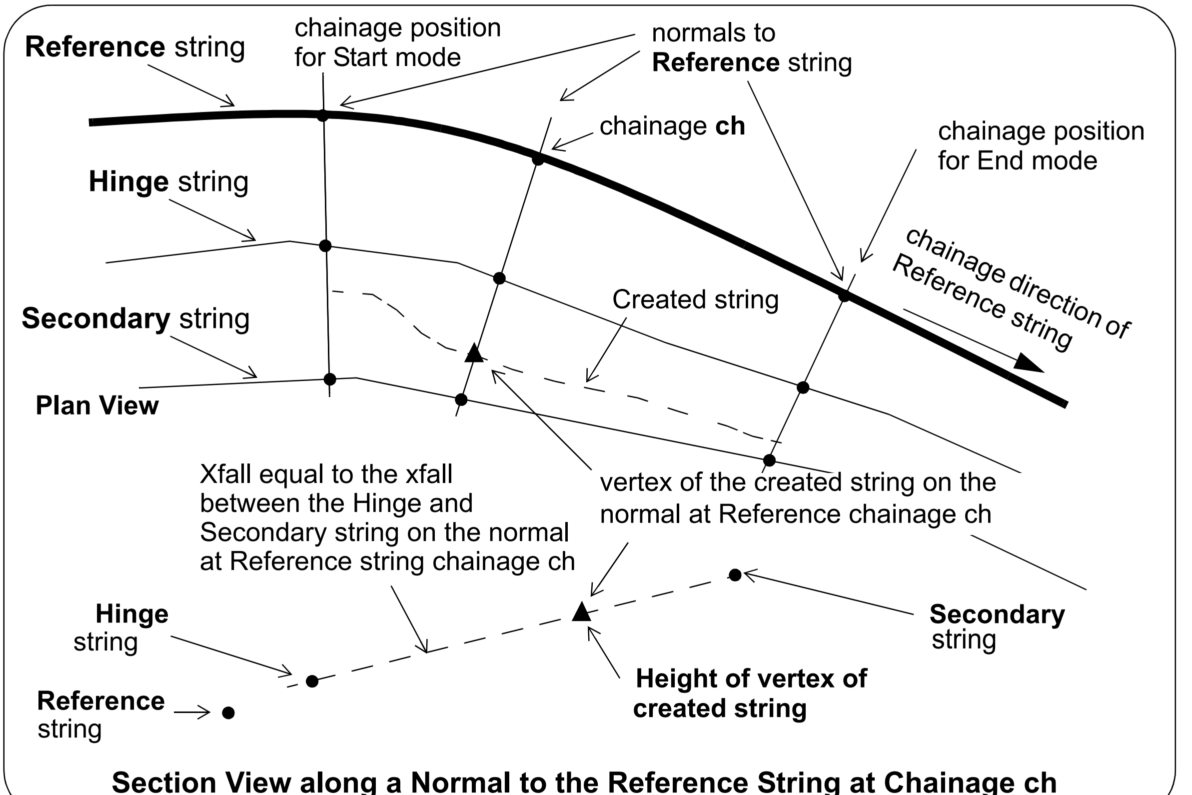
Section View along a Normal to the Reference String at Chainage Start Mode

The **heights** of the subsequent vertices of the created string are defined in (created string chainage, height) space, and lie on the line with grade ***Grade->*** and going through the first vertex.



Chainage-Height Diagram for the Created String

The **plan** position of the vertex of the created string on the normal at Reference chainage **ch** is such that it is on the line through the Hinge string with xfall equal to the **xfall between the Hinge and Secondary strings**, and having the calculated height.



Section View along a Normal to the Reference String at Chainage ch

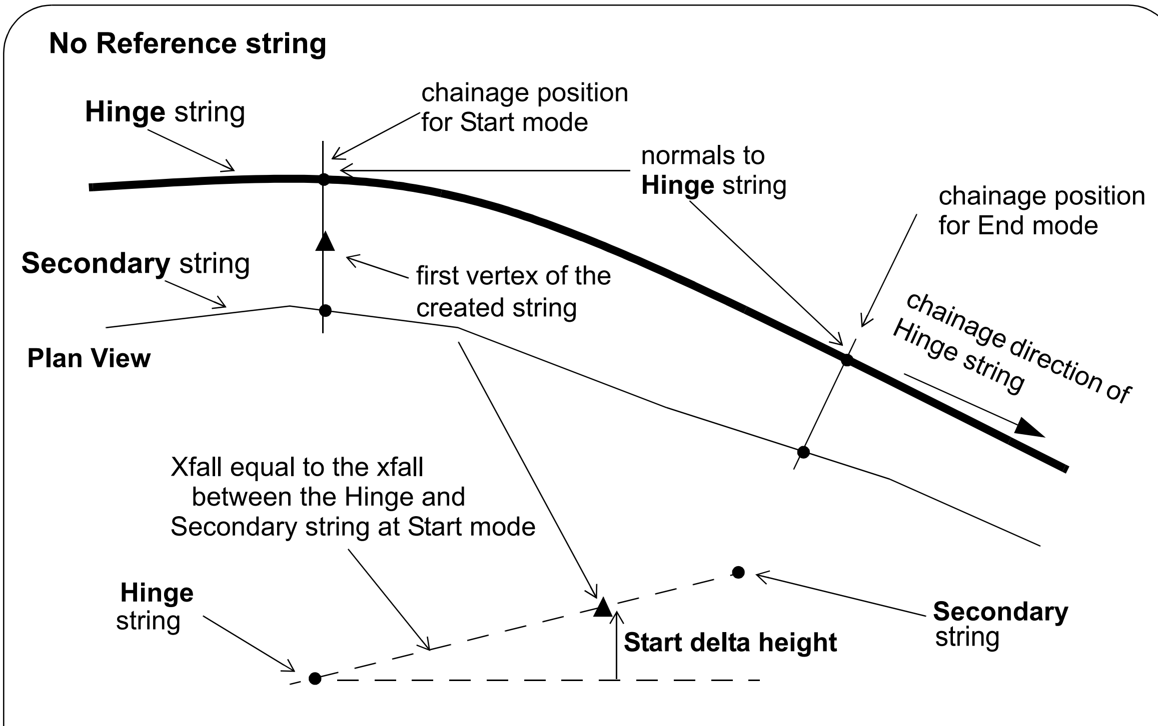
Reference String NOT Selected

If a **Reference** string is **NOT** selected then the **Start mode** and **End Mode** refer to the **Hinge** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Hinge string** and cutting the **Secondary** string. The vertices of the created string lie on these normals.

The **first vertex** of the created string is on the normal to the Hinge string at chainage **Start mode**, and its height is the height of the Hinge string on the Start mode normal, plus Start delta height.

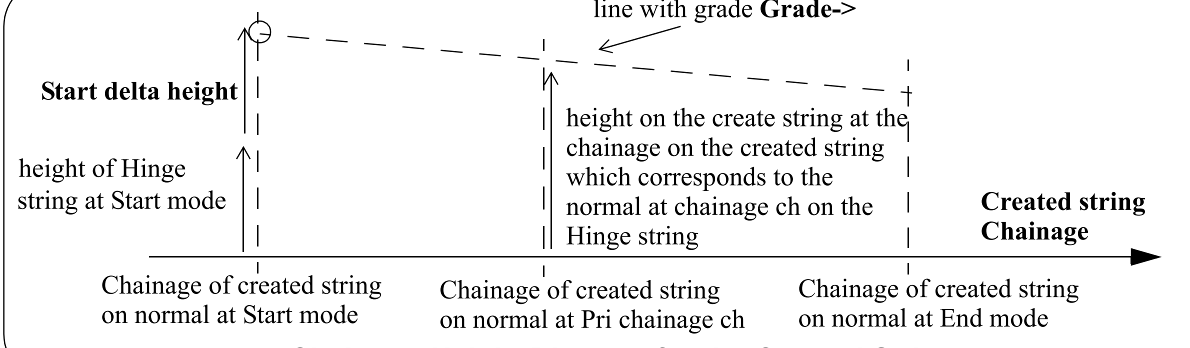
height of first vertex = height of Hinge string on Start mode normal + Start delta height

The plan position of the first vertex is on the Start mode normal, and with an offset from the Hinge string such that the Xfall is equal to the **xfall between the Hinge and Secondary strings**, and the change of height from the Hinge string is **Start delta height**.



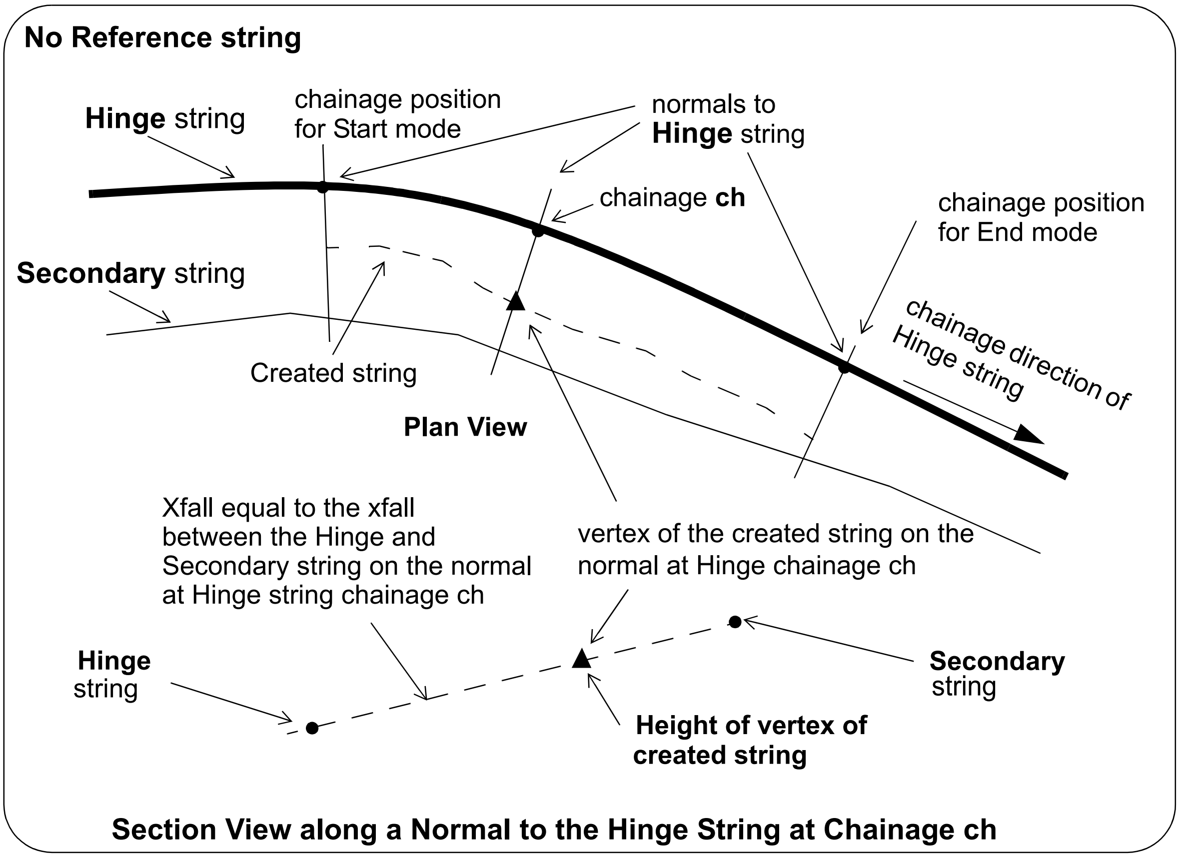
Section View along a Normal to the Hinge String at Chainage Start Mode

The **heights** of the subsequent vertices of the created string are defined in (created string chainage, height) space, and lie on the line with grade ***Grade->*** and going through the first vertex.



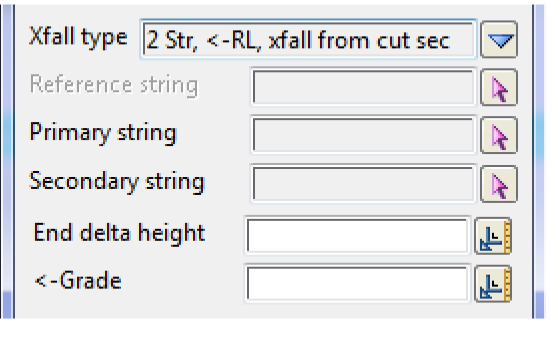
Chainage-Height Diagram for the Created String

The **plan** position of the vertex of the created string on the normal at Hinge chainage **ch** is such that it is on the line through the Hinge string with xfall equal to the **xfall between the Hinge and Secondary strings**, and having the calculated height.



Continue to the next choice 20.8.11.1.21 2 Strings, End Delta Height & Grade, Xfall from Secondary String, Normal to Reference or Hinge String or return to 20.8.11.1 Xfall Types or 20.8.11 String by Xfall and Grade.

20.8.11.1.21 2 Strings, End Delta Height & Grade, Xfall from Secondary String, Normal to Reference or Hinge String



This option requires a **Hinge** and a **Secondary** string, an **End delta height** and backward **Grade**, and an **optional Reference** string.

Reference String Selected

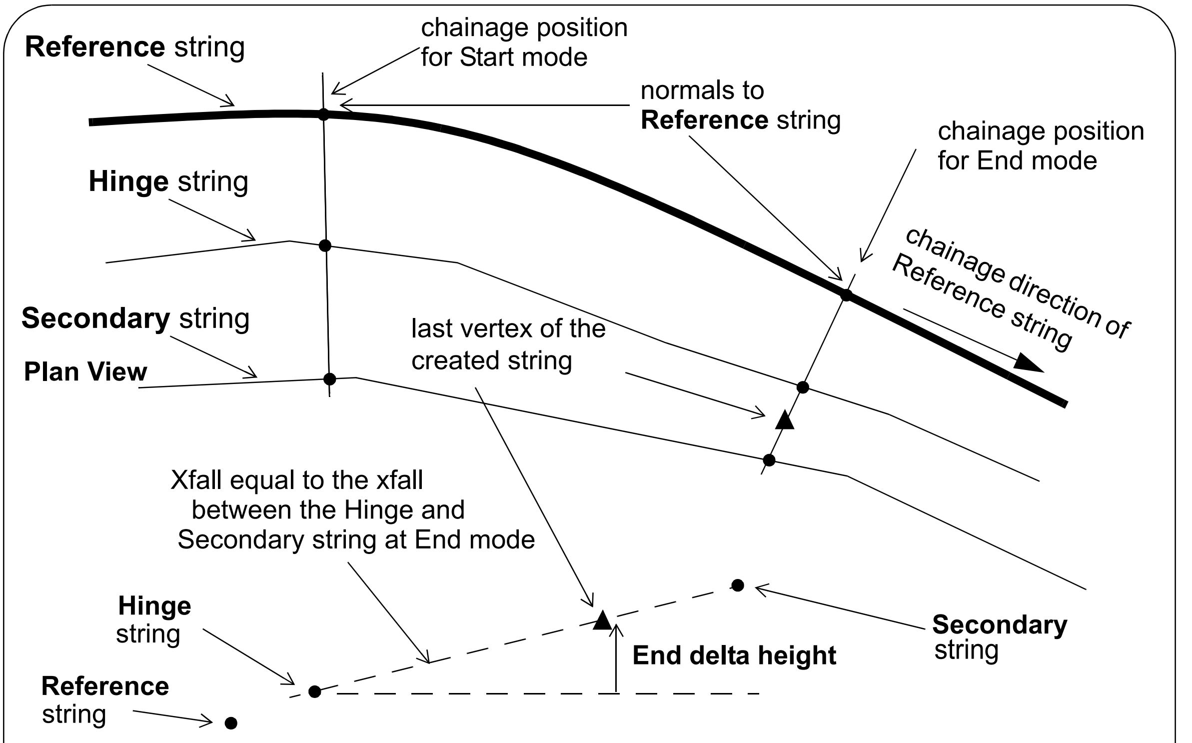
If a **Reference** string is selected then the **Start mode** and **End Mode** refer to the **Reference** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Reference string** and cutting the **Hinge** and the **Secondary** strings. The vertices of the created string lie on these normals.

The **last vertex** of the created string is on the normal to the Reference string at chainage **End**

mode, and its height is the height of the Hinge string on the End mode normal, plus End delta height.

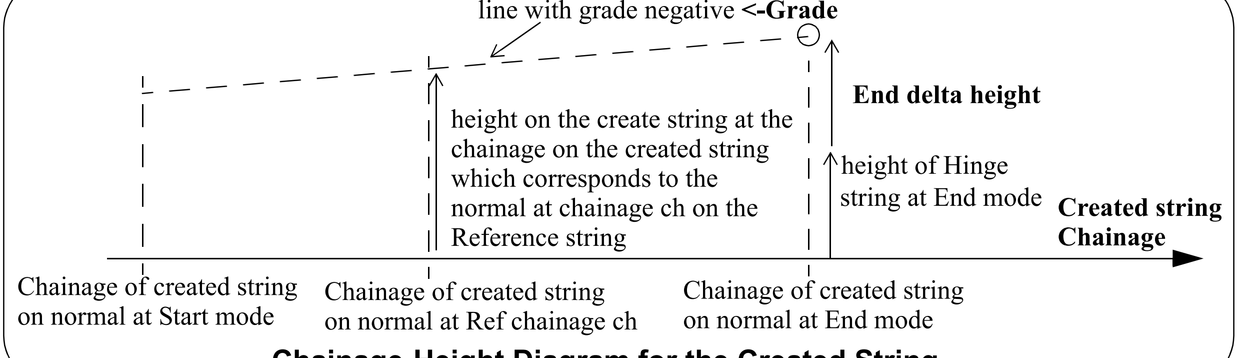
height of last vertex = height of Hinge string on End mode normal + End delta height

The plan position of the last vertex is on the End mode normal, and with an offset from the Hinge string such that the Xfall is equal to the **xfall between the Hinge and Secondary strings**, and the change of height from the Hinge string is **End delta height**.



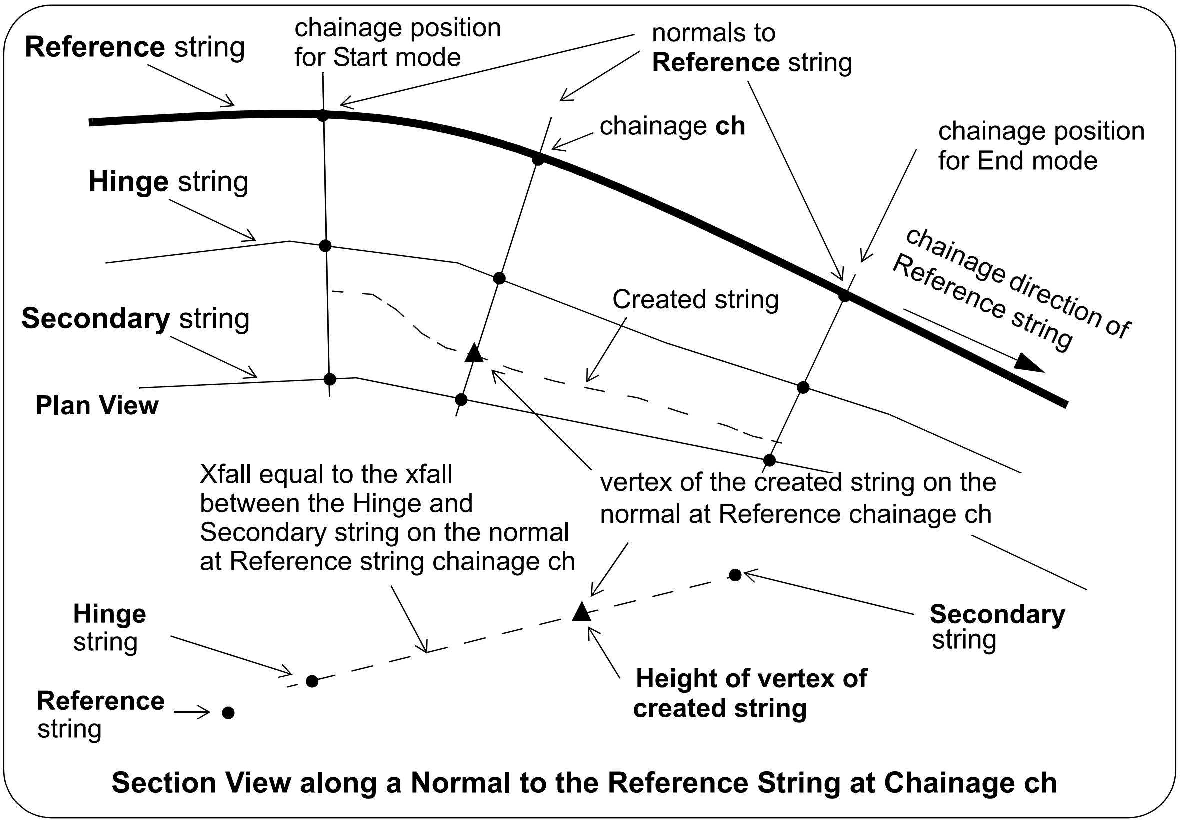
Section View along a Normal to the Reference String at Chainage End Mode

The **heights** of the previous vertices of the created string are defined in (created string chainage, height) space, and lie on the line with grade negative ***<-Grade*** and going through the last vertex.



Chainage-Height Diagram for the Created String

The **plan** position of the vertex of the created string on the normal at Reference chainage **ch** is such that it is on the line through the Hinge string with xfall equal to the **xfall between the Hinge and Secondary strings**, and having the calculated height.



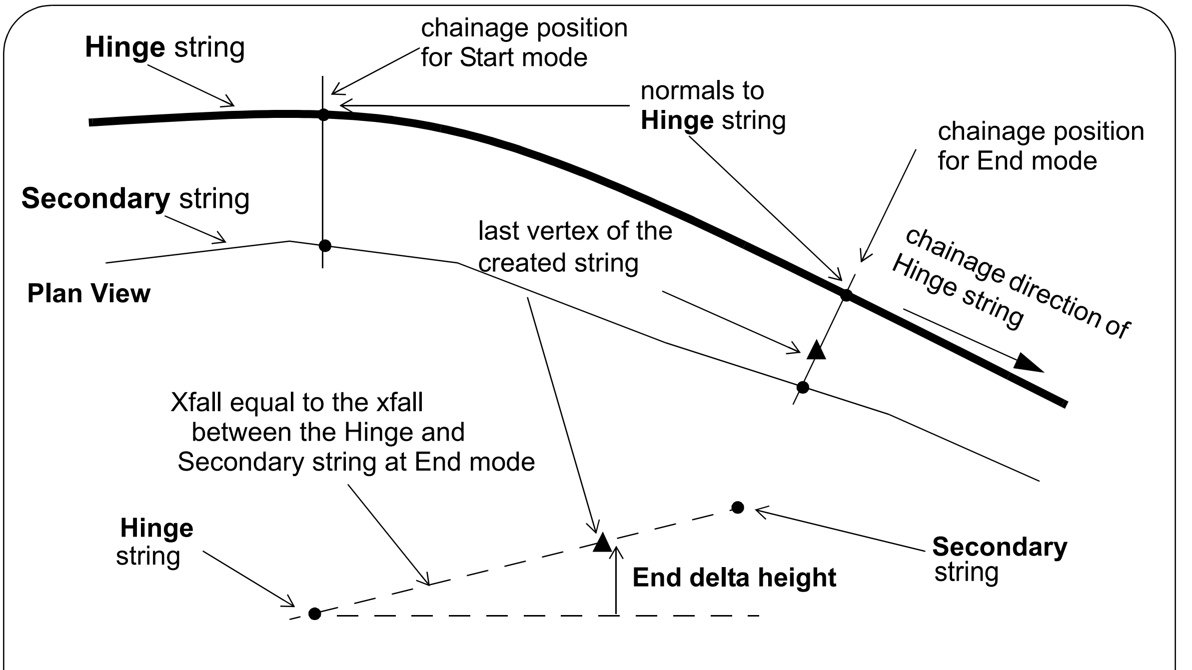
Reference String NOT Selected

If a **Reference** string is **NOT** selected then the **Start mode** and **End Mode** refer to the **Hinge** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Hinge string** and cutting the **Secondary** string. The vertices of the created string lie on these normals.

The **last vertex** of the created string is on the normal to the Hinge string at chainage **End mode**, and its height is the height of the Hinge string on the End mode normal, plus End delta height.

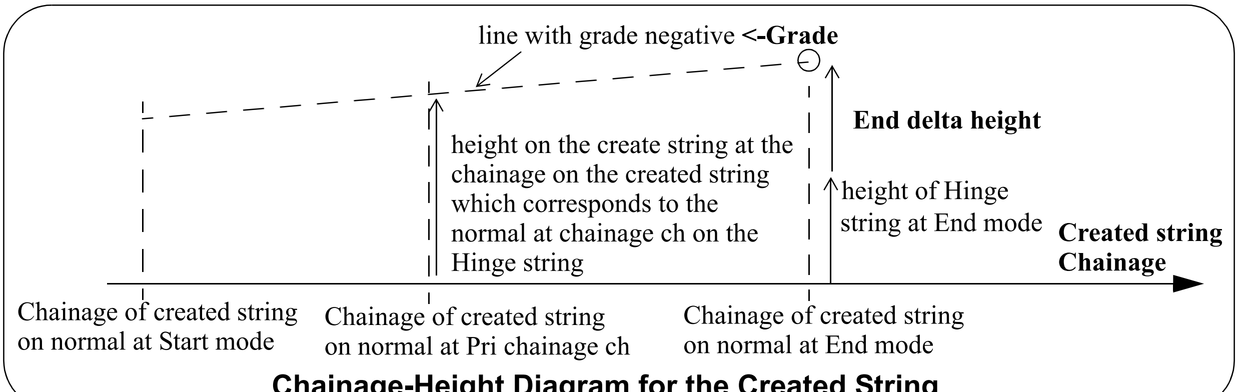
height of last vertex = height of Hinge string on End mode normal + End delta height

The plan position of the last vertex is on the End mode normal, and with an offset from the Hinge string such that the Xfall is equal to the **xfall between the Hinge and Secondary strings**, and the change of height from the Hinge string is **End delta height**.



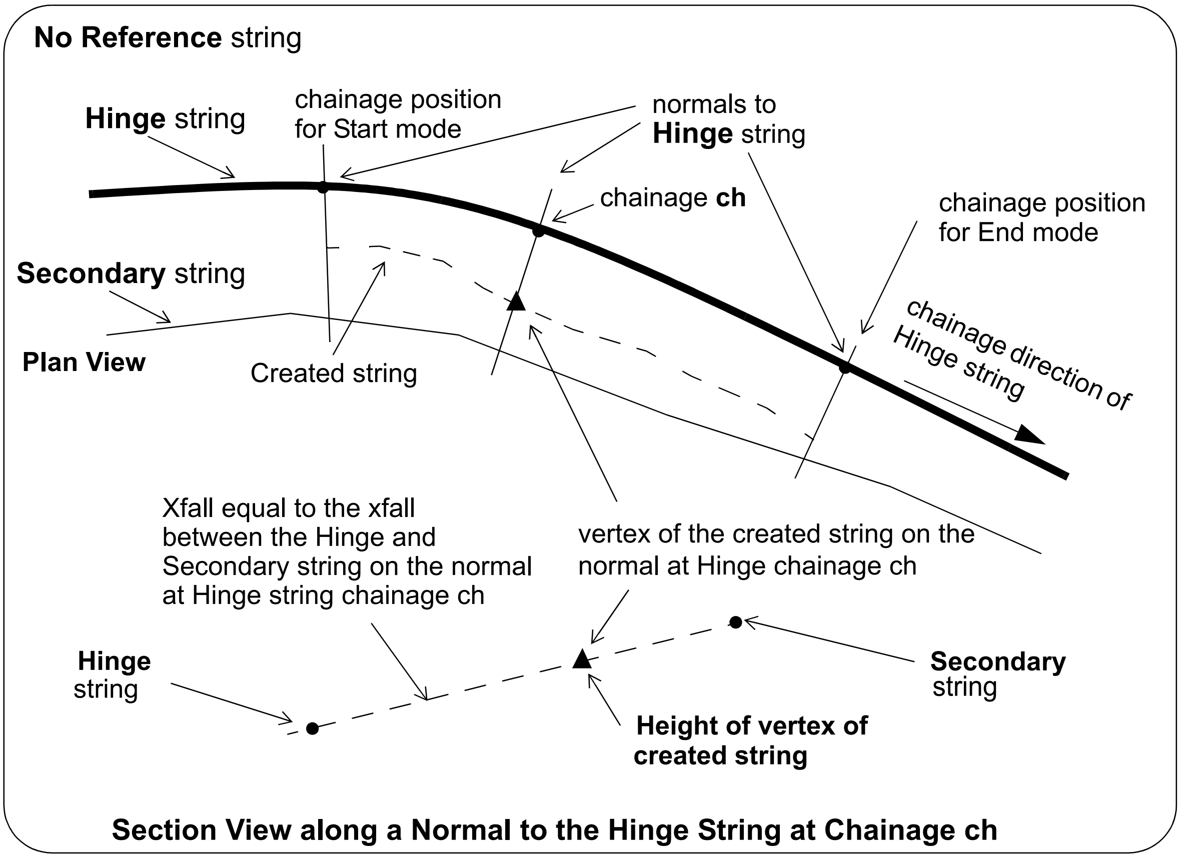
Section View along a Normal to the Hinge String at Chainage End Mode

The **heights** of the previous vertices of the created string are defined in (created string chainage, height) space, and lie on the line with grade negative ***<-Grade*** and going through the last vertex.



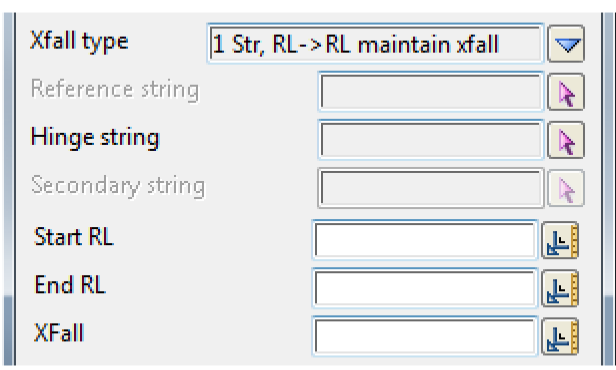
Chainage-Height Diagram for the Created String

The **plan** position of the vertex of the created string on the normal at Hinge chainage **ch** is such that it is on the line through the Hinge string with xfall equal to the **xfall between the Hinge and Secondary strings**, and having the calculated height.



Continue to the next choice 20.8.11.1.22 1 String, Start RL and End RL, Fixed Xfall or return to 20.8.11.1 Xfall Types or 20.8.11 String by Xfall and Grade.

20.8.11.1.22 1 String, Start RL and End RL, Fixed Xfall



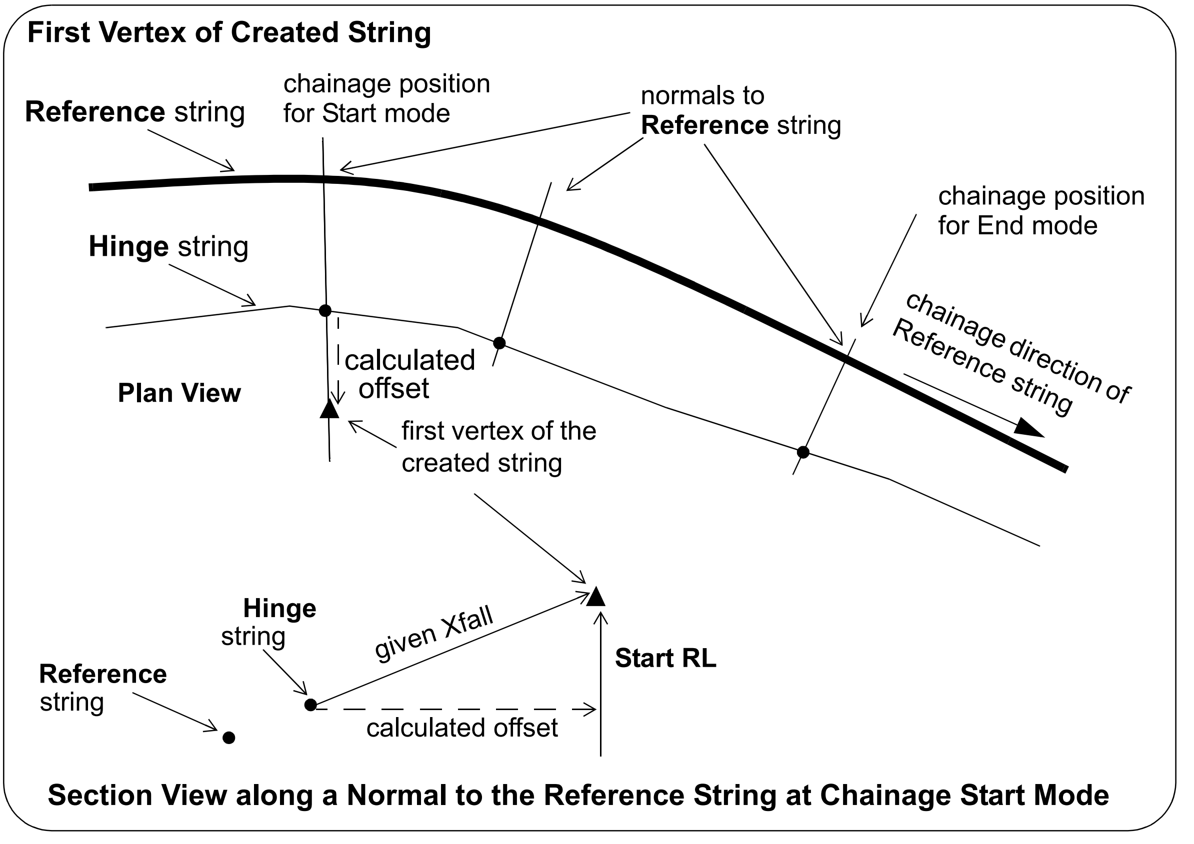
This option requires a **Hinge** string, a **Start RL** and **End RL**, a **Xfall**, and an **optional Reference** string.

Reference String Selected

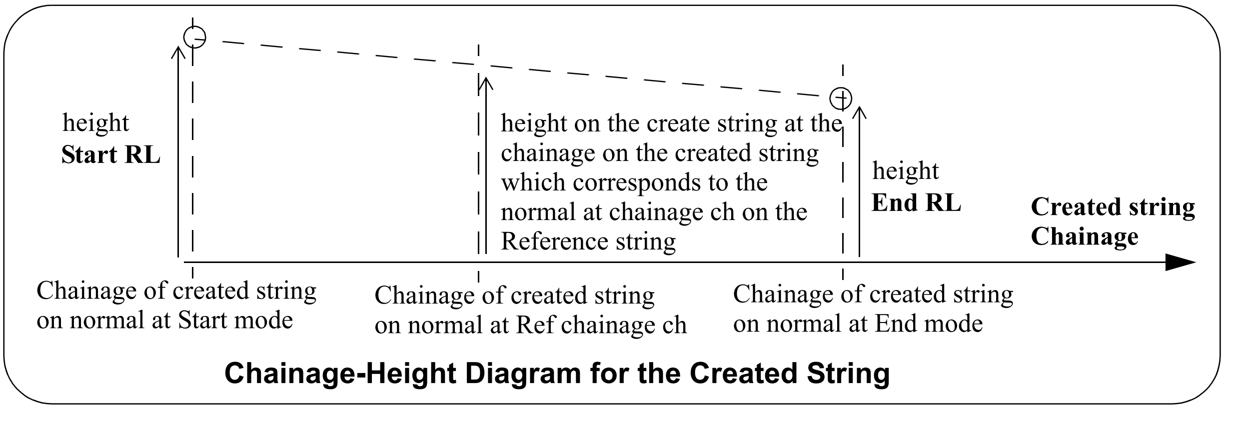
If a **Reference** string is selected then the **Start mode** and **End Mode** refer to the **Reference** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Reference string** and cutting the **Hinge** string. The vertices of the created string lie on these normals.

The **first vertex** of the created string is on the normal to the Reference string and has a **height**

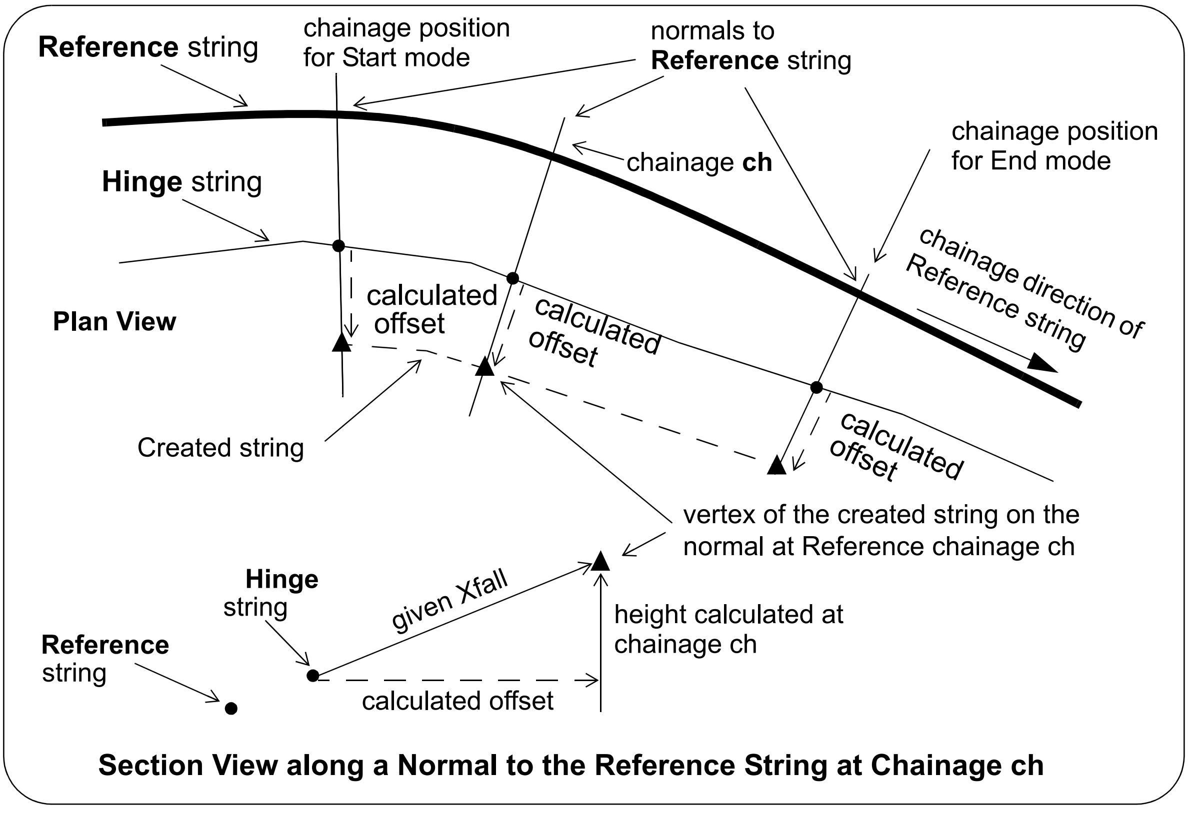
Start RL. The offset for the first vertex is calculated and is such that the given **Xfall** from Hinge string will give the vertex the height **Start RL**.



The **heights** of the subsequent vertices of the created string are defined in (created string chainage, height) space, and lie on the line with **Start RL,** and ending with **End RL** on the normal at End mode.



The **plan** position of the vertex of the created string on the normal at Reference chainage **ch** is constructed by going out from the hinge string with xfall equal to the **Xfall**, and for an offset so that the height is equal to the required height at chainage **ch**.

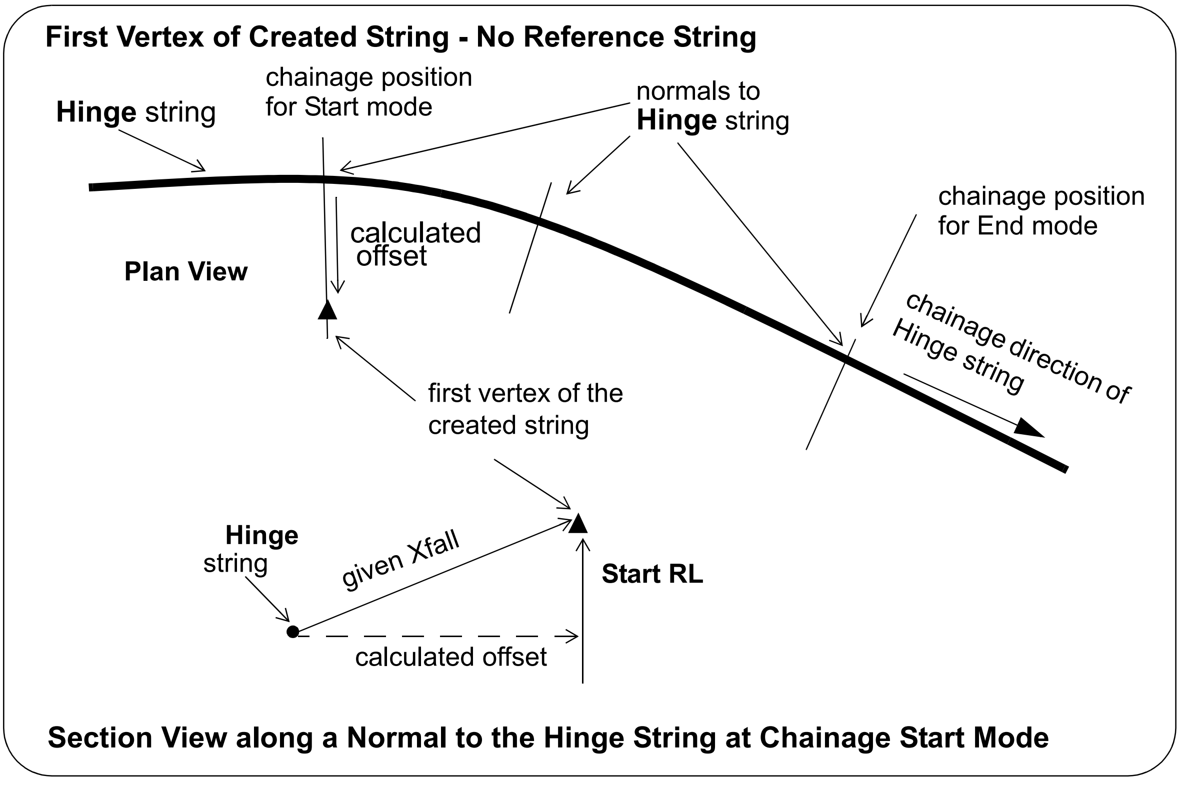


For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.

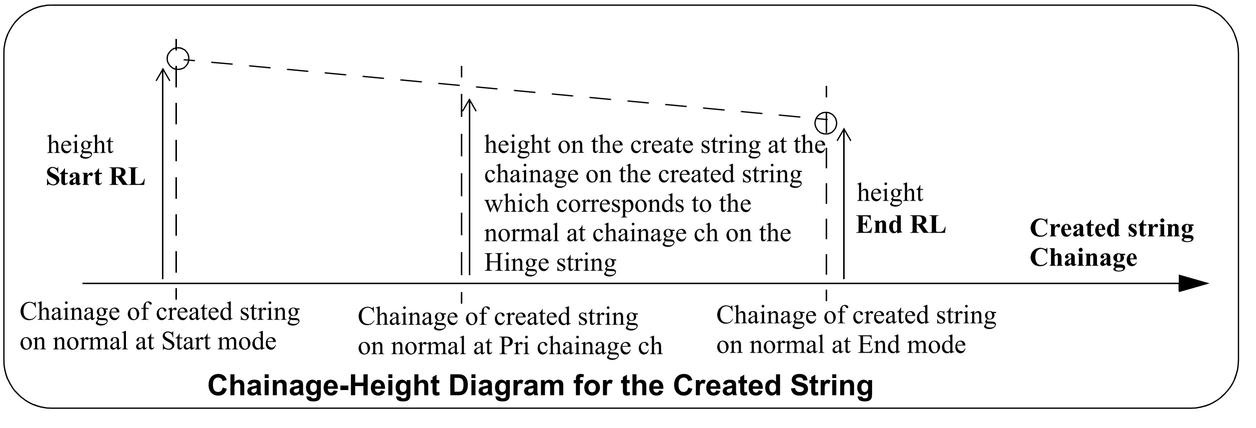
Reference String NOT Selected

If a **Reference** string is **NOT** selected then the **Start mode** and **End Mode** refer to the **Hinge** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Hinge string**. The vertices of the created string lie on these normals.

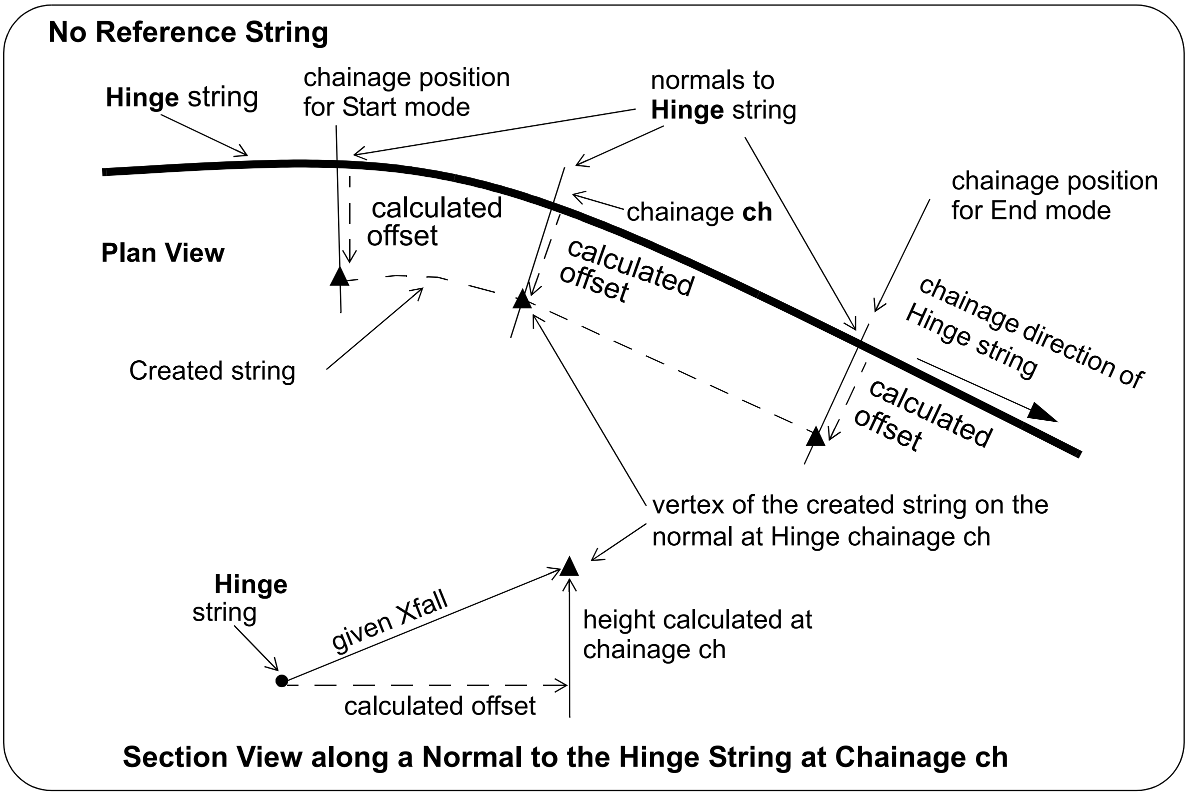
The **first vertex** of the created string is on the normal to the Hinge string and has a **height Start RL**. The offset for the first vertex is calculated and is such that the given **Xfall** from Hinge string will give the vertex the height **Start RL**.



The **heights** of the subsequent vertices of the created string are defined in (created string chainage, height) space, and lie on the line with **Start RL,** and ending with **End RL** on the normal at End mode.



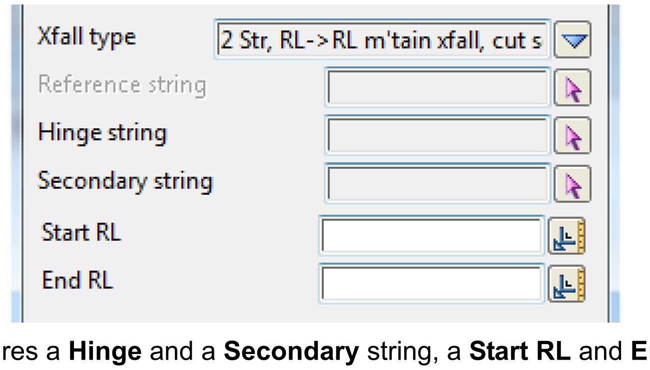
The **plan** position of the vertex of the created string on the normal at Hinge chainage **ch** is constructed by going out from the hinge string with xfall equal to the **Xfall** and for an offset so that the height is equal to the required height at chainage **ch**.



For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.

Continue to the next choice 20.8.11.1.23 2 Strings, Start RL and End RL, Xfall from Secondary String, Normal to Reference or Hinge String or return to 20.8.11.1 Xfall Types or 20.8.11 String by Xfall and Grade.

20.8.11.1.23 2 Strings, Start RL and End RL, Xfall from Secondary String, Normal to Reference or Hinge String

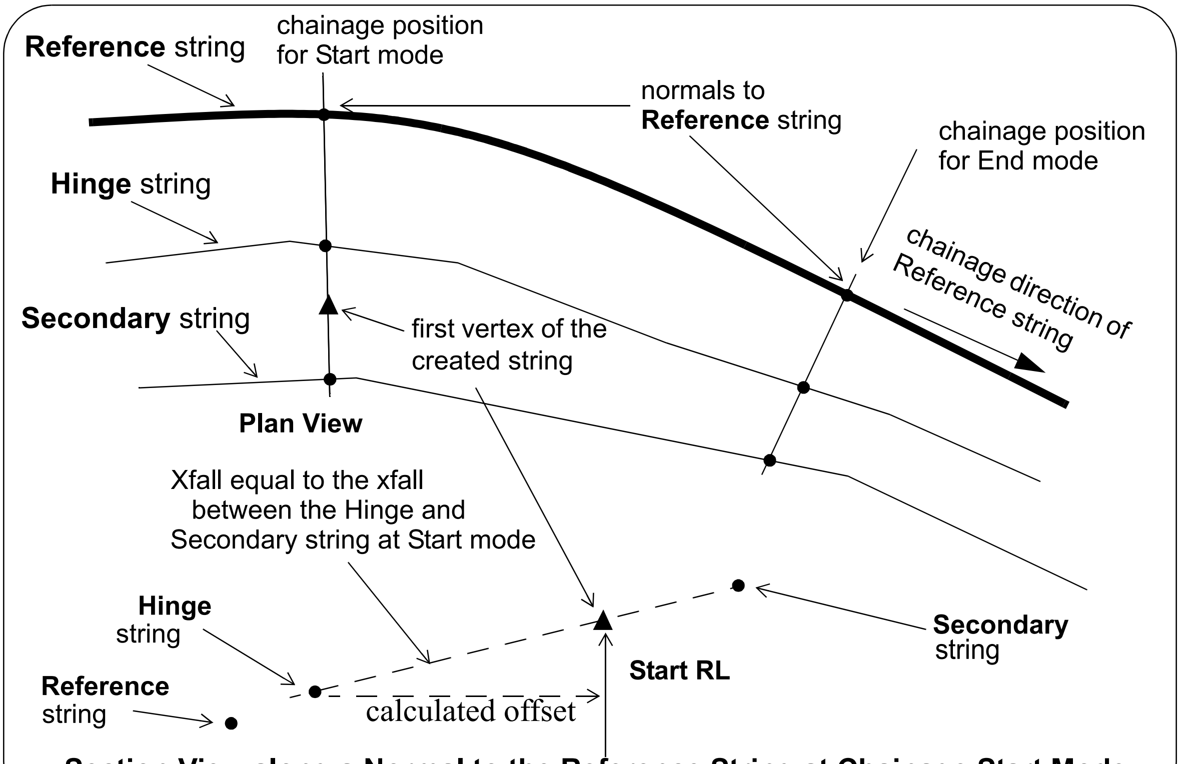


This option requires a **Hinge** and a **Secondary** string, a **Start RL** and **End RL**, and an **optional Reference** string.

Reference String Selected

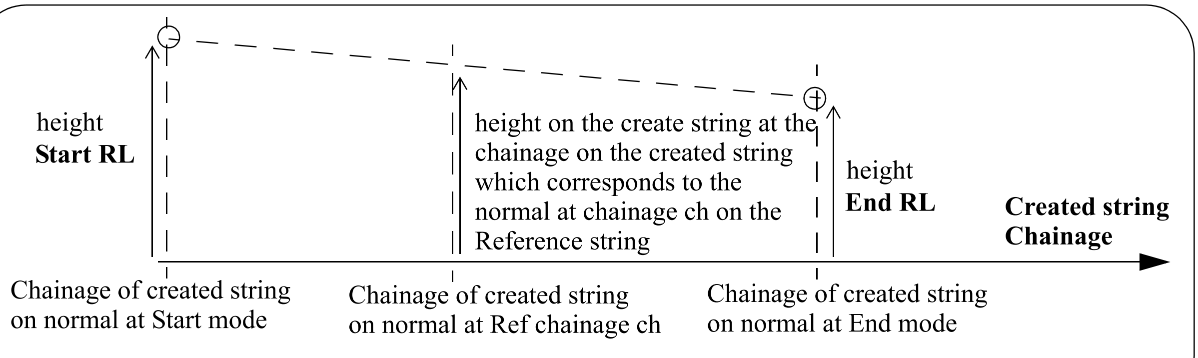
If a **Reference** string is selected then the **Start mode** and **End Mode** refer to the **Reference** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Reference string** and cutting the **Hinge** and the **Secondary** strings. The vertices of the created string lie on these normals.

The **first vertex** of the created string is on the normal to the Reference string at chainage **Start mode**, with height **Start RL**. The plan position of the first vertex is constructed by going out from the hinge string with xfall equal to the **xfall between the Hinge and Secondary strings**, and for the required offset so that the height is **Start RL**.



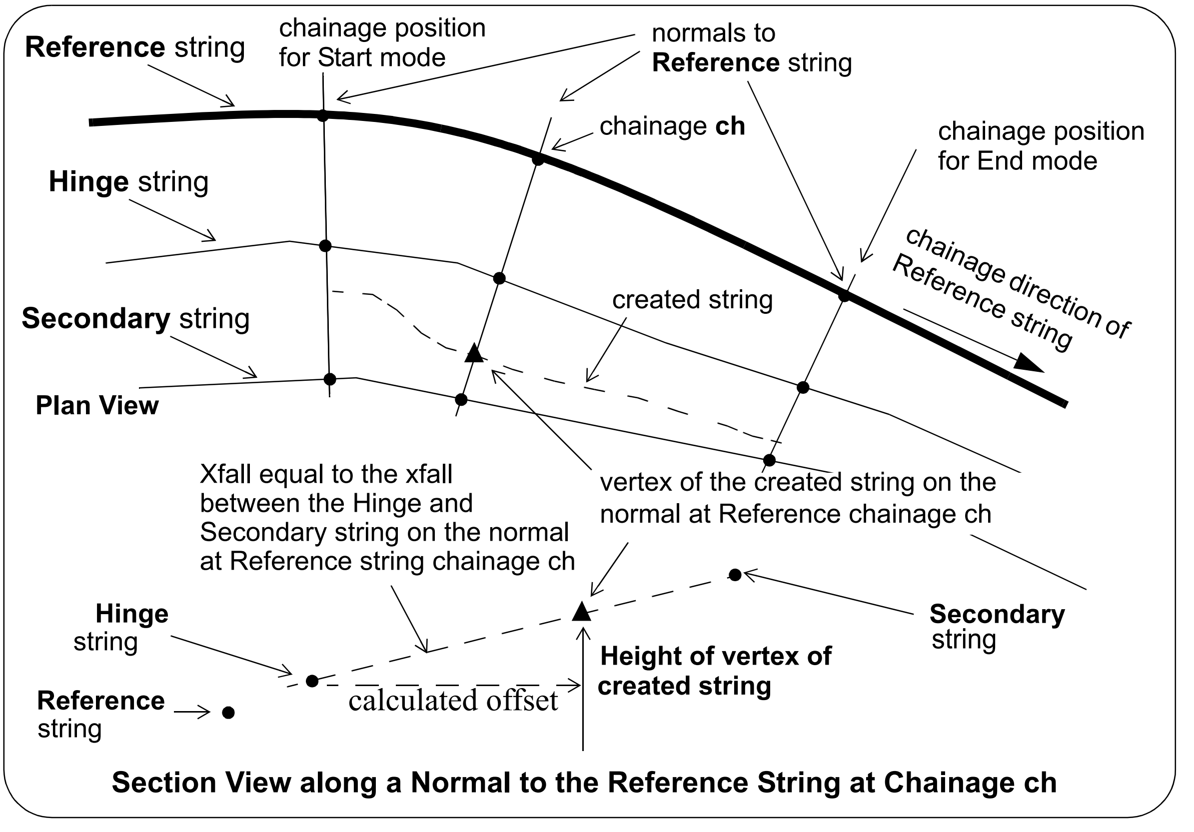
Section View along a Normal to the Reference String at Chainage Start Mode

The **heights** of the subsequent vertices of the created string are defined in (created string chainage, height) space, and lie on the line with **Start RL,** and ending with **End RL** on the normal at End mode.



Chainage-Height Diagram for the Created String

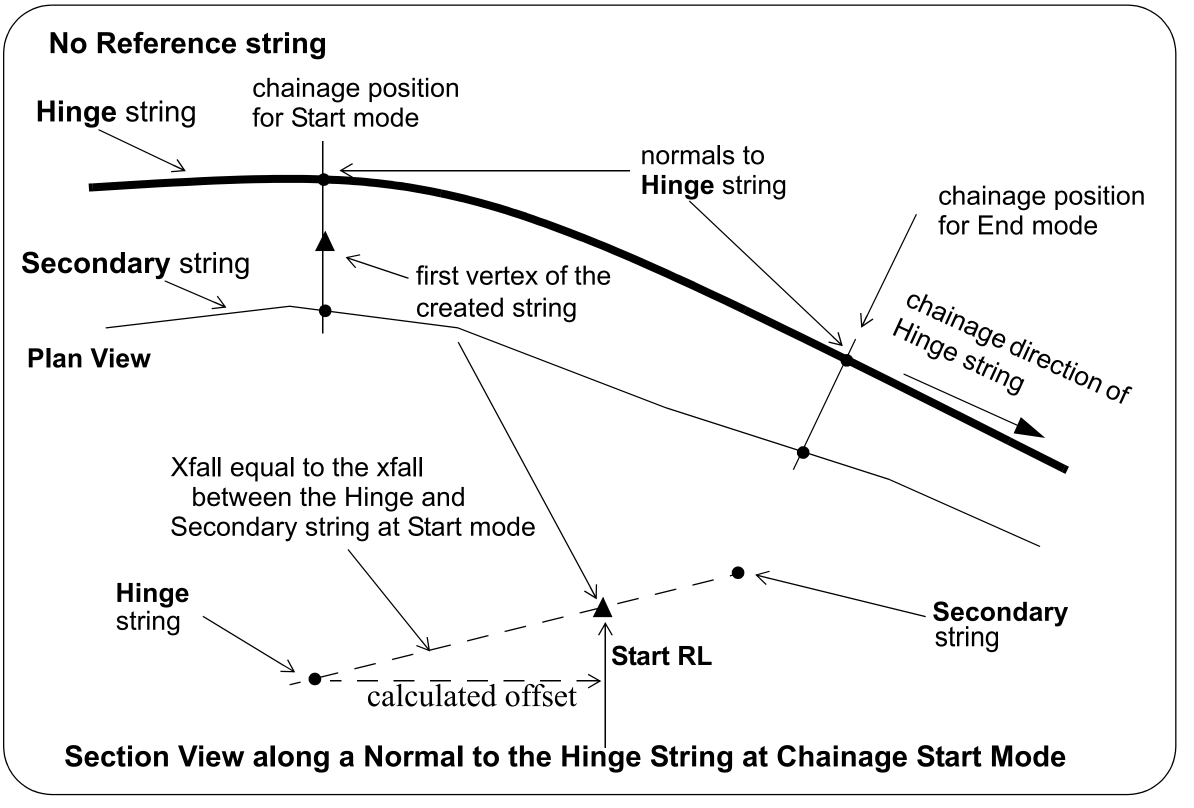
The **plan** position of the vertex of the created string on the normal at Reference chainage **ch** is constructed by going out from the hinge string with xfall equal to the **xfall between the Hinge and Secondary strings** at chainage **ch**, and for an offset so that the height is equal to the required height at chainage **ch**.



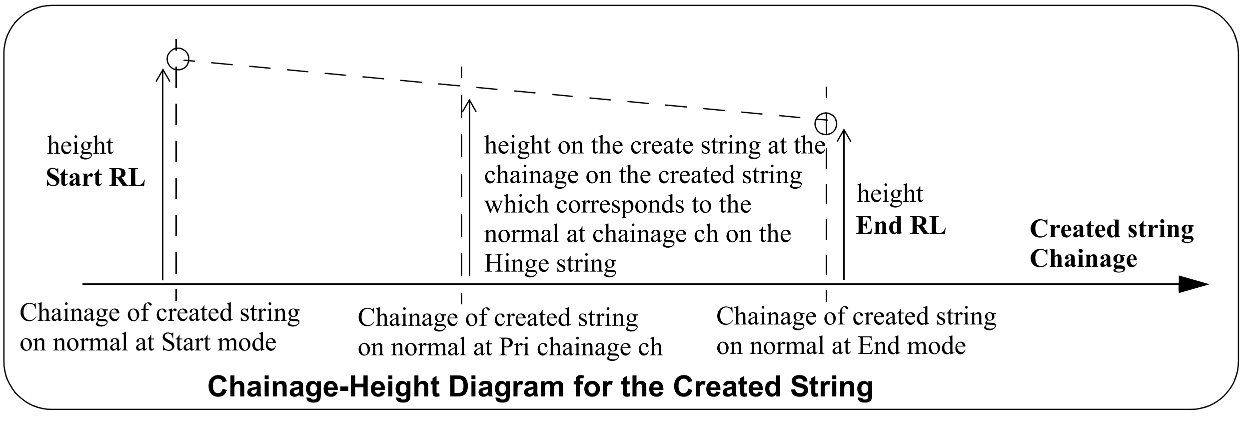
Reference String NOT Selected

If a **Reference** string is **NOT** selected then the **Start mode** and **End Mode** refer to the **Hinge** string, and at the given chainage interval between the Start mode and the End mode, sections are taken **normal** (perpendicular) to the **Hinge string** and cutting the **Secondary** string. The vertices of the created string lie on these normals.

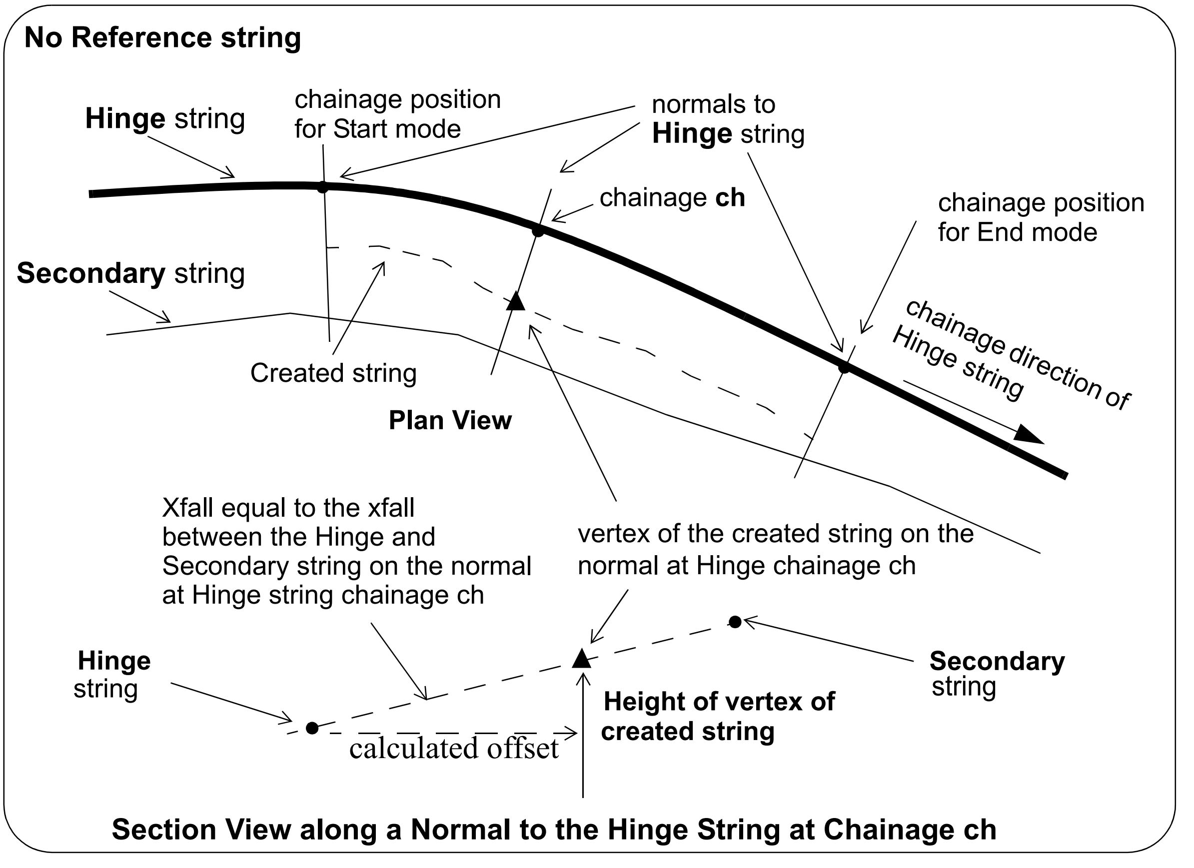
The **first vertex** of the created string is on the normal to the Hinge string at chainage **Start mode**, with height **Start RL**. The plan position of the first vertex is constructed by going out from the hinge string with xfall equal to the **xfall between the Hinge and Secondary strings**, and for the required offset so that the height is **Start RL**.



The **heights** of the subsequent vertices of the created string are defined in (created string chainage, height) space, and lie on the line with **Start RL,** and ending with **End RL** on the normal at End mode.



The **plan** position of the vertex of the created string on the normal at Hinge chainage **ch** is constructed by going out from the hinge string with xfall equal to the **xfall between the Hinge and Secondary strings** at chainage **ch**, and for an offset so that the height is equal to the required height at chainage **ch**.



Continue to the next section 20.8.12 More Roads or return to 20.8.11.1 Xfall Types or 20.8.11 String by Xfall and Grade.

20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade

Because of the number of different ways **Create by Xfall and Grade** can be used, there is a strict sign convention followed for heights, offsets and xfalls.

In all cases, a positive height or delta height is **up** and a negative height or delta height is **down**.

For any string, there is a definition for the left and right of a string (see Left and Right Side of a String), and there is a definition of Offset and Xfall for just that string (see Offset and Offset Distance for a String.)

However, ***Create by Xfall and Grade*** involves not just one string, but often two or three strings, so needs a standard sign convention of Offset and Xfall.

In **Create by Xfall and Grade**, if a Reference string is selected, then the direction of the Reference string is paramount and the sign convention for Offset and Xfall for a Hinge and Secondary string refers back to the Reference string. See Offset and Xfall - When Controlled by a Reference String.

In **Create by Xfall and Grade**, if a Reference string is NOT selected, then the direction of the Hinge string is paramount and the sign convention for Offset and Xfall for the Hinge and Secondary string refer back to the Hinge string. See Offset and Xfall - No Reference String and Controlled by Hinge String.

See Height
See Left and Right Side of a String.
See Offset and Offset Distance for a String
See Xfall for a String
See Offset and Xfall - When Controlled by a Reference String
See Offset and Xfall - No Reference String and Controlled by Hinge String

Height

For any string a positive height or delta height is up and a negative height or delta height is down.

Left and Right Side of a String

For any string:

the **right** side of the string is defined to be to the right of the string when travelling down the string in the **direction of the increasing chainage.**

the **left** side of the string is defined to be to the left of the string when travelling down the string in the **direction of the increasing chainage.**

Offset and Offset Distance for a String

Offset at chainage ch on a particular string is plan distance measured normal (perpendicular) to the string, starting at zero on the string and **positive** when going out to the **right** of the string, and **negative** when going out to the **left** of the string. Offset is only a (x,y) plane distance. It does not involve z values.

The **offset distance** of a point (x,y,z) from a particular string is the plan distance from the point (x,y) to the position on the string calculated by dropping the point (x,y) perpendicularly in the (x,y) plane onto the string. The offset distance is positive if the point (x,y,z) is on the right of the string, and negative is the point (x,y,z) is on the left of the string.

Xfall for a String

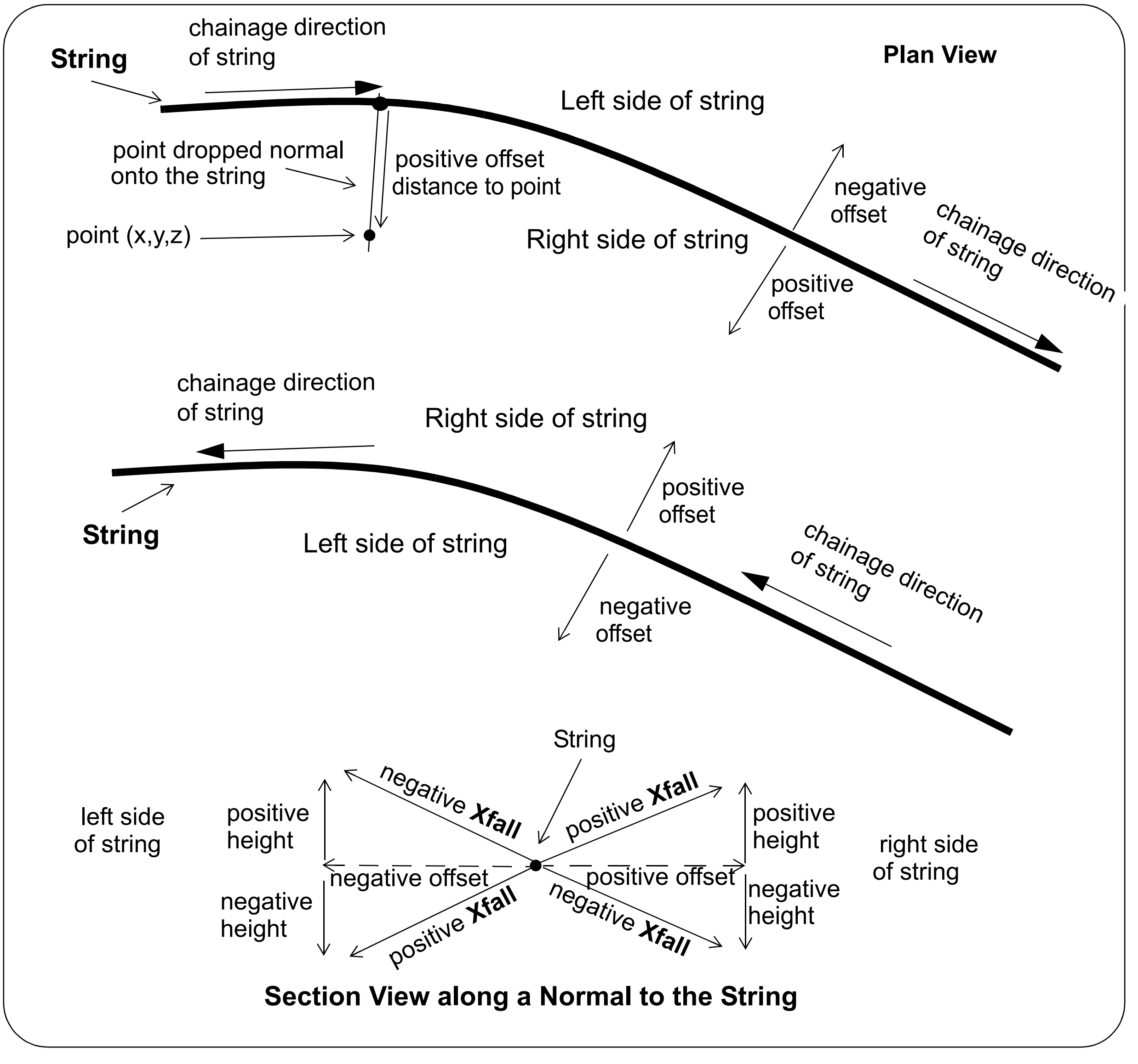
For **Xfall**:

a **xfall** is **positive** when it is going up when measured in the direction of increasing positive offset. That is, it is going up when going to the **right** of the string.

a **xfall** is **negative** when it is going down when measured in the direction of increasing positive offset. That is, it is going down when going to the **right** of the string.

a **xfall** is **negative** when it is going up in the direction of increasing negative offset (decreasing positive offset). That is, it is going up when going to the **left** of the string.

a **xfall** is **positive** when it is going down in the direction of increasing negative offset (decreasing positive offset). That is, it is going down when going to the **left** of the string.



Offset and Xfall - When Controlled by a Reference String

Offset

When a Reference string is selected, the **Reference** string is used to determine the sign convention for Offset and Xfall for the Hinge and Secondary strings. Hence the direction of the Hinge and Secondary strings are not used.

When a Reference string is selected:

an **offset** is **positive** when it is going in the direction of increasing positive offset with respect to the Reference string. That is, it is moving in the direction of going to the **right** of the Reference string.

an **offset** is **negative** when it is going in the direction of increasing negative offset (decreasing positive offset) with respect to the Reference string. That is, it is moving in the direction of going to the **left** of the Reference string.

Xfall

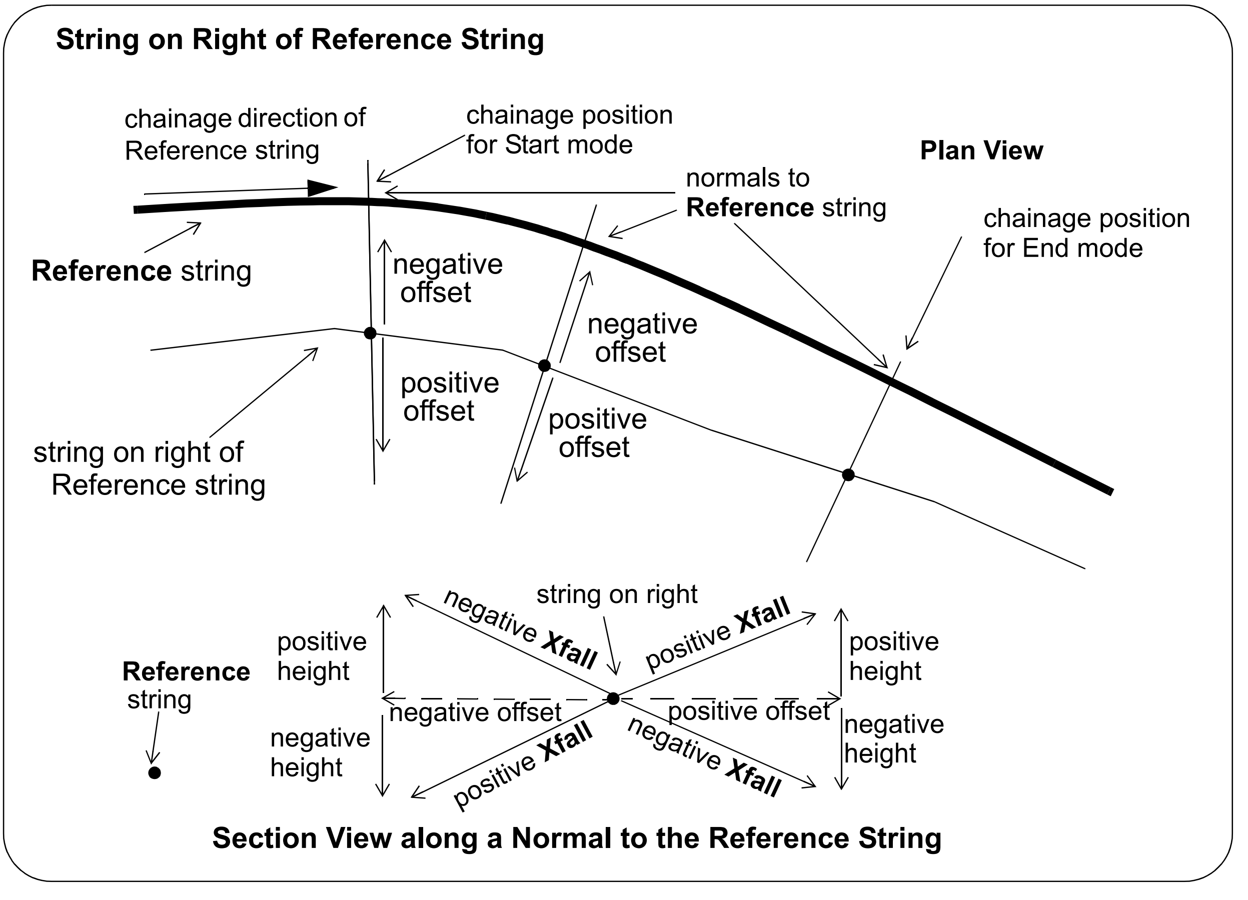
When a Reference string is selected:

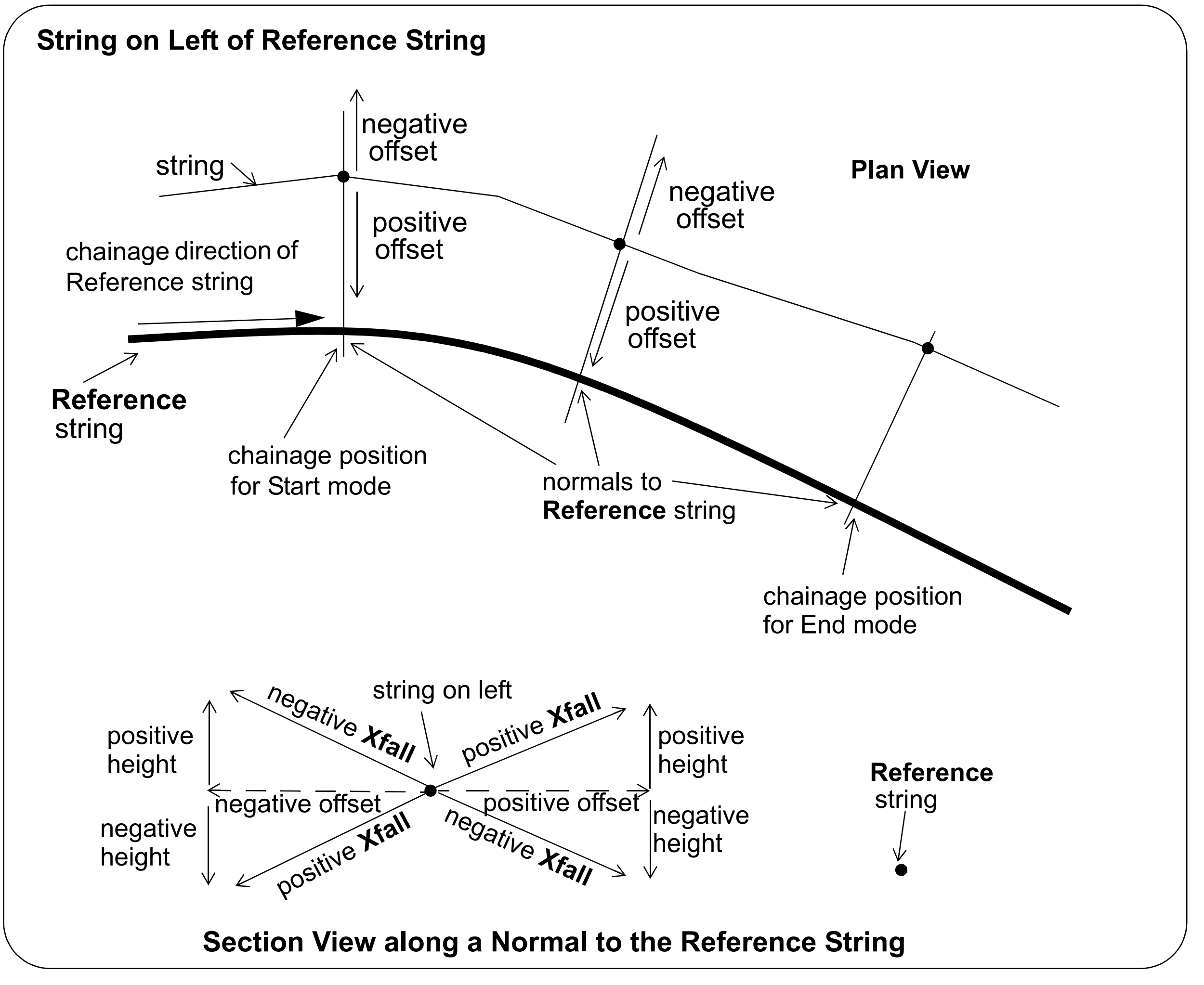
a **xfall** is **positive** when it is going up when measured in the direction of increasing positive offset with respect to the Reference string. That is, it is going up when moving in the direction of going to the **right** of the Reference string.

a **xfall** is **negative** when it is going down when measured in the direction of increasing positive offset with respect to the Reference string. That is, it is going down when moving in the direction of going to the **right** of the Reference string.

a **xfall** is **negative** when it is going up when measured in the direction of increasing negative offset (decreasing positive offset) with respect to the Reference string. That is, it is going up when moving in the direction of going to the **left** of the Reference string.

a **xfall** is **positive** when it is going down in the direction of increasing negative offset (decreasing positive offset) with respect to the Reference string. That is, it is going down when moving in the direction of going to the **left** of the Reference string.





Offset and Xfall - No Reference String and Controlled by Hinge String

When no Reference string is selected, then the Hinge string is used to determine the sign convention for Offset and Xfall for the Hinge and Secondary strings. Hence the direction of the Secondary strings is not used.

Offset

When no Reference string is selected then the Hinge string is used and:

an **offset** is **positive** when it is going in the direction of increasing positive offset with respect to the Hinge string. That is, it is moving in the direction of going to the **right** of the Hinge string.

an **offset** is **negative** when it is going in the direction of increasing negative offset (decreasing positive offset) with respect to the Hinge string. That is, it is moving in the direction of going to the **left** of the Hinge string.

Xfall

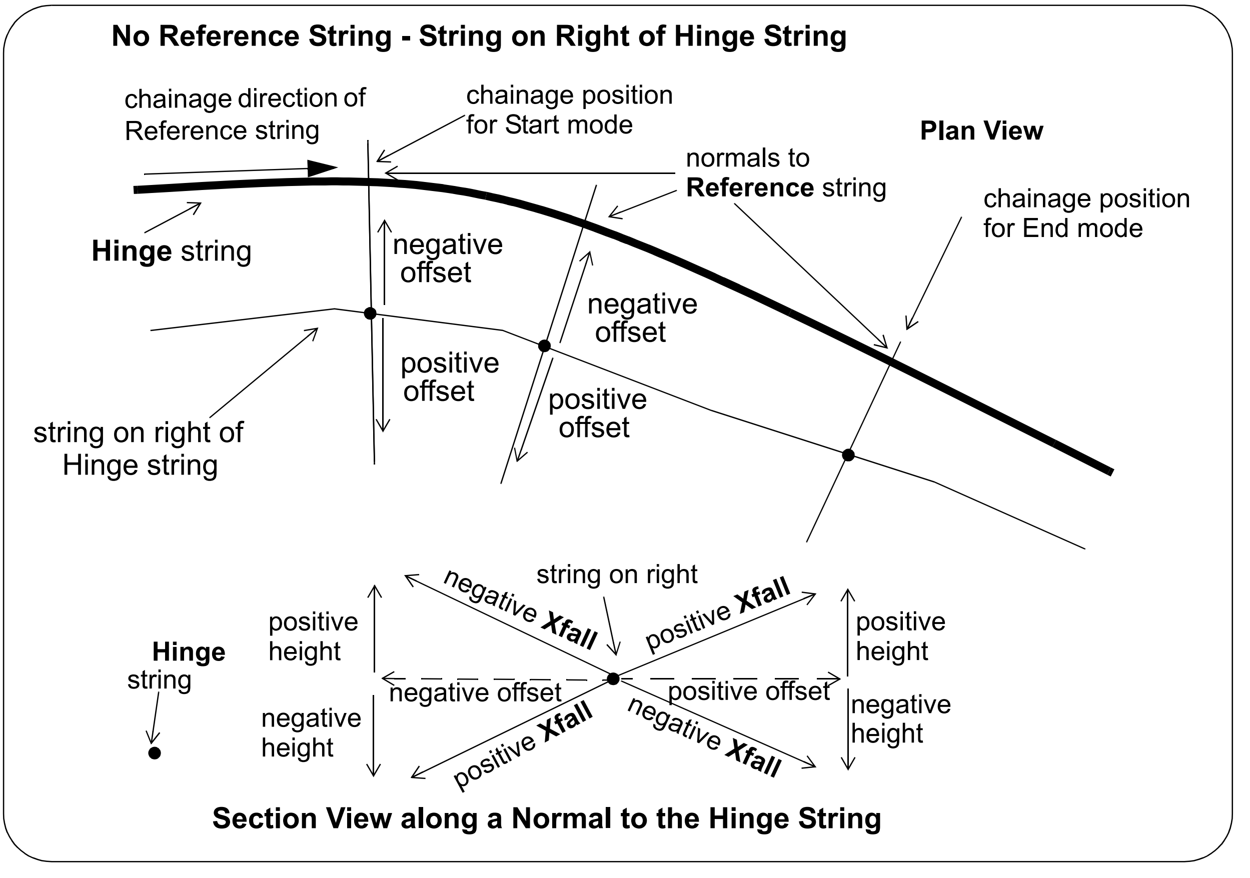
When no Reference string is selected then the Hinge string is used and:

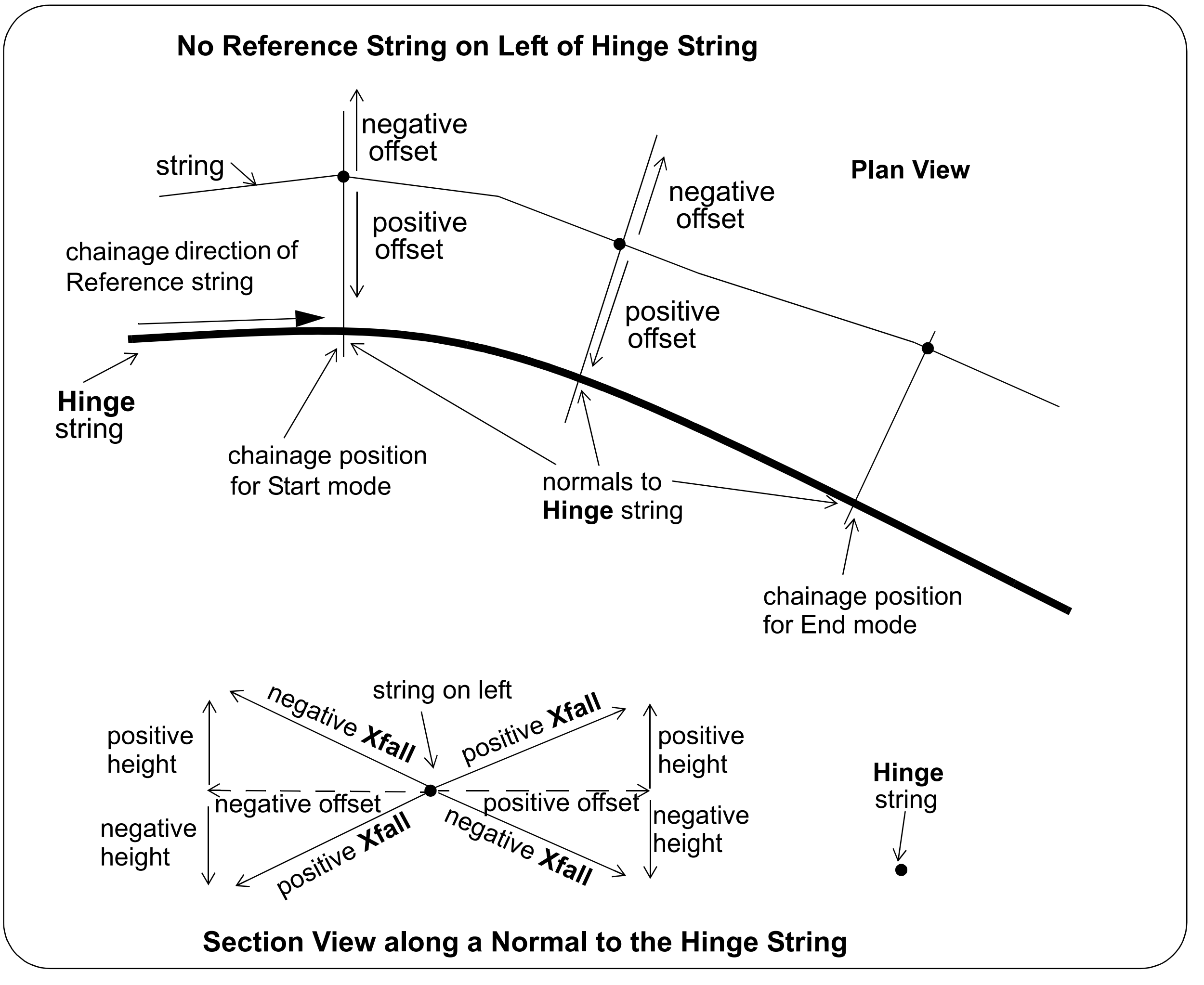
a **xfall** is **positive** when it is going up when measured in the direction of increasing positive offset with respect to the Hinge string. That is, it is going up when moving in the direction of going to the **right** of the Hinge string.

a **xfall** is **negative** when it is going down when measured in the direction of increasing positive offset with respect to the Hinge string. That is, it is going down when moving in the direction of going to the **right** of the Hinge string.

a **xfall** is **negative** when it is going up when measured in the direction of increasing negative offset (decreasing positive offset) with respect to the Hinge string. That is, it is going up when moving in the direction of going to the **left** of the Hinge string.

a **xfall** is **positive** when it is going down when measured in the direction of increasing negative offset (decreasing positive offset) with respect to the Hinge string. That is, it is going down when moving in the direction of going to the **left** of the Hinge string.



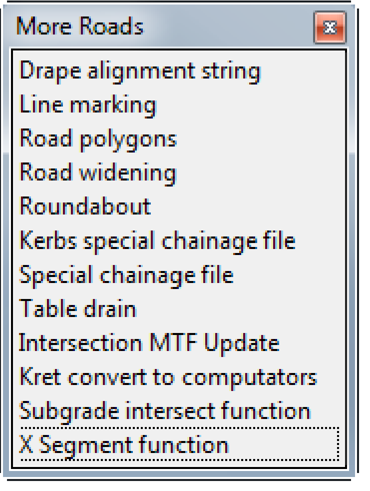

No Reference String on Left of Hinge String
negative offset
string
positive offset
negative offset
Plan View
chainage direction of Reference string
positive offset
Hinge string
chainage position for Start mode
normals to Hinge string
chainage position for End mode
string on left
negative Xfall
positive Xfall
positive height
positive height
negative offset
positive offset
negative height
negative height
positive Xfall
negative Xfall
Hinge string
Section View along a Normal to the Hinge String

20.8.12 More Roads

Position of menu: **Design =>Roads =>More**

The **more roads** menu contains miscellaneous options to drape alignments, create line marking etc.

The **more roads** walk-right menu is



Drape alignment string	16.10.3 Drape Alignment (Macro)
Line marking	24.2 Line Marking
Road polygons	20.14.5 Polygons from Sections
Road widening	20.8.12.4 Road Widening with Minimum and Maximum Crossfall
Roundabout	20.8.12.5 Create Roundabout
Kerbs special chainage file	20.8.12.6 Kerb Special Chainage File
Special chainage file	20.8.12.7 Special Chainage File
Table drain	20.8.12.8 Table Drain - Intersection of Slopes from Two Strings
Intersection MTF Update	20.8.12.9 Intersection MTF Update
Kret convert to computators	20.8.12.10 Kret Convert to Computators
Subgrade intersect function	20.8.12.11 Subgrade Intersect Function
X Segment function	20.8.12.12 X Segment function

20.8.12.1 Drape Alignment

Position of option on menu: **Design =>Roads =>More =>Drape alignment string**

This option has already been documented in **Tins => Drape=> Drape align.**

For the option *Drape align,* please continue to the section 16.10.3 Drape Alignment (Macro).

20.8.12.2 Line Marking

Position of option on menu: **Design =>Roads =>More =>Line marking**

This option creates an alignment string for a traffic island and has already been documented as

Drafting => Line marking

For more information on the option *Line marking*, please continue to the section 24.2 Line Marking.

20.8.12.3 Road Polygons

Position of option on menu: **Design =>Roads =>More =>Road polygons**

This option takes a model of cross-sections and create polygons between specified points on the cross-sections. This option is already documented under

Design => X-Sections =>Polygons from sections

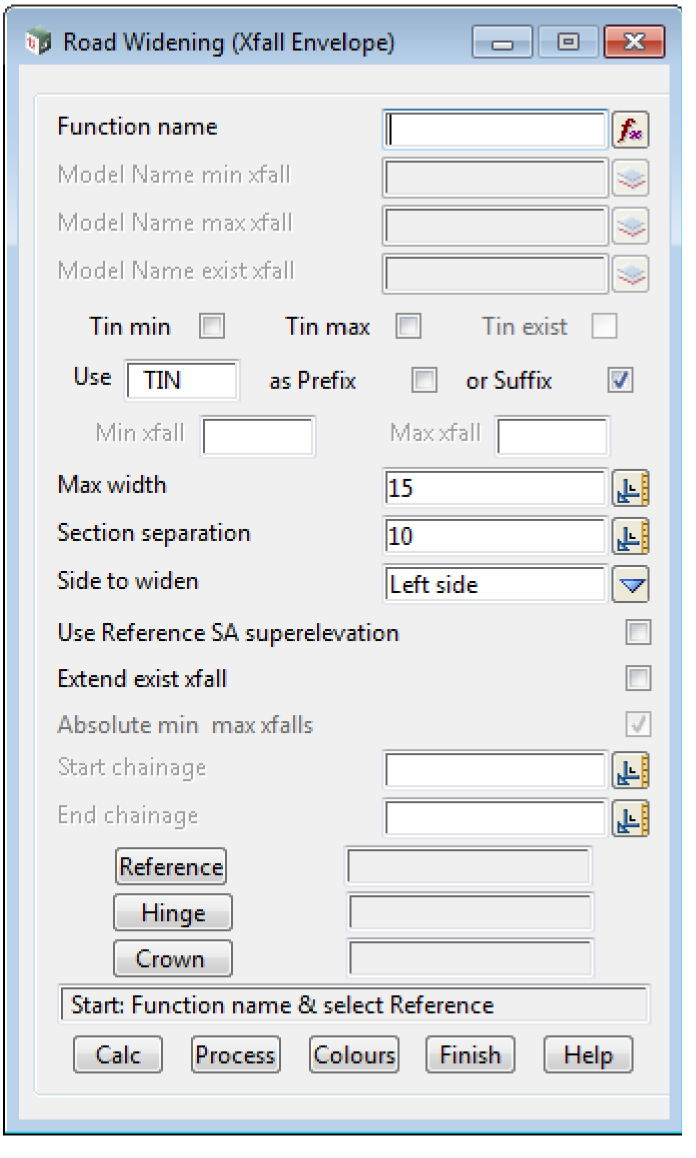
For more information on the option *Polygons from sections*, please go to the section 20.14.5 Polygons from Sections in the this chapter

20.8.12.4 Road Widening with Minimum and Maximum Crossfall

Position of option on menu: **Design =>Roads =>More =>Road widening**

This option creates cross sections using the minimum and maximum cross-fall when going out from a selected reference string (usually and alignment string).

Adding these cross sections to a section view when designing the vertical geometry gives an envelope of maximum and minimum crossfall points.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up

Model name min/max xfall model box

model for minimum/maximum cross fall cross sections.

Min xfall input box -2

minimum cross-fall (in %)

Max xfall input box **-5**

maximum cross-fall (in %)

Max width input box 10

maximum width to go out at the minimum and maximum cross falls.

Chg interval input box 20

chainage interval for calculating the minimum and maximum cross fall cross sections.

View to add view box

view to add the created cross sections to.

Reference string select

pick the reference string with direction for the x-sections to go out from. The cross sections are created to the right of the reference string as defined by the direction that the Reference string is picked. To get the sections on the other side, pick the reference string in the opposite direction.

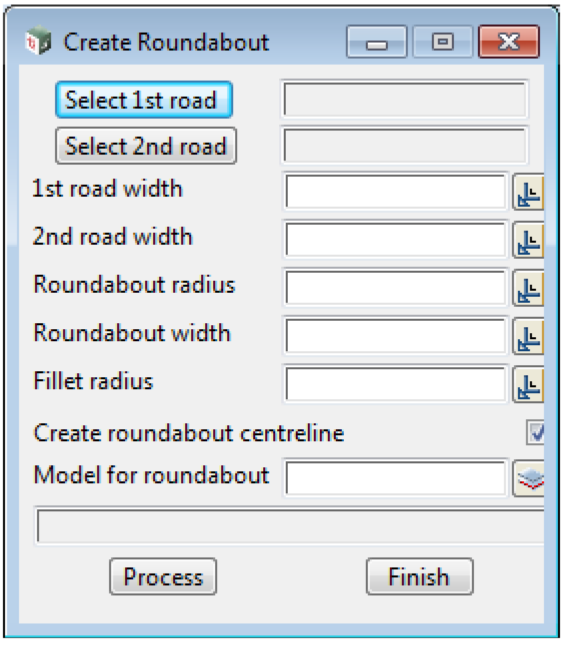
Process button

create the minimum and maximum cross fall cross sections.

20.8.12.5 Create Roundabout

Position of option on menu: **Design =>Roads =>More =>Roundabout**

This option creates a roundabout between two alignment strings. The road widths for the two roads (the distance from the road centre line to the edge of the road) plus the roundabout radius (the distance from the centre of the roundabout to the centreline of the roundabout road), the roundabout width (added and subtracted from the roundabout width to give the inner and outer edges of the roundabout roadway - the inner and outer roundabout radii) and the fillet radius for the turning lanes.



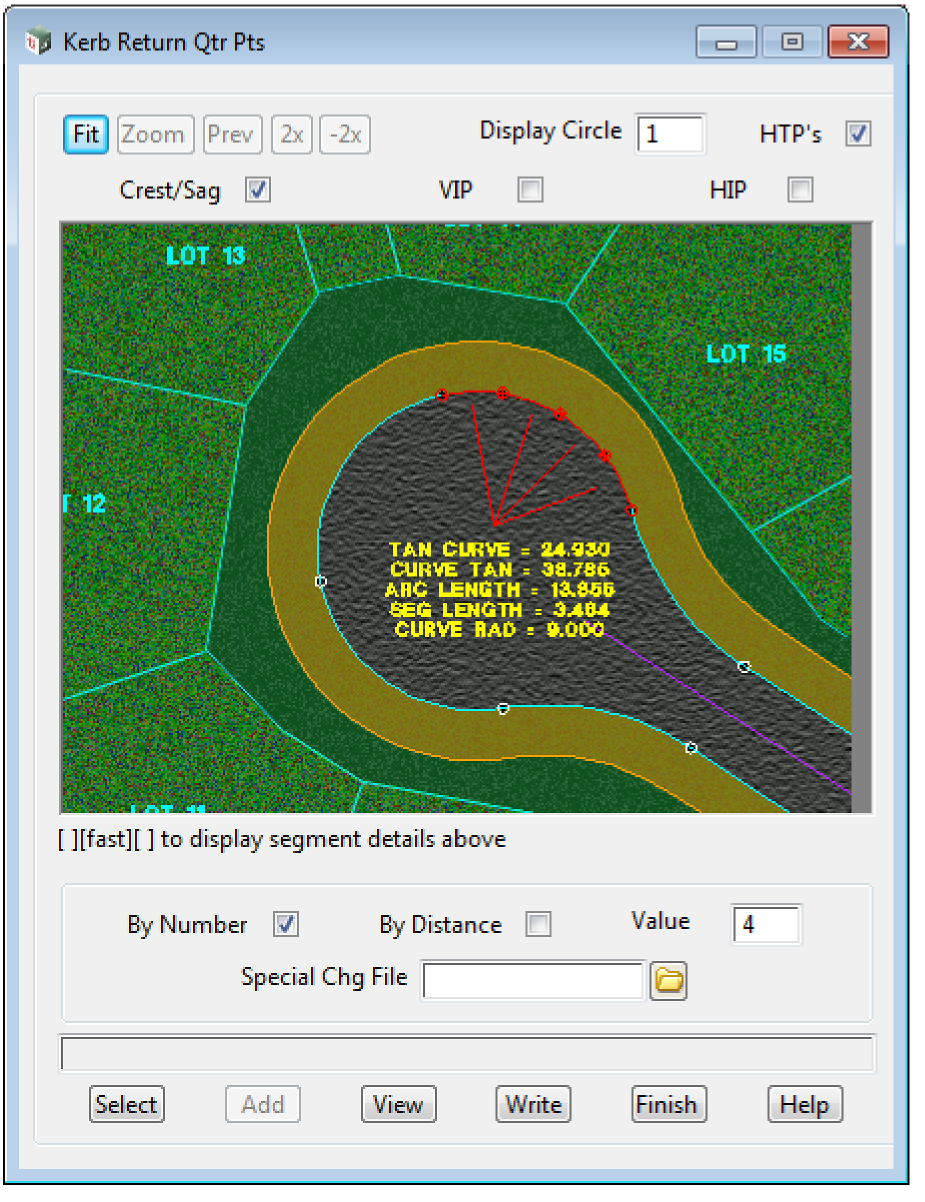
The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Select 1st road	string select		
first alignment string for the roundabout.			
Select 2nd road	string select		
second alignment string for the roundabout.			
1st road width	input box		
distance from the first alignment string to the edge of the road.			
2nd road width	input box		
distance from the second alignment string to the edge of the road.			
Roundabout radius	input box		
distance from the centre of the roundabout to the centreline of the roundabout road.			
Roundabout width	input box		
distance to add and subtract from the roundabout width to give the inner and outer edges of the roundabout roadway (the roundabout radii).			
Fillet radius	input box		
fillet radius of the turning lanes.			
Create roundabout centreline	tick box		
if ticked, an alignment string is created for the centreline of the roundabout.			
Model for roundabout	input box		
model for the created roundabout.			
Process	button		
run the option.			

20.8.12.6 Kerb Special Chainage File

Position of option on menu: **Design =>Roads =>More =>Kerb special chainage file**

This option is used to create a special chainage file for an alignment string. The method of creating the chainages is defined differently for each segment of the alignment string.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
By number	tick box	tick	

if ticked, break the selected segment of the selected string into the number of pieces given by the **Value** *field.*

By distance	tick box		

if ticked, break the selected segment of the selected string into pieces of length given by the **Value** *field.*

Value	tick box	tick	

the number of pieces or the chainage length to break the selected segment into.

Special chg file	file box		

file to write chainages to

Select	button		

pick the alignment string to create chainages for. The alignment string is then drawn in the panel draw box. Each segment of the alignment string is selected by clicking MB over it in the panel draw box. The special chainages are then defined for that segment.

Add button

add the chainages given by number of length to the special chainages list

View button

view the special chainages list. The special chainages list can be edited

Write button

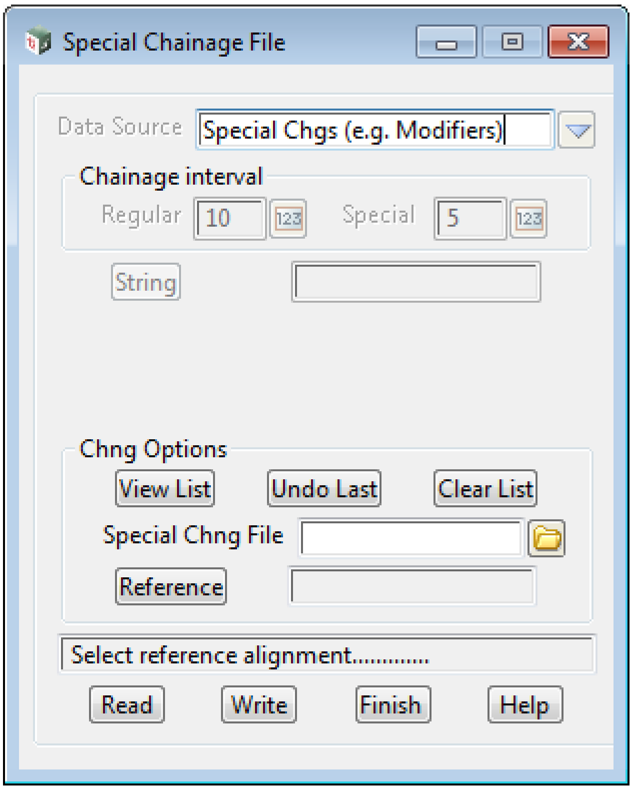
write the special chainages list to the special chainages file

20.8.12.7 Special Chainage File

Position of option on menu: **Design =>Roads =>More =>Special chainage file**

This option is used to create a special chainage file for an alignment string.

The chainages are created for a selected reference string but can then be defined by dropping strings and/or individual points onto the reference string.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Data source	choice box	Special chainages	Special chainages String Points
Regular			

Special

String string select

Reference string select

select the string to create special chainages for.

View list button

view the special chainage list.

Undo last button

undo the last set of created chainages.

Clear list button

clear the chainages list.

Special chng file file box

file to read chainages from or write chainages to

read button

read in a file of special chainages.

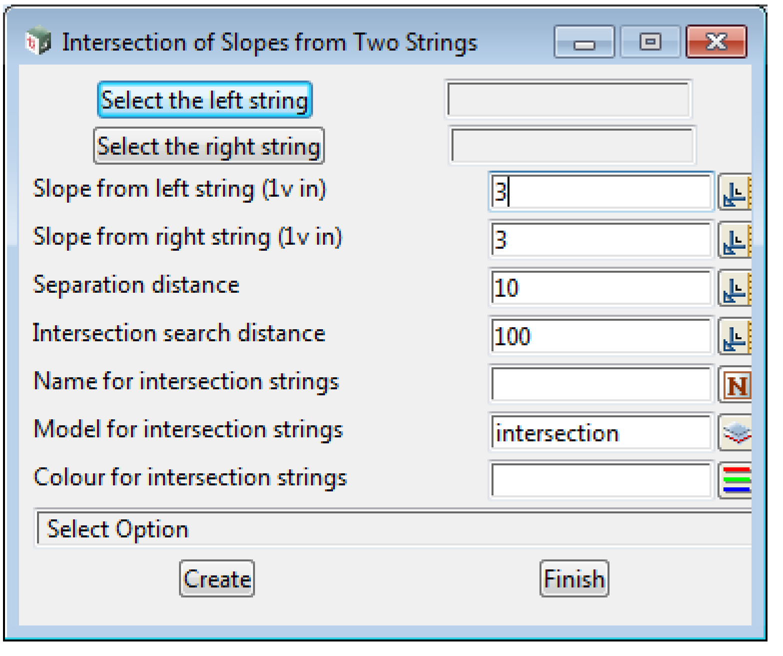
Write button

write the chainages list to the special chainages file.

20.8.12.8 Table Drain - Intersection of Slopes from Two Strings

Position of option on menu: **Design =>Roads =>More =>Table drain**

This option is used to create a string which is the intersection of slopes from a left and right string.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Select the left string string select

select the left hand side string

Select the right string string select

select the right hand side string

Cross fall (1 in...) from left/right string input box 3

batter slope to go from left/right string

Separation distance input box 10

distance to create batter lines and find intersection

Intersection search distance input box 100

distance to search to find an intersection of the batter slopes

Name for intersection strings input box

name for the strings created by the intersection of the batters

Model/Colour for intersection strings intersection available models/colours

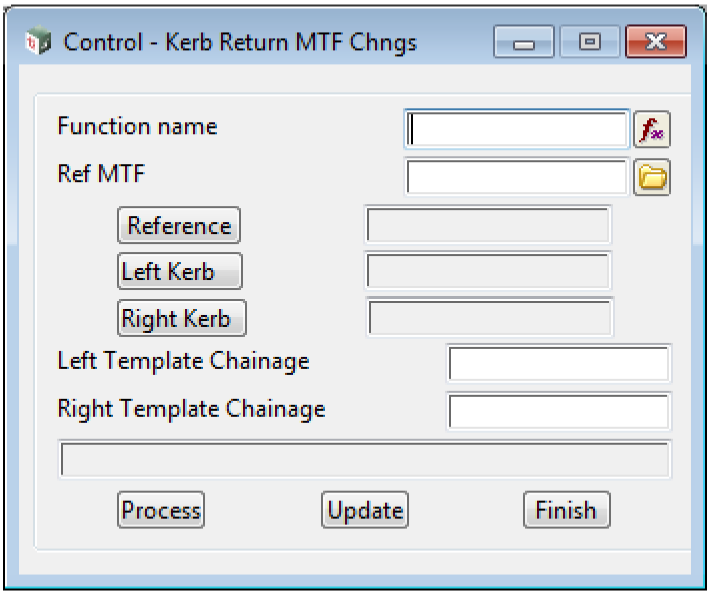
model/colour for the intersection strings

Create button

Create the intersection strings by battering of the left and right strings

20.8.12.9 Intersection MTF Update

Position of option on menu: **Design =>Roads =>More =>Intersection MTF updates**



The fields and buttons used in this panel have the following functions.

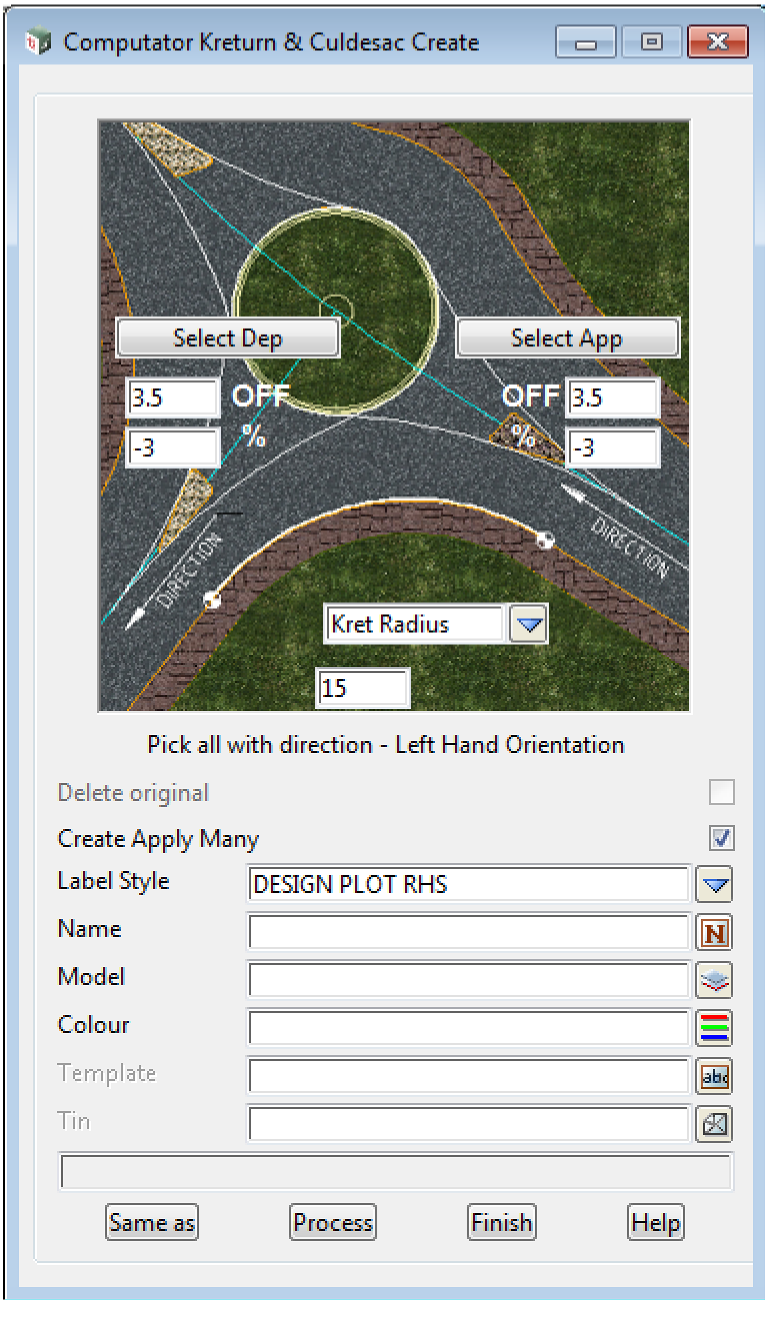
Field Description	Type	Defaults	Pop-Up
Function name	function box		available functions
name for the function			
Ref MTF	file box		*.mtf files
Reference	string select		
select the reference string			
Left kerb	string select		
select the left kerb string			
Right kerb	string select		
select the right kerb string			
Left template chainage	input		
Right template chainage	input		
Process	button		
Update	button		

20.8.12.10 Kret Convert to Computators

Position of option on menu: **Design =>Roads =>More =>Kret convert to computators**

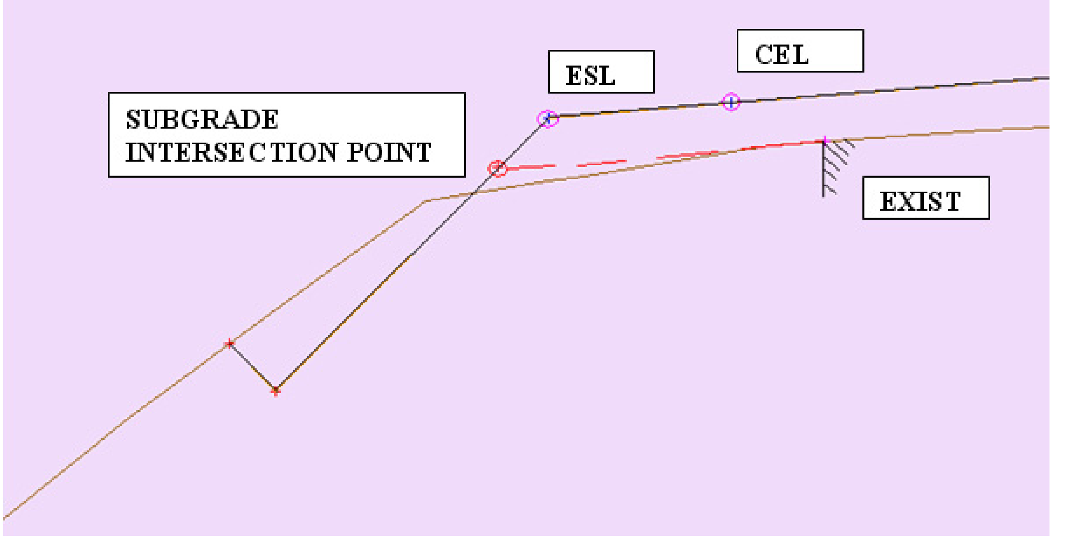
This section of documentation is a work in progress and will be updated in subsequent releases.

Selecting **Kret convert to computators**, displays the **Computator Kreturn & Culdesac Create** panel on the screen



20.8.12.11 Subgrade Intersect Function

Position of option on menu: **Design =>Roads =>More =>Subgrade Intersect Function**



Typical Application

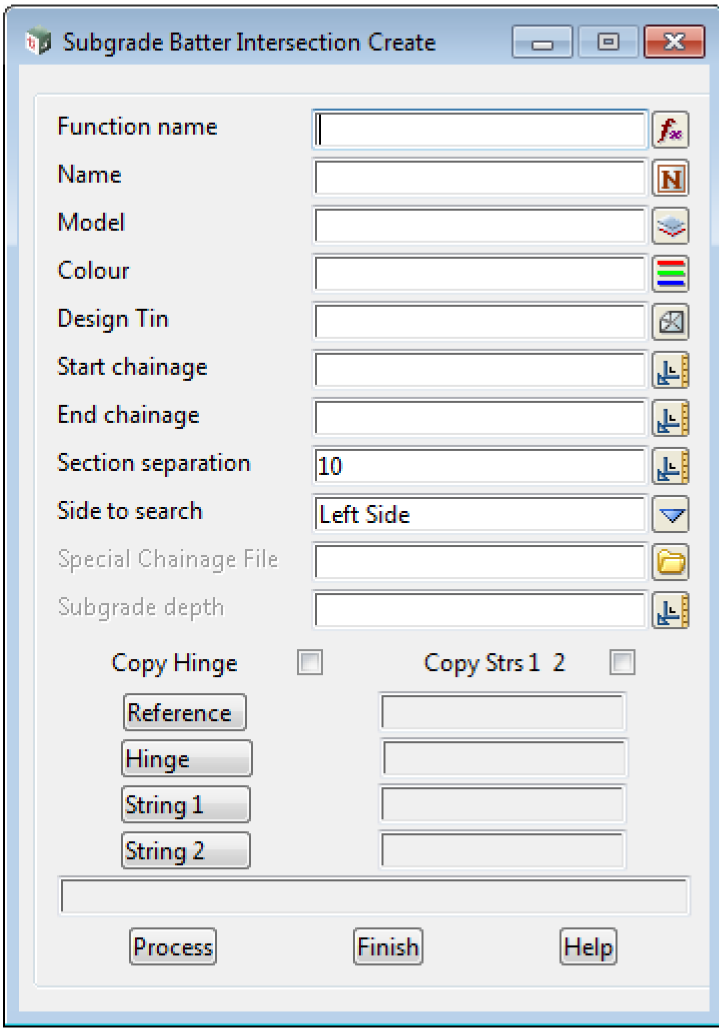
1. Widening job where the intersection of the **design xfall** from the **existing road edge** and the **design batter** is required.
2. Design strings are **ESL** (shoulder edge) and **CEL** (carriageway edge)

This panel is used to create a string that could be used in the subgrade surface of a road design, in particular road widening.

The crossfall between these strings is calculated, and applied from the hinge string, offset by the subgrade depth, to intersect with the design tin specified.

The intersecting string created is placed on the selected model.

Selecting **Subgrade Intersect Function**, displays the **Subgrade Batter Intersection Create** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function name	function box		select function

name for use in recalc or chains

Name	name box		

Name for string created

Model	model box		

model name for string created

Colour	input		available colours

Colour for string created

Design Tin	input		select tin

Design surface above

Start Chainage	measures box		At Point, Point to Point, String from Point, String to Point

Enter start chainage

End Chainage	measures box		At Point, Point to Point, String from Point, String to Point)

Enter end chainage

Section separation	measures box	10	At Point, Point to Point, String from Point, String to Point)

Distance along the reference to create points on the string created

Side to search	choice box	Left side	Left or Right

Side to search in relation to reference string

Special chainages	input		*.spf files

a file containing chainages, one per line, that are also used as chainages to create points on the string created

Subgrade Depth	measures box		At Point, Point to Point, String from Point, String to Point)

Depth below the design tin (in metres)

Copy Hinge tick

If ticked, makes a copy of the hinge in the model specified.

Copy Strs 1 2 tick

If ticked, makes a copy at subgrade depth of strings 1 & 2 in the model specified.

Reference string select

Selection must be Super alignment, used to calculate chainages

Hinge string string select

selected as a start point for the subgrade (Existing bitumen or cutback strings e.g.)

String 1 string select

string in the design surface, used in xfall calculations

String 2 string select

string in the design surface, used in xfall calculations

Process button

runs the option

20.8.12.12 X Segment function

Position of option on menu: **Design =>Roads =>More =>X Segment function**

This option is used for the removal of segments on design and boxing cross sections.

Typically it is used where road widening has been applied to both sides of a roadway.

Design sections in this case would be created using "Cuts through strings", where the two widenings are now joined.

The design may also have a small overlay on the existing roadway and boxing definitions have been defined for each side of the widening.

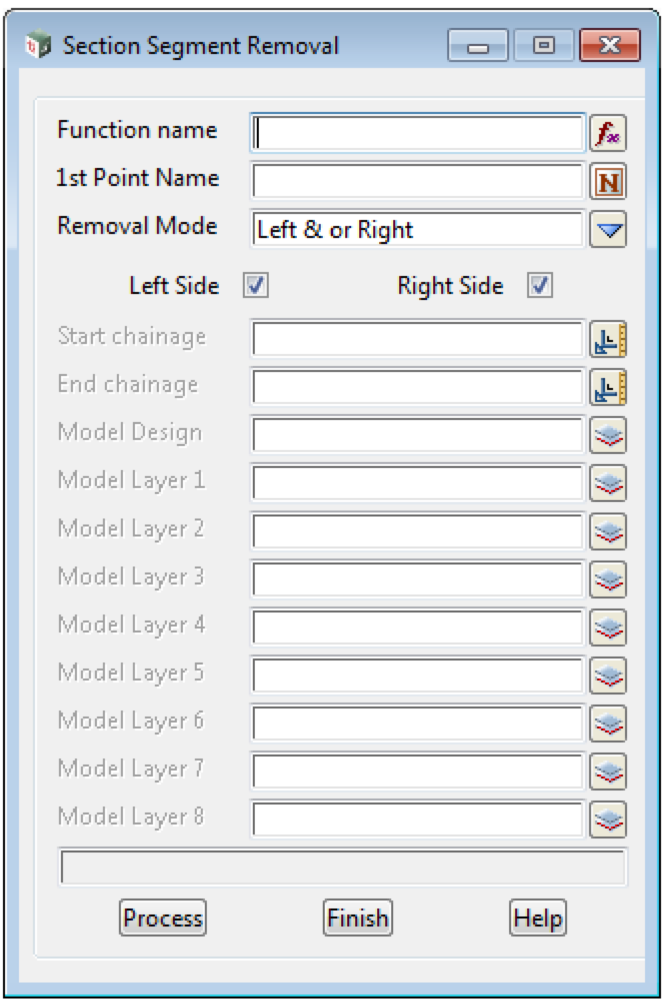
In each case again, the sections, especially the boxing subgrade, are joined.

The sections are converted to super strings and segments that need to be removed are defined at vertex points by name.

A point name is entered as the "First Point" and segments are defined as either left or right of that point or a second point along the section can be specified.

A "Design" section model can be entered along with multiple boxing models...Layer1, Layer2, Layer3 etc.

Selecting **X Segment function**, displays the **Section Segment Removal** panel on the screen



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function Name	input		

name for use in recalc or chains

1st point name input

Point name on design section

Removal Mode choice box Left & or Right Left & or Right, 2nd Point

Removal type left or right of first point or to a second point

Left Side tick box

Right Side tick box

Start Chainage real

if blank, then all the sections are used

End Chainage real

if blank, then all the sections are used

Model Design input

existing model name of design sections

Model Layer (n) input

existing model name of boxing sections

Process button

runs the option

20.9 Components

Position of menu: **Design =>Roads =>Components**

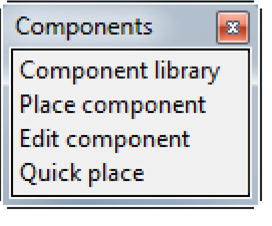
Components are parametric road objects that can be placed on existing design strings to generate complex geometry. This includes geometry for:

- Intersections
- Roundabouts
- Exit Ramps
- Entry Ramps
- Bus Bays
- Parking Bays
- CHR Intersections
- Culdesacs
- Hammerheads

Each component has a number of parameters that may be modified before and after they are placed to meet your requirements.

Components are stored in a library. Every time you place a component, you will reference the library definition. Once it is placed, if you wish to customise only that placed instance of the component, you may detach it from the library so that changes to the component will not affect the library component and vice versa.

The **Components** walk-right menu is



For *the options see*

Component library	20.9.1 Component Library
Place component	20.9.32 Place a Component
Edit component	20.9.34 Edit a Placed Component
Quick place	20.9.35 Component Quick Place

For information on

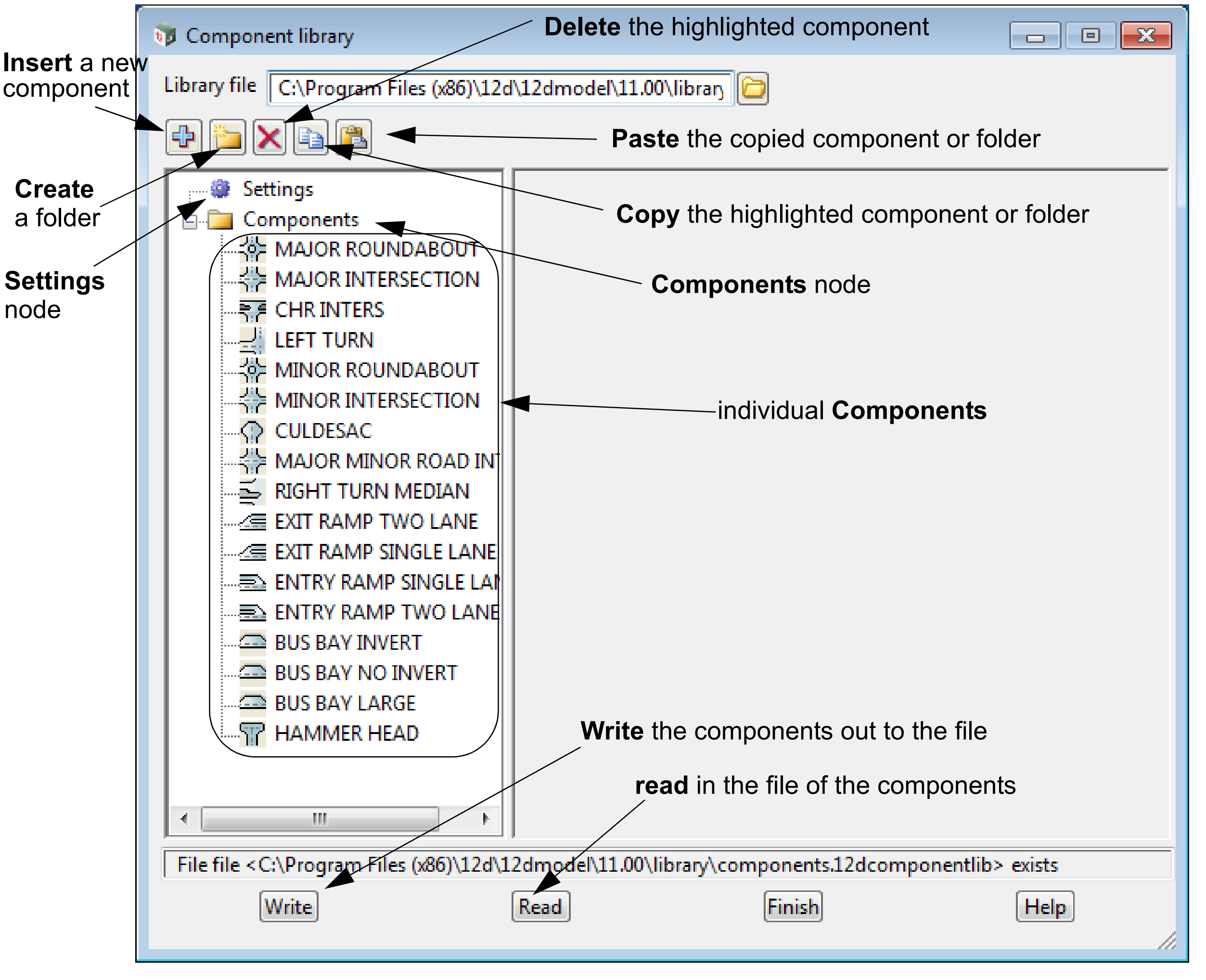
Editing Components	20.9.32.1 Editing Components
Specific Component Types	20.9.2 Component Types

20.9.1 Component Library

Position of option on menu: Design =>Roads =>Components => Component library

This panel is used to define the components in your current project. Components are initially based off predefined 'base' components (see Base component). Once you add a base or template component to your library, you may customise it as you require.

Selecting **Component library** brings up the **Component library** panel.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Library file	file box		*.12dcompnonentslib files

the current component library you are editing

 icon

Inserts a new component (See 20.9.1.4 New Component)

Folder icon

creates a new folder or subfolder under the **Components** *node.*

This is to allowing grouping of similar components.

Delete icon

deletes the highlighted component or folder.

Copy icon

copies the highlighted component or folder.

Paste icon

pastes the selected component or folder at the current highlighted level.

Write button

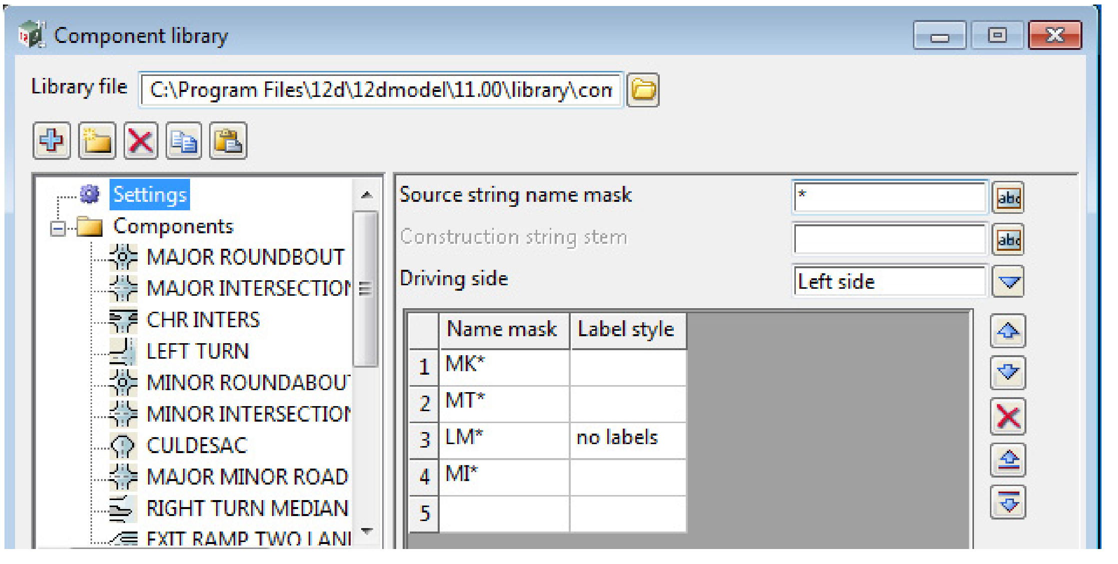
writes the components out to the given component library file.

Read button

reads in the given component library file.

20.9.1.1 Settings Node

This node defines library wide settings. Whenever you use a component from this library, these settings will apply.



20.9.1.1.1 Naming of Strings in a Component

Each string generated as part of the component can be named. They can either be set to something static or they can be dynamically generated using a number of variables as part of the name field.

The variables generally concern which side of the road the string is being created for, whether it was the approach or departure or the '**source**' or '**destination**'.

For example, in a left turn, the '**source**' is the string a driver is leaving from and the '**destination**' is the string entering.

$source_name - the name of the source string after the **source string name mask** is applied (see 20.9.1.1.2 Source String Name Mask)

$destination_name - the name of the destination string after the **source string name mask** is applied (see 20.9.1.1.2 Source String Name Mask)

$full_source_name - the full name of the source string

$full_destination_name - the full name of the destination string

$source_approach_or_departure - '**approach**' if we are generating strings for the approach of the source string, otherwise '**departure**'

$destination_approach_or_departure - '**approach**' if we are generating strings for the approach of the destination string, otherwise '**departure**'

$source_side - the side of the road we are currently generating strings for on the source string

$destination_side - the side of the road we are currently generating strings for on the destination string

$count[1-9] - allows up to 9 counters, of the form **%count1, $count2** ... **$count9**

Each time this counter variable is seen, it's value is incremented by 1.

20.9.1.1.2 Source String Name Mask

The **source string name mask** is defined in the **Component Library**. It allows you to extract part of a centreline string name. This is quite useful to maintain your own naming conventions.

For example, if you have a centreline named ***MC01*** and you wish to name your left turn ***MK01***, the part of the name that is common is ***01***.

If you set the source string name mask to **MC***, the name used when you use the **$source_name** or **$destination_name** variables will be everything represented by the '*'. In this case, **01**.

The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Source string name mask			

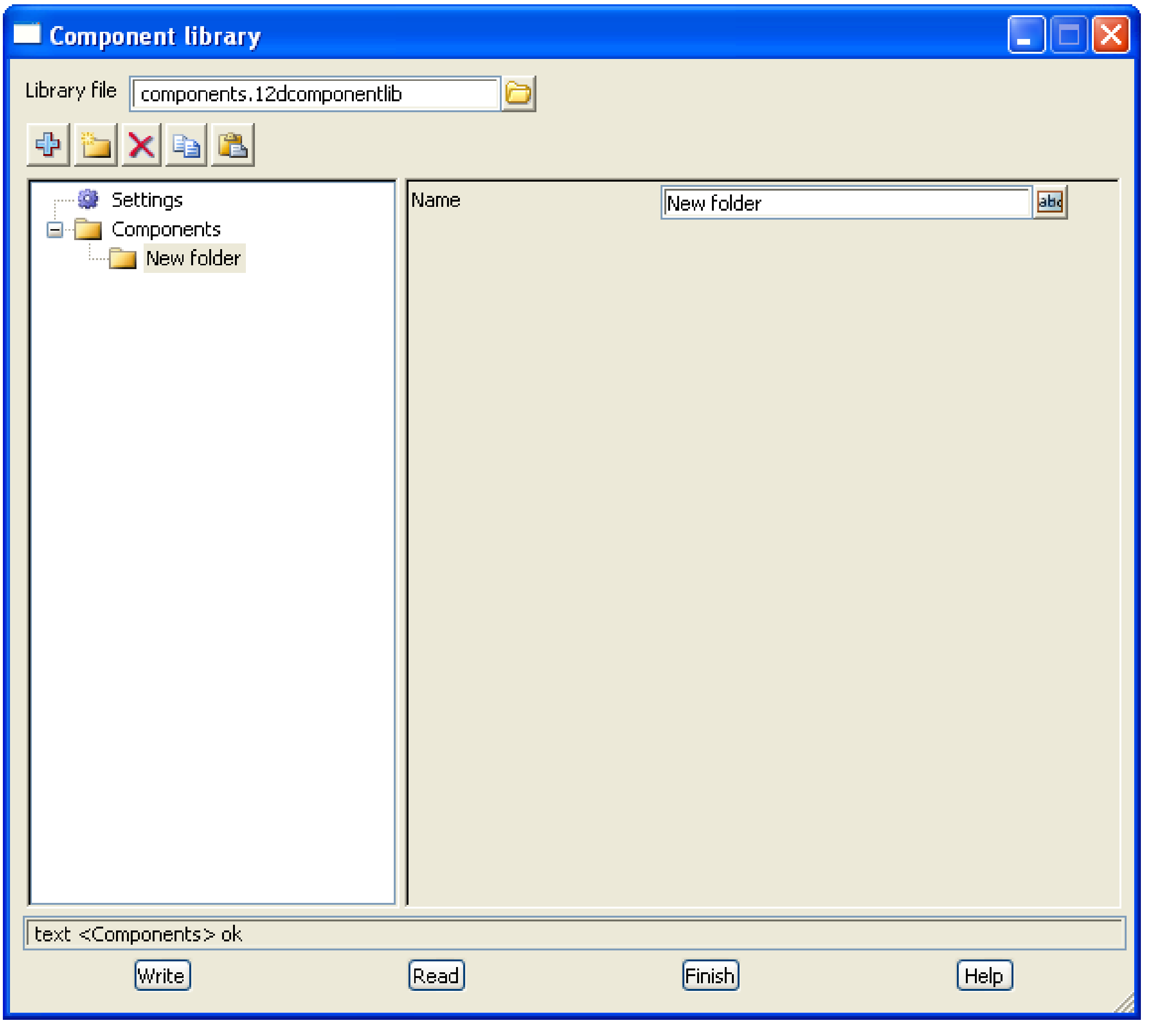
an optional field used to assist in naming individual strings created as part of a component. See 20.9.1.1.2 Source String Name Mask *for more information.*

Driving side	choice box	Left side	Left side, Right side

whether or not the component should be created for left or right hand side driving.

20.9.1.2 Folder Nodes

Folders can be created to group components. You can create a folder and give it whatever name you wish.



The fields and buttons used in this panel have the following functions:

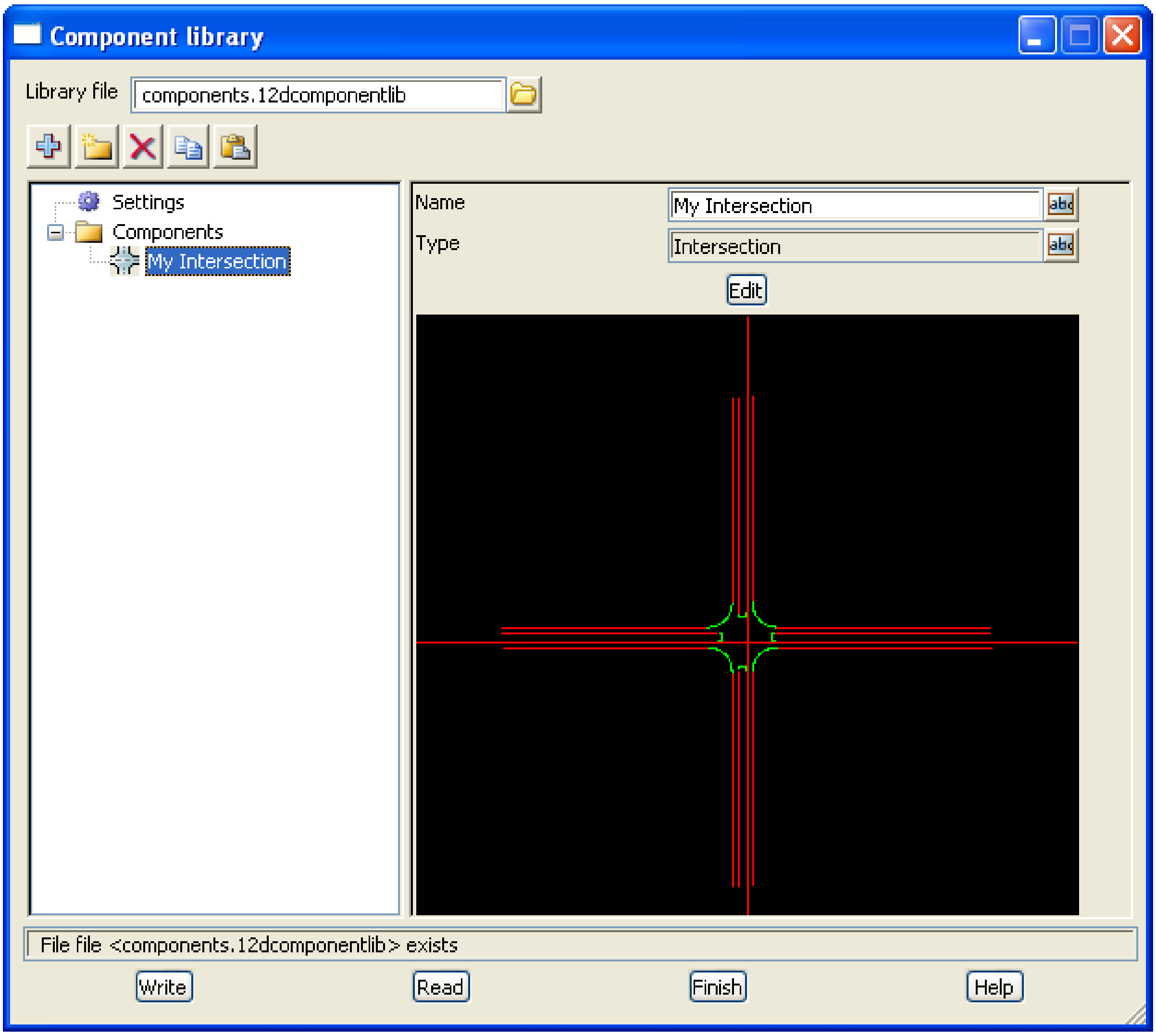
Field Description Type Defaults Pop-Up

Name

the name of the folder

20.9.1.3 Components Node

Each component node displays the name and type of the component, as well as what it will look like on a theoretical set of centreline strings. The generated geometry will change when you place it depending on the selected strings but this view gives you an idea of what it may look like.



The fields and buttons used in this panel have the following functions:

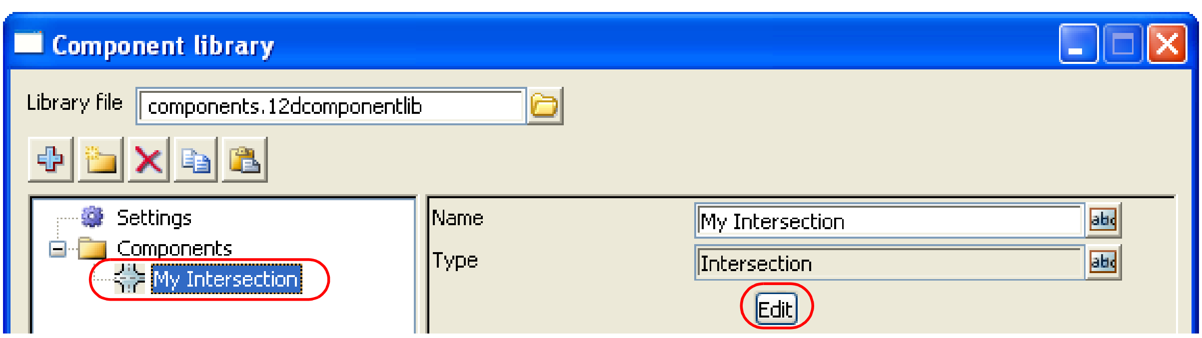
Field Description Type Defaults Pop-Up

Name

the name of the component

Type

the type of the component - read only



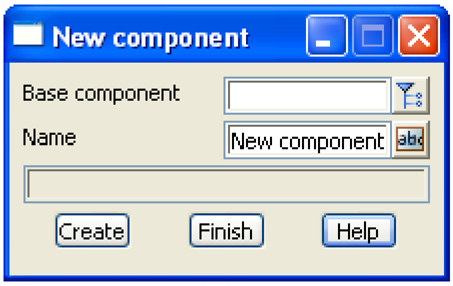
Edit button

edits the component. For more information please see [20.9.32.1.1 Editing in the Component library - Component Editor]

20.9.1.4 New Component

This panel allows you to create a new component from a base component. A base component may be from either one of your own library components, or a component shipped with ***12d Model***.

Selecting  brings up the **New component** panel.

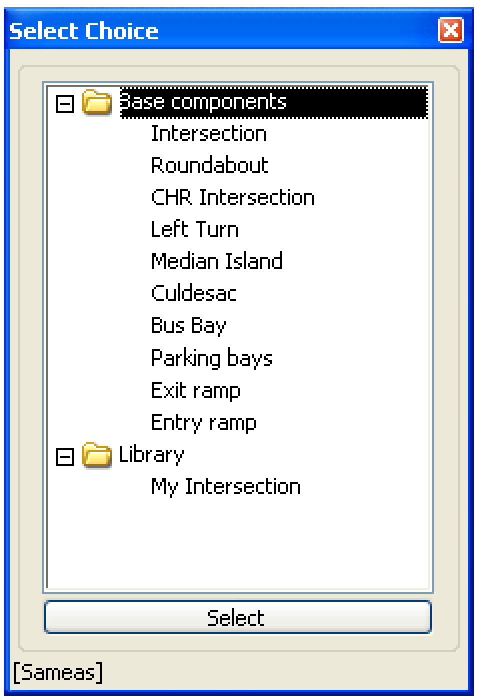


The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Base component

the component you wish your component to be based on



*This may be either a **Base Component**, as shipped by 12d, or one from your **Library**.*

*For information on **Component Types** please go to the section* [20.9.2 Component Types]

Name

the name of your component, which must be unique

Create button

creates the component and adds it to the library

20.9.2 Component Types

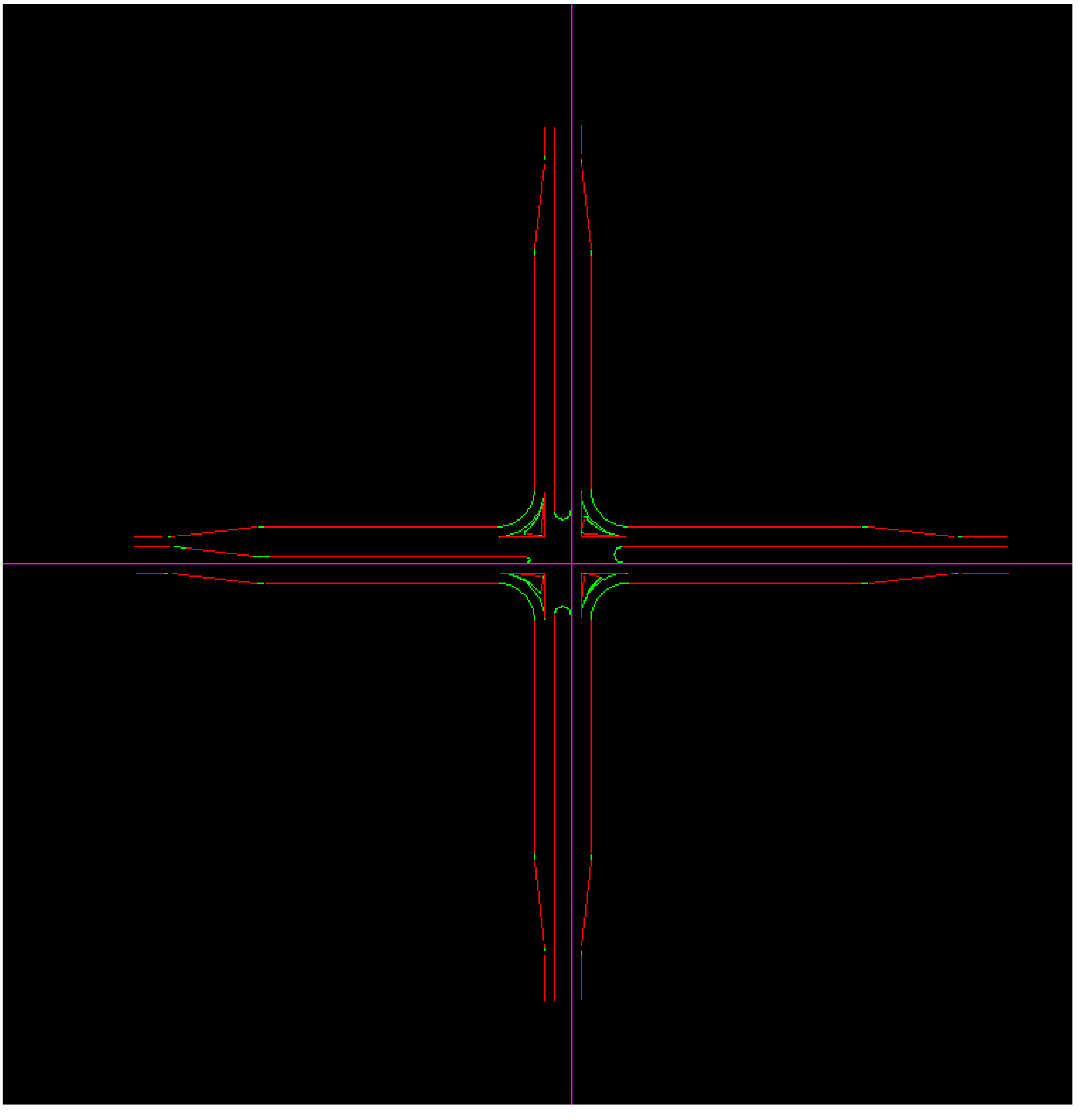
For

Minor Intersection	20.9.2.1 Minor Intersection
CHR Inters	20.9.27.2 CHR Intersection
Left Turn	20.9.13.2 Left Turn
Minor Roundabout	20.9.7.2 Roundabout
Median Island	20.9.15.2 Median Island
Culdesac	20.9.11.9 Culdesac
Bus Bay	20.9.11.8 Bus Bay
Parking bays	20.9.11.11 Parking Bays
Exit ramp	20.9.22.3 Exit Ramp
Entry ramp	20.9.17.2 Entry Ramp

20.9.2.1 Minor Intersection

An intersection component can be edited via the 20.9.1 Component Library or by editing the definition of a placed component. While an intersection is defined as a crossroad, the intersection will automatically close off any leg of the intersection that does not have valid strings.

For information on *Defining an Intersection* see 20.9.2.1.1 Defining an Intersection
Placing Intersections 20.9.7.1.4 Placing Intersections



20.9.2.1.1 Defining an Intersection

An intersection is broken into two roads - the **Major road** and the **Minor road**.

Each road, major or minor, may have an approach (the section of the road leading in to the intersection) and the departure (the section of the road leading away from the intersection).

The approach and departure for both the Major and Minor Road are defined by the same set of parameters.

For information on:
Placement Parameters Node see 20.9.3 Placement Parameters
Carriageway Node 20.9.4 Carriageway Node
Approach / Departure => Left turn Node 20.9.5 Left Turn - Approach / Departure
Approach / Departure => Left Turn =>Kerb Return Node 20.9.6 Approach / Departure =>Left

Turn =>Kerb Return Node

Approach / Departure => Median / Right Turn Node 20.9.7 Approach / Departure =>Median / Right Turn Node

20.9.3 Placement Parameters

This node is only available when editing a placed component. For more information on the fields displayed here, see the section on 20.9.7.1.4 Placing Intersections.

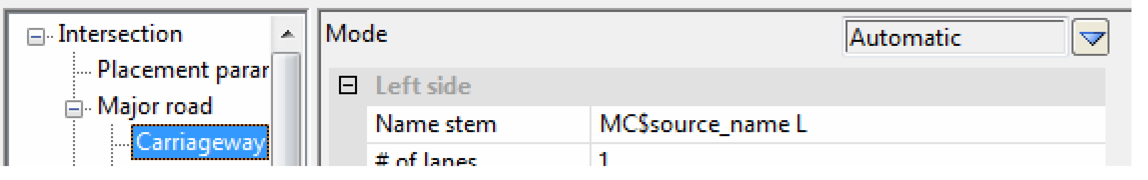
20.9.4 Carriageway Node

This node defines the carriageway of the road, in terms of the number of lanes and the type of lanes.

There are two modes for defining the carriageway: Automatic Mode and Manual Mode.

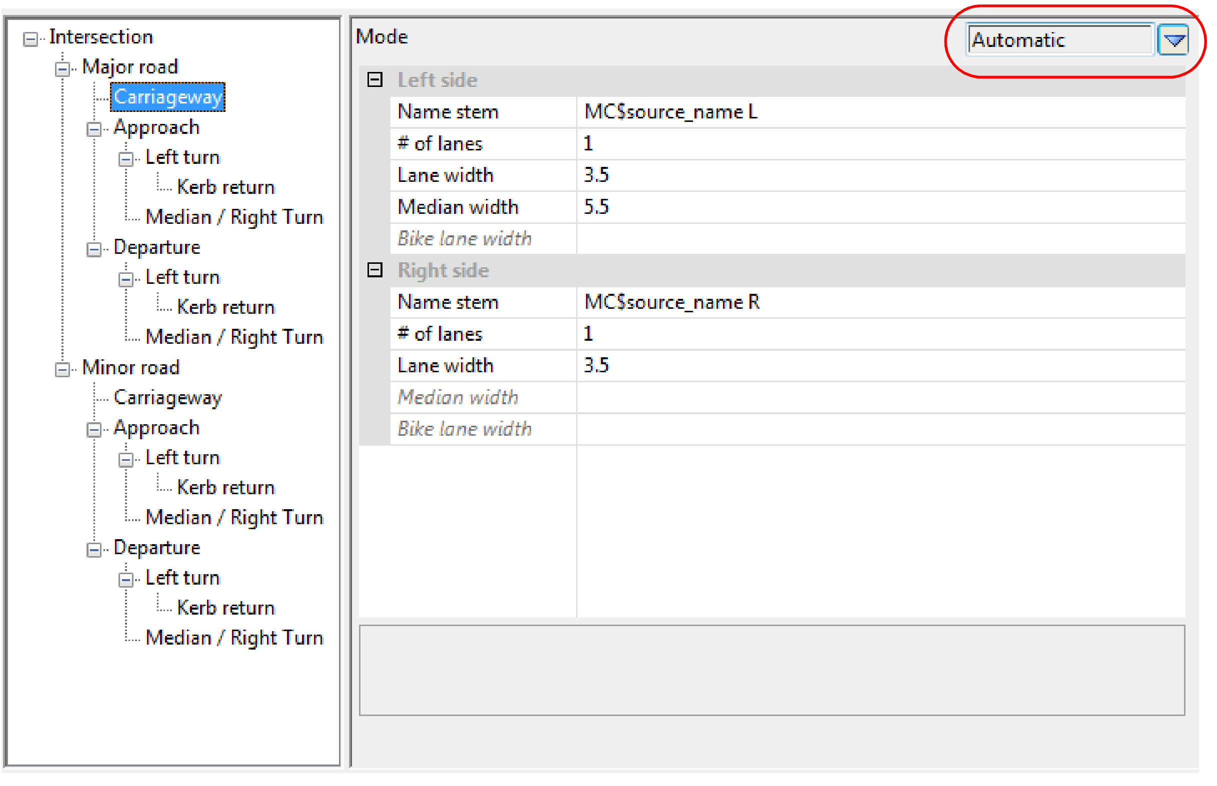
Note that regardless of the mode, you can only define a median on one side of the carriageway.

If you place your intersection using dual carriageways, the width will be ignored.



Automatic Mode

Automatic mode does not allow you to define names for each lane of your carriageway and only allows a basic layout. It is the simplest and quickest way to define a carriageway.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Left Side

Name Stem

the name to apply to each created lane on the left side of the carriageway

# of lanes

the number of lanes to create on the left side of the carriageway

Lane width

the lane width for each lane on the left side of the carriageway

Median width

the width for an optional median on the left side of the carriageway.

Bike lane width

the width for an optional bike lane on the left side of the median.

Right Side

Name Stem

the name to apply to each created lane on the right side of the carriageway

# of lanes

the number of lanes to create on the right side of the carriageway

Lane width

the lane width for each lane on the right side of the carriageway

Median width

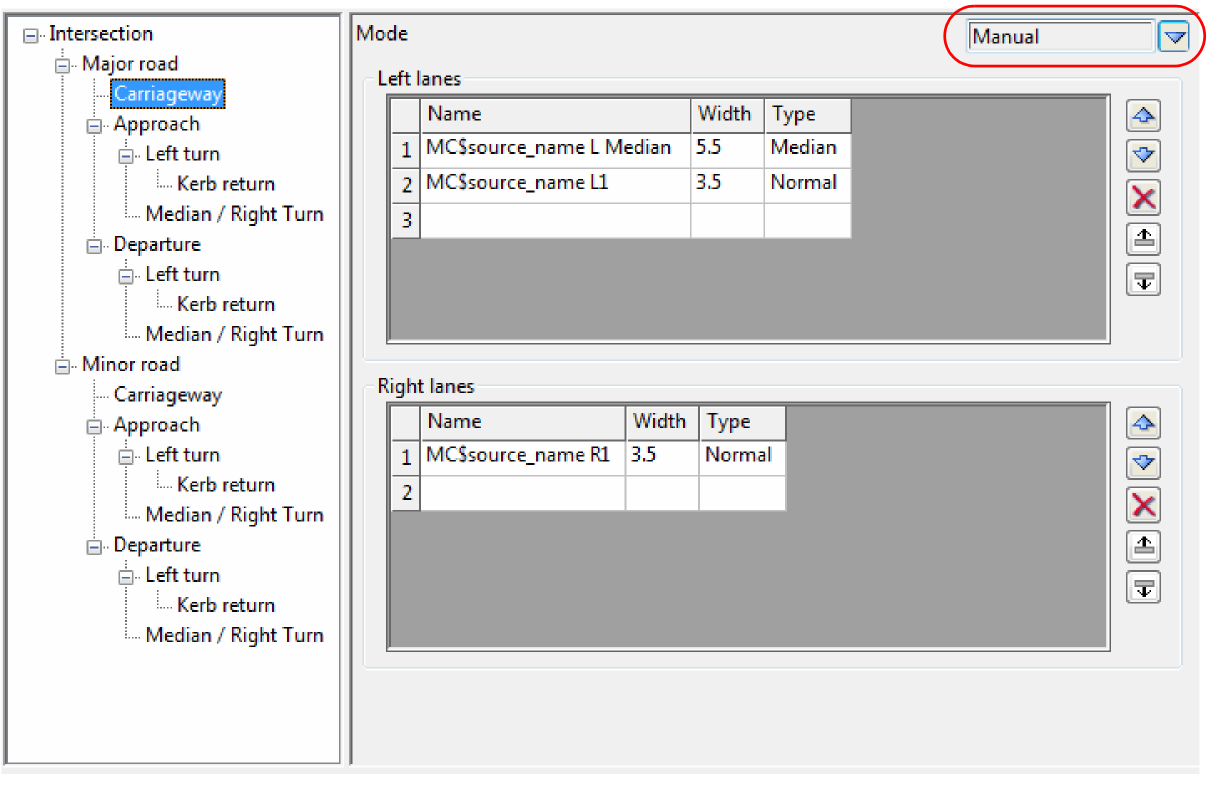
the width for an optional median on the right side of the carriageway.

Bike lane width

the width for an optional bike lane on the right side of the median

Manual Mode

Manual mode allows you to define the carriageway in more detail, by specifying each lane individually, along with its width and type.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Left Lanes Grid

Name

the name of the lane on the left side

Width

the width of the lane

Type

the type of lane (Normal, Bike or Median)

Right Lanes Grid

Name

the name of the lane on the right side

Width

the width of the lane

Type

the type of lane (Normal, Bike or Median)

20.9.5 Left Turn - Approach / Departure

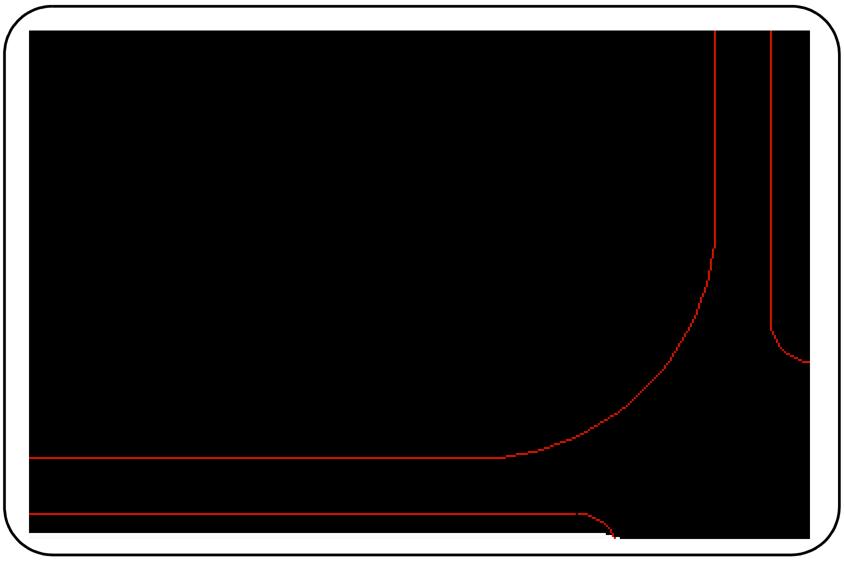
This node defines the **Left turn** of an intersection. If you are in Right Side driving mode, this will be called **Right Turn**.

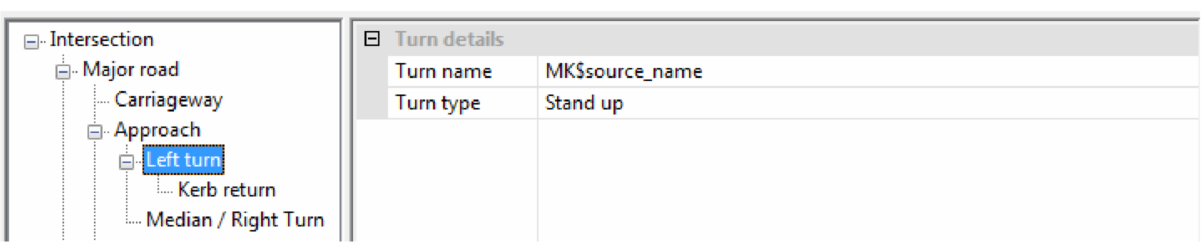
There are three types of turns currently available:

1. Stand up - for more information go to Stand Up Turn
2. Free - for more information go to Free Turn
3. High entry - for more information go to High Entry Turn

Both **Stand up** and **Free** turns allow you to define the type of kerb return being used.

Stand Up Turn





The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Turn name

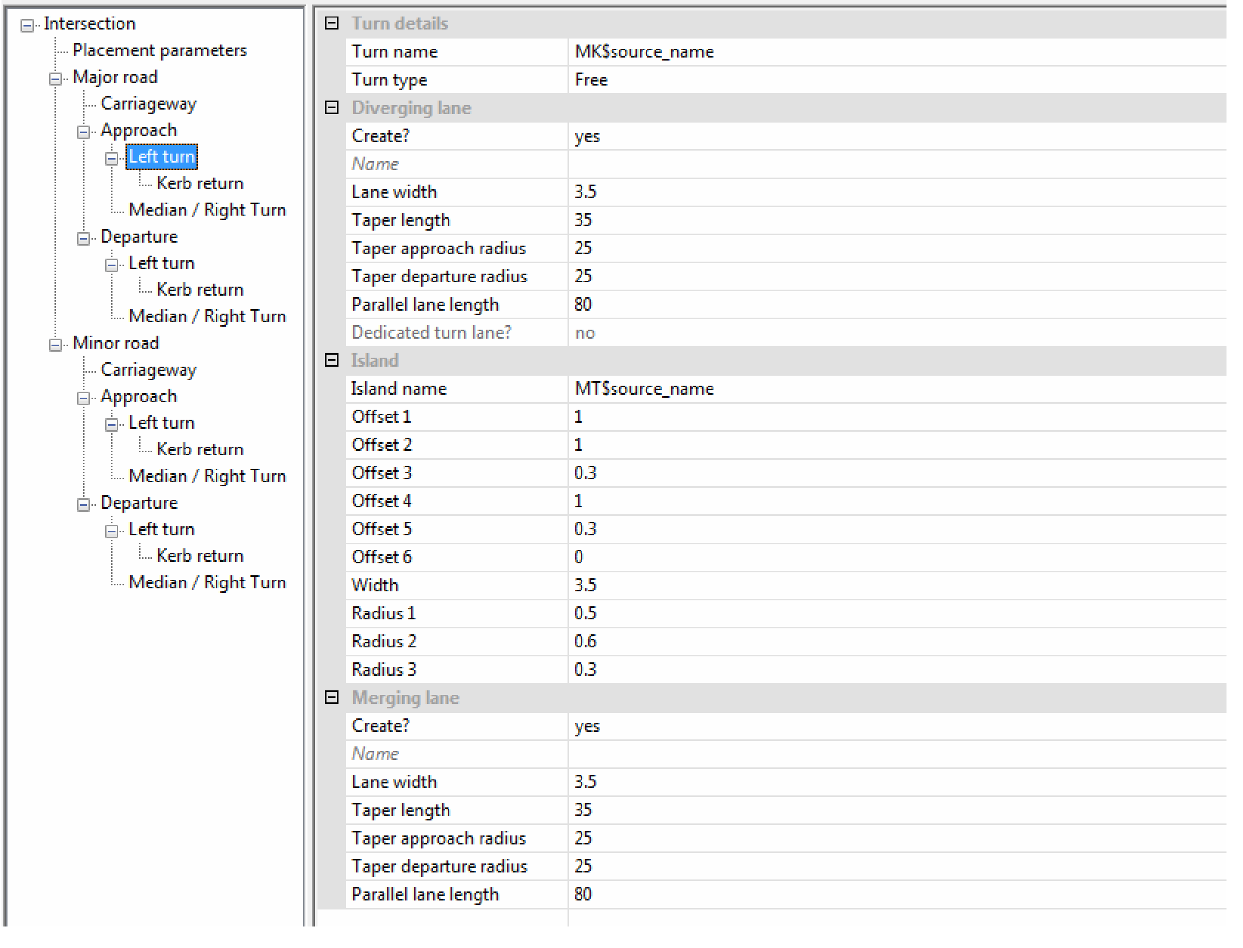
the name to use for created geometry

Turn type

the type of turn (in this case, Stand up)

Free Turn

A **Free turn** supports an optional diverging lane, a traffic island and an optional merging island.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Turn Details

Turn name

the name to use for created geometry

Turn type

the type of turn (in this case, Free)

Diverging Lane

Create?

whether or not to create a diverging lane

Name

an optional name to use for the diverging lane

Lane width

the width of the diverging lane

Taper length

the length of the leading taper on the diverging lane

Taper approach radius

the approach radius of the taper

Taper Departure radius

the departure radius of the taper

Parallel lane length

the length of the parallel lane of the diverging lane

Dedicated turn lane?

whether or not to create a dedicated turn lane - only available if not creating a diverging lane

Island

Island name

the name of the island string

Offset 1

the first offset of the painted island from the internal island

Offset 2

the second offset of the painted island from the internal island

Offset 3

the third offset of the painted island from the internal island

Offset 4

the fourth offset of the painted island from the internal island

Offset 5

the fifth offset of the painted island from the internal island

Offset 6

the sixth offset of the painted island from the internal island

Width

the width of the island from the kerb return

Radius 1

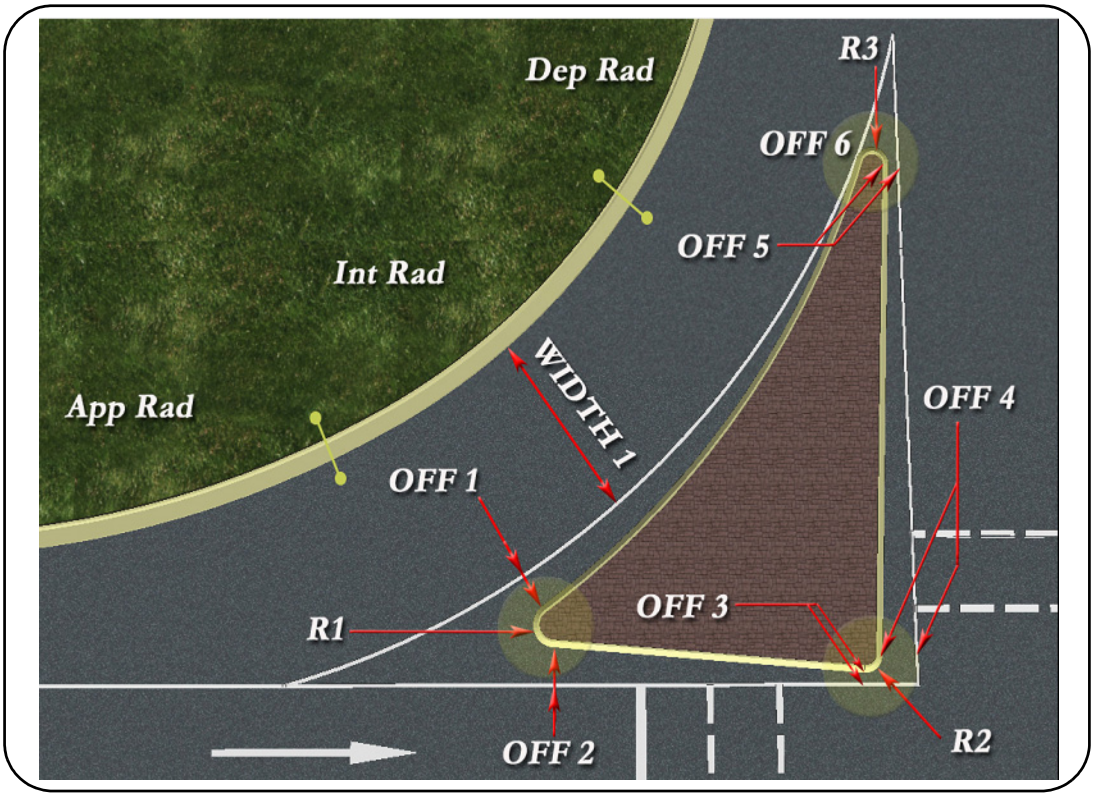
the first radius of the internal island

Radius 2

the second radius of the internal island

Radius 3

the third radius of the internal island



Merging Lane

Create?

whether or not to create a merging lane

Name

an optional name to use for the merging lane

Lane width

the width of the merging lane

Taper length

the length of the leading taper on the merging lane

Taper approach radius

the approach radius of the taper

Taper Departure radius

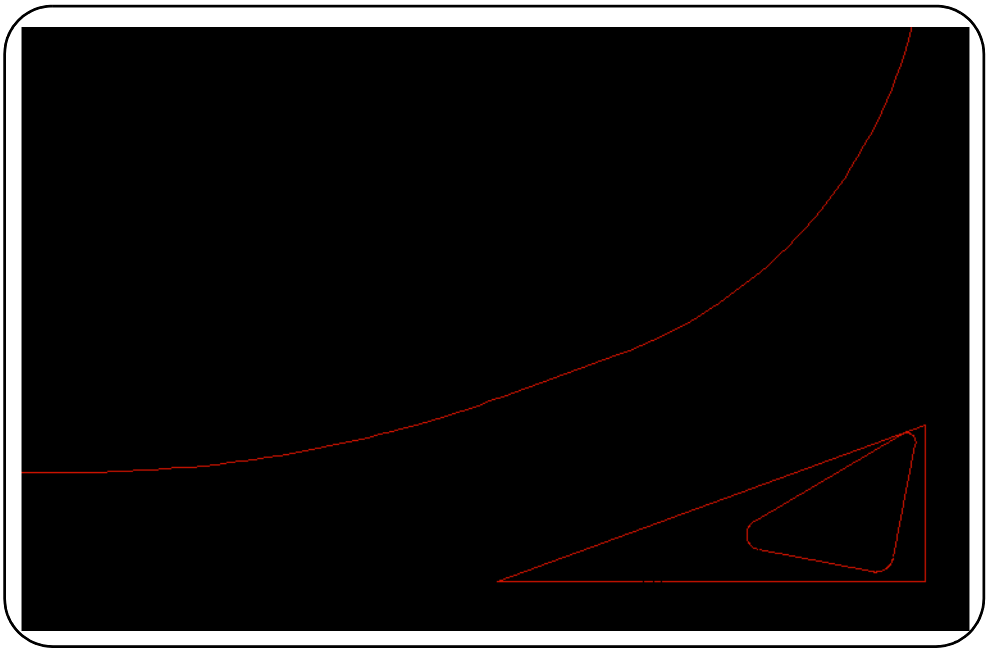
the departure radius of the taper

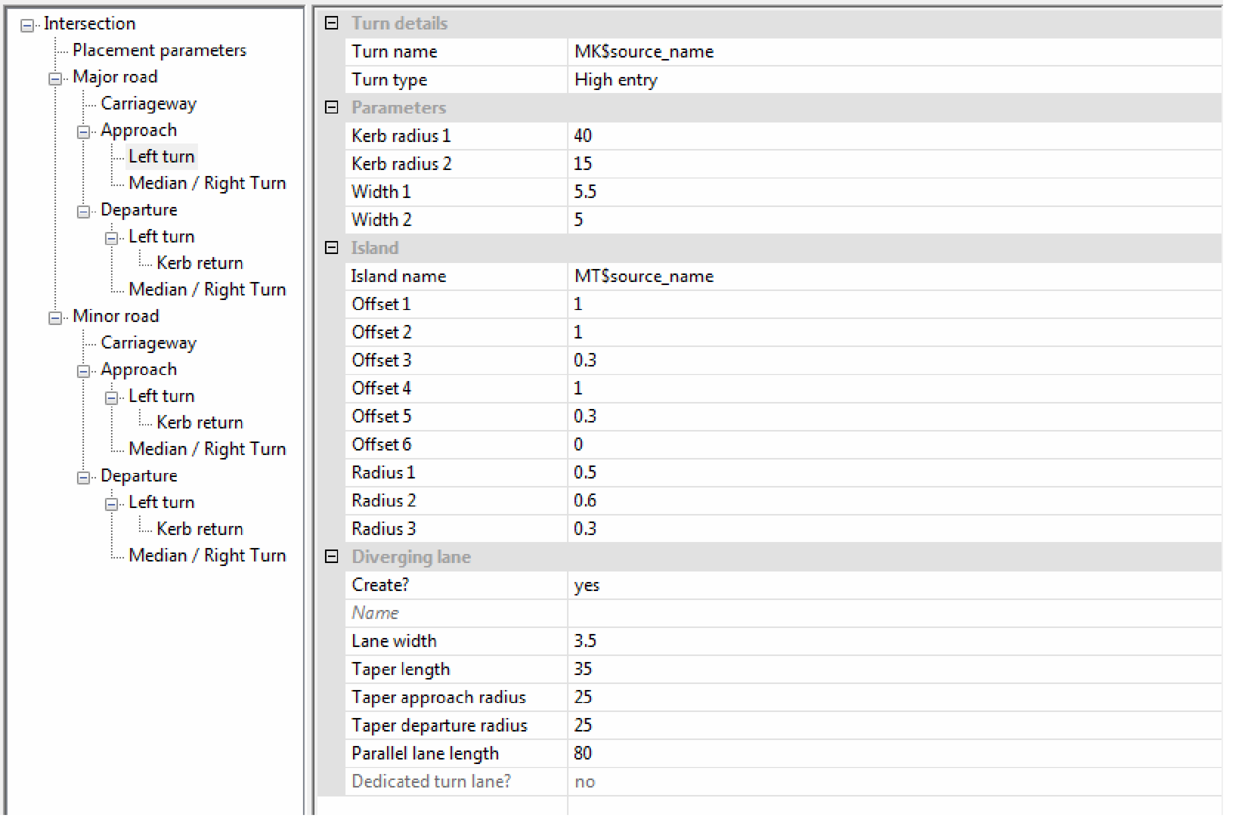
Parallel lane length

the length of the parallel lane of the merging lane

High Entry Turn

A **High Entry Turn** supports an island and optional diverging lane, as well as an optional dedicated turning lane when no diverging lane is used.





The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Turn Details

Turn name

the name to use for created geometry

Turn type

the type of turn (in this case, High entry)

Parameters

Kerb radius 1

the approach radius of the high entry kerb

Kerb radius 2

the departing radius of the high entry kerb

Width 1

the distance between the kerb return and the painted island

Width 2

the width of the painted island

Island

Island name

the name of the island string

Offset 1

the first offset of the painted island from the internal island

Offset 2

the second offset of the painted island from the internal island

Offset 3

the third offset of the painted island from the internal island

Offset 4

the fourth offset of the painted island from the internal island

Offset 5

the fifth offset of the painted island from the internal island

Offset 6

the sixth offset of the painted island from the internal island

Radius 1

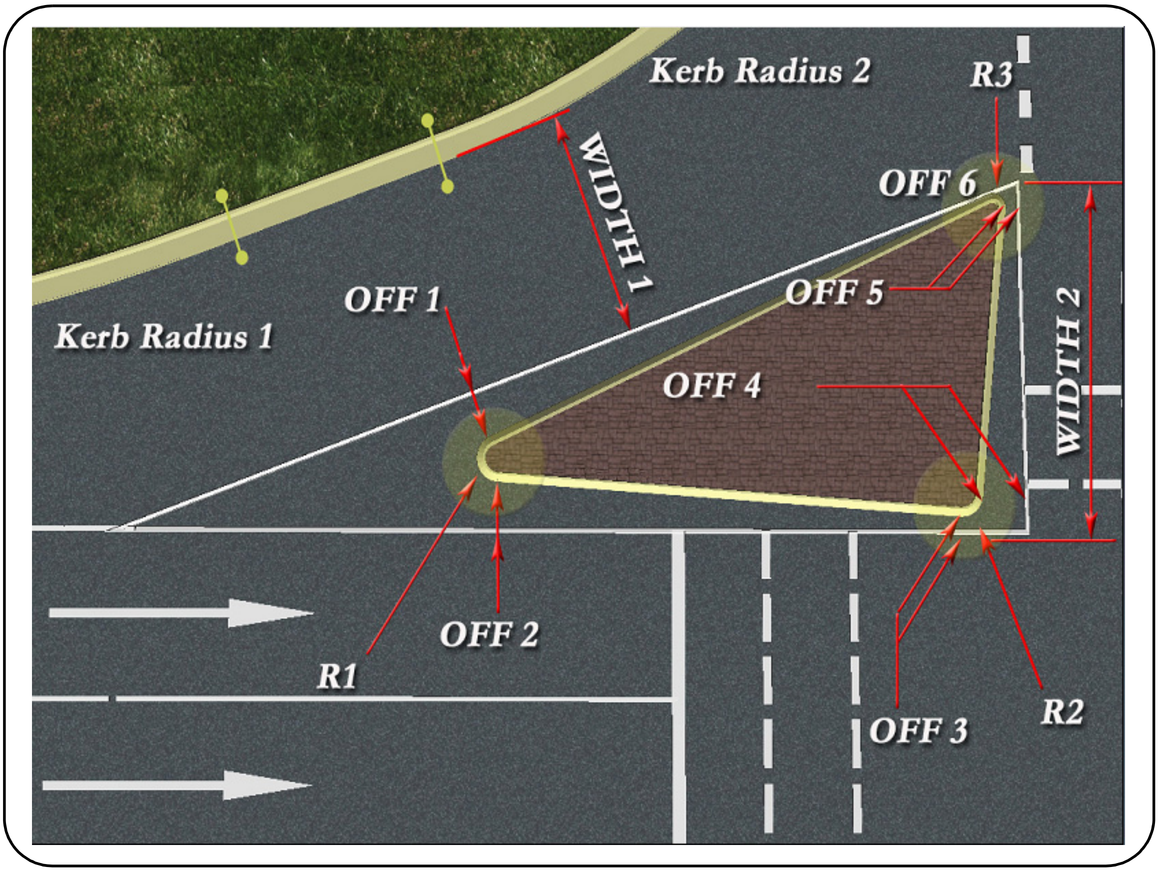
the first radius of the internal island

Radius 2

the second radius of the internal island

Radius 3

the third radius of the internal island

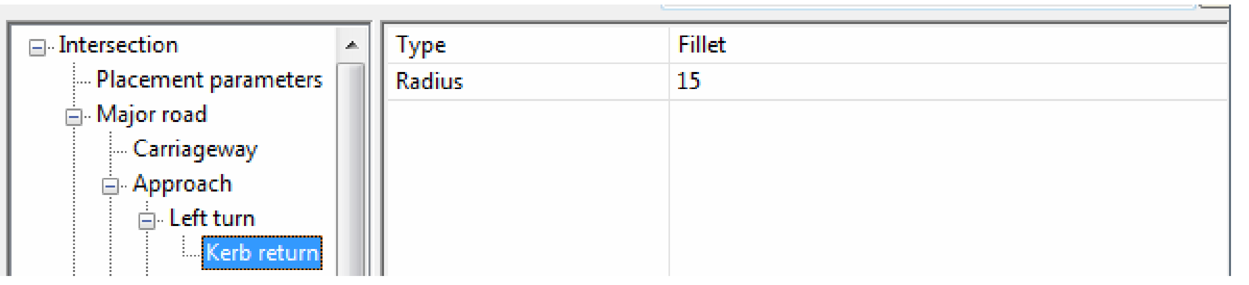

Kerb Radius 2
R3
WIDTH 1
OFF 6
OFF 5
OFF 1
Kerb Radius 1
OFF 4
WIDTH 2
OFF 2
R1
OFF 3
R2

20.9.6 Approach / Departure =>Left Turn =>Kerb Return Node

The stand up and free left turn types allow you to define the kerb return to be used.

These options include 20.9.6.1 Fillet Kerb Return, 20.9.6.2 Two Centred Kerb Return and 20.9.6.3 Three Centred Kerb Return

20.9.6.1 Fillet Kerb Return



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

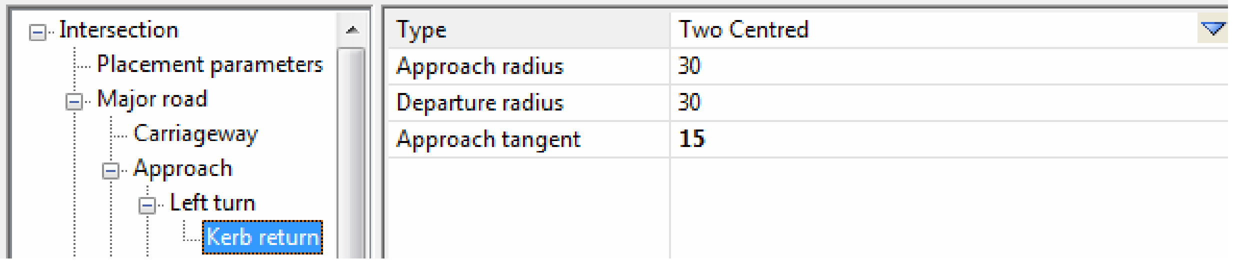
Type

the type of kerb return to create (in this case, ***Fillet****)*

Radius

the radius for the fillet kerb return

20.9.6.2 Two Centred Kerb Return



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Type

the type of kerb return to create (in this case, ***Two Centred****)*

Approach radius

the approach radius of the two centred curve

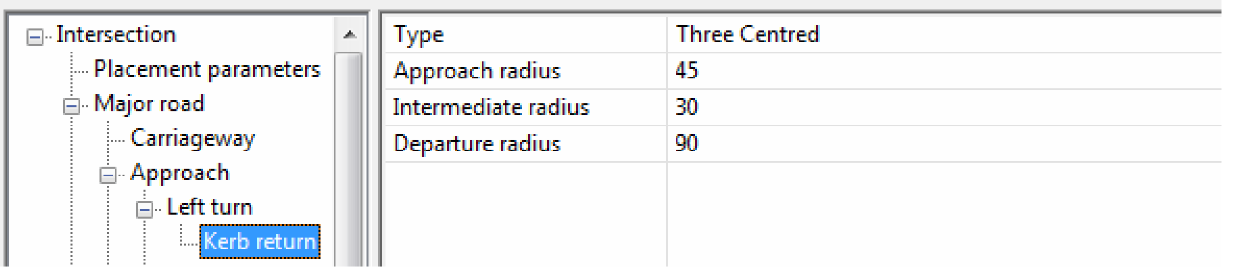
Departure radius

the departure radius of the two centred curve

Approach tangent

the approach tangent for the two centred curve

20.9.6.3 Three Centred Kerb Return



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Type

*the type of kerb to create (in this case, **Three Centred**)*

Approach radius

the approach radius of the three centred curve

Intermediate radius

the intermediate radius of the three centred curve

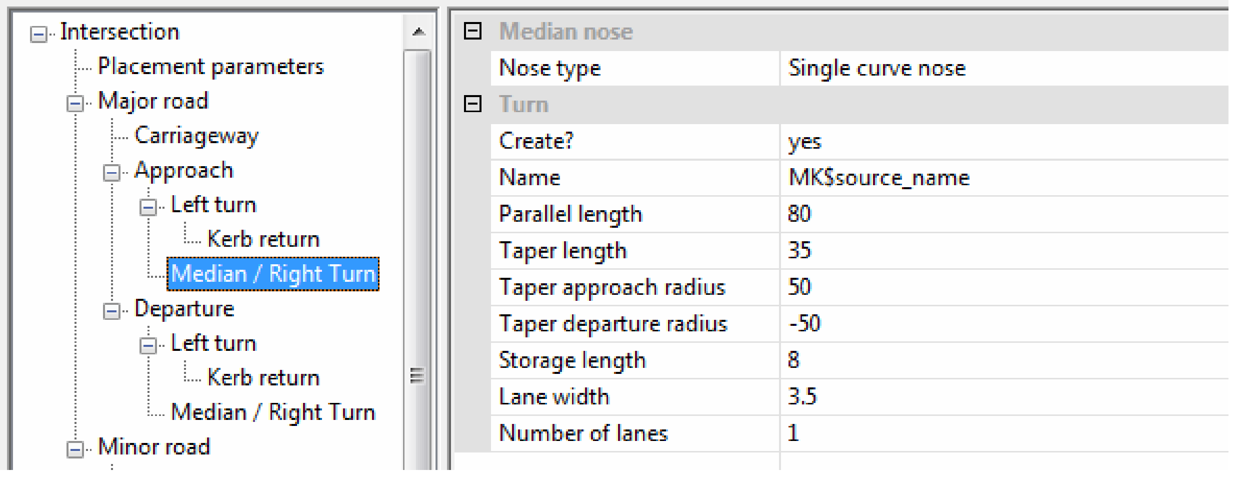
Departure radius

the departure radius of the three centred curve

20.9.7 Approach / Departure =>Median / Right Turn Node

This node is used to define the median and an optional right turn for an intersection road approach or departure.
Note that if you are using Right Side driving, then this will say Median / Left Turn.

This node is only used if a median has been defined as part of the carriageway.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Median Nose

Nose type

the type of nose to create for the median (see 20.9.7.1 Nose Types*)*

Turn

Create?

whether or not to create a right turn

Name

the name to apply to the turn

Parallel length

the length of the parallel lane

Taper

the length of the leading taper

Taper approach radius

the approach radius of the leading taper

Taper departure radius

the departure radius of the leading taper

Storage length

the storage length of the turn lane

Lane width

the width of the turn lane - must be less than the median width

Number of lanes

the number of turn lanes to create

20.9.7.1 Nose Types

For more information on *Single Curve Nose* see 20.9.7.1.1 Single Curve Nose

Two Curve Nose 20.9.7.1.2 Two Curve Nose

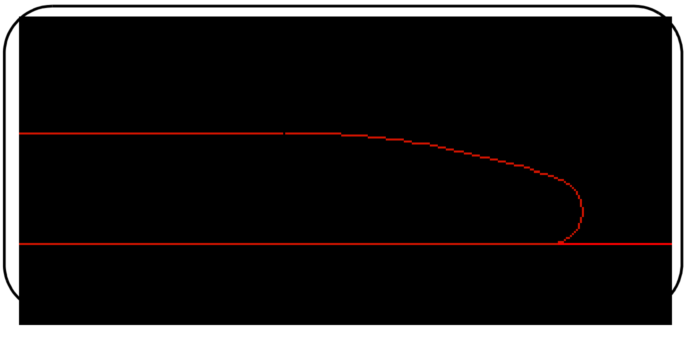
Three Curve Nose 20.9.7.1.3 Three Curve Nose

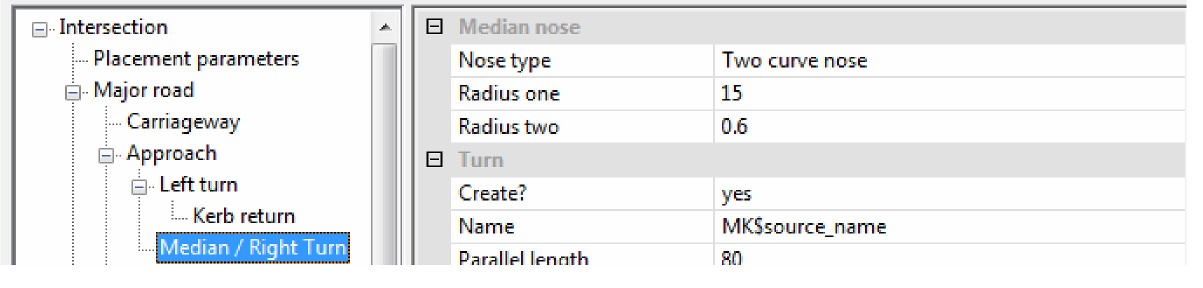
20.9.7.1.1 Single Curve Nose

a median nose created from a single fillet curve

20.9.7.1.2 Two Curve Nose

A median nose created from two curves





The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Radius one

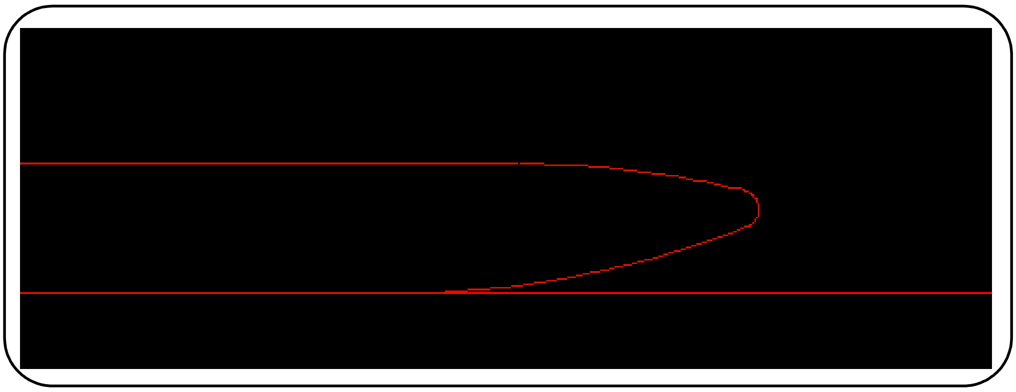
the radius of the first curve

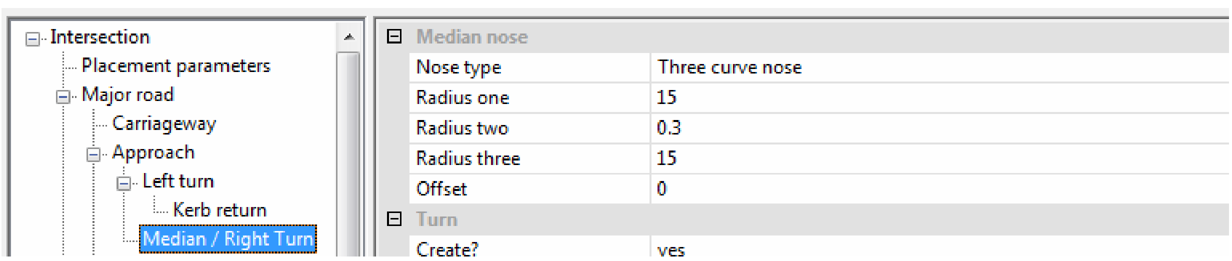
Radius two

the radius of the second curve

20.9.7.1.3 Three Curve Nose

A median nose created from three curves and an optional offset





The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Radius one

the radius of the first curve

Radius two

the radius of the second curve

Radius three

the radius of the third curve

Offset

the offset of the nose from the centreline

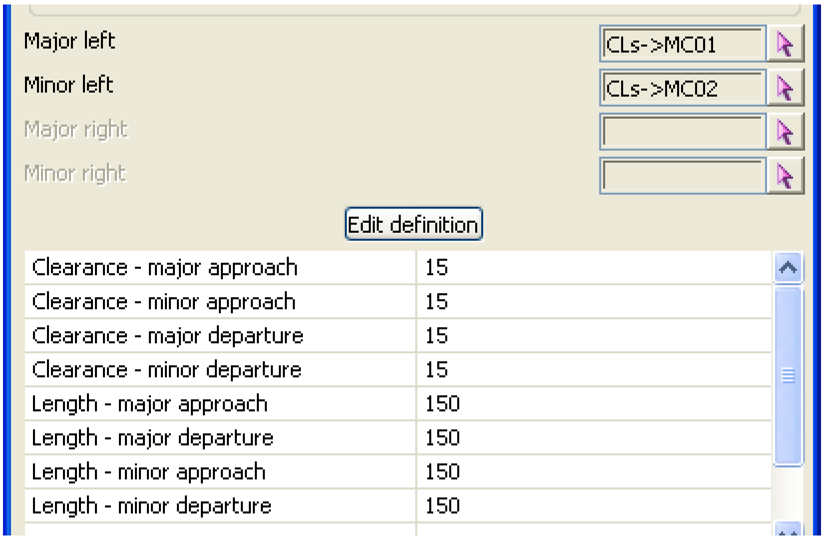
20.9.7.1.4 Placing Intersections

When placing an intersection, there are several placement specific settings. This means they are separate from the definition but help define how the intersection is placed.

These are defined when placing the component, and may be edited from the 20.9.34.1 Placed

Component Editor Some of these are also available from the 20.9.34.2 Edit a Placed Component Definition panel.

These parameters are shown below, as they are displayed when placing the component.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Major left

the left most major road

Minor left

the left most minor road

Major right

the right most major road (optional)

Minor right

the right most minor road (optional)

Major right and minor right can be set when there is a dual carriageway.

Edit definition button

Selecting this button displays the **Edit a Placed Component Definition** *panel. For more information please see* 20.9.34.2 Edit a Placed Component Definition

Clearance - major approach

the clearance of geometry on the approach of the major road, from the intersection

Clearance - minor approach

the clearance of geometry on the approach of the minor road, from the intersection

Clearance - major departure

the clearance of geometry on the departure of the major road, from the intersection

Clearance - minor departure

the clearance of geometry on the departure of the minor road, from the intersection

Length - major approach

the length of the geometry created for the approach of the major road

Length - minor approach

the length of the geometry created for the approach of the minor road

Length - major departure

the length of the geometry created for the departure of the major road

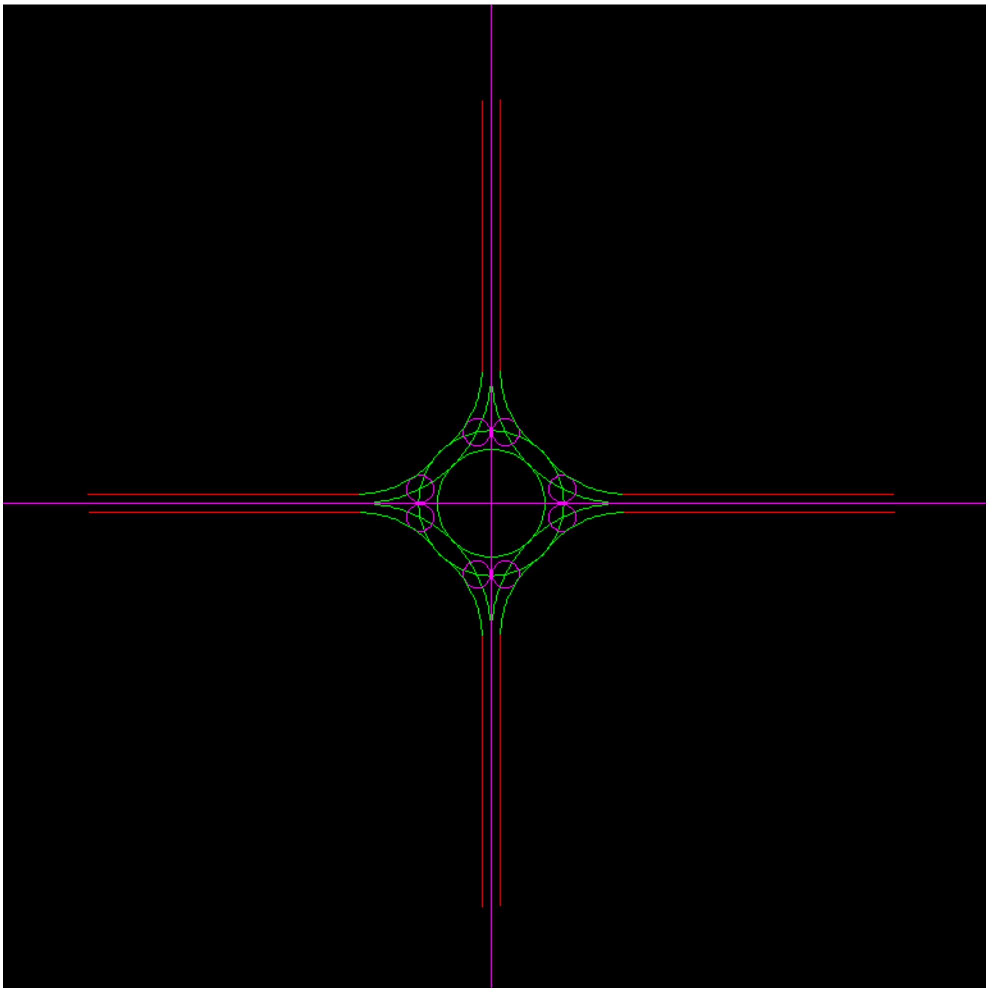
Length - minor departure

the length of the geometry created for the departure of the minor road

20.9.7.2 Roundabout

A Roundabout component can be edited via the 20.9.1 Component Library or by editing the definition of a placed component.

For information on *Defining a Roundabout* see 20.9.7.2.1 Defining a Roundabout
Placing Roundabouts 20.9.11.7.1 Placing Roundabouts



20.9.7.2.1 Defining a Roundabout

A Roundabout is defined around two intersecting roads - the **Major road** and the **Minor road**.

Each road, major or minor, may have an approach (the section of the road leading in to the roundabout) and the departure (the section of the road leading away from the roundabout).

The approach and departure for both the Major and Minor road are defined by the same

set of parameters.

For information on:

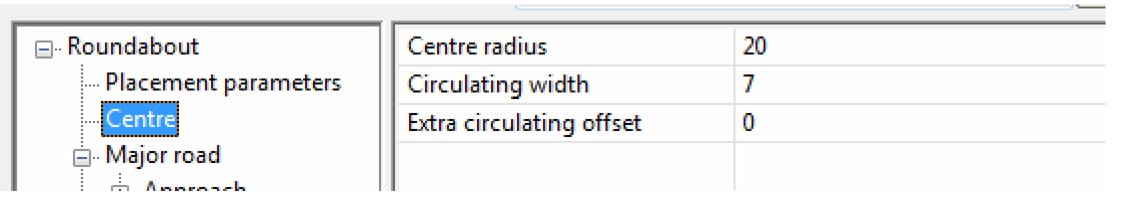
Placement Parameters Node see	20.9.8 Placement Parameters
Centre Node	20.9.9 Centre Node
Approach/ Departure Node	20.9.10 Approach / Departure Node
Approach/ Departure => Kerb Node	20.9.11 Approach / Departure => Kerb Node

20.9.8 Placement Parameters

This node is only available when editing a placed component. For more information on the fields displayed here, see the section on 20.9.11.7.1 Placing Roundabouts.

20.9.9 Centre Node

This node defines parameters for the centre of the roundabout.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Centre radius

the radius of the centre island of the roundabout

Circulating width

the width around the centre island

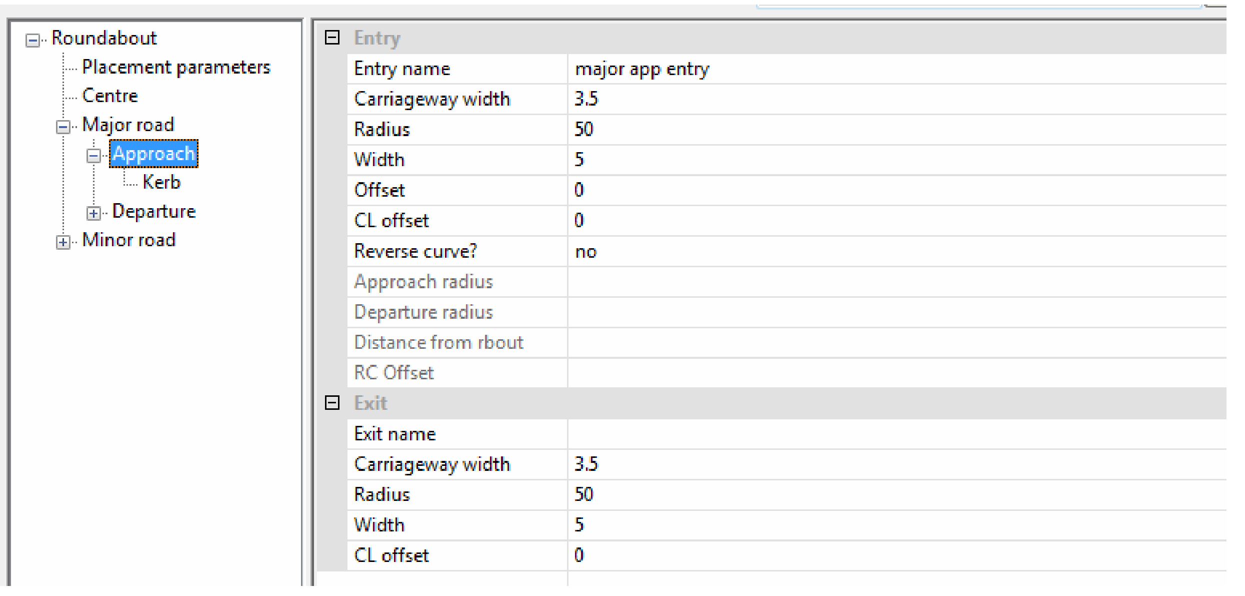
Extra circulating offset

an additional offset to apply to throat and kerb return strings from the circulating width

20.9.10 Approach / Departure Node

This node defines parameters for the approach or departure of a specific Roundabout road.

This node is made up of two sub parts - the **Entry** into the roundabout and the **Exit** from the roundabout. They can be configured independently.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Entry

Entry name

the name to apply to the entry string

Carriageway width

the width of the carriageway at the entry to the roundabout

Radius

the radius of the entry into the roundabout

Width

the width of the entry throat into the roundabout

Offset

an offset of the entry string from the centre of the roundabout

CL offset

an offset of the entry from the centreline, for medians

Reverse curve?

whether or not to create a reverse curve (yes or no)

Approach radius

the approach radius for the reverse curve

Departure radius

the departure radius of the reverse curve

Distance from rbout

the distance of the start of the reverse curve from the roundabout

RC Offset

the offset of the reverse curve from the centreline

Exit

Exit name

the name to apply to the exit string

Carriageway width

the width of the carriageway at the exit to the roundabout

Radius

the radius of the exit from the roundabout

Width

the width of the exit throat from the roundabout

CL Offset

an offset of the exit string from the centre of the roundabout

20.9.11 Approach / Departure => Kerb Node

This defines the kerb return between the entry at the approach / departure and the adjacent exit.

There are several types of kerb returns available.

- *Arc / Three Tangents / Arc see* 20.9.11.2 Arc / Three Tangents / Arc
- *Three Tangents / Locked Arc / Three Tangents see* 20.9.11.3 Three Tangents / Locked Arc / Three Tangents
- *Three Tangents / Arc / Three Tangents see* 20.9.11.4 Three Tangents / Arc / Three Tangents
- *Fillet see* 20.9.11.5 Fillet
- *Two Centre Curve see* 20.9.11.6 Two Centre Curve
- *Three Centre Curve see* 20.9.11.7 Three Centre Curve

Getting the geometry right for your roundabout can be complex. The geometry is not guaranteed to solve and you may need to try another kerb return type if the current one does not suit your needs.

20.9.11.1 Common Fields

Name

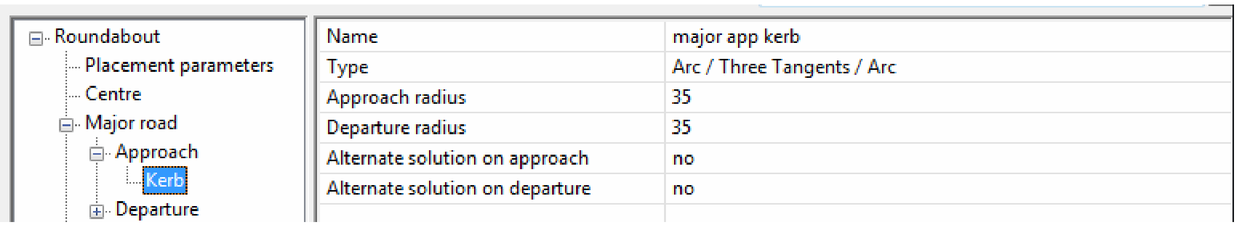
the name of the kerb return

Type

the type of kerb return

20.9.11.2 Arc / Three Tangents / Arc

This creates a kerb return with an approaching arc, an arc with three tangents (to the edge of the entry throat, the circulating width and the edge of the exit throat) and a departing arc.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Approach radius

the radius of the approach arc

Departure radius

the radius of the departure arc

Alternate solution on approach

whether or not to attempt a different solution on the approach arc

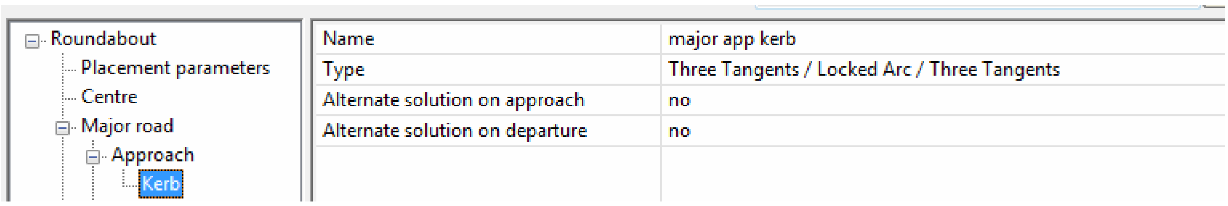
Alternate solution on departure

whether or not to attempt a different solution on the departure arc

20.9.11.3 Three Tangents / Locked Arc / Three Tangents

This creates a kerb return with the following components:

- an arc tangential to the entry carriageway, the edge of the entry throat and the circulating width of the roundabout
- an arc with a radius locked to the radius of the circulating width
- an arc tangential to the circulating width of the roundabout, the edge of the exit throat and the exit carriageway



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Alternate solution on approach

whether or not to attempt a different solution on the approach arc

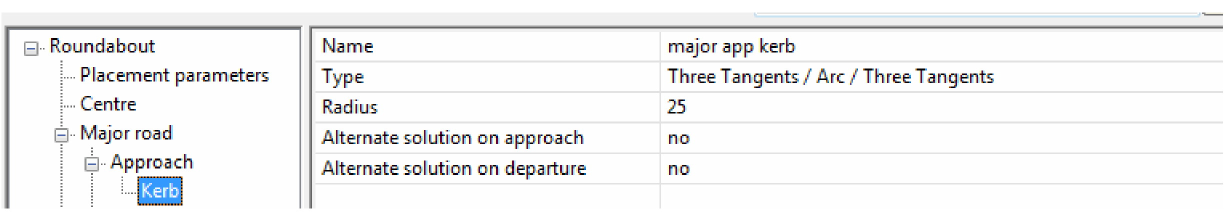
Alternate solution on departure

whether or not to attempt a different solution on the departure arc

20.9.11.4 Three Tangents / Arc / Three Tangents

This creates a kerb return with the following components:

- an arc tangential to the entry carriageway, the edge of the entry throat and the circulating width of the roundabout
- an arc with a user defined radius
- an arc tangential to the circulating width of the roundabout, the edge of the exit throat and the exit carriageway



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Radius

a user defined radius for the internal arc

Alternate solution on approach

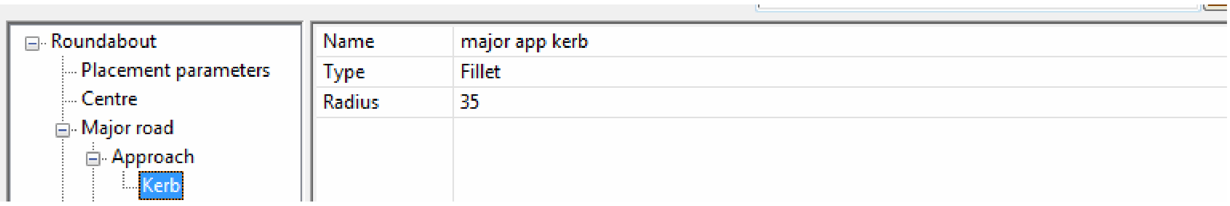
whether or not to attempt a different solution on the approach arc

Alternate solution on departure

whether or not to attempt a different solution on the departure arc

20.9.11.5 Fillet

This creates a kerb return as a simple fillet between the approach / departure and the adjacent carriageway.



The fields and buttons used in this panel have the following functions:

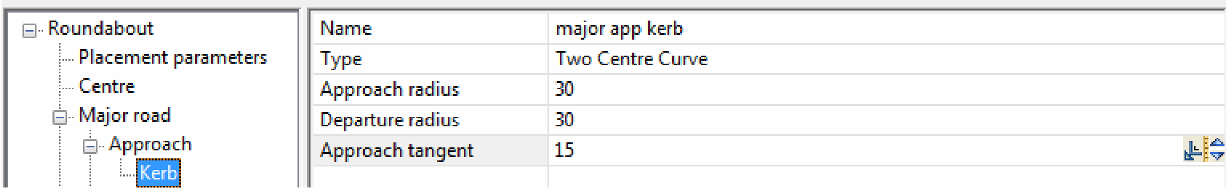
Field Description Type Defaults Pop-Up

Radius

the radius to use for the fillet

20.9.11.6 Two Centre Curve

This creates a kerb return as a two centre curve.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Approach radius

the radius for the approach of the curve

Departure radius

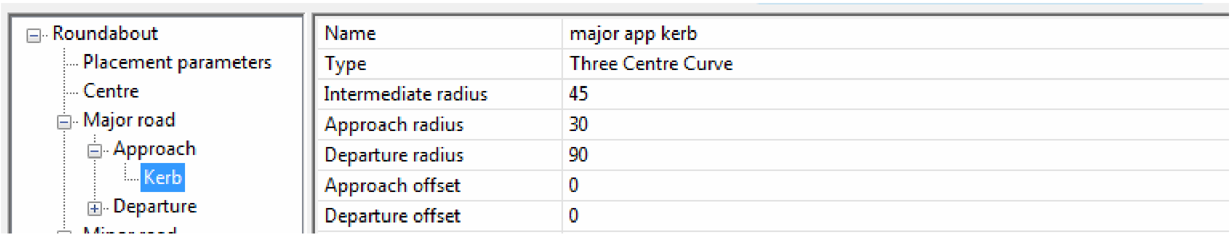
the radius for the departure of the curve

Approach tangent

the approach tangent to the curve

20.9.11.7 Three Centre Curve

This creates a kerb return as a three centre curve.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Intermediate radius

the radius of the intermediate part of the curve

Approach radius

the radius of the approach part of the curve

Departure radius

the radius of the departure part of the curve

Approach offset

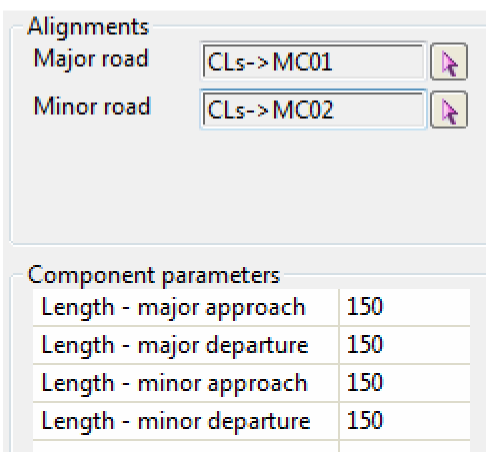
Departure offset

20.9.11.7.1 Placing Roundabouts

A roundabout is placed at the intersection of two roads - the Major and Minor road. When placing a roundabout, there are several placement specific settings. This means they are separate from the definition but help define how the roundabout is placed.

These are defined when placing the component, and may be edited from the 20.9.34.1 Placed Component Editor. Some of these are also available from the 20.9.34.2 Edit a Placed Component Definition panel.

These parameters are shown below, as they are displayed when placing the component.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Alignments

Major road

the major road of the roundabout

Minor road

the minor road of the roundabout

Component parameters

Length - major approach

the length strings to generate along the major approach

Length - major departure

the length strings to generate along the major departure

Length - minor approach

the length strings to generate along the minor approach

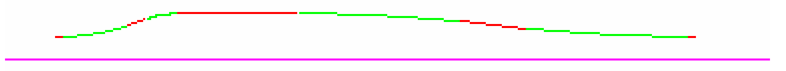
Length - minor departure

the length strings to generate along the minor departure

Note that setting a length to 0 will effectively close that leg of the roundabout.

20.9.11.8 Bus Bay

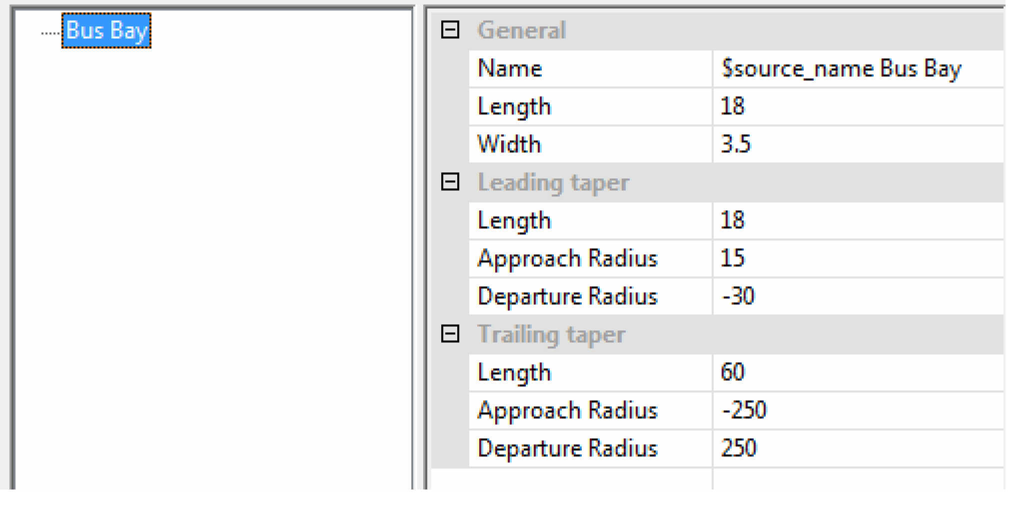
A **Bus Bay** component can be edited via the 20.9.1 Component Library or by editing the definition of a placed component.



For information on *Defining a Bus Bay* see 20.9.11.8.1 Defining a Bus Bay
Placing Bus Bays 20.9.11.8.2 Placing Bus Bays

20.9.11.8.1 Defining a Bus Bay

A **Bus Bay** is defined as being created from a known chainage on one centreline string.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

General

Name

the name of the Bus Bay string

Length

the length of the Bus Bay

Width

the width of the widest part of the Bus Bay

Leading Taper

Length

the length of the leading taper into the Bus Bay

Approach radius

the approach radius of the leading taper into the Bus Bay

Departure radius

the departing radius of the leading taper into the Bus Bay

Trailing Taper

Length

the length of the trailing taper from the Bus Bay

Approach radius

the approach radius of the trailing taper from the Bus Bay

Departure radius

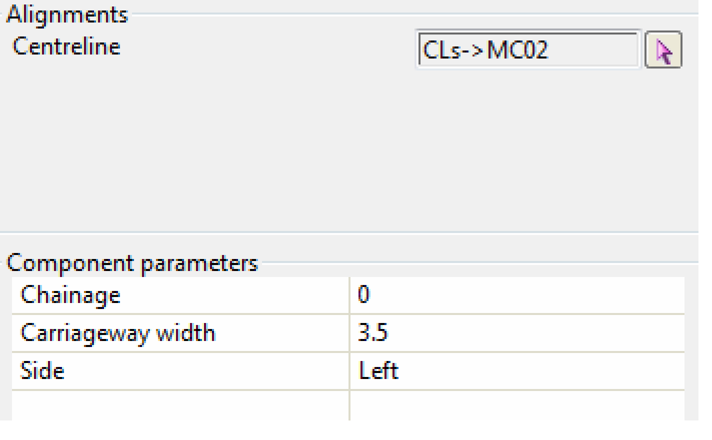
the departing radius of the trailing taper from the Bus Bay

20.9.11.8.2 Placing Bus Bays

A **Bus Bay** is placed at a known chainage along one centreline string. When placing a **Bus Bay**, there are several placement specific settings. This means they are separate from the definition but help define how the **Bus Bay** is placed.

These are defined when placing the component, and may be edited from the 20.9.34.1 Placed Component Editor some of these are also available from the 20.9.34.2 Edit a Placed Component Definition panel.

These parameters are shown below, as they are displayed when placing the component.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Centreline

the centreline string on which the Bus Bay is created

Chainage

the chainage of the Bus Bay on the nominated centreline string

Carriageway width

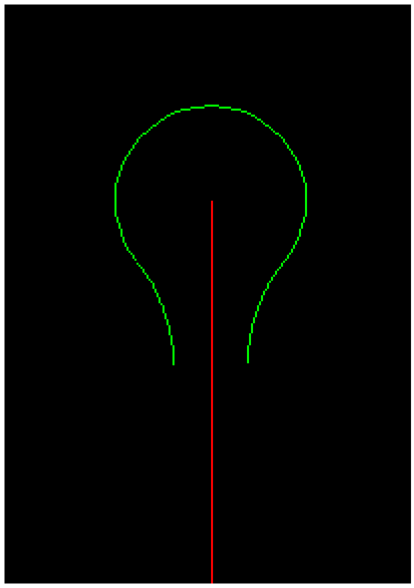
the width of the carriageway

Side

the side of the road on which the Bus Bay is created (Left or Right). This will influence the direction of the Bus Bay, depending on your drive side settings in your library.

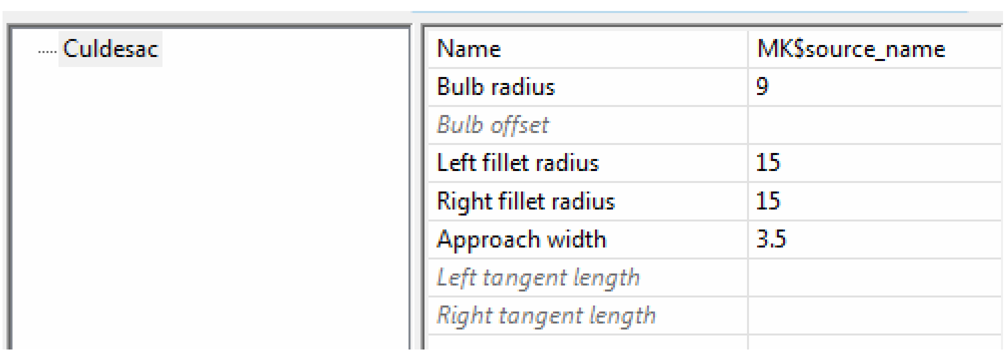
20.9.11.9 Culdesac

A **Culdesac** component can be edited via the 20.9.1 Component Library or by editing the definition of a placed component.



For information on *Defining a Culdesac* see 20.9.11.9.1 Defining a Culdesac
Placing Culdesac 20.9.11.9.2 Placing Culdesacs

20.9.11.9.1 Defining a Culdesac



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Name

the name of the culdesac string

Bulb radius

the radius of the bulb of the culdesac

Bulb offset

an optional offset of the bulb from the centreline

Left fillet radius

the radius of the left fillet of the culdesac

Right fillet radius

the radius of the right fillet of the culdesac

Approach width

the width of the carriageway approaching the culdesac

Left tangent length

an optional length of a tangent between the left fillet and the bulb

Right tangent length

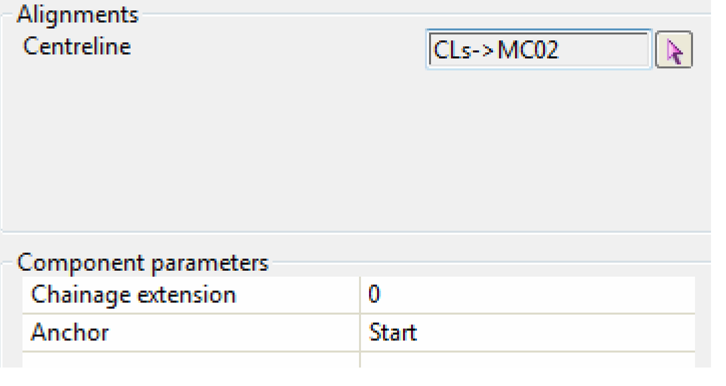
an optional length of a tangent between the right fillet and the bulb

20.9.11.9.2 Placing Culdesacs

A **Culdesac** is placed either at the start or end of a road. When placing a culdesac, there are several placement specific settings. This means they are separate from the definition but help define how the culdesac is placed.

These are defined when placing the component, and may be edited from the 20.9.34.1 Placed Component Editor. Some of these are also available from the 20.9.34.2 Edit a Placed Component Definition panel.

These parameters are shown below, as they are displayed when placing the component.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Centreline

the centreline the culdesac is created for

Chainage extension

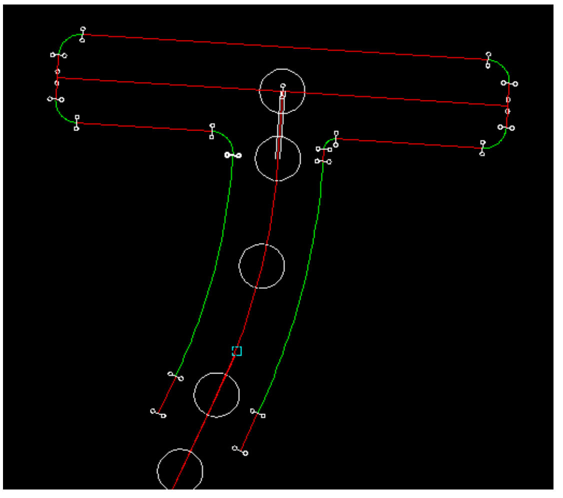
an extension from the anchor

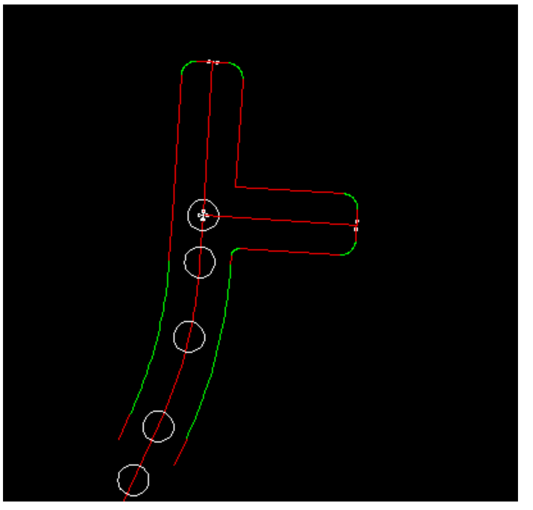
Anchor

where the culdesac is anchored (Start or End)

20.9.11.10 HAMMERHEAD

The HAMMERHEAD has parameters to cover most situations. For example, by modifying the parameters, the Hammerhead covers both of the cases:

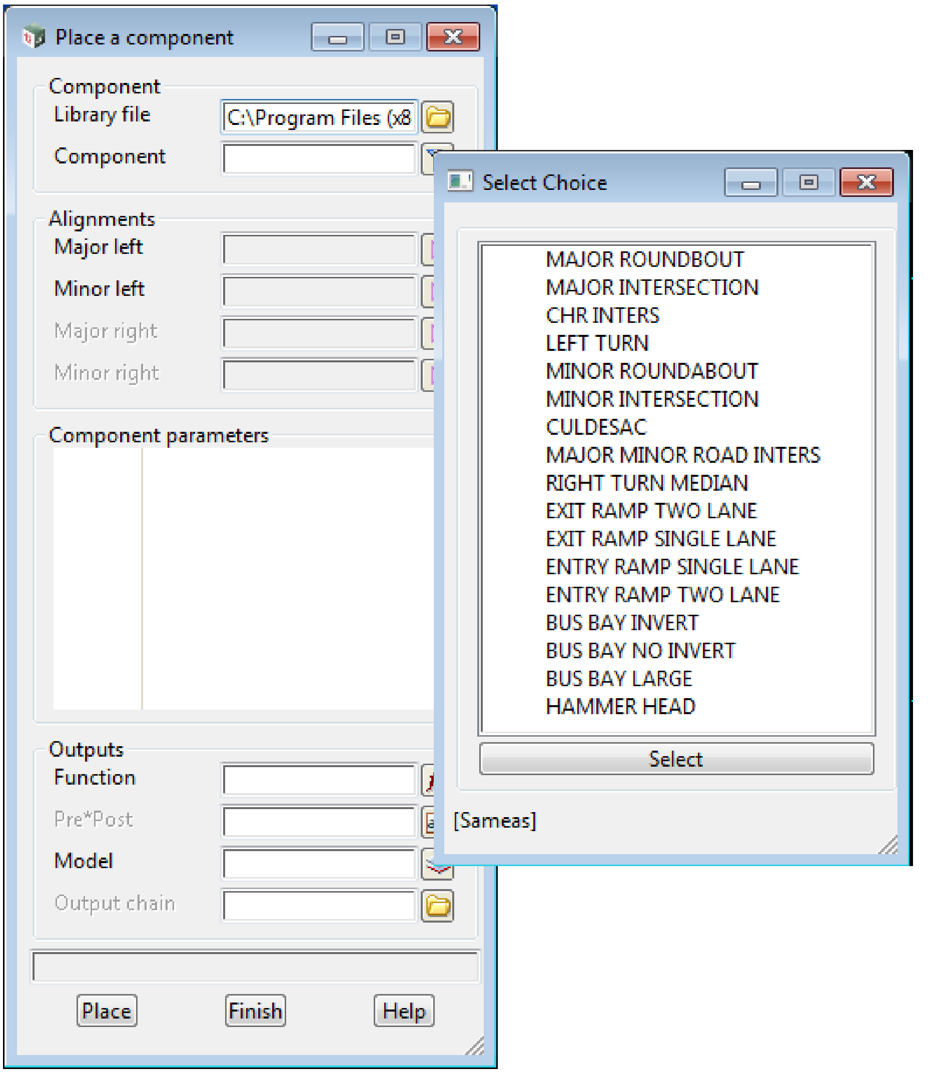




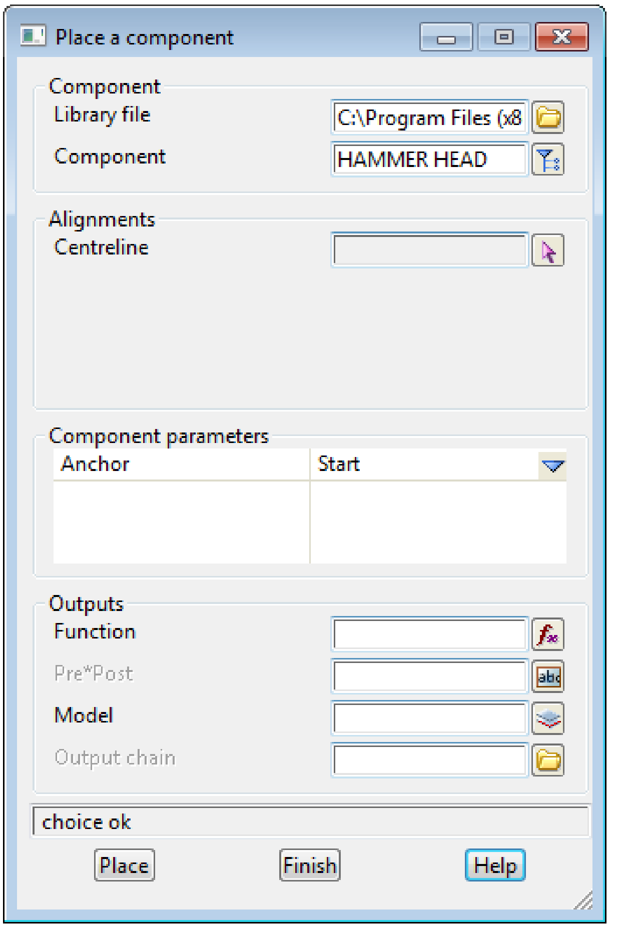
Selecting

Design =>Roads =>Components =>Place component

brings up the **Place a Component** panel.



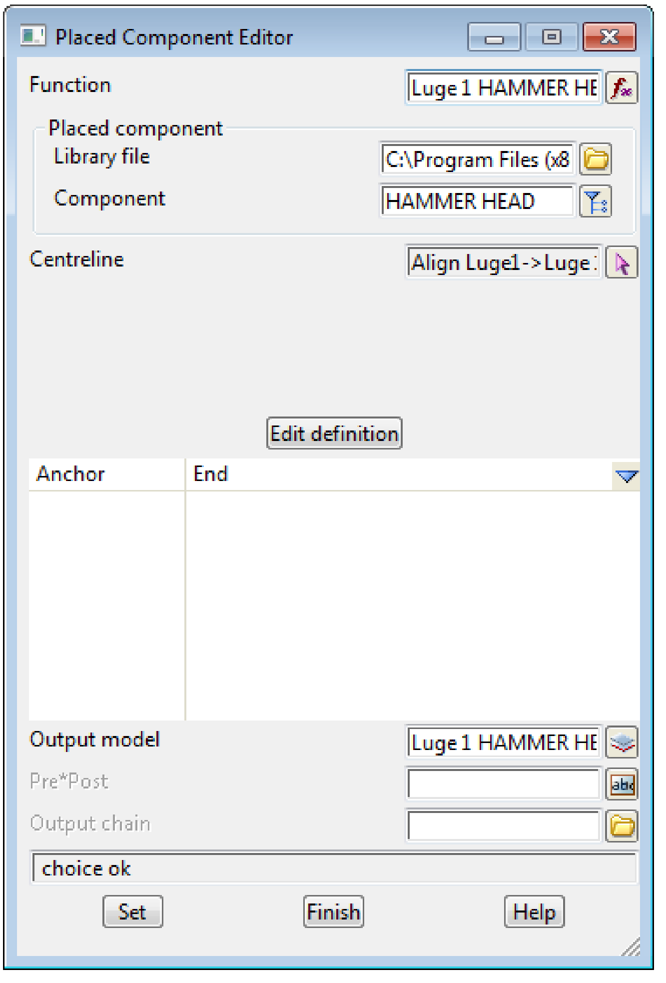
Clicking on the **Component** choice icon and selecting HAMMERHEAD puts information specific to a HAMMERHEAD on the panel.



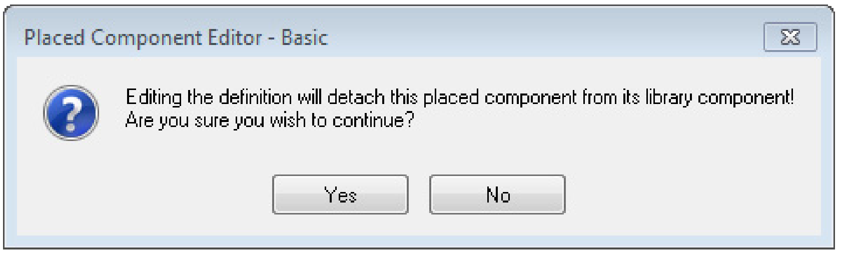
If you select a **Centreline**, the **Function** and **Model** boxes are automatically filled out using a combination of the string name, the word HAMMERHEAD and a unique number amongst the HAMMERHEAD functions already in the project.

Anchor says whether the HAMMERHEAD is to go at the **Start** or **End** of the Centreline.

Click **Place** to create the Hammerhead component.



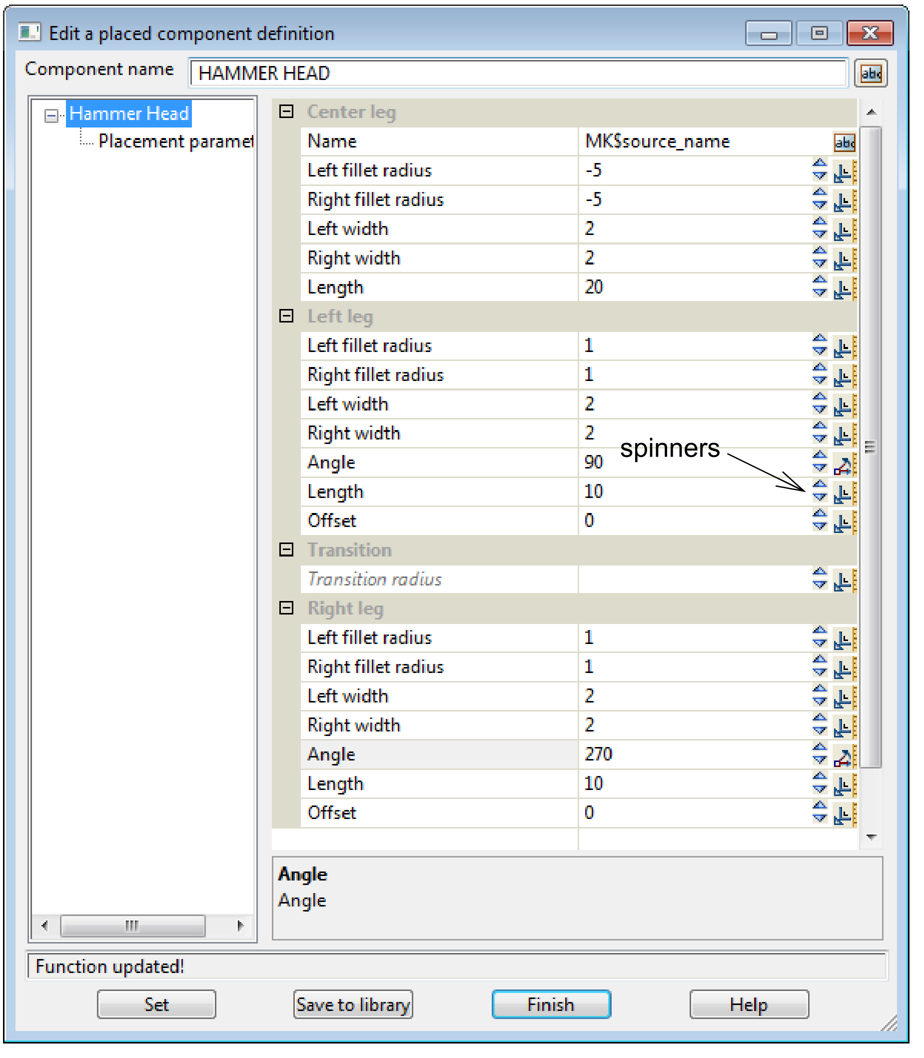
Clicking **Edit** definitions will first bring up a panel to let you know that editing the definition will modify the Hammerhead from the one that was in the library.



Clicking on **Yes** opens the **Edit a Placed Component Definition** panel.

Clicking on the tab **Placement parameters** will show the field **Anchor** (which was set before) but can be changed to either **Start** and **End**.

Clicking on **Hammer Head** displays the general tab of parameters.



The values in the fields can be modified and the Hammerhead will dynamically change. Clicking on the spinners will increase/decrease the value by a certain amount.

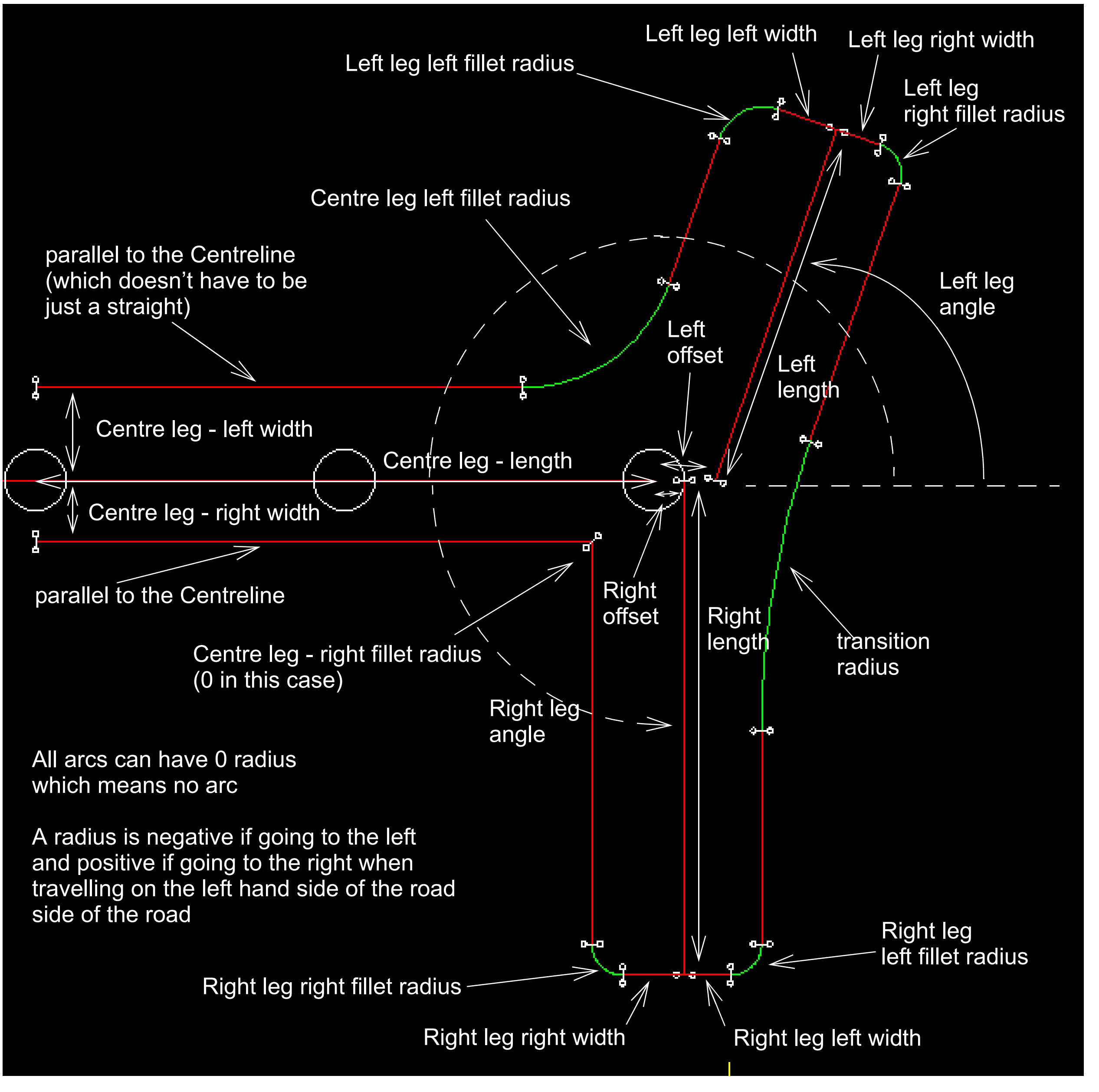
All arc radius values can be zero which means that there is no arc.

A radius is negative if it is going to the left and positive if it is going to the right, when travelling on the left hand side of the road.

Clicking on **Set** stores the current values of the parameters in the function.

Clicking on **Save to library** writes the component to the library with its current values. It is then available in the library to use in the future.

The definitions of all the parameters in the panel are:

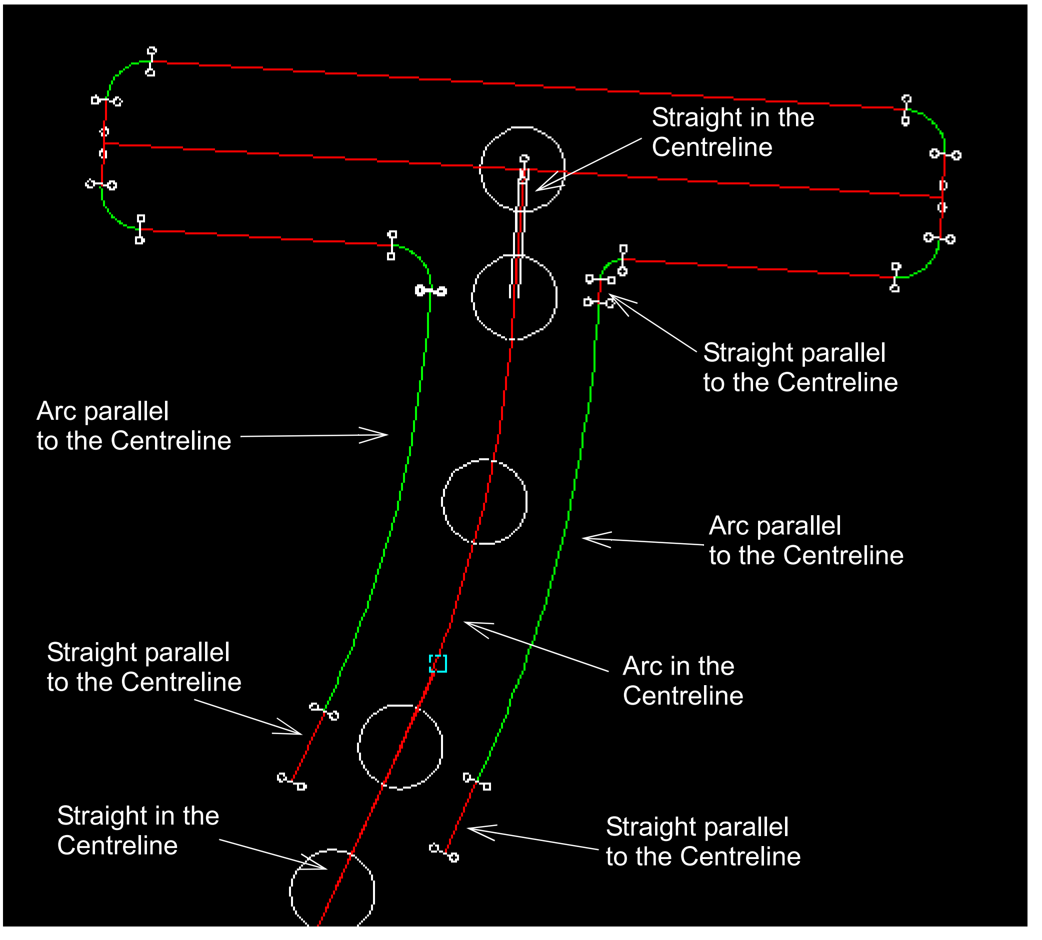


and the parameters that produced this diagram are:

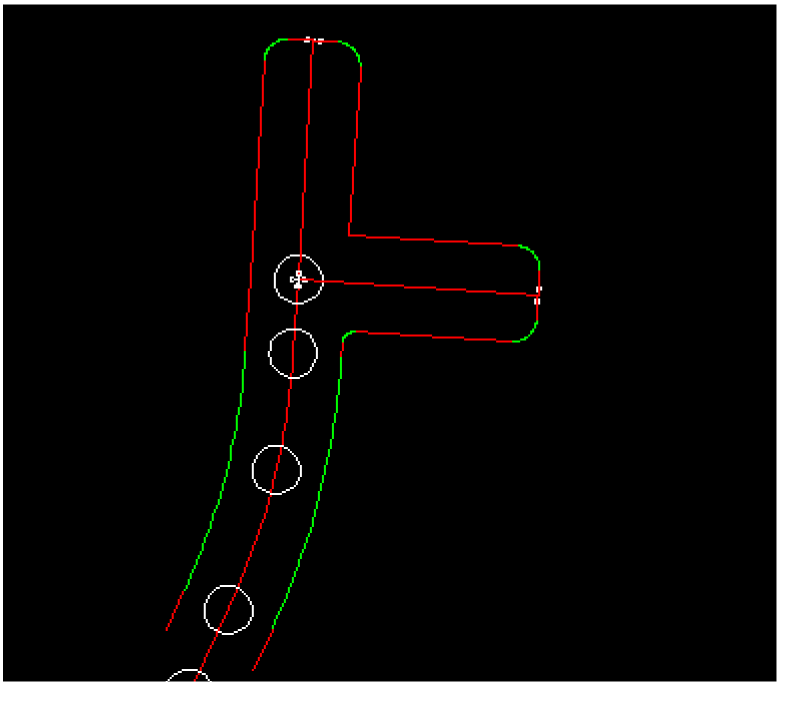
Center leg	
Name	MK$source_name
Left fillet radius	-5
Right fillet radius	0
Left width	3
Right width	2
Length	20
Left leg	
Left fillet radius	1.5
Right fillet radius	1
Left width	3.5
Right width	2.5
Angle	58
Length	12
Offset	-1
Transition	
Transition radius	-28.5
Right leg	
Left fillet radius	1
Right fillet radius	1
Left width	2.5
Right width	3
Angle	270
Length	17
Offset	1

Note that the Centreline does not have to be a straight. It could involve straights, arcs and transitions.

In the next example, the Hammerhead is coming off a Centreline and the Hammerhead **Centre leg width** value means that the **Centre leg** of the Hammerhead includes a straight, a curve and another straight from the Centreline.

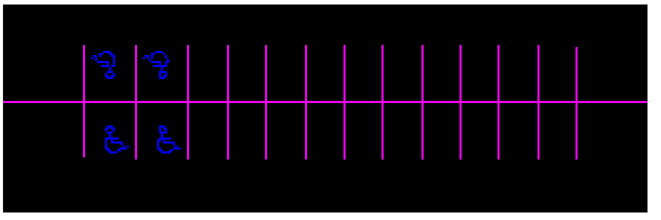


Also note that simply changing the **Left leg angle** to **0** gives



20.9.11.11 Parking Bays

Parking Bays can be edited via the 20.9.1 Component Library or by editing the definition of a placed component.



For information on *Defining a Parking Bay* 20.9.11.11.1 Defining Parking Bays
Placing Parking Bay 20.9.13.1 Placing Parking Bays

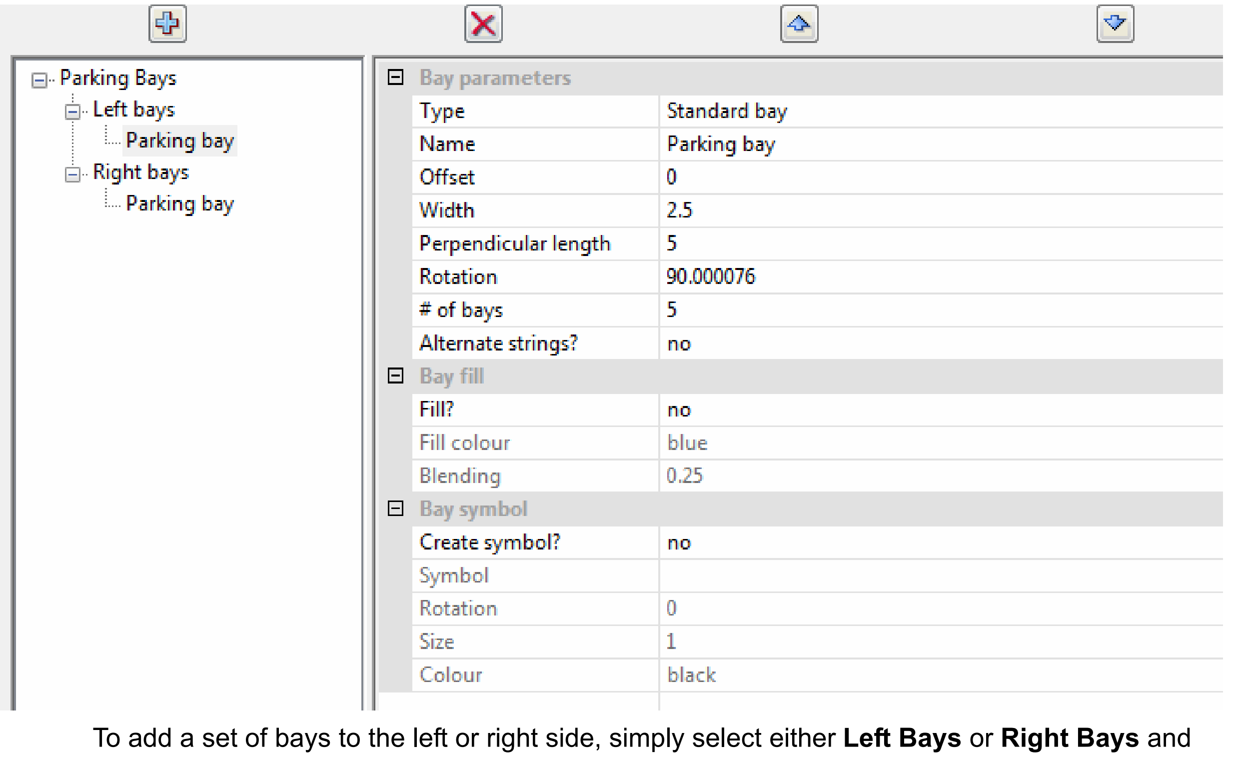
20.9.11.11.1 Defining Parking Bays

Parking Bays are created along a nominated centreline string. A **Parking Bays** component consists of a user defined number of 'sets of bays' on the left and right side.

A set of bays can consist of one or more actual parking bays. It is defined by a number of parameters for width, length, offset and rotation, as well as several cosmetic settings.

A set of bays can either be a 20.9.12 Standard Bay or a 20.9.13 Separation.

Bays on the left and right side are defined in the same manner.



To add a set of bays to the left or right side, simply select either **Left Bays** or **Right Bays** and click the  button

The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

adds a bay set to the selected side

deletes the selected bay set

moves the selected bay definition up

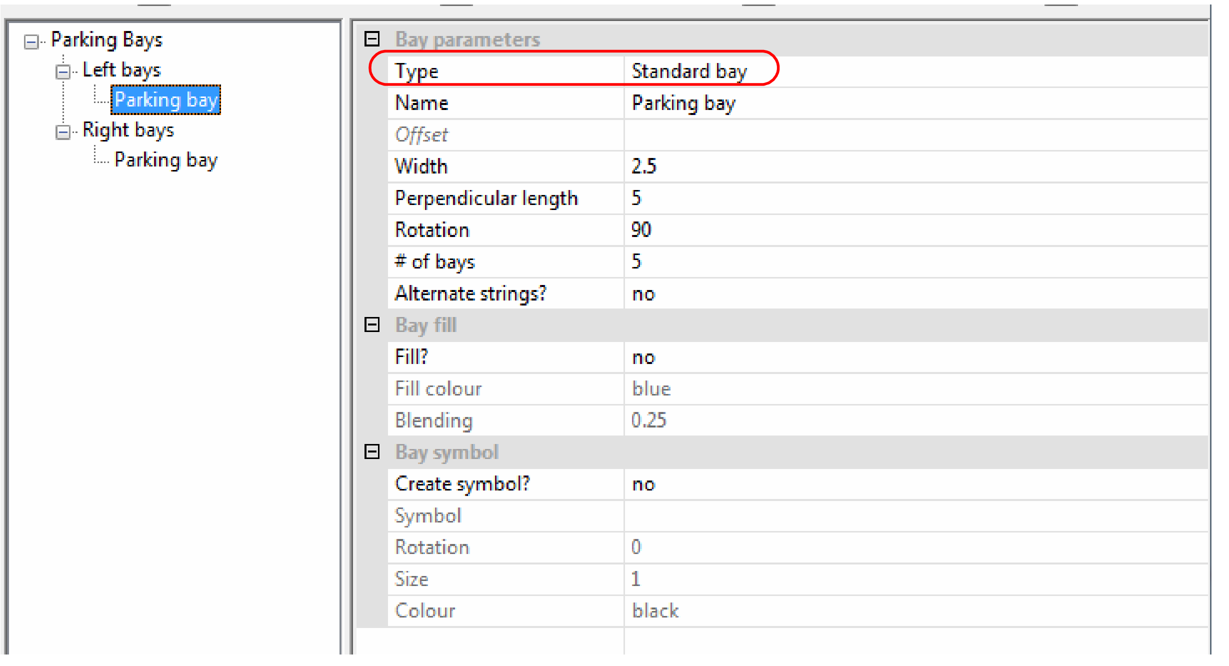
moves the selected bay definition down

Bay Parameters

Type

the type of bay - Standard or Separation. For more information please see 20.9.12 Standard Bay *and* 20.9.13 Separation

20.9.12 Standard Bay



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Bay Parameters

Name

the name of the bay

Offset

an optional offset of each bay from the centreline

Width

the width of each bay

Perpendicular length

the perpendicular length of each bay

Rotation

the rotation of each bay

# of bays

the number of bays to create for this set

Alternate strings?

whether or not to create every other string

Bay fill

Fill?

whether or not to fill each bay with a colour

Fill Colour

the colour to use when filling

Blending

an optional blend to apply, for transparency

Bay symbol

Create symbol?

whether or not to create a symbol in the centre (yes or no)

Symbol

the symbol to place

Rotation

the additional rotation of the symbol. All symbols will be rotated to match the rotation of the parking bay.

Size

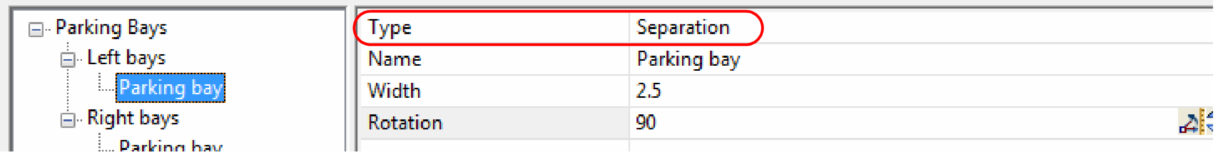
the size / scale of the symbol

Colour

the colour of the symbol

20.9.13 Separation

A separation represents a space between parking bays.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Name

the name of the separation

Width

the width of the separation

Rotation

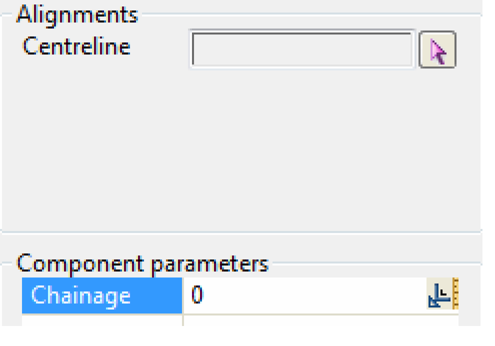
the rotation of the separation

20.9.13.1 Placing Parking Bays

Parking Bays are placed either at the start or end of a road. When placing a Parking Bay component, there are several placement specific settings. This means they are separate from the definition but help define how the Parking Bay is placed.

These are defined when placing the component, and may be edited from the 20.9.34.1 Placed Component Editor. Some of these are also available from the 20.9.34.2 Edit a Placed Component Definition panel.

These parameters are shown below, as they are displayed when placing the component.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Centreline string select

the centreline along which the bays are created

Chainage

the chainage at which the bays should start being created

20.9.13.2 Left Turn

A **Left Turn** is a subset of an intersection, compromising of just a **Left Turn.** A **Left Turn** can be edited via the 20.9.1 Component Library or by editing the definition of a placed component.

For information on *Defining a Left Turn* 20.9.13.2.1 Defining a Left Turn
Placing Left Turns 20.9.15.1.1 Placing Left Turns

20.9.13.2.1 Defining a Left Turn

A **Left Turn** is only defined by the parameters involving the **Left Turn** - the type of turn and the kerb return.

For information on Placement Parameters see 20.9.14 Placement Parameters
Left Turn Node 20.9.15 Left Turn
Left Turn => Kerb Return Node 20.9.15.1 Left Turn => Kerb Return

20.9.14 Placement Parameters

This node is only available when editing a placed component. For more information on the fields displayed here, see the section on 20.9.15.1.1 Placing Left Turns.

20.9.15 Left Turn

A **Left Turn** is defined in the same way as the Intersection Left Turn. For more information, please see the section on the 20.9.5 Left Turn - Approach / Departure

20.9.15.1 Left Turn => Kerb Return

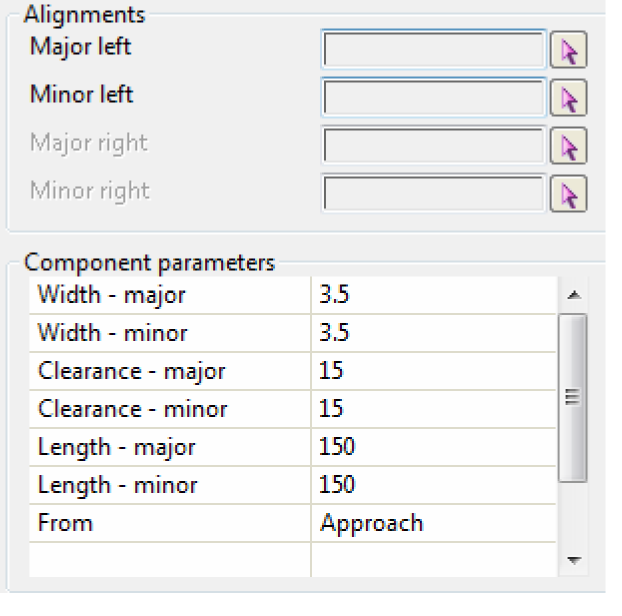
A **Left Turn Kerb Return** is defined in the same was as the Intersection kerb return. For more information, please see the section on the 20.9.6 Approach / Departure =>Left Turn =>Kerb Return Node

20.9.15.1.1 Placing Left Turns

Left Turns are placed at the intersection of two roads - a Major and a Minor, with an approach or departure. Like an intersection, they can also support dual carriageways. When placing a Left Turn component, there are several placement specific settings. This means they are separate from the definition but help define how the Left Turn is placed.

These are defined when placing the component, and may be edited from the 20.9.34.1 Placed Component Editor. Some of these are also available from the 20.9.34.2 Edit a Placed Component Definition panel.

These parameters are shown below, as they are displayed when placing the component.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Alignments

Major left

the left most major road

Minor left

the left most minor road

Major right

the right most major road (optional)

Minor right

the right most minor road (optional)

Major right and Minor right can be set when there is a dual carriageway.

Component parameters

Width - major

the width of the carriageway of the major road

Width - minor

the width of the carriageway of the minor road

Clearance - major

the clearance of geometry on major road

Clearance - minor

the clearance of geometry on minor road

Length - major

the length of the geometry created for major road

Length - minor

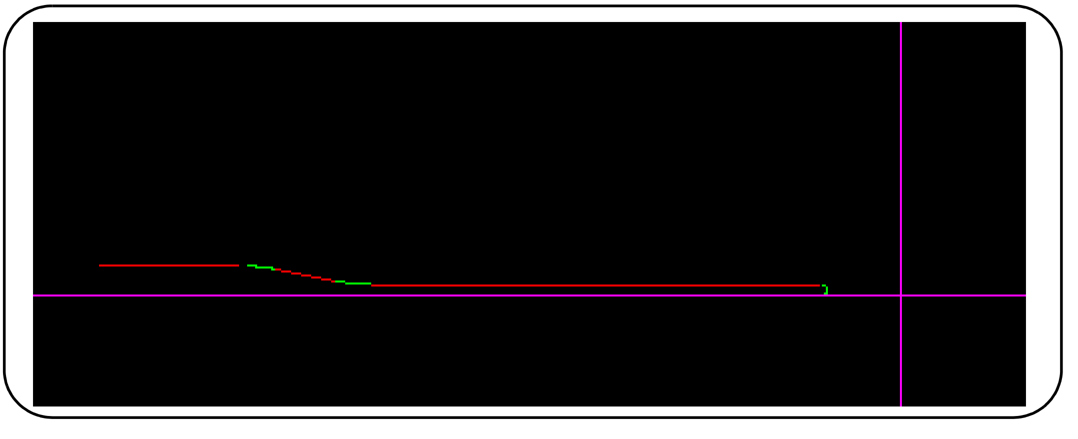
the length of the geometry created minor road

From

whether or not to create the left turn leading from the approach or the departure

20.9.15.2 Median Island

A **Median Island** is a subset of an intersection, compromising of just a median and an optional right hand turn. A **Median Island** can be edited via the 20.9.1 Component Library or by editing the definition of a placed component.

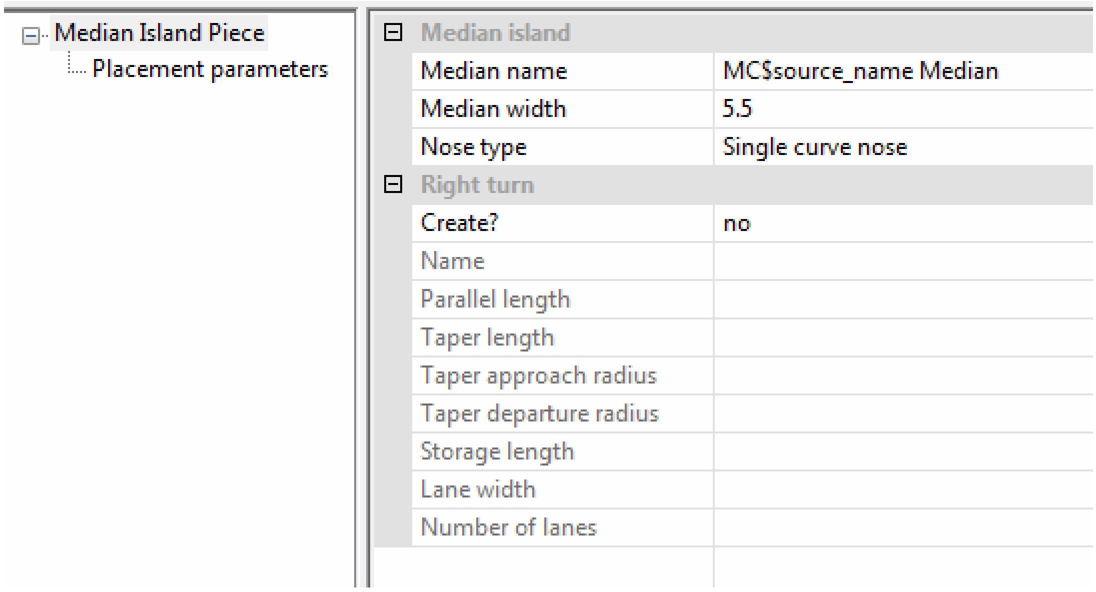


For information on *Defining a Median Island* 20.9.15.2.1 Defining a Median Island
Placing Median Islands 20.9.17.1 Placing Median Islands
Placement Parameters 20.9.17 Placement Parameters

20.9.15.2.1 Defining a Median Island

A median is only defined by the parameters involving the Median Island and the right hand turn.

20.9.16 Median Island Piece



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Median island

Median name

the name of the median string to create

Median width

the width of the median

Nose type

the type of nose to create for the median (see 20.9.16.1 Nose Types*)*

Right Turn

Create?

whether or not to create a right turn

Name

the name to apply to the turn

Parallel length

the length of the parallel lane

Taper length

the length of the leading taper

Taper approach radius

the approach radius of the leading taper

Taper departure radius

the departure radius of the leading taper

Storage length

the storage length of the turn lane

Lane width

the width of the turn lane - must be less than the median width

Number of lanes

the number of turn lanes to create

20.9.16.1 Nose Types

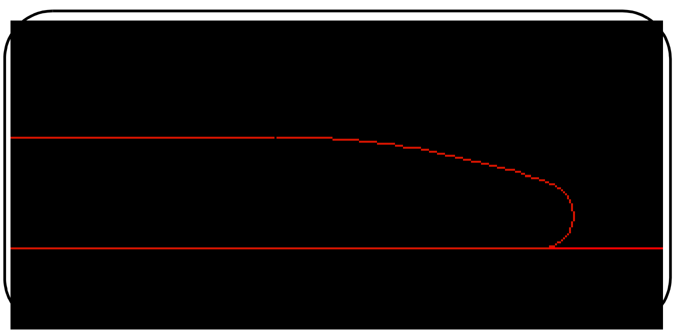
For more information on Single Curve Nose see 20.9.16.1.1 Single Curve Nose
Two Curve Nose 20.9.16.1.2 Two Curve Nose
Three Curve Nose 20.9.16.1.3 Three Curve Nose

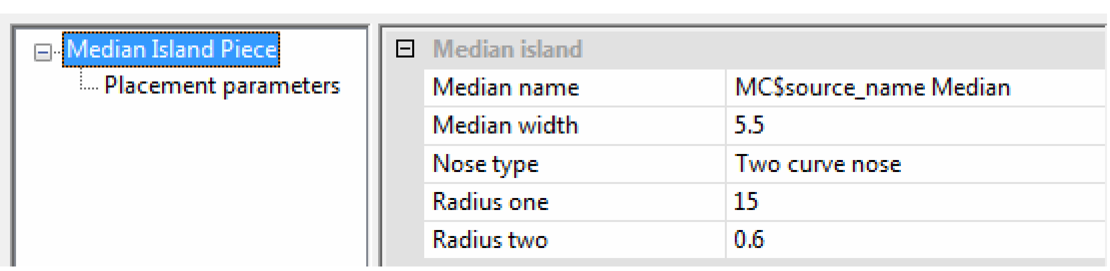
20.9.16.1.1 Single Curve Nose

a median nose created from a single fillet curve

20.9.16.1.2 Two Curve Nose

a median nose created from two curves





The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Radius one

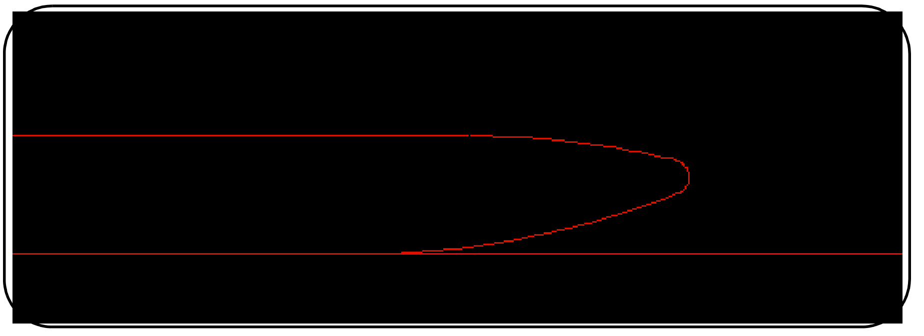
the radius of the first curve

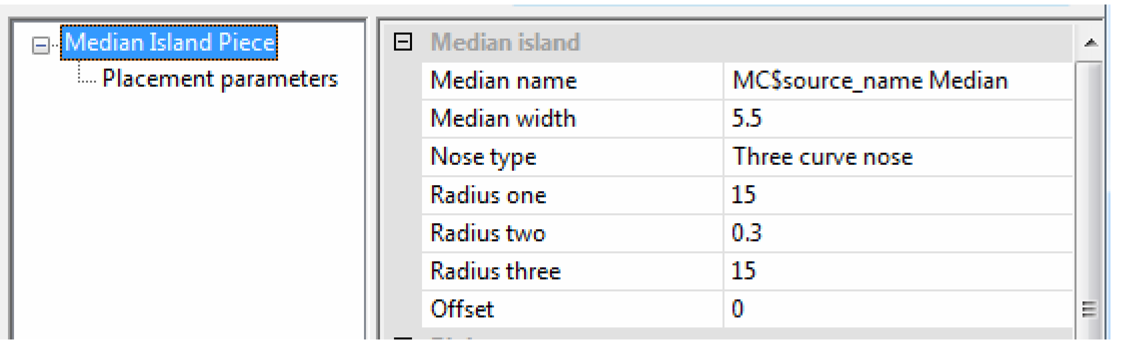
Radius two

the radius of the second curve

20.9.16.1.3 Three Curve Nose

a median nose created from three curves and an optional offset





The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Radius one

the radius of the first curve

Radius two

the radius of the second curve

Radius three

the radius of the third curve

Offset

the offset of the nose from the centreline

20.9.17 Placement Parameters

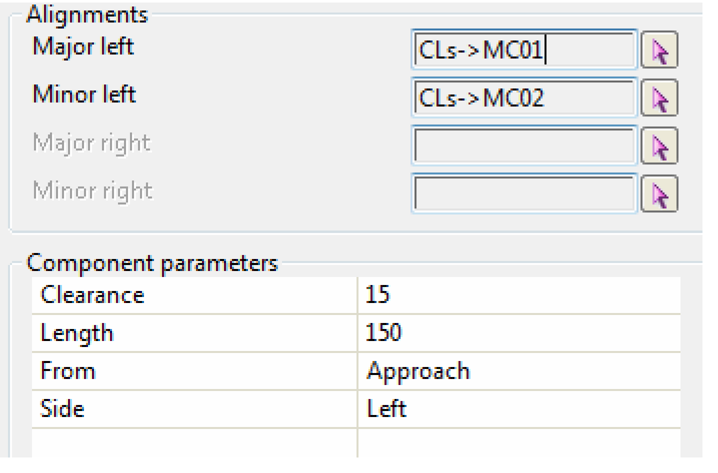
This node is only available when editing a placed component. For more information on the fields displayed here, see the section on 20.9.17.1 Placing Median Islands.

20.9.17.1 Placing Median Islands

Median Islands are placed at the intersection of two roads - a Major and a Minor, with an approach or departure. Like an intersection, they can also support dual carriageways. When placing a Median Island, there are several placement specific settings. This means they are separate from the definition but help define how the Median Island, is placed.

These are defined when placing the component, and may be edited from the 20.9.34.1 Placed Component Editor Some of these are also available from the 20.9.34.2 Edit a Placed Component Definition panel.

These parameters are shown below, as they are displayed when placing the component.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Alignments

Major left

the left most major road

Minor left

the left most minor road

Major right

the right most major road (optional)

Minor right

the right most minor road (optional)

Component parameters

Clearance

the distance from the intersection at which to start creating the median island

Length

the length of the geometry to create

From

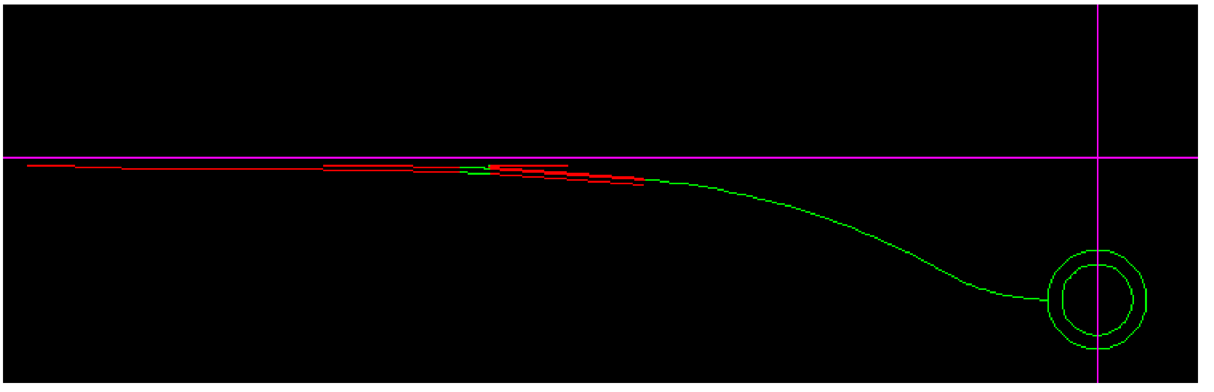
whether or not the median should be created on the approach or departure side of the major string

Side

the side on which the median should be bound

20.9.17.2 Entry Ramp

An **Entry Ramp** component creates an **Entry Ramp** from an overpass or other adjoining road to a freeway or other major road. It can be edited via the 20.9.1 Component Library or by editing the definition of a placed component.



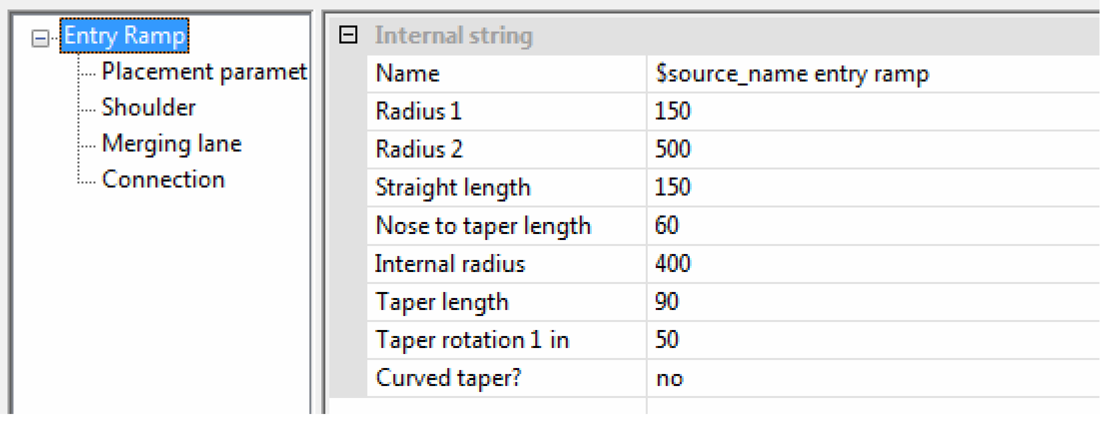
For information on *Defining an Entry Ramp* 20.9.17.2.1 Defining a Entry Ramp
Placing Entry Ramps 20.9.22.2.1 Placing Entry Ramps

20.9.17.2.1 Defining a Entry Ramp

An **Entry Ramp** is defined by the ramp itself, a shoulder, a merging lane and an optional connection.

For information on Entry Ramp node see	20.9.18 Entry Ramp Node
Placement Parameters	20.9.19 Placement Parameters
Shoulder node	20.9.20 Shoulder Node
Merging lane node	20.9.21 Merging Lane Node
Connection node	20.9.22 Connection

20.9.18 Entry Ramp Node



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Name

the name of the entry ramp string

Radius 1

the first radius of the entry reverse curve

Radius 2

the second radius of the entry reverse curve

Nose to taper length

The distance between the nose of the shoulder and the taper

Internal radius

the radius of the final, internal arc

Taper length

the length of the trailing taper

Taper rotation 1 in

the rotation of the trailing taper

Curved taper?

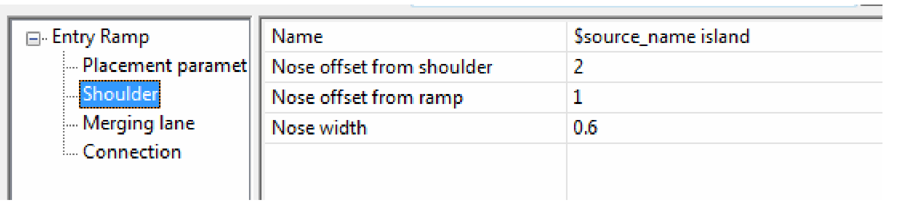
whether or not to use a curved taper (may be more suitable for curving centrelines)

20.9.19 Placement Parameters

This node is only available when editing a placed component. For more information on the fields displayed here, see the section on 20.9.22.2.1 Placing Entry Ramps.

20.9.20 Shoulder Node

This node defines properties for the shoulder of the **Entry Ramp**.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Name

the name of the shoulder string

Nose offset from shoulder

the offset of the island nose from the shoulder

Nose offset from ramp

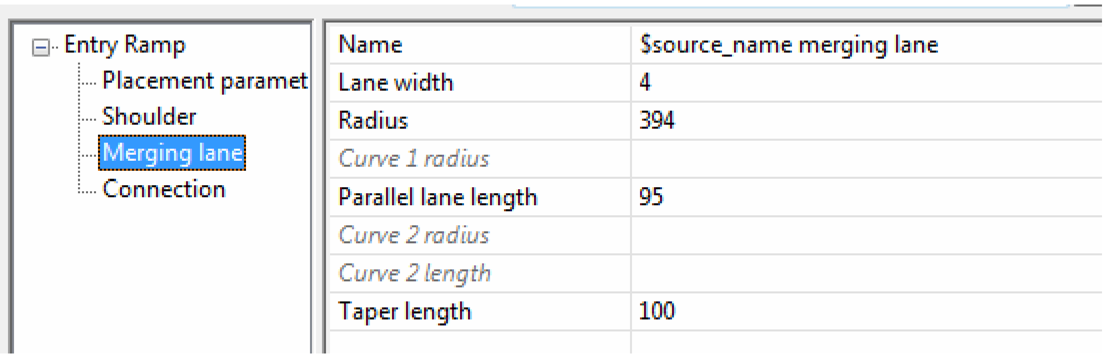
the offset of the island nose from the ramp itself

Nose width

the width of the nose

20.9.21 Merging Lane Node

This node defines the parameters for the **Merging lane**.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Name

the name of the merging lane string

Lane width

the width of the merging lane

Radius

the radius used to construct the merging lane

Curve 1 radius

the radius of the first optional curve, before the parallel lane length

Parallel lane length

the length of the parallel lane

Curve 2 radius

the radius of the second optional curve, after the parallel lane

Curve 2 length

the length of the second optional curve, after the parallel lane

Taper length

the length of the taper at the end of the merging lane

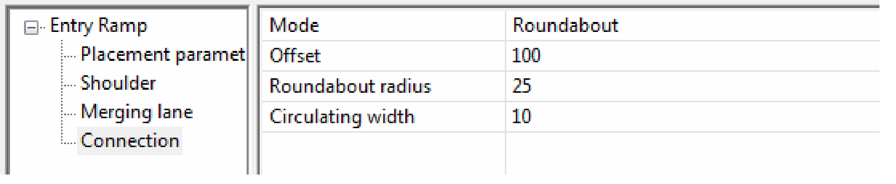
20.9.22 Connection

This node defines the properties for an optional connection from the overpass / minor road. At the moment, this supports either no connection strings or a roundabout.

Mode

the mode of the connection (either 20.9.22.1 Roundabout Mode *or* 20.9.22.2 No Strings Mode*)*

20.9.22.1 Roundabout Mode



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Offset

the offset from the intersection, along the minor road, that the connection should be created

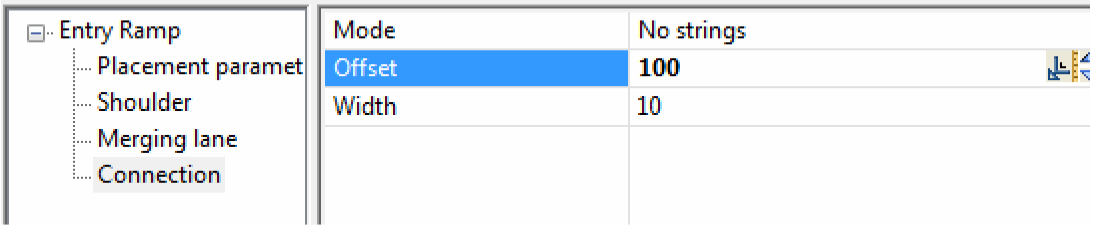
Roundabout radius

the radius of the roundabout

Circulating width

the circulating width of the roundabout

20.9.22.2 No Strings Mode



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Offset

the offset from the intersection, along the minor road, that the connection should be created

Width

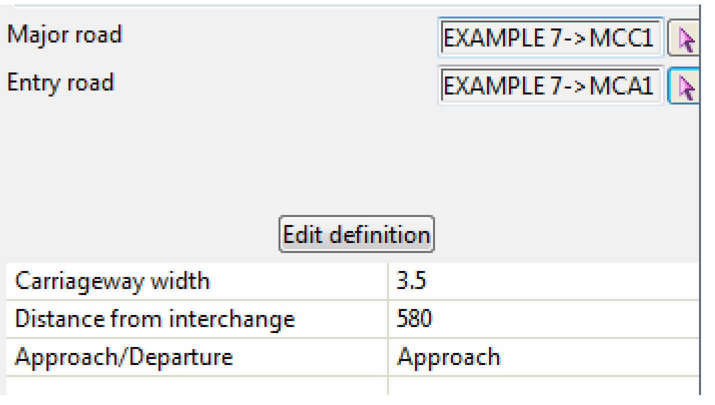
how far away from the minor road the entry ramp should start

20.9.22.2.1 Placing Entry Ramps

Entry Ramps are placed at the intersection of two roads - a major freeway and a minor adjoining or overpass road. When placing an **Entry Ramp**, there are several placement specific settings. This means they are separate from the definition but help define how the **Entry Ramp** is placed.

These are defined when placing the component, and may be edited from the 20.9.34.1 Placed Component Editor. Some of these are also available from the 20.9.34.2 Edit a Placed Component Definition panel.

These parameters are shown below, as they are displayed when placing the component



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Major road

the freeway or major road to enter

Entry road

the road to enter from

Edit definition button

Selecting this button displays the **Edit a Placed Component Definition** *panel. For more information please see* 20.9.34.2 Edit a Placed Component Definition

Carriageway width

the width of the freeway carriageway

Distance from interchange

the distance of the end of the entry ramp from the interchange

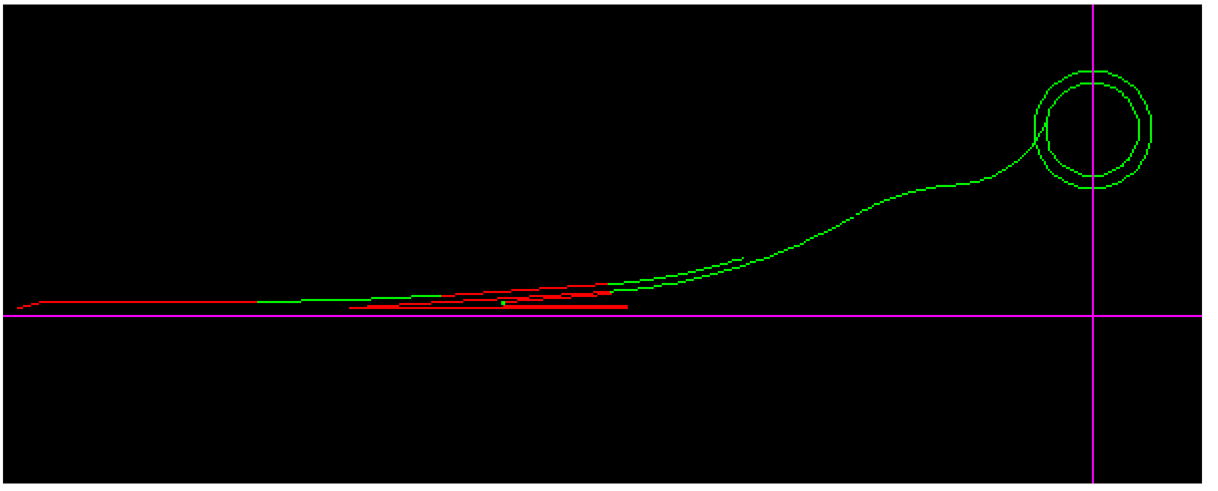
Approach/Departure

whether the entry ramp is bound to the approach or departure of the freeway / major road

20.9.22.3 Exit Ramp

An **Exit Ramp** component creates an **Exit Ramp** from a freeway or other major road to an overpass or other adjoining road. It can be edited via the 20.9.1 Component Library or by editing the definition of a placed component.

For more information on Defining an Exit Ramp see 20.9.22.3.1 Defining an Exit Ramp
Placing an Exit Ramp 20.9.27.1.1 Placing Exit Ramps



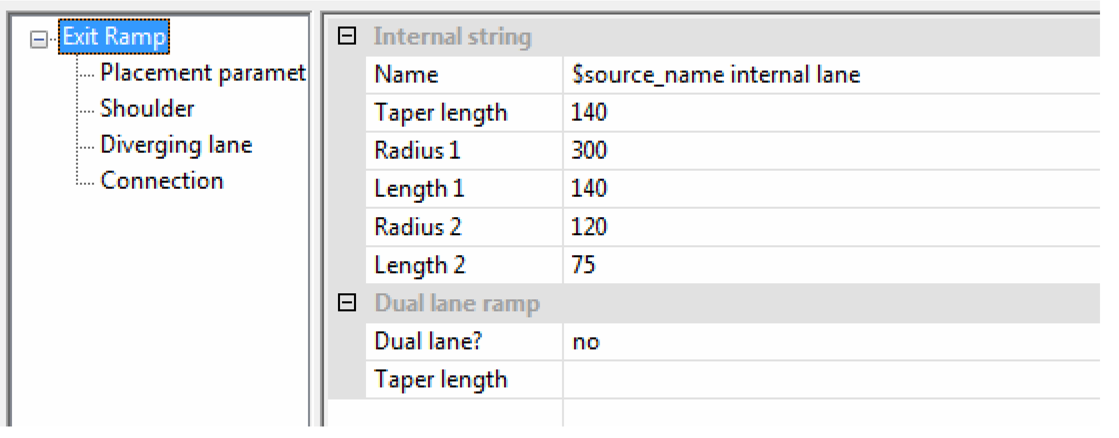
20.9.22.3.1 Defining an Exit Ramp

An **Exit Ramp** is defined by the ramp itself, a shoulder, a diverging lane and an optional connection.

For information on the Exit Ramp Node see 20.9.23 Exit Ramp Node
Placement Parameters Node 20.9.24 Placement Parameters
Shoulder Node 20.9.25 Shoulder Node
Diverging Lane Node 20.9.26 Diverging Lane Node
Connection Node 20.9.27 Connection Node

20.9.23 Exit Ramp Node

This node defines the ramp itself.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Internal String

Name

the name of the string to create

Taper length

the length of the ramp taper

Radius 1

the first radius of a reverse curve

Length 1

the first length of a reverse curve

Radius 2

the second radius of a reverse curve

Length 2

the second length of a reverse curve

Dual Lane Ramp

Dual lane

whether or not this is a dual lane exit ramp

Taper length

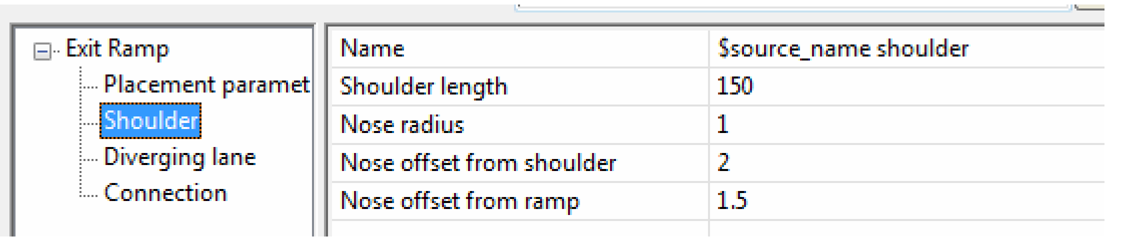
the taper length on the dual lane exit ramp

20.9.24 Placement Parameters

This node is only available when editing a placed component. For more information on the fields displayed here, see the section on 20.9.27.1.1 Placing Exit Ramps.

20.9.25 Shoulder Node

This node defines the shoulder part of the exit ramp.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Name

the name of the created shoulder string

Shoulder length

the length of the shoulder

Nose radius

the radius of the nose of the exit ramp island

Nose offset from shoulder

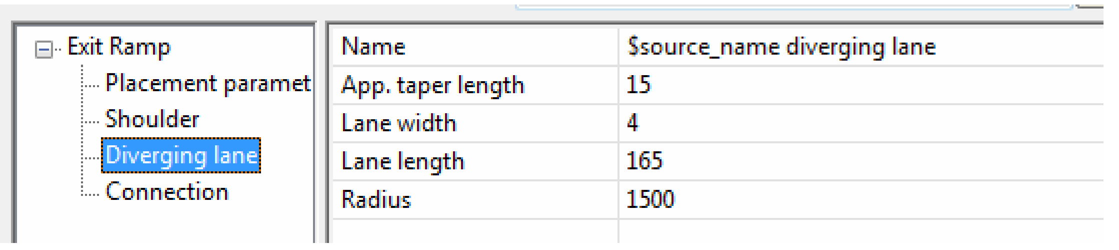
the offset of the island nose from the shoulder

Nose offset from ramp

the offset of the island nose from the ramp

20.9.26 Diverging Lane Node

This node defines the details for the diverging lane of the exit ramp.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Name

the name of the created string of the diverging lane

App. taper length

the length of approaching taper

Lane width

the width of the diverging lane

Lane length

the length of the diverging lane

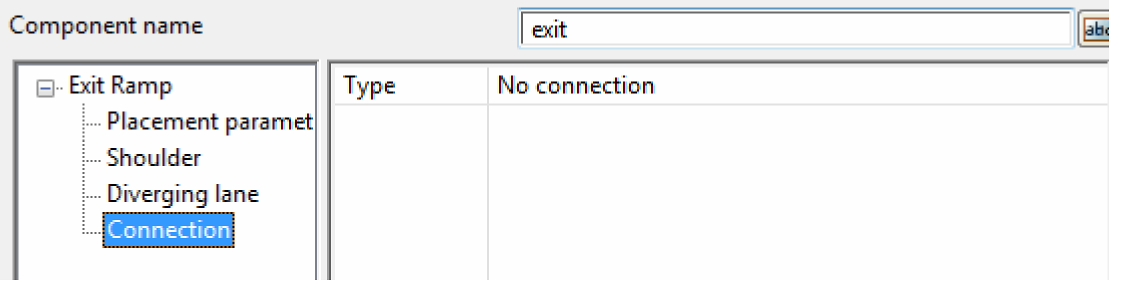
Radius

the radius for the diverging lane

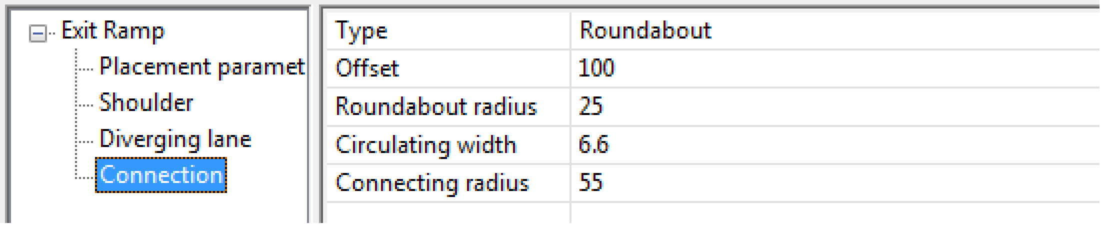
20.9.27 Connection Node

The connection node defines how the Exit Ramp connects to the adjoining road or overpass.

There are two types of connection currently; either no visible connection or a Roundabout (see 20.9.27.1 Roundabout Connection).



20.9.27.1 Roundabout Connection



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Offset

the offset of the roundabout from the major road

Roundabout radius

the radius of the centre of the roundabout

Circulating width

the circulating width around the roundabout

Connecting radius

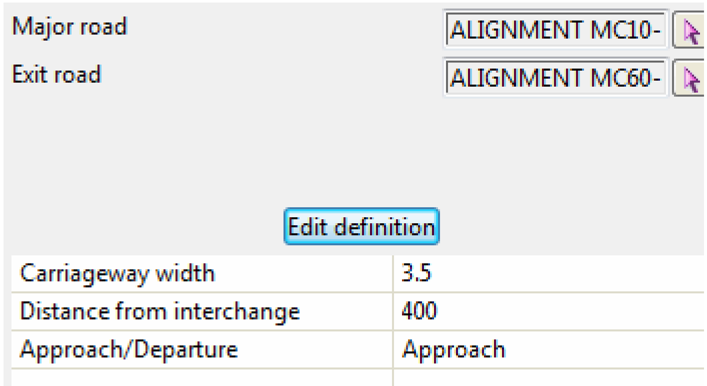
the radius of an arc connecting the exit ramp to the roundabout

20.9.27.1.1 Placing Exit Ramps

Exit Ramps are placed at the intersection of two roads - a major freeway and a minor adjoining or overpass road. When placing an **Exit Ramp**, there are several placement specific settings. This means they are separate from the definition but help define how the **Exit Ramp** is placed.

These are defined when placing the component, and may be edited from the 20.9.34.1 Placed Component Editor. Some of these are also available from the 20.9.34.2 Edit a Placed Component Definition panel.

These parameters are shown below, as they are displayed when placing the component.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Major road

the freeway or major road to exit from

Exit road

the road the ramp leads to

Carriageway width

the width of the major road

Distance from interchange

the distance from the interchange that the exit ramp should start

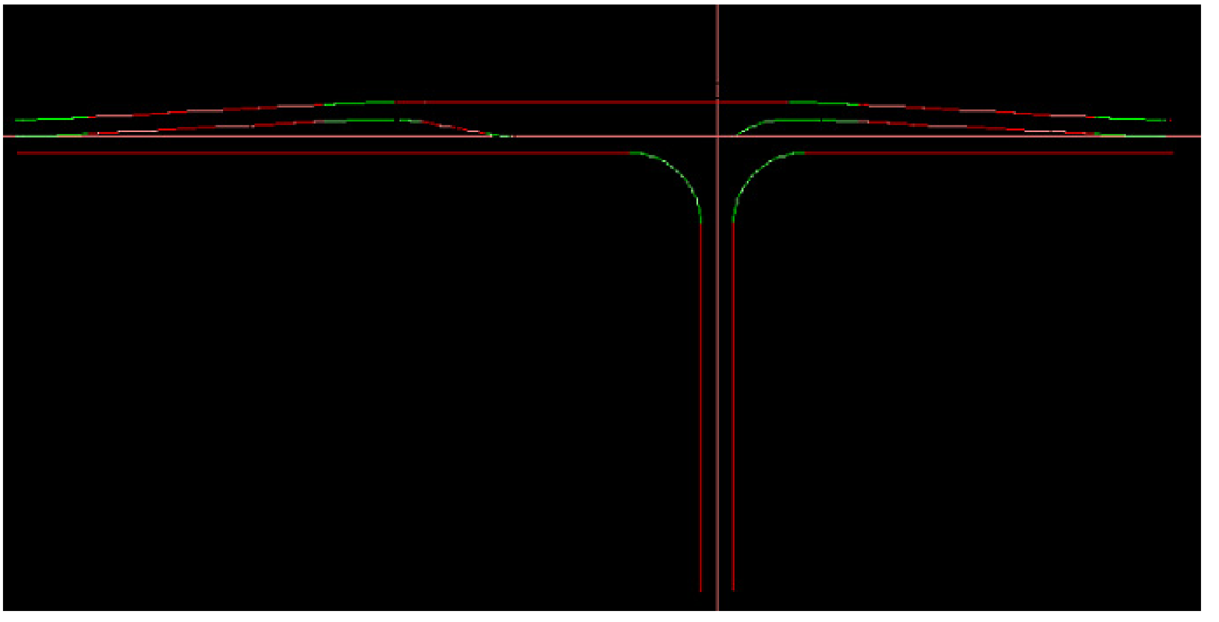
Approach / Departure

whether the exit ramp is bound to the approach or departure of the major road

20.9.27.2 CHR Intersection

A **CHR** or **Channelised Rural Intersection** creates an intersection between two roads, typically in a T-Junction configuration. It can be edited via the 20.9.1 Component Library or by editing the definition of a placed component.

For information on Predefined Values Node see 20.9.28 Predefined Values Node
Major Road Node 20.9.29 Major Road Node
Major Road => Kerb return Node 20.9.30 Major Road => Kerb Return Node
Minor Road Node 20.9.31 Minor Road Node
Placing CHR Intersections 20.9.31.3.1 Placing CHR Intersections

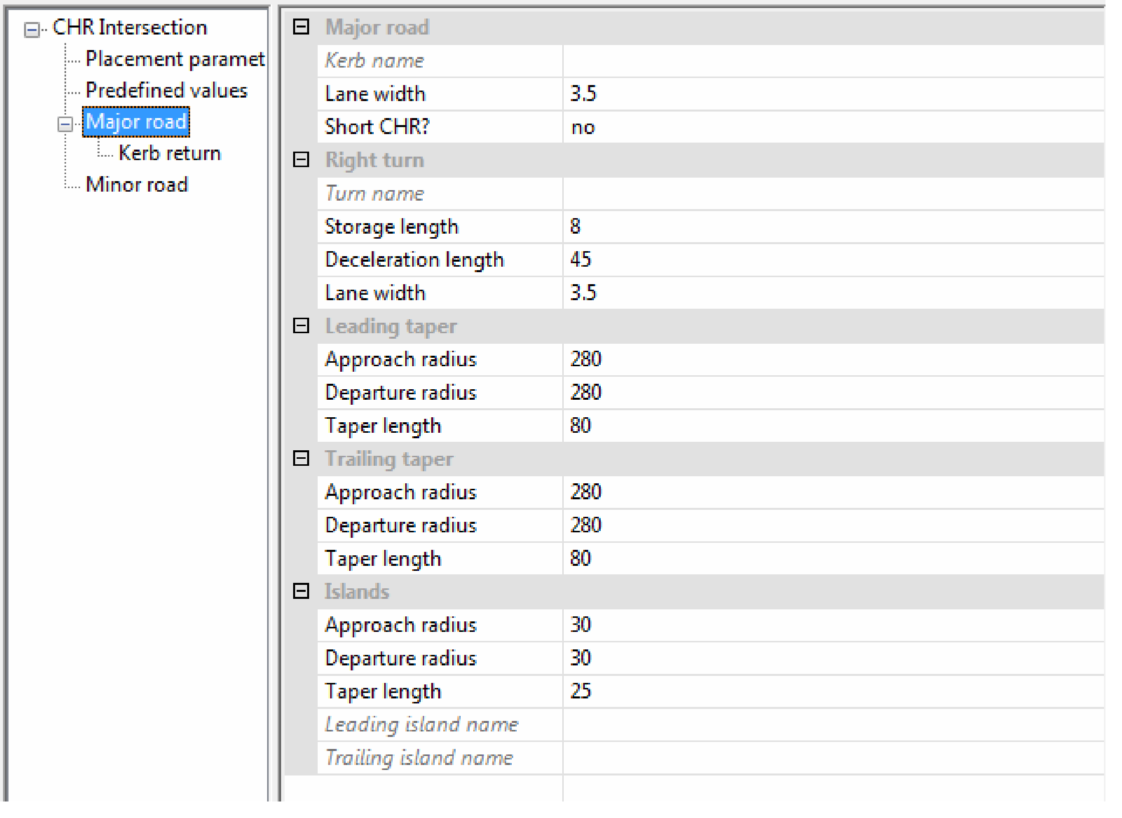


20.9.28 Predefined Values Node

The Predefined Values node is used to populate the details of the **CHR** based on one of a set of predefined speeds and a supplied lane width.

20.9.29 Major Road Node

This defines the details for the major road.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up

Major Road

Kerb name

the name of the kerb string to create

Lane width

the width of the carriageway

Short CHR?

whether or not to create a short CHR

Right Turn

Turn name

the name of the right turn string

Storage length

the storage length of the right turn

Deceleration length

the deceleration length of the right turn

Lane width

the width of the right turn lane

Leading Taper

Approach radius

the approach radius of the leading taper

Departure radius

the departure radius of the leading taper

Taper length

the length of the leading taper

Trailing Taper

Approach radius

the approach radius of the trailing taper

Departure radius

the departure radius of the trailing taper

Taper length

the length of the trailing taper

Islands

Approach radius

the approach radius of the islands

Departure radius

the departure radius of the islands

Taper length

the taper length for the islands

Leading island name

the name of the leading island of the CHR

Trailing island name

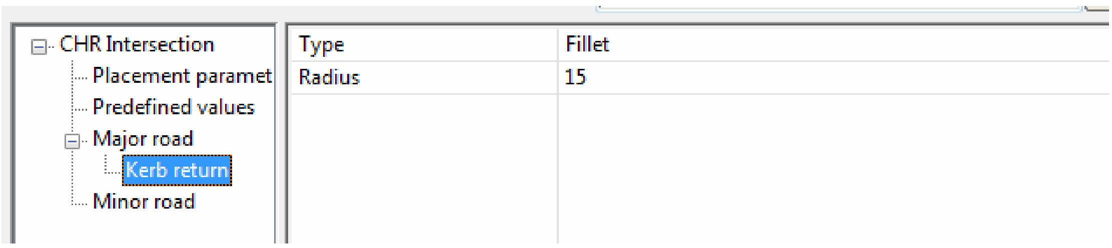
the name of the trailing island of the CHR

20.9.30 Major Road => Kerb Return Node

There are several different kerb return types available:

Fillet	20.9.30.1 Fillet Kerb Return
Two Centred	20.9.30.2 Two Centred Kerb Return
Three Centred	20.9.30.3 Three Centred Kerb Return

20.9.30.1 Fillet Kerb Return



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

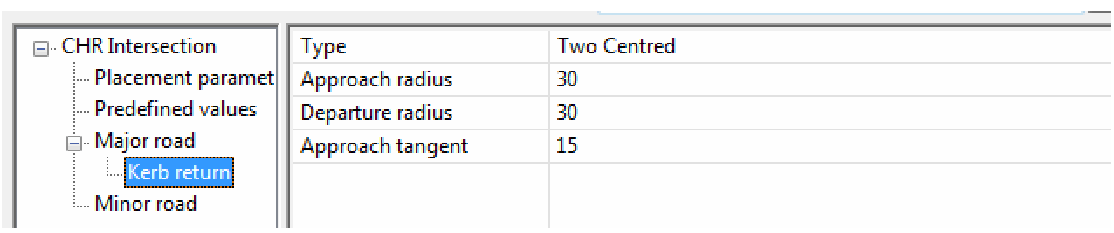
Type

*the type of kerb return to create (in this case, **Fillet**)*

Radius

the radius for the fillet kerb return

20.9.30.2 Two Centred Kerb Return



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Type

*the type of kerb return to create (in this case, **Two Centred**)*

Approach radius

the approach radius of the two centred curve

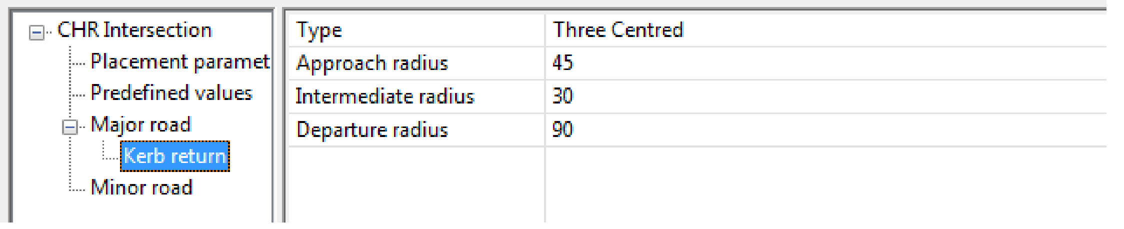
Departure radius

the departure radius of the two centred curve

Approach tangent

the approach tangent for the two centred curve

20.9.30.3 Three Centred Kerb Return



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Type

the type of kerb to create (in this case, ***Three Centred****)*

Approach radius

the approach radius of the three centred curve

Intermediate radius

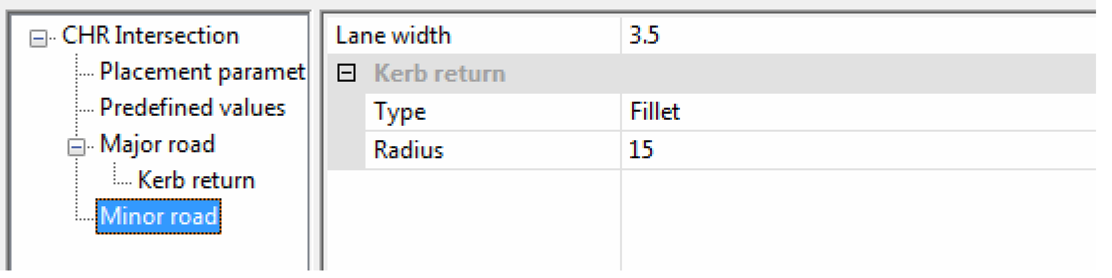
the intermediate radius of the three centred curve

Departure radius

the departure radius of the three centred curve

20.9.31 Minor Road Node

This node defines the details for the Minor road.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Lane width

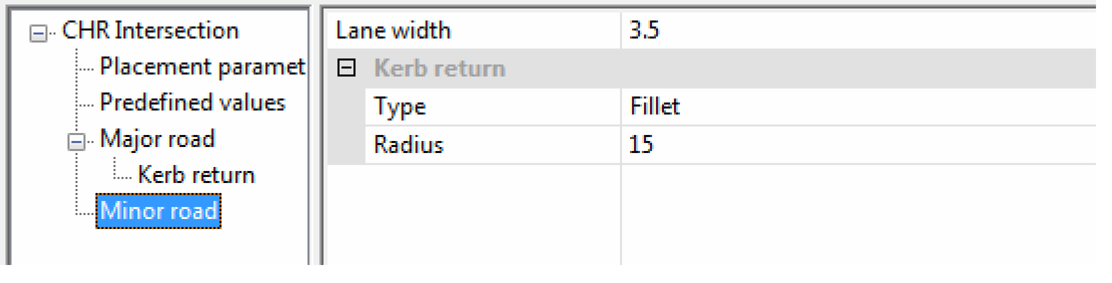
the width of the carriageway of the minor road

Kerb Return

There are several different kerb return types available:

Fillet	20.9.31.1 Fillet Kerb Return
Two Centred	20.9.31.2 Two Centred Kerb Return
Three Centred	20.9.31.3 Three Centred Kerb Return

20.9.31.1 Fillet Kerb Return



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

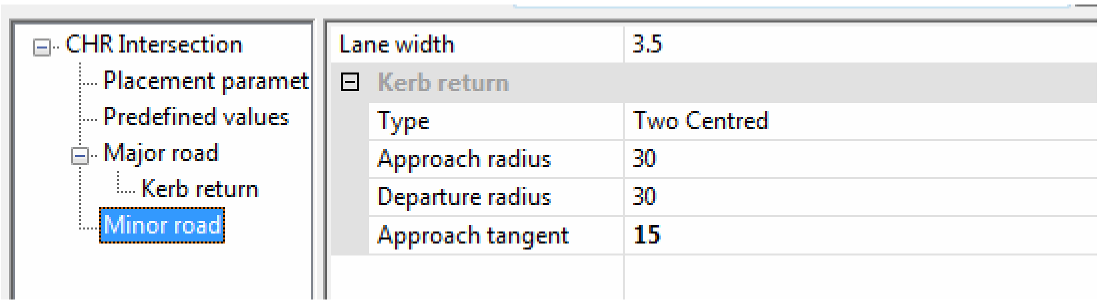
Type

*the type of kerb return to create (in this case, **Fillet**)*

Radius

the radius for the fillet kerb return

20.9.31.2 Two Centred Kerb Return



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Type

the type of kerb return to create (in this case, ***Two Centred****)*

Approach radius

the approach radius of the two centred curve

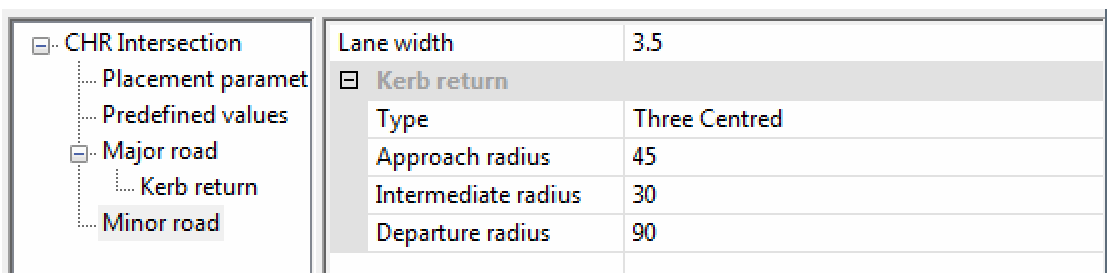
Departure radius

the departure radius of the two centred curve

Approach tangent

the approach tangent for the two centred curve

20.9.31.3 Three Centred Kerb Return



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Type

the type of kerb to create (in this case, ***Three Centred****)*

Approach radius

the approach radius of the three centred curve

Intermediate radius

the intermediate radius of the three centred curve

Departure radius

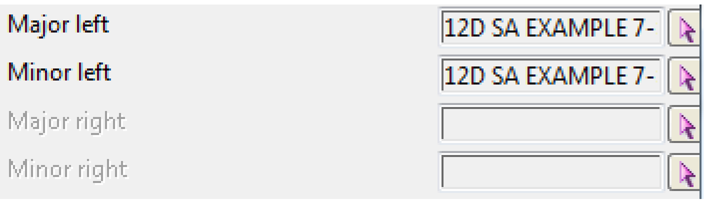
the departure radius of the three centred curve

20.9.31.3.1 Placing CHR Intersections

CHR Intersections are placed at the intersection of two roads - a major road and a minor road. Each road can be optionally defined with a second carriageway. When placing a CHR intersection, there are several placement specific settings. This means they are separate from the definition but help define how the CHR is placed.

These are defined when placing the component, and may be edited from the 20.9.34.1 Placed Component Editor Some of these are also available from the 20.9.34.2 Edit a Placed Component Definition panel.

These parameters are shown below, as they are displayed when placing the component.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Major left string select

the left most major road

Minor left string select

the left most minor road

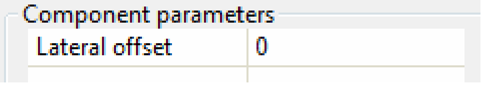
Major right string select

the right most major road (optional)

Minor right string select

the right most minor road (optional)

Major right and **Minor right** can be set when there is a dual carriageway.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Lateral offset

an offset of the CHR intersection from the major left control line. This is useful when the control line is not the centreline.

20.9.32 Place a Component

Position of option on menu: **Design =>Roads =>Components => Place component**

The **Place a component** panel allows you to place a component from your library.

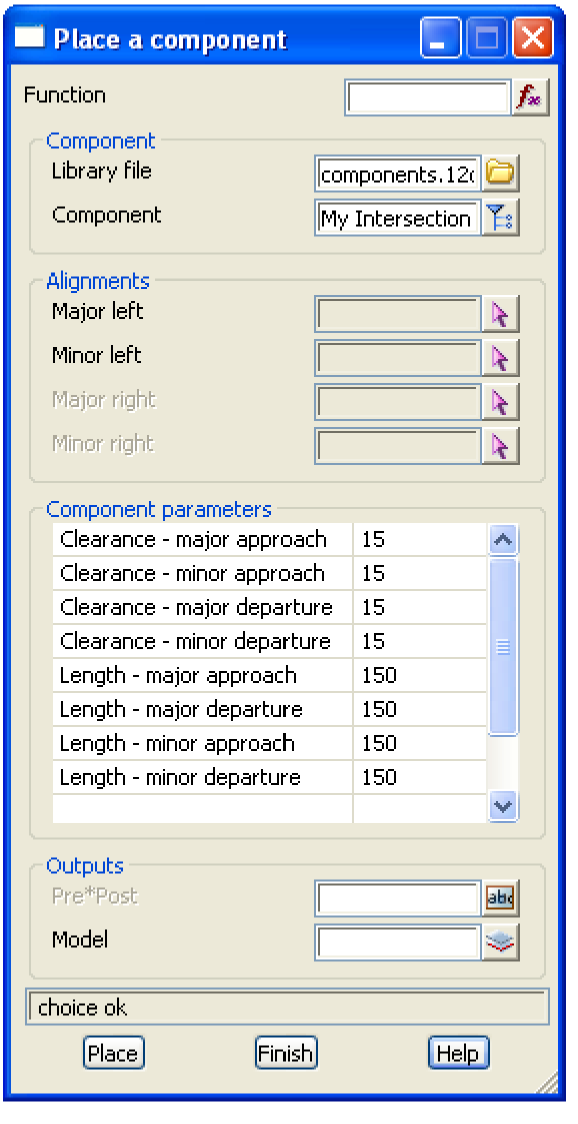
The placement specific parameters required will depend on which component you are attempting to place.

Each component is created as a function, which means you can easily recalc after you change the strings to which it is attached or make it part of a chain.

Please note that there are two types of parameters involved in defining a component:

1. **Component Specific** – meaning how the geometry is generated. These are edited via the library or by editing the definition of a placed component.
2. **Placement Specific** – how the component is placed. These are edited when placing a component on the **Place a component** panel or on **Editing a Placed Component.**

Selecting **Place component** brings up the **Place a component** panel.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up

Common Fields

Function function

the function to create

Library file file

the library file to read components from

Component choice box

the component to place

Pre*Post

*an optional pre*post to apply to generated strings*

Model model box

the model to create the component strings in

Place button

places the components

For more information on the placement specific parameters required for each component, see the section on the specific component type (see 20.9.2 Component Types).

20.9.32.1 Editing Components

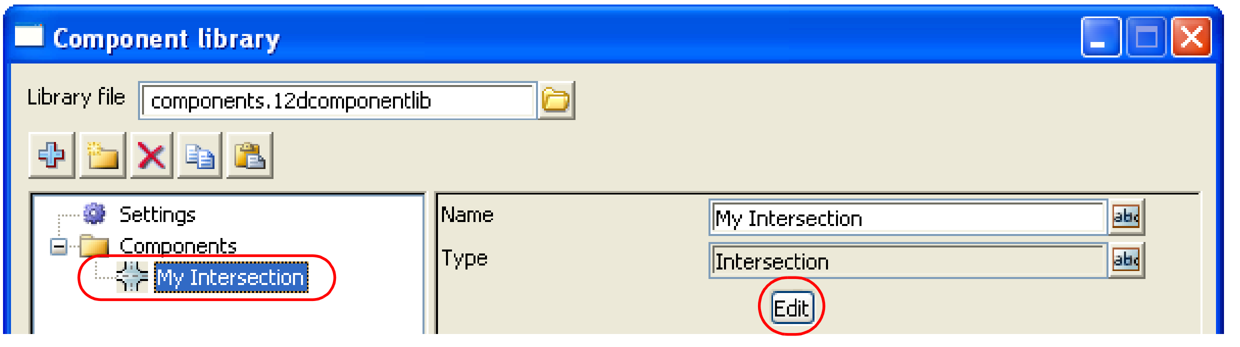
Components may be edited either in the library or directly from a placed component.

For information on **Editing in the Component library** go to 20.9.32.1.1 Editing in the Component library - Component Editor

For information on **Editing a Placed Component** go to 20.9.33.1 Editing a Placed Component

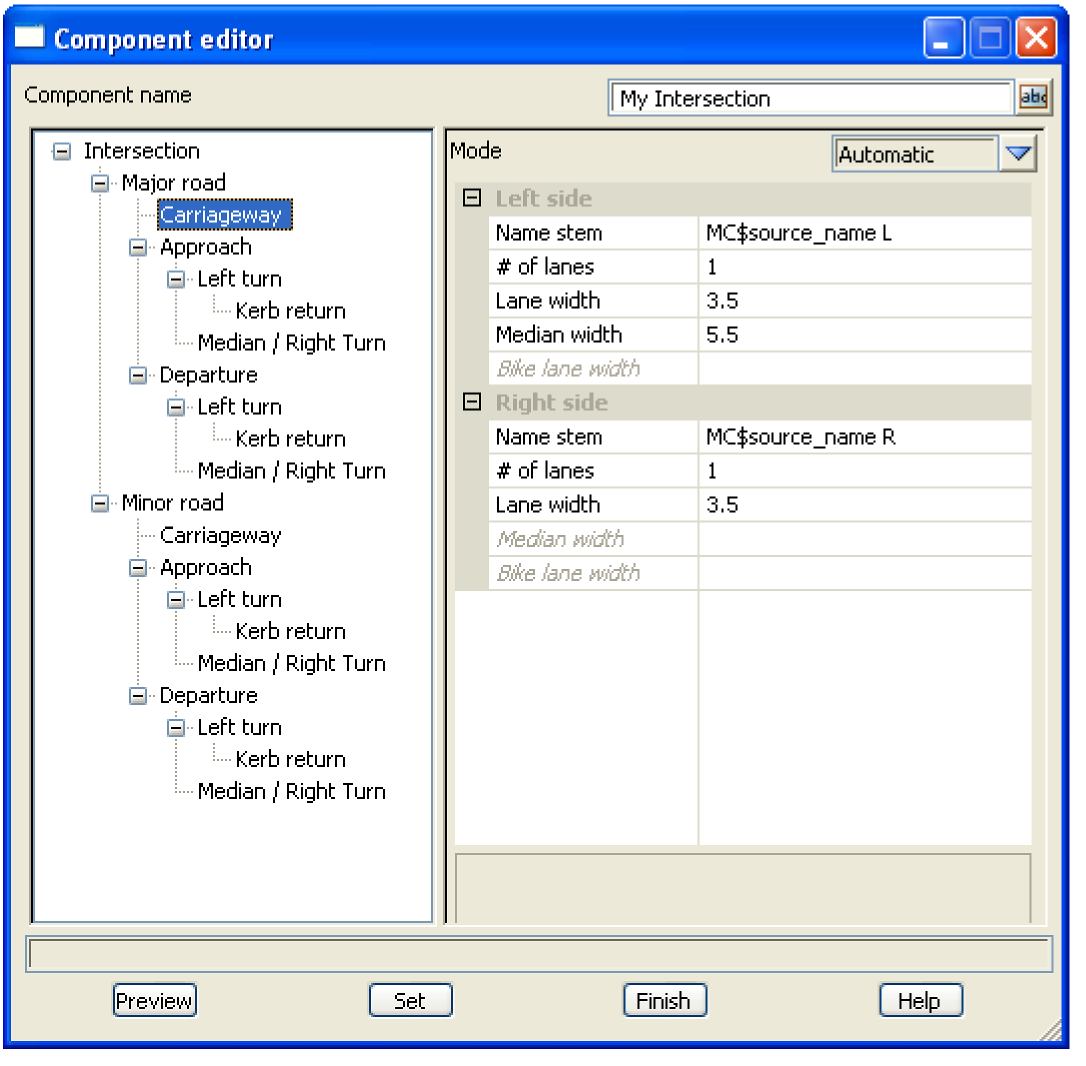
20.9.32.1.1 Editing in the Component library - Component Editor

Design =>Roads =>Components => Component library =>Component Node =>Edit button



To edit in the **Component Library**, select your component and then select the **edit** button, this brings up the **Component Editor panel.**

The **Component Editor panel** changes depending on the component, so for specific information on parameters, please see the section on that component type (see 20.9.2 Component Types). The following example shows editing an intersection component.



The fields and buttons used in this panel have the following functions:

Field Description   Type   Defaults   Pop-Up

Common fields and buttons

Component name

the name of the component

Preview button

shows a preview of the component with theoretical centreline strings. For more information please go to 20.9.33 Component Viewer.

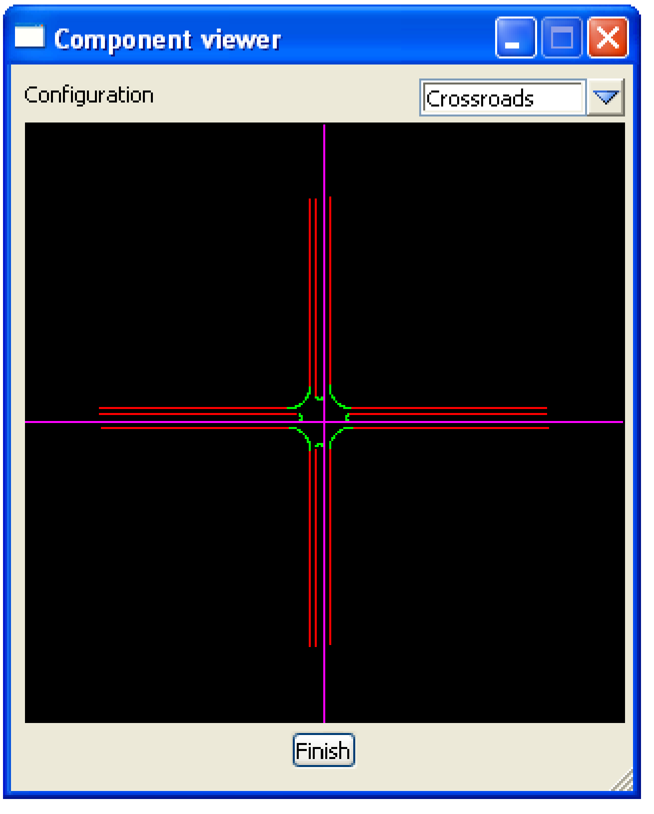
Set button

sets the component in the library

20.9.33 Component Viewer

The **Component viewer**, which is updated as you make changes to your library component, gives you an idea of what your component may look like when it is placed. The following example shows a standard intersection.

Selecting the **Preview** button on the **Component Editor** panel displays the **Component viewer** panel.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Configuration	choice box		Crossroads, T Junction

this field allows you to change what strings are generated to show you how the component will change based on different centrelines.

Options include: ***Crossroads*** *and* ***T-Junction***

Note that this does not affect the component in anyway – this is only for previewing purposes.

20.9.33.1 Editing a Placed Component

You may edit a placed component, either by changing the placement specific parameters or by changing the definition itself.

A placed component may be edited by using the **Functions** =>**Editor** (**Utilities** => **Functions** =>**Editor)** or by using the **Edit Component** option on the **Components** menu **(Design** =>**Roads** =>**Components** => **Edit Component).**

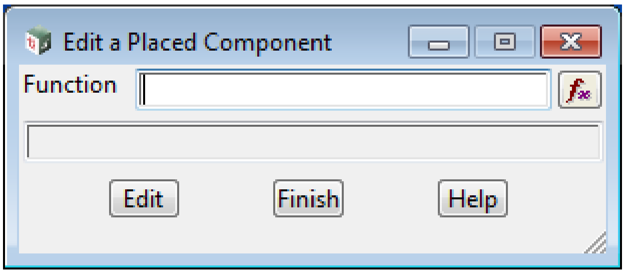
As when placing the component, the placement specific parameters will vary based on the type of component.

For more information on **Edit a Placed Component** please see 20.9.34 Edit a Placed Component

20.9.34 Edit a Placed Component

Position of menu: **Design =>Roads =>Components => Edit Component**

The **Edit a Placed Component** panel, edits an existing component by its function.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Function

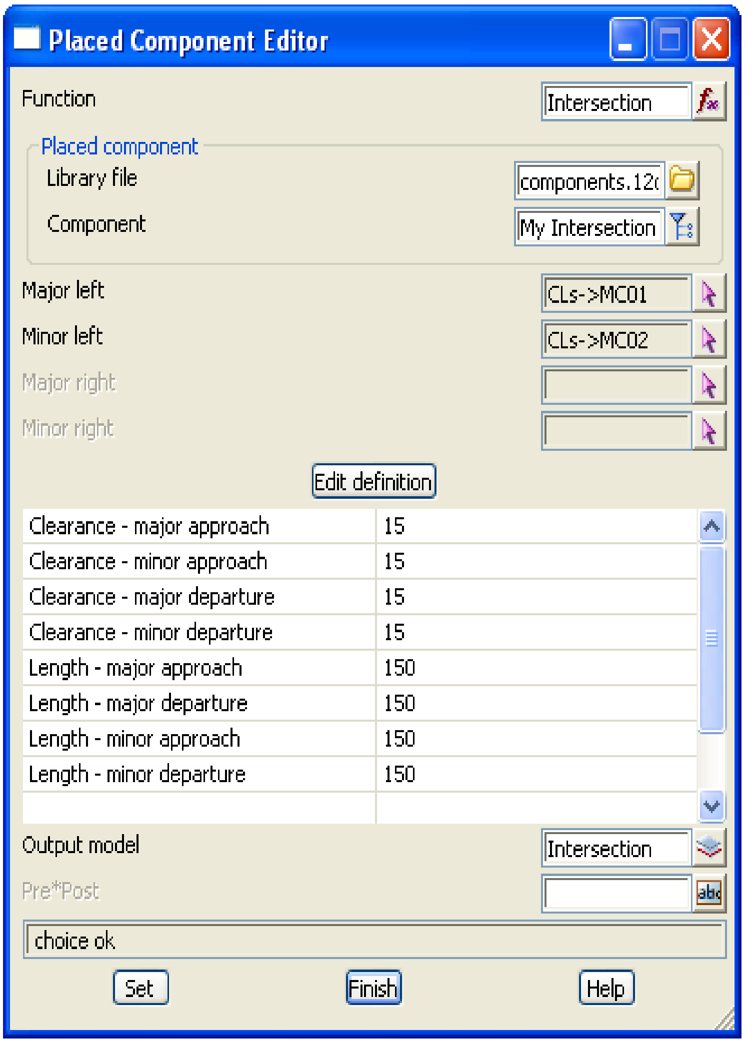
the component function

Edit button

edits the placed component (see 20.9.34.1 Placed Component Editor).

20.9.34.1 Placed Component Editor

This panel is only accessible by using the 20.9.34 Edit a Placed Component **(Design =>Roads =>Components => Edit Component)** or by editing the function directly from the **Utilities => Functions =>Editor.**



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Function	function		

the function you wish to edit

Placed Component

Library	file		

the library your component is read from

Component	choice box		

the component that is being placed

Please note that Library and Component will be read only if you have opted to detach the component from the library.

Edit Definition	button		

allows you to edit the definition of the placed component, which will detach it from the library. After you do this, any changes to the library component will not affect your component.For more information please see 20.9.34.2 Edit a Placed Component Definition

For information on the placement specific parameters, please see the section for that component (see 20.9.2 Component Types).

Output Model model box

the model the geometry will be generated into

Pre*Post

*an optional pre*post*

Set button

Sets the details

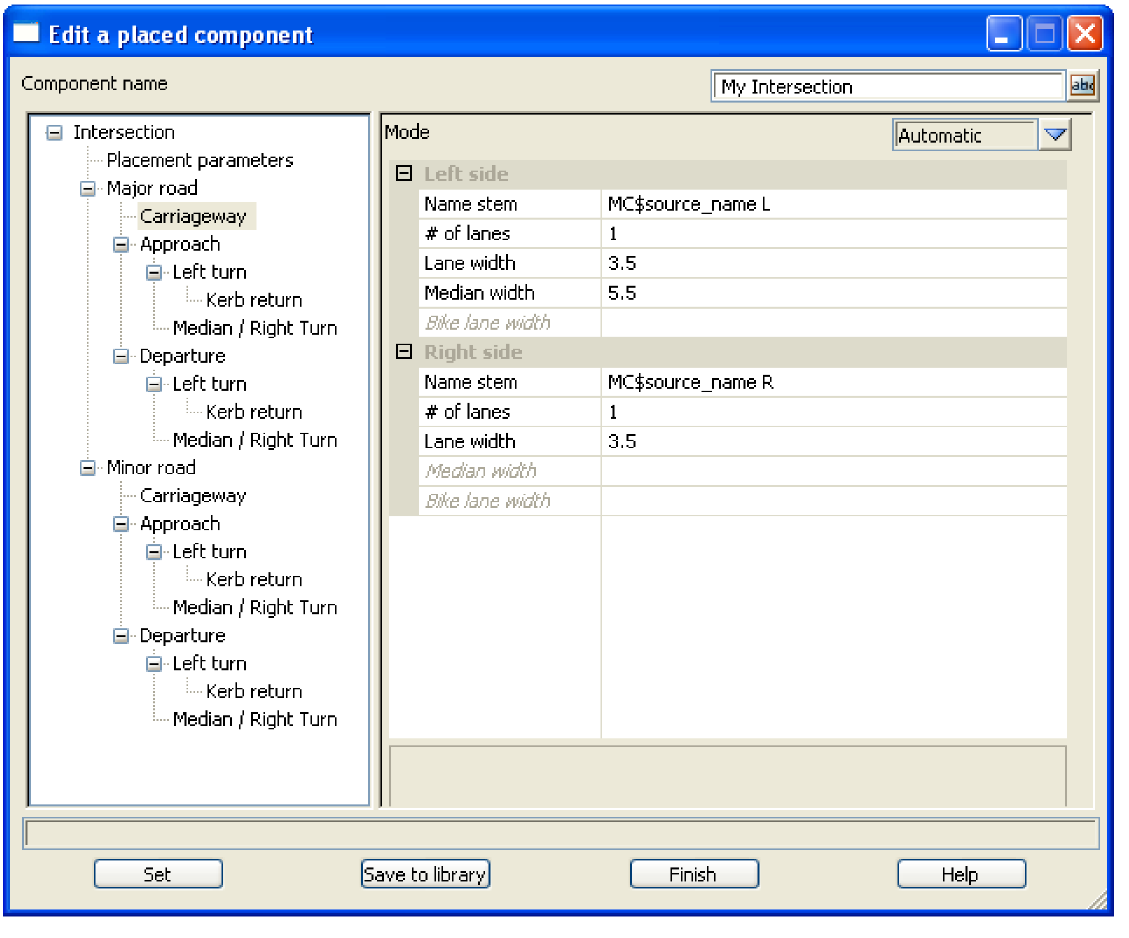
20.9.34.2 Edit a Placed Component Definition

To edit the definition of a placed component, you must edit the function (**Utilities** => **Functions** =>**Editor or Design** =>**Roads** =>**Components** => **Edit Component**) and then select '**Edit Definition**' on the 20.9.34.1 Placed Component Editor panel.

The following example shows the intersection editor. For details on the component specific parameters, see the section on that component type (see 20.9.2 Component Types).

As you edit the component, your changes will be reflected on the component strings.

If you do not wish to keep the changes, simply press '**Finish**' without pressing '**Set**'.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Component name

the name of the component

Set button

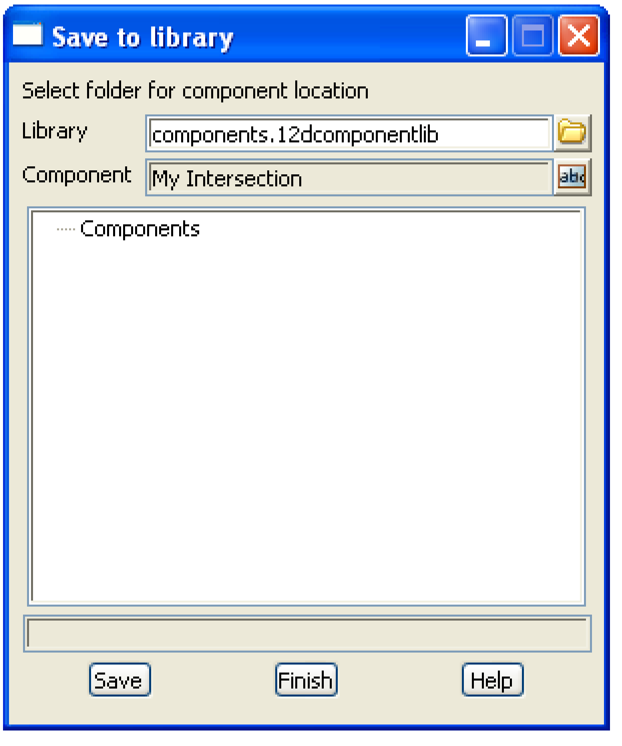
sets the component specific parameters

Save to library button

allows you to save your component definition back into your library. This is useful after you have made changes which you wish to reuse in other components.For more information please see 20.9.34.3 Save to Library

20.9.34.3 Save to Library

This panel allows you to save an edited definition back in to your library.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Library file

the library file to save to

Component

the name of the component you are saving

The panel displays a list of all the folders in your library. Select which folder you wish to save it into.

Save button

saves the component into your library

20.9.35 Component Quick Place

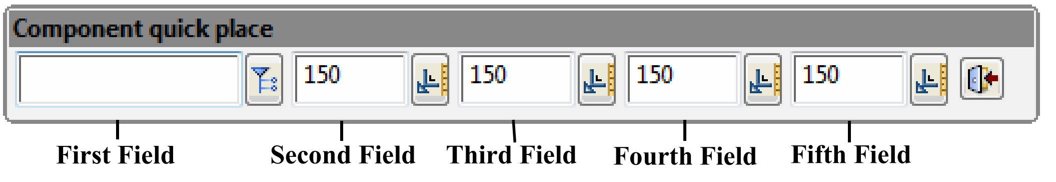
Position of option on menu: **Design =>Roads =>Components => Quick place**

The **Component quick place** toolbar is an easy way to quickly drop components on to existing strings.

Simply start the toolbar, pick your component and hover your mouse over where you would like the component created. If the placement point is correct, a preview of the component will be drawn on your view.

Once you have picked the position correctly, hitting **C** on your keyboard will create the component.

You can then press **E** to edit the component.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

First field

the list of components from your local library (searched for in the working directory, or in your library)

Second field

the length on the major approach (does not apply to all components)

Third field

the length on the major departure (does not apply to all components)

Fourth field

the length on the minor approach (does not apply to all components)

Fifth field

the length on the minor departure (does not apply to all components)

Last button

close the toolbar

20.10 Sight Lines

Position of menu: **Design =>Sight lines**
Utilities =>H-Z =>Sight lines

A **Sight Line** is a lines of sight emanate from an eye point and continuing until it is stopped by an object.

In ***12d Model*** Sight lines are stopped by the items:

(i) A Tin (tin or super tin)

(ii) Bridge tins

(iii) Super stings that are round pipes or rectangular pipes

(iv) Drainage strings

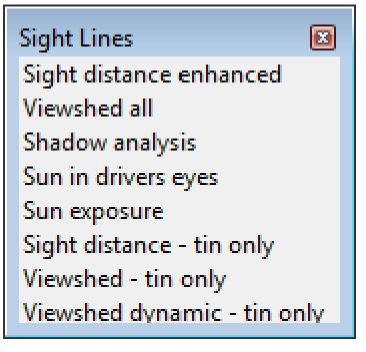
(v) Billboards

(vi) Extrusions down a super string

(vii) Trimeshes

(viii) Meshes

The sight line calculations are used in various forms in all the options on the **Sight Lines** menu.



See

20.10.2 Sight Distance Enhanced

20.10.3 Viewshed All

20.10.4 Shadow Analysis

20.10.5 Sun in Driver Eyes

20.10.6 Sun Exposure

20.10.7 Sight Distance - Tin Only

20.10.8 Viewshed - Tin Only

20.10.9 Dynamic Viewshed - Tin Only

20.10.1 Sight Lines

Position of option on menu: **Design =>Sight lines =>Sight distance enhanced**
Utilities =>H-Z =>Sight lines =>Sight distance enhanced

An **eye point** can see a t**arget point** if a straight line can be drawn from the eye point to the target point and the line is not stopped by any obstruction along the way.

If the target point **can** be seen from the eye point then it is **visible** to the eye point. If not, then it is **invisible** to the eye point.

Note that this is a reciprocal relationship - if **you** can **see** the **target** point **then** the **target** point can **see you**.

In ***12d Model*** the items that can stop an eye from seeing a target point are:

(a) A Tin (tin or super tin)

This is the ground surface. If in a plan view, the eye is on the tin then in z-value it must be above the tin.

For some options the eye can be off the Tin.

In some options, the tin can contain null regions.

(b) Bridge tins

A Bridge Tin is a tin where the underside of the tin is used in the visibility calculations.

These tins can be used to model some obstructions.

(c) Obstructions.

These are Objects in models that can stop a ray emanating from the eye point. Valid obstructions are

(i) Super stings that are round pipes or rectangular pipes

(ii) Drainage strings

(iii) Billboards

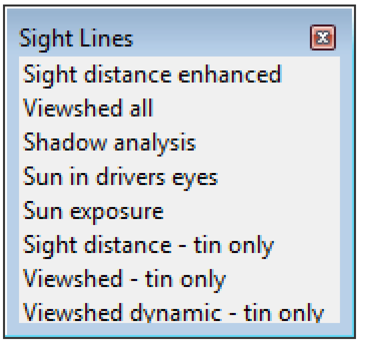
(iv) Extrusions

(v) Trimeshes

(vi) Meshes

A **Sight Line** is a line emanating from an eye point and continuing until it is stopped by one of the above items.

In ***12d Model***, visibility/invisibility/sight line calculations are used in various forms in all the options on the **Sight Line** menu.



Sight distance options - determining for each chainage down a road, how far ahead you can see.

Viewshed options - determining all the points that can be seen from a given eye position.

Shadow analysis - determining what shadows are created by objects.

Sun in drivers eyes - determining when the sun is in a drivers eyes.

Sun exposure - determining when the sun comes in a given window.

20.10.2 Sight Distance Enhanced

Position of option on menu: **Design =>Sight lines =>Sight distance enhanced**
Utilities =>H-Z =>Sight lines =>Sight distance enhanced

For a string (Super Alignment etc), the **sight distance enhanced** option calculates the maximum distance that can be seen from an eye point on the tin to a target point further along the string.

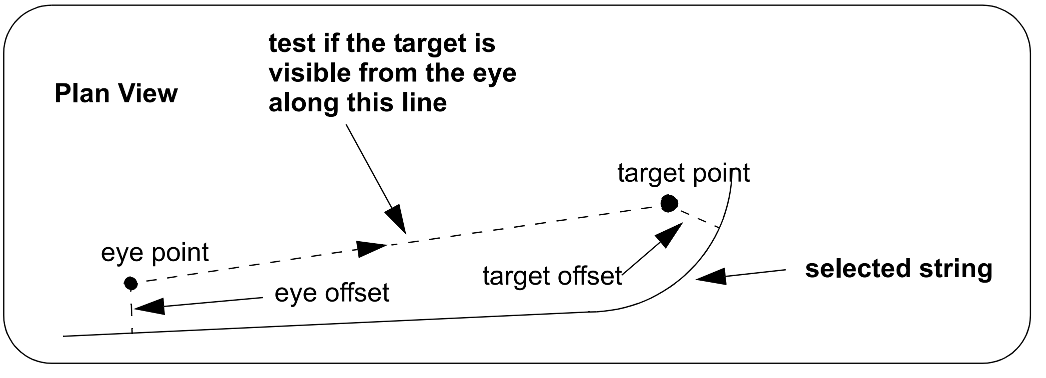
A string and (optionally) an eye point offset and height, and a target offset and height define the direction of the sight lines.

Tins, Bridge Tins and Obstructions are supplied for the sight line calculations.

To calculate the sight distances, ***12d Model*** positions the **eye point** at the given **start chainage** on the selected string, and then goes out perpendicularly for the **distance eye offset**. That position is dropped onto the tin, and finally raises it by the given **eye height.**

From the **eye point**, a target point is placed at a chainage distance equal to the given **minimum sight distance** along the string. The target point is offset by the given **target offset**, then dropped onto the tin and finally raised by the given **target height**.

The ***target point*** is then tested for visibility from the eye point, against the given tin. That is, a test is made to see if the *target point can be seen from the eye point*, against the surface given by the tin.



If the target point is visible, it is moved a given chainage distance **trial interval** further along the string and the visibility test redone.

The process is repeated until either the target point is invisible, or the target point is the given **maximum sight distance** (in chainage) away from the eye point.

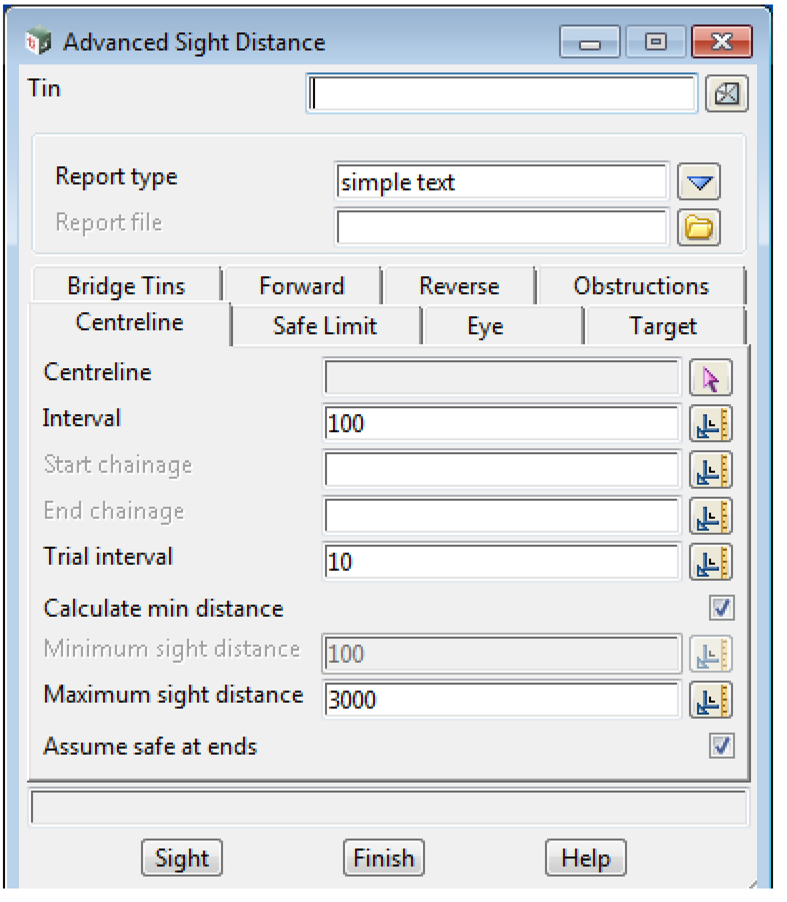
A line joining the eye and ***final target point position*** is created and placed in the **model for sight lines**. The ***chainage*** distance between the eye and the final target point is placed as a text string in the **model for sight text**, positioned at the eye point.

The eye point is then moved the given chainage distance **interval** along the string, and the process repeated until the eye point is past the given **end chainage**.

The eye point is then place at the **end chainage** and the negative of the **trial interval** and **interval** values used to calculate sight distance for the reverse direction along the string.

It is also possible to use a speed, reaction time and deceleration coefficient to calculate a minimum distance.

Selecting **Sight distance enhanced** brings up the **Advanced Sight Distance** panel



The fields and buttons used in this panel have the following functions.

Field Description — Type — Defaults — Pop-Up

Tin — tin box — — available tins

name of the tin to use as the surface for testing visibility.

Report type — choice box

type of report file to be produced (only if **Report file** *has a file name in it)*

Report file — file box — — depends on Report type

*if **not blank**, a report of the type selected in* **Report type** *is produced and written out to this file name.*

Centreline tab

Centreline — string-select

the string to be used for calculating the chainage position for the eye and target points, is selected from a view.

Interval — real box — 100 — measures pop-up

once the sight distance is calculated for the eye at a chainage, the eye chainage is incremented by this amount the sight distance calculation repeated.

Start/end chainage — real box

the sight distance is calculated for points on the selected string covering the chainage range given by the start and end chainage fields. If the **start/end chainage** *is blank, the start/end chainage of the selected string is used.*

Trial interval real box 10 measures pop-up

if the target point is visible, it is then moved along by this chainage increment and the sight test repeated.

There are also values under the **Safe Limit** tab that can be used in the sight distance calculations instead of the fixed **Minimum sight distance.**

If **Calculate min distance** is ticked, then the values on the **Safe Limit** tab can be entered and are used to calculate a minimum sight distance.

Calculate min distance tick box

*if ticked then the minimum sight distance is calculated from the values in the **Safe Limit** tab which allows for a* **Speed**, **Reaction time** *and a* **Deceleration coefficient**.

If not ticked then the constant **Minimum sight distance** *is used.*

Minimum sight distance real box 100 measures pop-up

If **Calculate min distance** *is **not ticked**, then this is the minimum chainage distance from the eye point to place the test target point. The first test of a target is made at this minimum chainage distance from the eye chainage.*

Maximum sight distance real box 3000 measures pop-up

maximum chainage distance to use for placing the test target point. The testing stops if the test target position goes over this chainage distance. In this case, the sight distance will be the **maximum sight distance**.

Safe Limit tab

Speed (km/hr) real box 60 measures pop-up

speed that the vehicle is travelling.

Reaction time (s) real box 2 measures pop-up

time in seconds for the driver to react.

Deceleration coefficient (m/s2) real box 0.36 measures pop-up

the Deceleration coefficient in metres per second per second.

Eye tab

the eye position is determined by finding the chainage along the selected string, going out perpendicularly for the given **eye offset** *(negative to the left, positive to the right), dropping that position onto the given tin, and then adding the* **eye height** *to the height on the tin. Hence the eye point is always the eye height above the tin.*

Height real box 1.3

height of the eye point above the given tin.

Offset real box -0.5

offset of the eye point from the picked string.

Target tab

the target position is determined by finding the chainage along the selected string, going out perpendicularly for the given **target offset** *(negative to the left, positive to the right), dropping that position onto the given tin, and then adding the* **target height** *to the height on the tin. Hence the target point is always the target height above the tin.*

Height real box 0.3

height of the target point above the given tin

Offset real box -0.5

offset of the target point from the picked string.

Bridge Tins tab

*A **Bridge Tin** is a tin where the underside of the tin is used in the visibility calculations.*

Tins grid of tins available tins

the underside of these tins are used in the visibility calculations. These tins can be used to model some obstructions.

Forward tab

Do calcs tick box

if ticked, do the calculations for the forward direction. That is, from the start of the string and in increasing string chainage.

Sight lines model box available models

*if **not blank**, the name of the model to contain the sight lines.*
*If **blank**, the sight lines are not created.*

Clean models first tick box not ticked

if ticked, the models of results are cleaned out before the option runs.

Good colour colour box default colour available colours

when the minimum sight distance is achieved, sight lines are created in the good colour

Bad colour colour box default colour available colours

when the minimum sight distance is not achieved, sight lines are created in the Bad colour

Sight text model box available models

if non-blank, the name of the model to contain the sight distance text.
If blank, the sight distance text is not created.

Sight text info text info

drawing information for the sight text.

Reverse tab

Do calcs tick box

if ticked, do the calcs for the reverse direction. That is, from the end of the sting and going in reverse string chainage direction.

Sight lines model box available models

*if **not blank**, the name of the model to contain the sight lines when going down the string in the reverse direction Note that the eye and target offsets are then applied in the reverse direction of the string.*
*If **blank**, the sight lines are not created.*

Clean models first tick box not ticked *if ticked, the models of results are cleaned out before the option runs.*

Good colour colour box default colour available colours

when the minimum sight distance is achieved, sight lines are created in the good colour

Bad colour colour box default colour available colours

when the minimum sight distance is not achieved, sight lines are created in the Bad colour

Sight text model box available models

if non-blank, the name of the model to contain the sight distance text.
If blank, the sight distance text is not created.

Sight text info text info

definition of the sight text.

Obstructions tab

all the 3d objects in the data source that can be used as obstructions, are used. For a list of valid obstructions, see 20.10.1 Sight Lines.

Data source type Model

data selection type - XX

Data source for obstructions

source of data to look through for elements to use as obstructions.

Sight button button

clicking this button runs the operation.

In the report, the definition for safe distance is

reaction_distance = ***reaction_time*** * ***speed_value*** /3.6

safe_distance = ***reaction_distance*** + ***speed_value*** * ***speed_value*** /(254
*(***deceleration_coefficient*** + ***grade**** ***change_deceleration_per_grade***)

where

reaction_time is in seconds

speed_value is the Operating speed in km/h

deceleration_coefficient is the coefficient of deceleration (longitudinal friction factor)

grade is longitudinal grade (a percentage), with sign + for upgrades and - for downgrades.

change_deceleration_per_grade is

Note: 3.6 is the conversion from *km/h* to *m/s*

20.10.3 Viewshed All

Position of option on menu: **Design =>Sight lines =>Viewshed all**
Utilities =>H-Z =>Sight lines =>Viewshed all

This panel is used to calculates the points, on or above a tin, that are visible and invisible from a user selected **eye** position. That is, it is calculating the sight lines from an eye point.

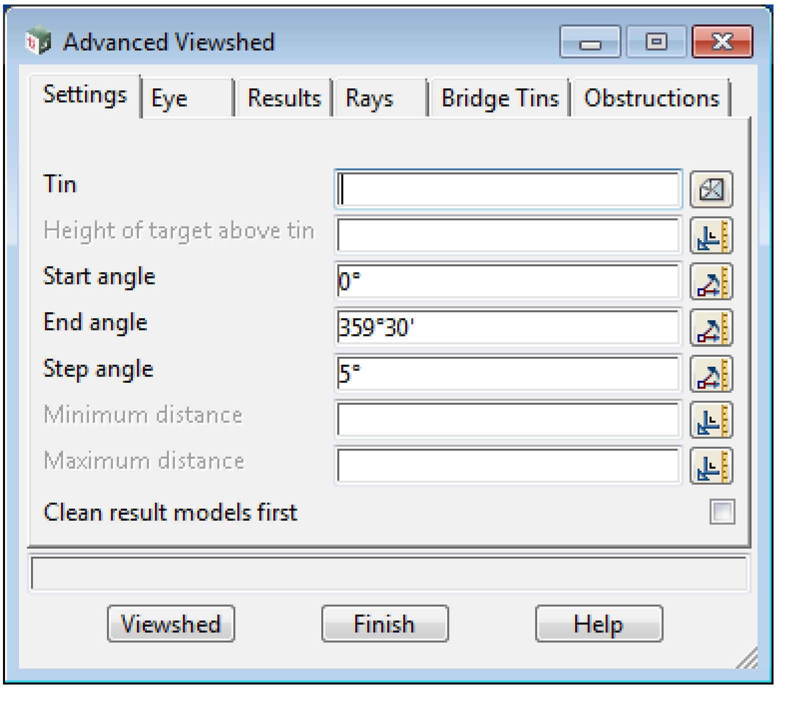
This is equivalent to calculating the points that can, or can not, see a selected **eye** point.

Bridge Tins and Obstructions are used in the sight line calculations. For a list of valid obstructions, see 20.10.1 Sight Lines

The calculations are made along rays emanating from the eye point from a minimum plan distance to a maximum plan distance from the eye point. The rays are created at regular angular steps from a start angle to an end angle.

The eye point can be off the tin and there can be null triangles in the tin.

Selecting **Viewshed all** displays the **Advanced Viewshed** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Settings tab

Tin	tin box		available tins

name of the tin to be used for the viewshed analysis.

Height of target above tin	real box		

the distance that the target is above the tin. This can be positive or negative.

Start angle	angle box	0	

the angle to begin taking sight rays emanating from the eye point. Measured in a counter clockwise direction from the positive x-axis.

End angle angle box 359.30

the angle to stop taking sight rays emanating from the eye point. Measured in a counter clockwise direction from the positive x-axis.

Step angle angle box 5

the angle between successive sight rays emanating from the eye point.

Minimum distance real box

*if **not blank**, the minimum distance from the eye point along the sight ray to begin recording visible/ invisible points.*
*If **blank**, the minimum distance is zero.*

Maximum distance real box

the maximum distance from the eye point along the sight ray to record visible/invisible points.
If blank, the entire tin is considered.

Clean models first tick box

*if **ticked**, the models of results are cleaned out before the option runs.*

Eye tab

Eye XYZ input/output **xyz ops** menu

X/Y/Z coordinate

the XYZ co-ordinates of the eye point.

Each co-ordinate can be individual entered or the point pick icon used to select an existing point.

***Note** - the z-value must be above the surface of the tin at the (x,y) eye position minus the target height. So the eye can not be below the tin by more than the target height, or even the target height below the tin.*

Results tab

More tick box

*if **ticked**, ask for name, model, linestyle, weight.*
*If **not ticked**, only ask for model.*

Name name box names in names.4d

the name for the visible/invisible sight strings.

Model model box available models

the model for the visible/invisible sight strings.

Linestyle linestyle box available linestyles

the linestyle for the visible/invisible sight strings.

Weight weight box

thickness of the string when plotted

Colour of visible bits colour box default line colour

*if **not blank**, the colour for the visible parts of the rays.*
*If **blank**, the visible parts are not created.*

Colour of invisible bits colour box default point colour

*if **not blank**, the colour for the invisible parts of the rays.*

*If **blank**, the invisible parts are not created.*

Separate strings tick box

if ticked, the visible and invisible sections are separate strings.

If not ticked, the visible and invisible sections are combined into one super string.

Rays tab

Visible Rays

*the **visible** rays are the lines from the **eye** point to the **point on the tin** where the **target goes from being visible to invisible**.*

If **Colour of visible bits** *is not blank then this is the colour used for these rays.*

If **Colour of visible bits** *is blank then the colour is the default line colour.*

The ray will be coloured where it is over invisible targets with the colour for the invisible bits (this can happen in undulating country).

More tick box

*if **ticked**, ask for name, model, linestyle, weight.*
*If **not ticked**, only ask for model.*

Name name box names in names.4d

the name for the visible ray strings.

Model model box available models

*if **not blank**, the model for the visible ray strings.*
*If **blank**, don't create the visible ray strings*

Linestyle linestyle box available linestyles

the linestyle for the visible ray strings.

Weight weight box

thickness of the visible ray strings when plotted

Invisible Rays

*the **invisible** rays are the lines from the **eye point** to the **point on the tin** where the **target goes from being invisible to visible**.*

If **Colour of invisible bits** *is not blank then this is the colour used for these rays.*

If **Colour of invisible bits** *is blank then the colour is the default point colour.*

The ray comes from the eye and is coloured with the **Colour of visible bits** *to the change over from visible to invisible and then coloured with the* **Colour of invisible bits** *to where the ray ends (the change over visible to invisible).*

More tick box

*if **ticked**, ask for name, model, linestyle, weight.*
*If **not ticked,** only ask for model.*

Name name box names in names.4d

the name for the invisible ray strings.

Model model box available models

*if **not blank**, the model for the invisible ray strings.*
*If **blank**, don't create the invisible ray strings*

Linestyle linestyle box available linestyles

the linestyle for the invisible ray strings.

Weight weight box

thickness of the invisible ray strings when plotted

Bridge Tins tab

*A **Bridge Tin** is a tin where the underside of the tin is used in the visibility calculations.*

Tins grid of tins available tins

the underside of these tins are used in the visibility calculations. These tins can be used to model some obstructions.

Obstructions tab

the following elements can also obstruct the view line.

(i) super strings with round or rectangular sections.
(ii) bill boards
(iii) extrusions along a string

Data source type Model

data selection type - XX

Data source for obstructions

source of data to look through for elements to use as obstructions.

Viewshed button

On selecting this button, the lines of sight emanating from the **eye XYZ** *point are calculated.*

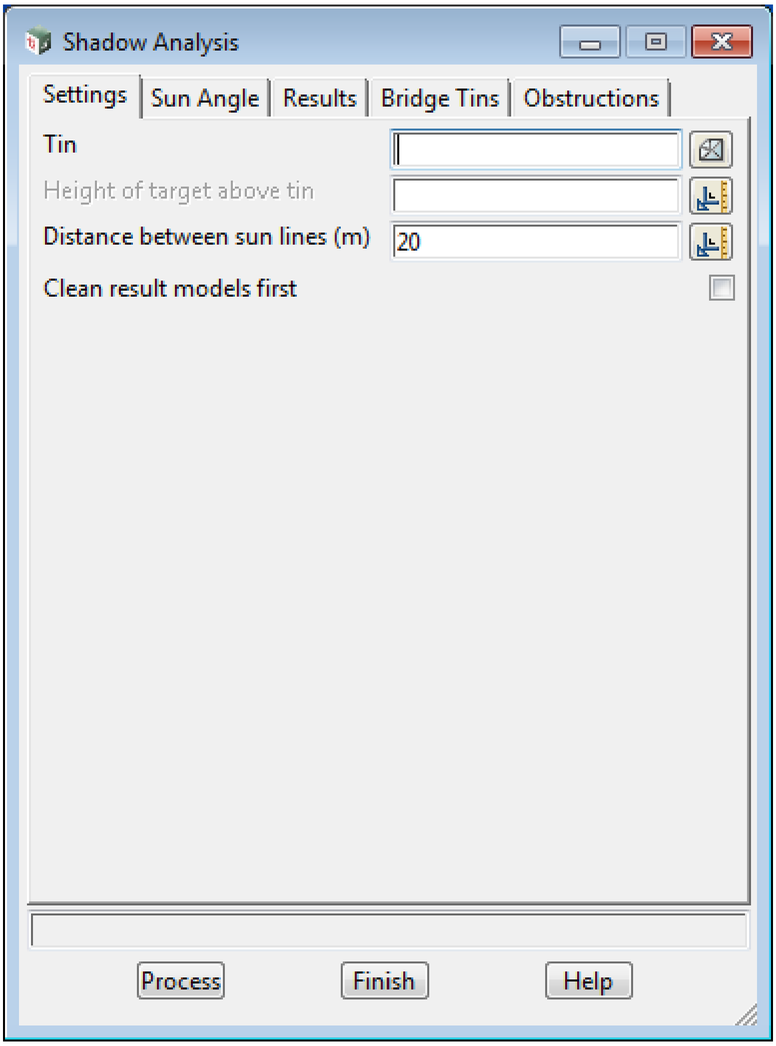
20.10.4 Shadow Analysis

Position of option on menu: **Design =>Sight lines =>Shadow analysis**
Utilities =>H-Z =>Sight lines =>Shadow analysis

This option is under development.

The **Shadow Analysis** option is similar to the *Viewshed* option except that the **sun** is used as the **eye** position. This means that the **visible** areas are those in **sunlight** and the **invisible** areas are those in the **shade** (in **shadows**).

Selecting **Shadow analysis** displays the **Shadow Analysis** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Settings tab

Tin tin box available tins

name of the tin to be used for the shadow analysis.

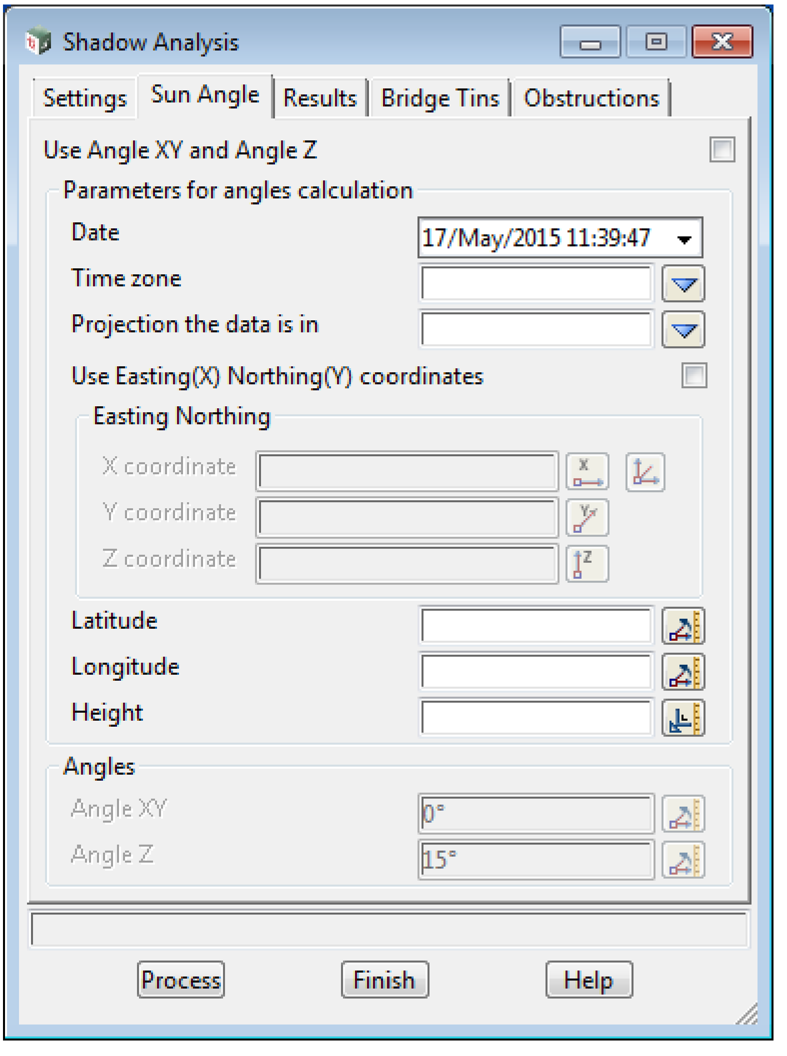
Distance between sun lines real box 20 measures pop up

the perpendicular distance between the result lines.

Clean models first tick box not ticked

if ticked, the models of results are cleaned out before the option runs.

Sun Angle tab



Use Angle XY and Angle Z tick box not ticked

*this controls whether Angle XY and Angle Z are typed in and used to specify the position of he sun, **or** if the sun position (and hence Angle XY and Angle Z) are calculated by giving the date and time of day, and either the (Easting, Northing) or the longitude and latitude of a point in the middle of the data.*

*If **ticked**, the user specifies the position of the sun with respect to the data by giving* **Angle XY** *and* **Angle Z** *(see below for their definitions). This method if used when the data is in an arbitrary coordinate system so there is no way to calculate the position of the sun from the data.*

*If **not ticked**, then the data needs to be in a Projection and the angle to the sun is calculated by giving the date and time, and either the (x,y) coordinate, or the longitude and latitude and of a point in the data.*

So if **Use Angle XY and Angle Z for sun position** *is **not** ticked:*

Date

the sun position is calculated for this date and time.

Time zone choice box

used to define the time zone for the time and date.

Projection the data is in projection box

the data to do Shadow Analysis on must be in this projection.

Use Easting (X) Northing (Y) coordinates tick box not ticked

*if **ticked**, the (x,y) coordinates of a point in the middle of the data is given. Using the Projection, the longitude and latitude can be calculated for this point. **Note** - because the data is in a projection, then the (x,y) coordinates are often referred to as Eastings and Northings.*

*If **not ticked**, the latitude and longitude of a point in the middle of the data is given. Using the Projection, the (x,y) position of the point can be calculated.*

If **Use Easting (X) Northing (Y) coordinates** is **ticked:**

X/Y/Z coordinate real box measure box

the x,y and z coordinates for a typical point in the data can be typed in or a point can be selected using he Point select and the values for x, y an dz will be written to these fields.

The z coordinate will also be written to the **Height** *field.*

If **Use Easting (X) Northing (Y) coordinates** is **not ticked:**

Longitude/Latitude angle box measure box

the longitude and latitude for a typical point in the data is typed in.

Height measure box

height of a typical point on the ground.

If **Use Easting (X) Northing (Y) coordinates** is **ticked:**

Angle XY angle box 0

the angle of the sun in the XY plane measured in a counter clockwise direction from the positive x axis. It is dms in HP notation.

Angle Z angle box 15

the angle of the sun in the vertical plane, measured from the horizontal plane with positive angle going up (Elevation angle?). It is dms in HP notation.

Results tab

More tick box

if ticked, ask for name, model, linestyle, weight.
If not ticked, only ask for model.

Name name box names in names.4d

the name for the sunshine (visible)/shade (invisible) sight strings.

Model model box available models

the model for the sunshine (visible)/shade (invisible) sight strings.

Linestyle linestyle box available linestyles

the linestyle for the sunshine (visible)/shade (invisible) sight strings.

Weight weight box

thickness of the sunshine (visible)/shade (invisible) sight strings when plotted

Colour of bits in sunshine colour box default line colour

*if **not blank**, the colour for the parts in sunshine. These are the visible parts of the view lines.*

*If **blank**, the sunshine parts are not created.*

Colour of bits in shade bits colour box default point colour

*if **not blank**, the colour for the parts in the shade (in shadows). These are is the invisible parts of the view lines.*

*If **blank**, the shade parts are not created.*

Separate strings tick box

*if **ticked**, the sunshine (visible) and shade (invisible) sections are separate strings.*

*If **not ticked**, the sunshine (visible) and shade (invisible) sections are combined into one multicoloured string.*

Bridge Tins tab

*A **Bridge Tin** is a tin where the underside of the tin is used in the visibility calculations.*

Tins grid of tins available tins

the underside of these tins are used in the visibility calculations. These tins can be used to model some obstructions.

Obstructions tab

the following elements can also obstruct the line from the sun.

(i) super strings with round or rectangular sections.

(ii) bill boards

(iii) extrusions along a string

Data source type Model

data selection type - XX

Data source for obstructions

source of data to look through for elements to use as obstructions.

Process **button**

On selecting this button, the lines of sight emanating from the sun are calculated.

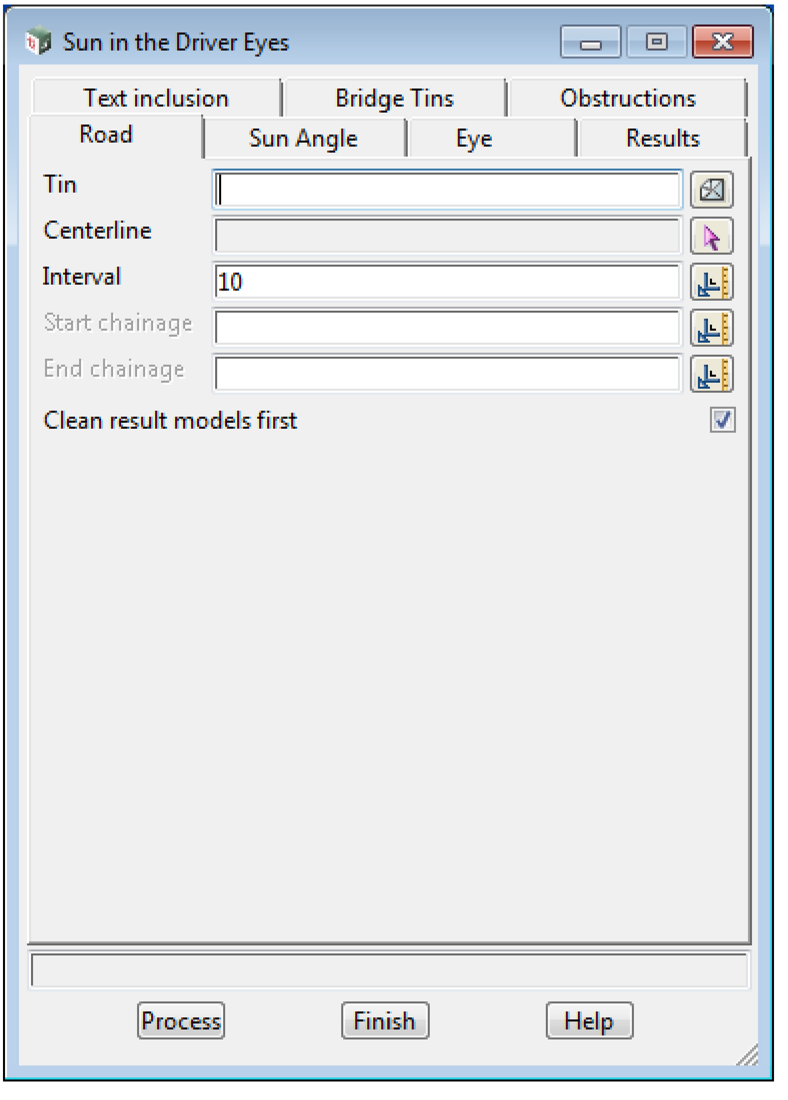
<Esc> can be used to terminate the option during the processing.

20.10.5 Sun in Driver Eyes

Position of option on menu: **Design =>Sight lines =>Sun in driver eyes**
Utilities =>H-Z =>Sight lines => Sun in driver eyes

This option is under development.

Selecting **Sun in driver eyes** displays the **Sun in the Driver Eyes** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Road tab			
Tin	tin box		available tins

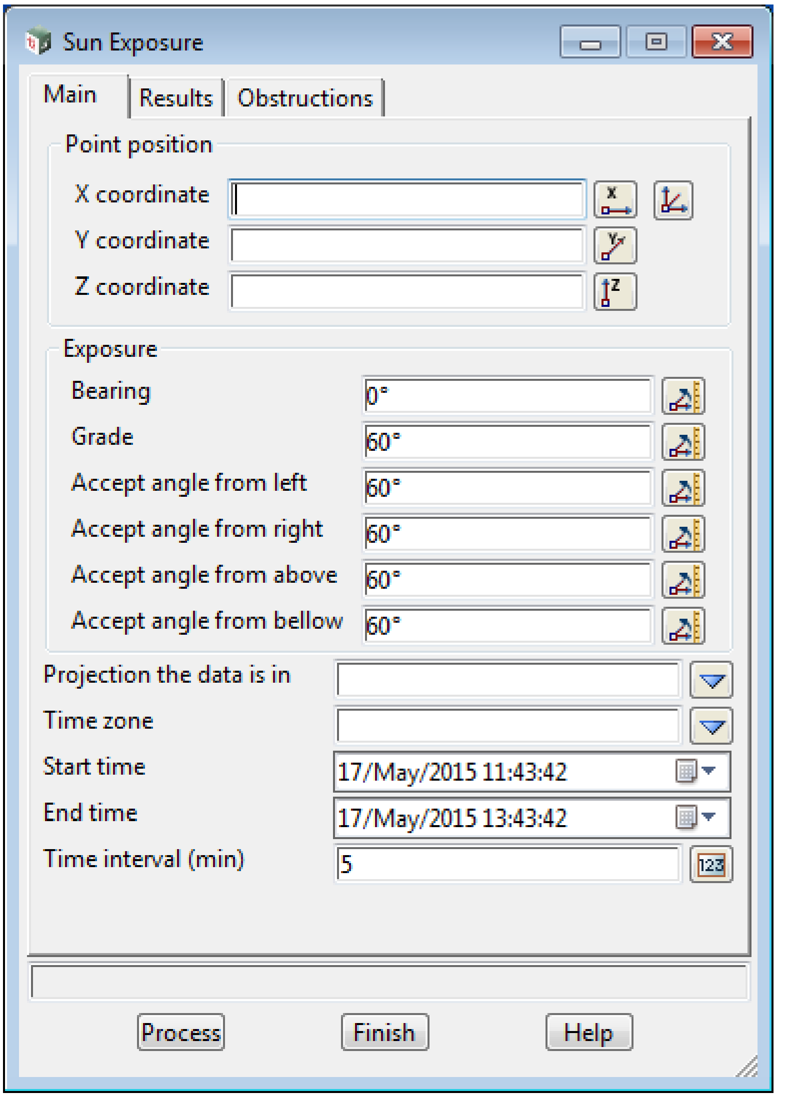
name of the tin to be used for the analysis.

20.10.6 Sun Exposure

Position of option on menu: **Design =>Sight lines =>Sun exposure**
Utilities =>H-Z =>Sight lines => Sun exposure

This option is under development.

Selecting **Sun exposure** displays the **Sun Exposure** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Main tab

Point position

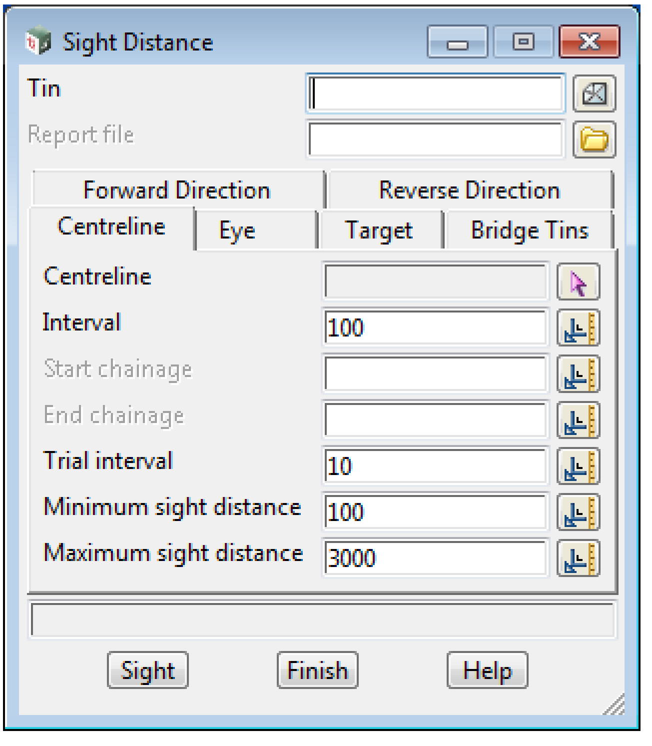
20.10.7 Sight Distance - Tin Only

Position of option on menu: **Design =>Sight lines =>Sight distance - tin only**
Utilities =>H-Z =>Sight lines =>Sight distance - tin only

For a string (Super Alignment etc), the **sight distance** option calculates the maximum distance along the road that can be seen from an eye point.

The calculations are done in the same manner as for the **Advanced Sight Distance** except that there is only a Tin and Bridge Tins. There are no Obstructions (see 20.10.2 Sight Distance Enhanced).

Selecting **Sight distance - tin only** brings up the **Sight Distance** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	input		available tins

name of the tin to use as the surface for testing visibility.

Report file	file box		*.rpt files

if non blank, a report is produced and written out to this file name.

Centreline tab

Centreline	string-select		

the string to be used for calculating the chainage position for the eye and target points, is selected from a view.

Interval	input	100	

once the sight distance is calculated for the eye at a chainage, the eye chainage is incremented by this amount the sight distance calculation repeated.

Start/end chainage	input		

the sight distance is calculated for points on the selected string covering the chainage range given by the start and end chainage fields. If the **start/end chainage** *is blank, the star/end chainage of the selected string is used.*

Trial interval input 10

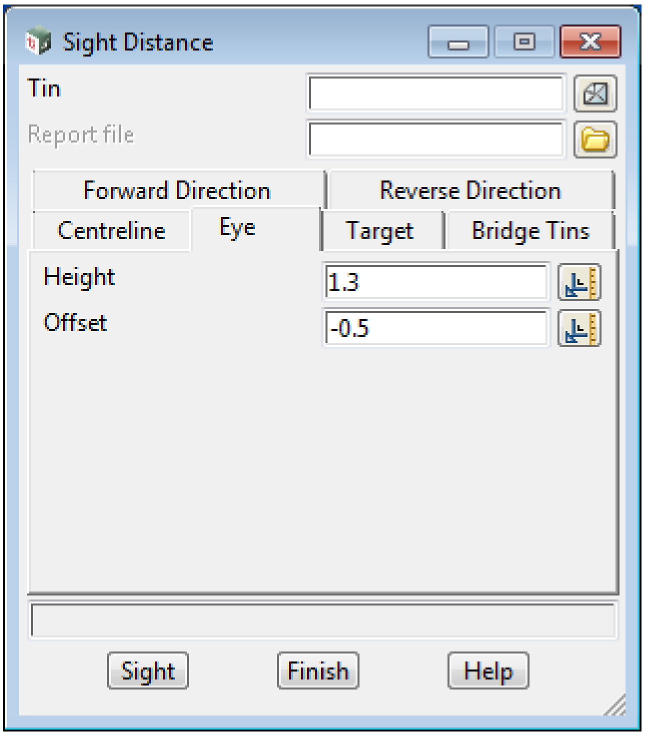
if the target point is visible, it is then moved along by this chainage increment and the sight test repeated.

Minimum sight distance input 100

minimum chainage distance from the eye point to place the test target point. The first test of a target is made at this minimum chainage distance from the eye chainage.

Maximum sight distance input 3000

maximum chainage distance to use for placing the test target point. The testing stops if the test target position goes over this chainage distance. In this case, the sight distance will be the **maximum sight distance**.



Eye tab

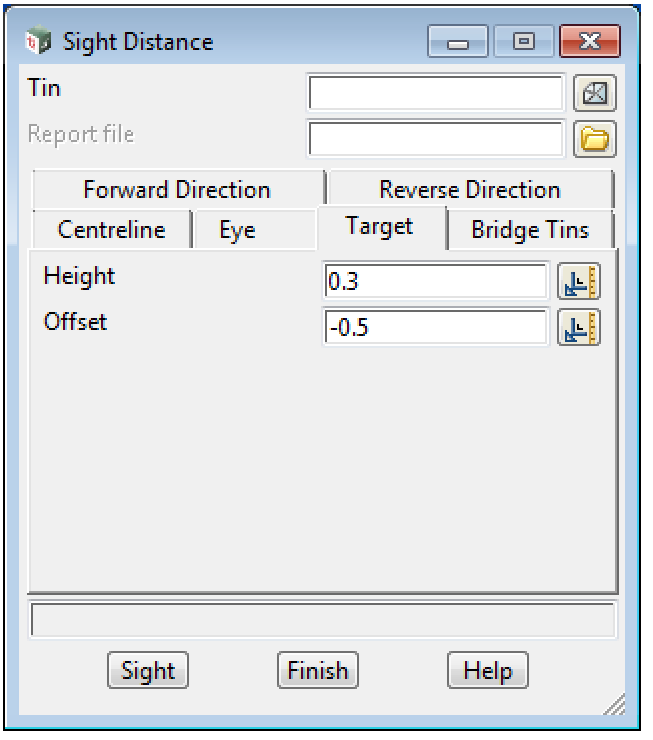
the eye position is determined by finding the chainage along the selected string, going out perpendicularly for the given **eye offset** *(negative to the left, positive to the right), dropping that position onto the given tin, and then adding the* **eye height** *to the height on the tin. Hence the eye point is always the eye height above the tin.*

Height input 1.3

height of the eye point above the given tin.

Offset input -0.5

offset of the eye point from the picked string.



Target tab

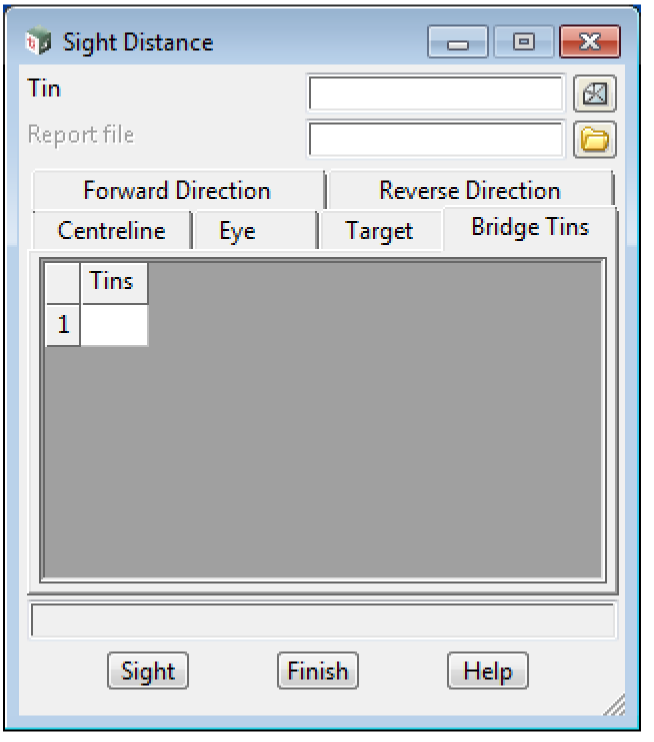
the target position is determined by finding the chainage along the selected string, going out perpendicularly for the given **target offset** *(negative to the left, positive to the right), dropping that position onto the given tin, and then adding the* **target height** *to the height on the tin. Hence the target point is always the target height above the tin.*

Height input 0.3

height of the target point above the given tin

Offset input -0.5

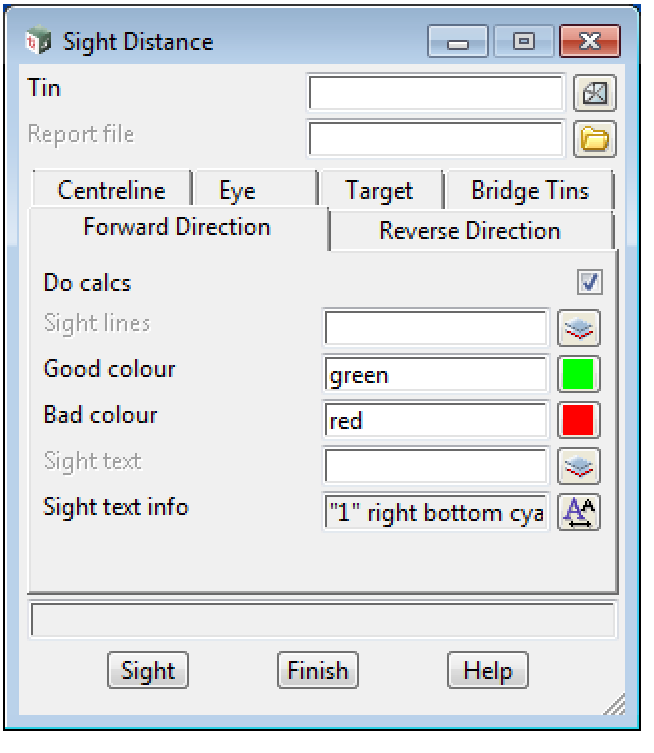
offset of the target point from the picked string.



Bridge Tins tab

Tin grid available tins

name of the tins to use as vertical obstructions when testing visibility.



Forward Direction tab

Do calcs tick box

if ticked, do the calcs for the forward direction

Sight lines model box available models

if non-blank, the name of the model to contain the sight lines.
If blank, the sight lines are not created.

Good colour colour box default colour available colours

when the minimum sight distance is achieved, sight lines are created in the good colour

Bad colour colour box default colour available colours

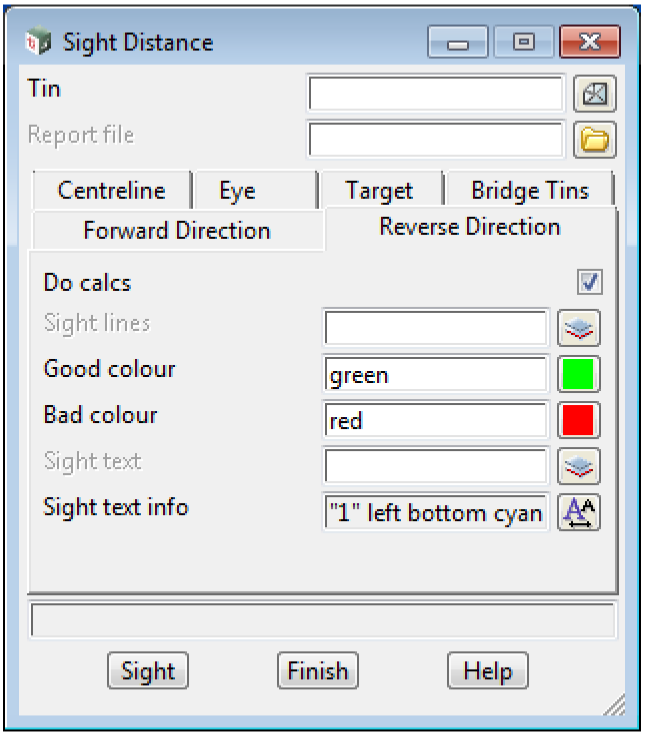
when the minimum sight distance is not achieved, sight lines are created in the Bad colour

Sight text model box available models

if non-blank, the name of the model to contain the sight distance text.
If blank, the sight distance text is not created.

Sight text info text info

definition of the sight text.



Reverse Direction tab

Do calcs tick box

if ticked, do the calcs for the reverse direction

Sight lines model box available models

if non-blank, the name of the model to contain the sight lines when going down the string in the reverse direction Note that the eye and target offsets are then applied in the reverse direction of the string.
If blank, the sight lines are not created.

Good colour colour box default colour available colours

when the minimum sight distance is achieved, sight lines are created in the good colour

Bad colour colour box default colour available colours *when the minimum sight distance is not achieved, sight lines are created in the Bad colour*

Sight text model box available models

if non-blank, the name of the model to contain the sight distance text.
If blank, the sight distance text is not created.

Sight text info text info

definition of the sight text.

Sight button

sight lines and text will be calculated for the given string and tin, and placed in the sight line and text models.

20.10.8 Viewshed - Tin Only

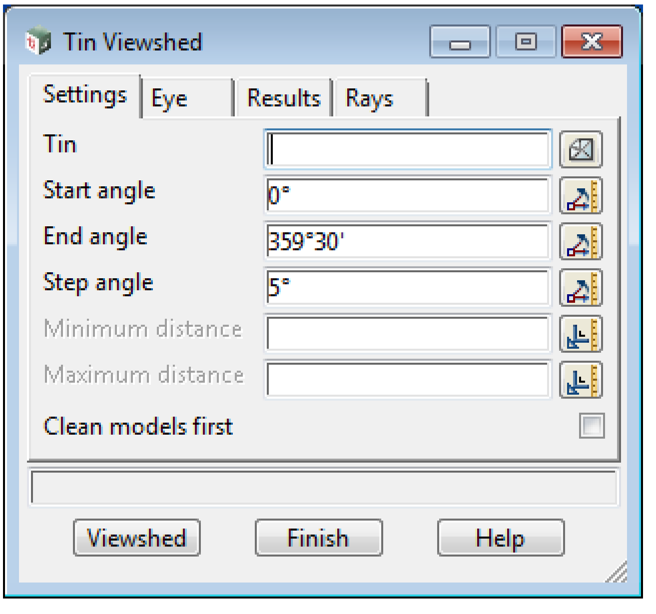
Position of option on menu: **Design =>Sight lines =>Viewshed - tin only**
Utilities =>H-Z =>Sight lines => Viewshed - tin only

This panel is used to calculate the points that are visible and invisible from a user selected eye position. This is equivalent to calculating the points that can or can not see a selected point.

The calculations are made along rays emanating from the eye point from a minimum to a maximum distance from the eye point. The rays are created at regular angular steps from a start angle to an end angle.

The calculations are done in the same manner as for the **Advanced Viewshed** except that there is only a Tin. There are no Bridge Tins or Obstructions (see 20.10.3 Viewshed All).

Selecting **Viewshed** displays the **Tin Viewshed** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Setting tab

Tin	tin box		available tins

name of the tin to be used for the viewshed analysis.

Start angle	angle box	0	

the angle to begin taking sight rays emanating from the eye point.

End angle	angle box	359.30	

the angle to stop taking sight rays emanating from the eye point.

Step angle	angle box	5	

the angle between successive sight rays emanating from the eye point.

Minimum distance	input		

if not blank, the minimum distance from the eye point along the sight ray to begin recording visible/ invisible points.
If blank, the minimum distance is zero.

Maximum distance	input		

the maximum distance from the eye point along the sight ray to record visible/invisible points. If blank, the entire tin is considered.

Clean model first tick box

if ticked, the models of results are cleaned out before the option runs.

Eye tab

Eye XYZ input/output **xyz ops** menu

X/Y/Z coordinate

the XYZ co-ordinates of the eye point.

Each co-ordinate can be individual entered or the point pick icon used to select an existing point.

Note - the z-value is normally above the surface of the tin and should not be on the tin.

Results tab

More tick box

if ticked, ask for name, model, linestyle, weight.

if not ticked, only ask for model.

Name input

name for the created strings

Model model box available models

the model for the visible/invisible sight strings.

Linestyle linestyle box

linestyle for the created strings

Weight weight box

weight for the created strings

Colour of visible bits colour box default line colour

if not blank, the colour for the visible parts of the rays. If blank, the visible parts are not created.

Colour of invisible bits colour box default point colour

if not blank, the colour for the invisible parts of the rays. If blank, the invisible parts are not created.

Separate strings tick box

if ticked, the visible and invisible sections are separate strings.

If not ticked, the visible and invisible sections are combined into one super string.

Rays tab

Visible Rays

the visible rays are the lines from the eye point to the point on the tin where the terrain goes from being visible to invisible.

If **Colour of visible bits** *is not blank then this is the colour used for these rays.*

If **Colour of visible bits** *is blank then the colour is the default line colour.*

The ray will be coloured where it is over invisible terrain with the colour for the invisible bits (this happens in undulating country. That is where there are valleys).

More tick box

if ticked, ask for name, model, linestyle, weight.

if not ticked, only ask for model.

Name input

name for the created strings

Model model box available models

if not blank, the model for the visible ray strings.
If blank, don't create the visible ray strings

Linestyle linestyle box

linestyle for the created strings

Weight weight box

weight for the created strings

Invisible Rays

the invisible rays are the lines from the eye point to the point on the tin where the terrain goes from being invisible to visible.

If **Colour of invisible bits** *is not blank then this is the colour used for these rays.*

If **Colour of invisible bits** *is blank then the colour is the default point colour.*

The ray comes from the eye and is coloured with the **Colour of visible bits** *to the change over from visible to invisible and then coloured with the* **Colour of invisible bits** *to where the ray ends (the change over visible to invisible).*

More tick box

if ticked, ask for name, model, linestyle, weight.

if not ticked, only ask for model.

Name input

name for the created strings

Model model box available models

if not blank, the model for the invisible ray strings.
If blank, don't create the invisible ray strings

Linestyle linestyle box

linestyle for the created strings

Weight weight box

weight for the created strings

Viewshed button

On selecting this button, the lines of sight emanating from the **eye XYZ** *point are calculated.*

20.10.9 Dynamic Viewshed - Tin Only

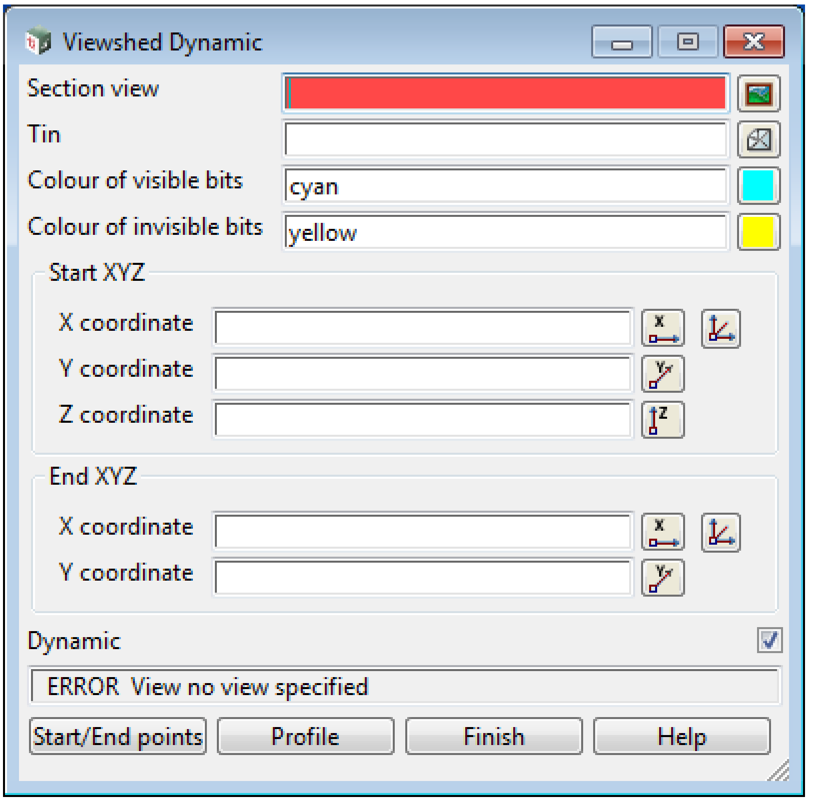
Position of option on menu: **Design =>Sight lines => Viewshed dynamic - tin only**
Utilities =>H-Z =>Sight lines => Viewshed dynamic - tin only

Dynamic viewshed is used to calculate and display on a section view, the points that are visible and invisible from a user selected eye position to a user selected target position.

The target point can be dynamically moved and the sight line is automatically recalculated and displayed on the section view.

The calculations are done in the same manner as for the **Advanced Viewshed** except that there is only a Tin. There are no Bridge Tins or Obstructions (see 20.10.3 Viewshed All).

Selecting **Viewshed dynamic** displays the **Viewshed Dynamic** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Section view	view box		available section view
section view to display the sight lines on			
Tin	tin box		available tins
name of the tin to be used for the viewshed analysis.			
Colour of visible bits	colour box		default line colour
the colour for the visible parts of the viewshed string			
Colour of invisible bits	colour box		default point colour
the colour for the invisible parts of the viewshed string			
Start XYZ	input/output		**xyz ops** menu

X/Y/Z coordinate

the XYZ co-ordinates of the start of the line to determine viewshed for.

Each co-ordinate can be individual entered or the point pick icon used to select an existing point.

End XY input/output **xyz ops** menu

X/Y coordinate

the XY co-ordinates of the end of the line to determine viewshed for.

Each co-ordinate can be individual entered or the point pick icon used to select an existing point.

No Z is required.

Dynamic tick box

if ticked and the **Start/End points** *button is used, once the Start point is selected, the viewshed analysis is carried out from the start point to the cursor position which represents the end point. The results are displayed in the section view.*

Start/End points button

if picked, an start point is selected and then the height displayed in an **enter height** *input box. If required, a new height can be typed into the* **enter height** *box. After typing an <enter> into the input box, the box is removed from the screen and the x, y and height is piped into the* **Start XYZ** *panel field.*

The cursor is then used to select the End point.

If Dynamic *is ticked, as the Cursor position is moved, the viewshed is carried out to the new cursor position and the results displayed in the section view.*

Profile button

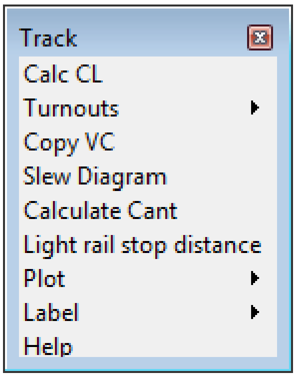
Selecting this button calculates the line of sight from the start to the end position.

20.11 Track

Position of menu: **Design =>Track**

The Track options are for working on rail design.

The **Track** walk-right menu is



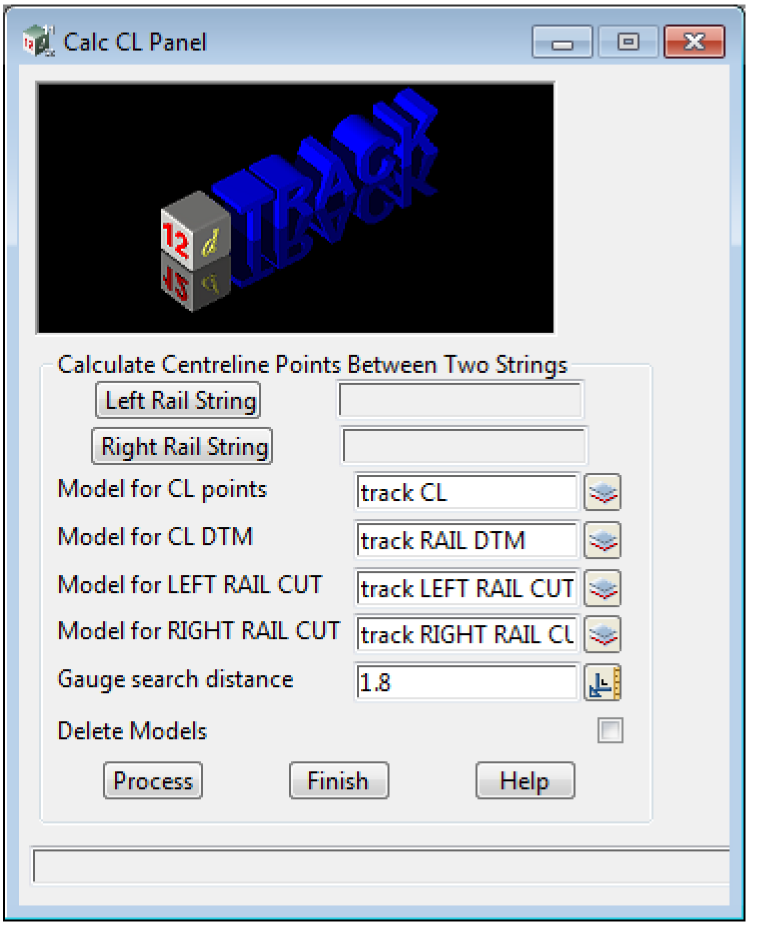
The ***Track*** options have their own Help system which is brought up by clicking on **Help** on the **Track menu**, or clicking **Help** on each of the **Track panels.**

For the option *Calc CL*, go to	20.11.1 Calc CL Panel
Turnouts	20.11.2 Track Turnouts
Copy VC	20.11.3 Copy VC
Slew Diagram	20.11.4 Rail Slew Calculator
Calculate Cant	20.11.5 Calculate Cant
Light rail stop distance	20.11.6 Light Rail Stopping Distance
Plot	20.11.7 Track Plot
Label	20.11.8 Track Label
Help	20.11.9 Help

20.11.1 Calc CL Panel

Position of menu: **Design =>Track =>Calc CL**

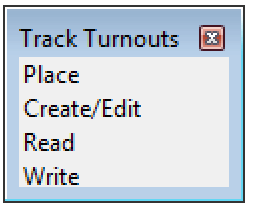
The ***Track*** options have their own Help system which is brought up by either clicking the **Help** button on the panel or by selecting the **Help** option on the Track menu.



20.11.2 Track Turnouts

Position of menu: **Design =>Track =>Turnouts**

The **Track Turnouts** walk-right menu is

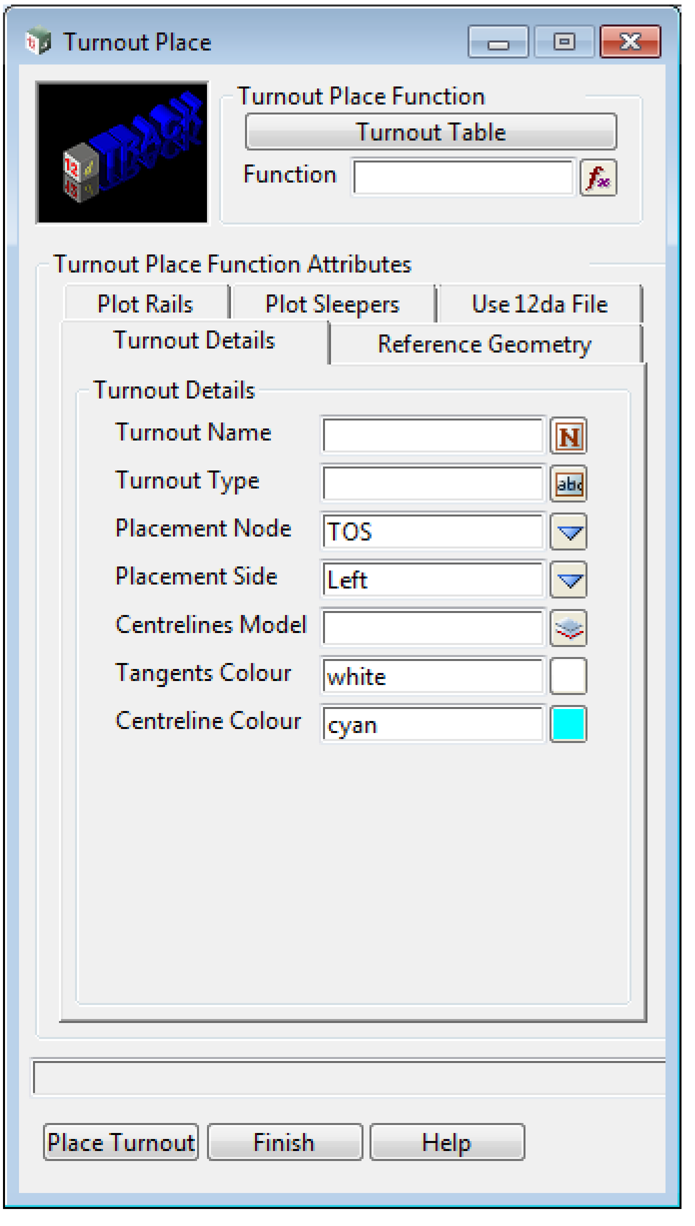


For the option *Place*, go to	20.11.2.1 Turnout Place
Create/ Edit	20.11.2.2 Turnouts Create/ Edit
Read	20.11.2.3 Read Turnouts
Write	20.11.2.4 Write Turnouts File

20.11.2.1 Turnout Place

Position of option on menu: **Design =>Track =>Turnouts => Place**

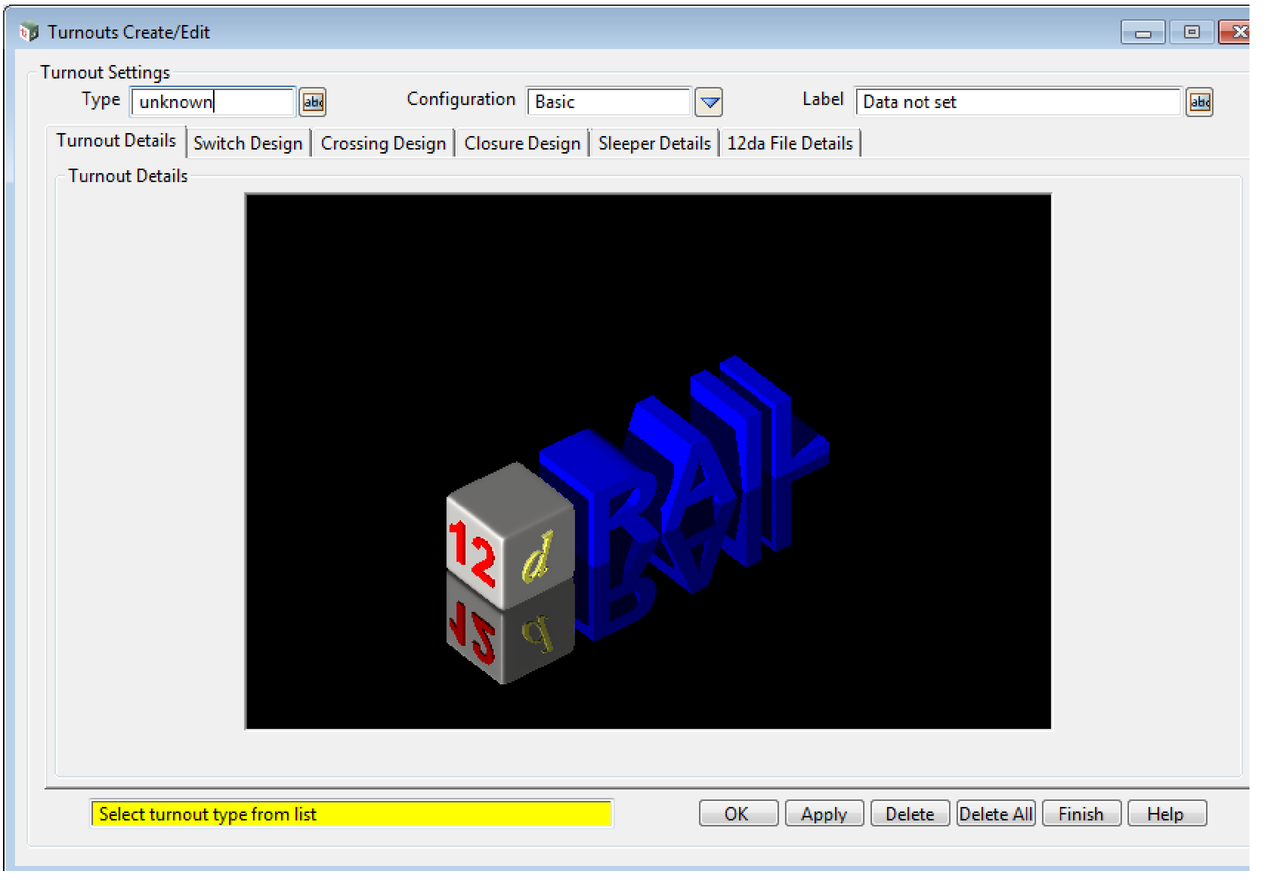
The ***Track*** options have their own Help system which is brought up by either clicking the **Help** button on the panel or by selecting the **Help** option on the Track menu.



20.11.2.2 Turnouts Create/ Edit

Position of option on menu: **Design =>Track =>Turnouts =>Create/Edit**

The ***Track*** options have their own Help system which is brought up by either clicking the **Help** button on the panel or by selecting the **Help** option on the **Track** menu.

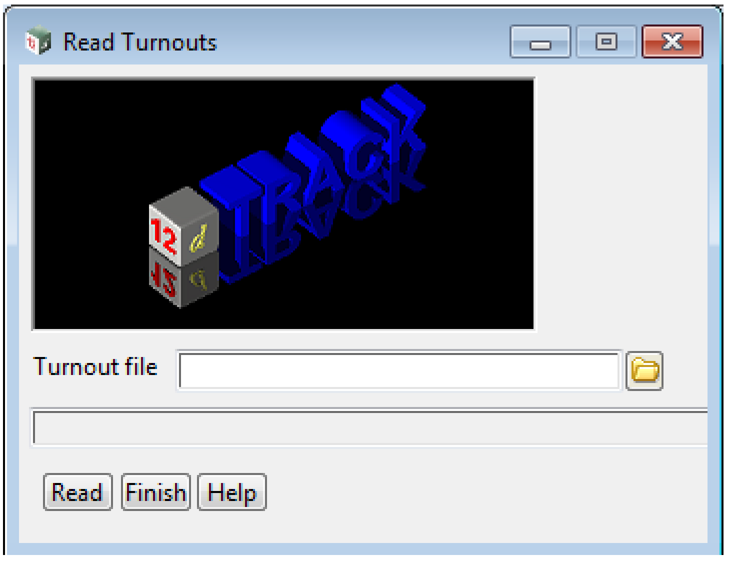


20.11.2.3 Read Turnouts

Position of option on menu: **Design =>Track =>Turnouts =>Read**

The ***Track*** options have their own Help system which is brought up by either clicking the **Help** button on the panel or by selecting the **Help** option on the **Track** menu.

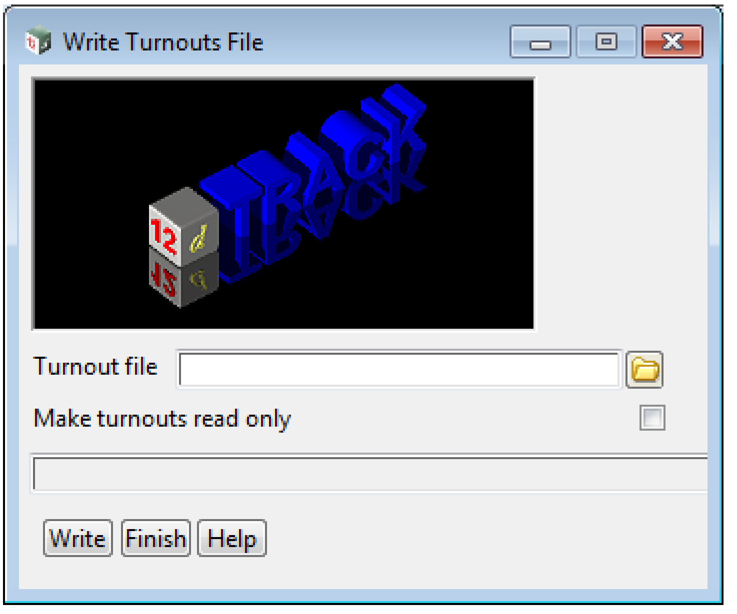
.



20.11.2.4 Write Turnouts File

Position of option on menu: **Design =>Track =>Turnouts =>Write**

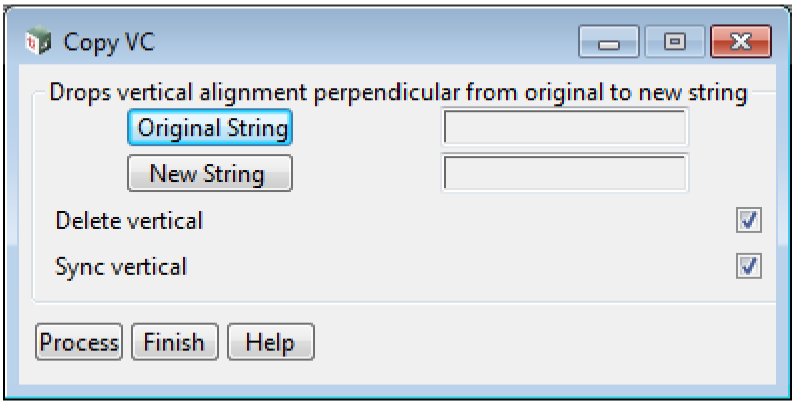
The ***Track*** options have their own Help system which is brought up by either clicking the **Help** button on the panel or by selecting the **Help** option on the **Track** menu.



20.11.3 Copy VC

Position of menu: **Design => Track => Copy VC**

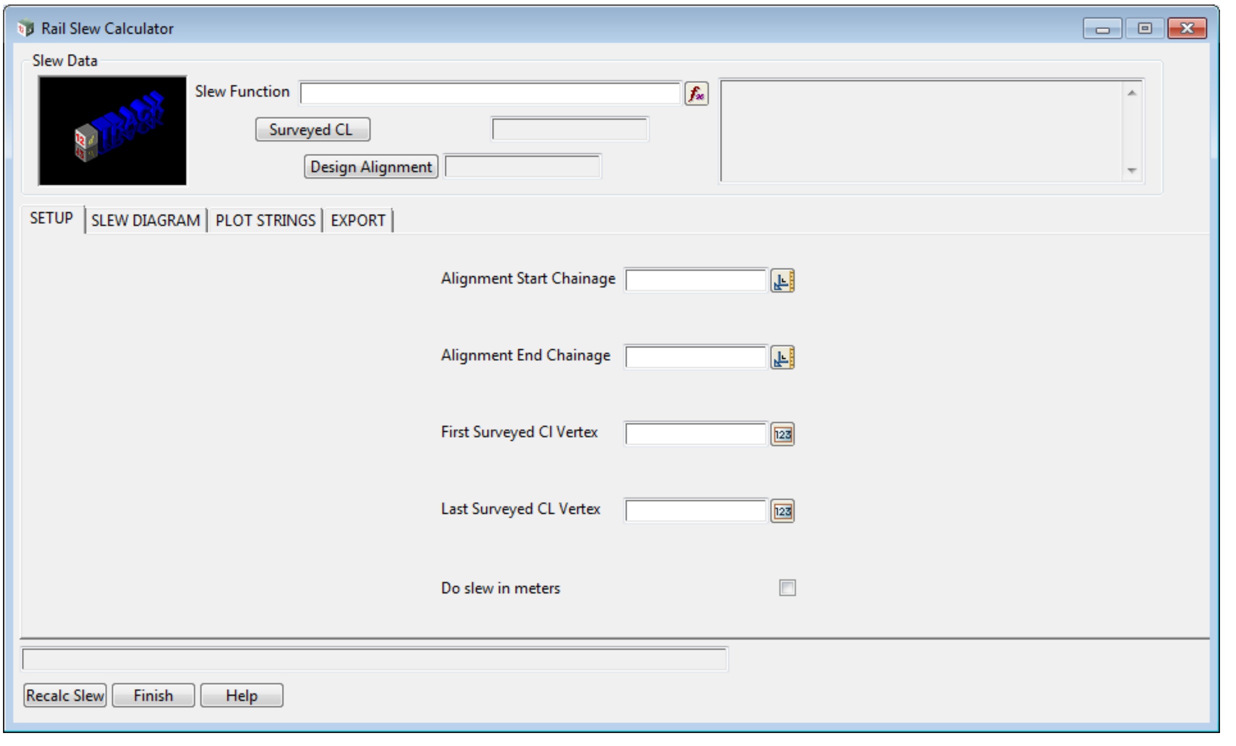
The ***Track*** options have their own Help system which is brought up by either clicking the **Help** button on the panel or by selecting the **Help** option on the Track menu.



20.11.4 Rail Slew Calculator

Position of menu: **Design => Track => Slew Diagram**

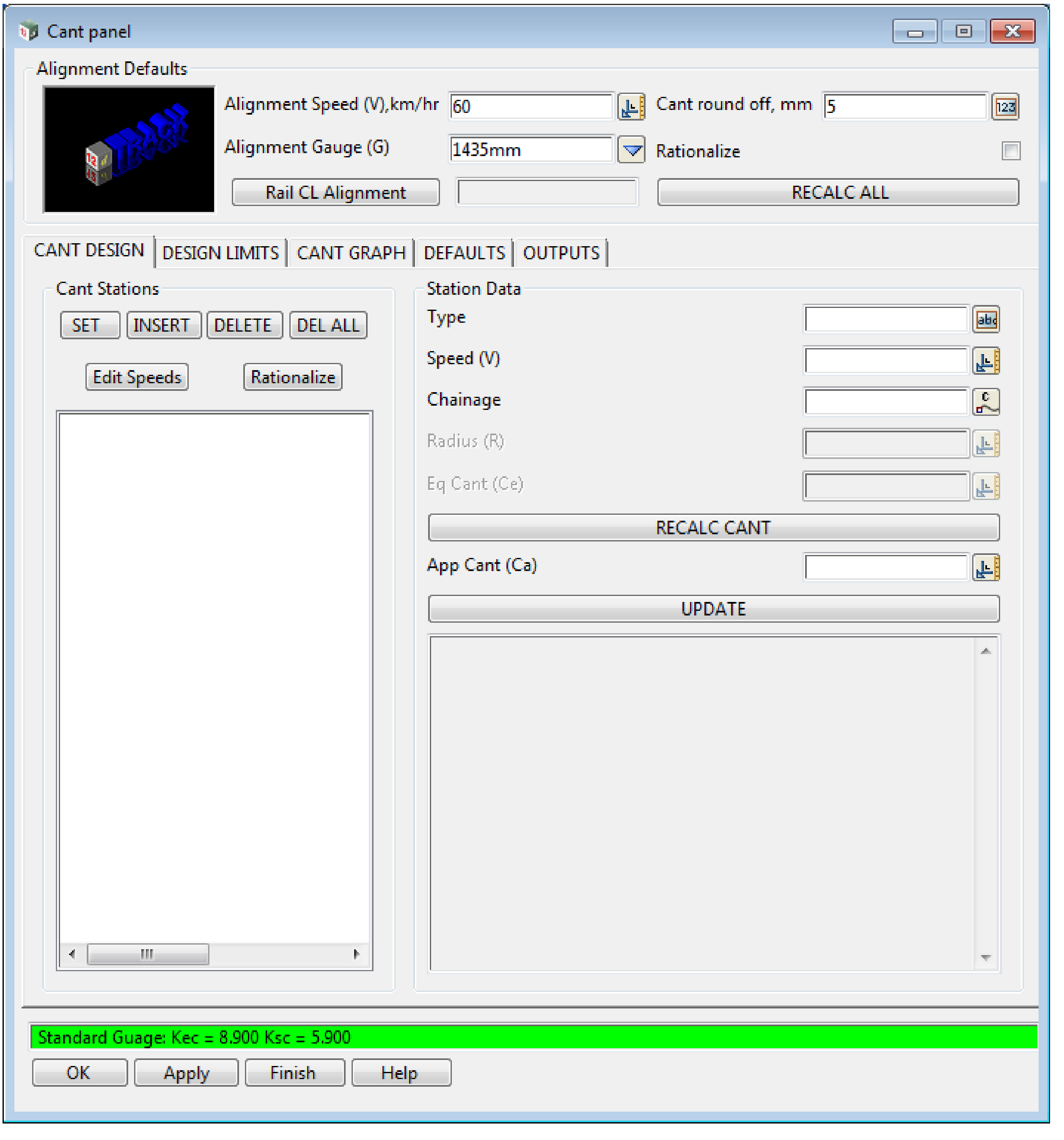
The ***Track*** options have their own Help system which is brought up by either clicking the **Help** button on the panel or by selecting the **Help** option on the Track menu.



20.11.5 Calculate Cant

Position of menu: **Design => Track => Calculate Cant**

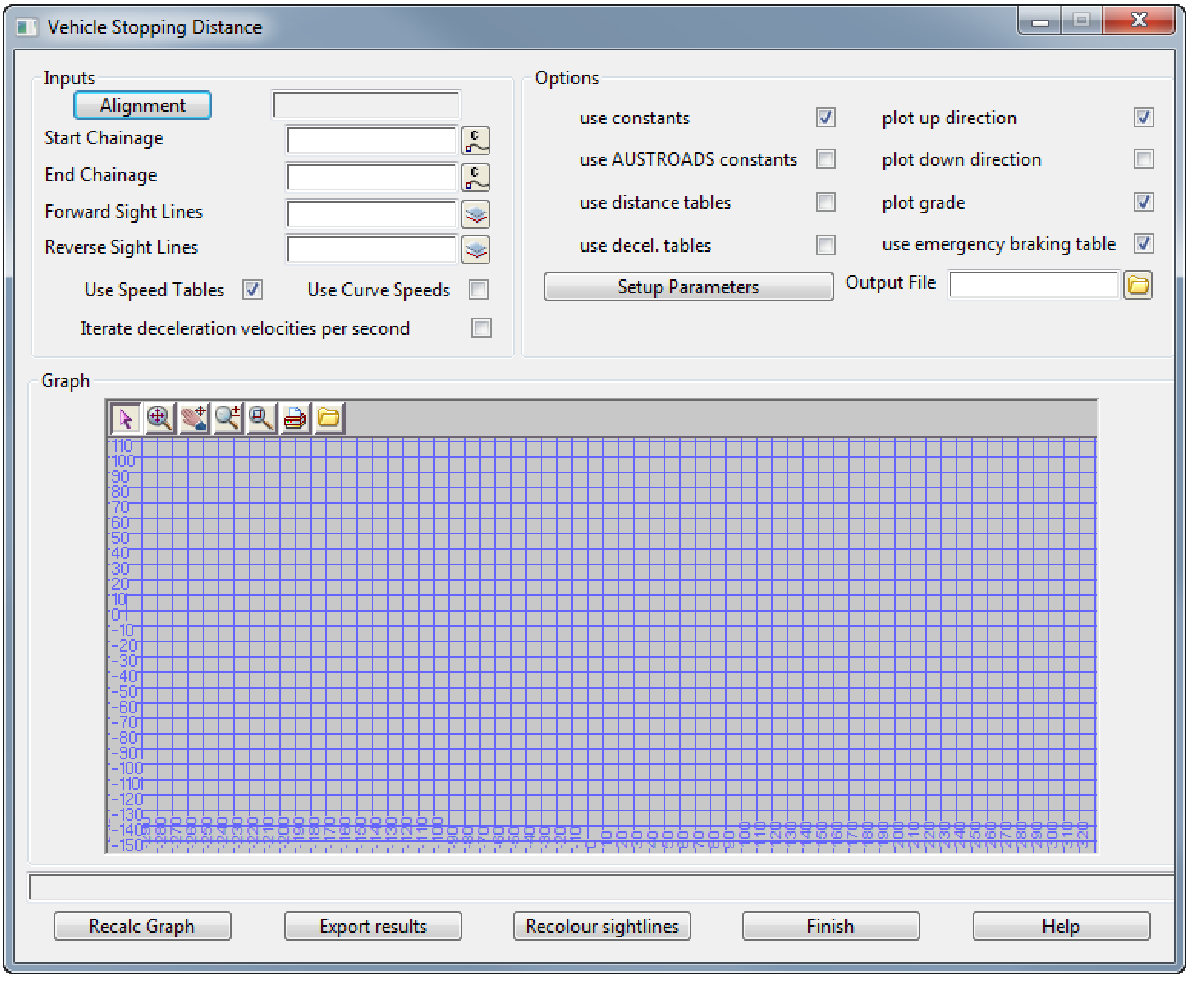
The ***Track*** options have their own Help system which is brought up by either clicking the **Help** button on the panel or by selecting the **Help** option on the Track menu.



20.11.6 Light Rail Stopping Distance

Position of menu: Design => Track => Light rail stop distance

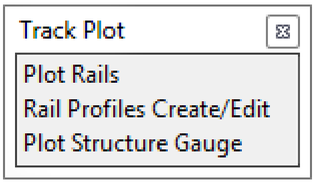
This section of documentation is a work in progress and will be updated in subsequent releases.



20.11.7 Track Plot

Position of menu: **Design =>Track =>Plot**

The **Track Plot** walk-right menu is

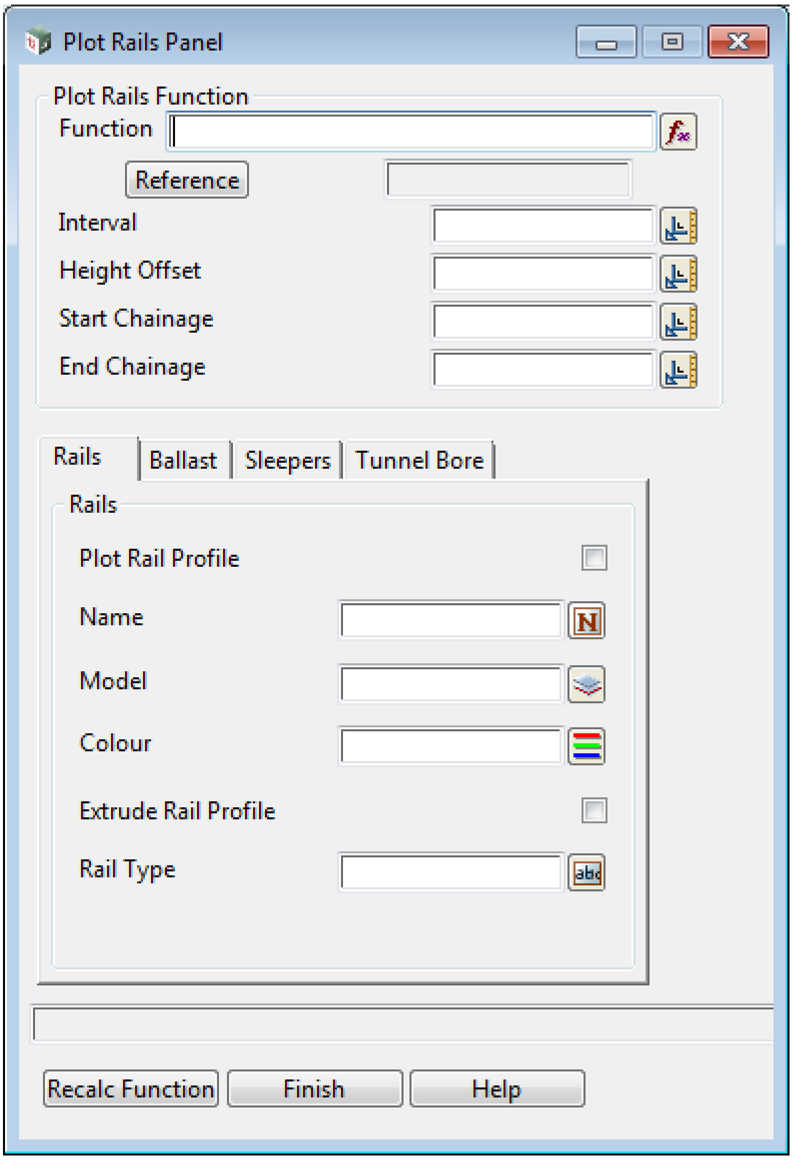


For the option *Plot Rails*, go to 20.11.7.1 Plot Rails Panel
Rail Profiles Create/Edit 20.11.7.2 Rail Create Profile Panel
Plot Structure Gauge 20.11.7.3 Structure Gauge Panel

20.11.7.1 Plot Rails Panel

Position of option on menu: **Design =>Track =>Plot =>Plot Rails**

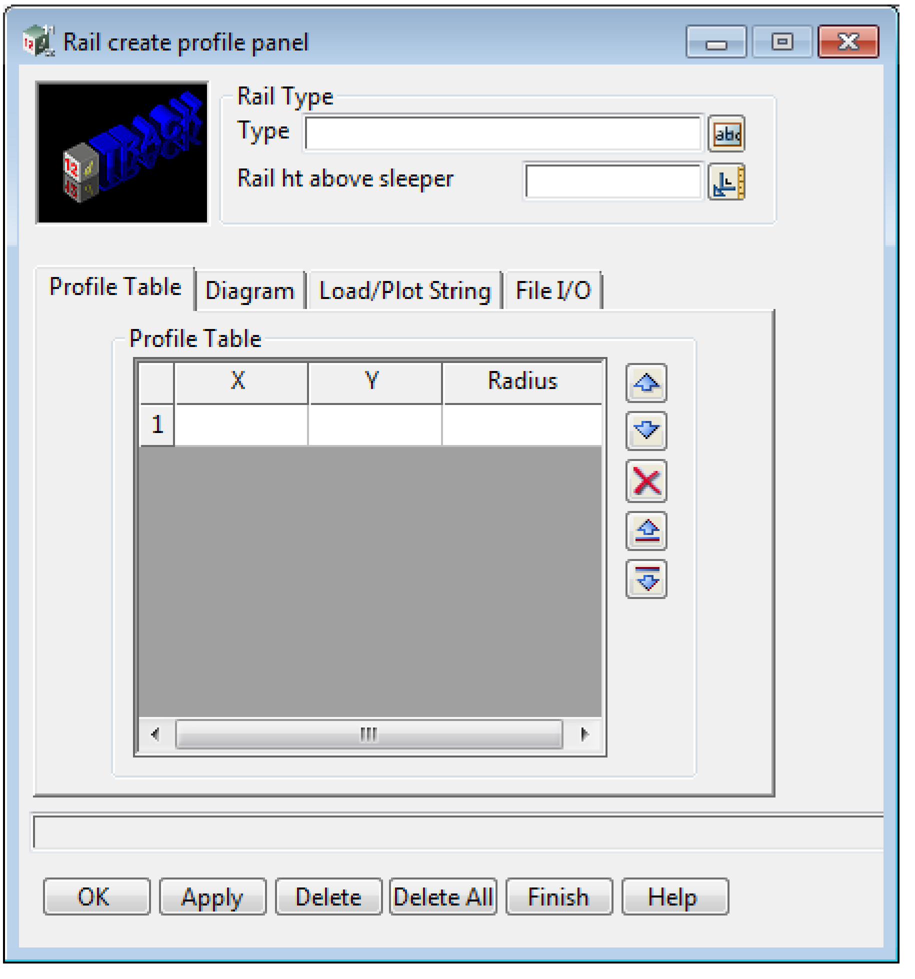
The ***Track*** options have their own Help system which is brought up by either clicking the **Help** button on the panel or by selecting the **Help** option on the Track menu.



20.11.7.2 Rail Create Profile Panel

Position of option on menu: **Design =>Track =>Plot =>Rail Profiles Create/Edit**

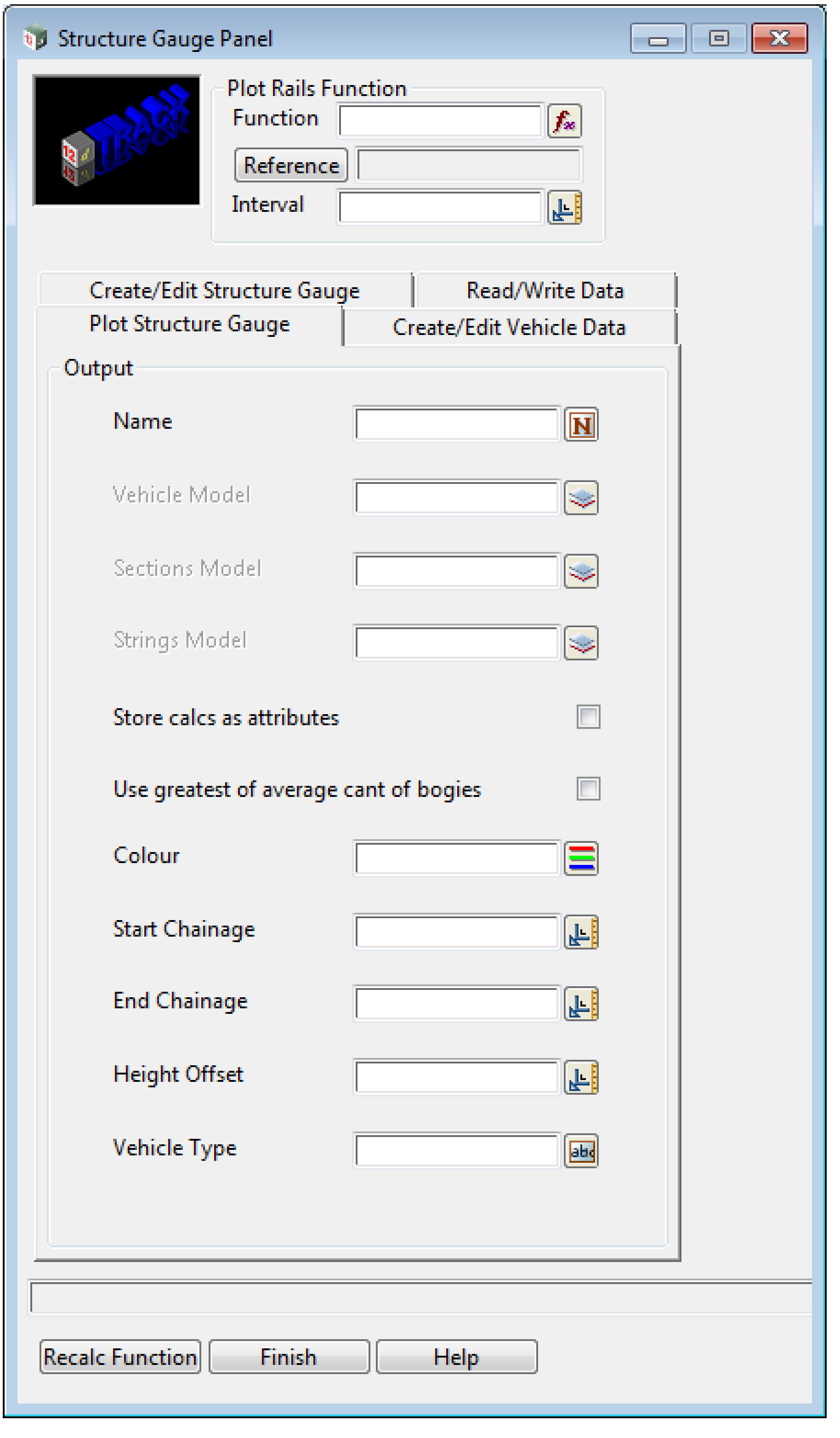
The ***Track*** options have their own Help system which is brought up by either clicking the **Help** button on the panel or by selecting the **Help** option on the **Track** menu.



20.11.7.3 Structure Gauge Panel

Position of option on menu: **Design =>Track =>Plot=>Plot Structure Gauge**

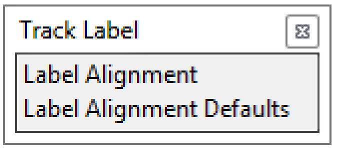
The ***Track*** options have their own Help system which is brought up by either clicking the **Help** button on the panel or by selecting the **Help** option on the Track menu.


Structure Gauge Panel
Plot Rails Function
Function
Reference
Interval
Create/Edit Structure Gauge
Read/Write Data
Plot Structure Gauge
Create/Edit Vehicle Data
Output
Name
Vehicle Model
Sections Model
Strings Model
Store calcs as attributes
Use greatest of average cant of bogies
Colour
Start Chainage
End Chainage
Height Offset
Vehicle Type
Recalc Function
Finish
Help

20.11.8 Track Label

Position of menu: **Design =>Track =>Label**

The **Track Label** walk-right menu is

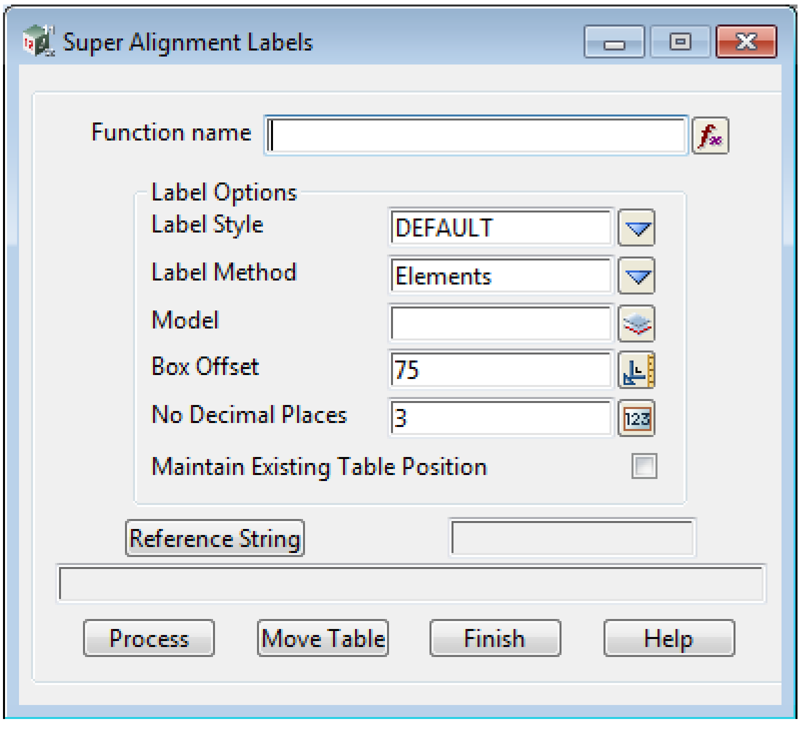


For the option *Label Alignment* go to 20.11.8.1 Super Alignment Labels
Label Alignment Defaults 20.11.8.2 Super Alignment Style Extras Create/Edit

20.11.8.1 Super Alignment Labels

Position of option on menu: **Design =>Track =>Label=>Label Alignment**

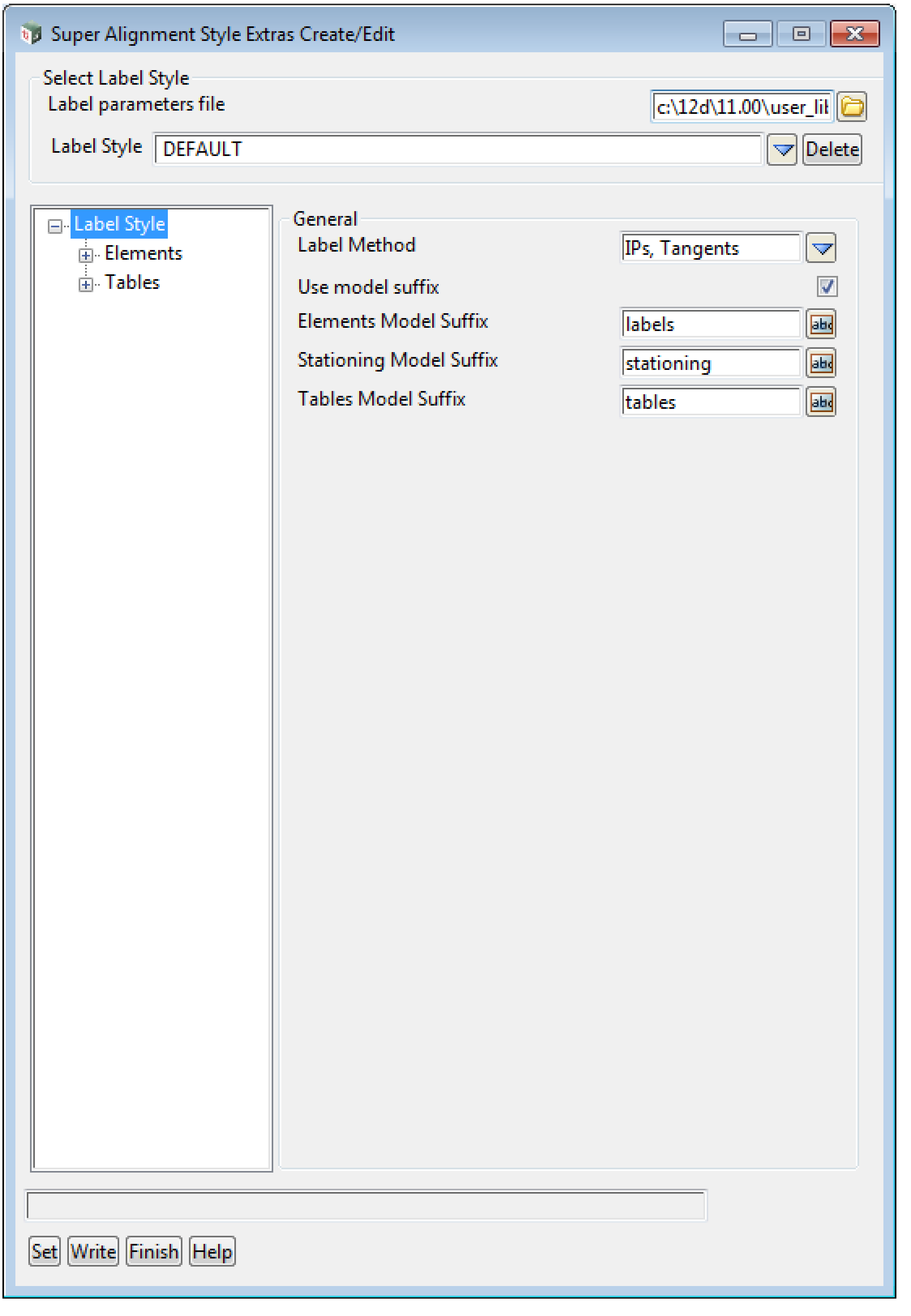
The ***Track*** options have their own Help system which is brought up by either clicking the **Help** button on the panel or by selecting the **Help** option on the Track menu.



20.11.8.2 Super Alignment Style Extras Create/Edit

Position of option on menu: **Design =>Track =>Label=>Label Alignment Defaults**

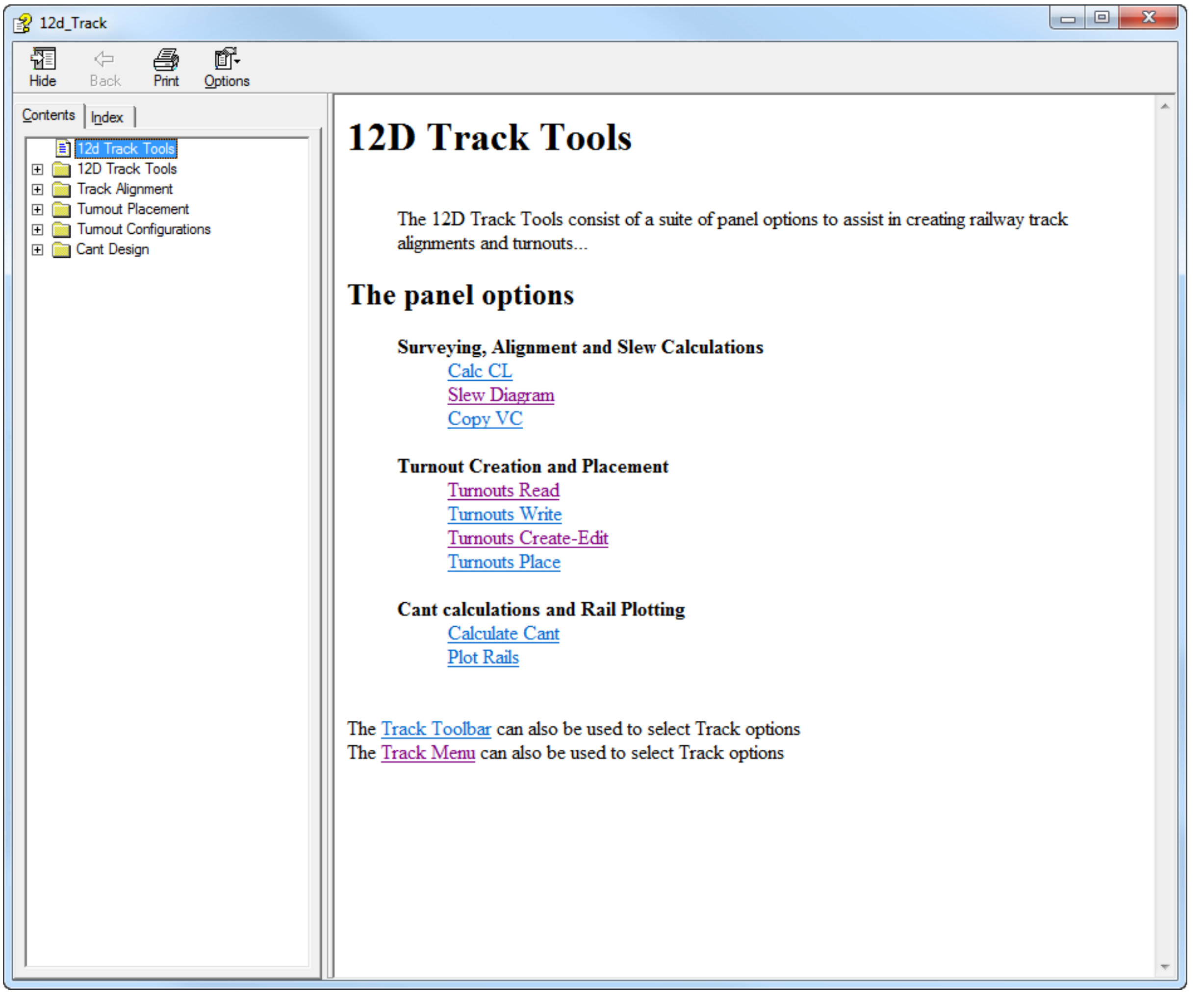
The ***Track*** options have their own Help system which is brought up by either clicking the **Help** button on the panel or by selecting the **Help** option on the Track menu.


Super Alignment Style Extras Create/Edit
Select Label Style
Label parameters file
c:\12d\11.00\user_lit
Label Style
DEFAULT
Delete
Label Style
Elements
Tables
General
Label Method
IPs, Tangents
Use model suffix
Elements Model Suffix
labels
Stationing Model Suffix
stationing
Tables Model Suffix
tables
Set
Write
Finish
Help

20.11.9 Help

Position of menu: **Design =>Track => Help**

The ***Track*** options have their own Help system which is brought up by clicking on **Help** on the **Track menu,** or clicking **Help** on each of the **Track** panels.



20.12 Tunnels and Structures

Position of menu: **Design =>Tunnel-Structures**

In *Version 11.0*, the creation of tunnel profiles and the generation of a tunnel or structure has been enhanced from *Version 10.0* to make the process easier.

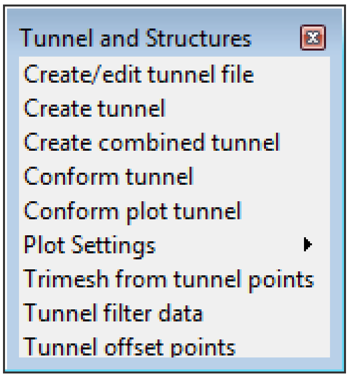
Tunnels and structures are defined by a number of ***profiles***, each of which must have the same number of named vertices, and these profiles are applied along a centreline (alignment). New profiles are generated at user given chainages along the tunnel/structure centreline, and any profile generated between two defined profiles is the linear interpolation by chainage of the two defined profiles.

The defined tunnel profiles and how they are applied down a string are specified in the **Create/ Edit Tunnel File** panel, and this information is stored in a *Tunnel Definition File* (ending in ***.12d_Tunnel***).

The *Tunnel Definition File* is used in other options such as **Create Tunnel** which generates the profile and longitudinal strings, and trimesh, for the tunnel, **Conform Tunnel** for tunnel conformance, and in ***12d Field*** for setting out tunnels.

Note: a ***Trimesh*** can be generated for a tunnel by the **Create Tunnel** panel.

The **Tunnel** menu is:



See

20.12.1 Creating/Editing a Tunnel Definition File
20.12.2 Create Tunnel

20.12.4 Conform Tunnel
20.12.5 Conform Plot Tunnel
20.12.6 Plot Settings
20.12.7 Trimesh from Tunnel Points

20.12.1 Creating/Editing a Tunnel Definition File

Position of option on menu: **Design =>Tunnel-Structures =>Create/edit tunnel file**

The **Create/Edit Tunnel File** panel is used to define the tunnel definition file which records:

(a) the profiles used to define the tunnel

(b) the tunnel centreline (the tunnel reference string) that the tunnel profiles are applied to

(c) whether the tunnel profiles are applied perpendicular (*i.e.* normal) or vertically to the tunnel centreline

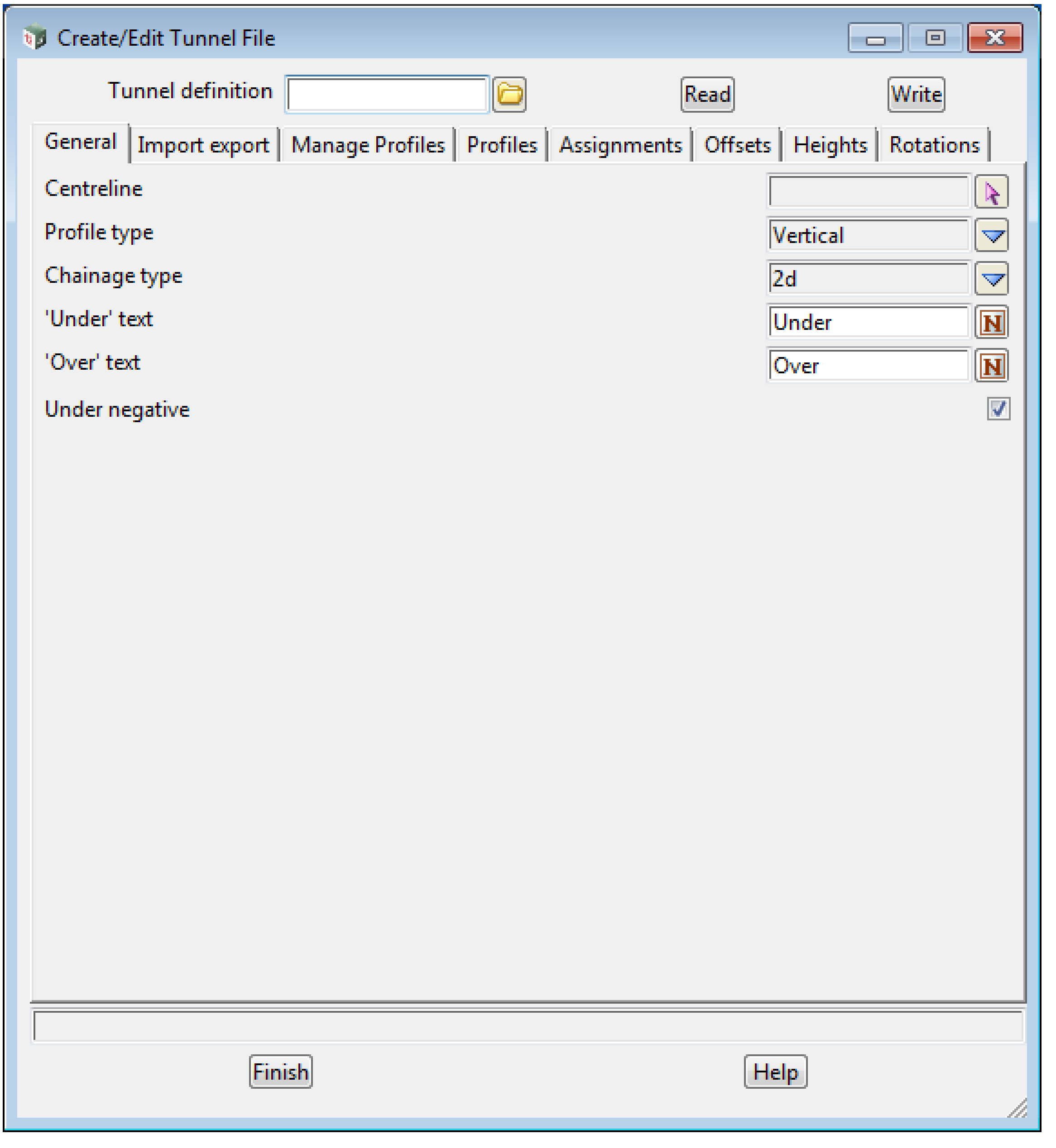
(d) whether 2d or 3d chainages are used in the tunnel definitions.

(e) the chainages where the profiles are applied, and optional height, grade and rotation changes when the profiles are applied along the tunnel centreline.

Note tunnel profiles and longitudinal strings can be generated perpendicular (*i.e.* normal) to the defined centreline resulting in a true 3d representation of the tunnel.

One restriction is that a tunnel cannot be completely vertical - there must be some chainage difference in the vertical alignment. But given that a chainage difference as small as 0.01mm can be used then this restriction is rarely a problem.

Selecting **Create/edit tunnel file** brings up the **Create/Edit Tunnel File** panel



Top Section

Tunnel definition tunnel file box available 12d_tunnel files

the tunnel file to contain all the definitions for the tunnel. That is, the tunnel centreline, the tunnel profiles and which profiles and how often they are applied along the centreline of the tunnel.

Read button

read in the file given in the **Tunnel definition** *field.*

Write button

write out all the values in the panel to the file giving in the **Tunnel definition** *field.*

Export (PRO/PRA) button

write out the appropriate values in the panel as a PRO *and* PRA *file.*

Note*: PRO and PRA are files use in TP Setout and TP Stakeout. See* 20.12.10 Definition of the PRO and PRA definition files.

General tab

Tunnel centreline

*the centreline that the tunnel definitions are applied to. The centreline must have valid vertical geometry for all the chainage ranges used in the **Assignments**, **Offsets Heights** and **Rotations** tabs.*

Profile type choice box perpendicular, vertical

if **perpendicular**, *the tunnel profiles are applied perpendicularly (i.e. normal) to the vertical alignment of the centreline. That is, perpendicular to the tangent vector to the centreline at that chainage. This results in a true 3d model.*

If **vertical,** *the tunnel is calculated vertical to the vertical alignment of the centreline. This may mean an effective loss of clearance on steeper grades.*

Chainage type choice box 2d, 3d

if **2d**, *the chainages in the tunnel definitions are taken as plan chainages.*

if **3d**, *are interpreted as 3d. That is, it is the plan/2d chainage of the **1st point** where the horizontal and vertical geometry coincide, plus the 3d length along the centreline from there.*

Manage Profiles tab

A tunnel can contain any number of defined profiles that are used to linearly extrapolate between.

*The **Manage Profile** tab lets you create a new profile:*

(a) using an existing profile as a seed profile and the new profile is given a copy of the seed profile's geometry. The profile geometry is then edited using the **Profiles** *tab.*

or

(b) create a new profile that has no geometry and the geometry is entered using the **Profiles** *tab.*

Seed profile profiles box available profiles in this tunnel definitions file

*if **not blank**, it must be a profile in this tunnel definitions file and it is used as a Seed profile for the new profile being created.*

New name text box

name for the new profile being created. The new name must be unique amongst the profiles in this tunnel definitions file.

Create button

create a new profile with the geometry being a copy of the geometry of the Seed profile.

Delete profile profiles box available profiles in this tunnel definitions file

the profile in this tunnel definitions file to delete.

Delete button

delete the profile given in the **Delete profile** *field.*

Profile to rename profiles box available profiles in this tunnel definitions file

the profile in this tunnel definitions file to be renamed.

New name text box

new name for the profile. The new name must be unique amongst the profiles in this tunnel definitions file.

Rename button

rename the profile given in the **Profile to rename** *field to the name given in the* **New name** *field.*

Import profile profiles box available profiles in this tunnel definitions file

the profile in this tunnel definitions file to delete.

Delete button

delete the profile given in the **Delete profile** *field.*

Profiles tab

A tunnel can contain any number of defined profiles that are used to linearly extrapolate between.

The ***Profile*** *tab lets you create new profiles, pick existing profiles to edit and graphically display the shape of the profile for you.*

Current Profile choice box list of profiles in the current tunnel

Select the profile you wish to create or edit. If the selected profile already has vertices defined, they will be displayed in the grid, and the shape drawn in the area under the grid.

Set button

Records the contents of the current grid as the profile given in the **Current profile** *field.* **Set** *must be pressed before moving to another profile or the contents of the current grid will be lost.*

Profile grid

Name text box

the name of the vertex of the profile. It must be unique to this profile.

Offset real box

the offset of the vertex from the centreline. Positive offset is to the right of the centreline, negative offset is to the left of the centreline.

Height real box

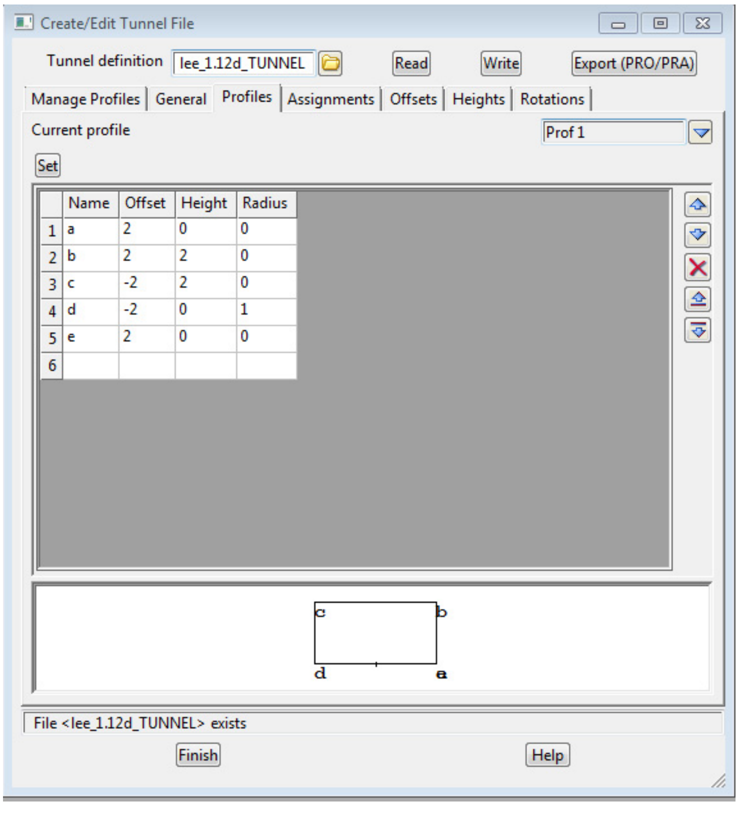
the height difference of the vertex from the centreline. Positive height is above the centreline, negative height is below the centreline.

Radius real box

the radius of the segment from this named vertex to the next named vertex. A positive radius is a right handed or clockwise curve with respect to the order of the vertices in the grid.

Zero (0) means that there is no radius and there is a straight line between this vertex and the next named vertex.

There must always be a value for the radius.



Assignments tab

This grid is used to define where the defined profiles are assigned on the centreline.
For a row in the grid, the named profile starts at the chainage in that row. For any chainage between this chainage and the chainage on the next row of the grid, a profile is ***generated*** *by interpolating by chainage between the name profile on this row and the named profile on the next row.*

The same profile can be used any number of times in the Assignment grid.

The chainages in the Assignments grid must be in ascending order.

Set button

Records the contents of the Assignments grid. **Set** *must be pressed before moving to another tab or the contents of the current grid will be lost.*

Assignments grid

Chainage real cell

The chainage where the named profile begins.

For any chainage between this chainage and the one on the next row, a profile is generated by interpolated by chainage between this profile and the one on the next row of the grid.

Name profiles cell available profiles

The name of the profile to start at this chainage.

Offsets/Heights/Rotations tab

A profile can be adjusted by applying an offset, a (delta) height and a rotation.
The Offsets/Heights/Rotations grid is used to define the offset/height/grade or angle applied to a profile.

The offset/height/grade or angle on a row of the grid starts at the chainage in that row. For any chainage between the chainage of this row and the chainage of the next row, an offset/height/grade or angle for the profile at the chainage is calculated by interpolating between the offset/height/grade or angle of the profile in this row and the offset/height/grade or angle on the following row of the grid.

The chainages in the Offsets/Heights/Rotations grid must be in ascending order.

***Notes**:*

***1**. the **height** is a **delta height**. That is, it is added to the height of the profile.*

***2.** the **rotation** in a row can be given as either a **Percent Grade** (positive is up) **or** an **Angle** (dms in hp notation and is measure in counterclockwise from the offset axis). The percent grade or angle will be referred to as the rotation.*

Set button

Records the contents of the Offsets/Heights/Rotations grid. **Set** *must be pressed before moving to another tab or the contents of the current grid will be lost.*

Offsets/Heights/Rotations grid

Chainage real cell

The chainage to start applying the offset/height/rotation.

For any chainage between this chainage and the one on the next row of the grid, an offset/height/rotation is generated by interpolated by chainage between this offset/height/rotation and the one in the next row of the grid.

Offset/Height/Grade or Angle real cell

the offset/height/grade or angle to start at this chainage.

For any chainage between this chainage and the one on the next row, an offset/height/rotation is calculated by interpolated by chainage between this offset/height/rotation and the one on the next row of the grid.

Type choice cell None, Linear, Cubic Rev C, Biquadratic

the type of interpolation used for the offset/height/rotation at chainage between the chainage of this row and the chainage on the next row.

***None**: no interpolation is done and the offset/height/ at this chainage is used. If there is a different offset/height/rotation on the next row then there will be a problem and you need to insert another row after this row with a chainage slightly less than the chainage on the next row. The offset/height/rotation of the inserted row needs to be the same as this row and the Type should be **Linear**.*

***Linear**: offset/height/rotation is linearly interpolated by chainage between the offset/height/rotation on this row and the offset/height/rotation on the next row.*

***Cubic**: offset/height/rotation is interpolated as a cubic by chainage between the offset/height/rotation on this row and the offset/height/rotation on the next row.*

***Rev C**: offset/height/rotation is interpolated as a reverse cubic by chainage between the offset/height/rotation on this row and the offset/height/rotation on the next row.*

***Biquadratic**: offset/height/rotation is interpolated as a back to back quadratics by chainage between the offset/height/rotation on this row and the offset/height/rotation on the next row.*

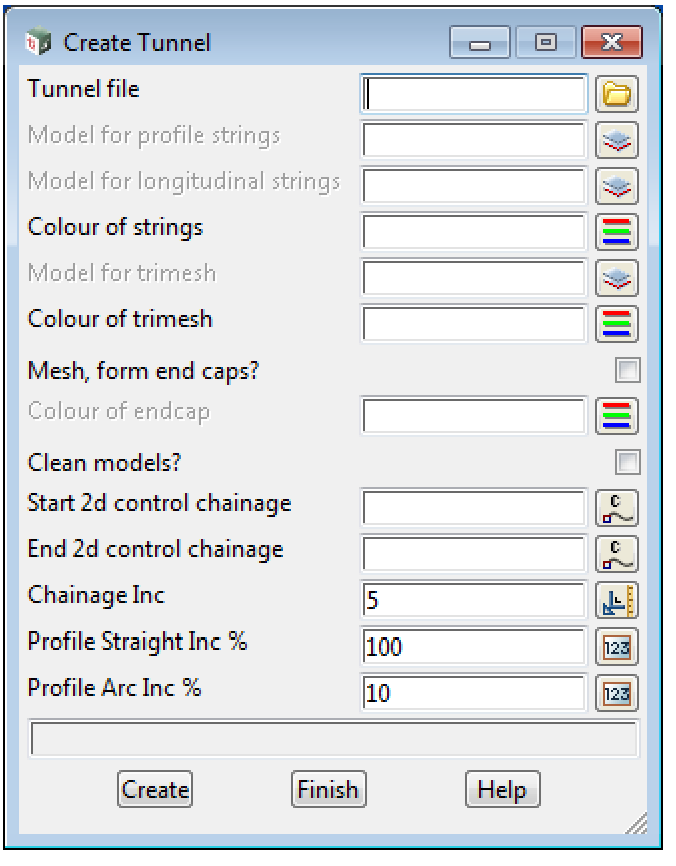
20.12.2 Create Tunnel

Position of option on menu: **Design =>Tunnel-Structures =>Create tunnel**

The **Create Tunnel** panel is used to generate the strings defining the shape of the tunnel - the profile and longitudinal strings, and the trimesh, for the tunnel.

The tunnel is defined in a ***Tunnel File*** that has been created in the **Create/Edit Tunnel File** panel.

Selecting **Create Tunnel** brings up the **Create Tunnel** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Tunnel file tunnel file box available 12d_tunnel files

the tunnel file containing all the definitions for the tunnel. That is, the tunnel centreline, the tunnel profiles and which profiles and how often they are applied along the centreline of the tunnel.

Model for profile strings model box

*if **not blank**, a string of the tunnel profile is created at each chainage point along the tunnel centreline and stored in this model.*

*If **blank**, no strings of the tunnel profiles are created.*

Model for longitudinal strings model box

*if **not blank**, strings are created by joining common vertices of the profiles along the tunnel centreline and stored in this model.*

*If **blank**, no longitudinal strings are created.*

Colour of strings colour box available colours

colour of the profiles and longitudinal strings.

Model for trimesh model box

*if **not blank**, a trimesh is created for the tunnel and placed in this model.*
*If **blank**, no trimesh is created.*

Colour of trimesh colour box available colours

colour of the trimesh.

Mesh, form end caps? tick box

*if **ticked**, the trimesh includes a mesh over each end of the tunnel. This creates a closed trimesh. A volumes is calculated and reported in a String inquire on a closed trimesh.*

*If not **ticked**, the trimesh ends are left open. A volumes is NOT reported in a String inquire on an open trimesh.*

Colour of endcap colour box available colours

colour of the endcap.

Clean models? tick box

*if **ticked** the profile and longitudinal strings, and trimesh models are cleaned before the new strings and trimesh are created.*

Start/End 2d control chainage real box

the Start/End 2d chainage on the centreline to create the trimesh, longitudinal strings and profile strings for.

Chainage Inc real box

the chainage increment to create the sections and vertices of the strings at. Whether the chainage is 2d or 3d is defined in the Tunnel Definition File.

Profile Straight/Arc Inc% real box

this defines how often vertices are generated around the tunnel profiles between the vertices of the tunnel profile, and hence how may vertices are created in the sections, how many longitudinal strings and the density of triangles in the trimesh.
*The value is a percentage to the length of the tunnel profile and for each tunnel section, vertices are generated at each of the defining vertices of the tunnel profile, and then at the **Profile Straight/Arc Inc%** distance between adjacent defining vertices of the tunnel profile.*
*Hence the **Profile Straight/Arc Inc%** options control the accuracy of the shapes generated for the tunnel - straights and arcs are detected, and defined values assigned accordingly.*
***Note** - the values **Profile Straight/Arc Inc%** give a good approximation to the number of vertices are created on each section but there may be a few more because all the defining vertices of the profile as in the sections. So if **Profile Straight/Arc Inc%** is 5 then there will be approximately 20 vertices on each section.*

Create button

create the tunnel profile strings, longitudinal strings, and trimesh.

20.12.3 Create Combined Tunnel

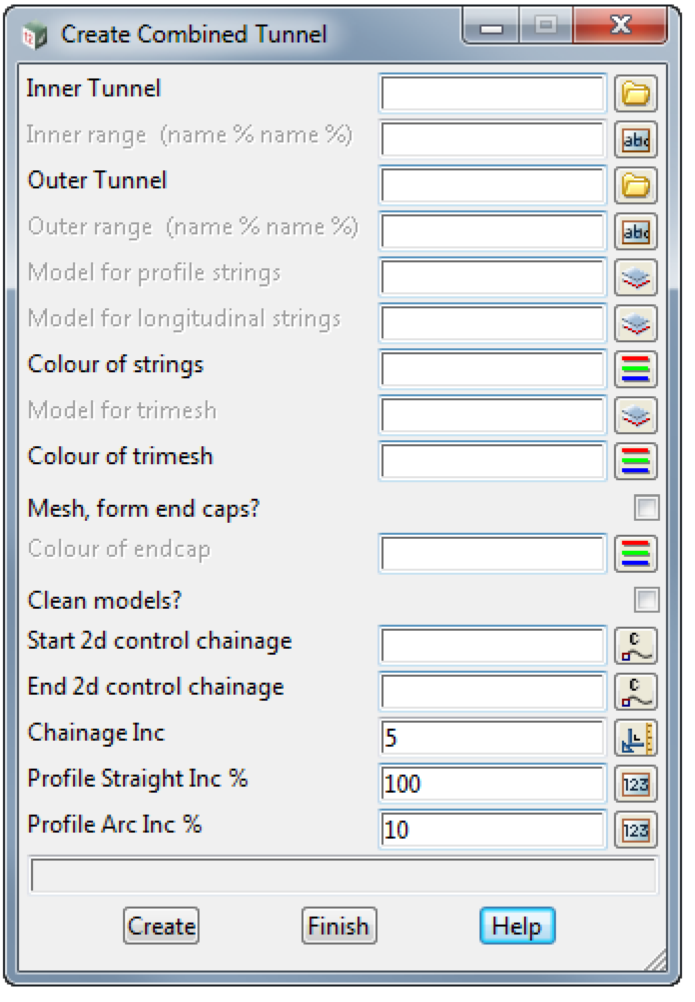
Position of option on menu: **Design =>Tunnel-Structures =>Create combined tunnel**

CURRENTLY A WORK IN PROGRESS

The **Create Combined Tunnel** panel is used to

The tunnel is defined in a ***Tunnel File*** that has been created in the **Create/Edit Tunnel File** panel.

Selecting **Create combined tunnel** brings up the **Create CombinedTunnel** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Inner Tunnel	tunnel file box		available 12d_tunnel files
Inner range (name % name %	name box		
Outer Tunnel	file box		
Outer range (name % name %)	name box		
Model for profile strings	model box		

*if **not blank**, a string of the tunnel profile is created at each chainage point along the tunnel centreline and stored in this model.*

*If **blank**, no strings of the tunnel profiles are created.*

Model for longitudinal strings model box

*if **not blank**, strings are created by joining common vertices of the profiles along the tunnel centreline and stored in this model.*

*If **blank**, no longitudinal strings are created.*

Colour of strings colour box available colours

colour of the profiles and longitudinal strings.

Model for trimesh model box

*if **not blank**, a trimesh is created for the tunnel and placed in this model.*

*If **blank**, no trimesh is created.*

Colour of trimesh colour box available colours

colour of the trimesh.

Mesh, form end caps? tick box

*if **ticked**, the trimesh includes a mesh over each end of the tunnel. This creates a closed trimesh. A volumes is calculated and reported in a String inquire on a closed trimesh.*

*If not **ticked**, the trimesh ends are left open. A volumes is NOT reported in a String inquire on an open trimesh.*

Colour of endcap colour box available colours

colour of the endcap.

Clean models? tick box

*if **ticked** the profile and longitudinal strings, and trimesh models are cleaned before the new strings and trimesh are created.*

Start/End 2d control chainage real box

the Start/End 2d chainage on the centreline to create the trimesh, longitudinal strings and profile strings for.

Chainage Inc real box

the chainage increment to create the sections and vertices of the strings at. Whether the chainage is 2d or 3d is defined in the Tunnel Definition File.

Profile Straight/Arc Inc% real box

this defines how often vertices are generated around the tunnel profiles between the vertices of the tunnel profile, and hence how may vertices are created in the sections, how many longitudinal strings and the density of triangles in the trimesh.

*The value is a percentage to the length of the tunnel profile and for each tunnel section, vertices are generated at each of the defining vertices of the tunnel profile, and then at the **Profile Straight/Arc Inc%** distance between adjacent defining vertices of the tunnel profile.*

*Hence the **Profile Straight/Arc Inc%** options control the accuracy of the shapes generated for the tunnel - straights and arcs are detected, and defined values assigned accordingly.*

***Note** - the values **Profile Straight/Arc Inc%** give a good approximation to the number of vertices are* *created on each section but there may be a few more because all the defining vertices of the profile as in the sections. So if **Profile Straight/Arc Inc%** is 5 then there will be approximately 20 vertices on each section.*

Create button

create the tunnel profile strings, longitudinal strings, and trimesh.

20.12.4 Conform Tunnel

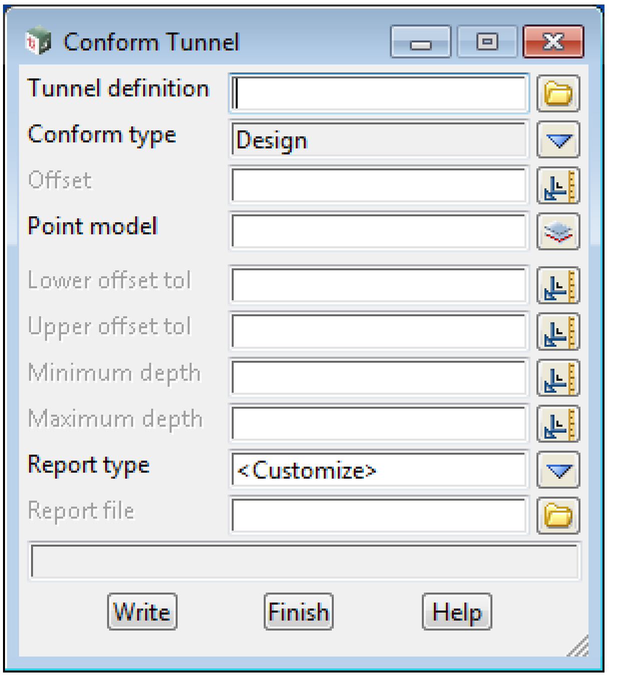
Position of option on menu: **Design =>Tunnel-Structures =>Conform Tunnel**

CURRENTLY A WORK IN PROGRESS

The **Conform Tunnel** panel creates a conformance report between points and a tunnel.

The tunnel is defined in a ***Tunnel definition*** file that has been created in the **Create/Edit Tunnel File** panel.

Selecting **Conform Tunnel** brings up the **Conform Tunnel** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tunnel definition	tunnel file box		available 12d_tunnel files

the tunnel file containing all the definitions for the tunnel. That is, the tunnel centreline, the tunnel profiles and which profiles and how often they are applied along the centreline of the tunnel.

Conform type	choice box		
Offset	measure box		
Point model	model box		
Lower offset tol	measure box		
Upper offset tol	measure box		
Minimum depth	measure box		
Maximum depth	measure box		

Report type choice box

Report file file box

Write button

create the tunnel conformance report.

20.12.5 Conform Plot Tunnel

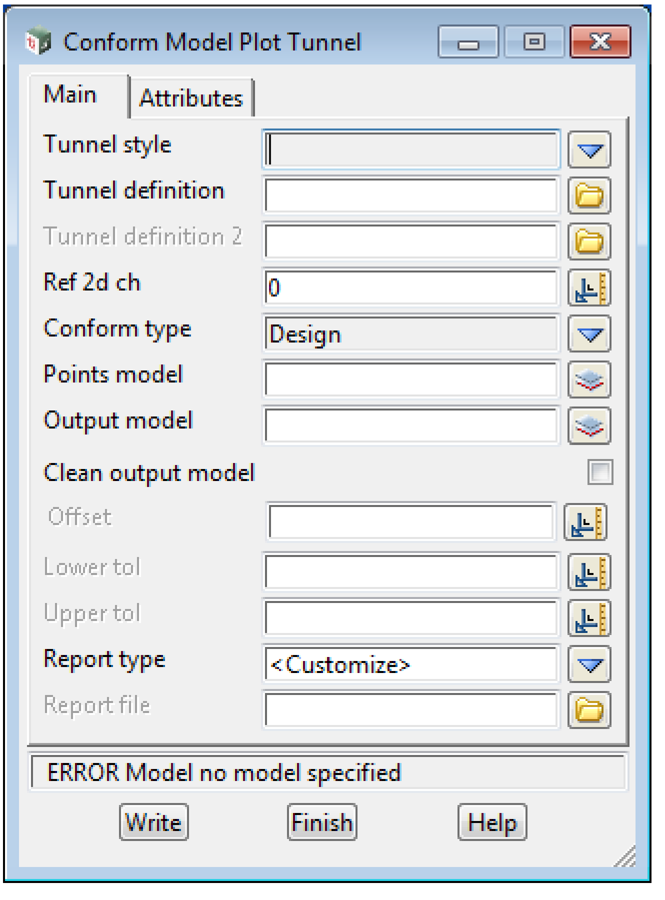
Position of option on menu: **Design =>Tunnel-Structures =>Conform plot tunnel**

CURRENTLY A WORK IN PROGRESS

The **Conform Model Plot Tunnel** panel creates a conformance plot for a tunnel.

The tunnel is defined in a ***Tunnel definition*** file that has been created in the **Create/Edit Tunnel File** panel.

Selecting **Conform plot tunnel** brings up the **Conform Model Plot Tunnel** panel

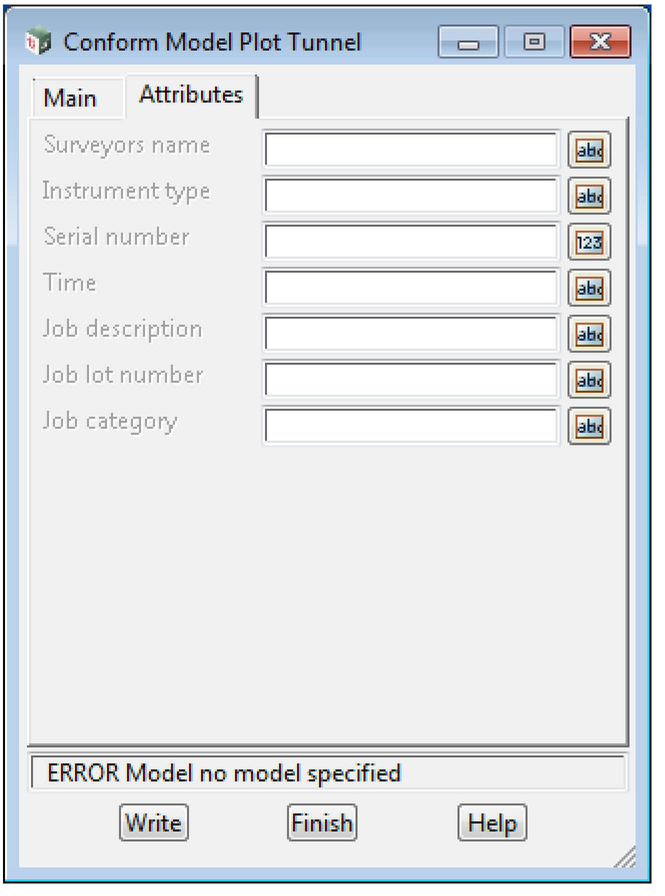


The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tunnel definition	tunnel file box		available 12d_tunnel files

the tunnel definition file containing all the definitions for the tunnel. That is, the tunnel centreline, the tunnel profiles and which profiles and how often they are applied along the centreline of the tunnel.

Field Description	Type	Defaults	Pop-Up
Point model	model box		



Write button

create the tunnel conformance plot.

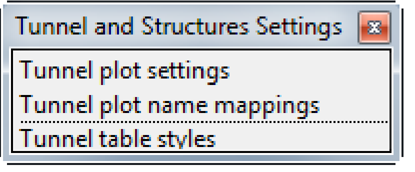
20.12.6 Plot Settings

Position of menu: **Design =>Tunnel-Structures =>Plot settings**

CURRENTLY A WORK IN PROGRESS

The **Plot Settings** options define the look of the tunnel conformance plot.

The **Plot Settings** menu is:



See

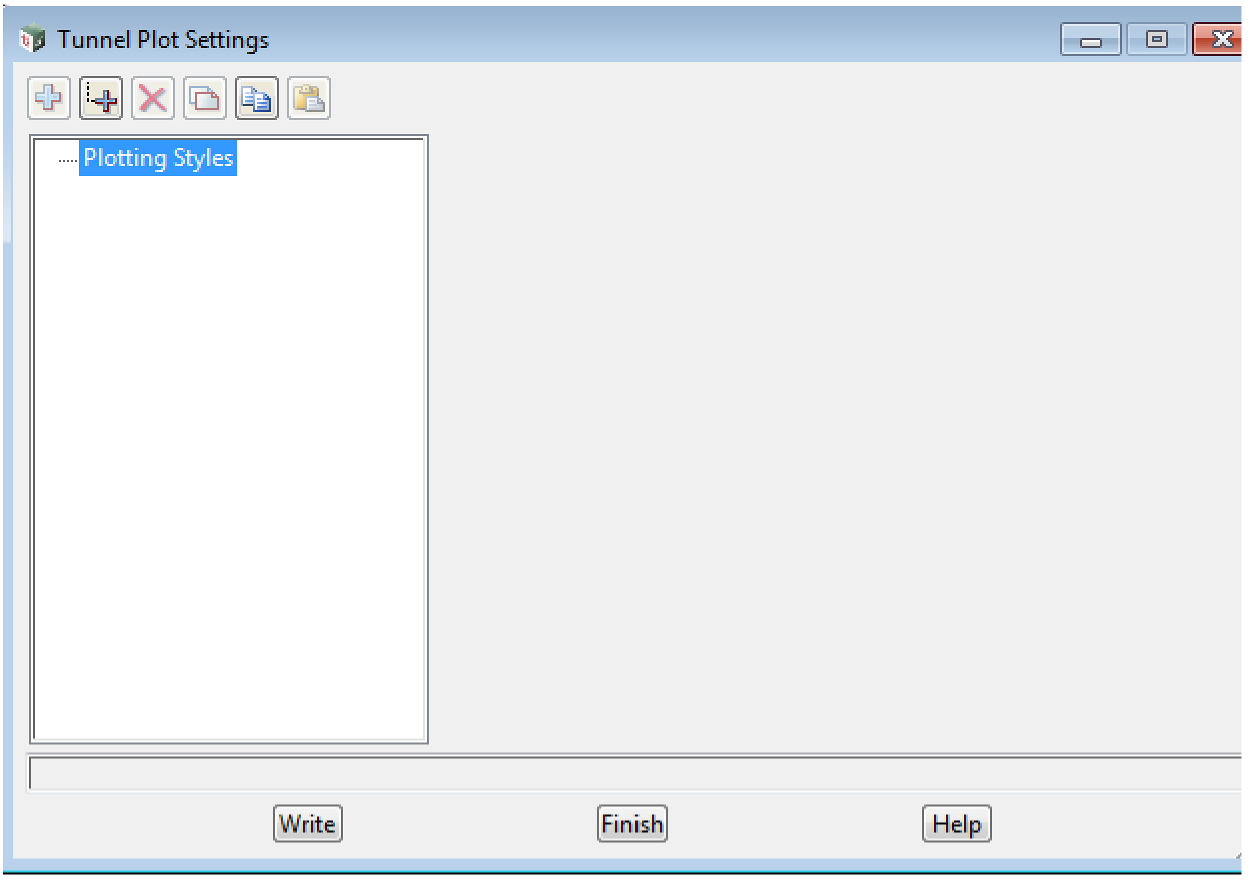
20.12.6.1 Tunnel Plot Settings
20.12.6.2 Tunnel Plot Name Mappings
20.12.6.3 Tunnel Table Styles

20.12.6.1 Tunnel Plot Settings

Position of option on menu: **Design =>Tunnel-Structures =>Plot settings =>Tunnel plot settings**

CURRENTLY A WORK IN PROGRESS

Selecting **Tunnel plot settings** brings up the **Tunnel Plot Settings** panel



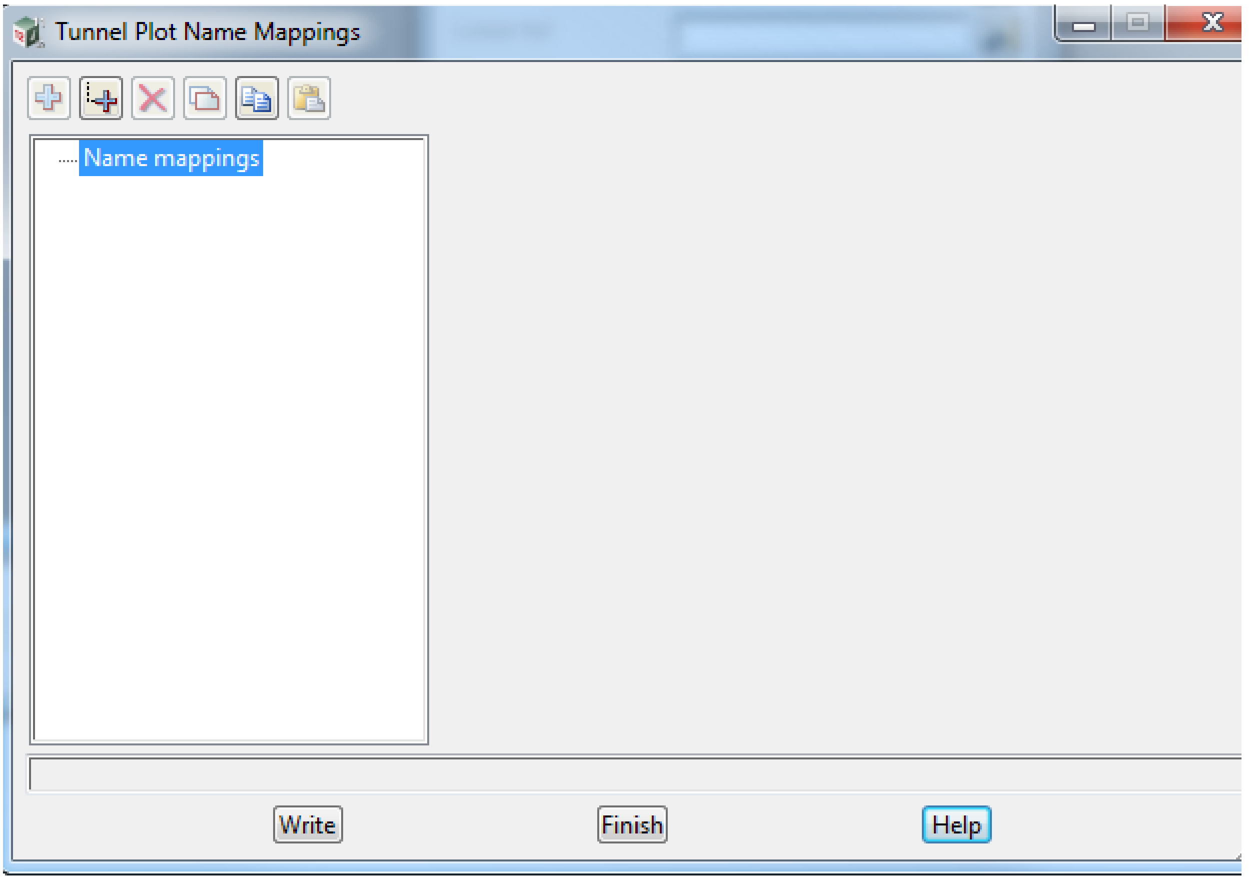
20.12.6.2 Tunnel Plot Name Mappings

Position of option on menu:

Design =>Tunnel-Structures =>Plot settings =>Tunnel plot name mappings

CURRENTLY A WORK IN PROGRESS

Selecting **Tunnel plot name mappings** brings up the **Tunnel Plot Name Mappings** panel



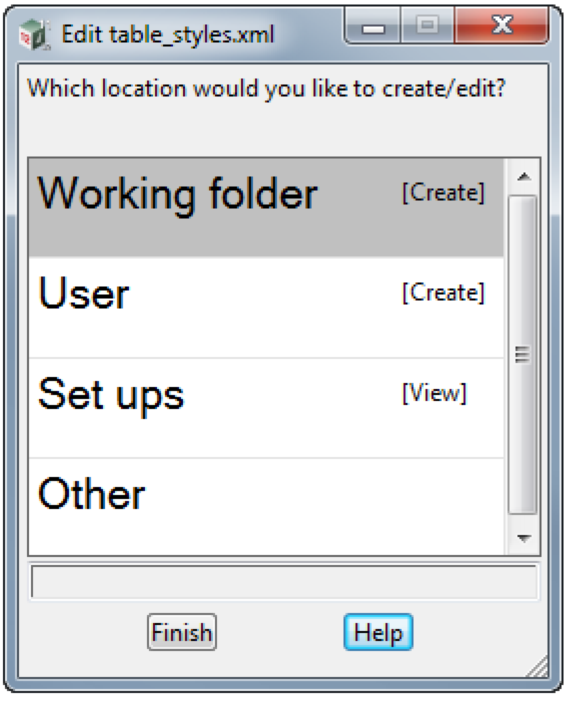
20.12.6.3 Tunnel Table Styles

Position of option on menu:

Design =>Tunnel-Structures =>Plot settings =>Tunnel table styles

CURRENTLY A WORK IN PROGRESS

Selecting **Tunnel table styles** brings up the **Edit tables_styles.xml** panel



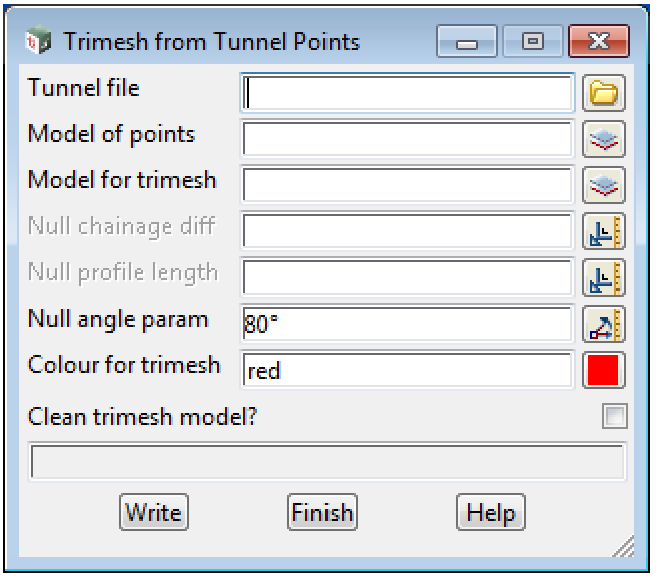
20.12.7 Trimesh from Tunnel Points

Position of option on menu: **Design =>Tunnel-Structures =>Trimesh from tunnel points**

Trimesh from tunnel points takes points that are approximately on the tunnel surface (usually survey shots of the surface of the tunnel), and creates a trimesh from them.

The tunnel itself is defined in a ***Tunnel File*** that has been created in the **Create/Edit Tunnel File** panel. See 20.12.1 Creating/Editing a Tunnel Definition File.

Selecting **Trimesh from tunnel points** brings up the **Trimesh from Tunnel Points** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tunnel file	tunnel file box		available 12d_tunnel files

the tunnel file containing all the definitions for the tunnel. That is, the tunnel centreline, the tunnel profiles and which profiles and how often they are applied along the centreline of the tunnel.

Model of points	model box		available models

the model of points that are approximately on the tunnel surface.

Model for trimesh	model box		available models

model for the created trimesh.

Null chainage diff	real box		

a triangle in a trimesh has an ***outside edge*** *if that edge of the triangle has no adjacent triangle in the trimesh. A triangle in the trimesh with an outside edge is called an* ***outside triangle****.*

Any triangle in the created trimesh that has an outside edge whose difference in the chainages of the two ends of the edge dropped onto the tunnel alignment, is greater than **Null chainage diff***, will be nulled.*

This is used for cleaning up triangles on the edge of the trimesh going along the length of the tunnel.

Null angle param	real box		

*a triangle in a trimesh has an **outside edge** if that edge of the triangle has no adjacent triangle in the trimesh. A triangle in the trimesh with an outside edge is called an **outside triangle**.*

Any outside triangle in the created trimesh whose angle of its normal is greater than the angle of the normal to the tunnel (at the first vertex of the triangle), is greater than **Null angle param** *will be nulled.*

This is useful for cleaning up the end caps, and edge triangles that deviate too much from the direction along the length of the tunnel.

Null profile length real box

*a triangle in a trimesh has an **outside edge** if that edge of the triangle has no adjacent triangle in the trimesh. A triangle in the trimesh with an outside edge is called an **outside triangle**.*

Any outside triangle in the created trimesh with an outside edge whose difference in the profile distance of the two ends of the edge dropped onto the profile around the tunnel, is greater than **Null profile length***, will be nulled.*

This should not be needed too often because **Null chainage diff** *and* **Null angle param** *would generally do the job. Also if there is a radical change in profile across this mesh face then this nulling may not work as intended.*

Colour for trimesh colour box available colours

colour of the trimesh.

Clean trimesh model? tick box

if ticked the model for the trimesh is cleaned before the new trimesh is created.

Create button

create the trimesh from the points.

20.12.8 Tunnel Filter Data

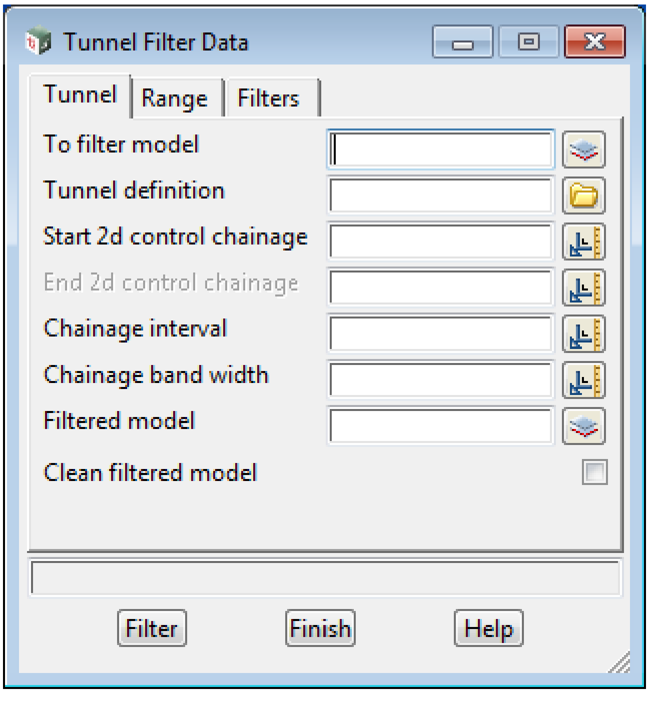
Position of option on menu: **Design =>Tunnel-Structures =>Tunnel filter data**

CURRENTLY A WORK IN PROGRESS

Tunnel Filter Data takes

The tunnel itself is defined in a ***Tunnel File*** that has been created in the **Create/Edit Tunnel File** panel. See 20.12.1 Creating/Editing a Tunnel Definition File.

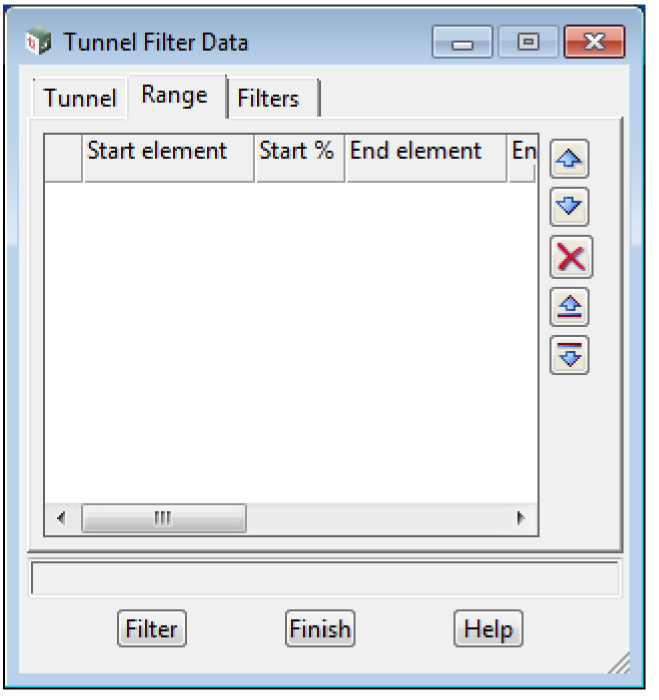
Selecting **Tunnel filter data** brings up the **Tunnel Filter Data** panel



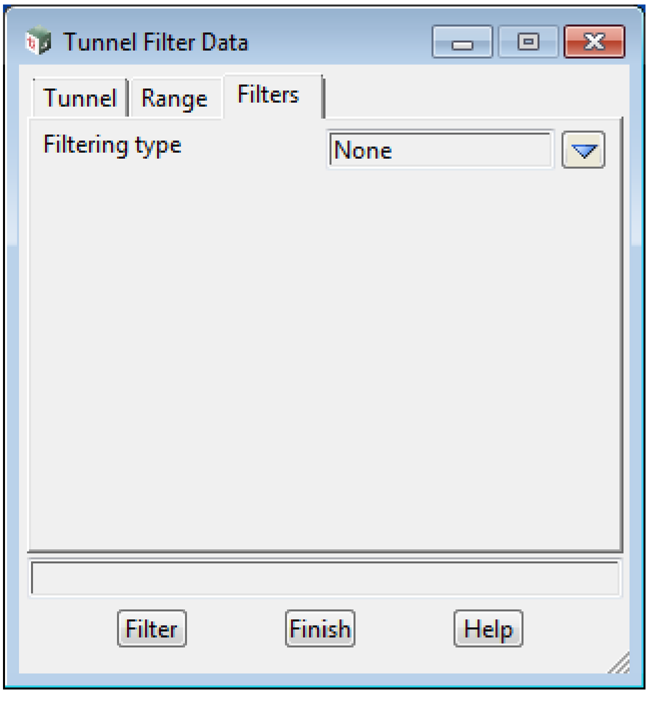
The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tunnel tab			
To filter model	model box		available models
Tunnel definition	file box		available 12d Tunnel files
Start/End 2d control chainage	measure boxes		
Chainage interval	measure box		
Chainage band width	measure box		
Filtered model	model box		
Clean filtered model	tick box		

Range tab



Filters tab



Filtering type choice box

Button at bottom

Filter button

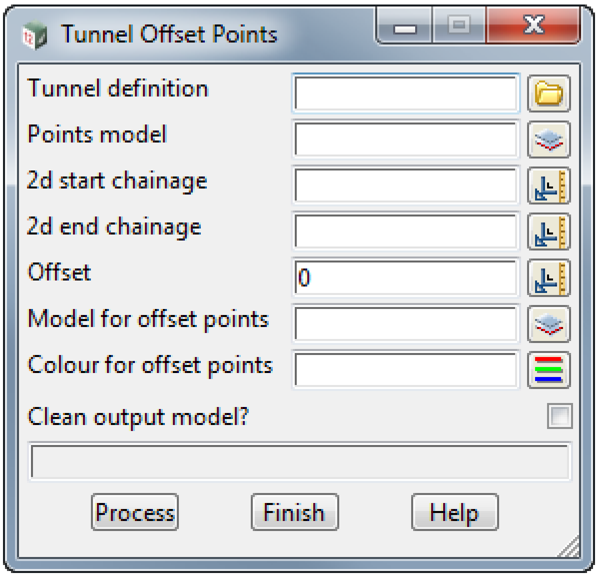
20.12.9 Tunnel Offset Points

Position of option on menu: **Design =>Tunnel-Structures =>Tunnel offset points**

Tunnel offset points takes points

The tunnel itself is defined in a ***Tunnel File*** that has been created in the **Create/Edit Tunnel File** panel. See 20.12.1 Creating/Editing a Tunnel Definition File.

Selecting **Tunnel offset points** brings up the **Tunnel Offset Points** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tunnel definition	tunnel file box		available 12d_tunnel files
the tunnel file containing all the definitions for the tunnel. That is, the tunnel centreline, the tunnel profiles and which profiles and how often they are applied along the centreline of the tunnel.			
Points model	model box		available models
the model of points that are approximately on the tunnel surface.			
2d start/end chainage	measure boxes		
Offset	measure box		
Model for offset points	model box		available models
Colour for o ffset points	colour box		
Clean output model?	tick box		
Process	button		

20.12.10 Definition of the PRO and PRA definition files

Profiles (.PRO)

```
// Sample tunnel profile
//
// Lines starting with // are interpreted as comments and ignored
//
// Format
//
// <profile name> 8 characters or less
// <element name> <offset> <level diff> <radius> for the radius
// negative is left and 0=straight
// .
// .
// <element name> <offset> <level diff> <radius>
// <element name> <offset> <level diff> terminates when there is no /
/ radius
//
// Each profile MUST have the same number of elements
// and the name of the elements must match.
//
// Generally speaking profiles are created anti-clockwise
//
// This profile is circular with dummy elements E2 and E5
// They transition into a straight element in the next profile
//
PROF_1
E1 0 1 -1
E2 1 2 0
E3 1 2 -1
E4 0 3 -1
E5 -1 2 0
E6 -1 2 -1
E7 0 1
// This profile has 4 metre vertical sides so transforms
// PROF_1 into an oval shape
//
PROF_2
E1 0 1 -1
E2 1 2 0
E3 1 6 -1
E4 0 7 -1
E5 -1 6 0
E6 -1 2 -1
E7 0 1
```

Profile Assignment File (.PRA)

```
// Sample tunnel profile assignment
//
// Lines starting with // are interpreted as comments and ignored
//
// Format
//
// <chainage> <profile name>
```

```
// <chainage> <profile name>
// .
// .
// <chainage> <profile name>
//
1000 PROF_1
1100 PROF_2
1400 PROF_2
1432.728 PROF_1
```

20.13 Overlay

Position of menu: **Design =>Overlay**

The overlay options are for the design and optimisation of pavement overlay design.

The **Overlay** options is available to all users with the **Detailed Alignment Design** and **Volumes** modules.

The **Optimal Overlay** is under development and has not been released.

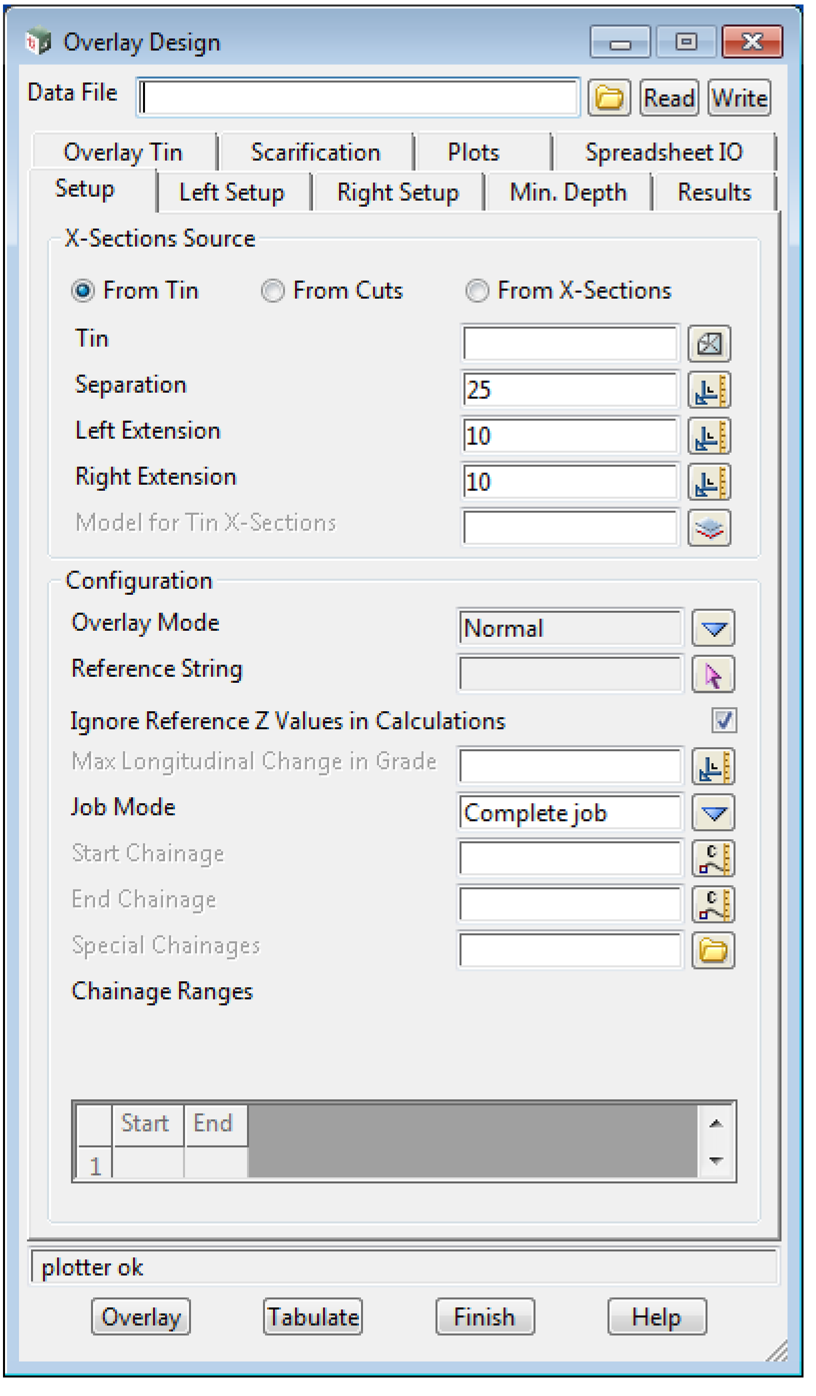
The **Overlay** walk-right menu is

For the option *Overlay*, please continue to the section 20.13.1 Overlay Design

20.13.1 Overlay Design

Position of option on menu: **Design =>Overlay =>Overlay design**

Selecting **Overlay** brings up the **Overlay Design** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data file	file box		*.ovd files

file with all the settings used in the Overlay panel. Used for storing information between runs.

Read	button		

read in a data file of settings for the panel

Write button

write out a data file of the settings in the panel

Buttons at bottom of panel

Overlay button

run the overlay.

Setup

Radio buttons

radio buttons to define how cross sections for existing road are produced

From Tin radio button

*if **selected**, the road x-sections are created by taking sections though a tin. The following fields are displayed*

Tin tin box available tins

sections are created through this tin to represent the existing road

Separation input

chainage distance between cross sections

Left/Right extension input

Left/Right distance to define the limits of the cross sections

Model for tin x-sections model box available models

model for the created cross sections

From Cuts radio button

*if **selected**, the road x-sections are created by taking cuts through strings. The following fields are displayed*

Strings model model box available models

sections are created by taking cuts through the strings in this model

Separation input

chainage distance between cross sections

Model for cuts model box available models

model for the created cross sections

From X-sections radio button

*if **selected**, the road x-sections already exist. The following fields are displayed*

X-sections model model box available models

existing model of x-sections to use for the road

Overlay mode choice box Normal Normal, Use Crossfalls Only
Use Reference Z Values and Crossfalls only

*if **Normal**, at each section the left/right z-value is calculated so that the left/right crossfall and minimum overlay is obeyed. The z-value for the section is the maximum of the calculated z-values for the left and right.*

*If **User Crossfalls only**, the crossfalls from the Left and Right Setups tabs are used and no minimum* *overlay applied.*

*If **Reference Z values and Crossfalls only**, no overlay calculations are performed and the crossfalls in the Left and Rights Setups tabs are applied to the actual z-values on the reference string.*

Reference string string select box

select the string for the centreline of the road

Ignore reference z values in calculations tick box

if not ticked, when the z-value is calculated at each section obeying just the minimum overlay criteria then use the maximum of the and the calculated z-value and the reference z-value at that section.

If ticked, use the calculated z-value at each section.

Start chainage input

chainage to start the overlay calculations

End chainage input

chainage to end the overlay calculations

Special chainages file box

special chainage file to define additional cross sections for overlay processing

Left/Right Setup tab

setup for the left/right side sections

Radio buttons

radio buttons to define how cross sections for existing road are produced

By offset radio button

*if **selected**, the width of the left/right cross sections is given in the **Offset** field*

Offset input

left/right offset distance to define the edge of the left/fight section

By String radio button

*if **selected**, the width of the left/right cross section is defined by a selected **Edge** string minus the **Cut Back/Extension** value. The following fields are displayed*

Edge string string select

select the left/right edge string

Cut Back/Extension input

value to subtract from the distance to the edge string to define the left/right section

*When **By String** is selected, the Chainage-Minimum X-Fall grid can also be automatically filled in by using the crossfall between the **Road Crown** and the **Edge String**.*

Road crown string select

*select the road crown the use for calculating the cross fall to the **Edge string***

Only critical points tick box

*if ticked, the crossfall between the **Road crown** and the **Edge string** is only calculated at the tangent points of the Reference string**.***

*If not ticked, the crossfall between the **Road crown** and the **Edge string** is calculated at the regular* *interval given by **Xfall Load Interval** and also at the tangent points of the Reference string.*

Xfall load interval input

regular interval to calculate crossfall

Load Xfalls from Strings button

*when selected, the **Chainage-Minimum X-Fall** grid is automatically filled in by using the crossfall between the **Road Crown** and the **Edge String***

By name radio button

*if **selected**, the width of the left/right cross section is defined by the name of a given string minus the **Cut Back/Extension** value. The following fields are displayed*

Name

name of the string to select as the left/right edge string

Cut Back/Extension input

value to subtract from the distance to the edge string to define the left/right section

Default Xfall input

*if **non blank**, the default crossfall to use for the left/right cross section.*

*If **blank**, the **Chainage-Minimum X-fall** grid is used.*

Chainage- Minimum Xfall grid

*this grid is only used if the **Default Xfall** is **blank***

Chainage input

chainage to use the Min xfall

Min X-fall input

minimum X-fall to use at that chainage

Min. Depth tab

there can be one minimum depth value for the entire job or the minimum depth can vary between chainages

Default depth input

*if **non blank**, the default minimum depth to use for the left and right cross section.*

*If **blank**, the **Chainage-Minimum Depth** grid is used.*

Chainage- Minimum Depth grid

*this grid is only used if the **Default depth** is **blank***

Chainage input

chainage to start the minimum depth

Min Depth input

*minimum depth to start at this chainage and either go to the next chainage **or** if there is a **Min Depth 2**, interpolate to Min Depth 2 from this chainage to the next chainage*

Min Depth 2 input

minimum depth to end with at the next chainage

Results tab

Transfer Z-values to reference string tick box

if ticked, the calculated z-values are used to update the z-values on the reference string

Clean models before overlay tick box

if ticked, the models of results are cleaned before the overlay option is run

Model for overlay x-sections model box available models

model for the created overlay x-sections

Model for overlay strings model box available models

model for the created overlay strings

Colour colour box available colours

colour for the created overlay strings

Model for xfall strings model box available models

*if **non blank**, left and right strings with z-value equal to the left/right crossfall are created and placed in this model*

Model for min/max points model box available models

*if **non blank**, points for the minimum and the maximum of the calculated z-values for the left and right sections are created and placed in this model*

X-sections depth range file file box available depth range files

a depth range file used for colouring x-sections by the depth between the overlay and the road tin

Overlay report file box *.rpt files

report on the overlay

Volumes report file box *.rpt files

volumes of overlay report

Write volumes as CSV tick box

if ticked, the volume report is written as a CSV file (for Excel(

Overlay Tin tab

*the **Overlay Tin** tab controls the creation of an overlay tin and depth polygons*

Create overlay tin tick box

if ticked, the fields in this tab are used to create an overlay tin

Tin name tin box available tins

name for the tin created from the overlay strings and sections

Tin colour colour box available colours

colour of the overlay tin

Model for tin model box available models

model for the overlay tin

Create depth polygons tick box

if ticked, depth polygons are created between the road tin and the overlay tin

Depth range file file box *.drf files

depth range file used when creating polygons

Model for depth polygons model box available models *model for the depth polygons*

Scarification tab

*the **Scarification** tab controls any scarification to be applied to the road sections before calculating overlays*

Perform scarification tick box

if ticked, scarification is performed on the road sections before overlay is calculated

Model for scarified x-sections model box available models

*if **non blank**, road x-sections after scarification are created and placed in this model*

Colour colour box available colours

colour of the scarified sections

Clean scarification x-sections model tick box

if ticked, the clean the model of scarified sections before running calculations

Default scarification depth input

*if **non blank**, the default scarification depth to use for road cross section.*

*If **blank**, the **Overlay-Depth** grid is used.*

Chainage- Minimum Depth grid

*this grid is only used if the **Default scarification depth** is **blank***

Chainage input

chainage to start the scarification depth

Depth input

*scarification depth to start at this chainage and either go to the next chainage **or** if there is a **Depth 2**, interpolate to Min Depth 2 from this chainage to the next chainage*

Depth 2 input

scarification depth to end with at the next chainage

Plots tab

Generate long section plots tick box

if ticked, long section plots are created

Long section PPF file box *.lplotppf files

binary ppf file to use for the long section plots

Plotter type plotter box model available plotters

type of plotter to use for the long section plots

Plot stem

the name to use for the plots - a number will be added when more than one page is produced

Clean plot model beforehand tick box

*if ticked and the **plotter type is model**, the plot models are cleaned before the plots are created*

Generate cross section plots tick box

if ticked, cross section plots are created

Cross section PPF file box *.lplotppf files

binary ppf file to use for the cross section plots

Plotter type plotter box model available plotters

type of plotter to use for the cross section plots

Plot stem

the name to use for the plots - a number will be added when more than one page is produced

Clean plot model beforehand tick box

*if ticked and the **plotter type is model**, the plot models are cleaned before the plots are created*

20.14 X-Sections

Position of menu: **Design =>X-Sections**

Cross sections can be created in a variety of ways including

(a) sections through triangulations created with the options in the menu **Tins =>Sections.**

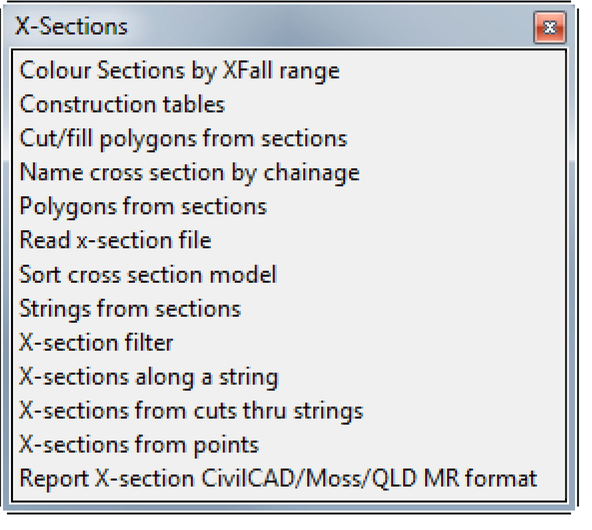
(b) sections generated by cutting through strings with the options **Utilities =>A-G =>Cuts.**

(c) design sections generated from options on the menu **Design =Apply**

(d) sections generated by the end area volumes options **Design =>Volumes =>End area**

The x-sections menu contains options to read cross-sections in from a file and for working with cross sections in a variety of way.

Walking right on **X-Sections** bring up the **X-Sections** menu:



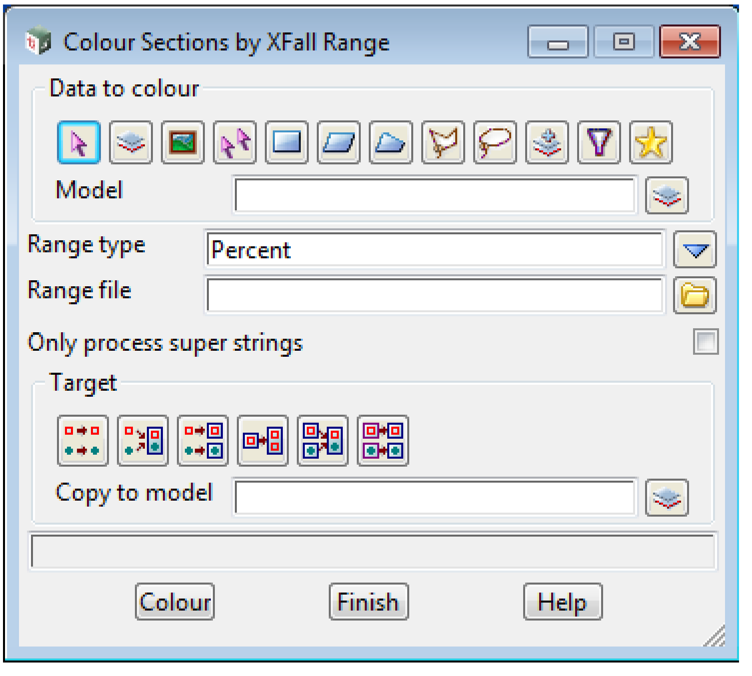
For Colour X-Sections by Xfall range go to	20.14.1 Colour Sections by XFall Range
Construction tables	20.14.2 Construction Tables
Cut/fill polygons from sections	20.14.3 Cut/Fill Polygons from Sections
Name cross sections by chainage	20.14.4 Name Cross Sections by Chainage
. *Polygons from sections*	20.14.5 Polygons from Sections
Read x-section file	20.14.6 Read X-Section Data
Sort cross-section mode	20.14.7 Sort X-Sections in a Model
Strings from sections	20.14.8 Strings from Sections
X-section filter	20.14.10 X Section Filter (Sections at Even Chainages)
X-sections along a string	16.15.4 X-Sections
X-sections from cuts thru strings	28.9.13.2 Cuts by Centreline
X-sections from points	20.14.9 Sections from Points
Report X-section CivilCAD/Moss/QLD MR format	27.11.12 X-Sections Report

20.14.1 Colour Sections by XFall Range

Position of option on menu: **Design =>X-Sections =>Colour X-Sections by Xfall range**
Utilities =>A-G =>Colour X-Sections by Xfall range

This panel will colour the segments of a set of x-sections, based on their x-fall.

Selecting **Colour X-Sections by Xfall range** displays the **Colour Sections by XFall Range** panel.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Data to colour

the set of x-sections to colour.

Data source type Model

data selection type - for a full description go to 4.19.3 Data Source.

Data source input

source of data to be processed - for a full description go to 4.19.3 Data Source.

Range type choice box Percent, 1v in, Degrees

the type of range file to use

Range file file

the range file to use for colouring

Only process super strings tick box

whether or not to ignore non super strings

Target

the target for the colour x-sections

Colour button

colours the x-sections

20.14.2 Construction Tables

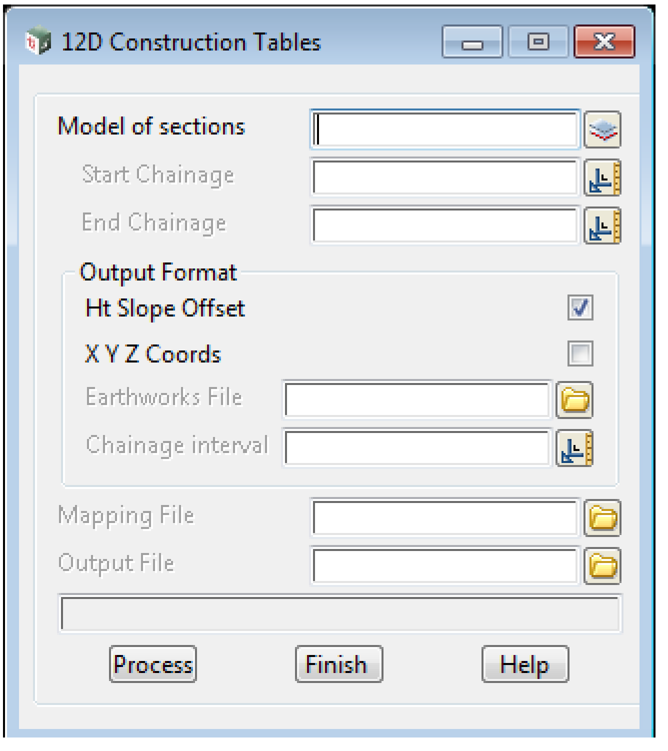
Position of option on menu: Design =>X-Sections =>Construction tables

This option is used to create construction tables from design cross section model.

Output can be in Ht, Slope or xfall, Offset or X,Y,Z.

A volume report file from the Apply MTF road function can be used if Earthworks is required.

Selecting **Construction tables** brings up the **12D Construction Tables** panel:



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Model of Sections	input		

model name for existing sections

Start Chainage	real		

if blank, means optional and the extents of the sections are used instead

End chainage	real		

if blank, means optional and the extents of the sections are used instead

Output Format

Height Slope Offset	tick	true	
XYZ Coords	tick	false	
Earthworks File	file		

*if non-blank, file from **Apply MTF** is used*

Chainage Interval	real		

if non-blank, value must be a multiple of chainages reported in the earthworks file

Map File	file		

Please refer to the extra help file for details

Output File file

Please refer to the extra help file for details

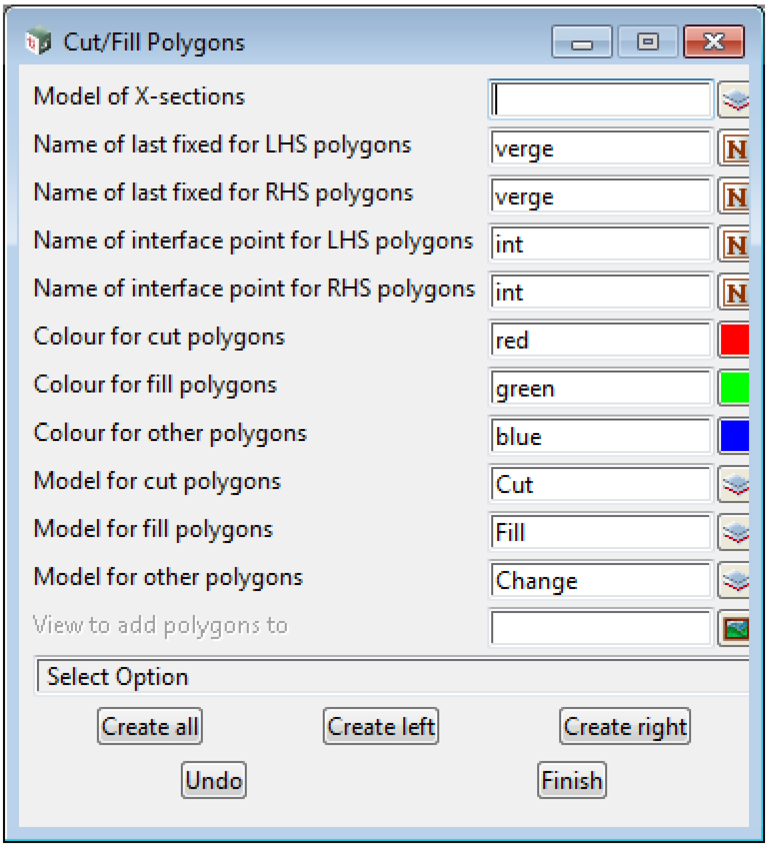
Process button

runs the option

20.14.3 Cut/Fill Polygons from Sections

Position of option on menu: Design =>X-Sections =>Cut/fill polygons from sections

This option takes cross sections and by using the user-supplied names of points on the cross section, cut and fill polygons for the strings joining the named points in successive cross sections.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model of x-sections	model box		available models

model containing the cross sections used to define the cut and fill polygons.

Name of last fixed for LHS polygons	input	verge	

Name of the cross section point to be the inner point of the left hand side polygons.

Name of last fixed for RHS polygons	input	verge	

name of the cross section point to be the inner point of the right hand side polygons.

Name of interface point for LHS polygons	input	int	

name of the cross section point to be the outer point of the left hand side polygons.

Name of interface point for RHS polygons	input	int	

name of the cross section point to be the outer point of the right hand side polygons.

Colour for cut polygons	input	red	available colours

colour for the cut polygons.

Colour for fill polygons	input	green	available colours

colour for the fill polygons.

Colour for other polygons input blue available colours

colour for the polygons that are neither cut or fill – they are either flat or change from cut to fill.

Model for cut polygons input cut available models

model for the cut polygons.

Model for fill polygons input fill available models

model for the fill polygons.

Model for other polygons input other available models

model for the polygons that are neither cut or fill – they are either flat or change from cut to fill.

View to add polygons input available views

if non-blank, the models of polygons are added to the view.

Create all button

run the option and create left and right polygons.

Create left button

run the option and only create the left polygons.

Create right button

run the option and only create the right polygons.

Undo button

undo the last set of polygons created whilst the panel has been up.

Finish button

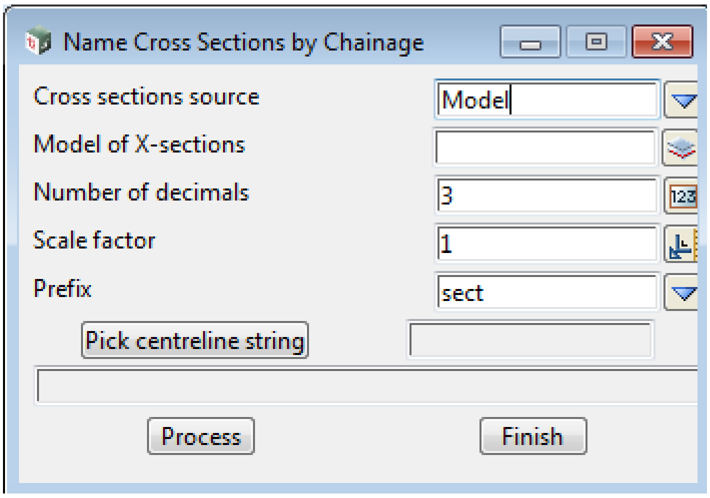
end the option and remove the panel. The undo capability for the polygons is terminated.

20.14.4 Name Cross Sections by Chainage

Position of option on menu: **Design =>X-Sections =>Name cross sections by chainage**

This option names cross-sections with the chainage from a selected string.

Note that the chainage from the string that the cross section comes from is in the name of the cross section so this option effectively gives the chainage to the cross section.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Cross sections source	choice box	model	string, model, view

type of data source.

Model/String/View of x-sections source box

source of the cross section data.

Number of decimals	input box	3

number of decimal places to use in the chainage label.

Scale factor	input box	1

factor to multiply the chainage by before creating the label.

Prefix	choice box	sect	sect, design, none

prefix for the chainage value - for cross-section options to work, this should be sect or design.

Pick centreline string output box

select the string that is used to find a chainage for the cross sections.

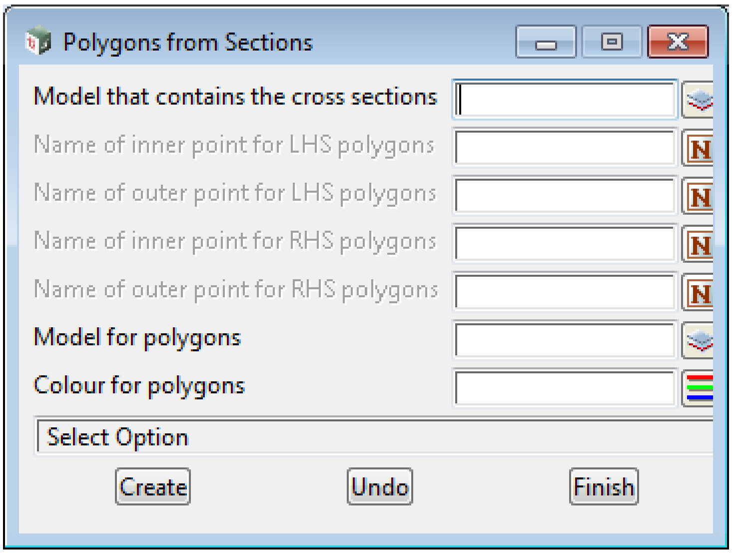
Process button

run the option.

20.14.5 Polygons from Sections

Position of option on menu: **Design =>X-Sections =>Polygons from sections**

Option to take a model of cross-sections and create polygons between specified points on the cross-sections.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Model that contains the cross sections model box

model of cross sections

Name of first/last point for lhs polygons input box

name of the first/last point on left hand side of the cross-section to creating polygon for.

Name of first/last point for rhs polygons input box

name of the first/last point on right hand side of the cross-section to creating polygon for.

Model/colour for polygons input box

model/colour of the created polygons.

Create button

create polygons between the first and last points on the left and right side of the cross-sections.

Undo button

undo the last set of polygons created whilst the panel was up.

20.14.6 Read X-Section Data

Position of option on menu: **Design =>X-Sections =>Read x-section file**

This option is used to read in cross section data from a file and convert it to x-section strings (4d strings).

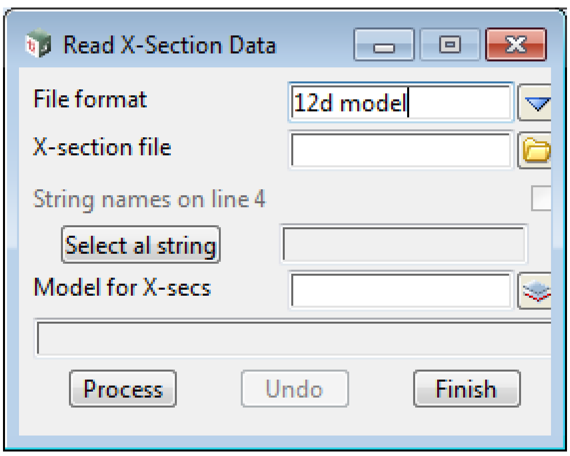
The data in the file is only centre-line chainage values, offset, heights and text (Code) so an alignment string needs to be selected to give an (x,y) position for the centre-line chainage and a direction (at right angles to the alignment string at the chainage) to convert the (offset, height) values into (x,y,z) points on a string.

NOTE - after reading in the x-section data and hence creating 4d strings, the

Design =>X-Sections =>Strings from sections

option can be run to create 3d strings by joining points with the same Code from adjacent x-sections. The created 3d strings are given the string name of the Code that they joined.

Selecting **Read x-section file** brings up the **Read X-Section Data** panel:



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
File format	input box	***12d Model***	12d Model, Alg, Moss, Chainage/Offset/Height/ Text

format of the x-section data in the file.

12d Model Format

The 12d Model *format is a simple text file format with the text CH followed by one or more spaces and then the alignment chainage (centre line chainage) of the x-section, and then on the following lines, the Offset value followed by one or more spaces, the height at that offset followed by one or more spaces, and then the text (Code) for that point on the x-section. The x-section continues until the next Ch line. For example, the following defines two x-sections at chainage 10 and 20 respectively.*

ch 10

-3 5 A

0 6 "Second label"

4 7 C

ch 20

-3.5 5.5 A

0 6 D

4.5 7.5 ""

Note that if the Code includes a space, then it must be enclosed in quotes.For example "Second label". Also the Code is not optional - if there is no code, put "".

Chainage/Offset/Height/Text Format

The Chainage/Offset/Height/Tex**t** *format is a very simple text file format with each line containing the alignment chainage (centre line chainage) for the x-section followed by one or more spaces and then on the same line the Offset value followed by one or more spaces, the height at that offset followed by one or more spaces, and then optionally, the text (Code) for that point on the x-section. The x-section continues until the alignment chainage changes. For example, the following defines two x-sections at chainage 10 and 20 respectively.*

10 -3 5 A

10 0 6 "Second label"

10 4 7 C

20 -3.5 5.5 A

20 0 6 D

20 4.5 7.5 E

Note that if the Code includes a space, then it must be enclosed in quotes.For example "Second label". Also the Code is optional (whereas it was compulsory for the **12d Model** *format).*

Moss (MX) Format

Please refer to Moss (MX) documentation or ask the group who supplied the data.

Alg Format

Not documented.

X-section file input box

file of x-section data.

String names on line 4 tick box

ONLY used for the ALG format which is not documented

Select al string string select

select the string that is used to define the (x,y) position of the zero offset of the x-section and also the bearing for the x-section.

Model for x-secs model box available models

model for the created x-sections

Process button

read the file and create the x-sections

Undo button

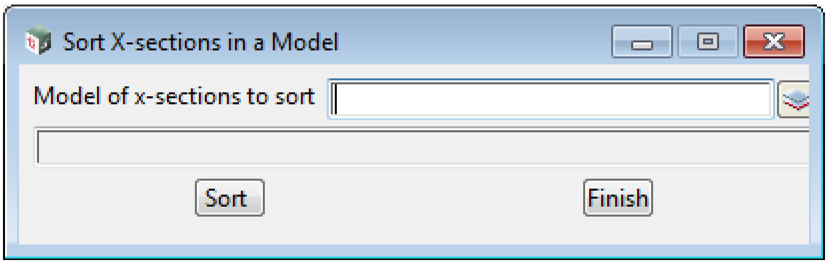
undo the last set of x-sections created since the panel has been active

20.14.7 Sort X-Sections in a Model

Position of option on menu: **Design =>X-Sections =>Sort cross section model**

This option is used to sort cross sections in a model so that they are in the order of the chainages embedded in the cross section name.

Selecting **Sort cross section model** brings up the **Sort X-Sections in a Model** panel:



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Model of x-sections	model box		available models

model containing the cross sections to be sorted into chainage order.

Sort button

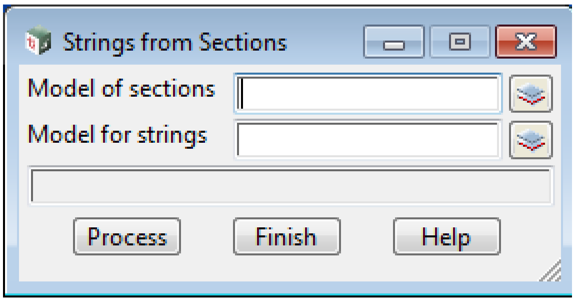
run the option.

20.14.8 Strings from Sections

Position of option on menu: **Design =>X-Sections =>Strings from sections**

The **strings from sections** option creates strings by joining the common named points on successive 4d strings (usually generated as sections).

Selecting **strings from sections** fires up the **strings from sections** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model of sections	input		available models

model of the 4d strings (sections) to try and join common points from to form strings.

Model for string	input		available models

model for the created strings to go to.

Process	button		

try and join points with the same name from successive sections to form strings.

20.14.9 Sections from Points

Position of option on menu: **Design =>X-Sections =>X-sections from points**

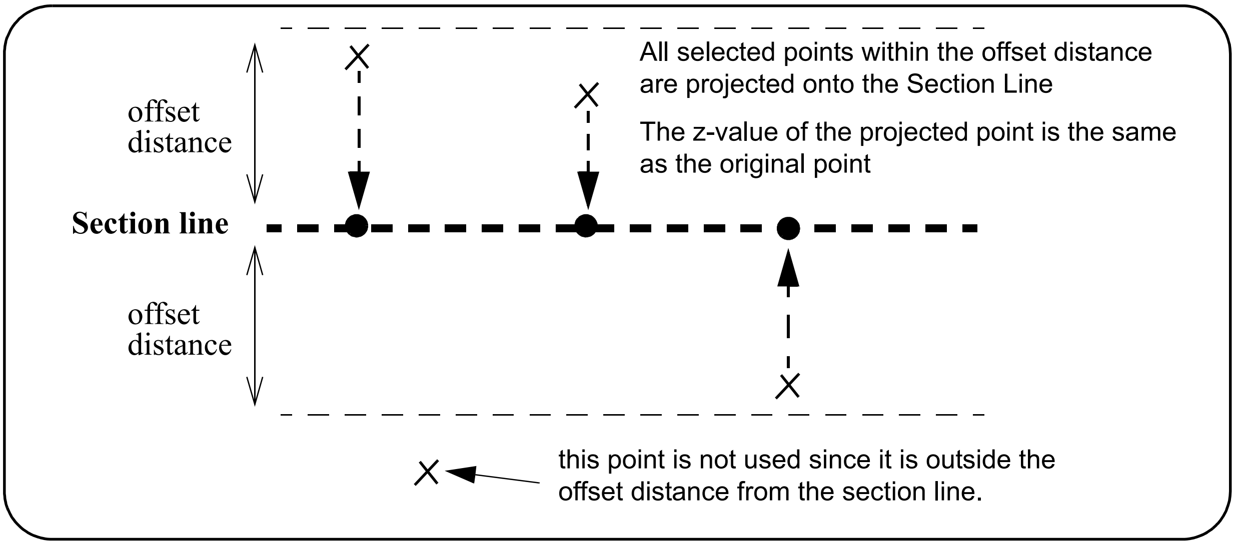
The **X-sections from points** option creates a cross section by first defining s section line and then dropping all selected points in a user defined offset distance from the section line, onto the section line.

(a) The z-value for the dropped point is the same as for the original point.

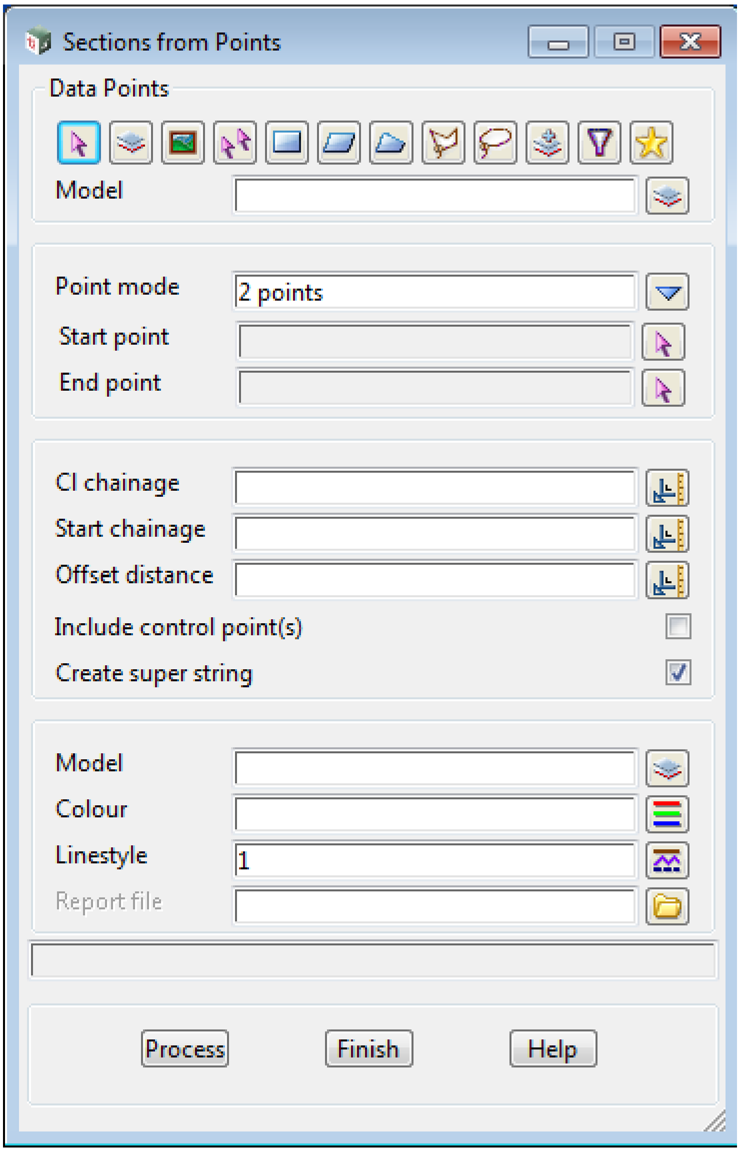
(b) The Point number (Vertex id) of the dropped point is the Point number (Vertex id) of the original point.

(c) The Vertex text of the dropped is the code (string name) of the original point.

The user specifies the start chainage for the created section and the value for the "centreline chainage" of the section.



Selecting **Sections from points** fires up the **Sections from points** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type			

data source type.

Data source

data source for points to create sections from.

Point mode	choice box	2 points	2 points, 1 point and bearing point on centreline

*if **2 points**, then the **Start point** and **End point** are selected. The section goes through the two points.*

*If **1 point and bearing**, then the **Start point** is selected and a **Bearing** given. The section goes through the start point with the given bearing.*

*If **point on centreline**, then a **Centreline** is selected and a **point on the centreline** selected. The section goes through the selected point on the centreline and is perpendicular to the selected centreline at the selected point.*

If Point mode is *2 points*:

Start point	string select	
End point	string select	

If Point mode is *1 point and bearing*:

Start point	string select	
Bearing	angle box	

If Point mode is *point on centreline*

Centreline string select

depends on **Point mode**.

Point on cl string select

depends on **Point mode**.

Cl chainage input

the chainage to give to the created section

Start chainage input

the start chainage to give to the created section.

Offset distance input

distance to search either side of the section for points to project onto the section

Include contrl points (s) tick box

if ticked, the selected start and end points are included as part of the section.

Create super string tick box

if ticked, then the created section is a super string.

Model model box available models

model for created sections.

Colour colour box available colours

colour for the created sections.

Linestyle linestyle box available linestyles

linestyle for the created sections.

Report file file box

a report.

Process button

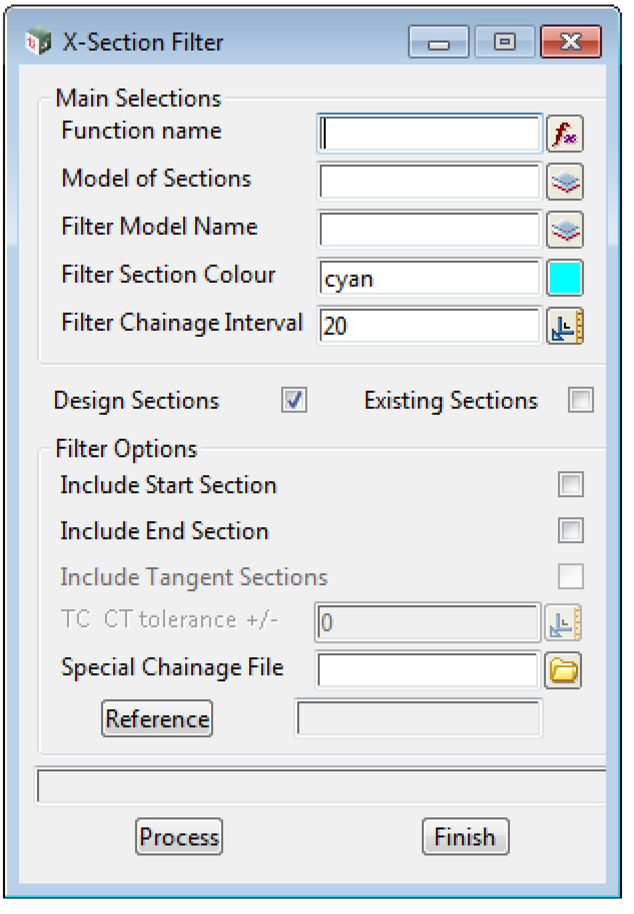
create a cross section from the selected points.

20.14.10 X Section Filter (Sections at Even Chainages)

Position of option on menu: **Design =>X-Sections =>X-section filter**

This panel selects user specified cross-sections from a model of cross-sections and copies them into another model. The cross sections must already exist - no new cross-sections are created.

It is a function so can be recalced when the cross-section model is changed.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Function name			

name of the function.

Model of sections	string select		

model of existing cross sections to select some cross sections from.

Filter model name	model box		

model for the selected cross sections

Filter section colour	colour box	cyan	

colour for the filtered cross sections

Filter chainage interval	input	20	

chainage interval to select cross sections. Sections are selected which are multiples of the ***filter chainage interval****.*

Design/Existing sections	radio button	design sections	

cross sections are 3d strings with the name

***Design** and the chainage of the cross-section (usually created by an Apply),*

or ***Sect** and the chainage of the cross-section (usually created by Tins =>Section =>X section)*

*If **Design sections** is selected, then sections starting with the name **Design** are selected.*

*If **Existing sections** is selected, then sections starting with the name **Sect** are selected.*

Include TC_CT sections tick box

if ticked, include cross sections at tangent points from the reference string.

Include start/end section tick box

include a cross section at the start/end chainage of the reference string

TC_CT tolerance +/- input box

*if **non-zero**, then regular cross sections (i.e. those selected by the filter chainage interval) are **not** included if they are closer than the given tolerance distance to a tangent point.*

Special chainage file input box

include cross sections with chainages given in the given special chainage file.

Reference output box

select reference string to use for the filter options requiring tangent points and start/end chainages.

Process button

copy the specified cross-sections to the Filtered Model.

20.14.10.1 X-Sections Along a String

Position of option on menu: **Design =>X-Sections =>X-Sections along a string**

This option has already been documented in **Tins => Sections=> X-Sections**

For the option ***X-Sections along a string***, please continue to the section 16.15.4 X-Sections.

20.14.10.2 X-Sections from Cuts Through Strings

Position of option on menu: **Design =>X-Sections =>X-Sections from cuts thru strings**

This option has already been documented in **Utilities =>A-G =>Cuts =>by centreline**

For more information please go to the section 28.9.13.2 Cuts by Centreline.

20.15 Check/Clash

Position of menu: **Design =>Check/clash**
Utilities =>A-G =>Check/clash

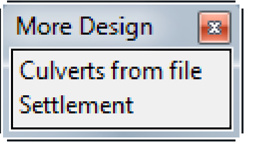
See 28.9.9 Check/Clash in 28 Utilities.

20.16 More Design

Position of menu: **Design =>More**

The **More Design** menu contains extra design options.

The **More Design** walk-right menu is

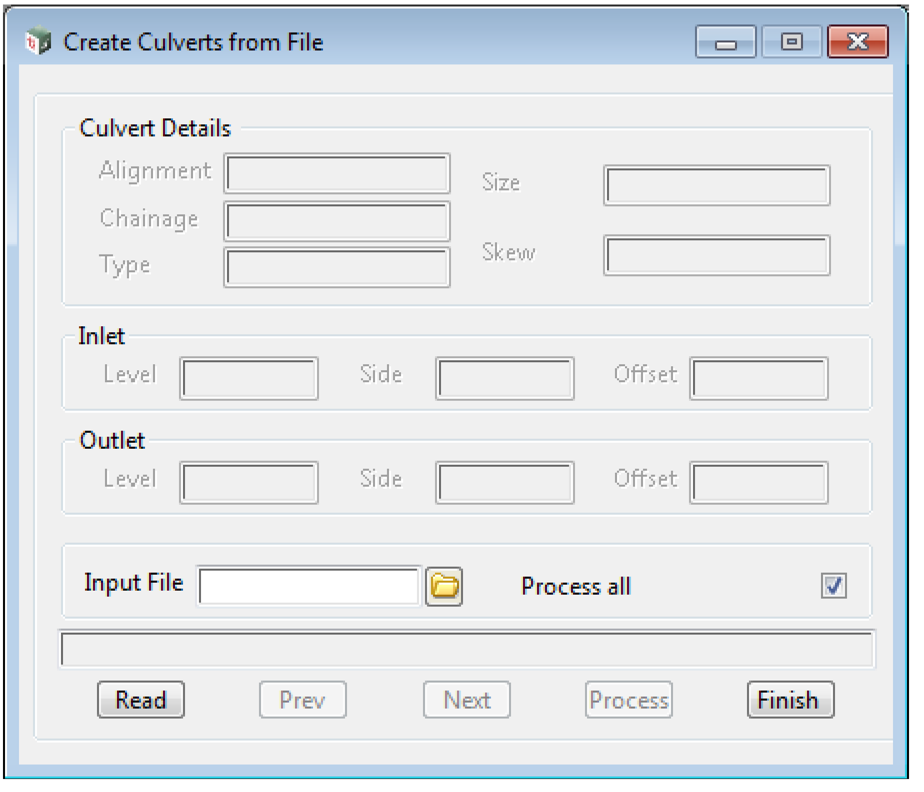


For the option *Culverts from file*, go to 20.16.1 Culverts from File.
Settlement 20.16.2 Settlement

20.16.1 Culverts from File

Position of option on menu: **Design =>More =>Culverts from file**

This option is currently under development.



20.16.2 Settlement

Position of option on menu: **Design =>More =>Settlement**

This option uses a tin of settlement values to adjust the z-values of strings.

The idea behind the option is that a number of string may have been surveyed as well as a number of control points. At a later date, the z-value at the control points is again measured and there is a major difference in the two z-value at each control point due to say settlement (subsidence) of the ground.

A tin of the ***difference*** of the z-values at each control point is called the ***settlement*** tin. The tin is not normally a constant value because *differential settlement* usually occurs. That is, the settlement is *different at each point over the surface*.

The Settlement option can adjust ***all*** the z-values of the selected data by the amount in the settlement tin.

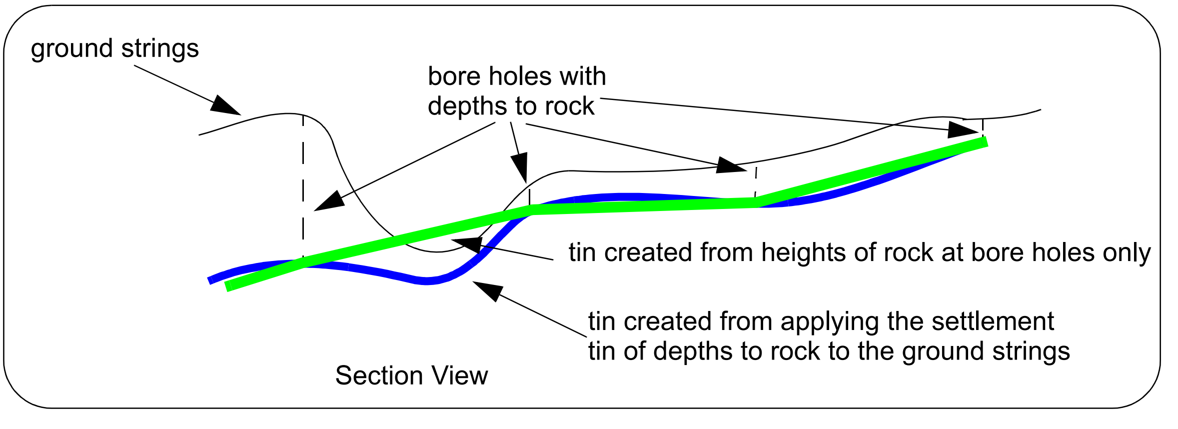
It can be applied to just to the vertices of the strings (mode ***relative points***), or to the vertices and the segments of the strings (mode ***relative string***). In ***relative string*** mode, where ever a segment crosses a triangle of the settlement tin, a new vertex is inserted into the string (with initial z-value interpolated from the adjacent string vertices) and the settlement tin value applied to the new vertex.

The ***settlement tin*** values can be added to, or subtracted from, the string values.

One interesting use of the *Settlement* option is when the settlement tin is constructed from the ***depths*** from the surface to a rock or soil strata (as given from ***bore hole*** logs).

The surface of the rock could be constructed by simply calculating the height of the rock at each bore hole and triangulating the heights. The original surface would have no affect on such a rock surface.

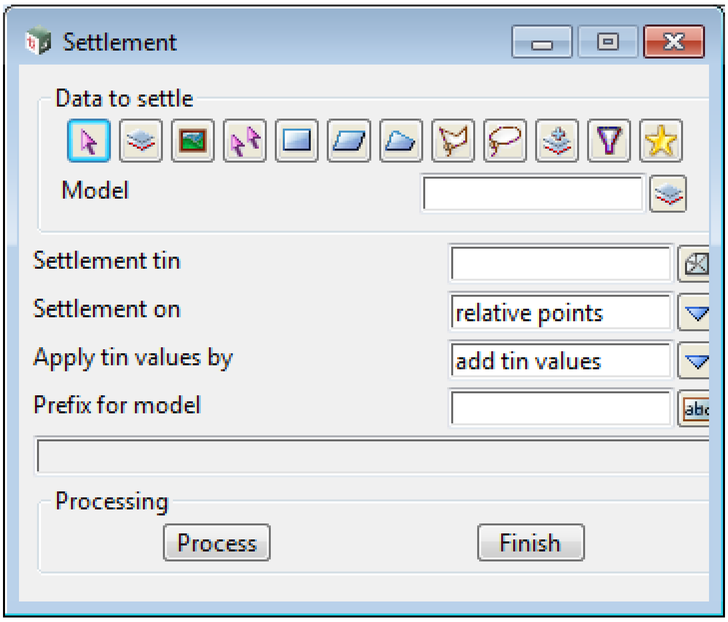
However by applying the ***Settlement*** option to the ground data, using the depths as the settlement tin, and then triangulating the resultant strings to give the rock surface, undulations in the ground data will also be reflected in the rock surface. This is what usually happens when the earth is folded and both surfaces fold in a similar manner.



The Settlement option can also be used ***without*** creating a tin of the difference of the z-values at each control point. A tin can simply be created from ***the new z-values*** of the control points.

In ***absolute points*** mode, each vertex of the strings are given the z-value from the tin. This is the same as draping just the vertices of strings onto a tin (see 16.10.1 Drape)

In ***absolute string*** mode, not only do the string vertices take their z-values from the tin, but where ever a segment crosses a triangle of the tin, a new vertex is inserted into the string and the z-value for the new vertex is taken from the tin. This is the same as a draping the strings onto the tin (see 16.10.1 Drape).



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data to settle		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Settlement tin			available tins

tin of values (settlement values) to be used to reduce/increase the z-values of strings by.

Settlement on	choice box	relative points	relative points relative strings absolute points absolute strings

if **Settlement on is:**

relative points *then for each point of a string, the value of the settlement string at that (x,y) position is added/subtracted from the z value at the point.*

relative string *then for each point of a string, the value of the settlement string at that (x,y) position is added/subtracted from the z value at the point plus for each visible segment joining points of the strings then if the segment crosses any edges of triangles from the settlement tin, a new point is inserted into the string which then has the settlement value applied to it (the initial z-value for the inserted point is interpolated from its adjacent points in the string).*

absolute points *then for each point of a string, the value of the settlement string at that (x,y) position is taken to be the z value at the point.*

absolute string *then for each point of a string, the value of the settlement string at that (x,y) position is taken to be the z value at the point plus for each visible segment joining points of the strings then if the segment crosses any edges of triangles from the settlement tin, a new point is inserted into the string which is given the settlement value at that point.*

Apply tin values by	choice box	add tin values	add tin values subtract tin values

if "add tin values", then the tin values are added to the string values.

If "subtract tin values", then the tin values are subtracted from the string values.

Prefix for models

prefix to apply to the name of each model in the data source to create new models for the processed data.

Process button

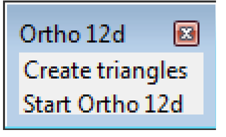
adjust the heights of the selected data.

20.17 Ortho 12d

Position of menu: **Design =>Ortho 12d**

The **Ortho 12d** menu contains the option to write out tins in a format suitable for the *Ortho 32* rectification software.

The **Ortho 12d** walk-right menu is



Continue to the next section 20.17.1 Create Triangles

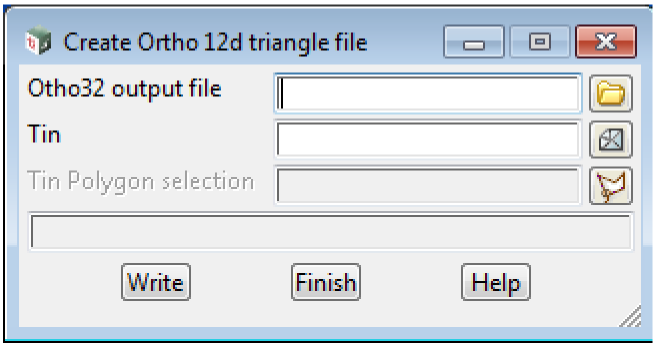
20.17.1 Create Triangles

Position of option on menu: **Design =>Ortho 12d =>Create triangles**

This option writes out a ***12d Model*** tin in a format suitable for the software ***Ortho12d*** software from Digital Mapping Systems (www.digmapsys.com).

Ortho12d is an image rectification package which can be used for ortho-rectifying photographs.

For more information on *Ortho12d*, please contact Ian Hall from Digital Mapping Systems on *ian.hall@digmapsys.com*



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

tin to write out in Ortho12d format

Tin polygon selection	Polygon select box		

if used, only triangles whose centroid is inside the polygon are written out.

Ortho12d output file	file box		*.T12 files

file to write the tin out to

Write	button		

write the tin out in the Ortho12d format

20.17.2 Moved/Reused Documentation

20.17.2.1 New Island Create (Function)

Position of option on menu: **Design =>Roads =>More =>New Island Create (Function)**

This option applies a kerb profile, either from the **12d** library or from a user defined file.

The template profile is applied to a Super Alignment, creating strings, and a triangulation.

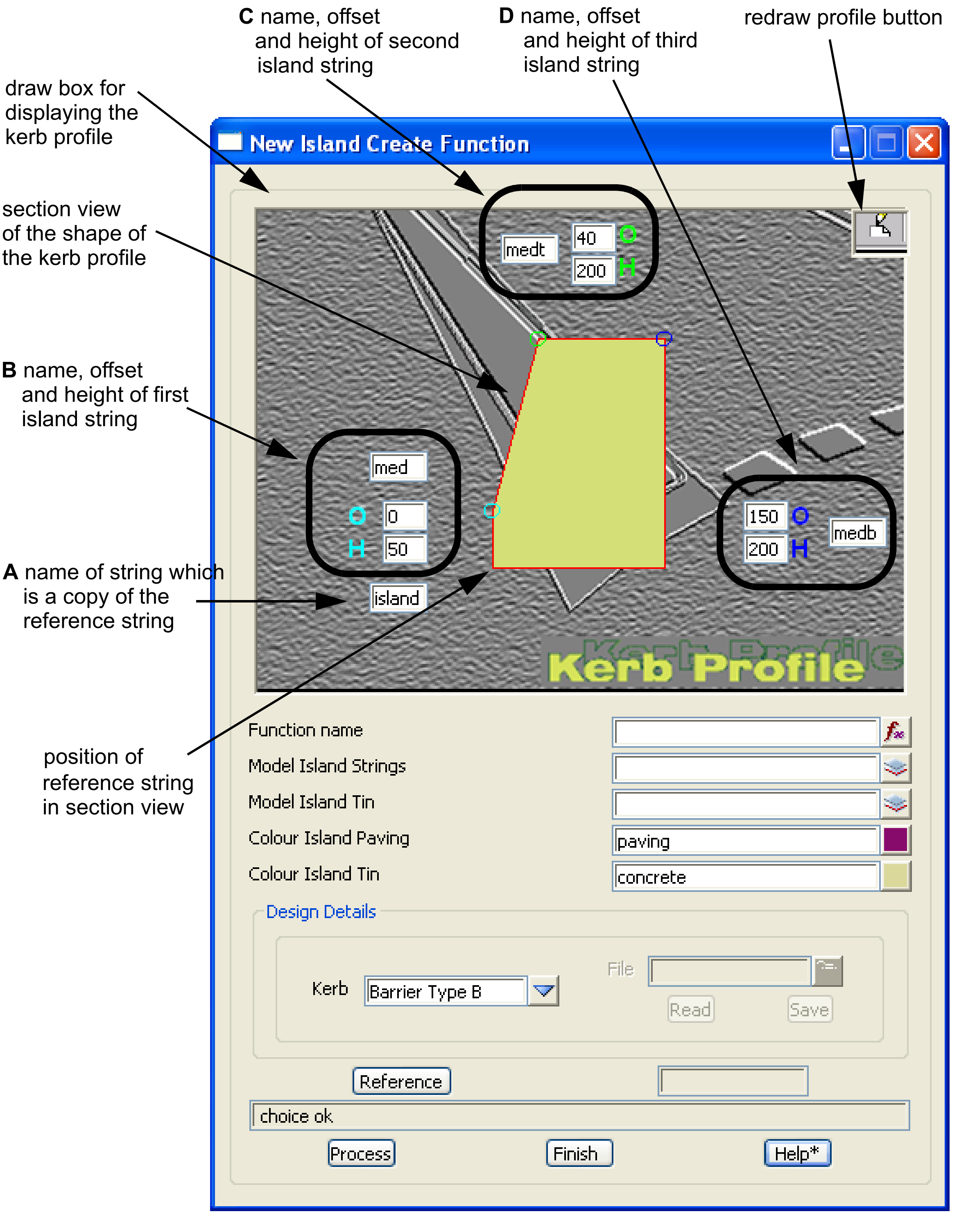
The option is saved as a function within **12d** and as a function can be re-run at any time if changes occur with the Super Alignment.

Some default kerb profiles are available under **Kerb** on the panel.

User defined kerbs are created by entering offset, height and string names in the fields in the draw box at the top of the panel, and then saved as a User Kerb Type file (*.ukt)

User Kerb Type files can then be written out, or read in as required.

On selecting the **New Island Create (Function)** option, the **New Island Create (Function)** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description Type Default Pop-Up

Function name function available functions

name of the Island Create function

Model Island Strings model box available models

model for the created island strings

Model Island Tin	model box		

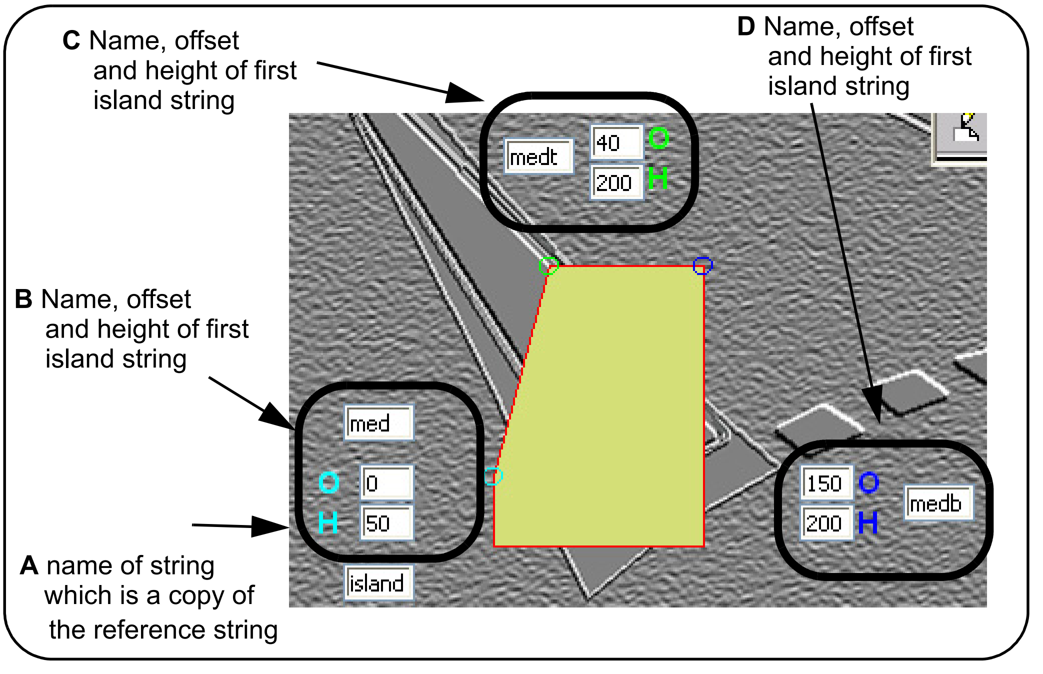
model for the tin of the created island

Colour Island Paving	input	paving	available colours

*if non-blank, then the colour is used to colour the island triangulation within innermost kerb string (**D**)*

Colour Island Tin	input	concrete	available colours

*if non-blank, then the colour is the colour is used to colour the island triangulation between the strings defined by **A** and **D**.*



In the drawing area:

A

a string is created which is a copy of the reference string and given this name. It is added to the **Model Island Strings** *model.*

B

name, offset and height from the reference string, of the first string of the island. It is added to the **Model Island Strings** *model. If offset and height are both zero, then the string is not created.*

C

name, offset and height from the reference string, of the second string of the island. It is added to the **Model Island Strings** *model. If offset and height are both zero, then the string is not created.*

D

name, offset and height from the reference string, of the third string of the island. It is added to the **Model Island Strings** *model. If offset and height are both zero, then the string is not created.*

Design Details

Kerb	choice box	Barrier Type A	Barrier Type A, Barrier Type B, Mountable, Semi - Mountable, User

selected kerb type is displayed in draw box above

If **Kerb** *is* ***User*** *then the* **File** *field and the* **Read** *and* **Save** *buttons are made active, and are used to read in an existing user defined Kerb type, or to create a new one.*

File file box files ending in.ukt

name of the user kerb type file (ukt) to read in or create

Read button

Reads in the user defined kerb definition from the ukt file given in the **File** *field and displays the profile and Offsets and Height in the draw box at the top of the Panel*

Save button

Writes out the kerb profile information (O and H values, and string names) as a user defined kerb, into the file given in the **File** *field.*

Reference select pick

Select the reference alignment which will have the kerb profile applied to it to create the island."pick with direction" in a clockwise manner as the kerb profile is always on the RHS.

Process button

runs the option which created the island string and the coloured island tin

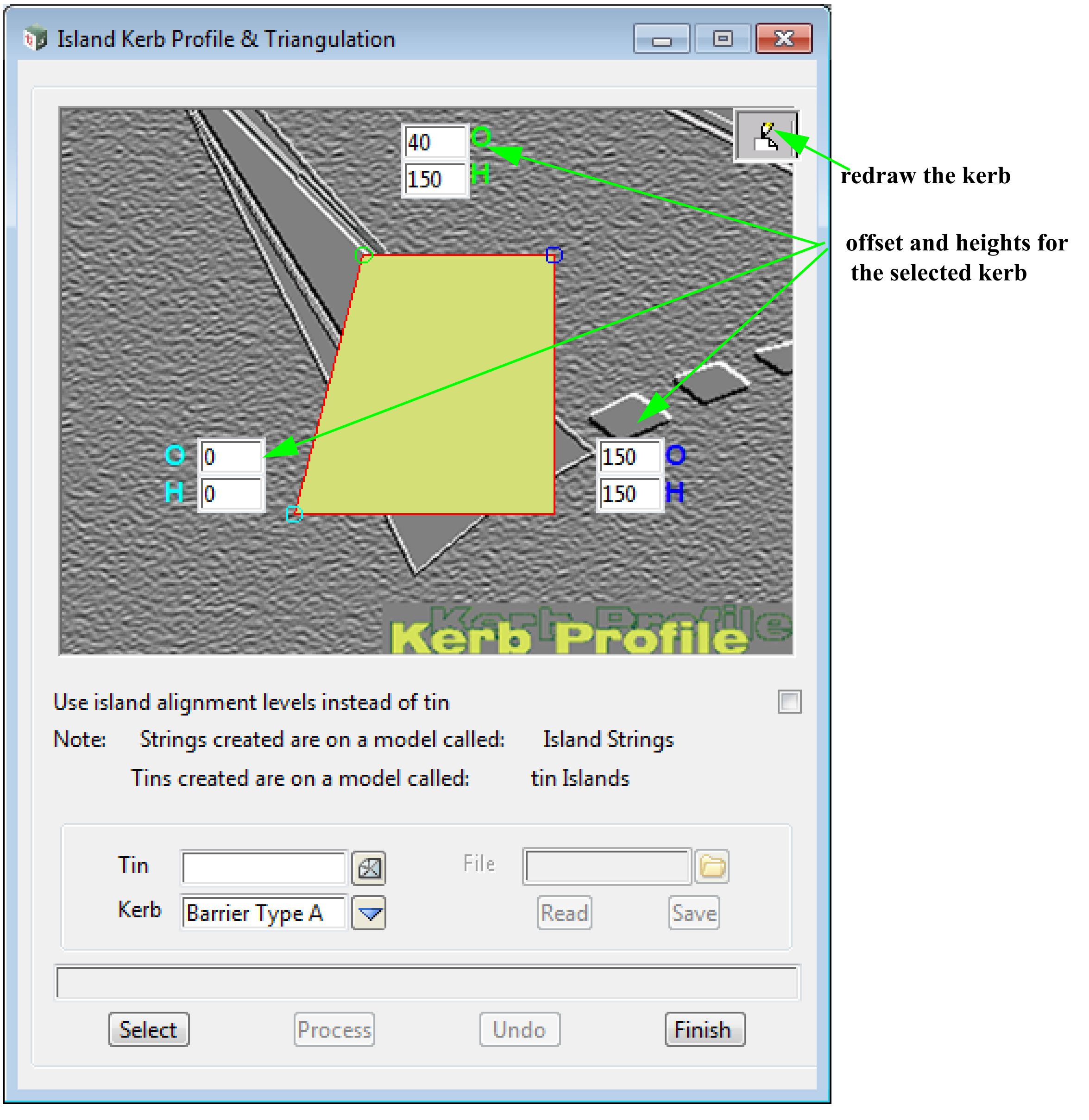
Note:

The New Island Create Function reads in the standard kerb profiles as templates and if required, creates a template from the ukt files. The function needs these templates because an apply is used (internally) to create the Island.

20.17.2.2 Traffic Island Profile and Triangulation Creation

Position of option on menu: **Design =>Roads =>More =>Island profile/tin**

This option creates strings by offsetting from a selected string using a kerb profile. The resultant strings are draped onto a given tin and then triangulated to form a traffic island.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Offset kerb face/top/back input box

offset for kerb face/top/back for the selected kerb type.

Height kerb face/top/back input box

height for kerb face/top/back.

Tin tin box available tins

triangulation to drape the traffic island strings onto.

Kerb type input box

type of kerb to apply to the reference string of the traffic island. If type **User** *is selected from the pop-up list, a file which defines the kerb profile can be selected and used.*

File file box *.ukt files

file to read/write the user defined kerb type to.

Read button

read the given **File** *to define a kerb type.*

Save button

save the kerb parameters to the given **File**.

Select button

select the string representing the traffic island.

Process button

create the traffic island strings and tin.

Undo button

undo the last traffic island created whilst the option was running.

20.17.2.3 Create Mass Haul String and Report

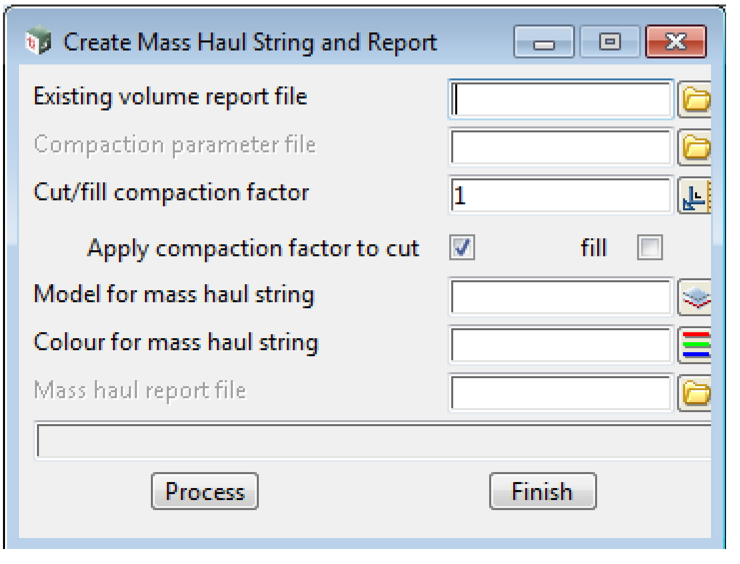
Position of option on menu: **Design =>Roads =>More =>Mass haul**

This option has been replaced by 20.8.8.3 Mass Haul String and Report.

This panel is used to read in a volumes report and optionally a cut/fill compaction parameter file and creates a mass haul string and a new volumes report with the compaction volumes.

The compaction parameter file simply contains the compaction factor for given chainage ranges. Either the cut volumes are multiplied by the compaction factor or the fill volumes are divided by the compaction factor, to give the compacted volumes in the new volumes report.

If the existing volume report contains an alignment string and that string can be found in the project then a new alignment string is created with the same horizontal geometry but with the accumulated volumes as the z-values.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Existing volume report file	file box		*.rpt files

name of the existing volumes report.

Field Description	Type	Defaults	Pop-Up
Cut/fill compaction factor	input	1.0	

cut/fill compaction factor for any chainages not in the compaction parameter file or if there is no file.

Field Description	Type	Defaults	Pop-Up
Compaction parameter file	input		*.mhf files

file containing compaction factors to be used.

Apply compaction factor to cut fill tick box

if cut, the cut volumes are multiplied by the compaction factors.
If fill, the fill volumes are divided by the compaction factors.

Model for mass haul string input

model for the created mass haul string.

Colour for mass haul string input

colour of the created mass haul string.

Mass haul report file input box

if non-blank, a mass haul volumes file is created.

Process button

run the option.

```
// Sample cut/fill compaction parameter file - the file ending
is .mhf
// All lines starting with // are comments and blank lines are
ignored
//
// A range is specified by a start and end chainage and a compaction
factor.
// A compaction factor of 1.0 means there is no compaction.
// A compaction factor of 0.9 means that 1.0 cubic metres of cut is
equivalent
// to 0.9 cubic metres of fill due to compaction.

// format is
// Start_ch    End_ch   Compaction_factor

      0         100        0.9
     100        500        1.0
     500        2000       1.1
```

21 Advanced Design

See

For some special definitions and information required for Modifiers and Templates, see

21.1 The Modifiers and Templates File - MTF

12d Model uses templates as a quick and easy method for defining design details along a string, and in the ***Apply MTF*** option, modifiers are introduced to handle the most complex design details such as sophisticated urban design including intersections and roundabouts.

12d Model uses templates as a quick and easy method for defining design details along a string.

So for simple work, the **Design=>Apply=> Apply** option cuts a left and right template into a surface to produce a design. For information on how Templates are defined in ***12d Model***, see 20.1 Templates in 12d Model

For more sophisticated design such as highways and complex urban design, intersections and roundabouts, templates are impractical and not an efficient solution. For these cases ***12d Model*** uses ***MTF modifiers*** to provide much finer control and provide all the power of string design without losing the ease of template design for simpler cases

The **Design =>Apply =>Apply MTF** option uses a **Modifiers and Templates File** or **MTF** file) applied to a Reference and Hinge string to tackle complex tasks and:

(a) run Hinge modifiers to adjust the Hinge string

(b) Insert Templates

(c) run Modifiers to adjust any links in the inserted templates, or create new links without needing a Template, and then make adjustments to these links.

(d) generate Sections, Boxing and Strings

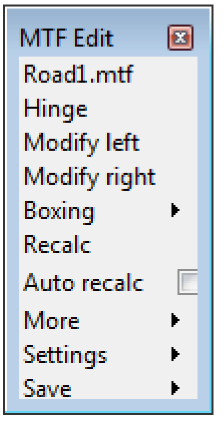
(a) defined shapes and produce **trimeshes** to give a solid representation of the road components.

For more definitions of these items used in MTF's, see 21.1.1 MTF Links, Points, Sections, Strings and Trimeshes.

Note - the *Alignment* module is required to use the *Apply MTF* option.

The **MTF** file used in an ***Apply MTF*** is created and edited **interactively** from within 12d Model.

An **MTF** can be created using the **Create MTF** panel (***Design =>MTF =>Create***) and edited using the **MTF Edit** menu.



Alternatively an **MTF** can be created and edited from within the **Apply MTF** option itself by:

once the **Apply MTF** option has been selected and the **Apply MTF Function** panel is on the screen, simply enter the name of the mtf file into the **MTF file** field and click on the *Folder* icon to bring up the pop-up menu and select **[Open]** from the pop-up.

If the **MTF** does not exist, then the **Create MTF** panel (with the MTF file name already filled in) is placed on the screen and the user simply selects the **Create** button. If the MTF file does exist, then it is loaded into the **MTF Edit** menu.

For full documentation on the **MTF Edit** menu, see 21.2 MTF Edit.

Note - although the *MTF file* can be edited with a text editor that **supports unicode**, it is easy to get the syntax wrong and create and invalid MTF file. So it is always best to use the interactive editor, **MTF Edit**.

For some special definitions and information required for Modifiers and Templates, see
21.1.1.1 MTF Links, Points and Strings
21.1.1.2 MTF Links and Layers
21.1.1.3 MTF Shapes and Trimeshes
21.1.2.1 Calculating Width, Height, Xfall or Slope from Original Definitions
21.1.2.2 Modify X, Hold Y in MTF Modifiers
21.1.2.3 New Value Usage

21.1.1 MTF Links, Points, Sections, Strings and Trimeshes

See

21.1.1.1 MTF Links, Points and Strings
21.1.1.2 MTF Links and Layers
21.1.1.3 MTF Shapes and Trimeshes

21.1.1.1 MTF Links, Points and Strings

A **Modifiers and Template File** (**MTF**), like a Template, is defined in terms of the links which go together to make up an **MTF section**.

As for a ***12d Template,*** an **MTF Section** consists of a **left hand side** ending at the **Hinge Point**, followed by a **right hand side**.

Each side if made up of a number of **MTF links** and each link is defined by any two of the three

(a) width of the link

(b) change in z-value from the beginning of the link to the end of the link. that is, the z-value at the end of the link minus the z-value at the start of the link. This is referred to as the **height** of the link.

(c) xfall or slope

and the links are connected sequentially to form a cross section.

Notice that the height is only a **relative** height and **not** a z-value.

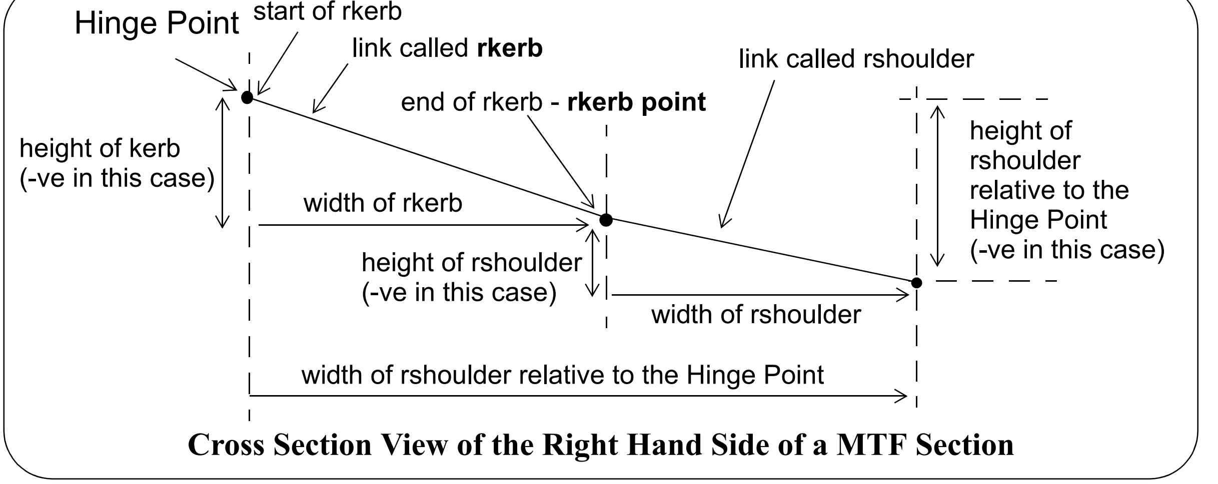
On the **left hand side**, the MTF links go from right to left.The vertex (point) at the end of the link is given the same name as the link. The far right of the left hand side is the Hinge Point. So on the left hand side, positive widths go to the left.

On the **right hand side**, the MTF links go from left to right. The vertex (point) at the end of the link is given the same name as the link. The far left of the right hand size is also the Hinge Point. So on the right hand side, positive widths go to the right.

Note that this definition of the MTF Section is given in terms of **MTF links** and each MTF Link is defined by a **relative width** and **relative height** to the **previous link**, and the left and right hand sides are positioned around a **Hinge Point**.

A **width from the Hinge point** and a **height above the Hinge point** can be calculated for each ***MTF Link point*** by accumulating the widths and heights of all the links from the Hinge point to the this link, and including this link. These values of the width and height for the Link Point are referred to as the **width relative to the Hinge Point** and **height relative to the Hinge Point**. Or sometimes ***absolute width*** *(or **absolute offset**)* and ***absolute height*** for short.

Note that this use of the word **absolute** must not to be confused with the absolute value of a number (which is always positive).



Cross Section View of the Right Hand Side of a MTF Section

Similarly for the Left Hand Side of a MTF Section except that the diagram goes to the left of the Hinge Point and positive widths are measure as **going to the left.**

Also if the **(x,y,z) coordinates** are known for the **Hinge point** and the **plan angle** that the

section makes through the Hinge point, then the **(x,y,z) coordinates** for each **Link point** can be calculated by accumulating all the relative widths and heights and applying them along the section line.

Each **MTF link** has a **name** and **colour** and it is best for all the links in an MTF section to have unique names.

The **point** at the **end** of the **MTF link** is given the **name** of the **link** and is called a **MTF Link Point,** or an **Link Point** or **MTF Point** for short

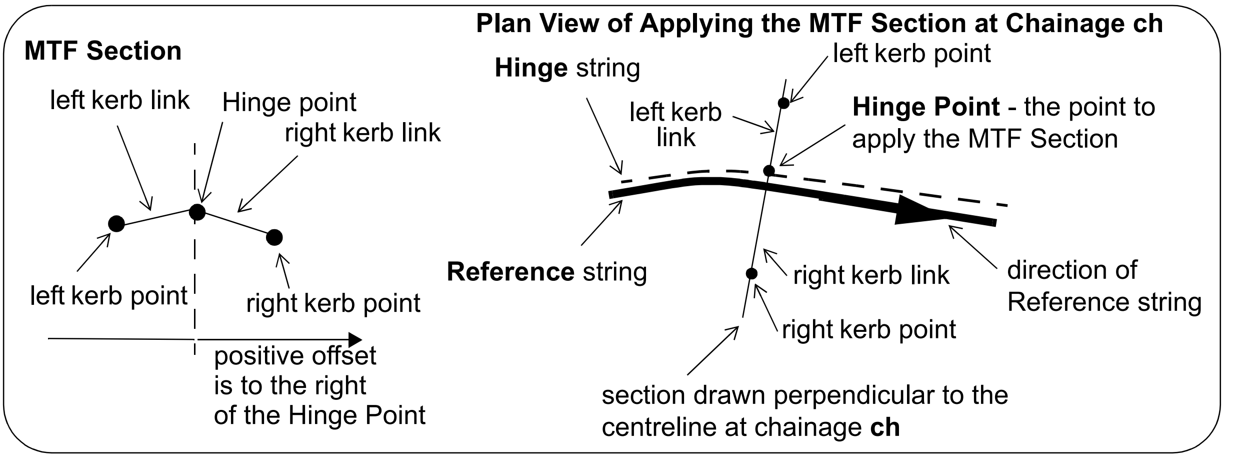
So on the left hand side of the Hinge Point, the MTF Link Point is at the left hand end of the link and on the right hand side of the Hinge Point, the MTF Link Point is at the right hand end of the link.

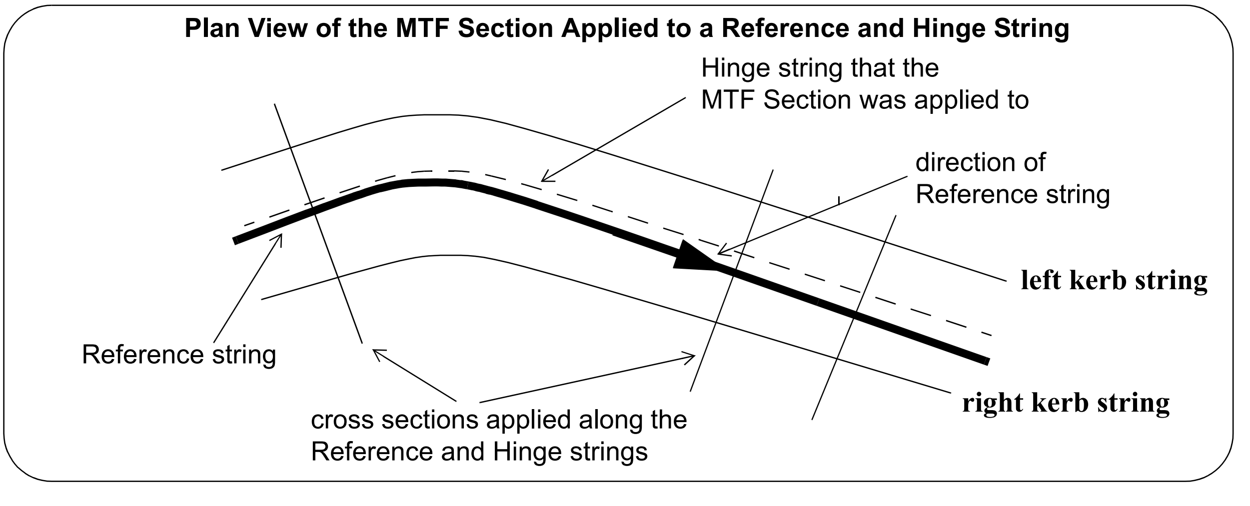
When the **MTF** is **Applied** to a **Reference** and a **Hinge String**, ***MTF links** and **points*** are created at each chainage, and it is also known, whether or not, the **same *MTF link* and point exists** at the **next chainage**.

So applying the **MTF** defines not only each *MTF link* and but also how each *link point* is connected **longitudinally** along the Reference and Hinge strings to form a string, **or more than one string** if there are some chainages where the *link point* does not exist.

The longitudinal strings (**MTF strings**) created by a *link point* are given the same name and colour as the *link point*.

As an example, the MTF with one link called *left kerb* and one link on the right called *right kerb* applied down the a Reference and Hinge String with produce two strings called *left kerb* and *right kerb:*





For brevity, when there is no confusion, the longitudinal MTF strings created by the MTF are often just called e **strings**.

Unlike the simpler *Apply Template*, the ***Apply MTF*** option does not automatically create strings for all the *MTF points* but allows the user to specify which strings are created. *See* 21.2.2.5.2 Create Strings.

Continue to the next section 21.1.1.2 MTF Links and Layers or return to 21.1.1 MTF Links, Points, Sections, Strings and Trimeshes

21.1.1.2 MTF Links and Layers

As well as having a name, each **MTF Link** also has a unique **Layer**. The default **Layer** is **Design**.

Layers are simply a way of grouping certain links together so that they can be referred to as one thing. For example, the default Layer **Design** is usually where all the links go that make up the Design.

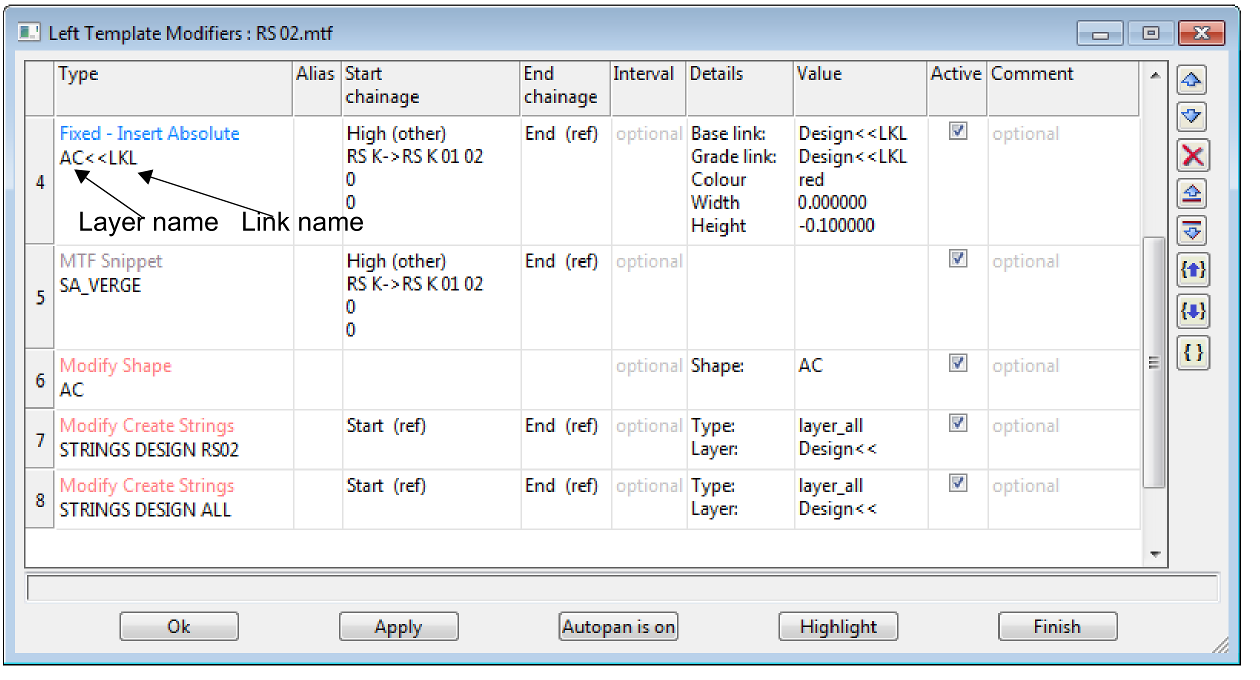
When a MTF Link is created, the Layer is given at the same time as the link name and colour and most MTF commands work for any Layer.

A **MTF Link** of name ***link_name*** in the layer ***layer_name*** is denoted by:

layer_name << link_name *when* ***link_name*** *is a left side link.*

layer_name >>link_name *when* ***link_name*** *is a a right side link.*

layer_name =>link_name *when the side for* ***link_name*** *has not yet been evaluated (this can happen in general Snippet code that can work when inserted in the Left or Right Modifiers panel.)*



In the **Apply MTF Function**, strings or sections are only automatically created for the layer **Design**.

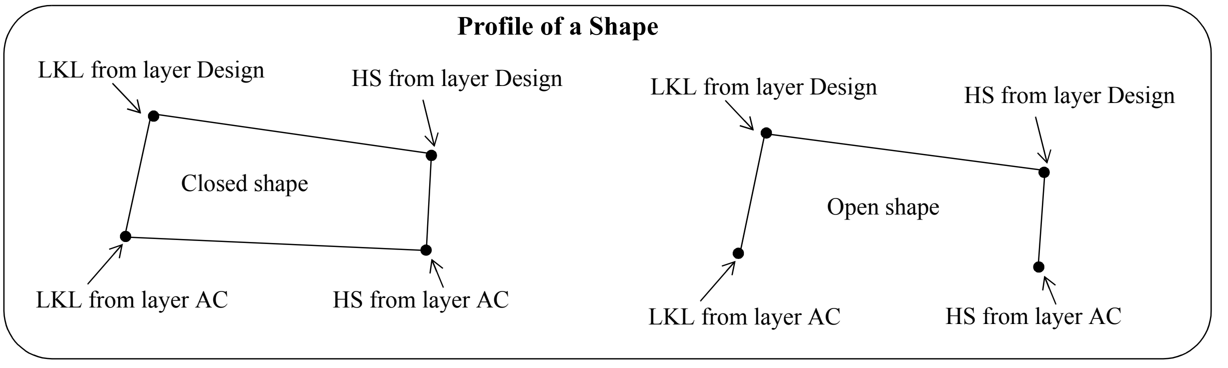
For other Layers, strings are created using the MTF command **Create Strings.** See 21.2.2.5.2 Create Strings.

Continue to the next section 21.1.1.3 MTF Shapes and Trimeshes or return to 21.1.1 MTF Links, Points, Sections, Strings and Trimeshes

21.1.1.3 MTF Shapes and Trimeshes

A **Shape** is made up of a group of **MTF link points** that in profile are joined together in a specific order. Each **MTF link point** can come from any **Layer**.

A Shape can be open or closed

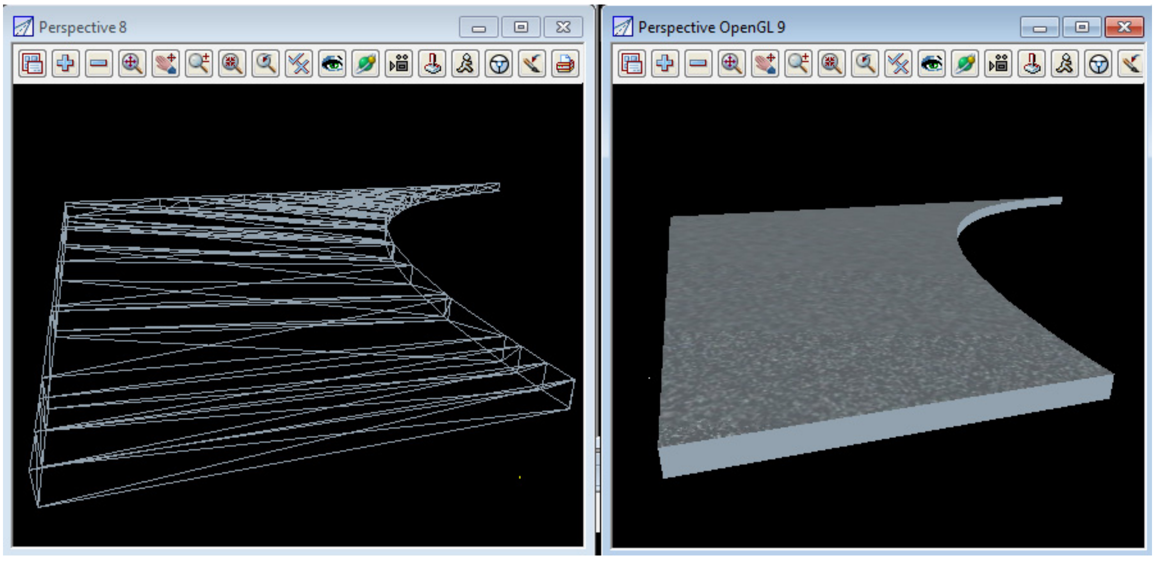


So shapes are similar to cross sections except a shape can go back under or over itself, and can be closed.

When and **MTF** is Applied to a Defence and Hinge string (**Apply MTF Function)**, the Shape will be defined at chainages along the Reference string and Hinge string, and the MTF Link Points of the Shape can generate MTF strings for the Shape.

A **Trimesh** can be constructed from the MTF sections and MTF strings generated by applying the Shape along the Reference and Hinge string. See 4.5.5 Trimeshes.

So the **Trimesh of a Shape** can be thought of as a profile swept (extruded) along the Reference and Hinge string. But remember that at any chainage, the **MTF** can change the positions of each MTF point so the profile of the Shape can change from chainage to chainage.



There is a **MTF** option to creates a Trimesh of a *Shape* and its *trimesh*. *See* 21.2.2.5.1 Create Shapes

Continue to the section 21.1.2 Calculating Link Values or return to 21.1.1 MTF Links, Points, Sections, Strings and Trimeshes.

21.1.2 Calculating Link Values

See

21.1.2.1 Calculating Width, Height, Xfall or Slope from Original Definitions

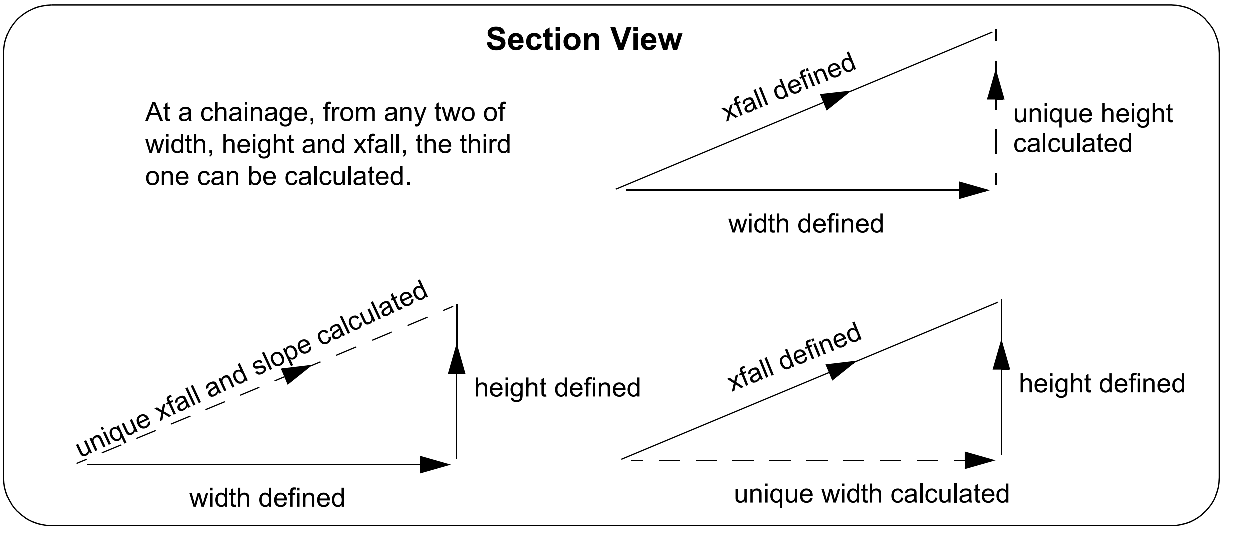
For most of the ***MTF Modifier*** commands and the ***MTF Absolute*** command, the command **modifies** the current value of the link of one for one of Width, Height, Xfall or Slope, and uses (**holds**) the current value of the link of another one of Width, Height, Xfall or Slope.

Together, the **modified** value and the **held** value defines the new values for the link.

It doesn't matter which two of width, height and xfall or slope were originally used to define the link, the option will internally convert the definition to what is required to be **modified** and **held**.

For example the link may be originally defined by **xfall** and **height** but the link **Modifier type** could be **Modify Width, Hold Xfall** or **Modify Xfall, Hold Width**.

Even though the existing link is not defined by width, for each required chainage, the xfall and height values of the existing link **uniquely define a width** at that chainage. And it is this calculated width that is then modified by **Modify Width, Hold Xfall** or used (held) by a **Modify Xfall, Hold Width.**



So no matter how the link is originally defined, any one of width, height or xfall/slope can be calculated and modified or held by the option.

Continue to the section 21.1.2.2 Modify X, Hold Y in MTF Modifiers or return to 21.1.2 Calculating Link Values.

21.1.2.2 Modify X, Hold Y in MTF Modifiers

A **link** in an **MTF** is initially defined by two out of three of width, height and xfall/slope because **only two** are needed to uniquely define the link. Any values not used in the link definition can be calculated from the defining values.so any of the values width, height, xfall and slope for the link are always known. See 21.1.2.1 Calculating Width, Height, Xfall or Slope from Original Definitions.

In an ***Apply MTF***, there are MTF Modifiers such as the **Modify link** command that act on the link and ***modifier*** one or some times two of the values defining the **link**.

One of the most common methods to modify the definition of a link is that **one** of the link values **is not modified** and keeps the same value (it is **held**) and **another value** is **modified** until a given condition is satisfied.

That is, for the link:

(a) **one** of Width, Height or Xfall **keeps its existing value** (i.e. **holds** its **existing value)**.

and

(b) **a different one** of Width, Height or Xfall **is modified** and takes the value needed to satisfy a given condition.

This is summarised as

Hold X, Modify Y

or **Modify Y, Hold X**

For example, for the **To String** modifier

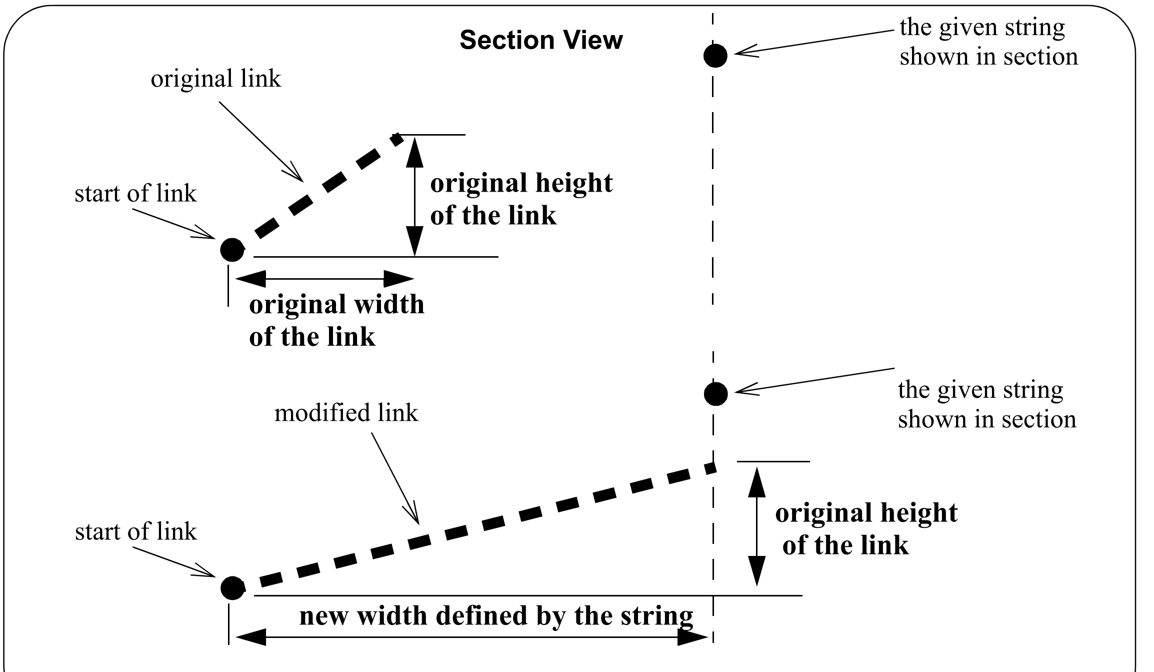
Modify Width, Hold Height.

(a) the **Height keeps its existing value** for the link (i.e. **holds** its existing value).

and

(b) the **Width is modified** to be the value that puts the end of the link directly under a given string.

(see 21.2.2.2.10 Fixed Link - To String)



Modify Width to a String and Hold Height

With the **To String** modifier, if the string is moved, the **Width** will also change to keep the end of the link under the string.

Continue to the section 21.1.2.3 New Value Usage or return to 21.1.2 Calculating Link Values.

21.1.2.3 New Value Usage

New value usage has the choices:

(a) *Add to the current value of the link*

If **Add to the current value of the link**, the value of the parameter width/height/xfall/slope at each chainage is **added to** the current value for the parameter. For compactness, in a grid, this will be displayed in an **Absolute** column with **Absolute not ticked** or **no.**

and

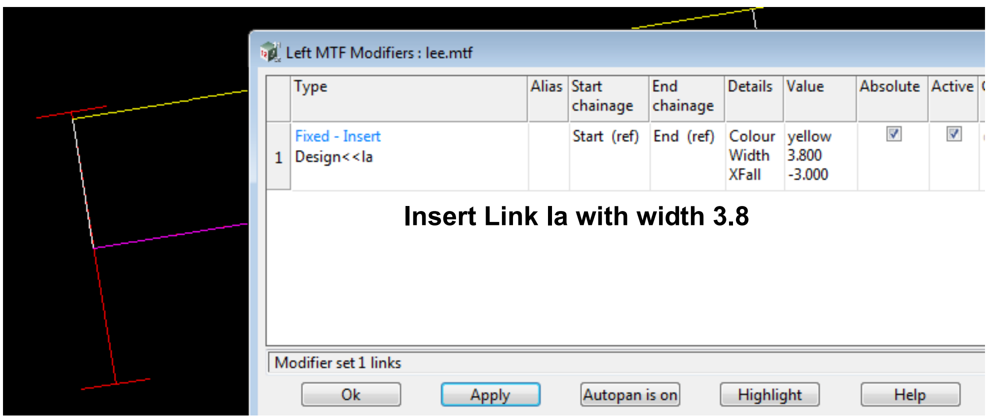
(b) *Replace the current value of the link*.

If **Replace the current value of the link**, the value of the parameter width/height/xfall/slope at each chainage **replaces** the current value for the parameter. For compactness, In a grid, this will be shown in an **Absolute** column with **Absolute ticked** or **yes**.

The different is best understood by running a Modifier command on a simple one Link job and then modifying the Width with the two values of **New value usage**.

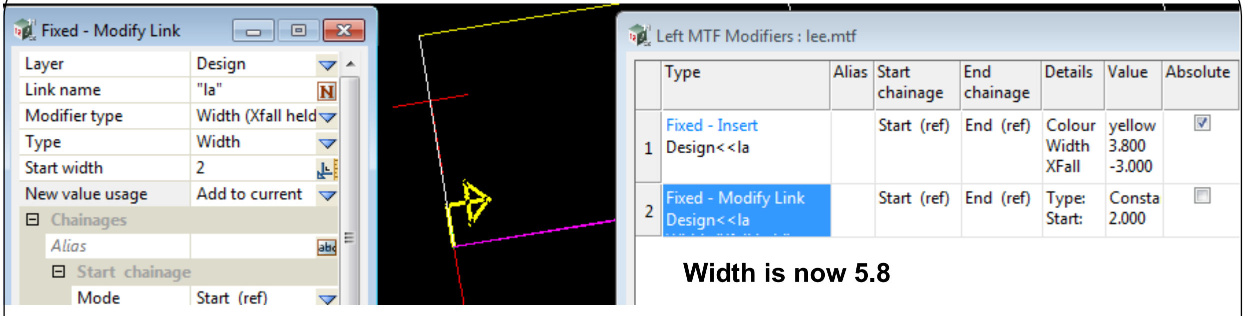
First insert a link called **la** with a **width** of **3.8.**

.



Then add a Modify a **Modify Link** command **to Modify the Width** with a **constant width of 2,** and **New value usage** with the value *Add to the current value of the link* (also denoted as **Absolute off)**.

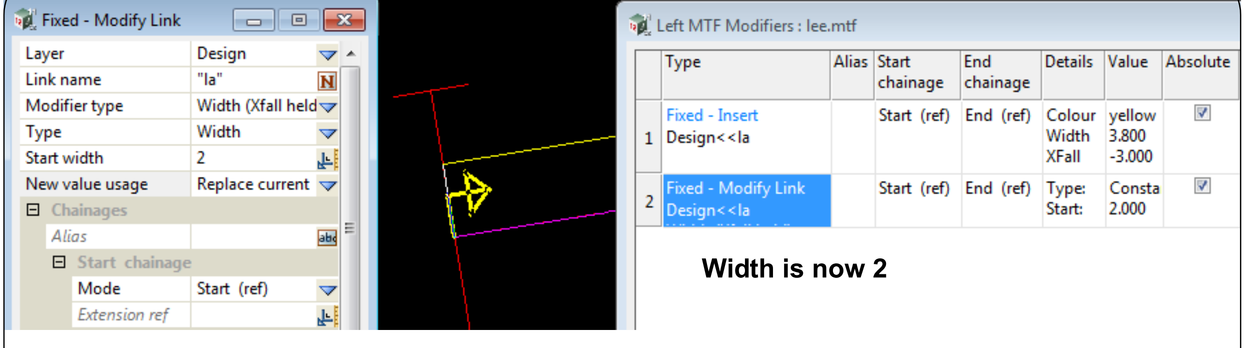
The **Width** is then **5.8.**



Modify Link command **to Modify the Width** with a **constant width of 2**
New value usage: Add to the current value of the link (Absolute **off**)

Now change the value of **New value usage** to *Replace the current value of the link* (i.e. **Absolute on).**

The **Width** is then **2.**



Modify Link command to Modify the Width with a **constant width of 2**
New value usage: Replace the current value of the link to current value (Absolute **on**)

So a **New value usage** of *Add to the current value of the link* (i.e. Absolute **off**) **adds** to the current value of the link and you may not even know what the final width of the link is until the MTF is run.

Whereas a **New value usage** of *Replace the current value of the link* (i.e Absolute **on**) **resets** the value to a given value. So at that point in the MTF, you know exactly what the value of Width is.

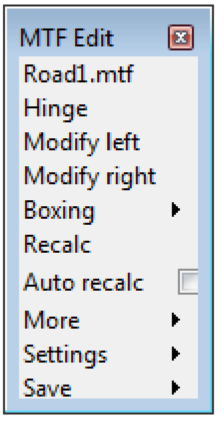
Note that there may be more *Width* commands further down in the MTF command list that may change the value of Width again so the final Width will only be definitely be known when the entire MTF is processed.

Continue to the section 21.2 MTF Edit or return to 21.1.2 Calculating Link Values.

21.2 MTF Edit

For general information about the MTF (Modifiers and Template File) see 21.1 The Modifiers and Templates File - MTF.

The **MTF Edit** menu is



For documentation on the options see:

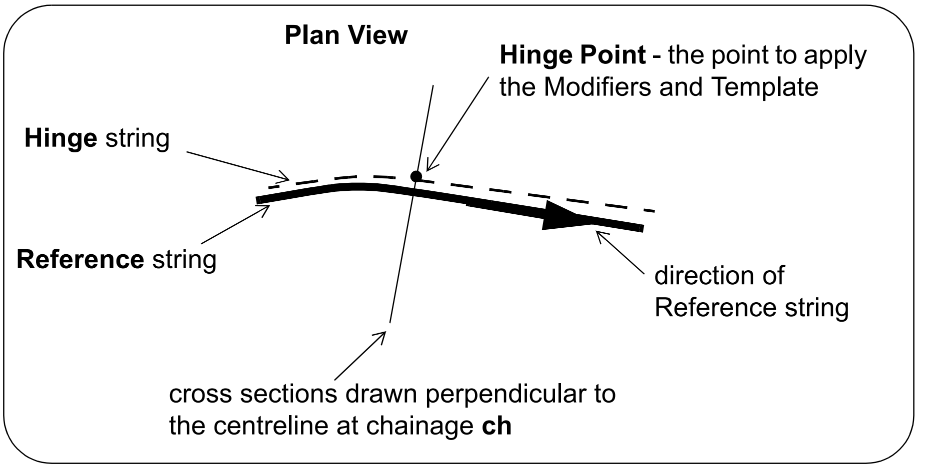
Hinge	21.2.1 MTF Hinge Modifiers
Modify left	21.2.2 Left and Right MTF Modifiers
Modify right	21.2.2 Left and Right MTF Modifiers
Boxing	21.2.3 MTF Boxing
Recalc	21.2.4 MTF Recalc
Auto recalc	21.2.5 MTF Auto Recalc
More	21.2.6 MTF More
Settings	21.2.7 MTF Settings
Save	21.2.8 MTF Save

Please continue to the next section 21.2.1 MTF Hinge Modifiers or return to 21.2 MTF Edit.

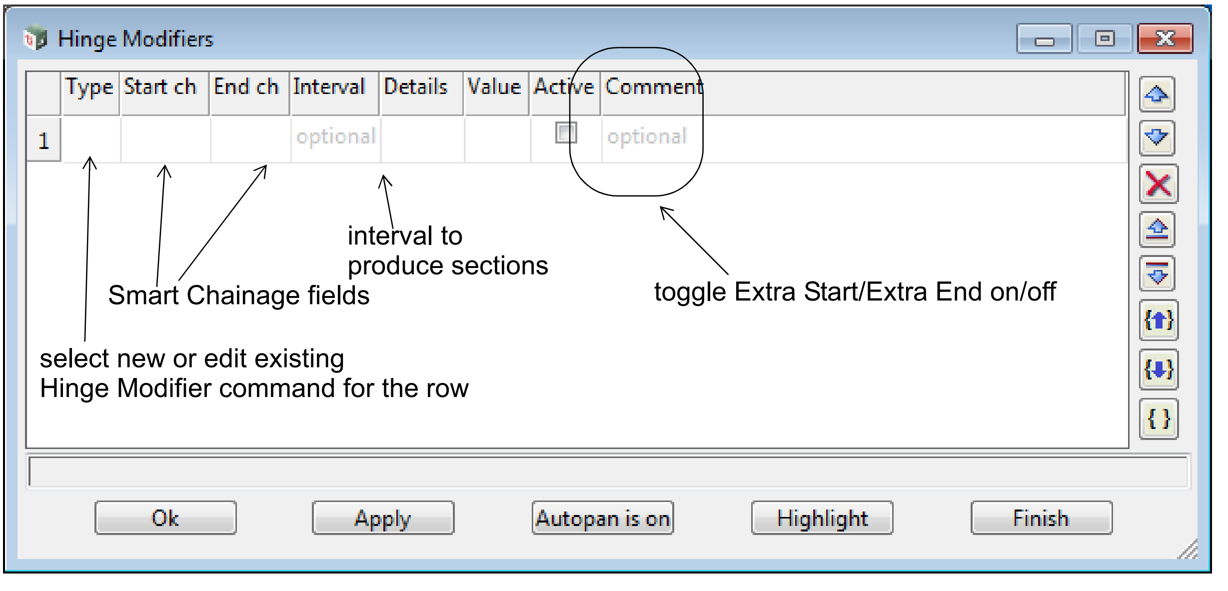
21.2.1 MTF Hinge Modifiers

The **hinge string** is selected using the **Hinge** panel field in the **Apply MTF Function** panel.

Hinge modifiers are used to **modify** the position of the point on the hinge string (the **Hinge point**) that is considered the zero offset and Height point when applying the Templates and Modifier commands. See 20.1 Templates in 12d Model.



Selecting **Hinge** from the **MTF Edit** menu brings up the **Hinge Modifiers** panel.



The **Hinge Modifiers** panel consists of a grid with of rows (or lines) with **commands** in them, and an **OK** or **Apply** button to record the results.

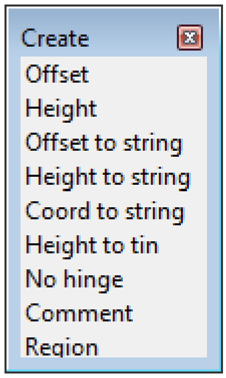
Hinge Modifier Grid

For information on the general operation of a grid including the icons on the right hand side, see 4.19.6 Grids in Panels.

*The **commands** in the* **Hinge Modifier** *grid are processed sequentially from the top to the bottom of the grid.*

When the Grid Row is Empty

*If the row of the grid is **empty**, clicking LB in the empty row will bring up the* **Create** *menu which contains all the available hinge commands.Note this may involve two clicks - one to highlight a column in a row and the second click to bring up the* **Create** *menu.*



Selecting a menu item will bring up an associated panel which displays the information required for the hinge command. When the panel is filled in and **OK** or **Apply** *selected, the panel information is written out to the row of the grid and is known as a* **hinge command**. *For information on each of the Hinge Commands, see* 21.2.1.1 Create Hinge Commands.

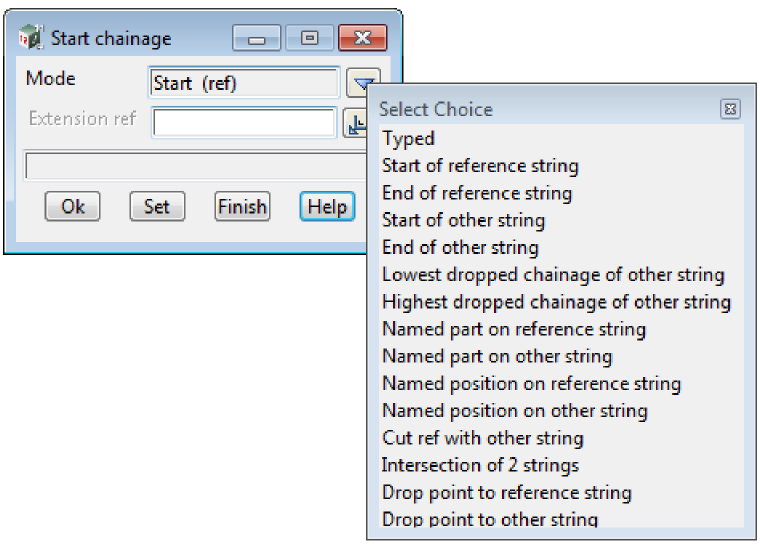
When the Grid Row is Not Empty

If the row of the grid is ***not empty*** *(and hence filled with a* **hinge command***) then clicking LB in the cell:*

(a) ***Type*** *will bring up the associated panel for the hinge command.*

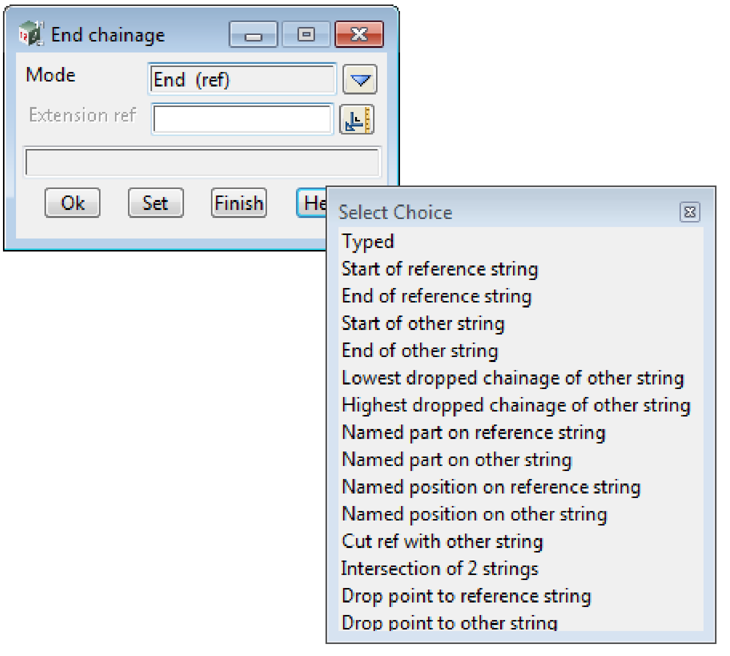
For information on each of the Hinge Commands, see 21.2.1.1 Create Hinge Commands.

(b) ***Start chainage*** *cell brings up the Smart* ***Start Chainage*** *panel*



The **Start Chainage Mode** *will determine what other fields are also on the panel (for example* **Extension ref** *for* **Mode** ***Start of reference string***. *For more information on Chainage Modes, see* 21.2.1 MTF Hinge Modifiers

(c) ***End Chainge*** *cell brings up the Smart* ***End Chainage*** *panel*



The **End Chainage Mode** *will determine what other fields are also on the panel (for example* **Extension ref** *for* **Mode** ***End of reference string***. *For more information on Chainage Modes, see* 21.2.1 MTF Hinge Modifiers

(d) ***Interval*** *will allow a real value to be typed into the cell*

Note that the ***Interval*** *column will not appear in the grid if* ***Show interval column?*** *is* ***not*** *ticked in the MTF Settings. See* 21.2.7.8 MTF Setting Show Interval Column

(e) ***Extra Start***, ***Extra End*** *or* ***Active*** *will toggle the* ***tick/not ticked***

Note that the ***Extra start*** *and* ***Extra end*** *columns will not appear in the grid if* ***Show extra start/ end column?*** *is* ***not*** *ticked in the MTF Settings. See* 21.2.7.6 MTF Setting Show Extra Start/ End column.

(f) ***Comment*** *will allow text to be typed into the cell*

Note *that clicking in the* **Type**, **Start chainage**, **End chainage**, **Interval**, **Extra Start**, **Extra End**, **Active** *and* **Comment** *cells may involve two clicks - one to highlight the cell in a row and the second click to edit or bring up the panel for the cell.*

Buttons at Bottom

OK button

OK *stores the values in the fields and removes the panel BUT no* ***recalc*** *is done.*

Apply button

Apply *stores the values in the grid and leaves the panel on the screen.*

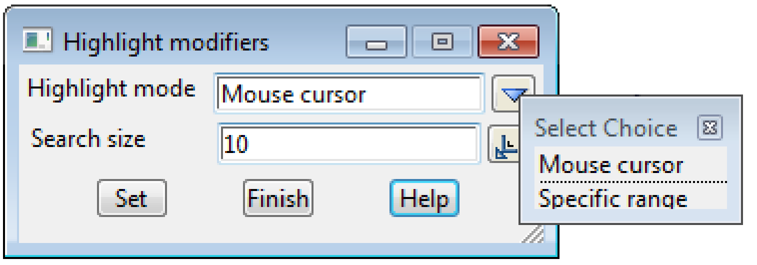
If ***Auto recalc*** *is* ***ticked*** *in the MTF, then whenever the* ***Apply*** *button is clicked, a* ***recalc*** *of the associated* ***Apply Many*** *for the MTF is done.*

Autopan on/off button

when clicked to say **Autopan is on** *then if the chainage range is not in the view that the Reference string is on then the view will be panned so that it is on the view.*

Highlight button

clicking the ***Highlight*** *button brings up the* **Highlight Modifiers** *panel.*

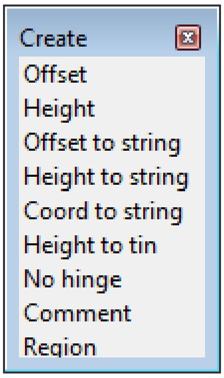


Continue to the next section 21.2.1.1 Create Hinge Commands.

21.2.1.1 Create Hinge Commands

The **commands** in the **Hinge Modifiers** grid are processed sequentially from top to bottom.

The **Hinge Modifier commands** are selected from the **Create** menu that is displayed when clicking in a cell in the **Type** column of the **Hinge Modifiers** panel.



adjust offset of hinge string
adjust offset of hinge to a height
use offset to a string
use height from a string
take coords from a string
use height from a string
stop the apply
comment
define a region

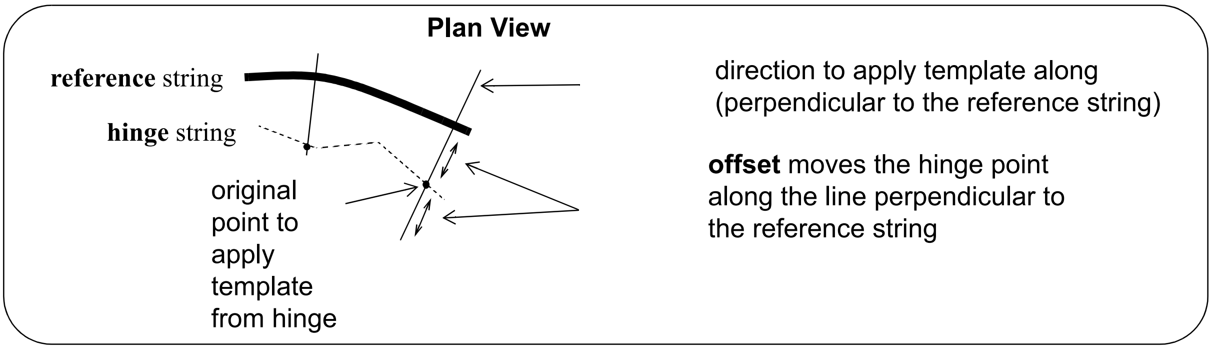
For a description of the options, see

Offset	21.2.1.2 Offset
Height	21.2.1.3 Height
Offset to string	21.2.1.4 Offset to String
Height to string	21.2.1.5 Height to String
Coord to sting	21.2.1.6 Coord to String
Height to tin	21.2.1.7 Height to Tin
No hinge	21.2.1.8 No Hinge
Comment	21.2.1.9 Comment
Region	21.2.1.10 Region

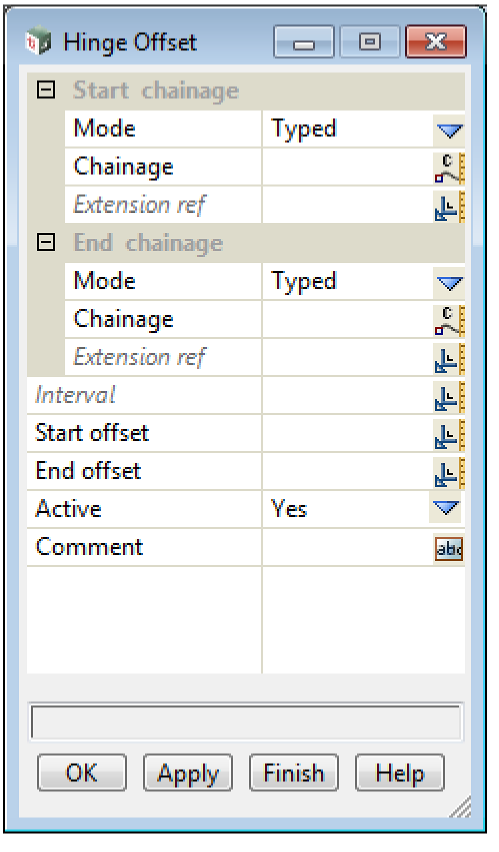
21.2.1.2 Offset

The **offset** hinge modifier will move the hinge point a given offset distance (perpendicular to the reference string) from its current plan position. A positive offset is to the right of the hinge string and a negative offset to the left.

.



Selecting **Offset** brings up the **Hinge Offset** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Start chainage, End chainage, Interval			

defines the start/end chainages for modifying the offset of the hinge string.

for information on these panel fields, see 21.2.1.11 Common Fields and Buttons on Hinge Modifier Panels.

Start/End offset	input		measures menu

start/end offset.

Absolute tick box tick

*if **ticked**, the offset is set to the values given in the* **start** *and* **end value** *fields.*
*if **not ticked**, the values given in the* **start** *and* **end value** *fields are added to the existing offsets.*

Cubic tick box

*if **ticked**, the offset is varied as a reverse cubic between the start and end chainages.*
*if **not ticked**, the offset is varied linearly between the start and end chainages.*

Active, Comment, Extra start, Extra end, OK, Apply

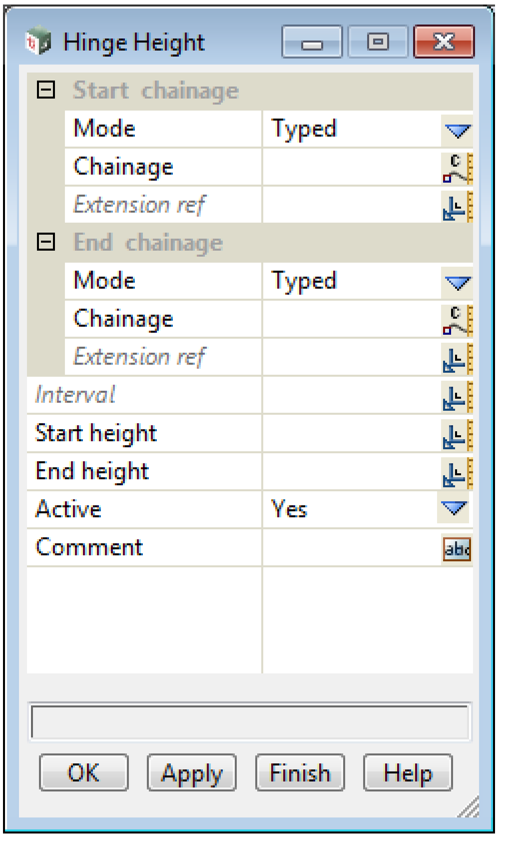
for information on these panel fields, see 21.2.1.11 Common Fields and Buttons on Hinge Modifier Panels.

Go to the next section 21.2.1.3 Height or return to 21.2.1 MTF Hinge Modifiers

21.2.1.3 Height

The **Height** modifier linearly varies the height of the hinge point between the given chainages.

Selecting **Height** brings up the **Hinge Height** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Start chainage, End chainage, Interval			

defines the start/end chainages for modifying the height of the hinge string.

for information on these panel fields, see 21.2.1.11 Common Fields and Buttons on Hinge Modifier Panels.

Start/End height	real box		measures menu

start/end height.

Extra start/end	tick box	tick	

*if **ticked**, add an extra x-section 0.1 mm **before** the start/end chainage.*

Absolute	tick box	tick	

*if **ticked**, the height is set to the values given in the* **start** *and* **end value** *fields.*
*if **not ticked**, the values given in the* **start** *and* **end value** *fields are added to the existing heights.*

Cubic	tick box		

*if **ticked**, the height is varied as a reverse cubic between the start and end chainages.*
*if **not ticked**, the height is varied linearly between the start and end chainages.*

Active, Comment, Extra start, Extra end, OK, Apply

for information on these panel fields, see 21.2.1.11 Common Fields and Buttons on Hinge

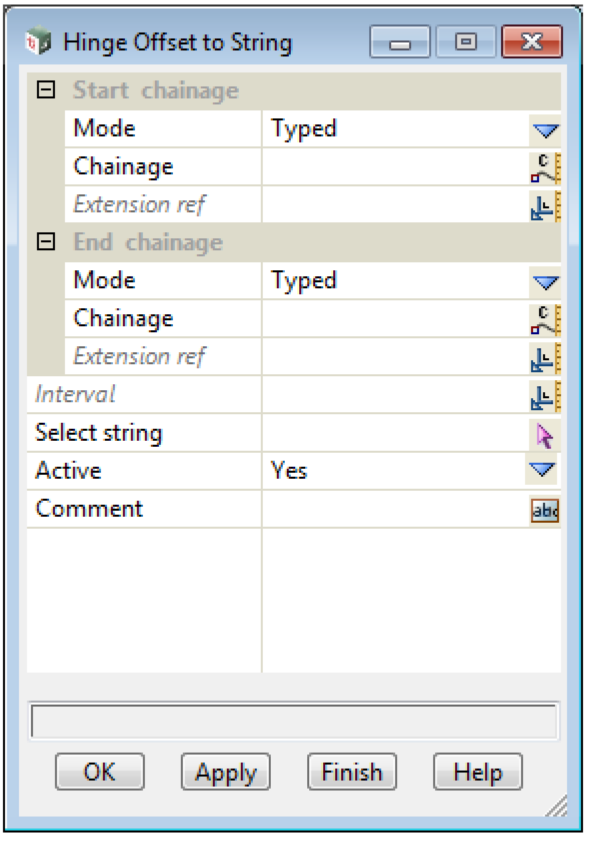
Modifier Panels.

Go to the next section 21.2.1.4 Offset to String or return to 21.2.1 MTF Hinge Modifiers

21.2.1.4 Offset to String

The **Offset to string** hinge modifier makes the offset of the hinge point the same as the offset of a selected string

Selecting **Offset to string** brings up the **Hinge Offset to String** panel



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Start chainage, End chainage, Interval

defines the start/end chainages for modifying the offset of the hinge string.

For information on these panel fields, see 21.2.1.11 Common Fields and Buttons on Hinge Modifier Panels.

*When the **Start mode** is **Start (ref),** or **Typed** and the chainage is **blank**, the modification begins at the low dropped chainage of the selected string.*

*When the **End mode** is **End (ref),** or **Typed** and the chainage is **blank**, the modification ends at the high dropped chainage of the selected string.*

String string-select

select string to use for defining offset from hinge.

Active, Comment, Extra start, Extra end, OK, Apply

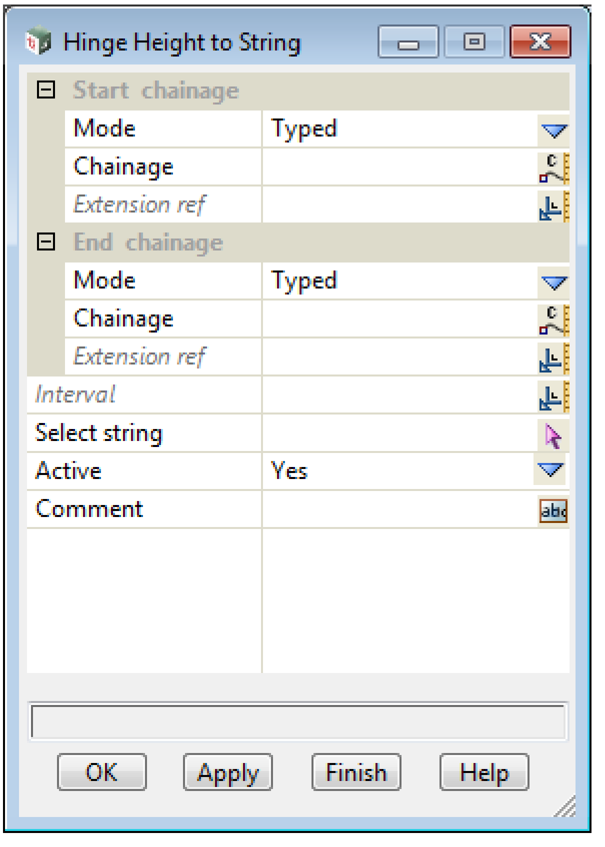
for information on these panel fields, see 21.2.1.11 Common Fields and Buttons on Hinge Modifier Panels.

Go to the next section 21.2.1.5 Height to String or return to 21.2.1 MTF Hinge Modifiers

21.2.1.5 Height to String

The **Height to string** hinge modifier makes the height of the hinge point the same as the height of a selected string

Selecting **Height to string** brings up the **Hinge Height to String** panel



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Start chainage, End chainage, Interval

defines the start/end chainages for modifying the height of the hinge string

For information on these panel fields, see 21.2.1.11 Common Fields and Buttons on Hinge Modifier Panels.

When the ***Start mode*** *is* ***Start (ref),*** *or* ***Typed*** *and the chainage is* ***blank****, the modification begins at the low dropped chainage of the selected string.*

When the ***End mode*** *is* ***End (ref),*** *or* ***Typed*** *and the chainage is* ***blank****, the modification ends at the high dropped chainage of the selected string.*

String string-select

select string to use for defining height of hinge.

Active, Comment, Extra start, Extra end, OK, Apply

for information on these panel fields, see 21.2.1.11 Common Fields and Buttons on Hinge Modifier Panels.

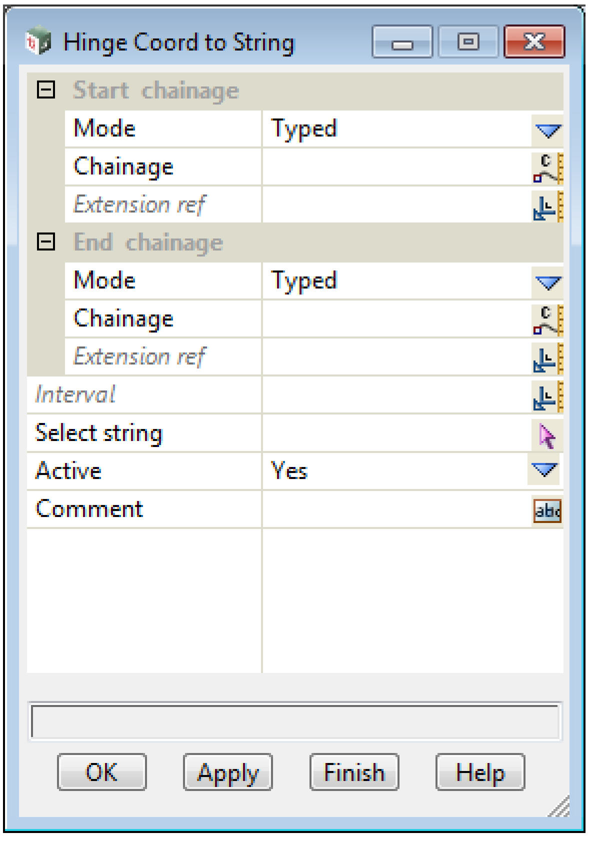
Go to the next section 21.2.1.6 Coord to String or return to 21.2.1 MTF Hinge Modifiers

21.2.1.6 Coord to String

The **Coord** hinge modifier is used to replace the x, y and z position of the hinge point by the x, y and z position of **another *12d Model* string**.

Hence **Coord** replaces the hinge string by another string between the given chainages.

Selecting **Coord to string** brings up the **Hinge Coord to String** panel



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Start chainage, End chainage, Interval

defines the start/end chainages for modifying the coordinates of the hinge string

For information on these panel fields, see 21.2.1.11 Common Fields and Buttons on Hinge Modifier Panels.

*When the **Start mode** is **Start (ref),** or **Typed** and the chainage is **blank**, the modification begins at the low dropped chainage of the selected string.*

*When the **End mode** is **End (ref),** or **Typed** and the chainage is **blank**, the modification ends at the high dropped chainage of the selected string.*

String string-select

select string to use for defining the coordinates of the hinge string.

Active, Comment, Extra start, Extra end, OK, Apply

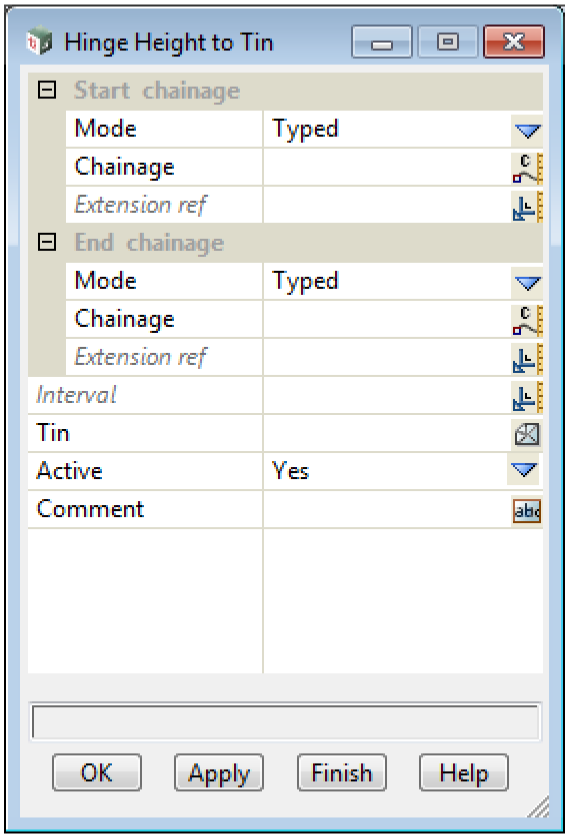
for information on these panel fields, see 21.2.1.11 Common Fields and Buttons on Hinge Modifier Panels.

Go to the next section 21.2.1.7 Height to Tin or return to 21.2.1 MTF Hinge Modifiers.

21.2.1.7 Height to Tin

The **Height to tin** hinge modifier makes the height of the hinge point the same as the height of a tin at the same (x,y) location as the hinge point.

Selecting **Height to tin** brings up the **Hinge Height to Tin** panel



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Start chainage, End chainage, Interval

defines the start/end chainages for modifying the height of the hinge string

for information on these panel fields, see 21.2.1.11 Common Fields and Buttons on Hinge Modifier Panels.

Tin tin box

tin to use for z-values. At each vertex of the hinge string in he chainage range, set the z-value to be the tin z-value at the same (x,y) position as the vertex.

Active, Comment, Extra start, Extra end, OK, Apply

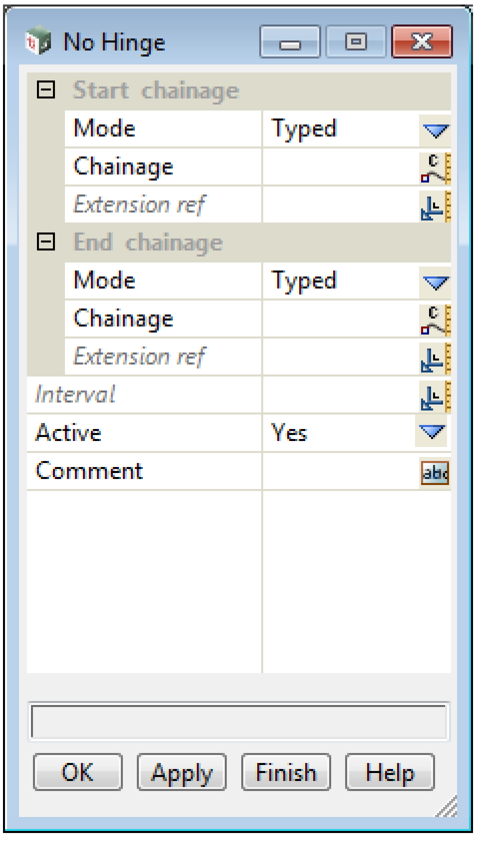
for information on these panel fields, see 21.2.1.11 Common Fields and Buttons on Hinge Modifier Panels.

Go to the next section 21.2.1.8 No Hinge or return to 21.2.1 MTF Hinge Modifiers.

21.2.1.8 No Hinge

The **Nohinge** hinge modifier is used to **stop** the hinge string (and hence the apply) between given chainages. This will leave a gap in the strings created by the apply between the given chainages.

Selecting **No hinge** brings up the **No Hinge** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Start chainage, End chainage, Interval

defines the start/end chainages for stopping the hinge string

for information on these panel fields, see 21.2.1.11 Common Fields and Buttons on Hinge Modifier Panels.

Active, Comment, Extra start, Extra end, OK, Apply

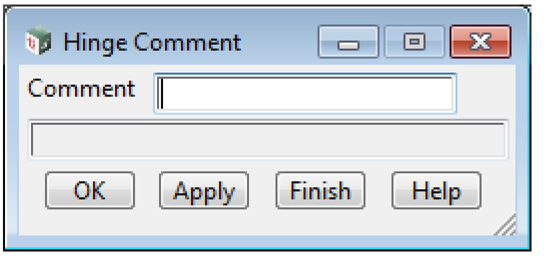
for information on these panel fields, see 21.2.1.11 Common Fields and Buttons on Hinge Modifier Panels.

Go to the next section 21.2.1.9 Comment or return to 21.2.1 MTF Hinge Modifiers

21.2.1.9 Comment

The **Comment** option inserts a comment into the row.

Selecting **Comment** brings up the **Hinge Comment** panel



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Comment, OK, Apply

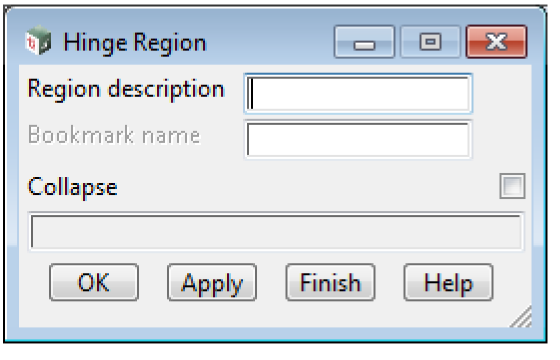
for information on these panel fields, see 21.2.1.11 Common Fields and Buttons on Hinge Modifier Panels.

Go to the next section 21.2.1.10 Region or return to 21.2.1 MTF Hinge Modifiers.

21.2.1.10 Region

The **Region** option inserts a **Region command** into the grid.

Selecting **Region** brings up the **Hinge Region** panel



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Region description text box

name for the Region.

Bookmark name text box

*if **not blank**, this is a secondary name, often a shorter name, that is used in the **Region** pop up rather than using the **Region description.***

*If **blank**, then when the grid is saved, the* **Bookmark name** *is set to be the same as the* **Region description**.

Collapse tick box

*if **ticked**, all the commands in the grid until the next **Region** command are collapsed into this Region command.*
*If **not ticked**, the commands in the grid until the next **Region** command are not collapsed.*

OK, Apply

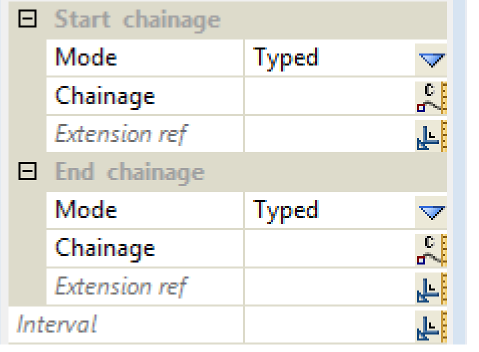
for information on these panel fields, see 21.2.1.11 Common Fields and Buttons on Hinge Modifier Panels.

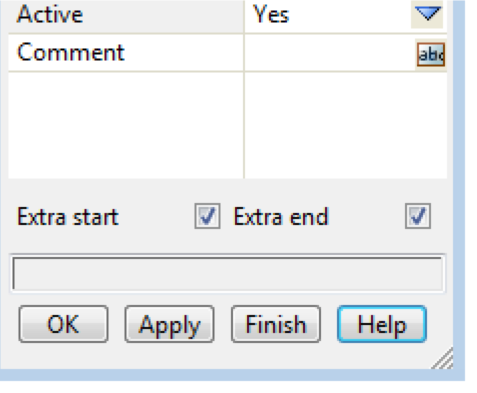
There are **Regions**, **Previous Region** and **Next Region** icons at the right hand side of the **Hinge Modifiers** panel that use this created *Region*. For information on using these icons and information on **Regions** in a grid, see 4.19.6 Grids in Panels.

Go to the next section 21.2.1.11 Common Fields and Buttons on Hinge Modifier Panels or return to 21.2.1 MTF Hinge Modifiers.

21.2.1.11 Common Fields and Buttons on Hinge Modifier Panels

Most of the ***Hinge Modifier*** panels have the following common panels fields and buttons.





Start/End chainage smart chainage box Start (ref)/End (ref)

defines the start/end chainages for use in the Modifier.
For more information on Start/End Chainages and Modes, see 21.4 Smart Chainages

Interval real box

*if **not blank**, the interval to use to create cross sections and strings over the given chainage range.*
*If **blank**, the interval given by any **MTF Edit => Interval commands**, and/or the* **Section separation** *value from the* **Apply MTF** *panel, is used.*

Extra start tick box tick

*if **ticked**, add an extra x-section a distance* **Extra start/value** *millimetres **before** the start chainage. See* 21.2.7.4 MTF Setting Show Extra Start/End *and* 21.2.7.5 MTF Setting Extra Start/End value.

Extra end tick box tick

*if **ticked**, add an extra x-section a distance* **Extra start/value** *millimetres **before** the end chainage. See* 21.2.7.4 MTF Setting Show Extra Start/End *and* 21.2.7.5 MTF Setting Extra Start/End value.

Comment text box

comment to add to the Comment column for the row of the grid.

Active yes/no choice box

*if **yes**, use this modifier.*
*If **no**, don't use this modifier.*

OK button

OK *stores the values in the fields and removes the panel BUT no **recalc** is done.*

Apply button

Apply *stores the values and leaves the panel on the screen.*

Go to the next section 21.2.2 Left and Right MTF Modifiers or return to 21.2.1 MTF Hinge Modifiers

21.2.2 Left and Right MTF Modifiers

In its **simplest** form, the **MTF** (**Modifiers and Templates File** defines what templates are used on either side of the hinge string, where the templates stop and start with linear interpolation between two templates over a chainage range.

For more complex work, **MTF Modifiers** are used to **modify** the strings defined by the templates, and even totally replace the use of templates and insert links and modify the inserted links and generate sections, strings and trimeshes.

Just as templates are defined on the left and the right side, there are ***left*** and ***right MTF Modifiers*** for working on the links on the left or right side of the Hinge string.

Most of the ***MTF Modifiers*** have a common methodology:

(a) an existing link from a template (referenced by its name from the template definition), or a new link that is created by an MTF Insert, can be removed or modified between given start and end chainages on the reference string.

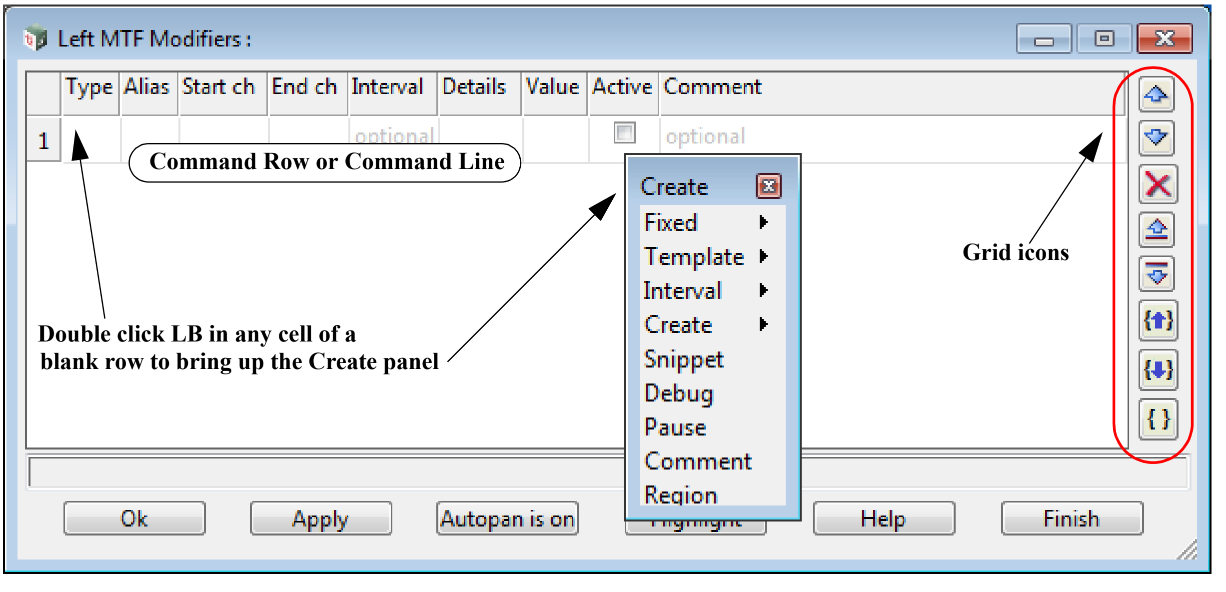
(b) values defining a link are modified over a given start and end chainage. How the values are modified depends on the particular modifier command.

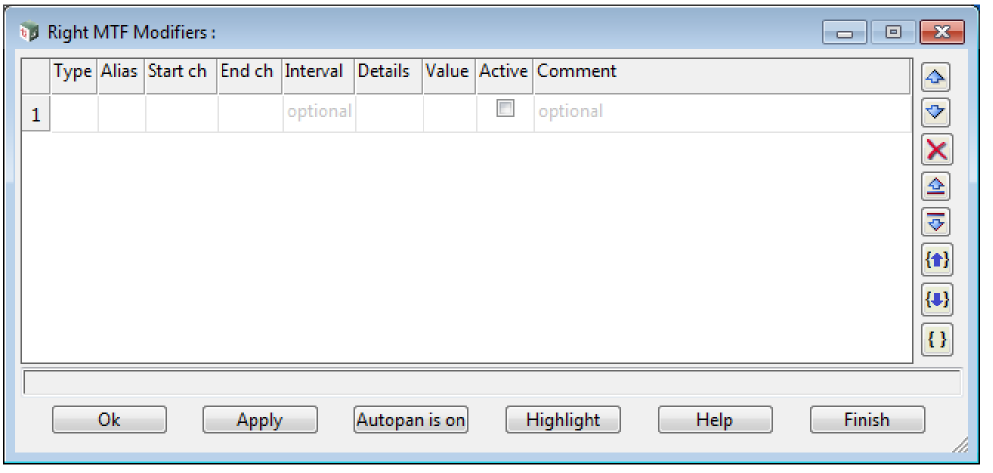
Note: regardless of how the link is originally defined (for example by width and xfall), any of the values width, height, xfall and slope then exist and can be modified. See 21.1.2.1 Calculating Width, Height, Xfall or Slope from Original Definitions.

(c) more than one ***MTF modifier*** may exist for a link at a given chainage hence the order of the MTF modifiers is important.

Modify left option brings up the **Left MTF Modifiers** panel which is used to create and edit modifiers to use on the left hand side of the Hinge string.

Similarly the **Modify right** option brings up the **Right MTF Modifiers** panel which is used to create and edit modifiers on the right hand side of the Hinge string.





The **Left/Right MTF Modifiers** panel consists of a grid with of rows (or lines) with **Commands** in them, and an **OK** or **Apply** button to record the results.

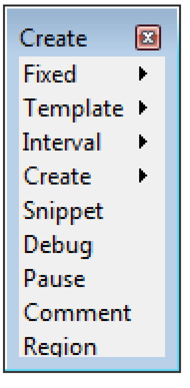
Left/Right MTF Modifier Grid

For information on the general operation of a grid including the icons on the right hand side, see 4.19.6 Grids in Panels.

The ***commands*** *in the* **Hinge Modifier** *grid are processed sequentially from the top to the bottom of the grid.*

When the Grid Row is Empty

If the row of the grid is ***empty****, clicking LB in the empty row will bring up the* **Create** *menu which contains all the available MTF Modifier commands.Note this may involve two clicks - one to highlight a column in a row and the second click to bring up the* **Create** *menu.*



Selecting a menu item will bring up an associated panel which displays the information required for the MTF Modifier command. When the panel is filled in and **OK** or **Apply** *selected, the panel information is written out to the row of the grid and is known as a* **MTF modifier command** *or* **modifier command** *for short. For information on each of the Hinge Commands, see* 21.2.2.1 Create MTF Commands.

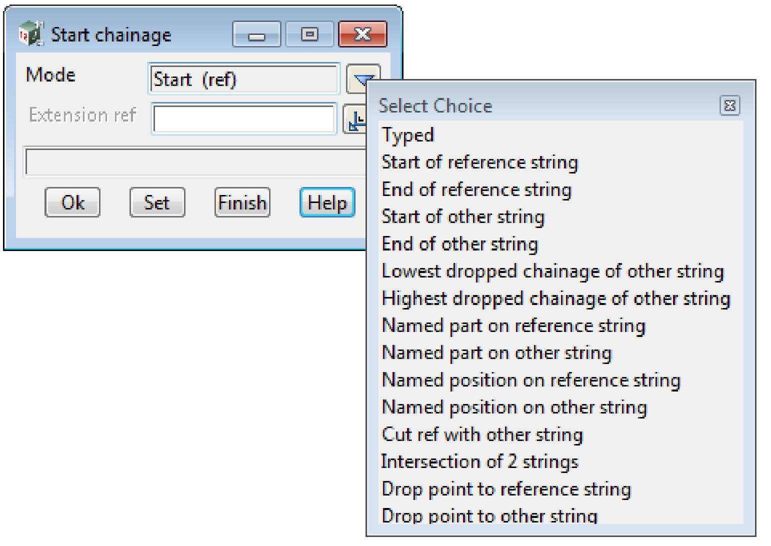
When the Grid Row is Not Empty

If the row of the grid is ***not empty*** *(and hence filled with a* **modifier command***) then clicking LB in the cell:*

(a) ***Type*** *will bring up the associated panel for the modifier command.*

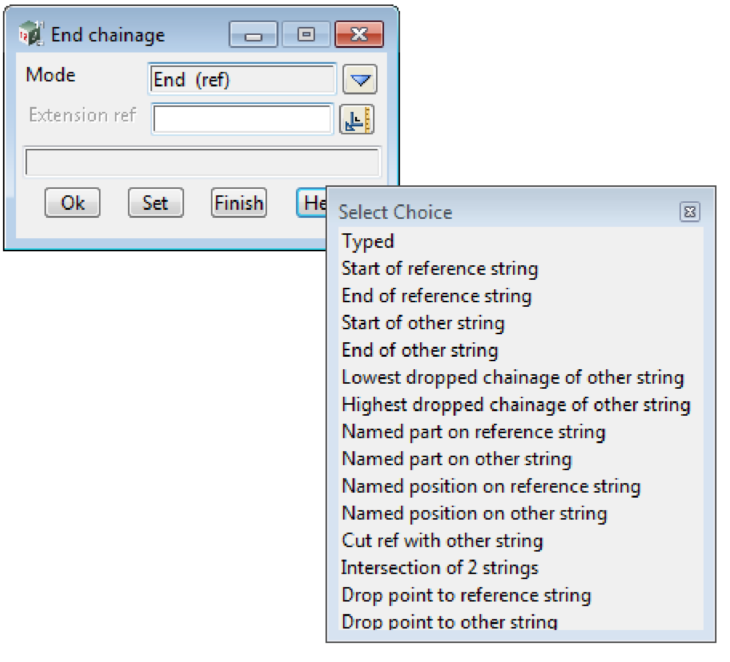
For information on each of the Hinge Commands, see 21.2.2.1 Create MTF Commands.

(b) ***Start chainage*** *cell brings up the Smart* ***Start Chainage*** *panel*



The **Start Chainage Mode** *will determine what other fields are also on the panel (for example* **Extension ref** *for* **Mode** ***Start of reference string****. For more information on Chainage Modes, see* 21.2.1 MTF Hinge Modifiers

(c) ***End Chainge*** *cell brings up the Smart* ***End Chainage*** *panel*



The **End Chainage Mode** *will determine what other fields are also on the panel (for example* **Extension ref** *for* **Mode** ***End of reference string****. For more information on Chainage Modes, see* 21.2.1 MTF Hinge Modifiers

(d) ***Interval*** *will allow a real value to be typed into the cell*

Note that the ***Interval*** *column will not appear in the grid if* ***Show interval column?*** *is* ***not*** *ticked in the MTF Settings. See* 21.2.7.8 MTF Setting Show Interval Column

(e) ***Extra Start****,* ***Extra End*** *or* ***Active*** *will toggle the* ***tick/not ticked***

Note that the ***Extra start*** *and* ***Extra end*** *columns will not appear in the grid if* ***Show extra start/ end column?*** *is* ***not*** *ticked in the MTF Settings. See* 21.2.7.6 MTF Setting Show Extra Start/ End column.

(f) ***Comment*** *will allow text to be typed into the cell*

Note that clicking in the **Type**, **Start chainage**, **End chainage**, **Interval**, **Extra Start**, **Extra End**, **Active** *and* **Comment** *cells may involve two clicks - one to highlight the cell in a row and the second click to edit or bring up the panel for the cell.*

Buttons at Bottom

OK button

OK *stores the values in the fields and removes the panel BUT no* ***recalc*** *is done.*

Apply button

Apply *stores the values in the grid and leaves the panel on the screen.*

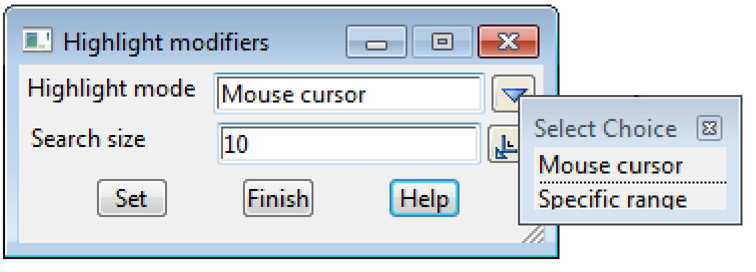
If ***Auto recalc*** *is* ***ticked*** *in the MTF, then whenever the* ***Apply*** *button is clicked, a* ***recalc*** *of the associated* ***Apply Many*** *for the MTF is done.*

Autopan on/off button

when clicked to say **Autopan is on** *then if the chainage range is not in the view that the Reference string is on then the view will be panned so that it is on the view. See*21.2.2.2 Fixed Link Modifiers.

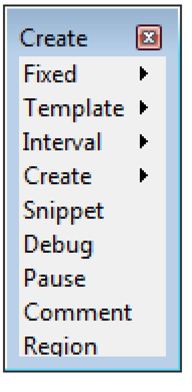
Highlight button

clicking the ***Highlight*** *button brings up the* **Highlight Modifiers** *panel.*



*See*21.2.2.1.6 Highlight Button

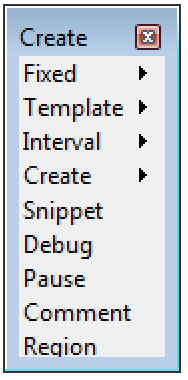
Each of the **Create** commands will be descried in the following sections.



The **Create** commands for the l***eft side modifiers*** and the ***right side modifiers*** work in the same way so the modifiers will only be described the once.

Continue to the next section 21.2.2.1 Create MTF Commands.

21.2.2.1 Create MTF Commands



modify fixed section
modify the cut, fill, final cut/fill of a template
define interval values, aliases etc
create shapes, strings, tins and trimeshes
insert snippets
turn on debugging
pause the MTF
add a comment row
define a region in the grid

When brought up from **Modifiers left**, the **Create** options work on the **links** are on the **left hand side** of the **Hinge string**. And on the left, the links are placed from right to left so the **start of a left side link** is at the **right hand end of the link**. See 21.1.1.1 MTF Links, Points and Strings.

When brought up from **Modifiers right**, the **Create** options work on the **links** are on the **right hand side** of the **Hinge string**. And on the right, the links are placed from left to right so the **start of a right side link** is at the **left hand end of the link**. See 21.1.1.1 MTF Links, Points and Strings.

For the options see

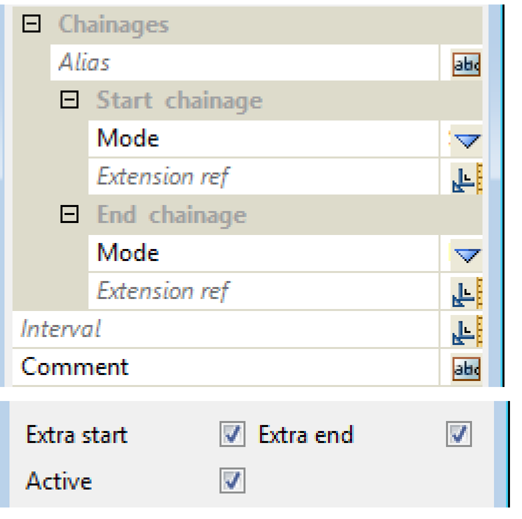
Fixed	21.2.2.2 Fixed Link Modifiers
Template	21.2.2.3 Template Modifiers
Interval	21.2.2.4 Interval
Create	21.2.2.5 MTF Create
Snippet	21.2.2.6 Snippet
Debug	21.2.2.7 Debug
Pause	21.2.2.8 Pause
Comment	21.2.2.9 Comment
Region	21.2.2.10 Region

For general information about the MTF Commands, see

21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels
21.2.2.1.2 Link or Link(s)
21.2.2.1.3 Width, Height and Xfall Fields on MTF Modifier Panels
21.2.2.1.4 Alias for MTF Modifiers
21.2.2.1.5 Autopan Button
21.2.2.1.6 Highlight Button

21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels

Most of the ***Modifier*** panels have the following common panels fields.



The fields and buttons have the following functions.

Field Description Type Defaults Pop-Up

Alias text box

*if **not blank**, a text name that can be used to refer to for this row in the grid.*

*The **Alias** is a reference to this command in the **Left/Right MTF Modifiers** grid that can be referred to by Smart Chainages used in the **Left/Right MTF Modifiers** commands in the grid*

See 21.2.2.1.4 Alias for MTF Modifiers

Start/End chainage smart chainage box Start (ref)/End (ref)

defines the start/end chainages for use in the Modifier.

For more information on Start/End Chainages and Modes, see 21.4 Smart Chainages

Interval real box

*if **not blank**, the interval to use to create cross sections and strings over the given chainage range. If **blank**, the interval given by any **MTF Edit** => **Interval commands**, and/or the* **Section separation** *value from the* **Apply MTF** *panel, is used.*

Extra start tick box tick

*if **ticked**, add an extra x-section a distance* **Extra start/value** *millimetres **before** the start chainage. See* 21.2.7.4 MTF Setting Show Extra Start/End *and* 21.2.7.5 MTF Setting Extra Start/End value.

Extra end tick box tick

*if **ticked**, add an extra x-section a distance* **Extra start/value** *millimetres **before** the end chainage. See* 21.2.7.4 MTF Setting Show Extra Start/End *and* 21.2.7.5 MTF Setting Extra Start/End value.

Comment text box

comment to add to the Comment column for the row of the grid.

Active tick box tick

*if **ticked**, use this modifier.*
*If **not ticked**, don't use this modifier.*

OK button

OK *stores the values in the fields and removes the panel BUT no **recalc** is done.*

Apply button

Apply *stores the values and leaves the panel on the screen.*

If the ***MTF*** *is being used in an* ***Apply MTF*** *and* ***Auto recalc*** *is ticked in the MTF, then whenever the* ***Apply*** *button is clicked, a* ***recalc*** *of the associated* ***Apply MTF*** *for the MTF is done.*

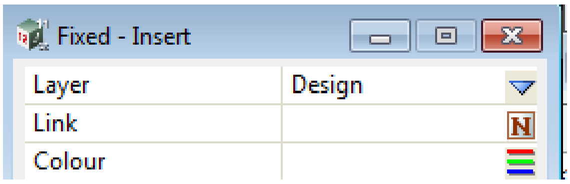
Go to the next section 21.2.2.1.2 Link or Link(s) or return to 21.2.2.1 Create MTF Commands

21.2.2.1.2 Link or Link(s)

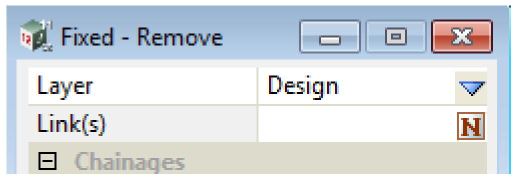
Some ***Modifier*** panels allow more than one link name to be specified.

If only one Link is allowed, then the panel field will be **Link**.

If more than on Link is allowed, then the panel field will be **Link(s)**.



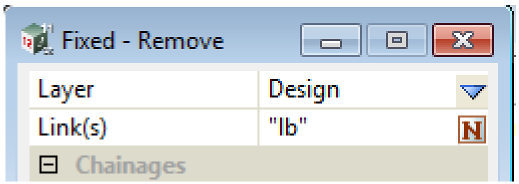
Option that allows only one Link name



Option that allows more than one Link name

When multiple Links are allowed, each Link name is separated by one or more spaces. If the Link name contains a space then it must be surrounded by double quotes. For example "name with a space".

When an MTF panel that allows multiple Link names is edited, all the Link names will be displayed surrounded by double quotes. This applies even if there is only one Link name in the field.



Go to the next section 21.2.2.1.3 Width, Height and Xfall Fields on MTF Modifier Panels or return to 21.2.2.1 Create MTF Commands

21.2.2.1.3 Width, Height and Xfall Fields on MTF Modifier Panels

When creating a link, the ***Modifier*** panels have the following common panels fields.

The fields and buttons have the following functions.

Field Description	Type	Defaults	Pop-Up
Width	real box		measure menu

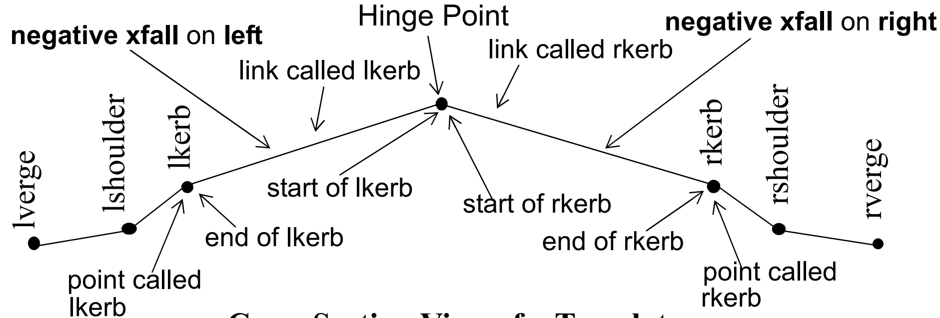
width of the link.

Height	real box		measure menu

the z-value of the end of the link minus the z-value of the start of the link. That is, the delta z for the link.

xfall	real box		measure menu

the crossfall of the link. This is the percent grade and positive is up and negative down when measured in the direction of the link. Note that for a link on the left side, the link goes from right to left. And for a link on the right hand side, a link goes from left to right.



Cross Section View of a Template

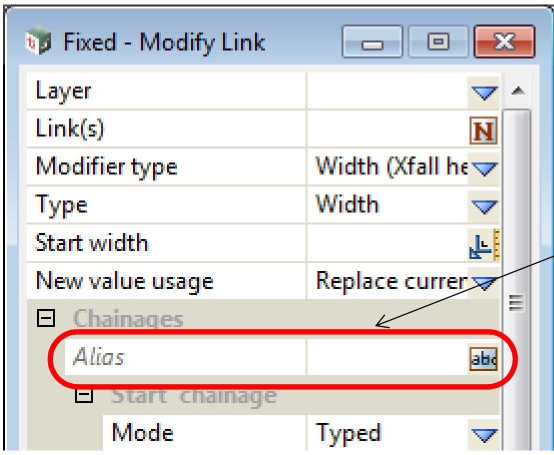
Go to the next section 21.2.2.1.4 Alias for MTF Modifiers or return to 21.2.2.1 Create MTF Commands

21.2.2.1.4 Alias for MTF Modifiers

For each *Left/Right Modifiers* Command, there is an optional text field called **Alias**.

An **Alias** can be used to uniquely refer to a particular command in the **Left** or **Right MTF Modifiers** grid **independently** of its row number.

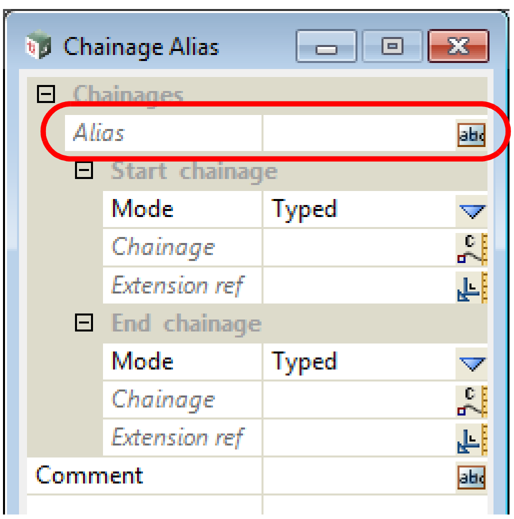
Most **MTF Commands** have an **Alias** field and if it is filled in then the value will be the **Alias** for that row.



If text is entered into the **Alias** field then that becomes the **Alias** for this row

There is also a special command **Chainage Alias** that defines an **Alias** for that row and those Start and End Chainages. See 21.2.2.4.1 Chainage Alias.

Apart from defining an **Alias** and associated **Start** and **End Chainages** for that row, **Chainage Alias** has no other effect.



If text is entered into the **Alias** field then that becomes the **Alias** for this row.

Once an **Alias** exists, it can be used in the two **Smart Chainage** modes, **Relative to Alias Start** and **Relative to Alias End**. See 21.4.19 Relative to Alias Start and 21.4.20 Relative to Alias End.

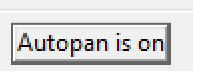
If an **Alias** is given for a command in the **Left MTF Modifiers** panel then it must be unique amongst all the Aliases for the **Left MTF Modifiers** commands.

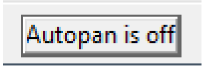
Similarly if an **Alias** is given for a command in the **Right MTF Modifiers** panel then it must be unique amongst all the Aliases for the **Right MTF Modifiers** commands.

Go to the next section 21.2.2.1.5 Autopan Button or return to 21.2.2.1 Create MTF Commands

21.2.2.1.5 Autopan Button

The **Autopan** button is at the button of the **Left/Right MTF Modifiers** panels and clicking on the button will toggle between the **on**/**off** states.

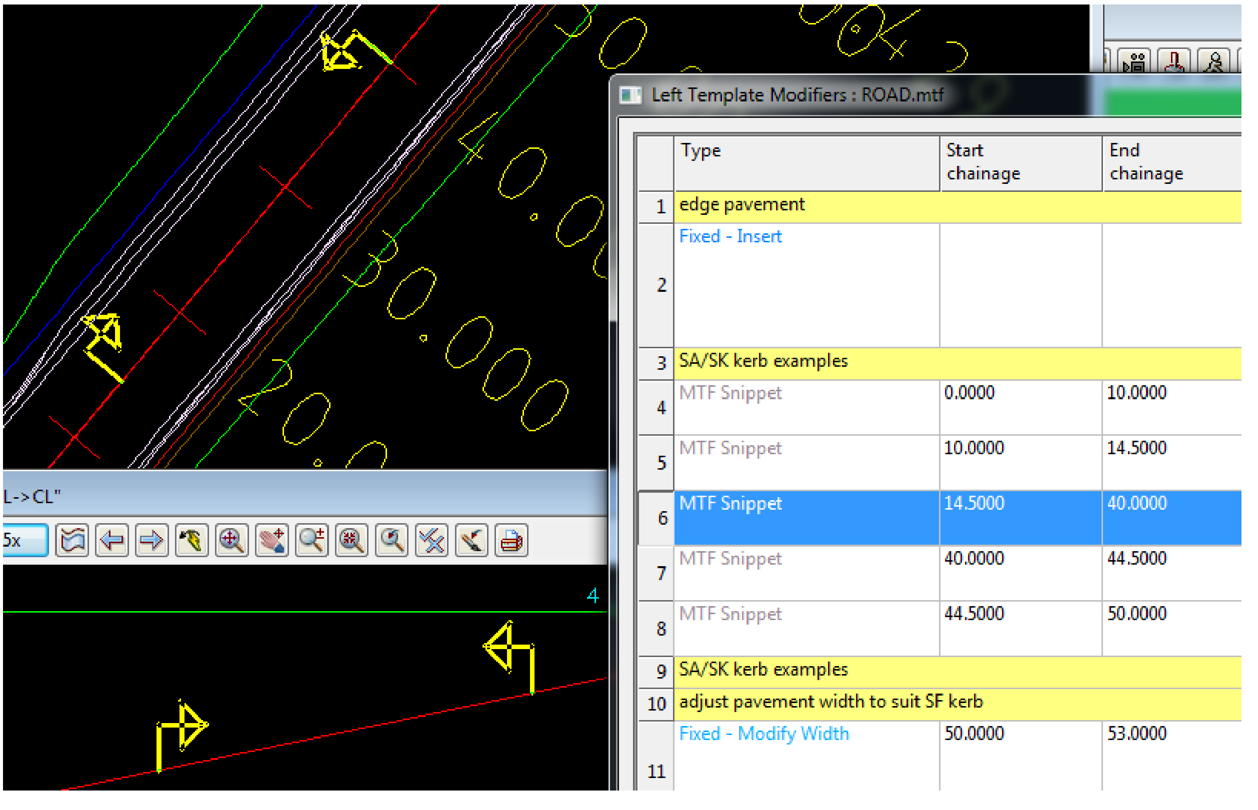




The **Autopan** button is used is conjunction with clicking LB in the Type, Start chainage and End Chainage columns.

For all **plan** views that the ***Reference string*** for the MTF is on, and all **section views** where the ***Reference string*** is ***profiled***:, clicking in the grid on the **Left/Right MTF Modifiers** panels will have the following effect:

(a) if LB is clicked on the ***number*** in the grid of a command then the entire line in the grid highlights, and highlight arrows are placed at the Start and End chainage of the command.

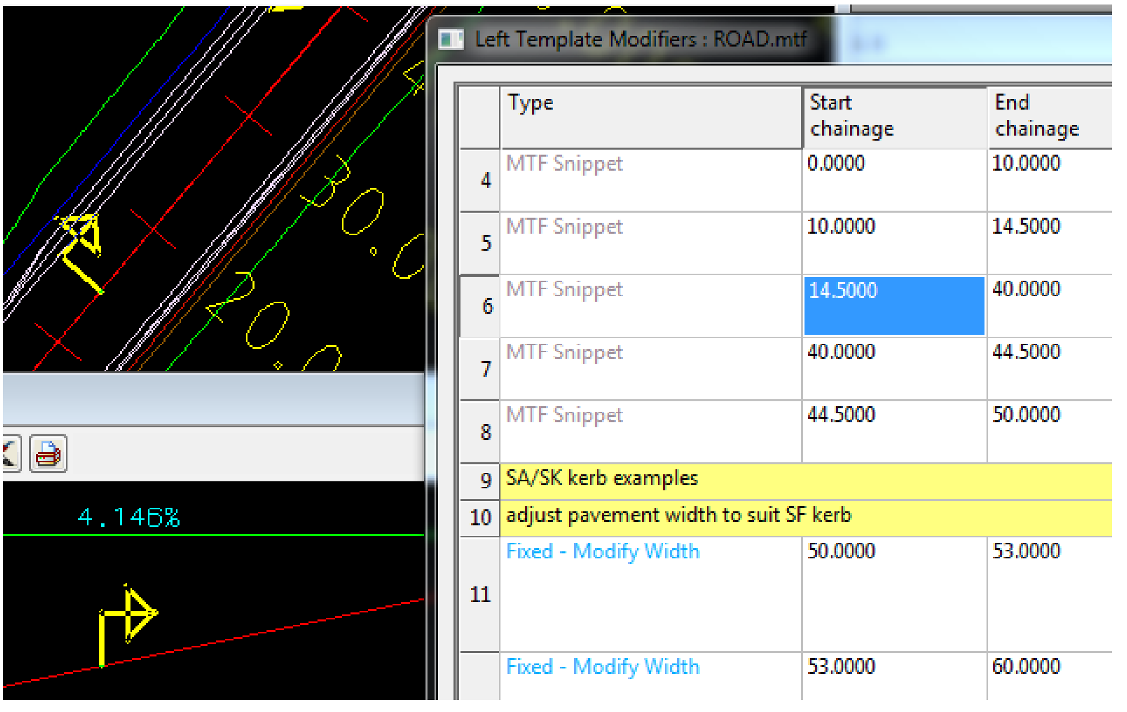


Important Note: the scale and the centre point of the views are not changed so if the Start/ End chainage is not visible on the view, then the highlight arrow for the Start/End will not appear.

(b) if LB is clicked in the ***Start chainage*** column of a command, then that cell of the grid highlights, and

If ***Autopan is on*** then the plan and/or section views will pan so that the Start chainage highlight arrow shows on the views.

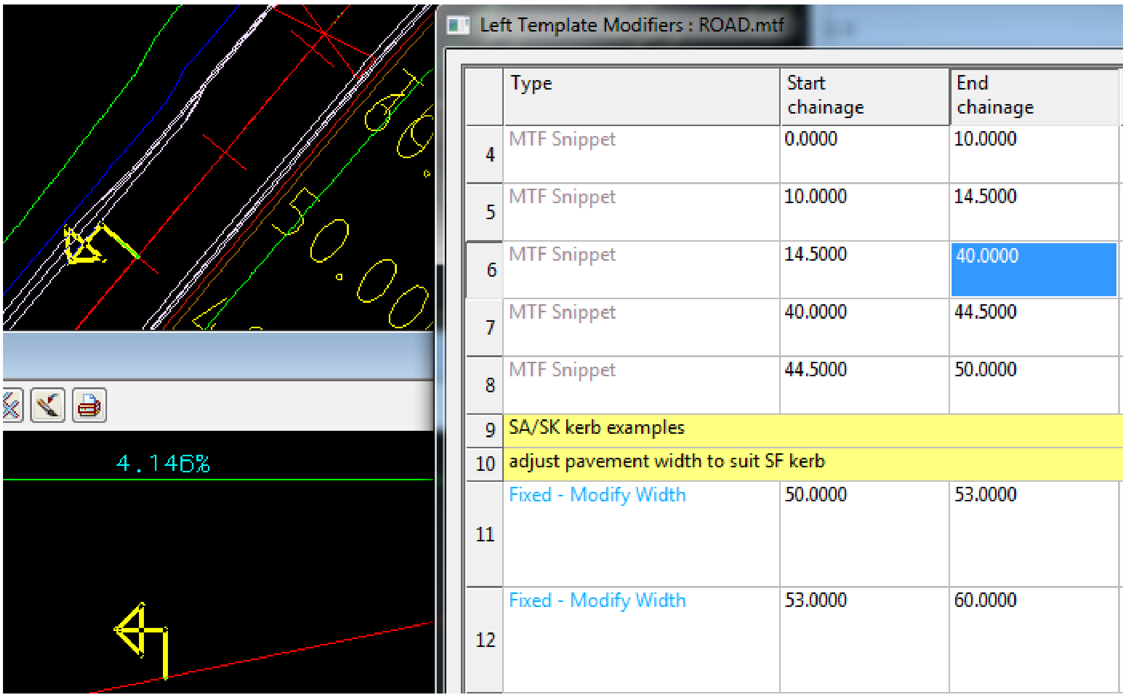
If ***Autopan is off*** then the Start chainage highlight arrow occurs but the view does not change. So the Start chainage highlight arrow will only show if the Start chainage is already visible on the plan and/or section views.



(c) if LB is clicked in the ***End chainage*** column of a command, then that cell of the grid highlights, and

If ***Autopan is on*** then the plan and/or section views will pan so that the End chainage highlight arrow shows on the views.

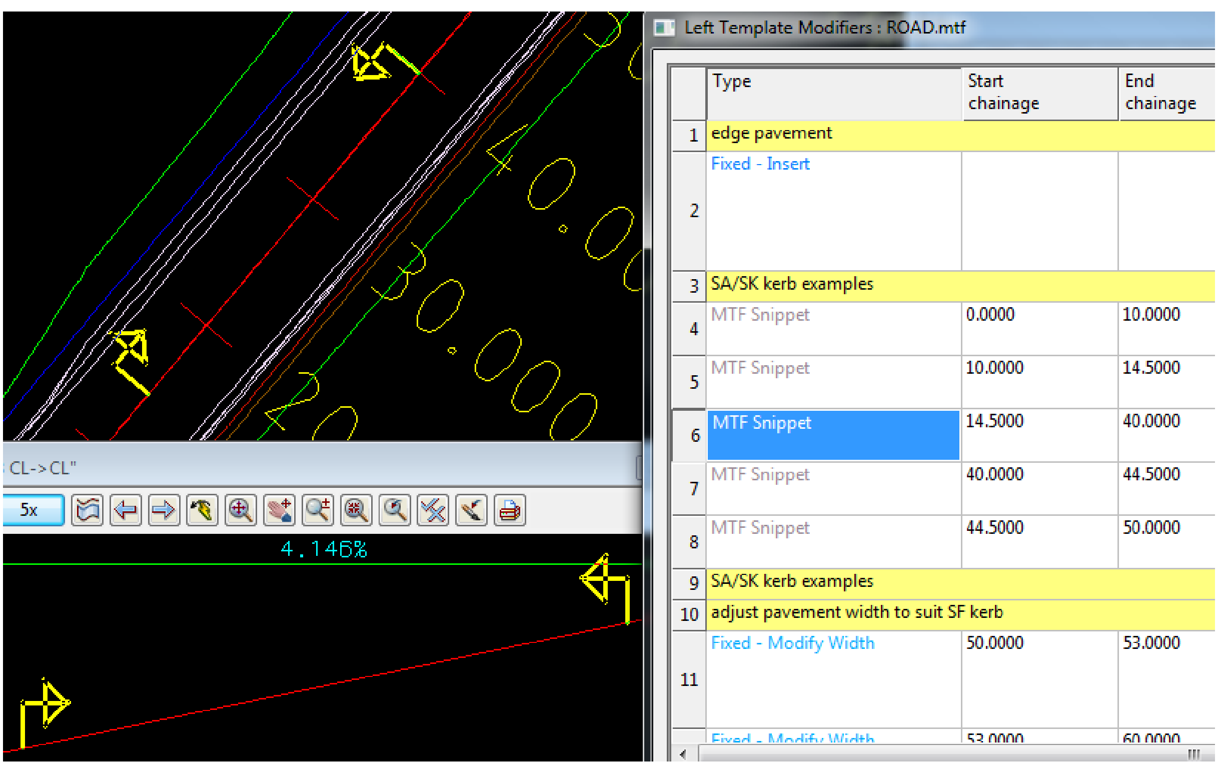
If ***Autopan is off*** then the End chainage highlight arrow occurs but the view does not change. So the End chainage highlight arrow will only show if the End chainage is already visible on the plan and/or section views.



(d) if LB is clicked in the ***Type*** column of a command, then that cell of the grid highlights, and

If ***Autopan is on*** then the plan and/or section views will do a fit about the Start chainage and End chainage so that both the Start chainage highlight arrow and the End chainage highlight arrow are visible in the plan and/or section views.

If ***Autopan is off*** then the plan and/or section views do not change. The Start chainage highlight arrow and the End chainage highlight arrow are turned on but they will only be visible if the Start chainage and/or End chainage are already visible on the views.



LJG??

reference string If ***Autopan is on*** is showing AND the ***Reference string*** for the MTF is on a plan view:

(a) whenever you click on ***Type*** column in the grid for a command that does not involve the entire string, the scale and centre of the view is modified so that the *Start* and *End* chainage extents of the command are displayed in the plan view (with the addition of an extra percentage that is set in env.4d)

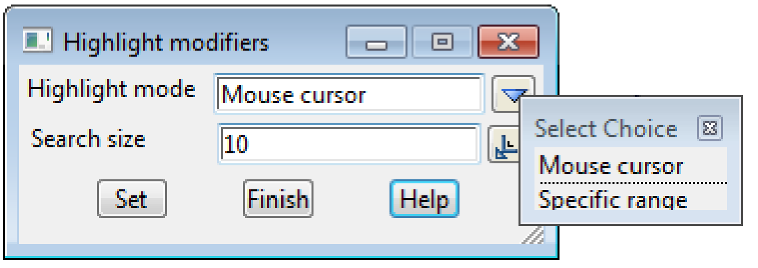
(b) whenever you click on ***Start* chainage** column of a command that does not involve the entire string, the scale and centre of the view is modified so that the *Start* and *End* chainage extents of the command are displayed in the plan view (with the addition of an extra percentage that is set in env.4d)

Go to the next section 21.2.2.1.6 Highlight Button or return to 21.2.2.1 Create MTF Commands

21.2.2.1.6 Highlight Button

Highlight button

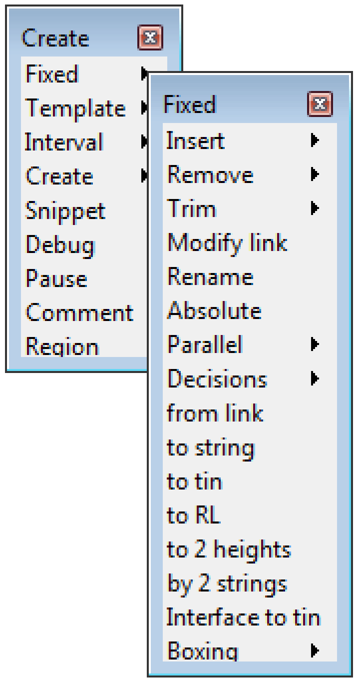
*clicking the **Highlight** button brings up the* **Highlight Modifiers** *panel.*



Return to 21.2 MTF Edit.

21.2.2.2 Fixed Link Modifiers

The **Fixed** walk-right brings up the **Fixed** menu with options to insert and modify the fixed links.

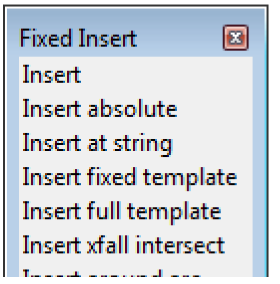


For

Insert	21.2.2.2.1 Fixed Link - Insert
Remove	21.2.2.2.2 Fixed Link - Remove
Trim	21.2.2.2.3 Fixed - Trim
Modify link	21.2.2.2.4 Fixed - Modify Link
Rename	21.2.2.2.5 Fixed Link - Rename Link
Absolute	21.2.2.2.6 Fixed Link - Absolute to a Base Link
Parallel	21.2.2.2.7 Fixed Link- Parallel
Decisions	21.2.2.2.8 Fixed Link - Decisions
from link	21.2.2.2.9 Fixed Link - From link
to string	21.2.2.2.10 Fixed Link - To String
to tin	21.2.2.2.11 Fixed Link - To Tin
to RL	21.2.2.2.12 Fixed - To RL
to 2 heights	21.2.2.2.13 Fixed - to 2 Heights
to 2 strings	21.2.2.2.14 Fixed - by 2 strings
Interface to tin	21.2.2.2.15 Fixed - Interface to Tin
Boxing	21.2.2.2.16 Fixed - Boxing

21.2.2.2.1 Fixed Link - Insert

The **Fixed Insert** walk-right brings up the **Fixed Insert** menu with options to insert fixed links, insert a fixed part of a template, and insert a full template.



For

Insert	21.2.2.2.1.1 Insert a Fixed Link
Insert absolute	21.2.2.2.1.2 Insert a Fixed Link Absolute
Insert at string	21.2.2.2.1.3 Insert a Fixed Link at a String
Insert fixed template	21.2.2.2.1.4 Insert Fixed Links from a Template
Insert full template	21.2.2.2.1.5 Insert Full Template
Insert xfall intersect	21.2.2.2.1.6 Fixed Link - Insert Xfall Intersect

21.2.2.2.1.1 Insert a Fixed Link

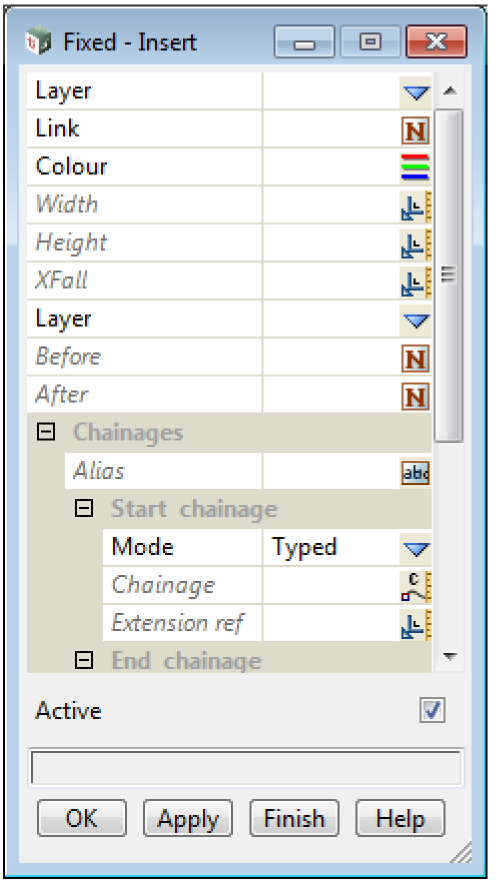
Fixed links can be inserted by specifying either width and height, width and xfall, or height and xfall, where width/height/xfall are **measured** from the **last fixed link before** this link.

Note - height is not a z-value (RL) but the difference in z-values between the start and the end of the link.

It doesn't matter which of the two is used to define the link, the third (plus slope) can be calculated.*See* 21.2 MTF Edit.

So the link can be modified by varying the width, xfall, height or slope.

Selecting **Insert** brings up the **Fixed - Insert** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box		available Layers

Layer for the new link. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Link	text box		

name for the new link.

Colour	colour box		available colours

colour of the link being created.

Width /Height/Xfall	real box		measures menu

width/ height/ crossfall of the link being created - only use two of the three.

For more information on these panel fields, see 21.2.2.1.3 Width, Height and Xfall Fields on MTF Modifier Panels.

Before choice box select name menu

*if **not blank**, the name of the link to insert the new link before.*
*If **blank** and **After** is **blank**, the link is appended to the end of the fixed part of the template.*

After choice box select name menu

*if **Before** is **not blank**, **After** is ignored.*
*If **not blank** and **Before** is **blank**, the name of the link to insert the new link after.*
*If **blank** and **Before** is **blank**, **Before** is used.*
***Note**: **One** of **Before** or **After** must be **blank**. If they are both blank then **Before** takes precedence over **After**.*

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages for inserting the new link

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Go to the next section 21.2.2.2.1.2 Insert a Fixed Link Absolute or return to 21.2.2.2.1 Fixed Link - Insert.

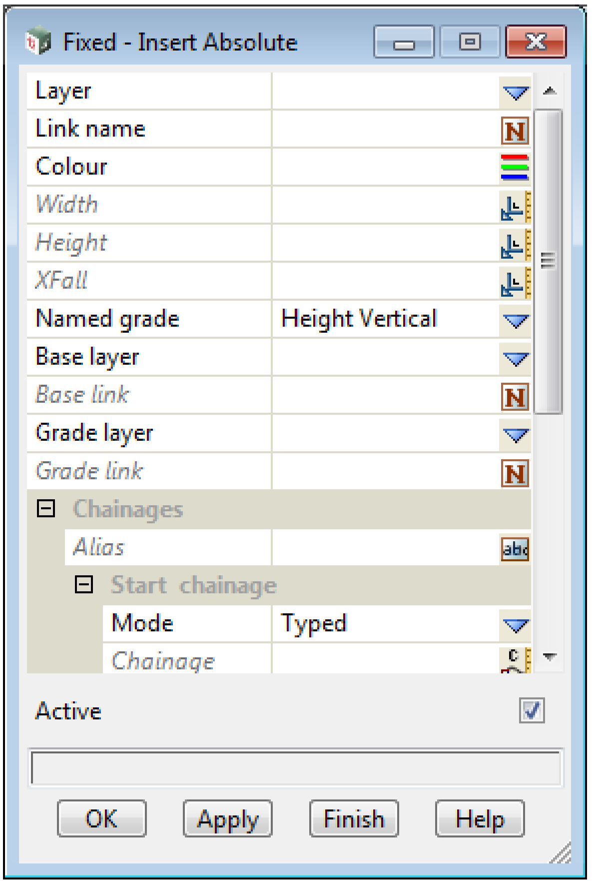
21.2.2.2.1.2 Insert a Fixed Link Absolute

Fixed links can be inserted by specifying either width and height, width and xfall, or height and xfall, where the width/height/xfall are **measured from a Base link,** or a **combination** of a **Base** link and a **Grade** link.

The new inserted link must end up outside any previous links.

Note - height is not a z-value (RL) but a delta height.

Selecting **Insert absolute** brings up the **Fixed - Insert Absolute** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box		available Layers

Layer for the new link. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Link name	names box		select name menu

name for the new link.

***Note**: the new inserted link must end up outside any previous links.*

Colour	colour box		available colours

colour of the link being created.

Width /height/xfall	input		measures menu

width/ height/ crossfall of the link being created - only use two of the three.

For more information on these panel fields, see 21.2.2.1.3 Width, Height and Xfall Fields on MTF Modifier Panels.

If a Grade link is given, only the combinations width and crossfall, or width and height, are allowed.

If there is no Grade link, any two of the three can be used.

Base link, Grade link names box select name menu

*If **Grade link is blank**:*

any combination of two of width, height an xfall can be used to define the new link.

if Base link is non blank, then it is the name of the link to take the width/height/xfall from.

If Base link is blank, then the option acts as a normal Insert and the width/height/xfall are taken from the last fixed link before this insert.

*If **Grade link is not blank**:*

only the combinations width and crossfall, or width and height, are allowed.

Width:

*if Base link is non blank, then it is the name of the link to take the **width** from.*

*If Base link is blank, then the **width** is from the last fixed link before this insert.*

Xfall or Height

*If **xfall** is given, then the xfall is applied from the Grade link.*

*If **height** is given, then the height is added to the height of the Grade link.*

***Note**: the new inserted link must end up outside any previous links.*

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages for inserting the new link

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

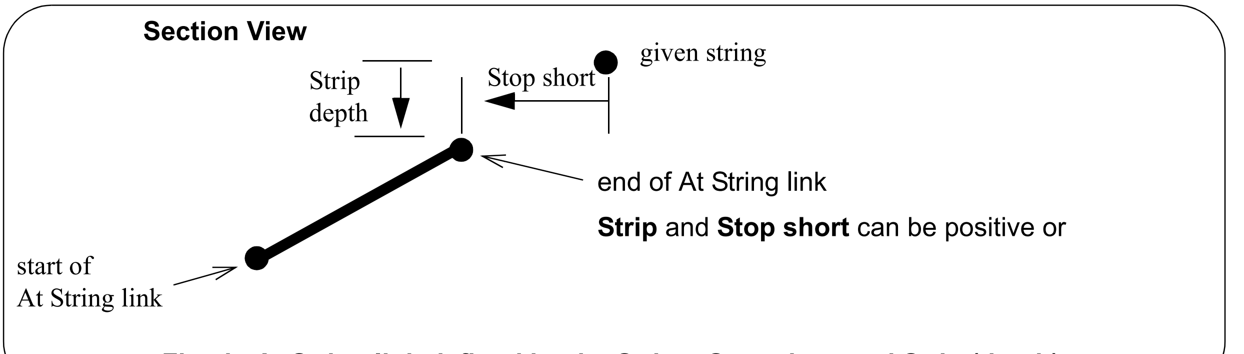
Go to the next section 21.2.2.2.1.3 Insert a Fixed Link at a String or return to 21.2.2.2.1 Fixed Link - Insert.

21.2.2.2.1.3 Insert a Fixed Link at a String

The **Insert at string** modifier copies an existing string between the two given chainages and inserts it as a link of the template.

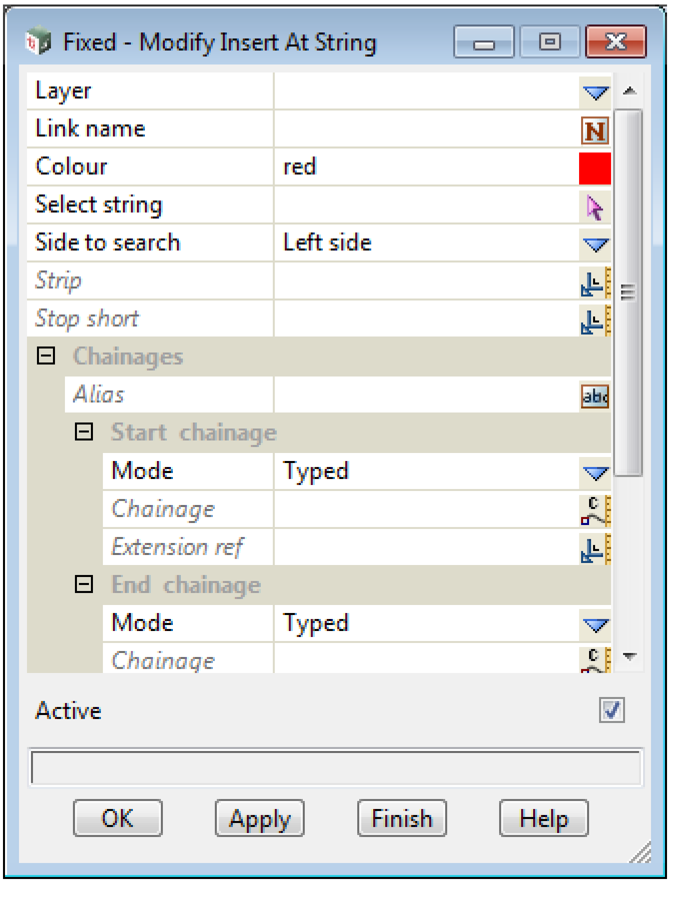
The **Insert at string** copies an existing string between the two given chainages and inserts it as a link of the template. The created link can also be given an offset and depth from the string.

Hence the end of the new link is constructed so that it is a given plan distance **Stop short** from the ***string***, and stops at the depth **Strip *below*** the ***string***.



Fixed - At String link defined by the String, Stop short and Strip (depth).

Selecting **Insert at string** brings up the **Fixed - Modify Insert at String** panel



The new fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

New link type choice box Width and Xfall, Width and Height, Height and Xfall

states how the link is to be defined.

Layer layer box available Layers

Layer for the new link. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Link name names box select name menu

name for the new link.

Colour colour box available colours

colour for the new link

Strip real box measures menu

the new link is constructed so that its end is **Strip** *below the selected string.*

Strip *is a depth and is positive in the downward direction It can be positive or negative.*

Stop short real box measures menu

the new link is constructed so that its end is the plan distance **Stop short** *of the selected string.*

When the command is a Left Modifier, positive plan distance is measured from the Hinge string as zero and then going out to the left.

When the command is a Right Modifier, positive plan distance is measured from the Hinge string as zero and then going out to the right.

Stop short *can be positive or negative.*

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages for inserting the new link

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Select string string select box

select the string to copy as a link

Side to search choice box Left side, Right side, Both sides

side to search to find the string.

Comment, **Extra start**, **Extra End, Active**

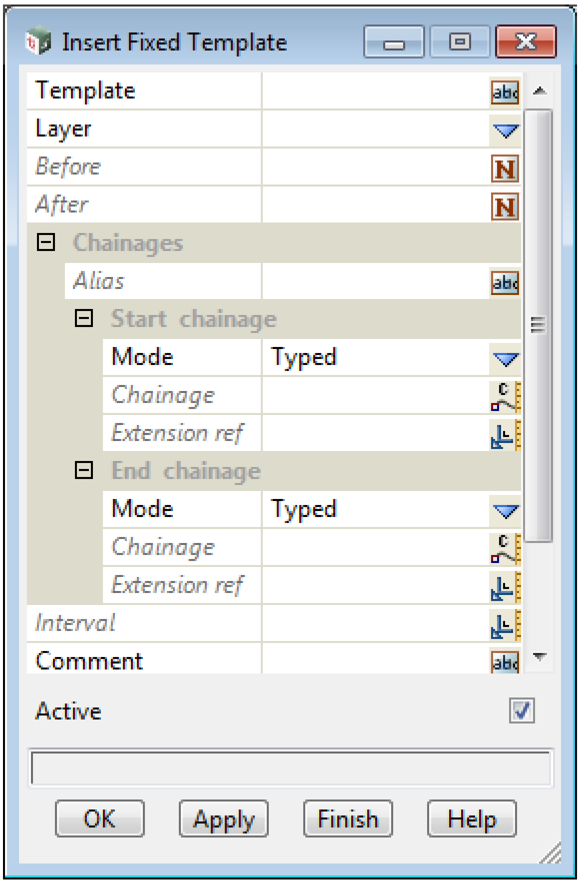
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Go to the next section 21.2.2.2.1.4 Insert Fixed Links from a Template or return to 21.2.2.2.1 Fixed Link - Insert.

21.2.2.2.1.4 Insert Fixed Links from a Template

The **Insert fixed template** modifier inserts all the fixed links from an existing Template.

Selecting **Insert fixed template** brings up the **Insert Fixed Template** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Template	template box		select template menu

the name of the Template to take the fixed links from.

Layer	layer box		available Layers

Layer for the new links. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Before	choice box		select name menu

*if **not blank**, the fixed links from the selected template are inserted before the given link.*
*If **blank** and **After** is **blank**, the fixed links from the selected template are appended to the end of the existing fixed links.*

After	choice box		select name menu

*if **Before** is **not blank**, **After** is ignored.*
*If **not blank** and **Before** is **blank**, the fixed links from the selected template are inserted after the given link.*
*If **blank** and **Before** is **blank**, **Before** is used.*
***Note**: **One** of **Before** or **After** must be **blank**. If they are both blank then **Before** takes precedence over **After**.*

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages for inserting the new fixed links

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

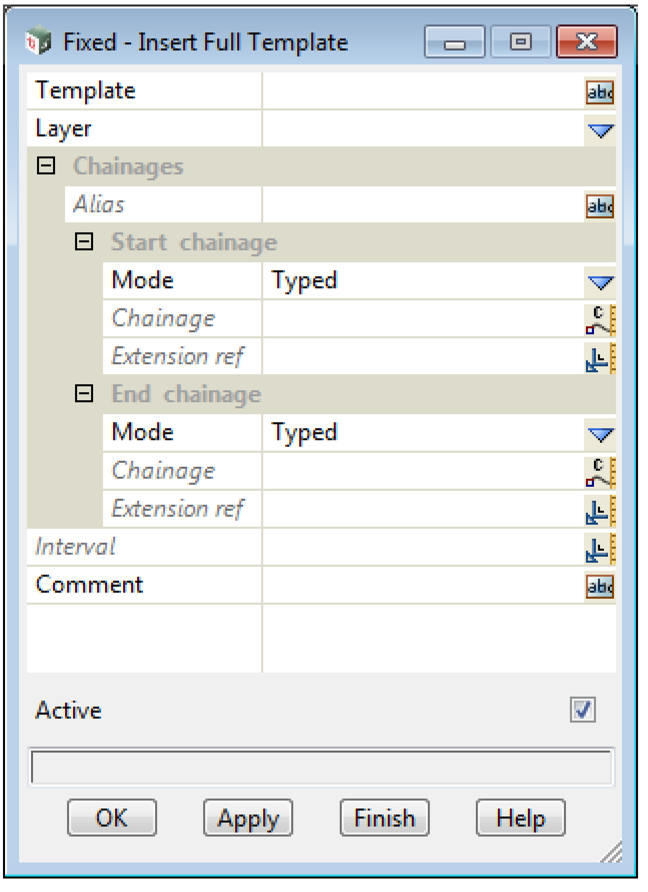
Go to the next section 21.2.2.2.1.5 Insert Full Template or return to 21.2.2.2.1 Fixed Link - Insert.

21.2.2.2.1.5 Insert Full Template

The **Insert Full Template** ***inserts*** all the fixed links from a selected template after the existing fixed links **AND** also ***replaces*** the decision, cut, fill and final cut/fill links of the existing template with those from the selected template.

So the option **leaves** the existing fixed links from the template but **removes the existing** decision, cut, fill and final cut/fill links in the template.

Selecting **Insert Full Template** brings up the **Fixed - Insert Full Template** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Template	template box		select template menu

the name of the template to take the fixed, decision, cut, fill and final cut/fill links from.

Layer	layer box		available Layers

Layer for the new links. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages for inserting the new fixed links

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Go to the next section 21.2.2.2.1.6 Fixed Link - Insert Xfall Intersect or return to 21.2.2.2.1 Fixed Link - Insert.

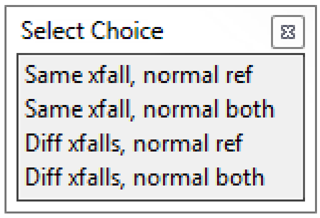
21.2.2.2.1.6 Fixed Link - Insert Xfall Intersect

The **Xfall insert** modifier **inserts** a new link.

And the new link is created by coming off the last link of the fixed template before the ***Insert Xfall Intersect*** command, and going out from that last fixed link in a variety of ways and meeting a **selected string**.

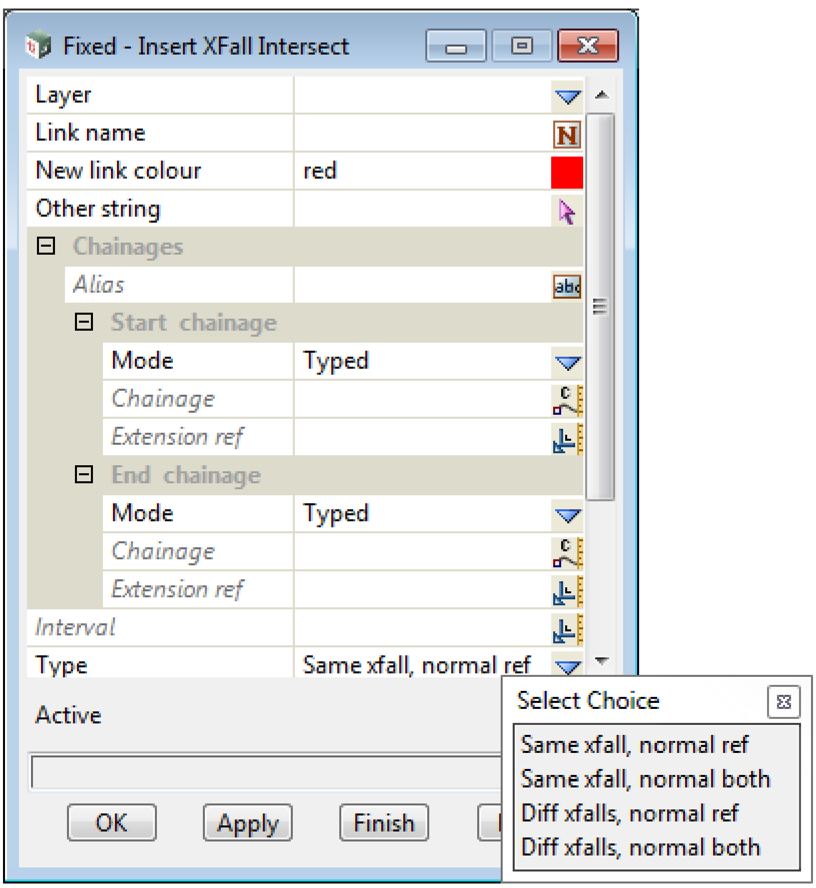
For example, over the chainage range, a given xfall is taken from the last link and the selected string, and the new link is the intersection of the two xfalls.

The different methods for creating the new link are:



and for the definitions go to 21.2.2.2.1.6.1 Fixed - Calculating the New Link for each Type.

Selecting **Insert xfall intersect** brings up the **Fixed - Xfall Intersect** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box		available Layers

Layer for the new link. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Field Description	Type	Defaults	Pop-Up
Link name	name box		available names

name of the new link.

New link colour colour box available colours

colour of the new link.

Other String string select

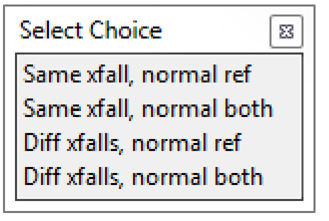
select the secondary string to use in the methods for creating the new link.

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages for inserting the new fixed links

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Type choice box



For the types of ways to create the new link, go to 21.2.2.2.1.6.1 Fixed - Calculating the New Link for each Type

Xfall, Xfall hinge, Xfall sec etc

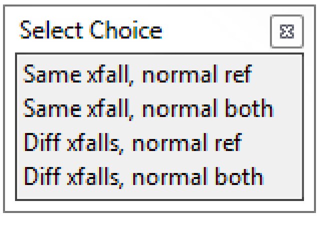
which of these fields are present, and what they are for, depends on **Type**.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Return to 21.2.2.2.1 Fixed Link - Insert.

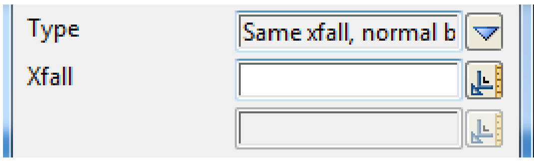
21.2.2.2.1.6.1 Fixed - Calculating the New Link for each Type



for the definitions of the calculations for each choice

go to Same Xfall and Normal to Reference String
Same Xfall and Normal to Reference String and to Other String
Two Xfalls and Normal to Reference String
Two Xfalls and Normal to Reference String and to Other String

Same Xfall and Normal to Reference String

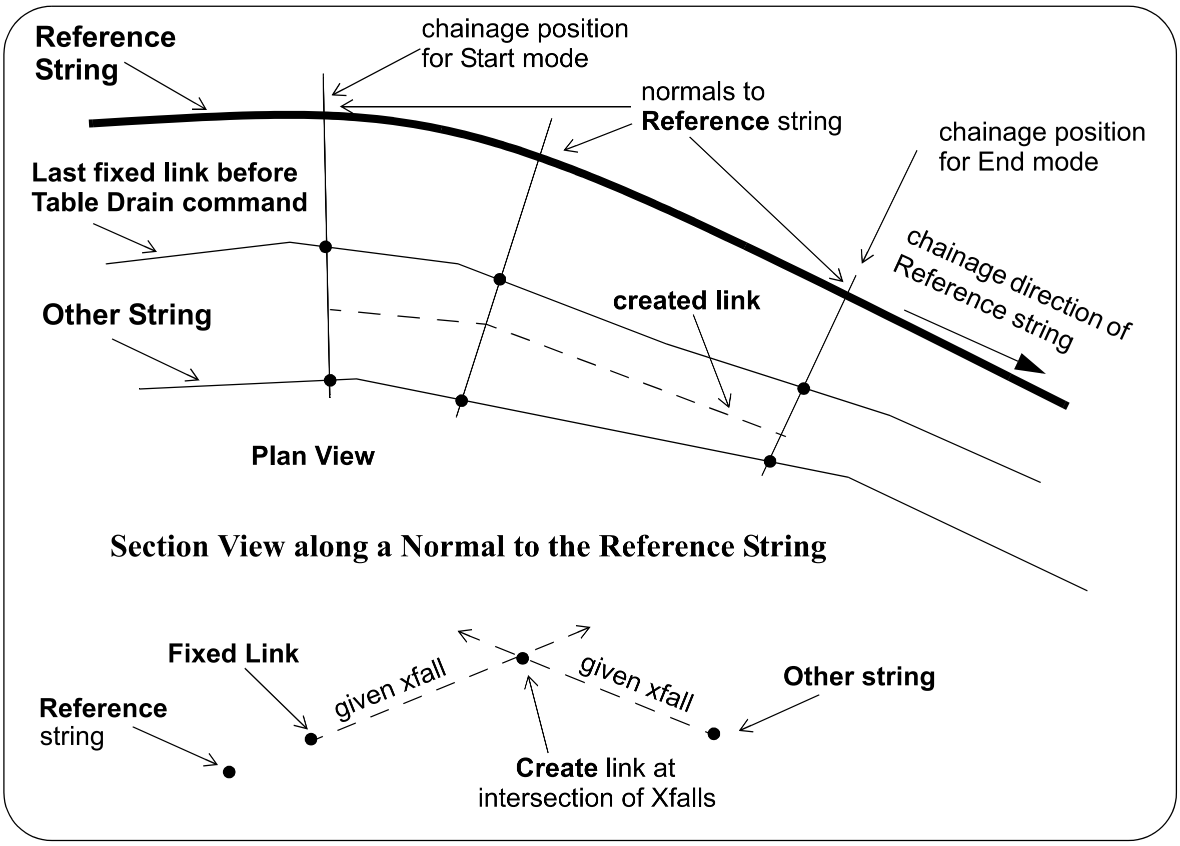


This option requires a **Xfall**.

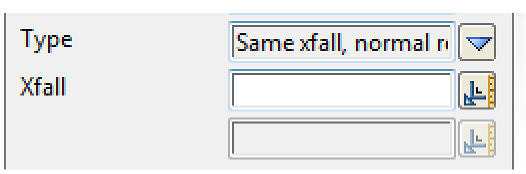
The **Start mode** and **End Mode** refer to **Reference** string and at each chainage (ch) between the Start mode and the End mode, a section is taken **normal** (perpendicular) to the **Reference string** and cutting the **last fixed link before the *Table Drain* command**, and the **Other string**.

On this section, the given xfall **Xfall** is taken from the last fixed link, and the xfall of **negative Xfall** from **Other string**, and a vertex of the new link created where the xfalls intersect.

For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.



Same Xfall and Normal to Reference String and to Other String



This option requires a **Xfall**.

The **Start mode** and **End Mode** refer to **Reference** string and at each chainage ch between the Start mode and the End mode, a section is taken **normal** (perpendicular) to the **Reference**

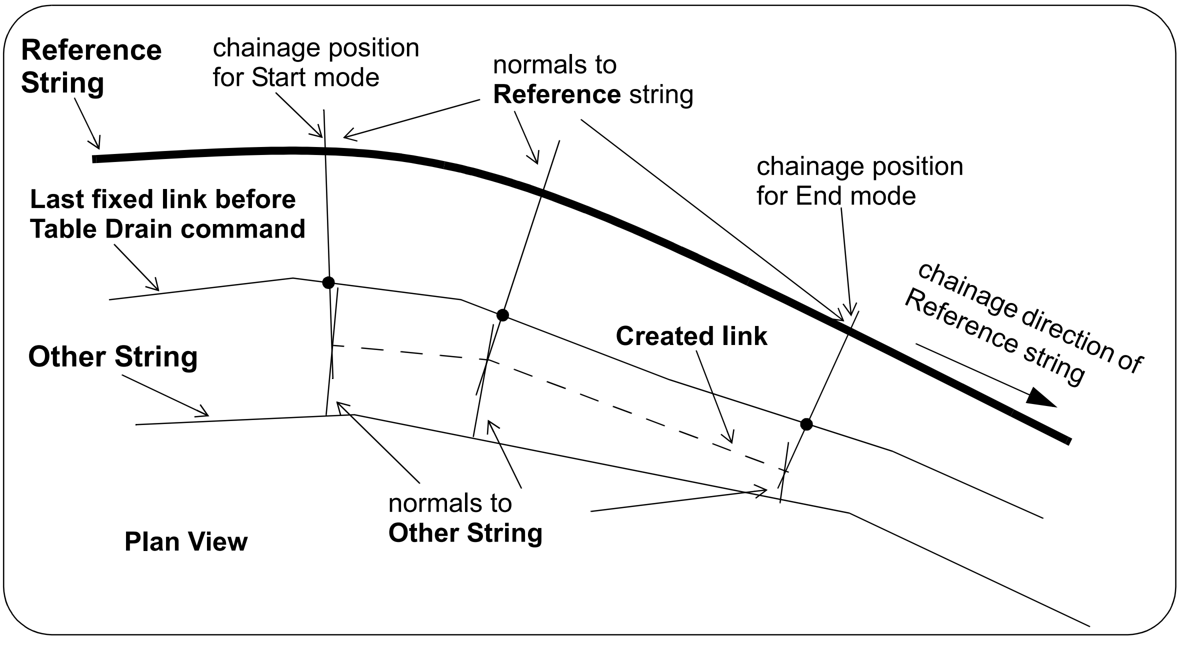
string and cutting the **last fixed link before the *Table Drain* command**, and the xfall **Xfall** is taken from the cut with the last fixed link and along the section.

This is then intersected with a section **normal** to the **Other string** and with the xfall **negative Xfall**.

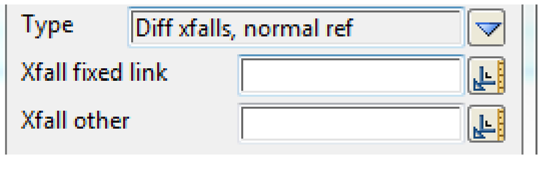
The new link is formed from the intersection points.

Note - it may not be possible to find such an intersection point.

For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.



Two Xfalls and Normal to Reference String

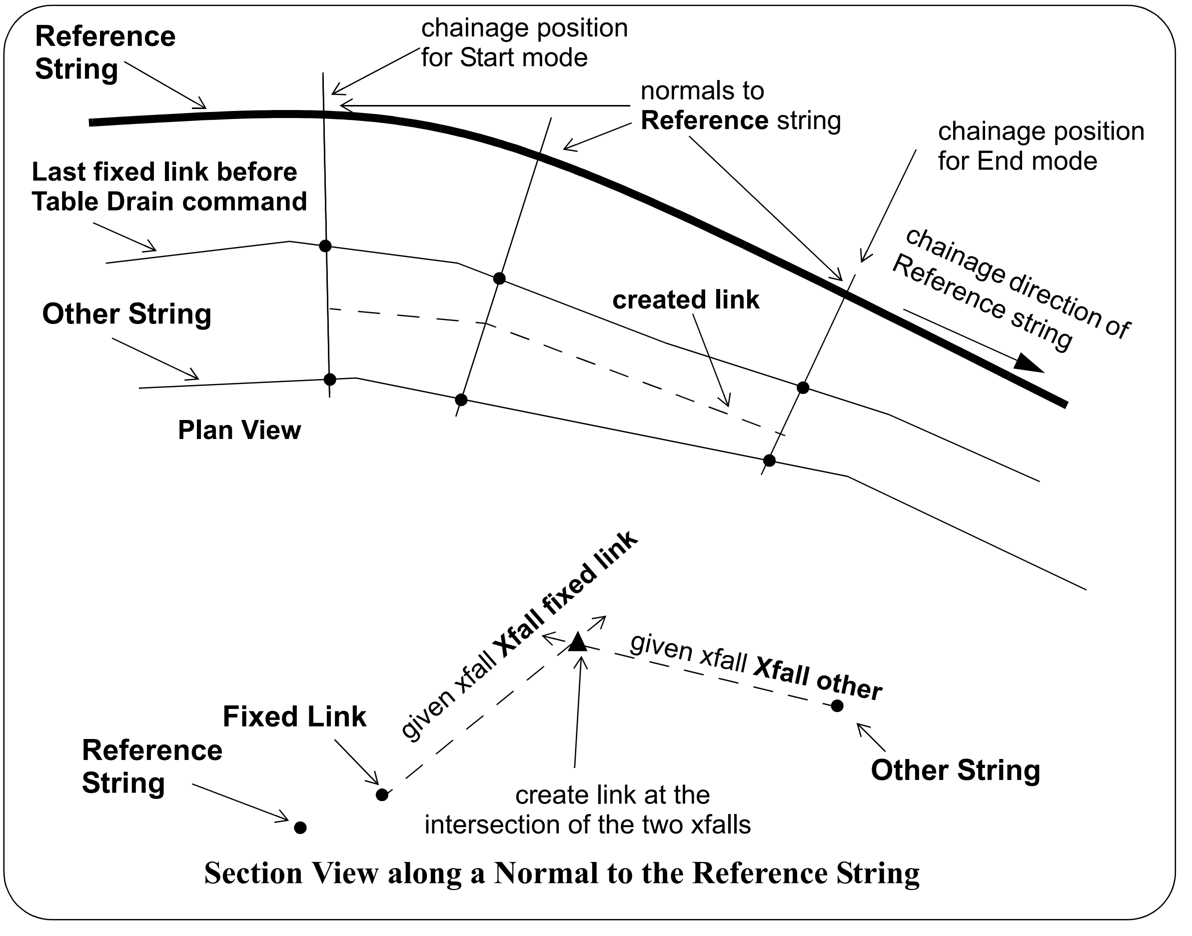


This option requires to xfalls: **Xfall fixed link** and **Xfall other**.

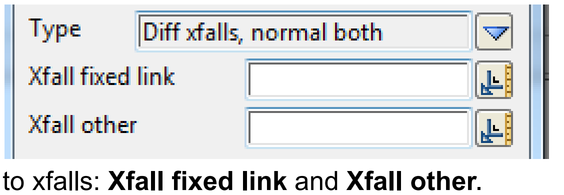
The **Start mode** and **End Mode** refer to **Reference** string and at each chainage (ch) between the Start mode and the End mode, a section is taken **normal** (perpendicular) to the **Reference string** and cutting the **last fixed link before the *Table Drain* command**, and the **Other string**.

On this section, the given xfall **Xfall fixed link** is taken from the last fixed link, and the xfall of **Xfall other** from **Other string**, and a vertex of the new link created where the xfalls intersect.

For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.



Two Xfalls and Normal to Reference String and to Other String



This option requires to xfalls: **Xfall fixed link** and **Xfall other**.

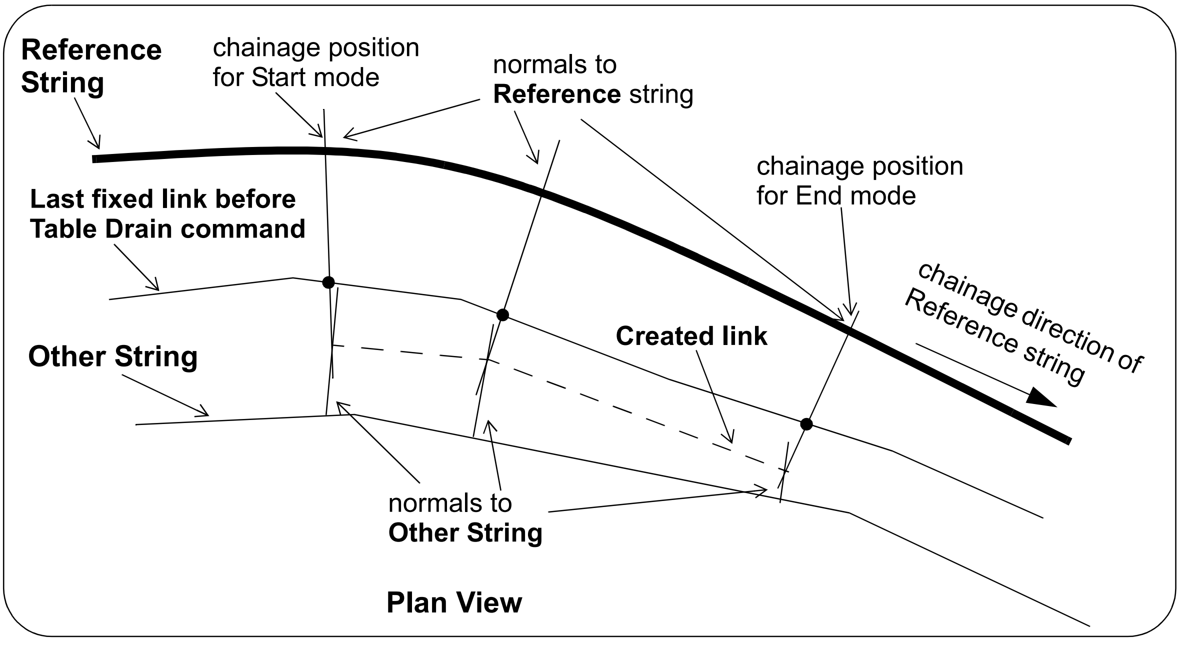
The **Start mode** and **End Mode** refer to **Reference** string and at each chainage ch between the Start mode and the End mode, a section is taken **normal** (perpendicular) to the **Reference string** and cutting the **last fixed link before the *Table Drain* command**, and the xfall **Xfall fixed link** is taken from the cut with the last fixed link and along the section.

This is then intersected with a section **normal** to the **Other string** and with the xfall **Xfall other**.

The new link is formed from the intersection points.

Note - it may not be possible to find such an intersection point.

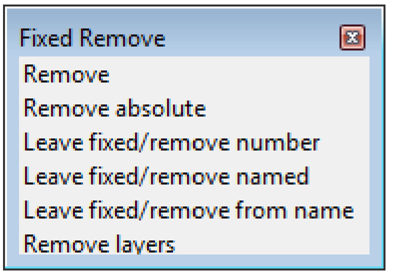
For information about the sign convention for offset and xfalls, go to 20.8.11.2 Sign Convention for Heights, Offsets and Xfalls in Create by Xfall and Grade.



Continue to the next section 21.2.2.2.2 Fixed Link - Remove or return to 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

21.2.2.2.2 Fixed Link - Remove

The **fixed Remove** walk-right brings up the **fixed Remove** menu with options to remove links.



To go straight to the documentation on each of the options on the **Fixed** modifier menu:

See

Remove 21.2.2.2.2.1 Remove a Fixed link

Remove absolute 21.2.2.2.2.2 Remove a Fixed link - Absolute

Leave fixed/remove number 21.2.2.2.2.3 Fixed Link - Remove All But Fixed & Remove a Number of Fixed Links

Leave fixed/remove named 21.2.2.2.2.4 Fixed Link - Remove All But Fixed & Remove Named Fixed Links

Leave fixed/remove from name 21.2.2.2.2.5 Fixed Link - Remove All But Fixed & Remove Fixed Links from a Named Link

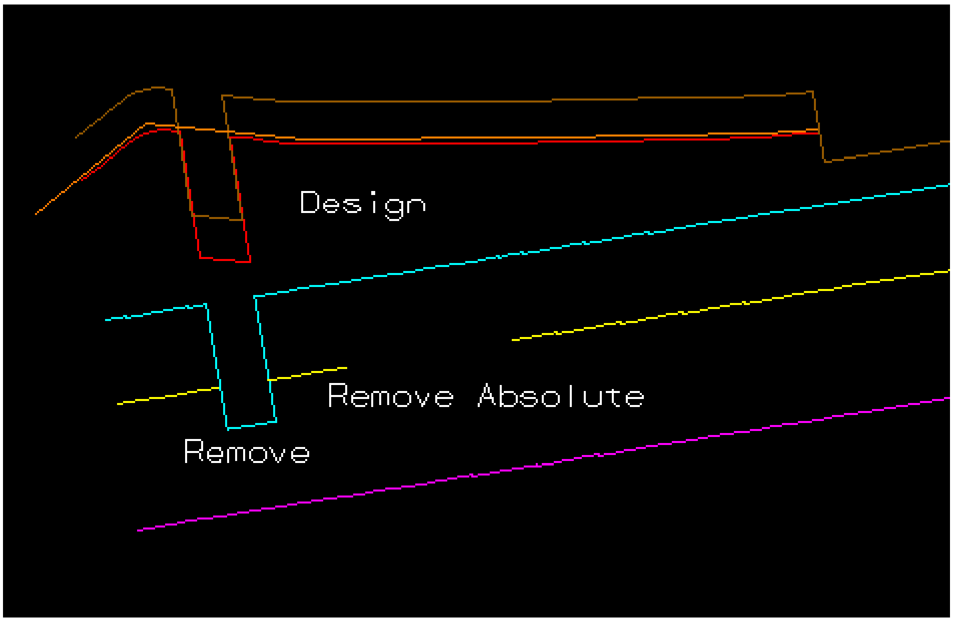
Remove layers 21.2.2.2.2.6 Fixed Link - Remove Layer

21.2.2.2.2.1 Remove a Fixed link

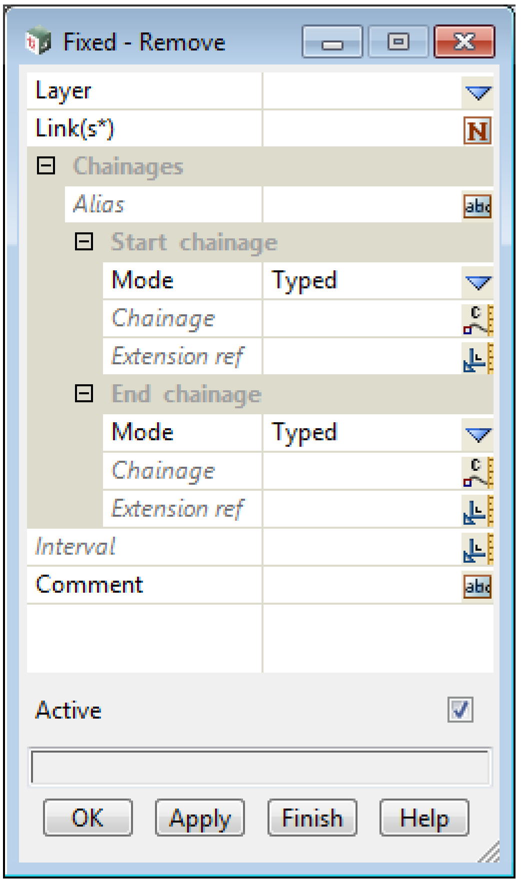
The **Remove** option deletes fixed links between given chainages.

Because the link is no longer there, any following links will then be relative to the link before the deleted link.

Note that there is also a **Remove Absolute** option that deletes a link but leave the absolute position of the other links undisturbed. See 21.2.2.2.2.2 Remove a Fixed link - Absolute.



Selecting **Remove** brings up the **Fixed - Remove** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box		available Layers

Layer to remove the link. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Field Description	Type	Defaults	Pop-Up
Link(s)	name box		select name menu

name of the links to remove. See 21.2.2.1.2 Link or Link(s).

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages for removing the given fixed links.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

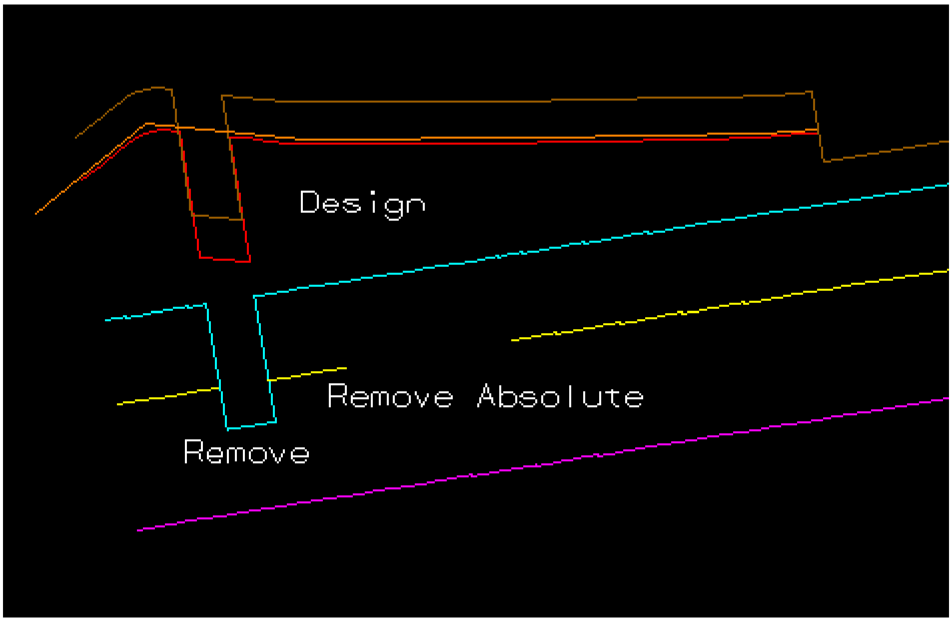
Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

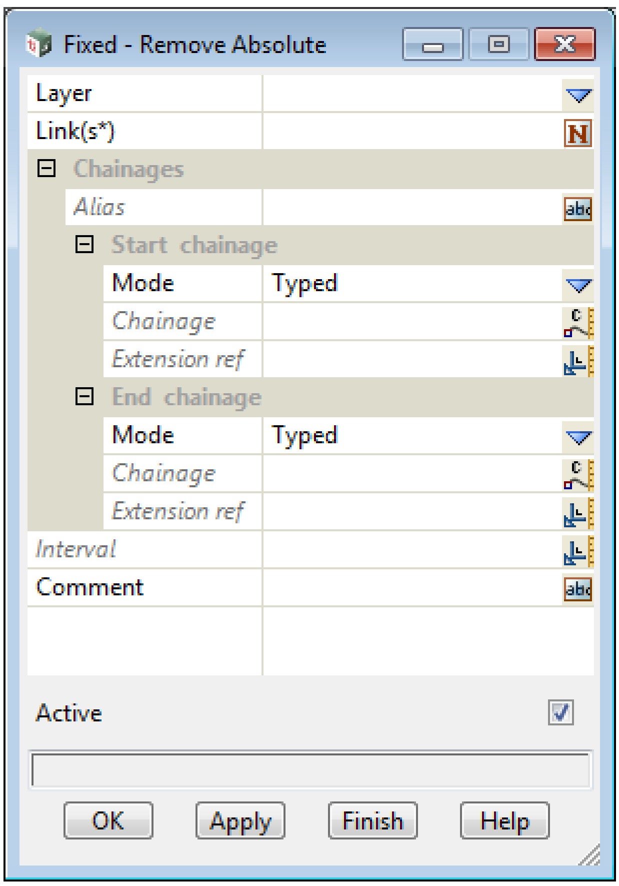
Go to the next section 21.2.2.2.2.2 Remove a Fixed link - Absolute or return to 21.2.2.2.2 Fixed Link - Remove.

21.2.2.2.2.2 Remove a Fixed link - Absolute

The **Remove Absolute** option deletes fixed links between given chainages whilst leaving the absolute position of the other links undisturbed.



Selecting **Remove absolute** brings up the **Fixed - Remove Absolute** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box		available Layers

Layer to remove the link. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Link(s)	name box		select name menu

name of the links to remove. See 21.2.2.1.2 Link or Link(s).

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages for removing the given fixed links.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

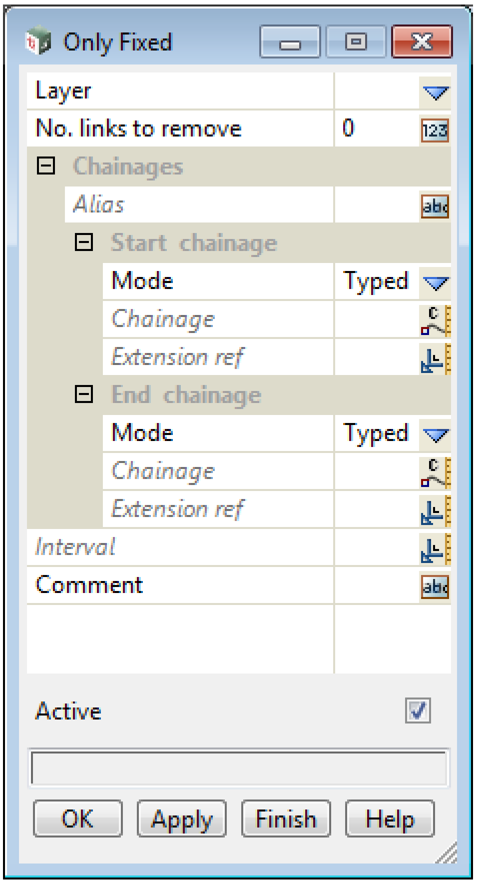
Go to the next section 21.2.2.2.2.3 Fixed Link - Remove All But Fixed & Remove a Number of Fixed Links or return to 21.2.2.2.2 Fixed Link - Remove.

21.2.2.2.2.3 Fixed Link - Remove All But Fixed & Remove a Number of Fixed Links

The **Only Fixed** option removes any of the variable cut and fill, final cut/fill links and decisions that have been defined up to this point in the modifiers grid.

Also starting from the outermost fixed link, it can also removed one or more of the remaining fixed links.

Selecting **Only Fixed** brings up the **Only Fixed** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages to remove the cut, fill and final cut/fill links, and some fixed links.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Layer	layer box		available Layers

Layer to remove the links from. For information on Layers, see 21.1.1.2 MTF Links and Layers.

No. links to remove	integer box	0	

*if **greater than zero**, then starting from the outside link of the fixed links, this number of fixed links are removed. If the number is larger than the number of fixed links, then it removes all the fixed links. So a large number will remove all the links - fixed, cut, fill and final cut/fill.*

*If **zero** then no fixed links are removed.*

*If **less than zero**, then starting from the first link of the fixed links, this number of fixed links are kept and the outer fixed links are removed. If the number is larger than the number of fixed links, then it* *removes all the fixed links.So a large number will remove all the links - fixed, cut, fill and final cut/fill.*

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

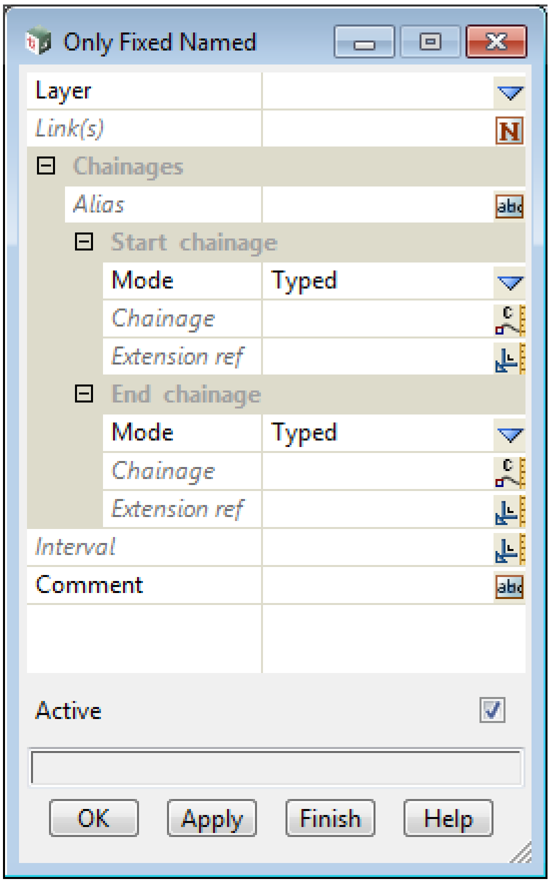
Go to the next section 21.2.2.2.2.3 Fixed Link - Remove All But Fixed & Remove a Number of Fixed Links or return to 21.2.2.2.2 Fixed Link - Remove.

21.2.2.2.2.4 Fixed Link - Remove All But Fixed & Remove Named Fixed Links

The **Only Fixed Named** option removes any of the variable cut and fill, final cut/fill links and decisions that have been defined up to this point in the modifiers grid.

It also removes a list of fixed links from the remaining fixed links.

Selecting **Only Fixed Named** brings up the **Only Fixed Named** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box		available Layers

Layer to remove the links from. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Link(s)	name box		names.4d list

*if **not blank**, one or more link names can be entered separated by spaces. If the link name contains spaces then the link name must be surrounded by the quotes ". For example " EB 1". See* 21.2.2.1.2 Link or Link(s).

All the links listed in the **Link(s)** *field are removed from the fixed links.*

*If **blank**, all but the fixed links are removed (same as Only Fixed with 0 links).*

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages to remove the cut, fill and final cut/fill links, and some fixed links.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF

Modifier Panels.

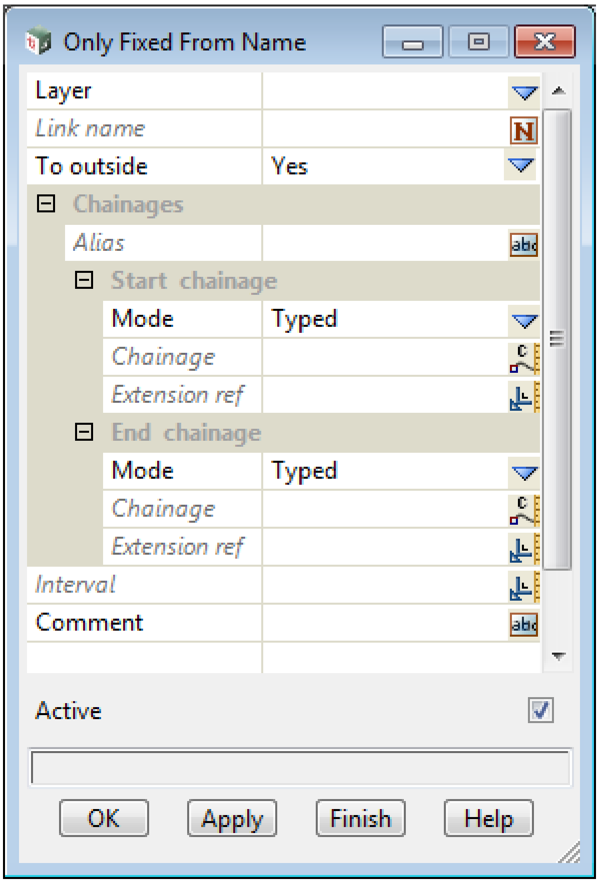
Go to the next section 21.2.2.2.2.5 Fixed Link - Remove All But Fixed & Remove Fixed Links from a Named Link or return to 21.2.2.2.2 Fixed Link - Remove.

21.2.2.2.2.5 Fixed Link - Remove All But Fixed & Remove Fixed Links from a Named Link

The **Leave fixed/remove from name** option removes any of the variable cut and fill, final cut/fill links and decisions that have been defined up to this point in the modifiers grid.

It also removes all the links from a named link to the last fixed link, **or** from a named link to the beginning of the fixed links.

Selecting **Leave fixed/remove from name** brings up the **Only Fixed from Name** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box		available Layers

Layer to remove the links from. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Link name	name box		names.4d list

name of the fixed link in the layer to start removing links from/to.

To outside	choice box		yes, no

*if **yes**, all the links from* **Link name** *out are removed.* **Link name** *is not removed.*
*If **no**, all the links from the beginning of the Fixed zone until* **Link name** *are removed.* **Link name** *is not removed.*

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages to remove the cut, fill and final cut/fill links, and some fixed links.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

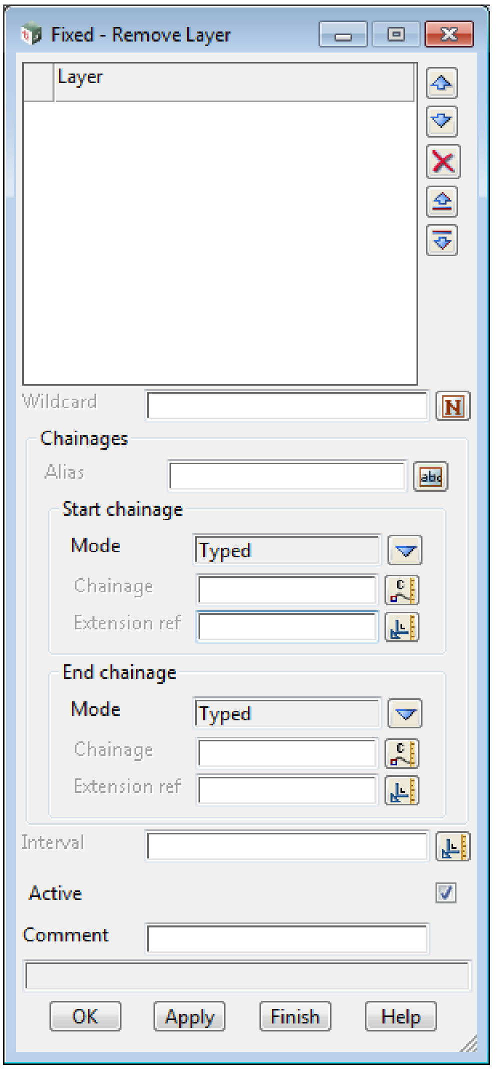
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Go to the next section 21.2.2.2.2.6 Fixed Link - Remove Layer or return to 21.2.2.2.2 Fixed Link - Remove.

21.2.2.2.2.6 Fixed Link - Remove Layer

The **Remove layers** option removes any links in the Layers that have been defined up to this point in the modifiers grid.

Selecting **Remove layers** brings up the **Only Fixed from Name** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Layer Grid

Layer	layer box		available Layers

link names that match **Wildcard** *are removed from the layers listed in the grid. For information on Layers, see* 21.1.1.2 MTF Links and Layers.

Wild card	text box		

text including wild cards (*) and wild characters () that the link names are matched against.

Link name	name box		names.4d list

name of the fixed link in the layer to start removing links from/to.

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages to remove the links from layers.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

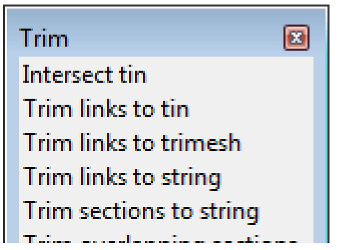
Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Return to 21.2.2.2.2 Fixed Link - Remove.

21.2.2.2.3 Fixed - Trim

The **Trim** walk-right brings up the **Trim** menu with options to perform various trimming operations.



See

Intersect tin	21.2.2.2.3.1 Intersect Tin
Trim links to tin	21.2.2.2.3.2 Trim Links to Tin
Trim sections to string	21.2.2.2.3.3 Trim Sections to String
Trim links to string	21.2.2.2.3.4 Trim Links to String
Trim overlapping sections	21.2.2.2.3.5 Trim Overlapping Sections

21.2.2.2.3.1 Intersect Tin

This option finds the intersection points of the link **Link** within the given vertical tolerance **Intersect z tol** of the stripped **Tin** and creates extra section at nominated chainage distances from the intersection chainages.

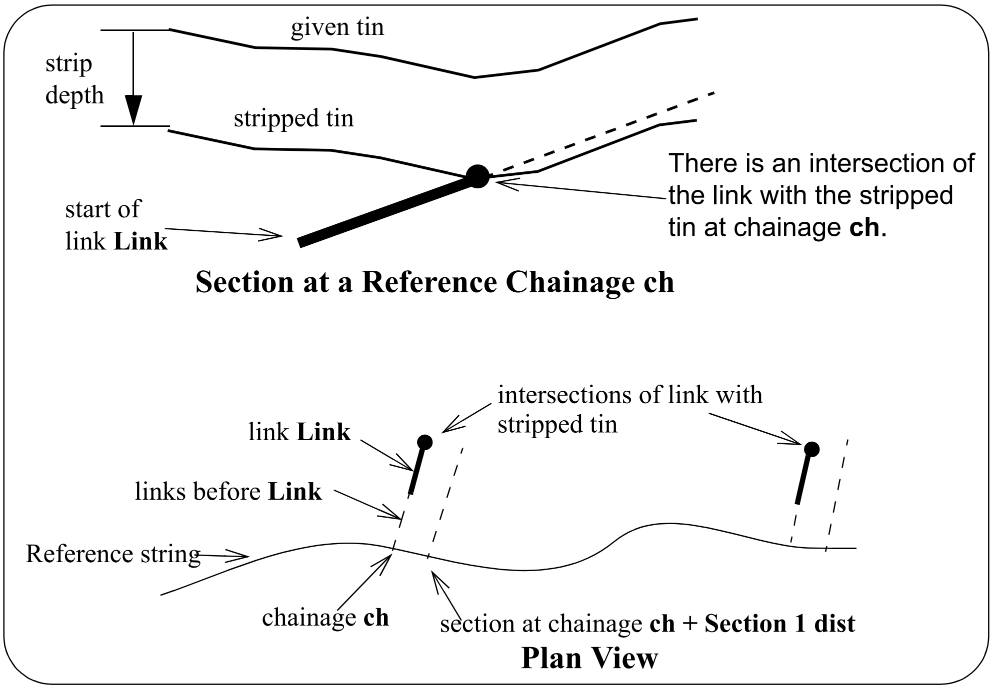
Extra sections are then created at the reference chainages

Reference chainage of intersection of link with (tin - Strip) + **Section 1 dist**

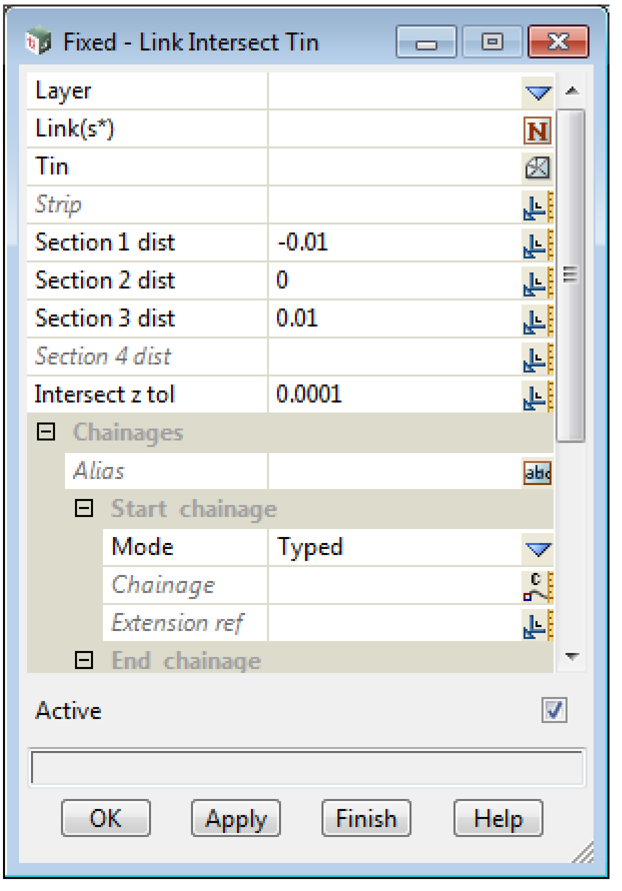
Reference chainage of intersection of link with (tin - Strip) + **Section 2 dist**

Reference chainage of intersection of link with (tin - Strip) + **Section 3 dist**

Reference chainage of intersection of link with (tin - Strip) + **Section 4 dist**



Selecting **Intersect tin** brings up the **Fixed - Link Intersect Tin** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box		available Layers

Layer of the link to cut with the stripped tin. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Link	name box		select name menu

name of the link to cut with the stripped tin. See 21.2.2.1.2 Link or Link(s).

Tin	tin box		available tins

name of the tin to use as the stepped tin to find the intersections with the link **Link**.

Strip	real box	0	

the depth to vertically drop the tin before calculating intersections. This is then referred to as the stripped tin.

Section 1, 2, 3, 4 dist	real box		

*if **not blank**, this value is added to the reference chainages of the intersection of the link with the stripped tin, and new sections are created at this new chainage. The value can be positive, negative and zero.*

*If **blank**, no section is created for this* **Section i dist.**

Notes

1. There may be more than one reference string chainage where the link intersects the stripped tin.

2. A section is only produced at the intersections of the link with the stripped tin if one of **Section i dist** *is zero.*

Intersect z tol real box 0.0001

the intersection of the link with the stripped tin can occur within this vertical distance of the striped tin.

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages for trimming the given fixed link against the tin.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

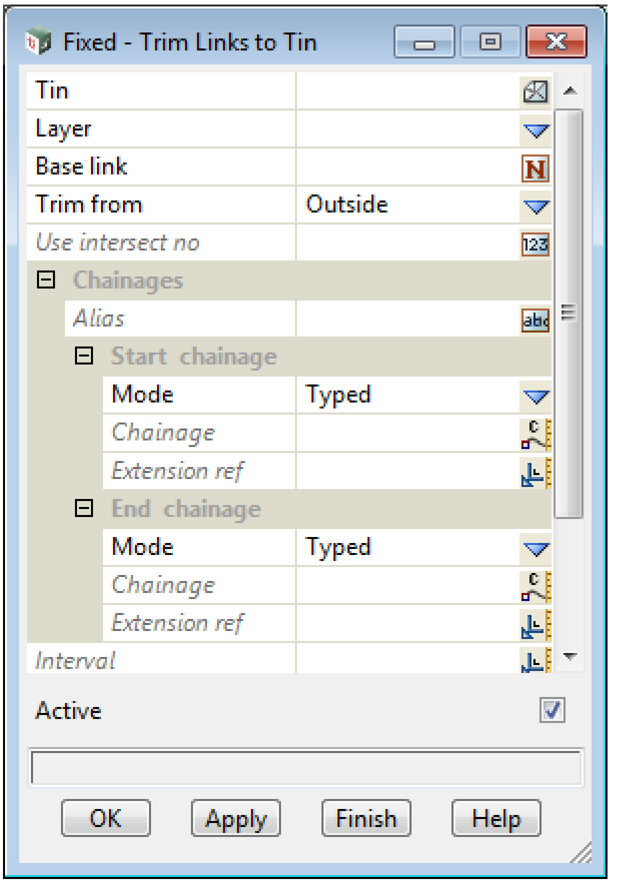
Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Go to the next section 21.2.2.2.3.2 Trim Links to Tin or return to 21.2.2.2.3 Fixed - Trim.

21.2.2.2.3.2 Trim Links to Tin

Selecting **Trim links to tin** brings up the **Fixed - Trim Links to Tin** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box		available Layers

Layer of strings to trim. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Link	name box		select name menu

name of the link to trim against the tin. See 21.2.2.1.2 Link or Link(s).

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages for trimming the given fixed link against the tin.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

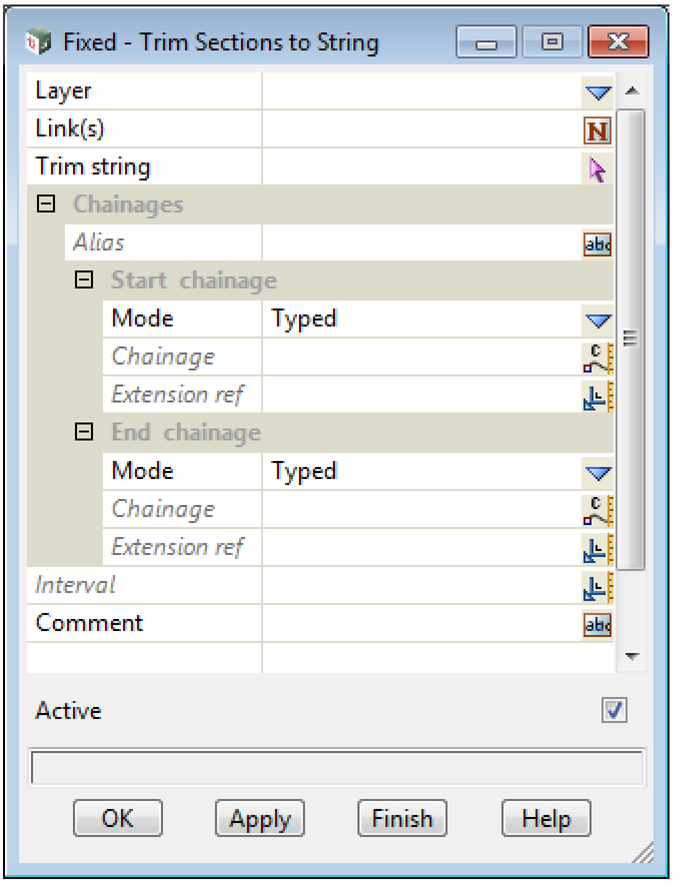
Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Go to the next section 21.2.2.2.3.3 Trim Sections to String or return to 21.2.2.2.3 Fixed - Trim.

21.2.2.2.3.3 Trim Sections to String

Selecting **Trim sections to string** brings up the **Fixed - Trim Sections to String** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box		available Layers

Layer of strings to trim. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Link(s)	name box		select name menu

name of the link to trim against the tin. See 21.2.2.1.2 Link or Link(s).

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages for trimming the given fixed link against the tin.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

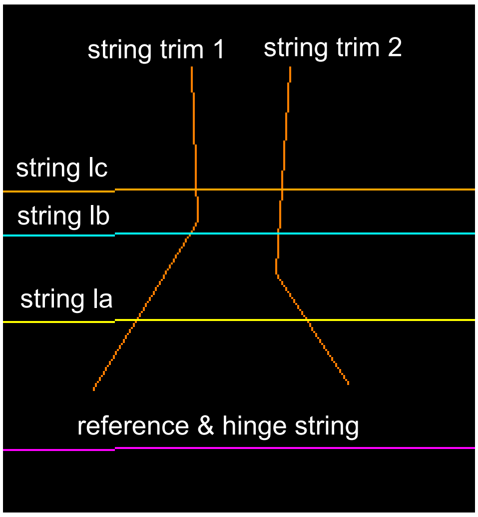
Comment, **Extra start**, **Extra End, Active, OK, Apply**

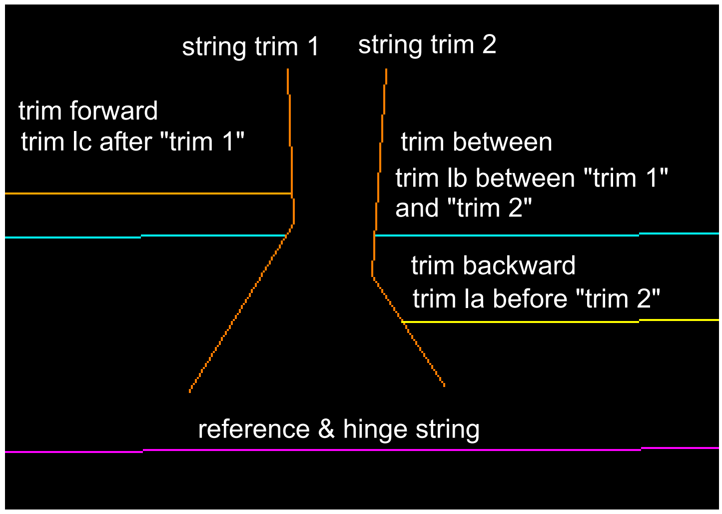
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Go to the next section 21.2.2.2.3.4 Trim Links to String or return to 21.2.2.2.3 Fixed - Trim.

21.2.2.2.3.4 Trim Links to String

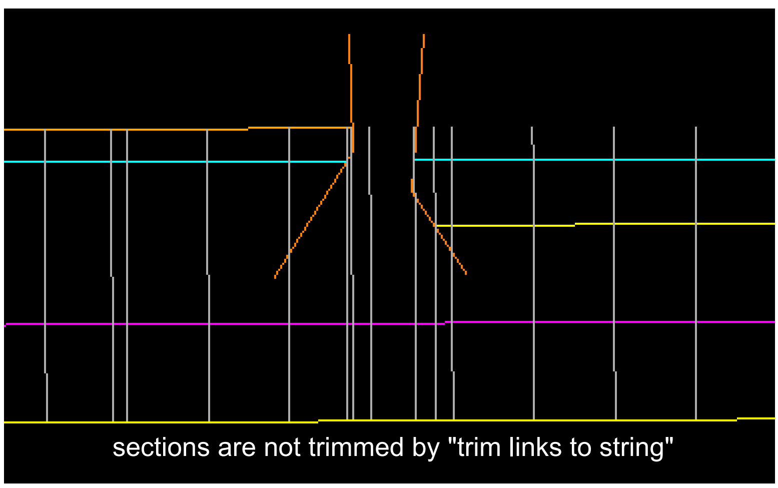
This option trims links before a selected string, or after a selected string, or between two selected strings.



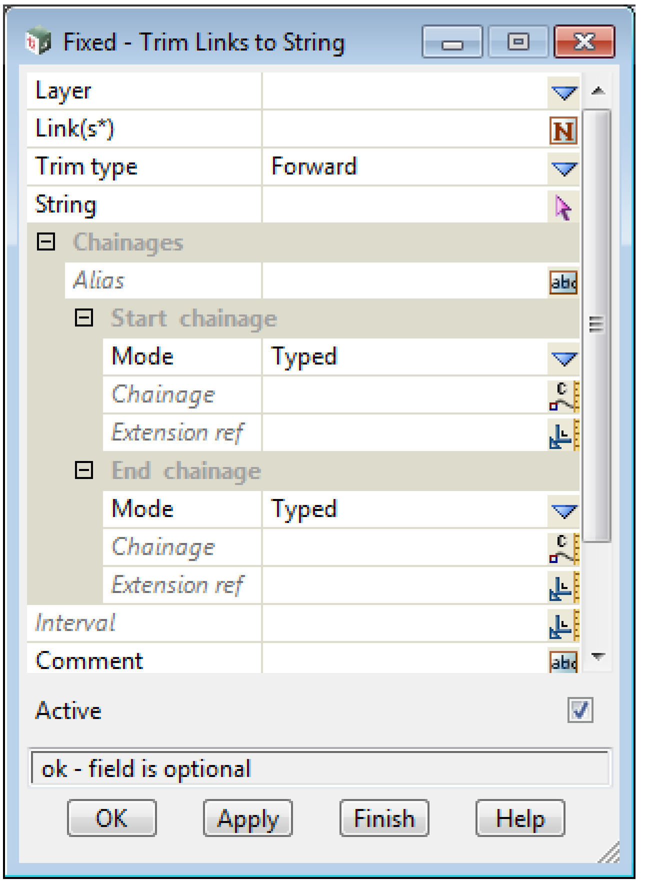


Important Note:

This option only trims the strings that are created from the links. It does NOT trim the sections. So it is a process that is performed **after** the sections and strings are first generated and it is only performed on the strings.



Selecting **Trim links to string** brings up the **Fixed - Trim Links to String** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box		available Layers

Layer of link to trim. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Link(s)	name box		select name menu

name of the links to trim against a string. See 21.2.2.1.2 Link or Link(s).

Trim type	choice box		forward, between, backward

type of the trim for the links

*if **forward**, there is a* **String** *select field and a string is user selected. All the links are trimmed so that they stop after the selected string.*

*If **between**, there is a* **Start string** *and* **End string** *select field and both strings are user selected. All the links are trimmed so that they stop between* **Start string** *and* **End string**.

*If **backward**, there is a* **String** *select field and a string is user selected. All the links are trimmed so that they only start at the selected string.*

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages for trimming the given fixed link against the tin.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Go to the next section 21.2.2.2.3.5 Trim Overlapping Sections or return to 21.2.2.2.3 Fixed - Trim.

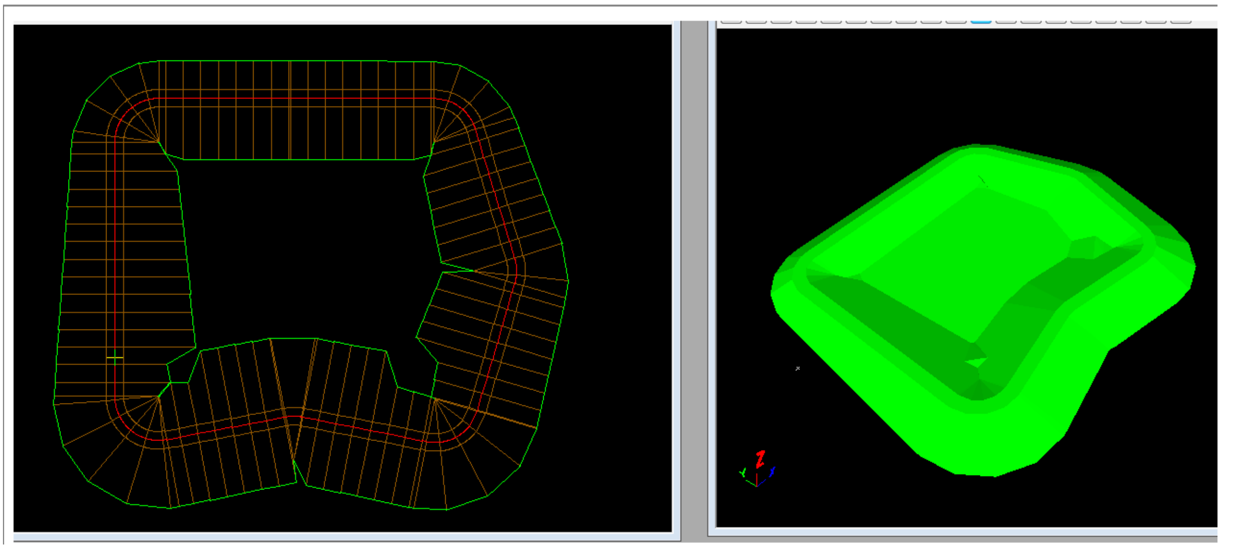
21.2.2.2.3.5 Trim Overlapping Sections

The **Trim Overlapping Sections** option takes a Layer and modifies the internally calculated sections so that the sections do not overlay each other.

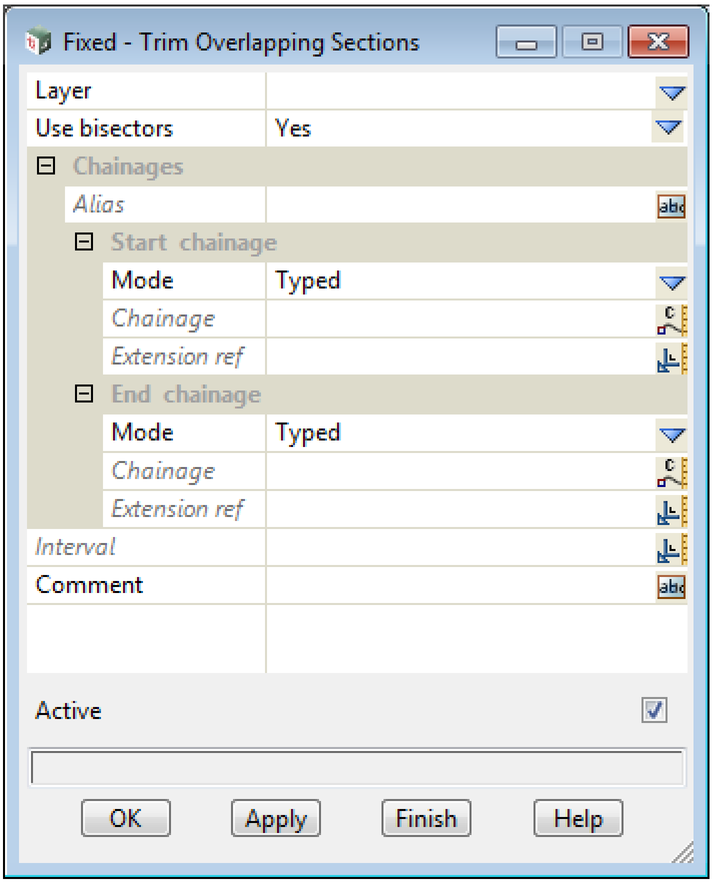
Any strings that are generated for that Layer will come from those modified internal sections and so will also not cross over themselves.

So any triangulation using those strings should not need any further cleanups to produce a fairly accurate tin. Certainly the tin should be accurate enough for any earthworks.

For the Design Layer, both sections and strings can be created and using the **Trim Overlapping Sections** option will produce results such as the example below.



Selecting **Trim overlapping sections** brings up the **Fixed - Trim Overlapping Sections** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box		available Layers

name of the Layer to modify the internally calculated sections so that the sections do not overlay each other. Any strings that are generated for that Layer after this process will come from the modified internal sections and so will also not cross over themselves.

For information on Layers, see 21.1.1.2 MTF Links and Layers.

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages for trimming the given fixed link against the tin.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Return to 21.2.2.2.3 Fixed - Trim.

21.2.2.2.4 Fixed - Modify Link

The **Fixed =>Link** option can modify the width, height, xfall or slope **of any Fixed link.**

At a given chainage, it doesn't matter which of the two values from width, height or xfall were used to defined a Fixed link, all of the values width, height, xfall and slope can be then calculated. See 21.1.2.1 Calculating Width, Height, Xfall or Slope from Original Definitions

So for a Fixed Link, the **Link** option allows to user to chose which one of **width**, **height**, **xfall** or **slope** to **modify** and **how to modify it**, and which of **width**, **height**, **xfall** or **slope** is **held** to be the **current value** for the **Fixed link.**

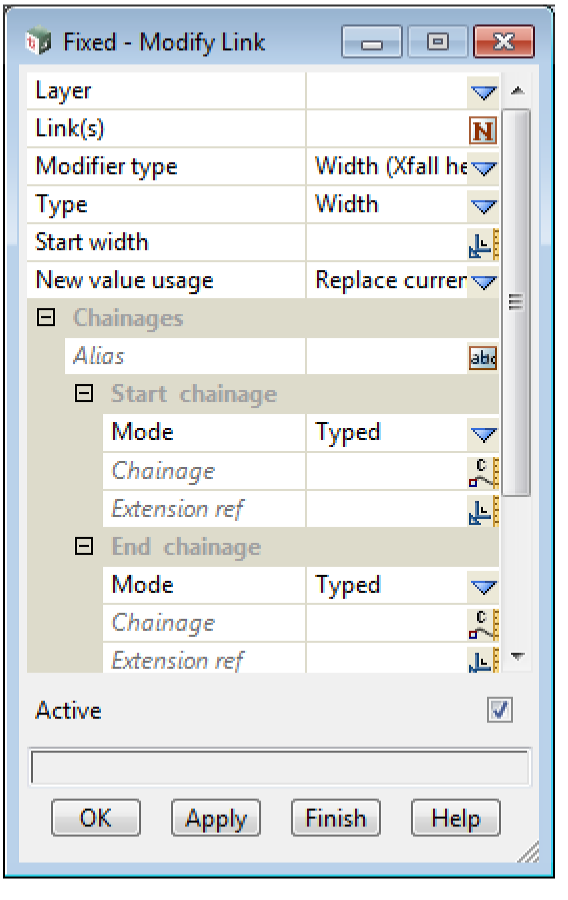
Notes

1. The combination Modify xfall and Hold slope makes no sense because xfall and slope measure exactly the same thing.
2. There is an important difference between Modify Xfall and Modify Slope.

   Modify Xfall linearly interpolates the xfall between the Start Chainage and the End Chainage where xfall is the percentage of the proportion of metres vertically to metres horizontally.

   Modify Slope linearly interpolates the slope between the Start Chainage and the End Chainage where slope is the ratio between one unit vertically to a number of units horizontally (1 in).

Selecting **Link** brings up the **Fixed - Modify Link** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box	Design	available Layers

name of the layer that the link to be modified is in. For information on Layers, see 21.1.1.2 MTF

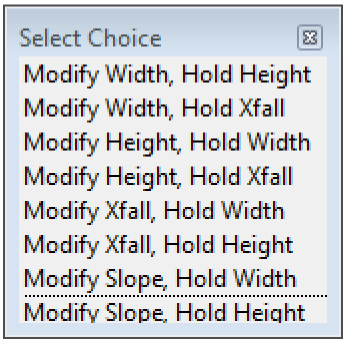
Links and Layers.

Link(s) name box names.4d pop-up

the name of the links in **Layer** *to modify the width/height/xfall/slope using the method given in* **Modifier type,** *between the chainages given by* **Start chainage** *and* **End chainage.** *See* 21.2.2.1.2 Link or Link(s).

When describing how the Link option works, this link is referred to as the ***Current link*** *and the values of width, height, xfall and slope for the Current Link as the* ***Current*** *or* ***Existing Values.***

Modifier type choice box



modify Width and take Height from current link
modify Width and take Xfall from current link
modify Height and take Width from current link
modify Height and take Xfall from current link
modify Xfall and take With from current link
modify Xfall and take Height from current link
modify Slope and take With from current link
modify Slope and take Height from current link

A fixed link can be modified in any of the **Modify type** *ways regardless of how the link was originally defined in terms of width, height, xfall or slope. See* 21.1.2.1 Calculating Width, Height, Xfall or Slope from Original Definitions.

The ***Modify*** *width/height/xfall/slope is the part of the current link that is modified from its current value by the method given by* **Modifier type**.

How *the width/height/xfall/slope is modified depends on the choice made in the* **Type** *field described below.*

The ***Hold*** *width/height/xfall is the part of the current link that is held at its current value for the link.*

New value usage choice box Add to the current value of the link, Replace the current value of the link

if ***Add to the current value of the link****, the value of the parameter width/height/xfall/slope at each chainage is* ***added to*** *the current value for the parameter. For compactness, in a grid, this will be displayed in an* **Absolute** *column with* ***Absolute not ticked*** *or* ***no.***

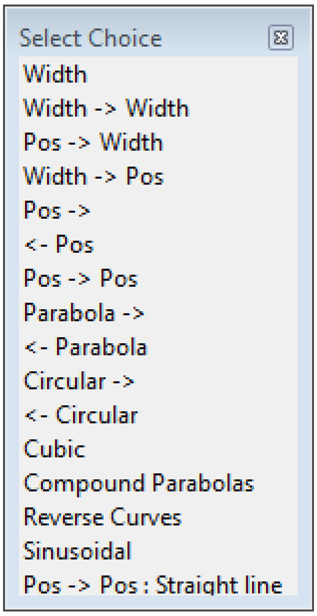
if ***Replace the current value of the link****, the value of the parameter width/height/xfall/slope at each chainage* ***replaces*** *the current value for the parameter. For compactness, In a grid, this will be shown in an* **Absolute** *column with* ***Absolute ticked*** *or* ***yes****.*

See 21.1.2.3 New Value Usage.

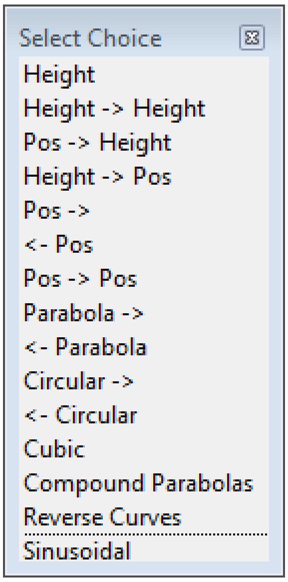
Type choice box

the choices for **Type** *depends on the choice selected for* **Modifier type**.

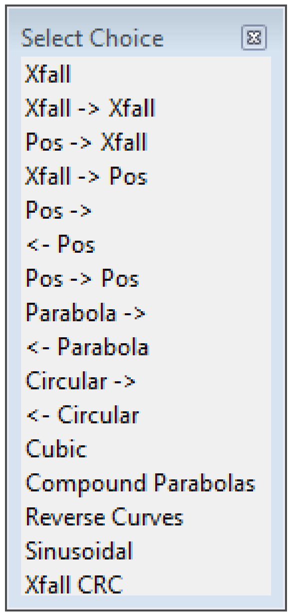
When **Modify type** is **Modify Width** the choices for **Type** are



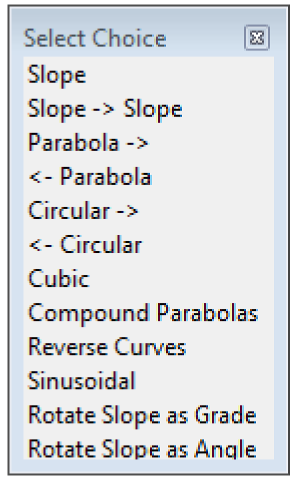
When **Modify type** is **Modify Height** the choices for **Type** are



When **Modify type** is **Modify Xfall** the choices for **Type** are



When **Modify type** is **Modify Slope** the choices for **Type** are



For a description of the choices of **Type** *when* **Modifier type** *is* ***Modify Width***, *see* 21.2.2.2.4.1 Type for Modifier Types "Modify Width, Hold Height/Xfall"

For a description of the choices of **Type** *when* **Modifier type** *is* ***Modify Height***, *see* 21.2.2.2.4.2 Type for Modifier Types "Modify Height, Hold Width/Xfall"

For a description of the choices of **Type** *when* **Modifier type** *is* ***Modify Xfall***, *see* 21.2.2.2.4.3 Type for Modifier Types "Modify Xfall, Hold Width/Height"

For a description of the choices of **Type** *when* **Modifier type** *is* ***Modify Slope***, *see* 21.2.2.2.4.4 Type for Modifier Types "Modify Slope, Hold Width/Height"

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages for modifying the new link

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

21.2.2.2.4.1 Type for Modifier Types "Modify Width, Hold Height/Xfall"

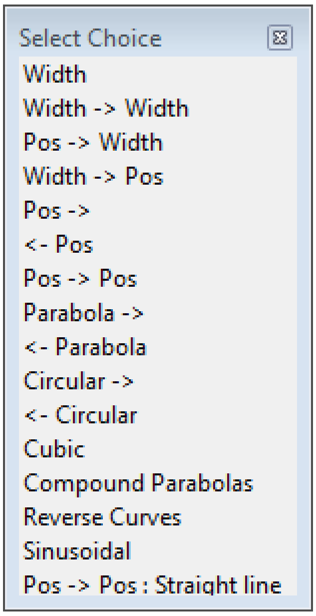
The choices for **Type** when **Modify type** is:

(a) **Modify Width, Hold Height**

or

(b) **Modify Width, Hold Xfall**

are



use a given width for the chainage range
interpolate between a given start width and a given end width
interpolate between the current start width and a given end width
interpolate between a given start width and the current end width
use the current start width for the chainage range
use the current start width for the chainage range
interpolate between the current start width and the current end width
use a parabola from the current start width
use a parabola that goes through the current end width
use an arc from the current start width
use an arc that goes through the current end width
use a cubic between the current start and end widths
use a compound parabola between the current start and end widths
use reverse curves between the current start and end widths
use a sinusoidal curve between the current start and end widths
use a straight line between the current start and end widths

For the description of each **Type** see

21.2.2.2.4.1.1 Width
21.2.2.2.4.1.2 Width -> Width
21.2.2.2.4.1.3 Pos -> Width
21.2.2.2.4.1.4 Width -> Pos
21.2.2.2.4.1.5 Pos ->
21.2.2.2.4.1.6 <- Pos
21.2.2.2.4.1.7 Pos-> Pos:
21.2.2.2.4.1.8 Parabola ->
21.2.2.2.4.1.9 <- Parabola
21.2.2.2.4.1.10 Circular->
21.2.2.2.4.1.11 <-Circular
21.2.2.2.4.1.12 Cubic
21.2.2.2.4.1.13 Pos-> Pos: Straight line

21.2.2.2.4.1.1 Width

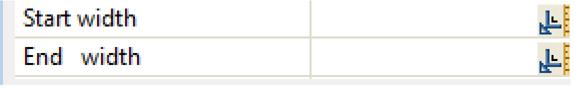
Use a Given Width from the Start Chainage to the End Chainage

At each chainage ch between the **Start chainage** and **End chainage**, the width is that given in the **Width** panel field. That is, the width has the constant value of **Width**.

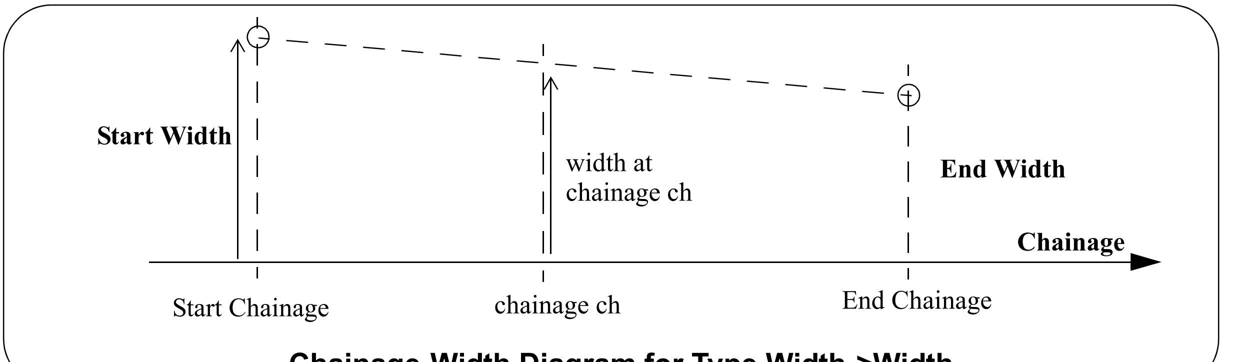
Go to the next section 21.2.2.2.4.1.2 Width -> Width or return to 21.2.2.2.4.1 Type for Modifier Types "Modify Width, Hold Height/Xfall" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.1.2 Width -> Width

Interpolate Between a Given Start Width and a Given End Width



At each chainage ch between the **Start chainage** and **End chainage**, the width is linearly interpolated by chainage between the given **Start width** and the **End width**.



Chainage-Width Diagram for Type Width->Width

Go to the next section 21.2.2.2.4.1.3 Pos -> Width or return to 21.2.2.2.4.1 Type for Modifier Types "Modify Width, Hold Height/Xfall" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.1.3 Pos -> Width

Interpolate Between the Existing Start Width & a Given End width

At each chainage ch between the **Start chainage** and **End chainage**, the width is linearly interpolated by chainage between the width at the **Start chainage** and the given width **Width** at the **End chainage**.

This allows you to use the width at the **End chainage** without knowing what the actual value of the width is.

Go to the next section 21.2.2.2.4.1.4 Width -> Pos or return to 21.2.2.2.4.1 Type for Modifier Types "Modify Width, Hold Height/Xfall" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.1.4 Width -> Pos

Interpolate Between a Given Start Width & the Existing End Width

At each chainage ch between the **Start chainage** and **End chainage**, the width is linearly interpolated by chainage between the given width **Width** at the **Start chainage** and the width at the **End chainage**.

This allows you to use the width at the **Start chainage** without knowing what the actual value of the width is.

Go to the next section 21.2.2.2.4.1.5 Pos -> or return to 21.2.2.2.4.1 Type for Modifier Types "Modify Width, Hold Height/Xfall" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.1.5 Pos ->

Use the Current Start Width for the Entire Chainage Range

At each chainage ch between the **Start chainage** and **End chainage**, the width is that at the **Start chainage**. That is, the width has the constant value of what it is a the **Start chainage**.

This allows you to use the width at the **Start chainage** without knowing what the actual value of the width is.

Go to the next section 21.2.2.2.4.1.6 <- Pos or return to 21.2.2.2.4.1 Type for Modifier Types "Modify Width, Hold Height/Xfall" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.1.6 <- Pos

Use the Current End Width for the Entire Chainage Range

At each chainage ch between the **Start chainage** and **End chainage**, the width is that at the **End chainage**. That is, the width has the constant value of what it is a the **End chainage**.

This allows you to use the width at the **End chainage** without knowing what the actual value of the width is.

Go to the next section 21.2.2.2.4.1.7 Pos-> Pos: or return to 21.2.2.2.4.1 Type for Modifier Types "Modify Width, Hold Height/Xfall" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.1.7 Pos-> Pos:

Interpolate Between the Existing Start Width and the Existing End Widths

At each chainage ch between the **Start chainage** and **End chainage**, the width is linearly interpolated by chainage between the width at the **Start chainage** and the width at the **End chainage**.

This allows you to use the widths at the **Start chainage** and the **End chainage** without knowing what the actual value of the widths are.

Go to the next section 21.2.2.2.4.1.8 Parabola -> or return to 21.2.2.2.4.1 Type for Modifier Types "Modify Width, Hold Height/Xfall" or 21.2.2.2.4 Fixed - Modify Link

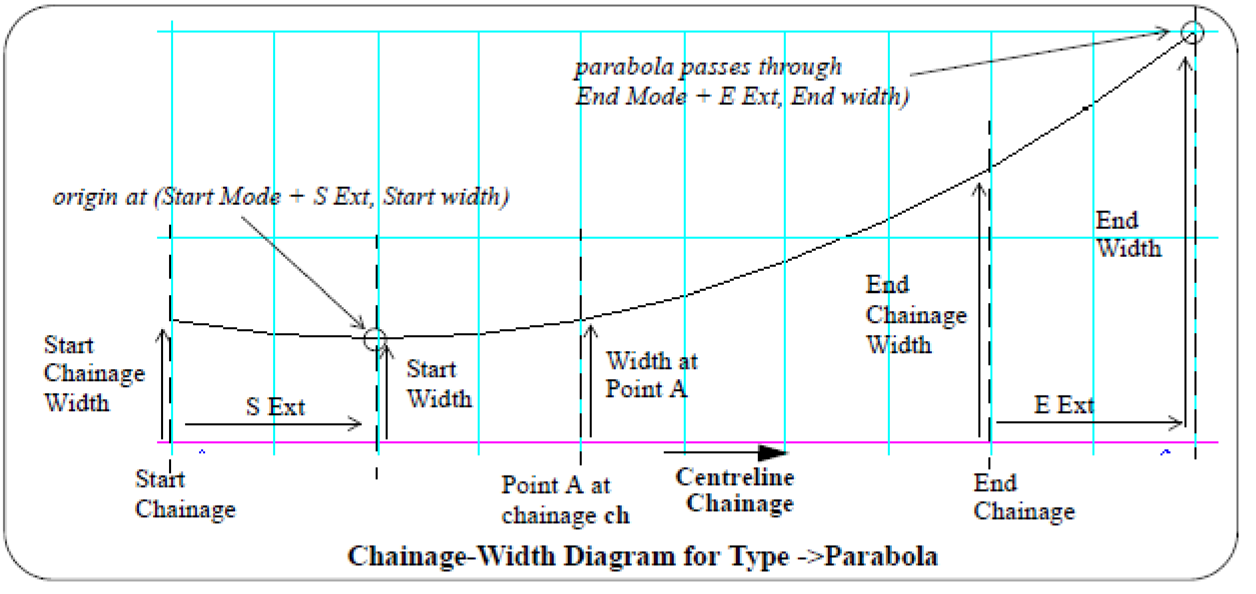
21.2.2.2.4.1.8 Parabola ->

Calculation of Width for Type Parabola ->

This option is still under development

*In the chainage-width diagram, the **Start width** is the width at chainage **Start mode plus S Ext**, and **End width** is the width at chainage **End mode plus E Ext**.*

*Between the start position (**Start mode + S Ext**, **Start width)**, and the end position (**End mode + E Ext**, **End width**), the width varies as a **parabola** a*X*X with its origin at (**Start mode + S Ext**, **Start width)** and going through (**End mode + E Ext**, **End width**).*



Go to the next section 21.2.2.2.4.1.9 <- Parabola or return to 21.2.2.2.4.1 Type for Modifier Types "Modify Width, Hold Height/Xfall" or 21.2.2.2.4 Fixed - Modify Link

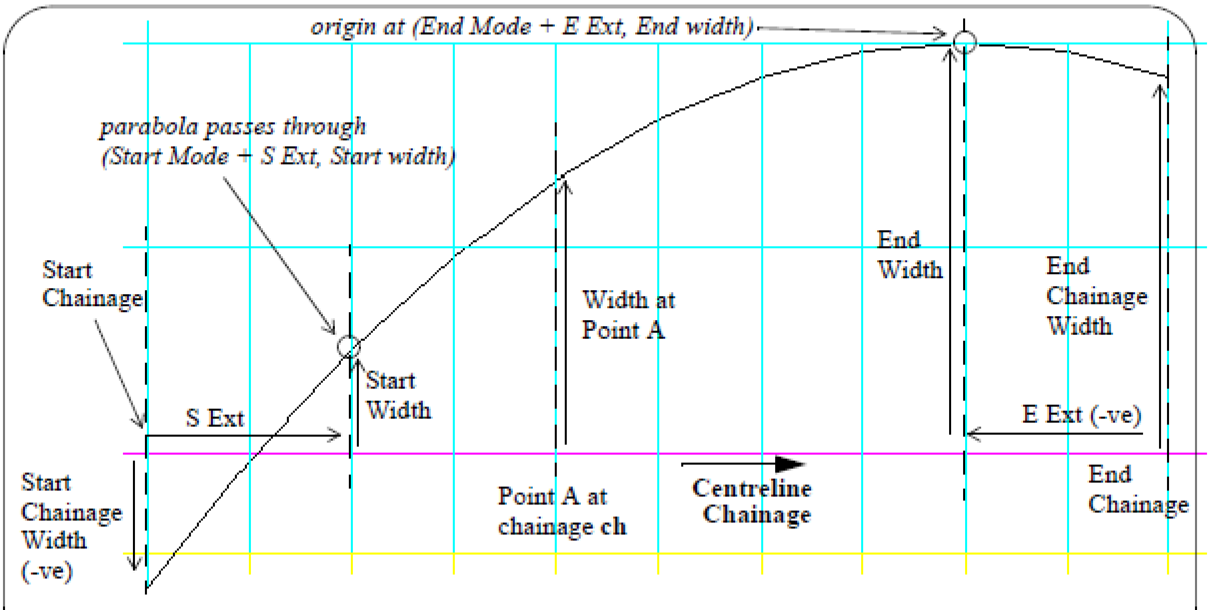
21.2.2.2.4.1.9 <- Parabola

Calculation of Width for Type <- Parabola

This option is still under development

*In the chainage-width diagram, the **Start width** is the width at chainage **Start mode plus S Ext**, and **End width** is the width at chainage **End mode plus E Ext**.*

*Between the start position (**Start mode + S Ext**, **Start width)**, and the end position (**End mode + E Ext**, **End width**), the width varies as a **parabola** a*X*X with its origin at (**End mode + E Ext**, **End width)** and going through (**Start mode + S Ext**, **Start width**).*



Chainage-Width Diagram for Type <-Parabola

Go to the next section [21.2.2.2.4.1.10 Circular->] or return to [21.2.2.2.4.1 Type for Modifier Types "Modify Width, Hold Height/Xfall"] or [21.2.2.2.4 Fixed - Modify Link]

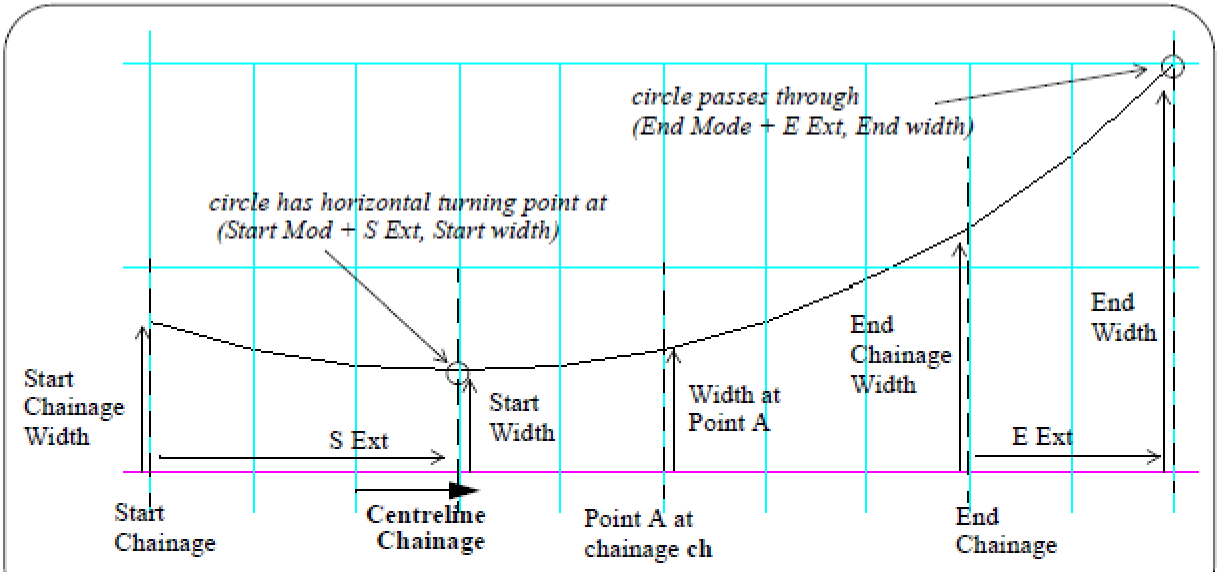
21.2.2.2.4.1.10 Circular->

Calculation of Width for Type Circular->

This option is still under development

*In the chainage-width diagram, the **Start width** is the width at chainage **Start mode plus S Ext**, and **End width** is the width at chainage **End mode plus E Ext**.*

*Between the start position (**Start mode + S Ext**, **Start width)**, and the end position (**End mode + E Ext**, **End width**), the width varies as a **circle** with its horizontal turning point at (**Start mode + S Ext**, **Start width)** and going through (**End mode + E Ext**, **End width**).*



Chainage-Width Diagram for Type Circular->

Go to the next section 21.2.2.2.4.1.11 <-Circular or return to 21.2.2.2.4.1 Type for Modifier Types "Modify Width, Hold Height/Xfall" or 21.2.2.2.4 Fixed - Modify Link

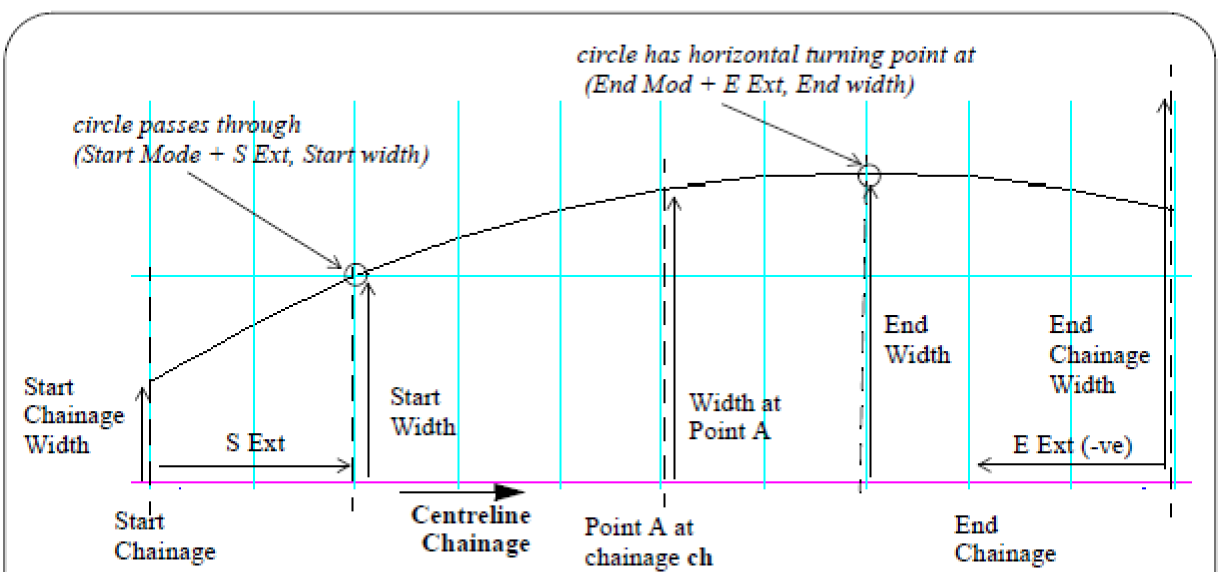
21.2.2.2.4.1.11 <-Circular

Calculation of Width for Type <-Circular

This option is still under development

*In the chainage-width diagram, the **Start width** is the width at chainage **Start mode plus S Ext**, and **End width** is the width at chainage **End mode plus E Ext**.*

*Between the start position (**Start mode + S Ext**, **Start width)**, and the end position (**End mode + E Ext**, **End width**), the width varies as a **circle** with its horizontal turning point at (**End mode + E Ext**, **End width**) and going through (**Start mode + S Ext**, **Start width)**.*



Chainage-Width Diagram for Type <-Circular

Go to the next section 21.2.2.2.4.1.12 Cubic or return to 21.2.2.2.4.1 Type for Modifier Types "Modify Width, Hold Height/Xfall" or 21.2.2.2.4 Fixed - Modify Link

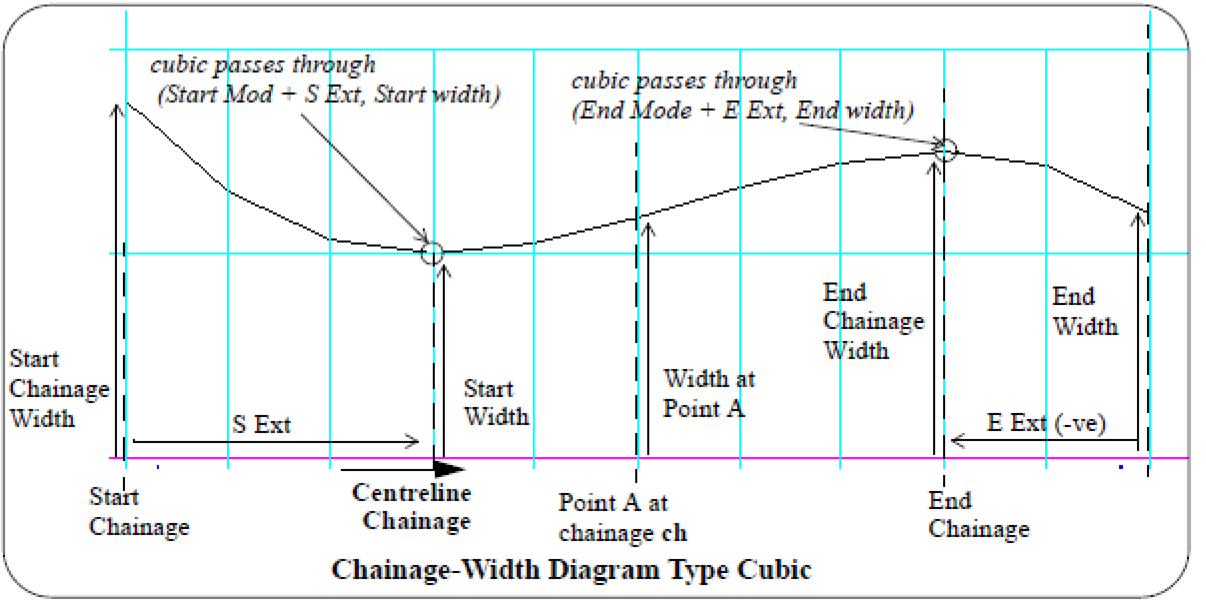
21.2.2.2.4.1.12 Cubic

Calculation of Width for Type Cubic

This option is still under development

*In the chainage-width diagram, the **Start width** is the width at chainage **Start mode plus S Ext**, and **End width** is the width at chainage **End mode plus E Ext**.*

*Between the start position (**Start mode + S Ext**, **Start width)**, and the end position (**End mode + E Ext**, **End width**), the width varies as a **cubic** going through (**Start mode + S Ext**, **Start width)** and (**End mode + E Ext**, **End width**).*



Chainage-Width Diagram Type Cubic

Go to the next section 21.2.2.2.4.1.13 Pos-> Pos: Straight line or return to 21.2.2.2.4.1 Type for Modifier Types "Modify Width, Hold Height/Xfall" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.1.13 Pos-> Pos: Straight line

Use a Straight Line Between the Existing Start & End Widths

At each chainage ch between the **Start chainage** and **End chainage**, the width is calculated so that it is on a straight line in plan between the width at the **Start chainage** and the width at the **End chainage**.

This allows you to join the widths at the **Start chainage** and the **End chainage** by a plan straight line without knowing what the actual value of the widths are.

Return to 21.2.2.2.4.1 Type for Modifier Types "Modify Width, Hold Height/Xfall" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.2 Type for Modifier Types "Modify Height, Hold Width/Xfall"

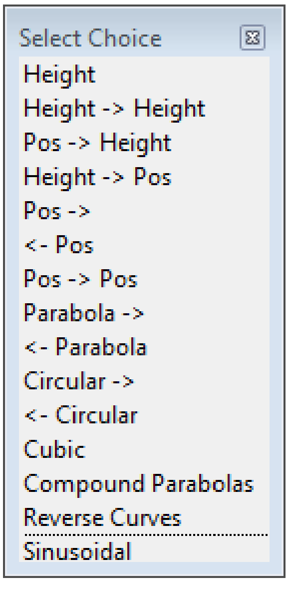
The choices for **Type** when **Modify type** is:

(a) **Modify Height, Hold Width**

or

(b) **Modify Height, Hold Xfall**

are



use a given height for chainage range
interpolate between a given start height and a given end height
interpolate between the current start height and a given end height
interpolate between a given start height and the current end height
keep the current start height for the chainage range
use the current end height for the chainage range
interpolate between the current start height and the current end height
use a parabola from the current start height
use a parabola that goes through the current end height
use an arc from the current start height
use an arc that goes through the current end height
use a cubic that goes through the current start and end heights
use a compound parabola between the current start and end heights
use reverse curves between the current start and end heights
use a sinusoidal curve between the current start and end heights

For the description of each **Type** see

21.2.2.2.4.2.1 Height
21.2.2.2.4.2.2 Height -> Height
21.2.2.2.4.2.3 Pos -> Height
21.2.2.2.4.2.4 Height -> Pos
21.2.2.2.4.2.5 Pos ->
21.2.2.2.4.2.6 <- Pos
21.2.2.2.4.2.7 Pos-> Pos:
21.2.2.2.4.2.8 Parabola ->
21.2.2.2.4.2.9 <- Parabola
21.2.2.2.4.2.10 Circular->
21.2.2.2.4.2.11 <-Circular
21.2.2.2.4.2.12 Cubic

21.2.2.2.4.2.1 Height

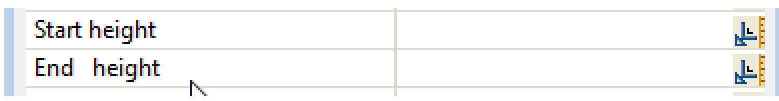
Use a Given Height from the Start Chainage to the End Chainage

At each chainage ch between the **Start chainage** and **End chainage**, the width is that given in the **Height** panel field. That is, the height has the constant value of **Height**.

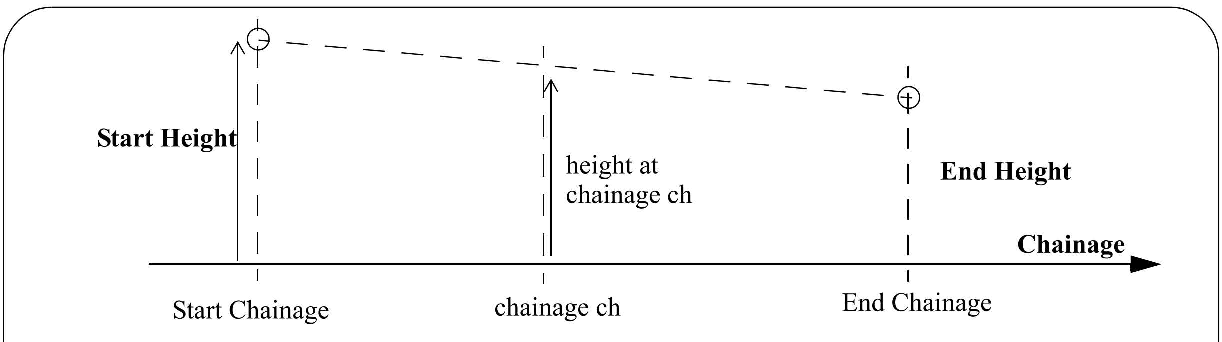
Go to the next section 21.2.2.2.4.2.2 Height -> Height or return to 21.2.2.2.4.2 Type for Modifier Types "Modify Height, Hold Width/Xfall" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.2.2 Height -> Height

Interpolate Between a Given Start Height and a Given End Height



At each chainage ch between the **Start chainage** and **End chainage**, the height is linearly interpolated by chainage between the given **Start height** and the **End height**.



Chainage-Height Diagram for Type Height->Height

Go to the next section 21.2.2.2.4.2.3 Pos -> Height or return to 21.2.2.2.4.2 Type for Modifier Types "Modify Height, Hold Width/Xfall" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.2.3 Pos -> Height

Interpolate Between the Existing Start Height & a Given End Height

At each chainage ch between the **Start chainage** and **End chainage**, the height is linearly interpolated by chainage between the height at the **Start chainage** and the given height **Height** at the **End chainage**.

This allows you to use the height at the **End chainage** without knowing what the actual value of the height is.

Go to the next section 21.2.2.2.4.2.4 Height -> Pos or return to 21.2.2.2.4.2 Type for Modifier Types "Modify Height, Hold Width/Xfall" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.2.4 Height -> Pos

Interpolate Between a Given Start Height & the Existing End Height

At each chainage ch between the **Start chainage** and **End chainage**, the height is linearly interpolated by chainage between the given height **Height** at the **Start chainage** and the height at the **End chainage**.

This allows you to use the height at the **Start chainage** without knowing what the actual value of the height is.

Go to the next section 21.2.2.2.4.2.5 Pos -> or return to 21.2.2.2.4.2 Type for Modifier Types "Modify Height, Hold Width/Xfall" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.2.5 Pos ->

Use the Current Start Height for the Entire Chainage Range

At each chainage ch between the **Start chainage** and **End chainage**, the height is that at the **Start chainage**. That is, the height has the constant value of what it is a the **Start chainage**.

This allows you to use the height at the **Start chainage** without knowing what the actual value of the height is.

Go to the next section 21.2.2.2.4.2.6 <- Pos or return to 21.2.2.2.4.2 Type for Modifier Types "Modify Height, Hold Width/Xfall" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.2.6 <- Pos

Use the Current End Height for the Entire Chainage Range

At each chainage ch between the **Start chainage** and **End chainage**, the height is that at the **End chainage**. That is, the height has the constant value of what it is a the **End chainage**.

This allows you to use the height at the **End chainage** without knowing what the actual value of the height is.

Go to the next section 21.2.2.2.4.2.7 Pos-> Pos: or return to 21.2.2.2.4.2 Type for Modifier Types "Modify Height, Hold Width/Xfall" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.2.7 Pos-> Pos:

Interpolate Between the Existing Start and End Heights

At each chainage ch between the **Start chainage** and **End chainage**, the height is linearly interpolated by chainage between the height at the **Start chainage** and the height at the **End chainage**.

This allows you to use the heights at the **Start chainage** and the **End chainage** without knowing what the actual value of the heights are.

Go to the next section 21.2.2.2.4.2.8 Parabola -> or return to 21.2.2.2.4.2 Type for Modifier Types "Modify Height, Hold Width/Xfall" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.2.8 Parabola ->

Calculation of Height for Type Parabola ->

The calculations are the same as the case for Width except Height is substituted for Width in all the formulae and diagrams. See 21.2.2.2.4.1.8 Parabola ->.

Go to the next section 21.2.2.2.4.2.9 <- Parabola or return to 21.2.2.2.4.2 Type for Modifier Types "Modify Height, Hold Width/Xfall" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.2.9 <- Parabola

Calculation of Height for Type <- Parabola

The calculations are the same as the case for Width except Height is substituted for Width in all the formulae and diagrams. See 21.2.2.2.4.1.9 <- Parabola.

Go to the next section 21.2.2.2.4.2.10 Circular-> or return to 21.2.2.2.4.2 Type for Modifier Types "Modify Height, Hold Width/Xfall" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.2.10 Circular->

Calculation of Height for Type Circular->

The calculations are the same as the case for Width except Height is substituted for Width in all the formulae and diagrams. See 21.2.2.2.4.1.10 Circular->.

Go to the next section 21.2.2.2.4.2.11 <-Circular or return to 21.2.2.2.4.2 Type for Modifier Types "Modify Height, Hold Width/Xfall" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.2.11 <-Circular

Calculation of Height for Type <-Circular

The calculations are the same as the case for Width except Height is substituted for Width in all the formulae and diagrams. See 21.2.2.2.4.1.11 <-Circular.

Go to the next section 21.2.2.2.4.2.12 Cubic or return to 21.2.2.2.4.2 Type for Modifier Types "Modify Height, Hold Width/Xfall" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.2.12 Cubic

Calculation of Height for Type Cubic

The calculations are the same as the case for Width except Height is substituted for Width in all the formulae and diagrams. See 21.2.2.2.4.1.12 Cubic.

Return to 21.2.2.2.4.2 Type for Modifier Types "Modify Height, Hold Width/Xfall" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.3 Type for Modifier Types "Modify Xfall, Hold Width/Height"

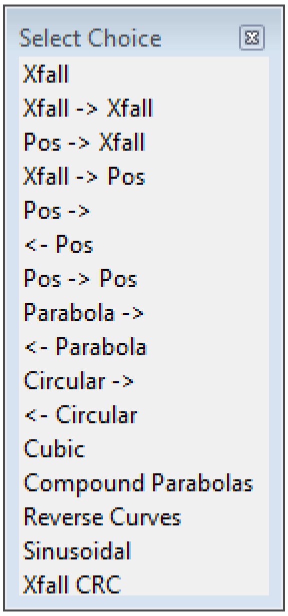
The choices for **Type** when **Modify type** is:

(a) **Modify Xfall, Hold Width**

or

(b) **Modify Xfall, Hold Height**

are



use a given xfall for chainage range
interpolate between a given start xfall and a given end xfall
interpolate between the current start xfall and a given end xfall
interpolate between a given start xfall and the current end xfall
keep the current start xfall for the chainage range
use the current end xfall for the chainage range
interpolate between the current start xfall and the current end xfall
use a parabola from the current start xfall
use a parabola that goes through the current end xfall
use an arc from the current start xfall
use an arc that goes through the current end xfall
use a cubic between the current start and end xfalls
use a compound parabola between the current start and end xfalls
use reverse curves between the current start and end xfalls
use a sinusoidal curve between the current start and end xfalls

For the description of each **Type** see

21.2.2.2.4.3.1 Xfall

21.2.2.2.4.3.2 Xfall -> Xfall

21.2.2.2.4.3.3 Pos -> Xfall

21.2.2.2.4.3.4 Xfall -> Pos

21.2.2.2.4.3.5 Pos ->

21.2.2.2.4.3.6 <- Pos

21.2.2.2.4.3.7 Pos-> Pos:

21.2.2.2.4.3.8 Parabola ->

21.2.2.2.4.3.9 <- Parabola

21.2.2.2.4.3.10 Circular->

21.2.2.2.4.3.11 <-Circular

21.2.2.2.4.3.12 Cubic

21.2.2.2.4.3.1 Xfall

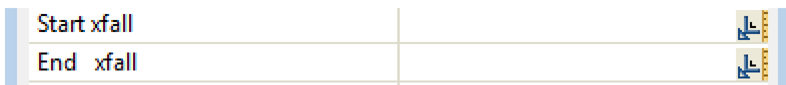
Keep a Given Xfall

At each chainage ch between the **Start chainage** and **End chainage**, the width is that given in the **Xfall** panel field. That is, the Xfall has the constant value of **Xfall**.

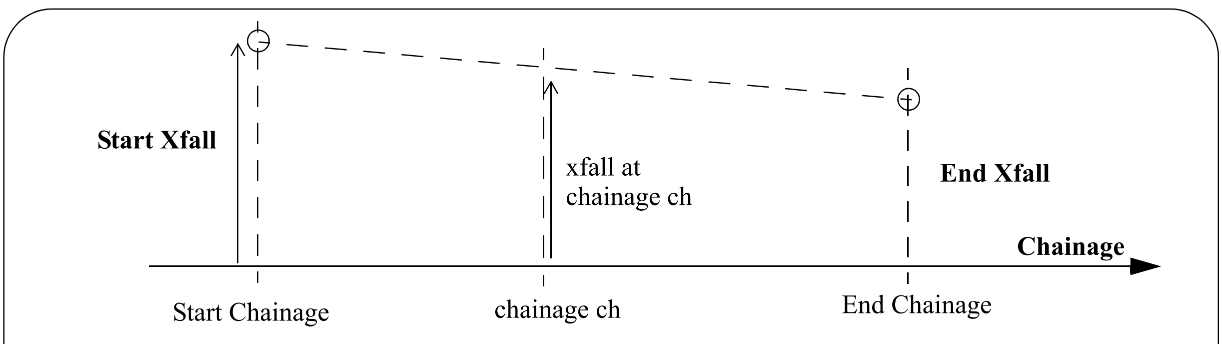
Go to the next section 21.2.2.2.4.3.2 Xfall -> Xfall or return to 21.2.2.2.4.3 Type for Modifier Types "Modify Xfall, Hold Width/Height" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.3.2 Xfall -> Xfall

Interpolate Between a Given Start and End Xfall



At each chainage ch between the **Start chainage** and **End chainage**, the Xfall is linearly interpolated by chainage between the given **Start Xfall** and the **End Xfall**.



Chainage-Height Diagram for Type Xfall->Xfall

Go to the next section 21.2.2.2.4.3.3 Pos -> Xfall or return to 21.2.2.2.4.3 Type for Modifier Types "Modify Xfall, Hold Width/Height" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.3.3 Pos -> Xfall

Interpolate Between Existing Start Xfall & Given End Xfall

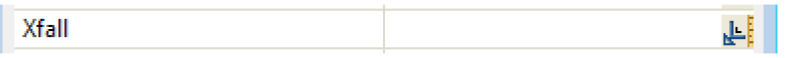
At each chainage ch between the **Start chainage** and **End chainage**, the Xfall is linearly interpolated by chainage between the Xfall at the **Start chainage** and the given Xfall **Xfall** at the **End chainage**.

This allows you to use the Xfall at the **End chainage** without knowing what the actual value of the Xfall is.

Go to the next section 21.2.2.2.4.3.4 Xfall -> Pos or return to 21.2.2.2.4.3 Type for Modifier Types "Modify Xfall, Hold Width/Height" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.3.4 Xfall -> Pos

Interpolate Between a Given Start Xfall & an Existing End Xfall



At each chainage ch between the **Start chainage** and **End chainage**, the Xfall is linearly interpolated by chainage between the given Xfall **Xfall** at the **Start chainage** and the Xfall at the **End chainage**.

This allows you to use the Xfall at the **Start chainage** without knowing what the actual value of the Xfall is.

Go to the next section 21.2.2.2.4.3.5 Pos -> or return to 21.2.2.2.4.3 Type for Modifier Types "Modify Xfall, Hold Width/Height" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.3.5 Pos ->

Keep the Same Xfall as at the Start Chainage

At each chainage ch between the **Start chainage** and **End chainage**, the Xfall is that at the **Start chainage**. That is, the Xfall has the constant value of what it is a the **Start chainage**.

This allows you to use the Xfall at the **Start chainage** without knowing what the actual value of the Xfall is.

Go to the next section 21.2.2.2.4.3.6 <- Pos or return to 21.2.2.2.4.3 Type for Modifier Types "Modify Xfall, Hold Width/Height" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.3.6 <- Pos

Keep the Same Xfall as at the End chainage

At each chainage ch between the **Start chainage** and **End chainage**, the Xfall is that at the **End chainage**. That is, the Xfall has the constant value of what it is a the **End chainage**.

This allows you to use the Xfall at the **End chainage** without knowing what the actual value of the Xfall is.

Go to the next section 21.2.2.2.4.3.7 Pos-> Pos: or return to 21.2.2.2.4.3 Type for Modifier Types "Modify Xfall, Hold Width/Height" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.3.7 Pos-> Pos:

Interpolate Between Existing Xfalls at Start & End Chainages

At each chainage ch between the **Start chainage** and **End chainage**, the Xfall is linearly interpolated by chainage between the Xfall at the **Start chainage** and the Xfall at the **End chainage**.

This allows you to use the Xfalls at the **Start chainage** and the **End chainage** without knowing what the actual value of the heights are.

Go to the next section 21.2.2.2.4.3.8 Parabola -> or return to 21.2.2.2.4.3 Type for Modifier Types "Modify Xfall, Hold Width/Height" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.3.8 Parabola ->

Calculation of Xfall for Type Parabola ->

The calculations are the same as the case for Width except Xfall is substituted for Width in all the formulae and diagrams. See 21.2.2.2.4.1.8 Parabola ->.

Go to the next section 21.2.2.2.4.3.9 <- Parabola or return to 21.2.2.2.4.3 Type for Modifier Types "Modify Xfall, Hold Width/Height" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.3.9 <- Parabola

Calculation of Xfall for Type <- Parabola

The calculations are the same as the case for Width except Xfall is substituted for Width in all the formulae and diagrams. See 21.2.2.2.4.1.9 <- Parabola.

Go to the next section 21.2.2.2.4.3.10 Circular-> or return to 21.2.2.2.4.3 Type for Modifier Types "Modify Xfall, Hold Width/Height" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.3.10 Circular->

Calculation of Xfall for Type Circular->

The calculations are the same as the case for Width except Xfall is substituted for Width in all the formulae and diagrams. See 21.2.2.2.4.1.10 Circular->.

Go to the next section 21.2.2.2.4.3.11 <-Circular or return to 21.2.2.2.4.3 Type for Modifier Types "Modify Xfall, Hold Width/Height" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.3.11 <-Circular

Calculation of Xfall for Type <-Circular

The calculations are the same as the case for Width except Xfall is substituted for Width in all the formulae and diagrams. See 21.2.2.2.4.1.11 <-Circular.

Go to the next section 21.2.2.2.4.3.12 Cubic or return to 21.2.2.2.4.3 Type for Modifier Types "Modify Xfall, Hold Width/Height" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.3.12 Cubic

Calculation of Xfall for Type Cubic

The calculations are the same as the case for Width except Xfall is substituted for Width in all the formulae and diagrams. See 21.2.2.2.4.1.12 Cubic.

Return to 21.2.2.2.4.3 Type for Modifier Types "Modify Xfall, Hold Width/Height" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.4 Type for Modifier Types "Modify Slope, Hold Width/Height"

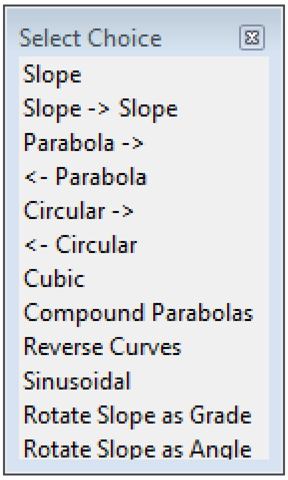
The choices for **Type** when **Modify type** is:

(a) **Modify Slope, Hold Width**

or

(b) Modify Slope, Hold Height

are



use a given slope for chainage range
interpolate between a given start slope and a given end slope
use a parabola from the current start slope
use a parabola that goes through the current end slope
use an arc from the current start slope
use an arc that goes through the current end slope
use a cubic between the current start and end slope
use a compound parabola between the current start and end slope
use reverse curves between the current start and end slopes
use a sinusoidal curve between the current start and end slopes

For the description of each **Type** see

21.2.2.2.4.4.1 Slope

21.2.2.2.4.4.2 Slope -> Slope

21.2.2.2.4.4.3 Parabola ->

21.2.2.2.4.4.4 <- Parabola

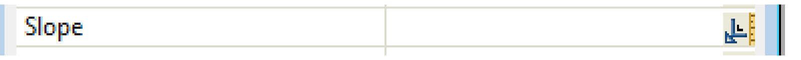
21.2.2.2.4.4.5 Circular->

21.2.2.2.4.4.6 <-Circular

21.2.2.2.4.4.7 Cubic

21.2.2.2.4.4.1 Slope

Use a Given Slope from the Start Chainage to the End Chainage

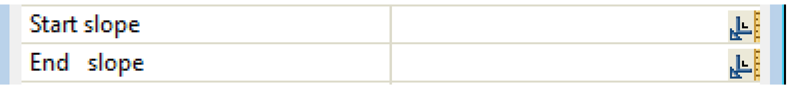


At each chainage ch between the **Start chainage** and **End chainage**, the width is that given in the **Slope** panel field. That is, the slope has the constant value of **Slope**.

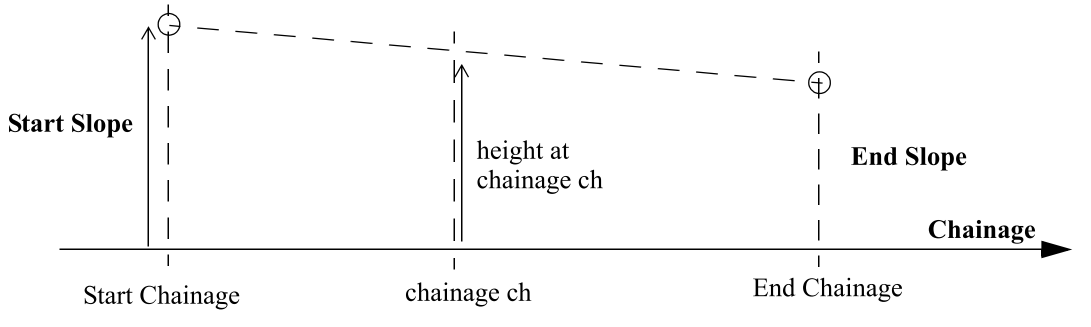
Go to the next section 21.2.2.2.4.4.2 Slope -> Slope or return to 21.2.2.2.4.4 Type for Modifier Types "Modify Slope, Hold Width/Height" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.4.2 Slope -> Slope

Interpolate Between a Given Start Slope and Given End Slope



At each chainage ch between the **Start chainage** and **End chainage**, the slope is linearly interpolated by chainage between the given **Start slope** and the **End slope**.



Chainage-Slope Diagram for Type Slope->Slope

Go to the next section 21.2.2.2.4.4.3 Parabola -> or return to 21.2.2.2.4.4 Type for Modifier Types "Modify Slope, Hold Width/Height" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.4.3 Parabola ->

Calculation of Slope for Type Parabola ->

The calculations are the same as the case for Width except Slope is substituted for Width in all the formulae and diagrams. See 21.2.2.2.4.1.8 Parabola ->.

Go to the next section 21.2.2.2.4.4.3 Parabola -> or return to 21.2.2.2.4.4 Type for Modifier Types "Modify Slope, Hold Width/Height" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.4.4 <- Parabola

Calculation of Slope for Type <- Parabola

The calculations are the same as the case for Width except Slope is substituted for Width in all the formulae and diagrams. See 21.2.2.2.4.1.9 <- Parabola.

Go to the next section 21.2.2.2.4.4.4 <- Parabola or return to 21.2.2.2.4.4 Type for Modifier Types "Modify Slope, Hold Width/Height" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.4.5 Circular->

Calculation of Slope for Type Circular->

The calculations are the same as the case for Width except Slope is substituted for Width in all the formulae and diagrams. See 21.2.2.2.4.1.10 Circular->.

Go to the next section 21.2.2.2.4.4.5 Circular-> or return to 21.2.2.2.4.4 Type for Modifier Types "Modify Slope, Hold Width/Height" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.4.6 <-Circular

Calculation of Slope for Type <-Circular

The calculations are the same as the case for Width except Slope is substituted for Width in all the formulae and diagrams. See 21.2.2.2.4.1.11 <-Circular.

Go to the next section 21.2.2.2.4.4.6 <-Circular or return to 21.2.2.2.4.4 Type for Modifier Types "Modify Slope, Hold Width/Height" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.4.4.7 Cubic

Calculation of Slope for Type Cubic

The calculations are the same as the case for Width except Slope is substituted for Width in all the formulae and diagrams. See 21.2.2.2.4.1.12 Cubic.

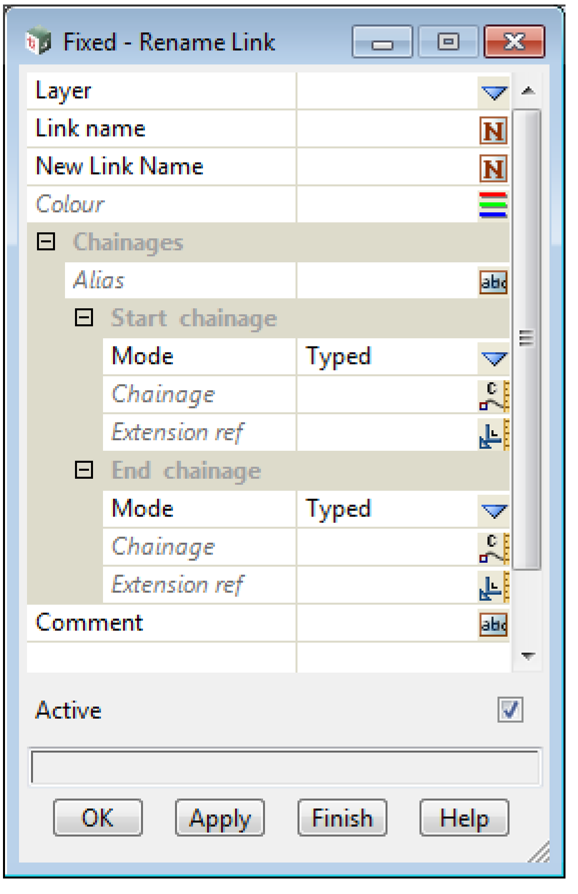
Go to the next section 21.2.2.2.4.4.7 Cubic or return to 21.2.2.2.4.4 Type for Modifier Types "Modify Slope, Hold Width/Height" or 21.2.2.2.4 Fixed - Modify Link

21.2.2.2.5 Fixed Link - Rename Link

The **Rename Link** option renames a link to a new name.

If strings are created in the ***Apply MTF*** then the original string stops and a string with the new name and colour is created between the two chainages. The original string starts again after the end chainage.

Selecting **Rename Link** brings up the **Fixed - Rename Link** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box	Design	available Layers

name of the layer that the link to rename is in. For information on Layers, see [21.1.1.2 MTF Links and Layers](#).

Old link name	name box		names.4d pop-up

the name of the link to rename between the chainages given by **Start mode** *and* **End mode**

New link name	name box		names.4d pop-up

the new name of the link between the chainages given by **Start mode** *and* **End mode.**
If strings are created in the Apply MTF then the old string will be stopped between the chainages, and a new string with the new name is created between the chainages.

Colour			available colours

if non blank, the colour for the new link.

If blank, the new has the colour of the old link.

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages to rename the link.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Continue to the next section 21.2.2.2.6 Fixed Link - Absolute to a Base Link or return to 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

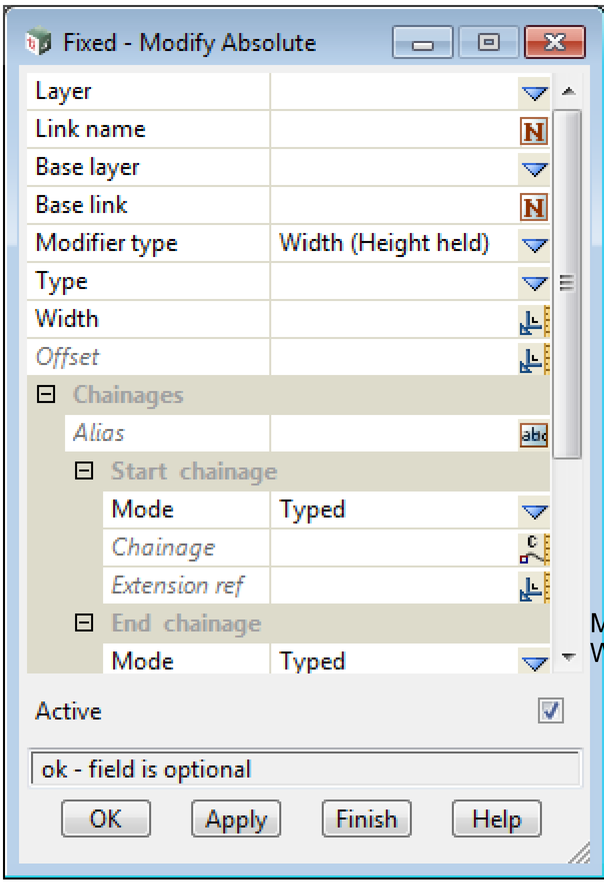
21.2.2.2.6 Fixed Link - Absolute to a Base Link

For an existing link, the **Absolute** option has most of the methods available in **Modify link** for defining either the width, height or xfall of the link, but the values are then **measured from a given Base link** rather than from the previous link.

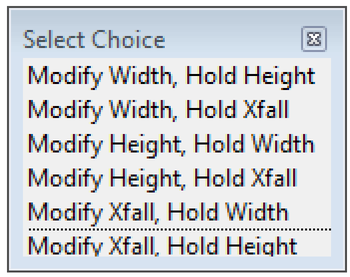
For example, over the chainage range, the **width from the Base link** can be set to a constant, be interpolated linearly between two widths, or have the width follow a parabolic, circular curve or cubic shape.

So **Absolute** is almost identical to ***Modify Link*** but values are measured from a given **Base Link** rather than the previous link (see 21.2.2.2.4 Fixed - Modify Link). But there is some additional parameters for controlling the values.

Selecting the **Absolute** brings up the **Fixed - Modify Absolute** panel.

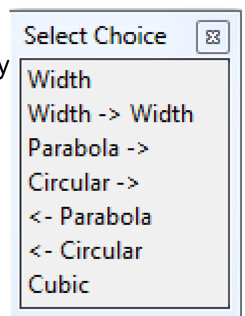


Choices for **Modifier Type**

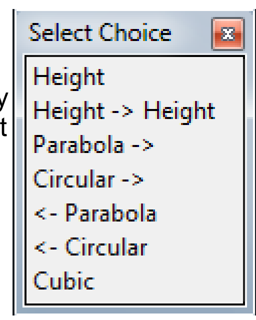


Choices for **Type** when **Modifier type** is

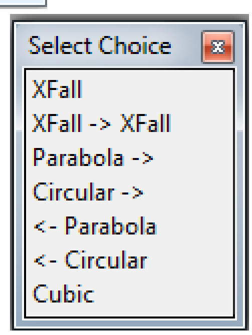
Modify Width



Modify Height



Modify Xfall



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box	Design	available Layers

name of the layer that the link to modify. For information on Layers, see 21.1.1.2 MTF Links and

Layers.

Link name name box names.4d pop-up

the name of the link to modify the width/height/xfall (by the method given in ***Type)*** *and the values are then measured from the* **Base link** *rather than the previous link.*

Base layer layer box Design available Layers

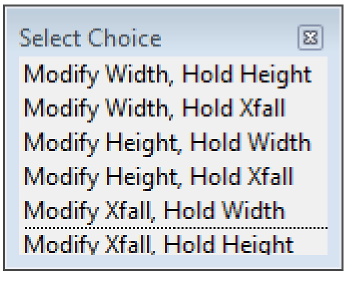
name of the layer for the **Base link**. *For information on Layers, see* 21.1.1.2 MTF Links and Layers.

Base link name box names.4d pop-up

the values of width/height/xfall calculated for **Link name** *are measured from the link given by* **Base link**.

Modifier type choice box

modify Width and take Height from current link
modify Width and take Xfall from current link
modify Height and take Width from current link
modify Height and take Xfall from current link
modify Xfall and take With from current link
modify Xfall and take Height from selected link



A fixed link can be modified in any of the **Modifier type** *ways regardless of how the link was originally defined in terms of width, height, xfall or slope.See* 21.1.2.1 Calculating Width, Height, Xfall or Slope from Original Definitions.

The ***Modify*** *width/height/xfall/slope is the part of the current link that is modified from its current value by the method given by* **Modifier type**.

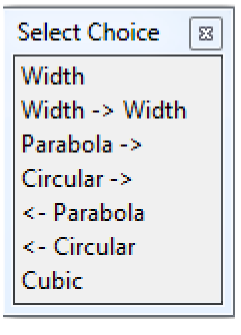
How *the width/height/xfall/slope is modified depends on the choice made in the* **Type** *field described below.*

The ***Hold*** *width/height/xfall is the part of the current link that is held at its current value for the link.*

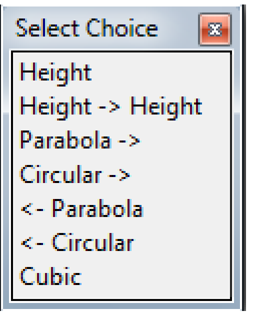
Type choice box

the choices for **Type** *depends on the choice selected for* **Modifier type**

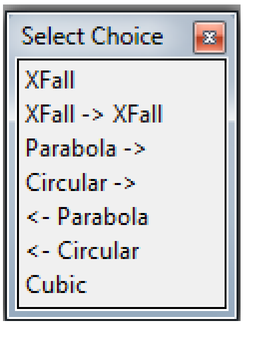
When **Modify type** is **Modify Width** the choices for **Type** are



When **Modify type** is **Modify Height** the choices for **Type** are



When **Modify type** is **Modify Xfall** the choices for **Type** are



For a description of the choices of **Type** *when the* **Modifier type** *is* ***Modify Width,*** *see* 21.2.2.2.6.1 For Modifier Types Modify Width, Hold Height or Xfall

For a description of the choices of **Type** *when the* **Modifier type** *is* ***Modify Height,*** *see* 21.2.2.2.6.2 For Modifier Types "Modify Height Maintain Width/Xfall"

For a description of the choices of **Type** *when the* **Modifier type** *is* ***Modify Xfall,*** *see* 21.2.2.2.6.3 For

Modifier Types "Modify Xfall Maintain Width/Height"

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages for modifying the link.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Continue to the next section 21.2.2.2.7 Fixed Link- Parallel or return to 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

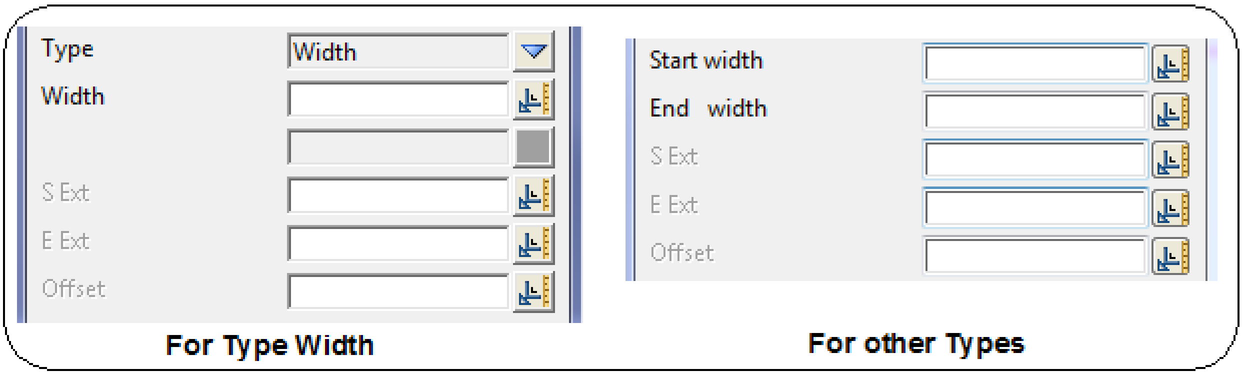
21.2.2.2.6.1 For Modifier Types Modify Width, Hold Height or Xfall

Go to Applying the Calculated Widths

Go to Calculating the Widths for each Type

Applying the Calculated Widths

When



Start width input measures menu

width at (Start mode + S Ext)

Start width can be positive or negative. See diagram after **Offset**

S Ext and E Ext have no effect for Type Width.

End width input measures menu

width at (End mode + E Ext)

End width can be positive or negative. See diagram after **Offset**

End width *does not exist for Type Width.*

S Ext/E Ext input measures menu

Start/End extension to add to the Start/end mode chainage. S Ext/E Ext can be positive or negative.

See diagram after **Offset**

Offset input measures menu

Final offset to be added to the width calculated at each point. Offset can be positive or negative.

For a chainage **ch** between the *Start mode* and the *End mode*, a *width* is calculated at chainage **ch** and that *width* goes out from the Base Link ***perpendicular*** to the Centreline.

The definition for calculating the width is best described in a **Chainage-Width** diagram. That is, in a plot with **Centreline chainage** as the **x-axis**, and **Width** at that chainage as the **y-axis**. This is similar to a long section plot where centreline chainage is the x-axis and height is the y-axis.

Then for a point **A** at chainage **ch** on the centreline, the **Width** calculated at chainage **ch** in the Chainage-Width diagram is used for the Width measured from the Base link corresponding to the section *Modify Width Maintain Height*, to the Centreline at chainage **ch**.

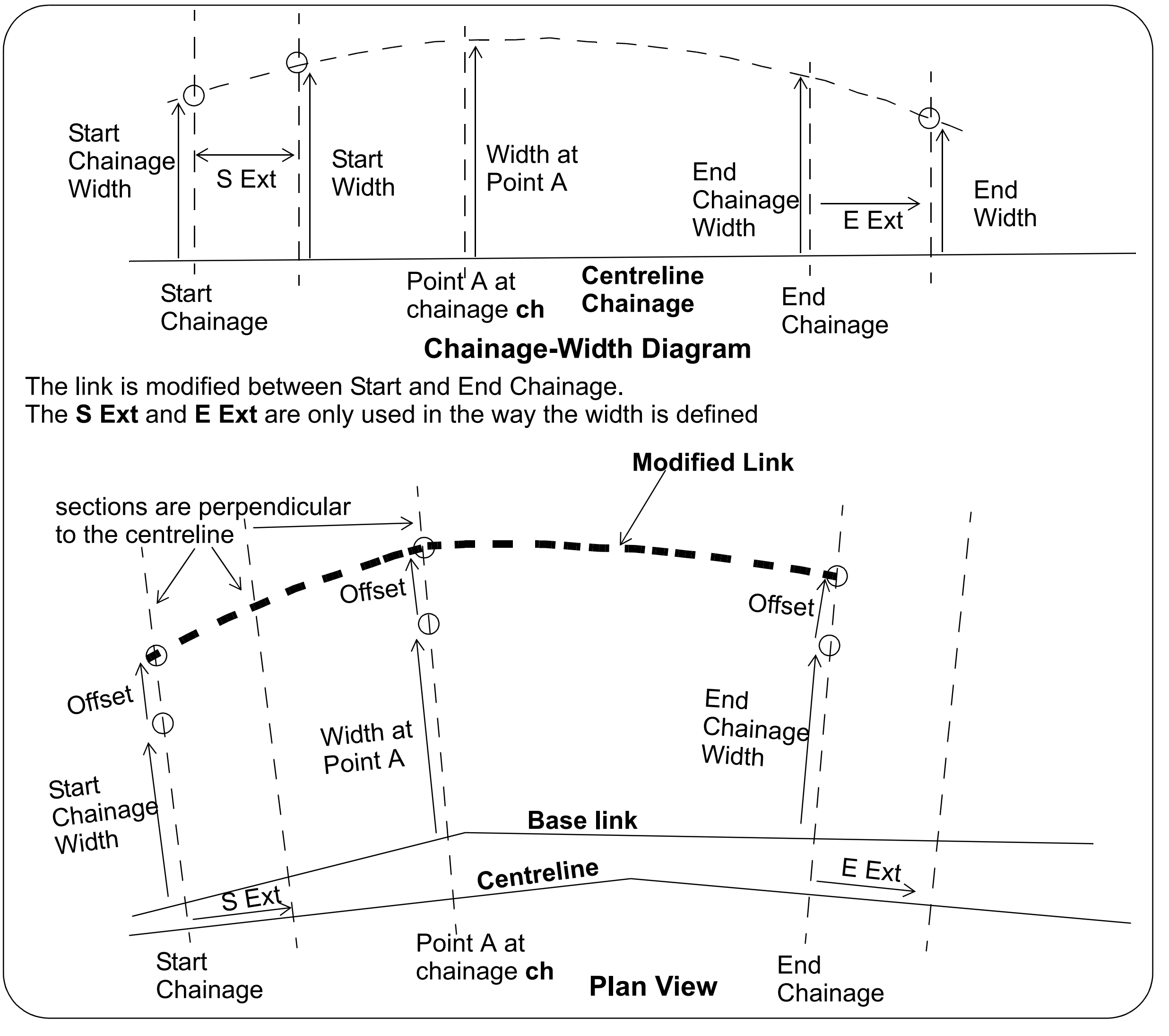
For the **Modify Width** commands, the method defining the widths may not be exactly at the Start/End mode chainages but at points a S Ext/E Ext chainage distance from the Start/End mode chainages.

Finally after the widths are calculated, a width Offset can be applied to all the widths.

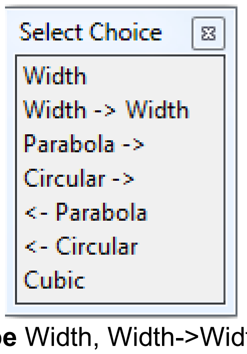
Although the definition for calculating the widths in the chainage-width diagram may involve points outside the Start mode/End mode chainage range, the calculated widths are **only applied** to the centreline **between** the **Start mode** chainage and the End mode chainage.

Please examine the diagrams below to help understand the process.

Relationship Between Chainage-Width Diagram and Centreline



Calculating the Widths for each Type



For **Type** Width, Width->Width, go to Calculation of Width for Types Width, Width -> Width

For **Type** Parabola->, go to Calculation of Width for Type Parabola ->

For **Type** Circular->, go to Calculation of Width for Type Circular->

For **Type** <-Parabola, go to [Calculation of WIdth for Type <- Parabola]

For **Type** <-Circular, go to [Calculation of Width for Type <-Circular]

For **Type** Cubic, go to [Calculation of Width for Type Cubic]

Calculation of Width for Types Width, Width -> Width

*In the chainage-width diagram, the **Start width** is the width at chainage **Start mode plus S Ext**, and **End width** is the width at chainage **End mode plus E Ext**.*

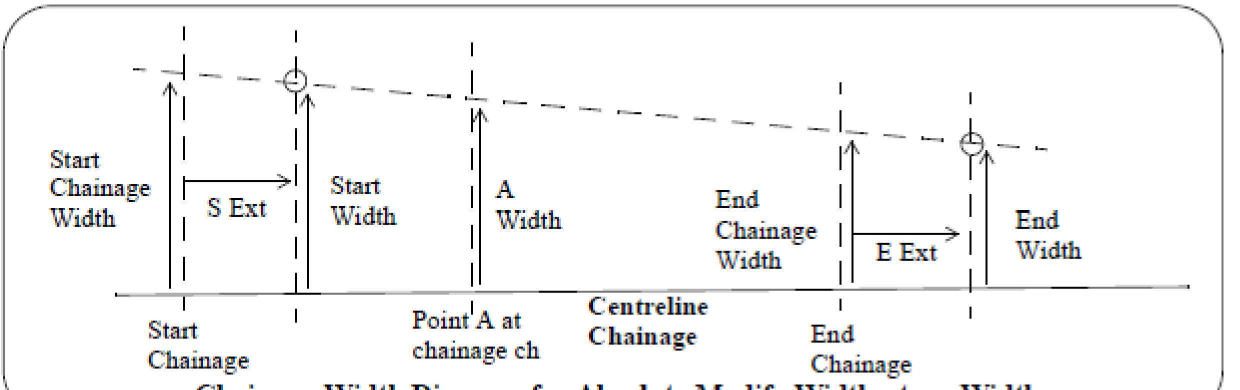
*The width is **linearly** interpolated between the **Start width** and the **End width**, **plus** the given **Offset** in the chainage-width diagram.*

That is, the rate change of width per unit chainage is

*rate of width change = (***End Width** - **Start Width***)/(***End mode** + **E Ext** - **Start mode** - **S Ext***)*

*and the width at chainage **ch** is*

*(***Start Width** + **Offset)** + *(ch -* **Start Mode** *chainage -* **S Ext***) * rate of width change*



Chainage-Width Diagram for Absolute Modify Width - type Width

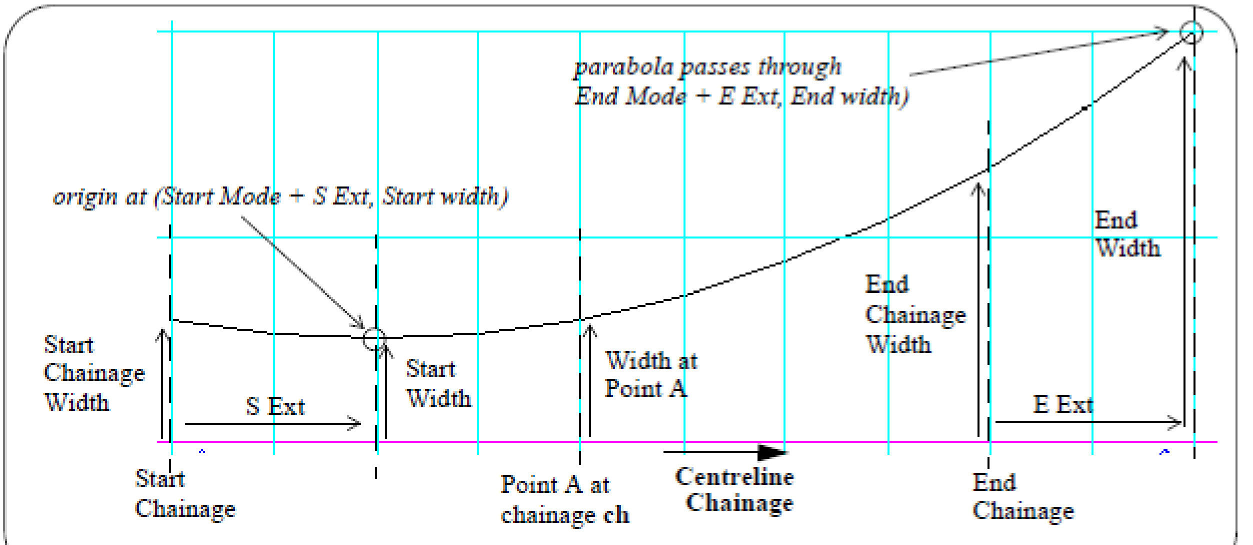
*These widths and the **Offset** are then used for the link being modified as the width that the link is **from the Base Link** when measured along the section perpendicular to the centreline. See .*

The Height/Xfall is maintained as that originally defined in the selected link.

Calculation of Width for Type Parabola ->

*In the chainage-width diagram, the **Start width** is the width at chainage **Start mode plus S Ext**, and **End width** is the width at chainage **End mode plus E Ext**.*

*Between the start position (**Start mode + S Ext**, **Start width)**, and the end position (**End mode + E Ext**, **End width**), the width varies as a **parabola** a*X*X with its origin at (**Start mode + S Ext**, **Start width)** and going through (**End mode + E Ext**, **End width**).*



Chainage-Width Diagram for Type ->Parabola

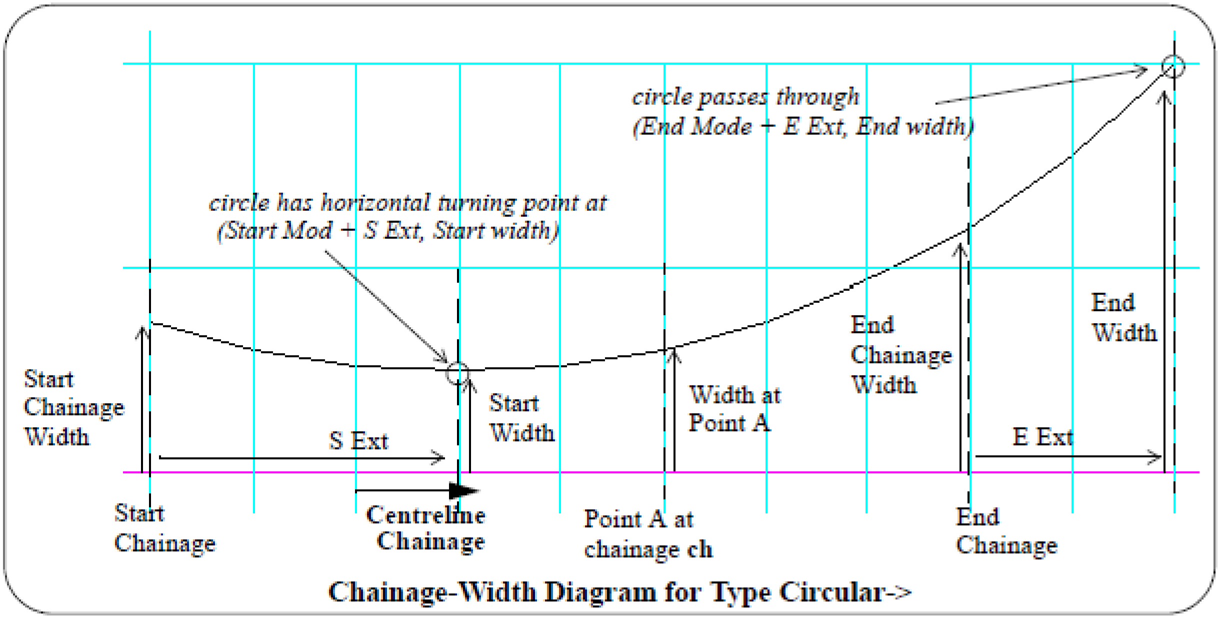
*These widths and the **Offset** are then used for the link being modified as the width that the link is **from the Base Link** when measured along the section perpendicular to the centreline. See*

The Height/Xfall is maintained as that originally defined in the selected link.

Calculation of Width for Type Circular->

*In the chainage-width diagram, the **Start width** is the width at chainage **Start mode plus S Ext**, and **End width** is the width at chainage **End mode plus E Ext**.*

*Between the start position (**Start mode + S Ext**, **Start width)**, and the end position (**End mode + E Ext**, **End width**), the width varies as a **circle** with its horizontal turning point at (**Start mode + S Ext**, **Start width)** and going through (**End mode + E Ext**, **End width**).*



Chainage-Width Diagram for Type Circular->

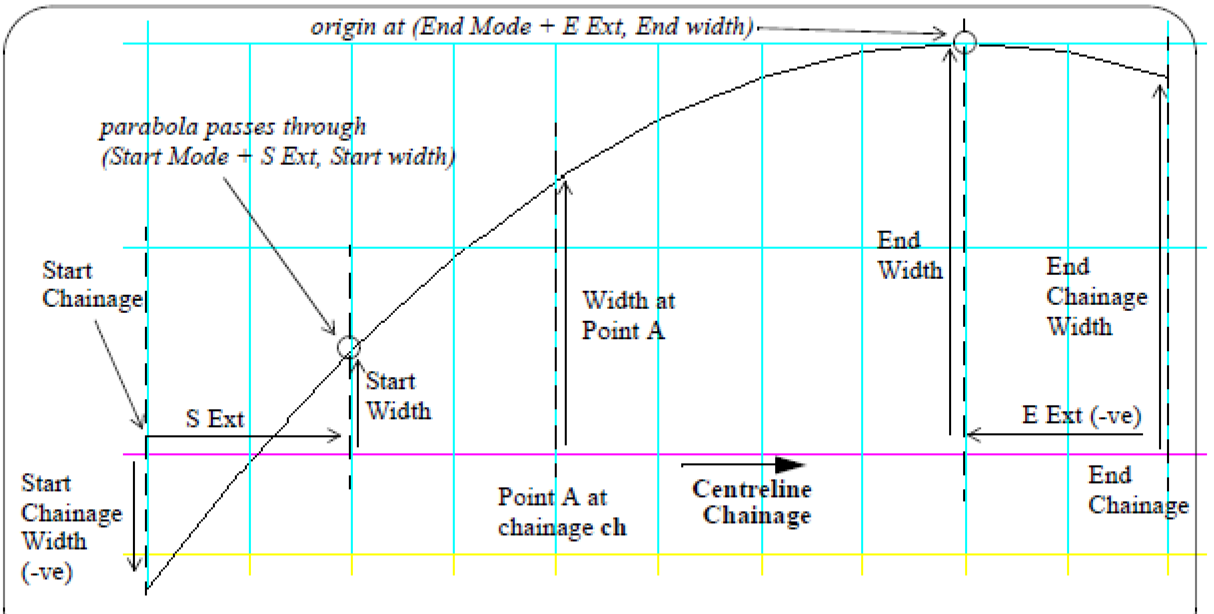
*These widths and the **Offset** are then used for the link being modified as the width that the link is **from the Base Link** when measured along the section perpendicular to the centreline. See* _

The Height/Xfall is maintained as that originally defined in the selected link.

Calculation of WIdth for Type <- Parabola

*In the chainage-width diagram, the **Start width** is the width at chainage **Start mode plus S Ext**, and **End width** is the width at chainage **End mode plus E Ext**.*

*Between the start position (**Start mode + S Ext**, **Start width)**, and the end position (**End mode + E Ext**, **End width**), the width varies as a **parabola** a*X*X with its origin at (**End mode + E Ext**, **End width)** and going through (**Start mode + S Ext**, **Start width)**.*



Chainage-Width Diagram for Type <-Parabola

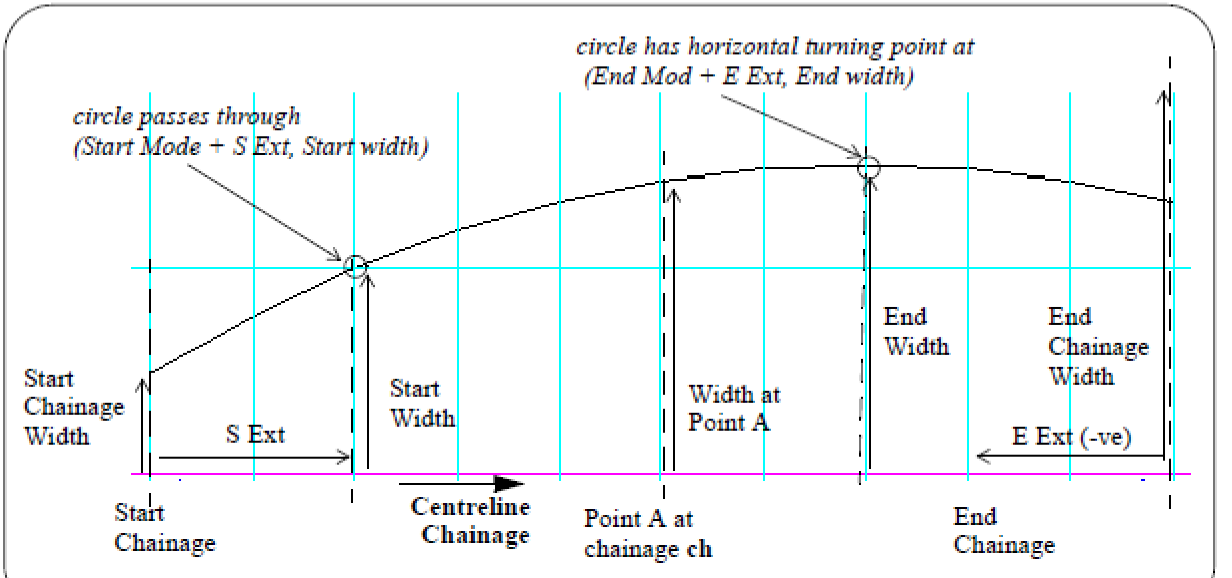
*These widths and the **Offset** are then used for the link being modified as the width that the link is **from the Base Link** when measured along the section perpendicular to the centreline. See* _

The Height/Xfall is maintained as that originally defined in the selected link.

Calculation of Width for Type <-Circular

*In the chainage-width diagram, the **Start width** is the width at chainage **Start mode plus S Ext**, and **End width** is the width at chainage **End mode plus E Ext**.*

*Between the start position (**Start mode + S Ext**, **Start width)**, and the end position (**End mode + E Ext**, **End width**), the width varies as a **circle** with its horizontal turning point at (**End mode + E Ext**, **End width**) and going through (**Start mode + S Ext**, **Start width)**.*



Chainage-Width Diagram for Type <-Circular

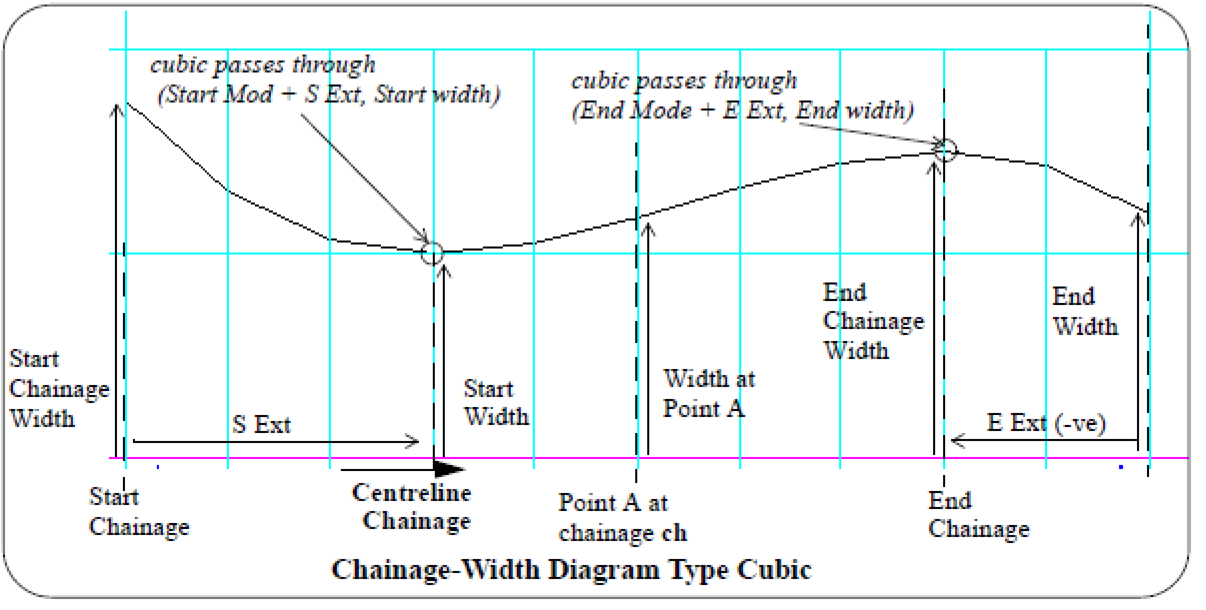
*These widths and the **Offset** are then used for the link being modified as the width that the link is **from the Base Link** when measured along the section perpendicular to the centreline. See*

The Height/Xfall is maintained as that originally defined in the selected link.

Calculation of Width for Type Cubic

*In the chainage-width diagram, the **Start width** is the width at chainage **Start mode plus S Ext**, and **End width** is the width at chainage **End mode plus E Ext**.*

*Between the start position (**Start mode + S Ext**, **Start width)**, and the end position (**End mode + E Ext**, **End width**), the width varies as a **cubic** going through (**Start mode + S Ext**, **Start width)** and (**End mode + E Ext**, **End width**).*



*These widths and the **Offset** are then used for the link being modified as the width that the link is **from the Base Link** when measured along the section perpendicular to the centreline. See*

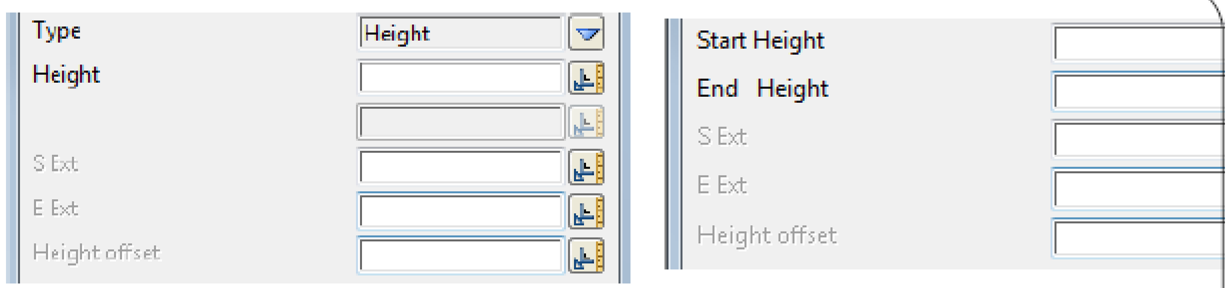
The Height/Xfall is maintained as that originally defined in the selected link.

21.2.2.2.6.2 For Modifier Types "Modify Height Maintain Width/Xfall"

Go to Applying the Calculated Heights

Go to Calculating the Heights for each Type

Applying the Calculated Heights



For Type Height   **For other Types**

Start height input measures menu

height at (Start mode + S Ext)

Start height can be positive or negative. See diagram after Height **Offset**

S Ext and E Ext have no effect for Type Height.

End width input measures menu

height at (End mode + E Ext)

End height can be positive or negative. See diagram after Height **Offset**

End height *does not exist for Type Height.*

S Ext/E Ext input measures menu

Start/End extension to add to the Start/end mode chainage. S Ext/E Ext can be positive or negative.

See diagram after **Offset**

Height offset input measures menu

Final height to be added to the height calculated at each point. **Height offset** *can be positive or negative.*

For a chainage **ch** between the *Start mode* and the *End mode*, a ***height*** is calculated for at **ch** chainage and that height is used for the link that goes out from the Base Link perpendicular to the Centreline.

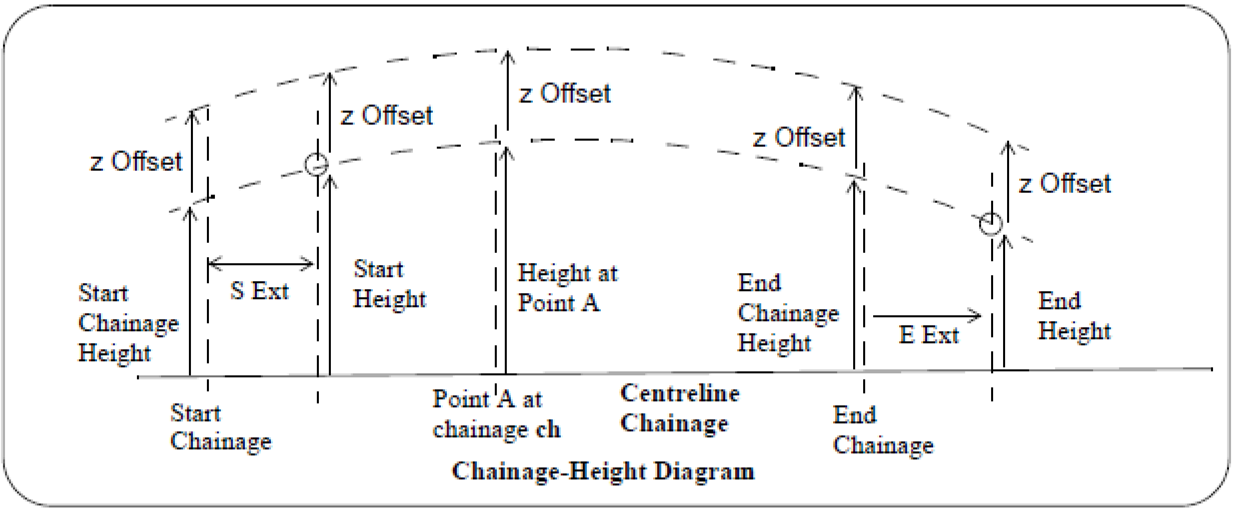
The definition for calculating the height is best described in a **Chainage-Height** diagram. That is, in a plot with **Centreline chainage** as the **x-axis**, and **Height** at that chainage as the **y-axis**. This is similar to a long section plot.

Then for a point **A** at a chainage **ch** on the centreline, the **Height** calculated at chainage **ch** in the Chainage-Height diagram is used for the Height measured from the Base link corresponding at the Centreline at chainage **ch**.

For the **Modify Height** commands, the method defining the heights may not be exactly at the Start/End mode chainages but at points a S Ext/E Ext chainage distance from the Start/End mode chainages.

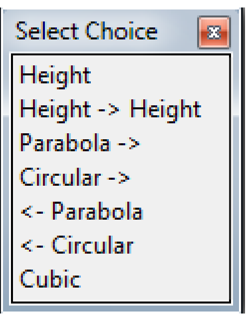
Finally after the heights are calculated, *Height offset* is added to all the heights.

Although the definition for calculating the heights in the chainage-height diagram may involve points outside the Start mode/End mode chainage range, the calculated heights are **only applied** to the centreline **between** the **Start mode** chainage and the End mode chainage.



Chainage-Height Diagram

Calculating the Heights for each Type



For **Type** Height, Height->Height, go to Calculation of Height for Types Height, Height -> Height

For **Type** Parabola->, go to Calculations for Type Parabola ->

For **Type** Circular->, go to Calculations for Type Circular->

For **Type** <-Parabola, go to Calculations for Type <- Parabola

For **Type** <-Circular, go to Calculations for Type <-Circular

For **Type** Cubic, go to Calculations for Type Cubic

Calculation of Height for Types Height, Height -> Height

*In the chainage-height diagram, the **Start height** is the height at chainage **Start mode plus S Ext**, and **End height** is the height at chainage **End mode plus E Ext**.*

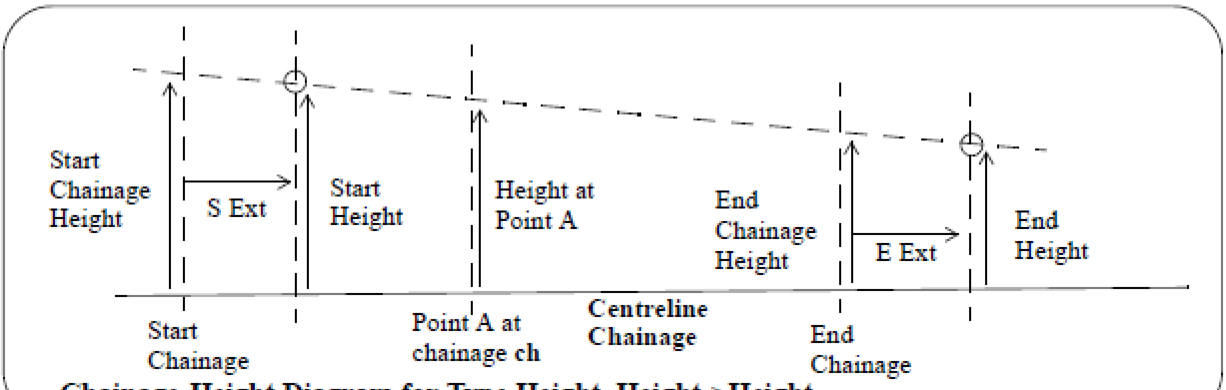
*The height is **linearly** interpolated between the **Start height** and the **End height**, **plus** the given **Height offset** in the chainage-width diagram.*

That is, the rate change of height per unit chainage is

*rate of height change = (***End height** - **Start height***)/(***End mode** + **E Ext** - **Start mode** - **S Ext***)*

*and the height at chainage **ch** is*

*(***Start height** + **Height offset)** + *(ch -* **Start Mode** *chainage -* **S Ex***) * rate of height change*



Chainage-Height Diagram for Type Height, Height->Height

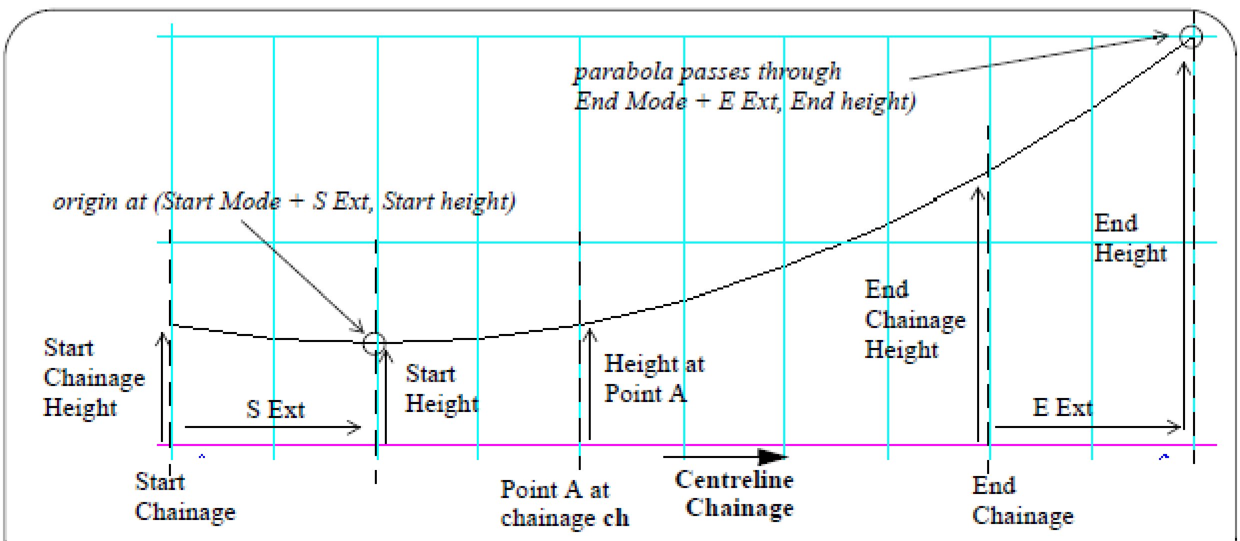
These heights and the ***Height offset*** *are then used for the link being modified as the height that the link is* ***from the Base Link*** *when measured along the section perpendicular to the centreline.*

The Width/Xfall is maintained as that originally defined in the selected link.

Calculations for Type Parabola ->

In the chainage-height diagram, the ***Start height*** *is the height at chainage* ***Start mode plus S Ext****, and* ***End height*** *is the height at chainage* ***End mode plus E Ext****.*

Between the start position ***(Start mode + S Ext****,* ***Start height)****, and the end position (****End mode + E Ext****,* ***End height****), the height varies as a* ***parabola*** *a*X*X with its origin at (****Start mode + S Ext****,* ***Start height)*** *and going through (****End mode + E Ext****,* ***End height****).*



Chainage-Height Diagram for Type ->Parabola

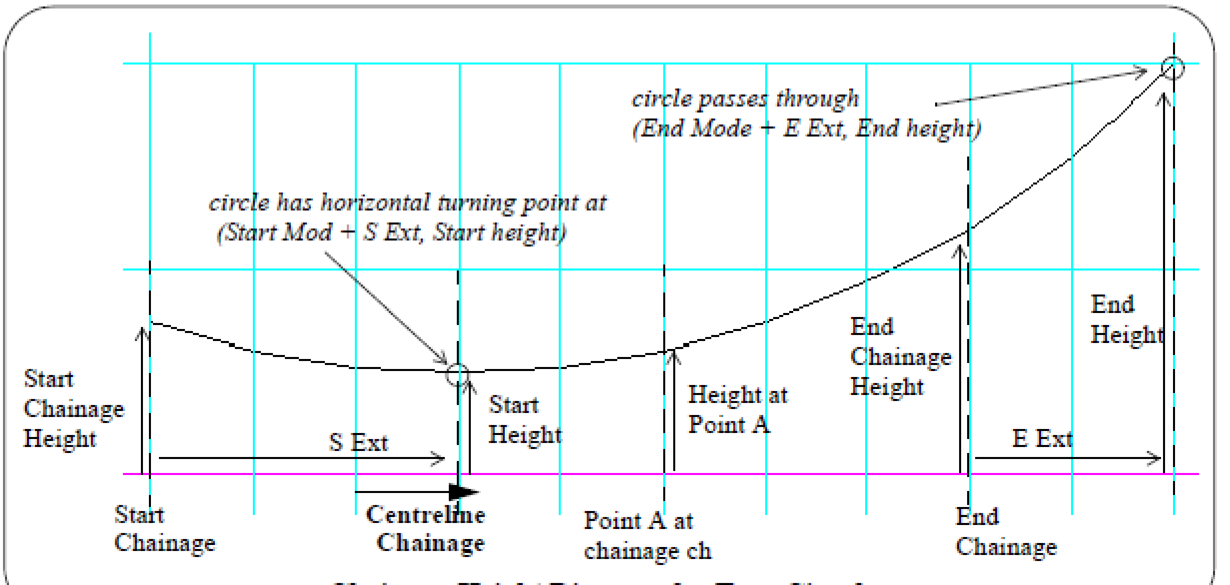
These heights and the ***Height offset*** *are then used for the link being modified as the height that the link is* ***from the Base Link*** *when measured along the section perpendicular to the centreline.*

The Width/Xfall is maintained as that originally defined in the selected link.

Calculations for Type Circular->

*In the chainage-height diagram, the **Start height** is the height at chainage **Start mode plus S Ext**, and **End height** is the height at chainage **End mode plus E Ext**.*

*Between the start position (**Start mode + S Ext**, **Start height)**, and the end position (**End mode + E Ext**, **End height**), the height varies as a **circle** with its horizontal turning point at (**Start mode + S Ext**, **Start height)** and going through (**End mode + E Ext**, **End height**).*



Chainage-Height Diagram for Type Circular->

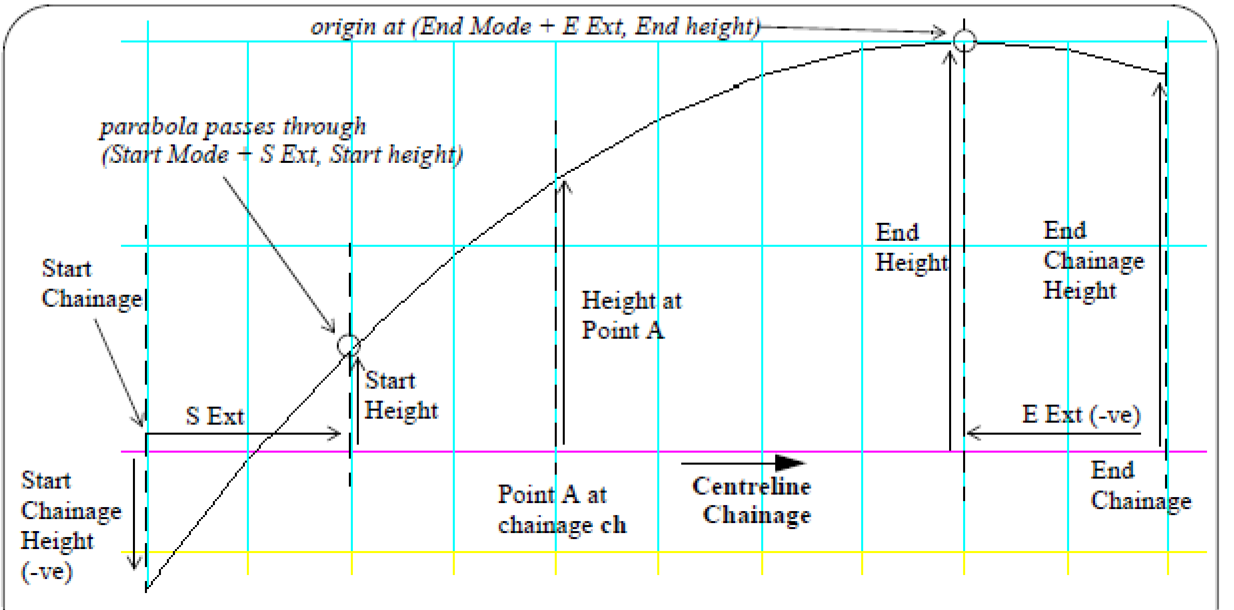
*These heights and the **Height offset** are then used for the link being modified as the width that the link is **from the Base Link** when measured along the section perpendicular to the centreline.*

The Width/Xfall is maintained as that originally defined in the selected link.

Calculations for Type <- Parabola

*In the chainage-height diagram, the **Start height** is the height at chainage **Start mode plus S Ext**, and **End height** is the height at chainage **End mode plus E Ext**.*

*Between the start position (**Start mode + S Ext**, **Start height)**, and the end position (**End mode + E Ext**, **End height**), the height varies as a **parabola** a*X*X with its origin at (**End mode + E Ext**, **End height)** and going through (**Start mode + S Ext**, **Start height**).*



Chainage-Height Diagram for Type <-Parabola

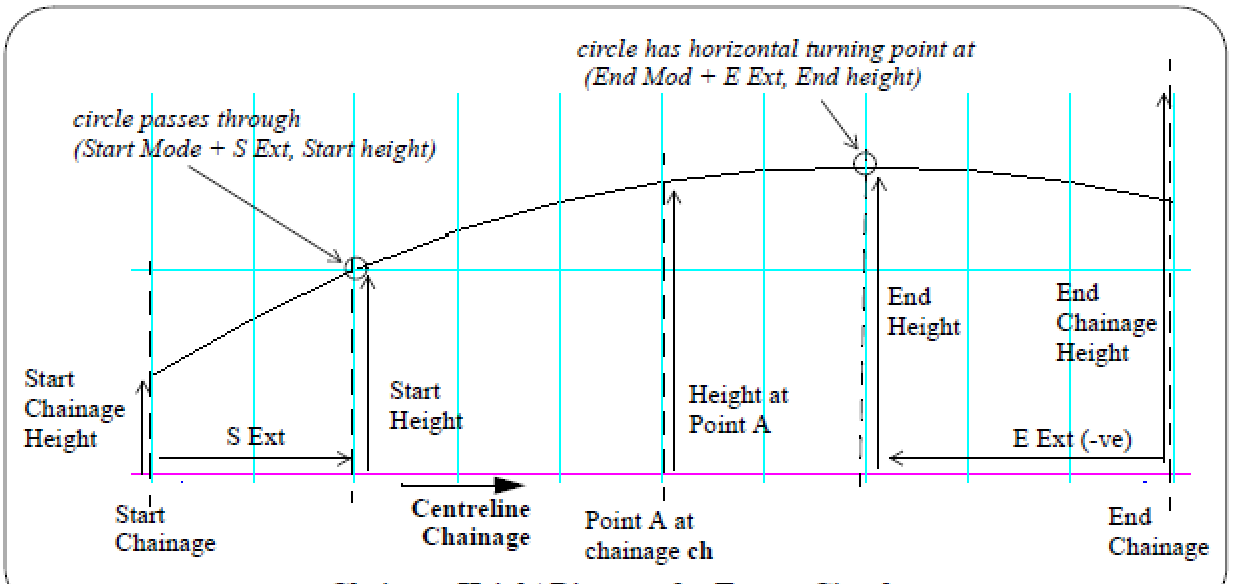
*These heights and the **Height offset** are then used for the link being modified as the width that the link is **from the Base Link** when measured along the section perpendicular to the centreline.*

The Width/Xfall is maintained as that originally defined in the selected link.

Calculations for Type <-Circular

*In the chainage-height diagram, the **Start height** is the height at chainage **Start mode plus S Ext**, and **End height** is the height at chainage **End mode plus E Ext**.*

*Between the start position (**Start mode + S Ext**, **Start height)**, and the end position (**End mode + E Ext**, **End height**), the height varies as a **circle** with its horizontal turning point at (**End mode + E Ext**, **End height**) and going through (**Start mode + S Ext**, **Start height)**.*



Chainage-Height Diagram for Type <-Circular

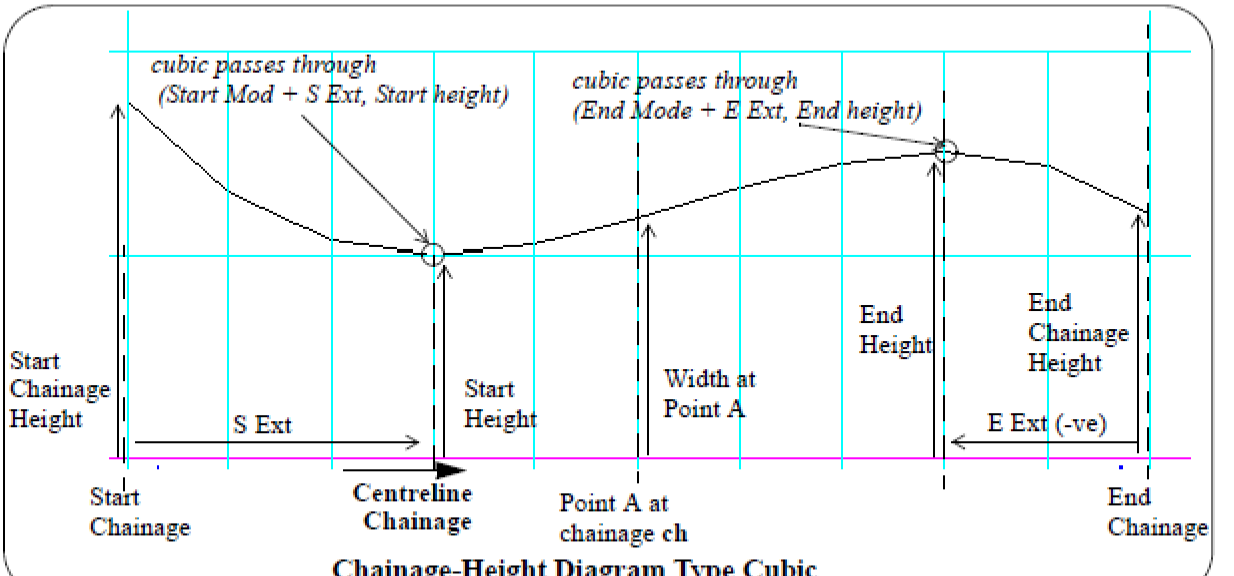
*These heights and the **Height offset** are then used for the link being modified as the height that the link is **from the Base Link** when measured along the section perpendicular to the centreline.*

The Width/Xfall is maintained as that originally defined in the selected link.

Calculations for Type Cubic

*In the chainage-height diagram, the **Start height** is the height at chainage **Start mode plus S Ext**, and **End height** is the height at chainage **End mode plus E Ext**.*

*Between the start position (**Start mode + S Ext**, **Start height)**, and the end position (**End mode + E Ext**, **End height**), the height varies as a **cubic** going through (**Start mode + S Ext**, **Start height)** and (**End mode + E Ext**, **End height**).*



Chainage-Height Diagram Type Cubic

*These heights and the **Height offset** are then used for the link being modified as the height that the link is **from the Base Link** when measured along the section perpendicular to the centreline.*

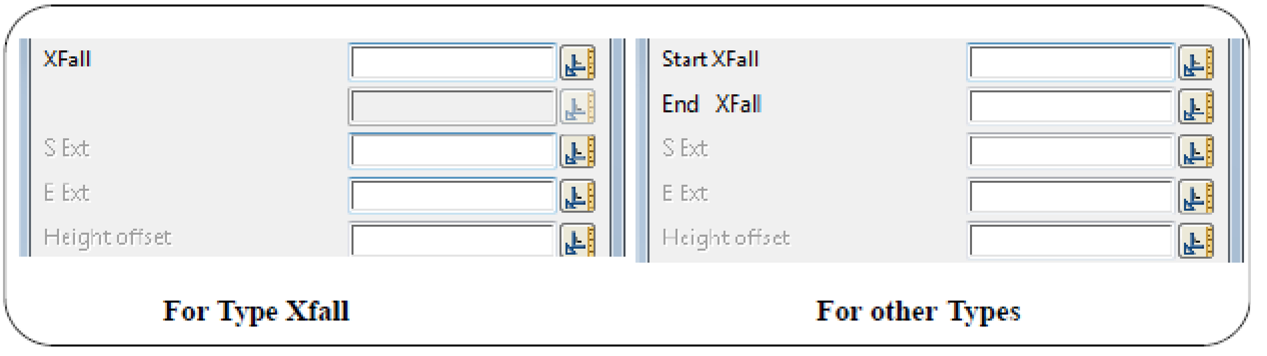
The Width/Xfall is maintained as that originally defined in the selected link.

21.2.2.2.6.3 For Modifier Types "Modify Xfall Maintain Width/Height"

Go to Applying the Calculated Xfall

Go to Calculating the Xfall for each Type

Applying the Calculated Xfall



Start xfall input measures menu

xfall at (Start mode + S Ext)

Start xfall can be positive or negative. See diagram after **Height offset**

S Ext and E Ext have no effect for Type Xfall.

End xfall input measures menu

xfall at (End mode + E Ext)

End xfall can be positive or negative. See diagram after **Offset**

End xfall *does not exist for Type Xfall.*

S Ext/E Ext input measures menu

Start/End extension to add to the Start/end mode chainage. S Ext/E Ext can be positive or negative.

See diagram after **Offset**

Height offset input measures menu

Final height offset to be added to the ***height*** *calculated at each point. Height offset can be positive or negative.*

For a chainage **ch** between the *Start mode* and the *End mode*, a ***xfall*** is calculated at **ch** chainage and that xfall is used for the link that goes out from the Base Link perpendicular to the Centreline.

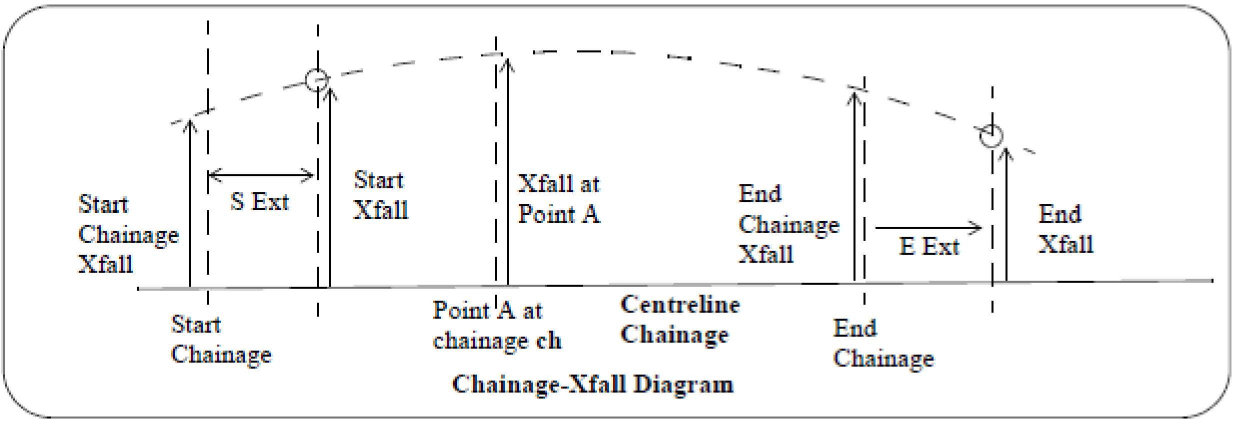
The definition for calculating the xfall is best described in a **Chainage-Xfall** diagram. That is, in a plot with **Centreline chainage** as the **x-axis**, and **Xfall** at that chainage as the **y-axis**. This is similar to a super elevation diagram where chainage is the x-axis and xfall is the y-axis.

Then for a point **A** at a chainage **ch** on the centreline, the **Xfall** calculated at chainage **ch** in the Chainage-Xfall diagram is used for the Xfall measured from the Base link corresponding at chainage **ch**.

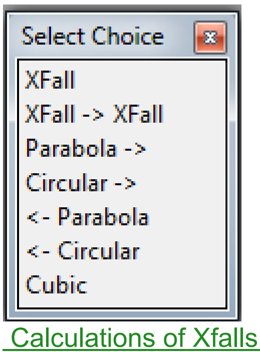
For the **Modify Xfall** commands, the method defining the xfalls may not be exactly at the Start/End mode chainages but at points a S Ext/E Ext chainage distance from the Start/End mode chainages.

Finally after the xfalls and height/widths are applied to the link, a Height offset is added **to all the heights**.

Although the definition for calculating the xfalls in the chainage-xfall diagram may involve points outside the Start mode/End mode chainage range, the calculated xfalls are **only applied** to the centreline **between** the **Start mode** chainage and the End mode chainage.



Calculating the Xfall for each Type



For **Type** Xfall, Xfall->Xfall, go to Calculations of Xfalls for Types Xfall, Xfall -> Xfall

For **Type** Parabola->, go to Calculations for Type Parabola ->

For **Type** Circular->, go to Calculations for Type Circular->

For **Type** <-Parabola, go to Calculations for Type <- Parabola

For **Type** <-Circular, go to Calculations for Type <-Circular

For **Type** Cubic, go to Calculations for Type Cubic

Calculations of Xfalls for Types Xfall, Xfall -> Xfall

*In the chainage-xfall diagram, the **Start xfall** is the xfall at chainage **Start mode plus S Ext**, and **End xfall** is the xfall at chainage **End mode plus E Ext**.*

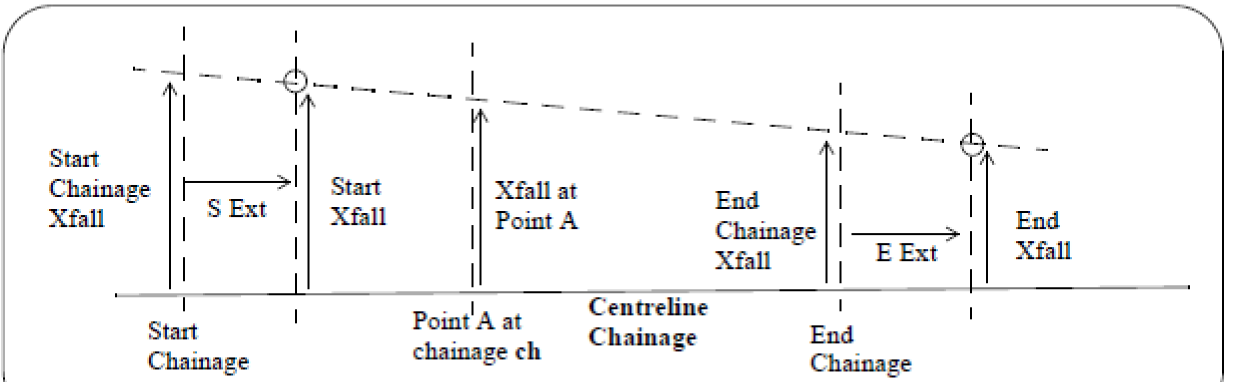
*The xfall is **linearly** interpolated between the **Start xfall** and the **End xfall**, **plus** the given **Xfall offset** in the chainage-width diagram.*

That is, the rate change of xfall per unit chainage is

*rate of xfall change = (***End xfall** - **Start xfall***)/(***End mode** + **E Ext** - **Start mode** - **S Ext***)*

*and the xfall at chainage **ch** is*

*(***Start xfall** + **Xfall offset)** + *(ch* - **Start Mode** *chainage* - **S Ex***) * rate of xfall change*



Chainage-Height Diagram for Type Xfall, Xfall->Xfall

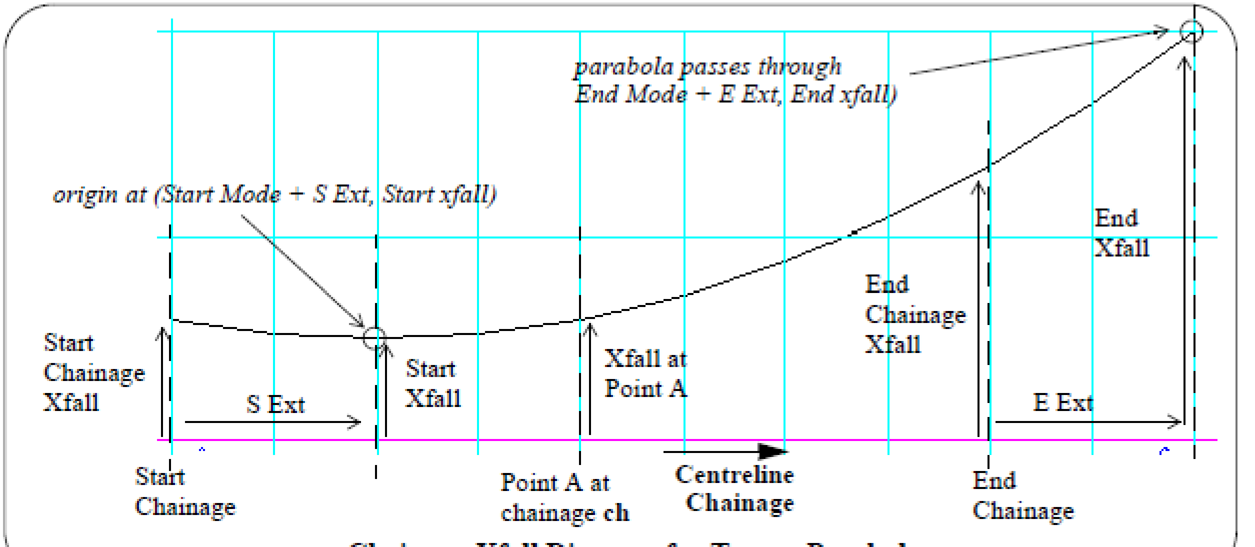
*These xfalls and the **Height offset** are then used for the link being modified as the xfall that the link is **from the Base Link** when measured along the section perpendicular to the centreline.*

The Width/Height is maintained as that originally defined in the selected link.

Calculations for Type Parabola ->

*In the chainage-xfall diagram, the **Start xfall** is the xfall at chainage **Start mode plus S Ext**, and **End xfall** is the xfall at chainage **End mode plus E Ext**.*

*Between the start position **(Start mode + S Ext, Start xfall)**, and the end position **(End mode + E Ext, End xfall)**, the xfall varies as a **parabola** a*X*X with its origin at **(Start mode + S Ext, Start xfall)** and going through **(End mode + E Ext, End xfall)**.*



Chainage-Xfall Diagram for Type ->Parabola

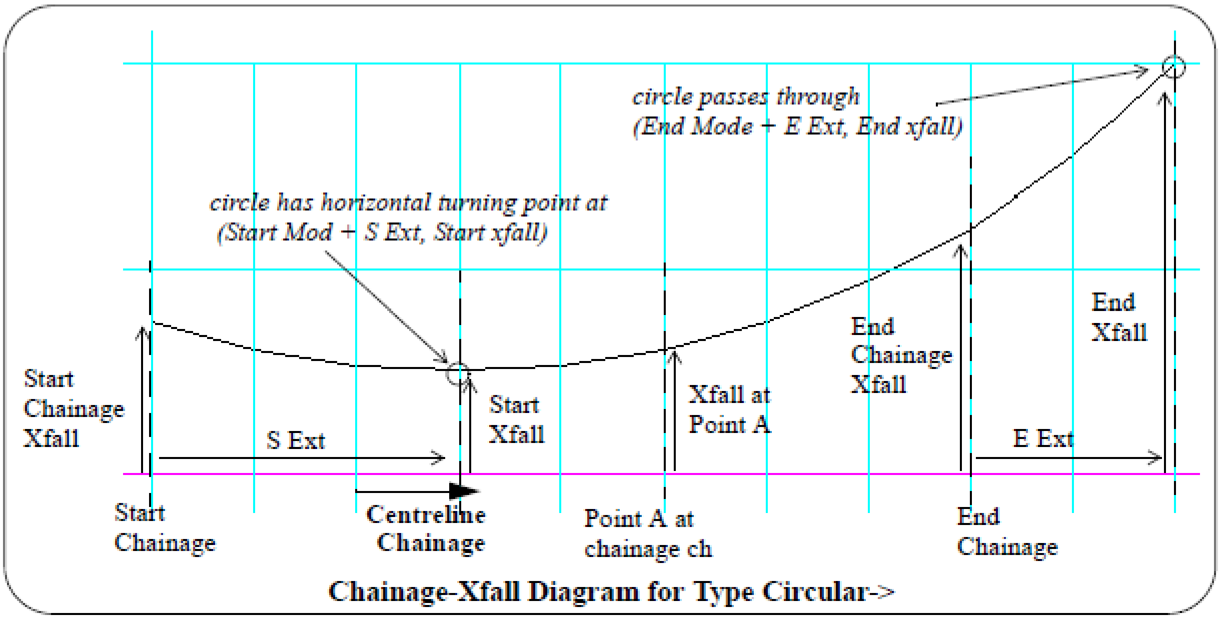
*These xfalls and the **Height offset** are then used for the link being modified as the xfall that the link is **from the Base Link** when measured along the section perpendicular to the centreline.*

The Width/Height is maintained as that originally defined in the selected link.

Calculations for Type Circular->

*In the chainage-xfall diagram, the **Start xfall** is the xfall at chainage **Start mode plus S Ext**, and **End xfall** is the xfall at chainage **End mode plus E Ext**.*

*Between the start position (**Start mode + S Ext**, **Start xfall)**, and the end position (**End mode + E Ext**, **End xfall**), the xfall varies as a **circle** with its horizontal turning point at (**Start mode + S Ext**, **Start xfall)** and going through (**End mode + E Ext**, **End xfall**).*

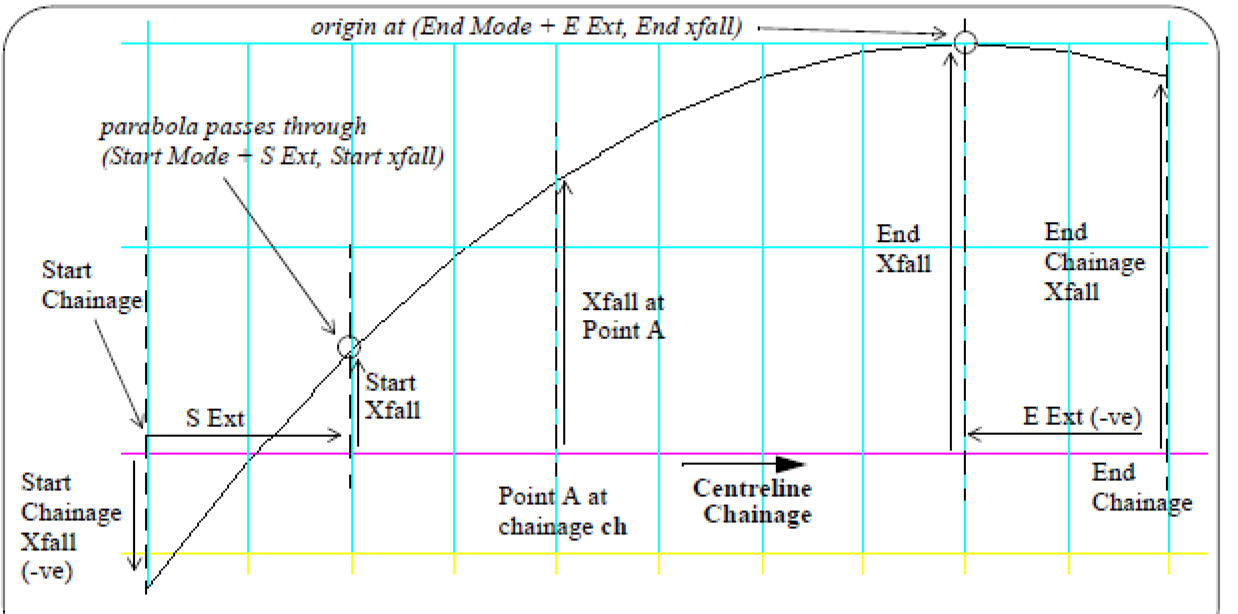


*These xfalls and the **Height offset** are then used for the link being modified as the xfall that the link is **from the Base Link** when measured along the section perpendicular to the centreline.*

Calculations for Type <- Parabola

*In the chainage-xfall diagram, the **Start xfall** is the xfall at chainage **Start mode plus S Ext**, and **End xfall** is the xfall at chainage **End mode plus E Ext**.*

*Between the start position (**Start mode + S Ext**, **Start xfall)**, and the end position (**End mode + E Ext**, **End xfall**), the xfall varies as a **parabola** a*X*X with its origin at (**End mode + E Ext**, **End xfall)** and going through (**Start mode + S Ext**, **Start xfall**).*



Chainage-Xfall Diagram for Type <-Parabola

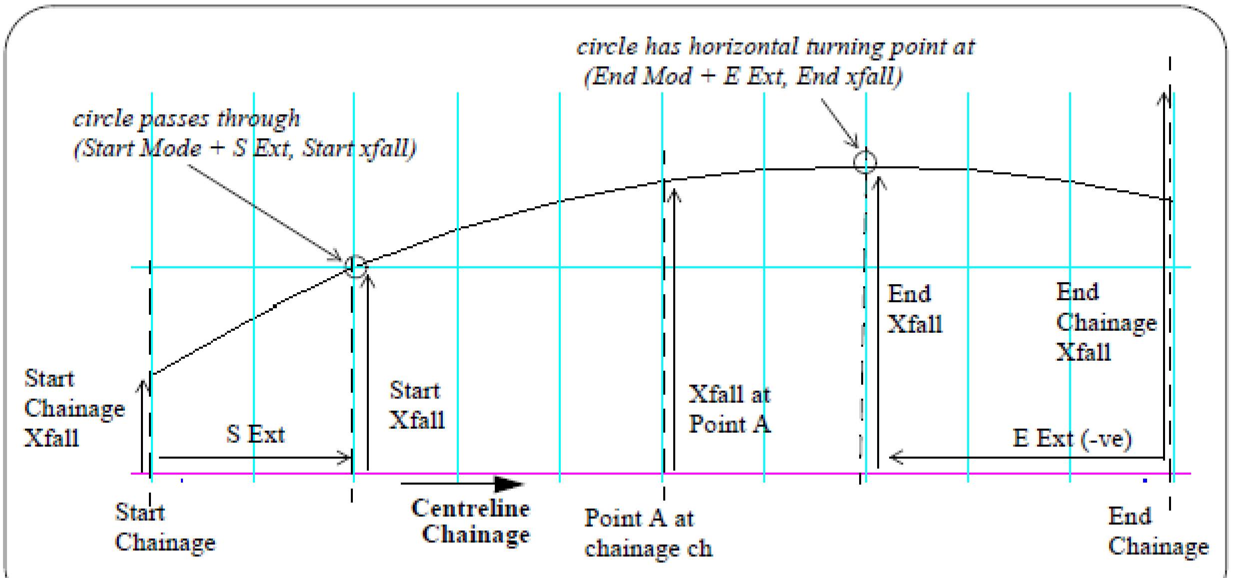
*These xfalls and the **Height offset** are then used for the link being modified as the xfall that the link is **from the Base Link** when measured along the section perpendicular to the centreline.*

The Width/Height is maintained as that originally defined in the selected link.

Calculations for Type <-Circular

*In the chainage-xfall diagram, the **Start xfall** is the xfall at chainage **Start mode plus S Ext**, and **End xfall** is the xfall at chainage **End mode plus E Ext**.*

*Between the start position (**Start mode + S Ext**, **Start xfall)**, and the end position (**End mode + E Ext**, **End xfall**), the xfall varies as a **circle** with its horizontal turning point at (**End mode + E Ext**, **End xfall**) and going through (**Start mode + S Ext**, **Start xfall)**.*



Chainage-Xfall Diagram for Type <-Circular

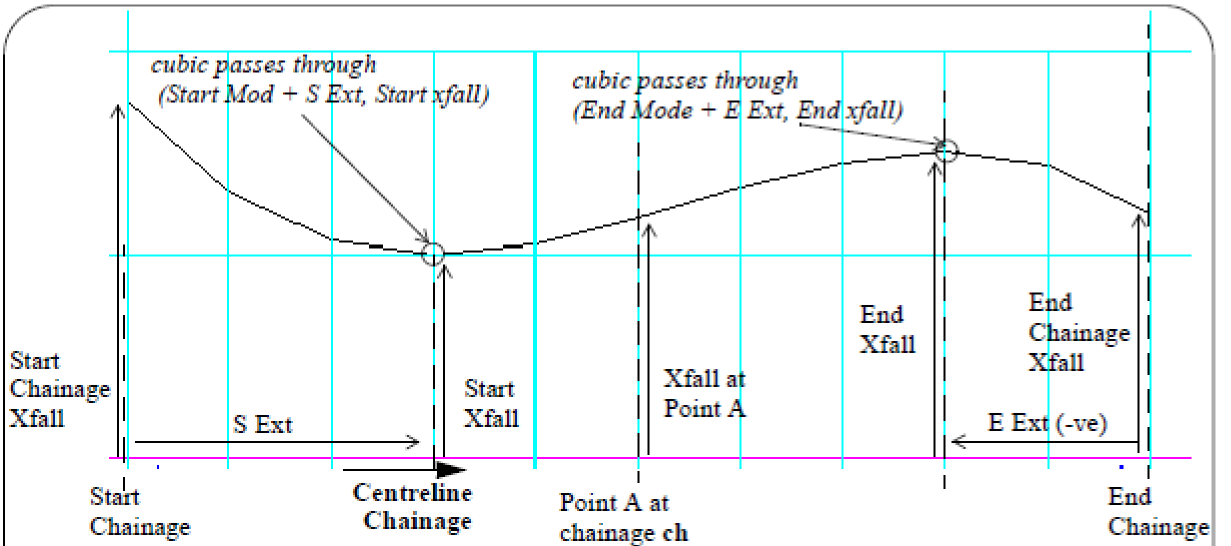
*These xfalls and the **Height offset** are then used for the link being modified as the xfall that the link is **from the Base Link** when measured along the section perpendicular to the centreline.*

The Width/Height is maintained as that originally defined in the selected link.

Calculations for Type Cubic

*In the chainage-xfall diagram, the **Start xfall** is the xfall at chainage **Start mode plus S Ext**, and **End xfall** is the xfall at chainage **End mode plus E Ext**.*

*Between the start position (**Start mode + S Ext**, **Start xfall)**, and the end position (**End mode + E Ext**, **End xfall**), the xfall varies as a **cubic** going through (**Start mode + S Ext**, **Start xfall)** and (**End mode + E Ext**, **End xfall**).*



Chainage-Xfall Diagram Type Cubic

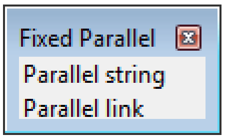
*These xfalls and the **Height offset** are then used for the link being modified as the xfall that the link is **from the Base Link** when measured along the section perpendicular to the centreline.*

The Width/Height is maintained as that originally defined in the selected link.

Continue to the next section 21.2.2.2.7 Fixed Link- Parallel or return to 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

21.2.2.2.7 Fixed Link- Parallel

The **Parallel** walk-right brings up the **fixed Parallel** menu with options to adjust all the links between specified fixed links and make them parallel offsets of a given link, of a selected string.



To go straight to the documentation on each of the options on the **Fixed** modifier menu:

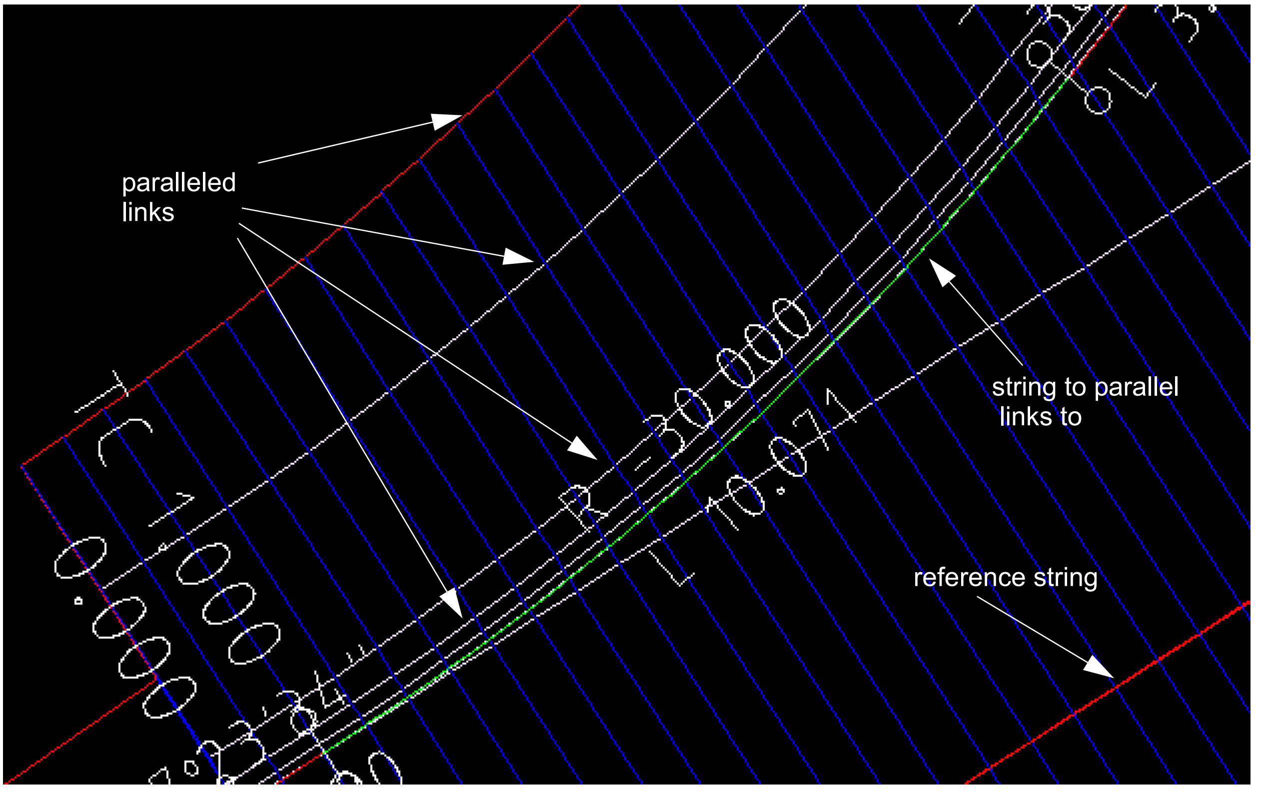
See

Parallel string	21.2.2.2.7.1 Fixed Link - Parallel Links to String
Parallel link	21.2.2.2.7.2 Fixed Link - Parallel Links to Link

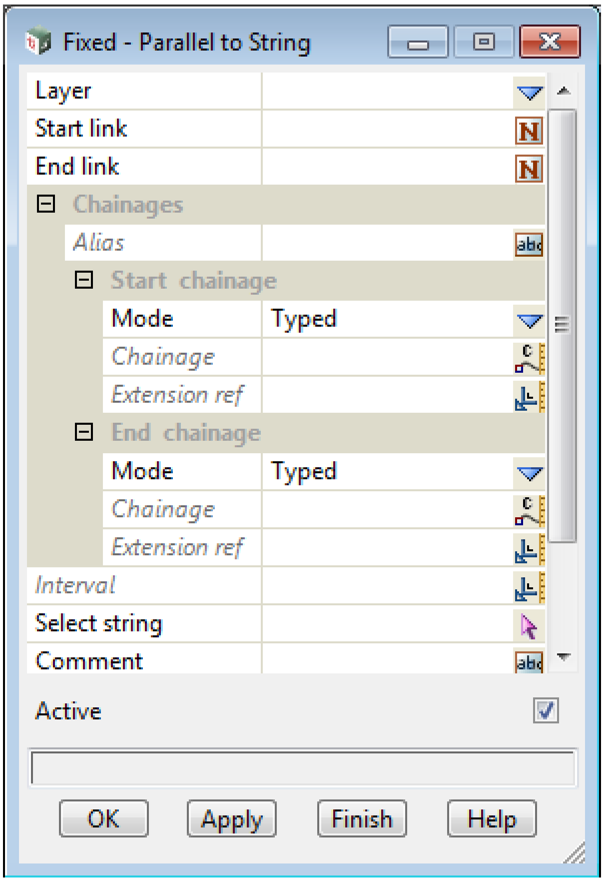
21.2.2.2.7.1 Fixed Link - Parallel Links to String

The **Parallel string** option takes all the fixed links between two nominated link names (inclusive) and adjusts them so that the specified widths, cross falls and heights of the links are used **perpendicular** to a **selected string** rather than measured perpendicular to the reference string.

If sections and strings are created in the ***Apply MTF,*** then vertices are still created on the section and on the strings **perpendicular** to the reference string.



Selecting **Parallel** brings up the **Fixed - Parallel to String** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box	Design	available Layers

name of the layer that the links to parallel are in. For information on Layers, see [21.1.1.2 MTF Links and Layers](#).

Start link	name box		names.4d pop-up

the name of the link to start making the widths distances measure parallel to the selected string.

End link	name box		names.4d pop-up

the name of the last link to make the widths distances measure parallel to the selected string.

Select string	string-select		

select the string to measure perpendicular to. The width, cross fall and height that a link is from the previous link are still used but they are measured along a perpendicular to the selected string rather than the reference string.

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages to parallel the links.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

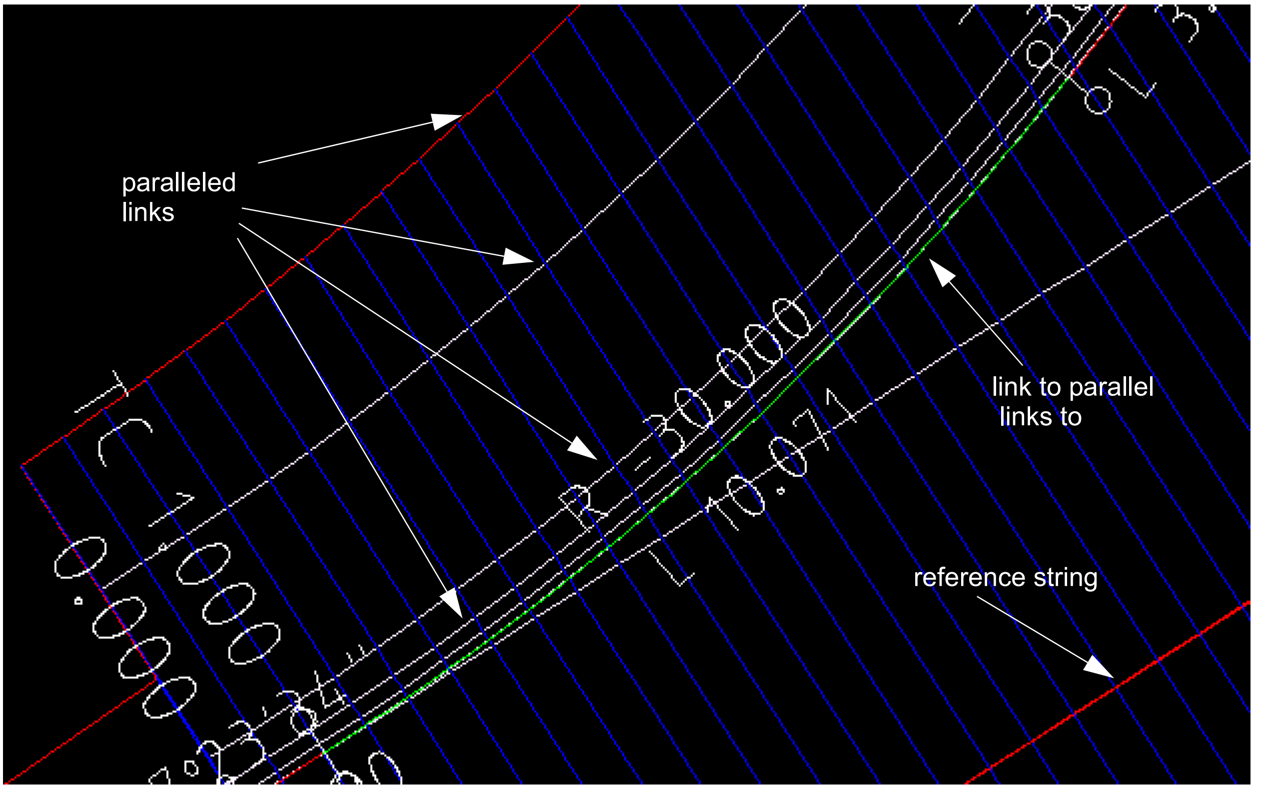
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Continue to the next section 21.2.2.2.7.2 Fixed Link - Parallel Links to Link or return to 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

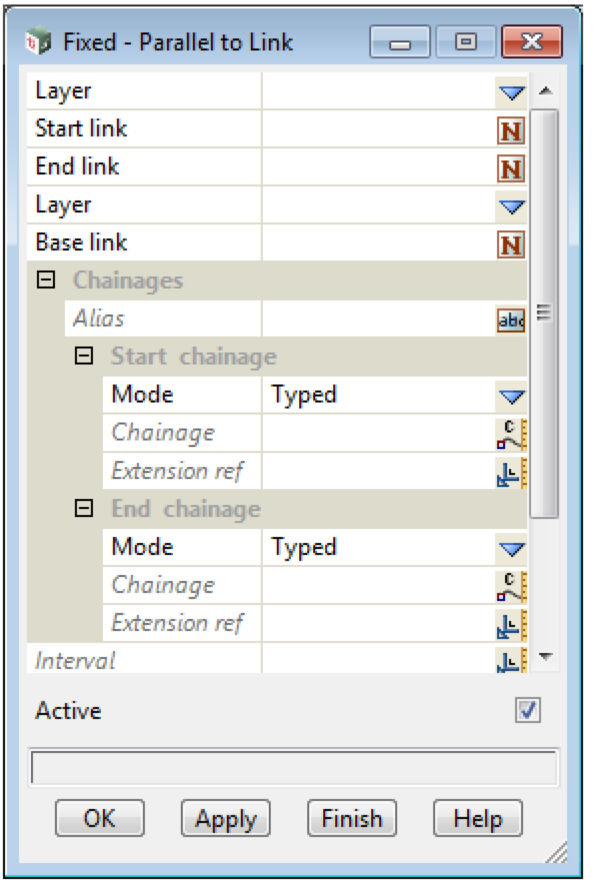
21.2.2.2.7.2 Fixed Link - Parallel Links to Link

The **Parallel link** option takes all the fixed links between two nominated link names (inclusive) and adjusts them so that the specified widths, cross falls and heights between the links are calculated perpendicular to a selected link **Base link** rather than measured perpendicular to the reference string.

If sections and strings are created in the ***Apply MTF,*** then vertices are still created on the section and on the strings **perpendicular** to the reference string.



Selecting **Parallel link** brings up the **Fixed - Parallel to Link** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box	Design	available Layers

name of the layer that the links to parallel are in. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Start Link	name box		names.4d pop-up

the name of the link to start making the widths distances measure parallel to the **Base link**.

End link	name box		names.4d pop-up

the name of the last link to make the widths distances measure parallel to the selected link.

Base Layer	layer box	Design	available Layers

name of the layer for the **Base link.** *For information on Layers, see* 21.1.1.2 MTF Links and Layers.

Base link	name box		

select the link to use to measure perpendicular to. The width, cross fall and height that a link is from the previous link are still used but they are measured along a perpendicular to the **Base link** *rather than the reference string.*

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages to parallel the links.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

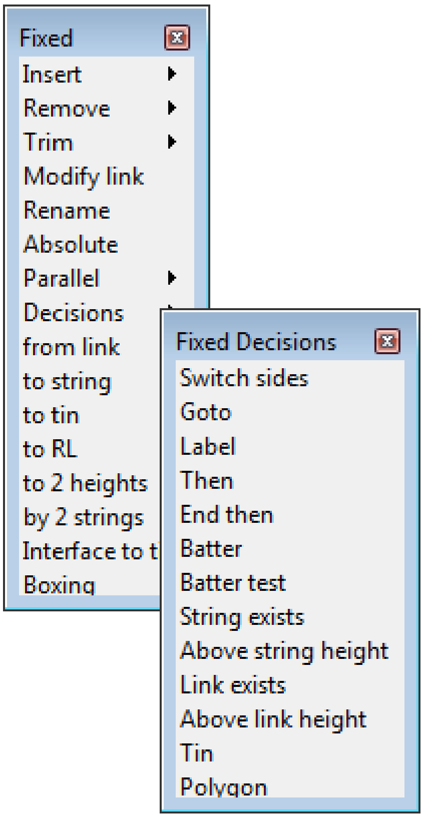
Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Continue to the next section 21.2.2.2.8 Fixed Link - Decisions or return to 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

21.2.2.2.8 Fixed Link - Decisions

The **Decisions** walk-right brings up the **Fixed Decisions** menu with options to make decisions on tests such as being able to batter into a tin, or how deep you are under a tin. There are labels and gotos to build complex tests.



See

Switch side	21.2.2.2.8.1 Fixed Decisions - Switch Sides
Goto	21.2.2.2.8.2 Fixed Decisions - Goto
Label	21.2.2.2.8.3 Fixed Decisions - Label
Then	21.2.2.2.8.4 Fixed Decisions - Then
End then	21.2.2.2.8.5 Fixed Decisions - End Then
Batter	21.2.2.2.8.6 Fixed Decisions - Batter
Batter test	21.2.2.2.8.7 Fixed Decisions - Batter Test
String exists	21.2.2.2.8.8 Fixed Decisions - String Exists
Above string height	21.2.2.2.8.9 Fixed Decisions - Above String Height
Link exists	21.2.2.2.8.10 Fixed Decisions - Link Exists
Above link height	21.2.2.2.8.11 Fixed Decisions - Above Link Height
Tin	21.2.2.2.8.12 Fixed Decisions - Tin
Polygon	21.2.2.2.8.13 Fixed Decisions - Polygon

21.2.2.2.8.1 Fixed Decisions - Switch Sides

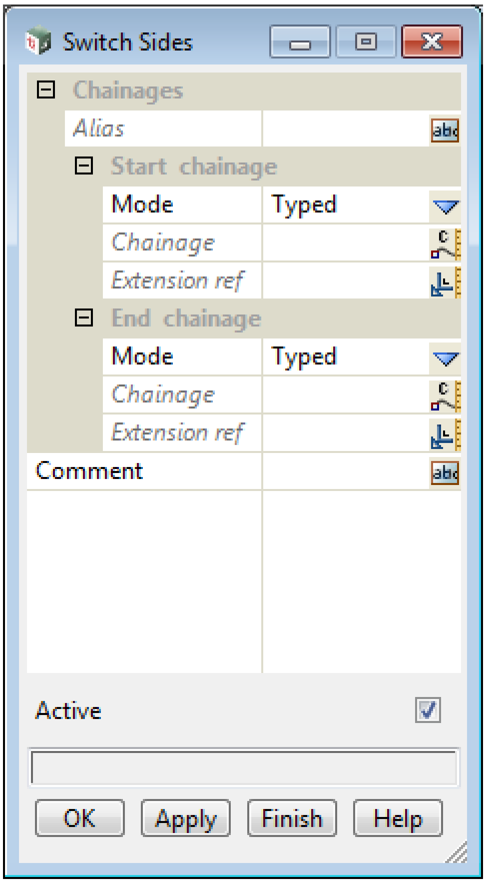
The **Switch Sides** command transfers processing to the other side of the **Modifiers**.

That is, if a **Switch Sides** command is processed in a **Left MTF Modifiers** grid, then processing stops at that point in the **Left MTF Modifiers** grid, and processing is transferred to the **Right MTF Modifiers** grid. In the **Right MTF Modifiers** grid it from the row where the last **Switch Sides** command was executed **in the Right MTF Modifiers** grid, or if that hasn't occurred before, processing starts at the top row of the **Right MTF Modifiers** grid.

And similarly if a **Switch Sides** command is processed in a **Right MTF Modifiers** grid

So processing can switch a number of times between the two sides.

Selecting **Switch sides** brings up the **Switch Sides** panel.



Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages where the Switch Sides command is.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

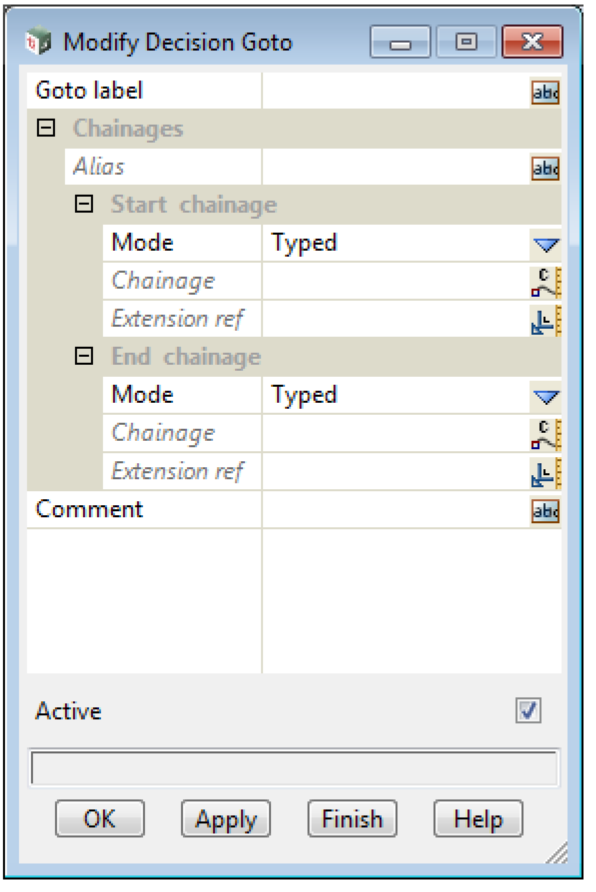
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Go to the next section 21.2.2.2.8.2 Fixed Decisions - Goto or return to 21.2.2.2.8 Fixed Link - Decisions.

21.2.2.2.8.2 Fixed Decisions - Goto

Goto will transfer control to the row of the **Left/Right MTF Modifiers** grid with the label **Goto label.**

Selecting **Goto** brings up the panel **Modify Decision Goto**



Goto label text box

transfer processing to the row of the Left/Right MTF Modifiers grid with this label.

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages where the GoTo command exists.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

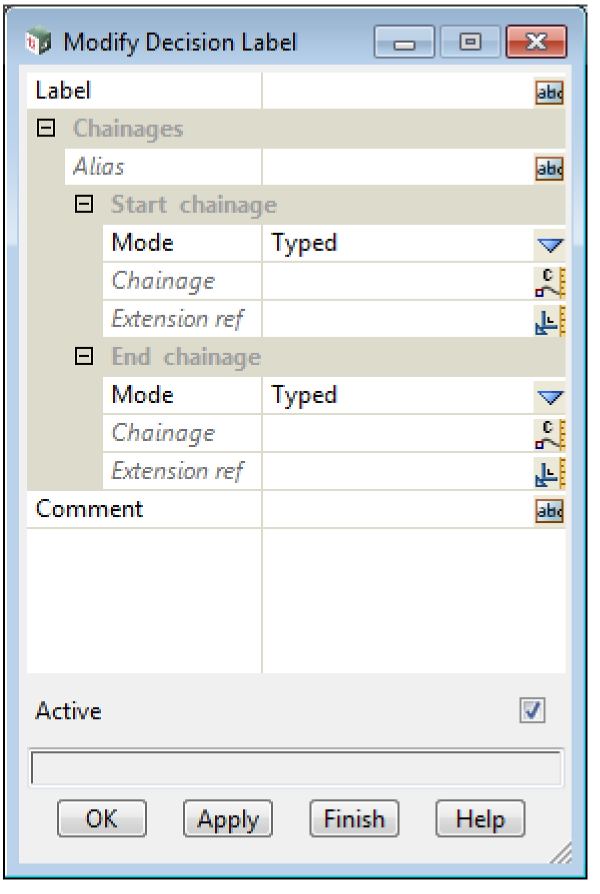
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Go to the next section 21.2.2.2.8.3 Fixed Decisions - Label or return to 21.2.2.2.8 Fixed Link - Decisions.

21.2.2.2.8.3 Fixed Decisions - Label

The **Label** option creates a row in the **Left/Right MTF Modifiers** grid with the label **Label.**

Selecting **Label** brings up the panel **Modify Decision Label**



Label text box

name of the label that goes with this row of the Left/Right MTF Modifiers grid.

Alias, **Start Chainage**, **End Chainage**, **Interval**

the Label exists when the chainage is between the Start and End chainages.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Go to the next section 21.2.2.2.8.4 Fixed Decisions - Then or return to 21.2.2.2.8 Fixed Link - Decisions.

21.2.2.2.8.4 Fixed Decisions - Then

The **Then** Command works in conjunction with an **End Then** command, and they both work with any one of the ***Decision*** commands that has a *test* and a **Goto Label** that is executed when the test succeeds (.e.g Batter Decision, String Exist, Tin).

The **Then** and the **End Then** must have the value for the **Label** field to be the same as the **GoTo Label** in a Test command.

If such an **Then-End Then** block exists for a Test command, then whenever the **GoTo Label** is executed for the Test command then processing is transferred to the **Then** command and continues from there.

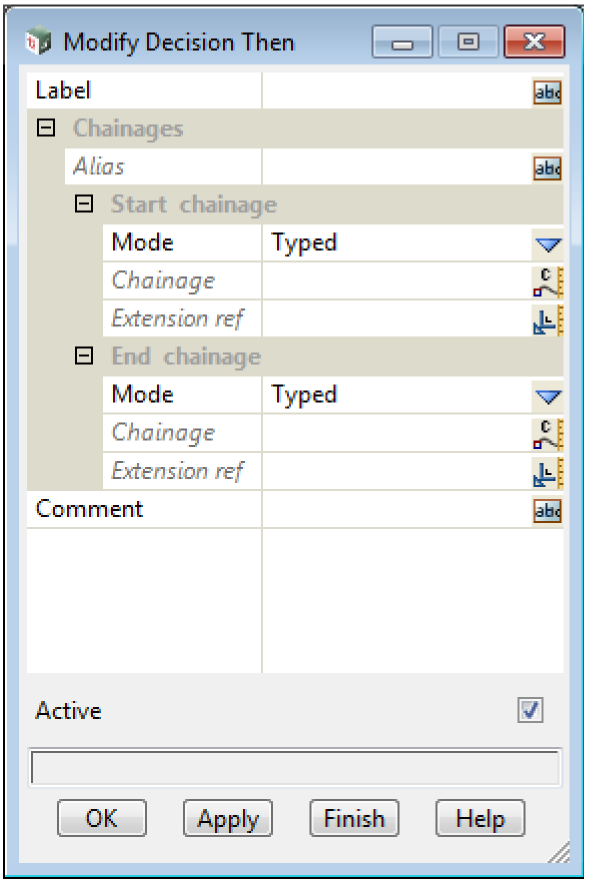
However if processing reaches a **Then** command by any mean other than from a Test command, then all the commands are ignored until the **End Then** command is reached.

So a **Then-End Then** block is ignored except by a Test command with a matching **GoTo Label**.

Notes

1. More than one Test command can use the same **Then-End Then** block.
2. Jumping into the middle of a **Then-End Then** block should not be done and will have unpredictable consequences.

Selecting **Then** brings up the **Modify Decision Then** panel



Label text box

*name of the label that is the same as its matching **End Then** command, and matches the* **Goto Label** *of any Test command that wants to use the **Then-End Then** block.*

Alias, **Start Chainage**, **End Chainge**, **Interval**

the Then exists when the chainage is between the Start and End chainages.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

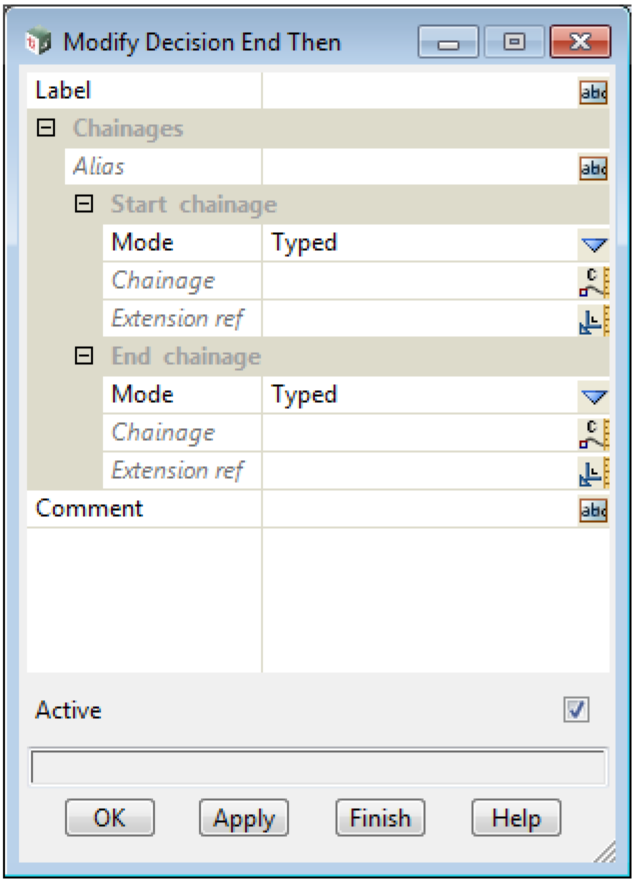
Go to the next section 21.2.2.2.8.5 Fixed Decisions - End Then or return to 21.2.2.2.8 Fixed Link - Decisions.

21.2.2.2.8.5 Fixed Decisions - End Then

MTF_Edit_Modify_Decision_End_Then

The **End Then** goes with a matching **Then** command. For a description of how the **Then-End Then** commands are used, see 21.2.2.2.8.4 Fixed Decisions - Then.

Selecting **End then** brings up the panel **Modify Decision End Then**



Label text box

*name of the label that is the same as its matching **Then** command, and matches the* **Goto Label** *of any Test command that wants to use the **Then-End Then** block.*

Alias, **Start Chainage**, **End Chainge**, **Interval**

the End Then exists when the chainage is between the Start and End chainages.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Go to the next section 21.2.2.2.8.6 Fixed Decisions - Batter or return to 21.2.2.2.8 Fixed Link - Decisions.

21.2.2.2.8.6 Fixed Decisions - Batter

The **Batter** option is used to **construct** a **new link** which stops if it comes within

(a) a **Strip** depth of a **Tin**

(b) an **Offset** distance from a **Tin**.

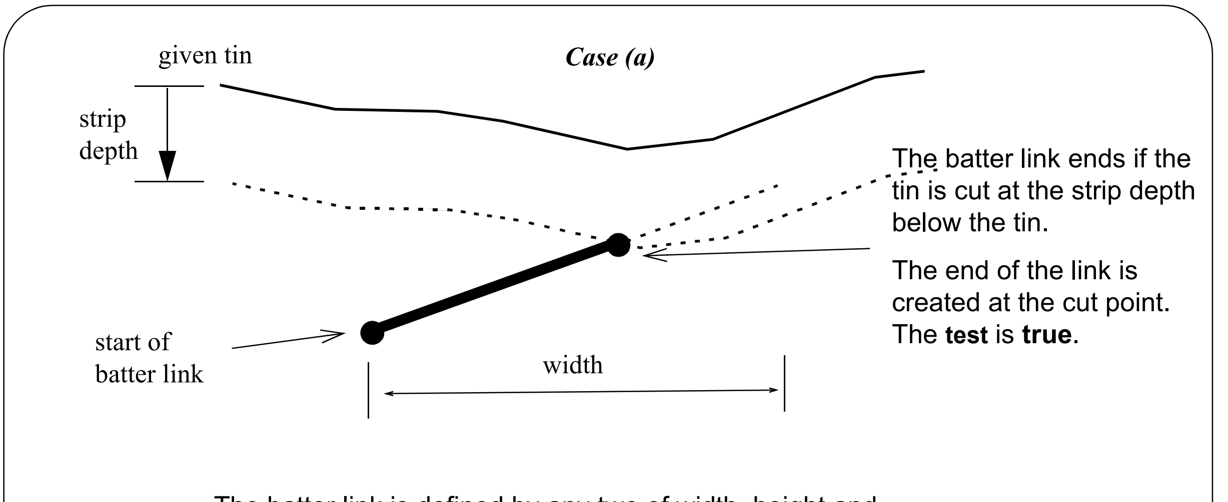
(c) a **Strip** depth of a **Tin** calculated at a given **Offset** from the link.

If the link does stop, control is transferred via a **Goto label**, otherwise control continues onto the next row in the **Left/Right MTF Modifiers** grid.

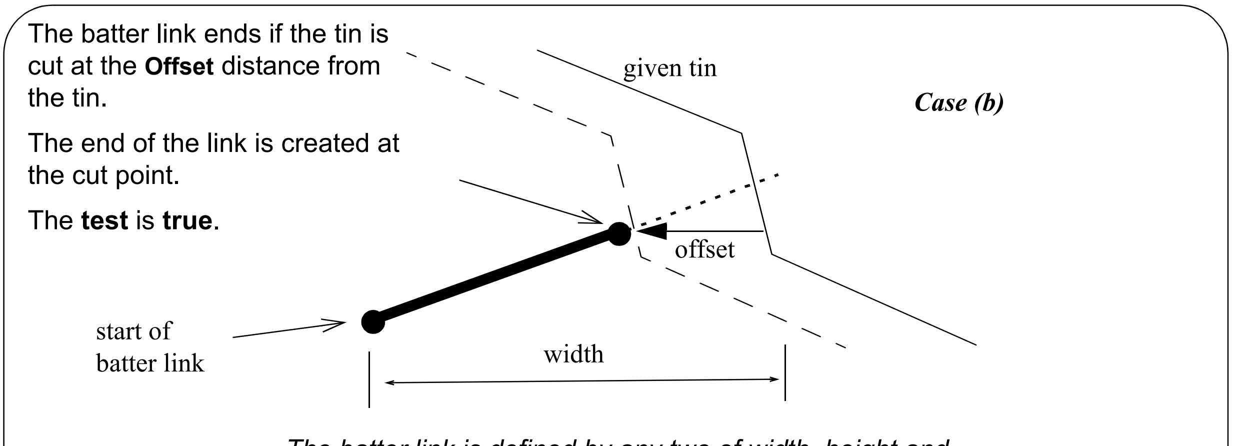
The maximum length of the **batter** link is defined by specifying values for two of the three fields **Width**, **Height** and **Slope**. The **Width**, **Height** and **Slope** values of the created link may be different if the link stops because of one of the conditions (a), (b) or (c).

Notes

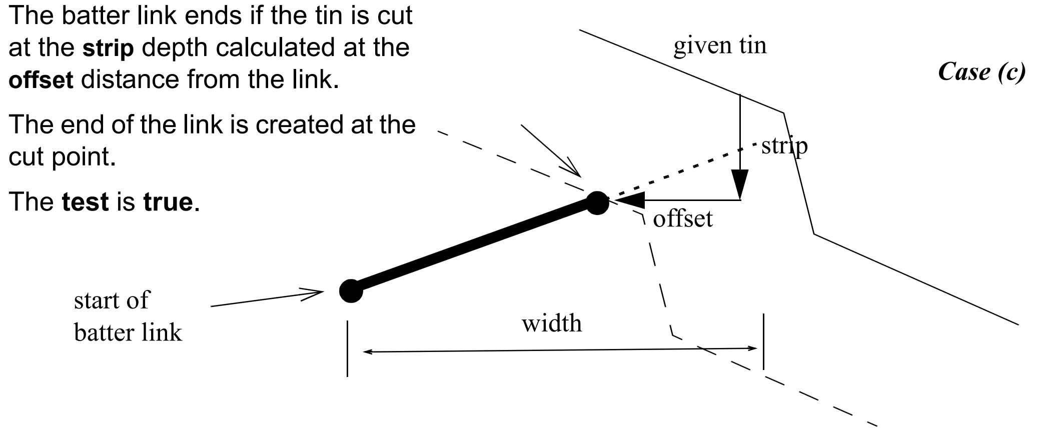
1. cases (a) and (b) are just special cases of (c)
2. strip and offset can be used to stop the link to allow for a fixed structure (such as a drain) to be inserted so that it ends up on the tin.



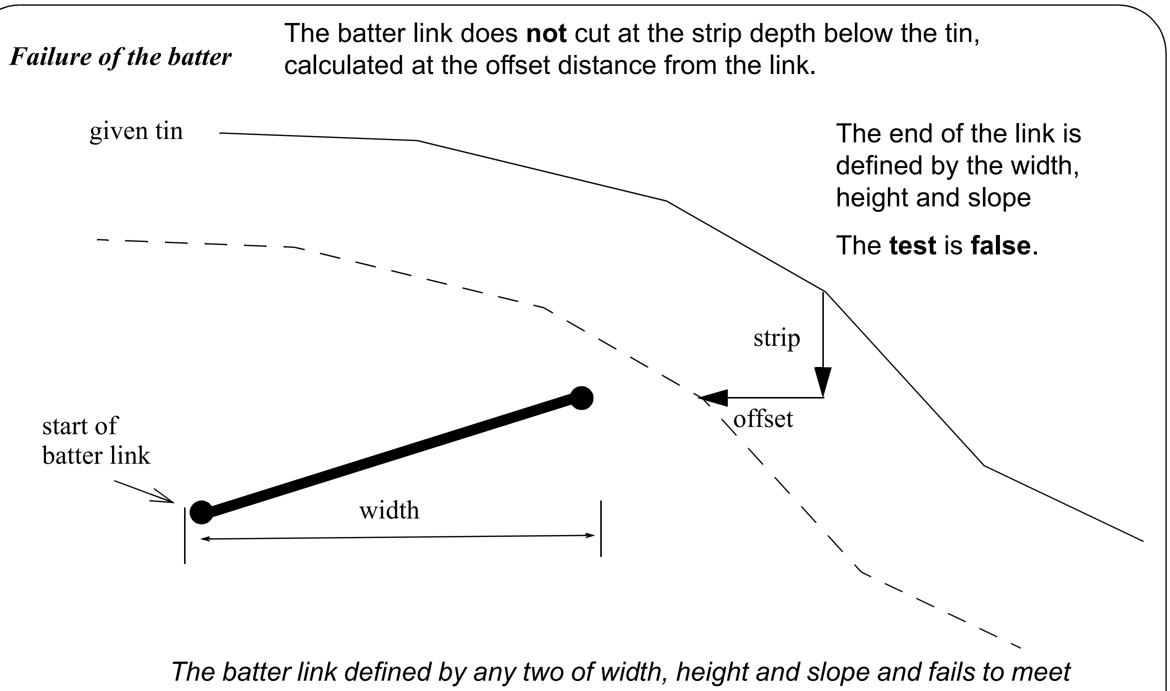
The batter link is defined by any two of width, height and slope and stops if it meets the tin at the strip depth.



The batter link is defined by any two of width, height and slope and stops if it meets the tin at the offset distance.

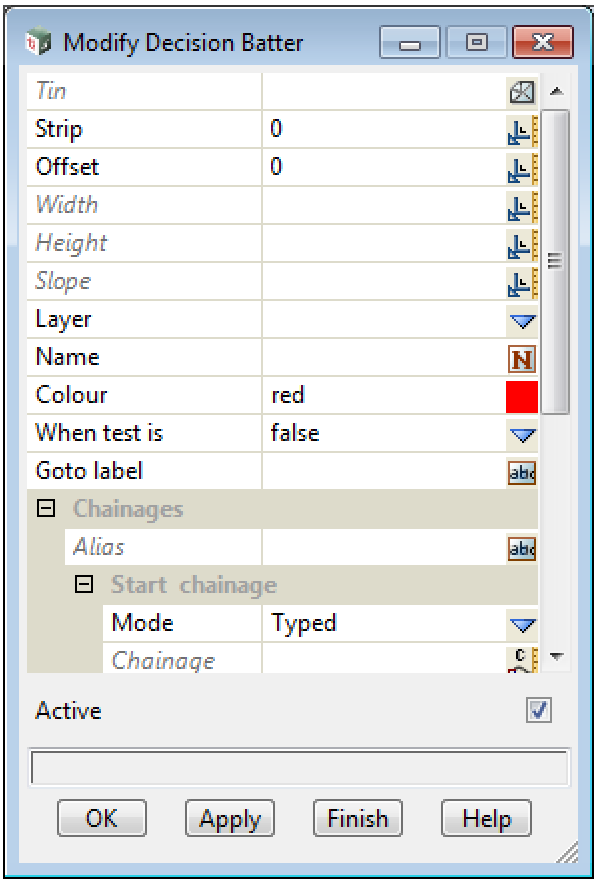


The batter link is defined by any two of width, height and slope and stops if it meets the tin at the strip depth calculated at the given offset distance from the link.



The batter link defined by any two of width, height and slope and fails to meet the tin at the strip depth calculated at the given offset distance from the link.

Selecting **Batter** brings up the **Modify Decision Batter** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
Tin	tin box		available tins

tin to batter to.

Strip	real box	0	

distance below the tin to stop at.

Offset	real box	0	

offset distance from the link to check strip depth

Width	real box		

the width for the link. Any two of Width, Height and Slope can be used.

Height	real box		

the height for the link. Any two of Width, Height and Slope can be used.

Slope	real box		

the slope, in 1v in, of the link. Positive is up and negative down.
Any two of Width, Height and Slope can be used.

Layer	layer box	Design	available layers

name of the layer that the new link is to go in. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Name	name box		available names

the name to be used for the created link.

Colour	colour box	cyan	available colours

the colour to be used for the created link

When test is choice box false true, false

*if **When test is** has the value **true**, and the test is **true** (i.e. the batter link comes within the depth* **Strip** *and offset* **Offset** *of the tin) then go to the label* **Goto label**. *Otherwise proceed to the next row.*

*If **When test is** has the value **false**, and the test is **false** then go to the label* **Goto label**. *Otherwise proceed to the next row.*

Goto label text box

used with **When test is**.

*Note that there may be a **Then-End Then** block that matches the* **GoTo Label**. *For a description of a **Then-End Then** block and how it words, see* 21.2.2.2.8.4 Fixed Decisions - Then.

Alias, **Start Chainage**, **End Chainge**, **Interval**

the batter is performed when the chainage is between the Start and End chainages.
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Go to the next section 21.2.2.2.8.7 Fixed Decisions - Batter Test or return to 21.2.2.2.8 Fixed Link - Decisions.

21.2.2.2.8.7 Fixed Decisions - Batter Test

The **Batter test** option ***tests*** if a user defined link comes within

(a) a **Strip** depth of a **Tin**

(b) an **Offset** distance from a **Tin**.

(c) a **Strip** depth of a **Tin** calculated at a given **Offset** from the link.

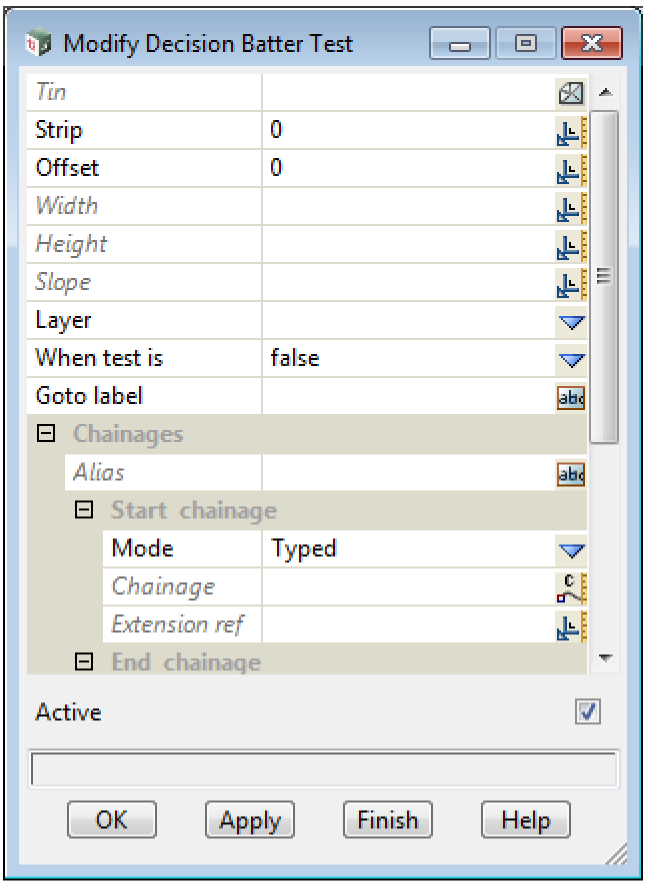
The **test link** is defined by specifying values for two of the three fields **Width**, **Height** and **Slope**.

Notes

1. The tests are the same as in 21.2.2.2.8.6 Fixed Decisions - Batter but **no link is created**. That is, The batter test is used to test if a batter will stop/not stop without creating a batter link.
2. Cases (a) and (b) are just special cases of (c)

For diagrams describing the tests, see 21.2.2.2.8.6 Fixed Decisions - Batter.

Selecting **Batter test** brings up the **Modify Decision Batter Test** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
Tin	tin box		available tins
tin to test battering to.			
Strip	real box	0	
distance below the tin to stop at.			
Offset	real box	0	
offset distance from the link to check strip depth			

Width real box

the width for the link. Any two of Width, Height and Slope can be used.

Height real box

the height for the link. Any two of Width, Height and Slope can be used.

Slope real box

the slope, in 1v in, of the link. Positive is up and negative down.
Any two of Width, Height and Slope can be used.

Layer layer box Design available layers

name of the layer that the test link is appended to. For information on Layers, see 21.1.1.2 MTF Links and Layers.

When test is choice box false true, false

*if **When test is** has the value **true**, and the test is **true** (i.e. the test link comes within the depth* **Strip** *and offset* **Offset** *of the tin) then go to the label* **Goto label**. *Otherwise proceed to the next row.*

*If **When test is** has the value **false**, and the test is **false** then go to the label* **Goto label**. *Otherwise proceed to the next row.*

Goto label text box

used with **When test is**.

*Note that there may be a **Then-End Then** block that matches the* **GoTo Label**. *For a description of a **Then-End Then** block and how it words, see* 21.2.2.2.8.4 Fixed Decisions - Then.

Alias, **Start Chainage**, **End Chainge**, **Interval**

the batter is tested when the chainage is between the Start and End chainages.
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

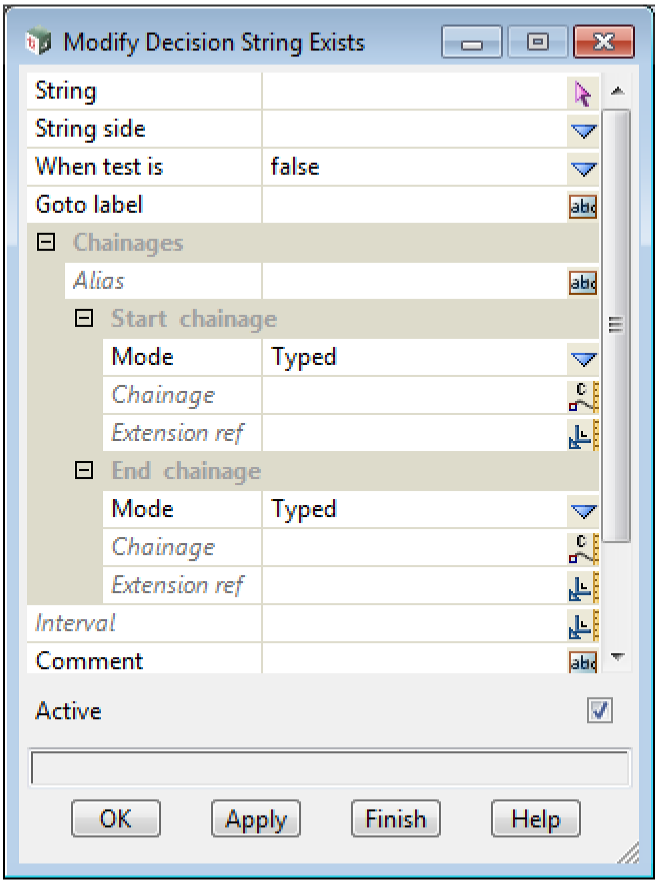
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Go to the next section 21.2.2.2.8.8 Fixed Decisions - String Exists or return to 21.2.2.2.8 Fixed Link - Decisions.

21.2.2.2.8.8 Fixed Decisions - String Exists

This options tests to see if a string is on the left and/or right of the reference string.

Selecting **String** brings up the panel **Modify Decision String Exists**



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
String	string select		

the string to test.

String side	choice box	both	left, right, both

*if **left**, only the left side of the reference string is searched for the given string.*

*If **right**, only the right side of the reference string is searched for the given string.*

*If **both**, the left and the right side of the reference string is searched for the given string.*

When test is	choice box	false	true, false

*if **When test is** has the value **true**, and the test is **true** (i.e. the given string is found) then go to the label* **Goto label**. *Otherwise proceed to the next row.*

*If **When test is** has the value **false**, and the test is **false** (i.e. the given string is not found) then go to the label* **Goto label**. *Otherwise proceed to the next row.*

Goto label	text box		

used with **When test is**.

*Note that there may be a **Then-End Then** block that matches the* **GoTo Label**. *For a description of a **Then-End Then** block and how it words, see* 21.2.2.2.8.4 Fixed Decisions - Then.

Alias, **Start Chainage**, **End Chainge**, **Interval**

the test is performed when the chainage is between the Start and End chainages.
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

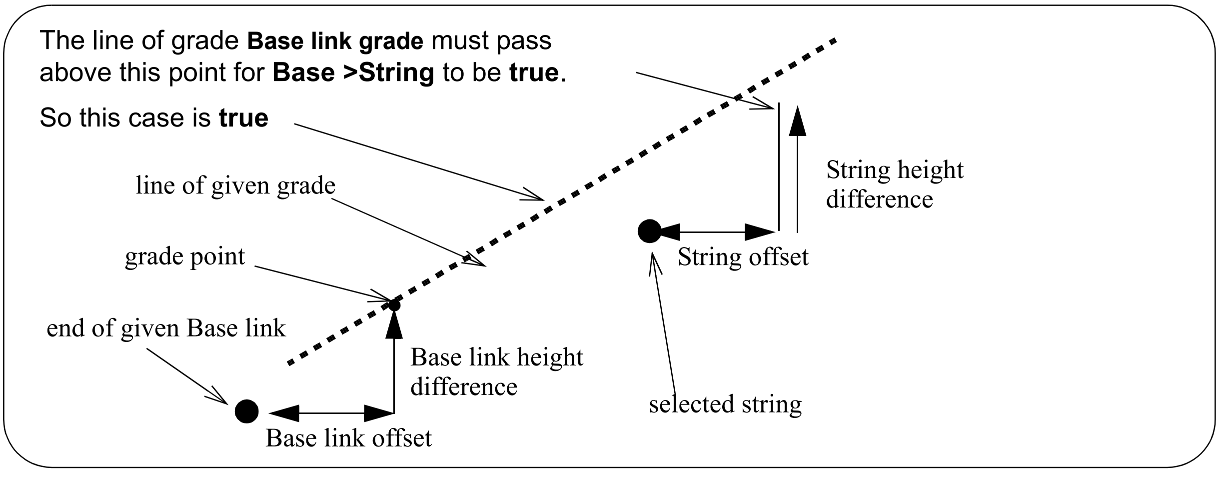
Go to the next section 21.2.2.2.8.9 Fixed Decisions - Above String Height or return to 21.2.2.2.8 Fixed Link - Decisions.

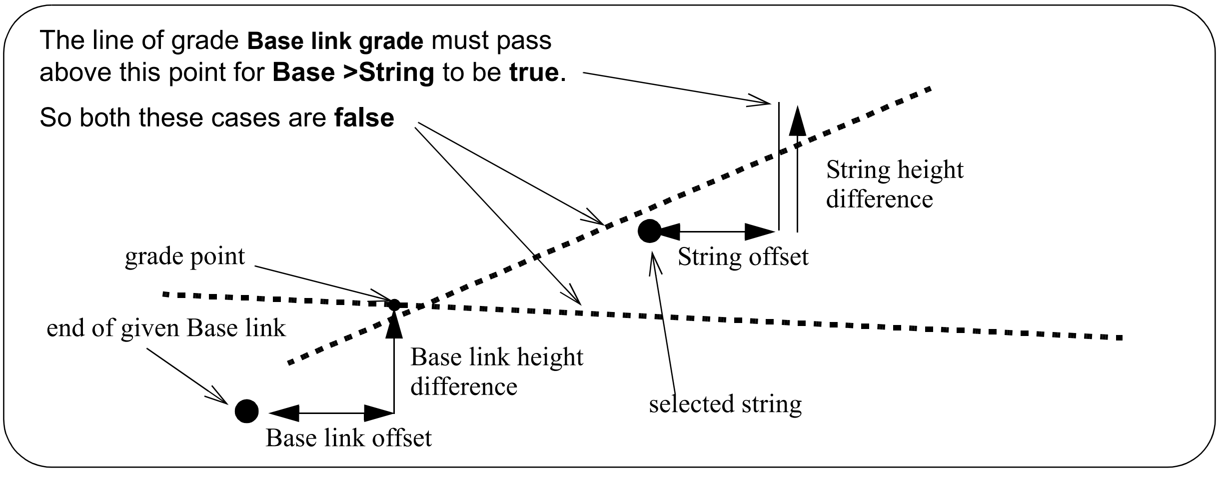
21.2.2.2.8.9 Fixed Decisions - Above String Height

The option ***tests*** if a line of a given grade (**grade line)** defined from the end of a Base link passes a given height difference above a selected string.

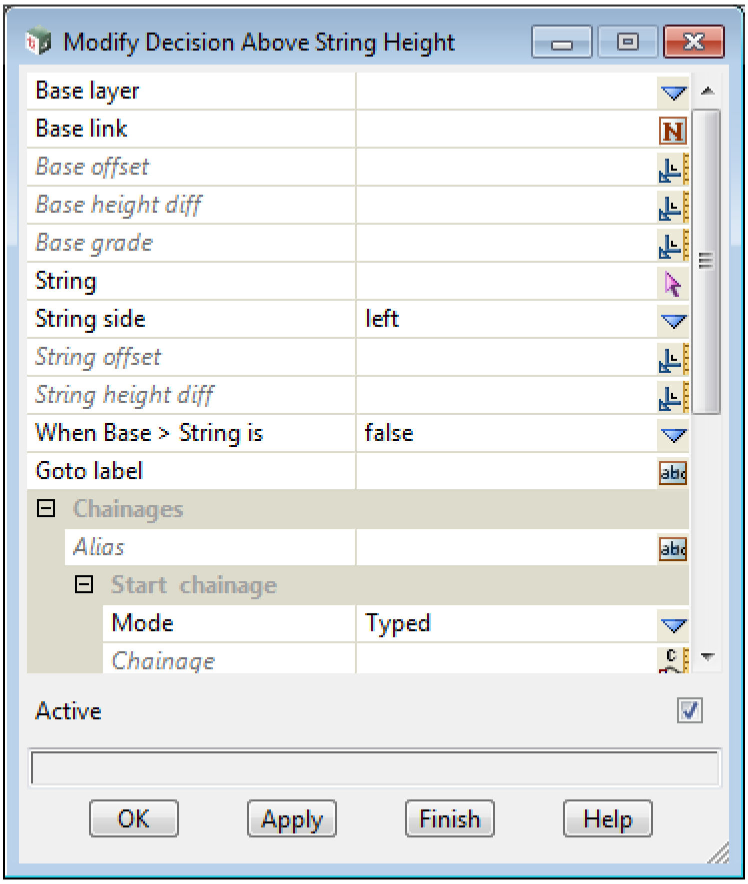
The **grade line** passes through the **grade point** which is a given **Base link offset** and **Base link height difference** from the end of the given link.

The height is tested at a given **string offset** from the selected string.





Selecting **Above string height** brings up the **Modify Decision Above String Height** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
Base layer	layer box	Design	available layers

name of the layer that the link to test for is in. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Base link — link box

name of the link in **Base layer** *to test if the line of grade going from the link is a delta height above a selected string.*

Base link offset — real box

option "sided" offset distance from the end of the link **Base link** *that the line of grade* **Base link grade** *goes through (the grade point).*

Base link height diff — real box

the value to add to the height of the end of the link **Base link** *that the line of grade* **Base link grade** *goes through (the grade point).*

Base link grade — real box

percentage grade of a line through the grade point that is tested to see if it passed withing a given offset and height difference of the selected string.

String — string select

the string to test.

Field Description	Type	Default	Pop-Up
String side	choice box	both	left, right, both

if ***left****, only the left side of the reference string is searched for the selected string.*

If ***right****, only the right side of the reference string is searched for the selected string.*

*If **both**, the left and the right side of the reference string is searched for the selected string.*

String offset real box

optional "absolute" offset distance from the string of the point that is tested for the **grade link** *going above.*

String height diff real box

optional height difference added to the height of the string. This total height is used at the String offset point to give the point that is tested for the **grade link** *going above.*

When Base >String is choice box false true, false

*if **When Base >String** has the value **true**, and the test is **true** then go to the label* **Goto label**. *Otherwise proceed to the next row.*

*If **When Base > String** has the value **false**, and the test is **false** then go to the label* **Goto label**. *Otherwise proceed to the next row.*

Goto label text box

used with **When Base > String is**.

*Note that there may be a **Then-End Then** block that matches the* **GoTo Label**. *For a description of a **Then-End Then** block and how it words, see* 21.2.2.2.8.4 Fixed Decisions - Then.

Alias, **Start Chainage**, **End Chainge**, **Interval**

the test is performed when the chainage is between the Start and End chainages.
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

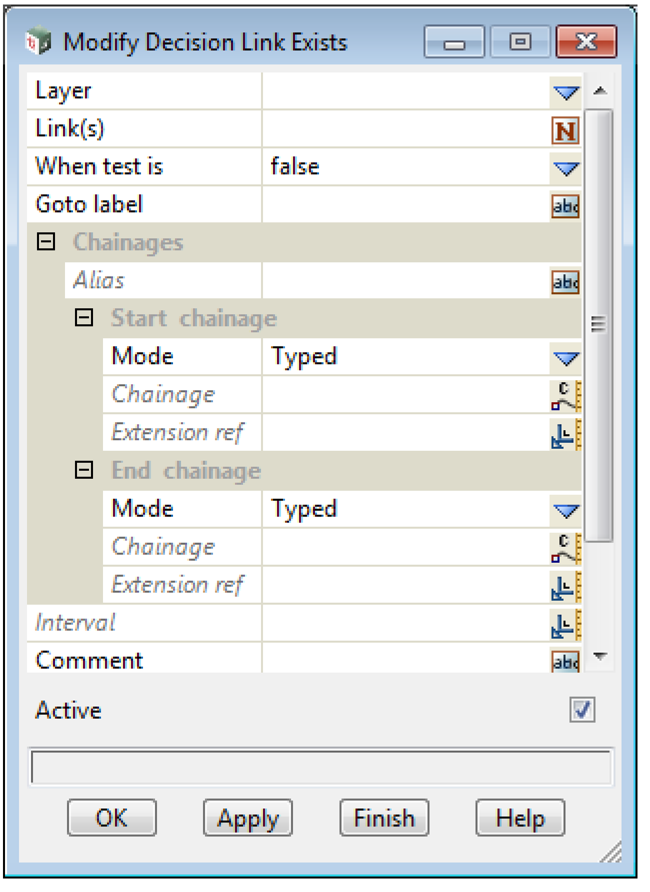
Go to the next section 21.2.2.2.8.10 Fixed Decisions - Link Exists or return to 21.2.2.2.8 Fixed Link - Decisions.

21.2.2.2.8.10 Fixed Decisions - Link Exists

This options tests to see if a given link exists.

Note that a link is only looked for on the side that the Link Exists command being processed is in.

Selecting **Link** brings up the panel **Modify Decision Link Exists**



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
Layer	layer box	Design	available layers

name of the layer that the link to test for is in. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Link(s)	name box		

name of the links in **Layer** *to test to see if any of them exist. See* 21.2.2.1.2 Link or Link(s).
The test only looks at the side that the Link Exists command being processed is in (Left or Right side).

When test is	choice box	false	true, false

*if **When test is** has the value **true**, and the test is **true** (i.e. the given link is found) then go to the label* **Goto label**. *Otherwise proceed to the next row.*

*If **When test is** has the value **false**, and the test is **false** (i.e. the given link is **not** found) then go to the label* **Goto label**. *Otherwise proceed to the next row.*

Goto label	text box		

used with **When test is**.

*Note that there may be a **Then-End Then** block that matches the* **GoTo Label**. *For a description of a **Then-End Then** block and how it words, see* 21.2.2.2.8.4 Fixed Decisions - Then.

Alias, **Start Chainage**, **End Chainge**, **Interval**

the test is performed when the chainage is between the Start and End chainages.
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

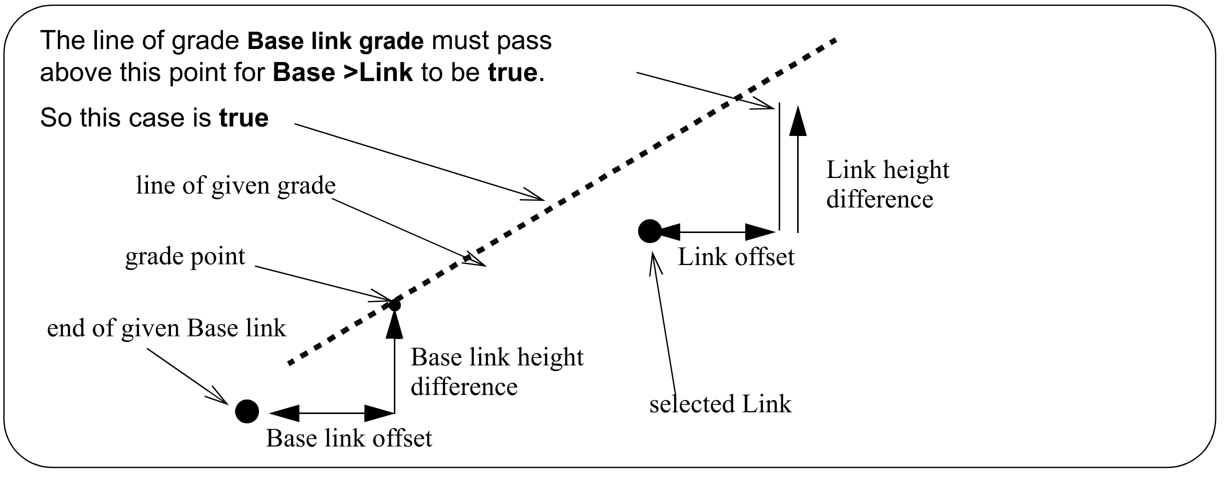
Go to the next section 21.2.2.2.8.12 Fixed Decisions - Tin or return to 21.2.2.2.8 Fixed Link - Decisions.

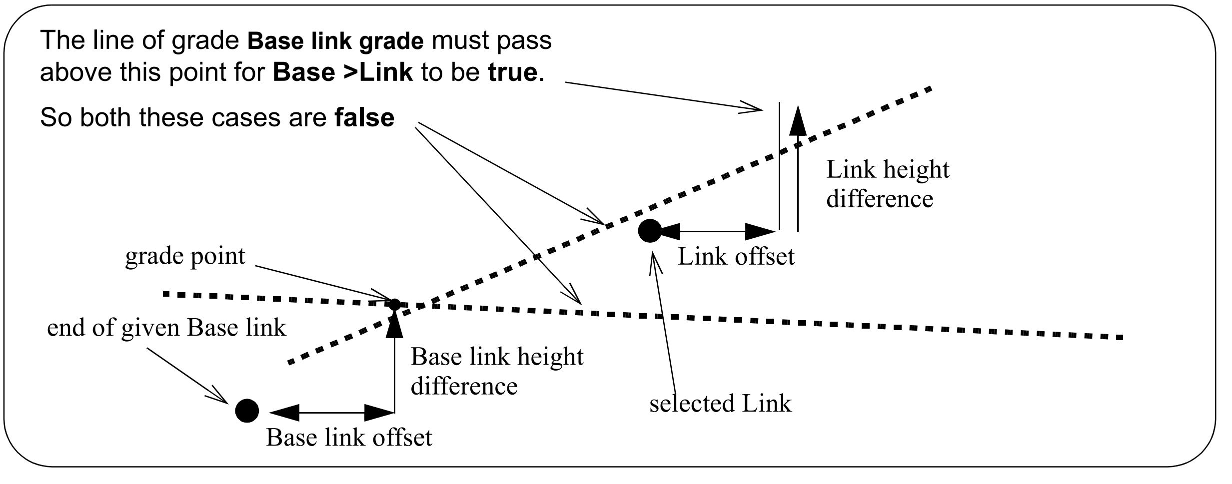
21.2.2.2.8.11 Fixed Decisions - Above Link Height

The option ***tests*** if a line of a given grade (**grade line)** defined from the end of a **Base link** passes a given height difference above a selected **Link**.

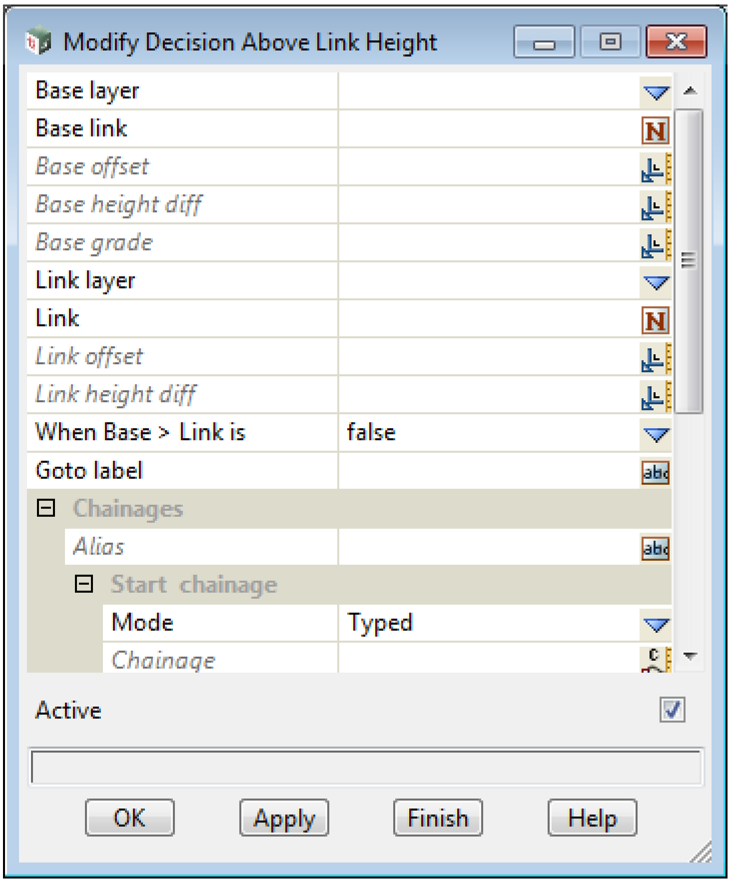
The **grade line** passes through the **grade point** which is a given **Base link offset** and **Base link height difference** from the end of the given link.

The height is tested at a given **Link offset** from the selected **Link**.





Selecting **Above link height** brings up the **Modify Decision Above Link Height** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
Base layer	layer box	Design	available layers

name of the layer that the base link to test for is in. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Base link	link box		

name of the base link in **Base layer** *to test if the line of grade going from the base link is a delta height above a selected* **Link**.

Base link offset	real box		

option "sided" offset distance from the end of the link **Base link** *that the line of grade* **Base link grade** *goes through (the grade point).*

Base link height diff	real box		

the value to add to the height of the end of the link **Base link** *that the line of grade* **Base link grade** *goes through (the grade point).*

Base link grade	real box		

percentage grade of a line through the grade point that is tested to see if it passed above a given offset and height difference of a selected **Link**.

Link layer	layer box	Design	available layers

name of the layer that the link to test that the grade line is above, is in. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Link	link box		

name of the link in **Layer** *to test if the line of grade going from the base link is a delta height above it.*

Link offset real box

optional "absolute" offset distance from the end of **Link** *of the point that is tested for the* **grade link** *going above.*

Link height diff real box

optional height difference added to the height of the end of **Link**. *This total height is used at the* **Link** *offset point to give the point that is tested for the* **grade link** *going above.*

When Base >String is choice box false true, false

if ***When Base >Link*** *has the value* ***true****, and the test is* ***true*** *then go to the label* **Goto label**. *Otherwise proceed to the next row.*

If ***When Base > Link*** *has the value* ***false****, and the test is* ***false*** *then go to the label* **Goto label**. *Otherwise proceed to the next row.*

Goto label text box

used with **When Base > Link is**.

Note that there may be a ***Then-End Then*** *block that matches the* **GoTo Label**. *For a description of a* ***Then-End Then*** *block and how it words, see* 21.2.2.2.8.4 Fixed Decisions - Then.

Alias, **Start Chainage**, **End Chainge**, **Interval**

the test is performed when the chainage is between the Start and End chainages.
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels**.**

Go to the next section 21.2.2.2.8.12 Fixed Decisions - Tin or return to 21.2.2.2.8 Fixed Link - Decisions.

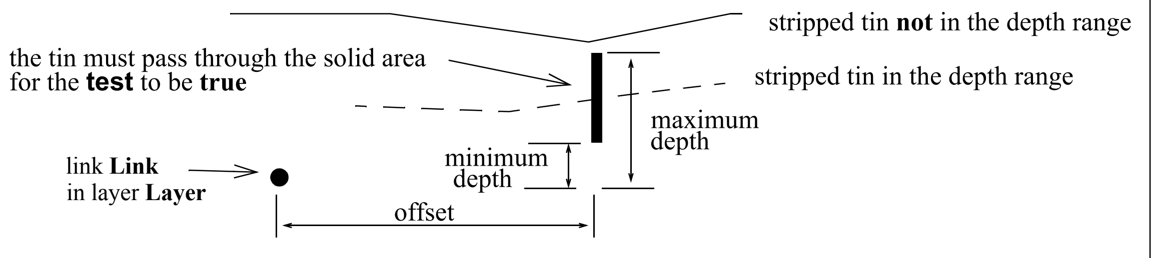
21.2.2.2.8.12 Fixed Decisions - Tin

Selecting **Tin** brings up the **Modify Decision Tin** panel which tests to see if the depth from the end of the given **Link** in a given **Layer**, offset by the amount given in the **Offset** field, is between the two values given in the **Minimum depth** and **Maximum depth** fields where the depths are taken from the stripped Tin (that is, the **Tin** dropped by the **Strip** value).

If the depth is between the values, then control is transferred to the row of the ***Modifiers*** grid with the label given by the **Goto label** field. Otherwise, control passes to the next row of the ***Modifiers*** grid.

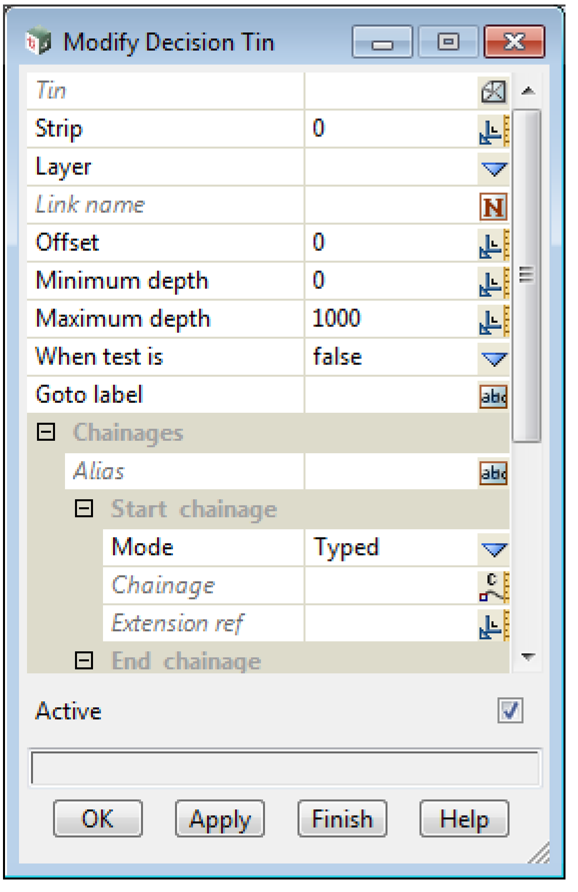
Offset, **Minimum depth** and **Maximum depth** can be positive or negative.

This is a test and does not create a link.



tin decision given by offset and minimum and maximum depths.

Selecting **Tin** brings up the panel **Modify Decision Tin**



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
Tin	tin box		available tins

the tin to calculate the depth to.

Strip	real box	0	

the depth to drop the tin before testing.

Layer	layer box	Design	available layers

name of the layer that the link to test is in. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Link name	input		

name o f link in **Layer** *to test against the tin.*

Offset	real box	0	

the depth is calculate at an offset distance of **offset** *from the end of the previous link.*

Minimum depth	real box	0	

if the depth is between the **Minimum depth** *and the* **Maximum depth**, *then the test is* ***true***, *otherwise the test is* ***false***.

Maximum depth	real box	1000	

see the description of the **Minimum depth** *field.*

When test is	choice box	false	true, false

if ***When test is*** *has the value* ***true***, *and the test is* ***true*** *(i.e. the depth is between the minimum and maximum depths) then go to the label* **Goto label**. *Otherwise proceed to the next row.*

*If **When test is** has the value **false**, and the test is **false** (i.e. the depth is **not** between the minimum and maximum depths) then go to the label* **Goto label**. *Otherwise proceed to the next row.*

Goto label text box

used with **When test is**.

*Note that there may be a **Then-End Then** block that matches the* **GoTo Label**. *For a description of a **Then-End Then** block and how it words, see* 21.2.2.2.8.4 Fixed Decisions - Then.

Alias, **Start Chainage**, **End Chainge**, **Interval**

the test is performed when the chainage is between the Start and End chainages.
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

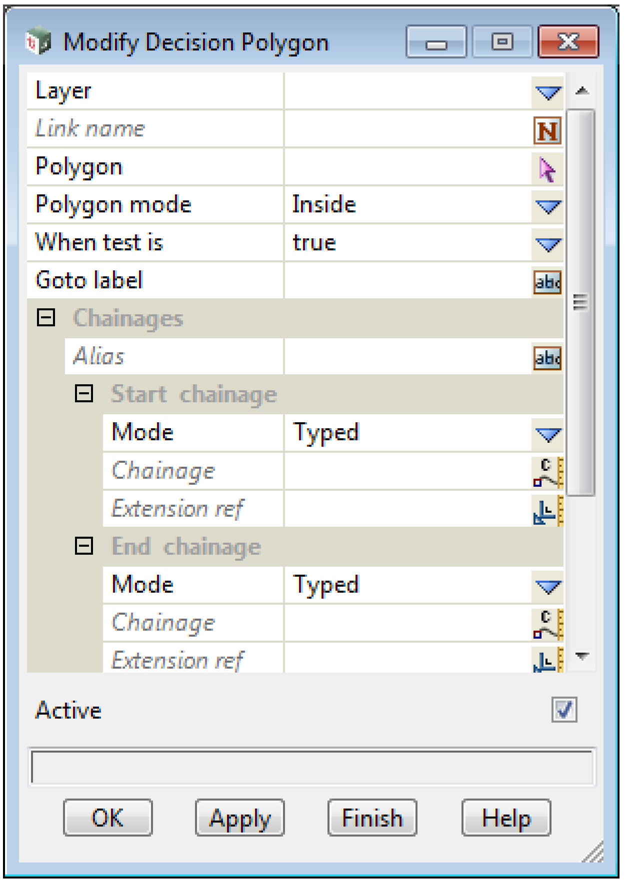
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Go to the next section 21.2.2.2.8.13 Fixed Decisions - Polygon or return to 21.2.2.2.8 Fixed Link - Decisions.

21.2.2.2.8.13 Fixed Decisions - Polygon

This options tests to see if the (x,y) coordinates of a link is inside a user selected polygon.

Selecting **Polygon** brings up the panel **Modify Decision Polygon**



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
Layer	layer box	Design	available layers

name of the layer that the link to test is in. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Link name	input		

name o f link in **Layer** *to test against the polygon.*

Polygon	string sect		

the polygon to.use for testing if the link is inside or outside of it.

Polygon mode	choice box	Inside	Inside, Outside

*if **Inside**,.the test is true if the (x,y) coordinates of the link is inside the polygon.*

*If **Outside**, the test is true if the (x,y) coordinates of the link is outside the polygon*

When test is	choice box	false	true, false

*if **When test is** has the value **true**, and the test is **true** (i.e. the end of the given link is inside the polygon) then go to the label* **Goto label**. *Otherwise proceed to the next row.*

*If **When test is** has the value **false**, and the test is **false** (i.e. the end of the given link is **not** inside the polygon) then go to the label* **Goto label**. *Otherwise proceed to the next row.*

Goto label text box

used with **When test is**.

Note that there may be a ***Then-End Then*** *block that matches the* **GoTo Label**. *For a description of a* ***Then-End Then*** *block and how it words, see* 21.2.2.2.8.4 Fixed Decisions - Then.

Alias, **Start Chainage**, **End Chainge**, **Interval**

the test is performed when the chainage is between the Start and End chainages.
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Return to 21.2.2.2.8 Fixed Link - Decisions.

21.2.2.2.9 Fixed Link - From link

For a given link (the current link), the **from Link** option takes **one** of Width, Height or Xfall **from a selected link** whilst holding one to be what it is defined as by the current link.

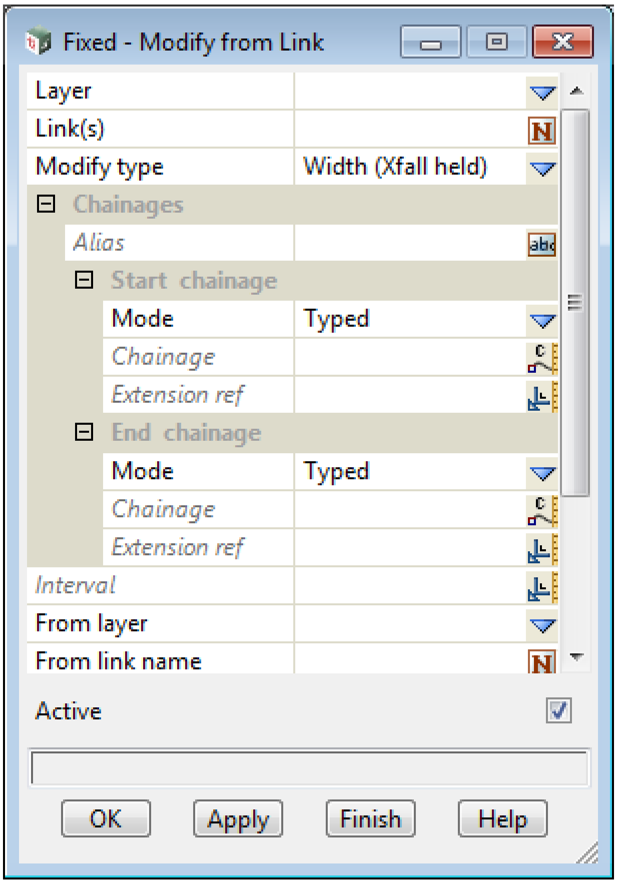
That is, for the current link:

(a) one of its existing Width, Height or Xfall **is modified** by taking its value **from a selected link**

and

(b) a different one of Width, Height or Xfall **keeps its existing value** (i.e. holds its existing value).

Selecting the **From link** brings up the **Fixed - Modify from Link** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box	Design	available layers

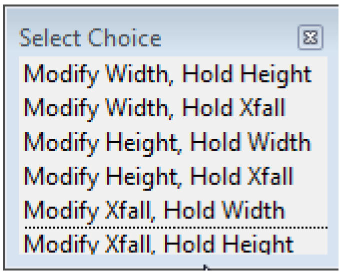
name of the layer that the link to be modified is in. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Link(s)	name box		names.4d pop-up

the name of the links in the Layer to modify the width/height/xfall/slope using the method given in **Modifier type,** *between the chainages given by* **Start mode** *and* **End mode.** *See* 21.2.2.1.2 Link or Link(s).

Each of the links is processed as the current link.

Modifier type choice box



modify Width to take width from selected link, hold Height to be its current value
modify Width to take width from selected link, hold Xfall to be its current value
modify Height to take height from selected link, hold Width to be its current value
modify Height to take height from selected link, hold Xfall to be its current value
modify Xfall to take xfall from selected link, hold Width to be its current value
modify Xfall to take xfall from selected link, hold Height to be its current value

A link can be modified in any way regardless of how the link was defined in terms of width, height and xfall or slope. See 21.1.2.1 Calculating Width, Height, Xfall or Slope from Original Definitions

*The **Modify Width/Height/Xfall** is the part of the current link that is modified and the new values are taken from a selected link.*

*The **Hold Width/Height/Xfall** is the part of the current link that is kept (held) at its current value.*

So for **Modifier type:**

***Modify Width**, **Hold Height** - the **Width** is taken from the link* **From link name** *that is in the layer* **From layer** *and zone* **From zone**, *and the **Height** is kept as it is.*

***Modify Width**, **Hold Xfall** - the **Width** is taken from the link* **From link name** *that is in the layer* **From layer** *and zone* **From zone**, *and the **Xfall** is kept as it is.*

***Modify Height**, **Hold Width -** the **Height** is taken from the link* **From link name** *that is in the layer* **From layer** *and zone* **From zone**, *and the **Width** is kept as it is.*

***Modify Height**, **Hold Xfall -** the **Height** is taken from the link* **From link name** *that is in the layer* **From layer** *and zone* **From zone**, *and the **Xfall** is kept as it is.*

***Modify Xfall**, **Hold Width -** the **Xfall** is taken from the link* **From link name** *that is in the layer* **From layer** *and zone* **From zone**, *and the **Width** is kept as it is.*

***Modify Xfall**, **Hold Height -** the **Xfall** is taken from the link* **From link name** *that is in the layer* **From layer** *and zone* **From zone**, *and the **Height** is kept as it is.*

From layer layer box Design available layers

name of the layer of the link to take values from.

From Link name name box names.4d pop-up

the name of the link in the **From Laye***r and* **From Zone** *to get the Width, Height or Xfall from.*

From zone choice box fixed, cut, fill

*the fixed, cut or fill zone that the **From link name** link is in.*

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages for the link to be modified.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Continue to the next section 21.2.2.2.10 Fixed Link - To String or return to 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

21.2.2.2.10 Fixed Link - To String

For a given link, (the current link), the **to string** option takes **one** of Width, Height or Xfall as the value you get by going from the start point of the current link to another selected string.

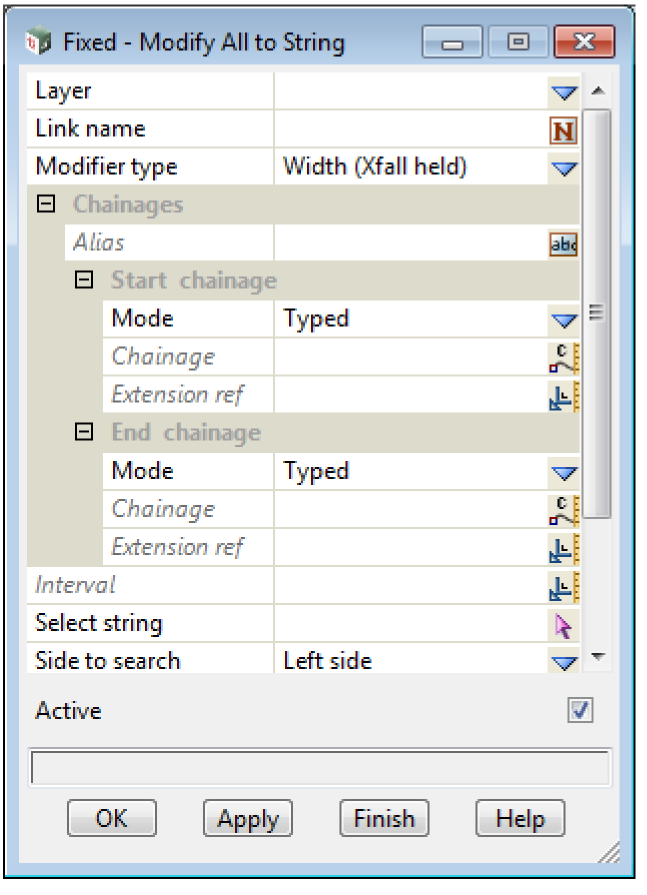
That is, for the current link:

(a) one of its existing Width, Height or Xfall **is modified** by taking its value the vector going from the start of the current link to another selected string**.**

and

(b) a different one of Width, Height or Xfall **keeps its existing value** (i.e. holds its existing value).

Selecting the **To string** brings up the **Fixed - Modify All to String** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box	Design	available layers

name of the layer that the link to be modified is in (the current link).
For information on Layers, see 21.1.1.2 MTF Links and Layers.

Link name	name box		names.4d pop-up

the name of the link in the Layer to modify the width/height/xfall/slope using the method given in **Modifier type,** *between the chainages given by* **Start mode** *and* **End mode.**
This is the current link.

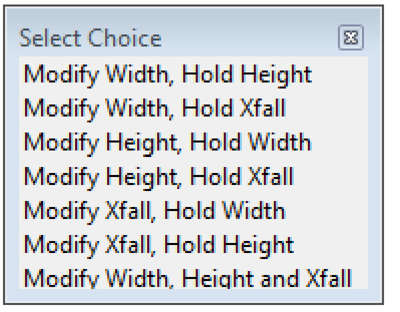
Select string	string-select		

select string to use for defining width/height/xfall for the link.

Side to search choice box left side left side, right side, both sides

side of the hinge string to start searching to find the string to define width/height/xfall.

Modifier type choice box



modify Width, hold Height to be its current value
modify Width, hold Xfall to be its current value
modify Height, hold Width to be its current value
modify Height, hold Xfall to be its current value
modify Xfall, hold Width to be its current value
modify Xfall, hold Height to be its current value
modify Width, Height and Xfall to get onto the selected string

A link can be modified in any way regardless of how the link was defined in terms of width, height and xfall or slope. See 21.1.2.1 Calculating Width, Height, Xfall or Slope from Original Definitions.

*The **Modify Width/Height/Xfall** is the part of the current link that is modified and the new values are taken from a selected link.*

*The **Hold Width/Height/Xfall** is the part of the current link that is kept (held) at its current value.*

So for **Modifier type:**

***Modify Width**, **Hold Height** see* 21.2.2.2.10.1 Fixed - to String: Modifier Types Modify Width, Hold Height/Xfall

***Modify Width**, **Hold Xfall** see* 21.2.2.2.10.1 Fixed - to String: Modifier Types Modify Width, Hold Height/Xfall

*For **Modify Height**, **Hold Width** see* 21.2.2.2.10.2 Fixed - to String: Modifier Types Modify Height, Hold Width/Xfall

*For **Modify Height**, **Hold Xfall** see* 21.2.2.2.10.2 Fixed - to String: Modifier Types Modify Height, Hold Width/Xfall

*For **Modify Xfall, Hold Width** see* 21.2.2.2.10.3 Fixed - to String: Modifier Types Modify Xfall, Hold Width/Height

*For **Modify Xfall, Hold Height** see* 21.2.2.2.10.3 Fixed - to String: Modifier Types Modify Xfall, Hold Width/Height

*For **Modify Width, Height and Xfall**, see* 21.2.2.2.10.4 Fixed - to String: Modifier Type Modify Width, Height and Xfall

Alias, **Start Chainage**, **End Chainage**, **Interval**

defines the start/end chainages for the link to be modified.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

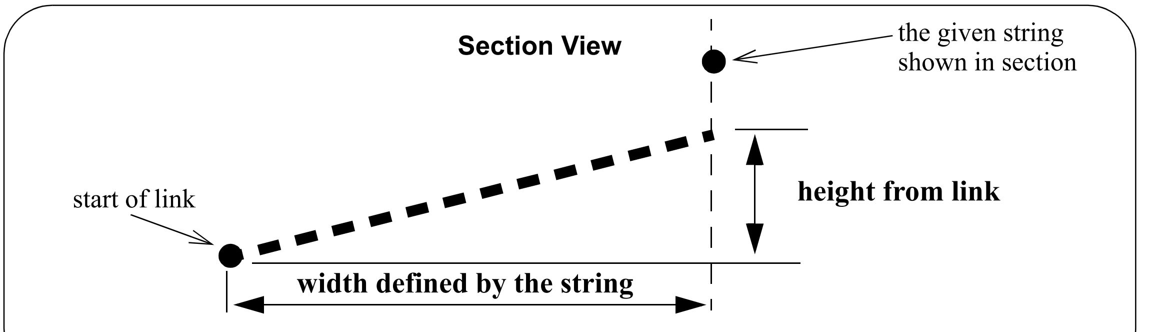
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Continue to the next section 21.2.2.2.11 Fixed Link - To Tin or return to 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

21.2.2.2.10.1 Fixed - to String: Modifier Types Modify Width, Hold Height/Xfall

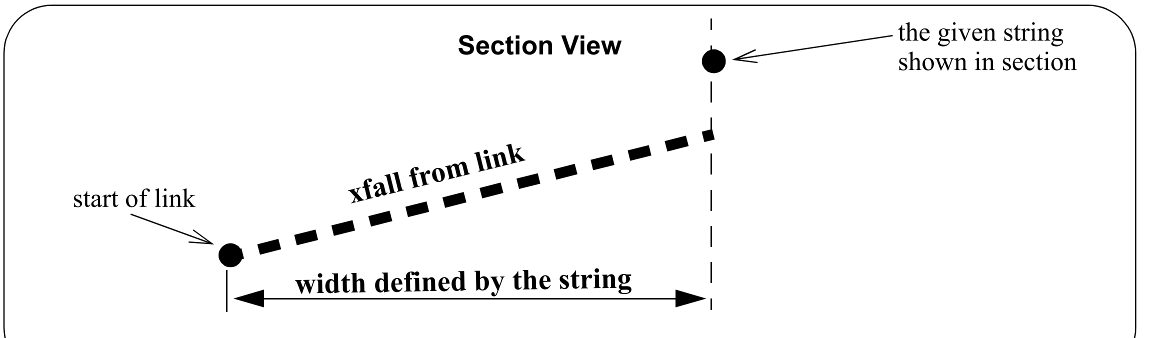
For a fixed link **Modify Width** calculates the **width** of the link as the width from the start point of the link, **to the selected string**.

For **Modify Width, Hold Height** the *height* is taken from the current link.



Modify Width to a String and Hold Height

For **Modify Width, Hold Xfall** the *xfall* is taken from the current link.

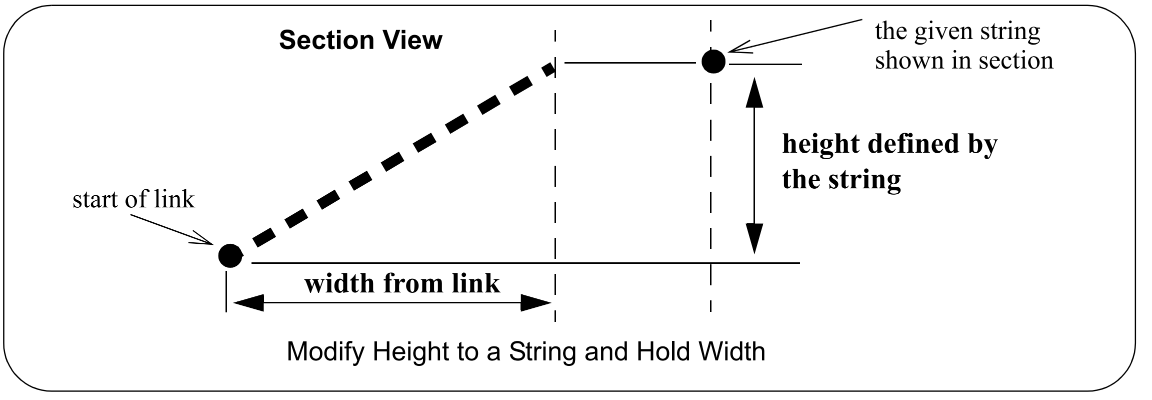


Modify Width to a String and Hold Xfall

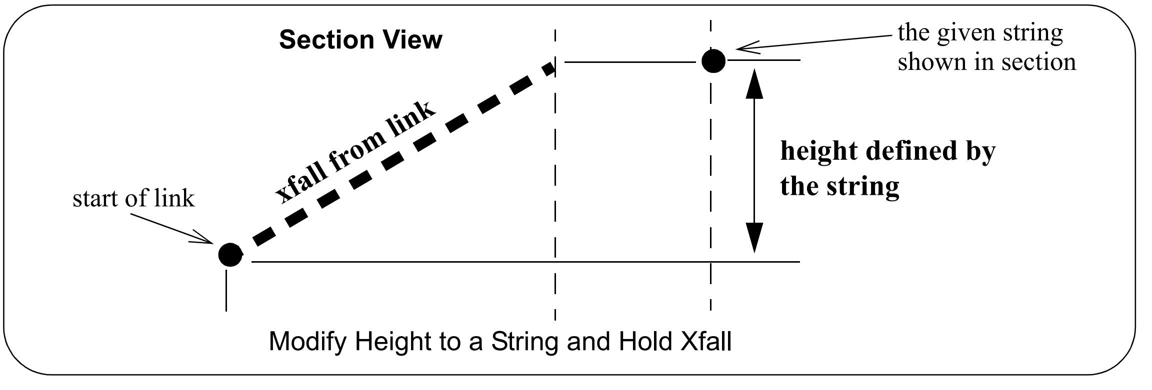
21.2.2.2.10.2 Fixed - to String: Modifier Types Modify Height, Hold Width/Xfall

For a fixed link **Modify Height** calculates the **height** of the link as the difference in the height at the start point of the link, and the height **at the selected string**.

For **Modify Height, Hold Width** the *width* is taken from the current link.



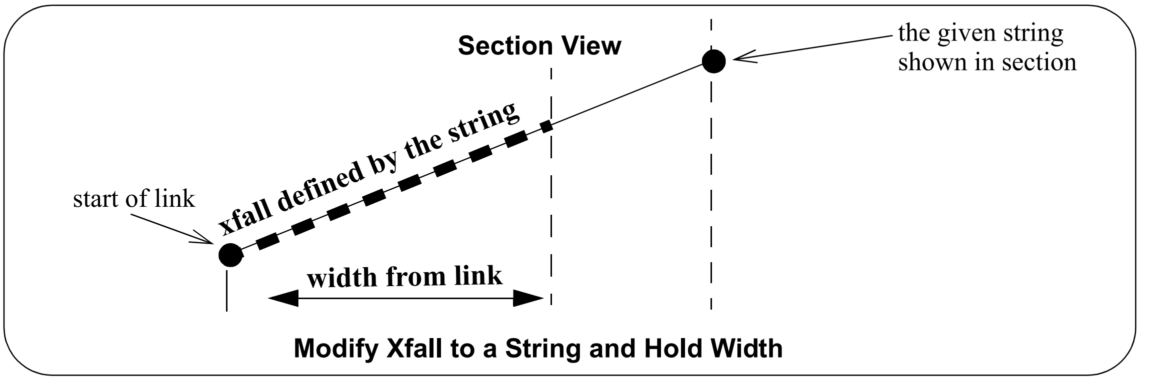
For **Modify Height, Hold Xfall** the *xfall* is taken from the current link.



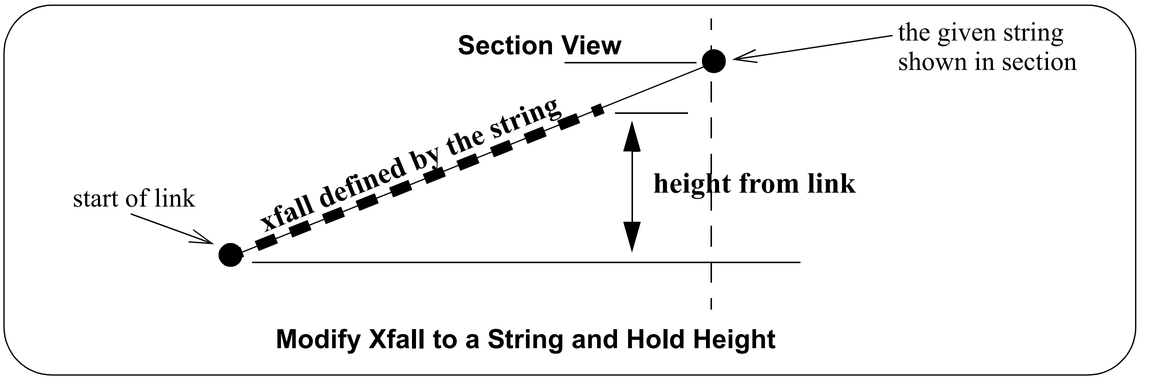
21.2.2.2.10.3 Fixed - to String: Modifier Types Modify Xfall, Hold Width/Height

For a fixed link **Modify Xfall** calculates the **xfall** of the link as the *xfall* from the start point of the link to the **selected string**.

For **Modify Xfall, Hold Width** the *width* is taken from the current link.

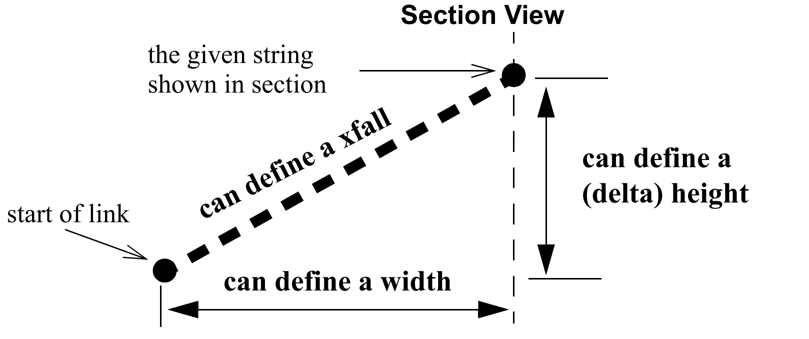


For **Modify Xfall, Hold Height** the *height* is taken from the link.



21.2.2.2.10.4 Fixed - to String: Modifier Type Modify Width, Height and Xfall

For a fixed link **Modify Width, Height and Xfall** calculates the required width, height and/or xfall that the link needs to get from the start point of the link **to the selected string**.



Modify Width, Height and Xfall - the start and end points are used to uniquely define the width, xfall and (delta) height

To use the option you must **first have a link defined** but there is a command **Fixed =>Insert =>Insert at string** which creates a link and also places it on a selected string. See 21.2.2.2.1.3 Insert a Fixed Link at a String.

21.2.2.2.11 Fixed Link - To Tin

For a given link, (the current link), the **to tin** option takes **one** of Width, Height or Xfall as the value you need so that the end of the link is at the point that is a given number of intersection with a given tin.

That is, for the current link:

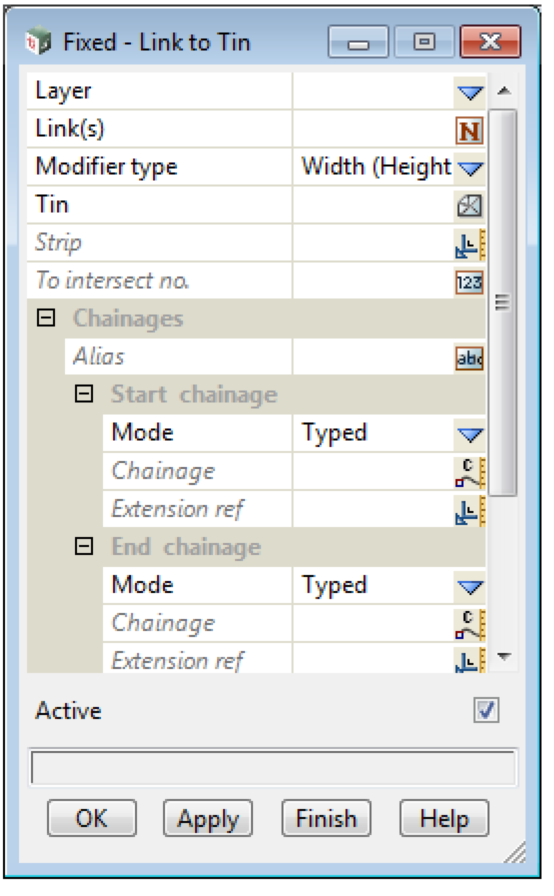
(a) one of Width, Height or Xfall **keeps its existing value** (i.e. holds its existing value).

and

(b) a different one of Width, Height or Xfall **is modified** so that the end of the link is at the point that is a given number of intersection with the given tin**.**

Note that the tin does not have to be the same tin as used in the ***Apply MTF*** function that is using the MTF.

Selecting the **To tin** brings up the **Fixed - Link to Tin** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box	Design	available layers

name of the layer that the link to be modified is in.
For information on Layers, see 21.1.1.2 MTF Links and Layers.

Field Description	Type	Defaults	Pop-Up
Link(s)	name box		names.4d pop-up

the name of the links in the Layer to modify the width/height/xfall/slope using the method given in **Modifier type,** *between the chainages given by* **Start mode** *and* **End mode.** *See* 21.2.2.1.2 Link or Link(s).
Each of the links is processed as the current link.

Tin tin box available tins

the tin to use for defining the width/height/xfall/slope by getting a point of intersection with the tin.

To intersect no. integer box

a tin could be intersected a number of times by the current link.

*If **not blank,** the number of the intersection with the tin to use. It should be greater than 0. If the number of intersections with the tin is less than this then the last intersection is used. For example, if the number is 5 then in the cases of 1,2,3,4 or 5 intersections, you get the last intersection.*

*If **blank**, the first intersect is used.*

Modifier type choice box

modify Width and take Height from current link
modify Width and take Xfall from current link
modify Height and take Width from current link
modify Height and take Xfall from current link
modify Xfall and take With from current link
modify Xfall and take Height from current link
modify Slope and take With from current link
modify Slope and take Height from current link

A link can be modified in any way regardless of how the link was defined in terms of width, height and xfall or slope. See 21.1.2.1 Calculating Width, Height, Xfall or Slope from Original Definitions.

*The **Modify Width/Height/Xfall** is the part of the current link that is modified and the new values are obtained by get an intersection with the tin.*

*The **Hold Width/Height/Xfall** is the part of the current link that is kept (held) at its current value.*

So for **Modifier type:**

***Modify Width**, **Hold Height** see* 21.2.2.2.11.1 Fixed - to Tin: Modifier Types Modify Width, Hold Height/Xfall

***Modify Width**, **Hold Xfall** see* 21.2.2.2.11.1 Fixed - to Tin: Modifier Types Modify Width, Hold Height/Xfall

*For **Modify Height**, **Hold Width** see* 21.2.2.2.11.2 Fixed - to Tin: Modifier Types Modify Height, Hold Width/Xfall

*For **Modify Height**, **Hold Xfall** see* 21.2.2.2.11.2 Fixed - to Tin: Modifier Types Modify Height, Hold Width/Xfall

*For **Modify Xfall, Hold Width** see* 21.2.2.2.11.3 Fixed - to Tin: Modifier Types Modify Xfall, Hold Width/Height

*For **Modify Xfall, Hold Height** see* 21.2.2.2.11.3 Fixed - to Tin: Modifier Types Modify Xfall, Hold Width/Height

*For **Modify Slope, Hold Width** see* 21.2.2.2.11.4 Fixed - to Tin: Modifier Types Modify Slope, Hold Width/Height

*For **Modify Slope, Hold Height** see* 21.2.2.2.11.4 Fixed - to Tin: Modifier Types Modify Slope, Hold Width/Height

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start and end chainages that the link is modified between.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

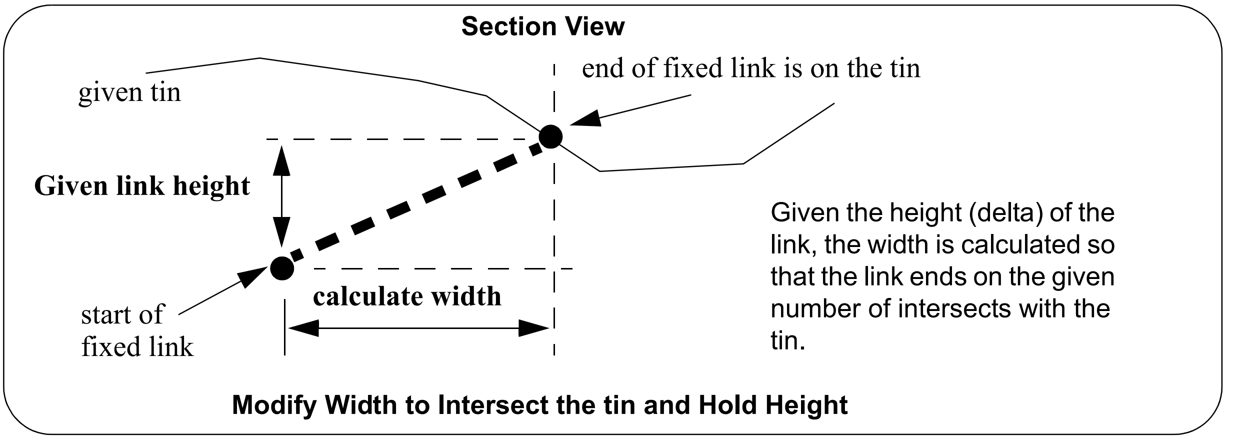
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Continue to the next section 21.2.2.2.12 Fixed - To RL or return to 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

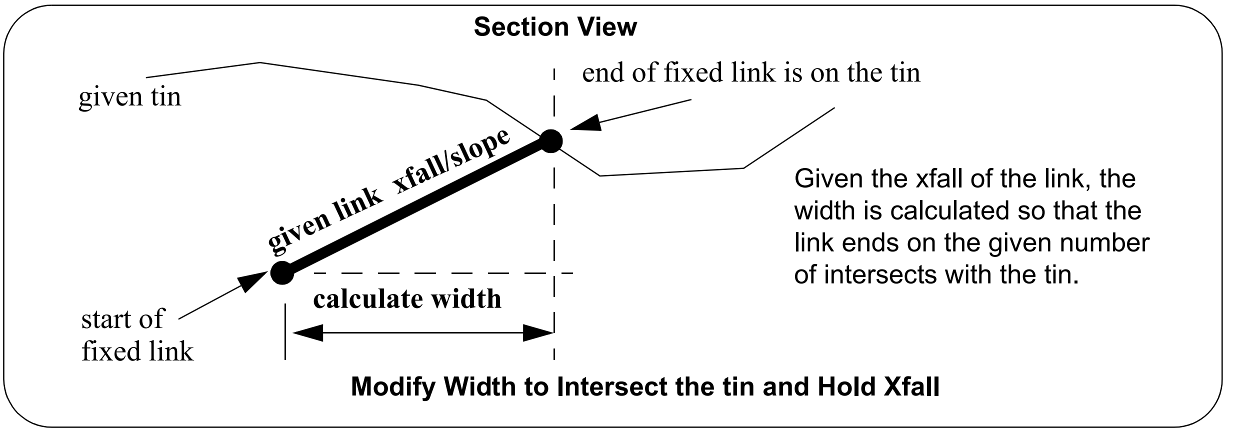
21.2.2.2.11.1 Fixed - to Tin: Modifier Types Modify Width, Hold Height/Xfall

For a fixed link **Modify Width** calculates the **width** of the link so that it intersects the tin at the number of intersects equal to **To intersect no**.

For **Modify Width, Hold Height** the *height* is taken from the current link.



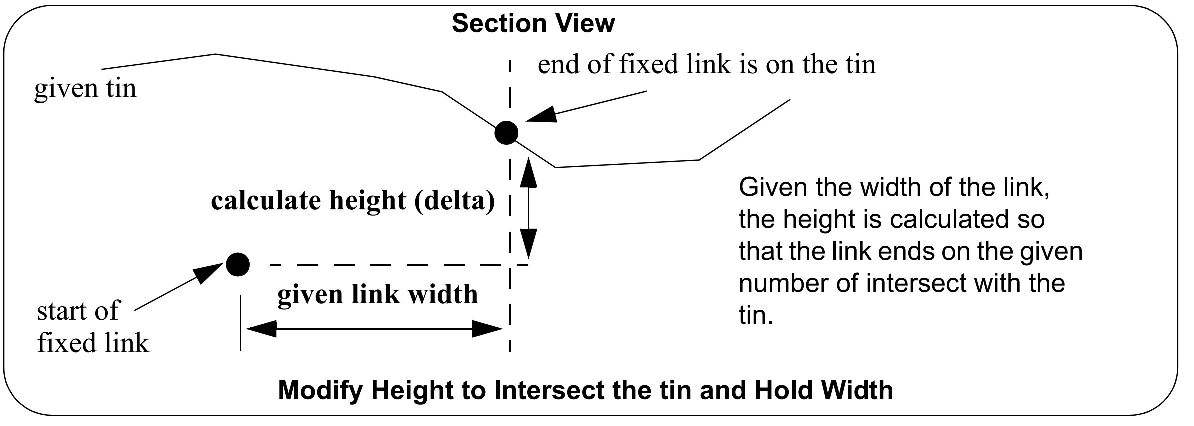
For **Modify Width, Hold Xfall** the *xfall* is taken from the current link.



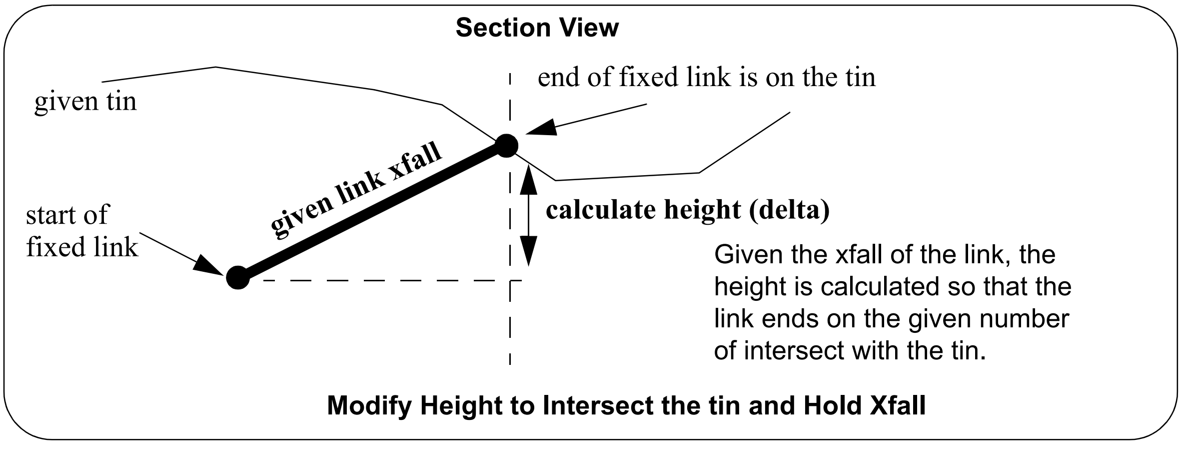
21.2.2.2.11.2 Fixed - to Tin: Modifier Types Modify Height, Hold Width/Xfall

For a fixed link **Modify Height** calculates the **height** of the link so that it intersects the tin at the number of intersects equal to **To intersect no**.

For **Modify Height, Hold Width** the *width* is taken from the current link.



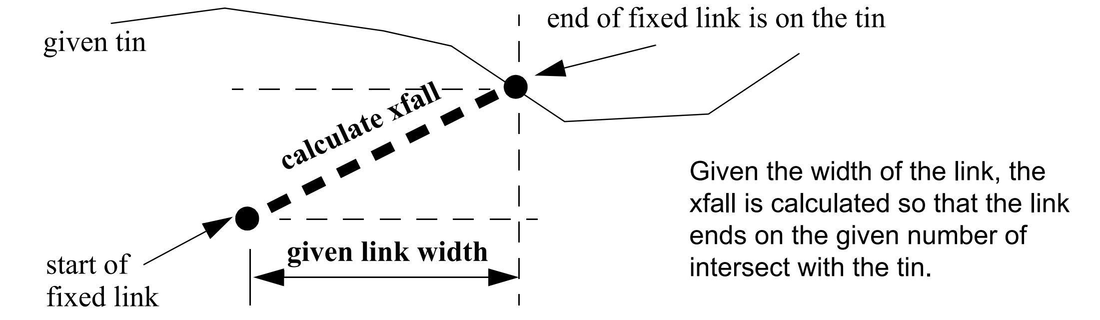
For **Modify Height, Hold Xfall** the *xfall* is taken from the current link.



21.2.2.2.11.3 Fixed - to Tin: Modifier Types Modify Xfall, Hold Width/Height

For a fixed link **Modify Xfall** calculates the **xfall** of the link so that it intersects the tin at the number of intersects equal to **To intersect no**.

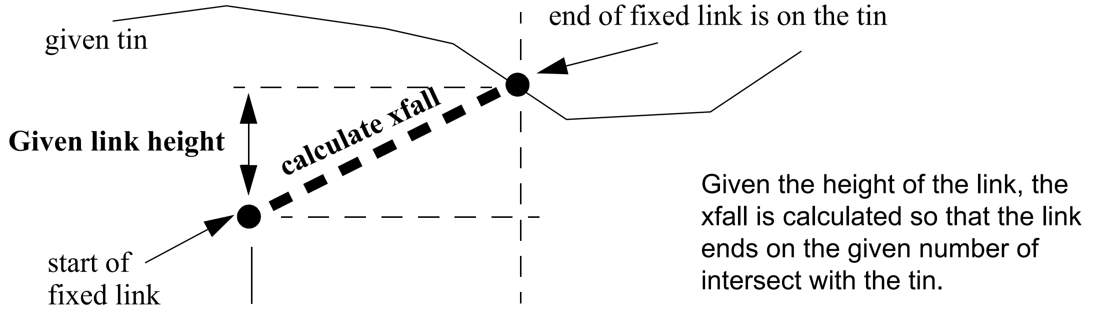
For **Modify Xfall, Hold Width** the *width* is taken from the current link.



Modify Xfall to Intersect the tin and Hold Width

For **Modify Xfall, Hold Height** the *height* is taken from current the link.

Section View

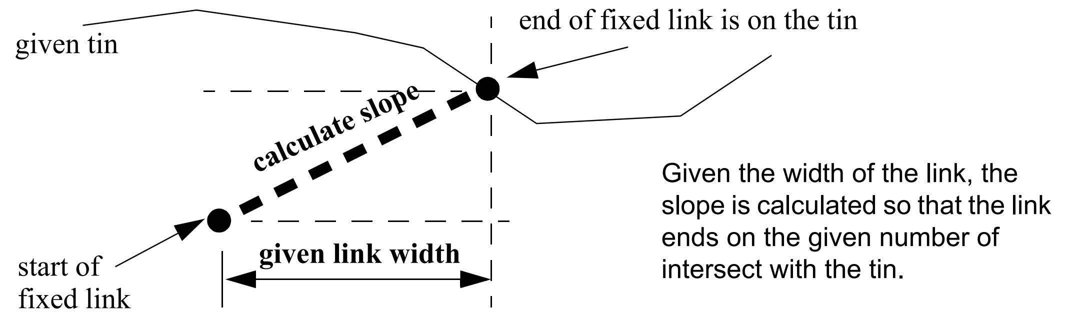


Modify Xfall to Intersect the tin and Hold Height

21.2.2.2.11.4 Fixed - to Tin: Modifier Types Modify Slope, Hold Width/Height

For a fixed link **Modify Slope** calculates the **slope** of the link so that it intersects the tin at the number of intersects equal to **To intersect no**.

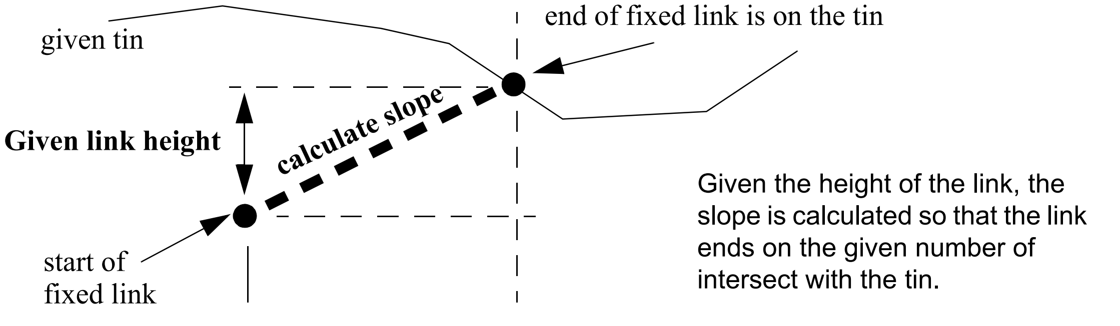
For **Modify Slope, Hold Width** the *width* is taken from the current link.



Modify Slope to Intersect the tin and Hold Width

For **Modify Slope, Hold Height** the *height* is taken from the current link.

Section View



Modify Slope to Intersect the tin and Hold Height

21.2.2.2.12 Fixed - To RL

For a given link (the current link), the **to RL** option **takes** (holds) **one** of Width, Height or Xfall as it currently is, and then modifies a different one of Width, Height or Xfall to get to a given RL.

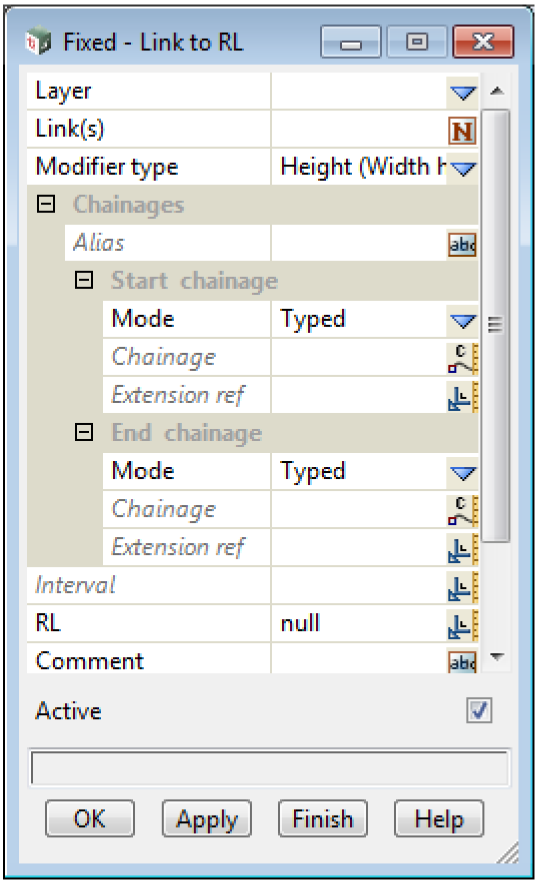
That is, for the current link:

(a) one of Width, Height or Xfall **keeps its existing value** (i.e. **holds** its **existing value)**.

and

(b) a different one of Width, Height or Xfall **is modified** to the value that is needed to get to a given RL (elevation).

Selecting the **To RL** brings up the **Fixed - Link to RL** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box	Design	available layers

name of the layer that the link to be modified is in.

For information on Layers, see 21.1.1.2 MTF Links and Layers.

Link(s)	name box		names.4d pop-up

the name of the links in the Layer to modify the width/height/xfall/slope using the method given in **Modifier type,** *between the chainages given by* **Start mode** *and* **End mode.** *See* 21.2.2.1.2 Link or Link(s).

Each of the links is processed as the current link.

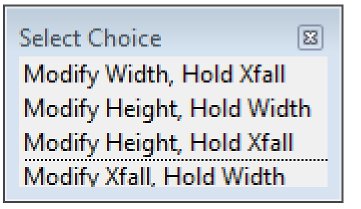
RL	real box		

the RL (Elevation) to be reached at the end of the link.

Side to search choice box left side left side, right side, both sides

side of the hinge string to start searching to find the string to define width/height/xfall.

Modifier type choice box



modify Width and take Xfall from current link
modify Height and take Width from current link
modify Height and take Xfall from current link
modify Xfall and take With from current link

A link can be modified in any way regardless of how the link was defined in terms of width, height and xfall or slope. See 21.1.2.1 Calculating Width, Height, Xfall or Slope from Original Definitions.

The ***Hold Width/Height/Xfall*** *is the part of the current link that is kept (held) at its current value.*

The ***Modify Width/Height/Xfall*** *is the part of the current link that is then modified and the new values are obtained by getting to a given RL.*

So for **Modifier type:**

***Modify Width**, **Hold Xfall** see* 21.2.2.2.12.1 Fixed - to RL: Modifier Type Modify Width, Hold Xfall

*For **Modify Height**, **Hold Width** see* 21.2.2.2.12.2 Fixed - to RL: Modifier Types Modify Height, Hold Width/Xfall

*For **Modify Height**, **Hold Xfall** see* 21.2.2.2.12.2 Fixed - to RL: Modifier Types Modify Height, Hold Width/Xfall

*For **Modify Xfall, Hold Width** see* 21.2.2.2.12.3 Fixed - to RL: Modifier Type Modify Xfall, Hold Width

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start and end chainages that the link is modified between.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

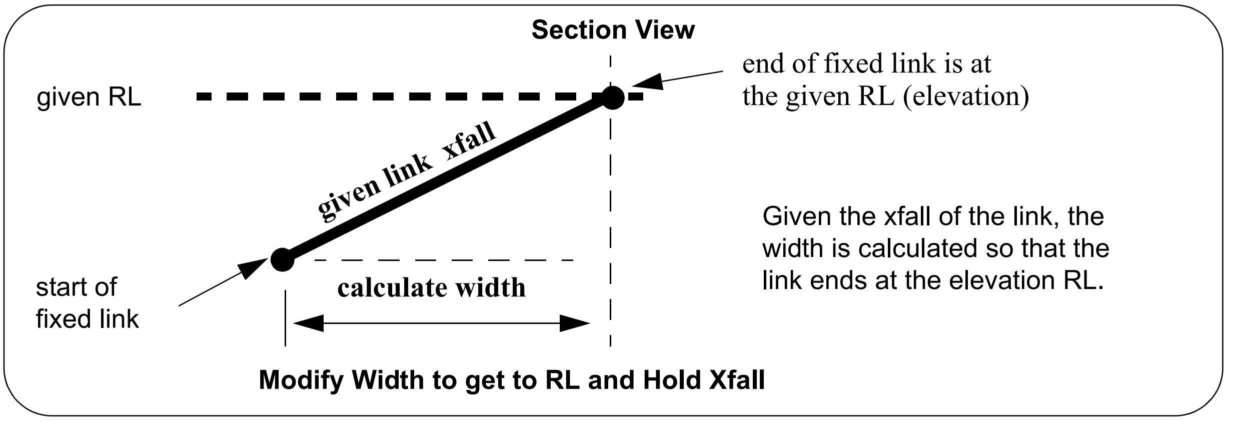
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Continue to the next section 21.2.2.2.13 Fixed - to 2 Heights or return to 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

21.2.2.2.12.1 Fixed - to RL: Modifier Type Modify Width, Hold Xfall

For a fixed link **Modify Width** calculates the **width** of the link required for link to end **at the given RL**.

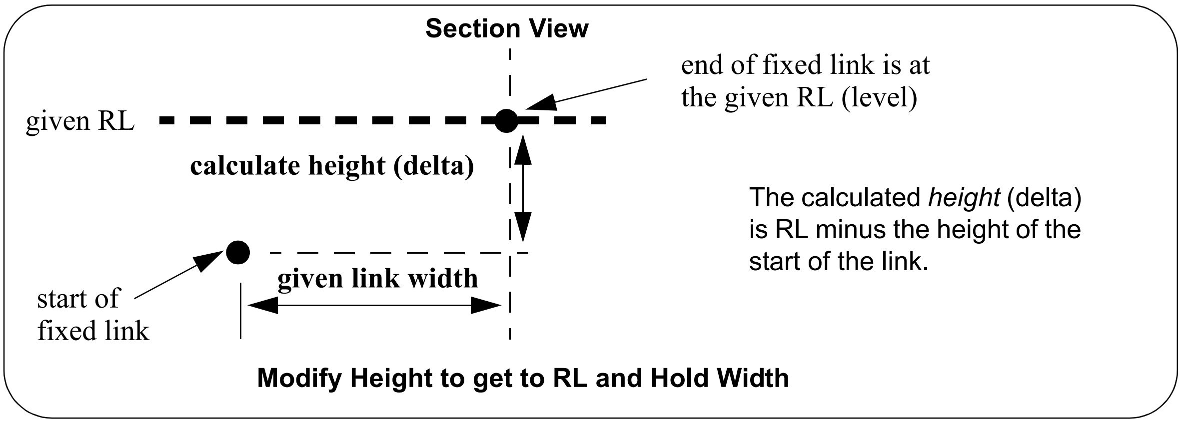
For **Modify Width, Hold Xfall** the *xfall* is taken from the current link.



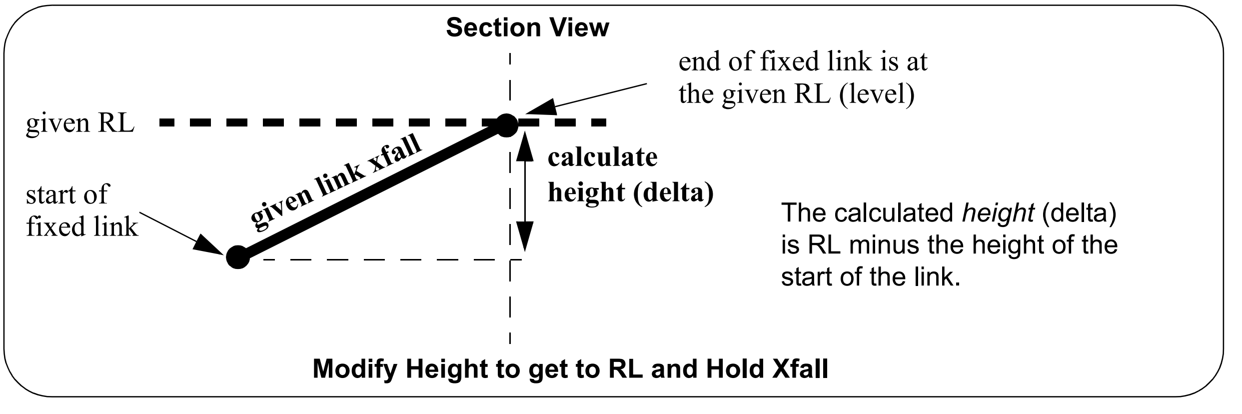
21.2.2.2.12.2 Fixed - to RL: Modifier Types Modify Height, Hold Width/Xfall

For a fixed link **Modify Height** calculates the **height** of the link as the difference in the height at the start point of the link, and the given **RL**.

For **Modify Height, Hold Width** the *width* is taken from the current link.



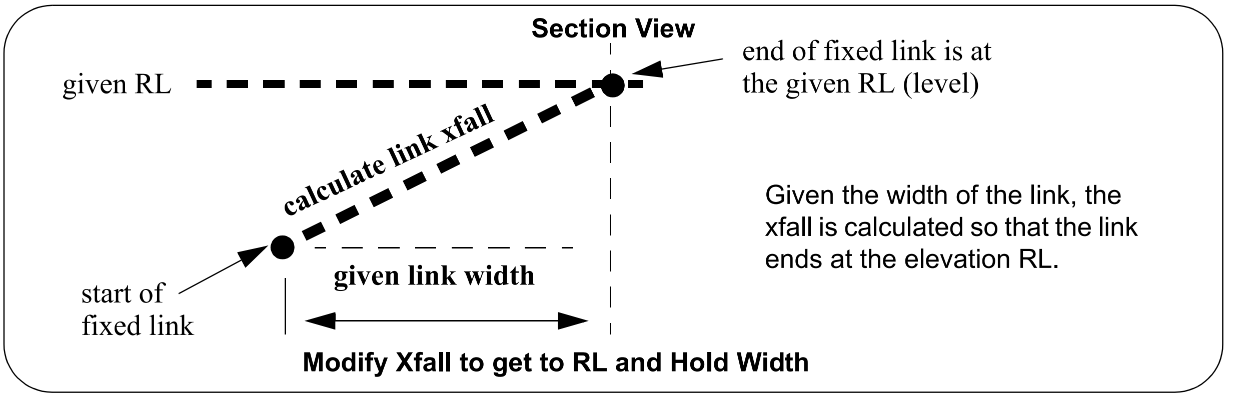
For **Modify Height, Hold Xfall** the *xfall* is taken from the current link.



21.2.2.2.12.3 Fixed - to RL: Modifier Type Modify Xfall, Hold Width

For a fixed link **Modify Xfall** calculates the **xfall** of the link as the *xfall* from the start point of the link to the **given RL**.

For **Modify Xfall, Hold Width** the *width* is taken from the current link.

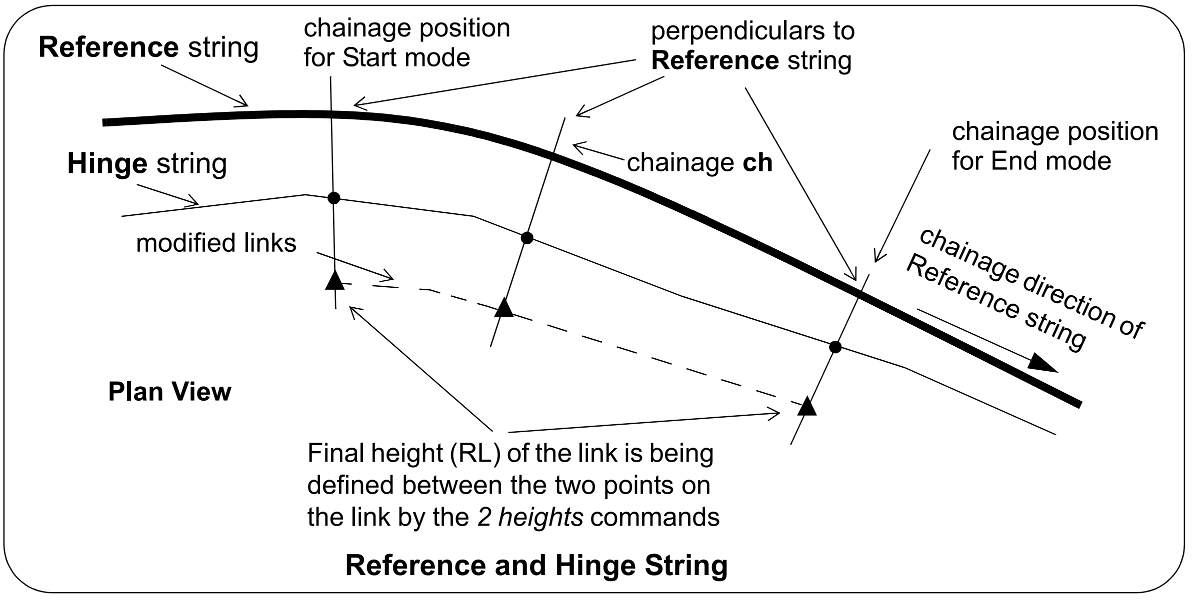


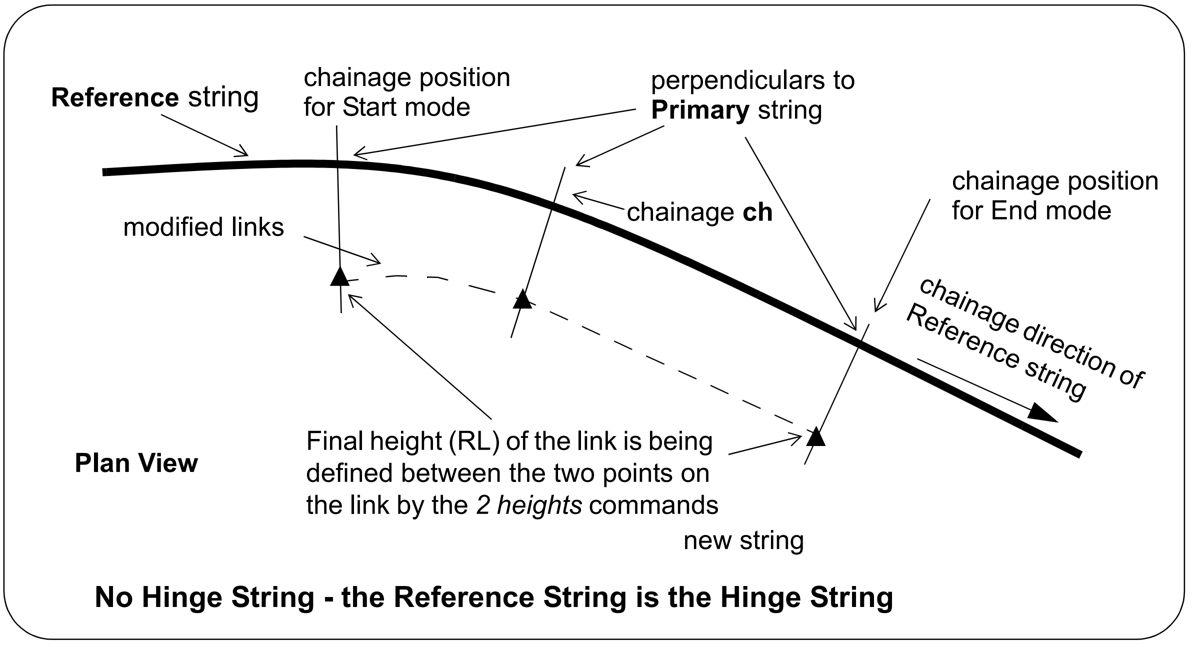
21.2.2.2.13 Fixed - to 2 Heights

MTF_Edit_Modify_Fixed___to_2_HeightsFor an existing link, the **to 2 heights** option has a number of methods for **defining the height** and **interpolation** occurs **over the length of the modified string** and **not** over the **length** of the **reference string** (the chainage range).

To options also includes **coming back** from an **RL** at the **end chainage** and using that to **define** the **RL** at the **start chainage**.

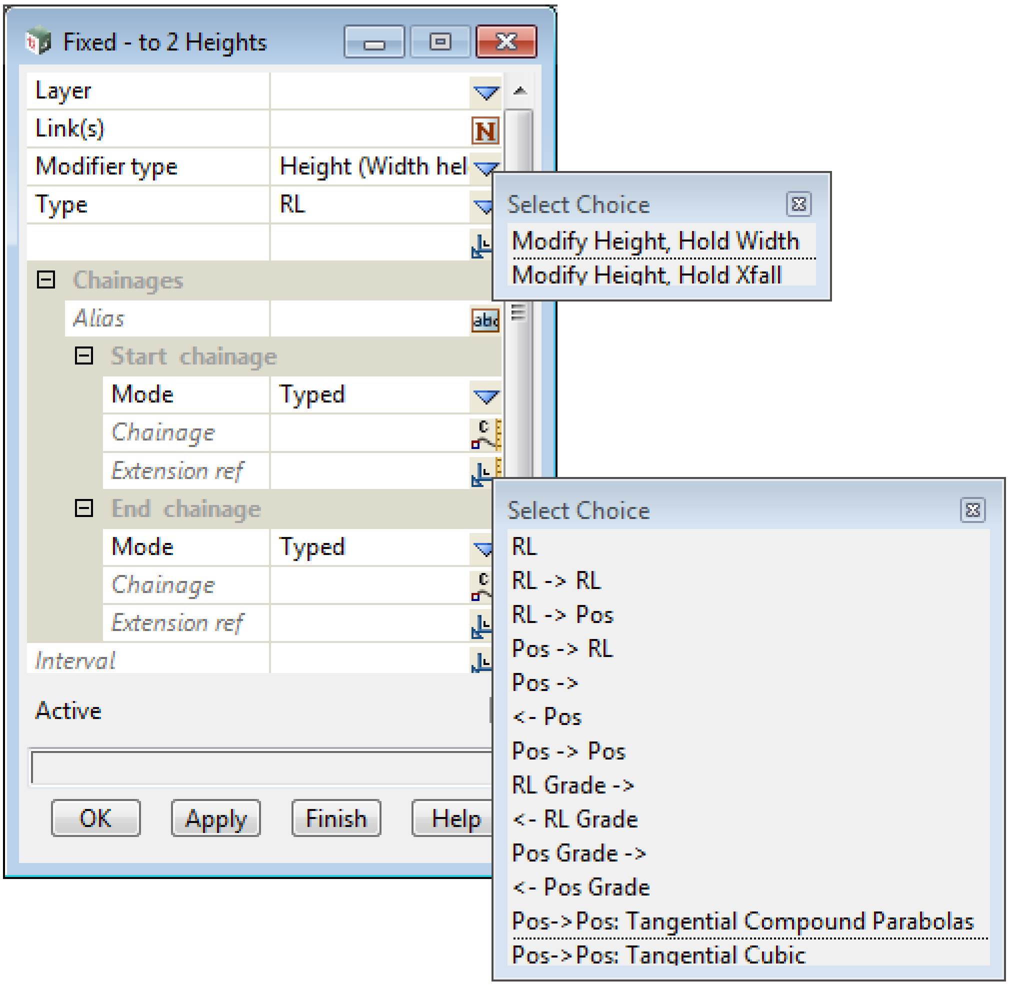
For example, over the chainage range, the height can be interpolated between two given RL's.





The modifier ***to RL (***21.2.2.2.12 Fixed - To RLmod has a **constant RL** whereas for **to 2 heights**, the RL at each chainage is calculated by a formula.

Selecting **to 2 heights** brings up the **Fixed - to 2 Heights** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box	Design	available layers

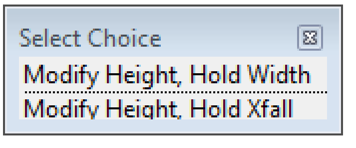
name of the layer that the link to be modified is in.

For information on Layers, see 21.1.1.2 MTF Links and Layers.

Field Description	Type	Defaults	Pop-Up
Link(s)	input		select name menu

name of the links to modify. See 21.2.2.1.2 Link or Link(s).

Field Description	Type	Defaults	Pop-Up
Modifier type	choice box		

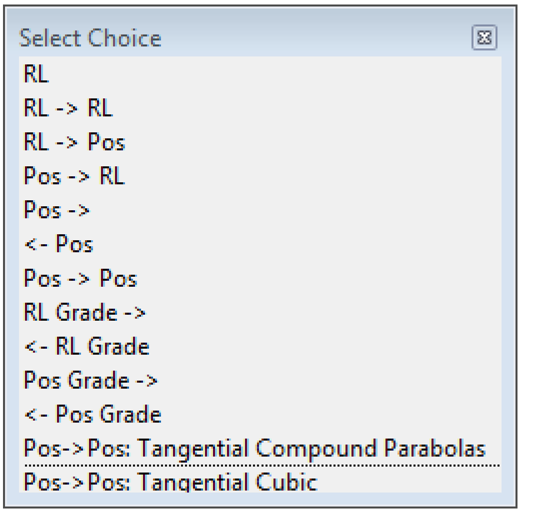


*The option **to 2 heights** defines the **Height** of the selected string so to completely define the link, only the Width or the Xfall is needed.*

*For the choice **Modify Height, Hold Width**, the **Width is kept** for the link (**hold**) and the **Height** is **calculated** by the method given by* **Type**.

*For the choice **Modify Height, Hold Xfall**, the **Xfall** is kept for the link (**hold**) and the **Height** is **calculated** by the method given by* **Type**.

Type choice box



*The **Height** for the current link is calculated by the method given by **Type**.*

For the calculation of Height for each type, go to 21.2.2.2.13.1 Fixed - Calculating the Heights for each Type

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start and end chainages that the link is modified between.

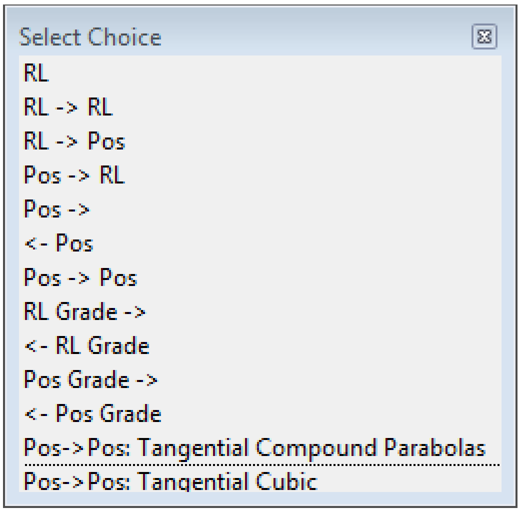
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Continue to the next section 21.2.2.2.14 Fixed - by 2 strings or return to 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

21.2.2.2.13.1 Fixed - Calculating the Heights for each Type

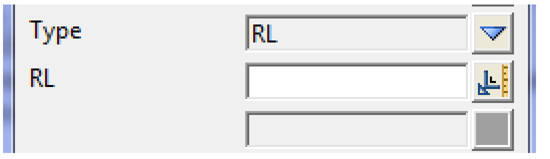


for the definitions of the calculations for each choice go to

21.2.2.2.13.1.1 RL: stay at a given RL
21.2.2.2.13.1.2 RL -> RL: interpolate between two given RL’s
21.2.2.2.13.1.6 RL -> Pos: interpolate between a given start RL & the calculated end height
21.2.2.2.13.1.7 Pos -> RL: interpolate between the calculated start height and a given RL at the end
21.2.2.2.13.1.3 Pos ->: all points have the same height as the start point
21.2.2.2.13.1.4 <- Pos: all points have the same height as the end point
21.2.2.2.13.1.5 Pos-> Pos: interpolate between the calculated start and end heights

21.2.2.2.13.1.8 RL Grade ->: start with a given Start RL and continue on the line at grade Grade->
21.2.2.2.13.1.9 <- RL Grade: all points are on a line with given grade and going through End RL
21.2.2.2.13.1.10 Pos Grade ->: start with the calculated height and continue at a given grade
21.2.2.2.13.1.11 <- Pos Grade: all points are on a line of given grade and going through the end point
21.2.2.2.13.1.12 Pos-> Pos: Tangential Compound Parabolas
21.2.2.2.13.1.13 Pos-> Pos: Tangential Cubic

21.2.2.2.13.1.1 RL: stay at a given RL

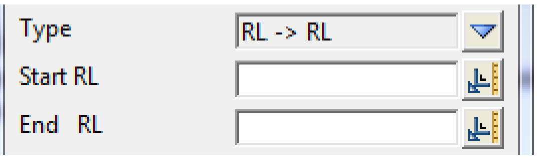


At the vertex of the link on the normal at each chainage ch between the **Start mode** and **End Mode**, the height at ch is that given in the **RL** panel field.

Hence the height is the same for all modified vertices between the *Start mode* and the *End mode*.

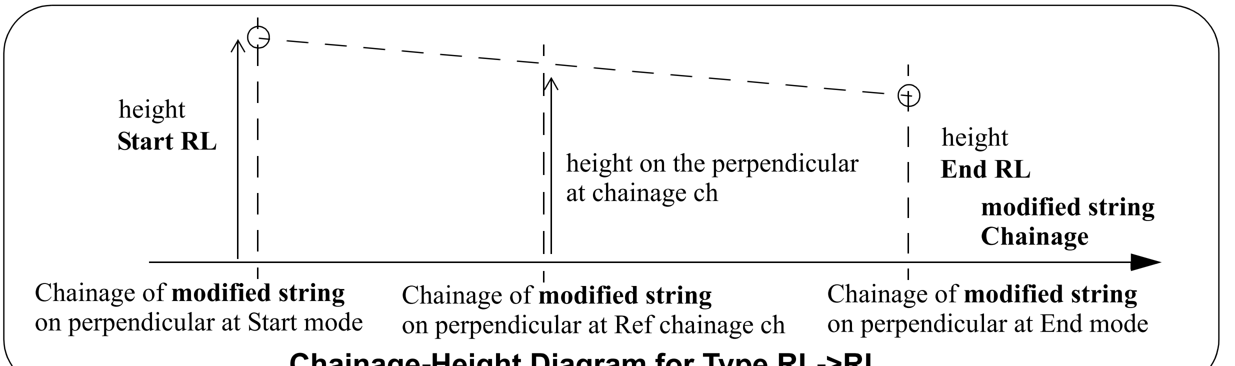
The modifiers with *Type* ***RL*** work the same as the modifier **to RL** with the RL being the *given RL* (see 21.2.2.2.12 Fixed - To RL).

21.2.2.2.13.1.2 RL -> RL: interpolate between two given RL's



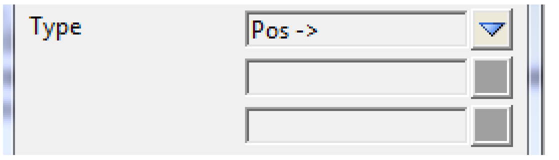
At each chainage **ch** between the **Start mode** and **End Mode**, the vertex of the modified string is on the perpendicular to the Reference string at chainage **ch**.

The height at of the vertex on the perpendicular to Reference chainage ch, is the *linear interpolation* of height **with respect to the string being modified**, between the given *Start RL* and the *End RL*.



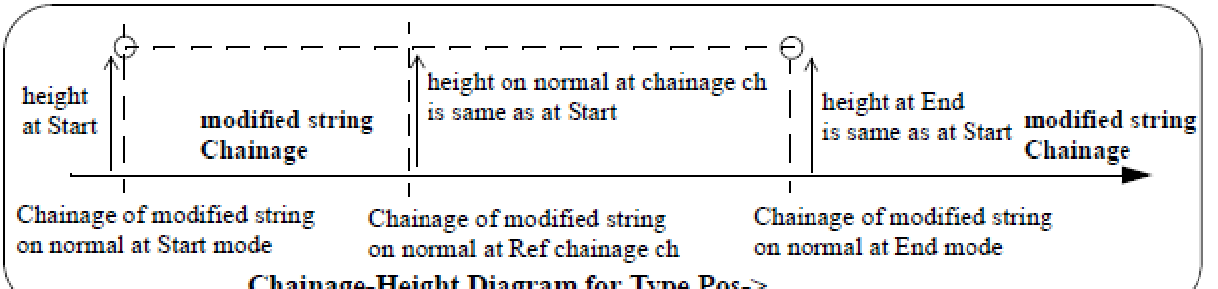
Chainage-Height Diagram for Type RL->RL

21.2.2.2.13.1.3 Pos ->: all points have the same height as the start point



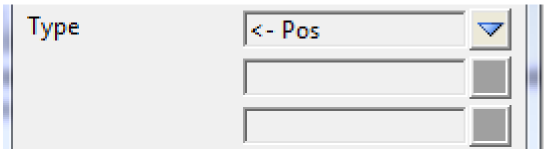
Let **Start RL** be the actual height calculated at the vertex of the string at the perpendicular to **Start mode** when all the modifiers before this modifier are applied.

Then for all vertices on the perpendiculars between **Start mode** and **End Mode**, the height of the vertex equal to **Start RL**.



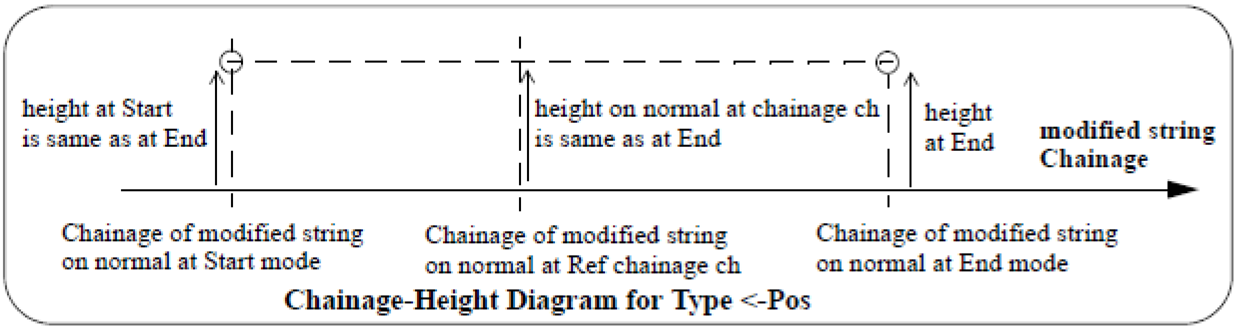
Chainage-Height Diagram for Type Pos->

21.2.2.2.13.1.4 <- Pos: all points have the same height as the end point



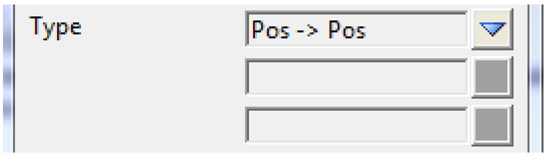
Let **End RL** be the actual height calculated at the vertex of the string at the perpendicular (normal) to **End mode** when all the modifiers before this modifier are applied.

Then for all vertices on the normals between the **Start mode** and **End Mode**, the height of the vertex is equal to **End RL**.



Chainage-Height Diagram for Type <-Pos

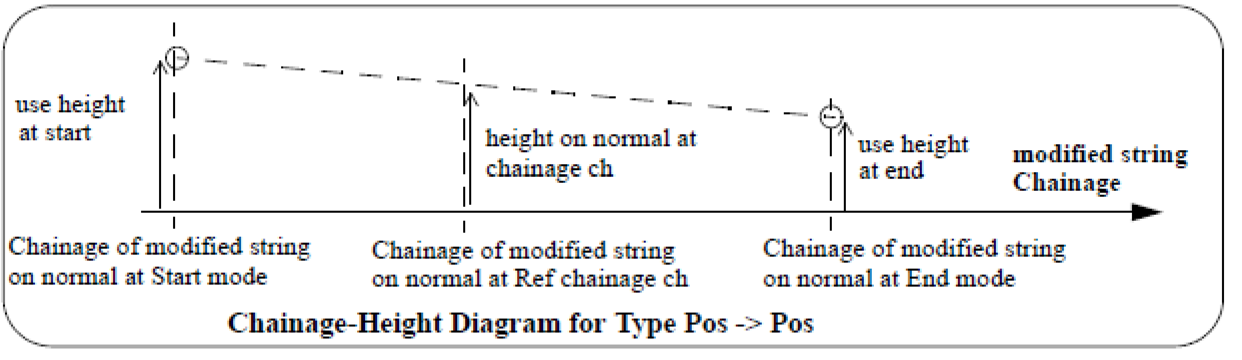
21.2.2.2.13.1.5 Pos-> Pos: interpolate between the calculated start and end heights



First let **Start RL** be the actual height calculated at the vertex of the string at the normal to **Start mode** when all the modifiers before this modifier are applied.

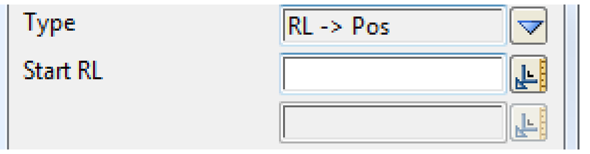
Let **End RL** be the actual height calculated at the vertex of the string at the normal to **End mode** when all the modifiers before this modifier are applied.

Then for each vertex on the normals between **Start mode** and **End Mode**, the height of the vertex is the linear interpolation **with respect to modified string chainage** between the *Start RL* and the *End RL*.



Chainage-Height Diagram for Type Pos -> Pos

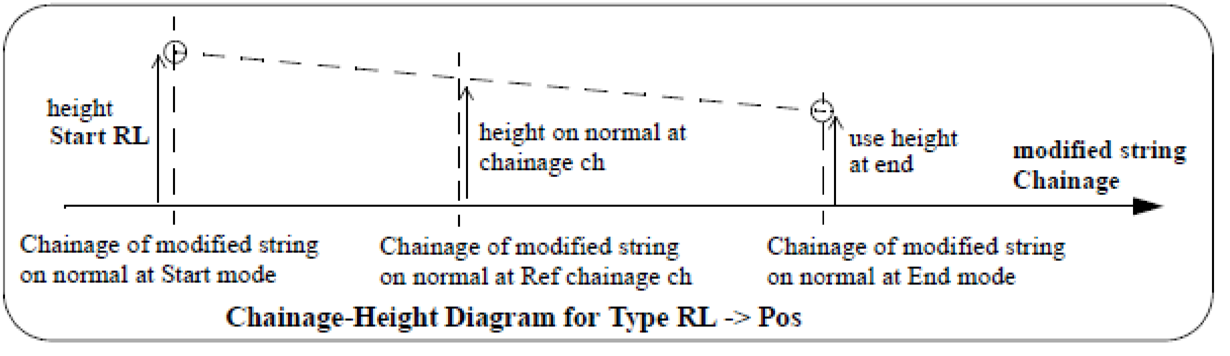
21.2.2.2.13.1.6 RL -> Pos: interpolate between a given start RL & the calculated end height



The height at the vertex on the normal at **Start mode** is the given **Start RL**.

And **End RL** is the **actual** height calculated at the vertex on the modified string at the normal at **End mode** when all the modifiers before this modifier are applied.

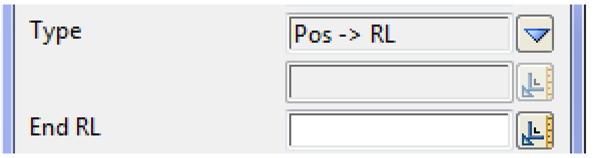
Then at each vertex on the normals between **Start mode** and **End Mode**, the height of the vertex is the linear interpolation **with respect to modified string chainage** between the *Start RL* and the *End RL*.



Chainage-Height Diagram for Type RL -> Pos

Hence for the modifiers of *Type* ***RL -> Pos***, the calculations at chainage ch are the same as the modifiers **Width/Height/Xfall to RL** with the RL being the interpolated RL at chainage ch (see Fixed Link - Modify Width, Height or Xfall to Get to an RL).

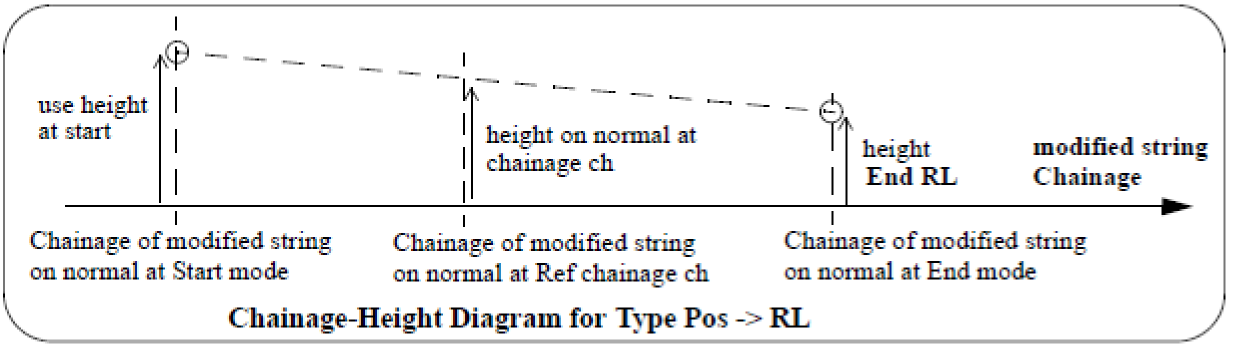
21.2.2.2.13.1.7 Pos -> RL: interpolate between the calculated start height and a given RL at the end



Start RL is the **actual** height calculated at the vertex on the normal at **Start mode** when all the modifiers before this modifier are applied.

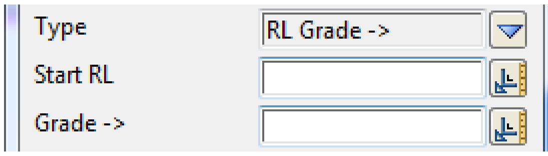
The height at the vertex on the normal at **End mode** is the given **End RL**.

Then at each vertex on the normals between **Start mode** and **End Mode**, the height of the vertex is the linear interpolation **with respect to modified string chainage** between the *Start RL* and the *End RL*.



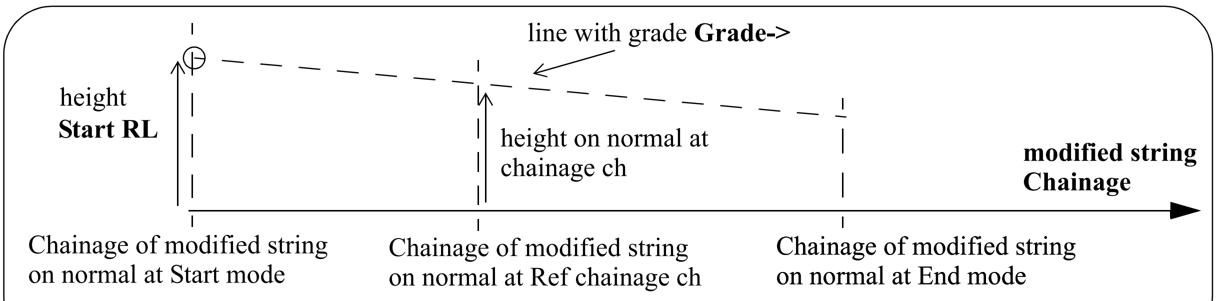
Chainage-Height Diagram for Type Pos -> RL

21.2.2.2.13.1.8 RL Grade ->: start with a given Start RL and continue on the line at grade Grade->



The height at the vertex on the normal at **Start mode** is the **Start RL**.

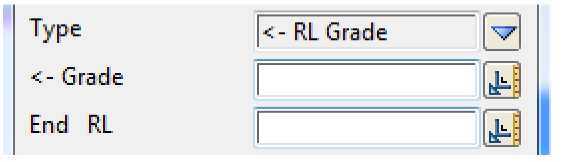
For each normal between the *Start mode* and the *End Mode*, the height of the vertex is on the line with grade ***Grade->*** with respect to the modified string, and going through the point (normal at Start mode, Start RL).



Chainage-Height Diagram for Type RL Grade ->

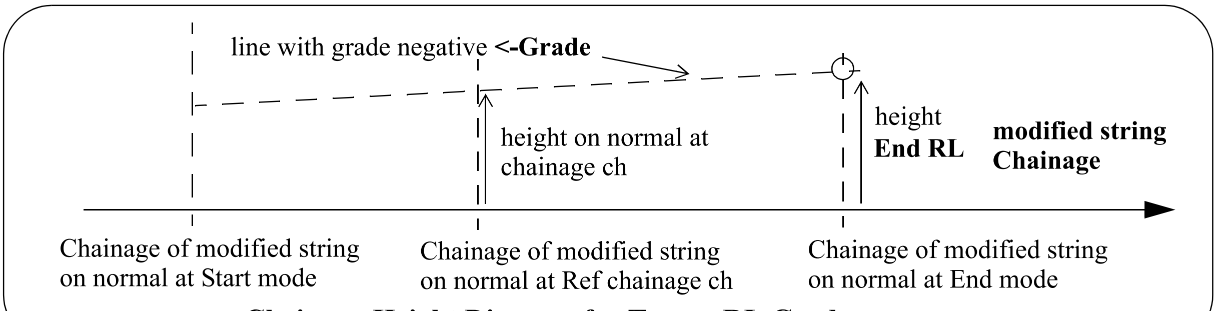
Important Note: Grade is calculated using **increasing Reference chainages**.

21.2.2.2.13.1.9 <- RL Grade: all points are on a line with given grade and going through End RL



The height on the vertex at **End mode** is **End RL**.

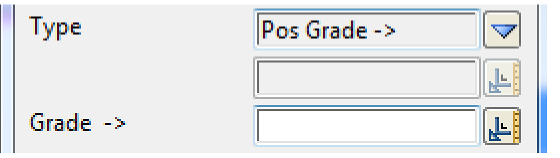
For each normal between the *Start mode* and the *End Mode*, the height of the vertex is on the line with grade of negative **<-Grade** with respect to the modified string chainage, and going through the point (normal at End mode, End RL).



Chainage-Height Diagram for Type <-RL Grade

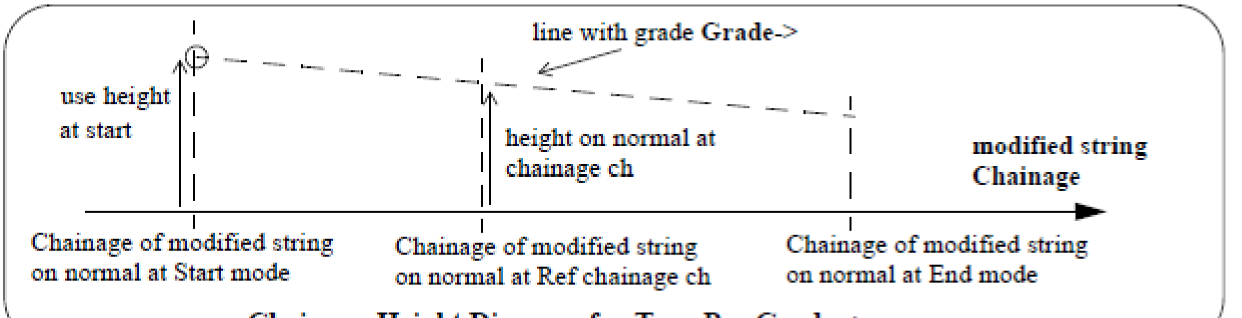
Important Note: <-Grade is calculated using **decreasing Alignment chainages**. So in a diagram with increasing chainage, the grade is negative **<-Grade**.

21.2.2.2.13.1.10 Pos Grade ->: start with the calculated height and continue at a given grade



Let **Start RL** be the actual height calculated at the vertex on the normal at **Start mode** when all the modifiers before this modifier are applied.

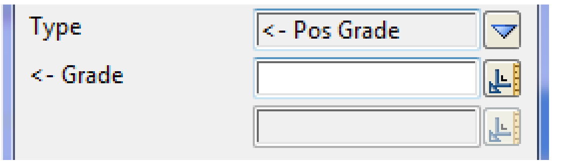
Then the height of each vertex on each normal between the *Start mode* and the *End Mode* is on the line with grade ***Grade->*** with respect to the modified string chainage, and going through the point (normal at Start mode, Start RL).



Chainage-Height Diagram for Type Pos Grade ->

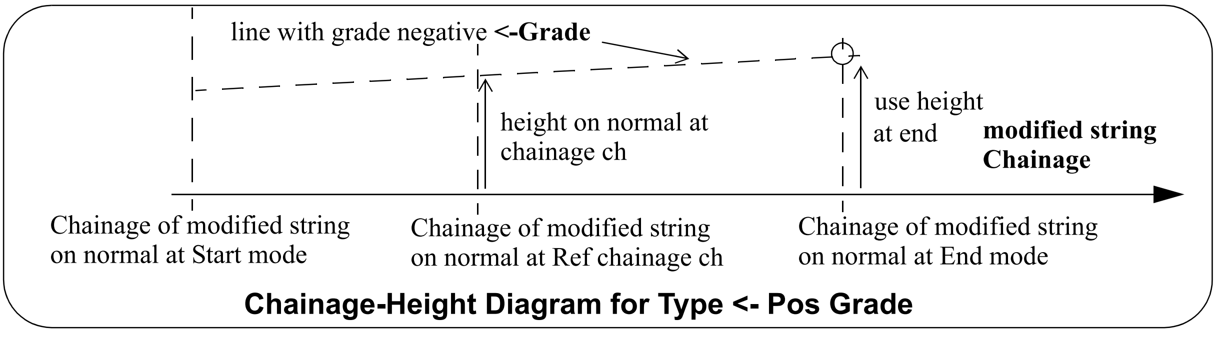
Important Note: **Grade->** is calculated using **increasing Alignment chainages.**

21.2.2.2.13.1.11 <- Pos Grade: all points are on a line of given grade and going through the end point



Let **End RL** be the actual height on the vertex on the normal at **End mode** when all the modifiers before this modifier are applied.

Then the height of each vertex on the normals between the *Start mode* and the *End Mode* is on the line with grade of negative **<-Grade** with respect to the modified string chainage, and going through the point (normal at End mode, End RL).



Chainage-Height Diagram for Type <- Pos Grade

Important Note: **<-Grade** is calculated using **decreasing Alignment chainages** and the **heights at the points** along the string being modified. So in a diagram with increasing chainage, the grade is negative **<-Grade**.

21.2.2.2.13.1.12 Pos-> Pos: Tangential Compound Parabolas

Let **Start RL** be the actual height calculated at the vertex of the string at the perpendicular to **Start mode** when all the modifiers **before this modifier** are applied.

Let **End RL** be the actual height calculated at the vertex of the string at the perpendicular to **End mode** when all the modifiers **before this modifier** are applied.

Then for each vertex on the perpendiculars between **Start mode** and **End Mode**, the height of the vertex is on the curve of tangential compound parabolas (in chainage-height space) between the *Start RL* and the *End RL*.

21.2.2.2.13.1.13 Pos-> Pos: Tangential Cubic

Let **Start RL** be the actual height calculated at the vertex of the string at the perpendicular to **Start mode** when all the modifiers **before this modifier** are applied.

Let **End RL** be the actual height calculated at the vertex of the string at the perpendicular to **End mode** when all the modifiers **before this modifier** are applied.

Then for each vertex on the perpendiculars between **Start mode** and **End Mode**, the height of the vertex is on the curve of tangential cubic (in chainage-height space) between the *Start RL* and the *End RL*.

21.2.2.2.14 Fixed - by 2 strings

MTF_Edit_Modify_Fixed___Modify_by_2_StringsThe **by 2 strings** option calculates one of the width, height and xfall or slope from two given strings.

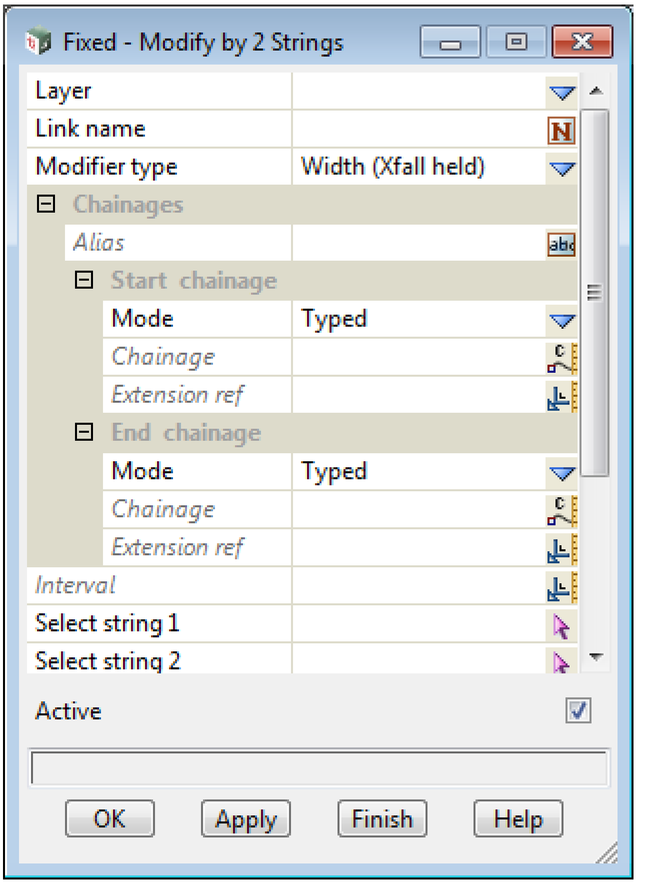
That is, for the current link:

(a) one of Width, Height or Xfall **keeps its existing value** (i.e. **holds** its **existing value)**.

and

(b) a different one of Width, Height or Xfall **is modified** and its value is taken form the differences of two selected strings.

Selecting the **by 2 strings** brings up the **Fixed - Modify by 2 Strings** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box	Design	available layers

name of the layer that the link to be modified is in.
For information on Layers, see 21.1.1.2 MTF Links and Layers.

Link name	name box		names.4d pop-up

the name of the link in the Layer to modify the width/height/xfall/slope using the method given in **Modifier type,** *between the chainages given by* **Start mode** *and* **End mode.**
This is the current link.

String 1	string-select		

select the first string to use for defining width/height/xfall for the link.

String 2	string-select		

select the second string to use for defining width/height/xfall for the link.

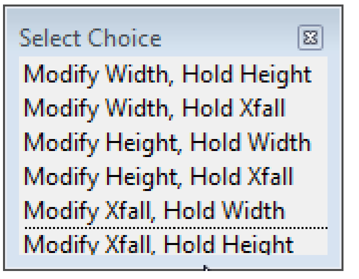
Side 1 to search choice box left side left side, right side, both sides

side of the hinge string to start searching to find string 1 to use in defining width/height/xfall.

Side 2 to search choice box left side left side, right side, both sides

side of the hinge string to start searching to find string 2 to use in defining width/height/xfall.

Modifier type choice box



modify Width and take Height from current link
modify Width and take Xfall from current link
modify Height and take Width from current link
modify Height and take Xfall from current link
modify Xfall and take With from current link
modify Xfall and take Height from current link

A link can be modified in any way regardless of how the link was defined in terms of width, height and xfall or slope. See 21.1.2.1 Calculating Width, Height, Xfall or Slope from Original Definitions

The ***Modify*** *width/height/xfall is the part of the current link that is modified from its current value by the method given by* **Modifier type**.

The ***Hold*** *width/height/xfall is the part of the current link that keeps its current value.*

For ***Modify Width****, see* 21.2.2.2.14.1 Fixed - by 2 Strings: Modifier Types Modify Width, Hold Height/Xfall
For ***Modify Height****, see* 21.2.2.2.14.2 Fixed - by 2 Strings: Modifier Types Modify Height, Hold Width/Xfall
For ***Modify Xfall****, see* 21.2.2.2.14.3 Fixed - by 2 Strings: Modifier Types Modify Xfall, Hold Width/Height

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages to calculate the values between the two strings.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

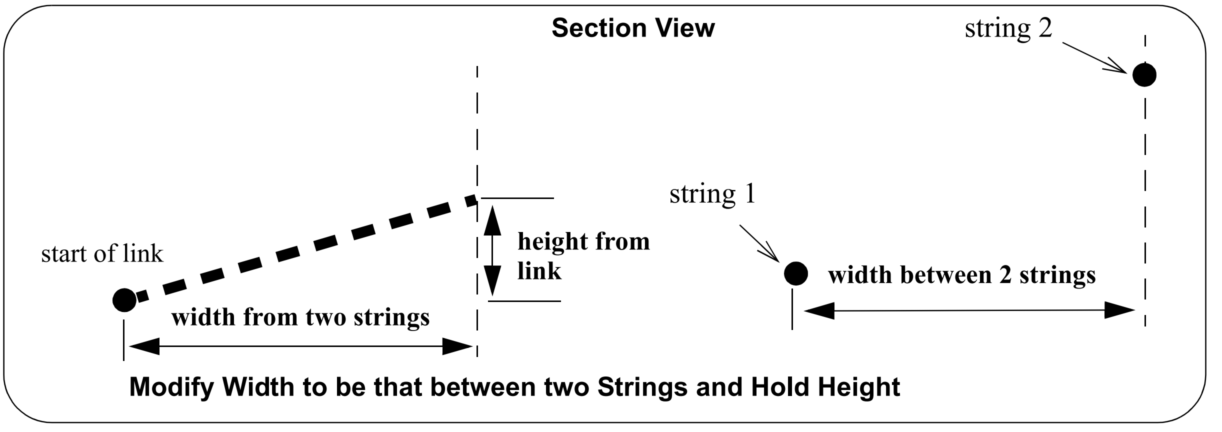
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Continue to the next section 21.2.2.2.15 Fixed - Interface to Tin or return to 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

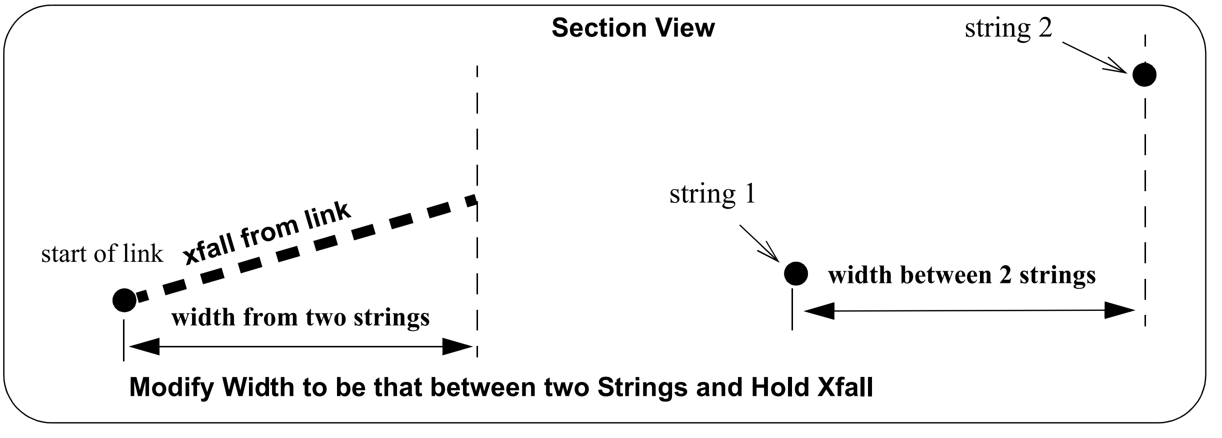
21.2.2.2.14.1 Fixed - by 2 Strings: Modifier Types Modify Width, Hold Height/ Xfall

For a fixed link **Modify Width** sets the **width** of the link to be the width between the two selected strings.

For **Modify Width, Hold Height** the *height* from the current link is kept.



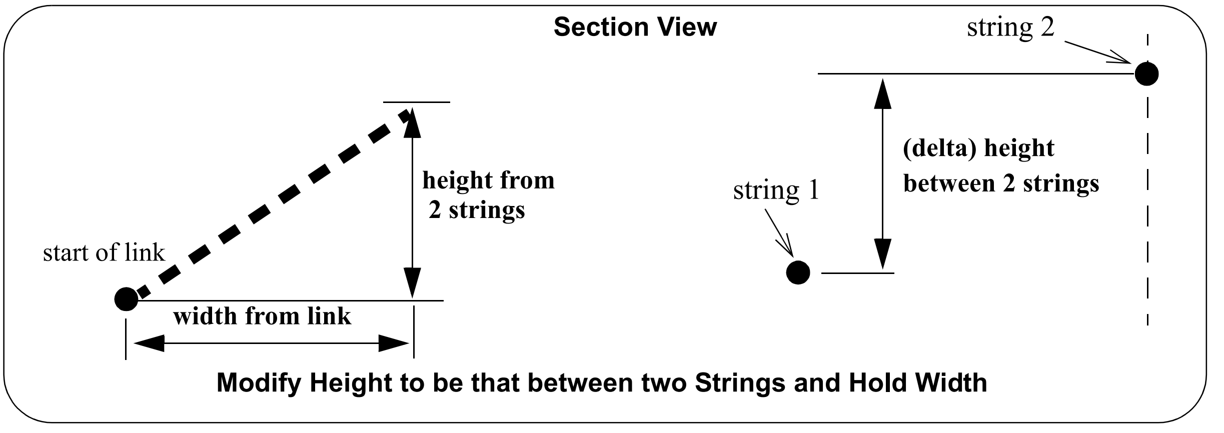
For **Modify Width, Hold Xfall**, the *xfall* from the current link is kept.



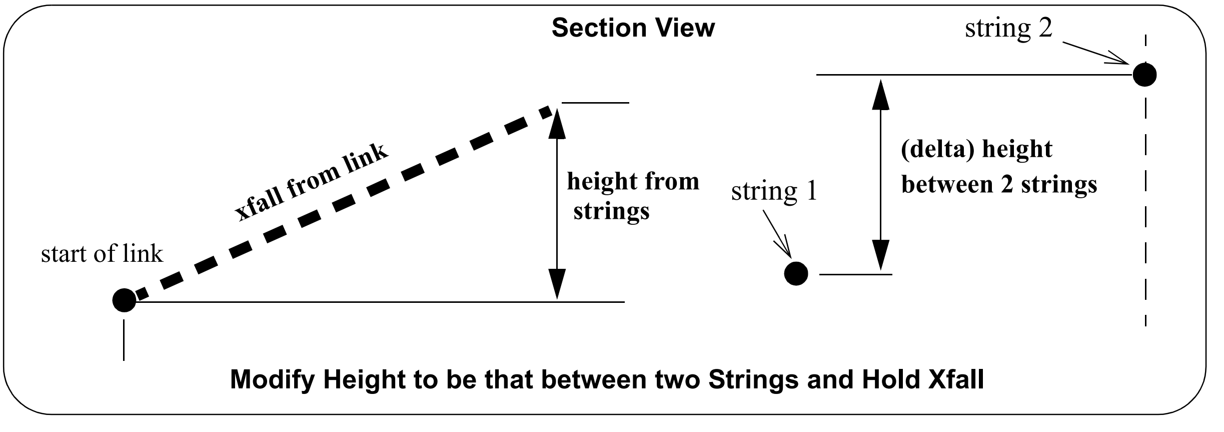
21.2.2.2.14.2 Fixed - by 2 Strings: Modifier Types Modify Height, Hold Width/ Xfall

For a fixed link **Modify Height** sets the **height** of the link to be the height between the two selected strings.

For **Modify Height, Hold Width** the *width* from the current link is kept.



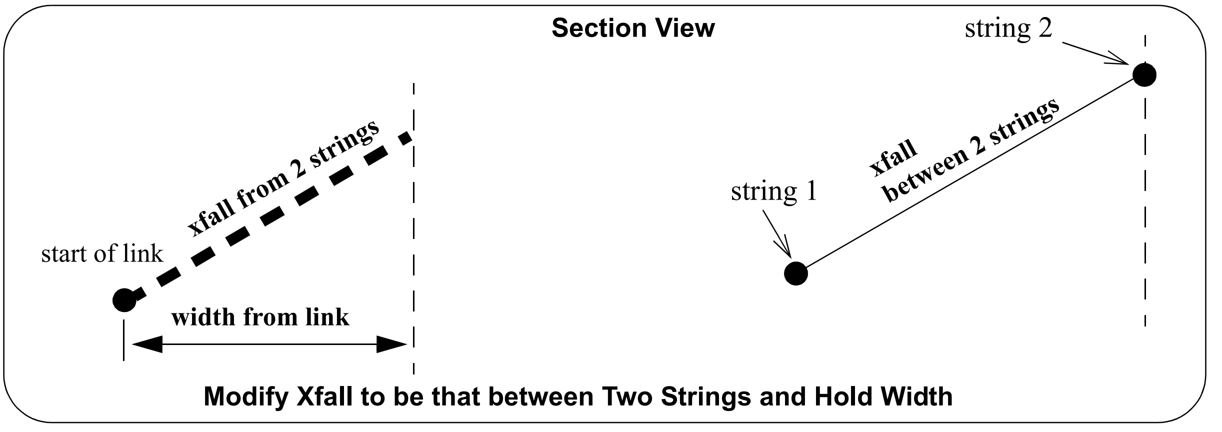
For **Modify Height, Hold Xfall** the *xfall* from the current link is kept.



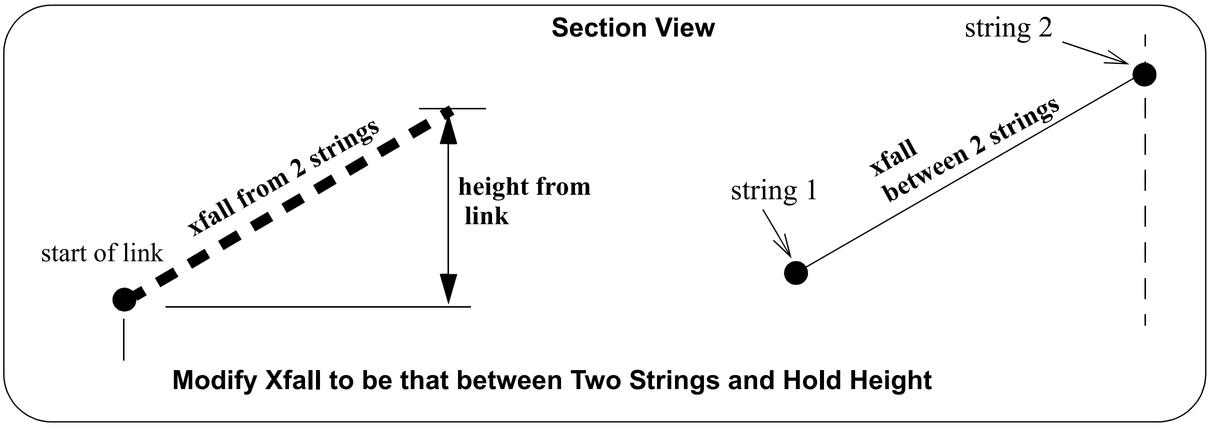
21.2.2.2.14.3 Fixed - by 2 Strings: Modifier Types Modify Xfall, Hold Width/ Height

For a fixed link **Modify Xfall** sets the **xfall** of the link to be the xfall between the two selected strings.

For **Modify Xfall, Hold Width** the *width* from the current link is kept.



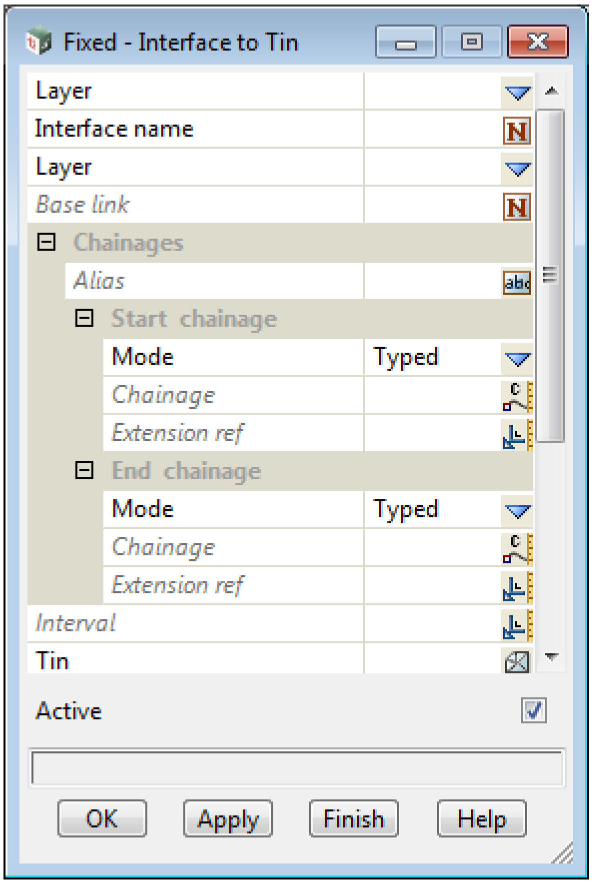
For **Modify Xfall, Hold Height** the *height* from the current link is kept.



21.2.2.2.15 Fixed - Interface to Tin

The **Fixed =>interface to string** option interfaces from the given link to a given tin dropped by a strip depth.

Selecting the **Interface to tin** brings up the **Fixed - Interface to tin** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box	Design	available layers

name of the layer that the link to interface off is in.

For information on Layers, see 21.1.1.2 MTF Links and Layers.

Field Description	Type	Defaults	Pop-Up
Base name	name box		names.4d pop-up

the name of the link in the layer **Layer** *to interface (batter) off.*
That is, a line is taken from the ***end*** *of the Base Link with either* **Final cut slope** *or* **Final fill slope**, *and extended until it cuts the stripped tin.*

Field Description	Type	Defaults	Pop-Up
Final cut slope 1 v in	real box	0	no slope, 0,1,2,3,4,5,10

the **cut slope** *is used if the Base link is in cut. That is, it is below the tin.*

The cut slope is the slope for the interface calculation to be done when the Base link is in cut. The batter continues at the cut slope until it hits the **Tin** *dropped by the* **Strip** *depth, or until it reaches the* **Maximum slope width**.

A cut slope of one vertical to the given value of horizontal units is used. The value 0 is used to designate a ***horizontal*** *slope - vertical slopes are not allowed.*

For ***final cut slope***, ***positive is up*** *and negative down*

Final fill slope 1 v in real box 0 no slope, 0,1,2,3,4,5,10

the **fill slope** *is used if the Base link is in cut. That is, it is above the tin.*

The fill slope is the slope to be used in the interface calculation when the Base link is in fill. The batter continues at the fill slope until it hits the **Tin** *dropped by the* **Strip** *depth, or until it reaches the* **Maximum slope width**.

A fill slope of one vertical to the given value of horizontal units is used. The value 0 is used to designate a ***horizontal*** *slope - vertical slopes are not allowed.*

For ***final fill slope***, ***positive is down*** *and negative is up.*

This definition of fill slope being positive when going down is used so that the value in the Final fill slope 1 v in *field is normally positive.*

Maximum slope width real box 100

the maximum width for the final slope. It it hasn't reached the tin by this distance then it stops.

Strip real box 0

the depth to vertically drop the tin before interfacing.

Tin tin box available tins

name of the tin to interface to.

Layer layer box Design available layers

name of the layer that the created link is added to.

Interface name name box

name for the link created by this link by battering.

Hence the end of string lies on the stripped tin. The string formed from the end of this link is called the ***interface*** *string.*

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages to calculate interface

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

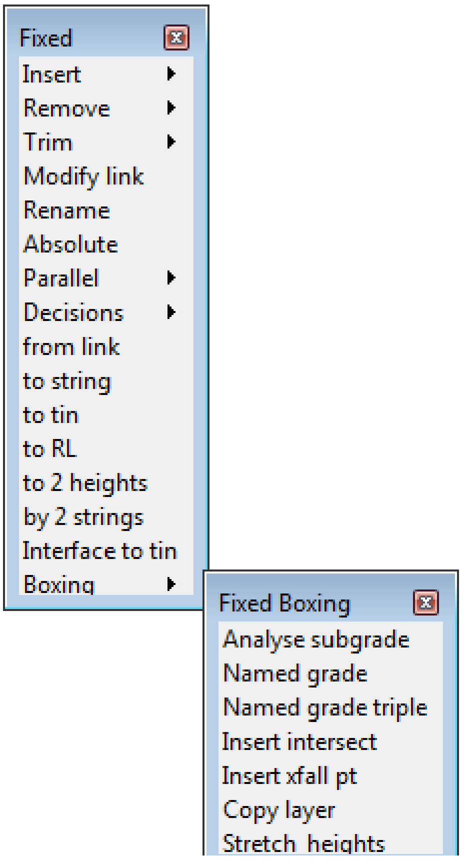
Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Go to the next section 21.2.2.2.16 Fixed - Boxing or return to 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

21.2.2.2.16 Fixed - Boxing

fixed_boxing



The Fixed =>Boxing bring the power of the **Boxing** from **Templates** to the **Left/Right Modifiers**.

See

21.2.2.2.16.1 Boxing - Analyse Subgrade

21.2.2.2.16.2 Boxing - Named Grade

21.2.2.2.16.3 Boxing - Named Grade Triple

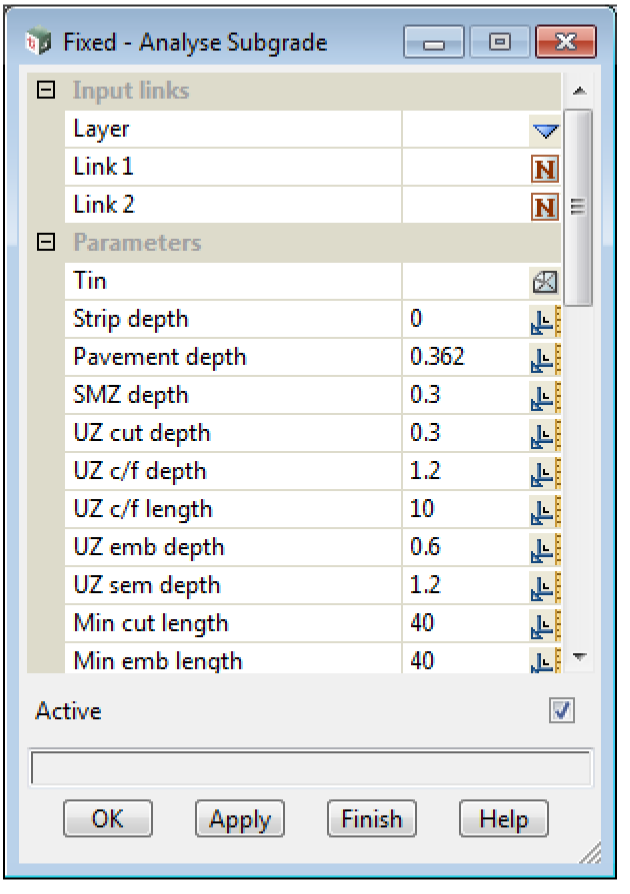
21.2.2.2.16.4 Boxing - Insert Intersect

21.2.2.2.16.5 Boxing - Insert Xfall Pt

21.2.2.2.16.6 Boxing - Copy Layer

21.2.2.2.16.1 Boxing - Analyse Subgrade

Selecting **Analyse subgrade** brings up the panel **Fixed - Analyse Subgrade**



Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start and end chainages to analyse subgrade for.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

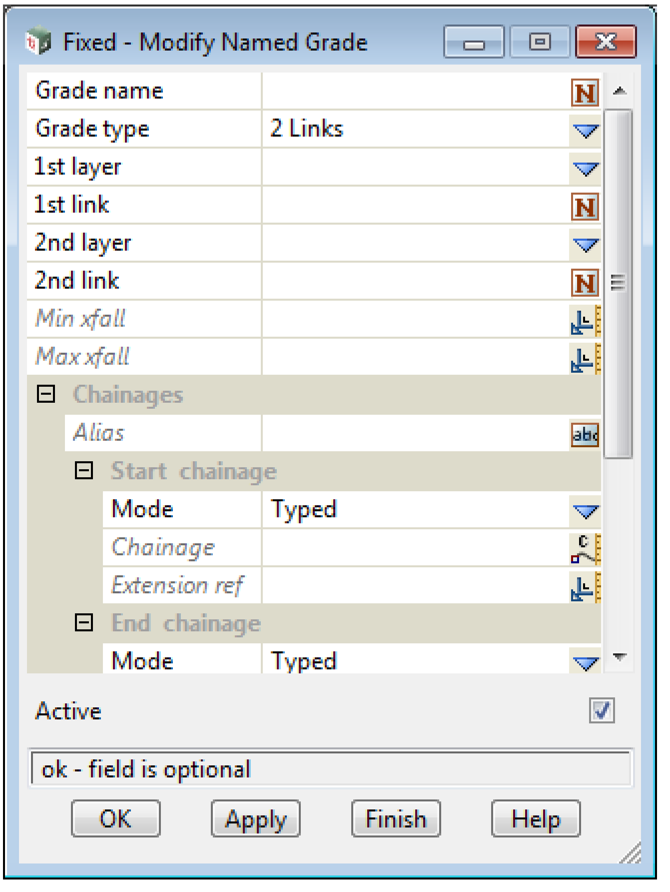
Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Go to the next section 21.2.2.2.16.2 Boxing - Named Grade or return to 21.2.2.2.16 Fixed - Boxing or 21.2.2.2 Fixed Link Modifiers.

21.2.2.2.16.2 Boxing - Named Grade

Selecting **Named grade** brings up the panel **Fixed - Modify Named Grade**



Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start and end chainages that the modifier exists between.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

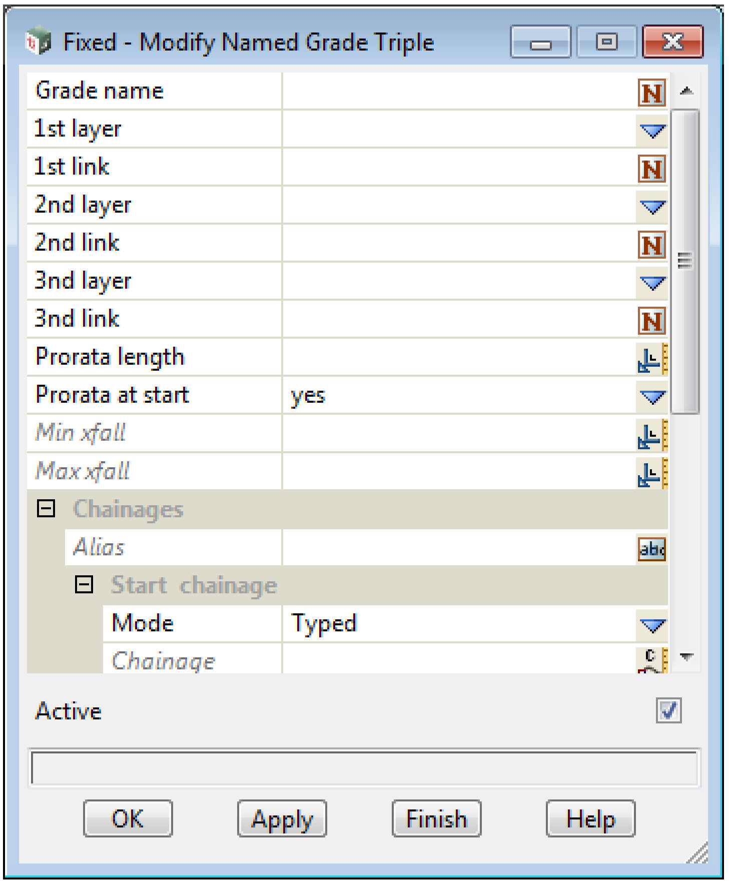
Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Go to the next section 21.2.2.2.16.3 Boxing - Named Grade Triple or return to 21.2.2.2.16 Fixed - Boxing or 21.2.2.2 Fixed Link Modifiers.

21.2.2.2.16.3 Boxing - Named Grade Triple

Selecting **Named grade triple** brings up the panel **Fixed - Modify Named Grade Triple**



Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start and end chainages that the modifier exists between.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

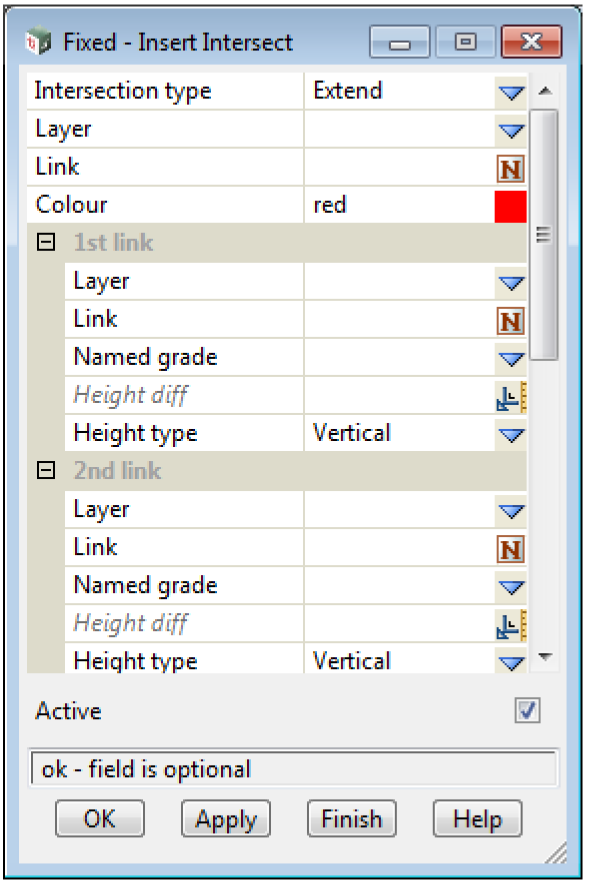
Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Go to the next section 21.2.2.2.16.4 Boxing - Insert Intersect or return to 21.2.2.2.16 Fixed - Boxing or 21.2.2.2 Fixed Link Modifiers.

21.2.2.2.16.4 Boxing - Insert Intersect

Selecting **Insert intersect** brings up the panel **Fixed - Insert Intersect**



Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start and end chainages that the modifier exists between.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

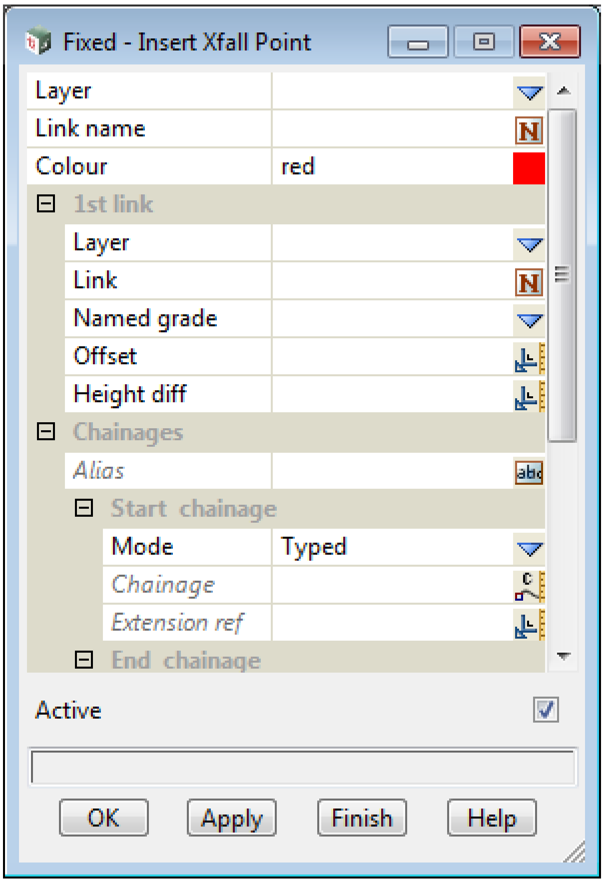
Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Go to the next section 21.2.2.2.16.5 Boxing - Insert Xfall Pt or return to 21.2.2.2.16 Fixed - Boxing or 21.2.2.2 Fixed Link Modifiers.

21.2.2.2.16.5 Boxing - Insert Xfall Pt

Selecting **Insert xfall pt** brings up the panel **Fixed - Insert Xfall Point**



Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start and end chainages that the modifier exists between.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Go to the next section 21.2.2.2.16.6 Boxing - Copy Layer or return to 21.2.2.2.16 Fixed - Boxing or 21.2.2.2 Fixed Link Modifiers.

21.2.2.2.16.6 Boxing - Copy Layer

The **Copy layer** command copies the **Link Points** from part of one layer to another layer, with an optional value **Height diff** added to the copied points.

That is, the ***Copy layer*** command copies vertices from part of a selected **Layer** (from the vertex at the end of a **Start link** to the vertex at the end of an **End link**) and copies the vertices to a given **Layer**, with an optional value **Height diff** added to the copied vertices.

The option works by

(a) Calculating for each of the **vertices** from the vertex at the end of the **Start link** to the vertex at the end of the **End link,** the offset and height from the **Hinge Point** (the beginning of the layer).

That is, calculate an offset and delta height from the Hinge Point rather than just relative to the previous link.

(b) Adding the value **Height diff** to each of the height of each vertex.

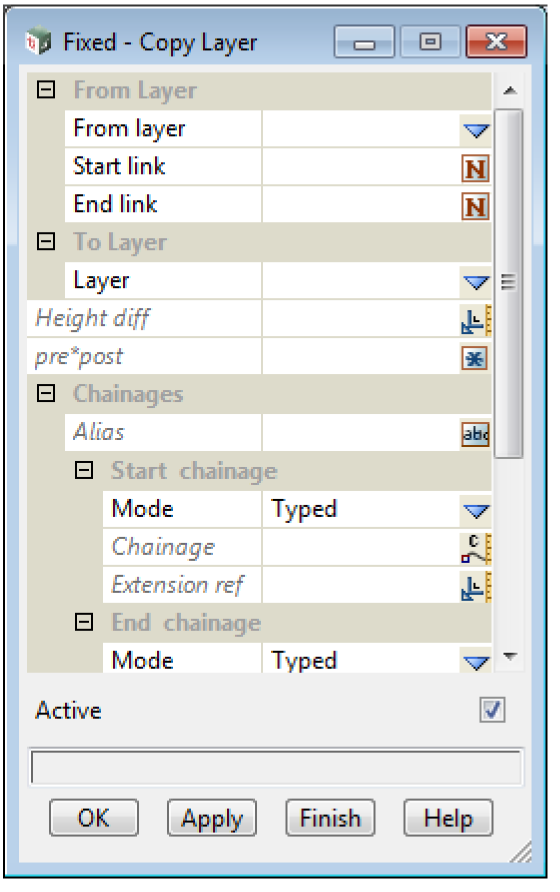
(c) Inserting the vertices into the **To Layer** using the calculated offsets and heights from the Hinge Point for each vertex.

Hence if **no links** already **exist** in the **To Layer** for the **calculated offset range** of the **copied vertices**, then the cross sectional shape of the copied links is maintained.

However if **links** already **exist** in the **To Layer** for the **calculated offset range** of the **copied vertices**, then the **existing** and **copied vertices** are **merged** together. The merging occurs by simply ordering the vertices by increasing offset and then joining adjacent vertices together. The cross sectional shape of the copied links may not be maintained.

Note that the **start vertex** of the **Start link** is not copied. This is normally not a problem unless you are trying to copy the first vertex of the first Link. To do this with the **Copy layer** command, you would need to add a new link with a width and height of zero at the beginning of the string. This new link would then be used as the **Start link.**

Selecting **Copy layer** brings up the panel **Fixed - Copy Layer**



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
From layer	layer box	Design	available layers

name of the layer to copy links from.
For information on Layers, see 21.1.1.2 MTF Links and Layers.

Start link link box

name of the link in **From layer** *to start copying from.*

End link link box

name of the link in **From layer** *to stop copying from.*

Layer	layer box	Design	available layers

name of the layer to copy links from.
For information on Layers, see 21.1.1.2 MTF Links and Layers.

Height diff real box

the value to add to the height of the vertices of the links from the **Start link** *to the* **End link.**

Pre*post text input

each copied vertex is given a new name by apply the **Pre*pos** *text to the original name of the vertex.*

*That is, the text before the * is prepended to the vertex name, and the text after the * is postpended to the vertex name.*

Alias, **Start Chainage**, **End Chainage**, **Interval**

defines the start and end chainages to copy layer points.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF

Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

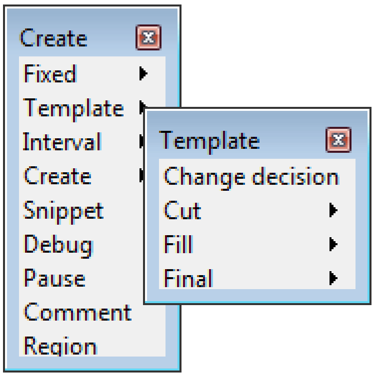
Return to 21.2.2.2.16 Fixed - Boxing or 21.2.2.2 Fixed Link Modifiers.

21.2.2.3 Template Modifiers

Each **12d Template** is made up of the sections *Fixed*, *Decisions*, *Cut*, *Fill* and *Final Cut/Fill*. See 20.1 Templates in 12d Model.

The options in the **MTF Template** menu are for working with the **Decisions**, **Cut**, **Fill** and **Final** Cut/Fill sections of inserted Templates.

The **Template** walk-right brings up the **Template** menu.



For

Change decision	21.2.2.3.1 Change Decision
Cut	21.2.2.3.2 Cut Link Modifiers
Cut	21.2.2.3.3 Fill Link Modifiers
Final	21.2.2.3.4 Final Modifiers

21.2.2.3.1 Change Decision

If a 12d Template has a **Decisions** section then the **Decisions** section is used before the **Cut**, **Fill** and **Final Cut/Fill** sections.

For an **Apply MTF**, a Template specified for a chainage range in the **Templates** section of the MTF (see 21.2.6.1 MTF Templates) is called the *original* template for that chainage range.

Using the **Change Decision** modifier, for a given chainage range it is possible to ***override*** the **Decisions** section from the original Template and:

(a) use the **Decisions** section **from another** Template instead of the **Decisions** section of the original Template.

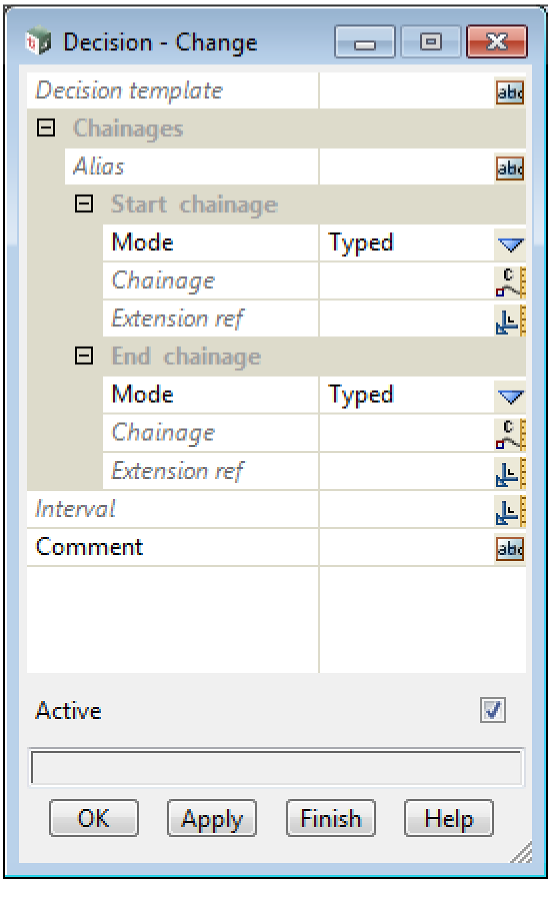
or

(b) use the cut, fill and final cut/fill sections of the original template **instead** of the **Decisions** section of the original template.

So the **Change Decision** option can be used to substitute a **Decisions** section from a different Template, or just **turn off** the **Decisions** section of the original Template.

In either case, the **Fixed** section of the *original* Template is ***still used.***

Selecting **Change decision** brings up the **Decision - Change** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Decision template	input		available templates

*if **non-blank**, this is the Template whose Decisions section is used over the given chainage range. If **blank**, the Cut, Fill and Final Cut/Fill sections of the original Template are used over the chainage range instead of the Decision section of the original Template.*

Note that if the new Template has no decisions, then the ***Cut and Fill section*** *of the original Template is used instead of the* ***Decisions*** *section of the original Template. That is, it is equivalent to leaving the* ***Decisions Template*** *field blank.*

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages to changes/stop the Template Decisions.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

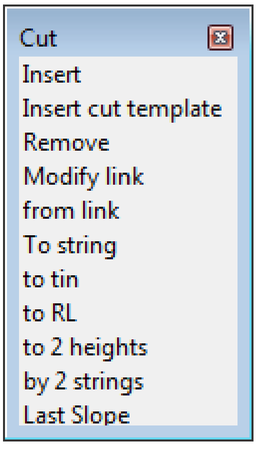
Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Return to 21.2.2.3 Template Modifiers.

21.2.2.3.2 Cut Link Modifiers

The **Cut** walk-right brings up the **Cut** menu with options to modify the cut links of the template.



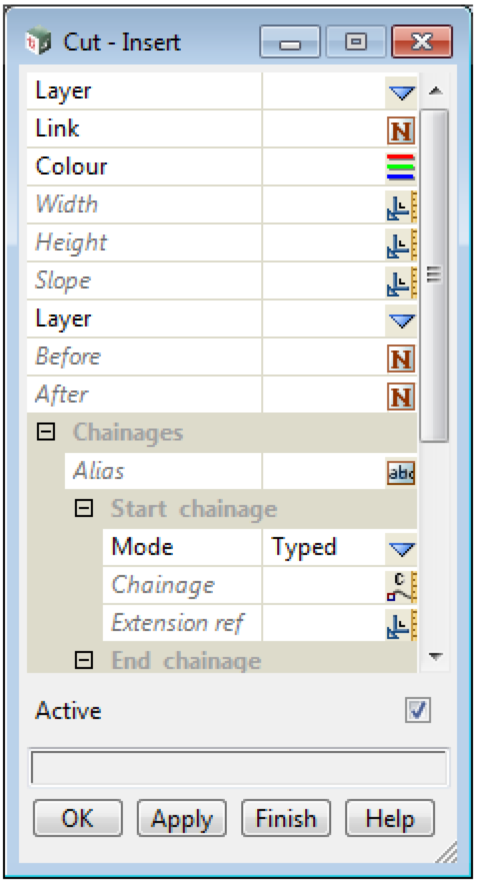
See

Insert	21.2.2.3.2.1 Cut Link - Insert
Insert cut template	21.3.0.1 Insert Cut Links from a Template
Remove	21.3.0.2 Cut Link - Remove
Width/Height	21.3.0.3 Cut Link - Modifier Width or Height
Slope	21.3.0.4 Cut Link - Modify Slope
from link	21.3.0.5 Cut Link - from Link
to string	21.3.0.6 Cut Link - to String
to tin	21.3.0.7 Cut Link - to Tin
to 2 heights	21.3.0.8 Cut Link - to Two Heights
by 2 strings	21.3.0.9 Cut Link - to 2 Strings21.3.0.9 Cut Link - to 2 Strings

21.2.2.3.2.1 Cut Link - Insert

Cut links can be created by the **cut Insert** command by specifying either the *width and height*, *width and slope* or *height and slope* for the link.

Selecting **Insert** brings up the **Cut - Insert** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	choice box		available Layers

Layer for the new link to go into.
For information on Layers, see 21.1.1.2 MTF Links and Layers.

Link name	input		select name menu

name of the cut link to create in the Layer.

Colour	input		available colours

colour of the link being created.

Width /height/slope	input		measures menu

width/ height/slope of the link being created - only use two of the three.

Before	choice box		select name menu

*if **not blank**, the name of the link in the Layer to insert the new link before.*
*If **blank** and **After** is **blank**, the link is appended to the end of the fixed part of the template.*

After	choice box		select name menu

*if **Before** is **not blank**, **After** is ignored.*

*If **not blank** and **Before** is **blank**, the name of the link in the layer to insert the new link after. If **blank** and **Before** is **blank**, **Before** is used.*

***Note**: Only one of **Before** or **After** can not be blank. If they are both blank then **Before** takes precedence over **After**.*

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages to changes/stop the Template Decisions.

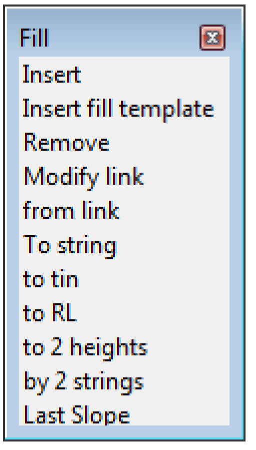
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels**.**

21.2.2.3.3 Fill Link Modifiers

The **Fill** walk-right brings up the **Fill** menu with options to modify the fill links of the template.



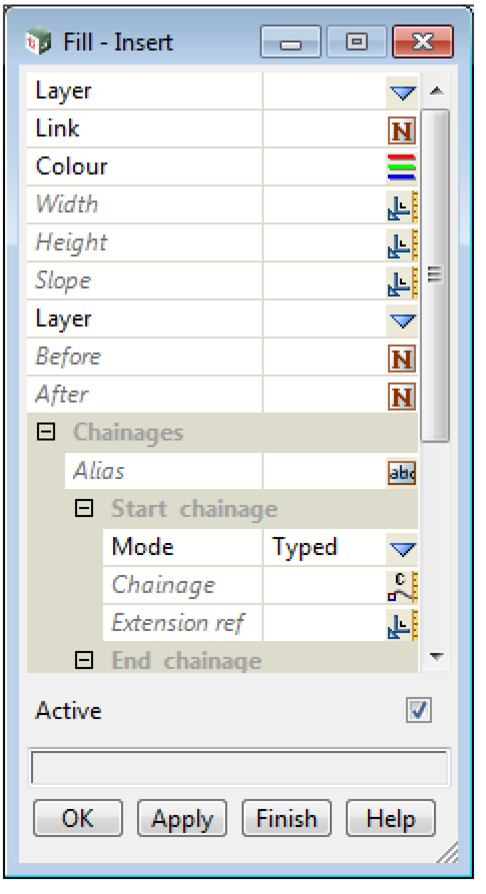
See

Insert	21.2.2.3.3.1 Fill Link - Insert
Insert fill template	21.3.0.10 Insert Fill Links from a Template
Remove	21.3.0.11 Fill Link - Remove
Width/Height	21.3.0.12 Fill Link - Modify Width or Height
Slope	21.3.0.13 Fill Link - Modify Slope
from link	21.3.0.14 Fill Link - from Link
to string	21.3.0.15 Fill Link - to String
to tin	21.3.0.16 Fill Link - to Tin
to 2 heights	21.3.0.17 Fill Link - to Two Heights
by 2 strings	21.3.0.18 Fill Link - to Two Strings

21.2.2.3.3.1 Fill Link - Insert

Fill links can be created by the **fill insert** modifier and by specifying either the width and height, width and slope or height and slope.

Selecting **Insert** brings up the **Fill - Insert** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	choice box		available Layers

Layer for the new link to go into.
For information on Layers, see 21.1.1.2 MTF Links and Layers.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu

name of the fill link to create in the given Layer.

Field Description	Type	Defaults	Pop-Up
Colour	input		available colours

colour of the link being created.

Field Description	Type	Defaults	Pop-Up
Width /height/slope	input		measures menu

width/ height/slope of the link being created - only use two of the three.

Field Description	Type	Defaults	Pop-Up
Before	choice box		select name menu

*if **not blank**, the name of the link in the Layer to insert the new link before.*
*If **blank** and **After** is **blank**, the link is appended to the end of the fixed part of the template.*

Field Description	Type	Defaults	Pop-Up
After	choice box		select name menu

*if **Before** is **not blank**, **After** is ignored.*

If ***not blank*** *and* ***Before*** *is* ***blank****, the name of the link in the layer to insert the new link after.*
If ***blank*** *and* ***Before*** *is* ***blank****,* ***Before*** *is used.*

Note*: Only one of* ***Before*** *or* ***After*** *can not be blank. If they are both blank then* ***Before*** *takes precedence over* ***After****.*

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages to changes/stop the Template Decisions.

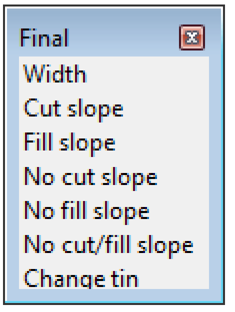
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels**.**

21.2.2.3.4 Final Modifiers

The **Final** walk-right brings up the **Final** menu with options to modify the final cut and fill links of the template.



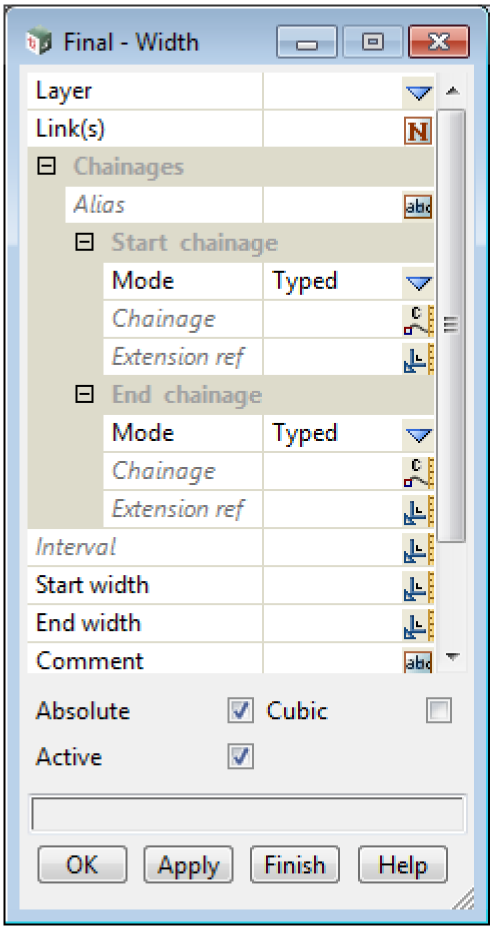
See

Width 21.2.2.3.4.1 Final Link - Width
Cut slope 21.2.2.3.4.2 Final Link - Cut or Final Fill Slope
Fill slope 21.2.2.3.4.2 Final Link - Cut or Final Fill Slope
No cut slope 21.2.2.3.4.3 Final Link - No Cut Slope or Final No Fill Slope or Final No Cut/Fill Slope
No fill slope 21.2.2.3.4.3 Final Link - No Cut Slope or Final No Fill Slope or Final No Cut/Fill Slope
No cut/fill slope 21.2.2.3.4.3 Final Link - No Cut Slope or Final No Fill Slope or Final No Cut/Fill Slope
Change tin 21.2.2.3.4.4 Change Final Tin
Final link name 21.2.2.3.4.5 Change Final Link Name

21.2.2.3.4.1 Final Link - Width

The *final* **Width** modifier varies the **Maximum slope width** of the *final* link of the template.

Selecting **Width** brings up the **Final - Width** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box	Design	available layers

name of the layer that the link to modify is in.

For information on Layers, see 21.1.1.2 MTF Links and Layers.

Link name	link box		select name menu

name of the final link to modify.

Start width	input		measures menu

width at the Start chainage for the **Maximum slope width** *of the final link of the template.*

End width	input		measures menu

width at the End chainage for the **Maximum slope width** *of the final link of the template.*

Cubic	tick box		

if ***ticked****, the width is varied as a reverse cubic between the start and end chainages.*
if ***not ticked****, the width is varied linearly between the start and end chainages.*

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages to modify the maximum width of the final link.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

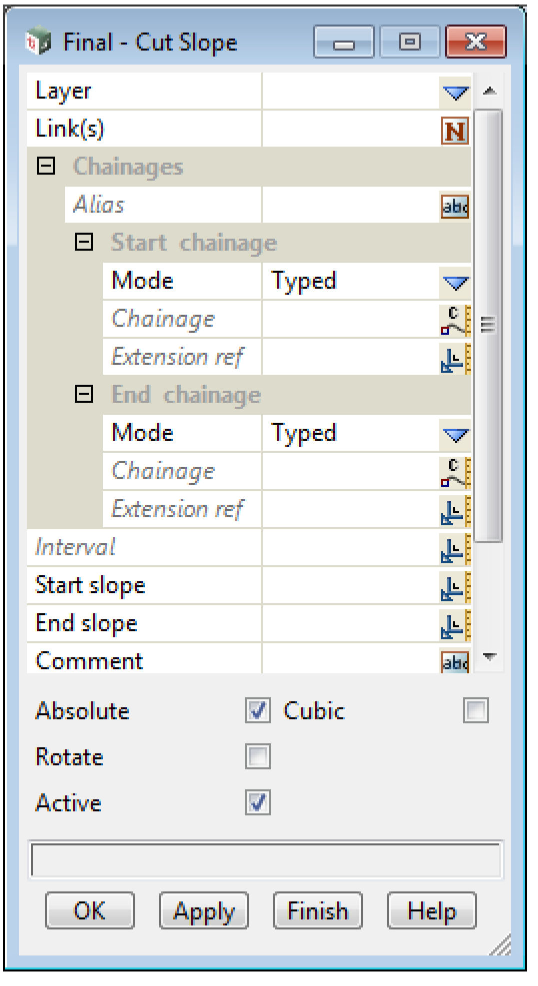
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

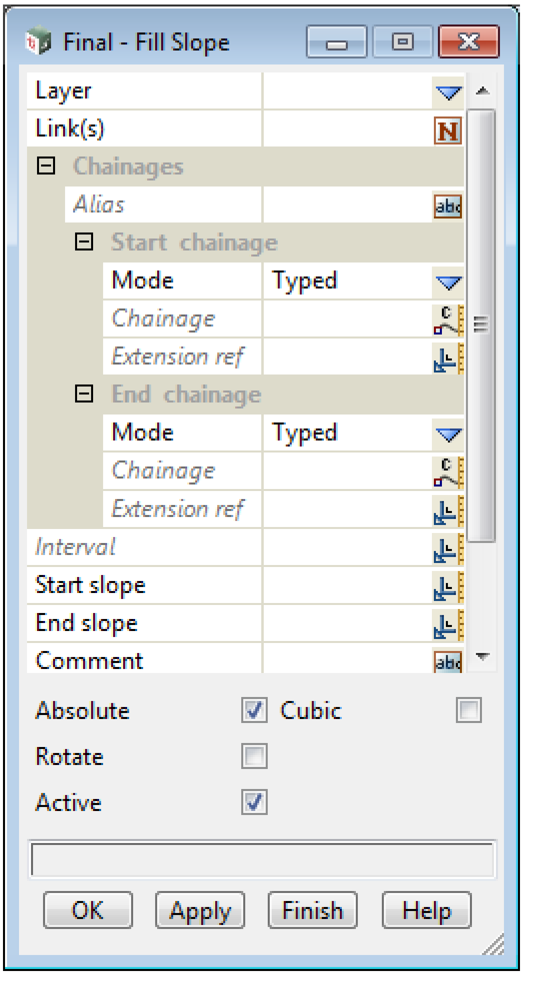
Continue to the next section 21.2.2.3.4.2 Final Link - Cut or Final Fill Slope or return to 21.2.2.3.4 Final Modifiers.

21.2.2.3.4.2 Final Link - Cut or Final Fill Slope

The final **Cut slope** modifier varies the cut slope of the *final* link of the template and the final **Fill slope** modifier varies the fill slope of the *final* link of the template.

Selecting the **Cut slope** or **Fill slope** options brings up the **Final - Cut Slope** and **Final - Fill Slope** panels respectively.





The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box	Design	available layers

name of the layer that the link to modify is in.

For information on Layers, see 21.1.1.2 MTF Links and Layers.

Link name	link box		select name menu

name of the final link to modify.

Start slope	input		measures menu

slope at the Start chainage for the **Final cut/fill slope 1v in** *of the final link of the template.*

End slope	input		measures menu

slope at the End chainage for the **Final cut/fill slope 1v in** *of the final link of the template.*

Important note:

*For a **cut** link, **positive** slope is **up**.*

*For a **fill** link, **positive** slope is **down**.*

Cubic	tick box		

*if **ticked**, the slope is varied as a reverse cubic between the start and end chainages.*

Rotate tick box

*if **ticked**, the slope is varied linearly **with respect to** the **angle,** between the start and end chainages.*

***Note** -*

*Only one of cubic and rotate can be set to **ticked.***
*If both are set to **not ticked**, then the slope is varied **linearly with respect to slope** between the start and end chainages (the default).*

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages to modify the final slope.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

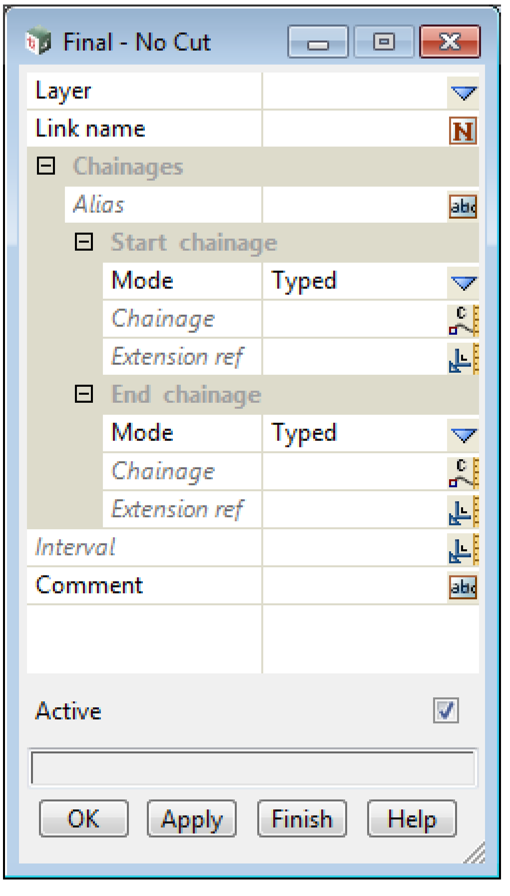
Continue to the next section 21.2.2.3.4.3 Final Link - No Cut Slope or Final No Fill Slope or Final No Cut/Fill Slope or return to 21.2.2.3.4 Final Modifiers.

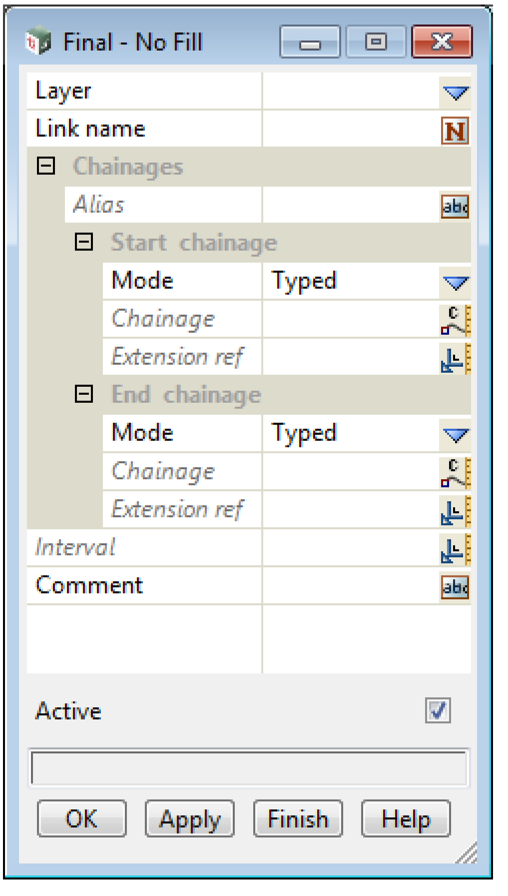
21.2.2.3.4.3 Final Link - No Cut Slope or Final No Fill Slope or Final No Cut/Fill Slope

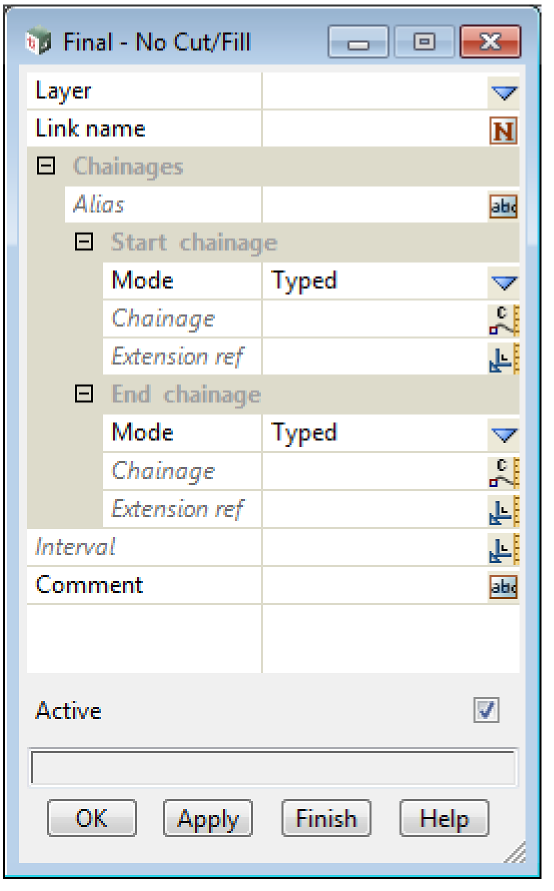
The final link can be stopped altogether or only stopped when it is in cut or only stopped when it is in fill.

The **No cut slope** modifier stops the final link when it is in *cut,* the **No fill slope** modifier stops the final link when it is in *fill* and the **No cut/fill fill slope** modifier stops the final link is any situation.

Selecting **No cut slope, No fill slope** or **No cut/fill slope** brings up the **Final - No Cut**, **Final - No Fill** or **Final - No Cut/Fill** panels respectively.







The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer	layer box	Design	available layers

name of the layer that the link to modify is in.

For information on Layers, see 21.1.1.2 MTF Links and Layers.

Link name	link box		select name menu

name of the final link to stop.

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages to stop the final link.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

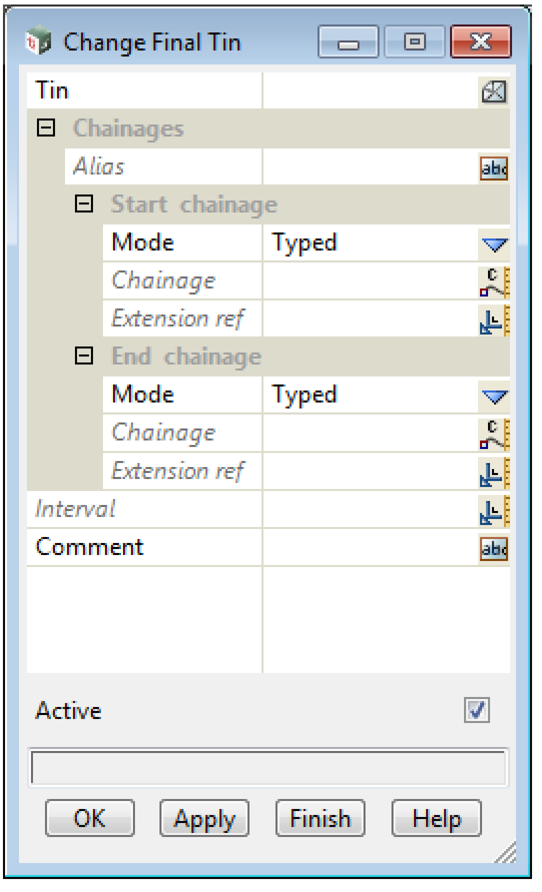
Continue to the next section 21.2.2.3.4.4 Change Final Tin or return to 21.2.2.3.4 Final Modifiers.

21.2.2.3.4.4 Change Final Tin

The **Change tin** modifier changes the tin that the final cut and fill slopes batter to.

Note that the default tin is the tin from the Apply MTF.

Selecting **Change tin** brings up the **Change Final Tin** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin for the final cut and fill links to batter to.

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages to change the tin that the final link to batters to.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

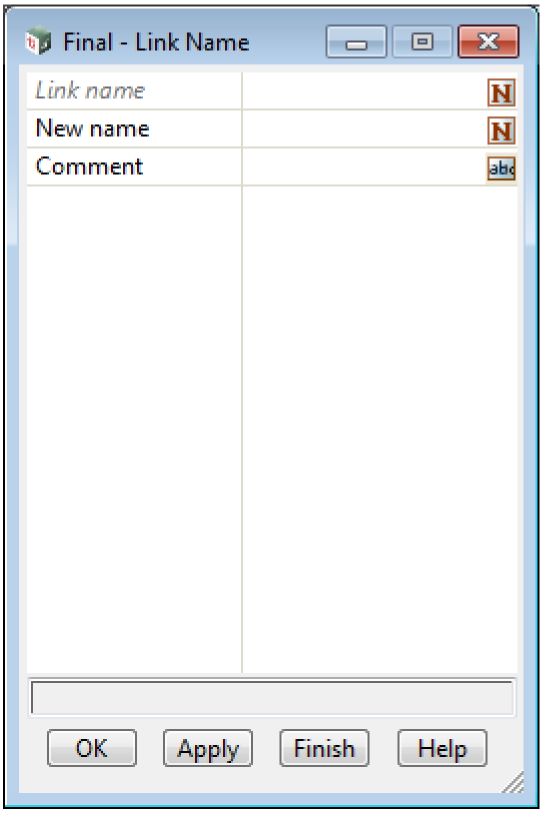
Continue to the next section 21.2.2.3.4.4 Change Final Tin or return to 21.2.2.3.4 Final Modifiers.

21.2.2.3.4.5 Change Final Link Name

NOTE: THIS OPTION HAS NOT BEEN IMPLEMENTED.

The **Final Link Name** modifier changes the name of the interface string.

Selecting **Final link name** brings up the **Final - Link Name** panel.



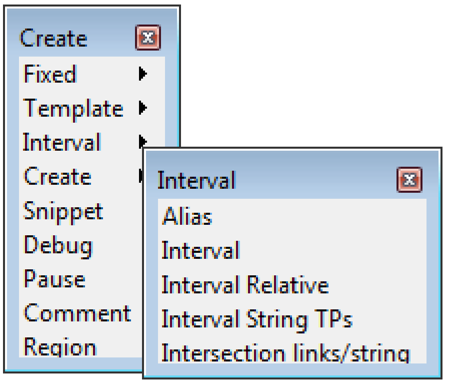
The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	name box		available names
current name of the interface string.			
New name	name box		available names
new name of the interface string.			

Return to 21.2.2.3.4 Final Modifiers.

21.2.2.4 Interval

The **Create** walk-right brings up the **Interval** menu.



For

Alias	21.2.2.4.1 Chainage Alias
Interval	21.2.2.4.2 Change Interval
Interval relative	21.2.2.4.3 Change Interval Relative
Interval String TPs	21.2.2.4.4 Interval String Tangent Points
Intersection links/string	21.2.2.4.5 Intersect Links with String

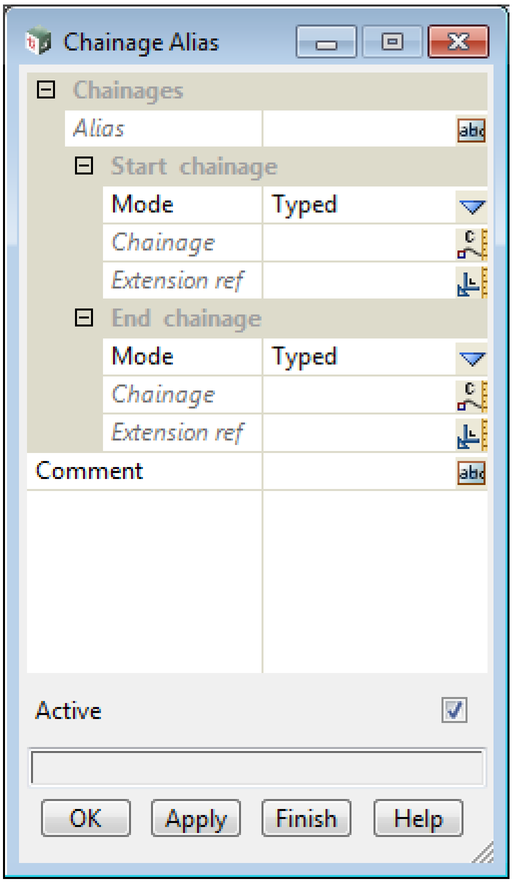
21.2.2.4.1 Chainage Alias

The **Alias** option creates a row in the grid with a given **Alias** name.

The *Alias* is then available to be used by *Smart Chainages*.

For the new information on **Smart Chainage,** see 21.4 Smart Chainages.

Selecting **Alias** brings up the panel **Chainage Alias**



Alias text box

a text name that can be used to refer to the row in the grid for this command.
See 21.2.2.1.4 Alias for MTF Modifiers

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages to define an Alias for this row.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Go to the next section 21.2.2.4.2 Change Interval or return to 21.2.2.4 Interval.

21.2.2.4.2 Change Interval

The **Interval** option changes the interval between the MTF sections over the given chainage range.

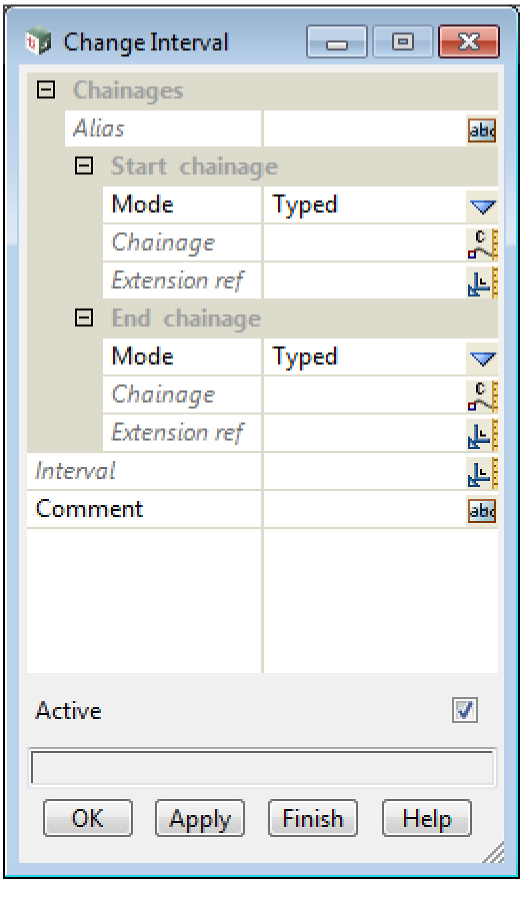
The sections are at the chainages that are integer multiples of the interval, and less than or equal to the start chainage and less than the end chainage. That is, the chainage values are referenced about zero and inside the start/end range.

For example, if the interval is 10 and the start chainage is -13 and the end chainage is 24 then there will be sections as

-10, 0, 10 and 20

The default value used for Interval is the **Separation distance** in the **Apply MTF** panel.

Selecting **Interval** brings up the **Change Interval** panel



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Alias, **Start Chainage**, **End Chainge**

defines the start and end chainages to change the interval between.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Interval input

*if **non blank**, the interval to use to create extra cross sections and strings over the given chainage range.*

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF

Modifier Panels.

Go to the next section 21.2.2.4.3 Change Interval Relative or return to 21.2.2.4 Interval.

21.2.2.4.3 Change Interval Relative

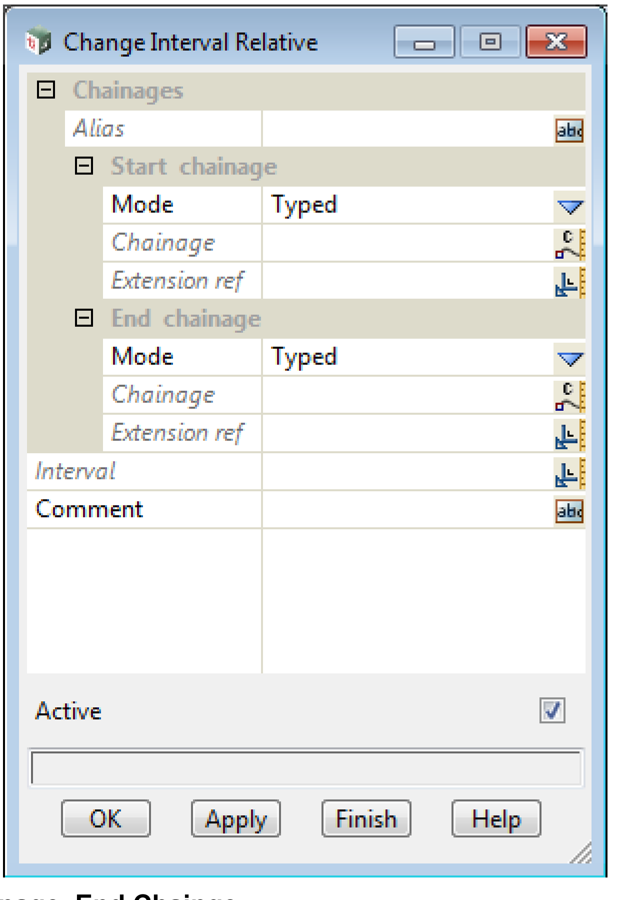
The **Interval Relative** option changes the interval between the MTF sections over the given chainage range but instead of being referenced about zero, the chainages are reference about the **start chainage**.

That is, the sections are at the chainages that are the start chainage plus integer multiples of the interval, and less than or equal to the start chainage and less than the end chainage. That is, the chainage values are referenced about the start chainage and inside the start/end range.

For example, if the interval is 10 and the start chainage is -13 and the end chainage is 24 then there will be sections as

-13, -3, 7 and 17

Selecting **Interval relative** brings up the **Change Interval Relative** panel



Alias, **Start Chainage**, **End Chainage**

defines the start and end chainages to change the interval between.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Interval input

*if **non blank**, the interval to use to create extra cross sections and strings over the given chainage range. The values for chainage are referenced about the Start chainage.*

Comment, **Extra start**, **Extra End, Active, OK, Apply**

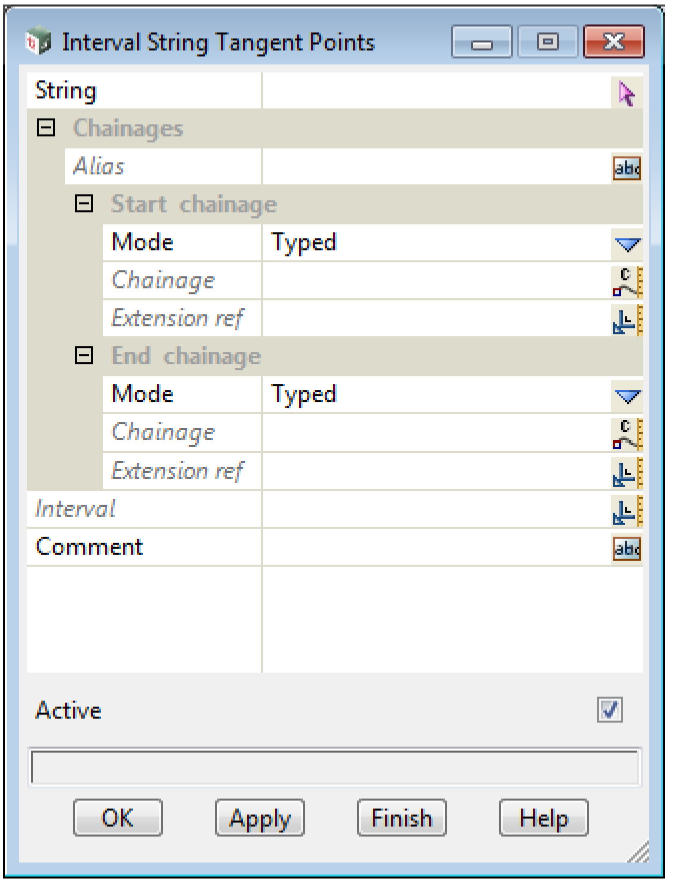
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Go to the next section 21.2.2.4.4 Interval String Tangent Points or return to 21.2.2.4 Interval.

21.2.2.4.4 Interval String Tangent Points

The **Interval String TPs** option drops the tangents points of a selected string onto the reference string and includes the dropped points as extra chainages where sections are generated.

Selecting **Interval string TPs** brings up the **Interval String Tangent Points** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
String	string select		

the string to select whose tangent points are dropped onto the reference string and if any of the dropped points are in the start and end chainage range then sections are generated at those dropped chainages.

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start and end chainages to drop tangent points.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

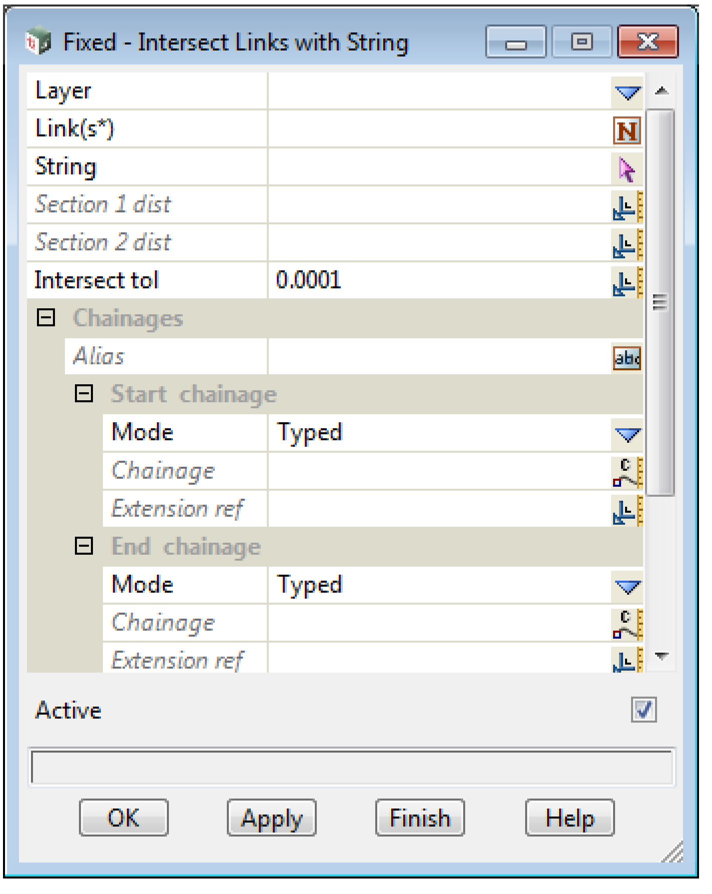
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Continue to next section 21.2.2.4.5 Intersect Links with String or return to 21.2.2.4 Interval.

21.2.2.4.5 Intersect Links with String

The **Intersection links/string** option

Selecting **Intersection links/string** brings up the **Fixed - Intersect Links with String** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
String	string select		

the string to select whose tangent points are dropped onto the reference string and if any of the dropped points are in the start and end chainage range then sections are generated at those dropped chainages.

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start and end chainages to drop tangent points.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

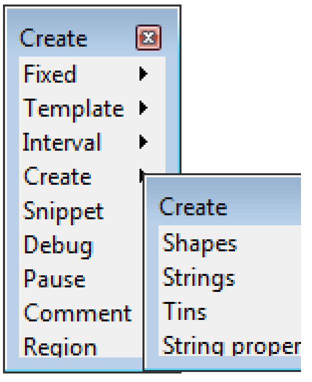
Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Return to 21.2.2.4 Interval.

21.2.2.5 MTF Create

The **Create** walk-right brings up the **Create** menu with options to create Shapes, Strings and Tins from selected MTF points.



For

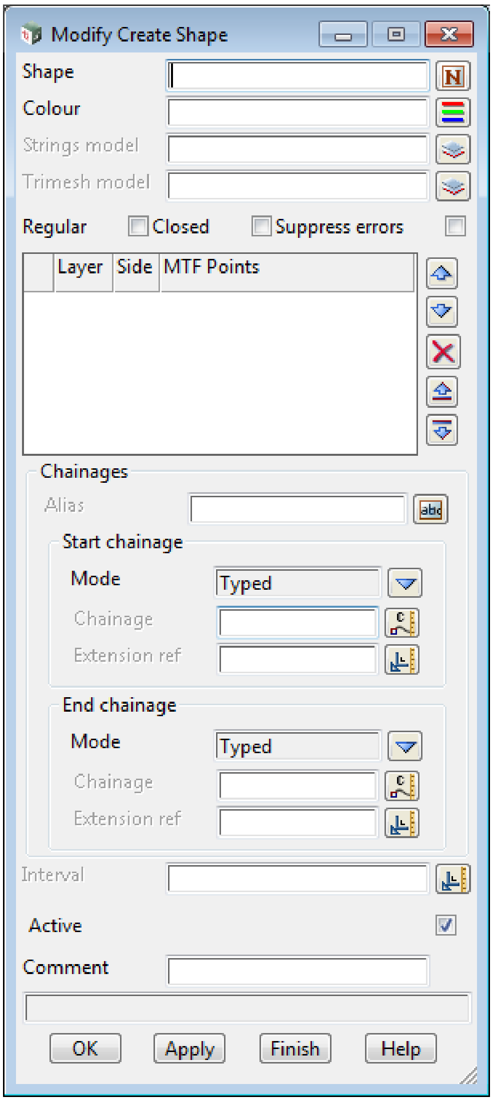
Shapes	21.2.2.5.1 Create Shapes
Strings	21.2.2.5.2 Create Strings
Tins	21.2.2.5.3 Create Tin

21.2.2.5.1 Create Shapes

The **Shapes** option creates a shape from nominated MTF Points, and optionally strings and trimeshes.

For information on MTF Points, Strings, Shapes and Trimeshes, see 21.1.1 MTF Links, Points, Sections, Strings and Trimeshes.

Selecting **Shape** brings up the panel **Modify Create Shape**



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Shape	text box		
name for the new shape			
Colour	colour box		available colours

colour of the shape.

Strings model model box available models

*if **not blank**, strings are created for the MTF point listed in the Shape grid and added to this model. The strings have the same name and colour as the MTF points.*

*If **blank**, strings are not created.*

Trimesh model model box available models

*if **not blank**, a trimesh is created for the shape and added to this model.*

*If **blank**, trimeshes are not created.*

Regular tick box

*if **ticked** the shape will only be formed when all of the nominated strings exist. E.g. it will form multiple shapes if strings come and go along the length of the apply.*

*If **not ticked**,.*

Closed tick box

*if **ticked**, the shape is closed off. That is, the last MTF point in the list is joined to the first MTF point.*

*If **not ticked**, the shape is not closed off.*

Shape Grid

Layer layer box available Layers

layer to take the MTF point from. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Side choice box Left, Right

side of the template to take the MTF point from.

Links LJG? text box

the name of the MTF point from the layer **Layer** *and side* **Side**.

The shape profile is constructed from the shape grid defines by joining each MTF point to the next MTF point in the list. See 21.1.1 MTF Links, Points, Sections, Strings and Trimeshes

If **Closed** *is ticked, t he last MTF point in the list is joined to the first MTF point.*

Alias, **Start Chainage**, **End Chainge**, **Interval**

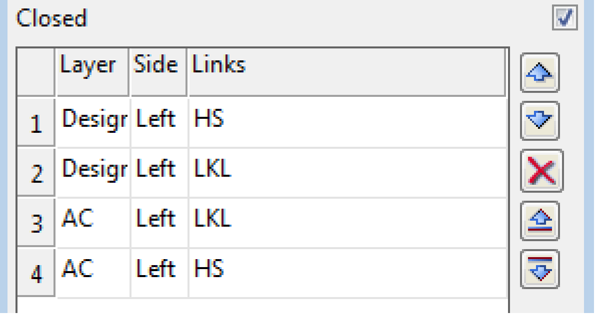
defines the start/end chainages to create the shape.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

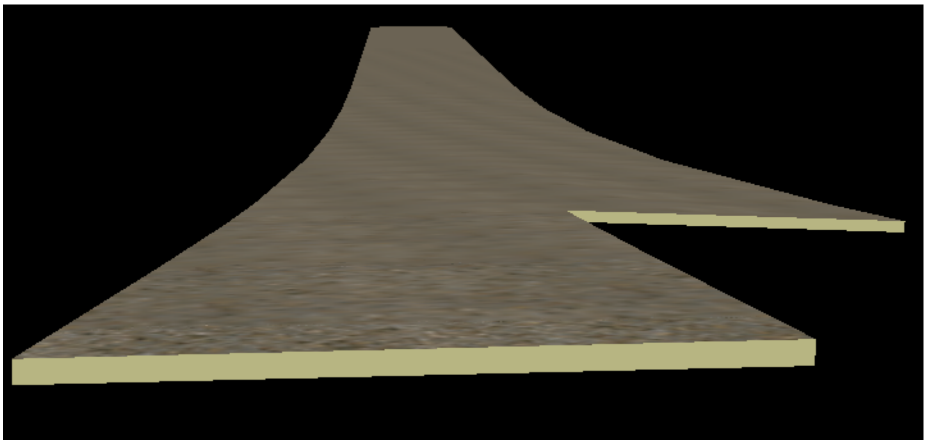
Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

For example, the list of MTF points



creates a shape at each chainage by joining the four MTF points in the order down the list, and then closes the shape by joining the fourth MTF point to the first MTF point.



Go to the next section 21.2.2.5.2 Create Strings or return to 21.2.2.5 MTF Create.

21.2.2.5.2 Create Strings

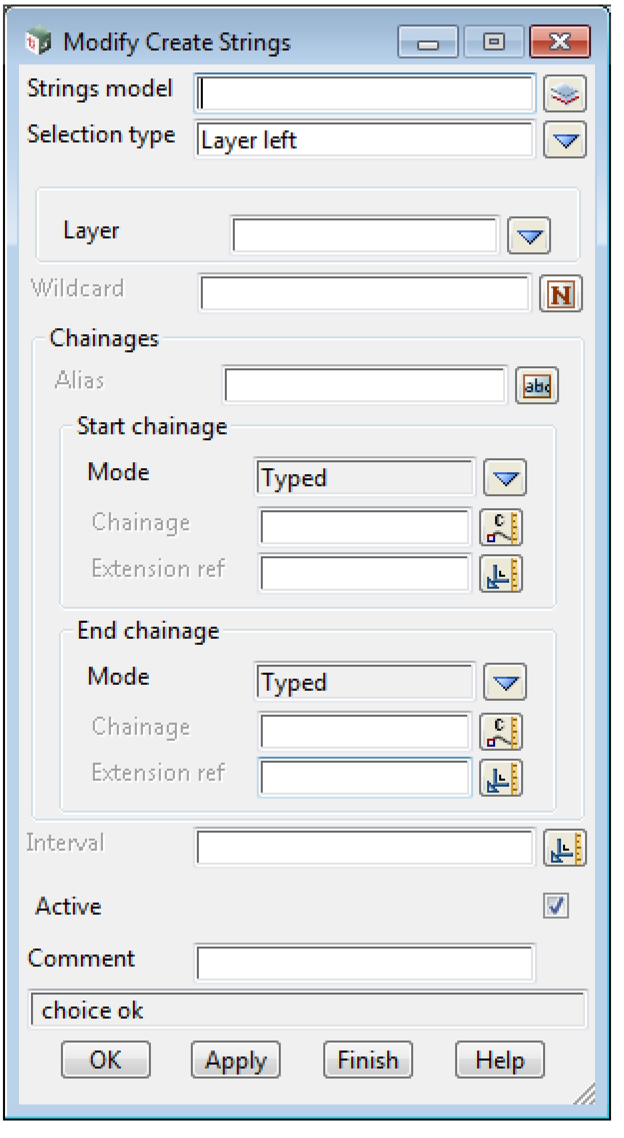
The **Strings** option creates longitudinal strings for nominated MTF points and places them in a given model.

One application of the **Strings** option is to create a model of strings for a given Layer instead of the **Boxing Layers** in the **Models** tab of the **Apply MTF Function.**

Note: Strings and sections are automatically created for all the MTF points in the default Layer **Design**.

For information on MTF Points, Strings, Shapes and Trimeshes, see 21.1.1 MTF Links, Points, Sections, Strings and Trimeshes.

Selecting **Strings** brings up the panel **Modify Create Strings**

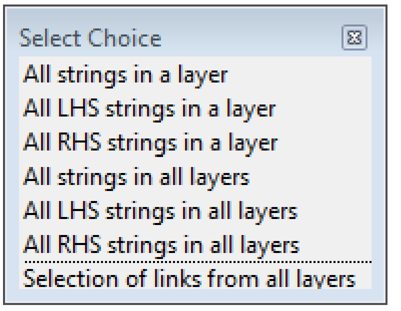


The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Strings model	model box		available models

the strings that created for the nominated points are placed in this model.

Selection type choice box



***Selection type** specifies where to get all the names of points created at any section in the Start and End chainage range.*

*For any **Selection type** other than **Selection of links from all layers**, the points then have to match the* **Wildcard** *and then strings are created with the name and colour of the point, and placed in the* **Strings model**.

If **All points in a layer**, *all points in the layer given in* **Layer**.

If **All LHS points in a layer**, *all points in the layer given in* **Layer** *that are on the left hand side.*

If **All RHS points in a layer**, *all points in the layer given in* **Layer** *that are on the right hand side.*

If **All points in all layers**, *all points in the all layers.*

If **All LHS points in all layers**, *all points in the all layers that are on the left hand side.*

If **All RHS points in all layers**, *all points in the all layers that are on the right hand side.*

Layer layer box available Layers

used when **Selection type** *is* All points in a layer, All LHS points in a layer, *or* All RHS points in a layer. *For information on Layers, see* 21.1.1.2 MTF Links and Layers.

Wildcard text box

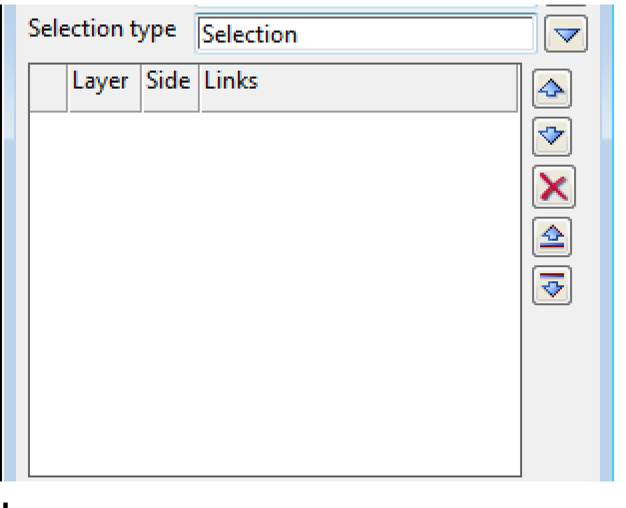
used for all **Selection type's** *except* Selection of points for all layers.

*text including wild cards * and wild characters ?.*

*If **blank**, all points from* **Selection type** *have strings created for them*

*If **not blank**, any names of points from the* **Selection type** *that **match** the* **Wildcard** *have strings created for them with the same name and colour as the point.*

If **Selection of points for all layers**, *then a grid is used to nominate the points to be selected. The points do no have to be from the same layer.*



Selection Grid

Layer	layer box	available Layers

layer to take the point from. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Side	choice box	Left, Right

side of the template to use to take the point.

Points	text box

the name of the point from the layer **Layer** *and side* **Side**.

For any of the points in the table, strings are created and placed in the **Strings model**.

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start and end chainages to create strings between.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

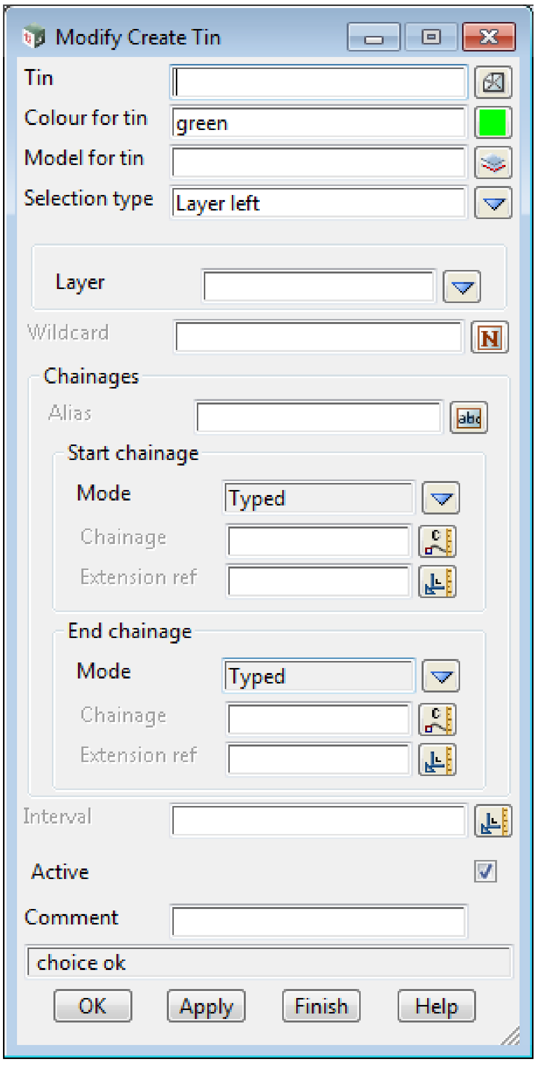
Go to the next section 21.2.2.5.3 Create Tin or return to 21.2.2.5 MTF Create.

21.2.2.5.3 Create Tin

The **Tins** option creates a tin from selected MTF strings.

For information on MTF Points, Strings, Shapes and Trimeshes, see 21.1.1 MTF Links, Points, Sections, Strings and Trimeshes.

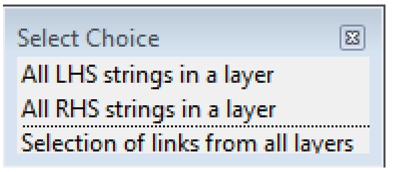
Selecting **Tins** brings up the panel **Modify Create Tin**



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins
the name for the created tin.			
Colour for tin	colour box		available colours
the colour of the created tin.			
Model for tin	model box		available models
the created tin is added to this model.			

Selection type choice box



Selection type *specifies where to get all the MTF points in sections in the Start and End chainage range, and create the strings that are used in the triangulation.*

For any ***Selection type*** *other than* ***Selection of links from all layers****, the points have to match the* **Wildcard**. *Then strings are created and used in the triangulation.*

If **All LHS strings in a layer***, then all points in the layer given in* **Layer** *that are on the left hand side and satisfy* **Wildcard** *are used to create strings, and the strings are used in the triangulation.*

If **All RHS strings in a layer***, then all points in the layer given in* **Layer** *that are on the right hand side and satisfy* **Wildcard** *are used to create strings, and the strings are used in the triangulation.*

Layer layer box available Layers

used for all **Selection type's** *except* Selection of points for all layers.

The Layer to take MTF points from. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Wildcard text box

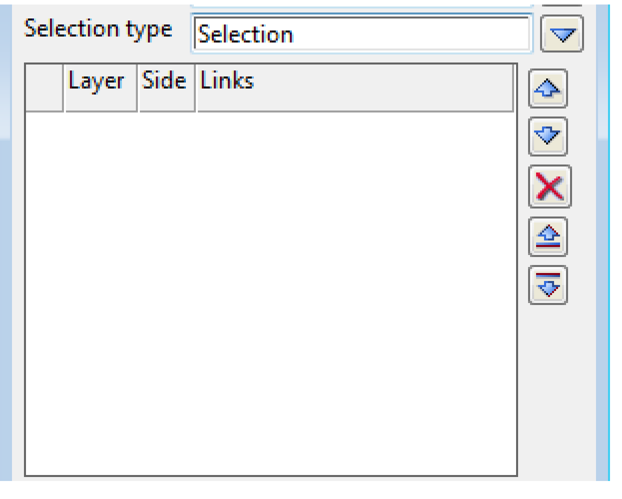
used for all **Selection type's** *except* Selection of points for all layers.

*Text including wild cards * and wild characters ?.*

If ***blank****, all points from* **Selection type** *have strings created for them, and the strings are used in the triangulation.*

If ***not blank****, any names of points from the* **Selection type** *that* ***match*** *the* **Wildcard** *have strings created for them and the strings are used in the triangulation.*

If **Selection of points for all layers***, then a grid is used to nominate the points to be selected. The points do no have to be from the same layer.*



Selection Grid

Layer layer box available Layers

layer to take the point from. For information on Layers, see 21.1.1.2 MTF Links and Layers.

Side choice box Left, Right

side of the template to use to take the point.

Points text box

the name of the point from the layer **Layer** *and side* **Side**.

For any of the points in the table, strings are created and placed in the **Strings model**.

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start and end chainages to create a tin.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

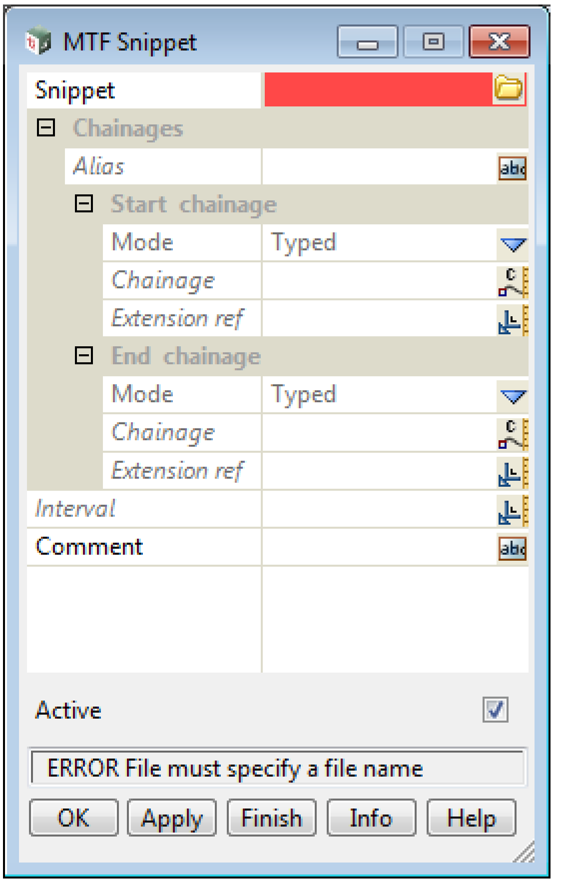
Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Return to 21.2.2.5 MTF Create.

21.2.2.6 Snippet

Clicking on **Snippet** brings up the **MTF Snippet** panel which is used to insert snippets.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Snippet	snippet box		*.mtfsnippet, *.mtfsnippetc files

snippet to run.

If information on the snippets shipped by 12d Solutions and available from **[Lib]** *in the Folder pop up, see* 20.4.3.5 12d Supplied Snippets.

For more information on the inserting and use of snippets, see 21.5 Defining and Using Snippets.

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start and end chainages to apply a snippet.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra End, Active, OK, Apply**

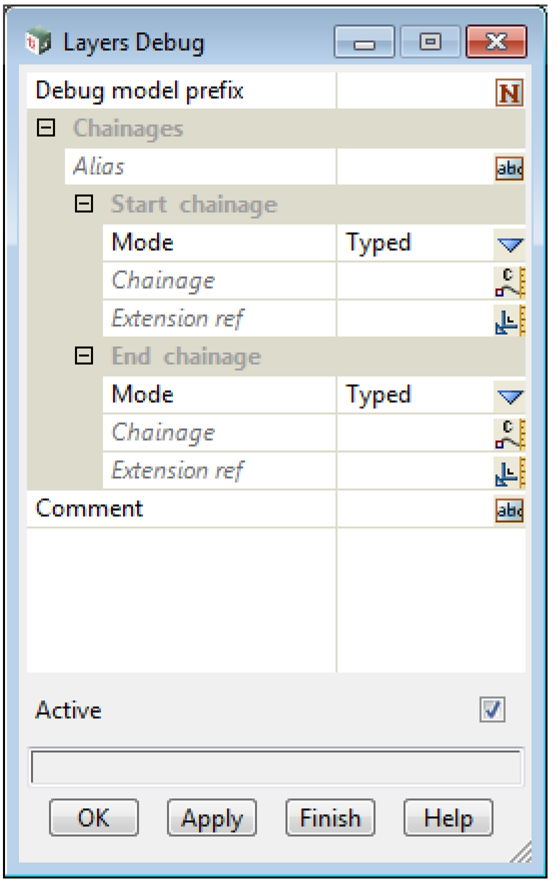
For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

For more information on the inserting and use of snippets, please go to the section 21.5 Defining and Using Snippets.

Go to the next section 21.2.2.7 Debug or return to 21.2.2.1 Create MTF Commands.

21.2.2.7 Debug

Clicking on **Debug** brings up the **Layers Debug** panel which is used to create debug information when you area having problems with your MTF, especially with Snippets.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Debug model prefix	text box		

debug models are created with the name of this prefix followed by the chainage.

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages to create debug models and file.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

Comment, **Extra start**, **Extra end, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

What the Panel Does

For each chainage. a model is created with the name **Debug model prefix** *followed by the chainage value. In the model all the link in all the layers are drawn nut with the offset as the x-value and height as the y-value so that the model can be looked at on a* ***Plan View****.*

A report file is also created at the start chainage and it contains information debugging such as the snippets used and the values of the parameters in the snippets.

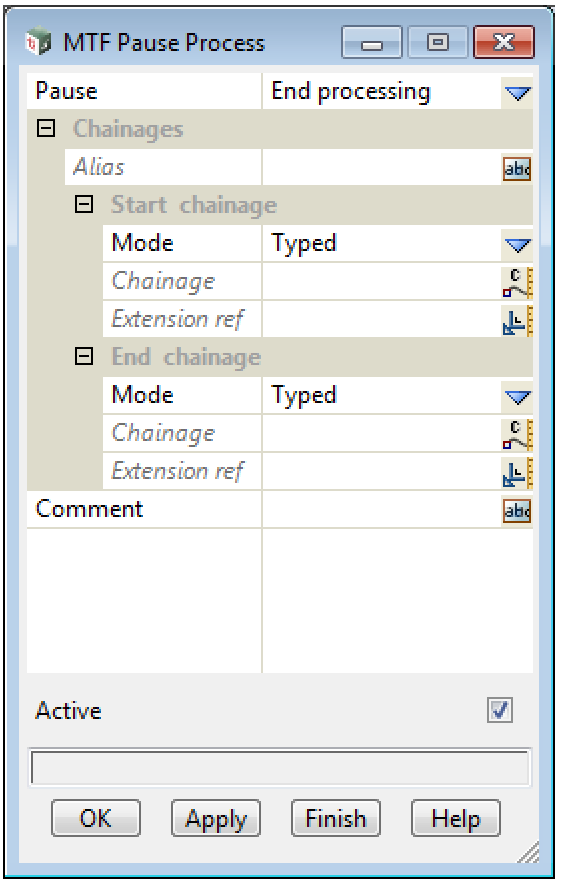
The Report has the name **Debug model prefix** *followed by the chainage value with the decimal point replaced by an underscore, with* ***12D_APPLY_MTF_DEBUG_DUMP*** *as the file extension.*

For example, **Lee Road debug 0_000.12D_APPLY_MTF_DEBUG_DUMP**

Go to the next section 21.2.2.8 Pause or return to 21.2.2.1 Create MTF Commands.

21.2.2.8 Pause

Clicking on **Pause** brings up the **MTF Pause Process** panel which is used to enable/disable the processing of all the MTF commands in the given chainage range.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pause	choice box	Stop processing	Start processing, Stop processing

*if **End processing**, then for a given chainage **ch** in the chainage range, all commands after this command will **NOT** be processed until a* **MTF Pause Process** *panel with* **Process** *set to* **Start processing** *(and whose chainage range includes this chainage **ch**) is met.*

*If **Start processing**, then for a given chainage **ch** in the chainage range, all commands after this command will be processed until a* **MTF Pause Process** *panel with* **Process** *set to* **End processing** *(and whose chainage range includes this chainage **ch**), is met.*

Alias, **Start Chainage**, **End Chainge**, **Interval**

defines the start/end chainages to use the Pause command.

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

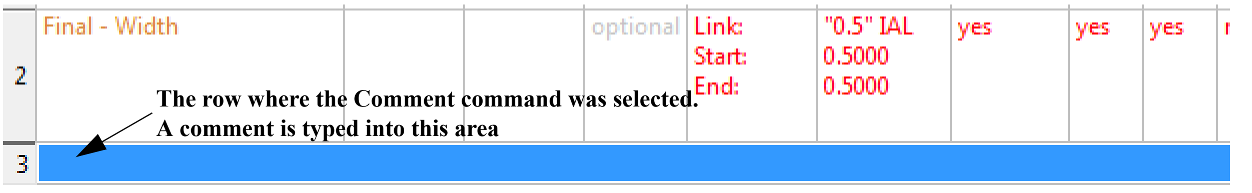
Comment, **Extra start**, **Extra End, Active, OK, Apply**

For information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

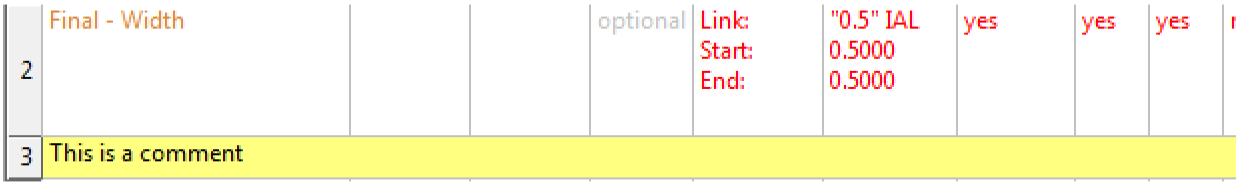
Go to the next section 21.2.2.9 Comment or return to 21.2.2.1 Create MTF Commands.

21.2.2.9 Comment

Selecting ***Comment*** turns the line of the ***Template Modifier*** into a field that a comment can be typed into.



Typing will then enter a comment into this line and the entire line contains just the comment.

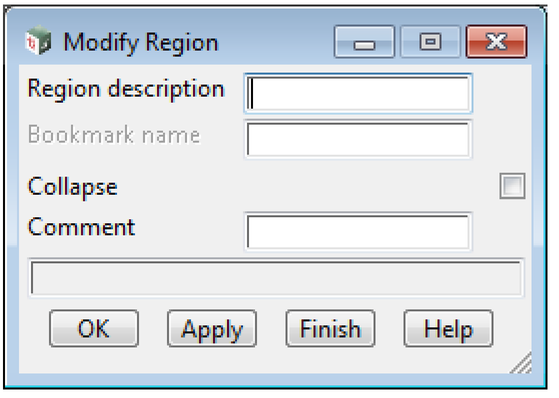


Go to the next section 21.2.2.10 Region or return to 21.2.2.1 Create MTF Commands.

21.2.2.10 Region

The **Region** option inserts a **Region command** into the grid.

Selecting **Region** brings up the **Modify Region** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Region description	text box		

name for the Region.

Bookmark name	text box		

*if **not blank**, this is a secondary name, often a shorter name, that is used in the **Region** pop up rather than using the **Region description.***

*If **blank**, then when the grid is saved, the* **Bookmark name** *is set to be the same as the* **Region description**.

Collapse	tick box		

*if **ticked**, all the commands in the grid until the next **Region** command are collapsed into this Region command.*

*If **not ticked**, the commands in the grid until the next **Region** command are not collapsed.*

OK, Apply

for information on these panel fields, see 21.2.2.1.1 Common Fields and Buttons on MTF Modifier Panels.

There are **Regions**, **Previous Region** and **Next Region** icons at the right hand side of the **Left/ Right Modifiers** panel that use this created *Region*. For information on using these icons and information on **Regions** in a grid, see 4.19.6 Grids in Panels.

Return to 21.2.2.1 Create MTF Commands.

21.2.3 MTF Boxing

In the **MTF**, up to eight (8) layers of boxing can be defined.

For each boxing layer, the boxing can be defined by

(a) the Left boxing only

(b) the Right boxing only

(c) the Left and Right boxing.

For cases (a) and (b), the Left boxing or the Right boxing defines the boxing across the **entire** design section and then only one of Left boxing or Right boxing is needed.

For case (c) where **both** Left and Right boxing are defined for a design section, then the *last point of the left boxing* is automatically connected to the *first point of the right boxing*.

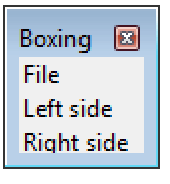
Warning for case (c), the Left boxing must end before the Right boxing begins. If there is an overlap, the Right boxing will be pushed to the **end** of the Left boxing.

Note - no interpolation or modifiers exist for boxing.

For a general discussion of Boxing, go to the section 21.6 What is Boxing?, for the scenarios of applying Boxing, including but not restricted to the MTF, go to the section 21.6.2 Applying Boxing, and in particular 21.6.2.1 Scenario 1 - Design Generated by One Apply MTF.

When boxing is used in the **Apply MTF**, the cut and fill areas and volumes are also calculated for all the inter-boxing layers. The ***last*** *layer* defined is also referred to as the ***subgrade*** layer and volumes are also given for the natural surface to the subgrade, and the design to the subgrade.

The **Boxing** walk-right in the **MTF Edit** menu brings up the **Boxing** menu with options to specify the file supplying the boxing definitions and options to apply the definitions to the Left and Right side of the design sections.



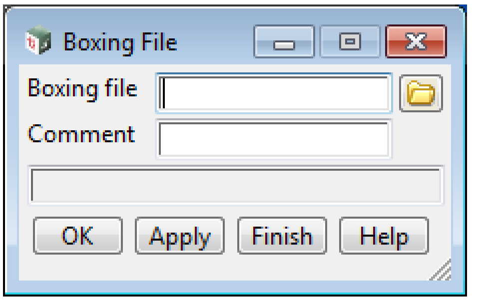
For the option

File	21.2.3.1 MTF Boxing File
Left/Right side	21.2.3.2 Left Side and Right Side Boxing
Left/Right side for each layer	21.2.3.2 Left Side and Right Side Boxing

Each of the options from this menu will now described.

21.2.3.1 MTF Boxing File

Selecting **File** from the **Boxing** menu brings up the **Boxing File** panel for defining the file of **Boxing Definitions** to be used in this **Apply MTF**.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Boxing file	input		*.bf files

name of the Boxing file containing the Boxing Definitions to be used for the MTF.

Comment column header

comment to add to the end of the line. In the text mtf file, the comment will be preceded by //.

OK/Apply button

OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

The full definition of all the boxing command used in a **Boxing Definition** is given in the section 21.7 Full Definition of Boxing.

Please continue to the next section 21.2.3.2 Left Side and Right Side Boxing.

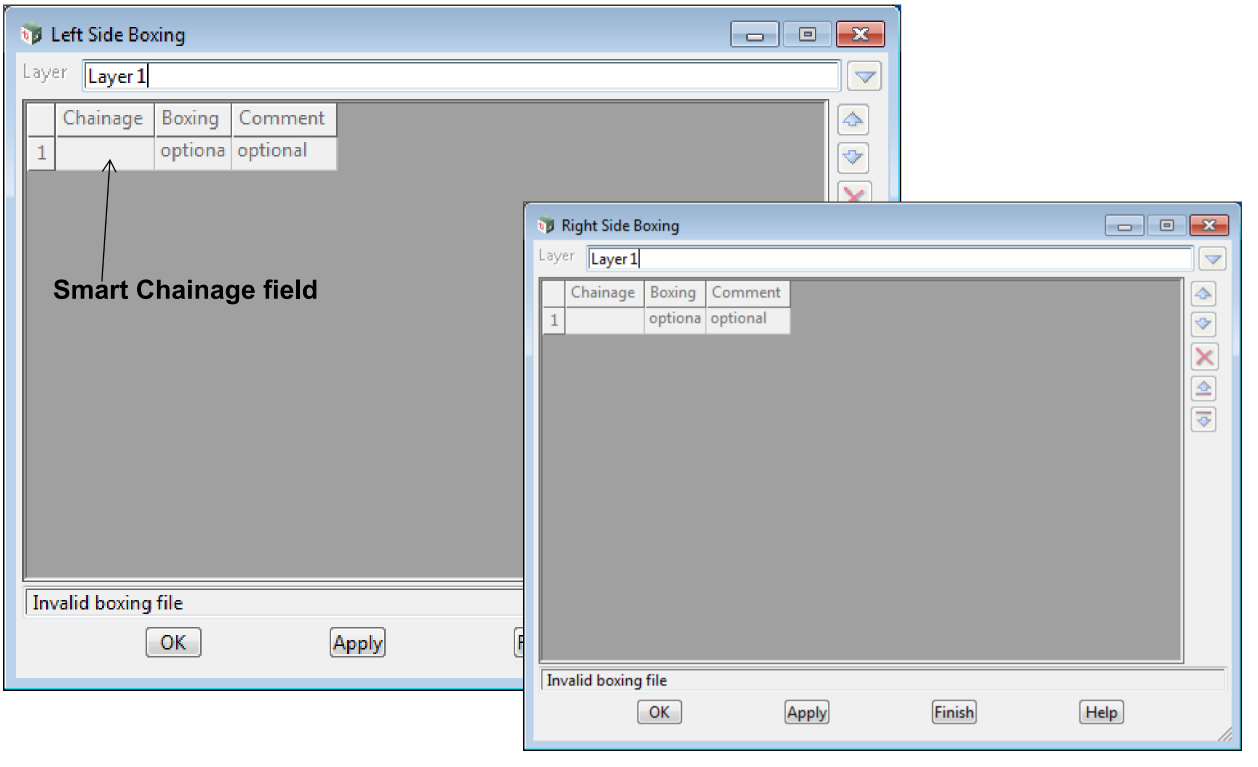
21.2.3.2 Left Side and Right Side Boxing

The panels brought up by selecting **Left side** or **Right side** from the **Boxing** menu brings up the **Left Side Boxing** and **Right Side Boxing** panels respectively. By default, this is for the first layer of boxing.

The **Left Side Boxing** and **Right Side Boxing** panels define for each boxing layer, what Boxing Definitions (from the boxing file) are applied, in what chainage ranges, along the reference string.

For the definition of boxing, go to the section 21.7 Full Definition of Boxing

The panels are brought up by selecting **Left side** or **Right side** for **Layer 2** to **Layer 8** from the **Boxing** menu and then selecting the *Layer number* in the **Layer** choice box. Similar panels exist for *Left* and *Right*, and for each selected layer, and so won't be documented separately.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Left/Right Boxing Grid

For more information on creating and editing data in a grid, see 4.19.6 Grids in Panels.

Chainage smart chainage box

chainage to start applying the Boxing Definition given in the Boxing column. The Boxing Definition goes until the chainage on the next row of the Grid.

*If you right click in the **Chainage** column or select **Browse** if the Browse menu comes up, then a* **Chainage** *panel comes up which has the most of the standard MTF choices for **Smart Chainages**. See* 21.2.1 MTF Hinge Modifiers.

Boxing boxing definition box available boxing definitions

name of the Boxing Definition to apply from this chainage to the chainage in the next row of the Grid.

Comment text box

comment for this row of the grid.

Buttons at Bottom

OK/Apply button

OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

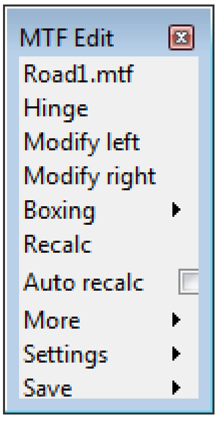
Please continue to the next section 21.2.4 MTF Recalc.

21.2.4 MTF Recalc

If ***Recalc*** is clicked, the ***Apply MTF*** associated with the MTF is ***recalced***.

Please continue to the next section 21.2.5 MTF Auto Recalc.

21.2.5 MTF Auto Recalc



if tick a recalc is done when "Apply" is clicked on Modifiers panels

If ***Auto recalc*** is ticked, then whenever the **Apply** button is clicked on any of the panels for the ***Modifiers - Left Side Modifiers*** or ***Modifiers - Right Side*** or ***Boxing Definitions*** used in the ***Boxing***, then a **recalc** of the associated ***Apply MTF*** for the MTF is done.

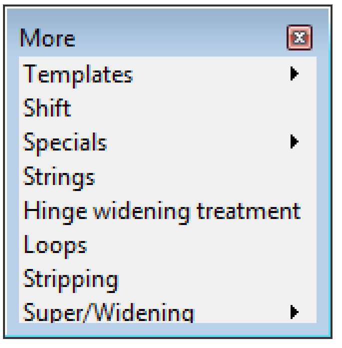
Please continue to the next section 21.2.6 MTF More.

21.2.6 MTF More

Position of menu: **Design =>MTF =>More**

The **More** walk-right menu has options to apply and interpolate between templates, shift chainages by a delta value, use special chainage values and files and apply Strings modifiers.

The **More** walk- right menu is



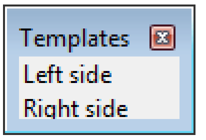
apply templates
shift chainge values
special chainages
apply string modifiers

For the options see:

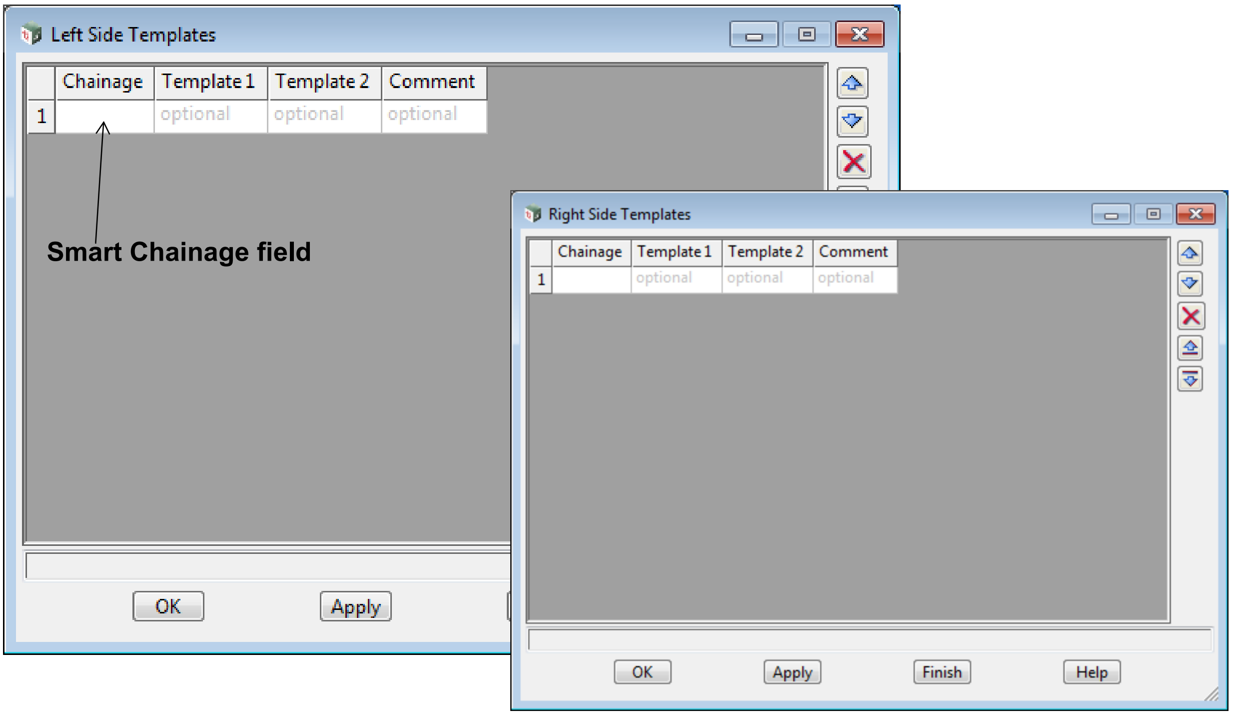
Templates	21.2.6.1 MTF Templates
Shift	21.2.6.2 MTF Shift
Specials	21.2.6.3 MTF Special Chainages
Strings	21.2.6.4 MTF String Modifiers
Hinge widening treatment	21.2.6.6 MTF Hinge Widening Treatment
Loops	21.2.6.7 MTF Remove Loops
Stripping	21.2.6.8 MTF Stripping
Super/Widening	21.2.6.9 MTF Super/Widening

21.2.6.1 MTF Templates

The **Templates** walk-right brings up the **Templates** menu



and selecting the **Left side or Right side** option brings up the **Left Side Templates** or **Right Side Templates** panels respectively which are used to specify which templates are used, and over what chainage range, for the Left and Right sides.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Templates Grid

For more information on creating and editing data in a grid, see 4.19.6 Grids in Panels.

Chainage smart chainage box

list of chainages for applying templates on the left/right hand side of the hinge string.

*If you right click in the **Chainage** column or select **Browse** if the Browse menu comes up, then a* **Chainage** *panel comes up which has the most of the standard MTF choices for **Smart Chainages**. See* 21.2.1 MTF Hinge Modifiers.

Template 1 template box available templates

template to start applying at the chainage on the same row. If template 2 is blank, then template 1 is applied until the next chainage in the chainage column.

Template 2 template box available templates

template to linearly interpolate to and finish with at the next chainage in the chainage column. In the file, the two templates will be separated by a comma.

Note - template 2 must have the same number of fixed and variable links as template 1 otherwise a gap *of the section separation length will be left between the end of template 1 and the start of template 2.*

Comment column header

comment to add to the end of the line. In the file, the comment will be preceded by //.

Buttons at Bottom

OK/Apply button

OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

Continue to the next section 21.2.6.2 MTF Shift or return to 21.2.6.2 MTF Shift21.2.6 MTF More.

21.2.6.2 MTF Shift

In an ***MTF***, most modifiers are defined in terms of chainage on the reference string. For example, the width modifier is defined to apply between a given start and end chainage on the reference string.

If the horizontal geometry of the reference string is modified, the reference chainages in the MTF that are defined by the Smart Chainage mode **Typed** will be incorrect.

Shift can help correct the **Typed** chainages in the MTF file when modifications are made to the reference string horizontal geometry *after the MTF has been defined*.

For the ***MTF*** and all special chainage files referred to in the ***mtf*** *that are only typed values*, **Shift** adds a delta chainage (the *shift distance*) to chainages within a user specified range.

However depending on the type of modifications made to the reference string, **shift** may have to be applied a number of times over a number of different chainage ranges. And apart from a simple change of start chainage, there may be chainages that can not be corrected with **Shift.**

Before using **Shift**, it is necessary to know ***how*** the reference string has changed from its original position so to use shift it is advantageous to make a **copy** of the reference string **before** any changes are made.

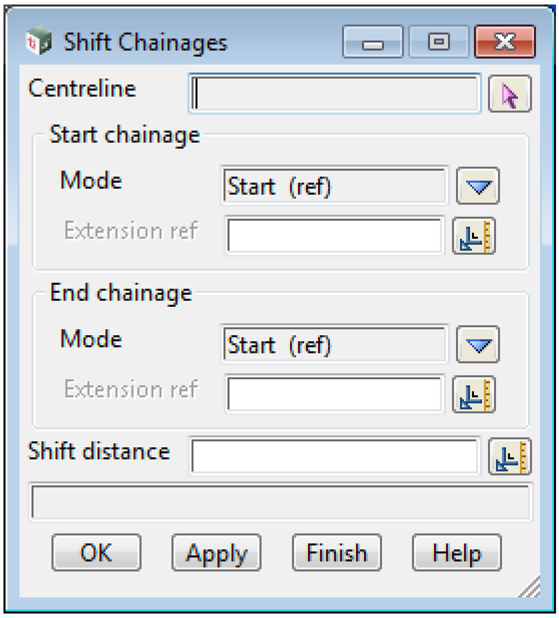
Examples Using **Shift**:

1. If only the start chainage of the reference string was modified, **shift** would be applied to the mtf for the entire ***original*** reference string and end chainages, with a *shift distance* equal to the difference between the new and the old start chainages for the reference string. For this case the horizontal geometry has not changed, just the chainages.
2. If the radius of a horizontal curve is changed in the middle of the reference string, all chainages from the beginning of the string up to point where the horizontal geometry starts to change would stay the same. At some point after the modified curve, the horizontal geometry of the string is unchanged and for this section of the mtf, a shift distance equal to the difference between the new and the old chainages of a non-modified point.

In all areas where the horizontal geometry has been modified, the chainages in the mtf would have to be carefully examined to see what changes are required.

Warning: **Shift** option is very powerful but is also very dangerous. Where ever possible, *Smart Chainage* **modes** other than **Typed** should be used.

Selecting **Shift** brings up the **Shift Chainages** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Centreline	string-select		

.

Start chainage smart chainage box

chainage in the mtf to start adding the **Shift distance.** *For information on* ***Smart Chainages****. See* 21.2.1 MTF Hinge Modifiers.

Any Typed chainages in the mtf file (and any associated special chainage files) between the ***Start Chainage*** *and the* ***End chainage*** *will have* **Shift distance added to them.**

End chainage smart chainage box

chainage in the mtf to stop adding the **Shift distance.** *For information on* ***Smart Chainages****. See* 21.2.1 MTF Hinge Modifiers.

Any Typed chainages in the mtf file (and any associated special chainage files) between the ***Start Chainage*** *and the* ***End chainage*** *will have* **Shift distance added to them.**

Shift distance real box

delta chainage to add to all the chainages in the mtf file and any special chainages that are between the ***Chainage to shift from*** *and the* ***Chainage to shift to***

Buttons at Bottom

OK/Apply button

OK *applies the shift and removes the panel.*
Apply *applies the shift and leaves the panel on the screen.*

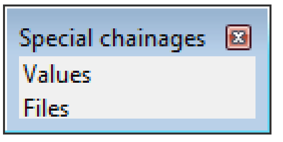
Continue to the next section 21.2.6.3 MTF Special Chainages or return to 21.2.6.2 MTF Shift21.2.6 MTF More.

21.2.6.3 MTF Special Chainages

The **Apply MTF** option automatically creates sections at a number of chainages including the supplied chainage interval, horizontal and vertical critical points, horizontal and vertical chord-arc tolerances, template change points and modifier change points.

However it is also possible to add extra sections at special chainages using the **Specials** option from the **MTF Edit** menu.

Walking right on the **Specials** menu brings up the **Special Chainages** walk-right menu:



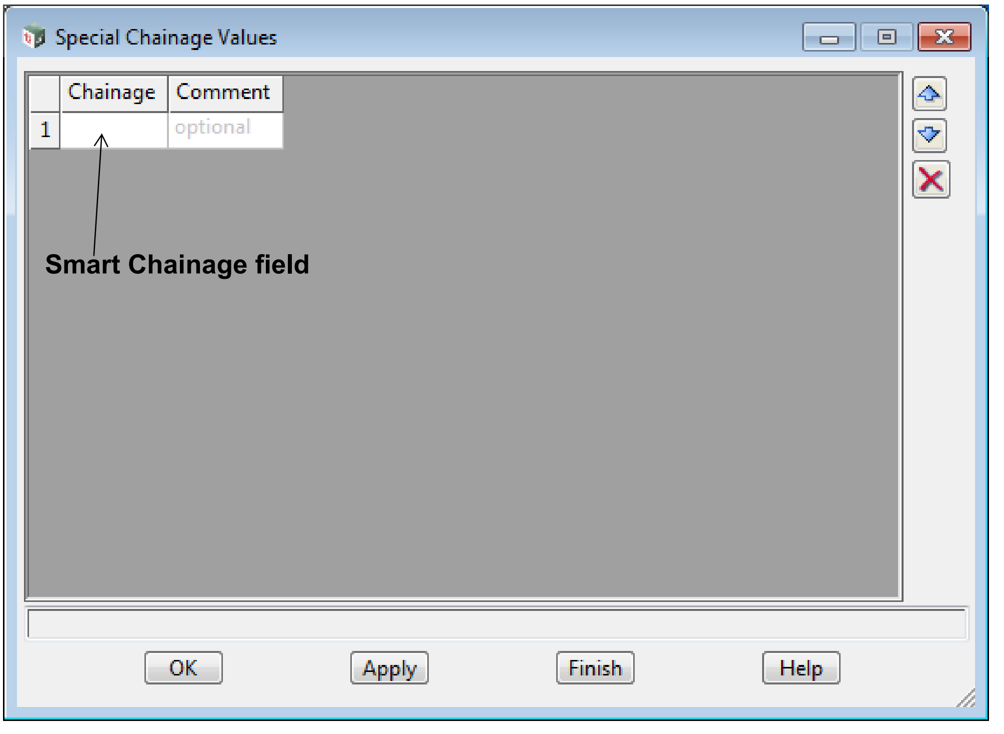
For the options see:

Values 21.2.6.3.1 MTF Special Chainage Values

Files 21.2.6.3.2 MTF Special Chainage Files

21.2.6.3.1 MTF Special Chainage Values

Selecting **Values** brings up the **Special Chainage Values** panel



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Special Chainage Values Grid

For information on the general operation of a grid including the icons on the right hand side, see 4.19.6 Grids in Panels.

Chainage smart chainage box

list of special chainages for creating sections at.

*If you right click in the **Chainage** column or select **Browse** if the Browse menu comes up, then a* **Chainage** *panel comes up which has the most of the standard MTF choices for **Smart Chainages**. See* 21.2.1 MTF Hinge Modifiers.

Comment text box

comment for this row of the grid.

Buttons at Bottom

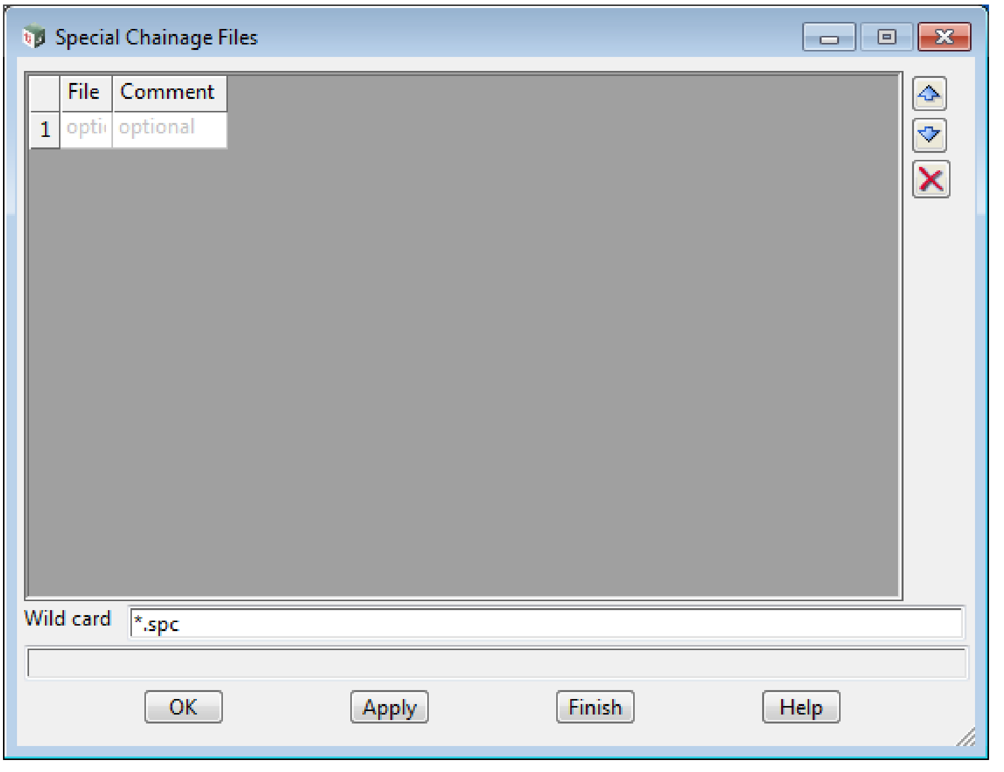
OK/Apply button

OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

Continue to the next section 21.2.6.3.2 MTF Special Chainage Files or return to 21.2.6.3 MTF Special Chainages or 21.2.6 MTF More

21.2.6.3.2 MTF Special Chainage Files

Selecting **Files** brings up the **Special Chainage Files** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
File	column header		size menu for the fields -*.spc files

list of files of special chainages for creating sections at.

Comment	column header		

comment to add to the end of the line. In the file, the comment will be preceded by //.

Wild card	input	*.spc	

the wild card used for pop-ups in the **special chainage files** *fields.*

OK/Apply	button		

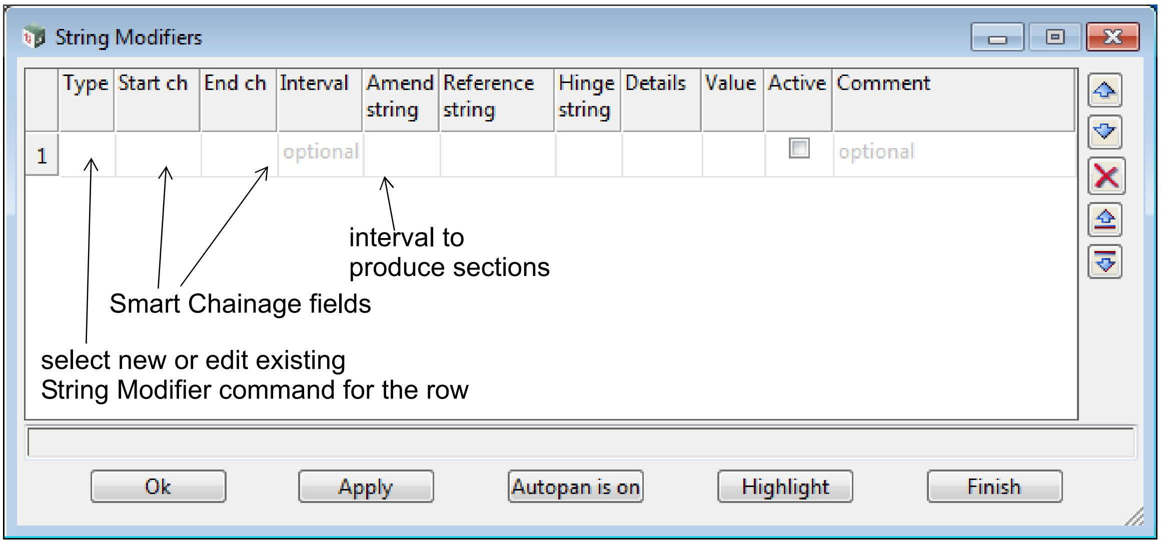
OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

Continue to the next section 21.2.6.4 MTF String Modifiers or return to 21.2.6.3 MTF Special Chainages or 21.2.6 MTF More.

21.2.6.4 MTF String Modifiers

For information on how the String Modifiers in the MTF editor work, see 21.2.6.5.3 Description of String Modifiers in MTF.

Selecting **Strings** from the **MTF Edit** menu brings up the **String Modifiers** panel.



The **String Modifiers** panel consists of a grid with of rows (or lines) with **commands** in them, and an **OK** or **Apply** button to record the results.

String Modifier Grid

For information on the general operation of a grid including the icons on the right hand side, see 4.19.6 Grids in Panels.

*The **commands** in the* **String Modifier** *grid are processed sequentially from the top to the bottom of the grid.*

When the Grid Row is Empty

*If the row of the grid is **empty**, clicking LB in the empty row will bring up the* **Create** *menu which contains all the available string modifier commands. Note this may involve two clicks - one to highlight a column in a row and the second click to bring up the* **Create** *menu.*

Selecting a menu item will bring up an associated panel which displays the information required for the hinge command. When the panel is filled in and **OK** or **Apply** *selected, the panel information is written out to the row of the grid and is known as a* **hinge command**. *For information on each of the Hinge Commands, see* 21.2.6.5 String Modifiers Commands.

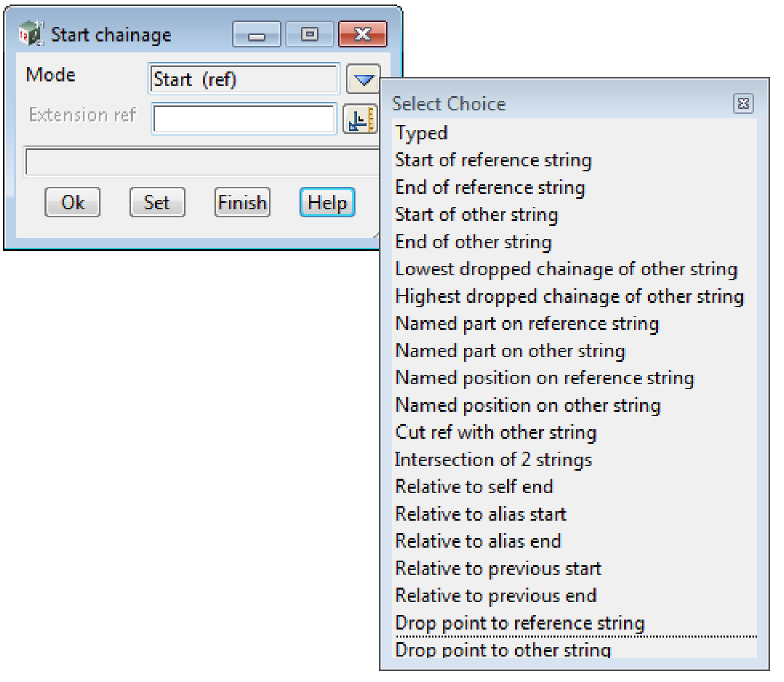
When the Grid Row is Not Empty

*If the row of the grid is **not empty** (and hence filled with a* **string modifier command**) *then clicking LB in the cell:*

(a) ***Type** will bring up the associated panel for the string modifier command.*

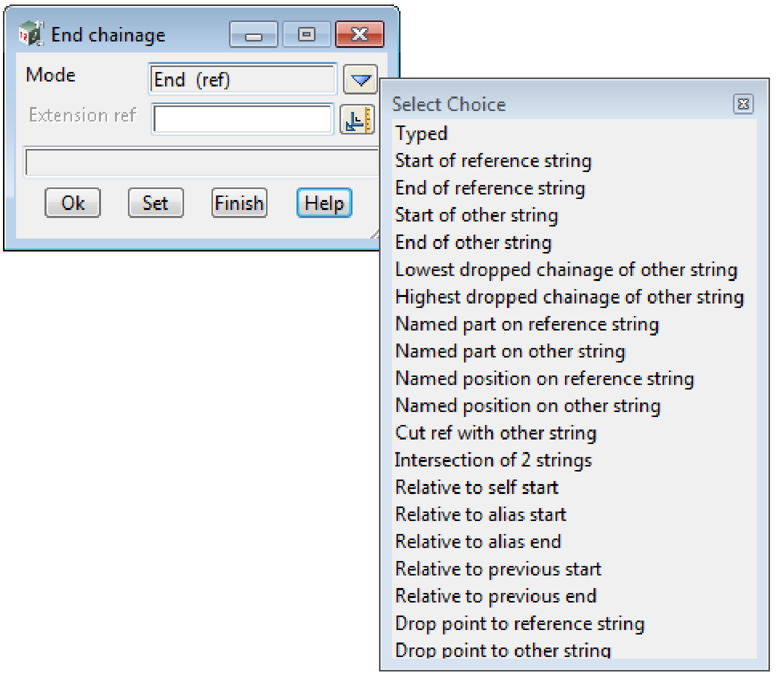
For information on each of the Hinge Commands, see 21.2.6.5 String Modifiers Commands.

(b) ***Start chainage** cell brings up the Smart **Start Chainage** panel*



The **Start Chainage Mode** *will determine what other fields are also on the panel (for example* **Extension ref** *for* **Mode** ***Start of reference string****. For more information on Chainage Modes, see* 21.4 Smart Chainages.

(c) ***End Chainge*** *cell brings up the Smart* ***End Chainage*** *panel*



The **End Chainage Mode** *will determine what other fields are also on the panel (for example* **Extension ref** *for* **Mode** ***End of reference string****. For more information on Chainage Modes, see* 21.4 Smart Chainages.

(d) ***Interval*** *will allow a real value to be typed into the cell*

Note that the **Interval** column will not appear in the grid if **Show interval column?** is **not** ticked in the MTF Settings. See 21.2.7.8 MTF Setting Show Interval Column

(e) **Active** *will toggle the* ***tick/not ticked***

(f) **Comment** will allow text to be typed into the cell

Note *that clicking in the* ***Type****,* ***Start chainage****,* ***End chainage****,* ***Interval****,* ***Extra Start****,* ***Extra End****,* ***Active*** *and* ***Comment*** *cells may involve two clicks - one to highlight the cell in a row and the second click to edit or bring up the panel for the cell.*

Buttons at Bottom

OK button

OK *stores the values in the fields and removes the panel BUT no* ***recalc*** *is done.*

Apply button

Apply *stores the values in the grid and leaves the panel on the screen.*

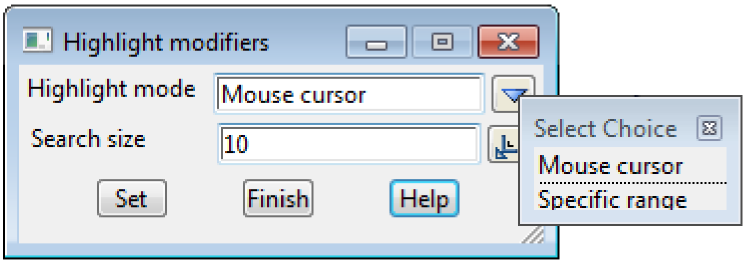
If ***Auto recalc*** *is* ***ticked*** *in the MTF, then whenever the* ***Apply*** *button is clicked, a* ***recalc*** *of the associated* ***Apply Many*** *for the MTF is done.*

Autopan on/off button

when clicked to say **Autopan is on** *then if the chainage range is not in the view that the Reference string is on then the view will be panned so that it is on the view.*

Highlight button

clicking the ***Highlight*** *button brings up the* **Highlight Modifiers** *panel.*



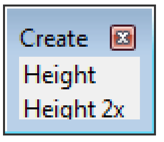
Continue to the next section 21.2.6.5 String Modifiers Commands. or return to 21.2.6 MTF More

21.2.6.5 String Modifiers Commands

The **commands** in the **String Modifiers** grid are processed sequentially from top to bottom.

The **String Modifier commands** are selected from the **Create** menu that is displayed when clicking in a cell in the **Type** column of the **String Modifiers** panel.

The **Create** menu is



modify by xfall and height
modify by 2-strings

For a description of the options, see

Height 21.2.6.5.1 Height
Height 2x 21.2.6.5.2 Height 2X

21.2.6.5.1 Height

The *height* the **Height** string modifier calculates for the amend point is given by:

(a) if the absolute flag is not set, the height of the hinge point, otherwize zero.

plus

(b) the interpolated height for the user given heights at the start and end chainages

plus

(c) the interpolated height for the user given cross falls at the start and end chainage.

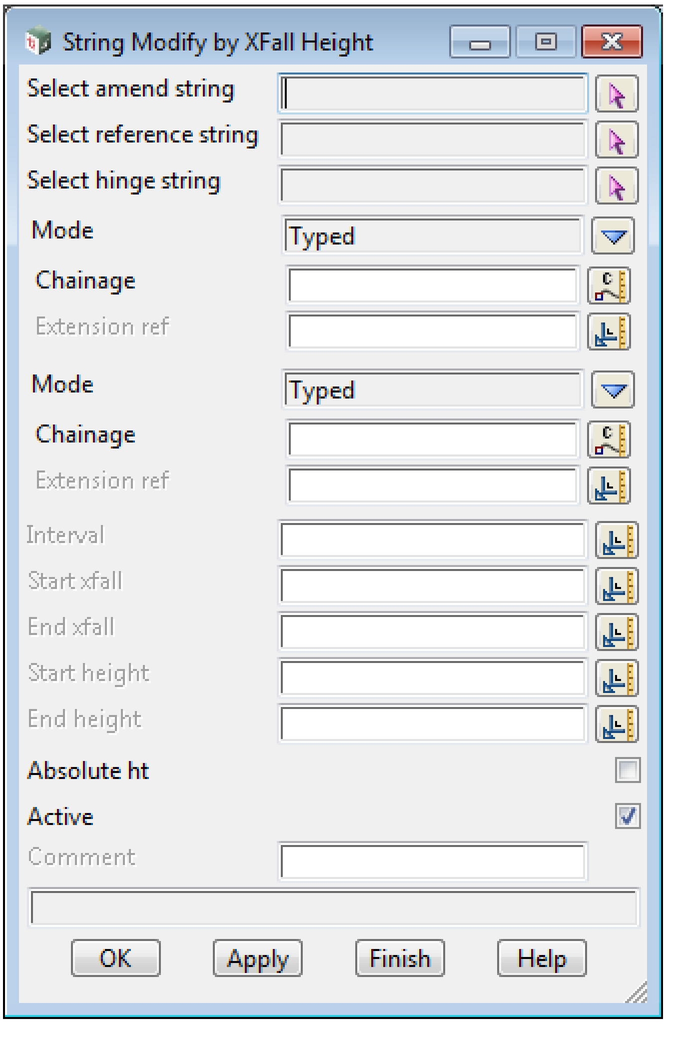
That is, if the **absolute flag** is **not set**:

new height = hinge height + (interpolated height) + (interpolated xfall) x (offset distance)

If the **absolute flag** is **set**:

new height = (interpolated height) + (interpolated xfall) x (offset distance)

Selecting **Height** brings up the **String Modify by Xfall Height** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Amend string	string-select		

select string to have heights amended.

Reference string string-select

select reference string.

Hinge string string-select

select hinge string.

Start/End chainage smart chainage box

start/end chainage on the reference string for applying the modifier. See 21.2.1 MTF Hinge Modifiers

Interval input

*if **non blank**, the chainage separation to apply the modifier.*
*If **blank**, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

Start/End xfall input measures menu

start/end crossfall for the modifier.

Start/End height input measures menu

start/end height for modifier.

Absolute height tick box

*if **ticked**, the calculated height does not include the hinge height.*
*if **not ticked**, the calculated height includes the hinge height.*

Comment input

comment for this command line

Active tick box tick

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

OK/Apply button

OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

For more information on how the String Modifiers in the MTF editor work, please go to the section 21.2.6.5.3 Description of String Modifiers in MTF.

Continue to the next section 21.2.6.5.2 Height 2X or return to 21.2.6.4 MTF String Modifiers or 21.2.6 MTF More.

21.2.6.5.2 Height 2X

The *height* the **Height 2x** string modifier calculates for the amend point is given by:

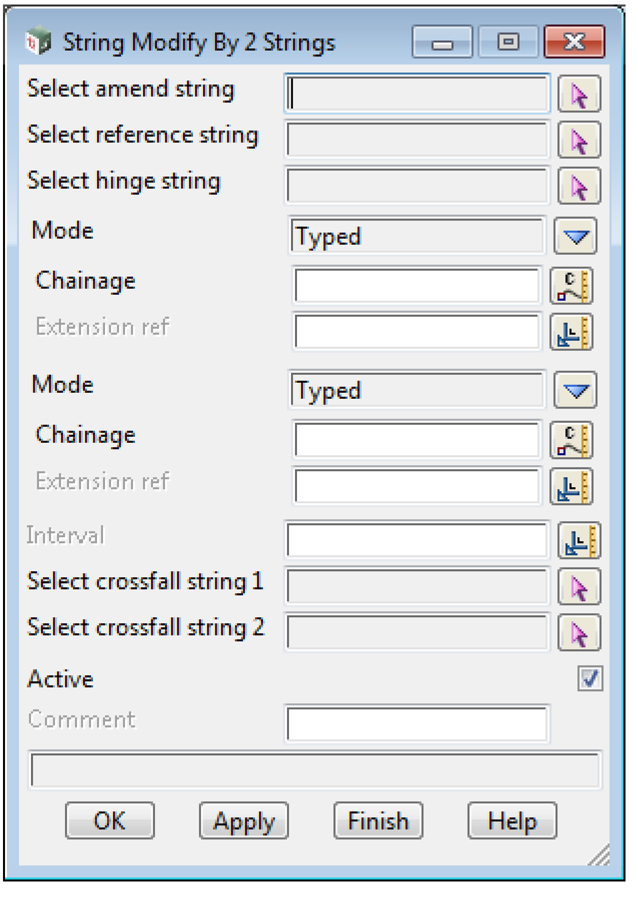
(a) the height of the hinge point

plus

(b) the offset distance multiplied by the cross fall between two user selected strings.

new height = hinge height + (xfall between string 1 and string 2) x (offset distance)

Selecting **Height 2x** brings up the **String Modify by 2 Strings** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Amend string	string-select		
select string to have heights amended.			
Reference string	string-select		
select reference string.			
Hinge string	string-select		
select hinge string.			
Start/End chainage	input		measures menu
start/end chainage on the reference string for applying the modifier.			
Interval	input		

*if **non blank**, the chainage separation to apply the modifier.*
*If **blank**, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

Crossfall string 1 string-select

select first string to define crossfall by.

Crossfall string 2 string-select

select second string to define crossfall by.

Comment input

comment for this command line.

Active tick box tick

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

OK/Apply button

OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

For more information on how the String Modifiers in the MTF editor work, please go to the section 21.2.6.5.3 Description of String Modifiers in MTF.

Continue to 21.2.6.6 MTF Hinge Widening Treatment or return to 21.2.6.4 MTF String Modifiers or 21.2.6 MTF More.

21.2.6.5.3 Description of String Modifiers in MTF

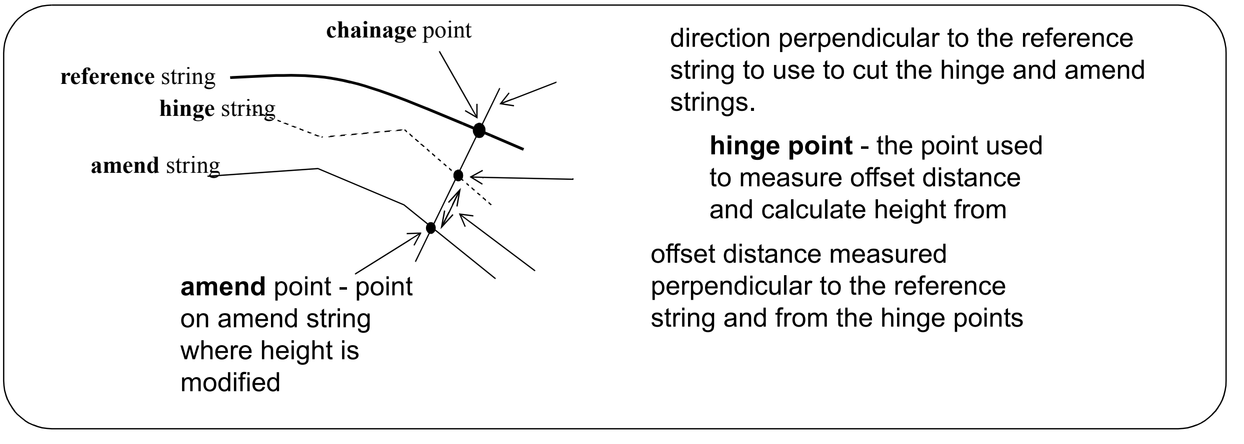
Unlike the other MTF options, the **string modifier** options modify the *heights* on an ***existing*** string called the ***amend string***.

Like the apply options, a reference string is used to define chainage and what is perpendicular at each chainage. A hinge string is used to define offsets and heights. If no hinge string is selected, the reference string is also used as the hinge string.

For a given chainage on the reference string (chainage point), a line perpendicular to the reference string is constructed to cut through the hinge string and the amend string (the cut points are called the ***hinge*** point and the ***amend*** point).

Horizontal offset is defined as the offset value from the hinge point, and the height of the ***amend*** point is a modification of the height at the hinge point.

The string modifier options create and/or modify the heights of the *amend* points on the amend string.



The *height* of the amend point is *derived from the hinge point* by starting with the height of the hinge point and applying a crossfall to it across the offset distance from the hinge string to the amend string.

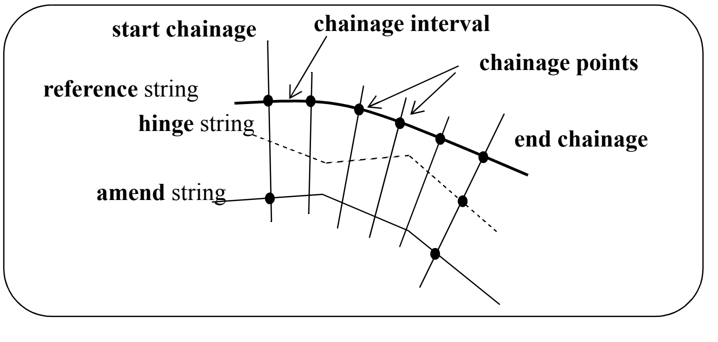
The cross fall can be defined by either:

(a) giving the cross fall

or

(b) using the cross fall between two user selected strings.

For the string options, the length of string being modified is restricted by giving a start and end reference chainage. A chainage interval can also be specified to define extra reference chainage points to use between the start and end chainages.



How and where the amend string is modified depends on the *type* of string and whether the chainage interval is blank (null) or not.

What Points are Modified

For 3d, 4d, polyline and super strings:

If the chainage interval is not blank:

(a) The start and end chainage points are projected *from the reference string* and ***inserted*** into the ***amend*** string. The *heights* of the inserted points are defined by the string modifier command.

(b) The chainage points at the given chainage interval are also projected from the reference string and ***inserted*** into the ***amend*** string. The *heights* of the inserted points are defined by the string modifier command.

(c) Finally, the ***vertices*** of the ***amend*** string are dropped perpendicularly *back onto* the hinge and reference string and then those points used to modify the *heights* of the same vertices of the amend string.

If the chainage interval is blank:

(a) The start and end chainage points are projected *from the reference string* and ***inserted*** into the ***amend*** string. The *heights* of the inserted points are defined by the string modifier command.

(b) The ***vertices*** of the ***amend*** string are dropped perpendicular *back onto* the hinge and reference string and then those points used to modify the *heights* of the same vertices of the amend string.

For Alignment strings:

If the chainage interval is not blank:

(a) All the vertical geometry is removed between the start and end chainages. That is, the vertical intersection points (VIP's) and their associated vertical curves are removed.

(b) The start and end chainage points are projected *from the reference string* and ***vertical intersection points (VIP's) inserted*** into the ***amend*** string. The *height* of the inserted VIP points are defined by the string modifier command.

If the chainage interval is blank:

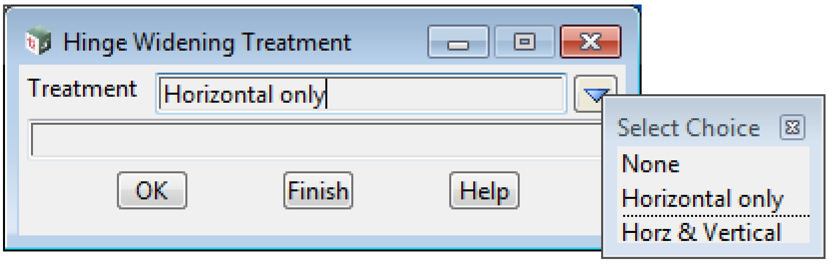
(a) The start and end chainage points are projected *from the reference string* and ***vertical intersection points (VIP's) inserted*** into the ***amend*** string. The *heights* of the inserted VIP points are defined by the string modifier command.

(b) The ***vertical intersection points (VIP's)*** of the ***amend*** string are dropped perpendicularly *back onto* the hinge and reference strings and then those points used to modify the *heights* of the same VIP's of the amend string. The vertical curves for the VIP's are not changed.

Return to 21.2.6.4 MTF String Modifiers or 21.2.1 MTF Hinge Modifiers

21.2.6.6 MTF Hinge Widening Treatment

The **Hinge Widening Treatment** specified in the **Hinge Widening Treatment** panel.



The choices are:

None -

Horizontal only -.

Horizontal & Vertical -.

OK - selecting *OK* sets the **Treatment**.

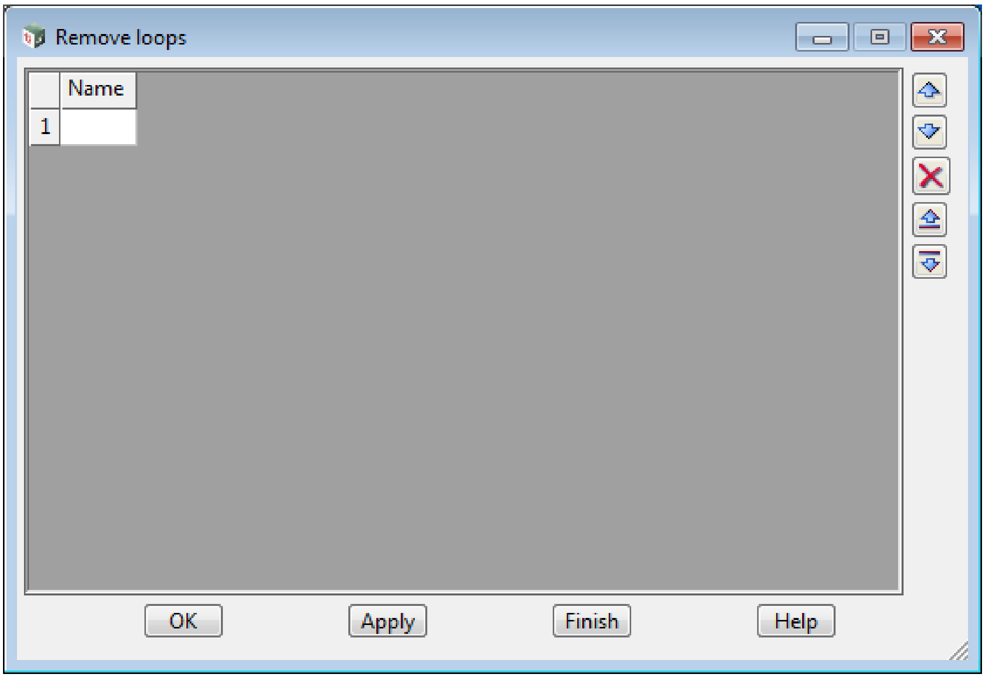
Continue to the next section 21.2.6.7 MTF Remove Loops or return to 21.2.6 MTF More.

21.2.6.7 MTF Remove Loops

The Loops options tries to remove loops in nominated strings generated by the Apply MTF.

Important Note - the sections generated from the Apply MTF will not be modified and will still intersect each other. In that case, the sections can be generated after the ***Apply MTF*** by cutting through strings. See 28.9.13.2 Cuts by Centreline.

Selecting **Loops** from the menu displays the **Remove loops** panel.



Name column header

names of the strings to process to try and remove any loops from. The names can include wild cards (*) and wild characters (!).

Continue to the next section 21.2.6.8 MTF Stripping or return to 21.2.6 MTF More.

21.2.6.8 MTF Stripping

A fixed stripping depth can be specified in the **Apply** option whilst the **Apply MTF** option allows for stripping depths that can **vary along the design**.

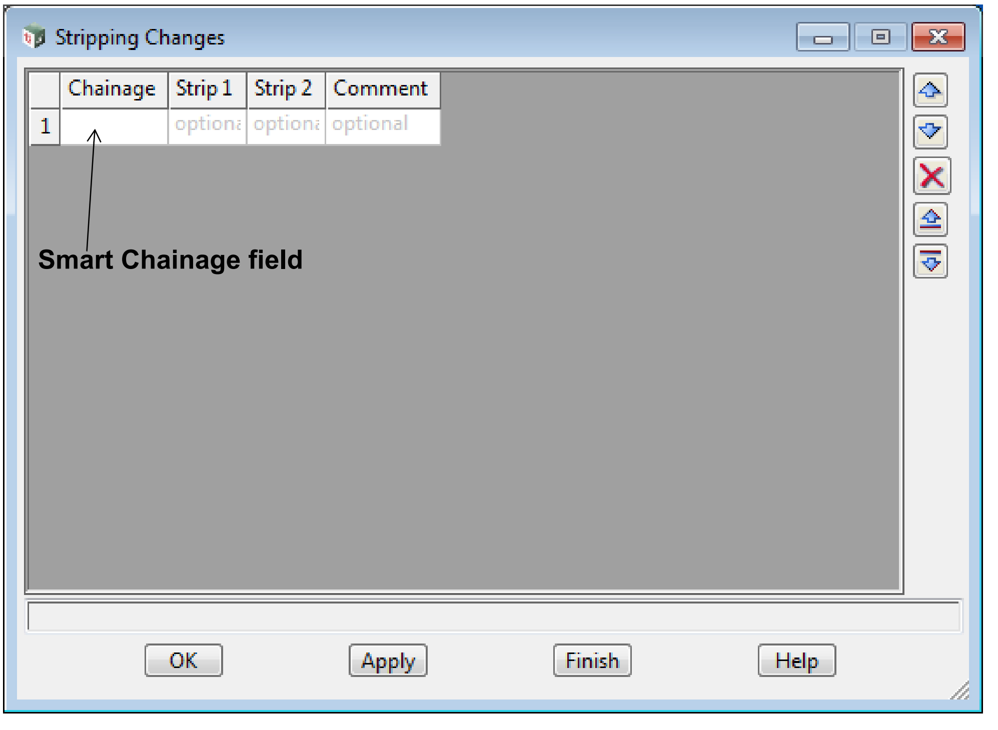
If a **non-zero stripping depth** exists at a chainage, the **cut** and **fill** calculations are done with respect to the section through the tin **dropped in height** by the stripping depth.

However, the **design strings** are generated by battering into the **unstripped** surface, and the stripping volume is the volume between the stripped and unstripped surface for the design.

Stripping depths are defined for chainages along the reference string and can be linearly interpolated between chainages.

NOTE: Cut and fill areas and volumes are adjusted for the stripping depths. The total stripping volume is given at the end of the volumes report.

Selecting the **Stripping** option on the **MTF edit** menu brings up the **Stripping Changes** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Stripping Chainages Grid

For information on the general operation of a grid including the icons on the right hand side, see 4.19.6 Grids in Panels.

Chainage smart chainage box

list of chainages for defining stripping depth.

*If you right click in the **Chainage** column or select **Browse** if the Browse menu comes up, then a* **Chainage** *panel comes up which has the most of the standard MTF choices for **Smart Chainages**. See* 21.2.1 MTF Hinge Modifiers.

Strip depth 1 real box measures menu

strip depth to apply at the chainage given for this row of information. If strip depth 2 is blank, then strip depth 1 is applied until the next chainage in the chainage column.

Strip depth 2 real box measures menu

*if **non-blank**, the strip depth to linearly interpolate to and finish with at the next chainage in the chainage column.*
if blank, then strip depth 1 applies to the next chainage in the chainage column.
In the mtf file, the two depths will be separated by a comma.

Comment column header

comment for the row.

Buttons at Bottom

OK/Apply button

OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

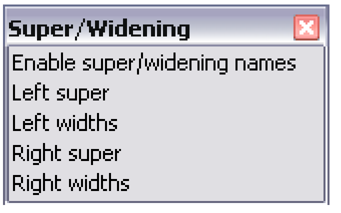
Continue to the next section 21.2.6.9 MTF Super/Widening or return to 21.2.6 MTF More.

21.2.6.9 MTF Super/Widening

The ***Super Alignment*** string can define the super elevation and widening for a design.

This super elevation and widening can be applied to the strings defined by the Apply MTF rather than having to use MTF crossfall and width modifiers.

This section defines which strings in the apply have super elevation/widening from the super alignment applied to them



For the options see

Enable super/widening names 21.2.6.9.1 Enable Super/Widening Names

Left super 21.2.6.9.2 Left Super

Left widths 21.2.6.9.3 Left Widths

Right super 21.2.6.9.4 Right Super

Right widths 21.2.6.9.5 Right Widths

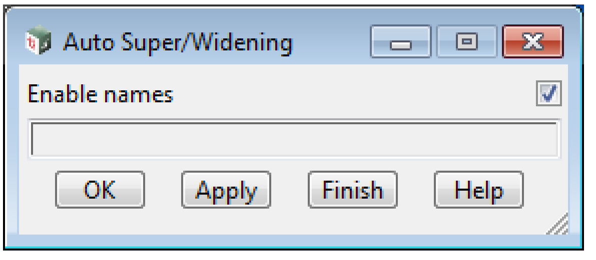
21.2.6.9.1 Enable Super/Widening Names

For ***12d Model*** Version 8, super elevation/widening from the super alignment was applied to strings in the fixed section of the template that were defined by crossfall.

For ***12d Model*** 9, this method was replaced by being able to ***define by string name***, which strings have super elevation and/or widening applied to them.

For backward compatibility, this option allows the user to either use string names, or to revert to the cross fall method used in version 8. The default is to use string names.

Selecting **Enable super/widening names** from the menu displays the **Auto Super/Widening** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Enable names	tick box	tick	

if ticked, the super elevation/widening in the super string is applied to the string names as given in the ***Left/Right super*** *and/or* ***Left/Right widths*** *options.*

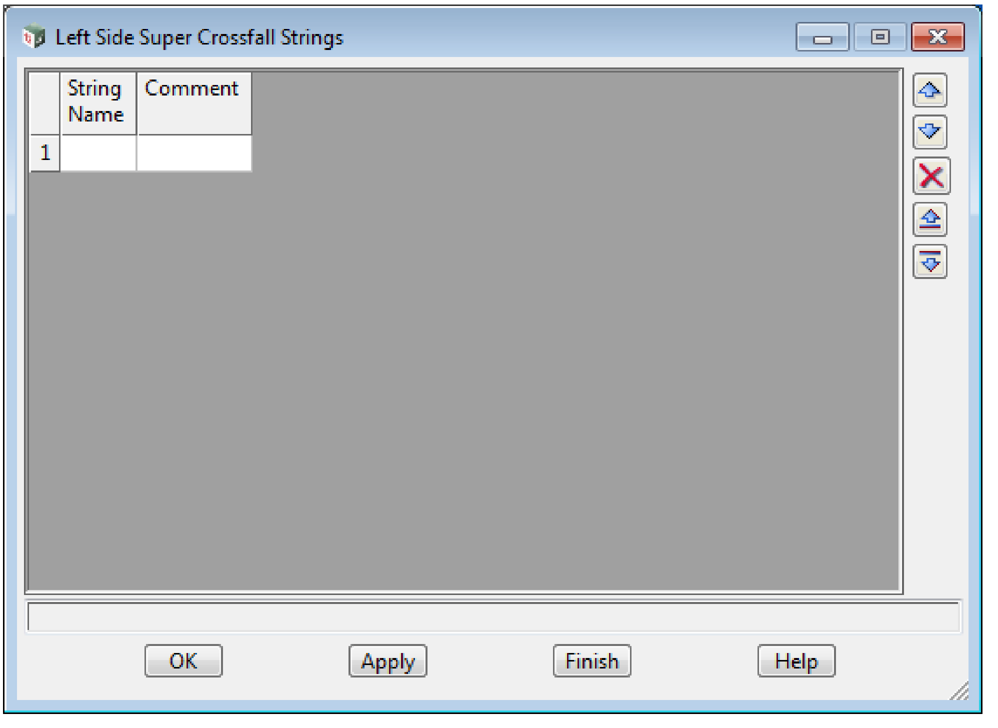
If not ticked, the super elevation/widening in the super string is applied to the string from the fixed template that are defined by crossfall.

Continue to the next section 21.2.6.9.2 Left Super or return to 21.2.6.9 MTF Super/Widening or return to 21.2.6 MTF More.

21.2.6.9.2 Left Super

This table defines the strings that have the left super elevation from the super alignment applied to them.

Selecting **Left Super** from the menu displays the **Left Side Super Crossfall Strings** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
String name	column header		

names of the strings to apply the left super elevation of the super alignment to.

Comment	column header		

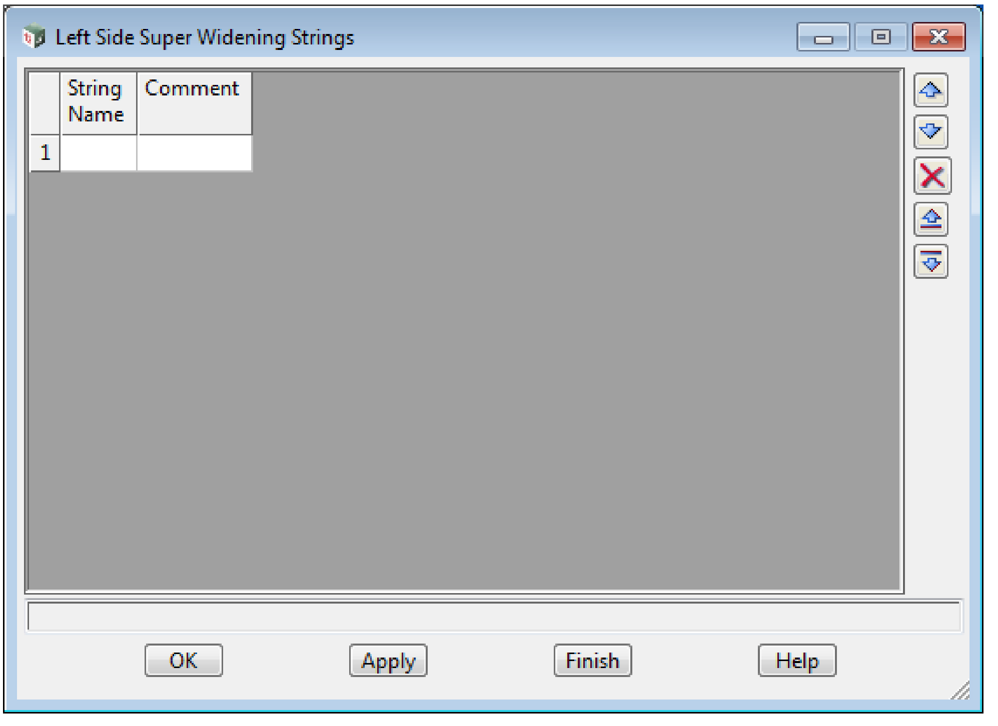
comment to be recorded.

Continue to the next section 21.2.6.9.3 Left Widths or return to 21.2.6.9 MTF Super/Widening or return to 21.2.6 MTF More.

21.2.6.9.3 Left Widths

This table defines the strings that have the left widening from the super alignment applied to them.

Selecting **Left widths** from the menu displays the **Left Side Super Widening Strings** panel.



Field Description | Type | Defaults | Pop-Up

String name column header

names of the strings to apply the left widening of the super alignment to.

Comment column header

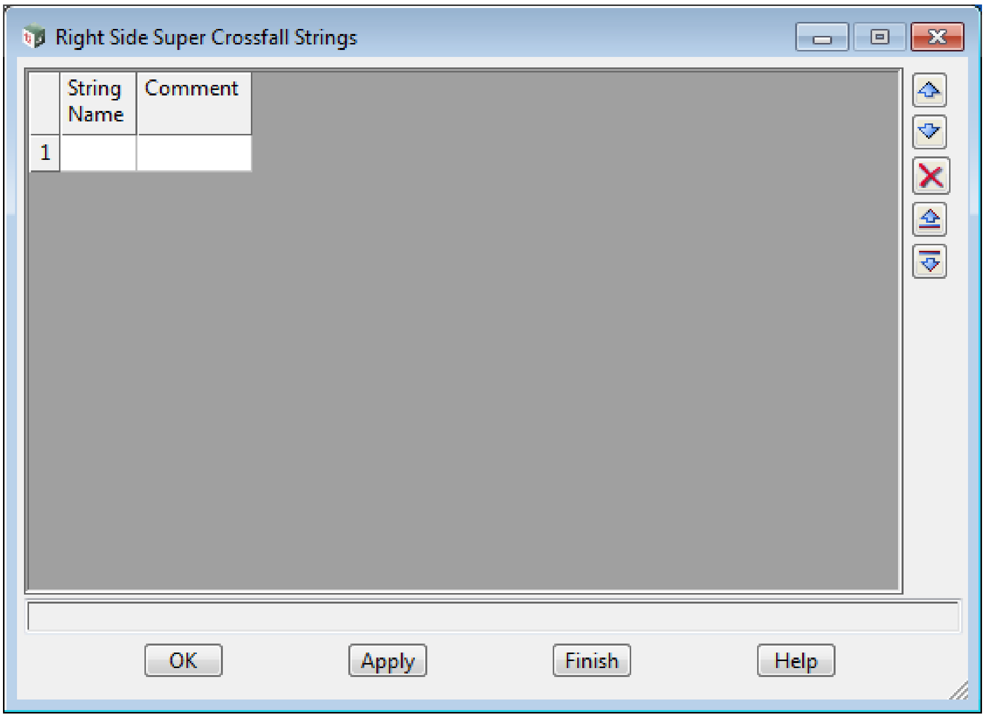
comment to be recorded.

Continue to the next section 21.2.6.9.4 Right Super or return to 21.2.6.9 MTF Super/Widening or return to 21.2.6 MTF More.

21.2.6.9.4 Right Super

This table defines the strings that have the right super elevation from the super alignment applied to them.

Selecting **Right super** from the menu displays the **Right Side Super Crossfall Strings** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
String name	column header		

names of the strings to apply the right super elevation of the super alignment to.

Field Description	Type	Defaults	Pop-Up
Comment	column header		

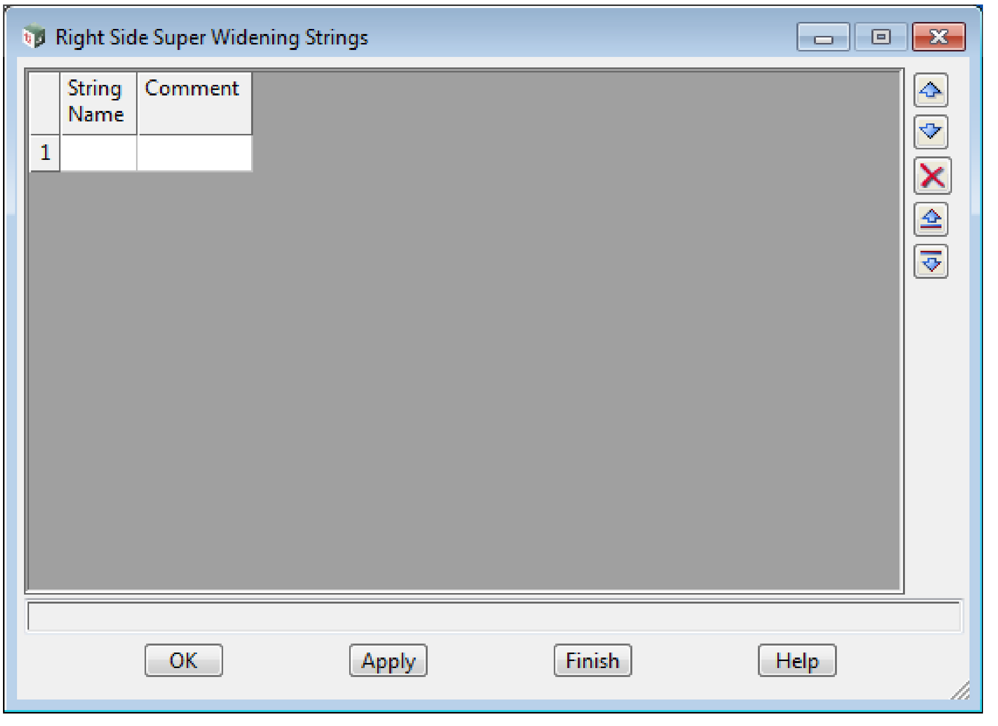
comment to be recorded.

Continue to the next section 21.2.6.9.5 Right Widths or return to 21.2.6.9 MTF Super/Widening or return to 21.2.6 MTF More.

21.2.6.9.5 Right Widths

This table defines the strings that have the right widening from the super alignment applied to them.

Selecting **Right widths** from the menu displays the **Right Side Super Widening Strings** panel.



String name column header

names of the strings to apply the right widening of the super alignment to.

Comment column header

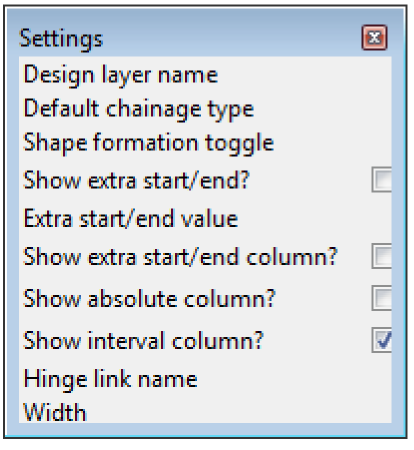
comment to be recorded.

Return to 21.2.6.9 MTF Super/Widening or.21.2.6 MTF More

21.2.7 MTF Settings

Position of menu: **Design =>MTF =>Settings**

The **Settings** walk- right menu is



set the name of Design Layer
set the default chainage type
set the shape formation toggle
set whether the **Extra start** and **Extra end** tick boxes are shown
set the distance for placing extra sections before the start & end chainages
set whether to show **Extra start** and **Extra end** columns
set whether to show **Absolute** column
set whether to show **Interval** column
set the name of the link for the hinge string
set the distance to search along a section when trying to find strings used in any of the MTF commands

For the options see:

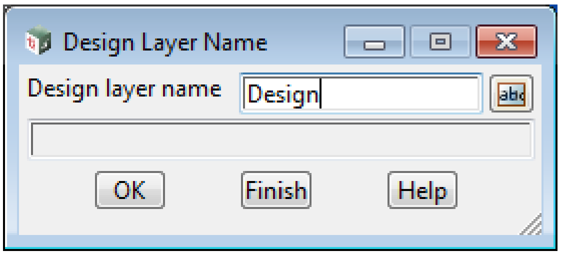
Design layer name 21.2.7.1 MTF Setting Design layer name
Design chainage type 21.2.7.2 MTF Setting Default Chainage Type
Shape formation toggle 21.2.7.3 MTF Setting Shape Formation Toggle
Show extra start/end? 21.2.7.4 MTF Setting Show Extra Start/End
Extra start/end value 21.2.7.5 MTF Setting Extra Start/End value
Show start/end column? 21.2.7.6 MTF Setting Show Extra Start/End column
Show absolute column? 21.2.7.7 MTF Setting Show Absolute Column
Show interval column? 21.2.7.8 MTF Setting Show Interval Column
Hinge link name 21.2.7.9 MTF Setting Hinge Link Name
Width 21.2.7.10 MTF Width

21.2.7.1 MTF Setting Design layer name

In ***12d Model***, the *MTF Modifiers* can produce any number of named layers but there is one special layer referred to as the **Design Layer**, that is used to produce design x sections, design strings, the road tin, *etc.*

Any MTF points in the Design Layer will automatically generate points on the design x sections, and design strings will be generated without them having to be done by the **Create =>Strings** MTF Modifier command.

The name of the special Design Layer can be changed by clicking on **Design layer name** to bring up the **Design Layer Name** panel.



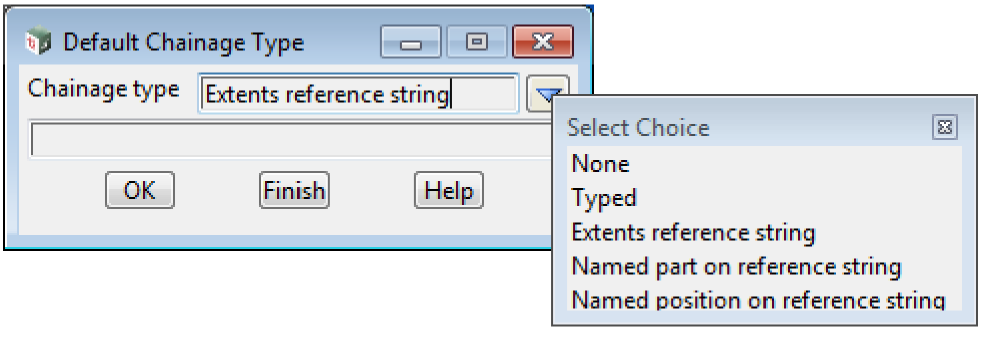
Design layer name - the name used for the special Design Layer.

OK - selecting *OK* sets the default name for the design layer.

Continue to the next section 21.2.7.2 MTF Setting Default Chainage Type or return to 21.2.7 MTF Settings.

21.2.7.2 MTF Setting Default Chainage Type

The **Default Chainage Type** panel specifies the default Smart Chainage Mode to use in the MTF Modifier commands in this MTF file.



The choices are:

None - no Smart Chainage choice is used as the default.

Typed - *Typed* is the default for Smart Chainages.

Extents reference string - the start of the reference string is used for *Start modes* and the end of the reference string is used for *End modes*.

Named part on reference string - *named part on reference string* is the default for Smart Chainages.

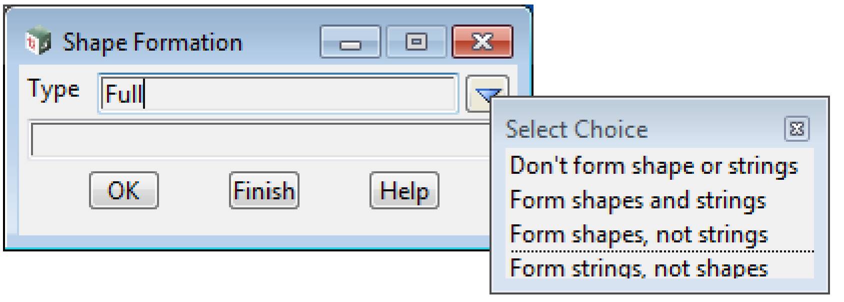
Named position on reference string - *named position on reference string* is the default for Smart Chainages.

OK - selecting *OK* sets the given *Chainage type* as the default.

Continue to the next section 21.2.7.3 MTF Setting Shape Formation Toggle or return to 21.2.7 MTF Settings.

21.2.7.3 MTF Setting Shape Formation Toggle

Creating shapes and strings can be a time consuming process so it is possible to set modes for if it occurs or not.



The choices are:

Don't form shapes or strings - don't create shapes or strings that are referred to in the **Create=> Shapes** and **Create =>Strings** MTF Modifier commands.

Form shapes or strings - generated the shapes and strings as specified in the **Create => Shapes** and **Create =>Strings** MTF Modifier commands.

Form shapes, not strings - generated the shapes as specified in the **Create => Shapes** MTF Modifier command but do not generated the strings referred to in the **Create =>Strings** MTF Modifier command.

Form strings, not shapes - generated the strings as specified in the **Create => Strings** MTF Modifier command but do not generated the shapes referred to in the **Create =>Shapes** MTF Modifier command.

OK - selecting *OK* sets the given *Shape Formation* **Type**.

Continue to the next section 21.2.7.4 MTF Setting Show Extra Start/End or return to 21.2.7 MTF Settings.

21.2.7.4 MTF Setting Show Extra Start/End

With MTF Modifier commands, a command applies from the start chainage and finished just before the end chainage. This is to prevent a doubling up in the definitions for the chainage between the end of one command and the beginning of the following command.

To make values change correctly between the MTF definitions at the start and end chainages, for most MTF Modifier commands there are **Extra start** and **Extra end** tick boxes,.

If **Extra start** is ticked on and the command is not the first, a section is created **Extra start/end value before** the Start chainage of the command. Because the definition for this extra section is given by the **previous** command. This ensures that the previous command applies to almost to the this Start chainage.

If **Extra end** is ticked on, a section is created **Extra start/end value** before the End chainage of the command. The definition of this extra section is given by the current command. This ensures that the current command applies almost to the end, especially when it is the last command and there is no next Start chainage.

For most situations, the **Extra start** and **Extra end** tick boxes for an **MTF Modifier Command** can be left ticked on and the default is to have them both ticked on.

If **Show Extra start/end?** is **ticked**, the **Extra start** and **Extra end** tick boxes are shown on the **MTF Modifier command** panels.

If **Show Extra start/end?** is **not ticked**, the **Extra start** and **Extra end** tick boxes are **not** shown on the **MTF Modifier command** panels

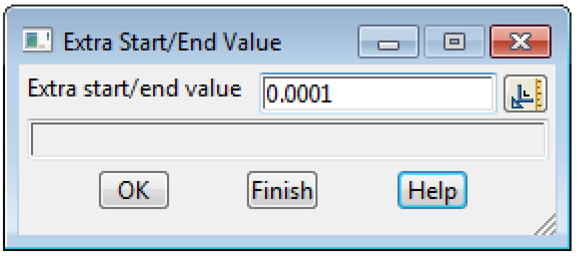
Notes:

1. The display of the **Extra start** and **Extra end** columns in the **Left/Right MTF Modifiers** panel is controlled by the 21.2.7.6 MTF Setting Show Extra Start/End column.
2. The **Extra start/end value** is set by 21.2.7.5 MTF Setting Extra Start/End value.

Continue to the next section 21.2.7.5 MTF Setting Extra Start/End value or return to 21.2.7 MTF Settings.

21.2.7.5 MTF Setting Extra Start/End value

For ***12d Model*** itself, the **Extra start/end value** does not need to be changed but when writing out sections that are to be loaded into some software products, the sections sometimes can't be closer than a fixed value in the other software. In that case modifying the **Extra start/end value** will allow you to get over that restriction.



Extra start/end value - the distance to use for placing extra sections before the start and end chainages.

OK - selecting *OK* sets the **Extra start/end value**.

Continue to the next section 21.2.7.6 MTF Setting Show Extra Start/End column or return to 21.2.7 MTF Settings.

21.2.7.6 MTF Setting Show Extra Start/End column

If **Show extra start/end column?** is **ticked**, the **Extra Start** and **Extra Start** columns are shown in the **Left/Right MTF Modifiers** panel.

If **Show extra start/end column?** is **not** ticked the **Extra Start** and **Extra Start** columns are **not** shown in the **Left/Right MTF Modifiers** panel.

Note that the Extra start/Extra end tick boxes appearing in the MTF Modifier Commands is controlled by the 21.2.7.4 MTF Setting Show Extra Start/End.

21.2.7.7 MTF Setting Show Absolute Column

The **Absolute** column is a condensed way of displaying the value of the **New value usage Absolute** field in the MTF command **Modify Link**.

If **Show absolute?** is **ticked**, the **Absolute** column is shown in the **Left/Right MTF Modifiers** panel.

If **Show absolute column?** the **Absolute** column is **not** shown in the **Left/Right MTF Modifiers** panel.

Note that in both cases, the **New value usage** field (which the Absolute column represents) **is** shown in the **MTF Modifier Link command** panels.

For an example of **New value usage** and **Absolute**, see 21.1.2.3 New Value Usage.

Continue to the next section 21.2.7.8 MTF Setting Show Interval Column or return to 21.2.7 MTF Settings.

21.2.7.8 MTF Setting Show Interval Column

If **Show interval column?** is **ticked**, the **Interval column** is shown in the **Left/Right MTF Modifiers** panel.

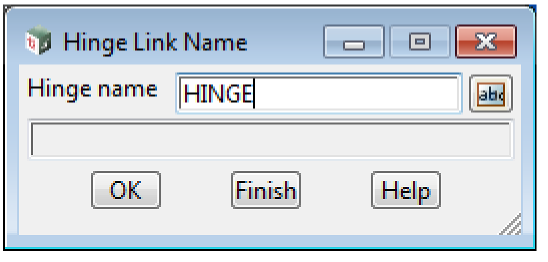
If **Show interval column?** the **Interval column** is **not** shown in the **Left/Right MTF Modifiers** panel.

Note that in both cases, the **Interval** field **is** shown on the **MTF Modifier command** panels.

Continue to the next section 21.2.7.9 MTF Setting Hinge Link Name or return to 21.2.7 MTF Settings.

21.2.7.9 MTF Setting Hinge Link Name

The name of the link for the Hinge string can now be given a user defined name.



Hinge name - the name used for the hinge link.

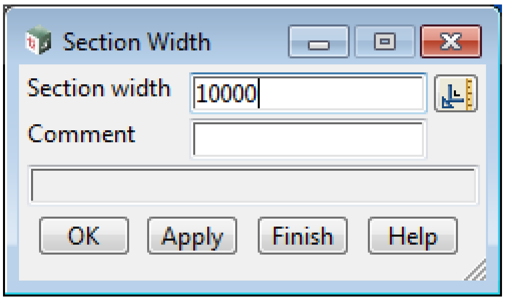
OK - selecting *OK* sets the **Hinge name**.

Continue to the next section 21.2.7.10 MTF Width or return to 21.2.7 MTF Settings.

21.2.7.10 MTF Width

A **section width** is used to limit the distance to search along a section when trying to find strings used in any of the MTF commands.

Selecting **Width** from the **MTF edit** men brings up the **Section Width** panel for setting the width.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Section width	input	10000	

__Section width__ defines the perpendicular offset from the Hinge string. It is used to limit the search distance for strings when performing MTF modifier commands.

Comment	column header		

comment to add to the end of the line. In the file, the comment will be preceded by //.

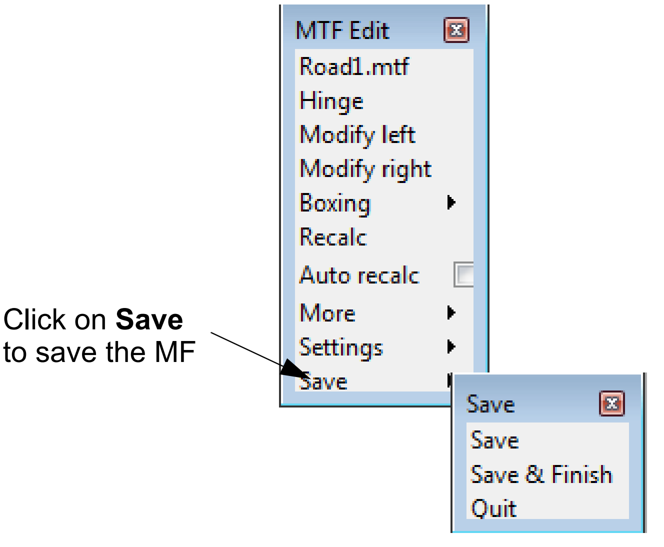
OK/Apply	button		

OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

Return to 21.2.7 MTF Settings.

21.2.8 MTF Save

Clicking **Save** on the **MTF Edit** menu (without walking right) saves the MTF file.



Clicking on the options on the walk right menu will:

If ***Save*** is clicked, the MTF is saved to disk and the **MTF Edit** panel is left on the screen.

If ***Quit*** is clicked, the MTF is **not** saved to disk and the **MTF Edit** panel is closed.

If ***Save & Finish*** is clicked, the MTF **is** saved to disk and the **MTF Edit** panel then closed.

Return to 21.1 The Modifiers and Templates File - MTF.

21.3 Options Not Yet Re-documented for V11

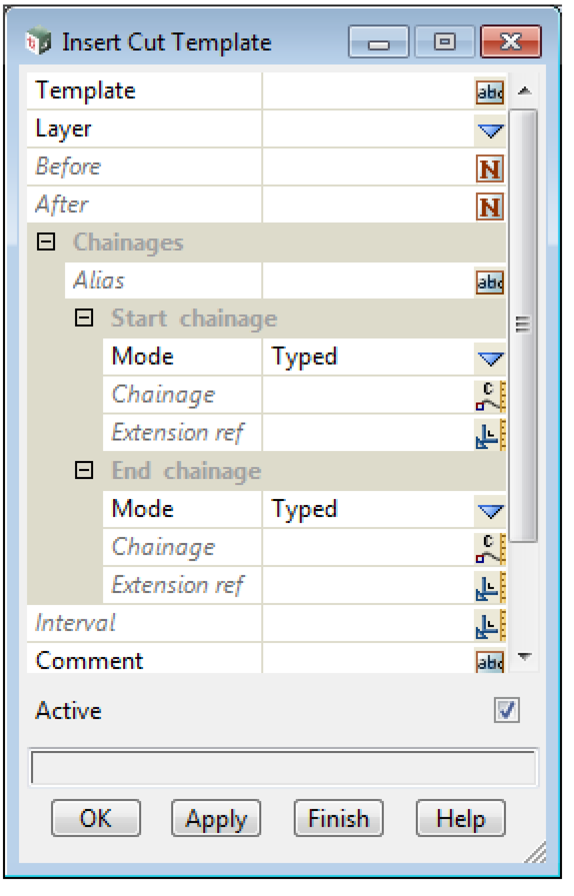
V11 introduced the concept of **Layers** and also removed the restriction that links could only be modified by the two of the three width, height, xfall or slope, that they were defined by.

These options have not yet been documented for the V11 changes.

21.3.0.1 Insert Cut Links from a Template

All the cut links from an existing template can be inserted by the **Insert cut template** command.

Selecting **Insert cut template** brings up the **Insert Cut Template** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Template	template box		select template menu

the name of the template to take the cut links from.

Field Description	Type	Defaults	Pop-Up
Before	input		select name menu

*if **non-blank**, the cut links from the selected template are inserted before the given link.*
*If **blank**, the cut links from the selected template are appended to the end of the existing cut links.*

Field Description	Type	Defaults	Pop-Up
Start/End mode	choice box	Start (ref)/End (ref)	

defines the start/end chainages for inserting the new template links.
For more information on Start/End mode, see 21.2.1 MTF Hinge Modifiers

Field Description	Type	Defaults	Pop-Up
Interval	input		

*if **non blank**, the interval to use to create cross sections and strings over the given chainage range.*

*If **blank**, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

Extra start/end tick box

*if **ticked**, add an extra x-section 0.1 mm **before** the start/end chainage.*

Comment input

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active tick box tick

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

OK/Apply button

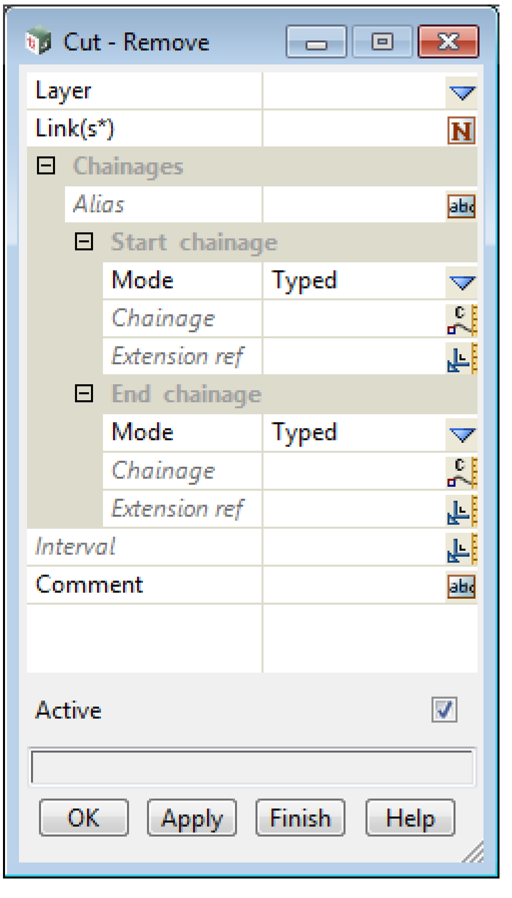
OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

Continue to the next section 21.3.0.2 Cut Link - Remove or return to 21.3.0.1 Insert Cut Links from a Template

21.3.0.2 Cut Link - Remove

Cut links can be deleted between given chainages by using the **Cut Remove** command.

Selecting **Remove** brings up the **Cut - Remove** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu

name of the link to removed.

Start/End mode choice box Start (ref)/End (ref)

defines the start/end chainages for removing the cut link.
For more information on Start/End mode, see 21.2.1 MTF Hinge Modifiers

Interval input

*if **non blank**, the interval to use to create cross sections and strings over the given chainage range.*
*If **blank**, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

Extra start/end tick box

*if **ticked**, add an extra x-section 0.1 mm **before** the start/end chainage.*

Comment input

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active tick box tick

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

OK/Apply button

OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

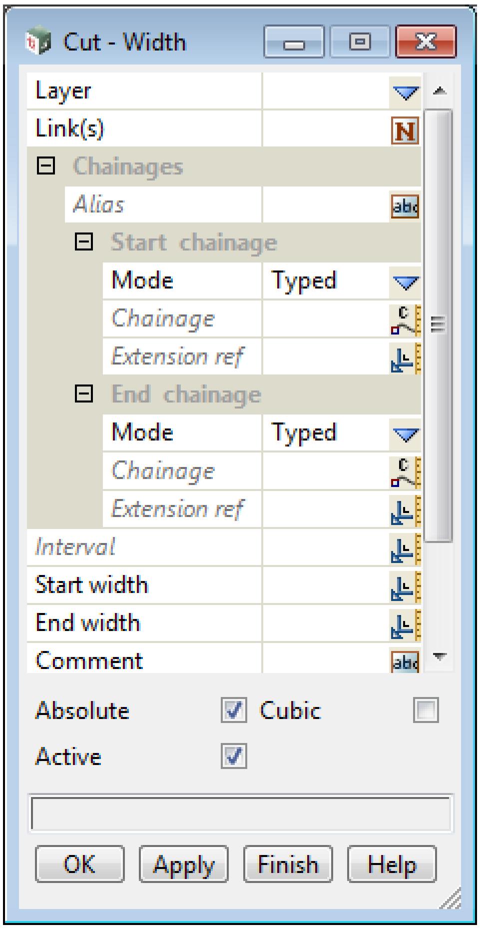
Continue to the next section 21.3.0.3 Cut Link - Modifier Width or Height or return to 21.3.0.1 Insert Cut Links from a Template.

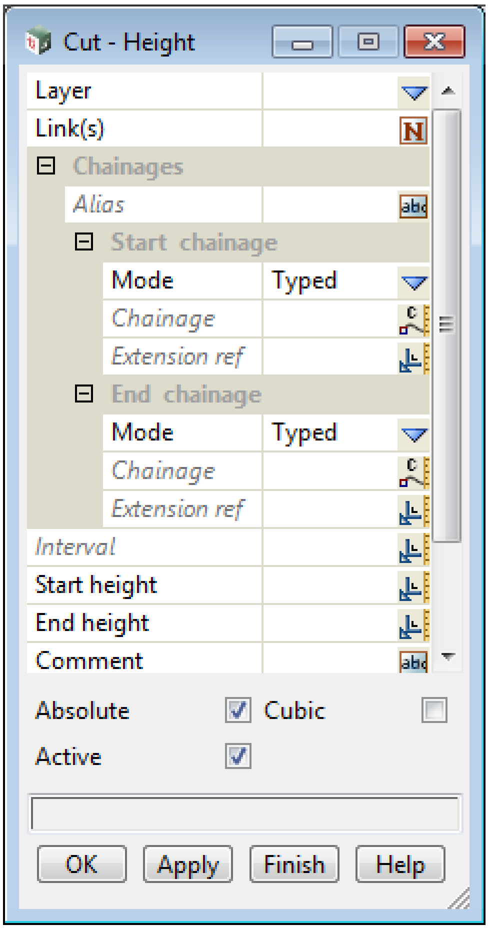
21.3.0.3 Cut Link - Modifier Width or Height

The **cut width** modifier is used to modify the width of variable cut links originally defined by width. That is, links defined by *width and slope* or *width and height*.

Similarly the **cut height** modifier is used to modify the height of variable cut links originally defined by height. That is, links defined by *height and width* or *height and slope*.

Selecting **Width or Height** brings up the **Cut - Width** and **Cut - Height** panels respectively.





The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu

name of the link to modify.

Start/End mode	choice box	Start (ref)/End (ref)	

defines the start/end chainages for modifying the links.
For more information on Start/End mode, see 21.2.1 MTF Hinge Modifiers

Interval	input		

if ***non blank****, the interval to use to create cross sections and strings over the given chainage range.*
If ***blank****, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

Start/End width/height	input		measures menu

start/end width/height for modifying the link.

Extra start/end	tick box		

if ***ticked****, add an extra x-section 0.1 mm* ***before*** *the start/end chainage.*

Absolute	tick box	tick	

if ***ticked****, the width/height is set to the values given in the* **start** *and* **end value** *fields.*
if ***not ticked****, the values given in the* **start** *and* **end value** *fields are added to the existing widths/heights.*

Cubic	tick box		

*if **ticked**, the width/height is varied as a reverse cubic between the start and end chainages.*
*if **not ticked**, the width/height is varied linearly between the start and end chainages.*

Comment input

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active tick box tick

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

OK/Apply button

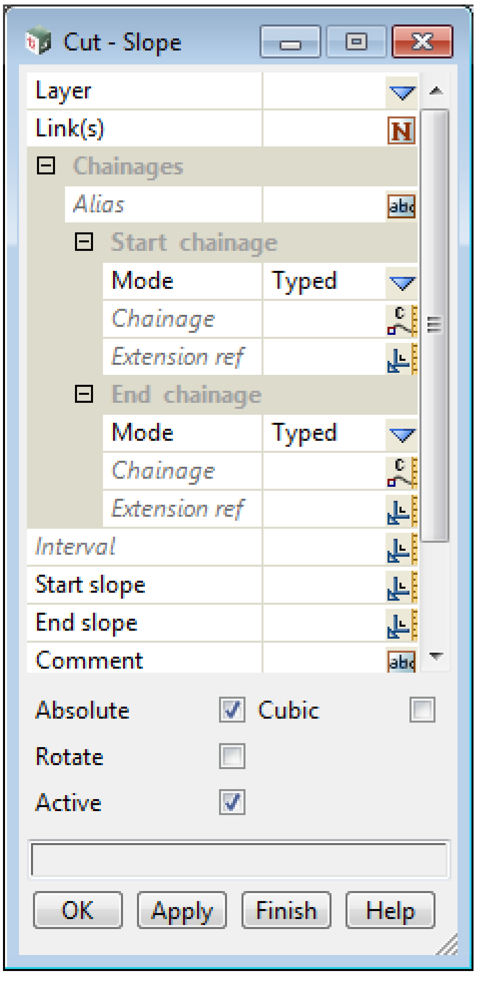
OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

Continue to the next section 21.3.0.4 Cut Link - Modify Slope or return to 21.3.0.1 Insert Cut Links from a Template.

21.3.0.4 Cut Link - Modify Slope

The **cut slope** modifier is used to modify the slope of variable cut links originally defined by slope. That is, links defined by *slope and width* or *slope and height*.

Selecting **Slope** brings up the **Cut - Slope** panel



The fields and buttons used in this panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Link name input select name menu

name of the link to modify.

Start/End mode choice box Start (ref)/End (ref)

defines the start/end chainages for modifying the link.
For more information on Start/End mode, see 21.2.1 MTF Hinge Modifiers

Interval input

*if **non blank**, the interval to use to create cross sections and strings over the given chainage range.*
*If **blank**, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

Start/End slope input measures menu

start/end slope for modifying the link.

Extra start/end tick box

*if **ticked**, add an extra x-section 0.1 mm **before** the start/end chainage.*

Absolute tick box tick

*if **ticked**, the slope is set to the values given in the* **start** *and* **end slope** *fields.*
*if **not ticked**, the slopes given in the* **start** *and* **end value** *fields are added to the existing slopes.*

Cubic tick box:

*Only **none** or **one** of **Cubic** and **Rotate** can be set to **tick.***

*The **default** is **none** - that is, **neither is ticked** and in the default case, the slope is varied **linearly with respect to slope** between the start and end chainages.*

Cubic tick box

*if **ticked**, the slope is varied as a reverse cubic between the start and end chainages.*

Rotate tick box

*if **ticked**, the slope is varied linearly **with respect to** the **angle,** between the start and end chainages.*

Comment input

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active tick box tick

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

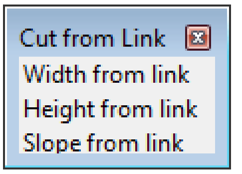
OK/Apply button

OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

Continue to the next section 21.3.0.5 Cut Link - from Link or return to 21.3.0.1 Insert Cut Links from a Template.

21.3.0.5 Cut Link - from Link

The *Cut* **from link** walk-right brings up the **Cut from Link** menu with options to take the width, height or slope from another link.



For *Width/Height/Slope from link*, go to the next section [Cut Width, Height or Slope from a Link](#)

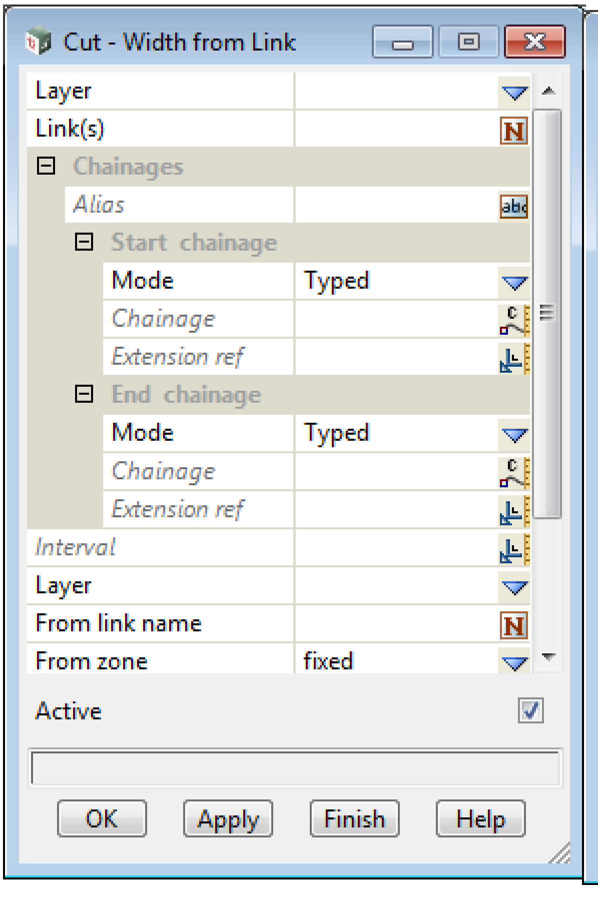
Cut Width, Height or Slope from a Link

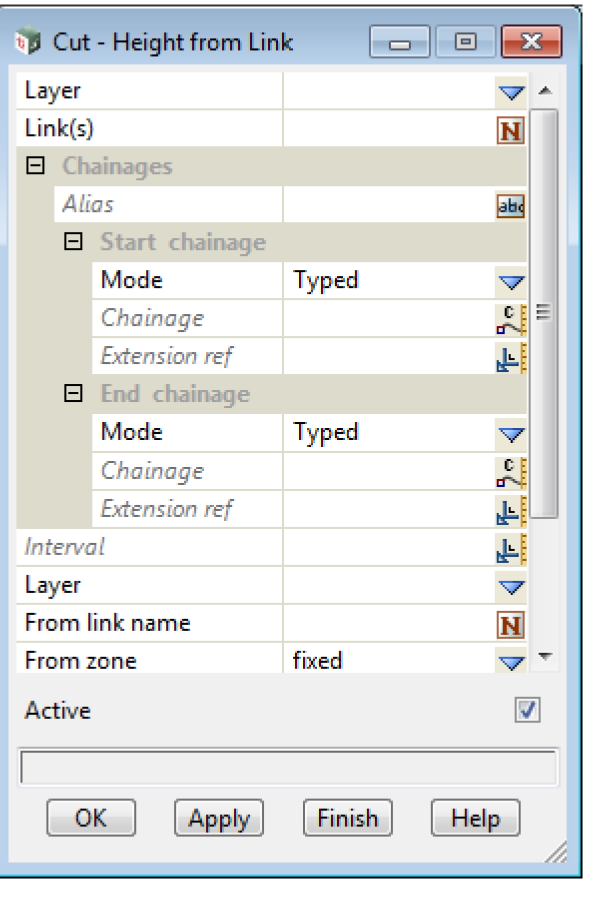
The **Width from link** modifier is used to modify the width of cut links originally defined by width to be the same width as another link. That is, the width of the link is a copy of the width of another link.

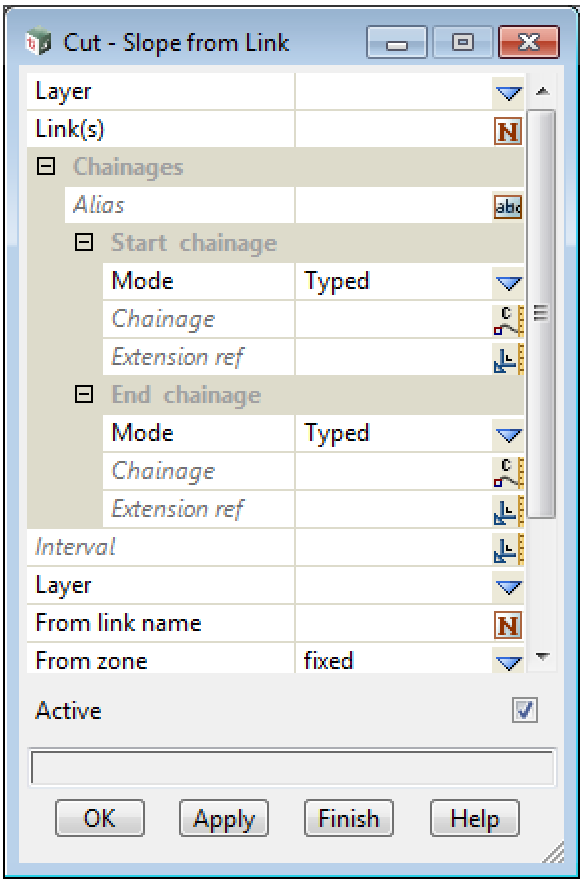
The **Height from link** modifier is used to modify the height of cut links originally defined by height to be the same height as another link. That is, the height of the link is a copy of the height of another link.

The **Slope from link** modifier is used to modify the slope of cut links originally defined by slope to be the same slope as another link. That is, the slope of the link is a copy of the slope of another link. The link to copy slope from can be defined in terms of slope or cross fall. If the link to copy is defined by cross fall, then the slope is calculated to match the cross fall.

Selecting **width from link, height from link or slope from link** brings up the **Cut - Width from Link**, **Cut Height - from Link** and **Cut - Slope from Link** panels respectively.







The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu
name of the link to modify.			
Start/End mode	choice box	Start (ref)/End (ref)	
defines the start/end chainages for modifying the link. *For more information on Start/End mode, see* 21.2.1 MTF Hinge Modifiers			
Interval	input		
*if **non blank**, the interval to use to create cross sections and strings over the given chainage range.* *If **blank**, the* **Section separation** *value from the* **Apply MTF** *panel is used.*			
From link name	input		select name menu
template link to take width/height/slope from.			
From zone	input	fixed	fixed, cut, fill
zone that the template link to take width/height/slope from, comes from.			
Extra start/end	tick box		
*if **ticked**, add an extra x-section 0.1 mm **before** the start/end chainage.*			
Comment	input		
comment to add to the end of the line. In the file, the comment will be preceded by //.			
Active	tick box	tick	
*if **ticked**, use this modifier.*			

*if **not ticked**, don't use this modifier.*

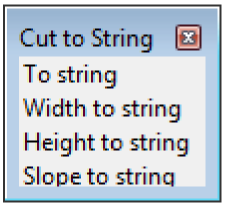
OK/Apply button

OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

Continue to the next section 21.3.0.6 Cut Link - to String or return to 21.3.0.1 Insert Cut Links from a Template.

21.3.0.6 Cut Link - to String

The *Cut* **to string** walk-right brings up the **Cut to String** menu with options to take the width, height or slope by going to another string.

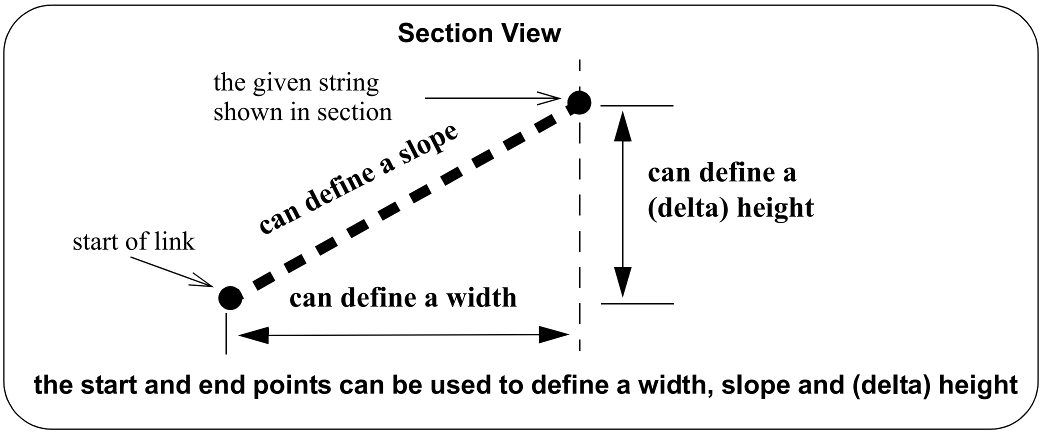


For *To string, go to* Cut Link - Modify To String
Width/Height/Slope to string Cut Link - Calculate Width, Height or Slope to a String

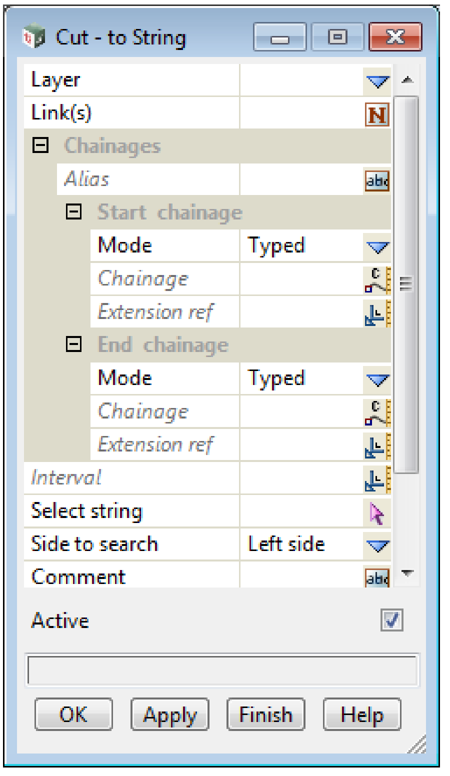
Cut Link - Modify To String

For any cut link, **To string** calculates the width, height and/or slope of the link needed to get from the start point of the link to the **to the selected string**.

However if the tin in the Apply MTF is cut before reaching the string, the ***cut link stops at the Apply MTF tin***.



Selecting **To string** bring up the **Cut - to String** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu

name of the link to modify.

Start/End mode	choice box	Start (ref)/End (ref)	

defines the start/end chainages for modifying the link
For more information on Start/End mode, see 21.2.1 MTF Hinge Modifiers ***except for this option****:*

When the ***Start mode*** *is* ***Start (ref),*** *or* ***Typed*** *and the chainage is* ***blank****, the modification begins at the low dropped chainage of the selected string.*

When the ***End mode*** *is* ***End (ref),*** *or* ***Typed*** *and the chainage is* ***blank****, the modification ends at the high dropped chainage of the selected string.*

Interval	input		

if ***non blank****, the interval to use to create cross sections and strings over the given chainage range. If* ***blank****, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

String	string-select		

select string to use for defining width/height/crossfall for the link.

Side to search	input	left side	left side, right side, both sides

side of the hinge string to start searching to find the string to define width/height/crossfall.

Extra start/end	tick box		

if ***ticked****, add an extra x-section 0.1 mm* ***before*** *the start/end chainage.*

Comment	input		

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active tick box tick

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

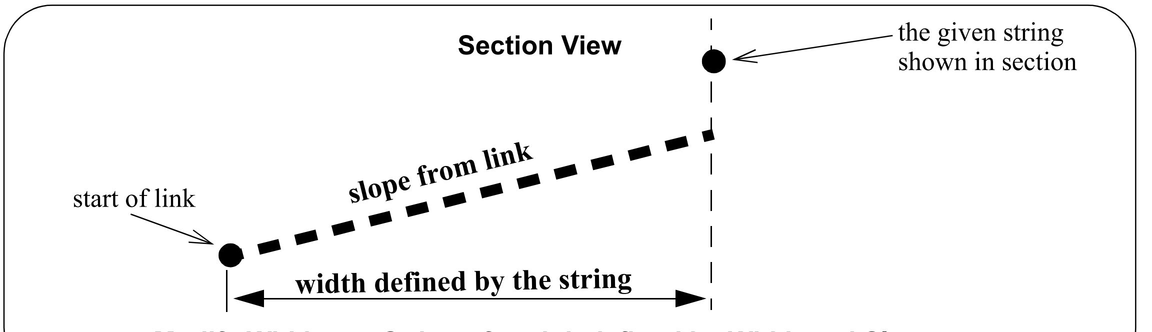
OK/Apply button

OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

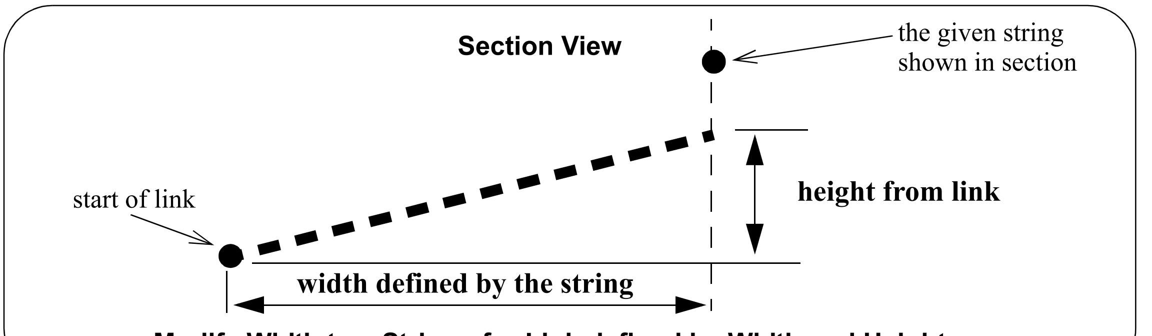
Cut Link - Calculate Width, Height or Slope to a String

Width to String:
For a cut link defined by *width and height* or *width and slope*, **Cut Width to string** calculates the **width** of the link as the width from the start point of the link, to the **to the selected string**. The *slope* or *height* is taken from the link. The option will give an error for a link defined by *height* and *slope*.

However if the tin in the Apply MTF is cut before reaching the string, the ***cut link stops at the Apply MTF tin***.



Modify Width to a String - for Link defined by Width and Slope

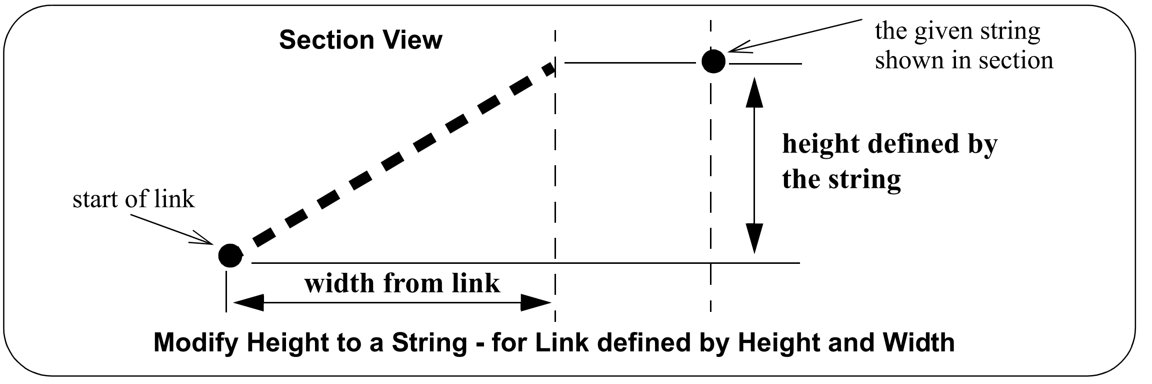


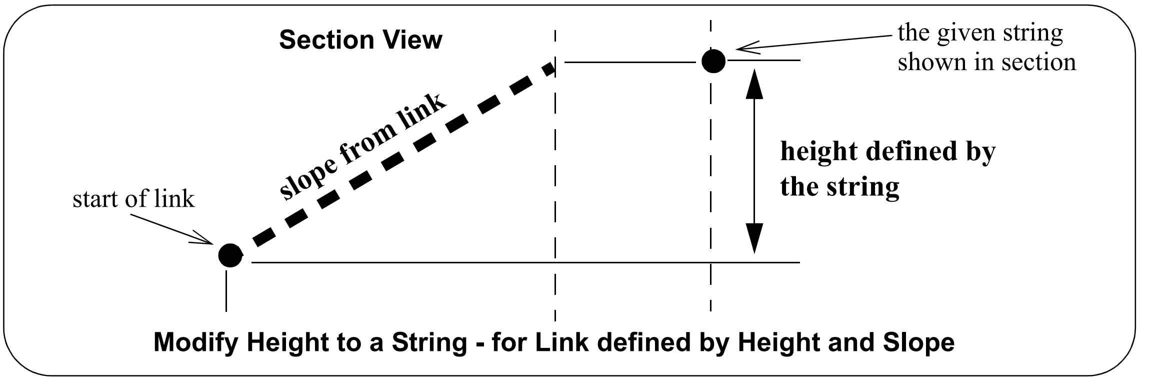
Modify Width to a String - for Link defined by Width and Height

Height to String:

For a cut link defined by *height and width* or *height and slope*, **Cut Height to string** calculates the **height** of the link as the difference in the height at the start point of the link, and the height **at the selected string**. The *slope* or *width* is taken from the link. The option will give an error for a link defined by *width* and *slope*.

However if the tin in the Apply MTF is cut before reaching the string, the ***cut link stops at the Apply MTF tin***.

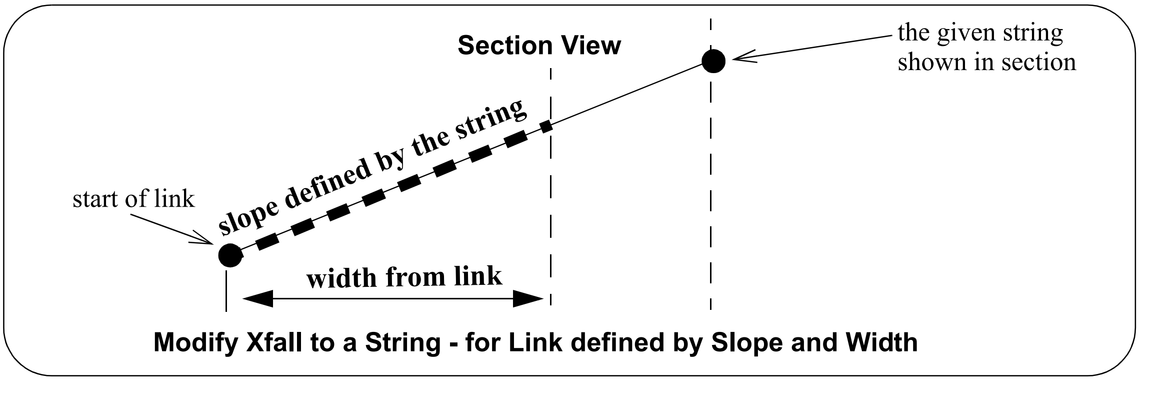


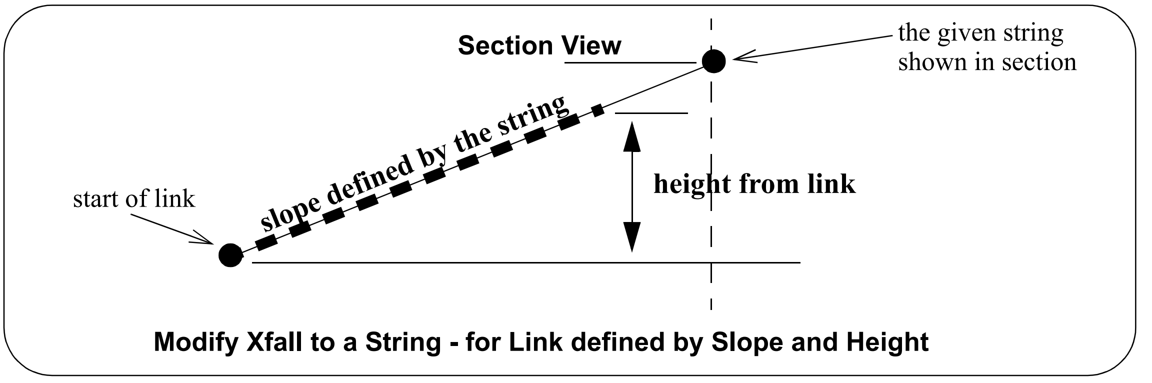


Xfall to String:

For a cut link defined by *slope and width* or *slope and height*, **Cut Slope to string** calculates the **slope** of the link as the *slope* from the start point of the link to the **selected string**. The *width* or *height* is taken from the link. The option will give an error for a link defined by *width* and *height*.

However if the tin in the Apply MTF is cut before reaching the string, the ***cut link stops at the Apply MTF tin***.



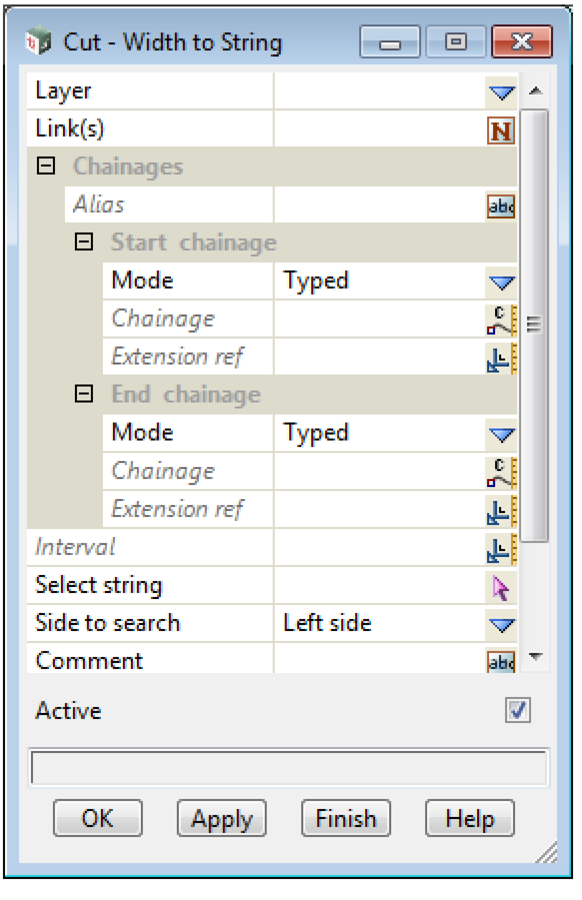


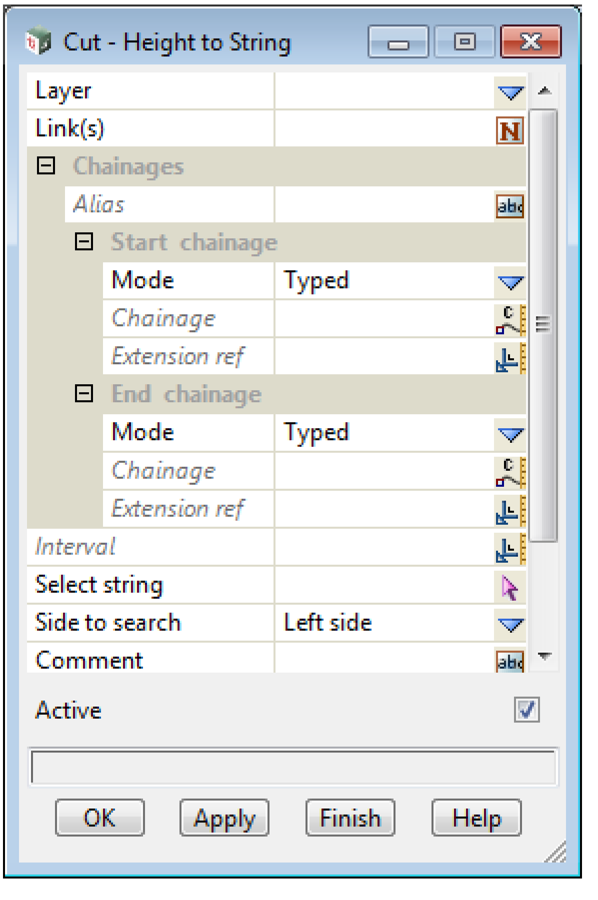
Note:

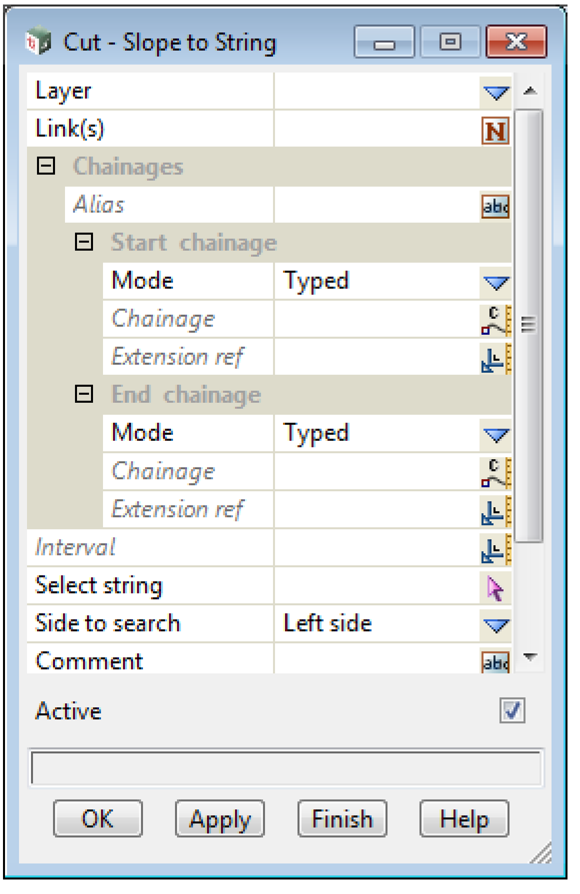
Using two of the above modifiers together and with the same string will place the end point of the link on the selected string. For example for a modifier defined by *width and slope*, using *width to string* and a *slope to string* with the same string will place the end of the link on that string.

But the To string option will do the same thing in one command. See Cut Link - Modify To String.

Selecting **Width to string, Height to string or Slope to string** brings up the **Cut - Width to String**, **Cut - Height to String** and **Cut - Slope to String** panels respectively.







The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu

name of the link to modify.

Start/End mode	choice box	Start (ref)/End (ref)	

defines the start/end chainages for modifying the link.
For more information on Start/End mode, see 21.2.1 MTF Hinge Modifiers ***except for this option****:*

*When the **Start mode** is **Start (ref),** or **Typed** and the chainage is **blank**, the modification begins at the low dropped chainage of the selected string.*

*When the **End mode** is **End (ref),** or **Typed** and the chainage is **blank**, the modification ends at the high dropped chainage of the selected string.*

Interval	input		

*if **non blank**, the interval to use to create cross sections and strings over the given chainage range. If **blank**, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

String	string-select		

select string to use for defining width/height/slope for the link.

Side to search	input	left side	left side, right side, both sides

side of the hinge string to start searching to find the string to define width/height/slope.

Extra start/end	tick box		

*if **ticked**, add an extra x-section 0.1 mm **before** the start/end chainage.*

Comment input

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active tick box tick

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

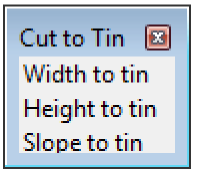
OK/Apply button

OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

Continue to the next section 21.3.0.7 Cut Link - to Tin or return to 21.3.0.1 Insert Cut Links from a Template.

21.3.0.7 Cut Link - to Tin

The *Cut* **to tin** walk-right brings up the **Cut to Tin** menu with options to calculate the width, height or slope to get to a given tin (which doesn't have to be the Apply MTF tin).

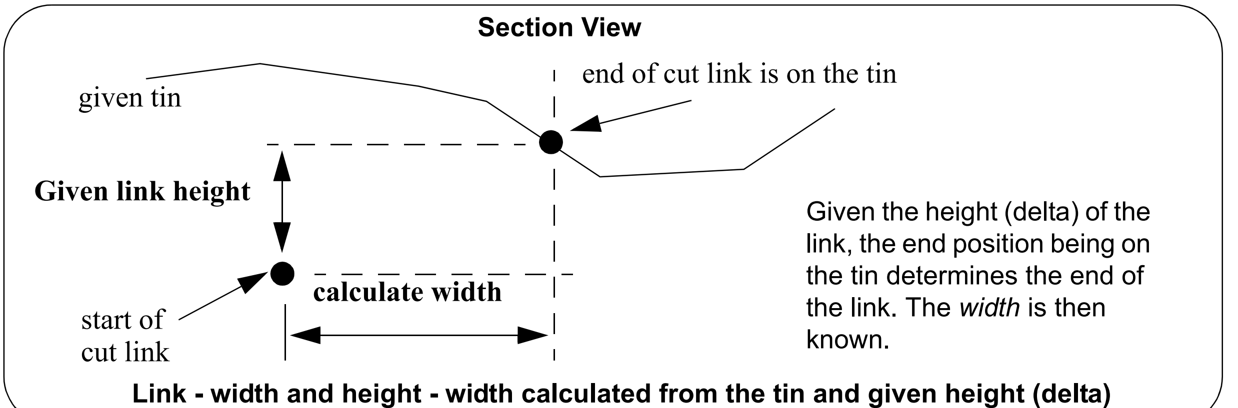


For *Width/Height/Tin to tin*, go to the next section Cut Link - Modify Width, Height or Xfall to Sit on User Tin

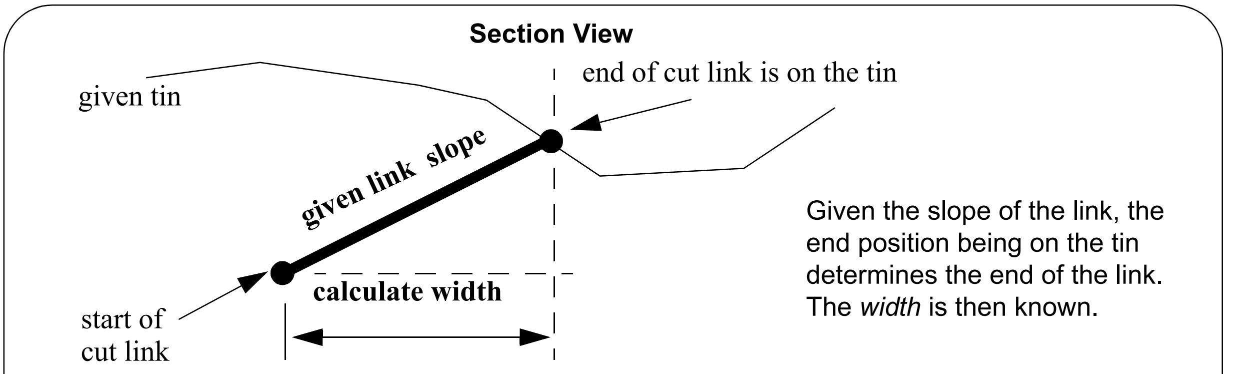
Cut Link - Modify Width, Height or Xfall to Sit on User Tin

Width to Tin

For a cut link defined by *width and height* or *width and slope*, **Width to tin** calculates the **width** of the link as the width required so that the link will sit on the user given tin at the ***height/slope given in the link***. The option gives an error for a link defined by *height* and *slope*.



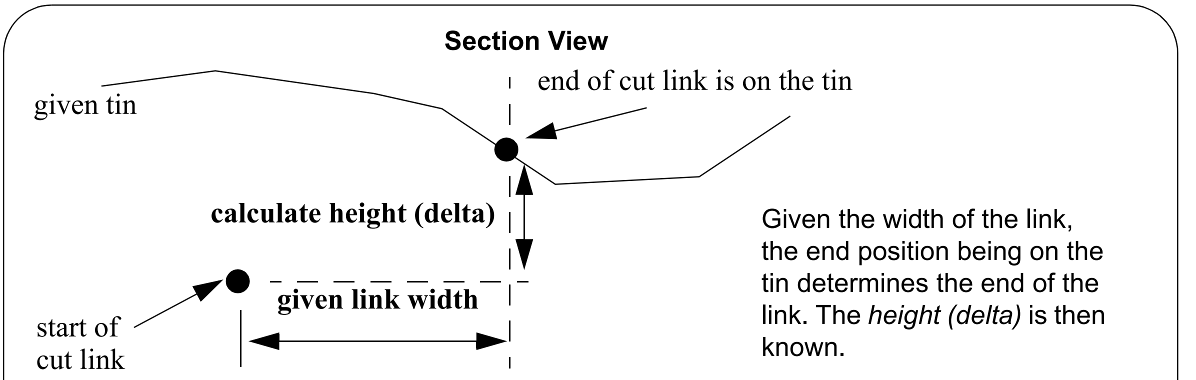
Link - width and height - width calculated from the tin and given height (delta)



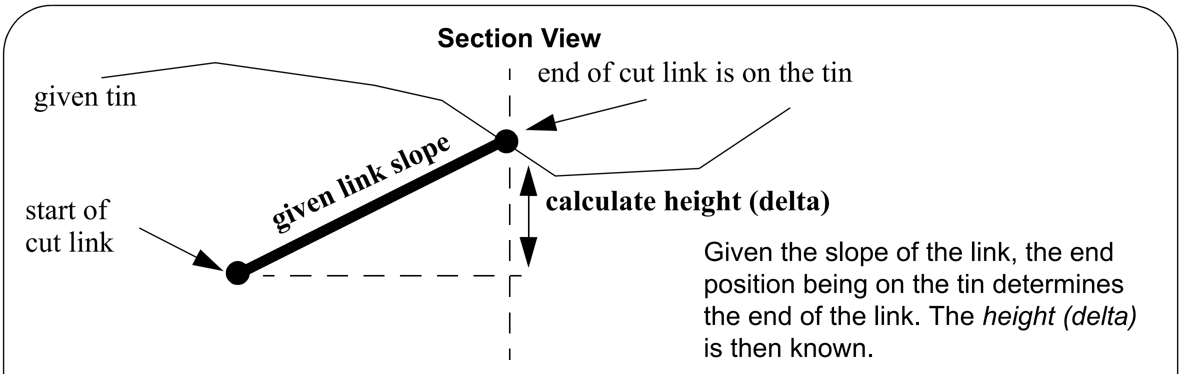
Link - width and slope - width calculated from the tin and given slope

Height to Tin

For a cut link defined by *height and width* or *height and slope*, the **Height to tin** calculates the ***height*** of the link as the difference in the height of the start point of the link, and the height that is required so that the link will sit on the user tin at the ***width/slope given in the link.*** The option gives an error for a link defined by *width and xfall*.



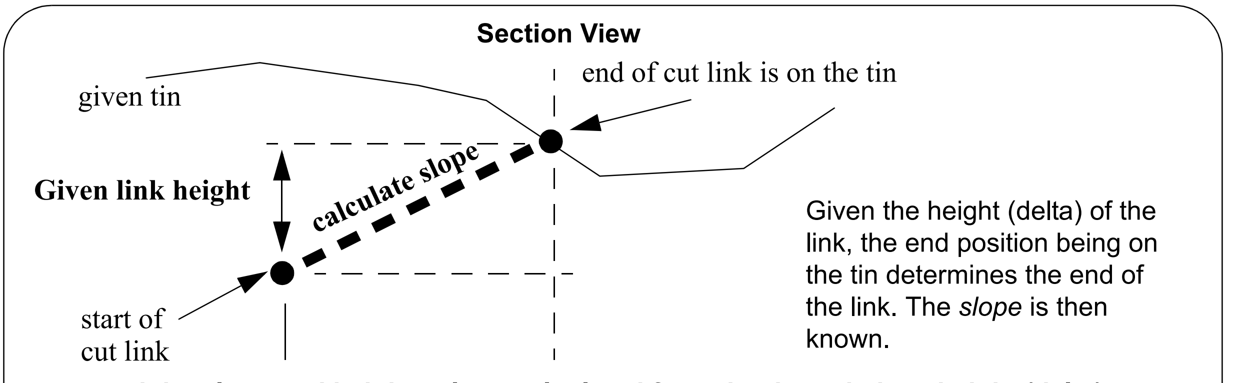
Link - height and width - height (delta) calculated from the tin and given width



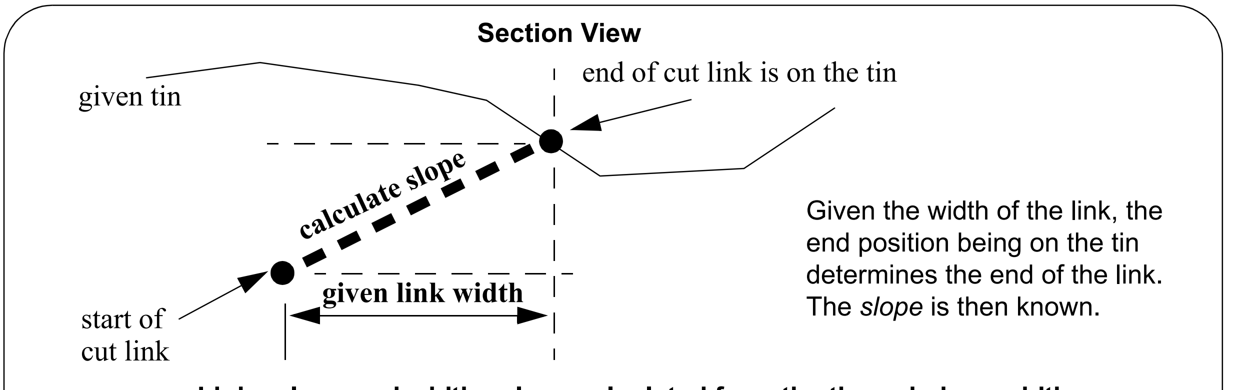
Link - height and slope - height (delta) calculated from the tin and given slope

Slope to Tin

For a cut link defined by *slope and width* or *slope and height*, **Slope to tin** calculates the **slope** of the link as the *slope* required so that the link will sit on the user tin at the ***width/height given in the link***. The option gives an error for a link defined by *width and height*.

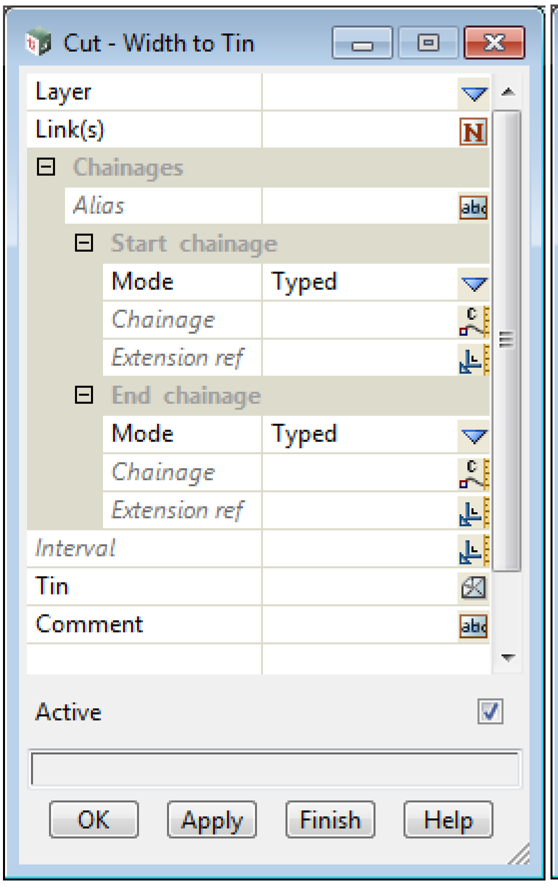


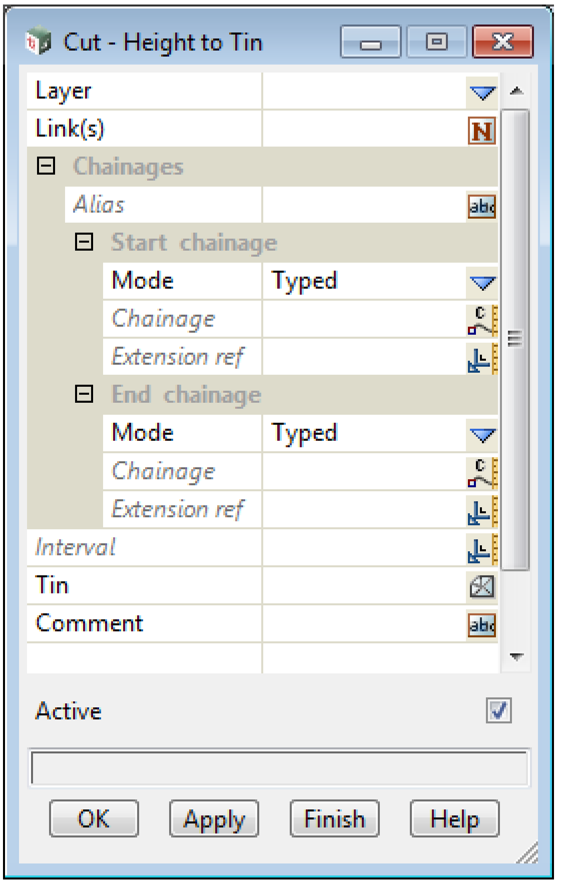
Link - slope and height - slope calculated from the tin and given height (delta)

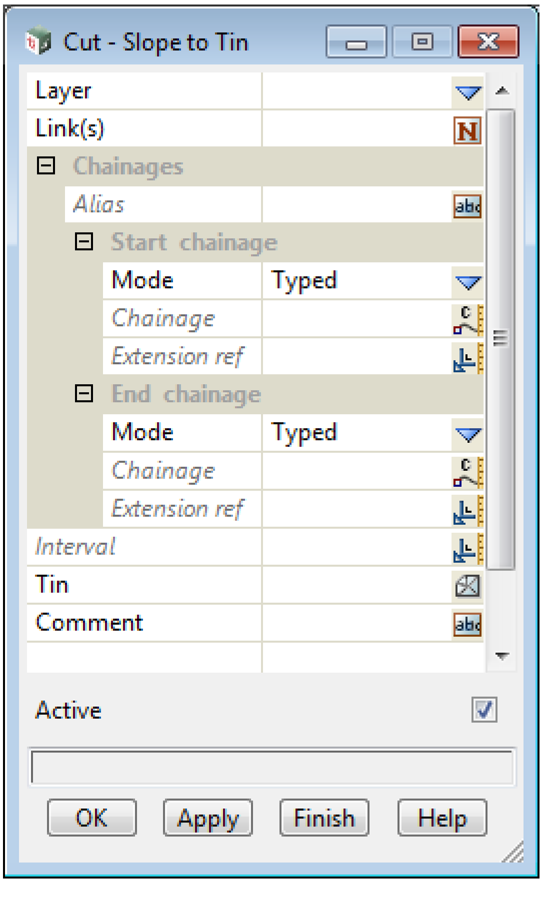


Link - slope and width - slope calculated from the tin and given width

Selecting **Width to tin**, **Height to tin, or Slope to tin** brings up the **Cut - Width to Tin**, **Cut - Height to Tin** and **Cut - Slope to Tin** panels respectively.


Cut - Width to Tin
Layer
Link(s)
Chainages
Alias
Start chainage
Mode
Typed
Chainage
Extension ref
End chainage
Mode
Typed
Chainage
Extension ref
Interval
Tin
Comment
Active
OK
Apply
Finish
Help


Cut - Height to Tin
Layer
Link(s)
Chainages
Alias
Start chainage
Mode
Typed
Chainage
Extension ref
End chainage
Mode
Typed
Chainage
Extension ref
Interval
Tin
Comment
Active
OK
Apply
Finish
Help


Cut - Slope to Tin
Layer
Link(s)
Chainages
Alias
Start chainage
Mode
Typed
Chainage
Extension ref
End chainage
Mode
Typed
Chainage
Extension ref
Interval
Tin
Comment
Active
OK
Apply
Finish
Help

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu

name of the link to modify.

Start/End mode	choice box	Start (ref)/End (ref)	

defines the start/end chainages for modifying the link
For more information on Start/End mode, see 21.2.1 MTF Hinge Modifiers

Interval	input		

*if **non blank**, the interval to use to create cross sections and strings over the given chainage range.*
*If **blank**, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

Tin	input		available tins

the tin to use for defining the width/height/slope

Extra start/end	tick box		

*if **ticked**, add an extra x-section 0.1 mm **before** the start/end chainage.*

Comment	input		

comment to add to the end of the line. In the file, the comment will be preceded by **//**.

Active	tick box	tick	

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

OK/Apply	button		

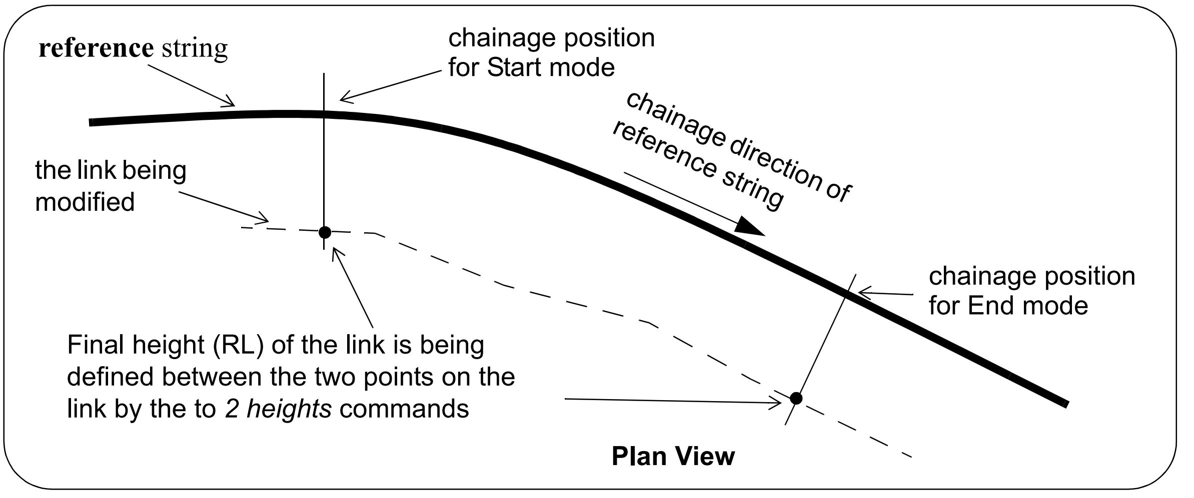
OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

Continue to the next section 21.3.0.8 Cut Link - to Two Heights or return to 21.3.0.1 Insert Cut Links from a Template.

21.3.0.8 Cut Link - to Two Heights

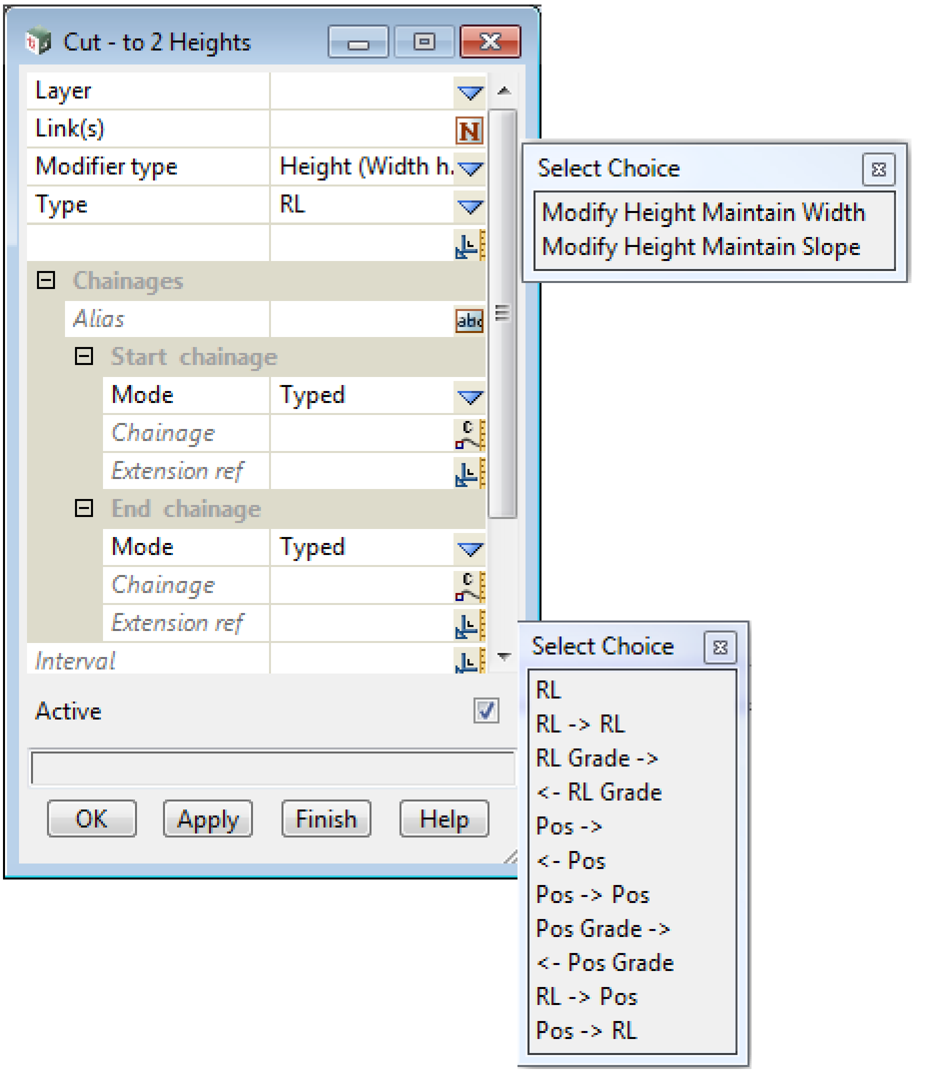
For an existing link, the **to 2 heights** option has a number of methods for **defining the height** from the start mode chainage to the end mode chainage. For example, over the chainage range, the height can be interpolated between two given RL's.

However if the *Apply MTF* tin is cut before reaching the end of the modified link, the ***cut link stops at the tin***.



Note: the ***2 Strings*** modifiers work with two points in a **section**. See 21.3.0.9 Cut Link - to 2 Strings.

Selecting **o 2 heights** brings up the **Cut - to 2 Heights** panel

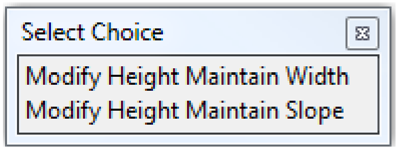


The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu

name of the link to modify.

Modifier type	choice box		



*The **Maintain** Slope/Width is the part of the new definition of the selected link that is taken from the selected* **Link name**. *See* Cut - Maintaining Width or Slope.

*The **Height** for the selected link is calculated by the method given by **Type**.*

Start/End mode	choice box	Start (ref)/End (ref)	

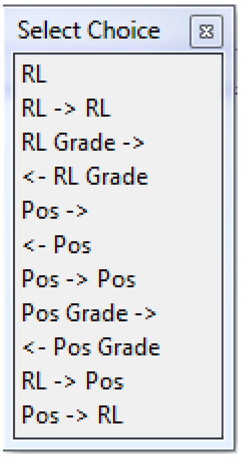
defines the start/end chainages for modifying the link.
For more information on Start/End mode, see 21.2.1 MTF Hinge Modifiers.

Interval	input		

*if **non blank**, the interval to use to create cross sections and strings over the given chainage range.*

If ***blank****, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

Type choice box



For the calculation of height for each type, go to Cut - Calculating the Heights for each Type

Extra start/end tick box

if ***ticked****, add an extra x-section 0.1 mm* ***before*** *the start/end chainage.*

Comment input

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active tick box tick

if ***ticked****, use this modifier.*
if ***not ticked****, don't use this modifier.*

OK/Apply button

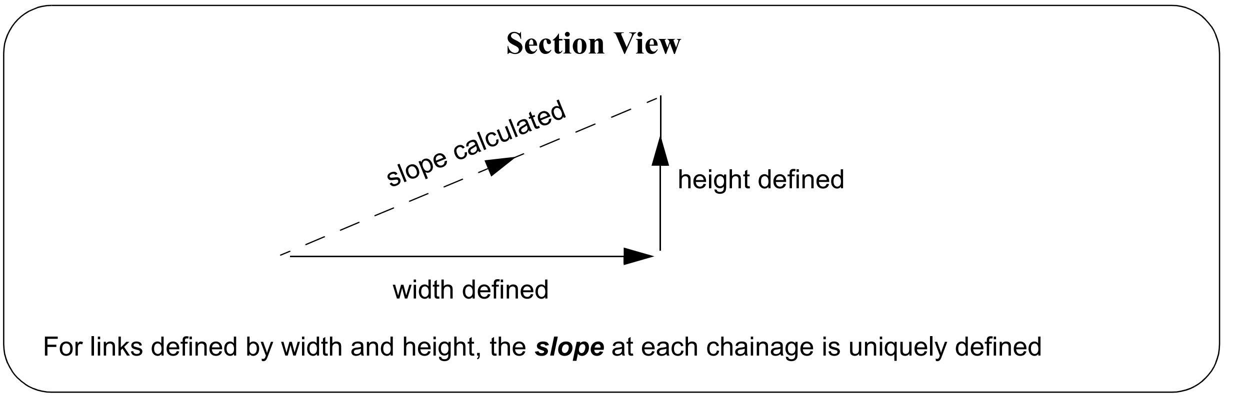
OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

Cut - Maintaining Width or Slope

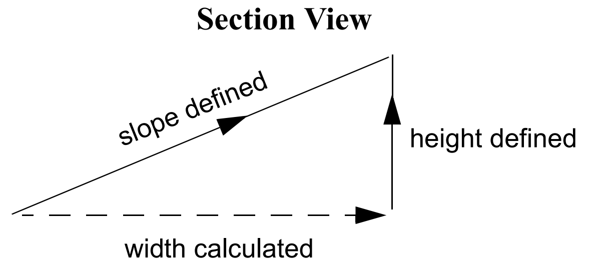
The option ***to 2 heights*** defines the ***height*** of the selected string so to completely define the link, only the width *or* the slope is needed.

So from the selected link only the *width* or the xfall is used (maintained) - the height is calculated by the **to 2 heights** option.

If the selected link is defined by ***width and height***, then at any chainage a ***slope*** can be calculated from the width and height at that chainage.



If the selected link is defined by ***slope and height***, then at any chainage a ***width*** can be calculated from the slope and height at that chainage.

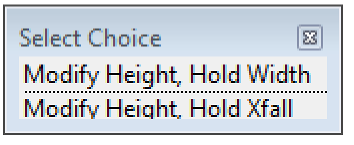


For links defined by slope and height, the ***width*** at each chainage is uniquely defined

Hence it doesn't matter if the selected link is defined by *width and slope*, *width and height*, or *slope and height*, at each chainage a *width and slope* are uniquely defined.

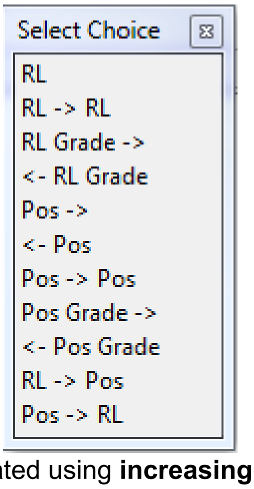
So no matter how the selected link is defined, at any chainage, a unique width or a slope is known and it is that **Width** or **Slope** can be maintained by the ***to 2 heights*** option.

So the choices for the option are:



Continue to the next section 21.3.0.9 Cut Link - to 2 Strings or return to 21.3.0.8 Cut Link - to Two Heights, 21.3.0.1 Insert Cut Links from a Template.

Cut - Calculating the Heights for each Type



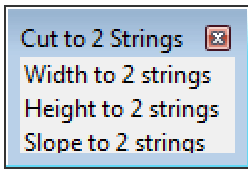
Important Note: Grade is calculated using **increasing Alignment chainages** and the **heights at the points** along the string being modifies. When going around a curve this will not be exactly the same as the grade along the string being modified.

The definitions of the calculations for each choice are the same as for the Fixed links. See 21.2.2.2.13.1 Fixed - Calculating the Heights for each Type.

Continue to the next section 21.3.0.9 Cut Link - to 2 Strings or return to 21.3.0.8 Cut Link - to Two Heights, 21.3.0.1 Insert Cut Links from a Template.

21.3.0.9 Cut Link - to 2 Strings

The *Cut* **to 2 strings** walk-right brings up the **Cut to 2 Strings** menu with options to calculate the width, height or slope from two given strings.



For *Width/Height/Slope to 2 Strings*, go to the next section Cut Link - Width, Height or Slope Between Two Strings

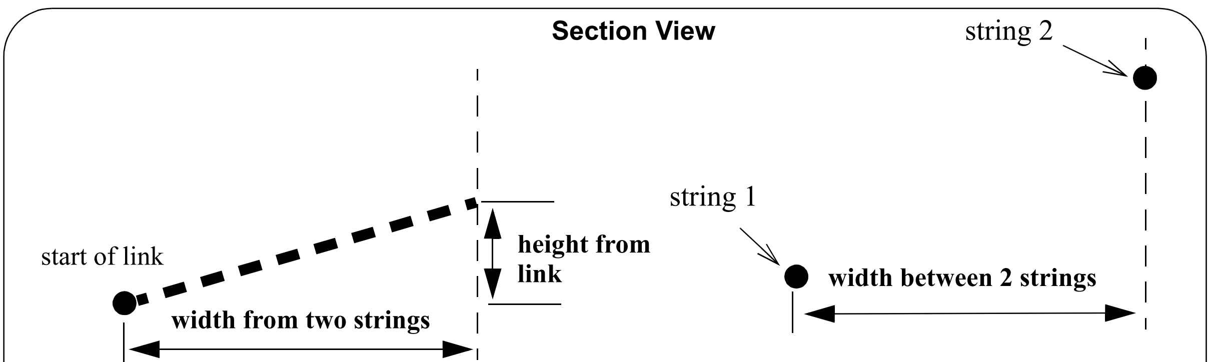
Cut Link - Width, Height or Slope Between Two Strings

Width Between Two Strings

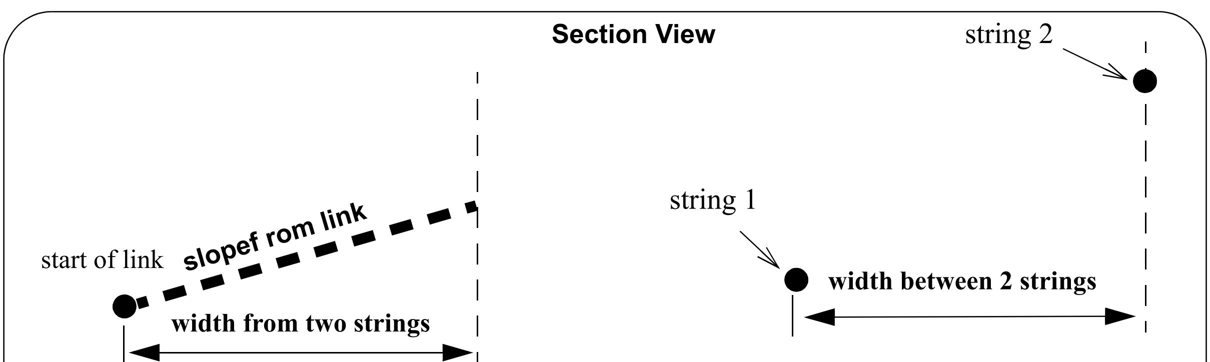
For a cut link defined by *width* and *height* or *width* and *slope*, **Width to 2 strings** sets the **width** for a link to be the **width between two** existing ***12d Model*** strings,

However if the Apply MTF tin is cut before reaching the string, the ***cut link stops at the tin***.

The option will give an error for a link defined by *height* and *slope*.



Modify Width to be that between two Strings - for Link defined by Width and Height



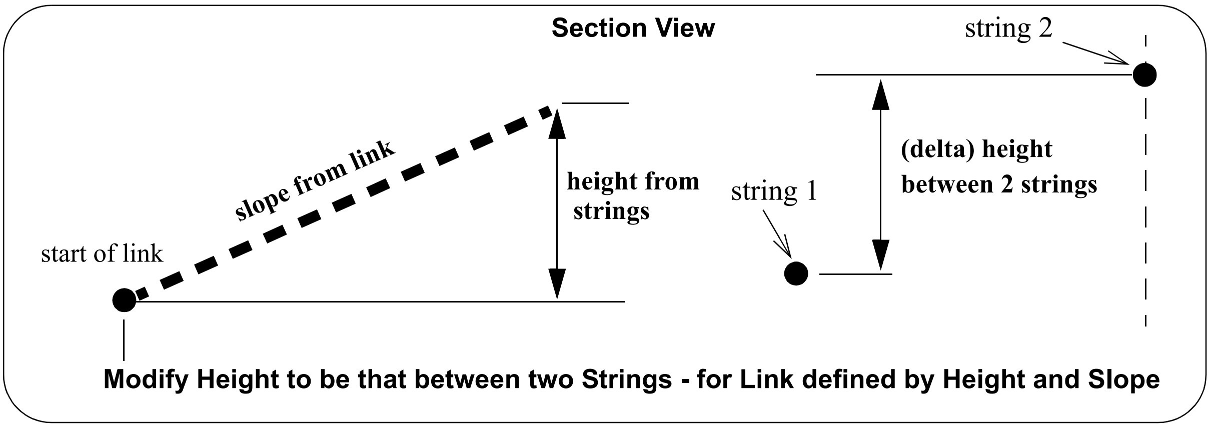
Modify Width to be that between two Strings - for Link defined by Width and Slope

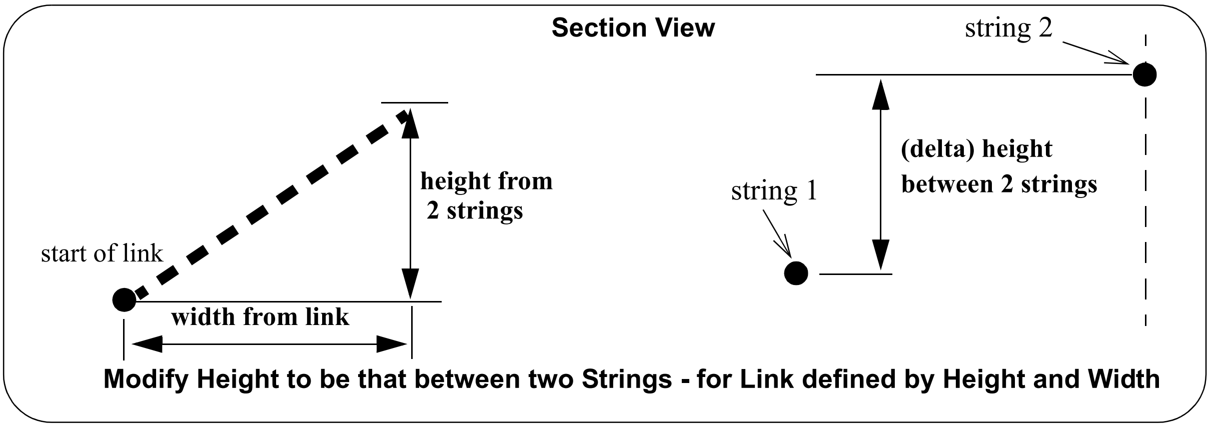
Height Between Two Strings

For a cut link defined by *height* and *width* or *height* and *slope*, **Height to 2 strings** sets the **height** for the link to be the **height between two** existing ***12d Model*** strings,

However if the Apply MTF tin is cut before reaching the string, the ***cut link stops at the tin***.

The option will give an error for a link defined by *width* and *slope*.



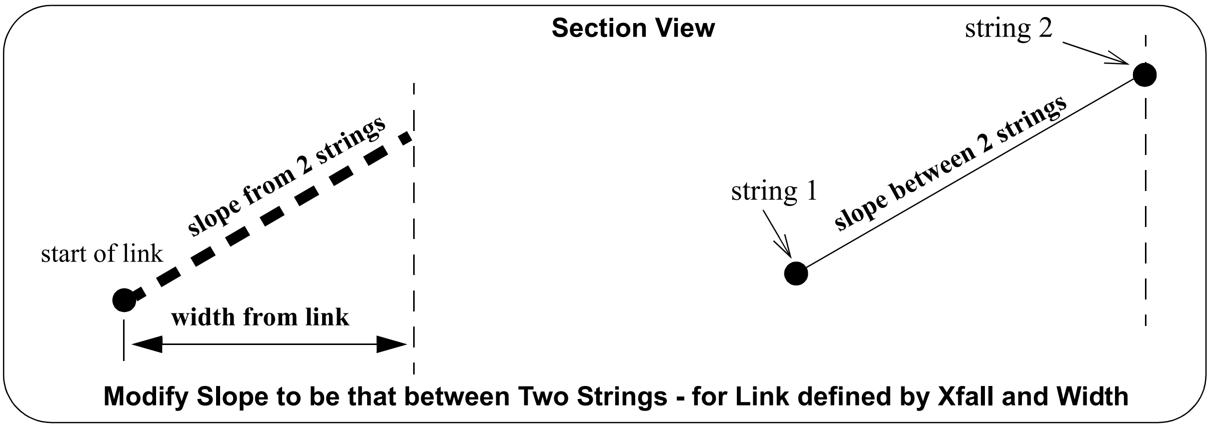


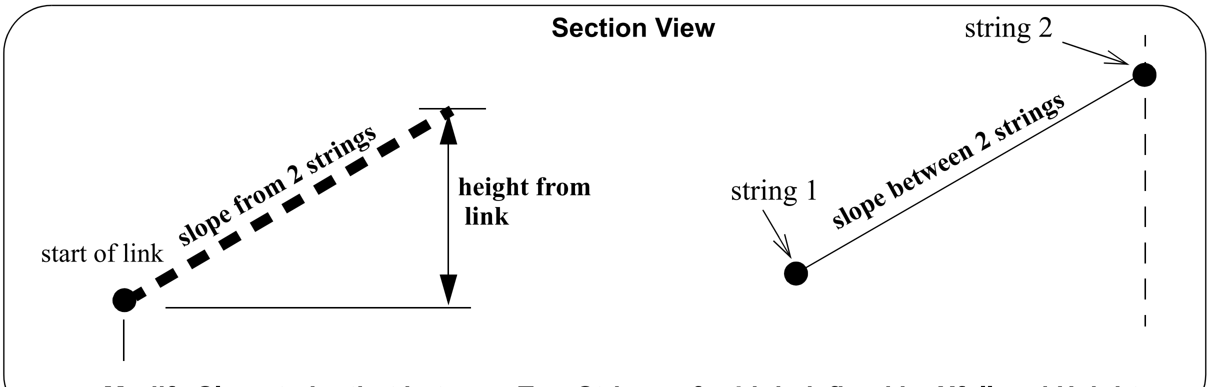
Slope Between Two Strings

For a cut link defined by *slope* and *width* or *slope* and *height*, **Slope to 2 strings** sets the **slope** for the link to be the **slope between two** existing ***12d Model*** strings,

However if the *Apply MTF* tin is cut before reaching the string, the ***cut link stops at the tin***.

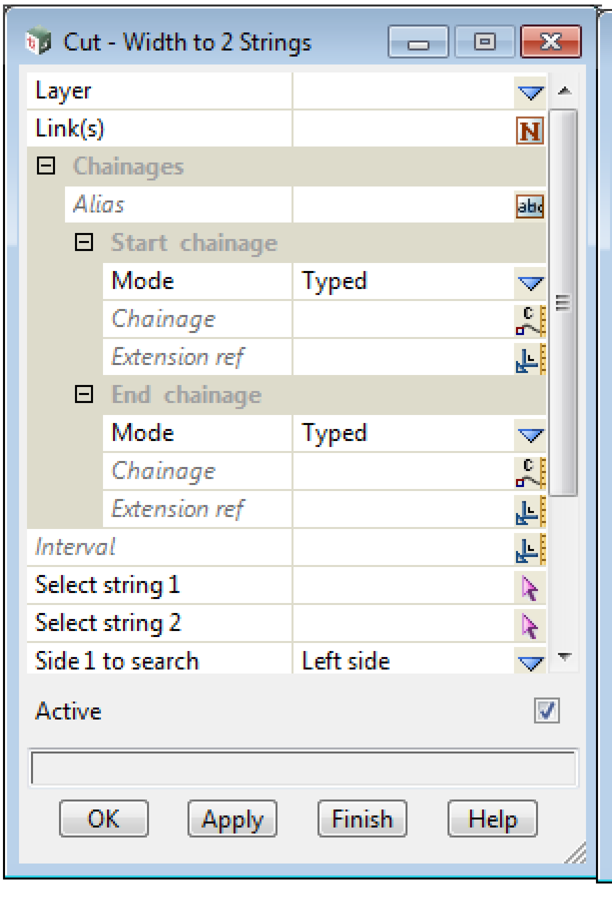
The option will give an error for a link defined by *width* and *height*.

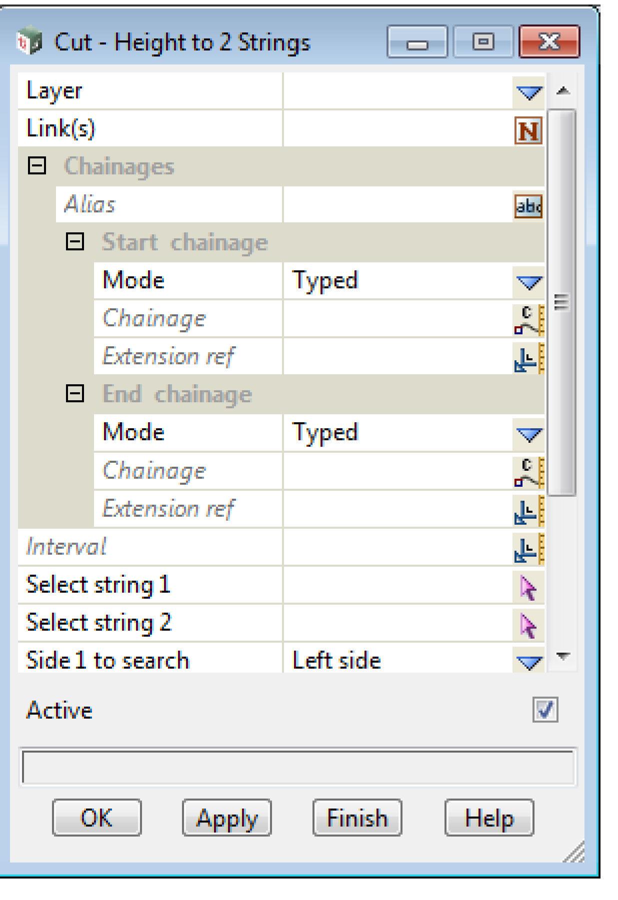


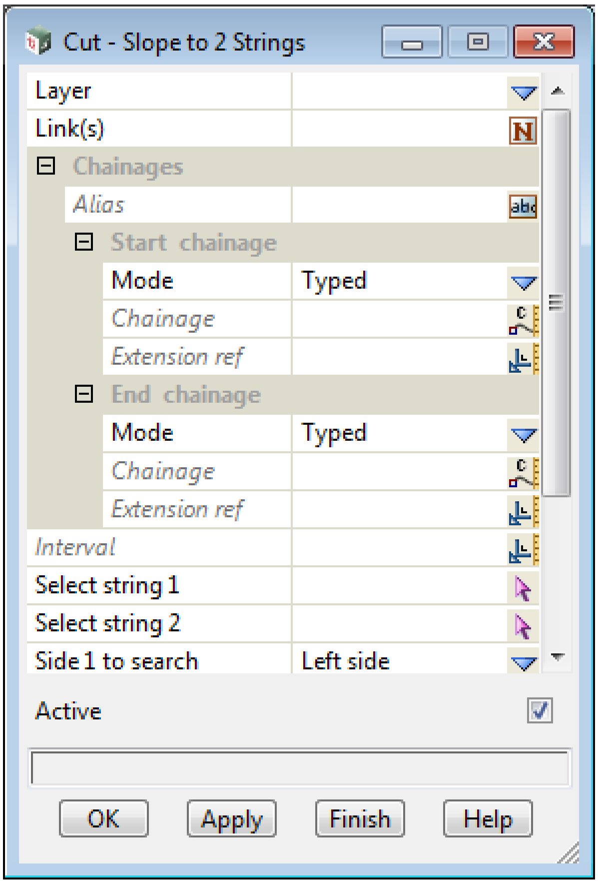


Modify Slope to be that between Two Strings - for Link defined by Xfall and Height

Selecting the **width to 2 strings, height to 2 strings or slope to 2 strings** option brings up the **Cut - Width to 2 Strings**, **Cut - Height to 2 Strings** and **Cut - Slope to 2 String** panels respectively.







The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu

name of the link to modify.

Start/End mode	choice box	Start (ref)/End (ref)	

defines the start/end chainages for modifying the link
For more information on Start/End mode, see 21.2.1 MTF Hinge Modifiers

Interval input

*if **non blank**, the interval to use to create cross sections and strings over the given chainage range.*
*If **blank**, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

String 1 string-select

select the first string to use for defining width/height/slope for the link.

String 2 string-select

select the second string to use for defining width/height/slope for the link.

Side 1 to search	input	left side	left side, right side, both sides

side of the hinge string to start searching to find string 1 to use in defining width/height/slope.

Side 2 to search	input	left side	left side, right side, both sides

side of the hinge string to start searching to find string 2 to use in defining width/height/slope.

Extra start/end tick box

*if **ticked**, add an extra x-section 0.1 mm **before** the start/end chainage.*

Comment input

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active tick box tick

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

OK/Apply button

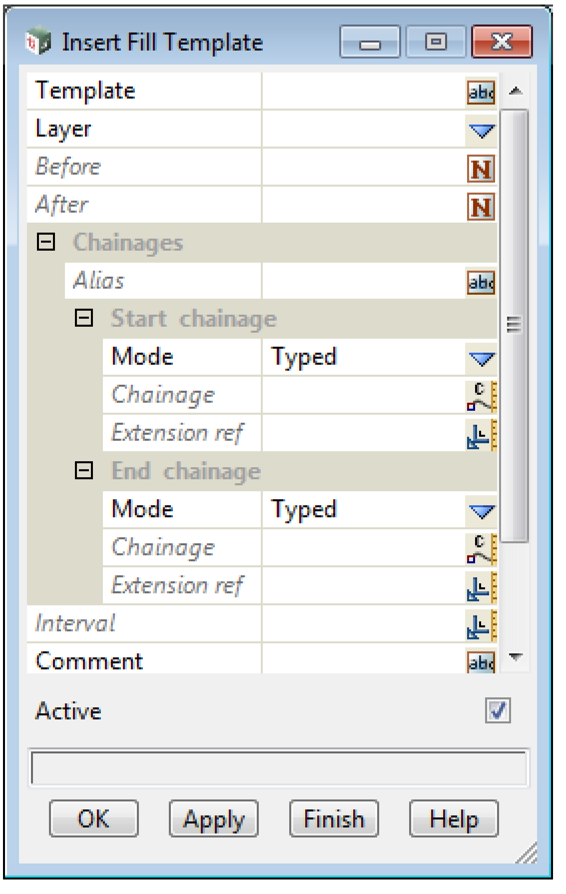
OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

Continue to the next section 21.3.0.10 Insert Fill Links from a Template or return to 21.3.0.1 Insert Cut Links from a Template.

21.3.0.10 Insert Fill Links from a Template

All the fill links from an existing template can be inserted by the **Insert Template** command.

Selecting **Insert fill template** brings up the **Insert Fill Template** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Template	template box		select template menu

the name of the template to take the fill links from.

Before	input		select name menu

if ***non-blank****, the fill links from the selected template are inserted before the given link.*
If ***blank****, the fill links from the selected template are appended to the end of the existing fill links.*

Start/End mode	choice box	Start (ref)/End (ref)	

defines the start/end chainages for inserting the new fill links.
For more information on Start/End mode, see 21.2.1 MTF Hinge Modifiers

Interval	input		

if ***non blank****, the interval to use to create cross sections and strings over the given chainage range.*
If ***blank****, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

Extra start/end	tick box		

if ***ticked****, add an extra x-section 0.1 mm* ***before*** *the start/end chainage.*

Comment	input		

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active tick box tick

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

OK/Apply button

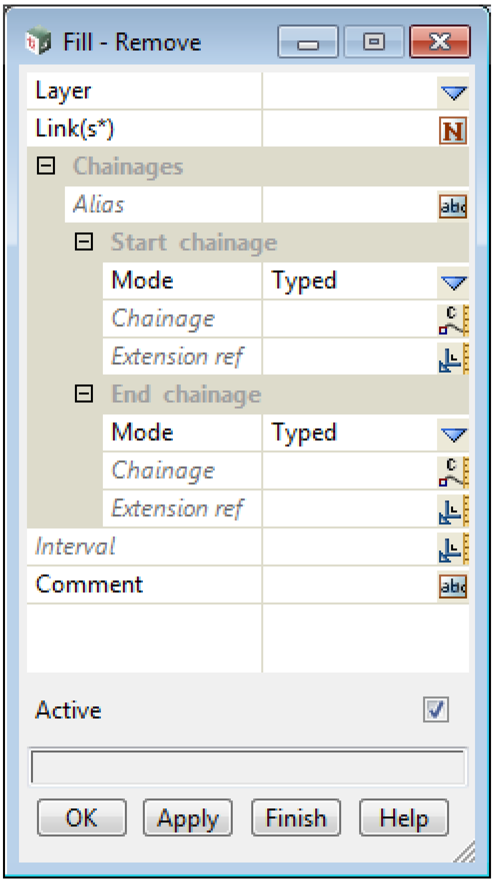
OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

Continue to the next section 21.3.0.11 Fill Link - Remove or return to 21.3.0.10 Insert Fill Links from a Template.

21.3.0.11 Fill Link - Remove

Fill links can be deleted between given chainages by using the **fill remove** modifier.

Selecting **Remove** brings up the **Fill - Remove** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu

name of the link to removed.

Start/End mode	choice box	Start (ref)/End (ref)	

defines the start/end chainages for removing the link
For more information on Start/End mode, see 21.2.1 MTF Hinge Modifiers

Interval	input		

if ***non blank****, the interval to use to create cross sections and strings over the given chainage range.*
If ***blank****, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

Extra start/end	tick box		

if ***ticked****, add an extra x-section 0.1 mm* ***before*** *the start/end chainage.*

Comment	input		

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active	tick box	tick	

if ***ticked****, use this modifier.*
if ***not ticked****, don't use this modifier.*

OK/Apply button

__OK__ stores the values in the fields and removes the panel.
__Apply__ stores the values and leaves the panel on the screen.

Continue to the next section 21.3.0.12 Fill Link - Modify Width or Height or return to 21.3.0.10 Insert Fill Links from a Template.

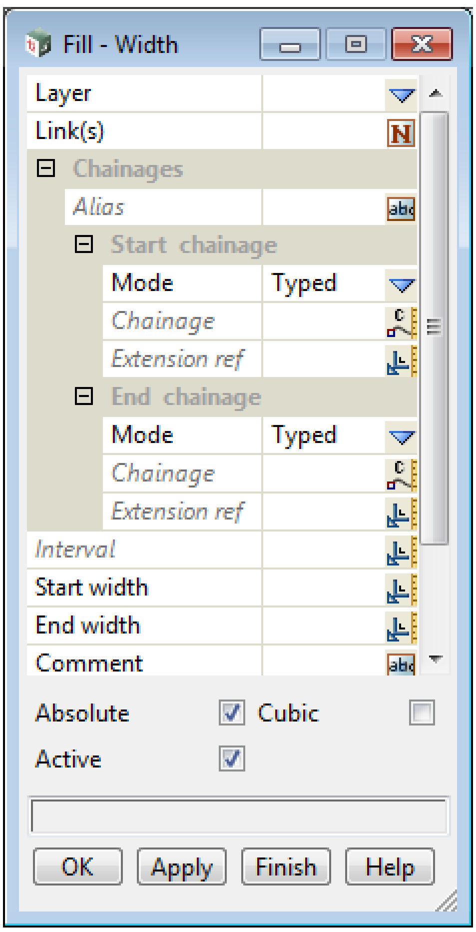
21.3.0.12 Fill Link - Modify Width or Height

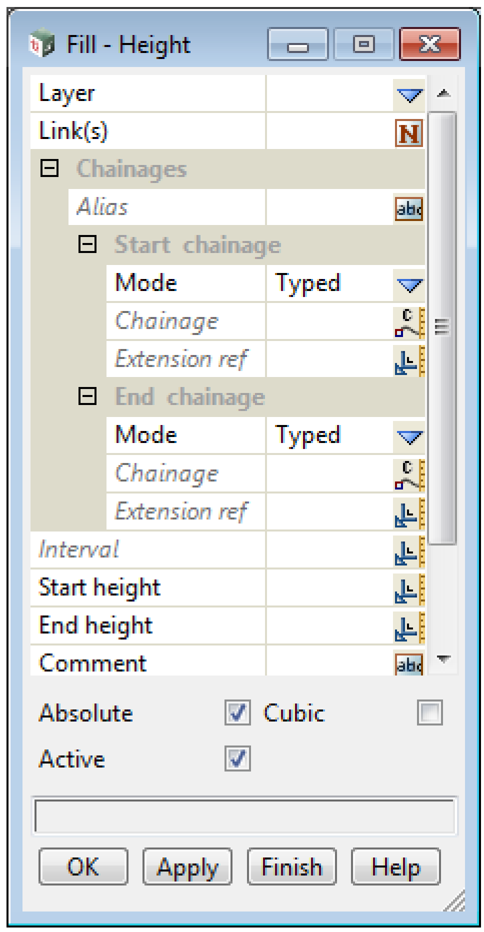
The **fill width** modifier is used to modify the ***width*** of variable fill links originally defined by width. That is, links defined by *width and slope* or *width and height*.

Similarly the **fill height** modifier is used to modify the ***height*** of variable fill links originally defined by height. That is, links defined by *height and slope* or *height and width*.

Important Note - for a Fill link, a **positive** height is **down**.

Selecting the **width and height** option brings up the **Fill - Width** and **Fill - Height** panels respectively.





The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu

name of the link to modify.

Start/End mode	choice box	Start (ref)/End (ref)	

defines the start/end chainages for modifying the link
For more information on Start/End mode, see 21.2.1 MTF Hinge Modifiers

Interval input

*if **non blank**, the interval to use to create cross sections and strings over the given chainage range.*
*If **blank**, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

Start/End width/height input measures menu

start/end width/height for modifying the template link.

***Important Note** - for a fill link, height **positive** is down.*

Extra start/end tick box

*if **ticked**, add an extra x-section 0.1 mm **before** the start/end chainage.*

Absolute tick box tick

*if **ticked**, the width/height is set to the values given in the* **start** *and* **end width/height** *fields.*
*if **not ticked**, the values given in the* **start** *and* **end height/width** *fields are added to the existing widths/heights.*

Cubic tick box

*if **ticked**, the width/height is varied as a reverse cubic between the start and end chainages.*
*if **not ticked**, the width/height is varied linearly between the start and end chainages.*

Comment input

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active tick box tick

*if **ticked**, use this modifier.*

*if **not ticked**, don't use this modifier.*

OK/Apply button

OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

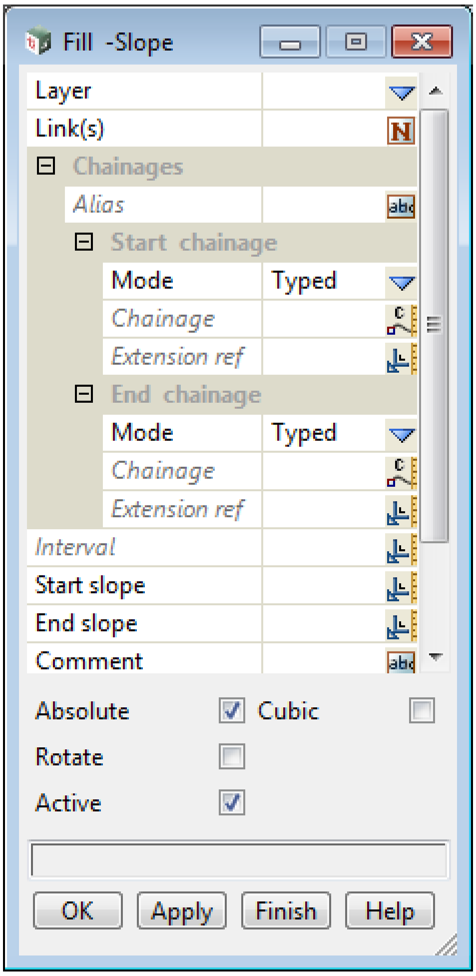
Continue to the next section 21.3.0.13 Fill Link - Modify Slope or return to 21.3.0.10 Insert Fill Links from a Template.

21.3.0.13 Fill Link - Modify Slope

The **fill slope** modifier is used to modify the ***slope*** of variable fill links originally defined by slope. That is, links defined by *slope and width* or *slope and height*.

Important Note - for a Fill link, a **positive** fill slope is **down**.

Selecting **slope** brings up the **Fill - Slope** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu

name of the link to modify.

Start/End mode	choice box	Start (ref)/End (ref)	

defines the start/end chainages for modifying the link
For more information on Start/End mode, see 21.2.1 MTF Hinge Modifiers

Interval	input		

if ***non blank****, the interval to use to create cross sections and strings over the given chainage range.*
If ***blank****, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

Start/End slope	input		measures menu

start/end slope for modifying the template link.

Important Note *- for a fill link, a* ***positive*** *slope is down.*

Extra start/end tick box

*if **ticked**, add an extra x-section 0.1 mm **before** the start/end chainage.*

Absolute tick box tick

*if **ticked**, the slope is set to the values given in the* **start** *and* **end slope** *fields.*
*if **not ticked**, the values given in the* **start** *and* **end slope** *fields are added to the existing slopes.*

Cubic and Rotate tick boxes:

*Only **none** or **one** of **Cubic** and **Rotate** can be set to **tick.***
*The **default** is **none** - that is, **neither is ticked** and in the default case, the slope is varied **linearly with respect to slope** between the start and end chainages.*

Cubic tick box

*if **ticked**, the slope is varied as a reverse cubic between the start and end chainages.*

Rotate tick box

*if **ticked**, the slope is varied linearly **with respect to** the **angle,** between the start and end chainages.*

Comment input

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active tick box tick

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

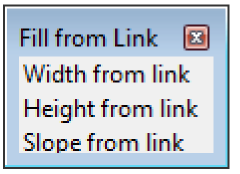
OK/Apply button

OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

Continue to the next section 21.3.0.14 Fill Link - from Link or return to 21.3.0.10 Insert Fill Links from a Template.

21.3.0.14 Fill Link - from Link

The *Fill* **from link** walk-right brings up the **Fill from Link** menu with options to take the width, height or slope from another link.



For *Width/Height/Slope from link*, go to the next section Fill Link - Take Width, Height or Slope from another Link

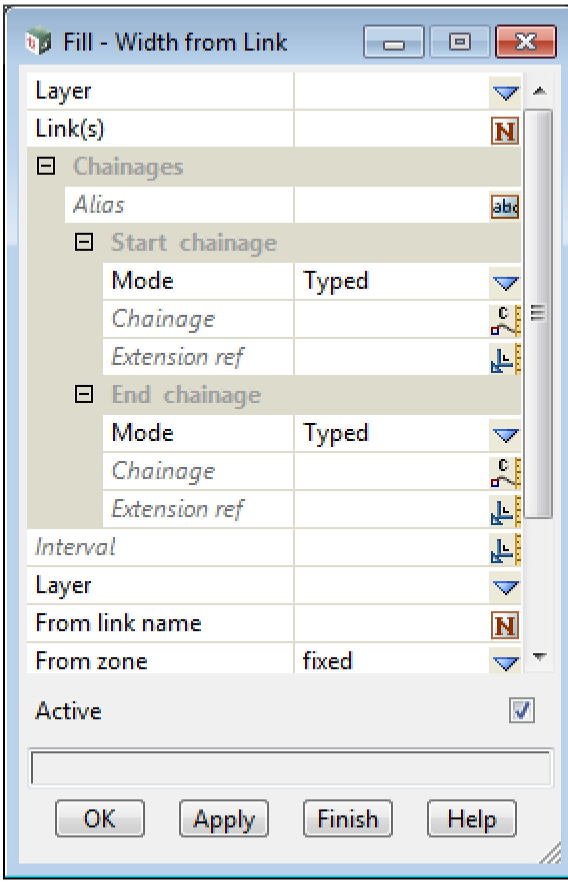
Fill Link - Take Width, Height or Slope from another Link

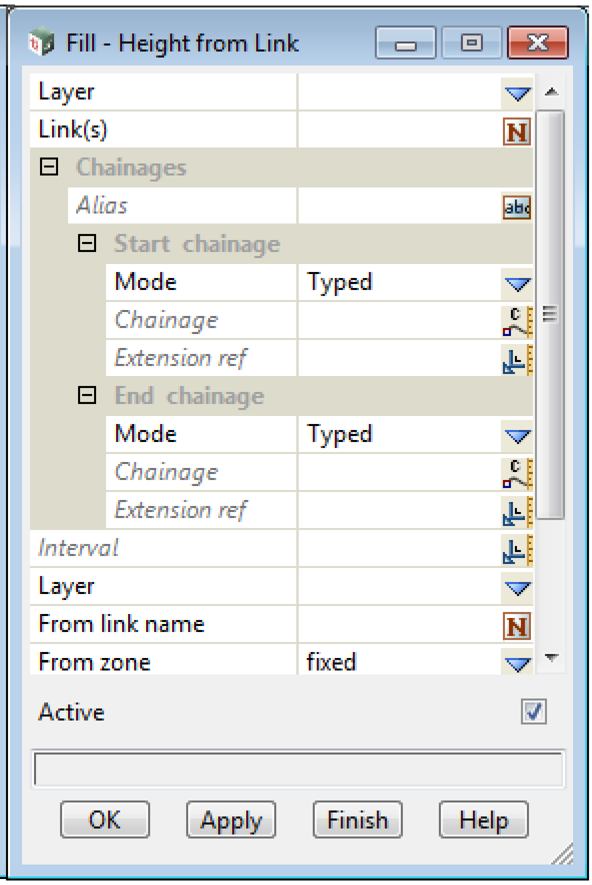
The **Width from link** modifier is used to modify the ***width*** of fill links originally defined by width to be the same width as another link (*width and height* or *width and slope*). That is, the width of the link is a copy of the width of another link.

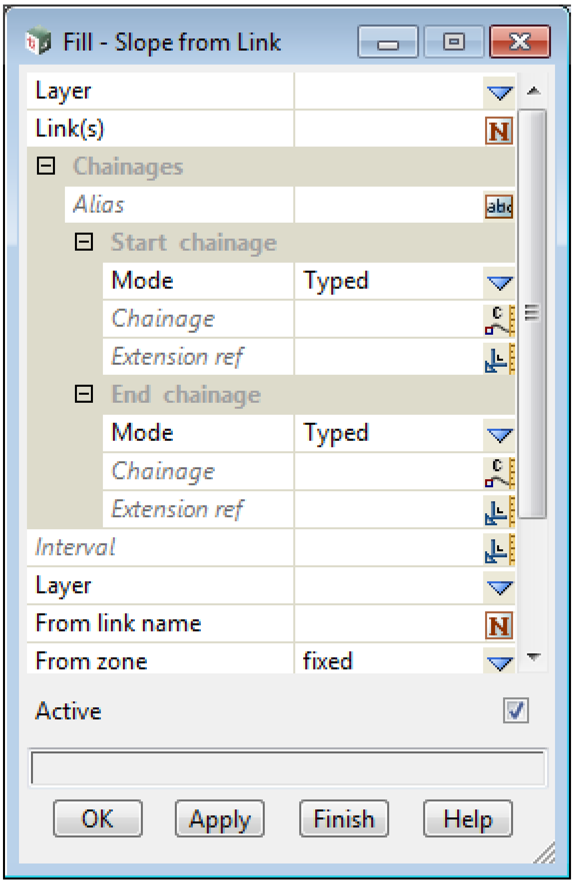
The **Height from link** modifier is used to modify the ***height*** of fill links originally defined by height to be the same height as another link (*height and width* or *height and slope*). That is, the height of the link is a copy of the height of another link.

The **Slope from link** modifier is used to modify the slope of fill links originally defined by slope to be the same slope as another link (*slope and width* or *slope and height*). That is, the slope of the link is a copy of the slope of another link. The link to copy slope from can be defined in terms of slope or cross fall. If the link to copy is defined by cross fall, then the slope is calculated to match the cross fall.

Selecting the **width from link, height from link or slope from link** option brings up the **Fill - Width from Link**, **Fill - Height from Link** and **Fill - Slope from Link** panels respectively.


Fill - Width from Link
Layer
Link(s)
Chainages
Alias
Start chainage
Mode
Typed
Chainage
Extension ref
End chainage
Mode
Typed
Chainage
Extension ref
Interval
Layer
From link name
From zone
fixed
Active
OK
Apply
Finish
Help


Fill - Height from Link
Layer
Link(s)
Chainages
Alias
Start chainage
Mode
Typed
Chainage
Extension ref
End chainage
Mode
Typed
Chainage
Extension ref
Interval
Layer
From link name
From zone
fixed
Active
OK
Apply
Finish
Help



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu
name of the link to modify.			
Start/End mode	choice box	Start (ref)/End (ref)	
defines the start/end chainages for modifying the link *For more information on Start/End mode, see* 21.2.1 MTF Hinge Modifiers			
Interval	input		
*if **non blank**, the interval to use to create cross sections and strings over the given chainage range.* *If **blank**, the* **Section separation** *value from the* **Apply MTF** *panel is used.*			
From link name	input		select name menu
template link to take width/height/slope from.			
From zone	input	fixed	fixed, cut, fill
zone that the template link to take width/height/slope from, comes from.			
Extra start/end	tick box		
*if **ticked**, add an extra x-section 0.1 mm **before** the start/end chainage.*			
Comment	input		
comment to add to the end of the line. In the file, the comment will be preceded by //.			
Active	tick box	tick	
*if **ticked**, use this modifier.*			

*if **not ticked**, don't use this modifier.*

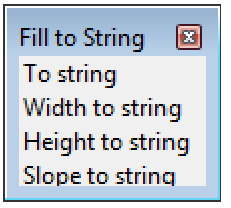
OK/Apply button

OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

Continue to the next section 21.3.0.15 Fill Link - to String or return to 21.3.0.10 Insert Fill Links from a Template.

21.3.0.15 Fill Link - to String

The *Fill* **to string** walk-right brings up the **Fill to String** menu with options to take the width, height or slope by going to another string.



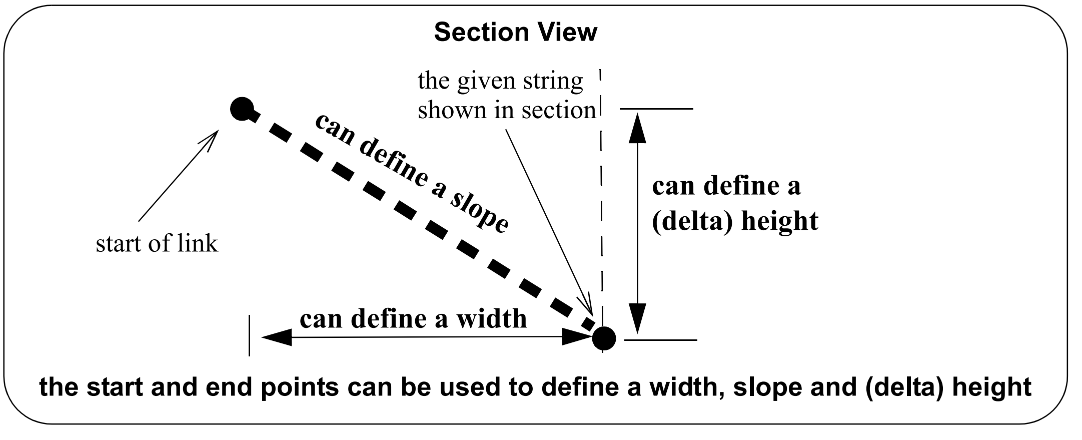
For *To string, go to* Fill Link - Modify To String

Width/Height/Slope to string Fill Link - Calculate Width, Height or Slope to a String

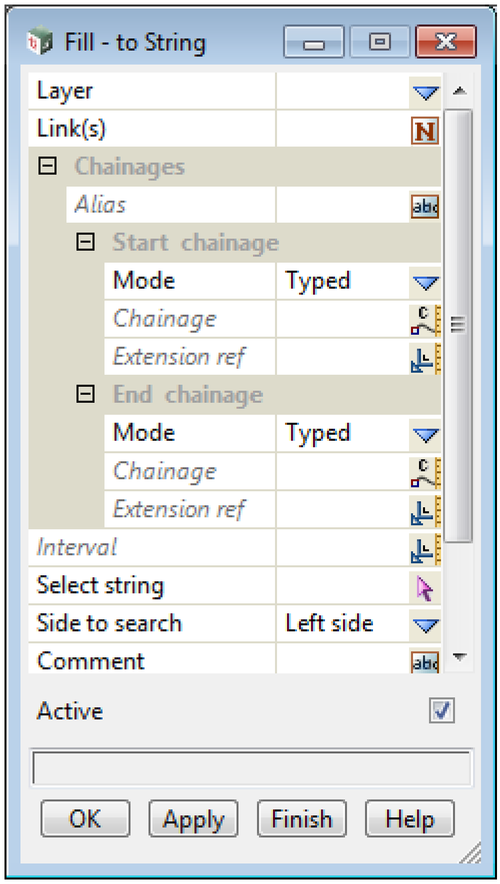
Fill Link - Modify To String

For any fill link, **To string** calculates the width, height and/or slope of the link needed to get from the start point of the link to the **to the selected string**.

However if the tin in the Apply MTF is cut before reaching the string, the ***cut link stops at the Apply MTF tin***.



Selecting **To string** brings up the **Fill - to String** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu

name of the link to modify.

Start/End mode	choice box	Start (ref)/End (ref)	

defines the start/end chainages for modifying the link
For more information on Start/End mode, see 21.2.1 MTF Hinge Modifiers ***except for this option****:*

*When the **Start mode** is **Start (ref),** or **Typed** and the chainage is **blank**, the modification begins at the low dropped chainage of the selected string.*

*When the **End mode** is **End (ref),** or **Typed** and the chainage is **blank**, the modification ends at the high dropped chainage of the selected string.*

Interval	input		

*if **non blank**, the interval to use to create cross sections and strings over the given chainage range. If **blank**, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

String	string-select		

select string to use for defining width/height/crossfall for the link.

Side to search	input	left side	left side, right side, both sides

side of the hinge string to start searching to find the string to define width/height/crossfall.

Extra start/end	tick box		

*if **ticked**, add an extra x-section 0.1 mm **before** the start/end chainage.*

Comment input

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active tick box tick

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

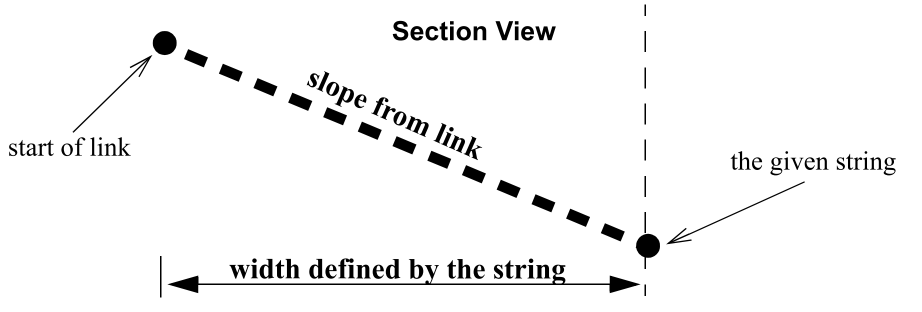
OK/Apply button

OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

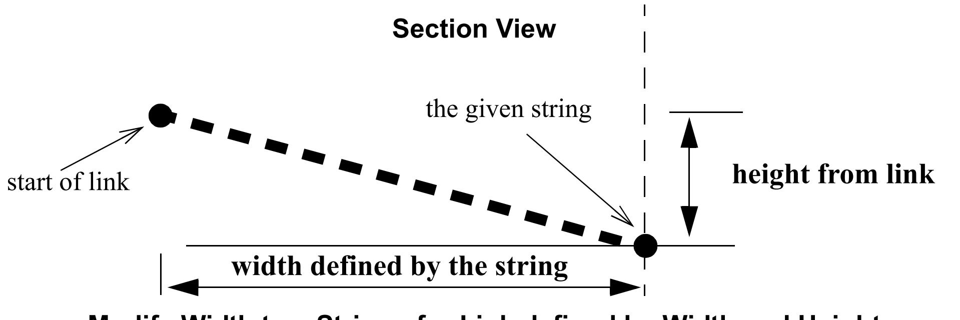
Fill Link - Calculate Width, Height or Slope to a String

Width to String:
For a fill link defined by *width and height* or *width and slope*, **Fill Width to string** calculates the **width** of the link as the width from the start point of the link, to the **to the selected string**. The *slope* or *height* is taken from the link. The option will give an error for a link defined by *height* and *slope*.

However if the tin in the Apply MTF is cut before reaching the string, the ***fill link stops at the Apply MTF tin***.



Modify Width to a String - for Link defined by Width and Slope

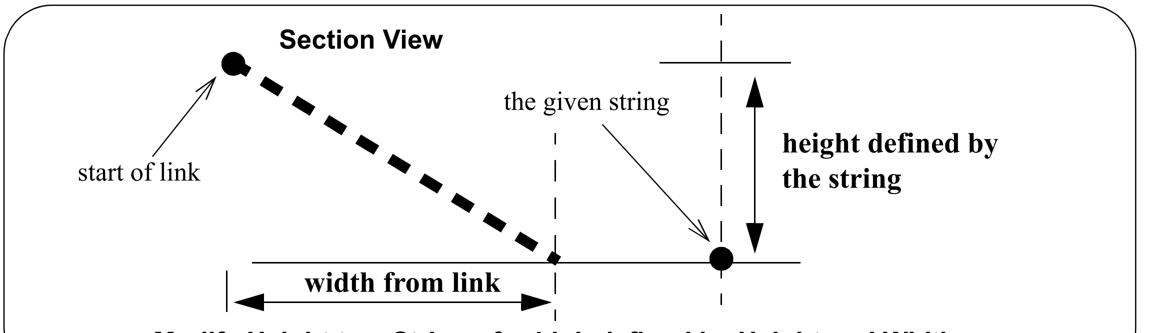


Modify Width to a String - for Link defined by Width and Height

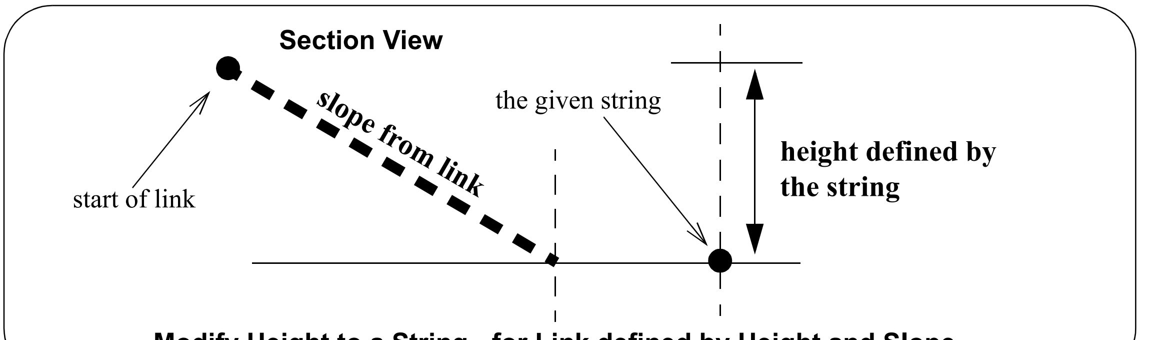
Height to String:

For a fill link defined by *height and width* or *height and slope*, **Fill Height to string** calculates the **height** of the link as the difference in the height at the start point of the link, and the height **at the selected string**. The *slope* or *width* is taken from the link. The option will give an error for a link defined by *width* and *slope*.

However if the tin in the Apply MTF is cut before reaching the string, the ***fill link stops at the Apply MTF tin***.



Modify Height to a String - for Link defined by Height and Width

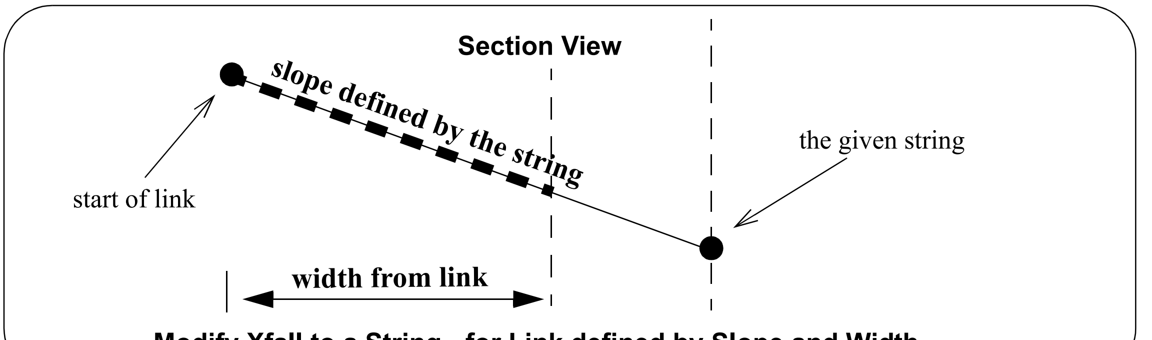


Modify Height to a String - for Link defined by Height and Slope

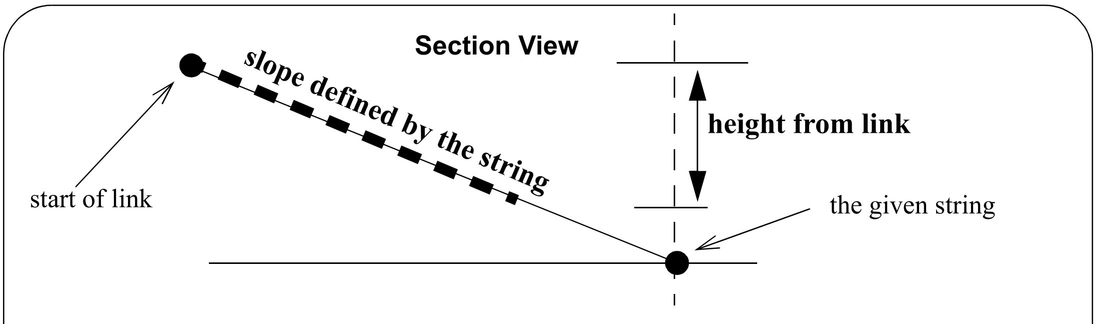
Slope to String:

For a fill link defined by *slope and width* or *slope and height*, **Fill Slope to string** calculates the **slope** of the link as the *slope* from the start point of the link to the **selected string**. The *width* or *height* is taken from the link. The option will give an error for a link defined by *width* and *height*.

However if the tin in the Apply MTF is cut before reaching the string, the ***fill link stops at the Apply MTF tin***.



Modify Xfall to a String - for Link defined by Slope and Width



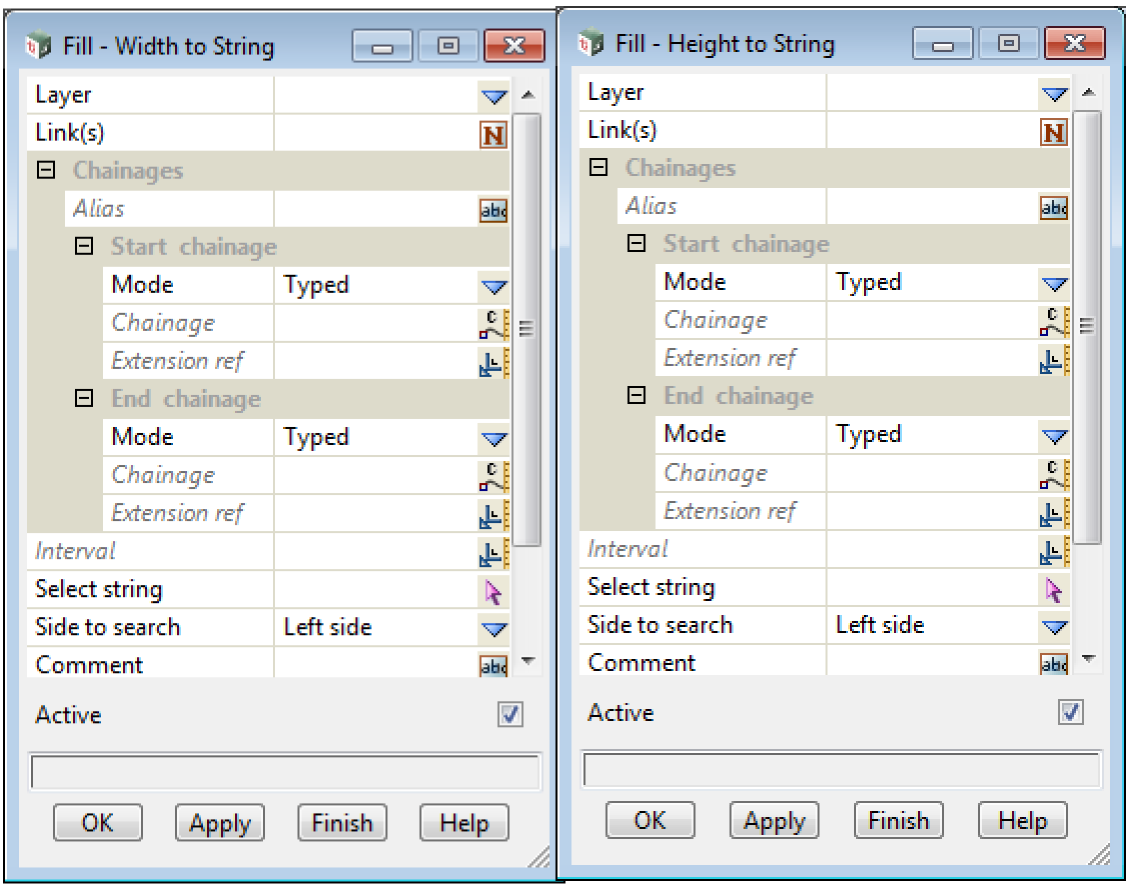
Modify Xfall to a String - for Link defined by Slope and Height

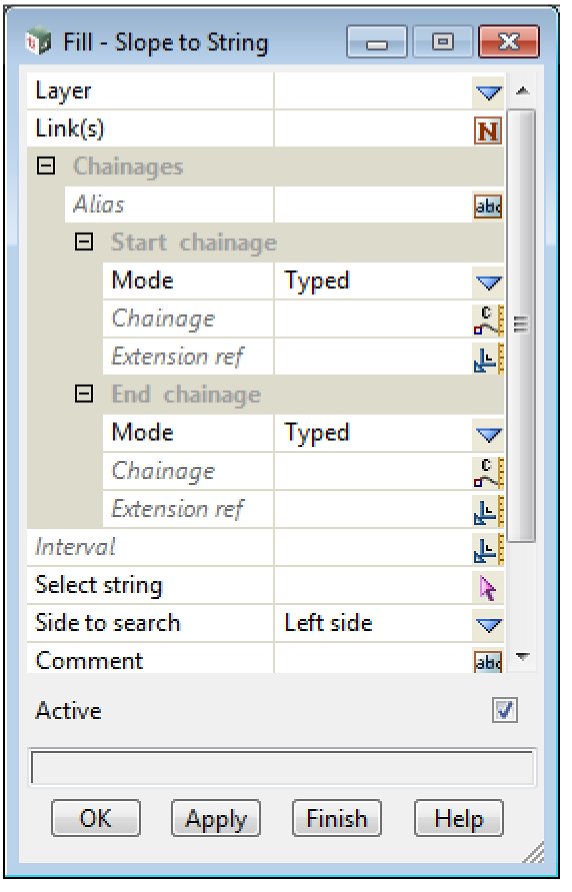
Note:

Using two of the above modifiers together and with the same string will place the end point of the link on the selected string. For example for a modifier defined by *width and slope*, using *width to string* and a *slope to string* with the same string will place the end of the link on that string.

But the To string option will do the same thing in one command. See Cut Link - Modify To String.

Selecting the **width to string, height to string or slope to string** option brings up the **Fill - Width to String**, **Fill - Height to String** and **Fill - Slope to String** panels respectively.





The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu

name of the link to modify.

Start/End mode	choice box	Start (ref)/End (ref)	

defines the start/end chainages for modifying the link
For more information on Start/End mode, see 21.2.1 MTF Hinge Modifiers ***except for this option**:*

*When the **Start mode** is **Start (ref),** or **Typed** and the chainage is **blank**, the modification begins at the low dropped chainage of the selected string.*

*When the **End mode** is **End (ref),** or **Typed** and the chainage is **blank**, the modification ends at the high dropped chainage of the selected string.*

Interval	input		

*if **non blank**, the interval to use to create cross sections and strings over the given chainage range. If **blank**, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

String	string-select		

select string to use for defining width/height/slope for the link.

Side to search	input	left side	left side, right side, both sides

side of the hinge string to start searching to find the string to define width/height/slope.

Extra start/end	tick box		

*if **ticked**, add an extra x-section 0.1 mm **before** the start/end chainage.*

Comment input

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active tick box tick

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

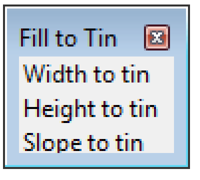
OK/Apply button

OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

Continue to the next section 21.3.0.16 Fill Link - to Tin or return to 21.3.0.10 Insert Fill Links from a Template.

21.3.0.16 Fill Link - to Tin

The *Fill* **to tin** walk-right brings up the **Fill to Tin** menu with options to calculate the width, height or slope to get to a given tin (which doesn't have to be the Apply MTF tin).

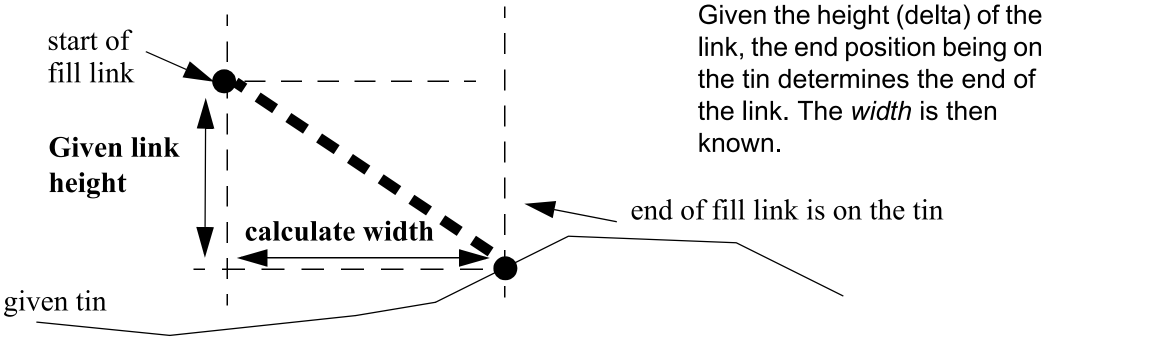


For *Width/Height/Tin to tin*, go to the next section Fill Link - Modify Width, Height or Xfall to Sit on User Tin

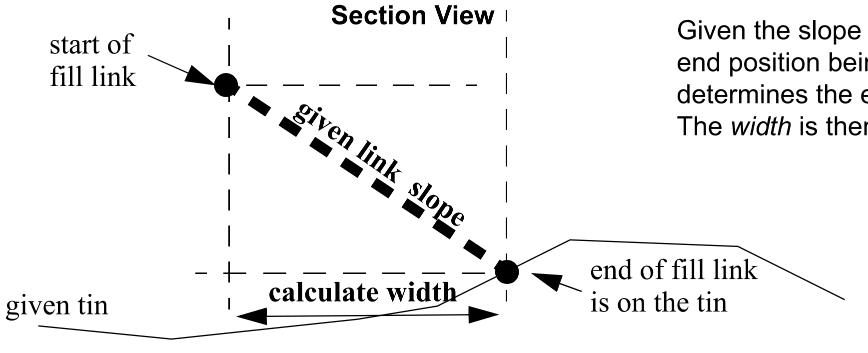
Fill Link - Modify Width, Height or Xfall to Sit on User Tin

Width to Tin

For a fill link defined by *width and height* or *width and slope*, **Width to tin** calculates the **width** of the link as the width required so that the link will sit on the user given tin at the ***height/slope given in the link***. The option gives an error for a link defined by *height and slope*.



Link - width and height - width calculated from the tin and given height (delta)



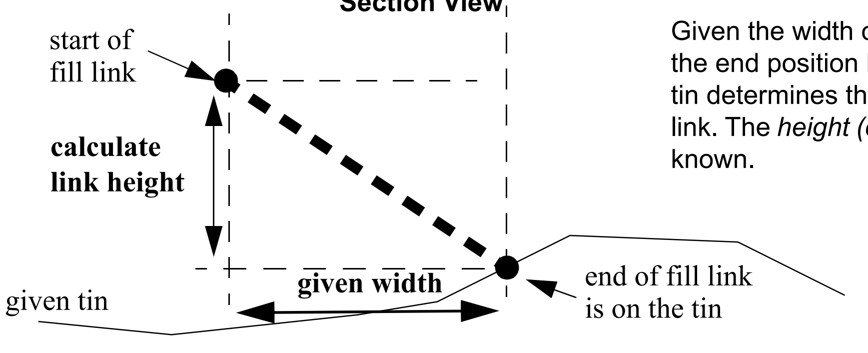
Given the slope of the link, the end position being on the tin determines the end of the link. The *width* is then known.

Link - width and slope - width calculated from the tin and given slope

Height to Tin

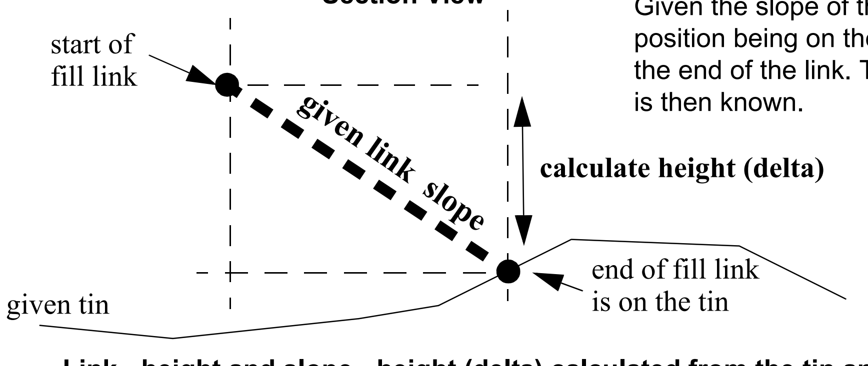
For a fill link defined by *height and width* or *height and slope*, the **Height to tin** calculates the ***height*** of the link as the difference in the height of the start point of the link, and the height that is required so that the link will sit on the user tin at the ***width/slope given in the link.*** The option gives an error for a link defined by *width and xfall*.

Important Note - for a Fill link, **positive** height and slopes are **down**.



Given the width of the link, the end position being on the tin determines the end of the link. The *height (delta)* is then known.

Link - height and width - height (delta) calculated from the tin and given width

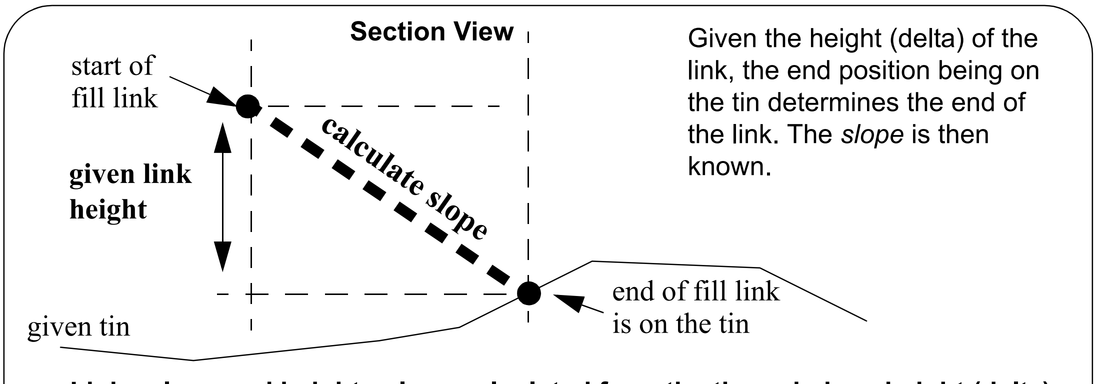


Given the slope of the link, the end position being on the tin determines the end of the link. The *height (delta)* is then known.

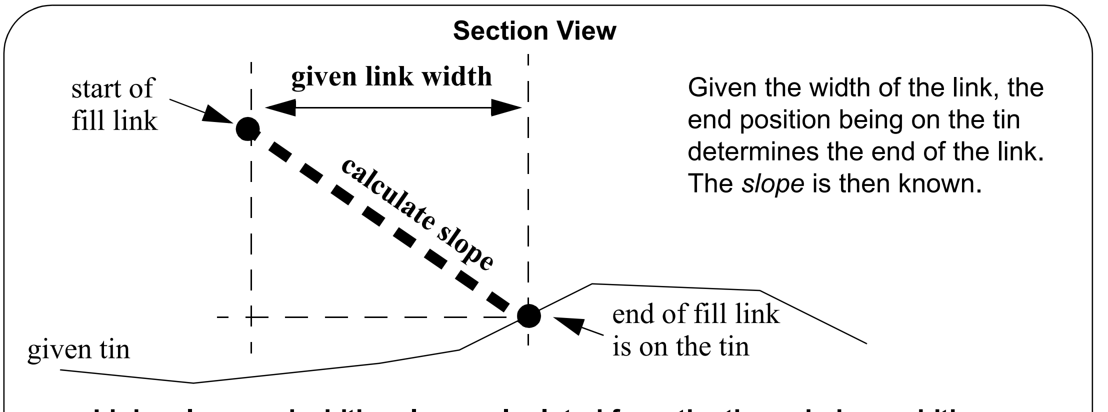
Link - height and slope - height (delta) calculated from the tin and given slope

Slope to Tin

For a fill link defined by *slope and width* or *slope and height*, **Slope to tin** calculates the **slope** of the link as the *slope* required so that the link will sit on the user tin at the ***width/height given in the link***. The option gives an error for a link defined by *width and height*.

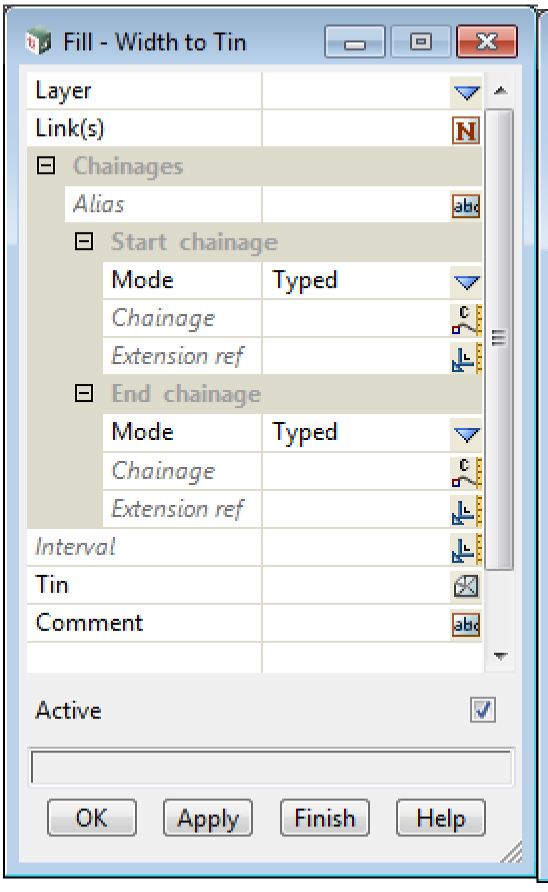


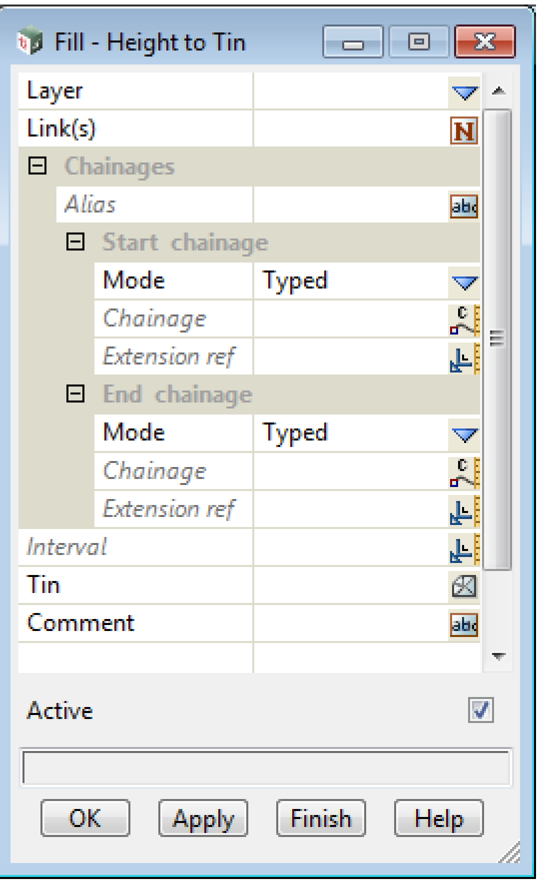
Link - slope and height - slope calculated from the tin and given height (delta)

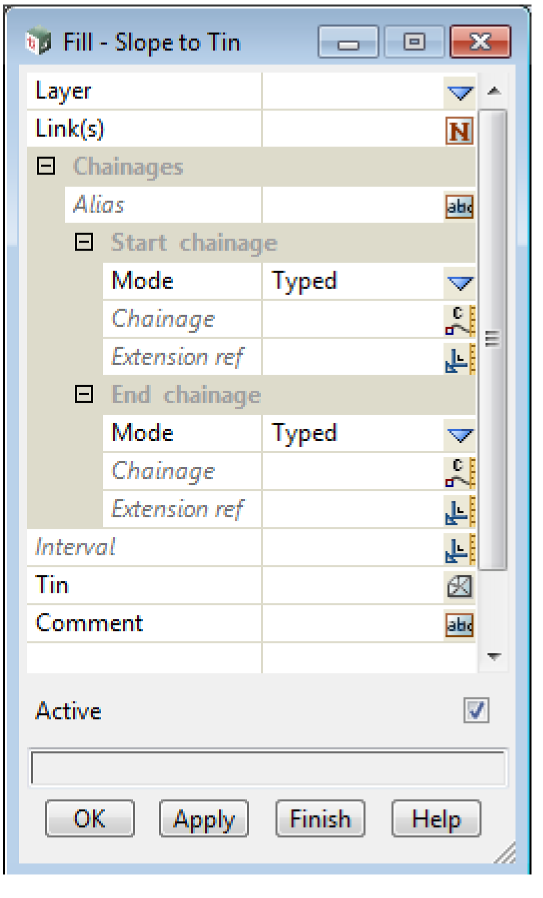


Link - slope and width - slope calculated from the tin and given width

Selecting **Width to tin**, **Height to tin, or Slope to tin** brings up the **Fill - Width to Tin**, **Fill - Height to Tin** and **Fill - Slope to Tin** panels respectively.


Fill - Width to Tin
Layer
Link(s)
Chainages
Alias
Start chainage
Mode
Typed
Chainage
Extension ref
End chainage
Mode
Typed
Chainage
Extension ref
Interval
Tin
Comment
Active
OK
Apply
Finish
Help


Fill - Height to Tin
Layer
Link(s)
Chainages
Alias
Start chainage
Mode
Typed
Chainage
Extension ref
End chainage
Mode
Typed
Chainage
Extension ref
Interval
Tin
Comment
Active
OK
Apply
Finish
Help


Fill - Slope to Tin
Layer
Link(s)
Chainages
Alias
Start chainage
Mode
Typed
Chainage
Extension ref
End chainage
Mode
Typed
Chainage
Extension ref
Interval
Tin
Comment
Active
OK
Apply
Finish
Help

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu

name of the link to modify.

Start/End mode	choice box	Start (ref)/End (ref)	

defines the start/end chainages for modifying the link
For more information on Start/End mode, see 21.2.1 MTF Hinge Modifiers

Interval	input		

*if **non blank**, the interval to use to create cross sections and strings over the given chainage range.*
*If **blank**, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

Tin	input		available tins

the tin to use for defining the width/height/slope

Extra start/end	tick box		

*if **ticked**, add an extra x-section 0.1 mm **before** the start/end chainage.*

Comment	input		

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active	tick box	tick	

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

OK/Apply	button		

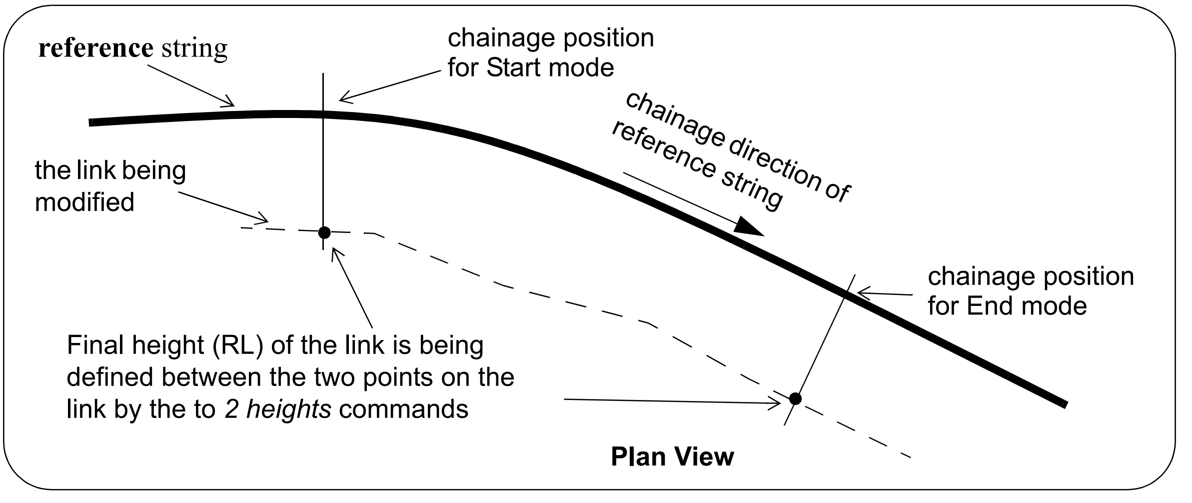
OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

Continue to the next section 21.3.0.17 Fill Link - to Two Heights or return to 21.3.0.10 Insert Fill Links from a Template.

21.3.0.17 Fill Link - to Two Heights

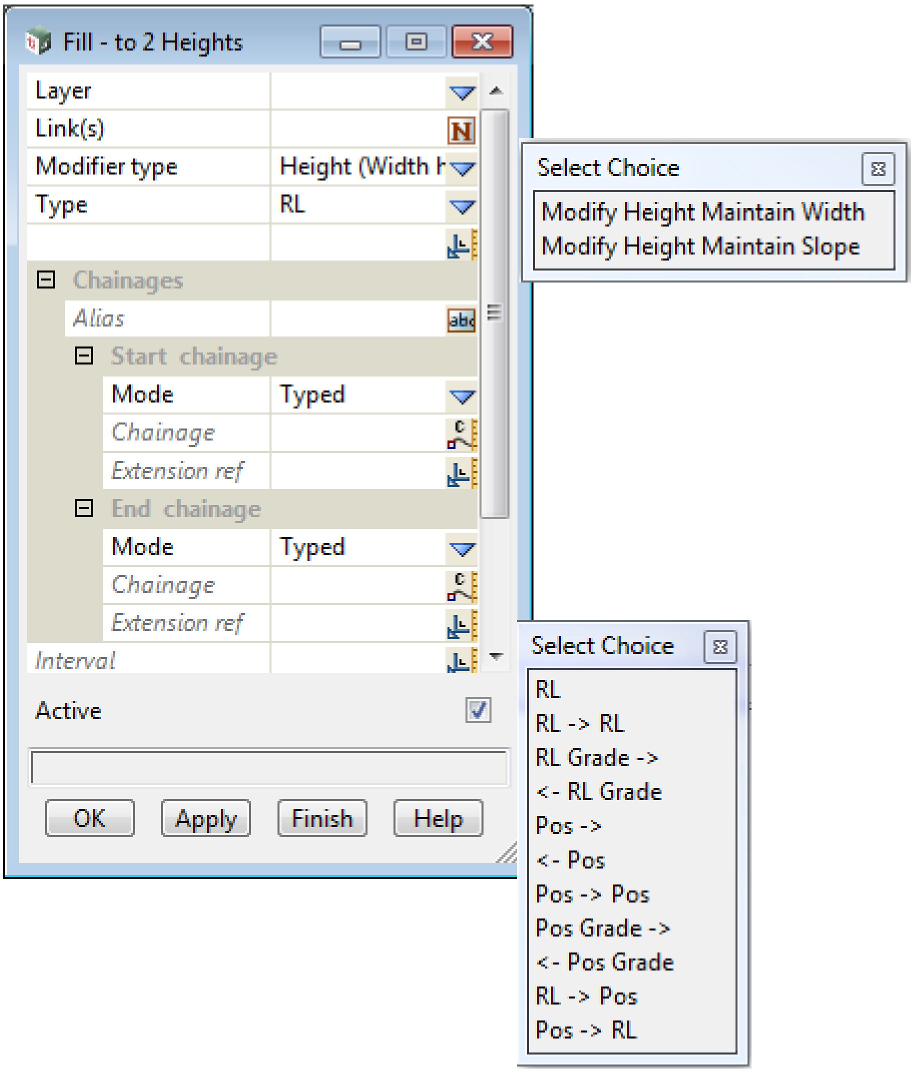
For an existing link, the **to 2 heights** option has a number of methods for **defining the height** from the start mode chainage to the end mode chainage. For example, over the chainage range, the height can be interpolated between two given RL's.

However if the *Apply MTF* tin is cut before reaching the end of the modified link, the ***fill link stops at the tin***.



Note: the ***2 Strings*** modifiers work with two points in a **section**. See 21.3.0.18 Fill Link - to Two Strings.

Selecting **o 2 heights** brings up the **Fill - to 2 Heights** panel

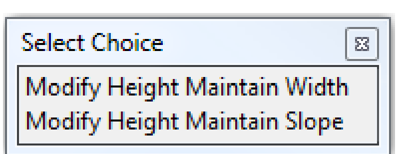


The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu

name of the link to modify.

Modifier type	choice box		



*The **Maintain** Slope/Width is the part of the new definition of the selected link that is taken from the selected* **Link name**. *See* Fill - Maintaining Width or Slope.

*The **Height** for the selected link is calculated by the method given by **Type**.*

Start/End mode	choice box	Start (ref)/End (ref)	

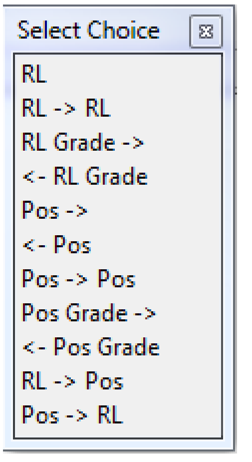
defines the start/end chainages for modifying the link.
For more information on Start/End mode, see 21.2.1 MTF Hinge Modifiers.

Interval	input		

*if **non blank**, the interval to use to create cross sections and strings over the given chainage range.*

*If **blank**, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

Type choice box



For the calculation of height for each type, go to Fill - Maintaining Width or Slope

Extra start/end tick box

*if **ticked**, add an extra x-section 0.1 mm **before** the start/end chainage.*

Comment input

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active tick box tick

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

OK/Apply button

OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

Fill - Maintaining Width or Slope

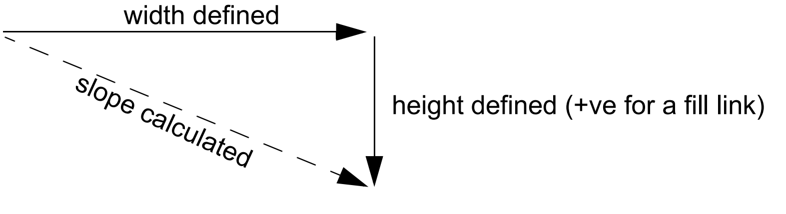
The option ***to 2 heights*** defines the ***height*** of the selected string so to completely define the link, only the width *or* the slope is needed.

So from the selected link only the *width* or the xfall is used (maintained) - the height is calculated by the **to 2 heights** option.

If the selected link is defined by ***width and height***, then at any chainage a ***slope*** can be calculated from the width and height at that chainage.

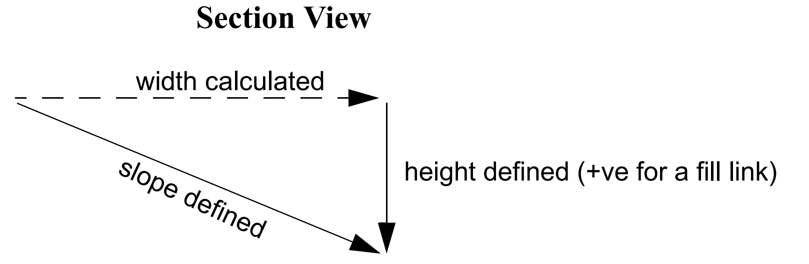
Important note - for a **fill link**, for slope and heights, **positive** is **down**.

Section View



For links defined by width and height, the ***slope*** at each chainage is uniquely defined

If the selected link is defined by ***slope and height***, then at any chainage a ***width*** can be calculated from the slope and height at that chainage.

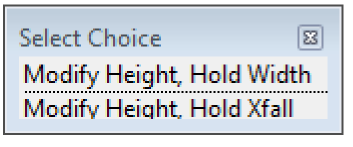


For links defined by slope and height, the ***width*** at each chainage is uniquely defined

Hence it doesn't matter if the selected link is defined by *width and slope*, *width and height*, or *slope and height*, at each chainage a *width and slope* are uniquely defined.

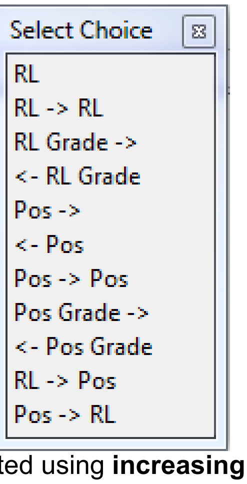
So no matter how the selected link is defined, at any chainage, a unique width or a slope is known and it is that **Width** or **Slope** can be maintained by the ***to 2 heights*** option.

So the choices for the option are:



Continue to the next section 21.3.0.18 Fill Link - to Two Strings or return to 21.3.0.17 Fill Link - to Two Heights, 21.3.0.10 Insert Fill Links from a Template.

Fill - Calculating the Heights for each Type



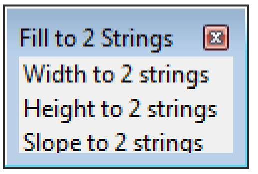
Important Note: Grade is calculated using **increasing Alignment chainages** and the **heights at the points** along the string being modifies. When going around a curve this will not be exactly the same as the grade along the string being modified.

The definitions of the calculations for each choice are the same as for the Fixed links. See 21.2.2.2.13.1 Fixed - Calculating the Heights for each Type.

Continue to the next section 21.3.0.18 Fill Link - to Two Strings or return to 21.3.0.17 Fill Link - to Two Heights, 21.3.0.10 Insert Fill Links from a Template.

21.3.0.18 Fill Link - to Two Strings

The *Fill* **to 2 strings** walk-right brings up the **Fill to 2 Strings** menu with options to calculate the width, height or slope from two given strings.



For *Width/Height/Slope to 2 Strings*, go to the next section [Fill Link - Width, Height or Slope Between Two Strings]

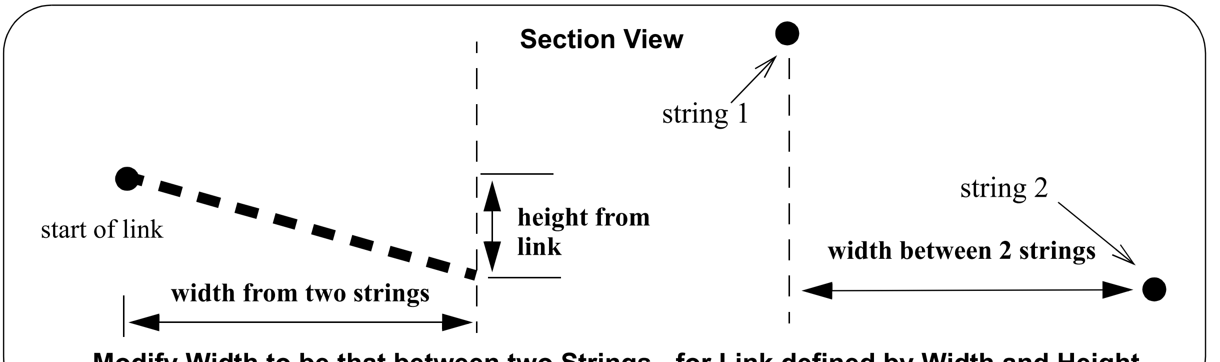
Fill Link - Width, Height or Slope Between Two Strings

Width Between Two Strings

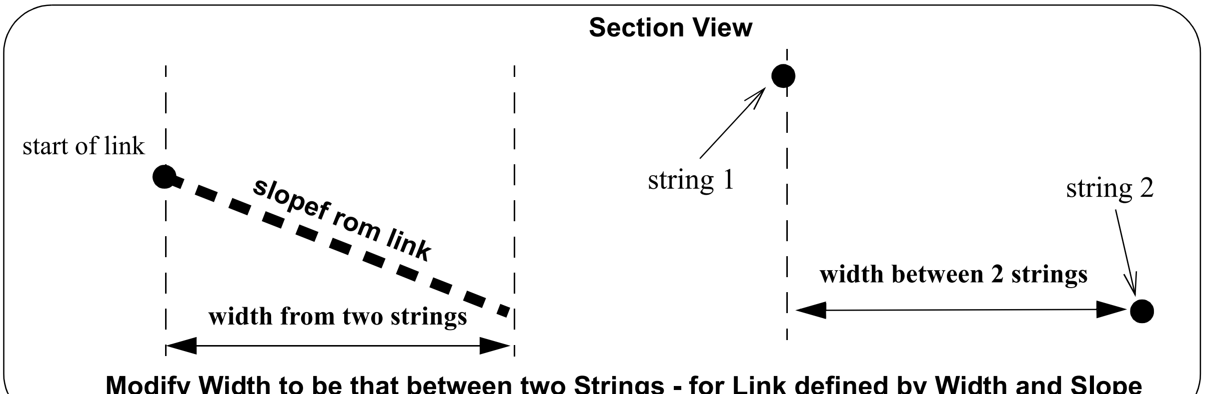
For a fill link defined by *width* and *height* or *width* and *slope*, **Width to 2 strings** sets the **width** for a link to be the **width between two** existing ***12d Model*** strings,

However if the Apply MTF tin is cut before reaching the string, the ***fill link stops at the tin***.

The option will give an error for a link defined by *height* and *slope*.



Modify Width to be that between two Strings - for Link defined by Width and Height



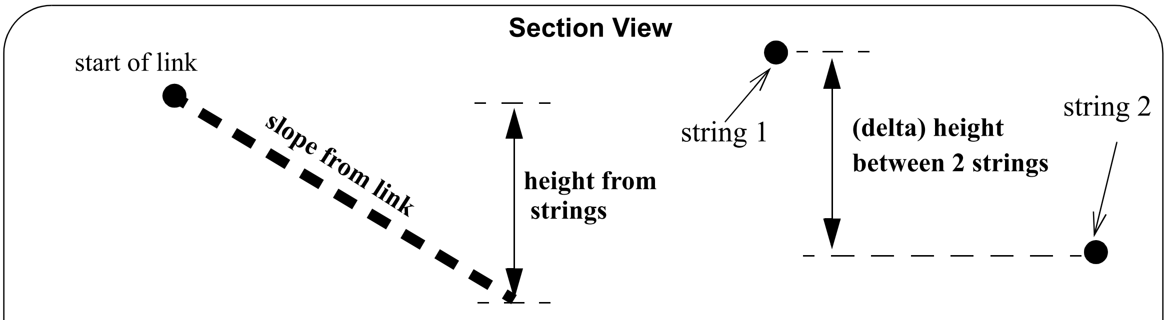
Modify Width to be that between two Strings - for Link defined by Width and Slope

Height Between Two Strings

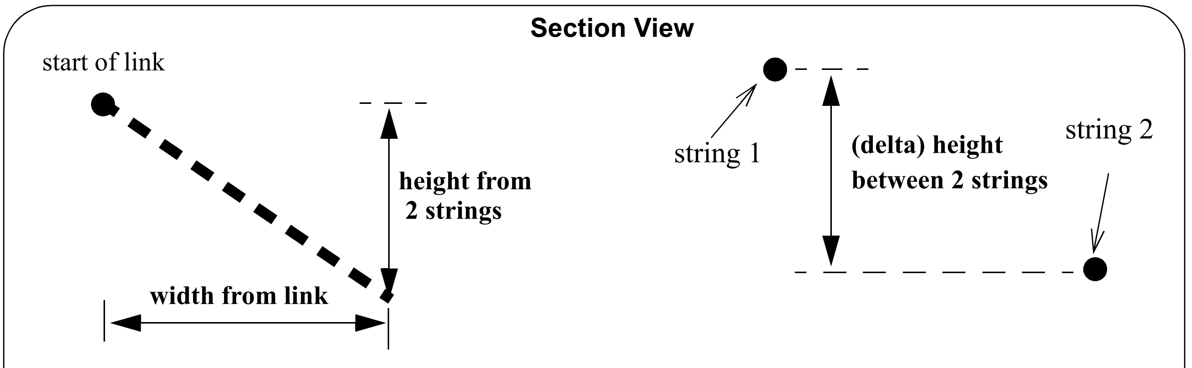
For a fill link defined by *height* and *width* or *height* and *slope*, **Height to 2 strings** sets the **height** for the link to be the **height between two** existing ***12d Model*** strings,

However if the Apply MTF tin is cut before reaching the string, the ***fill link stops at the tin***.

The option will give an error for a link defined by *width* and *slope*.



Modify Height to be that between two Strings - for Link defined by Height and Slope



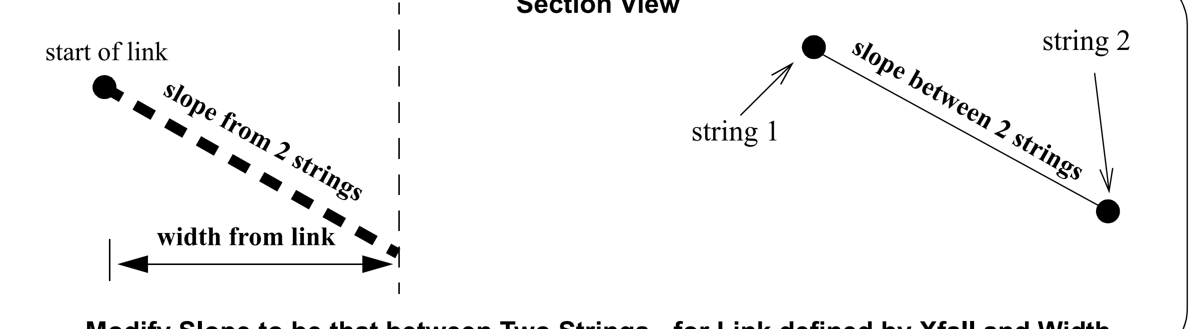
Modify Height to be that between two Strings - for Link defined by Height and Width

Slope Between Two Strings

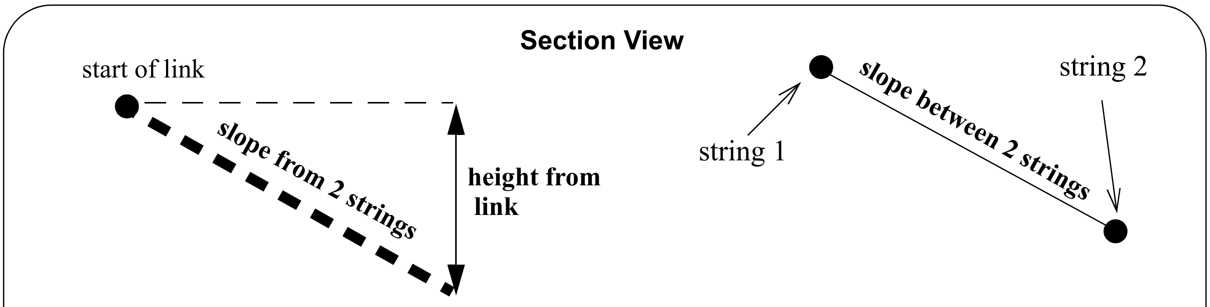
For a cut link defined by *slope* and *width* or *slope* and *height*, **Slope to 2 strings** sets the **slope** for the link to be the **slope between two** existing***12d Model*** strings,

However if the Apply MTF tin is cut before reaching the string, the ***cut link stops at the tin***.

The option will give an error for a link defined by *width* and *height*.

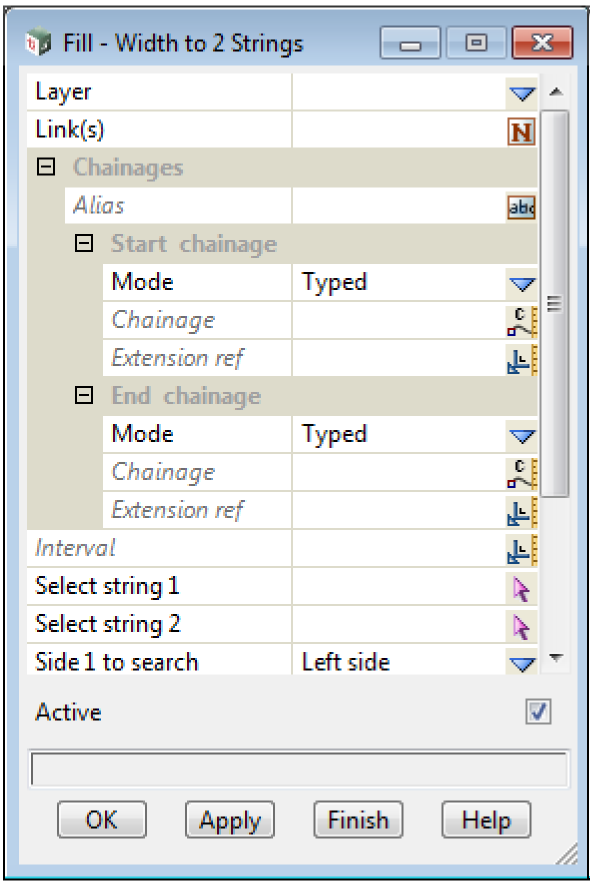


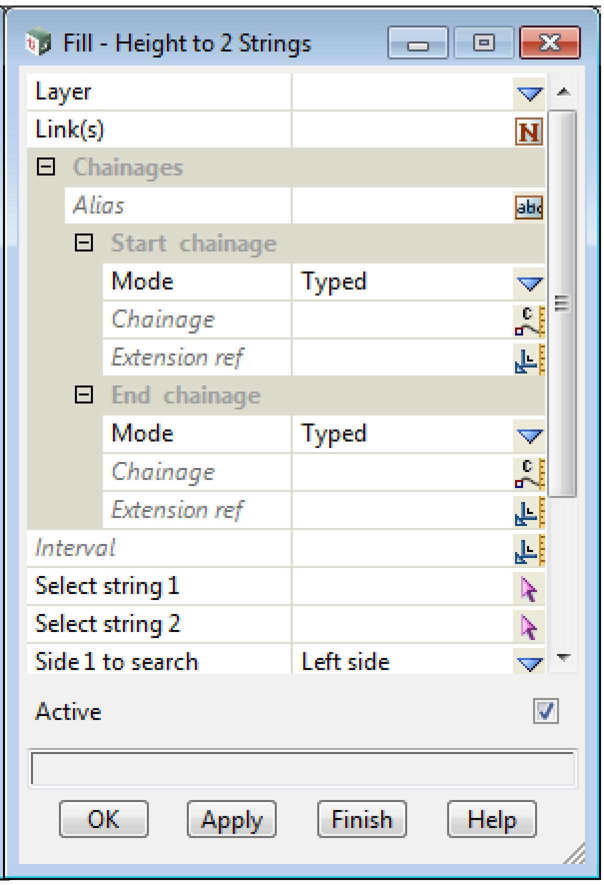
Modify Slope to be that between Two Strings - for Link defined by Xfall and Width

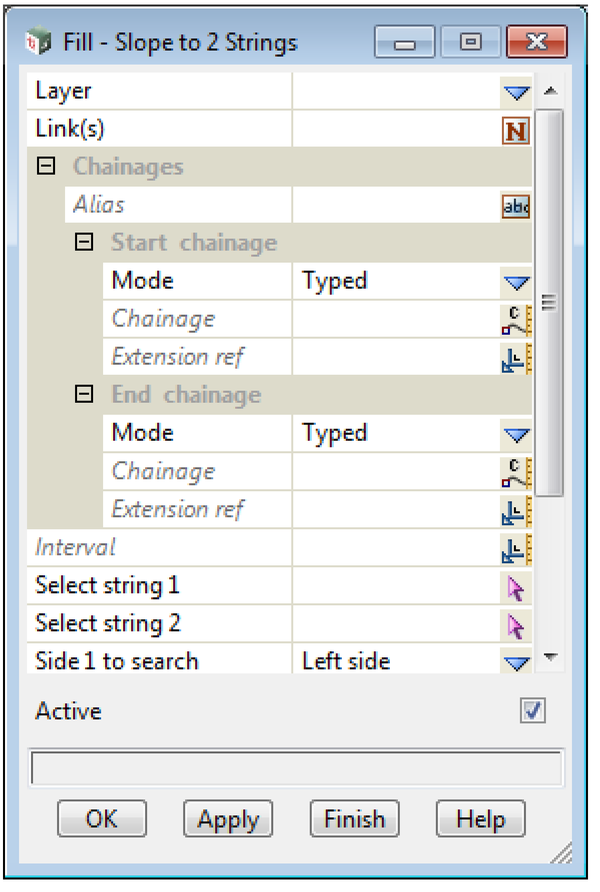


Modify Slope to be that between Two Strings - for Link defined by Xfall and Height

Selecting the **width to 2 strings**, **height to 2 strings or slope to 2 strings** option brings up the **Fill - Width to 2 Strings**, **Fill - Height to 2 Strings** and **Fill - Slope to 2 String** panels respectively.







The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Link name input select name menu

name of the link to modify.

Start/End mode choice box Start (ref)/End (ref)

defines the start/end chainages for modifying the link
For more information on Start/End mode, see 21.2.1 MTF Hinge Modifiers

Interval input

if ***non blank****, the interval to use to create cross sections and strings over the given chainage range. If* ***blank****, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

String 1 string-select

select the first string to use for defining width/height/slope for the link.

String 2 string-select

select the second string to use for defining width/height/slope for the link.

Side 1 to search input left side left side, right side, both sides

side of the hinge string to start searching to find string 1 to use in defining width/height/slope.

Side 2 to search input left side left side, right side, both sides

side of the hinge string to start searching to find string 2 to use in defining width/height/slope.

Extra start/end tick box

if ***ticked****, add an extra x-section 0.1 mm* ***before*** *the start/end chainage.*

Comment input

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active tick box tick

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

OK/Apply button

OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

Continue to the next section 21.4 Smart Chainages or return to 21.3.0.10 Insert Fill Links from a Template.

21.4 Smart Chainages

Many options in ***12d Model*** require a chainage value from another string, usually referred to as a **reference** string.

For example, in the Apply MTF and in all the left and right side modifiers in an MTF, there are ***start*** and ***end chainages*** where the chainage values refer to chainages on the reference string.

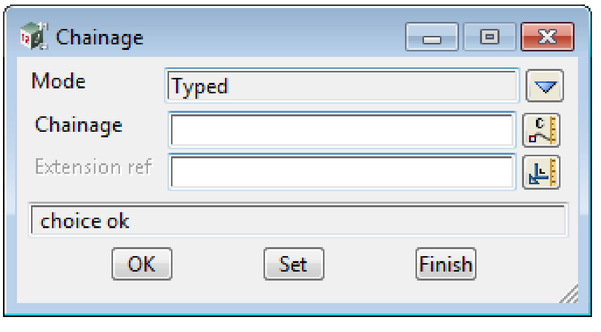
The traditional method of typing in a chainage value is supported but ***12d Model*** supports numerous other methods, called **smart chainages,** that calculate the chainage in various ways from the reference string geometry and from other strings.

The power of ***smart chainages*** is that when things move around, the smart chainages often don't have to be typed in again, but automatically recalculate themselves for the new situation. Hence the chainage values do not have to be revisited every time there is a change in geometry or for simple changes like modifying the start chainage of the reference string.

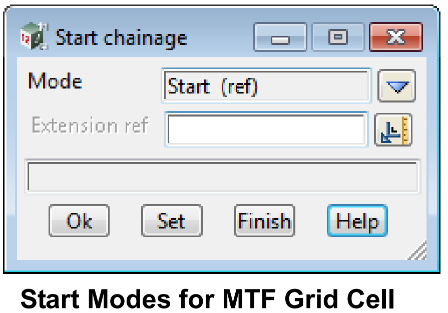
The ultimate goal is to be totally free of typed fixed chainages and to produce a **chainage free design**.

For panel fields and grids that allow smart chainages, instead of just a chainage field, there is also a **Mode** field with a pop up of all the smart chainages available for that field.

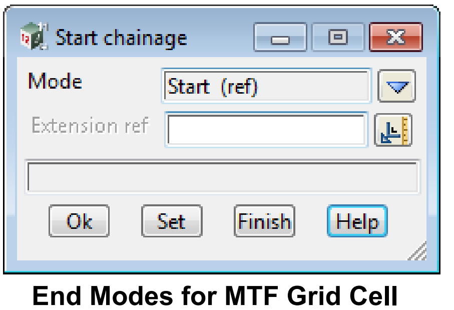
There is a **Modes**, **Start Mode** and/or **End Mode** depending on whether there is just a chainage value or the panel has start and end chainages and hence both a ***Start Mode*** and an ***End Mode***.



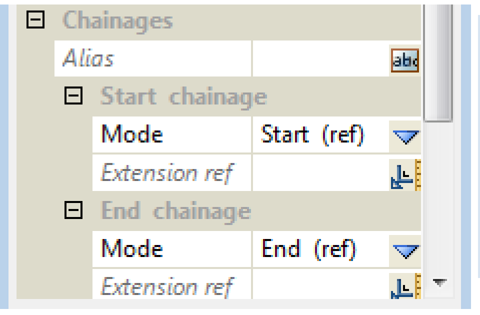
panel with just one chainage field in it and so just a Mode field



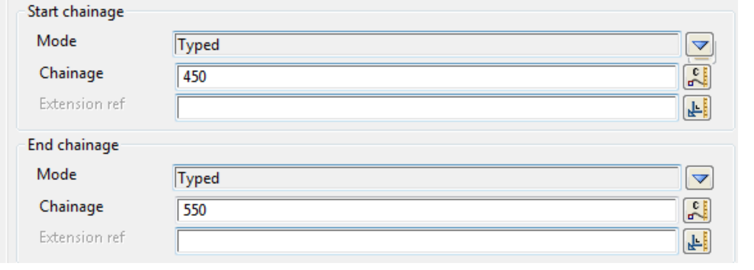
Start Modes for MTF Grid Cell



End Modes for MTF Grid Cell



Start and End Modes within MTF Modifiers

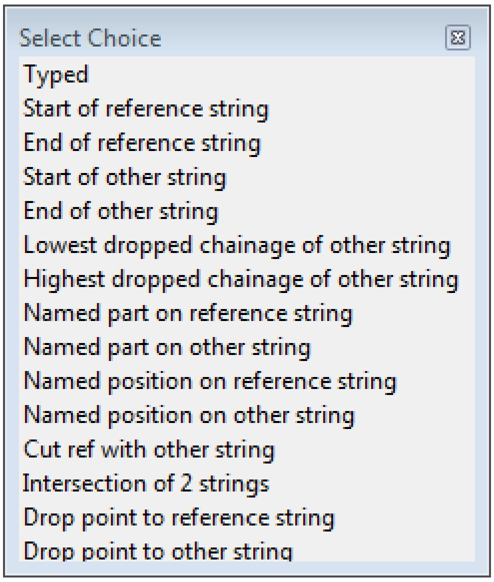


Start and End Modes within Apply MTF

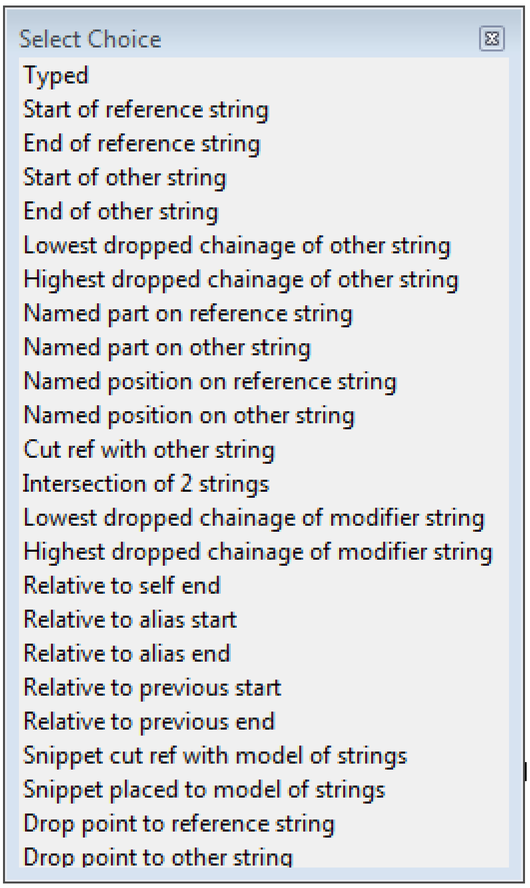
What choices are available for *Mode*, *Start Mode* and *End Mode* depends on the option that the smart chainage is in.

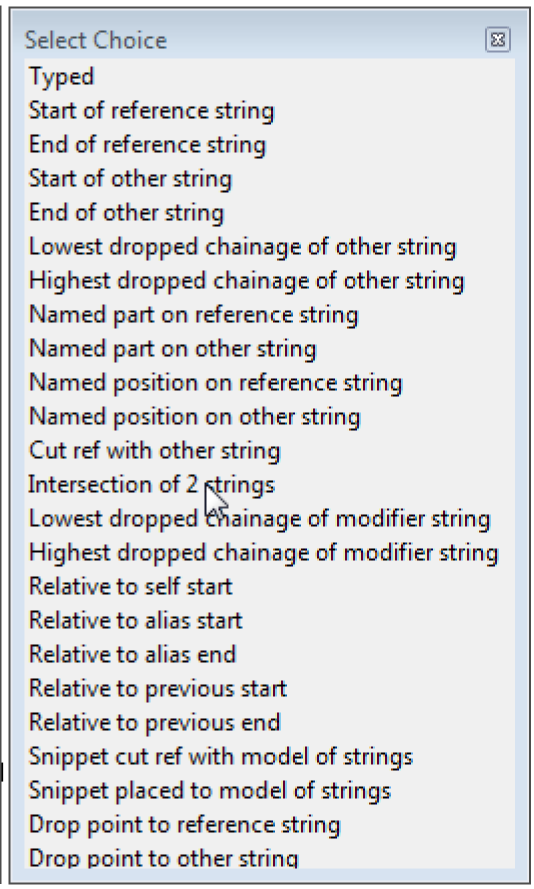
Finally most smart chainages also have an **Extension** field that is added to the chainage value calculated by the **Mode**.

For example, if the **Mode** was **End of reference string** and the **Extension** value was **-50** then the final smart chainage will be at chainage 50 before the end of the reference string, no matter what the actual end chainage is for the reference string. See 21.4.1 Extension Value.



Chainage Modes when option is not in the MTF Left and Right Modifiers





See

21.4.1 Extension Value
21.4.2 Typed
21.4.3 Start of Reference String
21.4.4 End of Reference String
21.4.5 Start of Other String
21.4.6 End of Other String
21.4.7 Lowest Dropped Chainage of Other String
21.4.8 Highest Dropped Chainage of Other String

21.4.1 Extension Value

For chainages given by **Start mode, End mode** or **Mode**, there is an **Extension ref** field in the **Chainage** panel and the **Extension ref** value is ***added*** to the reference chainage calculated by the **Smart Chainage**.

For example, if the **Mode** was **End of reference string** and the **Extension** value was **-50** then the final smart chainage will be at chainage 50 before the end of the reference string no matter what the actual end chainage is for the reference string.

In the cases when the **Mode** uses another string as well as the reference string, there is a **String** and **Extension other** field for the other string as well as an **Extension ref** field for the reference string.

For example, for the **Mode *Start of other string,*** there is a **Extension other** and an **Extension ref**.

To get the final chainage, the value of **Extension other** is added to the *Start chainage* of the string selected in **String** (the **other** string). This point on the other string is then dropped onto the Reference string.

The value of **Extension ref** is then added to the chainage of the dropped point onto the Reference string, to give the **final** chainage used.

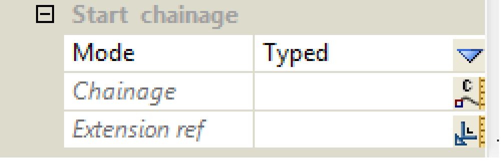
Notes

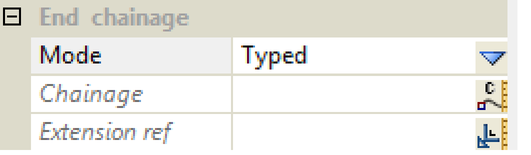
Extension other and **Extension ref** can be positive or negative.

Continue to the next section 21.4.2 Typed or return to 21.2.1 MTF Hinge Modifiers.

21.4.2 Typed

Selecting **Typed** adds a ***Chainage*** and an **Extension ref** field.





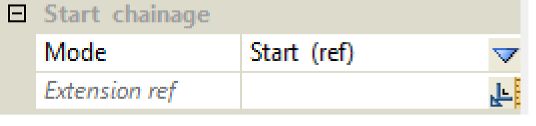
A reference string chainage is typed into the **Chainage** field and the value in the **Extension ref** field is then **added** to this value to give the chainage used as the ***final chainage***.

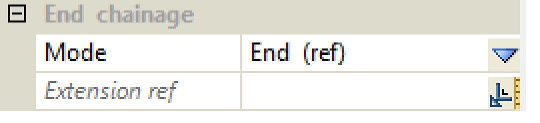
If the ***Chainage*** is left blank then the ***start/end chainage of the reference string*** is used as the **Chainage** value.

Continue to the next section 21.4.3 Start of Reference String or return to 21.2.1 MTF Hinge Modifiers.

21.4.3 Start of Reference String

Selecting **Start of reference string** adds an **Extension ref** field.



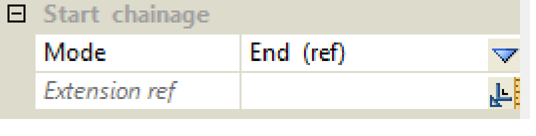


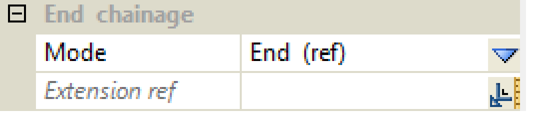
The value in the **Extension ref** field is **added** to the ***start chainage of the reference string*** to give the chainage used as the ***final chainage***.

Continue to the next section 21.4.4 End of Reference String or return to 21.2.1 MTF Hinge Modifiers.

21.4.4 End of Reference String

Selecting **End of reference string** adds an **Extension ref** field.



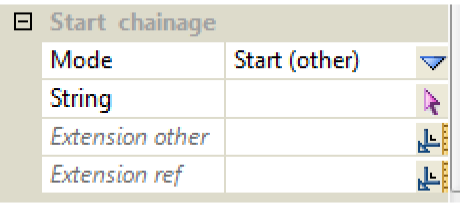


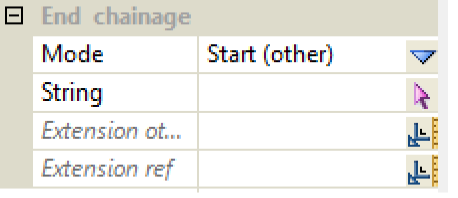
The value in the **Extension ref** field is **added** to the ***end chainage of the reference string*** to give the chainage used as the ***final chainage***.

Continue to the next section 21.4.5 Start of Other String or return to 21.2.1 MTF Hinge Modifiers.

21.4.5 Start of Other String

Selecting **Start of other string** adds ***String***, **Extension other** and **Extension ref** fields.



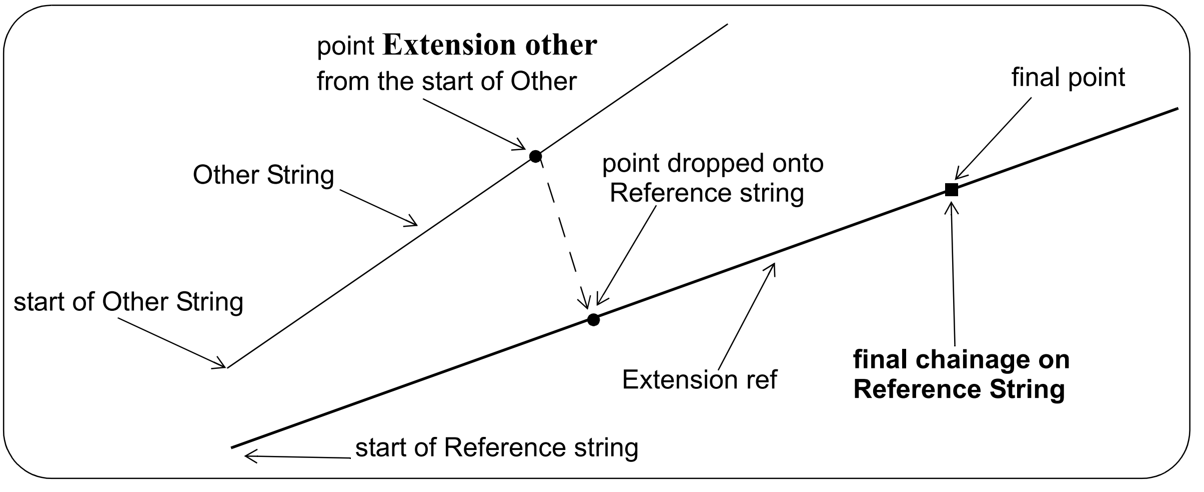


The **String** field is to select a string referred to as the **Other** string.

To get the final chainage, the value of **Extension other** is added to the ***Start chainage* of the Other string**. This point on the *other* string is dropped **onto** the Reference string.

The value of **Extension ref** is then added to the chainage on the Reference string of the point dropped onto the Reference string, to give the **final** chainage used.

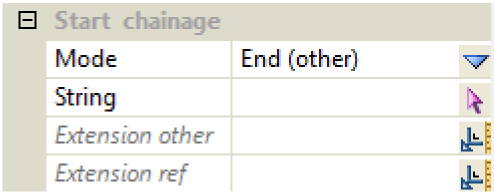
.

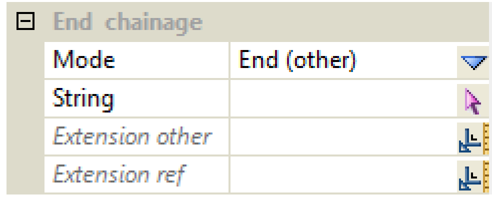


Continue to the next section 21.4.6 End of Other String or return to 21.2.1 MTF Hinge Modifiers.

21.4.6 End of Other String

Selecting **End of other string** adds ***String***, **Extension other** and **Extension ref** fields.



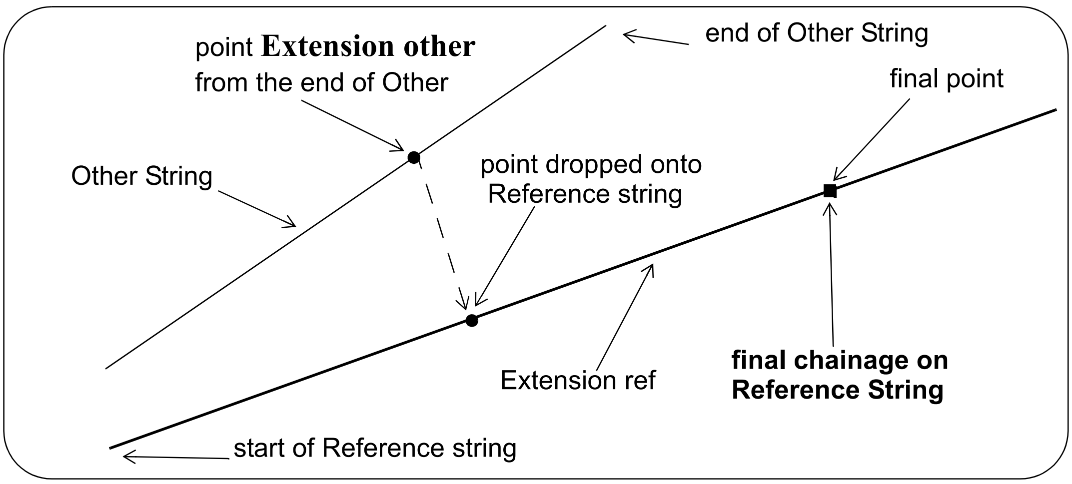


The **String** field is to select a string referred to as the **Other** string.

To get the final chainage, the value of **Extension other** is added to the ***End chainage* of the Other string**. This point on the *other* string is dropped **onto** the Reference string.

The value of **Extension ref** is then added to the chainage on the Reference string of the point dropped onto the Reference string, to give the **final** chainage used.

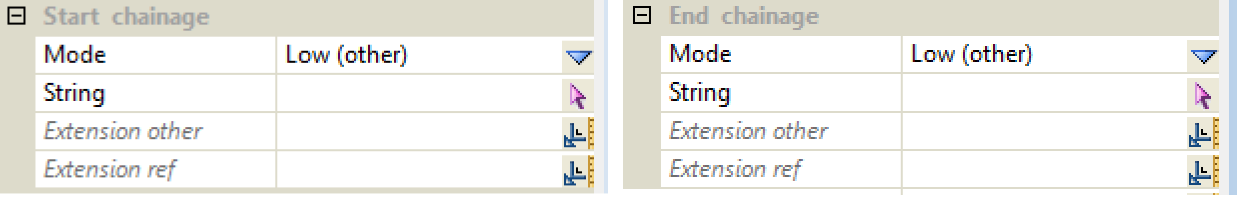
.



Continue to the next section [21.4.7 Lowest Dropped Chainage of Other String] or return to [21.2.1 MTF Hinge Modifiers].

21.4.7 Lowest Dropped Chainage of Other String

Selecting **Lowest dropped chainage of other string** adds ***String***, **Extension other** and **Extension ref** fields.



The **String** field is to select a string referred to as the **Other** string.

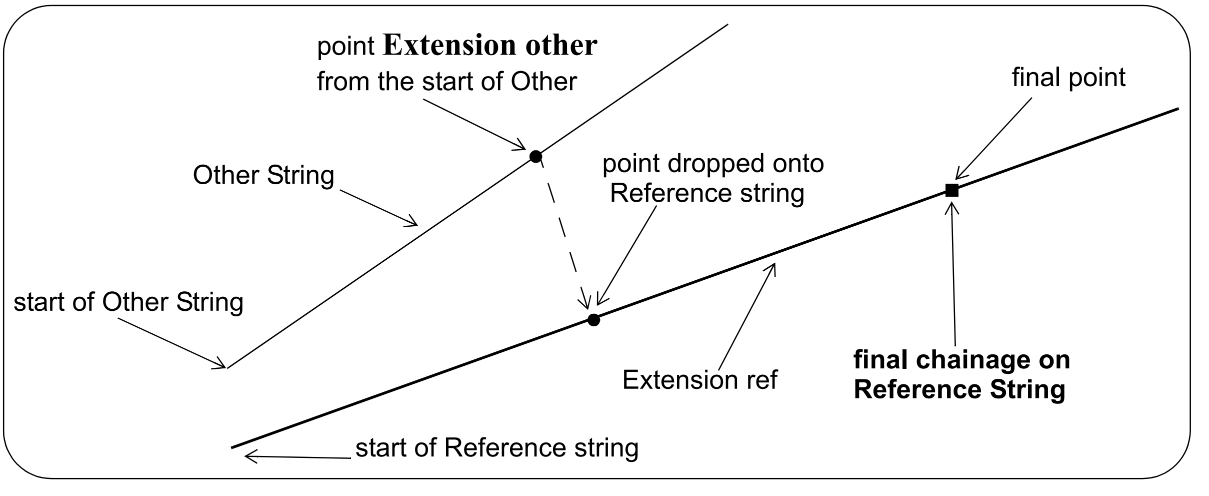
The *start* and *end* of the **Other** string are dropped onto the Reference string and the lower of the reference chainages of the two dropped end**??**.

How is Extension other used??

To get the final chainage, the value of **Extension other** is added to the ***Start chainage* of the Other string**. This point on the *other* string is dropped **onto** the Reference string.

The value of **Extension ref** is then added to the chainage on the Reference string of the point dropped onto the Reference string, to give the **final** chainage used.

.



Continue to the next section 21.4.8 Highest Dropped Chainage of Other String or return to 21.2.1 MTF Hinge Modifiers.

21.4.8 Highest Dropped Chainage of Other String

Selecting **Lowest dropped chainage of other string** adds ***String***, **Extension other** and **Extension ref** fields.

Start chainage		
Mode	High (other)	
String		
Extension other		
Extension ref		

End chainage		
Mode	High (other)	
String		
Extension other		
Extension ref		

The **String** field is to select a string referred to as the **Other** string.

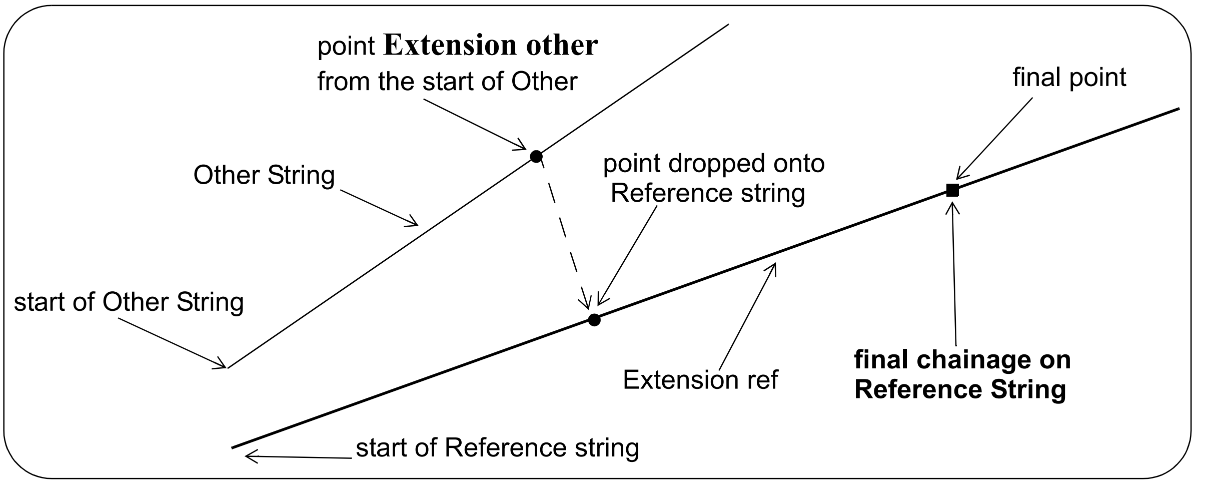
The *start* and *end* of the **Other** string are dropped onto the Reference string and the higher of the reference chainage of the two dropped end**??**.

How is Extension other used??

To get the final chainage, the value of **Extension other** is added to the ***Start chainage* of the Other string**. This point on the *other* string is dropped **onto** the Reference string.

The value of **Extension ref** is then added to the chainage on the Reference string of the point dropped onto the Reference string, to give the **final** chainage used.

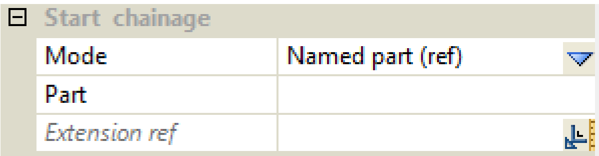
.

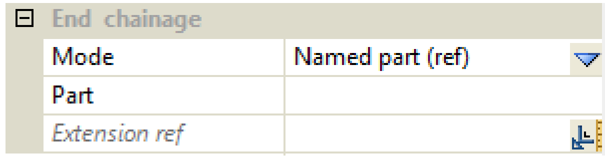


Continue to the next section [21.4.9 Named Part on Reference String]() or return to [21.2.1 MTF Hinge Modifiers]().

21.4.9 Named Part on Reference String

Selecting **Named part on reference string** adds the **Part** and **Extension ref** fields.

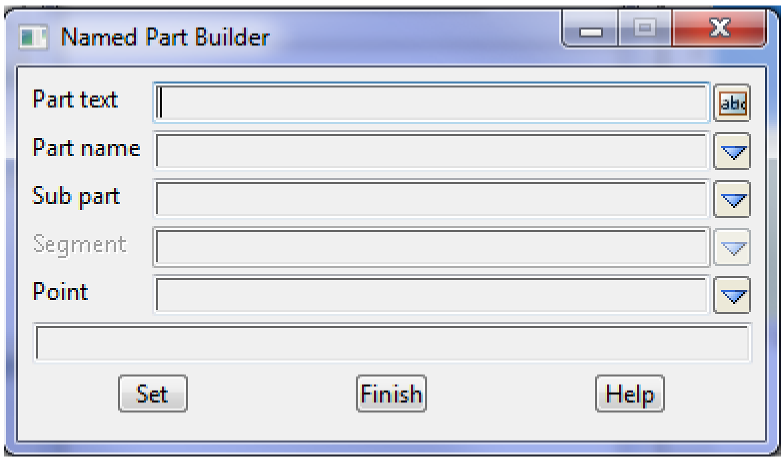




The syntax for the required **Named Part** on the **Reference** string can be typed into the **Part** field.

Or clicking LB or RB on the **+** for **Part** brings up the **Named Part Builder** for the reference string and this can be used to create the **Par text** for the section of a named part of the reference string that is required. Pressing **Set** then puts this value into the **Part** field.

.



The value in the **Extension ref** field is **added** to the reference chainage calculated for the selected named part to give the chainage used as the ***final chainage***.

Important Note

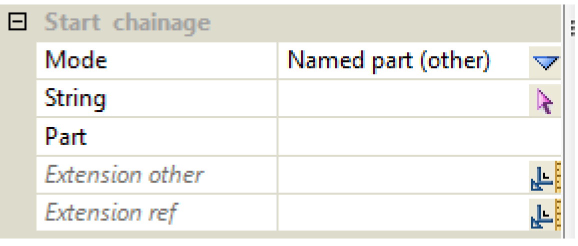
The **Named Parts** fields access the ***Vertical named parts*** as well as the ***Horizontal named Parts*** of a Super Alignment and the **Part name** pop up list **both** the horizontal and vertical named parts.

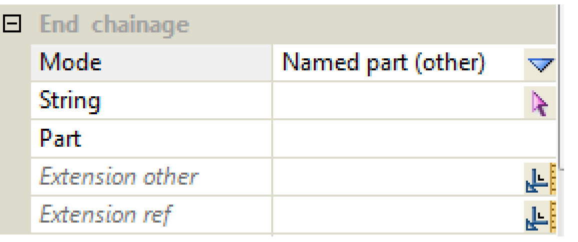
The **Modes** cannot distinguish between the names for horizontal and vertical named parts so **the horizontal names are given priority**. So if a vertical part has the same names as a horizontal part then the vertical name will not be shown in the **Part name** list.

Continue to the next section 21.4.10 Named Part on Other String or return to 21.2.1 MTF Hinge Modifiers.

21.4.10 Named Part on Other String

Selecting **Named part on other string** adds the **String**, **Part**, **Extension other** and **Extension ref** fields.



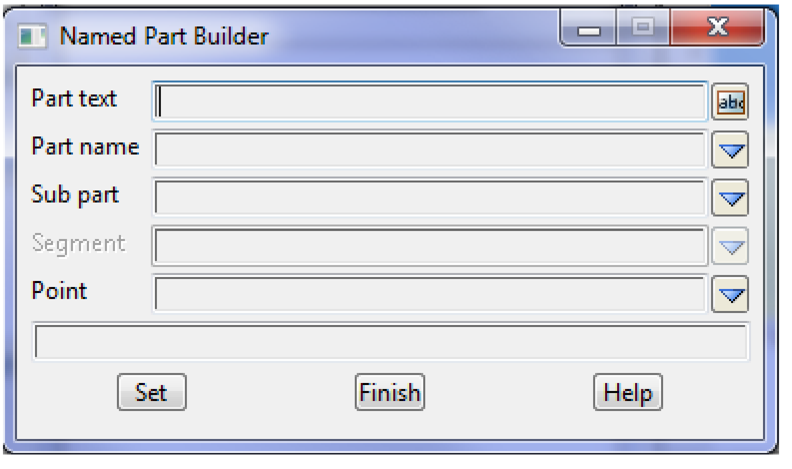


The **String** field is to select a string referred to as the **Other** string.

The syntax for the required **Named Part** on the **Other** string can be typed into the **Part** field.

Or clicking LB or RB on the **+** for **Part** brings up the **Named Part Builder** for the **Other** string and this can be used to create the **Par text** for the section of a named part of the Other string that is required. Pressing **Set** then puts this value into the **Part** field.

.



The value in the **Extension ref** field is **added** to the reference chainage calculated for the selected named part to give the chainage used as the ***final chainage***.

Important Note

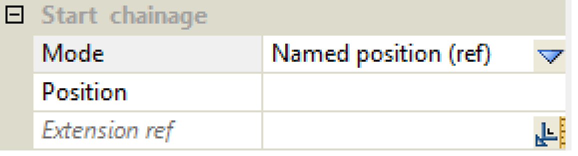
The **Named Parts** fields access the ***Vertical named parts*** as well as the ***Horizontal named Parts*** of a Super Alignment and the **Part name** pop up list **both** the horizontal and vertical named parts.

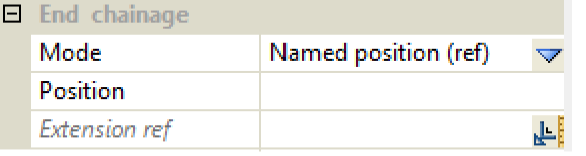
The **Modes** cannot distinguish between the names for horizontal and vertical named parts so **the horizontal names are given priority**. So if a vertical named part has the same names as a horizontal named part then the vertical name will not be shown in the **Part name** list.

Continue to the next section 21.4.11 Named Position on Reference String or return to 21.2.1 MTF Hinge Modifiers.

21.4.11 Named Position on Reference String

Selecting **Named position on reference string** adds the **Position** and **Extension ref** fields.





A ***Named Position*** for the ***Reference*** string can be typed into the **Position** field, or selected from the **Position** pop up which brings up the list of all *Named Positions* for the **Reference** string.

The value in the **Extension ref** field is **added** to the reference chainage calculated for the selected named position to give the chainage used as the ***final chainage***.

Important Note

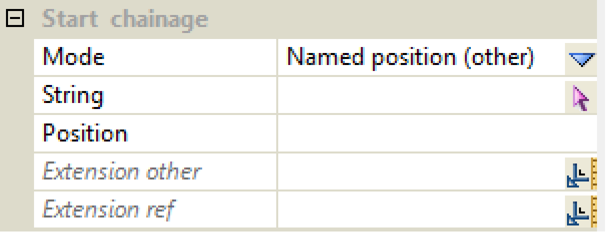
The **Named Position** fields access the ***Vertical named positions*** as well as the ***Horizontal named positions*** of a Super Alignment and the **Position** pop up list **both** the horizontal and vertical named positions.

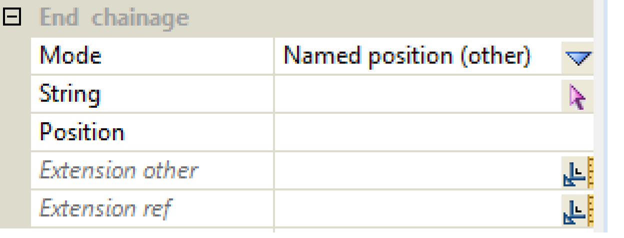
The **Modes** cannot distinguish between the names for horizontal and vertical named positions so **the horizontal names are given priority**. So if a vertical named part position has the same name as a horizontal named position then the vertical name will not be shown in the **Named Position** list.

Continue to the next section 21.4.12 Named Position on Other String or return to 21.2.1 MTF Hinge Modifiers.

21.4.12 Named Position on Other String

Selecting **Named position on other string** adds the **String**, **Position**, **Extension other** and **Extension ref** fields.





The **String** field is to select a string referred to as the **Other** string.

A ***Named Position*** for the ***Other*** string can be typed into the **Position** field, or selected from the **Position** pop up which brings up the list of all *Named Positions* for the **Other** string.

The value in the **Extension ref** field is **added** to the reference chainage calculated for the selected named part to give the chainage used as the ***final chainage***.

Important Note

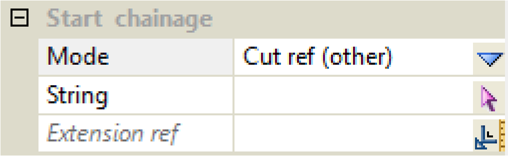
The **Named Position** fields access the ***Vertical named positions*** as well as the ***Horizontal named positions*** of a Super Alignment and the **Position** pop up list **both** the horizontal and vertical named positions.

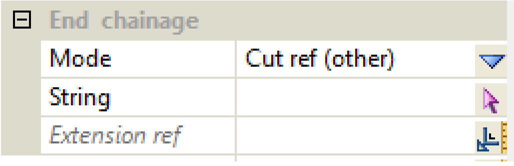
The **Modes** cannot distinguish between the names for horizontal and vertical named positions so **the horizontal names are given priority**. So if a vertical named part position has the same name as a horizontal named position then the vertical name will not be shown in the **Named Position** list.

Continue to the next section 21.4.13 Cut Reference with Other String or return to 21.2.1 MTF Hinge Modifiers.

21.4.13 Cut Reference with Other String

Selecting **Cut reference with other string** adds the **String** and **Extension ref** fields.



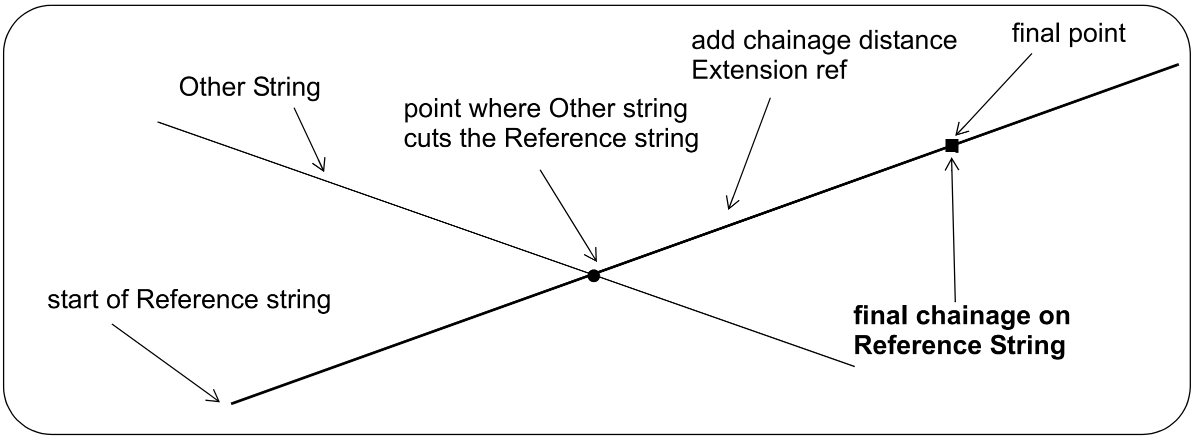


The **String** field is to select a string referred to as the **Other** string.

The point on the Reference string where the Other string cuts the Reference string is first calculated.

The value in the **Extension ref** field is **added** to the reference chainage calculated for the cut point to give the chainage used as the ***final chainage***.

.

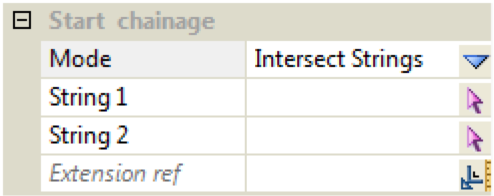


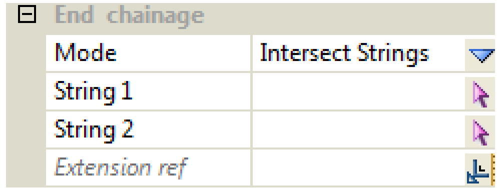
Note - if the other string does not cut the reference string, or cuts it in more than one place, then the *Mode* is invalid and an error message is generated.

Continue to the next section 21.4.14 Intersection of Two Strings or return to 21.2.1 MTF Hinge Modifiers.

21.4.14 Intersection of Two Strings

Selecting **Intersection of 2 strings** adds the **String 1**, **String 2** and **Extension ref** fields.





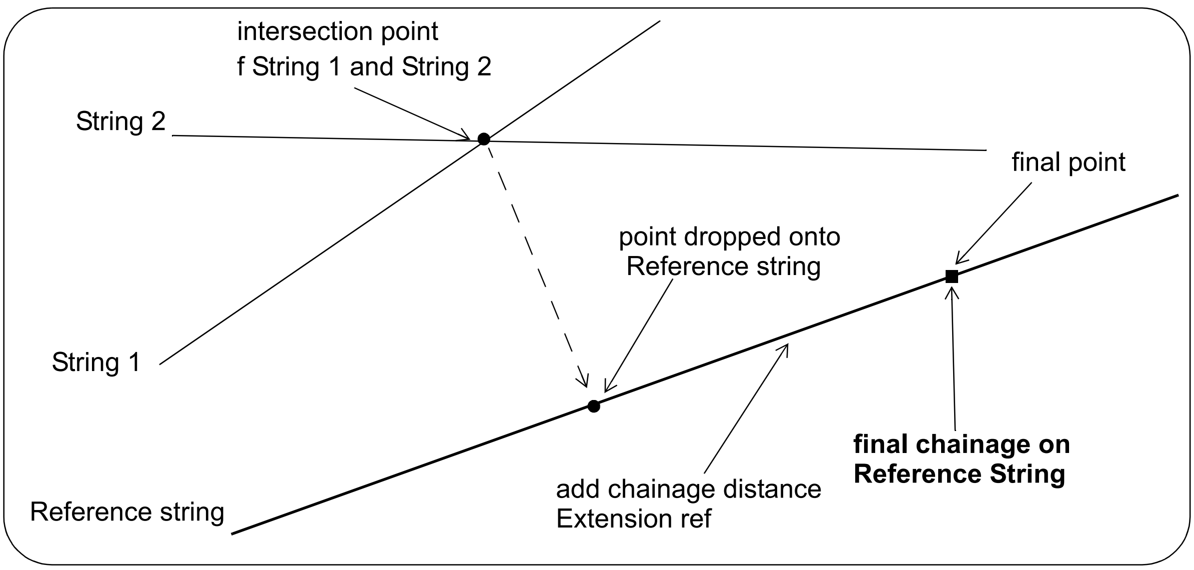
Two strings are selected with the **String 1** and **String 2** fields.

The value in the **Extension ref** field is **added** to the reference chainage calculated for the cut point to give the chainage used as the ***final chainage***.

The *intersection* point of the two selected strings is calculated and dropped onto the Reference string.

The value in the **Extension ref** field is **added** to the reference chainage of the dropped intersection point to give the chainage used as the ***final chainage***.

.

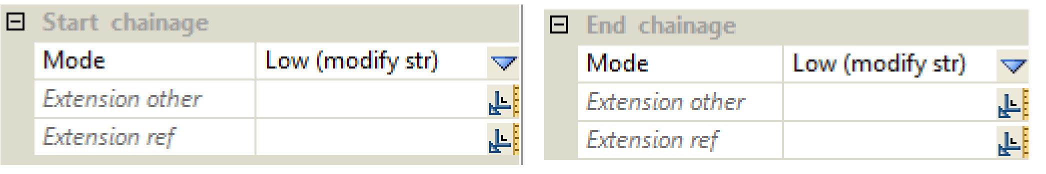


Continue to the next section 21.4.17 Relative to Self End or return to 21.2.1 MTF Hinge Modifiers.

21.4.15 Lowest Dropped Chainage of Modifier String

The Smart Chainage **Lowest dropped chainage of modifier string** is only available in the Apply MTF Left and Right Modifiers.

Selecting **Lowest dropped chainage of modifier string** adds **Extension other** and **Extension ref** fields.



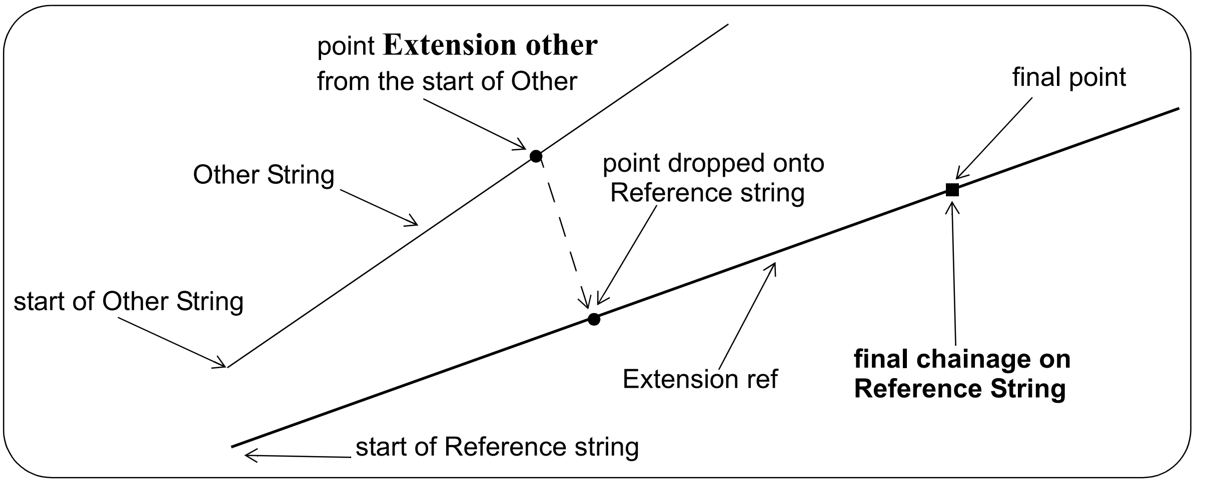
The *start* and *end* of the **Modifier** string are dropped onto the Reference string and the lower of the reference chainages of the two dropped ends**??**.

How is Extension other used??

To get the final chainage, the value of **Extension other** is added to the ***Start chainage* of the Other string**. This point on the *other* string is dropped **onto** the Reference string.

The value of **Extension ref** is then added to the chainage on the Reference string of the point dropped onto the Reference string, to give the **final** chainage used.

.

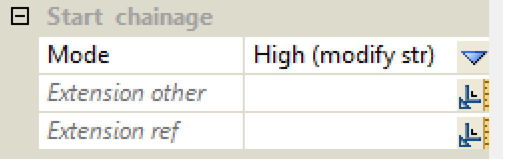


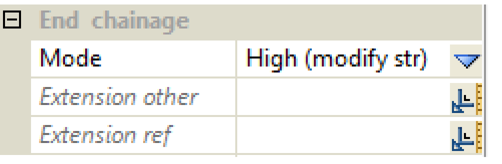
Continue to the next section 21.4.16 Highest Dropped Chainage of Modifier String or return to 21.2.1 MTF Hinge Modifiers.

21.4.16 Highest Dropped Chainage of Modifier String

The Smart Chainage **Lowest dropped chainage of modifier string** is only available in the Apply MTF Left and Right Modifiers.

Selecting **Lowest dropped chainage of modifier string** adds **Extension other** and **Extension ref** fields.





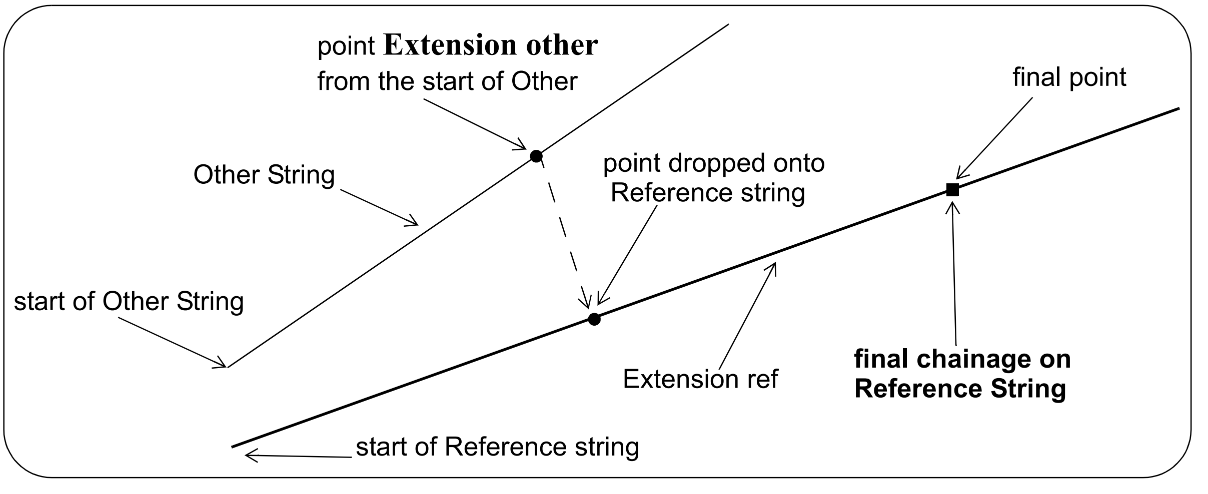
The *start* and *end* of the **Modifier** string are dropped onto the Reference string and the lower of the reference chainages of the two dropped ends**??**.

How is Extension other used??

To get the final chainage, the value of **Extension other** is added to the ***Start chainage* of the Other string**. This point on the *other* string is dropped **onto** the Reference string.

The value of **Extension ref** is then added to the chainage on the Reference string of the point dropped onto the Reference string, to give the **final** chainage used.

.

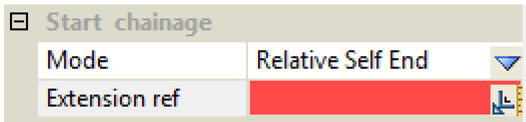


Continue to the next section 21.4.17 Relative to Self End or return to 21.2.1 MTF Hinge Modifiers.

21.4.17 Relative to Self End

The Smart Chainage **Relative to self end** is only available in the Apply MTF Left and Right Modifiers. This Mode is only available for a Start Chainage.

Selecting **Relative to Self End** adds an **Extension ref** field.



The value in the **Extension ref** field is **added** to the ***end chainage of the current MTF command*** to give the chainage used as the ***final chainage***.

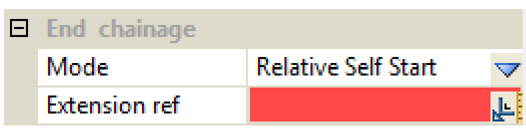
This Mode is useful when the **end chainage** of the current MTF command is known and you want the start chainage of the **same** command to be **relative** to the **known end chainage** of the **same command**.

Continue to the next section 21.4.18 Relative to Self Start or return to 21.2.1 MTF Hinge Modifiers.

21.4.18 Relative to Self Start

The Smart Chainage **Relative to self start** is only available in the Apply MTF Left and Right Modifiers. This Mode is only available for an End Chainage.

Selecting **Relative to Self Start** adds an **Extension ref** field.



The value in the **Extension ref** field is **added** to the ***start chainage of the current MTF command*** to give the chainage used as the ***final chainage***.

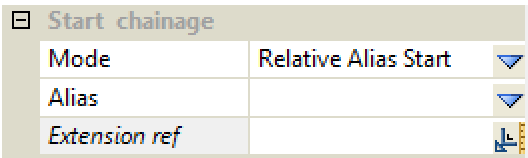
This Mode is useful when the **start chainage** of the current MTF command is known and you want the end chainage of the **same** command to be **relative** to the **known start chainage** of the **same command.**

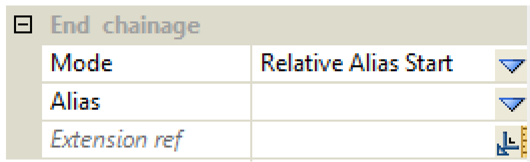
Continue to the next section 21.4.19 Relative to Alias Start or return to 21.2.1 MTF Hinge Modifiers.

21.4.19 Relative to Alias Start

The Smart Chainage **Relative to alias start** is only available in the Apply MTF Left and Right Modifiers.

Selecting **Relative to alias start** adds **Alias** and **Extension ref** fields.





An ***Alias*** from the current MTF can be typed into the **Alias** field, or selected from the **Alias** pop up which brings up the list of all *Aliases* in the current MTF.

The value in the **Extension ref** field is **added** to the ***start chainage*** *of the* ***selected Alias*** to give the chainage used as the ***final chainage***.

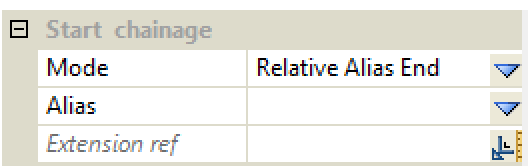
This Mode is useful when you want the start/end chainage to be **relative to the start chainage** of the command in the current MTF with the given Alias.

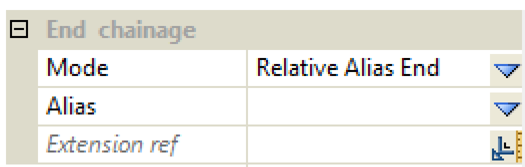
Continue to the next section 21.4.20 Relative to Alias End or return to 21.2.1 MTF Hinge Modifiers.

21.4.20 Relative to Alias End

The Smart Chainage **Relative to alias end** is only available in the Apply MTF Left and Right Modifiers.

Selecting **Relative to alias end** adds **Alias** and **Extension ref** fields.





An ***Alias*** from the current MTF can be typed into the **Alias** field, or selected from the **Alias** pop up which brings up the list of all *Aliases* in the current MTF.

The value in the **Extension ref** field is **added** to the ***end chainage*** *of the* ***selected Alias*** to give the chainage used as the ***final chainage***.

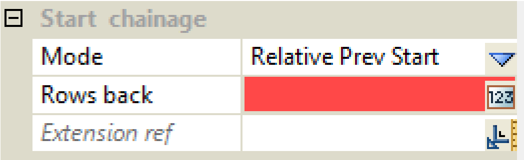
This Mode is useful when you want the start/end chainage to be **relative to the end chainage** of the command in the current MTF with the given Alias.

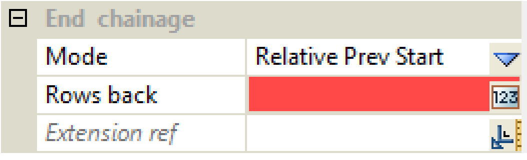
Continue to the next section 21.4.21 Relative to Previous Start or return to 21.2.1 MTF Hinge Modifiers.

21.4.21 Relative to Previous Start

The Smart Chainage **Relative to previous start** is only available in the Apply MTF Left and Right Modifiers.

Selecting **Relative to previous start** adds **Rows back** and **Extension ref** fields.





Rows back gives the number of rows in the current MTF to go back to get the **Start chainage** from. **Rows back** must be greater or equal to one.

The value in the **Extension ref** field is added to the ***start chainage*** *of the* ***selected row*** to give the chainage used as the ***final chainage***.

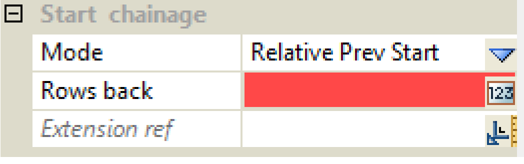
This Mode is useful when you want the start/end chainage to be **relative to the start chainage** of an earlier command that is a **known** number of **rows back** in the current MTF.

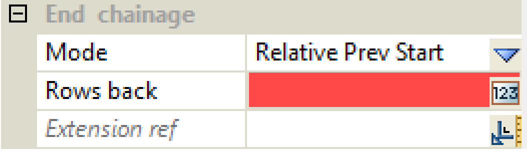
Continue to the next section 21.4.22 Relative to Previous End or return to 21.2.1 MTF Hinge Modifiers.

21.4.22 Relative to Previous End

The Smart Chainage **Relative to previous end** is only available in the Apply MTF Left and Right Modifiers.

Selecting **Relative to previous end** adds **Rows back** and **Extension ref** fields.





Rows back gives the number of rows in the current MTF to go back to get the **End chainage** from. **Rows back** must be greater or equal to one.

The value in the **Extension ref** field is added to the ***end chainage*** *of the* ***selected row*** to give the chainage used as the ***final chainage***.

This Mode is useful when you want the start/end chainage to be **relative to the end chainage** of an earlier command that is a **known** number of **rows back** in the current MTF.

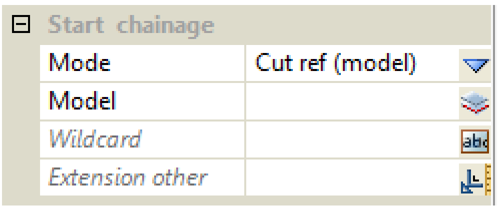
Continue to the next section 21.4.23 Snippet Cut Reference with Model of Strings or return to 21.2.1 MTF Hinge Modifiers.

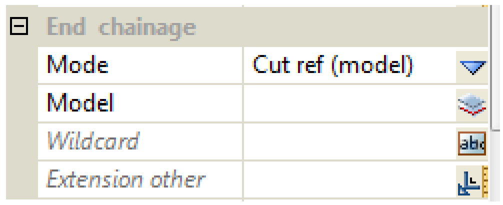
21.4.23 Snippet Cut Reference with Model of Strings

The Smart Chainage **Snippet cut ref with model of strings** is only available with the **Snippet** option in the Apply MTF Left and Right Modifier.

This Mode takes a model of strings and the Snippet is applied where **each string** in the model cuts the Reference string.

Selecting **Snippet cut ref with model of string** adds the **Model**, **Wildcard** and **Extension other** fields.





Model model box available models

the model of strings that is used for the snipped to be applied to where each of the strings cuts the Reference string.

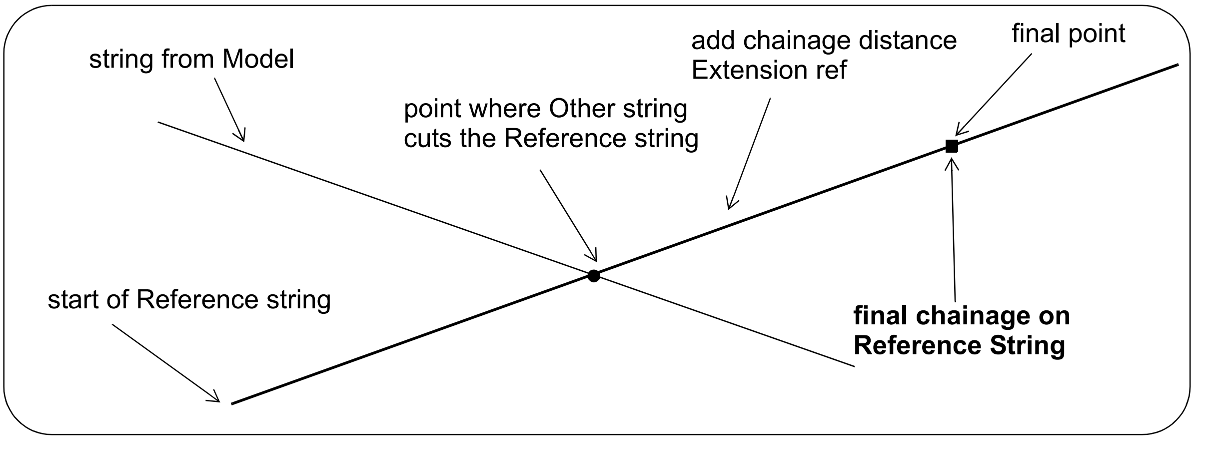
Wildcard text box

*if **not blank**, only those strings in the model* **Model** *that satisfy* **Wildcard** *are used.*
*If **blank**, all the strings in the model* **Model** *are used.*

For each string, the point on the Reference string where the string cuts the Reference string is first calculated.

The value in the **Extension ref** field is **added** to the reference chainage calculated for the cut point to give the chainage used as the ***final chainage***.

.



For more information on the inserting and use of snippets, see 21.5 Defining and Using Snippets.

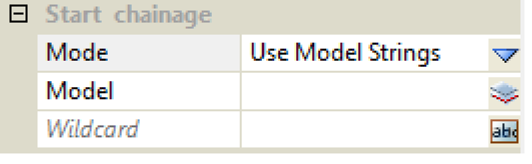
Continue to the next section 21.4.24 Snippet Placed to Model of Strings or return to 21.2.1 MTF Hinge Modifiers.

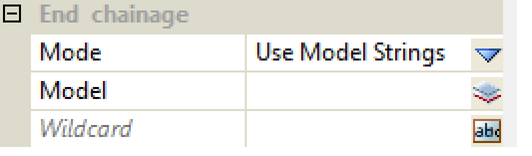
21.4.24 Snippet Placed to Model of Strings

The Smart Chainage **Snippet placed to model of strings** is only available with the **Snippet** option in the Apply MTF Left and Right Modifier.

This Mode takes a model of strings and the Snippet and for **each string** in the model, the end points of the string are dropped onto the Reference string and the Snippet is run with the **Start chainage** taken as the smaller of the two dropped chainages and the **End chainage** of the snippet taken as the larger of the two dropped chainages.

Selecting **Snippet cut ref with model of string** adds the **Model**, **Wildcard** and **Extension other** fields.





Model model box available models

the model of strings that the Snipped is applied to using the start and end of each string dropped onto the Reference string as the Start and End chainages for the Snippet.

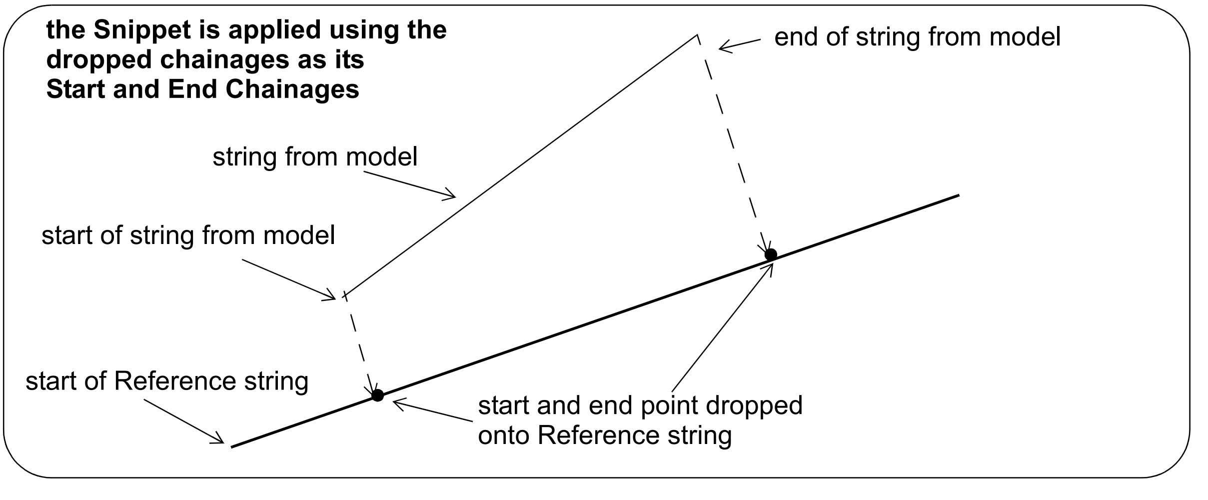
Wildcard text box

*if **not blank**, only those strings in the model* **Model** *that satisfy* **Wildcard** *are used.*

*If **blank**, all the strings in the model* **Model** *are used.*

For each string, the start and the end of the string are dropped onto the Reference string and the dropped chainages and used as the Start and End chainages of the Snippet

.

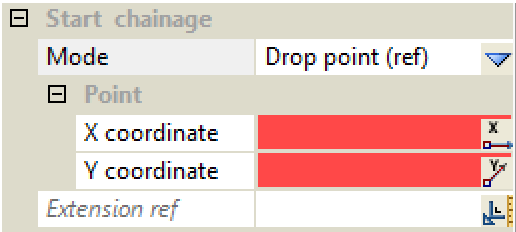


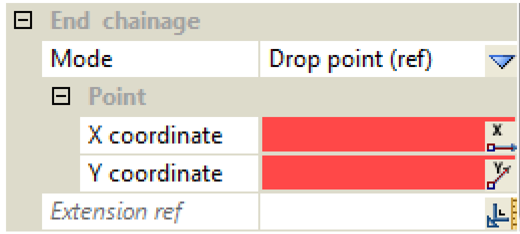
For more information on the inserting and use of snippets, see 21.5 Defining and Using Snippets.

Continue to the next section 21.4.25 Drop Point to Reference String or return to 21.2.1 MTF Hinge Modifiers.

21.4.25 Drop Point to Reference String

Selecting **Drop point to reference string** adds **X coordinate**, **Y coordinate** and **Extension ref** fields.





X coordinate and **Y coordinate** are the coordinate of the point to drop onto the Reference string to give a reference chainage.

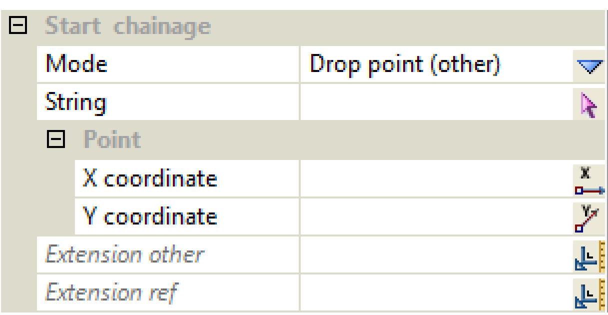
The value in **Extension ref** is added to the dropped chainage to give the chainage used as the ***final chainage***.

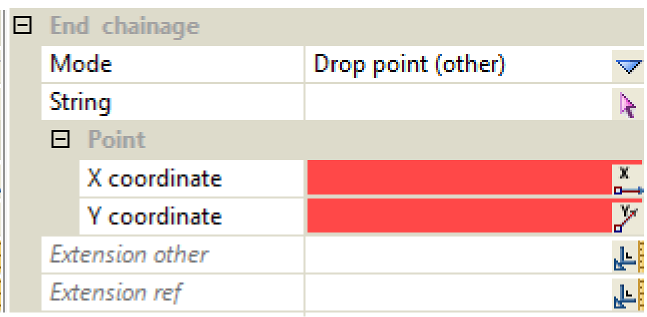
This Mode is useful when you know the coordinates of a point and you want the start/end chainage to be **relative** to the **chainage of the dropped point**.

Continue to the next section 21.4.26 Drop Point to Other String or return to 21.2.1 MTF Hinge Modifiers.

21.4.26 Drop Point to Other String

Selecting **Drop point to other string** adds **String**, **X coordinate**, **Y coordinate**, **Extension other** and **Extension ref** fields.





The **String** field is to select a string referred to as the **Other** string.

X coordinate and **Y coordinate** are the coordinate of the point to drop onto the **Other** string to give a chainage on the Other string.

The value in **Extension other** is added to the dropped chainage on the Other string and the point at this chainage is then dropped onto the Reference string.

The value in **Extension ref** is then added to the chainage of the point dropped onto the Reference string to give the chainage used as the ***final chainage***.

Return to 21.2.1 MTF Hinge Modifiers.

21.5 Defining and Using Snippets

The sections on snippets are:

21.5.1 What are MTF Snippets?

A snippet is a collection of many single MTF modifiers that can be inserted as a group into an MTF file.

However unlike an MTF file or an insert in an MTF, values to be used inside the snippet can be passed down to the snippet via parameters whose values are specified each time the snippet is placed.

So, snippets enable things such as driveway laybacks, kerb transitions, earthworks flares etc. to be a single entry in a MTF file thus reducing the size of the MTF file and enhancing it's readability.

Setting up a snippet is the same complexity as creating MTF items but once a snippet is created, it can be reused over and over again.

Snippets can also be used to hide complex modifiers from novice users.

Go to the next section 21.5.2 Creating a Snippet or back to 21.5 Defining and Using Snippets.

21.5.2 Creating a Snippet

There is currently no snippet Create/Editor in ***12d Model*** so for the moment snippets must be manually created as a text file. Snippet text files have the extension ***MTFSNIPPET***.

Note that this is case insensitive. That is, each letter can be in upper or lower case.

Luckily the format for each snippet command is almost identical to the normal modifier commands as written out to the left_side_modifier or right_side_modifier sections of the MTF file when viewed as a text file. The snippet commands differ slightly in that they can contain variables and parameters that can be substituted when the snippet is run.

So the easiest way to create a snippet file is to create a normal MTF inside ***12d Model*** with the modifiers to be included and used in the snippet. Once these modifier commands are in the MTF, they can be copied directly from the Modifiers grid in a snippet compatible format.

To do so, select one or more rows of modifiers in the grid. Then right-click on one of the row header to bring up the modifiers grid context menu and select **Copy (snippet paste)**. This will copy the selected commands to the Windows clipboard.

Now that the desired modifier commands have been copied to the clipboard, you can create a new text document in your preferred text editor and paste the commands into the new document. A snippet, at its simplest, can simply be these same modifier commands in a snippet file. However, the real power and flexibility of snippets comes from using the more advanced functionality discussed in the following sections.

It is also worth noting that the snippet commands can be created manually, rather than by copy and paste, by typing the required snippet and modifier commands directly into a text file. Note, however, that the syntax and format of the snippet and modifiers must be correct or you will get errors when the snippet is inserted or run.

How to construct the additional information in the snippet file, and how a snippet works, will be explained in the following sections.

Go to the next section 21.5.3 An Example of a Snippet or back to 21.5 Defining and Using Snippets.

21.5.3 An Example of a Snippet

The following is an example of a snippet called KERB_SA_DW places a layback at a driveway. It will be used as the example in describing the workings of a snippet.

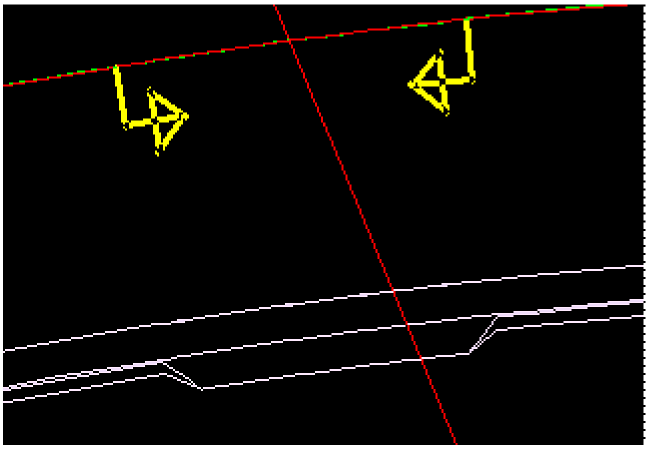
```
// PARAMETER WIDTH REAL "Layback width" 0.6
// PARAMETER DEPTH REAL "Layback depth" -0.04

// drop into DW

insert "SAL" "grey" 0.001 0    unknown named_position "MODIFIER_START" 0 named_position
 "MODIFIER_START" 0.5 absolute extra_start extra_end
insert "SAI" "grey" 0.5  $DEPTH unknown named_position "MODIFIER_START" 0 named_position
"MODIFIER_START" 0.5 absolute extra_start extra_end
insert "SAT" "grey" 0.03  0.15 unknown named_position "MODIFIER_START" 0 named_position
"MODIFIER_START" 0.5 absolute extra_start extra_end
insert "SAB" "grey" 0.18  0    unknown named_position "MODIFIER_START" 0 named_position
"MODIFIER_START" 0.5 absolute extra_start extra_end

// push out links
width  "SAT" named_position "MODIFIER_START" 0 named_position "MODIFIER_START" 0.5 0.03 $WIDTH
absolute extra_start extra_end
width  "SAB" named_position "MODIFIER_START" 0 named_position "MODIFIER_START" 0.5 0.18 0.0    absolute
extra_start extra_end
height "SAT" named_position "MODIFIER_START" 0 named_position "MODIFIER_START" 0.5 0.15 0.2    absolute
extra_start extra_end
height "SAB" named_position "MODIFIER_START" 0 named_position "MODIFIER_START" 0.5 0    0      absolute
extra_start extra_end

// DW to -0.5
insert "SKL" "grey" 0.001 0    unknown named_position "MODIFIER_START" 0.5 named_position
"MODIFIER_END" -0.5 absolute extra_start extra_end
insert "SKI" "grey" 0.5  $DEPTH unknown named_position "MODIFIER_START" 0.5 named_position
"MODIFIER_END" -0.5 absolute extra_start extra_end
insert "SKB" "grey" $WIDTH  0.2   unknown named_position "MODIFIER_START" 0.5 named_position
"MODIFIER_END" -0.5 absolute extra_start extra_end

//back out -0.5 to end
insert "SAL" "grey" 0.001 0    unknown named_position "MODIFIER_END" -0.5 named_position "MODIFIER_END"
0.0 absolute extra_start extra_end
insert "SAI" "grey" 0.5  $DEPTH unknown named_position "MODIFIER_END" -0.5 named_position
"MODIFIER_END" 0.0 absolute extra_start extra_end
insert "SAT" "grey" 0.03  0.15 unknown named_position "MODIFIER_END" -0.5 named_position
"MODIFIER_END" 0.0 absolute extra_start extra_end
insert "SAB" "grey" 0.18  0    unknown named_position "MODIFIER_END" -0.5 named_position "MODIFIER_END"
0.0 absolute extra_start extra_end

// push out links
width  "SAT" named_position "MODIFIER_END" -0.5 named_position "MODIFIER_END" 0.0 $WIDTH 0.03
```

absolute extra_start extra_end

width "SAB" named_position "MODIFIER_END" -0.5 named_position "MODIFIER_END" 0.0 0.0 0.18 absolute extra_start extra_end

height "SAT" named_position "MODIFIER_END" -0.5 named_position "MODIFIER_END" 0.0 0.2 0.15 absolute extra_start extra_end

height "SAB" named_position "MODIFIER_END" -0.5 named_position "MODIFIER_END" 0.0 0   0 absolute extra_start extra_end



Note: each line of the snippet is wrapping around because of the limited width of this document. Each line should be like:

```
// drop into DW
insert "SAL" "grey" 0.001 0  unknown named_position "MODIFIER_START" 0 named_position  "MODIFIER_START" 0.5 absolute extra_start extra_end
```

But that would be impossible to read.

Go to the next section 21.5.4 Editing a Snippet File or back to 21.5 Defining and Using Snippets.

21.5.4 Editing a Snippet File

The option

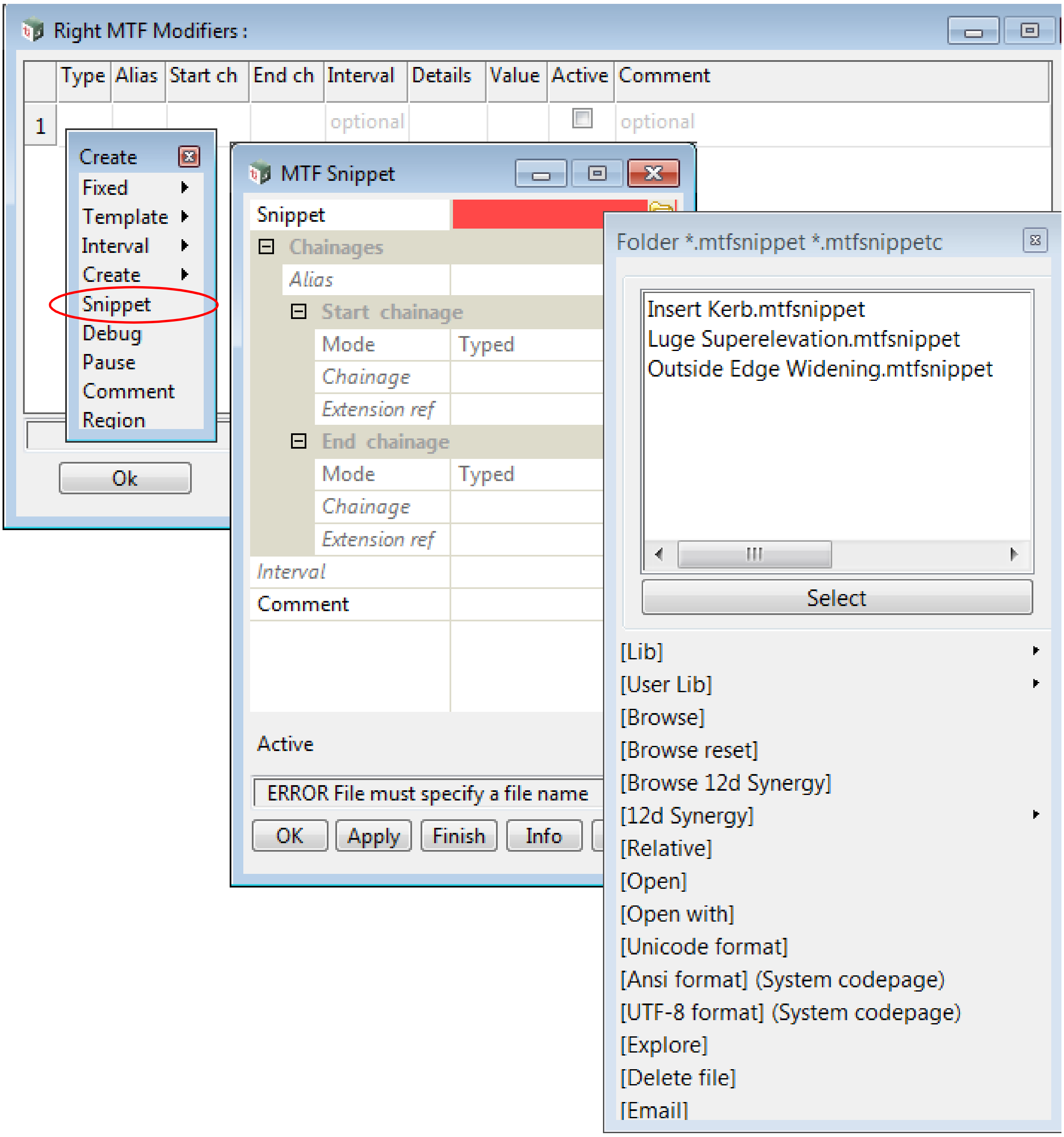
Design =>MTF =>Edit snippets

will list the local snippets files and if one is selected, opens it in the text editor.

Go to the next section 21.5.5 How To Select a Snippet in an MTF or back to 21.5 Defining and Using Snippets.

21.5.5 How To Select a Snippet in an MTF

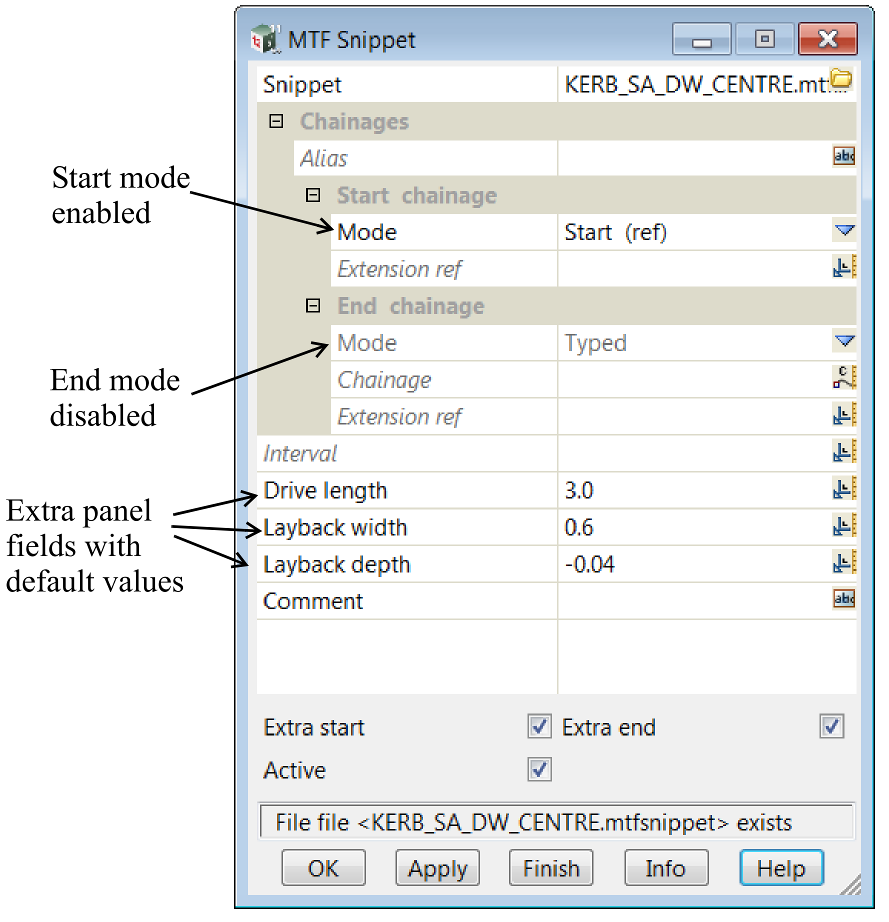
From the grid in the **Right MTF Modifiers** or **Left MTF Modifiers** panel, bring up the **Create** menu and select **Snippet**. This will display the **MTF Snippet** panel and clicking on the folder icon in the *Snippet* field brings up the choice of snippets to use.



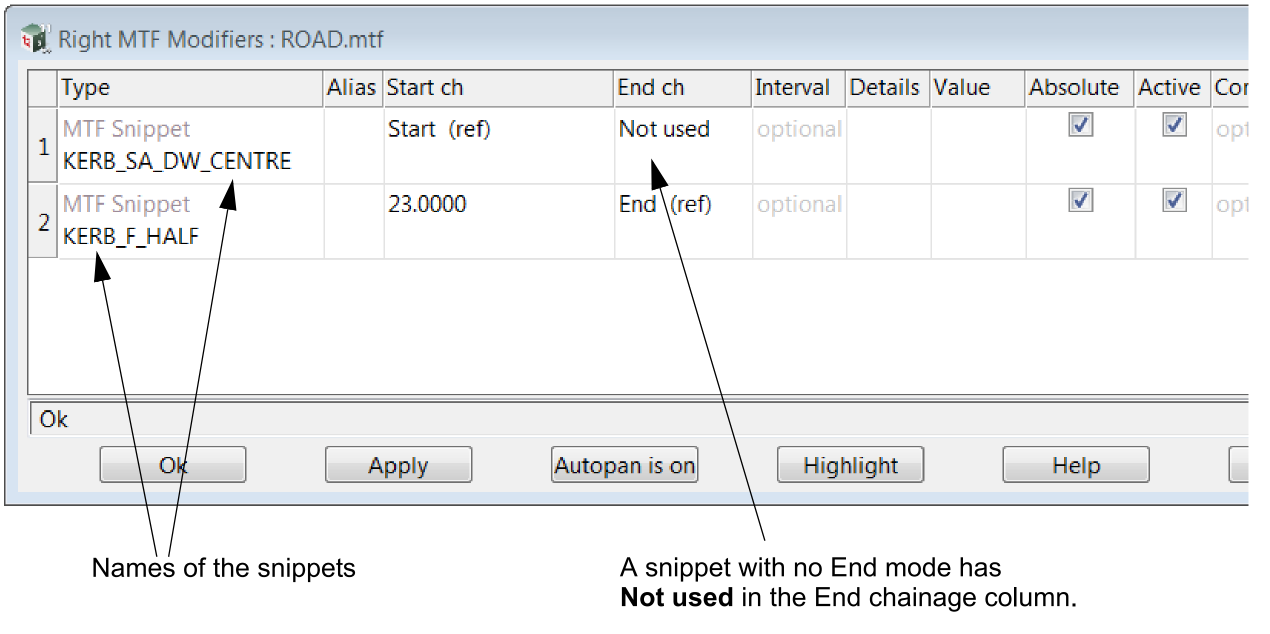
A snippet is then selected from the snippet pop-up list.

Selecting a snippet can modify the fields on the **MTF Snippet** panel. The ***Start*** and ***End modes*** may or may not be enabled, and there may be extra field on the panel, with or without values in them.

For example, selecting the *KERB_SA_DW* snippet modifies the **MTF Snipped** panel and there are two extra panel fields - ***Layback width*** with the value 0.6 and ***Layback height*** with the value -0.4. Selecting the *KERB_SA_DW_*CENT snippet adds the extra panel fields but also disables the ***End mode.***



If a Start mode or End mode is missing from the snippet, then the corresponding field will be left blank in the ***Start chainage/End chainage*** columns in the **Left** or **Right Templates Modifiers** panel. And the snippet name appears in the ***Type*** column under the text **MTF Snippet**.



How snippets control the ***Start*** and ***End*** modes is documented in the next section 21.5.7 Start and End Modes for a Snippet and how the extra panel fields are created is discussed in the sections 21.5.8.2 Snippet Parameter of Type REAL, 21.5.8.6 Snippet Parameter of Type CHOICE and 21.5.8.8 Snippet Parameter of Type TICK.

Go to the next section 21.5.6 Comments in Snippets or back to 21.5 Defining and Using Snippets.

21.5.6 Comments in Snippets

Snippets support the same commenting syntax as MTF files, which has two forms:

1. **A Line Comment**

   All characters from a double forward-slash **//** until the end of that line are ignored.

   However, if the first word after the line comment character is PARAMETER, INFO or DISPLAY then the line will be interpreted as a special command by 12d.

```
// PARAMETER this will not be interpreted as a normal line comment
// This is a normal line comment
@ def_tok WIDTH 3.5     // This is also a line comment
```

2. **A Block Comment**

   All characters between a starting /* and a terminating */ are ignored.

```
/* This comment can
 cover several lines
 until we get to the terminating */
```

Continue to the next section 21.5.7 Start and End Modes for a Snippet or return to 21.5 Defining and Using Snippets.

21.5.7 Start and End Modes for a Snippet

The ***Start*** and ***End*** modes on the **MTF Snippet** panel calculate a position on the Reference string just like any other *MTF modifier* but how the values are specified inside the snippet file is slightly different to a standard MTF modifier.

Also unlike other MTF modifiers, some snippets only require *one* of the ***Start mode*** and ***End mode*** and if that is the case, the mode not required will be greyed out so that it can't be filled in.

The lines in the snippet file are similar to the normal MTF modifiers when "named position" has been selected for the ***Start*** and ***End*** modes. However the value of the "named position" in the text command needs to be replaced by either MODIFIER_START or MODIFIER_END followed by a real number.

So for example, the "named_position" "name" "extension" in a MTF command is replaced by

"named_position" MODIFIER_START   snippet_extension

If a MODIFIER_START exists somewhere in the snippet then ***Start mode*** is enabled on the **MTF Snipped** panel.

If a MODIFIER_END exists somewhere in the snippet then ***End mode*** is enabled on the **MTF Snipped** panel.

Finally when a snippet runs, the chainage is calculated for the ***Start mode***/***End mode*** fields in the **MTF Snippet** panel and passed down to the snippet and where ever MODIFIER_START/ MODIFIER_END appears, the passed chainage value PLUS the additional snippet_extension for each occurrence of the MODIFIER_START/MODIFIER_END is used as the chainage for that part of the snippet.

The following example for the KERB_SA_DW snippet shows how the snippet determines when a ***Start mode*** and/or ***End mode*** is needed., and how the extra snippet_extension is used.

```
// drop into DW
insert "SAL" "grey" 0.001 0    unknown named_position "MODIFIER_START" 0 named_position
"MODIFIER_START" 0.5 absolute extra_start extra_end
...
// DW to -0.5
insert "SKL" "grey" 0.001 0    unknown named_position "MODIFIER_START" 0.5 named_position
"MODIFIER_END" -0.5 absolute extra_start extra_end
...
// back out to -0.5 to end
insert "SAL" "grey" 0.001 0    unknown named_position "MODIFIER_END" -0.5 named_position
"MODIFIER_END" 0.0 absolute extra_start extra_end
```

These lines in the KERB_SA_DW snippet a lip of kerb through a driveway layback. It starts off as an SA style kerb, the layback is an SK style and then back to the SA style.

1. Looking at the first line: - the SA->SK transition:

insert "SAL" "grey" 0.001 0   unknown named_position "MODIFIER_START" 0 named_position "MODIFIER_START" 0.5 absolute extra_start extra_end

In this line, the ***"named_position" "MODIFIER_START" 0*** replaces the standard chainage command in that position in the text modifier.

"named position" "MODIFIER_START" - this tells the snippet to have a ***Start mode*** on the **MTF Snippet** panel. The calculated chainage for the ***Start mode*** field on the **MTF Snippet** panel is then passed to the snippet to be used as the chainage (plus the snippet_extension) in that *"named_position" MODIFIER_START"* position in the modifier command.

0 - the value after MODIFIER_START is an additional value to add to the chainage calculated in from the ***Start mode*** in the **MTF Snippet** panel and passed down to the snippet.

So ***"named_position" "MODIFIER_START" 0*** is replaced by the chainage defined by the ***Start mode*** of the **MTF Snippet** panel.

Similarly the ***"named_position" "MODIFIER_START" 0.5*** replaces the standard chainage command in that position in the modifier and evaluates to being chainage defined by the ***Start mode*** of the **MTF Snippet** panel **plus** 0.5.

So this snippet line inserts a link from the chainage value passed in from the ***Start mode*** on the **MTF Snippet** panel, to 0.5 **after** the chainage value passed in from the ***Start mode*** on the **MTF Snippet** panel.

2. Looking at the second line: - SK Layback:

insert "SKL" "grey" 0.001 0 unknown named_position "MODIFIER_START" 0.5 named_position "MODIFIER_END" -0.5 absolute extra_start extra_end

The difference to the previous line is that this line has the keywords ***MODIFIER_START*** and ***MODIFIER_END*** after "named_position" commands.

"named position" "MODIFIER_END" - this tells the snippet to have a ***End mode*** on the **MTF Snippet** panel. The calculated chainage for the ***End mode*** field on the **MTF Snippet** panel is then passed to the snippet to be used as the chainage (plus the snippet_extension) in that *"named_position" MODIFIER_END"* position in the modifier command.

-0.5 - the value after MODIFIER_END is an additional value to add to the chainage calculated in from the ***End mode*** in the **MTF Snippet** panel and passed down to the snippet.

So this snippet line inserts a link from 0.5 past the chainage value passed from ***Start mode*** in the **MTF Snippet** panel through to 0.5 ***before*** the chainage value passed in from ***End mode*** in the **MTF Snippet** panel.

3. Looking at the third line: - SK ->SA transition:

insert "SAL" "grey" 0.001 0 unknown named_position "MODIFIER_END" -0.5 named_position "MODIFIER_END" 0.0 absolute extra_start extra_end

So this snippet line inserts a link from 0.5 before the chainage value passed to the snippet from the ***Start mode*** in the **MTF Snippet** panel, to the chainage value passed to the snippet from the ***Start mode*** in the **MTF Snippet** panel.

Go to the next section 21.5.8 Snippet Parameters and Commands or back to 21.5 Defining and Using Snippets.

21.5.8 Snippet Parameters and Commands

A powerful feature of snippets is that parameters (fields presented to and populated by the user) can be defined and used throughout the snippet file.

To pass values for the parameters through to the snippet, each parameter is displayed as the appropriate panel Field in the **MTF Snippet** panel, with the label (descriptive text) for the panel field defined in the snippet. For example, a Real box panel field for a real parameter and a Model box panel field for a model parameter.

The user enters the value into the field on the **MTF Snippet** panel. This entered value is then passed through to the snippet during processing.

Note - Even if a parameter is defined in a snippet, it will only appear in the MTF Snippet panel if the parameter is actually used in the snippet.

WARNING - once a snippet with parameters has been saved in a MTF file, the parameters for the snippet are saved ***in the MTF file.*** Then when the Apply MTF is run, the parameters and their values are read in from the MTF file and passed into the current snippet file.

Hence if the snippet file has been modified and parameter definitions added, modified or removed since the snippet was originally place in the MTF, these modifications of parameters will be **ignored** until the snippet panel is reopened in the MTF editor which forces the snippet file to be parsed again and the changes to the parameter definitions are then recognised.

For the syntax of each available parameter types and commands, see

21.5.8.2 Snippet Parameter of Type REAL
21.5.8.3 Snippet Parameter of Type INTEGER
21.5.8.4 Snippet Parameter of Type TEXT
21.5.8.5 Snippet Parameter of Type SELECT
21.5.8.6 Snippet Parameter of Type CHOICE
21.5.8.7 Snippet Parameter of Type CHOICE2
21.5.8.8 Snippet Parameter of Type TICK
21.5.8.9 Snippet Parameter of Type COLOUR
21.5.8.10 Snippet Parameter of Type NAME
21.5.8.11 Snippet Parameters for Models
21.5.8.12 Snippet Parameters for Tins
21.5.8.13 Snippet Parameter of Type LAYER
21.5.8.14 Snippet Parameter of Type NAMED_GRADE
21.5.8.15 Snippet Command INFO
21.5.8.16 Snippet Command DISPLAY
21.5.8.17 Snippet Command INCREMENT
21.5.8.18 Optional Parameters in Snippets

21.5.8.1 Common Definitions for Snippet Parameters

The general syntax for a snippet parameter is:

// **PARAMETER param_name param_type param_desc param_default other_values**

where

// PARAMETER param_name

This signifies that it is a parameter definition. Note that there must be at least one space after the // and before the word PARAMETER.

param_name

This is the name to use for the parameter. Where ever the parameter is to be used in the snippet, put **$(param_name)**. The characters of the parameter name can only be alphanumeric (upper and lower case) and underscores, and the name cannot contain spaces. The name doesn't have quotes around it. The parameter names are not case sensitive so two names only differing by case are considered to be identical. The parameters names must be unique in a snippet.

param_descr param_default

param_desc

This description is written in the **MTF Snippet** panel for the field to enter values for this parameter. The description can include spaces and if it does then the description must be enclose in double quotes (").

param_default

This value is **optional**.

But if it exists then this value is displayed as the value in the field for the parameter on the **MTF Snippet** panel. If the default value is text and it contains any spaces then the text must be enclosed in double quotes (").

Important Note:

There will only be a panel field in the **MTF Snippet** panel for the parameter **param_name** if the parameter is **used** in the snippet.

21.5.8.2 Snippet Parameter of Type REAL

If a floating point number (a real number) is required then the **REAL** parameter type is used.

A Snippet Parameter of type **REAL** inserts a Real box panel field into the **MTF Snippet** panel.

The user can type a real number into the panel field which is then passed down to the snippet.

The syntax for type **REAL** is:

// PARAMETER param_name **REAL** param_descr param_default_value

where

// PARAMETER param_name

This defines a parameter of the given name. See // PARAMETER param_name for what is allowed as the parameter name.

REAL - the parameter type.

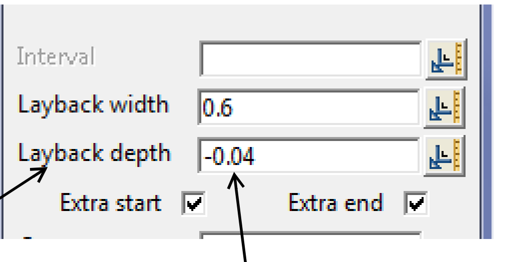
param_desc param_default

The description for the field in the **MTF Snippet** panel and an optional default value that is entered as the value in the field. See param_descr param_default.

The following lines from the KERB_SA_DW snippet define two REAL parameters WIDTH and DEPTH and the values for the parameters are then used inside the snippet commands by writing $WIDTH and $DEPTH.

```
// PARAMETER WIDTH REAL "Layback width" 0.6
// PARAMETER DEPTH REAL "Layback depth" -0.04
// drop into DW
...
// push out links
width "SAT" named_position "MODIFIER_START" 0 named_position "MODIFIER_START" 0.5
0.03 $(WIDTH) absolute extra_start extra_end
...
```

Real panels field are in MTF Snippet panel because of the parameters WIDTH and DEPTH in the snippet



Description provided by the parameter definition in the snippet

Default value provided by the parameter definition in the snippet

Important Note:

The Real panel field for a REAL parameter will only appear in the **MTF Snippet** panel if the parameter is used in the snippet.

Continue to the next section 21.5.8.3 Snippet Parameter of Type INTEGER or return to 21.5.8 Snippet Parameters and Commands or 21.5 Defining and Using Snippets.

21.5.8.3 Snippet Parameter of Type INTEGER

If a whole number (an integer) is required then the **INTEGER** parameter type is used.

A Snippet Parameter of type **INTEGER** inserts an Integer box panel field into the **MTF Snippet** panel.

The user can type an integer into the panel field and it is then passed down to the snippet.

The syntax for type **INTEGER** is:

// PARAMETER param_name **INTEGER** param_desc param_default

where

// PARAMETER param_name

This defines a parameter of the given name. See // PARAMETER param_name for what is allowed as the parameter name.

INTEGER - the parameter type.

param_desc param_default

The description for the field in the **MTF Snippet** panel and an optional default value that is entered as the value in the field. See param_descr param_default.

Important Note:
The Integer panel field for a INTEGER parameter will only appear in the **MTF Snippet** panel if the parameter is used in the snippet.

Continue to the next section 21.5.8.4 Snippet Parameter of Type TEXT or return to 21.5.8 Snippet Parameters and Commands or 21.5 Defining and Using Snippets.

21.5.8.4 Snippet Parameter of Type TEXT

If text is required then the **TEXT** parameter type is used.

A Snippet Parameter of type **TEXT** inserts an Text box panel field into the **MTF Snippet** panel.

The user can type text into the panel field and it is then passed down to the snippet.

The syntax for type **TEXT** is:

// PARAMETER param_name **TEXT** param_desc param_default

where

// PARAMETER param_name

This defines a parameter of the given name. See // PARAMETER param_name for what is allowed as the parameter name.

TEXT - the parameter type.

param_desc param_default

The description for the field in the **MTF Snippet** panel and an optional default value that is entered as the value in the field. See param_descr param_default.

Note for the TEXT type, the $parameter should be enclosed in the double quotes character in the modifier line.

Important Note:
The Text panel field for a TEXT parameter will only appear in the **MTF Snippet** panel if the parameter is used in the snippet.

Continue to the next section 21.5.8.5 Snippet Parameter of Type SELECT or return to 21.5.8 Snippet Parameters and Commands or 21.5 Defining and Using Snippets.

21.5.8.5 Snippet Parameter of Type SELECT

If a string is to be selected then the **SELECT** parameter type is used.

A Snippet Parameter of type **SELECT** inserts a String Select box panel field into the **MTF Snippet** panel.

Using the Select box, a user can pick a string and it is then passed down to the snippet.

The syntax for type **SELECT** is:

// PARAMETER param_name **SELECT** param_desc param_default

where

// PARAMETER param_name

This defines a parameter of the given name. See // PARAMETER param_name for what is allowed as the parameter name.

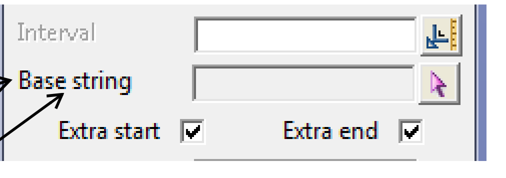
SELECT - the parameter type.

param_desc param_default

The description for the field in the **MTF Snippet** panel and an optional default value that is entered as the value in the field. See param_descr param_default.

```
// PARAMETER BB_STRING SELECT "Base string"

...

// parallel the selected string

fixed_parallel "SFML" "SFMB" named_position "MODIFIER_START" 0 named_position
"MODIFIER_FINAL" 0 "$BB_STRING" interval 0.5 absolute extra_start extra_end

...
```

String select panel field is in MTF Snippet panel because of the SELECT parameter BB_STRING in the snippet



Description provided by the parameter definition in the snippet

Here the parameter name is ***BB_STRING***, its type is ***SELECT*** and so a **Select** box appears in the **MTF Snippet** panel with the panel field text ***Base string***. The substitution point in the modifier is ***"$BB_STRING"***.

Note for the SELECT type, the $parameter should be quoted in the modifier line.

Important Note:
The String Select panel field for a SELECT parameter will only appear in the **MTF Snippet** panel if the parameter is used in the snippet.

Continue to the next section 21.5.8.6 Snippet Parameter of Type CHOICE or return to 21.5.8 Snippet Parameters and Commands or 21.5 Defining and Using Snippets.

21.5.8.6 Snippet Parameter of Type CHOICE

If the user is to select from a list of choices and the text of the choice is to be passed through to the snippet, then the **CHOICE** parameter type is used.

The Snippet Parameter of type **CHOICE** inserts a Choice box panel field into the **MTF Snippet** panel.

The format also defines the list of text to be displayed in the pop-up of choices and when a choice is selected, the text that is displayed is passed back to the snippet.

The syntax for type **CHOICE** is:

// PARAMETER param_name CHOICE param_desc choice_1 choice_2 ... choice_n

where

// PARAMETER param_name

This defines a parameter of the given name. See // PARAMETER param_name for what is allowed as the parameter name.

CHOICE - the parameter type.

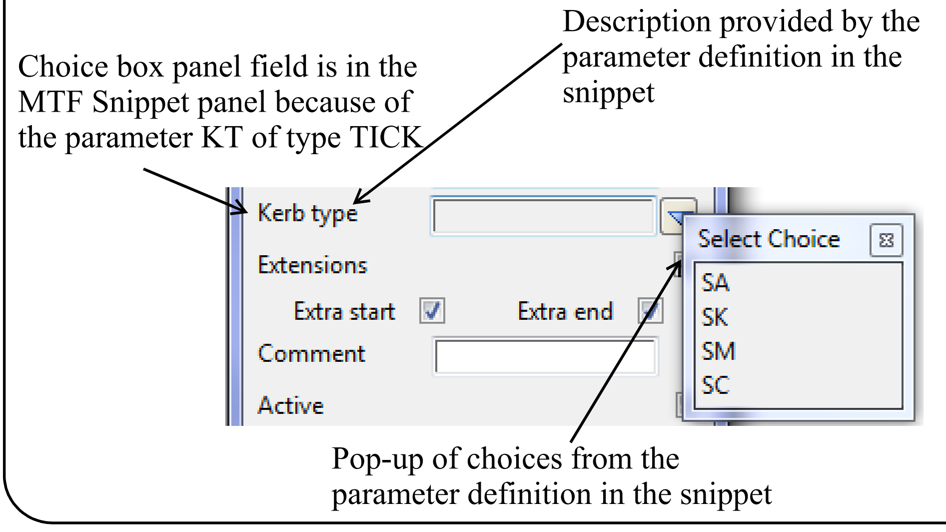
param_descr - this description is written to the **MTF Snippet** panel for the field to enter values for this parameter. The description can include spaces and if it does then the description must be enclose in double quotes (").

choice_1 choice_2 ... choice_n - the choices that will be displayed in the parameter pop-up in the **MTF Snippet** panel.

// PARAMETER KT **CHOICE** "Kerb type" "SA" "SK" "SM" "SC"

insert "$KT" "grey" 10 0   unknown named_position "MODIFIER_START" (0.0 - _E) named_position "MODIFIER_END" _E absolute extra_start extra_end

...



Important Note:
The Choice panel field for a CHOICE parameter will only appear in the **MTF Snippet** panel if the parameter is used in the snippet.

Continue to the next section 21.5.8.7 Snippet Parameter of Type CHOICE2 or return to 21.5.8 Snippet Parameters and Commands or 21.5 Defining and Using Snippets.

21.5.8.7 Snippet Parameter of Type CHOICE2

If the user is to select from a list of choices, but different text than what appears on the choice list is to be passed through to the snippet, then the **CHOICE2** parameter type is used.

The Snippet Parameter of type **CHOICE2** inserts a Choice box panel field into the **MTF Snippet** panel.

The format also defines the list of text to be displayed in the pop-up of choices for each text that is displayed, a parameter value that is passed back to the snippet if that choice is selected.

The syntax for type **CHOICE2** is:

// PARAMETER par_name **CHOICE2** par_desc par_def par_1 desc_1 par_2 desc_2 **...** par_n desc_n

where

// PARAMETER param_name

This defines a parameter of the given name. See // PARAMETER param_name for what is allowed as the parameter name.

CHOICE2 - the parameter type.

par_desc - this description is written in the **MTF Snippet** panel for the field to enter values for this parameter. The description can include spaces and if it does then the description must be enclose in double quotes (").

par_def - this value is **optional** but if it exists then this value is displayed as the value in the field for the parameter on the **MTF Snippet** panel. If the default value is text and it contains any spaces then the text must be enclosed in double quotes (").

par_i desc_i are pairs of parameters and description to go in the Choice box pop up. The description ***desc_i*** is displayed in the Choice pop up and if that description is selected, the parameter ***par_i*** is returned to the Snippet instead of the description.

For example

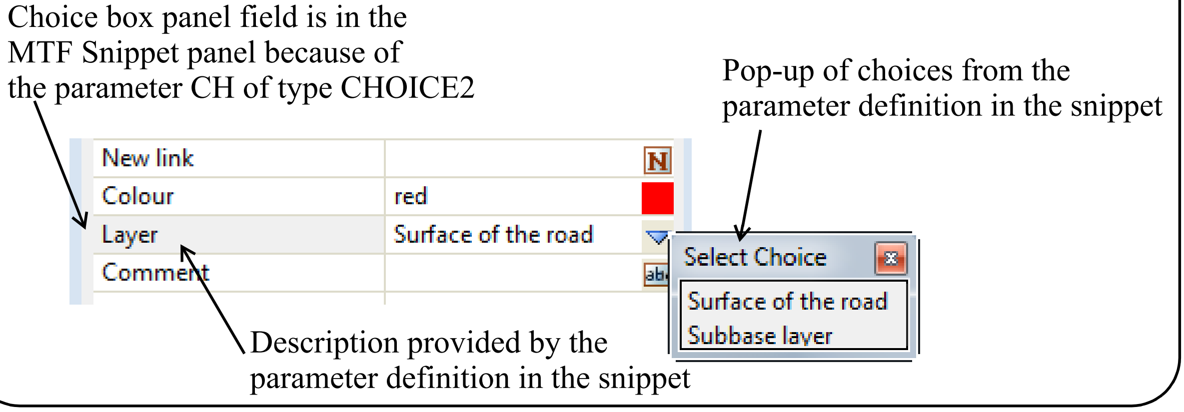
// PARAMETER CH CHOICE2 "Speed" S100 S80 "80 km/h" S100 "100 km/h"

This means that the text you have the default parameter, in this case **S100**, and then pairs of parameters and descriptions, **S80 "80 km/h"** etc.

// PARAMETER CH **CHOICE2** "Layer" "Design" "Design" "Surface of the road" "Subbase" "Subbase layer"

insert $CH=>$LINK" "$CLR" 0.001 0.20 unknown SCH 0.0 ECH 0.0 absolute extra_start extra_end

...



Important Note:

The Choice panel field for a CHOICE2 parameter will only appear in the **MTF Snippet** panel if the parameter is used in the snippet.

Continue to the next section 21.5.8.8 Snippet Parameter of Type TICK or return to 21.5.8 Snippet Parameters and Commands or 21.5 Defining and Using Snippets.

21.5.8.8 Snippet Parameter of Type TICK

If a true of false choice is required then the **TICK** parameter type is used.

The Snippet Parameter of type **TICK** inserts a Tick box panel field into the **MTF Snippet** panel. The user can set the tick to off or on.

The syntax for type TICK is:

// PARAMETER param_name **TICK** param_desc   param_default

where

// PARAMETER param_name

This defines a parameter of the given name. See // PARAMETER param_name for what is allowed as the parameter name.

TICK - the parameter type.

param_desc param_default

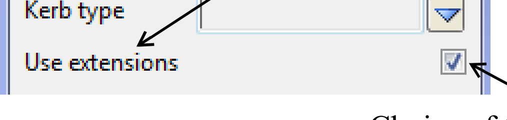
The description for the field in the **MTF Snippet** panel and an optional default value that is entered as the value in the field. See param_descr param_default.

The value **param_default** is **optional** but if it exists then it must be 1 or a 0.
If 1, the tick box come up set to tick/on.
If 0, the tick box come up set to not ticked/off.

```
// PARAMETER  KT  CHOICE   "Kerb type"  "SA"  "SK"  "SM"   "SC"
// PARAMETER  TI   TICK   "Use extensions"    1
#if $TI
#define _E 3
#else
#define _E 0
#endif
 insert "$KT" "grey" 10 0   unknown named_position "MODIFIER_START" (0.0 - _E)
named_position "MODIFIER_END" _E absolute extra_start extra_end

...
```

Tick box panel field is in the MTF Snippet panel because of the parameter TI of type TICK

Description provided by the parameter definition in the snippet



Choice of tick and not tick because the parameter type is TICK

Note that in the above example, the value of the Tick box parameter TI ($TI), is used in a #define to set the value for **_E** (see 21.5.18 Major Warning - You Will be Caught by This).

Important Note:
The Tick panel field for a TICK parameter will only appear in the **MTF Snippet** panel if the parameter is used in the snippet.

Continue to the next section 21.5.8.9 Snippet Parameter of Type COLOUR or return to 21.5.8 Snippet Parameters and Commands or 21.5 Defining and Using Snippets.

21.5.8.9 Snippet Parameter of Type COLOUR

If a colour is required then the **COLOUR** parameter type is used.

The Snippet Parameter of type **COLOUR** inserts a Colour box panel field into the **MTF Snippet** panel.

The user can type a colour name or colour number into the panel field, or select a colour from the pop-up of available colours in the project.

The syntax for type **COLOUR** is:

// PARAMETER   param_name   **COLOUR**   param_desc   param_default

where

// PARAMETER   param_name

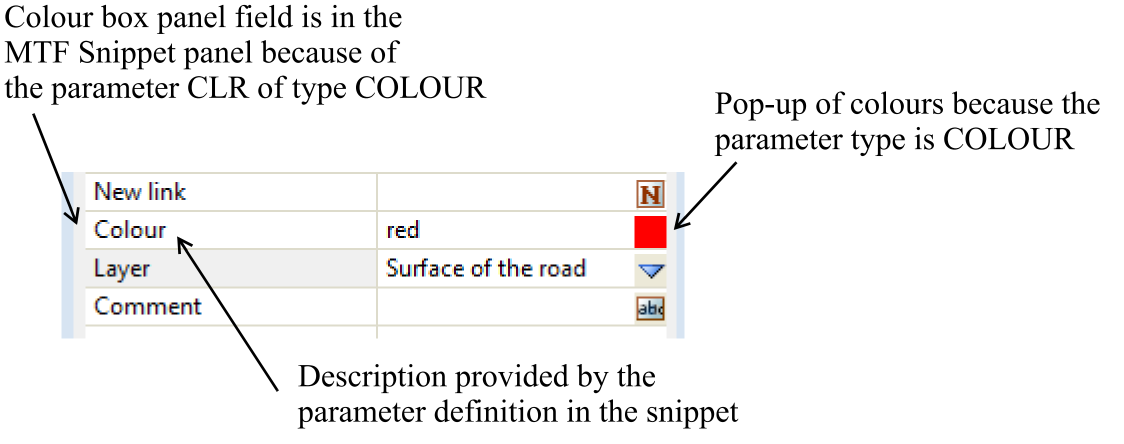
This defines a parameter of the given name. See // PARAMETER param_name for what is allowed as the parameter name.

COLOUR - the parameter type.

param_desc   param_default

The description for the field in the **MTF Snippet** panel and an optional default value that is entered as the value in the field. See param_descr   param_default.

```
// PARAMETER  CLR  COLOUR "Colour" "red"
insert "$INS_LYR=>$LINK"  "$CLR" 0.001  0.20 unknown SCH 0.0 ECH 0.0 absolute extra_start extra_end
...
```



Important Note:
The Colour panel field for a COLOUR parameter will only appear in the **MTF Snippet** panel if the parameter is used in the snippet.

Continue to the next section 21.5.8.10 Snippet Parameter of Type NAME or return to 21.5.8 Snippet Parameters and Commands or 21.5 Defining and Using Snippets.

21.5.8.10 Snippet Parameter of Type NAME

If a name of a string is required then the **NAME** parameter type is used.

The Snippet Parameter of type **NAME** inserts a Name box panel field into the **MTF Snippet** panel.

The user can type a string name into the panel field, or select a name from the pop-up of names that are taken from the names.4d file.

The syntax for type **NAME** is:

// PARAMETER param_name **NAME** param_descr param_default

where

// PARAMETER param_name

This defines a parameter of the given name. See // PARAMETER param_name for what is allowed as the parameter name.

NAME - the parameter type.

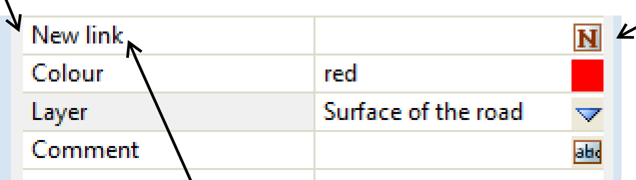
param_desc param_default

The description for the field in the **MTF Snippet** panel and an optional default value that is entered as the value in the field. See param_descr param_default.

```
// PARAMETER  LINK  NAME  "New link"
// PARAMETER  CLR  COLOUR "Colour" "red"
insert "$INS_LYR=>$LINK"  "$CLR" 0.001  0.20 unknown SCH 0.0 ECH 0.0 absolute extra_start extra_end
...
```

Name box panel field is in the MTF Snippet panel because of the parameter LINK of type NAME

Pop-up of names because the parameter type is NAME



Description provided by the parameter definition in the snippet

Important Note:
The String Name panel field for a NAME parameter will only appear in the **MTF Snippet** panel if the parameter is used in the snippet.

Continue to the next section 21.5.8.11 Snippet Parameters for Models or return to 21.5.8 Snippet Parameters and Commands or 21.5 Defining and Using Snippets.

21.5.8.11 Snippet Parameters for Models

There are three different Snippet parameter types for creating a Model box in the **MTF Snippet** panel. Which parameter is used depends on what is required to happen if the model exists or does not exit.

(a) use MODEL_CREATE when it doesn't matter if the selected model exists or doesn't exist.

See 21.5.8.11.1 Snippet Parameters of Type MODEL_CREATE

(b) use MODEL_MUST EXIST when the selected model must already exist.

See 21.5.8.11.2 Snippet Parameter of Type MODEL_MUST_EXIST

(c) use MODEL_MUST_NOT_EXIST when the model must **not** already exit.

21.5.8.11.3 Snippet Parameter Type MODEL_MUST_NOT_EXIST

Or return to 21.5.8 Snippet Parameters and Commands or 21.5 Defining and Using Snippets.

21.5.8.11.1 Snippet Parameters of Type MODEL_CREATE

The Snippet Parameter of type **MODEL_CREATE** inserts a Model box panel field into the **MTF Snippet** panel.

The user can type a model name into the panel field, or select a model from the pop-up of models for the Model box.

If the model does not exist when the snippet is run, then a new model is created.

The syntax for type **MODEL_CREATE** is:

// PARAMETER param_name **MODEL_CREATE** param_desc param_default

where

// PARAMETER param_name

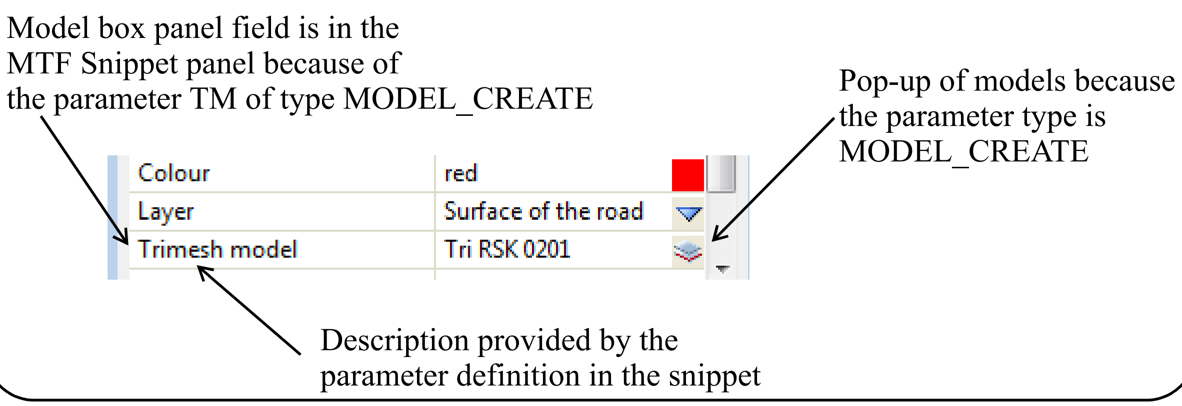
This defines a parameter of the given name. See // PARAMETER param_name for what is allowed as the parameter name.

MODEL_CREATE - the parameter type.

param_desc param_default

The description for the field in the **MTF Snippet** panel and an optional default value that is entered as the value in the field. See param_descr param_default.

```
// PARAMETER TM MODEL_CREATE "Trimesh model" "Tri RSK 0201"

create_strings "selection" "KSA<<LK" "KSA<<IK" "KSA<<TK" "KSA<<BK" start_ref 0 final_ref 0
                  "no_colour"  ""   "$TM" absolute extra_start extra_end

  ...
```



Important Note:
The Model panel field for a MODEL_CREATE parameter will only appear in the **MTF Snippet** panel if the parameter is used in the snippet.

Continue to the next section 21.5.8.11.2 Snippet Parameter of Type MODEL_MUST_EXIST or return to 21.5.8.11 Snippet Parameters for Models or 21.5.8 Snippet Parameters and Commands or 21.5 Defining and Using Snippets.

21.5.8.11.2 Snippet Parameter of Type MODEL_MUST_EXIST

The Snippet Parameter of type **MODEL_MUST_EXIST** inserts a Model box panel field into the **MTF Snippet** panel.

The user can type a model name into the panel field, or select a model from the pop-up of models for the Model box.

If the model does **not** exist when the Snippet is run, an error will occur and a message written to the panel message area.

The syntax for type **MODEL_MUST_EXIST** is:

// PARAMETER param_name **MODEL_MUST_EXIST** param_desc param_default

where

// PARAMETER param_name

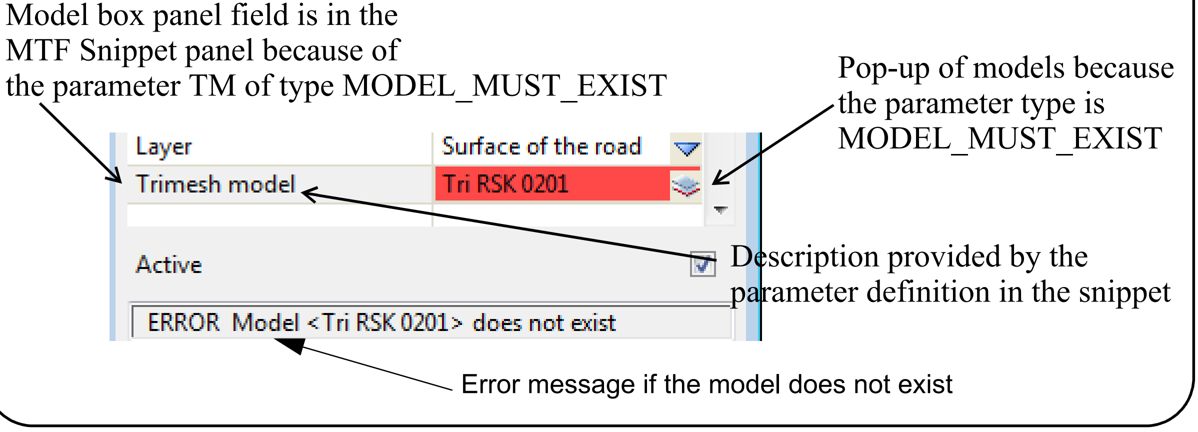
This defines a parameter of the given name. See // PARAMETER param_name for what is allowed as the parameter name.

MODEL_MUST_EXIST - the parameter type.

param_desc param_default

The description for the field in the **MTF Snippet** panel and an optional default value that is entered as the value in the field. See param_descr param_default.

```
// PARAMETER TM MODEL_MUST_EXIST "Trimesh model" "Tri RSK 0201"
create_strings "selection" "KSA<<LK" "KSA<<IK" "KSA<<TK" "KSA<<BK" start_ref 0 final_ref 0
          "no_colour"  ""     "$TM"    absolute   extra_start  extra_end
   ...
```



Important Note:
The Model panel field for a MODEL_MUST_EXIST parameter will only appear in the **MTF Snippet** panel if the parameter is used in the snippet.

Continue to the next section 21.5.8.11.3 Snippet Parameter Type MODEL_MUST_NOT_EXIST or return to 21.5.8.11 Snippet Parameters for Models or 21.5.8 Snippet Parameters and Commands or 21.5 Defining and Using Snippets.

21.5.8.11.3 Snippet Parameter Type MODEL_MUST_NOT_EXIST

The Snippet Parameter of type **MODEL_MUST_NOT_EXIST** inserts a Model box panel field into the **MTF Snippet** panel.

The user can type a model name into the panel field, or select a model from the pop-up of models for the Model box.

If the model **does** exist when the Snippet is run, an error will occur and a message written to the panel message area.

The syntax for type **MODEL_MUST_NOT_EXIST** is:

// PARAMETER param_name **MODEL_MUST_NOT_EXIST** param_desc param_default

where

// PARAMETER param_name

This defines a parameter of the given name. See [// PARAMETER param_name] for what is allowed as the parameter name.

MODEL_MUST_NOT_EXIST - the parameter type.

param_desc param_default

The description for the field in the **MTF Snippet** panel and an optional default value that is entered as the value in the field. See [param_descr param_default].

// PARAMETER TM MODEL_MUST_NOT_EXIST "Trimesh model" "Tri RSK 0201"

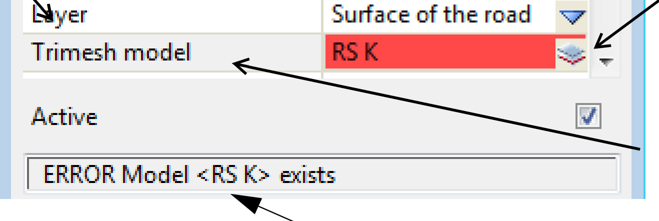
create_strings "selection" "KSA<<LK" "KSA<<IK" "KSA<<TK" "KSA<<BK" start_ref 0 final_ref 0

"no_colour" "" "$TM" absolute extra_start extra_end

...

Model box panel field is in the MTF Snippet panel because of the parameter TM of type MODEL_MUST_NOT_EXIST

Pop-up of models because the parameter type is MODEL_MUST_NOT_EXIST



Description provided by the parameter definition in the snippet

Error message if the model already exists

Important Note:

The Model panel field for a MODEL_MUST_NOT_EXIST parameter will only appear in the **MTF Snippet** panel if the parameter is used in the snippet.

Continue to the next section [21.5.8.12 Snippet Parameters for Tins] or return to [21.5.8.11 Snippet Parameters for Models] or [21.5.8 Snippet Parameters and Commands] or [21.5 Defining and Using Snippets].

21.5.8.12 Snippet Parameters for Tins

There are three different Snippet parameter types for creating a Tin box in the **MTF Snippet** panel. Which parameter is used depends on what is required to be done if the tin exists or does not exit.

(a) use TIN_CREATE when it doesn't matter if the selected tin exists or doesn't exist.

See 21.5.8.12.1 Snippet Parameter Type TIN_CREATE

(b) use TIN_MUST EXIST when the selected model must already exist.

See 21.5.8.12.2 Snippet Parameter Type TIN_MUST_EXIST

(c) use TIN_MUST_NOT_EXIST when the model must **not** already exit.

21.5.8.12.3 Snippet Parameter Type TIN_MUST_NOT_EXIST

Or return to 21.5.8 Snippet Parameters and Commands or 21.5 Defining and Using Snippets.

21.5.8.12.1 Snippet Parameter Type TIN_CREATE

The Snippet Parameter of type **TIN_CREATE** inserts a Tin box panel field into the **MTF Snippet** panel.

The user can type a tin name into the panel field, or select a tin from the pop-up of tins for the Tin box.

If the tin does not exist when the snippet is run, then a new tin is created.

The syntax for type **TIN_CREATE** is:

// PARAMETER param_name **TIN_CREATE** param_desc param_default

where

// PARAMETER param_name

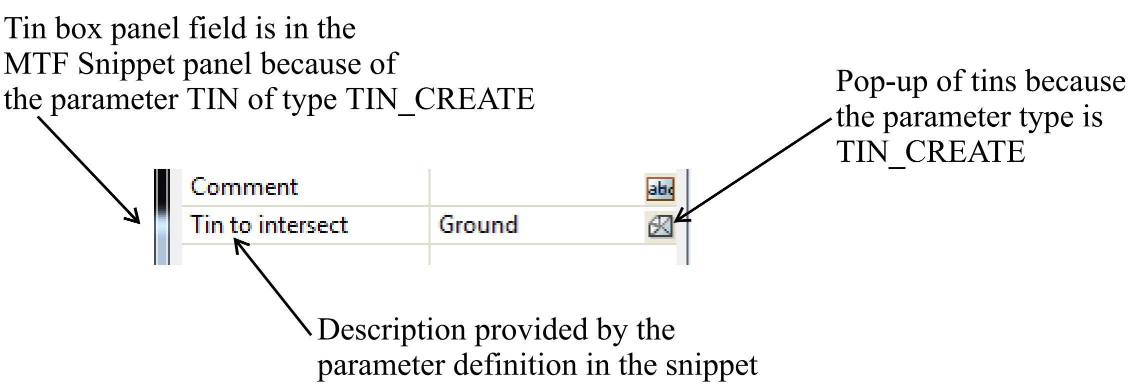
This defines a parameter of the given name. See // PARAMETER param_name for what is allowed as the parameter name.

TIN_CREATE - the parameter type.

param_desc param_default

The description for the field in the **MTF Snippet** panel and an optional default value that is entered as the value in the field. See param_descr param_default.

```
// PARAMETER TIN TIN_CREATE "Tin to intersect" "Ground"
tin_all_fixed "Design<<BA" start_ref 0 final_ref 0 "$TIN" "mod_wdt_hgt_fix" 1 absolute extra_start extra_end
...
```



Important Note:
The Tin panel field for a TIN_CREATE parameter will only appear in the **MTF Snippet** panel if the parameter is used in the snippet.

Continue to the next section 21.5.8.12.2 Snippet Parameter Type TIN_MUST_EXIST or return to 21.5.8.12 Snippet Parameters for Tins or 21.5.8 Snippet Parameters and Commands or 21.5 Defining and Using Snippets.

21.5.8.12.2 Snippet Parameter Type TIN_MUST_EXIST

The Snippet Parameter of type **TIN_MUST_EXIST** inserts a Tin box panel field into the **MTF Snippet** panel.

The user can type a tin name into the panel field, or select a tin from the pop-up of tins for the Tin box.

If the tin does **not** exist when the Snippet is run, an error will occur and a message written to the panel message area.

The syntax for type **TIN_MUST_EXIST** is:

// PARAMETER param_name **TIN_MUST_EXIST** param_desc param_default

where

// PARAMETER param_name

This defines a parameter of the given name. See // PARAMETER param_name for what is allowed as the parameter name.

TIN_MUST_EXIST - the parameter type.

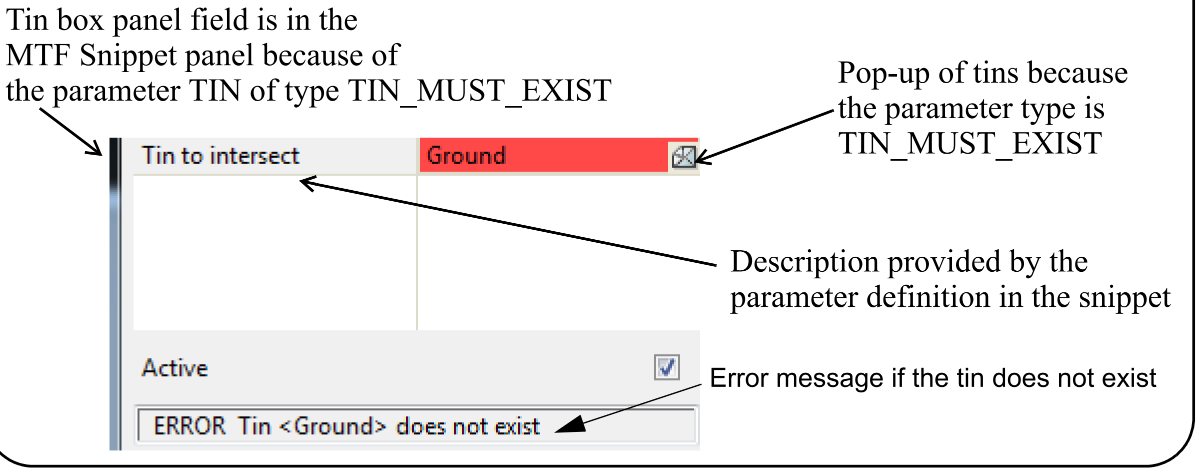
param_desc param_default

The description for the field in the **MTF Snippet** panel and an optional default value that is entered as the value in the field. See param_descr param_default.

// PARAMETER TIN **TIN_MUST_EXIST** "Tin to intersect" "Ground"

tin_all_fixed "Design<<BA" start_ref 0 final_ref 0 "$TIN" "mod_wdt_hgt_fix" 1 absolute extra_start extra_end

...



Important Note:
The Tin panel field for a TIN_MUST_EXIST parameter will only appear in the **MTF Snippet** panel if the parameter is used in the snippet.

Continue to the next section 21.5.8.12.3 Snippet Parameter Type TIN_MUST_NOT_EXIST or return to 21.5.8.12 Snippet Parameters for Tins or 21.5.8 Snippet Parameters and Commands or 21.5 Defining and Using Snippets.

21.5.8.12.3 Snippet Parameter Type TIN_MUST_NOT_EXIST

The Snippet Parameter of type **TIN_MUST_NOT_EXIST** inserts a Tin box panel field into the **MTF Snippet** panel.

The user can type a tin name into the panel field, or select a tin from the pop-up of tins for the Model box.

If the tin **does** exist when the Snippet is run, an error will occur and a message written to the panel message area.

The syntax for type **TIN_MUST_NOT_EXIST** is:

// PARAMETER   param_name   **TIN_MUST_NOT_EXIST**   param_desc   param_default

where

// PARAMETER   param_name

This defines a parameter of the given name. See [// PARAMETER param_name] for what is allowed as the parameter name.

TIN_MUST_NOT_EXIST - the parameter type.

param_desc   param_default

The description for the field in the **MTF Snippet** panel and an optional default value that is entered as the value in the field. See [param_descr   param_default].

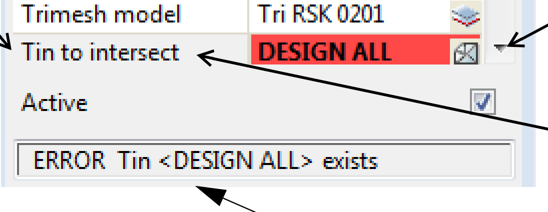
// PARAMETER TIN **TIN_MUST_NOT_EXIST** "Tin to intersect" "Ground"

tin_all_fixed "Design<<BA" start_ref 0 final_ref 0 "$TIN" "mod_wdt_hgt_fix" 1 absolute extra_start extra_end

...

Tin box panel field is in the MTF Snippet panel because of the parameter TIN of type TIN_MUST_NOT_EXIST

Pop-up of tins because the parameter type is TIN_MUST_NOT_EXIST



Description provided by the parameter definition in the snippet

Error message if the tin already exists

Important Note:
The Tin panel field for a TIN_MUST_NOT_EXIST parameter will only appear in the **MTF Snippet** panel if the parameter is used in the snippet.

Continue to the next section 21.5.8.13 Snippet Parameter of Type LAYER or return to 21.5.8.12 Snippet Parameters for Tins or 21.5.8 Snippet Parameters and Commands or 21.5 Defining and Using Snippets.

21.5.8.13 Snippet Parameter of Type LAYER

If a layer for links is required then the **LAYER** parameter type is used.

A Snippet Parameter of type **LAYER** inserts an Layer box panel field into the **MTF Snippet** panel.

The user can type a layer name into the panel field or select a layer name from the pop-up of layers previously defined. The Layer name is then passed down to the snippet.

The syntax for type **LAYER** is:

// PARAMETER param_name **LAYER** param_desc param_default_value

where

// PARAMETER param_name

This defines a parameter of the given name. See // PARAMETER param_name for what is allowed as the parameter name.

LAYER - the parameter type.

param_desc param_default

The description for the field in the **MTF Snippet** panel and an optional default value that is entered as the value in the field. See param_descr param_default.

The value **param_default** is **optional**. But if it exists then if the default layer value does not already exist, it will be added to the choice list of Layer names.

Important Note:
The Layer panel field for a LAYER parameter will only appear in the **MTF Snippet** panel if the parameter is used in the snippet.

Continue to the next section 21.5.8.14 Snippet Parameter of Type NAMED_GRADE or return to 21.5.8 Snippet Parameters and Commands or 21.5 Defining and Using Snippets.

21.5.8.14 Snippet Parameter of Type NAMED_GRADE

If a **named grade definition** is required then the **NAMED_GRADE** parameter type is used.

A Snippet Parameter of type **NAMED_GRADE** inserts an Named Grade box panel field into the **MTF Snippet** panel.

The user can type a named grade into the panel field or select a named grade from the pop-up of named grades previously defined. The Named Grade is then passed down to the snippet.

The syntax for type **NAMED_GRADE** is:

// PARAMETER param_name **NAMED_GRADE** param_description param_default_value

where

// PARAMETER param_name

This defines a parameter of the given name. See // PARAMETER param_name for what is allowed as the parameter name.

NAMED_GRADE - the parameter type.

param_desc param_default

The description for the field in the **MTF Snippet** panel and an optional default value that is entered as the value in the field. See param_descr param_default.

Important Note:
The Named Grade panel field for a NAMED_GRADE parameter will only appear in the **MTF Snippet** panel if the parameter is used in the snippet.

Continue to the next section 21.5.8.15 Snippet Command INFO or return to 21.5.8 Snippet Parameters and Commands or 21.5 Defining and Using Snippets.

21.5.8.15 Snippet Command INFO

There is an **Info** button on the **MTF Snippet** panel and when clicked, all the lines of the snippet file that start with "// INFO" are displayed in **Snippet Description** pop up.

The syntax for **INFO** is:

// INFO *text_to_display*

where

// INFO

this is at the beginning of the line and defines the command INFO. Note that there must be one or more spaces between the // and INFO.

text_to_display

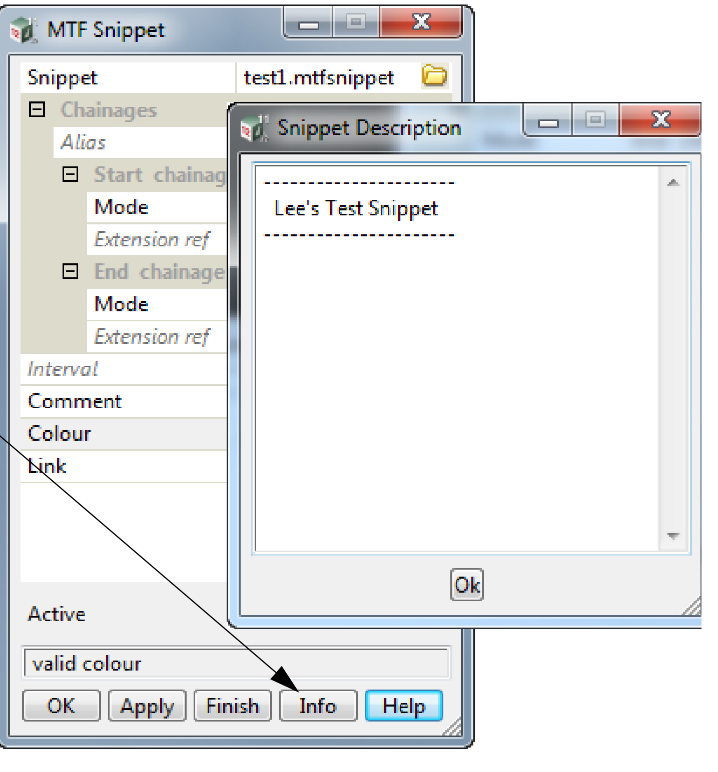
this is a line of text that is to be displayed in the **Snippet Description** pop up when the **Info** button is clicked on the **MTF Snippet** panel. The text is taken from one space after the word INFO and to the end of the line.

For example, in the Snippet code:

```
// PARAMETER      CLR    COLOUR      "Colour"      "red"
// PARAMETER      LN     TEXT        "Link"        ""
// INFO ----------------------
// INFO   Lee's Test Snippet
// INFO ----------------------
insert "Design=>$(LN)" $CLR  0.0  0.0 unknown $(_SCH) 0 $(_ECH) 0 absolute extra_start extra_end
```

INFO commands

Click on **Info** and the **Snippet Description** pop-up. displays all the INFO lines in the Snippet.



Continue to the next section 21.5.8.16 Snippet Command DISPLAY or return to 21.5 Defining and Using Snippets.

21.5.8.16 Snippet Command DISPLAY

The **parameter description** and the **current value** that the parameter has can be displayed in the **Detail** and **Value** columns of the **MTF Modifiers** panel by using the **DISPLAY** command in the snippet.

The syntax for **DISPLAY** is:

// DISPLAY param_name

where

// DISPLAY

this is at the beginning of the line and defines the command DISPLAY. Note that there must be one or more spaces between the // and DISPLAY.

param_name

this is the name of the parameters to have its description and current value displayed in the **MTF Modifiers** panel.

When the **OK** or **Apply** button is clicked on the **MTF Snippets** panel, the ***description*** of the parameter **param_name** is displayed in the **Details** column of the **MTF Modifiers** panel and the ***current value*** of the parameter **param_name** is displayed in the **Value** column of the **MTF Modifiers** panel.

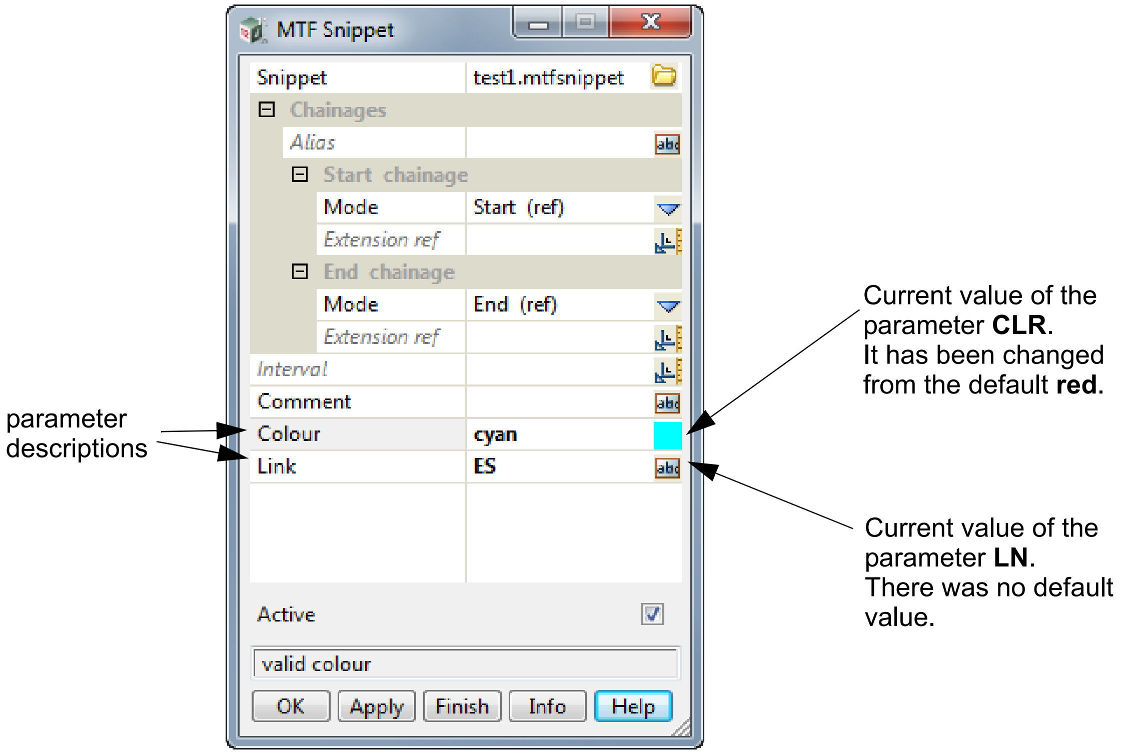
For example, in the Snippet code:

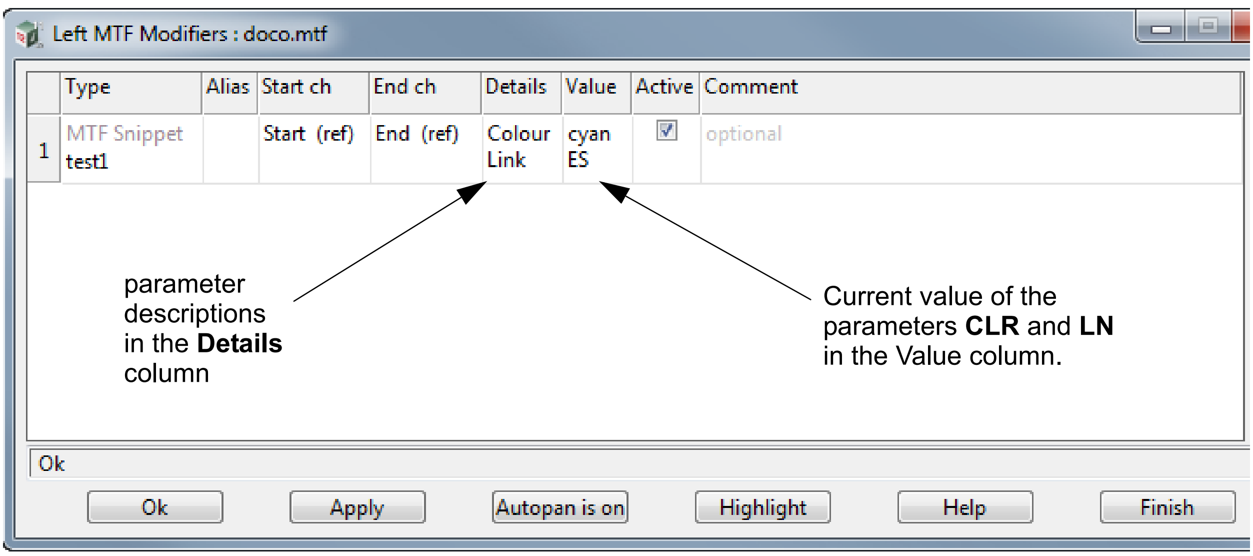
parameter description

```
// PARAMETER     CLR    COLOUR    "Colour"     "red"
// PARAMETER     LN     TEXT      "Link"       ""
// DISPLAY CLR
// DISPLAY LN
```

parameter description

```
insert "Design=>$(LN)" $CLR  0.0  0.0 unknown $(_SCH) 0 $(_ECH) 0 absolute extra_start extra_end
```





Continue to the next section 21.5.15.3 Temporary MTF Snippet File or return to 21.5 Defining and Using Snippets.

21.5.8.17 Snippet Command INCREMENT

There is an **INCREMENT** command to increment a snippet parameter by a given increment step. The increment step can be negative for a decrement.

The intended use is for TEXT parameters but it will actually work for any type of parameter but probably in an unexpected way.

How a parameter is incremented depends on the value of the parameter when it is interpreted as text. We will refer to the value of the parameter as text, as **value_text**.

For example, for an INTEGER, the value_text could be "13", for a TEXT it could be "STN003", for a REAL, it could be "23.45" and for a COLOUR, it could be "red". Obviously it usually doesn't make sense to increment some parameters like REAL or COLOUR but it can be done.

The INCREMENT command only increments either a

(a) a group of digits at the end of the text. That is each character is one of "0" to "9". The number of digits to increment is fixed (the length of the group).

or

(b) a group of letters at the end of the text. That is, each characters is one of "a" to "z". The number of letters to increment is fixed (the length of the group). Upper and lower case letters are considered to be the same thing.

Note that INCREMENT does not increment a mixture of digits and letters.

For the **value_text**, the group of characters at the end of the value_text that will be incremented is determined by looking at the last character of the value_text and

(a) if the last character is a **digit** then it will be a **digit group**.

The rest of **value_text** is then searched from right to left until a letter is found. So all the digits at the end of the value_text make up a **digit group** of a fixed length.

For example, for "STN001" the group is digit group of length 3 ("001").

For "ST2N02", the group is a digit group of length 2 ("02").

For "002", the group is a digit group of length 3 ("002").

The digit group increments by the increment step but it will always keep the same number of digits. So if the incremented digit group would need an extra digit then that extra leading digit is dropped. So the number wraps around.

For example, incrementing "568" by 1 will eventually reach "999" and the next increment will give you "000".

(b) if the last character is an letter (i.e. one of a to z) then it will be a **letter group**.

The rest of **value_text** is then searched from right to left until a digit is found. So all the letters at the end of value_text make up a **letter group** of a fixed length.

For example, for "123ABC" the group is an letter group of length 3 ("ABC").

"incrementing a letter" means that you take the letters position in the alphabet and add the increment step to it to give you the position of the new incremented letter. For example, **c** is the third letter and increment it by two give the fifth letter which is **e**. When you have a letter group then you work in base 26. For example, incrementing AZ by 1 gives you BA.

As in the case for a digit group, the letter group increments by the increment step but it will always keep the same number of letters. So if the incremented digit group would need an extra letter then that extra leading letter is dropped.

For example, incrementing "TUV" by 1 will eventually reach "ZZZ" and the next increment will give you "AAA".

The syntax for **INCREMENT** is:

// INCREMENT   param_name   increment_step

where

// INCREMENT

this is at the beginning of the line and defines the command INCREMENT. Note that there must be one or more spaces between the // and INCREMENT.

param_name

the name of the parameter to increment.

increment_step

this is the amount the increment the group of digits or letters. The increment step can be negative for a decrement.

As an example,

```
// PARAMETER  STRING_ID  TEXT   "String number"   1
// INCREMENT   STRING_ID   1
```

Continue to the next section 21.5.8.18 Optional Parameters in Snippets or return to 21.5 Defining and Using Snippets.

21.5.8.18 Optional Parameters in Snippets

Parameters can be set to be optional so that when they are displayed in the **MTF Snippet** panel and there is no value in the panel field, the Description is greyed out.

A parameters is made optional by adding **OPTIONAL** as the last word on the parameter line.

A default value for the parameter can still be passed in but in that case the Description in the **MTF Snippet** panel field will not be greyed out when it is first displayed because there is a value in it.

For example, in the Snippet code:

the default value has been omitted

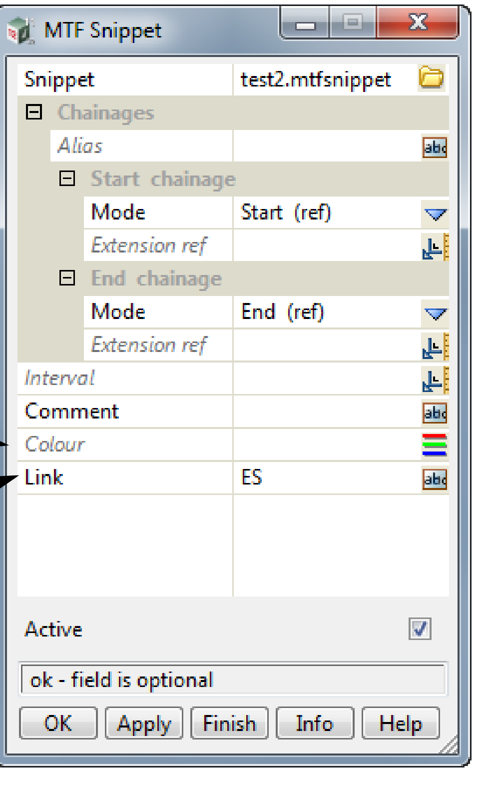
```
// PARAMETER   CLR   COLOUR   "Colour"   OPTIONAL
// PARAMETER   LN     TEXT       "Link"    "ES"   OPTIONAL
// DISPLAY CLR
// DISPLAY LN
```

default value

insert "Design=>$(LN)" $CLR 0.0 0.0 unknown $(_SCH) 0 $(_ECH) 0 absolute extra_start extra_end

Colour is optional

Link is optional but because there is a value in it, the word Link is not greyed out



WARNING

Be careful when making a parameter optional because the MTF may then have errors.

For example in the above example, if the **Colour** is not filled in then there will be an error in the MTF because the command

insert "Design=>$(LN)" $CLR 0.0 0.0 unknown $(_SCH) 0 $(_ECH) 0 absolute extra_start extra_end

will not be valid if CLR is blank and hence not there.

Continue to the next section 21.5.9 Automatic Parameters in Snippets or return to 21.5.8 Snippet Parameters and Commands or 21.5 Defining and Using Snippets.

21.5.9 Automatic Parameters in Snippets

In addition to the ability to create and define your own parameters in snippets, ***12d Model*** has a range of predefined parameters that are automatically created and calculated.

Automatic parameters allow for the use of values that are important to the snippet, but that you do not necessarily want to prompt the user to complete.

For example, string naming that is dependent on whether it is applied in the left side or right side modifiers. Rather than making the user select the correct left/right side value as a parameter, an automatic parameter can be used instead. In this way, automatic parameters can greatly simplify snippets and make them more portable.

Automatic parameters are fixed by the system and are identified by **starting with an underscore**, e.g. $(_AUTO_LR) or $_AUTO_LR.

All automatic parameter names are reserved in 12d. To ensure future compatibility, user-defined parameters should not start with an underscore.

Currently, the following automatic parameters are defined:

21.5.9.1 _AUTO_LR
21.5.9.2 _SCH 21.5.9.3 _ECH
21.5.9.4 _CL_REF 21.5.9.5 _CL_REF1 21.5.9.6 _CL_REF2
21.5.9.7 _CL_REF3 21.5.9.8 _CL_REF4
21.5.9.9 _NULL
21.5.9.10 _AUTO_0I, _AUTO_1J, ... , _AUTO_9R, _AUTO_AS, _AUTO_BT, ... _AUTO_HZ
21.5.9.11 _AUTO_05, _AUTO_16, _AUTO_27, ... _AUTO_49, _AUTO_AN, _AUTO_BO, ... _AUTO_MZ
21.5.9.12 _INS_05, _INS_LR, _INS_0I
21.5.9.13 _INS_SIDE_05
21.5.9.14 _SIDE_FX
21.5.9.15 _AUTO_LAYER_LR
21.5.9.16 _APPLY_TIN
21.5.9.17 _APPLY_DESIGN_MODEL
21.5.9.18 _HINGE
21.5.9.19 _SIDE
21.5.9.20 _SILENT_NG
21.5.9.21 _PROJECT_ATTRIBUTE

21.5.9.1 _AUTO_LR

Depending on the side from which the snippet is called this parameter will insert the value **L** if it is called from the left side modifiers and **R** if it is called from the right side modifiers.

```
insert "KLIP$(_AUTO_LR)" "red" 5 unknown -2 $(_SCH) 0 $(_ECH) 0 absolute
               extra_start extra_end
```

Continue to 21.5.9.2 _SCH or return to 21.5.9 Automatic Parameters in Snippets.

21.5.9.2 _SCH

Will insert **named_position MODIFIER_START**.

This is typically used to insert the start chainage mode that has been passed to the snippet.

Note that this parameter does **not** include the extension value.

```
insert "KLIP" "red" 5 unknown -2 $(_SCH) 0 $(_ECH) 0 absolute extra_start extra_end
```

Continue to 21.5.9.3 _ECH or return to 21.5.9 Automatic Parameters in Snippets.

21.5.9.3 _ECH

Will insert **named_position MODIFIER_END**.

This is typically used to insert the end chainage mode that has been passed to the snippet.

Note that this parameter does **not** include the extension value.

```
insert "KLIP" "red" 5 unknown -2 $(_SCH) 0 $(_ECH) 0 absolute extra_start extra_end
```

Continue to 21.5.9.4 _CL_REF or return to 21.5.9 Automatic Parameters in Snippets.

21.5.9.4 _CL_REF

This will insert the entire name of the Reference string from the Apply many function.

For example, if the reference string name is **RSR01**, $(_CL_REF) will insert **RSR01**

Continue to 21.5.9.5 _CL_REF1 or return to 21.5.9 Automatic Parameters in Snippets.

21.5.9.5 _CL_REF1

This will insert the last character of the name of the Reference string from the Apply many function.

For example, if the reference string name is **RSR01**, $(_CL_REF1) will insert **1**.

Continue to 21.5.9.6 _CL_REF2 or return to 21.5.9 Automatic Parameters in Snippets.

21.5.9.6 _CL_REF2

This will insert the last two characters of the name of the Reference string from the Apply many function.

For example, if the reference string name is **RSR01**, $(_CL_REF1) will insert **01**.

Continue to 21.5.9.7 _CL_REF3 or return to 21.5.9 Automatic Parameters in Snippets.

21.5.9.7 _CL_REF3

This will insert the last three characters of the name of the Reference string from the Apply many function.

For example, if the reference string name is **RSR01**, $(_CL_REF1) will insert **R01**.

Continue to 21.5.9.8 _CL_REF4 or return to 21.5.9 Automatic Parameters in Snippets.

21.5.9.8 _CL_REF4

This will insert the last four characters of the name of the Reference string from the Apply many function.

For example, if the reference string name is **RSR01**, $(_CL_REF1) will insert **SR01**.

Continue to 21.5.9.9 _NULL or return to 21.5.9 Automatic Parameters in Snippets.

21.5.9.9 _NULL

The current MTF syntax includes the use of $null for various values. In snippets, this would be interpreted as a parameter or token null.

Therefore, in snippets, if you would still like to include $null in MTF commands, they need to be replaced with this automatic parameter.

An example of MTF modifier syntax - note the present of $null values:

```
named_grade  "TEST"  start_ref  0  final_ref 0  "links"  "Design=>CLIN"  "Design=>EDGE"
        $null  $null  $null  $null  ""  absolute  extra_start  extra_end
```

The MTF snippet syntax for it - where $null has been replaced with $(_NULL):

```
named_grade  "TEST"  $(_SCH)  0  $(_ECH)  0  "links"  "Design=>CLIN"  "Design=>EDGE"
        $(_NULL)  $(_NULL)  $(_NULL)  $(_NULL)  ""  absolute  extra_start  extra_end
```

Continue to 21.5.9.10 _AUTO_0I, _AUTO_1J, … , _AUTO_9R, _AUTO_AS, _AUTO_BT, … _AUTO_HZ or return to 21.5.9 Automatic Parameters in Snippets.

21.5.9.10 _AUTO_0I, _AUTO_1J, … , _AUTO_9R, _AUTO_AS, _AUTO_BT, … _AUTO_HZ

This is a set of 18 different, but related, automatic parameters to assist with string naming conventions for multiple occurrences.

These automatic parameters follow the MOSS/MX naming convention for multiple occurrences:

Side	Occurrence																	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Left	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	G	H
Right	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Left side is 0 to 9, A to H. **Right side** is I to Z.

So, for the 1st occurrence of a string, you would use the **_AUTO_0I** parameter.

For the 2nd occurrence of the string, you would use the **_AUTO_1J** parameter, and so on until for the 18th occurrence you would use the **_AUTO_HZ** parameter.

12d will automatically insert either 0/I, 1/J, 2/K … H/Z depending on whether the snippet is called on the **left** or **right** side of the modifiers.

Hence the names of the automatic parameter _AUTO_0I include the **0I** to indicate that it substitutes **0** when on the Left and **I** when on the right.

```
insert "LANE$(_AUTO_0I)" "red" 3.5 unknown -3 $(_SCH) 0 $(_ECH) 0 absolute extra_start extra_end
insert "LANE$(_AUTO_1J)" "red" 3.5 unknown -3 $(_SCH) 0 $(_ECH) 0 absolute extra_start extra_end
insert "LANE$(_AUTO_2K)" "red" 3.5 unknown -3 $(_SCH) 0 $(_ECH) 0 absolute extra_start extra_end
```

Continue to 21.5.9.11 _AUTO_05, _AUTO_16, _AUTO_27, … _AUTO_49, _AUTO_AN, _AUTO_BO, …. _AUTO_MZ or return to 21.5.9 Automatic Parameters in Snippets.

21.5.9.11 _AUTO_05, _AUTO_16, _AUTO_27, … _AUTO_49, _AUTO_AN, _AUTO_BO, …. _AUTO_MZ

This is a set of 18 different, but related, automatic parameters to assist with string naming conventions for multiple occurrences.

These automatic parameters follow a similar, but alternative naming convention to the MOSS/MX for multiple occurrences:

Side	Occurrence																	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Left	0	1	2	3	4	A	B	C	D	E	F	G	H	I	J	K	L	M
Right	5	6	7	8	9	N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Left side is 0 to 9, A to H. **Right side** is I to Z.

So, for the 1st occurrence of a string, you would use the _AUTO_05 parameter.

For the 2nd occurrence of the string, you would use the _AUTO_16 parameter, and so on until for the 18th occurrence you would use the _AUTO_MZ parameter.

12d will automatically insert either 0/5, 1/6, 3/7 … M/Z depending on whether the snippet is called on the **left** or **right** side of the modifiers.

Hence the names of the automatic parameter _AUTO_05 include the **05** to indicate that it substitutes **0** when on the Left and **5** when on the right.

```
insert "LANE$(_AUTO_05)" "red" 3.5 unknown -3 $(_SCH) 0 $(_ECH) 0 absolute extra_start extra_end
insert "LANE$(_AUTO_16)" "red" 3.5 unknown -3 $(_SCH) 0 $(_ECH) 0 absolute extra_start extra_end
insert "LANE$(_AUTO_27)" "red" 3.5 unknown -3 $(_SCH) 0 $(_ECH) 0 absolute extra_start extra_end
```

Continue to 21.5.9.12 _INS_05, _INS_LR, _INS_0I or return to 21.5.9 Automatic Parameters in Snippets.

21.5.9.12 _INS_05, _INS_LR, _INS_0I

This is used as a normal TEXT parameter, which accepts a single character indicating the side.

Different automatic parameters are used depending on the character expected.

_INS_05 will insert **0** to **4** for left side, **5** to **9** for right side

_INS_LR will insert **L** for left side, **R** for right side

_INS_0I will insert **0** to **9** or **A** to **H** for left side, **I** to **Z** for right side

```
// PARAMETER _INS_05 TEXT "Insertion side (0...4 / 5...9)"
// PARAMETER _INS_LR TEXT "Insertion side (L / R)"
// PARAMETER _INS_0I TEXT "Insertion side (0...9,A...H / I...Z)"
```

Once set-up, the **_INS_??** automatic parameter is used to set the **_SIDE_FX** automatic parameter (see 21.5.9.14 _SIDE_FX).

Note that, as with any other parameter, in order for the _INS_05 / _INS_LR / _INS_0I parameter to appear in the MTF Snippet panel, it must be used at least once elsewhere in the snippet. This can be done in a token definition, directive, or expression.

Continue to 21.5.9.13 _INS_SIDE_05 or return to 21.5.9 Automatic Parameters in Snippets.

21.5.9.13 _INS_SIDE_05

This is similar to _INS_05 / _INS_0I / _INS_LR, but without the need to define it as a parameter.

The _INS_SIDE_05 takes a single character.

The value of _INS_SIDE_05 is used to determine the value for the _SIDE_FX automatic parameter.

_INS_SIDE_05 will be **0** to **4** for left side, **5** to **9** for right side

The _INS_SIDE_05 is unique as an automatic parameter, since its value can be set.

In this way, it behaves more like a Directive (See 21.5.12 Snippet Directives).

To set the value of the _INS_SIDE_05 automatic parameter

$(_INS_SIDE_05)<value>

<value> should be a parameter.

The _INS_SIDE_05 should not be used anywhere else in snippets, including user_message commands, except to define its value.

```
// PARAMETER LINK1 TEXT "Attach to link" "EDGE0R"
// Set the _INS_SIDE_05 auto parameter to the 5th character of the LINK1 parameter
$(_INS_SIDE_05)$(LINK1[5:5])
user_message "_SIDE_FX = $(_SIDE_FX) " // Display the current value of _SIDE_FX

insert "Design$(_SIDE_FX)KLIP0R" "red" 2 unknown -3 $(_SCH) 0 $(_ECH) 0 absolute
        extra_start   extra_end
```

Continue to 21.5.9.14 _SIDE_FX or return to 21.5.9 Automatic Parameters in Snippets.

21.5.9.14 _SIDE_FX

This will insert the link side indicator **<<** for link on left side, or **>>** for link on right side, depending on the most recent use of **_INS_***.

That is, if the last state of **_INS_*** corresponds to the left side, **_SIDE_FX** will insert **<<** to indicate insertion to the left.

If the last state of **_INS_*** corresponds to the right side, **_SIDE_FX** will insert **>>** to indicate insertion to the right. Hence if

_INS_05 = 0 to 4, then _SIDE_FX = **<<**

_INS_05 = 5 to 9, then _SIDE_FX = >>

_INS_LR = L, then _SIDE_FX = <<

_INS_LR = R, then _SIDE_FX = >>

_INS_0I = 0 to 9, A to H, then _SIDE_FX = <<

_INS_0I = I to Z, then _SIDE_FX = >>

_INS_SIDE_05 = 0 to 4, then _SIDE_FX = <<

_INS_SIDE_05 = 5 to 9, then _SIDE_FX = >>

Note that in order for the _SIDE_FX parameter to be set, the corresponding parameter _INS_* must also have been set, given a value (0 to 9, A to Z) and used somewhere in the snippet.

For this reason, if not using the _INS_05 directly, it is necessary to reference it in a temporary parameter, token or expression.

```
// PARAMETER _INS_05 TEXT "Insertion side (0...4 / 5...9)"
// PARAMETER _INS_LR TEXT "Insertion side (L / R)"
// PARAMETER _INS_0I TEXT "Insertion side (0...9,A...H / I...Z)"
// DISPLAY _INS_05
// DISPLAY _INS_LR
// DISPLAY _INS_0I

// The _INS_* parameter must be used at least once in the snippet
// This can be done in a token, directive, expression or command.
// The $(_INS_*) must be used to correctly set the _SIDE_FX
// automatic parameter.
user_message_print "_INS_05 = $(_INS_05)"// _SIDE_FX now based on this parameter
user_message_print "_SIDE_FX after _INS_05 = $(_SIDE_FX) "
insert_absolute "Design$(_SIDE_FX)EDGE" "red" 3.5 unknown -3 $(_SCH) 0 $(_ECH) 0 absolute
        extra_start   extra_end

user_message_print "_INS_LR = $(_INS_LR)"// _SIDE_FX now based on this parameter
user_message_print "_SIDE_FX after _INS_LR = $(_SIDE_FX) "
insert_absolute "Design$(_SIDE_FX)EDGE" "yellow" 3.5 unknown -3 $(_SCH) 0 $(_ECH) 0 absolute
        extra_start   extra_end

user_message_print "_INS_0I = $(_INS_0I)"// _SIDE_FX now based on this parameter
user_message_print "_SIDE_FX after _INS_0I = $(_SIDE_FX) "
insert_absolute "Design$(_SIDE_FX)EDGE" "green" 3.5 unknown -3 $(_SCH) 0 $(_ECH) 0 absolute
        extra_start   extra_end
```

The _SIDE_FX allows links to be inserted on any side of the reference, regardless of the side calling the snippet. For example, a snippet on the right side could insert a link on the left side.

Once the _SIDE_FX value is used in the snippet, its value will remain the same until re-defined by any of the _INS_05 / _INS_LR / INS_0I / _INS_SIDE_05 automatic parameters.

Continue to 21.5.9.15 _AUTO_LAYER_LR or return to 21.5.9 Automatic Parameters in Snippets.

21.5.9.15 _AUTO_LAYER_LR

Depending on which side the snippet is called from, will insert the value **layer_left** if called from the left side modifiers and **layer_right** if called from the right side modifiers.

This is typically used for the Create String modifier commands.

```
create_strings "$(_AUTO_LAYER_LR)" "Design=>" "*" $(_SCH) 0 $(_ECH) 0 "no_colour" ""
      "DES ROAD STRS"   absolute   extra_start   extra_end
```

Continue to 21.5.9.16 _APPLY_TIN or return to 21.5.9 Automatic Parameters in Snippets.

21.5.9.16 _APPLY_TIN

This will insert the name of the tin from the **Apply MTF** function

```
interface_tin "Design=>INT" $(_SCH) 0 $(_ECH) 0 "Design=>"   "$(_APPLY_TIN)" 1 2 100 0
              "red"    "green"    0   0    absolute   extra_start     extra_end
```

Continue to 21.5.9.17 _APPLY_DESIGN_MODEL or return to 21.5.9 Automatic Parameters in Snippets.

21.5.9.17 _APPLY_DESIGN_MODEL

This will insert the name of the **Road Surface Strings** model entered in the **Models** tab of the **Apply MTF** function panel.

```
create_strings "$(_AUTO_LAYER_LR)" "Design=>"  "*"  $(_SCH) 0 $(_ECH)  0  "no_colour"   ""
               "$(_APPLY_DESIGN_MODEL)"    absolute    extra_start     extra_end
```

Continue to 21.5.9.18 _HINGE or return to 21.5.9 Automatic Parameters in Snippets.

21.5.9.18 _HINGE

This will insert the name of the **hinge string**.

Since the user can define a custom hinge name (the default is **HINGE**), this is needed to ensure portability and compatibility with different users and setups.

```
named_grade "CWAY" $(_SCH) 0 $(_ECH) 0 "links" "Design=>HING" "Design=>EDGE"
       $null   $null   $null   $null   ""    absolute   extra_start     extra_end
```

Continue to 21.5.9.19 _SIDE or return to 21.5.9 Automatic Parameters in Snippets.

21.5.9.19 _SIDE

Depending on which side the snippet is called from, will insert the value **-1** if called from the left side modifiers and **+1** if called from the right side modifiers.

Continue to 21.5.9.20 _SILENT_NG or return to 21.5.9 Automatic Parameters in Snippets.

21.5.9.20 _SILENT_NG

This will insert **_SILENT_NAMED_GRADE**.

When inserted in front of a named grade, the named grade will not appear in any Named Grade choice boxes.

This is typically used for temporary or construction named grades in snippets that you do not wish the end user to see in the GUI.

Continue to 21.5.9.21 _PROJECT_ATTRIBUTE or return to 21.5.9 Automatic Parameters in Snippets.

21.5.9.21 _PROJECT_ATTRIBUTE

This will insert a **project attribute** from the current 12d project.

The particular attribute to be inserted is specified between square brackets [ ] and follows the standard convention of attribute paths in ***12d Model***.

```
user_message "$(_PROJECT_ATTRIBUTE[ProjectDetails/ProjectNumber/Value])"
```

This will insert the value of the **ProjectNumber** entered in the **Project Details** panel (if ProjectNumber is defined).

Continue to 21.5.9.22 Substrings of parameters or return to 21.5.9 Automatic Parameters in Snippets.

21.5.9.22 Substrings of parameters

Although there are some preset automatic parameters for extracting a few characters from a string, e.g. _CL_REF3, it is possible to extract a substring of all **User** parameters, and the **one** special **auto parameter case**, _CL_REF.

This is done with the following syntax:.

```
param[start:end]
```

param is the parameter or token name

start is the character start position (the first character is at position 1). It must be an integer greater than or equal to 1.

end is the character end position (the first character is at position 1). It must be an integer greater than or equal to the start position.

For the value of the selected **param**, only the text from **start** to **end** will be inserted.

Both **start** and **end** are compulsory. The colon is also compulsory.

If **end** is greater than the length of the complete param value, an end position equal to the string length will be used. That is, everything from the start up to the end of the string will be returned.

start must be less than **end**.

Substrings are currently supported on all user parameters. Automatic parameters and tokens are not supported.

For example,

if $(TEXT_PARAM) has the value **Some text**

then	$(TEXT_PARAM[1:3])	is	**Som**
	$(TEXT_PARAM[4:7])	is	**e t**
	$(TEXT_PARAM[1:20])	is	**Some text**
	$(TEXT_PARAM[5:9])$(TEXT_PARAM[1:4])	is	**textSome**

For the special auto parameter case, an example is:

$(_CL_REF[2:6])

Continue to the next section 21.5.10 Arithmetic in Snippets or return to 21.5.9 Automatic Parameters in Snippets or 21.5 Defining and Using Snippets.

21.5.10 Arithmetic in Snippets

Arithmetic expressions can be used for any of the real values in the lines of a snippet file as long as special rules are obeyed.

1. If a real value is replaced by an expression then the expression must be surrounded by round brackets. That is "**(**" and "**)**".
2. If the arithmetic operators plus **+**, minus **-**, multiplication ***** and division **/** are used then there must be a space on both sides of them.

   For example, **($(TL) - $(PL))** and **(0.0 - ($(DL) / 2.0) - 0.5)**

   Note that the negative of a number has the negative sign hard up against the number with no space in-between. For example **-0.5**

WARNING - a future interactive Snippet editor may not be able to support arithmetic in snippets. Those snippets with arithmetic will still need to be modified using a text editor.

An example of some lines from a snippet file using an arithmetic expression is:

```
// RTA Type F Half Kerb to SA transition

// PARAMETER TL REAL "Transition Length" 5.5
// PARAMETER PL REAL "Part Length" 1.340

// insert the links for the 'F' to 'short F' transition

insert "1" "grey" 0.001  0.000 unknown named_position "MODIFIER_START" 0.0 named_position
"MODIFIER_START" ($(TL) - $(PL)) absolute extra_start extra_end
insert "2" "grey" 0.499 -0.040 unknown named_position "MODIFIER_START" 0.0 named_position
"MODIFIER_START" ($(TL) - $(PL)) absolute extra_start extra_end
...
```

Continue to the next section 21.5.11 Trig and Maths Function Capabilities in Snippets or return to 21.5 Defining and Using Snippets.

21.5.11 Trig and Maths Function Capabilities in Snippets

There is now some trig and maths capability in the MTF snippet grammar.

The trig functions supported are:

sin(x), cos(x), tan(x) - trig functions

asin(x), acos(x), atan(x) - the inverse trig functions

square(x) - return x*x

sqrt(x)

fabs(x) - returns the absolute value of its x.

ceil(x) - the ceil functions return a floating-point value that represents the smallest integer that is greater than or equal to x.

floor(x) - the floor functions return a floating-point value that represents the largest integer that is less than or equal to x.

log(x) - the log functions return the natural logarithm (base e) of x if successful.

exp(x) - the exp function returns the exponential value of the floating-point parameter, x, if successful. That is, the result is e to the power x, where e is the base of the natural logarithm.

cube(x) - return x*x*x

pow(x,y) - returns the value of x to the power y.

fmod(x,y) - returns the floating-point remainder of x / y

pi, halfpi, twopi - so you don't have to keep typing the values in

The trig functions etc then allows calculations in a snippet like

```
// Swept angle of 2nd radius.
//
@def_tok  A2  "(asin(($(D2) - $(R1) +$(DEL_R1_R2) * cos($(A1))) / $(R2)) - $(A1) + halfpi)
```

Note - Mathematical formulae in snippets should always be quoted to prevent concatenation and unexpected results.

Continue to the next section 21.5.12.1.1 #define in Snippets - Do Not Use or return to 21.5 Defining and Using Snippets.

21.5.12 Snippet Directives

See

Or return to 21.5 Defining and Using Snippets.

21.5.12.1 Deprecated C Preprocessor from V10

When Snippets were first introduced in ***12d Model 10***, the C preprocessor syntax (*e.g.* #define or #if / #else / #endif) was used to increase the power of snippets by performing conditional tests and evaluations on the final MTF file containing the snippet.

The C preprocessor is used throughout ***12d Model*** (*e.g.* setup files) but once you progress from simple uses, it can be very confusing for users to fully understand.

```
// PARAMETER DL REAL "Drive length"   3.0
// PARAMETER WD REAL "Layback width"  0.6
// PARAMETER DP REAL "Layback depth" -0.04
#define _DR1 (0.0 - ($DL / 2.0) - 0.5)
#define _DR2 (0.0 - ($DL / 2.0))
#define _DR3 ( ($DL / 2.0))
#define _DR4 (($DL / 2.0) + 0.5)
// drop into DW
insert "SAL" "grey" 0.001 0.000 unknown named_position "MODIFIER_START" _DR1
named_position "MODIFIER_START" _DR2 absolute extra_start extra_end
insert "SAI" "grey" 0.499 $DP   unknown named_position "MODIFIER_START" _DR1
named_position "MODIFIER_START" _DR2 absolute extra_start extra_end
insert "SAT" "grey" 0.030 0.150 unknown named_position "MODIFIER_START" _DR1
named_position "MODIFIER_START" _DR2 absolute extra_start extra_end
insert "SAB" "grey" 0.180 0.000 unknown named_position "MODIFIER_START" _DR1
named_position "MODIFIER_START" _DR2 absolute extra_start extra_end
```

Due to the complexity and limitations of using the C preprocessor, a new preprocessor has been created in ***12d Model 11*** specifically for Snippets.

Snippets in ***12d Model 11*** can still use the C preprocessor style of directives and such snippets will still run in ***12d Model 11***. However, the use of the older style of directives has been superseded (deprecated) in ***12d Model 11*** and users are encouraged to only use the special Snippet Directives.

Continue to 21.5.12.1.1 #define in Snippets - Do Not Use or return to 21.5.12 Snippet Directives.

21.5.12.1.1 #define in Snippets - Do Not Use

IMPORTANT WARNING

From ***12d Model*** 11 onwards, **#define** has been replaced by **@def_tok** directive and #define should not be used.

It is only documented here in case you are working on an old Snippet and need to know what the command does.

#define can be used in a snippet file to define parameters for use inside the snippet file.

WARNING - a future interactive Snippet editor may not be able to support #define. Those snippets with #define will still need to be modified using a text editor.

The #define is placed at the beginning of a line, followed by one or more spaces and then the define_name to be used, and then one or more spaces and the expression that the define_name stands for.

```
#define   define_name    define_expression
```

Then the define_name can be used instead of the define_expression anywhere else in the snippet file following the #define.

Note - snippet parameters can be used in the define_expression as long as the definition of the parameter occurs before the #define.

For example

```
// PARAMETER DL REAL "Drive length"   3.0
// PARAMETER WD REAL "Layback width"  0.6
// PARAMETER DP REAL "Layback depth" -0.04

#define _DR1 (0.0 - ($DL / 2.0) - 0.5)
#define _DR2 (0.0 - ($DL / 2.0) )
#define _DR3 ( ($DL / 2.0) )
#define _DR4 ( ($DL / 2.0) + 0.5)

// drop into DW
insert "SAL" "grey" 0.001 0.000 unknown named_position "MODIFIER_START" _DR1 
named_position "MODIFIER_START" _DR2 absolute extra_start extra_end
insert "SAI" "grey" 0.499 $DP   unknown named_position "MODIFIER_START" _DR1 
named_position "MODIFIER_START" _DR2 absolute extra_start extra_end
insert "SAT" "grey" 0.030 0.150 unknown named_position "MODIFIER_START" _DR1 
named_position "MODIFIER_START" _DR2 absolute extra_start extra_end
insert "SAB" "grey" 0.180 0.000 unknown named_position "MODIFIER_START" _DR1 
named_position "MODIFIER_START" _DR2 absolute extra_start extra_end
...
```

Return to 21.5.12.1 Deprecated C Preprocessor from V10.

21.5.12.2 Snippet Directives for V11 Onwards

The syntax for a Snippet Directive starts with a **@**, then **a space**, followed by the **directive** and **any additional arguments**.

```
@ <directive> <argument> [<argument>... <argument>]
```

<directive> - name of the directive.

<argument> - any required arguments for the directive. The number and type of these arguments will depend on the directive.

```
@ if_val_eq "$(KT)" "SM"
// Do something in here
@ end if
```

For information on all the snippet directives, see the sections

21.5.12.3 Flow control.
21.5.12.4 Abbreviations.
21.5.12.5 Comparisons of Data Types.
21.5.12.6 Tokens and Tokens vs Tokens.
21.5.12.7 Tokens vs Strings.
21.5.12.8 Tokens vs Doubles.
21.5.12.9 Tokens vs Integers.
21.5.12.10 Values Defined and Value vs Value.
21.5.12.11 Values vs Strings.
21.5.12.12 Values vs Doubles.
21.5.12.13 Values vs Integers.
21.5.12.10 Values Defined and Value vs Value.

Or return to 21.5.12 Snippet Directives or 21.5 Defining and Using Snippets.

21.5.12.3 Flow control

In a snippet, the normal processing flow is that each snippet command is processed in the order in which they are encountered. It is possible, however, to change the order in which such commands are processed or even to totally skip commands. These are done through **flow control directives** in the snippets.

See

21.5.12.3.1 Conditional flow control
21.5.12.3.2 Transferring Flow Control

Or return to 21.5.12 Snippet Directives or 21.5 Defining and Using Snippets.

21.5.12.3.1 Conditional flow control

Snippet commands can be processed or ignored based on the result of a test.

This is known as a conditional test.

Snippet directives based on conditional statements (*i.e.* testing whether a statement is true or false) are generally of the form **if/else/end if**.

```
@ if….
@ end_if
```

OR

```
@ if…
@ else
@ end_if
```

When the **if** condition is **true**, commands will be processed until the **else** or **end_if** directive is found.

When the **if** condition is **false**, commands will be ignored **until** the **else** or **end_if** directive is found.

When the **if** condition is **false** and an **else** directive is found the commands following the **else** until the **end_if** will be processed.

An **end_if** directive must always be present.

An **else** directive cannot exist without a preceding **if** directive.

There is **no else_if** directive.

```
@ if...
// When the if condition is true, the commands in this section,
// until the else or end_if will be processed.
@ else
// When the if condition is false, the commands in this section,
// until the end_if will be processed
@ end_if
```

Embedding if directives within other conditional directives (if or else), *i.e.* nested ifs, are **not allowed**.

Continue to 21.5.12.3.2 Transferring Flow Control or return to 21.5.12.3 Flow control or 21.5.12 Snippet Directives.

21.5.12.3.2 Transferring Flow Control

In addition to the conditional flow control, it is possible within snippets to transfer processing to an entirely different and disconnected section of the snippet.

These transfers (jumps) are performed by the **goto** and **label** directives.

See

21.5.12.3.2.1 Goto - transfer processing to a given location.
21.5.12.3.2.2 Label - mark a location for goto

Or return to 21.5.12 Defining and Using Snippets.

21.5.12.3.2.1 Goto - transfer processing to a given location

The **goto** directive transfers processing to a corresponding **label** directive within the snippet.

The syntax is:

```
@ goto <label>
```

<label> - is the name of the label to which control will be transferred. One and only one label should be specified.

The **goto** directive must be defined with a corresponding **label** directive.

Control can only be transferred **forwards** in a file (i.e. further down the contents of the file); control **cannot** be passed **backwards**.

The corresponding **label** should **not** be inside a conditional flow group, if/else/end_if.

```
// PARAMETER KT TEXT "Kerb type" OPTIONAL

@ def_tok EXTRAS "absolute extra_start extra_end"

@ if_val_neq "$(KT)" "SA Kerb"
  @ goto fred
  user_message "Should not get here." $(_SCH) 0 $(_ECH) 0 $(EXTRAS)
@ end_if

user_message "Should not get here." $(_SCH) 0 $(_ECH) 0 $(EXTRAS)

@ label fred

user_message "Should get here." $(_SCH) 0 $(_ECH) 0 $(EXTRAS)
```

Continue to 21.5.12.3.2.2 Label - mark a location for goto or return to 21.5.12.3.1 Conditional flow control or 21.5.12.3 Flow control or 21.5.12 Snippet Directives.

21.5.12.3.2.2 Label - mark a location for goto

The **label** directive is used in conjunction with a **goto** directive to indicate a position within the snippet.

The syntax is:

```
@ label <label>
```

<label> - is the name of this label. A label should normally be paired with a corresponding **goto** directive, though it is not compulsory. A label can exist without a **goto**.

The **label** must be placed **after** the **goto** directive to which it corresponds. Control can only be transferred forwards in a file (i.e. further down the contents of the file); control cannot be passed backwards.

The **label** directive should not be placed inside a conditional flow group, if/else/end_if.

The label does not have to be unique within the file and multiple pairs of goto/labels are permitted. Note, however, that goto will simply jump to the first/next found matching label.

```
@ goto end_earthowrks
@ label end_earthowrks
.
@ goto end_earthowrks
@ label end_earthowrks
```

Continue to 21.5.12.4 Abbreviations or return to 21.5.12.3.1 Conditional flow control or 21.5.12.3 Flow control or 21.5.12 Snippet Directives.

21.5.12.4 Abbreviations

When reading or writing snippet directives, it is useful to know and remember the abbreviations below.

if - conditional test

def - define a token

tok - token

val - value

str - string (i.e. text)

int - integer

dbl - double

tol - tolerance

eq - equal

neq - not equal

gt - greater than

lt - less than

ge - greater than or equal to

le - less than or equal to

in_range - within a range of values defined by lower and upper bounds

nin_range - not within (i.e. outside) a range of values defined by lower and upper bounds

These abbreviations can be used to determine the appropriate directive to use or interpret and understand directives in use.

For example, the **if_val_le_dbl_tol**, is a conditional directive (**if**) to test if a value (**val**) is less than or equal to (**le**) any of double values (**dbl**) with a user-specified tolerance (**tol**).

Continue to 21.5.12.5 Comparisons of Data Types or return to 21.5.12 Snippet Directives.

21.5.12.5 Comparisons of Data Types

There are two basic types of comparison used in snippets.

(a) The **first** is **character-based** and is typically used on strings of text.

In a character-based comparison of two values, the first character from one string is compared against the first character from the other string. If both characters are equal, processing continues to the next character in each string.

Character comparisons in snippets are **not case sensitive**.

```
@ def_tok TOK1 "SOME TEXT"

@ if_tok_eq_str TOK1 "some text"
  user_message "This will be true because comparisons are case insensitive"
@ end_if
```

Character comparisons should not be used to compare two numbers. All directives including the **str** abbreviation, e.g. if_tok_eq_str, if_val_neq_str, use character-based comparisons.

```
@ def_tok TOK1 1

@ if_tok_eq_str TOK1 1.0
@ else
  user_message "False because if_tok_eq_str uses a character-based comparison"
@ end_if
```

(b) The **second** type of comparison is **numeric** and is typically used on numbers both integers and doubles.

In a numerical-based comparison of two values, each value is first converted into a double. If the value cannot be converted to a number, then an error is usually generated.

```
@ def_tok TOK1 1                     // This is a valid number
@ def_tok TOK2 2.345                 // As is this
@ def_tok TOK3 "1.78                 "// This can still be converted to a valid number
@ def_tok TOK4 "1e-4                 "// As can this
@ def_tok TOK5 "RL1.234"// But not this

@ if_tok_eq_dbl TOK5 "1.234"
  user_message "TOK5 is not a valid number"
@ else
  user_message "TOK5 cannot be converted to a valid number"
@ end_if
```

Assuming both values in a numerical comparison are valid numbers (or can be converted as such), the numbers are then compared in standard mathematical fashion.

```
@ def_tok TOK1 1

@ if_tok_eq_dbl TOK1 1.0
  user_message "True because if_tok_eq_dbl uses a numerical-based comparison"
@ end_if

@ if_tok_eq_dbl TOK1 "0.001e3"
  user_message "Also true because if_tok_eq_dbl uses a numerical-based comparison"
@ end_if
```

Both double and integer numbers can be used in directive comparisons for doubles (contain **dbl** abbreviation). However, only integer numbers can be used in directive comparisons for integers (contain **int** abbreviation). Doubles will not be converted or truncated to integers in such cases, but instead produce an error.

Continue to 21.5.12.6 Tokens and Tokens vs Tokens or return to 21.5.12 Snippet Directives.

21.5.12.6 Tokens and Tokens vs Tokens

The following Snippet Directives define tokens, check whether tokens are defined and compare tokens with other tokens.

See

Or return to 21.5 Defining and Using Snippets.

21.5.12.6.1 def_tok

The **def_tok** directive **defines** a **token** that can have an **optional value**.

The syntax is:

```
@ def_tok <token> [<string>... <string>]
```

<token> - this is the name of the token.

Wherever the token is to be used in the snippet, put **$(token)**

Note that in the older V10 syntax, this was $token.

The characters of the token name can only be alphanumeric (upper and lower case) and underscores but the name **cannot contain spaces**.

The **token names** are **not case sensitive**.

So two names differing only by case are considered to be identical. That is, **THIS**, **This**, **this** and **tHis** are all considered identical by ***12d Model***.

Tokens are treated similarly to parameters, so any name must be unique within a snippet for all parameters and tokens.

<string> - one or more optional strings of character (text).

Any string with spaces must be enclosed in quotes.

Multiple strings are concatenated.

The string can contain other tokens, though any tokens must have previously been defined.

The string can also contain mathematical expressions.

The snippet preprocessor will preserve any white space within quoted strings.

However, multiple strings or values will be concatenated without any white space.

If a string has spaces that need to be retained, enclose the string in quotes.

Tokens are used and substituted in the same way as parameters, *e.g.* $(TOKEN).

```
@ def_tok TOK          //defines a token TOK with no value
@ def_tok TOK A        //defines a token TOK with the value "A"
@ def_tok TOK "A B"    //defines a token TOK with the value "A B"
@ def_tok TOK A B      //defines a token TOK with the value "AB", concatenates
@ def_tok TOK A B "C D" //defines a token TOK with the value "ABC D", concatenates
@ def_tok TOK $(KT)    //defines a token TOK with the value of the substituted parameter
```

Continue to 21.5.12.6.2 if_tok or return to 21.5.12.6 Tokens and Tokens vs Tokens or 21.5.12 Snippet Directives.

21.5.12.6.2 if_tok

The **if_tok** directive tests whether **any** of the given **tokens** are **defined**.

The syntax is:

```
@ if_tok <token> [<token>... <token>]
// Commands to be processed when statement is true
@ end_if
```

<token> - the name of the token to test.

At least one token must be specified, however, multiple tokens can be included.

Multiple tokens should be separated by whitespace.

If any of the tokens are defined, the statement is true.

If none of the tokens are defined, the statement is false.

Multiple tokens are handled as if joined by logical OR statements.

NOTE: This directive simply tests whether the token is defined and not whether the value is valid or blank.

```
@ def_tok "T1"
@ def_tok "T2"

@ if_tok "T1"
  user_message "Found token T1"
@ end_if

@ if_tok "T1" "T2"
  user_message "Found token T1 or T2"
@ end_if

@ if_tok "T2"
  user_message "Found token T2"
@ end_if
```

See also:

21.5.12.6.3 if_ntok - for logically opposite directive

21.5.12.6.7 if_all_toks - for testing if all tokens are defined

Continue to 21.5.12.6.3 if_ntok or return to 21.5.12.6 Tokens and Tokens vs Tokens or 21.5.12 Snippet Directives.

21.5.12.6.3 if_ntok

The **if_ntok** directive tests whether **particular tokens** are **undefined**.

The syntax is:

```
@ if_ntok <token> [<token>... <token>]
// Commands to be processed when statement is true
@ end_if
```

<token> - the name of the token to test.

At least one token must be specified, however, multiple tokens can be included.

Multiple tokens should be separated by whitespace.

If **none** of the **tokens** are **defined**, the statement is **true** and the snippet commands until the next end_if directive are processed.

If **any** of the **tokens** are **defined**, the statement is **false** and the entire block of commands (until the next end_if directive) are skipped.

Multiple tokens are handled as if **joined** by **logical OR** statements.

NOTE: This directive simply tests whether the token is defined and not whether the value is valid or blank.

```
@ def_tok T1
@ def_tok T2

@ if_ntok T1
  user_message "Didn't find token T1"
@ end_if

@ if_ntok T1 T2
  user_message "Didn't find token T1 or T2"
@ end_if

@ if_ntok T2
  user_message "Didn't find token T2"
@ end_if
```

See also:

21.5.12.6.2 if_tok - logically opposite directive

21.5.12.6.8 if_nall_toks - for testing if all tokens are undefined

Continue to 21.5.12.6.4 if_tok_ntok or return to 21.5.12.6 Tokens and Tokens vs Tokens or 21.5.12 Snippet Directives.

21.5.12.6.4 if_tok_ntok

The **if_tok_ntok** directive tests if **any** token in **one set** is **defined** and if **any** token in **another set** is **not defined**.

The syntax is:

```
@ if_tok_ntok <token> [<token>... <token>] ; <token> [<token>... <token>]
```

<token> - the name of the token to test.

At least one token in each set must be specified, however, multiple tokens can be included.

Multiple tokens should be separated by whitespace.

Two sets of tokens are separated by a semi-colon, which must be included.

Tokens in the first set, between the **if_tok_ntok** and the semi-colon, are tested to see if they are defined. Tokens in the second set, between the semi-colon and end, are tested to see if they are not defined.

If any of the tokens in the first set are defined and any of the tokens in the second set are undefined, the statement is true.

If none of the tokens in the first set are defined or if all of the tokens in the second set are defined, the statement is false.

Note that both sets must be true for the overall statement to be true. If either set is false, then the overall statement will be false.

```
@ def_tok A
@ def_tok B
// token C is undefined
@ if_tok_ntok A ; C
  user_message "Token A or B are defined, but token C is undefined"
@ end_if

@ if_tok_ntok A ; B
  user_message "True: Token A is defined and Token B is undefined"
@ else
  user_message "False: Token A is defined, but token B is not undefined"
@ end_if

@ if_tok_ntok A B ; C
  user_message "True: Token A or B is defined and Token C is undefined"
@ end_if

@ if_tok_ntok C ; A
  // Won't be true in this example
@ else
  user_message "False: Token C is not defined or Token A is not undefined"
@ end_if
```

Continue to 21.5.12.6.5 if_two_toks or return to 21.5.12.6 Tokens and Tokens vs Tokens or 21.5.12 Snippet Directives.

21.5.12.6.5 if_two_toks

The **if_two_toks** directive tests the result of a logical comparison of if two tokens are defined.

The syntax is:

```
@ if_two_toks <token1> <op> <token2>
// Commands to be processed when statement is true
@ end_if
```

<token1> - the name of the token to test. Must be only one token

<op> - the logical operator to use for the two tokens. This can be **&&** for logical AND or **||** for logical OR

<token2> - the name of the token to test. Must be only one token.

The conditional result of this directive is dependent on the two tokens and the logical operator. The combination of possible values and their results are in the table below.

token1	op	token2	Result
defined	&&	defined	true
defined	&&	undefined	false
undefined	&&	defined	false
undefined	&&	undefined	false
defined	||	defined	true
defined	||	undefined	true
undefined	||	defined	true
undefined	||	undefined	false

```
@ def_tok T1
@ def_tok T2
// Token T3 is not defined

@ if_two_toks T1 && T2
  user_message "Tokens T1 and T2 are both defined"
@ end_if

@ if_two_toks T1 || T2
  user_message "Token T1 or T2 is defined"
@ end_if

@ if_two_toks T1 && T3
  user_message "Token T1 and T3 are both defined"// This shouldn't be shown
@ else
  user_message "Token T1 and T3 are not both defined" // This should be shown
@ end_if

@ if_two_toks T1 || T3
  user_message "Token T1 or T3 is defined"
@ end_if
```

See also:

21.5.12.6.6 if_two_toks_eq - for logical comparison against values

Continue to 21.5.12.6.6 if_two_toks_eq or return to 21.5.12.6 Tokens and Tokens vs Tokens or 21.5.12 Snippet Directives.

21.5.12.6.6 if_two_toks_eq

The **if_two_toks_eq** directive tests the result of a two tests of tokens against corresponding values.

The syntax is:

```
@ if_two_toks_eq <token1> <value1> <op> <token2> <value2>
// Commands to be processed when statement is true
@ end_if
```

<token1> - the name of the first token to test. Must be only one token.

<value1> - the value to compare against token1.

<op> - the logical operator to use for the two tokens.

This can be && for logical AND or || for logical OR

<token2> - the name of the second token to test. Must be only one token.

<value2> - the value to compare against token2.

When evaluating the statement, token1 is compared against value1 and token2 is compared against value2. The results of both comparisons are then combined using the logical operator, **op**, and evaluated to set the final result of the if_two_toks_eq directive.

token1	value1	token1 = value1	op	token2 = value2	token2	value2	Overall Result
A	A	true	&&	true	B	B	true
A	C	false	&&	true	B	B	false
A	A	true	&&	false	B	D	false
A	C	false	&&	false	B	D	false
A	A	true	||	true	B	B	true
A	C	false	||	true	B	B	true
A	A	true	||	false	B	D	true
A	C	false	||	false	B	D	false

```
@ def_tok TOK_A "alpha"
@ def_tok TOK_B "bravo"

@ if_two_toks_eq TOK_A "alpha" || TOK_B "bravo"
  user_message "TOK_A = alpha or TOK_B = bravo"
@ end_if

@ if_two_toks_eq TOK_A "charlie" && TOK_B "bravo"
  user_message "TOK_A = charlie and TOK_B = bravo" // This should not be output
@ else
  user_message "TOK_A != charlie or TOK_B != bravo" // This should be output
@ end_if
```

See also:

21.5.12.6.5 if_two_toks - testing if two tokens are defined with logical operator

Continue to 21.5.12.6.7 if_all_toks or return to 21.5.12.6 Tokens and Tokens vs Tokens or 21.5.12 Snippet Directives.

21.5.12.6.7 if_all_toks

The **if_all_toks** directive tests whether all given tokens are defined.

The syntax is:

```
@ if_all_toks <token> [<token>... <token>]
// Commands to be processed when statement is true
@ end_if
```

<token> - the name of the token to test.

At least one token must be specified, however, multiple tokens can be included.

Multiple tokens should be separated by whitespace.

If all of the tokens are defined, the statement is true.

If any of the tokens are undefined, the statement is false.

Multiple tokens are handled as if joined by logical AND statements.

NOTE: This directive simply tests whether the token is defined and not whether the value is valid or blank.

```
@ def_tok T1
@ def_tok T2
@ def_tok T3

@ if_all_toks T1
  user_message "Found token T1"
@ end_if

@ if_all_toks T1 T2
  user_message "Found token T1 and T2"
@ end_if

@ if_all_toks T1 T2 T3
  user_message "Found token T1, T2 and T3"
@ end_if
```

See also:

21.5.12.6.8 if_nall_toks - logically opposite directive

21.5.12.6.2 if_tok - test if any given tokens are defined

Continue to 21.5.12.6.8 if_nall_toks or return to 21.5.12.6 Tokens and Tokens vs Tokens or 21.5.12 Snippet Directives.

21.5.12.6.8 if_nall_toks

The **if_nall_toks** directive tests whether all of the given tokens are undefined.

The syntax is:

```
@ if_nall_toks <token> [<token>... <token>]
// Commands to be processed when statement is true
@ end_if
```

<token> - the name of the token to test.

At least one token must be specified, however, multiple tokens can be included. Multiple tokens should be separated by whitespace.

If all of the tokens are undefined, the statement is true.

If any of the tokens are defined, the statement is false.

Multiple tokens are handled as if joined by logical AND statements.

NOTE: This directive simply tests whether the token is defined and not whether the value is valid or blank.

```
@ def_tok T1
@ def_tok T2
@ def_tok T3

@ if_nall_toks T1
  user_message "Didn't find token T1"
@ end_if

@ if_nall_toks T1 T2
  user_message "Didn't find tokens T1 or T2"
@ end_if

@ if_nall_toks T1 T2 T3
  user_message "Didn't find tokens T1, T2 or T3"
@ end_if
```

See also:

21.5.12.6.7 if_all_toks - logically opposite directive

21.5.12.6.3 if_ntok - testing if any given tokens are undefined

Continue to 21.5.12.6.9 if_tok_eq_tok or return to 21.5.12.6 Tokens and Tokens vs Tokens or 21.5.12 Snippet Directives.

21.5.12.6.9 if_tok_eq_tok

The **if_tok_eq_tok** directive tests whether a token is equal to another token.

The syntax is:

```
@ if_tok_eq_tok <token1> <token2>
// Commands to be processed when statement is true
@ end_if
```

<token1> - the first token to be compared.

<token2> - the second token to be compared.

token1 and **token2** are compared using a character comparison (*i.e.* text-based, not numeric).

If the token1 and token2 values are equal, the statement is true.

If the token1 and token2 values do not match, the statement is false.

If either or both token1 or token2 are undefined, the statement is false.

```
@ def_tok TOK1 "alpha"
@ def_tok TOK2 "bravo"

@ if_tok_eq TOK1 TOK2
  user_message "Tokens 1 and 2 are equal"
@ else
  user_message "Tokens 1 and 2 are not equal or undefined"
@ end_if
```

See also:

21.5.12.6.10 if_tok_neq_tok - logically opposite directive

if_tok_eq_str - testing against a string

if_tok_eq_dbl - testing against a double

if_tok_eq_int - testing against an integer

Continue to 21.5.12.6.10 if_tok_neq_tok or return to 21.5.12.6 Tokens and Tokens vs Tokens or 21.5.12 Snippet Directives.

21.5.12.6.10 if_tok_neq_tok

The **if_tok_neq_tok** directive tests whether a token is not equal to another token.

The syntax is:

```
@ if_tok_neq_tok <token1> <token2>
// Commands to be processed when statement is true
@ end_if
```

<token1> - the first token to be compared.

<token2> - the second token to be compared.

Token1 and **token2** are compared using a character comparison (i.e. text-based, not numeric).

If token1 and token2 values do not match, the statement is true.

If token1 and token2 values match, the statement is false.

If either token1 or token2 is undefined, an error message is produced and the statement is evaluated as true.

```
@ def_tok TOK1 "alpha"
@ def_tok TOK2 "bravo"

@ if_tok_neq_tok TOK1 TOK2
  user_message "Tokens 1 and 2 are not equal"
@ else
  user_message "Tokens 1 and 2 are equal"
@ end_if
```

See also:

21.5.12.6.9 if_tok_eq_tok - logically opposite directive

if_tok_neq_str - comparison against string

if_tok_neq_dbl - comparison against double

if_tok_neq_int - comparison against integer

Continue to 21.5.12.6.11 def_tok_minus or return to 21.5.12.6 Tokens and Tokens vs Tokens or 21.5.12 Snippet Directives.

21.5.12.6.11 def_tok_minus

The **def_tok_minus** directive defines a token as the result of a single digit or character subtracted from another.

The syntax is:

```
@ def_tok_minus <token> <value1> <value2>
```

<token> - the name of the token to define.

<value1> - a **single** integer or character or parameter that evaluates to such

<value2> - a **single** integer or character or parameter that evaluates to such

The value of token is taken as the result of subtracting value2 from value1.

To determine the value of <token>, **value1** and **value2** are converted to their equivalent numerical ASCII character code (e.g. 0 = 48, 9 = 57, A = 65, Z = 90, a = 97, z = 122).

The ASCII character code for value2 is subtracted from the ASCII character code for value1. The result is set as the value of <token>.

Example with integers:

Raw snippet values: <value1> = 5, <value2> = 2

Equivalent ASCII character codes:<value1> = 53, <value2> = 50

Token value = 53 - 50 = 3

Example with letters:

Raw snippet values: <value1> = D, <value2> = B

Equivalent ASCII character codes:<value1> = 68, <value2> = 66

Token value = 68 - 66 = 2

For single-digit integers, this is the same as the basic arithmetic on the integers.

For this reason, it is recommended that <value1> and <value2> are of the same type and case. That is, both values are:

- *integers (0-9)*
- *upper-case single letters (A-Z); or*
- *lower-case single letters (a-z).*

Mixing value types - *e.g.* integer and letter - will likely result in unexpected behaviour.

It is possible for <value1> or <value2> to contain parameters, *e.g.* "$(WIDTH)", so long as the parameters evaluate to a single digit or character.

It is not possible to use tokens in either <value1> or <value2> due to the order in which snippets are processed and tokens substituted. For more information see 21.5.14 Order of Snippet Processing.

If value1 or value2 is not a single character, an error message will be produced.

```
// PARAMETER LANE1 INTEGER "Start lane" 5
// PARAMETER LANE2 INTEGER "End lane" 2

// ASCII character codes:
// 5 = 53
// 2 = 50

@ def_tok_minus NUM_LANES $(LANE1) $(LANE2)
// For LANE1 = 5, LANE2 = C, NUM_LANES = 53 - 50 = 3

user_message "Number of lanes between start and end = $(NUM_LANES)"
```

```
// PARAMETER LANE1 TEXT "Start lane character (A-Z)" "F"
// PARAMETER LANE2 TEXT "End lane character (A-Z)" "C"

// ASCII character codes:
// F = 70
// C = 67

@ def_tok_minus NUM_LANES $(LANE1) $(LANE2)
// For LANE1 = F and LANE2 = C, NUM_LANES = 70 - 67 = 3

user_message "Number of lanes between start and end = $(NUM_LANES)"
```

TIP - To ensure value1 and value2 are single characters, use the substring functionality to only use the first character.

```
// PARAMETER LANE1 TEXT "Start lane character (A-Z)" "FAST"
// PARAMETER LANE2 TEXT "End lane character (A-Z)" "CAR"

// Using substring functionality to only use the 1st character of each parameter
@ def_tok_minus NUM_LANES $(LANE1[1:1]) $(LANE2[1:1])

user_message "Number of lanes between start and end = $(NUM_LANES)"
```

Continue to 21.5.12.6.12 def_tok_minus_inc or return to 21.5.12.6 Tokens and Tokens vs Tokens or 21.5.12 Snippet Directives.

21.5.12.6.12 def_tok_minus_inc

The **def_tok_minus_inc** defines a token with the result of a single digit or single-character subtracted from another, plus one.

That is, it is similar to **def_tok_minus**, but the result is one more than the equivalent def_tok_minus value (i.e. it is an inclusive subtraction).

The syntax is:

```
@ def_tok_minus_inc <token> <value1> <value2>
```

<token> - the name of the token to define.

<value1> - a single integer or character or parameter that evaluates to such

<value2> - a single integer or character or parameter that evaluates to such

The value of token is taken as the result of subtracting value2 from value1 then adding one.

To determine the value of <token>, value1 and value2 are converted to their equivalent numerical ASCII character code (e.g. 0 = 48, 9 = 57, A = 65, Z = 90, a = 97, z = 122).

The ASCII character code for value2 is subtracted from the ASCII character code for value1 and one (1) is added. The result is set as the value of <token>.

Example with integers:

Raw snippet values: <value1> = 5, <value2> = 2

Equivalent ASCII character codes:<value1> = 53, <value2> = 50

Token value = 53 - 50 + 1 = 4

Example with letters:

Raw snippet values: <value1> = D, <value2> = B

Equivalent ASCII character codes:<value1> = 68, <value2> = 66

Token value = 68 - 66 + 1 = 3

For single-character integers, this is the same as the basic arithmetic on the integers.

For this reason, it is recommended that <value1> and <value2> are of the same type and case. That is, both values are:

- *integers (0-9)*
- *upper-case single letters (A-Z); or*
- *lower-case single letters (a-z).*

Mixing value types - *e.g.* integer and letter - will likely result in unexpected behaviour.

It is possible for <value1> or <value2> to contain parameters, *e.g.* "$(WIDTH)", so long as the parameters evaluate to a single character.

It is not possible to use tokens in either <value1> or <value2> due to the order in which snippets are processed and tokens substituted. For more information see 21.5.14 Order of Snippet Processing.

If value1 or value2 is not a single character, an error message will be produced.

```
// PARAMETER LANE1 INTEGER "Start lane" 5
// PARAMETER LANE2 INTEGER "End lane" 2

// ASCII character codes:
// 5 = 53
// 2 = 50

@ def_tok_minus_inc NUM_LANES $(LANE1) $(LANE2)
// For LANE1 = 5, LANE2 = C, NUM_LANES = 53 - 50 + 1 = 4

user_message "Number of lanes between start and end (inclusive) = $(NUM_LANES)"
```

```
// PARAMETER LANE1 TEXT "Start lane character (A-Z)" "F"
// PARAMETER LANE2 TEXT "End lane character (A-Z)" "C"

// ASCII character codes:
// F = 70
// C = 67

@ def_tok_minus NUM_LANES $(LANE1) $(LANE2)
// For LANE1 = F and LANE2 = C, NUM_LANES = 70 - 67 + 1 = 4

user_message "Number of lanes between start and end (inclusive) = $(NUM_LANES)"
```

TIP - To ensure value1 and value2 are single characters, use the substring functionality to only use the first character.

```
// PARAMETER LANE1 TEXT "Start lane character (A-Z)" "FAST"
// PARAMETER LANE2 TEXT "End lane character (A-Z)" "CAR"

// Using substring functionality to only use the 1st character of each parameter
@ def_tok_minus NUM_LANES $(LANE1[1:1]) $(LANE2[1:1])

user_message "Number of lanes between start and end (inclusive) = $(NUM_LANES)"
```

Continue to 21.5.12.7 Tokens vs Strings or return to 21.5.12.6 Tokens and Tokens vs Tokens or 21.5.12 Snippet Directives.

21.5.12.7 Tokens vs Strings

The following directives compare a token value against given strings.

See

21.5.12.7.1 if_tok_eq_str
21.5.12.7.2 if_tok_neq_str

21.5.12.7.1 if_tok_eq_str

The **if_tok_eq_str** directive tests if a token's **value** is **equal** to **any** given **strings**.

The syntax is:

```
@ if_tok_eq_str <token> <string> [<string>... <string>]
// Commands to be processed if statement is true
@ end_if
```

<token> - the name of the token to compare.

<string> - the string to compare against the value of token.

At least one string must be specified; however, multiple strings can be included.

Multiple strings should be separated by whitespace. Strings should be enclosed in quotes.

If the **value** of **token** is **equal** to **any** of the **strings**, the statement **is** true.

If the value of token is not equal to any of the strings (i.e. the token value is equal to none of the given strings), the statement is false.

Multiple strings are handled as if joined by logical OR statements.

```
@ def_tok A_TOKEN "a token"
@ def_tok B_TOKEN "b token"

@ if_tok_eq_str A_TOKEN "a token" "b token" // must equal one of the list, should pass
  user_message "if_tok_eq_str = TRUE , A = a token OR b token"

@ if_tok_eq_str A_TOKEN "x token" "y token" // must equal one of the list, should fail
@ else
  user_message "if_tok_eq_str = FALSE , A != x token OR y token"
@ end_if
```

Continue to 21.5.12.7.2 if_tok_neq_str or return to 21.5.12.7 Tokens vs Strings or 21.5.12 Snippet Directives.

21.5.12.7.2 if_tok_neq_str

The **if_tok_neq_str** directive tests if a token is **not equal** to **any** of the given **strings**.

The syntax is:

```
@ if_tok_neq_str <token> <string> [<string>... <string>]
// Commands to be processed if statement is true
@ end_if
```

<token> - the name of the token to compare.

<string> - the string to compare against the value of token.

At least one string must be specified, however, multiple strings can be included. Multiple strings should be separated by whitespace. Strings should be enclosed in quotes.

If the value of token is not equal to any of the strings, the statement is true.

If the value of token is equal to any of the string values, the statement is false.

Multiple strings are handled as if joined by logical OR statements.

```
@ def_tok A_TOKEN "a token"
@ def_tok B_TOKEN "b token"

@ if_tok_eq_str A_TOKEN "a token" "b token" // must equal one of the list, should pass
  user_message "if_tok_eq_str = TRUE , A = a token OR b token"

@ if_tok_eq_str A_TOKEN "x token" "y token" // must equal one of the list, should fail
@ else
  user_message "if_tok_eq_str = FALSE , A != x token OR y token"
@ end_if
```

Continue to 21.5.12.8 Tokens vs Doubles or return to 21.5.12.7 Tokens vs Strings or 21.5.12 Snippet Directives.

21.5.12.8 Tokens vs Doubles

The following directives compare a token value against given doubles.

In these directives, **<double>** can be either a real number or an evaluation of a previously defined token. For example $(T49).

See

21.5.12.8.1 if_tok_eq_dbl
21.5.12.8.2 if_tok_eq_dbl_tol
21.5.12.8.3 if_tok_neq_dbl
21.5.12.8.4 if_tok_neq_dbl_tol
21.5.12.8.5 if_tok_lt_dbl
21.5.12.8.6 if_tok_lt_dbl_tol
21.5.12.8.7 if_tok_le_dbl
21.5.12.8.8 if_tok_le_dbl_tol
21.5.12.8.9 if_tok_gt_dbl
21.5.12.8.10 if_tok_gt_dbl_tol
21.5.12.8.11 if_tok_ge_dbl
21.5.12.8.12 if_tok_le_dbl_tol
21.5.12.8.13 if_tok_gt_dbl
21.5.12.8.14 if_tok_gt_dbl_tol
21.5.12.8.15 if_tok_ge_dbl
21.5.12.8.16 if_tok_ge_dbl_tol
21.5.12.8.17 if_tok_in_range_dbl
21.5.12.8.18 if_tok_in_range_dbl_tol
21.5.12.8.19 if_tok_nin_range_dbl
21.5.12.8.20 if_tok_nin_range_dbl_tol

Or return to 21.5 Defining and Using Snippets.

21.5.12.8.1 if_tok_eq_dbl

The **if_tok_eq_dbl** directive tests if a token's **value** is **equal** to **any** given **double** values, within a fixed tolerance of 1.0x10-6 (i.e. 1.0E-6).

The syntax is:

```
@ if_tok_eq_dbl <token> <double> [<double>... <double>]
// Commands processed if statement is true
@ end_if
```

<token> - the name of the token to test.

<double> - the double value to compare against the value of token.

double can be a real number or the evaluation of a token eg $(T49).

At least one double value must be specified, however, multiple double values can be included.

Multiple double values should be separated by whitespace.

If the value of token is equal to any of the doubles within the tolerance, the statement is true.

If the value of token is not equal to any of the doubles within the tolerance, the statement is false.

If the token is undefined, the snippet will produce an error.

Multiple double values are handled as if joined by logical OR statements.

```
@ def_tok WIDTH 3.5

@ if_tok_eq_dbl WIDTH 3.0 3.5 4.0
  user_message "WIDTH is equal to 3.0, 3.5 or 4.0 within a tolerance of 1e-6"
@ end_if
```

Continue to 21.5.12.8.2 if_tok_eq_dbl_tol or return to 21.5.12.8 Tokens vs Doubles or 21.5.12 Snippet Directives.

21.5.12.8.2 if_tok_eq_dbl_tol

The **if_tok_eq_dbl_tol** directive tests if a token's **value** is **equal** to any given **double** value, within a given tolerance.

The syntax is:

```
@ if_tok_eq_dbl_tol <token> <double> [<double>... <double>] <tolerance>
// Commands processed if statement is true
@ end_if
```

<token> - the name of the token to test.

<double> - the double value to compare against the value of token.

At least one double value must be specified, however, multiple double values can be included. Multiple double values should be separated by whitespace.

Double values should not be quoted.

<tolerance> - the tolerance, expressed as a valid double value, for comparison.

This value must be specified and must be the last value.

If the value of token is equal to any of the doubles within the tolerance, the statement is true.

If the value of token is not equal to any of the doubles within the tolerance, the statement is false.

If the token is undefined, the snippet will produce an error.

Multiple double values are handled as if joined by logical OR statements.

```
@ def_tok WIDTH 3.5

@ if_tok_eq_dbl_tol WIDTH 3.5 0.1
  user_message "WIDTH is equal to 3.5 within a tolerance of 0.1"
@ end_if
```

Continue to 21.5.12.8.3 if_tok_neq_dbl or return to 21.5.12.8 Tokens vs Doubles or 21.5.12 Snippet Directives.

21.5.12.8.3 if_tok_neq_dbl

The **if_tok_neq_dbl** directive tests if a token's **value** is **not equal** to any of the given **doubles**, within a fixed tolerance of 1.0x10-6 (*i.e.* 1.0E-6).

The syntax is:

```
@ if_tok_neq_dbl <token> <double> [<double>... <double>]
// Commands processed if statement is true
@ end_if
```

<token> - the name of the token to test.

<double> - the double value to compare against the value of token.

At least one double value must be specified, however, multiple double values can be included.

Multiple double values should be separated by whitespace.

Double values should not be quoted.

If the value of token is not equal to any of the doubles within the tolerance, the statement is true.

If the value of token is equal to any of the doubles within the tolerance, the statement is false.

If the token is undefined, the snippet will produce an error.

Multiple double values are handled as if joined by logical OR statements.

```
@ def_tok WIDTH 3.5

@ if_tok_neq_dbl WIDTH 3.0
  user_message "WIDTH is not equal to 3.0 within a tolerance of 1e-6"
@ end_if
```

Continue to 21.5.12.8.4 if_tok_neq_dbl_tol or return to 21.5.12.8 Tokens vs Doubles or 21.5.12 Snippet Directives.

21.5.12.8.4 if_tok_neq_dbl_tol

The **if_tok_neq_dbl_tol** directive tests if a token's **value** is **not equal** to any given **double** values, within a given tolerance.

The syntax is:

```
@ if_tok_neq_dbl_tol <token> <double> [<double>... <double>] <tolerance>
// Commands processed if statement is true
@ end_if
```

<token> - the name of the token to test.

<double> - the double value to compare against the value of token.

At least one double value must be specified, however, multiple double values can be included.

Multiple double values should be separated by whitespace.

Double values should not be quoted.

<tolerance> - the tolerance, expressed as a valid double value, for comparison.

This value must be specified and must be the last value.

If the value of token is not equal to any of the double values within the tolerance, the statement is true.

If the value of token is equal to any of the double values within the tolerance, the statement is false.

If the token is undefined, the snippet will produce an error.

Multiple double values are handled as if joined by logical OR statements.

```
@ def_tok WIDTH 3.5

@ if_tok_neq_dbl_tol WIDTH 3.0 0.1
  user_message "WIDTH is not equal to 3.0 within a tolerance of 0.1"
@ end_if
```

Continue to 21.5.12.8.5 if_tok_lt_dbl or return to 21.5.12.8 Tokens vs Doubles or 21.5.12 Snippet Directives.

21.5.12.8.5 if_tok_lt_dbl

The **if_tok_lt_dbl** directive tests if a token is less than any given doubles, within a fixed tolerance of 1.0 x 10-6 (1.0E-6).

The syntax is:

```
@ if_tok_lt_dbl <token> <double> [<double>... <double>]
// Commands processed if true
@ end_if
```

<token> - the name of the token to test.

<double> - the double value to compare against the value of token. At least one double value must be specified, however, multiple double values can be included. Multiple double values should be separated by whitespace. Double values should not be quoted.

If the token value is less than any of the double values within the tolerance, the statement is true.

If the token value is not less than any of the double values within the tolerance (i.e. the token value is greater than or equal to all double values), the statement is false.

If the token is undefined, the snippet will produce an error.

Multiple doubles are handled as if joined by logical OR statements.

```
@ def_tok WIDTH 3.5

@ if_tok_lt_dbl WIDTH 4.0
  user_message "WIDTH is less than 4.0 within a tolerance of 1e-6"
@ end_if
```

Continue to 21.5.12.8.6 if_tok_lt_dbl_tol or return to 21.5.12.8 Tokens vs Doubles or 21.5.12 Snippet Directives.

21.5.12.8.6 if_tok_lt_dbl_tol

The **if_tok_lt_dbl_tol** directive tests if a token is less than any given doubles, within a given tolerance. The comparison is numerical.

The syntax is
:

```
@ if_tok_lt_dbl_tol <token> <double> [<double>... <double>] <tolerance>
// Commands processed if true
@ end_if
```

<token> - the name of the token to test.

<double> - the double value to compare against the value of token. At least one double value must be specified, however, multiple double values can be included. Multiple double values should be separated by whitespace. Double values should not be quoted.

<tolerance> - the tolerance, expressed as a valid double value, for comparison. This value must be specified and must be the last value.

If the token value is less than any of the double values within the tolerance, the statement is true.

If the token value is not less than any of the double values within the tolerance (i.e. the token is greater than or equal to all double values), the statement is false.

If the token is undefined, the snippet will produce an error.

Multiple doubles are handled as if joined by logical OR statements.

```
@ def_tok WIDTH 3.5

@ if_tok_lt_dbl_tol WIDTH 4.0 0.1
  user_message "WIDTH is less than 4.0 within a tolerance of 0.1"
@ end_if
```

Continue to 21.5.12.8.7 if_tok_le_dbl or return to 21.5.12.8 Tokens vs Doubles or 21.5.12 Snippet Directives.

21.5.12.8.7 if_tok_le_dbl

The **if_tok_le_dbl** directive tests if a token is less than or equal to any given doubles, within a fixed tolerance of 1.0 x 10-6 (1.0E-6).

The syntax is:

:

```
@ if_tok_le_dbl <token> <double> [<double>... <double>]
// Commands processed if true
@ end_if
```

<token> - the name of the token to test.

<double> - the double value to compare against the value of token. At least one double value must be specified, however, multiple double values can be included. Multiple double values should be separated by whitespace. Double values should not be quoted.

If the token value is less than or equal to any of the double values within the tolerance, the statement is true.

If the token value is not less than or equal to any of the double values within the tolerance, the statement is false.

If the token is undefined, the snippet will produce an error.

Multiple doubles are handled as if joined by logical OR statements.

```
@ def_tok WIDTH 3.5

@ if_tok_le_dbl WIDTH 3.5
  user_message "WIDTH is less than or equal to 3.5 within a tolerance of 1e-6"
@ end_if

@ if_tok_le_dbl WIDTH 3.5 4.0
  user_message "WIDTH is less than or equal to 3.5 or 4.0 within a tolerance of 1e-6"
@ end_if
```

Continue to 21.5.12.8.8 if_tok_le_dbl_tol or return to 21.5.12.8 Tokens vs Doubles or 21.5.12 Snippet Directives.

21.5.12.8.8 if_tok_le_dbl_tol

The if_tok_le_dbl_tol directive tests if a token is less than or equal to any given double values, within a given tolerance.

The syntax is:
:

```
@ if_tok_le_dbl_tol <token> <double> [<double>... <double>] <tolerance>
// Commands processed if true
@ end_if
```

<token> - the name of the token to test.

<double> - the double value to compare against the value of token. At least one double value must be specified, however, multiple double values can be included. Multiple double values should be separated by whitespace. Double values should not be quoted.

<tolerance> - the tolerance, expressed as a valid double value, for comparison. This value must be specified and must be the last value.

If the token value is less than or equal to any of the double values within the tolerance, the statement is true.

If the token value is not less than or equal to any of the double values within the tolerance, the statement is false.

If the token is undefined, the snippet will produce an error.

Multiple doubles are handled as if joined by logical OR statements.

```
@ def_tok WIDTH 3.5

@ if_tok_le_dbl_tol WIDTH 3.5 0.1
  user_message "WIDTH is less than or equal to 3.5 within a tolerance of 0.1"
@ end_if

@ if_tok_le_dbl WIDTH 3.5 4.0 0.1
  user_message "WIDTH is less than or equal to 3.5 or 4.0 within a tolerance of 0.1"
@ end_if
```

Continue to 21.5.12.8.9 if_tok_gt_dbl or return to 21.5.12.8 Tokens vs Doubles or 21.5.12 Snippet Directives.

21.5.12.8.9 if_tok_gt_dbl

The **if_tok_gt_dbl** directive tests if a token is greater than any given doubles, within a fixed tolerance of 1.0 x 10-6 (1.0E-6).

The syntax is:

:

```
@ if_tok_gt_dbl <token> <double> [<double>... <double>]
// Commands processed if true
@ end_if
```

<token> - the name of the token to test.

<double> - the double value to compare against the value of token. At least one double value must be specified, however, multiple double values can be included. Multiple double values should be separated by whitespace. Double values should not be quoted.

If the token value is greater than any of the double values within the tolerance, the statement is true.

If the token value is not greater than any of the double values within the tolerance (i.e. the token value is less than or equal to all double values), the statement is false.

If the token is undefined, the snippet will produce an error.

Multiple doubles are handled as if joined by logical OR statements.

```
@ def_tok WIDTH 3.5

@ if_tok_gt_dbl WIDTH 3.0
  user_message "WIDTH is greater than 3.0 within a tolerance of 1e-6"
@ end_if

@ if_tok_gt_dbl WIDTH 3.0 4.0
  user_message "WIDTH is greater than 3.0 or 4.0 within a tolerance of 1e-6"
@ end_if
```

Continue to 21.5.12.8.10 if_tok_gt_dbl_tol or return to 21.5.12.8 Tokens vs Doubles or 21.5.12 Snippet Directives.

21.5.12.8.10 if_tok_gt_dbl_tol

The **if_tok_gt_dbl_tol** directive tests if a token is greater than any given double values, within a given tolerance.

The syntax is:

:

```
@ if_tok_gt_dbl_tol <token> <double> [<double>... <double>] <tolerance>
// Commands processed if true
@ end_if
```

<token> - the name of the token to test.

<double> - the double value to compare against the value of token.

At least one double value must be specified, however, multiple double values can be included.

Multiple double values should be separated by whitespace.

Double values should not be quoted.

<tolerance> - the tolerance, expressed as a valid double value, for comparison. This value must be specified and must be the last value.

If the token value is greater than any of the double values within the tolerance, the statement is true.

If the token value is not greater than any of the double values within the tolerance (*i.e.* the token is less than or equal to all double values), the statement is false.

If the token is undefined, the snippet will produce an error.

Multiple doubles are handled as if joined by logical OR statements.

```
@ def_tok WIDTH 3.5

@ if_tok_gt_dbl_tol WIDTH 3.0 0.1
  user_message "WIDTH is greater than 3.5 within a tolerance of 0.1"
@ end_if

@ if_tok_gt_dbl_tol WIDTH 3.0 3.5
  user_message "WIDTH is greater than 3.0 or 3.5 within a tolerance of 0.1"
@ end_if
```

Continue to 21.5.12.8.11 if_tok_ge_dbl or return to 21.5.12.8 Tokens vs Doubles or 21.5.12 Snippet Directives.

21.5.12.8.11 if_tok_ge_dbl

The **if_tok_ge_dbl** directive tests if a token is greater than or equal to any given double values, within a fixed tolerance of 1.0 x 10-6 (1.0E-6).

:

```
@ if_tok_ge_dbl <token> <double> [<double>... <double>]
// Commands processed if true
@ end_if
```

<token> - the name of the token to test.

<double> - the double value to compare against the value of token. At least one double value must be specified, however, multiple double values can be included. Multiple double values should be separated by whitespace. Double values should not be quoted.

If the token value is greater than or equal to any of the double values within the tolerance, the statement is true.

If the token value is not greater than or equal to any of the double values within the tolerance (i.e. the token value is less than all double values), the statement is false.

If the token is undefined, the snippet will produce an error.

Multiple doubles are handled as if joined by logical OR statements.

```
@ def_tok WIDTH 3.5

@ if_tok_le_dbl WIDTH 3.5
 user_message "WIDTH is less than or equal to 3.5 within a tolerance of 1e-6"
@ end_if
@ if_tok_le_dbl WIDTH 3.5 4.0
 user_message "WIDTH is less than or equal to 3.5 or 4.0 within a tolerance of 1e-6"
@ end_if
```

Continue to 21.5.12.8.12 if_tok_le_dbl_tol or return to 21.5.12.8 Tokens vs Doubles or 21.5.12 Snippet Directives.

21.5.12.8.12 if_tok_le_dbl_tol

The **if_tok_le_dbl_tol** directive tests if a token is less than or equal to any given double values, within a given tolerance.

The syntax is:

```
@ if_tok_le_dbl_tol <token> <double> [<double> ... <double>] <tolerance>
// Commands processed if true
@ end_if
```

<token> - the name of the token to test.

<double> - the double value to compare against the value of token. At least one double value must be specified, however, multiple double values can be included. Multiple double values should be separated by whitespace. Double values should not be quoted.

<tolerance> - the tolerance, expressed as a valid double value, for comparison. This value must be specified and must be the last value.

If the token value is less than or equal to any of the double values within the tolerance, the statement is true.

If the token value is not less than or equal to any of the double values within the tolerance, the statement is false.

If the token is undefined, the snippet will produce an error.

Multiple doubles are handled as if joined by logical OR statements.

```
@ def_tok WIDTH 3.5

@ if_tok_le_dbl_tol WIDTH 3.5 0.1
  user_message "WIDTH is less than or equal to 3.5 within a tolerance of 0.1"
@ end_if

@ if_tok_le_dbl WIDTH 3.5 4.0 0.1
  user_message "WIDTH is less than or equal to 3.5 or 4.0 within a tolerance of 0.1"
@ end_if
```

Continue to 21.5.12.8.13 if_tok_gt_dbl or return to 21.5.12.8 Tokens vs Doubles or 21.5.12 Snippet Directives.

21.5.12.8.13 if_tok_gt_dbl

The **if_tok_gt_dbl** directive tests if a token is greater than any given doubles, within a fixed tolerance of 1.0 x 10-6 (1.0E-6).

The syntax is:

```
@ if_tok_gt_dbl <token> <double> [<double>... <double>]
// Commands processed if true
@ end_if
```

<token> - the name of the token to test.

<double> - the double value to compare against the value of token. At least one double value must be specified, however, multiple double values can be included. Multiple double values should be separated by whitespace. Double values should not be quoted.

If the token value is greater than any of the double values within the tolerance, the statement is true.

If the token value is not greater than any of the double values within the tolerance (i.e. the token value is less than or equal to all double values), the statement is false.

If the token is undefined, the snippet will produce an error.

Multiple doubles are handled as if joined by logical OR statements.

```
@ def_tok WIDTH 3.5

@ if_tok_gt_dbl WIDTH 3.0
  user_message "WIDTH is greater than 3.0 within a tolerance of 1e-6"
@ end_if

@ if_tok_gt_dbl WIDTH 3.0 4.0
  user_message "WIDTH is greater than 3.0 or 4.0 within a tolerance of 1e-6"
@ end_if
```

Continue to 21.5.12.8.14 if_tok_gt_dbl_tol or return to 21.5.12.8 Tokens vs Doubles or 21.5.12 Snippet Directives.

21.5.12.8.14 if_tok_gt_dbl_tol

The **if_tok_gt_dbl_tol** directive tests if a token is greater than any given double values, within a given tolerance.

The syntax is:

```
@ if_tok_gt_dbl_tol <token> <double> [<double>... <double>] <tolerance>
// Commands processed if true
@ end_if
```

<token> - the name of the token to test.

<double> - the double value to compare against the value of token. At least one double value must be specified, however, multiple double values can be included. Multiple double values should be separated by whitespace. Double values should not be quoted.

<tolerance> - the tolerance, expressed as a valid double value, for comparison. This value must be specified and must be the last value.

If the token value is greater than any of the double values within the tolerance, the statement is true.

If the token value is not greater than any of the double values within the tolerance (i.e. the token is less than or equal to all double values), the statement is false.

If the token is undefined, the snippet will produce an error.

Multiple doubles are handled as if joined by logical OR statements.

```
@ def_tok WIDTH 3.5

@ if_tok_gt_dbl_tol WIDTH 3.0 0.1
  user_message "WIDTH is greater than 3.5 within a tolerance of 0.1"
@ end_if

@ if_tok_gt_dbl_tol WIDTH 3.0 3.5
  user_message "WIDTH is greater than 3.0 or 3.5 within a tolerance of 0.1"
@ end_if
```

Continue to 21.5.12.8.15 if_tok_ge_dbl or return to 21.5.12.8 Tokens vs Doubles or 21.5.12 Snippet Directives.

21.5.12.8.15 if_tok_ge_dbl

The **if_tok_ge_dbl** directive tests if a token is greater than or equal to any given double values, within a fixed tolerance of 1.0 x 10-6 (1.0E-6).

The syntax is:

```
@ if_tok_ge_dbl <token> <double> [<double>... <double>]
// Commands processed if true
@ end_if
```

<token> - the name of the token to test.

<double> - the double value to compare against the value of token. At least one double value must be specified, however, multiple double values can be included. Multiple double values should be separated by whitespace. Double values should not be quoted.

If the token value is greater than or equal to any of the double values within the tolerance, the statement is true.

If the token value is not greater than or equal to any of the double values within the tolerance (i.e. the token value is less than all double values), the statement is false.

If the token is undefined, the snippet will produce an error.

Multiple doubles are handled as if joined by logical OR statements.

```
@ def_tok WIDTH 3.5

@ if_tok_ge_dbl WIDTH 3.5
 user_message "WIDTH is greater than or equal to 3.5 within a tolerance of 1e-6"
@ end_if

@ if_tok_ge_dbl WIDTH 3.5 4.0
 user_message "WIDTH is greater than or equal to 3.5 or 4.0 within a tolerance of 1e-6"
@ end_if
```

Continue to 21.5.12.8.16 if_tok_ge_dbl_tol or return to 21.5.12.8 Tokens vs Doubles or 21.5.12 Snippet Directives.

21.5.12.8.16 if_tok_ge_dbl_tol

The **if_tok_ge_dbl_tol** directive tests if a token is greater than or equal to any given double values, within a given tolerance.

The syntax is:

```
@ if_tok_ge_dbl_tol <token> <double> [<double>... <double>] <tolerance>
// Commands processed if true
@ end_if
```

<token> - the name of the token to test.

<double> - the double value to compare against the value of token. At least one double value must be specified, however, multiple double values can be included. Multiple double values should be separated by whitespace. Double values should not be quoted.

<tolerance> - the tolerance, expressed as a valid double value, for comparison. This value must be specified and must be the last value.

If the token value is greater than or equal to any of the double values within the tolerance, the statement is true.

If the token value is not greater than or equal to any of the double values within the tolerance (*i.e.* the token value is less than all double values), the statement is false.

If the token is undefined, the snippet will produce an error.

Multiple doubles are handled as if joined by logical OR statements.

```
@ def_tok WIDTH 3.5

@ if_tok_ge_dbl_tol WIDTH 3.5 0.1
  user_message "WIDTH is greater than or equal to 3.5 within a tolerance of 0.1"
@ end_if

@ if_tok_ge_dbl_tol WIDTH 3.5 4.0
  user_message "WIDTH is greater than or equal to 3.5 or 4.0 within a tolerance of 0.1"
@ end_if
```

Continue to 21.5.12.8.17 if_tok_in_range_dbl or return to 21.5.12.8 Tokens vs Doubles or 21.5.12 Snippet Directives.

21.5.12.8.17 if_tok_in_range_dbl

The **if_tok_in_range_dbl** directive tests if a token is between two double values, within a fixed tolerance of 1.0 x 10-6 (1.0E-6).

The syntax is:

```
@ if_tok_in_range_dbl <token> <lower> <upper>
// Commands processed if true
@ end_if
```

<token> - the name of the token to test.

<lower> - the double representing the lower bound of the range to check against the token. This value must be specified.

<upper> - the double representing the upper bound of the range to check against the token. This value must be specified.

If the token value is within the range from lower to upper within the tolerance, the statement is true.

If the token is not defined or is outside the range from lower to upper within the tolerance, the statement is false.

If the token is undefined, the snippet will produce an error.

```
@ def_tok WIDTH 3.5

@ if_tok_in_range_dbl WIDTH 3.0 4.0
  user_message "WIDTH is within range 3.0 to 4.0 within a tolerance of 1e-6"
@ else
  user_message "WIDTH is outside range 3.0 to 4.0 within a tolerance of 1e-6"
@ end_if
```

Continue to 21.5.12.8.18 if_tok_in_range_dbl_tol or return to 21.5.12.8 Tokens vs Doubles or 21.5.12 Snippet Directives.

21.5.12.8.18 if_tok_in_range_dbl_tol

The **if_tok_in_range_dbl** directive tests if a token is between two double values, within a given tolerance.

The syntax is:

```
@ if_tok_in_range_dbl <token> <lower> <upper> <tolerance>
// Commands processed if true
@ end_if
```

<token> - the name of the token to compare.

<lower> - the double value representing the lower bound of the range to check against the token. This value must be specified.

<upper> - the double value representing the upper bound of the range to check against the token. This value must be specified.

<tolerance> - the tolerance, expressed as a valid double value, for comparison. This value must be specified and must be the last value.

If the token value is within the range from lower to upper within the tolerance, the statement is true.

If the token is outside the range from lower to upper within the tolerance, the statement is false.

If the token is undefined, the snippet will produce an error.

```
@ def_tok WIDTH 3.5

@ if_tok_in_range_dbl_tol WIDTH 3.0 4.0 0.1
  user_message "WIDTH is within range 3.0 to 4.0 within a tolerance of 0.1"
@ else
  user_message "WIDTH is outside range 3.0 to 4.0 within a tolerance of 0.1"
@ end_if
```

Continue to 21.5.12.8.19 if_tok_nin_range_dbl or return to 21.5.12.8 Tokens vs Doubles or 21.5.12 Snippet Directives.

21.5.12.8.19 if_tok_nin_range_dbl

The **if_tok_nin_range_dbl** directive tests if a token is not between two double values, within a fixed tolerance of 1.0 x 10-6 (1.0E-6).

The syntax is:

```
@ if_tok_nin_range_dbl <token> <lower> <upper>
// Commands processed if true
@ end_if
```

<token> - the name of the token to compare.

<lower> - the double value representing the lower bound of the range to check against the token. This value must be specified.

<upper> - the double value representing the upper bound of the range to check against the token. This value must be specified.

If the token value is within the range from lower to upper within the tolerance, the statement is true.

If the token is outside the range from lower to upper within the tolerance, the statement is false.

If the token is undefined, the snippet will produce an error.

```
@ def_tok WIDTH 4.0

@ if_tok_nin_range WIDTH 3.0 3.5
  user_message "WIDTH is not within range 3.0 to 3.5 within a tolerance of 1e-6"
@ end_if
```

See also:

if_tok_in_range_dbl - logically opposite directive

if_val_nin_range_dbl - for testing values rather than tokens

Continue to 21.5.12.8.20 if_tok_nin_range_dbl_tol or return to 21.5.12.8 Tokens vs Doubles or 21.5.12 Snippet Directives.

21.5.12.8.20 if_tok_nin_range_dbl_tol

The **if_tok_nin_range_dbl_tol** directive tests if a token is between two double values, within a given tolerance.

The syntax is:

```
@ if_tok_nin_range_dbl <token> <lower> <upper> <tolerance>
// Commands processed if true
@ end_if
```

<token> - the name of the token to compare.

<lower> - the double value representing the lower bound of the range to check against the token. This value must be specified.

<upper> - the double value representing the upper bound of the range to check against the token. This value must be specified.

<tolerance> - the tolerance, expressed as a valid double value, for comparison. This value must be specified and must be the last value.

If the token value is not within the range from lower to upper within the tolerance, the statement is true.

If the token is within the range from lower to upper within the tolerance, the statement is false.

If the token is undefined, the snippet will produce an error.

```
@ def_tok WIDTH 4.0

@ if_tok_nin_range_dbl_tol WIDTH 3.0 3.5 0.1
  user_message "WIDTH is not within the range 3.0 to 3.5 within a tolerance of 0.1"
@ end_if
```

See also:

if_tok_nin_range_dbl - for comparison with fixed tolerance

if_tok_in_range_dbl_tol - logically opposite directive

if_val_nin_range_dbl_tol - for testing of values rather than tokens

Continue to 21.5.12.9 Tokens vs Integers or return to 21.5.12.8 Tokens vs Doubles or 21.5.12 Snippet Directives.

21.5.12.9 Tokens vs Integers

The following directives compare a token value against given integers.

See

21.5.12.9.1 if_tok_eq_int
21.5.12.9.2 if_tok_neq_int
21.5.12.9.3 if_tok_lt_int
21.5.12.9.4 if_tok_le_int
21.5.12.9.5 if_tok_gt_int
21.5.12.9.6 if_tok_ge_int

Or return to 21.5 Defining and Using Snippets.

21.5.12.9.1 if_tok_eq_int

The **if_tok_eq_int** directive tests if a token equals any of the given integers.

The syntax is:

```
@ if_tok_eq_int <token> <integer> [<integer>... <integer>]
// Commands to be processed if true
@ end_if
```

<token> - the name of the token to compare.

<integer> - the integer value to compare against the token. At least one integer must be specified, however, multiple integers can be included. Multiple integers should be separated by whitespace.

If the token value is equal to any of the integers, the statement is true.

If the token value is not equal to any of the integers, the statement is false.

If the token is undefined, the snippet will produce an error.

Multiple integers are handled as if joined by logical OR statements.

```
@ def_tok NUM_LANES 2

@ if_tok_eq_int NUM_LANES 2
  user_message "NUM_LANES equals 2"
@ end_if

@ if_tok_eq_int NUM_LANES 2 3 4
  user_message "NUM_LANES equals 2, 3 or 4"
@ end_if
```

See also:

if_tok_neq_int - logically opposite directive

if_tok_eq_dbl - for comparison with doubles

if_val_eq_int - for testing of values rather than tokens

Continue to 21.5.12.9.2 if_tok_neq_int or return to 21.5.12.9 Tokens vs Integers or 21.5.12 Snippet Directives.

21.5.12.9.2 if_tok_neq_int

The **if_tok_neq_int** directive tests if a token does not equal any of the integers.

The syntax is:

```
@ if_tok_neq_int <token> <integer> [<integer>... <integer>]
// Commands to be processed if true
@ end_if
```

<token> - the name of the token to compare.

<integer> - the integer value to compare against the token. At least one integer must be specified, however, multiple integers can be included. Multiple integers should be separated by whitespace.

If the token value does not equal any of the integers, the statement is true.

If the token value equals any of the integers, the statement is false.

If the token is undefined, the snippet will produce an error.

Multiple integers are handled as if joined by logical OR statements.

```
@ def_tok NUM_LANES 2

@ if_tok_neq_int NUM_LANES 2
  user_message "NUM_LANES is not equal to 2"
@ end_if

@ if_tok_neq_int NUM_LANES 2 3 4
  user_message "NUM_LANES equal to 2, 3 or 4"
@ end_if
```

See also:

if_tok_eq_int - logically opposite directive

if_tok_neq_dbl - for comparison with doubles

if_val_neq_int - for testing of values rather than tokens

Continue to 21.5.12.9.3 if_tok_lt_int or return to 21.5.12.9 Tokens vs Integers or 21.5.12 Snippet Directives.

21.5.12.9.3 if_tok_lt_int

The **if_tok_lt_int** directive tests if a token is less than any given integers. The comparison is numerical.

The syntax is:

```
@ if_tok_lt_int <token> <integer> [<integer>... <integer>]
// Commands processed if true
@ end_if
```

<token> - the name of the token to compare.

<integer> - the integer value to compare against the token. At least one integer must be specified, however, multiple integers can be included. Multiple integers should be separated by whitespace.

If the token value is less than any of the integers, the statement is true.

If the token value is not less than any of the integers (*i.e.* the token value is greater than or equal to all integer values), the statement is false.

If the token is undefined, the snippet will produce an error.

Multiple integers are handled as if joined by logical OR statements.

```
@ def_tok NUM_LANES 2

@ if_tok_lt_int NUM_LANES 3
  user_message "NUM_LANES is less than 3"
@ end_if

@ if_tok_lt_int NUM_LANES 2 3 4
  user_message "NUM_LANES is less than 2, 3 or 4"
@ end_if
```

See also:

if_tok_le_int - for less than or equal to comparison against integers

if_tok_gt_int - for greater than comparison against integers

if_tok_ge_int - for greater than or equal to comparison against integers

if_tok_lt_dbl - for comparison against doubles

if_val_lt_int - for testing of values rather than tokens

Continue to 21.5.12.9.4 if_tok_le_int or return to 21.5.12.9 Tokens vs Integers or 21.5.12 Snippet Directives.

21.5.12.9.4 if_tok_le_int

The **if_tok_le_int** directive tests if a token is less than or equal to any given integers. The comparison is numerical.

The syntax is:

```
@ if_tok_le_int <token> <integer> [<integer>... <integer>]
// Commands processed if true
@ end_if
```

<token> - the name of the token to compare.

<integer> - the integer value to compare against the token. At least one integer must be specified, however, multiple integers can be included. Multiple integers should be separated by whitespace.

If the token value is less than or equal to any of the integers, the statement is true.

If the token value is not less than or equal to any of the integers (i.e. the token value is greater than all integer values), the statement is false.

If the token is undefined, the snippet will produce an error.

Multiple integers are handled as if joined by logical OR statements.

```
@ def_tok NUM_LANES 2

@ if_tok_le_int NUM_LANES 2
  user_message "NUM_LANES is less than or equal to 2"
@ end_if

@ if_tok_le_int NUM_LANES 2 3 4
  user_message "NUM_LANES is less than or equal to 2, 3 or 4"
@ end_if
```

See also:

if_tok_lt_int - for less than comparison against integers

if_tok_gt_int - for greater than comparison against integers

if_tok_ge_int - for greater than or equal to comparison against integers

if_tok_le_dbl - for comparison against doubles

if_val_le_int - for testing of values rather than tokens

Continue to 21.5.12.9.5 if_tok_gt_int or return to 21.5.12.9 Tokens vs Integers or 21.5.12 Snippet Directives.

21.5.12.9.5 if_tok_gt_int

The **if_tok_gt_int** directive tests if a token is greater than any given integers. The comparison is numerical.

The syntax is:

```
@ if_tok_gt_int <token> <integer> [<integer>... <integer>]
// Commands processed if true
@ end_if
```

<token> - the name of the token to compare.

<integer> - the integer value to compare against the token. At least one integer must be specified, however, multiple integers can be included. Multiple integers should be separated by whitespace.

If the token value is greater than any of the integers, the statement is true.

If the token value is not greater than any of the integers (*i.e.* the token value is less than or equal to all integer values), the statement is false.

If the token is undefined, the snippet will produce an error.

Multiple integers are handled as if joined by logical OR statements.

```
@ def_tok NUM_LANES 3

@ if_tok_neq_int NUM_LANES 2
  user_message "NUM_LANES is greater than 2"
@ end_if

@ if_tok_neq_int NUM_LANES 2 3 4
  user_message "NUM_LANES is greater than 2, 3 or 4"
@ end_if
```

See also:

if_tok_le_int - for less than or equal to comparison against integers

if_tok_lt_int - for less than comparison against integers

if_tok_ge_int - for greater than or equal to comparison against integers

if_tok_gt_dbl - for comparison against doubles

if_val_gt_int - for testing of values rather than tokens

Continue to 21.5.12.9.6 if_tok_ge_int or return to 21.5.12.9 Tokens vs Integers or 21.5.12 Snippet Directives.

21.5.12.9.6 if_tok_ge_int

The **if_tok_ge_int** directive tests if a token is greater than or equal to any given integers.

The syntax is:

```
@ if_tok_ge_int <token> <integer> [<integer>... <integer>]
// Commands processed if true
@ end_if
```

<token> - the name of the token to compare.

<integer> - the integer value to compare against the token. At least one integer must be specified, however, multiple integers can be included. Multiple integers should be separated by whitespace.

If the token value is greater than or equal to any of the integers, the statement is true.

If the token value is not greater than or equal to any of the integers (*i.e.* the token value is less than all integer values), the statement is false.

If the token is undefined, the snippet will produce an error.

Multiple integers are handled as if joined by logical OR statements.

```
@ def_tok NUM_LANES 2

@ if_tok_ge_int NUM_LANES 2
  user_message "NUM_LANES is greater than or equal to 2"
@ end_if

@ if_tok_ge_int NUM_LANES 2 3 4
  user_message "NUM_LANES is greater than or equal to 2, 3 or 4"
@ end_if
```

See also:

if_tok_lt_int - for less than comparison against integers

if_tok_gt_int - for greater than comparison against integers

if_tok_le_int - for less than or equal to comparison against integers

if_tok_ge_dbl - for comparison against doubles

if_val_ge_int - for testing of values rather than tokens

Continue to 21.5.12.10 Values Defined and Value vs Value or return to 21.5.12.9 Tokens vs Integers or 21.5.12 Snippet Directives.

21.5.12.10 Values Defined and Value vs Value

The following directives test whether values are empty and compare values with other values. Values can be parameter values (*i.e.* substituted) or raw values (*i.e.* in the file).

Note: these directives should not be used with tokens. The order in which parameters, directives and tokens are processed and substituted (see section 21.5.14 Order of Snippet Processing) will likely result in unexpected results.

See

21.5.12.10.1 if_val
21.5.12.10.2 if_nval
21.5.12.10.3 if_all_vals
21.5.12.10.4 if_nall_vals

Or return to 21.5 Defining and Using Snippets.

21.5.12.10.1 if_val

The **if_val** directive tests if any of the given values are empty.

The syntax is:

```
@ if_val <value> [<value>... <value>]
// Commands processed if statement is true
@ end_if
```

<value> - the value to test. At least one value must be specified, however, multiple values can be included. Multiple values should be separated by whitespace.

If any of the values are not empty, the statement is true.

If all of the values are empty, the statement is false.

Multiple values are handled as if joined by logical OR statements.

NOTE: This directive simply tests whether the value is defined and exists and not whether the value is valid or blank.

```
// PARAMETER PA TEXT "PA" OPTIONAL
// PARAMETER PB TEXT "PB" OPTIONAL
// PARAMETER PC TEXT "PC" OPTIONAL

@ if_val "$(PA)"
user_message "Found PA"
@ end_if

@ if_val "$(PA)" "$(PB)"
user_message "Found PA or PB"
@ end_if

@ if_val "$(PA)" "$(PB)" "$(PC)"
user_message "Found PA or PB or PC"

@ end_if
```

Continue to 21.5.12.10.2 if_nval or return to 21.5.12.10 Values Defined and Value vs Value or 21.5.12 Snippet Directives.

21.5.12.10.2 if_nval

The **if_nval** directive tests if any of the given values are empty.

The syntax is:

```
@ if_nval <value> [<value>... <value>]
// Commands processed if statement is true
@ end_if
```

<value> - the value to test. At least one value must be specified, however, multiple values can be included. Multiple values should be separated by whitespace.

If any of the values are empty, the statement is true.

If none of the values are empty, the statement is false.

Multiple values are handled as if joined by logical OR statements.

```
// PARAMETER PA TEXT "PA" OPTIONAL
// PARAMETER PB TEXT "PB" OPTIONAL
// PARAMETER PC TEXT "PC" OPTIONAL

@ if_nval "$(PA)"
user_message "Not found PA"
@ end_if

@ if_nval "$(PA)" "$(PB)"
user_message "Not found PA and PB"
@ end_if

@ if_nval "$(PA)" "$(PB)" "$(PC)"
user_message "Not found PA,PB and PC"
@ end_if
```

Continue to 21.5.12.10.3 if_all_vals or return to 21.5.12.10 Values Defined and Value vs Value or 21.5.12 Snippet Directives.

21.5.12.10.3 if_all_vals

The **if_all_vals** directive tests whether all given values are not empty.

The syntax is:

```
@ if_all_vals <value> [<value>... <value>]
// Commands to be processed when statement is true
@ end_if
```

<value> - the value to test. At least one value must be specified, however, multiple values can be included. Multiple values should be separated by whitespace.

If all of the values are not empty, the statement is true.

If any of the values are empty, the statement is false.

Multiple values are handled as if joined by logical AND statements.

```
// PARAMETER PA TEXT "PA" OPTIONAL
// PARAMETER PB TEXT "PB" OPTIONAL

@ if_all_vals "$(PA)"
  user_message "PA exists"
@ end_if

@ if_all_vals "$(PA)" "$(PB)"
  user_message "PA and PB both exist!"
@ end_if

@ if_all_vals "$(PA)" "$(PB)" "$(PC)"
  user_message "PA, PB and PC all exist!"
@ else
  user_message "One of PA, PB or PC does not exist"
@ end_if
```

Continue to 21.5.12.10.4 if_nall_vals or return to 21.5.12.10 Values Defined and Value vs Value or 21.5.12 Snippet Directives.

21.5.12.10.4 if_nall_vals

The **if_nall_vals** directive tests if all of the given values are empty.

The syntax is:

```
@ if_nall_vals <value> [<value>... <value>]
// Commands to be processed when statement is true
@ end_if
```

<value> - the value to test. At least one value must be specified, however, multiple values can be included. Multiple values should be separated by whitespace.

If all of the values are empty, the statement is true.

If any of the values are not empty, the statement is false.

Multiple values are handled as if joined by logical AND statements.

```
// PARAMETER PA TEXT "PA" OPTIONAL
// PARAMETER PB TEXT "PB" OPTIONAL
// PARAMETER PC TEXT "PC" OPTIONAL

@ if_nall_vals "$(PA)"
user_message "PA does not exist."
@ end_if

@ if_nall_vals "$(PA)" "$(PB)"
user_message "PA and PB do not exist.!"
@ end_if

@ if_nall_vals "$(PA)" "$(PB)" "$(PC)"
user_message "PA,PB and PC do not exist."
@ end_if
```

Continue to 21.5.12.11 Values vs Strings or return to 21.5.12.10 Values Defined and Value vs Value or 21.5.12 Snippet Directives.

21.5.12.11 Values vs Strings

The following directives compare a value against given strings. Comparison is character-based and not numerical (e.g. "1" does not equal "1.0" or "01").

See

21.5.12.11.1 if_val_eq_str
21.5.12.11.2 if_val_neq_str

Or return to 21.5 Defining and Using Snippets.

21.5.12.11.1 if_val_eq_str

The **if_val_eq_str** directive tests if a value is equal to any of the given strings.

The syntax is:

```
@ if_val_eq_str <value> <string> [<string>... <string>]
// Commands processed if statement is true
@ end if
```

<value> - is the value to compare. One and only one value must be specified.

<string> - is the string to compare against the value. At least one string must be specified, however, multiple strings can be included. Multiple strings should be separated by whitespace.

Strings or values containing whitespace should be enclosed in quotation marks.

If the value is equal to any of the strings, the statement is true.

If the value is not equal to any of the string values, the statement is false.

Multiple strings are handled as if joined by logical OR statements.

```
// PARAMETER KT TEXT "Kerb type" "SA"

@ if_val_eq_str "$(KT)" "SA"
                user_message "Kerb type is equal to SA"
@ end_if

@ if_val_eq_str "$(KT)" "SA" "SM"
                user_message "Kerb type is equal to either SA or SM"
@ end_if
```

Continue to 21.5.12.11.2 if_val_neq_str or return to 21.5.12.11 Values vs Strings or 21.5.12 Snippet Directives.

21.5.12.11.2 if_val_neq_str

The **if_val_neq_str** directive tests if a value is not equal to any of the given strings.

The syntax is:

```
@ if_val_neq_str <value> <string> [<string>... <string>]
// Commands processed if statement is true
@ end if
```

<value> - is the value to compare. One and only one value must be specified.

<string> - is the string to compare against the value. At least one string must be specified, however, multiple strings can be included. Multiple strings should be separated by whitespace.

Strings or values containing whitespace should be enclosed in quotation marks.

If the value is not equal to any of the strings, the statement is true.

If the value is not equal to none of the strings (*i.e.* it is equal to all strings), the statement is false.

Multiple strings are handled as if joined by logical OR statements.

```
// PARAMETER KT TEXT "Kerb type" "SB"

@ if_val_neq_str "$(KT)" "SA"
  user_message "Kerb type is not equal to SA"
@ end_if

@ if_val_neq_str "$(KT)" "SA" "SM"
  user_message "Kerb type is not equal to either SA or SM"
@ end_if
```

Continue to 21.5.12.12 Values vs Doubles or return to 21.5.12.11 Values vs Strings or 21.5.12 Snippet Directives.

21.5.12.12 Values vs Doubles

The following directives compare a token value against given doubles. Comparison is numerical.

See

Or return to 21.5 Defining and Using Snippets.

21.5.12.12.1 if_val_eq_dbl

The **if_val_eq_dbl** directive tests if a value is equal to any of the given double values, with a fixed tolerance of 1.0 x 10-6 (1.0E-6).

The syntax is:

```
@ if_val_eq_dbl <value> <double> [<double>... <double>]
// Commands processed if statement is true
@ end_if
```

<value> - the value to compare. The value should evaluate to a numeric value- integer or double.

<double> - the double to compare against the value. At least one double must be specified, however, multiple double can be included. Multiple double should be separated by whitespace.

Values containing whitespace should be enclosed in quotation marks.

If the value is equal to any of the doubles within the tolerance, the statement is true.

If the value is not equal to any of the doubles within the tolerance, the statement is false.

If the value or any doubles do not evaluate to a valid number, an error is produced.

Multiple doubles are handled as if joined by logical OR statements.

```
// PARAMETER WIDTH REAL "Lane width" 3.5

@ if_val_eq_dbl $(WIDTH) 3.5
  user_message "WIDTH is equal to 3.5 within a tolerance of 1e-6"
@ end_if
```

Continue to 21.5.12.12.2 if_val_eq_dbl_tol or return to 21.5.12.12 Values vs Doubles or 21.5.12 Snippet Directives.

21.5.12.12.2 if_val_eq_dbl_tol

The **if_val_eq_dbl_tol** directive tests if a value is equal to any given double value, within a given tolerance.

The syntax is:

```
@ if_val_eq_dbl_tol <value> <double> [<double>... <double>] <tolerance>
// Commands processed if statement is true
@ end_if
```

<value> - the value to compare. The value should evaluate to a numeric value- integer or double.

<double> - the double to compare against the value. At least one double must be specified, however, multiple double can be included. Multiple double should be separated by whitespace.

<tolerance> - the tolerance, expressed as a valid double value, for comparison. This value must be specified and must be the last value.

If the value is equal to any of the double values within the tolerance, the statement is true.

If the value is not equal to any of the double values within the tolerance, the statement is false.

If the value or any doubles do not evaluate to a valid number, an error is produced.

Multiple double values are handled as if joined by logical OR statements.

```
// PARAMETER WIDTH REAL "Lane width" 3.5

@ if_val_eq_dbl_tol $(WIDTH) 3.5 0.1
  user_message "WIDTH is equal to 3.5 within a tolerance of 0.1"
@ end_if
```

Continue to 21.5.12.12.3 if_val_neq_dbl or return to 21.5.12.12 Values vs Doubles or 21.5.12 Snippet Directives.

21.5.12.12.3 if_val_neq_dbl

The **if_val_neq_dbl** directive tests if a value is not equal to any of the given doubles, with a fixed tolerance of 1.0 x 10-6 (1.0E-6).

The syntax is:

```
@ if_val_neq_dbl <value> <double> [<double>... <double>]
// Commands processed if statement is true
@ end_if
```

<value> - the value to compare. The value should evaluate to a numeric value - integer or double.

<double> - the double to compare against the value. At least one double must be specified, however, multiple double can be included. Multiple double should be separated by whitespace.

If the value is not equal to any of the doubles within the tolerance, the statement is true.

If the value is equal to all of the doubles within the tolerance, the statement is false.

If the value or any doubles do not evaluate to a valid number, an error is produced.

Multiple doubles are handled as if joined by logical OR statements.

```
// PARAMETER WIDTH REAL "Lane width" 3.5

@ if_val_neq_dbl $(WIDTH) 3.0
  user_message "WIDTH is not equal to 3.0 within a tolerance of 1e-6"
@ end_if
```

Continue to 21.5.12.12.4 if_val_neq_dbl_tol or return to 21.5.12.12 Values vs Doubles or 21.5.12 Snippet Directives.

21.5.12.12.4 if_val_neq_dbl_tol

The **if_val_neq_dbl_tol** directive tests if a value is not equal to any given doubles, within a given tolerance.

The syntax is:

```
@ if_val_neq_dbl_tol <value> <double> [<double>... <double>] <tolerance>
// Commands processed if statement is true
@ end_if
```

<value> - the value to compare. The value should evaluate to a numeric value- integer or double.

<double> - the double to compare against the value. At least one double must be specified, however, multiple double can be included. Multiple double should be separated by whitespace.

<tolerance> - the tolerance, expressed as a valid double value, for comparison. This value must be specified and must be the last value.

If the value is not equal to any of the doubles within the tolerance, the statement is true.

If the value is equal to all of the doubles within the tolerance, the statement is false.

If the value or any doubles do not evaluate to a valid number, an error is produced.

Multiple double values are handled as if joined by logical OR statements.

```
// PARAMETER WIDTH REAL "Lane width" 3.5

@ if_val_neq_dbl_tol $(WIDTH) 3.0 0.1
  user_message "WIDTH is not equal to 3.0 within a tolerance of 0.1"
@ end_if
```

Continue to 21.5.12.12.5 if_val_lt_dbl or return to 21.5.12.12 Values vs Doubles or 21.5.12 Snippet Directives.

21.5.12.12.5 if_val_lt_dbl

The **if_val_lt_dbl** directive tests if a value is less than any of the given double values within a fixed tolerance of 1.0 x 10-6 (1.0E-6).

The syntax is:

```
@ if_val_lt_dbl <value> <double> [<double>... <double>]
// Commands processed if true
@ end_if
```

<value> - the value to compare. The value should evaluate to a numeric value - integer or double.

<double> - the double to compare against the value. At least one double must be specified, however, multiple double can be included. Multiple double should be separated by whitespace.

If the value is less than any of the doubles within the tolerance, the statement is true.

If the value is not less than any of the doubles within the tolerance (*i.e.* the value is greater than or equal to all doubles), the statement is false.

If the value or any doubles do not evaluate to a valid number, an error is produced.

Multiple doubles are handled as if joined by logical OR statements.

```
// PARAMETER WIDTH REAL "Lane width" 3.5

@ if_val_lt_dbl $(WIDTH) 4.0
  user_message "WIDTH is less than 4.0 within a tolerance of 1e-6"
@ end_if

@ if_val_lt_dbl $(WIDTH) 3.0 3.5 4.0
  user_message "WIDTH is less than 3.0, 3.5 or 4.0 within a tolerance of 1e-6"
@ end_if
```

Continue to 21.5.12.12.6 if_val_lt_dbl_tol or return to 21.5.12.12 Values vs Doubles or 21.5.12 Snippet Directives.

21.5.12.12.6 if_val_lt_dbl_tol

The **if_val_lt_dbl_tol** directive tests if a value is less than any given double values, within a given tolerance.

The syntax is:

```
@ if_val_lt_dbl_tol <value> <double> [<double>... <double>] <tolerance>
// Commands processed if true
@ end_if
```

<value> - the value to compare. The value should evaluate to a numeric value - integer or double.

<double> - the double to compare against the value. At least one double must be specified, however, multiple double can be included. Multiple double should be separated by whitespace.

<tolerance> - the tolerance, expressed as a valid double value, for comparison. This value must be specified and must be the last value.

If the value is less than any of the doubles within the tolerance, the statement is true.

If the value is not less than any of the doubles within the tolerance (*i.e.* the value is greater than or equal to all doubles), the statement is false.

If the value or any doubles do not evaluate to a valid number, an error is produced.

Multiple doubles are handled as if joined by logical OR statements.

```
// PARAMETER WIDTH REAL "Lane width" 3.5

@ if_val_lt_dbl_tol $(WIDTH) 4.0 0.1
  user_message "WIDTH is less than 4.0 within a tolerance of 0.1"
@ end_if

@ if_val_lt_dbl_tol $(WIDTH) 3.0 3.5 4.0 0.1
  user_message "WIDTH is less than 3.0, 3.5 or 4.0 within a tolerance of 0.1"
@ end_if
```

Continue to 21.5.12.12.7 if_val_le_dbl or return to 21.5.12.12 Values vs Doubles or 21.5.12 Snippet Directives.

21.5.12.12.7 if_val_le_dbl

The **if_val_le_dbl** directive tests if a value is less than or equal to any of the given double values within a fixed tolerance of 1.0 x 10-6 (1.0E-6).

The syntax is:

```
@ if_val_le_dbl <value> <double> [<double>... <double>]
// Commands processed if true
@ end_if
```

<value> - the value to compare. The value should evaluate to a numeric value - integer or double.

<double> - the double to compare against the value. At least one double must be specified, however, multiple double can be included. Multiple double should be separated by whitespace.

If the value is less than or equal to any of the doubles within the tolerance, the statement is true.

If the value is not less than or equal to any of the doubles within the tolerance (*i.e.* the value is greater than all doubles), the statement is false.

If the value or any doubles do not evaluate to a valid number, an error is produced.

Multiple doubles are handled as if joined by logical OR statements.

```
// PARAMETER WIDTH REAL "Lane width" 3.5

@ if_val_le_dbl $(WIDTH) 3.5
  user_message "WIDTH is less than or equal to 3.5 within a tolerance of 1e-6"
@ end_if

@ if_val_le_dbl $(WIDTH) 3.0 3.5 4.0
  user_message "WIDTH is less than or equal to 3.0, 3.5 or 4.0 within a tolerance of 1e-6"
@ end_if
```

Continue to 21.5.12.12.8 if_val_le_dbl_tol or return to 21.5.12.12 Values vs Doubles or 21.5.12 Snippet Directives.

21.5.12.12.8 if_val_le_dbl_tol

The **if_val_le_dbl_tol** directive tests if a value is less than or equal to any given doubles, within a given tolerance.

The syntax is:

```
@ if_tok_le_dbl_tol <value> <double> [<double>... <double>] <tolerance>
// Commands processed if true
@ end_if
```

<value> - the value to compare. The value should evaluate to a numeric value - integer or double.

<double> - the double to compare against the value. At least one double must be specified, however, multiple double can be included. Multiple double should be separated by whitespace.

<tolerance> - the tolerance, expressed as a valid double value, for comparison. This value must be specified and must be the last value.

If the value is less than or equal to any of the doubles within the tolerance, the statement is true.

If the value is not less than or equal to any of the doubles within the tolerance (*i.e.* the value is greater than all of the doubles), the statement is false.

If the value or any doubles do not evaluate to a valid number, an error is produced.

Multiple doubles are handled as if joined by logical OR statements.

```
// PARAMETER WIDTH REAL "Lane width" 3.5

@ if_val_le_dbl_tol $(WIDTH) 3.5 0.1
  user_message "WIDTH is less than or equal to 3.5 within a tolerance of 0.1"
@ end_if

@ if_val_le_dbl_tol $(WIDTH) 3.0 3.5 4.0 0.1
  user_message "WIDTH is less than or equal to 3.0, 3.5 or 4.0 within a tolerance of 0.1"
@ end_if
```

Continue to 21.5.12.12.9 if_val_gt_dbl or return to 21.5.12.12 Values vs Doubles or 21.5.12 Snippet Directives.

21.5.12.12.9 if_val_gt_dbl

The **if_val_gt_dbl** directive tests if a value is greater than any of the given doubles, within a fixed tolerance of 1.0 x 10-6 (1.0E-6).

The syntax is:

```
@ if_val_gt_dbl <value> <double> [<double>... <double>]
// Commands processed if true
@ end_if
```

<value> - the value to compare. The value should evaluate to a numeric value - integer or double.

<double> - the double to compare against the value. At least one double must be specified, however, multiple double can be included. Multiple double should be separated by whitespace.

If the value is greater than any of the doubles within the tolerance, the statement is true.

If the value is not greater than any of the doubles within the tolerance (*i.e.* the value is less than or equal to all doubles), the statement is false.

If the value or any doubles do not evaluate to a valid number, an error is produced.

Multiple doubles are handled as if joined by logical OR statements.

```
// PARAMETER WIDTH REAL "Lane width" 3.5

@ if_val_gt_dbl $(WIDTH) 3.0
  user_message "WIDTH is greater than 3.5 within a tolerance of 1e-6"
@ end_if

@ if_val_gt_dbl $(WIDTH) 3.0 3.5 4.0
  user_message "WIDTH is greater than 3.0, 3.5 or 4.0 within a tolerance of 1e-6"
@ end_if
```

Continue to 21.5.12.12.8 if_val_le_dbl_tol or return to 21.5.12.12 Values vs Doubles or 21.5.12 Snippet Directives.

21.5.12.12.10 if_val_gt_dbl_tol

The **if_val_gt_dbl_tol** directive tests if a value is greater than any given double values, within a given tolerance.

The syntax is:

```
@ if_val_gt_dbl_tol <value> <double> [<double>... <double>] <tolerance>
// Commands processed if true
@ end_if
```

<value> - the value to compare. The value should evaluate to a numeric value - integer or double.

<double> - the double to compare against the value. At least one double must be specified, however, multiple double can be included. Multiple double should be separated by whitespace.

<tolerance> - the tolerance, expressed as a valid double value, for comparison. This value must be specified and must be the last value.

If the value is greater than any of the doubles within the tolerance, the statement is true.

If the value is not greater than any of the doubles within the tolerance (*i.e.* the value is less than or equal to all doubles), the statement is false.

If the value or any doubles do not evaluate to a valid number, an error is produced.

Multiple doubles are handled as if joined by logical OR statements.

```
// PARAMETER WIDTH REAL "Lane width" 3.5

@ if_val_gt_dbl_tol $(WIDTH) 3.0 0.1
  user_message "WIDTH is greater than 3.0 within a tolerance of 0.1"
@ end_if

@ if_val_gt_dbl_tol $(WIDTH) 3.0 3.5 4.0 0.1
  user_message "WIDTH is greater than 3.0, 3.5 or 4.0 within a tolerance of 0.1"
@ end_if
```

Continue to 21.5.12.12.11 if_val_ge_dbl or return to 21.5.12.12 Values vs Doubles or 21.5.12 Snippet Directives.

21.5.12.12.11 if_val_ge_dbl

The **if_val_ge_dbl** directive tests if a value is greater than or equal to any of the given double values, within a fixed tolerance of 1.0 x 10-6 (1.0E-6).

The syntax is:

```
@ if_val_ge_dbl <value> <double> [<double>... <double>]
// Commands processed if true
@ end_if
```

<value> - the value to compare. The value should evaluate to a numeric value - integer or double.

<double> - the double to compare against the value. At least one double must be specified, however, multiple double can be included. Multiple double should be separated by whitespace.

If the value is greater than or equal to any of the doubles within the tolerance, the statement is true.

If the value is not greater than or equal to any of the doubles within the tolerance(*i.e.* the value is less than all of the double values), the statement is false.

If the value or any doubles do not evaluate to a valid number, an error is produced.

Multiple doubles are handled as if joined by logical OR statements.

```
// PARAMETER WIDTH REAL "Lane width" 3.5

@ if_val_ge_dbl $(WIDTH) 3.5
  user_message "WIDTH is greater than or equal to 3.5 within a tolerance of 1e-6"
@ end_if

@ if_val_gt_dbl $(WIDTH) 3.0 3.5 4.0
  user_message "WIDTH is greater than or equal to 3.0, 3.5 or 4.0 within a tolerance of 1e-6"
@ end_if
```

Continue to 21.5.12.12.12 if_val_ge_dbl_tol or return to 21.5.12.12 Values vs Doubles or 21.5.12 Snippet Directives.

21.5.12.12.12 if_val_ge_dbl_tol

The **if_val_ge_dbl_tol** directive tests if a value is greater than or equal to any given double values, within a given tolerance.

The syntax is:

```
@ if_val_ge_dbl_tol <value> <double> [<double>... <double>] <tolerance>
// Commands processed if true
@ end_if
```

<value> - the value to compare. The value should evaluate to a numeric value - integer or double.

<double> - the double to compare against the value. At least one double must be specified, however, multiple double can be included. Multiple double should be separated by whitespace.

<tolerance> - the tolerance, expressed as a valid double value, for comparison. This value must be specified and must be the last value.

If the value is greater than or equal to any of the double within the tolerance, the statement is true.

If the value is not greater than or equal to any of the doubles within the tolerance (*i.e.* the value is less than all doubles), the statement is false.

If the value or any doubles do not evaluate to a valid number, an error is produced.

Multiple doubles are handled as if joined by logical OR statements.

```
// PARAMETER WIDTH REAL "Lane width" 3.5

@ if_val_ge_dbl_tol $(WIDTH) 3.0 0.1
  user_message "WIDTH is greater than or equal to 3.0 within a tolerance of 0.1"
@ end_if

@ if_val_ge_dbl_tol $(WIDTH) 3.0 3.5 4.0 0.1
  user_message "WIDTH is greater than or equal to 3.0, 3.5 or 4.0 within a tolerance of 0.1"
@ end_if
```

Continue to 21.5.12.12.13 if_val_in_range_dbl or return to 21.5.12.12 Values vs Doubles or 21.5.12 Snippet Directives.

21.5.12.12.13 if_val_in_range_dbl

The **if_val_in_range_dbl** directive tests if a value exists and is between two double values, within a fixed tolerance of 1.0 x 10-6 (1.0E-6).

The syntax is:

```
@ if_val_in_range_dbl <value> <lower> <upper>
// Commands processed if true
@ end_if
```

<value> - the value to compare. The value should evaluate to a numeric value- integer or double.

<lower> - the double representing the lower bound of the range to check against value. This value must be specified.

<upper> - the double representing the upper bound of the range to check against value. This value must be specified.

If the value is within the range from lower to upper within the tolerance, the statement is true.

If the value is not within the range from lower to upper within the tolerance, the statement is false.

If the value or any doubles do not evaluate to a valid number, an error is produced.

```
// PARAMETER WIDTH REAL "Lane width" 3.5

@ if_val_in_range_dbl $(WIDTH) 3.0 4.0
  user_message "WIDTH is between 3.0 and 4.0 within a tolerance of 1e-6"
@ end_if
```

Continue to 21.5.12.12.14 if_val_in_range_dbl_tol or return to 21.5.12.12 Values vs Doubles or 21.5.12 Snippet Directives.

21.5.12.12.14 if_val_in_range_dbl_tol

The **if_val_in_range_dbl_tol** directive tests if a value exists and is between two doubles, within a given tolerance.

The syntax is:

```
@ if_val_in_range_dbl_tol <value> <lower> <upper> <tolerance>
// Commands processed if true
@ end_if
```

<value> - the value to compare. The value should evaluate to a numeric value- integer or double.

<lower> - the double value representing the lower bound of the range to check against value. This value must be specified.

<upper> - the double value representing the upper bound of the range to check against value. This value must be specified.

<tolerance> - the tolerance, expressed as a valid double value, for comparison. This value must be specified and must be the last value.

If the value is within the range from lower to upper within the tolerance, the statement is true.

If the value is not within the range from lower to upper within the tolerance, the statement is false.

If the value or any doubles do not evaluate to a valid number, an error is produced.

```
// PARAMETER WIDTH REAL "Lane width" 3.5

@ if_val_in_range_dbl_tol $(WIDTH) 3.0 4.0 0.1
  user_message "WIDTH is between 3.0 and 4.0 within a tolerance of 0.1"
@ end_if
```

Continue to 21.5.12.12.15 if_val_nin_range_dbl or return to 21.5.12.12 Values vs Doubles or 21.5.12 Snippet Directives.

21.5.12.12.15 if_val_nin_range_dbl

The **if_val_nin_range_dbl** directive tests if a value exists and is not between two double values, within a given tolerance.

The syntax is:

```
@ if_val_nin_range_dbl  <value> <lower> <upper>
// Commands processed if true
@ end_if
```

<value> - the value to compare. The value should evaluate to a numeric value- integer or double.

<lower> - the double representing the lower bound of the range to check against value. This value must be specified.

<upper> - the double representing the upper bound of the range to check against value. This value must be specified.

If the value is not within the range from lower to upper within the tolerance, the statement is true.

If the value is within the range from lower to upper within the tolerance, the statement is false.

If the value or any doubles do not evaluate to a valid number, an error is produced.

```
// PARAMETER WIDTH REAL "Lane width" 4.0

@ if_val_nin_range_dbl $(WIDTH) 3.0 3.5
  user_message "WIDTH is outside the range 3.0 to 3.5 within a tolerance of 1e-6"
@ end_if
```

Continue to 21.5.12.12.16 if_val_nin_range_dbl_tol or return to 21.5.12.12 Values vs Doubles or 21.5.12 Snippet Directives.

21.5.12.12.16 if_val_nin_range_dbl_tol

The **if_val_nin_range_dbl_tol** directive tests if a value exists and is not between two double values, within a given tolerance.

The syntax is:

```
@ if_val_nin_range_dbl_tol  <value> <lower> <upper> <tolerance>
// Commands processed if true
@ end_if
```

<value> - the value to compare. The value should evaluate to a numeric value- integer or double.

<lower> - the double representing the lower bound of the range to check against value. This value must be specified.

<upper> - the double representing the upper bound of the range to check against value. This value must be specified.

<tolerance> - the tolerance, expressed as a valid double value, for comparison. This value must be specified and must be the last value.

If the value is not within the range from lower to upper within the tolerance, the statement is true.

If the value is within the range from lower to upper within the tolerance, the statement is false.

If the value or any doubles do not evaluate to a valid number, an error is produced.

```
// PARAMETER WIDTH REAL "Lane width" 4.0

@ if_val_nin_range_dbl_tol $(WIDTH) 3.0 3.5 0.1
  user_message "WIDTH is outside the range 3.0 to 3.5 within a tolerance of 0.1"
@ end_if
```

Continue to 21.5.12.13 Values vs Integers or return to 21.5.12.12 Values vs Doubles or 21.5.12 Snippet Directives.

21.5.12.13 Values vs Integers

The following directives compare a value against given integers.

See

21.5.12.13.1 if_val_eq_int
21.5.12.13.2 if_val_neq_int
21.5.12.13.3 if_val_eq_all_ints
21.5.12.13.4 if_val_neq_all_ints
21.5.12.13.5 if_val_lt_int
21.5.12.13.6 if_val_le_int
21.5.12.13.7 if_val_gt_int
21.5.12.13.8 if_val_ge_int
21.5.12.13.9 if_val_lt_all_ints
21.5.12.13.10 if_val_le_all_ints
21.5.12.13.11 if_val_gt_all_ints
21.5.12.13.12 if_val_ge_all_ints

Or return to 21.5 Defining and Using Snippets.

21.5.12.13.1 if_val_eq_int

The **if_val_eq_int** directive tests if a value equals any of the given integers.

The syntax is:

```
@ if_val_eq_int <value> <integer> [<integer>... <integer>]
// Commands to be processed if true
@ end_if
```

<value> - the value to test. The value should evaluate to a valid integer.

<integer> - the integer to compare against the value. At least one integer must be specified, however, multiple integers can be included. Multiple integers should be separated by whitespace.

If the value is equal to any of the integers, the statement is true.

If the value is not equal to any of the integers, the statement is false.

If the value or any integers do not evaluate to a valid integer, an error is produced.

Multiple integers are handled as if joined by logical OR statements.

```
// PARAMETER NUM_LANES INTEGER "Number of lanes" 2

@ if_val_eq_int  $(NUM_LANES) 2
  user_message "NUM_LANES is equal to 2"
@ end_if
```

Continue to 21.5.12.13.2 if_val_neq_int or return to 21.5.12.13 Values vs Integers or 21.5.12 Snippet Directives.

21.5.12.13.2 if_val_neq_int

The **if_val_neq_int** directive tests if a value is not equal to any of the integers.

The syntax is:

```
@ if_val_neq_int <value> <integer> [<integer>... <integer>]
// Commands to be processed if true
@ end_if
```

<value> - the value to test. The value should evaluate to a valid integer.

<integer> - the integer value to compare against the value. At least one integer must be specified, however, multiple integers can be included. Multiple integers should be separated by whitespace.

If the value is not equal to any of the integers, the statement is true.

If the value is equal to all of the integers, the statement is false.

If the value or any integers do not evaluate to a valid integer, an error is produced.

Multiple integers are handled as if joined by logical OR statements.

```
// PARAMETER NUM_LANES INTEGER "Number of lanes" 2

@ if_val_neq_int  $(NUM_LANES) 3
  user_message "NUM_LANES is not equal to 3"
@ end_if
```

Continue to 21.5.12.13.3 if_val_eq_all_ints or return to 21.5.12.13 Values vs Integers or 21.5.12 Snippet Directives.

21.5.12.13.3 if_val_eq_all_ints

The **if_val_eq_all_ints** directive tests if a value is equal to all of the given integers.

The syntax is:

```
@ if_val_eq_all_ints <value> <integer> [<integer>... <integer>]
// Commands to be processed if true
@ end_if
```

<value> - the value to test. The value should evaluate to a valid integer.

<integer> - the integer to compare against the value. At least one integer must be specified, however, multiple integers can be included. Multiple integers should be separated by whitespace.

If the value is equal to all of the integers, the statement is true.

If the value is not equal to all of the integers, the statement is false.

If the value or any integers do not evaluate to a valid integer, an error is produced.

Multiple integers are handled as if joined by logical AND statements.

```
// PARAMETER NUM_LANES INTEGER "Number of lanes" 2

@ if_val_eq_all_ints $(NUM_LANES) 2
  user_message "NUM_LANES is equal to 2"
@ end_if

@ if_val_eq_all_ints  $(NUM_LANES) 2 3 4
  user_message "NUM_LANES is equal to 2, 3 and 4"  // This should never be true
@ end_if
```

Continue to 21.5.12.13.4 if_val_neq_all_ints or return to 21.5.12.13 Values vs Integers or 21.5.12 Snippet Directives.

21.5.12.13.4 if_val_neq_all_ints

The **if_val_neq_all_ints** directive tests if a value is not equal to all of the given integers.

The syntax is:

```
@ if_val_neq_all_ints <value> <integer> [<integer>... <integer>]
// Commands to be processed if true
@ end_if
```

<value> - the value to test. The value should evaluate to a valid integer.

<integer> - the integer to compare against the value. At least one integer must be specified, however, multiple integers can be included. Multiple integers should be separated by whitespace.

If the value is not equal to all of the integer values, the statement is true.

If the value is equal to any of the integer values, the statement is false.

If the value or any integers do not evaluate to a valid integer, an error is produced.

Multiple integers are handled as if joined by logical AND statements.

```
// PARAMETER NUM_LANES INTEGER "Number of lanes" 2

@ if_val_neq_all_ints $(NUM_LANES) 3 4 5
  user_message "NUM_LANES is not equal to 3, 4 AND 5"
@ end_if
```

Continue to 21.5.12.13.5 if_val_lt_int or return to 21.5.12.13 Values vs Integers or 21.5.12 Snippet Directives.

21.5.12.13.5 if_val_lt_int

The **if_val_lt_int** directive tests if a value is less than any given integers.

The syntax is:

```
@ if_val_lt_int <value> <integer> [<integer>... <integer>]
// Commands processed if true
@ end_if
```

<value> - the value to test. The value should evaluate to a valid integer.

<integer> - the integer value to compare against the value. At least one integer must be specified, however, multiple integers can be included. Multiple integers should be separated by whitespace.

If the value is less than any of the integers, the statement is true.

If the value is not less than any of the integers (*i.e.* the value is greater than or equal to all integer values), the statement is false.

If the value or any integers do not evaluate to a valid integer, an error is produced.

Multiple integers are handled as if joined by logical OR statements.

```
// PARAMETER NUM_LANES INTEGER "Number of lanes" 2

@ if_val_lt_int $(NUM_LANES) 3 4
  user_message "NUM_LANES is less than 3 or 4"
@ end_if
```

See also:

if_val_le_int - for less than or equal to comparison against integers

if_val_gt_int - for greater than comparison against integers

if_val_ge_int - for greater than or equal to comparison against integers

if_val_lt_dbl - for comparison against doubles

if_tok_lt_int - for testing of tokens rather than values

Continue to 21.5.12.13.6 if_val_le_int or return to 21.5.12.13 Values vs Integers or 21.5.12 Snippet Directives.

21.5.12.13.6 if_val_le_int

The **if_val_le_int** directive tests if a value is less than or equal to any given integers.

The syntax is:

```
@ if_val_le_int <value> <integer> [<integer>... <integer>]
// Commands processed if true
@ end_if
```

<value> - the value to test. The value should evaluate to a valid integer.

<integer> - the integer value to compare against the value. At least one integer must be specified, however, multiple integers can be included. Multiple integers should be separated by whitespace.

If the value is less than or equal to any of the integers, the statement is true.

If the value is not less than or equal to any of the integers (*i.e.* the value is greater than all integer values), the statement is false.

If the value or any integers do not evaluate to a valid integer, an error is produced.

Multiple integers are handled as if joined by logical OR statements.

```
// PARAMETER NUM_LANES INTEGER "Number of lanes" 2

@ if_val_le_int $(NUM_LANES) 3 4
  user_message "NUM_LANES is less than or equal to 3 or 4"
@ end_if
```

See also:

if_val_lt_int - for less than comparison against integers

if_val_gt_int - for greater than comparison against integers

if_val_ge_int - for greater than or equal to comparison against integers

if_val_le_dbl - for comparison against doubles

if_tok_le_int - for testing of tokens rather than values

Continue to 21.5.12.13.7 if_val_gt_int or return to 21.5.12.13 Values vs Integers or 21.5.12 Snippet Directives.

21.5.12.13.7 if_val_gt_int

The **if_val_gt_int** directive tests if a value is greater than any given integers.

The syntax is:

```
@ if_val_gt_int <value> <integer> [<integer>... <integer>]
// Commands processed if true
@ end_if
```

<value> - the value to test. The value should evaluate to a valid integer.

<integer> - the integer to compare against the value. At least one integer must be specified, however, multiple integers can be included. Multiple integers should be separated by whitespace.

If the value is greater than any of the integers, the statement is true.

If the value is not greater than any of the integers (*i.e.* the value is less than or equal to all integer values), the statement is false.

If the value or any integers do not evaluate to a valid integer, an error is produced.

Multiple integers are handled as if joined by logical OR statements.

```
// PARAMETER NUM_LANES INTEGER "Number of lanes" 5

@ if_val_gt_int $(NUM_LANES) 3 4
  user_message "NUM_LANES is greater than 3 or 4"
@ end_if
```

See also:

if_val_le_int - for less than or equal to comparison against integers

if_val_lt_int - for less than comparison against integers

if_val_ge_int - for greater than or equal to comparison against integers

if_val_gt_dbl - for comparison against doubles

if_tok_gt_int - for testing of tokens rather than values

Continue to 21.5.12.13.8 if_val_ge_int or return to 21.5.12.13 Values vs Integers or 21.5.12 Snippet Directives.

21.5.12.13.8 if_val_ge_int

The **if_val_ge_int** directive tests if a value is greater than or equal to any given integers.

The syntax is:

```
@ if_val_ge_int <value> <integer> [<integer>... <integer>]
// Commands processed if true
@ end_if
```

<value> - the value to test. The value should evaluate to a valid integer.

<integer> - the integer value to compare against the value. At least one integer must be specified, however, multiple integers can be included. Multiple integers should be separated by whitespace.

If the value is greater than or equal to any of the integers, the statement is true.

If the value is not greater than or equal to any of the integers (*i.e.* the value is less than all integer values), the statement is false.

If the value or any integers do not evaluate to a valid integer, an error is produced.

Multiple integers are handled as if joined by logical OR statements.

```
// PARAMETER NUM_LANES INTEGER "Number of lanes" 3

@ if_val_ge_int $(NUM_LANES) 3 4
  user_message "NUM_LANES is greater than or equal to 3 or 4"
@ end_if
```

See also:

if_val_lt_int - for less than comparison against integers

if_val_gt_int - for greater than comparison against integers

if_val_le_int - for less than or equal to comparison against integers

if_val_ge_dbl - for comparison against doubles

if_tok_ge_int - for testing of tokens rather than values

Continue to 21.5.12.13.9 if_val_lt_all_ints or return to 21.5.12.13 Values vs Integers or 21.5.12 Snippet Directives.

21.5.12.13.9 if_val_lt_all_ints

The **if_val_lt_all_ints** directive tests if a value is less than all given integers. The comparison is numerical.

The syntax is:

```
@ if_val_lt_all_ints <value> <integer> [<integer>... <integer>]
// Commands processed if true
@ end_if
```

<value> - the value to test. The value should evaluate to a valid integer.

<integer> - the integer value to compare against the value. At least one integer must be specified, however, multiple integers can be included. Multiple integers should be separated by whitespace.

If the value is less than all of the integers, the statement is true.

If the value is not less than all of the integers (*i.e.* the value is greater than or equal to any integer values), the statement is false.

If the value or any integers do not evaluate to a valid integer, an error is produced.

Multiple integers are handled as if joined by logical AND statements.

```
// PARAMETER NUM_LANES INTEGER "Number of lanes" 2

@ if_val_lt_all_ints $(NUM_LANES) 3 4 5
  user_message "NUM_LANES is less than 3, 4 AND 5"
@ end_if
```

See also:

if_val_le_all_ints- for less than or equal to comparison against integers

if_val_gt_all_ints - for greater than comparison against integers

if_val_ge_all_ints - for greater than or equal to comparison against integers

if_val_lt_int - for comparison against any integers

Continue to 21.5.12.13.10 if_val_le_all_ints or return to 21.5.12.13 Values vs Integers or 21.5.12 Snippet Directives.

21.5.12.13.10 if_val_le_all_ints

The **if_val_le_all_ints** directive tests if a value is less than or equal to all given integers.

The syntax is:

```
@ if_val_le_all_ints <value> <integer> [<integer>... <integer>]
// Commands processed if true
@ end_if
```

<value> - the value to test. The value should evaluate to a valid integer.

<integer> - the integer value to compare against the value. At least one integer must be specified, however, multiple integers can be included. Multiple integers should be separated by whitespace.

If the value is less than or equal to all of the integers, the statement is true.

If the value is not less than or equal to all of the integers (*i.e.* the value is greater than any integer), the statement is false.

If the value or any integers do not evaluate to a valid integer, an error is produced.

Multiple integers are handled as if joined by logical AND statements.

```
// PARAMETER NUM_LANES INTEGER "Number of lanes" 3

@ if_val_le_all_ints $(NUM_LANES) 3 4 5
  user_message "NUM_LANES is less than or equal to 3, 4 AND 5"
@ end_if
```

See also:

if_val_lt_all_ints - for less than comparison against integers

if_val_gt_all_ints - for greater than comparison against integers

if_val_ge_all_ints - for greater than or equal to comparison against integers

if_val_le_int - for comparison against any integers

Continue to 21.5.12.13.11 if_val_gt_all_ints or return to 21.5.12.13 Values vs Integers or 21.5.12 Snippet Directives.

21.5.12.13.11 if_val_gt_all_ints

The **if_val_gt_all_ints** directive tests if a value is greater than all given integers. The syntax is:

The syntax is:

```
@ if_val_gt_all_ints <value> <integer> [<integer>... <integer>]
// Commands processed if true
@ end_if
```

<value> - the value to test. The value should evaluate to a valid integer.

<integer> - the integer value to compare against the value. At least one integer must be specified, however, multiple integers can be included. Multiple integers should be separated by whitespace.

If the value is greater than all of the integers, the statement is true.

If the value is not greater than all of the integers (*i.e.* the value is less than or equal to any integer), the statement is false.

If the value or any integers do not evaluate to a valid integer, an error is produced.

Multiple integers are handled as if joined by logical AND statements.

```
// PARAMETER NUM_LANES INTEGER "Number of lanes" 5

@ if_val_gt_all_ints $(NUM_LANES) 2 3 4
  user_message "NUM_LANES is greater than 2, 3 AND 4"
@ end_if
```

See also:

if_val_le_all_ints - for less than or equal to comparison against integers

if_val_lt_all_ints - for less than comparison against integers

if_val_ge_all_ints - for greater than or equal to comparison against integers

if_val_gt_int - for comparison against any integers

Continue to 21.5.12.13.12 if_val_ge_all_ints or return to 21.5.12.13 Values vs Integers or 21.5.12 Snippet Directives.

21.5.12.13.12 if_val_ge_all_ints

The **if_val_ge_all_ints** directive tests if a value is greater than or equal to all given integers.

The syntax is:

```
@ if_val_ge_all_ints <value> <integer> [<integer>... <integer>]
// Commands processed if true
@ end_if
```

<value> - the value to test. The value should evaluate to a valid integer.

<integer> - the integer value to compare against the value. At least one integer must be specified, however, multiple integers can be included. Multiple integers should be separated by whitespace.

If the value is greater than or equal to all of the integers, the statement is true.

If the value is not greater than or equal to all of the integers (*i.e.* the value is less than any integer values), the statement is false.

If the value or any integers do not evaluate to a valid integer, an error is produced.

Multiple integers are handled as if joined by logical AND statements.

```
// PARAMETER NUM_LANES INTEGER "Number of lanes" 5

@ if_val_ge_all_ints $(NUM_LANES) 3 4 5
  user_message "NUM_LANES is greater than or equal to 3, 4 AND 5"
@ end_if
```

See also:

if_val_lt_all_ints - for less than comparison against integers

if_val_gt_all_ints - for greater than comparison against integers

if_val_le_all_ints - for less than or equal to comparison against integers

if_val_ge_int - for comparison against any integers

Continue to 21.5.13 Snippet Variables or return to 21.5.12.13 Values vs Integers or 21.5.12 Snippet Directives.

21.5.13 Snippet Variables

In ***12d Model 11*** snippets, it is possible to define variables whose values are calculated from the design model. For example, we can capture the offset or width of a link as a variable. This allows snippets to be more flexible and work with normal modifier commands that can alter the position of links.

See

21.5.13.1 Defining Link Variables
21.5.13.2 Defining General Variables
21.5.13.3 Snippet Helper Functions
21.5.13.4 Using Variables in Other Variables
21.5.13.5 Using Variables in Commands

Or return to 21.5 Defining and Using Snippets.

21.5.13.1 Defining Link Variables

The first necessary step in using snippet link variables is to define them. It is assumed that there are already some existing links in the design - not necessarily in the snippet or even output to a model. Defining a link variable allows 12d to identify which particular links need to be measured and calculated.

The syntax to define a link variable is:

```
link_variable <link_name> <chg_range> <alias>
```

<link_name> is the shorthand in MTF code for picking the desired link from the model. Refer to Sect. 19.1.1.2 of the 12d Help for more info.

<chg_range> is the chainage range for this variable, typically the standard range for the entire snippet: $(_SCH) 0 $(_ECH) 0

<alias> is the name for this particular link_variable. We will be able to refer to this link_variable elsewhere in our snippet by this name.

A link_variable is tied to a particular link in the MTF model. It is simply an alias to a particular link in our design.

So, for example, if we have a link called KLIP on the Design layer that we want to refer to, we would do this:

```
link_variable "Design=>KLIP" $(_SCH) 0 $(_ECH) 0 "LINK1"
```

"Design=>KLIP" is the link "KLIP" from the "Design" layer. The side (left/right) will be automatically selected depending on which side this snippet is applied.

$(_SCH) 0 $(_ECH) 0 is the standard snippet text defining the chainage range

"LINK1" is our custom alias for this particular link.

Continue to 21.5.13.2 Defining General Variables or return to 21.5.13 Snippet Variables

21.5.13.2 Defining General Variables

In addition to link variables, which are specifically tied to a particular link in the model, it is possible to define general variables for intermediate calculations or storage of values. These are done with the generic variable command.

The general variable command has the syntax:

```
variable <name> <chg_range> <value>
```

<name> is the name for this variable and must be in quotes, e.g. "VAR1"

<chg_range> is the chainage range for this variable, typically the standard range for the entire snippet: $(_SCH) 0 $(_ECH) 0

<value> is the value for this variable, which can include other variables, directives and formulae.

For example:

```
// PARAMETER WIDTH REAL "Road width" 3.5
variable "HALF_W" $(_SCH) 0 $(_ECH) 0 "($(WIDTH) / 2)"
```

So, if the WIDTH parameter had a value of 3.5, the resulting variable HALF_W would have a calculated value of 3.5 / 2 = 1.75. This value is calculated at run-time and at every section to which this snippet applies.

Continue to 21.5.13.3 Snippet Helper Functions or return to 21.5.13 Snippet Variables

21.5.13.3 Snippet Helper Functions

Snippets often require the calculation of various values and measurements. For example, the current offset of a link from the centreline or the height difference between two links.

These measurements and values could be used for performing conditional tests or simply as the basis for other parts of the snippet.

To this end, there are various helper functions available within snippets. 12d will automatically calculate the relevant value at every applicable section every time it is run. These can, therefore, also be used within variable commands.

```
link_variable "Design=>KLIP" $(_SCH) 0 $(_ECH) 0 "LINK1"
variable "HALF_W" $(_SCH) 0 $(_ECH) 0 "(link_width(LINK1) / 2)"
```

In this example, the value of the variable "HALF_W" is calculated from the width of the link_variable with the alias "LINK1" divided by 2. This way, if another modifier or snippet modified the link, we could still use it within our snippet without the user needing to manually enter any width manually.

The list of available helper functions is:

link_width(XX) - the width of the link XX relative to the previous link

link_height(XX) - the height of the link XX relative to the previous link

link_grade(XX) - the grade of the link XX in y/x (not in %) relative to the previous link

rel_offset(XX) - the unsigned offset of the link XX, typically +ve for LHS and RHS

abs_offset(XX) - the signed offset of the link XX, typically -ve for LHS and +ve for RHS

abs_height(XX) - the signed height difference from the hinge of the link XX, -ve lower, +ve higher

abs_level(XX) - the level of the link XX, equal to (hinge level + height difference)

x_coord(XX) - the X ordinate of the link XX

y_coord(XX) - the Y ordinate of the link XX

apply_tin_height(XX) - the height of the Apply MTF tin at the given link XX

other_tin_height(LINK;TIN) - to capture the height of any tin (TIN) at a link (LINK) (note the semicolon separating the captured link and the tin name).

named_grade(NG) - Capture signed grade (m/m) of a named grade, NG

named_grade_hdsd_fact(NG) - For named grade, NG, the factor to multiply a horizontal distance by to get a slope distance.

named_grade_sdhd_fact(NG) - For named grade, NG, the factor to multiply a slope distance by to get a horizontal distance.

named_grade_zdsd_fact(NG) - For named grade, NG, the factor to multiply a height difference by to get a slope distance.

named_grade_sdzd_fact(NG) - For named grade, NG, the factor to multiply a slope distance by to get a height difference.

named_grade_hdzd_fact(NG) - For named grade, NG, the factor to multiply a horizontal distance by to get a height difference.

named_grade_zdhd_fact(NG) - For named grade, NG, the factor to multiply a height difference by to get a horizontal distance.

More helper functions may be added in the future.

Continue to 21.5.13.4 Using Variables in Other Variables or return to 21.5.13 Snippet Variables

21.5.13.4 Using Variables in Other Variables

Once variables have been defined, they can be used and substituted into other variables.

To do so, wherever you would like to insert the variable, use the syntax:

```
variable(name)
```

Where variable is the keyword and name is the name of the variable to be substituted.

```
// PARAMETER NUM_LANES INTEGER "Number of lanes" 1
// PARAMETER LANE_WIDTH REAL "Lane width" 3.5
// PARAMETER XFALL "Crossfall %" -3

variable "W0" $(_SCH) 0 $(_ECH) 0 "($(LANE_WIDTH) * $(NUM_LANES))"
variable "H0" $(_SCH) 0 $(_ECH) 0 "(variable(W0) * $(XFALL) / 100)"
```

Continue to 21.5.13.5 Using Variables in Commands or return to 21.5.13 Snippet Variables

21.5.13.5 Using Variables in Commands

Once variables have been defined, they can be used in a selection of modifier commands. To do so, wherever you would like to insert the variable, use the syntax:

```
variable <name>
```

Where variable is a keyword and name is the name of the variable to be substituted.

For example:

```
// PARAMETER NUM_LANES INTEGER "Number of lanes" 1
// PARAMETER LANE_WIDTH REAL "Lane width" 3.5

variable "W0" $(_SCH) 0 $(_ECH) 0 "($(LANE_WIDTH) * $(NUM_LANES))"

insert "Design=>EDGE" "red" variable W0 unknown -3.0 $(_SCH) 0 $(_ECH) 0 absolute extra_start
extra_end
```

Note that an MTF modifier commands must be specifically written to support the use of variables. Currently, the MTF modifier commands and their values that support variables are:

Modify_Decision_Batter: width, height, slope

Modify_Decision_Batter_Test: width, height, slope

Modify_XFall_Point: m_offset, m_height1

Modify_Insert_Absolute: width, height, slope

Modify_Insert_Link: width, height, slope

Modify Link All

Modify_All_To_2_RL

More may be added in the future.

Continue to 21.5.14 Order of Snippet Processing or return to 21.5.13 Snippet Variables

21.5.14 Order of Snippet Processing

When a snippet is used in ***12d Model***, it is as part of an MTF.

Each Snippet referred to in the MTF is in its own file, and each snippet file goes through set steps to calculate and substitute values in the snippet, before it is used in the MTF file.

It is important to understand each step in the processing of a snippet file, and the order of the steps, to ensure that snippet runs as you expect it to.

See

Step 1. Initial Loading of the Snippet File
Step 2. Parameters from the MTF Snippet Panel Are Substituted
Step 3. Snippet Directives Executed
Step 4. Token Values Substituted
Step 5. Snippet Inserted into MTF
Step 6. C Preprocessor Run as Part of MTF processing

Or return to 21.5 Defining and Using Snippets.

Step 1.Initial Loading of the Snippet File

The snippet file is read and loaded into memory.

All comments and blank lines are removed.

An example of a snippet file that we will use at each step is:

```
// PARAMETER FRUIT TEXT "Enter a fruit"

@ def_tok FRUIT_TOKEN "$(FRUIT)"

@ if_val_eq_str "$(FRUIT)" "apple"
  user_message "Correct, you entered apple!"
@ else
  user_message "You entered $(FRUIT), I wanted an apple!"
@ end_if

@ if_val_eq_str "$(FRUIT_TOKEN)" "apple"
  user_message "Correct, you entered apple!"
@ else
  user_message "You entered $(FRUIT_TOKEN), I wanted an apple!"
@ end_if
```

Warning - this may not do what you first think it will as we will see shortly.

Continue to Step 2. Parameters from the MTF Snippet Panel Are Substituted or return to 21.5.14 Order of Snippet Processing

Step 2.Parameters from the MTF Snippet Panel Are Substituted

In the next step on the snippet file, any values of parameters that are specified in the **MTF Snippet** panel are substituted.

That is, all instances of parameter variables, *e.g.* $TOK or $(TOK), are replaced with the value that was selected by the user in the **MTF Snippet** panel.

Automatic parameters (*e.g.* $(_SCH)) are also substituted.

If a parameter cannot be substituted, for example, it has not been defined, it is left as-is in the temporary **MTF** file and an error to the user will be produced.

As an example of the parameter substitution, if in the snippet given in **Step 1**, the user entered a value of **apple** (or accepted the default value) for the **Enter a fruit** parameter field on the **MTF Snippet** panel, then after the Parameter Substitution for FRUIT, the snippet would then look like:

```
@ def_tok FRUIT_TOKEN "apple"

@ if_val_eq_str "apple" "apple"
  user_message "Correct, you entered apple!"
@ else
  user_message "You entered apple, I wanted an apple!"
@ end_if

@ if_val_eq_str "$(FRUIT_TOKEN)" "apple"
  user_message "Correct, you entered apple!"
@ else
  user_message "You entered $(FRUIT_TOKEN), I wanted an apple!"
@ end_if
```

Continue to Step 3. Snippet Directives Executed or return to 21.5.14 Order of Snippet Processing

Step 3.Snippet Directives Executed

In the next pass of the snippet file, any snippet directives are parsed and executed.

At this point, no parameters (or automatic parameters) exist, because they have all been substituted in the previous processing step.

So the snippet started in **Step 1**, will now look like:

```
user_message "Correct, you entered apple!"
user_message "You entered $(FRUIT_TOKEN), I wanted an apple!"
```

Notice that the flow-control previously present has been evaluated and only the applicable commands (in this case, user_message commands) remain.

How did we get the second line ?

The second conditional test directive:

@ if_val_eq_str "$(FRUIT_TOKEN)" "apple"

has evaluated to **false**, because in *if_val_eq_str* the value "$(FRUIT_TOKEN)" is being treated as the raw text string **$(FRUIT_TOKEN)** which is not equal to **apple**.

Hence the corresponding **@ else** block of commands is used and we get the line

user_message "You entered $(FRUIT_TOKEN), I wanted an apple!"

is left in the snippet. Which is probably not what you wanted.

In this particular case, the snippet author would have been better off using the **@ if_tok_eq_str** directive instead of the **@ if_val_eq_str** directive.

```
@ if_tok_eq_str FRUIT_TOKEN "apple"
  user_message "Correct, you entered apple!"
@ else
  user_message "You entered $(FRUIT_TOKEN), I wanted an apple!"
@ end_if
```

The conditional test directive:

@ if_tok_eq_str FRUIT_TOKEN "apple"

will evaluated to **true**, because FRUIT_TOKEN is **apple** and we then get the line in the snippet:

user_message "Correct, you entered apple!"

Continue to Step 4. Token Values Substituted or return to 21.5.14 Order of Snippet Processing

Step 4.Token Values Substituted

In the next processing stage of the snippet, tokens have their values substituted, in a similar manner to that in which parameters were substituted in the earlier step.

At this point, this is simply a replacement of tokens in the snippet file, $TOKEN or $(TOKEN), with their value.

There is no flow control at this point, since it has already been performed in the previous step.

So the snippet started in **Step 1**, will now look like:

```
user_message "Correct, you entered apple!"
user_message "You entered apple, I wanted an apple!"
```

Note that each line containing tokens is processed as many times as necessary until no further tokens remain to be substituted. This allows token values to contain other tokens.

```
@ def_tok TOK1 1.0
@ def_tok TOK2 2.0
@ def_tok TOK3 "($(TOK1) + $(TOK2))"// TOK3
will eventually equal 3.0
```

Although the substitution of parameters and tokens is similar, it is important to understand they occur at separate stages of the snippet processing.

After Step 4, the **temporary** MTF snippet file is generated (.temp_mtf).

All intelligence, formulae and user input have been removed in the temporary MTF snippet file.

Continue to Step 5. Snippet Inserted into MTF or return to 21.5.14 Order of Snippet Processing

Step 5.Snippet Inserted into MTF

At this stage, whatever remains of the snippet is inserted into the MTF File. No further processing is done by the 12d Snippet Preprocessor.

However, note, that the older C preprocessor- the one the handles #define, #ifdef, etc.- has not yet been run.

Continue to Step 6. C Preprocessor Run as Part of MTF processing or return to 21.5.14 Order of Snippet Processing

Step 6.C Preprocessor Run as Part of MTF processing

In the next pass, the C preprocessor- i.e. the one handling #define, #if, etc.- is used on the entire MTF, which now includes the interpreted and processed snippets.

Any parts of any snippets still using the older C preprocessor directives (#define, #if, etc) will be processed at this stage as part of the overall MTF. At this point there is no concept of a snippet as a separate entity- all snippet results have already been inserted into the MTF.

Note

Because of the power and predictability of the **Snippet Directives**, they should always be used instead of any C Preprocessor commands.

For more information on the substitutions and C preprocessor directives supported in the MTF, refer to the ***12d Model Help*** section 21.10.1.16 Substitutions in the Modifiers and Templates File.

Return to 21.5.14 Order of Snippet Processing or 21.5 Defining and Using Snippets, or continue to 21.5.15 Debugging Snippets.

21.5.15 Debugging Snippets

See

Or return to 21.5 Defining and Using Snippets.

21.5.15.1 Print Messages and Log Lines to the Output Window

To help with writing and debugging snippets, V11 provides the ability to write messages to the Output Window. These are done via special snippet commands *user_message_print*, *user_message_log* and *user_message* with the syntax:

(a) user_message_print

user_message_print text_msg **"variable"** "var_name" st_ch extra_st end_ch extra_end

"variable" - optional.

"var_name" is the name of a variable.

"variable" is optional but if it exists then "var_name" must also exist.

st_ch extra_st end_ch extra_end - optional

st_ch is a chainage

extra_st is added to st_ch to give the **start chainage**

end_ch is a chainage

extra_end is added to end_ch to give the **end chainage**

Print the text ***text_msg*** to the Output Window and then print the value of the variable *var_name* to the Output Window.

The messages are produced for every section in the chainage range given by the start chainage and the end chainage.

If the start and end chainage modes are omitted, they are taken to be from the start to the end of the reference string.

user_message_print "Value of REL_OFFSET: " "variable" "REL_OFFSET" $(_SCH) 0 $(_ECH) 0

For every section between the start and the end chainage, this will print the text "Value of REL_OFFSET " followed by the value of the variable REL_OFFSET to the Output Window,

(b) user_message_log, user_message

user_message_log text_msg **"variable"** "var_name" st_ch extra_st end_ch extra_end

user_message text_msg **"variable"** "var_name" st_ch extra_st end_ch extra_end

"variable" - optional.

"var_name" is the name of a variable.

"variable" is optional but if it exists then "var_name" must also exist.

st_ch extra_st end_ch extra_end - optional

st_ch is a chainage

extra_st is added to st_ch to give the **start chainage**

end_ch is a chainage

extra_end is added to end_ch to give the **end chainage**

Create a log line in the Output Window of the text ***text_msg*** followed by the value of the variable *var_name*.

The log lines are produced for every section in the chainage range given by the start chainage and the end chainage.

If the start and end chainage modes are omitted, they are taken to be from the start to the end of the reference string.

```
user_message_log "Value of REL_OFFSET: " "variable" "REL_OFFSET" $(_SCH) 0 $(_ECH) 0
```

For every section between the start and the end chainage, a log line will be created in the Output Window with the text "Value of REL_OFFSET " followed by the value of the variable REL_OFFSET

TIP: If you only want one message, rather than a message for every section, use:

$(_SCH) 0 ($_SCH) 0.0001

```
user_message_print "This will only print once!" $(_SCH) 0 $(_SCH) 0.0001
```

Continue to the next section21.5.15.2 Intermediate Parsing Results or return to 21.5.15 Debugging Snippets or 21.5 Defining and Using Snippets.

21.5.15.2 Intermediate Parsing Results

As explained in the section Order of Snippet Processing, any MTF Snippet file is processed in several rounds, with each round of the process performing various parsing, evaluation and substitution steps. To assist with debugging and understanding the manner in which snippets are processed by ***12d Model***, there is an additional snippet instruction that can be used.

```
// OUTPUT_PARSING_STAGES <filename>
```

For example:

```
// OUTPUT_PARSING_STAGES "dump.txt"
```

With this instruction present in a snippet, 12d will write a detailed log of the snippet at each stage of processing to the given file. The contents of the file are updated/overwritten on each recalc of the Apply MTF function.

For compiled snippets, even if this instruction is present, no output will be produced.

Continue to the next section21.5.15.3 Temporary MTF Snippet File or return to 21.5.15 Debugging Snippets or 21.5 Defining and Using Snippets.

21.5.15.3 Temporary MTF Snippet File

To assist with the development and debugging of MTF snippets, there are a number of environment variables to control the naming of, and stop the deletion of, the temporary MTF file that is created after the Snippet Directives are evaluated and the snippet parameters have been substituted (see Snippet Directives).

At the point of the temporary MTF snippet file being created (.mtfsnippet.temp_mtf), all intelligence has been removed from the output. All formulae, variables, directives and fixed decisions have been removed and only the raw modifier commands remain.

(a) Don't Delete Snippet Temp File

MTF_SNIPPET_TEMP_FILE_USE_SNIPPET_NAME_KEEP_4D 1/0

If enabled (non zero), the temporary MTF file that is generated when MTF snippets are processed and parsed will not be deleted. Each run of the **Apply MTF** function will create or update the contents of a file containing the snippet after processing. The file will be created in the current project's working directory.

(b) Snippet Temp File Use Snippet Name

MTF_SNIPPET_TEMP_FILE_USE_SNIPPET_NAME_4D 1/0

If enabled (non zero), the temporary snippet file generated when MTF snippets are processed will be given the name of the snippet file with the extension ".temp_mtf".

For example, if the snippet file is called "INSERT_KERB.mtfsnippet" then, by default and when enabled, the temporary snippet file produced will be "INSERT_KERB.mtfsnippet.temp_mtf".

This env variable enables the user to have this consistently named temporary file open in an editor and it can then be refreshed each time a snippet is run.

(c) Snippet Temp File Custom Extension

MTF_SNIPPET_TEMP_FILE_EXTENSION_4D *Extension_text*

If ***Extension_text*** is **not blank**, the temporary snippet file generated when MTF snippets have been processed will have the extension *Extension_text* rather than the default extension of "temp_mtf".

(d) Unique Snippet Temp File Name

MTF_SNIPPET_TEMP_FILE_UNIQUE_4D 1/0

If enabled (non zero), a temporary snippet file will be generated for each MTF Snippet instance in the corresponding MTF. To better relate the temporary snippet file with the actual instance in the MTF, this setting will produce a temporary snippet file for every inserted snippet and be named like so:

<Snippet_File> - <MTF> <side> <modifier_number>.<extension>

For example,

CP.mtfsnippet - RSCR01.mtf Left No. 7.temp_mtf

Corresponds to the CP.mtfsnippet file inserted in the left side modifiers of RSCR01.mtf file as modifier number 7.

These environment variables are defined in the ***MTF & Boxing >MTF General*** section of the ***env.4d Editor*** which is brought up by selecting ***Project =>Management =>env.4d.*** *See* MTF

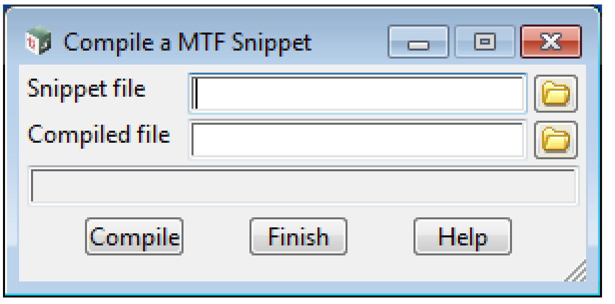
& Boxing > MTF General in 7 Project.

Continue to the next section21.5.16 Compiling Snippets or return to 21.5.15 Debugging Snippets or 21.5 Defining and Using Snippets.

21.5.16 Compiling Snippets

It is possible to compile a snippet via the option

Design =>MTF =>Snippet =>Compile snippet



This can be done to protect any intellectual property (IP) and minimise concerns people may have in developing and sharing snippets.

Compiling a snippet encrypts and encodes the snippet contents into a new file with an extension of **mtfsnippetc** (note the 'c' at the end, denoting a compiled snippet).

A compiled snippet can be run and used by anyone, but the snippet code itself cannot be viewed or edited.

Temporary snippet files, if enabled, are not generated for compiled snippets.

Original Snippet

// INFO Example compiled snippet

Compiled Snippet

nyKgSBlhX6wGqQizY/fRfDcT/YReG2J34wsghywz6Ys+cfVn+Pi+crz/4c9XgNbYI+kmS14F

flh+I/bNu41RYyD4RtNvIiS9

Continue to the next section21.5.17 Tips and Tricks or return to 21.5.15 Debugging Snippets or 21.5 Defining and Using Snippets.

21.5.17 Tips and Tricks

• When creating snippets, create the MTF modifier commands first and then copy and paste into a snippet. This helps ensure the format and syntax of the modifier commands is correct.

• Use comments liberally to document your snippet. This makes it easier for you and anyone else to see and understand your thought processes, reasoning and logic within the snippet, making it easier to debug and modify.

• Define tokens for commonly-used values to make code more readable and portable.

```
@ def_tok ABS "absolute"
@ def_tok ES "extra_start"
@ def_tok EE "extra_end"
@ def_tok ASE "$(ABS) $(ES) $(EE)"
Insert "LINK" "red" 3.5 unknown -3 $(_SCH) 0 $(_ECH) 0 $(ASE)
```

• When writing or testing snippets, output messages using user_message commands with values of parameters, variables and tokens to check correct calculation of values. These can be commented out or removed before finalising your snippet.

• The user_message commands will, by default, output a message for every applicable Apply MTF section. To cut down on the number of repetitive messages, specify a start and end chainage.

```
// PARAMETER WIDTH REAL "Link width" 1.0
@ def_tok MSG_CH "$(_SCH) 0 $(_SCH) 0.0001"// Define a token for easier readability
user_message "Link width = $(WIDTH)" $(MSG_CH)// Only output one message
```

• For all directives, there is a space between the @ and the name of the directive!

• For expressions and formula, use parentheses as much as needed. If in doubt, add more and double-check the results

• Parameter names should not contain spaces and should not start with an underscore.

• Token names should not contain spaces and should not start with an underscore.

• It is recommended that token names only contain alphanumeric characters and underscores.

• Parameter names and token names must be unique within the snippet file!

• Use the new V11 syntax, where possible! The V10 syntax- e.g. #define, $VAL - is deprecated and only supported for backwards-compatibility.

Continue to the next section 21.5.18 Major Warning - You Will be Caught by This or return to 21.5 Defining and Using Snippets.

21.5.18 Major Warning - You Will be Caught by This

WARNING - once a snippet with parameters has been saved in a MTF file, the parameters for the snippet are saved ***in the MTF file.*** Then when the Apply MTF is run, the parameters and their values are read in from the MTF file and passed into the current snippet file.

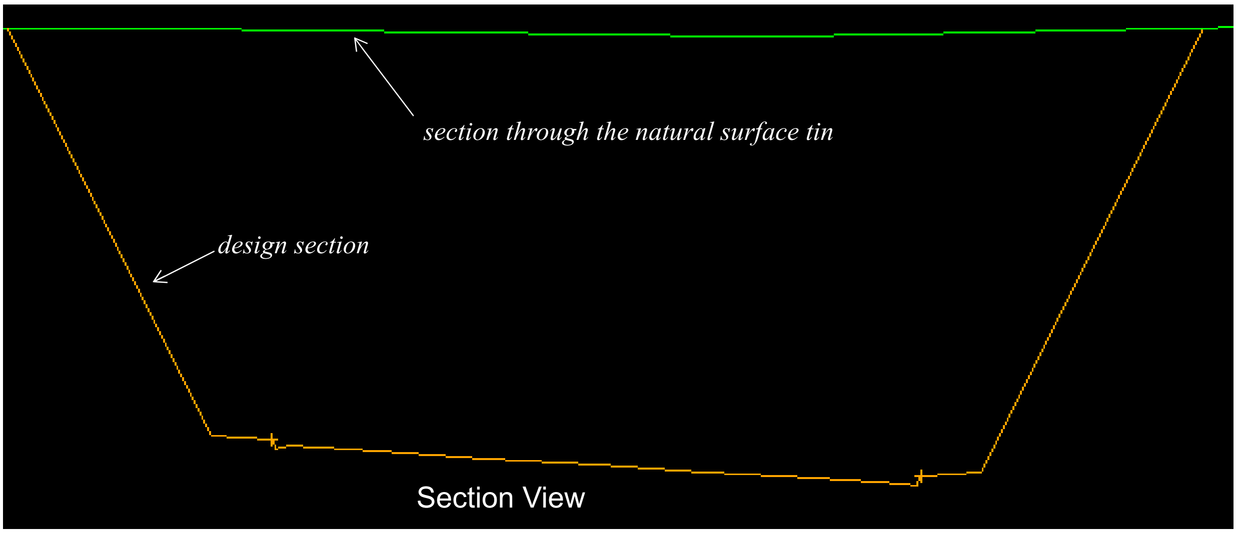
Hence if the snippet file has been modified and parameter definitions added, modified or removed since the snippet was originally place in the MTF, these modifications of parameters will be **ignored** until the snippet panel is reopened in the MTF editor which forces the snippet file to be parsed again and the changes to the parameter definitions are then recognised.

Note that when the Apply MTF for the MTF is run, the MTF used the ***current*** snippet file so if the snippet file has been modified since the snippet was placed in the MTF, the ***modified*** snippet file will be used (apart from any new parameter definitions and defaults)

Return to 21.5 Defining and Using Snippets.

21.6 What is Boxing?

A design is represented in 3d as a collection of strings. The design can also be approximated as a series of sections perpendicular to the centreline at various chainages. These are known as design cross section, or design sections.



See

[21.6.1 Representing Cross Sections in 12d Model](#)
[21.6.2 Applying Boxing](#)
[21.6.2 Applying Boxing](#)
[21.7 Full Definition of Boxing](#)

Go to the next section 21.6.1 Representing Cross Sections in 12d Model

21.6.1 Representing Cross Sections in 12d Model

In ***12d Model***, cross sections, are represented as ***4d super strings***.

A 4d super string has an (x,y,z) coordinate for each vertex, PLUS text at each vertex. Hence, there are four bits of information, an (x,y,z,text) value, for each vertex.

Cross sections are generated in ***12d Model*** by options such as

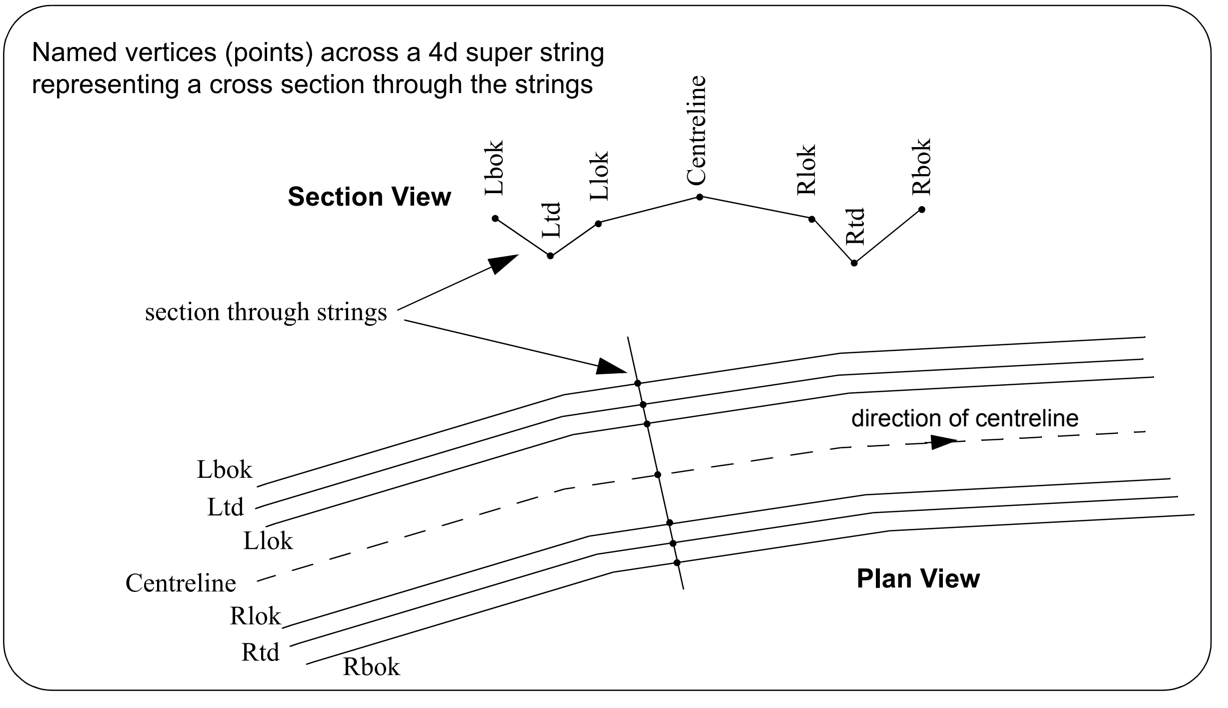
(a) cuts through strings along a centreline (see 28.9.13.2 Cuts by Centreline)

(b) the functions **Apply MTF** (see 20.3.2 Apply MTF Function) or **Apply** (see 20.3.1 Apply Template Function)

(c) the **Boxing Many Function** (see 20.5.4 Boxing Many Function)

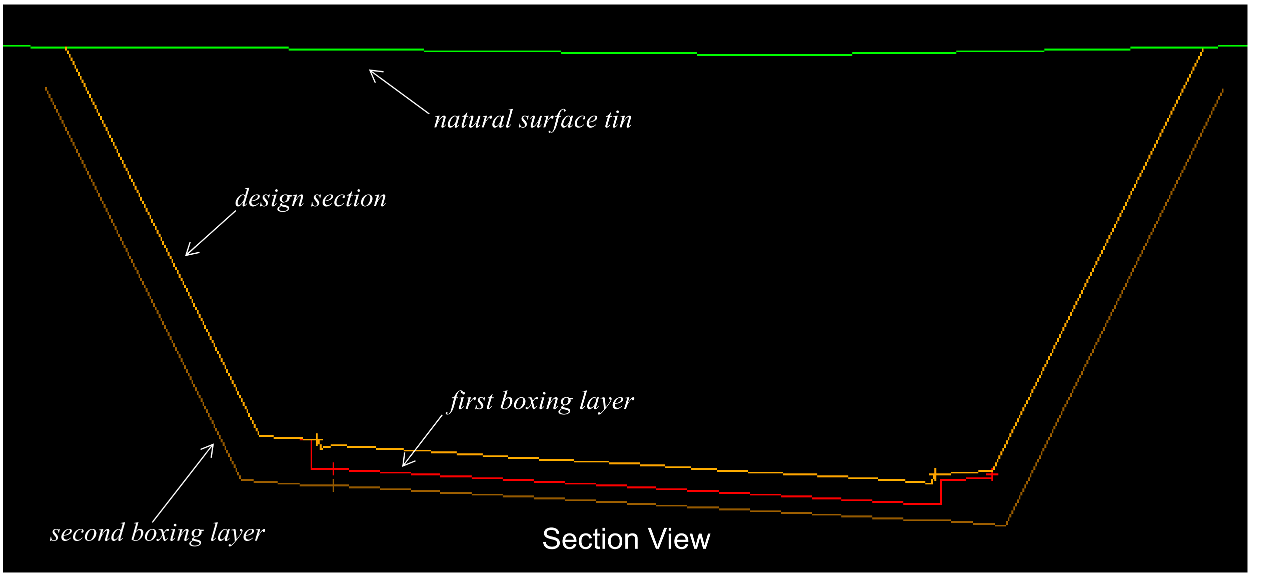
For these cross sections, the **text** at each vertex of the 4d super string (vertex/point text), is the **name of the string** that the section cuts through at that vertex.

Also when the cross sections are generated by these options, the start chainage of the 4d string is set so that the chainages **along the 4d string** are the same as the **offset** of the vertices from the centreline used when generating the sections.



Boxing Layers

A Boxing Layer is simply another surface, usually modelling a construction material layer, under (or above) the design. The Boxing Layer is represented by strings, and also as a series of sections at the same chainages as the design sections.



Boxing Layers are normally defined in terms of the named vertices across the design section, offsets from the centreline, in terms of other boxing layers already specified, or in relationships to other strings.

When boxing is generated in the **Apply MTF** or the **Boxing Many Function**, there can be up to **eight** (8) layers of boxing and the ***last*** *layer* defined is referred to as the ***subgrade*** layer.

In ***12d Model*** there are special commands called ***Boxing Commands*** that are be used to build a boxing layer.

A series of ***12d Model*** boxing commands are grouped together as a ***Boxing Definition*** and when a *Boxing Definition* is applied to a cross section, it generates a boxing section. That is, it generates a section that is part of a particular Boxing Layer.

Each ***Boxing Definition*** is given a name, and is stored in a ***Boxing*** file. The names of the *Boxing Definitions* must be unique within the *Boxing* file.

How the **Boxing** file and **Boxing Definitions** are applied in ***12d Model*** is described in the next section 21.6.2 Applying Boxing, and this is followed by the full definition of ***12d Model*** boxing and the ***12d Model*** boxing commands in the sections starting with 21.7 Full Definition of Boxing.

21.6.2 Applying Boxing

For any ***12d Model*** project there are a number of different data creation scenarios and the appropriate method to use to generate boxing depends on how the data has been created, or is available, inside ***12d Model.***

See 21.6.2.1 Scenario 1 - Design Generated by One Apply MTF

See 21.6.2.2 Scenario 2 - More than One Apply MTF Needed to Generate the Design

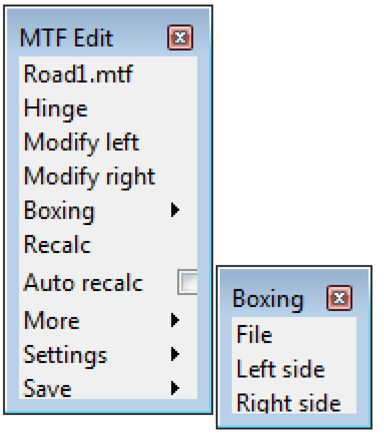
See 21.6.2.3 Scenario 3 - Design Provided as Strings Only

See 21.6.2.4 Scenario 4 - Design Sections Already Exist

21.6.2.1 Scenario 1 - Design Generated by One Apply MTF

If a design is generated by just ***one Apply MTF*** function, then up to eight (8) layers of boxing can also be generated at the same time by the ***Apply MTF*** function.

In the **Apply MTF** option, the ***MTF***, controls which ***Boxing*** file is used, and how the various Boxing Definitions from that file are used to create each layer of boxing.



In the **MTF Edit** menu, the *Boxing* ***File*** option specifies which **Boxing** file of ***Boxing Definitions*** is used.

For each boxing layer, the boxing can be defined by

(a)the Left boxing only

(b)the Right boxing only

or

(c)the Left and Right boxing.

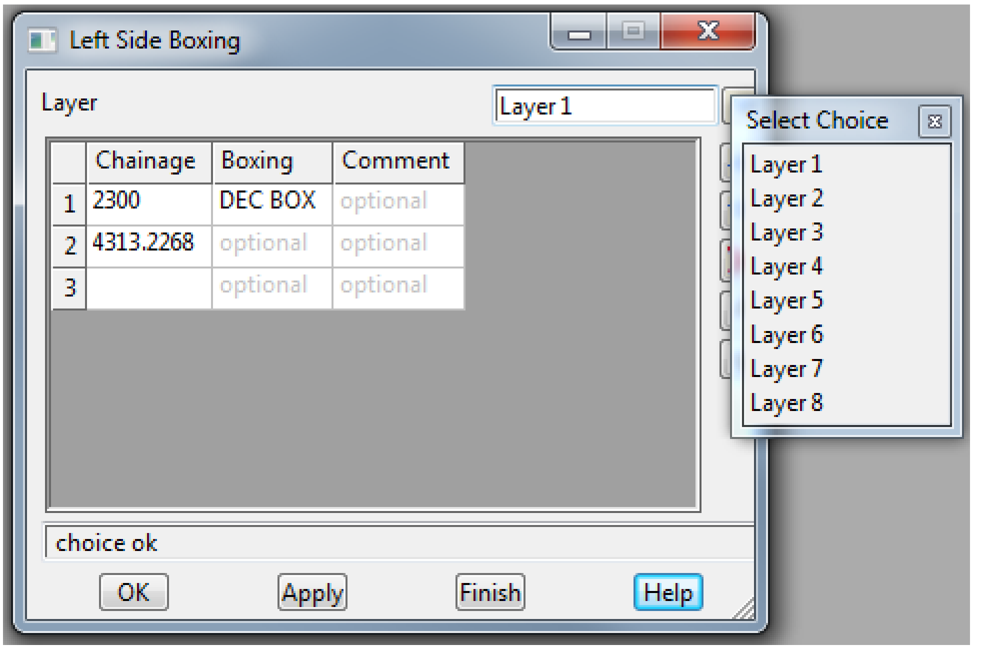
For cases (a) and (b), the Left boxing or the Right boxing defines the boxing across the entire design section and then only Left boxing or Right boxing is needed.

For case (c) where **both** Left and Right boxing are used to define the boxing across the design section, the ***last*** *point of the* ***Left*** *boxing* is automatically **connected** to the ***first point*** *of the* ***Right*** *boxing*.

Warning for case (c), the Left boxing must end before the Right boxing begins. If there is an overlap, the Right boxing will be pushed to the **end** of the Left boxing.

Note - no interpolation or modifiers exist for boxing.

For each layer, the ***Left side*** /***Right side*** specifies what Boxing Definition to use over what chainage range.



See 21.1 The Modifiers and Templates File - MTF and 21.4 Smart Chainages.

Go to the next section 21.6.2.2 Scenario 2 - More than One Apply MTF Needed to Generate the Design or back to 21.6.2 Applying Boxing.

21.6.2.2 Scenario 2 - More than One Apply MTF Needed to Generate the Design

In more complex designs, a number of separate ***Apply MTF'***s may be needed, and even some strings hand crafted, to produce the final design. Hence the design is only fully represented by a ***collection of strings***.

In this situation, design sections are produced as a ***post process*** using ***Cuts through strings*** by either

Utilities =>A-G =>Cuts =>by centreline

which can cut through a Data Source of strings (see 28.9.13.2 Cuts by Centreline), or by

Design =>Boxing =>Boxing many (function)

which can only cut through one model of strings (see 20.5.4 Boxing Many Function).

Note - the design sections produced by both options are ***4d super strings*** where the text at each vertex of the super string is the **name** of the string **cut** to create that vertex of the super string.

The methods described in 21.6.2.4 Scenario 4 - Design Sections Already Exist are then used to generate boxing from the created design sections.

Go to the next section 21.6.2.3 Scenario 3 - Design Provided as Strings Only or back to 21.6.2 Applying Boxing.

21.6.2.3 Scenario 3 - Design Provided as Strings Only

Sometimes you only have the strings defining the design (the design strings).

This case is similar to 21.6.2.3 Scenario 3 - Design Provided as Strings Only and **design sections** are produced as 4d super string from the design strings using ***Cuts through strings*** by either

Utilities =>A-G =>Cuts =>by centreline

which can cut through a Data Source of strings (see 28.9.13.2 Cuts by Centreline), or by

Design =>Boxing =>Boxing many (function)

which can only cut through one model of strings (see 20.5.4 Boxing Many Function).

The methods described in 21.6.2.4 Scenario 4 - Design Sections Already Exist are then used to generate boxing from the created design sections.

Go to the next section 21.6.2.4 Scenario 4 - Design Sections Already Exist or back to 21.6.2 Applying Boxing.

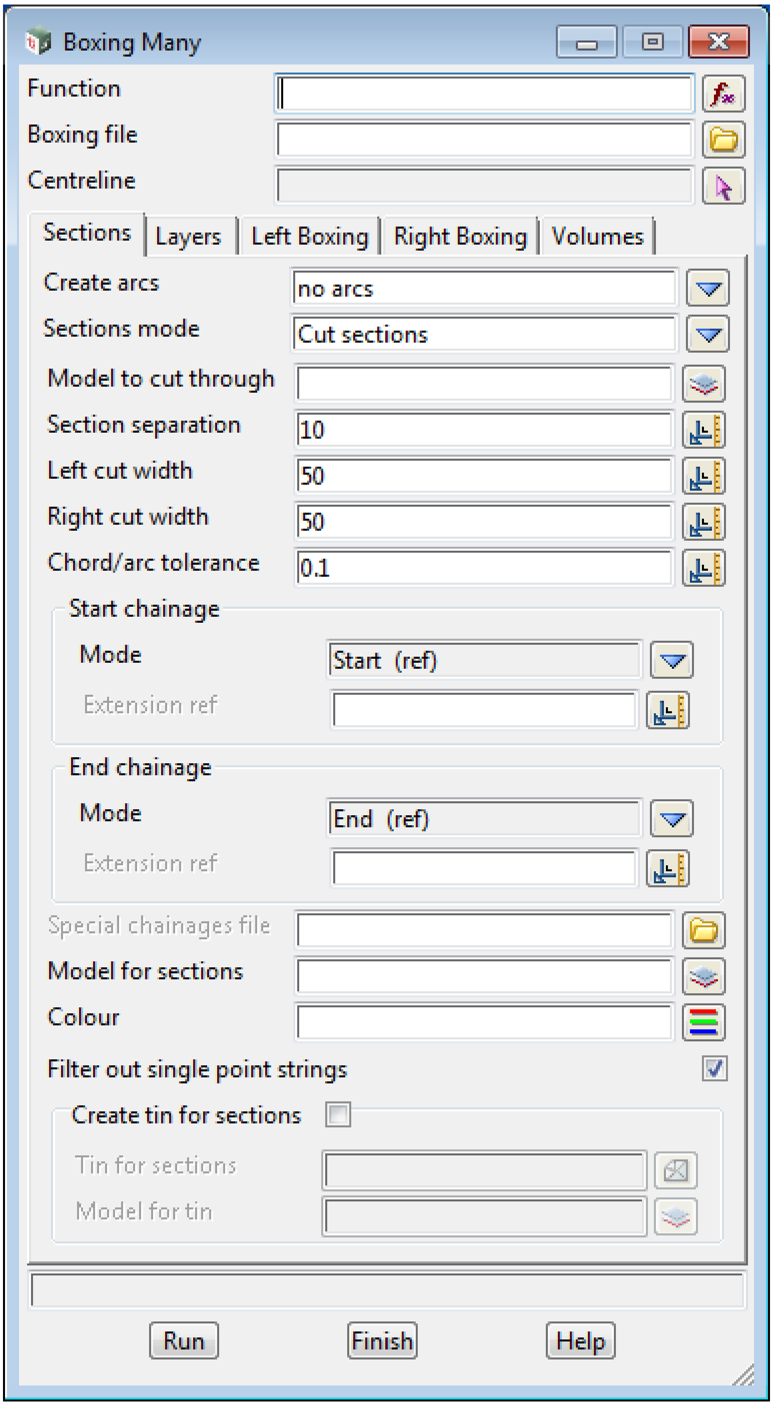
21.6.2.4 Scenario 4 - Design Sections Already Exist

There is the case where you already have the design sections as 4d super strings.

The design sections may have been created by an earlier ***Apply MTF*** or ***Apply*** run, or generated by cuts through strings as in 21.6.2.2 Scenario 2 - More than One Apply MTF Needed to Generate the Design or 21.6.2.3 Scenario 3 - Design Provided as Strings Only, or simply provided.

Boxing can then be generated from the ***design sections*** using the options:

(a) *Design =>Boxing =>Boxing many function*



The ***Boxing Many Function*** can generate up to eight layers of boxing, as well as calculating volumes. See 20.5.4 Boxing Many Function.

Note that the ***Boxing Many Function*** can also create the cuts through a model of strings to create the design sections and so can be used to cover 21.6.2.2 Scenario 2 - More than One Apply MTF Needed to Generate the Design. 21.6.2.3 Scenario 3 - Design Provided as Strings Only and 21.6.2.4 Scenario 4 - Design Sections Already Exist in the one function.

(b) *Design =>Boxing =>Boxing many*

A simplified version of the *Boxing Many Function.*

For *Boxing Many*, an MTF is used to define the application of the Boxing Definitions, and the Boxing file containing the Boxing Definitions is either that defined in the MTF **or** a different Boxing file. See 20.5.6 Apply Boxing Many.

The boxing can be applied to a model of sections, or just a **single selected section**.

(c) *Design =>Boxing =>Boxing*

An even more simplified version of the *Boxing Many Function*.

The ***Boxing*** option can only apply one Boxing Definition to all the design sections in a model. See 20.5.5 Apply Boxing.

Note - in the case when only 4d super string are provided, the option

Design =>X-Sections =>Strings from sections

can be used to generate design strings from these sections (see 20.14.8 Strings from Sections).

Go to the next section 21.7 Full Definition of Boxing or back to 21.6 What is Boxing? or 21.6.2 Applying Boxing.

21.7 Full Definition of Boxing

In ***12d Model*** there are special commands called ***Boxing Commands*** that are be used to build up a boxing layer.

A series of ***12d Model*** boxing commands are grouped together as a ***Boxing Definition*** and when a *Boxing Definition* is applied to a cross section, it generates a boxing section. That is, it generates a section that is part of a particular Boxing Layer.

Each ***Boxing Definition*** (the rules for calculating a specific type of boxing) is given a name, and is stored in a ***Boxing*** file (with a file name ending in **.bf**).

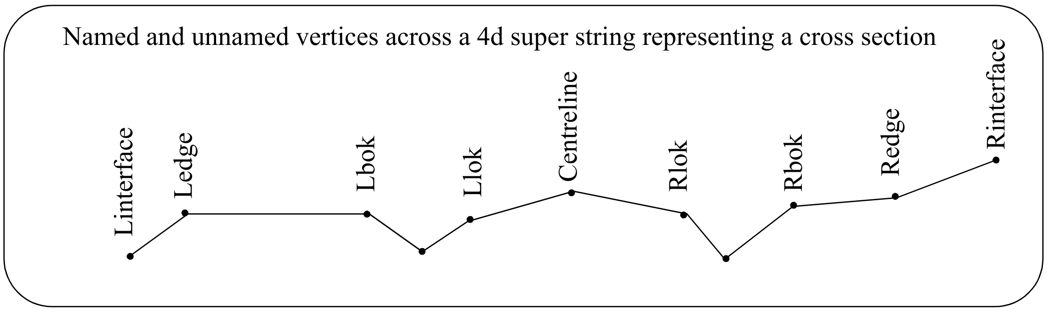
Any number of *Boxing Definitions* can be placed in the one *Boxing* file as long as each *Boxing Definition* in the ***Boxing*** file has a unique name.

Note that the same name of a *Boxing Definition* can appear in more than one *Boxing* file and even through they have the same name, the *Boxing Definition* in a different *Boxing* file can be completely different.

Please continue to the next section 21.7.1 Terminology Used in Boxing Commands

21.7.1 Terminology Used in Boxing Commands

The ***12d Model*** *Boxing Commands* work on a cross section, which is a 4d super string (see 21.6.1 Representing Cross Sections in 12d Model).



A *Boxing Definition* is a group of boxing commands and the boxing commands work from **left to right** across the cross section to produce a boxing section.

Most of the **boxing commands** apply from the **offset** specified at the beginning of that command (**Start Offset** for the command) to the start offset of the **next** boxing command (like chainages and templates in the MTF). That is, the start offset of a boxing command is the **End Offset** for the ***previous*** boxing command.

So most **boxing commands** go from the **Start Offset in the command** to the **Start Offset in the next boxing command**.

An **offset**, such as the **Start Offset**, can be defined as:

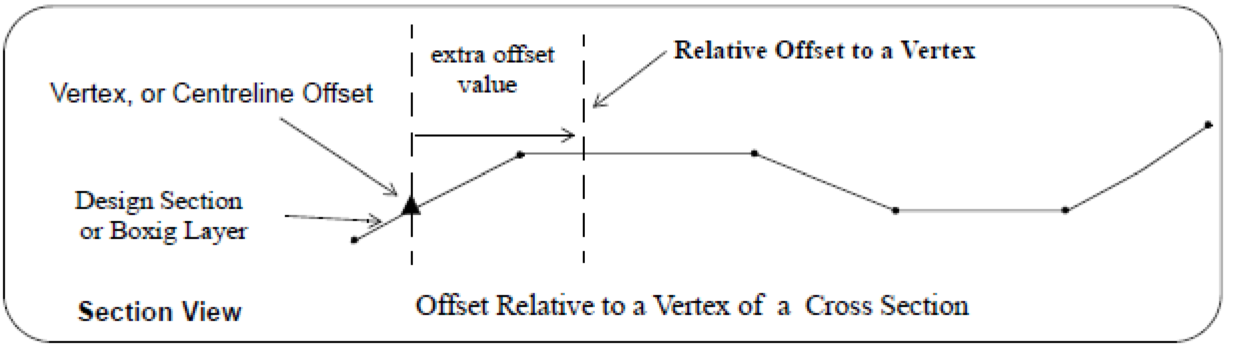
(a) a **fixed** Centreline offset value (*e.g.* 10.5)

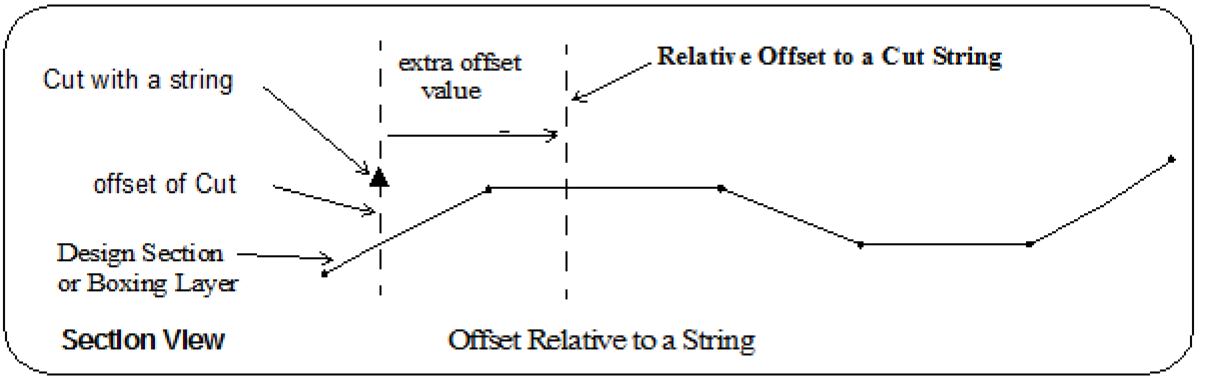
(b) specified **relative to a vertex name** on a design cross section, or any boxing section (at the section chainage) in a Boxing Layer already created.

(c) specified **relative to a selected string** that cuts the design cross section, or a specified Boxing Layer, in plan.

The **offset** represents a **plan** position on the cross section (a plan point on the cross section) which may or may not be the plan position of a vertex on the cross section.

The terminology used to specify such an offset is ***relative offse***t to a ***vertex*** or a ***string***.





Note - if not all the vertex names and strings mentioned in a *Boxing Definition* are present in the cross section at the centreline chainage, then no boxing section is produced for that centreline chainage.

Hence

1. an **offset relative to a vertex name** (**relative vertex offset**) is defined to be the **offset** at a **given named vertex** on the design section, or a boxing section previously created, **plus** a **given offset value**.

   That is

   take the **offset at** the 4d vertex called **"vertex name"** and add **"offset value"** to it. The offset value can be positive of negative.

   If the "vertex name" is blank, then the "**offset value**" is taken to be the *actual* offset on the 4d super string.

   Hence the *offset* can be specified as either an actual ***offset value*** or given **relative to the name of a vertex** on the super string (**relative offset**).

   For convenience, in both cases the offset will be denoted by **relative vertex offset**.

2. an **offset relative to string** (**relative string offset**) is defined to be the **offset** on the cross section where the string cuts the section in plan, **plus** a **given offset value**.

   That is

   take the **offset at** the plan cut of the section and the string and add **"offset value"** to it. The offset value can be positive of negative.

A major advantage of defining boxing in terms of vertex names and strings is that whenever the vertices or strings are modified using MTF modifiers, or string edits, the boxing across the section is also automatically modified on a recalc.

A *Boxing Definition* may be used in either a *Left Boxing* or *Right Boxing* in an **MTF**, or in a **Boxing Many Function** option, however unlike the case for MTF modifiers, the *Boxing Definition* is applied to the vertex names going from **left to right** regardless of whether the *Boxing Definition* is used on the left or the right. So it is possible to create the entire boxing with just the *Left Boxing* or *Right Boxing*. If both are used, the end of the *Left Boxing* is joined to the start of the *Right Boxing*.

Warning

For a *Boxing Definition* to work, any cross section vertex names referred to in the *Boxing Definition* must be **unique** for that cross section. Similarly any strings cut must have unique names.

In the *Apply MTF* and *Apply*, the **LHS prefix** and **RHS prefix** panel files can be used to give vertices on cross sections different names even when the same Template is used on the left and the right side of the centreline.

Boxing ***sections*** are generated by the *Apply MTF*, *Boxing Many Function, Boxing Many and Boxing options,* but ***boxing strings*** are only generated by the *Apply MTF* and *Boxing Many Function*. For the other cases, the option **Strings=>Utilities=>Strings from sections** can be used to create boxing strings from the boxing sections.

Vertical Walls

If a *Boxing Definition* creates two different height values at the one offset (i.e. a wall), then an extra offset point is automatically inserted into the boxing section (at offset 0.1 mm from the first vertex) so that **no vertices in the created boxing section are on top of each other.**

So in effect a **vertical wall** is created.

And since no vertices are created that are directly above each other, the boxing sections can be triangulated.

Please continue to the next section 21.7.2 Edit Boxing File or return to 21.7.1 Terminology Used in Boxing Commands.

21.7.2 Edit Boxing File

The option

Design =>Boxing =>Create

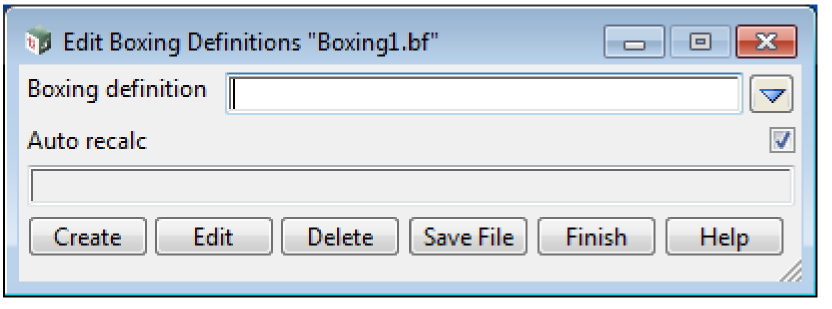
creates a ***Boxing*** file which contains ***Boxing Definitions***, and then brings up the **Edit Boxing Definitions** panel to create the actual *Boxing Definitions* (rules) to save in the *Boxing* file.

Similarly selecting a *Boxing* file from the walk-right list of

Design =>Boxing =>Edit

also brings up the **Edit Boxing Definitions** panel to create/edit boxing definitions from the selected *Boxing* file.

The **Edit Boxing Definitions** panel is



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Boxing definition	input		boxing definitions in file

name of the Boxing Definition (in the current boxing file) to create or edit. Each Boxing Definition must have a unique name in the Boxing file

Create button

create a Boxing Definition with name given by the **Boxing definition** *panel field*
If the Boxing Definition given in the **Boxing definition** *field* ***does not exist****, then the* **Boxing Rules** *panel is placed on the screen and is used to created the new Boxing Definition for the Boxing file (see* 21.7.2.1 Boxing Rules Panel*).*

If the Boxing Definition already exists, then nothing will happen when clicking on **Create***.*

Edit button

edit an existing Boxing Definition with name given in the **Boxing definition** *panel field*
If the Boxing Definition given in the **Boxing definition** *field* ***exists****, then the* **Boxing Rules** *panel is placed on the screen and is used to edit the existing Boxing Definition (see* 21.7.2.1 Boxing Rules Panel*).*

If the Boxing Definition does not exist, then nothing happens when clicking on **Edit***.*

Delete button

delete the existing Boxing Definition with the name given in the **Boxing definition** *panel field*
A **Yes-No** *panel confirms the deletion.*

Save file button

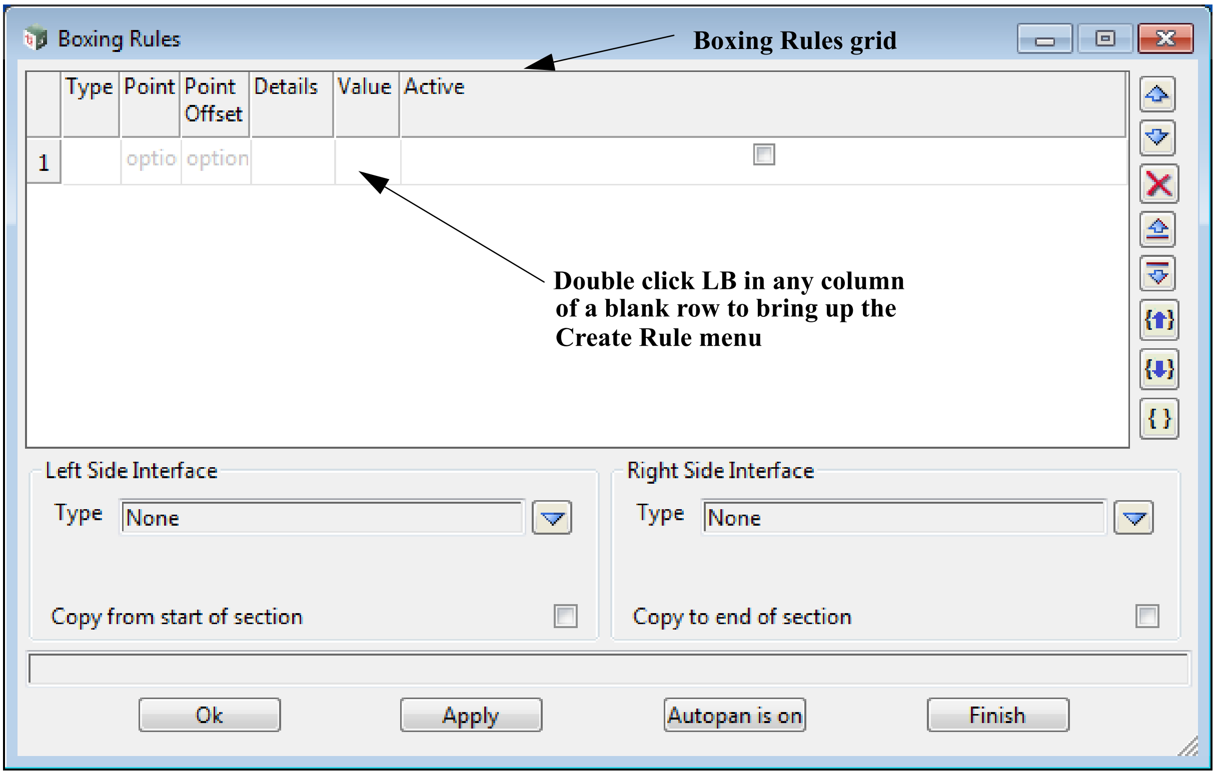
save the Boxing file with the Boxing Definitions to disk.

For more information on how Boxing works in ***12d Model***, go to the section 21.6 What is Boxing?.

Go to the next section 21.7.2.1 Boxing Rules Panel or return to 21.7.2 Edit Boxing File.

21.7.2.1 Boxing Rules Panel

Selecting the **Create** or **Edit** button on the **Edit Boxing Definitions** panel brings up the **Boxing Rules** panel which is used to build up/edit the nominated *Boxing Definition* from the **Boxing Commands.**



The **Boxing Rules** panel consists of:

(a) a **Boxing Rules** grid with rows (or lines) with **commands** in them

The commands for the Boxing Rules grid are fully described in the section 21.7.2.2 Commands for Boxing Rules Grid.

For more information on creating and editing data in a grid, see 4.19.6 Grids in Panels.

(b) **Left Side Interface** and **Right Side Interface** areas

See the following document ion on fields and buttons for this panel

(c) **Copy from start of section** and **Copy to end of section** tick boxes

See the following document ion on fields and buttons for this panel

(d) an **OK** or **Apply** button to record the results.

See the following document ion on fields and buttons for this panel

The fields and buttons used in this panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Boxing Rules Grid

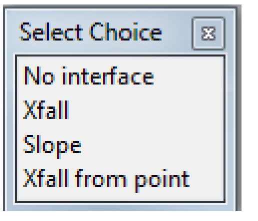
commands for defining the Boxing Definition.

The commands will be described in the section 21.7.2.2 Commands for Boxing Rules Grid.

For information on the general operation of a grid including the icons on the right hand side, see 4.19.6 Grids in Panels.

Left Side Interface

Type choice box



*If **No interface:***

nothing is done

*If **Xfall:***

Xfall real box 0

*if non-zero, a batter with the given xfall is applied going to the **left** from the first point of the boxing, and goes until it intersects the design section. The intersection point is added as the first point in the boxing. If no intersection is made, nothing is added to the boxing. A positive xfall is up and negative xfall down.*

*If **Slope:***

Slope real box 0

*if non-zero, a batter with the given slope is applied going to the **left** of the first point of the boxing, and goes until it intersects the design section. The intersection point is added as the first point in the boxing. If no intersection is made, nothing is added to the boxing. A positive slope is up and negative slope down.*

*If **Xfall from point:***

Set point button

clicking on **Set point** *brings up the* **Boxing LHS Xfall** *panel which allows the battering xfall to be defined in more complex ways. This is described in the section* 21.7.2.2.17 Boxing Left Side Interface Xfall from Point.

Copy from start of section tick box

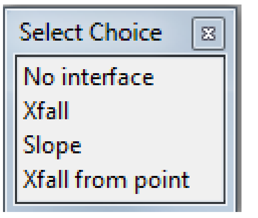
if ticked, then the part of the design section from the start of the design section to the offset that is the start of the boxing section, is copied and made the beginning of the boxing section.

*Hence the **Copy from start of section** tick is used to begin the boxing section with the start of the design section to make a full width left side boxing section.*

If not ticked, nothing is done

Right Side Interface

Type choice box



*If **No interface:***

nothing is done

*If **Xfall:***

Xfall real box

if non-blank, a batter with the given xfall is applied to the last point of the boxing and goes until it intersects the design section. The intersection point is added as the last point in the boxing. If no intersection is made, nothing is added to the boxing. A xfall slope is up and negative xfall down.

*If **Slope:***

Slope real box

if non-blank, a batter with the given slope is applied to the last point of the boxing and goes until it intersects the design section. The intersection point is added as the last point in the boxing. If no intersection is made, nothing is added to the boxing. A positive slope is up and negative slope down.

*If **Xfall from point:***

Set point button

clicking on **Set point** *brings up the* **Boxing RHS Xfall** *panel which allows the battering xfall to be defined in more complex ways. This is described in the section* 21.7.2.2.18 Boxing Right Side Interface Xfall from Point.

Copy to end of section tick box

if ticked, then the part of the design section from the offset of the last boxing vertex to the last vertex on the design section, is copied and made the end of the boxing section.

Hence the RHS copy flag is used to continue the boxing section from the last boxing point, with the design section to the end of the design section to make a full width right side boxing section.

If not ticked, nothing is done.

Buttons at Bottom

OK button

OK *stores the values in the fields and removes the panel BUT no **recalc** is done.*

Apply button

Apply *stores the values and leaves the panel on the screen.*

*If the **Boxing Definition** is being used in an **Apply MTF MTF** and **Auto recalc** is ticked in the MTF, then whenever the **Apply** button is clicked, then a **recalc** of the associated **Apply MTF** for the MTF is done.*

*If the **Boxing Definition** is being used in a **Boxing Many Function** then whenever the **Apply** button is clicked, a **recalc** of the **Boxing Many Function** is done.*

Notes

1. When a Boxing Definition is applied on the Left side in the MTF of an **Apply MTF**, or the Left side in a **Boxing Many Function**, any *Right Side Interface* and *Copy to end of section* is ignored. Similarly, when a Boxing Definition is applied on the right side the MTF of an **Apply MTF**, or the Right side in a **Boxing Many Function**, any *Left Side Interface* and *Copy from start of section* is ignored.

   Hence a boxing definition can have all of *Left Side Interface, Copy from start of section and Right Side Interface, Copy to end of section* but which set is used depends on whether the boxing definition is used on the Left or the Right.

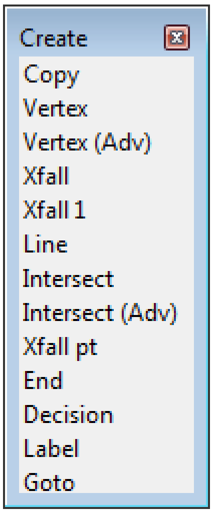
   In **Apply MTF**'s and **Boxing Many Function**s, if only the Left side boxing or only the Right side boxing exists, then the boxing for the entire design section can be created by just the Left/Right side boxing and then **all** of the *Left Side Interface* and the *Right Side Interface* is used.

For more information on how Boxing works in ***12d Model***, please go to the section 21.6 What is Boxing?

21.7.2.2 Commands for Boxing Rules Grid

For information on the general operation of a grid including the icons on the right hand side, see 4.19.6 Grids in Panels.

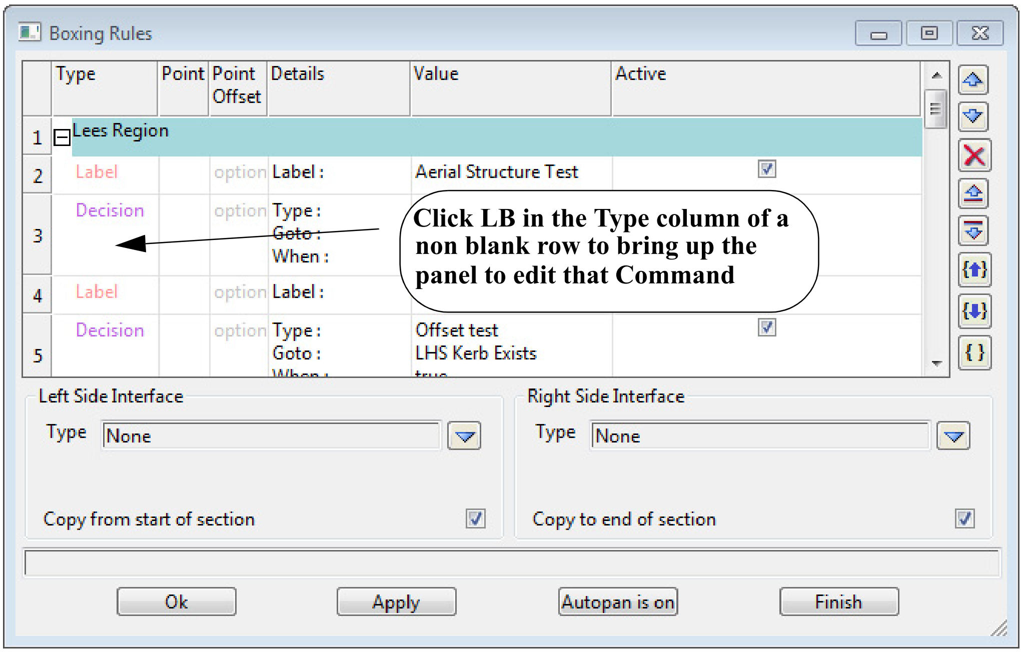
If the row of the grid is **empty**, clicking LB in the empty row will bring up the **Create** menu which contains all the available boxing commands.Note this may involve two clicks - one to highlight a column in a row and the second click to bring up the **Create Rule** menu.



Selecting a menu item will bring up an associated panel which displays the information required for the boxing command.

When the panel is filled in and **OK** or **Apply** selected, the panel information is written out to the **command line** in the correct format for that boxing command.

If the row of the grid is **not empty** (and hence filled with a **command**) clicking LB in the **Type** column will bring up the associated panel for the boxing command. Note this may involve two clicks - one to highlight the **Type** column in a row and the second click to bring up the panel to edit that **Command**.



The information in the panel can be changed and if **OK** or **Apply** is selected, the modified panel information is written out to the row a a **command line**.

Each of the panels created by selecting a command from the **Create** menu, will now be described.

For

Copy, go to 21.7.2.2.1 Boxing Copy
Vertex 21.7.2.2.2 Boxing Vertex
Vertex (Adv) 21.7.2.2.3 Boxing Vertex Advanced
X-fall 21.7.2.2.4 Boxing Xfall - line through a point with a given crossfall
X-fall 1 21.7.2.2.5 Boxing Xfall 1 - line through a point with a crossfall taken from a point
Line 21.7.2.2.6 Boxing Line - line through two points
Intersect 21.7.2.2.7 Boxing Intersect - Intersection of Two Lines
Intersect (Adv) 21.7.2.2.8 Boxing Intersect Advanced
Xfall pt 21.7.2.2.9 Boxing Xfall Point
Drop (kerb) 21.7.2.2.10 Boxing Drop
Decision 21.7.2.2.11 Boxing Decision
Label 21.7.2.2.12 Boxing Label
Goto 21.7.2.2.13 Boxing Goto
Comment 21.7.2.2.14 Boxing Comment
Extend tin 21.7.2.2.15 Boxing Extend Tin
End 21.7.2.2.16 Boxing End

The text file format of all the Boxing Commands in the *Boxing* file is given in the section 21.10.2 Text Format of the Boxing File.

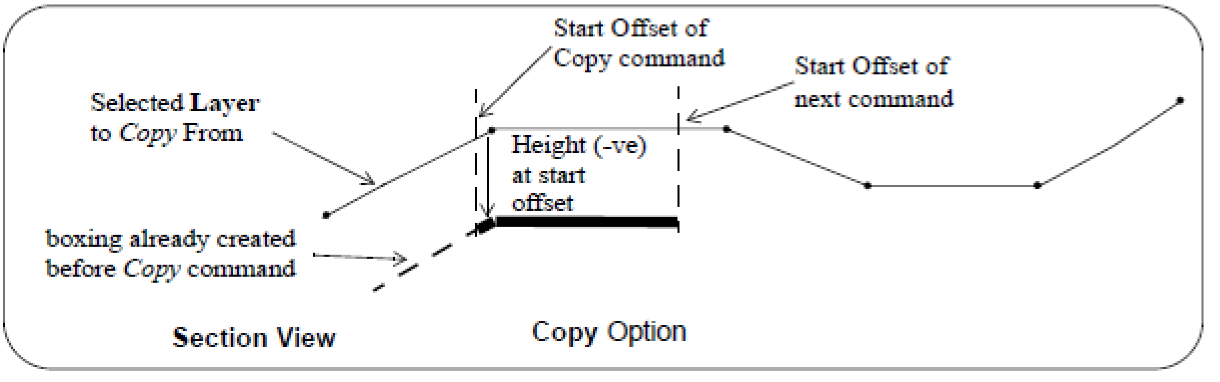
Go to the next section 21.7.2.2.1 Boxing Copy or return to 21.7.2.2 Commands for Boxing Rules Grid.

21.7.2.2.1 Boxing Copy

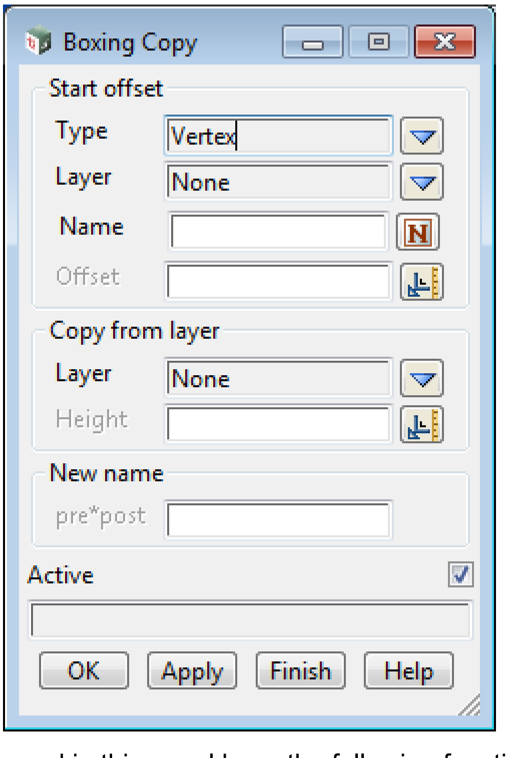
The *Copy* command copies part of the selected **Layer** from the **Start Offset** defined for the *Copy* command, to the **Start Offset of the *next* boxing command**.

The present slope of each segment is kept but the ***height of each segment is adjusted*** by a ***constant*** value provided by user.

Note: The height adjustment may be zero so that part of the selected **Layer** is copied as is.



Selecting **Copy** brings up the **Boxing Copy** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

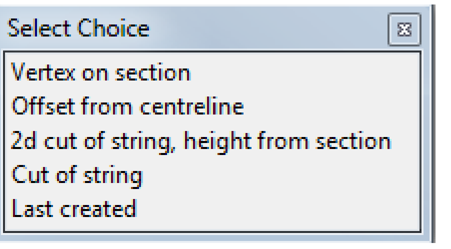
Start Offset

Start Offset defines where the Copy commands starts from.

If the 3d cut of string is used (Cut of string), the Z-Diff is used with **Height** *in determining how far the*

Layer is copied.

Type choice box



*For the documentation on the methods of defining the **Start Offset** except for* **Last created**, *see* 21.7.2.3 Defining Relative Offset for Boxing Commands

*If **Last created:***

*the Copy command starts at the Start Offset but for the boxing commands **Intersect** and **Intersect Adv**, the offset position for the vertex is calculated by intersection and its actual value may not be known.*

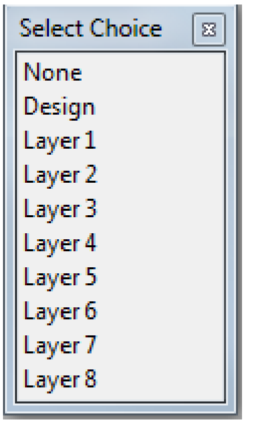
*So for **Last created**, the offset is the offset of the **last vertex** of the selected* **Layer**.

Offset real box 0

offset to add to the offset of the last vertex. **Offset** *can be positive or negative but the final Start Offset must still be on* **Layer**.

Copy from layer

Layer choice box



Height real box 0

*If Start Offset Type is NOT: **Cut of string:***

Height is added to the z-values of the part of the section of the Copy from layer **Layer** *being copied to give the z-values of the boxing. For example, if Height is -0.3, the part of the* Copy from layer **Layer** *being copied is dropped by 0.3*

*If Start Offset Type is: **Cut of string:***

At the plan cut of the string and the START OFFSET **Layer**, ***Z-Diff (****the difference of the z-values between the string and START OFFSET* **Layer*)*** *is calculated and* **Height** *added to it. This combined value is added to the z-values of the part of the COPY FROM LAYER* **Layer** *being copied, to give the z-values of the boxing*

New name

Pre*post text input

when part of a section is copied for the boxing, the vertices of the boxing are given names from either the vertices of the section, or the selected strings that are used in limiting the section, or a unique number not related to vertices or strings.

These constructed boxing vertex names are then modified by the text given in the **Pre*post** *field.*

*The text before the * is prepended to the vertex names, and the text after the * is postpended to the vertex names.*

Active tick box

*if **ticked**, use this Boxing Command.*
*If **not ticked**, don't use this Boxing Command.*

OK button

OK *stores the values in the fields and removes the panel BUT no **recalc** is done.*

Apply button

Apply *stores the values and leaves the panel on the screen.*

*If the **Boxing Definition** is being used in an **Apply MTF** and **Auto recalc** is ticked in the MTF, then whenever the **Apply** button is clicked, then a **recalc** of the associated **Apply MTF** for the MTF is done.*

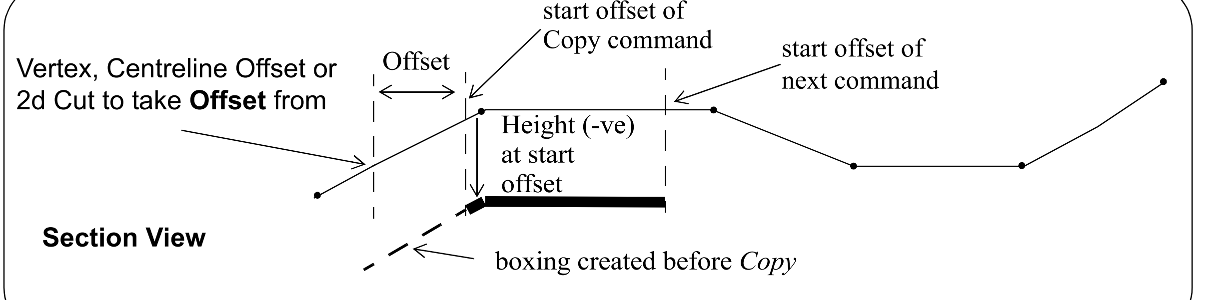
*If the **Boxing Definition** is being used in a **Boxing Many Function** then whenever the **Apply** button is clicked, a **recalc** of the **Boxing Many Function** is done.*

For the typed formats of the **Copy** and **Copy with height adjustment** commands created by this panel, see 21.10.2.1 Text Format - Boxing Copy and 21.10.2.2 Text Format - Boxing Copy with Height Adjustment.

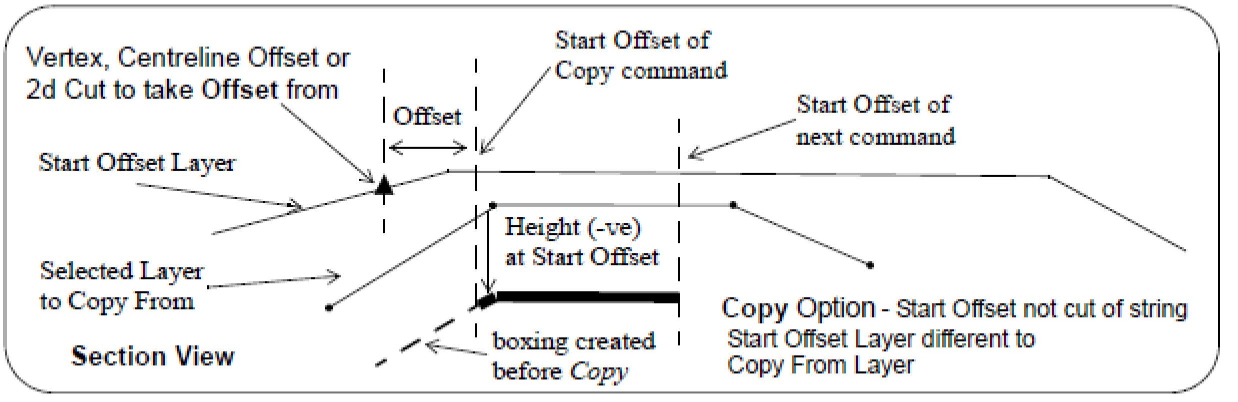
For more information on how Boxing works in ***12d Model***, please go to the section 21.6 What is Boxing?.

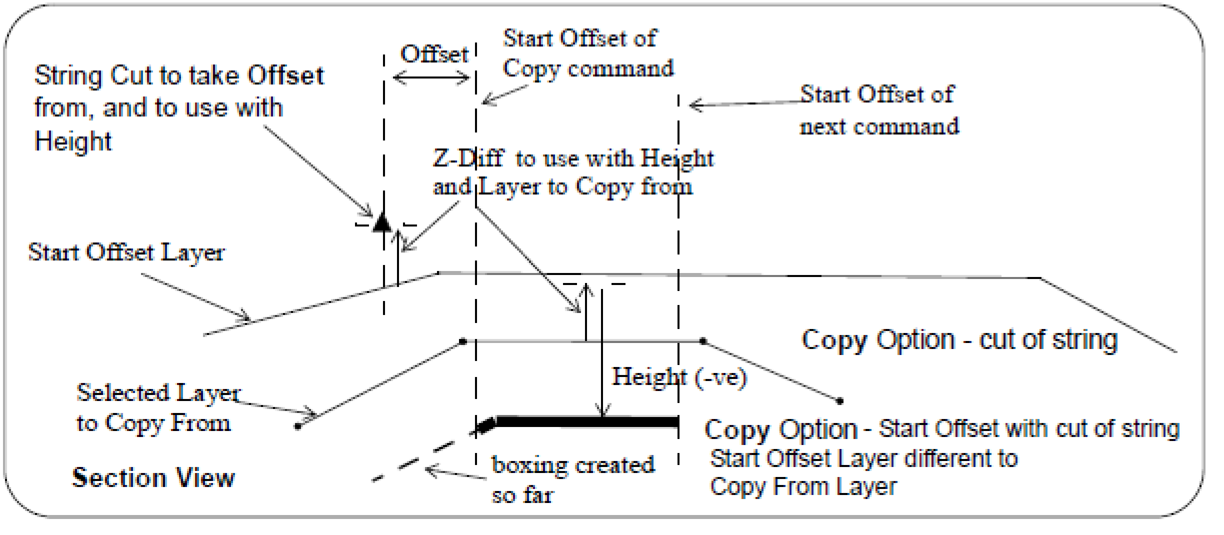
Go to the next section 21.7.2.2.2 Boxing Vertex or return to 21.7.2.2 Commands for Boxing Rules Grid.

Copy diagrams for more methods of defining *Offsets*:



Copy Option - Start Offset and Copy from Layer are the same Layer

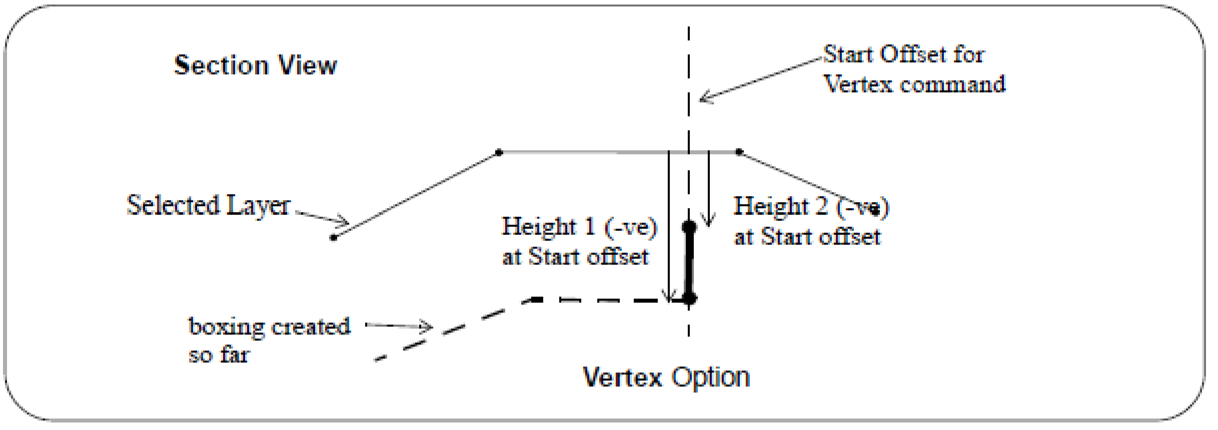

Vertex, Centreline Offset or 2d Cut to take Offset from
Start Offset of Copy command
Offset
Start Offset of next command
Start Offset Layer
Height (-ve) at Start Offset
Selected Layer to Copy From
Copy Option - Start Offset not cut of string
Start Offset Layer different to Copy From Layer
boxing created before Copy
Section View


String Cut to take Offset from, and to use with Height
Offset
Start Offset of Copy command
Start Offset of next command
Z-Diff to use with Height and Layer to Copy from
Start Offset Layer
Copy Option - cut of string
Height (-ve)
Selected Layer to Copy From
Copy Option - Start Offset with cut of string
Start Offset Layer different to Copy From Layer
boxing created so far
Section View

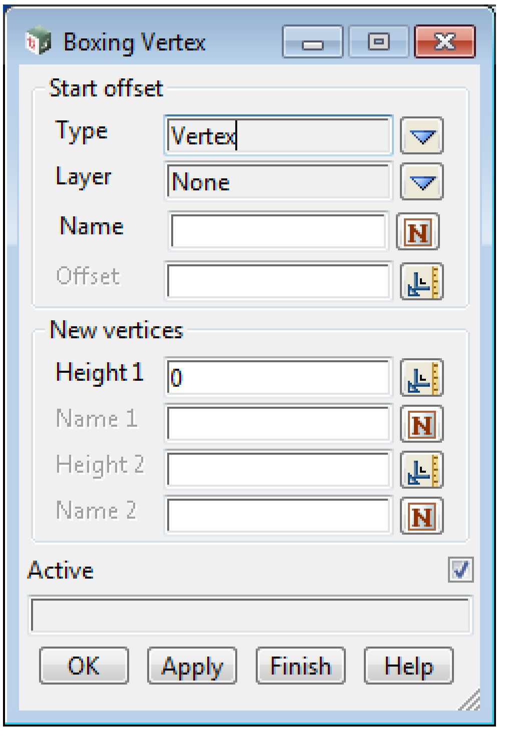
21.7.2.2.2 Boxing Vertex

The **Vertex** command inserts into the boxing section, one or two new vertices that are at the same *Start Offset*. So **Vertex** can be used to quickly create a vertical wall at a given *Start Offset*.

Note - to allow the boxing sections and strings to triangulate, ***12d Model*** increases the offset of the second vertex by ten thousandth of a unit.



Selecting **Vertex** brings up the **Boxing Vertex**



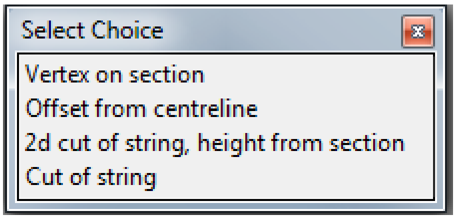
The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Start Offset

Layer to use with **Offset** *to defined the Start Offset where the one or two vertices are created.*

Type choice box



*For the documentation on the methods of defining the **Start Offset,** see* 21.7.2.3 Defining Relative Offset for Boxing Commands

New vertices

Height 1 real box

*If Start Offset Type is NOT: **Cut of string:***

Height 1 is added to the z-value of the **Layer** *at the Start Offset to give the z-values of the new boxing vertex. For example, if Height 1 is -0.3, the vertex is dropped by 0.3 from* **Layer.**

*If Start Offset Type is: **Cut of string:***

At the plan cut of the string and **Layer**, ***Z-Diff*** *(the difference of the z-values between the string and the* **Layer)** *is calculated and **Height 1** added to it. This combined z-difference is added to z-value of the* **Layer** *at the **Start Offset** to give the z-value of the new boxing vertex.*

Name 1 name box

if non blank, a vertex is created on the boxing section at Start Offset with this name and the appropriate height using Type and Height 1.

If blank, a vertex is created with a default name.

Height 2 real box

*If Start Offset Type is NOT: **Cut of string:***

Height 2 is added to the z-value of the **Layer** *at the Start Offset to give the z-values of the new boxing vertex. For example, if Height 2 is -0.5, the vertex is dropped by 0.5 from* **Layer.**

*If Start Offset Type is: **Cut of string:***

At the plan cut of the string and **Layer**, ***Z-Diff*** *(the difference of z-values between the string and the* **Layer***) is calculated and **Height 2** added to it. This combined z-difference is added to z-value of the* **Layer** *at the **Start Offset** to give the z-value of the new boxing vertex.*

Name 2 name box

if non blank, a vertex is created on the boxing section at Start Offset with this name and the appropriate height using Type and Height 2.

If blank, no vertex is created.

Note - to allow the boxing sections and strings to triangulate, if two vertices are created, ***12d Model*** actually increases the offset of the second vertex by ten thousandth of a unit.

Active tick box

*if **ticked**, use this Boxing Command.*
*If **not ticked**, don't use this Boxing Command.*

OK button

OK *stores the values in the fields and removes the panel BUT no* ***recalc*** *is done.*

Apply button

Apply *stores the values and leaves the panel on the screen.*

If the ***Boxing Definition*** *is being used in an* ***Apply MTF*** *and* ***Auto recalc*** *is ticked in the MTF, then whenever the* ***Apply*** *button is clicked, then a* ***recalc*** *of the associated* ***Apply MTF*** *for the MTF is done.*

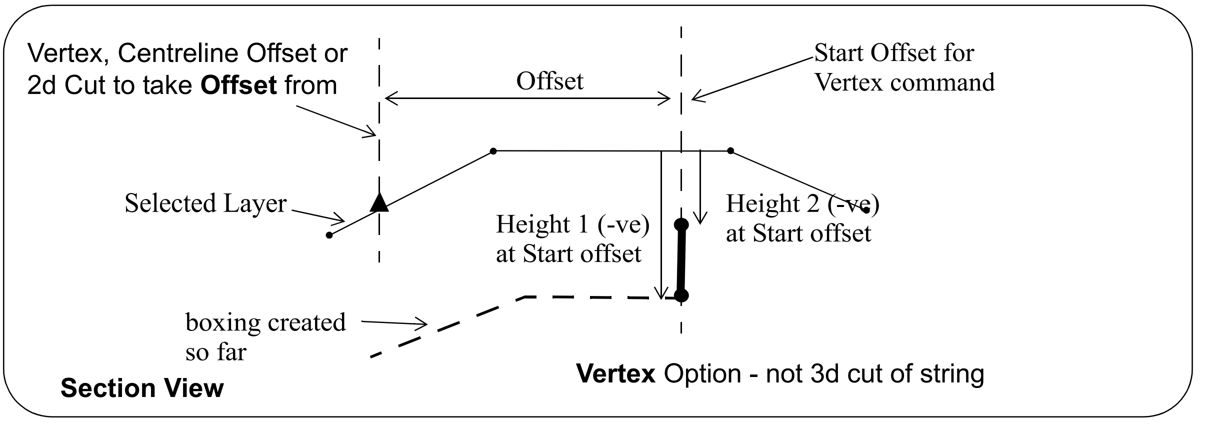
If the ***Boxing Definition*** *is being used in a* ***Boxing Many Function*** *then whenever the* ***Apply*** *button is clicked, a* ***recalc*** *of the* ***Boxing Many Function*** *is done.*

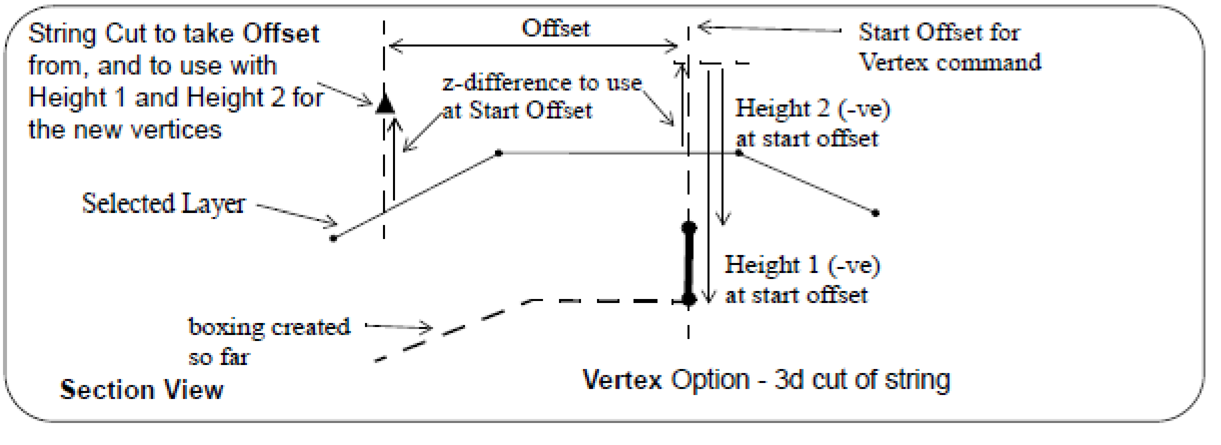
For the typed format of the **Vertex** command created by this panel, see 21.10.2.3 Text Format - Boxing Vertex.

For more information on how Boxing works in ***12d Model***, please go to the section 21.6 What is Boxing?.

Go to the next section 21.7.2.2.3 Boxing Vertex Advanced or return to 21.7.2.2 Commands for Boxing Rules Grid.

Vertex diagrams for more methods of defining *Offsets*:

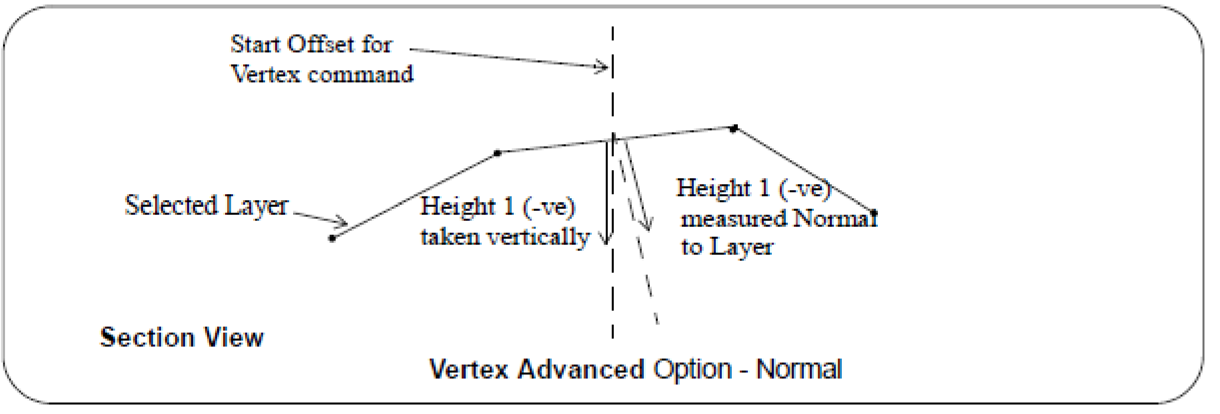




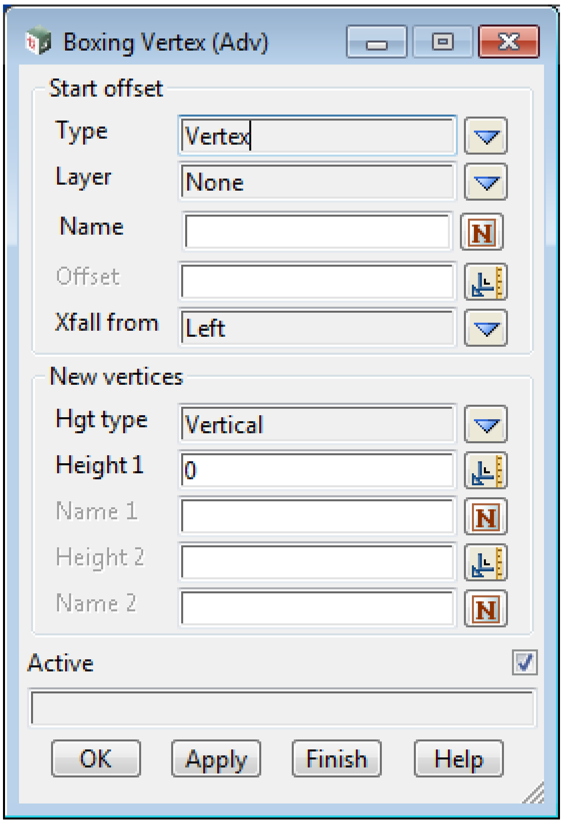
21.7.2.2.3 Boxing Vertex Advanced

The **Vertex Advanced** command allows a vertex to be placed where the distance is measured **NORMAL** or vertical or to a segment.

Vertex only allowed a vertical distance and not a normal distance.



Selecting **Vertex (Adv)** brings up the **Boxing Vertex (Adv)** panel



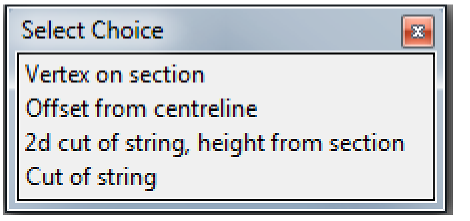
The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Start Offset

Layer to use with **Offset** *to defined the Start Offset where the one or two vertices are created.*

Type choice box



*For the documentation on the methods of defining the **Start Offset,** see* 21.7.2.3 Defining Relative Offset for Boxing Commands

Xfall from choice box Left, Right

When **Hgt type** *is Normal:*

if Left, the xfall to measure the distance NORMAL to is taken from the segment of the Start Offset layer that is slightly to the left of the Start Offset.

if Right, the xfall to measure the distance NORMAL to is taken from the segment of the Start Offset layer that is slightly to the right of the Start Offset.

When **Hgt type** *is Vertical,* **Xfall from** *is not used.*

New Vertices

Hgt type choice box Vertical, Normal

When **Hgt type** *is Normal, the distance is measured NORMAL to a segment.*

If **Xfall from** *is Left, the xfall to measure the distance NORMAL to is taken from the segment of the Start Offset layer that is slightly to the left of the Start Offset.*

If **Xfall from** *is Right, the xfall to measure the distance NORMAL to is taken from the segment of the Start Offset layer that is slightly to the right of the Start Offset.*

When **Hgt type** *is Vertical, the distance is measured in the vertical plane and* **Xfall from** *is not used.*

Name 1 name box

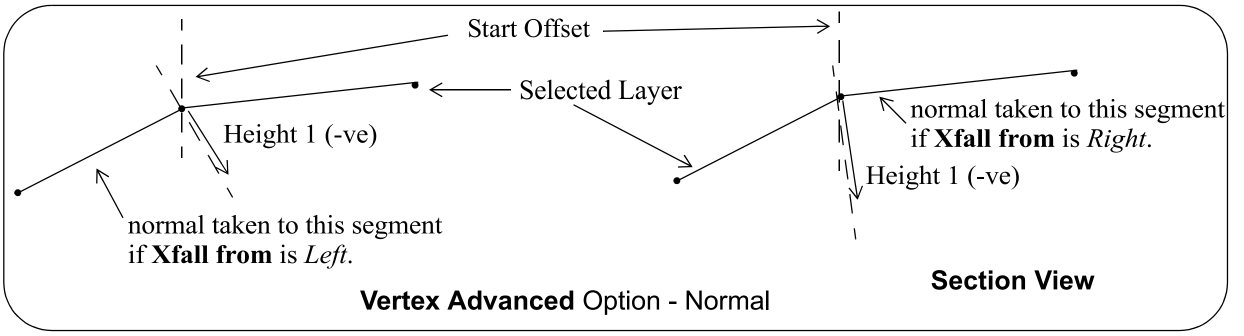
if non blank, a vertex is created on the boxing section with this name and the appropriate position using **Type***,* **Xfall from***,* **Hgt type** *and* **Height 1**.

If blank, a vertex is created with a default name.

Height 1 real box

*If Start Offset Type is NOT: **Cut of string:***

If **Hgt type** *is **Normal**, the distance* **Height 1** *is taken from the point on the* **Layer** *at the Start Offset, and going out NORMAL to the segment specified by Start Offset and* **Xfall from**.



Vertex Advanced Option - Normal

Section View

If **Hgt type** *is* ***Vertical****, the distance* **Height 1** *is taken from the point on the* **Layer** *at the Start Offset, and going vertically. For example, if Height 1 is -0.5, the vertex is dropped by 0.5 from* **Layer.**

If Start Offset Type is: ***Cut of string:***

??

Name 2 name box

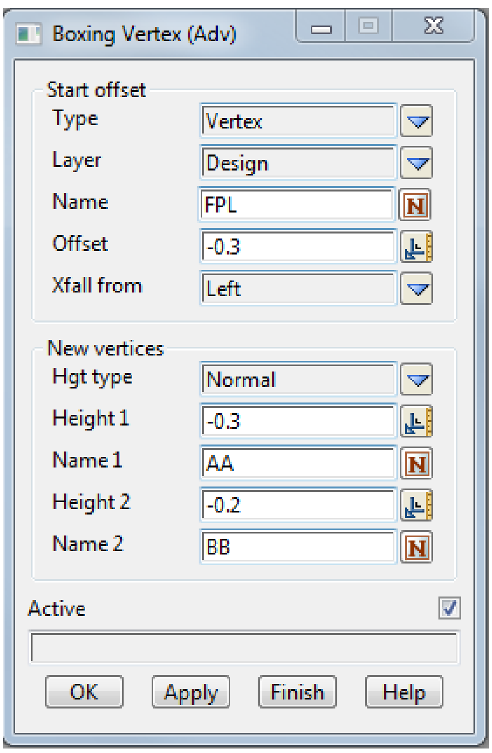
If blank, no vertex is created.

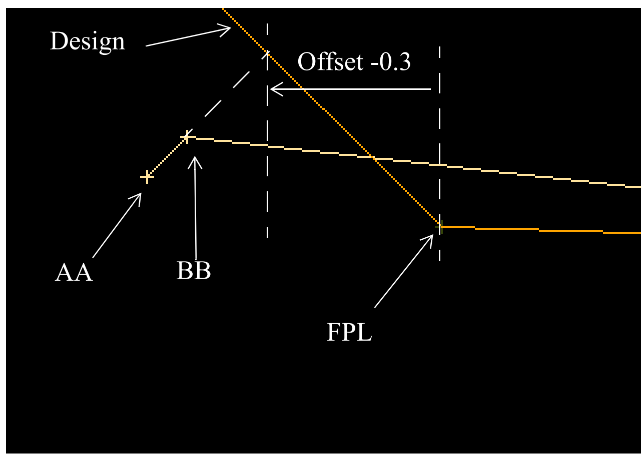
Height 2 real box

If Start Offset Type is NOT: ***Cut of string:***

If **Hgt type** *is* ***Normal****, the distance* **Height 2** *is taken from the point on the* **Layer** *at the Start Offset, and going out NORMAL to the segment specified by Start Offset and* **Xfall from**.

If **Hgt type** *is* ***Vertical****, the distance* **Height 2** *is taken from the point on the* **Layer** *at the Start Offset, and going vertically. For example, if Height 2 is -0.5, the vertex is dropped by 0.5 from* **Layer.**





Vertex (Adv) - Normal to Design Layer

Active tick box

if ***ticked****, use this Boxing Command.*
If ***not ticked****, don't use this Boxing Command.*

OK button

OK *stores the values in the fields and removes the panel BUT no* ***recalc*** *is done.*

Apply button

Apply *stores the values and leaves the panel on the screen.*

If the ***Boxing Definition*** *is being used in an* ***Apply MTF*** *and* ***Auto recalc*** *is ticked in the MTF, then whenever the* ***Apply*** *button is clicked, then a* ***recalc*** *of the associated* ***Apply MTF*** *for the MTF is done.*

*If the **Boxing Definition** is being used in a **Boxing Many Function** then whenever the **Apply** button is clicked, a **recalc** of the **Boxing Many Function** is done.*

Important Notes

1. the boxing vertices must be created from left to right so be careful that vertex Name 2 is no created to the left of vertex of Name 1. If it is then it will be move to be slightly to the right of vertex Name 1.
2. to allow the boxing sections and strings to triangulate, if two vertices are created that are on top of each other, ***12d Model*** actually increases the offset of the second vertex by ten thousandth of a unit.

For more information on how Boxing works in ***12d Model***, please go to the section 21.6 What is Boxing?.

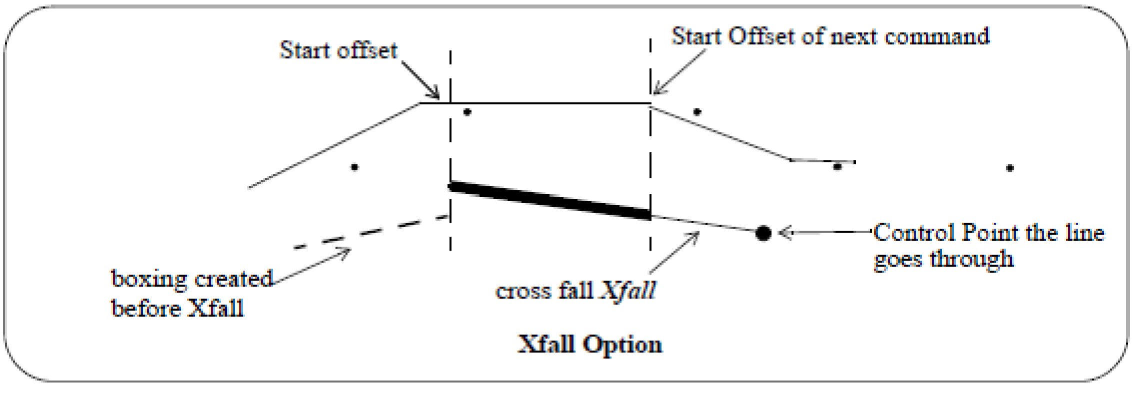
Go to the next section 21.7.2.2.4 Boxing Xfall - line through a point with a given crossfall or return to 21.7.2.2 Commands for Boxing Rules Grid.

21.7.2.2.4 Boxing Xfall - line through a point with a given crossfall

The Xfall command creates a link from the Start Offset to the Start Offset on the *next* boxing command, with a

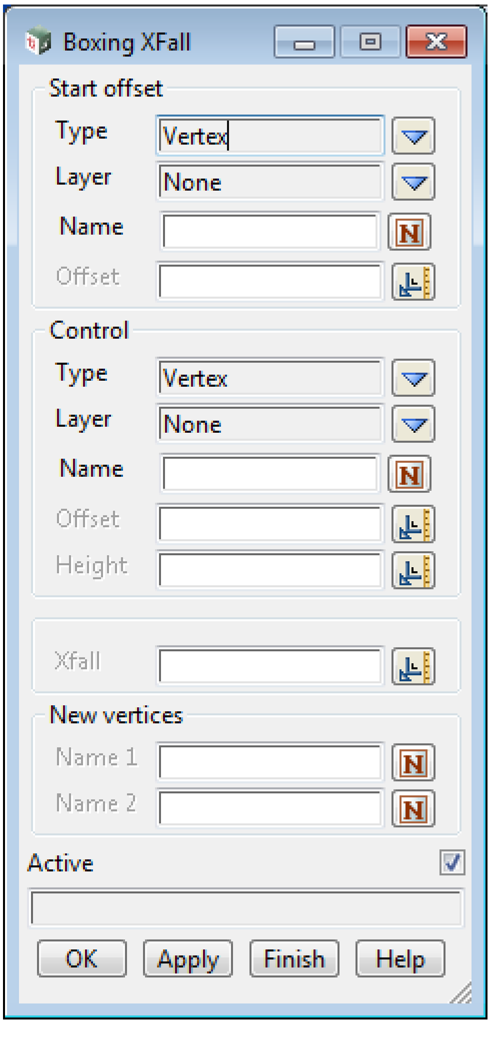
(a) user given cross fall (Xfall)

(b) start height given by projecting the line with a given xfall and going through a given control point, through to the Start Offset,



Xfall Option

Selecting **Xfall** brings up the **Boxing Xfall** panel



The fields and buttons used in this panel have the following functions.

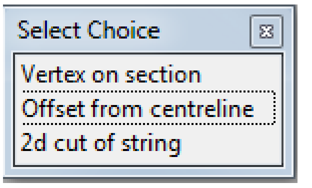
Field Description Type Defaults Pop-Up

Start Offset

Start Offset defines the offset where the new line starts from.

*For the **Xfall** command, the 3d cut of string is not needed. Only **2d cut of string** is required.*

Type choice box

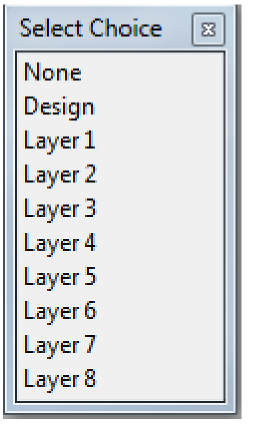


*For the documentation on the methods of defining the **Start Offset**, see* 21.7.2.3 Defining Relative Offset for Boxing Commands

Control

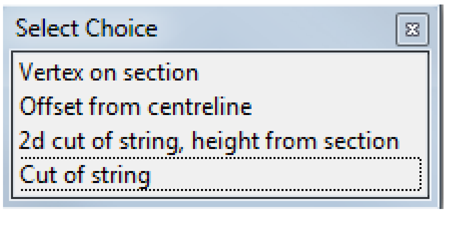
the definition of the Control Point that the line goes through.

Layer choice box



Layer to use with Control **Offset** *to define the offset of the Control Point.*

Type choice box



*If **Vertex on section**:*

Name name box

name of the vertex on the Control **Layer** *to use with* **Offset** *to defined the Control Offset. And used with* **Height** *to define the height of the Control Point*

Offset real box 0

this value is added to the offset of the vertex given in the **Name** *field to define the Control Offset.* **Offset** *can be positive or negative as long as the final Control Offset is still **on** the Control* **Layer**.

Height real box 0

Height is added to the z-value of the Control **Layer** *at the Control Offset to give the height of the Control Point.*

If ***Offset from centreline:***

Offset real box 0

an actual offset from centreline (actually just the chainage of the section) which is used as the Control Offset. **Offset** *can be positive or negative as long as the Control Point is still on the Control* **Layer***.*

Height real box 0

Height is added to the z-value of the Control **Layer** *at the Control Offset to give the height of the Control Point.*

If ***2d cut of string, height from section:***

String string select

the selected string needs to cut the Control **Layer** *in plan wherever the boxing is being applied.*

The Control Offset is the offset of the plan cut of the selected string with the Control **Layer***, plus the* **Offset** *value.* **Offset** *can be positive or negative as long as the final Control Offset is still on the Control* **Layer***.*

Offset real box 0

offset to add to the offset of the plan cut of the Control **Layer** *with the selected string to give the Start Offset.* **Offset** *can be positive or negative but the final Control Offset must still be on the Control* **Layer***.*

Height real box 0

Height is added to the z-value of the Control **Layer** *at the Control Offset to give the height of the Control Point.*

If ***Cut of string:***

String string select

the selected string needs to cut the Control **Layer** *in plan wherever the boxing is being applied.*

The Control Offset is the offset of the plan cut of the selected string with the Control **Layer***, plus the* **Offset** *value.* **Offset** *can be positive or negative but the final position must still be on the Control* **Layer***.*

Offset real box 0

offset to add to the offset of the plan cut of the section with the selected string to give the Control Offset. **Offset** *can be positive or negative but the final Control Offset must still be on the section.*

Height real box 0

At the plan cut of the string and the Control **Layer***, the difference of the z-values between the string and the Control* **Layer** *is calculated and* **Height** *added to it. This combined value is added to the z-values on the Control* **Layer***, to give the z-value of the Control Point.*

Xfall input 0

xfall of the line going through the Control Point.

New vertices

Name 1 name box

if non blank, the vertex created at the Start Offset is given this name

If blank, the vertex created at the Start Offset is given a default name.

Name 2 name box

if non blank, the vertex created at the End Offset (the Start Offset of the next command) is given this name

If blank, the vertex created at the End Offset is given a default name.

Active tick box

*if **ticked**, use this Boxing Command.*
*If **not ticked**, don't use this Boxing Command.*

OK button

OK *stores the values in the fields and removes the panel BUT no **recalc** is done.*

Apply button

Apply *stores the values and leaves the panel on the screen.*

*If the **Boxing Definition** is being used in an **Apply MTF** and **Auto recalc** is ticked in the MTF, then whenever the **Apply** button is clicked, then a **recalc** of the associated **Apply MTF** for the MTF is done.*

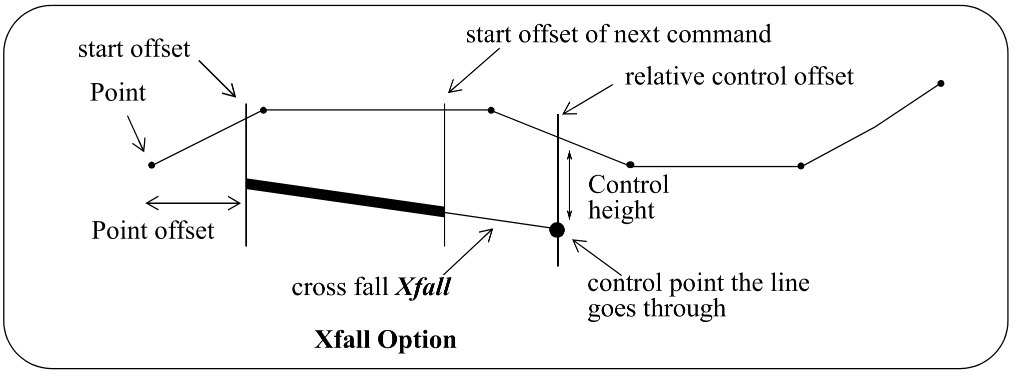
*If the **Boxing Definition** is being used in a **Boxing Many Function** then whenever the **Apply** button is clicked, a **recalc** of the **Boxing Many Function** is done.*

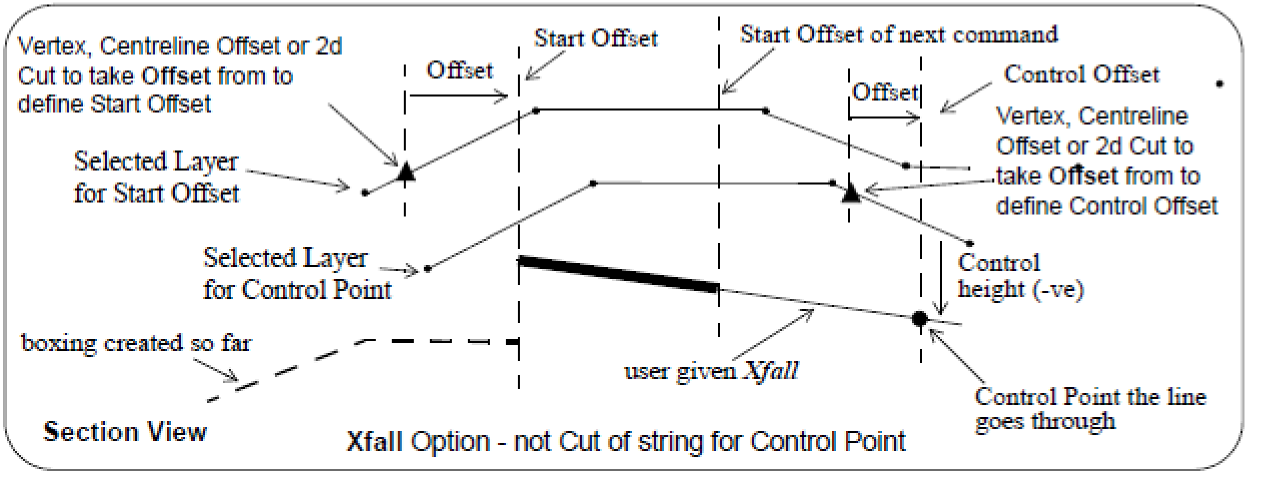
For the typed formats of the **Xfall** command created by this panel, see 21.10.2.5 Text Format - Boxing Xfall.

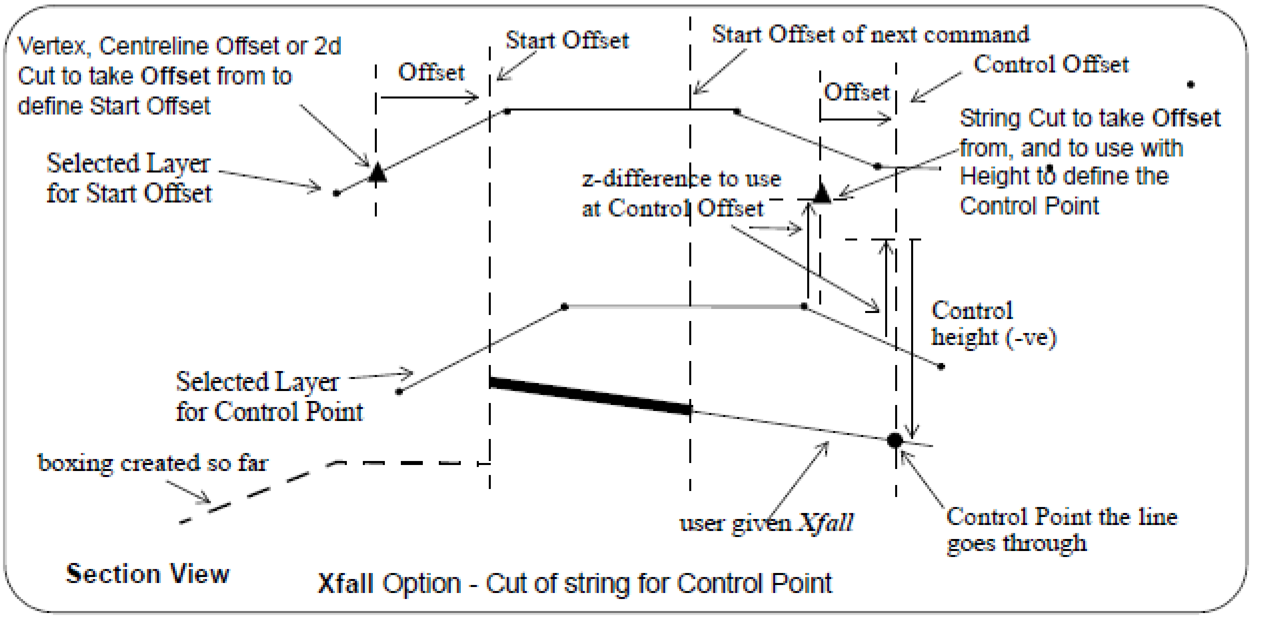
For more information on how Boxing works in ***12d Model***, please go to the section 21.6 What is Boxing?.

Go to the next section 21.7.2.2.5 Boxing Xfall 1 - line through a point with a crossfall taken from a point or return to 21.7.2.2 Commands for Boxing Rules Grid.

Xfall diagrams for more methods of defining *Offsets*:




Vertex, Centreline Offset or 2d Cut to take Offset from to define Start Offset
Offset
Start Offset
Start Offset of next command
Offset
Control Offset
Vertex, Centreline Offset or 2d Cut to take Offset from to define Control Offset
Selected Layer for Start Offset
Selected Layer for Control Point
Control height (-ve)
boxing created so far
user given Xfall
Control Point the line goes through
Section View
Xfall Option - not Cut of string for Control Point

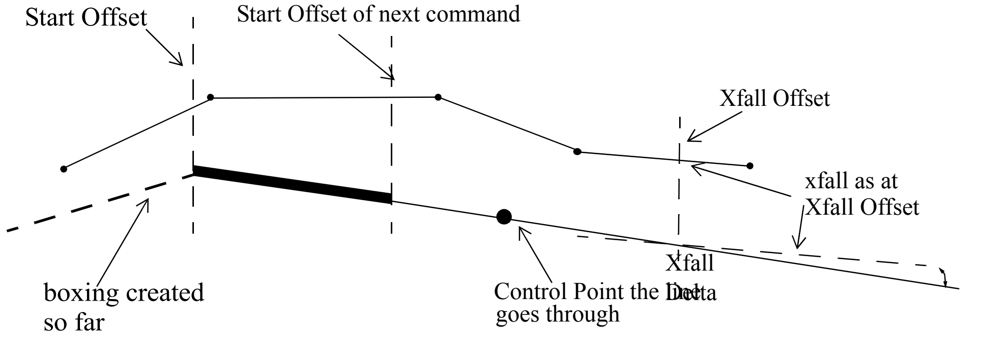

Vertex, Centreline Offset or 2d Cut to take Offset from to define Start Offset
Offset
Start Offset
Start Offset of next command
Offset
Control Offset
String Cut to take Offset from, and to use with Height to define the Control Point
Selected Layer for Start Offset
z-difference to use at Control Offset
Control height (-ve)
Selected Layer for Control Point
boxing created so far
user given Xfall
Control Point the line goes through
Section View
Xfall Option - Cut of string for Control Point

21.7.2.2.5 Boxing Xfall 1 - line through a point with a crossfall taken from a point

The **Xfall 1** command creates a link from the Start Offset, to the Start Offset on the *next* boxing command, with

(a) a Xfall which is the xfall at a given offset plus an additional xfall **Delta**

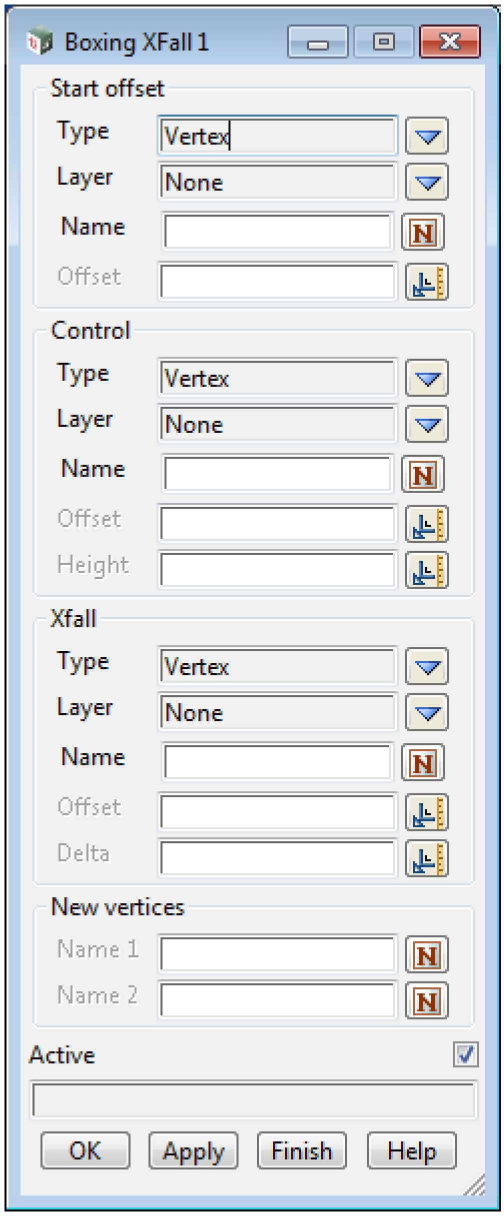
(b) a start height given by projecting to the Start Offset, the line with the specified Xfall and going through a given Control Point.



Section View **Xfall 1** Option

Note - the difference between the boxing commands ***Xfall*** and ***Xfall 1*** is that in the command **Xfall**, the user supplies the value of the xfall whereas in the command **Xfall 1**, the xfall is taken from a link on the Xfall **Layer**.

Selecting **Xfall 1** brings up the **Boxing Xfall 1** panel



The fields and buttons used in this panel have the following functions.

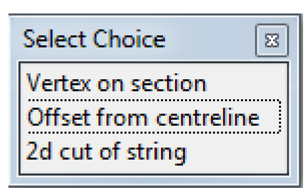
Field Description Type Defaults Pop-Up

Start offset

Start Offset defines the offset where the new line starts from.

*For the **Xfall 1** command, the 3d cut of string is not needed. Only **2d cut of string** is required.*

Type choice box

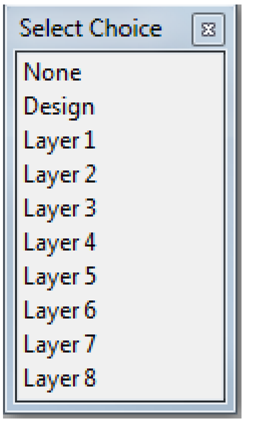


*For the documentation on the methods of defining the **Start Offset,** see* 21.7.2.3 Defining Relative Offset for Boxing Commands

Control

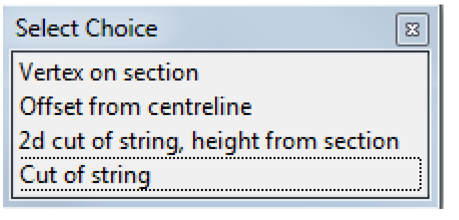
the definition of the Control Point for the line to go through.

Layer choice box



Layer to use with Control **Offset** *to define the offset of the Control Point.*

Type choice box



*If **Vertex on section**:*

Name name box

name of the vertex on the Control **Layer** *to use with* **Offset** *to defined the Control Offset. And used with* **Height** *to define the height of the Control Point*

Offset real box 0

this value is added to the offset of the vertex given in the **Name** *field to define the Control Offset.* **Offset** *can be positive or negative as long as the final Control Offset is still* ***on*** *the Control* **Layer**.

Height real box 0

Height is added to the z-value of the Control **Layer** *at the Control Offset to give the height of the Control Point.*

*If **Offset from centreline:***

Offset real box 0

an actual offset from centreline (actually just the chainage of the section) which is used as the Control Offset. **Offset** *can be positive or negative as long as the Control Point is still on the Control* **Layer**.

Height real box 0

Height is added to the z-value of the Control **Layer** *at the Control Offset to give the height of the Control Point.*

*If **2d cut of string, height from section:***

String string select

the selected string needs to cut the Control **Layer** *in plan wherever the boxing is being applied.*

The Control Offset is the offset of the plan cut of the selected string with the Control **Layer***, plus the* **Offset** *value.* **Offset** *can be positive or negative as long as the final Control Offset is still on the Control* **Layer***.*

Offset real box 0

offset to add to the offset of the plan cut of the Control **Layer** *with the selected string to give the Start Offset.* **Offset** *can be positive or negative but the final Control Offset must still be on the Control* **Layer***.*

Height real box 0

Height is added to the z-value of the Control **Layer** *at the Control Offset to give the height of the Control Point.*

If ***Cut of string:***

String string select

the selected string needs to cut the Control **Layer** *in plan wherever the boxing is being applied.*

The Control Offset is the offset of the plan cut of the selected string with the Control **Layer***, plus the* **Offset** *value.* **Offset** *can be positive or negative but the final position must still be on the Control* **Layer***.*

Offset real box 0

offset to add to the offset of the plan cut of the section with the selected string to give the Control Offset. **Offset** *can be positive or negative but the final Control Offset must still be on the section.*

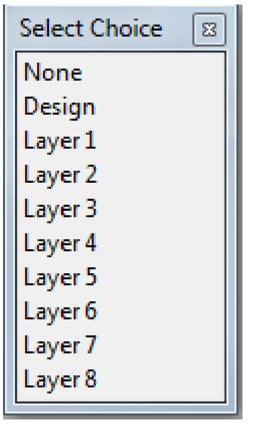
Height real box 0

At the plan cut of the string and the Control **Layer***, the difference of the z-values between the string and the Control* **Layer** *is calculated and* **Height** *added to it. This combined value is added to the z-values on the Control* **Layer***, to give the z-value of the Control Point.*

Xfall

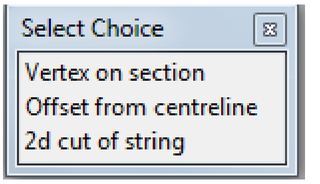
the xfall of the line going through the Control Point is:
the xfall of the Xfall **Layer** *at Xfall Offset, plus the value* **Delta***.*

Layer choice box



Layer to use with **Offset** *to defined the Xfall*

Type choice box



*If **Vertex on section**:*

Name name box

name of the vertex on Xfall **Layer** *to use with* **Offset** *to define the Xfall Offset.*

Offset real box

this value is added to the offset of the vertex given in the **Name** *field to define the Xfall Offset.* **Offset** *can be positive or negative as long as the final Xfall Offset is still **on** the Xfall* **Layer**.

Delta real box

the Xfall of the line is the xfall of the Xfall **Layer** *at Xfall Offset plus* **Delta**

*If **Offset from centreline:***

Offset real box

an actual offset from centreline (actually just the chainage of the section) which is used as the Xfall Offset. **Offset** *can be positive or negative as long as it is still on the Xfall* **Layer**.

Delta real box

the Xfall of the line is the xfall of the segment of Xfall **Layer** *at Xfall Offset plus* **Delta**

*If **2d cut of string:***

String string select

the selected string needs to cut the Xfall **Layer** *in plan wherever the boxing is being applied.*

The Xfall Offset is the offset of the plan cut of the selected string with the Xfall **Layer**, *plus the* **Offset** *value.* **Offset** *can be positive or negative but the final position Xfall Offset must still be on the Xfall* **Layer**.

Offset real box

offset to add to the offset of the plan cut of the Xfall **Layer** *with the selected string to give the Xfall Offset.* **Offset** *can be positive or negative but the final Xfall Offset must still be on the Xfall* **Layer**.

Delta real box

the Xfall of the line is the xfall of the segment of Xfall **Layer** *at Xfall Offset plus* **Delta.**

New vertices

Name 1 name box

if non blank, the vertex created at the Start Offset is given this name

If blank, the vertex created at the Start Offset is given a default name.

Name 2 name box

if non blank, the vertex created at the End Offset (the Start Offset of the next command) is given this name

If blank, the vertex created at the End Offset is given a default name.

Active tick box

*if **ticked**, use this Boxing Command.*
*If **not ticked**, don't use this Boxing Command.*

OK button

OK *stores the values in the fields and removes the panel BUT no **recalc** is done.*

Apply button

Apply *stores the values and leaves the panel on the screen.*

*If the **Boxing Definition** is being used in an **Apply MTF** and **Auto recalc** is ticked in the MTF, then whenever the **Apply** button is clicked, then a **recalc** of the associated **Apply MTF** for the MTF is done.*

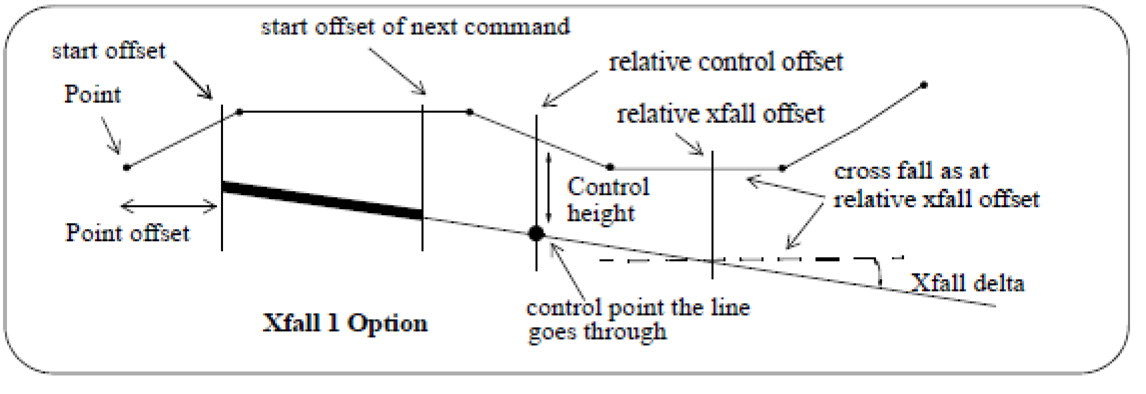
*If the **Boxing Definition** is being used in a **Boxing Many Function** then whenever the **Apply** button is clicked, a **recalc** of the **Boxing Many Function** is done.*

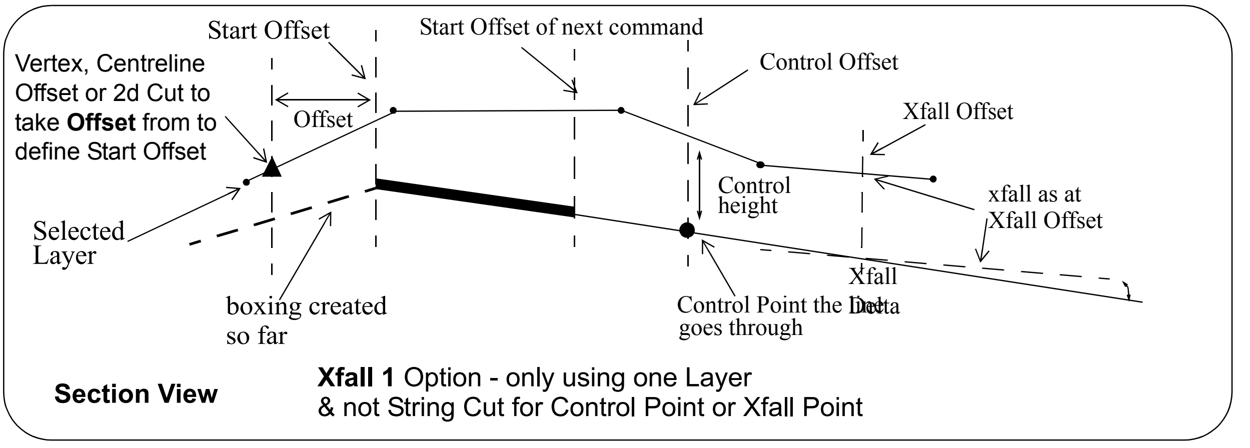
For the typed format of the **xfall1** command created by this panel, see 21.10.2.6 Text Format - Boxing Xfall1.

For more information on how Boxing works in ***12d Model***, please go to the section 21.6 What is Boxing?.

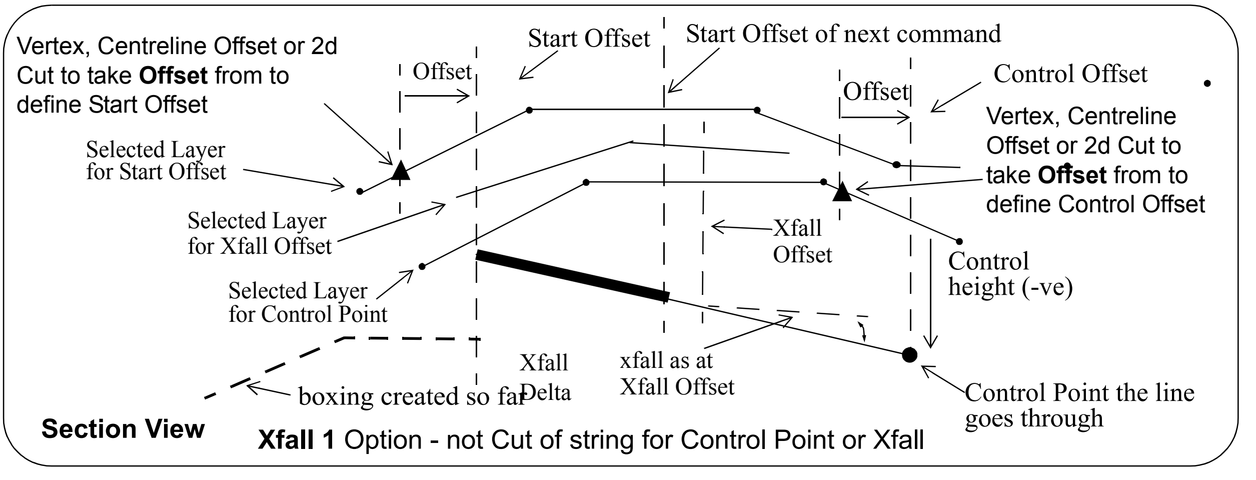
Go to the next section 21.7.2.2.6 Boxing Line - line through two points or return to 21.7.2.2 Commands for Boxing Rules Grid.

Xfall 1 diagrams for more methods of defining *Offsets*:

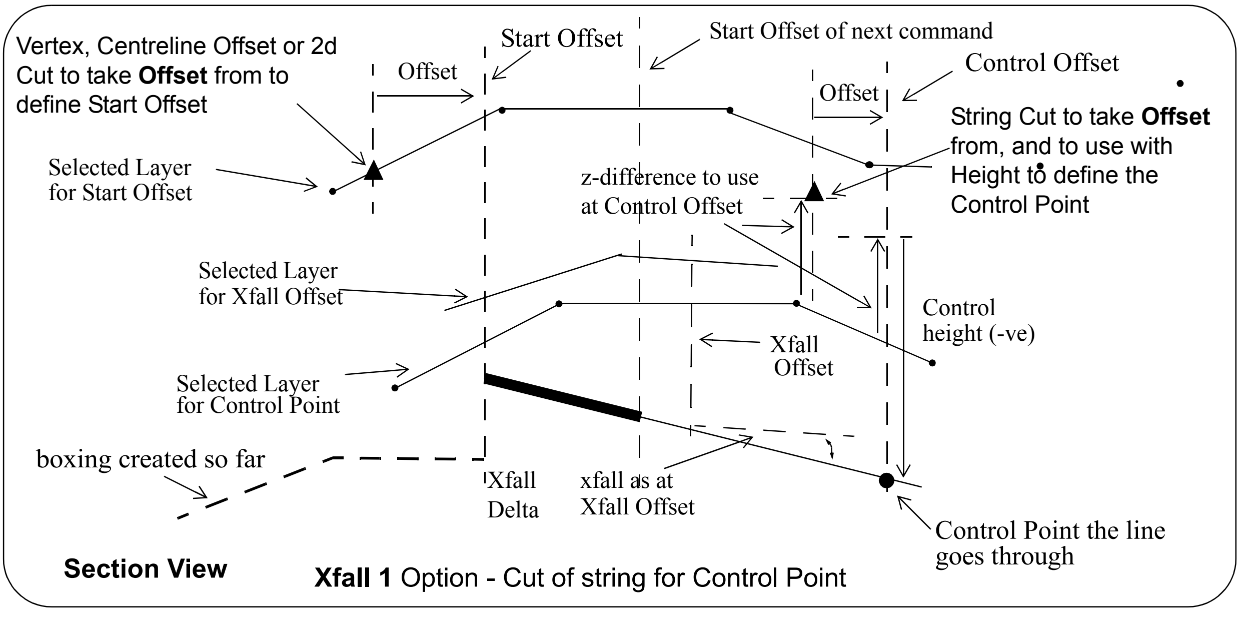




Start Offset
Start Offset of next command
Vertex, Centreline Offset or 2d Cut to take **Offset** from to define Start Offset
Control Offset
Offset
Xfall Offset
Control height
xfall as at Xfall Offset
Selected Layer
Xfall Delta
boxing created so far
Control Point the line goes through
Section View
Xfall 1 Option - only using one Layer & not String Cut for Control Point or Xfall Point



Vertex, Centreline Offset or 2d Cut to take **Offset** from to define Start Offset
Start Offset
Start Offset of next command
Offset
Offset
Control Offset
Vertex, Centreline Offset or 2d Cut to take **Offset** from to define Control Offset
Selected Layer for Start Offset
Selected Layer for Xfall Offset
Xfall Offset
Selected Layer for Control Point
Control height (-ve)
Xfall Delta
xfall as at Xfall Offset
boxing created so far
Control Point the line goes through
Section View
Xfall 1 Option - not Cut of string for Control Point or Xfall



Vertex, Centreline Offset or 2d Cut to take **Offset** from to define Start Offset
Start Offset
Start Offset of next command
Offset
Offset
Control Offset
String Cut to take **Offset** from, and to use with Height to define the Control Point
Selected Layer for Start Offset
z-difference to use at Control Offset
Selected Layer for Xfall Offset
Control height (-ve)
Xfall Offset
Selected Layer for Control Point
boxing created so far
Xfall Delta
xfall as at Xfall Offset
Control Point the line goes through
Section View
Xfall 1 Option - Cut of string for Control Point

21.7.2.2.6 Boxing Line - line through two points

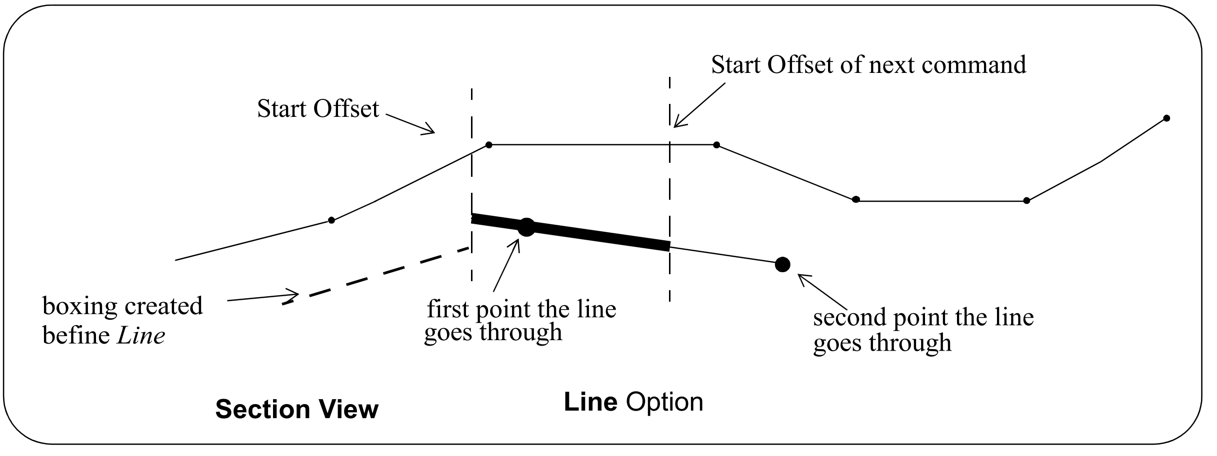
The Line command creates a line from the Start Offset to the Start Offset of the *next* boxing command, with a

(a) slope given by two points

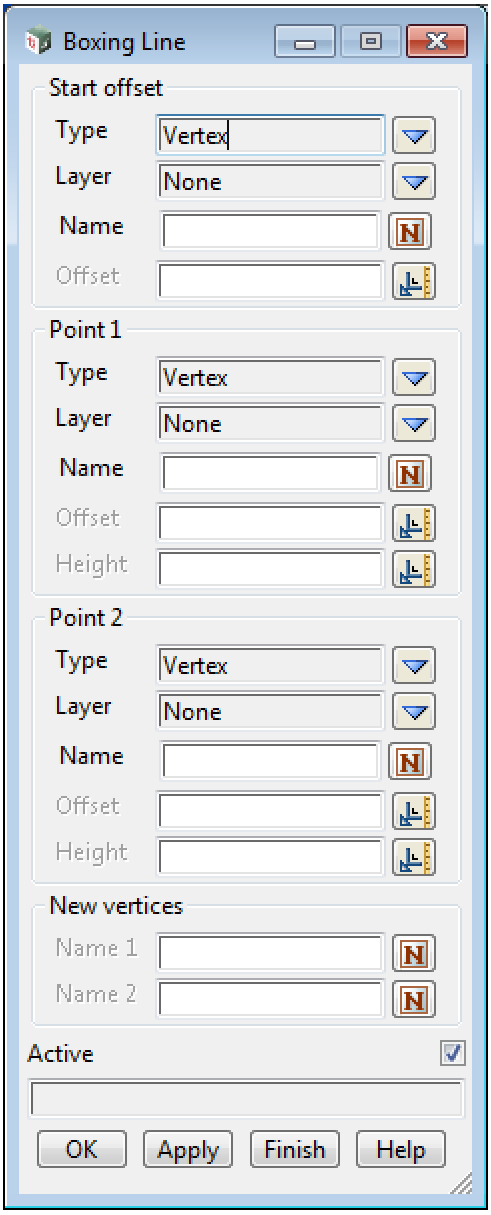
(b) start height given by projecting to the Start Offset, the line through the two points.

The first point defining the line has the offset **Point 1 Offset** which may be given in terms of a Vertex, or Cut with a string, and a height given by the height of a selected Layer, or at the Cut string, plus a given **Height**.

The second point defining the line has the offset **Point 2 Offset** which may be given in terms of a Vertex, or Cut with a string, and a height given by the height of a selected Layer, or at the Cut string, plus a given **Height**.



Selecting **Line** brings up the **Boxing Line** panel



The fields and buttons used in this panel have the following functions.

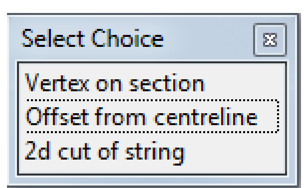
Field Description Type Defaults Pop-Up

Start offset

Start Offset defines the offset where the new line starts from.

*For the **Line** command, the 3d cut of string is not needed. Only **2d cut of string** is required.*

Type choice box

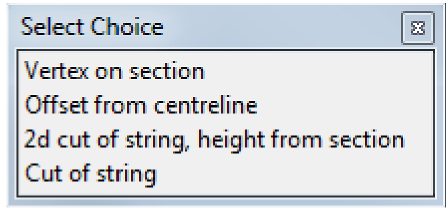


*For the documentation on the methods of defining the **Start Offset**, see* 21.7.2.3 Defining Relative Offset for Boxing Commands

Point 1

the definition of the first point that the line goes through.

Type choice box



For the documentation on the methods of defining the Point 1 Offset, see 21.7.2.3 Defining Relative Offset for Boxing Commands

*If **Vertex on section**:*

Height real box 0

Height is added to the z-value on the **Layer** *at the **Point 1 Offset** to give the height of Point 1.*

*If **Offset from centreline:***

Height real box 0

Height is added to the z-value on the **Layer** *at the **Point 1 Offset** to give the height of Point 1.*

*If **2d cut of string, height from section:***

Height real box 0

Height is added to the z-value on the **Layer** *at the **Point 1 Offset** to give the height of Point 1.*

*If **Cut of string:***

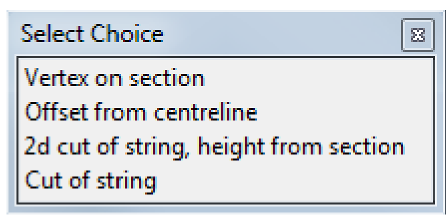
Height real box 0

Height is added to the z-value of the cut string to give the z-value of Point 1.

Point 2

the definition of the second point that the line goes through.

Type choice box



*For the documentation on the methods of defining the **Point 2 Offset,** see* 21.7.2.3 Defining Relative Offset for Boxing Commands

*If **Vertex on section**:*

Height real box 0

Height is added to the z-value on the **Layer** *at the Point 2 Offset to give the height of Point 2.*

*If **Offset from centreline:***

Height real box 0

Height is added to the z-value of the **Layer** *at the Point 2 Offset to give the height of Point 2.*

*If **2d cut of string, height from section:***

Height real box 0

Height is added to the z-value on the **Layer** *at the Point 2 Offset to give the height of Point 2.*

*If **Cut of string:***

Height real box 0

Height is added to the z-value of the cut string to give the z-value of Point 2.

New Vertices

Name 1 name box

if non blank, the vertex created at the Start Offset is given this name

If blank, the vertex created at the Start Offset is given a default name.

Name 2 name box

if non blank, the vertex created at the End Offset (the Start Offset of the next command) is given this name

If blank, the vertex created at the End Offset is given a default name.

Active tick box

*if **ticked**, use this Boxing Command.*
*If **not ticked**, don't use this Boxing Command.*

OK button

OK *stores the values in the fields and removes the panel BUT no **recalc** is done.*

Apply button

Apply *stores the values and leaves the panel on the screen.*

*If the **Boxing Definition** is being used in an **Apply MTF** and **Auto recalc** is ticked in the MTF, then whenever the **Apply** button is clicked, then a **recalc** of the associated **Apply MTF** for the MTF is done.*

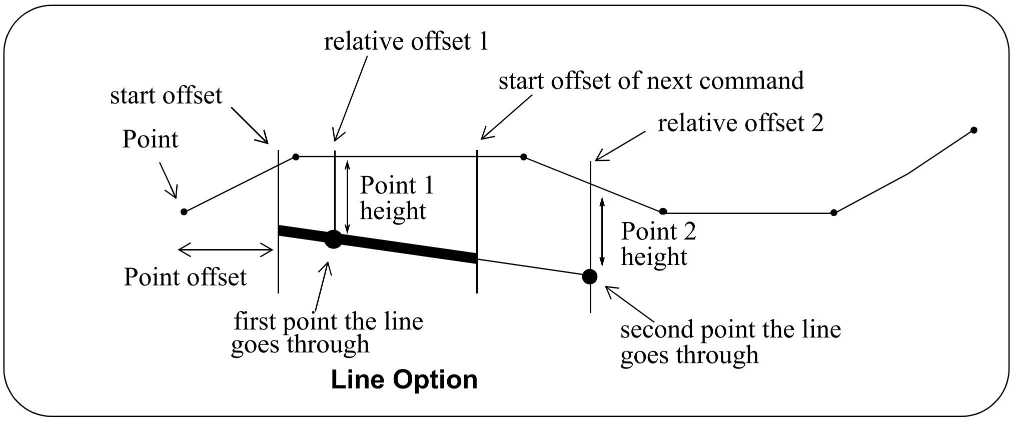
*If the **Boxing Definition** is being used in a **Boxing Many Function** then whenever the **Apply** button is clicked, a **recalc** of the **Boxing Many Function** is done.*

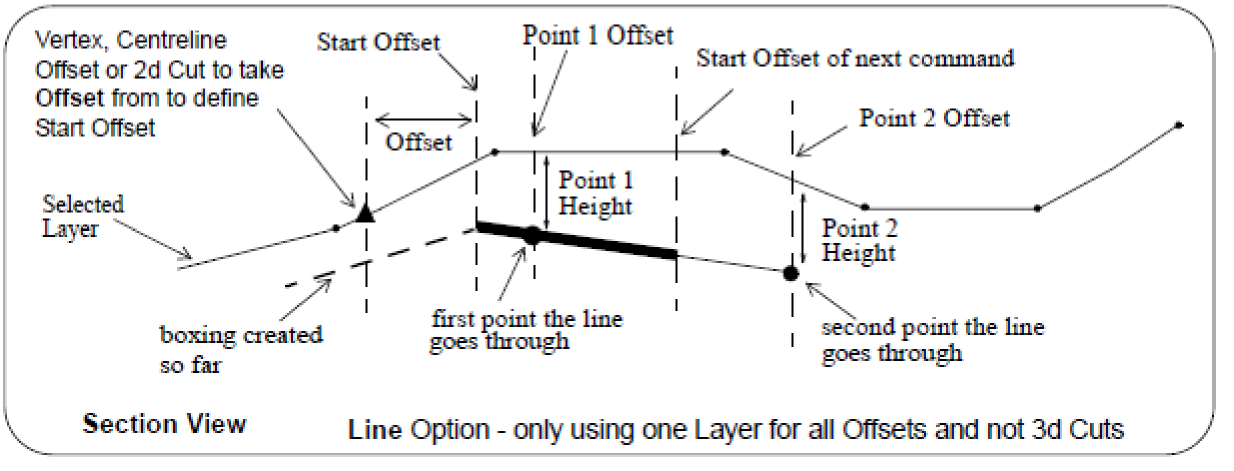
For the typed formats of the **Line** command created by this panel, see 21.10.2.4 Text Format - Boxing Line.

For more information on how Boxing works in ***12d Model***, please go to the section 21.6 What is Boxing?.

Go to the next section 21.7.2.2.7 Boxing Intersect - Intersection of Two Lines or return to 21.7.2.2 Commands for Boxing Rules Grid.

Line diagrams for more methods of defining *Offsets*:


relative offset 1
start offset of next command
start offset
relative offset 2
Point
Point 1
height
Point 2
height
Point offset
first point the line
goes through
second point the line
goes through
Line Option


Vertex, Centreline
Offset or 2d Cut to take
Offset from to define
Start Offset
Start Offset
Point 1 Offset
Start Offset of next command
Point 2 Offset
Offset
Point 1
Height
Selected
Layer
Point 2
Height
boxing created
so far
first point the line
goes through
second point the line
goes through
Section View
Line Option - only using one Layer for all Offsets and not 3d Cuts

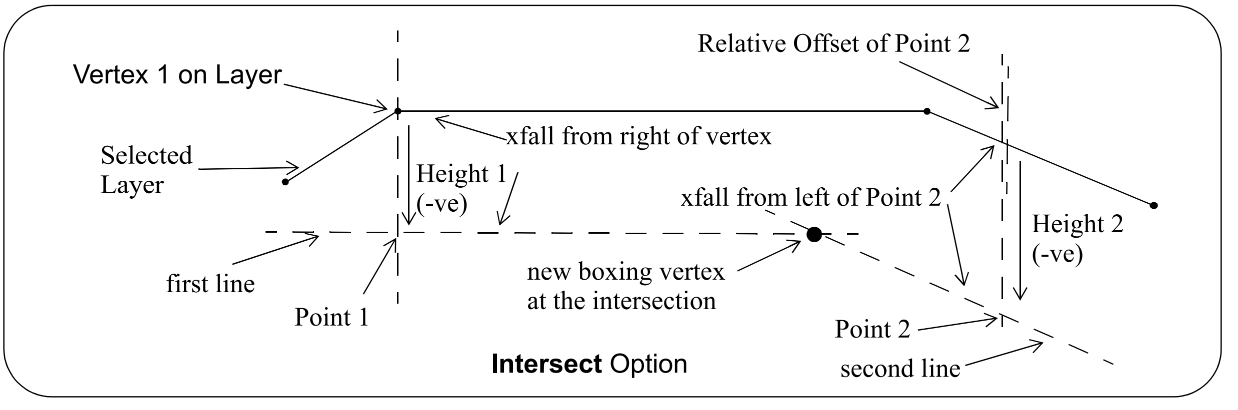
21.7.2.2.7 Boxing Intersect - Intersection of Two Lines

The **Intersect** command creates a vertex on the boxing section that is the **intersection** of two lines.

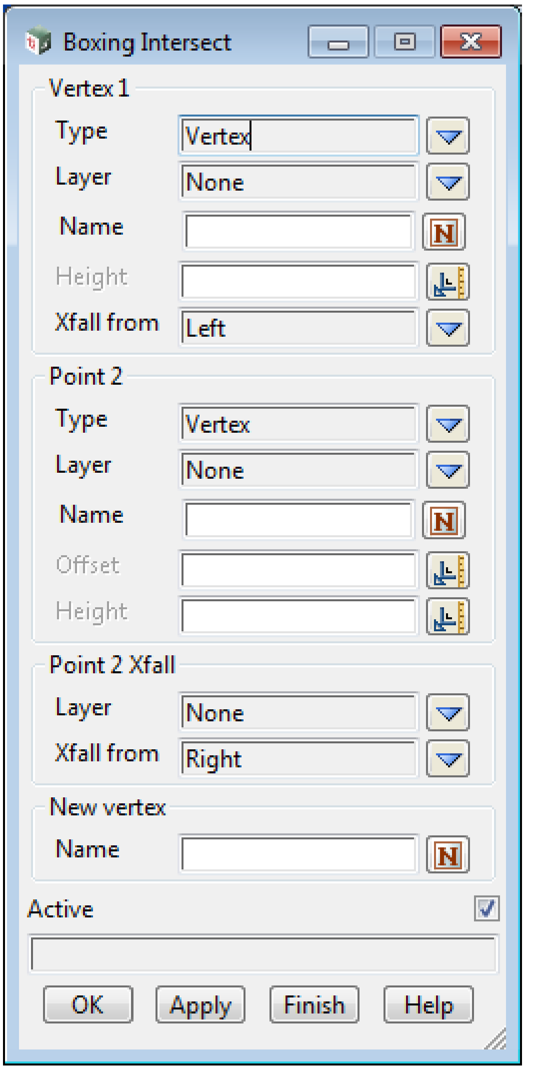
The first line goes through Point 1 where the offset for Point 1 is the offset of a vertex from a Layer, and the height at Point 1 is the height on the vertex plus of a given **Height**. The **xfall** of the line is taken from the Layer segment to either the Left or Right of the selected vertex.

The second line goes through Point 2 where the offset for Point 2 is a relative offset to a vertex or a selected string, and the height at Point 2 is either the height on the Layer section plus a given **Height**, or the height at the cut string, with the addition of a given **Height**. The **Xfall** of the line is taken from the Layer segment to either the Left or Right of the relative offset of Point 2.

A new boxing vertex is created at the intersection of the two lines.



Selecting **Intersect** brings up the **Boxing Intersect** panel



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

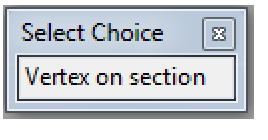
Vertex 1

the definition of the point that the first line goes through. The point must be vertically over a vertex from a Layer

Layer choice box Design, Layer 1 ... Layer 8

Layer to select the vertex from to define the offset of the point the first line goes through.

Type choice box



If ***Vertex on section****:*

Name name box

name of the vertex on **Layer** *to use with* **Height** *to defined the offset and height for point that the first line goes through*

Height real box

Height is added to the z-value of the selected vertex to give the point the first line goes through.

Xfall from choice box Left, Right

if Left, the xfall for the line is taken from the segment that is a little left of the selected vertex.

If Right, the xfall for the line is taken from the segment that is a little right of the selected vertex

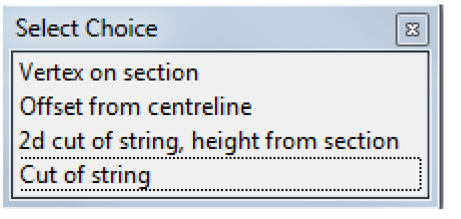
Point 2

the definition of the point that the second line goes through

Layer choice box Design, Layer 1 ... Layer 8

Layer to select the vertex from to define the offset of the point the second line goes through.

Type choice box



If ***Vertex on section****:*

Name name box

name of the vertex on the cross section to use with **Offset** *to defined the Point 2 Offset. And used with* **Height** *to define the height of the Point 2.*

Offset real box 0

this value is added to the offset of the vertex given in the **Name** *field to define the Point 2 Offset.* **Offset** *can be positive or negative as long as the final Point 2 Offset is still* ***on*** *the section.*

Height real box 0

Height is added to the z-value of the section at the Point 2 Offset to give the height of Point 2.

If ***Offset from centreline:***

Offset real box 0

an actual offset from centreline (actually just the chainage of the section) which is used as the Point 2 Offset. **Offset** *can be positive or negative as long as the Point 2 Offset is still on the section.*

Height real box 0

Height is added to the z-value of the section at the Point 2 Offset to give the height of Point 2.

If ***2d cut of string, height from section:***

String string select

the selected string needs to cut the sections in plan wherever the boxing is being applied.

The Point 2 Offset is the offset of the plan cut of the selected string with the section, plus the **Offset** *value.* **Offset** *can be positive or negative as long as the final Point 2 Offset is still on the section.*

Offset real box 0

this value is added to the offset of the plan cut of the section with the selected string to give the Point 2 Offset. **Offset** *can be positive or negative but the final Point 2 Offset must still be on the section.*

Height real box 0

Height is added to the z-value of the section at the Point 2 Offset to give the height of Point 2.

*If **Cut of string:***

String string select

the selected string needs to cut the sections in plan wherever the boxing is being applied.

The Point 2 Offset is the offset of the plan cut of the selected string with the section, plus the **Offset** *value.* **Offset** *can be positive or negative but the final position must still be on the section.*

Offset real box 0

this value is added to the offset of the plan cut of the section with the selected string to give the Point 2 Offset. **Offset** *can be positive or negative but the final Point 2 Offset must still be on the section.*

Height real box 0

Height is added to the z-value of the cut string to give the z-value of Point 2.

Point 2 Xfall

the definition of xfall of the second line

Layer choice box Design, Layer 1 ... Layer 8

Layer to select the vertex from to define the offset of the point the second line goes through.

Xfall from choice box Left, Right

if Left, the xfall of the second line is taken from the segment that is a little to the left of the relative offset.

If Right, the xfall of the second line is taken from the segment that is a little to the right of the relative offset

New vertex

a new vertex is created at the intersection of the two lines

Name name box

the vertex created at the intersection of the two lines is given this name

Active tick box

*if **ticked**, use this Boxing Command.*
*If **not ticked**, don't use this Boxing Command.*

OK button

OK *stores the values in the fields and removes the panel BUT no **recalc** is done.*

Apply button

Apply *stores the values and leaves the panel on the screen.*

*If the **Boxing Definition** is being used in an **Apply MTF** and **Auto recalc** is ticked in the MTF, then whenever the **Apply** button is clicked, then a **recalc** of the associated **Apply MTF** for the MTF is done.*

*If the **Boxing Definition** is being used in a **Boxing Many Function** then whenever the **Apply** button is clicked, a **recalc** of the **Boxing Many Function** is done.*

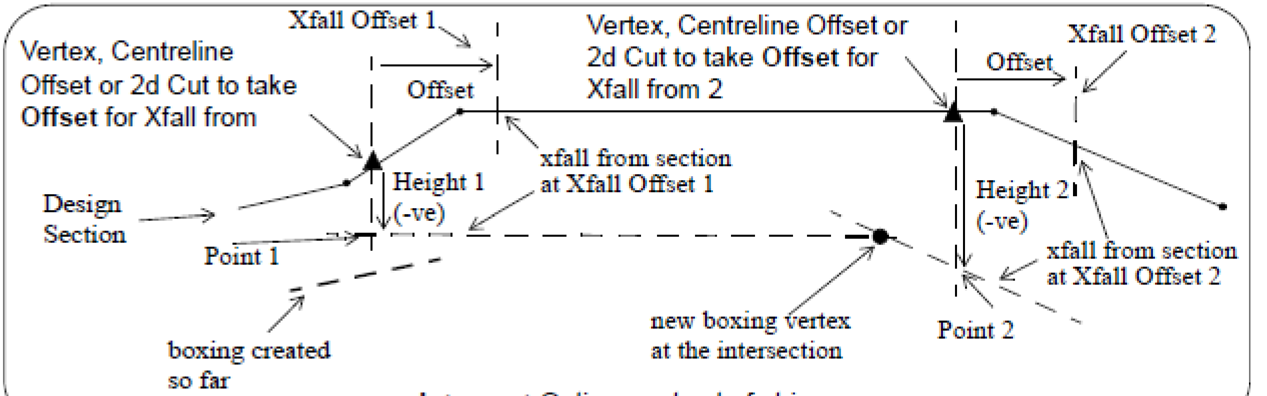
For the typed format of the **intersect** command created by this panel, see 21.10.2.7 Text Format

- Boxing Intersect.

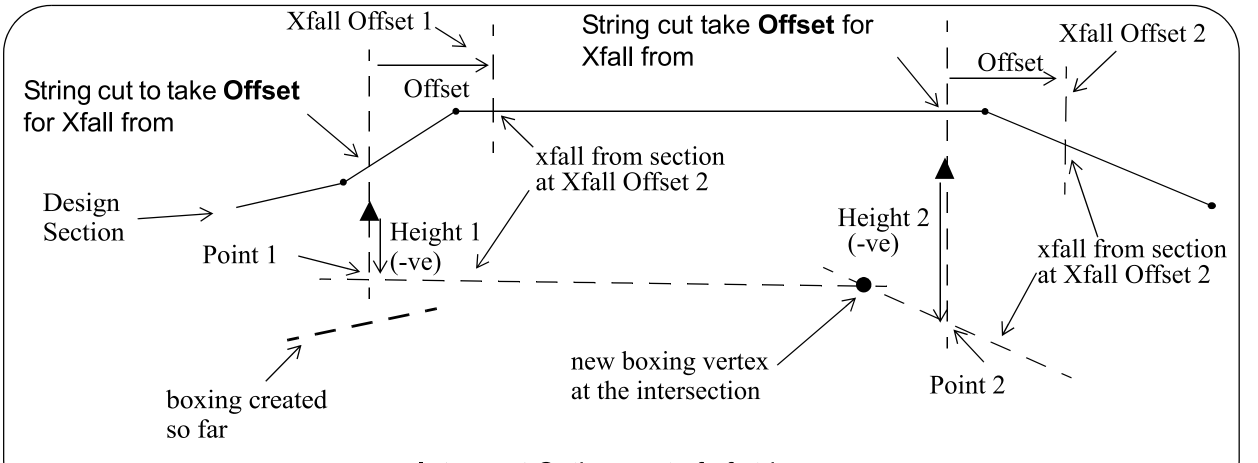
For more information on how Boxing works in ***12d Model***, please go to the section 21.6 What is Boxing?.

Go to the next section 21.7.2.2.8 Boxing Intersect Advanced or return to 21.7.2.2 Commands for Boxing Rules Grid.

Intersect diagrams for more methods of defining *Offsets*:



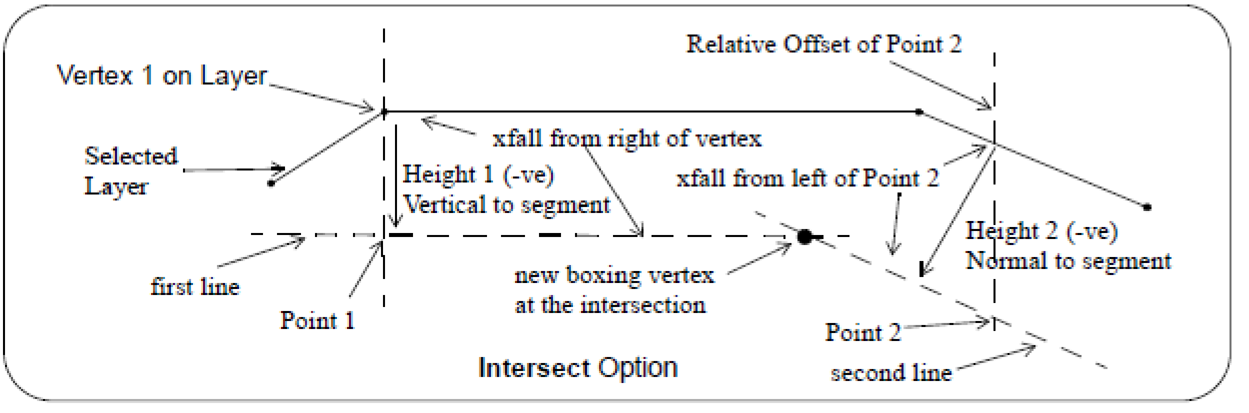
Intersect Option - not cut of string



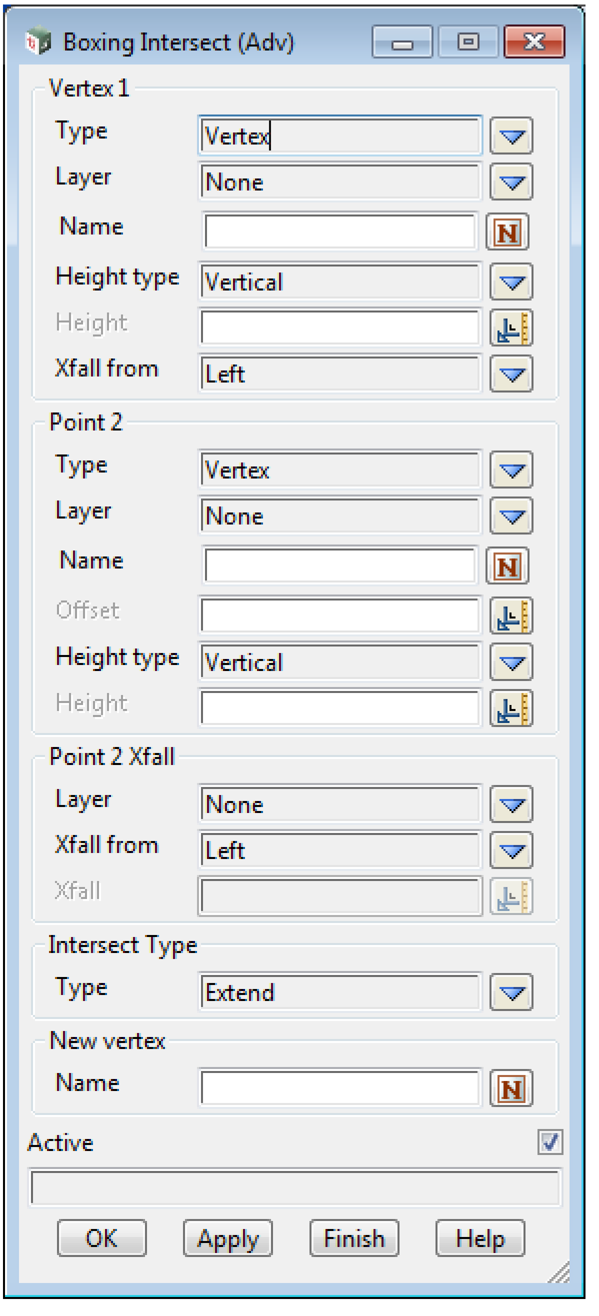
Intersect Option - cut of of strings

21.7.2.2.8 Boxing Intersect Advanced

The **Intersect Advanced** command is similar to the **Intersect** command where a vertex on the boxing section is the intersection of two lines. The major difference is that the distance of the lines from vertices etc can be measured NORMAL to a segment, not just a vertical distance (See 21.7.2.2.7 Boxing Intersect - Intersection of Two Lines).



Selecting **Intersect (Adv)** brings up the **Boxing Intersect (Adv)** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

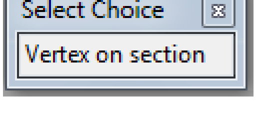
Vertex 1

the definition of the point that the first line goes through. The point can be normal to a segment left or right of a vertex from a Layer.

Layer - after Type	choice box		Design, Layer 1 ... Layer 8

Layer to select the vertex from to define the offset of the point the first line goes through.

Type	choice box		



*If **Vertex on section**:*

Name name box

name of the vertex on **Layer** *to use with* **Type**, **Xfall from**, **Height type** *and* **Height** *to define the offset and height for point that the first line goes through*

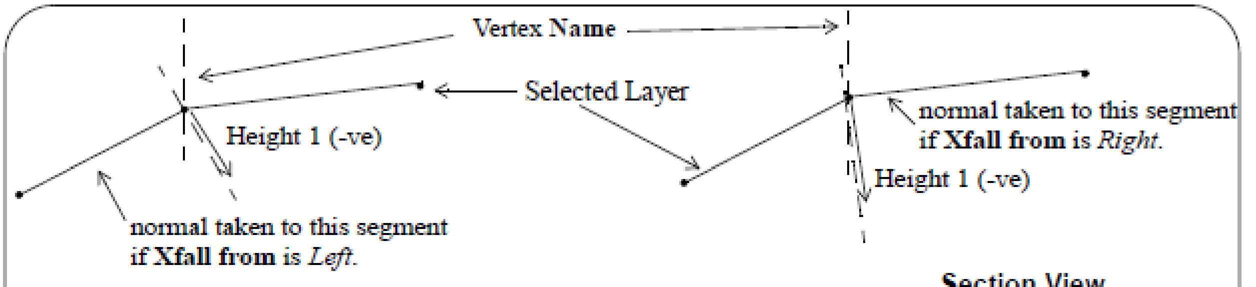
Height type choice box Vertical, Normal

When **Height type** *is Normal, the distance is measured NORMAL to a segment.*

If **Xfall from** *is Left, the xfall to measure the distance NORMAL to is taken from the segment that is slightly to the left of* **Name**.

If **Xfall from** *is Right, the xfall to measure the distance NORMAL to is taken from the segment that is slightly to the right of* **Name**.

When **Height type** *is Vertical, the distance is measured in the vertical plane and* **Xfall from** *is not used. For example, if Height is -0.5, the vertex is dropped by 0.5 from* **Layer**



Vertex Advanced Option - Normal

Height real box

depending on **Height type**, *Height is added to added vertically or normally to give the z-value of the point the first line goes through.*

Xfall from choice box Left, Right

if Left, the xfall for the line is taken from the segment that is a little left of the selected vertex.

If Right, the xfall for the line is taken from the segment that is a little right of the selected vertex

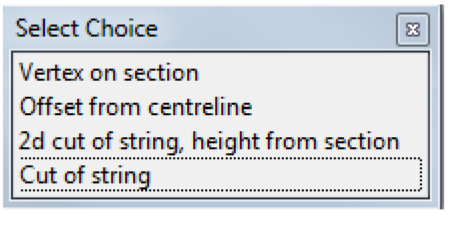
Point 2

the definition of the point that the second line goes through.

Layer choice box Design, Layer 1 ... Layer 8

Layer to select in defining the offset of the point the second line goes through. The point can be normal to a segment left or right of the relative offset from Layer.

Type choice box



*If **Vertex on section**:*

Name name box

name of the vertex on the cross section to use with **Offset** *to defined the Point 2 Offset. And used with* **Height type** *and* **Height** *to define the height of Point 2.*

Offset real box 0

this value is added to the offset of the vertex given in the **Name** *field to define the Point 2 Offset.* **Offset** *can be positive or negative as long as the final Point 2 Offset is still* ***on*** *the section.*

If ***Offset from centreline:***

Offset real box 0

an actual offset from centreline (actually just the chainage of the section) which is used as the Point 2 Offset. **Offset** *can be positive or negative as long as the Point 2 Offset is still on the section.*

If ***2d cut of string, height from section:***

String string select

the selected string needs to cut the sections in plan wherever the boxing is being applied.

The Point 2 Offset is the offset of the plan cut of the selected string with the section, plus the **Offset** *value.* **Offset** *can be positive or negative as long as the final Point 2 Offset is still on the section.*

Offset real box 0

this value is added to the offset of the plan cut of the section with the selected string to give the Point 2 Offset. **Offset** *can be positive or negative but the final Point 2 Offset must still be on the section.*

If ***Cut of string:***

String string select

the selected string needs to cut the sections in plan wherever the boxing is being applied.

The Point 2 Offset is the offset of the plan cut of the selected string with the section, plus the **Offset** *value.* **Offset** *can be positive or negative but the final position must still be on the section.*

Offset real box 0

this value is added to the offset of the plan cut of the section with the selected string to give the Point 2 Offset. **Offset** *can be positive or negative but the final Point 2 Offset must still be on the section.*

Height type choice box Vertical, Normal

When **Height type** *is Normal, the distance is measured NORMAL to a segment.*

If **Xfall from** *is Left, the xfall to measure the distance NORMAL to is taken from the segment that is slightly to the left of the Point 2 offset.*

If **Xfall from** *is Right, the xfall to measure the distance NORMAL to is taken from the segment that is slightly to the right of the Point 2 offset.*

When **Height type** *is Vertical, the distance is measured in the vertical plane and* **Xfall from** *is not used. For example, if Height is -0.5, the vertex is dropped by 0.5 from* **Layer**

Height real box 0

depending on **Height type**, **Height** *is added vertically or normally to give the z-value of the point the second line goes through.*

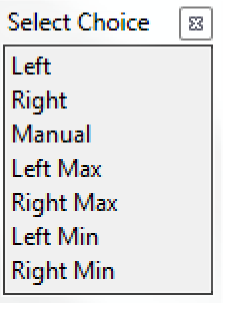
Point 2 Xfall

the definition of xfall of the second line. The xfall comes from the segment in the elected Layer slightly to the left or right of the Point 2 offset

Layer choice box Design, Layer 1 ... Layer 8

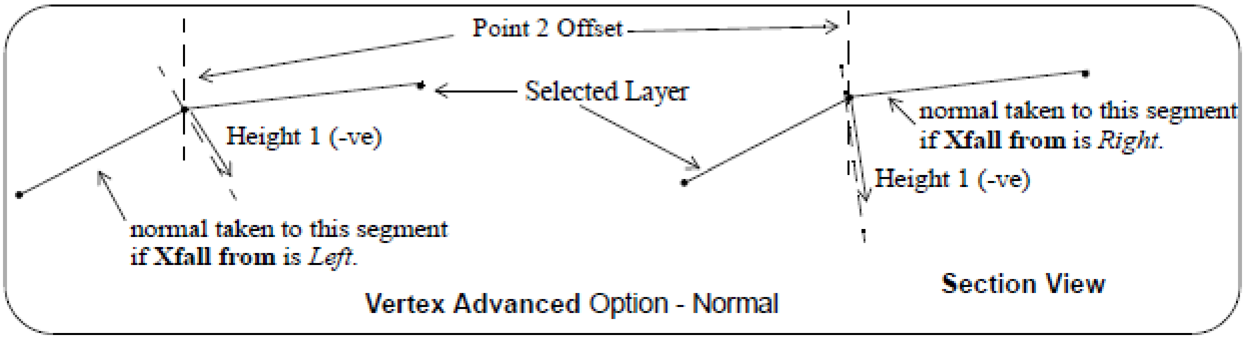
Layer to select the xfall from for the second line.

Xfall from choice box



if Left, the xfall of the second line is taken from the segment that is a little to the left of Point 2 Offset.

If Right, the xfall of the second line is taken from the segment that is a little to the right of Point 2 Offset.



If Manual, the xfall is taken from the **Xfall** *field.*

If Left Max, the xfall on the left side of Point 2 will never exceed the value in the **Xfall** *field. If the xfall is greater than the value in* **Xfall** *field then the xfall takes the value in the* **Xfall** *field.*

If Right Max, the xfall on the right side of Point 2 will never exceed the value in the **Xfall** *field. If the xfall is greater than the value in* **Xfall** *field then the xfall takes the value in the* **Xfall** *field.*

If Left Min, the xfall on the left side of Point 2 will never be less than the value in the **Xfall** *field. If the xfall is less than the value in* **Xfall** *field then the xfall takes the value in the* **Xfall** *field.*

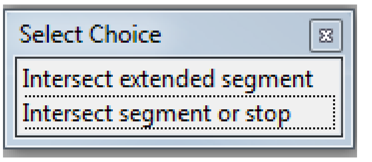
If Right Min, the xfall on the right side of Point 2 will never be less than the value in the **Xfall** *field. If the xfall is less than the value in* **Xfall** *field then the xfall takes the value in the* **Xfall** *field.*

For example, to force a crown in the subgrade no matter what the super elevation of the road is, you would use Left Min on the left hand side of the road and Right Max on the right side of the road.

Intersection Type

the definition of xfall of the second line. The xfall comes from the segment in the elected Layer slightly to the left or right of the Point 2 offset

Type choice box



*if **Intersect extended segment**,*

*If **Intersect segment stop**,*

New vertex

a new vertex is created at the intersection of the two lines

Name name box

the vertex created at the intersection of the two lines is given this name

Active tick box

*if **ticked**, use this Boxing Command.*
*If **not ticked**, don't use this Boxing Command.*

OK button

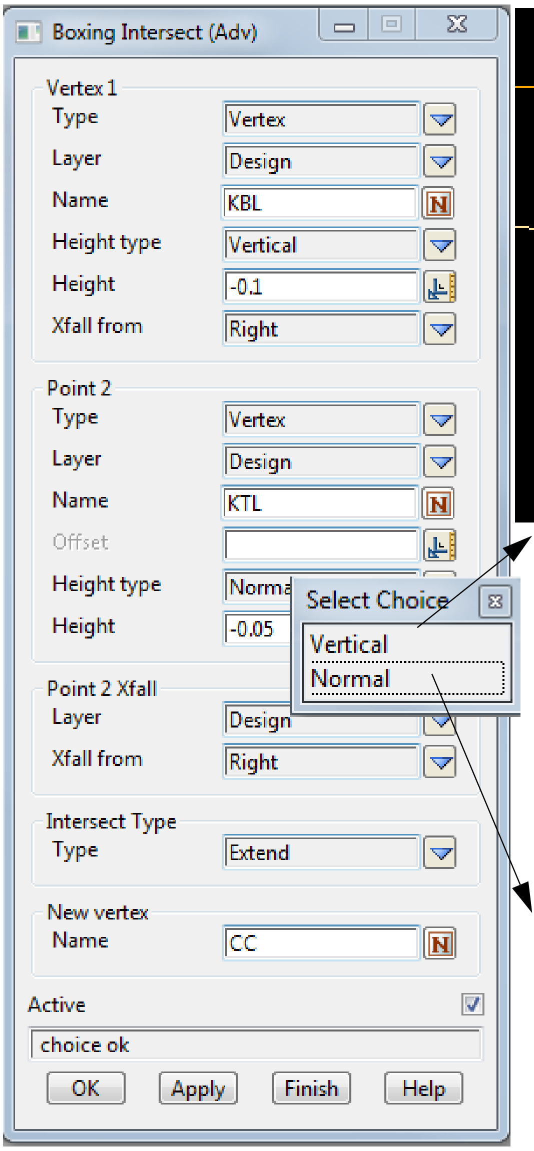
OK *stores the values in the fields and removes the panel BUT no **recalc** is done.*

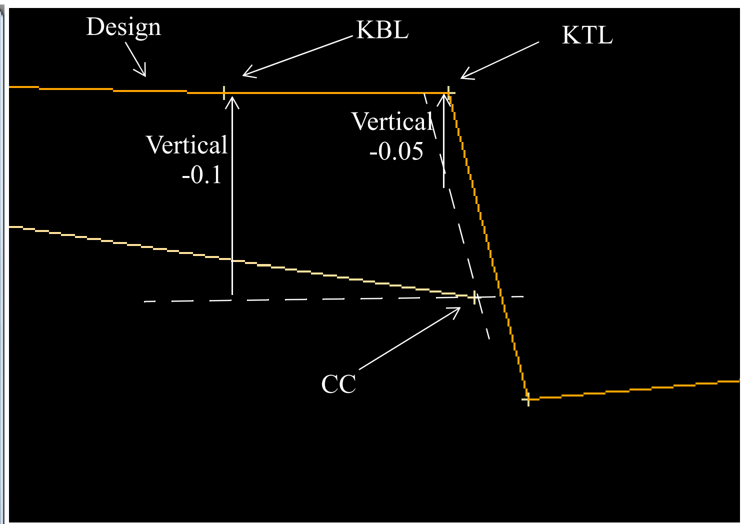
Apply button

Apply *stores the values and leaves the panel on the screen.*

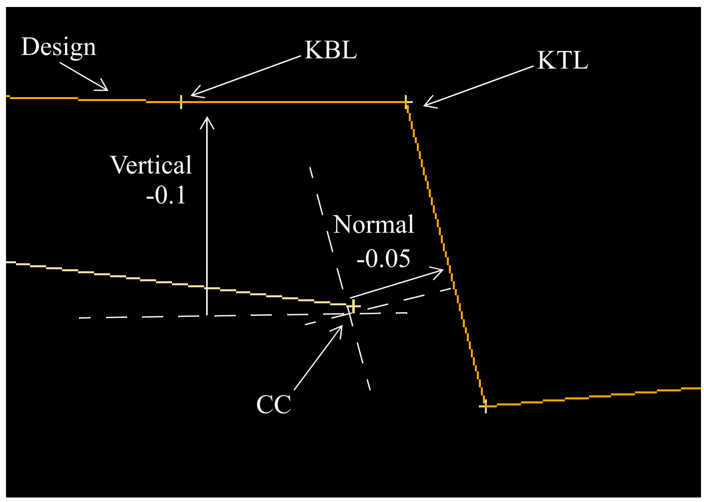
*If the **Boxing Definition** is being used in an **Apply MTF** and **Auto recalc** is ticked in the MTF, then whenever the **Apply** button is clicked, then a **recalc** of the associated **Apply MTF** for the MTF is done.*

*If the **Boxing Definition** is being used in a **Boxing Many Function** then whenever the **Apply** button is clicked, a **recalc** of the **Boxing Many Function** is done.*





Intersect (Adv) - Point 2 Vertical to Design Layer



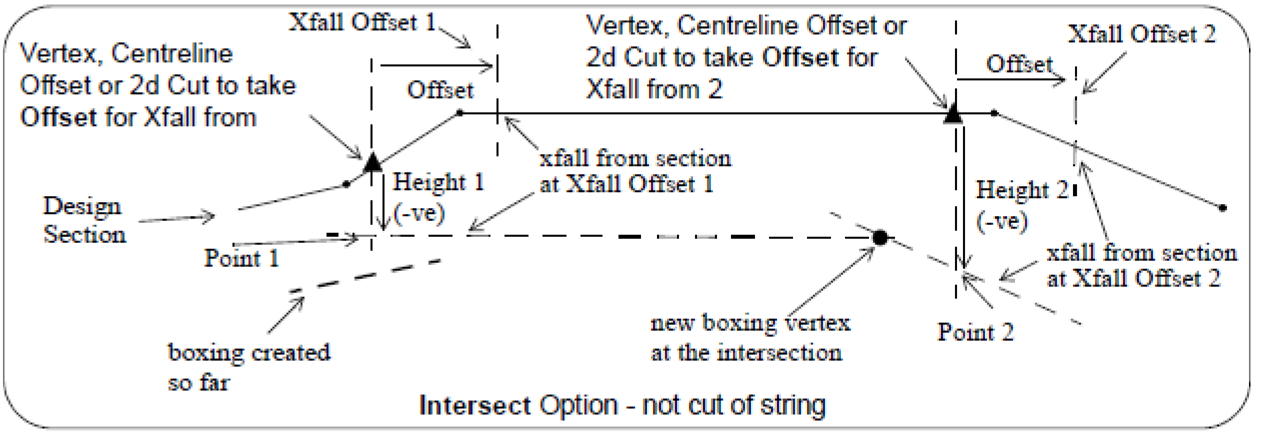
Intersect (Adv) - Point 2 Normal to Design Layer

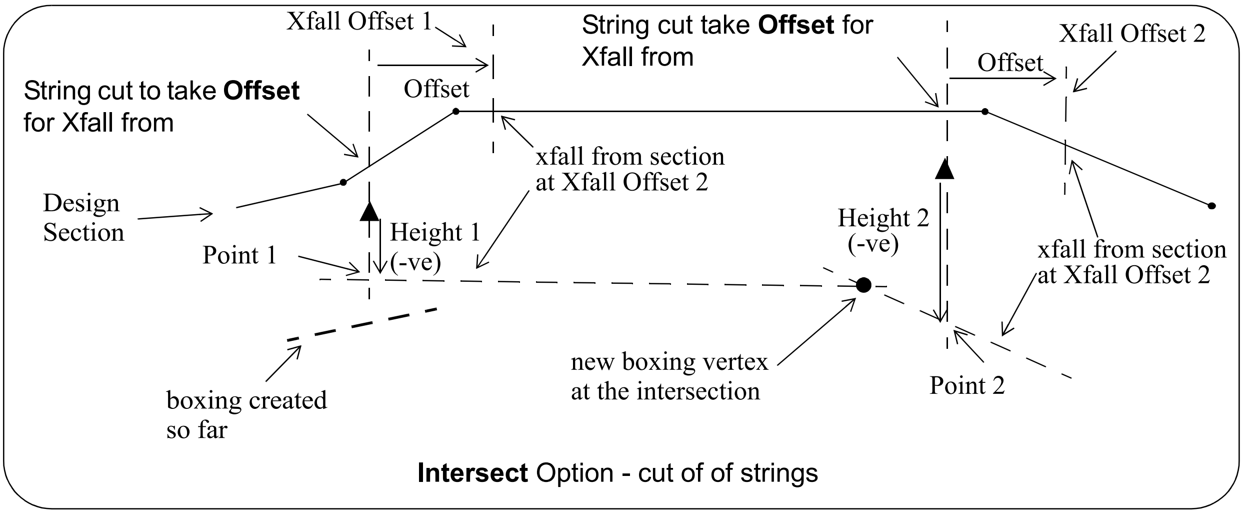
For the typed format of the **intersect Advanced** command created by this panel, see 21.10.2.7 Text Format - Boxing Intersect.

For more information on how Boxing works in ***12d Model***, please go to the section 21.6 What is Boxing?.

Go to the next section 21.7.2.2.9 Boxing Xfall Point or return to 21.7.2.2 Commands for Boxing Rules Grid.

Intersect Adv diagrams for more methods of defining *Offsets*:


Xfall Offset 1
Vertex, Centreline Offset or 2d Cut to take Offset for Xfall from
Offset
Vertex, Centreline Offset or 2d Cut to take Offset for Xfall from 2
Offset
Xfall Offset 2
xfall from section at Xfall Offset 1
Height 1 (-ve)
Height 2 (-ve)
Design Section
Point 1
xfall from section at Xfall Offset 2
boxing created so far
new boxing vertex at the intersection
Point 2
Intersect Option - not cut of string


Xfall Offset 1
String cut take **Offset** for Xfall from
Xfall Offset 2
Offset
String cut to take **Offset** for Xfall from
Offset
xfall from section at Xfall Offset 2
Design Section
Height 1 (-ve)
Height 2 (-ve)
Point 1
xfall from section at Xfall Offset 2
new boxing vertex at the intersection
Point 2
boxing created so far
Intersect Option - cut of of strings

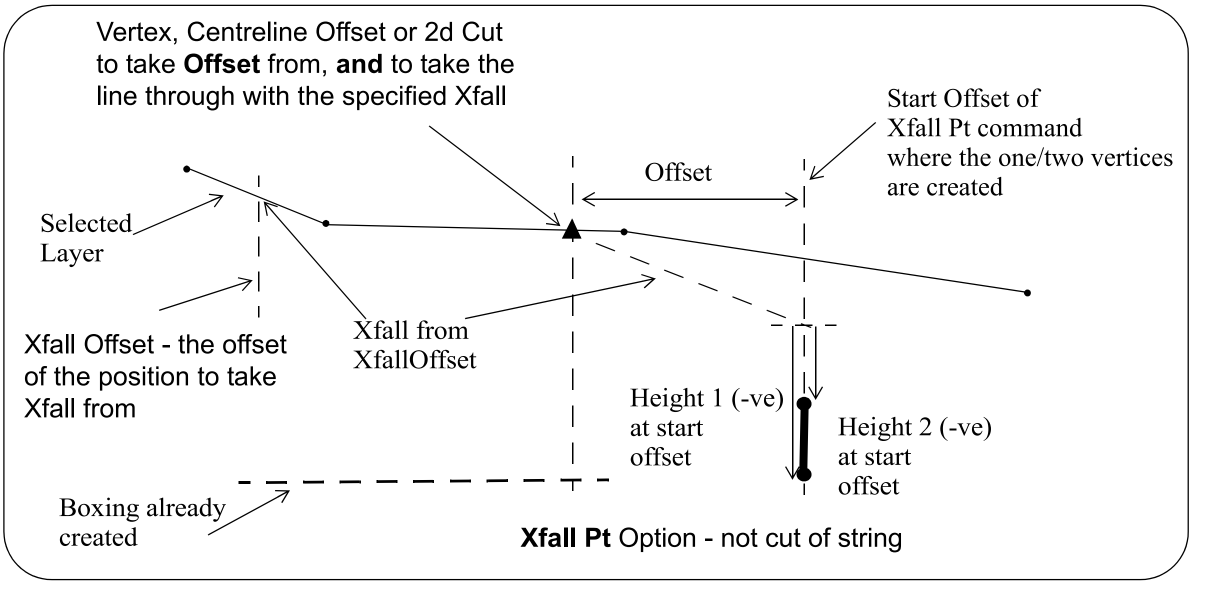
21.7.2.2.9 Boxing Xfall Point

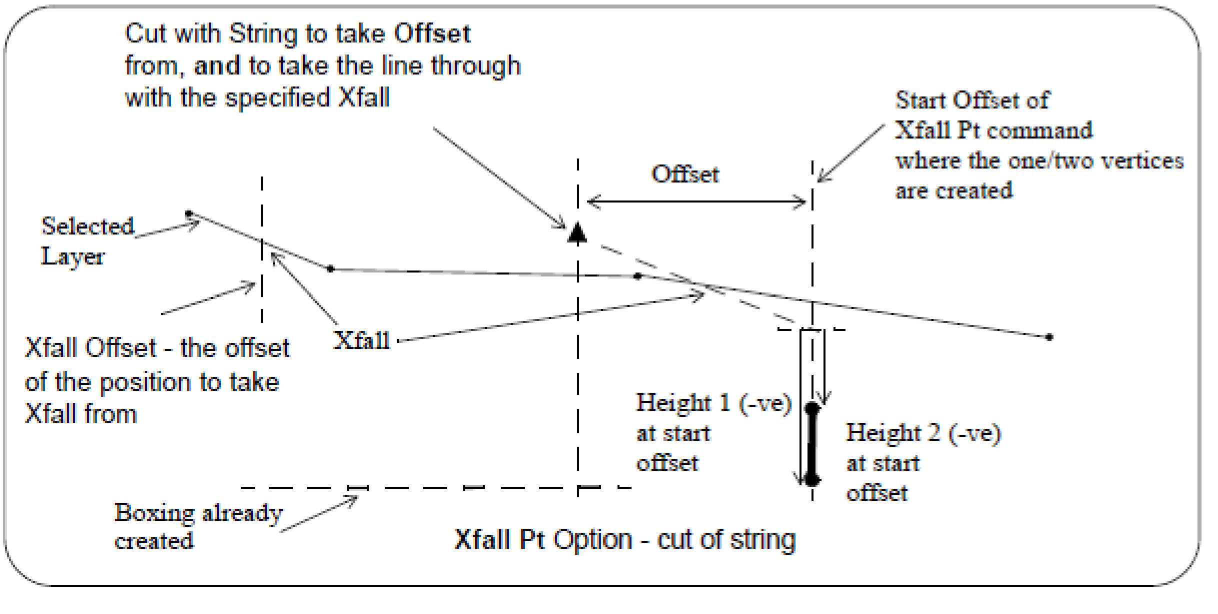
The **Xfall Pt** command is similar to the **Vertex** command in that it inserts one or two new vertices at the given Start *Offset*.

However, instead of adjusting the height from the z-value of the Layer, the ***Xfall Pt*** command extrapolates from the z-value of the selected Layer or 2d cut with a selected string (depending on Type) using the Xfall at the Xfall Offset given in the ***Xfall Pt*** command.

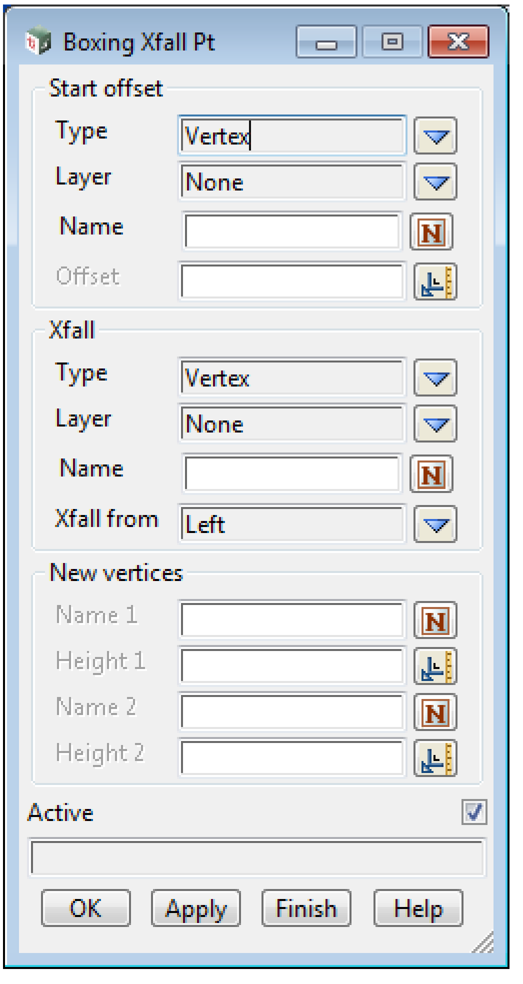
Like **Vertex**, **Xfall Pt** can be used to quickly create a vertical wall of two vertices.

Note - to allow the boxing sections and strings to triangulate, ***12d Model*** increases the offset of the second vertex by ten thousandth of a unit.





Selecting **Xfall pt** brings up the **Boxing Xfall Pt** panel



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

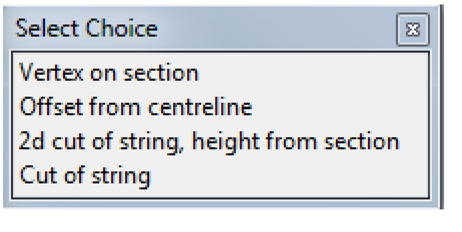
Start offset

Start Offset defines the vertex, plan cut or cut string to define where the grade line start goes through, and then the Offset defines where the line is projected to and creates the one or two new vertices.

Layer - after Type choice box None, Design, Layer 1 ... Layer 8

Layer to use with **Type** *to select the point the line goes through, and for how far.*

Type choice box



*If **Vertex on section**:*

Name name box

name of the vertex on the given **Layer** *that the line will go though. The line will have xfall given in the **Xfall** section.* **Offset** *is added to the offset of the vertex to defined the Start Offset, and the line is extended thought the vertex to Start Offset.*

Height 1/Height 2 *are then used with the projected point to define the heights of the new boxing* *vertices.*

Offset real box 0

this value is added to the offset of the vertex given in the **Name** *field to define the Start Offset.* **Offset** *can be positive or negative as long as the final Start Offset is still* ***on*** *the section.*

Height 1/Height 2 real box 0

Height 1/Height 2 is added to the z-value of the line that is extended to Start Offset to give the z-values of the new boxing vertices.

*If **Offset from centreline:***

Offset real box 0

an actual offset from centreline (actually just the chainage of the section) which is used as the Start Offset. **Offset** *can be positive or negative as long as it is still on the section.*

*If **Offset from centreline is used**, the* **Xfall** *section is not used.*

Height 1/Height 2 real box 0

Height 1/Height 2 is added to the z-value of the section at the Start Offset, to give the z-values of the new boxing vertices. For example, if Height is -0.3, the vertex is dropped by 0.3

*If **2d Cut of string, height from section:***

String string select

The Start Offset is the offset of the plan cut of the selected string with the section from **Layer***, plus the* **Offset** *value.*

*The **plan cut** on the section from* **Layer** *is used as the position to take a line through with xfall given at the Xfall Offset defined in the Xfall section. This goes through the plan cut on* **Layer** *and is extended through to the z-value at Start Offset.*

Height 1/Height 2 *are then added to the z-value at Start Offset to define the heights of the new boxing vertices.*

The selected string needs to cut the sections from **Layer** *in plan wherever the boxing is being applied.*

Offset real box 0

offset to add to the offset of the plan cut of the section from **Layer** *with the selected string to give the Start Offset.* **Offset** *can be positive or negative but the final Start Offset must still be on* **Layer***.*

Height 1/Height 2 real box 0

Height 1/Height 2 is used with the z-value of the extended line at the Start Offset to give the z-value of the new boxing vertices.

*If **Cut of string:***

String string select

*the Start Offset is the offset of the **plan** cut of the selected string with the section from* **Layer***, plus the* **Offset** *value.* **Offset** *can be positive or negative but the final position must still be on the section.*

*The line goes through the **3d cut** with the xfall given at the Xfall Offset defined in the Xfall section. This line goes through the 3d cut point and is extended to Start Offset to give a z-value.*

Height 1/Height 2 *are then added to the projected point to define the heights of the new boxing vertices.*

The selected string needs to cut the sections in plan wherever the boxing is being applied.

Offset real box 0

offset to add to the offset of the plan cut of the section with the selected string to give the Start Offset. **Offset** *can be positive or negative but the final Start Offset must still be on the section.*

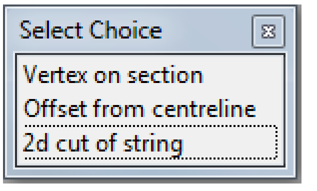
Height 1/Height 2 real box 0

Height 1/Height 2 is used with the z-value of the extended line at the Start Offset to give the z-value of the new boxing vertices.

Xfall

the definition of the offset that Xfall is taken from.

Type choice box



For the documentation on the methods of defining the Xfall Offset, see 21.7.2.3 Defining Relative Offset for Boxing Commands.

The xfall is taken from the point on the section in **Layer** *with offset Xfall Offset.*

Name 1 name box

if non blank, a vertex is created on the boxing section at Start Offset with this name and the appropriate height.

If blank, no vertex is created.

Height 1 real box

Height 1 to used in the calculation height of the new boxing vertex

Name 2 name box

if non blank, a vertex is created on the boxing section at Start Offset with this name and the appropriate height.

If blank, no vertex is created.

Height 2 real box

Height 2 to used in the calculation height of the new boxing vertex

Note - to allow the boxing sections and strings to triangulate, if two vertices are created, ***12d Model*** actually increases the offset of the second vertex by ten thousandth of a unit.

Active tick box

*if **ticked**, use this Boxing Command.*
*If **not ticked**, don't use this Boxing Command.*

OK button

OK *stores the values in the fields and removes the panel BUT no **recalc** is done.*

Apply button

Apply *stores the values and leaves the panel on the screen.*

*If the **Boxing Definition** is being used in an **Apply MTF** and **Auto recalc** is ticked in the MTF, then whenever the **Apply** button is clicked, then a **recalc** of the associated **Apply MTF** for the MTF is done.*

*If the **Boxing Definition** is being used in a **Boxing Many Function** then whenever the **Apply** button is clicked, a **recalc** of the **Boxing Many Function** is done.*

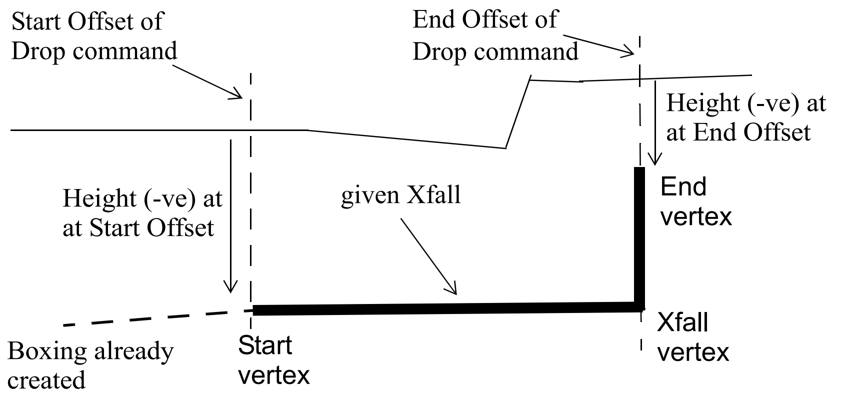
For the typed format of the **Xfall Point** command created by this panel, see 21.10.2.8 Text Format - Boxing Xfall Point.

For more information on how Boxing works in ***12d Model***, please go to the section 21.6 What is Boxing?.

Go to the next section 21.7.2.2.10 Boxing Drop or return to 21.7.2.2 Commands for Boxing Rules Grid.

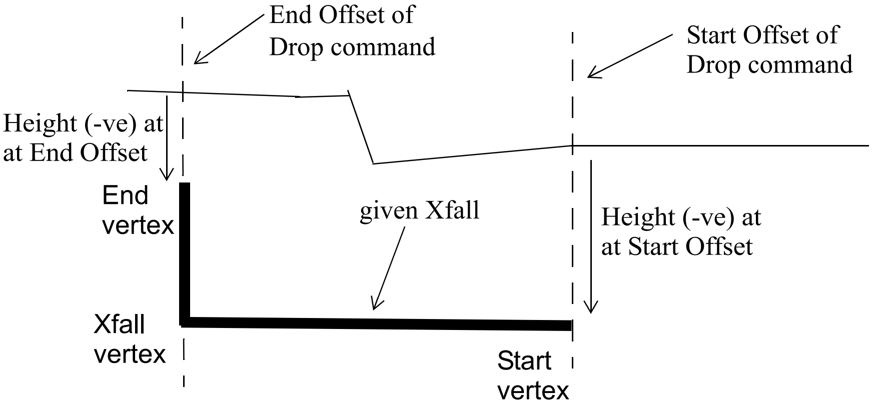
21.7.2.2.10 Boxing Drop

The **Drop** command is a quick way of creating three vertices, the first two on a line of given Xfall, and the third vertically above the second vertex.



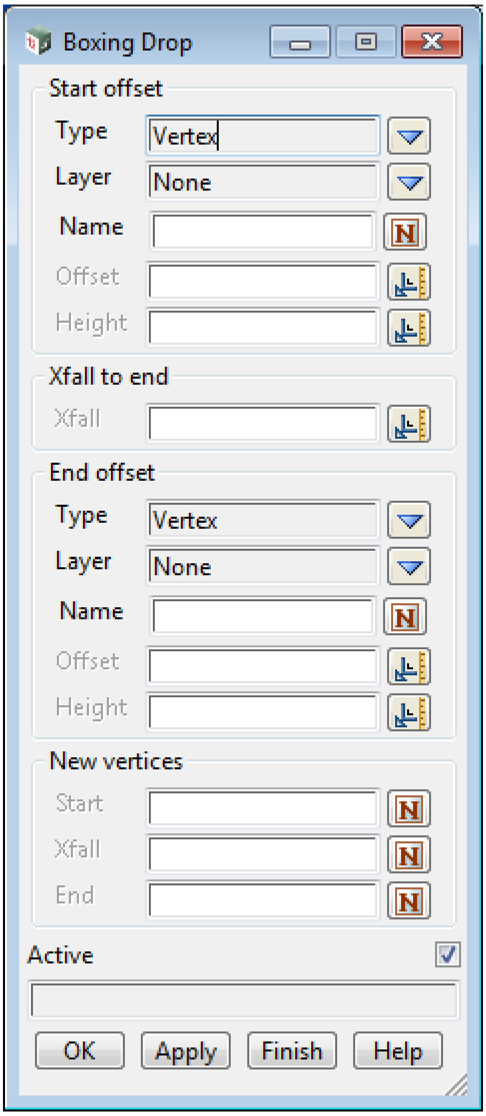
Drop Option - Start Offset before End Offset

If the **End** Offset is before the **Start** Offset then the ***Drop*** is done in reverse. That is, the line still goes through the Start vertex but it then goes to the **left**.



Drop Option - End Offset before Start Offset

Selecting **Drop (kerb)** brings up the **Boxing Drop** panel



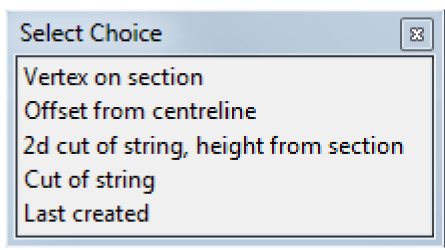
The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Start offset

the first vertex is created with offset Start Offset and z-value determined by Start Offset Type and Height.

Type	choice box		



For the documentation on the methods of defining the ***Start Offset*** *except for* **Last created**, *see* 21.7.2.3 Defining Relative Offset for Boxing Commands

Height	real box	0	

If Start Offset Type is NOT: ***Cut of string:***

*the height of the fist vertex is **Height** added to the z-values of the selected* **Layer** *at Start Offset. For example, if Height is -0.3, the first vertex is 0.3 below the* **Layer** *at Start Offset.*

*If Start Offset Type is: **Cut of string:***

At the plan cut of the string and the START OFFSET **Layer***, **Z-Diff (**the difference of the z-values between the string and START OFFSET* **Layer*****)*** *is calculated and* **Height** *added to it. This combined value is added to the z-value of* **Layer** *at the Start Offset to give the z-values of the first vertex*

Xfall to end

Xfall real box

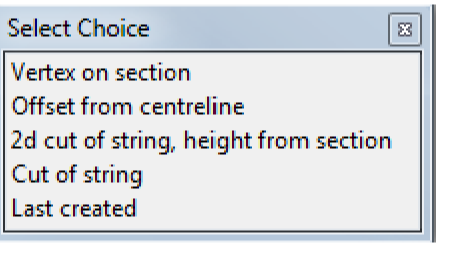
a line is created going through the first vertex (the Start vertex) with the xfall given in the **Xfall** *field. Positive Xfall is up and negative Xfall is down.*

End offset

the second vertex lies on the line through the first vertex when it is extrapolated to End Offset.

The third vertex is created with offset End Offset and z-value determined by End Offset Type and Height.

Type choice box



*For the documentation on the methods of defining the **Start Offset** except for* **Last created***, see* 21.7.2.3 Defining Relative Offset for Boxing Commands.

Height real box 0

*If End Offset Type is NOT: **Cut of string:***

*the height of the last vertex is **Height** added to the z-value of the selected* **Layer** *at End Offset. For example, if Height is -0.1, the third vertex is 0.1 below the* **Layer** *at End Offset.*

*If End Offset Type is: **Cut of string:***

At the plan cut of the string and the END OFFSET **Layer***, **Z-Diff (**the difference of the z-values between the string and END OFFSET* **Layer*****)*** *is calculated and* **Height** *added to it. This combined value is added to the z-value of* **Layer** *at the End Offset to give the z-values of the third vertex*

***IMPORTANT NOTE**: if the End Offset is **before** the Start Offset then the line from the first vertex is reversed and goes to the left rather than the right. The use of **Offset inside** the Start Offset and End Offset sections is also reversed so that a **positive offset goes to the left**.*

New Vertices

Start name box

If **Start** *is non blank then it is the name of the first vertex, otherwise a default name is used.*

Xfall name box

If **Xfall** *is non blank then it is the name of the second vertex, otherwise a default name is used.*

End name box

If **End** *is non blank then it is the name of the third vertex, otherwise a default name is used.*

Active tick box

*if **ticked**, use this Boxing Command.*
*If **not ticked**, don't use this Boxing Command.*

OK button

OK *stores the values in the fields and removes the panel BUT no **recalc** is done.*

Apply button

Apply *stores the values and leaves the panel on the screen.*

*If the **Boxing Definition** is being used in an **Apply MTF** and **Auto recalc** is ticked in the MTF, then whenever the **Apply** button is clicked, then a **recalc** of the associated **Apply MTF** for the MTF is done.*

*If the **Boxing Definition** is being used in a **Boxing Many Function** then whenever the **Apply** button is clicked, a **recalc** of the **Boxing Many Function** is done.*

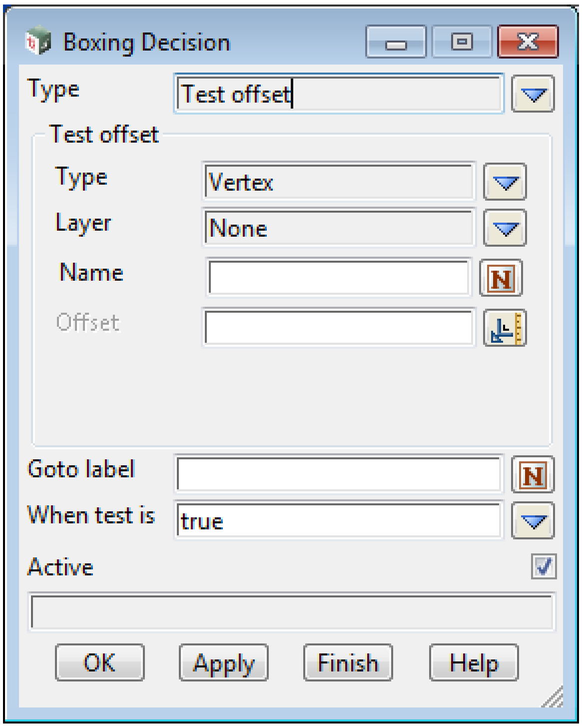
For more information on how Boxing works in ***12d Model***, please go to the section 21.6 What is Boxing?.

Go to the next section 21.7.2.2.11 Boxing Decision or return to 21.7.2.2 Commands for Boxing Rules Grid.

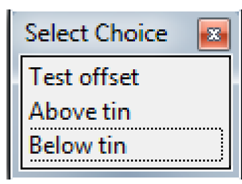
21.7.2.2.11 Boxing Decision

A Test is specified and depending on the result, processing of the boxing commands is sent to a specified ***Label*** in the *Boxing Definition* where processing then continues from the next command after the Label, or no jump is made and processing continues to the next boxing command in the *Boxing Definition.*

Selecting **Decision** brings up the **Boxing Decision** panel



Type choice box

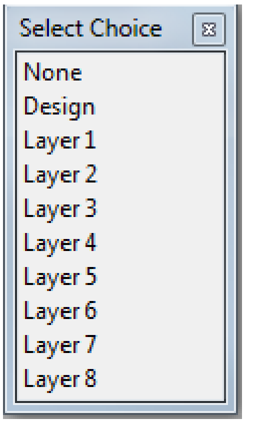


determines the type of test to be made.

If **Type** *is Test offset:*

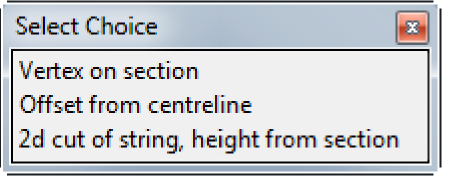
Test Offset

Layer - after Type choice box



Layer to use with **Type** *to defined the test.*

Type choice box



*If **Vertex on section**:*

the test is satisfied if the Test Offset defined by **Layer, Name** *and* **Offset** *is still on* **Layer.**

Name name box

name of the vertex on **Layer** *to use with* **Offset** *to defined the Test Offset. If it is not a name of a vertex then the test is not satisfied.*

Offset real box 0

this value is added to the offset of the vertex given in the **Name** *field to define the Test Offset.* **Offset** *can be positive or negative.*

*If **Offset from centreline:***

the test is satisfied if the Offset is on **Layer.**

Offset real box 0

an actual offset from centreline (actually just the chainage on **Layer***) which is used as the Test Offset.* **Offset** *can be positive or negative.*

*If **2d Cut of string, height from section:***

the test is satisfied if the selected string cuts **Layer** *in plan and the Test Offset given by the plan cut plus* **Offset** *is still on* **Layer***.*

String string select

the selected string needs to cut the sections of **Layer** *in plan wherever the boxing decision is being applied.*

The Test Offset is the offset of the plan cut of the selected string with **Layer***, plus the* **Offset** *value.*

Offset real box 0

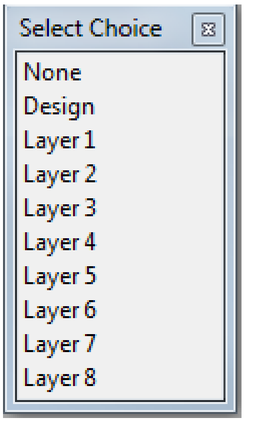
offset to add to the offset of the plan cut of the section from **Layer** *with the selected string to give the Test Offset.* **Offset** *can be positive or negative but the final Test Offset must still be on* **Layer***.*

End of definition for **Type** *equal to Test offset.*

If **Type** *is* ***Above Tin or Below Tin:***

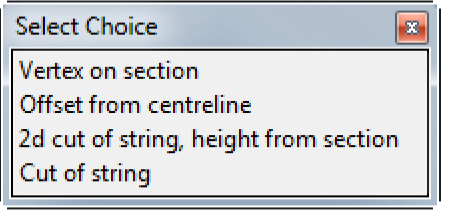
Above/Below Tin decision

Layer - Below Type choice box



Layer to use with **Type** *to defined the height for the test.*

Type choice box



If ***Vertex on section****:*

the test is satisfied if the ***height*** *on the section at Test Offset defined by* **Layer, Name** *and* **Offset** *is* ***ABOVE/BELOW*** *the sum of the value given in the field* **Tin height offset** *and the height of the tin given in the field* **Tin** *at Test Offset***.**

Name name box

name of the vertex on **Layer** *to use with* **Offset** *to defined the Test Offset. If it is not a name of a vertex then the test is not satisfied.*

Offset real box 0

this value is added to the offset of the vertex given in the **Name** *field to define the Test Offset.* **Offset** *can be positive or negative.*

If ***Offset from centreline:***

the test is satisfied if the height of the section at Offset is ***ABOVE/BELOW*** *the sum of the value given in the field* **Tin height offset** *and the height of the tin given in the field* **Tin** *at Test Offset***.**

Offset real box 0

an actual offset from centreline (actually just the chainage on **Layer***) which is used as the Test Offset.* **Offset** *can be positive or negative.*

If ***2d Cut of string, height from section:***

the test is satisfied if the selected string cuts **Layer** *in plan and the* ***height*** *of the section at Test Offset given by the plan cut plus* **Offset** *is* ***ABOVE/BELOW*** *the sum of the value given in the field* **Tin height offset** *and the height of the tin given in the field* **Tin** *at Test Offset.*

String string select

the selected string needs to cut the sections of **Layer** *in plan wherever the boxing decision is being applied.*

The Test Offset is the offset of the plan cut of the selected string with **Layer***, plus the* **Offset** *value.*

Offset real box 0

offset to add to the offset of the plan cut of the section from **Layer** *with the selected string to give the Test Offset.* **Offset** *can be positive or negative but the final Test Offset must still be on* **Layer***.*

*If **Cut of string:***

the selected string can not have only null z-values.

The test is satisfied if the selected string cuts **Layer** *in plan and the **height** on the cut string at Test Offset given by the plan cut plus* **Offset** *is **ABOVE/BELOW** the sum of the value given in the field* **Tin height offset** *and the height of the tin given in the field* **Tin** *at Test Offset.*

String string select

the selected string needs to cut the sections of **Layer** *in plan wherever the boxing decision is being applied.*

The Test Offset is the offset of the plan cut of the selected string with **Layer***, plus the* **Offset** *value.*

Offset real box 0

offset to add to the offset of the plan cut of the section from **Layer** *with the selected string to give the Test Offset.* **Offset** *can be positive or negative but the final Test Offset must still be on* **Layer***.*

End of definition for **Type** *equal to **Above Tin** or **Below Tin**.*

Goto label text box

the name of the Label in the Boxing Definition to pass control to.

When test is choice box True, False

*if **True**, then the Goto is done if the test **is** satisfied.*

*If **False**, then the Goto is done test is **not** satisfied.*

If the Goto is not done, then control passes to the next boxing command in the Boxing Definition.

Active tick box

*if **ticked**, use this Boxing Command.*
*If **not ticked**, don't use this Boxing Command.*

OK button

OK *stores the values in the fields and removes the panel BUT no **recalc** is done.*

Apply button

Apply *stores the values and leaves the panel on the screen.*

*If the **Boxing Definition** is being used in an **Apply MTF** and **Auto recalc** is ticked in the MTF, then whenever the **Apply** button is clicked, then a **recalc** of the associated **Apply MTF** for the MTF is done.*

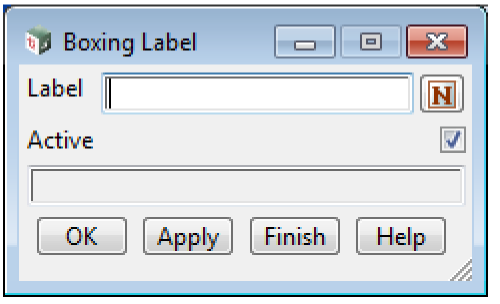
*If the **Boxing Definition** is being used in a **Boxing Many Function** then whenever the **Apply** button is* *clicked, a **recalc** of the **Boxing Many Function** is done.*

For more information on how Boxing works in ***12d Model***, please go to the section 21.6 What is Boxing?.

Go to the next section 21.7.2.2.12 Boxing Label or return to 21.7.2.2 Commands for Boxing Rules Grid.

21.7.2.2.12 Boxing Label

Selecting **Label** brings up the **Boxing Label** panel which is used to define a label for this line in the *Boxing Definition*, and control can be passed to via a **Goto** or a **Decision** boxing command.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
Goto Label	input		

name of the label for this line in the Boxing Definition.

Active	tick box		

*if **ticked**, use this Boxing Command.*
*If **not ticked**, don't use this Boxing Command.*

OK	button		

OK *stores the values in the fields and removes the panel BUT no **recalc** is done.*

Apply	button		

Apply *stores the values and leaves the panel on the screen.*

*If the **Boxing Definition** is being used in an **Apply MTF** and **Auto recalc** is ticked in the MTF, then whenever the **Apply** button is clicked, then a **recalc** of the associated **Apply MTF** for the MTF is done.*

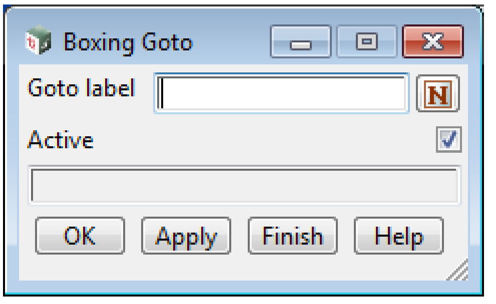
*If the **Boxing Definition** is being used in a **Boxing Many Function** then whenever the **Apply** button is clicked, a **recalc** of the **Boxing Many Function** is done.*

For more information on how Boxing works in ***12d Model***, please go to the section 21.6 What is Boxing?.

Go to the next section 21.7.2.2.13 Boxing Goto or return to 21.7.2.2 Commands for Boxing Rules Grid.

21.7.2.2.13 Boxing Goto

Selecting **Goto** brings up the **Boxing Goto** panel which is used to transfer control to the line in the *Boxing Definition* with the **Label** given in the **Goto label** field of the panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
Goto label	input		

name of the Label in the Boxing Definition to transfer control to.

Active	tick box		

*if **ticked**, use this Boxing Command.*
*If **not ticked**, don't use this Boxing Command.*

OK	button		

OK *stores the values in the fields and removes the panel BUT no **recalc** is done.*

Apply	button		

Apply *stores the values and leaves the panel on the screen.*

*If the **Boxing Definition** is being used in an **Apply MTF** and **Auto recalc** is ticked in the MTF, then whenever the **Apply** button is clicked, then a **recalc** of the associated **Apply MTF** for the MTF is done.*

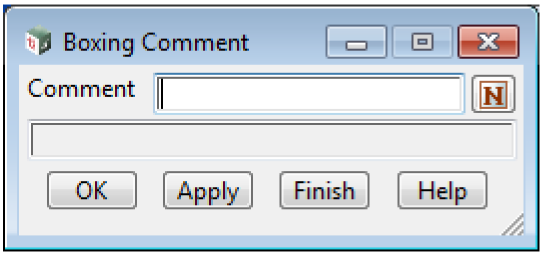
*If the **Boxing Definition** is being used in a **Boxing Many Function** then whenever the **Apply** button is clicked, a **recalc** of the **Boxing Many Function** is done.*

For more information on how Boxing works in ***12d Model***, please go to the section 21.6 What is Boxing?.

Go to the next section 21.7.2.2.14 Boxing Comment or return to 21.7.2.2 Commands for Boxing Rules Grid.

21.7.2.2.14 Boxing Comment

Selecting **Comment** brings up the **Boxing Comment** panel and the panel is used to insert a comment as a line of the *Boxing Definition*. Comments are not processed when running the boxing commands.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
Comment	input		

user comment - for information purposes only

OK	button		

OK *stores the values in the fields and removes the panel BUT no* ***recalc*** *is done.*

Apply	button		

Apply *stores the values and leaves the panel on the screen.*

If the ***Boxing Definition*** *is being used in an* ***Apply MTF*** *and* ***Auto recalc*** *is ticked in the MTF, then whenever the* ***Apply*** *button is clicked, then a* ***recalc*** *of the associated* ***Apply MTF*** *for the MTF is done.*

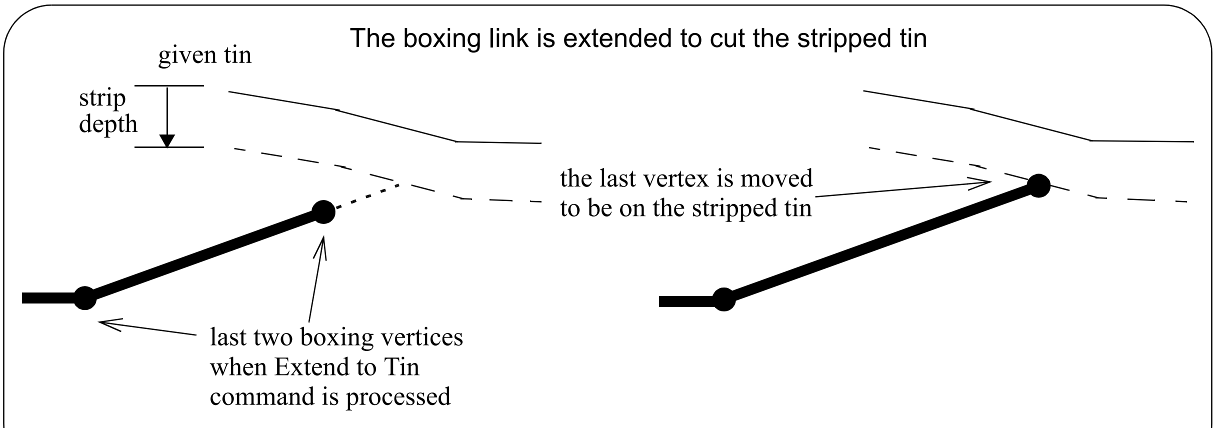
If the ***Boxing Definition*** *is being used in a* ***Boxing Many Function*** *then whenever the* ***Apply*** *button is clicked, a* ***recalc*** *of the* ***Boxing Many Function*** *is done.*

For more information on how Boxing works in ***12d Model***, please go to the section 21.6 What is Boxing?.

Go to the next section 21.7.2.2.15 Boxing Extend Tin or return to 21.7.2.2 Commands for Boxing Rules Grid.

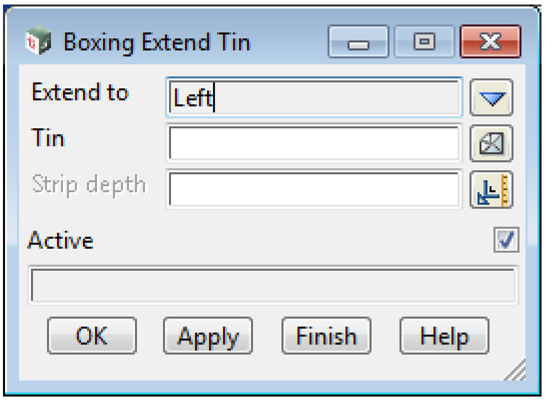
21.7.2.2.15 Boxing Extend Tin

The *Extend to Tin* option is used to extend the link containing the last two vertices (or the first two vertices) already created in the boxing layer until is hits the selected **Stripped Tin**.



Extend to Tin Option - Right Case

Selecting **Extend tin** brings up the **Boxing Extend Tin** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
Extend to	choice box		Left, Right

*if **Right**, then the link through the last two boxing vertices is extended until intersects the stripped tin. The last vertex is then **moved** to the intersection point. Other boxing commands can then follow.*

*If **Left**, then the link through the first two boxing vertices is extended until intersects the stripped tin. The first vertex is then **moved** to the intersection point. Note that this is moving the first vertex of the boxing defined so far. The "Copy from start of section" with the Left Side Interface will add vertices **before** this vertex.*

***Note** - there must be at least two vertices in the boxing before this command can be used.*

If the extend fails, then no vertex is moved and an error message is written to the output window and the boxing processing terminated at that chainage.

Tin	tin box		available tins

the tin, dropped by the **Strip depth**, *to batter to.*

Strip depth	input		

distance below the tin to stop at.

Active tick box

*if **ticked**, use this Boxing Command.*
*If **not ticked**, don't use this Boxing Command.*

OK button

OK *stores the values in the fields and removes the panel BUT no **recalc** is done.*

Apply button

Apply *stores the values and leaves the panel on the screen.*

*If the **Boxing Definition** is being used in an **Apply MTF** and **Auto recalc** is ticked in the MTF, then whenever the **Apply** button is clicked, then a **recalc** of the associated **Apply MTF** for the MTF is done.*

*If the **Boxing Definition** is being used in a **Boxing Many Function** then whenever the **Apply** button is clicked, a **recalc** of the **Boxing Many Function** is done.*

For more information on how Boxing works in ***12d Model***, please go to the section 21.6 What is Boxing?.

Go to the next section 21.7.2.2.16 Boxing End or return to 21.7.2.2 Commands for Boxing Rules Grid.

21.7.2.2.16 Boxing End

Most Boxing commands go from the ***Start Offset*** defined with the command, to the *Start Offset* of the **next** Boxing command.

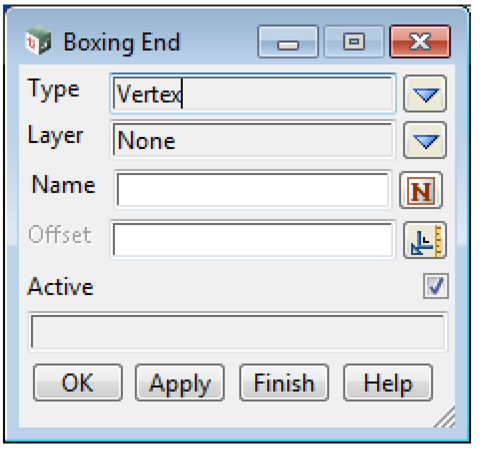
However sometimes a Boxing command needs to be stopped before the ***Start Offset*** of the next Boxing command.

The **End** command can be used to provide a "Start Offset" to act as the finishing position for the previous boxing command. So the **End** command simply terminates the previous Boxing command.

Apart from supplying the End Offset for the previous command, End is a "no operation" command.

Other Boxing commands can then follow the **End** command.

Selecting **End** brings up the **Boxing End** panel

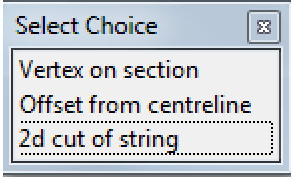


The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

the Relative Offset defined becomes the End Offset for the previous command.

Type choice box



*For the documentation on the methods of defining the **Relative Offset** see* 21.7.2.3 Defining Relative Offset for Boxing Commands.

OK button

OK *stores the values in the fields and removes the panel BUT no **recalc** is done.*

Apply button

Apply *stores the values and leaves the panel on the screen.*

*If the **Boxing Definition** is being used in an **Apply MTF** and **Auto recalc** is ticked in the MTF, then whenever the **Apply** button is clicked, then a **recalc** of the associated **Apply MTF** for the MTF is done.*

*If the **Boxing Definition** is being used in a **Boxing Many Function** then whenever the **Apply** button is* *clicked, a **recalc** of the **Boxing Many Function** is done.*

For the typed format of the **End** command created by this panel, see 21.10.2.9 Text Format - Boxing End.

For more information on how Boxing works in ***12d Model***, please go to the section 21.4 Smart Chainages.

Return to 21.7.2.2 Commands for Boxing Rules Grid.

21.7.2.2.17 Boxing Left Side Interface Xfall from Point

The **Xfall from Point** command creates a line from the left hand end of the boxing already defined and then batters to the ***left*** at a specified xfall until it intersects the design surface. The xfall of the batter is defined as

(a) the xfall on the section at a given section offset

(b) the xfall on the section at a *relative vertex offset*

(c) the xfall on the section at a *relative string offset*

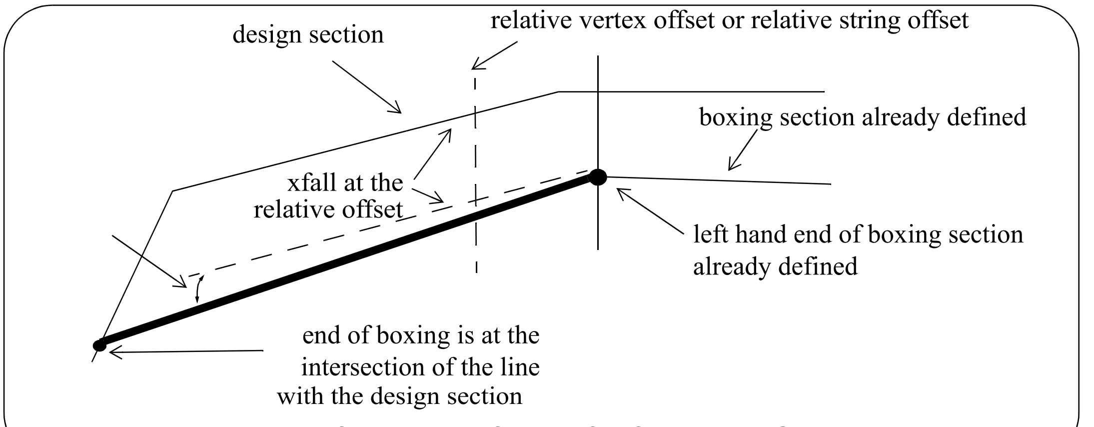
plus in each case:
an additional xfall value **Delta**.

If the design surface is not intersected by the batter, then no extra line is created.

Xfall is percent cross-fall and a positive xfall is up and negative xfall is down.

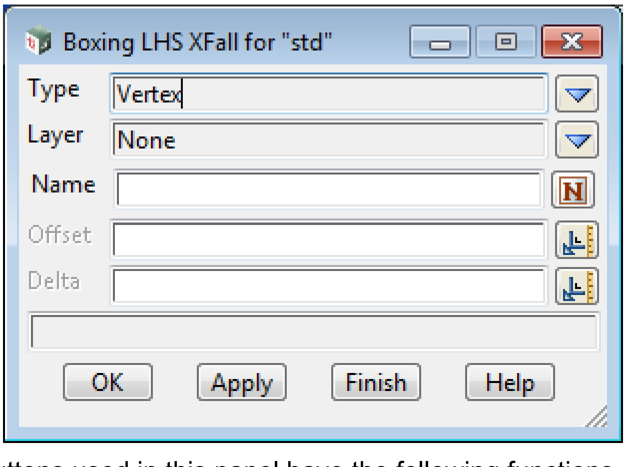
Note that the **Xfall from Point** creates a line at the ***beginning*** of the boxing created by running all the boxing commands in the boxing grid. So it is applied **after** all the other boxing commands have been run.

If **Copy from start of section** is ticked, then the part of the design section from the start of the design section to the offset that is the start of the boxing section, is copied and made the beginning of the boxing section.



Left Hand Interfaced Xfall from Point Option

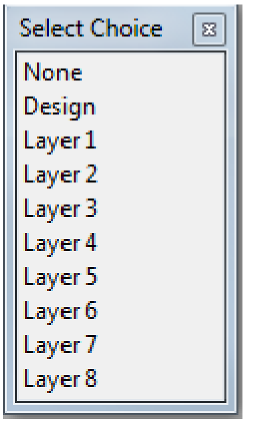
On the **Boxing Rules** panel, selecting the **Set Point** button when the *Left Side Interface* Type is **Xfall from Point** brings up the **Boxing LHS Xfall** panel



The fields and buttons used in this panel have the following functions.

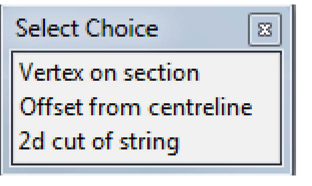
Field Description Type Defaults Pop-Up

Layer - after Type choice box



Layer of sections to use with **Type**.

Type choice box



If ***Vertex on section****:*

Name name box

name of the vertex on **Layer** *to use with* **Offset** *to define the section offset to take the Xfall value from*

Offset real box

this value is added to the offset of the vertex given in the **Name** *field to define the section offset to take the Xfall value from.* **Offset** *can be positive or negative as long as the final section offset is still* ***on*** *the section from* **Layer**.

If ***Offset from centreline:***

Offset real box

an actual offset from centreline (actually just the chainage of the section) which is used as the section from **Layer** *offset to take the Xfall value from.* **Offset** *can be positive or negative as long as it is still on the section. from* **Layer.**

If ***2d cut of string:***

String string select

the selected string needs to cut the sections from **Layer** *in plan wherever the boxing is being applied.*

The section offset to take the Xfall value from is the offset of the plan cut of the selected string with the section from **Layer***, plus the* **Offset** *value.* **Offset** *can be positive or negative but the final section offset must still be on the section from* **Layer**.

Offset real box

offset to add to the offset of the plan cut of the section from **Layer** *with the selected string to give the section offset to take the Xfall value from.* **Offset** *can be positive or negative but the final section offset must still be on the section from* **Layer**.

Delta input 0

add the value **Delta** *to the xfall taken from the section from* **Layer**.

OK/Apply button

OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

For the typed format of the **left_xfall** command created by this panel, see 21.10.2.13 Text Format - Boxing Left_Xfall, for **left_slope** see 21.10.2.14 Text Format - Boxing Left_slope and **left_copy** see 21.10.2.15 Text Format - Boxing Left_copy.

For more information on how Boxing works in ***12d Model***, please go to the section 21.4 Smart Chainages.

Go to the next section 21.7.2.2.18 Boxing Right Side Interface Xfall from Point or return to 21.7.2 Edit Boxing File.

21.7.2.2.18 Boxing Right Side Interface Xfall from Point

The **Xfall from Point** command creates a line from the right hand end of the boxing already defined and then batters to the ***right*** at a specified xfall until it intersects the design surface. The xfall of the batter is defined as

(a) the xfall on the section at a given section offset

(b) the xfall on the section at a *relative vertex offset*

(c) the xfall on the section at a *relative string offset*

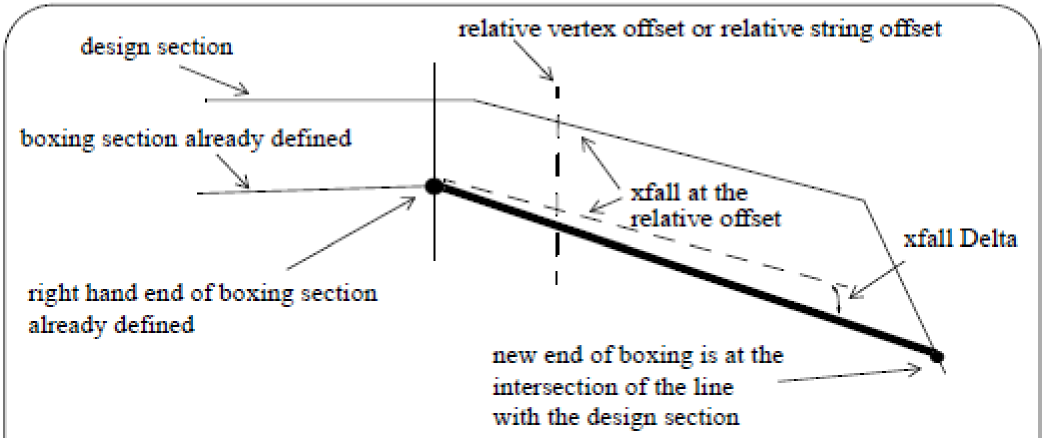
plus in each case:
an additional xfall value **Delta**.

If the design surface is not intersected by the batter, then no extra line is created.

xfall is percent cross-fall and a positive xfall is up and negative xfall is down.

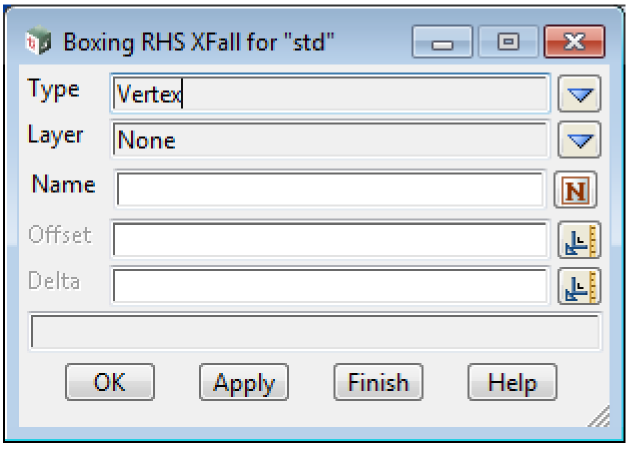
Note that the **Xfall from Point** creates a line at the ***end*** of the boxing created by running all the boxing commands in the boxing grid. So it is applied **after** all the other boxing commands have been run.

If **Copy to end of section** is ticked, then the part of the design section from the offset of the last boxing vertex to the last vertex on the design section, is copied and made the end of the boxing section.



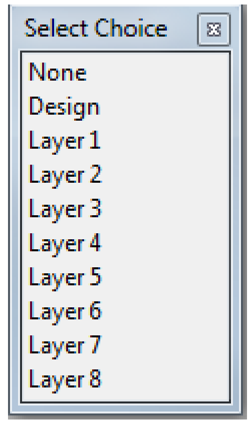
Right Hand Interfaced Xfall from Point Option

On the **Boxing Rules** panel, selecting the **Set Point** button when the *Right Side Interface* Type is **Xfall from Point** brings up the **Boxing RHS Xfall** panel



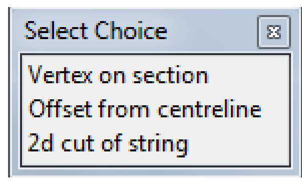
The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Layer - after Type	choice box		



Layer of sections to use with **Type**.

Type	choice box		



If ***Vertex on section****:*

Name name box

name of the vertex on the section from **Layer** *to use with* **Offset** *to define the section offset to take the Xfall value from*

Offset real box

this value is added to the offset of the vertex given in the **Name** *field to define the section from* **Layer** *offset to take the Xfall value from.* **Offset** *can be positive or negative as long as the final section offset is still* ***on*** *the section from* **Layer***.*

If ***Offset from centreline:***

Offset real box

an actual offset from centreline (actually just the chainage of the section) which is used as the section from **Layer** *offset to take the Xfall value from.* **Offset** *can be positive or negative as long as it is still on the section from* **Layer***.*

If ***2d cut of string:***

String string select

the selected string needs to cut the sections from **Layer** *in plan wherever the boxing is being applied.*

The section offset to take the Xfall value from is the offset of the plan cut of the selected string with the section from **Layer***, plus the* **Offset** *value.* **Offset** *can be positive or negative but the final section offset must still be on the section from* **Layer***.*

Offset real box

offset to add to the offset of the plan cut of the section from **Layer** *with the selected string to give the section offset to take the Xfall value from.* **Offset** *can be positive or negative but the final section offset must still be on the section from* **Layer***.*

Delta input 0

add the value **Delta** *to the xfall taken from the section from* **Layer**.

OK/Apply button

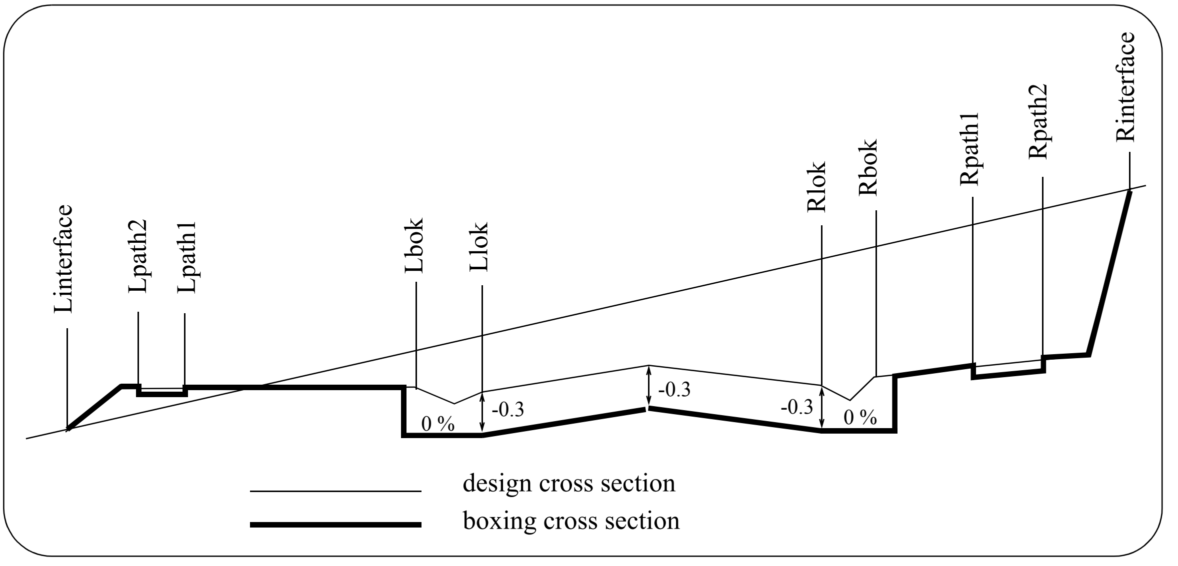
OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

For the typed format of the **right_xfall** command created by this panel, see 21.10.2.10 Text Format - Boxing Right_Xfall, for **right_slope** see 21.10.2.11 Text Format - Boxing Right_slope and **right_copy** see 21.10.2.12 Text Format - Boxing Right_copy.

For more information on how Boxing works in ***12d Model***, please go to the section 21.4 Smart Chainages.

Go to the next section 21.7.2.2.19 Boxing Definitions Examples or return to 21.7.2 Edit Boxing File.

21.7.2.2.19 Boxing Definitions Examples



	Type	Point	Point offset	Details	Value	Active
1	Copy	Point Linterface	0	Height:	0.000	yes
2	Copy	Point Lpath2	0	Height:	-0.075	yes
3	Copy	Point Lpath1	0	Height:	0.000	yes
4	Xfall	Point Lbok	-0.15	Control name: Control offset: Control height: Xfall:	Llok 0.000 -0.300 0.000	yes
5	Copy	Point Llok	0	Height:	-0.300	yes
6	Xfall	Point Rlok	0	Control name: Control offset: Control height: Xfall:	Rlok 0.000 -0.300 0.000	yes
7	Copy	Point Rbok	0.15	Height:	0.000	yes
8	Copy	Point Rpath1	0	Height:	-0.095	yes
9	Copy	Point Rpath2	0	Height:	0.000	yes
10	End	Point Rinterface	0			yes

Line 1
copy from “Linterface” to “Lpath2”

Line 2
copy & drop by 0.075 from “Lpath 2" to “Lpath 1"

Line 3
copy from “Lpath1” to 0.15 before “Lbok”

Line 4
go from 0.15 before “Lbok” to “Llok” at depth 0.3 below “Llok”, with cross fall 0

Line 5
copy and drop by 0.3 from “Llok” to “Rlok”

Line 6
go from “Rlok” to 0.15 past “Rbok” at depth 0.3 below “Rlok”, with cross fall 0

Line 7
copy from 0.15 past “Rbok” to “Rpath1”

Line 8
copy & drop by 0.095 from “Rpath1” to “Rpath2”

Line 9
copy from “Rpath2” to “Rinterface”

Line 10
Just giving the End Offset for the Copy on Line 9

boxing “1” **{**

```
"Linterface",       copy                                          // line 1
"Lpath2",           copy,      -0.075                             // line 2
"Lpath1",           copy                                          // line 3
"Lbok" - 0.15,      xfall,     "Llok",     -0.3,      0           // line 4
"Llok",             copy,      -0.3                               // line 5
"Rlok",             xfall,     "Rlok",     -0.3,      0           // line 6
"Rbok" + 0.15,      copy                                          // line 7
"Rpath1",           copy       -0.095                             // line 8
"Rpath2",           copy                                          // line 9
"Rinterface",       end                                           // line 10
}
```

Description

Line 1	copy from “Linterface” to “Lpath2”
Line 2	copy and drop by 0.075 from “Lpath 2" to “Lpath 1"
Line 3	copy from “Lpath1” to 0.15 before “Lbok”
Line 4	go from 0.15 before “Lbok” to “Llok” at depth 0.3 below “Llok”, with cross fall 0
Line 5	copy and drop by 0.3 from “Llok” to “Rlok”
Line 6	go from “Rlok” to 0.15 past “Rbok” at depth 0.3 below “Rlok”, with cross fall 0
Line 7	copy from 0.15 past “Rbok” to “Rpath1”
Line 8	copy and drop by 0.095 from “Rpath1” to “Rpath2”
Line 9	copy from “Rpath2” to “Rinterface”
Line 10	just giving the End Offset for the Copy in Line 9

```
boxing “left narrow” {                    // boxing defined from just before lkerb to m001
            “lkerb”   - 0.7  ,              copy
            “m001”   - 1.0,                 copy    ,     -0.6
            "m001",                         end
  }
boxing “left” {     // boxing defined from the left start of the section to m001 (the centreline say)
            “lkerb”,      copy,        -0.6
            “m001”,       end
            left_slope     5.0
            left_copy       1
  }
boxing “right” {                // boxing defined from m001 to the far right of the section
            “m001”,       copy,        0.6
            “rkerb”,      end
            right_slope   5.0
            right_copy 1
  }
boxing “full” {         // a full width boxing - defines the entire boxing going from left to right
            “lkerb”,      copy,        -0.6
            “rkerb”,      end
            left_slope     5.0
            left_copy    1
            right_slope -5.0
            right_copy 1
  }
```

For more information on how Boxing works in ***12d Model***, please go to the section 21.4 Smart Chainages or to the section 21.6.2 Applying Boxing.

21.7.2.3 Defining Relative Offset for Boxing Commands

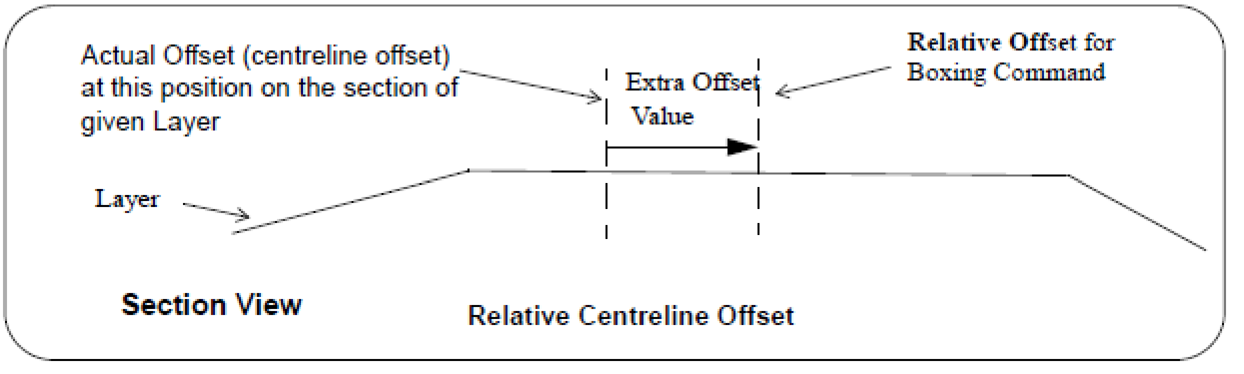
Most boxing commands use ***Relative Offsets*** to define offset positions across Design or Boxing Layers. For example, the ***Start Offset*** is a relative offset and is used in most boxing commands to specify where the boxing command starts.

There are a number of ways the ***Relative Offset*** can be defined:

(a) **Offset from centreline**

Relative to an actual Offset for the Design or Boxing Layer - relative offset

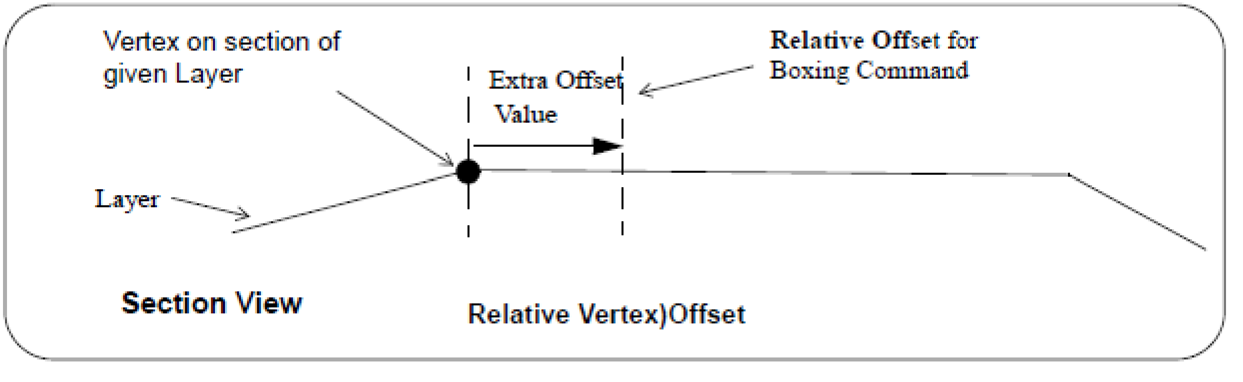
Adding a given value to an actual Offset of the given Layer.



(b) Vertex on section

Relative to a vertex on the Design or Boxing Layer - relative vertex offset

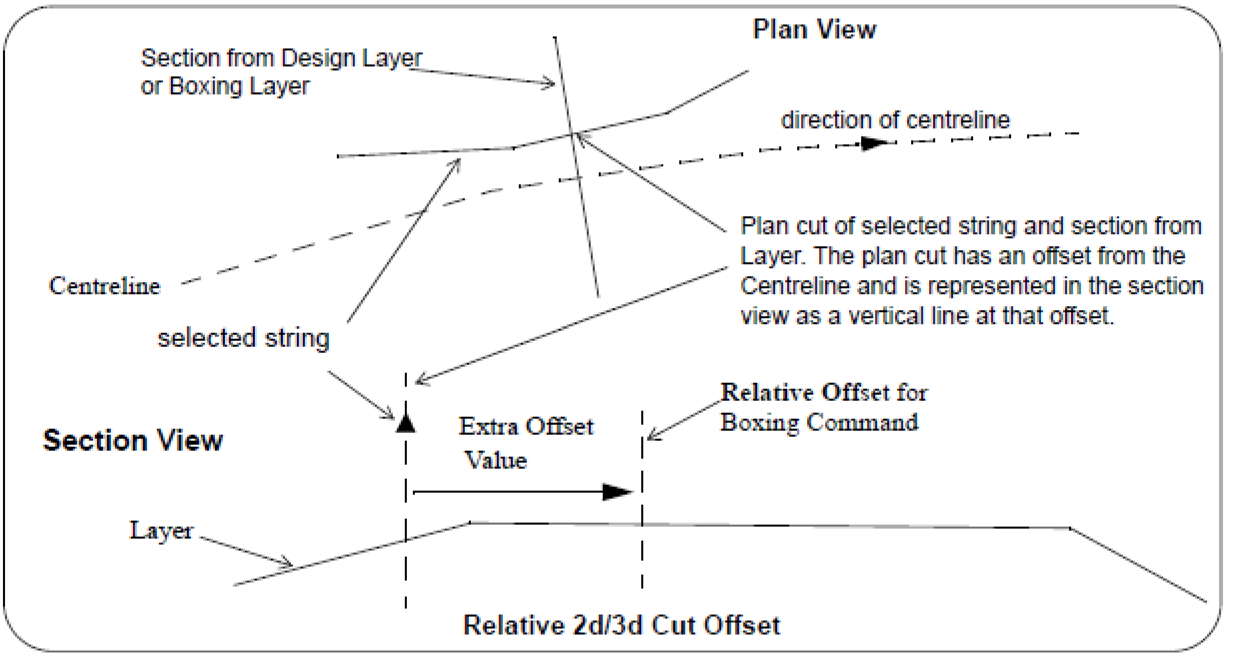
The offset of the given vertex is calculated and then a user given value added to it to the give the *relative vertex offset*.



(c) 2d cut of string

Relative to ***2d cut*** of a selected string with a given Layer.

First calculate the offset on the selected Layer of the plan cut of a selected string (2d cut). A user given value is then added to it to the give the *relative 2d offset*.

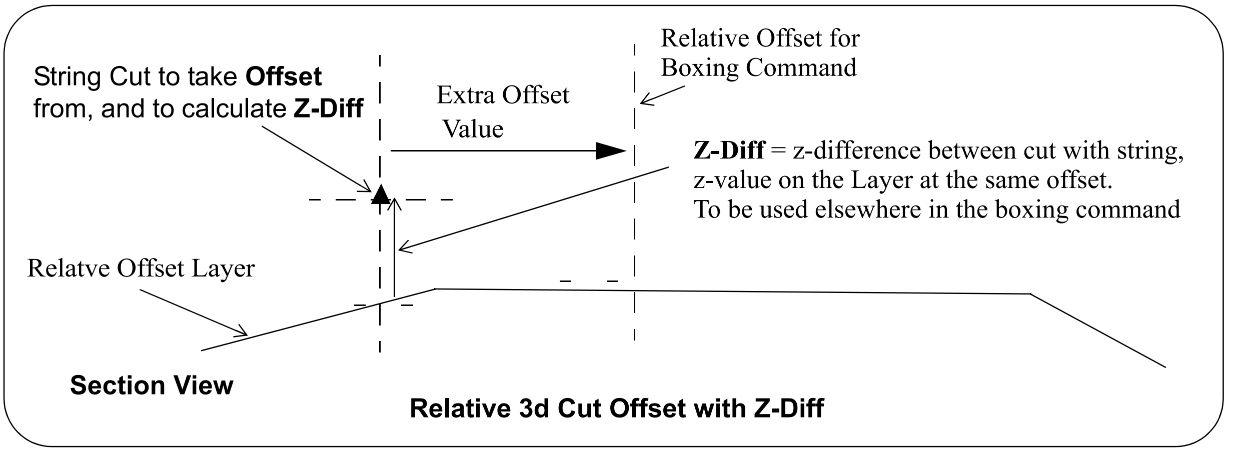


(d) Cut of string

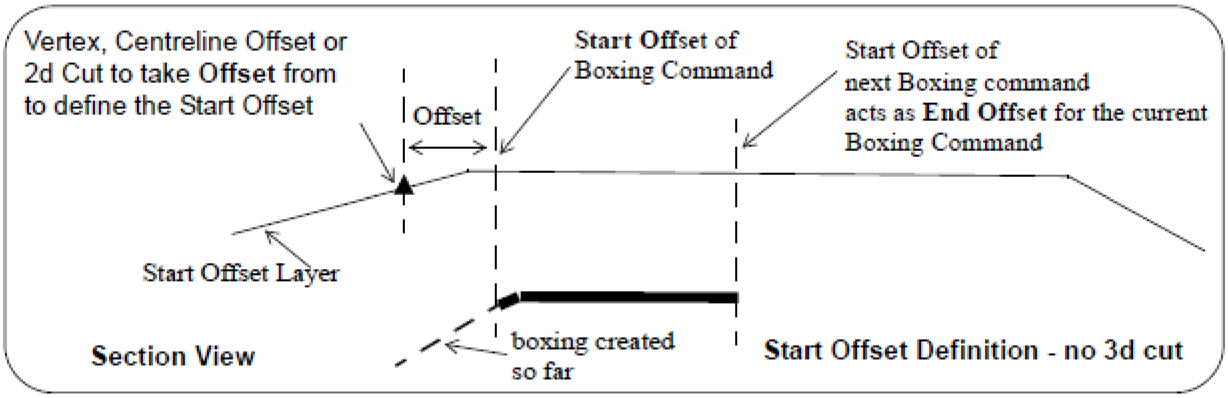
Relative to ***3d cut*** of a selected string

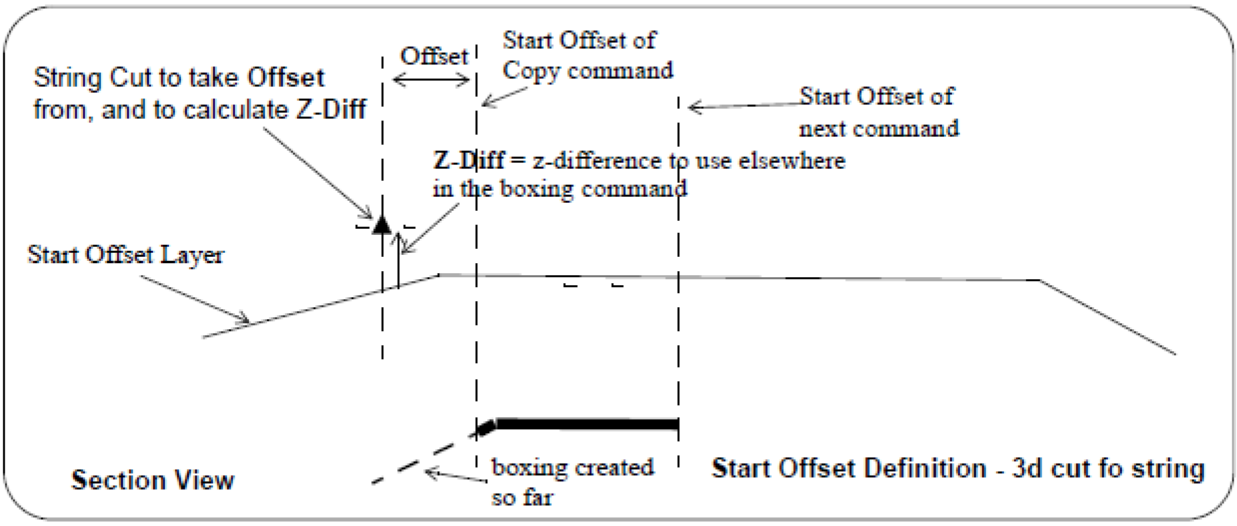
This is actually the same in *offset* value as the **2d** cut of a selected string but some boxing commands also take the ***3d cut*** of the selected string and use the **height difference** of the cut of the string and the height on the Layer elsewhere in the command. This is referred to as the Z-Diff of the 3d cut.

Note that some strings only have a horizontal (plan) definition and have no z-values at all. In that case, there is no 3d cut but there can be a 2d cut (plan cut). For example, a string with only null heights or a Super Alignment with no vertical geometry.



So for the case where Relative Offset is the ***Start Offset***:



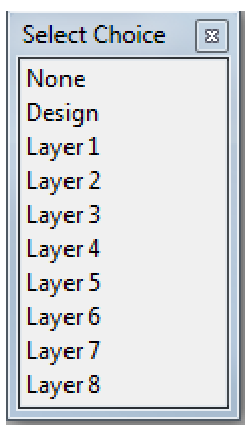


On a Boxing Command panel, the fields and buttons where a Relative Offset is required have the following functions:

Field Description Type Defaults Pop-Up

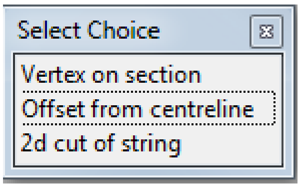
Relative Offset (e.g. Start Offset, Xfall Offset etc)

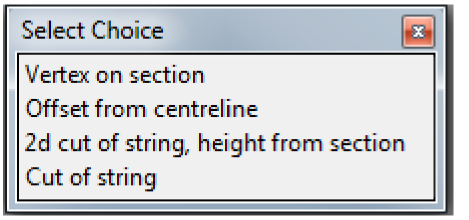
Layer - after Type choice box



Layer to use with **Offset** *to defined the Relative Offset.*

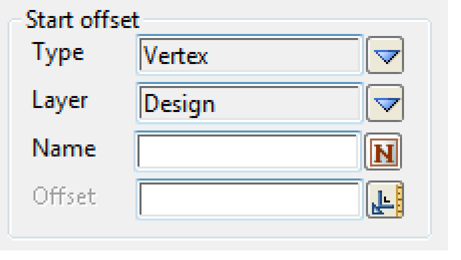
Type choice box





If ***Vertex on section****:*

this is the Relative (Vertex) Offset



Name name box

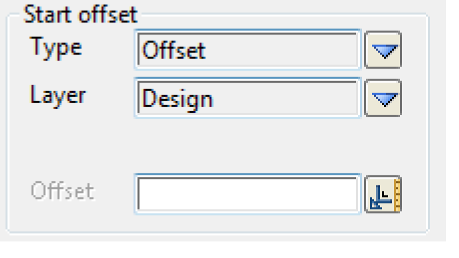
name of the vertex on the **Layer** *to use with* **Offset** *to define the Start Offset.*

Offset real box

this value is added to the offset of the vertex given in the **Name** *field to define the Relative Offset.* **Offset** *can be positive or negative as long as the final Start Offset is still* ***on*** **Layer***.*

If ***Offset from centreline:***

this is the Relative (Centreline) Offset

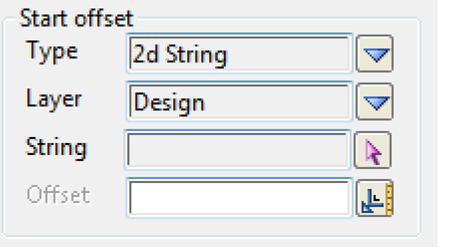


Offset real box

an actual offset from centreline (actually just the chainage of the section). **Offset** *can be positive or negative as long as it is still on* **Layer***.*

If ***2d cut of string:***

this is the Relative (2d Cut) Offset



String string select

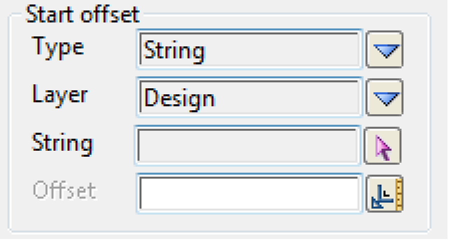
the selected string needs to cut the **Layer** *in plan wherever the boxing is being applied.*

Offset real box

the Relative Offset is the offset of the plan cut of the selected string with the **Layer***, plus the* **Offset** *value.* **Offset** *can be positive or negative but the final position Relative Offset must still be on* **Layer***.*

*If **Cut of string:***

this is the Relative (3d Cut) Offset



String string select

the selected string needs to cut the sections of the selected **Layer** *in plan wherever the boxing is being applied.*

Offset real box 0

The Relative Offset is the offset of the plan cut of the selected string with the section, plus the **Offset** *value.* **Offset** *can be positive or negative but the final position must still be on* **Layer***.*

Z-Diff *is the difference between the z-value of the cut string, and the z-value on the* **Layer** *at the same plan position as the cut with the string. The use of Z-Diff depends on the boxing command.*

21.8 Full Definition of Template Decisions

12d Model supports templates with an unlimited number of fixed links which are in the **Fixed** table of the **Temple** definition and they are processed first.

At the end of the fixed links,***12d Model*** checks to see if a **Decisions** table exists.

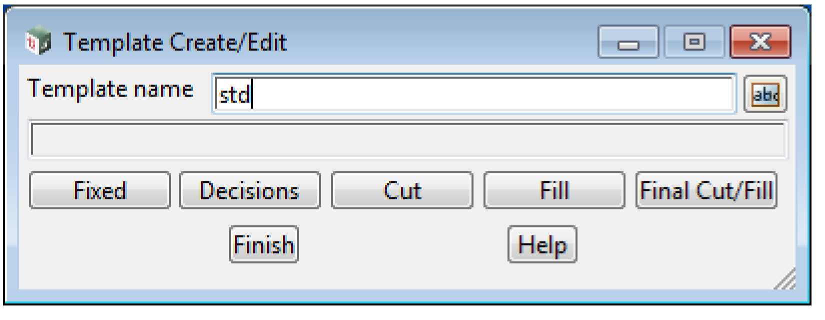
If a **Decisions** table exists, then it is used and the Decision commands processed.

If a ***Continue Cut/Fill*** command is reached in the decision command processing, then this stops the Decision command processing, and the processing moves onto the **Cut**, **Fill** and **Final Cut/Fill** tables.

If no ***Continue Cut/Fill*** command is reached in the decisions, then the processing stops after the decision commands are exhausted and the **Cut**, **Fill** and **Final Cut/Fill** tables ***ARE NOT USED***.

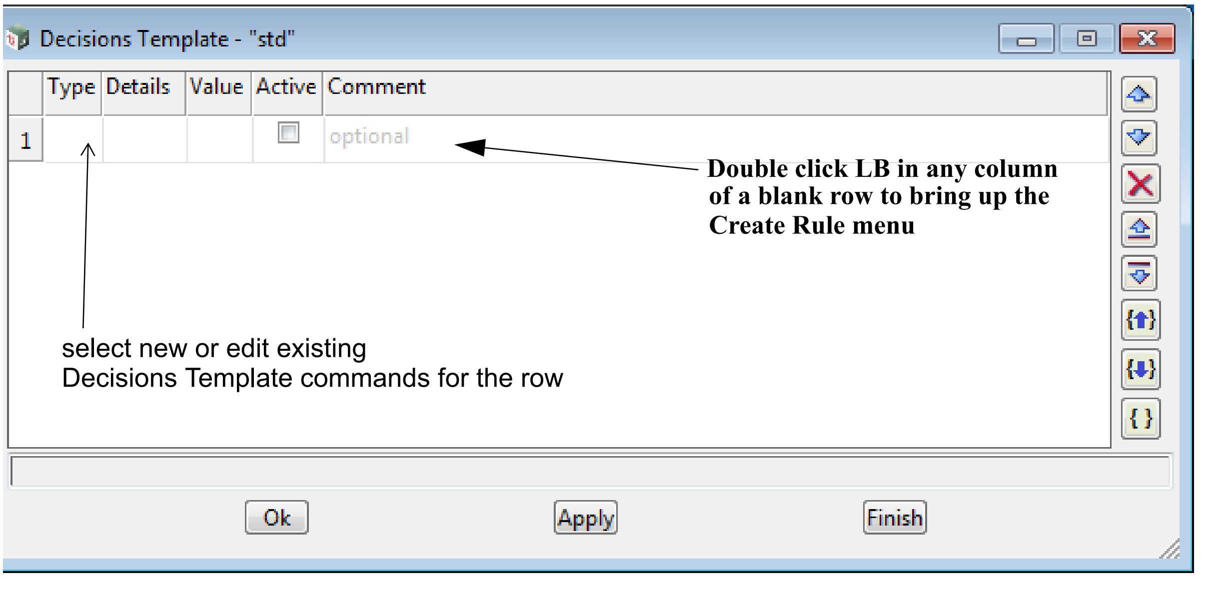
If **no Decisions** table exists, then the **Cut**, **Fill** and **Final Cut/Fill** tables are used after the **Fixed** table.

Templates are created and edited using the **Templates=>Create/edit** option which brings up the **Template Create/Edit** panel.



The description of the fields and the buttons **Fixed**, **Cut**, **Fill**, **Final Cut/Fill** in this panel are given in the section 20.2.1 Templates.

Selecting **Decisions** on the **Template Create/Edit** panel brings up the **Decisions Template** panel:



The **Decisions Template** panel consists of a grid with of rows (or lines) with **commands** in them, and an **OK** or **Apply** button to record the results.

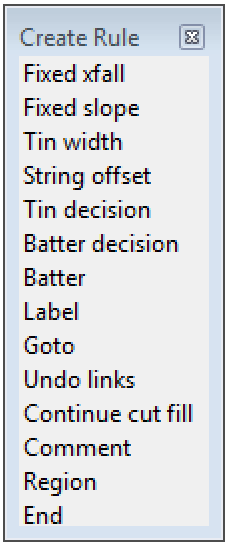
Decisions Templates Grid

For information on the general operation of a grid including the icons on the right hand side, see 4.19.6 Grids in Panels.

The ***commands*** *in the* **Decisions Template** *grid form a simple language where the commands are processed sequentially unless control is passed to a labelled line from where sequential processing continues, or the processing is terminated by an* ***end*** *command. For a description of all the Decisions Template Commands, see* 21.8.1 Decisions Template Commands.

When the Grid Row is Empty

If the row of the grid is ***empty****, clicking LB in the empty row will bring up the* **Create Rule** *menu which contains all the available decisions commands.Note this may involve two clicks - one to highlight a column in a row and the second click to bring up the* **Create Rule** *menu.*

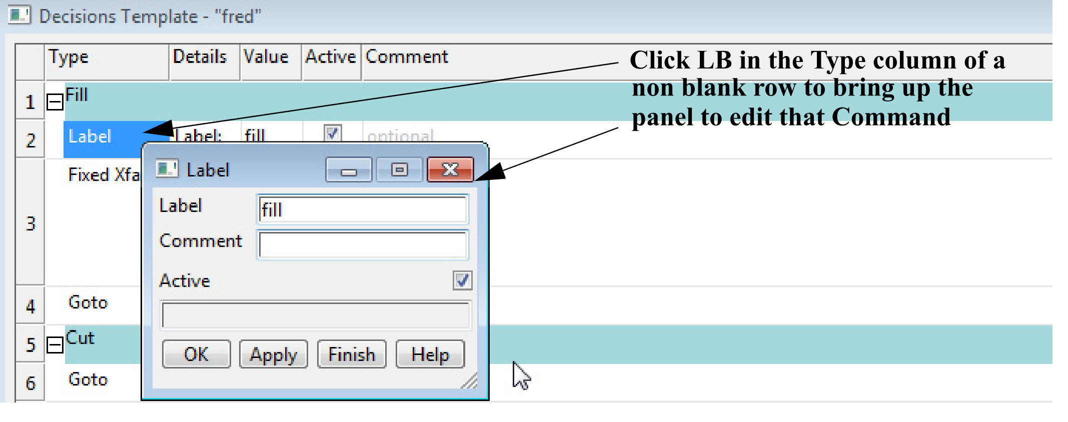


Selecting a menu item will bring up an associated panel which displays the information required for the Decisions Template command. When the panel is filled in and **OK** or **Apply** *selected, the panel information is written out to the row of the grid and is known as a* **decisions template command**.

When the Grid Row is Not Empty

If the row of the grid is ***not empty*** *(and hence filled with a* **Decisions Template command***) then clicking LB in the cell:*

(a) ***Type*** *will bring up the associated panel for the Decisions Template command.*



For a description of all the Decisions Template Commands, see 21.8.1 Decisions Template Commands.

(b) ***Active*** *will toggle the* ***tick/not ticked***

(c) ***Comment*** *will allow text to be typed into the cell*

Note *that clicking in the* ***Type***, ***Start chainage***, ***End chainage***, ***Interval***, ***Extra Start***, ***Extra End***, ***Active*** *and* ***Comment*** *cells may involve two clicks - one to highlight the cell in a row and the second click to edit or bring up the panel for the cell.*

Buttons at Bottom

OK button

OK *stores the values in the fields and removes the panel.*

Apply button

Apply *stores the values in the grid and leaves the panel on the screen.*

If the template is written out using **File I/O=>Templates output** or **Templates=>Utilities=>Output**, the Decisions Template commands will be written out in the order that they are displayed in the grid.

Please continue to the next section 21.8.1 Decisions Template Commands.

21.8.1 Decisions Template Commands

The **commands** in the **Decisions Template** grid form a simple language where the commands are processed sequentially unless control is passed to a labelled line from where sequential processing continues, or the processing is terminated by an **end** command.

The Decisions Template Commands can:

(a) create a fixed link - **fixed xfall** and **fixed slope**

(b) create a link of fixed width but with a slope automatically selected so that the link reaches a given tin in the width - **tin width**

(c) create a link that goes within an offset of a given string **string offset**

(d) test for being between two depths below a tin, and transfer control if true - **tin decision**

(e) test that a link would get within a given depth from a tin at a given offset, and transfer control if true - **batter decision**

(f) create a link of set slope that batters to a depth below a tin at a given offset - **batter**

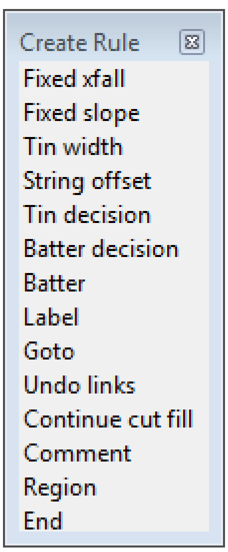
(g) create a label which can have processing passed to - **label**

(h) transfer processing to a given label -**goto**

(i) end the processing of commands - **end**

The **Decisions Commands** are selected from the **Create Rule** menu that is displayed when clicking in a cell in the **Type** column of the **Decisions Template** panel.

The **Create Rule** menu is



fixed width and xfall
fixed width and slope
fixed width, up to a tin
go to an offset from a string
test depth below a tin
test if a batter cuts a tin
batter to a tin
create a label
goto a label
undo links
stop processing decisions, and continue with the cut/fill tables
add comment line
create a region row
terminate the decisions

For a description of each command, see

Fixed xfall	21.8.2 Template Decisions - Fixed Xfall
Fixed slope	21.8.3 Template Decisions - Fixed Slope
Tin width	21.8.4 Template Decisions - Tin Width
String offset	21.8.5 Template Decisions - String Offset
Tin decision	21.8.6 Template Decisions - Tin Decision
Batter decision	21.8.7 Template Decisions - Batter Decision
Batter	21.8.8 Template Decisions - Batter
Label	21.8.9 Template Decisions - Label
Goto	21.8.10 Template Decisions - Goto
Undo links	21.8.11 Template Decisions - Undo

Continue cut fill	21.8.12 Template Decisions - Continue Cut Fill
Comment	21.8.13 Template Decisions - Comment.
Region	21.8.14 Template Decisions - Region.
End	21.8.15 Template Decisions - End

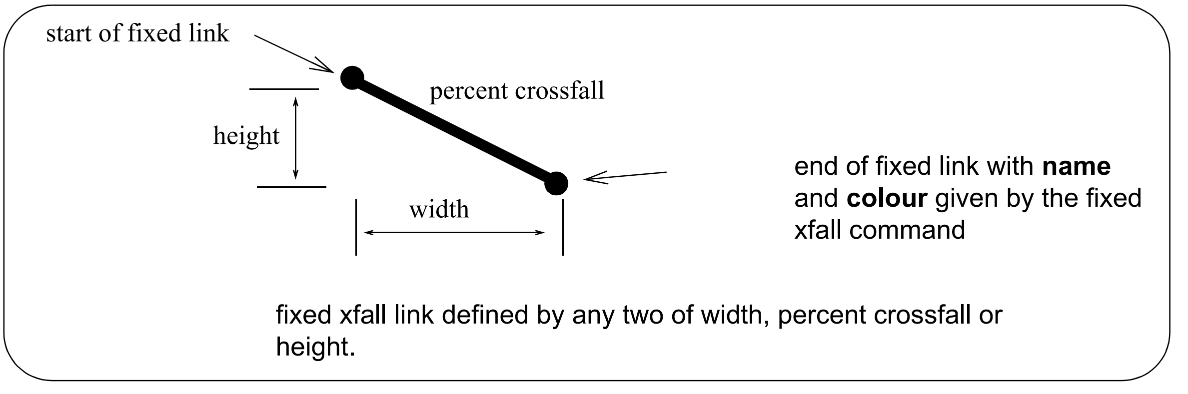
For examples of a using decisions, go to the section 21.8.17 Decisions Examples.

Please continue to the next section 21.8.2 Template Decisions - Fixed Xfall.

21.8.2 Template Decisions - Fixed Xfall

Selecting **Fixed xfall** brings up the **Fixed Xfall** panel which is used to constructed the command for a fixed link with the possibility of using cross fall (xfall). The fixed link is defined by specifying values for two of the three fields width, height and x-fall.

Note - if defining a slope rather than a xfall is required, use **Fixed Slope** (see 21.8.3 Template Decisions - Fixed Slope).



The format of the Fixed Xfall command in the panel is

Fixed Xfall Width *value* Height *value* XFall *value* Name *text* Colour *colour* // *comment*

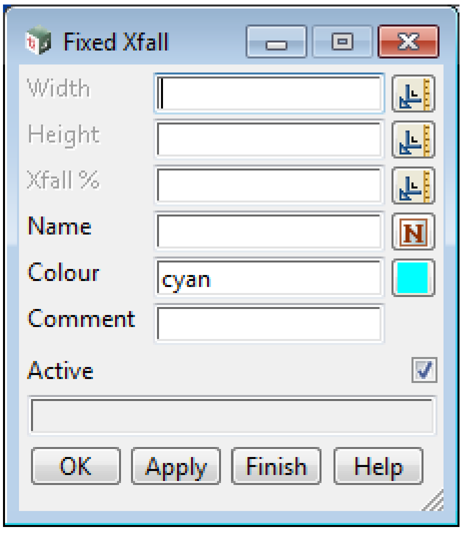
where only two of the three commands Width, Height and Xfall are used.

The format of the command typed into the template file is

Fixed_Xfall *width_value* *height_value* *xfall_value* *name* *colour* // **comment**

where one of *width_value*, *height_value* or *xfall_value* is the key word *unknown.*

The panel brought up when **Fixed xfall** is selected from the **Create Rule** menu is:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
Width	input		

the width for the link.

Height input

the height for the link.

Xfall % input

the x-fall, in percent cross-fall, of the link. Positive is up and negative down.

Name input

the name to be used for the created point and string.

Colour input cyan available colours

the colour to be used for the created string

Comment, **Active, OK and Apply**

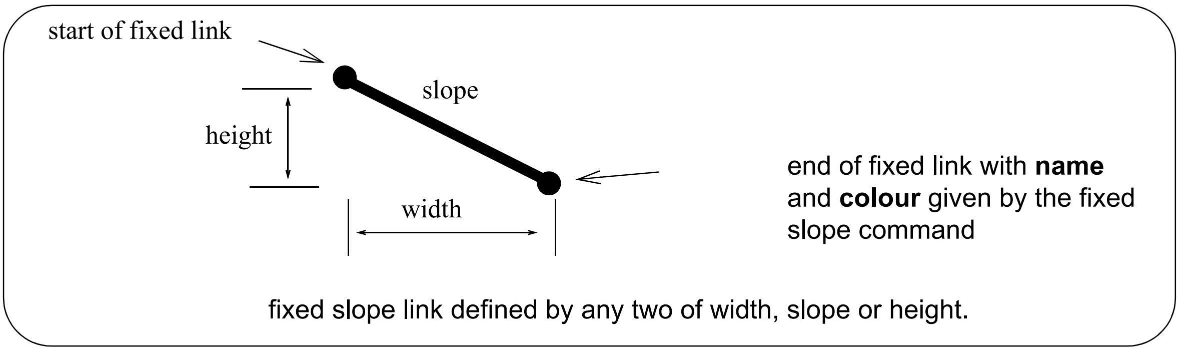
For information on these panel fields, see 21.8.16 Common Fields in Template Decisions Panels.

Continue to the next section 21.8.3 Template Decisions - Fixed Slope or return to 21.8 Full Definition of Template Decisions.

21.8.3 Template Decisions - Fixed Slope

Selecting **Fixed slope** brings up the **Fixed Slope** panel which is used to constructed a fixed link. The fixed link is defined by specifying values for two of the three fields width, height and slope.

Note - if defining a cross fall (xfall) rather than a slope is required, use **Fixed Xfall** (see 21.8.2 Template Decisions - Fixed Xfall).



The format of the Fixed Slope command in the panel is

Fixed Slope Width *value* Height *value* Slope *value* Name *text* Colour *colour // comment*

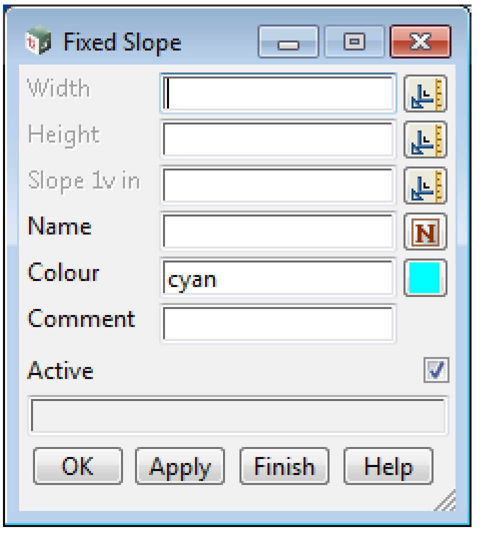
where only two of the three commands Width, Height and Slope are used.

The format of the command typed into the template file is

Fixed_Slope *width_value height_value slope_value name colour // comment*

where one of *width_value*, *height_value* or *slope_value* is the key word *unknown*

The panel brought up when **Fixed slope** is selected from the **Create Rule** menu is:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
Width	input		

the width for the link.

Height	input		

the height for the link.

Slope 1v in	input		

the slope, in 1v in, of the link. Positive is up and negative down.

Name	input		

the name to be used for the created point and string.

Colour	input	cyan	available colours

the colour to be used for the created string

Comment, **Active, OK and Apply**

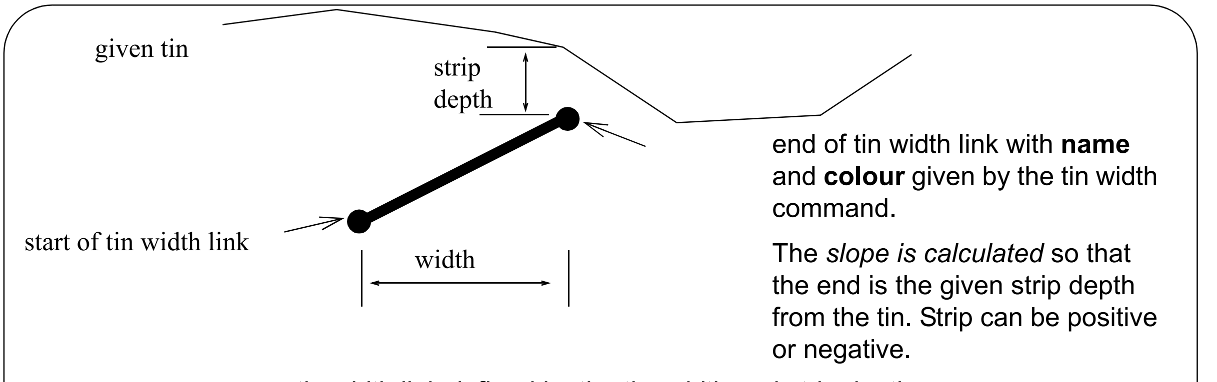
For information on these panel fields, see 21.8.16 Common Fields in Template Decisions Panels.

Continue to the next section 21.8.4 Template Decisions - Tin Width or return to 21.8 Full Definition of Template Decisions.

21.8.4 Template Decisions - Tin Width

Selecting **Tin width** brings up the **Tin Width** panel which is used to construct a link which has a ***given width*** and stops at the depth **strip** below the tin. Control then passes to the next line of the table.

strip can be positive (end point is below the tin) or negative (end point is above the tin).



tin width link defined by the tin, width and strip depth.

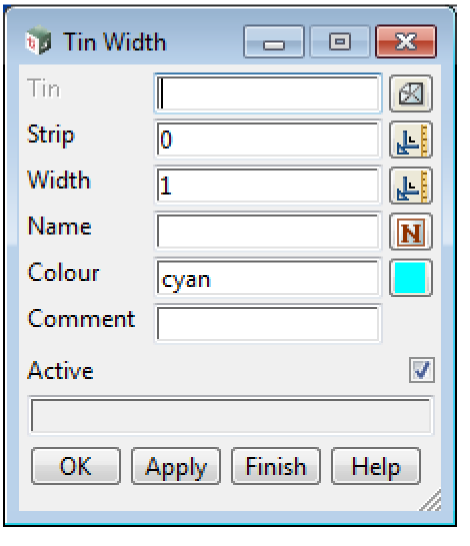
The format of the Tin Width command in the panel is

Tin Width *tin_name* Strip *value* Width *value* Name *text* Colour *colour* // *comment*

The format of the command typed into the template file is

Tin_Width *tin_name* *strip_value* *width_value* *name* *colour* // *comment*

The panel brought up when **Tin width** is selected from the **Create Rule** menu is:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
Tin	input		available tins
tin to batter to.			
Strip	input	0	

distance below the tin to stop at (strip depth). ***strip*** *can be positive (point is below the tin) or negative (point is above the tin).*

Width input 1

width of the link

Name input

the name to be used for the created point and string.

Colour input cyan available colours

the colour to be used for the created string

Comment, Active, OK and Apply

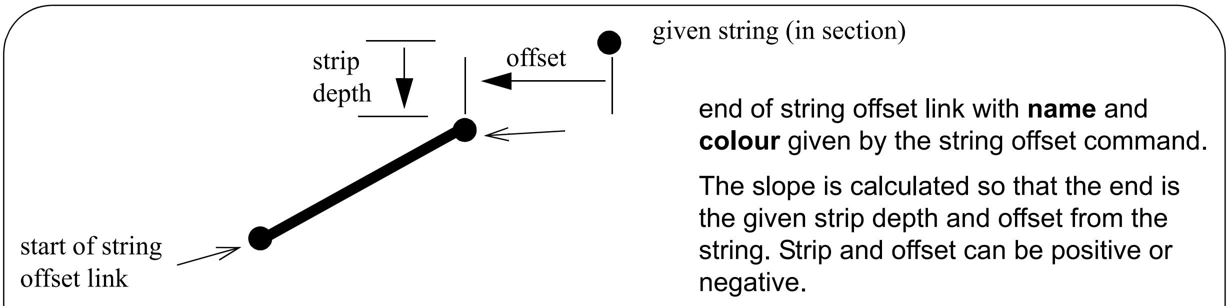
For information on these panel fields, see 21.8.16 Common Fields in Template Decisions Panels.

Continue to the next section 21.8.5 Template Decisions - String Offset or return to 21.8 Full Definition of Template Decisions.

21.8.5 Template Decisions - String Offset

Selecting **String offset** brings up the **String Offset** panel which is used to construct a link which goes to a ***given offset from string*** and stops at the depth **strip *below the string***. Control then passes to the next line of the table.

offset and **strip** can be positive or negative.



string offset link defined by the string, offset and strip depth.

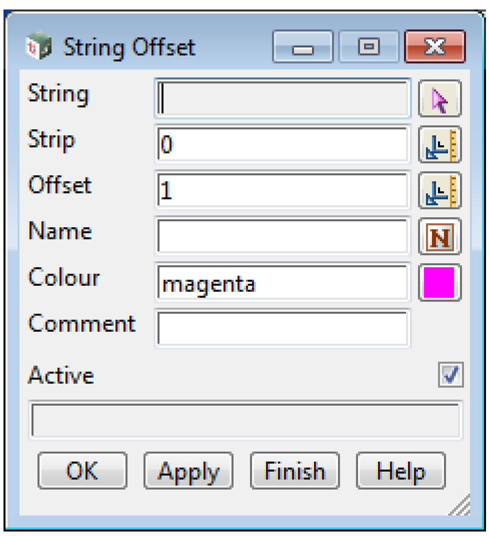
The format of the String Offset command in the panel is

> **String Offset** *string_name* Strip *value* Offset *value* Name *text* Colour *colour // comment*

The format of the command typed into the template file is

> **String_Offset** *string_name strip_value offset_value name colour // comment*

The panel brought up when **String offset** is selected from the **Create Rule** menu is:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
String	string select		
string to go to.			
Strip	input	0	

distance below the string to stop at.

Offset	input	1	

offset from the string to stop at.

Name	input		

the name to be used for the created point and string.

Colour	input	cyan	available colours

the colour to be used for the created string

Comment, **Active, OK and Apply**

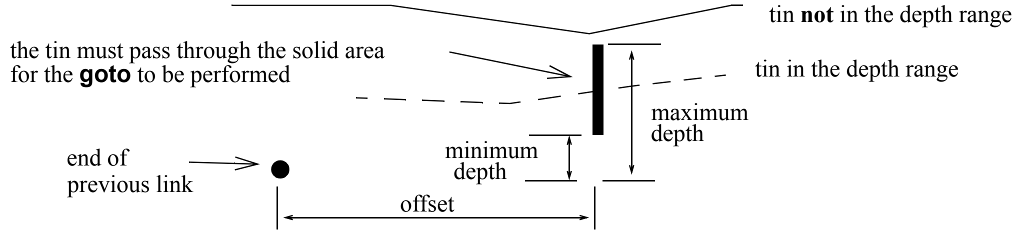
For information on these panel fields, see 21.8.16 Common Fields in Template Decisions Panels.

Continue to the next section 21.8.6 Template Decisions - Tin Decision or return to 21.8 Full Definition of Template Decisions.

21.8.6 Template Decisions - Tin Decision

Selecting **Tin decision** brings up the **Tin Decision** panel which tests to see if the depth from the end of the previous link, offset by the amount given in the **offset** field, is between the two values given in the **minimum depth** and **maximum depth** fields. If the depth is between the values, then control is transferred to the line with the label given by the **goto** field. Otherwise, control passes to the next line of the table.

offset, **minimum** and **maximum** depth can be positive or negative.



tin decision given by offset and minimum and maximum depths.

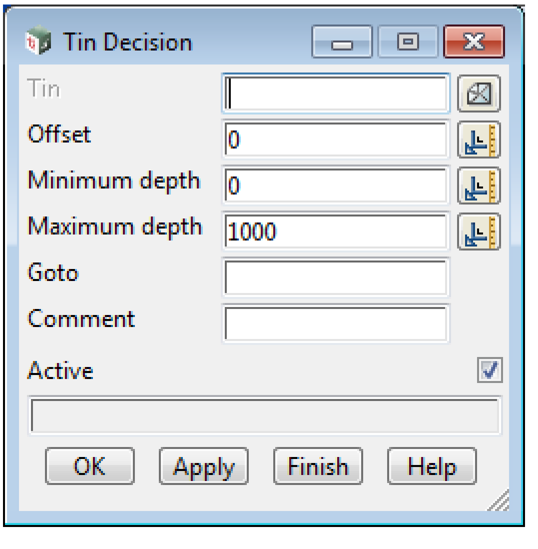
The format of the Tin Decision command in the panel is

Tin Decision *tin_name* Offset *value* Min *value* Max *value* Goto *label // comment*

The format of the command typed into the template file is

Tin_Decision *tin_name offset_value min_value max_value goto_label // comment*

The panel brought up when **Tin decision** is selected from the **Create Rule** menu is:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
Tin	tin box		available tins

the tin to calculate the depth to.

Field Description	Type	Default	Pop-Up
Offset	input	0	

the depth is calculate at an offset distance of **offset** *from the end of the previous link.*

Minimum depth input 0

if the depth is between the minimum and maximum depth, then control is passed to the line with the label given in the goto field, otherwise control passes onto the next line of the table.

Maximum depth input 1000

see previous field.

Goto input

label to go to if the depth is between the minimum and maximum depths.

Comment, **Active, OK and Apply**

For information on these panel fields, see 21.8.16 Common Fields in Template Decisions Panels.

Continue to the next section 21.8.7 Template Decisions - Batter Decision or return to 21.8 Full Definition of Template Decisions.

21.8.7 Template Decisions - Batter Decision

Selecting **Batter decision** brings up the **Batter Decision** panel which is used to test whether a given link comes within

(a) a **strip** depth of a tin

(b) an offset distance from a tin.

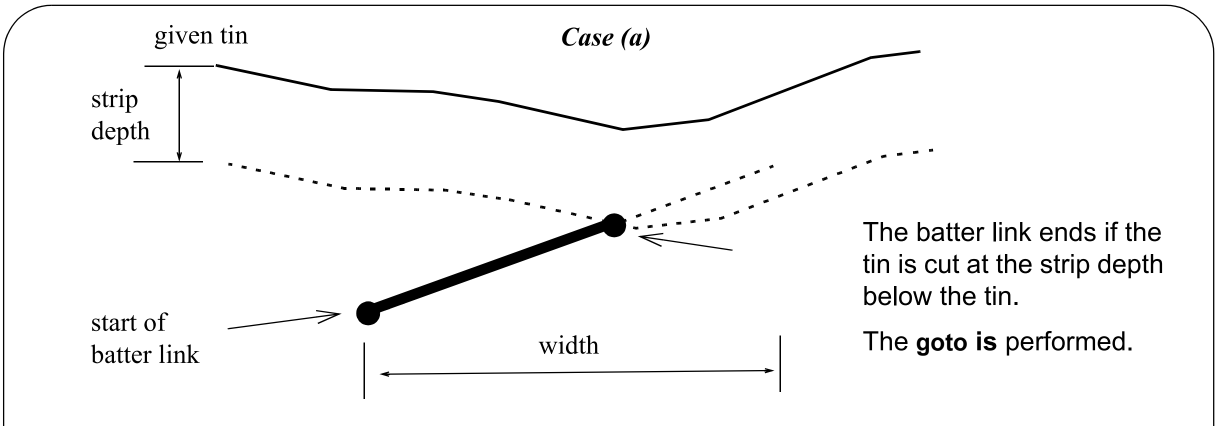
(c) a strip depth of a tin calculated at a given offset from the link.

If the test is satisfied, then control is transferred via a goto, otherwise control continues onto the next line in the table.

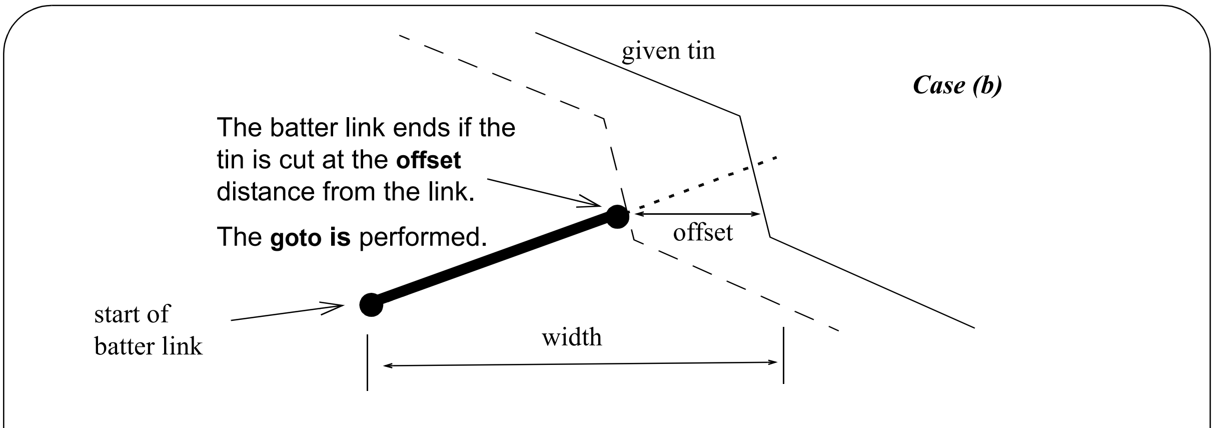
The **test link** is defined by specifying values for two of the three fields width, height and slope.

Notes

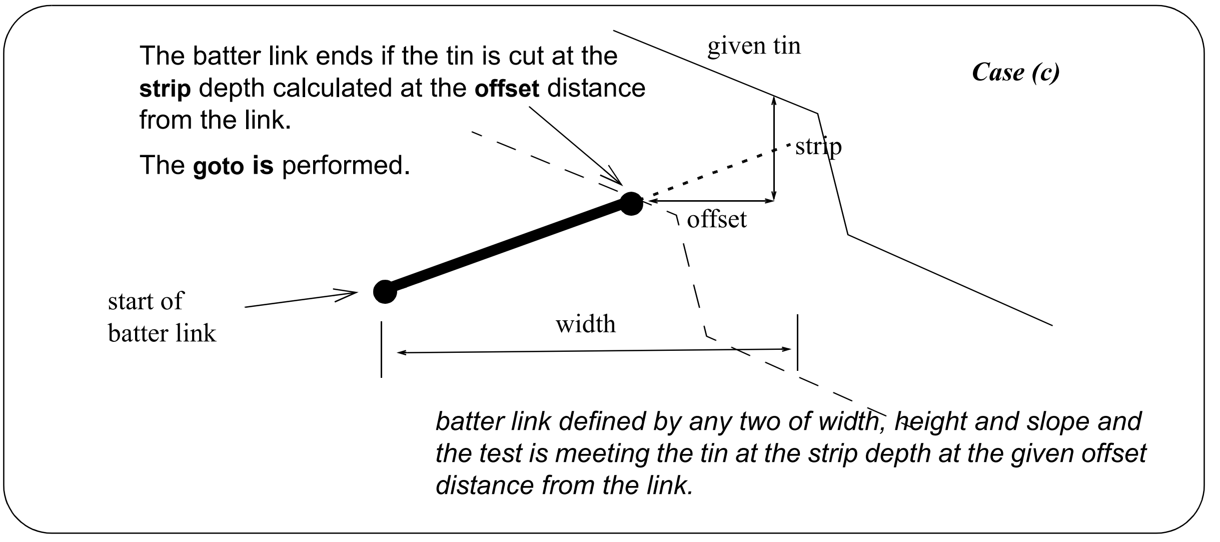
1. no link is created, just the test is performed
2. cases (a) and (b) are just special cases of (c)
3. the batter decision is mainly used to test if a batter will stop without performing the batter.

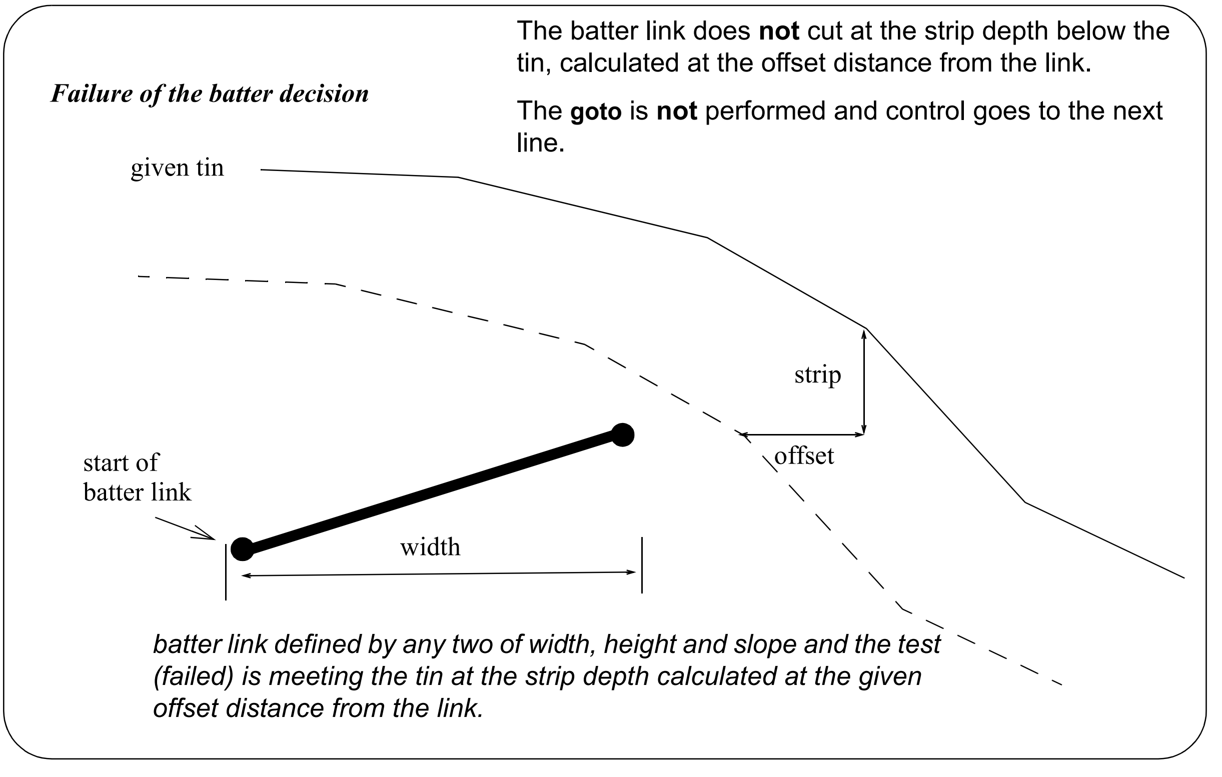


batter link defined by any two of width, height and slope and the test is meeting the tin at the strip depth.



batter link defined by any two of width, height and slope and the test is meeting the tin at the offset distance.





The format of the Batter Decision command in the panel is

Batter Decision *tin_name* Strip *value* Offset *value* Width *value* Height *value* Slope *value* Goto *label* // *comment*

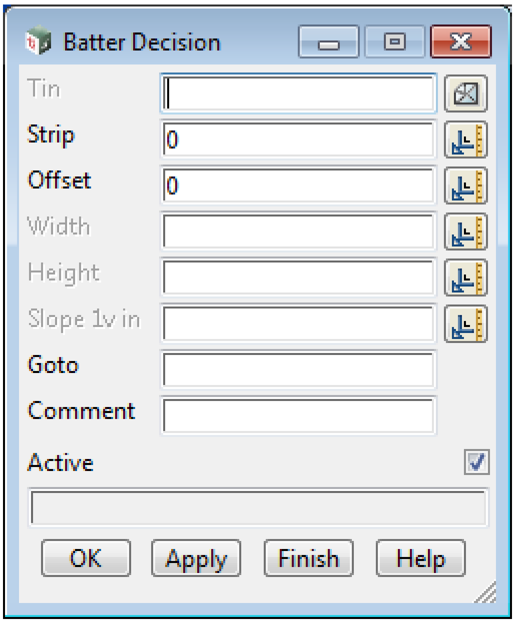
where only two of the three commands Width, Height and Slope are used.

The format of the command typed into the template file is

Batter_Decision *tin_name strip_value offset_value width_value height_value slope _value* goto_*label* // *comment*

where one of *width_value*, *height_value* or *slope_value* is the key word *unknown*

The panel brought up when **Batter decision** is selected from the **Create Rule** menu is:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
Tin	tin box		available tins

tin to test the batter to.

Strip	input	0	

distance below the tin to stop at.

Offset	input	0	

offset distance from the link to check strip depth

Width input

the width for the link.

Height input

the height for the link.

Slope 1v in input

the slope, in 1v in, of the link. Positive is up and negative down.

Goto input

label to goto if the test link comes within the **strip** *depth of the tin at the given* **offset** *distance.*

Comment, **Active, OK and Apply**

For information on these panel fields, see 21.8.16 Common Fields in Template Decisions Panels.

Continue to the next section 21.8.8 Template Decisions - Batter or return to 21.8 Full Definition of Template Decisions.

21.8.8 Template Decisions - Batter

Selecting **Batter** brings up the **Batter** panel which is used to construct a link which stops if it comes within

(a) a **strip** depth of a tin

(b) an offset distance from a tin.

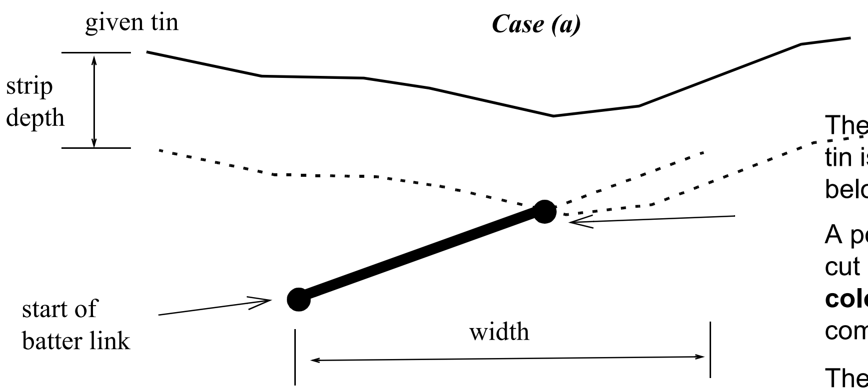
(c) a strip depth of a tin calculated at a given offset from the link.

If the link does stop, control is transferred via a goto, otherwise control continues onto the next line in the table.

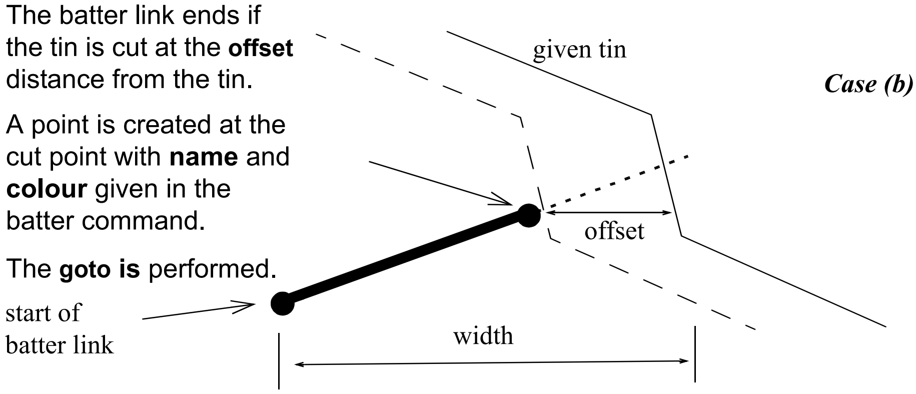
The **batter** link is defined by specifying values for two of the three fields width, height and slope.

Notes

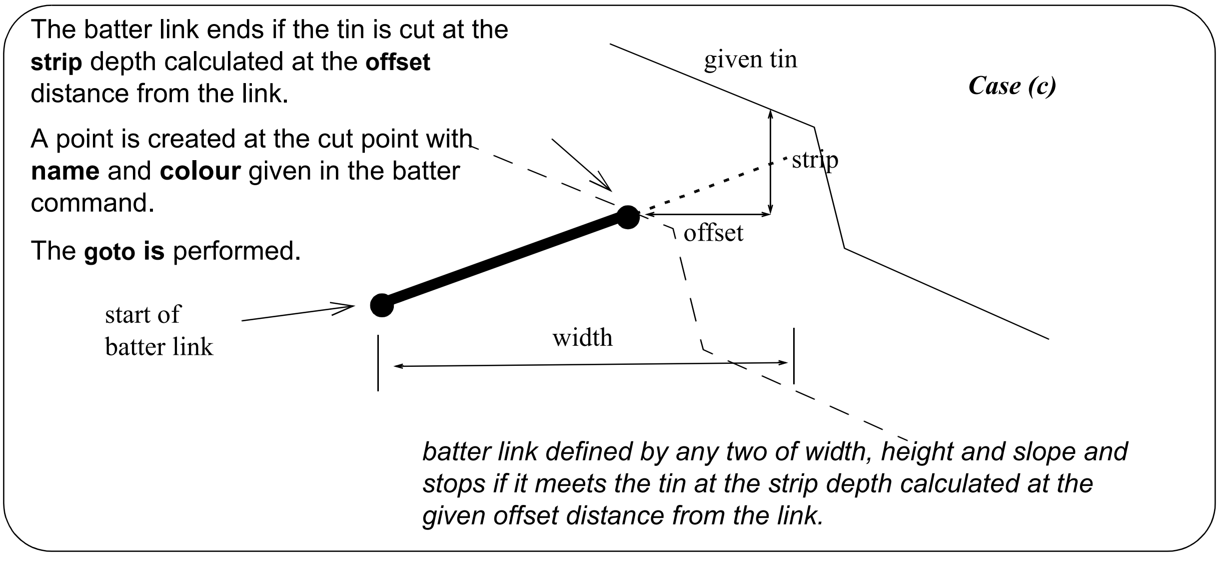
1. cases (a) and (b) are just special cases of (c)
2. strip and offset can be used to stop the link to allow for a fixed structure (such as a drain) to be inserted so that it ends up on the tin.

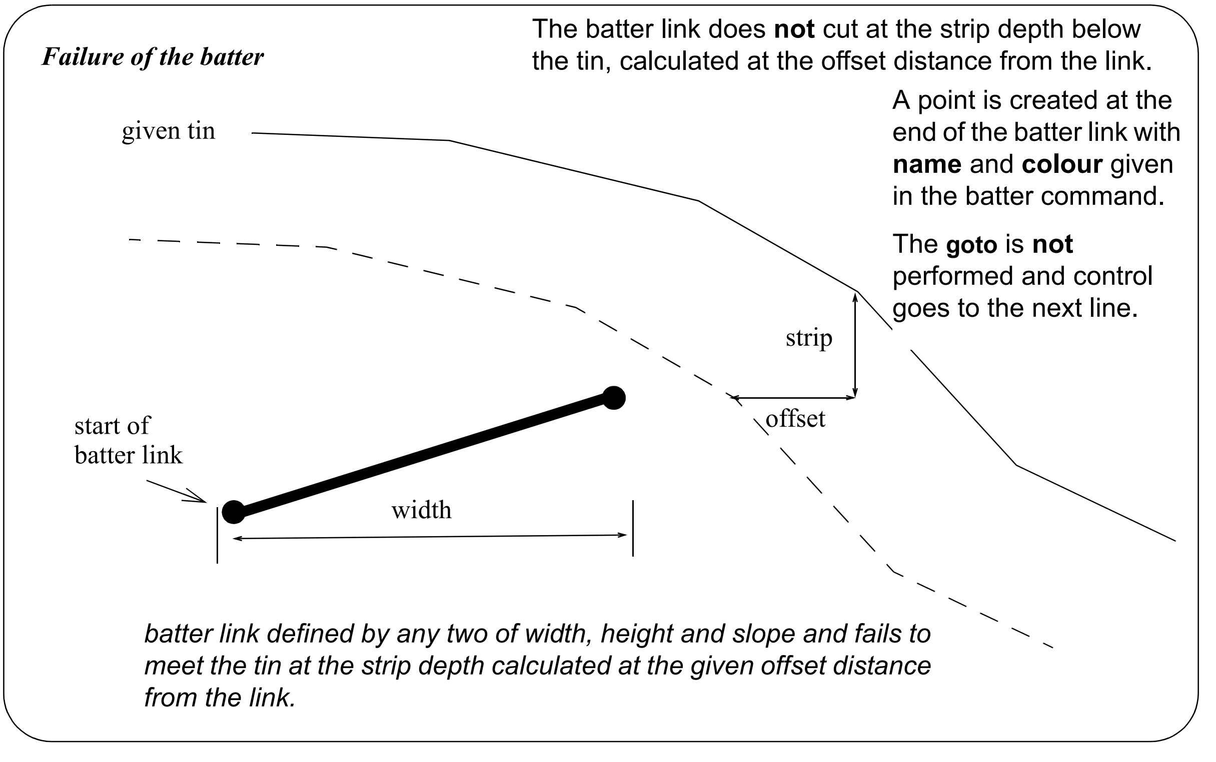


batter link defined by any two of width, height and slope and stops if it meets the tin at the strip depth.



batter link defined by any two of width, height and slope and stops if it meets the tin at the offset distance.





The format of the Batter command in the panel is

Batter *tin_name* Strip *value* Width *value* Height *value* Slope *value* Name *text* Colour *colour* Goto *label* // *comment*

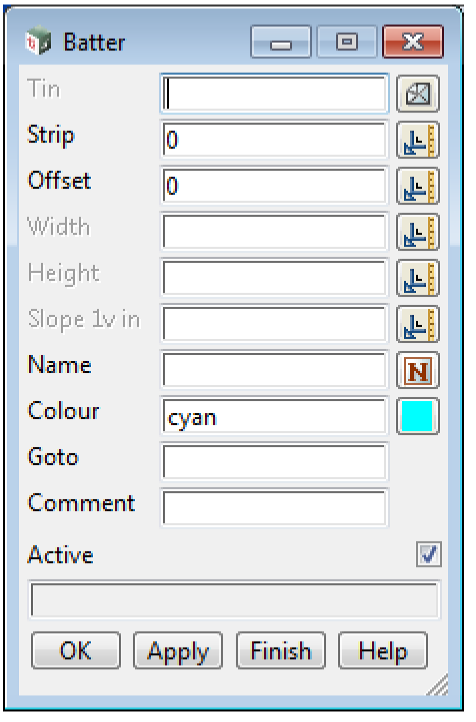
where only two of the three commands Width, Height and Slope are used.

The format of the command typed into the template file is

Batter *tin_name strip_value width_value height_value slope_value* name colour goto_*label* // *comment*

where one of *width_value*, *height_value* or *slope_value* is the key word *unknown*

The panel brought up when **Batter** is selected from the **Create Rule** menu is:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
Tin	tin box		available tins
tin to batter to.			
Strip	input	0	
distance below the tin to stop at.			
Offset	input	0	
offset distance from the link to check strip depth			
Width	input		
the width for the link.			
Height	input		
the height for the link.			
Slope 1v in	input		
the slope, in 1v in, of the link. Positive is up and negative down.			
Name	input		
the name to be used for the created point and string.			
Colour	colour box	cyan	available colours
the colour to be used for the created string			
Goto	input		
label to goto if the batter link comes within the depth **strip** *of the tin.*			

Comment, **Active, OK and Apply**

For information on these panel fields, see 21.8.16 Common Fields in Template Decisions Panels.

Continue to the next section 21.8.9 Template Decisions - Label or return to 21.8 Full Definition of Template Decisions.

21.8.9 Template Decisions - Label

Selecting **Label** brings up the **Label** panel which is used to define a label for the line for which control can be passed to via a goto.

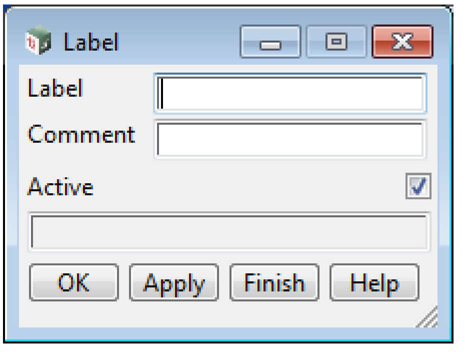
The format of the Label command in the panel is

Label *label_name* *// comment*

The format of the command typed into the template file is

Label *label_name* *// comment*

The panel brought up when **Label** is selected from the **Create Rule** menu is:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
Label	input		

name of the label for the line in the table.

Comment, **Active, OK and Apply**

For information on these panel fields, see 21.8.16 Common Fields in Template Decisions Panels.

Continue to the next section 21.8.10 Template Decisions - Goto or return to 21.8 Full Definition of Template Decisions.

21.8.10 Template Decisions - Goto

Selecting **Goto** brings up the **Goto** panel which is used to transfer control to the line with the label given in the **goto** field of the panel.

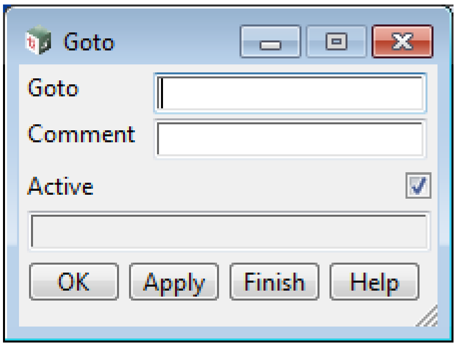
The format of the Goto command in the panel is

Goto *label_name* *// comment*

The format of the command typed into the template file is

Goto *label_name* *// comment*

The panel brought up when **Goto** is selected from the **Create Rule** menu is:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
Goto	input		

name of the label to transfer control to.

Comment, **Active, OK and Apply**

For information on these panel fields, see 21.8.16 Common Fields in Template Decisions Panels.

Continue to the next section 21.8.11 Template Decisions - Undo or return to 21.8 Full Definition of Template Decisions.

21.8.11 Template Decisions - Undo

Selecting **undo** brings up the **Undo Decision** panel which is used to delete a given number of points (and hence links) from the template being constructed.

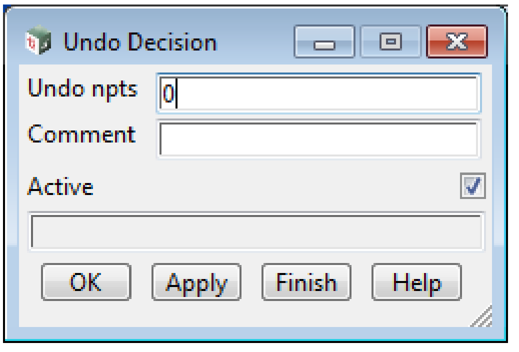
The format of the Label command in the panel is

Undo *number_of_points* *// comment*

The format of the command typed into the template file is

Undo *number_of_links* *// comment*

The panel brought up when **Undo** is selected from the **Create Rule** menu is:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up
Undo npts	input	0	

number of points to delete from the template, if the number is larger than the number of points in the template all points in the template are removed.

Comment, **Active, OK and Apply**

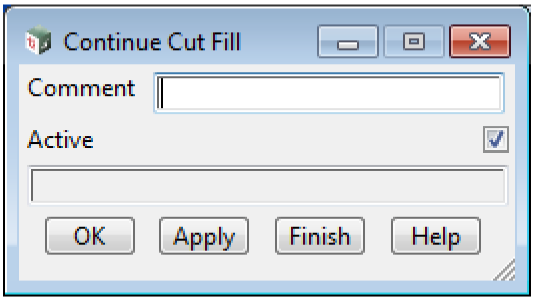
For information on these panel fields, see 21.8.16 Common Fields in Template Decisions Panels.

Continue to the next section 21.8.12 Template Decisions - Continue Cut Fill or return to 21.8 Full Definition of Template Decisions.

21.8.12 Template Decisions - Continue Cut Fill

If the ***Continue Cut Fill*** statement is reached when processing the decision commands, the decision section is terminated, and the processing continues to the Cut and Fill tables.

The panel brought up when **Continue Cut Fill** is selected from the **Create Rule** menu is:



Comment, **Active, OK and Apply**

For information on these panel fields, see 21.8.16 Common Fields in Template Decisions Panels.

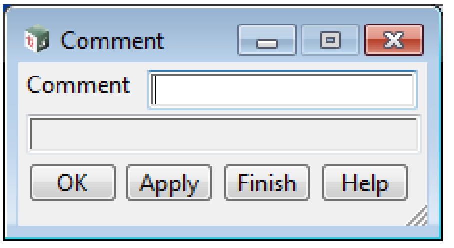
Clicking on ***OK*** or ***Apply*** inserts the ***Continue Cut Fill***, and any *Comment*, into the current line of the *Command* table.

Continue to the next section 21.8.13 Template Decisions - Comment or return to 21.8 Full Definition of Template Decisions.

21.8.13 Template Decisions - Comment

Selecting **Comment** brings up the **Comment** panel and inserts a comment into the line of the table. Comments are ignored by the decisions calculations.

The panel brought up when **Comment** is selected from the **Create Rule** menu is:



The fields and buttons used in this panel have the following functions.

Field Description Type Default Pop-Up

Comment, **Active, OK and Apply**

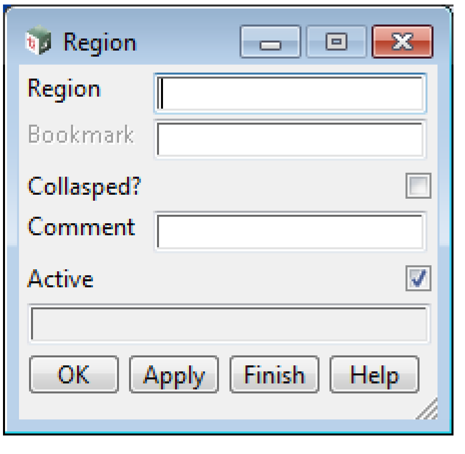
For information on these panel fields, see 21.8.16 Common Fields in Template Decisions Panels.

Continue to the next section 21.8.14 Template Decisions - Region or return to 21.8 Full Definition of Template Decisions.

21.8.14 Template Decisions - Region

The **Region** option inserts a **Region command** into the row.

Selecting **Region** brings up the **Decisions Region** panel



There are **Regions**, **Previous Region** and **Next Region** icons at the right hand side of the **Hinge Modifiers** panel. For information on using these icons and information on **Regions** in a grid, see 4.19.6 Grids in Panels.

Continue to the next section 21.8.15 Template Decisions - End or return to 21.8 Full Definition of Template Decisions.

21.8.15 Template Decisions - End

Selecting **End** brings up the **End** panel and inserts an end command into the line of the table. When control reaches an **end** command, the decisions calculations are terminated.

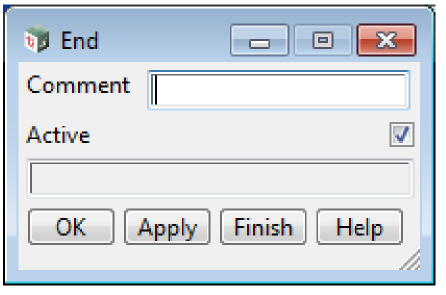
The format of the End command in the panel is

End *// comment*

The format of the command typed into the template file is

End *// comment*

The panel brought up when **End** is selected from the **Create Rule** menu is:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Default	Pop-Up

Comment, **Active, OK and Apply**

For information on these panel fields, see 21.8.16 Common Fields in Template Decisions Panels.

Continue to the next section 21.8.16 Common Fields in Template Decisions Panels or return to 21.8 Full Definition of Template Decisions.

21.8.16 Common Fields in Template Decisions Panels

All the ***Template Decisions*** panels have the common panel fields **Comments** and **Active**, and buttons **OK** and **Apply**.

Comment input

*user comment - for information purposes only. In the file, the comment text will be preceded by **//.***

Active tick box tick

*if **ticked**, use this modifier.*
*If **not ticked**, don't use this modifier.*

OK button

OK *stores command and the values in the fields and removes the panel.*

Apply button

Apply *stores command and the values in the fields and leaves the panel on the screen.*

Continue to the next section 21.8.17 Decisions Examples or return to 21.8 Full Definition of Template Decisions.

21.8.17 Decisions Examples

One Tin Example

```
//From: 12D Solutions
//
//Basic Decisional Template for cut in 9m depth, 5m depth, and less
//Note the Tin Used is "SURVEY"
//
//
template "std" {
  fixed {
    link 5 unknown -3 cyan "eb"
  }
  decisional {
// if at offset 1m it is below "SURVEY" then goto drain
    Tin_Decision      "SURVEY" 1 0 1000 "drain"

    Fixed_Xfall       1 unknown -3 "edge1" cyan
    Batter            "SURVEY" 0 100 unknown -3 "fill" cyan "done"

// label drain
// create a table drain
    Label             "drain"
    Fixed_Slope       1 -0.5 unknown "invert" cyan
    Fixed_Slope       1 0.5 unknown "edge" cyan

// now at end of table drain, check depth of 5, 9
    Tin_Decision      "SURVEY" 0 0 5 "smallestcut"
    Tin_Decision      "SURVEY" 0 0 9 "smallcut"

// deeper than 9m, batter to within 9m of surface
    Batter            "SURVEY" 9 100 unknown 0.75 "cut1" cyan "bench"

// label smallcut -
    Label             "smallcut"
    Batter            "SURVEY" 5 100 unknown 0.75 "small" cyan "bench2"

// label smallestcut -
    Label             "smallestcut"
    Batter            "SURVEY" 0 100 unknown 2 "smallcut" cyan "done"

// label bench
    Label             "bench"
    Fixed_Slope       3 -0.3 unknown "bench1" cyan
    Batter            "SURVEY" 5 100 unknown 1.5 "cut2" cyan "bench2"

// label bench2 -
    Label             "bench2"
    Fixed_Slope       3 -0.3 unknown "bench2" cyan
    Batter            "SURVEY" 0 100 unknown 2 "cut3" cyan "done"

// label done -

    Label             "done"
    End
  }
```

```
  final {
    cut_slope na fill_slope na search_distance 100
  }
}
```

Three Tins Example

```
//From: 4D Solutions
//Date: 11/07/97
//
// Example with three surfaces - rock, shale, terrain
// Assume that rock is below shale is below terrain

template "std" {
  fixed {
    link 3 unknown -3 cyan "kerb"
    link 1 unknown -4 magenta "shoulder"
    link 2 unknown -5 purple "verge"
  }
  decisional {
// test if in rock
    Tin_Decision  "rock" 0 0 1000 "cut_rock"

// not in rock, test if in shale
    Tin_Decision  "shale" 0 0 1000 "cut_shale"

// not in rock or shale, test if below or above terrain
    Tin_Decision  "terrain" 0 0 1000 "cut_terrain"

// fill terrain - loop on creating some benches
    Label    "fill_terrain"
    Batter   "terrain" 0 0 8 unknown -3 "f1" blue "alldone"
     Batter  "terrain" 0 0 1 unknown 0 "f2" "dark green" "alldone"
    Goto     "fill_terrain"

// cut rock
    Label              "cut_rock"
    Tin_Decision       "rock" 0 0 0.3 "cut_rock_done"
    Batter            "rock" 0.3 0 5 unknown 0.5 "r1" cyan "cut_rock_done"
    Batter            "rock" 0 0 2.5 unknown 0 "r2" yellow "cut_rock_done"
    Goto               "cut_rock"

// cut rock done - work on shale
    Label              "cut_rock_done"
    Tin_Decision       "shale" 0 0 1000 "cut_shale"
    Goto               "cut_shale_done"

// cut shale
    Label            "cut_shale"
    Tin_Decision     "shale" 0 0 0.6 "cut_shale_done"
    Batter         "shale" 0.6 0 6 unknown 1 "s1" magenta "cut_shale_done"
    Batter        "shale" 0 0 3 unknown 0 "s2" "dark red" "cut_shale_done"
    Goto               "cut_shale"
```

```
// cut shale done - work on terrain
    Label              "cut_shale_done"
    Tin_Decision       "terrain" 0 0 1000 "cut_terrain"
    Goto               "alldone"

// cut terrain
    Label              "cut_terrain"
    Batter             "terrain" 0 0 3 unknown 1.5 "t1" red "alldone"
    Batter             "terrain" 0 0 2 unknown 0 "t2" green "alldone"
    Goto               "cut_terrain"

// all done - end of decisional
    Label              "alldone"
  }
  final {
    cut_slope 1 fill_slope 2 search_distance 100
  }
}
template "headwall" {
  fixed {
    link 3 unknown -3 cyan "kerb"
    link 1 unknown -4 magenta "shoulder"
    link 2 unknown -5 purple "verge"
  }
  decisional {
    Tin_Decision       "" 0 0 1000 "done" // should never happen
    Batter             "" 0 0 10 -100 unknown "int" cyan "done"
    Label              "done"
  }
  final {
    cut_slope na fill_slope na search_distance 100
  }
}
```

Return to 21.8 Full Definition of Template Decisions.

21.9 Placing Parts for Super Alignments

Placing a super alignment consists of:

(a) defining the horizontal geometry which ultimately consisting of lines, arcs, transitions and tapers

(b) defining the vertical geometry which ultimately consisting of lines, parabolas and arcs.

A simple way to create horizontal and vertical geometry for the super alignment is by using horizontal intersection points (HIPs) with arcs and transitions (eg spirals), and vertical intersection points (VIPs) with parabolas or arcs for placing the vertical geometry.

However the super alignment also allows for more **complex construction methods** to define the lines, arcs and transitions (the horizontal elements) that within ***12d Model*** are be grouped together to make up the Parts of the horizontal geometry.

For example, a horizontal line can be defined to be a given parallel offset of a selected line segment from another string. Or an arc of a known radius going through a point of another string and having a a leading and a trailing transition.

Similarly **complex construction methods** are available to define the lines, parabolas and arcs (the vertical elements) that are grouped together within***12d Model*** to make up the vertical geometry.

So for a super alignment, the Horizontal Geometry is made up of **horizontal parts**, which may be as simple as a straight line between two points or an arc with a given centre and radius, or a horizontal intersection point with given leading and trailing transitions, or the **horizontal parts** can be very complicated where for example the points of a line are defined by the offset intersection of the two arcs from other strings.

Similarly for a super alignment, the Vertical Geometry is made up of **vertical parts**.

The definition of **parts** that are not fully defined are classified as being

(a) **fixed** if the part is fully defined

(b) **floating** if the part has one degree of freedom. That is, one unknown is still to be determined.

(c) **free** if the part has two degrees of freedom. That is, two unknowns are still to be determined.

When the **parts** definitions have some degree of freedom, the extra restriction that the horizontal parts must be linked **tangentially** is imposed to try and fully defined the horizontal geometry.

If, with tangentiality, ***12d Model*** can fully determine the horizontal geometry then the horizontal geometry is said to **solve**. For more information, see 21.9.1 Fixed, Floating and Free, and Solving.

Once it solves, the horizontal geometry consists of linked known segments of types lines, arcs and transition, and so can be drawn in a plan view.

So the horizontal geometry is made up of the constructive definitions called **parts**, and if the horizontal geometry solves, it also contains the resultant **horizontal segments**.

For example, if a ***Free Arc with Known Radius*** (an arc with known radius and leading and trailing transitions of known lengths) is placed between two known lines then the tangentiality condition means that the position of the ***Free Arc with Known Radius*** (arc and two transitions) is uniquely defined as a arc and leading and trailing transitions on the intersection point of the two lines. Or just a fillet between the two lines if the two transitions have zero length.

Similarly the Vertical Geometry is made up of **parts**, which are made up of simple or complicated combinations of straight lines, arcs or a parabolas. Again the extra restriction that the vertical parts must be linked tangentially is often necessary to fully defined the vertical geometry. The vertical geometry is then said to be **solved** and the solved vertical geometry simply consists of linked known segments of types lines, arcs and parabolas, and so can be drawn in a section view.

So the vertical geometry is made up of the constructive definitions called **parts**, and if the vertical geometry solves, it also contains the resultant **vertical segments**.

Important Note

The horizontal and vertical parts of a super alignment are only displayed when the super alignment is being edited. When the super alignment is not being edited, the segments of the solved parts are displayed and any unsolved parts are displayed as red crosses.

See 21.9.1 Fixed, Floating and Free, and Solving
See 21.9.2 Types of Horizontal Parts
See 21.9.3 Types of Vertical Parts
See 21.9.4 Definitions of Super Alignment Horizontal Parts
See 21.9.5 Definitions of Super Alignment Vertical Parts
For the documentation of the options on the **Edit SA** toolbar, see 14.4.11 Edit Super Alignment.

21.9.1 Fixed, Floating and Free, and Solving

The definition of fixed and floating part has to do with how many unknowns, or degrees of freedom, exist for a part.

Fixed parts, as the names suggests, are totally fixed in position, radius etc.

Floating parts have one unknown, or one degree of freedom. Solving for that one unknown will then totally determine the part.

Free parts have two unknowns, or two degree of freedom. Solving for both unknowns then totally determines the part.

The most common examples for the major parts used in the horizontal and vertical geometry are:

Horizontal Geometry

Fixed line - its position in space is totally defined

(a) line given by two points on the line

(b) line given by one point on the line and a bearing

Floating lines - one unknown

(a) line going through one known point. The bearing of the line is unknown

(b) line of a given bearing. A point that the line goes through is needed to fix it.

This unknown as the names suggests, are totally fixed in position, radius etc.

Free lines - a totally unconstrained line

Fixed arc - the arcs position in space and radius is totally defined

(a) arc given by centre and radius

(b) arc defined by three points on the arc

(c) arc defined by centre and two points on the arc

Floating arc - one unknown

(a) arc partially defined by one point on the arc and a radius.

(b) two points on an arc.

This unknown as the names suggests, are totally fixed in position, radius etc.

Free arc - two unknowns

(a) only the radius of the arc is known

(b) only one point on the arc is known

(c) only the arc length is known

Vertical Geometry

Fixed line - its position and shape in (chainage, height) is totally defined

(a) line given by two points on the line

(b) line given by one point on the line and a grade

Floating lines - one unknown

(a) line going through one known point. The grade of the line is unknown

(b) line of a given grade. A point that the line goes through is needed to fix it.

Free lines - a totally unconstrained line

Fixed parabola - the parabolas position in (chainage, height) space is totally defined

(a) parabola given by three points on the parabola

(b) parabola defined by its apex and effective signed radius

Floating parabola - one unknown

(a) one point on the parabola and K value are known

(b) two points on an parabola are known.

Free parabola - two unknowns

(a) only the K value of the arc is known

(b) only one point on the parabola is known

(c) only the length of the parabola is known

Fixed arc - the arcs position in (chainage, height) and radius is totally defined

(a) arc given by centre and radius

(b) arc defined by three points on the arc

(c) arc defined by centre and two points on the arc

Floating arc - one unknown

(a) arc partially defined by one point on the arc and a radius.

(b) two points on an arc.

This unknown as the names suggests, are totally fixed in position, radius etc.

Free arc - two unknowns

(a) only the radius of the arc is known

(b) only one point on the arc is known

(c) only the arc length is known

Solving

The horizontal and vertical geometry of a super alignment is made up of a series of Horizontal and Vertical Parts where the sequencing and types of the Parts, plus the condition that, if possible, each Part is **tangential** to the adjacent elements, allows ***12d Model*** to calculate and solve for all the unknowns in the Part definitions and totally define the super alignment.

The order and the whether adjacent parts are *Fixed*, *Floating* or *Free* is very important as only certain combinations will satisfy the tangentiality conditions and hence solve to give a super alignment.

If, with tangentiality, ***12d Model*** can fully determine the horizontal or vertical geometry then the horizontal or vertical geometry is said to **solve**.

Once it solves, the horizontal geometry consists of linked known segments of types lines, arcs and transition, and so can be drawn in a plan view.

So the horizontal geometry is made up of the constructive definitions called **parts**, and if the horizontal geometry solves, it also contains the resultant **horizontal segments**.

Similarly the vertical geometry is made up of the constructive definitions called **parts**, and if the vertical geometry solves, it also contains the resultant **vertical segments**.

The horizontal and vertical parts of a super alignment are only displayed when the super alignment is being edited. When the super alignment is not being edited, the solved segments are displayed and any unsolved parts are displayed as red crosses.

See 21.9.2 Types of Horizontal Parts

See 21.9.3 Types of Vertical Parts
See 21.9.4 Definitions of Super Alignment Horizontal Parts
See 21.9.5 Definitions of Super Alignment Vertical Parts

21.9.2 Types of Horizontal Parts

See 21.9.2.1 Horizontal IPs
See 21.9.2.2 Horizontal Lines - Fixed
See 21.9.2.3 Horizontal Lines - Floating
See 21.9.2.2 Horizontal Lines - Fixed
See 21.9.2.5 Horizontal Arcs - Fixed
See 21.9.2.6 Horizontal Arcs - Floating
See 21.9.2.7 Horizontal Arcs - Free
See 21.9.2.8 Horizontal Transitions - Floating
See 21.9.2.9 Horizontal Transitions - Free
See 21.9.2.10 Horizontal Transitions - Compound

21.9.2.1 Horizontal IPs

Speed, go to 21.9.4.1.1 HIP: Speed
Radius, go to HIP: Radius:
Length, go to HIP: Arc LengthReturn to 21.9.3 Types of Vertical Parts

Return to 21.9.2 Types of Horizontal Parts

21.9.2.2 Horizontal Lines - Fixed

Two points, go to Fixed Line: Two Points
Point & direction, go to Fixed Line: Point & Direction

Return to 21.9.2 Types of Horizontal Parts

21.9.2.3 Horizontal Lines - Floating

Through point, go to Floating Line: Through a Point:
Direction, go to Floating Line: Known Direction
From end, go to Floating line: Known End and Nominal Length

Return to 21.9.2 Types of Horizontal Parts

21.9.2.4 Horizontal Lines - Free

Free, go to Free Line: No Constraints

Return to 21.9.2 Types of Horizontal Parts

21.9.2.5 Horizontal Arcs - Fixed

Centre, radius and ends, go to Fixed Arc: Known Centre, Radius and Start and End
Three points, go to Fixed Arc: Arc Passes through Three Known Points
Point, radius & direction, go to Fixed Arc: Known Start Point and Direction, Known Radius

Return to 21.9.2 Types of Horizontal Parts

21.9.2.6 Horizontal Arcs - Floating

Point and radius, go to Floating Arc: Known Radius and Passes through a Known Point

Point and direction, go to Floating Arc: Known Point and Tangent Direction at the Point

Two points, go to Floating Arc: Arc Passes through Two known Points

From end through point, go to Floating Arc: Arc Passes through Known point and End of Previous or Next Element

From end, radius and length, go to Floating Arc: Arc has Know Radius and Length, and End of Previous or Next Element

From end, radius and sweep, go to Floating Arc: Arc has Know Radius and Sweep Angle, and End of Previous or Next Element

Return to 21.9.2 Types of Horizontal Parts

21.9.2.7 Horizontal Arcs - Free

Radius, go to Free Arc: Known Radius

Through point, go to Free Arc: Passes through Known Point

Length, go to Free Arc: Arc has a Known Length

From end, go to Free Arc: Arc has a Known End Point

Return to 21.9.2 Types of Horizontal Parts

21.9.2.8 Horizontal Transitions - Floating

Arc to arc, go to Floating Transition: From Arc to Arc

Taper, go to Floating Transition: Taper

Transition, go to Floating Transition: Known Length

Return to 21.9.2 Types of Horizontal Parts

21.9.2.9 Horizontal Transitions - Free

Transition, go to Free Transition: Unconstrained

Back to back, go to Free Transition: Back to Back Transitions of Known Lengths Between Lines

Arc to arc, go to Free Transition: Back to Back Transitions Between Two Arcs

Taper, go to Free Taper: Taper of Known Length

Return to 21.9.2 Types of Horizontal Parts

21.9.2.10 Horizontal Transitions - Compound

Three centred, go to Compound Transitions: Three Centre Curve

Two centred, go to Compound Transition: Two Centred Curve

Compound transition, lengths, go to Compound Transition: Two Transition Curves of Known Lengths

Compound line length, go to Compound Transition: Two Transitions Separated by Line of Known Length

Compound curve, go to Compound Transition: Two Arcs of Known Radii

Compound taper, go to Compound Transition: Two Arcs of Known Radii with a Taper

21.9.3 Types of Vertical Parts

See 21.9.3.1 Vertical IPs
See 21.9.3.2 Vertical Lines - Fixed
See 21.9.3.3 Vertical Lines - Floating
See 21.9.3.4 Vertical Lines - Free
See 21.9.3.5 Vertical Parabolas - Fixed
See 21.9.3.6 Vertical Parabolas - Floating
See 21.9.3.7 Vertical Parabolas - Free
See 21.9.3.8 Vertical Arcs - Fixed
See 21.9.3.9 Vertical Arcs - Floating
See 21.9.3.10 Vertical Arcs - Free

21.9.3.1 Vertical IPs

Speed, go to VIP: Speed
Radius, go to VIP: Radius
K value, go to VIP: K-value
Length, go to VIP: Length:
Asymmetric, go to VIP: Asymmetric

Return to 21.9.3 Types of Vertical Parts

21.9.3.2 Vertical Lines - Fixed

Two points, go to Fixed Line: Two Points
Point & grade, go to Fixed Line: Known Point and Grade

Return to 21.9.3 Types of Vertical Parts

21.9.3.3 Vertical Lines - Floating

Point, go to Floating Line: Known Point
Grade, go to Floating Line: Known Grade
From end, go to Floating Line: Known End and Nominal Length

Return to 21.9.3 Types of Vertical Parts

21.9.3.4 Vertical Lines - Free

Free, go to Free line - No Constraints

Return to 21.9.3 Types of Vertical Parts

21.9.3.5 Vertical Parabolas - Fixed

Three points, go to Fixed Parabola - Parabola Passes through Three Known Points
Apex & radius, go to Fixed Parabola: Known Apex and Effective Radius

Return to 21.9.3 Types of Vertical Parts

21.9.3.6 Vertical Parabolas - Floating

Two points, go to Floating Parabola: Through Two Points
Apex, go to Floating Parabola: Known Apex
Radius, go to Floating Parabola: Known Point and Effective Radius
Grade, go to Floating Parabola: Known Point and Grade at that Point
K value, go to Floating Parabola: Known Point and K Value
From end & grade, go to Floating Parabola: Known Length and Grade at End
From end & radius, go to Floating Parabola: Known Radius and Nominal Length
From end & point, go to Floating Parabola: known Point and Known End

Return to 21.9.3 Types of Vertical Parts

21.9.3.7 Vertical Parabolas - Free

Through point, go to Free Parabola: Known Point
Length, go to Free Parabola: Known Length
Radius, go to Free Parabola: Know Effective Radius
RL, go to Free Parabola: Known Parabolic Length/100
K value, go to Free Parabola: Known K Value
Maximum length, go to Free Parabola: Fit with Maximum Length
Asymmetric, go to Free Asymmetric Parabola: Two Known Lengths
Compound, go to Free Compound Parabola: Two Parabolas with Optional Total Parabolic Length

Return to 21.9.3 Types of Vertical Parts

21.9.3.8 Vertical Arcs - Fixed

Fixed, go to Fixed Arc- Known Centre, Start and End Points

Return to 21.9.3 Types of Vertical Parts

21.9.3.9 Vertical Arcs - Floating

Point and radius, go to Floating Arc: Known Point and Radius
Point and grade, go to Floating Arc: Known Point and Grade at Point
Two points, go to Floating Arc: Through Two Known Points

Return to 21.9.3 Types of Vertical Parts

21.9.3.10 Vertical Arcs - Free

Radius, go to Free Arc: Known Radius
Point, go to Free Arc: Known Point on the Arc
Length, go to Free Arc: Known Length of Arc

Return to 21.9.3 Types of Vertical Parts

21.9.4 Definitions of Super Alignment Horizontal Parts

See 21.9.4.1 Horizontal IPs
See 21.9.4.2 Horizontal Lines
See 21.9.4.3 Horizontal Arcs
See 21.9.4.4 Horizontal Transitions
See 21.9.4.5 Horizontal Computators

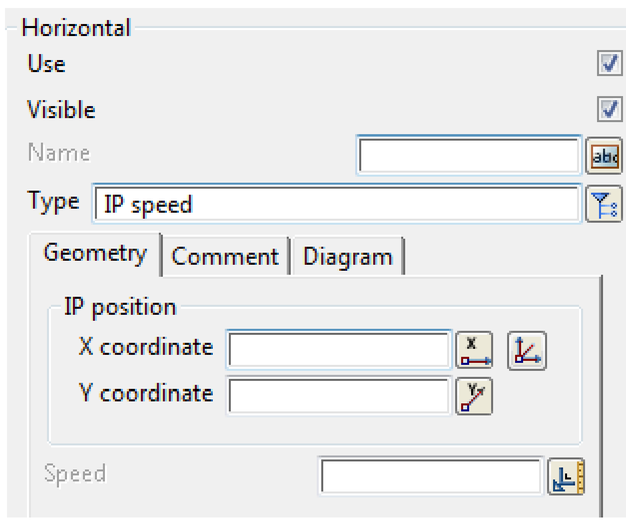
21.9.4.1 Horizontal IPs

See HIP: Speed
See HIP: Radius:
See HIP: Arc Length

21.9.4.1.1 HIP: Speed

A horizontal intersection point (HIP) is created and the sizes of the arc and transition curves for the HIP come from the Design Template of the super alignment.

Same As Button
Use the **Same As** button to select an existing HIP from the screen



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Type choice box types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

X coordinate, Y coordinate

the (x,y) coordinates for the HIP can be typed in, or selected using the **X**, **Y** *or* **XY** *icons.*

Speed

if **Speed** *is not bank, then that is the speed used when looking up the Design Template for the HIP*

If **Speed** *is left bank, then the* **Design Speed** *for the super alignment in used when looking up the Design Template.*

Comment tab

the text type into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

Set button

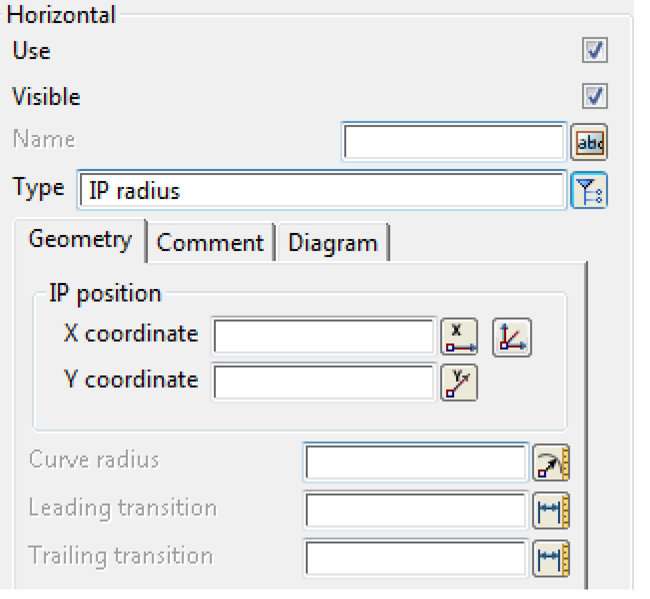
the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

after clicking the **Same As** *button, an existing HIP can be selected from the screen and its information piped into the appropriate fields in the panel.*

HIP: Radius:
A horizontal intersection point (HIP) is created and the sizes of the arc and transition curves for the HIP are given.

Same As Button
Use the **Same As** button to select an existing HIP from the screen



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.

If not ticked, the Part is not used in the horizontal geometry.

Field Description	Type	Defaults	Pop-Up
Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Field Description	Type	Defaults	Pop-Up
Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Field Description	Type	Defaults	Pop-Up
Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

X coordinate, Y coordinate

the (x, y) coordinates for the HIP can be typed in, or selected using the **X**, **Y** *or* **XY** *icons.*

Curve radius

if not blank, the radius used for the arc on the HIP.

If blank then no arc is used on the HIP.

Note: *if the* **Curve radius** *is blank then the* **Leading transition** *and* **Trailing transition** *must also be blank. That is, there is* ***no arc or transitions*** *on the HIP.*

Leading transition

if not blank, then it is the length of the leading transition on the HIP. This can only be non zero if the **Curve radius** *is also non zero*

If blank then there is no leading transition on the HIP.

Leading transition

if not blank, then it is the length of the trailing transition on the HIP. This can only be non zero if the **Curve radius** *is also non zero*

If blank then no trailing transition on the HIP.

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

Set button

the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

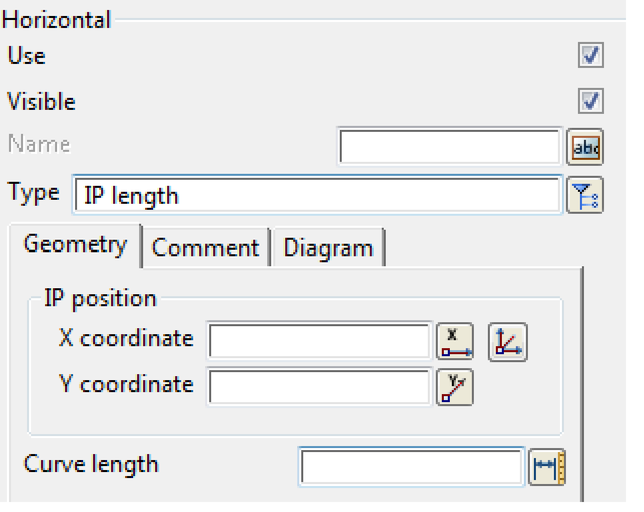
after clicking the **Same As** *button, an existing HIP can be selected from the screen and its information piped into the appropriate fields in the panel.*

HIP: Arc Length

A horizontal intersection point (HIP) is created with an arc of a given arc length.

Same As Button

Use the **Same As** button to select an existing HIP from the screen



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Visible — tick box — ticked

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name — text box — blank

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Type — choice box — Pop-Up: types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

X coordinate, Y coordinate

the (x,y) coordinates for the HIP can be typed in, or selected using the **X**, **Y** *or* **XY** *icons.*

Curve length

the length used for the arc on the HIP. This field can not be left blank.

If zero then no arc is used on the HIP.

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

Set — button

the **Set** *button must be clicked for the information for this Part to be used.*

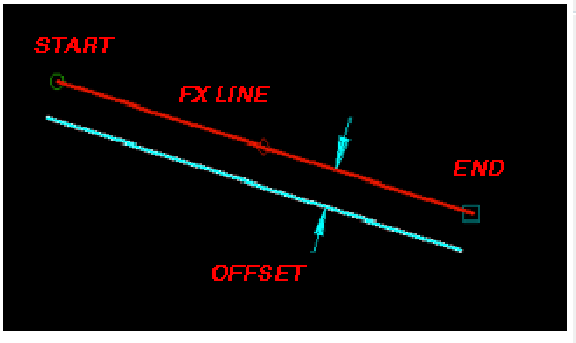
Same As button

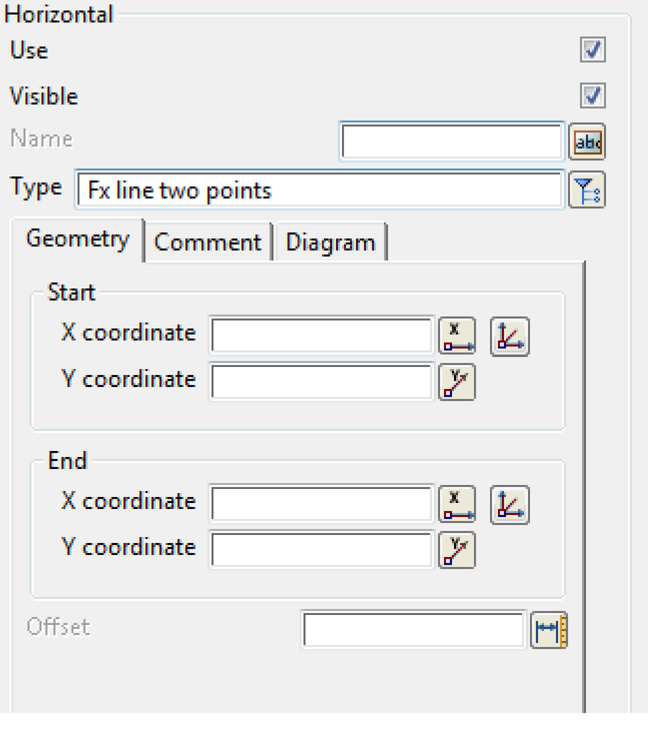
after clicking the **Same As** *button, an existing HIP can be selected from the screen and its information piped into the appropriate fields in the panel.*

21.9.4.2 Horizontal Lines

See Fixed Line: Two Points
See Fixed Line: Point & Direction
See Floating Line: Through a Point:
See Floating Line: Known Direction
See Floating line: Known End and Nominal Length
See Free Line: No Constraints

Fixed Line: Two Points
The *Fixed Line* is a given offset from a line that passes through two known points.





Offset:
-ve to the left of the line from start to end points
+ve to the right

Same As Button:
Use the **Same As** button to select an existing line segment from the screen

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Start X coordinate, Y coordinate

the start (x,y) coordinates for the line can be typed in, or selected using the **X**, **Y** *or* **XY** *icons.*

End X coordinate, Y coordinate

the end (x,y) coordinates for the line can be typed in, or selected using the **X**, **Y** *or* **XY** *icons.*

Offset measure box Point to point or String to point

offset of the Fixed line from the line through the two points

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

Set button

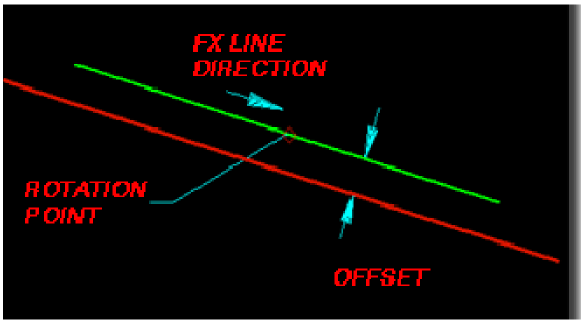
the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

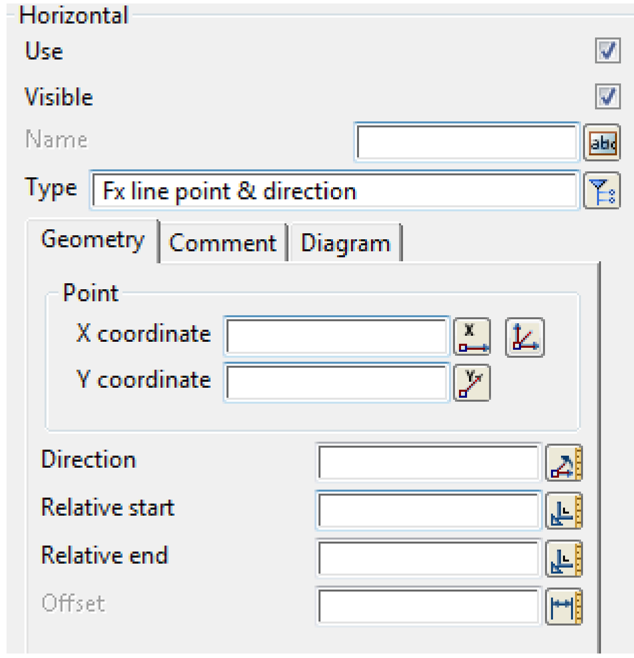
after clicking the **Same As** *button, an existing segment can be selected from the screen and its start and end coordinates are piped into the appropriate fields in the panel.*

Fixed Line: Point & Direction

The *Fixed Line* is a given offset from a line going through a known point and with a known bearing.



Offset:
-ve to the left of the line
+ve to the right



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Field Description	Type	Defaults	Pop-Up
Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Field Description	Type	Defaults	Pop-Up
Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Field Description	Type	Defaults	Pop-Up
Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

X coordinate, Y coordinate

the (x,y) coordinates that the line goes through can be typed in, or selected using the **X**, **Y** *or* **XY** *icons.*

Field Description	Type	Defaults	Pop-Up
Direction	measure box		available measures

the direction as a bearing can be typed in using 4.17.1 HP Notation *(dd.mmssss) or selected using the measure box.*

Relative start measure box available measures

the plan (2d) distance, relative to the given point, to start the line at. A negative value is to the left of the point, and a positive value is to the right of the point.

Relative end measure box available measures

the plan (2d) distance, relative to the given point, to end the line at. A negative value is to the left of the point, and a positive value is to the right of the point.

Offset measure box Point to point or String to point

offset of the Fixed line from the line through the two points

Comment tab

the text typed into the text box is stored as a comment for the Part.

***Note:** If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

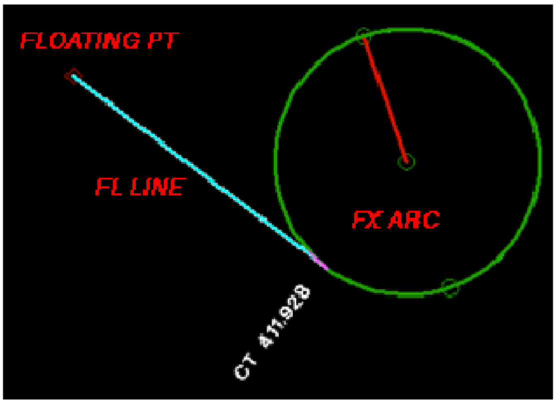
Set button

the **Set** *button must be clicked for the information for this Part to be used.*

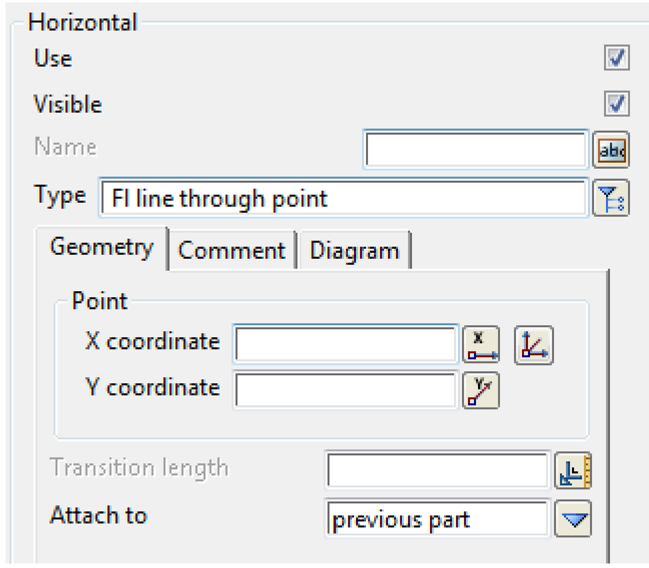
Same As button

after clicking the **Same As** *button, an existing HIP can be selected from the screen and its information piped into the appropriate fields in the panel.*

Floating Line: Through a Point:
The *Floating Line* line passes through a **known point** but the bearing is unknown.



Example:
Floating line from a point becomes a tangent to an arc when attached to a following fixed arc



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

X coordinate, Y coordinate

the (x,y) coordinates for the floating line can be typed in, or selected using the **X**, **Y** *or* **XY** *icons.*

Transition length	measure box		available measures

if non zero, a transition curve of this length is at the end of the Floating Line that **Attach to** *specifies.*
If blank, no transition curve is used

Attach to	choice box		previous part, next part

the Part that the Floating Line is made tangential to. If there is a transition, it is at this end of the Floating line.

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

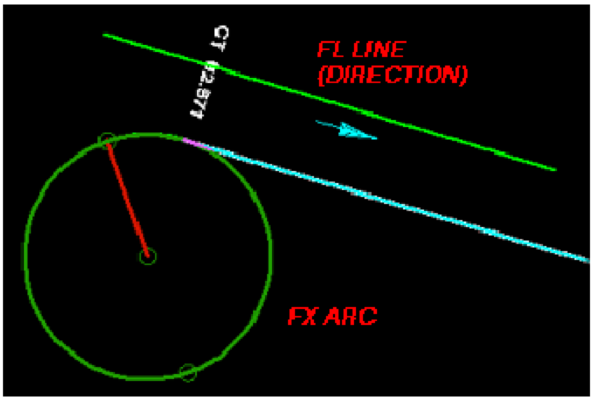
Set button

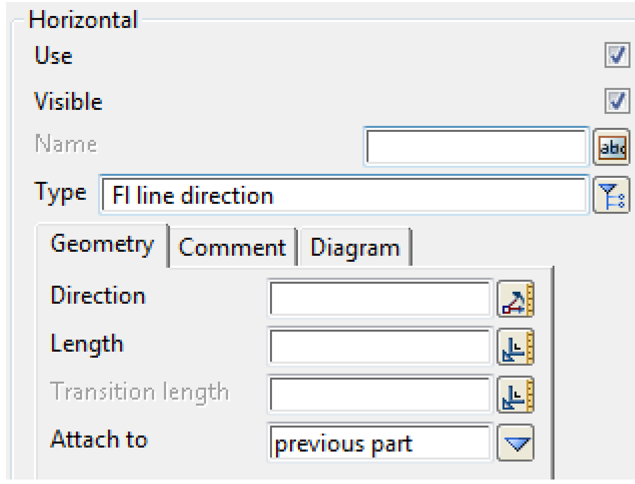
the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

after clicking the **Same As** *button, an point is selected from the screen and its coordinates are piped into the appropriate fields in the panel.*

Floating Line: Known Direction
The *Floating Line* has a **known bearing** and a length.





Example:
Floating line with a direction becomes a tangent to an arc when attached to a previous fixed arc

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab
the geometrical information defining the Part

Direction	measure box		available measures

the direction of the line as a bearing can be typed in using 4.17.1 HP Notation *(dd.mmssss) or selected using the measure box.*

Length	measure box		available measures

length of the Floating Line

Transition length	measure box		available measures

if non zero, a transition curve of this length is at the end of the Floating Line that **Attach to** *specifies.*
If blank, no transition curve is used

Attach to choice box previous part, next part

the Part that the Floating Line is made tangential to. If there is a transition, it is at this end of the Floating line.

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

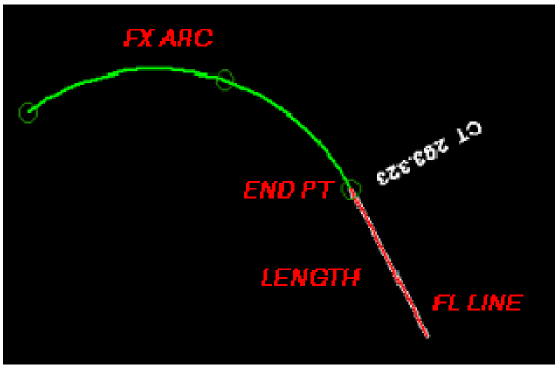
Set button

the **Set** *button must be clicked for the information for this Part to be used.*

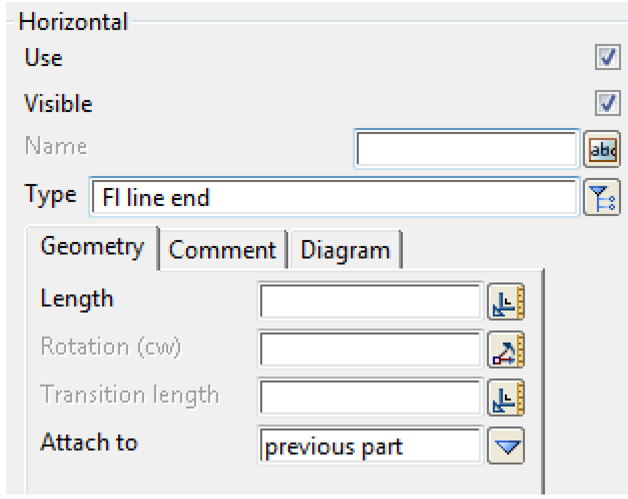
Same As button

after clicking the **Same As** *button, an existing HIP can be selected from the screen and its information piped into the appropriate fields in the panel.*

Floating line: Known End and Nominal Length
The Floating Line has a **known nominal length** and is attached tangentially to the end of previous part or the beginning of the next part.



Example:
Extends from the end of a previous fixed arc for a given length



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Field Description	Type	Defaults	Pop-Up
Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Field Description	Type	Defaults	Pop-Up
Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Field Description	Type	Defaults	Pop-Up
Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Field Description	Type	Defaults	Pop-Up
Length	measure box		available measures

the nominal length of the Floating Line

Field Description	Type	Defaults	Pop-Up
Rotation (cw)	measure box		available measures

if non zero then the bearing of the line is decreased by Rotation degrees (and is no longer tangential). The transition, if it exists, is still tangential to the **Attach to** *part. See diagram at the end of this panel description.*

Field Description	Type	Defaults	Pop-Up
Transition length	measure box		available measures

if non zero, a transition curve of this length is at the end of the Part that **Attach to** *specifies. So if the Floating Line is* **Attach to** *the Previous Part, then the Previous Part comes first, then the transition and then the line. If the Floating Line is* **Attach to** *the Next Part, then the line comes first and then the transition, and the Next Part.*

If blank, no transition curve is used

Attach to	choice box	previous part, next part

the Part that the Floating Line is attached to, and made tangential to. If there is a transition, it is at this end of the Floating line.

Comment tab

the text typed into the text box is stored as a comment for the Part.

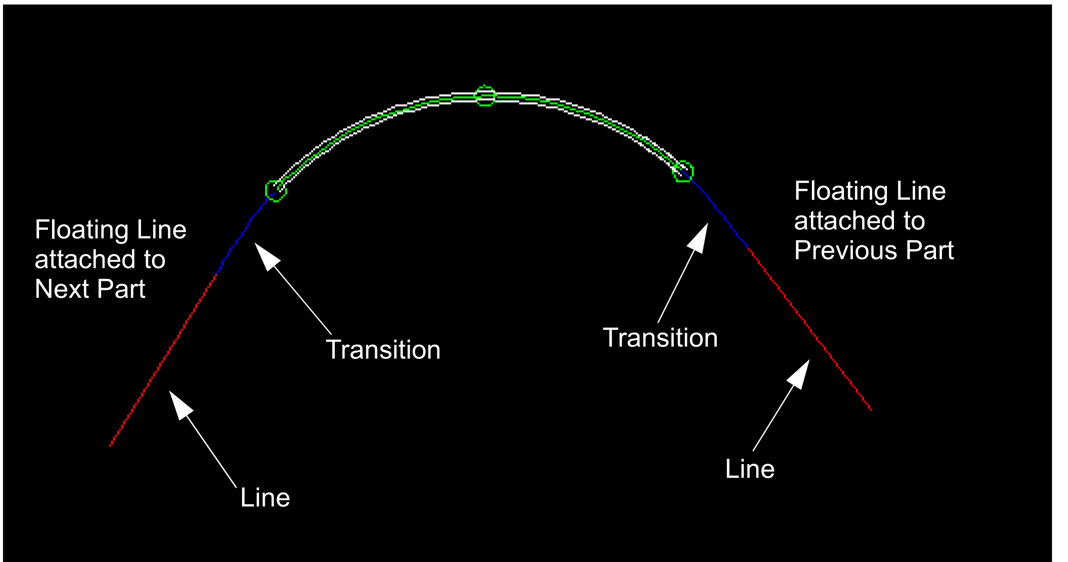
Note: *If a comment exists, a # is placed after the type of the part in the Horizontal Parts list.*

Set	button

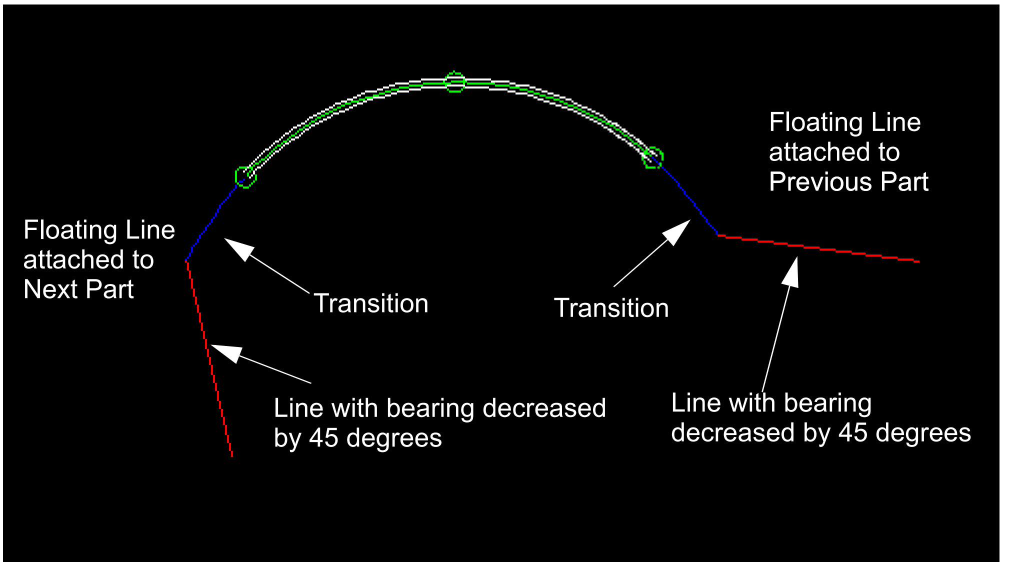
the **Set** *button must be clicked for the information for this Part to be used.*

Same As	button

not used for this Part



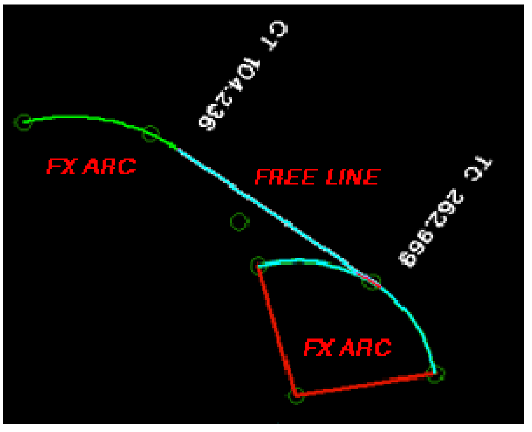
Two Examples of *Floating Lines* with Rotation = 0

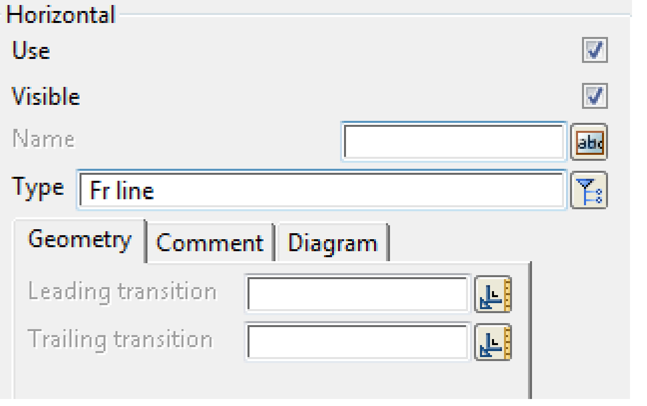


Two Examples of *Floating Lines* with Rotation = 45 degrees

Free Line: No Constraints

The **Free Line** is not constrained. It is determined by having to be tangential to the Parts at either end of the Line.





Example:
Free line is a tangent from a previous fixed arc to the following fixed arc.

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Leading transition	measure box		available measures

if non zero, a transition curve of this length is at the beginning of the Free Line.
If blank or 0, no leading transition curve is used.

Trailing transition	measure box		available measures

if non zero, a transition curve of this length is at the end of the Free Line.
If blank or 0, no trailing transition curve is used.

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* **#** *is placed after the type of the part in the Horizontal Parts list.*

Set button

the **Set** *button must be clicked for the information for this Part to be used.*

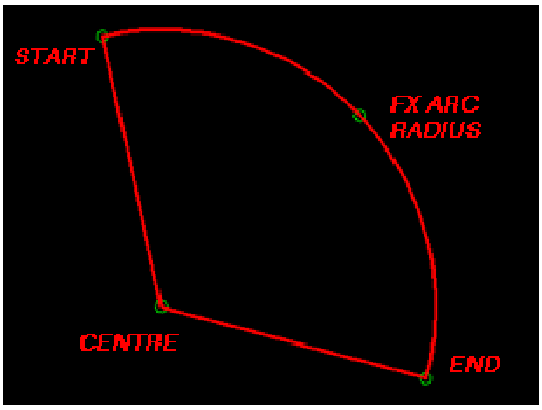
Same As button

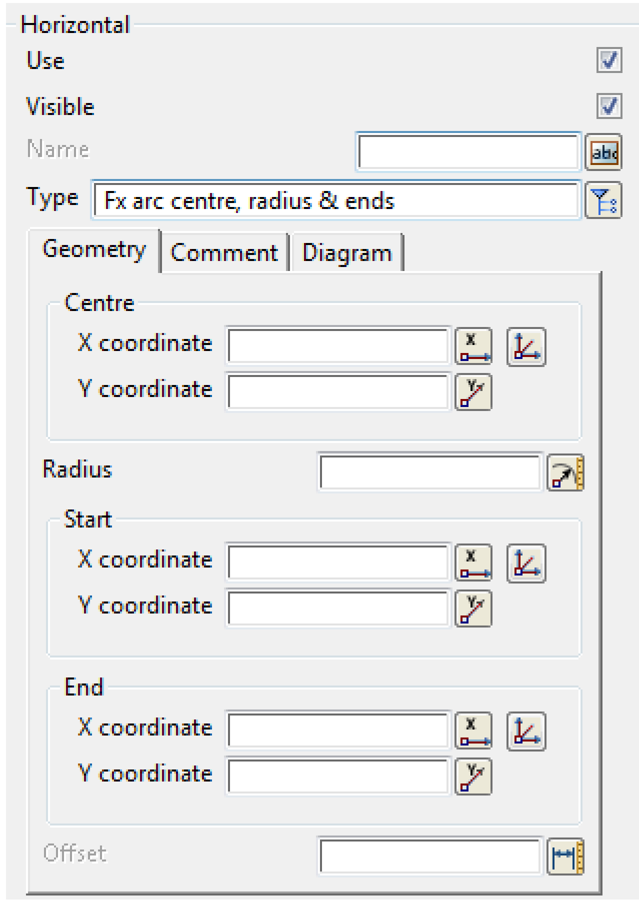
not used

21.9.4.3 Horizontal Arcs

See Fixed Arc: Known Centre, Radius and Start and End
See Fixed Arc: Arc Passes through Three Known Points

Fixed Arc: Known Centre, Radius and Start and End
The *Fixed Arc* is fully defined by a given offset from an arc with **known arc centre**, a **known radius** and **known start** and **end** points.





Offset:
-ve to the left of the arc: direction is start to end
+ve to the right of the arc

Same As Button:
Use the **Same As** button to select an existing arc from the screen

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Type choice box types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Centre X coordinate, Y coordinate

the centre (x,y) coordinates for the arc can be typed in, or selected using the **X**, **Y** *or* **XY** *icons.*

Radius radius box Measure Radius

the radius of the arc. This can be typed in or selected using the Measure Radius pop-up.

Start X coordinate, Y coordinate

the start (x,y) coordinates for the arc can be typed in, or selected using the **X**, **Y** *or* **XY** *icons.*

End X coordinate, Y coordinate

the end (x,y) coordinates for the arc can be typed in, or selected using the **X**, **Y** *or* **XY** *icons.*

Offset measure box Measure Length

offset of the Fixed Arc from the arc defined above by Centre, Radius, Start and End points

Positive offset is to the right of the arc going in the direction of the arc from Start point to End point. Negative offset is to the left of the arc going in the direction of the arc from Start point to End point.

Comment tab

the text typed into the text box is stored as a comment for the Part.

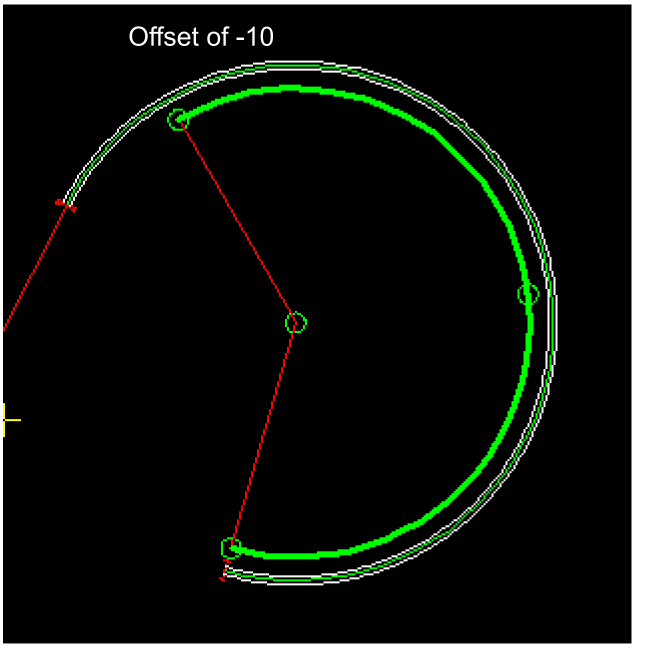
Note: *If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

Set button

the **Set** *button must be clicked for the information for this Part to be used.*

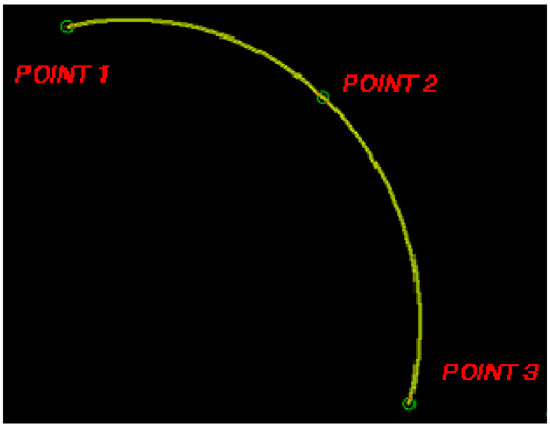
Same As button

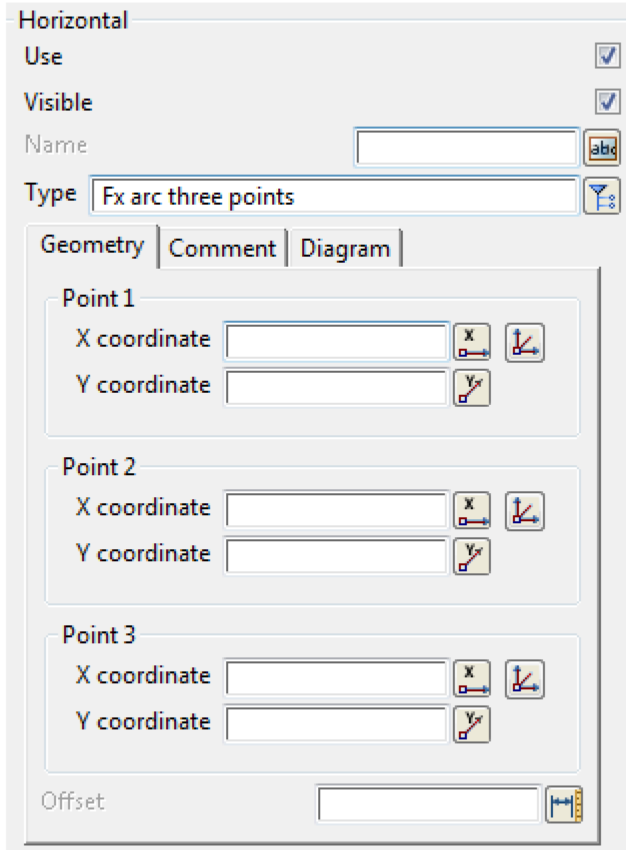
after clicking the **Same As** *button, an existing arc can be selected from the screen and its information piped into the appropriate fields in the panel.*



Fixed Arc: Arc Passes through Three Known Points

The *Fixed Arc* is fully defined by a given offset from an arc going through **three known points**.





Offset:
-ve to the left of the arc: direction is point 1 to 3
+ve to the right of the arc

Same As Button:
Use the **Same As** button to select an existing arc from the screen

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Point 1 X coordinate, Y coordinate

the (x,y) coordinates of the first point on the arc can be typed in, or selected using the **X**, **Y** *or* **XY** *icons*

Point 2 X coordinate, Y coordinate

the (x,y) coordinates of the second point on the arc can be typed in, or selected using the **X**, **Y** *or* **XY** *icons*

Point 3 X coordinate, Y coordinate

the (x,y) coordinates of the third point on the arc can be typed in, or selected using the **X**, **Y** *or* **XY** *icons*

Offset measure box Measure Length

offset of the Fixed Arc from the arc defined by the three points above.

Positive offset is to the right of the arc going in the direction of the arc from Point 1 to Point 3. Negative offset is to the left of the arc going in the direction of the arc from Point 1 to Point 3.

Comment tab

the text typed into the text box is stored as a comment for the Part.

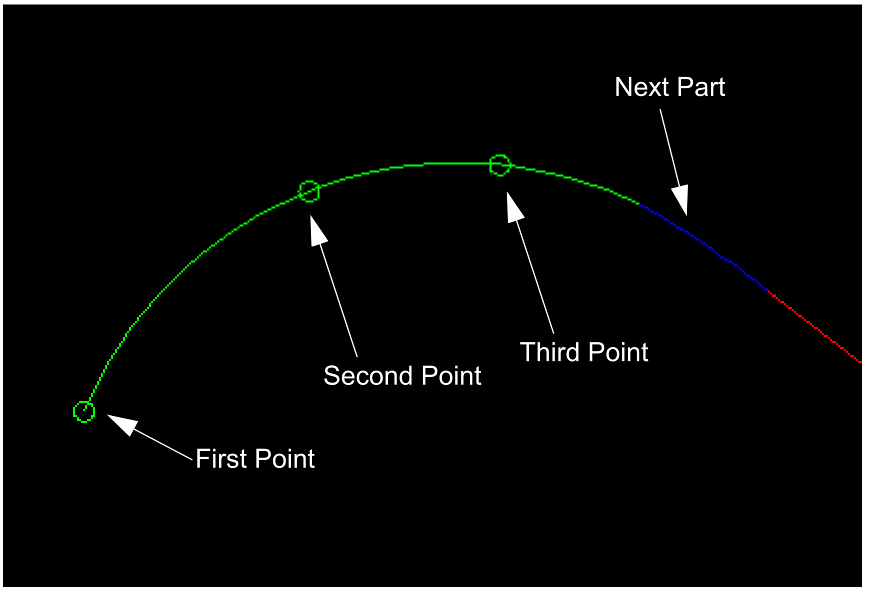
Note: *If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

Set button

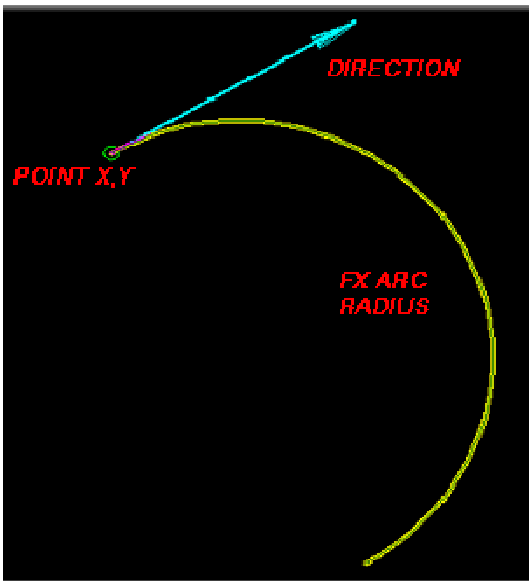
the **Set** *button must be clicked for the information for this Part to be used.*

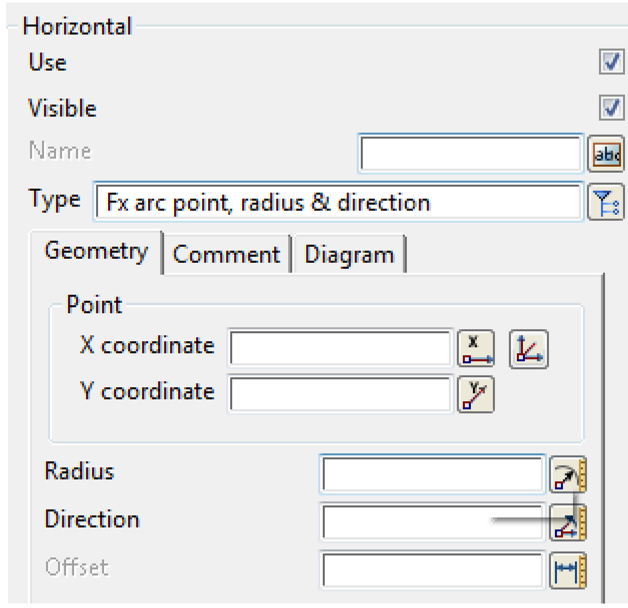
Same As button

after clicking the **Same As** *button, an existing arc can be selected from the screen and its information piped into the appropriate fields in the panel.*



Fixed Arc: Known Start Point and Direction, Known Radius
The *Fixed Arc* is fully defined by a given offset from an arc **starting** at a **known point** with a **known direction** for the tangent to the arc at that point, and a **known radius**.





Offset:
-ve to the left of the arc: direction is from given point
+ve to the right of the arc

Radius:
-ve to the left of direction
+ve to the right of the direction

Same As Button:
Use the **Same As** button to select an existing arc from the screen

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Field Description	Type	Defaults	Pop-Up
Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Field Description	Type	Defaults	Pop-Up
Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Field Description	Type	Defaults	Pop-Up
Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

X coordinate, Y coordinate

the (x,y) coordinates on the point on the arc can be typed in, or selected using the **X**, **Y** *or* **XY** *icons.*

Radius radius box Measure Radius

the radius of the arc. This can be typed in or selected using the Measure Radius pop-up.

Direction measure box available measures

the direction of the tangent of the point on the arc as a bearing can be typed in using 4.17.1 HP Notation *(dd.mmssss) or selected using the measure box.*

Offset measure box Measure Length

offset of the Fixed Arc from the arc defined above by Point, Radius, and Direction (bearing) at Point.

Positive offset is to the right of the arc going in the direction of the arc.
Negative offset is to the left of the arc going in the direction of the arc.

Comment tab

the text typed into the text box is stored as a comment for the Part.

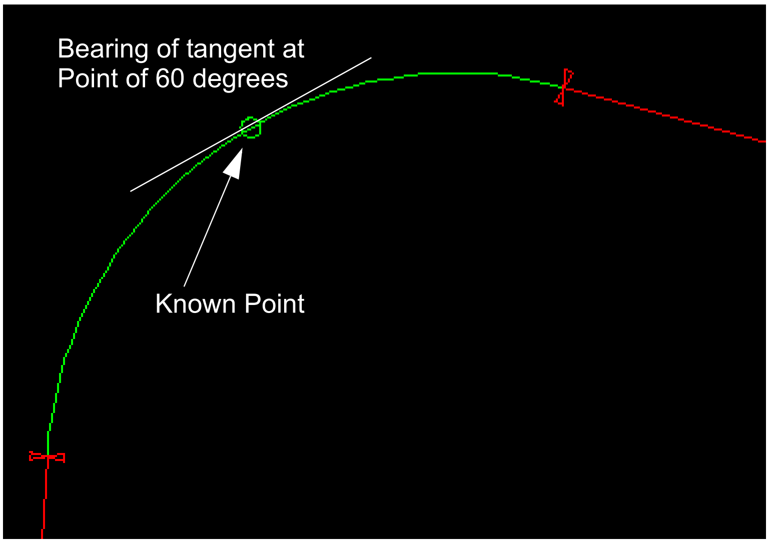
Note: *If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

Set button

the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

after clicking the **Same As** *button, an existing arc can be selected from the screen and its information piped into the appropriate fields in the panel.*

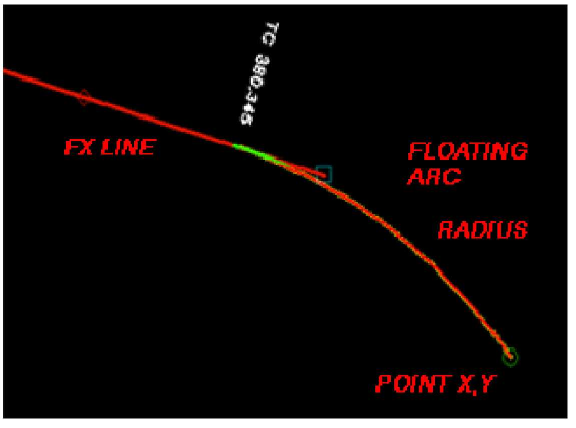


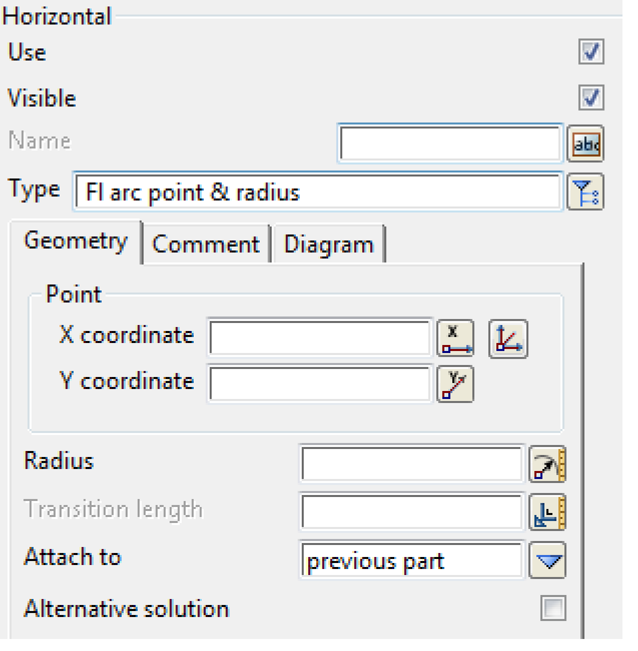
Floating Arc: Known Radius and Passes through a Known Point

The *Floating Arc* passes through a **known point** and has a **known radius**.

It may not always be possible for the *Floating Arc* to actually solve.

For example in the case where it follows a fixed line, if the perpendicular distance between the known point and the line is greater than the given radius, then there is no solution.





Example:
The Floating Arc may become fully defined when it is attached to a previous fixed line

Radius:
-ve to the left of direction
+ve to the right of the direction

Same As Button:
Use the **Same As** button to select an existing arc from the screen

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Field Description	Type	Defaults	Pop-Up
Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Field Description	Type	Defaults	Pop-Up
Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Field Description	Type	Defaults	Pop-Up
Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

X coordinate, Y coordinate

the (x,y) coordinates of the point on the arc can be typed in, or selected using the **X**, **Y** *or* **XY** *icons.*

Radius radius box Measure Radius

the radius of the arc. This can be typed in or selected using the Measure Radius pop-up.

Transition length measure box available measures

if non zero, a transition curve of this length is at the end of the Floating Arc that **Attach to** *specifies. If blank, no transition curve is used*

Attach to choice box previous part, next part

the Part that the Floating Arc is attached to, and made tangential to. If there is a transition, it is at this end of the Floating line.

Alternative solution tick box not ticked

there can be two solutions. One solution is first shown.

If ticked, the other solution is shown.

Comment tab

the text typed into the text box is stored as a comment for the Part.

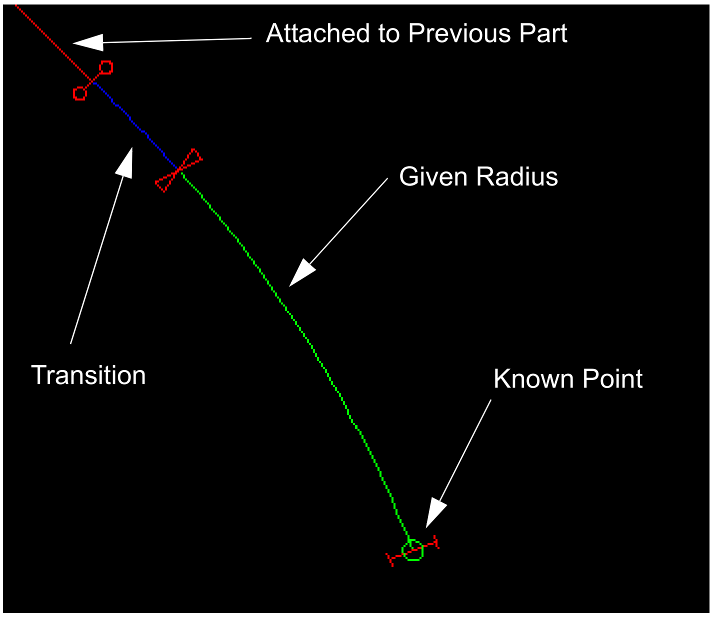
Note: *If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

Set button

the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

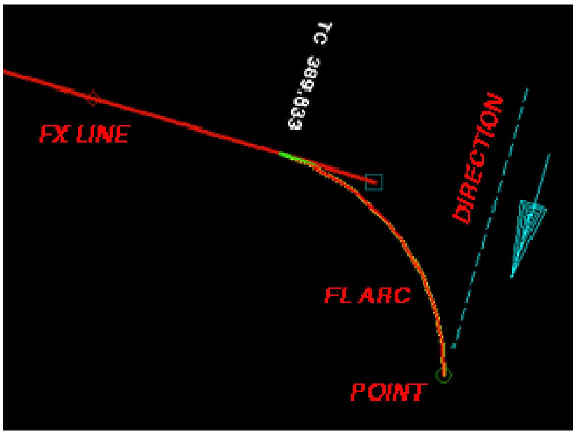
after clicking the **Same As** *button, an existing arc can be selected from the screen and its information piped into the appropriate fields in the panel.*

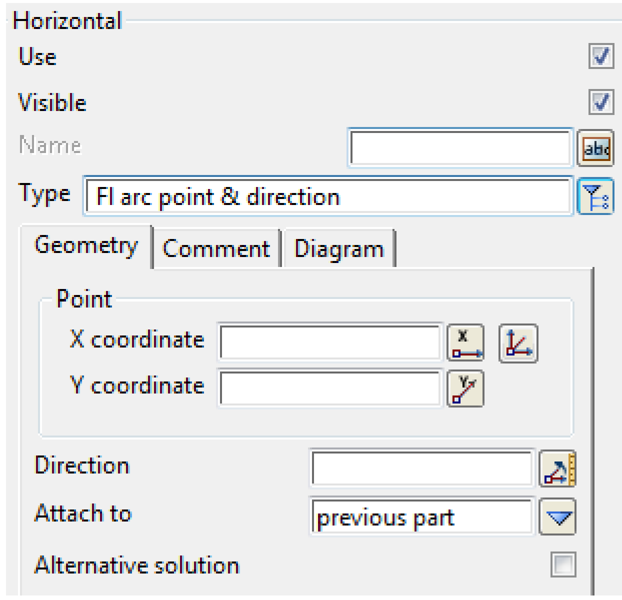


Floating Arc: Known Point and Tangent Direction at the Point
The *Floating Arc* passes through a **known point** and has a **known direction** for the **tangent** to the arc at the **known point.**

It may not always be possible for the *Floating Arc* to actually solve.

For example in the case where it follows a fixed line, if the perpendicular distance between the known point and the line is greater than the given radius, then there is no solution.





Example:
The Floating Arc may become fully defined when it is attached to a previous fixed line

Radius:
-ve to the left of direction
+ve to the right of the direction

Same As Button:
Use the **Same As** button to select an existing arc from the screen

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Field Description	Type	Defaults	Pop-Up
Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Field Description	Type	Defaults	Pop-Up
Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Field Description	Type	Defaults	Pop-Up
Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab
the geometrical information defining the Part

X coordinate, Y coordinate

the (x,y) coordinates for the point on the arc can be typed in, or selected using the **X**, **Y** *or* **XY** *icons.*

Direction measure box available measures

the direction of the tangent of the point on the arc as a bearing can be typed in using 4.17.1 HP Notation *(dd.mmssss) or selected using the measure box.*

Attach to choice box previous part, next part

the Part that the Floating Arc is attached to, and made tangential to.

Alternative solution tick box not ticked

there can be two solutions. One solution is first shown.

If ticked, the other solution is shown.

Comment tab

the text typed into the text box is stored as a comment for the Part.

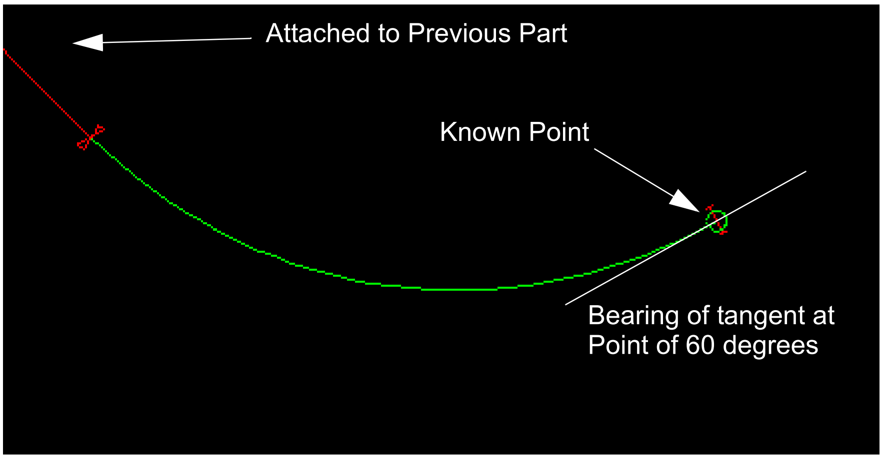
Note: *If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

Set button

the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

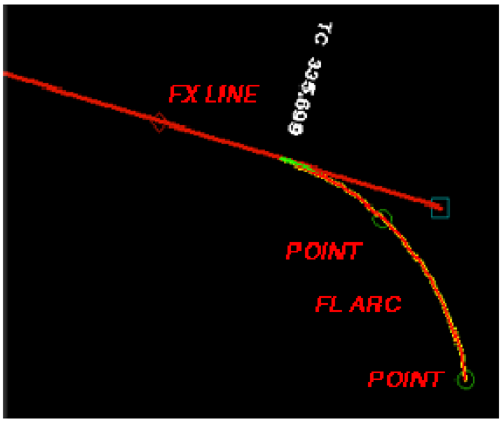
after clicking the **Same As** *button, an existing arc can be selected from the screen and its information piped into the appropriate fields in the panel.*

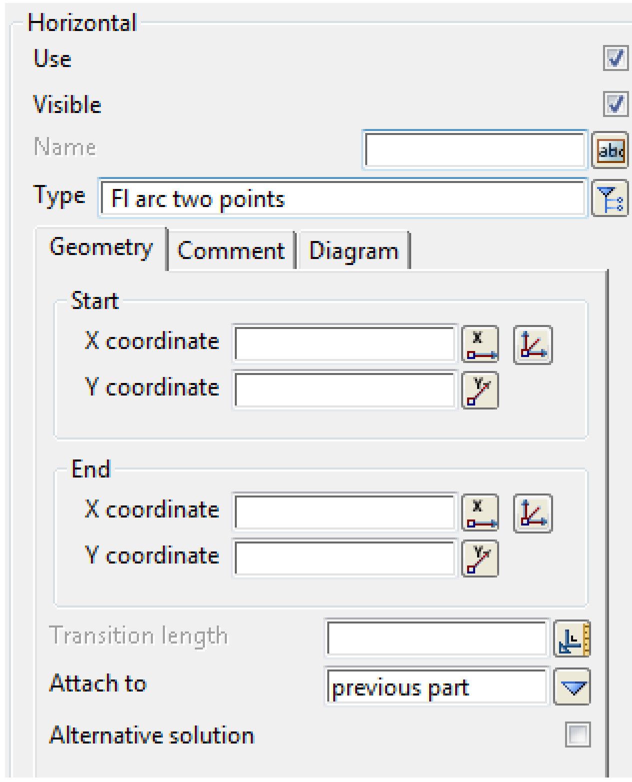


Floating Arc: Arc Passes through Two known Points
The *Floating Arc* passes through **two known points.**

It may not always be possible for the *Floating Arc* to actually solve.

For example in the case where it follows a fixed line and the two known points are on the lines, then there is no solution.





Example:
The Floating Arc may become fully defined when it is attached to a previous fixed line

Same As Button:
Use the **Same As** button to select an existing arc from the screen

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Field Description	Type	Defaults	Pop-Up
Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Field Description	Type	Defaults	Pop-Up
Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Field Description	Type	Defaults	Pop-Up
Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Start X coordinate, Y coordinate

the (x,y) coordinates of the first point on the arc can be typed in, or selected using the **X**, **Y** *or* **XY** *icons*

End X coordinate, Y coordinate

the (x,y) coordinates of the second point on the arc can be typed in, or selected using the **X**, **Y** *or* **XY** *icons*

Transition length measure box available measures

if non zero, a transition curve of this length is at the end of the Floating Arc that **Attach to** *specifies. If blank, no transition curve is used*

Attach to choice box previous part, next part

the Part that the Floating Arc is attached to, and made tangential to. If there is a transition, it is at this end of the Floating line.

Alternative solution tick box not ticked

there can be two solutions. One solution is first shown.

If ticked, the other solution is shown.

Comment tab

the text typed into the text box is stored as a comment for the Part.

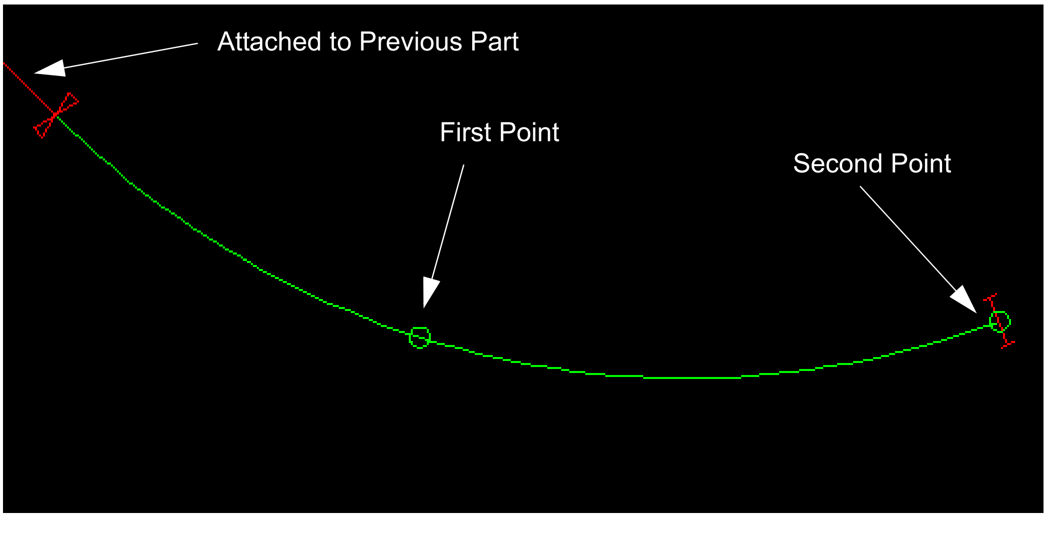
Note: *If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

Set button

the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

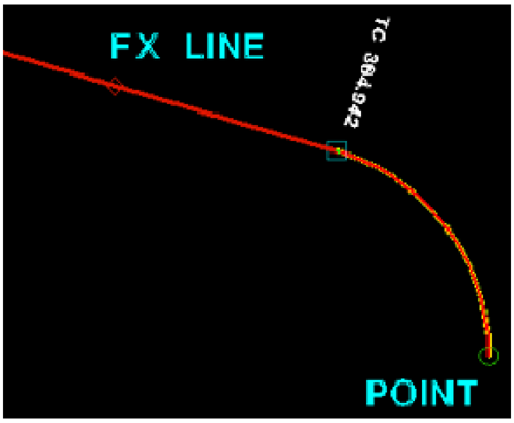
after clicking the **Same As** *button, an existing arc can be selected from the screen and its information piped into the appropriate fields in the panel.*

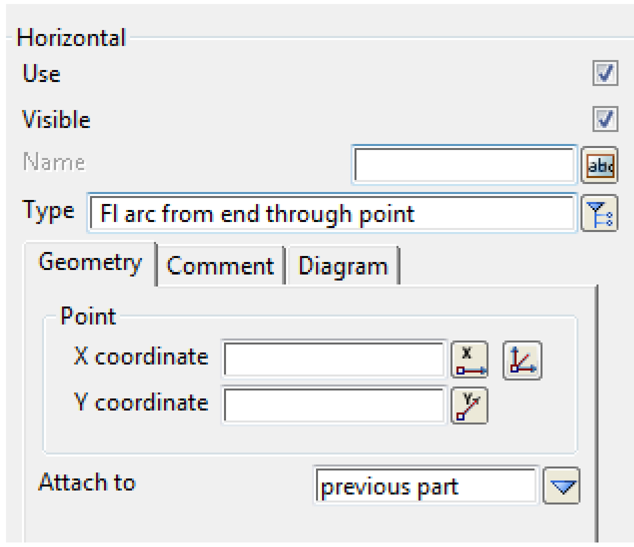


Floating Arc: Arc Passes through Known point and End of Previous or Next Element
The *Floating Arc* is attached to the actual end of the previous or next element and also passes through **a known point**.

It may not always be possible for the *Floating Arc* to actually solve.

For example in the case where it follows a fixed line and the known point is on the line, then there is no solution.





Example:
The Float Arcing may become fully defined when it is attached to a previous fixed line

Same As Button:
Use the **Same As** button to select an existing arc from the screen

The fields and buttons used in the panel have the following functions.

Field Description   Type   Defaults   Pop-Up

Use   tick box   ticked

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Visible   tick box   ticked

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name   text box   blank

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Type   choice box   types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

X coordinate, Y coordinate

the (x,y) coordinates of a point on the arc can be typed in, or selected using the **X**, **Y** *or* **XY** *icons*

Attach to choice box previous part, next part

the Part that the Floating Arc is attached to, and made tangential to.

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

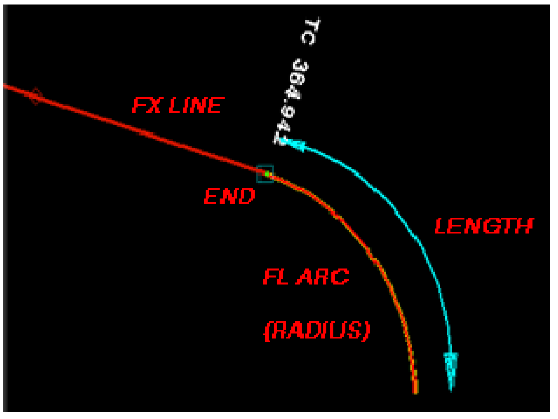
Set button

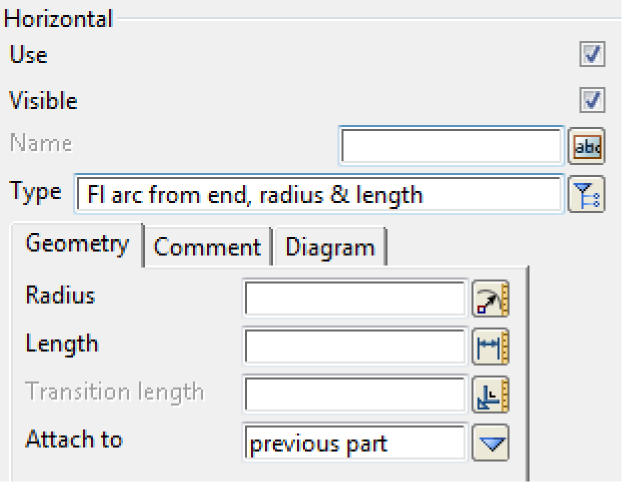
the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

after clicking the **Same As** *button, an existing arc can be selected from the screen and its information piped into the appropriate fields in the panel.*

Floating Arc: Arc has Know Radius and Length, and End of Previous or Next Element
The *Floating Arc* is attached to the actual **end of** the previous or next **element** and also has a **known radius** and **known length.**





Example:
The Floating Arc may become fully defined when it is attached to a previous fixed line

Radius:
-ve to the left of direction of travel
+ve to the right of the direction of travel

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Radius	radius box		Measure Radius

the radius of the arc. This can be typed in or selected using the Measure Radius pop-up.

Length	measure box		Measure Length

the length of the arc.

Transition length	measure box		available measures

if non zero, a transition curve of this length is at the end of the Floating Arc that **Attach to** *specifies.*

If blank, no transition curve is used

Attach to choice box previous part, next part

the Part that the Floating Arc is attached to, and made tangential to. If there is a transition, it is at this end of the Floating line.

Comment tab

the text typed into the text box is stored as a comment for the Part.

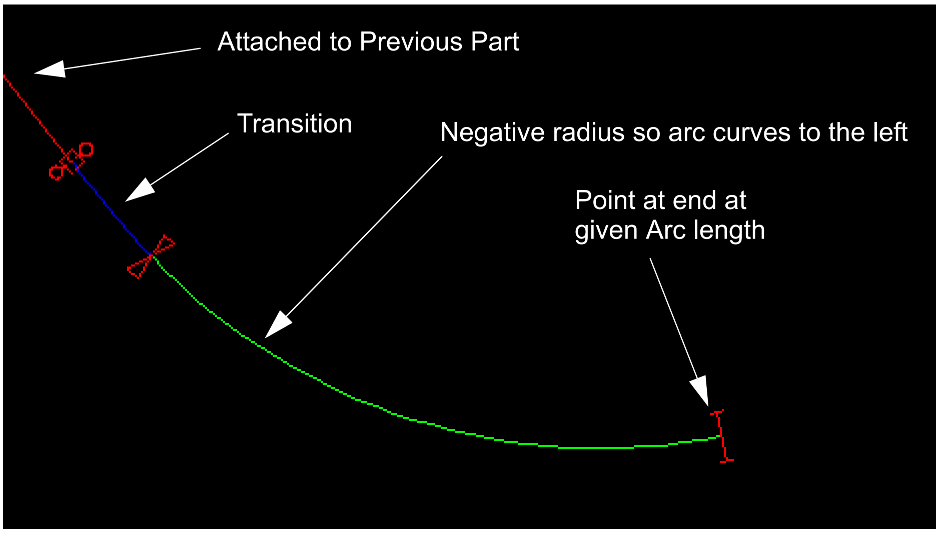
Note: *If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

Set button

the **Set** *button must be clicked for the information for this Part to be used.*

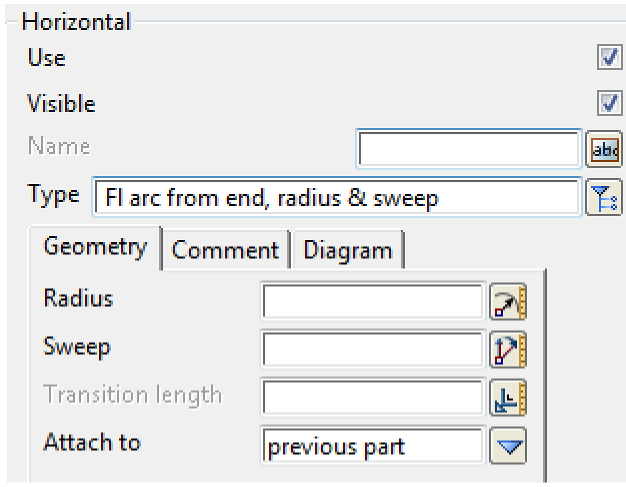
Same As button

after clicking the **Same As** *button, an existing arc can be selected from the screen and its information piped into the appropriate fields in the panel.*



Floating Arc: Arc has Know Radius and Sweep Angle, and End of Previous or Next Element

The *Floating Arc* is attached to the actual **end of** the previous or next **element**, has a **known radius** and **known sweep angle.**



Example:
The Floating Arc may become fully defined when it is attached to a previous fixed line

Radius:
-ve to the left of direction of travel
+ve to the right of the direction of travel

The fields and buttons used in the panel have the following functions.

Field Description — Type — Defaults — Pop-Up

Use — tick box — ticked

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Visible — tick box — ticked

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name — text box — blank

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Type — choice box — types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Radius — radius box — Measure Radius

the radius of the arc. This can be typed in or selected using the Measure Radius pop-up.

Sweep — measure box — available measures

the angle in degrees in 4.17.1 HP Notation *(dd.mm.ssss)that the arc sweep through*

Transition length measure box available measures

if non zero, a transition curve of this length is at the end of the Floating Arc that **Attach to** *specifies. If blank, no transition curve is used*

Attach to choice box previous part, next part

the Part that the Floating Arc is attached to, and made tangential to. If there is a transition, it is at this end of the Floating line.

Comment tab

the text typed into the text box is stored as a comment for the Part.

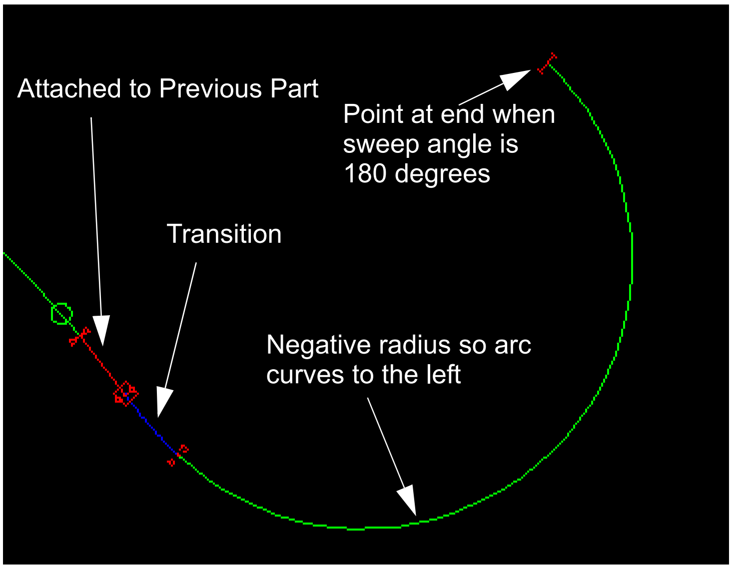
Note: *If a comment exists, a* **#** *is placed after the type of the part in the Horizontal Parts list.*

Set button

the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

after clicking the **Same As** *button, an existing arc can be selected from the screen and its information piped into the appropriate fields in the panel.*

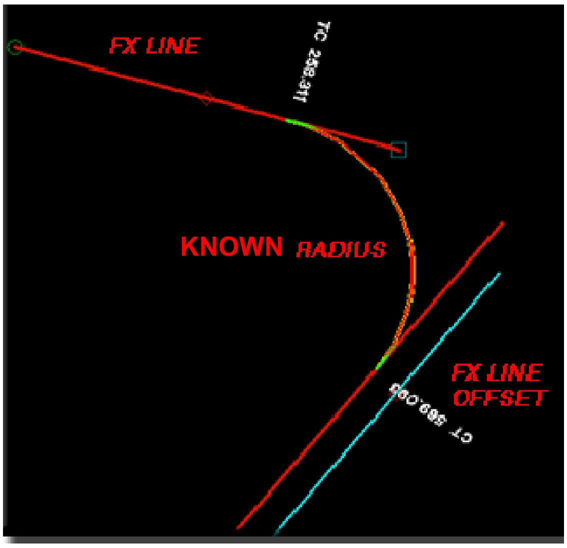


Free Arc: Known Radius

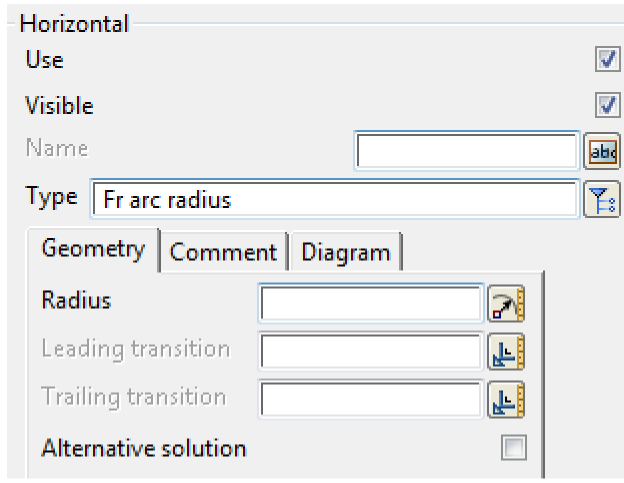
The *Free Arc* has a **known radius**.

It may not always be possible for the *Free Arc* to actually solve.

For example in the case where the Free Arc is between two fixed lines and the two lines are on top of each other.



Example:
The Free Arc become fully defined when it is placed between two fixed lines



Radius:
-ve to the left of direction of travel
+ve to the right of the direction of travel

The fields and buttons used in the panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Use | tick box | ticked

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Visible | tick box | ticked

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name | text box | blank

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Type | choice box | types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Radius | radius box | Measure Radius

the radius of the arc. This can be typed in or selected using the Measure Radius pop-up.

Leading transition | measure box | available measures

if non zero, a leading transition curve of this length is at the start of the Free Arc.

If blank or zero, no leading transition curve is used

Trailing transition measure box available measures

if non zero, a trailing transition curve of this length is at the end of the Free Arc. If blank or zero, no trailing transition curve is used.

Alternative solution tick box not ticked

there can be two solutions. One solution is shown first.

If ticked, the other solution is shown.

Comment tab

the text typed into the text box is stored as a comment for the Part.

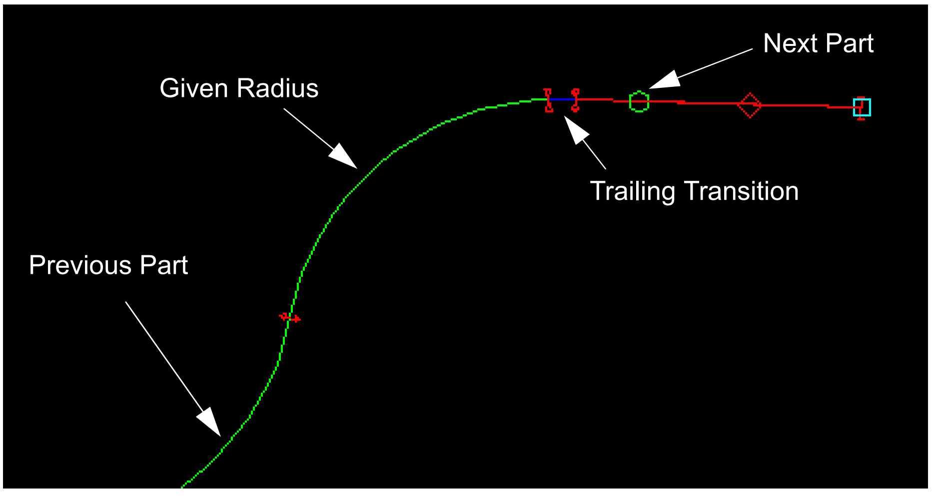
Note: *If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

Set button

the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

after clicking the **Same As** *button, an existing arc can be selected from the screen and its information piped into the appropriate fields in the panel.*

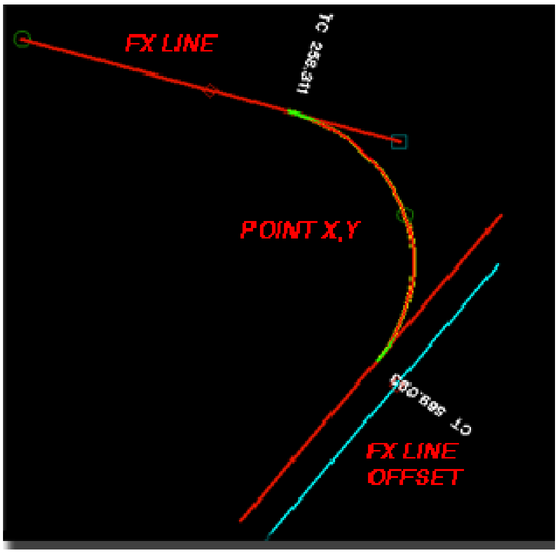


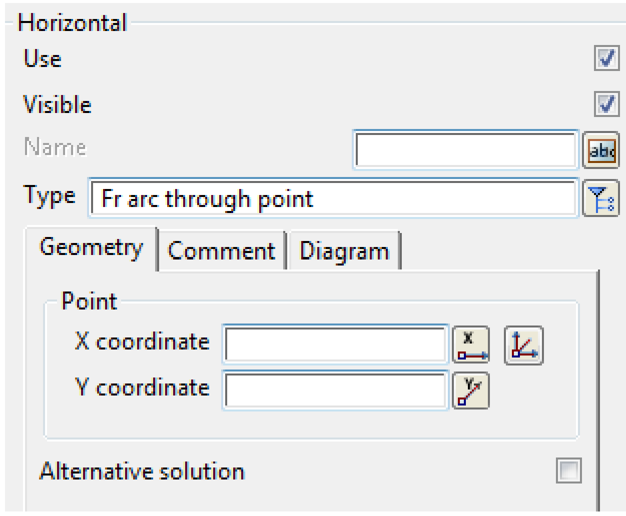
Free Arc: Passes through Known Point

The *Free Arc* passes through a **known point**.

It may not always be possible for the *Free Arc* to actually solve.

For example in the case where the Free Arc is between two fixed lines and the two lines are on top of each other.





Example:
The Free Arc become fully defined when it is placed between two fixed lines

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

X coordinate, Y coordinate

the (x,y) coordinates of a point on the arc can be typed in, or selected using the **X**, **Y** *or* **XY** *icons.*

Alternative solution	tick box	not ticked	

there can be two solutions. One solution is shown first.

If ticked, the other solution is shown.

Comment tab

the text typed into the text box is stored as a comment for the Part.

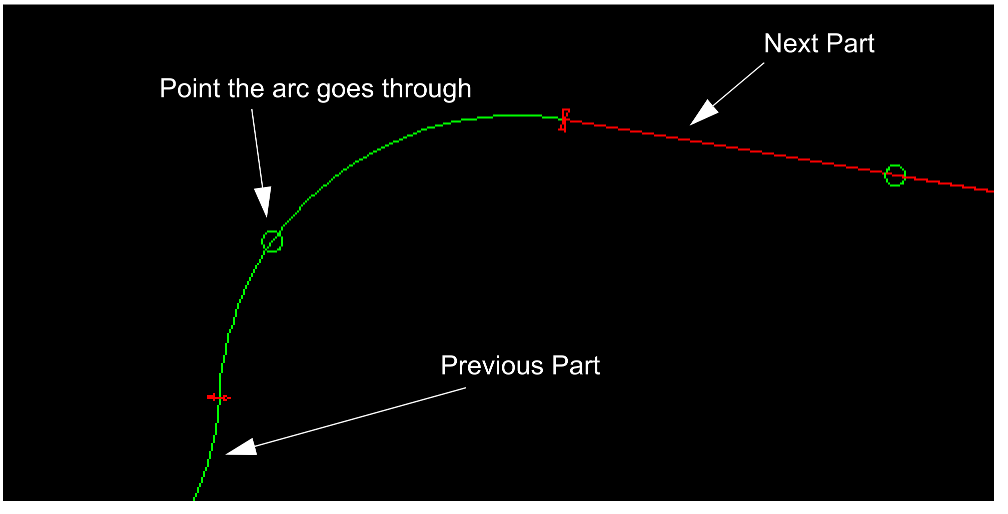
Note: *If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

Set button

the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

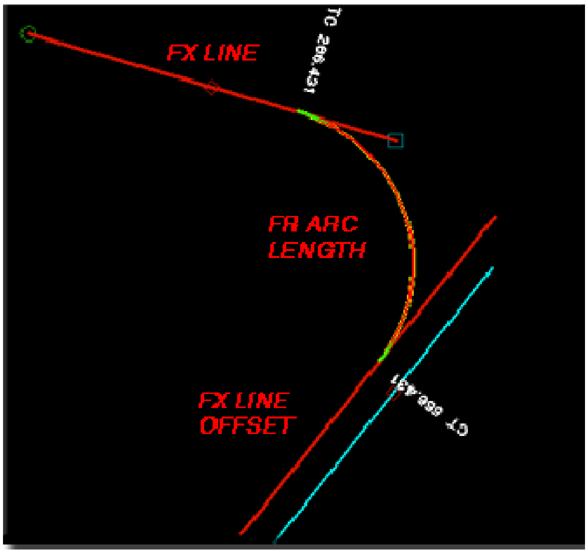
after clicking the **Same As** *button, an existing arc can be selected from the screen and its information piped into the appropriate fields in the panel.*

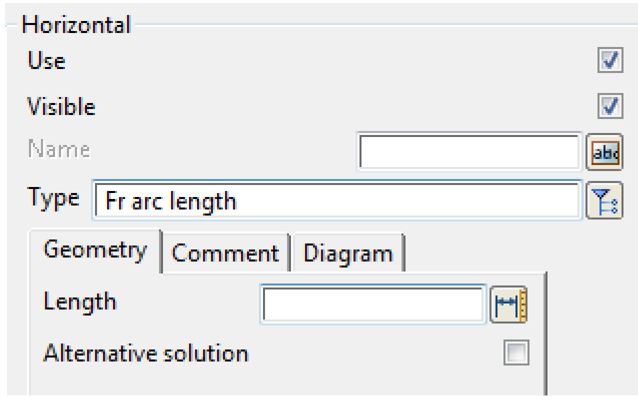


Free Arc: Arc has a Known Length
The *Free Arc* has a **known length**.

It may not always be possible for the *Free Arc* to actually solve.

For example in the case where the Free Arc is between two fixed lines and the two lines are on top of each other.





Example:
The Free Arc become fully defined when it is placed between two fixed lines

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Length	length box		Measure Length

the length of the arc. This can be typed in or selected using the Measure Length pop-up.

Alternative solution tick box not ticked

there can be two solutions. One solution is shown first.

If ticked, the other solution is shown.

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

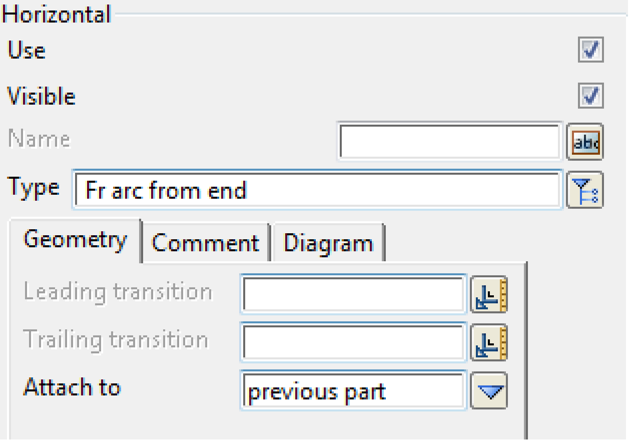
Set button

the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

after clicking the **Same As** *button, an existing arc can be selected from the screen and its information piped into the appropriate fields in the panel.*

Free Arc: Arc has a Known End Point
The *Floating Arc* is attached to a **known end point** of a Previous or Next **Part**.



Example:
The Free Arc become fully defined when it is placed between two fixed lines

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Leading transition	measure box		available measures

if non zero, a leading transition curve of this length is at the start of the Free Arc.
If blank or zero, no leading transition curve is used

Trailing transition	measure box		available measures

if non zero, a trailing transition curve of this length is at the end of the Free Arc.
If blank or zero, no trailing transition curve is used.

Attach to choice box previous part, next part

the Part that the Free Arc is attached to.

Comment tab

the text typed into the text box is stored as a comment for the Part.

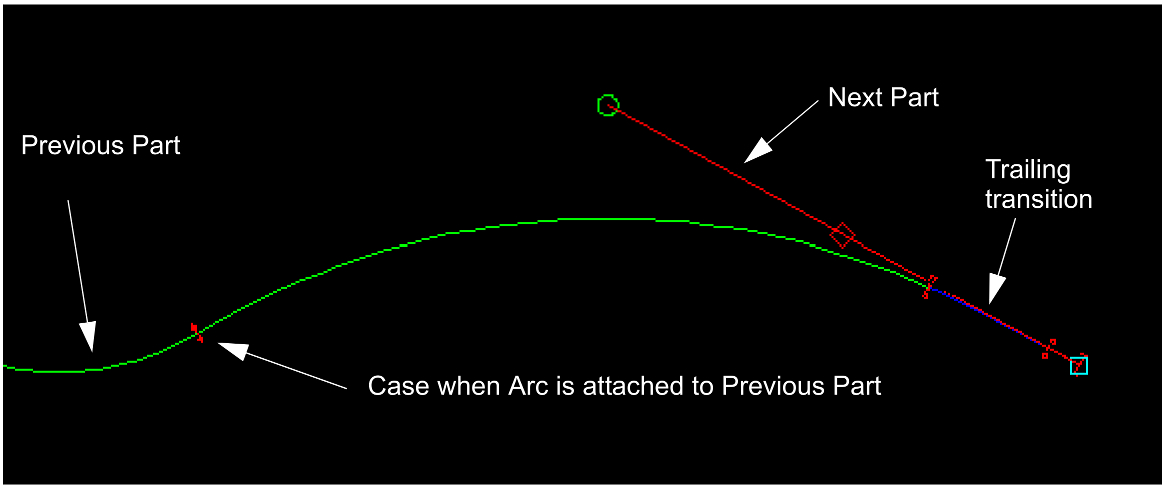
***Note:** If a comment exists, a # is placed after the type of the part in the Horizontal Parts list.*

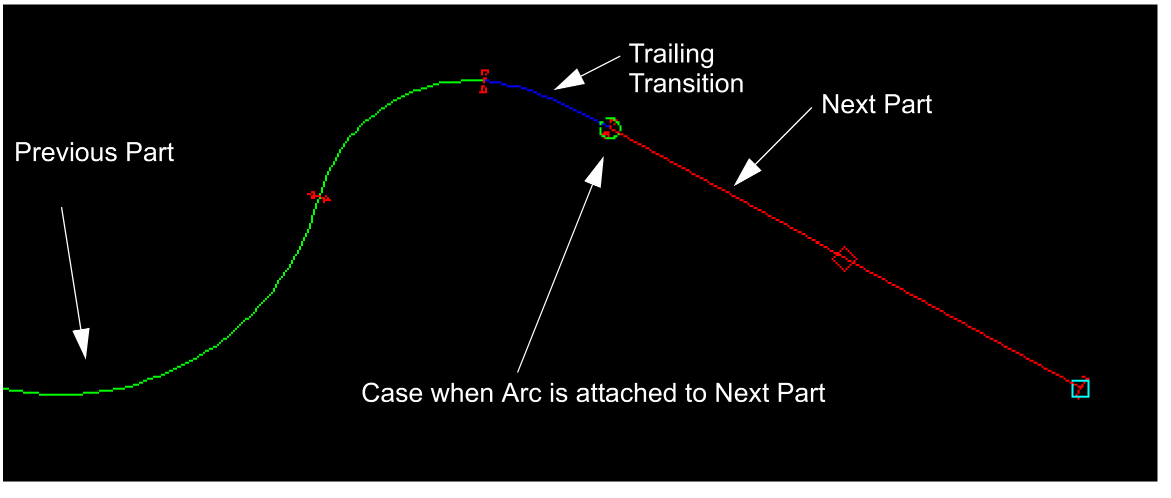
Set button

the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

after clicking the **Same As** *button, an existing arc can be selected from the screen and its information piped into the appropriate fields in the panel.*



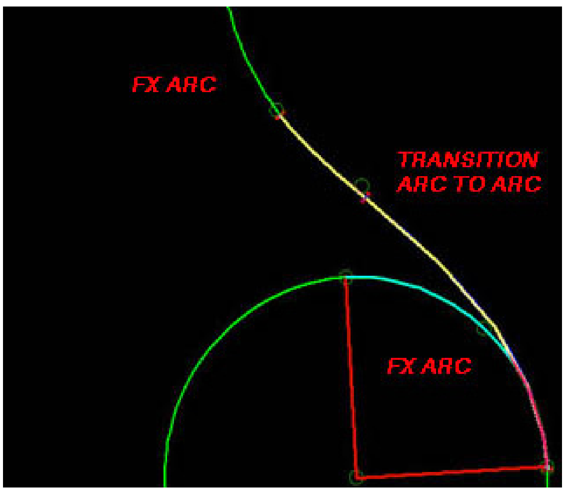


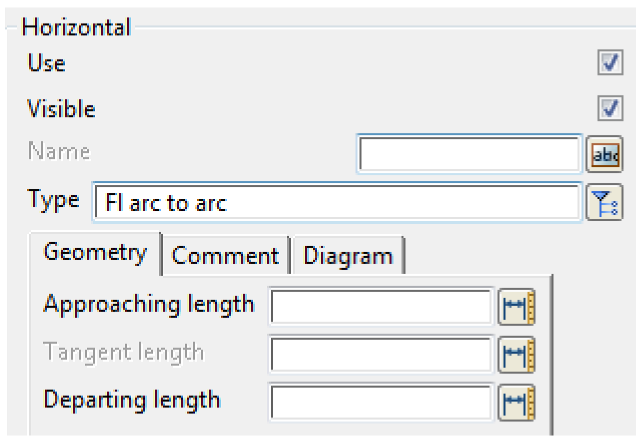
21.9.4.4 Horizontal Transitions

See Floating Transition: From Arc to Arc
See Floating Transition: Taper
See Floating Transition: Known Length
See Free Transition: Unconstrained
See Free Transition: Back to Back Transitions of Known Lengths Between Lines
See Free Transition: Back to Back Transitions Between Two Arcs
See Free Taper: Taper of Known Length
See Compound Transitions: Three Centre Curve
See Compound Transition: Two Centred Curve
See Compound Transition: Two Transition Curves of Known Lengths
See Compound Transition: Two Transitions Separated by Line of Known Length
See Compound Transition: Two Arcs of Known Radii
See Compound Transition: Two Arcs of Known Radii with a Taper

Floating Transition: From Arc to Arc

The *Floating Transition* consists of one or two **transition curves** going **between two arcs**.





Example:
The Floating Transition become fully defined when it is placed between two arcs

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Field Description	Type	Defaults	Pop-Up
Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name text box blank

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Type choice box types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Approaching length measure box Measure Length

the length of the leading transition curve at the start of the Floating Transition.This must be non zero.
This will be tangential to the Previous Part.

Tangent length length box Measure Length

if non zero, a straight of this length is placed between the two transition curves.
This will be tangential to the Previous Part.

Departing length measure box Measure Length

the length of the trailing transition curve at the end of the Floating Transition.This must be non zero.
This will be tangential to the Next Part.

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* **#** *is placed after the type of the part in the Horizontal Parts list.*

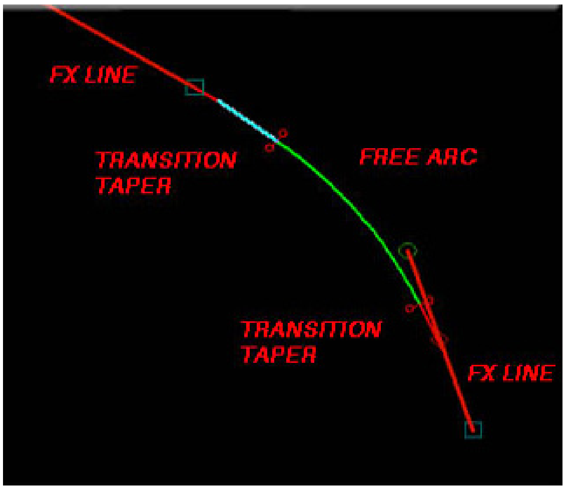
Set button

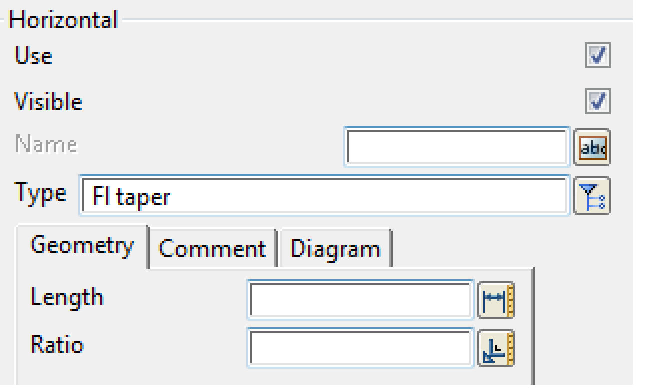
the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

Floating Transition: Taper

The *Floating Transition* consists of.





Example:
The Floating Taper become fully defined when it is placed between two fixed lines

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Length	length box		Point to point, String to point
Ratio	measure box		available measures

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

Set button

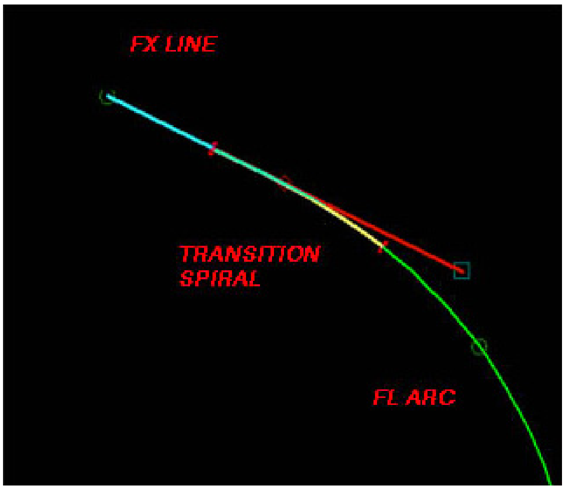
the **Set** *button must be clicked for the information for this Part to be used.*

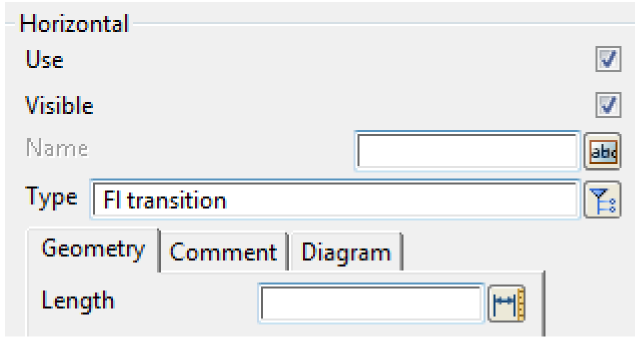
Same As button

after clicking the **Same As** *button, an existing HIP can be selected from the screen and its information piped into the appropriate fields in the panel.*

Floating Transition: Known Length

The *Floating Transition* consists of a **transition curve** of **known length**.





Example:
The Floating Transition become fully defined when it is placed between a fixed line and a floating arc

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Field Description	Type	Defaults	Pop-Up
Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Field Description	Type	Defaults	Pop-Up
Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Field Description	Type	Defaults	Pop-Up
Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Field Description	Type	Defaults	Pop-Up
Length	length box		Measure Length

the length of the transition. This must be non zero.

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

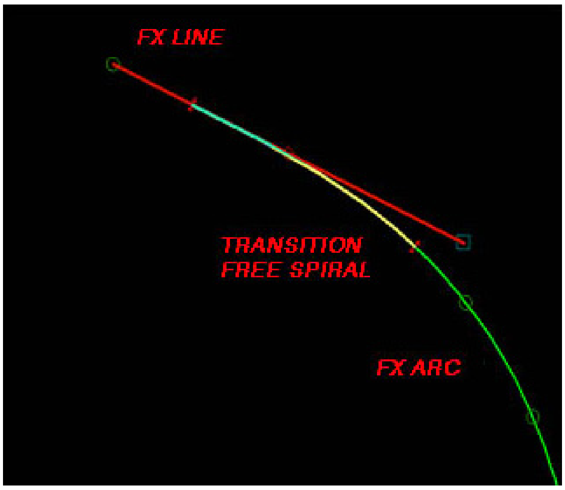
Field Description	Type	Defaults	Pop-Up
Set	button		

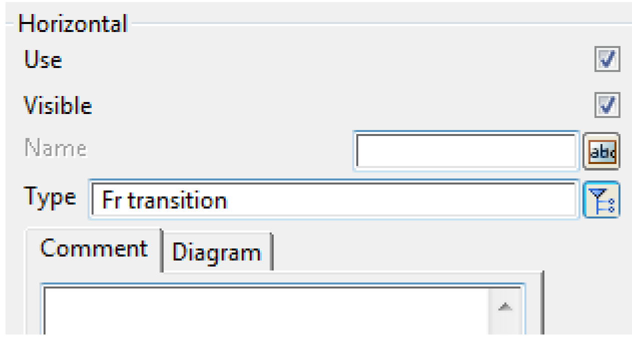
the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

Free Transition: Unconstrained

The *Free Transition* is unconstrained.





Example:
The Free Transition become fully defined when it is placed between a fixed line and a fixed arc

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Comment tab

the text typed into the text box is stored as a comment for the Part.

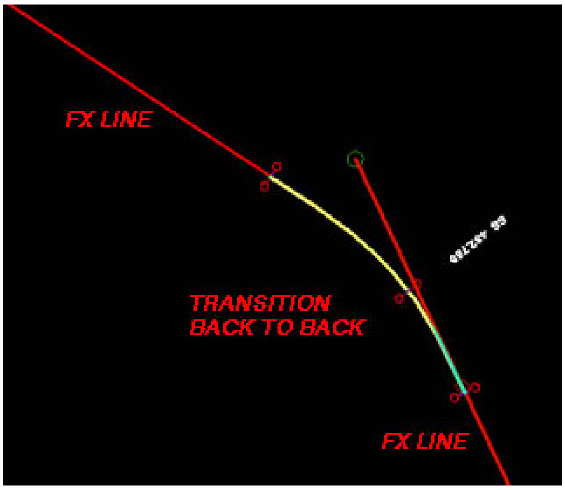
Note: *If a comment exists, a* **#** *is placed after the type of the part in the Horizontal Parts list.*

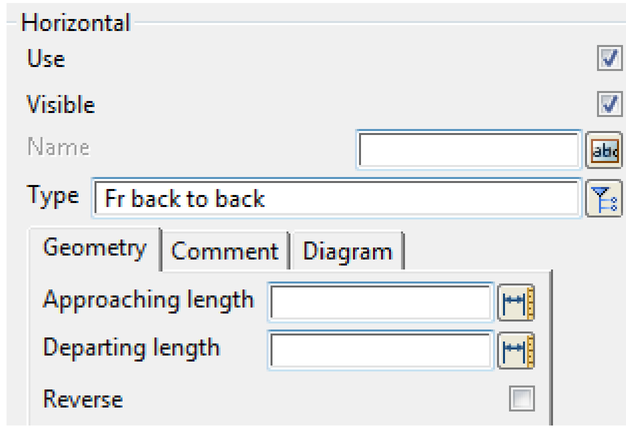
Set	button		

the **Set** *button must be clicked for the information for this Part to be used.*

Same As	button		

Free Transition: Back to Back Transitions of Known Lengths Between Lines
The *Free Transition* consists of a **two back to back transitions** of **known lengths between two lines.** The transitions can be the same hand or reversed.





Example:
The Free Transition become fully defined when it is placed between two fixed lines

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Field Description	Type	Defaults	Pop-Up
Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Field Description	Type	Defaults	Pop-Up
Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Field Description	Type	Defaults	Pop-Up
Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab
the geometrical information defining the Part

Field Description	Type	Defaults	Pop-Up
Approaching length	measure box		Measure Length

the length of the leading transition curve at the start of the Floating Transition. This must be non zero. This will be tangential to the Previous Part.

Field Description	Type	Defaults	Pop-Up
Departing length	measure box		Measure Length

the length of the trailing transition curve at the end of the Floating Transition. This must be non zero. This will be tangential to the Next Part.

Reverse tick box not ticked

if ticked, transitions of opposite hands are used

Comment tab

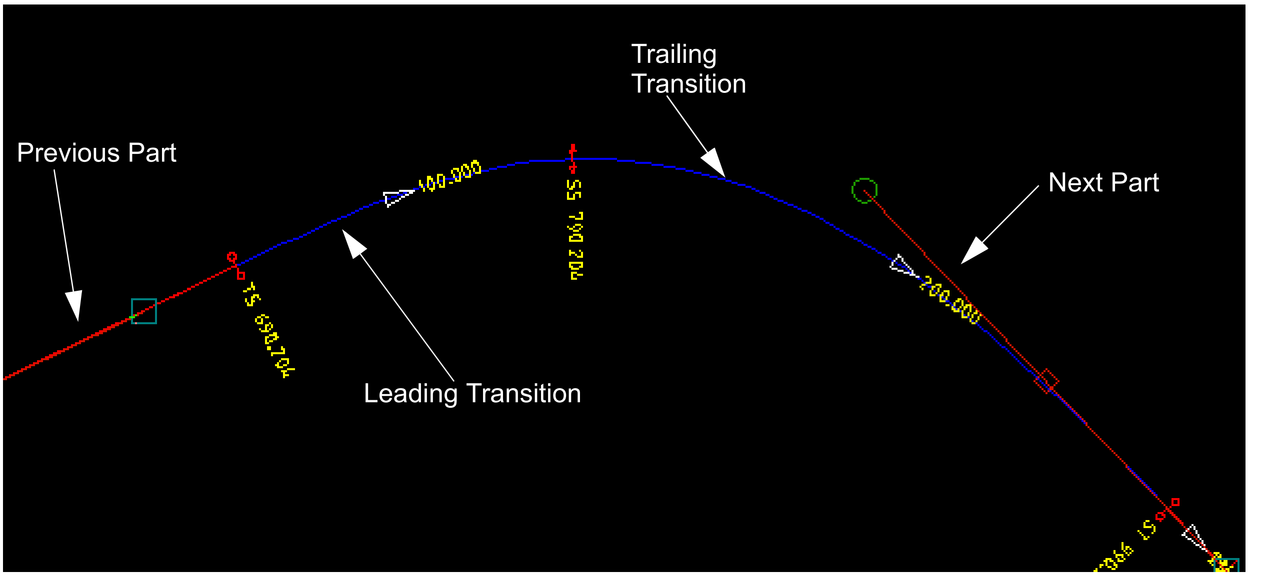
the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

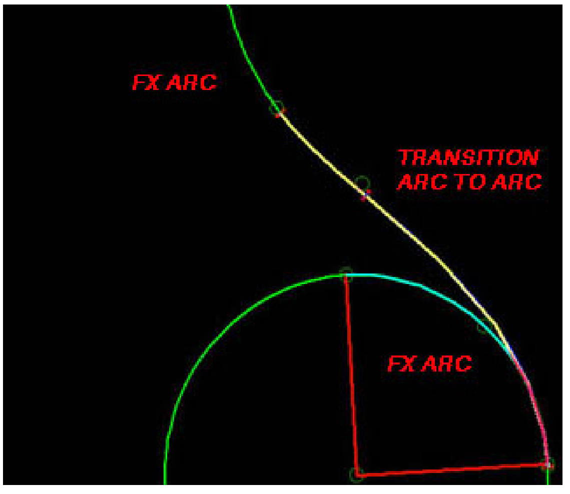
Set button

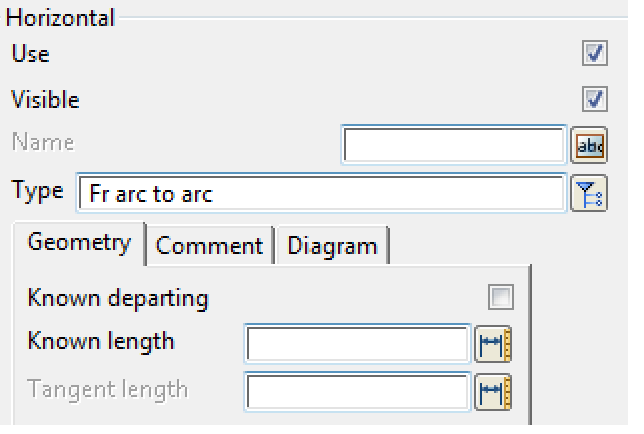
the **Set** *button must be clicked for the information for this Part to be used.*

Same As button



Free Transition: Back to Back Transitions Between Two Arcs
The *Free Transition* consists of a **two back to back transitions** between two **arcs**. The **length** of **one transition** is needed. The transitions can be the same hand or reversed.





Example:
The Free Transition become fully defined when it is placed between two fixed arcs.

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Known departing	tick box	not ticked	

if ticked,

If not ticked,

Known length	length box		Point to point, String to point

Tangent length length box Point to point, String to point

Comment tab

the text typed into the text box is stored as a comment for the Part.

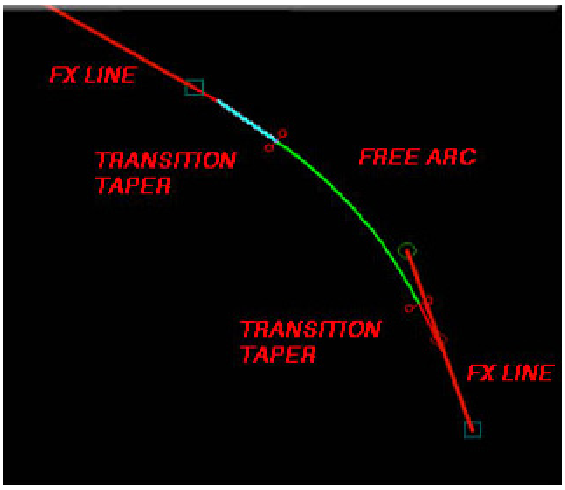
Note: *If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

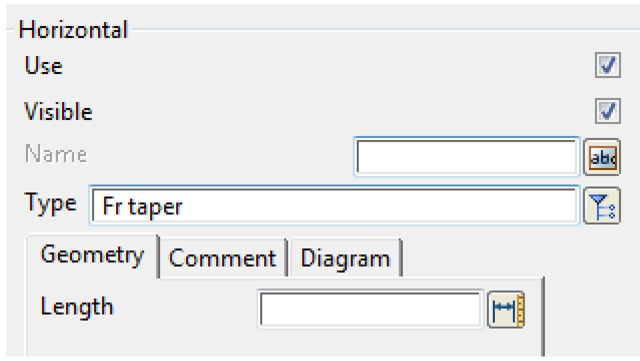
Set button

the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

Free Taper: Taper of Known Length
The *Free Transition* consists of a **taper** of **known length**.





Example

The Free Transition Taper becomes fully defined when it is place

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Length	length box		Point to point, String to point

Comment tab

the text typed into the text box is stored as a comment for the Part.

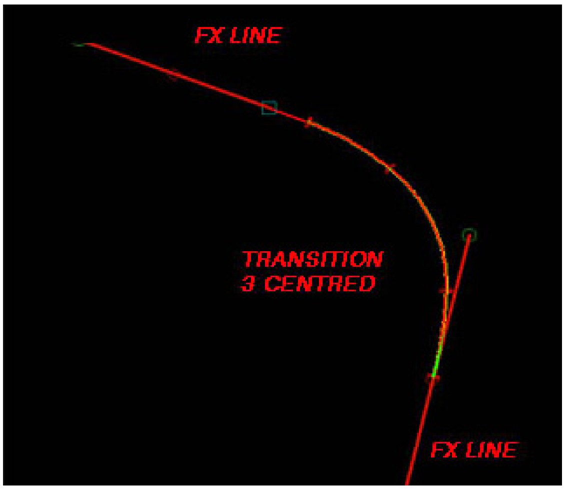
Note: *If a comment exists, a* **#** *is placed after the type of the part in the Horizontal Parts list.*

Set	button		

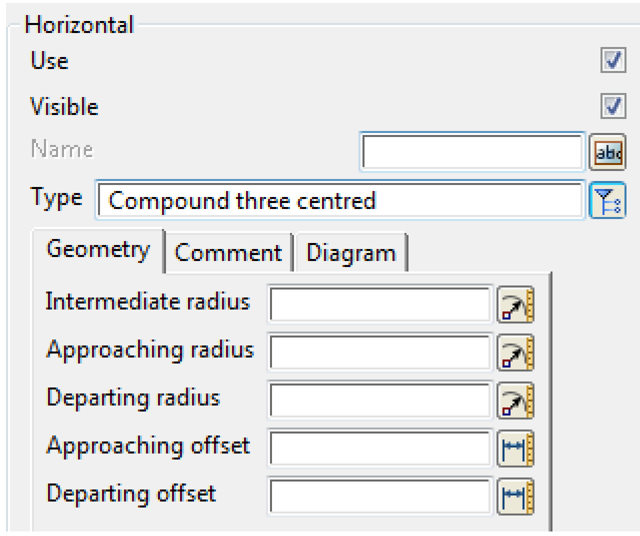
the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

Compound Transitions: Three Centre Curve
The *Compound Transition* consists of a **three centred curve**. That is, **three arcs** with **given radii** and an **approaching** and **departing offset**.



Example
The Compound Transition becomes fully defined when it is place between two fixed lines



Radius and Offset
-ve to the left of the direction of travel
+ve to the right of the direction of travel

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab
the geometrical information defining the Part

Intermediate radius	radius box		Point, String from point
Approaching radius	radius box		Point, String from point

Departing radius	radius box	Point, String from point
Approaching offset	length box	Point to point, String to point
Departing offset	length box	Point to point, String to point

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a # is placed after the type of the part in the Horizontal Parts list.*

Set button

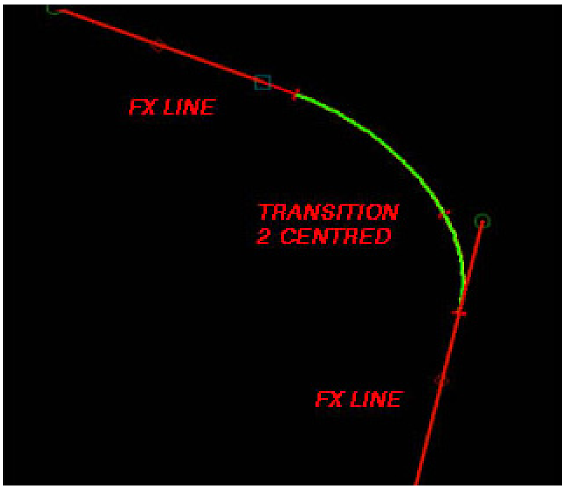
the **Set** *button must be clicked for the information for this Part to be used.*

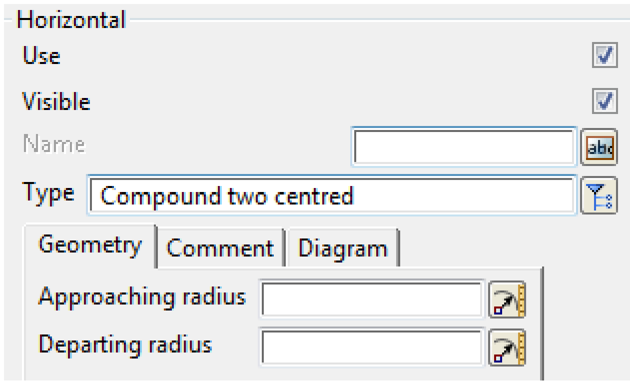
Same As button

after clicking the **Same As** *button, an existing HIP can be selected from the screen and its information piped into the appropriate fields in the panel.*

Compound Transition: Two Centred Curve

The *Compound Transition* consists of a **two centred curve**. That is, **two arcs** with **given radii**.





Example

The Compound Transition becomes fully defined when it is place between two fixed elements

Radius

-ve to the left of the direction of travel
+ve to the right of the direction of travel

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Field Description	Type	Defaults	Pop-Up
Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Field Description	Type	Defaults	Pop-Up
Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Field Description	Type	Defaults	Pop-Up
Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Field Description	Type	Defaults	Pop-Up
Approaching radius	radius box		Point, String from point
Departing radius	radius box		Point, String from point

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

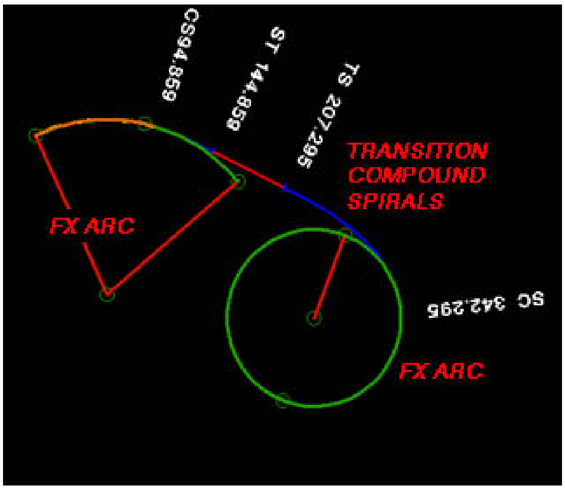
Set button

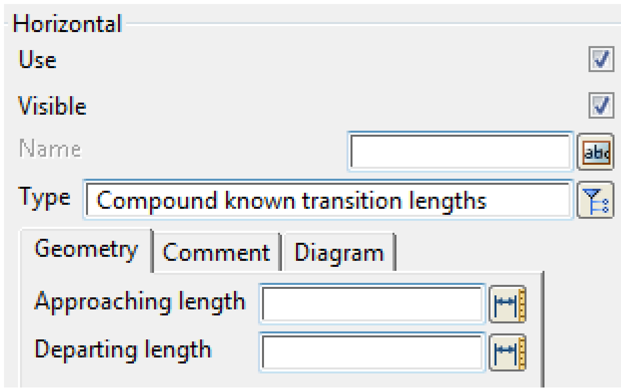
the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

after clicking the **Same As** *button, an existing HIP can be selected from the screen and its information piped into the appropriate fields in the panel.*

Compound Transition: Two Transition Curves of Known Lengths
The *Compound Transition* consists of a **two transition curves** of **known lengths**.





Example
The Compound Transition becomes fully defined when it is place between two fixed elements

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab
the geometrical information defining the Part

Approaching length	length box		Point to point, String to point
Departing length	length box		Point to point, String to point

Comment tab
the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

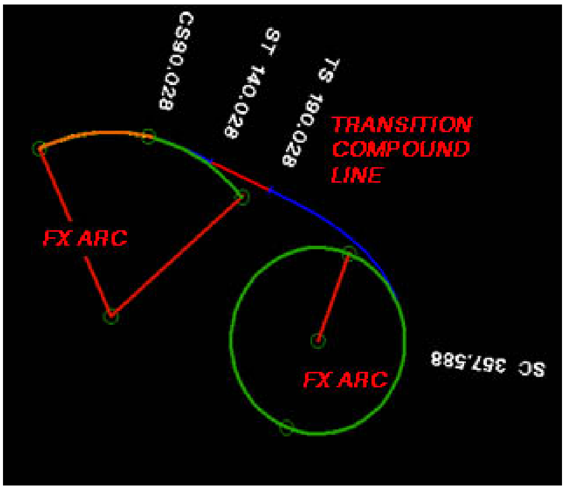
Set button

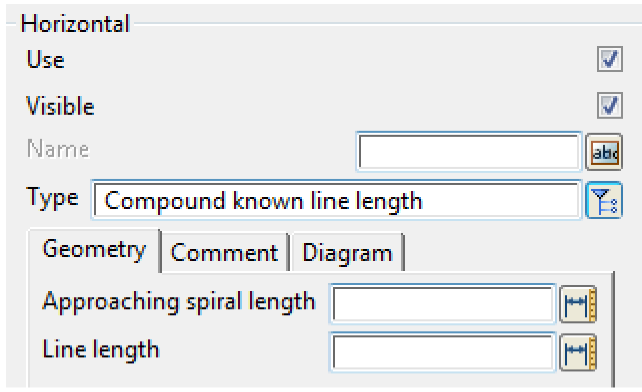
the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

after clicking the **Same As** *button, an existing HIP can be selected from the screen and its information piped into the appropriate fields in the panel.*

Compound Transition: Two Transitions Separated by Line of Known Length
The *Compound Transition* consists of a **approaching transition curves** of **known length** followed by a **line** of **known length** and then a **departing transition curve** with a **calculated length**.





Example
The Compound Transition becomes fully defined when it is place between two fixed elements. The second transition curve length is calculated.

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab
the geometrical information defining the Part

Approaching spiral length	length box		Point to point, String to point
Line length	length box		Point to point, String to point

Comment tab
the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

Set button

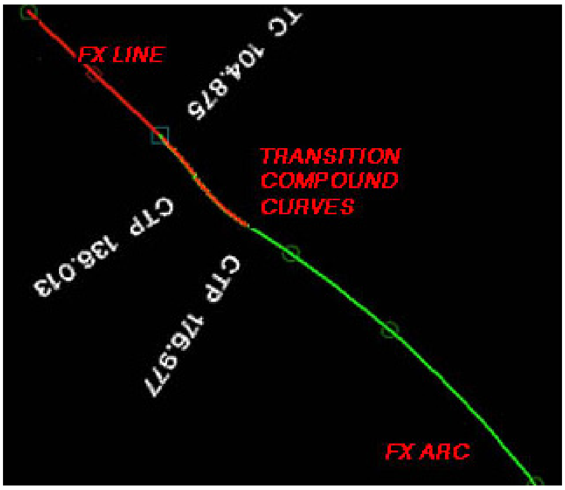
the **Set** *button must be clicked for the information for this Part to be used.*

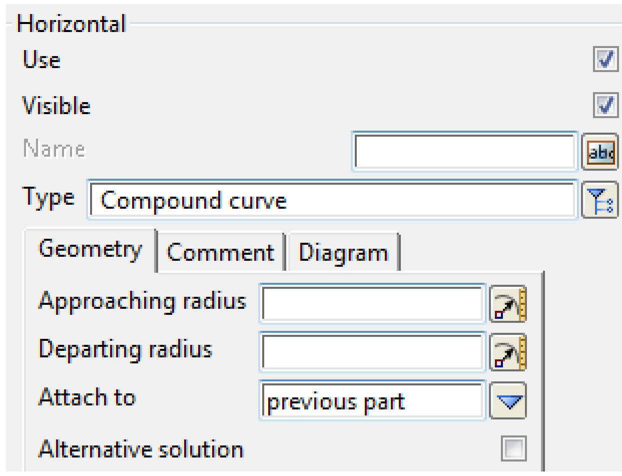
Same As button

after clicking the **Same As** *button, an existing HIP can be selected from the screen and its information piped into the appropriate fields in the panel.*

Compound Transition: Two Arcs of Known Radii

The *Compound Transition* consists of **two arcs** of **known radii.**





Example

The Compound Transition between a fixed line and a fixed arc.

Radius

-ve to the left of the direction of travel
+ve to the right of the direction of travel

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Field Description	Type	Defaults	Pop-Up
Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Field Description	Type	Defaults	Pop-Up
Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Field Description	Type	Defaults	Pop-Up
Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Field Description	Type	Defaults	Pop-Up
Approaching radius	radius box		Point, String from point
Departing radius	radius box		Point, String from point
Attach to	choice box		previous part, next part
Alternative solution	tick box	not ticked	

Comment tab

the text typed into the text box is stored as a comment for the Part.

***Note:** If a comment exists, a* # *is placed after the type of the part in the Horizontal Parts list.*

Set button

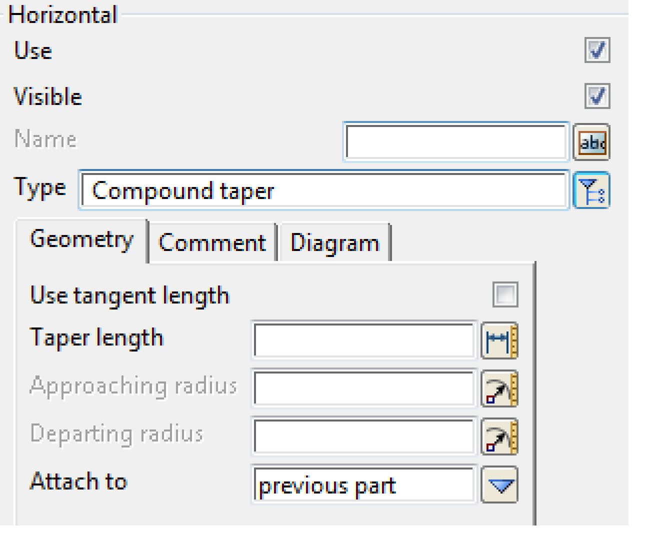
the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

after clicking the **Same As** *button, an existing HIP can be selected from the screen and its information piped into the appropriate fields in the panel.*

Compound Transition: Two Arcs of Known Radii with a Taper

The *Compound Transition* consists of



Example

Radius
-ve to the left of the direction of travel
+ve to the right of the direction of travel

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the horizontal geometry.
If not ticked, the Part is not used in the horizontal geometry.

Field Description	Type	Defaults	Pop-Up
Visible	tick box	ticked	

if ticked, then the Part is drawn in the horizontal geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Field Description	Type	Defaults	Pop-Up
Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Field Description	Type	Defaults	Pop-Up
Type	choice box		types of horizontal parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Field Description	Type	Defaults	Pop-Up
Use tangent length	tick box	not ticked	

if ticked,

If not ticked,

Field Description	Type	Defaults	Pop-Up
Taper length	length box		Point to point, String to point

Approaching radius	radius box	Point, String from point
Departing radius	radius box	Point, String from point
Attach to	choice box	previous part, next part

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a # is placed after the type of the part in the Horizontal Parts list.*

Set button

the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

after clicking the **Same As** *button, an existing HIP can be selected from the screen and its information piped into the appropriate fields in the panel.*

21.9.4.5 Horizontal Computators

Not yet documented.

21.9.5 Definitions of Super Alignment Vertical Parts

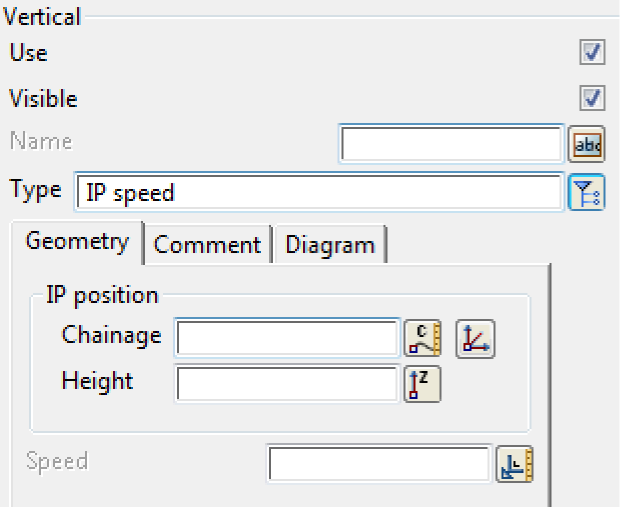
See 21.9.5.1 Vertical IPs
See 21.9.4.2 Horizontal Lines

21.9.5.1 Vertical IPs

See VIP: Speed
See VIP: K-value
See VIP: Radius
See VIP: Length:
See VIP: Asymmetric

VIP: Speed

A vertical intersection point (VIP) is created and the sizes of the parabolas for the VIP come from the Design Template of the super alignment.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.
If not ticked, the Part is not used in the vertical geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type choice box types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Chainage, Height

the (chainage,height) coordinates for the VIP can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons.*

Speed

if **Speed** *is not blank, then that is the speed used when looking up the Design Template for the VIP*

If **Speed** *is left blank, then the* **Design Speed** *for the super alignment in used when looking up the Design Template.*

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* **#** *is placed after the type of the part in the Vertical Parts list.*

Set button

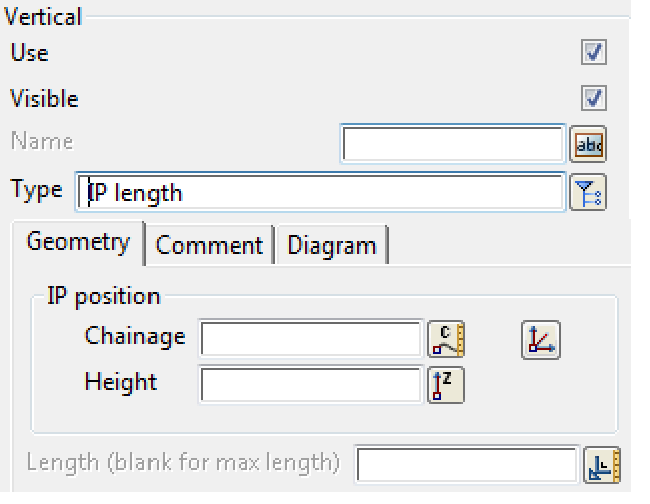
the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

after clicking the **Same As** *button, an existing VIP can be selected from the screen and its information piped into the appropriate fields in the panel.*

VIP: Length:

A vertical intersection point (VIP) is created and either length of the parabola on the VIP is given, or the maximum allowable parabola is used.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.
If not ticked, the Part is not used in the vertical geometry.

Field Description	Type	Defaults	Pop-Up
Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.
If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Field Description	Type	Defaults	Pop-Up
Name	text box	blank	

if not blank, then the Part is given this name.
If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Horizontal Parts list.*

Field Description	Type	Defaults	Pop-Up
Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Chainage, Height

the (chainage,height) coordinates for the VIP can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons.*

Length (blank for max length)

the length used for the parabola on the VIP. This field can not be left blank.

If zero then a parabola of maximum length is used on the VIP.

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* **#** *is placed after the type of the part in the Vertical Parts list.*

Set button

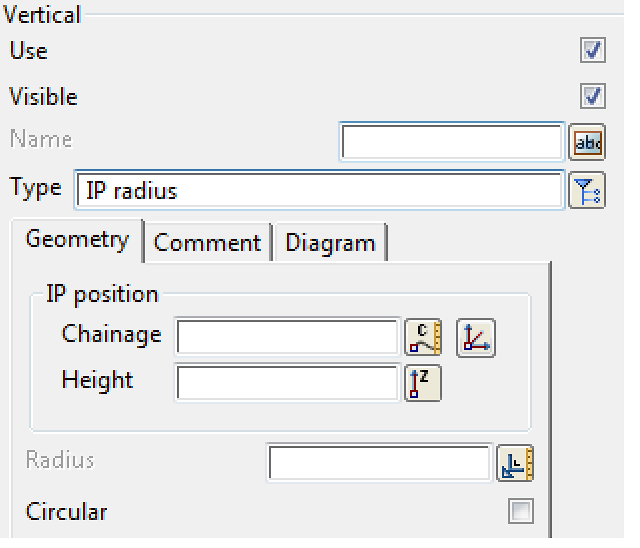
the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

after clicking the **Same As** *button, an existing VIP can be selected from the screen and its information piped into the appropriate fields in the panel.*

VIP: Radius

A vertical intersection point (VIP) is created and the effective radius for the parabola on the VIP is given.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Chainage, Height

the (chainage,height) coordinates for the VIP can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons.*

Radius

if not blank or non zero, the radius entered is used for the parabola/arc on the VIP.

If blank or zero then no parabola/arc is used on the VIP.

Circular	tick box	not ticked	

if ticked, then the curve on the VIP is an arc

If not ticked, then the curve on the VIP is a parabola.

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Vertical Parts list.*

Set button

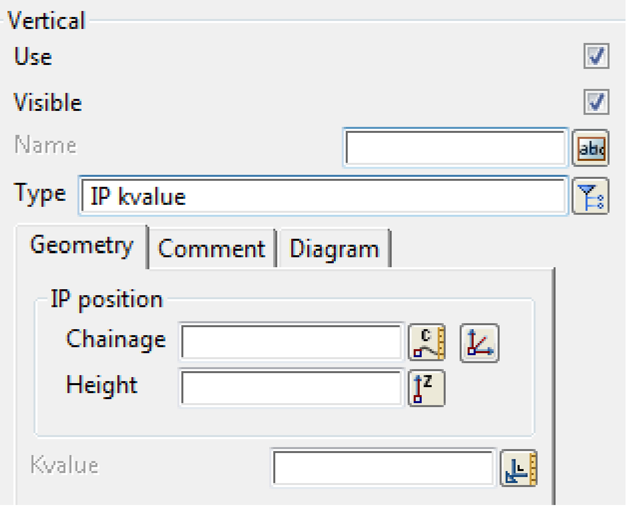
the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

after clicking the **Same As** *button, an existing VIP can be selected from the screen and its information piped into the appropriate fields in the panel.*

VIP: K-value

A vertical intersection point (VIP) is created and the K value for the parabola on the VIP is given.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Chainage, Height

the (chainage,height) coordinates for the VIP can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons.*

K-value	measure box		available measures

the K value to use for the parabola on the VIP

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* **#** *is placed after the type of the part in the Vertical Parts list.*

Set button

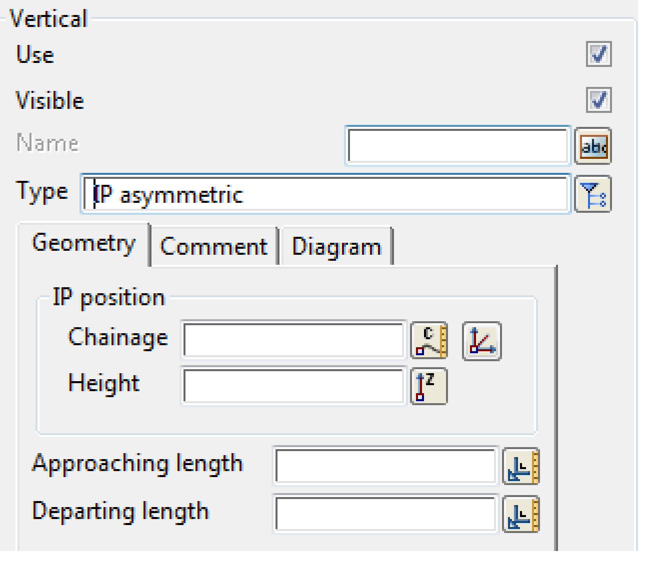
the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

after clicking the **Same As** *button, an existing VIP can be selected from the screen and its information piped into the appropriate fields in the panel.*

VIP: Asymmetric

A vertical intersection point (VIP) is created with an asymmetric parabola on the VIP. The two parabolic lengths are given for the asymmetric parabola.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Chainage, Height

the (chainage,height) coordinates for the VIP can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons.*

Approaching length	measure box		available measures

the parabolic length of the first parabola of the asymmetric parabola. This parabola is tangential to the Previous Part.

Departing length	measure box		available measures

the parabolic length of the second parabola of the asymmetric parabola. This parabola is tangential to *the Next Part.*

Comment tab

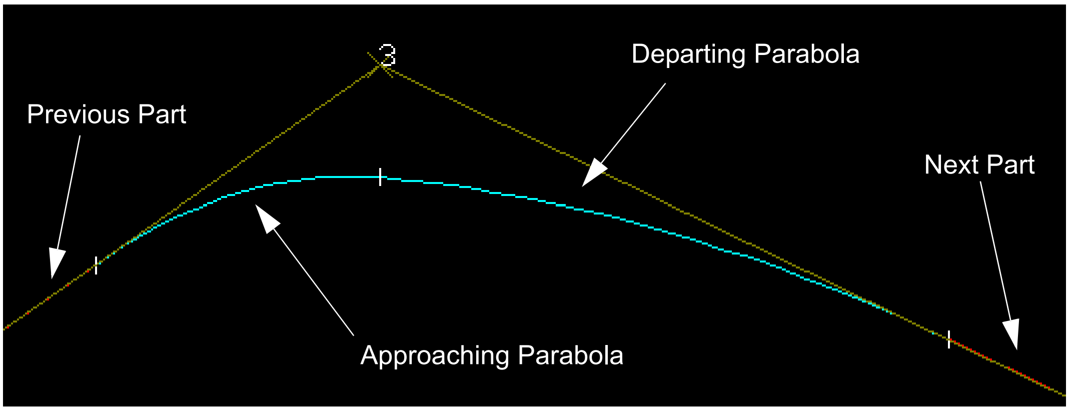
the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Vertical Parts list.*

Set button

the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

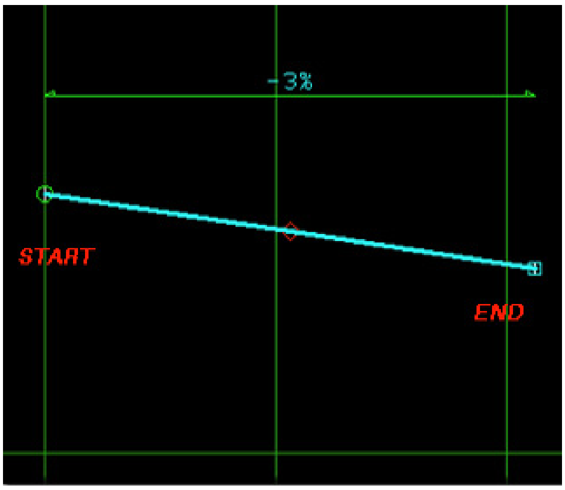


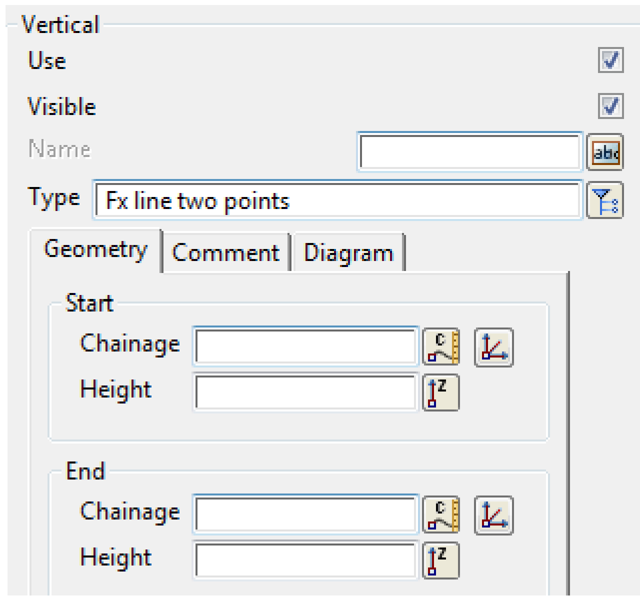
21.9.5.2 Vertical Lines

See Fixed Line: Two Points
See Fixed Line: Known Point and Grade
See Floating Line: Known Point
See Floating Line: Known Grade
See Floating Line: Known End and Nominal Length
See Free line - No Constraints

Fixed Line: Two Points

A *Fixed Line* is created by specifying two known points as the start and end points of the line.





The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Field Description	Type	Defaults	Pop-Up
Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Field Description	Type	Defaults	Pop-Up
Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Field Description	Type	Defaults	Pop-Up
Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Start Chainage, Height

the coordinates in (chainage,height) of the start of the line can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons.*

End Chainage, Height

the coordinates in (chainage,height) of the end of the line can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons.*

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Vertical Parts list.*

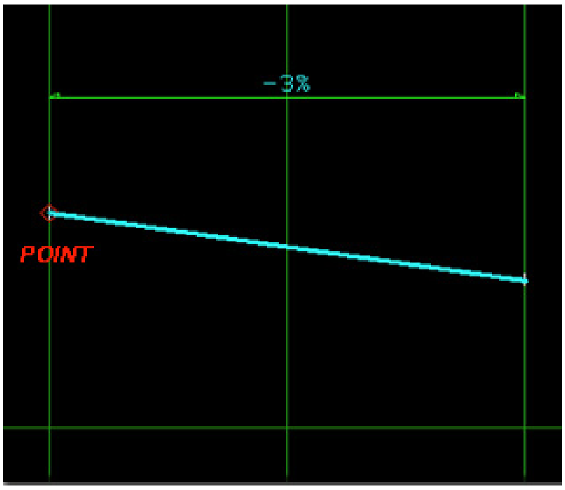
Set button

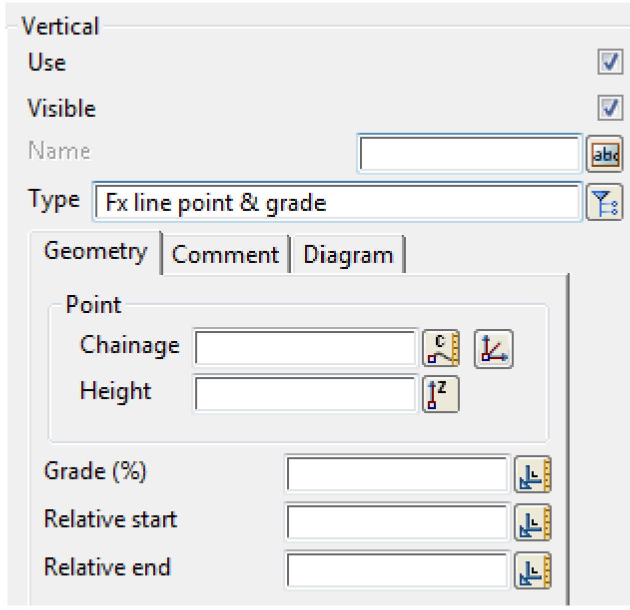
the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

Fixed Line: Known Point and Grade

A *Fixed Line* is created by specifying a known point and a grade for the line.





The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Point Chainage, Height

the coordinates in (chainage,height) of the point that the line goes through can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons.*

Grade (%)	measure box		available measures

the grade (in percent) of the line through the point

Relative start measure box available measures

the chainage distance, relative to the given point, to start the line at. A negative value is to the left of the point, and a positive value is to the right of the point.

Relative end measure box available measures

the chainage distance, relative to the given point, to end the line at. A negative value is to the left of the point, and a positive value is to the right of the point.

Comment tab

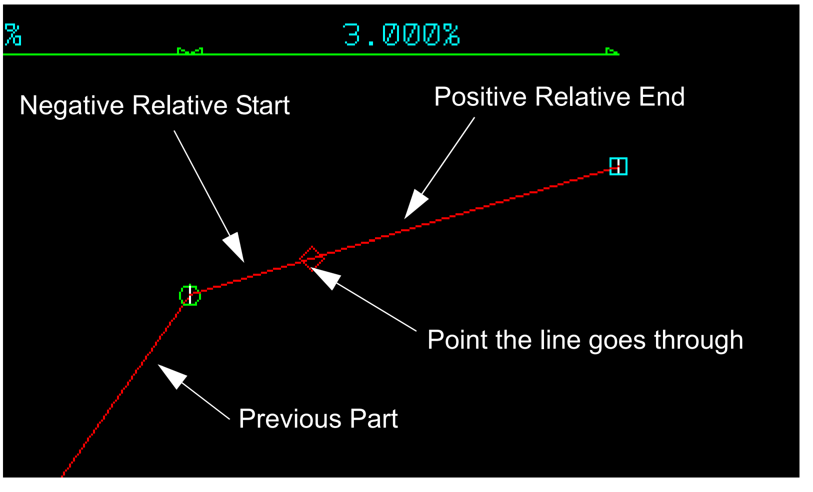
the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Vertical Parts list.*

Set button

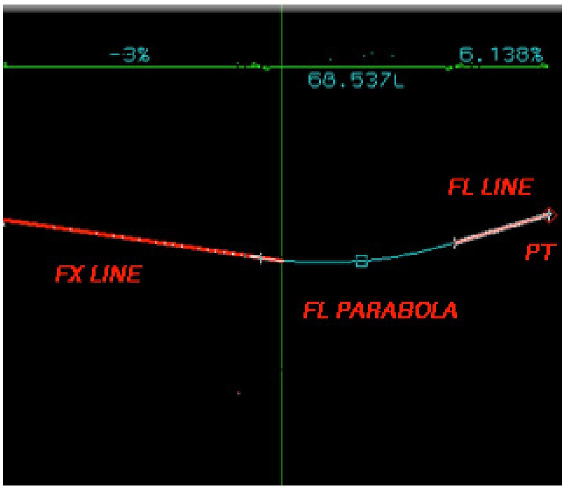
the **Set** *button must be clicked for the information for this Part to be used.*

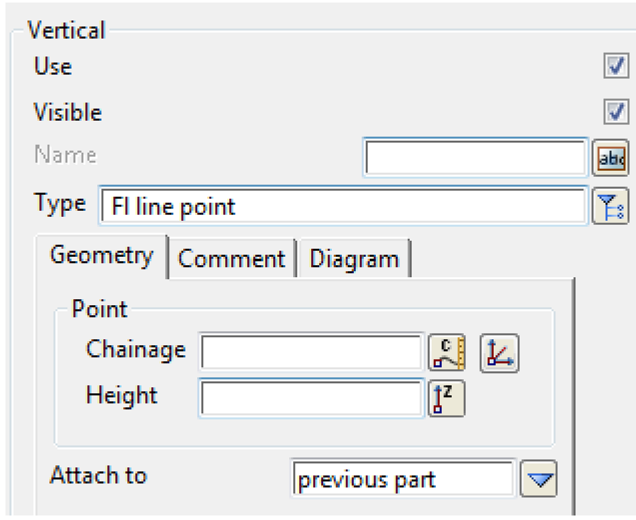
Same As button



Floating Line: Known Point

The *Floating Line* line passes through a **known point** but the grade is unknown.





Example

Floating Line attached to the previous parabola

The fields and buttons used in the panel have the following functions.

Field Description   Type   Defaults   Pop-Up

Use   tick box   ticked

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible   tick box   ticked

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name   text box   blank

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type   choice box   types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Chainage, Height

the (chainage,height) coordinates that the line passes through can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons.*

Attach to   choice box   previous part   previous part, next part

the Part that the Floating Line is attached to and tried to make tangential to

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Vertical Parts list.*

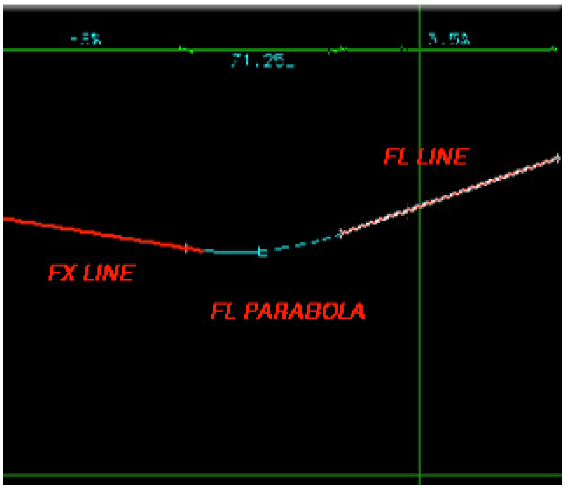
Set button

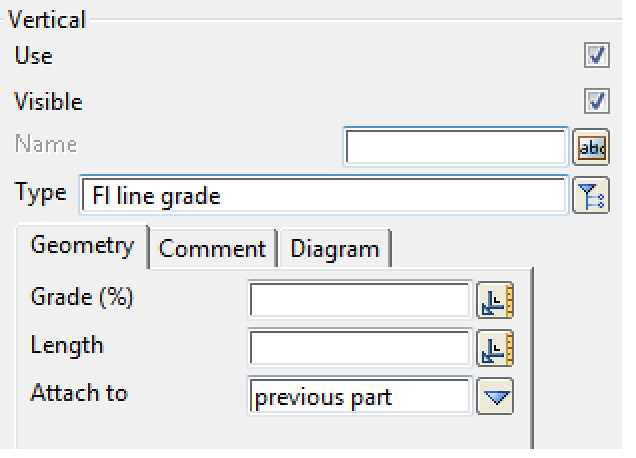
the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

Floating Line: Known Grade

The *Floating Line* line has a **known grade** but no point on the line is known.





Example

Floating Line attached to the previous parabola

The fields and buttons used in the panel have the following functions.

Field Description   Type   Defaults   Pop-Up

Use   tick box   ticked

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible   tick box   ticked

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name   text box   blank

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type   choice box   types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Grade (%)   measure box   available measures

the grade (in percent) of the line through the point

Length   measure box   available measures

nominal length of the line

Attach to   choice box   previous part   previous part, next part

the Part that the Floating Line is attached to and tried to make tangential to

Comment tab

the text typed into the text box is stored as a comment for the Part.

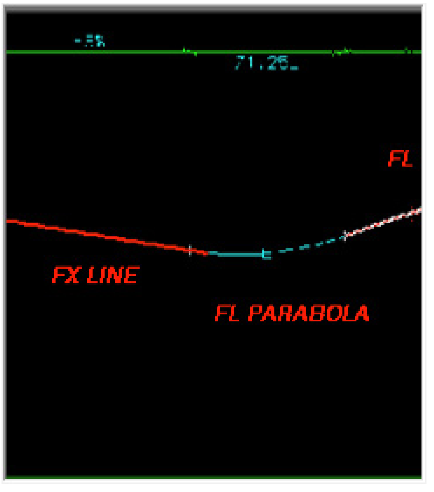
Note: *If a comment exists, a* # *is placed after the type of the part in the Vertical Parts list.*

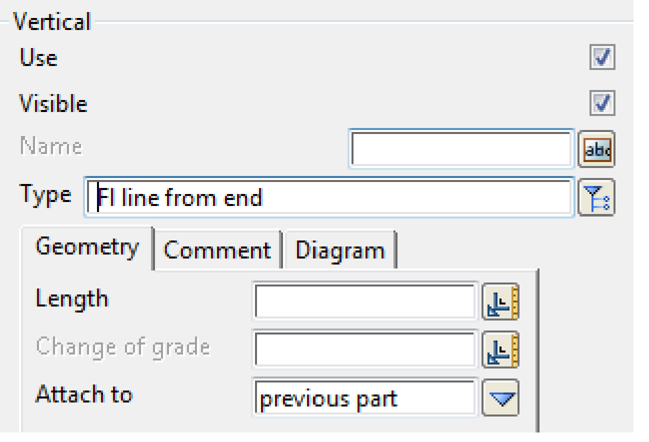
Set button

the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

Floating Line: Known End and Nominal Length
The *Floating Line* has a **known nominal length** and is attached tangentially to the end of previous part or the beginning of the next part.





The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Length	measure box		available measures

the nominal length of the Floating Line. Must be non zero.

Change of grade	measure box		available measures

LJG??? if non zero then the grade of the line is increased by **Change of grade** *(units of percent grade). It is no longer tangential to the* **Attach to** *part.*

Attach to	choice box		previous part, next part

the Part that the Floating Line is attached to, and made tangential to. If there is a **Change of grade**, *then after the line is calculated to be tangential, the* **Change of grade** *is added to the grade of the line.*

Comment tab

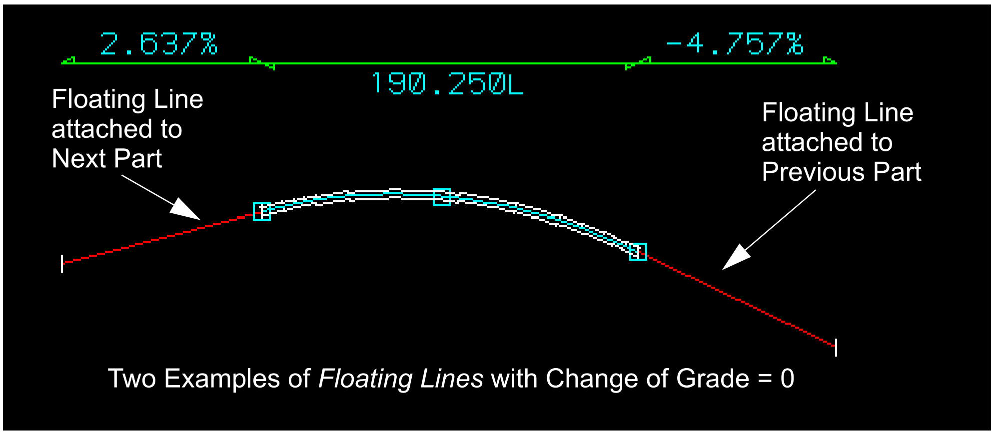
the text typed into the text box is stored as a comment for the Part.

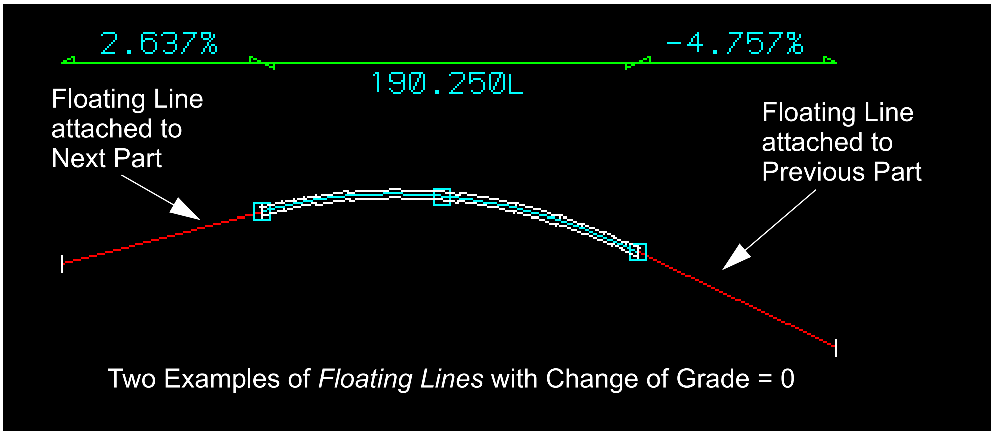
Note: *If a comment exists, a* # *is placed after the type of the part in the Vertical Parts list.*

Set button

the **Set** *button must be clicked for the information for this Part to be used.*

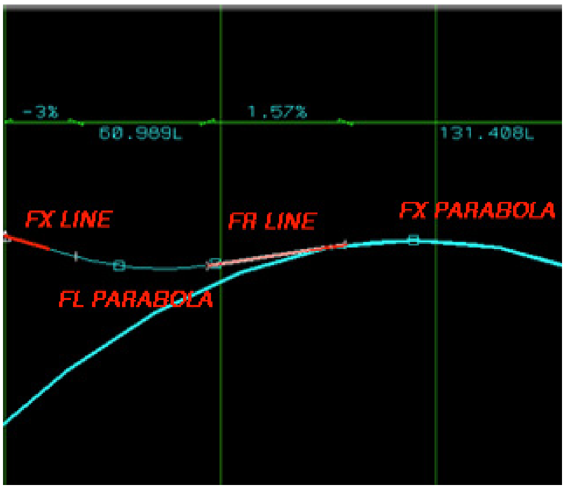
Same As button

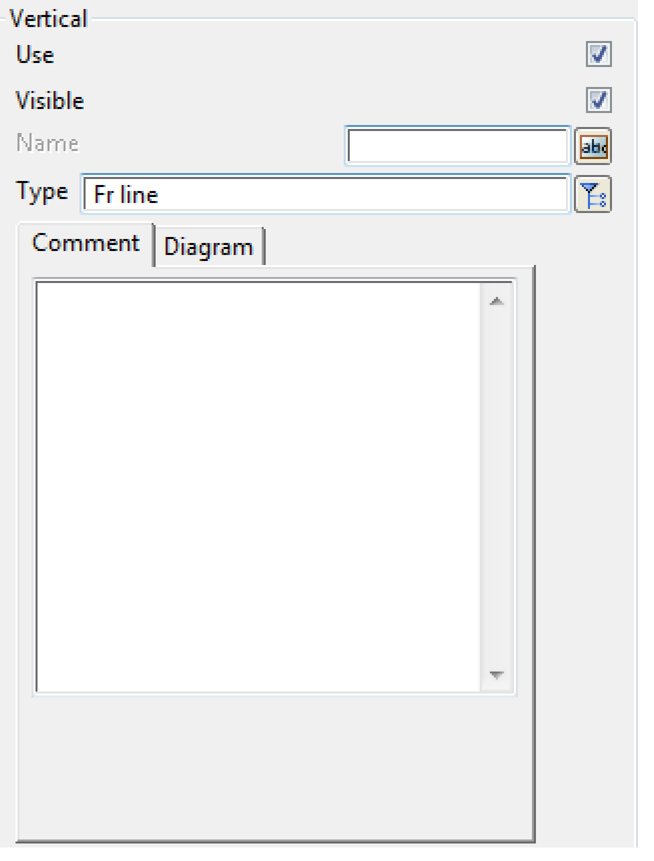




Free line - No Constraints

The *Free Line* is not constrained. It is determined by having to be tangential to the Parts at either end of the Line.





Example

Typical application is a Free Line tangent between two solved parts

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* **#** *is placed after the type of the part in the Vertical Parts list.*

Set button

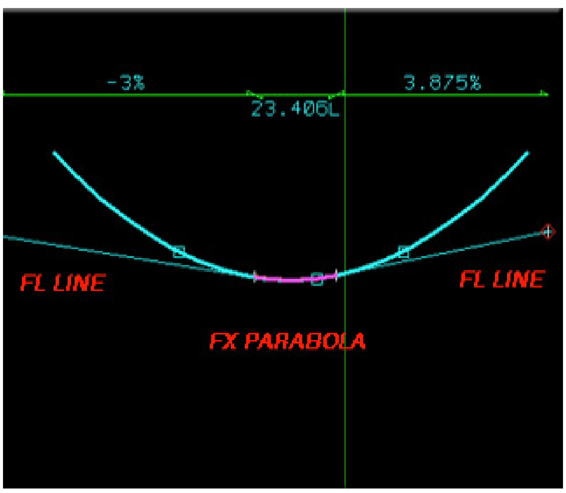
the **Set** *button must be clicked for the information for this Part to be used.*

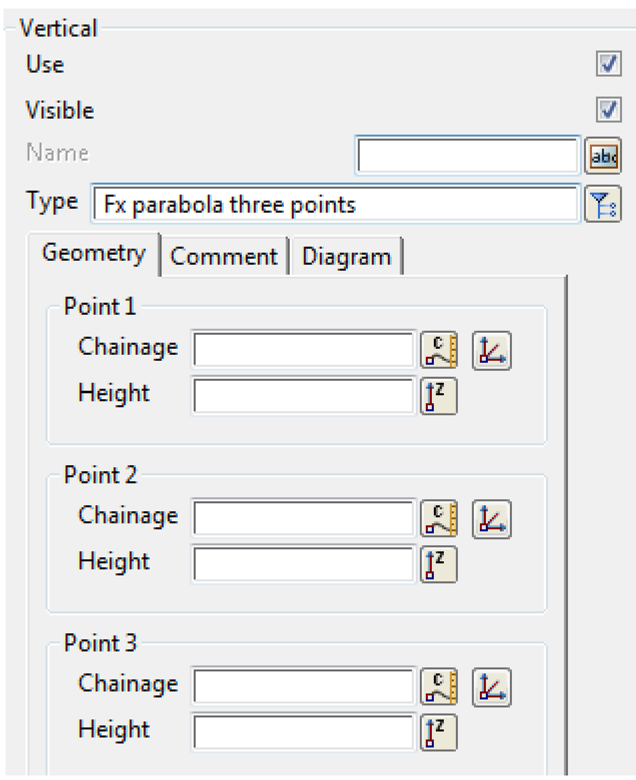
Same As button

21.9.5.3 Vertical Parabolas

See Fixed Parabola - Parabola Passes through Three Known Points
See Fixed Parabola: Known Apex and Effective Radius
See Floating Parabola: Through Two Points
See Floating Parabola: Known Apex
See Floating Parabola: Known Point and Effective Radius
See Floating Parabola: Known Point and Grade at that Point
See Floating Parabola: Known Point and K Value
See Floating Parabola: Known Length and Grade at End
See Floating Parabola: Known Radius and Nominal Length
See Floating Parabola: known Point and Known End
See Free Parabola: Known Point
See Free Parabola: Known Length
See Free Parabola: Know Effective Radius
See Free Parabola: Known Parabolic Length/100
See Free Parabola: Known K Value
See Free Parabola: Fit with Maximum Length
See Free Asymmetric Parabola: Two Known Lengths
See Free Compound Parabola: Two Parabolas with Optional Total Parabolic Length

Fixed Parabola - Parabola Passes through Three Known Points
The *Fixed Parabola* is fully defined by going through three known points.





The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Point 1 Chainage, Height

the (chainage,height) coordinates for the first point the parabola goes through can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons.*

Point 2 Chainage, Height

the (chainage,height) coordinates for the second point the parabola goes through can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons.*

Point 3 Chainage, Height

the (chainage,height) coordinates for the third point the parabola goes through can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons.*

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* **#** *is placed after the type of the part in the Vertical Parts list.*

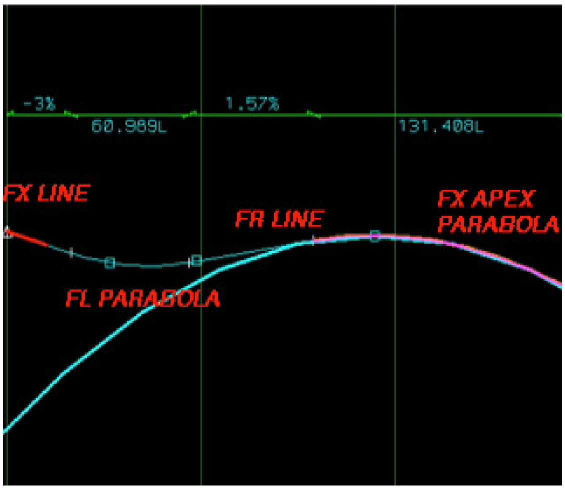
Set	button		

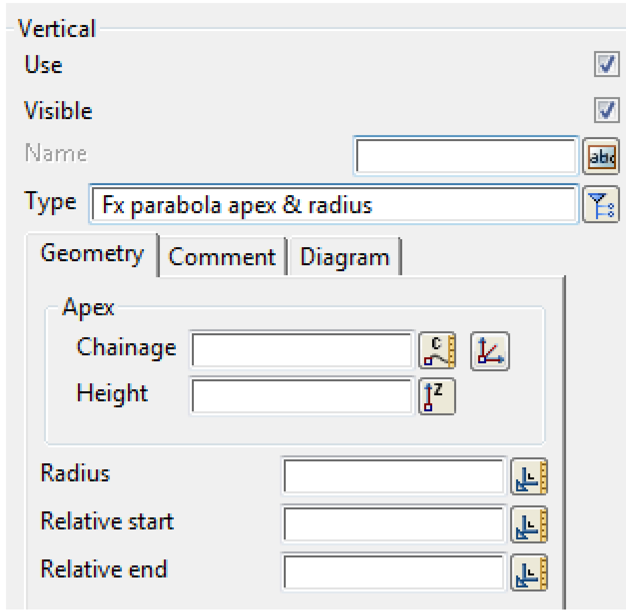
the **Set** *button must be clicked for the information for this Part to be used.*

Same As	button		

Fixed Parabola: Known Apex and Effective Radius

The *Fixed Parabola* is fully defined by the known apex and effective radius of the parabola.





The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Field Description	Type	Defaults	Pop-Up
Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Field Description	Type	Defaults	Pop-Up
Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Field Description	Type	Defaults	Pop-Up
Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Apex Chainage, Height

the (chainage,height) coordinates for the apex of the parabola can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons.*

Field Description	Type	Defaults	Pop-Up
Radius	measure box		available measures

the effective radius of the parabola (this is 100 times the K value). Positive radius for a sag point (minimum) and a Negative radius for a crest point (maximum).

Relative start measure box available measures

the chainage distance, relative to the given apex, to start the parabola at. A negative value is to the left of the apex, and a positive value is to the right of the apex.

Relative end measure box available measures

the chainage distance, relative to the given apex, to end the parabola at. A negative value is to the left of the apex, and a positive value is to the right of the apex.

Comment tab

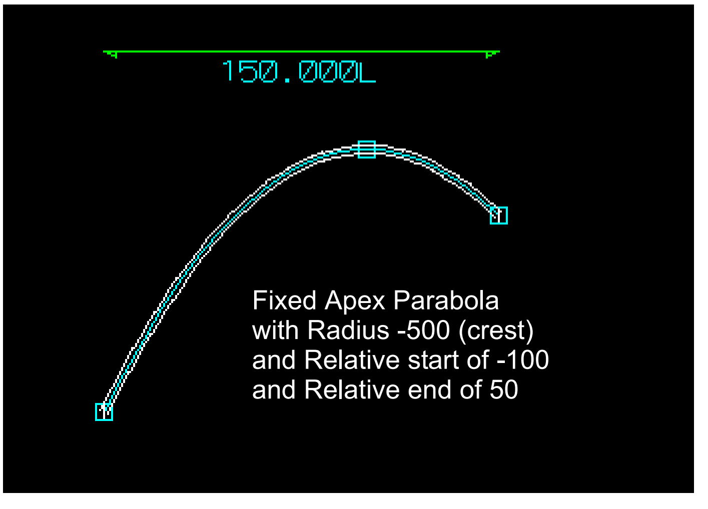
the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Vertical Parts list.*

Set button

the **Set** *button must be clicked for the information for this Part to be used.*

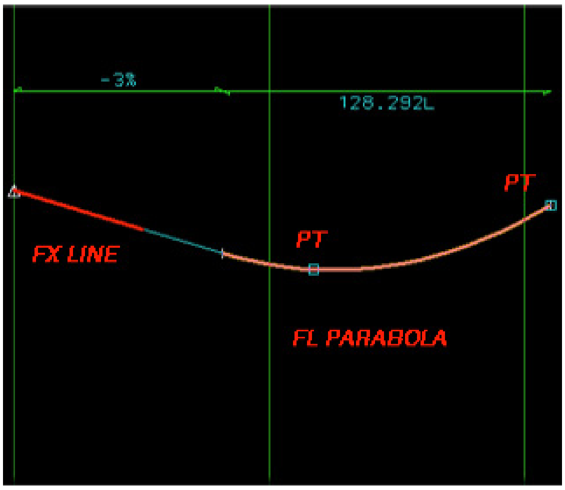
Same As button

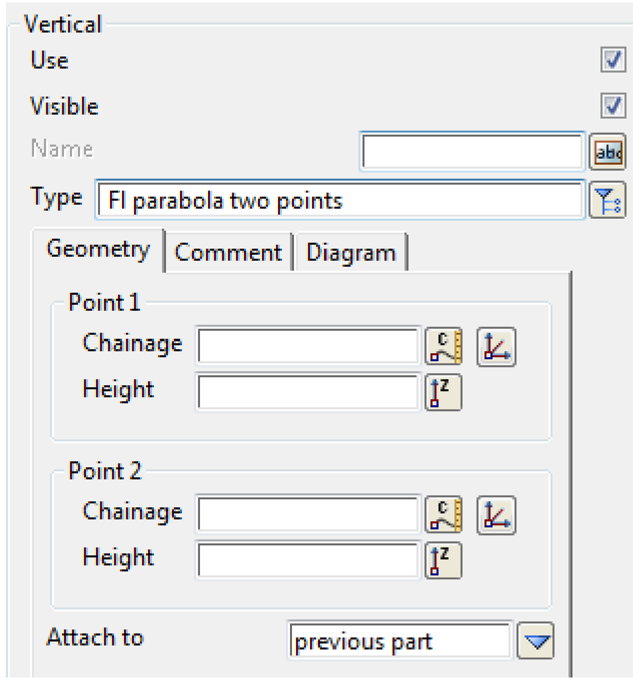


Floating Parabola: Through Two Points

This *Floating Parabola* passes through two known points. It is fully determined by being tangential to either the previous part or the following part.

It may not always be possible for the Floating Parabola to actually solve.





The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Field Description	Type	Defaults	Pop-Up
Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Field Description	Type	Defaults	Pop-Up
Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Field Description	Type	Defaults	Pop-Up
Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Point 1 Chainage, Height

the (chainage,height) coordinates for the first point the parabola goes through can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons.*

Point 2 Chainage, Height

the (chainage,height) coordinates for the second point the parabola goes through can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons.*

Attach to choice box previous part, next part

the Part that the Floating Parabola is attached to, and made tangential to.

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Vertical Parts list.*

Set button

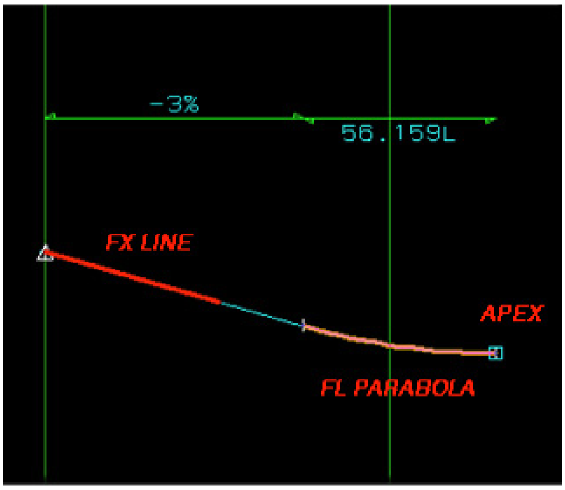
the **Set** *button must be clicked for the information for this Part to be used.*

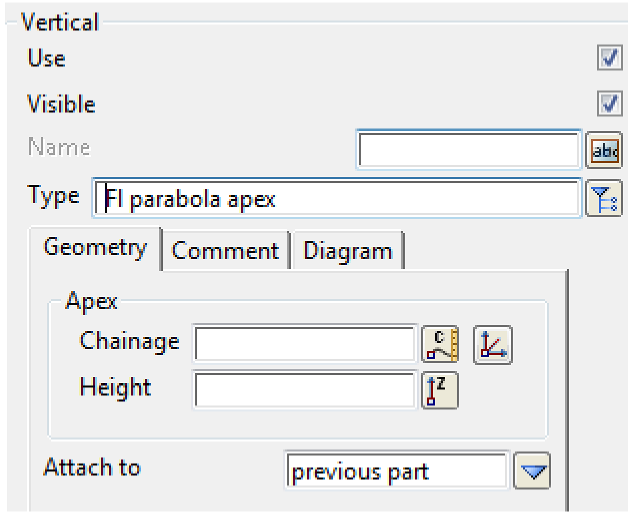
Same As button

Floating Parabola: Known Apex

This *Floating Parabola* has a known apex. It is fully determined by being tangential to either the previous part or the following part.

It may not always be possible for the Floating Parabola to solve.





The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Apex Chainage, Height

the (chainage,height) coordinates for the apex of the parabola can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons.*

Attach to	choice box	previous part	previous part, next part

the Part that the Floating Parabola is attached to, and made tangential to.

Comment tab

the text typed into the text box is stored as a comment for the Part.

***Note:** If a comment exists, a # is placed after the type of the part in the Vertical Parts list.*

Set button

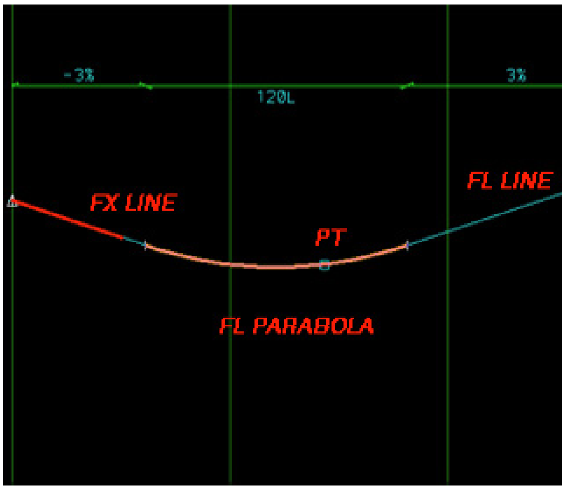
*the **Set** button must be clicked for the information for this Part to be used.*

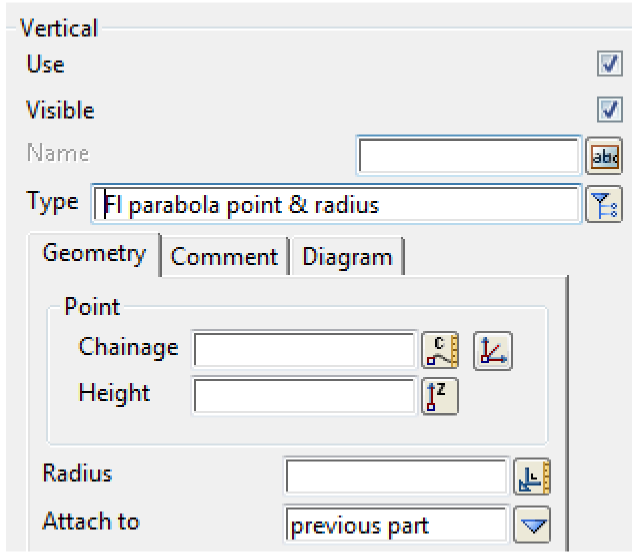
Same As button

Floating Parabola: Known Point and Effective Radius

This *Floating Parabola* goes through a known point and has a given effective radius. It is fully determined by being tangential to either the previous part or the following part.

It may not always be possible for the Floating Parabola to solve.





The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Chainage, Height

the (chainage,height) coordinates for the point on the parabola can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons.*

Radius	measure box		available measures

the effective radius of the parabola (this is 100 times the K value).

Attach to choice box previous part previous part, next part

the Part that the Floating Parabola is attached to, and made tangential to.

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Vertical Parts list.*

Set button

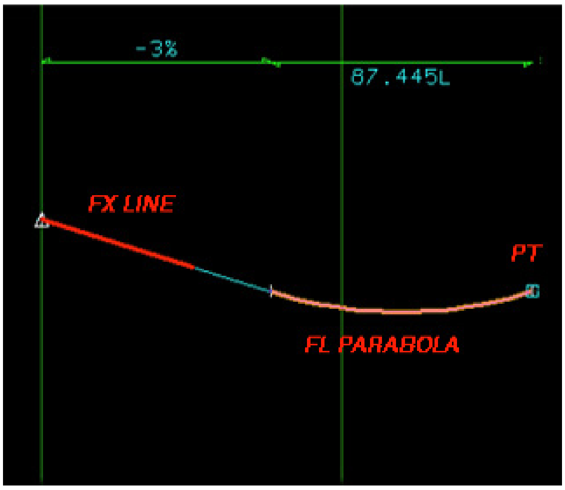
the **Set** *button must be clicked for the information for this Part to be used.*

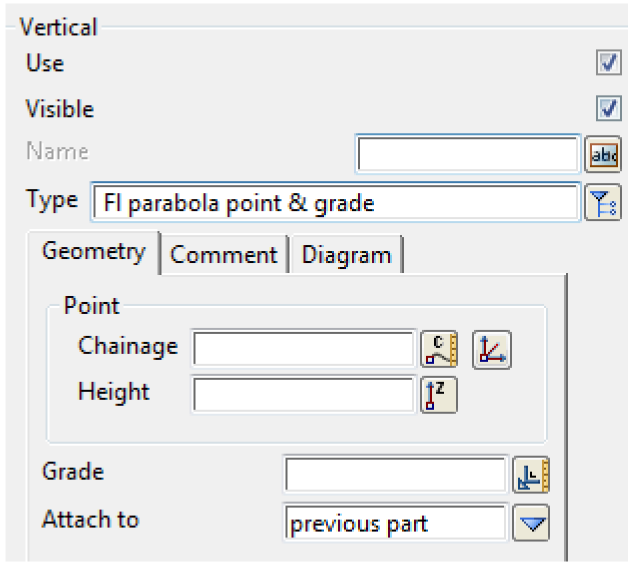
Same As button

Floating Parabola: Known Point and Grade at that Point

This *Floating Parabola* goes through a known point and has a given at the known point. It is fully determined by being tangential to either the previous part or the following part.

It may not always be possible for the Floating Parabola to solve.





The fields and buttons used in the panel have the following functions.

Field Description — Type — Defaults — Pop-Up

Use — tick box — ticked

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible — tick box — ticked

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name — text box — blank

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type — choice box — types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Chainage, Height

the (chainage,height) coordinates for the point on the parabola can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons.*

Grade — measure box — available measures

the grade of the parabola in units of percent grade

Attach to choice box previous part previous part, next part

the Part that the Floating Parabola is attached to, and made tangential to.

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a # is placed after the type of the part in the Vertical Parts list.*

Set button

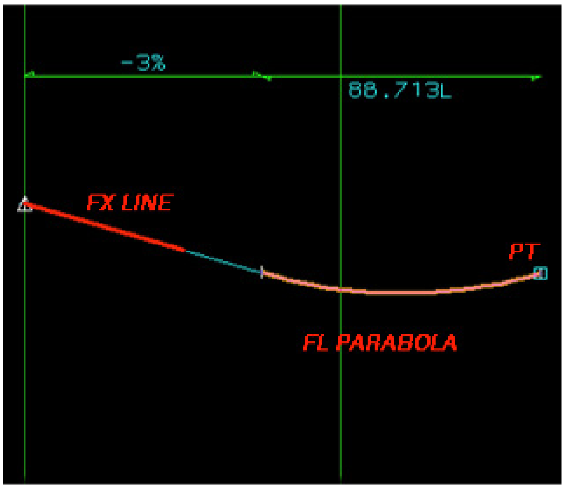
the **Set** *button must be clicked for the information for this Part to be used.*

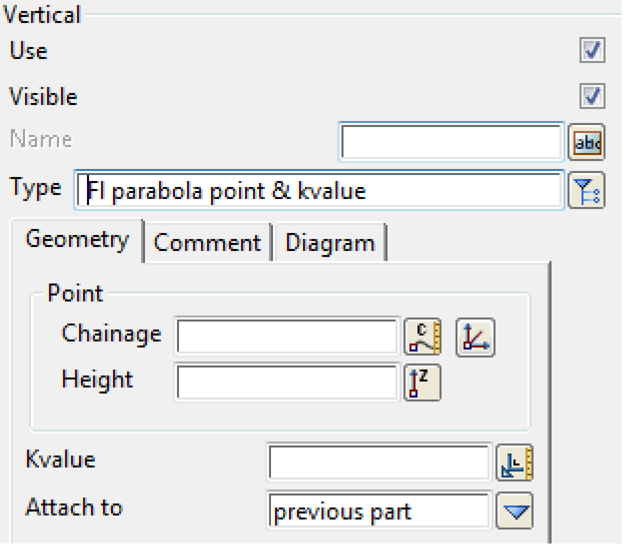
Same As button

Floating Parabola: Known Point and K Value

This *Floating Parabola* goes through a known point and has a given K value. It is fully determined by being tangential to either the previous part or the following part.

It may not always be possible for the Floating Parabola to solve.





The fields and buttons used in the panel have the following functions.

Field Description   Type   Defaults   Pop-Up

Use   tick box   ticked

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible   tick box   ticked

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name   text box   blank

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type   choice box   types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Chainage, Height

the (chainage,height) coordinates for the point on the parabola can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons.*

K value   measure box   available measures

the K value of the parabola

Attach to choice box previous part previous part, next part

the Part that the Floating Parabola is attached to, and made tangential to.

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a # is placed after the type of the part in the Vertical Parts list.*

Set button

the **Set** *button must be clicked for the information for this Part to be used.*

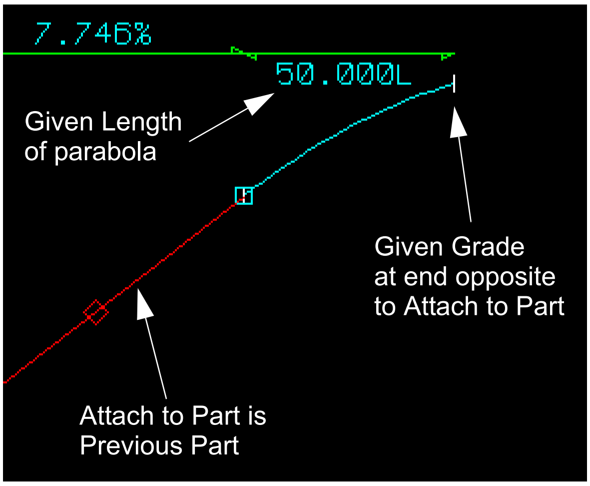
Same As button

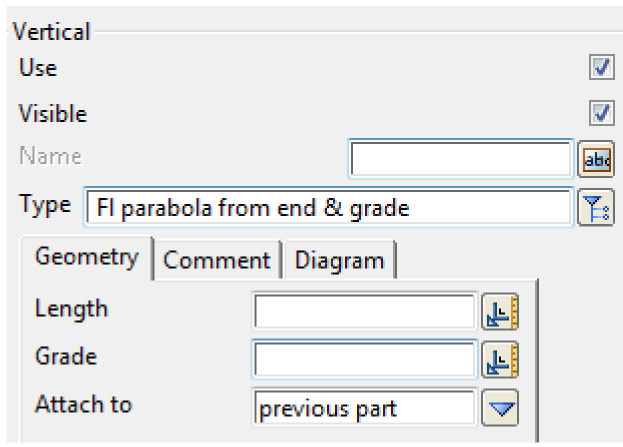
after clicking the **Same As** *button, an existing VIP can be selected from the screen and its information piped into the appropriate fields in the panel.*

Floating Parabola: Known Length and Grade at End

This *Floating Parabola* has a known length and a given grade at the opposite end of the parabola to the *Attached To* part. It is fully determined by being tangential to either the previous part or the following part.

It may not always be possible for the Floating Parabola to solve.





The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Length	measure box		available measures

the length of the Floating parabola. Must be non zero.

Grade	measure box		available measures

the parabola has a given length **Length** *and has the grade* **Grade** *at the opposite end from the* **Attach to** *part. The units for Grade is percent grade.*

Attach to	choice box		previous part, next part

the Part that the Floating Parabola is attached to, and made tangential to.

Comment tab

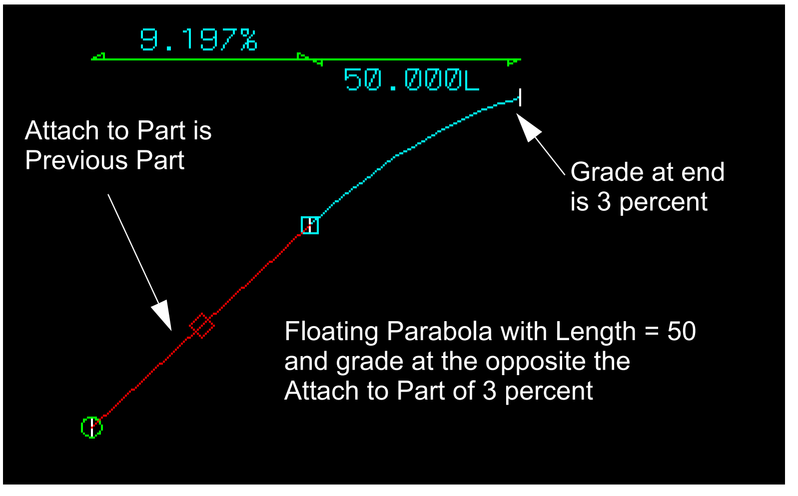
the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* **#** *is placed after the type of the part in the Vertical Parts list.*

Set button

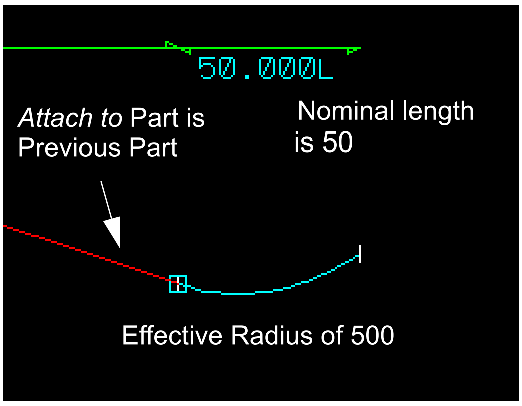
the **Set** *button must be clicked for the information for this Part to be used.*

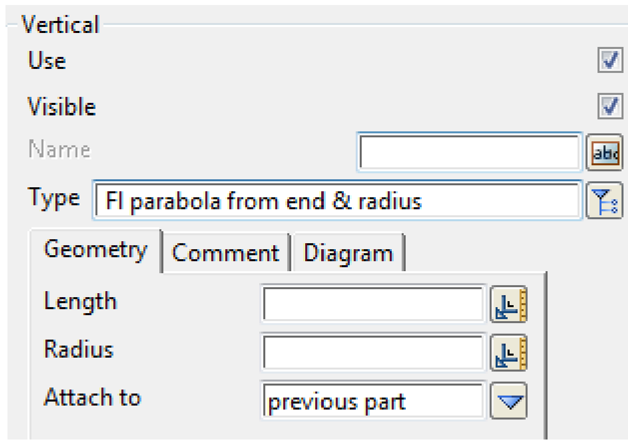
Same As button



Floating Parabola: Known Radius and Nominal Length
This *Floating Parabola* has a known effective radius (100 times K value). It is fully determined by being tangential to either the previous part or the following part.

It may not always be possible for the Floating Parabola to solve.





The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Length	measure box		available measures

the nominal length of the Floating Parabola. Must be non zero.

Radius	measure box		available measures

the effective radius of the parabola (this is 100 times the K value).

Attach to	choice box	previous part	previous part, next part

the Part that the Floating Parabola is attached to, and made tangential to

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Vertical Parts list.*

Set button

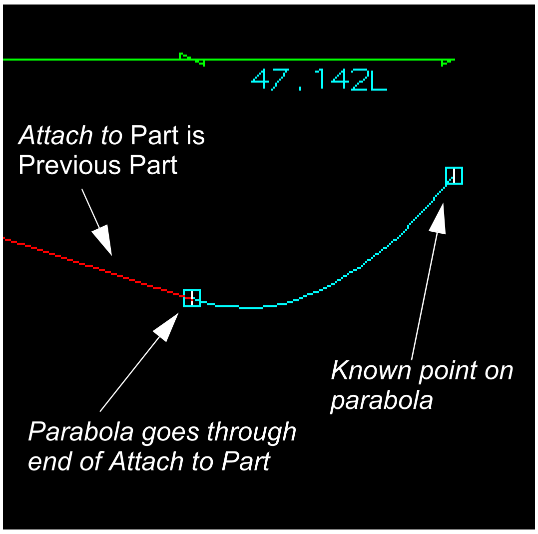
the **Set** *button must be clicked for the information for this Part to be used.*

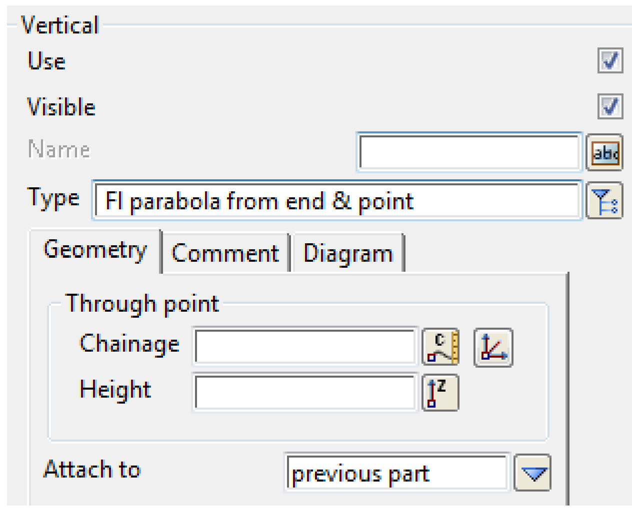
Same As button

Floating Parabola: known Point and Known End

This *Floating Parabola* goes through a known point and it attached to another known point from either the previous part or the following part. Being tangential to the Part fully determines the Floating Parabola.

It may not always be possible for the Floating Parabola to solve.





The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Field Description	Type	Defaults	Pop-Up
Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Field Description	Type	Defaults	Pop-Up
Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Field Description	Type	Defaults	Pop-Up
Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Through point Chainage, Height

the (chainage,height) coordinates for the point on the parabola can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons.*

Field Description	Type	Defaults	Pop-Up
Attach to	choice box	previous part	previous part, next part

the Part that the Floating Parabola is attached to, and made tangential to

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* **#** *is placed after the type of the part in the Vertical Parts list.*

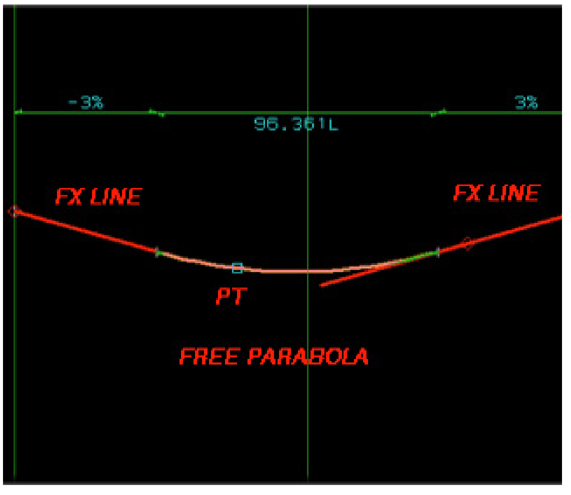
Set button

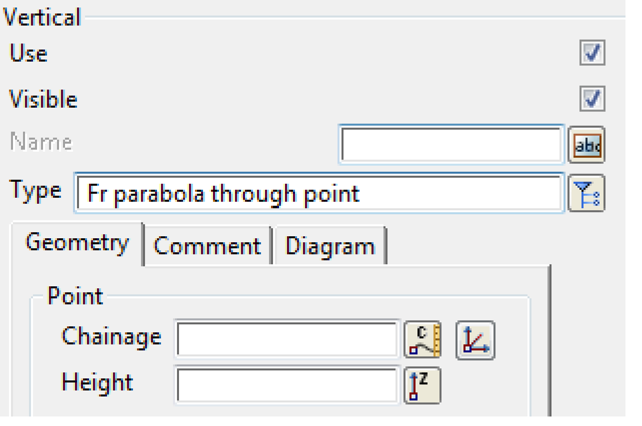
the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

Free Parabola: Known Point
This Free Parabola passes through a known point. Being tangential to the parts on either side fully determines the parabola.

It may not always be possible for the Free Parabola to solve.





The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Chainage, Height

the (chainage,height) coordinates for the point on the parabola can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons.*

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* **#** *is placed after the type of the part in the Vertical Parts list.*

Set button

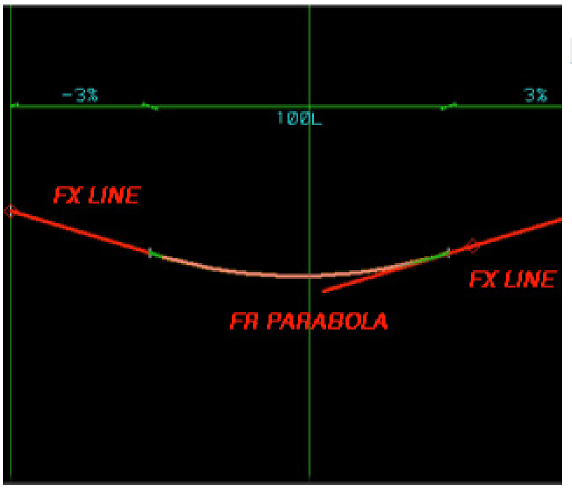
the **Set** *button must be clicked for the information for this Part to be used.*

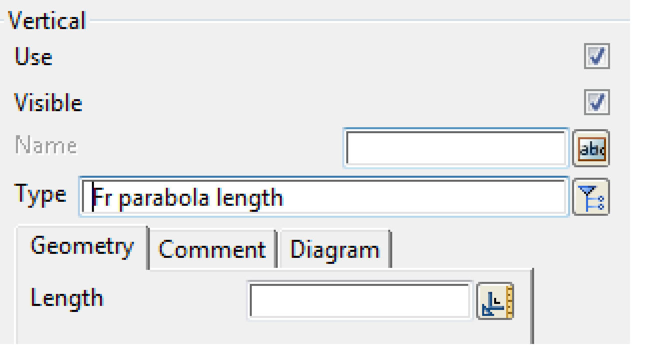
Same As button

Free Parabola: Known Length

This Free Parabola has a known length. Being tangential to the parts on either side fully determines the parabola.

It is not always possible for the Free Parabola to solve.





The fields and buttons used in the panel have the following functions.

Field Description   Type   Defaults   Pop-Up

Use   tick box   ticked

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible   tick box   ticked

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name   text box   blank

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type   choice box   types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Length   measure box   available measures

the length of the Free Parabola. Must be non zero.

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* **#** *is placed after the type of the part in the Vertical Parts list.*

Set   button

the **Set** *button must be clicked for the information for this Part to be used.*

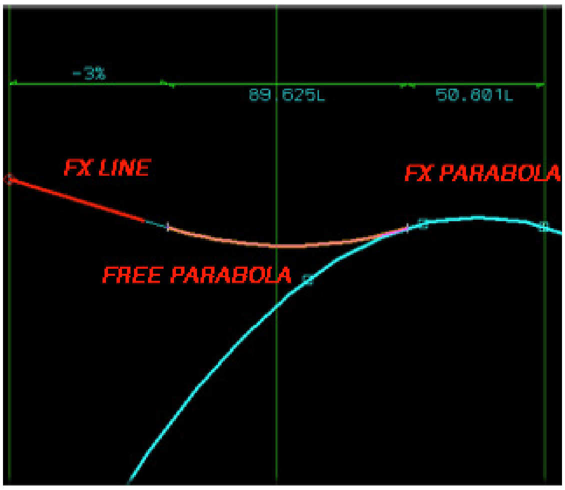
Same As button

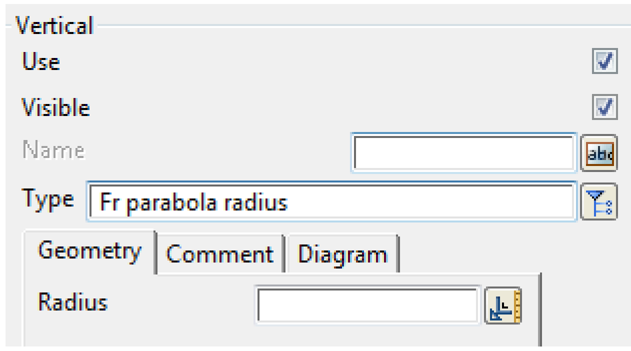
21.9.5.4 Vertical Parabolas

Free Parabola: Know Effective Radius

This Free Parabola has a known effective radius. Being tangential to the parts on either side fully determines the parabola.

It is not always possible for the Free Parabola to solve.





The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Radius	measure box		available measures

the effective radius of the parabola (this is 100 times the K value).

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Vertical Parts list.*

Set button

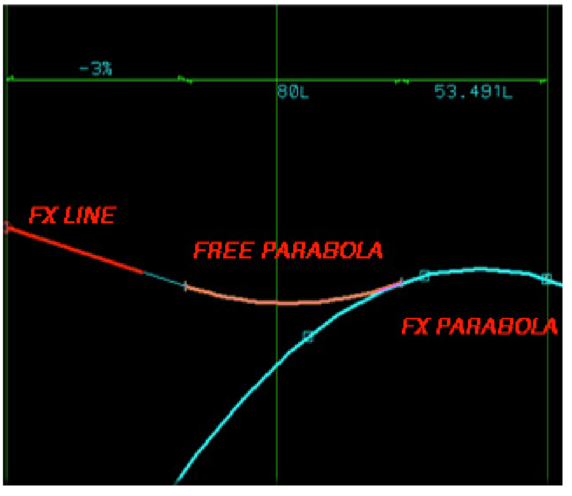
the **Set** *button must be clicked for the information for this Part to be used.*

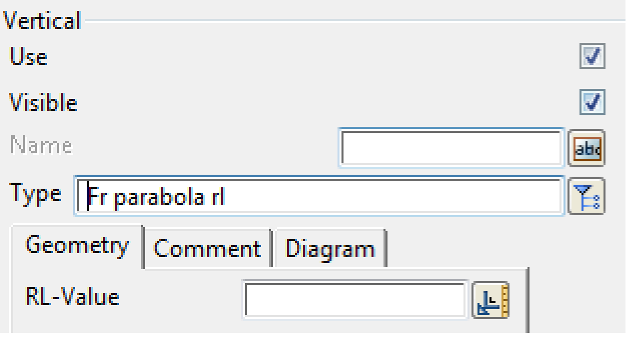
Same As button

Free Parabola: Known Parabolic Length/100

This Free Parabola has a known parabolic length divided by 100. Being tangential to the parts on either side fully determines the parabola.

It is not always possible for the Free Parabola to solve.





The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Length/100 Value	measure box		available measures

the parabolic length divided by 100

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* **#** *is placed after the type of the part in the Vertical Parts list.*

Set	button		

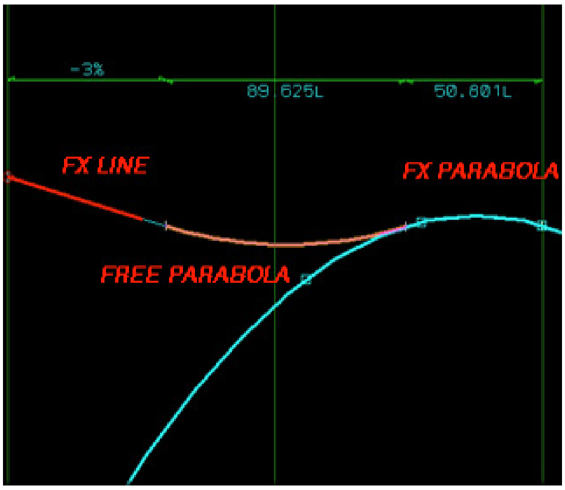
the **Set** *button must be clicked for the information for this Part to be used.*

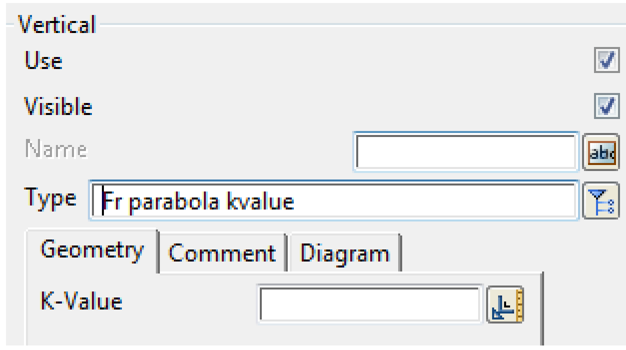
Same As button

Free Parabola: Known K Value

This Free Parabola has a known K value. Being tangential to the parts on either side fully determines the parabola.

It is not always possible for the Free Parabola to solve.





The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

K-Value	measure box		available measures

the K value of the parabola

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Vertical Parts list.*

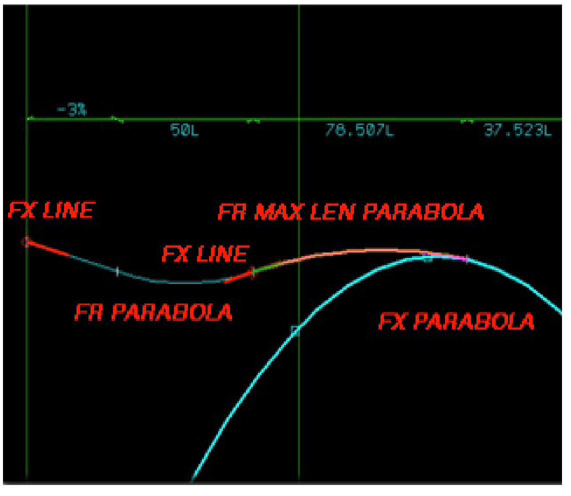
Set	button		

the **Set** *button must be clicked for the information for this Part to be used.*

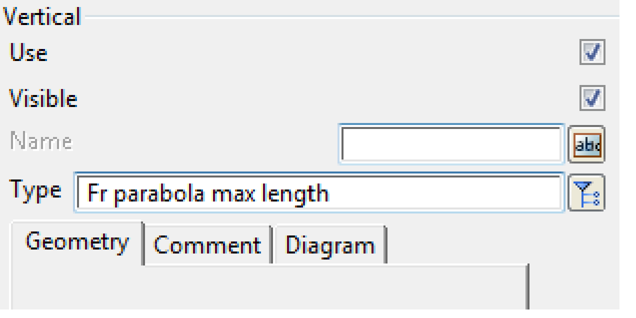
Same As button

Free Parabola: Fit with Maximum Length
This Free Parabola has a maximum parabolic length which will be determined by the parts on either side. Also being tangential to the parts on either side fully determines the parabola.

It is not always possible for the Free Parabola to solve.



Example:
The Free Transition is between the line of the left and the parabola on the right



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Field Description	Type	Defaults	Pop-Up
Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Field Description	Type	Defaults	Pop-Up
Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Field Description	Type	Defaults	Pop-Up
Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Vertical Parts list.*

Set button

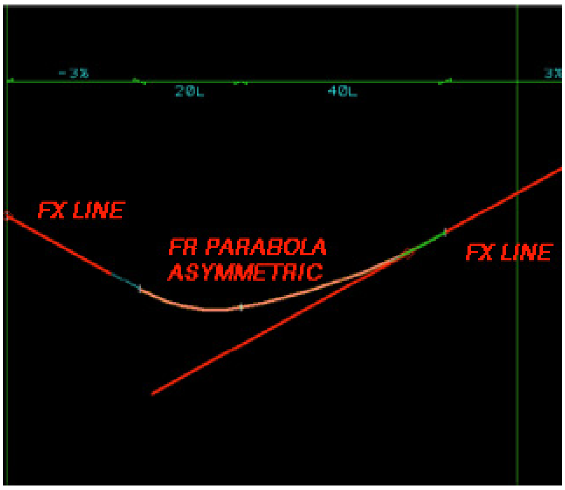
the **Set** *button must be clicked for the information for this Part to be used.*

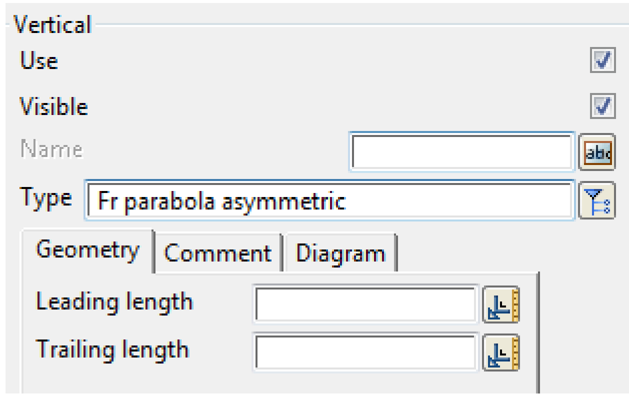
Same As button

Free Asymmetric Parabola: Two Known Lengths

This Free Parabola has asymmetric parabolic with **known leading** and **trailing lengths**. Being tangential to the parts on either side fully determines the parabola.

It is not always possible for the Free Parabola to solve.





The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Leading length	measure box		available measures

the length of the leading parabola of the asymmetric parabola

Trailing length	measure box		available measures

the length of the trailing parabola of the asymmetric parabola

Comment tab

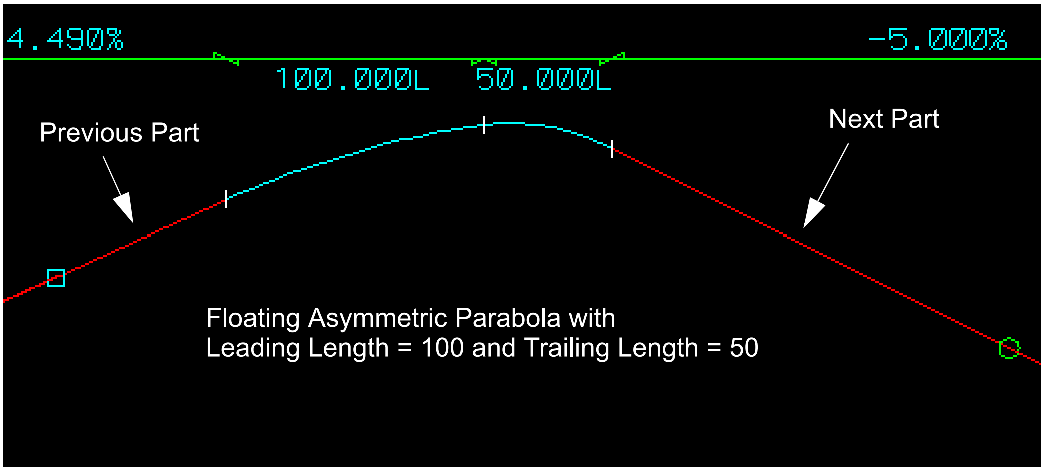
the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Vertical Parts list.*

Set button

the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

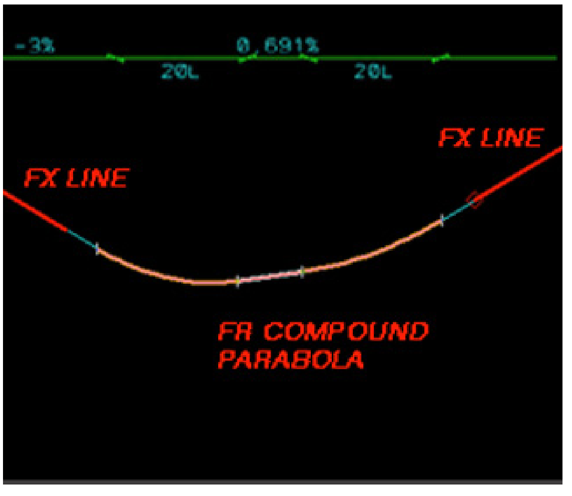


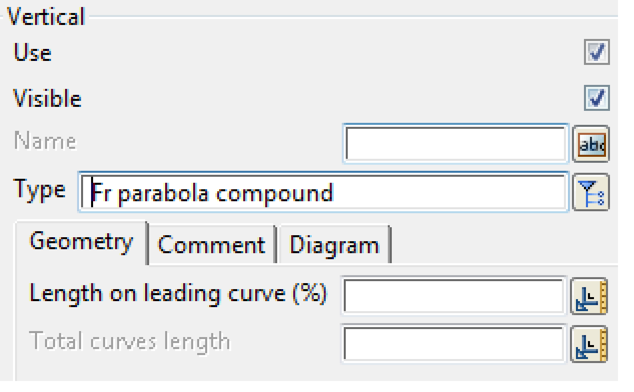
Free Compound Parabola: Two Parabolas with Optional Total Parabolic Length
This Free Parabola has two parabolas with a given total length and a ratio of the division between the two parabolas. There is an option total length of the two parabolas.

With a given Total Length, a straight between the two parabolas may need to be automatically inserted to get a solution.

Being tangential to the parts on either side fully determines the two parabolas and any necessary straight. The grades on the previous and next parts also determine if the parabolas are crest or sag parabolas.

It is not always possible for the Free Parabola to solve.





The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Length on leading curve (%)	measure box		available measures

the percentage of the total parabolic length going to the first parabola. Must be non zero.

Total curves length measure box available measures

if blank, the two parabolic lengths are calculated in the given ratio to fit.

If no zero, this value and the **Length on leading curve (%)** *fully determines the lengths of the two parabolic curves but a straight between the two parabolas may need to be automatically inserted to get a solution.*

Comment tab

the text typed into the text box is stored as a comment for the Part.

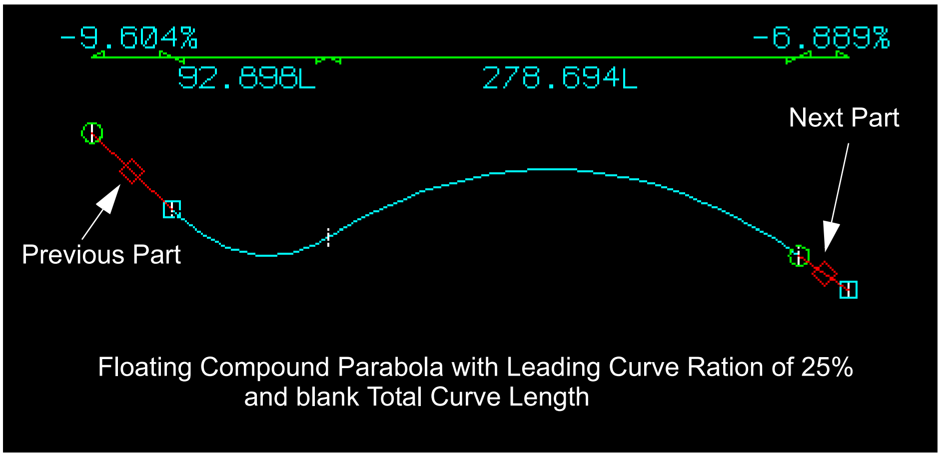
Note: *If a comment exists, a* # *is placed after the type of the part in the Vertical Parts list.*

Set button

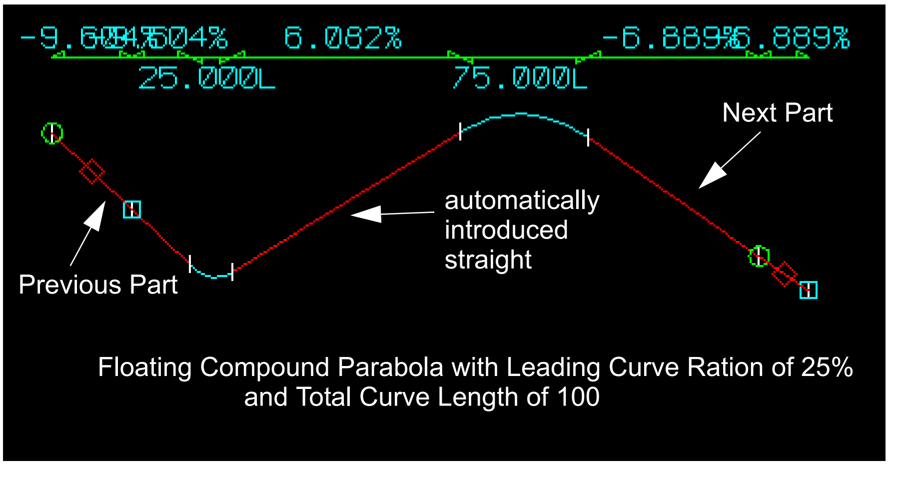
the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

after clicking the **Same As** *button, an existing VIP can be selected from the screen and its information piped into the appropriate fields in the panel.*



Floating Compound Parabola with Leading Curve Ration of 25%
and blank Total Curve Length

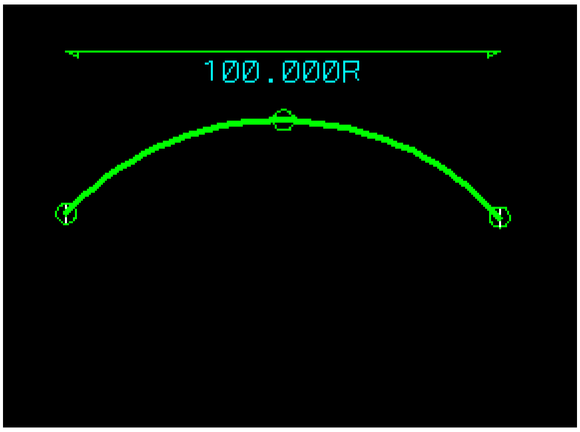


Floating Compound Parabola with Leading Curve Ration of 25%
and Total Curve Length of 100

21.9.5.5 Vertical Arcs

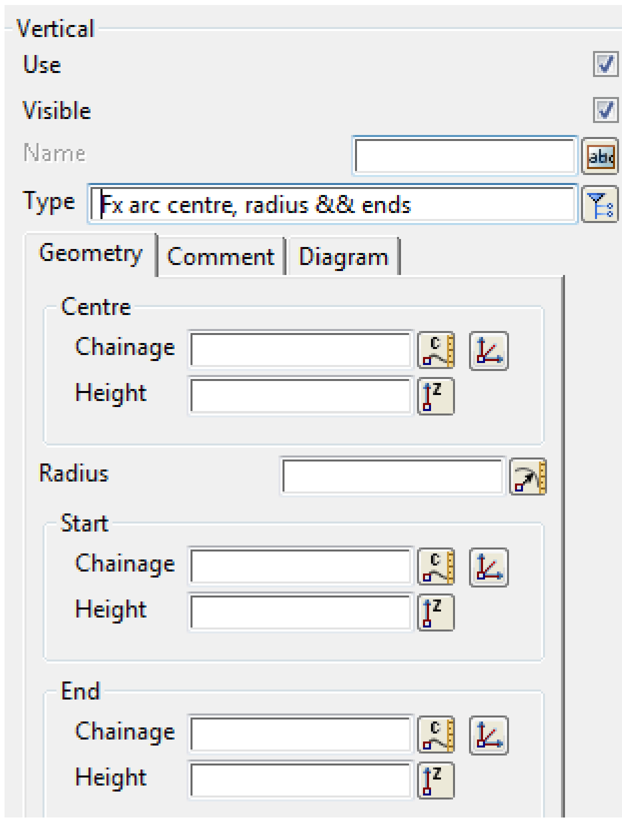
See Fixed Arc- Known Centre, Start and End Points
See Floating Arc: Known Point and Radius
See Floating Arc: Known Point and Grade at Point
See Floating Arc: Through Two Known Points
See Free Arc: Known Radius
See Free Arc: Known Point on the Arc
See Free Arc: Known Length of Arc

Fixed Arc- Known Centre, Start and End Points
The *Fixed Arc* is fully defined by a **known arc centre**, a **known radius** and **known start** and **end** points.



Example
Arc with radius of 100 and given centre, start and end points.

Vertical Exaggeration = 1



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.
If not ticked, the Part is not used in the vertical geometry.

Field Description	Type	Defaults	Pop-Up
Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Field Description	Type	Defaults	Pop-Up
Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

***Note:** If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type choice box types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Centre Chainage, Height

the (chainage,height) coordinates of the centre of the arc can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons*

Radius radius box Measure Radius

radius of the arc. Must be non zero.

Start Chainage, Height

the (chainage,height) coordinates of the start of the arc can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons*

End Chainage, Height

the (chainage,height) coordinates of the end of the arc can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons*

Comment tab

the text typed into the text box is stored as a comment for the Part.

***Note:** If a comment exists, a* **#** *is placed after the type of the part in the Vertical Parts list.*

Set button

the **Set** *button must be clicked for the information for this Part to be used.*

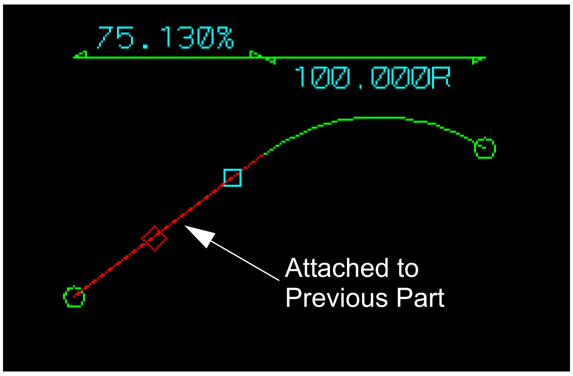
Same As button

Floating Arc: Known Point and Radius

The *Floating Arc* passes through a known point and has a known radius. Being tangential to a previous or next part is needed to fully determine the *Floating Arc*.

It is not always be possible for the *Floating Arc* to solve.

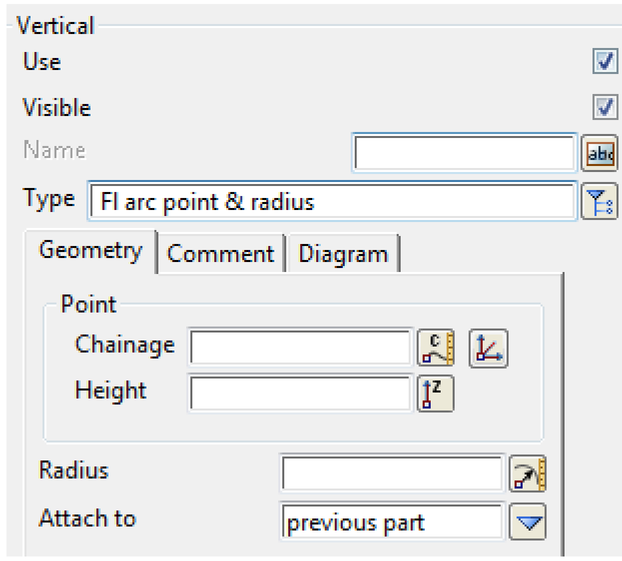
For example in the case where it follows a fixed line, if the perpendicular distance between the known point and the line is greater than the given radius, then there is no solution.



Example

Floating Arc going through a known point and with radius of 100.

Vertical Exaggeration = 1



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Chainage, Height

the (chainage,height) coordinates of the point the arc goes through can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons*

Radius	radius box		Measure Radius

radius of the arc. Must be non zero.

Attach to choice box previous part previous part, next part

the Part that the Floating Arc is attached to, and made tangential to.

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Vertical Parts list.*

Set button

the **Set** *button must be clicked for the information for this Part to be used.*

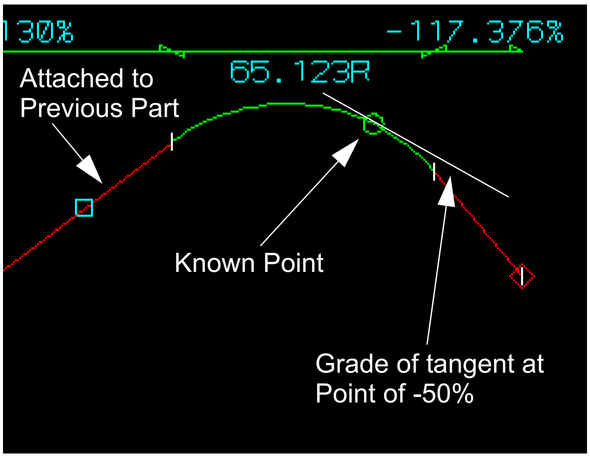
Same As button

Floating Arc: Known Point and Grade at Point

The *Floating Arc* passes through a known point and has a known grade for the tangent to the arc at the known point.

It may not always be possible for the Floating Arc to actually solve.

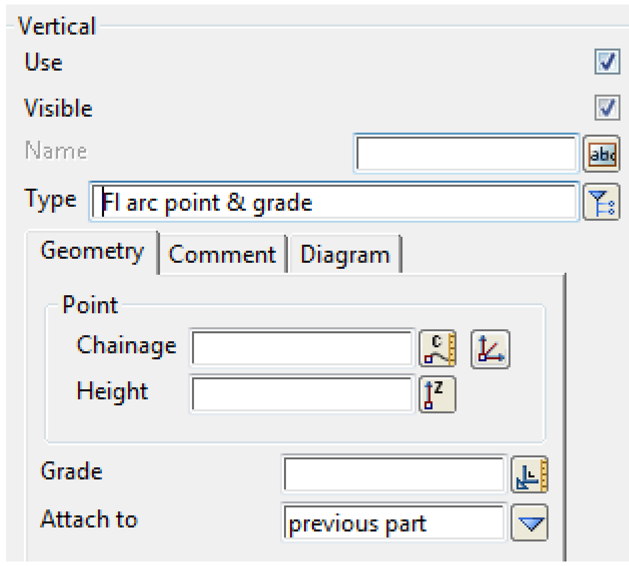
For example in the case where it follows a fixed line, if the perpendicular distance between the known point and the line is greater than the given radius, then there is no solution.



Example

Floating Arc going through a known point with a grade at that point of -50 per cent.

Vertical Exaggeration = 1



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.
If not ticked, the Part is not used in the vertical geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Point Chainage, Height

the (chainage,height) coordinates of the point the arc goes through can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons*

Grade	measure box		available measures

the grade (in percent grade) of the tangent at the given point

Attach to	choice box	previous part	previous part, next part

the Part that the Floating Arc is attached to, and made tangential to.

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* # *is placed after the type of the part in the Vertical Parts list.*

Set button

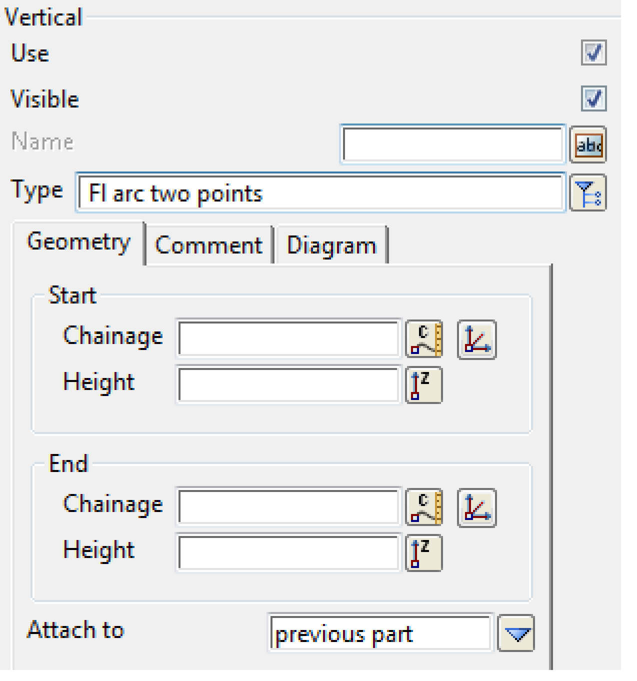
the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

Floating Arc: Through Two Known Points

The *Floating Arc* passes through two known points.

It is not always possible for the Floating Arc to solve.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Start Chainage, Height

the (chainage,height) coordinates for the first point the parabola goes through can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons.*

End Chainage, Height

the (chainage,height) coordinates for the second point the parabola goes through can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons.*

Attach to choice box previous part previous part, next part

the Part that the Floating Arc is attached to, and made tangential to.

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* **#** *is placed after the type of the part in the Vertical Parts list.*

Set button

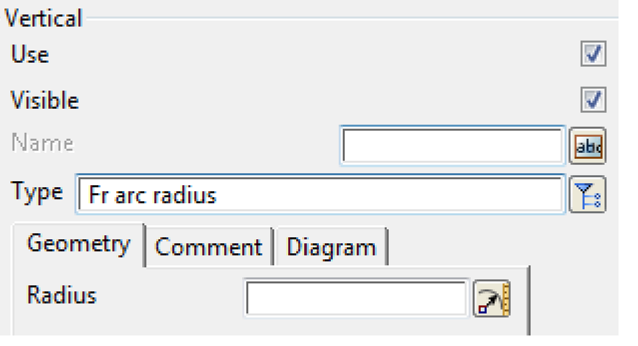
the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

Free Arc: Known Radius

The *Free Arc* has a **known radius**. The Free arc is fully determined by being tangential to the parts on either side.

It is not always possible for the *Free Arc* to solve.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Radius	radius box		Point, String from point

the radius of the arc

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* **#** *is placed after the type of the part in the Vertical Parts list.*

Set	button		

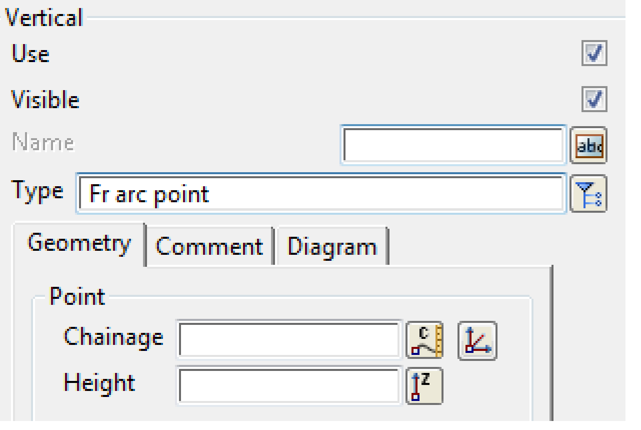
the **Set** *button must be clicked for the information for this Part to be used.*

Same As	button		

Free Arc: Known Point on the Arc

The *Free Arc* passes through a **known point**. The Free arc is fully determined by being tangential to the parts on either side.

It is not always possible for the *Free Arc* to solve.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Point Chainage, Height

the (chainage,height) coordinates of the point that the arc goes through can be typed in, or selected using the **C**, **Z** *or* **CZ** *icons.*

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* **#** *is placed after the type of the part in the Vertical Parts list.*

Set	button		

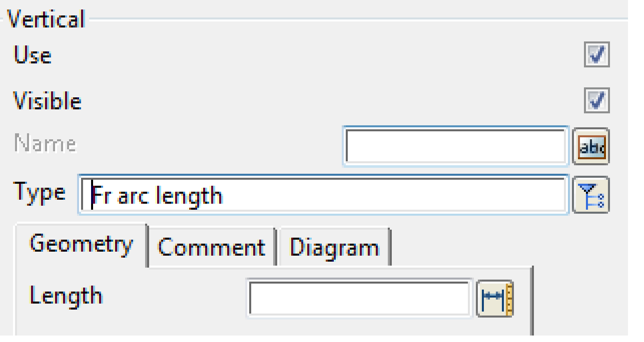
the **Set** *button must be clicked for the information for this Part to be used.*

Same As button

Free Arc: Known Length of Arc

The *Free Arc* has a **known length**. The Free arc is fully determined by being tangential to the parts on either side.

It is not always possible for the *Free Arc* to solve.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Use	tick box	ticked	

if ticked, then the Part is used in the vertical geometry.

If not ticked, the Part is not used in the vertical geometry.

Visible	tick box	ticked	

if ticked, then the Part is drawn in the vertical geometry.

If not ticked, then the Part is not drawn whenever all the Parts before it, or all the Parts after it, also have **Visible** *not ticked. The effect may not be apparent until leaving the Editor and all the construction work is removed.*

Name	text box	blank	

if not blank, then the Part is given this name.

If blank then the Part has no name.

Note: *If the part has a name, then a* **!** *is placed after the type of the part in the Vertical Parts list.*

Type	choice box		types of vertical parts

the type of this part. To change the Part type, choose another type from the pop-up list.

Geometry tab

the geometrical information defining the Part

Length	length box		Point to point, String to point

the length of the Free arc. Must be non zero.

Comment tab

the text typed into the text box is stored as a comment for the Part.

Note: *If a comment exists, a* **#** *is placed after the type of the part in the Vertical Parts list.*

Set	button		

the **Set** *button must be clicked for the information for this Part to be used.*

Same As	button		

21.9.5.6 Vertical Computators

Not yet documented.

21.10 Text Format of Files

Go to 21.10.1 Text Format of the MTF File

21.10.2 Text Format of the Boxing File

21.10.1 Text Format of the MTF File

In the text version of the mtf file, most of (a) to (h) begin with a key word, following by an equals sign (=) and then special commands enclosed in { }.

```
key_word   =  {
                 commands
              }
```

The key words for each case above are

(a) left_side, right_side

(b) specials

(c) hinge_modifier

(d) left_side_modifier, right_side_modifier

(e) stripping

(f) boxing_file, left_boxing, right_boxing, left_boxing_2, right_boxing_2 ... left_boxing_8, right_boxing_8

(g) auto_super_tables, left_auto_super, left_auto_width, right_auto_super, right_auto_width

(h) loops

(i) section_width

(j) string_modifiers

If any of the key words exist, then they must be in the following order in the **mtf** file:

left_side	=	{ ... }
right_side	=	{ ... }
specials	=	{ ... }
hinge_modifier	=	{ ... }
left_side_modifier	=	{ ... }
right_side_modifier	=	{ ... }
stripping	=	{ ... }
boxing_file	=	“something.bf”
left_boxing	=	{ ... }
right_boxing	=	{ ... }
section_width	=	*value*
string_modifiers	=	{ ... }
auto_super_tables	=	1 or 0
left_auto_super	=	{ ... }
left_auto_width	=	{ ... }
right_auto_super	=	{ ... }
right_auto_width	=	{ ... }
loop_removals	=	{ ... }

The format for *boxing_file* is simply

boxing_file = *"something.bf"*

where *something.bf* is the name of a file containing boxing definitions to use for the mtf.

The format for (g) is simply

section_width = *value* // default is 10000 where *value* is the distance to search from the hinge string for strings required in some options. If section_width is missing, then it takes the default value of 10000.

The key words, left_side and right_side which are for initially applying templates (for part (a)), have already been described under the **Design=>Apply=>Apply MTF** option but will be summarised in the next section 21.2.1 MTF Hinge Modifiers.

The documentation for the **text file syntax** of the MTF file now follows.

See 21.10.1.1 MTF Templates File Format

21.10.1.2 MTF Specials File Format

21.10.1.3 MTF Hinge Modifier File Format

21.10.1.4 MTF MTF Modifiers File Format

21.10.1.12 MTF Stripping File Format

21.10.1.13 MTF Boxing File Format

21.10.1.14 MTF Width File Format

21.10.1.15 MTF String Modifiers File Format

21.10.1.16 Substitutions in the Modifiers and Templates File

21.10.1.1 MTF Templates File Format

The templates on the left and right hand sides of the centre line (or Hinge string if one is selected) are specified separately in the *Modifiers and Templates File*. Apart from a key word denoting whether the following part of the definition is for the left side or the right side, the set out for the left-side is identical to the right-side. Hence, only the left-side will be described in detail.

The left-side definition begins with the key words

left_side =

A list of chainages (in ascending order, one per line) with corresponding template names then follows. This list of chainages and templates is enclosed in curly braces **{ }**.

Note if the template name includes spaces, then the name must be enclosed in quotes **".** For example, "left 1".

The chainage-template lists are assembled as follows

(a) To represent a template starting at a given chainage, the chainage value followed by the template name is given. The chainage and name are separated by one or more spaces. For example, the template **std** starting at chainage 150 is represented by

150 std

The template is assumed to apply until the chainage given on the next line of the left-side definition.

If the template is to go to the end of the centre-line, add a line with a chainage greater than or equal to the end chainage. For example,

150 std

99999

(b) if no template exists from a chainage, simply include the chainage with no template name following it. For example, if there is no template from chainage 250, this is represented by

250

The non-existence of a template is assumed to apply until the chainage given on the next line of the left-side definition

(c) the case of a linear change from one template to another template over a specified chainage range is represented by giving the start chainage of the linear change, followed on the same line by the start template, a comma, and the end template. For example, if the template is to vary linearly between the template **std** and the template **left**, beginning at the chainage 350, then the line in the file would be

350 std, left

The linear change takes place over the interval beginning at the chainage given on the defining line and ending at the chainage given on the next line of the left-side. Distances and percent cross-falls are interpolated linearly and slopes are interpolated on the radian value of the slope angles.

Combining these rules, the following lines in a template file

```
left_side = {
          100       std
          200       std , "left 1"
          250       "left 1"
          300
          350       std
          99999
}
```

describes the situation

1. the left-hand side of the centre-line has no template from the beginning of the centre-line until chainage 100.
2. at chainage 100, the template **std** begins and continues until chainage 200.

3. there is a linear change from the template **std** to the template **left 1** between chainage 200 and chainage 250.
4. the template **left 1** goes from chainage 250 to chainage 300.
5. there is a gap between chainage 300 and chainage 350.
6. the template **std** goes from chainage 350 to chainage 99999, or if the end chainage is smaller than 99999, to the end of the centre-line.

The right-side template definition begins with the key word right_side. The rest of the definition follows the same rules at the left hand side of the centre-line. For example,

```
right_side = {
         100       std
         200       std , right
         250       right
         300       right , std
         350       std
         400
}
```

The left and right sides can vary independently.

Restrictions on the Template File Definitions

1. If one template stops at the same chainage that another template begins, then the two templates must have the same number of fixed and variable links. The templates will be varied linearly from the stopping template to the starting template over one section separation distance.
2. If there is a linear variation between two templates, the two templates must have the same number of fixed and variable links.
3. If two templates do not have the same number of links, they must be separated by a gap, that is, by a region with no template.

Summarising:

The templates on the left and right hand sides of the centre line are specified separately in the Modifiers and Templates File.

The left-side (right-side) definition begins with the key words

left_side =   right_side =

with a list of chainages (in ascending order, one per line) with template names. This list of chainages and templates is enclosed in curly braces **{ }**.

The chainage-template lists are assembled as follows

(a) To represent a template starting at a given chainage, the chainage value followed by the template name is given. The chainage and name are separated by one or more spaces. For example, the template **std** starting at chainage 150 is represented by

150   std

The template is assumed to apply until the chainage given on the next line of the left-side definition.

If the template is to go to the end of the centre-line, add a line with a chainage greater than or equal to the end chainage. For example,

150   std

99999

(b) if no template exists from a chainage, simply include the chainage with no template name following it. For example, if there is no template from chainage 250, this is represented by

250

The non-existence of a template is assumed to apply until the chainage given on the next line of the left-side definition

(c) the case of a linear change from one template to another template over a specified chainage range is represented by giving the start chainage of the linear change, followed on the same line by the start template, a comma, and the end template. For example, if the template is to vary linearly between the template **std** and the template **left**, beginning at the chainage 350, then the line in the file would be

350 std, left

The linear change takes place over the interval beginning at the chainage given on the defining line and ending at the chainage given on the next line of the left-side. Distances and percent cross-falls are interpolated linearly and slopes are interpolated on the radian value of the slope angles.

A Left and Right Side File Example

```
left_side = {
         100      std
         200      std , "left 1"
         250      "left 1"
         300
         350      std
         99999
}
right_side = {
         100      std
         200      std , right
         250      right
         300      right , std
         350      std
         400
}
```

The left_side describes the situation:

1. the left-hand side of the centre-line has no template from the beginning of the centre-line until chainage 100.
2. at chainage 100, the template **std** begins and continues until chainage 200.
3. there is a linear change from the template **std** to the template **left 1** between chainage 200 and chainage 250.
4. the template **left 1** goes from chainage 250 to chainage 300.
5. there is a gap between chainage 300 and chainage 350.
6. the template **std** goes from chainage 350 to chainage 99999, or if the end chainage is smaller than 99999, to the end of the centre-line.

Go to the next section 21.10.1.2 MTF Specials File Format or back to 21.10.1 Text Format of the MTF File

21.10.1.2 MTF Specials File Format

In the Modifiers and Templates File, the **specials** definition begins with the key word

specials =

This is followed by a list of

(a) chainages, in any order, one per line

and/or

(b) the names of files (enclosed in quotes **“**) which include lists of chainages (The default ending for a special chainage file is *.spc)

The list of chainages and/or file names is enclosed in curly braces {}.

A Specials Example

```
specials = {
        125.3
        1925.4
        “fred”        // file of chainages to read in
        3007
        “joe”
}
```

Go to the next section 21.10.1.3 MTF Hinge Modifier File Format or back to 21.10.1 Text Format of the MTF File

21.10.1.3 MTF Hinge Modifier File Format

In the Modifiers and Templates File, the **hinge modifier** definition begins with the key word

hinge_modifier =

followed by one or more of the hinge modifier commands **offset**, **height**, **coord** and **nohinge** enclosing curly braces {}.

```
hinge_modifier = {
                      hinge modifier commands
                 }
```

The definition of the hinge modifier commands will now be given.

Offset

The **Offset** modifier will move the hinge point a given offset distance (perpendicular to the reference string) from its current plan position. A positive offset is to the right of the hinge string and a negative offset to the left.

The format of the modifier to vary the **offset** is

offset st_ch end_ch st_offset end_offset absrel type extra_start extra_end

where

st_ch	start chainage for the modifier
end_ch	end chainage for the modifier
st_offset	offset to be used at the start modifier chainage, st_ch
end_offset	offset to be used at the end modifier chainage, end_ch.
absrel	relative (default) or absolute
type	linear (default) or cubic
extra_start	optional - adds in an extra x-section 0.1mm before start chainage
extra_end	optional - adds in an extra x-section 0.1mm before end chainage

If **absrel** is **relative** (the default) then

the st_offset is added to the current position of the hinge point at chainage st_ch.
the end_offset is added to the current position of the hinge point at chainage end_ch.

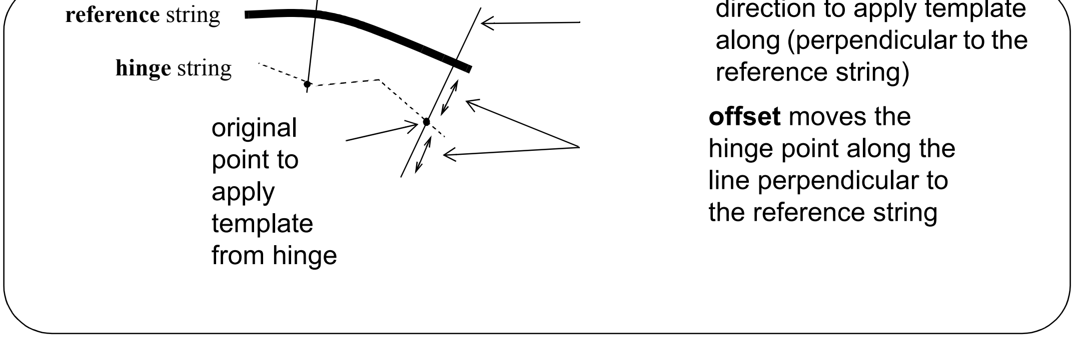
If **absrel** is **absolute**, then

the offset of the hinge point is taken with respect to the original position of the hinge string at chainage st_ch.
the offset of the hinge point is taken with respect to the original position of the hinge string at chainage end_ch.

If **type** is **linear** (the default), then the offset is varied linearly (with respect to the reference chainage) between the offsets at chainage st_ch and chainage end_ch.

If **type** is **cubic**, then the offset is varied as a reverse cubic (with respect to the reference chainage) between the offsets at chainage st_ch and chainage end_ch.

.



Offset to String

The offset for the hinge point can also be varied by **going out to another *12d Model* string**.

offset st_ch end_ch full_string_name extra_start extra_end

where

st_ch start chainage for the modifier
end_ch end chainage for the modifier

full_string_name name of a ***12d Model*** string to take the hinge point out to. The format of the string name is "model_name->string_name".

extra_start optional - adds in an extra x-section 0.1mm before start chainage
extra_end optional - adds in an extra x-section 0.1mm before end chainage

Height

The **height** modifier varies the height of the hinge point between the given chainages.

height st_ch end_ch st_height end_height absrel type extra_start extra_end

where

st_ch start chainage for the modifier
end_ch end chainage for the modifier

st_height height to be used at the start modifier chainage, st_ch
end_height height to be used at the end modifier chainage, end_ch.

absrel relative (default) or absolute
type linear (default) or cubic

extra_start optional - adds in an extra x-section 0.1mm before start chainage
extra_end optional - adds in an extra x-section 0.1mm before end chainage

If **absrel** is **relative** (the default) then

the st_height is added to the current height of the hinge point at chainage st_ch.
the end_height is added to the current height of the hinge point at chainage end_ch.

If **absrel** is **absolute**, then

the height of the hinge point is set to st_height above the original hinge string at chainage st_ch.
the height of the hinge point is set to end_height above the original hinge string at chainage end_ch.

If **type** is **linear** (the default), then the height is varied linearly (with respect to the reference chainage) between the height at chainage st_ch and chainage end_ch.

If **type** is **cubic**, then the height is varied as a reverse cubic (with respect to the reference chainage) between the height at chainage st_ch and chainage end_ch.

Height to String

The height of the hinge point can also be specified by **taking the height from another** 12d Model **string**.

height st_ch end_ch full_string_name extra_start extra_end

where

st_ch start chainage for the modifier
end_ch end chainage for the modifier

full_string_name name of a ***12d Model*** string to take the hinge point height from. The format of the string name is "model_name->string_name".

extra_start optional - adds in an extra x-section 0.1mm before start chainage
extra_end optional - adds in an extra x-section 0.1mm before end chainage

Coord to String

The **coord** modifier is used to replace the x, y and z position of the hinge point by the x, y and z position of **another *12d Model* string** between given chainages. Hence **coord** replaces the hinge string by another string between the given chainages.

coord st_ch end_ch full_string_name extra_start extra_end

where

st_ch	start chainage for the modifier
end_ch	end chainage for the modifier
full_string_name	name of the ***12d Model*** string used to replace the hinge string between the chainages st_ch and end_ch. The format of the string name is “model_name->string_name”.
extra_start	optional - adds in an extra x-section 0.1mm before start chainage
extra_end	optional - adds in an extra x-section 0.1mm before end chainage

Nohinge

The **nohinge** modifier is used to **stop** the hinge string (and hence the apply) between given chainages. This will leave a gap in the strings created by the apply between the given chainages.

nohinge st_ch end_ch extra_start extra_end

where

st_ch	start chainage to stop the apply
end_ch	end chainage for stopping the apply
extra_start	optional - adds in an extra x-section 0.1mm before start chainage
extra_end	optional - adds in an extra x-section 0.1mm before end chainage

Please continue to the next section A Hinge Modifier Example.

A Hinge Modifier Example

```
hinge_modifier  =  {
        offset    0   250   0    3             // linearly offset the hinge by 0 to 3
                                               // over the chainage range 0 to 250.
        height    125   300   2    2           // add 2 to the hinge height
                                               // over the chainage range 125 to 300.
        coord     300   400   “mod->new_string”   // use the position of the string
                                                  // new_string over the chainage
                                                  // range 300 to 400.
        nohinge    400   500   2    2          // stop the apply between the chainages
                                               // over the chainage range 400 to 500.
        }
```

Go to the next section 21.10.1.4 MTF MTF Modifiers File Format or back to 21.10.1 Text Format of the MTF File

21.10.1.4 MTF MTF Modifiers File Format

In the Modifiers and Templates File, the **template modifier** definition begins with the key word

left_side_modifier =

and/or

right_side_modifier =

followed by one or more of the template modifier commands enclosing curly braces **{ })**

(a) fixed link modifiers - commands working on fixed links

insert

remove

width, **height**, **xfall, xfall_crc, copy_width, copy_height, copy_xfall, tin_height, tin_xfall**

(b) stop and start decisions commands

decision

(c) variable cut and fill link modifiers - commands working on cut/fill links

insert_cut, **insert_fill**

remove_cut, **remove_fill**

cut_width, **cut_height**, **cut_slope, copy_cut_width, copy_cut_height, copy_cut_slope, tin_cut_height, tin_cut_slope**

fill_width, **fill_height**, **fill_slope, copy_fill_width, copy_fill_height, copy_fill_slope, tin_fill_height, tin_fill_slope**

(d) final cut/fill link modifiers

final_width, **final_cut_slope**, **final_no_cut_slope**

final_fill_slope, **final_no_fill_slope**

(e) snippets - inserting groups of modifier commands

snippet

That is,

```
left_side_modifier  =  {
                          template modifier commands
                       }
```

and/or

```
right_side_modifier  =  {
                          template modifier commands
                        }
```

The definition of the template modifier commands and their file format will now be given.

For **Start mode and End mode**, go to the section 21.10.1.5 Start and End Chainages in the Modifier File Format.

For **Fixed Link** modifiers, go to the section 21.10.1.6 MTF Fixed Link Modifiers File Format.

For **Decision Link** modifiers, go to the section 21.10.1.7 Decision File Format.

For **Cut and Fill Link** modifiers, go to the section 21.10.1.8 Cut and Fill Variable Link Modifiers File Format.

For **Final Link** modifiers, go to the section 21.10.1.9 Final Link Modifiers File Format.

For **Snippets**, go to the section 21.10.1.10 Snippets File Format.

Please continue to the next section 21.10.1.5 Start and End Chainages in the Modifier File Format.

21.10.1.5 Start and End Chainages in the Modifier File Format

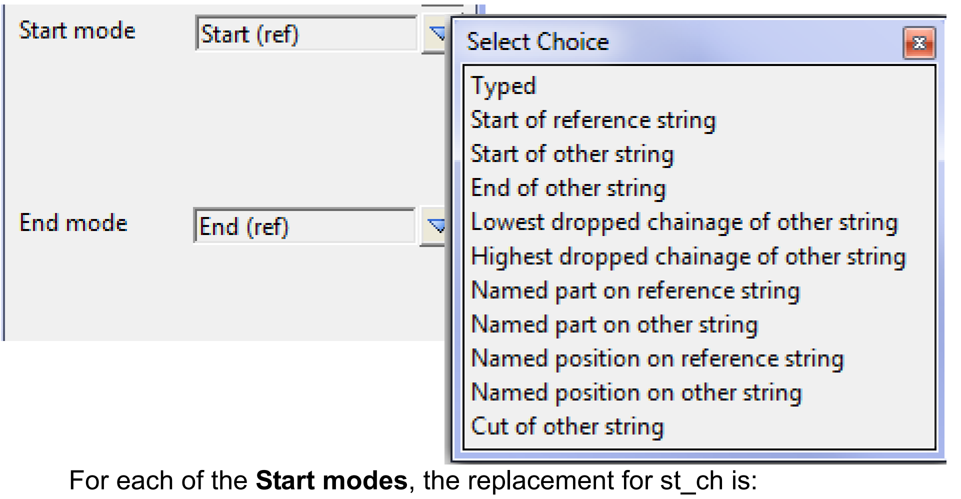
Modifiers work between two chainages and in each modifier command, the position of the start and end chainages for the commands are denoted by:

st_ch start chainage for the modifier

end_ch end chainage for the modifier

However there are a number of ways for defining the start and end chainages in the modifier command (called Start mode and End mode on the modifier panels.

Start Mode - st_ch



For each of the **Start modes**, the replacement for st_ch is:

(a) Typed

the actual typed in chainage value

(b) Start of reference string

$null

(c) Start of other string

chainage_start "model_of_string->string_name"

where model_of_string and string_name are the model and name of the string to take the start chainage from. The double quotes (") are needed.

(d) End of other string

chainage_final "model_of_string->string_name"

where model_of_string and string_name are the model and name of the string to take the end chainage from. The double quotes (") are needed.

(e) Lowest dropped chainage of other string

chainage_low "model_of_string->string_name"

where model_of_string and string_name are the model and name of the string to take the lowest dropped chainage from. The double quotes (") are needed.

(f) Highest dropped chainage of other string

chainage_high "model_of_string->string_name"

where model_of_string and string_name are the model and name of the string to take the highest dropped chainage from. The double quotes (") are needed.

(g) Named part of reference other string

named_part "part_specifier" start_extension_value

where part_specifier is the full specification of the part of the reference string to use. The start_extension_value is added to the chainage. The double quotes (") are needed.

(h) Named part of other string

named_part "model_of_string->string_name" "part_specifier" start_extension_value

where model_of_string and string_name are the model and name of the string to take the name part from, and part_specifier is the full specification of the part from that string. The start_extension_value is added to the chainage. The double quotes (") are needed.

(i) Named position of reference

named_position "pos_name" start_extension_value

where pos_name is the name of the named position of the reference string to use. The start_extension_value is added to the chainage. The double quotes (") are needed.

(j) Named position on other string

named_position "model_of_string->string_name" "pos_name" start_extension_value

where model_of_string and string_name are the model and name of the string to take the name position from, and pos_name is the name of the named position from that string. The start_extension_value is added to the chainage. The double quotes (") are needed.

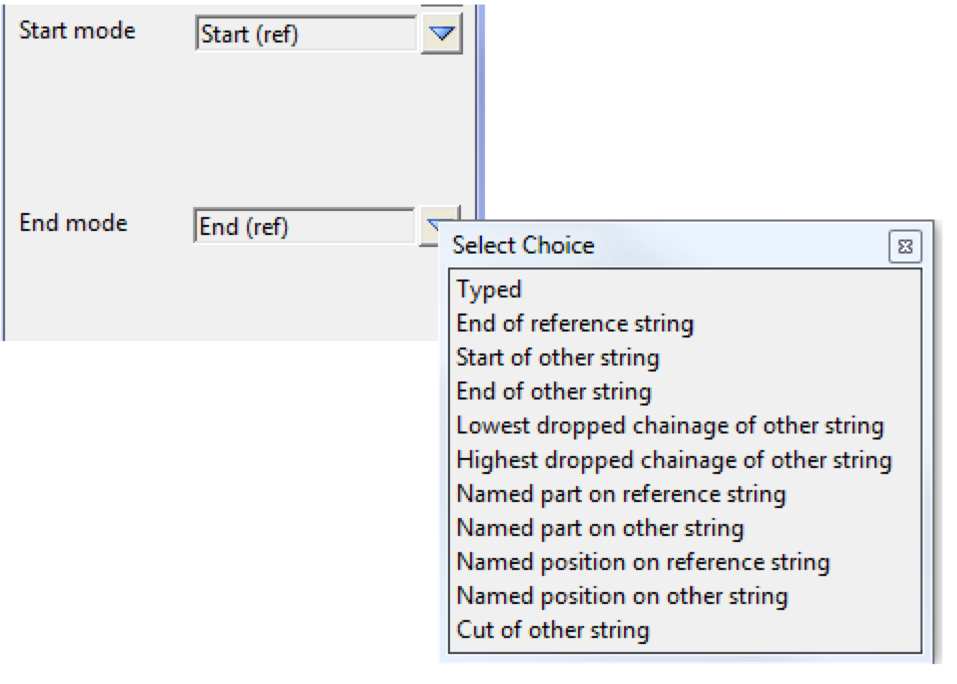
(k) Cut of other string

cut_other "model_of_string->string_name" "name" start_extension_value

where model_of_string and string_name are the model and name of the selected string. A chainage is calculated from the cut of the string with the reference string. The start_extension_value is added to the chainage. The double quotes (") are needed.

Note: If there are no cuts, or if there is more than one cut, then the command is not processed.

End Mode - end_ch



For each of the **End modes**, the replacement for end_ch is:

(a) Typed

the actual typed in chainage value

(b) End of reference string

$null

(c) Start of other string

chainage_start "model_of_string->string_name"

where model_of_string and string_name are the model and name of the string to take the start chainage from. The double quotes (") are needed.

(d) End of other string

chainage_final "model_of_string->string_name"

where model_of_string and string_name are the model and name of the string to take the end chainage from. The double quotes (") are needed.

(e) Lowest dropped chainage of other string

chainage_low "model_of_string->string_name"

where model_of_string and string_name are the model and name of the string to take the lowest dropped chainage from. The double quotes (") are needed.

(f) Highest dropped chainage of other string

chainage_high "model_of_string->string_name"

where model_of_string and string_name are the model and name of the string to take the highest dropped chainage from. The double quotes (") are needed.

(g) Named part of reference other string

named_part "part_specifier" start_extension_value

where part_specifier is the full specification of the part of the reference string to use. The start_extension_value is added to the chainage. The double quotes (") are needed.

(h) Named part of other string

named_part "model_of_string->string_name" "part_specifier" end_extension_value

where model_of_string and string_name are the model and name of the string to take the name part from, and part_specifier is the full specification of the part from that string. The end_extension_value is added to the chainage. The double quotes (") are needed.

(i) Named position of reference

named_position "pos_name" end_extension_value

where pos_name is the name of the named position of the reference string to use. The end_extension_value is added to the chainage. The double quotes (") are needed.

(j) Named position on other string

named_position "model_of_string->string_name" "pos_name" end_extension_value

where model_of_string and string_name are the model and name of the string to take the name position from, and pos_name is the name of the named position from that string. The end_extension_value is added to the chainage. The double quotes (") are needed.

(k) Cut of other string

cut_other "model_of_string->string_name" "name" start_extension_value

where model_of_string and string_name are the model and name of the selected string. A chainage is calculated from the cut of the string with the reference string. The end_extension_value is added to the chainage. The double quotes (") are needed.

Note: is there is no cuts or more than one cut then the command is not processed.

Please continue to the next section 21.10.1.6 MTF Fixed Link Modifiers File Format.

21.10.1.6 MTF Fixed Link Modifiers File Format

Fixed - Link Insert

Fixed links can be created by one of three **insert** commands by specifying either width and height, width and xfall, or height and xfall.

insert link_name colour width height **unknown** st_ch end_ch *optional // use width, height*

insert link_name colour width **unknown** xfall st_ch end_ch *optional // use width, xfall*

insert link_name colour **unknown** height xfall st_ch end_ch *optional // use height, xfall*

where **unknown** takes the place of the one of width, height or xfall not being used, and

link_name	name of the link being created
colour	colour of the link being created
width, height, xfall	width, height or xfall of the created link
st_ch	start chainage for creating the link
end_ch	end chainage for creating the link

and *optional* can be none, one or more of the following

name	if non-blank, insert before the link name in the template if blank, then append after the last link of the fixed template table
interval	if interval exists, then sections are created at the interval_value for the range of this command if interval is missing, then the interval for the Apply MTF is used
interval_value	only exists if *interval* exists
extra_start	optional - adds in an extra x-section 0.1 *mm* before start chainage
extra_end	optional - adds in an extra x-section 0.1 *mm* before end chainage
inactive	if *inactive* is there then the command is not processed. If inactive is not there then the command is processed
// text	the text is a comment

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

Fixed - Link Remove

Fixed links can be deleted by the **remove** command:

remove link_list st_ch end_ch *optional*

where

link_list	one or more names of links to be removed, in the form name1 name2 ... namei
st_ch	start chainage for removing the link
end_ch	end chainage for removing the link

and *optional* can be none, one or more of

interval	if interval exists, then sections are created at the interval_value for the range of this command if interval is missing, then the interval for the Apply MTF is used
interval_value	only exists if *interval* exists
extra_start	optional - adds in an extra x-section 0.1 *mm* before start chainage
extra_end	optional - adds in an extra x-section 0.1 *mm* before end chainage

inactive if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed

// text the text is a comment

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

Fixed - Width

The **width** modifier is used to modify the width of fixed links originally defined by width.

The format of the modifier to vary the **width** of the fixed links given by link_list is

width link_list st_ch end_ch st_wid end_wid absrel type optional

where

link_list one or more names of fixed links given in the template definition; with format name1 name2 ... namei

st_ch start chainage for the modifier
end_ch end chainage for the modifier

st_wid width to be used at the start modifier chainage, st_ch
end_wid width to be used at the end modifier chainage, end_ch.

absrel relative (default) or absolute

If **absrel** is **relative** (the default) then

the st_wid is added to the current width of links in link_list at chainage st_ch.
the end_wid is added to the current width of links in link_list at chainage end_ch.

If **absrel** is **absolute**, then

the width of links in link_list are set to st_wid at chainage st_ch.
the width of links in link_list are set to end_wid at chainage end_ch.

type linear (default) or cubic

If **type** is **linear** (the default), then the width of links in link_list are varied linearly (with respect to the reference chainage) between the width at chainage st_ch and chainage end_ch.

If **type** is **cubic**, then the width of links in link_list is varied as a reverse cubic (with respect to the reference chainage) between the width at chainage st_ch and chainage end_ch.

and *optional* can be none, one or more of

interval if interval exists, then sections are created at the interval_value for the range of this command
if interval is missing, then the interval for the Apply MTF is used

interval_value only exists if *interval* exists

extra_start optional - adds in an extra x-section 0.1mm before start chainage
extra_end optional - adds in an extra x-section 0.1mm before end chainage

inactive if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed

// text the text is a comment

The width for a link can also be varied by **going out to another *12d Model* string** or by taking the **width between two *12d Model* strings**.

width link_list st_ch end_ch full_string_name side optional
width link_list st_ch end_ch str_name_1 side_1 str_name_2 side_2 optional

where

link_list one or more names of fixed links given in the template definition;

	with format name1 name2 ... namei
st_ch	start chainage for the modifier
end_ch	end chainage for the modifier
full_string_name	name of a ***12d Model*** string to take links in link_list out to. The format of the string name is "model_name->string_name".
side	side to search for string: -1 for left, 0 for left and right, 1 for right
str_name_1	name of the first ***12d Model*** string. The format of the string name is "model_name->string_name".
side_1	side to search for string_1: -1 for left, 0 for left and right, 1 for right
str_name_2 name	name of the second ***12d Model*** string. The format of the string is "model_name->string_name".
side_2	side to search for string_2: -1 for left, 0 for left and right, 1 for right

and *optional* can be none, one or more of

interval	if interval exists, then sections are created at the interval_value for the range of this command if interval is missing, then the interval for the Apply MTF is used
interval_value	only exists if *interval* exists
extra_start	optional - adds in an extra x-section 0.1mm before start chainage
extra_end	optional - adds in an extra x-section 0.1mm before end chainage
inactive	if *inactive* is there then the command is not processed. If inactive is not there then the command is processed
// text	the text is a comment

The **width** of links in link_list are taken to be the distance between **str_name_1** and **str_name_2**.

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

Fixed- Width from Link

The **width from link** modifier is used to modify the width of fixed links originally defined by width to be the same as another link. That is, the width of the link is a copy of the width of another link.

The format of the modifier to vary the **width from link** of the fixed links given by link_list is

copy_width link_list st_ch end_ch from_link zone optional

where

link_list	one or more names of fixed links given in the template definition; with format name1 name2 ... namei
st_ch	start chainage for the modifier
end_ch	end chainage for the modifier
from_link	name of link to take the width from
zone	section of the template that the from_link is from (i.e. fixed, cut or fill)

and *optional* can be none, one or more of

interval	if interval exists, then sections are created at the interval_value for the range of this command if interval is missing, then the interval for the Apply MTF is used

interval_value only exists if *interval* exists

extra_start optional - adds in an extra x-section 0.1mm before start chainage
extra_end optional - adds in an extra x-section 0.1mm before end chainage

inactive if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed
// text the text is a comment

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

Fixed - Height

The **height** modifier is used to modify the height of fixed links originally defined by height.

The format of the modifier to vary the **height** of the fixed links in the **link_list** is almost identical to varying the width and is:

height link_list st_ch end_ch st_ht end_ht absrel type optional

where

link_list one or more names of fixed links given in the template definition; with format name1 name2 ... namei

st_ch start chainage for the modifier
end_ch end chainage for the modifier

st_ht height to be used at the start modifier chainage, st_ch
end_ht height to be used at the end modifier chainage, end_ch.

absrel relative (default) or absolute

If **absrel** is **relative** (the default) then

the st_ht is added to the current height of links in link_list at chainage st_ch.
the end_ht is added to the current height of links in link_list at chainage end_ch.

If **absrel** is **absolute**, then

the height of links in link_list are set to st_ht at chainage st_ch.
the height of links in link_list are set to end_ht at chainage end_ch.

type linear (default) or cubic

If **type** is **linear** (the default), then the height of links in link_list are varied linearly (with respect to the reference chainage) between the height at chainage st_ch and chainage end_ch.

If **type** is **cubic**, then the height of links in link_list is varied as a reverse cubic (with respect to the reference chainage) between the height at chainage st_ch and chainage end_ch.

and *optional* can be none, one or more of

interval if interval exists, then sections are created at the interval_value for the range of this command
if interval is missing, then the interval for the Apply MTF is used
interval_value only exists if *interval* exists

extra_start optional - adds in an extra x-section 0.1mm before start chainage
extra_end optional - adds in an extra x-section 0.1mm before end chainage

inactive if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed
// text the text is a comment

The height for a link can also be varied by **taking the height from a *12d Model* string** or by taking the **height between two *12d Model* strings**.

height link_list st_ch end_ch full_string_name side optional
height link_list st_ch end_ch str_name_1 side1 str_name_2 side2 optional

where

link_list — one or more names of fixed links given in the template definition; with format name1 name2 ... namei

st_ch — start chainage for the modifier
end_ch — end chainage for the modifier

full_string_name — name of a ***12d Model*** string to take links in link_list out to. The format of the string name is “model_name->string_name”.

side — side to search for string: -1 for left, 0 for left and right, 1 for right

str_name_1 — name of the first ***12d Model*** string. The format of the string name is “model_name->string_name”.

side2 — side to search for string: -1 for left, 0 for left and right, 1 for right

str_name_2 name — name of the second ***12d Model*** string. The format of the string is “model_name->string_name”.

side2 — side to search for string: -1 for left, 0 for left and right, 1 for right

and *optional* can be none, one or more of

interval — if interval exists, then sections are created at the interval_value for the range of this command
if interval is missing, then the interval for the Apply MTF is used
interval_value — only exists if *interval* exists

extra_start — optional - adds in an extra x-section 0.1mm before start chainage
extra_end — optional - adds in an extra x-section 0.1mm before end chainage

inactive — if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed
// text — the text is a comment

The **height** of links in link_list are taken to be the height between **str_name_1** and **str_name_2**.

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

Fixed - Height from Link

The **height from link** modifier is used to modify the height of fixed links originally defined by height to be the same height as another link. That is, the height of the link is a copy of the height of another link.

The format of the modifier to vary the **height from link** of the fixed links given by link_list is

copy_height link_list st_ch end_ch from_link zone optional

where

link_list — one or more names of fixed links given in the template definition; with format name1 name2 ... namei

st_ch — start chainage for the modifier
end_ch — end chainage for the modifier

from_link — name of link to take the height from
zone — section of the template that the from_ link is from (i.e. fixed, cut or fill)

and *optional* can be none, one or more of

interval — if interval exists, then sections are created at the interval_value for the range of this command
if interval is missing, then the interval for the Apply MTF is used
interval_value — only exists if *interval* exists

extra_start — optional - adds in an extra x-section 0.1mm before start chainage
extra_end — optional - adds in an extra x-section 0.1mm before end chainage

inactive — if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed
// text — the text is a comment

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

Fixed - Cross Fall

The **xfall** modifier is used to modify the cross fall of fixed links originally defined by xfall.

The format of the modifier to vary the cross-fall of the fixed links in the **link_list** is almost identical to varying the width and is:.

xfall link_list st_ch end_ch st_xfall end_xfall absrel type optional

where

link_list — one or more names of fixed links given in the template definition; with format
name1 name2 ... namei

st_ch — start chainage for the modifier
end_ch — end chainage for the modifier

st_xfall — xfall to be used at the start modifier chainage, st_ch
end_xfall — xfall to be used at the end modifier chainage, end_ch.

absrel — relative (default) or absolute

If **absrel** is **relative** (the default) then

the st_xfall is added to the current xfall of links in link_list at chainage st_ch.
the end_xfall is added to the current xfall of links in link_list at chainage end_ch.

If **absrel** is **absolute**, then

the xfall of links in link_list are set to st_xfall at chainage st_ch.
the xfall of links in link_list are set to end_xfall at chainage end_ch.

type — linear (default) or cubic

If **type** is **linear** (the default), then the xfall of links in link_list are varied linearly (with respect to the reference chainage) between the xfall at chainage st_ch and chainage end_ch.

cubic, then the xfall of links in link_list are varied as a reverse cubic (with respect to the reference chainage) between the xfall at chainage st_ch and chainage end_ch.

rotate, then the angle of the xfall of the links in link_list is varied linearly (with respect to the reference chainage) between the angle of the xfall at chainage st_ch and chainage end_ch.

and *optional* can be none, one or more of

interval — if interval exists, then sections are created at the interval_value for the range of this command
if interval is missing, then the interval for the Apply MTF is used
interval_value — only exists if *interval* exists

extra_start — optional - adds in an extra x-section 0.1mm before start chainage
extra_end — optional - adds in an extra x-section 0.1mm before end chainage

inactive — if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed

// text the text is a comment

The **xfall** of a link can be defined by using the xfall from the **beginning of the link** (i.e. the end of the previous link or the hinge string if it is the first link) **to another *12d Model***, or by taking the **xfall between two *12d Model* strings**.

xfall link_list st_ch end_ch full_string_name side optional
xfall link_list st_ch end_ch str_name_1 side1 str_name_2 side2 optional

where

link_list one or more names of fixed links given in the template definition; with format
name1 name2 ... namei

st_ch start chainage for the modifier
end_ch end chainage for the modifier
full_string_name name of a ***12d Model*** string to use to calculate xfall. The format of the string name is “model_name->string_name”.
side side to search for string: -1 for left, 0 for left and right, 1 for right
str_name_1 name of the first ***12d Model*** string. The format of the string name is “model_name->string_name”.
side1 side to search for string_1:
-1 for left, 0 for left and right, 1 for right

str_name_2 name of the second ***12d Model*** string.

side2 side to search for string_2:
-1 for left, 0 for left and right, 1 for right

and *optional* can be none, one or more of

interval if interval exists, then sections are created at the interval_value for the range of this command
if interval is missing, then the interval for the Apply MTF is used
interval_value only exists if *interval* exists
extra_start optional - adds in an extra x-section 0.1mm before start chainage
extra_end optional - adds in an extra x-section 0.1mm before end chainage

inactive if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed
// text the text is a comment

The **xfall** of links in link_list are taken to be the xfall between **str_name_1** and **str_name_2**.

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

Fixed - Cross Fall Circular Reverse Curve

The **xfall_crc** modifier is used to modify the cross fall of fixed links originally defined by **xfall** using the circular reverse curve formula.

The format of the modifier to vary the cross-fall of the fixed links (using circular reverse curves) in the **link_list** is:

xfall_crc link_list st_ch end_ch st_xfall end_xfall st_len end_len optional

where

link_list one or more names of fixed links given in the template definition; with format
name1 name2 ... namei

st_ch   start chainage for the modifier
end_ch   end chainage for the modifier

st_xfall   xfall to be used at the start modifier chainage, st_ch
end_xfall xfall to be used at the end modifier chainage, end_ch.

st_len   length of the start circular arc to be used at the start modifier chainage, st_ch.
end_len   length of the end circular arc to finish at the end modifier chainage, st_ch.

and *optional* can be none, one or more of

interval   if interval exists, then sections are created at the interval_value for the range of this command
if interval is missing, then the interval for the Apply MTF is used
interval_value   only exists if *interval* exists

extra_start   optional - adds in an extra x-section 0.1mm before start chainage
extra_end   optional - adds in an extra x-section 0.1mm before end chainage

inactive   if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed
// text   the text is a comment

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

Fixed - Xfall from Link

The **xfall from link** modifier is used to modify the xfall of fixed links originally defined by xfall to be the same xfall as another link. That is, the xfall of the link is a copy of the xfall of another link.

The format of the modifier to vary the **xfall from link** of the fixed links given by link_list is

copy_xfall   link_list   st_ch   end_ch   from_link   zone   optional

where

link_list   one or more names of fixed links given in the template definition; with format name1 name2 ... namei

st_ch   start chainage for the modifier
end_ch   end chainage for the modifier

from_link   name of the link to take the xfall from
zone   section of the template that the from_link is from (i.e. fixed, cut or fill)

and *optional* can be none, one or more of

interval   if interval exists, then sections are created at the interval_value for the range of this command
if interval is missing, then the interval for the Apply MTF is used
interval_value   only exists if *interval* exists

extra_start   optional - adds in an extra x-section 0.1mm before start chainage
extra_end   optional - adds in an extra x-section 0.1mm before end chainage

inactive   if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed
// text   the text is a comment

Important Note

The link to copy cross fall from can be defined in terms of cross fall or slope.

If the link to copy is defined by slope, then the cross fall is calculated to match the slope.

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

Fixed - Tin Height

The **tin height** modifier is used to modify the height of the link so that the link will sit on the tin at the given width.

The tin height can only be used for a fixed link defined by width and height.

The format of the modifier to vary the **tin height** of the fixed links in the **link_list** is:

tin_height   link_list   st_ch   end_ch   tin_name   optional

where

link_list   one or more names of fixed links given in the template definition; with format name1 name2 ... namei

st_ch   start chainage for the modifier
end_ch   end chainage for the modifier

tin_name   name of the tin to be used in defining the height.

and *optional* can be none, one or more of

interval   if interval exists, then sections are created at the interval_value for the range of this command
if interval is missing, then the interval for the Apply MTF is used
interval_value   only exists if *interval* exists

extra_start   optional - adds in an extra x-section 0.1mm before start chainage
extra_end   optional - adds in an extra x-section 0.1mm before end chainage

inactive   if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed
// text   the text is a comment

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

Fixed - Tin Xfall

The **tin xfall** modifier is used to modify the xfall of the link so that the link will sit on the tin at the given width.

The tin xfall can only be used for a fixed link defined by width and xfall.

The format of the modifier to vary the **tin xfall** of the fixed links in the **link_list** is:

tin_xfall   link_list   st_ch   end_ch   tin_name

where

link_list   one or more names of fixed links given in the template definition; with format name1 name2 ... namei

st_ch   start chainage for the modifier
end_ch   end chainage for the modifier

tin_name   name of the tin to be used in defining the xfall.

and *optional* can be none, one or more of

interval   if interval exists, then sections are created at the interval_value for the range of this command
if interval is missing, then the interval for the Apply MTF is used
interval_value   only exists if *interval* exists

extra_start   optional - adds in an extra x-section 0.1mm before start chainage
extra_end   optional - adds in an extra x-section 0.1mm before end chainage

inactive   if *inactive* is there then the command is not processed.

	If inactive is not there then the command is processed
// text	the text is a comment

Please continue to the next section 21.10.1.7 Decision File Format.

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

21.10.1.7 Decision File Format

Each template is made up of the sections fixed, decisions, cut, fill and final cut/fill. If a template has a decisions section then by default it is used instead of the cut, fill and final cut/fill sections.

Hence by default, either the
decisions section

or
the cut, fill and final cut/fill sections

from the template being used over the chainage range.

Using the **decision** modifier, it is possible to override the default and for a given chainage range
use the cut, fill and final cut/fill sections **instead** of the decisions section

or
use the decisions section **from another** template instead of the current templates sections.

The format of the decision modifier is

decision template_name st_ch end_ch *optional*

decision " " st_ch end_ch *optional*

where

template_name name of template to take the decision section from.

If the template name is given as **" "**, then the decision section for the template being used in the chainage range is ignored, and the cut, fill and final cut/fill tables for the template are used instead. Hence this overrides the default of using the decisions section in preference to the cut, fill and final cut/fill sections of the template.

st_ch start chainage for applying/removing the decision
end_ch end chainage for applying/removing the decision

and *optional* can be none, one or more of

interval if interval exists, then sections are created at the interval_value for the range of this command
if interval is missing, then the interval for the Apply MTF is used
interval_value only exists if *interval* exists

extra_start optional - adds in an extra x-section 0.1mm before start chainage
extra_end optional - adds in an extra x-section 0.1mm before end chainage

inactive if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed
// text the text is a comment

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

Note

The full description of the **Decisions** section of a template is given in 21.8 Full Definition of Template Decisions.

Please continue to the next section 21.10.1.8 Cut and Fill Variable Link Modifiers File Format.

21.10.1.8 Cut and Fill Variable Link Modifiers File Format

Cut and Fill Link Insert

Cut and fill links can be created by one of three **insert** commands by specifying either width and height, width and slope, or height and slope.

insert_cut link_name colour width height **unknown** st_ch end_ch *optional // width, ht*
insert_fill link_name colour width height **unknown** st_ch end_ch *optional // width, ht*

insert_cut link_name colour width **unknown** slope st_ch end_ch *optional // width, slope*
insert_fill link_name colour width **unknown** slope st_ch end_ch *optional // width, slope*

insert_cut link_name colour **unknown** height slope st_ch end_ch *optional // ht, slope*
insert_fill link_name colour **unknown** height slope st_ch end_ch *optional // ht, slope*

where **unknown** takes the place of the one of width, height or slope not being used, and

link_name	name of the link being created
colour	colour of the link being created
width, height, slope	width, height or slope of the created link
st_ch	start chainage for creating the link
end_ch	end chainage for creating the link

and *optional* can be none, one or more of the following

name	if non-blank, insert before the link name in the template if blank, then append after the last link of the fixed template table
interval	if interval exists, then sections are created at the interval_value for the range of this command if interval is missing, then the interval for the Apply MTF is used
interval_value	only exists if *interval* exists
extra_start	optional - adds in an extra x-section 0.1mm before start chainage
extra_end	optional - adds in an extra x-section 0.1mm before end chainage
inactive	if *inactive* is there then the command is not processed. If inactive is not there then the command is processed
// text	the text is a comment

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

Cut and Fill Link Remove

Cut and fill links can be deleted by the **remove** commands:

remove_cut link_list st_ch end_ch *optional*

remove_fill link_list st_ch end_ch *optional*

where

link_list	one or more names of links to be removed, in the form name1 name2 ... namei
st_ch	start chainage for removing the link
end_ch	end chainage for removing the link

and *optional* can be none, one or more of

interval	if interval exists, then sections are created at the interval_value for the range of this command

	if interval is missing, then the interval for the Apply MTF is used
interval_value	only exists if *interval* exists
extra_start	optional - adds in an extra x-section 0.1mm before start chainage
extra_end	optional - adds in an extra x-section 0.1mm before end chainage
inactive	if *inactive* is there then the command is not processed. If inactive is not there then the command is processed
// text	the text is a comment

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

Cut and Fill Width

The **cut_width** modifier is used to modify the width of variable cut links originally defined by width. Similarly the **fill_width** modifier is used to modify the width of variable fill links originally defined by width.

The format of the modifier to vary the width of the variable cut/fill links in **link_list** is

cut_width link_list st_ch end_ch st_wid end_wid absrel type optional
fill_width link_list st_ch end_ch st_wid end_wid absrel type optional

where

link_list	one or more names of cut/fill links given in the template definition; with format name1 name2 ... namei
st_ch	start chainage for the modifier
end_ch	end chainage for the modifier
st_wid	width to be used at the start modifier chainage, st_ch
end_wid	width to be used at the end modifier chainage, end_ch.
absrel	relative (default) or absolute

If **absrel** is **relative** (the default) then

the st_wid is added to the current width of links in link_list at chainage st_ch.
the end_wid is added to the current width of links in link_list at chainage end_ch.

If **absrel** is **absolute**, then

the width of links in link_list are set to st_wid at chainage st_ch.
the width of links in link_list are set to end_wid at chainage end_ch.

type linear (default) or cubic

If **type** is **linear** (the default), then the width of links in link_list are varied linearly (with respect to the reference chainage) between the width at chainage st_ch and chainage end_ch.

If **type** is **cubic**, then the width of links in link_list are varied as a reverse cubic (with respect to the reference chainage) between the width at chainage st_ch and chainage end_ch.

and *optional* can be none, one or more of

interval	if interval exists, then sections are created at the interval_value for the range of this command if interval is missing, then the interval for the Apply MTF is used
interval_value	only exists if *interval* exists
extra_start	optional - adds in an extra x-section 0.1mm before start chainage
extra_end	optional - adds in an extra x-section 0.1mm before end chainage
inactive	if *inactive* is there then the command is not processed. If inactive is not there then the command is processed
// text	the text is a comment

The **width** for a variable link can also be varied by **going out to another *12d Model* string**.

cut_width link_list st_ch end_ch full_string_name side optional
fill_width link_list st_ch end_ch full_string_name side optional

where

link_list one or more names of cut/fill links given in the template definition; with format
name1 name2 ... namei

st_ch start chainage for the modifier
end_ch end chainage for the modifier

full_string_name name of a ***12d Model*** string to take links in link_list out to. The format of the string name is “model_name->string_name”.

side side to search for string: -1 for left, 0 for left and right, 1 for right

and *optional* can be none, one or more of

interval if interval exists, then sections are created at the interval_value for the range of this command
if interval is missing, then the interval for the Apply MTF is used
interval_value only exists if *interval* exists

extra_start optional - adds in an extra x-section 0.1mm before start chainage
extra_end optional - adds in an extra x-section 0.1mm before end chainage

inactive if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed
// text the text is a comment

The **width** for a link can also be varied by using the **width between two *12d Model* strings str_name_1** and **str_name_2**.

cut_width link_list st_ch end_ch str_name_1 side1 str_name_2 side2 optional

fill_width link_list st_ch end_ch str_name_1 side1 str_name_2 side2 optional

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

Cut and Fill Width from Link

The cut and fill **width from link** modifiers are used to modify the width of cut and fill links originally defined by width, to be the same as another link. That is, the width of the link is a copy of the width of another link.

The format of the modifier to vary the cut and fill **width from link** of the fixed links given by link_list is

copy_cut_width link_list st_ch end_ch from_link zone optional
copy_fill_width link_list st_ch end_ch from_link zone optional

where

link_list one or more names of fixed links given in the template definition; with format
name1 name2 ... namei

st_ch start chainage for the modifier
end_ch end chainage for the modifier

from_link name of the link to take the width from
zone section of the template that the from_link is from (i.e. fixed, cut or fill)

and *optional* can be none, one or more of

interval — if interval exists, then sections are created at the interval_value for the range of this command
if interval is missing, then the interval for the Apply MTF is used
interval_value — only exists if *interval* exists

extra_start — optional - adds in an extra x-section 0.1mm before start chainage
extra_end — optional - adds in an extra x-section 0.1mm before end chainage

inactive — if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed
// text — the text is a comment

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

Cut and Fill Height

The **cut_height** modifier is used to modify the height of variable cut links originally defined by height. Similarly the **fill_height** modifier is used to modify the height of variable fill links originally defined by height.

The format of the modifier to vary the height of the variable cut/fill links in **link_list** is

cut_height link_list st_ch end_ch st_ht end_ht absrel type optional
fill_height link_list st_ch end_ch st_ht end_ht absrel type optional

where

link_list — one or more names of cut/fill links given in the template definition; with format
name1 name2 ... namei

st_ch — start chainage for the modifier
end_ch — end chainage for the modifier

st_ht — height to be used at the start modifier chainage, st_ch
end_ht — height to be used at the end modifier chainage, end_ch.

absrel — relative (default) or absolute

If **absrel** is **relative** (the default) then

the st_ht is added to the current height of links in link_list at chainage st_ch.
the end_ht is added to the current height of links in link_list at chainage end_ch.

If **absrel** is **absolute**, then

the height of links in link_list are set to st_ht at chainage st_ch.
the height of links in link_list are set to end_ht at chainage end_ch.

type — linear (default) or cubic

If **type** is **linear** (the default), then the height of links in link_list are varied linearly (with respect to the reference chainage) between the height at chainage st_ch and chainage end_ch.

If **type** is **cubic**, then the height of links in link_list are varied as a reverse cubic (with respect to the reference chainage) between the height at chainage st_ch and chainage end_ch.

and *optional* can be none, one or more of

interval — if interval exists, then sections are created at the interval_value for the range of this command
if interval is missing, then the interval for the Apply MTF is used
interval_value — only exists if *interval* exists

extra_start — optional - adds in an extra x-section 0.1mm before start chainage extra_end optional - adds in an extra x-section 0.1mm before end chainage

inactive if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed

// text the text is a comment

The **height** for a variable link can also be varied by **taking the height from a *12d Model* string**.

cut_height link_list st_ch end_ch full_string_name side optional
fill_height link_list st_ch end_ch full_string_name side optional

where

link_list one or more names of cut/fill links given in the template definition; with format name1 name2 ... namei

st_ch start chainage for the modifier
end_ch end chainage for the modifier

full_string_name name of a ***12d Model*** string to take links in link_list out to. The format of the string name is “model_name->string_name”.

side side to search for string: -1 for left, 0 for left and right, 1 for right

and *optional* can be none, one or more of

interval if interval exists, then sections are created at the interval_value for the range of this command
if interval is missing, then the interval for the Apply MTF is used

interval_value only exists if *interval* exists

extra_start optional - adds in an extra x-section 0.1mm before start chainage
extra_end optional - adds in an extra x-section 0.1mm before end chainage

inactive if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed

// text the text is a comment

The **height** for a link can also be varied by using the **height between two *12d Model* strings str_name_1** and **str_name_2**.

cut_height link_list st_ch end_ch str_name_1 side1 str_name_2 side2 optional
fill_height link_list st_ch end_ch str_name_1 side1 str_name_2 side2 optional

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

Cut and Fill Height from Link

The cut and fill **height from link** modifiers are used to modify the height of cut and fill links originally defined by height, to be the same as another link. That is, the height of the link is a copy of the height of another link.

The format of the modifier to vary the cut and fill **height from link** of the fixed links given by link_list is

copy_cut_height link_list st_ch end_ch from_link zone optional
copy_fill_height link_list st_ch end_ch from_link zone optional

where

link_list one or more names of fixed links given in the template definition; with format name1 name2 ... namei

st_ch start chainage for the modifier
end_ch end chainage for the modifier

from_link name of the link to take the height from
zone section of the template that the from_link is from (i.e. fixed, cut or fill)

and *optional* can be none, one or more of

interval if interval exists, then sections are created at the interval_value for the range of this command
if interval is missing, then the interval for the Apply MTF is used
interval_value only exists if *interval* exists

extra_start optional - adds in an extra x-section 0.1mm before start chainage
extra_end optional - adds in an extra x-section 0.1mm before end chainage

inactive if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed
// text the text is a comment

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

Cut and Fill Slope

The **cut_slope** modifier is used to modify the slope of variable cut links originally defined by slope. Similarly the **fill_slope** modifier is used to modify the slope of variable fill links originally defined by slope.

The format of the modifier to vary the cut/fill slope of the variable links in **link_list** is

cut_slope link_list st_ch end_ch st_slope end_slope absrel type optional
fill_slope link_list st_ch end_ch st_slope end_slope absrel type optional

where

link_list one or more names of cut/fill links given in the template definition; with format
name1 name2 ... namei

st_ch start chainage for the modifier
end_ch end chainage for the modifier

st_slope slope to be used at the start modifier chainage, st_ch
end_slope slope to be used at the end modifier chainage, end_ch.

absrelrelative (default) or absolute
If **absrel** is **relative** (the default) then

the st_slope is added to the current slope of links in link_list at chainage st_ch.
the end_slope is added to the current slope of links in link_list at chainage end_ch.

If **absrel** is **absolute**, then

the slope of links in link_list is set to st_slope at chainage st_ch.
the slope of links in link_list is set to end_slope at chainage end_ch.

type linear (default) or cubic

If **type** is **linear** (the default), then the slope of links in link_list is varied linearly (with respect to the reference chainage) between the slope at chainage st_ch and chainage end_ch.

cubic, then the angle of the slope of links in link_list are varied as a reverse cubic (with respect to the reference chainage) between the angle of the slope at chainage st_ch and chainage end_ch.

rotate, then the angle of the slope of the links in link_list are varied linearly (with respect to the reference chainage) between the angle of the slope at chainage st_ch and chainage end_ch.

and *optional* can be none, one or more of

interval — if interval exists, then sections are created at the interval_value for the range of this command
if interval is missing, then the interval for the Apply MTF is used

interval_value — only exists if *interval* exists

extra_start — optional - adds in an extra x-section 0.1mm before start chainage

extra_end — optional - adds in an extra x-section 0.1mm before end chainage

inactive — if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed

// text — the text is a comment

The **slope** of a link can also be defined by using the slope from the **beginning of the link** (i.e. the end of the previous link or the hinge string if its the first link) to a given ***12d Model*** string.

cut_slope link_list st_ch end_ch full_string_name side optional
fill_slope link_list st_ch end_ch full_string_name side optional

where

link_list — one or more names of cut/fill links given in the template definition; with format
name1 name2 ... namei

st_ch — start chainage for the modifier

end_ch — end chainage for the modifier

full_string_name — name of a ***12d Model*** string to use to calculate slope. The format of the string name is “model_name->string_name”.

side — side to search for string: -1 for left, 0 for left and right, 1 for right

and *optional* can be none, one or more of

interval — if interval exists, then sections are created at the interval_value for the range of this command
if interval is missing, then the interval for the Apply MTF is used

interval_value — only exists if *interval* exists

extra_start — optional - adds in an extra x-section 0.1mm before start chainage

extra_end — optional - adds in an extra x-section 0.1mm before end chainage

inactive — if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed

// text — the text is a comment

The **slope** of a link can also be defined by using the slope **between two *12d Model* strings str_name_1** and **str_name_2**.

cut_slope link_list st_ch end_ch str_name_1 side1 str_name_2 optional
fill_slope link_list st_ch end_ch str_name_1 side1 str_name_2 side2 optional

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

Cut and Fill Slope from Link

The cut and fill **slope from link** modifiers are used to modify the slope of cut and fill links originally defined by slope, to be the same **slope** or **xfall** as another link. That is, the slope of the link is a copy of the slope of another link.

The format of the modifier to vary the cut and fill **slope from link** of the fixed links given by link_list is

copy_cut_slope link_list st_ch end_ch from_link zone optional
copy_fill_slope link_list st_ch end_ch from_link zone optional

where

link_list one or more names of fixed links given in the template definition; with format name1 name2 ... namei

st_ch start chainage for the modifier
end_ch end chainage for the modifier

from_link name of the link to take the slope or xfall from
zone section of the template that the from_link is from (i.e. fixed, cut or fill)

and *optional* can be none, one or more of

interval if interval exists, then sections are created at the interval_value for the range of this command
if interval is missing, then the interval for the Apply MTF is used
interval_value only exists if *interval* exists

extra_start optional - adds in an extra x-section 0.1mm before start chainage
extra_end optional - adds in an extra x-section 0.1mm before end chainage

inactive if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed
// text the text is a comment

Important Note

The link to copy slope from can be defined in terms of slope or **cross fall**.

If the link to copy is defined by cross fall, then the slope is calculated to match the cross fall.

Hence this command can be used with cut and fill slopes to match to rotation of cross fall of fixed links for super-elevation.

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

Cut and Fill Tin Height

The **cut and fill tin height** modifiers are used to modify the height of the link so that the link will sit on the tin at the given width.

The tin height can only be used for a cut or fill link defined by width and height.

The format of the modifiers to vary the **cut** and **fill tin height** of the links in the **link_list** are:

tin_cut_height link_list st_ch end_ch tin_name optional
tin_fill_height link_list st_ch end_ch tin_name optional

where

link_list one or more names of cut/fill links given in the template definition; with format name1 name2 ... namei

st_ch start chainage for the modifier
end_ch end chainage for the modifier

tin_name name of the tin to be used in defining the height.

and *optional* can be none, one or more of

interval if interval exists, then sections are created at the interval_value for the range of this command
if interval is missing, then the interval for the Apply MTF is used

interval_value only exists if *interval* exists

extra_start optional - adds in an extra x-section 0.1mm before start chainage
extra_end optional - adds in an extra x-section 0.1mm before end chainage

inactive if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed
// text the text is a comment

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

Cut and Fill Tin Slope

The **cut and fill tin slope** modifier is used to modify the slope of the link so that the link will sit on the tin at the given width.

The cut and fill tin slope can only be used for a cut/fill link defined by width and slope.

The format of the modifiers to vary the **cut** and **fill tin slope** of the links in the **link_list** are:

tin_cut_slope link_list st_ch end_ch tin_name optional
tin_fill_slope link_list st_ch end_ch tin_name optional

where

link_list one or more names of fixed links given in the template definition; with format name1 name2 ... namei

st_ch start chainage for the modifier
end_ch end chainage for the modifier

tin_name name of the tin to be used in defining the slope.

and *optional* can be none, one or more of

interval if interval exists, then sections are created at the interval_value for the range of this command
if interval is missing, then the interval for the Apply MTF is used
interval_value only exists if *interval* exists

extra_start optional - adds in an extra x-section 0.1mm before start chainage
extra_end optional - adds in an extra x-section 0.1mm before end chainage

inactive if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed
// text the text is a comment

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

Please continue to the next section 21.10.1.9 Final Link Modifiers File Format.

21.10.1.9 Final Link Modifiers File Format

Final Width

The format of the modifier to **vary the width** of the final cut and fill link link_name is

final_width link_name st_ch end_ch st_wid end_wid absrel type optional

where

link_name name of final link given in the template definition

st_ch start chainage for the modifier
end_ch end chainage for the modifier

st_wid width to be used at the start modifier chainage, st_ch
end_wid width to be used at the end modifier chainage, end_ch.

absrel relative (default) or absolute

If **absrel** is **relative** (the default) then

the st_wid is added to the current width of link_name at chainage st_ch.
the end_wid is added to the current width of link_name at chainage end_ch.

If **absrel** is **absolute**, then

the width of link_name is set to st_wid at chainage st_ch.
the width of link_name is set to end_wid at chainage end_ch.

type linear (default) or cubic

If **type** is **linear** (the default), then the width of link_name is varied linearly (with respect to the reference chainage) between the width at chainage st_ch and chainage end_ch.

If **type** is **cubic**, then the width of link_name is varied as a reverse cubic (with respect to the reference chainage) between the width at chainage st_ch and chainage end_ch.

and *optional* can be none, one or more of

interval if interval exists, then sections are created at the interval_value for the range of this command
if interval is missing, then the interval for the Apply MTF is used
interval_value only exists if *interval* exists

extra_start optional - adds in an extra x-section 0.1mm before start chainage
extra_end optional - adds in an extra x-section 0.1mm before end chainage

inactive if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed
// text the text is a comment

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

Final Cut and Fill Slope

The format of the modifier to vary the cut/fill slope of the final link **link_name** is

final_cut_slope link_name st_ch end_ch st_slope end_slope absrel type optional
final_fill_slope link_name st_ch end_ch st_slope end_slope absrel type optional

where

link_name name of final link given in the template definition

st_ch start chainage for the modifier
end_ch end chainage for the modifier

st_slope slope to be used at the start modifier chainage, st_ch
end_slope slope to be used at the end modifier chainage, end_ch.

absrel relative (default) or absolute

If **absrel** is **relative** (the default) then

the st_slope is added to the current slope of link_name at chainage st_ch.

the end_slope is added to the current slope of link_name at chainage end_ch.

If **absrel** is **absolute**, then

the slope of link_name is set to st_slope at chainage st_ch.

the slope of link_name is set to end_slope at chainage end_ch.

type linear (default) or cubic

If **type** is **linear** (the default), then the slope of link_name is varied linearly (with respect to the reference chainage) between the slope at chainage st_ch and chainage end_ch.

cubic, then the angle of the slope of link_name is varied as a reverse cubic (with respect to the reference chainage) between the angle of the slope at chainage st_ch and chainage end_ch.

and *optional* can be none, one or more of

interval if interval exists, then sections are created at the interval_value for the range of this command
if interval is missing, then the interval for the Apply MTF is used
interval_value only exists if *interval* exists

extra_start optional - adds in an extra x-section 0.1mm before start chainage
extra_end optional - adds in an extra x-section 0.1mm before end chainage

inactive if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed
// text the text is a comment

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

Stopping Final Cut and Fill Slope

The final link can be stopped altogether in either cut or fill.

The format of the modifier to stop the cut/fill slope of the final link link_name is

final_no_cut_slope link_name st_ch end_ch optional
final_no_fill_slope link_name st__ch end_ch optional

where

link_name name of final link given in the template definition

st_ch start chainage for the final link to stop
end_ch end chainage for the final link to stop

and *optional* can be none, one or more of

interval if interval exists, then sections are created at the interval_value for the range of this command
if interval is missing, then the interval for the Apply MTF is used
interval_value only exists if *interval* exists

extra_start optional - adds in an extra x-section 0.1mm before start chainage
extra_end optional - adds in an extra x-section 0.1mm before end chainage

inactive if *inactive* is there then the command is not processed.
If inactive is not there then the command is processed
// text the text is a comment

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

Please continue to the next section 21.10.1.10 Snippets File Format.

21.10.1.10 Snippets File Format

Snippets are a method of inserting groups of modifiers at once. Snippets can have parameters but the values of the parameters are saved in the actual MTF.

snippet snippet_file st_ch end_ch param_name_1 param_value_1**...**
param_name_n param_value_n *optional*

where **unknown** takes the place of the one of width, height or xfall not being used, and

snippet_file	file containing the snippet definition
st_ch	start chainage for creating the link
end_ch	end chainage for creating the link
param_name_n	name of the nth parameter - must be in quotes (")
param_value_n	value for the nth parameter - must be in quotes

and *optional* can be none, one or more of

interval	if interval exists, then sections are created at the interval_value for the range of this command if interval is missing, then the interval for the Apply MTF is used
interval_value	only exists if *interval* exists

extra_start optional - adds in an extra x-section 0.1mm before start chainage
extra_end optional - adds in an extra x-section 0.1mm before end chainage

inactive	if *inactive* is there then the command is not processed. If inactive is not there then the command is processed
// text	the text is a comment

See 21.10.1.5 Start and End Chainages in the Modifier File Format for the format for st_ch and end_ch.

For more information on the inserting and use of snippets, see 21.5 Defining and Using Snippets.

Please continue to the next section 21.10.1.11 A Template Modifier File Example.

21.10.1.11 A Template Modifier File Example

```
left_side   =  {
   -99999.9     "std"
    99999.9
}
right_side   =  {
  -99999.9     "std"
   99999.9
}
//                                                    -- assumes ---
left_side_modifier   =  {
   width   "kerb"   103   203.0   0.0   3.0   absolute linear// widen from width 0 to width 3 - linear
   width    "kerb"   203.0   303.0   3.0   3.0   absolute          // keep width at 3
   width   "kerb"   303.0   403.0   3.0   0.0   absolute           // widen from width 3 to width 0 - linear
   width   "kerb"   153.0   253.0   0.0   9.0   cubic   relative
   width   "kerb"   253.0   353.0   9.0   0.0   cubic              // relative is default
   xfall   "kerb"   53.0   153.0   0.0   6.0   relative   linear
   xfall   "kerb"   153.0   353.0   6.0   6.0   relative           // linear
   xfall   "kerb"   353.0   453.0   6.0   0.0                      // relative & linear
   width    "kerb"   440.0   480.0   "table drain->table drain"   extra_start   extra_end
   xfall   "kerb"   440.0   480.0   "table drain->table drain"   extra_start   extra_end
   width   "kerb"   440.0   480.0   1.0   1.0   relative
   cut_width   "a"   103.0   203.0    0.0    9.0
   cut_width   "a"   203.0   303.0    9.0    9.0
   cut_width   "a"   303.0   403.0    9.0    0.0
   cut_slope   "a"   103.0   203.0    0.0    3.0
   cut_slope   "a"   203.0   303.0    3.0    3.0
   cut_slope   "a"   303.0   403.0    3.0    0.0
   cut_width   "a"   500.0   520.0   "table drain->table drain"   extra_start   extra_end
   cut_slope   "a"   500.0   520.0   "table drain->table drain"   extra_start   extra_end
   fill_width   "a"   103.0   203.0    0.0   9.0
   fill_width   "a"   203.0   303.0    9.0    9.0
   fill_width   "a"   303.0   403.0    9.0    0.0
   fill_slope   "a"   103.0   203.0    0.0    3.0
   fill_slope   "a"   203.0   303.0    3.0    3.0
   fill_slope   "a"   303.0   403.0    3.0    0.0
   fill_width   "a"   560.0   610.0   "table drain->table drain"   extra_start   extra_end
   fill_slope   "a"   560.0   610.0   "table drain->table drain"   extra_start   extra_end
   final_width   "final"   103.0   403.0    100.0    50.0
   final_cut_slope    "final"   103.0   403.0    0.0   3.0
   final_fill_slope    "final"    103.0   403.0    0.0   2.0
   final_no_cut_slope    "final"   190.0   310.0    extra_start    extra_end
   final_no_fill_slope    "final"   190.0   310.0    extra_start    extra_end
}
right_side_modifier = {
}
```

Go to the next section 21.10.1.12 MTF Stripping File Format or back to 21.10.1 Text Format of the MTF File

21.10.1.12 MTF Stripping File Format

The stripping depth is given in the Modifiers and Templates File (*.mtf) and is similar to the format for applying templates except there is no separate left and right side.

The stripping definition begins with the key words

stripping =

A list of chainages (in ascending order, one per line) with corresponding stripping depth then follows. The list of chainages and stripping depths is enclosed in curly braces {}.

The chainage-stripping lists are assembled as follows

(a) to represent a stripping depth starting at a given chainage, the chainage value followed by the stripping depth is given. The chainage and stripping depth are separated by one or more spaces. For example, the stripping depth **0.03** starting at chainage 150 is represented by

```
150        0.03
```

The stripping depth is assumed to apply until the chainage given on the next line of the stripping definition.

If the stripping depth is to go to the end of the reference string, add a line with a chainage greater than or equal to the end chainage. For example

```
150        0.03
9999       0.0
```

Warning - unlike applying templates, a stripping depth **must exist** after a chainage, however the stripping depth can be zero.

(b) linearly varying stripping depth between chainages is represented by giving the **start chainage** for the variation followed on the same line by the **start depth** and the **end depth** separated by a comma.

The linear change takes place over the interval beginning at this chainage and ending at the chainage given on the next line of the stripping definition.

For example, to linearly vary the stripping depth from 0.02 to 0.03 between the chainages 350 and 500, use

```
350        0.02 , 0.03
500        0.03
```

A Stripping Example

An example of a stripping command is

```
stripping = {
          -999      0.03
          30        0.0
          70        0.03,0.05
          110.5     0.05
          999       0.05
      }
```

Go to the next section 21.10.1.13 MTF Boxing File Format or back to 21.10.1 Text Format of the MTF File

21.10.1.13 MTF Boxing File Format

Apart from a key word specifying whether the following part of the definition is for the left or the right, the set out for the left boxing is identical to the right boxing. Hence only the left boxing will be described in detail.

The left boxing definition begins with the key words

left_boxing =

A list of chainages (in ascending order, one per line) with corresponding boxing names then follows. The list of chainages and boxing is enclosed in curly braces **{ }**.

The chainage-boxing lists are assembled as follows

(a) to represent a boxing starting at a given chainage, the chainage value followed by the boxing name is given. The chainage and boxing name are separated by one or more spaces. For example, the boxing **left** starting at chainage 150 is represented by

```
150        left
```

The boxing is assumed to apply until the chainage given on the next line of the left_boxing definition.

(b) if no boxing exists from a chainage, simply include the chainage with no boxing name following it. For example, if there is no boxing from chainage 250, this is represented by

```
250
```

The non-existence of boxing is assumed until the chainage given on the next line of the left_side definition.

If the boxing is to go to the end of the reference string, add a line with a chainage greater than or equal to the end chainage. For example

```
150        left
9999
```

When using the **Apply MTF** option, the name of the boxing definitions file is given by a **boxing_file** command **in the Modifiers and Templates File**.

```
boxing_file  =          boxing_definitions_file
```

When using the **boxing many** option, if a file name is given in the **boxing file** panel field, it is used for the boxing definitions file, otherwise the boxing_file command in the Modifiers and Templates File is used.

Note - if the boxing name includes spaces, then the name must be enclosed in quotes **“**. For example, “left 1".

An Applying Boxing Example

```
boxing_file  = “boxing_for_client.bf”
left_boxing  = {
                    -999        left
                    30
                    70          “left narrow”
                    110.5       left
                    999
             }
right_boxing  = {
                    -999        right
                    30
                    70          “right narrow”
                    110.5       right
```

999
}

Warning - no interpolation or modifiers exist for boxing.

Go to the next section 21.10.1.14 MTF Width File Format or back to 21.10.1 Text Format of the MTF File

21.10.1.14 MTF Width File Format

The format for **section width** in the mtf file is simply

section_width = *value* // default is 10000

where *value* is the distance to search from the hinge string for strings required in some options. If section_width is missing, then it takes the default value of 10000.

For example

section_width = 100.0

Go to the next section 21.10.1.15 MTF String Modifiers File Format or back to 21.10.1 Text Format of the MTF File

21.10.1.15 MTF String Modifiers File Format

In the Modifiers and Templates File, the **string modifiers** definition begins with the key word

string_modifiers =

followed by one or more occurrences of the string modifier command **height** enclosed by curly braces **{ })**

The definition of the string modifier commands will now be given.

Height

The *height* the **Height** string modifier calculates for the amend point is given by:

(a) if the absolute flag is not set, the height of the hinge point, otherwize zero.

plus

(b) the interpolated height for the user given heights at the start and end chainages

plus

(c) the interpolated height for the user given cross falls at the start and end chainage.

That is, if the absolute flag is not set:

new height = hinge height + (interpolated height) + (interpolated xfall) x (offset distance)

If the absolute flag is set:

new height = (interpolated height) + (interpolated xfall) x (offset distance)

The format of the **Height** string modifier is

height amend_str ref_str hinge_str st_ch end_ch ch_int st_xfall end_xfall st_ht end_ht absolute

where

amend_str	name of a ***12d Model*** string to take the hinge point out to. The format of the string name is "model_name->string_name".
ref_str	name of reference string. The format of the string name is "model_name->string_name".
hinge_str	name of hinge string. The format of the string name is "model_name->string_name".
st_ch	start chainage for the modifier
end_ch	end chainage for the modifier
ch_int	interval between chainage points - can be the word *null*
st_xfall	xfall to be used at the start modifier chainage, st_ch
end_xfall	xfall to be used at the end modifier chainage, end_ch.
st_ht	height to be used at the start modifier chainage, st_ch
end_ht	height to be used at the end modifier chainage, end_ch.
absolute	*optional* - if it exists, include the height of the hinge string

Height and Two Strings

The *height* the **Height 2x** string modifier calculates for the amend point is given by:

(a) the height of the hinge point

plus

(b) the offset distance multiplied by the cross fall between two user selected strings.

new height = hinge height + (xfall between string 1 and string 2) x (offset distance)

The format of the **Height 2x** string modifier is

height amend_str ref_str hinge_str st_ch end_ch ch_int xfall_str1 xfall_str2

where

amend_str name of a ***12d Model*** string to take the hinge point out to. The format of the string name is "model_name->string_name".

ref_str name of reference string. The format of the string name is "model_name->string_name".

hinge_str name of hinge string. The format of the string name is "model_name->string_name".

st_ch start chainage for the modifier

end_ch end chainage for the modifier

ch_int interval between chainage points - can be the word *null*

xfall_str_1 name of first string to calculate cross fall between. The format of the string name is "model_name->string_name".

xfall_str_2 name of second string to calculate cross fall between. The format of the string name is "model_name->string_name".

Go to the next section 21.10.1.16 Substitutions in the Modifiers and Templates File or back to 21.10.1 Text Format of the MTF File

21.10.1.16 Substitutions in the Modifiers and Templates File

The **many template variables** are special variables standing for information about the reference string used in the Apply MTF operation.

The many template variables are only evaluated when the many template file is used in an **Apply MTF** operation and provide a mechanism to generalize the many template file.

The many template file also recognizes the CCCP preprocessor rules including #include, #define, #if etc.

Many Template Variables

The list of the many template variables is:

$tangent_curve_n	chainage of the tangent curve point for the nth HIP
$tangent_spiral_n	tangent spiral
$curve_spiral_n	curve spiral
$spiral_curve_n	spiral curve
$curve_spiral_n	curve spiral
$spiral_tangent_n	spiral tangent
$curve_tangent_n	curve tangent
$start_spiral_n	length of the start spiral on the nth HIP point
$end_spiral_n	length of the end spiral
$radius_n	absolute radius of the curve
$signed_radius_n	signed radius of the curve
$vertical_tangent_curve_n	chainage of the tangent curve point for the nth VIP
$vertical_curve_tangent_n	chainage of the curve tangent point for the nth VIP
$vertical_sag_n	chainage of the sag point for the nth HIP
$vertical_crest_n	crest
$vertical_crest_sag_n	crest or sag

For example,

$tangent_spiral_3   get the chainage for the tangent spiral on HIP 3.

The variables can be used in expressions with the arithmetic operations +, -, * and / as long as the expression is surrounded by round brackets (   ).

For example, in the width command:

```
width   E   ($tangent_spiral_2 - 50)   ($tangent_spiral_2 + 50)   3   5   absolute
```

the expression ($tangent_spiral_2 - 50) will yield the chainage of fifty metres before the tangent spiral point of the second HIP point.

The variables can also be used in **#define**'s and **#include**'s according to the following CCCP rules.

(a) **#define**   *NAME*   *Expansion of NAME*

if the #define is longer than one line, the backslash character **** is added to the end of each line that has another line of the #define following to indicate that the #define continues to the next line.

(b) **#define** can be used to define a block of things with more than one argument

For example,

#define SUPER(name,ip,xf1,xf2) \
\
xfall name START_TAPER(ip) xf1 xf2 absolute \
xfall name FIXED_TAPER(ip) xf2 xf2 absolute \
xfall name END_TAPER(ip) xf2 xf1 absolute

means that SUPER("EB",3,-3.0,3.0) will be expanded to the three lines:

xfall "EB" START_TAPER(3) -3.0 3.0 absolute
xfall "EB" FIXED_TAPER(3) 3.0 3.0 absolute
xfall "EB" END_TAPER(3) 3.0 -3.0 absolute

Note that **** is used to indicate that the #define continues to the next line.

(c) (if (*express*) {*expression1* } else {*expression 2*})

is an expression *express,* which evaluates to
if *express* is true, then the value of the expression is *expression1* otherwize *expression 2*.

(d) **word##p**

is equivalent to **wordp**

(e) **#include *file_name***

A number of #defines can be placed in a file and included in the mtf by using the #include.

As an example combining the above rules, the meaning of

LEADING_TAPER_START(2)

given by the following $define:

#define LEADING_TAPER_START(ip) (if($radius_##ip < 75) {$tangent_spiral_##ip - 50)\
else { $tangent_spiral_##ip - 75 })

is

if the radius of the second HIP is less than 75, then set LEADING_TAPER_START(2) to the chainage of the tangent spiral point of the second HIP point, less fifty metres, **otherwize** LEADING_TAPER_START(2) is set to the chainage of the tangent spiral point of the second HIP point, less seventy five metres

A more complicated example now follows which shows how #defines can be used to easily apply super-elevation according to rules involving leading and trailing spirals.

```
// ---------------------------------------------------
#define LEADING_TAPER_START(ip)    ( if($radius_##ip < 75)\
 { $tangent_spiral_##ip - 50 } else { $tangent_spiral_##ip - 75 })
```

// Note: LEADING_TAPER_START evaluates to a chainage.

```
#define LEADING_TAPER_END(ip)    ( if($radius_##ip < 75)\
 { $tangent_spiral_##ip + 10 } else { $tangent_spiral_##ip + 25 })
```

// Note: LEADING_TAPER_END evaluates to a chainage.

```
#define TRAILING_TAPER_START(ip) ( if($radius_##ip < 75)\
 { $spiral_tangent_##ip - 10 } else { $spiral_tangent_##ip - 25 })
```

// Note: TRAILING_TAPER_START evaluates to a chainage.

```
#define TRAILING_TAPER_END(ip)   ( if($radius_##ip < 75)\
 { $spiral_tangent_##ip + 50 } else { $spiral_tangent_##ip + 75 })
```

// Note: TRAILING_TAPER_END evaluates to a chainage.

```
#define START_TAPER(ip) LEADING_TAPER_START(ip)\
          LEADING_TAPER_END(ip)
```

// Note: START_TAPER evaluates to two chainages: chainge1 chainage2.

```
#define FIXED_TAPER(ip) LEADING_TAPER_END(ip)\
          TRAILING_TAPER_START(ip)
```

// Note: FIXED_TAPER evaluates to two chainages: chainge1 chainage2.

```
#define END_TAPER(ip)   TRAILING_TAPER_START(ip)\
          TRAILING_TAPER_END(ip)
```

// Note: TRAILING_TAPER evaluates to two chainages: chainge1 chainage2.

```
#define SUPER(name,ip,x1,x2) \
\
  xfall name START_TAPER(ip) x1 x2 absolute \
  xfall name FIXED_TAPER(ip) x2 x2 absolute \
  xfall name END_TAPER(ip)   x2 x1 absolute
```

With the above #defines, SUPER evaluates to three lines of xfall MTF Modifiers which rotates cross-fall, keeps it constant and then counter-rotates the cross-fall about leading and trailing spirals for an HIP point.

For example,

SUPER(EB,2,-3,3)

would give expand to the following three lines of xfall commands for link **EB** which will rotate xfall from -3 to 3, hold it at 3 and then rotate it back form 3 to -3, all about the leading and trailing spirals for horizontal intersection point number two:

xfall EB begin_chainage_for_start_taper end_chainage_for_start_taper -3 3 absolute

xfall EB begin_chainage_for_fixed_taper end_chainage_for_fixed_taper_ch 3 3 absolute

xfall EB begin_chainage_for_end_taper end_chainage_for_end_taper 3 -3 absolute

The above definitions of SUPER could be placed in a file, say

f:\12d\12dmodel\library\super_mtf.def

and then simply included into a mtf file using the a #include:

#include f:\12d\12dmodel\library\super_mtf.def

Examples of Substitutions in the Modifiers and Templates File

The following example shows how #defines can be used to easily apply super-elevation according to rules involving leading and trailing spirals.

The definition of SUPER is given in the file, f:\12d\12dmodel\library\super_mtf.def, which is simply included in the mtf file using a #include.

```
// ----------------------------------------------------------------
// include the file containing the definition of SUPER
// ----------------------------------------------------------------

#include "f:\12d\12dmodel\library\super_mtf.def"
```

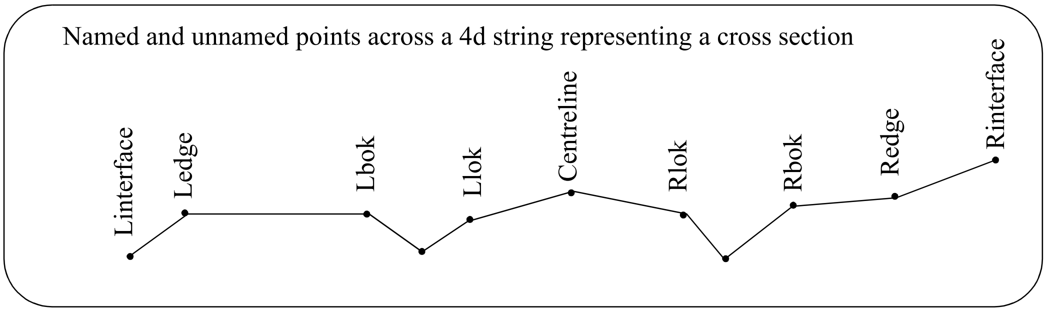
```
// ------------------------------------------------------------------
// apply templates left and right and then modifier them
// -------------------------------------------------------------------

left_side = {
   -999999 std
    999999
}

right_side = {
  -999999 std
   999999
}

specials = {
}

hinge_modifier = {
}

left_side_modifier = {
// linearly widen the link EB by 12 metres and bring it back
// in again.

width EB ($tangent_spiral_2 - 50) ($tangent_spiral_2 + 50) 0  12
width EB ($tangent_spiral_2 + 50) ($spiral_tangent_7 - 50) 12  12
width EB ($spiral_tangent_7 - 50) ($spiral_tangent_7 + 50) 12  0

// define superelevation

 SUPER(EB,2,-3,3)
 SUPER(EB,4,-3,3)
 SUPER(EB,5,-3,3)
 SUPER(EB,6,-3,3)
}

right_side_modifier = {

 SUPER(EB,3,-3,3)
 SUPER(EB,7,-3,3)
}
stripping = {
}
//boxing_file = "yourfile.bf"
left_boxing = {
}
right_boxing = {
}
string_modifiers = {
}
//section_width = 10000.0
```

21.10.2 Text Format of the Boxing File

The interactive editor for creating the *Boxing Definitions* in the *Boxing* file has been given in the previous sections 21.7.2 Edit Boxing File.

This section will describe the text layout of the text boxing file.

Boxing is defined by a series of commands going from **left to right** across the points of a cross section. (Cross sections are generated as 4d strings by options such as **apply** and **cuts**).



The definitions of boxing cross sections are set up in a file ending in **.bf** (called the boxing file) and any number of boxing definitions can be placed in the boxing file as long as each boxing definition is given a unique name.

In the boxing file, a *Boxing Definition* with name **boxing_name** is set out as

```
boxing      boxing_name   {
                                boxing commands
                          }
```

The boxing commands apply from the offset specified at the beginning of that command (**start offset** for the command) to the start offset of the **next** command in the file (like chainages and templates in the Apply MTF file).Hence the **end offset** for a command is the start offset of the following command in the file.

For example, in the sequence

```
offset1,     command1
offset2,     command2
```

command1 goes from offset1 (the start offset for command1) to offset2 (the end offset for command1).

The **offsets** can be **fixed** values or can be specified **relative** to a point name on the 4d string representing the x-section. One restriction is that all the points mentioned in a boxing definition, must be present in each x-section the boxing definition is applied to.

The notation used to specify an actual offset, or offset relative to a point name is

```
        offset_value
or      “point_name”
or      “point_name”   +/-   offset_value
```

and means in the first case

take the specified **offset** value (offset_value can be positive or negative).

and for the other two relative cases

take the **offset at** the string or 4d point called **point_name** and add **offset_value** to it (offset_value can be positive of negative).

Hence the offset can be specified as either a value or given **relative to point_name** (**relative offset**). For convenience, the offset will be denoted by **rel_off**.

The major advantage of defining boxing in terms of point names is that whenever the points are modified using MTF Modifiers, the boxing across the section is also automatically modified.

Since the offset part of a command can consist of text plus or minus a number, commas are used to separate each field of the command.

A boxing definition may be used in either a left_boxing or right_boxing mtf command (see the section, creating boxing), however unlike the case for templates, the boxing definition is applied to the point names going from left to right regardless of whether the boxing definition is used on the left or the right.

In practice, the definitions for left and right boxing are usually different because of the necessity of having unique point names across the entire cross section.

Vertical Walls

If a boxing definition creates two different height values at the one offset (i.e. a wall), then extra offset points are automatically inserted into the boxing x-section (at 0.1 mm from the previous point) so that no points are on top of each other.

So in effect a vertical wall is created but since no points are created that are directly above each other, the data can be triangulated.

The commands in the boxing definition are:

see 21.10.2.1 Text Format - Boxing Copy
see 21.10.2.2 Text Format - Boxing Copy with Height Adjustment
see 21.10.2.4 Text Format - Boxing Line
see 21.10.2.5 Text Format - Boxing Xfall
see 21.10.2.6 Text Format - Boxing Xfall1
see 21.10.2.9 Text Format - Boxing End
see 21.10.2.10 Text Format - Boxing Right_Xfall
see 21.10.2.11 Text Format - Boxing Right_slope
see 21.10.2.12 Text Format - Boxing Right_copy
see 21.10.2.13 Text Format - Boxing Left_Xfall
see 21.10.2.14 Text Format - Boxing Left_slope
see 21.10.2.15 Text Format - Boxing Left_copy
see 21.10.2.16 Text Format - Boxing Definitions Example

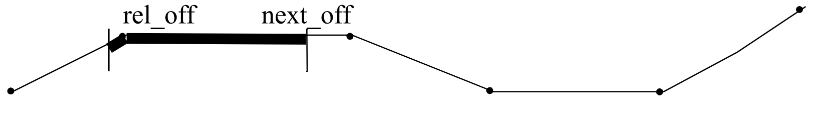
21.10.2.1 Text Format - Boxing Copy

rel_off, **copy**

This copies the links (or parts of links) from the start offset **rel_off** to the offset on the next line, keeping the present crossfall and height of each link

.

rel_off, **copy**
next_off, ...



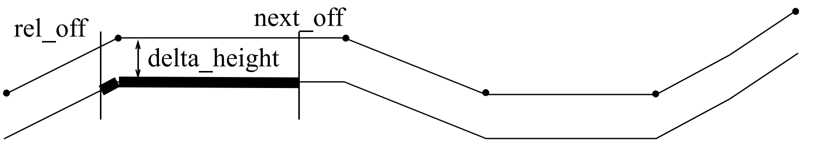
For the boxing panel that creates this boxing command, see 21.7.2.2.1 Boxing Copy.

21.10.2.2 Text Format - Boxing Copy with Height Adjustment

rel_off, **copy**, delta_height

This copies the links (or parts of links) from the start offset **rel_off** to the offset on the next line, keeping the present slope of each link but adjusting the height of each link by **delta_height**.

rel_off, **copy**, delta_height
next_off, ...



For the boxing panel that creates this boxing command, see 21.7.2.2.1 Boxing Copy.

21.10.2.3 Text Format - Boxing Vertex

For the boxing panel that creates this boxing command, see 21.7.2.2.2 Boxing Vertex.

21.10.2.4 Text Format - Boxing Line

Line Through Two Given Points

rel_off, **line**, rel_off_1, delta_height_1, rel_off_2, delta_height_2

This creates a link from the start offset **rel_off** to the offset on the next line, with a

(a) slope given by two points

(b) start height given by projecting to the start offset, the line through the two points.

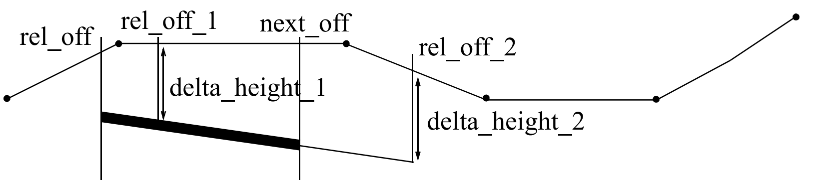
The first point defining the line has an offset **rel_off_1** and a height given by the height of the design x-section at offset **rel_off_1** adjusted by **delta_height_1**.

The second point defining the line has an offset **rel_off_2** and a height given by the height of the design x-section at offset **rel_off_2** adjusted by **delta_height_2**.

The typed format of the **line** commands created by this panel is (all on one line):

point_name + point_offset,**line**, point_1_name + point_1_offset,delta_height_1,
point_2_name + point_2_offset,delta_height_2

rel_off, **line**, rel_off_1, delta_height_1, rel_off_2, delta_height_2
next_off, ...



For the boxing panel that creates this boxing command, see 21.7.2.2.6 Boxing Line - line through two points.

21.10.2.5 Text Format - Boxing Xfall

Line Through a Given Point with a Given Crossfall

rel_off, **xfall**, rel_off_1, delta_height_1, xfall_value

This creates a link from the start offset to the offset on the next line, with a

(a) user given xfall

(b) start height given by projecting to the start offset, the line with a given xfall and going through a given point.

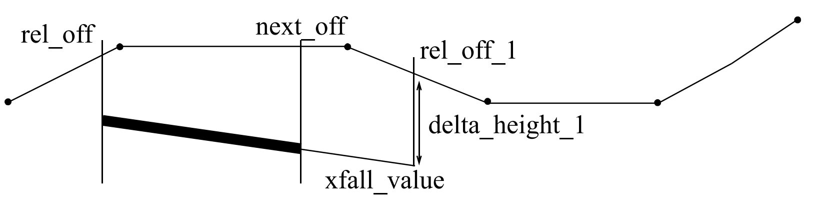
The point defining the line has an offset **rel_off_1** and a height given by the height of the design x-section at offset **rel_off_1** adjusted by **delta_height_1**.

The typed formats of the **xfall** commands created by this panel is:

point_name + point_offset, **xfall,** control_name + control_offset, control_height, xfall

The crossfall of the line is given by **xfall_value**

rel_off, **xfall**, rel_off_1, delta_height_1, xfall_value
next_off, ...



For the boxing panel that creates this boxing command, see 21.7.2.2.4 Boxing Xfall - line through a point with a given crossfall.

21.10.2.6 Text Format - Boxing Xfall1

Line Through a Given Point with Crossfall Taken from a Point

rel_off, **xfall1**, rel_off_1, delta_height_1, rel_off_2, delta_xfall

This creates a link from the start offset **rel_off** to the offset on the next line, with a

(a) xfall which is the xfall at a given offset plus an addition given xfall

and

(b) start height given by projecting to the start offset, the line with the given xfall and going through a given point.

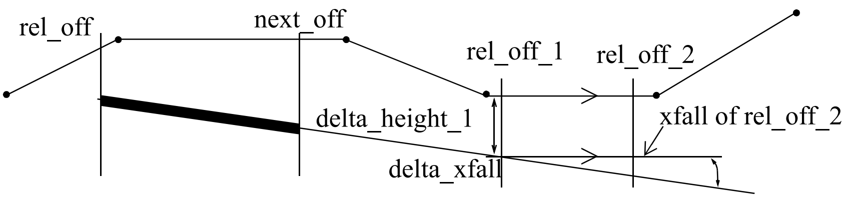
The point defining the line has offset **rel_off_1** and a height given by the height of the design x-section at offset **rel_off_1** adjusted by **delta_height_1**

The typed formats of the **xfall1** commands created by this panel is:

point_name + point_offset, **xfall1,** control_name + control_offset, control_height,
xfall_name + xfall_offset, xfall_delta

The xfall of the line is **delta_xfall** added to the xfall on the design x-section at the offset **rel_off_2**

rel_off, **xfall1**, rel_off_1, delta_height_1, rel_off_2, delta_xfall
next_off, ...



For the boxing panel that creates this boxing command, see 21.7.2.2.5 Boxing Xfall 1 - line through a point with a crossfall taken from a point.

21.10.2.7 Text Format - Boxing Intersect

For the boxing panel that creates this boxing command, see 21.7.2.2.7 Boxing Intersect - Intersection of Two Lines.

21.10.2.8 Text Format - Boxing Xfall Point

For the boxing panel that creates this boxing command, see 21.7.2.2.9 Boxing Xfall Point.

21.10.2.9 Text Format - Boxing End

rel_off, **end**

The typed format of the **line** commands created by this panel is:

point_name + point_offset,   **end**

This command in used to give an end offset for any of the above commands when no command follows to use for the end offset. That is, the offset **rel_off** is used as the **end offset** for the **previous command**.

end doesn't begin another command and none of the above commands can follow an **end** command.

All of the above boxing commands are known as **standard** commands. Hence the **end** command is the **last** standard command.

For the boxing panel that creates this boxing command, see 21.7.2.2.16 Boxing End.

21.10.2.10 Text Format - Boxing Right_Xfall

Batter off Right End of Boxing with Crossfall Taken from a Point

The **right_xfall** command has two versions - providing a fixed cross fall or using the crossfall from part of the design x-section.

right_xfall   rel_off,   delta_xfall

right_xfall   xfall_value

The typed formats of the **right_xfall** commands created by this panel is:

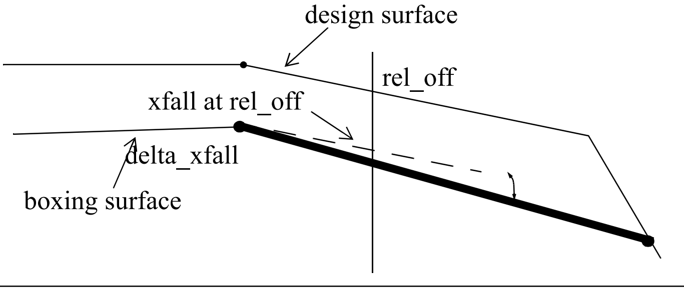
right_xfall,   xfall_name + xfall_offset, xfall_delta
For the first case, the **right_xfall** command batters off the right end of the boxing using the xfall from the design section at a given offset *rel_off* plus an additional given xfall *delta_xfall*, until the design section is intersected. The intersection point is used as the next point in the boxing.

Hence the xfall of the batter is **delta_xfall** added to the xfall on the design x-section at the offset **rel_off**

For the second case, the **right_xfall** command batters off to the right end of the boxing at the given xfall **xfall_value** until the design section is intersected. The intersection point is used as the next point in the boxing.

In the right_xfall command, the xfall is percent cross-fall and a positive xfall is up and negative down.

right_xfall1   rel_off,   del
ta_xfall



For the boxing panel that creates this boxing command, see 21.7.2.2.18 Boxing Right Side Interface Xfall from Point.

21.10.2.11 Text Format - Boxing Right_slope

Batter Off the Right End of Boxing with a Given Slope

right_slope slope_value

The **right_slope** commands batter off to the right end of the boxing at the given slope **slope_value** until the design section is intersected. The intersection point is used as the next point in the boxing. In this command, a positive slope is up and negative down.

For the boxing panel that creates this boxing command, see 21.7.2.2.18 Boxing Right Side Interface Xfall from Point.

21.10.2.12 Text Format - Boxing Right_copy

Copy from the Last Boxing Point to the end of Design Section

right_copy value

If value is non-zero, the right_copy command copies from the last boxing point to the last point on the design section. Hence the right_copy command is used to continue the boxing section from the intersection point with the design section to the end of the design section (a full width boxing section).

Note
The **right_xfall** command **takes precedence over** the **right_slope** command. That is, if both commands exist then only the right_xfall command is used.

For the boxing panel that creates this boxing command, see 21.7.2.2.18 Boxing Right Side Interface Xfall from Point.

21.10.2.13 Text Format - Boxing Left_Xfall

Batter off Left End of Boxing with Crossfall Taken from a Point

left_xfall rel_off, delta_xfall

left_xfall xfall_value

The typed formats of the **left_xfall** commands created by this panel is:

left_xfall, xfall_name + xfall_offset, xfall_delta

The left_xfall is identical to the right_xfall command except that it is applied at the beginning of the boxing definition and batters off to the left rather than to the right.

In the left_xfall command, the xfall is percent cross-fall and a positive xfall is up and negative down.

For the boxing panel that creates this boxing command, see 21.7.2.2.17 Boxing Left Side Interface Xfall from Point.

21.10.2.14 Text Format - Boxing Left_slope

Batter Off the Left End of Boxing with a Given Slope

left_slope slope_value

The left_slope is identical to the right_slope command except that it is applied at the beginning of the boxing definition and batters off to the left rather than to the right. In this command, a positive slope is up and negative down.

The left_slope and left_copy are identical to the right_slope and right_copy except that they are applied at the beginning of the boxing definition and batter off and copy to the left rather than to the right. In this command, a positive slope is up and negative down.

The left_copy command must come before the left_slope command but they both come after all of the standard commands.

Note

The **left_xfall** command **takes precedence over** the **left_slope** command. That is, if both commands exist then only the left_xfall command is used.

For the boxing panel that creates this boxing command, see 21.7.2.2.17 Boxing Left Side Interface Xfall from Point.

21.10.2.15 Text Format - Boxing Left_copy

Copy from the Start of Design Section to the First Boxing Point

left_copy value

The left_copy is identical to the right_copy except that it is applied at the beginning of the boxing definition and copies to the left rather than to the right.

If *value* is non-zero, the **left_copy** command copies from the last boxing point to the last point on the design section. Hence the left_copy command is used to continue the boxing section from the first boxing point to the start of the design section (a full width boxing section).

Note

The left_xfall, left_slope, left_copy, right_xfall, right_slope and right_copy commands should come after all of the standard commands.

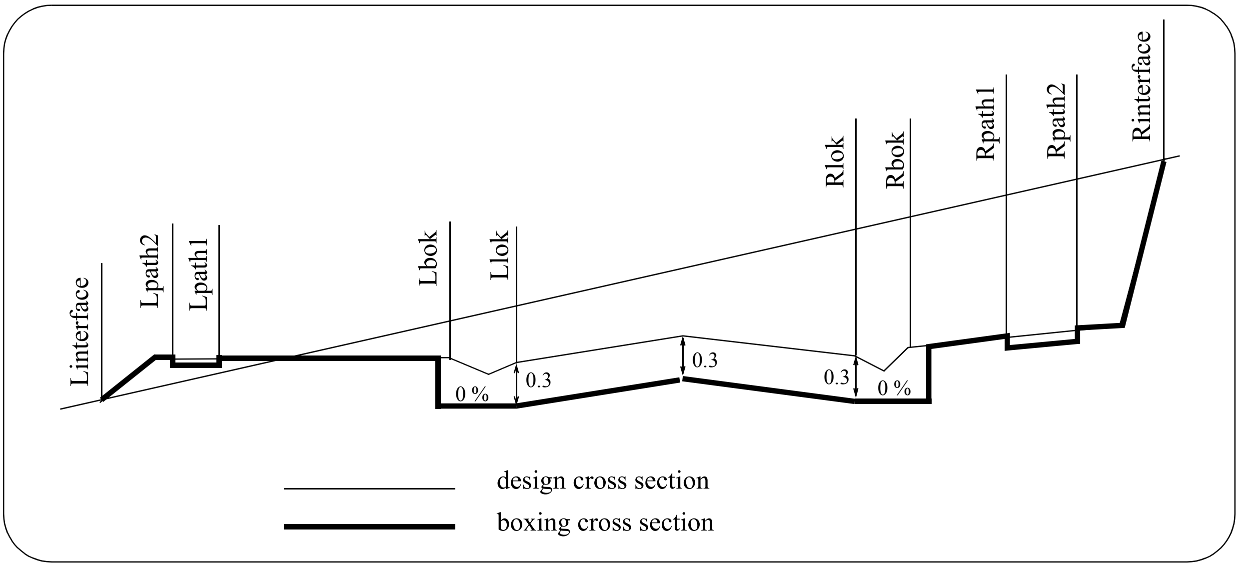
When a boxing definition is applied on the left side, any right_xfall, right_slope or right_copy is ignored. Similarly, when a boxing definition is applied on the right side, any left_xfall, left_slope or left_copy is ignored.

Hence a boxing definition can have a left_xfall, left_slope, left_copy and a right_xfall, right_slope, right_copy but which set is used depends on whether the boxing definition is used on the left or the right.

When the **boxing** panel is used to create the boxing, only one boxing definition is needed and all of the right_xfall, right_slope, right_copy, left_xfall, left_slope and left_copy commands are used.

For the boxing panel that creates this boxing command, see 21.7.2.2.17 Boxing Left Side Interface Xfall from Point.

21.10.2.16 Text Format - Boxing Definitions Example




```
boxing "1" {

    "Linterface",        copy                                       // line 1
    "Lpath2",            copy,     -0.075                           // line 2
    "Lpath1",            copy                                       // line 3
    "Lbok" - 0.15,       xfall,    "Llok",     -0.3,      0         // line 4
    "Llok",              copy,     -0.3                             // line 5
    "Rlok",              xfall,    "Rlok",     -0.3,      0         // line 6
    "Rbok" + 0.15,       copy                                       // line 7
    "Rpath1",            copy      -0.095                           // line 8
    "Rpath2",            copy                                       // line 9
    "Rinterface",        end                                        // line 10
}
```

Description

Line 1	copy from "Linterface" to "Lpath2"
Line 2	copy and drop by 0.075 from "Lpath 2" to "Lpath 1"
Line 3	copy from "Lpath1" to 0.15 before "Lbok"
Line 4	go from 0.15 before "Lbok" to "Llok" at depth 0.3 below "Llok", with cross fall 0
Line 5	copy and drop by 0.3 from "Llok" to "Rlok"
Line 6	go from "Rlok" to 0.15 past "Rbok" at depth 0.3 below "Rlok", with cross fall 0
Line 7	copy from 0.15 past "Rbok" to "Rpath1"
Line 8	copy and drop by 0.095 from "Rpath1" to "Rpath2"
Line 9	copy from "Rpath2" to "Rinterface"

```
boxing "left narrow"  {                  // boxing defined from just before lkerb to m001
            "lkerb"   - 0.7  ,        copy
            "m001"   - 1.0,           copy   ,     -0.6
            "m001",                   end
  }
```

```
boxing "left" {      // boxing defined from the left start of the section to m001 (the centreline say)
        "lkerb",     copy,     -0.6
        "m001",      end
        left_slope   5.0
        left_copy    1
  }
boxing "right" {          // boxing defined from m001 to the far right of the section
        "m001",      copy,     0.6
        "rkerb",     end
        right_slope  5.0
        right_copy 1
  }
boxing "full" {      // a full width boxing - defines the entire boxing going from left to right
        "lkerb",     copy,     -0.6
        "rkerb",     end
        left_slope   5.0
        left_copy    1
        right_slope -5.0
        right_copy 1
  }
```

For more information on how Boxing works in ***12d Model***, please go to the section 21.4 Smart Chainages.

21.11 Superseded MTF Commands

These MTF Modifier options were superseded by the commands in ***12d Model 11*** that did not care how the links were originally defined.

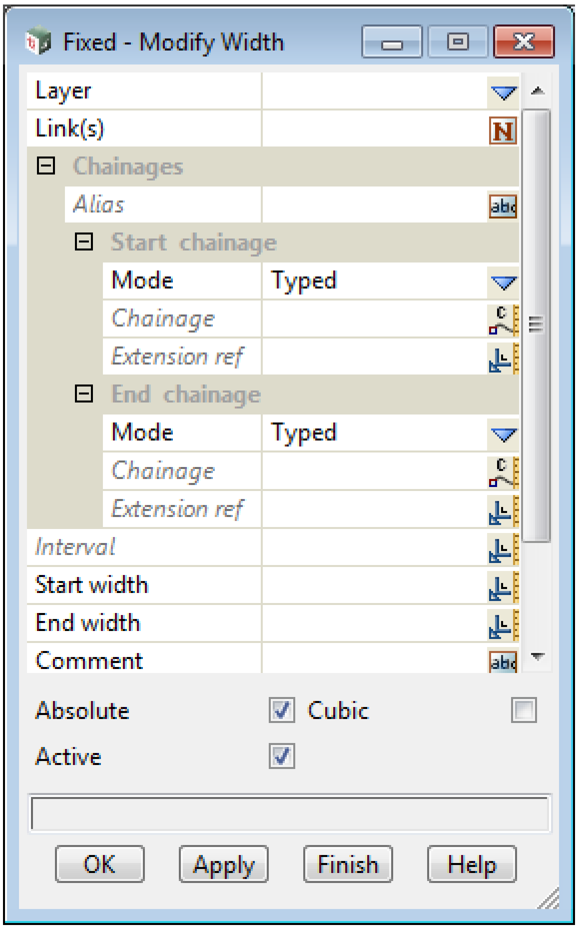
21.11.0.1 Fixed Link - Modify Width or Height - Superseded

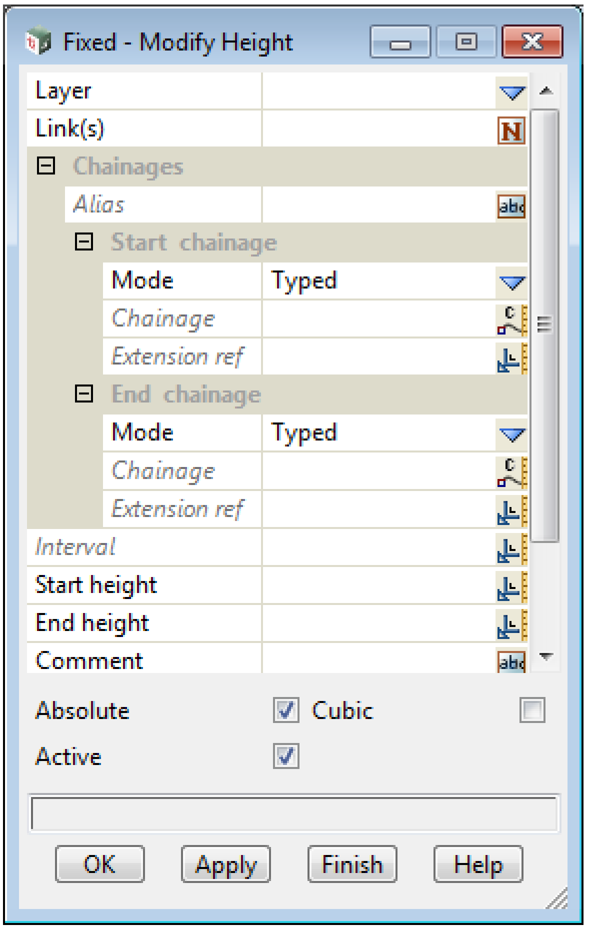
This option has been superseded in V11 by the Fixed Link modifier, **Modify Link**. See 21.2.2.2.4 Fixed - Modify Link.

The **Width** modifier is used to ***modify the width*** of fixed links originally defined by width (*width and height* or *width and xfall*).

The **Height** modifier is used to ***modify the height*** of fixed links originally defined by height (*height and width* or *height and xfall*).

Selecting **width and height** brings up the **Fixed - Modify Width** and **Fixed - Modify Height** panels respectively.





The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu

name of the link to modify.

Start/End mode	choice box	Start (ref)/End (ref)	

defines the start/end chainages for modifying the link.

Interval input

*if **non blank**, the interval to use to create cross sections and strings over the given chainage range. If **blank**, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

Start/End width/height input measures menu

start/end width/height for modifying the template link.

Extra start/end tick box

*if **ticked**, add an extra x-section 0.1 mm **before** the start/end chainage.*

Absolute tick box tick

*if **ticked**, the width/height is set to the values given in the* **start** *and* **end value** *fields.*
*if **not ticked**, the values given in the* **start** *and* **end value** *fields are added to the existing widths/heights.*

Cubic tick box

*if **ticked**, the width/height is varied as a reverse cubic between the start and end chainages.*
*if **not ticked**, the width/height is varied linearly between the start and end chainages.*

Comment input

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active tick box tick

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

OK button

OK *stores the values in the fields and removes the panel BUT no **recalc** is done.*

Apply button

Apply *stores the values and leaves the panel on the screen.*

*If the **MTF** is being used in an **Apply MTF** and **Auto recalc** is ticked in the MTF, then whenever the **Apply** button is clicked, a **recalc** of the associated **Apply MTF** for the MTF is done.*

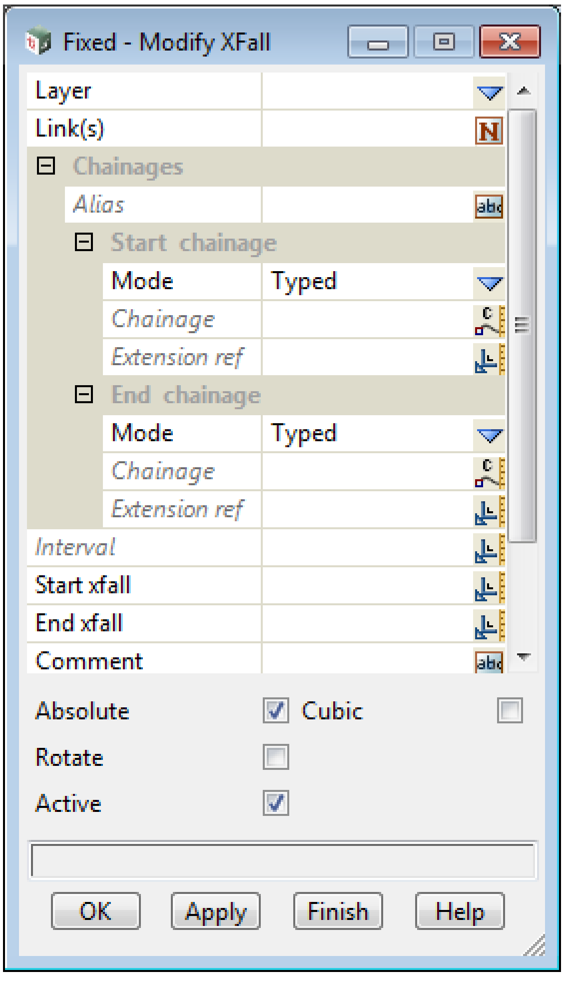
Continue to the next section 21.11.0.2 Fixed Link - Modify Xfall - Superseded or go to the replacement options 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

21.11.0.2 Fixed Link - Modify Xfall - Superseded

This option has been superseded in V11 by the Fixed Link modifier, **Modify Link**. See 21.2.2.2.4 Fixed - Modify Link.

The **Xfall** modifier is used to ***modify the cross fall*** of fixed links originally defined by cross fall (*xfall and width* or *xfall and height*).

Selecting **Xfall** brings up the **Fixed - Modify Xfall** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu

name of the link to modify.

Start/End mode	choice box	Start (ref)/End (ref)	

defines the start/end chainages for modifying the link.

Interval	input		

if ***non blank****, the interval to use to create cross sections and strings over the given chainage range. If* ***blank****, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

Start/End xfall	input		measures menu

start/end crossfall for modifying the template link.

Extra start/end	tick box		

*if **ticked**, add an extra x-section 0.1 mm **before** the start/end chainage.*

Absolute tick box tick

*if **ticked**, the xfall is set to the values given in the* **start** *and* **end xfall** *fields.*
*if **not ticked**, the values given in the* **start** *and* **end xfall** *fields are added to the existing xfall.*

Cubic and Rotate tick boxes:

*Only none or one of **Cubic** and **Rotate** can be set to **tick.***
*The **default** is none - that is, **neither is ticked** and in the default case, the crossfall is varied **linearly with respect to crossfall** between the start and end chainages.*

Cubic tick box

*if **ticked**, the crossfall is varied as a reverse cubic between the start and end chainages.*

Rotate tick box

*if **ticked**, the xfall is varied linearly **with respect to** the **angle,** between the start and end chainages.*

Comment input

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active tick box tick

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

OK button

OK *stores the values in the fields and removes the panel BUT no **recalc** is done.*

Apply button

Apply *stores the values and leaves the panel on the screen.*

*If the **MTF** is being used in an **Apply MTF** and **Auto recalc** is ticked in the MTF, then whenever the **Apply** button is clicked, a **recalc** of the associated **Apply MTF** for the MTF is done.*

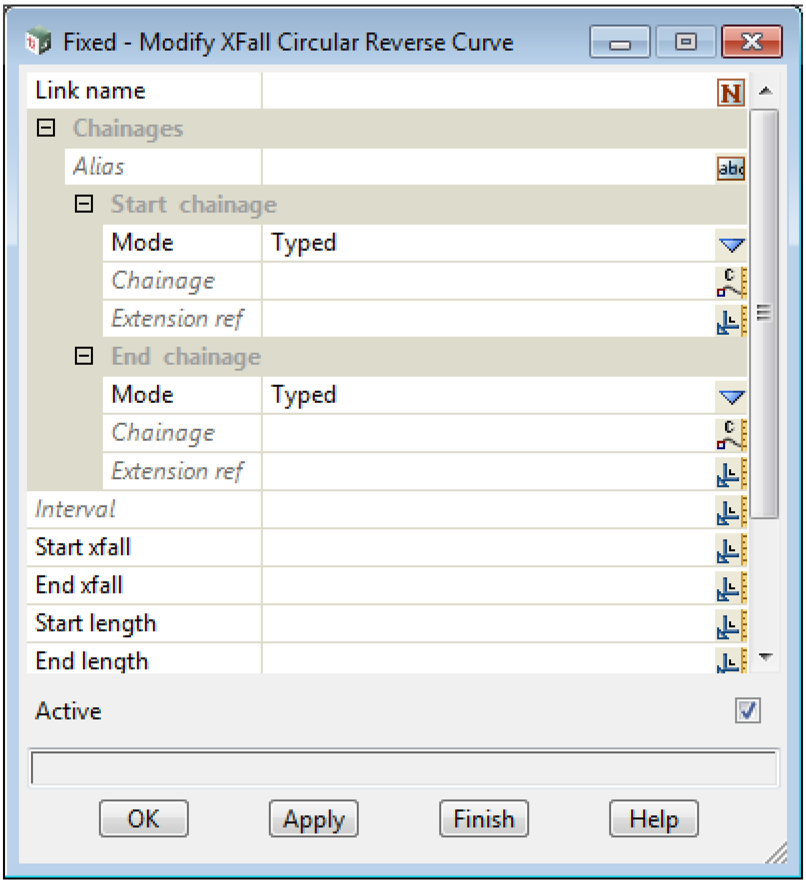
Continue to the next section 21.11.0.3 Fixed Link - Modify Xfall by CRC Formula - Superseded or go to the replacement options 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

21.11.0.3 Fixed Link - Modify Xfall by CRC Formula - Superseded

This option has been superseded in V11 by the Fixed Link modifier, **Modify Link**. See 21.2.2.2.4 Fixed - Modify Link.

The **Xfall CRC** modifier is used to modify the cross fall of fixed links originally defined by **xfall** and uses the **circular reverse curve** formula to modify the cross fall.

Selecting **Xfall CRC** brings up the **Fixed - Modify Xfall Circular Reverse Curve** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu

name of the link to modify.

Start/End mode	choice box	Start (ref)/End (ref)	

defines the start/end chainages for modifying the link.

Interval	input		

*if **non blank**, the interval to use to create cross sections and strings over the given chainage range. If **blank**, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

Start/End xfall	input		measures menu

start/end crossfall for modifying the template link.

Start/End length	input		measures menu

start/end length for circular arcs.

Extra start/end	tick box		

*if **ticked**, add an extra x-section 0.1 mm **before** the start/end chainage.*

Comment input

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active tick box tick

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

OK button

OK *stores the values in the fields and removes the panel BUT no **recalc** is done.*

Apply button

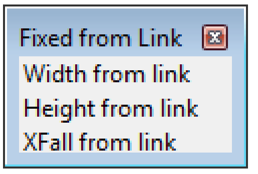
Apply *stores the values and leaves the panel on the screen.*

*If the **MTF** is being used in an **Apply MTF** and **Auto recalc** is ticked in the MTF, then whenever the **Apply** button is clicked, a **recalc** of the associated **Apply MTF** for the MTF is done.*

Continue to the next section 21.11.0.4 Fixed Link - From Link - Superseded or go to the replacement options 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

21.11.0.4 Fixed Link - From Link - Superseded

The **fixed from Link** walk-right brings up the **fixed from Link** menu with options to take the width, xfall or height from another link.



For *Width/Height/Xfall from link*, go to the next section [Fixed Link - Take Width, Height or Xfall from another Link](#)

Fixed Link - Take Width, Height or Xfall from another Link

Width from Link

The **Width from link** modifier is used to ***modify the width*** of fixed links originally defined by width (*width and height* or *width and xfall*) to be the same width as another link. That is, the width of the link is a **copy** of the **width** of another link.

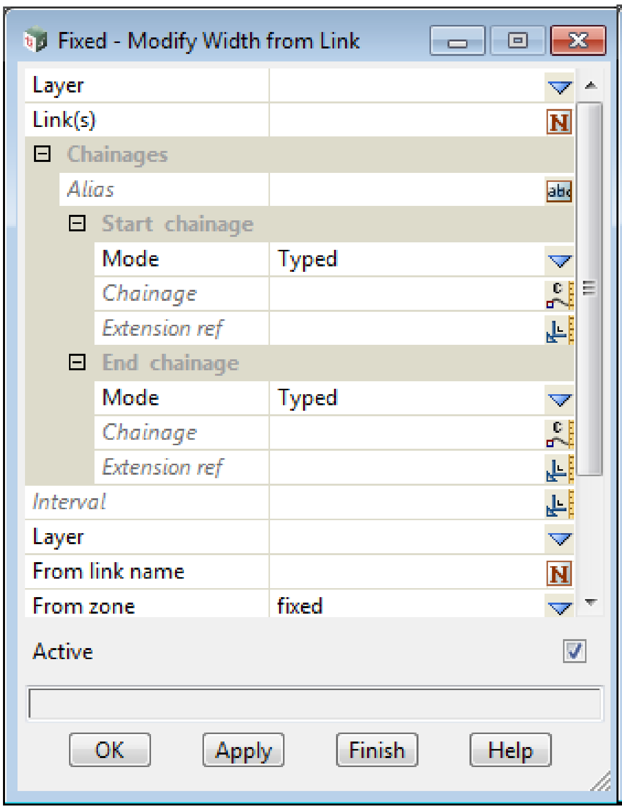
Height from Link

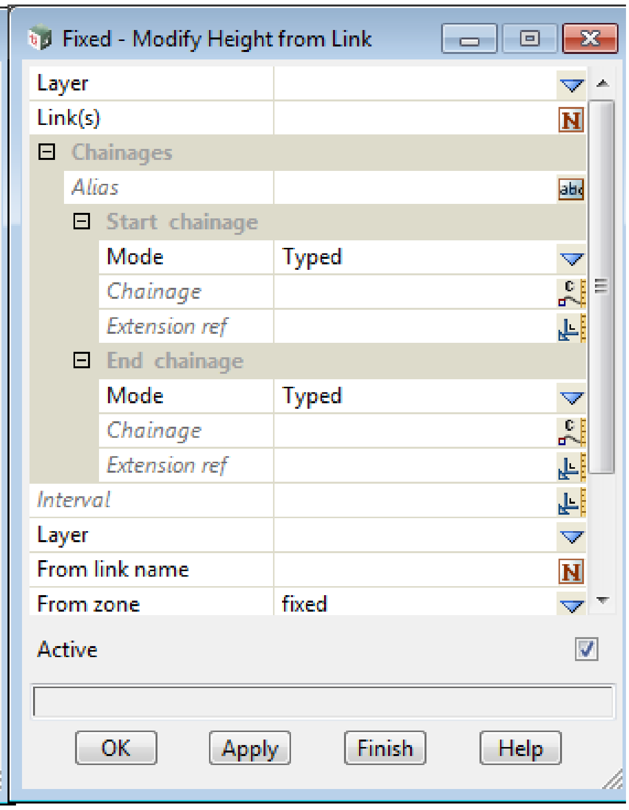
The **Height from link** modifier is used to ***modify the height*** of fixed links originally defined by height (*height and xfall* or *height and width*) to be the same height as another link. That is, the height of the link is a **copy** of the **height** of another link.

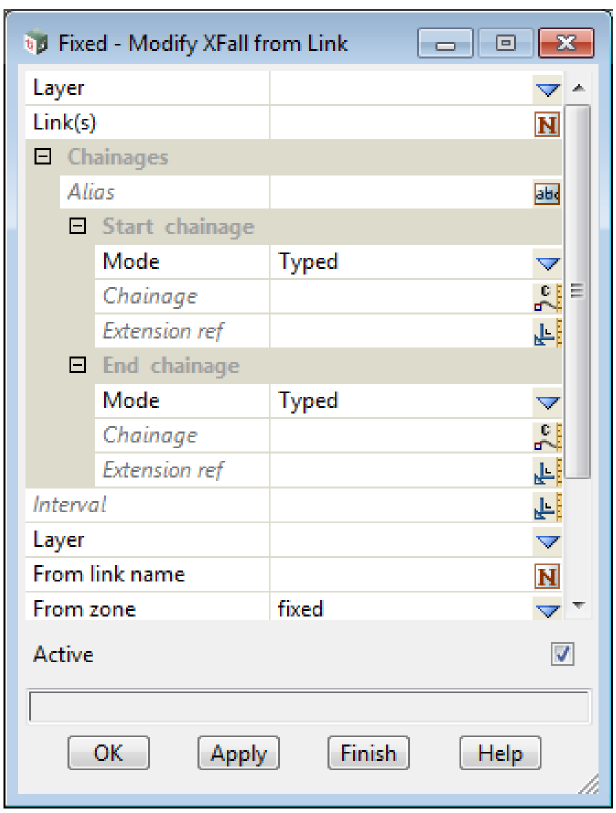
Xfall from Link

The **Xfall from link** modifier is used to ***modify the cross fall*** of fixed links originally defined by xfall *(xfall and height* or *xfall and width*) to be the same xfall as another link. That is, the cross fall of the link is a **copy** of the **cross fall** of another link. The link to copy cross fall from can be defined in terms of cross fall or slope. If the link to copy is defined by slope, then the cross fall is calculated to match the slope.

Selecting the **Width from link, Height from link or Xfall from link** option brings up the **Fixed - Modify Width from Link**, **Fixed- Modify Height from Link** and **Fixed- Modify Xfall from Link** panels respectively.







The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Link name	input		select name menu

name of the link to modify.

Start/End mode	choice box	Start (ref)/End (ref)	

defines the start/end chainages for modifying the link.

Interval input

*if **non blank**, the interval to use to create cross sections and strings over the given chainage range. If **blank**, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

From link name	input		select name menu

template link to take width/height/xfall from.

From zone	input	fixed	fixed, cut, fill

zone that the template link to take width/height/xfall from, comes from.

Extra start/end tick box

*if **ticked**, add an extra x-section 0.1 mm **before** the start/end chainage.*

Comment input

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active	tick box	tick

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

OK button

OK *stores the values in the fields and removes the panel BUT no **recalc** is done.*

Apply button

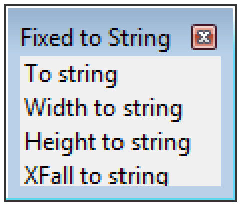
Apply *stores the values and leaves the panel on the screen.*

*If the **MTF** is being used in an **Apply MTF** and **Auto recalc** is ticked in the MTF, then whenever the **Apply** button is clicked, a **recalc** of the associated **Apply MTF** for the MTF is done.*

Continue to the next section 21.11.0.5 Fixed Link - to String - Superseded or go to the replacement options 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

21.11.0.5 Fixed Link - to String - Superseded

The **fixed from string** walk-right brings up the **fixed from string** menu with options to take the width, xfall or height by going to another string.

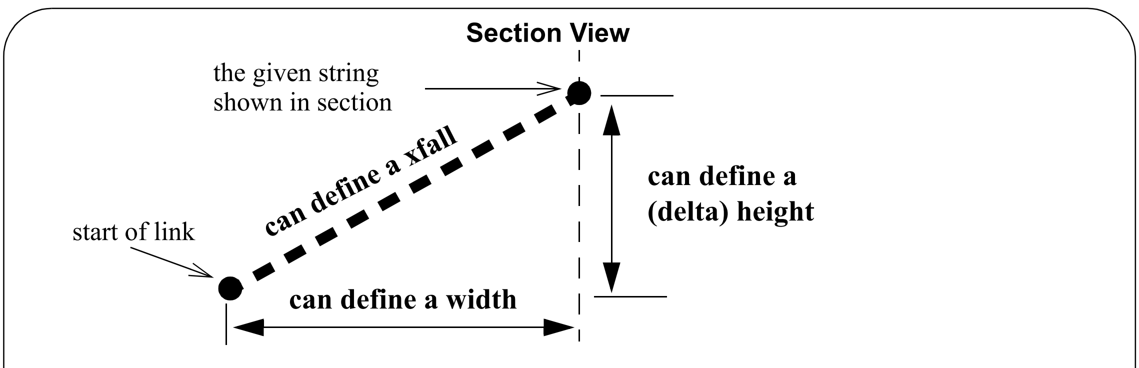


For *To string, go to* — Fixed Link - Modify To String

Width/Height/Xfall to string — Fixed Link - Calculate Width, Height or Xfall to a String

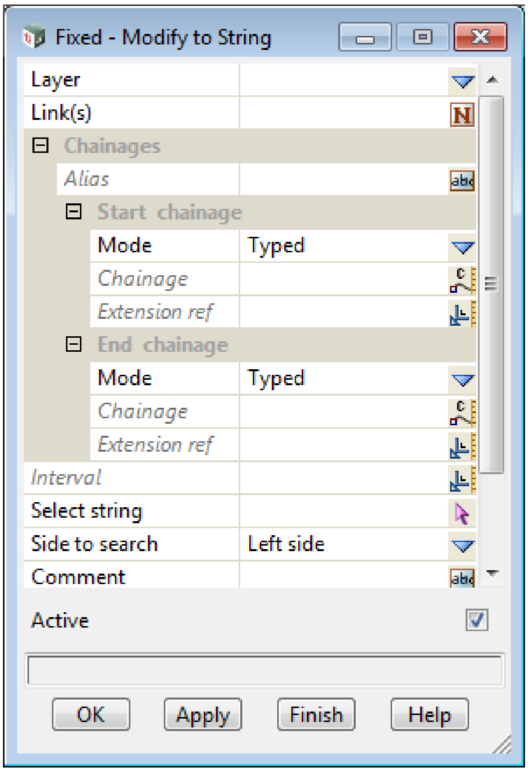
Fixed Link - Modify To String

For any fixed link, **To string** calculates the required width, height and/or xfall of the link needed to get from the start point of the link **to the selected string**.



the start and end points can be used to define a width, xfall or (delta) height

Selecting **To string** bring up the **Fixed - Modify to String** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu

names of the links to modify.

If there is more then one link name then the names must be separated by spaces. If the link name includes a space, then the name must be enclosed in the quotes " (eg "EB 1").

Start/End mode	choice box	Start (ref)/End (ref)	

defines the start/end chainages for modifying the link.

*When the **Start mode** is **Start (ref),** or **Typed** and the chainage is **blank**, the modification begins at the low dropped chainage of the selected string.*

*When the **End mode** is **End (ref),** or **Typed** and the chainage is **blank**, the modification ends at the high dropped chainage of the selected string.*

Interval	input		

*if **non blank**, the interval to use to create cross sections and strings over the given chainage range. If **blank**, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

String	string-select		

select string to use for defining width/height/crossfall for the link.

Side to search	input	left side	left side, right side, both sides

side of the hinge string to start searching to find the string to define width/height/crossfall.

Extra start/end	tick box		

*if **ticked**, add an extra x-section 0.1 mm **before** the start/end chainage.*

Comment	input		

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active tick box tick

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

OK button

OK *stores the values in the fields and removes the panel BUT no **recalc** is done.*

Apply button

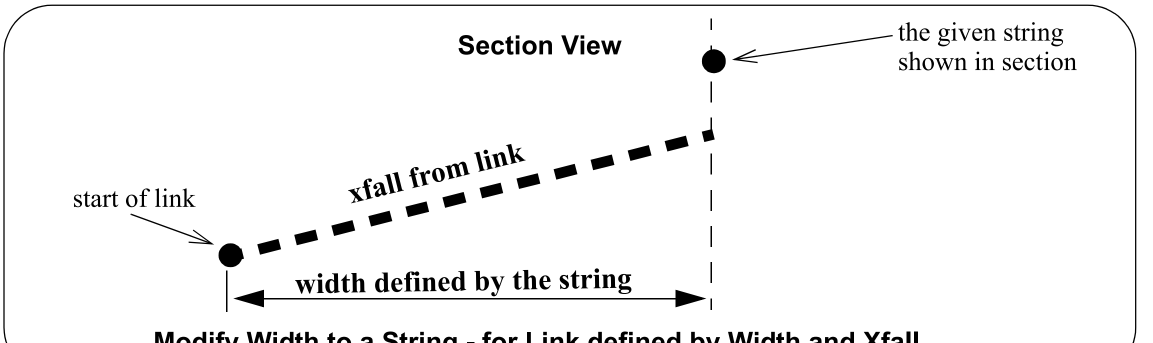
Apply *stores the values and leaves the panel on the screen.*

*If the **MTF** is being used in an **Apply MTF** and **Auto recalc** is ticked in the MTF, then whenever the **Apply** button is clicked, a **recalc** of the associated **Apply MTF** for the MTF is done.*

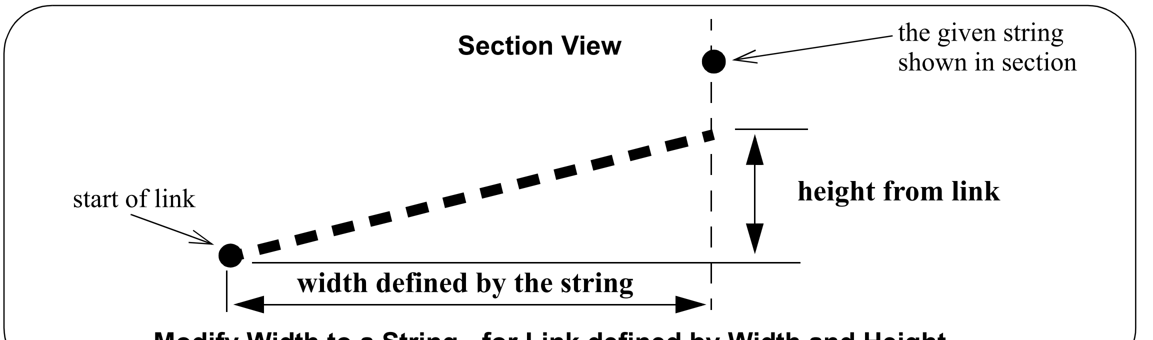
Fixed Link - Calculate Width, Height or Xfall to a String

Width to String:

For a fixed link defined by *width* and *height* or *width* and *xfall*, **Width to string** calculates the **width** of the link as the width from the start point of the link, to the **to the selected string**. The *xfall* or *height* is taken from the link. The option will give an error for a link defined by *height* and *xfall*.



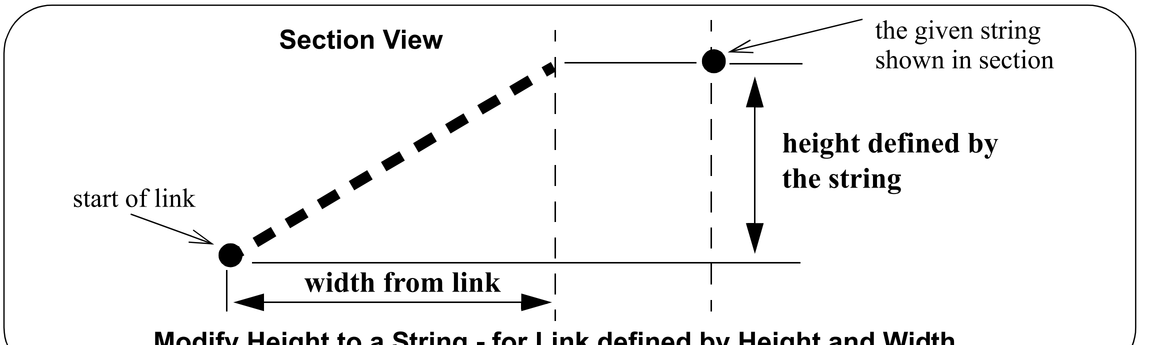
Modify Width to a String - for Link defined by Width and Xfall



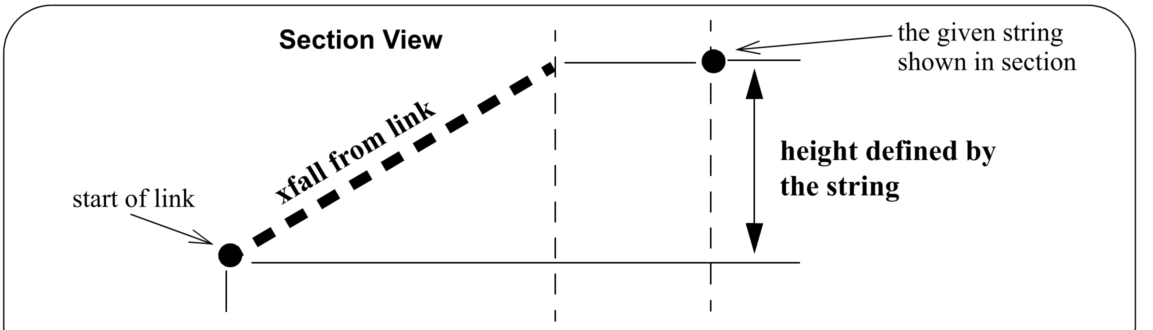
Modify Width to a String - for Link defined by Width and Height

Height to String:

For a fixed link defined by *height* and *width* or *height* and *xfall*, **Height to string** calculates the **height** of the link as the difference in the height at the start point of the link, and the height **at the selected string**. The *xfall* or *width* is taken from the link. The option will give an error for a link defined by *width* and *xfall*.



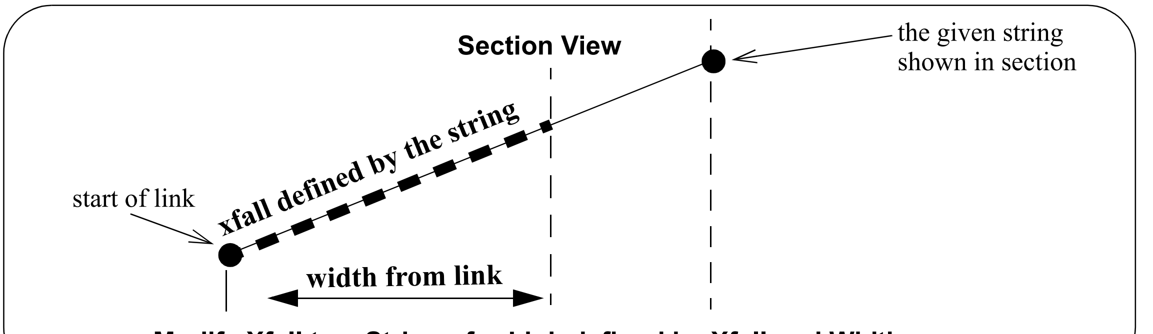
Modify Height to a String - for Link defined by Height and Width



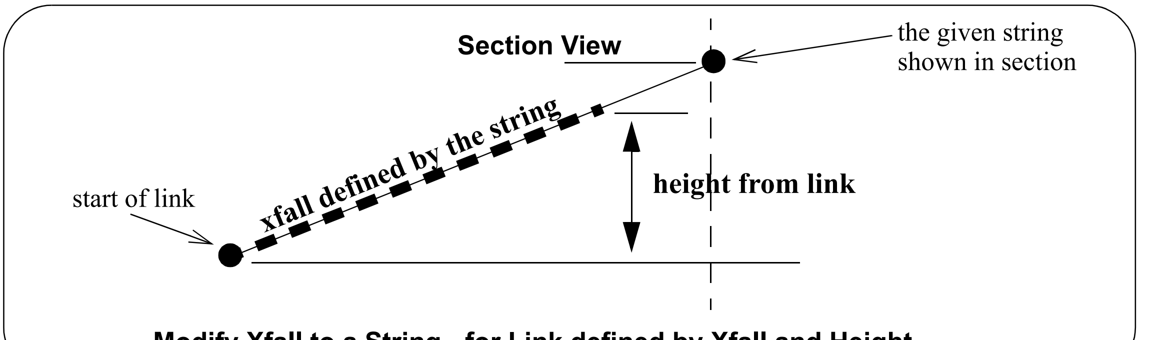
Modify Height to a String - for Link defined by Height and Xfall

Xfall to String:

For a fixed link defined by *xfall and width* or *xfall* and *height*, **Xfall to string** calculates the **xfall** of the link as the *xfall* from the start point of the link to the **selected string**. The *width* or *height* is taken from the link. The option will give an error for a link defined by *width* and *height*.



Modify Xfall to a String - for Link defined by Xfall and Width



Modify Xfall to a String - for Link defined by Xfall and Height

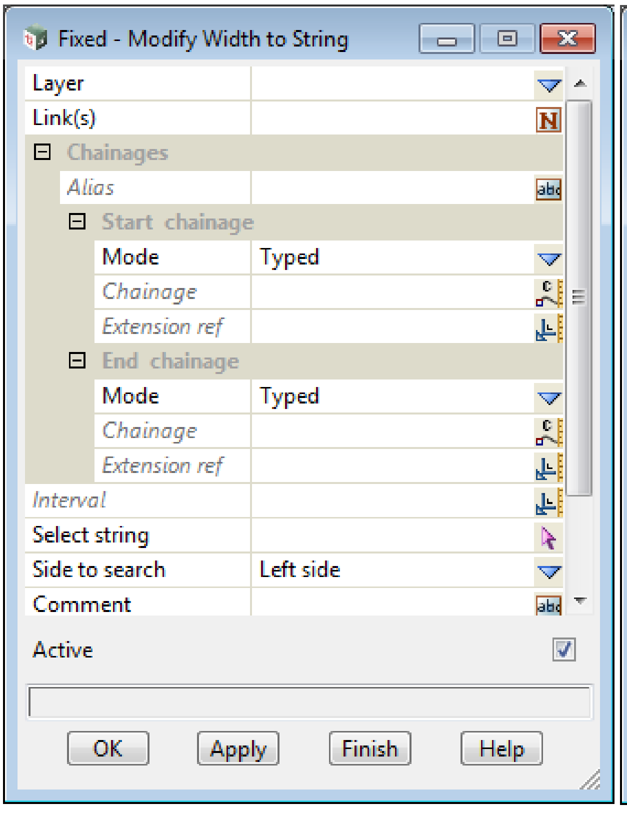
Note:

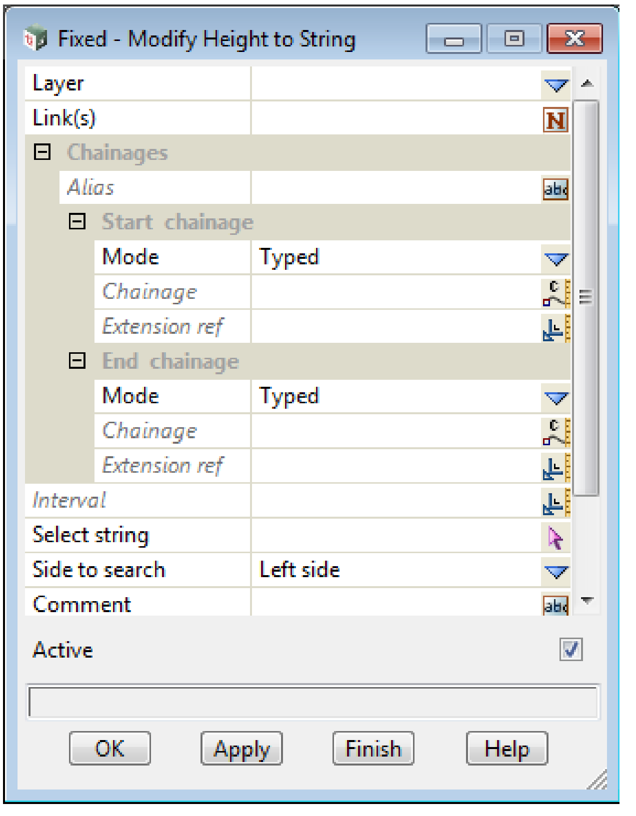
Using two of the above modifiers together and with the same string will place the end point of the link on the selected string. For example for a modifier defined by *width and xfall*, using *width to string* and a *xfall to string* with the same string will place the end of the link on that string.

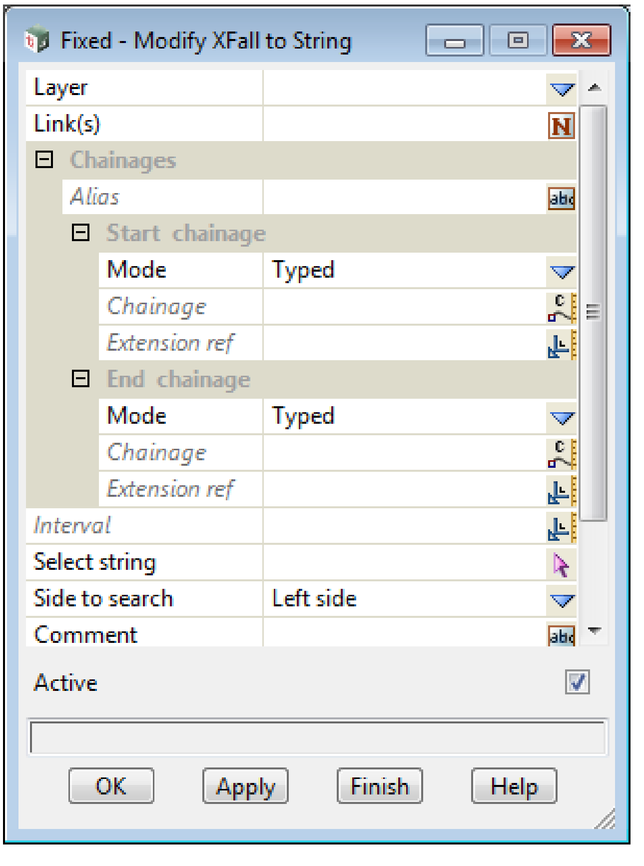
But the To string option will do the same thing in one command. See Fixed Link - Modify To String.

Selecting the **width to string, height to string or xfall to string** option brings up the **Fixed - Modify Width**

to String, **Fixed - Modify Height to String** and **Fixed - Modify Xfall to String** panels respectively.







The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu

name of the link to modify.

Start/End mode	choice box	Start (ref)/End (ref)	

defines the start/end chainages for modifying the link.

*When the **Start mode** is **Start (ref),** or **Typed** and the chainage is **blank**, the modification begins at the low dropped chainage of the selected string.*

*When the **End mode** is **End (ref),** or **Typed** and the chainage is **blank**, the modification ends at the high dropped chainage of the selected string.*

Interval input

*if **non blank**, the interval to use to create cross sections and strings over the given chainage range. If **blank**, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

String string-select

select string to use for defining width/height/crossfall for the link.

Side to search input left side left side, right side, both sides

side of the hinge string to start searching to find the string to define width/height/crossfall.

Extra start/end tick box

*if **ticked**, add an extra x-section 0.1 mm **before** the start/end chainage.*

Comment input

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active tick box tick

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

OK button

OK *stores the values in the fields and removes the panel BUT no **recalc** is done.*

Apply button

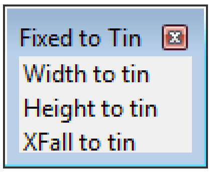
Apply *stores the values and leaves the panel on the screen.*

*If the **MTF** is being used in an **Apply MTF** and **Auto recalc** is ticked in the MTF, then whenever the **Apply** button is clicked, a **recalc** of the associated **Apply MTF** for the MTF is done.*

Continue to the next section 21.11.0.6 Fixed Link - to Tin - Superseded or go to the replacement options 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

21.11.0.6 Fixed Link - to Tin - Superseded

The **fixed to tin** walk-right brings up the **fixed to Tin** menu with options to calculate the width, xfall or height to get to a given tin.

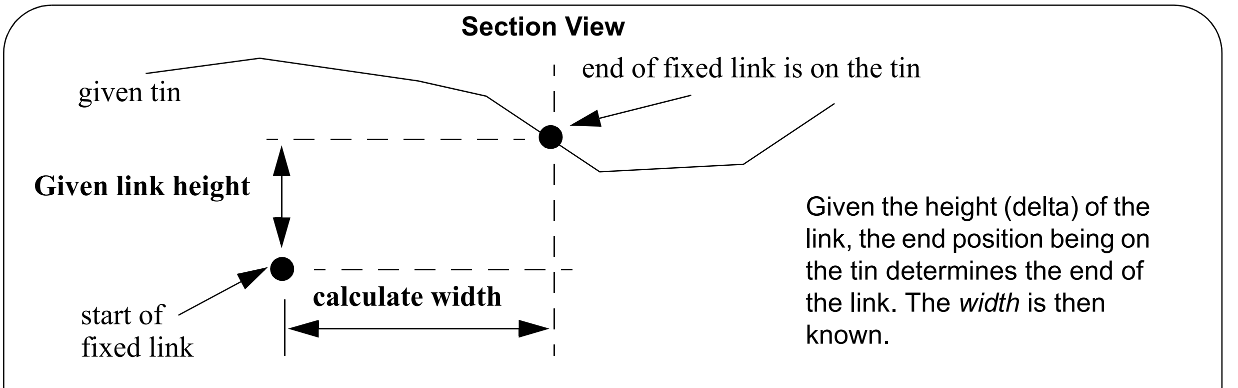


For *Width/Height/Xfall to tin*, go to the next section Fixed Link - Modify Width, Height or Xfall to Sit on User Tin

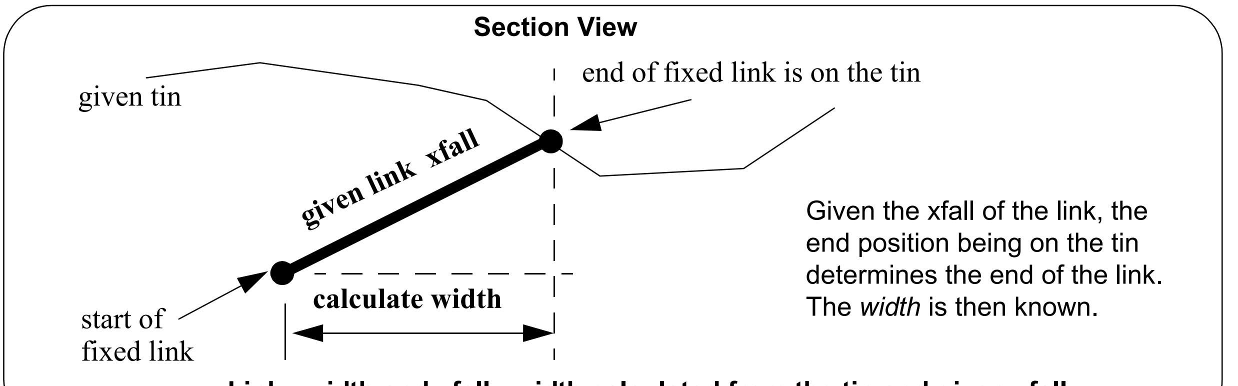
Fixed Link - Modify Width, Height or Xfall to Sit on User Tin

Width to Tin

For a fixed link defined by *width* and *height* or *width* and *xfall*, **Width to tin** calculates the **width** of the link as the width required so that the link will sit on the user given tin at the ***height/xfall given in the link***. The option gives an error for a link defined by *height* and *xfall*.



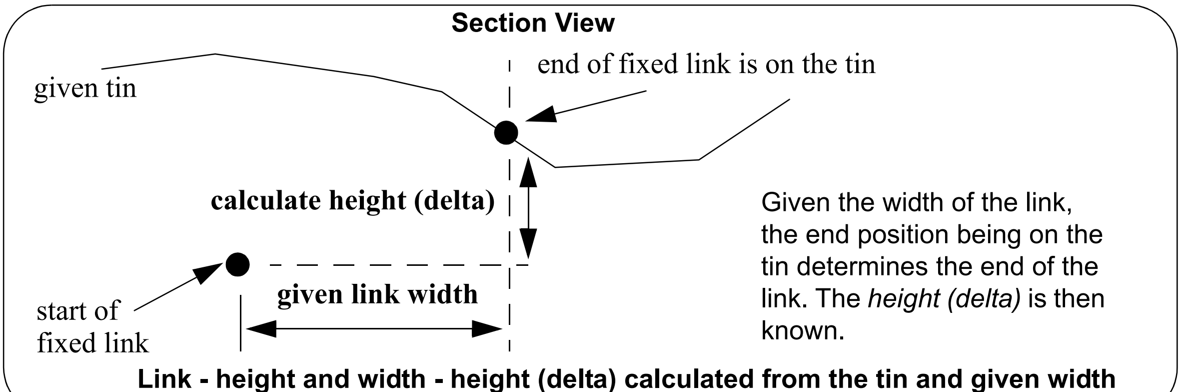
Link - width and height - width calculated from the tin and given height (delta)



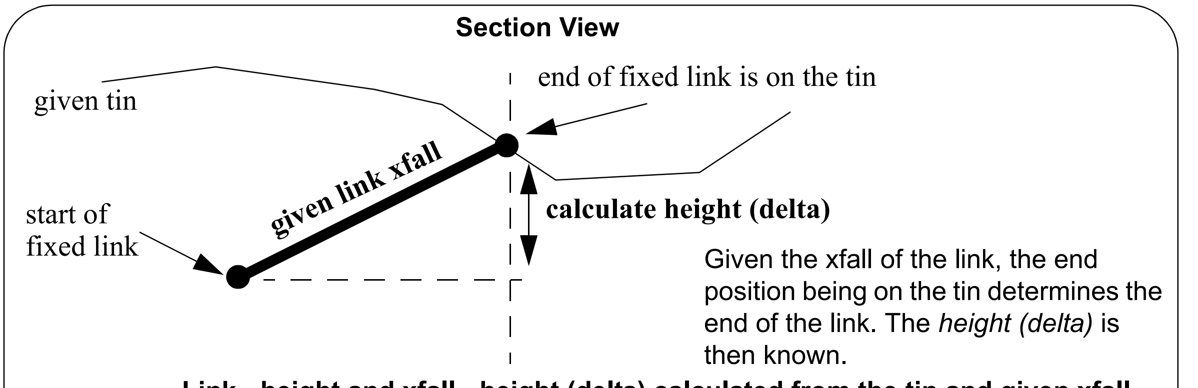
Link - width and xfall - width calculated from the tin and given xfall

Height to Tin

For a fixed link defined by *height* and *width* or *height* and *xfall*, the **Height to tin** calculates the ***height*** of the link as the difference in the height of the start point of the link, and the height that is required so that the link will sit on the user tin at the ***width/xfall given in the link.*** The option gives an error for a link defined by *width* and *xfall*.



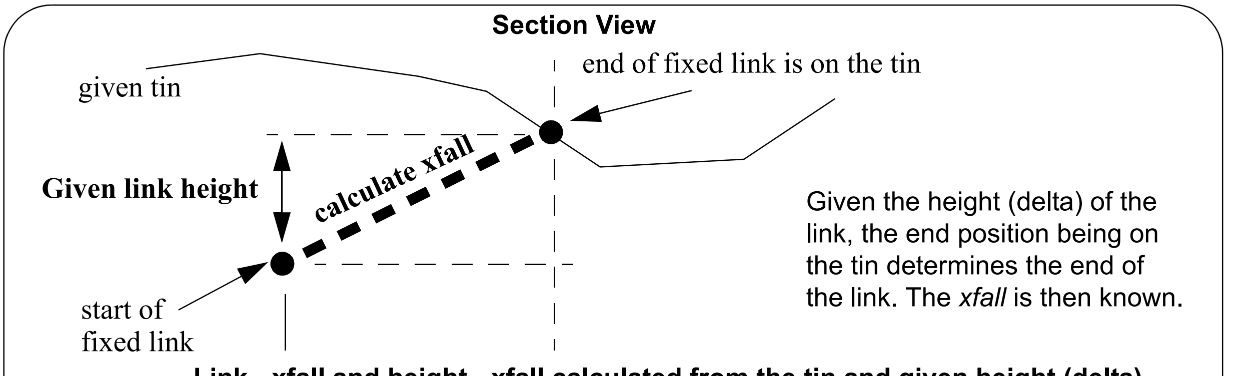
Link - height and width - height (delta) calculated from the tin and given width



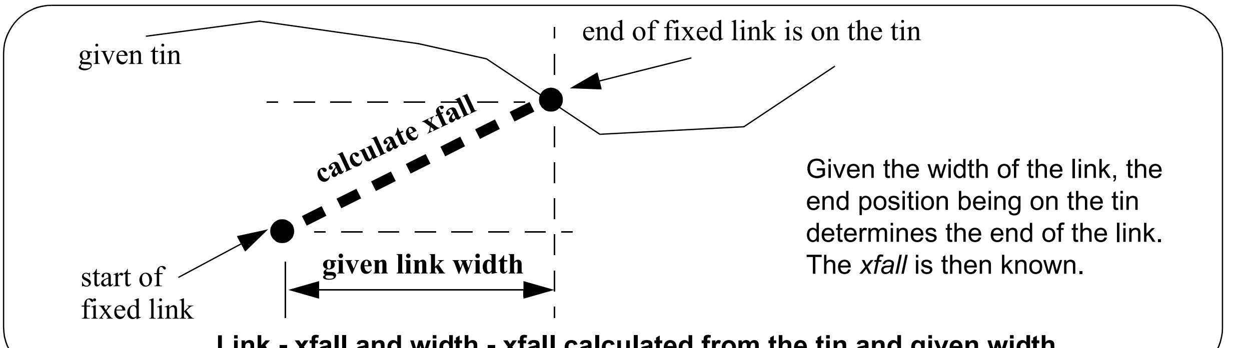
Link - height and xfall - height (delta) calculated from the tin and given xfall

Xfall to Tin

For a fixed link defined by *xfall and width* or *xfall* and *height*, **Xfall to tin** calculates the **xfall** of the link as the *xfall* required so that the link will sit on the user tin at the ***width/height given in the link***. The option gives an error for a link defined by *width* and *height*.

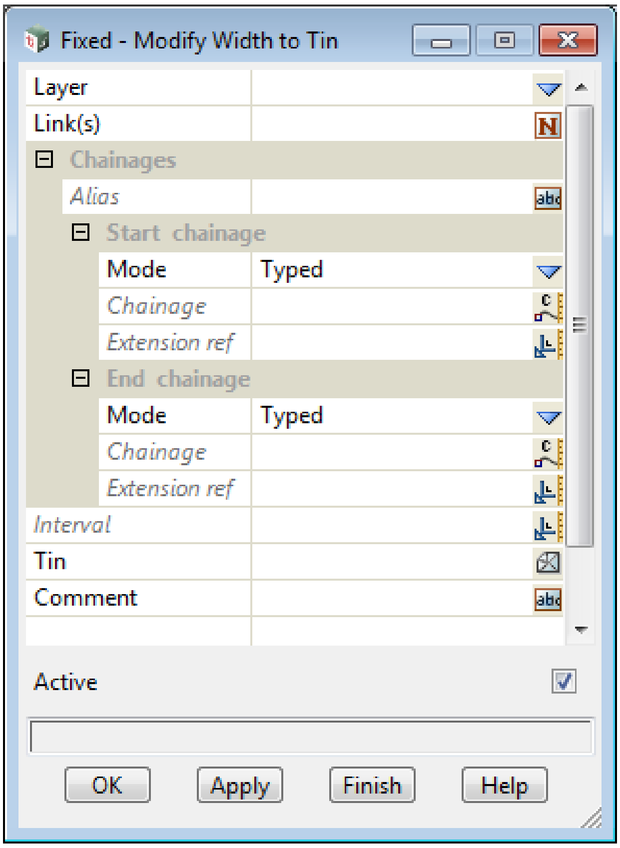


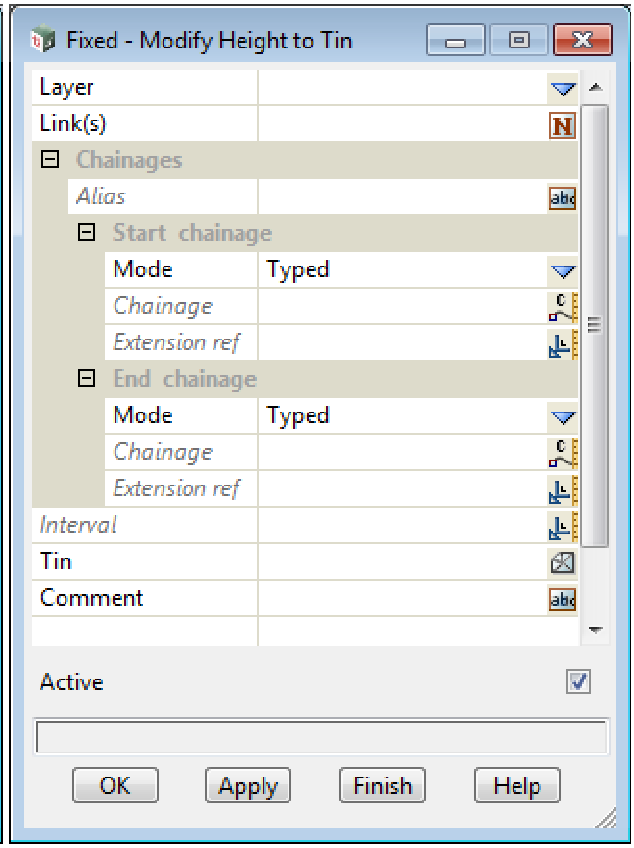
Link - xfall and height - xfall calculated from the tin and given height (delta)

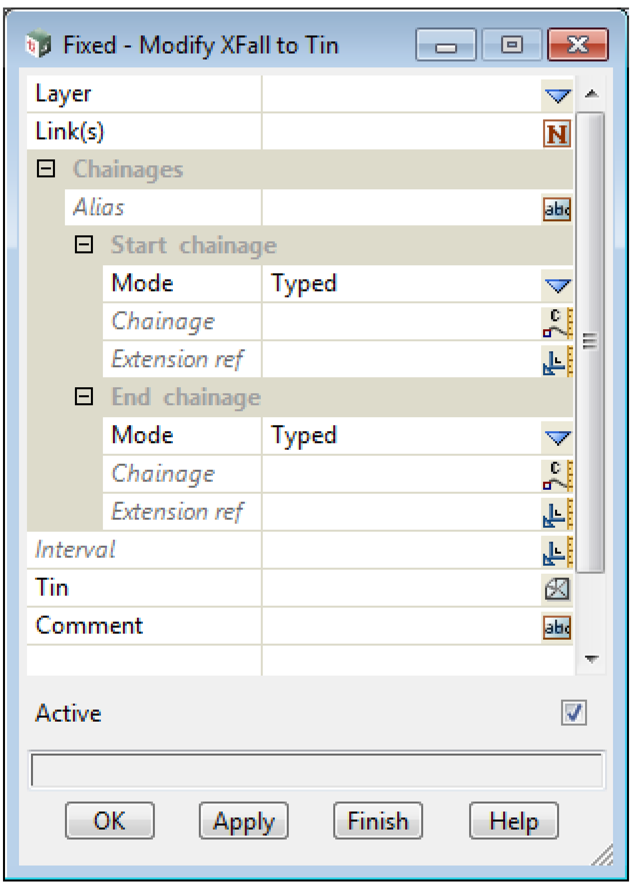


Link - xfall and width - xfall calculated from the tin and given width

Selecting **Width to tin**, **Height to tin, or Xfall to tin** brings up the **Fixed - Modify Width to Tin**, **Fixed - Modify Height to Tin** and **Fixed - Modify Xfall to Tin** panels respectively.


Fixed - Modify Width to Tin
Layer
Link(s)
Chainages
Alias
Start chainage
Mode
Typed
Chainage
Extension ref
End chainage
Mode
Typed
Chainage
Extension ref
Interval
Tin
Comment
Active
OK
Apply
Finish
Help


Fixed - Modify Height to Tin
Layer
Link(s)
Chainages
Alias
Start chainage
Mode
Typed
Chainage
Extension ref
End chainage
Mode
Typed
Chainage
Extension ref
Interval
Tin
Comment
Active
OK
Apply
Finish
Help


Fixed - Modify XFall to Tin
Layer
Link(s)
Chainages
Alias
Start chainage
Mode
Typed
Chainage
Extension ref
End chainage
Mode
Typed
Chainage
Extension ref
Interval
Tin
Comment
Active
OK
Apply
Finish
Help

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu

name of the link to modify.

Start/End mode	choice box	Start (ref)/End (ref)	

defines the start/end chainages for modifying the link.

Interval input

*if **non blank**, the interval to use to create cross sections and strings over the given chainage range. If **blank**, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

Tin input available tins

the tin to use for defining the width/height/xfall

Extra start/end tick box

*if **ticked**, add an extra x-section 0.1 mm **before** the start/end chainage.*

Comment input

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active tick box tick

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

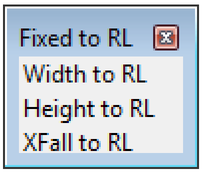
OK/Apply button

OK *stores the values in the fields and removes the panel.*
Apply *stores the values and leaves the panel on the screen.*

Continue to the next section 21.11.0.7 Fixed Link - to RL - Superseded or go to the replacement options 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

21.11.0.7 Fixed Link - to RL - Superseded

The **fixed to RL** walk-right brings up the **fixed to RL** menu with options to calculate the width, xfall or height to get to a given RL.



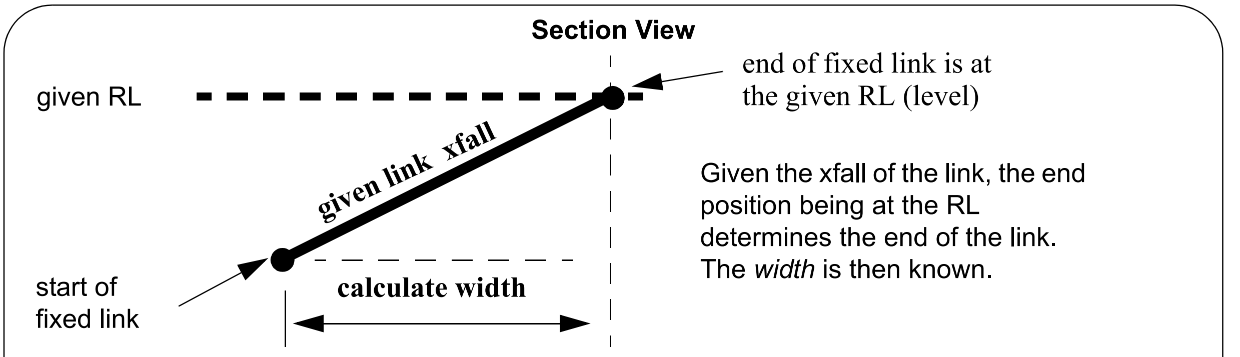
For *Width/Height/Xfall to tin*, go to the next section Fixed Link - Modify Width, Height or Xfall to Get to an RL

Fixed Link - Modify Width, Height or Xfall to Get to an RL

Width to RL

The **Modify Width to RL**, only applies to links defined by *Width and Xfall* NOT Width and Height or Xfall and Height.

For a fixed link defined by *width* and *xfall*, **Modify Width to RL** calculates the **width** required for the link to end at the given RL using the ***xfall given in the link***.



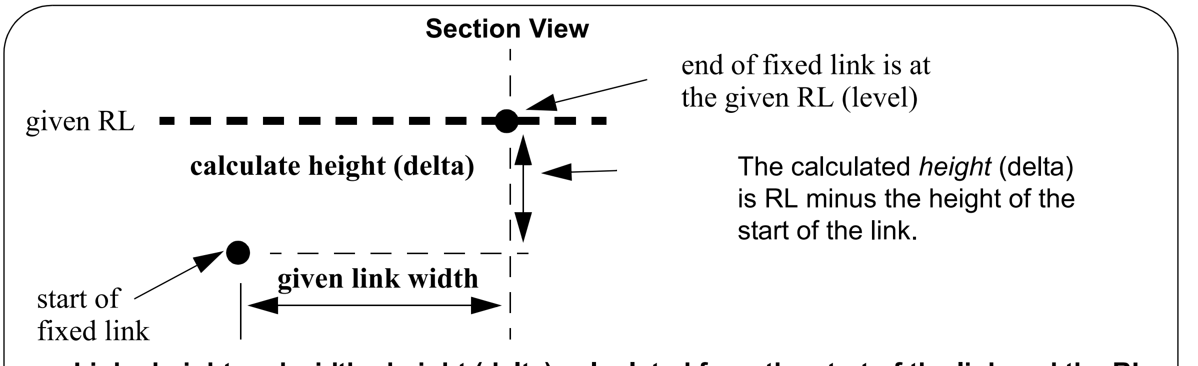
Link - width and xfall - width calculated to get to the RL at the given width

The **Modify Width to RL** gives an error for a link defined by *width and height* or *xfall and height.*

Height to RL

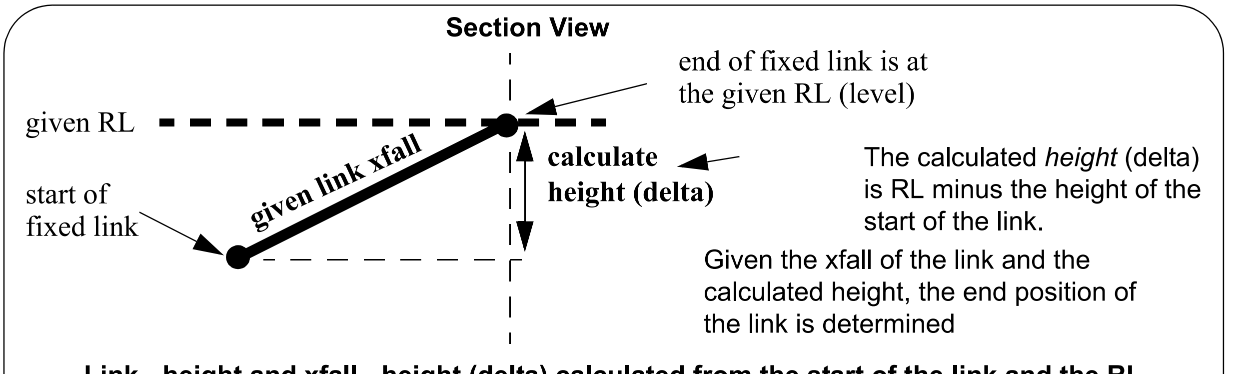
The **Modify Height to RL**, only applies to links defined by *Width and Height and Height NOT Width and Xfall.*

For a fixed link defined by *height and width*, the **Modify Height to RL** calculates the ***height*** required for the link as the difference in height of the start point of the link and the given **RL.**



Link - height and width - height (delta) calculated from the start of the link and the RL

For a fixed link defined by *height and xfall*, the **Modify Height to RL** calculates the ***height*** required for the link to end at the given RL using the ***xfall given in the link***. That is, the difference in the height of the start point of the link, and the height that is required so that the link will sit on the RL at the ***given xfall.***



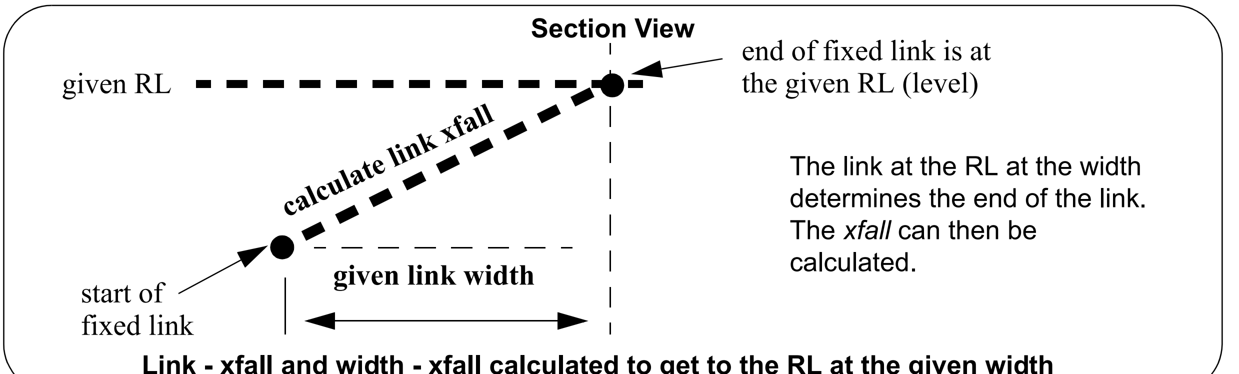
Link - height and xfall - height (delta) calculated from the start of the link and the RL

The **Modify Height to RL** gives an error for a link defined by *width and xfall.*

Xfall to RL

The **Modify Xfall to Get an RL**, only applies to links defined by *Xfall and Width*, NOT Xfall and Height or *Width and Height.*

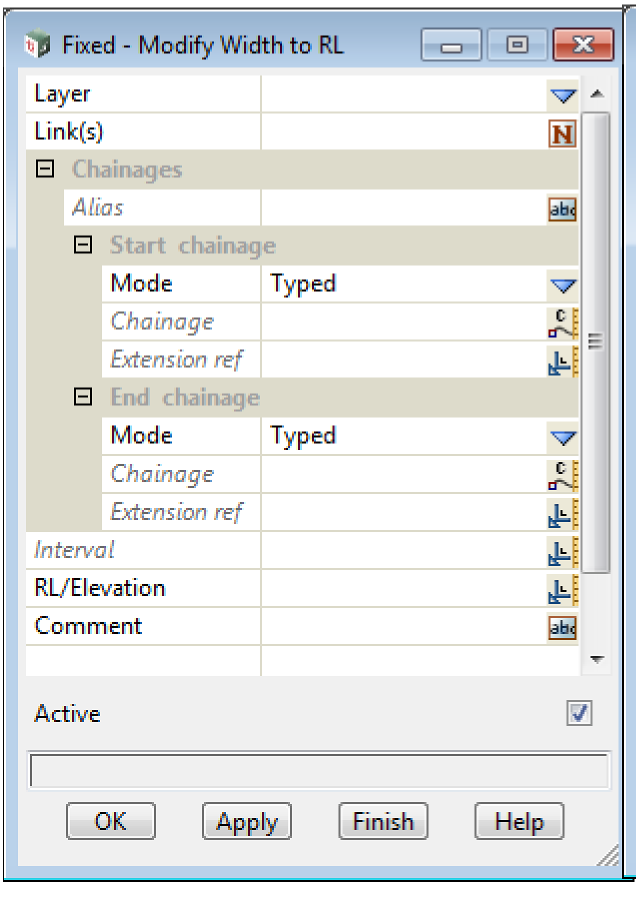
For a fixed link defined by *xfall and width*, **Modify Xfall to RL** calculates the **xfall** of the link required for the link to end at the given RL using the ***width given in the link***.

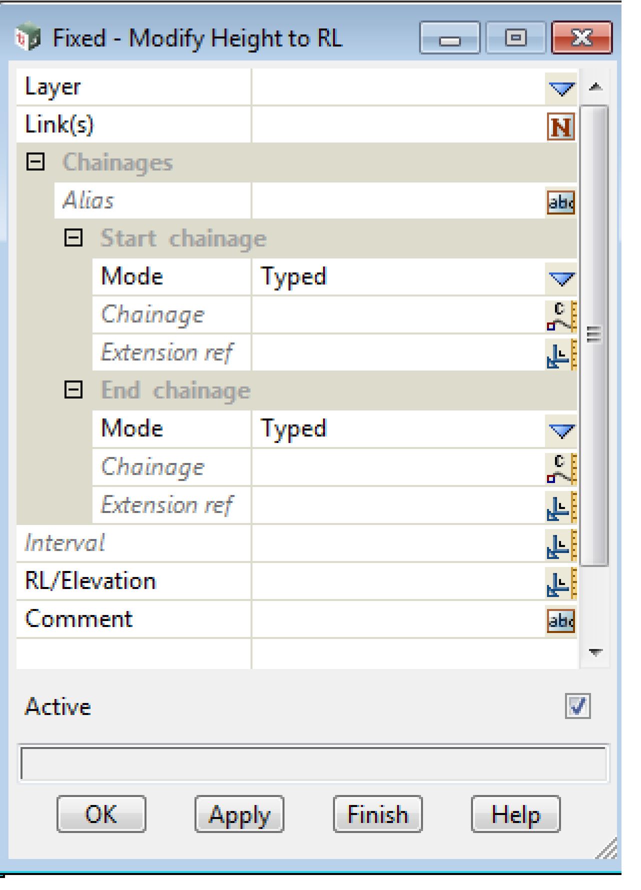


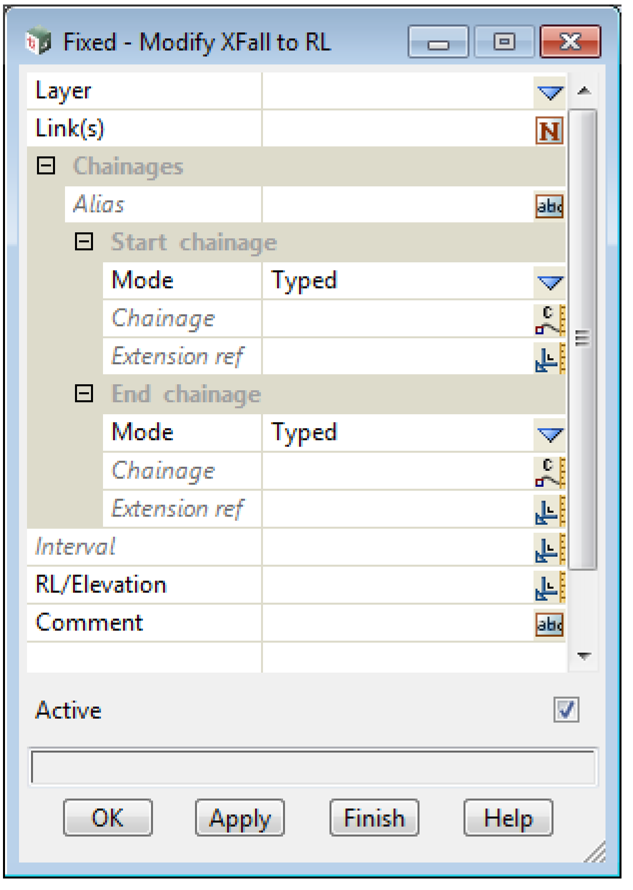
Link - xfall and width - xfall calculated to get to the RL at the given width

The **Modify Xfall to RL** gives an error for a link defined by *xfall and height*, or *width and height*.

Selecting **Width to RL**, **Height to RL, or Xfall to RL** brings up the **Fixed - Modify Width to RL**, **Fixed - Modify Height to RL** and **Fixed - Modify Xfall to RL** panels respectively.







The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu

name of the link to modify.

Start/End mode	choice box	Start (ref)/End (ref)	

defines the start/end chainages for modifying the link.

Interval input

*if **non blank**, the interval to use to create cross sections and strings over the given chainage range. If **blank**, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

RL/Elevation double box

the RL (Elevation) to be reached at the end of the link.

Extra start/end tick box

*if **ticked**, add an extra x-section 0.1 mm **before** the start/end chainage.*

Comment input

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active tick box tick

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

OK button

OK *stores the values in the fields and removes the panel BUT no **recalc** is done.*

Apply button

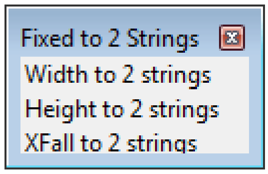
Apply *stores the values and leaves the panel on the screen.*

*If the **MTF** is being used in an **Apply MTF** and **Auto recalc** is ticked in the MTF, then whenever the **Apply** button is clicked, a **recalc** of the associated **Apply MTF** for the MTF is done.*

Continue to the next section 21.11.0.8 Fixed Link - to Two Strings - Superseded or go to the replacement options 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

21.11.0.8 Fixed Link - to Two Strings - Superseded

The **fixed to 2 strings** walk-right brings up the **fixed to 2 Strings** menu with options to calculate the width, xfall or height from two given strings.



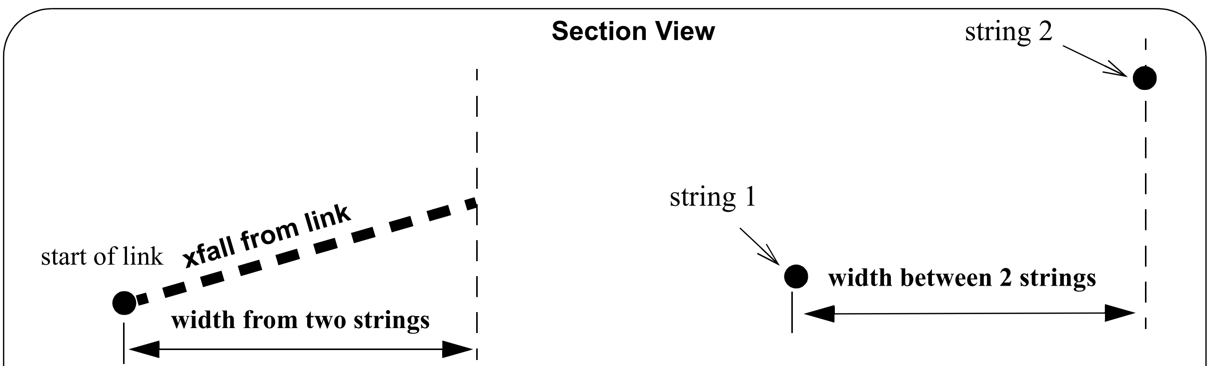
For *Width/Height/Xfall to 2 Strings*, go to the next section Fixed Link - Width, Height or Xfall between Two Strings

Fixed Link - Width, Height or Xfall between Two Strings

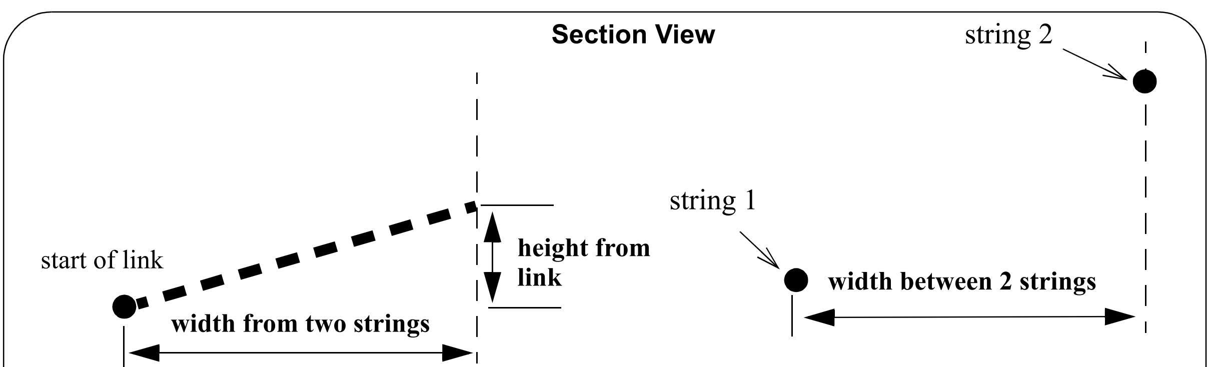
Width Between Two Strings

For a fixed link defined by *width* and *height* or *width* and *xfall*, **Width to 2 strings** sets the **width** for a link to be the **width between two** existing ***12d Model*** strings,

The option will give an error for a link defined by *height* and *xfall*.



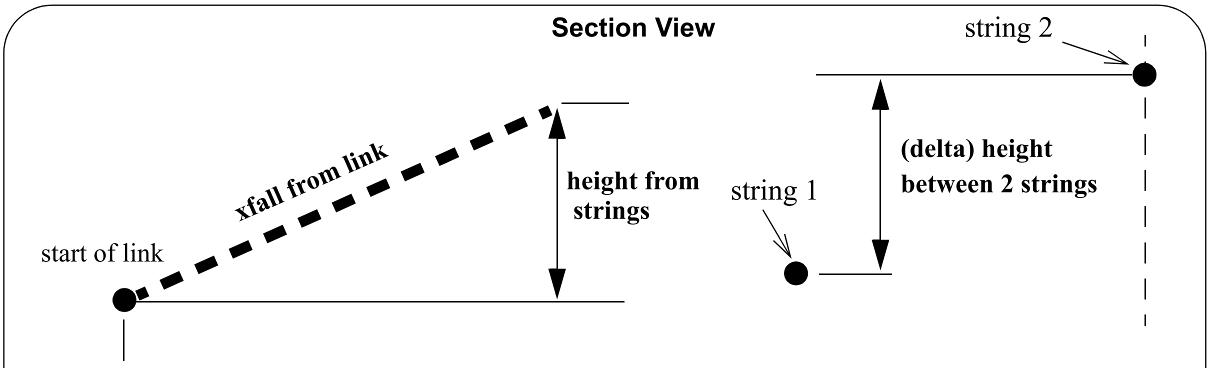
Modify Width to be that between two Strings - for Link defined by Width and Xfall



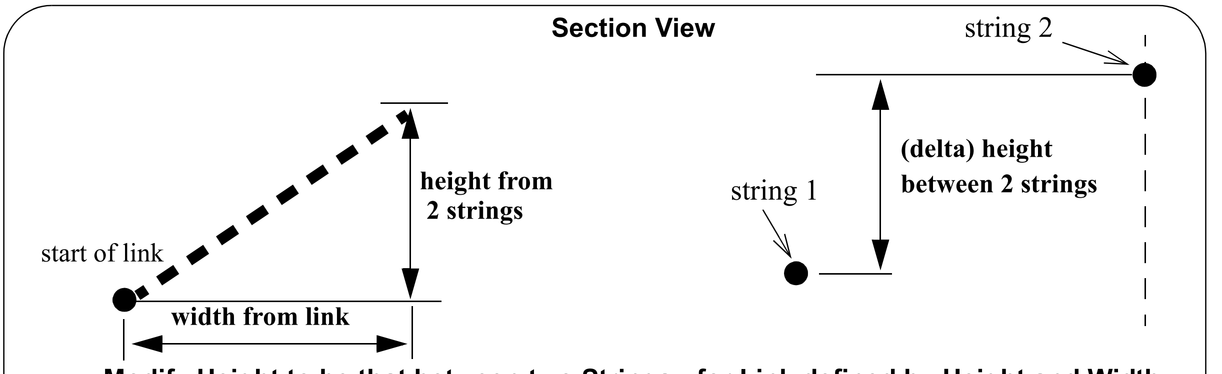
Modify Width to be that between two Strings - for Link defined by Width and Height

Height Between Two Strings

For a fixed link defined by *height* and *width* or *height* and *xfall*, **Height to 2 strings** sets the **height** for the link to be the **height between two** existing ***12d Model*** strings, The option will give an error for a link defined by *width* and *xfall*.



Modify Height to be that between two Strings - for Link defined by Height and Xfall

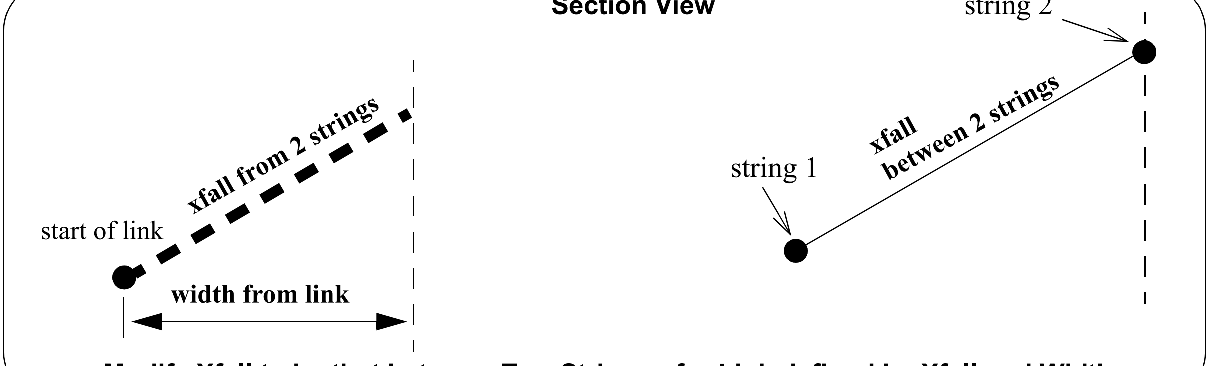


Modify Height to be that between two Strings - for Link defined by Height and Width

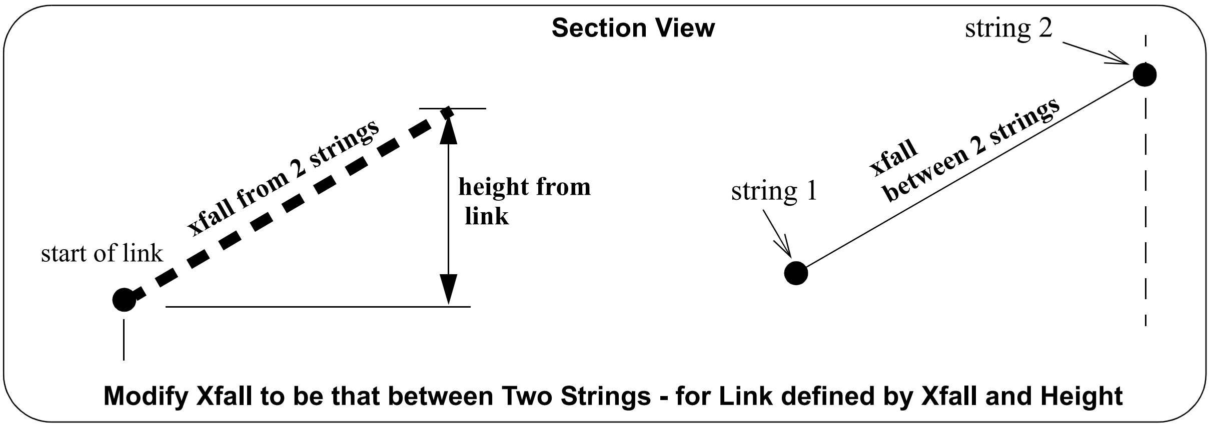
Xfall Between Two Strings

For a fixed link defined by *xfall* and *width* or *xfall* and *height*, **Xfall to 2 strings** sets the **xfall** for the link to be the **xfall between two** existing ***12d Model*** strings,

The option will give an error for a link defined by *width* and *height*.

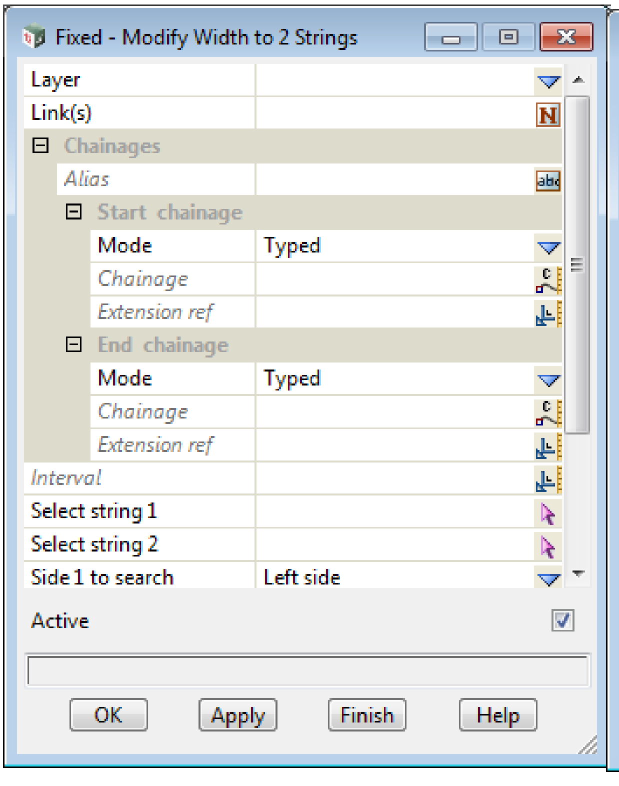


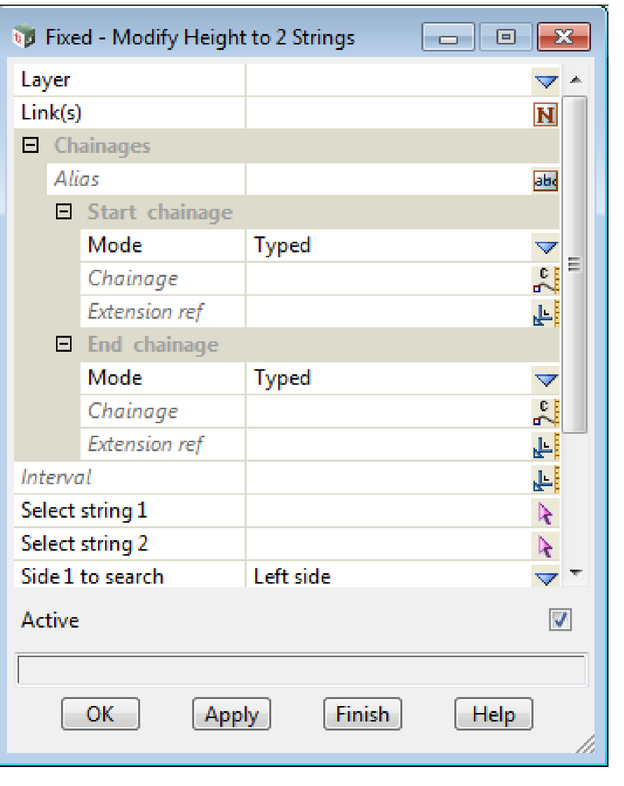
Modify Xfall to be that between Two Strings - for Link defined by Xfall and Width

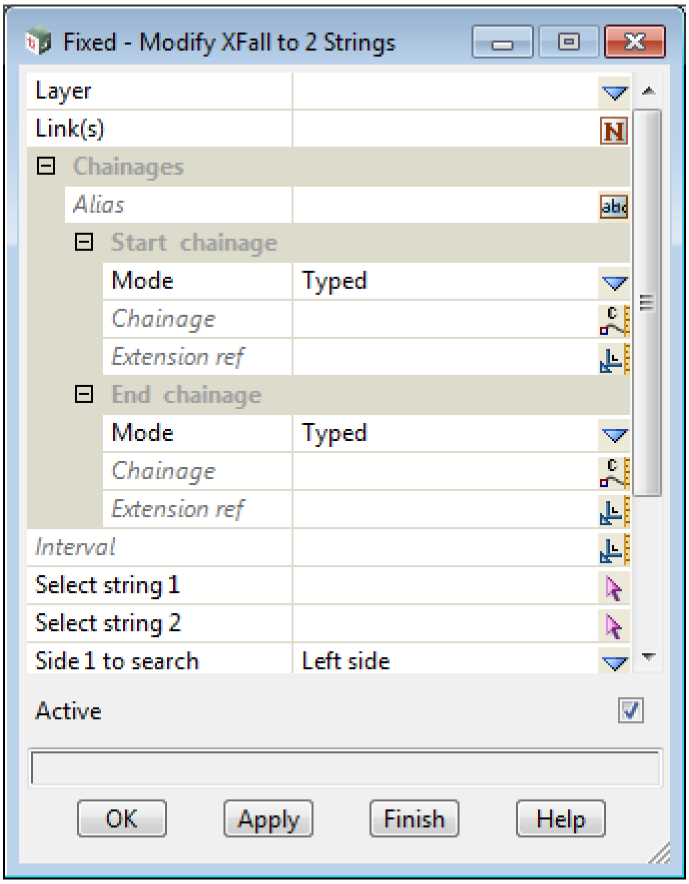


Modify Xfall to be that between Two Strings - for Link defined by Xfall and Height

Selecting the **width to 2 strings**, **height to 2 strings or xfall to 2 strings** option brings up the **Fixed - Modify Width to 2 Strings**, **Fixed - Modify Height to 2 Strings,** and **Fixed - Modify Xfall to 2 Strings** panels respectively.







The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Link name	input		select name menu

name of the link to modify.

Start/End mode	choice box	Start (ref)/End (ref)	

defines the start/end chainages for modifying the link,

Interval	input		

*if **non blank**, the interval to use to create cross sections and strings over the given chainage range. If **blank**, the* **Section separation** *value from the* **Apply MTF** *panel is used.*

String 1	string-select		

select the first string to use for defining width/height/crossfall for the link.

String 2	string-select		

select the second string to use for defining width/height/crossfall for the link.

Side 1 to search	input	left side	left side, right side, both sides

side of the hinge string to start searching to find string 1 to use in defining width/height/crossfall.

Side 2 to search	input	left side	left side, right side, both sides

side of the hinge string to start searching to find string 2 to use in defining width/height/crossfall.

Extra start/end	tick box		

*if **ticked**, add an extra x-section 0.1 mm **before** the start/end chainage.*

Comment input

comment to add to the end of the line. In the file, the comment will be preceded by //.

Active tick box tick

*if **ticked**, use this modifier.*
*if **not ticked**, don't use this modifier.*

OK button

OK *stores the values in the fields and removes the panel BUT no **recalc** is done.*

Apply button

Apply *stores the values and leaves the panel on the screen.*

*If the **MTF** is being used in an **Apply MTF** and **Auto recalc** is ticked in the MTF, then whenever the **Apply** button is clicked, a **recalc** of the associated **Apply MTF** for the MTF is done.*

Continue to the next section 21.11.0.9 Fixed Link - Absolute - Superseded or go to the replacement options 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

21.11.0.9 Fixed Link - Absolute - Superseded

These options were never released. They should be replaced in your MTF by the appropriate **Absolute** option.

Fixed Link - Width Absolute - Superseded

These options were never released. They should be replaced in your MTF by the appropriate **Absolute** option.

Return to 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

21.11.0.10 Fixed Link - to Two Points - Superseded

These options were never released. They should be replaced in your MTF by the appropriate t**o 2 heights** options.

Return to 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

Fixed Link - Modify to 2 Points - Superseded

These options were never released. They should be replaced in your MTF by the appropriate t**o 2 heights** options.

Return to 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

Fixed Link - Modify Width, Height or Xfall to 2 Points - - Superseded

These options were never released. They should be replaced in your MTF by the appropriate t**o 2 heights** options.

Return to 21.2.2.2 Fixed Link Modifiers or 21.2.2 Left and Right MTF Modifiers.

22 Drainage and Sewer

Position of menu: **Design =>Drainage-Sewer**

The **Drainage-Sewer** option is used to place drainage and sewer networks within a subdivision, along highways and culverts for cross drainage. Pumps may also be included in the network. The network is placed in three dimensions including manholes (maintenance holes), and for sewer work, lot controls and house connections can be defined. A 12d drainage string is used for all of these options.

If used in conjunction with the services on a section view, interference with neighbouring pipe strings can be taken into consideration when placing the network.

Note that the drainage string and plots capabilities are only available with the drainage and sewer options, and the sewer extensions (property control, house controls and Melbourne Water plots) are only available with the sewer option.

The **drainage string** is based on a **polyline** string so that **arcs** can exist between manholes (maintenance holes). The manholes may be placed anywhere along the polyline but generally manholes are located at the polyline indices.

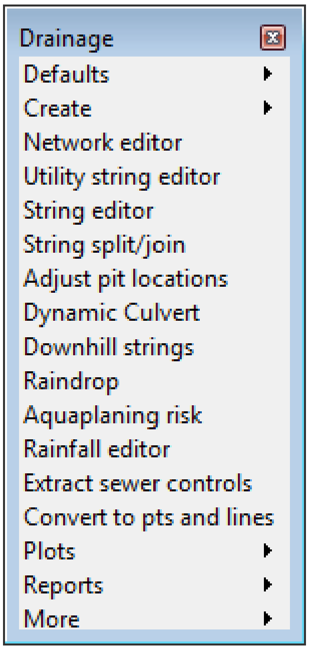
The use of the drainage string for drainage is only a subset of its use for sewer so the steps for the sewer will be given since they cover drainage as well.

The sewer process consists of a number of steps (the drainage process does not use steps 2 and 3):

1. creating the drainage or sewer network
2. vertical alignment, pipe cover and utility clash checking using the Drainage Network Editor,
3. checking that residential blocks are controlled by the sewer network
4. creating the house connections for the sewer
5. network design using custom routines, spread sheets or advanced network analysis packages via the Drainage Network Editor
6. producing Drainage Plots (plan and long section) and Reports (material quantity reports, excavation volumes and manhole construction reports)

These steps are described in the rest of this chapter.

The **Drainage** walk-right menu is laid out to reflect the standard sequence of placing drainage and wastewater pipes, *i.e.* creating, editing and plotting.



For a description of the drainage string in ***12d Model***, please go to the section Drainage Strings

For a definition of a drainage network and a junction in ***12d Model***, please go to the section Networks and Junctions.

For a description of the drainage defaults in ***12d Model***, please go to the section Drainage Definitions - Manholes and Pipes

For a description of the drainage design import/export interface, please go to the section "Drainage import/export"

For the option *Defaults*, please continue to the section Defaults.
Create, please continue to the section Create.
Editor, please continue to the section String Editor.
Network editor, please continue to the section Drainage Network Editor.
Utility string editor, please continue to the section Utility String Editor.
String editor, please continue to the section String Editor.
String split/join, please continue to the section String split/join.
Adjust Pit Locations, please continue to the section Adjust Pit Locations.
Dynamic Culvert, please continue to the section Dynamic Culvert Design.
Downhill strings, please continue to the section Downhill Strings.
Raindrop, please continue to the section Raindrop/Teardrop.
Aquaplaning risk, please continue to the section Aquaplaning Risk Assessment.
Rainfall editor, please continue to the section Rainfall File ppf Editor.
Extract Sewer Controls, please continue to the section Extract Sewer Property Controls.
Convert to pts and lines, please continue to the section Convert to Points and Lines.
Plots, please continue to the section Drainage Plots.
Reports, please continue to the section Reports.
More, please continue to the section More Drainage

22.1 Drainage Strings

A **drainage string** consists of a series of manholes (maintenance holes) at user selected (x,y,z) positions. The manholes can be joined by either straight or curved pipes.

Manholes may be any type of drainage structure and are represented by circles or rectangles. Drainage inlets, pump wet wells, gross pollutant traps, soak away pit, open channel changes in direction are all examples of manholes. The manholes have cover, grate and sump levels as well as wall and bottom thicknesses.

Pipes are the conduits connecting the manholes. Round pipes, box culverts, trapazoidal channels, pump rising mains, basin links, weirs and orifices are all examples of pipes.

Like all ***12d Model*** strings, the drainage string has an implied direction, starting at the first manhole and going in the direction towards the next manhole in the drainage string. This order is normally determined by the creation order (or string order) of the manholes.

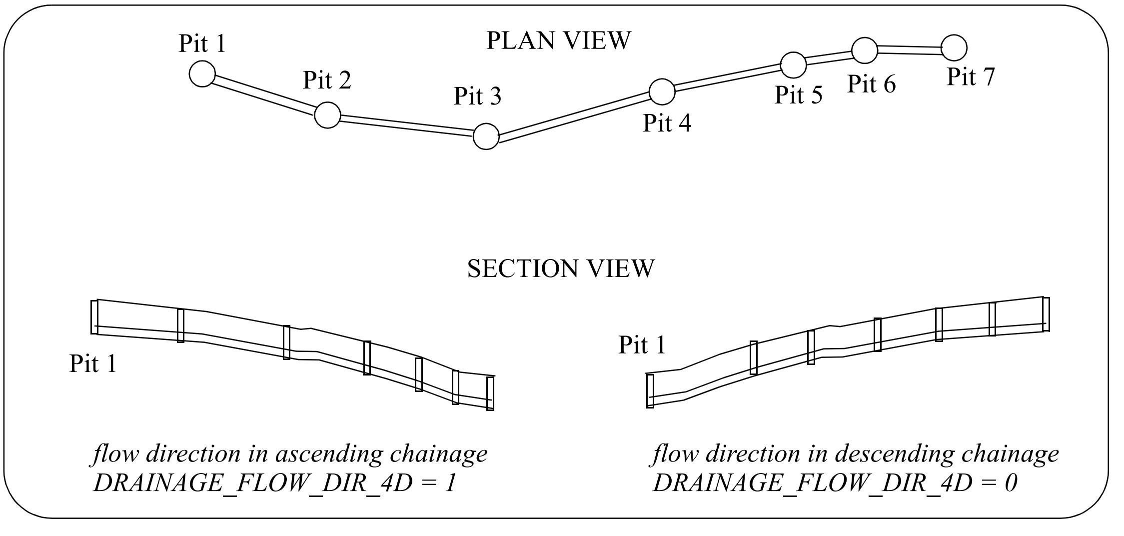
The chainage of the drainage string starts at the **first** manhole and then increases along the direction of the drainage string.

In ***12d Model***, drainage strings have a property indicating

(a) the most upstream manhole is entered as the first point of the string and hence the water flows in the direction of ascending string chainage. The flow direction is said to be in ascending chainage.

or

(b) the most downstream manhole is entered as the first point of the string and hence the water flows in the direction of descending chainage. The flow direction is said to be in descending chainage)



When a drainage string with flow direction in ascending order is profiled in a section view, the left hand side of a manhole is normally upstream and the right hand side of a manhole downstream.

It is recommended that drainage strings are entered with the flow direction in ascending chainage so that the minimum grade and cover can be satisfied as the drainage string is being placed.

That is, if the drainage string flow is in ascending chainage direction, then as manholes are appended, minimum *cover* and minimum *grade* can be automatically maintained. In all cases the invert levels can be recalculated using the Drainage Network Editor.

22.1.1 Pipes

A pipe is automatically created when a second or additional manhole is added to a drainage string. The properties of the pipe follow:

22.1.1.1 Type

22.1.1.2 Size (diameter, width, top width and natural channel shapes (dynamic only).

22.1.1.3 Inverts Levels

22.1.1.4 Wall Thickness

22.1.1.5 Minimum cover limit

22.1.1.6 Length

The pipe length by default is set to manhole centre to manhole centre. This length is displayed in the DNE and on the section view (Toggle->Grades) when the drainage string is profiled.

The pipe length maybe be also be calculated from the inside edge of the drainage manhole by using pit connections points enabled in the Drainage Network Editor on the Global Tab.

22.1.1.7 Grade

The pipe grade is calculated from the length and invert properties discussed above. This grade is displayed on the section view (Toggle->Grades) when the drainage string is profiled. It is shown as % or 1 in via

Section View Menu View => Settings =>Grade annot

Please continue to the next section Networks and Junctions.

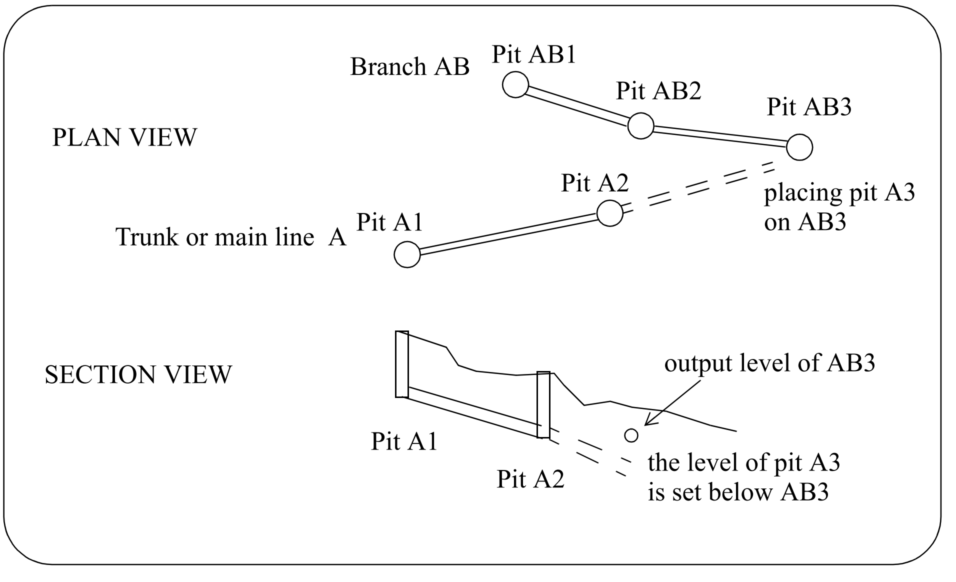
22.2 Networks and Junctions

In ***12d Model***, a **drainage network** consists of one or more drainage strings in the *same* model. Consequently, all the drainage strings in the same model are considered to be part of the same drainage network. It is suggested that **all** drainage strings in a network be entered with the same flow direction.

If two drainage strings from the same model have a manhole at exactly the same (x,y) location, then 12d Model assumes that the co-incident manholes **are** the same manhole and that the situation represents a **junction**. Most junction manholes are at the ends of the drainage strings. A junction may have a maximum of one manhole that is not at the end of the drainage string.

When water flows into a drainage string (from a branch drainage string) it is referred to as a trunk line. A trunk line may be a branch for another downstream trunk line.

Also for a network, if all the drainage strings are entered with the flow direction in ascending chainage, not only can minimum cover and minimum grade be maintained as manholes are appended, but if the branches are laid down before the trunk, then as you connect each branch to the trunk, the invert level for the trunk will be set to below the invert level of the branch line (less the default drop for the manholes). In all cases the invert levels for the entire network be recalculated using the Drainage Network Editor.



Please continue to the next section Drainage Definitions - Manholes and Pipes.

22.3 Drainage Definitions - Manholes and Pipes

A file of pipe and manhole (maintenance hole) definitions is used to create pipe and manhole (maintenance hole) types to allow tailoring for a particular project. The manhole or pipe type is one method to set many of the objects properties such as inlet capacity, thickness and roughness along with the objects user defined attributes.

When 12d Model starts up, it checks to see if an environment variable called DRAINAGE_4D exists and if it does, then the file it points to is used to provide the available types of manholes (maintenance holes) and pits.

If the environment variable is not set, then 12d Model searches for a file called **drainage.4d** in the standard 12d Model search sequence for set up files.

The drainage definitions file format is an text format and consists of one or more pipe and manhole definitions. Each definition in the file begins with the key word **Pipe** or **Manhole**, followed by the pipe or manhole type and then curly braces **{ }**. The order that the definitions appear in the file determines the order they appear in the drop down lists inside 12d Model.

IMPORTANT: the file is only read when 12d Model starts up. When the file is changed while, 12d Model is running, you must restart 12d Model for the changes to become active. If any syntax errors have been made editing the file, the line number will be displayed in the output window. The error generally slightly above this line. The most common errors are missing curly braces **{ }** and forgetting quotes around entries containing spaces

A minimal example of a drainage definitions file is:

```
// ---------------------------------------------------------------
// drainage.4d                          1/6/96
// Used to define the types of Pipes and Manholes
// ---------------------------------------------------------------
 Pipe  "PVC"  {
 }

 Pipe  "VC"  {
 }
 Pipe  "PVC Extra Heavy"  {
 }
 Pipe  "Plastic" {
 }

 Manhole  "CONC COVER"  {
 }
 Manhole "Gatic"  {
 }
 Manhole  "Rubber"  {
 }
```

Notes

1. spaces in text - any text string that includes spaces or only numbers, must be enclosed in double quotes **"**.
2. comments - anything after // until the end of the line is ignored.
3. blank lines - blank lines are ignored
4. Duplicate definitions are not allowed.

Please continue to the next section Drainage Definitions - Manhole Types.

22.3.1 Drainage Definitions - Manhole Types

Manhole types may be used to set the following manhole properties via the DNE.

- manhole diameter, length/width and thickness
- manhole description, notes, group, ku/kw method, ku/kw values, rational engine design freeboard,
- manhole level modes for cover level, grate level, survey setout level and sump levels,
- survey setout xy modes and road chainage modes
- user defined manhole attributes

Each definition (manhole block) in the file begins with the key word **Manhole**, followed by the manhole type and then curly braces **{ }**. The order that the definitions appear in the file determines the order they appear in the drop down lists inside 12d Model.

The minimum requirement for a pit type definition is

Manhole "type name" {

}

The type name must be unique and the braces {} cannot be () or [].

Optional manhole commands may be placed inside the braces. These commands include may include Manhole Editor Commands, Manhole Drainage Network Editor Commands and Manhole Calculation Commands.

22.3.1.1 Manhole Editor Commands

These optional commands change the properties of the manhole objects. Changes to these commands will used in creating new drainage strings and the Drainage Network Editor will prompt you to update the object if these settings are different to the strings current settings. A list of these commands follows

mhdiam x.x	set the manhole as circular (internal diameter in base units)

mhsize x.x y.y	set the manhole as rectangular (over rides mhdiam) length and width in base units

```
mhthickness    {
   diam_thickness   x.xxx a.aaa b.bbb c.ccc d.ddd
}
```

x.xxx	nominal diameter choices will appear in the DNE->Pipe->Diameter drop down
a.aaa	optional front thickness (base units) 0.000 if omitted
b.bbb	optional back thickness (base units) front thickness if omitted
c.ccc	optional left thickness in direction of chainage (base units) front thickness if omitted
d.ddd	optional right thickness in direction of chainage (base units) front thickness if omitted

22.3.1.2 Manhole Level Modes

There are several modes that may be used to calculate x,y,and z values for the manholes. A decription of each fllows.

FS tin	the manhole centre x,y location is used to obtain the level from the drainage finished surface tin

NS tin	the manhole centre x,y location is used to obtain the level from the drainage natural surface tin

Setout string	the manhole centre x,y location is dropped perpendicular onto the setout string. The x,y or z value is then obtained from this string. If this string is missing then the pit x,y location or finiahed surface tin level is used instead.

Centre string the manhole centre x,y location is dropped perpendicular onto the road centre string and the chainage value is obtained. If this string is missing an problem message will be created in the output window.

Sz + Setout stringthe manhole centre x,y location is dropped perpendicular onto the setout string. The Sz value for the manhole is then added to the z value obtained from this string. If this string is missing the Sz value is added to the finiahed surface tin level at the pit centre x,y location.

Manual the x,y,z or chainage value is manually entered by the user.

Cover RL the z level is set the the Cover RL level (after it has been recalculated.

Max Obvert The maximum obvert level from off the conduits connected to this manhole is used.

Floating sump the manhole sump level is set to the lowest invert of all the conduits connected to this manhole + the sump offset

22.3.1.3 Manhole Drainage Network Editor Commands

These optional commands store user defined data on the manholes or change the calculation modes available in the Drainage Network Editor. The set pit details button in the editor will recalculate these for all manholes in the network. A list of these commands follows

mhdesc "description" creates a pit text attribute "pit type description" and is used in the drainage pit schedule report.

mhnotes "note" creates a pit text attribute "pit type remarks" and is used in the drainage pit schedule report.

mhgroup "group" routines that select manholes will select from a manhole belonging to the same group

attribute_integer "attribute name1" x x is an integer value (no decimal, stored exactly by computers

attribute_real "attribute name2" x.xxx x is a real value (used to store numbers with decimals or very large or very small numbers

attribute_text "attribute name3" "text" text is a series of words or numbers not intended for calculations

Note: If a non-special attribute name is set for some, but not all manhole types, that attribute will be deleted on all manholes with types where the attribute is not defined.

The following special attribute commands create/modify an attribute as described above but these attributes also control calculations performed by the set pit details button on the Drainage Network Editor. The DNE fields will be locked when these attributes are defined for the selected pit type. If these attribute are not defined for the selected pit type the DNE field will not be locked and remain unchanged.

attribute_integer "cover rl mode" x DNE field ->Pit=>Main=>Cover RL mode

Mode	**x**
FS tin	0
Setout string	1
Manual	2
NS tin	3
Max Obvert	4
Sz + setout string	8

attribute_integer "grate rl mode" x DNE field ->Pit=>Main=>Grate RL mode

Mode	x
FS tin	0
Setout string	1
Manual	2
NS tin	3
Max Obvert	4
Cover RL	7
Sz + setout string	8

attribute_real "sump offset" x.xx DNE field ->Pit=>Main=>Sump offset
if sump RL mode is floating then this is the offset (negative down) from the lowest pipe invert <base units>

attribute_integer "ku method" DNE field is Pit=>Main=>Ku method

Mode	x	Description
Direct	0	user enters the ku value stored as pit real attribute "ku"
Ku,Kw - Missouri/Hare Charts		
	1	ku calculated during analysis
Ku,Kw>0 - Missouri/Hare Charts		
	2	ku calculated during analysis (ku < 0.0 changed to 0.0)
Ku - Culvert Inlet - Generic (101 or 201)		
	3	Culvert analysis using inlet control and backwater control (entrance and exit losses automatically set)
Remaining	xxx	Culvert analysis - use numbers from DNE drop down list

attribute_real "ku" x.xx DNE field is Pit=>Main=>Ku
x.xx is used to calculate pit upstream hgl when ku method is Direct

attribute_real "kw" x.xx DNE field is Pit=>Main=>Kw
x.xx is used to calculate pit hgl when ku method is Direct

attribute_integer "setout xy mode" x DNE field is Pit=>Setout=>Setout xy mode

Mode	x
FS tin	0
Setout string	1
Manual	2

attribute_integer "setout z mode" x DNE field is Pit=>Setout=>Setout z mode

Mode	x
FS tin	0
Setout string	1
Manual	2
NS tin	3
Max Obvert	4
Cover RL	7
Sz + setout string	8

attribute_integer "road chainage mode" DNE field is Pit=>Setout=>Chainage mode

Mode	x
No Road	0
Centre string	1
Manual	2

attribute_real "setout adjustment" DNE field is Pit=>Setout=>Sxy

attribute_real "setout adjustment z" DNE field is Pit=>Setout=>Sz

attribute_real "design freeboard" DNE field is Pipe=>Design=>Freeboard limit
at US pit

The following example of a channel ip point is given below. The setout modes are set, the cover and grate level modes are set and the ku (losses) are set. Finally, the inlet capacity is set to an on-grade pit with200% inlet capacity so that even in a major storm with a choke factor of 0.5 it will still have 100% inlet capacity.

```
Manhole "CHNL auto" {

  mhdesc  "channel hip-vip"
  attribute_text "lplot description1"  "OPEN CHANNEL"

  mhsize  0.0
  mhdiam  0.0
  attribute_integer "setout xy mode"        0 // centre of the channel
  attribute_integer "setout z mode"         6 // sump invert is the bottom of the channel
  attribute_integer "cover rl mode"         4 // max obvert - top of the channel
  attribute_integer "grate rl mode"         4 // max obvert - top of the channel
  attribute_integer "ku method"             0    // direct
  attribute_real    "ku"                    0.0  // zero unless interested in bend losses

  cap_config G

  cap_percent 200 // if a choke factor of 0.5 is applied then it will still have 100% inlet capacity
}

//  attribute_real    "design freeboard"      Pipe=>Design=>Freeboard limit at US pit
```

22.3.1.4 Manhole Drainage Analysis Inlet Capacity Commands

Manhole inlet configuration and bypass pit entries determine if these inlet capacity commands are used. Both may be set in the Drainage Network Editor.

22.3.1.4.1 cap_config

The inlet configuration may be set via the following command

cap_config x

Mode	x
Manhole	m
Ongrade	g
Sag	s

Inlet Configuration = manhole - no water will enter the pit through the grate. Commands not used.

Inlet configuration = on grade or sag

Bypass pit not set - 100% of the approach flow will enter the pit. Commands not used.
Bypass pit is entered. The following commands define the stormwater inlet capacity characteristics.

22.3.1.4.2 Inlet Capacity Equation

The inlet capacity equation is built up with optional components (**single polynomial** + **curve polynomial** + **curve coordinates**). Generally, only one of the components is used for each manhole type but they may all be used if desired.

inlet capacity = inlet efficiency * inlet multiplier *
[**single polynomial** + curve multiplier (**curve polynomial** + **curve coordinates**)]

An inlet efficiency (choke factor) is specified in the Drainage Network Editor. An inlet efficiency (choke factor) of 0 would stop all water from entering the inlet.

The inlet multiplier is specified with a cap_multi parameter.

Multipliers

cap_multi x.x inside a cap_curve_grade block or cap_curve_sag block, the curve inlet capacity is multiplied by this value
outside a cap_curve_grade or cap_curve_sag, the total inlet capacity is multiplied by this value

Single Polynomial

The inlet capacity for an inlet may be specified by a single polynomial equation based on the approach flow. This is the most simplistic method and generally used for percentage capture or fixed capture rates.

inlet capacity = cap_fixed
+ cap_percent*0.01*Qa
+ cap_coeff*Qa^cap_power

Example
This example creates an inlet with a fixed inlet capacity of 0.010 (cms or cfs).

```
Manhole "fixed inlet capacity" {
        cap_fixed 0.010
}
```

Default values

cap_multi = 1.0

cap_fixed = 0.0

cap_percent = 0.0

cap_coeff = 0.0

cap_power = 1.0

Curve Polynomial

For on-grade inlets, the polynomial parameters may change with road grade and cross fall threshold values. The formula is the same for cap_fixed, cap_percent, cap_coef and cap_power. Note that each curve may have its own curve multiplier specified with a cap_multi parameter (discussed below). Some hydraulic model tests have their on grade inlet results converted to polynomial equations.

Example
This example creates an inlet where the inlet capacity polynomials have been determined for 2 road grades (1% and 3%). Note that the road_grade 0.0 command is used for the 1% road grade. Since this is the flattest road grade curve we have calculated we will start using it at a road grade of 0%.

Note that the second curve "NJ G3" will be used when the road grade reaches 2.5. The threshold value where 12d should change to the next curve is generally slight less than the road grade from the source.

```
Manhole "On grade pit type NJ" {
        cap_config G

        cap_curve_grade   "NJ 1G" {
                road_grade  0
                cap_coeff  0.215
                cap_power  0.67
        }

        cap_curve_grade   "NJ 3G" {
                road_grade  2.5
                cap_coeff  0.24
                cap_power  0.673
        }
}
```

Curve Coordinates (On grade and SAG)

For on-grade and sag inlets, the inlet capacity may be determined by entering coordinates along the inlet capacity curve. These coordinates are usually obtained from hydraulic model studies or analytical methods such as HEC-22.

For on grade inlets, the coordinates are Qapproach and Qin, and the curves may change with road grade and cross fall threshold values. The inlet capacity curves are never extrapolated.

Example

```
Manhole "Ongrade coordinates" {
        cap_config G
        cap_curve_grade "0.5G" {
                road_grade  0
                coord 0.000 0.000
                coord 0.060 0.060
                coord 0.140 0.112
                coord 0.260 0.174
                coord 0.430 0.244
                coord 0.500 0.270
        }
        cap_curve_grade "1G" {
                road_grade  0.75
                coord 0.000 0.000
                coord 0.060 0.060
                coord 0.140 0.108
                coord 0.260 0.164
                coord 0.430 0.227
                coord 0.500 0.248
        }
}
```

For sag inlets, the coordinates are Depth (base units) and Qin, and there is only one curve. Each curve has a curve muliplier specified with a cap_multi parameter (discussed below).

Example

```
Manhole "SAG coordinates" {
        cap_config S
        cap_curve_sag "SAG" {
                coord 0.000 0.040
                coord 0.045 0.101
                coord 0.070 0.151
                coord 0.095 0.245
                coord 0.120 0.302
                coord 0.170 0.347
                coord 0.220 0.371
                coord 0.270 0.391
        }
}
```

22.3.1.4.3 Polynomial Inlet Capacity Commands

The following commands are used to set the parameters in the following polynomial equation

```
inlet capacity =    cap_fixed
                    + cap_percent*0.01*Qa
                    + cap_coeff*Qa^cap_power
```

cap_fixed	x.x	cms or cfs
cap_percent	x.x	percentage (0 to 100)
cap_coeff	x.x	multiplier
cap_power	x.x	exponent

22.3.1.4.4 Inlet Curve Block Commands

Inlet curve blocks may be specified for both on-grade or sag inlets. Inside the curve block you may include the Polynomial Inlet Capacity Commands and Coordinate Inlet Capacity Commands.

```
cap_curve_grade "unique name for the pit type" {
        road_grade x.xx
        road_xfall x.xx
}
```

Inside the cap_curve_grade block the road grade and road crossfal threshold values (percent) may be set. The road grade and crossfall are calculated by the Drainage Network Editor. When the 12d analysis engine selects the inlet curve, all curves with the same road_xfall are grouped together and then within the crossfall group the road_grade curves is selected.The inlet curve with the maximum grade threshold that is less than or equal to the road grade is selected.

Rules for 'cap_curve_grade' entries:

- Only applicable to on-grade pits.
- All cap_curve_grade names must be unique within a Manhole block
- If both 'road_grade' and 'road_xfall' entries are omitted, only one cap_curve_grade entry is allowed within a pit.
- The cap_curve_grade 'coord' entries (if used) must be in order of increasing Qa.

```
cap_curve_sag "unique name for the pit type" {
}
```

Rules for 'cap_curve_sag' entries:
Only applicable to sag pits.
Only one cap_curve_sag entry is allowed within a pit, and it must have a valid name.

22.3.1.4.5 Coordinate Inlet Capacity Commands

The coord command must be used inside the cap_curve_grade or cap_curve_sag grouping

coord x.xx y.yy

x.xx must be in increasing order.

For cap_curve_grade group, the coord command has the parameters Qapproach and Qin

For cap_curve_sag group, the coord command has the parameters Depth and Qin

Please continue to the next section Drainage Definitions - Pipe Types.

22.3.2 Drainage Definitions - Pipe Types

Pipe types may be used to set the following pipe properties via the DNE.

- pipe nominal/actual diameters and thickness
- roughness method and value
- rational method design mode and design percent depth
- minimum pipe height for the rational design engine
- user defined pipe attributes

Each definition (pipe block) in the file begins with the key word **Pipe**, followed by the pipe type and then curly braces **{ }**. The order that the definitions appear in the file determines the order they appear in the drop down lists inside 12d Model.

The minimum requirement for a pipe type definition is

```
Pipe "name" {
}
```

The name must be unique and the braces {} cannot be () or [].

Example:

```
Pipe "CHNL GRASS PROPOSED" {  //Open Channel created below the tin. cover set in cover file to 0.0
  roughness_n 0.040
  attribute_integer "design size mode"     3  // open channel mode
}
```

22.3.2.1 Pipe Drainage Network Editor Commands

22.3.2.1.1 Pipe Thickness

```
pipethickness     {
                     diam_thickness   x.xxx y.yyy a.aaa b.bbb c.ccc d.ddd
                  }
```

x.xxx nominal diameter choices will appear in the DNE->Pipe->Diameter drop down
y.yyy internal diameter (base units) will be entered into the DNE->Pipe->Diameter field
a.aaa optional top thickness (base units) 0.000 if omitted
b.bbb optional bottom thickness (base units) top thickness if omitted
c.ccc optional left thickness in direction of chainage (base units) top thickness if omitted
d.ddd optional right thickness in direction of chainage (base units) top thickness if omitted

22.3.2.1.2 Pipe Attributes

attribute_integer "attribute name1" x x is an integer value (no decimal, stored exactly by computers

attribute_real "attribute name2" x.xxx x is a real value (used to store numbers with decimals or very large or very small numbers

attribute_text "attribute name3" "text" text is a series of words or numbers not intended for calculations

The following special attribute commands create/modify an attribute as described above but these attributes also control calculations performed by the set pit details button on the Drainage Network Editor. The DNE fields will be locked when these attributes are defined for the selected pipe type. If these attribute are not defined for the selected pipe type the DNE field will not be locked and remain unchanged.

roughness_n x.xx> DNE field ->Pipe=>Main=>Roughness
DNE field ->Pipe=>Main=>Roughness type (set to Manning)

roughness_k x.xx DNE field ->Pipe=>Main=>Roughness
Colebrook k roughness value in millimetres
DNE field ->Pipe=>Main=>Roughness type (set to Colebrook)

attribute_real x.xx DNE field ->Pipe=>Design=>Min pipe height
min height in base units

attribute_integer "design size mode" x DNE field ->Pipe=>Design=>Design mode

Mode	**x**
Pressurised Pipe: Freeboard design	0
Part-full Pipe: Freeboard design	1
Part-full Pipe: Flow depth design	2
Open Channel: Freeboard design	3

attribute_real "design percent depth" x.xx DNE field
Pipe=>Design=>Flow-depth at pipe entrance

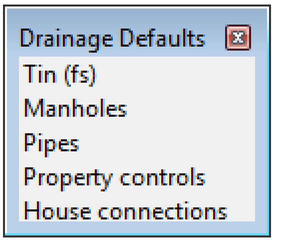
Please continue to the next section Defaults.

22.4 Defaults

Position of menu: **Design =>Drainage-Sewer =>Defaults**

The **defaults** menu sets default tin, manhole (maintenance hole) information, drainage pipe information, property control (sewer module only) and house connection (sewer module only) defaults which are all used when defining drainage networks.

The **Defaults** walk-right menu is

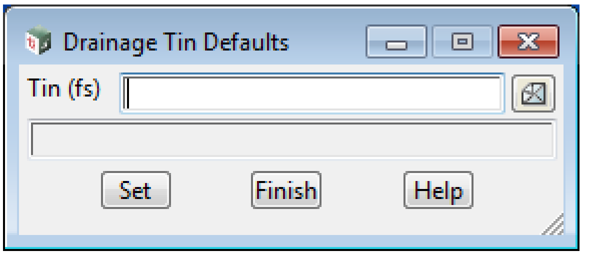


For the option *Tins (fs)*, please continue to the section Tin (fs).
Manholes, please continue to the section Manholes (Maintenance Holes).
Pipes, please continue to the section Pipes.
Property controls, please continue to the section Property Controls.
House connections, please continue to the chapter House Connections.

22.4.1 Tin (fs)

Position of option on menu: **Design =>Drainage-Sewer =>Defaults =>Tin (fs)**

On selecting the **Tin (fs)** option, the **Drainage Tin Defaults** panel is displayed.



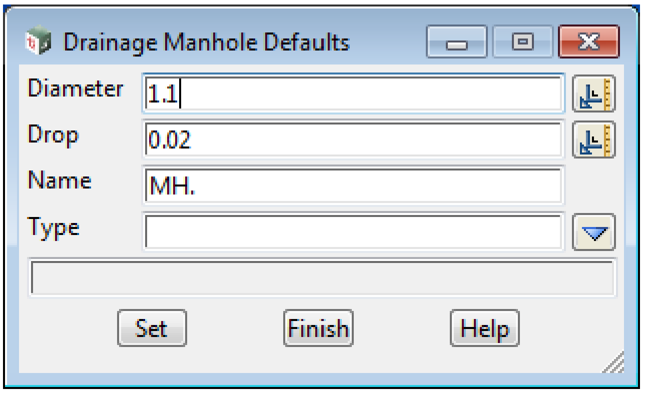
This panel is for setting the default finished surface tin in the **Create Drainage Strings** panel which is used for creating a new drainage string.

For the drainage string, the finished surface tin is used as the surface that manholes automatically sit on when z float is set on, and for defining cover when placing controls and connections.

22.4.2 Manholes (Maintenance Holes)

Position of option on menu: **Design =>Drainage-Sewer =>Defaults =>Manholes**

On selecting the **Manholes** option, the **Drainage Manhole Defaults** panel is displayed.



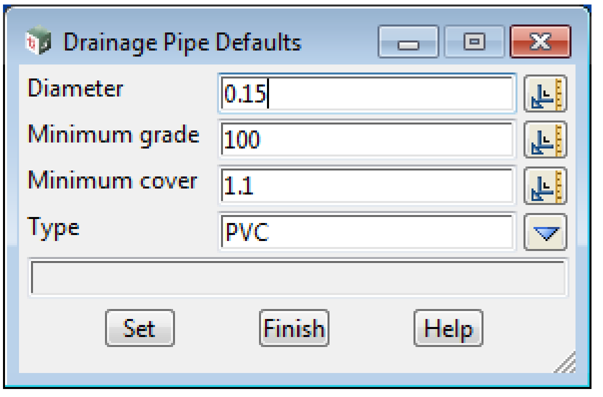
These defaults are used when creating a manhole in a drainage string. The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Diameter	input	1.1	
default diameter of a drainage manhole			
Drop	input	0.02	
drop (metres) through the manhole			
Name	input	MH.	
default name for the manhole Note that if a manhole name is EOL or eol, then the diameter of the manhole is forced to be zero.			
Type	input	CONC COVER	CONC COVER, GATIC
the default cover or lid type of the manholes			
Set	button		
set the drainage manhole defaults to the values in the above fields.			

22.4.3 Pipes

Position of option on menu: **Design =>Drainage-Sewer =>Defaults =>Pipes**

On selecting the **Pipes** option, the **Drainage Pipes Defaults** panel is displayed.



This panel is for setting the default drainage pipe diameter, grade, cover and type. These defaults are used when creating a pipe in a drainage string.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Diameter	input	0.1	
default diameter of the pipe.			
Minimum grade	input	1.0	
the minimum grade (measured as 1: value) used when laying down the pipe.			
Minimum cover	input	1.0	
the minimum cover, measured in world units from the surface to the top of the pipe (obvert); used when laying down the pipe.			
Type	input	PVC	PVC, VC, PVC X/HEAVY
the default type of the pipe			
Set	button		
set the drainage pipe defaults to the values in the above fields.			

IMPORTANT NOTE.

If the drainage string is laid down in the direction of flow (and hence the flow direction is in ascending chainage), then the minimum grade and minimum cover along the pipe are maintained as the drainage string is created. Otherwise the minimum grade and cover cannot be maintained.

Cover for the pipe segment can also be calculated and/or set afterwards by the **pipe=>cover** option in the drainage string editor.

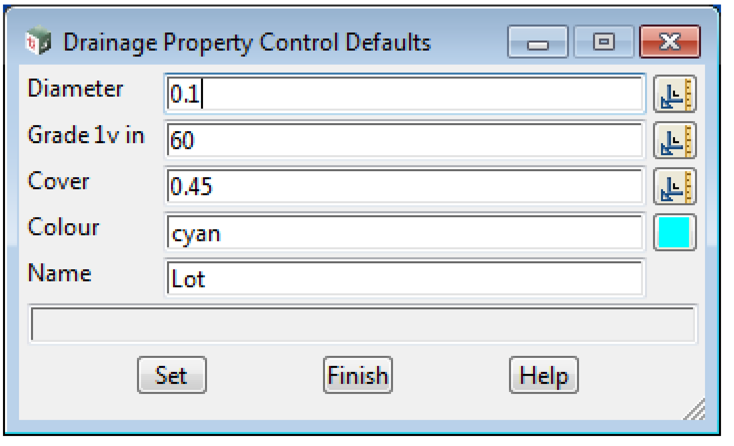
Minimum cover and minimum grade for the pipe segment to the end of the line can be set afterwards by the **pipe=>default grading** option in the drainage string editor.

22.4.4 Property Controls

Position of option on menu: **Design =>Drainage-Sewer =>Defaults =>Property Controls**

On selecting the **Property controls** option, the **Drainage Property Control Defaults** panel is displayed.

On a section view, the *Profile =>One substring* and *Profile =>Many substrings* options will profile the property control. Note that the centre (axis) of the property control is drawn on the section view, not the invert (bottom) or the obvert (top).



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Diameter	input	0.1	

default diameter of the property control.

Grade 1v in	input	60	

grade (units are "1v in" given value) to use for the property control

Cover	input	1.0	

*cover measured from the surface to the **top** of the property control (world units) to be maintained from the end of the property control in the house block to the drainage string.*

Colour	input	cyan	available colours

colour to use to draw the property control

Name	input	Lot	

name for the property control - usually the lot number

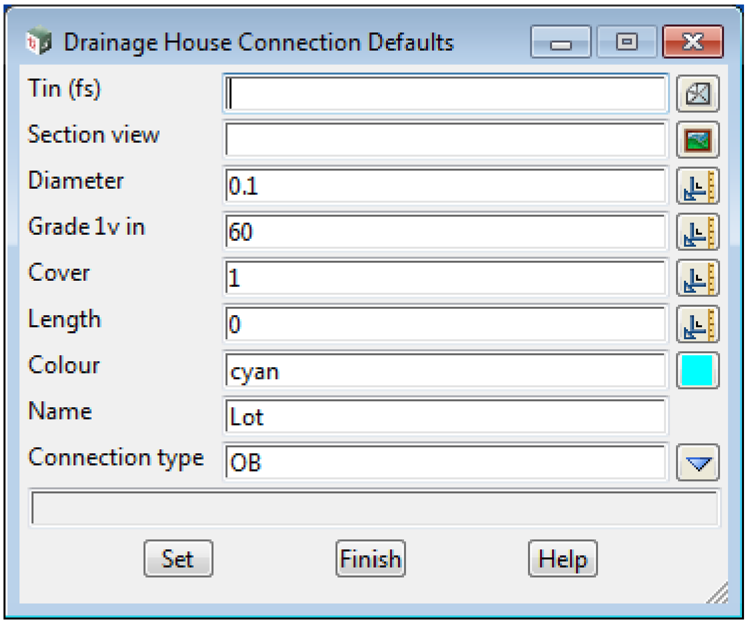
Set	button		

set the property control defaults to the values in the above fields.

22.4.5 House Connections

Position of option on menu: **Design =>Drainage-Sewer =>Defaults =>House connections**

On selecting the **House connections** option, the **Drainage House Connection Defaults** panel is displayed.



This panel is for setting the default information used for connections from the drainage pipe to the house blocks.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin (fs)	input	drainage tin	available tins

name of the tin to automatically add to the section view used for drawing the profile along the house connection.

Section view	input		

the section view used to profile along the house connections as they are placed in plan view.

Diameter	input	0.1	

default diameter of the house connection.

Grade 1v in	input	60	

grade (units are "1v in") to use for the house connection

Cover	input	1.0	

cover (world units) to use for the house connection

Length	input	2.0	

length (metres) to use for some types of house connections

Colour	input	cyan	available colours

default colour used for the house connection

Name	input	Lot	

name for the house connection - usually the lot number

Connection type	input	A special	A, A special, B, C, OB,

Special Jump Up

the default type of house connection, Please continue to the section House Connection Types *for a description of each connection type*

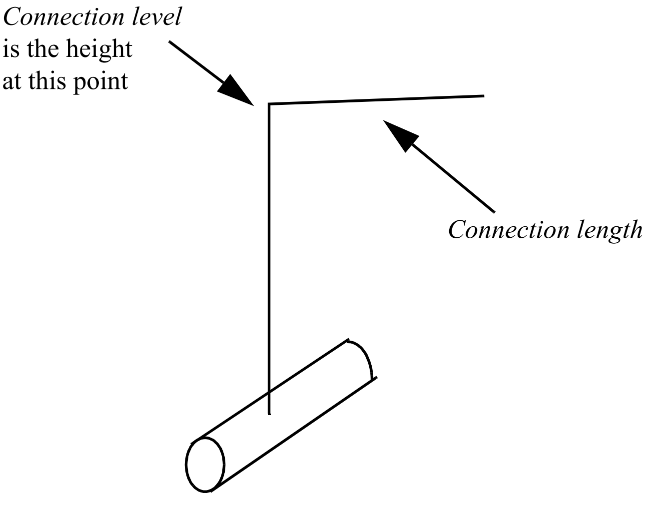
Set button

set the house connection defaults to the values in the above fields.

House Connection Types

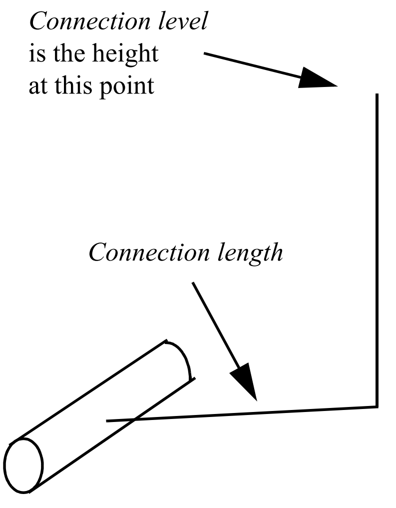
All house connection calculations do not take into account any thickness of pipe, joint sizes or actual entry points into the sewer. Hence they are ***approximate only*** and should only ever be used as a guide. Any quantities calculations should allow for a suitable margin of error.

House Connection -Type A



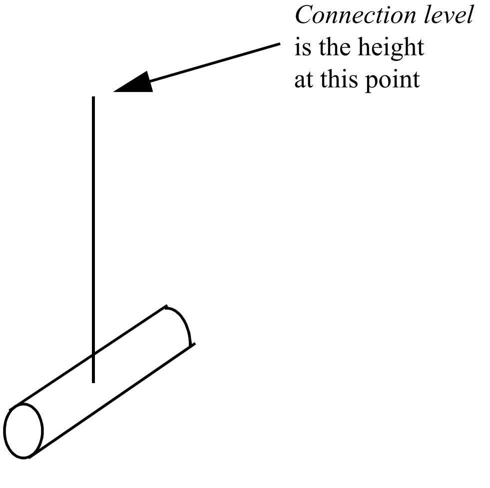
House Connection: Type A

House Connection - Type A Special



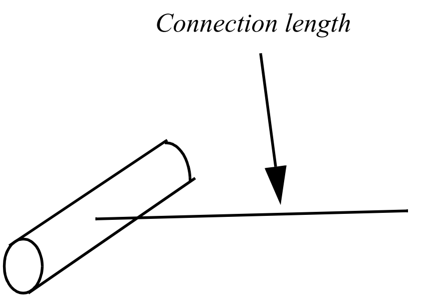
House Connection: Type A Special

House Connection - Type B



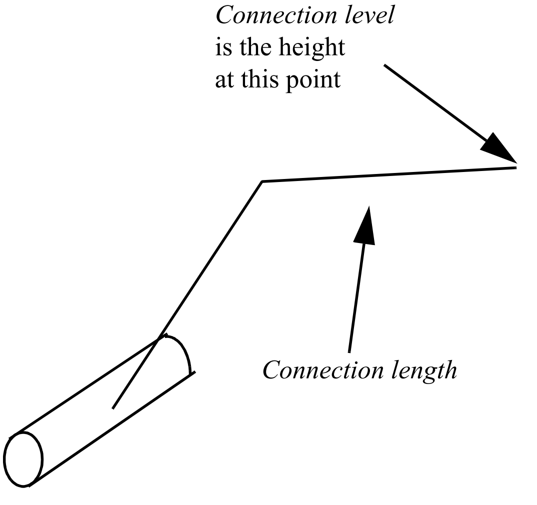
House Connection: Type B

House Connection - Type C



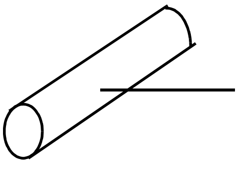
House Connection: Type C

House Connection - Type Special Jump Up



House Connection: Type Special Jump Up

House Connection - Type OB (Oblique)



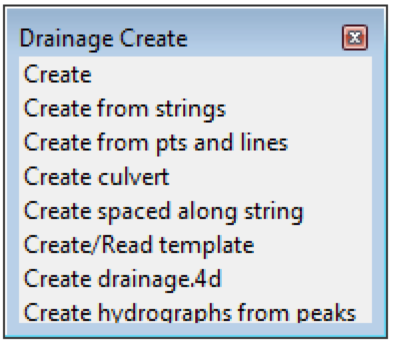
House Connection: Type OB

Please continue to the next section Create.

22.5 Create

Position of menu: **Design =>Drainage-Sewer =>Create**

The **Create** walk-right menu is



For *Create*, please see Create Drainage String.

Create from strings, please see Create from Strings.

Create from pts and lines, please see Create from Points and Lines.

Create culvert, please see Create Culvert Change.

Create spaced along string, please see Create Drainage Spaced Along String.

Create/Read template, please see Create/Read Drainage Template.

Create drainage.4d, please see Create drainage.4d.

Create hydrographs from peaks, please see Create Hydrographs from Peaks.

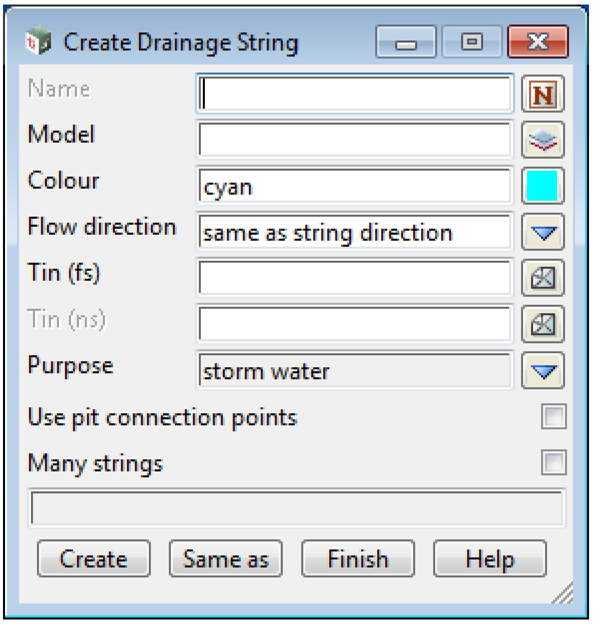
22.5.1 Create Drainage String

Position of option on menu: **Design =>Drainage-Sewer =>Create =>Create**

This section of documentation is a work in progress and will be updated in subsequent releases.

The **Create** option is used to produce **new** drainage strings and networks. If a drainage string already exists, the **Editor** option is used to modify it.

On selecting the **Create** option, the **Create Drainage String** panel is displayed.



To create a new drainage string, the name, colour, model and style of the new string are entered into the appropriate fields, plus the finished surface triangulation that the manholes are normally flush with, the natural surface triangulation, and the **Create** button selected.

The new fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	name box		
Model	model box		
Colour	colour box		
Flow direction	input	ascending chainage	ascending chainage descending chainage
if ascending chainage, then the flow direction is in ascending chainage (recommended) *if descending chainage, then the flow direction is in descending chainage.*			
Tin (fs)	input	drainage=>defaults=>tin	available tins
the finished surface tin. If manholes are "floating", the top of the manhole is automatically place on the tin surface ("floated" on the surface).			
Tin (ns)	input		available tins
the natural surface tin used in longsection plots.			
Purpose	choice box		
Use pit connection points	tick box		

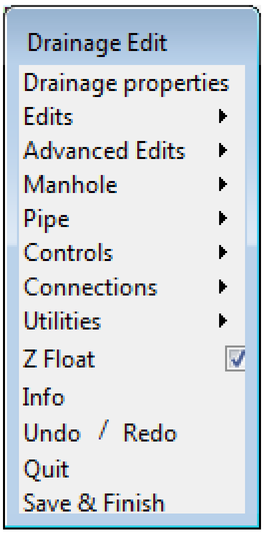
Many strings tick box

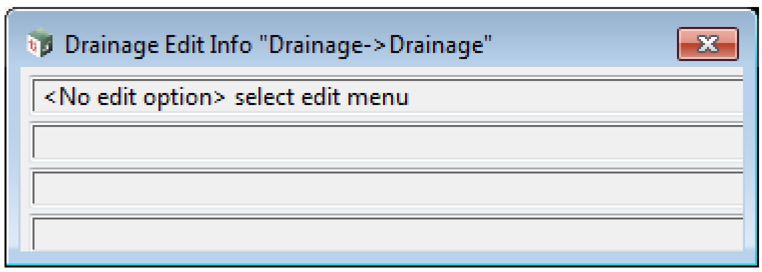
Create button

The **Create Drainage String** panel is then removed and the **Drainage Edit** menu and **Drainage Edit Info** panel fired up.

As for a 3d string, to create a new drainage string with some of the **same** name, colour, model and style as an existing string (not necessarily a drainage), the **Same as** button is chosen and the appropriate string selected.

The **Drainage Edit** menu contains all the available options for editing a drainage string and its associated block controls and house connections. The **Drainage Edit Info** panel contains information areas. The **Drainage Edit** menu and **Drainage Edit Info** panel are





To create a new drainage string, the user selects the **Append** option from the **Edits** walk right menu on the **Drainage Edit** menu.

Since the **Drainage Edit** menu and **Drainage Edit Info** panel are the same as those used when editing a drainage string, the options will be discussed under the drainage **Edit** option.

For documentation on editing Drainage string, please continue to the next section String Editor.

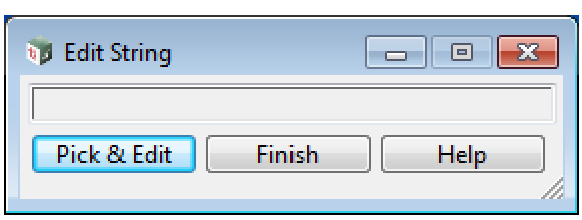
For drainage utilities and import/export see More Drainage

22.6 String Editor

Position of option on menu: **Design =>Drainage-Sewer =>String editor**

This is the same option as **Editor** from the **Strings** walk-right menu on the 12d Model menu.

The string editor is used to modify any 12d Model strings. After selecting the **Editor** option, the **Edit String** panel is placed on the screen to record any error messages.



The option is already in the **pick** mode (the **pick & edit** button only needs to be selected if the pick was cancelled) and the user simply picks and accepts the string to be edited.

From the picked string's type, the editor is able to determine the edits that apply to the string and list them in the string's **edit** menu.

If a drainage string is selected, the **Drainage Edit** menu and **Drainage Edit** panel (as shown in the previous section) are placed on the screen.

The individual edit operations for a drainage string will now be discussed in detail.

Please continue to the next section Drainage Edit.

22.6.1 Create from Strings

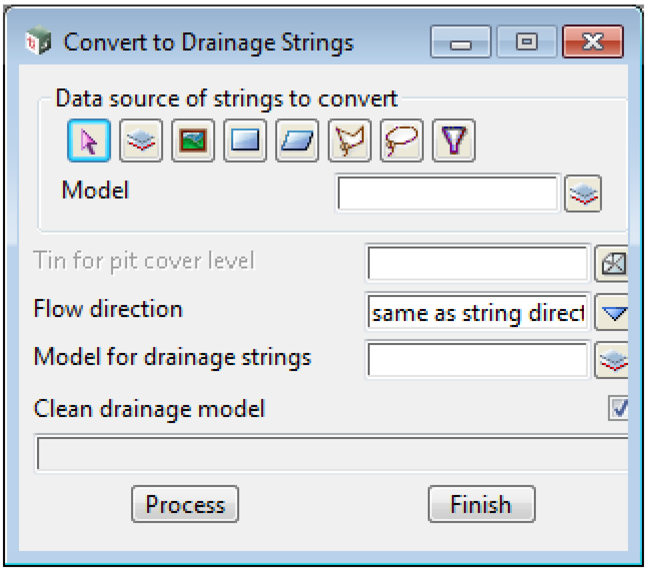
Position of option on menu: **Design =>Drainage-Sewer=>Create=>Create from strings**

12d will convert the super strings into 12d drainage strings. The default pipe, pit and tin data will be used to set the levels for the network. Do not use the other string convert commands found on the menu system.

See Also

Drainage overview

On selecting the **Create from strings** option, the **Convert to Drainage Strings** panel is displayed.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Data source	source box	model	

data source for strings to be converted

Model for drainage strings model box

The new drainage strings will be added to this model. If it does not exist it will be created.

Tin for pit cover level tin box

The new drainage manhole cover level will be set to the tin level at the manhole centre (optional)

Flow direction choice box same as string direction

same as string direction *if the strings have been drawn in the direction of water flow*
opposite to string direction *if the strings are drawn opposite to the direction of flow.*

Clean drainage model tick box

When selected, all strings in the model will be deleted before creating the new strings.

Process button

Coverts the strings to the drainage strings.

Finish button

Removes the panel from the screen.

Important notes:

The imported strings must all be drawn in the same direction. Either all in the direction the water flows or all opposite the direction of flow..

Pits are created at all vertices on the strings.

Trunk lines must have a vertex where the branch lines join.

String names can be used to control the order in when the drainage lines are numbered. These names will be transferred to the 12d drainage strings. Later, the string names can be changed in the Drainage Network Editor.

Pits can always be renamed in 12d after the import is complete.

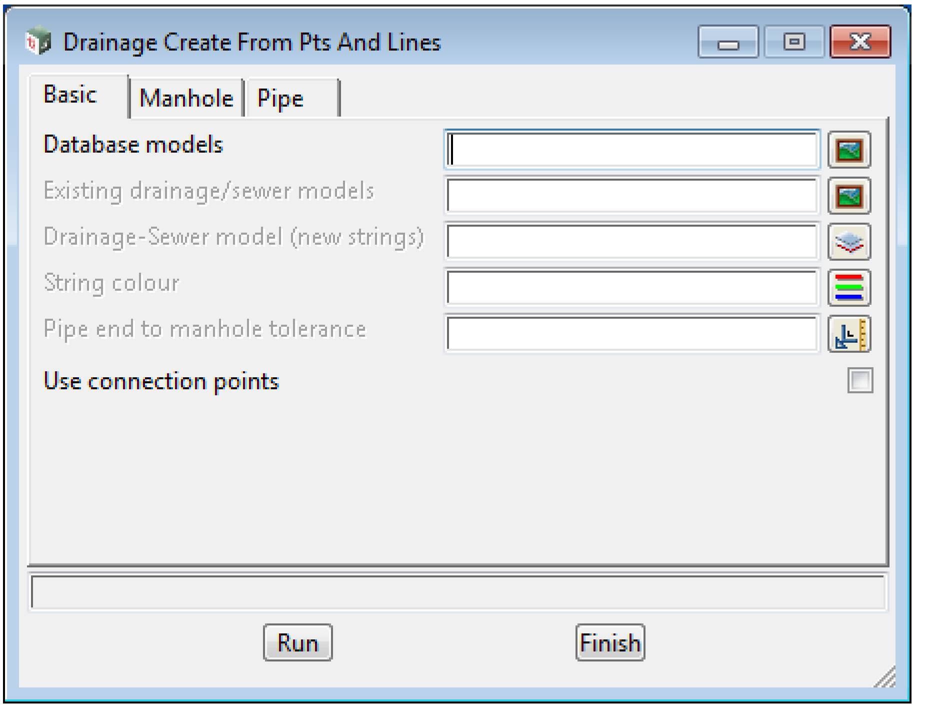
The drainage lines must have string names to use the Set Pit Names feature on the network editor.

22.6.2 Create from Points and Lines

Position of option on menu: **Design =>Drainage-Sewer=>Create=>Create from pts and lines**

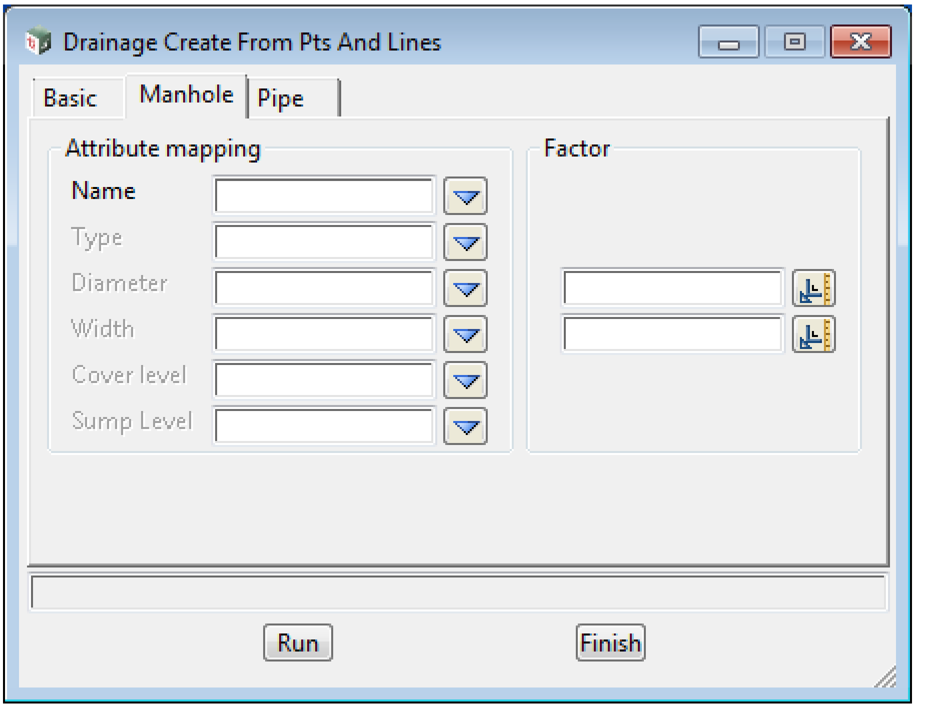
This section of documentation is a work in progress and will be updated in subsequent releases.

On selecting the **Create from pts and lines** option, the **Drainage Create from Ptd and Lines** panel is displayed.



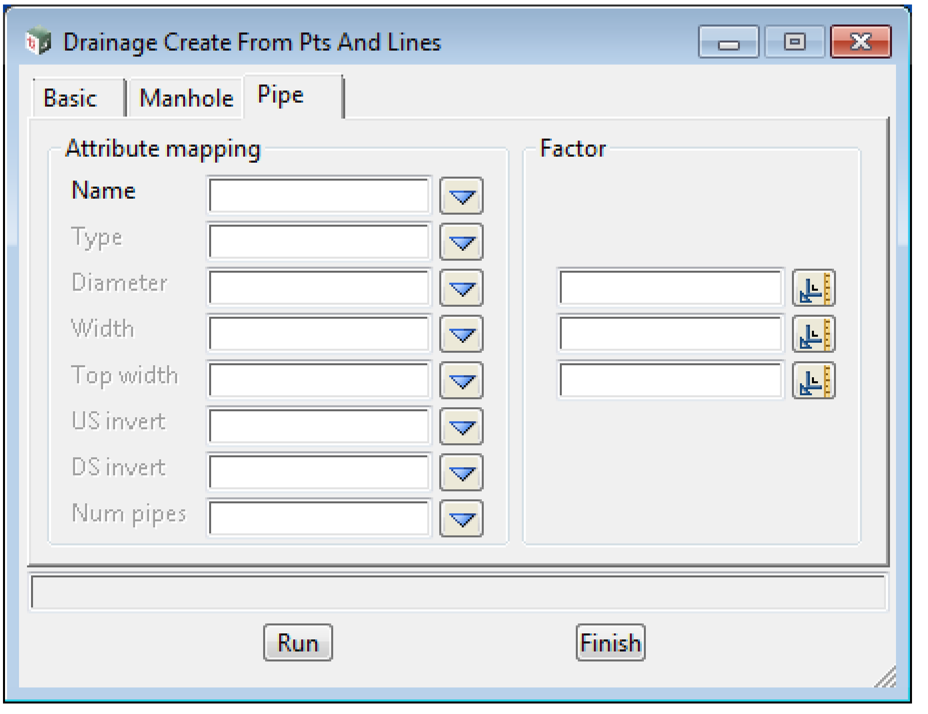
The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Basic Tab			
Database models	view box		
Existing drainage/sewer models	view box		
Drainage-Sewer model (new strings)	model box		
String colour	colour box		
Pipe end to manhole tolerance	measure box		
Use connection points	tick box		
Manhole Tab			



Name	choice box
Type	choice box
Diameter	choice box
Width	choice box
Cover level	choice box
Sump level	choice box
Factor	measure boxes

Pipe Tab



Name	choice box
Type	choice box
Diameter	choice box
Width	choice box
Top width	choice box
US invert	choice box
DS invert	measure boxes
Num pipes	measure boxes
Factor	measure boxes

Buttons at Bottom

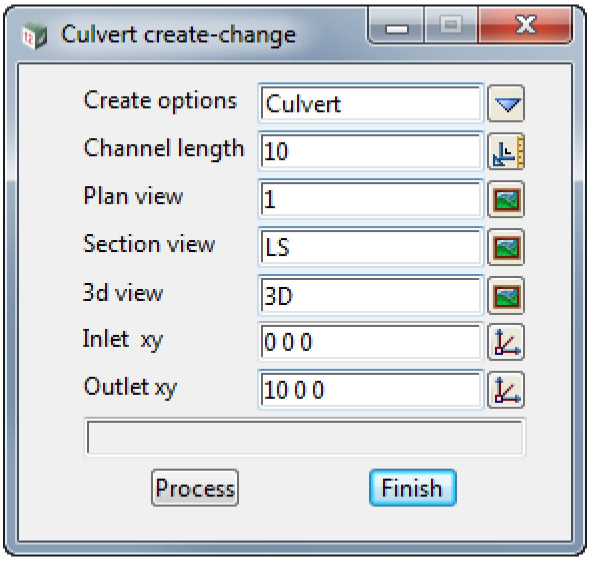
Run	button

22.6.3 Create Culvert Change

Position of option on menu: **Design =>Drainage-Sewer=>Create=>Create culvert**

This section of documentation is a work in progress and will be updated in subsequent releases.

On selecting the **Create culvert** option, the **Culvert Create-Change** panel is displayed.



The fields and buttons used in this panel have the following functions:

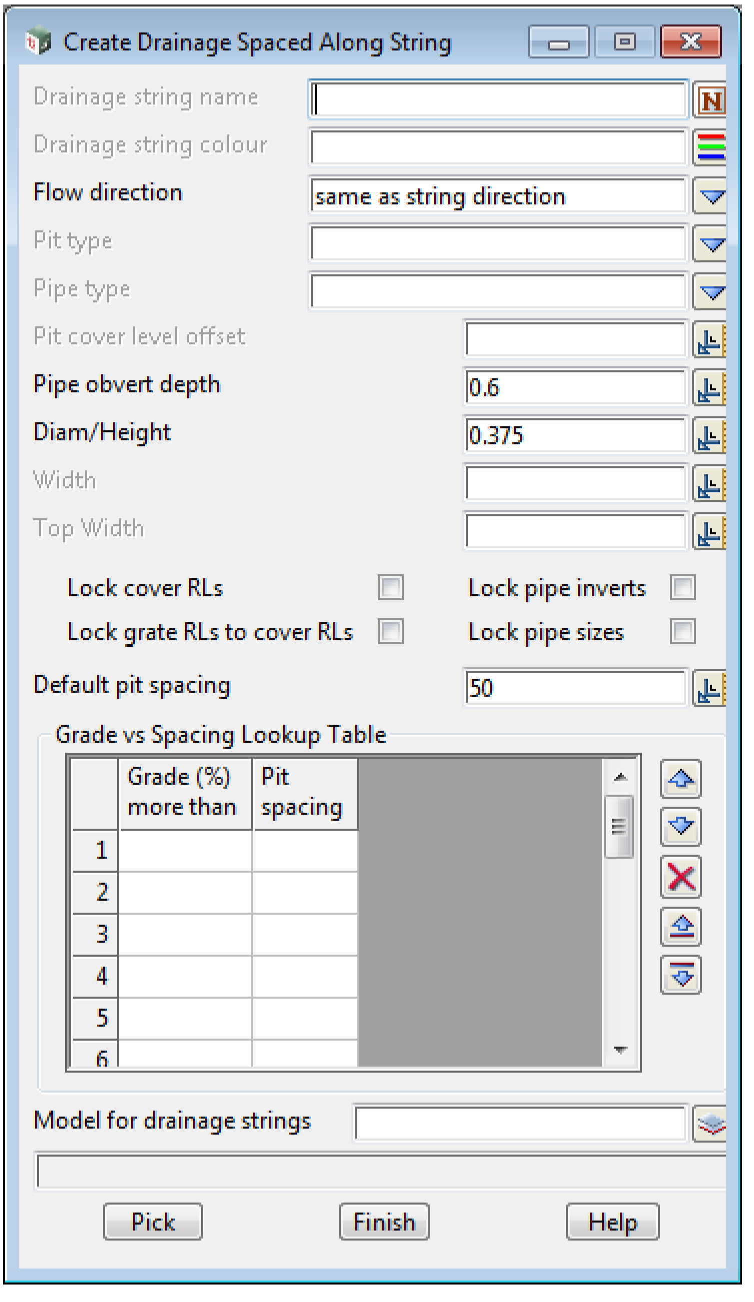
Field Description	Type	Defaults	Pop-Up
Create options	choice box	Culvert	
Channel length	measure box	10	
Plan view	view box	1	
Section view	view box	LS	
3d view	view box	3D	
Inlet xy	xy box	0 0 0	
Outlet xy	xy box	10 0 0	
Process	button		

22.6.4 Create Drainage Spaced Along String

Position of option on menu: **Design =>Drainage-Sewer=>Create=>Create spaced along string**

This section of documentation is a work in progress and will be updated in subsequent releases.

On selecting the **Create spaced along string** option, the **Create Drainage Spaced Along String** panel is displayed.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Drainage string name	name box		
Drainage string colour	colour box		
Flow direction	choice box		

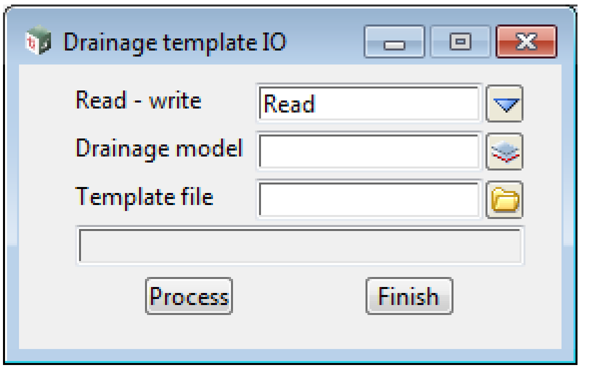
Pit type	choice box
Pipe type	choice box
Pit cover level offset	measure box
Pipe obvert depth	measure box
Diam/Height	measure box
Width	measure box
Top Width	measure box
Lock cover RLs	tick box
Lock pipe inverts	tick box
Lock grate RLs to cover RLs	tick box
Lock pipe sizes	tick box
Default pit spacing	measure box
Grade (%) more than	grid
Pit spacing	grid
Model for drainage strings	model box
Pick	button

22.6.5 Create/Read Drainage Template

Position of option on menu: **Design =>Drainage-Sewer=>Create=>Create/Read template**

This section of documentation is a work in progress and will be updated in subsequent releases.

On selecting the **Create/Read template** option, the **Drainage Template I/O** panel is displayed.



The fields and buttons used in this panel have the following functions:

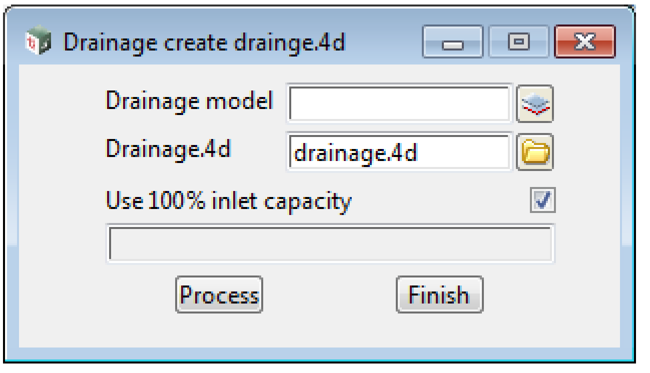
Field Description	Type	Defaults	Pop-Up
Read-write	choice box		
Drainage model	model box		
Template file	file box		
Process	button		

22.6.6 Create drainage.4d

Position of option on menu: **Design =>Drainage-Sewer=>Create=>Create drainage.4d**

This section of documentation is a work in progress and will be updated in subsequent releases.

On selecting the **Create drainage.4d** option, the **Drainage Create Drainage.4d** panel is displayed.



The fields and buttons used in this panel have the following functions:

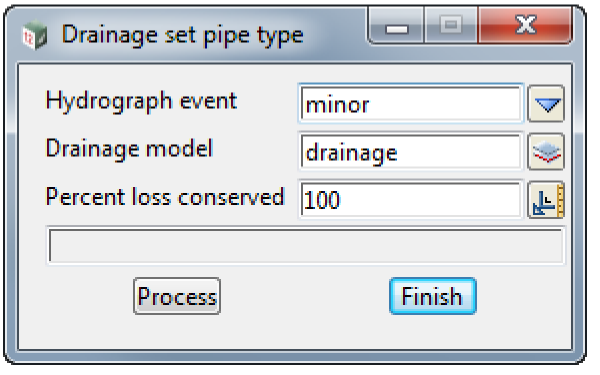
Field Description	Type	Defaults	Pop-Up
Drainage model	model box		
Drainage.4d	file box		
Use 100% inlet capacity	tick box		
Process	button		

22.6.7 Create Hydrographs from Peaks

Position of option on menu: **Design =>Drainage-Sewer=>Create=>Create hydrographs from peaks**

This section of documentation is a work in progress and will be updated in subsequent releases.

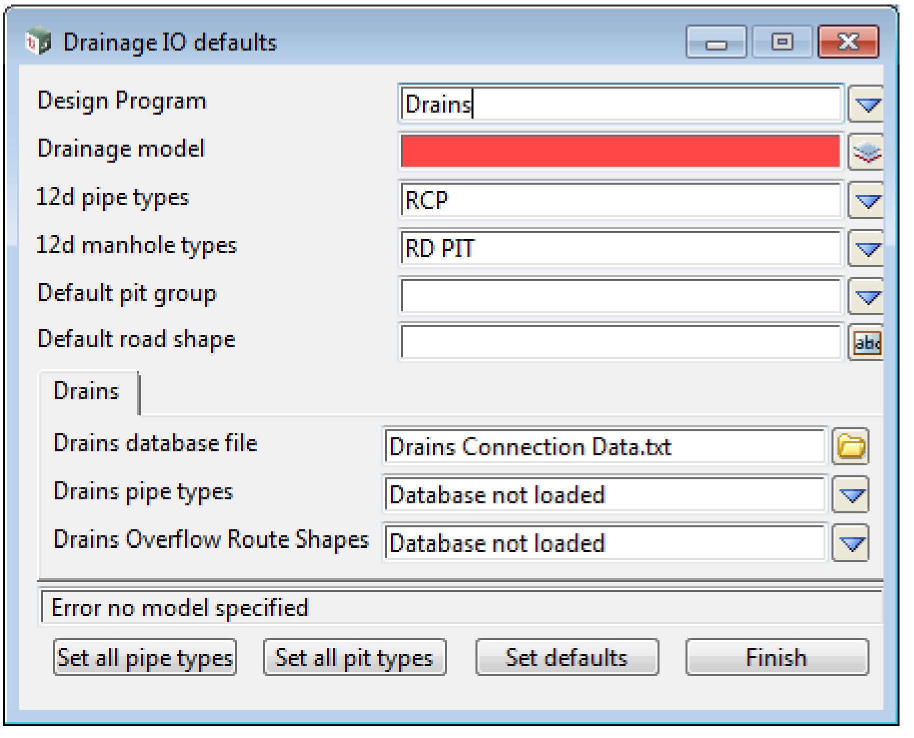
On selecting the **Create hydrographs from peaks** option, the **Drainage Set Pipe Type** panel is displayed.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Hydrograph event	choice box		
Drainage model	model box		
Percent loss conserved	measure box		
Process	button		

22.6.8 Drainage IO Defaults



22.7 Drainage Network Editor

Position of option on menu: Design =>Drainage-Sewer =>Network editor

The drainage network editor enables the user to edit all of the drainage strings in a model (a network). The global and default settings may be stored and loaded via . There are five main tabs and plus function buttons on the bottom of the panel.

Using the Drainage Network Editor

Recommended order to use the tabs and Buttons.

(This is not required but it will lead you through the process in an systematic method.)

Tab	Function Buttons
1. Global tab	3. Set Pit Names
2. Defaults tab	4. Set Catchments
5. Pipe tab	8. Set Pit Details
6. Pit tab	9. Regrade Network
7. Catchment	Storm Analysis
	Plot
	Import/Export

For information on specific tasks see the following sections,

Manhole Setout Settings

Road Design File for Pit Setout - x,y, level, road chainage and setout offset

Hydrology Settings

Catchment Areas

Checking the Automatic Catchment Linking

Coefficients of Runoff

Percent Impervious

Times of Concentration Methods

Tc Path Strings

Hydraulic Design Settings

Pipe Friction Method

Pipe Friction Values and Freeboard Limit

Setout to Grate Offset

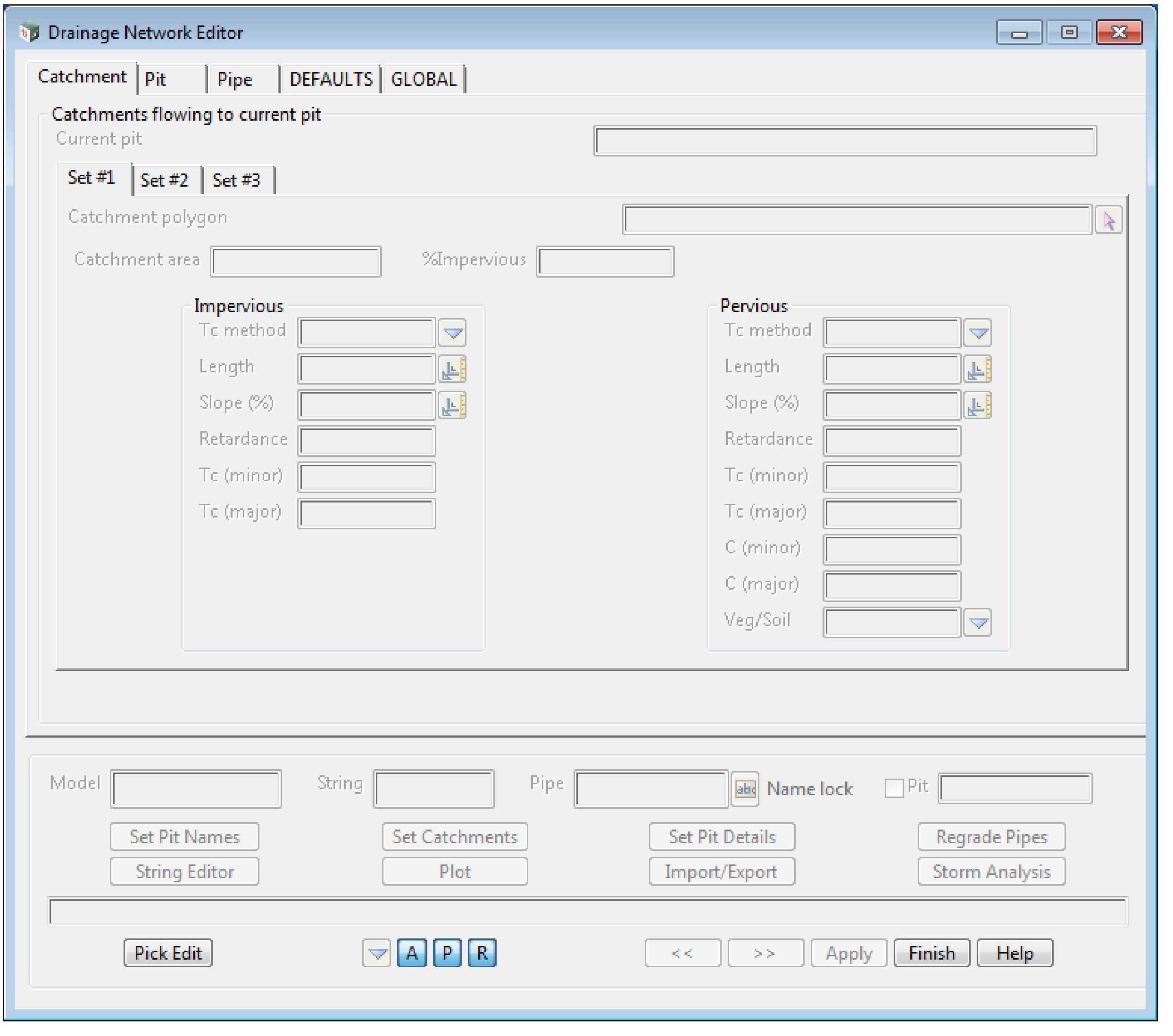
Pit Losses Ku, and Direct Flow

Outlet and Tailwater Conditions

Pipe Design Parameters

Bypass Flow Settings

On selecting the Drainage network editor option, the Drainage network editor panel is displayed.


Drainage Network Editor
Catchment
Pit
Pipe
DEFAULTS
GLOBAL
Catchments flowing to current pit
Current pit
Set #1
Set #2
Set #3
Catchment polygon
Catchment area
%Impervious
Impervious
Tc method
Length
Slope (%)
Retardance
Tc (minor)
Tc (major)
Pervious
Tc method
Length
Slope (%)
Retardance
Tc (minor)
Tc (major)
C (minor)
C (major)
Veg/Soil
Model
String
Pipe
Name lock
Pit
Set Pit Names
Set Catchments
Set Pit Details
Regrade Pipes
String Editor
Plot
Import/Export
Storm Analysis
Pick Edit
A
P
R
<<
>>
Apply
Finish
Help

22.7.1 Global Tab

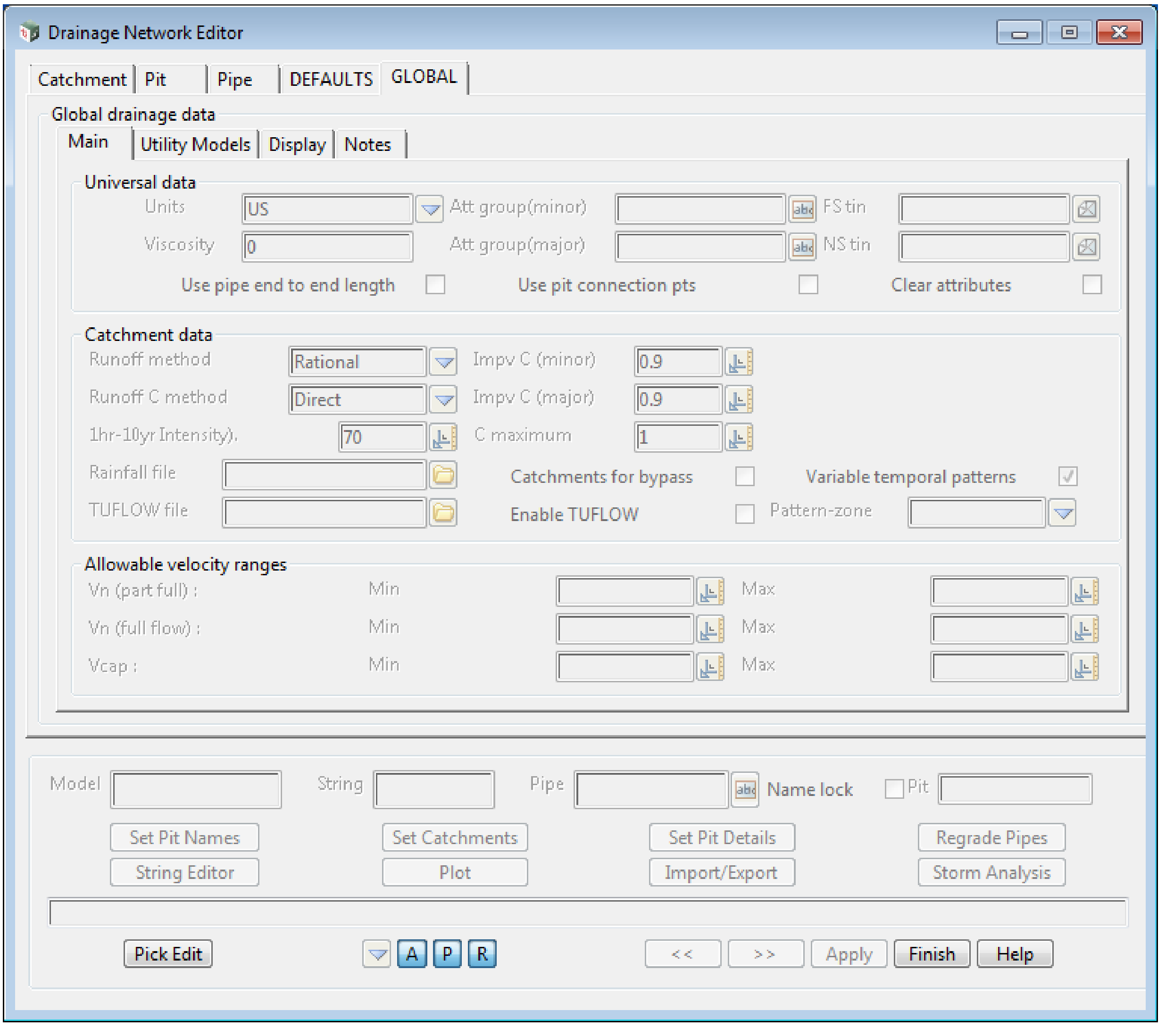
Design values for the hydrology and hydraulics calculations are set either globally (one value for the entire network) or via Defaults for the manholes or pipes. Defaults values may be overridden by explicit settings found on the catchment, pits or pipes tab. Explicit manhole/pipe settings need only be specified if the default value is not desired.ragg

There are 4 sub tabs. See Main tab, Utility Models tab, Notes tab **and** Display tab

22.7.1.1 Main tab

Universal data Group

Units - Metric or US. Catchment areas in ha/acres and rainfall-infiltration-storage in mm/in
Viscosity - Used with Colebrook-White energy loss calculations
TUFLOW file - with the dynamic drainage analysis and TUFLOW modules this specifies the TUFLOW tcf file to be used for the 2d storm analysis.
Att group (minor) - results from the storm analysis (minor event) may be contained within this pit/pipe attribute group
Att group (major) - results from the storm analysis (major event) may be contained within this pit/pipe attribute group
Use pit connection pts - when not selected, pipes connect at the centre of the manhole. When selected the pipes connect at the pit connection points. See Pit Connection Points for more details.
Finished surface tin - used for determining pipe cover and surface levels for the manholes.
Natural surface tin - specified so that it can be included on the drainage longsection plots
Clear attributes - when selected all pit/pipe attributes will be deleted before the Apply/AutoApply saves the data currently in the panel.



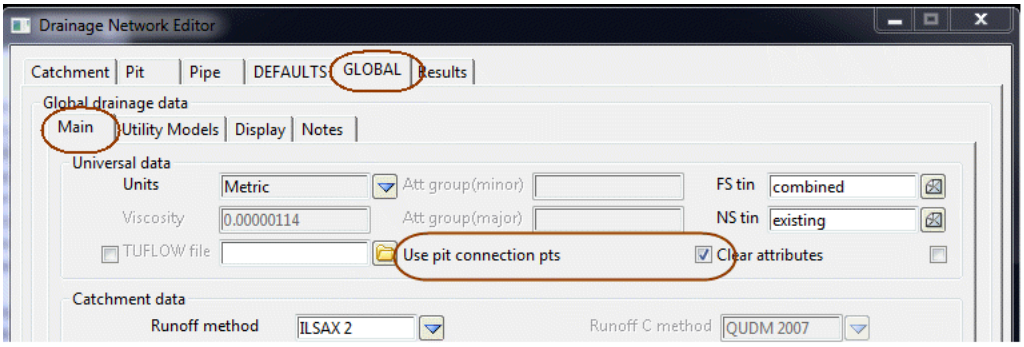
Catchment data Group

see Coefficients of Runoff

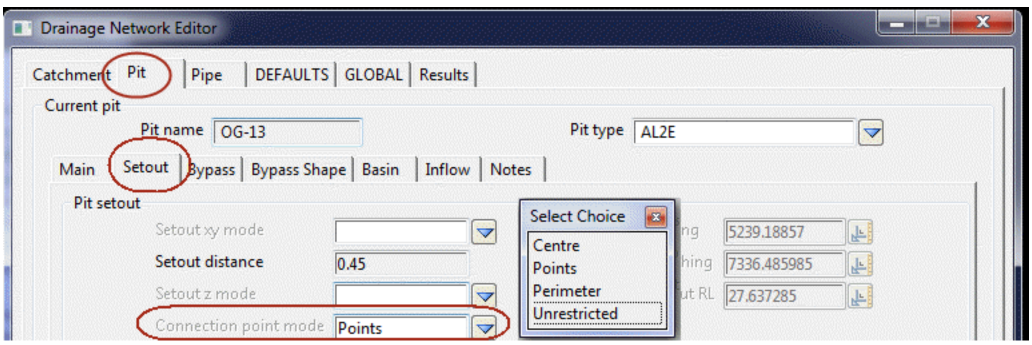
For **Global pipe data** see Pipe Friction Method.

22.7.2 Pit Connection Points

Pit Connection points allow the pipes connected to the pit to join at locations other than the inside perimeter of the pit wall. This feature is off by default and may be enabled on the **DNE->Global->Main** tab.



All pits have their connection point initially set to **Points**. This setting is found on the **DNE->Pits->Setout** tab.



Pit Connection Point Modes

Pit connection points may be moved via **Strings->Points Edit->Move** (except for **Centre** mode described below). The other modes constrain the movement of the connection points. If a pipe is manually moved to a new connection point, it will be locked to the connection point and will not move if the pit or neighbouring pit is moved (see exceptions below).

Centre (rectangle and circular) - This mode is the same as having the **Use connection points** turned off**.**The connection points will be located on the inside perimeter of the pit wall with the centre line of the pipe intersecting the centre of the pit. In this mode the connection points may not be adjusted.

Points (rectangle) - A connection point is created at the mid point of each internal side of the pit. This may be changed for a **Pit type** by using the **con_points** command in the drainage.4d file. In this mode the pipe ends will snap to the connection points. It is possible to place more than one pipe on the same connection point (the elevation of the pipes is not checked for clashes).

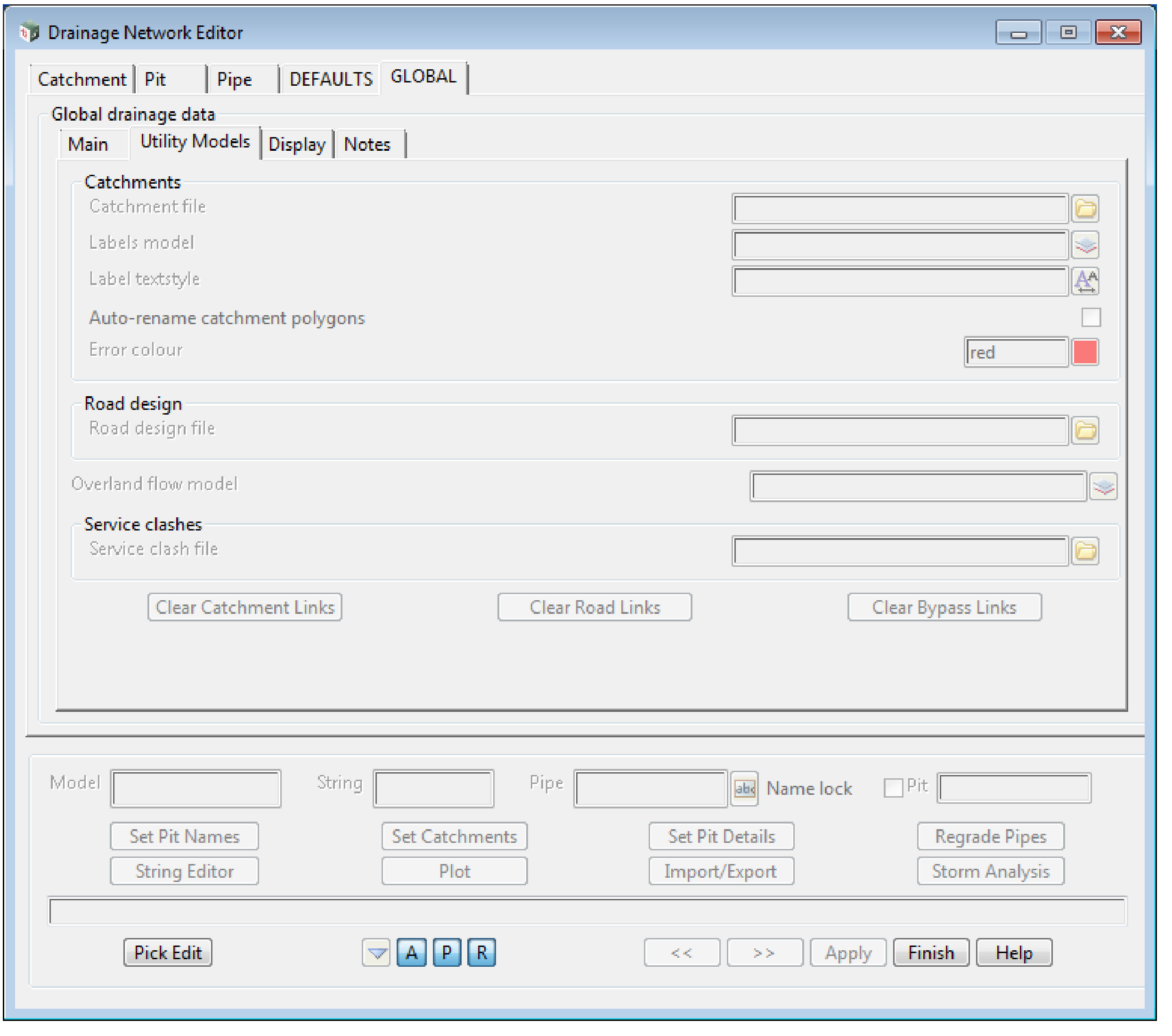
Points (circular) - The connection point may be moved anywhere around the pit internal wall as there are no connection points on the circular pits. Again, it is possible to place more than one pipe on the same connection location (the elevation of the pipes is not checked for clashes).
If the manhole centre is moved the connection point locks are removed.

Perimeter (rectangular and circular) - Same as **Points** (circular) above.

Unrestricted (rectangular and circular) - There are no constraints on the location of the pit connection points.This mode is intended for irregular shapes such as GPT structures and stormwater basins.

22.7.2.1 Utility Models tab

The utility models tab is used to specify the following data:



Catchment polygons, the catchment slope strings, catchment labelling data

Bypass flow strings

Road setout strings., road centre line strings and crossfall/grade offsets

Service/utility models and the allowable clearances

Additional details

Catchments Areas

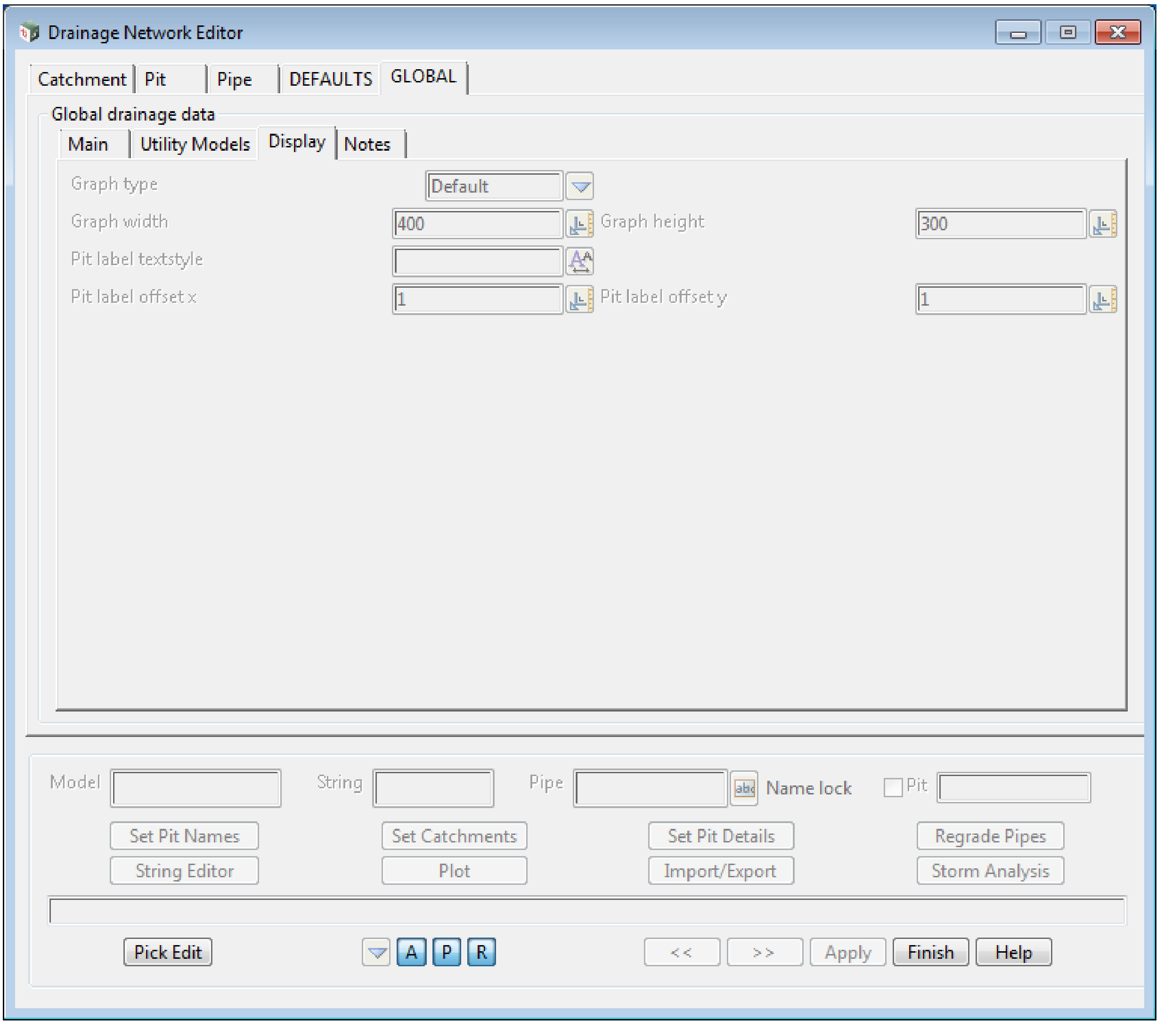
Bypass flow

Manhole setout via setout strings

Service and Utility Clashes

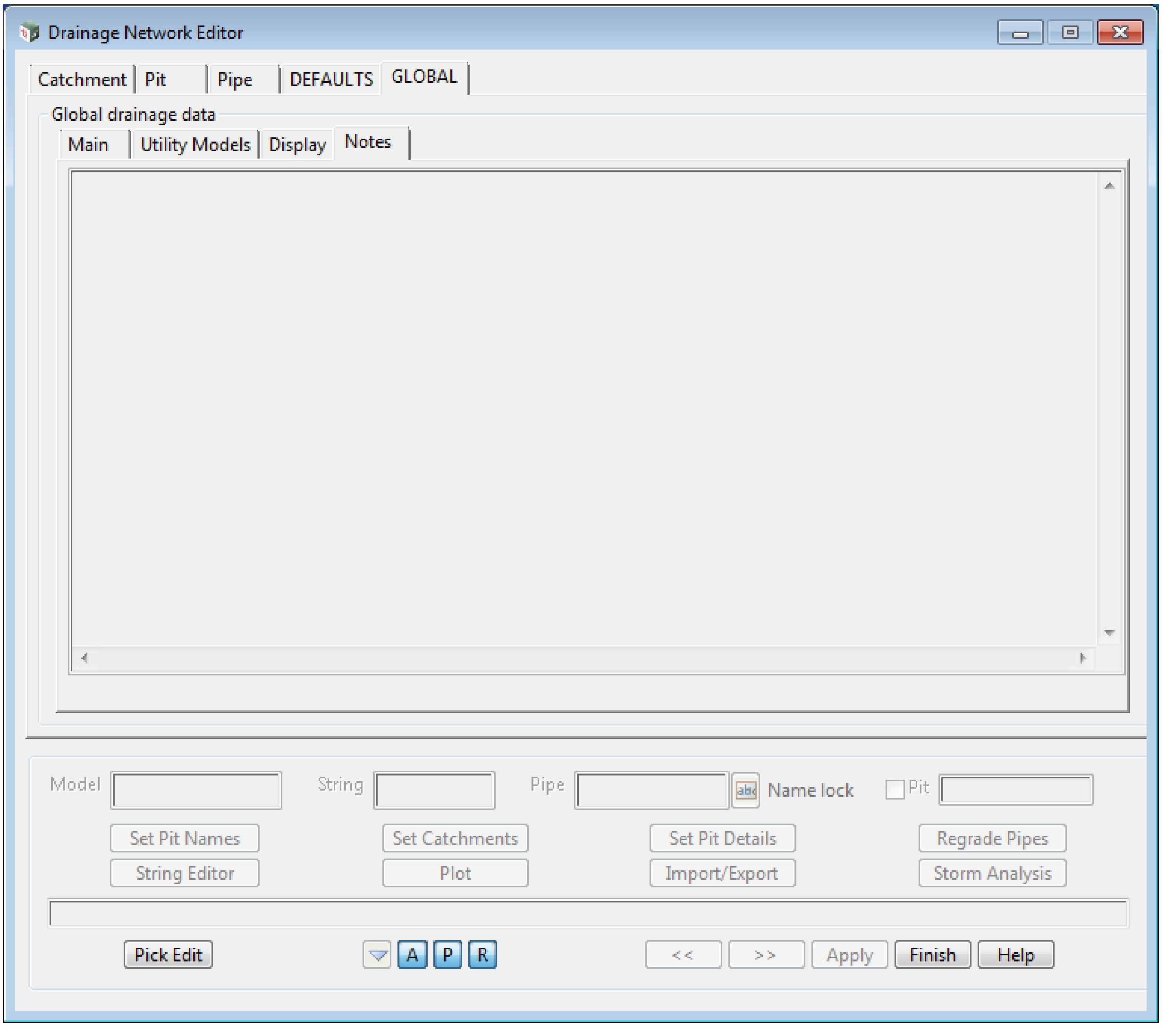
22.7.2.2 Display tab

The display tab is used to specify the following data:



22.7.2.3 Notes tab

The notes tab is used to specify the following data:

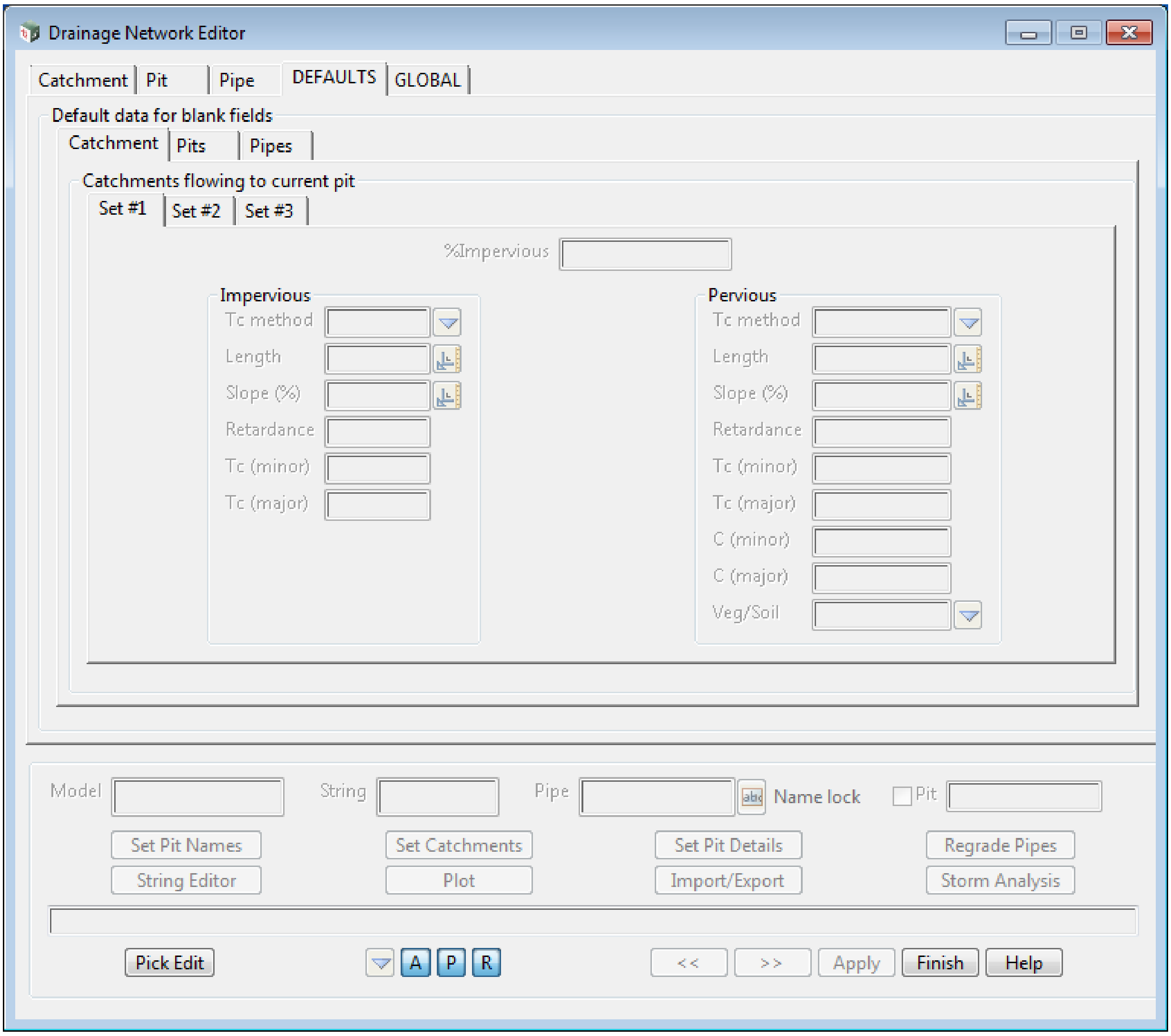

Drainage Network Editor
Catchment
Pit
Pipe
DEFAULTS
GLOBAL
Global drainage data
Main
Utility Models
Display
Notes
Model
String
Pipe
Name lock
Pit
Set Pit Names
Set Catchments
Set Pit Details
Regrade Pipes
String Editor
Plot
Import/Export
Storm Analysis
Pick Edit
A
P
R
<<
>>
Apply
Finish
Help

22.7.3 Defaults tab

The values on the defaults tabs are used unless the values are explicitly set on the **Main** Catchment, Pit and Pipe tabs.

22.7.3.1 Catchment subtab

The Catchment subtab is used to specify the following data:



Catchment Areas

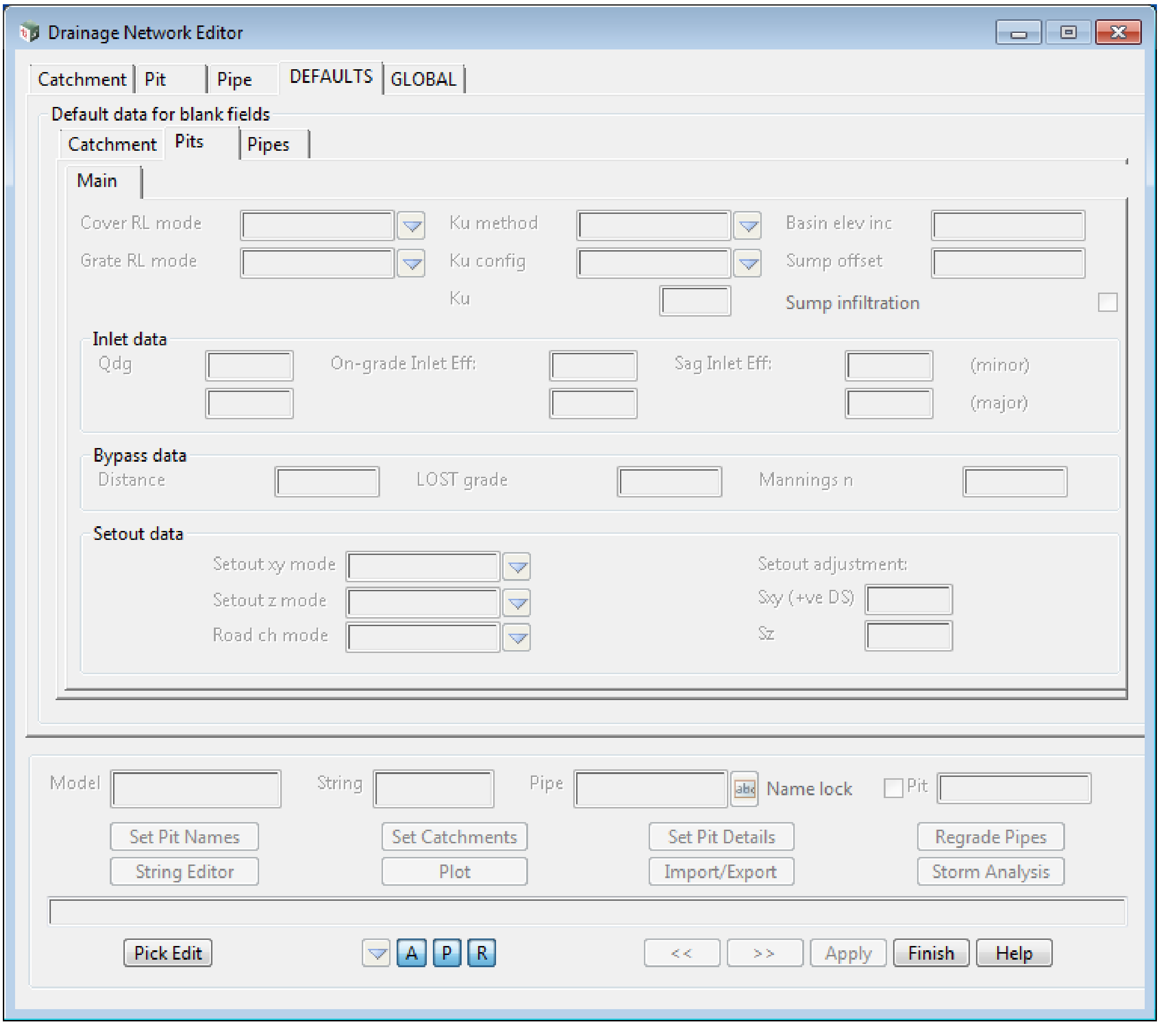
Percent Impervious

Times of Concentration Methods

Coefficients of Runoff

22.7.3.2 Pits subtab

The Pits subtab is used to specify the following data:



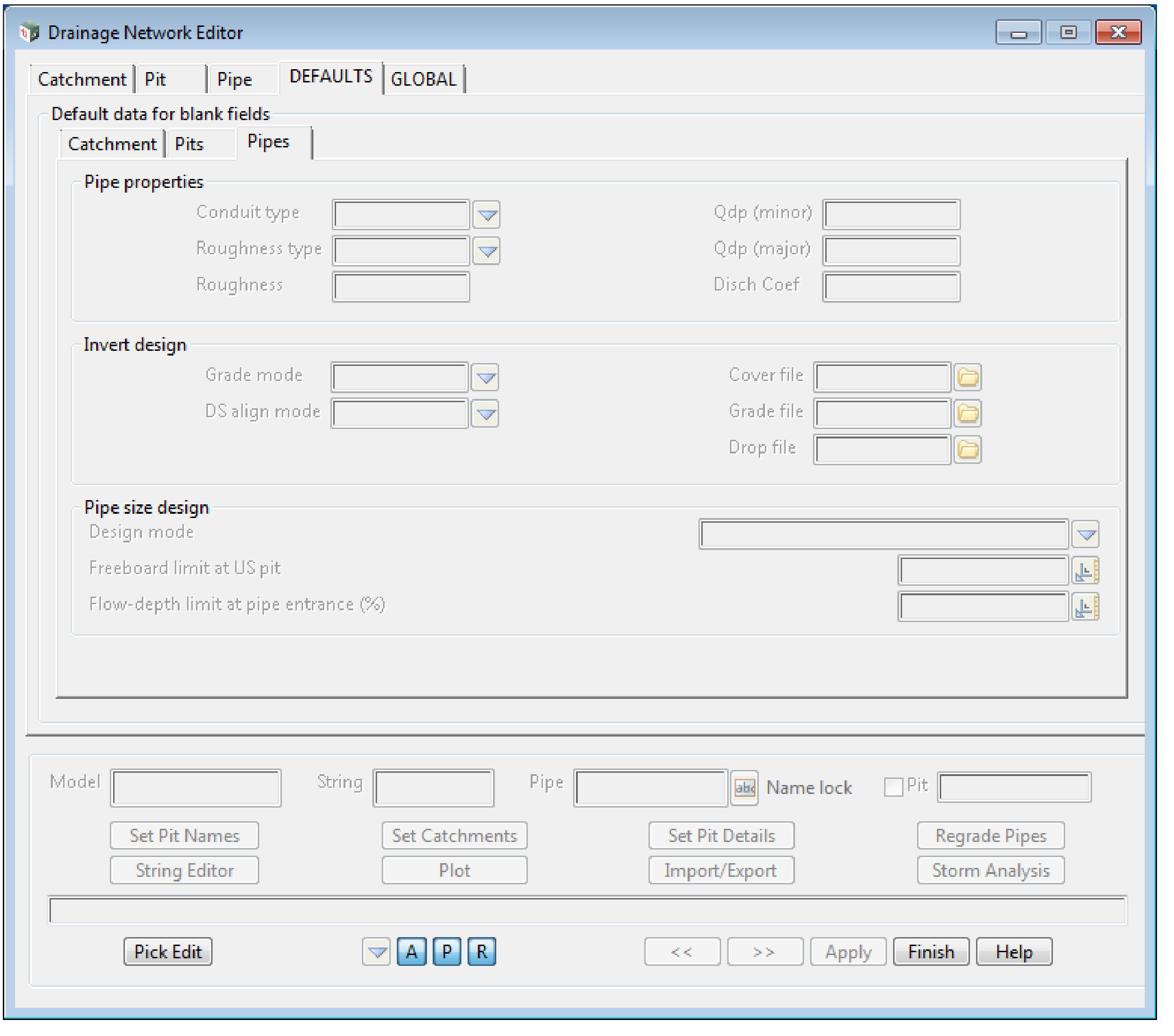
Manhole Setout Settings

Pit Losses Ku, and Direct Flow

Setout to Grate Offset

22.7.3.3 Pipes subtab

The Pipes subtab is used to specify the following data:



Pipe Friction Method

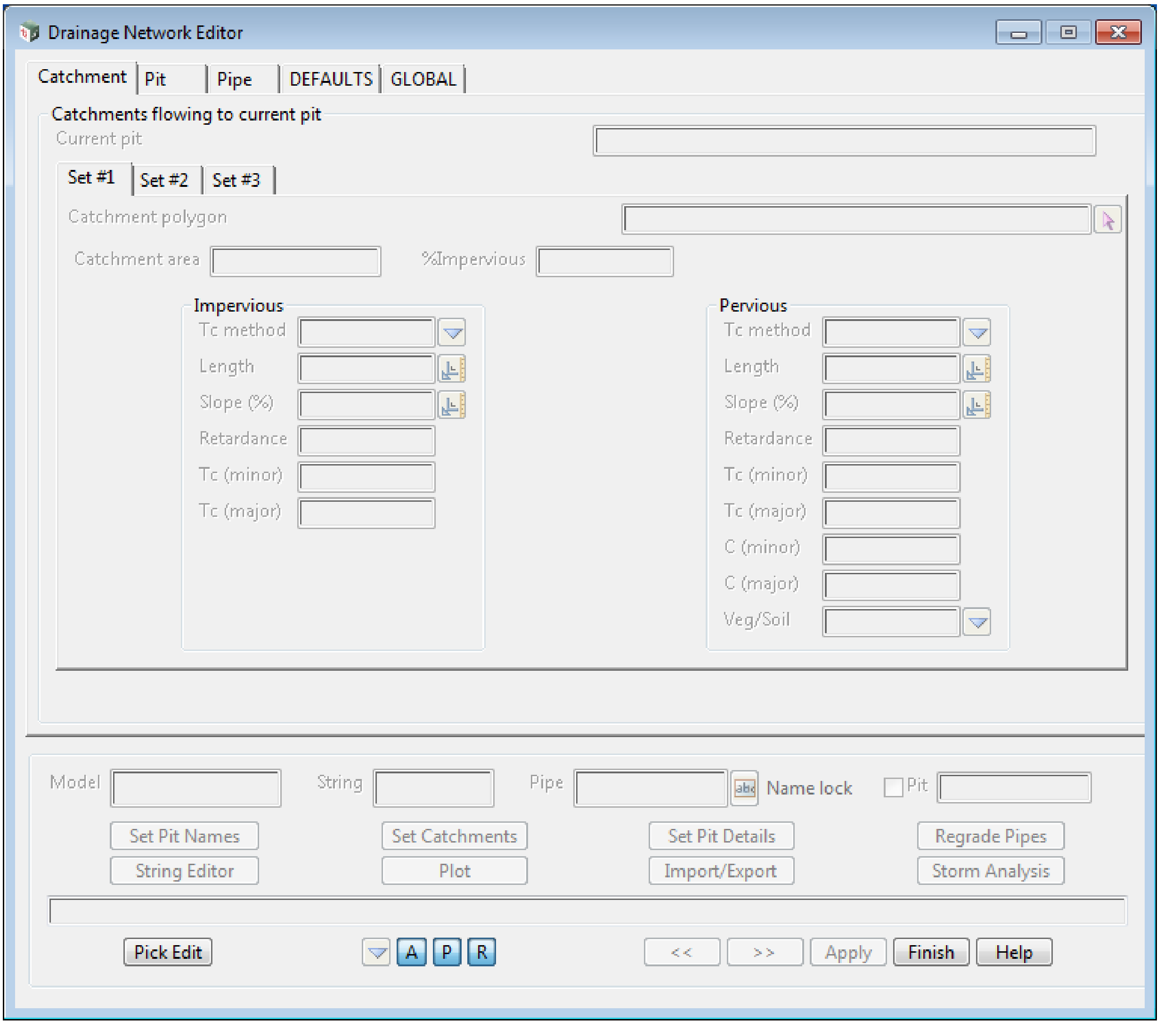
Pipe Friction Values and Freeboard Limit

Pit Losses Ku, and Direct Flow

Pipe Design Parameters

22.7.4 Catchment tab

The Catchment tab is used to specify the following data:



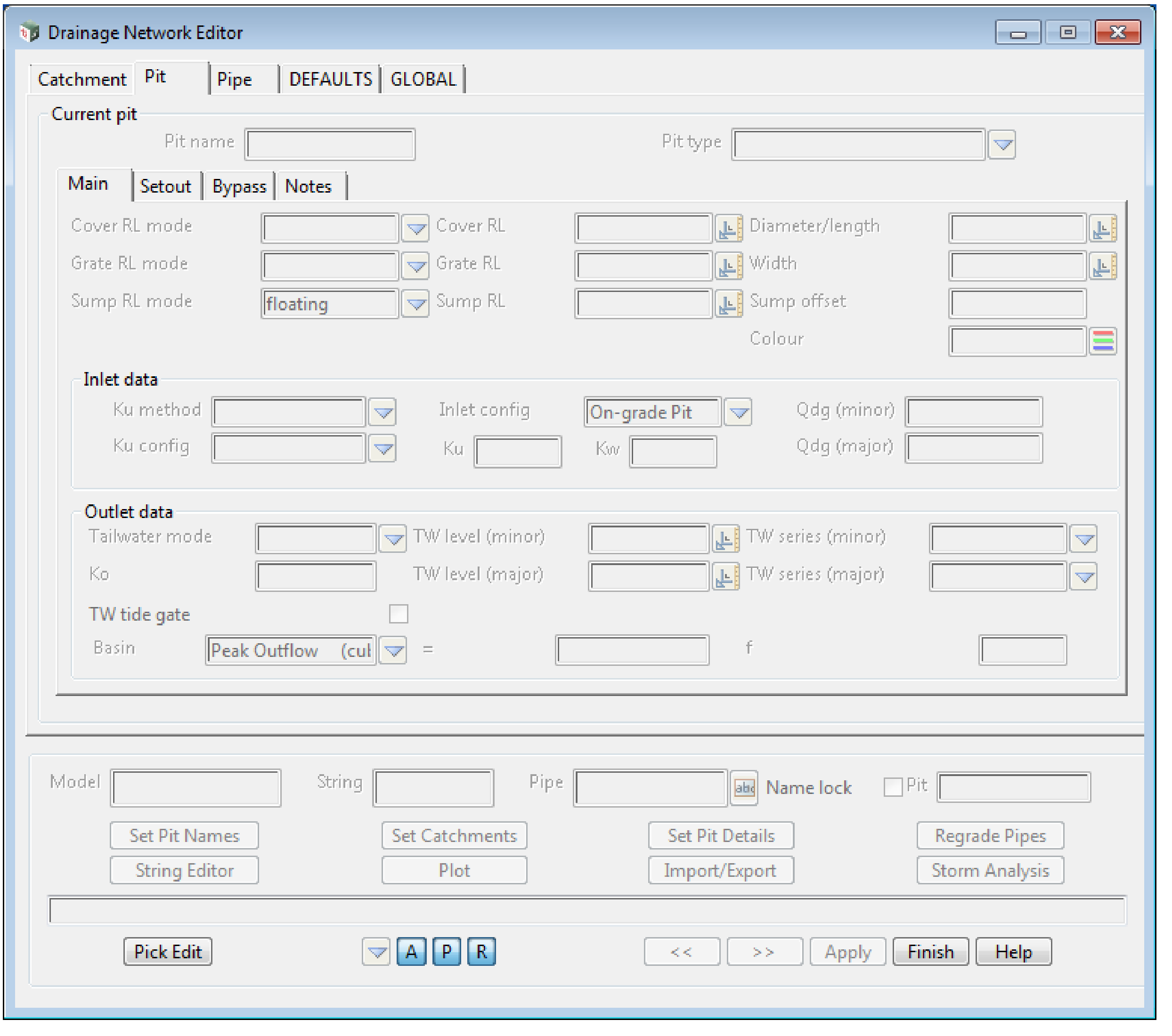
The pit tab has the explicit settings for the values entered on the Defaults catchment tab. When these fields are blank the default value is used. Three catchment sets are available. Typically the user will use one set for different types of catchment areas. For example: Set 1-Road area, Set 2-Lots, Set 3-Parks.

All 3 sets use the same default values except the percent impervious which has a separate default for each set.

Catchment Areas

22.7.5 Pit tab

The Pit tab is used to specify the following data:



The pit tab has the explicit settings for the values entered on the Defaults pit tab. When these fields are blank the default value is used.

Manhole Setout Settings

Pit Losses Ku, and Direct Flow

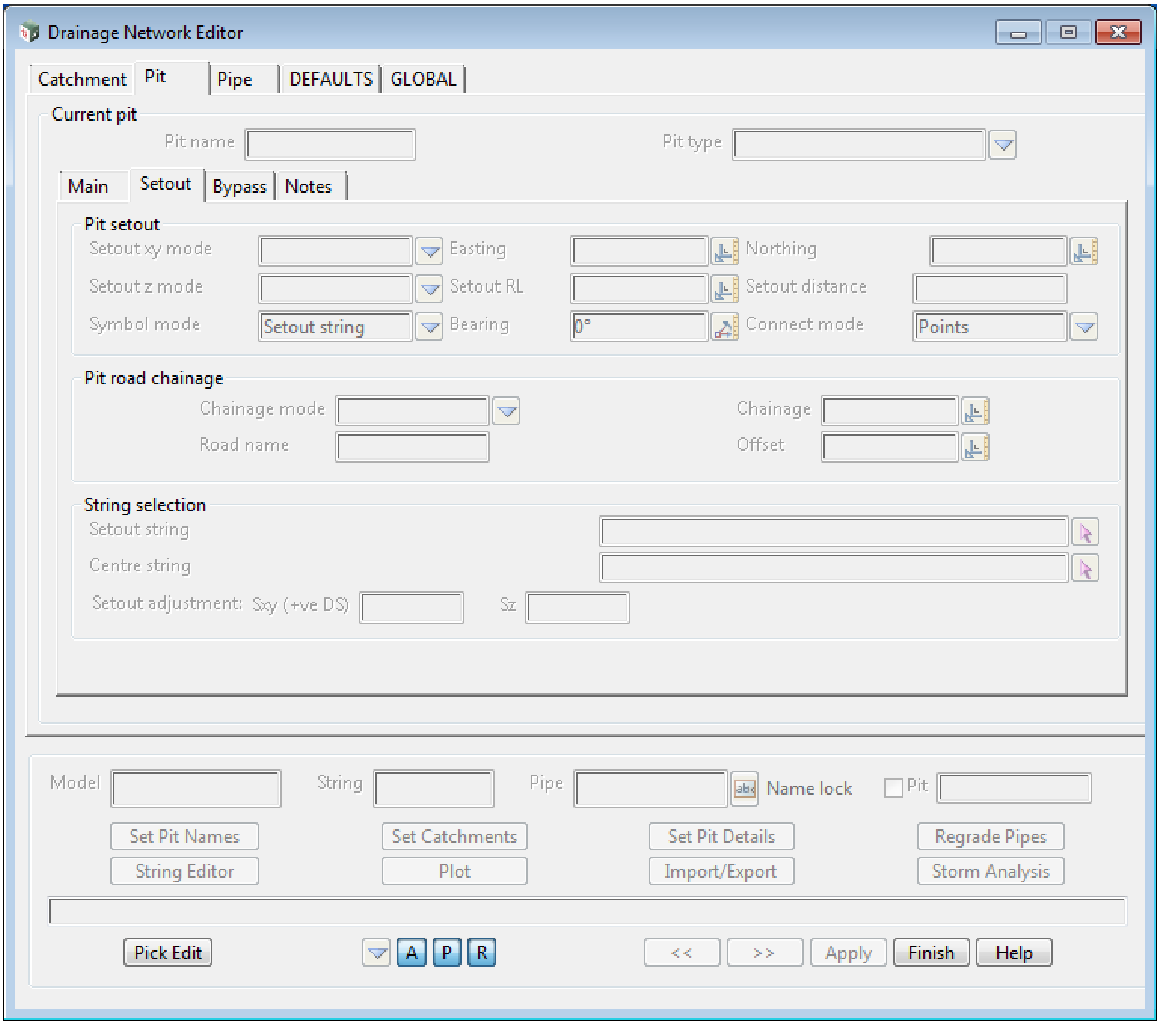
Setout to Grate Offset

Outlet and Tailwater Conditions

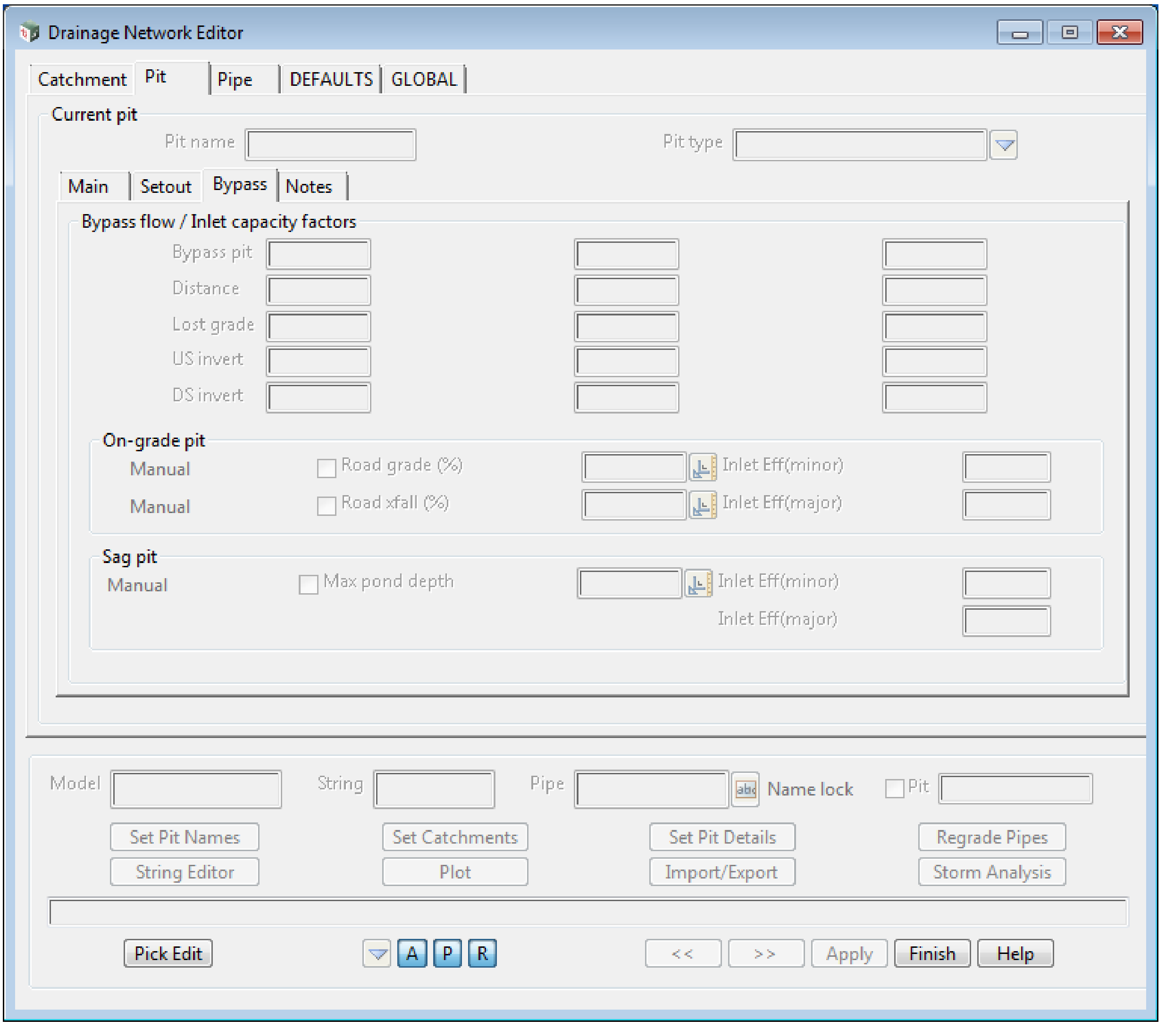
Pipe Design Parameters

Bypass Flow Settings

22.7.5.1 Setout subtab



22.7.5.2 Bypass subtab

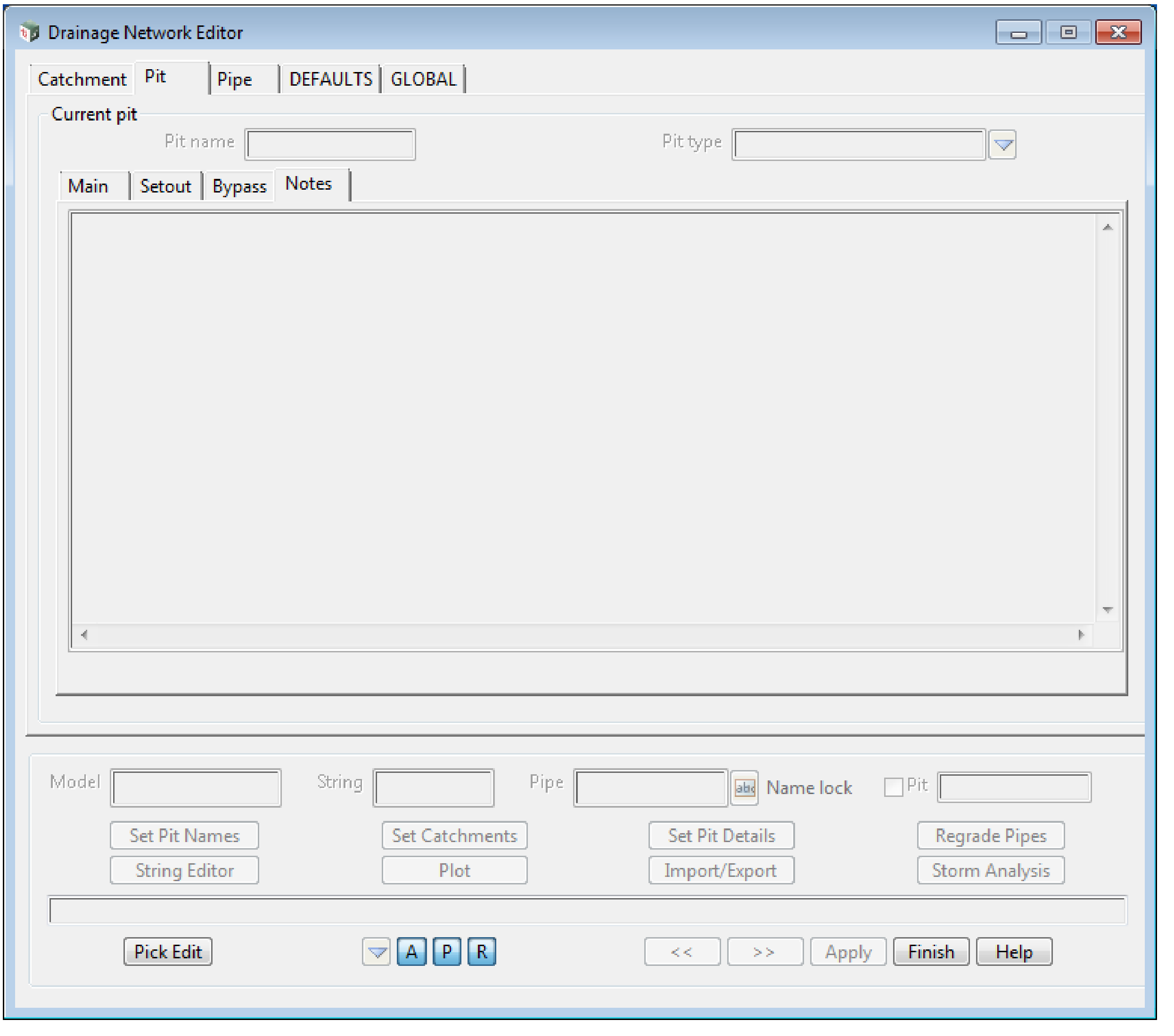


22.7.5.3 Bypass Shape subtab

22.7.5.4 Basin subtab

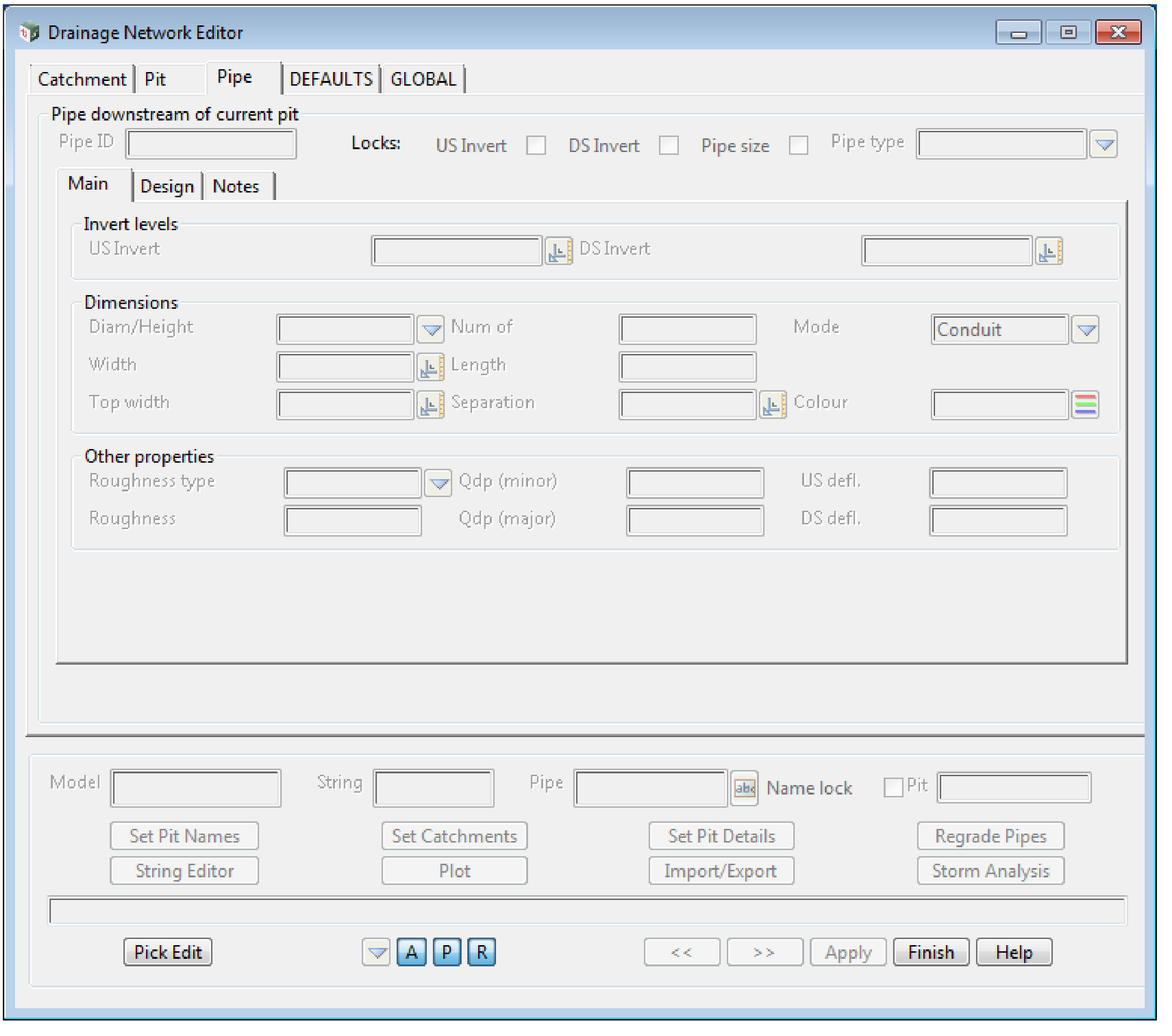
22.7.5.5 Inflow subtab

22.7.5.6 Notes subtab



22.7.6 Pipe tab

The Pipe tab is used to specify the following data:

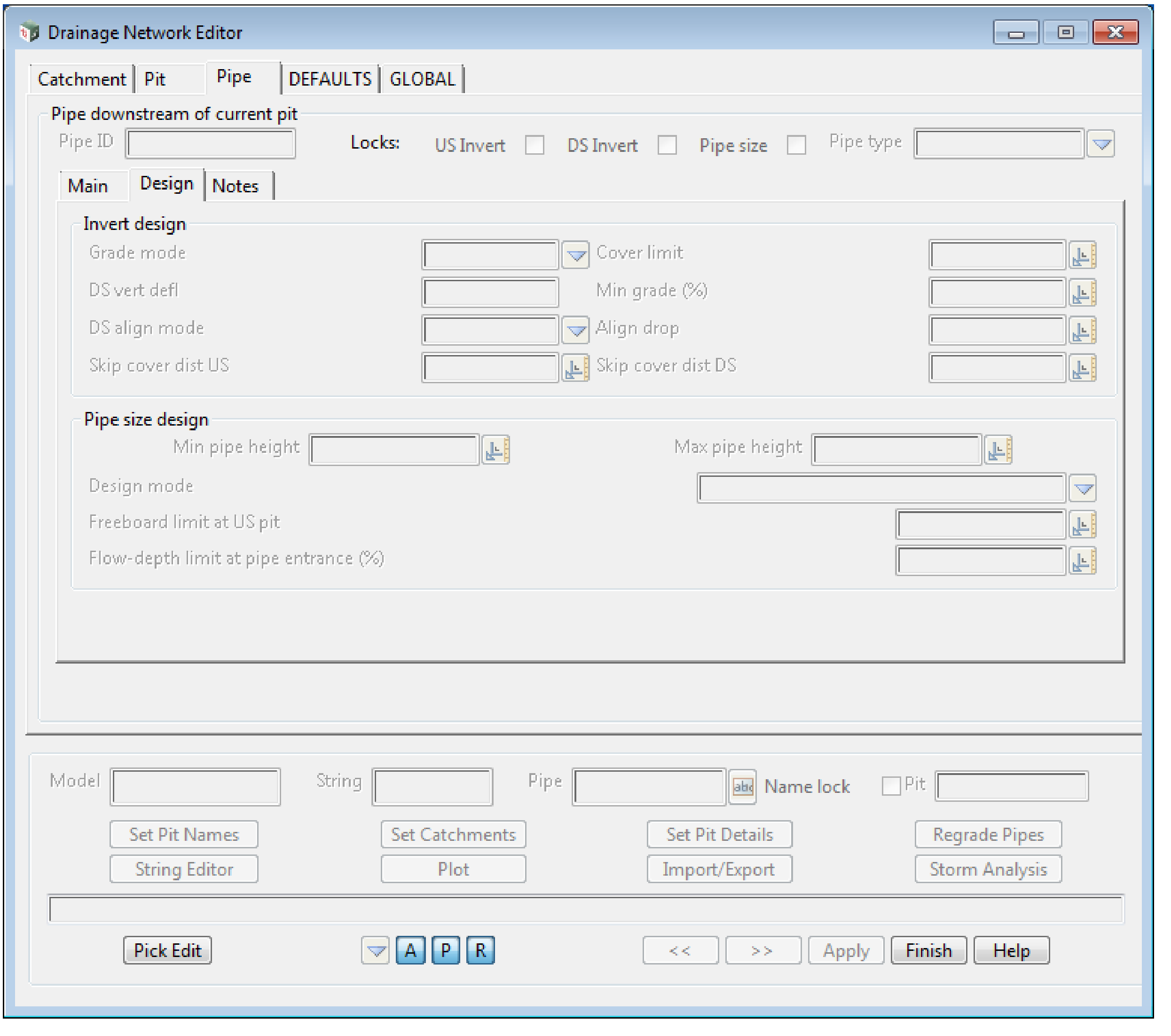


The pipe tab has the explicit settings for the values entered on the Defaults pipe tab. When these fields are blank the default value is used. It has 2 sub tabs.

Pipe:Main subtab

Pipe sizes, Max pipe height and Multiple Pipes and Box Culverts

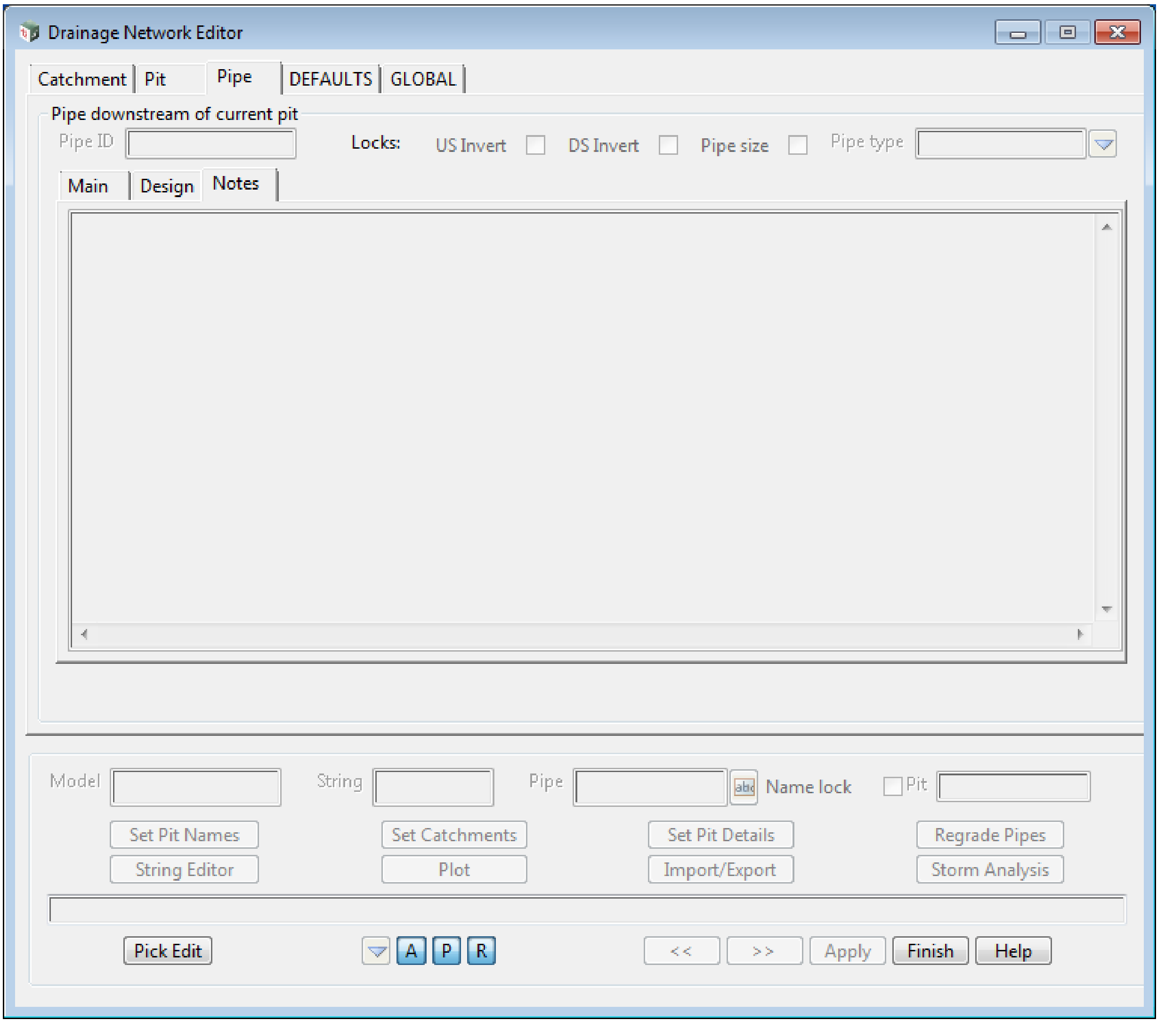
22.7.6.1 Design subtab



Pipe Friction Values and Freeboard Limit
Pipe Friction Method
Pipe Friction Values and Freeboard Limit
Pipe Design Parameters

22.7.6.2 Channels subtab

22.7.6.3 Notes subtab



22.7.6.4 Set Catchments

This selection links the catchment strings to the manholes and recalculates the areas. For SAG pits the catchments strings from all 3 sets are draped onto the design surface to locate the lowest overflow point. This level is used for the ponding depth. Information messages are written to the output window during this process.

See Set Catchments

22.7.6.5 Regrade Pipes

This option resets the pipe invert levels using the pipe design parameters. Information, warning and problem messages will be displayed in the output window. These messages will include pipe cover warnings, service crossing data and invert alignment messages. The user may place too many restraints using the regrade options results in no feasible solutions to the grading. These messages will also be shown.

22.7.6.6 Set Pit Details

This option resets the following manhole values: cover level. If activated the following are also set: road design string, setout x,y and z, road centre line chainage and offset, the manhole symbol rotation, the road grade and crossfall, the bypass pit,

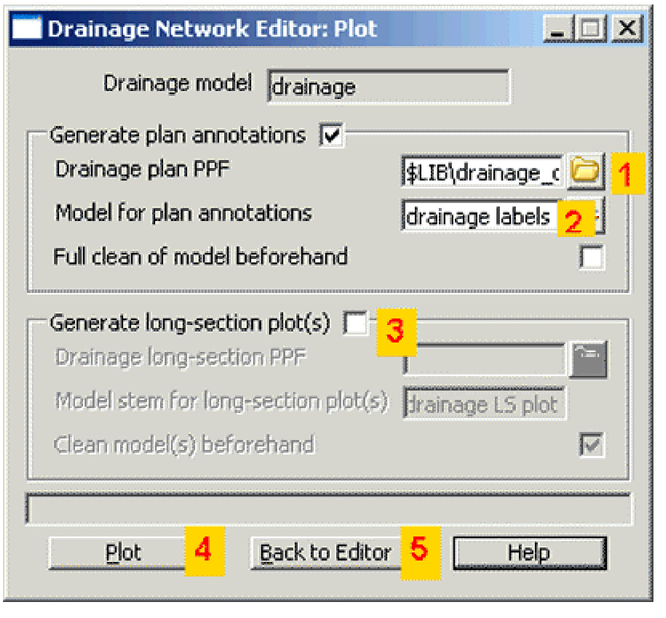
22.7.6.7 Set Pit Names

Tin for pit cover level   tin box

The new drainage manhole cover level will be set to the tin level at the manhole centre (optional)

22.7.6.8 Plot

This plotting option will create a drainage plan and/or long section using the ppf files entered. The ppf editors can be launched from this panel by selecting the More Information folder icon beside the ppf field.



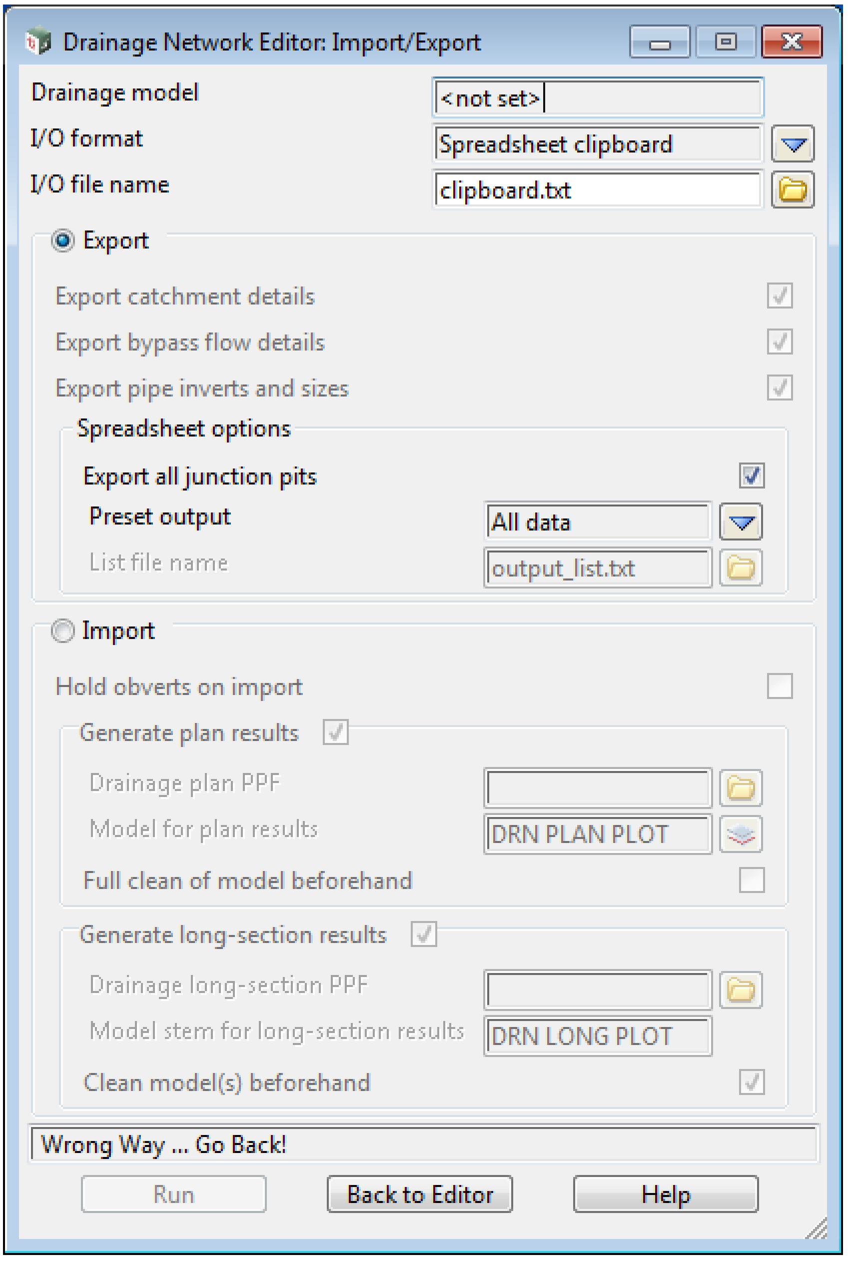
See Using Drainage Network Plot Button

22.7.6.9 Import/Export

22.7.6.10 String Editor

22.7.6.11 Set Catchments

S

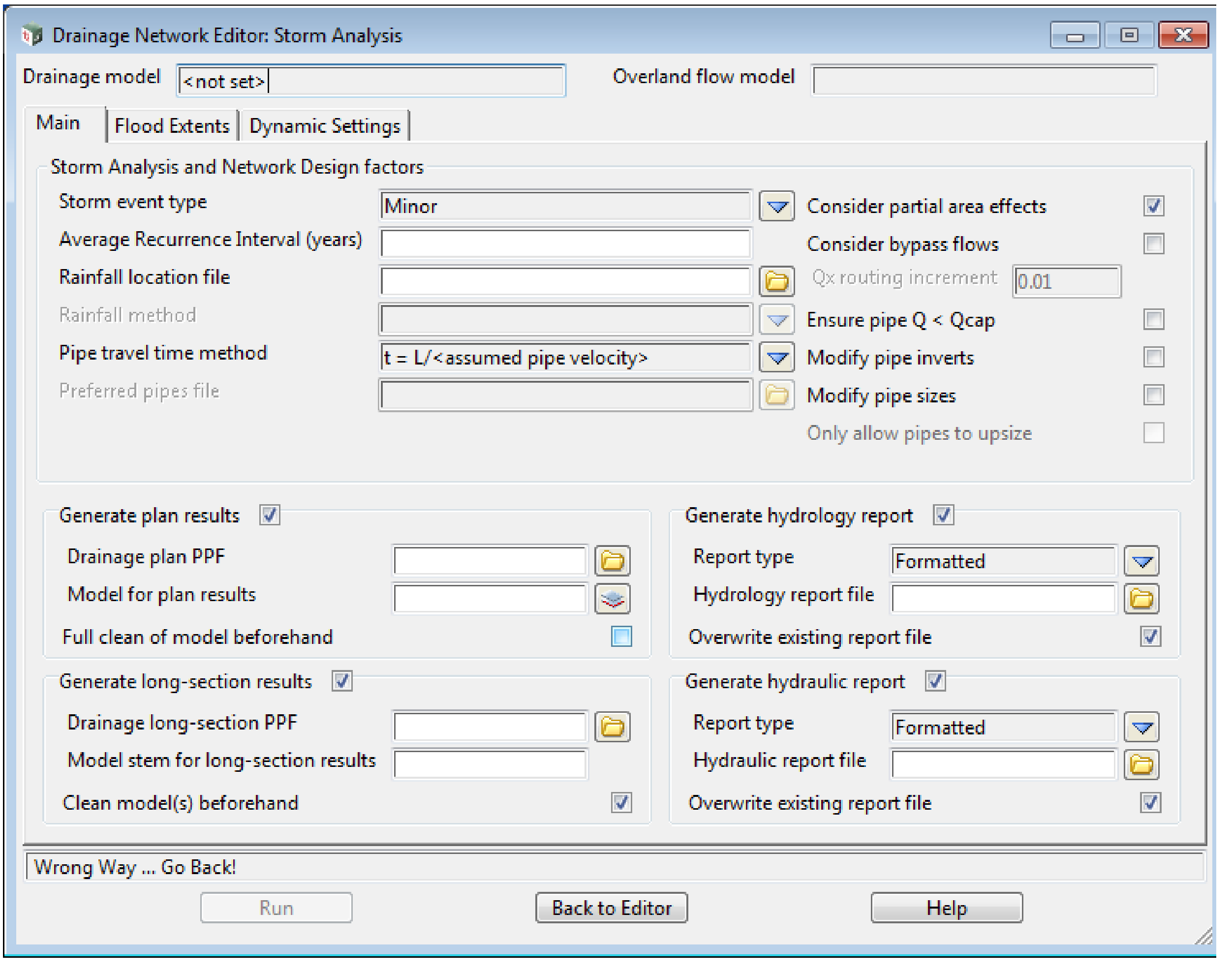


See Drainage Export and Import to Design Software

22.7.7 Drainage Network Design

See Also Using [Drainage Design in 12d Drainage Design]

On selecting the Storm Analysis button, the Drainage Network Design panel is displayed.



Field DescriptionType		DefaultsPop-Up

Run		button

Run 12d hydrology and hydraulic check routine

Edit		button

return to the Drainage Network Editor

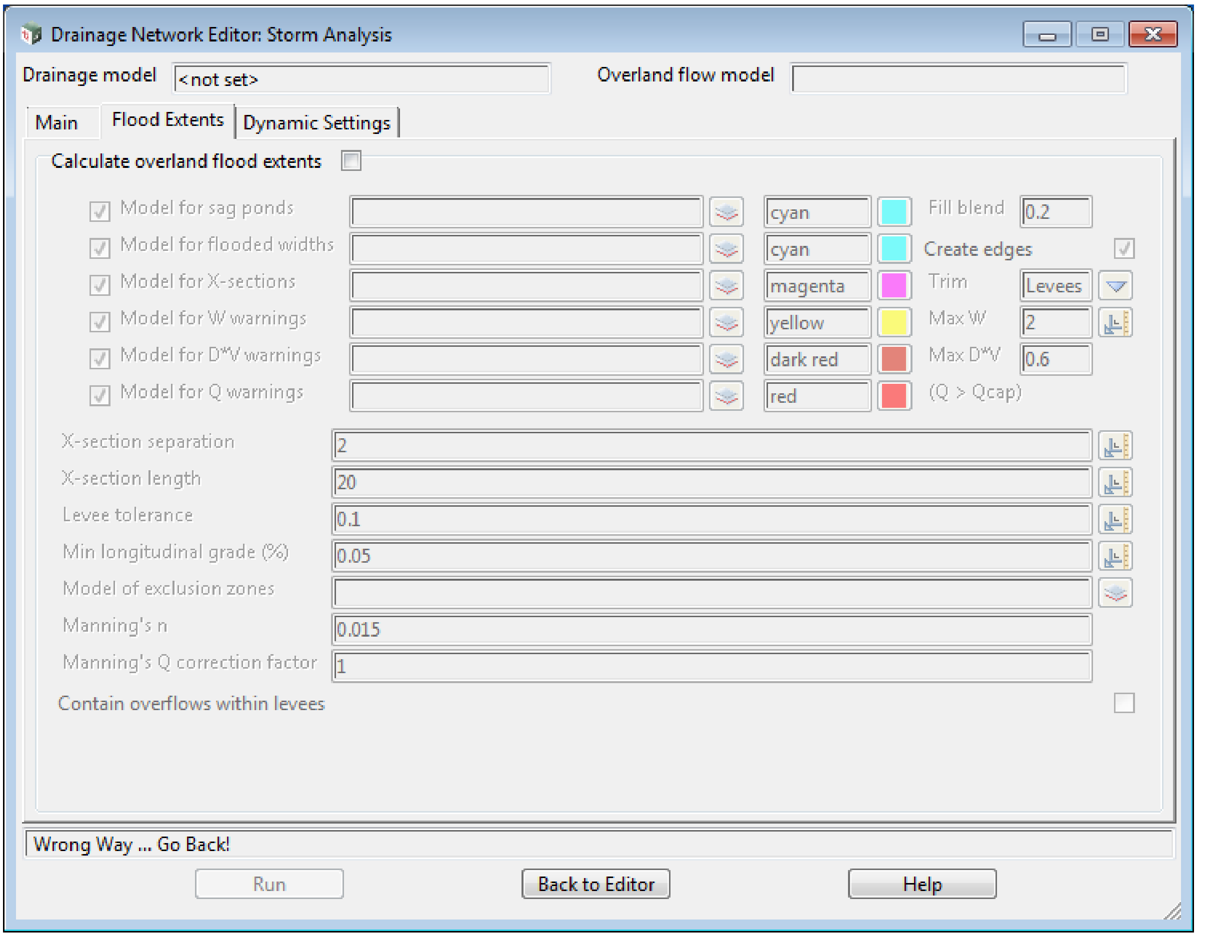
Finish		button

removes the panel

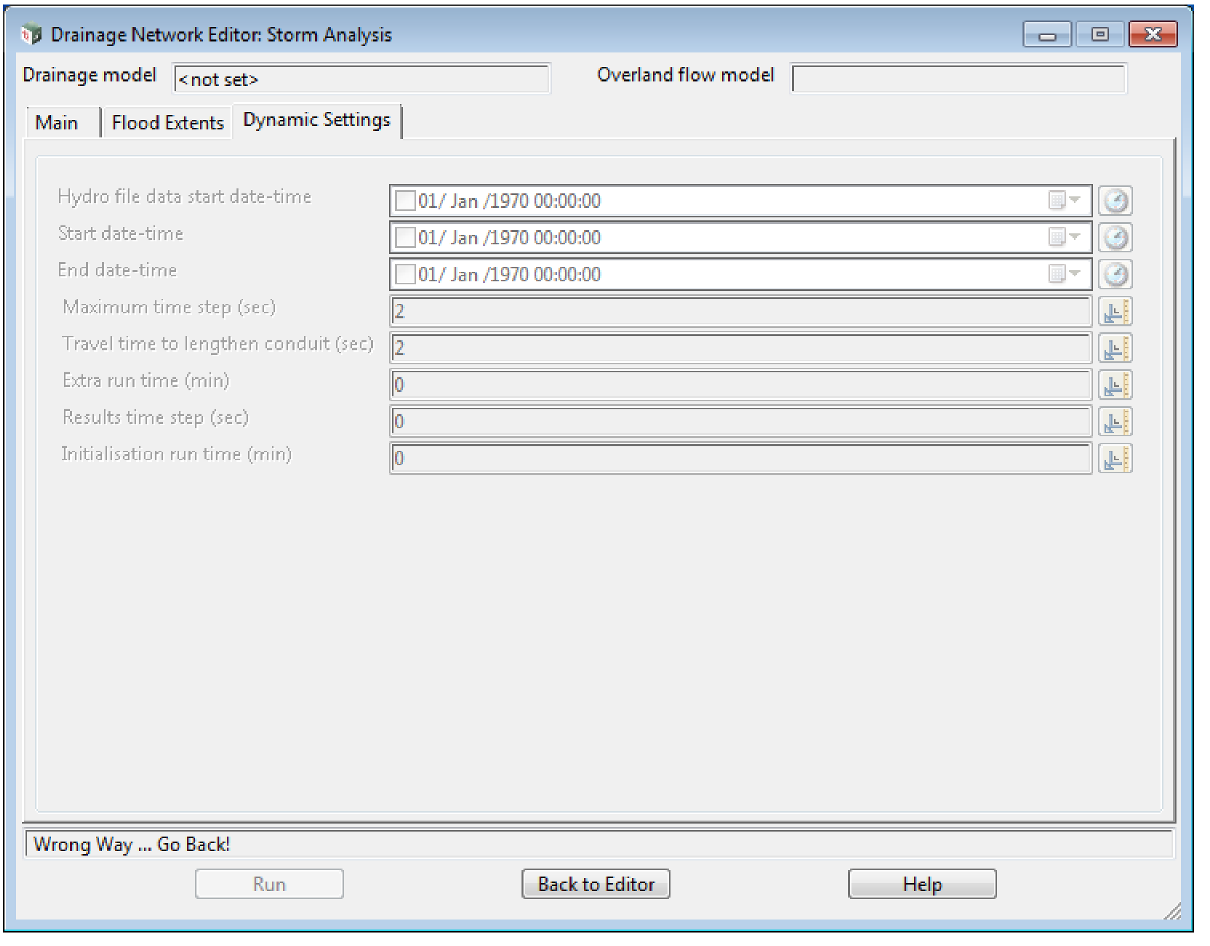
Help		button

displays this help page

22.7.8 Flood Extents Tab

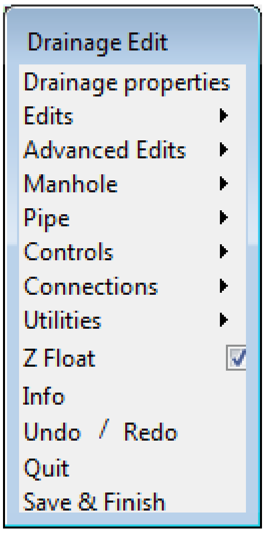


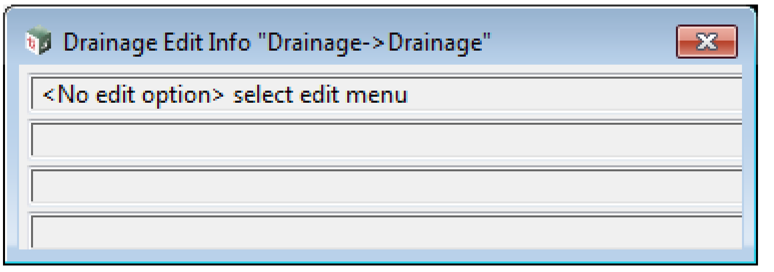
22.7.9 Dynamic Settings Tab



22.7.10 Drainage Edit

On picking a drainage string when in the string **Editor** option, or by selecting **Create** in the **Create Drainage String** panel, the **Drainage Edit** menu and panel are placed on the screen.





The edit is **cancelled** by selecting the **Quit**. No changes to the string are recorded and the **Drainage Edit** option terminates.

The edit is **finished** and the changes recorded when the **Finish** is chosen. The **Drainage Edit** option then terminates.

When either button is selected, a **Yes-No-Cancel** panel is displayed and the user must confirm the selection.

The **Drainage Edit Info** panel is principally used to display information and messages during the creation and editing of the drainage string. It is toggled on and off by **Info** on the **Drainage Edit** menu.

The main message area indicates the purpose of the mouse buttons at each step.

Message area 1 displays the current Drainage Edit option and message area 4 indicate the next step in the edit option.
Message areas 2 and 3 are used to display information about the string as the cursor is moved near the drainage string and the drainage manholes.

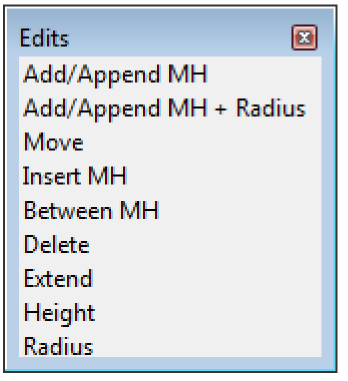
Each of the options in the **Drainage Edit** menu will now be described.

For the option *Drainage properties*, please continue to the section .
Edits, please continue to the section Edits.
Advanced Edits, please continue to the section Advanced Edits.
Manhole, please continue to the section Manhole (Maintenance Hole).
Pipe, please continue to the section Pipe.
Controls, please continue to the section House Connections.
Connections, please continue to the chapter House Connections.
Utilities, please continue to the chapter Utilities.
Z float, please continue to the chapter Z Float.

22.7.10.1 Edits

The options in the **Edits** menu are used for placing the initial manholes of a new drainage string, editing a new drainage string once the initial manholes are placed or editing an existing drainage string.

The **Edits** walk-right menu is



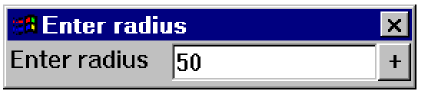
22.7.10.1.1 Append, Append+Radius

The **Append**, **Append+Radius** operations for a drainage string are similar to the **Append** options for a polyline string except that manholes are also automatically placed at each added point.

Note - a manhole can be removed from the point afterwards using the **Manhole=>Remove** option. The point will still be there but without a manhole on it.

If the **Append + Radius** option selected, then before each point is appended, an enter radius typed-input box is placed on the screen.

The enter radius typed-input box looks like



The radius is entered into the typed-input box, terminated with <enter>. The entered value is taken as the radius of the arc to the next drainage string point and the arc will be drawn correctly as the cursor is moved to the next point.

A **radius** value of **0** is taken to mean no arc.

If **z Float** is set to tick, then the top of the manhole is automatically placed on the terrain given in the tin field for the drainage string.

If height snap is toggled on in the **Snaps** menu, the height will be displayed an a new height entered but the new height will only be used if **z Float** is turned off for the manhole.

Each manhole in the string is connected by a drainage pipe. The diameter, type and cover at each end of the **drainage pipe** are taken from the **Drainage Pipe Defaults** panel.

The **Append** options are terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.1.2 Move

If used in a plan view, the **Move** option is used to move the (x,y) position of a manhole.

If used in a section view, the **Move** option can be used to change the height of the top of any manhole or the height of either end of the drainage pipe connecting adjacent manholes in the drainage string.

After the **Move** option is chosen, the user selects either a manhole in a plan view or the top of a manhole or a pipe end in a section view.

If a manhole is selected from a plan view, then the plan position of the selected manhole will move with the cursor. The new position for the top of the manhole can be chosen by either cursor selection or typed input in exactly the same way as for moving a point on a 3d string.

If the top of a manhole or the end of a pipe connecting adjacent manholes is chosen in a section view, then only the height of the selected point can be modified. That is, if a point is selected in a section view, then it is constrained to move in the z-direction only.

The **Move** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.1.3 Insert

The **Insert** option is designed to place a new manhole between two adjacent manholes (note that the inserted manhole does not have to be on the line joining the two adjacent manholes).

To insert a new manhole, the two adjacent manholes are chosen by selecting the line connecting the two manholes. Once the **line** is selected, the new manhole is assumed to be at the current cursor position. As the cursor is moved, the drainage string is redrawn reflecting the changing position of the inserted manhole.

The **Insert** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.1.4 Between

The **Between** option is similar to the **Insert** option except that the inserted manhole **does** have to be on the line joining the two adjacent manholes. To accomplish this, the cursor position is projected onto the manhole-manhole line to give the new manhole position.

Note - once the manhole is placed, it is no longer constrained to be on the one straight or arc joining adjacent manholes. If this is required, then the manhole is placed using the **Manhole=>Add** option.

The **Between** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.1.5 Delete

The **Delete** option is used to delete manholes/and or points from the drainage string.

The manhole/point to be deleted is selected with the cursor and when the selection is accepted (MB), it is **deleted**.

Once a manhole/point has been deleted, the delete option is still current and can be repeated without re-selecting the **Delete** option.

The string, minus the deleted manhole/point, is redrawn after each deletion.

The **Delete** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.1.6 Extend

The **Extend** option is used to move a manhole/point along the line joining the manhole/point to one of its adjacent manhole/point.

That is, the bearing of the manhole/point-manhole/point line is kept constant and the manhole/point moved along that line either **towards** or **away** from its neighbouring manhole/point.

Extending, like moving a manhole/point, is a two step process.

Step (a) - selecting the manhole/point-manhole/point line and the manhole/point to be moved along that line
Step (b) - selecting the final position for the manhole

Both steps are identical to extending an point in a polyline string.

Once the extend is completed, the extend option is still current and can be repeated without re-selecting the extend option.

The **Extend** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

Note - Extend can be used on the end manholes of the drainage string

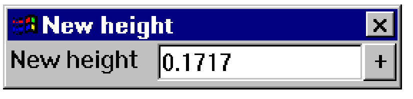
22.7.10.1.7 Height

The **Height** option is used to modify the height (z value) of the top of any manhole, or the height of the ends of the pipes connecting adjacent manholes in the drainage string.

The top of the manhole can be selected in either a plan or a section view. The pipe ends can only be selected in a section view.

After the **Height** option is chosen, the user must select the manhole top or pipe end that is going to have its height modified.

Once the manhole or pipe end has been selected, an New height typed-input box is displayed on the screen with the items current height (z value) in it.

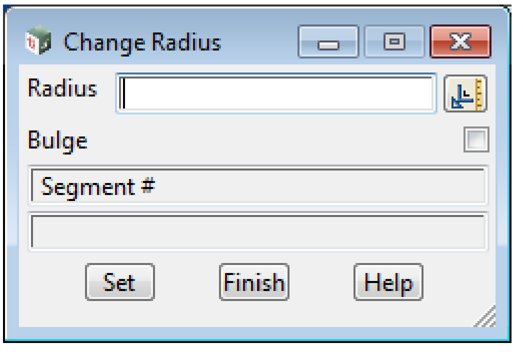


The required height is entered into the typed-input box, terminated with <enter>. The entered value is taken as the height of the manhole or pipe and the string redrawn with the new height at that point. The typed-input box then disappears.

The **Height** option is terminated by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.1.8 Radius

Selecting **Radius** brings up the **Change Radius** panel which is used to modify the radius of any arc/line joining adjacent manhole points.



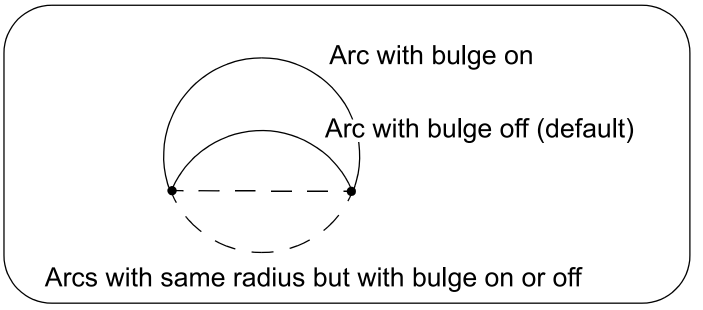
After selecting the **Radius** option, the user selects the arc/straight to be modified and the current arc radius and bulge setting will be displayed in the **Change Radius** panel.

New values can then be entered and the arc modified by selecting the **Set** button.

If the radius is positive, the arc is drawn from the start point to the next point on the polyline in a clockwise direction. If the radius is negative, the arc is drawn from the start point to the next point on the polyline in a counter-clockwise direction.

For a given radius (positive or negative), there are two possible cases for the arc- one where the arc is less than a semi-circle, the other when the arc is greater than a semi-circle.

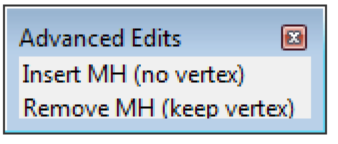
If bulge is turned on, the larger arc is used. The default is bulge turned off.



22.7.10.2 Advanced Edits

The options in the **Advanced Edits** menu are used for placing the adding manholes at locations other than a vertex and removing these manholes.

The **Advanced Edits** walk-right menu is



22.7.10.2.1 Insert MH (no vertex)

The **Insert MH (no vertex)** operation for a drainage adds a manhole on the existing string. When either of the 2 adjacent manholes with a vertex are moved, this manhole will move as well.

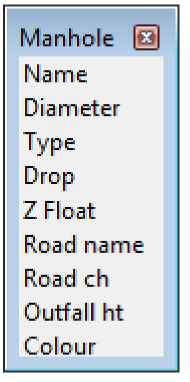
22.7.10.2.2 Remove MH (no vertex)

The **Remove MH (no vertex)** operation for a drainage removes a manhole on the existing string. If the manhole is on a vertex the vertex remains in the horizontal geometry but does not affect the vertical geometry.

22.7.10.3 Manhole (Maintenance Hole)

The options in the **Manhole** menu are used to modify information about individual manholes in the drainage string.

The **Manhole** walk-right menu is



22.7.10.3.1 Add

The **add** option is similar to the **Edits=>Between** option in that a new manhole is created that must be on the line or arc between two adjacent manholes.

However, once the manhole is placed, it is **always constrained** to be on the one straight or arc joining its adjacent manholes.

To **add** a new manhole, the two adjacent manholes are chosen by selecting the line connecting the two manholes. Once the **line** is selected, the new manhole is assumed to be at the current cursor position. As the cursor is moved, the drainage string is redrawn reflecting the changing position of the inserted manhole.

The **add** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.3.2 Remove

The **Remove** option is used to remove manholes from the drainage string but unlike the **Edits=>Delete** option, the underlying point on the drainage string remains. Hence only the manhole is deleted, but not the polyline point underneath.

The manhole to be removed is selected with the cursor and when the selection is accepted (MB), it is **removed**.

Once a manhole has been removed, the **Remove** option is still current and can be repeated without re-selecting the **Remove** option.

The string, minus the deleted manhole/point, is redrawn after each removal.

The **Remove** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.3.3 Name

The **Name** option is used to change a manhole's label.

After selecting the option, the mouse is used to pick the manhole to have a name change.

Once a manhole has been selected, an Enter text typed-input box is displayed on the screen containing the selected manhole's name.

The new name is entered into the typed-input box, terminated with <enter>. The typed-input box then disappears.

The **Name** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

NOTE - if the name of the manhole is *EOL* or *eol* then the diameter of the manhole is forced to zero.

22.7.10.3.4 Diameter

The **Diameter** option is used to change a manhole's diameter.

After selecting the option, the mouse is used to pick the manhole to have its diameter modified.

Once a manhole has been selected, an **Enter value** typed-input box is placed on the screen displaying the selected manhole's current diameter.

The new diameter is entered into the typed-input box, terminated with <enter>. The typed-input box then disappears.

The **Diameter** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.3.5 Type

The **Type** option is used to change the type of the cover (lid) placed on a manhole.

After selecting the option, the mouse is used to pick the manhole to have its cover type modified.

Once a manhole has been selected, a **Manhole type** typed-input box is displayed on the screen containing the selected manhole's cover type.

The list of common covers for the manhole can be obtained by clicking the **[+]** in the **Manhole type** input box and getting a menu of valid cover types.

The new cover type is entered into the typed-input box, terminated with <enter>. The typed-input box then disappears.

The **Type** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.3.6 Drop

The **Drop** option is used to change the vertical distance between the bottom of the pipes on either side of a manhole, *i.e.*, the difference in the invert levels of the pipe.

Since the drop in simply the difference in vertical distance between two pipes on either side of a manhole, the actual value of the drop can be changed by moving either of the pipes up or down the manhole.

After selecting the option, the end of the pipe to be moved is selected with the mouse.

Once the pipe end been selected, an enter value typed-input box is displayed containing the drop across the manhole that the pipe end connects into.

The new drop type is entered into the typed-input box, terminated with <enter>.

If the end of the pipe was on the upstream side of the manhole (normally the right hand side of a manhole in a section view), the end of the pipe will be moved upward until the drop across the manhole equals the entered value.

If the end of the pipe was on the downstream side of the manhole (normally the left hand side of a manhole in a section view), the end of the pipe will be moved down until the drop across the manhole equals the entered value.

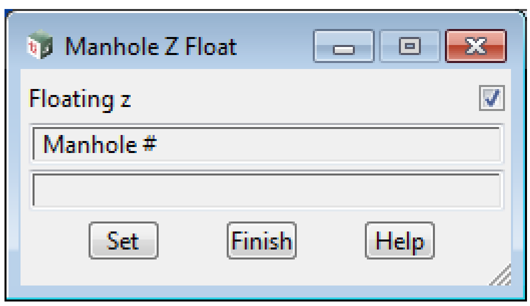
The typed-input box then disappears.

The **Drop** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.3.7 Z Float

A **floating manhole** takes the z-value for the top of the manhole from the tin selected for the drainage string. Hence as a floating manhole moves around in a plan view, the top of manhole will automatically change to suit the new z-value of the drainage string tin.

Selecting **Z float** brings up the **Manhole Z Float** panel which is used to change a manhole from floating to non-floating and vice-versa.



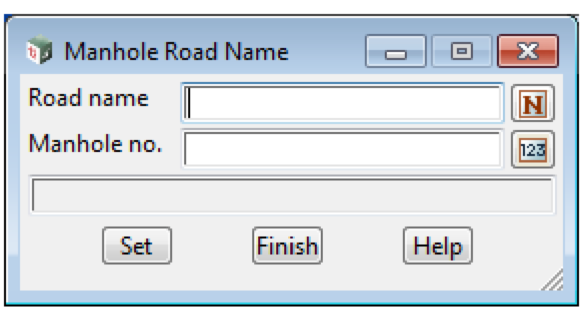
After selecting the **Z Float** option, the user selects the manhole to be modified and the current **floating z status** is displayed in the **Manhole Z Float** panel.

The tick box can then be changed and the manhole modified by then selecting the **Set** button.

22.7.10.3.8 Road Name

A **road name** can be set for a manhole of the line.

Selecting **Road name** brings up the **Manhole Road Name** panel which is used to give and change a road name for a manhole.



The option is running as soon as the panel is on the screen and the user is asked to selected a manhole.

<Select manhole> [picks][][Menu]

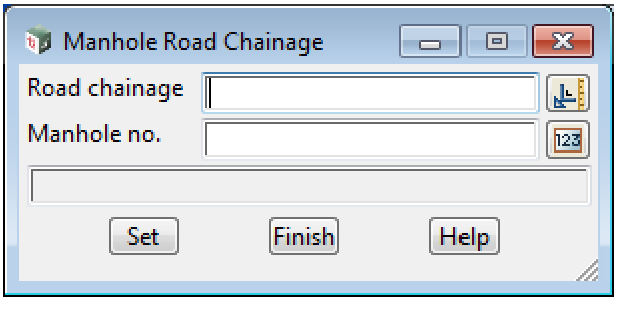
When the manhole is selected, any existing road name and the manhole number is placed in the road name and vertex no. fields respectively of the **Manhole Road Name** panel.

A new **road name** is then typed into the road name panel field and the **Set** button selected to record the road name with the manhole.

22.7.10.3.9 Road Chainage

A **road chainage** can be set for a manhole of the line.

Selecting **Road ch** brings up the **Manhole Road Chainage** panel which is used to give and change a road chainage for a manhole.



The option is running as soon as the panel is on the screen and the user is asked to selected a manhole.

<Select manhole> [picks][][Menu]

When the manhole is selected, any existing road chainage and the manhole number is placed in the road chainage and vertex no. fields respectively of the **Manhole Road Chainage** panel.

A new **road chainage** is then typed into the road chainage panel field and the **Set** button selected to record the road chainage with the manhole.

22.7.10.3.10 Outfall Height

When a drainage string is created, the furthest downstream is often an outfall and the height of the outfall known.

When the furthest downstream manhole is considered to be an outfall, the **Outfall ht** option is used to set and change the outfall height. Once set, the outfall height is taken to be the height at the *bottom* of the furthest downstream manhole.

If this option is not used, the outfall height is left undefined.

After selecting the option, an enter value typed-input box is displayed on the screen containing the existing outfall height or **null** if no outfall height has been set.

The new outfall height is entered into the typed-input box, terminated with <enter>.

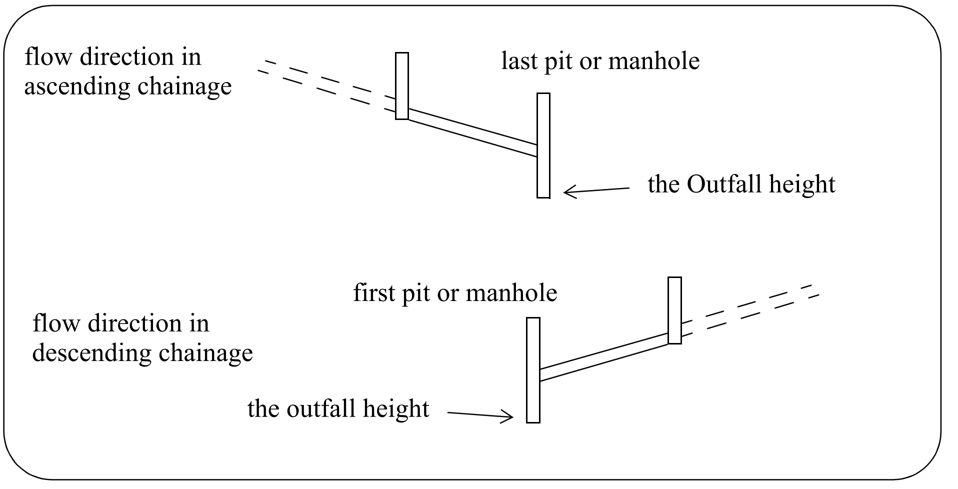
The height of the bottom of the furthest downstream manhole in the drainage string is then set to the given height and the typed-input box removed.

The **Outfall ht** option automatically terminates after use.

Note:

For a drainage string with flow direction in ascending chainage, the furthest downstream manhole is the *last* manhole.

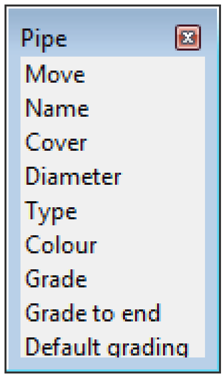
For a drainage string with flow direction in descending chainage, the furthest downstream manhole is the *first* manhole.


flow direction in
ascending chainage
last pit or manhole
the Outfall height
first pit or manhole
flow direction in
descending chainage
the outfall height

22.7.10.4 Pipe

The options in the **Pipe** menu are used to modify information about the pipes joining adjacent manholes in the drainage string.

The **Pipe** walk-right menu is



22.7.10.4.1 Move

The **Move** option is used to move a pipe connecting two manholes whilst keeping the grade of the pipe constant.

After selecting the option, the mouse is used to pick the pipe to be moved.

Once a pipe has been selected, the pipe will be moved up or down so that the cursor remains on the pipe (or on the extension of the pipe if the cursor is on the other side of the manholes at either end of the pipe). The grade of the pipe is kept constant.

The manholes at either end of the pipe will also be extended if necessary so that the pipe still connects into the adjacent manholes.

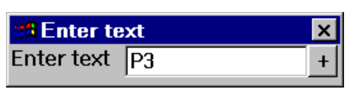
The **Move** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.4.2 Name

The **Name** option is used to change a pipe's label.

After selecting the option, the mouse is used to pick the pipe to have a name change.

Once a pipe has been selected, an Enter text typed-input box is displayed on the screen containing the selected pipe's name.



The new name is entered into the typed-input box, terminated with <enter>. The typed-input box then disappears.

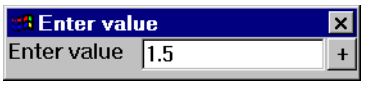
The **Name** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.4.3 Cover

The **Cover** option is used to place the selected pipe so that minimum cover is maintained along the top of the pipe (obvert) with respect to the finished surface tin (tin (fs)).

After selecting the option, the pipe to set the cover for is selected with the mouse.

Once a pipe has been selected, an Enter value typed-input box is displayed on the screen containing the existing cover.



The new cover is entered into the typed-input box, terminated with <enter>. The typed-input box then disappears and the pipe is adjusted so that the specified cover is maintained for the full length of the pipe at the pipes existing grade.

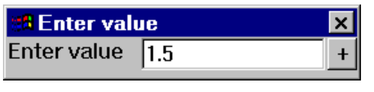
The **Cover** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.4.4 Diameter

The **Diameter** option is used to change a drainage pipe's diameter.

After selecting the option, the mouse is used to pick the pipe to be modified.

Once a drainage pipe has been selected, an Enter value typed-input box is placed on the screen displaying the selected pipe's current diameter.



The new diameter is entered into the typed-input box, terminated with <enter>. The typed-input box then disappears

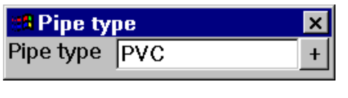
The **Diameter** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.4.5 Type

The **Type** option is used to change the type of material that the pipe is made of.

After selecting the option, the mouse is used to pick the pipe to have its material type modified.

Once a pipe has been selected, a Pipe type typed-input box is placed on the screen displaying the selected pipes material type.



The list of common materials for the pipe can be obtained by clicking B3 in the type input box and getting a menu of valid material types.

The new material type is entered into the typed-input box, terminated with <enter>. The typed-input box then disappears.

The **Type** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.4.6 Grade

The **Grade** option is used to specify an exact grade for a drainage pipe. The value for the grade is entered using typed input and is in the units "1v in".

In this option, the new grade is given to a pipe by keeping one end fixed and raising or lowering the other end by the amount required to give the pipe the new grade.

After the **Grade** option is selected, the pipe whose grade is to be modified is selected **at the end that is going to be moved**.

Once the pipe end has been selected, an enter value typed-input box is displayed on the screen containing the current grade of the pipe. The new grade is entered into the typed-input box, terminated with <enter>.

The selected end of the drainage pipe is then moved up or down so that the pipe has the new grade.

If necessary, the manhole at the moved end of the pipe will be lengthened so that the pipe still connects into the adjacent manholes.

The **Grade** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.4.7 Grade to End

The **Grade to end** option is used to specify a ***fixed*** grade for all the drainage pipes from a selected pipe to the **low chainage** end (beginning) of the line. That is, it grades from *right* to *left* on a long section of the drainage string.

The value for the grade is entered using typed input and is in the units "1v in".

This option was originally written for pipes with flow direction in descending chainage direction which is the reason why it works from the selected pipe back towards the beginning of the line.

Hence **Grade to end** should only be run on pipes with flow in descending chainage direction. If the flow is in ascending chainage direction, use the reverse string option (**string=>strings edit=>reverse**) before using this option, and then reverse the string again after the option is run. Note that the **Default grading** option looks at maintaining minimum *grade* and minimum *cover* for flow in either ascending or descending chainage direction and supersedes **Grade to end**.

In this option, the new grade is given to a pipe by keeping the high chainage end of the selected pipe fixed and raising or lowering the low chainage end by the amount required to give the pipe the new grade. The minimum cover is over ridden during this process. If necessary, the bottom of the manhole will be lowered so that the pipe still connects into its adjacent manholes.

The next lower chainage pipe is then dropped through the default drop and given the new grade. The drop is measured from the lowest pipe invert of all pipes in the network entering the manhole.

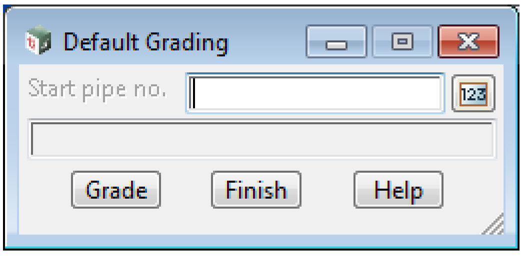
This process is repeated until the low chainage end (beginning) of the line is reached.

After the **Grade to end** option is selected, the first pipe whose grade is to be modified is selected.

Once the pipe end has been selected, an enter value typed-input box is displayed on the screen containing the current grade of the selected pipe. The new grade is entered into the typed-input box, terminated with <enter>. The option then does the grading.

The **Grade to end** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.4.8 Default Grading



The **Default grading** option starts at the given pipe and then works to the end of the line when the flow direction is in ascending chainage, or the beginning of the line when the flow direction is in descending chainage. The **Default grading** option maintains minimum grade and minimum cover.

Default grading is not designed to optimise the placement of the pipes but provides one solution preserving minimum grade and cover.

22.7.10.5 Property Controls

Property, block or lot **controls** are trial connections from a sewer line (drainage string) to a user specified plan point.

The property controls are used as checks that selected house blocks can be serviced by the sewer line. That is, they are used to test if the house block is under the control of the sewer line.

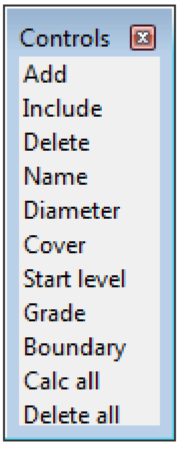
For the trial connection, once the cover (measured from the finished surface to the top of the property connection line) that the property connection must maintain and the grade are specified, the position and depth that the trial property connection must have at the sewer line can be calculated, and where the height of the *centre line* of the property control when it reaches the sewer line is displayed in any section view containing a profile of the sewer line.

If the calculated depth of the trial property connection at the sewer line (drainage string) is below the sewer pipe, then no such connection would be feasible and the house block would not be totally controlled by the sewer line.

On a section view, the *Profile =>One substring* and *Profile =>Many substrings* options will profile the property control. Note that the *centre line* (axis) of the property control is drawn on the section view, not the invert (bottom) or the obvert (top).

NOTE - property controls are only accessible by the ***Sewer module.***

The options in the **Controls** menu are used to place and modify the trial connections. The **Controls** walk-right menu is



22.7.10.5.1 Add

The **Add** option is used to create a new trial control line going from a selected drainage pipe to a user specified plan position (the free end of the connection).

After selecting the option, the mouse is used to pick the plan position of the connection point on the drainage pipe, and then the points defining the path to the plan position of the free end of the trial connection.

After the free end has been selected, the property name typed-input box is displayed on the screen so that a label for the control can be entered (this usually consists of the lot number).

When the property name box is placed on the screen, it will already have some text in it. This text comes from the name field in the **Drainage Property Control Defaults** panel.

A z-value equal to the drainage tin value minus the default control cover depth is automatically given to the free end of the control. The default control grade is then used to define z-values along the trial path (ensuring that the default control cover depth is always observed) to give a z-value back at the drainage pipe (the connection height of the control).

A cross at the calculated connection height is displayed whenever the drainage string is profiled on a section view.

The colour for the new control is taken from the **Drainage Property Control Defaults** panel.

The **Add** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.5.2 Include

The **Include** option is used to make a control out of an existing polyline string.

The polyline string must start at the connection point on the drainage string and end at the free end (use *Strings=>Strings Edit=>Reverse* if the string's direction is incorrect).

After selecting the option, the mouse is used to pick the polyline string and its plan position is copied and used to define a new control.

A z-value equal to the default drainage tin at that point, minus the default control cover depth is automatically given to the free end of the new control. The default control grade is then used to define z-values along the trial path (ensuring that the default control cover depth is always observed) to give a z-value back at the drainage pipe.

The **Include** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.5.3 Delete

The **Delete** option is used to delete a control.

After selecting the option, the mouse is used to pick the block control to be deleted.

Once a block control has been selected, it will deleted and removed from the screen.

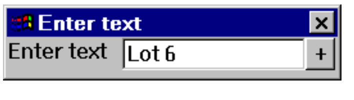
The **Delete** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.5.4 Name

The **Name** option is used to change the **Property name** of a control.

After selecting the option, the mouse is used to pick the control that will have a name change.

Once a control has been selected, an enter text typed- input box is displayed on the screen containing the selected control's name.



The new name is entered into the typed-input box, terminated with <enter>. The typed-input box then disappears.

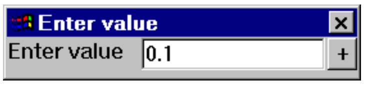
The **Name** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.5.5 Diameter

The **Diameter** option is used to change the diameter of a control.

After selecting the option, the mouse is used to pick the control that will have its diameter modified.

Once a control has been selected, an Enter value typed- input box is placed on the screen displaying the selected control's current diameter.



The new diameter is entered into the typed-input box, terminated with <enter>. The typed-input box then disappears.

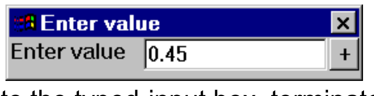
The **Diameter** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.5.6 Cover

The **Cover** option is used to change the distance that the control is below the tin (fs) for the drainage string.

After selecting the option, the cursor is used to pick the control whose cover will be modified.

Once a control has been selected, an Enter value typed- input box is placed on the screen displaying the selected block control's current cover.



The new cover is entered into the typed-input box, terminated with <enter>. The typed-input box then disappears and the new connection height using the new cover calculated.

The **Cover** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.5.7 Grade

The **Grade** option is used to change the grade of a control (units "1v in").

After selecting the option, the mouse is used to pick the control whose grade will be modified.

Once a control has been selected, an enter value typed- input box is placed on the screen displaying the selected control's grade.

The new grade is entered into the typed-input box, terminated with <enter>. The typed-input box then disappears and the new connection height using the new grade calculated.

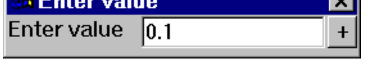
The **Grade** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.5.8 Boundary

The **Boundary** option is used to specify a boundary trap depth which is used as a final drop at the drainage pipe end of the control.

After selecting the option, the mouse is used to pick the control whose boundary depth will be modified.

Once a control has been selected, an Enter value typed- input box is placed on the screen displaying the selected boundary depth.



The new boundary trap depth is entered into the typed-input box, terminated with <enter>. The typed-input box then disappears and the new connection height using the new boundary trap depth is calculated.

The **Boundary** option is terminated by selecting **Cancel** from the **Pick Ops** menu, or by selecting **Quit**, **Finish**, or a new option from the **Drainage Edit** menu.

22.7.10.5.9 Calc all

The **Calc all** option is used to re-calculate the connection heights for all controls of the drainage line.

The calculation uses the plan layout of the control, the control's cover, grade and boundary trap depth and the default tin for the drainage line.

The **Calc bc's** option automatically terminates after use.

22.7.10.5.10 Delete All

The **Delete all** option is used to remove all the controls defined for the drainage line being edited.

The **Delete all** option automatically terminates after use.

22.7.10.6 House Connections

The **Connections** option is used to create the connections from the sewer line (drainage string) to the house blocks in a subdivision.

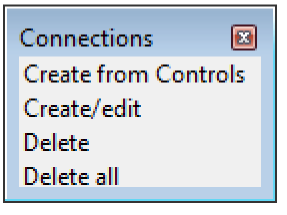
When placing a house connection, the user indicates the two corners of the frontage of the block to allow the cross-fall of the block to be calculated and allow the frontage to be used in positioning the house connection.

When placing house connections, a section view is used to automatically display the connection to facilitate checking the type of connection used and any obstructions that may need to be avoided.

NOTES

1. House connections are only accessible by the ***Sewer module.***
2. All house connection calculations do not take into account any thickness of pipe, joint sizes or actual entry points into the sewer. Hence they are ***approximate only*** and should only ever be used as a guide. Any quantities calculations should allow for a suitable margin of error.

The **Connections** walk-right menu is



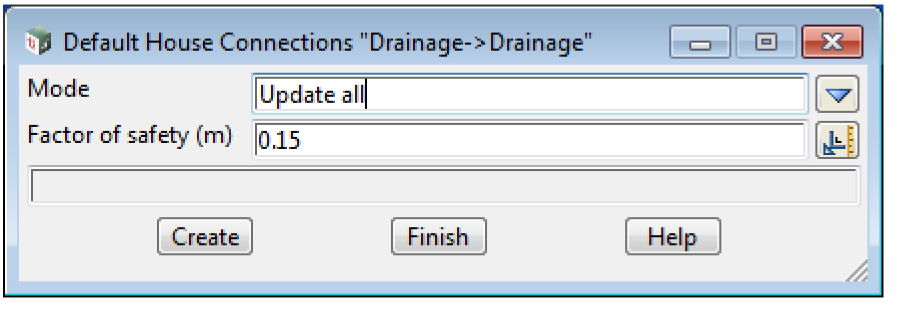
The process for creating a house connection will now be described in detail.

22.7.10.6.1 Create from Controls

The **Create from Controls** option is used to create a new house connection from existing controls.

or modify an existing house connection.

After selecting the **Create from Controls** option, the **Default House Connections** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Mode	input	update all	remove existing first, update new, update all

if ***remove existing first****, all the existing connections are deleted before new ones are created from the controls.*
if ***update new****, connections are only created from controls with names different from any existing connection.*
if ***update all****, connections are created from all controls.*

Factor of safety input 0.15

the connection height for the control is adjust by this depth from the control connection depth.

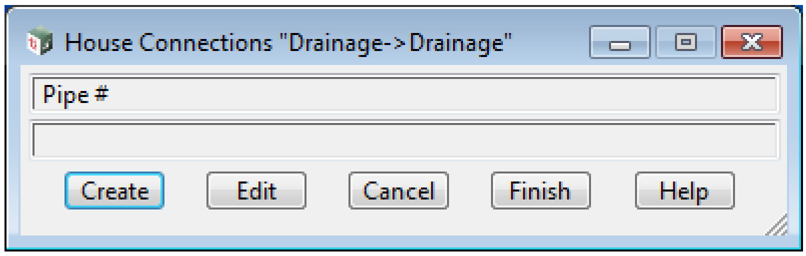
Create button

after selecting **Create***, the connections are created from the selected controls.*

22.7.10.6.2 Create/Edit

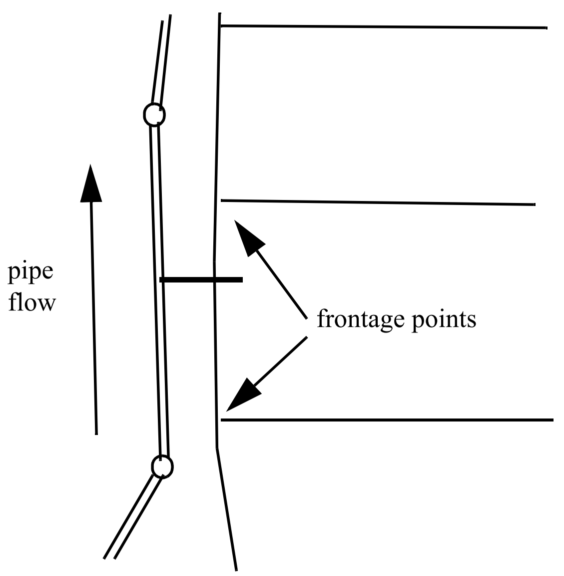
The **Create/edit** option is used to create a new house connection or modify an existing house connection.

After selecting the **Create/edit** option, the **House Connections** panel is displayed.



The process for creating a house connection will now be described in detail.

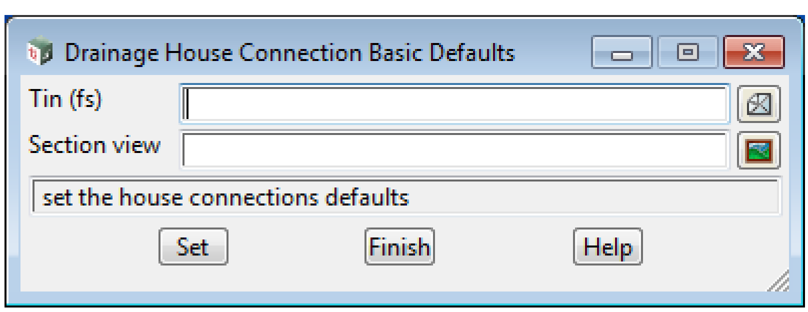
Creating A House Connection



Step 1 Select *Create*

To create a new house connection, the **Create** button is selected from the **House Connections** menu.

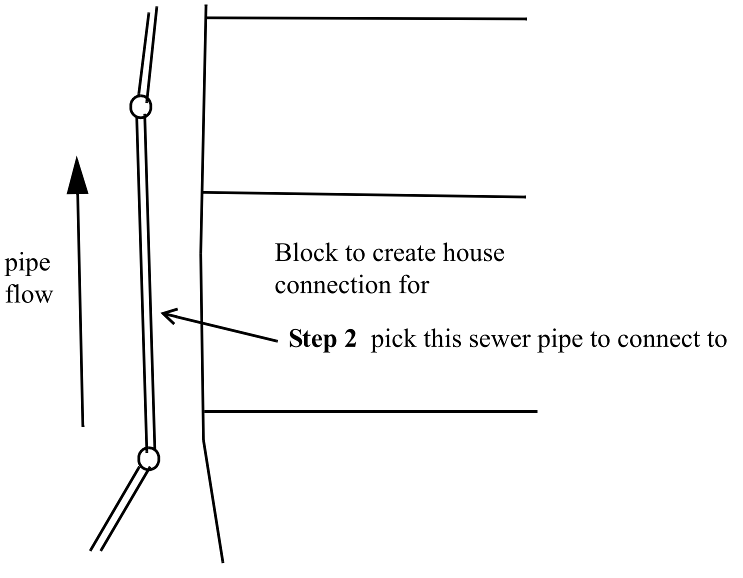
If either the default **drainage tin** or **section view** have not be defined, the **Drainage House Connection Basic Defaults** panel will be placed on the screen after the **Create** button is selected.



The missing values need to be filled in and the **Set** selected. This information is needed so that the house connections can be drawn up in a section view as soon as they are created.

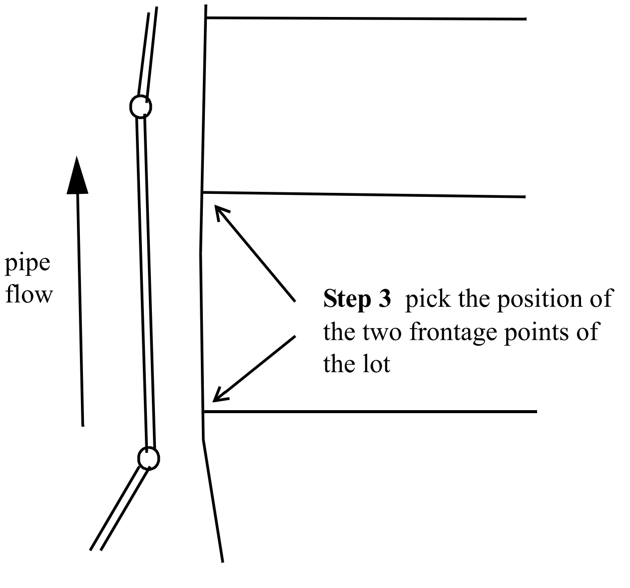
Step 2 Selecting the Sewer (drainage) Pipe to Connect to

The user is then asked to select the sewer pipe to connect into.



Step 3 Selecting the House Lot Frontage

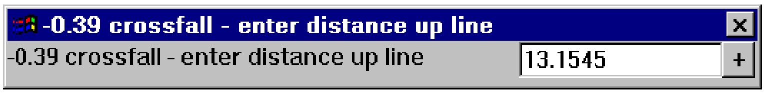
Next the user indicates the two frontage points for the house block.



Step 4 Positioning the House Connection on the Sewer Pipe

From the two frontage points and the default drainage tin, the program calculates the crossfall for the block frontage.

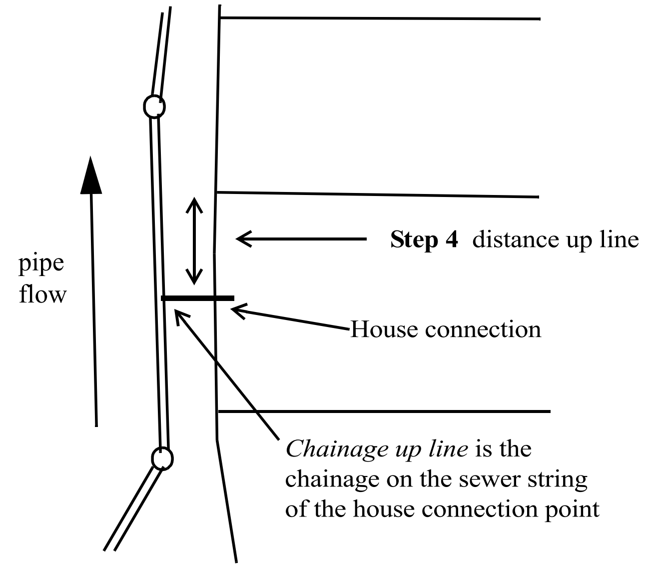
The crossfall is displayed in the name of the enter distance up line typed-input box which is now placed on the screen. The distance up line is the distance along the frontage that the house connection will be placed.



Depending on the crossfall, a default value for the distance up line is displayed. If the ground crossfall is greater than 1% the lot mid point is the suggested connection location. For crossfalls less than 1%, the connection is located 3m off the ***lowest*** lot boundary.

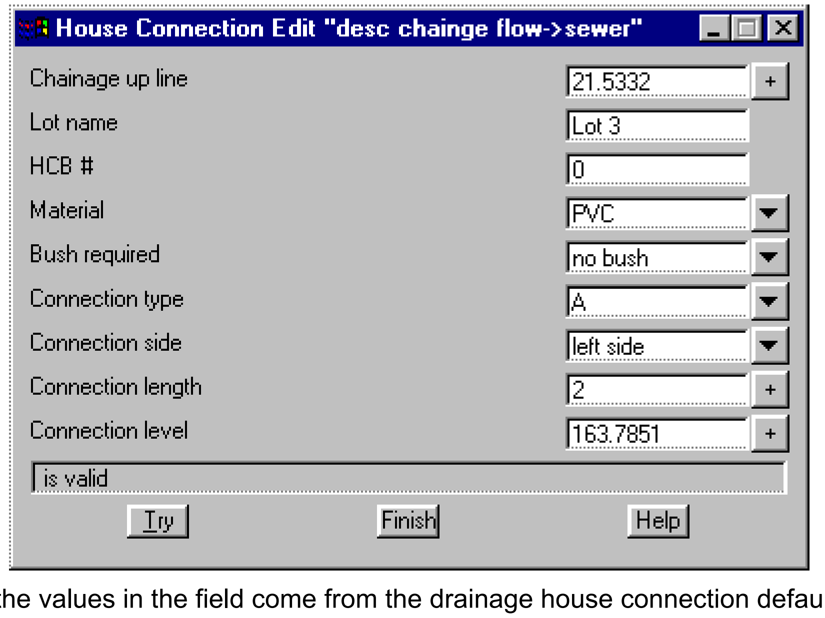
A pop-up also exists for placing the connection point at the distance 2.0, 3.0, 5.0 along the frontage, the mid point of the frontage, and the distances 2.0, 3.0 and 5.0 from the other end of the frontage.

The distance up line value is entered into the typed-input box, terminated with <enter>. The typed-input box is then removed from the screen.



Step 5 Defining the House Connection Type

Next the **House Connection Edit** panel is placed on the screen.



Most of the values in the field come from the drainage house connection defaults.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Chainage up line	input/output	chainage of connection	

the chainage on the sewer string where the house connection is attached to the sewer line.

Lot name	input	from drainage defaults	

name of the house block

HCB #	input		

a user supplied house connection branch (HCB) number

Material	input	from drainage defaults	PVC, PVC X/HEAVY, VC

the material used for the house connection.

Bush required	input	no bush	no bush, PVC to VC

the type of bush used

Connection type	input	from drainage defaults	A, A Special, B, C, OB, Special jump up

the type of connection used. Go to the section House Connection Types *for a description of each connection type*

Connection side	input	right side	right side, left side

the side of the line to make the connection

Connection length	input	from drainage defaults	

length of pipe for the house connection

Connection level	input	calculated	

height of the house connection. When this is first displayed, it is calculated using the invert level of the sewer pipe where the house connection is attached, the house connection type and connection length (if required by the house connection type), the default house connection grade, the default house connection cover and the surface level at the end of the house connection.

Try	button		

using the parameters in the panel fields, create the house connection. Also calculate a section along the house connection and draw it in the drainage default section view.

Finish	button		

end the option, remove the panel.

Step 5 Continued

After entering the appropriate values in the **House Connection Edit** panel and then selecting the **Try** button, the house connection will created.

To help check the house connection parameters, a section along the house connection will automatically be calculated and displayed in the section view given in the drainage defaults panel. Any tins or models on the section view will be included in the section.

Step 6 Modifying the Connection

If any of the fields in the **House Connection Edit** panel are changed, selecting the **Try** button will modify the house connection and redraw the section in the default drainage section view.

Editing A House Connection

To edit an existing house connection, the **Edit** button is selected from the **House Connections** menu and then the appropriate house connection selected.

The **House Connection Edit** panel will then be placed on the screen with the details of the selected house connection.

If any of the fields in the **House Connection Edit** panel are changed, the **Try** is used button to modify the house connection and redraw the section in the default drainage section view.

22.7.10.6.3 Delete

The **Delete** option is used to delete a connection.

After selecting the option, the mouse is used to pick the house connection to be deleted.

Once a house connection has been selected and accepted, it is deleted and removed from the screen.

The **Delete** option is terminated by selecting **Cancel** from the **Pick Ops** menu, by selecting a new option from the **Drainage Edit** menu.

22.7.10.6.4 Delete all

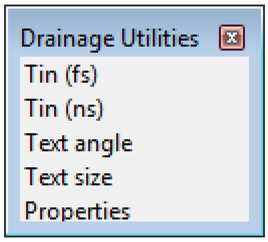
The **Delete all** option is used to delete all connections.

After selecting the option, all the connections are deleted and the option terminates.

22.7.10.7 Utilities

The options in the **Utilities** menu are used to modify default drainage tin, the name, colour, style and start chainage of the drainage string, and the size, angle, and offsets used when displaying the text for the names of all the manholes.

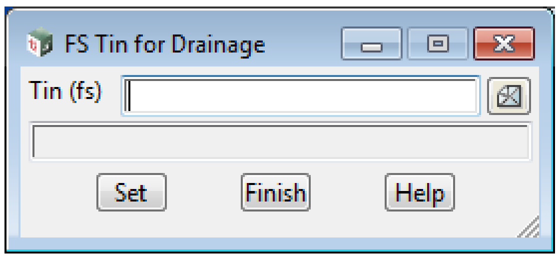
The **Drainage Utilities** walk-right menu is



22.7.10.7.1 Tin (fs)

The **Tin (fs)** option changes the default finished surface tin (tin (fs)) for the drainage string.

After selecting the option, the **FS Tin for Drainage** panel is displayed.

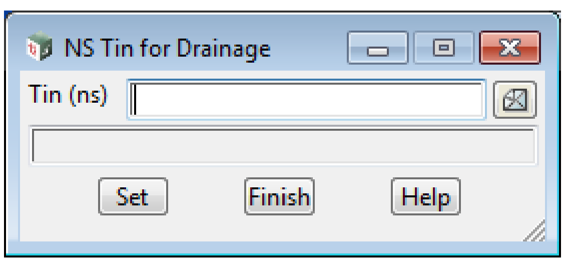


The existing default tin is displayed in the tin panel field. If a new tin is required, simply enter the new tin name into the tin (fs) field and select the **Set** button.

22.7.10.7.2 Tin (ns)

The **Tin (ns)** option changes the default natural surface tin (tin (ns)) for the drainage string.

After selecting the option, the **NS Tin for Drainage** panel is displayed.



The existing default tin is displayed in the tin (ns) panel field. If a new tin is required, simply enter the new tin name into the tin field and select the **Set** button.

22.7.10.7.3 Angle

The **Angle** option changes the angle of the text for manhole labels.

After selecting the option, an enter angle typed-input box is displayed on the screen containing the current text angle.

The new angle is entered into the typed-input box, terminated with <enter>. The typed-input box then disappears.

The **Angle** option then terminates.

22.7.10.7.4 Size

The **Size** option is used to change the size of the text for manhole labels.

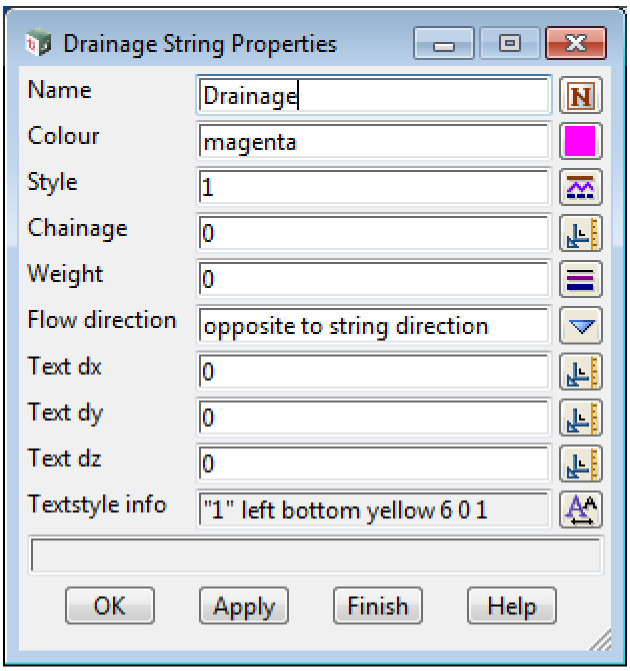
After selecting the option, an enter value typed-input box is displayed on the screen containing the current text size.

The new size is entered into the typed-input box, terminated with <enter>. The typed-input box then disappears.

The **Size** option then terminates.

22.7.10.7.5 Properties

The **Properties** option brings up the **Drainage String Properties** panel for the current drainage string.



Any of the data in the panel fields can be modified and then set for the drainage string by selecting the **OK** or **Apply** button.

22.7.10.8 Z Float

The status of **Z Float** is used each time a new manhole is created.

If **Z Float** is set to tick, then the top of the manhole is set to z float which means that automatically sits on the default drainage tin.

If **Z Float** is not set to tick, then the top of the manhole is given a set z value when it is created.

The z float status of the manhole can be changed after it has been created by using the **Drainage Edit=>Manhole=>Z Float** option.

22.7.11 Utility String Editor

Position of option on menu: **Design =>Drainage-Sewer =>Utility String Editor**

This editor is used to edit properties of the strings used by the DNE. Properties always change at an existing vertex.

Cross Section Strings (Manning's n) - set the left and right bank n values. the centre n value is assigned by the DNE.

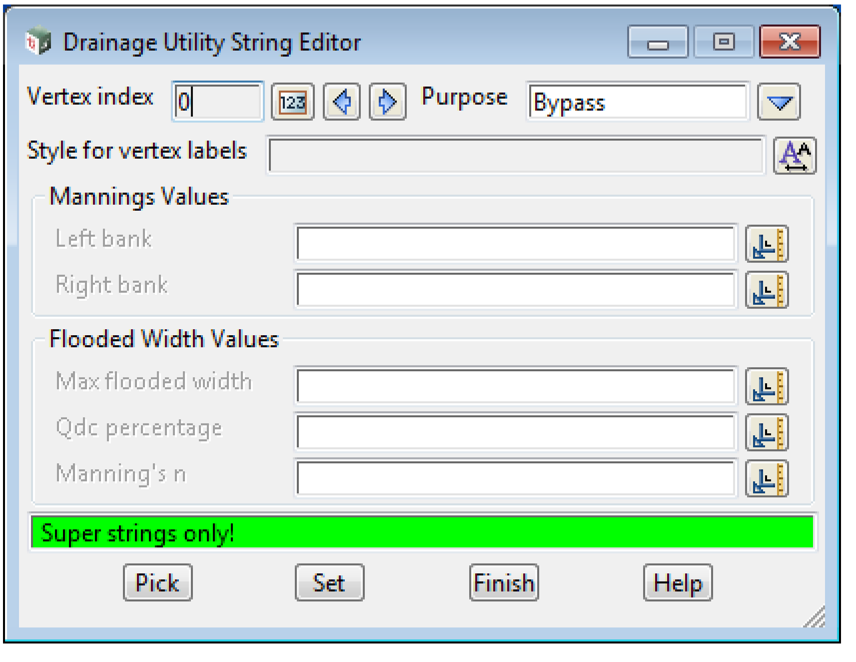
Flooded Width Values on Bypass Flow Strings - the default setting of the bypass flow strings when used to calculate flooded widths in the Drainage Analysis.Changes are in effect until the end of the string or it has been re specified at vertex at a higher chainage.

Vertex labels are created whenever properties are set (Texstyle is required).

Usage

First the string is selected at the vertex where the values are to be assigned. Next select the purpose of the string (cross section or bypass) to unlock the appropriate fields. A textstyle favourite is required as the vertex is labelled with the assigned values. Enter the values into the fields and then select Set to set the values as vertex attributes and create the label as a vertex annotation.

On selecting the **Utility String Editor** option, the **Drainage Utility String Editor** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Vertex index	vertex	selected vertex	

Once the string is selected use this to move between vertices

Style for Vertex Labels textstyle favourite textstyles

A label is created on the vertex using this textstyle favourite

Left bank Input

mannings n value, to delete clear and select set

Right bank Input

mannings n value, to delete clear and select set

Max Flooded with Input

This changes the threshold were warning bars are created during flooded width calculations. It remains in effect till the next change or the end of the bypass flow string.

Qdc percentage Input

This changes the percentage of the Qdirect+Qcatchment that is used to interpolate the discharges during flooded width calculations. It is generally set on the first vertex past upstream pit and remains in effect till the next change or the end of the bypass flow string. The flow changes from the bypass flow at the upstream pit to this value/100(Qdirect+Qcatchment).*

Manning's n Input

This changes the Manning's n value used for during flooded width calculations. It remains in effect till the next change or the end of the bypass flow string.

Pick button

Use this button to select the string. Select near the vertex you want assign the values to.

Set button

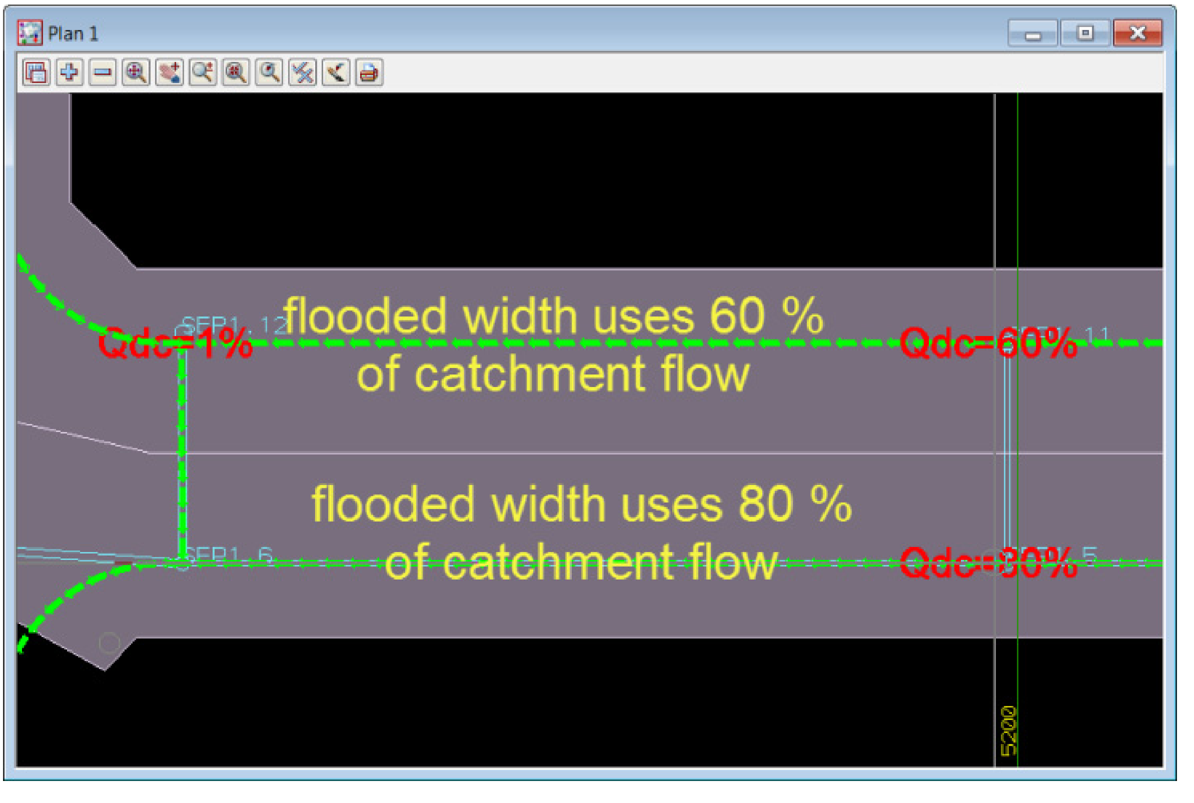
Creates the attributes and the label on the vertex.

Finish button

Removes the panel from the screen.

Help button

Launches the 12d help

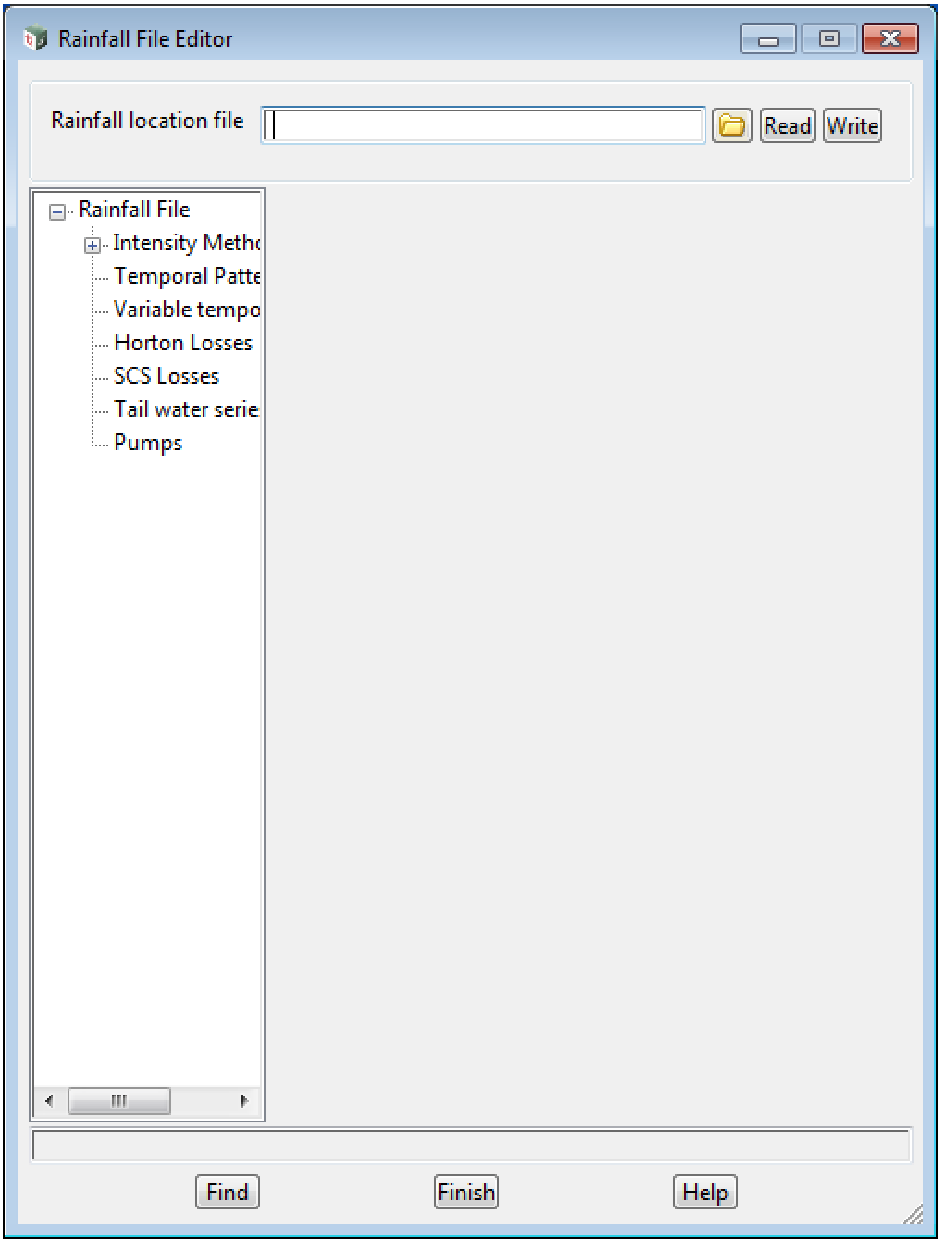


In the example above, The Qdc is set to 80% at the eastern end of the catchment. The Qdc is set to 2% for the water overtopping the road. The 18% of the area is not considered large enough to do the flooded width calculation for. If desired, another bypass flow string approaching from the east could be drawn but the final bypass string to the west (direction during bypass should remain).

22.7.12 Rainfall File ppf Editor

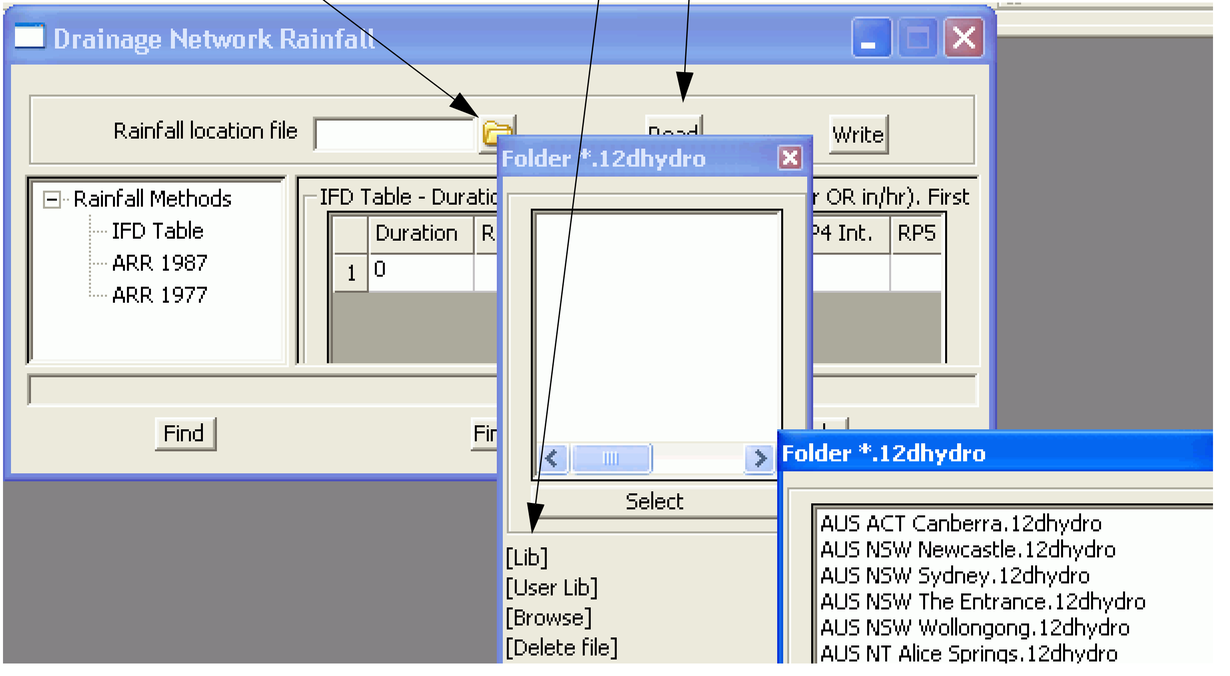
Position of option on menu: **Design =>Drainage-Sewer =>Rainfall editor**

On selecting the **Rainfall editor** option, the **Rainfall File Editor** panel is displayed.



Data is entered using one (or more if desired) input methods and then saved by entering a **Meteorology file** name and selecting **Write.** The standard 12d system file search paths are used (project folder, user library folder and then library folder).

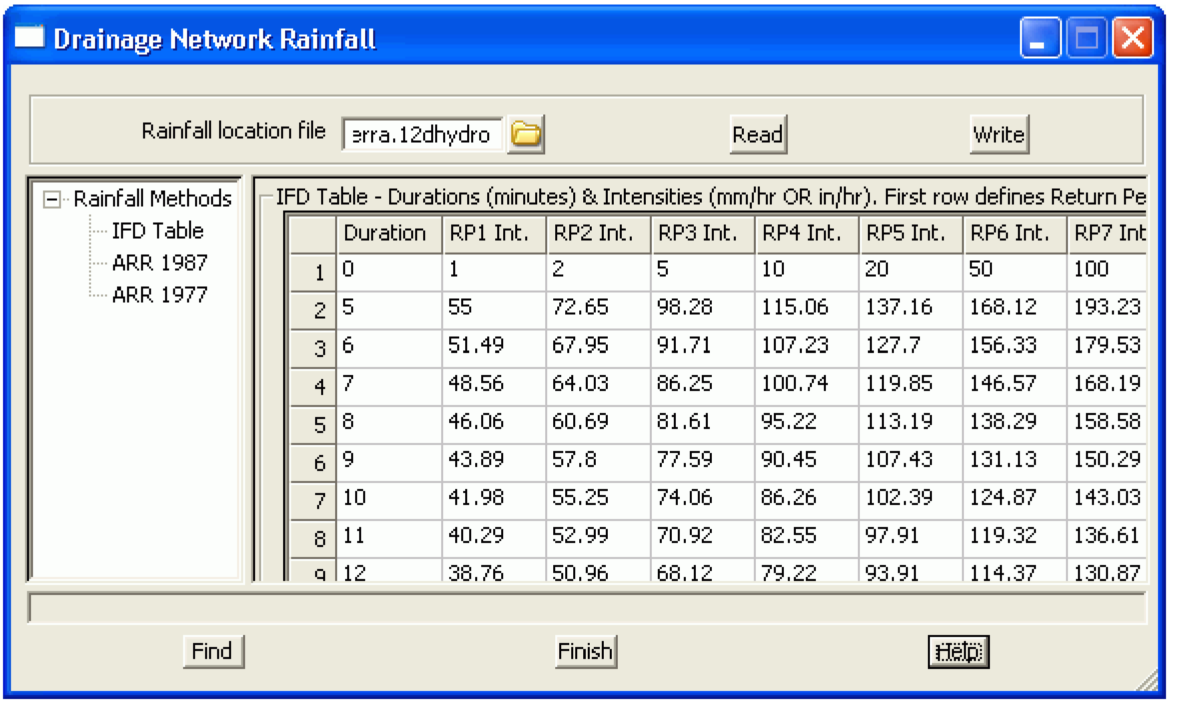
Select the folder icon and then walk right on the **Lib** item to display a list of sample files.
Select a file the select **Read. YOU MUST SELECT THE READ BUTTON!**



22.7.12.0.1 IFD Tables

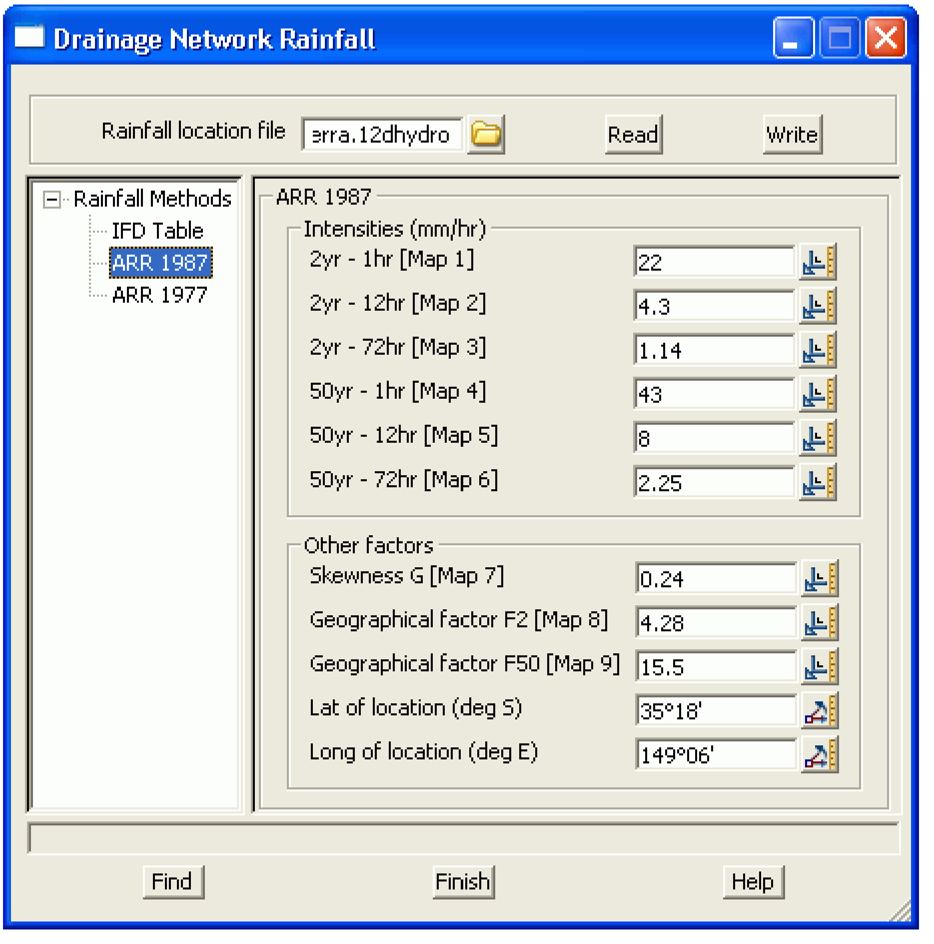
IFD tables are often available from meteorological services. The table input format follows. The first row is used to define up to 9 return periods and the following rows list the rainfall intensities for the duration entered in the first column.

Hint: to increase the size of the grid control select another method, ARR 1987 for example, and then select IFD table again.



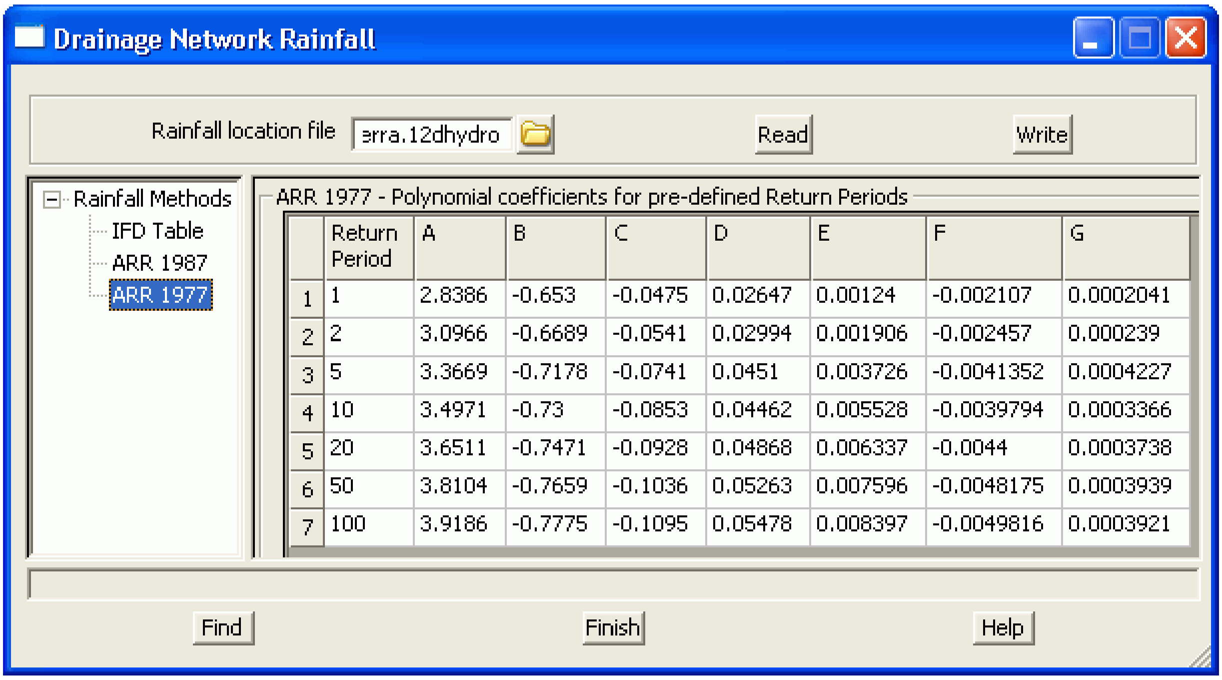
22.7.12.0.2 Australian Rainfall and Runoff 1987 Method

The rainfall intensities and other factors from Volume 2 of ARR 1987 are entered in this table.

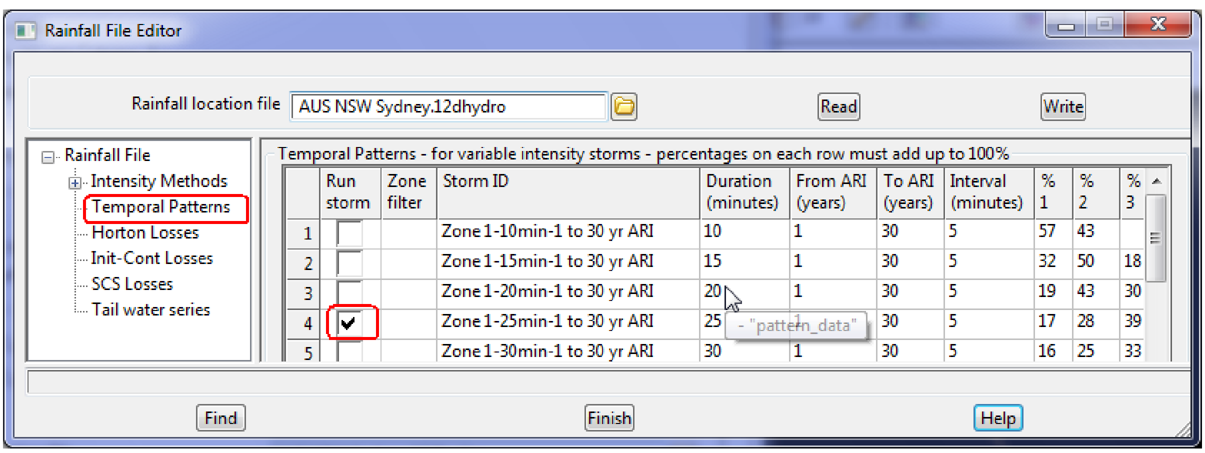


22.7.12.0.3 Australian Rainfall and Runoff 1977 Method

The seven coefficients for each return period from ARR 1977 are entered in this table.



22.7.12.1 Rainfall Temporal Patterns



Temporal patterns are referred to as storms in dynamic drainage. Several example hydro files are included in the 12d library. These examples have the temporal patterns for the minor and major storms in the 8 zones of Australia.

Temporal patterns are not required for the SCS NZ method as the standard 24 hour temporal pattern from TP108 is built into the 12d analysis engine.

The **Run storm**, **Zone filter, From ARI** and **To ARI** columns are used determine which storms are analysed (run). The **Run storm** column must be checked for that temporal pattern to be analsyed. Many storms may be selected.

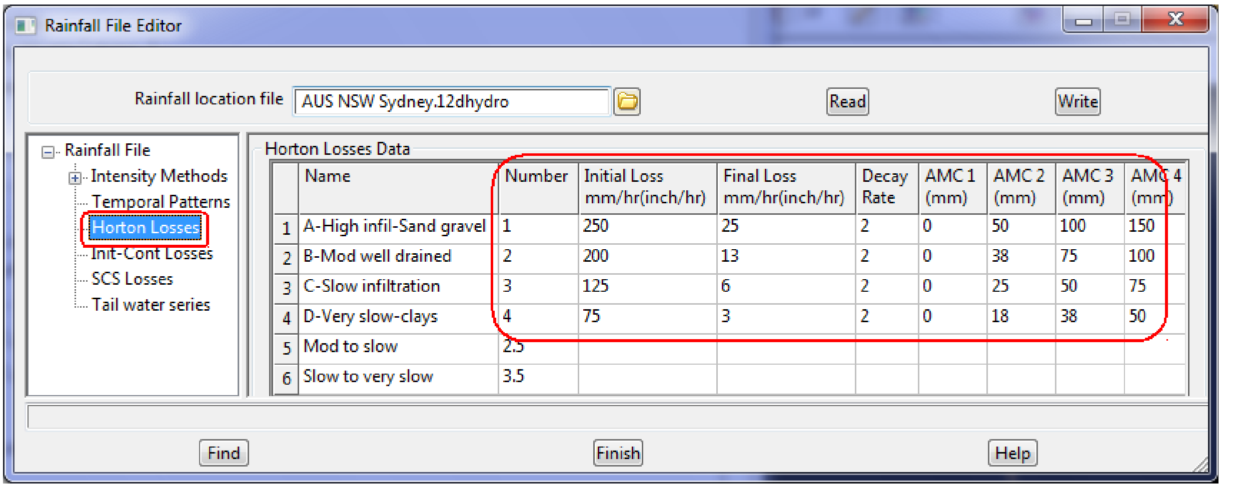
The **Zone filter** is optional. Entering a value here will allow the selected storms to be further filterd. A **Zone filter** field (accepts wild card characters) is found on the DNE Global tab that is used to determine which of the selected storms (paragraph above) are analysed.

The **ARI** field on the Run panel is used with the **From ARI** and **To ARI** columns. The value on the run panel must be within the From-To range for the storm to be analysed.

The **Duration** column determines the total length of the storm. This value divided by the **Interval** must be a whole number and this number determines the number of **%** values to be entered to the right of the **Interval column.** The total of the percentage must equal 100.

22.7.12.2 Horton Losses

The pervious portion of the catchments used in the **ILSAX 2** analysis will have a loss type defined describing the soil type. The loss type is defined in the DNE Default->catchment and catchment tabs.



These soil types use the classifications of Terstriep and Stall (1974), based on the system developed by the U.S. Department of Agriculture. The default values entered from the library represent the soil types of

1. Type A - low runoff potential, high infiltration rates (consists of sand and gravel
2. Type B - moderate infiltration rates and moderately well-drained
3. Type C - slow infiltration rates (may have layers that impede downward movement of water)
4. Type D - high runoff potential, very slow infiltration rates (consists of clays with a permanent high water table and a high swelling potential)

Numbers are assigned to each soil type to allow interpolation between the defined soil types. When interpolated values are used they must be included in the list (2.5 and 3.5 for example). Interpolated values do not need loss data entered. If loss data is entered for the interpolated names then this data will be used rather than an interpolation occurring. If any loss data is entered then all of the values must be entered.

Four preset AMC points are defined in the rainfall file to mark AMC conditions ranging from dry (AMC1) to saturated (AMC4). The required data for each line is the **Initial loss rate, Final loss rate, decay rate** and 4 antecedent moisture conditions **(AMCs).** The AMC values are entered in depth of rainfall (mm) they represent the total rainfall prior to the start of the temporal pattern.

The AMC point numbers are set once for all catchments on the DNE Global tab. Value between 1 and 4 (decimal value are permitted) are entered for the minor and major events.

22.7.12.3 SCS NZ Losses (Initial Abstraction and Curve Numbers)

The SCS NZ method uses Initial abstraction (Ia) and the curve number (CN) to determine the losses for the catchments. Names are given to the SCS curve numbers in the rainfall file. These names and Ia (entered as storage values) and selected in the DNE catchment data.

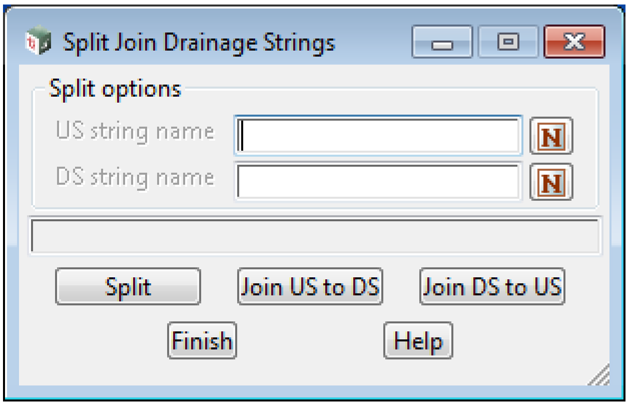
A curve number of 0 results in zero runoff while a CN=100 results in 100% runoff. TP 108 recommends the selection of the curve number by identifying 1) the soil type and 2) the land use. A CN=98 an Ia=0 are recommended for impervious areas.

22.7.13 String split/join

Position of option on menu: **Design =>Drainage-Sewer =>String split/join**

This option must be used on drainage strings instead of the standard split or join commands.

On selecting the **String split/join** option, the **String split/join** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
US string name	input		

The upstream section of the split string is assigned this name.

DS string name	input		

The downstream section of the split string is assigned this name.

Split	button		

The split may only occur at a manhole. The upstream and downstream sections are renamed if names are provided above.

Join US to DS	button		

The upstream segment must be selected first and then the downstream. The properties from the upstream string are used for the new string created. If there is a gap in between the joined strings, a pipe will be inserted with the default pipe properties. The attributes of the upstream pit on the downstream string will be discarded.

Join DS to US	button		

The downstream segment must be selected first and then the upstream. The properties from the downstream string are used for the new string created. If there is a gap in between the joined strings, a pipe will be inserted with the default pipe properties. The attributes of the downstream pit on the upstream string will be discarded.

Finish	button		

remove the panel from the screen

Help	button		

display the help for this panel

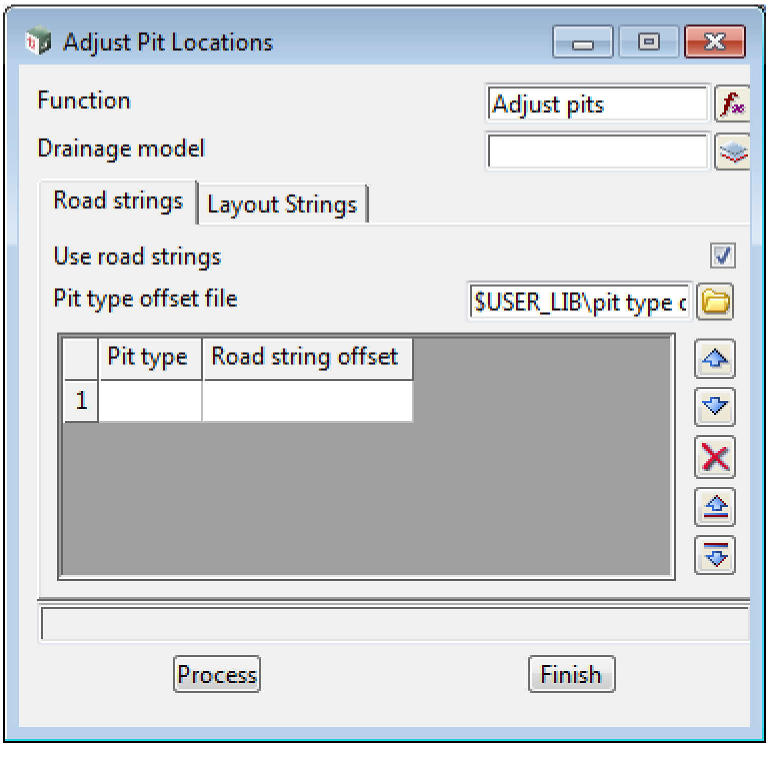
22.7.14 Adjust Pit Locations

Position of option on menu: **Design =>Drainage-Sewer =>Adjust Pit Locations**

Key Points

1. Pits are moved perpendicular to the road string.
2. The string to be moved to must be closer than the **search distance** or it will not be moved.

On selecting the **Adjust pit locations** option, the **Adjust Pit Locations** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Function function

This function will re adjust the drainage pits

Drainage model model

The drainage model to have the pits moved.

Use road strings tick

*When selected, a DNE road string is checked for each pit. When found, the pit will be moved perpendicular to this string at the **Road string offset** distance. If the road string is not found, the layout string will be searched for using the **Max layout search distance**. If not selected the only the layout strings with an offset of zero will be used.*

Pit type offset file file

*The grid data below is stored in this file when the **Process button** is selected.*

Pit type input

*Only used when **Use road strings** is selected. Each pit scans for its pit type in the **Pit type** column. When found, the pit is placed at the **Road string offset** distance from the road string.*

Road string offset input

*Only used when **Use road strings** is selected. The pit is placed at the **Road string offset** distance from the road string. Positive is away from the road centreline and negative is towards the centreline. When no **Road string offset** is set via the **pit type**, the offset value is zero.*

Data Source of Pit Layout Stringssourcemodel

*Used when **Use road strings** is NOT selected or when no road string is found. As road strings are often in several models, so the filter option is usually the best. Select a view with the road string models and then use the string info tab and name field (* wild card can be used) to select the strings to move to.*

Max Layout Search Distancereal 1.0

Only used for layout strings. Road strings do not have a search distance. If the closest string in the pit layout strings is farther than this distance the pit will not be moved.

Run button

This moves the pits horizontally. Undo is available

Finish button

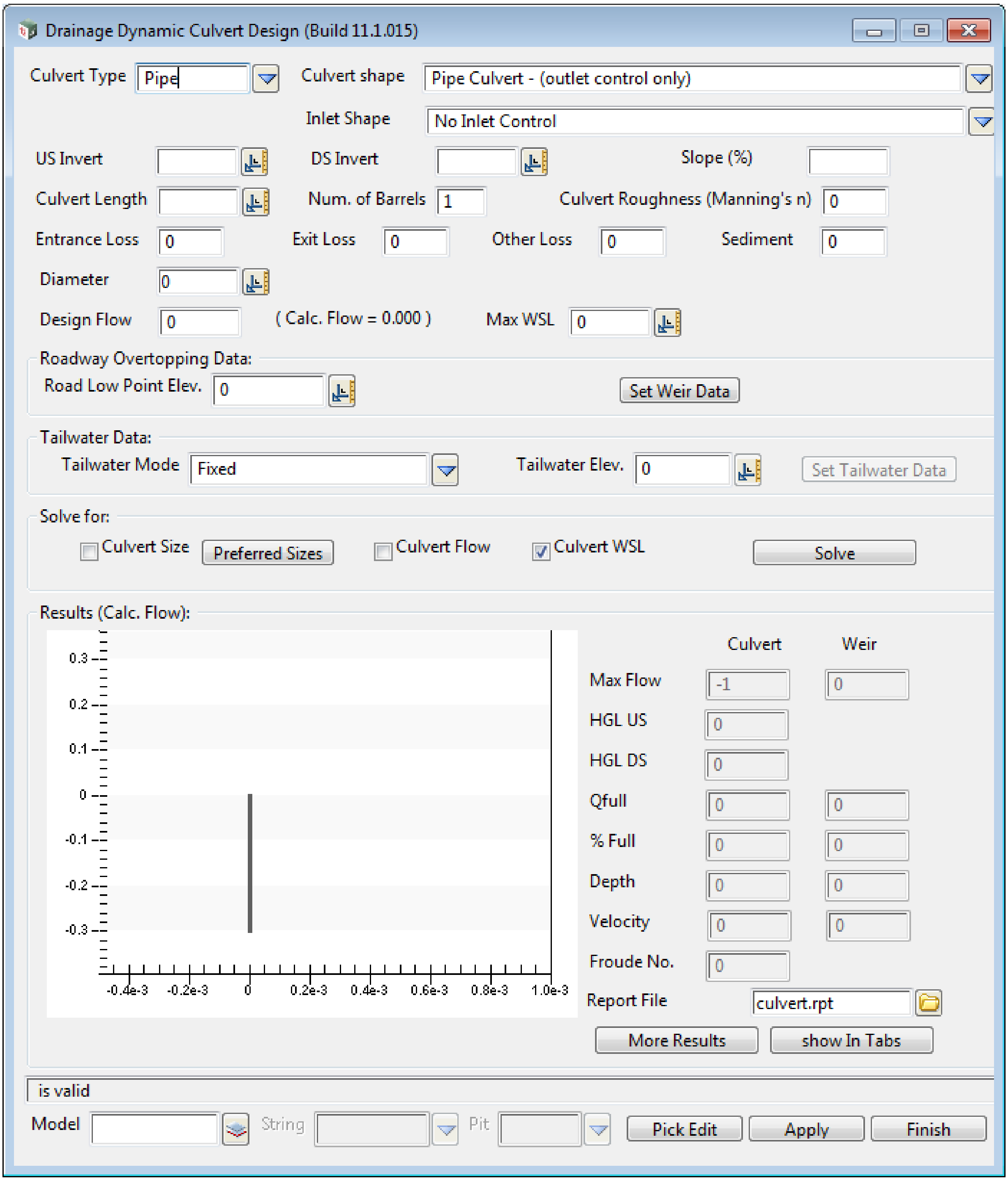
Removes the panel from the screen.

22.7.15 Dynamic Culvert Design

Position of option on menu: **Design =>Drainage-Sewer =>Dynamic Culvert**

This section of documentation is a work in progress and will be updated in subsequent releases.

On selecting the **Dynamic Culvert** option, the **Drainage Dynamic Culvert Design** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Culvert type	choice box	Pipe	

Culvert shape	choice box	Pipe Culvert - (outlet control only)
Inlet Shape	choice box	No Inlet Control
US Invert	measure box	
DS Invert	measure box	
Slope (%)	input	
Culvert Length	measure box	
Num. of Barrels	input	1
Culvert Roughness (Manning's n)	input	
Entrance Loss	input	
Exit Loss	input	
Other Loss	input	
Sediment	input	
Diameter	measure box	
Design Flow	input	
Max WSL	measure box	
Road Low Point Elev.	measure box	
Set Weir Data	button	
Tailwater Mode	choice box	
Tailwater Elev.	measure box	
Set Tailwater Data	button	
Culvert Size	tick box	
Preferred Sizes	button	
Culvert Flow	tick box	
Culvert WSL	tick box	
Solve	button	
Max Flow	input	
HGL US	input	
HGL DS	input	
Qfull	input	
% Full	input	
Depth	input	
Velocity	input	
Froude No.	input	
Report File	file box	
More Results	button	
Show in Tabs	button	

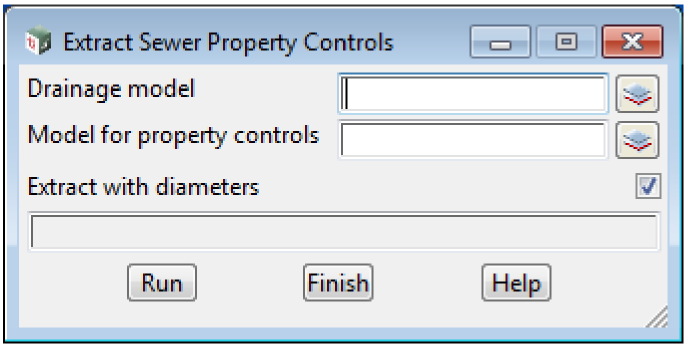
Model	model box
String	choice box
Pit	choice box
Pick Edit	button
Apply	button

22.7.16 Extract Sewer Property Controls

Position of option on menu: **Design =>Drainage-Sewer =>Extract Sewer Property Controls**

The sewer property control strings are a sub string of the drainage string and therefore may only be profiled using a right mouse click of the profile button. To include these control strings on plots or export to other packages they need to be converted to super strings.

On selecting the **Extract Sewer Property Controls** option, the **Extract Sewer Property Controls** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Drainage model	mode box		

All drainage strings in this model will have their property controls strings duplicated as super strings.

Model for property controls	model box		

The super strings for the property controls are placed in this model.

Extract with diameters	tick box	on	

The super strings will have the constant pipe dimension set to the control diameter.

Run	button		

Create the super strings representing the property controls.

Finish	button		

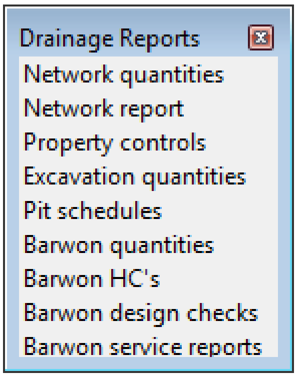
Remove the panel from the screen.

22.8 Reports

Position of menu: **Design =>Drainage-Sewer=>Reports**

Reports menu contains an option to report on the property controls for the drainage strings and an option produce network quantities.

The **Reports** walk-right menu is



For the option *Network quantities*, please continue to the section Drainage Network Quantities.
Network report, please continue to the section Network Report.
Property control, please continue to the section Property Controls.
Excavation quantaties, please continue to the section Excavation Quantities.
Pit schedules, please continue to the section Pit Schedules.
Barwon quantities, please continue to the section Barwon Quantities.
Barwon HC's, please continue to the section Barwon House Connections.
Barwon design checks, please continue to the section Barwon Design Checks.
Barwon services reports, please continue to the section Barwon Services Report

22.8.1 Drainage Network Quantities

Position of option on menu: **Design => Drainage => Reports=> Network Quantities**

The **Network quantities** report contains information about the manholes and pipes that make up the selected drainage strings. Configuration files allow the user to specify the depth ranges and sizes of pipes to report on.

This option creates quantity tables for manholes, pipes and house connections.

The manholes/pipes/house connections are summarised by user defined depths and types.

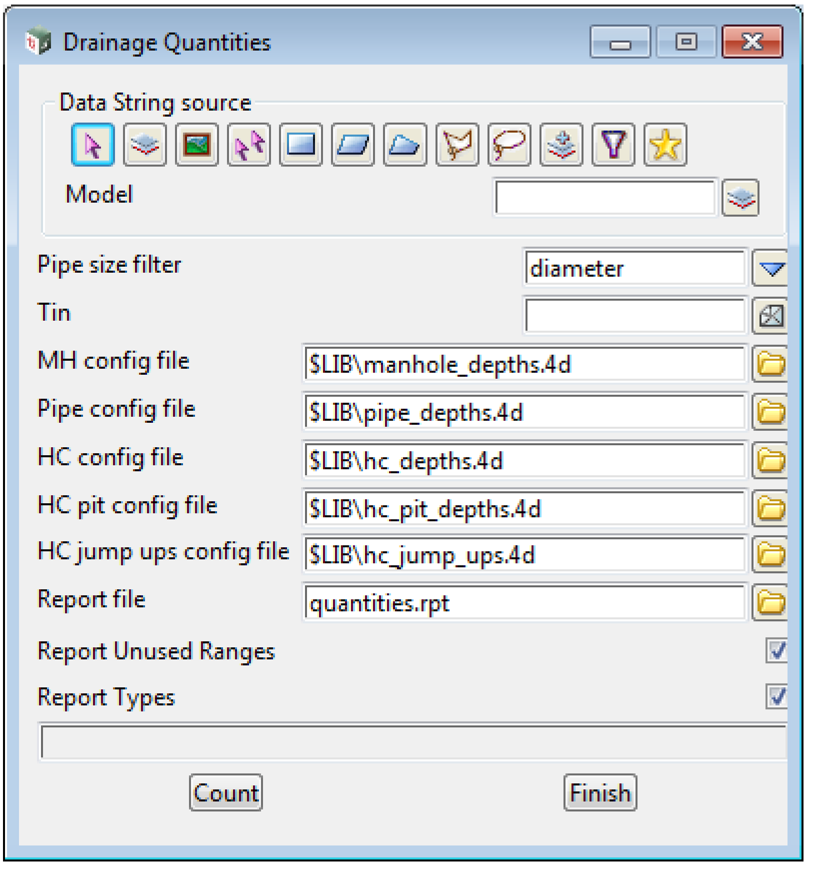
See Also

Drainage overview

Key points

1. Items are counted/totalled by depth and optionally type.
2. The routine will not "double count" items even if the ranges overlap.
3. Types are case sensitive, types with spaces in the name must be enclosed in quotes and the wild card * may be used.
4. Use vertically offset tins and "banded" depth ranges to get quantities under roads, foot paths, *etc.* This is discussed later in detail.
5. Erase count file fields if the items are not to be counted.

On selecting the **Network quantities** option, the **Drainage quantities** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Data String Source Choice

usually the entire model is selected but view is also available for combining models

Pipe size filter Choice diameter or pipe size attribute

The second item in a count line is used to filter by ***diameter*** *(in meters/ft) or the* ***pipe size attribute****. The* ***diameter*** *can only be used when no box culverts or trapezoidal channels are used. The* ***pipe size attribute*** *is the label generally used in the plan plots. This is the pipe size in mm/inches (375), for box culverts width x height (750x375) and for trapezoidal channels TopwidthBottomwidthxHeight (T5000B1000x500).*

Tin tin box

This tin will be used for the pipe and pit depths.

MH config file file box

This file specifies the types and depth ranges for the pits. Details of this file are contained below.

Pipe config file file box

This file specifies the types and depth ranges for the pipes. Details of this file are contained below.

HC config file file box

This file specifies the types and depth ranges for the house connections. Details of this file are contained below.

HC pit config file file box

This file specifies the types and depth ranges for the HC pits. Details of this file are contained below.

HC jump ups file file box

This file specifies the types and depth ranges for the house connections jump ups. Details of this file are contained below.

Report file file box

a sample report file is given below.

Report unused ranges tick box

*the depth ranges for the pit/pipe/house connections are defined in the *.4d files. Selecting this option will cause the depth ranges in the file to be printed even if there are no pit/pipe/house connections in these depth ranges (zero quantity values will be shown).*

Report types tick box

*Selecting this option will cause the pit/pipe/house connection types used in the model types to be listed (even if quantities are not requested in the *.4d files). Since this is a complete of the type used in the model, the list informs the user what types have not been included in the quantity calculation.*

Count button

executes the option.

Finish button

removes the dialogue from the screen

The *.4d files listed above are contained in the 12d **library** directory. Each line is the file performs a count (count lines). No items are counted twice. Therefore, if an item is counted its type and then a count line is found the wild card is used for the type, the type already counted will not be included in the count.

The format for a count line is three or four values (space delimited) per line. Size is optional.

```
<type (from drainage.4d)>  <size> <starting depth>   <ending depth>
```

Notes:

All **types** with spaces in the name must be enclosed in quotes The wild card * may be used.

The **size** is optional and if omitted the all sizes will be counted in this group (do not use the * for a wild card).

The **starting depth** and **ending depth** are required for all count lines.

Quantities Under Roads and Footpaths

By creating super tins with vertically offset sections, quantities under roads, footpaths, *etc.* can be determined. for example.

Offset your road design tin up by 1000m (**Tins->Utility->Translate/Copy)** and then use the depth range 1000-1999 for pipes under roads.

Create a tin from the footpaths only, null by angle length with a small length to remove the road and then offset it vertically by 2000m. the depth range 2000-2999 is not the quantities under the footpath.

Sample count lines

```
// sum concrete cover manholes is various ranges

"CONC COVER" 0.0 1.6
"CONC COVER" 1.5 3.0
"CONC COVER" 3.0 999.9  // this is expected to be zero
"CONC COVER" -999.0 0.0 // trap errors

// any that are not Concrete cover will be counted here

* 0.0 1.6
* 1.6 3.0
* 3.0 999.9
```

```
Manhole Quantities
====================================


CONC COVER          0.00    1.60    13      16.506
CONC COVER          1.60    3.00     1       1.510
CONC COVER          3.00   999.9     0       0.000
CONC COVER        -999.0     0.0     0       0.000

*                     0.00  1.60     0       0.000
*                     1.60  3.00     0       0.000
*                     3.00 999.9     0       0.000

  total length = 18.016


  Types Used
  ----------

  CONC COVER

  Diameters Used
  ----------

  1.100
```

Since the **Report unused ranges** tick box was selected, these lines were printed even though there were no pits in the data ranges.

This data results from selecting the **Report types** tick box.

Sample count lines for pipes follow.

```
// sum class 2 pipes by diameter and for various ranges

// count 375

2 0.375 0.0 2.0
2 0.375 2.0 5.0
2 0.375 5.0 999.

// count 450

2 0.450 0.0 2.0
2 0.450 2.0 5.0
2 0.450 5.0 999.

// count 525

2 0.525 0.0 2.0
2 0.525 2.0 5.0
2 0.525 5.0 999.

// count pipe sizes that were missed

2 * 0.0 2.0
2 * 2.0 5.0
2 * 5.0 999.

// count all other missed pipes

* 0.0 999.
```

22.8.2 Network Report

After selecting the **Network report** option, the **Report on Selected Items** panel is displayed.

This is the same as the option **Reports** =>**Co-ord/ Brd-dst** on the main menu.

For the given model, it prints out the string information for each string in the model.

If a sewer network model is given, each sewer string in the network model is reported on.

For further information, go to Coordinates or Bearing-Distance Report.

22.8.3 Excavation Quantities

Position of option on menu: **Design =>Drainage-Sewer =>Reports=>Excavation quantities**

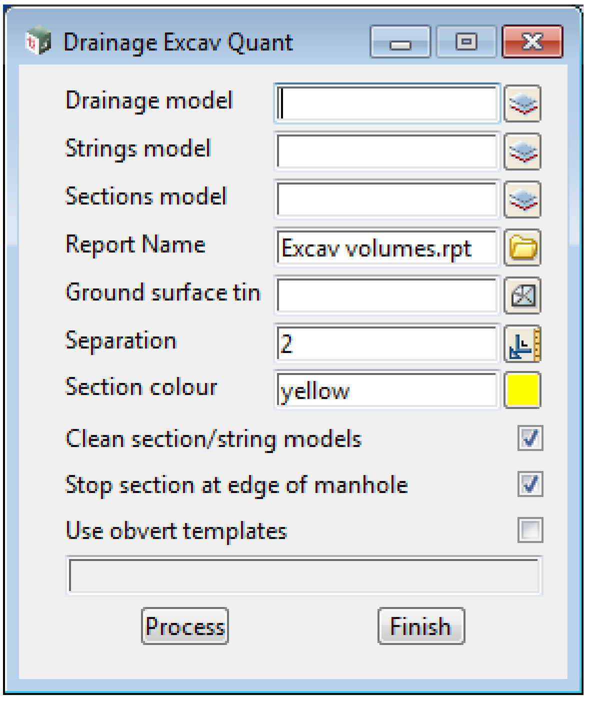
This routine uses 12d templates to calculate the excavation volume for all of the drainage strings in a model. An option to create section for a tin on top of the pipe is also available so that the drainage long sections can include hatching between the obvert of the pipe and the design tin under roads.
Templates with names set to the pipe diameters (times 1000) are used for the calculations, thus trench shapes can be customised and over excavation for bedding materials can be included. Net area calculations to exclude pipe area are not supported.

Key points

1. One template for each pipe size (mm)
2. If obvert templates are used, add the prefix "obvert " to the pipe size
3. Carefully consider the tin selected.

A template must exist for each pipe size in the model (pipe size x 1000). For example a 0.3m pipe will require a template to exist named 300. A 0.5ft pipe would require a template named 500. A sample template library is included in the 12d library in the file **pipe_template.tpl**.
The templates are run along the strings and the total volumes are reported. Volumes for each strings are given in the report file.

If a tin is created from these strings then volumes by depth can be determined using
Design=>Volumes=>Exact=>Tin to tin
On selecting the **Excavation quantities** option, the **Drainage Excavation Quantities** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Drainage model	input box		

Model to contain all of the pit and pipe network to be worked on.

Field Description	Type	Defaults	Pop-Up
Strings model	model box		

Strings generated from the templates will be stored in this model

Sections model model box

Sections generated from the templates will be stored in this model

Report name input box

cut and fill volumes will ne sent to this report

Ground Surface Tin tin box

tins from which the volumes will be calculated

Separation real box

distance between the sections

Sections colour colour box

Sections generated from the templates will be assigned this colour (strings colours are defined in the templates)

Clean section/strings modeltick box

Delete the strings in these models before processing.

Stop section at edge of pit tick box

*Template are run from pit centre to centre if this is not selected. The templates stop at the edge of the pit if selected. This is often selected with the following option **Use obvert templates.***

Use obvert templates tick box

*Templates must be named with the prefix "obvert". i.e. **obvert 300.** The template is still run along the invert of the pipe but the user now has a section "set" of templates that can be used to create a tin on top of the pipe as well as below.*

An example report file follows.

```
 ----------------------------- BEGIN APPLY TEMPLATE REPORT ------------------------------

  apply template to string report -

    string         E
    tin            design
    separation         10.000
    left template  375
    right template 375
    cut volumes and areas are negative
    fill volumes and areas are positive


  chainage-  ------sectional   information------   ------intermediate   information----      ---------accumulative
information-------
           ---------- ---cut area --fill area ------------ ----cut vol ---fill vol    -cut volume-- -fill volume- -
--balance---


    0.000              -1.434      0.000                                                0.000        0.000        0.000
                                                             -0.771       0.000
    0.550              -1.367      0.000                                              -0.771        0.000       -0.771
                                                            -14.222       0.000
   10.000              -1.642      0.000                                             -14.992        0.000      -14.992
                                                            -15.293       0.000
   20.000              -1.416      0.000                                             -30.286        0.000      -30.286
                                                             -1.845       0.000
   21.313              -1.393      0.000                                             -32.130        0.000      -32.130
                                                             -0.794       0.000
   21.863              -1.493      0.000                                             -32.924        0.000      -32.924

 total cut                        -32.924
 total fill                         0.000
 balance                          -32.924
 ie excess of cut over fill        32.924
```

22.8.4 Pit Schedules

See Also

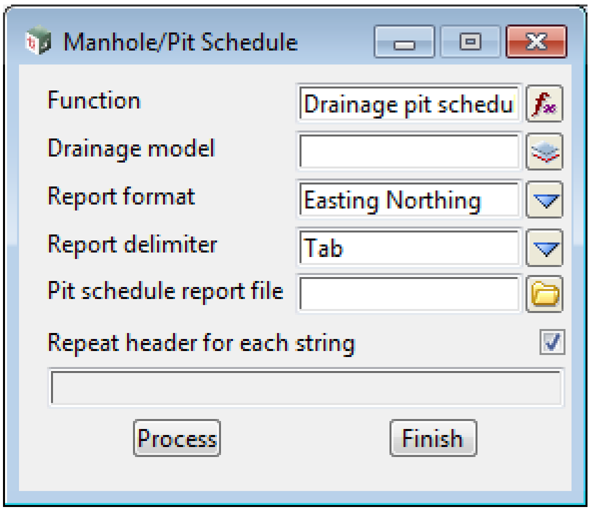
Selecting design string or tin?
Drainage overview

Usage

Position of option on menu: **Design => Drainage => Reports => Pit schedule**

This routine prints the calculations from the last time Set Pit Details was selected in the Drainage Network Editor.

On selecting the **Pit schedule** option, the **Manhole/Pit Schedule** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Drainage model name	input box	drainage network	
model containing the drainage strings			
Pit schedule file name	input box	pit report	
file to be created			
Report Format	choice box	Road chainge.,Easting...	
file format			
Data delimiter	choice box	Tab, Space	
tab delimiters are best for spreadsheets and space for some text editors			
Repeat header for each line	tick box	selected	
when selected, the column headings will be printed each drainage line			
Process	button		
Create the pit report			
Finish	button		
remove the panel from the screen			

Notes:

The columns of data may be separated by spaces or a tab. (tab is used for spreadsheet transfers). The internal width and length data are retrieved from the **drainage.4d** file for the pit type specified. If you want a longer description for the pit then the type used inside 12d this can also be entered in the drainage.4d file. The remarks for each pit are entered as user defined pit attribute named **remarks** and may be set using the attribute editor (on the drainage menu) or via a spreadsheet.

Easting Northing Sample

```
.PIT SCHEDULE
  Pit                                INTERNAL           INLET          OUT-
LET                  PIT
  No    TYPE    EASTING NORTHING    WD       LEN     DIA   INV LEV   DIA   INV LEV  FIN
RL    DEPTH         REMARKS
   B1   SA2    5302.458 7336.936  450.000  900.000                    375   28.210
29.387    1.177
   A2   SA2    5264.372 7322.036  450.000  900.000    375   27.470
28.646    1.226
   C1   SA2    5224.155 7336.936  450.000  900.000                    375   26.690
27.863    1.173
   A3   SA2    5187.910 7322.036  450.000  900.000    375   25.930
27.158    3.628
   A1   SA2    5309.458 7321.100  450.000  900.000                    225   28.550
29.577    1.027
   A2   SA2     5264.372 7322.036  450.000  900.000    225   27.470    375   27.420
28.646    1.226
   A3   SA2     5187.910 7322.036  450.000  900.000    375   25.930    375   23.530
27.158    3.628
  A4   SA2   5157.411 7321.332  450.000  900.000   375   23.090              26.714
3.624 outlet to existing system
NOTE:
1. ALL SETOUT POINTS QUOTED TO CENTRE OF PIT
```

Road Chainage Offset Example

DRAINAGE LINE A

PIT	PIT LOCATION		LOCATION OFFSETS	
No.	EASTING	NORTHING	STATION	CTRLOFFSTYPEREMARKS
A/1	5354.629	7336.936	231.171	d002-7.450
A/2	5340.691	7320.911	217.233	d0028.575
A/3	5293.458	7320.886	170.000	d0028.600
A/4	5250.131	7320.886	126.673	d0028.600
A/5	5217.194	7322.036	93.736	d0027.450
A/6	5183.458	7322.036	60.000	d0027.450
A/7	5152.699	7322.036	29.241	d0027.450

Notes

The Set pit details must be run at least once to before printing the report. If the pits are moved or the designed strings changed then this option must re rerun.

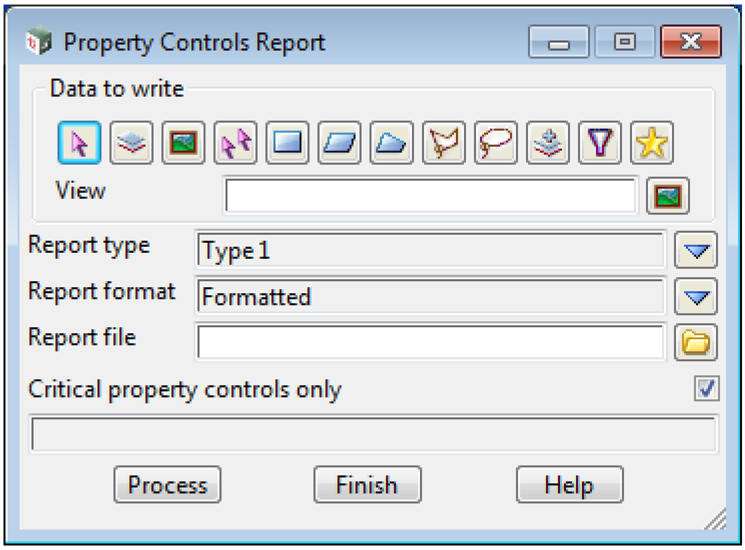
The easting northing data obtained for the **road design string** option is obtained by dropping the pit centre perpendicular onto the selected road design string. This data is stored as pit attributes **setout x** and **setout y**. It is calculated when the **Set Pit Detail** is selected in the Drainage Network Editor.

22.8.5 Property Controls

Position of menu: **Design =>Drainage-Sewer=>Reports=>Property Controls**

The property controls report contains information about all the property controls for the selected drainage strings, and if required, denote the critical property control for a lot.

On selecting **Property controls,** the **Property Controls Report** panel is displayed.



The fields and buttons used in this panel have the following functions.

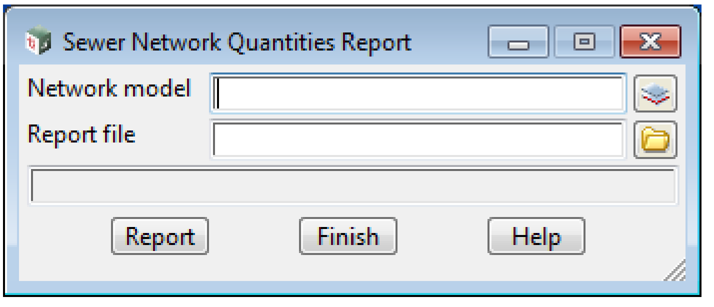
Field Description	Type	Defaults	Pop-Up
Data source type			
data source type.			
Data source			
data source for the drainage strings to process.			
Format	choice box	ascii	ascii, Excel spread sheet
format for the property control report.			
Report file	file box		
name of the file for the report.			
Critical property controls only	tick box	tick	
if tick, only the critical property controls are reported.			
Process	button		
run the option.			

22.8.6 Barwon Quantities

The report generated from this option includes

- the lengths of each pipe type for each line and the total length for each pipe type for all pipes in the network.
- the quantity of concrete used for the manholes in the network.
- the number and type of house connections in the network, including caps, bends and bushes.

After selecting the **Barwon quantities** option, the **Sewer Network Quantities Report** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Network model			available models

model containing the sewer strings.

Report file	file box		

name of the file for the report.

Report	button		

run the option.

22.8.7 Barwon House Connections

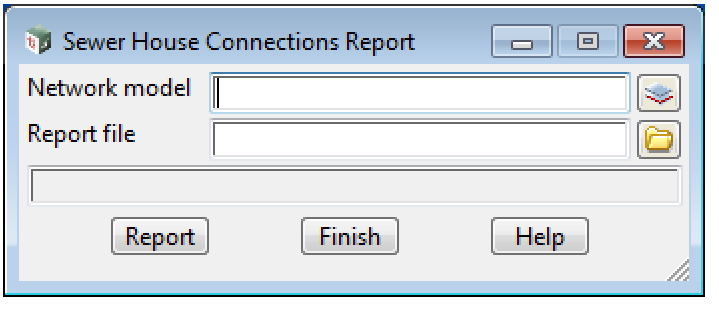
Position of menu: **Design =>Drainage-Sewer=>Reports=>House connections**

The house connections report contains information about all the house connections for the selected drainage (sewer) strings.

The house connection report includes for each house connection (branch) the

- name of the sewer line
- downstream manhole for the house connection
- lot name
- house connection number and type
- chainage of the house connection
- the invert level at the end of the house connection (IL branch)
- the invert level of the house connection at the sewer pipe (IL sewer)
- the drop over the house connection (branch depth)

On selecting **House connections,** the **Sewer House Connections Report** panel is displayed.



The fields and buttons used in this panel have the following functions.

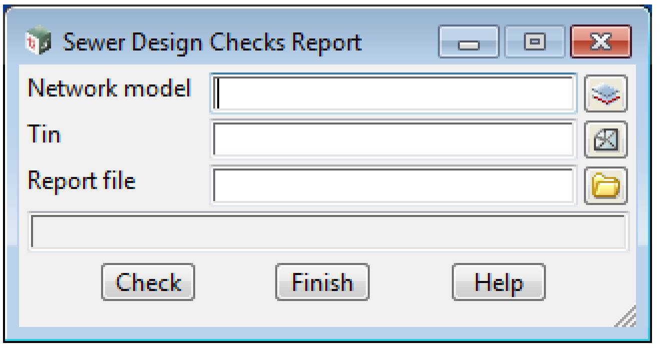
Field Description	Type	Defaults	Pop-Up
Network model			available models
model containing the sewer strings.			
Report file	file box		
name of the file for the report.			
Report	button		
run the option.			

22.8.8 Barwon Design Checks

The report generated from this option includes the design checks

- network validations (checks that there are strings, no closed loops)
- manholes don't have negative drops
- other sewer lines connecting in don't have negative drops
- sewers flows downhill with a minimum grade
- manholes are not too close together (*i.e.* on top of each other)
- a minimum cover for each sewer line
- block controls are above the pipe invert level

After selecting the **Design checks** option, the **Sewer Design Checks Report** panel is displayed



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Network model			available models

model containing the sewer strings.

Tin			available tins

tin used for checking minimum cover against.

Report file	file box		

name of the file for the report.

Report	button		

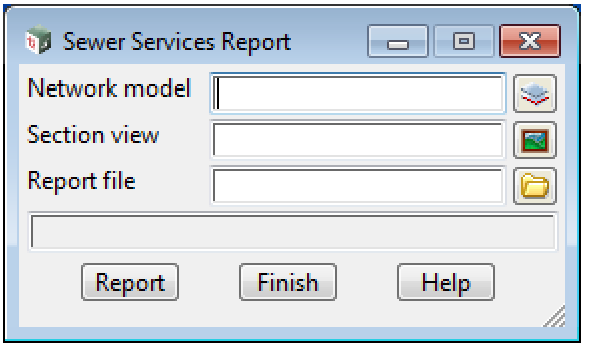
run the option.

22.8.9 Barwon Services Report

For each sewer string in the network, this option generates a report which includes the

- section through any tins on the section view
- name and model of any services in the corridor defined by the section view
- co-ordinates and chainages of the parts of the service in the corridor, and the chainage and offset for each of the point of the parts projected onto the sewer centre-line.
- clearance at the point where any service goes under or over the sewer string.

After selecting the **Services report** option, the **Sewer Services Report** panel is displayed



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Network model			available models

model containing the sewer strings.

Section view			available section views

section view defining the corridor and service models.

Report file	file box		

name of the file for the report.

Report	button		

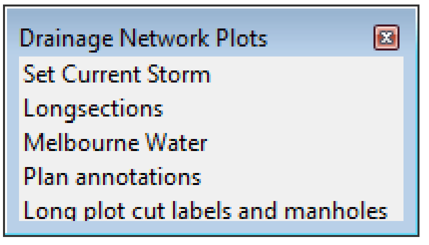
run the option.

22.9 Drainage Plots

Position of menu: **Design =>Drainage-Sewer=>Plots**

Currently the **Plots** menu contains an option to produce a long section including any HGL values, flows and user defined attributes read in from other sources, and a second option to output the network to Melbourne Water’s sewer format.

The **Plots** walk-right menu is



The options **Longsections** and **Melbourne Water** will now be described in more detail.

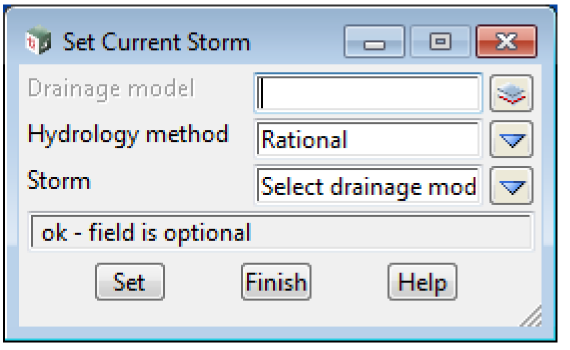
For the option *Set Current Storm*, please continue to the section Set Current Storm.
Long sections, please continue to the section Drainage Longsections.
Melbourne Water, please continue to the section Melbourne Water.
Plan annotations, please continue to the section Plot Annotations.
Long plot cut labels and manholes, please continue to the section Long Plot Cut Labels and Manholes.

22.9.1 Set Current Storm

Position of menu: **Design =>Drainage-Sewer=>Plots=>Set Current Storm**

This section of documentation is a work in progress and will be updated in subsequent releases.

Selecting **Set Current Storm** brings up the **Set Current Storm** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Drainage model	model box		available models
Hydrology method	choice box		
Storm	choice box		
Set	button		

22.9.2 Drainage Longsections

Position of menu: **Design =>Drainage-Sewer=>Plots=>Longsections**

The **Longsections** option is used to generate the longsection plots for all lines in a drainage network.

Given the plot sheet size and the horizontal and vertical scales, the longsections for the drainage lines are plotted starting at the top of the sheet and moving across the sheet. Once one row is full, if there is room the plot moves down the page and begins a new row. When a plot sheet is full, a new plot sheet is automatically begun.

Hence the drainage lines are plotted one after another on one or more plotter sheets.

The drainage lines are plotted in string name alphabetical order.

The drainage longsection plot includes

- the manholes, drainage pipe and any house connections
- the height of the finished surface at the manhole
- manhole names and cover types
- distances between manholes
- the invert depth of the pipe on either side of a manhole
- the grades and types of the pipes
- any services in the corridor - including their name, invert level and distance from the nearest down-stream manhole
- if the information exists, the velocity, flow, HGL values and diagram

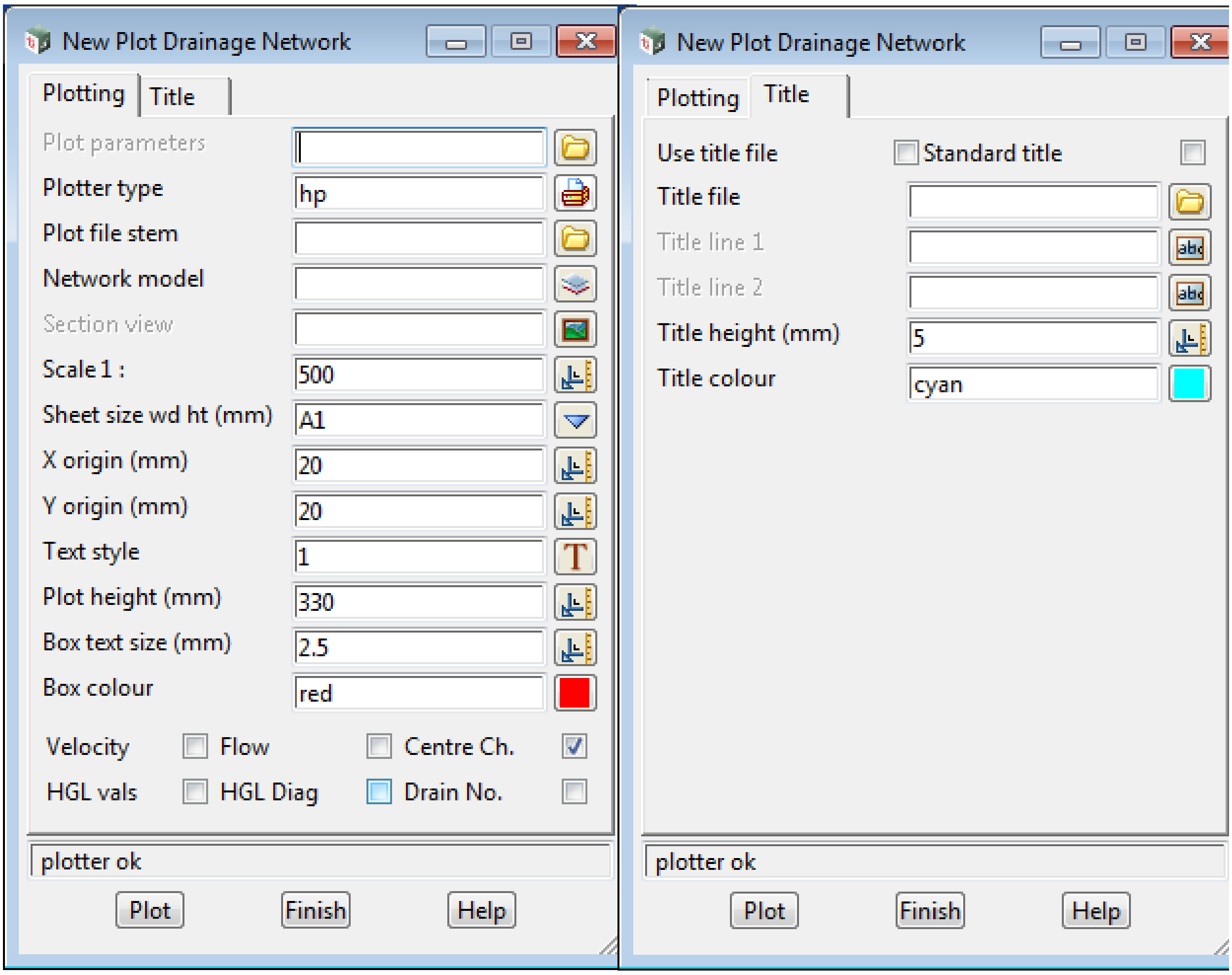
After selecting the **Longsections** option, the **New Plot Drainage Network** panel is displayed.

The drainage long section plot is tailored by using the plot parameter file (.ppf file) given in the plot parameters field. A default .ppf file is set by pointing to it with the environment variable

DRAINAGE_PPF_4D	path name of default .ppf file

In the path name to the default .ppf file, $LIB is used to stand for the library folder set by LIB_4D. For example, "$LIB/drainage.ppf" is the file drainage.ppf in the library area.

The **title** button on the bottom of this panel controls the use of a title block file and the plotting of a border and two lines of title. If the **Title** button is selected, the **drainage plot title** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Plot parameters	input		*.ppf

file of plot parameters used for extra control of the long section plot.

Plotter type	input	hp	model, windows, hp, dxf, postscript, *etc.*

format for the plot output.

Network model input

the model containing all the drainage lines for the network.

Section view input

the section view to be used to define the vertical exaggeration, corridor widths, tins to section through, services models to section, etc.

Plot file stem input

since more than one plot page may be produced, the plot file names are constructed from the stem plus a plot page sequence number, followed by the appropriate plotter type ending.

Scale 1: input

horizontal scale for plotting the drainage long section. The vertical exaggeration is taken from the section view given in the section view *field.*

Sheet size wd ht (mm) input available sheet sizes

the width and height values (separated by space) or the name of a user defined sheet size.

X origin (mm) input

the x position on the plot sheet for the bottom left hand corner of the longsection plots. Same as left_margin in the plot parameter file.

Y origin (mm) input

the y position on the plot sheet for the bottom left hand corner of the longsection plots. Same as bottom_margin in the plot parameter file.

Text style input 1 available text styles

the default text style to use in the longsection plot.

Plot ht (mm) input

the maximum allowable height for a longsection plot for a drainage line. Datum breaks are applied to any part of the longsection that will not fit into the plot ht

Box text size (mm) input 3

size (in millimetres) to plot the chainages, heights etc. in the boxes in the plots of the drainage longsections

Box colour input cyan available colours

colour used for the text and the boxes.

Velocity tick box tick

*if **tick**, the velocity values for the pipes are drawn on the longsection plot.*

Flow tick box tick

*if **tick**, the flow values for the pipes are drawn on the longsection plot.*

Centre CH. tick box tick

*if **tick**, the road centre line chainages are drawn on the longsection plot.*

HGL vals tick box tick

*if **tick**, the HGL values for the pipes are drawn on the longsection plot.*

HGL diag tick box tick

*if **tick**, lines joining the HGL values for the pipes are drawn on the longsection plot.*

Drain No. tick box tick

*if **tick**, the drainage line names are drawn on the longsection plot.*

Plot button

plot the drainage longsections for the drainage lines in the model given in the network model *field.*

The fields and buttons in the *title* tab are

Use title file tick box

*if **tick,** a user defined title block file is used.*

Standard Title tick box tick

*if **tick**, the standard 12d Model border and two lines of title are placed on the bottom of the plot*

Title file input *.tf

*if **non-blank** and* use title file *is set to **tick**, then the file given in this field is used to generate a user defined title block for the plot.*

Title line 1/2 input

first/second line of title information

Title height (mm) input 5

height (in millimetres) to draw the characters in the two lines of title information.

Title colour input cyan available colours

colour used for the border and the title information.

Please continue to the next section Drainage Longsection Plot Parameter File.

22.9.2.1 Drainage Longsection Plot Parameter File

The **Longsections plot** option is used to make special long section plots for a **network** of drainage strings.

Some of the look of the drainage long section plot can be controlled from the **plot drainage network** panel itself, however a wider selection of control parameters is available by using a drainage long plot, plot parameter file.

The drainage long section plot parameters are placed in a file with ending **.ppf**.

Each parameter consists of a parameter name followed by one or more spaces and then the parameter value. There is only one parameter per line.

Anything on a line after a double forward slash *//* is considered to be a comment.

The set of all parameters for the drainage long section plot is enclosed within a set of curly brackets **{ }** with the header

drainage_long_plot "plot set name"

before the curly brackets.

That is,

```
drainage_long_plot "plot set name" {
        plot parameters
        one per line
}
```

If there is more than one *drainage_long_plot* parameter set in the file, only the first set is used.

There may also be parameter sets for other plot types such as *section_x_plot* in the same file. The other sets will be ignored when doing a drainage long section plot.

The plot parameters are documented in following groups:

For the *Plot Sheet layout*, please continue to the section Plot Sheet Layout.

Labelling the drainage string name, please continue to the section Labelling the Drainage String Name on the Plot.

Boxes area, please continue to the section Boxes Area.

Chainages and uprights, please continue to the section Chainages and Uprights.

Staggering, please continue to the section Staggering of Chainages and Uprights.

Below datum area, please continue to the section Below Datum Area.

Arrows area, please continue to the section Arrow Areas.

Grade arrows parameters, please continue to the section Parameters for the Arrows for Grades of the Pipes.

Pipe diameter arrows parameters, please continue to the section Parameters for the Arrows for Diameters of the Pipes.

Pipe velocity arrows parameters, please continue to the section Parameters for the Arrows for Velocity in the Pipes.

Pipe flow arrows parameters, please continue to the section Parameters for the Arrows for Flow in the Pipes.

Drainage line name arrows parameters, please continue to the section Parameters for the Arrows giving the Drainage Line Name.

User defined pipe arrows parameters, please continue to the section Parameters for the Arrows for User Defined Pipe Attributes.

Graph area parameters, please continue to the section Graph Area.

Top area parameters, please continue to the section Top Area.

Bubbles area parameters, please continue to the section Manhole name, Manhole Types and Surrounding Bubbles.

Change of direction parameters, please continue to the section Change of Direction Through Pits and Junctions.

Symbols at manhole parameters, please continue to the section Symbols at Manholes.

House connection parameters, please continue to the section Labelling House Connections

Symbols at Property controls parameters, please continue to the section Symbols at Property Controls.

Property controls parameters, please continue to the section Labelling Property Controls.

Hatching cut and fill parameters, please continue to the section Hatching Cut and Fill Areas.

Labelling cuts parameters, please continue to the section Labelling Cuts of Drainage Through Strings in a Model.

Title block parameters, please continue to the section Title Block Information.

Panel modifying parameters, please continue to the section Parameters that Modify Fields In the Plot Drainage Network Panel.

22.9.2.1.1 Plot Sheet Layout

The plot sheet is considered to have only positive co-ordinates with the origin (0,0) in the left hand corner. The units for the plot are millimetres.

The overall size of the plot sheet is given by either a defined sheet size, or by the width and height of the plot given in millimetres and separated by one or more spaces.

sheet_size	*text*	// sheet name, or
	"*mm* *mm*"	// sheet size: width height

The sheet size name and width and heights can be specified by the user in a file named sheets.4d which is in the normal set up areas, or is pointed to by the environment variable

SHEET_SIZES_4D	*file*	// file of plotter sheets sizes

The plotting area is restricted to within the plot sheet by giving the margins

left_margin	*mm*
right_margin	*mm*
top_margin	*mm*
bottom_margin	*mm*

The drainage long section plot will break an individual plot up if it doesn't fit across the sheet. There can be one or more rows of plot on the same sheet.

The top row is done first, followed by the second top row, then the third and so on until the bottom row. If there is only one row, it is considered to be the bottom row.

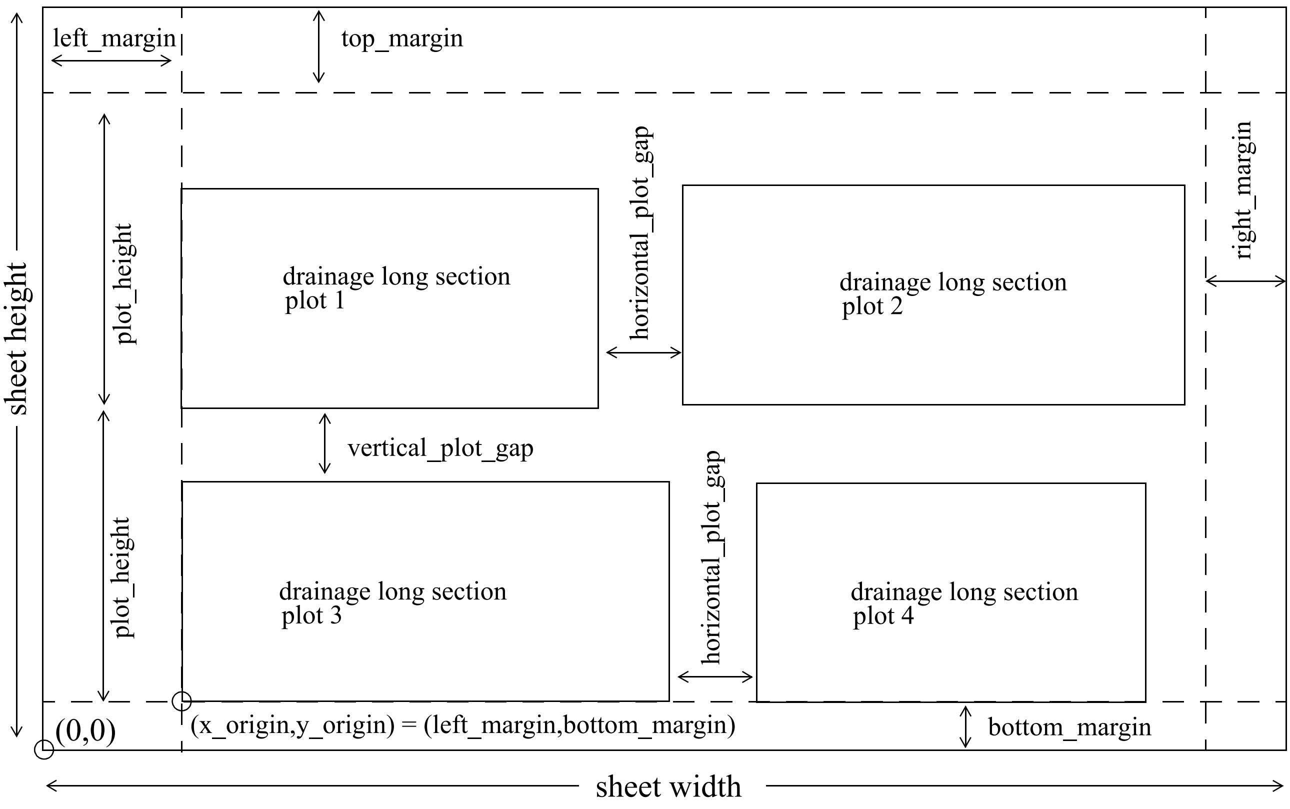
When a sheet is full, a follow on sheet is created.

As soon as one drainage string is completed, the next drainage string in the network model is plotted beginning on the same row as the previous drainage string and with a horizontal gap of size *horizontal_plot_gap* between the plots. If there is not enough room on the row to start the next plot, it will begin on a new row.

The position of the left hand bottom corner of the first plot in the bottom row is given by the parameters, *x_origin* and *y_origin* which are the same as *left_margin* and *bottom_margin* respectively.

If there are two or more rows of plots, the position of the first plot in each row is given by adding multiples of the *plot_height* to the *y_origin* (*bottom_margin*).

network_model	*text*	// model of drainage strings
x_origin	*mm*	// Position of the left hand bottom,
		// same as *left_margin*.
y_origin	*mm*	// corner of first plot in the bottom row,
		// same as *bottom_margin*.
only_one_line	*0*	// more than one row on a sheet
	1	// only one row of plot on a sheet
plot_height	*mm*	// total height of a plot row.
		// It includes the vertical_plot_gap.
horizontal_plot_gap	*mm*	// gap between plots on same row
vertical_plot_gap	*mm*	// gap between rows of plots,
		// also the size of the top area



The drainage long plot itself consists of eleven areas. From the bottom up, they are

drainage string name, boxes, below datum, arrow 1, bottom stagger, arrow 2, graph, arrow 3, top stagger, arrow 4, top.

The **drainage string name area** is where the name of the drainage string can be plotted.

The **boxes area** is where the chainages and various values for the drainage strings are labelled.

The **below datum area** is a region between the boxes area and the datum line.

The **arrow 1 area** is for drawing arrows where the arrows go between the staggered uprights and below the graph area. The datum line is at the bottom of the arrow 1 area.

The **bottom stagger area** is where the upright line staggers occur before going up from the boxes area to the graph area.

The **arrow 2 area** is for drawing arrows below the graph but where the arrows go between non-staggered uprights.

The **graph area** is the area where the actual plots of the strings are drawn.

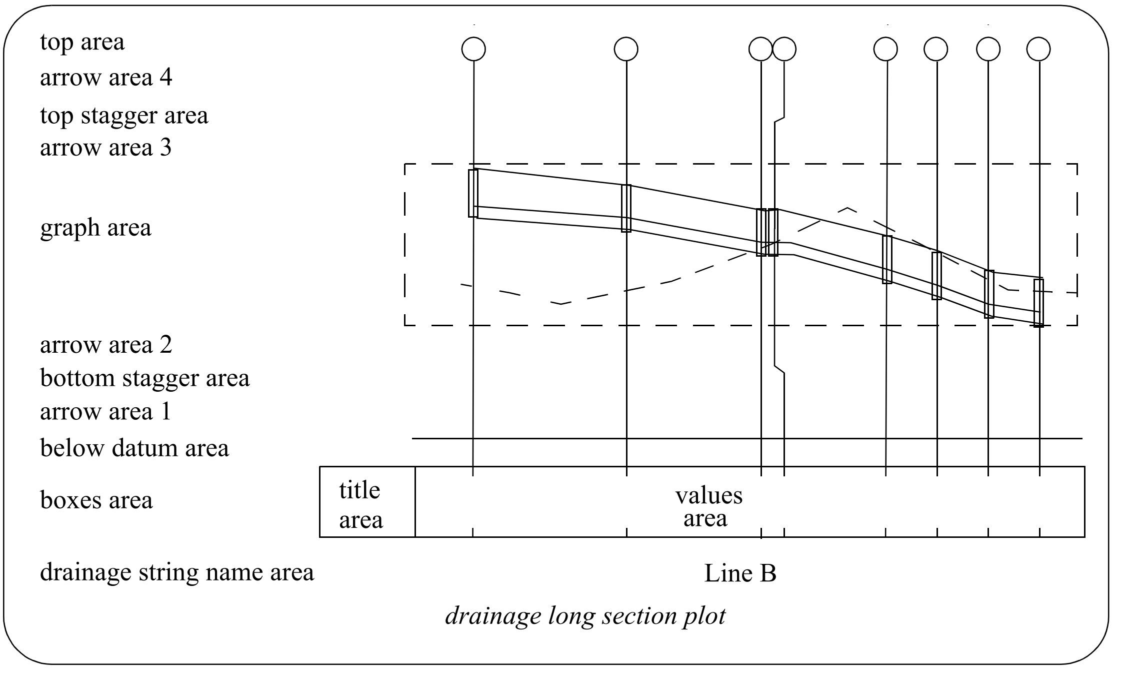
The **arrow 3 area** is for drawing arrows above the graph area and where the arrows go between non-staggered uprights.

The **top stagger area** is where the upright line staggers occur above the graph area.

The **arrow 4 area** is for drawing arrows where the arrows go between the staggered uprights and above the graph area.

The **top area** is an annotation area above the arrow 4 area and is used for bubbles, manhole names (pit names), junctions, deflection angles, *etc.*

The areas and the information in them will now be described in more detail.



22.9.2.1.2 Labelling the Drainage String Name on the Plot

The plot can be labelled with the name of the drainage string under the boxes area.

The name is made up of concatenation the text strings:

plot_name_pre_text drainage-string-name plot_name_post_text

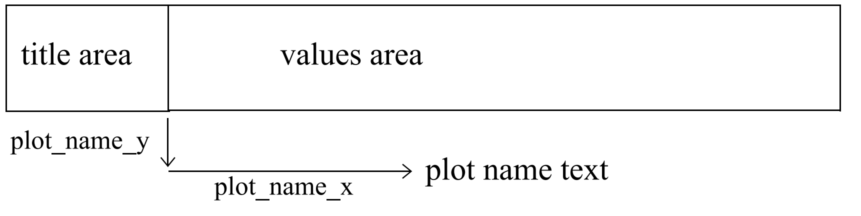
The plot name is positioned under the boxes.

plot_name_mode	***0***	// don't use the drainage string name
	1	// use the drainage string name in the // label
plot_name_pre_text	*text*	
plot_name_post_text	*text*	
plot_name_textstyle	*textstyle*	
plot_name_text_size	*mm*	
plot_name_text_colour	*colour*	
plot_name_x	*mm*	
plot_name_y	*mm*	

The *plot_name_x* is measured from the beginning of the height boxes.

The default for *plot_name_x* is centred on the values area.

The *plot_name_y* is measured from the bottom of the box area with positive being **down**.



labelling drainage string name on drainage long section plots

Example of Labelling Drainage String Name

plot_name_pre_text	“Drainage Long Section Plot for String”
plot_name_mode	1
plot_name_post_text	“”
plot_name_text_size	15
plot_name_text_colour	red
plot_name_textstyle	ISO
plot_name_y	30

22.9.2.1.3 Boxes Area

Many of the drainage string values (invert levels and depth, hgl values, natural and finished surface, *etc.*) can be labelled in the boxes area at the bottom of the drainage long section plot.

Each type of information is plotter in a row made up of a title, and the actual values given at the chainage of each pit in the drainage string.

Each row of information is surrounded by lines to form a box, and the stacked boxes form the boxes area at the bottom of the drainage plot.

The **title** for the information, is drawn in the **title area** of the **boxes area** and the values are drawn in the **values area** of the **boxes area**.

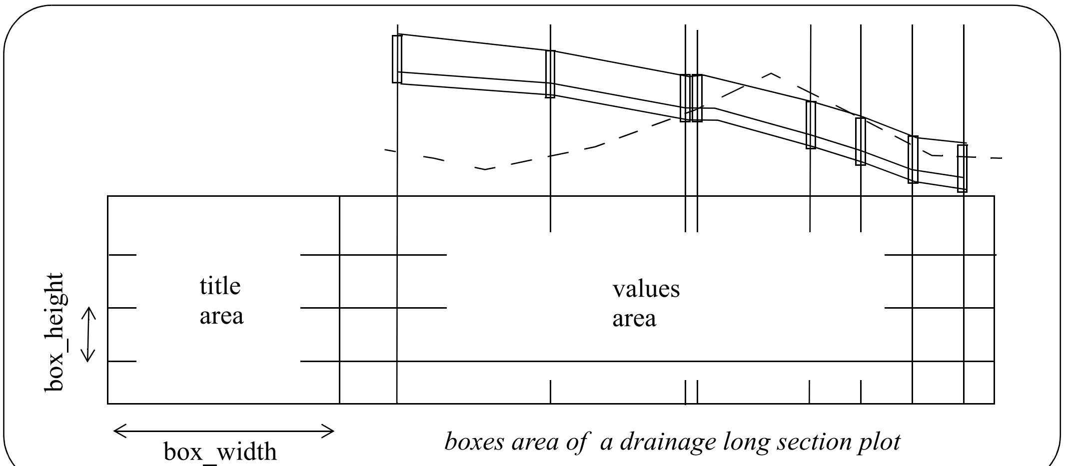
Consequently the boxes area is made up of rows of text consisting of:

title followed by the **values** along the drainage string.

The **titles area** for the left plot in the bottom row starts at the co-ordinate (*x_origin*,*y_origin*) and each row is begun by adding the distance *plot_height* to the *y_origin* (see previous section).

The width of the **title area** is given by the **box_width** parameter and the height of each box is given by **box_height**.

box_width	*mm*	// width of the label boxes.
box_height	*mm*	// height of each box



boxes area of a drainage long section plot

The colour of the box line work is given by:

box_colour	*colour*	// colour of the lines in the boxes

A default text size and colour can be specified for the title text and the values, or sizes, colours and text styles can be given for each individual box (given later in this section).

title_box_text_size	*mm*	// size of title text in boxes
title_box_text_colour	*colour*	// colour of title text in boxes
box_text_size	*mm*	// size of value text in boxes
box_text_colour	*colour*	// colour of values in boxes

The values text is written at right angles to the bottom of the boxes. It can be either top or bottom justified with respect to the box (*box_text_left_justify*).

The values text can be on the left, right or centred on the uprights and is given by the parameter *box_text_side*.

The width of the values area is determined by the number of chainages to be labelled and whether the values are staggered to prevent over writing (see next section).

box_text_left_justify	*0*	// top justify text in values area
	1	// bottom justify text in values area
box_text_side	*0*	// left (default)
	1	// right
	2	// centre

The default order of the boxes from the bottom up is

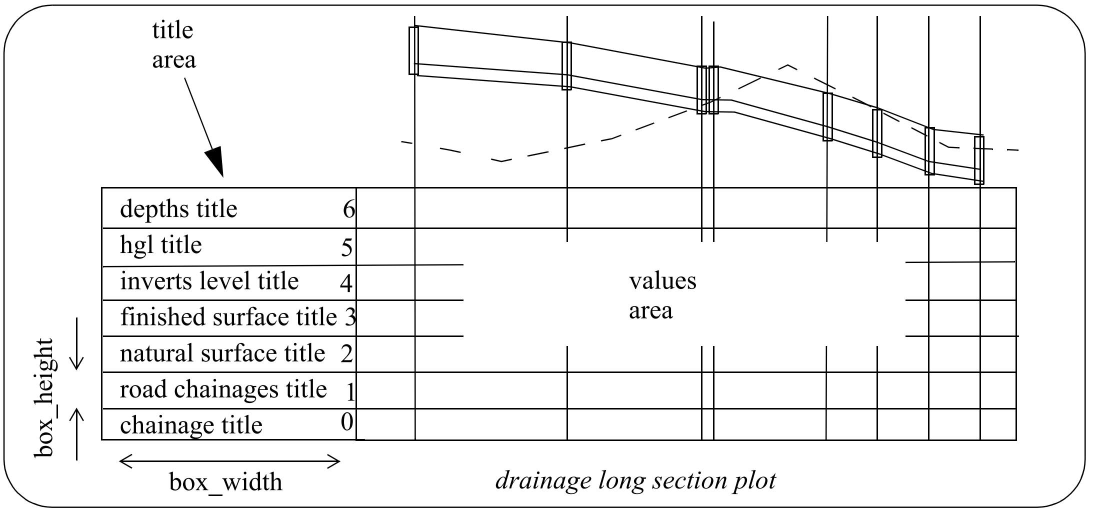
0. drainage string chainages
1. road centre line chainages - user choice
2. natural surface heights - user choice
3. finished surface heights - user choice
4. before pit and after pit invert levels
5. before pit and after pit hgl values - user choice
6. before pit and after pit depth to inverts - user choice

There are parameters to change the order for 1-6.
*Drainage string chainage*s is fixed at the bottom (box 0) but what is in the boxes above box 0 is can be set by the box_n parameters:

box_n	*number_from_list*	// the n'th box above the drainage string
		// chainages will contain given item number
		// form the list.

For example, to have *finished surface heights* in the first box the drainage string chainages, use:

box_1 3 // the (1) will be replaced by (3)



chainage_box_size	*mm*	// box height for chainage box
centre_chainage_box_size	*mm*	// box height for centre chainage box
surface_box_size	*mm*)	// box height for surface box
f_surface_box_size	*mm*	// box height for finished surface box
invert_height_box_size	*mm*	// box height for invert height box
hgl_box_size	*mm*	// box height for hgl box
invert_depth_box_size	*mm*	// box height for invert depth box

The default for each box size is *box_height*

The boxes (1), (2), (3) and (5) can be suppressed by the parameters

draw_centre_chainage	*0/1*	// 0 = don't have centre line chainages

```
                                          //      box
draw_ns_text            0/1               // 0 = don't draw ns values from
                                          //     ns tin for drainage string
draw_fs_text            0/1               // 0 = don't draw fs values from
                                          //      fs tin or top of manhole
                                          //      for drainage string
draw_fs_mode            0                 // 0 = fs_text is from fs tin
                        1                 // 1 = fs_text is top of manhole
draw_hgl_value          0/1               // 0 = don't have hgl value boxes
```

The definition of depth to invert and whether box (6) is suppressed or not, are given by the parameter *depth_mode:*

```
depth_mode              0                 //don't draw depth to invert values
                        1                 // draw depth to invert values to fs tin
                                          // for drainage string
                        2                 // draw depth to invert values to top of
                                          // manhole for drainage string
                        3                 // draw depth to invert values to ns tin
                                          // for drainage string
```

The default for the number of decimal places used in the values in the boxes is:

```
number_of_decimals      integer           // default number of decimal places
```

The text, size, colour and textstyle for the title text and values text for each box can be set by

(0) drainage string chainages

```
plot_title_chainage_name      text        // first line of title for chainages box
plot_title_chainage_name_2    text        // second line of title for chainages box
chainage_title_colour         colour      // colour of text
chainage_title_text_size      mm          // size of title in chainage box
chainage_title_textstyle      textstyle   // textstyle for title in chainage box
chainage_text_colour          colour      // colour of chainage values
chainage_text_size            mm          // size of values
chainage_textstyle            textstyle   // textstyle of values
chainage_decimals             integer     // number of decimal places in chainage
```

(1) road centreline chainages

```
plot_title_centre_chainage_name     text       // title for road centre
                                               // line chainages box
plot_title_centre_chainage_name_2   text       // second line of title for road
                                               // centre line chainages box
chainage_title_colour_cl            colour     // colour of text
chainage_title_text_size_cl         mm         // size of title in cl box
chainage_title_textstyle_cl         textstyle  // textstyle for title
chainage_text_colour_cl             colour     // colour of cl values
chainage_text_size_cl               mm         // size of values
chainage_textstyle_cl               textstyle  // textstyle of values
chainage_cl_decimals                integer    // number of decimal places in chainage cl
```

(2) natural surface values

```
plot_title_surface_name       text        // title for the drainage ns values box
plot_title_surface_name_2     text        // second line of title for drainage ns values
ns_title_colour               colour      // colour of text
ns_title_text_size            mm          // size of title in ns box
ns_title_textstyle            textstyle   // textstyle for title
ns_text_colour                colour      // colour of ns values
ns_text_size                  mm          // size of values
```

ns_textstyle	*textstyle*	// textstyle of values
ns_decimals	*integer*	// number of decimal places in ns

(3) finished surface values

plot_title_finished_name	*text*	// title for the drainage fs values box
plot_title_finished_name_2	*text*	// second line of title for drainage fs values
fs_title_colour	*colour*	// colour of text
fs_title_text_size	*mm*	// size of title in fs box
fs_title_textstyle	*textstyle*	// textstyle for title
fs_text_colour	*colour*	// colour of ns values
fs_text_size	*mm*	// size of values
fs_textstyle	*textstyle*	// textstyle of values
fs_decimals	*integer*	// number of decimal place in fs

(4) invert levels

plot_title_invert_name	*text*	// title for the invert levels box
plot_title_invert_name_2	*text*	// second line of title for the invert levels box
il_title_colour	*colour*	// colour of text
il_title_text_size	*mm*	// size of title in il box
il_title_textstyle	*textstyle*	// textstyle for title
il_text_colour	*colour*	// colour of il values
il_text_size	*mm*	// size of values
il_textstyle	*textstyle*	// textstyle of values
il_decimals	*integer*	// number of decimal places in il

(5) hgl values

plot_title_hgl_name	*text*	// title for hgl values box
plot_title_hgl_name_2	*text*	// second line of title for hgl values box
hgl_title_colour	*colour*	// colour of text
hgl_title_text_size	*mm*	// size of title in hgl box
hgl_title_textstyle	*textstyle*	// textstyle for title
hgl_text_colour	*colour*	// colour of hgl values
hgl_text_size	*mm*	// size of values
hgl_textstyle	*textstyle*	// textstyle of values
hgl_decimals	*integer*	// number of decimal places in hgl

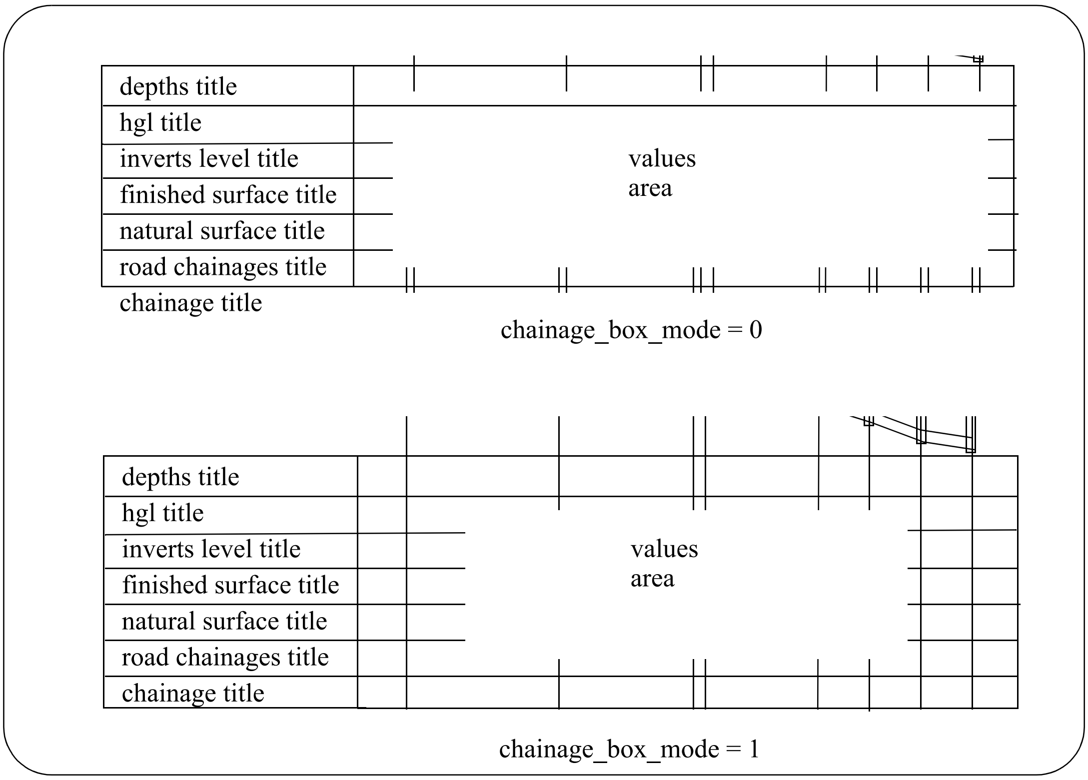
(6) depths

plot_title_depth_name	*text*	// title for depth of inverts box
plot_title_depth_name_2	*text*	// second line of title for depth of inverts box
depth_title_colour	*colour*	// colour of text
depth_title_text_size	*mm*	// size of title in depth box
depth_title_textstyle	*textstyle*	// textstyle for title
depth_text_colour	*colour*	// colour of depth values
depth_text_size	*mm*	// size of values
depth_textstyle	*textstyle*	// textstyle of values
depth_decimals	*integer*	// number of decimal places in depth

Drawing the Boxes

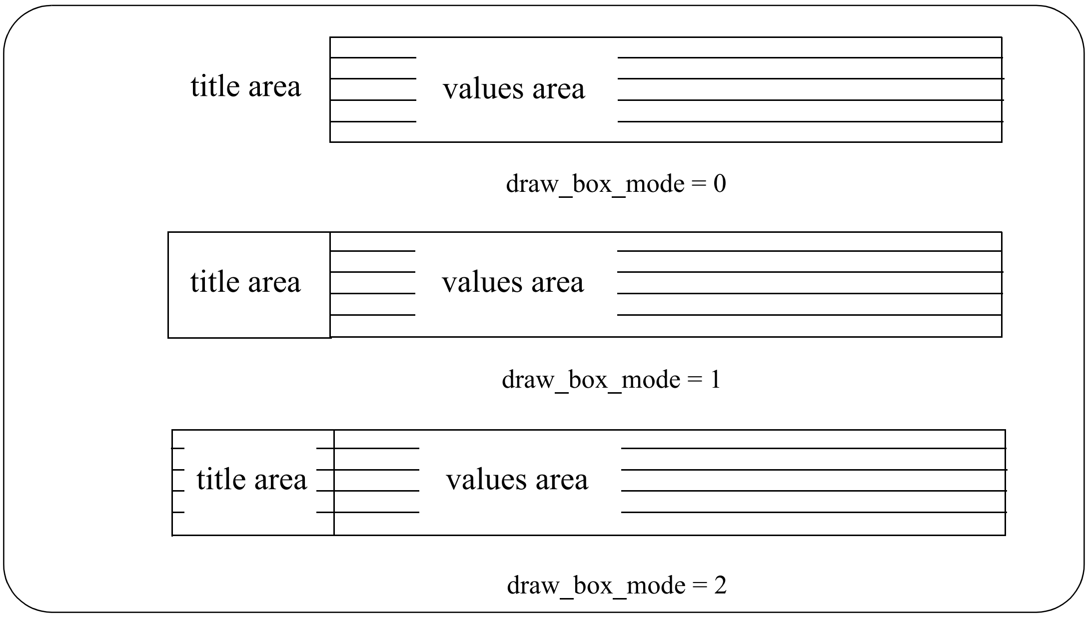
The bottom line from the boxes can be drawn or not drawn:

chainage_box_mode	*0*	// don't draw the bottom line
	1	// draw all the box line work - **default**



A box can be drawn/not drawn around the title area of the boxes area.

draw_box_mode	*0*	// don't draw the title area box
	1	// draw the title area box, no lines - **default**
	2	// draw title box area with lines



22.9.2.1.4 Chainages and Uprights

The chainage box contains drainage string chainages for

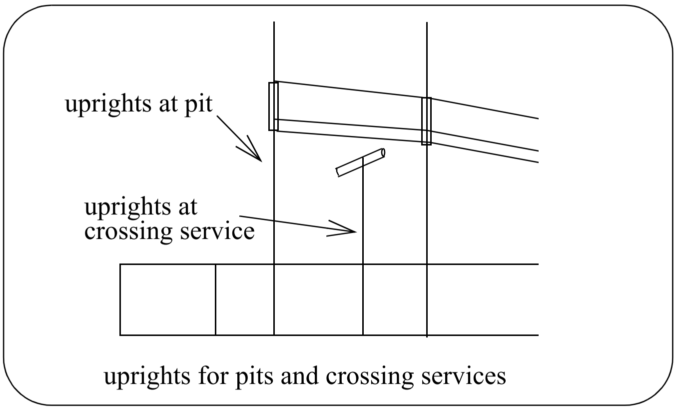
(a) the chainages of each pit in the drainage string

(b) the chainages on the drainage string where any strings in models on the section view are **cut** by the drainage string (crossing services)

For a **pit**, the invert levels, depths, finished and natural surface levels, and hgl of pipes at a pit can all labelled at the chainage of the pit.

For a **crossing service**, the level of the crossing service and the chainage on the drainage string where the crossing occurs are labelled.

Note - crossing services can also be labelled using cuts of drainage line through strings. However, this does not create a chainage in the chainage box.

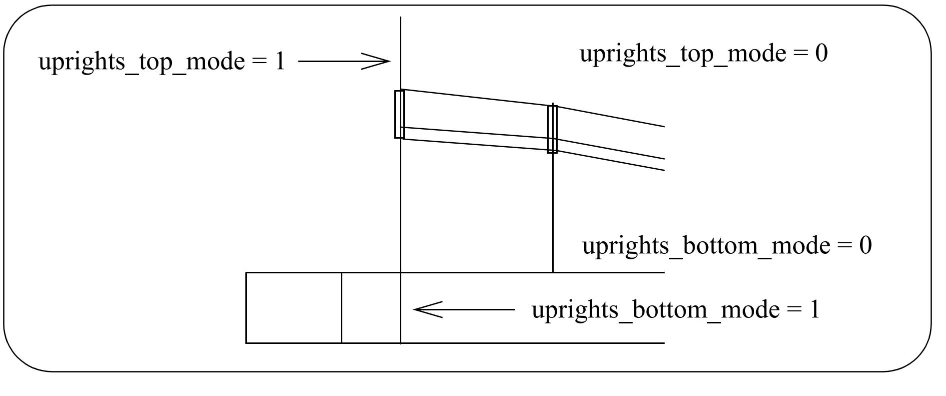


Uprights, or leader lines, can be drawn from the chainage values at the bottom of the boxes to the pit in the graph area, and from the top of the pit to the top area.

Using the parameters *uprights_top_mode* and *uprights_bottom_mode*, it is possible to suppress the drawing from the top of the pit to the top area, and also from the top of the boxes to the bottom of the boxes.

uprights_top_mode	*0*	// stop at top of pit
	1	// go to top area (default)
uprights_bottom_mode	*0*	// stop at top of boxes
	1	// go to bottom of boxes (default)

Also uprights can be drawn from the chainage values bottom of the boxes to the crossing service in the graph area.



The chainage values at the uprights for the pits can be running chainage along the drainage string, pipe length chainages (*i.e.* the chainage starts at zero for each pipe) or both.

chainage_mode	*0*	//pipe length chainage
	1	// running chainage
	2	// both pipe length and running

If the *chainage_mode* includes "pipe length chainage" (modes 0 and 2), the pipe length can either be labelled with a 0.0 chainage at the start pit for the pipe and the pipe length at the end pit, or just have the length centred between the two pits.

centre_pipe_length	***0***	// default - 0 at start pit, length at end pit
	1	// centre pipe length

The chainage values for the crossing services can be running chainage along the drainage string, chainage length from the previous pit (*i.e.* the chainage starts at zero at the previous pit) or both.

service_chainage_mode	*0*	//pipe length chainage
	1	// running chainage
	2	// both pipe length and running

The colour of the uprights for the pits and crossing services are given by:

manhole_line_colour	*colour*	// colour of uprights to the pits
service_line_colour	*colour*	// colour of upright to the crossing // services

The crossing services are labelled with the drainage string chainage of the crossing point, the invert level of the service at the crossing and the name of the service which is made up of:

service diameter   name of service   invert level at the service at the crossing.

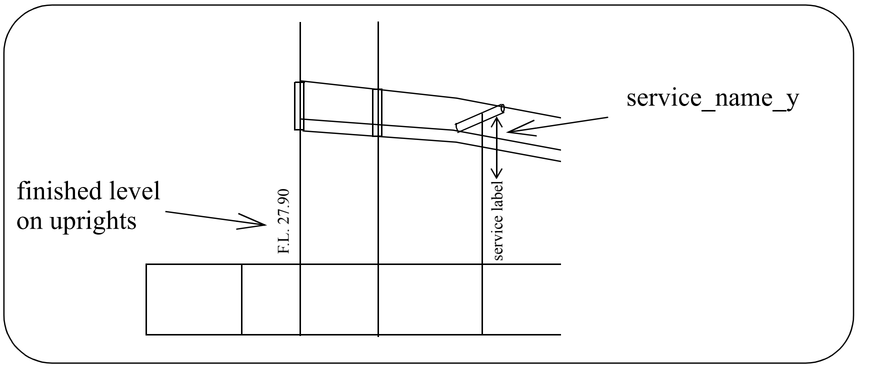
The service label is drawn the distance service_name_y below the crossing point:

service_name_y	*mm*	// def 1, distance of label below crossing point

The colours, text styles and sizes of the service information is specified by:

service_ch_text_colour	*colour*	// colour of services ch text
service_ch_text_size	*mm*	// size of services ch text
service_ch_textstyle	*mm*	// textstyle of services ch text
service_il_text_colour	*colour*	// colour of services il text
service_il_text_size	*mm*	// size of services il text
service_il_textstyle	*mm*	// textstyle of services il text
service_name_text_colour	*colour*	// colour of services name
service_name_text_size	*mm*	// size of services name
service_name_textstyle	*mm*	// textstyle of services name
service_name_decimals	*integer*	// number of decimal is il in name

A **finished surface** value or **top of manhole** value can also be written vertically along the uprights.



Whether the value is finished surface level or top of manhole level, the values position, size, colour, *etc.* is controlled by the parameters:

draw_fs_vertical	*0*	// don't draw fs vertical values
	1	// always draw values
	2	// only draw values when different -see // draw_fs_vertical_mode

draw_fs_vertical_mode	*0*	// the fs value is the value from the **fs tin**
		// for the drainage string.
		// When draw_fs_vertical is 1,
		// always draw the fs value.
		// When draw_fs_vertical is 2,
		// only draw fs value when it is
		// different from the top of manhole value

	1	// the fs value is the **top of manhole**
		// for the drainage string.
		// When draw_fs_vertical is 1,
		// always draw the fs value.
		// When draw_fs_vertical is 2,
		// only draw fs value when it is
		// different from the value of the fs tin.

		draw_fs_vertical_mode	
		0	**1**
draw_fs_vertical	**0**	nothing	nothing
	1	fs tin	top of manhole
	2	fs tin when different from top of manhole	top of manhole when different from fs tin

fs_vertical_pre_text	*text*	// text before the vertical fs value
fs_vertical_post_text	*text*	// text after the vertical fs value
fs_vertical_decimals	*integer*	// number of decimals in vertical fs
fs_vertical_colour	*colour*	// colour of the vertical fs text
fs_vertical_size	*mm*	// size of the vertical fs text
fs_vertical_textstyle	*textstyle*	// textstyle of the vertical fs
fs_vertical_x	*mm*	// x adjustment to position of text
fs_vertical_y	*mm*	// y adjustment to position of text

22.9.2.1.5 Staggering of Chainages and Uprights

If the text values are placed at the real chainage positions at the bottom of the plot, text over writing can easily occur if the chainages are very close together.

To prevent such over writing, the text can be **staggered**.

That is, if the text is going to over write a previous text value, the next text value is actually moved along until there is no over writing.

Since the text is no longer at the correct chainage position, the uprights to the pits and services start at the text position and then bend back to the correct chainage position on the plot. The region where the bending occurs is called the stagger area.

For the drainage plot, there is an area below the graph where the uprights bend backwards from the staggered text position to the real chainage position (bottom stagger area).

And there is a second area above the graph where the uprights bend forwards from the real chainage position to the staggered text position (top stagger area).

Hence annotation above the top stagger area will line up with the staggered values below the bottom stagger area.

The stagger area below the graph area is defined by

```
stagger_height_1        mm      // distance from the top of arrow_area_1
                                // to the start of the staggers
stagger_height_2        mm      // distance over which stagger occurs
stagger_gap_bottom      mm      // distance from end of staggers to the
                                // bottom of arrow_area_2
```

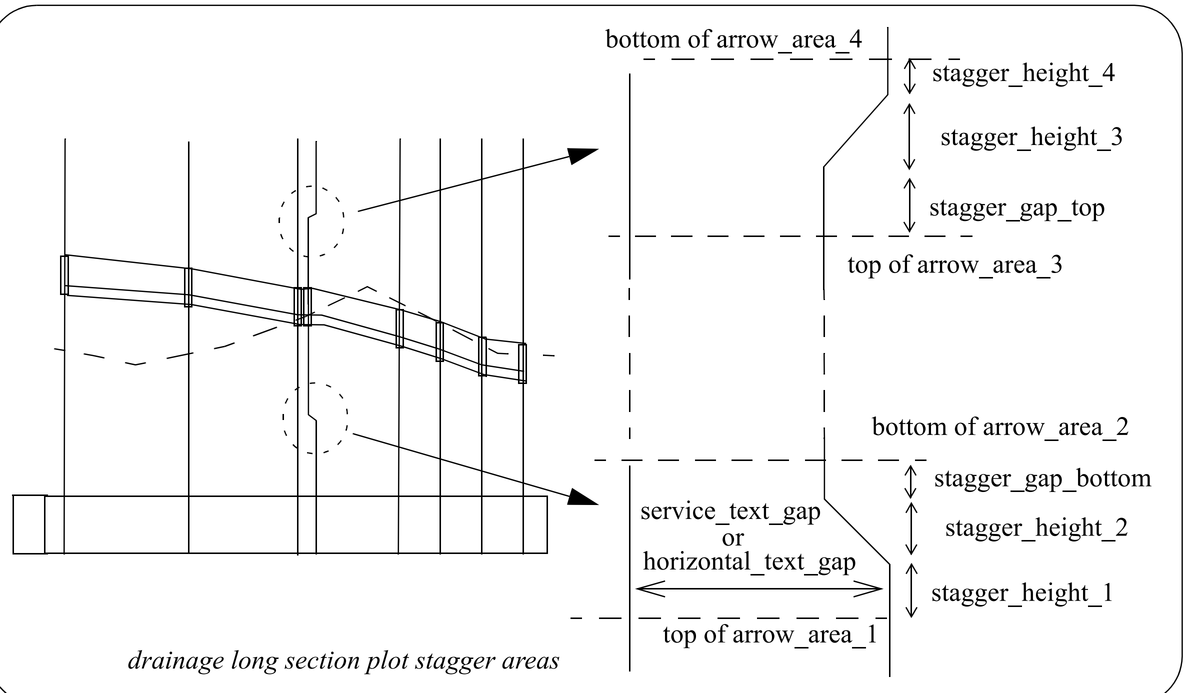
The stagger area above the graph area is defined

```
stagger_gap_top         mm      // distance from the top of arrow_area_3
                                // to the start of the staggers
stagger_height_3        mm      // distance over which stagger occurs
stagger_height_4        mm      // distance from end of staggers to the
                                // bottom of arrow_area_4
```

The distance to be left for text to avoid over writing is:

```
horizontal_text_gap     mm      // minimum distance to leave for text
                                // after pit values
service_text_gap        mm      // minimum distance to leave for text
                                // after service values
```

When staggering occurs, it is possible for the values area to be longer than the graph area.



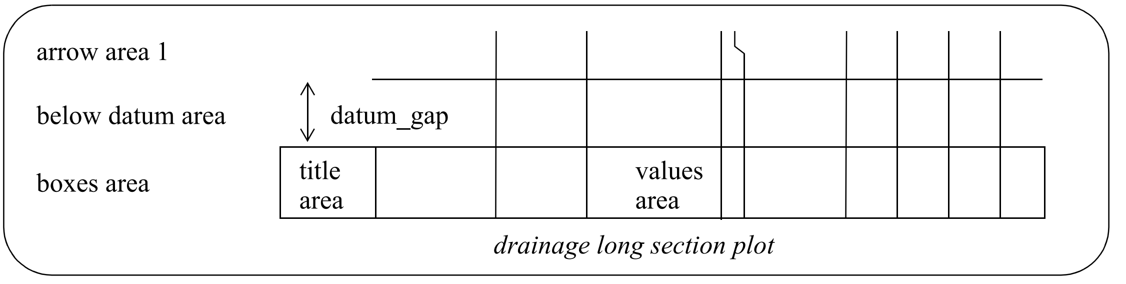
drainage long section plot stagger areas

22.9.2.1.6 Below Datum Area

The **below datum area** is the region between the boxes area and the arrow_1 area. The datum line is drawn at the top of the below datum area.

The size of the below datum area is

datum_gap	*mm*	// height of the below datum area



The datum value is placed above the datum line, preceded by a datum name on the left hand side of the title area.

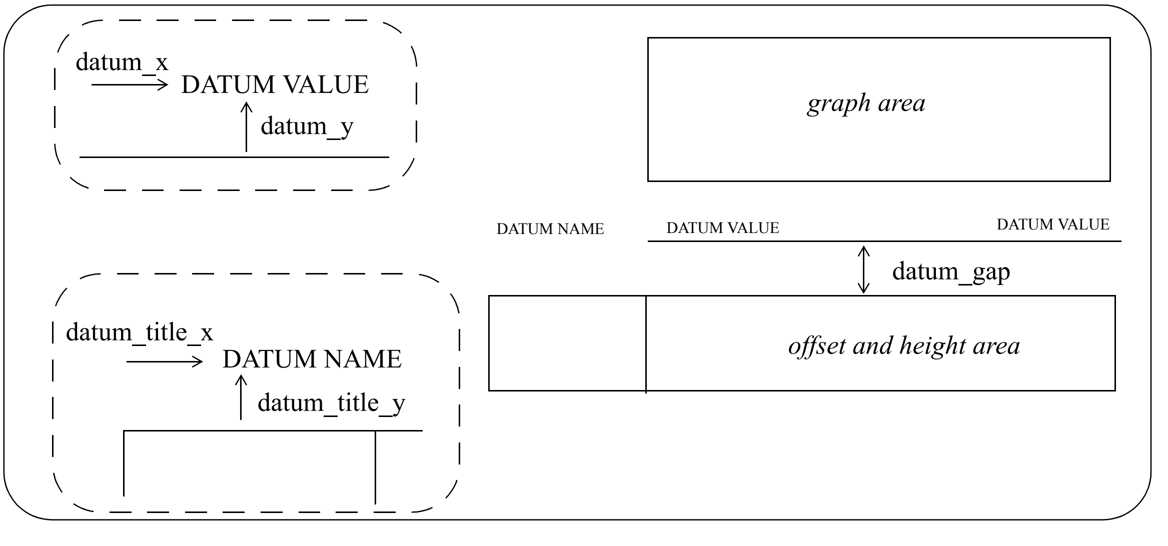
Since the datum value is automatically calculated to try and fit the plot vertically into the graph area, the datum value can change along the plot. When a datum change occurs, the new datum value is written on the datum line at the relevant chainage position.

The datum name, text size, colour and position are controlled by:

datum_name	*text*	// datum title
datum_title_x	*mm*	// distance to write name from lhs
datum_title_y	*mm*	// distance above datum line
datum_title_textstyle	*textstyle*	// textstyle of name
datum_title_text_size	*mm*	// text size
datum_title_colour	*colour*	// text colour

The datum value, number of decimal places, text size, colour and position are controlled by:

datum_decimals	*integer*	// number of decimal places in datum value // >0 drop trailing zeros after decimal point // < 0 keep trailing zeros
datum_x	*mm*	// x dist. to write value from manhole
datum_y	*mm*	// y dist. to write value above datum line
datum_text_size	*mm*	// size of datum value
datum_textstyle	*textstyle*	// textstyle for datum
datum_text_colour	*colour*	// colour of the datum text and line



22.9.2.1.7 Arrow Areas

Apart from information labelled in the boxes and top areas, the drainage long section plot can place arrows between pits for other information such as

(a) pipe grade or slope
(b) pipe diameter
(c) velocity
(d) flow
(e) drainage line name
(f) user defined pipe attributes

A specific arrow area is defined for each set of arrows.

The **arrow areas** are designed for drawing and labelling arrows between pits.

All the values refer to pipes connecting adjacent pits and the relevant pipe is indicated by drawing an arrow between the uprights from the chainage text to the pits.

Because staggering of the chainage text can occur, the position of the uprights to draw the arrows between can vary depending on whether the arrow is above or below a stagger area.

Hence, there are four arrow areas:

arrow area 1 which is below the graph and the bottom stagger area. The arrows go between the *staggered* chainage positions of the pits.

The datum line is at the bottom of arrow 1 area.

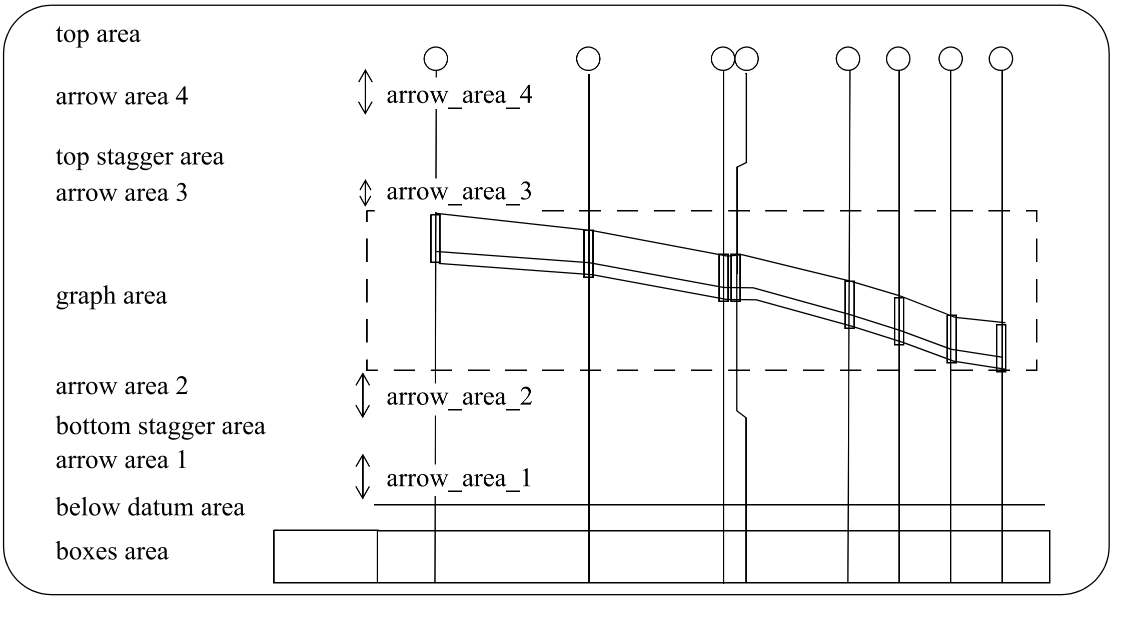
arrow area 2 which below the graph but above the bottom stagger area. Hence the arrows go between the non-*staggered* chainage positions of the pits.

arrow area 3 which is above the graph but below the top stagger area. The arrows go between the non-*staggered* chainage positions of the pits.

arrow area 4 is above the graph and the top stagger area. The arrows go between the *staggered* chainage positions of the pits.

The heights of the four areas (which can be zero) are defined by:

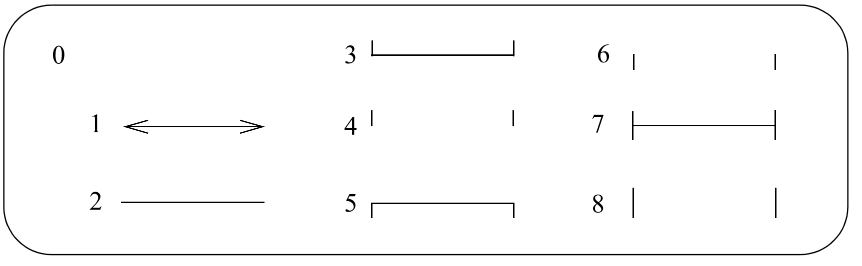
arrow_area_1	*mm*	// height of arrow_1 area
arrow_area_2	*mm*	// height of arrow_2 area
arrow_area_3	*mm*	// height of arrow_3 area
arrow_area_4	*mm*	// height of arrow_4 area



The parameters for controlling the arrows between manholes and associated text for the values of pipe slope, pipe diameter, flow, velocity, drainage line name and user defined pipe attributes are now given.

Each set of parameters has an arrow_mode with the following values:

..._arrow_mode	0	// no arrow
	1	// arrow
	2	// line
	3	// line with uprights at ends
	4	// uprights, no line
	5	// line with downrights
	6	// downrights, no line
	7	// line with up and downrights at ends
	8	// up and downrights, no line



22.9.2.1.8 Parameters for the Arrows for Grades of the Pipes

Parameters specifying the arrow area used and the position of the arrows.

draw_pipe_grade	*0*	// don't draw pipe grade
	1	// draw pipe grade
pipe_grade_arrow_area	*m*	// the arrow area for the arrows // m = 1,2,3 or 4
pipe_grade_y	*mm*	// distance that the arrow is above the // bottom of the arrow area.

For the arrow, the value of the grade can be specified as either **1 in** or **percent** grade:

percentage_grade	*0*	// the value is a "1 in" grade
	1	// the value is a percent grade

Parameters for the title text on the left hand side of the plot:

pipe_grade_title	*text*	//title on the left hand side of the arrow
pipe_grade_title_textstyle	*textstyle*	// textstyle of the title
pipe_grade_title_text_size	*mm*	// size of the title
pipe_grade_title_text_colour	*colour*	// colour of the title
pipe_grade_title_offset	*mm*	//distance to raise or lower the text // from the arrow position
pipe_grade_title_x	*mm*	// distance from the left hand side of the // plot

Parameters for the arrow type:

pipe_grade_arrow_mode	*0*	// no arrow
	1	// arrow
	2	// line
	3	// line with uprights at ends
	4	// uprights, no line
	5	// line with downrights
	6	// downrights, no line
	7	// line with up and downrights at ends
	8	// up and downrights, no line
pipe_grade_arrow_colour	*colour*	// colour of the arrow
pipe_grade_arrow_size	*mm*	// height of the arrow
pipe_grade_arrow_gap	*0*	// no gap in arrow
	1	// leave gap in arrow for text

Parameters for the text on the arrow:

pipe_grade_arrow_pre_text	*text*	// text before the arrows text
pipe_grade_arrow_post_text	*text*	// text after the arrows text

pipe_grade_arrow_decimals	*integer*	// The number of decimal places used
		// when writing out the pipe grade
		// If > 0, all trailing zeros after the
		// decimal place are removed.
		// If < 0, the absolute value is taken as
		// the number of decimal places and **no**
		// trailing zeros are removed after the
		//decimal point.

pipe_grade_arrow_textstyle	*textstyle*	// textstyle of arrow text
pipe_grade_arrow_text_size	*mm*	// size of the text
pipe_grade_arrow_text_colour	*colour*	// colour of the text
pipe_grade_arrow_text_offset	*mm*	// distance to raise or lower the text // from the arrow position

22.9.2.1.9 Parameters for the Arrows for Diameters of the Pipes

The diameter of the drainage pipes can be plotted between each pit.

Inside 12d Model, the pipe diameter is given a world units (usually metres), but on the diameter arrow, the diameter is multiplied by 1000 and written out as an integer value. This is normally millimetres.

For diameters entered in feet and the plot diameter to be in inches, the following parameter should be set

pipe_diameter_scale_factor 12. // default is 1000.

Parameters specifying the arrow area used and the position of the pipe diameter arrows.

draw_pipe_diameter	*0*	// don't draw pipe diameter
	1	// draw pipe diameter
pipe_diameter_arrow_area	*m*	// the arrow area for the arrows // m = 1,2,3 or 4
pipe_diameter_y	*mm*	// distance that the arrow is above the // bottom of the arrow area.

Parameters for the title text on the left hand side of the plot:

pipe_diameter_title	*text*	// title on the left hand side of the arrow
pipe_diameter_title_textstyle	*textstyle*	// textstyle of the title
pipe_diameter_title_text_size	*mm*	// size of the title
pipe_diameter_title_text_colour	*colour*	// colour of the title
pipe_diameter_title_offset	*mm*	// distance to raise or lower the text // from the arrow position
pipe_diameter_title_x	*mm*	// distance from the left hand side of the // plot

Parameters for the arrow type:

pipe_diameter_arrow_mode	*0*	// no arrow
	1	// arrow
	2	// line
	3	// line with uprights at ends
	4	// uprights, no line
	5	// line with downrights
	6	// downrights, no line
	7	// line with up and downrights at ends
	8	// up and downrights, no line
pipe_diameter_arrow_colour	*colour*	// colour of the arrow
pipe_diameter_arrow_size	*mm*	// height of the arrow
pipe_diameter_arrow_gap	*0*	// no gap in arrow
	1	// leave gap in arrow for text

Parameters for the text on the arrow:

pipe_diameter_arrow_pre_text	*text*	// text before the arrows text
pipe_diameter_arrow_post_text	*text*	// text after the arrows text

On the arrow, the pipe type can be also plotted after the pipe_diameter_arrow_post_text:

pipe_type_mode	*0*	// don't include the pipe type
	1	// include the pipe type after the // post_text
pipe_diameter_arrow_textstyle	*textstyle*	// textstyle of arrow text
pipe_diameter_arrow_text_size	*mm*	// size of the text
pipe_diameter_arrow_text_colour	*colour*	// colour of the text
pipe_diameter_arrow_text_offset	*mm*	// distance to raise or lower the text // from the arrow position

22.9.2.1.10 Parameters for the Arrows for Velocity in the Pipes

Parameters specifying the arrow area used and the position of the arrows.

draw_pipe_velocity	*0*	// don't draw pipe velocity
	1	// draw pipe velocity
pipe_velocity_arrow_area	*m*	// the arrow area for the arrows // m = 1,2,3 or 4
pipe_velocity_y	*mm*	// distance that the arrow is above the // bottom of the arrow area.

Parameters for the title text on the left hand side of the plot:

pipe_velocity_title	*text*	//title on the left hand side of the arrow
pipe_velocity_title_textstyle	*textstyle*	// textstyle of the title
pipe_velocity_title_text_size	*mm*	// size of the title
pipe_velocity_title_text_colour	*colour*	// colour of the title
pipe_velocity_title_offset	*mm*	//distance to raise or lower the text // from the arrow position
pipe_velocity_title_x	*mm*	// distance from the left hand side of // the plot

Parameters for the arrow type:

pipe_velocity_arrow_mode	*0*	// no arrow
	1	// arrow
	2	// line
	3	// line with uprights at ends
	4	// uprights, no line
	5	// line with downrights
	6	// downrights, no line
	7	// line with up and downrights at ends
	8	// up and downrights, no line
pipe_velocity_arrow_colour	*colour*	// colour of the arrow
pipe_velocity_arrow_size	*mm*	// height of the arrow
pipe_velocity_arrow_gap	*0*	// no gap in arrow
	1	// leave gap in arrow for text

Parameters for the text on the arrow:

pipe_velocity_arrow_pre_text	*text*	// text before the arrows text
pipe_velocity_arrow_post_text	*text*	// text after the arrows text

pipe_velocity_arrow_decimals	*integer*	// The number of decimal places used
		// when writing out the pipe velocity
		// If > 0, all trailing zeros after the
		// decimal place are removed.
		// If < 0, the absolute value is taken as
		// the number of decimal places and **no**
		// trailing zeros are removed after the
		// decimal point.

pipe_velocity_arrow_textstyle	*textstyle*	// textstyle of arrow text
pipe_velocity_arrow_text_size	*mm*	// size of the text
pipe_velocity_arrow_text_colour	*colour*	// colour of the text
pipe_velocity_arrow_text_offset	*mm*	// distance to raise or lower the text // from the arrow position

22.9.2.1.11 Parameters for the Arrows for Flow in the Pipes

Parameters specifying the arrow area used and the position of the arrows.

draw_pipe_flow	*0*	// don't draw pipe flow
	1	// draw pipe flow
pipe_flow_arrow_area	*m*	// the arrow area for the arrows // m = 1,2,3 or 4
pipe_flow_y	*mm*	// distance that the arrow is above the // bottom of the arrow area.

Parameters for the title text on the left hand side of the plot:

pipe_flow_title	*text*	// title on the left hand side of the arrow
pipe_flow_title_textstyle	*textstyle*	// textstyle of the title
pipe_flow_title_text_size	*mm*	// size of the title
pipe_flow_title_text_colour	*colour*	// colour of the title
pipe_flow_title_offset	*mm*	// distance to raise or lower the text // from the arrow position
pipe_flow_title_x	*mm*	// distance from the left hand side of // the plot

Parameters for the arrow type:

pipe_flow_arrow_mode	*0*	// no arrow
	1	// arrow
	2	// line
	3	// line with uprights at ends
	4	// uprights, no line
	5	// line with downrights
	6	// downrights, no line
	7	// line with up and downrights at ends
	8	// up and downrights, no line
pipe_flow_arrow_colour	*colour*	// colour of the arrow
pipe_flow_arrow_size	*mm*	// height of the arrow
pipe_flow_arrow_gap	*0*	// no gap in arrow
	1	// leave gap in arrow for text

Parameters for the text on the arrow:

pipe_flow_arrow_pre_text	*text*	// text before the arrows text
pipe_flow_arrow_post_text	*text*	// text after the arrows text

pipe_flow_arrow_decimals	*integer*	// The number of decimal places used
		// when writing out the pipe flow
		// If > 0, all trailing zeros after the
		// decimal place are removed.
		// If < 0, the absolute value is taken as
		// the number of decimal places and **no**
		// trailing zeros are removed after the
		// decimal point.

pipe_flow_arrow_textstyle	*textstyle*	// textstyle of arrow text
pipe_flow_arrow_text_size	*mm*	// size of the text
pipe_flow_arrow_text_colour	*colour*	// colour of the text
pipe_flow_arrow_text_offset	*mm*	//distance to raise or lower the text // from the arrow position

22.9.2.1.12 Parameters for the Arrows giving the Drainage Line Name

Parameters specifying the arrow area used and the position of the arrows.

draw_drainage_line	*0*	// don't draw drainage line arrow
	1	// draw drainage line arrow
drainage_line_arrow_area	*m*	// the arrow area for the arrows // m = 1,2,3 or 4
drainage_line_y	*mm*	// distance that the arrow is above the // bottom of the arrow area.

Parameters for the title text on the left hand side of the plot:

drainage_line_title	*text*	// title on the left hand side of the arrow
drainage_line_title_textstyle	*textstyle*	// textstyle of the title
drainage_line_title_text_size	*mm*	// size of the title
drainage_line_title_text_colour	*colour*	// colour of the title
drainage_line_title_offset	*mm*	// distance to raise or lower the text // from the arrow position
drainage_line_title_x	*mm*	// distance from the left hand side of // the plot

Parameters for the arrow type:

drainage_line_arrow_mode	*0*	// no arrow
	1	// arrow
	2	// line
	3	// line with uprights at ends
	4	// uprights, no line
	5	// line with downrights
	6	// downrights, no line
	7	// line with up and downrights at ends
	8	// up and downrights, no line
drainage_line_arrow_colour	*colour*	// colour of the arrow
drainage_line_arrow_size	*mm*	// height of the arrow
drainage_line_arrow_gap	*0*	// no gap in arrow
	1	// leave gap in arrow for text

Parameters for the text on the arrow:

drainage_line_arrow_pre_text	*text*	// text before the arrows text
drainage_line_arrow_post_text	*text*	// text after the arrows text
drainage_line_arrow_textstyle	*textstyle*	// textstyle of arrow text
drainage_line_arrow_text_size	*mm*	// size of the text
drainage_line_arrow_text_colour	*colour*	// colour of the text
drainage_line_arrow_text_offset	*mm*	// distance to raise or lower the text // from the arrow position

22.9.2.1.13 Parameters for the Arrows for User Defined Pipe Attributes

There can be up to twenty sets of parameters which are used to define arrows for *user defined attributes* on the pipes.

Each *user defined attribute* is referred to by a given *user name* and it is this *user name* that is specified by a parameter and links the set of parameters to the information in the drainage string.

Note: *user defined attributes* on drainage pipes are usually set by special macros.

The parameters for the n'th set (n=1,2, 20) are:

Parameters specifying the pipe attribute used for this set of arrows and the arrow area used.

draw_pipe_attr_n	*0*	// don't draw nth set of arrows
	1	// draw the nth set of arrows
pipe_attr_n_name	*text*	// the name of the attribute to be labelled // on the n'th set of arrows
pipe_attr_n_arrow_area	*m*	// the arrow area for the arrows // m = 1,2,3 or 4
pipe_attr_n_y	*mm*	// distance that the arrow is above the // bottom of the arrow area.

Parameters for the title text on the left hand side of the plot:

pipe_attr_n_title	*text*	//title on the left hand side of the arrow
pipe_attr_n_title_textstyle	*textstyle*	// textstyle of the nth attribute's title
pipe_attr_n_title_text_size	*mm*	// size of the nth attribute's title
pipe_attr_n_title_text_colour	*colour*	// colour of the nth attribute's title
pipe_attr_n_title_offset	*mm*	//distance to raise or lower the text // from the nth arrow position
pipe_attr_n_title_x	*mm*	// distance from the left hand side of the // plot

Parameters for the arrow type:

pipe_attr_n_arrow_mode	*0*	// no arrow
	1	// arrow
	2	// line
	3	// line with uprights at ends
	4	// uprights, no line
	5	// line with downrights
	6	// downrights, no line
	7	// line with up and downrights at ends
	8	// up and downrights, no line
pipe_attr_n_arrow_colour	*colour*	// colour of the nth arrow
pipe_attr_n_arrow_size	*mm*	// height of the nth arrow
pipe_attr_n_arrow_gap	*0*	// no gap in arrow
	1	// leave gap in arrow for text

Parameters for the text on the arrow:

pipe_attr_n_arrow_pre_text	*text*	// text before the arrows text
pipe_attr_n_arrow_post_text	*text*	// text after the arrows text

pipe_attr_n_arrow_decimals	*integer*	// Only if the attribute is a real number.
		// The number of decimal places used
		// when writing out the value of the
		// attribute.
		// If > 0, all trailing zeros after the
		// decimal place are removed.
		// If < 0, the absolute value is taken as
		// the number of decimal places and **no**
		// trailing zeros are removed after the
		// decimal point.

```
pipe_attr_n_arrow_textstyle     textstyle    // textstyle of arrow text
pipe_attr_n_arrow_text_size     mm           // size of the text
pipe_attr_n_arrow_text_colour   colour       // colour of the text
pipe_attr_n_arrow_text_offset   mm           // distance to raise or lower the text
                                             // form the nth attribute's arrow position
```

22.9.2.1.14 Graph Area

The **graph area** is the area where the actual plots of the drainage strings are drawn.

The length of the graph area is determined by the length of the drainage string to be plotted, the horizontal scale (given by scale) of the plot and the sheet width.

scale *value* // 1:value - horizontal scale
// The vertical scale is determined by the
// horizontal scale and the vertical
// exaggeration for the section view.

The vertical exaggeration is taken from the section view specified for the plot by the parameter:

view_name *text*

The tins to be sectioned through by the drainage strings and any service models and corridor settings for the graph area are also taken from the specified section view.

The height of the graph is **calculated** by subtracting the height of the other nine boxes from the plot height.

If the plot will not fit horizontally into a row on the sheet, then the plot will be broken at an appropriate pit and the plot continued on another row.

Datum breaks at pits are used to try and fit the plot vertically into the graph area but if the plot still cannot fit, then it will be truncated at the bottom.

The types of strings that can be drawn in the graph area of the drainage long section plot are:

(a) drainage string — the strings from the network model

(b) tins — sections of the drainage string through any tins on the section view.

(c) hgl — hydraulic grade line values from the drainage string

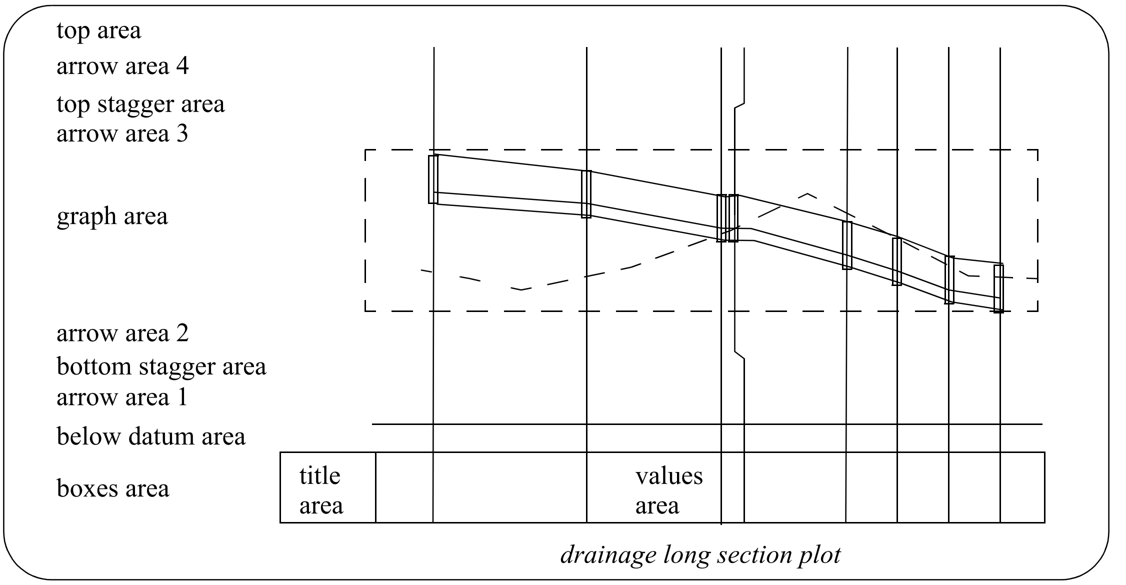
(d) services — parts of strings from any models on the section view that cut the corridor for the section view.

Whether to draw the hydraulic grade line is controlled by:

draw_hgl_diag *0/1* // 1 = draw hydraulic grade line

The **colour** of the strings in the plot is the actual string colour for case (a) and (d), the colour of the tin used for the section in case (b) and user specified colour for (c).

hgl_colour *colour*



drainage long section plot

22.9.2.1.15 Top Area

The **top area** is an annotation area above the arrow_4_area at the top of the plot.

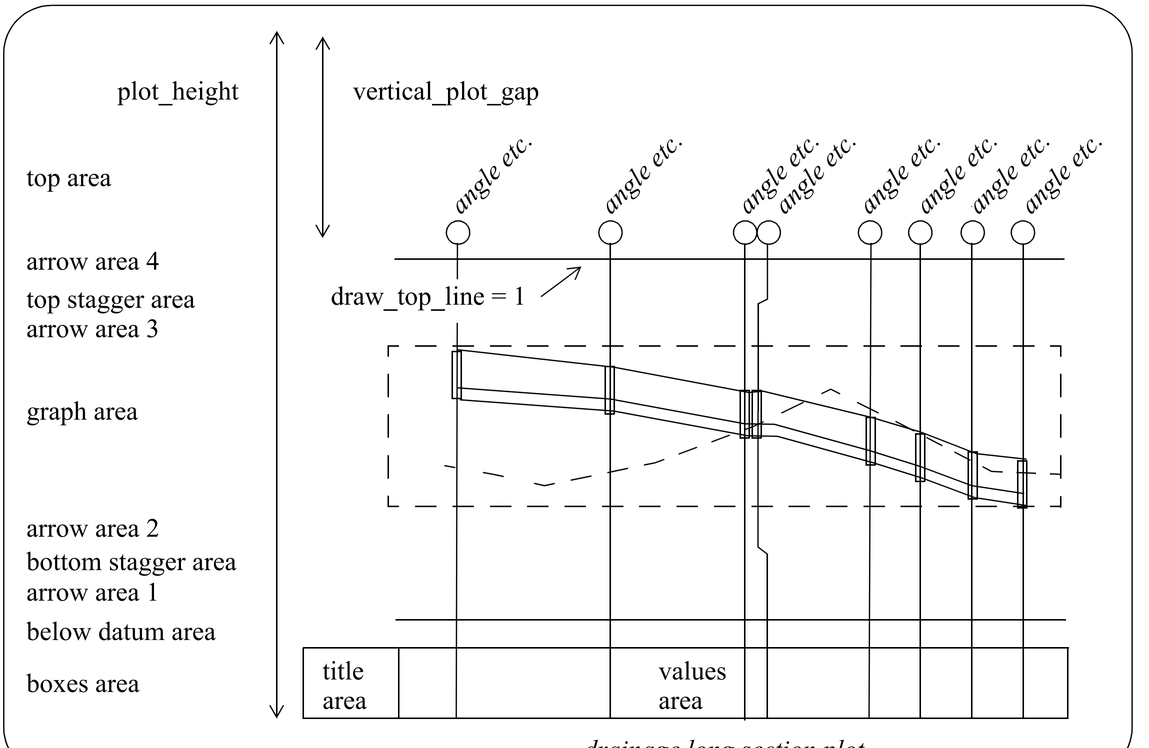
It can be used for pit names, pit types, line names, junctions and deflection angles through pits.

The height of the top area is given by

vertical_plot_gap	*mm*	// size of the top area. If it is not large // enough, the text will over write the // plot in the row above.

A line can be drawn at the bottom of the top area:

draw_top_line	*0/1*	// 1= draw line at bottom of top area // (*i.e.* top of arrow_4_area)



drainage long section plot

22.9.2.1.16 Manhole name, Manhole Types and Surrounding Bubbles

The drainage string name, the manhole name (pit number) and manhole type (pit type) can be drawn on the drainage long section plot, with or without a bubble around it.

The bubble text is made up of

line_name / pit_name   pit_type

where the bits actually used are controlled by

draw_line_name	*0/1*	// 1 = include line name. The "/" is only // included if the line_name is drawn.
draw_pit_name	*0/1*	// 1 = include pit_name
draw_pit_type	*0/1*	// 1 = include pit type

The size, colour and text style of the bubble text is given by:

bubble_text_size	*mm*	// size of text in bubble def angled_text_size
bubble_text_colour	*colour*	// colour of text def angled_text_colour
bubble_textstyle	*textstyle*	// textstyle for text def angled_text_textstyle
bubble_text_x	*mm*	// horizontal distance from the default // position (at the pit or in top area)
bubble_text_y	*mm*	// vertical distance from the default // position (at the pit or in top area)
bubble_pre_text	*text*	// default ""
bubble_post_text	*text*	// default ""
bubble_text_angle	*value*	// default 0
bubble_text_justify	*value*	// the default is centre justified

A bubble is drawn of radius **bubble_radius** is drawn around the bubble text.

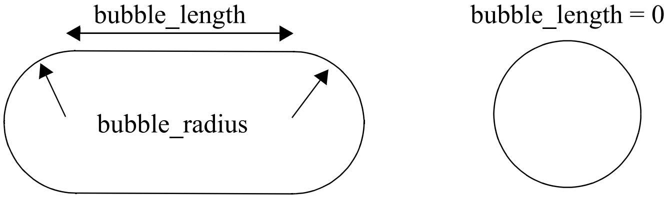
If **bubble_radius** is zero, then no bubble is drawn.

If the *bubble_radius* is zero, the bubble text is still drawn but without the enclosing bubble.

bubble_radius	*mm*	// if non-zero a bubble of this radius is // drawn around the text
bubble_colour	*colour*	// colour of bubbles

The "bubble" can be drawn as two semi-circles and two straight lines

bubble_length	*mm*	// length of the straight line in between // the semi-circles



The bubble text (and bubble) can be positioned at the bottom of the **top area**, or can be placed a fixed distance above the top of the corresponding manhole.

If the bubble text is drawn at the bottom of the top area, the upright is automatically drawn up to the bubble text.

If the bubble text is drawn above the pit, it is positioned by the *distance_above_pit* parameter **plus** the top stagger distances, *stagger_gap_top*, *stagger_height_3* and *stagger_height_4*.

This is necessary because the bubble text may need to be staggered.

draw_text_at_pit	*0*	// draw bubble and/or text in the top

		// area.
	1	// draw it above the pit

distance_above_pit	*mm*	// distance to add to the top stagger area
		// heights to draw the bubble above the
		// pit.

uprights_top_mode	*0/1*	// 1 = draw line from pit to bubble. Used to be // *draw_line_to_text*

22.9.2.1.17 Change of Direction Through Pits and Junctions

If there is a change of direction of the pipes of the plotted drainage string going through the pit, the deflection angle (in degrees, minutes and seconds) is drawn above the bubble text.

The deflection angle text is made up:

angled_pre_text   deflection angle   angled_post_text

Also any junctions at a pit in the plotted drainage string can be labelled with the name of the drainage strings coming into the pit, and the angle of the pipes at the junction.

Hence the junction text is made up of **two** lines:

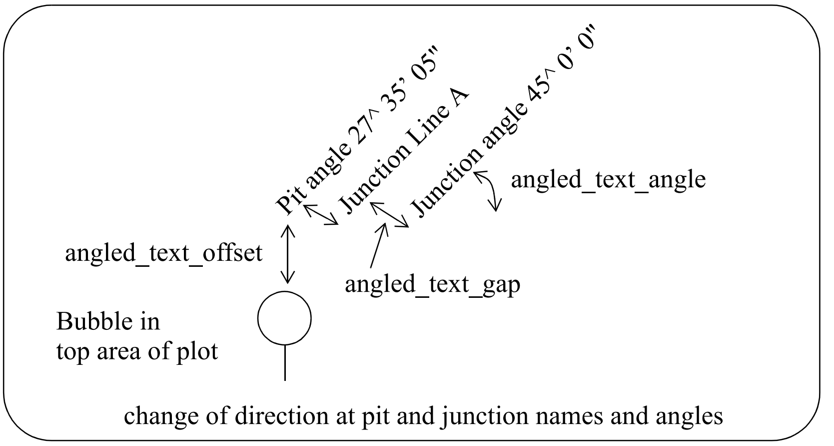
junction_pre_text   line_name   junction_post_text
junction_angle_pre_text   junction angle   junction_angle_post_text

If bubbles are drawn, the deflection angle and junction name and angle are drawn at the distance ***angled_text_offset*** above the bubble and with the distance ***angled_text_gap*** between the lines of information.

If bubbles are not drawn, the bubble text and the deflection angle and junction name and angle is placed the distance ***angled_text_offset*** above bottom of the top area. Each line of text is separated by the distance ***angled_text_gap***.

The deflection angle and junction name and angle are drawn at an angle of ***angled_text_angle***.

angled_text_offset	*mm*	// distance of text above bubble text // or above bottom of top area
angled_text_gap	*mm*	// distance between text to the right // of bubble text.
angled_text_angle	*value*	// angle of text



change of direction at pit and junction names and angles

The pre and post text, colour, size and text style for the deflection angle of the drainage string pipes going through the pit are controlled by:

angled_pre_text	*text*	// default " "
angled_post_text	*text*	// default " "
angled_text_colour	*colour*	// colour of deflection angle text
angled_text_size	*mm*	// size of deflection angle text
angled_textstyle	*textstyle*	// textstyle of deflection angle text

The junction drainage string name can be included/not included in the first line of junction information:

junction_name_mode	*0*	// don't include the junction string name
	1	// default - use junction string name

The pre and post text, colour, size and text style for the junction name and the angle that the junction makes at the pit are controlled by:

junction_pre_text	*text*	// default "JUNCTION "
junction_post_text	*text*	// default " "
junction_text_colour	*colour*	// colour of junction name text
junction_text_size	*mm*	// size of junction name text
junction_textstyle	*textstyle*	// textstyle for junction name text
junction_angle_pre_text	*text*	// default " "
junction_angle_post_text	*text*	// default " "
junction_angle_text_colour	*colour*	// colour of junction angle text
junction_angle_text_size	*mm*	// size of junction angle text
junction_angle_textstyle	*textstyle*	// textstyle for junction angle text

22.9.2.1.18 Symbols at Manholes

Manholes can be labelled with symbols which depend on the manhole type. There can be up to twenty (20) sets of manhole symbols.

manhole_symbol_n_type	*mh_type*	// manhole type to have symbols
manhole_symbol_n_mode	*0*	// cross
	1	// up from centre of box
	2	// up and down from centre of box
	3	// square
	4	// triangle, base at bottom
	5	// circle
	6	// use a 12d symbol

0 + 1 ˈ 2 | 3 □ 4 △ 5 ○

predefined symbols for manhole_symbol_n_*modes*

manhole_symbol_n_position	***1***	// at top of manhole - default
	3	// above top of boxes
	4	// above highest point
	100	// to primary string
	101	// to first found tin
manhole_symbol_n_size	*mm*	// size
manhole_symbol_n_x	*mm*	// x adjustment to position - default 0
manhole_symbol_n_y	*mm*	// y adjustment to position - default is 0
manhole_symbol_n_style	*text*	// the name of the linestyle (symbol)
manhole_symbol_n_angle	*value*	// default value is 0
manhole_symbol_n_colour	*colour*	//

For example:

manhole_symbol_1_type	"CONC COVER"	// manhole type to use
manhole_symbol_1_position	1	// on top of manhole
manhole_symbol_1_mode	6	// use 12d symbol
manhole_symbol_1__style	"shrub"	// name of 12d symbol
manhole_symbol_1_size	1	
manhole_symbol_1_x	0	
manhole_symbol_1_y	0	

22.9.2.1.19 Labelling House Connections

NOTE - house connections are only accessible by the ***Sewer module.***

The house connections can be drawn from the pipe to the house connection level, with a line across the top (a T). The full width the T can be specified by the user.

draw_house_connections	yes/no
house_connection_width	value

The lot name for the house connection and the connection type can be labelled above the house connection. The house connection label is made up as:

connection_pre_text lot_name *connection_mid_text* connection_type *connection_post_text*

where lot_name and connection_type are stored with the house connection on the drainage string.

The parameters controlling the drawing of the house connection label are

house_connection_mode	*0*	// don't label the house connection
	1	// label the house connection with name
		// and type(default)
	2	// label the house connection with name only
connection_pre_text	*text*	// default " "
connection_mid_text	*text*	// default " "
connection_post_text	*text*	// default " "
connection_text_colour	*colour*	// colour of connection label
connection_text_size	*mm*	// size of connection label
connection_textstyle	*textstyle*	// textstyle connection label
connection_text_x	*mm*	// x adjustment to position of text - def 0
connection_text_y	*mm*	// y adjustment to position of text - def 0
connection_text_justify	*text*	// default is Bottom_left
connection_text_angle	*value*	// default vertical 90
connection_text_position	*1*	// above point height value
	3	// above top of boxes
	4	// above highest point
	100	// to primary string
	101	// to first found tin

Label connection type:

(this is used when the house_connection_mode is turned off, only the connection type is labelled)

house_connection_type_mode	*0*	// default, don't label connection type
	1	// label connection type
connection_type_text_x	*mm*	// x adjustment to position of text - def 0
connection_type_text_y	*mm*	// y adjustment to position of text - def 0
connection_type_pre_text	*text*	// text before type - def " "
connection_type_post_text	*text*	// text after type - def " "
connection_type_text_size	*mm*	// def box_text_size
connection_type_text_colour	*colour*	// def box_text_colour
connection_type_textstyle	*text*	//
connection_type_text_justify	*text*	//
connection_type_text_angle	*value*	// def vertical 90
connection_type_text_position	*1*	// above point height value - default
	3	// above top of boxes
	4	// above highest point
	100	// to primary string
	101	// to first found tin

Label the distance from the down stream pit to the house connection

house_connection_ds_pit_mode	*0*	// default, don't label chainage
	1	// label from down stream
	2	// label from up stream
	3	// label from left
	4	// label from right
connection_ds_pit_text_x	*mm*	// x adjustment to position of text
connection_ds_pit_text_y	*mm*	// y adjustment to position of text
connection_ds_pit_pre_text	*text*	// ("") text before type
connection_ds_pit_post_text	*text*	// ("") text after type
connection_ds_pit_text_size	*mm*	// def box_text_size
connection_ds_pit_text_colour	*colour*	//def (box_text_colour
connection_ds_pit_textstyle	*text*	//

connection_ds_pit_text_justify	*text*	// default Bottom_Left
connection_ds_pit_text_angle	*value*	// default is vertical (90)
connection_ds_pit_text_position	***1***	// above point height value - default
	3	// above top of boxes
	4	// above highest point
	100	// to primary string
	101	// to first found tin
connection_ds_pit_decimals	*value*	// def number_of_decimals

Label the connection depth from the finished surface

house_connection_depth_mode	*1*	//default, label house connection depth
	0	// don't label house connection depth
connection_depth_text_x	*mm*	// x adjustment to position of text - def 0
connection_depth_text_y	*mm*	// y adjustment to position of text - def 0
connection_depth_pre_text	*text*	// text before type - def " "
connection_depth_post_text	*text*	// text after type - def " "
connection_depth_text_size	*mm*	// default box_text_size
connection_depth_text_colour	*colour*	// default box_text_colour
connection_depth_textstyle	*text*	//
connection_depth_text_justify	*text*	// default Bottom_left
connection_depth_text_angle	*value*	// default is vertical (90)
connection_depth_text_position	***1***	// above point height value - default
	3	// above top of boxes
	4	// above highest point
	100	// to primary string
	101	// to first found tin
connection_depth_decimals	*value*	// default number_of_decimals

Label the finished surface:

house_connection_fs_mode	*1*	//default, label house connection finished // surface
	0	// don't label house con finished surface
connection_fs_text_x	*mm*	// x adjustment to position of text - def 0
connection_fs_text_y	*mm*	// y adjustment to position of text - def 0
connection_fs_pre_text	*text*	// text before type -def " "
connection_fs_post_text	*text*	// text after type - def " "
connection_fs_text_size	*mm*	// default box_text_size
connection_fs_text_colour	*colour*	// default box_text_colour
connection_fs_textstyle	*text*	//
connection_fs_text_justify	*text*	// default Bottom_left
connection_fs_text_angle	*value*	// default (vertical 90)
connection_fs_text_position	*1*	// above point height value - default
	3	// above top of boxes
	4	// above highest point
	100	// to primary string
	101	// to first found tin

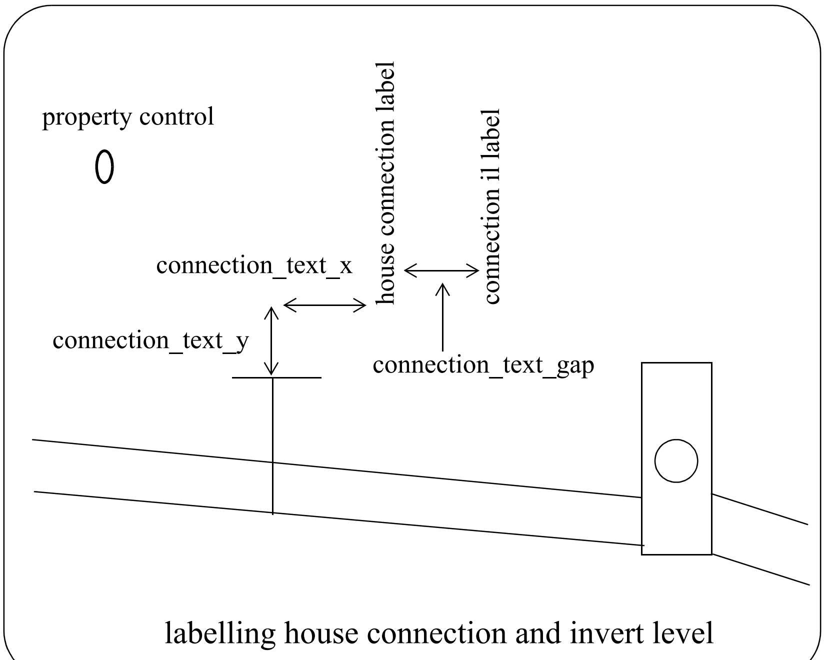
connection_fs_decimals	*integer*	// default number_of_decimals
		// If > 0, all trailing zeros after the
		// decimal place are removed.
		// If < 0, the absolute value is taken as
		// the number of decimal places and **no**
		// trailing zeros are removed after the
		// decimal point.

The invert level of the house connection can also be plotted. The house connection invert level label is made up as:

connection_il_pre_text connection_invert_level *connection_il_post_text*

The parameters controlling the drawing of the house connection invert level label are:

house_connection_il_mode	*1*	// default, label the house connection il
	0	// don't label the house connection il
connection_il_text_position	***1***	// above point height value - default
	3	// above top of boxes
	4	// above highest point
	100	// to primary string
	101	// to first found tin
connection_il_text_x	*mm*	// x adjustment to position of text - def 0
connection_il_text_y	*mm*	// y adjustment to position of text - def 0
connection_text_gap	*mm*	// distance from connection label and il label
connection_il_decimals	*integer*	// def -3, number of decimal places in il value
		// >0 drop trailing zeros after decimal point
		// < 0 keep trailing zeros
connection_il_pre_text	*text*	// default " ", text before type
connection_il_post_text	*text*	// default " ", text after type
connection_il_text_colour	*colour*	// colour of connection il -def box_text_colour
connection_il_text_size	*mm*	// size of connection il - def box_text_size
connection_il_textstyle	*textstyle*	// textstyle connection il
connection_il_text_justify	*text*	//
connection_il_text_angle	*value*	// (vertical 90)



labelling house connection and invert level

22.9.2.1.20 Symbols at Property Controls

NOTE - property controls and house connections are only accessible by the ***Sewer module.***

The position of the property controls can be drawn as circles (which will display as an oval depending on the vertical exaggeration) with the diameter of the property control.

draw_property_controls yes/no

The position of the property control can also be indicated by placing a symbol at the property control.

The position of the property control at the drainage string can be labelled with symbols. There can be up to twenty (20) sets of property control symbols.

property_control_symbol_n_mode	*0*	// cross
	1	// up from centre of box
	2	// up and down from centre of box
	3	// square
	4	// triangle, base at bottom
	5	// circle
	6	// use a 12d symbol

predefined symbols for property_control_symbol_n_modes 0

property_control_symbol_n_position	***0***	// at bottom (invert) of property control
	1	// at top (obvert) of property control
	2	// at centre (axis) of property control
property_control_symbol_n_size	*mm*	// size
property_controle_symbol_n_x	*mm*	// x adjustment to position - default 0
property_control_symbol_n_y	*mm*	// y adjustment to position - default is 0
property_control_symbol_n_style	*text*	// the name of the 12d symbol
property_control_symbol_n_angle	*value*	// default value is 0
property_control_symbol_n_colour	*colour*	//colour of the symbol

For example:

property_control_symbol_1_position	0	// at bottom of prop control
property_control_symbol_1_mode	6	// use 12d symbol
property_control_symbol_1__style	"shrub"	// name of 12d symbol
property_control_symbol_1_size	1	
property_control_symbol_1_x	0	
property_control_symbol_1_y	0	

22.9.2.1.21 Labelling Property Controls

NOTE - property controls and house connections are only accessible by the ***Sewer module.***

The property control can be labelled with its name where the property control is at the drainage string. The property control name label is made up as:

property_control_pre_text property_control_name *property_control_pre_post_text*

where property_control_name is the stored with the property control on the drainage string.

The parameters controlling the labelling the property control are

property_control_mode	*0*	// don't label the property control with name
	1	// label the property control with name
property_control_pre_text	*text*	// default " "
property_control_post_text	*text*	// default " "
property_control_text_colour	*colour*	// colour of label
property_control_text_size	*mm*	// size of label
property_control_textstyle	*textstyle*	// textstyle label
property_control_text_x	*mm*	// x adjustment to position of text - def 0
property_control_text_y	*mm*	// y adjustment to position of text - def 0
property_control_text_justify	*text*	// default is Bottom_left

property_control_text_angle	*value*	// default vertical 90
property_control_text_position	*1*	// above pc centre height value
	3	// above top of boxes
	4	// above highest point
	100	// to primary string
	101	// to first found tin

Label the distance from the down stream pit to the property control

property_control_ds_pit_mode	*0*	// default, don't label chainage
	1	// label from down stream
	2	// label from up stream
	3	// label from left
	4	// label from right
property_control_ds_pit_text_x	*mm*	// x adjustment to position of text
property_control_ds_pit_text_y	*mm*	// y adjustment to position of text
property_control_ds_pit_pre_text	*text*	// ("") text before type
property_control_ds_pit_post_text	*text*	// ("") text after type
property_control_ds_pit_text_size	*mm*	// def box_text_size
property_control_ds_pit_text_colour	*colour*	//def (box_text_colour
property_control_ds_pit_textstyle	*text*	//
property_control_ds_pit_text_justify	*text*	// default Bottom_Left
property_control_ds_pit_text_angle	*value*	// default is vertical (90)
property_control_ds_pit_text_position	***1***	// above pc centre height value - default
	3	// above top of boxes
	4	// above highest point
	100	// to primary string
	101	// to first found tin
property_control_ds_pit_decimals	*value*	// def number_of_decimals

Label the depth from the finished surface to the invert level of the property control:

property_control_depth_mode	*1*	//default, label property control depth
	0	// don't label pc depth
property_control_depth_text_x	*mm*	// x adjustment to position of text - def 0
property_control_depth_text_y	*mm*	// y adjustment to position of text - def 0
property_control_depth_pre_text	*text*	// text before type - def " "
property_control_depth_post_text	*text*	// text after type - def " "
property_control_depth_text_size	*mm*	// default box_text_size
property_control_depth_text_colour	*colour*	// default box_text_colour
property_control_textstyle	*text*	//
property_control_depth_text_justify	*text*	// default Bottom_left
property_control_depth_text_angle	*value*	// default is vertical (90)
property_control_depth_text_position	***1***	// above pc centre height value - default
	3	// above top of boxes
	4	// above highest point
	100	// to primary string
	101	// to first found tin
property_control_depth_decimals	*value*	// default number_of_decimals

Label the finished surface:

property_control_fs_mode	*1*	//default, label pc finished surface
	0	// don't label pc finished surface
property_control_fs_text_x	*mm*	// x adjustment to position of text - def 0
property_control_fs_text_y	*mm*	// y adjustment to position of text - def 0
property_control_fs_pre_text	*text*	// text before type -def " "
property_control_fs_post_text	*text*	// text after type - def " "
property_control_fs_text_size	*mm*	// default box_text_size
property_control_fs_text_colour	*colour*	// default box_text_colour

```
property_control_fs_textstyle        text        //
property_control_fs_text_justify     text        // default Bottom_left
property_control_fs_text_angle       value       // default (vertical 90)
property_control_fs_text_position    1           // above pc centre height value - default
                                     3           // above top of boxes
                                     4           // above highest point
                                     100         // to primary string
                                     101         // to first found tin
property_control_fs_decimals         integer     // default number_of_decimals
                                                 // If > 0, all trailing zeros after the
                                                 // decimal place are removed.
                                                 // If < 0, the absolute value is taken as
                                                 // the number of decimal places and no
                                                 // trailing zeros are removed after the
                                                 // decimal point.
```

The invert level of the property control at the drainage string can also be plotted. The property control invert level label is made up as:

property_control_il_pre_text property_control_invert_level *property_control_il_post_text*

The parameters controlling the drawing of the property control invert level label are:

```
property_control_il_mode             1           // default, label the pc il
                                     0           // don't label the pc il
property_control_il_text_position    1           // above pc centre height value - default
                                     3           // above top of boxes
                                     4           // above highest point
                                     100         // to primary string
                                     101         // to first found tin
property_control_il_text_x           mm          // x adjustment to position of text - def 0
property_control_il_text_y           mm          // y adjustment to position of text - def 0
property_control_text_gap            mm          // distance from pc name label and il label
property_control_il_decimals         integer     // def -3, number of decimal places in il value
                                                 // >0 drop trailing zeros after decimal point
                                                 // < 0 keep trailing zeros
property_control_il_pre_text         text        // default " ", text before type
property_control_il_post_text        text        // default " ", text after type
property_control_il_text_colour      colour      // colour of pc il -def box_text_colour
property_control_il_text_size        mm          // size of pc il - def box_text_size
property_control_il_textstyle        textstyle   // textstyle pc il
property_control_il_text_justify     text        //
property_control_il_text_angle       value       // (vertical 90)
```

22.9.2.1.22 Hatching Cut and Fill Areas

This option is used to hatch cut and/or fill areas between sets of tins.

For each set, the name of the two tins, the hatch linestyle, colour and separation and whether cut and/or fill regions are required are all user definable.

Up to twenty (20) separate sets of tins be hatched.

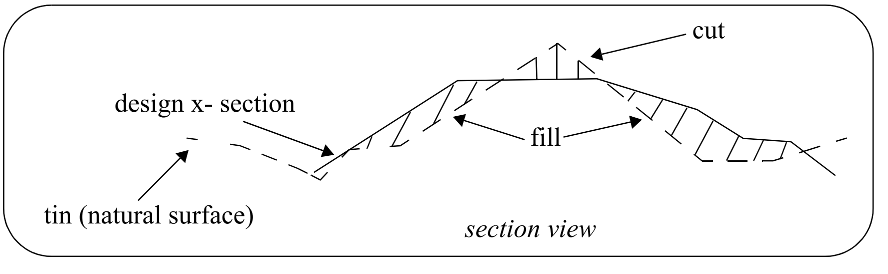
The parameters for labelling cuts and/or fill regions between tins are given by:

```
hatch_original_tin_n        tin_name     // tin_name for original surface
hatch_new_tin_n             tin_name     // tin_name for final surface

hatch_cut_separation_n      mm           // distance between cut hatch lines
                            0            // don't do cut hatching
hatch_cut_angle_n           degrees      // angle in degrees of cut hatching
hatch_cut_colour_n          colour       // colour of the cut hatching
hatch_cut_linestyle_n       linestyle    // linestyle for cut hatching
hatch_cut_draw_sides_n      1/0          // 1 = draw sides of cut regions
hatch_cut_draw_original_n   1/0          // 1 = draw original tin in cut regions
hatch_cut_draw_new_n        1/0          // 1 = draw new tin in cut regions

hatch_fill_separation_n     mm           // distance between fill hatch line
                            0            // don't do fill hatching
hatch_fill_angle_n          degrees      // angle in degrees of fill hatching
hatch_fill_colour_n         colour       // colour of the fill hatching
hatch_fill_linestyle_n      linestyle    // linestyle for fill hatching
hatch_fill_draw_sides_n     1/0          // 1 = draw sides of fill regions
hatch_fill_draw_original_n  1/0          // 1 = draw original tin in fill regions
hatch_fill_draw_new_n       1/0          // 1 = draw new tin in fill regions
```

Notes

(a) cut is when the new tin is below the original tin.
fill is when the new tin is above the original tin.

(b) cut hatching is turned off by setting hatch_cut_separation_n to 0.0.
fill hatching is turned off by setting hatch_fill_separation_n to 0.0.



22.9.2.1.23 Labelling Cuts of Drainage Through Strings in a Model

The cuts that the drainage string makes though any strings in user given models can be automatically labelled on the long section plots.

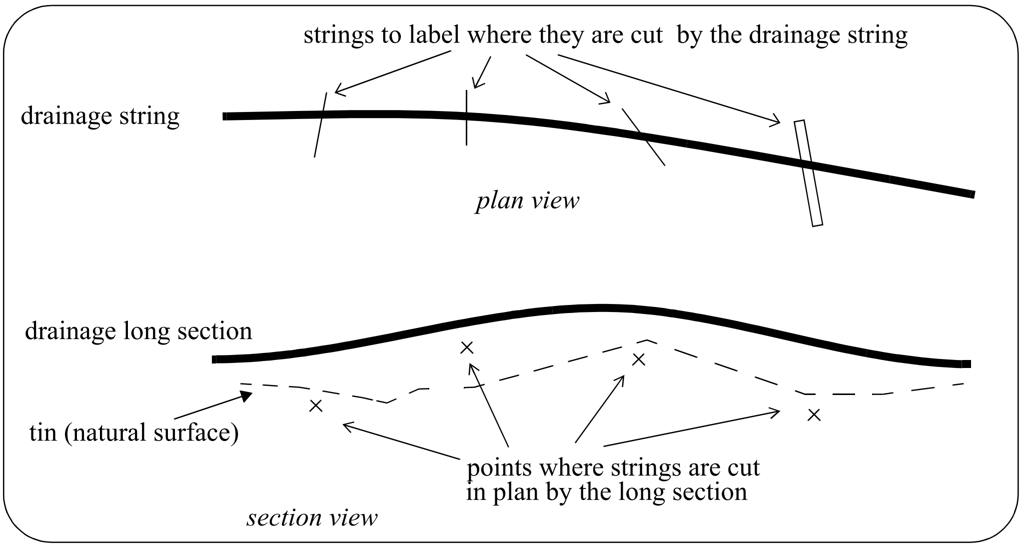
The **height**, **chainage** and **name** of the cut string can be labelled as well as a **symbol** drawn. The height of tins at the same offset value can also be labelled.

The chainage position for the labelling is the chainage of the cut string.

The height position for the labelling can be specified as the

(a) top of the boxes on the long section
(b) height value of the cut string
(c) height of the primary string
(d) height of a tin.

The actual position of the label is defined relative to the above point.



Note:

Only case (b) involves the actual height of the cut string. For all other cases, only the chainage of the cut string is used. Hence for all cases except (b), the string does need to have a sensible height to be used for cuts through strings.

For example, a boundary string may have null heights but only the chainage is required and the height of the tin at that chainage can be used as the height (case (d)).

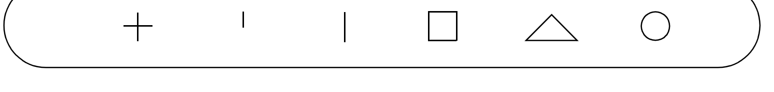
Text justification refers to the actual position and is given by

"top-left"	"top-centre"	"top-right"
"middle-left"	"middle-centre"	"middle-right"
"bottom-left"	"bottom-centre"	"bottom-right"

A choice of six special symbols and/or or any 12d symbols can be drawn at the cut point.

The special 12d Model symbols of size one millimetre are drawn in a square box centred on (0,0) with sides of length two millimetres. That is, the box co-ordinates are (-1,-1), (1,1), (1,-1), (-1,-1).

The six special shapes are



Up to twenty five (25) separate models of strings can be cut and labelled.

22.9.2.1.24 Parameters for Labelling Where the Drainage Line Cuts Strings in a Model

The method for specifying which strings are to be checked for cuts is by first specifying the **model** which contains the strings, and then a **name mask** which is used to restrict the strings in the model to only those whose name matches the name mask.

Up to twenty five different sets of models and name masks can be used so that different cut sets can be labelled in different ways.

The parameters for selecting and labelling the n'th set (where n can be from 1 to 25) of cuts of the design string with the strings in the model are given by:

cuts_n_model	*model_name*	// model of strings to be cut

The selection of the strings from the model *model_name* whose cut points are to be labelled is all the strings whose name satisfies the name mask *cuts_n_mask*:

cuts_n_mask	*name_mask*	// strings to check for cuts
		// and if a cut occurs,
		// parameters show how to
		// label the cut

where *name_mask* is a text string containing the name masks, each separated by one or more spaces, to test the string name against. Each mast can include wild cards and wild characters.

For example

	cuts_1_mask	"ke*"
or	cuts_1_mask	"?bank*"

or, if both masks are required,

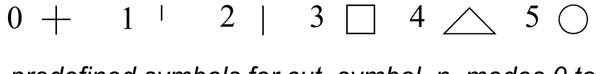
cuts_1_mask	"ke* ?bank*"

If *cuts_n_mask* is missing, then all strings in the model are used. This is equivalent to *name_mask* being "*".

All strings in the model *cuts_n_model* whose name satisfy the name mask *cuts_n_mask* are then checked for cuts with the drainage strings, and if a cut occurs, the cut point will be labelled according to the rest of the parameters in the n'th set.

The parameters for drawing a **symbol** at the cut points are

cuts_symbol_n_mode	0	// cross
	1	// up from centre of box
	2	// up and down from centre of box
	3	// square
	4	// triangle, base at bottom
	5	// circle
	6	// use a 12d symbol



predefined symbols for cut_symbol_n_modes 0 to 5

If *cuts_symbol_n_mode* is 6, then the 12d symbol is given by

cuts_symbol_n_style	*plotsymbol*	// 12d symbol to draw at cut

Important Note

The plot symbol of name *plotsymbol* is defined in the file given by:

(a) the parameter *plot_symbols* in the ppf file

plot_symbols *filename*

or if plot_symbols is not defined, then

(b) in the file pointed to by the environment variable PLOT_SYMBOLS_4D

PLOT_SYMBOLS_4D *filename* *// default plotsym.4d*

or if PLOT_SYMBOLS_4D is not defined, then

(c) in the file *plotsym.4d*

which is searched for in the standard set up file sequence

If none of the above files are defined, or if the symbol does not exist in the above files, then it will be searched for in the standard 12d symbols file which is:

(d) either pointed to by the environment variable SYMBOLS_4D

SYMBOLS_4D filename // default symbols.4d

or if the environment variable SYMBOLS_4D does not exist, in the file, *symbols.4d*

The position of the symbol or plot symbol is given by:

cuts_symbol_n_position	1	// above point height value
	3	// above top of boxes
	100	// to primary string
	101-500	// to tin1 or tin2, *etc.*

The symbol can be adjusted by the parameters:

cuts_symbol_n_x	*mm*	// offset adjustment to position
cuts_symbol_n_y	*mm*	// height adjustment to position
cuts_symbol_n_angle	*degrees*	// rotation about point
cuts_symbol_n_colour	*colour*	// colour of symbol

and for all values of *cuts_symbol_n_mode* **other** than 6:

cuts_symbol_n_size	*mm*	// size of symbol, 0 don't draw

The value of the **chainage** of the cut string can be labelled using the parameters

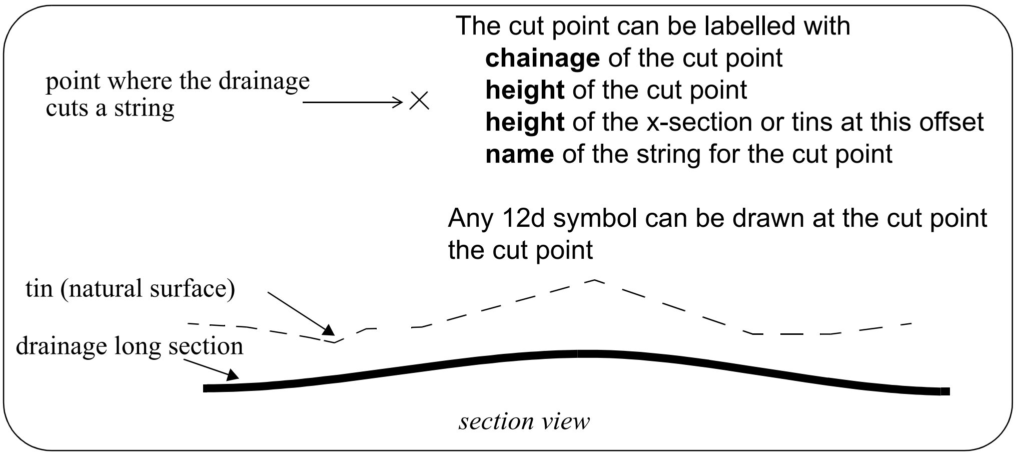
cuts_chainage_n_position	1	// above cut strings height value
	3	// above top of boxes
	100	// to primary string
	101-500	// to tin1 or tin2, *etc.*
cuts_chainage_n_x	*mm*	// chainage adjustment to position
cuts_chainage_n_y	*mm*	// height adjustment to position
cuts_chainage_n_angle	*degrees*	// rotation about point
cuts_chainage_n_size	*mm*	// size of text, 0 don't label
cuts_chainage_n_colour	*colour*	// colour of text
cuts_chainage_n_textstyle	*text*	// textstyle of text chainage
cuts_chainage_n_pre_text	*text*	// text before the chainage value
cuts_chainage_n_post_text	*text*	// text after the chainage value
cuts_chainage_n_justification	*justification*	// justification of the text
cuts_chainage_n_no_decimals	*integer*	// number of decimals in chainage

The value of a **height** at the chainage of the point can be calculated and labelled using the parameters

cuts_height_n_mode	*1*	// use height of cut point itself
	3	// use real world height of position
		// above boxes
	100	// height of primary string
	101-500	// use height of to tin1 or tin2, *etc.*
cuts_height_n_position	*1*	// at points position
	3	// above top of boxes
	100	// to primary string
	101-500	// to tin1 or tin2, *etc.*
cuts_height_n_x	*mm*	// chainage adjustment to position
cuts_height_n_y	*mm*	// height adjustment to position
cuts_height_n_angle	*degrees*	// rotation about point
cuts_height_n_size	*mm*	// size of text, 0 don't label
cuts_height_n_colour	*colour*	// colour of text
cuts_height_n_textstyle	*text*	// textstyle of text height
cuts_height_n_pre_text	*text*	// text before the height value
cuts_height_n_post_text	*text*	// text after the height value
cuts_height_n_justification	*justification*	// justification of the text
cuts_height_n_no_decimals	*integer*	// number of decimals in height

A **label** which can include the **name** of the cut string is drawn by using the parameters

cuts_label_n_position	*1*	// above cut strings height value
	3	// above top of boxes
	100	// to primary string
	101-500	// to tin1 or tin2, *etc.*
cuts_label_n_mode	*0*	// don't include cut string name
	1	// include cut string name in label
cuts_label_n_x	*mm*	// chainage adjustment to position
cuts_label_n_y	*mm*	// height adjustment to position
cuts_label_n_angle	*degrees*	// rotation about point
cuts_label_n_size	*mm*	// size of text, 0 don't label
cuts_label_n_colour	*colour*	// colour of text
cuts_label_n_textstyle	*text*	// textstyle of text label
cuts_label_n_pre_text	*text*	// text before the string name
cuts_label_n_post_text	*text*	// text after the string name
cuts_label_n_justification	*justification*	// justification of the text


The cut point can be labelled with
chainage of the cut point
height of the cut point
height of the x-section or tins at this offset
name of the string for the cut point
point where the drainage cuts a string
Any 12d symbol can be drawn at the cut point the cut point
tin (natural surface)
drainage long section
section view

22.9.2.1.25 Title Block Information

The plot can have a standard ***12d Model*** title block or a user defined title block.

The standard title block consists of a simple border around the plot and two lines of text in a box underneath the plot. For a user defined title block, all the line work and text is defined by the user.

Standard Title Block

For the standard ***12d Model*** title block, there are extra parameters for two lines of text and text size and colour. The standard title block is turned on or off by the parameter *plot_border*.

plot_border	*yes/no*	*// yes plots a standard title block* *// default yes*
title_1	*text*	
title_2	*text*	
title_text_size	*value*	
title_colour	*colour*	

User Title Block

For the user defined title block, the title block drawing commands are kept in a file whose name is supplied by the user. The title block drawing commands are almost identical to the linestyle drawing commands and is given at the beginning of chapter Advanced Plotting.

Hence for a user defined title block, there are just two parameters - one to say a title block file is being used and the other to give the name of the title block file. The *plot_border* parameter should also be set to *no* so that the standard title block is not also drawn.

use_title_file	*yes/no*	*// yes draws the title block given in title_file* // default no
title_file	*filename*	
plot_border	*no*	*// turn off standard title block*

Some special plot parameters are used to pass information down to variables in a user defined title block

For example, inside the title block file it is possible to have runtime user defined text variables. The actual text values for these text variables are passed down to the title block file from the plot parameter file via the parameters user_text_n (n = 1,2,... 1000)

user_text_n	*text*

The special plot parameters are:

time_format	*text*	// format for $time
user_text_n	*text*	// where n = 1,2,... 1000 // passed down to $user_text_n
title_1	*text*	// passed down to $title_1
title_2	*text*	// passed down to $title_2
start_page_number	*integer*	// used as the starting value for // $page_number. If missing, // $page_number starts at 1.

start_drawing_number	*integer*	// added to $drawing_number in title
		// block file. If missing,
		// $drawing_number starts at 1.

drawing_number_prefix	*text*	// passed down to // $drawing_number_prefix
drawing_number_postfix	text	// passed down to // $drawing_number_postfix

22.9.2.1.26 Parameters that Modify Fields In the Plot Drainage Network Panel

A number of parameters match those in the **plot drainage network** panel.

When the plot parameter file is first read, any parameters in the panel will be replaced by the values of any corresponding parameters from the parameter file.

However, if the parameter is subsequently modified in the panel, the panel value will be the value used for the parameter.

The plot parameters that also occur in the **plot drainage network** panel are:

network_model	*text*
scale	*value*
plotter_type	*text*
plot_stem	*text*
view_name	*text*
sheet_size	*text* or "*width* *height*"
plot_height	*mm*
x_origin	*mm*
y_origin	*mm*
global_textstyle	*textstyle*
box_text_size	*mm*
box_colour	*colour*
plot_border	*yes/no*
title_1	*text*
title_2	*text*
title_text_size	*value*
title_text_colour	*colour*
use_title_file	*yes/no*
title_file	*filename*

22.9.3 Melbourne Water

Position of menu: **Design =>Drainage-Sewer=>Plots=>Melbourne Water**

The **Melbourne Water** option is used to generate the longsection plots for all lines in a drainage network to the Melbourne Water sewer standards.

Given the plot sheet size and the horizontal and vertical scales, the longsections for the drainage lines are plotted starting at the top of the sheet and moving across the sheet. Once one row is full, if there is room the plot moves down the page and begins a new row. When a plot sheet is full, a new plot sheet is automatically begun.

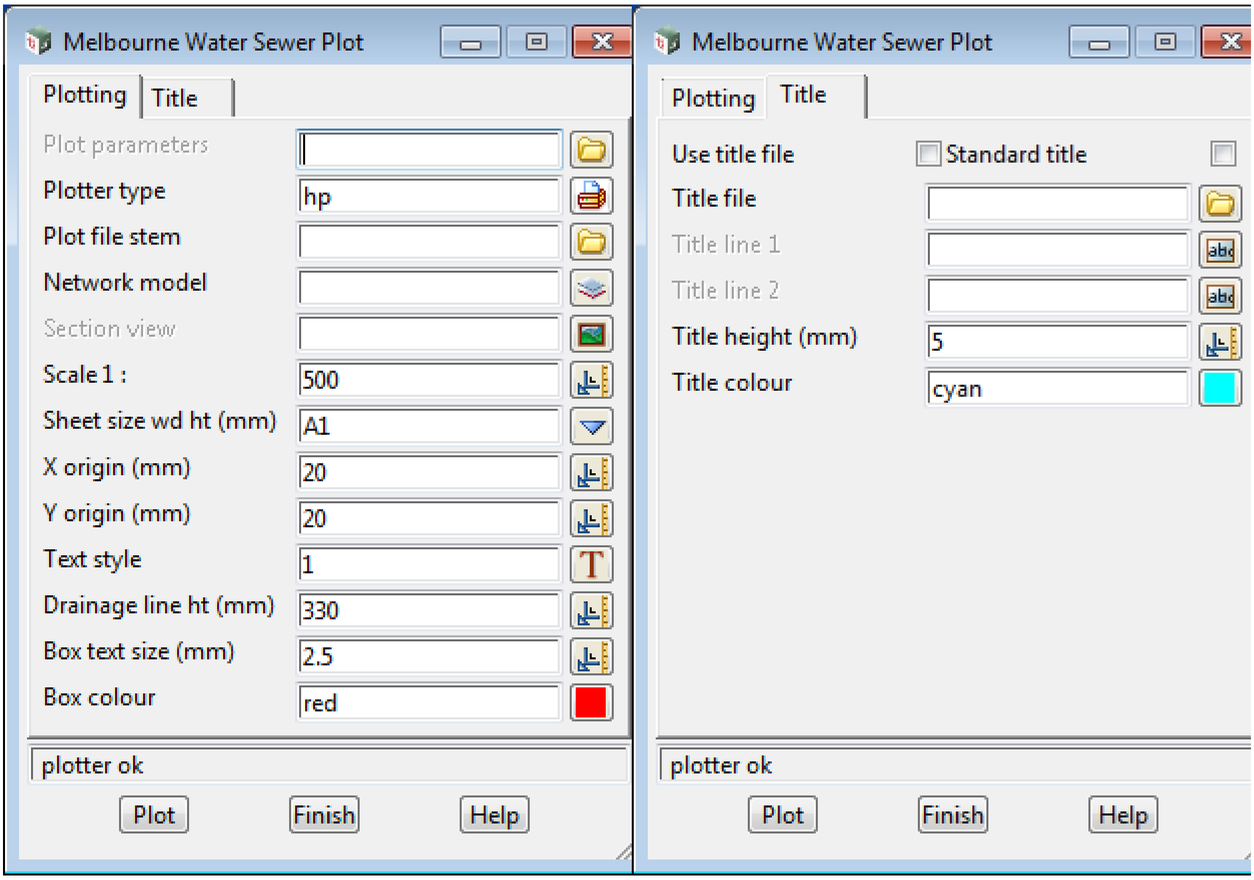
Hence the drainage lines are plotted one after another on one or more plotter sheets.

The drainage lines are plotted in string name alphabetical order.

The drainage longsection plot includes

- the manholes, drainage pipe and any house connections
- the height of the natural surface at the manhole
- manhole names and cover types
- distances between manholes
- the invert depth of the pipe on either side of a manhole
- the grades and types of the pipes
- any services in the corridor - including their name, invert level and distance from the nearest downstream manhole

After selecting the **Melbourne Water** option, the **Melb plot drainage network** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Plot parameters	input		*.ppf

file of plot parameters used for extra control of the long section plot.

Plotter type	input	hp	hp, dxf, postscript, *etc.*

format for the plot output.

Network model input

the model containing all the drainage lines for the network.

Section view input

the section view to be used to define the vertical exaggeration, corridor widths, tins to section through, services models to section, etc.

Plot file stem input

since more than one plot page may be produced, the plot file names are constructed from the stem plus a plot page sequence number, followed by the appropriate plotter type ending.

Scale 1: input

horizontal scale for plotting the drainage long section. The vertical exaggeration is taken from the section view given in the section view *field.*

Sheet size wd ht (mm) input available sheet sizes

the width and height values (separated by space) or the name of a user defined sheet size.

X origin (mm) input

the x position on the plot sheet for the bottom left hand corner of the longsection plots.

Y origin (mm) input

the y position on the plot sheet for the bottom left hand corner of the longsection plots.

Drainage line ht (mm) input boxes boxes, centreline

the maximum allowable height for a longsection plot for a drainage line. Datum breaks are applied to any part of the longsection that will not fit into the drainage line ht.

Box text size (mm) input 3

size (in millimetres) to plot the chainages, heights, etc. in the boxes in the plots of the drainage longsections

Box colour input cyan available colours

colour used for the text and the boxes.

Plot button

plot the drainage longsections for the drainage lines in the model given in the network model *field.*

The fields and buttons in *title* tab are:

Use title file tick box

*if **tick,** a user defined title block file is used.*

Standard Title tick box tick

*if **tick**, the standard 12d Model border and two lines of title are placed on the bottom of the plot*

Title file input *.tf

*if **non-blank** and* use title file *is set to **tick**, then the file given in this field is used to generate a user defined title block for the plot.*

Title line 1/2 input

first/second line of title information

Title height (mm) input 5

height (in millimetres) to draw the characters in the two lines of title information.

Title colour input cyan available colours

colour used for the border and the title information.

Please continue to the next section Melbourne Water Plot Parameter File.

22.9.3.1 Melbourne Water Plot Parameter File

The **Melbourne Water plot** option is used to make special long section plots for a **network** of new drainage strings to Melbourne Water sewer standards.

Some of the look of the long section plot can be controlled from the **Melb plot drainage network** panel itself, however a wider selection of control parameters is available by using a Melbourne Water long plot, plot parameter file.

Because the Melbourne Water format is so fixed, most of the parameter have sensible default values and can be left out. However, for completeness, all the parameters are given.

The Melbourne Water Sewer (MWS) long section plot parameters are placed in a file with ending **.ppf**.

Each parameter consists of a parameter name followed by one or more spaces and then the parameter value. There is only one parameter per line.

Anything on a line after a double forward slash **//** is considered to be a comment.

The set of all parameters for the Melbourne Water sewer long section plot is enclosed within a set of curly brackets **{ }** with the header

melb_water_sewer_long_plot "plot set name"

before the curly brackets.

That is,

melb_water_sewer_long_plot "plot set name" **{**

plot parameters
one per line

}

If there is more than one melb_water_sewer_long_plot parameter set in the file, only the first set is used.

There may also be parameter sets for other plot types such as section_x_plot in the same file. The other sets will be ignored when doing a Melbourne Water long section plot.

The plot parameters are documented in following groups:

For the *Plot Sheet layout*, please continue to the section Plot Sheet Layout.

Chainages, staggering and uprights, please continue to the section Chainages, Staggering and Uprights.

Boxes area, please continue to the section Boxes Area.

Datum area, please continue to the section Datum Value.

Arrows area, please continue to the section Arrow Areas.

Graph area parameters, please continue to the section Graph Area.

Top area parameters, please continue to the section Top Area.

Manhole parameters, please continue to the section Manholes.

Junctions parameters, please continue to the section Junctions.

Services parameters, please continue to the section Services.

Property controls and house connection parameters, please continue to the section Property Controls and House Connections.

Symbols at manhole parameters, please continue to the section Symbols at Manholes.

Hatching cut and fill parameters, please continue to the section Hatching Cut and Fill Areas.

Labelling cuts parameters, please continue to the section Labelling Cuts of Drainage Through Strings in a Model.

Title block parameters, please continue to the section Title Block Information.

Panel modifying parameters, please continue to the section Parameters that Modify Fields In the Melbourne Water Network Panel.

22.9.3.1.1 Plot Sheet Layout

The plot sheet is considered to have only positive co-ordinates with the origin (0,0) in the left hand corner. The units for the plot are millimetres.

The overall size of the plot sheet is given by either a defined sheet size, or by the width and height of the plot given in millimetres and separated by one or more spaces.

sheet_size	*text*	// sheet name, or
	"*mm* *mm*"	// sheet size: width height

The sheet size name and width and heights can be specified by the user in a file named sheets.4d which is in the normal set up areas, or is pointed to by the environment variable

SHEET_SIZES_4D	*file*	// file of plotter sheets sizes

The Melbourne Water long section plot will break an individual plot up if it doesn't fit across the sheet. There can be one or more rows of plot on the same sheet.

The top row is done first, followed by the second top row, then the third and so on until the bottom row. If there is only one row, it is considered to be the bottom row.

When a sheet is full, a follow on sheet is created.

As soon as one drainage string is completed, the next drainage string in the network model is plotted beginning on the same row as the previous drainage string and with a horizontal gap of size horizontal_plot_gap between the plots. If there is not enough room on the row to start the next plot, it will begin on a new row.

The position of the left hand bottom corner of the first plot in the bottom row is given by the parameters, x_origin and y_origin.

If there are two or more rows of plots, the position of the first plot in each row is given by adding multiples of the plot_height to the y_origin.

network_model	text	// model of sewer strings
x_origin	mm	// Position of the left hand bottom
y_origin	mm	// corner of first plot in the bottom row.
only_one_line	0	// more than one row on a sheet
	1	// only one row of plot on a sheet
plot_height	mm	// total height of a plot row.
		// It includes the vertical_plot_gap.
horizontal_plot_gap	mm	// gap between plots on same row
vertical_plot_gap	mm	// gap between rows of plots

A textstyle can be specified which is used for all the text in the plot.

global_textstyle	textstyle	//used for all plot text

The Melbourne Water long plot itself consists of nine areas. From the bottom up, they are

boxes, arrow 1, bottom stagger, arrow 2, graph, arrow 3, top stagger, arrow 4, top.

The **boxes area** is where the chainages and various values for the drainage strings are labelled.

The **arrow 1 area** is for drawing arrows where the arrows go between the staggered uprights and below the graph area. The datum line is at the bottom of the arrow 1 area.

The **bottom stagger area** is where the upright line staggers occur before going up from the boxes area to the graph area.

The **arrow 2 area** is for drawing arrows below the graph but where the arrows go between non-staggered uprights.

The **graph area** is the area where the actual plots of the strings are drawn.

The **arrow 3 area** is for drawing arrows above the graph area and where the arrows go between non-staggered uprights.

The **top stagger area** is where the upright line staggers occur above the graph area.

The **arrow 4 area** is for drawing arrows where the arrows go between the staggered uprights and above the graph area.

The **top area** is an annotation area above the arrow 4 area and is used for manhole names, junctions, deflection angles, *etc.*

Although the arrow areas exist, the Melbourne Water plot does not have any arrows in them.

22.9.3.1.2 Chainages, Staggering and Uprights

All of the labelling of the Melbourne Water sewer plot is done at the chainages of the manholes on the drainage line. For example, the invert levels, depths and natural surface of pipes at a manhole are all labelled at the chainage of the manhole.

The labels for these values are done at an angle of zero degrees rather than ninety degrees for the sewer plots.

Uprights, or leader lines, are drawn from the values at the bottom of the plot to the manhole in the graph area.

If the text values are placed at the real chainage positions at the bottom of the plot, text over writing can easily occur if the chainages are very close together.

To prevent such over writing, the text can be **staggered**. That is, if the text is going to over write a previous text value, the next text value is actually moved along until there is no over writing.

Since the text is no longer at the correct chainage position, the uprights to the manholes and services start at the text position and then bend back to the correct chainage position on the plot. The region where the bending occurs is called the stagger area.

For the Melbourne Water plot, there is an area below the graph where the uprights bend backwards from the staggered text position to the real chainage position (bottom stagger area).

And there is a second area above the graph where the uprights bend forwards from the real chainage position to the staggered text position (top stagger area).

Hence annotation above the top stagger area will line up with the staggered values below the bottom stagger area.

The stagger area below the graph area is defined by

stagger_height_1	*mm*	// distance from the top of arrow_area_1 // to the start of the staggers
stagger_height_2	*mm*	// distance over which stagger occurs
stagger_gap_bottom	*mm*	// distance from end of staggers to the // bottom of arrow_area_2

The stagger area above the graph area is defined

stagger_gap_top	*mm*	// distance from the top of arrow_area_3 // to the start of the staggers
stagger_height_3	*mm*	// distance over which stagger occurs
stagger_height_4	*mm*	// distance from end of staggers to the // bottom of arrow_area_4

The distance to be left for text to avoid over writing is:

horizontal_text_gap	*mm*	// minimum distance to leave for text // after manhole values
services_text_gap	*mm*	// minimum distance to leave for text // after service values

When staggering occurs, it is possible for the values area to be longer than the graph area.

The **colour** of the uprights that are drawn from the text to the manholes is given by:

manhole_line_colour	*colou*r	// colour of uprights to the manholes

22.9.3.1.3 Boxes Area

The drainage string values invert level, depth to invert and surface level can be labelled in the Melbourne Water plot with one line of title, and the actual values given at the chainage of each manhole in the drainage string.

The title or **label** for the strings, is drawn in the **labels area** of the **boxes area** and the values are drawn in the **values area** of the **boxes area**.

Consequently the boxes area is made up of rows of text consisting of:

labels followed by the **values** along the string.

Thus the boxes area is built up as a series of individual boxes and the **boxes** area is made up of two areas side by side - the **labels** area and the **values** area.

The order of the boxes from the bottom up is

(a) chainage values
(b) upstream and downstream depth to inverts
(c) upstream and downstream invert levels
(d) natural surface heights

The label text size and colour are specified by

```
plot_title_text_size      mm        // size of label text in boxes
plot_title_text_colour    colour    // colour of label text in boxes
```

The width of the label box is given by the **box_width** parameter and the height of each box is given by **box_height**.

```
box_width     mm    // width of the label boxes.
box_height    mm    // height of each box
```

The label text for each box can be set by

```
plot_title_chainage_name    text    // label for chainages box
plot_title_surface_name     text    // label for the sewer ns values
plot_title_invert_name      text    // label for the invert levels
plot_title_depth_name       text    // label for depth of inverts
```

For the Melbourne Water plots, there are also additional parameters for special Melbourne Water labels. They are

```
plot_title_item_name       text    // label for item line
plot_title_d_and_g_name    text    // label for diameter and grade
plot_title_detail_name     text    // label for detail line at top of plot
```

The pipe type can also be included with the diameter and grade values

```
label_pipe_type    yes/no
```

The **labels area** for the left plot in the bottom row starts at the co-ordinate (x_origin,y_origin) and each row is begun by adding the distance plot_height to the y_origin.

```
x_origin    mm    // x coord of bottom lh corner of bot row
y_origin    mm    // y coord of bottom lh corner of bot row
```

The **values area** starts at the end of the label area.

As for the label boxes, the height of each individual box area is given by the box_height parameter.

The width of the heights area is determined by the number of chainages to be labelled and whether the values are staggered to prevent over writing.

```
box_text_size      mm        // size of value text
box_text_colour    colour    // colour of values in boxes
```

The colour of the box line work is given by:

```
box_colour    colour    // colour of the lines in the boxes
```

The total height of the boxes area is simply given by number of boxes drawn multiplied by the height of one box).

The total width of the boxes area is the width of the labels area plus the width of the values area which depends on the amount of staggering that occurs.

22.9.3.1.4 Datum Value

The datum value is placed above the top of the natural surface heights box.

Since the datum value is automatically calculated to try and fit the plot vertically into the graph area, the datum value can change along the plot. When a datum change occurs, the new datum value is written on the datum line at the relevant chainage position.

The datum name, text size and colour are controlled by:

```
datum_name              text        // text to write before the datum value
datum_text_size         mm          // size of datum text and value
datum_colour            colour      // colour of the datum text and line
datum_value_pre_text    text        // text before the datum value
datum_value_post_text   text        // text after the datum value
                                    // (default " m")
datum_value_decimals    integer     // number of decimal places in the
                                    // datum value.
                                    // If > 0, trailing zeros are removed.
                                    // If < 0, the absolute value is taken as
                                    //   the number of decimal places and no
                                    //   trailing zeros are removed.
```

22.9.3.1.5 Arrow Areas

The **arrow areas** were designed for drawing arrows between manholes and labelling them with values such as pipe slope, pipe diameter, flow, velocity and drainage line name.

These are not required for the Melbourne Water sewer plot.

However, values can still be given to defined the sizes of the arrow areas.

There are four arrow areas:

arrow_1_area which is below the graph and the bottom stagger area. The arrows go between the staggered positions of the manhole chainages.

The datum line is at the bottom of the arrow 1 area.

arrow_2_area which below the graph but above the bottom stagger area. Hence the arrows go between the non-staggered chainage positions of the manholes.

arrow_3_area which is above the graph but below the top stagger area. The arrows go between the non-staggered chainage positions of the manholes.

arrow_4_area is above the graph and the top stagger area. The arrows go between the staggered chainage positions of the manholes.

The heights of the four areas (which can be zero) are defined by:

```
arrow_area_1    mm    // height of arrow_1 area
arrow_area_2    mm    // height of arrow_2 area
arrow_area_3    mm    // height of arrow_3 area
arrow_area_4    mm    // height of arrow_4 area
```

22.9.3.1.6 Graph Area

The **graph area** is the area where the actual plots of the drainage strings are drawn.

The length of the graph area is determined by the length of the drainage string to be plotted, the horizontal scale (given by scale) of the plot and the sheet width.

scale	*value*	// 1:value - horizontal scale
		// The vertical scale is determined by the
		// horizontal scale and the vertical
		// exaggeration for the section view.

The vertical exaggeration is taken from the section view specified for the plot by the parameter:

view_name	*text*

The tins to be sectioned through by the sewer strings and any service models and corridor settings for the graph area are also taken from the specified section view.

The height of the graph is **calculated** by subtracting the height of the other nine boxes from the plot height.

If the plot will not fit horizontally into a row on the sheet, then the plot will be broken at an appropriate manhole and the plot continued on another row.

Datum breaks at manholes are used to try and fit the plot vertically into the graph area but if the plot still cannot fit, then it will be truncated at the bottom.

The types of strings that can be drawn in the graph area of the sewer long section plot are:

(a)	drainage string	the strings from the network model
(b)	tins	sections of the drainage string through any tins on the section view.
(c)	services	parts of strings from any models on the section view that cut the corridor for the section view.

The **colour** of the strings in the plot is the actual string colour for cases (a) and (c), and the colour of the tin used for the section in case (b).

22.9.3.1.7 Top Area

The **top area** is an annotation area above the arrow_4_area at the top of the plot.

It can be used for manhole names, manhole types, pegs, junctions and deflection angles through manholes.

The height of the top area is given by

vertical_plot_gap	*mm*	// size of the top area. If it is not large
		// enough the text will over write the
		// plot in the row above.

22.9.3.1.8 Manholes

The manhole name, manhole type, special manhole symbol and the change of direction of the pipes going through the manhole can all be displayed at the top of the plot.

The manhole name and type are drawn first, followed by the angle of any change of direction of the pipes going through the manhole (the angle is in degrees, minutes and seconds) and finally any junction information.

The text is drawn at the distance **angled_text_y_offset** above the top of the plot, **angled_text_x_offset** to the right of the manhole position and each piece of information is spaced horizontally by the distance **angled_text_gap** to the right of the manhole name if the **angled_text_angle** is zero, or the distance **angled_text_gap** below the previous text if **angled_text_angle** is non-zero.

The size of the manhole names and type is given by **angle_text_size** and the size of the change of direction angle is given by **manhole_angle_text_size**.

The colour of the manhole names, type and change of direction angle is given by **angle_text_colour**.

label_manhole_type	*yes/no*	// include manhole type
angled_text_size	*mm*	// size of manhole name text
manhole_angle_text_size	*mm*	// size of text for change of // angle through manhole
angled_text_x_offset	*mm*	// distance of text above top of the plot
angled_text_y_offset	*mm*	// distance of text to the right of manhole
angled_text_gap	*mm*	// distance of text to the right of previous // text
angled_text_angle	*value*	// angle of text
angled_text_colour	*colour*	// colour of text

The symbol at the manhole is controlled by the manhole name.

manhole_symbol_n_size	*mm*	
manhole_symbol_n_colour	*colour*	
manhole_symbol_n_y	*mm*	// distance above top line

manhole_symbol_n_repeats	*integer*	// draw the symbol this
		// many times with a
		// decreasing size)

manhole_line_colour	*colour*	// colour of line through // manhole

22.9.3.1.9 Junctions

Any junctions at the manhole are also labelled with the name of the lines coming into the manhole.

The junction text is made up of two lines:

JUNCTION LINE   line_name

angle of the junction

The junction text is at the same height as the manhole and angle text and at a distance **angled_text_gap** to the right of the manhole and angle text, or a previous junction label. The size of the text is given by

junction_angle_text_size	*mm*

When a junction does occur, the incoming pipe is drawn at the correct position on the manhole of the main line.

22.9.3.1.10 Services

Any services in the corridor are drawn on the plot and if the service actually cuts the drainage string, then a line is drawn to the cut point and the line labelled with the service name. The name is drawn below the cut point and is right justified.

service_text_size	*m*m	// size of the text for the service name
service_text_x_offset	*m*m	// adjustment to the drawing position
service_text_y_offset	*mm*	// for the service name
service_text_colour	*colour*	// colour of the service name
service_line_colour	*colour*	// colour of the line to the cut point

22.9.3.1.11 Property Controls and House Connections

The position of the property controls can be drawn on the plot and can have a user specified diameter.

draw_property_controls	*yes/n*o
property_control_diameter	*value*

The house connections can be drawn from the pipe to the house connection level, with a line across the top (a T). The full width the T can be specified by the user.

draw_house_connections	*yes/no*
house_connection_width	*value*

22.9.3.1.12 Symbols at Manholes

Manholes can be labelled with symbols which depend on the manhole type. There can be up to twenty (20) sets of manhole symbols.

manhole_symbol_n_type	*mh_type*	// manhole type to have symbols
manhole_symbol_n_mode	*0*	// cross
	1	// up from centre of box
	2	// up and down from centre of box
	3	// square
	4	// triangle, base at bottom
	5	// circle
	6	// use a 12d symbol

predefined symbols for manhole_*symbol_n_modes*

manhole_symbol_n_position	***1***	// at top of manhole - default
	3	// above top of boxes
	4	// above highest point
	100	// to primary string
	101	// to first found tin
manhole_symbol_n_size	*mm*	// size
manhole_symbol_n_x	*mm*	// x adjustment to position - default 0
manhole_symbol_n_y	*mm*	// y adjustment to position - default is 0
manhole_symbol_n_style	*text*	// the name of the 12d symbol
manhole_symbol_n_angle	*value*	// default value is 0
manhole_symbol_n_colour	*colour*	//

For example:

manhole_symbol_1_type	"CONC COVER"	// manhole type to use
manhole_symbol_1_position	1	// on top of manhole
manhole_symbol_1_mode	6	// use 12d symbol
manhole_symbol_1__style	"shrub"	// name of 12d symbol
manhole_symbol_1_size	1	
manhole_symbol_1_x	0	
manhole_symbol_1_y	0	

22.9.3.1.13 Hatching Cut and Fill Areas

This option is used to hatch cut and/or fill areas between sets of tins.

For each set, the name of the two tins, the hatch linestyle, colour and separation and whether cut and/or fill regions are required are all user definable.

Up to twenty (20) separate sets of tins be hatched.

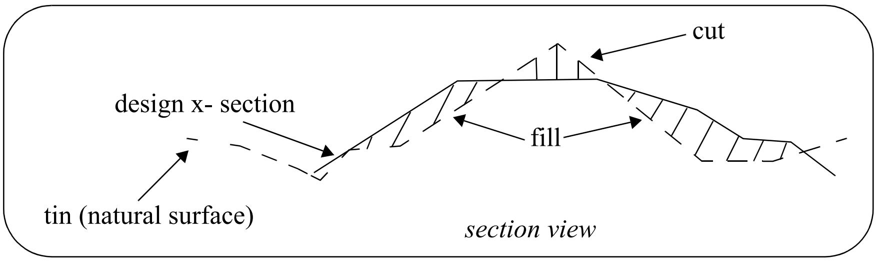
The parameters for labelling cuts and/or fill regions between tins are given by:

hatch_original_tin_n	*tin_name*	// tin_name for original surface
hatch_new_tin_n	*tin_name*	// tin_name for final surface
hatch_cut_separation_n	*mm*	// distance between cut hatch lines
	0	// don't do cut hatching
hatch_cut_angle_n	*degrees*	// angle in degrees of cut hatching
hatch_cut_colour_n	*colour*	// colour of the cut hatching
hatch_cut_linestyle_n	*linestyle*	// linestyle for cut hatching
hatch_cut_draw_sides_n	1/0	// 1 = draw sides of cut regions
hatch_cut_draw_original_n	1/0	// 1 = draw original tin in cut regions
hatch_cut_draw_new_n	1/0	// 1 = draw new tin in cut regions
hatch_fill_separation_n	*mm*	// distance between fill hatch line
	0	// don't do fill hatching
hatch_fill_angle_n	*degrees*	// angle in degrees of fill hatching
hatch_fill_colour_n	*colour*	// colour of the fill hatching
hatch_fill_linestyle_n	*linestyle*	// linestyle for fill hatching
hatch_fill_draw_sides_n	1/0	// 1 = draw sides of fill regions
hatch_fill_draw_original_n	1/0	// 1 = draw original tin in fill regions
hatch_fill_draw_new_n	1/0	// 1 = draw new tin in fill regions

Notes

(a) cut is when the new tin is below the original tin.
fill is when the new tin is above the original tin.

(b) cut hatching is turned off by setting hatch_cut_separation_n to 0.0.
fill hatching is turned off by setting hatch_fill_separation_n to 0.0.



22.9.3.1.14 Labelling Cuts of Drainage Through Strings in a Model

The cuts that the drainage string makes though any strings in user given models can be automatically labelled on the long section plots.

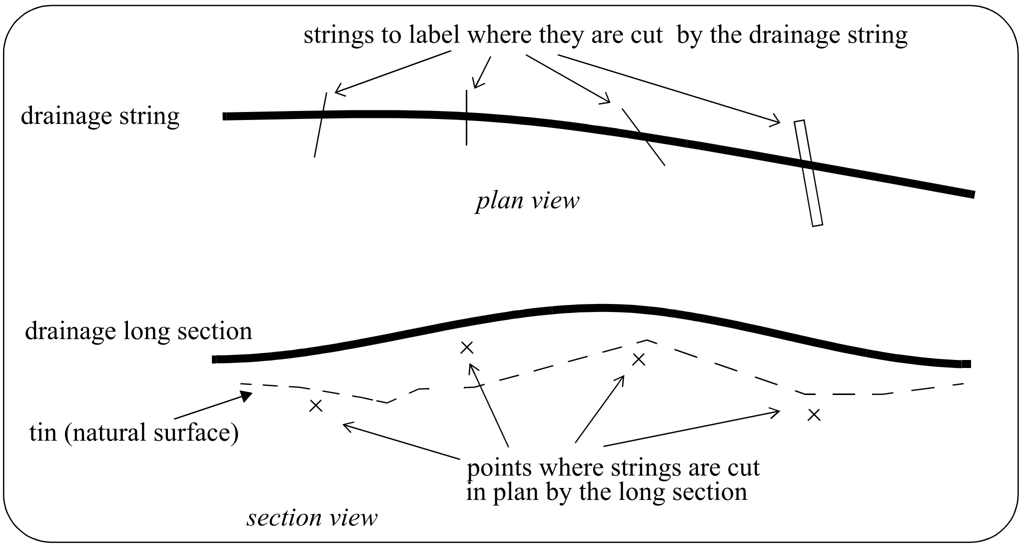
The **height**, **chainage** and **name** of the cut string can be labelled as well as a **symbol** drawn. The height of tins at the same offset value can also be labelled.

The chainage position for the labelling is the chainage of the cut string.

The height position for the labelling can be specified as the

(a) top of the boxes on the long section
(b) height value of the cut string
(c) height of the primary string
(d) height of a tin.

The actual position of the label is defined relative to the above point.



Note:

Only case (b) involves the actual height of the cut string. For all other cases, only the chainage of the cut string is used. Hence for all cases except (b), the string does need to have a sensible height to be used for cuts through strings.

For example, a boundary string may have null heights but only the chainage is required and the height of the tin at that chainage can be used as the height (case (d)).

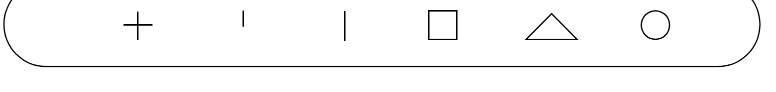
Text justification refers to the actual position and is given by

"top-left"	"top-centre"	"top-right"
"middle-left"	"middle-centre"	"middle-right"
"bottom-left"	"bottom-centre"	"bottom-right"

A choice of six special symbols and/or or any 12d symbols can be drawn at the cut point.

The special 12d Model symbols of size one millimetre are drawn in a square box centred on (0,0) with sides of length two millimetres. That is, the box co-ordinates are (-1,-1), (1,1), (1,-1), (-1,-1).

The six special shapes are



Up to twenty five (25) separate models of strings can be cut and labelled.

22.9.3.1.15 Parameters for Labelling Where the Drainage Line Cuts Strings in a Model

The method for specifying which strings are to be checked for cuts is by first specifying the **model** which contains the strings, and then a **name mask** which is used to restrict the strings in the model to only those whose name matches the name mask.

Up to twenty five different sets of models and name masks can be used so that different cut sets can be labelled in different ways.

The parameters for selecting and labelling the n'th set (where n can be from 1 to 25) of cuts of the design string with the strings in the model are given by:

cuts_n_model	*model_name*	// model of strings to be cut

The selection of the strings from the model *model_name* whose cut points are to be labelled is all the strings whose name satisfies the name mask *cuts_n_mask*:

cuts_n_mask	*name_mask*	// strings to check for cuts
		// and if a cut occurs,
		// parameters show how to
		// label the cut

where *name_mask* is a text string containing the name masks, each separated by one or more spaces, to test the string name against. Each mast can include wild cards and wild characters.

For example

	cuts_1_mask	"ke*"
or	cuts_1_mask	"?bank*"

or, if both masks are required,

cuts_1_mask	"ke* ?bank*"

If *cuts_n_mask* is missing, then all strings in the model are used. This is equivalent to *name_mask* being "*".

All strings in the model *cuts_n_model* whose name satisfy the name mask *cuts_n_mask* are then checked for cuts with the drainage strings, and if a cut occurs, the cut point will be labelled according to the rest of the parameters in the n'th set.

The parameters for drawing a **symbol** at the cut points are

cuts_symbol_n_mode	0	// cross
	1	// up from centre of box
	2	// up and down from centre of box
	3	// square
	4	// triangle, base at bottom
	5	// circle
	6	// use a 12d symbol

predefined symbols for cut_symbol_n_modes 0 to 5

If *cuts_symbol_n_mode* is 6, then the 12d symbol is given by

cuts_symbol_n_style	*plotsymbol*	// 12d symbol to draw at cut

Important Note

The plot symbol of name *plotsymbol* is defined in the file given by:

(a) the parameter *plot_symbols* in the ppf file

plot_symbols *filename*

or if plot_symbols is not defined, then

(b) in the file pointed to by the environment variable PLOT_SYMBOLS_4D

PLOT_SYMBOLS_4D *filename* *// default plotsym.4d*

or if PLOT_SYMBOLS_4D is not defined, then

(c) in the file *plotsym.4d*

which is searched for in the standard set up file sequence

If none of the above files are defined, or if the symbol does not exist in the above files, then it will be searched for in the standard 12d symbols file which is:

(d) either pointed to by the environment variable SYMBOLS_4D

SYMBOLS_4D filename // default symbols.4d

or if the environment variable SYMBOLS_4D does not exist, in the file, *symbols.4d*

The position of the symbol is given by:

cuts_symbol_n_position	1	// above point height value
	3	// above top of boxes
	100	// to primary string
	101-500	// to tin1 or tin2, *etc.*

The symbol can be adjusted by the parameters:

cuts_symbol_n_x	*mm*	// offset adjustment to position
cuts_symbol_n_y	*mm*	// height adjustment to position
cuts_symbol_n_angle	*degrees*	// rotation about point
cuts_symbol_n_colour	*colour*	// colour of symbol

and for all values of *cuts_symbol_n_mode* **other** than 6:

cuts_symbol_n_size	*mm*	// size of symbol, 0 don't draw

The value of the **chainage** of the cut string can be labelled using the parameters

cuts_chainage_n_position	1	// above cut strings height value
	3	// above top of boxes
	100	// to primary string
	101-500	// to tin1 or tin2, *etc.*

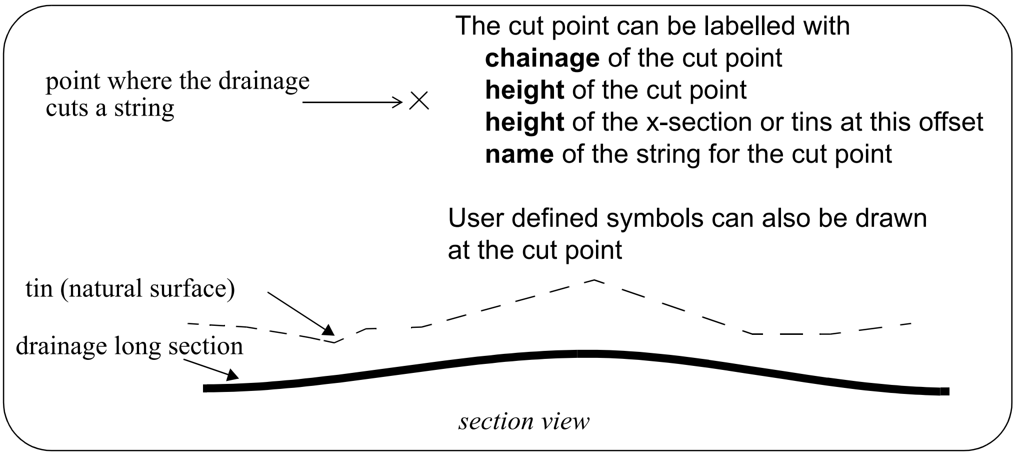
cuts_chainage_n_x	*mm*	// chainage adjustment to position
cuts_chainage_n_y	*mm*	// height adjustment to position
cuts_chainage_n_angle	*degrees*	// rotation about point
cuts_chainage_n_size	*mm*	// size of text, 0 don't label
cuts_chainage_n_colour	*colour*	// colour of text
cuts_chainage_n_textstyle	*text*	// textstyle of text chainage
cuts_chainage_n_pre_text	*text*	// text before the chainage value
cuts_chainage_n_post_text	*text*	// text after the chainage value
cuts_chainage_n_justification	*justification*	// justification of the text
cuts_chainage_n_no_decimals	*integer*	// number of decimals in chainage

The value of a **height** at the chainage of the point can be calculated and labelled using the parameters

cuts_height_n_mode	*1*	// use height of cut point itself
	3	// use real world height of position // above boxes
	100	// height of primary string
	101-500	// use height of to tin1 or tin2, *etc.*
cuts_height_n_position	*1*	// at points position
	3	// above top of boxes
	100	// to primary string
	101-500	// to tin1 or tin2, *etc.*
cuts_height_n_x	*mm*	// chainage adjustment to position
cuts_height_n_y	*mm*	// height adjustment to position
cuts_height_n_angle	*degrees*	// rotation about point
cuts_height_n_size	*mm*	// size of text, 0 don't label
cuts_height_n_colour	*colour*	// colour of text
cuts_height_n_textstyle	*text*	// textstyle of text height
cuts_height_n_pre_text	*text*	// text before the height value
cuts_height_n_post_text	*text*	// text after the height value
cuts_height_n_justification	*justification*	// justification of the text
cuts_height_n_no_decimals	*integer*	// number of decimals in height

A **label** which can include the **name** of the cut string is drawn by using the parameters

cuts_label_n_position	1	// above cut strings height value
	3	// above top of boxes
	100	// to primary string
	101-500	// to tin1 or tin2, *etc.*
cuts_label_n_mode	0	// don't include cut string name
	1	// include cut string name in label
cuts_label_n_x	*mm*	// chainage adjustment to position
cuts_label_n_y	*mm*	// height adjustment to position
cuts_label_n_angle	*degrees*	// rotation about point
cuts_label_n_size	*mm*	// size of text, 0 don't label
cuts_label_n_colour	*colour*	// colour of text
cuts_label_n_textstyle	*text*	// textstyle of text label
cuts_label_n_pre_text	*text*	// text before the string name
cuts_label_n_post_text	*text*	// text after the string name
cuts_label_n_justification	*justification*	// justification of the text


The cut point can be labelled with
chainage of the cut point
height of the cut point
height of the x-section or tins at this offset
name of the string for the cut point
point where the drainage cuts a string
User defined symbols can also be drawn at the cut point
tin (natural surface)
drainage long section
section view

22.9.3.1.16 Title Block Information

The plot can have a standard ***12d Model*** title block or a user defined title block.

The standard title block consists of a simple border around the plot and two lines of text in a box underneath the plot. For a user defined title block, all the line work and text is defined by the user.

Standard Title Block

For the standard ***12d Model*** title block, there are extra parameters for two lines of text and text size and colour. The standard title block is turned on or off by the parameter *plot_border*.

plot_border	*yes/no*	*// yes plots a standard title block* *// default yes*
title_1	*text*	
title_2	*text*	
title_text_size	*value*	
title_colour	*colour*	

User Title Block

For the user defined title block, the title block drawing commands are kept in a file whose name is supplied by the user. The title block drawing commands are almost identical to the linestyle drawing commands and is given at the beginning of chapter Advanced Plotting.

Hence for a user defined title block, there are just two parameters - one to say a title block file is being used and the other to give the name of the title block file. The *plot_border* parameter should also be set to *no* so that the standard title block is not also drawn.

use_title_file	*yes/no*	*// yes draws the title block given in title_file* // default no
title_file	*filename*	
plot_border	*no*	*// turn off standard title block*

Some special plot parameters are used to pass information down to variables in a user defined title block

For example, inside the title block file it is possible to have runtime user defined text variables. The actual text values for these text variables are passed down to the title block file from the plot parameter file via the parameters user_text_n (n = 1,2,... 1000)

user_text_n		*text*

The special plot parameters are:

time_format	*text*	// format for $time
user_text_n	*text*	// where n = 1,2,... 1000 // passed down to $user_text_n
title_1	*text*	// passed down to $title_1
title_2	*text*	// passed down to $title_2
start_page_number	*integer*	// used as the starting value for // $page_number. If missing, // $page_number starts at 1.

start_drawing_number	*integer*	// added to $drawing_number in title
		// block file. If missing,
		// $drawing_number starts at 1.

drawing_number_prefix	*text*	// passed down to // $drawing_number_prefix
drawing_number_postfix	text	// passed down to // $drawing_number_postfix

22.9.3.1.17 Parameters that Modify Fields In the Melbourne Water Network Panel

A number of parameters match those in the **New Melb Plot Drainage Network** panel.

When the plot parameter file is first read, any parameters in the panel will be replaced by the values of any corresponding parameters from the parameter file.

However, if the parameter is subsequently modified in the panel, the panel value will be the value used for the parameter.

The plot parameters that also occur in the **Melb plot drainage network** panel are:

network_model	*text*
scale	*value*
plotter_type	*text*
plot_stem	*text*
view_name	*text*
sheet_size	*text* or "width height"
plot_height	*mm*
x_origin	*mm*
y_origin	*mm*
global_textstyle	*textstyle*
box_text_size	*mm*
box_colour	*colour*
plot_border	*yes/no*
title_1	*text*
title_2	*text*
title_text_size	*value*
title_text_colour	*colour*
use_title_file	*yes/no*
title_file	*filename*

22.9.4 Plot Annotations

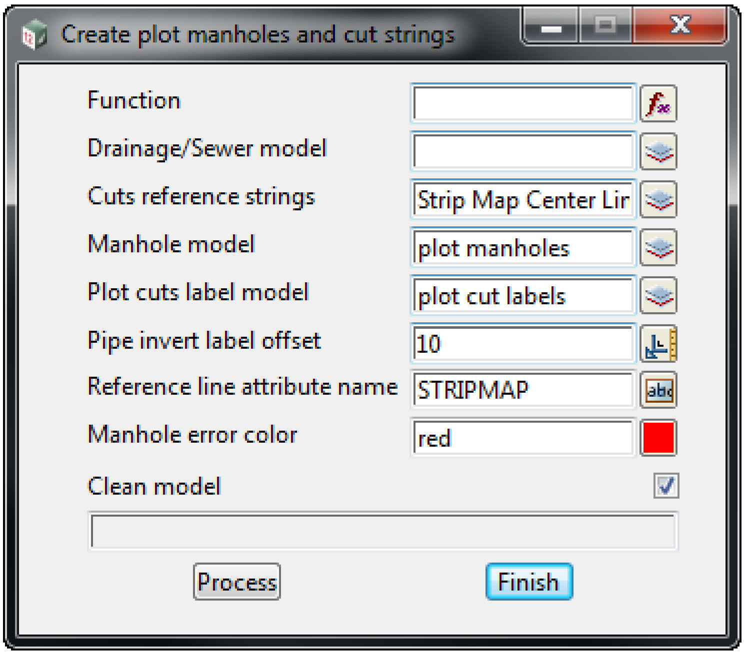
See Drainage Long Plot PPF Editor.

22.9.5 Long Plot Cut Labels and Manholes

Position of menu: **Design =>Drainage-Sewer=>Plots=>Long plot cut labels and manholes**

This section of documentation is a work in progress and will be updated in subsequent releases.

Selecting **Long plot cut labels and manholes** brings up the **Create plot manholes and cut strings** panel.



The fields and buttons used in this panel have the following functions.

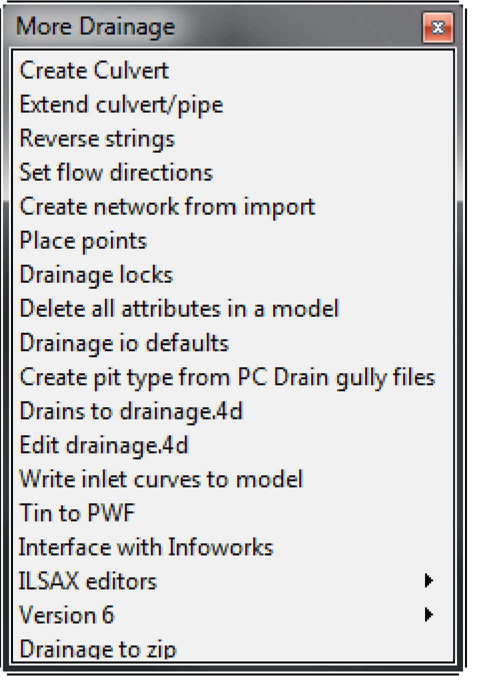
Field Description	Type	Defaults	Pop-Up
Function	function box		
Drainage/Sewer model	model box		
Cuts reference strings	model box		
Manhole model	model box		
Plot cuts label model	model box		
Pipe invert label offset	measure box		
Reference line attribute name	name box		
Manhole error colour	colour box		
Clean model	tick box		
Process	button		

22.10 More Drainage

Position of menu: **Design =>Drainage-Sewer=>More**

The More Drainage menu items contain the drainage import export routines and tools that enhance the basic drainage functions on the Drainage Sewer menu.

The **More Drainage** walk-right menu is:



Frequently Asked Questions (FAQ)

Drainage Overview

The following are the basic steps from creating a urban drainage network to producing your plan/ drainage longsection drawings and pit layout table. The optional catchment area models and bypass flow model will be discussed afterwards.

Set drainage defaults

Draw the drainage strings

Set zero chainage to downstream end of the pipe string (Reverse drainage strings) (optional)

Automatically assign pit names

Define catchment areas (optional)

Define Pit/manhole Inlet Capacities via Overland flow strings (optional)

Hydraulic/hydrology calcs via spreadsheets, Drains, RAT2000, PCdrain, XP SWMM or ILSAX

Drainage longsection plots

Pit layout/construction schedules

Manhole and Pipe Table of Quantities (summarise by type, depth and size)

Drainage line excavation volume calculations

Roadway flooded width calculations

Drainage Export and Import to Design Software

Copy/Paste from spreadsheets

Running Drains

Running PCdrain (Windows)

Running Micro Drainage - Win DES

Running XPSWMM

Running RAT2000

22.10.0.1 Training Course Notes

Stormwater Part 1, Stormwater Part 2 and Dynamic Stormwater Design training course notes (Adobe PDF format for printing) and sample 12d data sets are available on the Installation DVD and the 12d User Forum. The training course moves through a worked example in detail. The course notes are also included below for your reference.

Stormwater Part 1

Stormwater Part 2

Dynamic Stormwater Design

22.10.1 Drainage FAQ

What is **pipe capacity**?

Pipe capacity is full pipe flow with no pressurisation. The capacity figure should be used to help designers know if the friction loss in the pipe is greater or less than the pipe slope.

Case 1: Flow less than capacity....friction slope is less than pipe slope. except for maybe for pit losses, surcharging is not a problem

Case 2: Flow greater than capacity....friction slope is MORE than pipe slope. You can do this for a short length of pipe but watch out for surcharging because you cannot keep it up for too long without flooding occurring!

Why do the **obvert/invert levels not change** when I import data via the "pit/pipe interface"?

On the interface panel there is an **Options** button that displays a panel with the **"hold obverts on import"** option. The default is off but if it is selected the pipe obverts will not change even if import data has new levels. Note that the invert levels will change is the pipe diameters change.

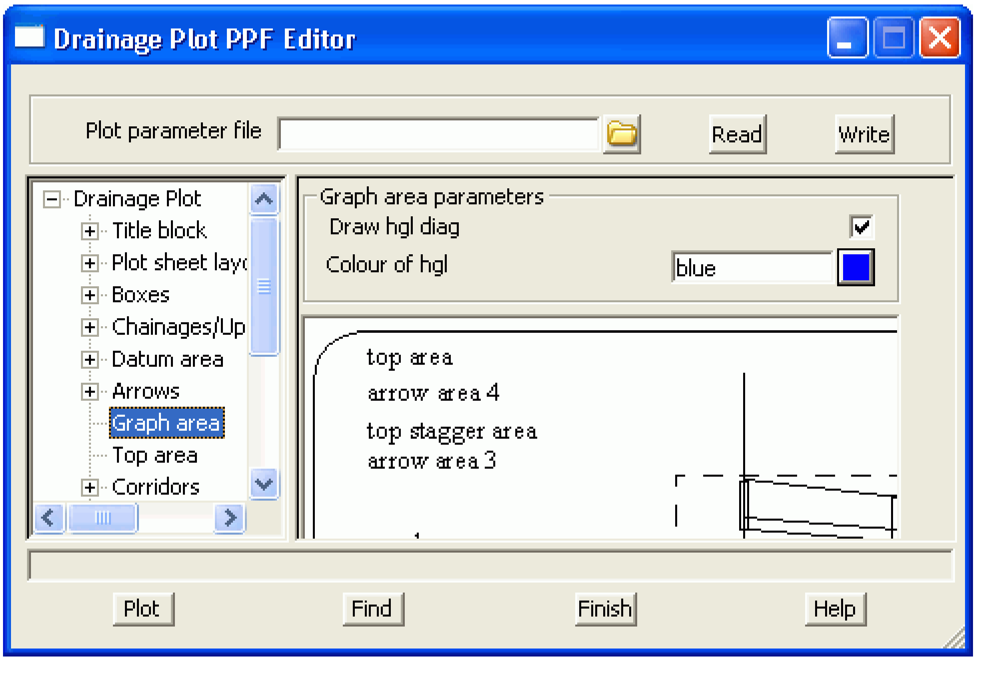
Can I **manually enter catchment areas** in the drainage design programs?

Yes. The catchment strings are optional in 12d. If no catchment string has been linked to a manholes and there has been no manhole "area" attribute a default area of 0.0001 will be sent. When results are imported back into 12d the total areas from the design package will be saved in the manhole "area" attribute.

How do I include/exclude the **hydraulic grade line** on drainage longsections..

First, there must be HGL results. These may come from the 12d rational drainage design, imported from another design package (including spreadsheets) or manually entered as manhole/pipe attributes. If HGL results exists then they may be viewed in the section view of the drainage line.

Second, the HGL results may be added to the drainage longsection drawings by editing the plot parameter file (ppf) as shown below.



22.10.2 Creating Drainage Strings

Drainage strings may be created in 12d in a number of ways.

1. Import 2d strings from other drawing packages and convert them to drainage string inside 12d.
2. draw the string in 12d using a design tin to the manhole cover levels and pipe inverts.
3. Enter the details in a spreadsheet format and paste the spreadsheet into 12d.

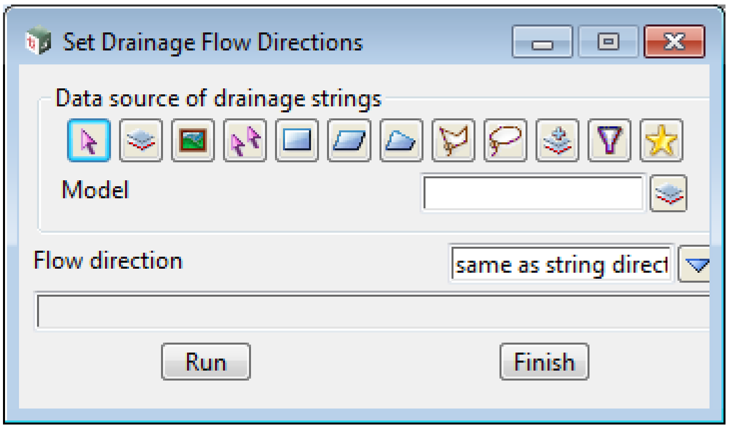
Regardless of the method, take the time to set your defaults before you start. You can always change the data later.

22.10.3 Set Drainage Flow Direction

Position of option on menu: **Design =>Drainage-Sewer =>More=>Set flow directions**

The routine changes the strings drainage flow direction to either **same as string direction** or **opposite to string direction.** This setting affects the pipe grading, network connectivity and hydraulic calculations.

On selecting the **Set flow directions** option, the **Set Drainage Flow Directions** panel is displayed.



The fields and buttons used in this panel have the following functions.

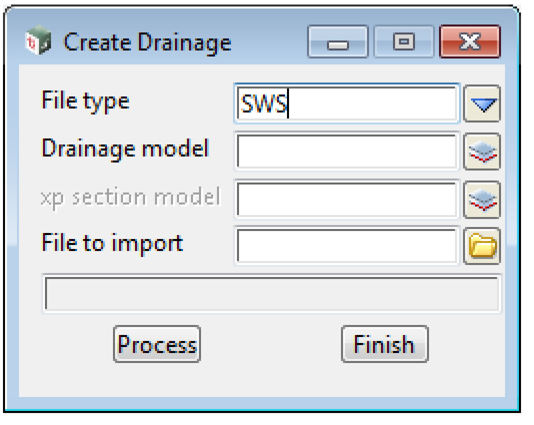
Field Description	Type	Defaults	Pop-Up
Source of drainage strings	source box		
Select the desired drainage strings			
Flow direction	choice box	same as string direction	same as string direction, opposite to string direction
all drainage strings will have this flow direction set			
Run	button		
set flow directions			
Finish	button		
remove the panel from the screen			

22.10.4 Create Network from Import

Position of option on menu: **Design =>Drainage-Sewer =>More=>Create network from import**

This routine creates drainage strings from existing MicroDrainage WinDes (sws and fws) and xpsoftware xpswmm (xpx) files.

On selecting the **Create network from import** option, the **Create Network from Import** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
File type	choice box	SWS	SWS,xpswmm

SWS (or FWS) for WinDes files, xpstorm files are the xpsoftware xpx files

Drainage model	model box		

the drainage strings created will be stored in this model.

File to import	file box		

select sws, fws or xpx file types

Process	button		

create import file and create strings

Finish	button		

remove the panel from the screen

22.10.5 Place Points

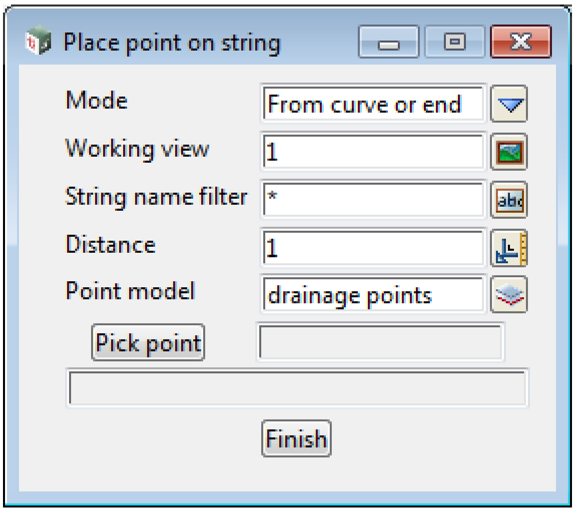
Position of option on menu: **Design =>Drainage-Sewer =>More=>Place points**

This routine creates points along a string so that drainage pits can be placed at a fixed offset from an intersection. As this is a very common task it combines several standard 12d options into one step.

The user selects a point near where the pit it to be located. The routine finds the closest string on the **working view** that matches the **String name filter** and creates a temporary point on the string.

The **Mode** determines a reference point located on the string and a marker of length, **distance**, is created from the reference point towards the temporary point.

On selecting the **Place points** option, the **Place point on string** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Mode	choice box	From curve or end	From curve or end, From end, Nearest point

if **From curve or end**, the reference point is closest (by chainage) end of curve or end of string
if **From end**, the reference point is the closest (by chainage) end of the string is located
if ***Nearest point***, the reference point is *the closest vertex to the temporary point is located*

Working view	view box	1	

*strings on this view will be searched using the **string name filter**. The **point model** will also be added to this view.*

String name filter	input	*	

*the closest string on the **working view** matching this filter will be selected*

Distance	input	1	

the chainage distance from the reference point to the end of the marker (note that the marker is drawn as a straight line but the distance uses the chainage value).

Point Model	model box	drainage points	

*the marker points are placed in this model using the **FLOW LINE** linestyle*

Pick point	button		

select a point near where the point marker is to be created. Only the x,y value is used from the selected point

Finish button

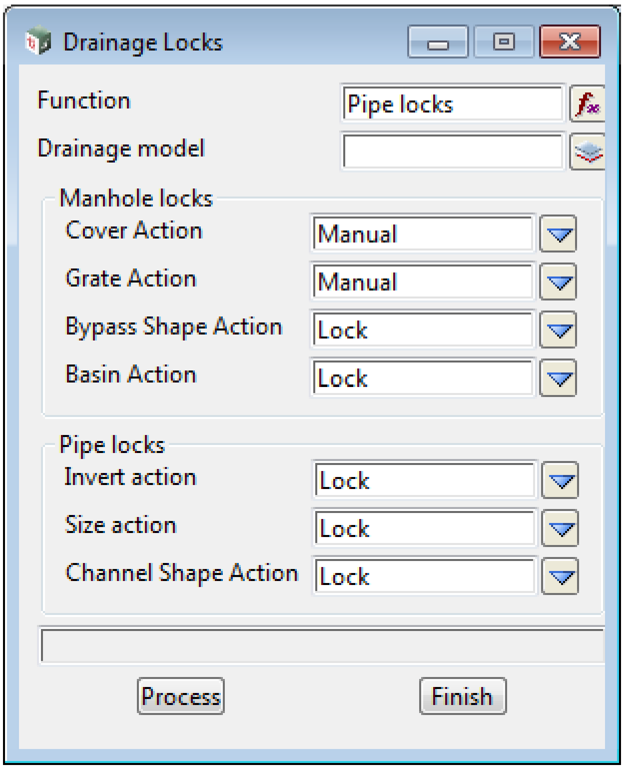
remove the panel from the screen

22.10.6 Drainage Pit and Pipe Locks

Position of option on menu: **Design =>Drainage-Sewer =>More=>Pipe locks**

The routine locks/unlocks (sets to manual) the pit and pipe levels in a drainage network model. A function is created so that the locks can easily be applied or removed. This is often used for existing networks or by designers wishing to set all levels manually.

On selecting the **Pipe locks** option, the **Pit and pipe locks** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function	function box	Pipe locks	

A function can be created to lock and unlock the drainage string properties (via attributes)

Invert action	choice box	Lock	Lock,Unlock,None,Default

Sets the pipe invert lock flags for the DNE
if Lock, the flag is enabled
if Unlock, the flags are cleared
if None, the flags are left unchanged
if Default, the flags are cleared (same as unlock)

Size action	choice box	Lock	Lock,Unlock,None,Default

Sets the pipe diameter, width and top width lock flags for the DNE
if Lock, the flag is enabled
if Unlock, the flags are cleared
if None, the flags are left unchanged
if Default, the flags are cleared (same as unlock)

Cover action	choice box	Manual	Manual,None,Default

Sets the mode for the cover levels (manual is the equivalent of locked)

if Manual, the mode is set to manual,
if None, the mode if left unchanged,
if Default, the field is cleared and the default setting becomes active

Grate action choice box Manual Manual,None,Default

Sets the mode for the grate levels (manual is the equivalent of locked)
if Manual, the mode is set to manual,
if None, the mode if left unchanged,
if Default, the field is cleared and the default setting becomes active

Drainage model model box

existing strings in this model are processed

Process button

Applies and removes the locks as indicated

Finish button

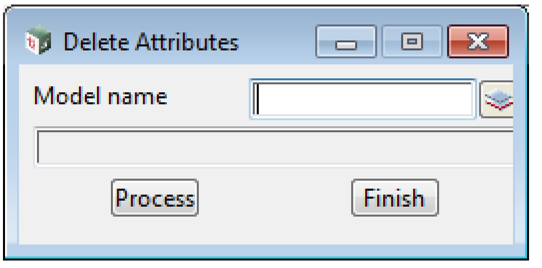
remove the panel from the screen

22.10.7 Delete All Attributes in a Model

Position of option on menu: **Design =>Drainage-Sewer =>More=> Delete All Attributes in a Model**

This routine deletes all of the model, string, pit and pipe attributes in the selected model.

On selecting the **Delete All Attributes in a Model** option, the **Delete Attributes** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model name	model box		

existing model

Process	button		

model, string, pit and pipe attributes will be deleted

Finish	button		

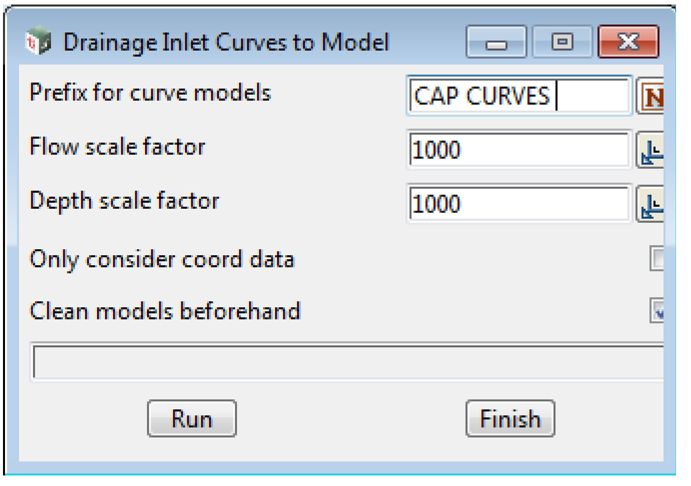
remove the panel from the screen

22.10.8 Write Inlet Curves to Model

Position of option on menu: **Design =>Drainage-Sewer =>More=>Write inlet curves to model**

This routines reads the inlet capacity curves from the drainage.4d file and creates plot strings. This tool is an efficient way to review the inlet capacity data in the file.

On selecting the **Write inlet curves to model**, the **Drainage Inlet Curves to Model** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Prefix for curves models	model box	CAP CURVES	

All inlet curves for a manhole type will be placed in a model named with this prefix and the manhole type.

Flow scale factor	input	1000	

Inlet and bypass flow values will be multiplied by this value before creating the strings.

Depth scale factor	input	1000	

Depth values for sag curves will be multiplied by this value before creating the strings.

Only consider coord data	tick box	off	

Inlet capacity curves may be expressed as formulas, curve coordinate or both. Selecting this box stops curves with only formulas from being plotted as strings.

Clean models before hand	tick box	on	

The curve models are cleaned before the new strings are created

Run	button		

Models and strings are created from the drainage.4d file

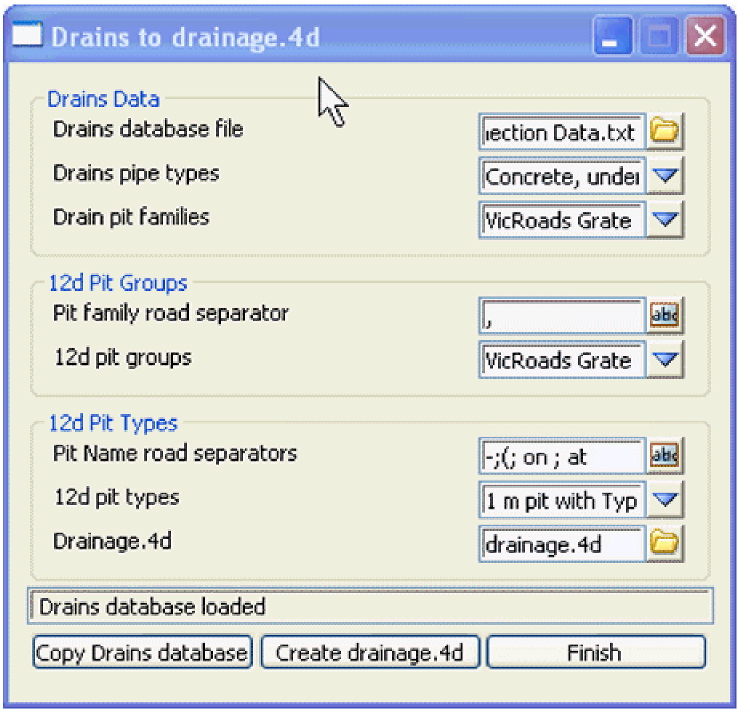
Finish	button		

remove the panel from the screen

22.10.9 Creating a drainage.4d file from the Drains database dump

Position of option on menu: **Design =>Drainage-Sewer =>More=>Drains to drainage.4d**

On selecting the **Drains to drainage.4d** option, the **Drains to drainage.4d** panel is displayed.



Key Points

When you are finished, open the drainage.4d file and check the road grade and crossfall values for the ongrade curves!

1. If Drains is used to selected the pit sizes
   then the 12d pit type must be the prefix of the Drains pit size (**Pit Name road separators** are used).

   If 12d is used to select the pit sizes to export to Drains, the Drains **Pit Size** and 12d **Pit types** have to match, exactly! The pit databases supplied by Drains often have road grade and/or crossfall attached to the end of the pit size. If pit sizes are to be sent from 12d, **this must be removed**! Find **Prepare the Drains Pit Database** for more details.
2. Select **Copy Drains database (12d will search the usual locations for the Drains data-base).**
3. Review the **Drains pipe types** and **Drains pit families** lists to ensure you have the correct database.
4. 12d pit groups are not used in Version 8.
5. **Pit Name road separators** can only be used if **Drains** is selecting the pit types. If 12d pit types are to be exported to Drains, this field should be left blank. If you change these entries you must press the enter key to update the **12d pit types** list below.
6. Check the **12d pit types** and if they are acceptable select **Create drainage.4d.**

The Details

YOU MUST RESTART 12D FOR THE NEW DRAINAGE.4D FILE TO BECOME ACTIVE!

Field Description Type Defaults Pop-Up

Drains database file file box Drains Connection Data.txt

You must update this file from Drains before each use of this panel. Inside Drains select Project=>Overflow Route database. Then select OK and then YES. This will cause Drains to export the database to the file "Drains Connection Data.txt".

Selecting ***Copy Drains database*** *will cause the panel to search for the database dump in the folders* ***C:\Program Files\Drains\Program*** *and* ***C:\Program Files\Drains\Demo\Program****. If the Drains program is installed in another folder then you must browse for the file. The file will be read and the panel updated with eother selection.*

Drainage.4d file box drainage.4d

The drainage.4d will be created in the 12d working folder unless otherwise specified. It will only be used for 12d projects in this folder.

Pit families choice box Drains pit families

These are the Drains pit families that will be exported to the drainage.4d file. These will become the names ongrade inlet capacity curves for all the pit sizes that belong to the family. The pit family name will be searched for words like ***grade****,* ***slope*** *etc to try to determine the values for road grade and crossfall for the 12d capacity curves.*

Pipe type choice box Drains pipe types

These are the Drains pipe types that will be exported to the drainage.4d file.

Pit group separatorinput box

Pit groups are not used in Version 8. These characters will be used to remove the road grade crossfall data from the ***Pit families*** *above. The data before this character will become the* ***12d pit groups.*** *Press* ***Enter*** *or select* ***Read Drains database*** *to create a new list of* ***12d pit groups.***

12d pit groups choice box 12d pit groups

These are created from the Pit family list above by deleting all text after the ***Pit group separator****.*

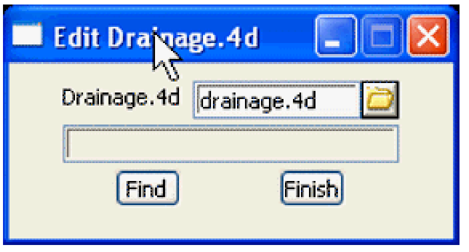
Create drainage.4dbutton

Create a drainage.4d file.

22.10.10 Editing a drainage.4d file

Position of option on menu: Design =>Drainage-Sewer =>More=>Edit drainage.4d

On selecting the Edit drainage.4d option, the Edit drainage.4d panel is displayed.



Select the **Find** button to search the 12d path for the current **drainage.4d** file. Select the **More info** button and then **Edit** to edit the file.

22.10.11 Drainage IO Defaults

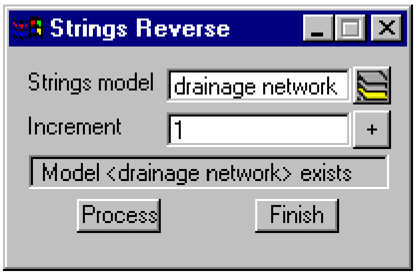
22.10.11.1 Set zero chainage to downstream end of line (Reversing the strings)

Position of options on menu: **Design =>Drainage-Sewer=>More=>Reverse all strings**

For drainage direction of flow see Reversing the strings

If desired, the reverse function may be used to move the zero chainage to the downstream end of the line. This should be done after entering all of the drainage strings and before naming the pits. From the main menu select

Design=>Drainage-sewer=>More=>Reverse all strings



This will also change the drainage flow direction attribute from **ascending chainage** to **descending chainage.**

Leave the **Increment** as 1.

Displaying the Auto Pit Names

22.10.12 Importing and Exporting

See Also

Drainage overview
Drainage Misc Utilities
Spreadsheet clipboard
Running Drains
Running PCdrain (Windows)
Running Micro Drainage - Win DES
Running XPSWMM
Running RAT2000

22.10.12.1 Spreadsheet clipboard

Spreadsheets are an effective method to manage the numerous variables urban drainage designers create in the modelling process. Spreadsheet data can be transferred to and from 12d in tab delimited files and stored within 12d as “user definable attributes”. These attributes are linked to the pit and pipes within a network. Drainage long section plots can display the pipe attributes in the “arrows” data area and pit attributes in the bubbles area. Drainage plan drawing can also show these pit and pipe attributes.

Drainage strings will be created if they do not exist in the model but pits cannot be added to existing strings.

See also

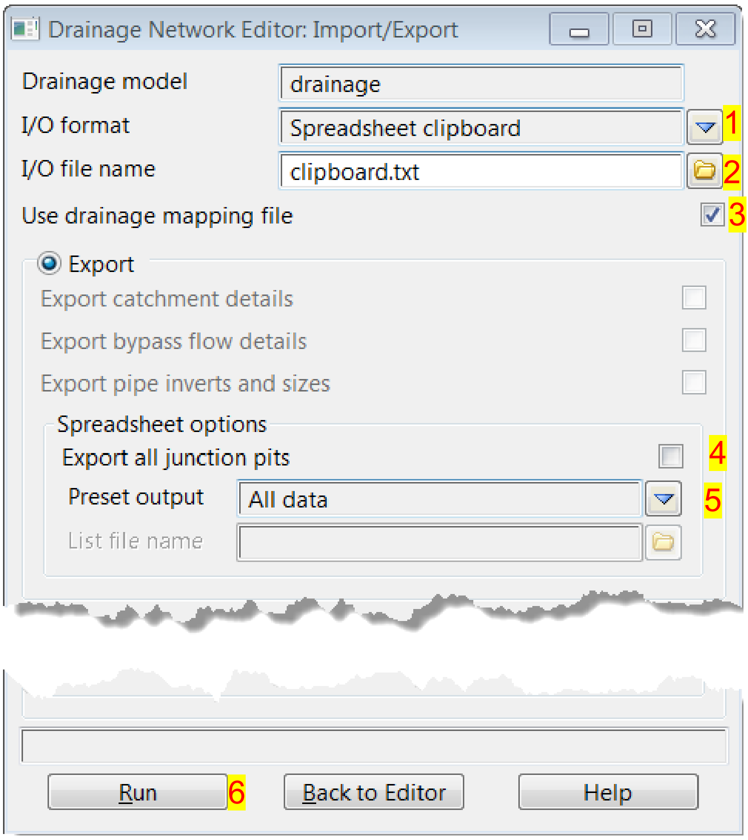
12d to spreadsheet transfers

Spreadsheet to 12d update and create

Spreadsheet options

22.10.12.2 12d to spreadsheet transfers

This interface is accessed the **Import/Export** button on the Drainage **Network Editor**.

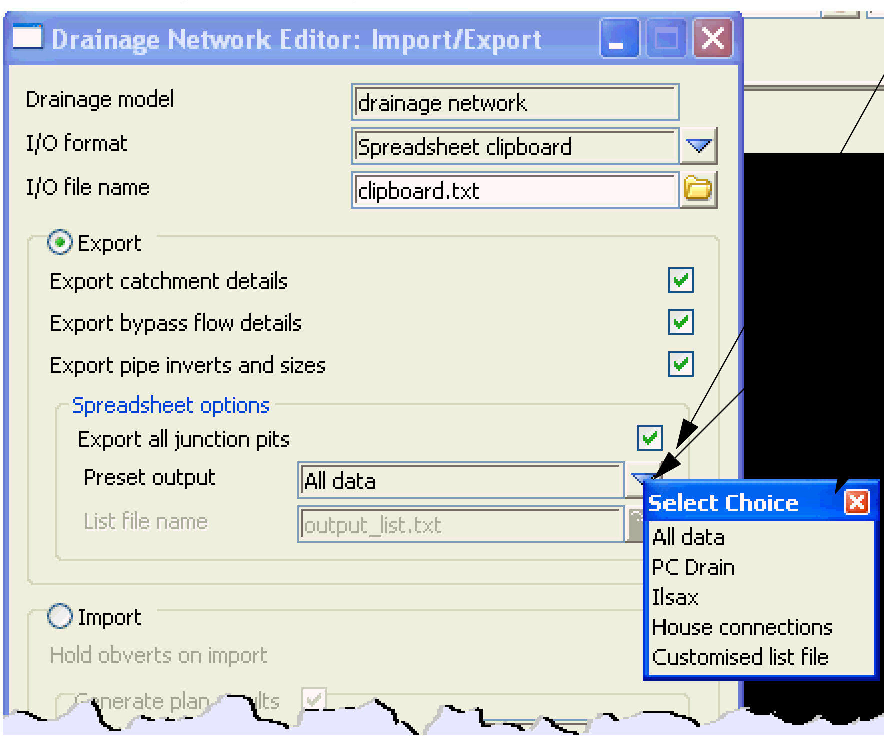


1. Select Spreadsheet clipboard
2. leave as **clipboard.txt** to send the data to the windows clipboard as well as this file.
3. Mapping files are the most current 12d technology. Leave this selected.

   These options are not used for spreadsheet export.
4. Usually leave this off! Select to export the junction pit at the end of all drainage lines (very rarely needed).
5. You may also select to limit the output if desired. If you like using spreadsheets for data entry, the PCdrain data and ILSAX data formats are useful for adding data for the first time for either program.
6. Select Run to place the data on the clipboard.

22.10.12.2.1 Options

The **Spreadsheet Options** section allows the user to define the amount of data exported.



Export all junction pits when selected will repeat the data for the junction pit at the end of each branch line.

Preset Output this choice box offers the following choices:

All Data: All of the 12d drainage string data and the user defined attributes will be exported to the clipboard in a tab delimited format. The 12d data names and the user defined attribute names will appear at the top of the spreadsheets columns.

ILSAX: For the ILSAX program, the spreadsheet column headings will change depending on the pipe and catchment indicators (P2 card) and the inlet type (P3 card). Therefore, use the ILSAX pipe editor macro to set up one pit/catchment for the type of data you wish to enter. Now when you export the pipe network data the column headings will include the names of the relevant parameters.

User defined below: The **Customised list file name** is used to define the drainage values, their order and format you desire.

The **customised list file** is a text file where each line contains a drainage variable or a spreadsheet IO command (blank lines are ignored unless preceded by the header command). The spreadsheet IO commands are all lower case and listed below:

header	to define a line of text to be exported
blank	to leave a blank column in the output
pit data	the following attributes are for the pit.
downstream pit data	the following attributes are for the downstream pit.
upstream pit data	the following attributes are for the upstream pit(s).
pipe data	the following attributes are for the pit's outlet pipe
downstream pipe data	the following attributes are for the downstream pipe(s)
upstream pipe data	the following attributes are for the upstream pipe(s)
variable name	a 12d drainage variable names
factor	the following variable is multiplied by this factor
decimals	the following variable will export with these decimal places

The simplest way to create your own customised tab delimited file is to set the **Preset Output** field to **All data** and leave the **customised list file name** field blank. Selecting **Set**, **Finish** and then **Copy** from the main dialogue. The data will be placed on the clipboard and a **customised list file**, named **output_list.txt** will be created containing the names of all of the drainage variables in the 12d model. Use a text editor to add/or delete the variable names, change their order and/or add spreadsheet IO commands. **Save the file with a new name!** The **output_list.txt** file is overwitten on every export.

A listing of a customised list file follows. Note the words in the header file have a “tab” between them so that they will be spaces across the spreadsheet columns.

```
header
Pipe Details
    header
    Name Length   U/S IL   D/S IL   Slope(%)   Class   Dia I.D.   Rough   Pipe Is   No. Pipes

    pit data
    *pit name

    pipe data
    *length
    low ch invert
    high ch invert

    factor
    100
    *grade

    pipe type

    factor
    1000
    diameter
```

After creating your customised list file, select **Options** again and change the **Preset Output** field to **User Defined below** and enter the new **customised list file** name that you saved above. Select **Set** then **Finish** and finally **Copy** to put the formatted data onto the clipboard.

The data can be pasted into a spreadsheet program for checking or additional formatting.

CUSTOM FORMATED DATA MIGHT NOT BE PASTED BACK INTO 12d!

The data must be in the “12d drainage spreadsheet” format to be read into 12d.

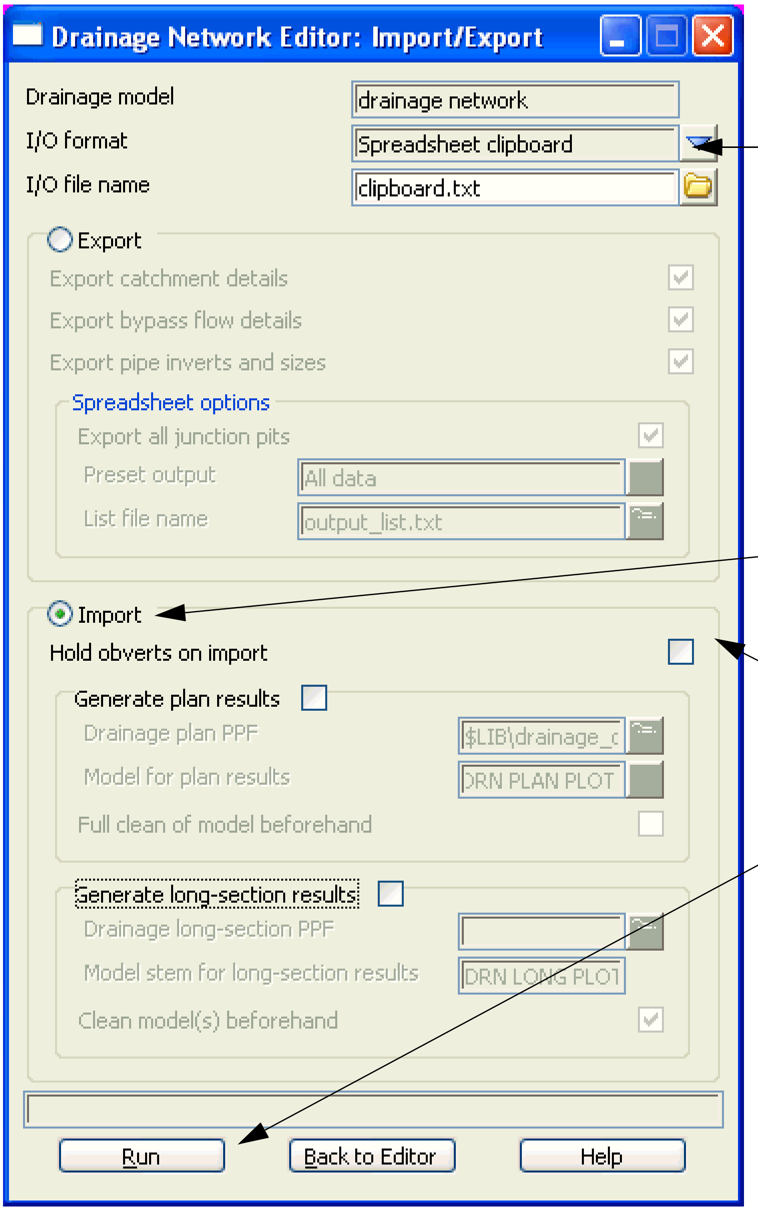
Caution with pit names in the form 1-1 or 1/1. Some spreadsheets will interpret these values as dates. If you use these formats for your pit names you will have to paste command them in once, format the columns that contain the pits names as text data and then paste the information in again.

One final word on using the copy/paste commands in the Microsoft Excel program. The Paste Special command using the “Skip Blanks” option will allow you to copy a large block of 12d data (with blanks in it) on top your data so that your data is preserved where it coincides with the blanks. To use this option paste the data into a blank spreadsheet and then select copy again. The Paste special option with “Skip Blanks” will now be available.

22.10.12.2.2 Spreadsheet to 12d Update and Create

This item is accessed from the **Import/Export** button on the **Drainage Network Editor**.

The following panel will appear.



Select the **Spreadsheet clipboard** format**.**

Select **Import**

This is ignored on spreadsheet imports.

Select **Run** to update the drainage model.

Tab delimited, “12d drainage spreadsheet” format or “from to” format data must be on the clipboard in order to update a 12d drainage model or create a new model. These format are described below.

22.10.12.2.3 Updating an Existing Model

The data usually is generated by 12d using the **Export** option, pasted into a spreadsheet and then copied back to the clipboard so that 12d can be updated.

When 12d exports the drainage model to a spreadsheet it includes a column for the unique string identifier and a unique pit identifier (unique to the drainage model not the 12d project). The names of the strings and pits may be changed via the spreadsheet if these columns are present at import time.

If the pit id column is not present, 12d will search the drainage model for a matching pit name. When the pit is a junction between drainage lines, only the trunk line will be the data updated.

22.10.12.2.4 Creating a New Model

It is possible to create a new string or an entire drainage network using this format. However, pits cannot be added to an existing string. The entire drainage string must be created at once. Two formats are available, the “from-to pit” format and the “12d drainage spreadsheet” format.

At present the network editor must select a drainage string to become active. Therefore, if you are not adding strings to a network, you will have to great a drainage network with one “dummy” pit. Select this one “dummy” pit to activate the editor. After importing the data and the new drainage lines are created the “dummy” pit may be deleted.

12d drainage spreadsheet Format

The top left cell in the clipboard data must be the text “12d” to specify this format. The minimum amount of data required to create a new string is the string name, pit name, x and y coordinates. You can add as much additional data as you have available. This would include pipe diameters inverts etc. The pits must be listed from upstream to downstream order. If the string is to join a trunk line, the junction pit must be included for both the tributary and the trunk line.

An example file exists called **new_network.txt** is supplied in the library. Open this file in a spreadsheet or a text editor and copy it to the clipboard. Set the **I/O Action** to **Import** and select **Run**. The new drainage lines will exist in the model currently being edited.

From-to Pit Format

The top left cell in the clipboard data must be the text “from to” to specify this format. The minimum amount of data required to create a new string is the upstream pit name "*pit name), the downstream pit name (*ds pit name) and the x(x location) and y(y location) coordinates of the upstream pit. If the string is to join a trunk line, the junction pit must be included for both the tributary and the trunk line.

An optional column for the pit cover elev (cover elev) may be specified. Once the network has been created additional pipe and pit data may be added using the “12d drainage spreadsheet” format described above.

An example file exists called **new_from_to_network.txt** is supplied in the library. It is shown below. Open this file in a spreadsheet or a text editor and copy it to the clipboard. Enter a new model name in the **Drainage model** field and select paste. The new drainage model will now exist.

from to	pit	pit	pit	pit
*pit name	*ds pit name	x location	y location	cover elev
text	text	real	real	real
E/1	A/3	5309.458	7336.935993	29.2173
D/1	A/4	5277.189	7336.935989	28.5071
C/1	B/3	5251.238738	7423.99485	31.5257
A/1	A/2	5354.629222	7336.935998	30.2115
A/2	A/3	5340.019987	7322.035996	29.89
A/3	A/4	5293.458002	7322.035991	28.8652
A/4	A/5	5250.182625	7322.035986	27.9127
A/5	A/6	5217.194202	7322.035983	27.1867
A/6	A/7	5183.458002	7322.035979	26.4442
A/7		5152.698693	7322.035975	25.7672
B/1	B/2	5289.42875	7422.289079	32.7197
B/2	B/3	5264.638564	7393.947083	30.7948
B/3	B/4	5249.738564	7384.207593	30.4187
B/4	B/5	5249.738564	7351.201545	29.1444
B/5	A/5	5233.426685	7336.935984	27.544

22.10.12.3 “12d drainage spreadsheet” Format

Each column of data is used for a 12d drainage variable or a user defined attribute. Each row represents a pit and the downstream pipe (controlled by the direction of flow variable) within the drainage network. A sample is shown below.

12d	pit	pit	pit	pit	pit	pit
*string Name	*pit name	pit type	pit low ch invert	pit high ch invert	pit id	string id
text	text	text	real	real	integer	integer
E	E/1	SA2	28.108	28.108	1	67389
E	A/3	SA2	27.7559	27.7559	2	67389
D	D/1	SA2	27.3961	27.3961	3	68100
D	A/4	SA2	26.8018	26.8018	4	68100
C	C/1	SA2	30.67	30.67	5	72072
C	B/3	SA2	29.563	29.563	6	72072
A	A/1	SA2	29.1026	29.1026	7	82469
A	A/2	SA2	28.7811	28.7311	8	82469
A	A/3	SA2	27.7652	27.7059	9	82469
A	A/4	SA2	26.8127	26.7518	10	82469
A	A/5	SA2	26.0867	26.0244	11	82469
A	A/6	SA2	25.3442	25.2942	12	82469
A	A/7	SA2	24.6672	24.6672	13	82469
B	B/1	SA2	31.2759	31.2759	14	192066
B	B/2	SA2	29.351	29.301	15	192066
B	B/3	SA2	29.123	29.073	16	192066
B	B/4	SA2	28.0444	27.8951	17	192066
B	B/5	SA2	26.3447	26.2947	18	192066
B	A/5	SA2	26.0744	26.0744	19	192066

Duplicate Definitions

Strings Variables such as “direction” are may be defined for numerous pits on the same string. Searching in a top down direction through the file, the last definition found for the string will be set.

Invert levels may be set via pipe data or pit data or combined. It is recommended that the user only use one method and not combine them. Both are exported so delete the ones you are not going to use. The variables are processed from left to right, so if duplicate definitions of an invert level or found the right most data will be set.

The format definition

1. Row1, column 1 must contain either “12d”, or “from to”. Therefore, the first column must be a 12d drainage variable (cannot be a user defined attribute).
2. Row 1. The text <pit> at the top of the column indicates the column contains a user defined pit attribute and similarly <pipe> indicates a user defined pipe attribute.
3. Row 2. This row contains the names of the 12d drainage variable names and the pit/pipe attributes. All names are case sensitive so be careful where you use capital letters. A list of 12d drainage variables is found below.

   Names beginning with an asterix (*) will not be processed (except pit/string names when unique identifiers are present in the data). 12d drainage variables names beginning with an asterix indicate that this data was calculated at export time and cannot be read back into 12d (for example, pipe length, pipe grade and deflection angle).

   Prefixing an user defined attribute name with “DELETE ” (no quotes, note the space after the DELETE) will cause the attribute to be deleted from all pits/pipes within the model.

4.Row 3. The text in this row define the type of attribute to be stored within 12d. The only valid choices are;

integer
real
text

If you want to change an attribute type you must delete the attribute and create it again. If you simply change the attribute type in the third row then that attribute will not be updated.

5.Blank lines may be inserted as desired.

6.You are not required to fill in all of the cells in the spreadsheets. Blank cells are ignored (you must use a space to remove all data from text attributes (the space will not be stored).

7.Pipe names are included in the data so that they can be changed but they are **not** used to identify the pipe. Pipe data will always be assigned to the pipe following the pit in the direction of **ascending** chainage. If flow directions is ascending then the pipe data will be for the downstream pipe. If the flow direction is descending then the pipe data will apply to the upstream pipe.

22.10.12.4 12d Drainage Variable Names

Manhole Variables	Pipe Variables	String Variables
*string Name	pipe name	direction
*pit name	pipe type	[illegible]
pit type	low ch invert	[illegible]
pit diameter	high ch invert	string id
pit low ch invert	diameter	
pit high ch invert	*length	
pit road chainage	*grade	
pit road name	low hgl	
*pit angle	high hgl	
*pit drop	pit hgl	
*pit depth	flow	
*pit chainage	velocity	
x location		
y location		
cover elev		
*fs elev		
*ns elev		
pit id		

22.10.13 Quick Check Lists for Drainage Design Software

The drainage design software packages each have their specific requirements regarding the manhole types, names and then method they use to model pit inlet capacities. The following section details the specific requirement for each package.

See Drains Version 2+ Requirements and Notes

See PCdrain Requirements

See Running RATHGL/RAT2000 and XP-SWMM

See Micro Drainage - WinDes Requirements

22.10.13.1 Drains Version 2+ Requirements and Notes

See also Drainage overview

Basic Check List

1. All manholes must have a unique name (Manhole names cannot begin with "O ")
2. The pipe type used in 12d must exist in the Drains pipe database

Overland flow with Inlet Capacity Check List

1. Make sure you are using Drains Version 2-5 clipboard data. Inside Drains select **Project=>Pit database**. The pit family selection should be available. (CAP1 etc. is Version 1).
2. All Overflow route shapes, pit families and pit types used in 12d must exist in the Drains pit database
3. Overland flow lines must be within 1.0 metres/feet of the manhole

22.10.14 Running Drains - Version 2+

Key Points

The Drains database (each project has its own) and the12d database (drainage.4d) must be synchronised (**More**).

12d Version 8 needs you to create a new drainage.4d file to include the new inlet capacity curves (pit groups are no longer used).

For pits with a bypass pit and road grade defined (non ILLUDAS), 12d selects pit families for Drains in the following way

The users selects the pit type (Drains pit size) in 12d and 12d selects the ongrade or sag curve from the drainage.4d file (see inlet capacity calculations). The names of these curves are the Drains pit families. If there are no curves for the pit type in the drainage.4d file then the Drains pit type is exported as ILLUDAS with no family or size data.

If the 12d pit type is **Headwall** then the pit will be exported as a headwall.

Catchment lengths, slopes and roughness values are NOT exported in the Drains Rational format. The default values for these parameters are NOT exported in the Drains ILSAX format but the explicit settings from the catchment panel are exported. If used, Drains requires length, slope and roughness for both impervious (paved) and pervious (grass). Supplementary values must entered in Drains.

Data is copied from 12d to the Windows clipboard and then pasted into Drains. 12d can not delete any objects in Drains, it can only add and update.

When updating 12d from Drains, always copy the DATA to 12d before the results. 12d will update the network but will not add or delete pits.

If a new pit is added in Drains, the user willhave the option to create it in 12d. If it is created only the pit will be added and not the pipes.

Why Do I get ILLUDAS pits?

The bypass pit is blank. There must bypass pit for the water to go to. Bypass flow strings must pass within 1 pit diameter (beware pits with zero manhole diameters) and you can enter them manually.

The pit type selected has no capacity curves in the drainage.4d file. The curve names become the pit family for the export to Drains. No pit family means ILLUDAS.

If you are intentionally using a drainage.4d file with NO inlet capacity curves, the pit families you set in Drains will be remembered by 12d when you import back into 12d. The inlet capacity curves will not be required for your next export to Drains.

Drains Interface Overview

The Drains program performs the rational or ILSAX hydrology calculations as well as hydraulic grade line calculations that determine pipe sizes and pipe invert levels.

The data sent to Drains includes

pit names and types, easting and northing data with surface levels

finished surface profile along the centre line of the pipes

Headwalls and their levels

optional - bypass inlets, road grades/crossfalls and **SAG Inlet Calculations** (ponding volumes and depths). Pit family selection using road grade and crossfall data.

optional - composite catchment area create from three12d areas per inlet (must alter the mapping file)

catchment characteristics, k values and overland travel times

pipe sizes, type and invert levels

Data is copied from 12d to the Windows clipboard and then pasted into Drains **(Edit=>Paste data from spreadsheet)**. 12d can not delete any objects in Drains, it can only add and update.

The Drains menu selection **Run=>Design** is used to design the network. Once the drainage network has been designed in Drains the updated design data (**Edit=>Copy data to spreadsheet)** and/or the hydraulic results (**Edit=>Copy results to spreadsheet)** are sent back to 12d via the clipboard.

Always copy the DATA to 12d before the results as the results are deleted inside 12d with every update of the data!

Catchment Data

Drains has one catchment per inlet and therefore only 12d's catchment set one is used in the interface. The 12d catchment is split a pervious (grass) area and an impervious (paved) area using the percent impervious fraction. The rational and ilsax spreadsheet export formats (selected in the import/export panel) are different.

The Rational format of the Drains interface does not export Drains "more detailed data". The ilsax format does support the "more detailed data" format but the user must define the length, slope and roughness in the catchment panel. No 12d DNE default data will be exported. As 12d does not have a supplementary area this data will have to be entered into Drains. Gutter lengths and slopes are calculated from the upstream section of the longest overland flow path entering the inlet.

Supplementary data entered in Drains will be remembered in 12d for the next export. The grassed percentage for the next export is calculated as the (100 - % impervious - % supplementary).

Synchronising the Drains database and the drainage.4d file.

The 12d pipe types are always used to interface with Drains. Ongrade or SAG pits will be created for export to Drains if the 12d pit has a bypass pit, road grade and a pit type with inlet capacity curves. If all three are not present it will be exported as an ILLUDAS pit.

Every Drains file begins with a default database and Drains uses that database for the life of the project. This database must by synchronised with the **drainage.4d** file in 12d to ensure the Drains pit families, pit sizes, pipe types and overflow route types match. Therefore it is highly recommended that copy of the drainage.4d file be kept in the 12d working folder.

The Drains and 12d data must match as follows.

Drains database	**drainage.4d**
Pipe type	Pipe type entries
Only if bypass flow is required	
Drains database	**drainage.4d**
Pit size	Pit type
Pit famly	ongrade or sag curve names
Overland route database	default in mapping file

Note: If Drains is to be used to select the pit size then an exact match in pit types is not required. For this case the 12d pit type need only be the prefix of the Drains pit size.

The following 6 steps will help ensure 12d is synchronised with Drains. More details are given in the sections below.

1)Prepare the Drains Pit Database

IF 12d is to select the Drains **Pit Size**, then the Drains **Pit Size** and 12d **Pit types** have to match, exactly! If Drains is used to determine the pit size then you may skip this step.

The pit databases supplied by Drains often have road grade and/or crossfall attached to the end of the pit size. **THIS EXTRA DATA MUST BE REMOVED**!

For example, in Drains database you may find the following

Pit family	Pit size
NSW RTA SA Inlet, 3% crossfall, 1% grade	SA1 (Type 2) - 1% longitudinal grade
and	
NSW RTA SA Inlet, 3% crossfall, 3% grade	SA1 (Type 2) - 3% longitudinal grade

The pit size for both is the same, **SA1,** but 12d can not tell this because of all the extra data at the end. In Drains, simply remove the extra information so that they are both **SA1**. If the network already exists in Drains, then Drains will automatically change the names of the sizes you have already selected. No time lost here.

Once you have changed the Drains pit size names, you are ready to use the **Drains to drainage.4d** routine to create a drainage.4d file that is synchronised with Drains.

2)Export the Drains database from your Drains file

Inside Drains Open your drains file or begin with a blank file. Select **Project=>Overflow Route database**. Then select **OK** and then **YES**. This will cause Drains to export the database to the file "Drains Connection Data.txt" and store it in the folder with the Drains program.

3)Create a drainage.4d file from the Drains file

From the 12d menu select **Design->Drainage-Sewer->More->Drains to drainage.4d**.

Select **Copy Drains database** to copy the file to the 12d working folder.

select **Create drainage.4d.** The Drains database dump file is copied to the 12d working folder and a drainage.4d file is created in the 12d working folder.

More details below.

4)Edit the drainage.4d file

From the 12d menu select **Design->Drainage-Sewer->More->Edit drainage.4d**. Select **Find** then edit from the file **more info** button. Set the road grade and crossfalls for the 12d pit groups.**More details below.**

5)Restart 12d

From the 12d menu select **Project->Restart**.

6)Set the Overflow shape, Update Pit and Pipe types (Optional)

If the network has already been created using pit and pipe types that no longer exist in the drainage.4d file, they will have to be updated before the export to Drains can occur. You may update them using the drainage network editor (one at a time) or you may set **all** of the pit and pipes types to **one value** using this routine.Later you may change them individually using the **Drainage network editor**

This routine will also set a default overflow route shape for export (values can be modified in the Drains program if desired).

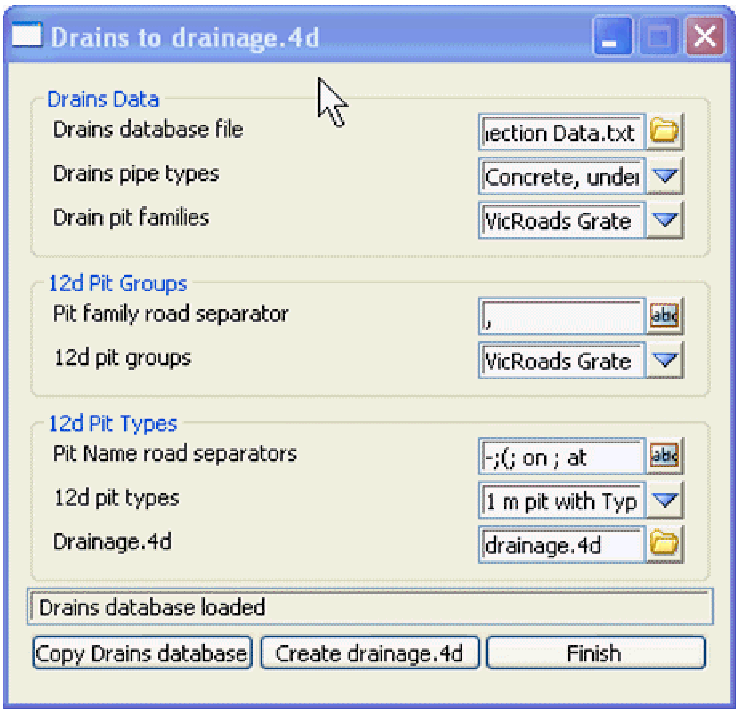
From the 12d menu select **Design->Drainage-Sewer->More->Drainage io defaults.**

More details below.

Drains to drainage.4d file

Position of option on menu: **Design =>Drainage-Sewer =>More=>Drains to drainage.4d**

On selecting the **Drains to drainage.4d** option, the **Drains to drainage.4d** panel is displayed.



Key Points

When you are finished, open the drainage.4d file and check the road grade and crossfall values for the ongrade curves!

If Drains is used to selected the pit sizes
then the 12d pit type must be the prefix of the Drains pit size (**Pit Name road separators** are used).

If 12d is used to select the pit sizes to export to Drains, the Drains **Pit Size** and 12d **Pit types** have to match, exactly! The pit databases supplied by Drains often have road grade and/or crossfall attached to the end of the pit size. If pit sizes are to be sent from 12d, **this must be removed**! Find **Prepare the Drains Pit Database** for more details.

Select **Copy Drains database (12d will search the usual locations for the Drains database).**

Review the **Drains pipe types** and **Drains pit families** lists to ensure you have the correct database.

12d pit groups are not used in Version 8.

Pit Name road separators can only be used if **Drains** is selecting the pit types. If 12d pit types are to be exported to Drains, this field should be left blank. If you change these entries you must press the enter key to update the **12d pit types** list below.

Check the **12d pit types** and if they are acceptable select **Create drainage.4d.**

The Details

YOU MUST RESTART 12D FOR THE NEW DRAINAGE.4D FILE TO BECOME ACTIVE!

Field Description	Type	Defaults	Pop-Up
Drains database file	file box	Drains Connection Data.txt	

You must update this file from Drains before each use of this panel. Inside Drains select Project=>Overflow Route database. Then select OK and then YES. This will cause Drains to export the database to the file “Drains Connection Data.txt”.

Selecting **Copy Drains database** will cause the panel to search for the database dump in the folders **C:\Program Files\Drains\Program** and **C:\Program Files\Drains\Demo\Program**. If the Drains program is installed in another folder then you must browse for the file. The file will be read and the panel updated with eother selection.

Drainage.4d file box drainage.4d

The drainage.4d will be created in the 12d working folder unless otherwise specified. It will only be used for 12d projects in this folder.

Pit families choice box Drains pit families

These are the Drains pit families that will be exported to the drainage.4d file. These will become the names ongrade inlet capacity curves for all the pit sizes that belong to the family. The pit family name will be searched for words like **grade**, **slope** etc to try to determine the values for road grade and crossfall for the 12d capacity curves.

Pipe type choice box Drains pipe types

These are the Drains pipe types that will be exported to the drainage.4d file.

Pit group separatorinput box

Pit groups are not used in Version 8. These characters will be used to remove the road grade crossfall data from the **Pit families** above. The data before this character will become the **12d pit groups.** Press **Enter** or select **Read Drains database** to create a new list of **12d pit groups.**

12d pit groups choice box 12d pit groups

These are created from the Pit family list above by deleting all text after the **Pit group separator**.

Create drainage.4dbutton

Create a drainage.4d file.

Editing the Drainage.4d file

Position of option on menu: Design =>Drainage-Sewer =>More=>Edit drainage.4d

On selecting the Edit drainage.4d option, the Edit drainage.4d panel is displayed.

Select the **Find** button to search the 12d path for the current **drainage.4d** file. Select the **More info** button and then **Edit** to edit the file.

The drainage.4d file contains Manhole and Pipe commands. 12d also uses the Manhole commands to specify a pit group by using the prefix “group”. Details follow.

Headwalls

The pit type “Headwall” (case sensitive for Drains) is reserved for the inlet headwall for a conduit. Drains does not use this for an outlet headwall.

If the inlet type is On Grade or there is no bypass pit, the grate level is exported as the Drains surface level. If the Inlet type is marked as a SAG pit then the surface level will be calculated from the low point on the catchment string (catchment set #1). Also see **SAG Inlet Calculations**.

Pit Families

Version 2+ of the Drains clipboard interface uses a **pit family** to describe the kerb shape and optionally, the name include the road crossfall and/or grade attached as a suffix.

An example pit group is the drainage.4d file is shown below.12d uses the road grade and/or the road crossfall to select which pit family should be sent to Drains. In this example the road crossfall would not be used in selecting the pit family.

It is up to the user to decide the grade when the next pit family should be used. In this case the threshold value for the gutter grade is set midway between the published values of the inlet curves. For example at a gutter grade of 2% 12d starts sending the ***NSW RTA Pits - 3% slope*** pit family.

The pit families listed on the right must match exactly with those in the Drains pit database.

Pit Types and Pit Sizes

Each Drains **pit family** has several **pit sizes**. The Drains **pit sizes** link to the12d pit types and therefore all Drains **pit sizes** should exist in the 12d drainage.4d file.

The **pit size** will be read back from Drains into 12d as the **pit type** so that it can be placed on the drainage long sections and pit schedules. If no matching pit type is found in 12d then a character will be dropped off the end of the Drains pit size and 12d will be searched for the new shorter pit type. This will be repeated till a match is found (or there are no more characters in the Drains pit size).

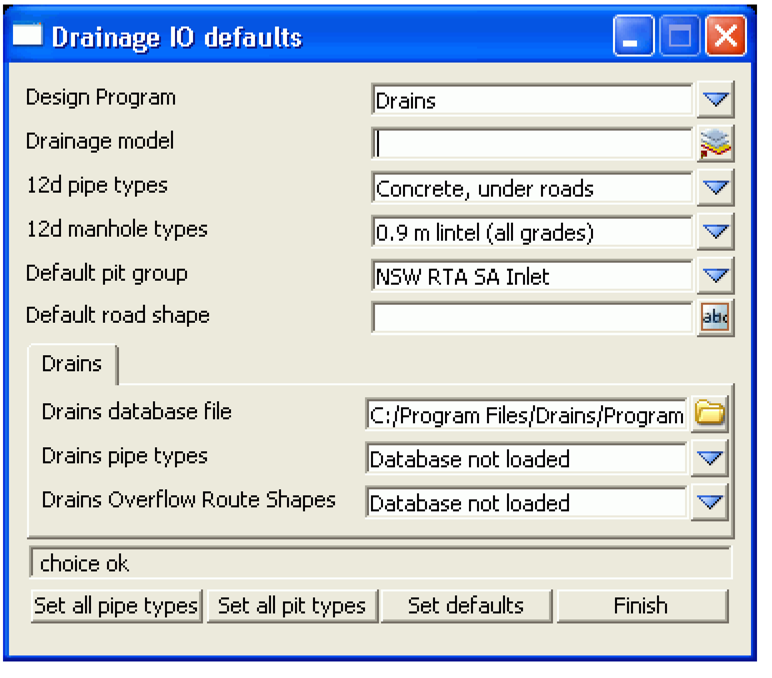
Pipe Types

The pipe type selected in 12d must exist in the pipe database inside Drains. Simple “2” for class 2 or “RCP “do not exist in Drains.

Setting the Overflow Route, the Pit and Pipe types

Position of option on menu: Design =>Drainage-Sewer =>More=>Drainage IO Defaults

On selecting the Drainage IO Defaults option, the Drainage IO Defaults panel is displayed.



The routine changes ALL of your pit and pipe types in a model to a single value. If you have changed your drainage.4d file after creating you drainage network, the pit and pipe types you originally selected may no longer be valid (i.e. in the drainage.4d file).

Field DescriptionType Defaults Pop-Up

Design Programchoice Drains Drains, PCdrain

The drainage model to be updated.

Drainage modelfile

The drainage model to be updated.

12d pipe types choice values from drainage.4d

Set all pipe types will set all pipes in the model to this value

12d pit types choice values from drainage.4d

Set all pit types will set all pits in the model to this value

Default pit groupschoice values from drainage.4d

pit definitions in the drainage.4d file that have **group** as a prefix are included.

Default road shapechoice values from drainage.4d

type the desired name or if using Drains select the desired shape from the **Drains Overflow route shapes**.

Drains Tab

Drains database filefile

pressing enter in this field will start a search for the Drains database dump. The search path is the specified folder, C:\Program Files\Drains\Program then C:\Program Files\Drains\Demo\Program. If the file is found the choice fields below are populated. **It is highly recommended that this file be in the 12d working folder.**

Drains pipe typeschoice l values from Drains file

the pipe types are retrieved from the last Drains database dump. Changing this value will update the **12 pipe types** above.

Drains Overflow Route Shapeschoice values from Drains file

the overflow route shapes are retrieved from the last Drains database dump. Changing this value will update the **Default road shape** above.

Set all pipe typesbutton

all pipe types in model are set to this value

Set all pit types button

all pit types in model are set to this value

Set defaults button

the defaults for the **Drains Overflow Route Shapes** and **12d pit group** are set

Finish button

removes the panel

Drains Version 2+ Requirements

Pit Names

The 12d pit names cannot be more than 9 characters long. 12d uses 2 additional characters to the pit name at export time create names for the pipes, overflow routes and catchments. For example pit “A-1” will have a bypass route “F A-1”, a catchment “C A-1” and a downstream pipe “P A-1”.

Bypass Flow (Overland Flow Routes)

There are 3 requirements for Drains bypass flow (bypass pit, pit inlet capacity curve and road grade/crossfall).

Select a **pit type** that has ongrade or sag inlet capacity curves defined in the drainage.4d file. The Drains and 12d databases must be in sync.

Bypass strings in the **Bypass route model** specified (Network editor->Global->Utility Models->Bypass flow model. For more details see **Bypass Flow.**

Road grade and crossfall calculated (Network editor->Global->Utility Models->Road design file)

The overland flow strings are not allowed to pass through the outlet pit on the network.

SAG Inlet Calculations

SAG inlets are inlets where the water ponds at the surface rather than flowing past. If a SAG inlet has a catchment string the overflow depth and volume are calculated. The catchment string from Set #1 is draped onto the design tin and the lowest point in found on the draped string (stored as a 12d pit attribute ***overflow level***. The storage volume inside the string up to this point is measured and stored as a 12d pit attribute ***overflow volume from level***. and are subtracted from the to determine

The Max Ponding Depth = **overflow level**- **grate level**

If the **manual flag** is selected for the "sag pit pond depth", 12d will NOT calculate the value but will use the value entered by the user. 12d will check if a volume has been calculated before or imported from Drains. If the volume exists then it is exported. If it does not exist, this ponding depth will be used calculate the ponding level (depth + grate level). The ponding level will be used to calculate a ponding volume from the design tin and the catchment boundary.

If you want to manually delete this volume and force 12d to recalc to volume using the user defined level, delete the pit attribute, "overflow volume from level". To delete the attribute use Strings->Properties->Attributes, select the pit, go to the pit tab, right mouse select the row with this attribute and select delete).

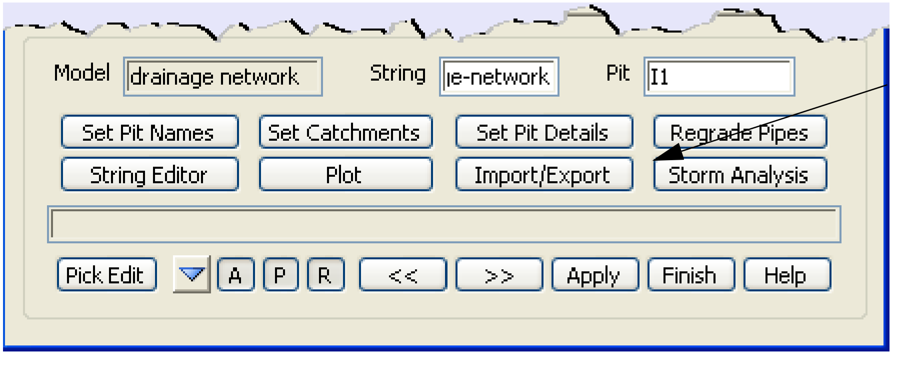
Results

Drains exports the maximum data from all of the rainfall events analysed. Therefore, ensure you analyse only the rainfall events desired before coping the results to the clipboard. To verify the data that is being sent to 12d, copy the data into a spreadsheet so you can view it there first. The pit sizes selected in Drains will be stored in 12d as the pit type. Therefore the pit sizes in Drains should exist as pit types in the drainage.4d file. If pit families are changed in Drains the pit group in 12d will be updated by search for the pit family in the drainage.4d file.

12d to Drains

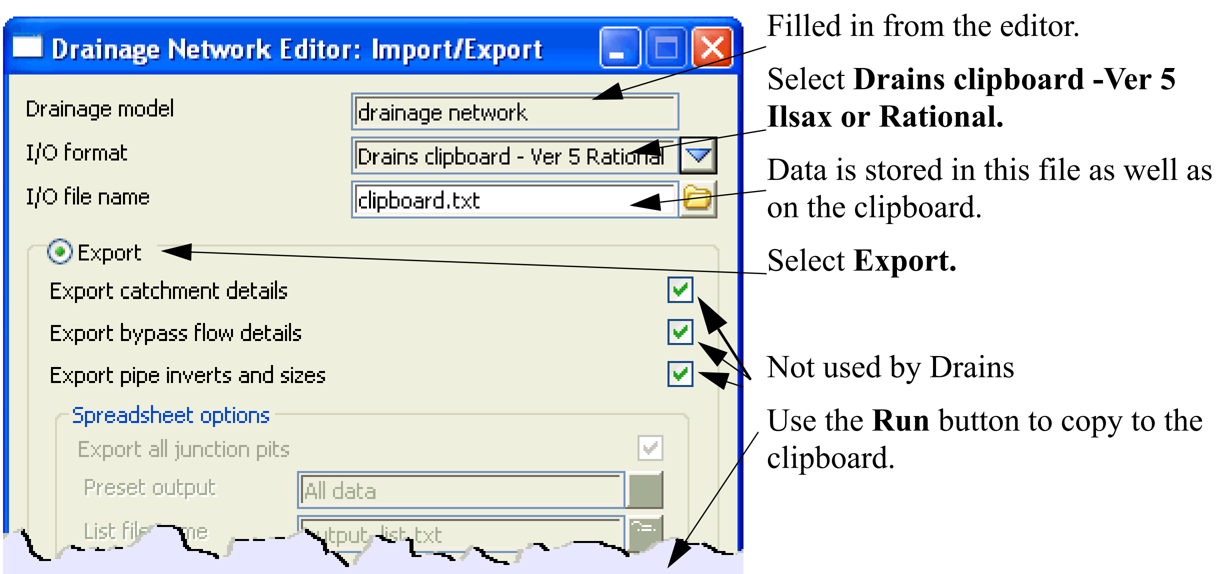
Setup your drainage network models and ensure they have been assigned pit names.

Copy the data to the clipboard



From the Drainage network editor select the **Import/Export** button.

The following interfaces dialogues will appear.



From within the Drains program select **Edit =>Paste data from spreadsheet**. If you paste the data into a Drains project that has a hydrological model and rainfall data already defined the project will be ready to run.

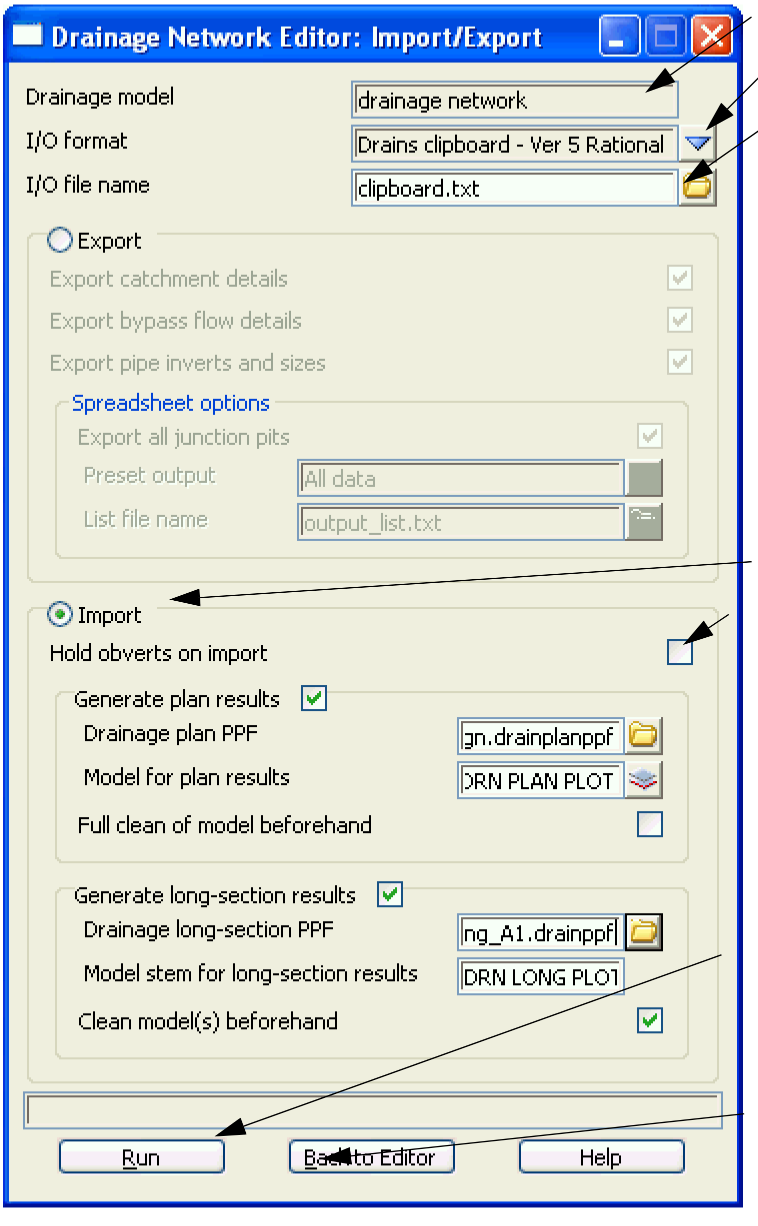
Use the Drains Run=>Standard design to design your pipe sizes and invert levels. The Run=>Advanced Design will select the size of the pits as well.

Drains to 12d Update

The following steps are required to update the 12d model with the Drains hydraulic results and changes to the pipe sizes and inverts.

To update the pipes and invert levels in 12d, select **Edit->Copy Data to Spreadsheet** from the Drains menu.

From within the 12d Drainage network editor select **Import/Export**.



Filled in by the editor.

Select Drains clipboard - Version 5

Leave as clipboaard.txt

Select **Import**

This will ignore the invert levels read from Drains and the current pipe obverts will remain fixed.

Plan and long section drawings may be created at the import time so that you can see the results on the drawings.

Select **Run** to update the drainage model. To see the changes in the section views you will have to select **Regen** on the section view toolbar.

To return to the network editor select **Edit**

IMPORTANT: THE DATA MUST BE PASTED BEFORE THE RESULTS!

12d erases the hydraulic and hydrology data when the physical data is updated. Therefore, always paste the data before the results.

Drains Mapping File

The rational format and the ilsax format have separate mapping files.

The Drains spreadsheet format consists of sections starting with a header, then the data area and ending with a blank line. The mapping file defines the format of the section headers and the format of the data within the sections

	A	B	C	D	E	F	G	H	I
1	PIT / NODE DETAILS			Version 9					
2	Name	Type	Family	Size	Ponding	Pressure	Surface	Max Pond	Base
3					Volume	Change	Elev (m)	Depth (m)	Inflow
4					(cu.m)	Coeff. Ku			(cu.m/s
5	A2	Node	Sutherlan	0.85 m lint	1	1	26.451	0	
6	A3	OnGrade	Sutherlan	0.85 m lint	1	1.2	27.194	0	
7	A4	OnGrade	Sutherlan	0.85 m lint	1	1.2	27.92	0	
8	A5	OnGrade	Sutherlan	0.85 m lint	1	1.2	28.872	0	
9	D1	Sag	NSW RTA	SA1	1	2	28.492	0.068	
10									
11	DETENTION BASIN DETAILS								
12	Name	Elev	Surf. Area	Init Vol. (c	Outlet Typ	K	Dia(mm)	Centre RL	Pit Famil
13									
14	SUB-CATCHMENT DETAILS								
15	Name	Pit or	Total	Paved	Grass	Supp	Paved	Grass	Supp
16		Node	Area	Area	Area	Area	Time	Time	Time
17			(ha)	%	%	%	(min)	(min)	(min)
18	C A2	A2	0.039	80	20	0	5	10	0
19	C A3	A3	0.0408	80	20	0	5	10	
20	C A4	A4	0.0705	80	20	0	5	10	

The headers in the mapping file define the exact text to be exported in the Drains section headers. During an import the first 4 columns of the Drains header are compared to the headers in the mapping file to determine which section is being read. The data format line is then used to decode the the data.

22.10.15 PCdrain Requirements

See also Drainage overview

Basic Check List

1. All manholes must have a unique name
2. Only one outlet allowed in the drainage model
3. First export should have **“Export default catchment/pit/overland parameters”** selected
4. Pit types in used in 12d must exist in the PCdrain Inlet gully file selected.

Overland flow with Inlet Capacity Check List

1. A gutter profile named “4d” must exist in PCdrain before the interchange file is read. This gutter profile may be created in PCdrain from the menu selection **Data=>Gutter profiles** then **New.**
2. Overland flow lines must be within 1.0 metres/feet of the manhole

Data is exchanged to and from PCdrain via the interchange (*.int) file. Gutter profiles and inlet type must be specified in PCdrain before the interchange file is read into PCdrain.

The data sent to PCdrain includes

- pit names and types, easting and northing data with surface levels
- pipe deflection angles at pits
- finished surface profile along the centre line of the pipes
- optional - crossing services - level, size and location along the pipes
- optional - bypass inlets, road grades and SAG inlet ponding depths
- optional - up to 2 catchment areas per inlet
- optional - default catchment characteristics, k values and overland travel times
- optional - pipe sizes and invert levels

22.10.15.1 PCdrain Requirements

Pit names

The pit name from 12d is assigned to both the structure and catchment name in PCdrain. These names cannot exceed 7 characters.

Pit type

The 12d pit type is transferred to the structure type in PCdrain. These names must match those specified in the PCdrain Inlet charts selected (**Data=>Inlet charts**). Select the desired inlet charts BEFORE importing the interchange file.

12d pit types with an “S” in the name are treated by 12d and PCdrain as a SAG inlet pit. 12d will strip off all characters after the “S” before adding the ponding depth. If a catchment string in set #1 is available for the SAG pit then the ponding depth will be calculated.The12d pit type will remain unchanged. A typical example would be a pit type “1TC” with the sag tick box on would become “1TC0.100” if a ponding depth of 0.1 was calculated.

Bypass Flow

When a catchment string is specified for the pit, the maximum depth before bypass flow commences is calculated. The lowest point on the catchment string is determined by draping it onto the drainage strings tin. The maximum depth before bypass is calculated pit setout level less the setout to grate offset less the lowest point on the catchment string.

PCdrain differentiates between pits (no surface inflow) and gully pits via the 12d pit type. The bypass flow strings can only be drawn within 1 pit diameter of the gully pits. Keep the bypass flow strings away from the PCdrain pits.

Catchments

Again, since PCdrain differentiates between inlets and manholes (using the 12d pit type), ensure that catchments are only drawn for gully inlet and NOT manholes.

12d to PCdrain Pit Data

12d	PCdrain
Pit-Setout-Easting (northing)	Table 1- Easting (Northing)
Pit-bypass-bypass pit	Table 1 - Bypass structure
Pit-Main-grate Level	Table 5- Surface Level Overrider
Pit-Bypass-Road Grade (%)	Table 5- Road grade over rider
Pit-Setout-Chainage	Table 5- Road Chainage
Pit-Setout-Offset	Table 5- Road offset
Pipe-Main -length	Table 7 - Pipe length
Pipe-Main-diam	Table 7 - Pipe1 size
Pipe-Main-# pipes	Table 7 - Pipe1 number of conduits
Pipe-Design-Alignment Modes	Table 7 - Alignment of conduits
Pit-Main-Ku(Kw)	Table 8 - Ku (Kw)
Pipe-Main-US invert	Table 9 - Upstream level of pipe
Pipe-Main-DS invert	Table 9 - Downstream level of pipe

12d to PCdrain Catchment Data

Perviouus Impervious	12d	PCdrain
Total	Area	Catchment Area
Total	Area (set 2)	Catchment Area2
Pervious	C minor and Major	Minor coef; Major coef
Pervious	C minor and Major (set 2)	Minor coef2; Major coef2
Pervious	Tc minor (Direct method only)	Time of concentration overrider
Pervious	Length	Length of overland flow
Pervious	Slope	Grade of overland flow

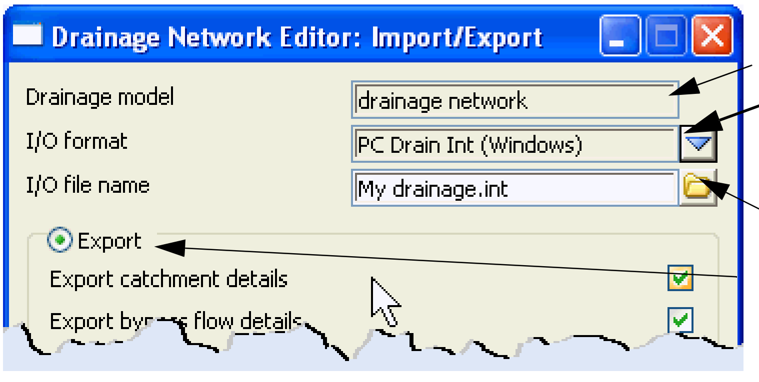
Perviouus Impervious	12d	PCdrain
Impervious	Length	Length of gutter
Impervious	Slope	Grade of Gutter

22.10.15.2 12d to PCdrain

1. Export the data to PCdrain selected via the Network editor

   Design=>Drainage-Sewer=>Drainage Network Editor

After selecting the drainage network, select the **Import/Export** button and the following dialogue will appear.



The drainage model will be completed by the editor.

Select **PCdrain Int (Windows)** from the drop down list.

Enter the name of the int file to be created.

Select **Export**.

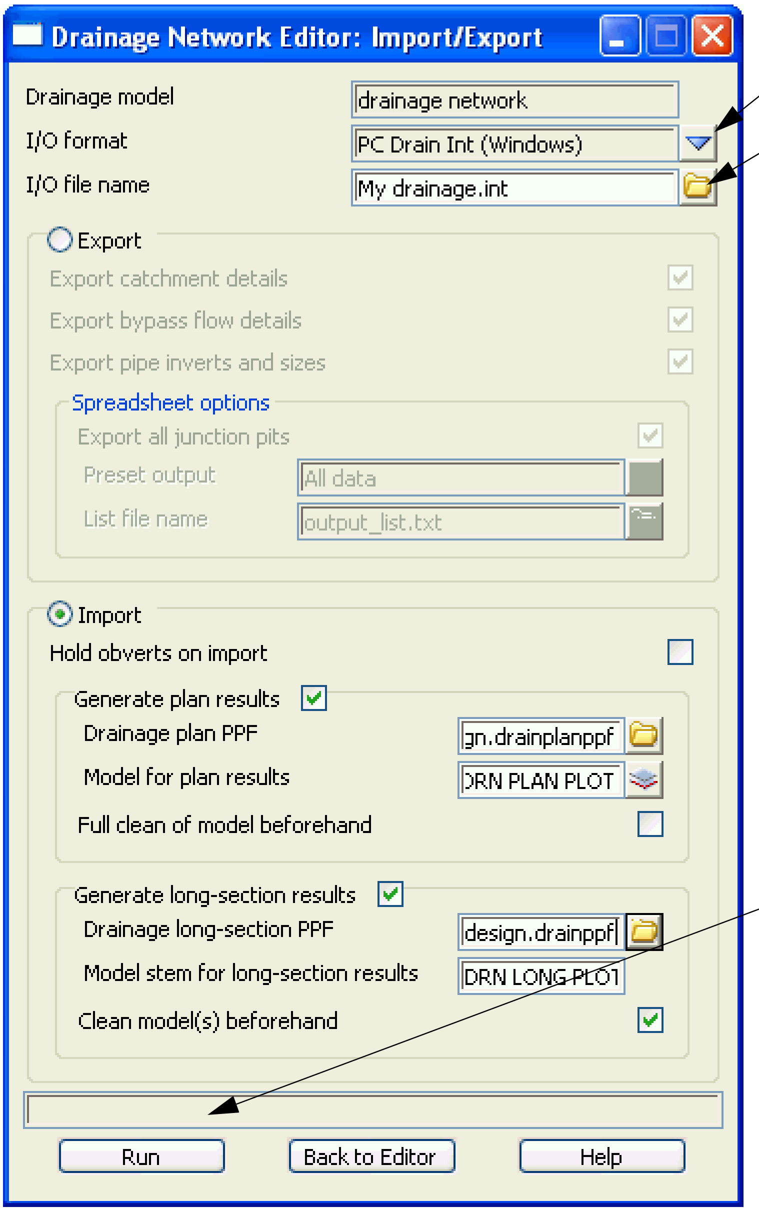
Select the **Run** button and the interface file will be created.

2. Launch the PCdrain for Windows program. If you have a project set up with the design parameters, rainfall data, inlet charts and gutter profiles then open it now and skip to step 8. Otherwise continue with step 5.
3. The Design Parameters can be set as desired with the menu selection **Data=>Design Parameters**.
4. Select the rainfall data using the **Data=>Rainfall** menu selection.
5. Select the inlet charts using the **Data=>Inlet Charts** menu selection. The pit types specified in 12d must be included in these settings. **More - PCdrain to 12d pit converter**
6. At least one gutter profile in PCdrain needs to be defined. These are set through the menu selection **Data=>Gutter Profiles**. The default gutter section name (**Road ID)** from 12d is **4d** and therefore it is recommended you create a profile with this name and your own description. If you have changed the profile names in 12d (through the spreadsheet interface or the Attribute editor) these new profile names will have to exist in PCdrain.
7. Save this file now so that you can retrieve it later if required. It can be used as a starting template for new jobs.
8. **File=>Import** from the menu. Select the file exported in step 1. The information from 12d may be viewed by selecting **Data=>Network** and then selecting the desired tabs.
9. The HGL level and the pipe elevation at the outlet should be set using the menu selection **Data=>Outlet.**
10. If you have not exported pipe data then the pipe size must be determined. Use the menu selection **Process=>Select Pipe Sizes.**

22.10.15.3 PCdrain to 12d

Export the results to 12d using the **File=>Export** menu selection. Note the name of the interchange file you are creating as you will need to enter it inside 12d.

Return to 12d and select **Import/Export** from the network editor and the following dialogue will appear,



Select **PCdrain Int (Windows)** from the drop down list.

Select the output file name you created in PCdrain.

Select **Run** and the file will be read into 12d and the drainage network will be updated. To see the changes in a section view, select **Regen.**

A listing of the data imported is stored on the 12d output window.

Inlets that have been specified as SAG inlets will have the ponding depth removed from the end of the PCdrain structure type before the data is stored as the12d pit type.

22.10.15.4 RAT2000 Requirements

See also Drainage overview

Basic Check List

1. All manholes must have a unique name (they become nodes in RAT2000)
2. Only one outlet allowed in the drainage model
3. First export should have **“Export default catchment/pit/overland parameters”** selected
4. First export should have **“Export pipe diameters and Inverts”** selected
5. Outlet conditions must be set in RAT2000.

6. Start-up data may be specified in the file **drainage startup.xpx** (the standard 12d system file path will be searched).

Overland flow with Inlet Capacity Check List

1. Overland flow lines must be within 1.0 metres/feet of the manhole
2. If inlet capacity curves are to be used then the curve name must match the 12d pit type with the road grade/crossfall data added.

23.0 Running RAT2000

Data is exchanged to and from RAT2000 via the interchange (*.xpx) file. The data exported from the 12d drainage model is appended to the xpx startup data specified in the file **drainage startup.xpx** (the standard 12d system file path will be searched). This file may be edited using a text editor or you may create your own default file by exporting your RAT2000 global data this file.

RAT2000 Requirements

Manhole names

All manholes will become nodes in RAT2000 and therefore they must have a unique name inside 12d.

Manhole Types

The manhole types are used to determine the pit inlet capacities if overland flow routes are selected. Fixed inlet capacities values may be set in the drainage.4d file while inlet capacites curves must be included in the drainage_startup.xpx file. The range of available curves for 12d to choose from are set in the drainage.4d file and the curves are determined using the pit type, road grade and road crossfall.

Pipe Types

Pipe types are not transfered to RAT2000.

Overland Flow

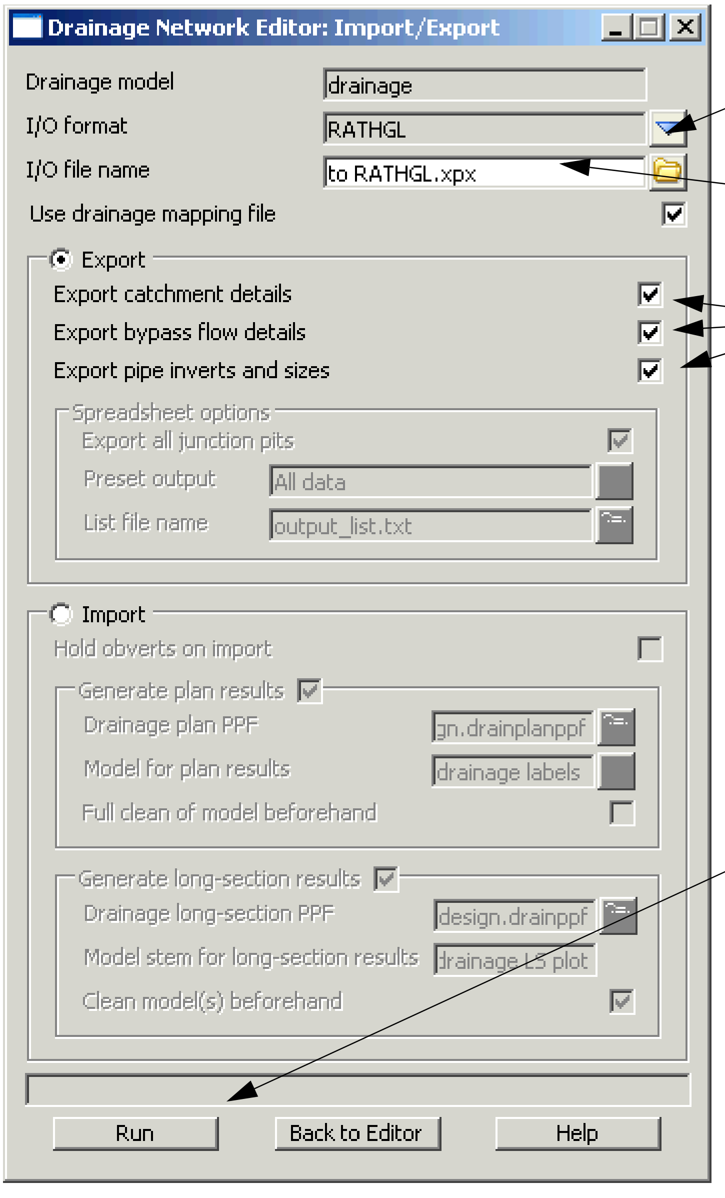
When an overland flow model is selected the pit inlet capacities and bypass routes are calculated.

23.1 12d to RAT2000

1. Export the data to RAT2000 by selecting

Design=>Drainage-Sewer=>More=>Pit/Pipe Design Interface

The following dialogue will appear.



Select RAT2000 from the drop down list.

Enter the name of the xpx file to be created.

Optional.

Not supported in RAT2000.

See notes below on **Updating an existing RAT2000 file**

Select the **Run** button and several checks will be made on the drainage network at this time. The interface file will be created.

23.2 Updating an existing RAT2000 file

Adding new Nodes and Pipes

If new manholes and pipes have been added to 12d then these new entities will now be included in the xpx file and imported into the RAT2000 project. 12d identifies the nodes in XP by the node names. Therefore if nodes names are changed in either program then they must be manually changed in both programs.

If a manhole is to be inserted in an existing link then the existing link must be deleted in RAT2000 so that the new links and manholes may be created in the void.

WARNING! If the tick box **Export catchment/pit/overland flow default data** is selected for the update then any changes to these defaults in RAT2000 will be lost. Two options are available to the user.

Option 1 - Preferred method if updating pipe lengths, invert levels for your design.

De select the tick box and export the data. The new node and links created in RAT2000 will not have all the defaults set but the existing nodes and links will not have revisions to defaults over-written.

Option 2 - Preferred method if you have created numerous new links and nodes.

Leave the tick box selected but export the RAT2000 to a temporary xpx fle first. This will save the existing setting. After RAT2000 is launched and loaded with the new data, import your temporary xpx file you just created to restore the original data.

Deleting Nodes and Pipes

If a node or link is to be deleted then delete the entity from both 12d and RAT2000.

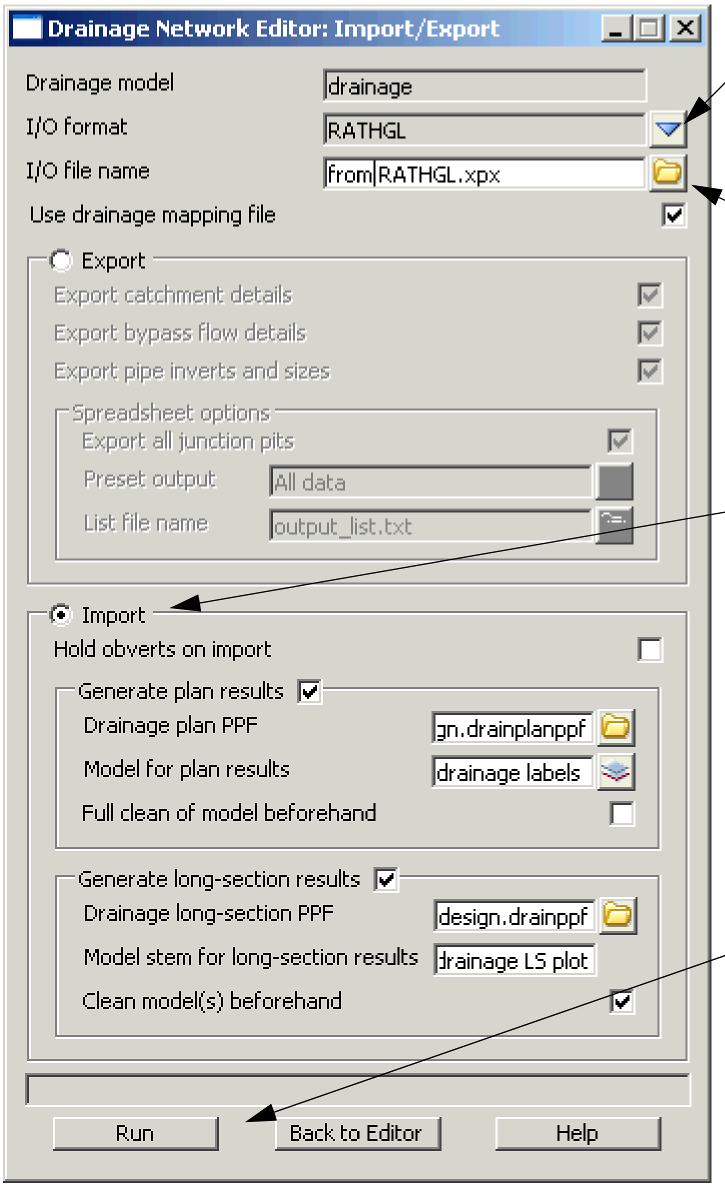
23.3 RAT2000 Results to update 12d

If RAT2000 is lauched automatically using option 1 above then as soon as RAT2000 is exited an interface file is automatically created (using the same name as the import xpx file). This file will now be read back into 12d to update the pipe network and import hydraulic results for drainage longsection plots.

Return to 12d and select

Design=>Drainage-Sewer=>More=>Pit/Pipe Design Interface

The following dialogue will appear,



Select **RAT2000**from the drop down list.

Select the output file name you created in RAT2000.

The remaining fields are not used for the import process.

Select **Import**

Select **Run** and the file will be read into 12d and the drainage network will be updated. To see the changes in a section view, select **Regen.**

23.3.0.1 XP SWMM Program Requirements

See also Drainage overview

Basic Check List

1. All manholes must have a unique name (they become nodes in XP SWMM)
2. Only one outlet allowed in the drainage model
3. First export should have **“Export default catchment/pit/overland parameters”** selected
4. First export should have **“Export pipe diameters and Inverts”** selected
5. Outlet conditions must be set in XP SWMM.
6. Start-up data (run times and hydrology data) may be specified in the file **master_drainage.xp** (the standard 12d system file path will be searched).

23.3.1 Running RATHGL/RAT2000 and XP-SWMM

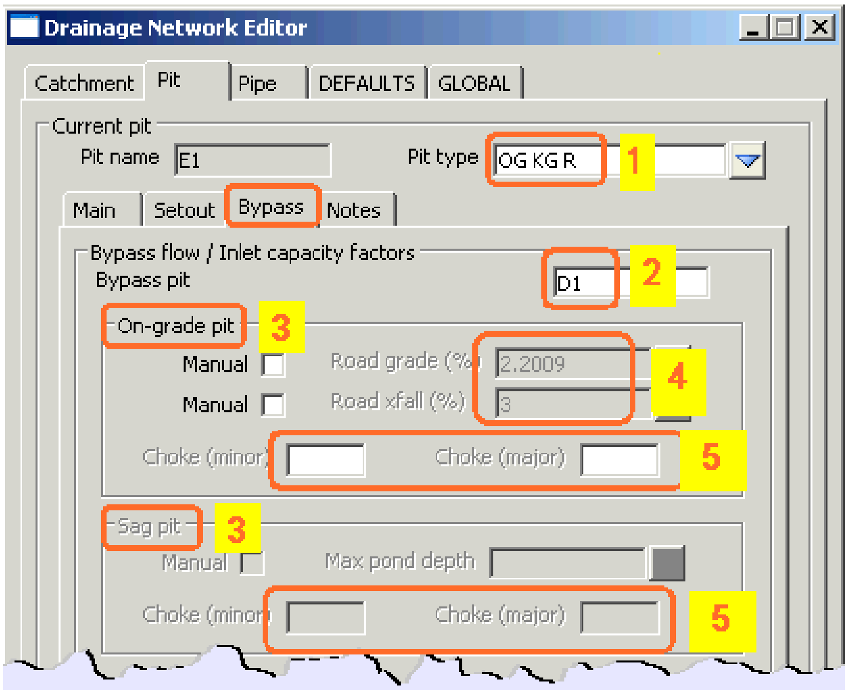
The drainage design with all three of the XP software programs follows the same methodology. The process is substantially automated with the XP-SWMM program so that the XPX file is automatically read by XP-SWMM and automatically created when leaving XP-SWMM.

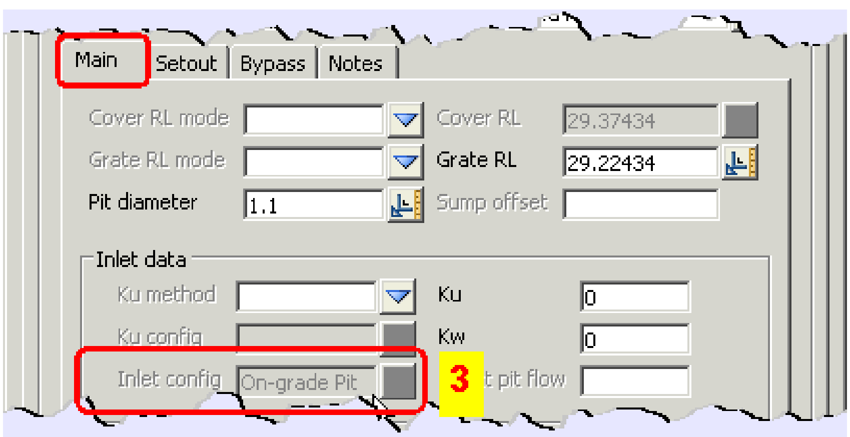
Drainage design with XP programs includes the following steps.

1. 12d creates an XPX file that is read by the XP programs.
2. The XP program is then run in the design mode to determine the pipe sizes and invert levels.
3. If bypass and overland flows are to be modelled then the inlet capacities need to be defined and then run the XP program in the Full Analysis Model.
4. The XP program creates an XPX file for 12d to import.

23.4 xpstorm and xpswmm bypass requirements

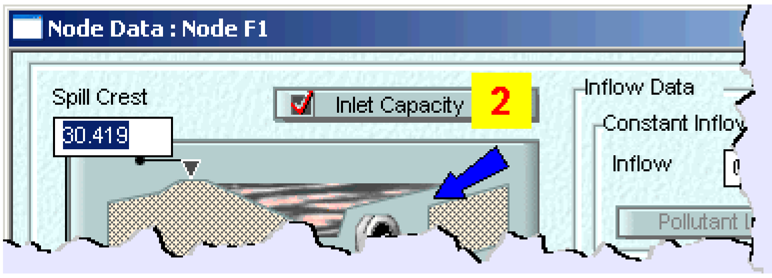
12d will select the inlet capacity curves for the xp programs when the following bypass settings are complete

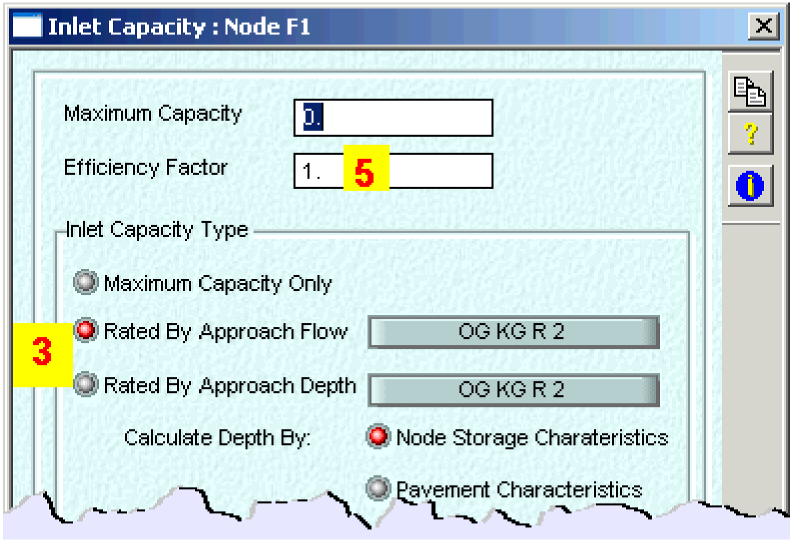




1. The 12d pit type determines the inlet capacity values or curves to be selected from.
2. The bypass pit must be defined for the pit. If the bypass pit follows the pipe network a xp-multi link with a natural channel is created. Otherwise a xp single link is created. The default xp natural shape is “Road Section”
3. The Ongrade/SAG pit on the Main tab determines whether depth or approach curves are exported to (details below).
4. If ongrade is selected then the 12d inlet capacity curves may be selected by road grade, crossfall, both or neither. (details below).
5. The 12d choke factors will be exported as the xp inlet effeciency factor (1 - 12d choke)

The numbers below relate the same numbers in the 12d panels above.





1.

2. Inlet capacity is turned on when 12d has a bypass inlet.

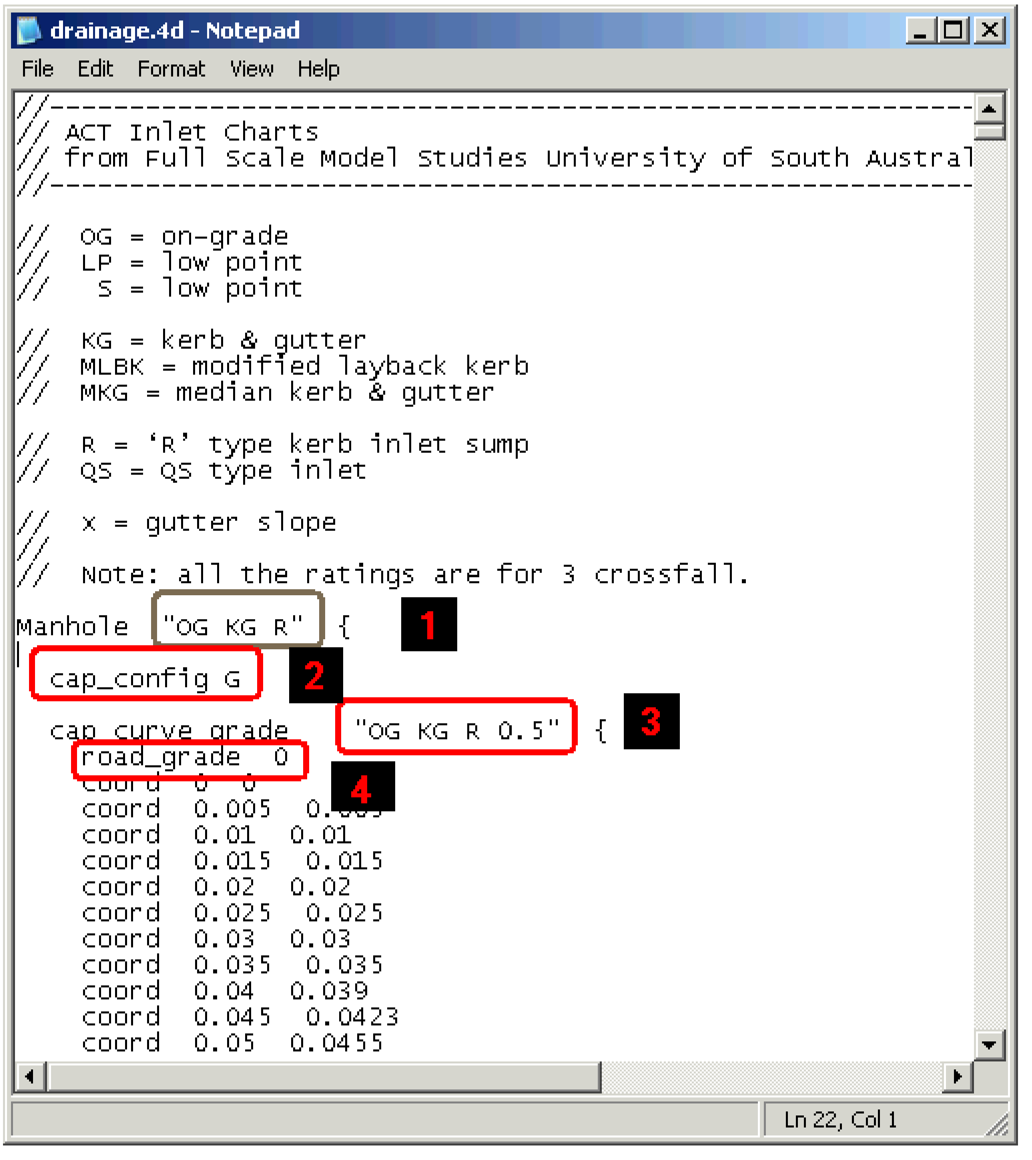
3. 12d **Ongrade** inlets export as **Rated by Approach Flow** and 12d **SAG** inlets export a **Rated By Approach Depth** with depth calculated by **Node Storage Characteristics.**

4.

5. The minor/major selection is the 12d export panel determines the effienciecy factors that are exported/ Efficiency factor = 1.0 - 12d choke.

12d Inlet Capacity Curve names

Below is an extract from the drainage.4d file for Canberra rating curves.



1. The manhole is the inlet type selected above.
2. cap_config G forces the inlet to be used as qa 12d ongrade inlet. An S would be SAG and an M a manhole.
3. The curve name indicated here is exported as the xp inlet capacity curve.
4. The curve name will be exported starting at road grade of 0. Since one curve has a road grade value all curves must have a road grade value and the user must have a road grade calculated on the 12d bypass tab.

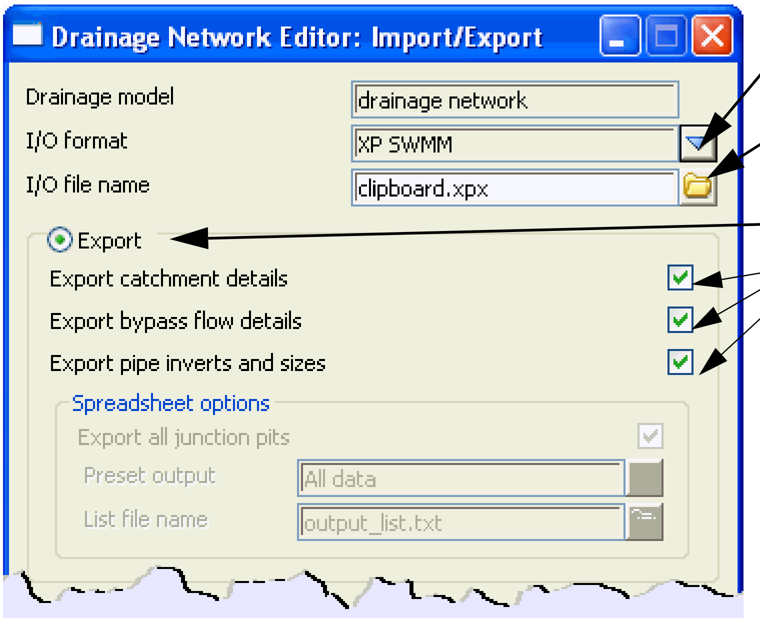
23.5 12d to the XP Programs

The x,y pit layouts and the cover/surface levels are obtained from your drainage network while the catchment and overland flow data comes from the models specified in the drainage interface dialogue.

The steps required to transfer the data to the XP programs are as follows.

1. Setup your drainage network models.
2. To create the XPX file for XP programs start the Drainage Network Editor and select **Import/ Export**

The following interfaces dialogues will appear.



Select **XPSWMM or RAT HGL** from the drop down list.

Enter the name of the xpx file to be created.

Select **Export**

Not used at this time.

Finally, Select the **Run** button to create the file.

If exporting to RAT-HGL the following dialogue will appear.

If you are using old versions of RAT-HGL (1996 or earlier) select use **Local** otherwise select **4D units** (eastings and northings).

Select process.

3. The XP SWMM program will automatically startup and load the XPX file.

   From within RAT-HGL, either select **File =>New** and follow the input prompt or load a file that contains all of the pit inlet rating curves, hydrological and design data without a pipe network. Many users have such RAT-HGL files setup so as to streamline the design process.

   The xpx file for RAT2000 will have the file startup.xpx added to it so that you may include all the startup global data that you require. The
   Fixed inlet capacities and rating curve names indicating road grade and crossfall may be set in the **drainage.4d** file. The format for these names is pit name-crossfall-road grade (ex SA2-3-4). These curves must exist in this file.
4. To read in the pipe file created above, select **Special =>Import Data** and select the xpx data file. Warnings will be given stating that several fields are inactive. This is expected as more data is sent to RAT-HGL than is needed at this time. Select the **Close Square** on the Help title area and the pipe network and catchments should now appear on the screen.
5. If you want RAT-HGL to redesign you network, change the analysis mode to Design mode by

selecting **Special=>Job Control** and **S**elect Design mode. Do not do this if you want to analyse the network you layed out in 12d (used for existing systems).

6. Select the rainfall events to design/analyse and the **LB** (twice) on **OK** to return to the layout. Now select the outlet and enter the starting tailwater levels.
7. Now you can run RAT-HGL (**Special =>Solve**).

23.6 XP Programs to 12d

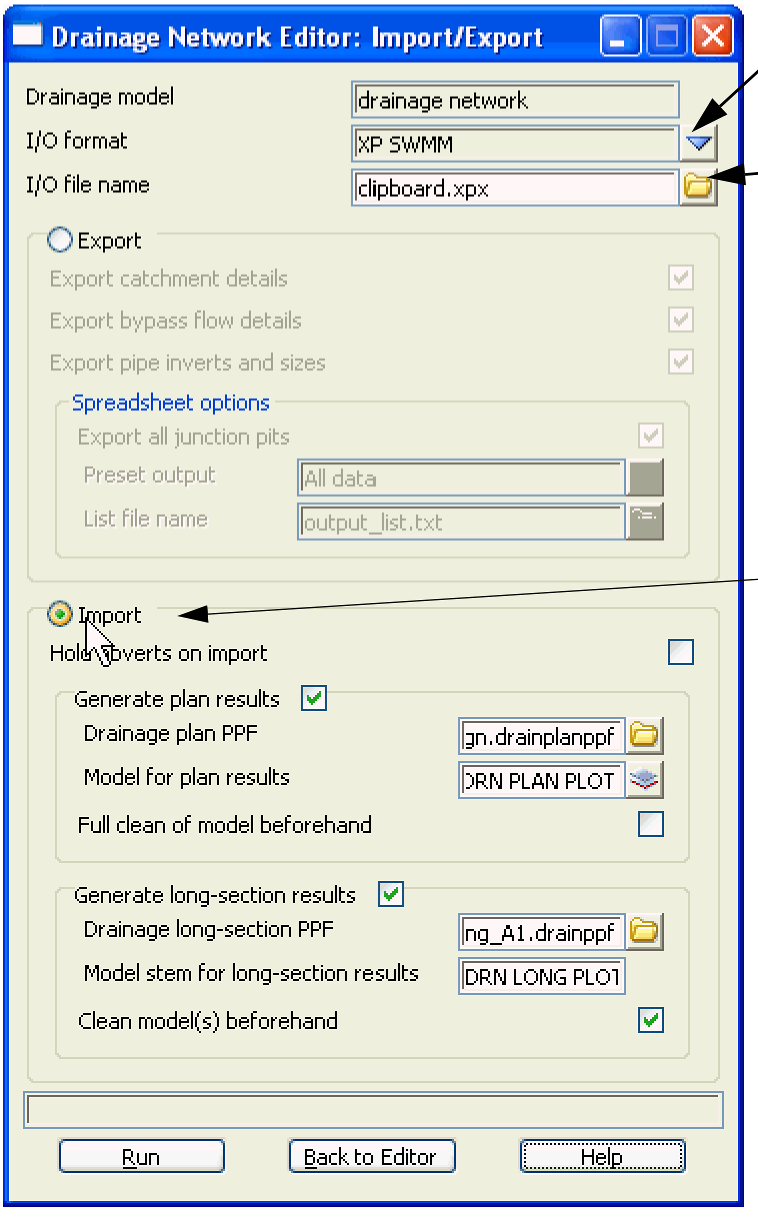
Once you have your design finished, the following steps are required to update your 12d model. Your design may contain several return periods in the analysis (Rp1 to Rp7) but 12d reads only the results from Rp1. The following table is taken from the RATHGL output file (*.out extension) and the results indicated are read back into 12d via the xpx file.

```
HGL  PIPE  NETWORK  ANALYSIS  SUMMARY    RETURN PERIOD   5 YEARS
       PROJECT:12d DRAINAGE LAYOUT

 --------------------------------------------------------
 |ITEM DIM| RESULT
 --------------------------------------------------------
 RN      -      1      2      3      4      5      6
 NN      -     1-1    2-1    1-2    3-1    1-3    1-4
 DN      -     1-2    1-2    1-3    1-3    1-4    1-4
 MT      -     99     99     99     99     99      0
 DHGL    M  29.258 29.258 28.139 28.139 27.269   .000  DS HGL
 DD      M    .225   .375   .375   .300   .450   .000  Diameter
 DCTL          HGL    HGL    HGL    HGL    HGL
 DO      M    .225   .375   .375   .300   .450   .450
 QO   M3/S    .033   .181   .237   .075   .334   .353  Flow
 VELD  M/S    .826  1.639  2.148  1.065  2.100   .000
 NORM    M    .106   .264   .271   .153   .289   .000
 CRIT    M    .152   .314   .345   .215   .396   .000
 KP      -     .00    .00    .00    .00    .00    .00
 SF    M/M   .0040  .0082  .0140  .0046  .0107  .0000
 LEN     M   45.93  18.44  54.65  31.31  49.91    .00
 HGLP    M  29.444 29.409 28.905 28.284 27.802   .000  US HGL
 DU      M    .225   .375   .375   .300   .450   .000
 VELU  M/S    .826  1.639  2.148  1.065  2.100   .000  Velocity
 UCTL          HGL    HGL    HGL    HGL    HGL
 KU      -    1.50   1.50   1.50   1.50   1.50    .00  Ku
 KL      -    1.50   1.50   1.50   1.50   1.50    .00
 KR      -    1.50   1.50   1.50   1.50   1.50    .00
 KW      -    1.50   1.50   1.50   1.50   1.50    .00  Kw
 UHGL    M  29.496 29.615 29.258 28.370 28.139 27.269
 LHGL    M  29.496 29.615 29.258 28.370 28.139 27.269
 RHGL    M  29.496 29.615 29.258 28.370 28.139 27.269
 UWSL    M  29.496 29.615 29.258 28.370 28.139 27.269  HGL PIT
 MWSL    M  30.297 29.615 29.403 28.875 28.340 27.369
 AF   M3/S    .033   .207   .023   .094   .040   .000
 IF   M3/S    .033   .181   .023   .076   .023   .023
 IC   M3/S    .036   .227   .026   .076   .023   .023
 BF   M3/S    .000   .026   .000   .019   .017   .000
 ID      M    .000   .000   .000   .000   .000   .000
 ITW     M    .000   .000   .000   .000   .000   .000
 IVEL  M/S    .000   .000   .000   .000   .000   .000
 IVD  M2/S    .000   .000   .000   .000   .000   .000
 FC      -             *             #      #      #
 --------------------------------------------------------
```

In addition to the results, the following input data is read back into the 12d model so that it may be exported back to RATHGL in the future (if required). 100% of your RATHGL data is not included in the XPX formats and the contents of the XPX file will depend upon your design mode. Therefore, use caution if you read an XPX file into an existing RATHGL model and check your data once inside RATHGL.

1. From within RAT-HGL, produce an XPX file for 12d to read by selecting **Special=>Export Data** and following the default prompts**.**
2. From within 12d, select the **Import/Export** button on the Drainage Network Editor.The fol-

lowing panel will appear.



Select to select **RATHGL** or **XP-SWMM**.

Select the file name specified in step 1

Select **Import.**

Select **Run** to update the drainage model and import hydraulic/ hydrological results.

23.6.1 Running Micro Drainage - WinDes

See also Drainage overview

Basic Check List

1. All manholes must have a unique name (they become nodes)
2. Only one outlet allowed in the drainage model
3. First export should have **"Export default catchment/pit/overland parameters"** selected
4. First export should have **"Export pipe diameters and Inverts"** selected
5. Outlet conditions must be set in Win DES.

23.7 Micro Drainage - WinDes Requirements

Pit - Pipe names

The pipe names are used in the WinDes interface NOT the pit names. Therefore the pipes are often named separately to follow the WinDES required naming convention.

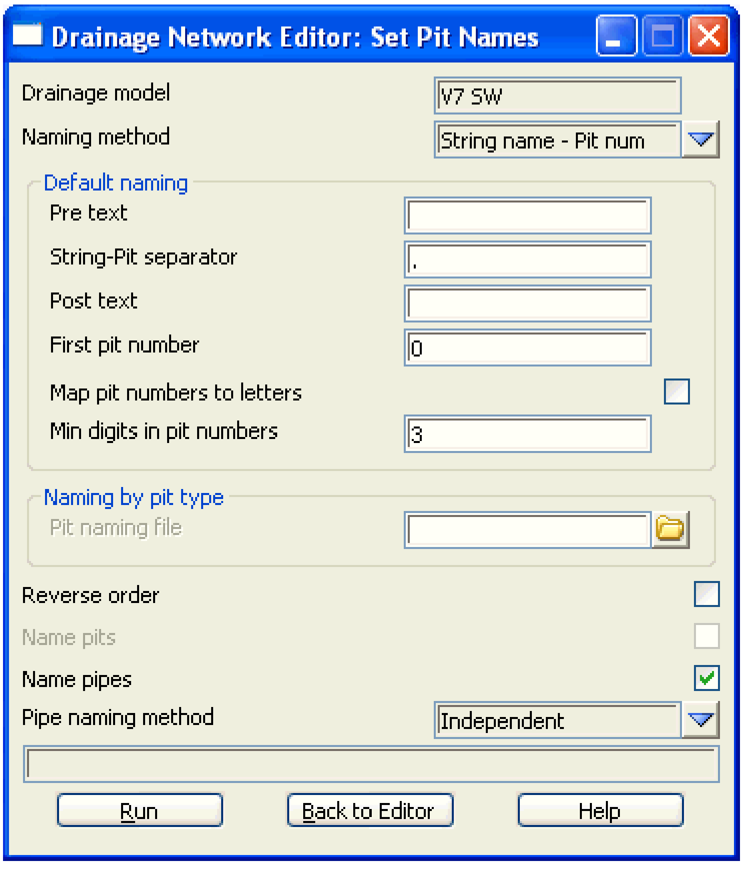
12d string names

The trunk line drainage string must be named "1" and then moving from upstream to down stream the branch line strings increase 2,3,4 etc. If the branch lines have sub branches then they are named in a similar upstream to downstream sequence. The sub branch string are named before moving downstream on the trunk line. If this string numbering system is not followed 12d will renumber the strings and request the user to run the drainage misc utilities to rename the pipes and pits.

12d pipe names

The trunk line pipe segments start at the upstream end at 1.000 and increase downstream (1.001, 1.002 etc). The assign pit names will automatically name the pits for you if you have followed the string naming convention discussed above. "Number of digits" must be set to 3 to ensure the leading zeros are used.

The panel below shows the naming convention.



12d pit names

The pit names may be set to any desired naming convention. Using the same names as the pipes makes the pit/pipe identification easier for the user but it is not required. As of April 2007 the maximum length is 8 characters but Windes is planning to remove this restriction.

The *.sws or *.fws files may be open directly inside WinDes.

23.8 Global /default variable mapping

The default 12d drainage mapping file maps the following Windes data

Windes	12d attribute name	pit, pipe or model	default value
Return period		model	
Time of entry		model	
UK region		model	
min grade		model	
min cover		model	
Outfall manhole name		pit	
Roughness method		model	
pipe file		model	blank. standard.pip
manhole file		model	blank. standard.mhs

23.9 Additional Notes

Pipe Resizing

Pipes are automatically resized when opened in Windes and the sizes used are from the specified *.pip file (standard.pip by default)

When locked pipe sizes are sent to Windes the pipe sizes can be increased but not made smaller.

Pit Resizing

Pit diameters are automatically updated by Windes depending on the size of the attached pipes. This resizing is determined by the Windes file, standard.mhs.

23.10 Open channel drainage design.

Win Des can (hydraulically) model any shape of conduit. The conduits are given numbers between 1 and 100. 12d selects these conduits by setting the **pipe type** to **WINDES** (case sensitive) and pipe diameter to the WinDes conduit number divided by 1000. (examples 0.032 indicates WinDes conduit 32).

12d can create the conduit shape by applying a template to the drainage string. The user must create a template with the same name as the WinDes shape number (32 for example). The templates are created from the main menu

Design->Template->Create Edit

For more information about template design please refer to the 12d training manual.

Once the template has been created it is applied to the drainage string to create the cross sections and strings to represent the conduit shape.

From the main menu **Design=>Drainage-Sewer=>User=>Drainage Volume Calculations**

For more information see **Drainage line excavation volume calculations**.

23.10.1 Drainage Excavation Quantities

Position of option on menu: Design => Drainage => Reports => Excavation Quantities

See Also

Drainage overview

This routine uses 12d templates to calculate the excavation volume for all of the drainage strings in a model. An option to create section for a tin on top of the pipe is also available so that the drainage long sections can include hatching between the obvert of the pipe and the design tin under roads.

Templates with names set to the pipe diameters (times 1000) are used for the calculations, thus trench shapes can be customised and over excavation for bedding materials can be included. Net area calculations to exclude pipe area are not supported.

Key points

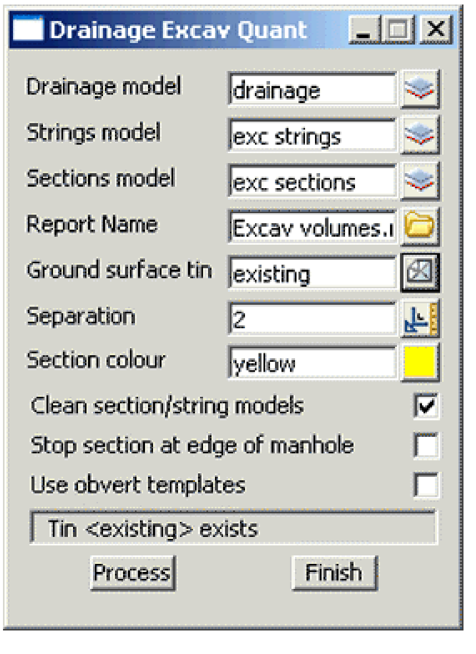
1. One template for each pipe size (mm)
2. If obvert templates are used, add the prefix “obvert “ to the pipe size
3. Carefully consider the tin selected.

A template must exist for each pipe size in the model (pipe size x 1000). For example a 0.3m pipe will require a template to exist named 300. A 0.5ft pipe would require a template named 500. A sample template library is included in the 12d library in the file **pipe_template.tpl**.

The templates are run along the strings and the total volumes are reported. Volumes for each strings are given in the report file.

If a tin is created from these strings then volumes by depth can be determined using **Design=>Volumes=>Exact=>Tin to tin**

On selecting the **Excavation quantities** option, the **Drainage Excavation Quantities** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Drainage model input box

Model to contain all of the pit and pipe network to be worked on.

Strings model model box

Strings generated from the templates will be stored in this model

Sections model model box

Sections generated from the templates will be stored in this model

Report name input box

cut and fill volumes will ne sent to this report

Ground Surface Tin tin box

tins from which the volumes will be calculated

Separation real box

distance between the sections

Sections colour colour box

Sections generated from the templates will be assigned this colour (strings colours are defined in the templates)

Clean section/strings model tick box

Delete the strings in these models before processing.

Stop section at edge of pit tick box

Template are run from pit centre to centre if this is not selected. The templates stop at the edge of the pit if selected. This is often selected with the following option ***Use obvert templates.***

Use obvert templates tick box

Templates must be named with the prefix "obvert". i.e. ***obvert 300.*** *The template is still run along the invert of the pipe but the user now has a section "set" of templates that can be used to create a tin on top of the pipe as well as below.*

An example report file follows.

```
---------------------------- BEGIN APPLY TEMPLATE REPORT ------------------------------

 apply template to string report -

   string         E
   tin            design
   separation         10.000
   left template  375
   right template 375
   cut volumes and areas are negative
   fill volumes and areas are positive


chainage- ------sectional  information------  ------intermediate  information----    ---------accumulative  information-------
         ---------- ---cut area --fill area ------------ ----cut vol ---fill vol    -cut volume-- -fill volume- ---balance---


    0.000                -1.434       0.000                                              0.000         0.000         0.000
                                                             -0.771       0.000
    0.550                -1.367       0.000                                             -0.771         0.000        -0.771
                                                            -14.222       0.000
   10.000                -1.642       0.000                                            -14.992         0.000       -14.992
                                                            -15.293       0.000
   20.000                -1.416       0.000                                            -30.286         0.000       -30.286
                                                             -1.845       0.000
   21.313                -1.393       0.000                                            -32.130         0.000       -32.130
                                                             -0.794       0.000
   21.863                -1.493       0.000                                            -32.924         0.000       -32.924

total cut                         -32.924
total fill                          0.000
balance                           -32.924
ie excess of cut over fill         32.924

---------------------------- END APPLY TEMPLATE REPORT -----------------------------
```

23.10.2 Attribute Editor

Position of option on menu: Design => Drainage => More => Top Ten attributes editor

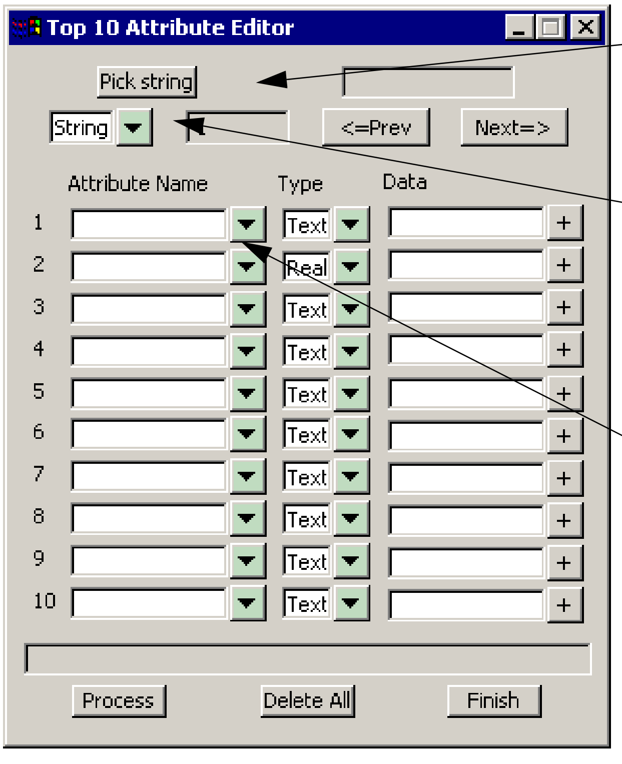
Most of the detailed catchment data is stored within 12d as user defined attributes. These attributes are automatically created by 12d when required but you are free to change them or add more as desired. The attributes may be exported to a spreadsheet and edited and then imported back into 12d or edited inside 12d using this panel.

See Also

Drainage overview

Usage

From the menu select **Design => Drainage => More => Top ten attribute editor**



First Select **Pick** to select the string that contains the user attributes (the drainage string). The strings will be highlighted in white when they are selected.

All catchment data is store with the pits in drainage strings. To access the pit attributes, select the drop down icon and then select **Pit**. A circle will be drawn around the pit selected. **Next** and **Prev** will now move you from pit to pit.

Select the drop down icon and then select the **Attribute Name** from the list of existing user defined attributes. These attributes include all of the attributes in the model that the string exists in.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
pick string	button		

used to pick the initial string in a model

string-pit-pipe	choice box		**string,pit,pipe**

select the type of attribute to be displayed. Pit and pipe attributes are only available for drainage strings.

attribute name	input box		

3 top 10 attributes lists are maintained (pit, pipe and string). The attributes that you can select from are all of the attributes that exist on all of the strings in the model. If the attribute does not exist for the *string/pit/pipe that you are displaying the **data** field will display **Not found.***

type choice box **Text, Real, Integer**

for existing attributes this will display Text, Real or Integer.
When defining a new attribute select the type of data to be stored in the attribute

data input box

the data stored in the attribute is displayed/edited/created in this field.

<= prev button

move to next string in the model
pit on the string
pipe on the string

next => button

move to next string in the model
pit on the string
pipe on the string

process button

updates the attributes displayed in the dialogue.

Notes:

First LB select Pick to select the string that contains the user attributes. All catchment data is stored with the pits in drainage strings. The strings will be highlighted in white when they are selected.

To access the pit attributes **LB** this field then select **Pit**. A circle will be drawn around the pit selected.

LB the **Attribute Name** field and then select from the list of existing user defined attributes. These attributes include all of the attributes in the model that the string exists in. They may not be defined for the string you are editing. If the string does not have that attribute defined **not found** will be displayed in the **Data** field.

To change the value for the attribute enter the new value in the **data** field. If the attribute does not exist, deleting the **not found** text and adding data will create it. The following message will be displayed whenever you are creating a new attribute.

23.10.3 Delete All Attributes in a Model

Position of option on menu: Design => Drainage => More => Delete all attributes in a model

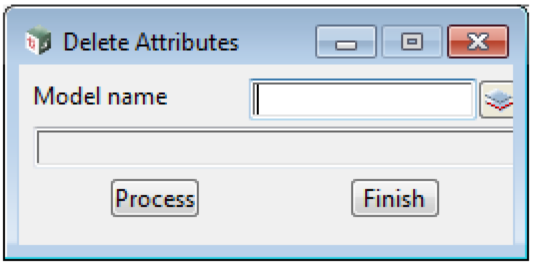
See Also

Drainage overview

This option deletes all model, string, pit and pipe attributes in the model specified. This option allows the user to "start from scratch" while maintaining the drainage, catchment and bypass flow strings.

Usage

From the menu select **Design => Drainage => More => delete all attributes in a model**



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model name	input box		

the model specified will have all of its attributes and all of the strings attributes deleted

Field Description	Type	Defaults	Pop-Up
Process	button		

Deletes all of the attributes

Notes: THERE IS NO UNDO!

23.10.4 Flooded Width Flow Analysis and HEC-RAS

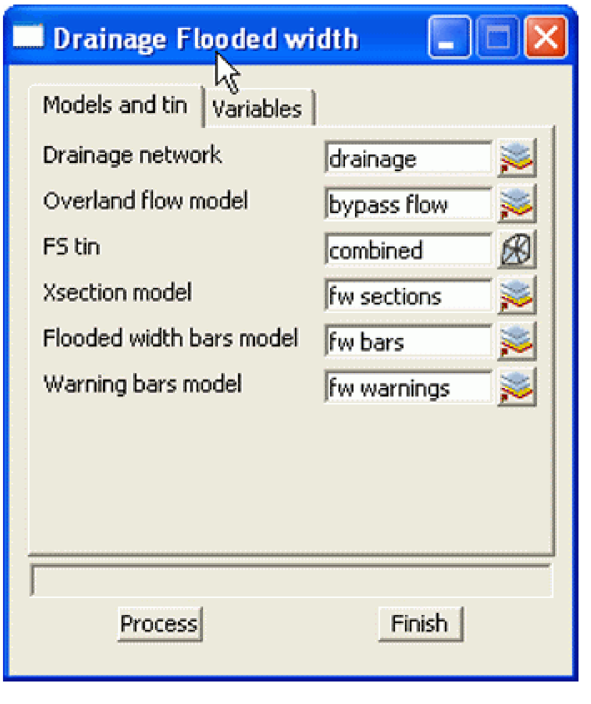
Position of option on menu: **Design =>Drainage-Sewer =>Calc flooded widths**

The Calculate Flooded width procedure creates cross sections along the **bypass flow** paths and then calculates the flooded width at each section using Manning's normal depth calculations. A HEC-RAS project (same name as the bypass flow string) is also created for each line. The flooded width is indicated on each section as a blue line if it is less than a user defined width and a red line if the flooded width exceeds the limit. Details of the calculations such as the velocity, depth, wetted perimeter and slope can be exported to a spreadsheet for further analyse (velocity times depth calculations for example). The discharges imports from the urban stormwater design packages are shown in the following table.

Design Program	Discharge Event
PCdrain	Minor ARI
Drains	Maximum flow event analysed
ILSAX	Maximum flow event analysed
RAT HGL	First return period analysed

The user defines the length of these sections and the interval at which they are to be spaced. 12d calculates the normal flow depth interpolating the pit approach and bypass flows from the hydrology models (ILSAX, Drains, PC Drains or RAT HGL). The cross sections are taken perpendicular to the flow line and the slope is for the normal depth calculations is determined using the distance along the flow line and the change in elevation between the two lowest points in the primary flow channel. The flow line need not intersect the low points on the section but the flow line does mark the primary flow channel. If the depth of the flow exceeds the banks of the primary channel, then all adjacent flow channels will be considered as active flow area.

On selecting the **Calc flooded widths**, the **Drainage flooded width** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Models and tin

Drainage network model box drainage

Existing drainage strings must have approach and bypass flow pit attributes. See doco below

Overland flow model model box bypass

Cross sections are cut perpendicular to these existing strings

FS tin tin box design

This tin is used to cut the cross sections and to determine the slope for normal depth calcs

Xsection model model box fwsections

Model is cleaned before processing. Calculated values such as velocity and slope are stored with these strings.

Flooded width bars model model box fw bars

Model is cleaned before processing. Blue or yellowstrings with levels indicate normal depth flooded widths.

Warning bars model model box fw warnings

Model is cleaned before processing. Strings created when vel x depth exceeds limit.

Variables

Max flooded width input 2

the limit where the blue flooded width bars turn red

Manning's n input 0.014

The n value to be used in the normal depth calculations.

Flow correction factor input 0.8

The is the factor described in ARR 1987 for calculating depths of flow in gutter channels.

Distance between sections input 5

The interval at which cross sections and therefore flooded width will be calculated along the flow path.

Section Length input 5

The length of each cross section. The cross section will be centred on the overland flow path.

Trim sections at levee tick box off

Trims the cross section at the crest on either side of the flow channel. A levee point is the crest in the cross section found as you move away from the flow line location.

Levee tolerance input 0.1

The amount the cross section needs to drop as you move away from the centre line in order to identify a levee.

VxD warning limit input 0.6

The velocity times depth limit that when exceeded will cause a flooded width bar to be generated in the warning bars model.

VxD warning colour colour box orange

Colour of the velocity x depth bars

Process button

Clean output models, calculate flooded width and create HEC-RAS projects named from the string names.

Finish button

remove the panel from the screen

23.11 Limitations where overland flow lines join

Where overland flow lines converge at an inlet, 12d does not know the flow split between the 2 approaching paths. Therefore, 12d uses the total flow from all lines as the flow at the inlet for each line. This may overestimate the flooded width along the flow lines at these points.

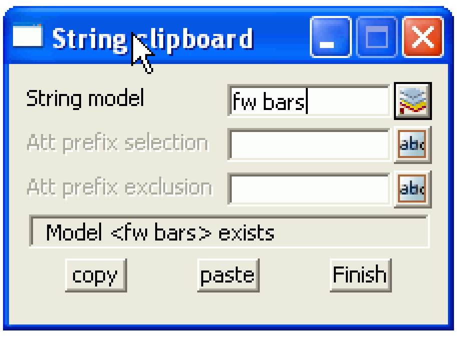
23.12 Limitations at SAG pits

The flow width are not shown adjacent the sag inlets. The depth of flow due to ponding and the approach flow coming from several directions may overestimated flooded width in these areas. Therefore not flood depths are calculated approaching SAG inlets.

Summary Tables

The hydraulic calculations and warning messages are stored as string attributes on the flooded width bars. If these attributes are exported to a spreadsheet via the clipboard a summary table my be created. To copy these attributes to the clipboard select

File IO->User->String attributes-properties to/from clipboard



The **string model** may be either the flooded width bars or the warning bars. Both models of strings contain attributes on the strings.

The prefix selection and prefix exclusion are filters for reducing the number attributes that are exported to the clipboard.

23.13 Cross Sections, Discharges and Warnings

The analyse flooded width will proceed along each flow path and identify every pit on the line. Cross sections will be constructed in the model with the length and interval entered in the input dialogue. These cross sections may be plotted using the main menu selection **Plot=>X plot=>X plot**. The **Sort Sections** must **not** be selected for these sections to be plotted.

Discharges will be determined for each cross section by linearly interpolating the discharge using distance between the pits. The bypass discharge (pit attribute - calculated bypass flow) will be taken from the upstream pit and the approach discharge (pit attribute - calculated approach flow) from the downstream pit.

The slope is calculated by subtracting the lowest points nearest to the centre line and dividing the cross section separation. The levee tolerance is NOT used for locating this point thus any rise in section moving away from the centre line marks the end of the low point search in that direction.

12d will give warning messages in the output window when it encounters the following conditions and these messages will be stored as string attributes on the flooded width strings. Descriptions of these messages follow.

Inverts do not go downhill

12d locates the lowest point (adjacent to the flow line without moving over a local crest) on each cross section to calculate the slope between the cross sections. This message indicates that the downstream minimum elevation is higher than the upstream minimum elevation.

Sometimes flow lines will go uphill. If you have specified an overflow from a SAG location then the flow line will go uphill until it crosses the overflow crest.

If the flow line is not supposed to be going uphill at this section, check to see where the flow line intersects the cross section located upstream of the one identified in the warning message. If it is in a local sag point that is not the lowest point on the section, move the flow line.

The program will use a slope of 0.5% to calculate a width at this location. This results in very wide flooded width sections to draw the user’s attention to the problem area.

Vertical Walls Assumed at the Ends of the Cross Sections

If the depth of flow exceeds the ground surface elevation at the ends of the cross section a warning message the warning message shown above is shown. The cross sections causing the warning follows.

The vertical wall is placed at cross section chainage –20. Note that the flow line is always at chainage 0

23.13.1 Convert Drainage String to Polyline

Position of option on menu: Design => Drainage => More=> Convert drainage string to polyline

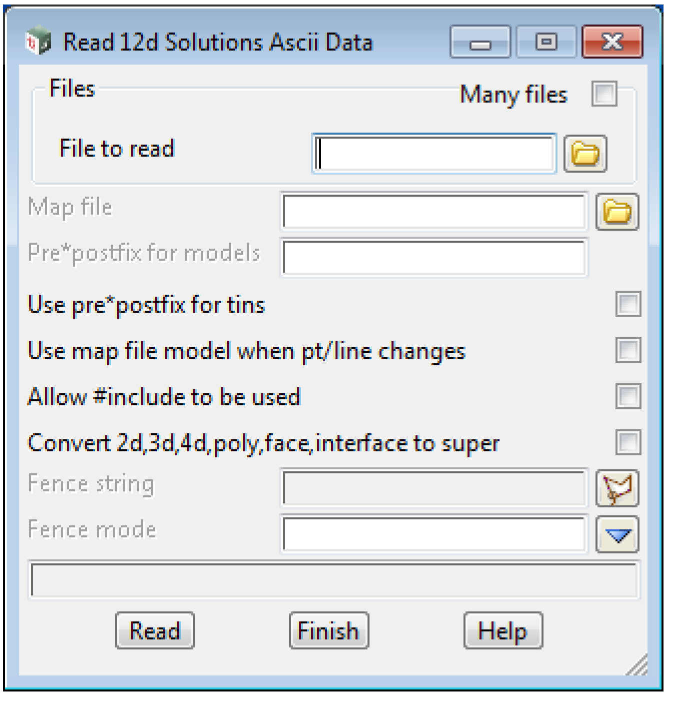
See Also

Drainage overview

All drainage strings in the specified model are exported to a 12da file. When this 12da file is imported back into ***12d Model*** the strings will be converted to polylines. Import the strings using

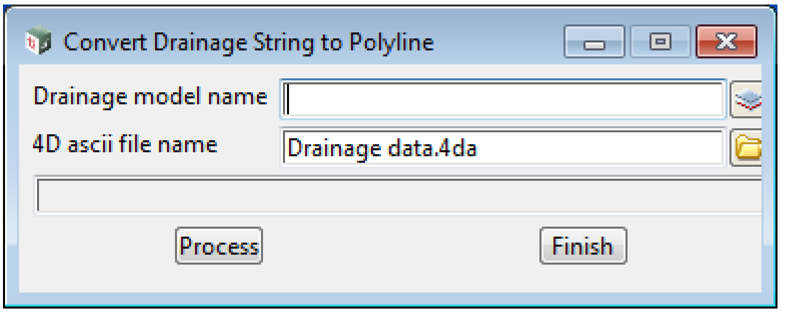
File_io=>Data Input=>12da/4da data

Specify a prefix for the model when it is read back in. Otherwise the strings will be placed in the same model as the originals.



Usage

From the menu select **Design => Drainage => More=> Version 6=> Convert drainage string to polyline**



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

drainage model name input box

Drainage model containing the drainage strings

12da file name input box drainage data.4da

A temporary file that will hold the converted string data.

process button

Exports the drainage strings and converts the file to polylines.

23.13.2 Drainage.4d file

A sample pit entry in the drainage.4d file is as follows:
Enhancements have been made if exporting to the Drains program. See Pit Families and Pit Groups
The drainage.4d file controls many of the settings for the pit and pipes types inside 12d. This section details the format of the drainage.4d file. Changes to this file take effect only after 12dmodel has been restarted. If there are any errors in the drainage.4d file they will be listed in the output window at startup (just after the shp file listing). The error in the drainage.4d file is generally located just above the line indicated in the output window.

All text to the right of the // is ignored by 12d (comments). There are pit and pipe type definitions. Because of historic reasons the pit types are defined using a **Manhole** command.

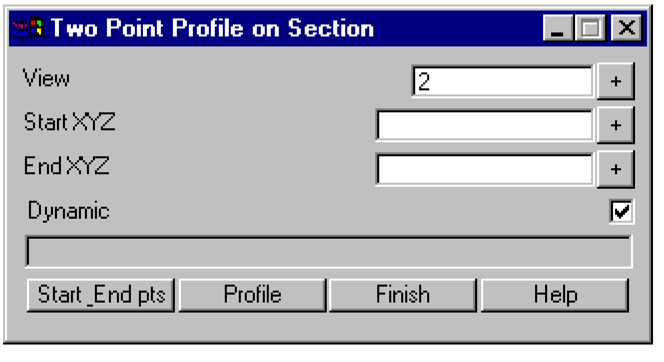
Many of the settings in the drainage network editor (DNE) can be controlled by the pit types and the pipe types defined in the drainage.4d file. The **special manhole attributes** defined below control these fields.

The original **drainage.4d** file is found in the “program files\12d\12dmodel\10.00\set_ups” directory. **Do NOT** change this file. Copy it into your user folder “\12d\10.00\User” and edit it there. Files in the user directory are used by preference and they are never over written by a 12d update.

Two Point Dynamic Profiles

If at any time you want to obtain a temporary section view, 12d has a dynamic profiling capability (2 point profile). First, add your tin design model to the section view.

Next RB on the section view title area and then LB select 2 points. The following panel will appear.



Select **Start_End pts** to begin and then pick/accept a point on a Plan View Now as you move your cursor the section view will be dynamically updated (the dynamic box should be checked on and a surface tin model added to the section view). When you obtain the cross section you want LB select then accept (MB) the second point.

LB select Finish when done.

23.13.3 ILSAX Editors

Position of menu: Design => Drainage => More => ILSAX Editors

The **ILSAX editors** walk-right menu is

See ILSAX Pipe/Catchment Editor

See ILSAX Rainfall File Editor

23.13.3.1 ILSAX Pipe/Catchment Editor

Position of option on menu: Design => Drainage => More=> ILSAX editor=>ILSAX pipe editor

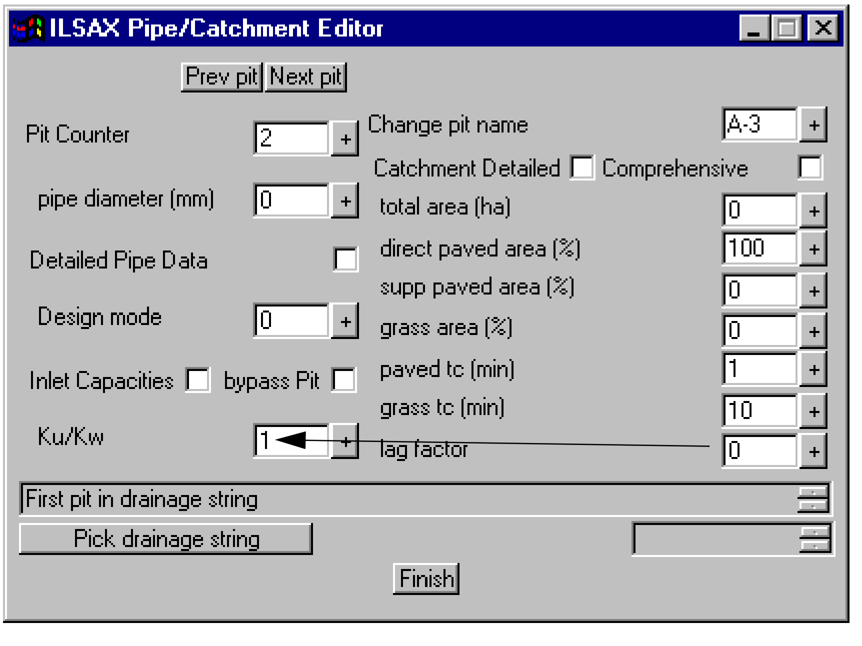
See Also

ILSAX Rainfall File Editor
Drainage overview

The ILSAX pipe data editor allows the user to edit pipe and catchment parameters. Most commonly used functions are supported but some of the less used functions are not included at this time. The same names have been used as those in the ILSAX drainage manual for easy reference.
Unlike the rainfall editor, all of the data in the pipe data editor is stored with the drainage string as user attributes. These user attributes can be changed using the editor (recommended for the novice user) or they may be output to a spreadsheet, changed and then read back into 12d.
CAUTION: If the drainage string is deleted then all of the attributes are deleted at the same time.

Usage

The ILSAX pipe data editor is accessed by selecting **Design=>Drainage-Sewer=>More=>ILSAX Editors=>ILSAX Pipe file editor**. The following dialogue will appear.
.



The panel to the left is only and example of what the editor panel may look like. This dialogue will change in size and complexity depending on the check boxes selected. The left side of the dialogue is reserved for pipe and pit data while the right side contains catchment data. The first step is to select the drainage string to be edited. Click on **Pick drainage string** and then select the drainage string from one of the views.

There are two ways to move between pits. The **Prev pit** and **Next pit** will move the user between the pits with the current pit name been shown in the **Change pit name** field. Do **<u>not</u>** use the **Change pit name** field to move between pits. It will not work! This field is used to manually change the pit name. Entering the **pit number** and pressing **Enter** is the second method for selecting pits. This is a good way to move between pits on long drainage lines (from pit 20 to pit 1 for example).

The **pipe diameter** will change the diameter of the pipe leaving the pit in the direction of increasing chainage. Note that the invert level of the pipe will remain fixed as the obvert level changes.

The most common **Design mode** is 1 for design. This ignores the present pipe size and resizes the pipe as required.

When the **Inlet capacity** and **Bypass pit** tick boxes are checked, additional fields are added to the dialogue. These will be discussed in the section 5.0 above

The **Catchment Detailed** and **Comprehensive** tic boxes also add additional fields to the dialogue. Again, the ILSAX drainage manual contains detailed descriptions of these parameters.

23.13.3.2 ILSAX Rainfall File Editor

Position of option on menu: Design => Drainage => More => ILSAX Editor=>ILSAX rainfall file editor

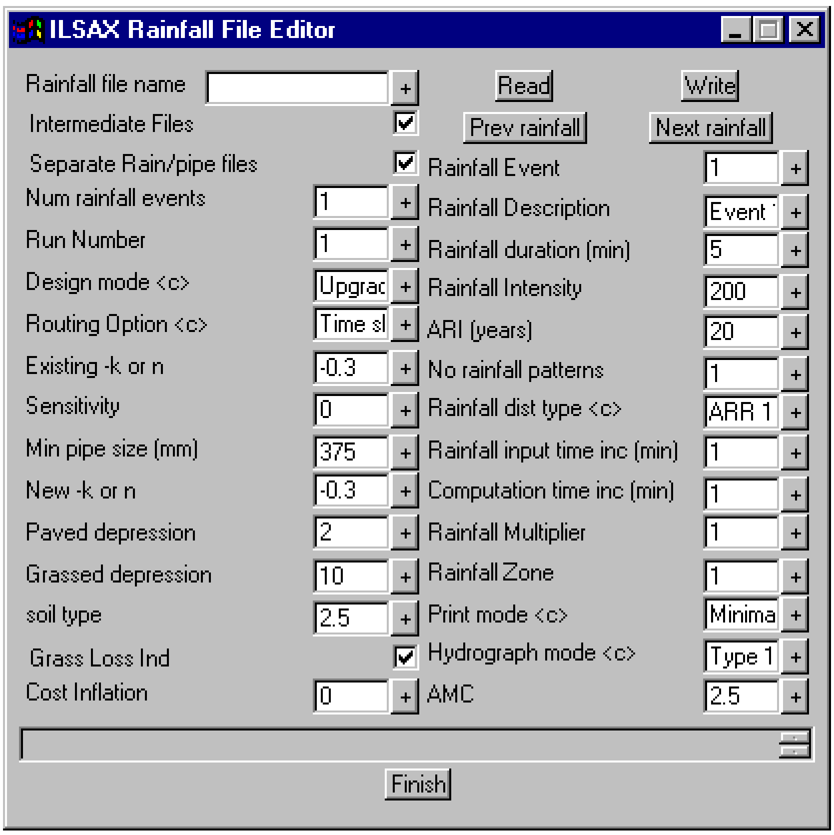
The ILSAX rainfall file editor assists in the creation and editing of the ILSAX rainfall files. It is truly a file editor and no data is stored inside the ***12d Model***. Most common features of the ILSAX rainfall file are included but some have been omitted as they have been rarely used. The files can be created using the editor and then manually edited using a word processor if required.

See Also

ILSAX pipe editor
Drainage overview

Usage

This panel is accessed from the menu selection **Design => Drainage => More=> ILSAX editors => Ilsax rainfall file editor**



The **Rainfall file name** must be specified before the **Read** or **Write** buttons will operate. If you want to create a file, fill in the **Rainfall file name** field and then **LB** select **Write** to save the data.

Intermediate Files and **Separate Rain/pipe files** must be ticked to have ILSAX run within 12d.

The minimum value for **Num Rainfall Events** is 1.

The remaining data in the left column is the data for the ILSAX R3 and R4 cards and the data in the right column is the data for the ILSAX R2,R6, R6B and R8 cards. Please refer to the ILSAX manual for a description of these values. The fields are not in the same order as the ILSAX files but instead the fields at the top of the column are those changed most frequently between rainfall events.

The **Prev Rainfall** and **Next Rainfall** buttons select the rainfall events up to the number specified in **Num rainfall events**. If you wish to add or decrease the number of events analysed change the **Num rainfall events** value.

CAUTION: the **Finish** button does not perform a save so make sure you click **Write** before **Finish.**

The fields and buttons used in this panel are described in the ILSAX users manual.

23.13.4 PCdrain to 12d pit converter

Position of option on menu: Design => Drainage Sewer => More=>Create Pit type from PC Drain gully files

This option is used to read a PCdrain gully file and create the same pit type in the 12d drainage.4d file. The sag pits in the gully file have an "S" added as a suffix as they are imported.

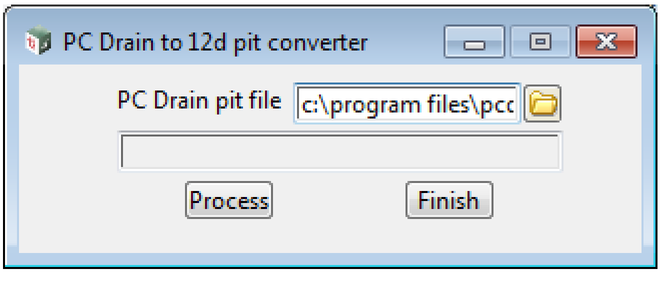
Important: 12d must be restarted to see the new pit types.

See Also

Drainage overview

Usage

This panel is accessed from the menu selection **Design => Drainage Sewer => More=>Create Pit type from PC Drain gully files**



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
PCdrain pit file	file box		

the PCdrain gully file to be imported into 12d

Process	button		

import the data file

Finish	button		

removes the dialogue from the screen

23.13.5 Calc pit overflow areas

Position of option on menu: Design => Drainage Sewer => More=>Calc pit overflow areas

This option is used to graphically display the overflow storage volume at a sag pit. The following pit attributes must exist for the flood extents to be calculated.

overflow volume	value greater than zero required.
sag pit	must be equal to 1.
catchment model id	set by labelling catchments
catchment string id	set by labelling catchments

The maximum storage volume is read from the drainage pit attribute "overflow volume". This may be entered manually using the **Attribute Editor** or it will be created when data is read from the drainage design programs Drains or XP SWMM design programs.

This routine locates the lowest point on the catchment string by draping the string on the tin specified and adds the overflow limit specified to this value. This becomes the **overflow limit.**

The volume at this level is calculated and the compared to the **overflow volume** read from the user defined attribute.If the overflow volume is less than the volume in the catchment then the routine iterates to find the flood level for the overflow volume.

If the overflow volume is greater than the volume in the catchment, the results depend on the **Use overflow limit** tick box.

If the box is selected, the **overflow limit** (calculated above) is reported at the flood level in the catchment.

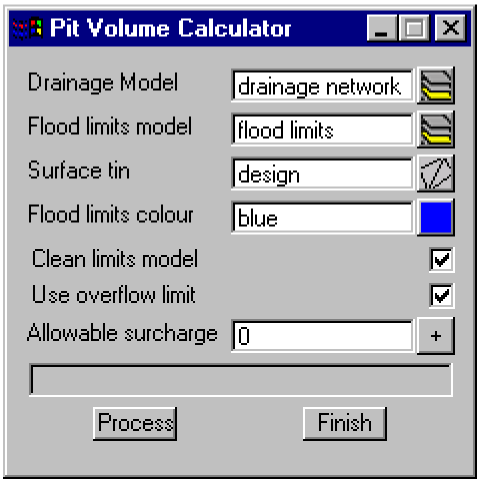
If the tick box is not selected the routine iterates to find the flood level where the storage equals the **overflow volume** read. This option allows the user to see the maximum flood level should the catchment low point become blocked.

See Also

Drainage overview

Usage

This panel is accessed from the menu selection **Design => Drainage Sewer => More=>Calc pit overflow areas**



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Drainage model	model box		

all pits in this model that have a non zero "overflow volume" and "sag pit" set to 1 will be processed

Flood limits model model box

flood limits strings will be created in this model

Surface Tin tin box

ground surface tin used to calculate the volumes and flood limits

Flood limits colour colour box

flood limits strings will be created using this colour

Clean limits model tick box

*if selected all strings in the **Flood limits model** will be deleted before the calculations commence.*

Use overflow limit tick box

*if the elevation calculated from the storage volume is higher than the lowest point on the catchment string then the **allowable surcharge** value below will be added to the lowest point on the catchment string and this elevation will be used to determine the flooding limits*

Allowable surcharge real box

*this value is used only if **Use overflow limit** is ticked. Its purpose is described in the field above.*

Process button

executes the option.

Finish button

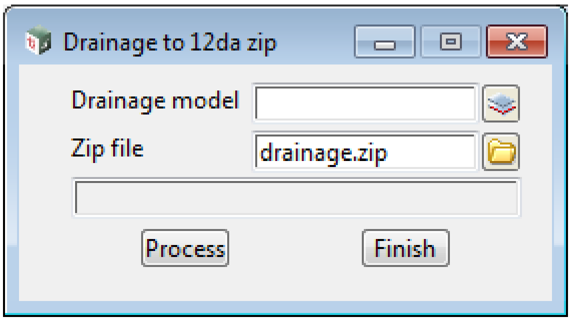
removes the dialogue from the screen

23.13.6 Drainage to Zip

Position of option on menu: **Design =>Drainage-Sewer => More=> Drainage to zip**

This zip routine is intended for support purposes only. It will save your project! This function zips up project views and only the models and files delated to the drainage model.

On selecting the **Drainage to zip** option, the **Drainage to Zip** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Drainage model	model box		

This model and the related models through the drainage network editor will be zipped into one file.

Zip file	file box	drainage.zip	

This zip file is intended to be sent to support personnel.

Process	button		

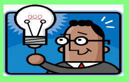
Saves the project and creates the zip file

Finish	button		

remove the panel from the screen

1.12d Model Drainage Modules and Work Flow

12d Model has a suite of drainage modules for stormwater design. This course covers the items listed below on this page except 24,25 and 28.



If viewing the pdf version of this manual, bookmarks on the left are hyperlinks as well as the green text. **Use Alt+left arrow** to return to where you were reading.

1.1 Drainage Module

Part A - Locate Structures and Connect the Pipe Network

1. Obtain/create Survey data and design surfaces (TINs) for pipe and manhole levels,
2. Identify overland flow paths and drainage inlet structure locations,
3. Set drainage default for the pits and pipes (most common settings, they may be changed later).
4. Read a drainage model template file containing you default and global settings.
5. Create the drainage strings in 12d or convert strings from CAD.
6. Assign pit names using the DNE. This allows easy reference to the structures.
7. Change pit and pipe types where they are different from the default values
8. Link the drainage model to the road design models for an integrated design.
9. Check and adjust horizontal alignment of the pipes and pits.
10. Set the vertical alignment using the DNE (set pit details and regrade pipes).
11. Check for service clashes and adjust vertical as required.

Part B - Hydrology and Hydraulic Data Preparation

12. Set catchment areas, % impervious, C and tc values for hydrology,
13. Set grate levels, Ku methods (pit and culverts), pipe roughness etc for hydraulic design
14. Draw/create bypass flow paths if bypass/inlet capacity calcs are required.
15. user defined export and update interfaces with Drains, PCdrain, Micro drainage-WinDes, RAT2000 and XP SWMM/STORM.

Part C - Design Documentation

16. Print pit schedules and plan/long section drawings for construction,
17. Import and Export the network to spreadsheets,
18. Calculate network quantities (by depth ranges) for costing and system checks,
19. Calculate earthwork excavation volumes,
20. Repeat as required for design changes,
21. And of course peer review at various points of design,
22. Electronic models for survey setout.

1.2 Drainage Analysis Module

Part D - Rational Hydrology, Surface Flow Calculations and Pipe Hydraulics

23. Calculate rational hydrology peak discharges,
24. Normal depth surface flow hydraulics using channel sections cut from the ground surface. Results are shown in plan, section and 3d views (and reports) with warnings given for exceeding max widths, channel capacity and velocity x depth hazard conditions,
25. inlet capacities at sag and on grade locations determine bypass and pipe flows,

26. Hgl calculations use pipe and manhole losses (Ku/Kw) . Pipe are sized using either freeboard or flow depth methods. This includes full culvert design with backwater and inlet control calculations.
27. Review results in plan drawings, long sections and printed reports,
28. Analysis of the major flood event and checks for surface flow and hazard conditions.

1.3 Dynamic Drainage Module - Unsteady flow analysis

This module uses ILSAX and SCS hydrology together with the solution of the unsteady flow equations to analyse the network. Inlet capacity charts are used to analyse surface and subsurface hgl lines. Natural channel shapes and elevation-area curves for basin are created directly from the ground surface. User defined reports are created via the spreadsheet interface.

1.4 TUFLOW and Roadflow Modules

These modules use the TUFLOW engine (1D/2D hydrodynamic computational engine) for simulating free-surface water flow for urban waterways, rivers, floodplains, estuaries and coastlines. Using tools including 12d grid tins and the 12d TUFLOW control file editor the 12d, TUFLOW input data is created, edited and viewed all within 12d. This module may be combined with the dynamic drainage module for a coupled 1D/2D analysis to include pipes and culverts.

2.12d Model Stormwater Courses

In these documents, the generic term **pit** refers to manholes, inlets and catch basins. When the term **manhole** is used on the 12d menu system it refers to any type of pit.

2.1 Stormwater Design Part 1

These notes are for this course designed to cover creating drainage networks, rational method design and reporting.

- create a super tin for pipe cover and pit cover levels,
- set Defaults and layout a drainage network from CAD and in 12d,
- use the 12d Drainage Network Editor to assign names to the pit/pipes, avoid service clashes, grade pipes, align obverts, minimise depth and many other design tools,
- designate catchment areas and produce catchment plans,
- run the 12d storm rational hydrology and hydraulics engine,
- transfer data to and from electronic spreadsheets to enable the user to easily review the data and add user defined data to the 12d pipe network. This data may include such data as pipe bedding types and trench width,
- create a drainage template containing customised default design parameters,
- create pit setout schedules to export to spreadsheets or word processors for final formatting,
- produce long section drainage profiles including HGL data, flows, invert levels and service crossings,
- create plan drawings with pipe sizes, flows, pit symbols, linestyles for pipe sizes, design parameters for pit and pipes and user defined data,
- locate pits/manholes at exact chainage and offset locations.

2.2 Stormwater Design - Part2

This course continues on with surface flow analysis, inlet capacity and bypass flow and customising the 12d drainage setup file (pit and pipe type database).

- customising the drainage.4d file Drainage Definitions - Manholes and Pipes,
- 12d storm analysis with inlet capacity calculations and bypass flow,
- flooded width analysis and flooding at SAG pits,
- drainage trench excavation volume calculations,
- pipe and pit quantity calculations/reports,
- open channel calculations,
- adjusting pit locations for changes in horiz road geometry
- analysing the major flood events,
- creating drainage symbols with grates and upstream side inlets,
- detailed drainage plan labelling and long sections with hatching under roads.

2.3 Stormwater Basin Design Hydraulics and Hydrology

- The course discusses unsteady flow hydraulics and their use in the design of storage basins and tanks.

3.Using the Course Notes

Areas marked in yellow (grey for black and white prints) are the exact steps required to complete the tasks. The numbers in the panels are the order to preform the tasks and they are always written to the right of the button, drop down or data field.

The additional text explains in more detail the tasks you are preforming.

4.Customising12d Model Drainage

12d Drainage has a series of files that customise your drainage components and design process. These includes

- drainage.4d (required) — manhole types and properties
pipe types and properties
- drainage model template — contains global and default settings for the drainage network. Including references to the following files.
- cover files — specify pipe cover for each pipe type
- grade file — specify pipe/channel slope by height
- drop file — specify drop across manholes by deflection angle
- catchment file — specify polygons for 3 land use types
- road design file — link manholes to strings to determine x,y z coordinates
- services file — specify clearance tolerances for models of service crossing.

The drainage.4d file has been configured for the training version. However, when you start working on your projects you may want customise the drainage module. More - Customising the drainage module

The drainage.4d file is the required drainage setup file that contains:

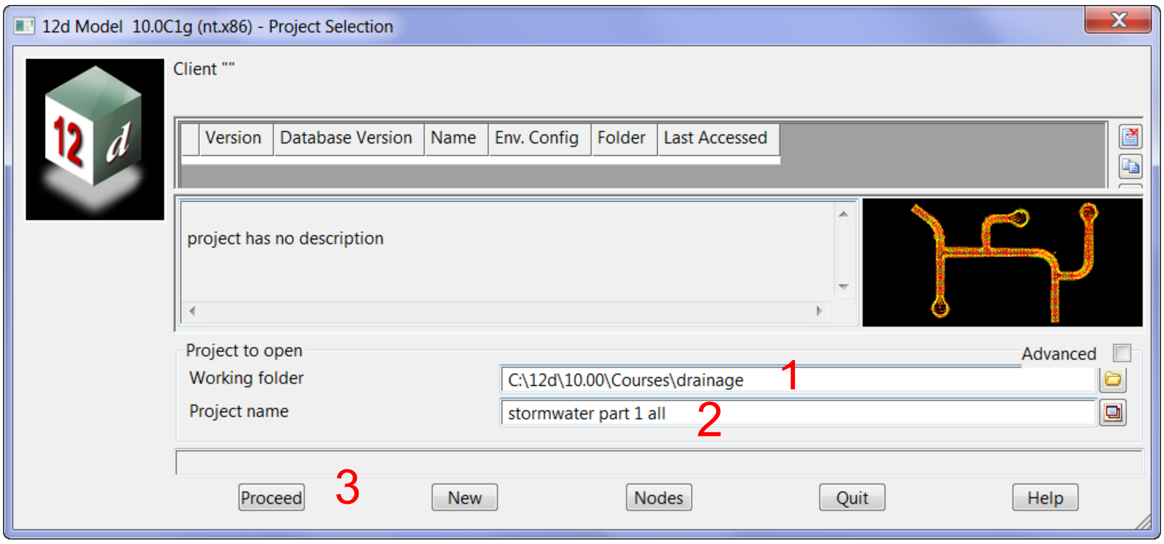
- manhole and inlets with wall thickness and inlet capacity curves for ongrade and sag conditions
- pipe types (RCP, Class 2 etc.) with nominal pipe diameters and wall thicknesses,
- user defined attributes lists to be assigned to the network via pit and pipe types.
- manhole and pipe type parameters are used to control settings in the DNE. This allows the user to set numerous properties by selecting only the pit or pipe type. More.

5.Survey data and design surfaces (TINs)

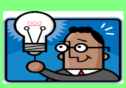
We will start a 12d project from the beginning by first creating the project and then reading in the survey and design data. The data can then be triangulated so that we have a final surface to measure pipe cover from and set structure cover levels to.

Start up the 12d model by selecting the 12d icon from your desktop.

The project selection panel will appear. The bottom corner of the panel is shown below



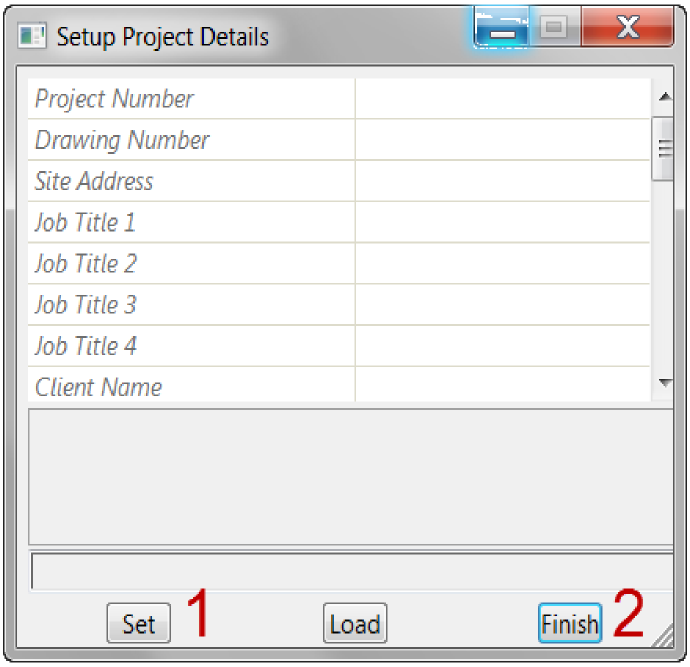
1) **LB** to browse to the **working folder** indicated.
2) Type a project name for your work **Stormwater Part 1 all** for example.
3) Select **Proceed.**



Alternate Step 2. If you do not want to create the tin or roadway then select the existing project Stormwater Part1 found in the courses\drainage folder. Skip to the section Drainage Overland Flow Investigation or go straight to Read in a drainage model template

12d will first look in your working folder for any input data files and output files will be created here unless you specify another folder. 12d stores it's data in a folder **not** a single file. Therefore, a folder named **Stormwater Part 1 all** will be created when you select **Proceed**.

12d model will load and you will see the following panel.



Enter data as desired. This data is both for reference and can be included on your drawings.

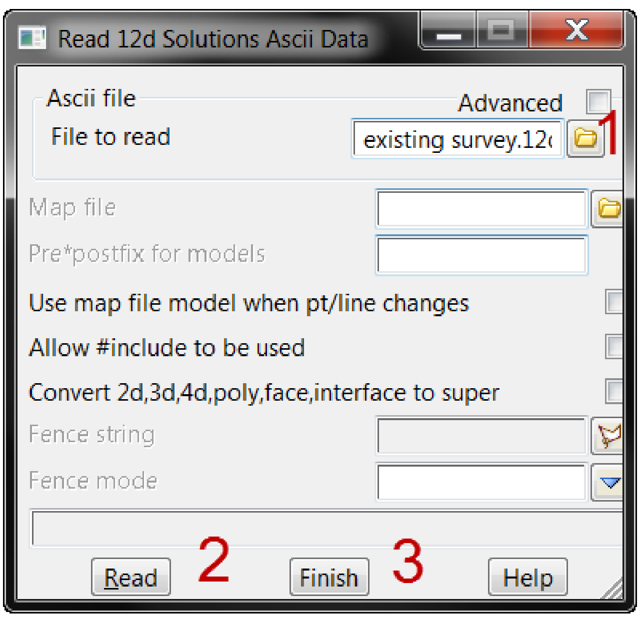
Later these details can be changed via

Project=>Management=Details Editor

5.1 Importing the Raw Survey Data

You have created a new project into which we will import the survey data. From the main menu select.

File I/O->Data Input->12da/4da data

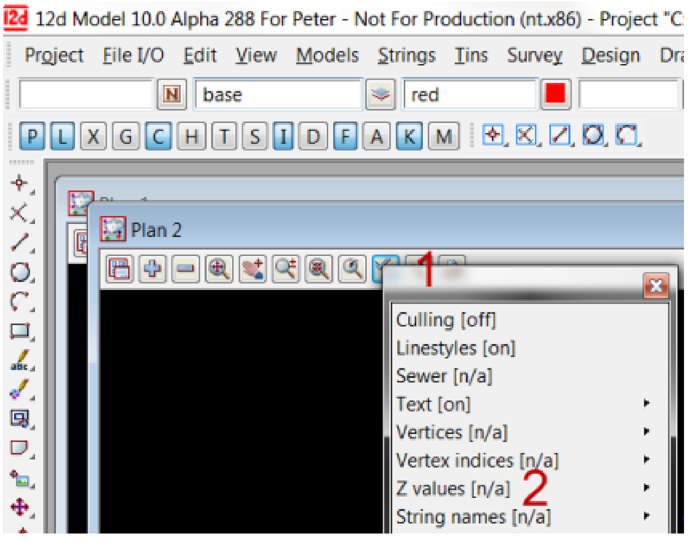


1) LB the folder icon to display the file list. Select the file **existing survey.**

2) LB Read to read the data

3) **LB Finish** to remove the panel

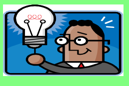
The data will appear on a new plan view.



Now we will display the z values for the survey data.

1) LB the toggle button

2) LB **Z values**

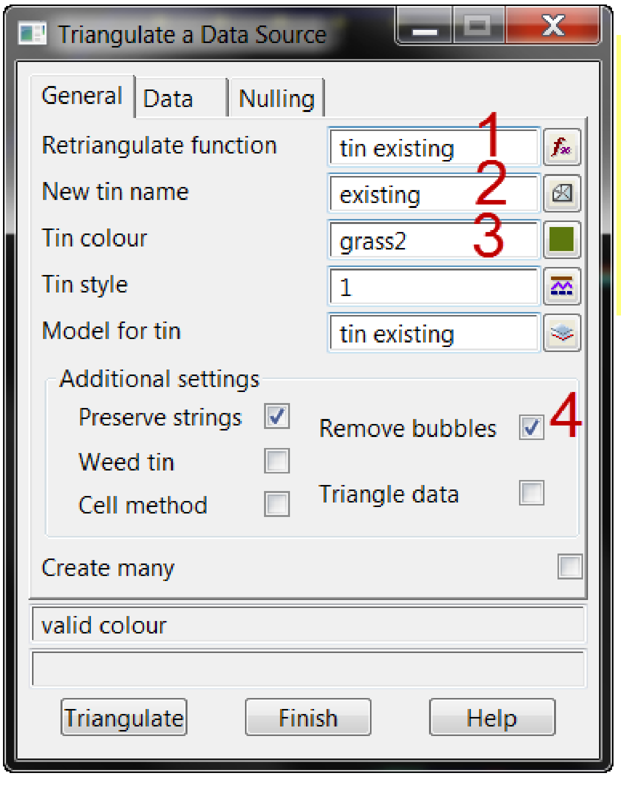


Use your mouse wheel to zoom in and read the z values.
Press Ctrl and Middle mouse drag to pan

5.2 Creating the existing Ground Surface

From the main menu select

Tins=>Create=> Triangulate data



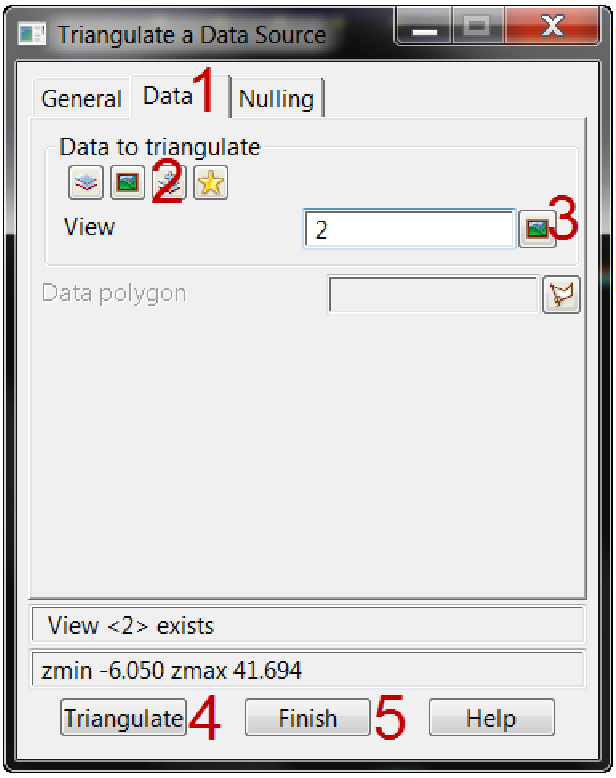
1) Type a name for the function to make it easy to recalculate later in new data arrives.
2) Type a new tin name then press **Enter**
3) Type grass2 instead of green. This will look better in the visualisation.
4) LB **Remove Bubbles** so that breaklines will attempt not to form triangles back onto themselves (if the breakline is a contour, this removes contour bubbles).

Preserve strings will make one side of a triangle follow the string thereby preserving the levels along the string.

Weed tin removes all duplicate points from the tin database.

The **Cell method** is a good speed enhancement for data that is in a grid type pattern.

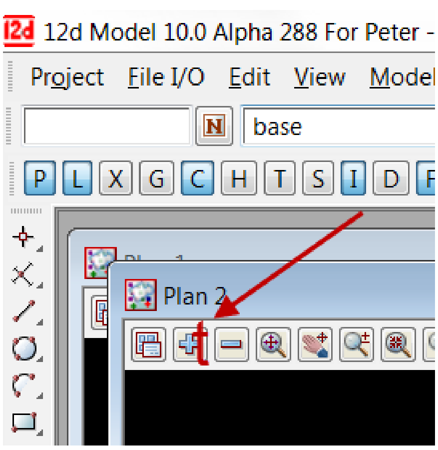
Triangle data is used for string that form triangles (tins imported from other programs).



1) LB the **Data** tab
2) LFinishB the **View** icon
3) Select view **2**
4) LB **Triangulate** to create the new tin.
5) **LB**

The tin will be stored in the model **tin existing**. A tin, like all 12d strings, must be stored in a model. 12d prefixes all its tins with **tin** for data management.

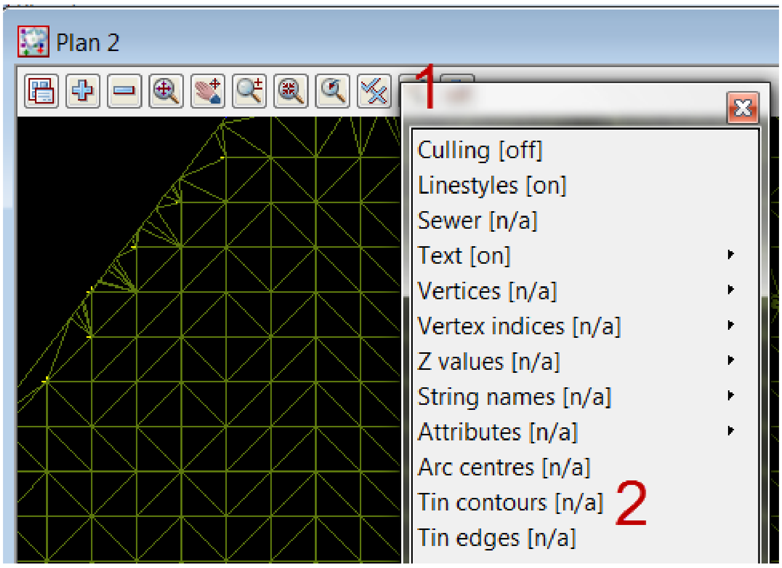
5.3 Viewing the Ground Surface Tin



1) Move your pointer over the "+" sign on the plan view 2 title toolbar
2) press the **t** key (all of the models beginning with lower case t will appear)

3) Double click on **tin existing**
4) **LB** Finish

The tin will be shown with the tin edges turned on. This is the default when you have all tin display modes turned off.



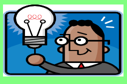
Display Contours

1) LB the toggle button
2) LB **Tin contours**.

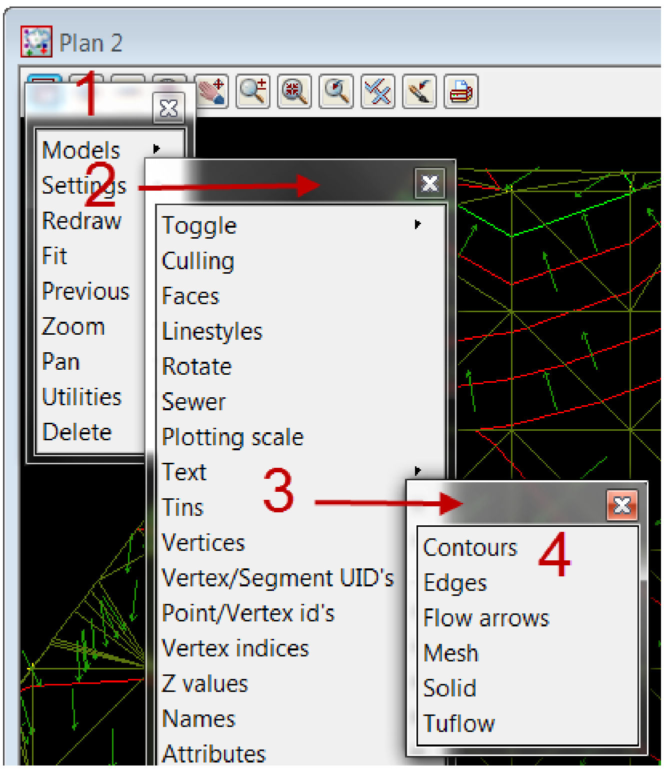
The contours are displayed with their default colours and increment (1 unit).

Display Flow

1. LB the toggle button
2. LB **Tin flow**.
   Flow arrows are now visible at the centre of every triangle.



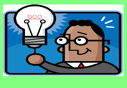
To plot flow arrows, create them in a model using
Tins->Tin analysis->Flow arrows



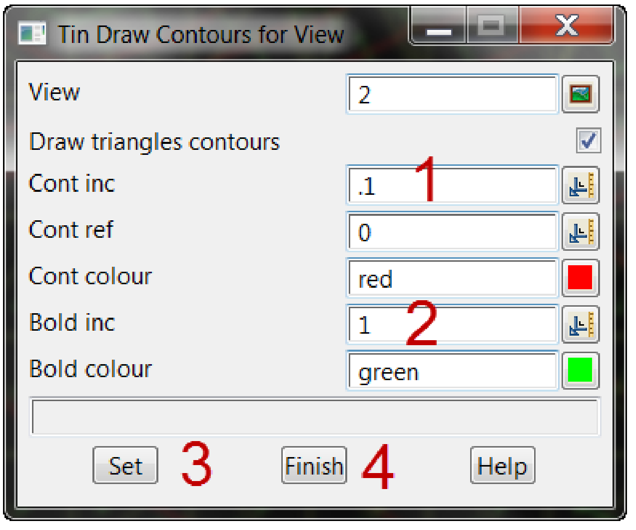
Change Contour Intervals and Colours

1) LB the View menu button
2) Walk right (do not **LB** select) on settings.
3) Walk right on **Tins.**
4) LB **Contours**

4) LB **Contours**



ShortCut! Press F11 to display the Menu shown in step 4

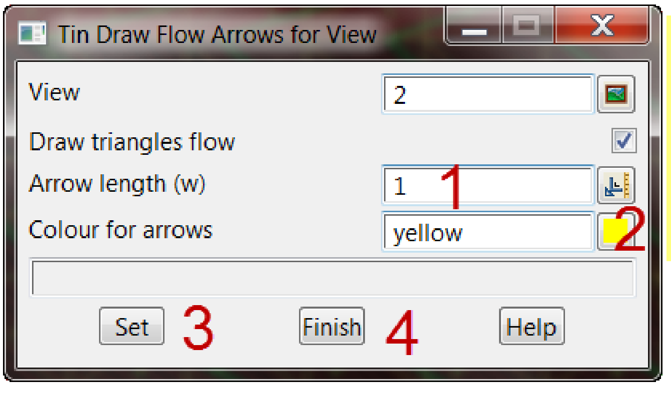


1) Type a new **Cont inc**
2) Type a new **Bold inc**.
3) LB **Set** to see the new contours**.**
4) LB **Finish** when done.

Change the Flow Arrow Length and Colour

Select the **Menu** button on the **plan view tool bar** and then walk right on.

Settings=>Tins=>Flow Arrows and the following dialog will appear. Remember to walk right!

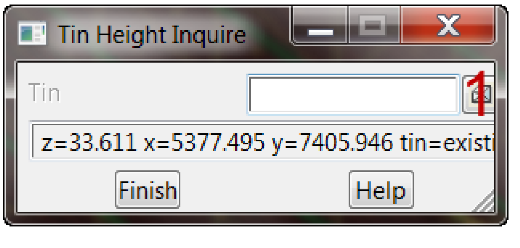


1) Type a new **Arrow length**
2) LB the colour icon to select a new **Colour for arrows.**
3) LB **Set** to see the new flow arrows**.**
4) LB **Finish** when done.

5.4 Inquiring about Heights on the Surface

The elevation anywhere on the tin can be obtained by simply moving the pointer over the desired spot. To obtain the tin elevations select,

Tins=>Inquire=>Height



If you do not select a tin, the last tin added to the view will have its height displayed. You will see the data in the panel change as you move the pointer around the screen. You may explicitly select a tin if another tin is desired.

Move the panel to the bottom right corner of the screen and leave it there.

5.5 Viewing the Surface Tin in a 3d Perspective

To create a 3d perspective view select

View=>New=>Perspective Open GL

1) Add the model **tin existing** to the view.

2) **Toggle** the **contours** on.

3) **Toggle** the **shading** on (note that the contours are no longer seen).

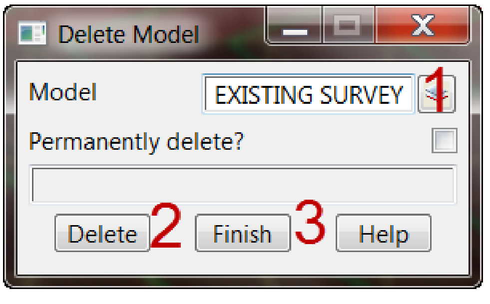
4) Use the Orbit control (planet on the toolbar) and zoom wheel to move around the view.

5.6 Reducing the number of points for the 12d Practice Version

We can delete the survey data to save space for those using the practice version of 12d. If you have a full version of 12d this is not required. From the main menu select.

Models=>Delete=>Delete a Model

and the following dialog will appear.



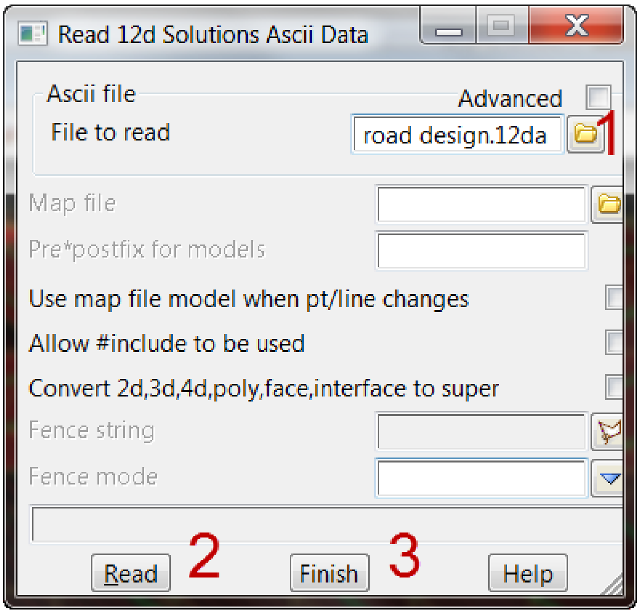
1) LB the model icon and select **EXISTING SURVEY**
2) LB **Delete** and then confirm each panel
3) LB **Finish**

The model is now in the 12d trash can (see bottom right corner of the 12d screen).

5.7 Importing the Road Design Centre lines

Repeat the process of importing 12da data.

File I/O->Data Input->12da/4da data

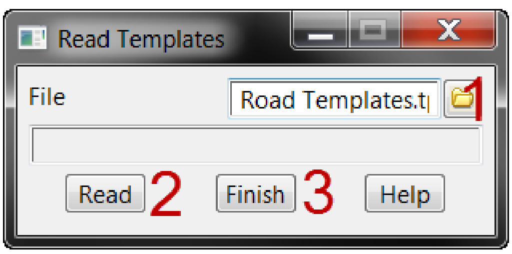


1) LB folder icon and select **road design**
For 12d practise users select **road design practice size use 10m spacing.12da**
2) LB **Read** (the data will not appear on the view yet).
3) LB **Finish**

5.8 Using Create Roads

We are going to create the roadways from the road centre lines (vertical grading has already been done). First we will need the road templates that we will read from a templates file.

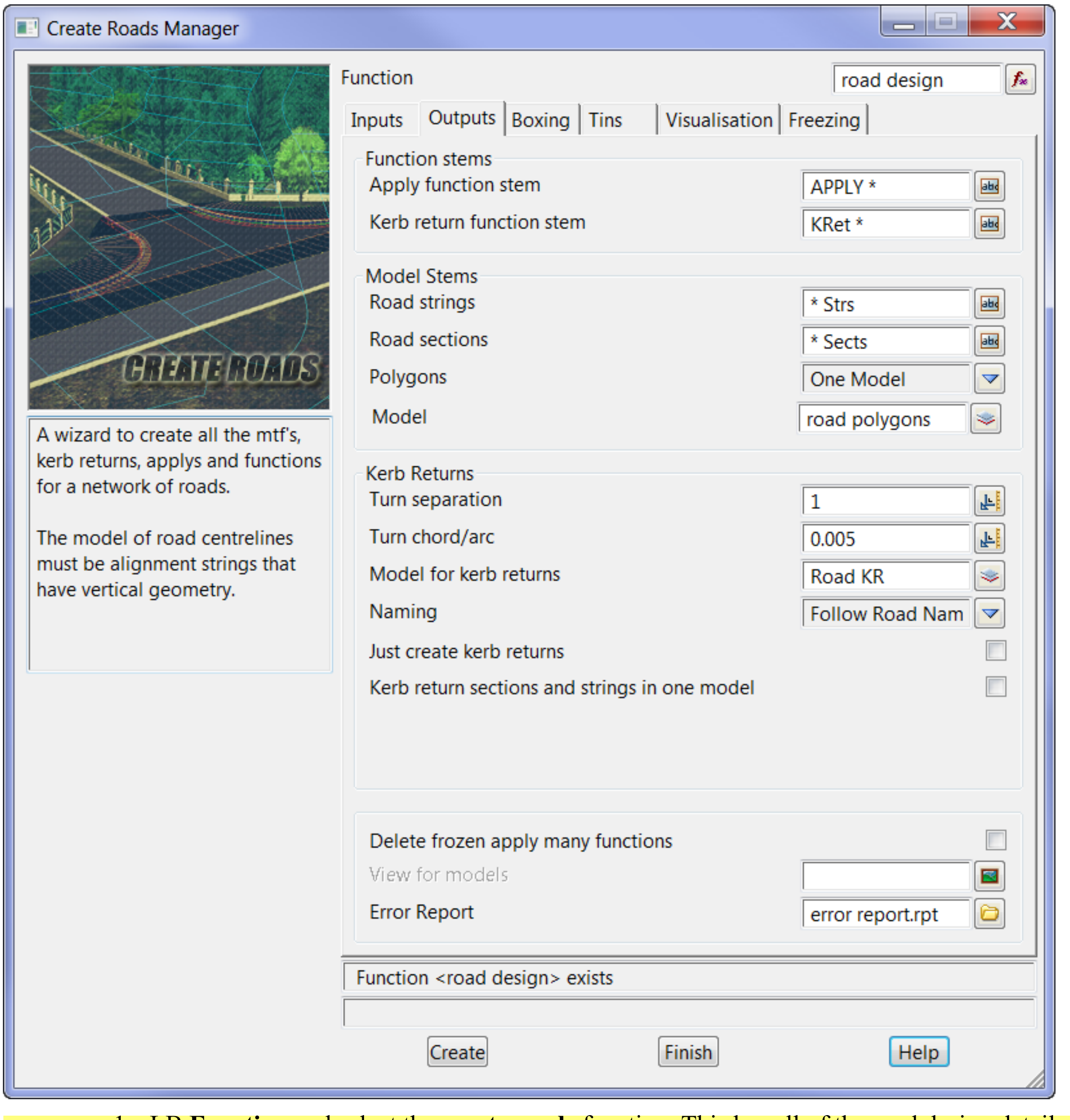
File IO=>Templates input



1) LB folder icon and select **lib** and select **Road templates**
2) LB **Read**
3) LB **Finish**

Now we are ready to create the road strings and design tin.

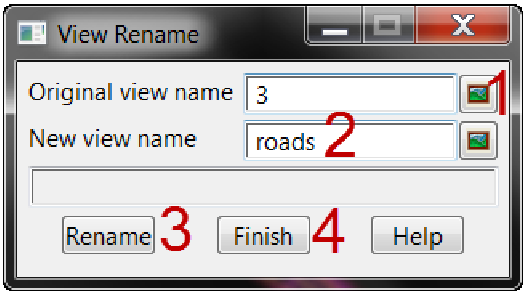
Design=>Roads=>Create=>Create Roads - Manager



1. LB **Function** and select the **create roads** function. This has all of the road design details included in it.
2. LB the **Create** button
3. LB the **Finish** button.
4. Add **tin design** onto the plan view.

To view the road strings, add on all of the **road *** models to view **3**. To view the tin add on the **tin design** model.You can also add the **tin design** model onto the perspective view. **Important:** On the OpenGL perspective view, 12d will show the tin data with the greatest z values (when viewed from the top side).

Rename view 3 to roads to help organise your work.

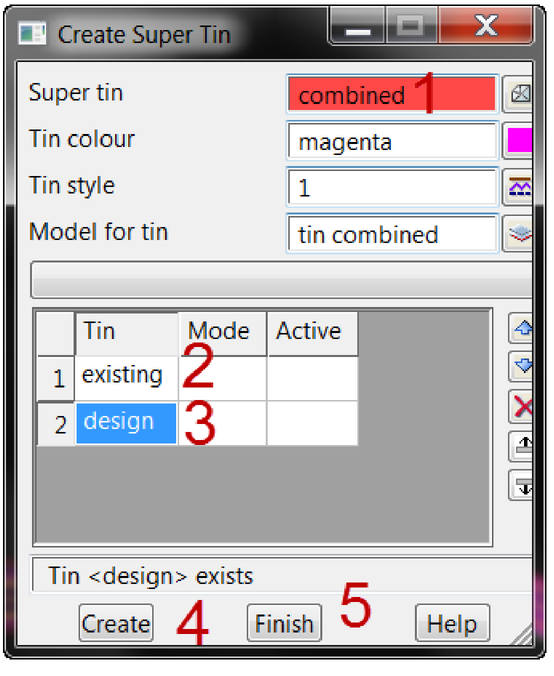


1. LB the **View** icon to select view **3**
2. Type the new name **roads**
3. LB the **Rename** button
4. LB the **Finish** button.

5.9 Creating a Super Tin from the Survey and Design Data.

The drainage fs tin may be used to set pipe inverts, pit levels, sag catchment overflow points and catchment slopes for tc strings. Therefore, the tin often needs to include several tins combined into one tin. To create a tin that is the combination of the survey data and the road design data you will need to create a 12d super tin. From the main menu

Tins=>Create=>supertin



1. Type a **New tin name** of **combined.** Press **Enter!**
2. RB in the cell and select the **existing** tin.
3. RB in the cell and select the **design** tin.
4. LB the **Create** button.
5. LB the **Finish** button.
6. Add **tin combined** onto the plan view.

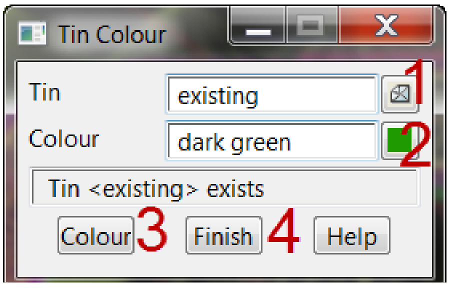
Notes about super tins:

- tin colour is only used in section views,
- the super tin is the 2 tins "glued" together,
- the super tin is Opaque so it will "cover up" other models already on the view. Use the Menu button on the View toolstrip and select **Models->models to back** and select the super tin.

- **The order of the tins is very important.** Tin 1 is the first to be drawn then tin 2 is drawn. i.e. wherever tin 2 exists it will be used. If there is no tin 2 at a location then tin 1 will be used.

5.10 Changing the Colour of a Tin

To emphasise the super tin is just the tins "glued" together, we will change the colour of the **existing** tin and see how the super tin changes colour as well.



Tins=>Colour=>Colour of tin

1. LB tin icon and select the **existing** tin
2. LB colour icon and select **dark green**
3. LB **Colour**
4. LB **Finish**

Toggle the **Tin solid** on and note the colours of the super tin.

6.Drainage Overland Flow Investigation

Where a drainage designer chooses to start their design is depends a great deal on the project and the designer. Identifying overland flow routes, with the crest and sag points, is essential because it is on these routes that the inlets are to be placed. Inlet structures are then placed at the critical locations (sag points, flattening of grades, intersections, upstream of pedestrian crossings etc.). and then the spacing of additional inlets is determined by the size of the catchments. Finally, the pipe drainage system can be created linking the drainage structures.

Following is a list of 12d tools to identify surface flow routes:

Change the Flow Arrow Length and Colour

Downhill Strings

Locating Crests and Sag Points

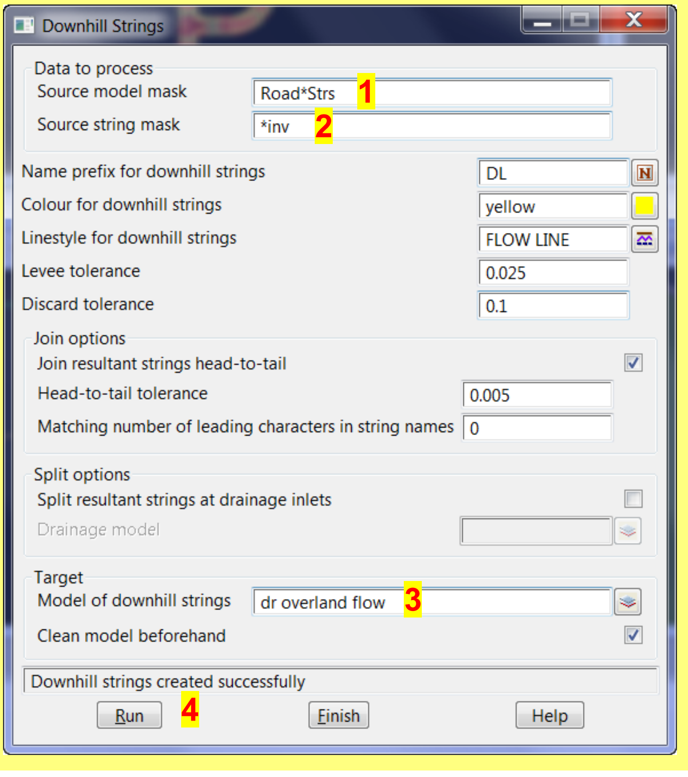
Rain drop

6.1 Downhill Strings

Roads are generally designed with a flow path in the cross section shape. In this example the roadway has a dual crossfall with kerb and gutters. We will use the downhill strings option to locate all the inverts strings, change their line style to a flow arrow, split them at their crest and sag points and ensure the chainage increases in the downhill direction.

From the main menu select

Design->Drainage-Sewer->Downhill strings

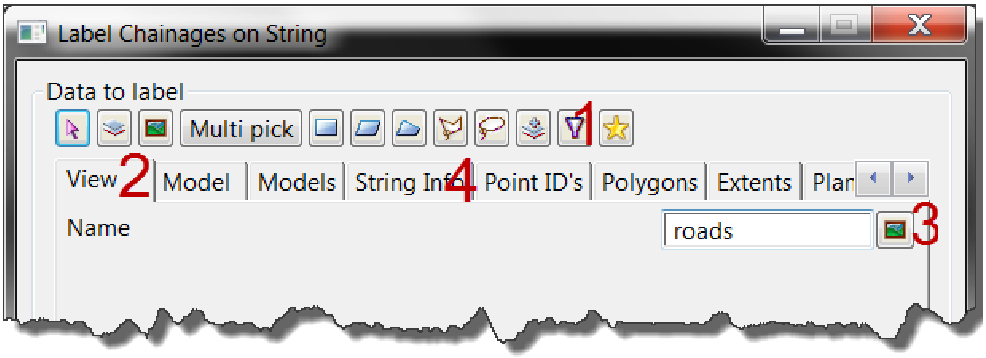


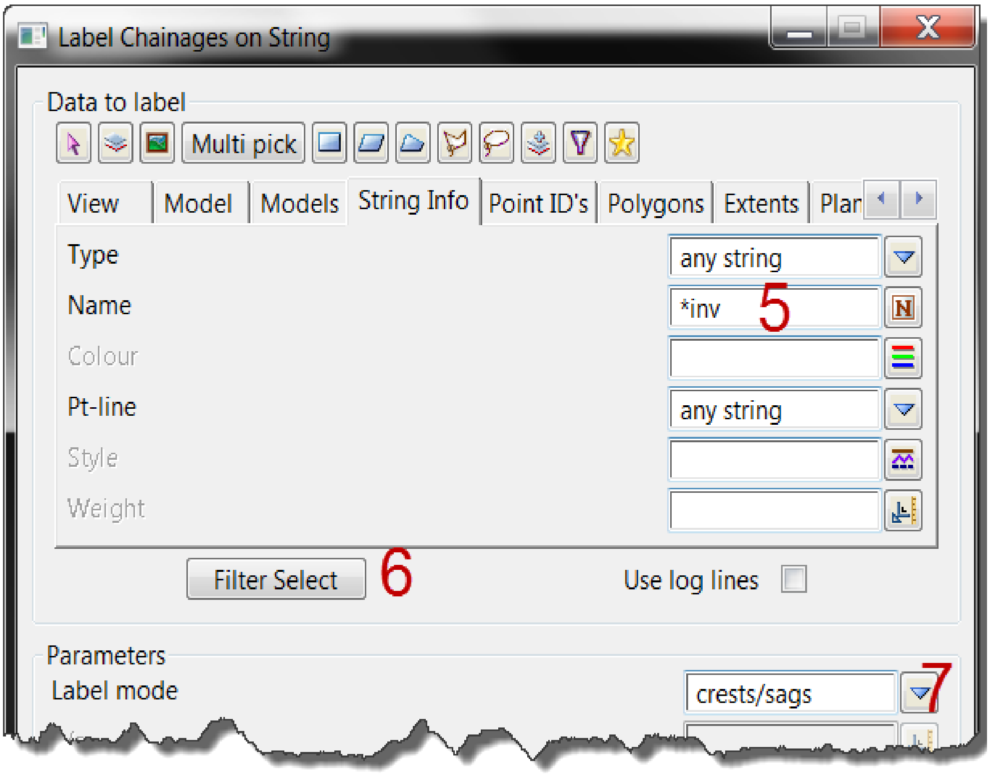
1. type **Road*Strs** (it is case sensitive). These are the models that contain all of the road strings. **Road 1 Strs** for example.
2. type ***inv**. The road string models have a **linv** and **rinv** string for the invert of the kerb and gutters.
3. type **dr overland flow** to create a model for the new strings
4. Select **Run**
5. add the model **dr overland flow** onto the **roads** view.

6.2 Locating Crests and Sag Points

This step will place tick marks at the crest and sag points of your kerb strings and label them with the levels.

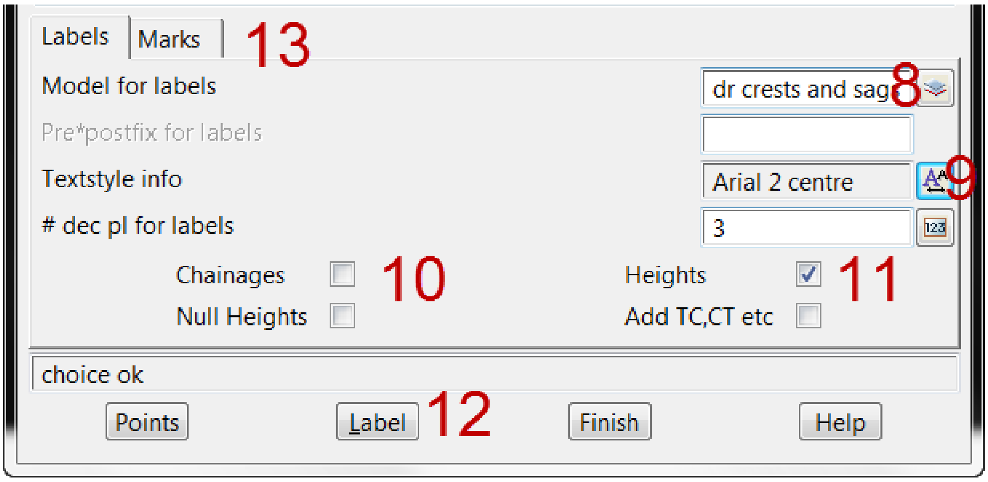
Strings =>Label => Chainages



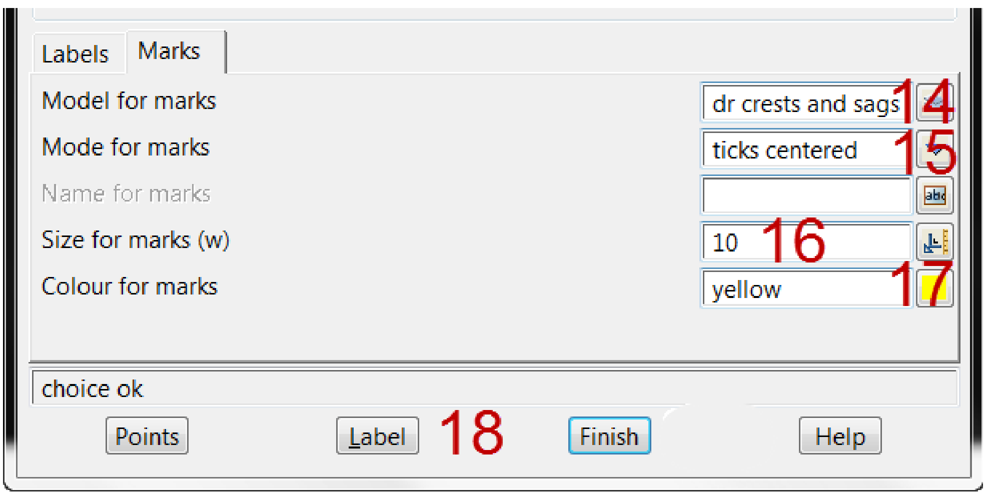


Add all of the models beginning with Road onto plan view 1.

1. Select the filter button
2. Select the **View** tab
3. Select the **Roads** view
4. Select the **String Info** tab
5. Type ***inv** to so that the linv and rinv strings will be selected.
6. Select **Filter Select** to select the inv strings to be labelled.
7. Select **crests/ sags** from drop down list**.**

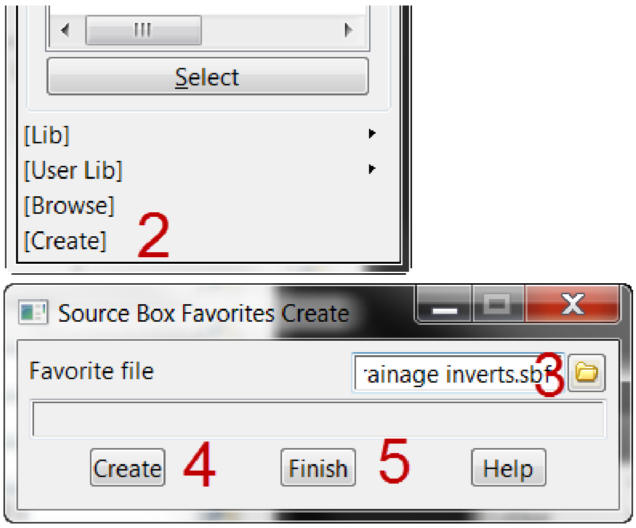


8. Enter a model to contain the text labels. A ,1 after the model name, requests that the model be added to view 1. This saves you adding the model to the view later to see the labels.
9. Select icon to define the text style.
10. Untick **Chainages**
11. Tick **Heights**
12. Select **Label**
13. Select **Marks** tab



14. Select the model used above for the tick marks.
15. **LB** to select **ticks centred** from the drop down list**.**
16. 10m marks stand out well on the screen.
17. Set colour to **yellow**
18. Select **Label** to create the labels.

6.3 Creating a Filter Favourite



1. Before closing the panels, create a filter favourite that can be used later. Select the Star button
2. Select **[Create]**
3. Type **drainage inverts**
4. Select **Create**
5. Select **Finish**. The favourite is now created for later use.

Select **Finish** to remove the panel.

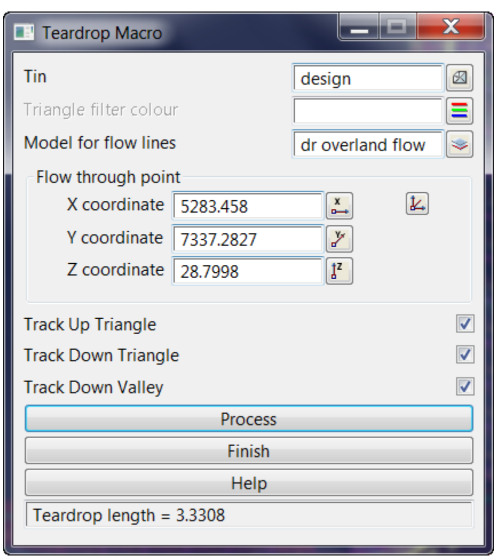
Now add the model **dr crests and sags** onto the **roads** view.

6.4 Rain drop

The raindrop routine will create a line from a selected point and follow either up or down the tin.

From the main menu select

Design->Drainage-Sewer->Rain drop



Try selecting several points on the tin to see the flow lines that are created.

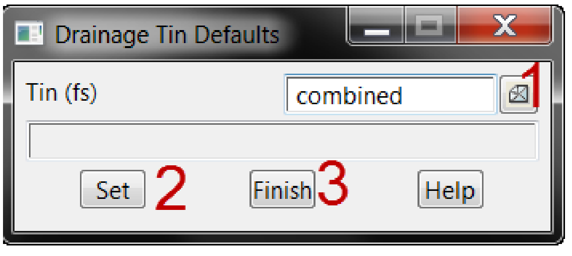
7.Setting Drainage Defaults (Initial Settings)

Before creating drainage strings in12d we should set default values for the pit and pipes. These defaults are only used as initial settings for new manholes and pipes. This ensures that most of the pit and pipes will have the desired type, size etc and then we only need to modify the pit and pipes that are different to the default values.The defaults for the drainage network are accessed through three panels; pipe defaults, manhole defaults and Tin defaults.

7.1 Tin Default

The 12d drainage module is designed so that no levels need to be manually entered into the drainage system (although they can be and locked if desired). The primary source for these levels is a design tin but levels from design strings and polygons can be used as well.

Design =>Drainage-Sewer =>Defaults=>Tin



The default TIN is used to set the initial manhole cover level and the pipe invert levels (via pipe cover or depth settings discussed later). Super tins may be used if you want to place manholes on both the existing and the design ground surface (see Creating a Super Tin from the Survey and Design Data.).

1. LB the tin icon and select the **combined** tin.
2. LB the **Set button**
3. LB the **Finish** button.

If you place a pit outside the tin boundary:

1. then no elevation will be set for the top of the pit, (it can be set later manually or by linking it to a road design string).
2. Pipe invert levels cannot be set using the default cover. Pipe invert levels must be set manually as 12d cannot automatically determine cover levels without a TIN.
3. Finally, if you are exporting to an external drainage design program that accepts surface levels along the string, then an error message will be displayed at export time. The message will say that the surface level string is shorter than the pipe length.

7.2 Manhole (Pit) Defaults

Design =>Drainage-Sewer =>Defaults=>Manholes

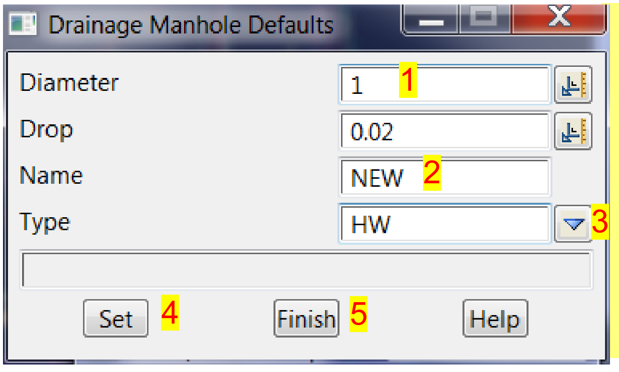
Notes on pit Diameters

The pit diameter/size is specified in metres/feet **not** mm/inches. The size of the pit is generally set in the drainage.4d file as either circular (mhdiam) or rectangular (mhsize). If both commands are specified then the mhsize command takes precedence. Plan drawings may use a symbol via the drainage plot annotations.

The **Diameter** is used for:

1. for visual service clash identification in long section drawings,

2. to clip the pipe lines drawn in the plan annotations so that symbols can be inserted in the space created,
3. maximum distance the bypass flow strings can be drawn from an inlet centre.



1. Type the desired diameter. If the manhole type below has a diameter/height defined, this setting is ignored.
2. Type the **Name** as **NEW** to remind you to assign a name latter.
3. LB the **Type** drop down and select the most common manhole type.
4. LB the **Set** button.
5. LB the **Finish** button.

The minimum **drop** will be used to set the invert level of the outlet pipe relative to the invert level of the inlet pipes. The drop should always be entered as a positive value. The DNE has many more options for aligning the pipe inverts at the pits. The drop may vary with the pipe deflection angle via the DNE**->Defaults->Pipe->Drop file**.

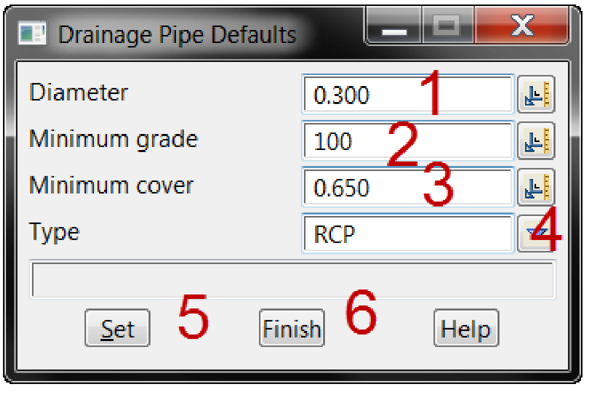
Setting the default name to **New** is a good reminder that you have not confirmed the pit type. Use the Set Pit Names function on the DNE to assign the pit names.

The **pit type** list is obtained from the drainage.4d file. Set this to the most common pit type and then later change the ones that are different using the DNE.

You **must** click the **Set** button to set the default values. Finish alone will **not** set the defaults.

7.3 Pipe Defaults

The pipe **diameter** is set in metres or feet **not** mm or inches. Set this to a common pipe diameter for your project then alter other pipes or change to box culverts in the DNE.



1. Type the common pipe diameter.
2. Type the minimum grade (1 in xxx). This value can changed by pipe height via the **DNE->Defaults->Pipe->Grade file**
3. Type the minimum cover. This value can be changed by pipe type via the **DNE->Defaults->Pipe->Cover file**
4. LB the **Type** drop down and select the most common pipe type.
5. LB the **Set** button.
6. LB the **Finish** button.

Pipe Thickness for Pipe Cover

When 12d set the pipe inverts it checks the **minimum cover** from the top of the pipe to the finished ground surface at the ends f the pipe. Pipe thickness is changed by pipe type as defined in the drainage.4d file.

Cover = surface level - thickness - diameter(height) - invert

If the grade of the pipe is less than the **minimum** grade, the grade of the pipe is increased. The DNE has a pipe grade file where the minimum grade may be changed via the pipe diameter.

Finally, 12d checks if there is anywhere along the pipe length that has less than the **minimum cover**. If there is such a low point in the design surface, the pipe is shifted vertically downwards to achieve the cover required. The DNE has a pipe cover file that will change the pipe cover via the pipe type. If the pipe type is not found in this file then this minimum cover will be used. Circular pipes have the cover checked along the centre line and box culverts along the centre line and edges.

When using the 12d editors to change the pipe diameters the invert levels will remain fixed and the obverts will change. The inverts may be reset using Regrade Network on the network editor.

The list of available pipe types is set in the drainage.4d file.

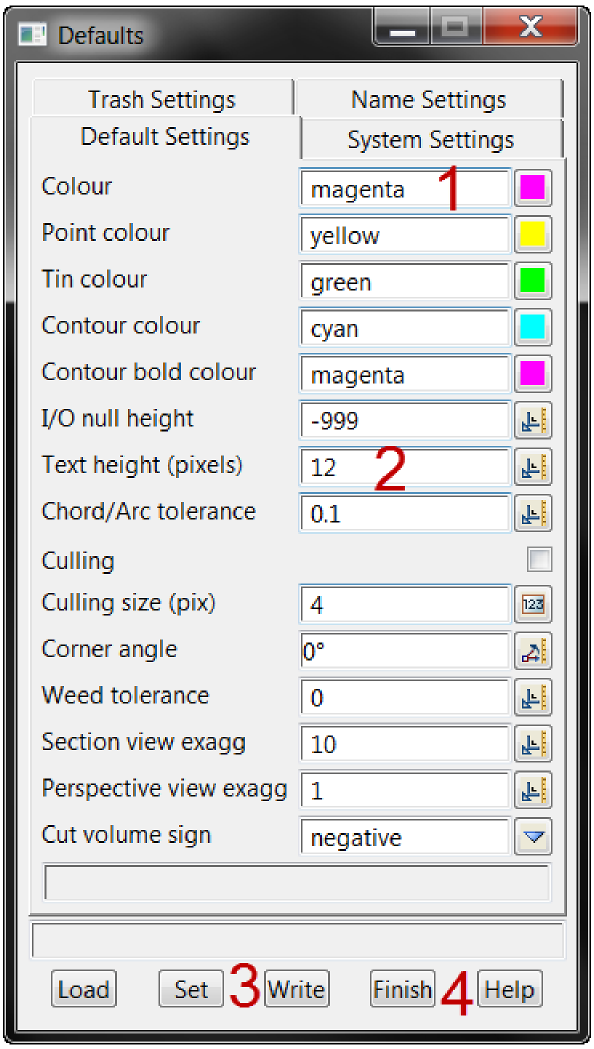
7.4 String Colour and pit Label Text Size/location (string defaults)

12d can automatically label the pits at a fixed offset from the pit using view text OR you can use the network editor Plot to create text labels that can be moved/rotated etc.

For view text, the default line colour and text size are set by selecting

Project=>Management =>Defaults

The following panel will appear.



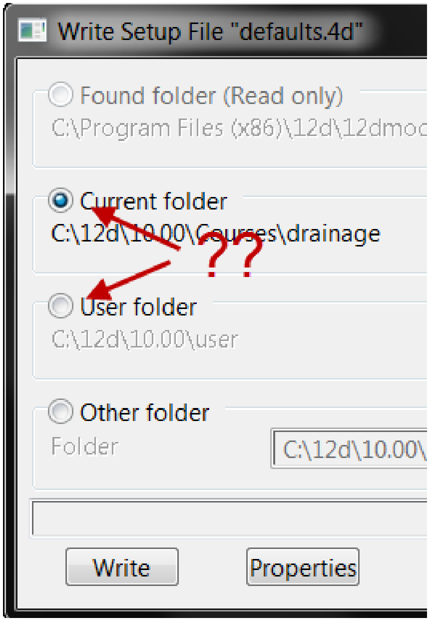
1. LB **colour** icon to set the default colour of the drainage string and manhole labels.
2. Type the height of the text in pixels
3. LB the **Set** button. This will set the defaults for this project only.
4. LB the **Finish** button.

The manhole label textstyle and offset may be changed via the DNE->Global->display tab

Optionally, you may LB the **Write** button to set the defaults for other projects. The following panel will appear. Selecting **Current folder** will save ALL these defaults for projects in this working folder only. The defaults set in the **user** or **setups** directories will not be used if you select this option.

Selecting **User folder** will save your defaults so that all other 12d projects will use these defaults. This is the most common option (unless your network administrator has not given you write access to this folder (check **Properties** in windows explorer).

Select **Write** then **Finish**



8.Creating Drainage Strings

We are now going to create a culvert to cross road 3.

Key Points

1. Before creating the drainage strings, read in a drainage model template (Design->Drainage-Sewer->Create->Create/Read template) so that all of DNE settings will set. There is a sample one in the 12d library (drainage_template_QLD.12da). If you forget, do it later but **READ A TEMPLATE FILE.** We will create one at the end of this course.
2. You will be placing centre of the culvert headwall. The headwall centre is the selection point for the pit. Pipes can be connected this pit Use setout strings for construction setout points!
3. Drawing all drainage lines in the same direction may help you keep your drainage work organised. Either draw all string uphill to downhill **(**flow **same as string direction) OR** downhill to uphill (flow **opposite to string direction**). You can mix the flow directions if desired.
4. The Drainage Network Editor **Regrade Pipes** is the key to setting pipe invert levels! Use this after modifying the drainage plan layout as it will update all initial grading described below.

   However, 12d will perform an initial grading as you draw the string. If you choose **opposite to string direction** then will need to select Pipe=>Default Grading then Grade to achieve this initial estimate. If you create branch lines before the trunk lines the initial grading will ensure the trunk line inverts remain below the branch line. Regardless, the DNE regrade pipes will sort the grading for all strings in the model.

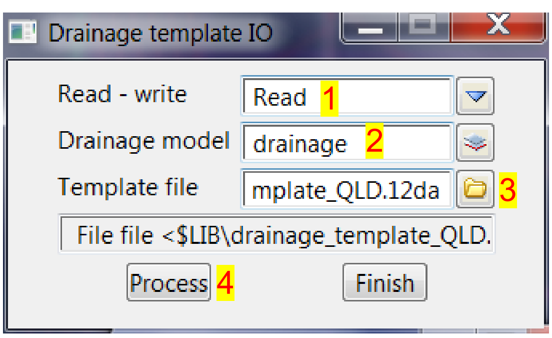
8.1 Read in a drainage model template

A drainage model template contains your favourite global and default settings for the DNE. These settings are stored as model attributes and the template may be read before or after the drainage strings have been created. Caution: existing default and global setting may be overridden.

You can create you own templates as well. After you have completed a drainage job and all of your global and default settings are set, create a template to save in your user library.

From the main menu select

Design =>Drainage-Sewer =>Create->Create/Read template



1. Leave as **Read**.
2. Type **drainage** to create a new model for the drainage strings
3. LB the folder icon, then lib to locate drainage_template_QLD.12da
4. LB **Process** to create tthe model and import the default and global settings.

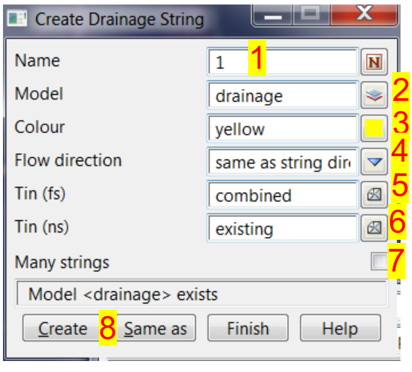
8.2 Creating the Drainage Strings in 12d

12d has a routine that creates a culvert very quickly. **Design->Drainage-Sewer->Create->Culvert**. It will also add a channel to the outlet as well! Try this, then the general method discussed below. The culvert is placed in a model **culvert** and the levels are all set to the manual mode.

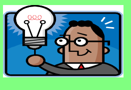
A drainage string is created by selecting

Design =>Drainage-Sewer =>Create->Create

The initial manhole and pipe types will those set in Setting Drainage Defaults (Initial Settings).



1. Type a string name (used for pit naming later).
2. Type a new model name for the network
3. LB the **Colour** icon and select colour (optional).
4. Leave unless drawing in uphill direction.
5. From the default, leave as is.
6. Existing tin can be used for long section dwg levels
7. Select if creating more than 1 string.
8. LB the **Create** button, the menu below will appear.

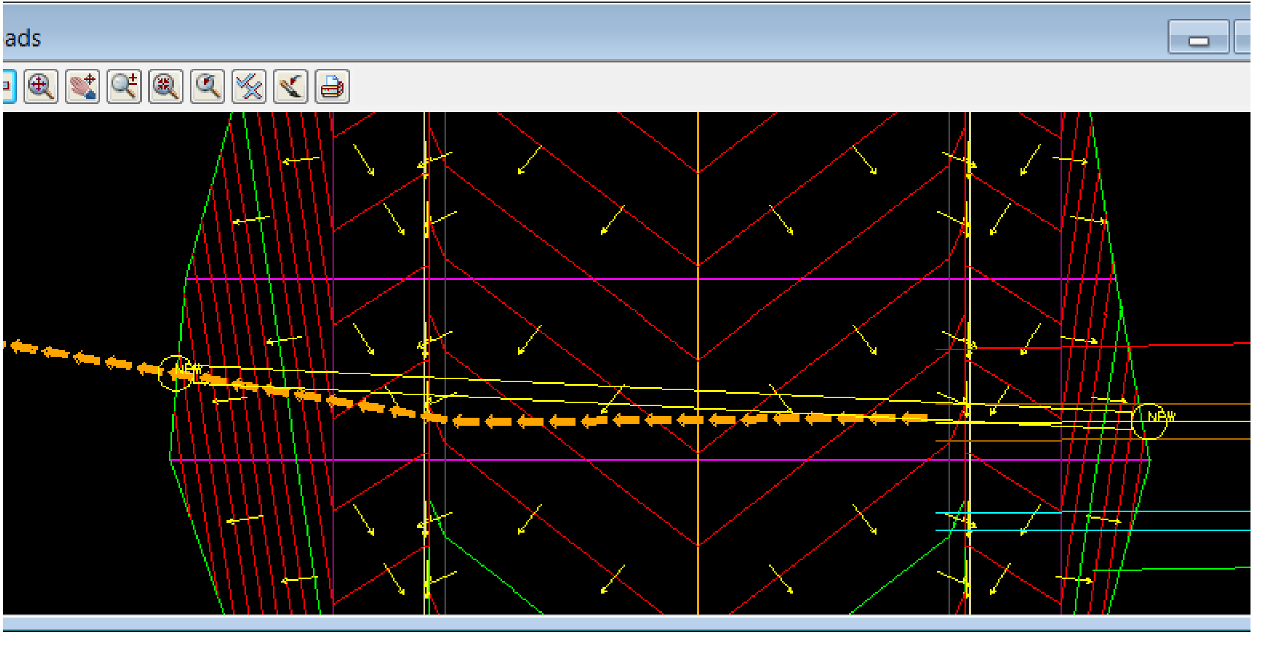


If you have already created a drainage string, click the **Same as** button and select the drainage string to obtain the panel values from that string. After selecting the string, change the string **Name.**

We are going to initially place our head walls on edge of the batters and then fine tune the location later.

To create your first manhole select

1. **Edits =>Add/Append MH**. A +MH will appear with your pointer. Refer to the plan below for the headwall locations.
2. LB and accept for the upstream (east) and downstream (west) headwalls.
3. LB the **Finish** button after the last headwall is placed.



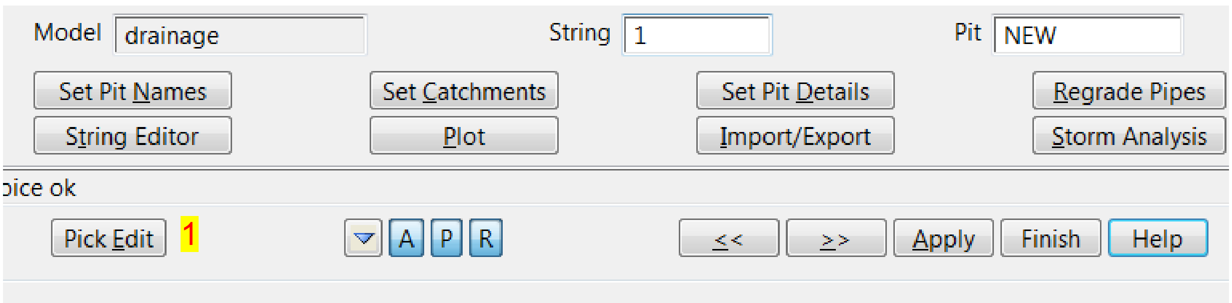
8.3 Change Pit and Pipe Types and Sizes

We will use the DNE to change the structures to inlet and outlet headwalls and the pipe to a box culvert.

The drainage network editor is accessed through the main menu by selecting

Design=>Drainage-Sewer=>Network Editor

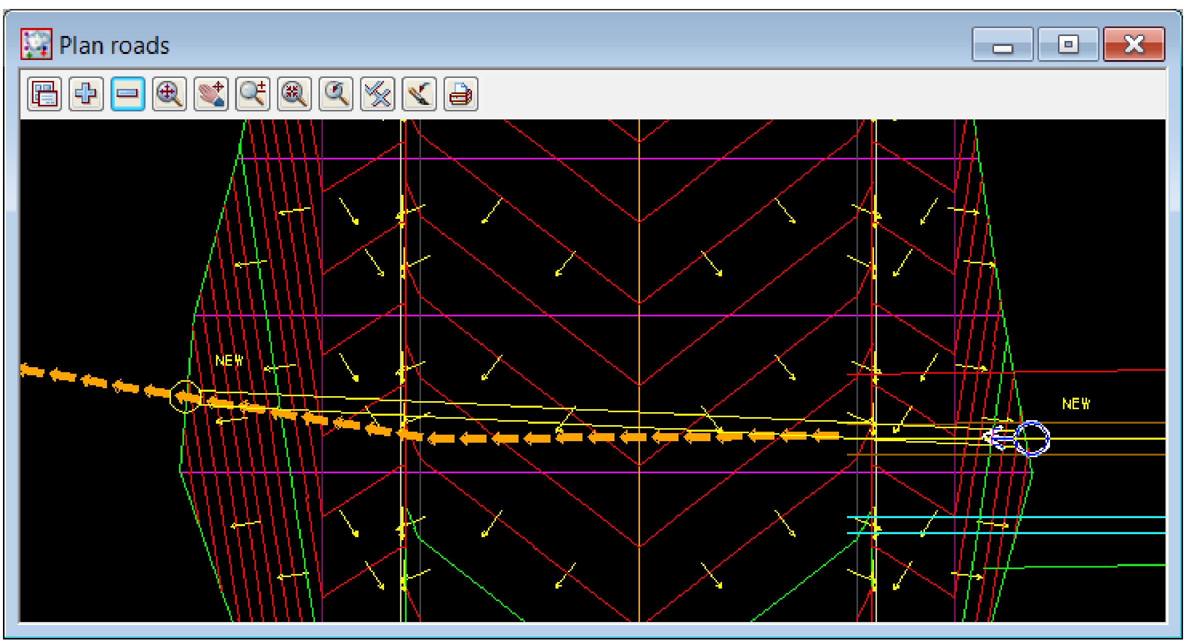
The bottom section of the network editor panel is shown below. You can change tabs but no data can be entered in the panel until a drainage pit or pipe has been selected and accepted.

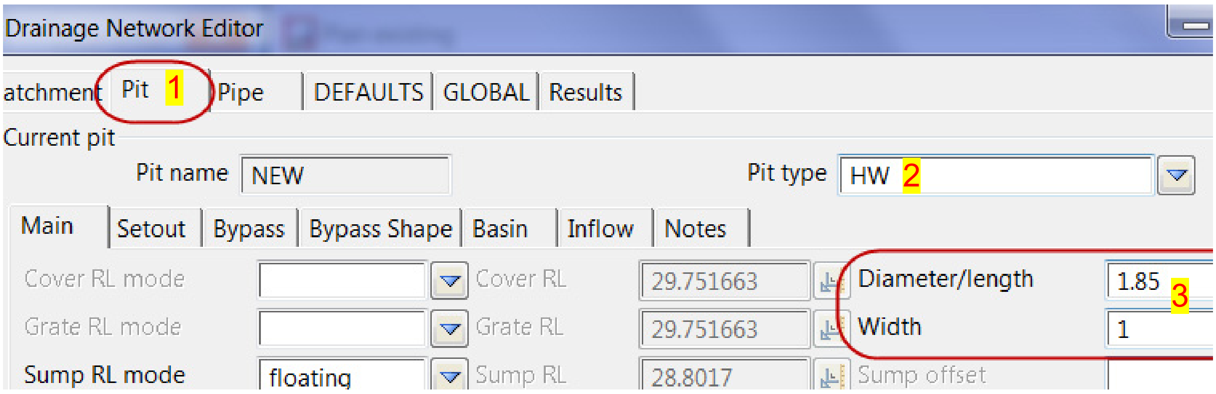


1. Pick and accept the drainage string near the western headwall and the network will be loaded into the panel (only drainage strings may be picked).

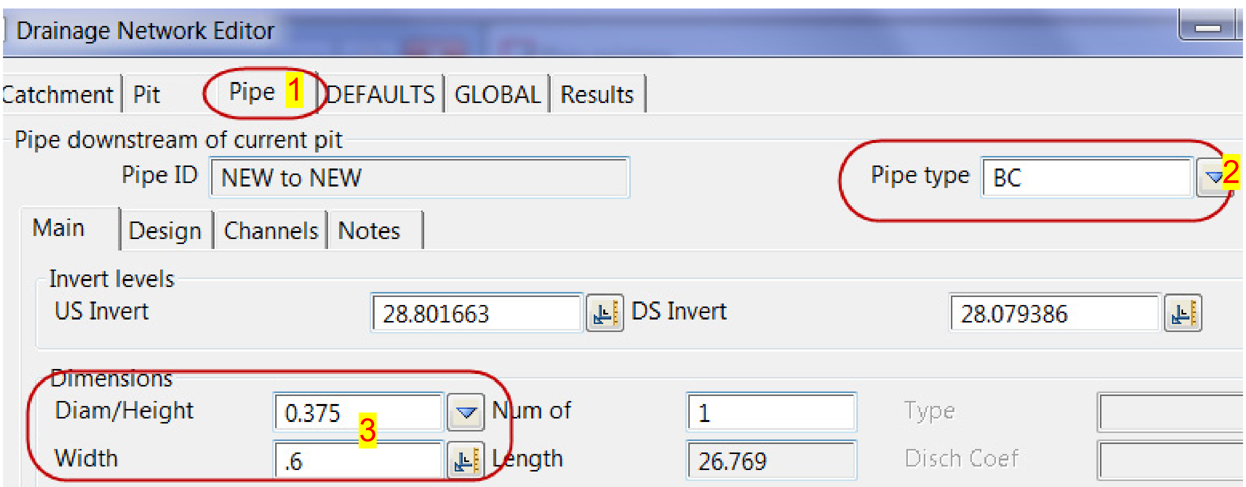
The action buttons on the drainage editor now become active.

The pit closest to the point selected is highlighted with a circle and an arrow shows the direction of flow and the pipe being edited (see image below). When the outlet is selected there is no arrow. If the arrow is in the wrong direction or the outlet is at the upstream end of the string see Flow in the Wrong Direction to correct this.

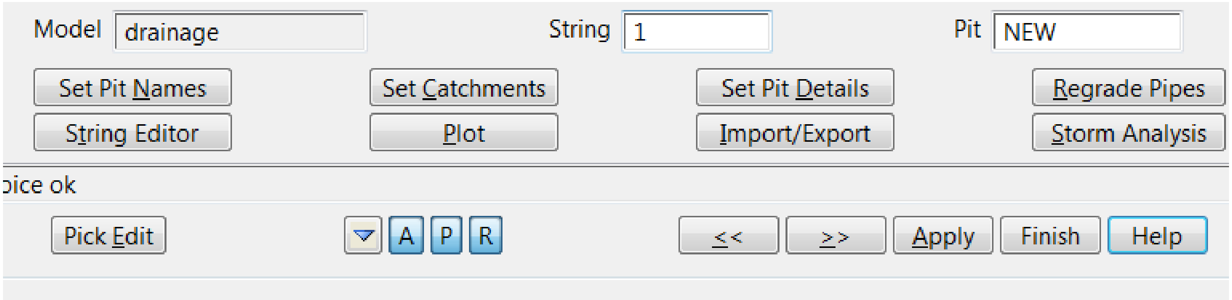




1. Select the **Pit** Tab
2. Note the **Pit type** is already **HW** as we set this as the default. The type may be used to set pit properties. See Drainage Definitions - Manhole Types in 12d help. If you did not set this as your default (Manhole (Pit) Defaults) then change it now.
3. Diameter/length is in the direction of the road once it is linked otherwise it is east-west. Width is 90 degrees to this, If width is blank the pit is circular.
4. Select **Apply** to see the changes.

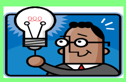


1. Select the **Pipe** Tab
2. Change the **Pipe type** to **BC** (for box culvert). This may be used to set some of the pipe properties. See Drainage Definitions - Pipe Types in 12d help.
3. Change **Diam/height** to 0.375 and width to 0.600. If width is blank the pipe is circular.



1. Now move to the outlet using the **Pick Edit,** **<<** or **>>** buttons. << and >> are left and right in section view and NOT east west.
2. Enter the pit length and width for the outlet headwall.

A popup will appear indicating that the pipe thickness will be changed because the pipe type has a different thickness setting in the drainage.4d. Select **Stop asking for this session** and then click **Update**.



Note: **A P R**. The **A (auto apply)** is active by default. Changes are saved and displayed when you press any button on the DNE except **Finish** or the red **X** in the top right corner. If you turn **auto apply** off, it will be back on the next time you start the DNE.

8.4 Flow in the Wrong Direction

The direction of flow is used by 12d to determine which end of the string is the outlet and therefore the direction of decreasing invert elevations when regrading the string.

The direction of flow will be indicated with the arrow on the pit when using the DNE. No arrow will appear if the outlet is selected. If you find the arrow is pointing in the wrong direction there are 2 methods to access the flow directions.

Method 1: Use the DNE to select the string with the flow in the wrong direction and then select the **String Editor** button. From the menu select **Utility->Properties.** Change the **Flow direction** value to the other value. Select **OK** then **Save & Finish**. Now use the **Pick** button to select the drainage network again.

Method 2: Use the **Strings->Properties->String** option from the main menu. Select the string and then change the **Flow direction** value to the other value.

9.Drainage Strings Levels

The DNE quickly sets the levels for your drainage network. All the levels can be calculated by the DNE or locked with user specified values.

9.1 Pit Levels

Pits have 4 levels:

1. the cover level is used in plotting and pit depth calculations. This is also the maximum obvert level drawn for all connecting pipes.
2. the grate level is used for the freeboard measurements, the surcharge to bypass level and the reference level for pond flooding depths.
3. the setout level is used for survey setout to construct the pit.
4. the sump level (bottom of the pit) is used in plotting and for depth calculations. If set to floating mode, it is calculated from the lowest pipe invert connected to the pit plus the sump offset.

The grate level is the most important level in 12d hydraulics and will be discussed more in those sections.

9.2 Pipe Levels

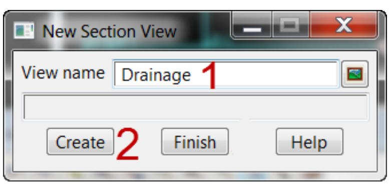
Pipes have the upstream and downstream invert levels that can set by the minimum grade and cover criteria (see Vertical Alignment - Pipes). The invert drop across a pit can also be fixed or change with the deflection angle. At this time we are only concerned with the pipe cover and grade.

9.3 Section View of a Drainage String

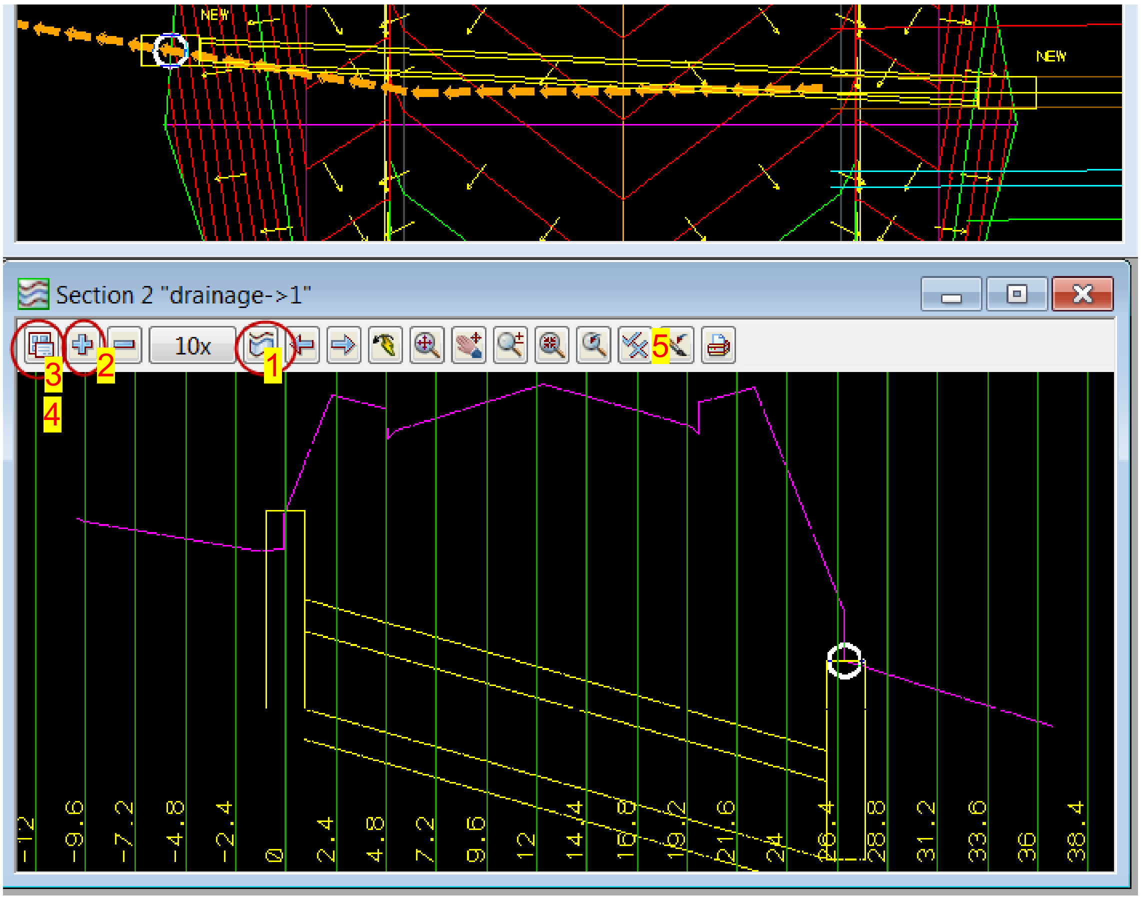
Create a section view to profile the drainage string.

From the main menu select

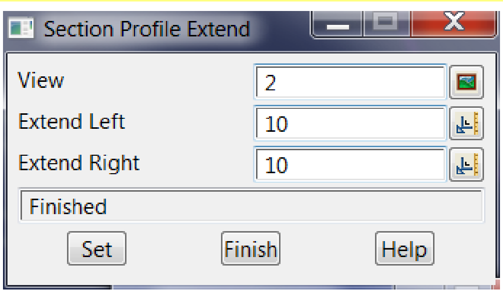
View->Create->Section view

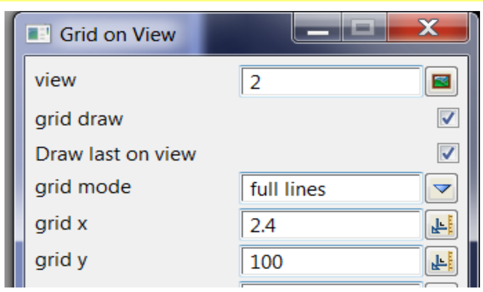


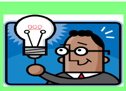
1. Type **Drainage**
2. Select the **Create** button



1. LB the **profile** button then pick and accept the drainage string.
2. LB the + button and add the **tin combined** to the view.
3. LB the view menu button and select **Settings->Extend** to view 10m past the ends of the culvert.
4. LB the view menu button and select **Settings->Grid** to set the horizontal grid to 2.4m (a standard culvert length)
5. Toggles **Grades** on to see the pipe length and grade



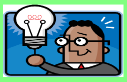




LB the view menu button and select **Settings->Grade annot** to set **Draw grades as 1 in**.

9.4 Set Pit Details

Selecting **Set Pit Details** will set the pit levels and setout coordinates for all of the pits in the network using either the default settings for the model or the explicit settings for the pit (if used). Other attributes are set as well but they will be discussed later. A prompt box appears asking you to confirm. Select **Yes**.



To toggle off/on the confirm request, RB **Set Pit Details**. The same works for **Regrade pipes** and **Set Catchments**

9.5 Regrade Pipes

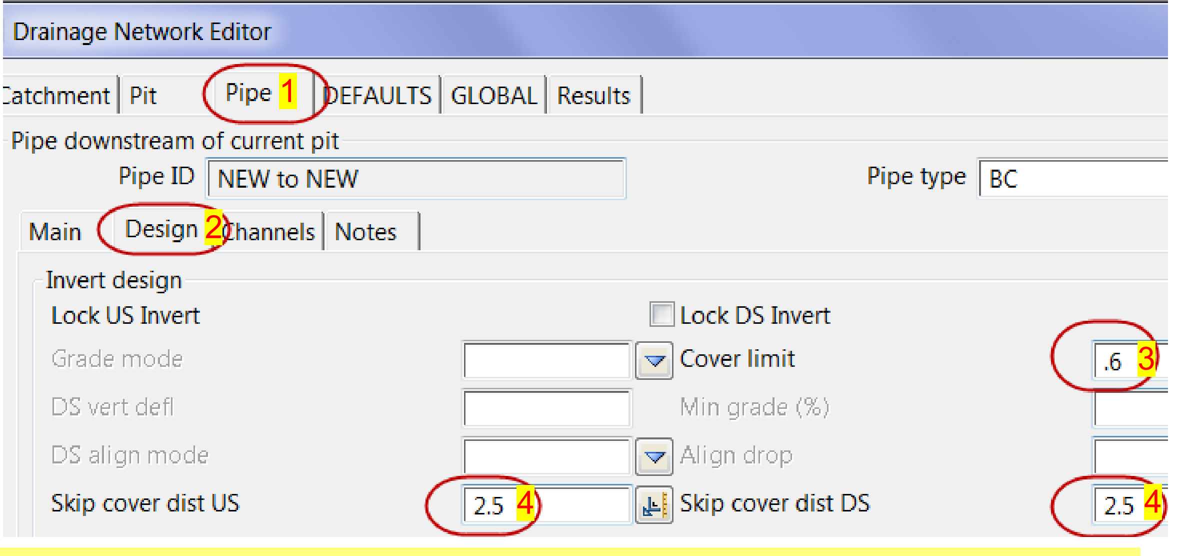
Selecting the **Regrade Pipes** button sets the pipe inverts for all of the pipe in the network using either the default settings for the model or the explicit settings for the pipe (if used). A prompt box appears asking you to confirm. Select **Yes**. To toggle off/on the confirm request RB **Regrade Pipes**.

Minimum grade and cover are checked and trunk lines are lowered to accept incoming branch lines. Grade is calculated using either the pit centre to centre distance or the end of pipe to end of pipe distance (DNE->Global **Use end to end pipe length).** Default pipe cover and grade are set on the DNE **Defaults-Pipe** tab and if not found there the **Drainage->Defaults-Pipe** settings are used.

Invert alignment is based on the pipe cover, the pipe minimum grade and the drop across the pit (see Pipe Grade Modes). The tin specified in the **Global-Main Finished Surface Tin** box is used for these calculations. Since every pipe that enters the pit may have a different drop mode, the drop mode is set on the downstream end of the pipe, NOT the pit.

If the inverts are locked on the **Pipe** tab then some the design criteria may not be able to be achieved. Messages in the output window will indicate these problems.

Now we will regrade the culvert checking the cover only under the roadway. First select the upstream (east) headwall.



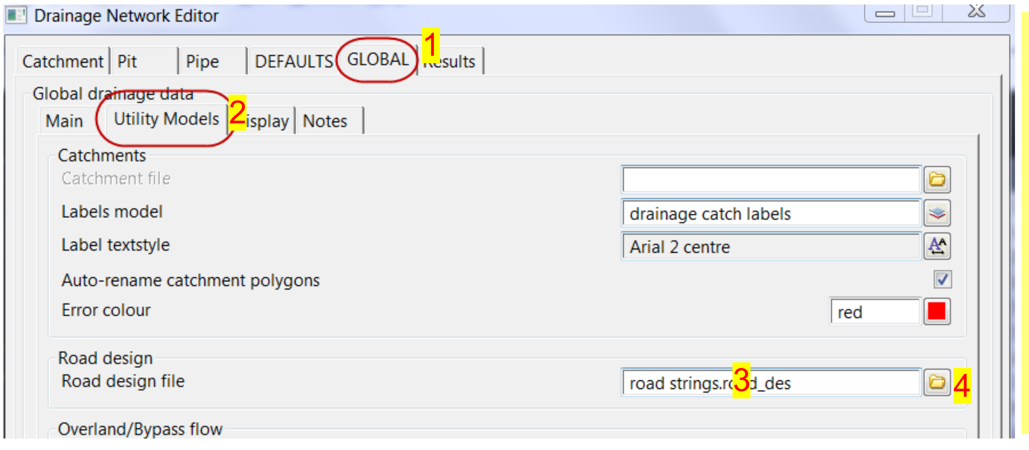
1. Select the **Pipe** tab
2. Select the **Design** tab
3. Type **0.6** to over ride the default cover we set earlier (this is the minimum distance measured from the top of the pipe).
4. Type **2.5** for the Skip cover distances (US and DS). These distances are measured from the ends of the pipe. This will stop 12d from checking the cover in these areas.
5. Select **Regrade Pipe** to reset the inverts.

For circular pipes, the cover is checked along the centre line of the pipe. For box culverts the cover is also measured along the sides. The wall thickness is set via the pipe type and diameter from the drainage.4d file.

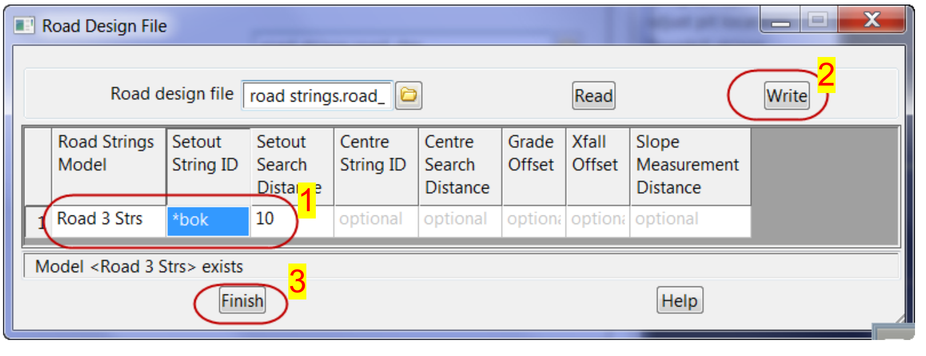
Note: The obvert of the pipe will never be drawn above the grate levels. If the cover or grate level is below the pipe invert you will get an X pipe. **Set pit details** will recalculate the pit levels and Regrade pipe to set the inverts..

9.6 Setout Strings to Align the Headwalls with the Roadway

Now link the headwalls to the roadway so they are perpendicular to the batter slope.

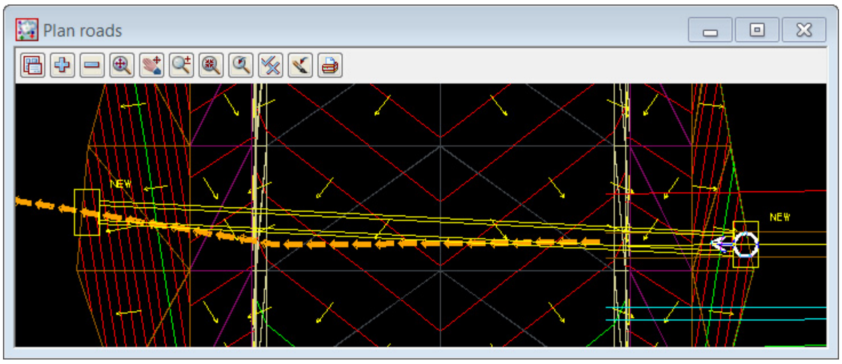


1. Select **Global**
2. Select **Utility Models**
3. Type **road strings**
4. Select the **folder** icon then **Open**

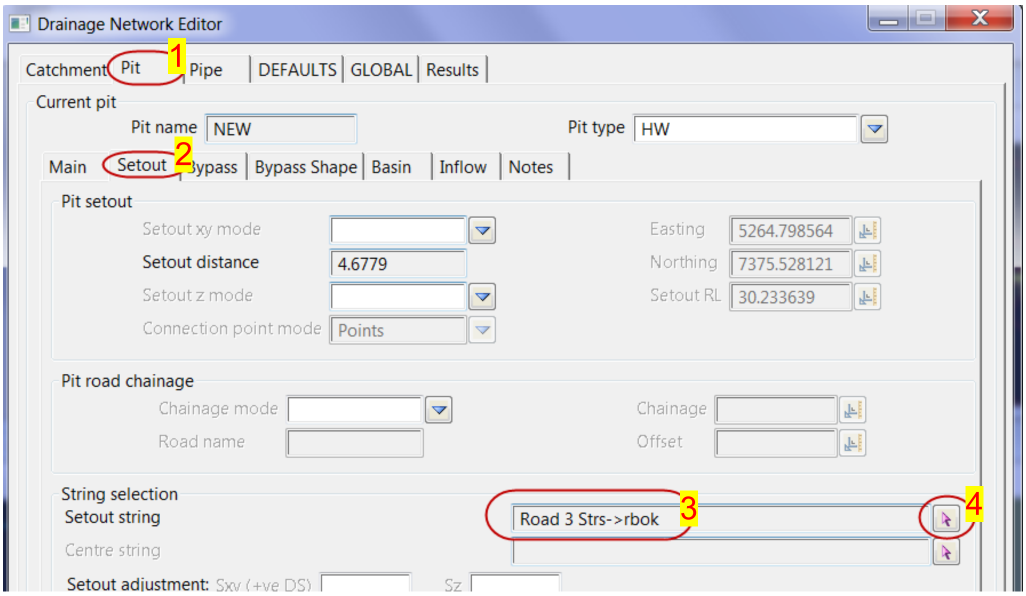


1. RB in the **Road Strings Model** cell and select **Road 3 Strs** model
   type ***bok** so that the headwall will be linked to the closest lbok or rbok string.
   type **10** as the max distance to search for a string
2. LB **Write** to save the file
3. Select **Finish**
4. Select **Set Pit Details** to create the links to the road strings.

The culvert headwalls now align with the roadway instead of east-west.



Now confirm the manhole link to the road strings (setout strings).



1. LB **Pit**
2. LB **Setout**
3. Note: selected string
4. Note: manual pick/clear button

DNE string links exist until you manually clear them or you delete the string.

OR you clear the link by RB on the manual pick and select **Clear,**

OR you clear all road string links via **Global->Utility Models->Clear Road Links.**

We have a good link so do not do this at this time.

9.7 Adjust Pipe Length

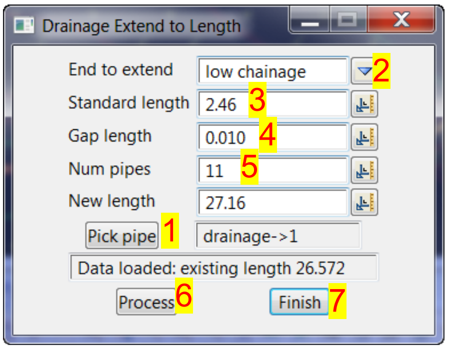
Now adjust the culvert length to an even number of pipe lengths.

Only one editor at a time can access the strings. The DNE locks all of the strings in the model. Close the DNE now, before we start the next editor.

From the main menu select

Design->Drainage-Sewer->More->Extend pipe/culvert

As the culvert appears to be too long we will shorten it to 10 standard lengths (half on both ends).



1. Pick and accept the culvert
2. Change **End to extend** to **high chainage**
3. Change **Standard length** to 2.46
4. Change **Gap length** to 0.010
5. Change **Num pipes** to 11 and press enter. the new length is displayed
6. Select **Process**
7. Select **Finish**

After selecting process, the section view will need updating. Also the pit cover levels are set to the pipe invert level. The next time you press **Set pit details** in the DNE they will be reset using the selected method.

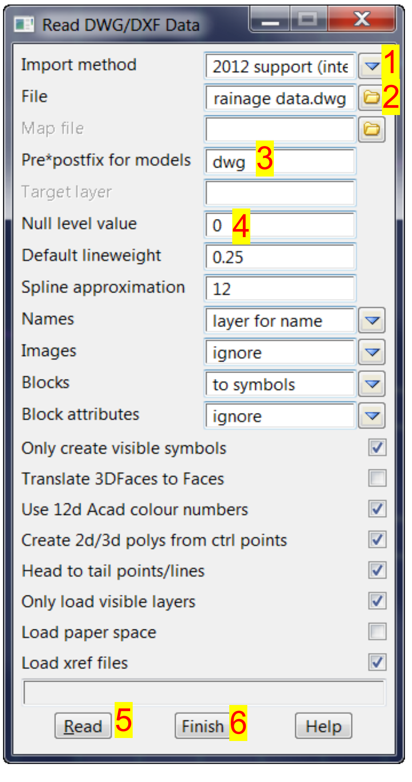
10.Importing Drainage Layouts from AutoCAD

We are now going to import a pipe layout that was drawn in CAD along with the overland flow routes and catchment areas (These could have been created in 12d as well). When these were drawn the following rules were followed:

1.Strings are drawn where the pipes are NOT to setout x,y locations,
2.Polylines are used in AutoCAD,
3.lines drawn from upstream to downstream (direction of flow),
4.a vertex was placed at every pit location.

To import the AutoCAD drawing, from the main menu select

File IO->Data Input=>DWG/DXF/DXB



1. Select 2012 support
2. LB the folder icon and select the **drainage data.dwg** file
3. Type a **prefix** for the models. It will help organise the layers from AutoCAD as every layer goes into a separate model in 12d. Specifying a prefix causes all of the layer names to be prefixed with this text and therefore kept together in the model list. The prefix used is dwg<space>.
4. Enter the null value **0** (no level in CAD value)
5. LB **Read** button. If you select the **Read** button more than once the data will be imported again and you will get duplicate, triplicate...etc data.
6. LB **Finish**

The models that have been created are,

dwg Catchs Future
dwg Catch Lots
dwg Catch Reserve
dwg Catch Roads
dwg network

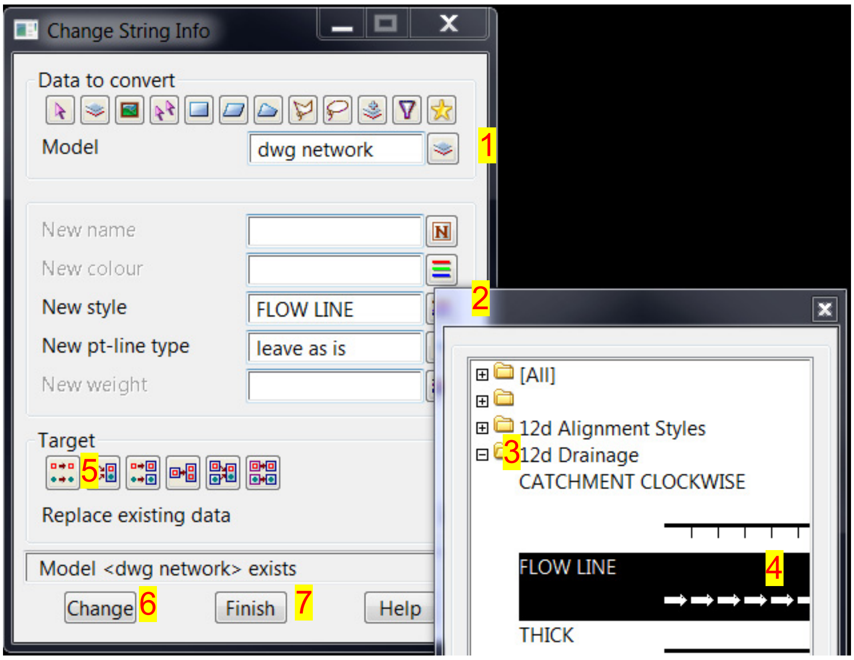
dwg network is a drainage layout that we will use to locate the pits in our drainage design.

10.1 Checking the String Direction of CAD Network Strings

A quick way to check the direction of the strings imported from CAD is to change the linestyle to a style that indicates the direction.

From the main menu select,

Utilities->A-G->Change



1. Select the dwg network model that contains the drainage network.
2. Select the **New Style** icon
3. Select + on the **Drainage 12d** group
4. Select **FLOW LINE** from the style list
5. Select the **Replace** option from the **Target** buttons
6. Select **Change**
7. Select Finish

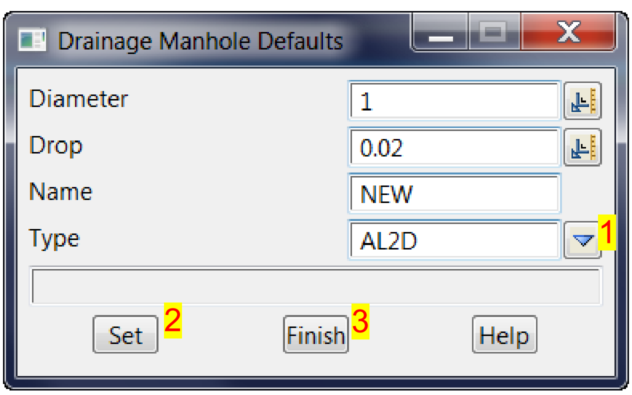
10.2 Create the drainage strings from the CAD strings

Before creating the drainage strings, Read in a drainage model template so that all of DNE settings will be set. If you forget, do it later but **READ A TEMPLATE FILE.** If the drainage strings are going to be added to a model that already has the global and default settings set, these strings will use those model settings.

12d will convert the import strings into 12d drainage strings. The default pipe, pit and tin data will be used to set the levels for the network. Do not use the other string convert commands found on the menu system.

Most of the pits will be grated roadway side entry pits (AL2D). Set this pit type as the default.

Design =>Drainage-Sewer =>Defaults=>Manholes

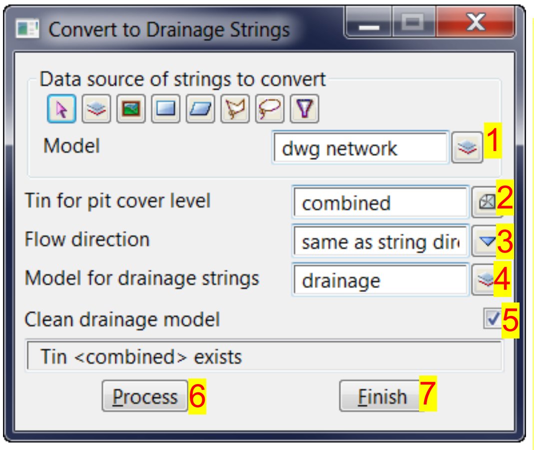


1. LB the **Type** drop down and select **AL2D**.
2. LB the **Set** button.
3. LB the **Finish** button.

Next from the menu select

Design=>Drainage-Sewer=>Create=>Create from strings

The following panel will appear.



1. Select the existing **dwg network** model
2. Select the **combined** tin. The cover level for the manholes will be obtained at the manhole centres from this tin.
3. Leave as **same as string direction**. The strings were drawn in the same direct as the flow in th CAD program.
4. Select the existing **drainage** model. If this model was not created from the read drainage model template function then you could type the name now and the model would be created.
5. **Clean drainage model** when selected will delete all strings in the model before the new drainage strings are created.
6. **Process** create the drainage strings
7. **Finish** closes the panel

The imported strings must all be drawn in the same direction. Either all in the direction the water flows or all opposite the direction of flow.

Pits are created at all vertices on the strings.

Trunk lines must have a vertex where the branch lines join.

Delete the new plan that was created and observe the new drainage strings created (road view).

11.Horizontal Alignment and Drainage String Edits

11.1 Road design strings (centre line) for Adjust Pit Locations

During the design process, roadways are often moved slightly, requiring the pit location to be adjusted. Linking the inlets to the road design strings enables 12d to quickly adjust these locations and create a report of the pit requiring adjustment.

The Adjust Pit Locations routine will move the centre of the pit perpendicularly to a **layout string** or offset from a **road setout** string.

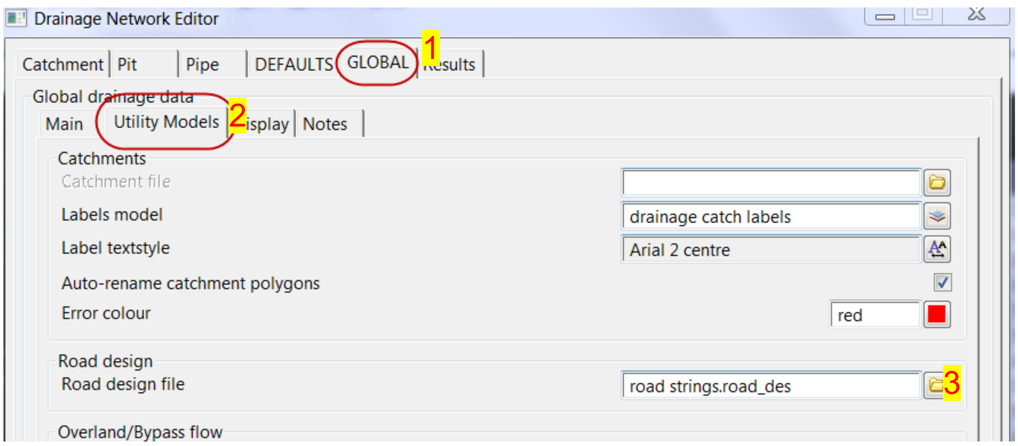
Layout strings do not need to be set up in advance but they do not allow for an offset.

Road design strings need to be setup in the DNE but the allow an offset from the setout string that can vary with the pit type. Often large pits will be offset more into the footpath or roadway.

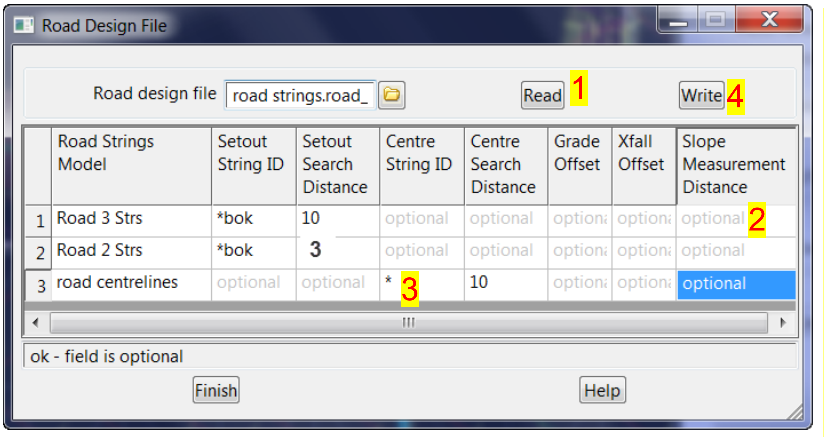
Road and layout strings can be used at the same time. If no road string is found, a layout string will be searched for.

In this example were are going to use the road setout strings not layout strings. We have already linked the culvert headwalls to the road design strings on Road 3. We will now do the same for the roadway inlets so they will align with the roadway and we can offset the centre of the pit a fixed distance from the setout string. This will require a link to the road centre line as well. A positive offset is away from the centre line and negative towards the centre line.

Start the DNE and select the network.

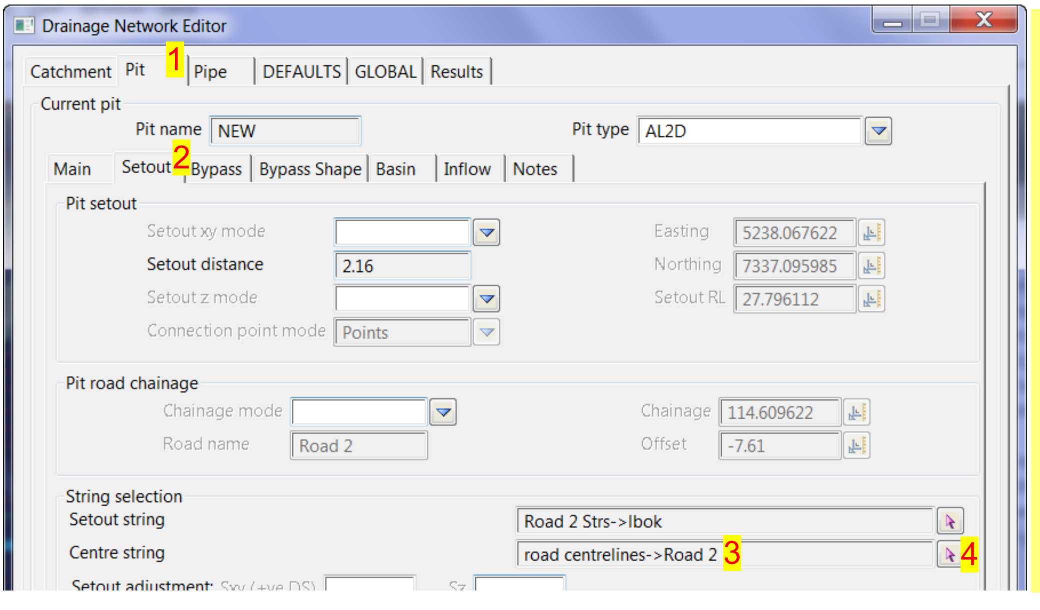


1. Select **Global**
2. Select **Utility Models**
3. Select the **folder** icon then **Open**



1. RB in the **Road Strings Model** cell and select **Road 2 Strs** model
   type ***bok** so that the inlets will be linked to the closest lbok or rbok string.
   type **3** as the max distance to search for a string
2. press enter until row 3 is added
3. type * for **Centre String ID**. This wild card will allow a string with any name (blank will not work).
   type **10** as the max distance to search for a string
4. LB **Write**, then **Replace** to save the file
5. Select **Finish** then **Set Pit Details** to create the links to the road strings.
6. Regrade pipe to set the inverts.

To confirm that the inlets were linked to setout (*bok) and centre line strings,



1. LB **Pit**
2. LB **Setout**
3. Note: selected string
4. Note: manual pick/clear button

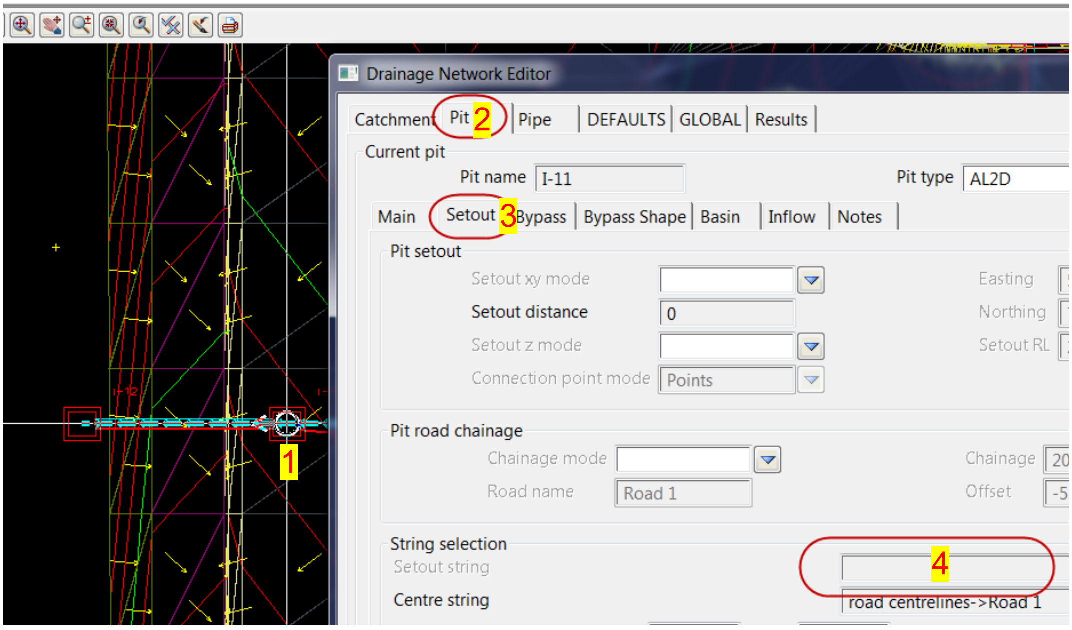
11.2 Log Lines with the Drainage Network Editor

Double clicking on a log line in the Output window will take the DNE directly to the pit the message refers to!

Selecting **Set Pit Details** and messages are created in the **Output Window** for information and error checking. If the line of text begins with a exclamation mark it is a log line.

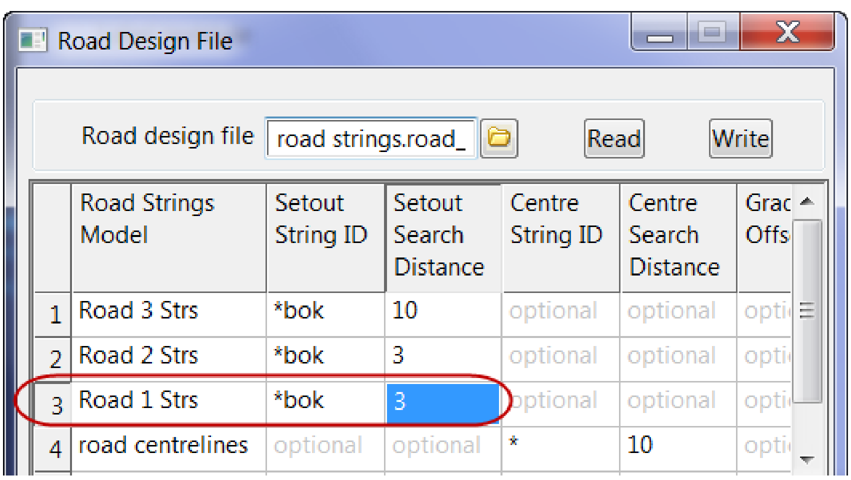
```
!INFO: Pit "NEW" Setout distance is:      0.450
!PROBLEM: Pit "NEW" Setout string not found
         ... needed for pit grate rl
         ... needed for pit z setout
         ... needed for pit xy setout
         ... needed for pit symbol angle
```

1. Double click on the word **PROBLEM**.



1. Note: the DNE (if open) moves to inlet NEW.
2. Select the **Pit** tab
3. Select the **Setout** tab.
4. Note that the **Setout string** field is blank

No road setout string has been found because we have not included the model **Road 1 Strs** in the **Road strings file** found on the **Global->Utility Models** tab of the DNE. The *bok strings in the **Road 2 Strs** and **Road 3 Strs** were more than the search distance away.



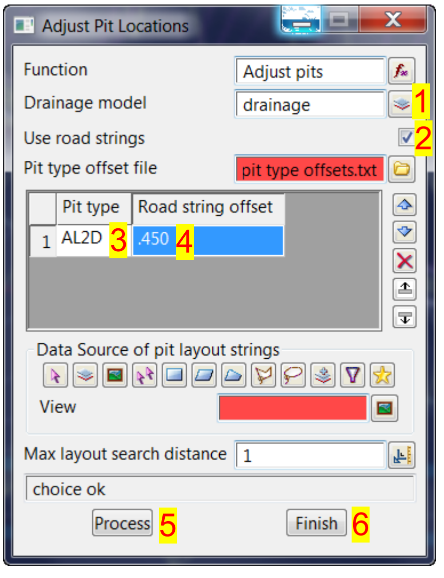
1. Add the new row and complete with the data as you have done before.
2. After **Writing** the file select **Set Pit details.**
3. Confirm that I-11 now has a setout string.

11.3 Adjust Pit Locations

Now we can offset the inlets from the bok setout strings.

Close the DNE and then from the main menu select,

Design->Drainage-Sewer->Adjust pit locations



1. Select the model **drainage**.
2. ensure **Use road strings** is selected.
3. RB and select **AL2D** pit type
4. type **0.450** for the offset away from the road centre line
5. Layout strings and layout search distance are not being used at this time.
6. Select **Process** and redraw the plan view to see the new pit locations.
7. Select **Finish**

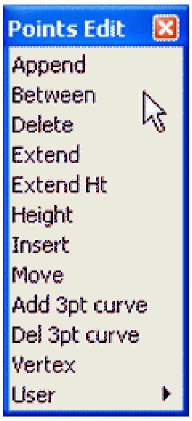
The message box indicates that 13 of the 18 pits were adjusted. Check the output window for a list of the pits that were adjusted. If no road string is found then the layout string is searched for. Since we did not use layout strings we get the warning messages for 3 of the pits. Note: You will not have pit names assigned yet (Set Pit names (and pipes)) so your names will be just NEW.

```
WARNING: Adjust Pit Locations - pit "I-07" found no perpendicular layout string within
the specified search distance.
INFO: Adjust Pit Locations - pit "I-04" adjusted by 2.610 units.
WARNING: Adjust Pit Locations - pit "I-08" found no perpendicular layout string within
the specified search distance.
INFO: Adjust Pit Locations - pit "I-05" adjusted by 2.610 units.
INFO: Adjust Pit Locations - pit "I-06" adjusted by 2.610 units.
INFO: Adjust Pit Locations - pit "I-07" adjusted by 2.610 units.
INFO: Adjust Pit Locations - pit "I-08" adjusted by 2.610 units.
WARNING: Adjust Pit Locations - pit "I-10" found no perpendicular layout string within
the specified search distance.
INFO: Adjust Pit Locations - pit "I-09" adjusted by 2.610 units.
INFO: Adjust Pit Locations - pit "I-10" adjusted by 2.610 units.
WARNING: Adjust Pit Locations - pit "I-11" found no perpendicular layout string within
the specified search distance.
WARNING: Adjust Pit Locations - pit "I-12" found no perpendicular layout string within
the specified search distance.
```

11.4 Manual Horizontal Alignment - Moving, Adding and Deleting Pits

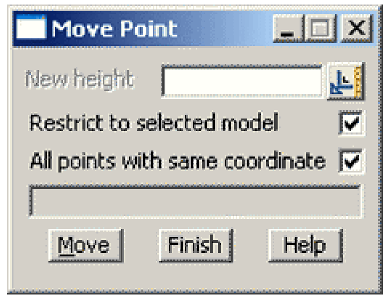
The most common functions for adjusting the horizontal alignment of manholes and pipes are found on the

Strings->Points Edit menu.



The **Append, Between, Delete, Insert and Move** commands are the most common.

The **Move** command has extra options for moving junction pits. Selecting **All points with the same coordinate** will move all points on the view that are at the same x,y coordinate. The point selected will move first and the others will follow **after** the new location has been accepted. **Restrict to selected model** will stop data from other models at the same location from being moved.



The second option

Strings->Edit

gives you all the same commands and more but requires you to select a string first and the edit commands will be restricted to the selected string.

12.Completing the Intersection Drainage Design

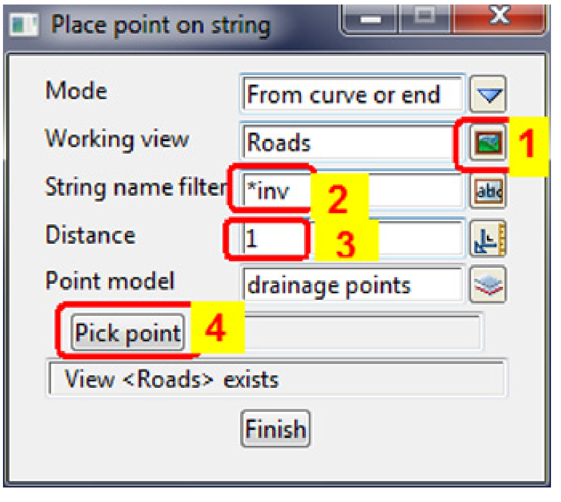
When placing inlets around an intersection, it is often required to place the centre of the inlet a specified distance from the curve so that the grate and/or the side inlet will be located on the straight section of kerb. If there is a pedestrian crossing, the inlet should be placed at a specified distance upstream to be clear of the crossing.

12.1 Placing Marker Points around the Intersection

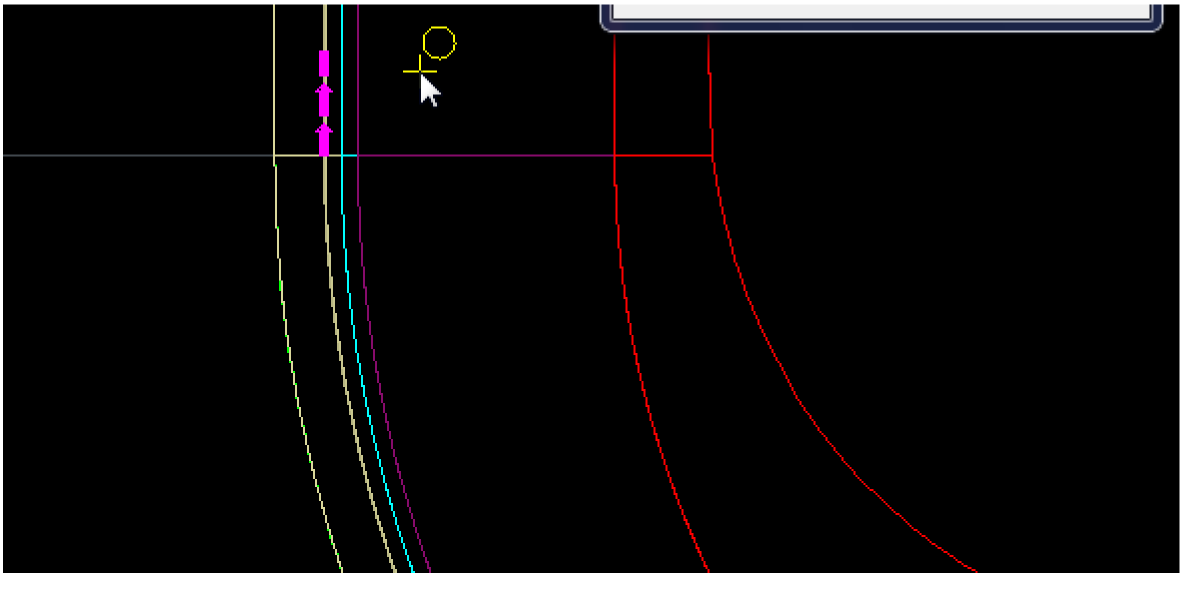
The place points routine is a fast way to place inlet location markers at a specified location from the curved section of the kerb.

From the main menu select

Design->Drainage-Sewer->More->Place points

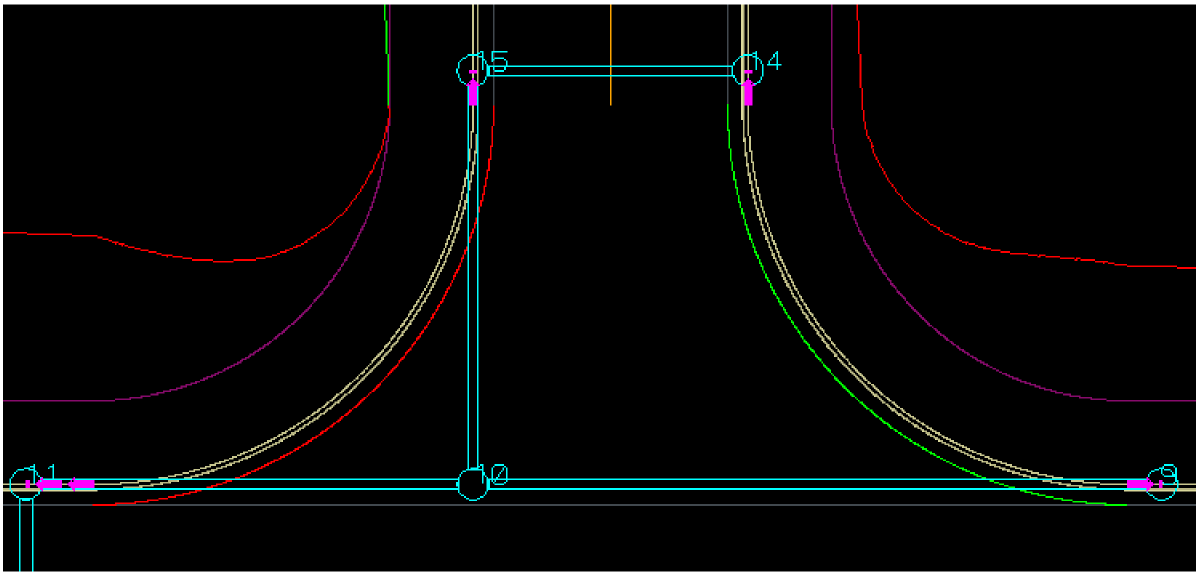


1. Select the **Roads** view as it contains the inv strings where we want the pipes.
2. Type ***inv** so the linv and rinv strings can be selected.
3. Type the distance from the end of the string or the start of curve where the pit is to be placed.
4. Select **Pick point** and then snap near where the pit is to be placed. The routine will find the closest *inv string and then search for the closest end or curve. It will then measure from the point towards your initial snap to create the marker (see magenta arrows below).
5. repeat for the remaining 3 ends of the kerb returns.



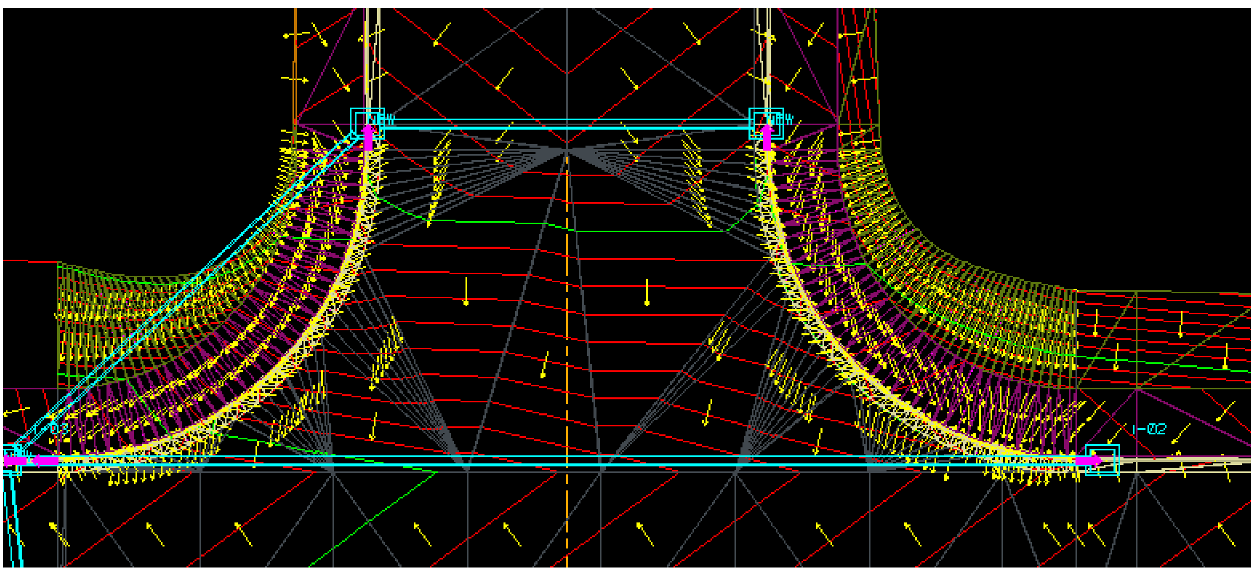
12.2 Option 1 (Connection at Manhole in the Road)

The option below has been selected for you to complete so that it has 2 inlets and a manhole. Use the tools you have learnt thus far to create the drainage string and assign the pit names. Special care is required when creating a junction pit (10) shown below. There must be a pit on the trunk line where the branch line connects and there branch line must point snap onto the centre of the trunk line pit (see Junction Pits).



12.3 Option 2(Connection pipe under the footpath)

Some road authorities prefer not to have a manhole in the intersection. Move the connection pit to the inlet to the west and delete the manhole.



A check list is on the following page once you have completed the layout.

12.4 Intersection Check List

1. Did you place the pipes in the correct location so that the correct cover could be measured?
2. When placing the manhole did you use the RB and **Perpendicular** to place the manhole?
3. Did you use **Strings->Points Edit->Insert** to add a pit to the trunk line at the junction location?
4. Did you assign a unique string number to the new string so that the set pit names routine would run?
5. Did you select **Set Pit Details** in the DNE to ensure the cover, grate and setout data was correctly calculated?
6. Did you check the output window for problem messages? Did you fix them?

7. Did you recalc the function **Adjust pits** so that the inlets are 0.450 behind the bok string.

Below are 2 other methods for placing pits that you may find useful.

12.5 Offsets from strings

As an alternative to the place points above you can use the snaps CAD to place the pits “on the fly”. Once the points move has been started use the **RB** and select
Snaps Cad=>Points=>Locate Offset. Follow the prompts given in the message area (bottom left corner of the screen). You will need experience with the 12d “directional pick” to use this capability.

12.6 Placing pits at specific Easting Northing Locations

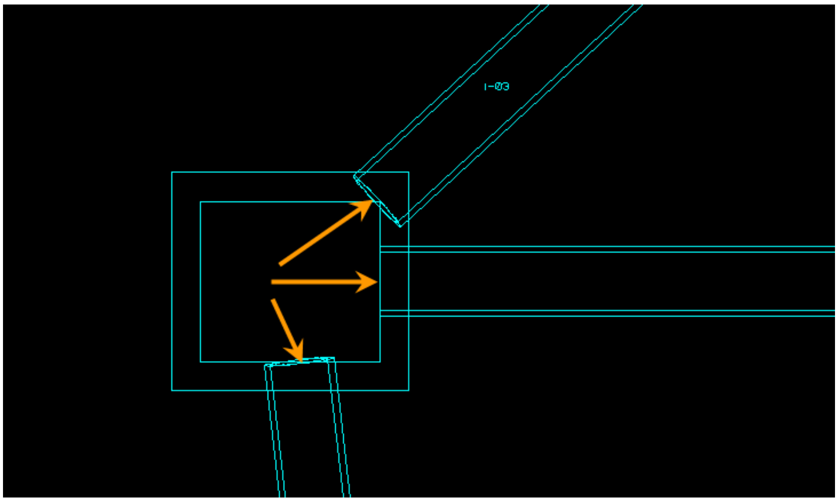
For locating pits at specific x, y coordinates, simply start typing the x coordinate instead of clicking onto a location. An input panel will appear for you to enter the x and y coordinate separated by a space.

13.Manholes - A Closer Look

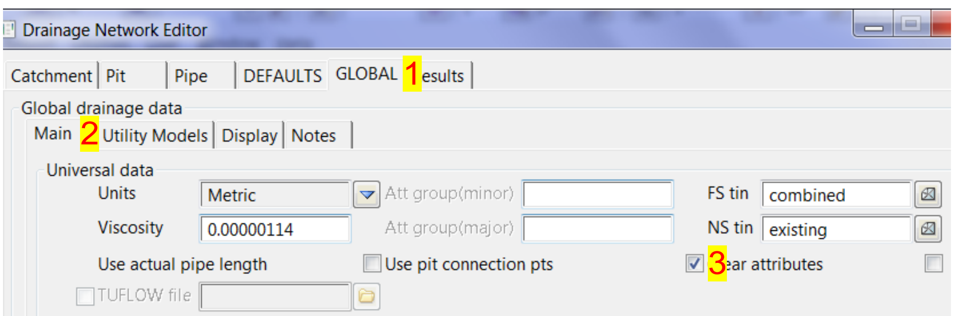
A manhole is made up of 3 points; the centre of the manhole and the pit-pipe connection points. The pipe exists as the connection between two manholes. The pipe may have a horizontal radius but not vertical.

13.1 Pit-Pipe Connection Points

The default mode has the pipes align between the centres of the manholes. The pit-pipe connection points are at the intersection of the manhole interior wall and the centre of the pipe (see below).

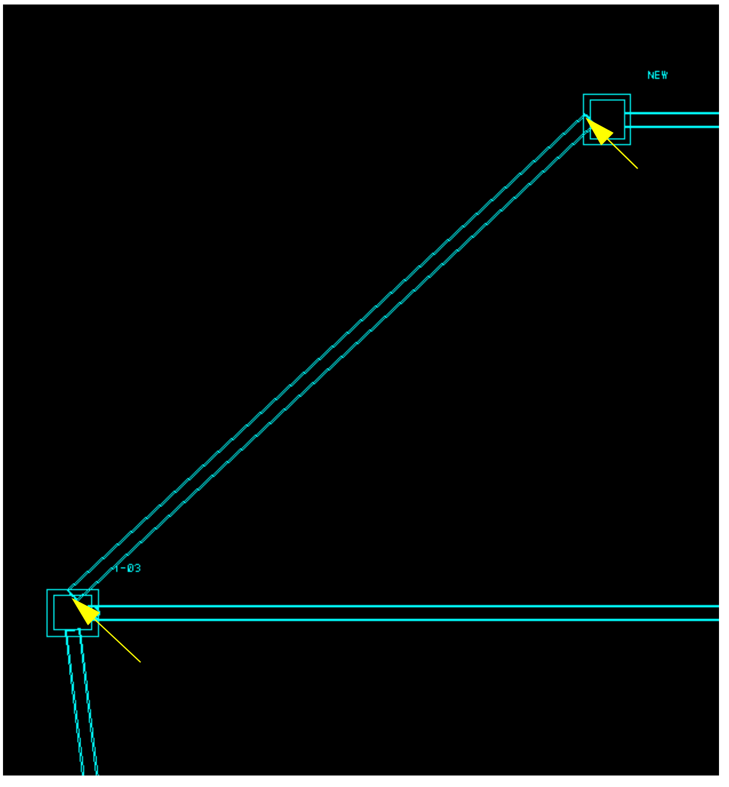


These connection points may be moved by enabling **Use pit connection points** on the **DNE->Global->Main** tab. All pits have their connection point initially set to **Points**. This setting is found on the **DNE->Pits->Setout** tab.



1. Select the **Global** tab.
2. Select the **Main** tab
3. tick the **Use pit connection pts** tick box
4. Select **Apply**

Pit connection points may be moved via **Strings->Points Edit->Move** (except for **Centre** mode described below). If a pipe is manually moved to a new connection point, it will be locked to the connection point and will not move if the pit or neighbouring pit is moved.



1. Close the DNE to unlock the drainage strings.
2. Use **Strings->Points Edit->Move to** move the junction pit you added to complete the intersection westward to the inlet.
3. Use **Strings->Points Edit->Move** to move the ends of the pipes as shown. Note that the pipes snap to the centre of the edges.

13.1.1 Pit Connection Point Modes

Centre (rectangle and circular) - This mode is the same as having the **Use connection points** turned off**.** The connection points will be located on the inside perimeter of the pit wall with the centre line of the pipe intersecting the centre of the pit. In this mode, the connection points may not be adjusted.

Points (rectangle) - A connection point is created at the mid point of each internal side of the pit. This may be changed for a **Pit type** by using the **con_points** command in the drainage.4d file. In this mode, the pipe ends will snap to the connection points. It is possible to place more than one pipe on the same connection point (the elevation of the pipes is not checked for clashes).

Points (circular) - The connection point may be moved anywhere around the pit internal wall as there are no connection points on the circular pits. Again, it is possible to place more than one pipe on the same connection location (the elevation of the pipes are not checked for clashes). If the manhole centre is moved the connection point locks are removed.

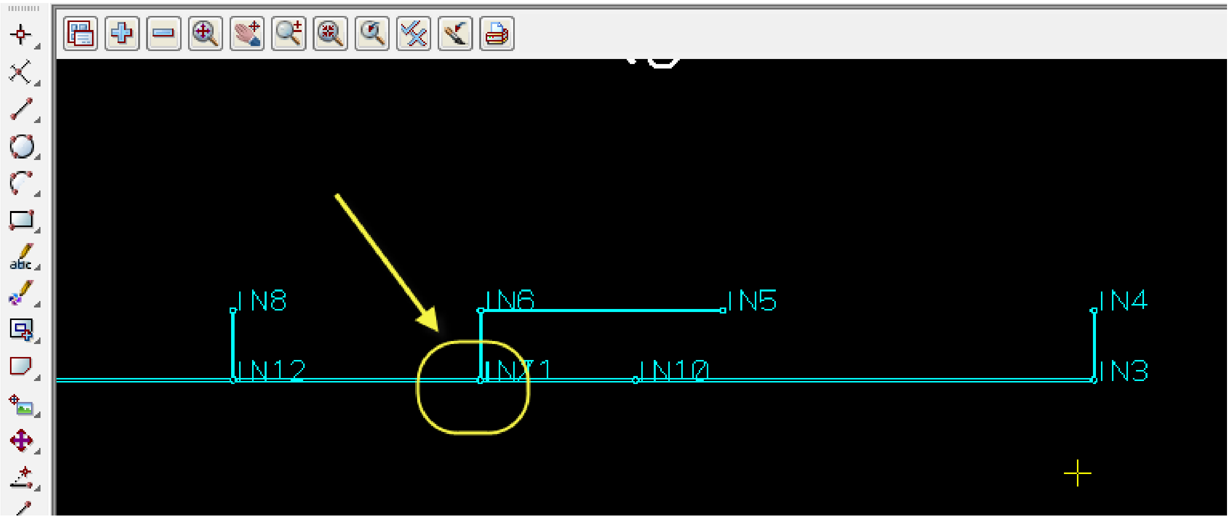
Perimeter (rectangular and circular) - Same as **Points** (circular) above.

Unrestricted (rectangular and circular) - There are no constraints on the location of the pit connection points.This mode is intended for irregular shapes such as GPT structures and stormwater basins.

13.2 Junction Pits

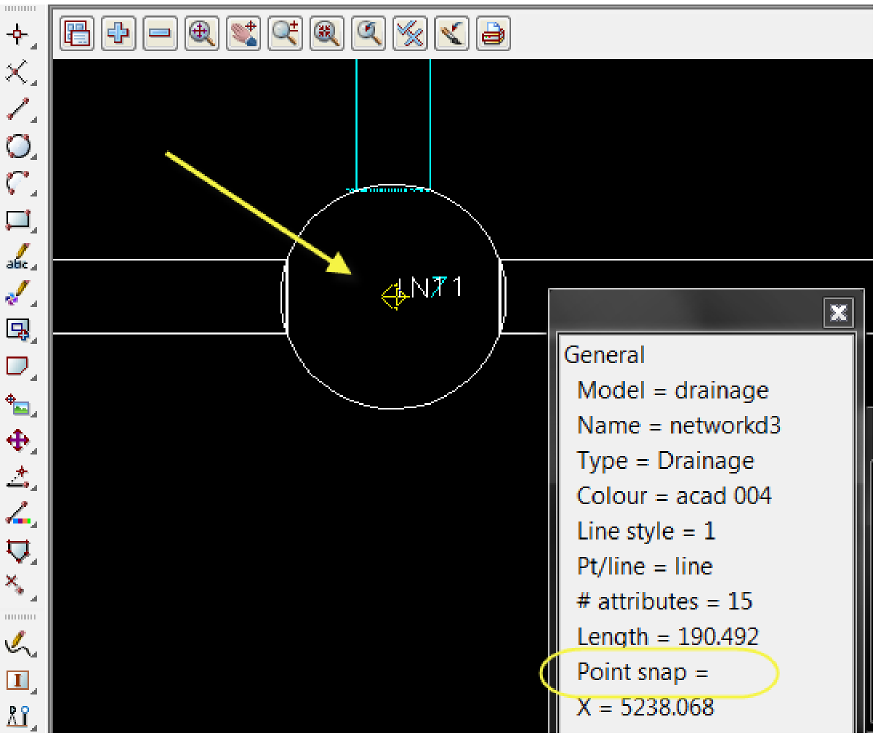
Key Points

1. The branch line must **Point Snap** onto the centre of a pit on the trunk line.
2. Both strings must be in the same model.
3. The downstream end of the string (depends on flow direction) must join onto the trunk line.



When creating a junction, pit turn the point snap on and the line snap off (the F3 and F4 keys are convenient for this). Zoom into the pit so that you can snap the being moved onto the centre of the stationary pit.

Recall a pit contains three points; one at each pit pipe connection points and one at the pit centre. You want to snap onto at the pit centre for a junction pit. In the figure below, the blue line is being placed to join the white line. Note that the diamond indicates that there is a point snap as well as the information panel.



If you cannot see your branch lines joining in the section view. check the following:

1. The centre of the pits did not align, **Use Strings=>Points Edit=>Move** to move the branch string pit the “downstream end” of the branch line must be the junction pit.

2. The branch string and the trunk string have not been created in the same drainage model. From the main menu select **Strings->Inquire (F2)** and select the strings to check their models. If this is the problem, use **Strings=>Edit=>Change** and specify the correct drainage model (enter the model BEFORE picking the string) or **Strings=>Edit=>Duplicate** to duplicate one of the strings into the correct model.

13.3 Drainage Section Views (downhill left to right OR right to left)

The long section views and the profile plots are running downhill from left to right. If you want them downhill from right to left use the **reverse** function.

To reverse only one string, from the main menu select

Strings =>Strings Edit =>Reverse

and pick the drainage strings to reverse. This will also change the drainage flow direction attribute from **same as string direction** to **opposite to string direction.**

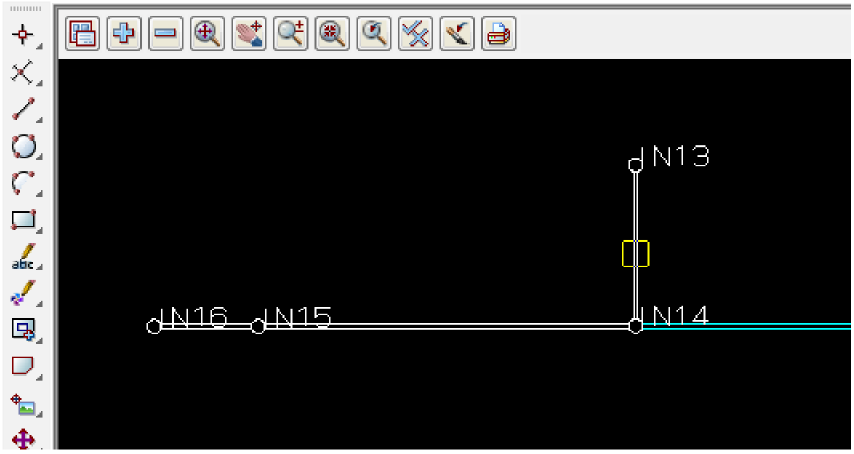
To reverse all of the strings in a model select the Reverse all strings option,

Design =>Drainage-Sewer =>More=>Reverse all strings

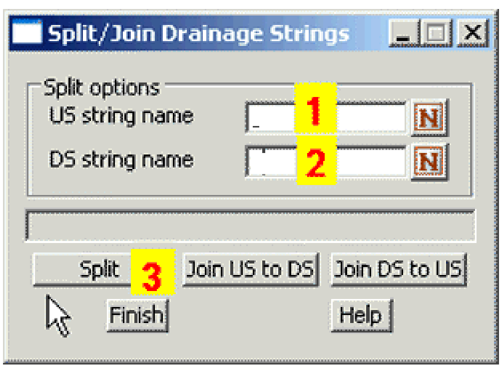
13.4 Drainage Split and Join

In this example, we are going to **split** the string shown in white below at IN14 (your pit are still NEW) so that we can **join** the west end (IN14 to IN16) to the trunk line on the southern side of the road (cyan). This will allow us to produce a long section that will extend the full length of the road 2 to the headwall outlet.

Use the **String split/join** from the Drainage menu not the one on the **Strings->Strings edit.**



Split Option

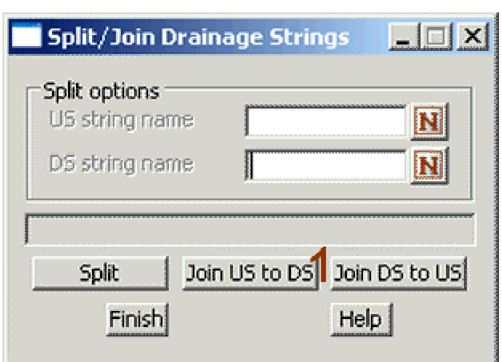


The split must be done at a pit.

1. A new name for the upstream string (optional). Leave it blank
2. A new name for the downstream string (optional) Leave it blank.
3. LB the **Split** button and then select the pit where the split is to occur. (IN14).

Join Option

This routine will add a new pipe between the 2 strings if required. When **Join US to DS** is selected, the properties and attributes of the upstream string and pit at the join will transfer to the new joined string.m string. **Join DS to US** will cause the properties and attributes to come from the ds string. Upstream and downstream is determined by the flow direction



1. Select the **Join US to DS** button.
2. Pick and accept the cyan drainage string, then the IN14 to IN16 string (follow the messages in the 12d message area).

14.Drainage Network Editor (DNE)

We have had a brief introduction to the DNE when we worked with the culvert. We will now look at many more of the DNE capabilities.

Read in a drainage model template! There is no need to set all of your global and default settings each time you start a new job. If we had not already read in a template we wold do it now. Templates are a way to store all of your Network Editor - Global, Default Settings and Explicit Settings

The drainage network editor is used to automatically or manually change the properties of your drainage network. These abilities include:

Vertical Alignment - Manholes

Set Pit names (and pipes)

Service and Utility Clashes

Hydrology

Catchment Areas

Catchment Areas and Percent Impervious

Coefficients of Runoff

Times of Concentration

Tc Path Strings

Bypass flow routes

Hydraulics

Cover RL, Grate RL, Setout RL and Sump RL modes,

Culvert Hydraulics and Tailwater

Pit Losses Ku, and Direct Flow

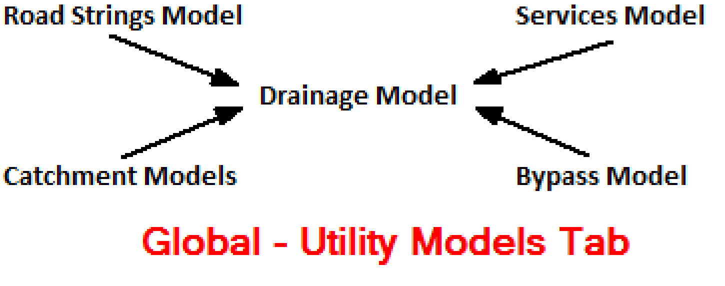
Pipe Friction Method, Roughness Values and Direct pipe flow

Design mode, Freeboard Limit and Flow-depth limit

Pipe Size Design

Pipe Design Parameters - Sizes, Invert alignment, Min Cover, Max Height

The DNE also links your drainage model to your 12d design models.

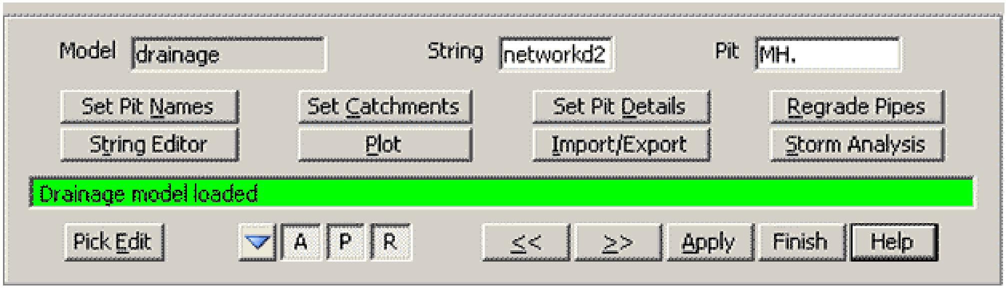


The drainage network editor is accessed through the main menu by selecting

Design=>Drainage-Sewer=>Network Editor

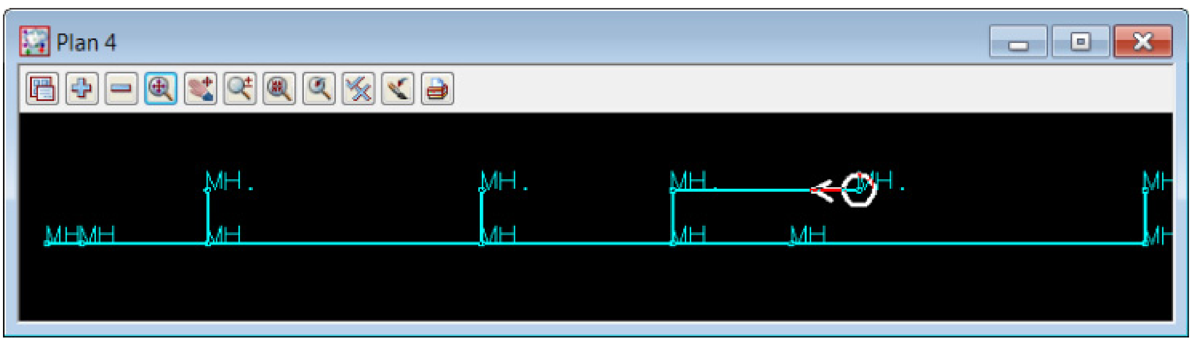
There is no need to fill in all the fields. On the **Catchment**, **pit** and **pipe** tabs, blank fields will use default values from the **Defaults** tab. Check the tool tip for the default value being used.

The bottom section of the network editor panel is shown below. You can change tabs but no data can be entered in the panel until a drainage pit or pipe has been selected.



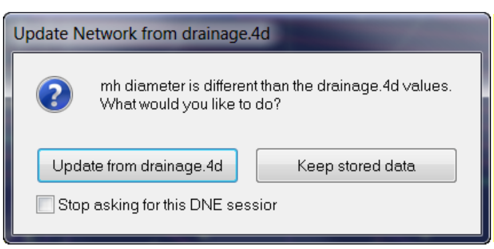
1. Pick and accept a drainage string and the network will be loaded into the panel (only drainage strings may be picked).

The pit closest to the point selected is highlighted with a circle and an arrow shows the direction of flow and the pipe being edited (see image below). When the outlet is selected there is no arrow. The option buttons on the drainage editor now become active.



14.1 Update from drainage.4d

We have set our default diameter as 1m and default manhole as AL2D. In the drainage.4d file the AL2D manhole is defined having a diameter/length of 0.93m and a width of 0.85m. When you select one of the manholes the DNE finds the difference and asks you if you want to update the manhole diameter.



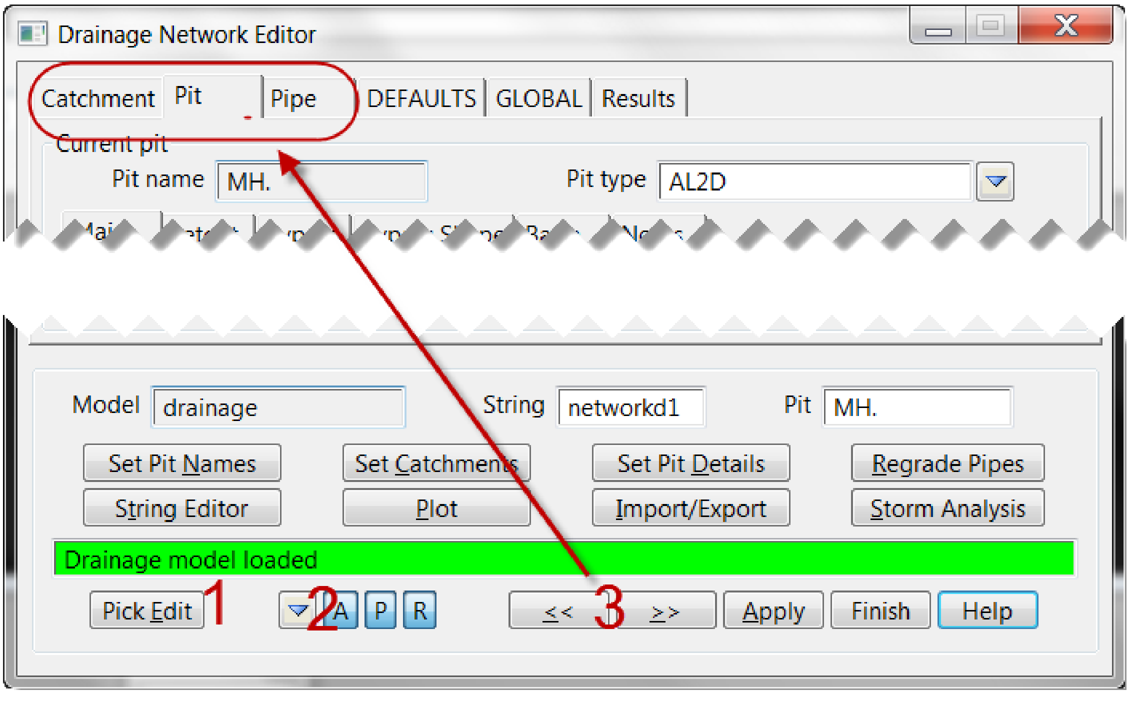
1. Select **Update from drainage.4d.** The manhole diameter will now be changed to 0.93 for this manhole.
2. If you select **Stop asking for this DNE session** first, you will not be prompted before the DNE updates the manhole data.
3. Selecting **Set Pit details** will update all manhole diameters in the network.

A surveyor who received a 12da file of the drainage network without the drainage.4d file would select **Keep stored data**.

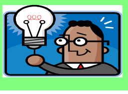
14.2 Moving through the Drainage Network

There are 3 ways to change pits in the network editor. The Next-Previous (chainage NOT flow direction or east west) buttons will work if you are on the **Defaults**, **Global** or **Results** tabs. If the editor finds an error on a pit, it will take you to the error and you must correct the error before moving to the next pit.

Try all 3. The **GoTo List** will be interesting as we have not assigned pit names yet.



1. Pick Edit Button - Pick and pit or pipe in section or plan view
2. GoTo List - Select the pit from a list of pit name
3. Next and Previous (chainage) buttons - loop through all of the pits in the network



The most important button on the DNE is the **Set Pit Details** button. This button calculates the all of the manhole levels and data required for the drainage calculations.

14.3 Auto-Apply, Auto-Pan, Auto-Profile and Auto-Redraw

To tell if these buttons are on or off you need to know your computer display settings. For Windows 7, the default is Blue for on.

With **Auto-Apply** (A) enabled, data is saved when any of the 15 lower buttons are pressed **except Finish** and **Help**.

Auto-Pan is always active and any plan view showing the active drainage network selected will always auto pan if the selected pit is not in the view.

With the Auto-Profile (**P**) enabled, the same will happen for the section view.

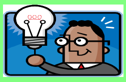
With **Auto-Redraw (R)** enabled, the editor will regenerate the section view when changes are made in the vertical. This saves selecting **Regen** on the section views.

14.4 Set Pit names (and pipes)

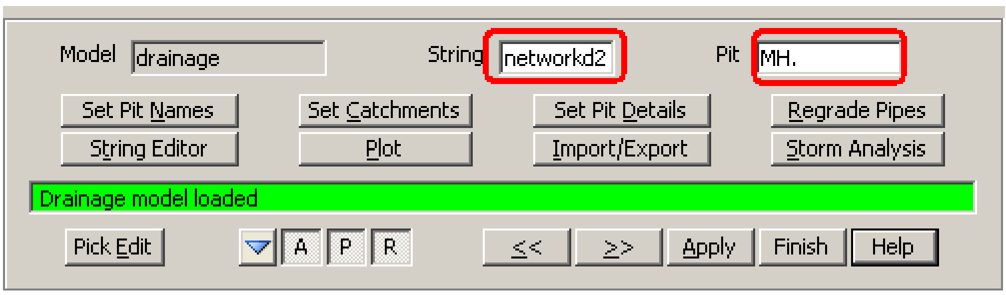
Pit names are used to identify and label the pits for access. The DNE Goto drop down and outlout window loglines (warning/problem messages) become much more meaningful. meaningful. **Storm analysis** requires that unique pit names be set.

Use the DNE to manually change single pit names (in the **Pit** field) or quickly change all the names using the **Set Pit Names,** button.

The **Set Pit Names** button requires unique string names. To view string names on the plan view, go to the Plan View tool bar and select **Toggle=>Names**. If they do not appear see Displaying View Text.

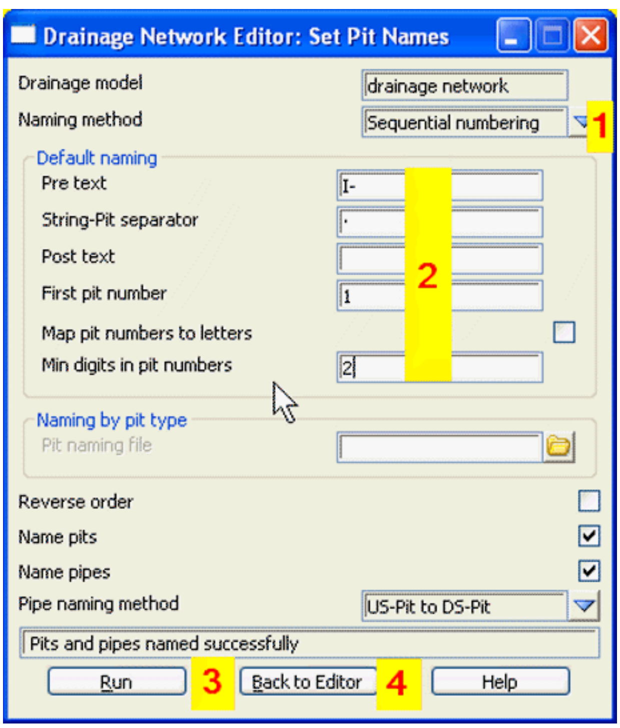


Models->String Info table is a fast was to view all the string names in the model. Double clicking on the lines will launch an editor to change the string names. The DNE also sorts the strings based on these namesand this controls the plotting order.



The string names may be changed in the **String** field when the **Catchment, Pit** or **Pipe** tabs are selected at the top of the DNE. The string names must be unique.

1. Select **Set Pit Names** and the following panel will appear.

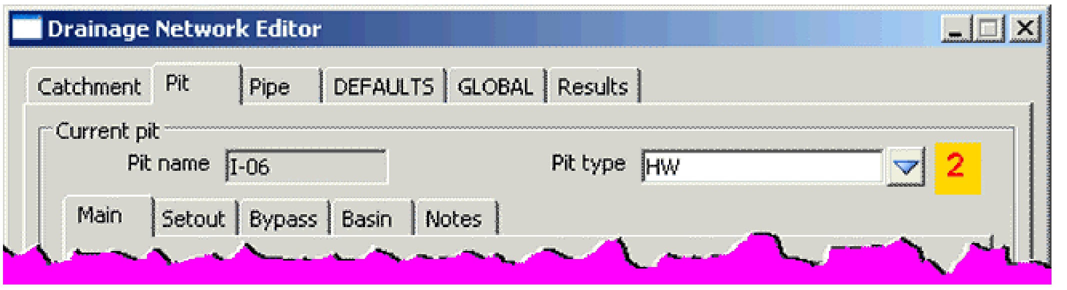


1. Select **Sequential Numbering** to number the manholes starting at First Pit Number.
2. Change the **Default naming** parameters**.**
   Map pit numbers to letters causes the first pit to be A instead of 1.
   Min digits in pit numbers set to 2 causes 1 to be 01.
3. **Run** will update the quick text pit names on the plan view.
   Try other settings!
   You may find you want to change some of the string names back in the network editor.
4. **Back to the Editor** return to the main editor panel.

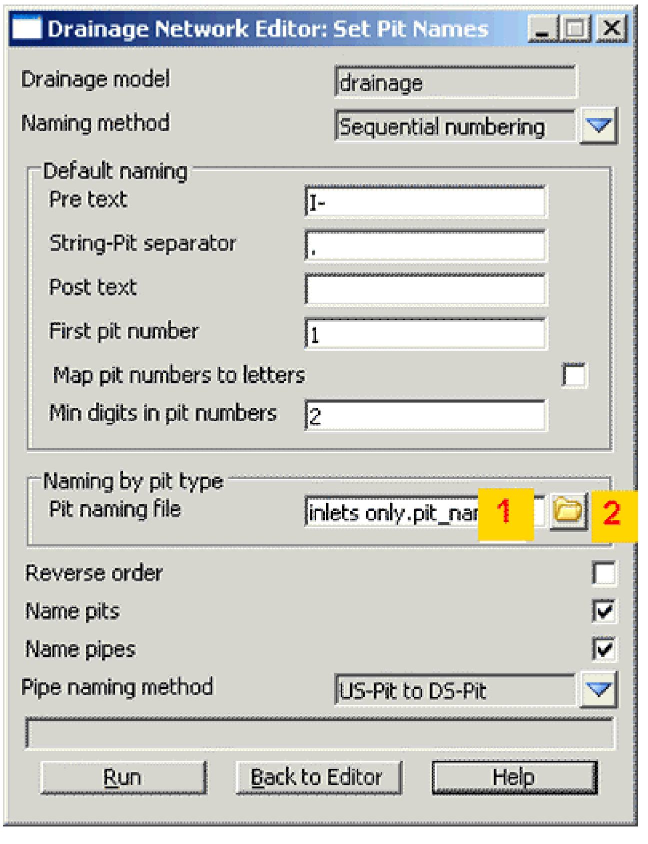
   Reverse order starting numbering at the high chainage end of the string instead of the low chainage.

14.5 Set Pit Names using Pit Type

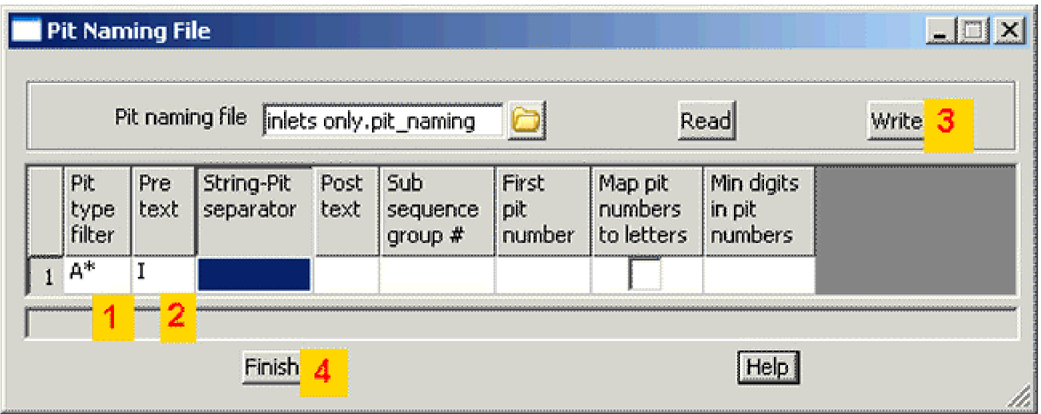
We can customise the pit name prefix, suffix, numbering sequence etc for each pit type. Currently all of the pits have the default pit type. We will now change the outlet pit type to HW and then rename the pits.



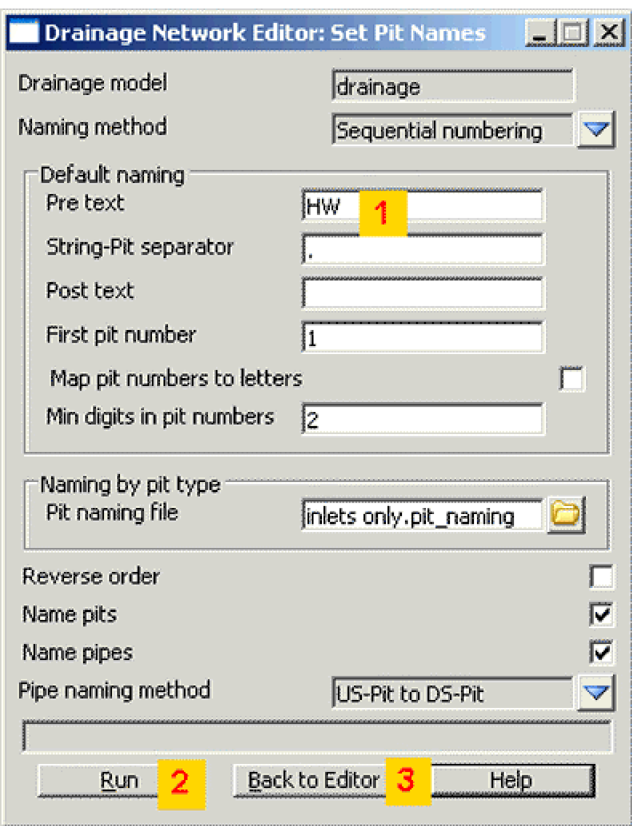
1. Move to the outlet using any of the 3 methods.
2. Change the **Pit type** to HW
3. LB Select **Pit Names**



1. Type the file name for the pit naming scheme. An example may be found in the library if desired.
2. LB the folder icon and select **Open.**



1. RB to select the **Pit type** or type text with wild card characters as shown
2. Type the **Pre text**
3. LB **Write** to save the settings to the file.
4. LB **Finish** to return to the panel.

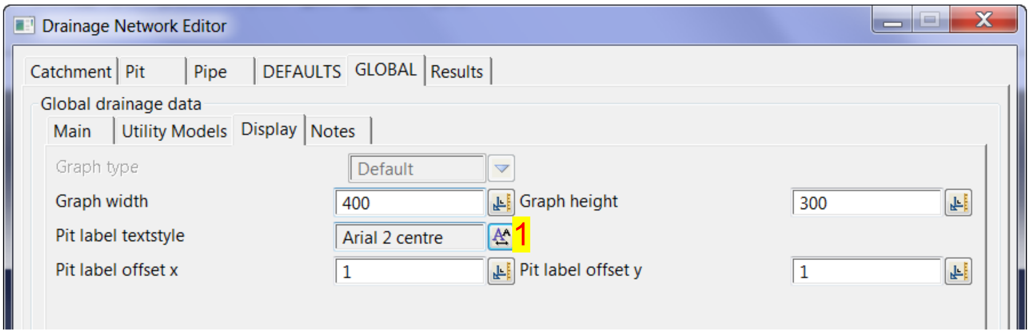


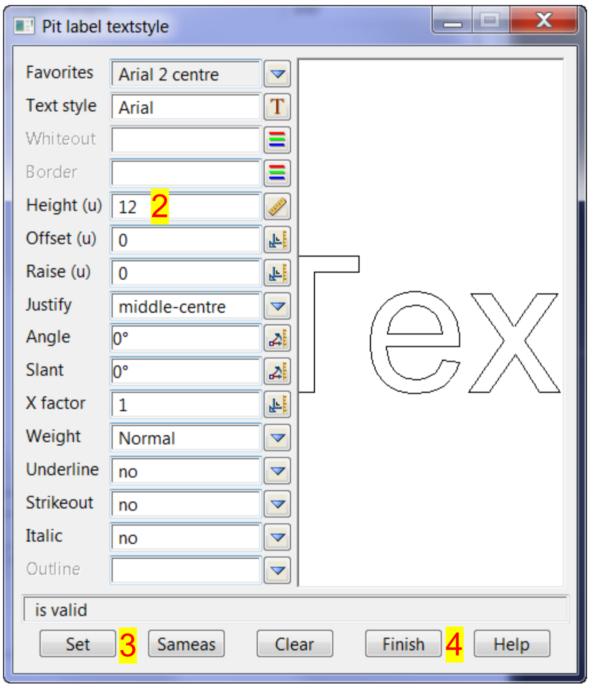
1. Type **Pre text** for those not included in the pit naming file.
2. LB **Run** to name the pits
3. LB **Back to Editor** to return to the main panel.

Note that the headwall still get the sequential numbering continued from the last numbering method. To start at a new number the **Sub sequence group number** would have to be included in the file.

14.6 Change Pit Name Textstyle and Offset

The textstyle and offset for the pit name is set on the **Global->Display** tab of the DNE. It is always shown as pixel text size and cannot be changed to world.



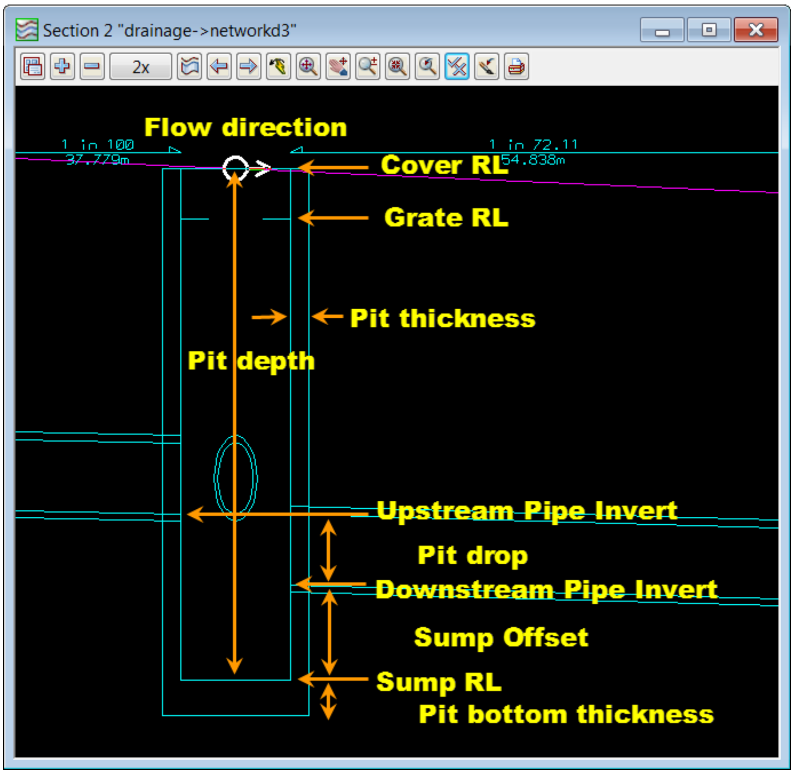


1. Select the **Pit label textsyle** icon and then the **Arial 2 centre** favourite.
2. Type the new **Height (u)** as **12** pixels.
3. Select **Set**.
4. Select **Finish**.
5. Select **Apply** from the bottom of the DNE to see the new textstyles.

15.Vertical Alignment - Manholes

The diagram below shows the properties of a manhole. The RL levels may be set manually or calculated via the their associated mode and the **Set Pit Details** button. The **Grate RL** is used by 12d hydraulics when determining the freeboard level, bypass flows, inlet capacity values at sag inlets and depth of flooding at sag inlets.

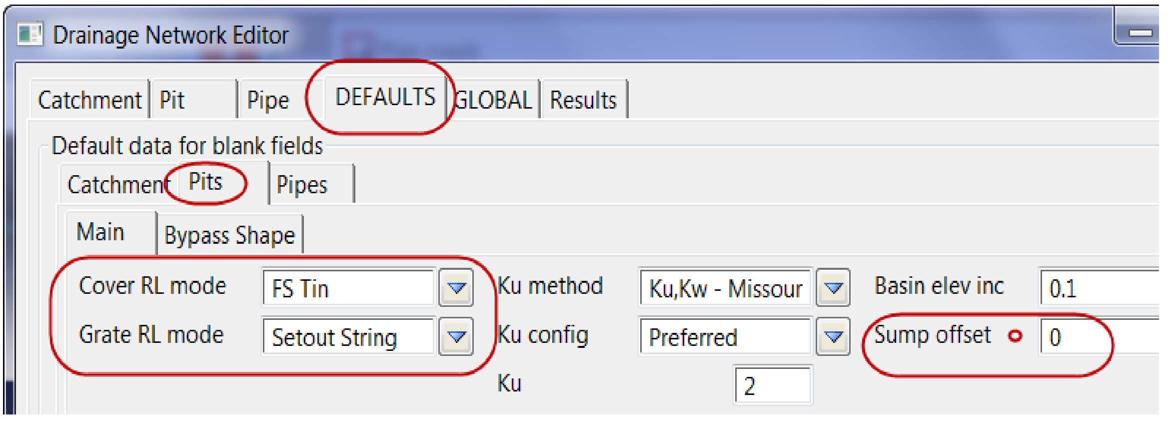
The **Setout RL** is the level used by surveyors to setout the pit.



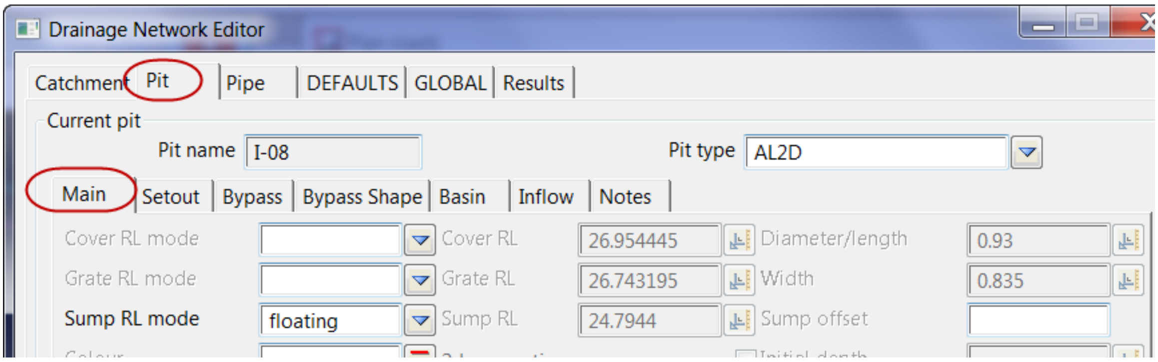
Flow direction	string property
Cover RL	DNE->Pit->Main
Grate RL	DNE->Pit->Main
Setout RL	DNE->Pit->Setout (not shown)
Pit thickness	drainage.4d via type
Pit depth	calculated value
Upstream pipe invert	DNE->pipe->design
Pit drop	Defaults(manhole) or DNE->Default->Pipes Drop file
Downstream pipe invert	DNE->pipe->design
Sump offset	DNE->Pit->Main
Sump RL	DNE->Pit->Main
Pit bottom thickness	drainge.4d via type

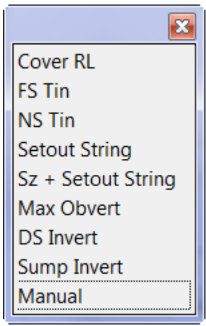
15.1 Cover RL, Grate RL, Setout RL and Sump RL modes

The default values for the RL modes are set on the **DNE->Defaults->Pits** tab.



These may be explicitly for an individual pit on the **DNE->Pit->Main** and Setout tabs





RL Modes

Cover RL Only available for Grade and Setout RL modes. z value will be the same as the cover RL.

FS Tin z value from the pit centre x,y location on the DNE->Global->Main->FS tin.

NS Tin z value from the pit centre x,y location on the DNE->Global->Main->NS tin.

Setout String z value from the string found by dropping the pit centre x,y location perpendicularly onto the setout string (DNE->Pit->Setout->Setout string) and then moving the distance DNE->Pit->Setout->Setout adjustment Sxy.

Sz+Setout String zvalue as for the **Setout String** plus the value DNE->Pit->Setout->Setout adjustment Sz.

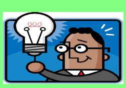
Max Obvert z value of the maximum pipe obvert of all the connecting pipes.

DS Invert z value

Sump Invert Only available for Setout RL. z value from DNE->Global->Main->Sump RL

Manual z value is entered in the RL field to the right of the mode (field becomes active when this is selected).

floating Only available for Sump RL mode. z value is of the minimum pipe invert of all connecting pipes plus the value DNE->Pit->Main->Sump offset

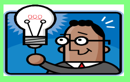


RL modes for the entire model may be set via
Design->Drainage-Sewer->More->Pipe Locks

This routine has lock modes for pipes as well.

16. Vertical Alignment - Pipes

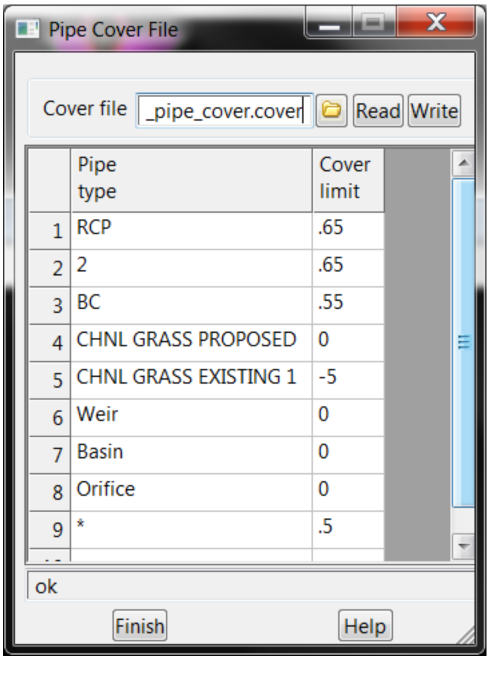
The pipe invert levels may be set manually or calculated using the DNE->Regrade Pipes button.



Pipe invert mode may be set to locked for the entire model via **Design->Drainage-Sewer->More->Pipe Locks**

There are many controls to guide 12d in the setting of the pipe inverts and the user may over constrain the network so that no solution is possible. If the minimum cover, grade or drop criteria cannot meet the criteria you have set, a problem message will be in the output window.

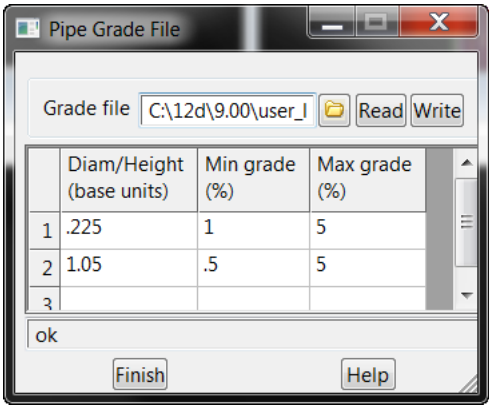
16.1 Minimum Cover



The minimum cover can be set via the pipe type and the pipe cover file (optional). If the pipe type is not found in the pipe cover file then the Defaults-Pipes-Cover for the project is used.

16.2 Minimum Grade

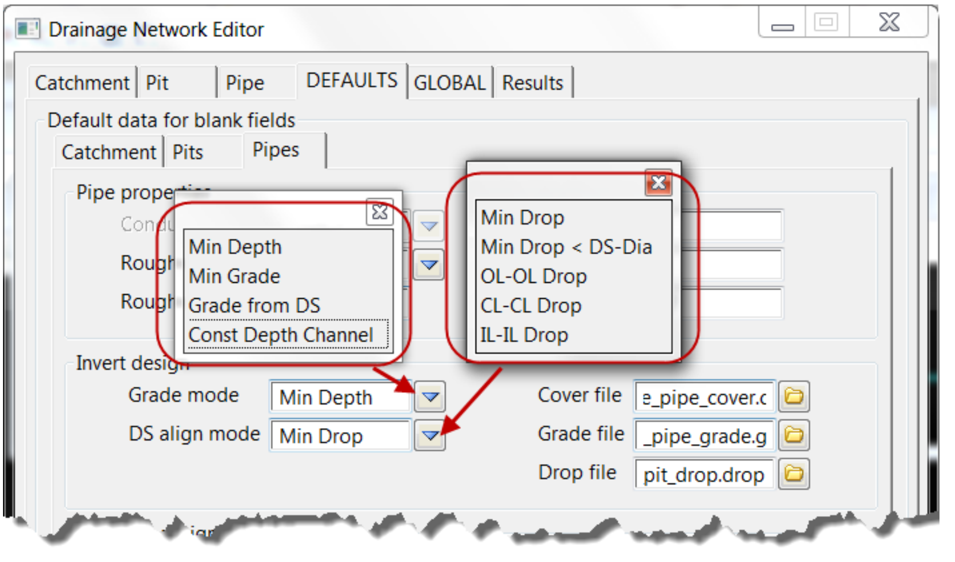
Pipe grade has been traditionally measured based on the horizontal distance from pit centre to pit centre. However, this causes issues with surveyors when they use electronic models to set out the drainage pipes. All drainage pipes lengths can now be calculated from the horizontal distance between inside pit edges (or pit connection point for the unrestricted pit connection mode). The alternate pipe length results in a different pipe grade to be calculated in 12d. This option is set one the **DNE->Global->Main Use Actual Pipe Lengths**.



The minimum grade can be set via the pipe diameter and the pipe grade file (optional). The diameter is the maximum diameter that this grade will be used for. If the diameter exceed the greatest diameter in the grid, the last grade will be used.

If the pipe grade file is not used then the grade from the Defaults-Pipes-Min grade for the project is used.

16.3 Pipe Grade Modes



Minimum Depth The cover sets the inverts at each end of the pipe and then the downstream invert is lowered if the minimum grade is not achieved.

Minimum grade The pipe is set to the min grade and then lowered to meet the cover requirements.

Grade from DS Warning! Minimum cover may not be maintained in this mode. Each pipe has its downstream (ds) invert level set via its **DS pipe alignment** mode. If the downstream pit is the outlet then the cover is used to set the invert level. The upstream invert is set using the minimum grade.

Const Depth Channel Warning! Minimum cover, minimum grade and pit alignments are not used in this mode. The obvert of the pipe is set to the finished tin level at the pit centre.

16.4 Downstream Alignment Modes

The amount of the drop is determined first from the **Drop file** and then if required from the manhole default settings for the project. How the drop value is applied is set via the **DS align mode**.

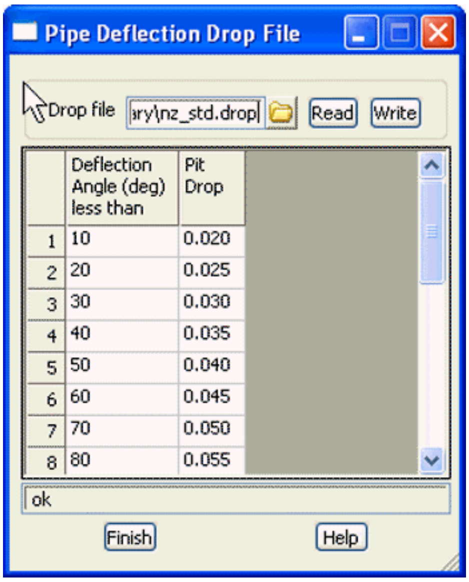
Inverts are moved down, if required, according to the setting in **DS align mode.**

Min drop ensures that the inverts drop a minimum of the drop but may be more.

Min drop<DS-Dia ensures that the inverts drop a minimum of the drop but may not be more than the downstream pipe diameter. This restricts the max drop so that the water jet will not completely impact the opposite wall of the pit.

OL-OL Drop uses the drop value for the obverts. If the downstream pipe is a smaller, then the drop will be applied to the inverts so that water will not be trapped in the pit. If the downstream pipe is larger then the obverts will be aligned as long as the min drop is maintained on the inverts.

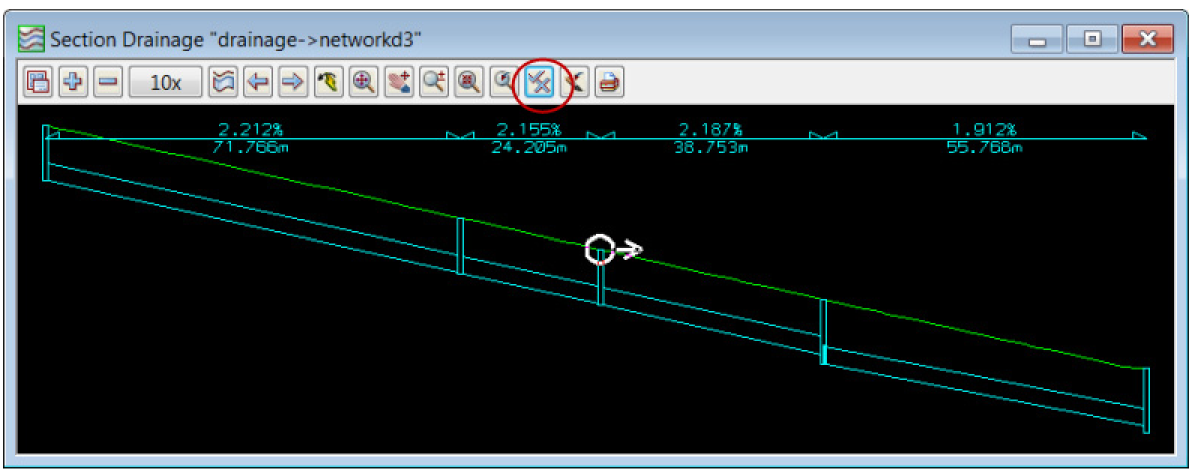
IL-IL Drop uses the drop value for the inverts.



The **Drop file** contains the drops that change with the pipe deflection angle. A sample file is included in the library.

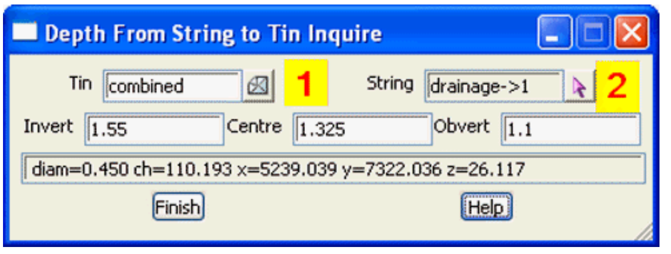
If only one drop value is to be used set the deflection angle to 180 degrees.

After setting the pit and pipe levels, add the model **tin combined** to the **Drainage** section view. The trunk profile will be similar to the following. Select the toggle button, to toggle on the grades (and pipe lengths).



To inquire about the cover on a pipe use the following option.

Tins->Inquire->Depth from string



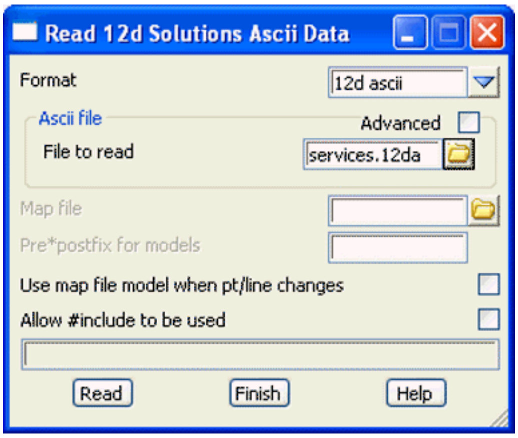
1. LB to select the tin
2. LB to select the string.
3. Move the pointer in plan or section to monitor the data.

The DNE calculates cover from the top of the pipe NOT the obvert so cover may appear incorrect. Also for box culverts, cover is measured at the edges as well as the centre of the pipe. Finally for multiple pipes, each pipe is checked for cover.

17.Service and Utility Clashes

We will import some services from a 12da file. From the main menu select,

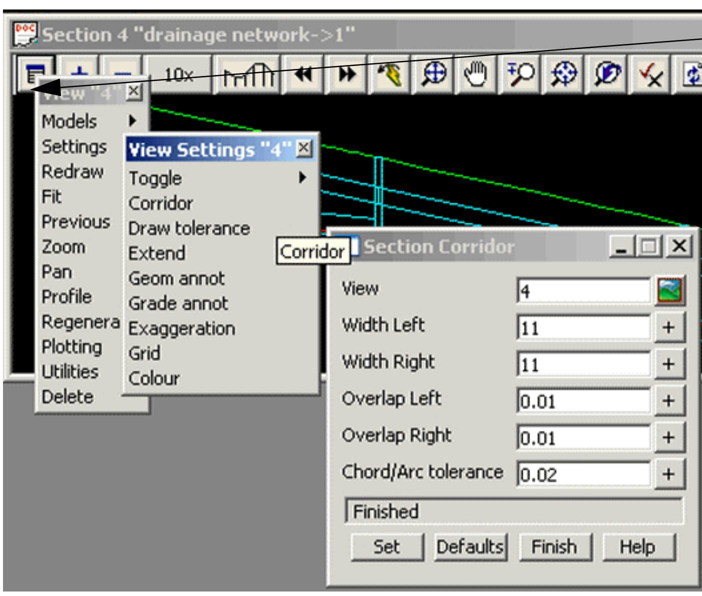
File I/O->Data Input->12da/4da data



1. LB the **Folder icon**
2. Select the service file
3. LB the **Read** button.
4. LB the **Finish** button.
5. Add the model **services** onto the plan and section views.

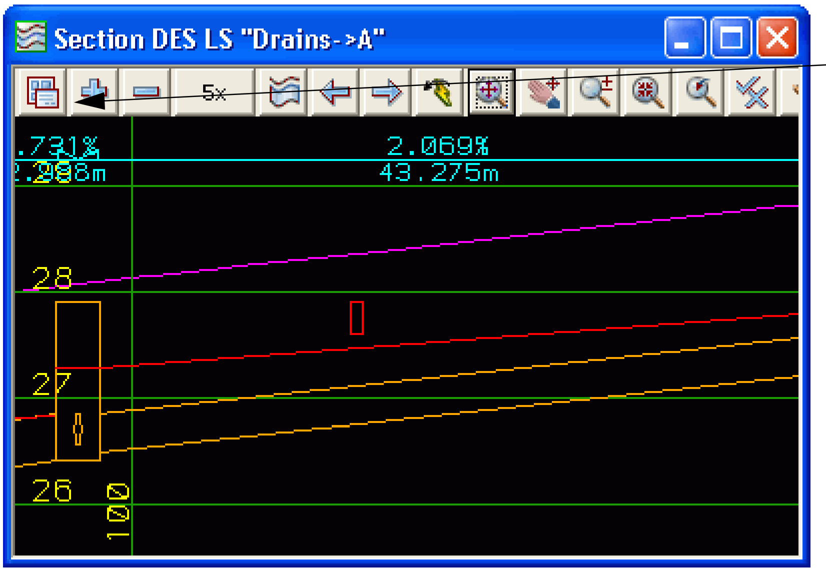
12d service clash routines notify the user of crossing services but not parallel services that are close to each other. The clearance values are vertical distances at the centre line of the drainage and service strings. The minimum clearance may be less than the vertical clearance if the drainage or services are on very steep slopes.

To view parallel services, add the services model onto a section view, profile a drainage string and then set the corridor value for the section view.

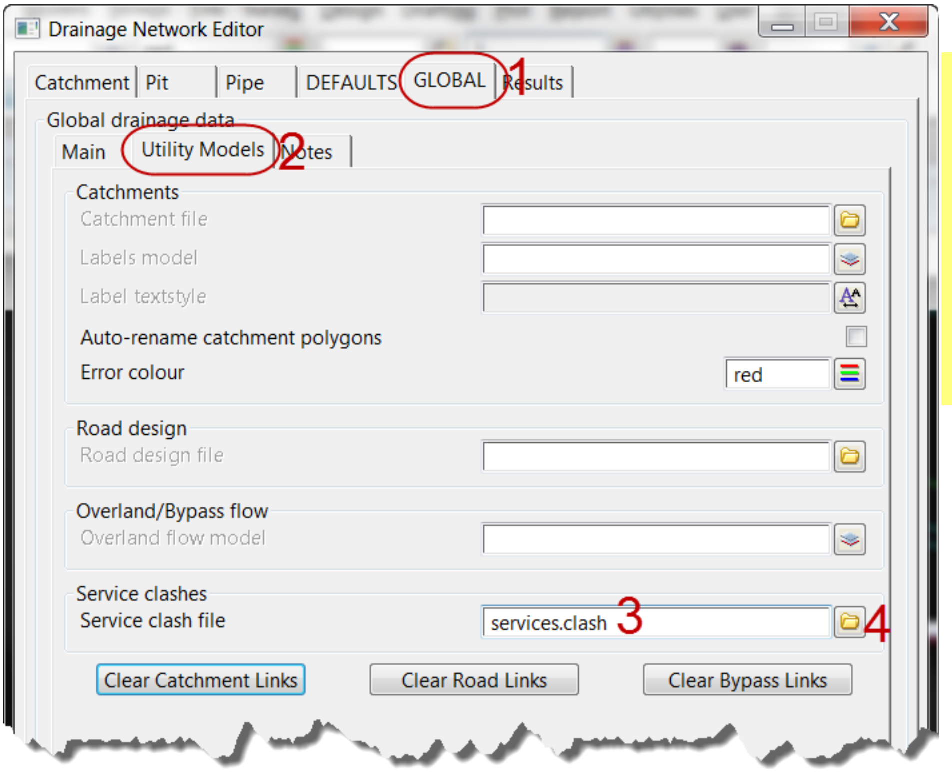


Settings->Corridor and then set the **Width left** and **Width Right** to the desired clearance. If the service can be seen then it is within the tolerance. 11 is used in this example only so that you can see the service on the other side of the road.

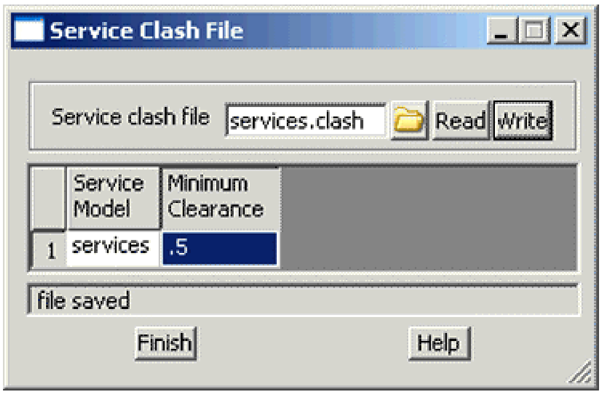
Use the **Next** and **Prev** button to switch drainage strings.



To obtain a report of all strings inside or crossing the drainage string profiled, select the **View menu** button then **Utilities->Report**

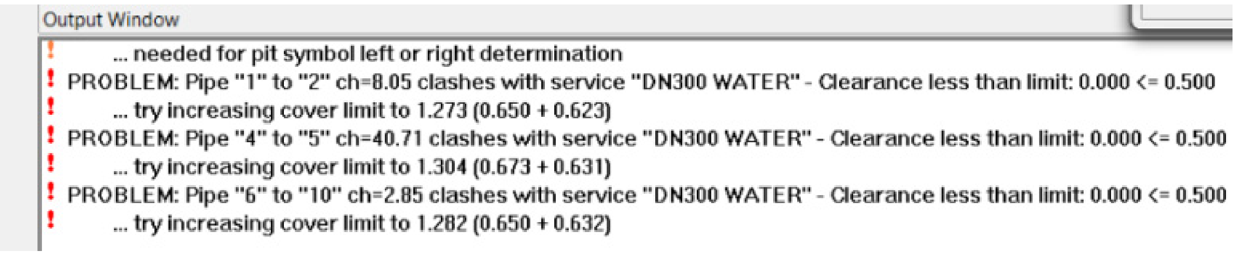


1. Select the **Global tab**
2. Select **Utilities Model** tab
3. Type **services** to create a new services file.
4. Select **Open** from the icon list.

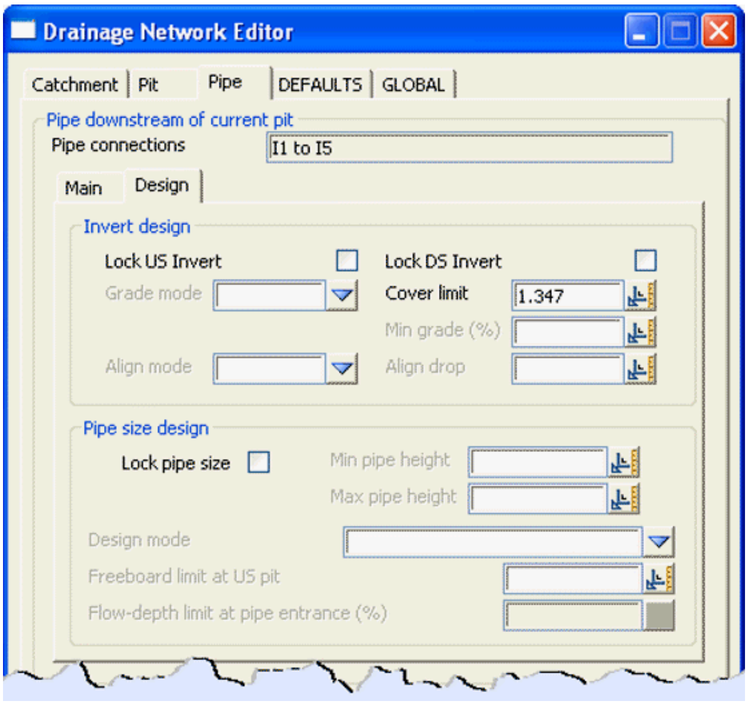


In the **Service model** column **RB** to select the model. Enter **Minimum Clearance** for the services in this model. Include an extra amount for the thickness of the drainage pipe. If the service model contains other drainage/sewer strings you must add the thickness of these pipes as well.

If different clearances are required for different services then place the services in different models. Warnings will be issued when you **Regrade Pipes, Set Pit details, Import** or **Storm Analysis**. Cover levels or fixed inverts can be used to avoid the services.



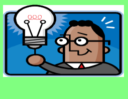
To quickly move to the section view of the string, Double click on the message in the output window (note the red! mark). The DNE will move to the pipe and with auto profile button enables the section view will update to this string.



The most common method to avoid the clash is to increase the **Pipe cover limit** for this pipe segment so that the pipe is pushed down.

If the clash problem is above the pipe then the **Max pipe height** may be used and multiple pipes are selected.

This method is preferred over locking the inverts as this leaves more flexibility for aligning the inverts.



Often changing the **Grade mode** to **Min grade** on a branch line will raise the downstream invert. This may allow the entire trunk line to rise and thus reduce excavation costs. This is especially true if the service clash is near the upstream end of the pipe.

Once the invert levels have been reset by selecting **Regrade pipes,** the output window will indicate the final clearance.

After a pipe design run in **Storm Analysis**, details of the service clash data will again be listed in the output window.

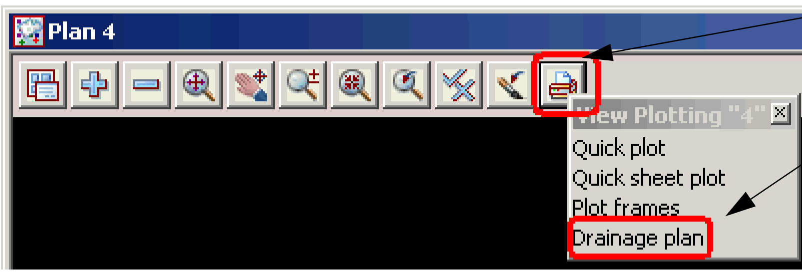
18.Drainage Plan Plots

The drainage plan plots create detailed drawings with symbols for the pits, linestyles for the pipes and string names ready to export to the desired layers in CAD. Pipes levels are 3d and setout points ready for downloading to survey total stations are prepared.

18.1 Labelling the Pits and Pipes

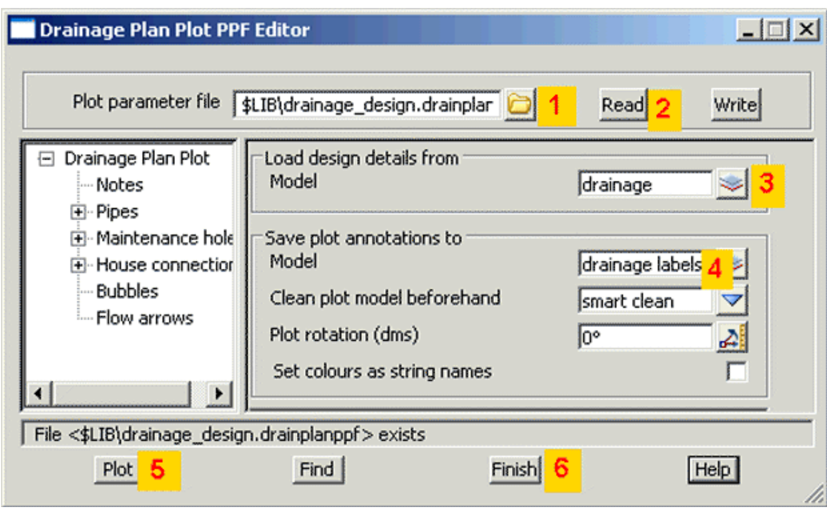
Plan plots are used to label the pits and the pipes. Drainage Plan Annotations may be accessed from one of three locations. The third is the most common.

Location 1: From the plan toolbar



Select the **plot** button and then **Drainage plan**

Note (The following panel has been reduced in size).

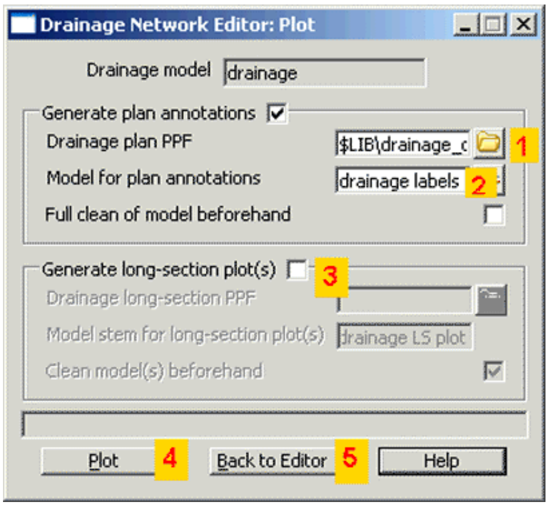


1. Select a ppf file from the library
2. LB **Read**
3. Select your drainage model
4. Type a model name for the new labels.
5. Select **Plot**
6. Select **Finish**

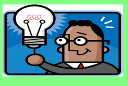
Now add the drainage labels model onto the plan view.

Location 2: The **Plot Button** on the **Drainage Network Editor**

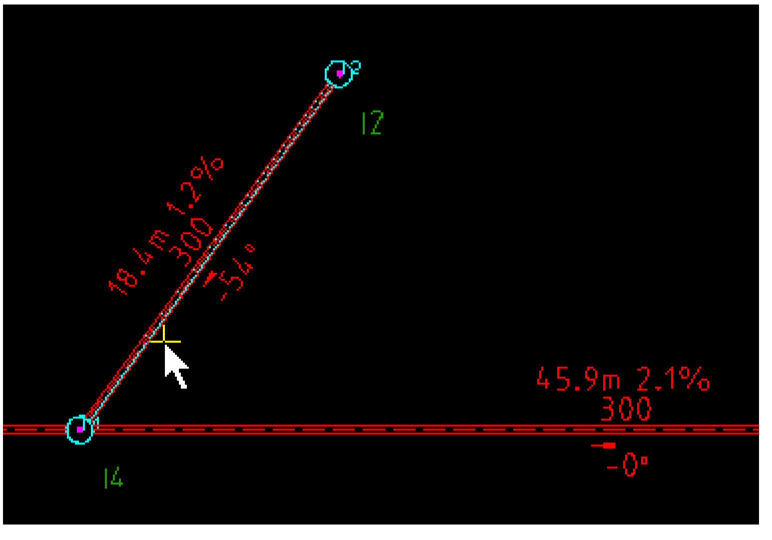
The following panel will be displayed. This option can plot both the long section and plan at the same time. Select a ppf file from the library (drainage_design).



1. Enter a model for the plan annotations.
2. Select **Full clean model beforehand** if you have not manually moved any of your manhole labels.
3. Turn off the long section plot for now.
4. Select **Plot.**
5. Select **Back to Editor**
6. Now add the **Model for plan annotations** onto the plan view.
7. Note: if you rename the manholes you will need to replot these labels!



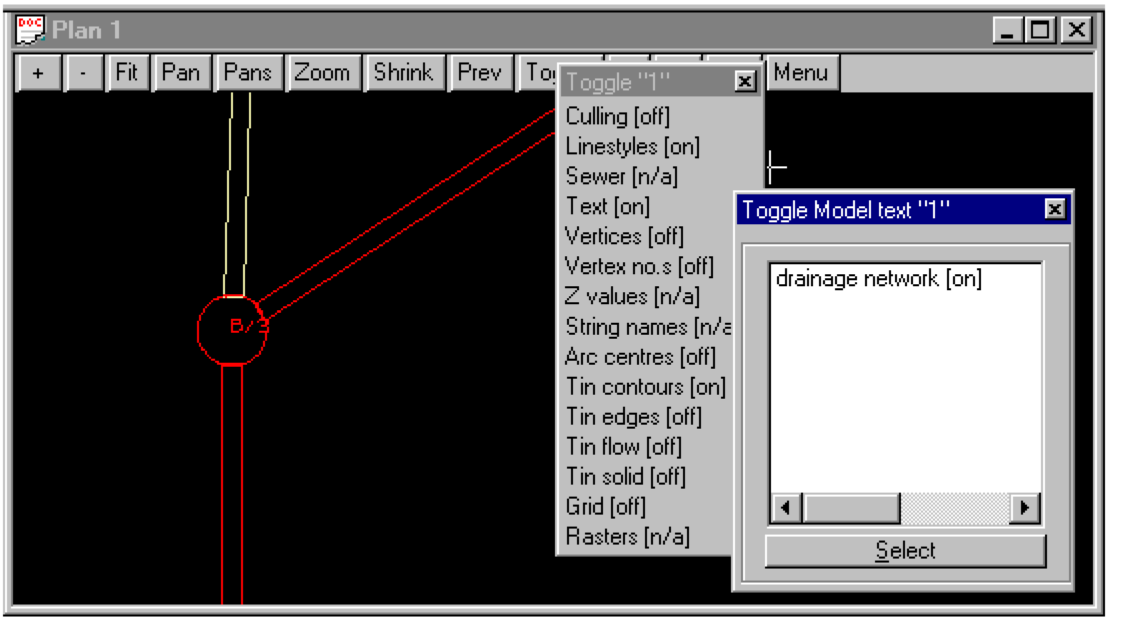
Note that **Full clean of model beforehand** was not selected. The drainage plan plot does a smart clean where it deletes and reprints text that has not been manually moved and updates text that has been moved.



The text properties can all be customised using the plot parameter file but this will be discussed later in the plotting section. These labels are **not** automatically updated when you change the names or pipe diameters. You must rerun the plot routine to update the labels.

18.1.1 Turn off View Text Pit Labels

To turn off the automatic view text pit labels for this view select Toggle=>Text and then walk right to select the drainage model. Do not click on **Text**, rather walk right. If you click **Text** you will toggle on/off all of your text on the view, not just the drainage model.



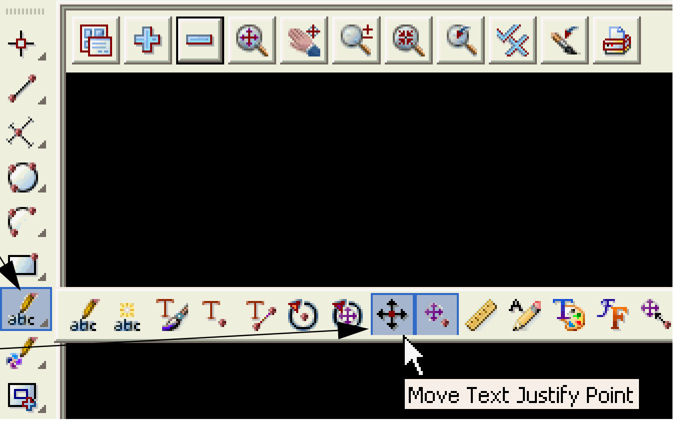
18.1.2 Moving Text

The labels created may be manually moved using the **CAD toolbar** but if the model is relabelled the text will return to its original location! Text moved via the **Drafting->Multi string translate** will remain in the moved position when **Smart Clean** is selected in the Plan Annotation panel.

Before selecting text turn on your **teXt snap.**

To move a single line a text use the **CAD** toolbar. Select the **Move text justify** button. To use this toolbar you must **DRAG** the **Create text** button to the right

and then release when the pointer is on the **Move Text Justify Point.**



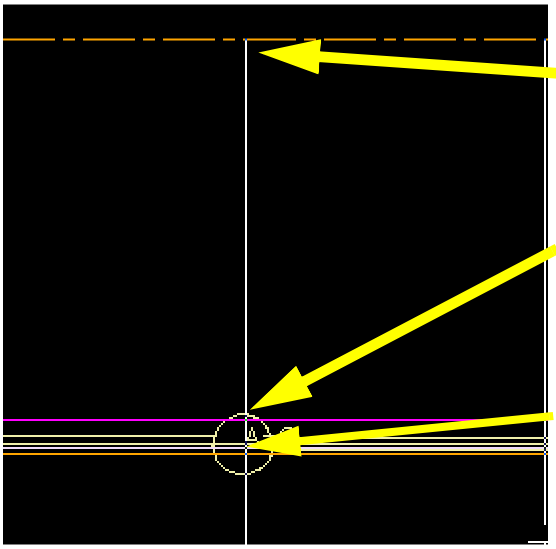
To move a pre-defined **Group** of text select

Drafting->Multi string translate

Select **Group** and then pick and accept one of the text items in the group. Move it to its new location and accept.

19.Construction Setout

Pits are setout on the construction site using a variety of techniques. The DNE creates a construction setout point (x,y,z) that can either be located at the pit centre or the pit centre can be dropped perpendicularly onto a design string for the xy location. The z value for the setout point has a number of modes as described below. Road centre line chainage may also be calculated



Road Centre Line

Road Chainage and offset

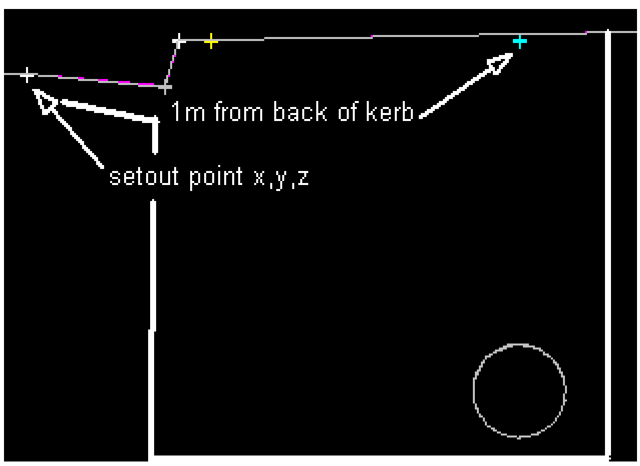
Setout String

x y from setout string

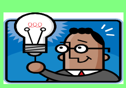
Pit Centre

x y at pit centre

In the diagram below the setout point is lip of kerb. The setout x,y location level z will be obtained from the setout string and the pipe will be shown at its proper position so that the pipe cover is calculated correctly.



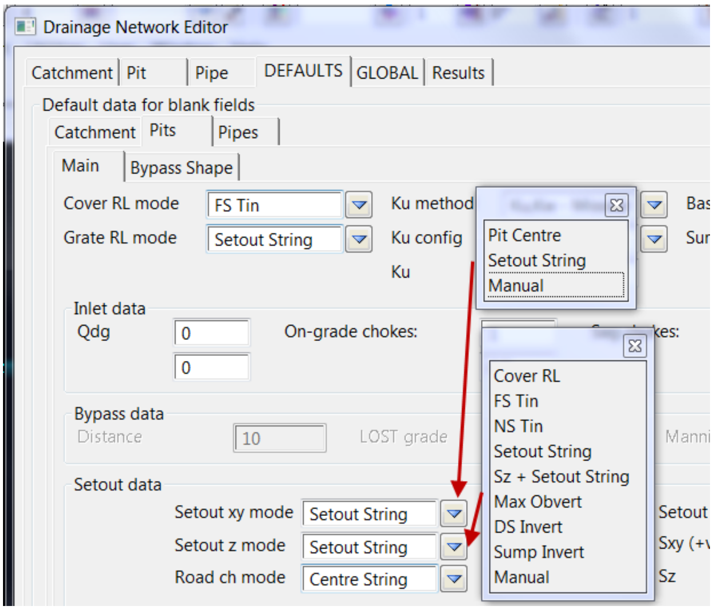
Pipes are setout using the pit connection points and the invert levels.



Design->Drainage-Sewer->Convert to Pts and Lines will create 12d super strings with the pipe attributes on them.

Setout reports can be created and the drainage plan plot generates strings for the surveyor to download to the instruments.

19.1 Pit Setout (xy)



Setout xy mode - Pit centre is the centre point of the manhole (the intersection of the joining pipes). Often the setout point for a manhole or catch basin is not the centre of the manhole but rather a point on the kerb or back on the foot path. The **setout string** option will drop the centre point of the manhole onto the closest string in the **Road design** model list specified on the **Global->Utility models** tab.

Setout z mode options are listed in the panel. This levels are used in the pit schedules.

There are currently 9 modes for determining the pit setout levels.

Cover RL - this ensures the level is the same as the Cover RL (not available for Cover level).

FS tin -The level is obtained from the finished surface tin at the pit centre.

NS tin -The level is obtained from the existing surface tin at the pit centre.

Setout String - The pit centre is dropped perpendicular onto the setout string and the level is obtained from the elevation on the string.

Sz + Setout String an offset is added to the level from Setout string described above.

Max Obvert- all connecting pipes, usually open channels, are checked for the highest obvert.

DS Invert - the invert of the outlet pipe (there is only one).

Sump Invert - the lowest pipe invert plus the **Sump offset.**

Manual - The user must manually type in the cover levels (rarely used as a default).

Explicit settings for the setout strings and the auto calculated values are found on the **Pit-Setout** tab. If the **manual** mode is selected the Easting and Northing locations may be picked in plan view or typed into the input boxes.

19.2 Pit Setout (z) Level

The setout level defines the level to be printed in the pit setout tables and in the drainage longsection plots. The **FS Tin** selection obtains the level from the FS tin, specified on the **Global-Main** tab, at the centre point of the pit.The **Setout String** location obtains the z level from the setout string as described in the section above. Explicit settings and the auto calculated value are found on the **Pit-Setout** tab.

19.3 Road Centre Line Chainage

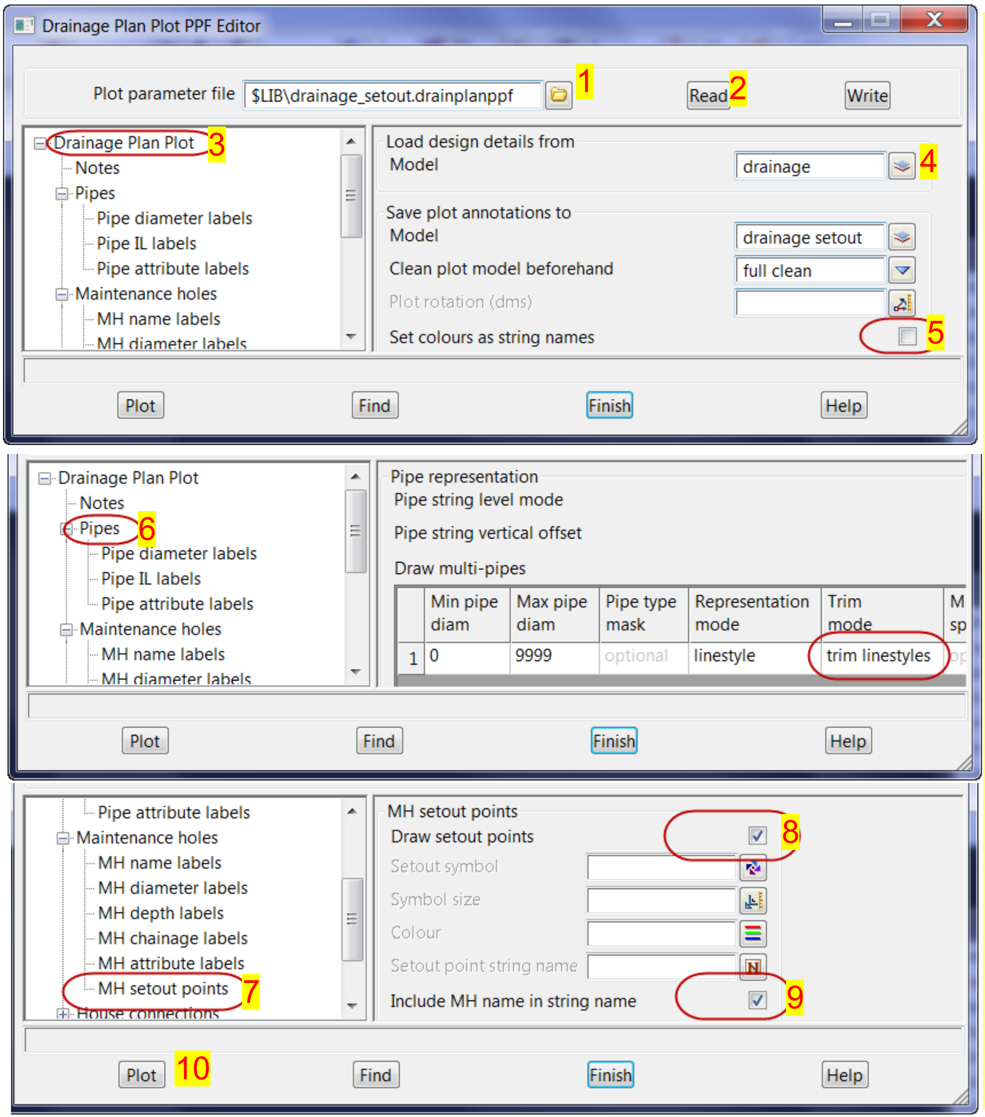
If **Road chainage mode** is set to **Centre string**, then the **Centre String ID** in the **Road design file** (shown above) is used to select the road string to measure the chainage and offset from. The values and explicit settings for the road chainage and offset are found on the **Pit->Setout** ta

19.4 Pipe Setout

Pipe setout is along the centre line of pipes.

19.5 Plan Plots for Surveyors

Use the drainage plan plot routine to create survey setout point for each pit and strings for each pipe with the invert levels. If **drainage_setout** is not in your library, start with **drainage_inverts**.



1. Select **drainage_setout** ppf from the library
2. LB **Read**
3. LB **Drainage Plan Plot** branch
4. Select the existing **drainage** model
5. ensure NOT selected
6. Select **Pipes** branch and note **Trim mode**
7. Select **Maintenance holes->MH setout points** branch
8. Note:**Draw setout points** is selected.
9. String names are sent to the survey instruments so select this.
10. Select Plot
11. Add **drainage setout** model onto a plan view

19.6 Construction Manhole/Pit Schedules

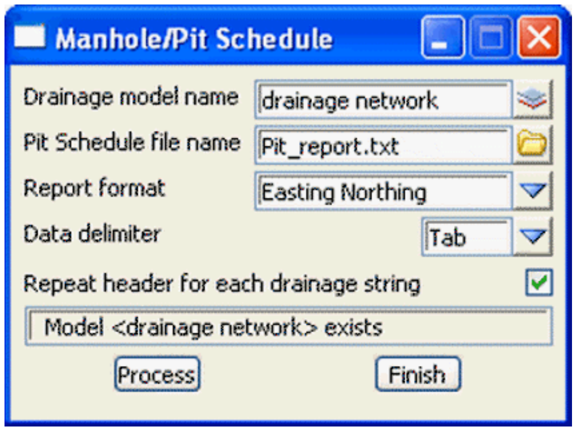
Manhole/pit schedules or construction tables are generated in tab or space delimited formats.

This routine does not recalculate the setout locations. Similar to plotting, it prints the results of the last time the calculation were done (DNE->Set Pit Details).

Position of option on menu: **Design => Drainage => Reports => Pit schedule**

This routine prints the calculations from the last time Set Pit Details was selected in the Drainage Network Editor.

On selecting the **Pit schedule** option, the **Manhole/Pit Schedule** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Drainage model name	input box	drainage network	
model containing the drainage strings			
Pit schedule file name	input box	pit report	
file to be created			
Report Format	choice box	Road chainge.,Easting...	
file format			
Data delimiter	choice box	Tab, Space	
tab delimiters are best for spreadsheets and space for some text editors			
Repeat header for each line	tick box	selected	
when selected, the column headings will be printed each drainage line			
Process	button		
Create the pit report			
Finish	button		
remove the panel from the screen			

Notes:

The columns of data may be separated by spaces or a tab. (tab is used for spreadsheet transfers). The internal width and length data are retrieved from the **drainage.4d** file for the pit type specified. If you want a longer description for the pit then the type used inside 12d this can also be entered in the drainage.4d file. The remarks for each pit are entered as user defined pit attribute named **remarks** and may be set using the attribute editor (on the drainage menu) or via a spreadsheet.

Easting Northing Sample

```
.PIT SCHEDULE
  Pit                                  INTERNAL          INLET         OUTLET                    PIT
 No    TYPE    EASTING NORTHING    WD       LEN     DIA   INV LEV   DIA   INV LEV  FIN RL    DEPTH        REMARKS
  B1    SA2    5302.458 7336.936  450.000  900.000                  375   28.210   29.387    1.177
  A2    SA2    5264.372 7322.036  450.000  900.000    375   27.470                 28.646    1.226
  C1    SA2    5224.155 7336.936  450.000  900.000                  375   26.690   27.863    1.173
  A3    SA2    5187.910 7322.036  450.000  900.000    375   25.930                 27.158    3.628
  A1    SA2    5309.458 7321.100  450.000  900.000                  225   28.550   29.577    1.027
  A2   SA2     5264.372 7322.036  450.000  900.000    225   27.470   375   27.420   28.646    1.226
  A3   SA2     5187.910 7322.036  450.000  900.000    375   25.930   375   23.530   27.158    3.628
 A4    SA2   5157.411 7321.332  450.000  900.000   375   23.090                  26.714    3.624 outlet to existing
```

```
system
NOTE:
1. ALL SETOUT POINTS QUOTED TO CENTRE OF PIT
```

Road Chainage Offset Example

```
DRAINAGE LINE A
PIT                         PIT LOCATION                  LOCATION OFFSETS
No.                         EASTING       NORTHING      STATION       CTRLOFFSTYPEREMARKS

A/1                         5354.629      7336.936      231.171       d002-7.450
A/2                         5340.691      7320.911      217.233       d0028.575
A/3                         5293.458      7320.886      170.000       d0028.600
A/4                         5250.131      7320.886      126.673       d0028.600
A/5                         5217.194      7322.036      93.736        d0027.450
A/6                         5183.458      7322.036      60.000        d0027.450
A/7                         5152.699      7322.036      29.241        d0027.450
```

Notes

The Set pit details must be run at least once to before printing the report. If the pits are moved or the designed strings changed then this option must re rerun.

The easting northing data obtained for the **road design string** option is obtained by dropping the pit centre perpendicular onto the selected road design string. This data is stored as pit attributes **setout x** and **setout y**. It is calculated when the **Set Pit Details** is selected in the Drainage Network editor.

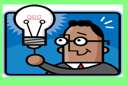
20.DNE and Rational Hydrology

The rational methods uses the formula Q = CIA/360
where

A = catchment area (hectares)
C = runoff coefficient
I = rainfall intensity (mm/hr)
360 conversion factor to m^3/s for area (ha) and rainfall (mm/hr)

The rainfall intensity requires the input/calculation of the time of concentration (tc) for the catchment and then a return period to be used in the IFD table (see 12d Rational Method Hydrology - Drainage Rainfall Editor).

The total area entered/measured may be split into impervious and pervious and analysed separately.



If you use a single composite C value for your catchments, enter a %impervious of zero and ignore the impervious settings.

Many authorities increase the C values for major storms. If you do not then enter the same C for minor and major. Similarly Tc values are sometimes reduced for major storms so you have the option to enter a minor and major C value. An example is provided below.

Total Area = **0.1**		%imp=**80**	Cminor	Cmajor	Tc minor	Tc major
Impervious	Area*%imp		**0.9**	**0.92**	**5** min	**5** min
Pervious	Area-Imp area		**0.3**	**0.31**	**10** min	**10** min

20.1 Network Editor - Global, Default Settings and Explicit Settings

This section will discuss the **Global**, **Defaults** and explicit setting for the hydrology parameters. Design values for the hydrology and hydraulics calculations are set

s either globally (one value for the entire network)

s or via defaults.

Defaults values may be overridden by explicit settings found on the top level **catchment, pits** or **pipes** tabs. Explicit over ride catchment settings need only be specified if the default value is not desired. Only fill them in if you want to use a value different than the default. The tool tip will indicate the default values.

Default values must be entered for **all** of the fields for Set 1. Set 2 and Set 3 blank default values will use the defaults you have used in Set 1.

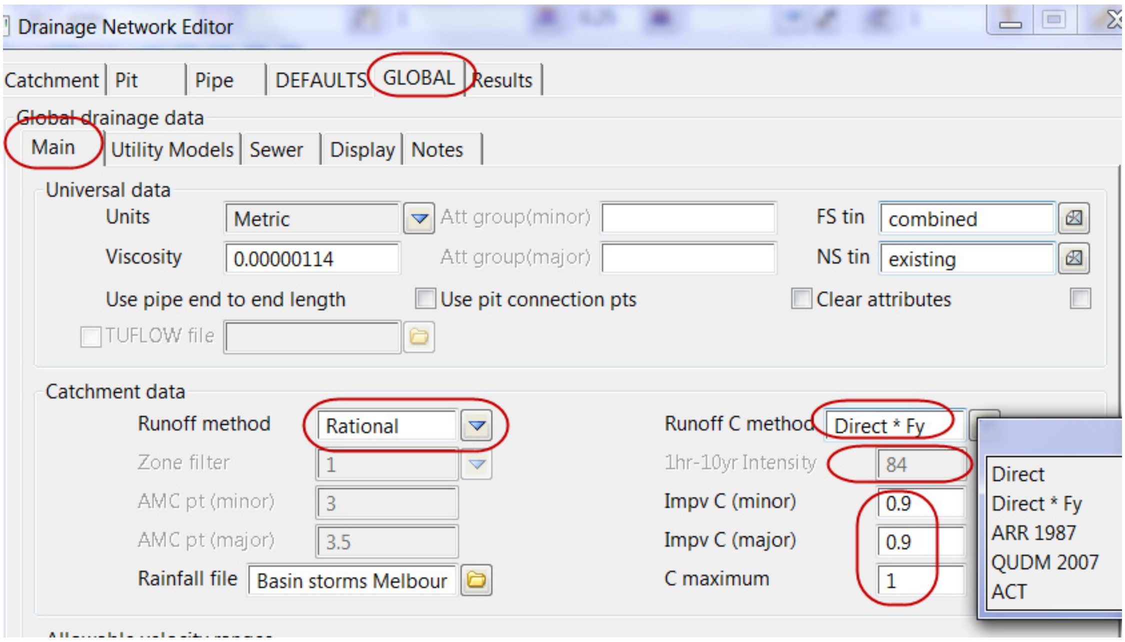
20.1.1 Catchment Areas and Percent Impervious

There is no default catchment area to apply to all catchments. You may type in the catchment area or create a string where 12d will measure the plan area (Drawing Catchment Strings in 12d).

The percent impervious will split the catchment area into the pervious and impervious sub catchments. Separate C and tc values are used for the pervious and impervious areas. The default percent impervious for the various sets are entered on the **Defaults->Catchment Defaults** tab and the explicit settings are on the **Catchment** tab. The percent impervious is used also used to determine the composite C value if using the **ARR 1987** method for calculating runoff coefficients.

20.1.2 Coefficients of Runoff

Global Settings



Runoff C methods include **Direct, Direct * Fy, ARR 1987, QUDM 2007** and **ACT.**

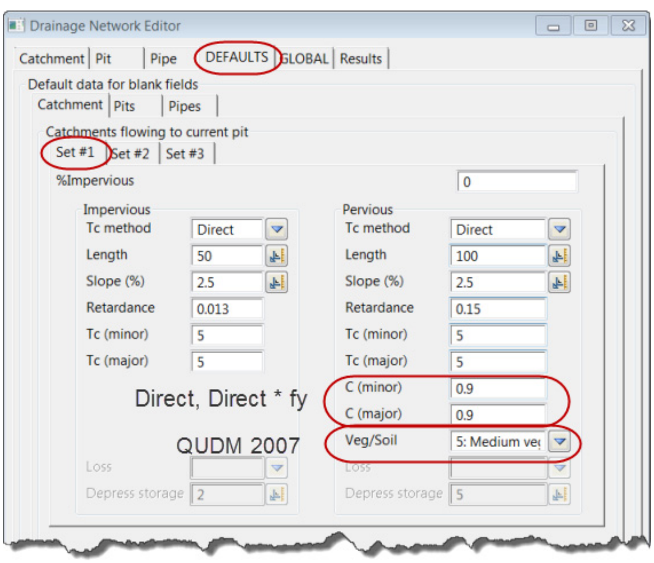
Direct: There is a global impervious C value for minor and major storms. This can only be set once. The C values for the pervious areas may be changed for every catchment in your model.

Direct * Fy: For urban areas in Australia the flood frequency factors (fy) are constant for urban hydrology. You are required to enter the f10 values, the same value for minor and major. For rural areas the direct method is required.

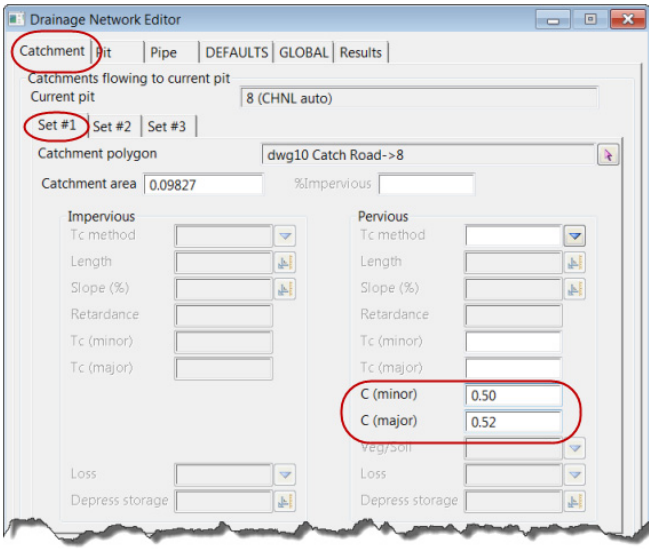
ARR 1987: The composite C value is calculated using the **1hr-10yr intensity,** the percent impervious, ARR frequency factors and the return period specified when hydrology runs are made. **You must enter the 1hr-10yr intensity value. No C values are entered if this method is used!**

QUDM 2007: Similar to the ARR 1987 method (except when the %impervious for the catchment = 0.0). The **Veg/Soil** type must be entered for the catchment via the default catchment tab.

Default Settings

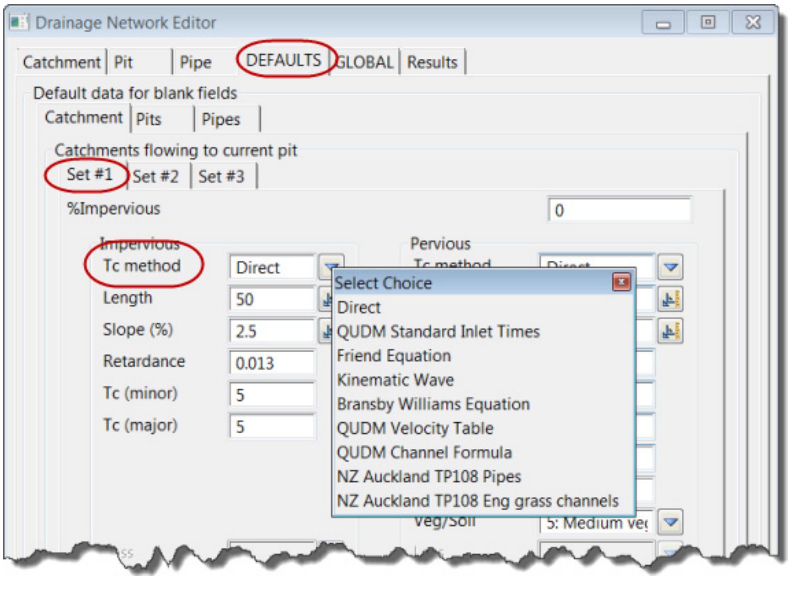


Explicit Settings



20.1.3 Times of Concentration

There are several methods for entering times of concentration for the catchment areas (see list below). Since each catchment may use a different tc method, all of the tc parameter fields on the defaults tab are active and required. They must be filled in even if you do not plan on using that value.



1. **Direct method** requires minor and major tc values.
2. **Friend**, **Kinematic Wave, Bransby Williams** and **QDUM** methods require the retardance, length and slope of the catchments to be entered. Default values must be entered but the optional explicit settings for slope and length can be entered on the catchment tabs or a catchment characteristic strings may be drawn (see Catchment Tc path strings). The length of this string is used for the length parameter and the design tin is used with the string to calculate the slope using the equal area method.
3. Data for the remaining methods is entered in a similar fashion.

20.2 Catchment Areas

Key Points

1. You do not have to draw catchment strings. You can enter the catchment areas manually (ha or acres).

2. When drawing catchment polygons, start near the inlet for auto linking.
3. 3 catchment sets are available and all catchment polygons must be in these models.
4. Specify catchment models via Network Editor->Global->Utility Models **Catchment file**.
5. Network Editor **Set Catchment** button links the polygons and calculates the areas
6. Once a string has been linked to a inlet, it will remain linked until it is deleted or the link has been cleared (**Clear Catchment Links** on the Globals->Utility Models tab).
7. Manual linking available via DNE->Catchments->Set x->Pick button (RB for a manual clear).
8. Manholes (set via cap_config in drainage.4d or **Inlet config** on the **Pit->Main** tab cannot have linked catchments.
9. Use the CAD polygon tool for drawing catchments OR close the string for sag pits
10. You can disable the auto selection of a string via Right mouse on the pick button then select **Clear**.

The catchment strings may be drawn in a CAD package and then imported into 12d or drawn inside 12d. The strings may be easily drawn in 12d with the tin contours and/or flow arrows displayed in the plan view.

When a catchment string is created to define the area for an inlet then all other data entry types will be ignored and the area from the string will be used.

There are 3 sets of catchments and it is up to the user to decide how they are to be used. Each set has its own percent impervious. The most common is to use the sets as land use types (roads, lots and park land for example). Another frequently used option is to use set 1 for all the impervious areas and set 2 the pervious and set 3 for special areas. The 3 catchment sets are drawn in three different models.

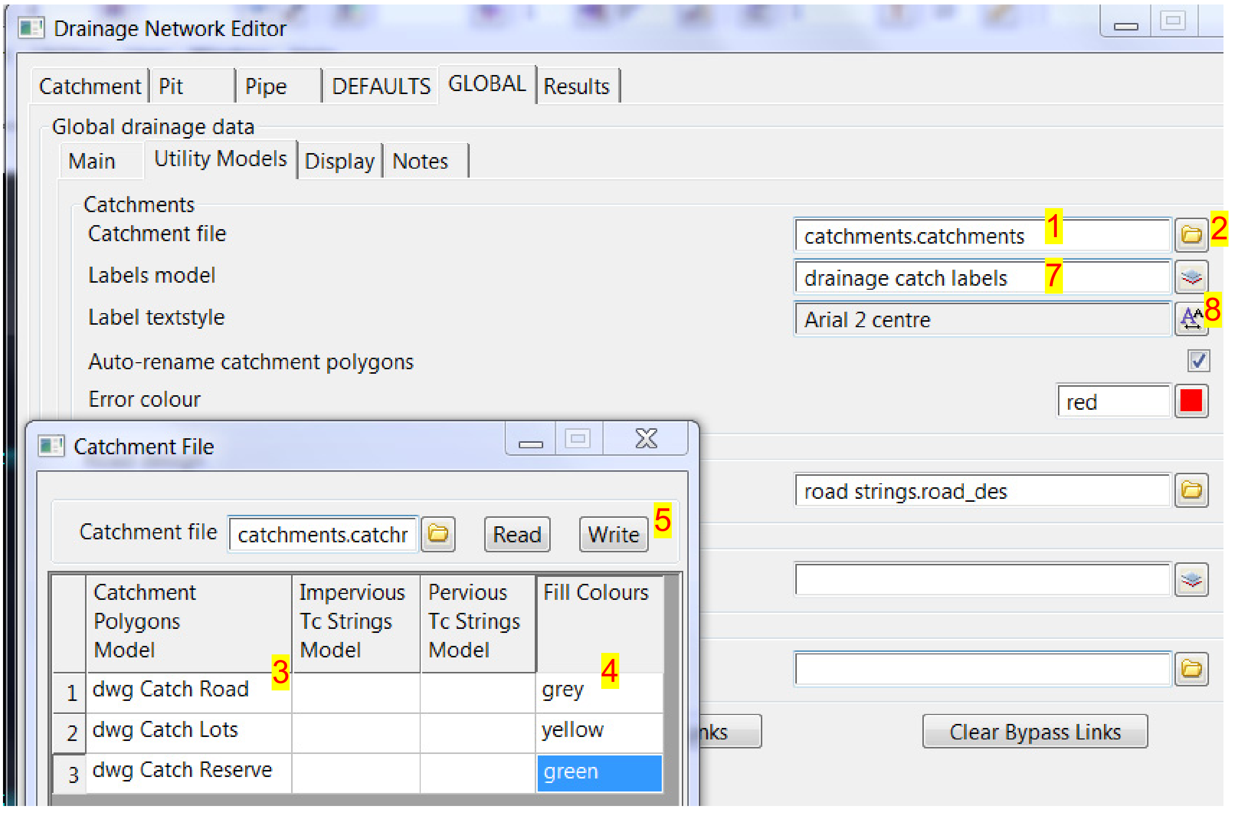
If exporting to external drainage design programs, the package may not accept all three sets so check the interface notes before defining the catchments.

Set Catchments - Auto Linking

In each set/model, 12d will automatically link the catchment string to the inlet that is closest to vertex 1 on your catchment string. This is the preferred method. If this is not possible, then an inlet may be manually linked to a catchment string using the Catchment manual link. More on catchment links may be found in Catchment string links,

Also see Checking the Automatic Catchment Linking

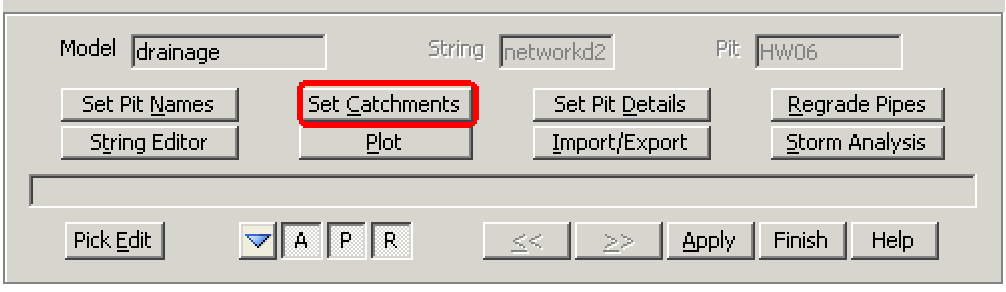
Start the **Drainage network editor** and move to the **Global** Tab and then the **Utility Models** sub tab.



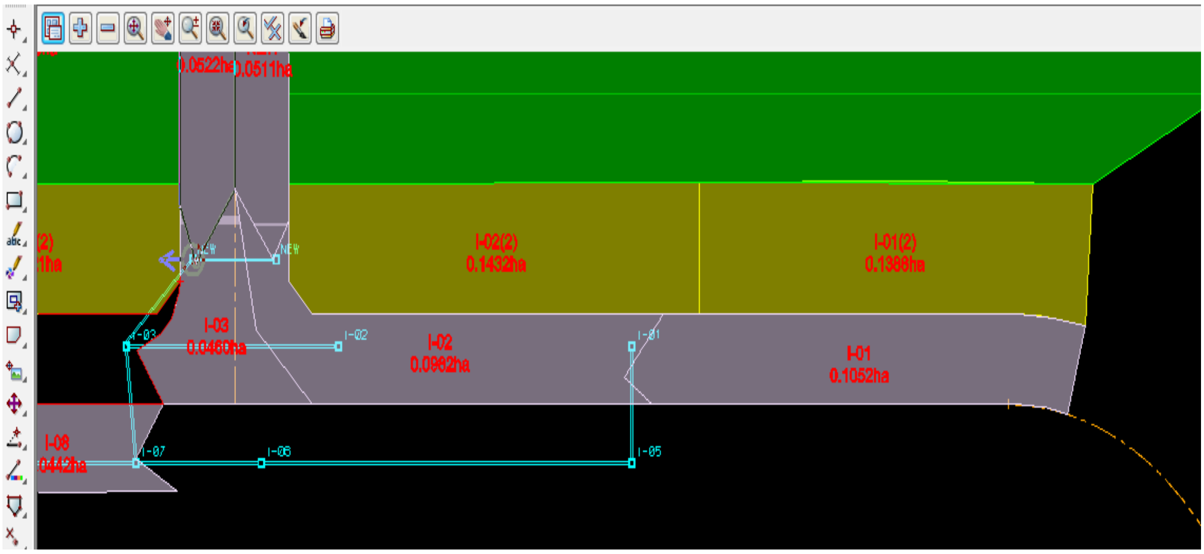
1. Type a filename for the catchment file
2. LB **More info** button and then select **Open**
3. RB to select the catchment model for set 1 (used in the **CAD control bar** above). Every inlet can have up to 3 catchments, row 1 is catchment set 1, row 2 for catchment set 2 and row 3 for catchment set 3. Often the sets are used as catchment types. for example
   Set 1 - Roads, Set 2 Lots and Set 3 Parks
4. .Optional - RB to select a fill colour for the catchments
5. LB **Write** to save the file
6. LB **Finish**
7. Type a new model name for the catchment labels (optional)
8. LB **More info** button to select textstyle (required if **Labels model** is used)

The **Auto-rename catchment polygons** will set the name of the catchment string to the pit name that it is linked to. If it is not linked to any inlet it will be named **"not used".** The model can be checked for **not used** strings by selecting **Models->String Info Table.** The catchment will also be filled with the **Error colour** if a catchment string is not linked.

Label Catchments



Finally select **Set catchments**. This will link the catchments to the inlets and label the catchments. Now add the mod **labels** onto the plan view.



Adding the edges to the fills

To view the edges go to the View Menu button and select **Settings->Faces**

1. LB **Draw Edges**
2. LB **Set**
3. LB **Finish**

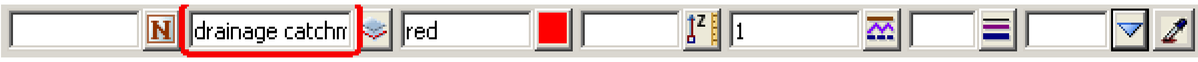
There are some cases where linking the closest inlet to vertex 1 is not feasible. In these cases you may manually link the inlet to a catchment.

20.3 Drawing Catchment Strings in 12d

There are 2 catchments on the south east side of the road that have not be drawn. Use the CAD polygon tool to draw the catchments

Before creating the catchment string set the **CAD control bar** data.

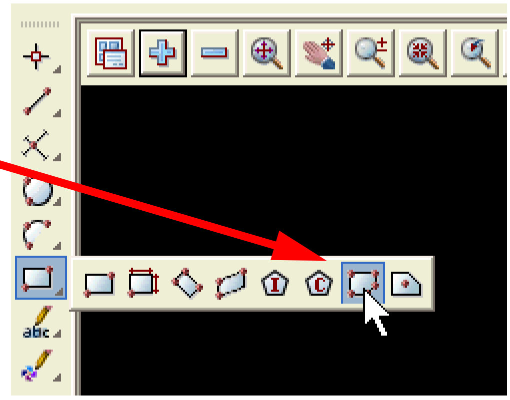
Type the name of a model for the catchment strings



Now to create the catchment string use the **Create line string** button on the CAD toolbar.

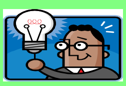
DRAG the **Create line** button and release at the **Create Line String** button.

The first point should always be placed near the inlet. 12d will assume that the catchment will drain to the inlet closest to this first point.



Draw the catchment strings with the accuracy you feel appropriate. Catchment strings for sag pit or drawn along a crest lines so take extra care near the crest low points where the water may overflow and bypass. Continue selecting and accepting the points on the catchment string and the press **ESC** to finish creating this string. You are now ready to create the next catchment string.

Sag inlet catchments: **DO NOT START AT THE INLET, just start nearer to this inlet than any other!** If you start at the inlet then move out to the crest of the catchment, the catchment overflow level cannot be determined from the catchment string.



CAD tips

Use the CAD toolbar eye dropper to load the properties from an existing catchment polygon.

Try CAD trace instead of CAD polygon to trace around edges of existing catchment areas. It is at the end of the CAD string fly out. Follow the directions in the 12d message area (bottom left of window).

Trace begins as normal string draw,
select **t** to start the trace, pick the start point on a string then
pick the end point of the trace, (if the trace is going the wrong way select **f** or **b**).

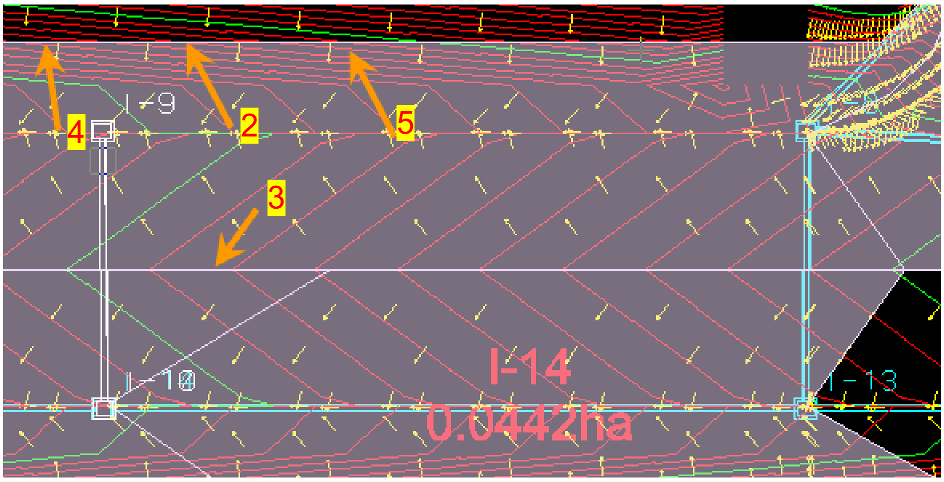
Now you can select another string to trace or
select **p** to return to pick mode.
select **c** to close and finish
and you are off creating another catchment string.

Once the catchments are drawn they become linked to the drainage network in the **Drainage network editor.** We will label the catchment with the inlet name and area at the same time.

20.4 Splitting Catchment Strings to Insert an Inlet

The catchment for the northwest sag pit needs to be split into 2 catchments for the inlet labelled I-9.

We will use mostly CAD tools except for Strings->Strings edit->Split. The trick is to roll the mouse roller ball once whenever you need to redraw the screen.

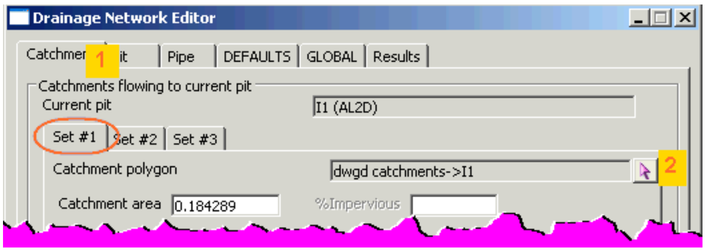


1. Cad String->Close to close the string. (roller ball!)
2. Strings->Strings Edit->Split, select string at split point (string area will go solid white - accept) roller ball!
3. repeat for south side
4. CAD vertex->Append select the string away from the split point, roller ball and draw the new catchment boundary. Single ESC to finish this string.
5. CAD vertex->Append select the string away from the split point, roller ball and trace the new catchment boundary. Double ESC to finish this string and end appending

Catchment manual link

The manual links are used when the first point on the catchment string is closest to the wrong inlet. Note that the following restrictions still apply.

1. The string selected for catchment 1 must be in the model for set 1. To check if you have selected a valid string select the **Set Catchments** button.
2. If the catchment string has already been linked to another inlet (automatic or manual) then the new link will be created and the old link erased.
3. If you change the catchment model for one of the sets on the **Network Editor->Global->Utility Models->catchments** than all of the manual links in that set will be erased.



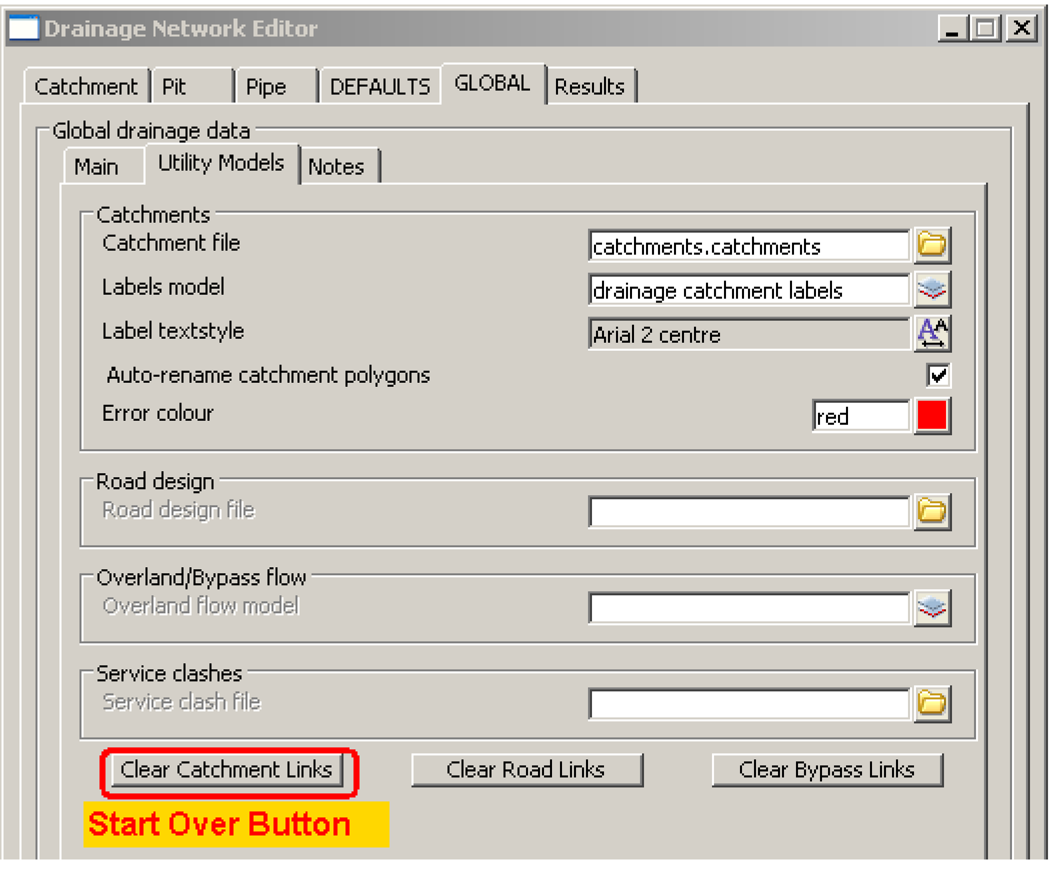
1. Change to the **Catchment** tab.Move to the desired inlet. I-9 in the plan above. The inlet will be circled in the plan view and its name shown in the **Current Pit** field.
2. LB Set 1 then LB the **Catchment polygon** button and pick the desired catchment string. **NOTE!** If you decide to enter a value and NOT use the selected string **RB** on the button and select **Clear.**
   If the **Auto apply** tick box is not selected then you will have to select the **Apply** button for the manual link to become active.
3. LB **Set Catchments** button. A prompt box appears asking you to confirm. Select **Yes**. To toggle off/on the confirm request RB **Set Catchments**. The measured catchment area will be shown in the **Catchment area** field using the units specified in the **Global-Utility models-Units** field.
4. Repeat these steps for the sag pit to the west.

Verifying the Automatic Catchment Linking

The automatic inlet-catchment linking is easily checked by after selecting the **Set Catchments** button on the network editor by any or all of the following:

1. Specifying a **Catchment labels model** with **Labels textstyle** on the **Global->Utility models** tab and . The catchment is indicated when the inlet is selected using the network editor. Since there may be three catchments per inlet the catchment data last viewed in the editor is the catchment that is highlighted.
2. Selecting the **Auto-rename catchment polygons** on the **Global->Utility Models** tab will set the name of the catchment string to the pit name that it is linked to. If it is not linked to any inlet, it will be named **"not used".** The model can be checked for **not used** strings by selecting **Models->String Info Table.**
3. The unlinked catchment strings will be filled with the **Error colour** on the **Global->Utility models** tab.

Reseting All Catchment links to Start Over



There may be occasions where you will want to clear all of the catchment links and begin with a fresh start. Select **Clear Catchment Links** on Network Editor->Global->Utility Models.

20.5 Tc Path Strings

These strings are used to calculate the time of concentration for the impervious and pervious areas. For each catchment set, they are drawn in two models; one for the impervious paths and one for pervious paths. The models are specified using the **Catchment file** field on the network editor (Global->Utility model tab). The 3 rows in the **catchment file** correspond to the 3 catchments available for each inlet. Therefore it is possible to have a maximum of 6 Tc paths models!

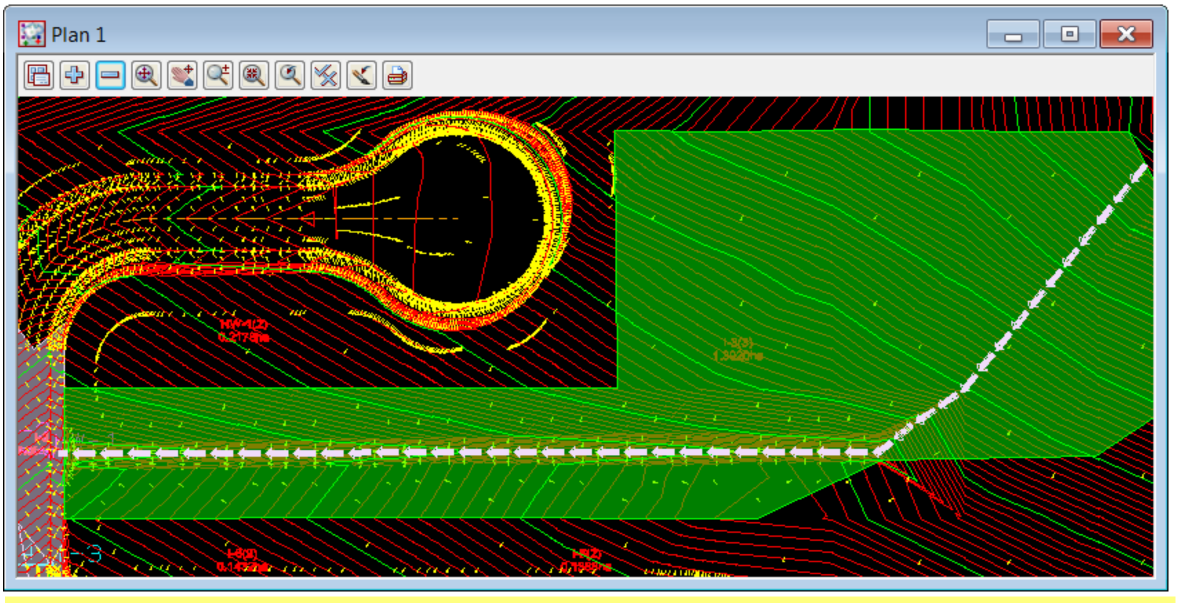
Key Points

1. Each Catchment set may have 2 Tc paths models. Pervious and impervious paths are kept in separate models.
2. End the Tc path string near the inlet that it is to be linked to.
3. Enter the paths models via **Catchment file** field on the (Global->Utility model tab)
4. You must select a Tc method (explicit or implicit) via the **Defaults->Catchments tab** or the **Catchments Tabs**. Just specifying the models is NOT enough!
5. Select **Set Catchments** to perform the calculations of length and slope.

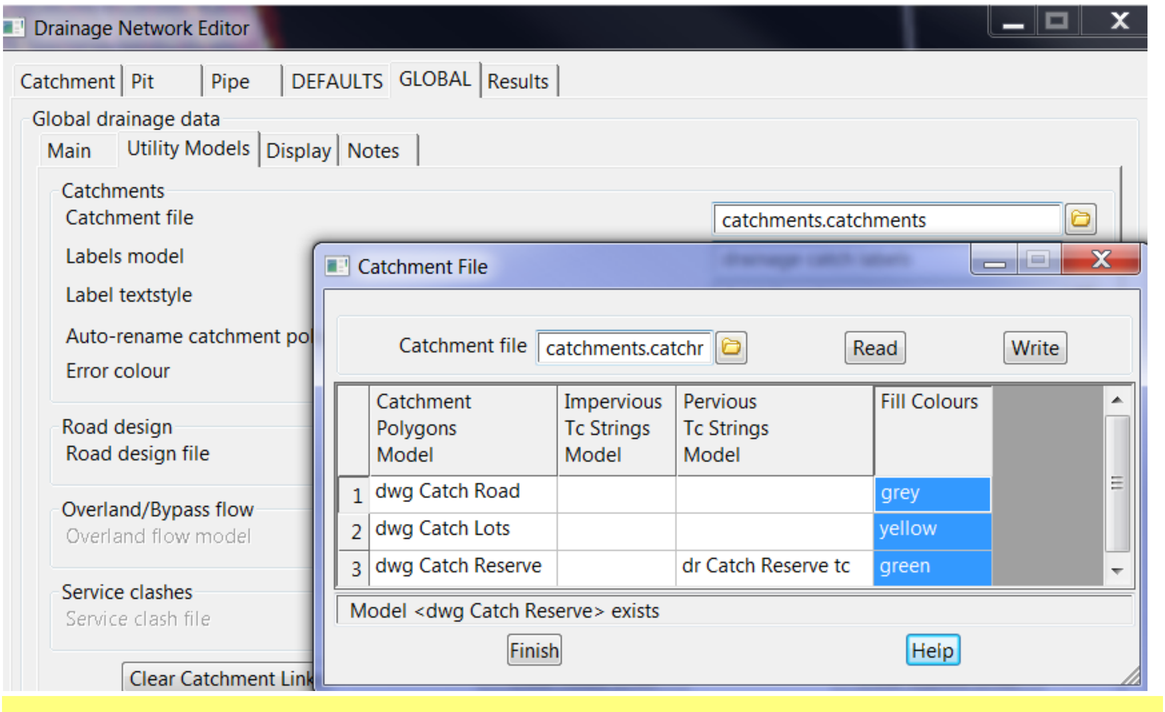
The tc strings can be drawn in the same way as the catchment strings but make sure that you change the model name first! The tc string model is then entered in either the impervious or pervious paths model columns (You could have up to 6 tc string models!).

20.5.1 Catchment slope (equal area)

The length of this string is used for the length parameter and the design tin is used with the string to calculate the slope using the equal area method. These strings are drawn from upstream to downstream, finishing nearest to the inlet they are to be linked to.

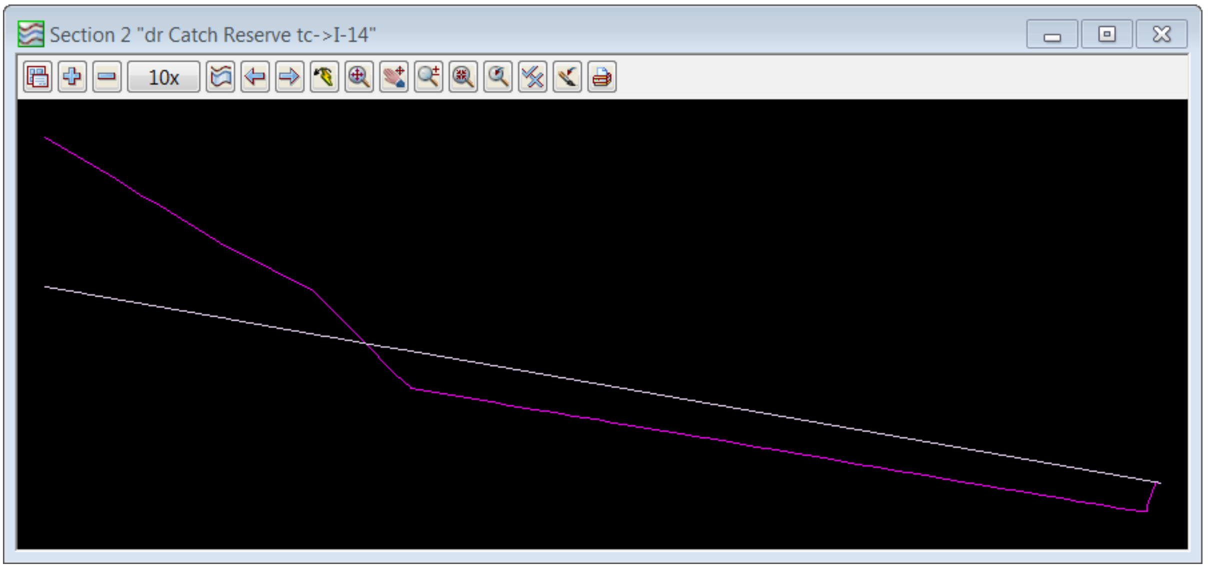


1. Change the Cad toolbar model to dr Catch Reserve tc and the line type to FLOW LINE
2. Use Cad line->Cad line to draw the flow path from the culvert inlet up to the top of the catchment.
3. Use Cad string->Reverse to that the string is in the same direction as the flow.



1. Change the Cad toolbar model to **dr Catch Reserve tc** and the line type to **FLOW LINE**
2. Use Cad line->Cad line to draw the flow path from the culvert inlet up to the top of the catchment.

The equal area slope is calculated when **Set Catchments** is selected. The slope string be profiled, with tin combined added to the view, to see the slope and the equal areas (see below) above and belong the tin.



21.Network Editor - Hydraulics

This section will discuss the hydraulic **Global**, **Defaults** and explicit settings for the hydraulic parameters. The explicit settings for the parameters described on the **defaults** tab will also be found on the **Pit** or **Pipe** tabs.

21.1 Grate Levels

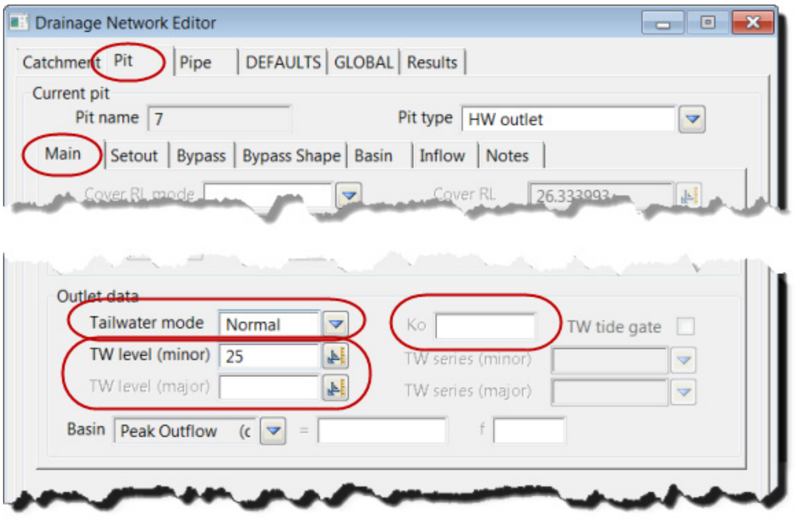
Important! Get the Grate level correct! The freeboard is measured from this level and if you do bypass calcs the water will bypass if the hgl reaches this level. If the grate level is at or below the obvert of the pipe then the pit cannot surcharge and the pipe cannot flow full (it is an open channel). The grate level should not exceed the cover level except for pressurised, bolt down manholes and culvert headwalls.

Special Structures

- **Bolt down manholes** will have the grate level above the cover level. The height above the cover will determine the hydraulic head required to "pop" the cover off and begin surcharging.
- **Headwalls** cover levels are often the top of the headwall structure and and the grate level is set to the highest point of the channel before bypass begins.

21.2 Outlet and Tailwater Conditions

The most downstream pit on each network requires tailwater conditions. Often the invert level on the downstream end of the last pipe also needs to have the invert level locked to either discharge into a waterway or join into an existing drainage system. When the most downstream pit is selected the following fields will become active on the DNE **Pit-Main** tab**.** If these field are not active and you think you are at the outlet see Flow in the Wrong Direction.

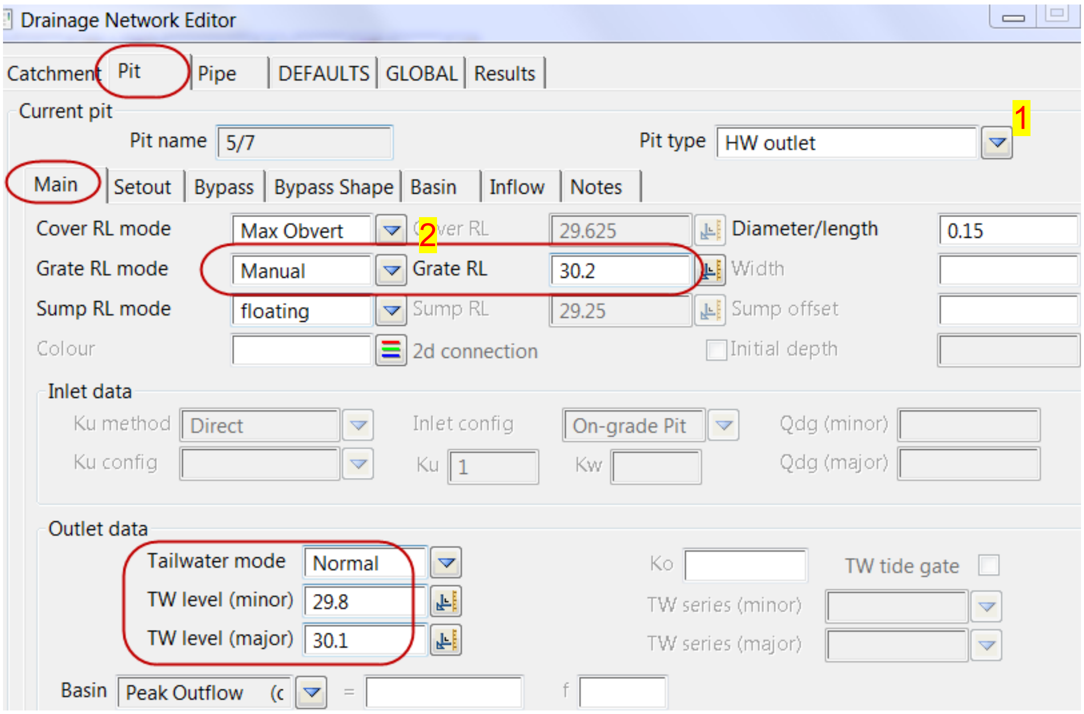


Minimum will use the least of the **Critical** or **Normal** depths. If a fixed level is available for the minor and/or major storms, these value may be entered here.

The **Ko** is the loss coefficient for the exit losses into the downstream system. A value of 1 is typical for discharging into a pond or creek, or a Ku value for the pipe configuration in the pit the network is joining.

21.3 Culvert Hydraulics and Tailwater

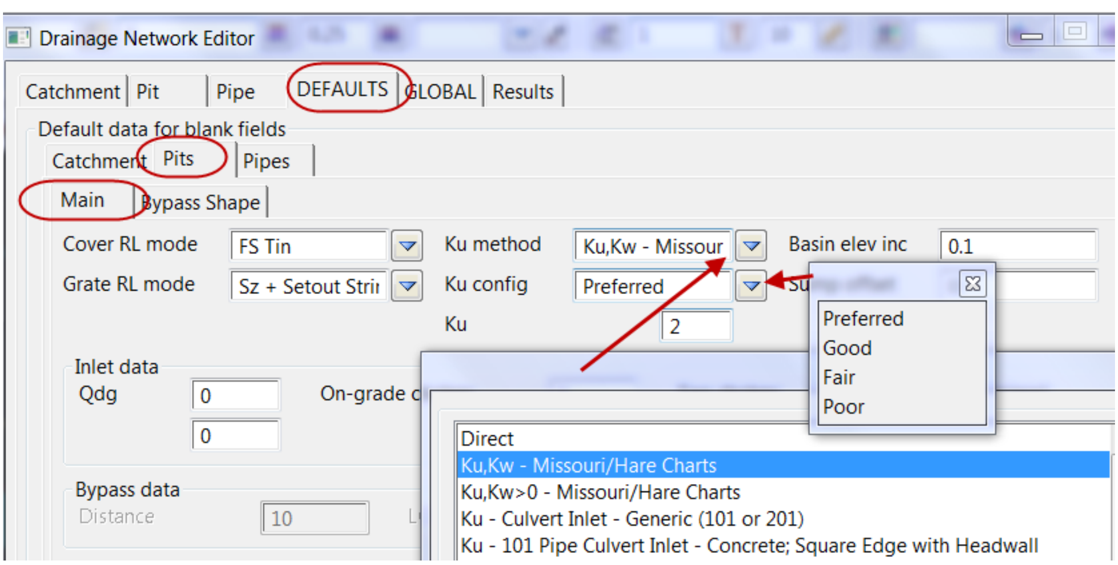
To set the tailwater conditions for the culvert we created at the beginning of the course, select the outlet



1. Select **HW outlet.**
2. Select **Max Obvert** so the top of the headwall is at the obvert
3. Select **Manual** so that the grate level can be set above the max twl+ko losses.
4. Type **30.2** as the road overtopping level
5. Type **0.150** for the HW thickness.
6. Set the tailwater mode to normal and enter the expected twl.
7. Type the minor and major tailwater levels. These

In part 2 of the course we add an open channel to calculate a realistic tailwater for the culvert.

21.4 Pit Losses Ku, and Direct Flow



The **Qdg** (direct flow) (cms/cfs) is water flowing into the manhole. It is added to the approach flow and is subject to pit inlet capacity. This field will be disabled on the pit tab if the inlet type is set to a manhole.

The **Pit loss Ku** is used to model the energy losses through the pits and inlet control on culvert inlets. Three Ku methods are available, **Direct** (user entered), **Ku,Kw via charts** (may be negative), or **Ku,Kw >0 via charts** where all negative values are changed to zero. The remaining methods specify various headwall types for culverts. These will use inlet control curves and backwater energy loss coefficients.

Ku config has 4 options: **Preferred, Good, Fair** and **Poor**. The settings have no effect for 100% grate flow, straight through and 90° bends. For pipes with bends they determine the charts to use. The following are guidelines in selecting the Ku config.

Preferred	water impacts the opposite wall where it exits
Good	water impacts the side wall where it exits
Fair	water impacts the side wall and exits on the end
Poor	water impacts the opposite wall of the pit and exits on the side wall.

21.5 Introduction to 12d Ku/Kw Calculations

The following description is a very general overview of the Ku calculations in 12d. For a detailed description please see the 12d forum site http://forums.12dmodel.com/.

When 12d uses the Ku and Kw Charts, the values of upstream pipe angle, (Qgrate/Qoutlet), (Upstream diameter/outlet diameter) and (pit depth/outlet diameter) are calculated and used in the Ku/Kw charts. Three cases exist which determine which chart is used.

Case 1 - Pits with 100% Grate Flow

The angle between the ground approach flow and the exit pipe is measured. Charts, compiled from Sangster et al (1958) are used; G1 is used for angles less than 15 degrees (rare) and G2 for angles > 15 degrees. 12d's names G1 and G2 can be referenced to other publications in the table below.

Case 2 - Pits with More than 50% Through Flow

12d has 10 charts (T1-T10) compiled from the Hare (1981) and cross referenced to the QUDM and ACTDS Charts. In general T10 charts have greater losses than T1 charts.

Ku Config	0°	22.5°	45°	67.5°	90°
Preferred	T1	T2	T4	T8	T10
Good	T1	T2	T5	T8	T10
Fair	T1	T3	T6	T9	T10
Poor	T1	T3	T7	T9	T10

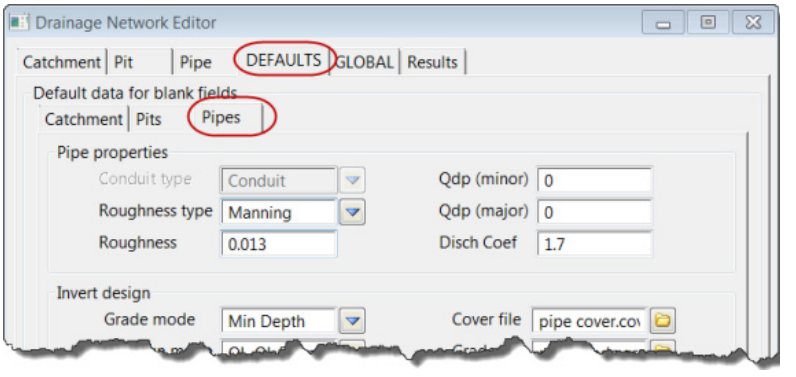
12d Chart Cross References

12d Pit Config	G1	G2	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10
QUDM Ku Chart #	32	32	33	34	35	37	37	38	40	42	44	46
QUDM Kw Chart #	32	32	33	34	36	37	37	39	41	43	45	47
ACTDS Ku Chart #	1	1	2	13	14	10	9	16	18	20	22	7
ACTDS Kw Chart #	1	1	2	13	15	10	9	17	19	21	23	8
ACTDS Pit Type #	1	2	3	11	12	8	7	13	14	15	16	6

Case 3 - Pits with between 0 and 50% Through Flow

With the flow condition between grate flow and through flow, a K value interpolation is performed based on the percentage through flow.

21.6 Pipe Friction Method, Roughness Values and Direct pipe flow

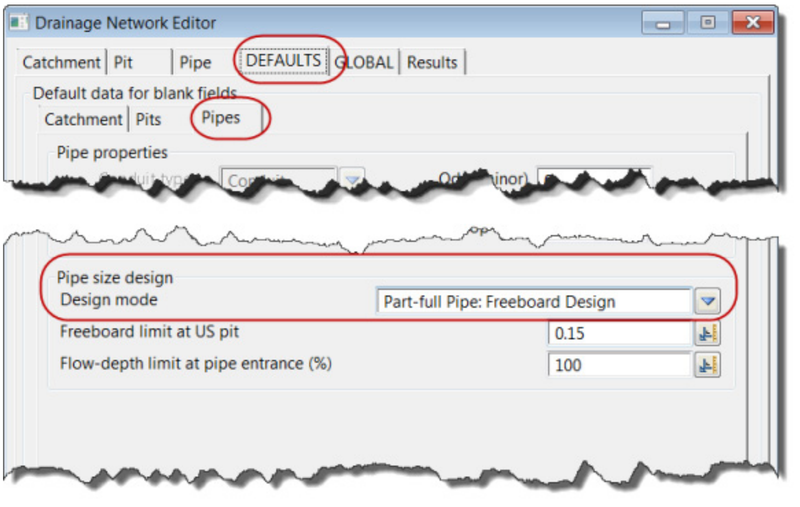


The **default** roughness and pipe roughness method are set here (**Colebrook (mm)** or **Manning**).

Qdp (direct pipe flow is flow) at the upstream end of the pipe that is included in the pipe flow calculations but is not included in the upstream pit losses nor restricted by the pits inlet capacity.

The ranges for pipe peak velocities are used for checking purposes only. If the velocities are outside this range, warning messages will be given in the output window.

21.7 Design mode, Freeboard Limit and Flow-depth limit



The **Design mode** has 4 options.

Pressurised Pipe: Freeboard Design does not use partial depths in the pipes and pipe sizes selected by checking the pit freeboard.

Part-full Pipe: Freeboard Design is similar to option 1 except gradual varied flow and hydraulic jumps are calculated in the pipes. Critical depth is the minimum depth at the upstream end of the pipe.

Part-full Pipe: Flow-depth Design is the similar to option 2 except the pipe sizes are selected by checking the normal depth in the pipe against the **Flow-depth limit**. Freeboard is also checked in this mode and if required the pipe will increase in size.

Open Channel: Freeboard Design is similar to option 2 except depths at the upstream end of the pipe may be less than critical depth for steep pipes (supercritical flow at the entrance).

The **Freeboard limit** is used for all **Design modes**. The freeboard is measured down from the grate level (**Cover RL** plus **Grate offset**).

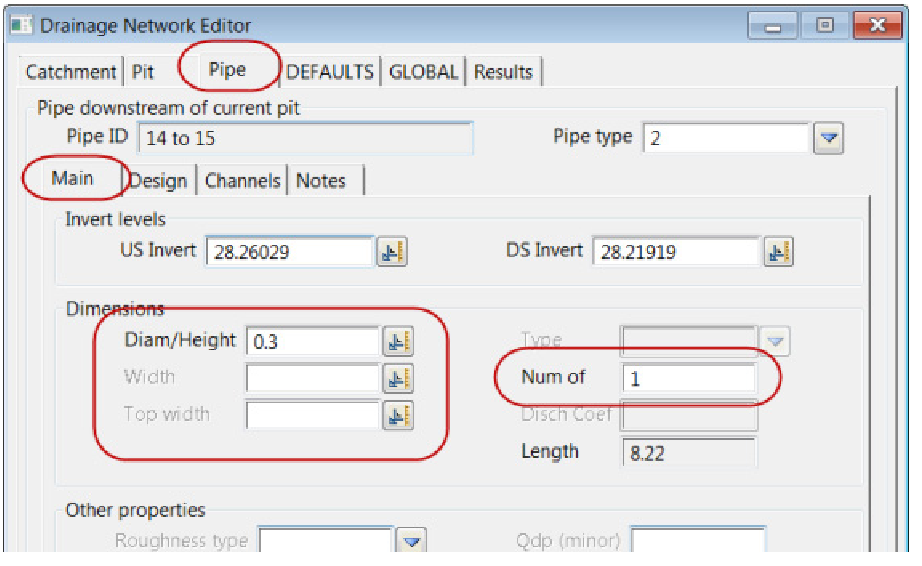
The **Flow-depth limit at pipe entrance (%)** is used in **Design mode** 3. If the flow depth in the pipe is greater than this value the pipe size is increased.

21.8 Pipe Design Parameters - Sizes, Invert alignment, Min Cover, Max Height

The invert levels during design are controlled by the pipe sizes, max pipe height, min pipe cover and invert alignment mode.

21.9 Pipe sizes, Max pipe height and Multiple Pipes and Box Culverts

The 12d design engine will select pipe sizes from the file specified on the **Drainage Network Design** panel, **Preferred pipes file** field list. See selecting pipe sizes. However, the maximum pipe height allowed before multiple pipes are used and the selection of box culverts is set on the **pipe->main** and **pipe >design** tabs respectively.



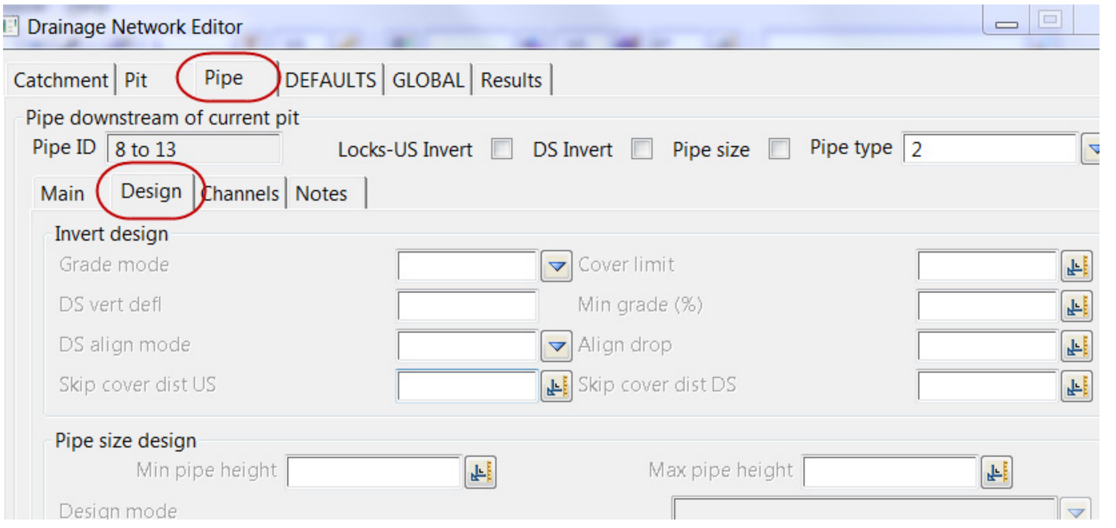
Num of specifies the number of identical pipes. The pipe flow is divided by this value when calculating losses.

To specify a box section in your network, select the pipe and enter a **width** for the pipe.

A **Top width** is used for trapezoidal channels. Note that if the hgl exceeds the top of the channel it will have friction on the soffit just as a box culvert.

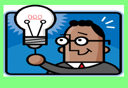
For box culverts, the design engine increases the widths and maintains the height through the available sizes. Once the maximum height has been reached, the next culvert height and minimum width is checked.

21.10 Pipe Size Design



On the **Pipe** tab the **Lock Pipe size** prevents the 12d design engine from resizing the pipe.

Min pipe height can be set for each pipe segment (there is no default for this value).



Specifying a minimum pipe size may speed up 12d design. The starting value for pipe sizing will not be less than this value. That includes all downstream pipes as well. So if you know that the pipe needs to be this size or bigger, enter it here.

Max pipe height can be set for each pipe segment (there is no default for this value). If the 12d design engine requires a larger pipe, then multiple pipes will be selected.

21.11 Calculate Bypass flow routes

This option is required for pit inlet capacity calculations and is covered in the Stormwater Part 2 training.

As an introduction, the bypass strings determine the downstream bypass inlet for each inlet. Inlet capacity is determined from commands in the drainage.4d file. Road grade and crossfall measurements require the inlet to be linked to a setout string and pond depth measurements require a link to catchment string.

22.Drainage Design in 12d Drainage Design

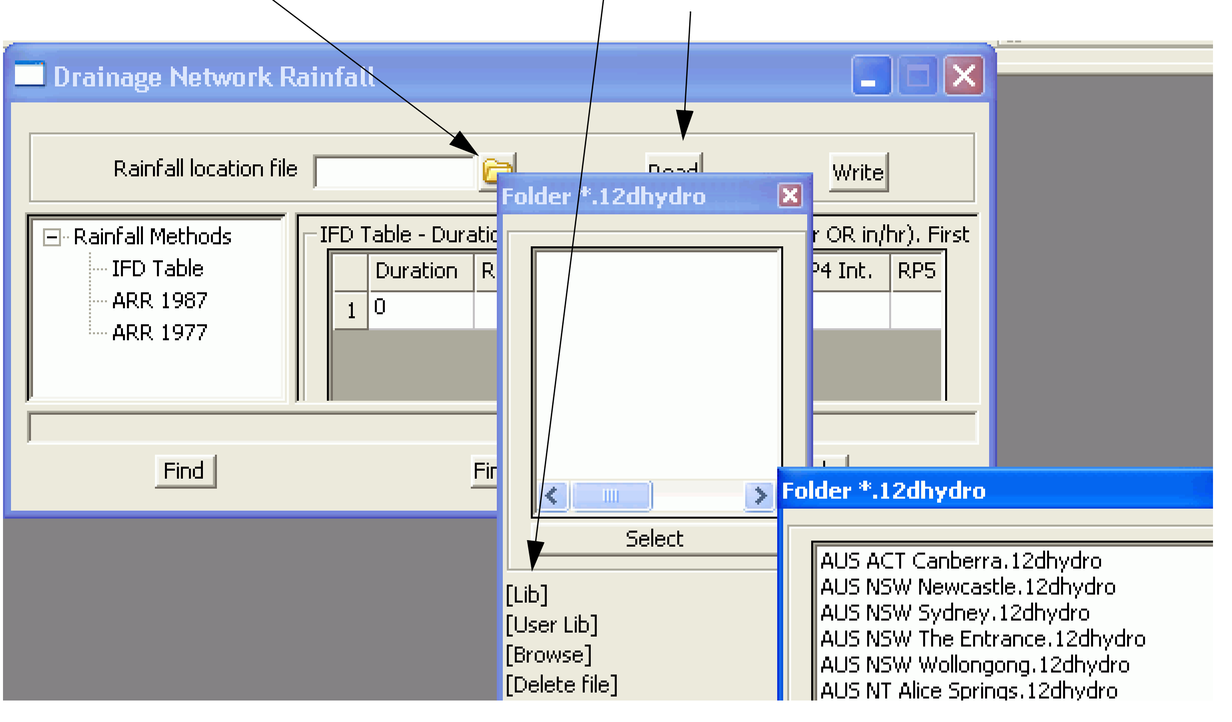
12d has a sophisticated rational method hydrology and hydraulic grade line pipe design engine. In addition it has the capability to export this data to several other popular drainage packages. Regardless of the design method selected, the drainage network in 12d is updated from the design so that drainage plans, long sections and pit schedules can be quickly produced.

22.1 12d Rational Method Hydrology - Drainage Rainfall Editor

The **Drainage Rainfall Editor** is used to input rainfall IFD data using several methods. The data is stored in hydro files (each file is for a specific location) that can be shared between 12d projects. The data is edited using an editor similar to those used for the plot parameter files (ppf). Seven methods for entering/ calculating the rainfall intensities are shown in the panel below. From the main menu select,

Design->Drainage-Sewer->Rainfall EditorData is entered using one (or more if desired) input methods and then saved by entering a **Meteorology file** name and selecting **Write.** The standard 12d system file search paths are used (project folder, user library folder and then library folder).

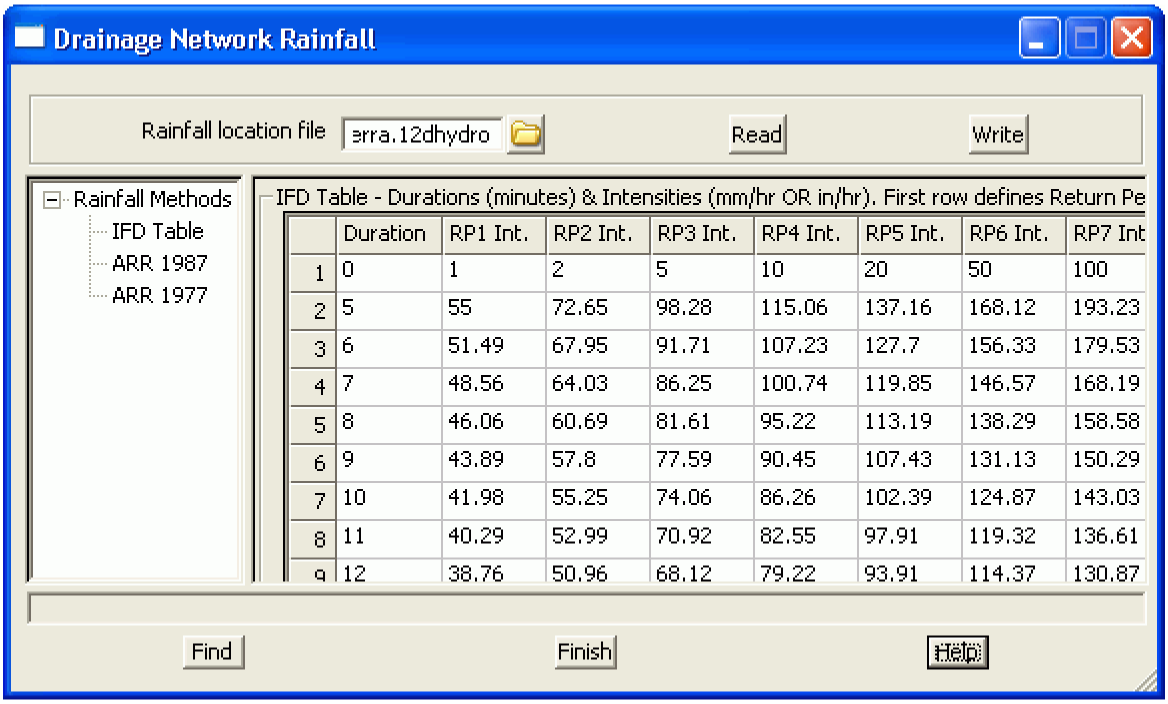
Select the folder icon and then walk right on the **Lib** item to display a list of sample files. Select a file the select **Read. YOU MUST SELECT THE READ BUTTON!**



22.1.1 IFD Tables

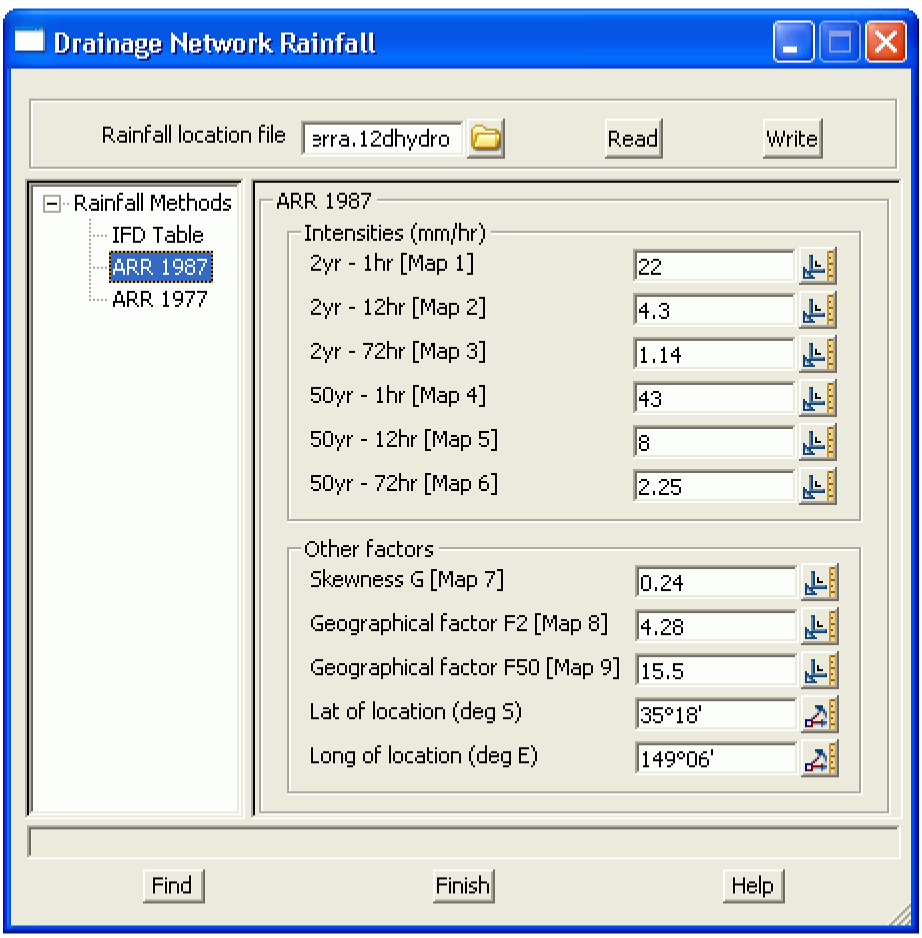
IFD tables are often available from meteorological services. The table input format follows. The first row is used to define up to 9 return periods and the following rows list the rainfall intensities for the duration entered in the first column.

Hint: to increase the size of the grid control select another method, ARR 1987 for example, and then select IFD table again.



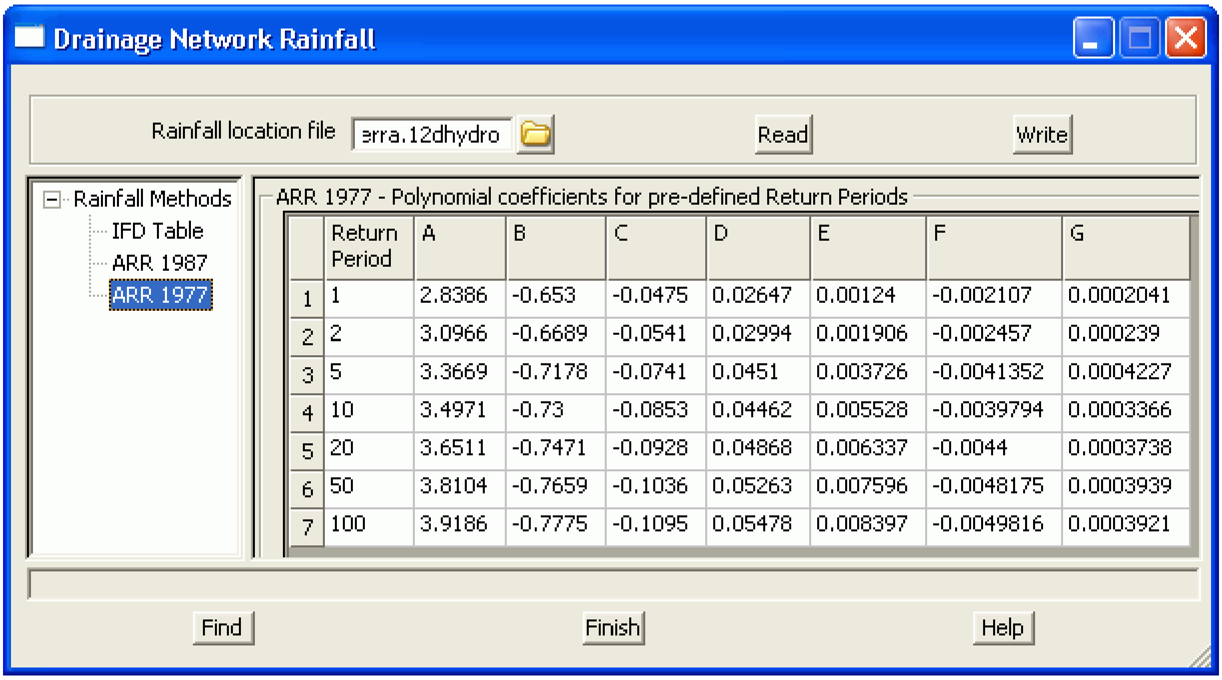
22.1.2 Australian Rainfall and Runoff 1987 Method

The rainfall intensities and other factors from Volume 2 of ARR 1987 are entered in this table.



22.1.3 Australian Rainfall and Runoff 1977 Method

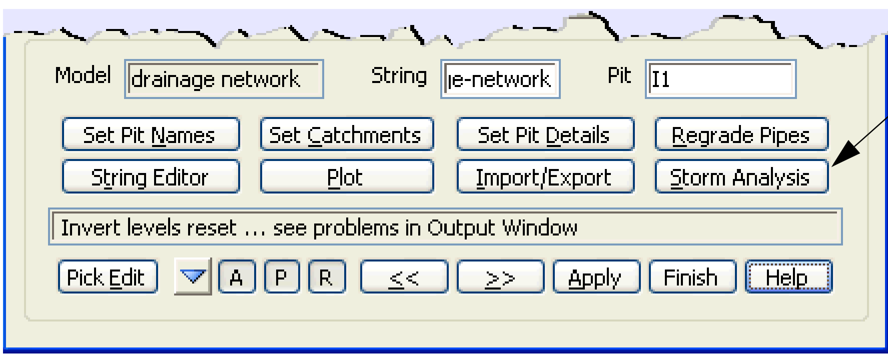
The seven coefficients for each return period from ARR 1977 are entered in this table.



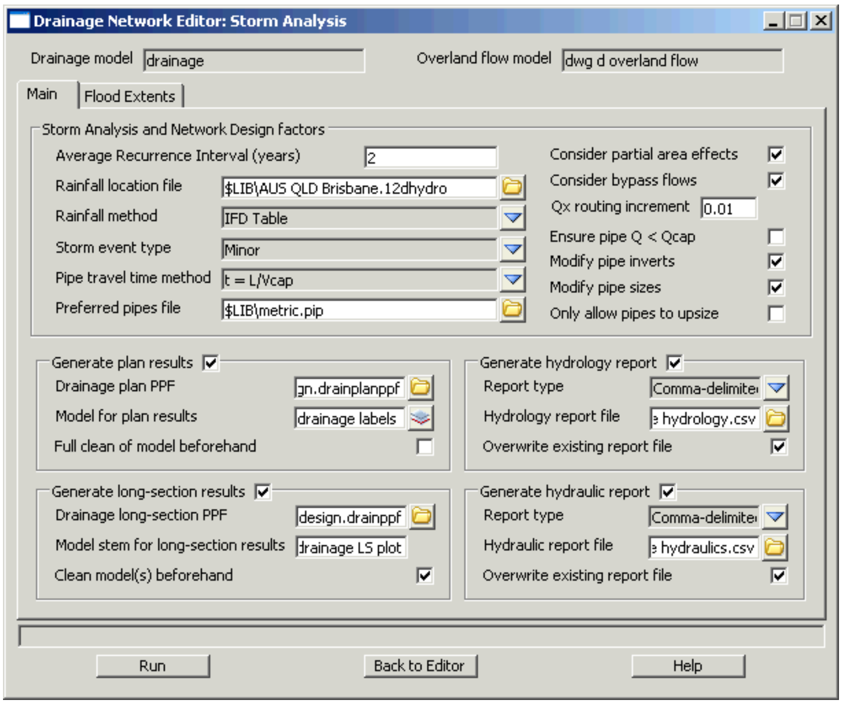
	Return Period	A	B	C	D	E	F	G
1	1	2.8386	-0.653	-0.0475	0.02647	0.00124	-0.002107	0.0002041
2	2	3.0966	-0.6689	-0.0541	0.02994	0.001906	-0.002457	0.000239
3	5	3.3669	-0.7178	-0.0741	0.0451	0.003726	-0.0041352	0.0004227
4	10	3.4971	-0.73	-0.0853	0.04462	0.005528	-0.0039794	0.0003366
5	20	3.6511	-0.7471	-0.0928	0.04868	0.006337	-0.0044	0.0003738
6	50	3.8104	-0.7659	-0.1036	0.05263	0.007596	-0.0048175	0.0003939
7	100	3.9186	-0.7775	-0.1095	0.05478	0.008397	-0.0049816	0.0003921

22.2 Drainage Network Design

The **Storm Analysis** button on the **Network Editor** executes the 12d drainage design, plots the drainage long section and plan annotation and prepares the hydrology and hydraulic design tables.



From the **Drainage Network Editor** select **Storm Analysis.** The following design panel will appear.



Storm Analysis Factors

The valid **ARI** will depend of the method selected but you cannot extrapolate beyond your data. Select the folder icon on the **Rainfall location file** and then walk right on the **Lib** line to select one of the rainfall files in the 12d library. If the file has only one type of rainfall definition then the **Rainfall method** field will be completed. Otherwise select the **Rainfall method** desired.

Storm event type determines which set of design values (**minor or major)** will be used for this run. **Pipe travel time method** should be set to the authorities requirements.

Enable the 12d rational method engine partial area calculations by selecting the **Partial area effects** box.

Network Design Factors

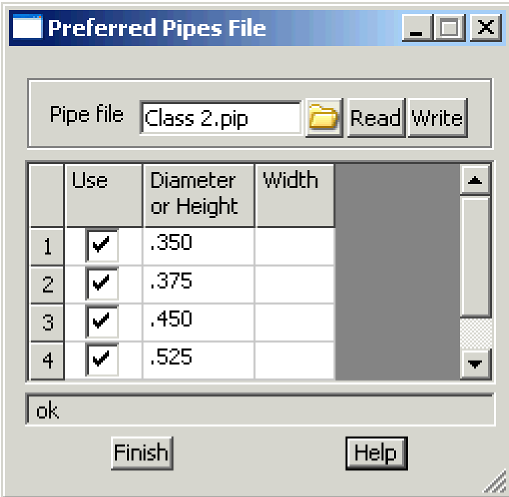
Modify Pipe Sizes

These values control the values to be designed in the run. **Consider bypass flows** causes the engine to use the bypass flow routes and inlet capacity data from the drainage.4d file.

Ensure Pipe Q < Qcap will increase the pipe size if this criteria is violated.

Modify pipe inverts will allow the design engine to shift the inverts if required (usually pipe size changes).

If **Modify pipe sizes** is selected then a files containing the available pipe sizes must be supplied. The pipe sizes in this file are in the **Units** specified in the drainage network editor. To create a new file, enter the file name and then select the folder icon followed by the **Edit** line. The following panel will appear.



The diameters/heights are required and the width is optional to specify a box culvert.

YOU MUST SELECT THE WRITE BUTTON!

The **Upsize only** selection will stop pipes in the system from being reduced in the design. Regardless of this selection, the 12d design engine will not allow a smaller pipe to be selected in the downstream direction.

22.3 Pipe Sizes too Large?

Your hydraulic settings can change you pipe sizes. Check

Grate levels, freeboard, pipe design criteria and read the notes below!

A few comments on why you may have large pipes in your design.

If one pipe is sized large then 12d will not allow a smaller pipe downstream. So when pipes seem large, check the most upstream large pipe. HGL restraints will require you to look downstream of the large pipe.

Check List

1. High roughness values, accidentally setting roughness to Manning with a 0.6 roughness value (Colebrook)!
2. Grate levels not set correctly. Freeboard is measured from these levels.
3. Pipe min pipe cover set very close to freeboard.
4. Selecting **Ensure Q<Qcap** in the storm analysis dialogue. This is required by some authorities but can cause larger pipes in flat areas.
5. NOT selecting **Modify pipe sizes** in the storm analysis dialogue. 12d will not change the pipe size.
6. Selecting **Only allow pipes to upsize** in the storm analysis dialogue. 12d will not check if a smaller pipe will do.

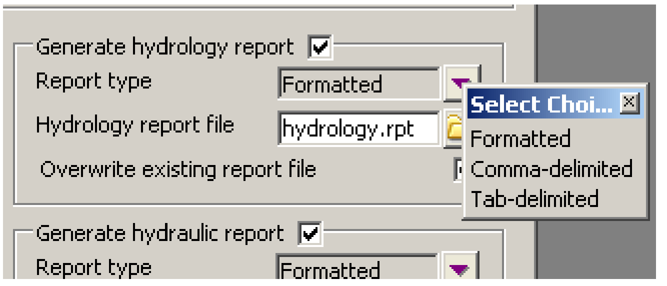
Generate Results in Plan

This selection automatically runs the drainage plot annotation function. A **Drainage plan ppf** must be entered and samples are supplied in the 12d library. A **Model for plan results** is required if this option is selected. The **Full clean model before hand** tick box forces the model to be cleaned before the labels are created. When not selected a “Smart clean” is performed.

Generate Results in Long Section

This selection automatically runs the drainage long section plotter. A **Drainage long section ppf** is required and examples are found in the 12d library. A **Model stem for long section results** is required if this option is selected. In almost all cases the **Clean model before hand** tick box should be selected.

Generate hydrology report

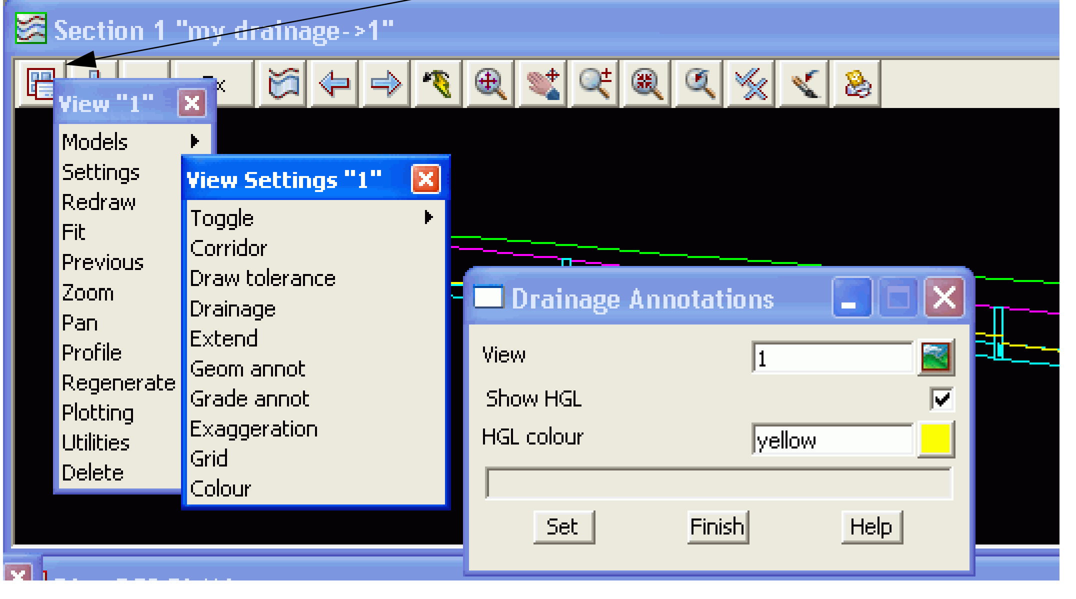


The **hydrology report** may be formatted for inserting into a 12d model/ text editor (formatted) or spreadsheet (comma or tab delimited). In almost all cases **Overwrite existing report file** will be selected.

22.4 The Run Button and HGL data on the Section View

When the **Run** button is selected the discharges are calculated, the HGL check is performed and the pipes sizes and inverts are designed (if selected). The plan and long section drawings will also be updated with the new data (if selected).

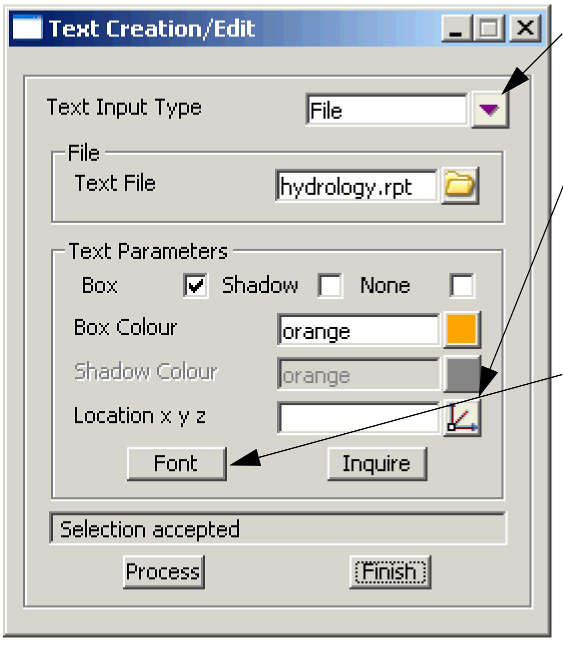
The HGL values will also be available on the 12d section views when profiling the drainage strings. The colour of the HGL line may be changed via the view's menu button then **Settings->Drainage**.



22.5 Importing Text into a 12d model

Formatted text may be inserted into a 12d model by selecting

Drafting->Text and Tables->Create edit paragraph text



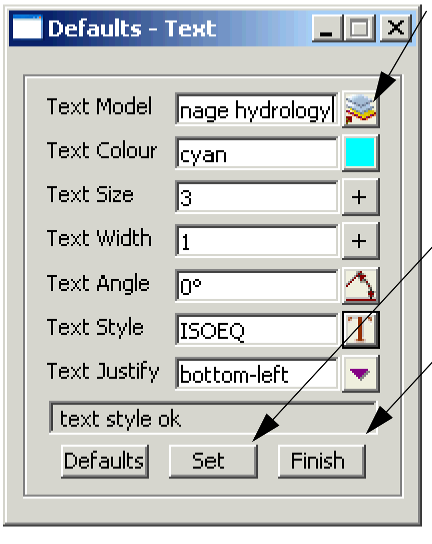
Change to **File.**

Select the folder icon and then pick the formatted text file. It will be displayed then select **Set**.

Next select the location in plan for the text.

The font selected must be a fixed space font or the data will not align properly.

Select the **Font** to display the following panel.



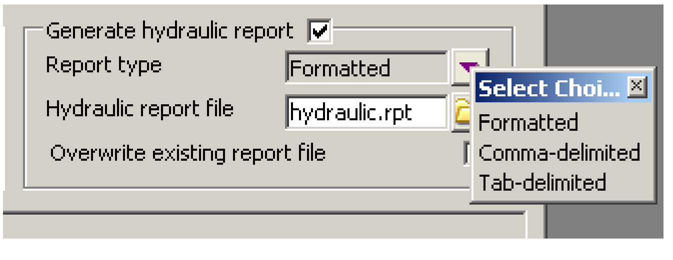
Enter a **Text Model** for the report.

The **Text Style** must be a fixed space font.

Select **Set** then **Finish.**

Now add the **Text Model** onto the

Generate hydraulic report



The **hydraulic report** may be formatted for inserting into a 12d model/text editor (formatted) or spreadsheet (comma or tab delimited). In almost all cases **Overwrite existing report file** will be selected.

If you want both the hydrology and hydraulic report in the same file, enter the same file name in both file fields but turn of the **Overwrite existing report file** for the hydraulic report.

22.5.1 Design Results

Results from the design runs are shown in several forms:

1. Hydrology and hydraulic reports
2. Drainage plan annotations
3. Drainage long sections
4. Hydraulic Grade line on the Section view
5. Output window data - Service/utility clashes

Samples of the hydrology and hydraulics report are shown below.

12D MODEL - HYDROLOGICAL DESIGN SHEET

```
Project:          Storm water Part 1
Drainage Model:   drainage
Location File:    $LIB\AUS QLD Cairns.12dhydro
Tc Method:        Direct
Rainfall Method:  IFD Table
Runoff C Method:  Direct
```

Minor 5 Year Storm Event

Pit Name	Pit Type	Setout Easting	Setout Northing	Setout Level	Catch ID	Time Tc	Intensity I	Runoff C	Area A	Full CA	Full Sum CA	Full Qc=CIA	Partial CA	Partial Sum CA	Partial Qc=CIA	Approach Flow Qa
(-)	(-)	(m)	(m)	(m)	(-)	(min)	(mm/hr)	(-)	(ha)	(ha)	(ha)	(L/s)	(ha)	(ha)	(L/s)	(L/s)
1	A2	5349.23	7322.04	30.09	1P	10.00	163.00	0.75	0.0083	0.0062	0.0732	33.1	0.0031	0.0701	39.7	39.7
					1I	5.00	204.00	0.90	0.0745	0.0670			0.0670			
2	A2	5303.31	7322.04	29.08	1P	10.00	163.00	0.75	0.0059	0.0044	0.0522	23.7	0.0022	0.0500	28.4	28.4
					1I	5.00	204.00	0.90	0.0531	0.0478			0.0478			
3	A2	5248.65	7322.04	27.88	1P	10.00	163.00	0.75	0.0053	0.0040	0.0470	21.3	0.0020	0.0451	25.5	25.5
					1I	5.00	204.00	0.90	0.0478	0.0431			0.0431			
4	A2	5198.75	7322.04	26.78												
5	A2	5314.17	7336.94	29.32	1P	10.00	163.00	0.75	0.0518	0.0389	0.4586	207.6	0.0194	0.4392	248.9	248.9
					1I	5.00	204.00	0.90	0.4664	0.4197			0.4197			
6	A2	5276.19	7336.94	28.49	1P	10.00	163.00	0.75	0.0184	0.0138	0.1631	73.8	0.0069	0.1562	88.5	88.5
					1I	5.00	204.00	0.90	0.1659	0.1493			0.1493			

12D MODEL - HYDRAULIC DESIGN SHEET

Project: Storm water Part 1
Drainage Model: drainage
Location File: $LIB\AUS QLD Cairns.12dhydro
Rainfall Method: IFD Table
Pipe Cover Limit: 1.1 m
Manning n Roughness: 0.01
Freeboard Limit: 0.3 m

Minor 5 Year Storm Event

Pipe ID	Pipe Type	Pipe Length	Pipe Size	Full Pipe Area Af	Pipe Grade	Pipe Grade	Full-area Tct	Full-area I	Full-area Sum A	Full-area Sum CA	Full-area Qc=CIA	Part-area Tct	Part-area I	Part-area Sum A	Part-area Sum CA	Part-area Qc=CIA
(-)	(-)	(m)	(mm)	(sq.m)	(%)	(1 in)	(min)	(mm/hr)	(ha)	(ha)	(L/s)	(min)	(mm/hr)	(ha)	(ha)	(L/s)
1 to 2	2	45.93	300	0.071	2.15	46.6	10.00	163.00	0.0827	0.0732	33.1	5.00	204.00	0.0786	0.0701	39.7
2 to 3	2	54.65	450	0.159	2.16	46.4	10.37	160.79	0.6600	0.5841	260.9	5.11	202.86	0.6230	0.5558	313.2
3 to 4	2	49.91	450	0.159	2.15	46.5	10.63	159.25	0.8974	0.7942	351.3	5.37	200.29	0.8491	0.7574	421.4
5 to 2	2	18.44	450	0.159	1.22	82.0	10.00	163.00	0.5182	0.4586	207.6	5.00	204.00	0.4923	0.4392	248.9
6 to 3	2	31.31	300	0.071	1.88	53.3	10.00	163.00	0.1843	0.1631	73.8	5.00	204.00	0.1751	0.1562	88.5

Pipe ID	Pipe Flow Q	Capacity Flow Qcap	Q/Qcap Ratio	Full Pipe Vel Vf=Q/Af	Norm Depth Vel Vn=Q/An	Crit Depth Vel Vc=Q/Ac	Capacity Vel Vcap=Qcap/Af	US Pit Grate RL	Pipe US IL	Pipe DS IL	DS Pit Grate RL	Cover Min DS	Pipe Bend DS	Pipe Drop
(-)	(L/s)	(L/s)	(-)	(m/s)	(m/s)	(m/s)	(m/s)	(m)	(m)	(m)	(m)	(m)	(deg)	(m)
1 to 2	39.7	184.3	0.22	0.56	2.08	1.09	2.61	30.09	28.68	27.69	29.08	1.10	-0.0	0.174
2 to 3	313.2	544.5	0.58	1.97	3.54	2.15	3.42	29.08	27.52	26.34	27.88	1.10	0.0	0.024
3 to 4	421.4	544.0	0.77	2.65	3.78	2.71	3.42	27.88	26.32	25.24	26.78	1.10	0.0	
5 to 2	248.9	409.5	0.61	1.56	2.70	1.87	2.57	29.32	27.80	27.58	29.08	1.10	53.9	0.059
6 to 3	88.5	172.3	0.51	1.25	2.45	1.51	2.44	28.49	27.10	26.51	27.88	1.10	28.4	0.196

Pipe ID	US Pit Ku	P'head Loss (Ku.V'head)	Pipe T'head Loss	US Pit HGL	Pipe US HGL	Pipe DS HGL	DS Pit HGL	HGL Grade	HGL Grade	F'board US
(-)	(-)	(m)	(m)	(m)	(m)	(m)	(m)	(%)	(1 in)	(m)
1 to 2	2.00	0.03	0.62	29.01	28.98	28.37	28.37	1.34	74.7	1.08
2 to 3	2.00	0.40	0.49	28.37	27.97	27.48	27.48	0.89	112.3	0.72
3 to 4	2.00	0.72	0.86	27.48	26.77	25.54	25.54	2.46	40.7	0.40
5 to 2	2.00	0.25	0.08	28.70	28.45	28.37	28.37	0.45	222.0	0.62
6 to 3	2.00	0.16	0.16	27.80	27.64	27.48	27.48	0.50	201.9	0.69

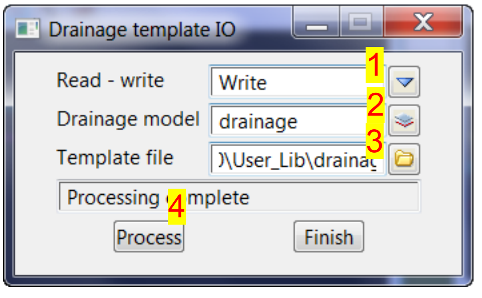
23.Create a Drainage Model Template (Saving Defaults and Globals)

A drainage model template contains your favourite global and default settings for the DNE. These settings are stored as model attributes and the template may be read before or after the drainage strings have been created. Caution: existing default and global setting may be overridden.

You can create you own templates as well. Now that you have completed a drainage job and all of your global and default settings are set, create a template to save in your user library.

From the main menu select

Design =>Drainage-Sewer =>Create->Create/Read template



1. Change to **Write**.
2. Select **drainage** for you existing network
3. LB the folder icon, then user_lib, type a name for the template the select **Open**
4. LB **Process** to create the drainage model template for your next project.

24.Drainage Data Input and Output to Spreadsheets

Spreadsheets are an effective method to manage the numerous variables urban drainage designers create in the modelling process. Spreadsheet data can be transferred to and from 12d in tab delimited files and stored within 12d as “user definable attributes”. These attributes are linked to the pit and pipes within a network. Drainage long section plots can display the pipe attributes in the “arrows” data area and pit attributes in the bubbles area. Drainage plan drawing can also show these pit and pipe attributes.

Drainage strings will be created if they do not exist in the model but pits cannot be added to existing strings.

See also

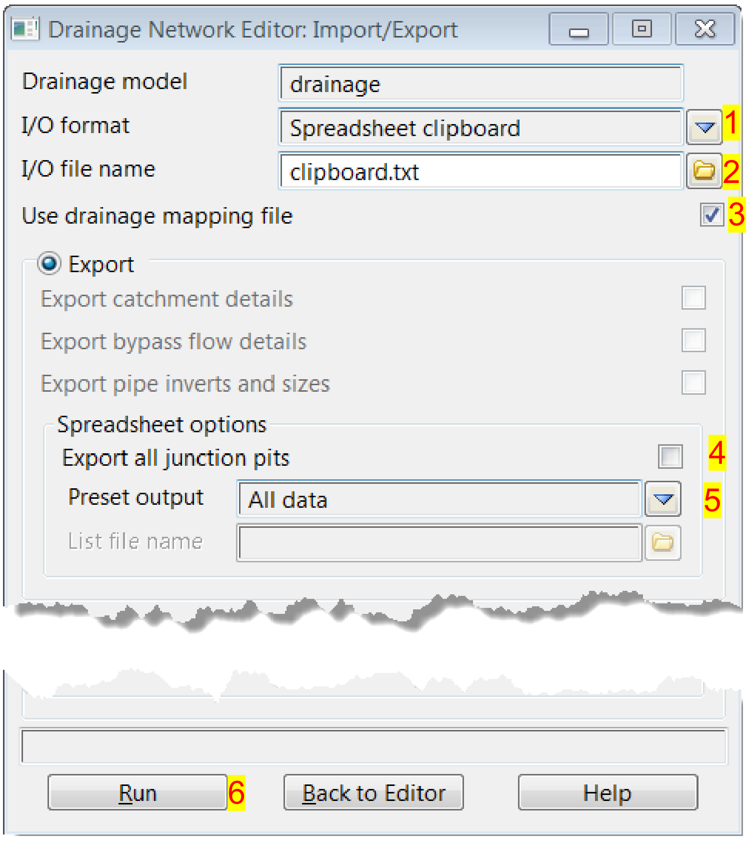
12d to spreadsheet transfers

Spreadsheet to 12d update and create

Spreadsheet options

24.1 12d to spreadsheet transfers

This interface is accessed the **Import/Export** button on the Drainage **Network Editor**.



1. Select Spreadsheet clipboard
2. leave as **clipboard.txt** to send the data to the windows clipboard as well as this file.
3. Mapping files are the most current 12d technology. Leave this selected.

   These options are not used for spreadsheet export.
4. Usually leave this off! Select to export the junction pit at the end of all drainage lines (very rarely needed).
5. You may also select to limit the output if desired. If you like using spreadsheets for data entry, the PCdrain data and ILSAX data formats are useful for adding data for the first time for either program.
6. Select Run to place the data on the clipboard.

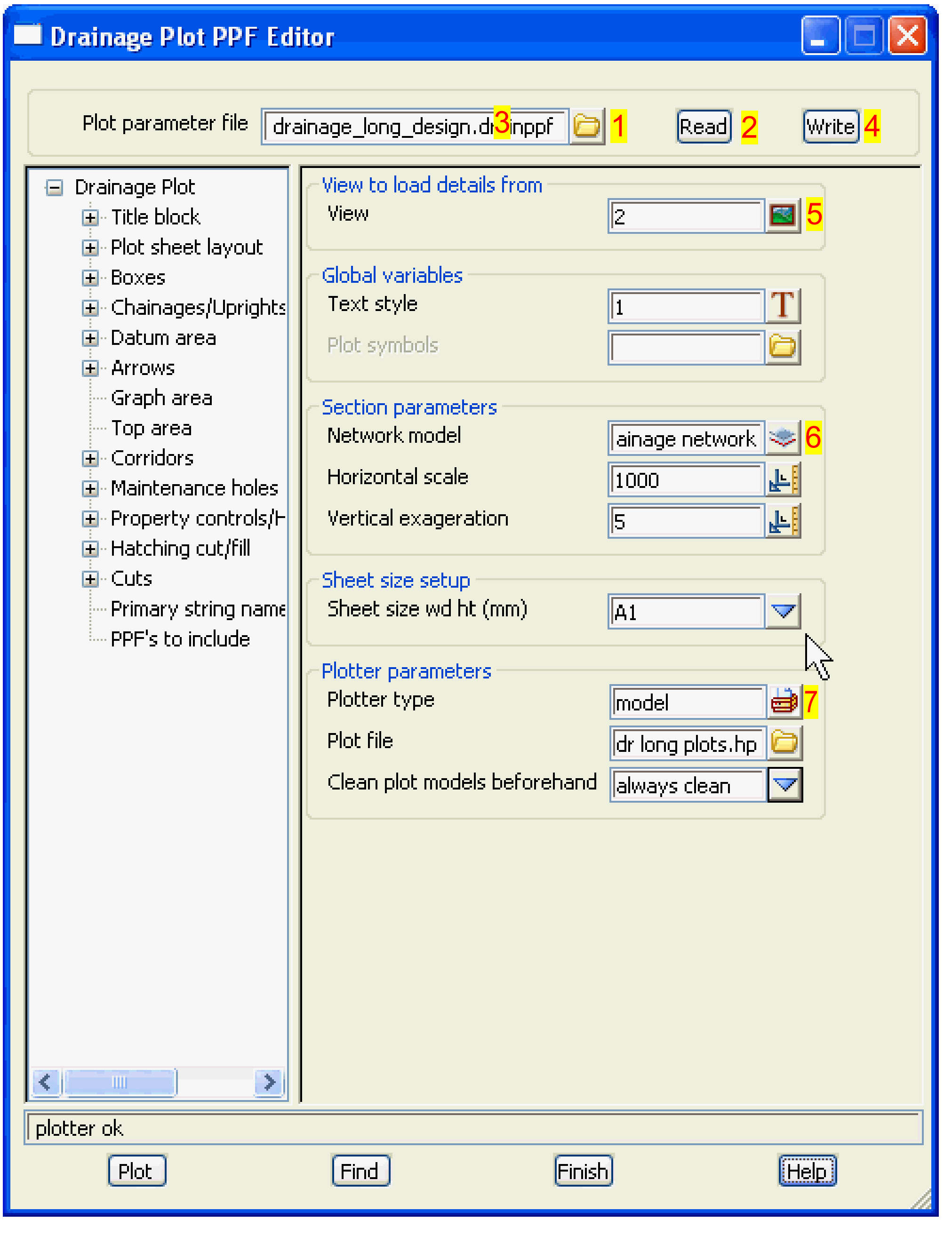
25.Long Section Plotting

Detailed description of the 12d drainage long section plotting may be found in the 12d Reference manual. The long section plots are customised using the drainage plot parameter files (drainppf). Title blocks, user defined text may be added and then plotted directly or to various file formats (dwg, dgn etc.). From the main menu

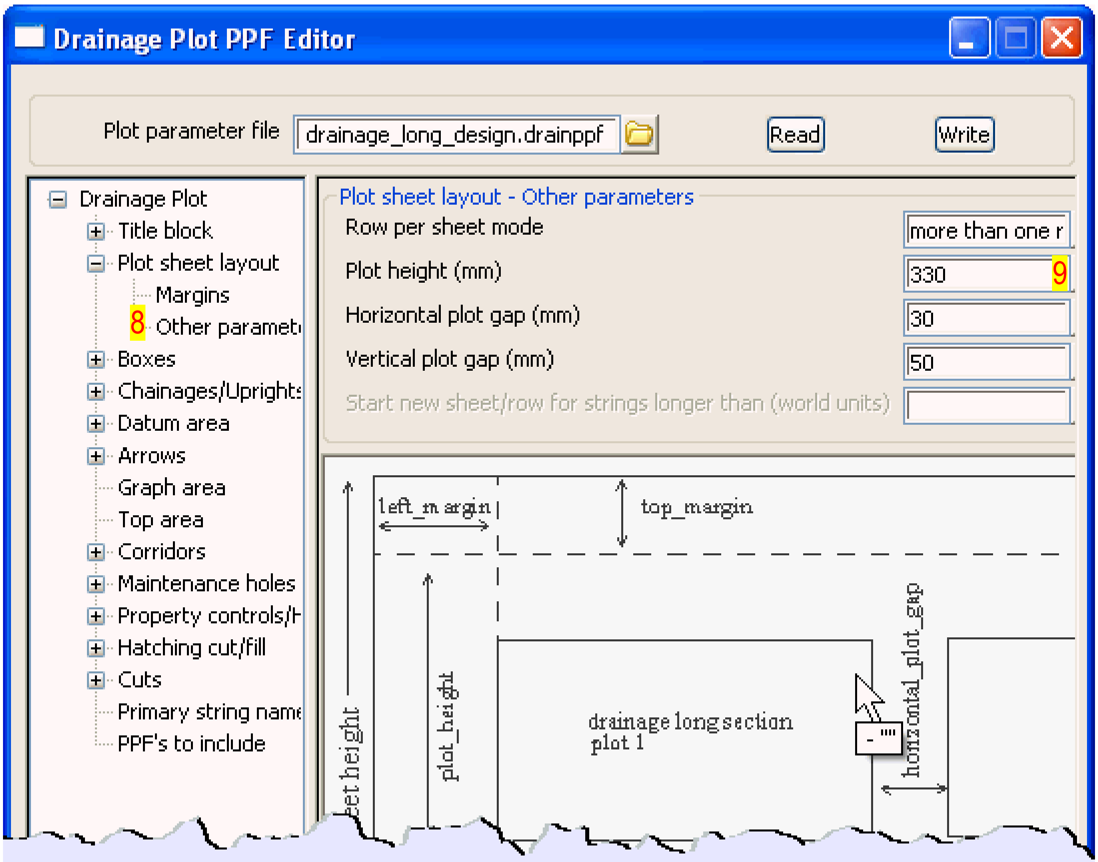
Design=>Drainage-Sewer=>Plots=>Longsections

See Also

Set Pit Details to set road chainage and name data

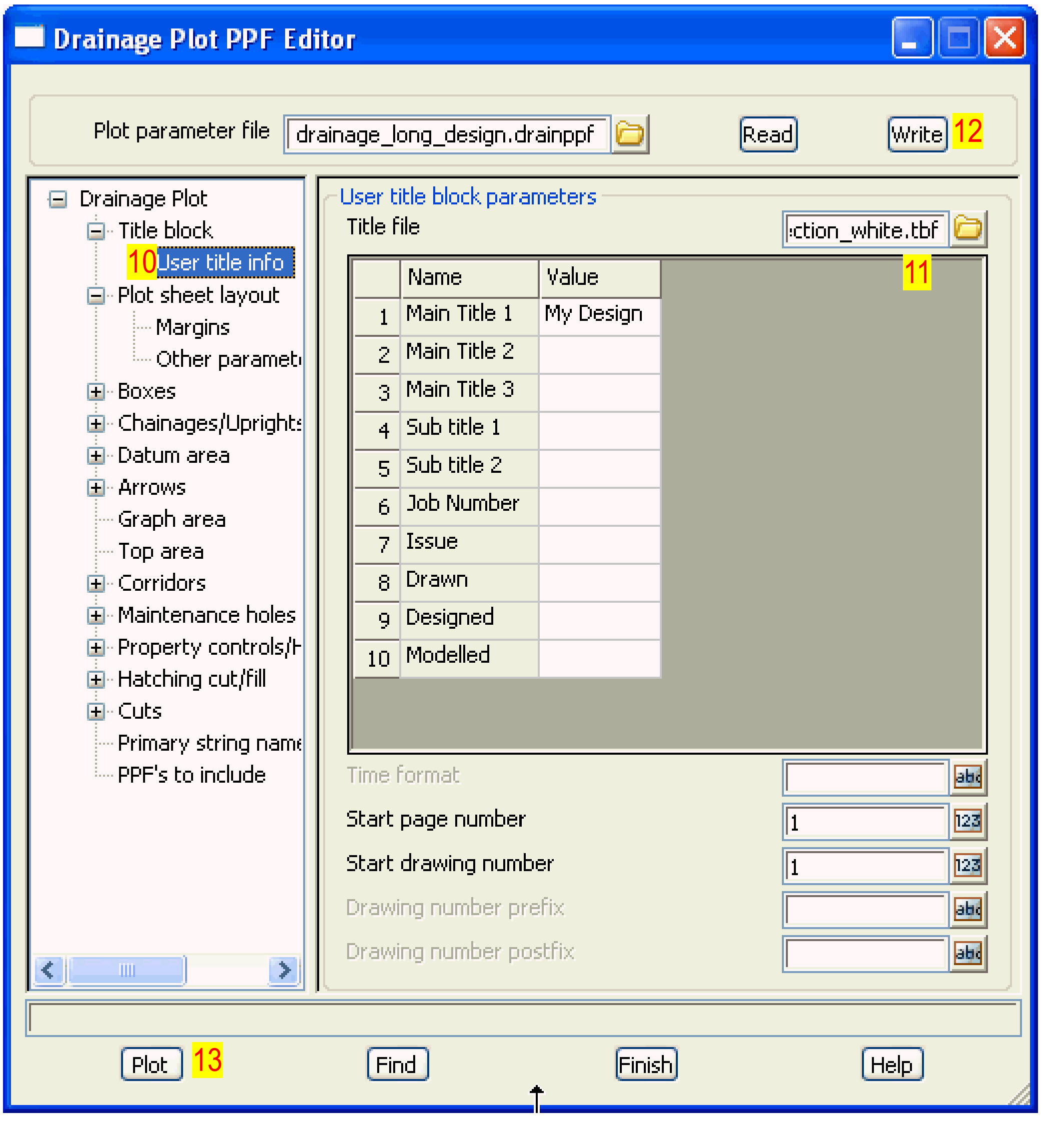


1. To access the drainppf files supplied select the icon and then walk right on **Lib** to select this drainppf file.
2. Select **Read**
3. Enter a new name for this drawing
4. select **Write**. This will save the setting we are about to make should you want to replot this long section.
5. This section view determines the additional models (such as services) to show plot. These are referred to as corridor models. The vertical exaggeration is also obtained from this view.
6. The **network model** field will be completed with the model of the string being profiled. If this is not your drainage network model then select it now.
7. When **Plotter Type** is set to model then **plot file stem** is the model name prefix for plots that will be created. The first sheet of plots will be in model **plot1**, the second in **plot2** etc.



8. Select + on the **Plot sheet layout branch** and then select **Other parameters.**
9. The **plot height** determines how much room is left vertically for the actual plot. This specifies the total height of the plot. 12d then constructs the box area and arrow area on the bottom and then arrow area on the top. The amount left over is used for the long section itself.

   To stop datum breaks from occurring increase this height, increase your plot scale or decrease your vertical exaggeration. If there is too much white space in the graph area then reduce this value.



10. The **+Drainage plot+title block+User title info** allow you to enter the text for the title block.

11. The list displayed is retrieve from the **title file** selected above. Enter the data for the plot

12. Select **Write** to save the changed to the local drainppf file you entered earlier.

13. Select **Plot** and the plots will be send to the **plot file stem** entered. These models may be added (one at a time) to a plan view to inspect them before plotting to paper or exporting to other drawing packages.

26.More Information

This section provides detailed description of items introduced previously in the training notes.

26.1 Linking to Strings in General

Many pit properties may be calculated from strings linked to the pit. The following rules apply to all strings linked via the DNE.

1. The selection of strings is limited to the models specified on the DNE->Utility Model tab.
2. When no link exists for the pit, the closest string matching the selection criteria will be chosen. If a link exists, then no new string will be searched for. Even if the search criteria changes or a closer string is created. The link must be deleted before new links are searched for.
3. The link will not be broken unless
   another string is manually selected
   the link is manually Cleared with the RB on the manual string select
   the string is deleted (manually or when a recalc is done on template applies),
   the model links are reset on the DNE->Utility Model tab,
   the model is removed from the DNE->Utility Model tab.
4. Once link is broken the calculated values will be kept unchanged, with the exception of bypass pits.

26.2 Catchment string links

Each catchment set has its own model of catchment strings and the sets are linked independently.

If the area (in ha) is present in the area field, no new string link be searched for.

Vertex number 1 on the catchment string determines which pit the string will link to. If vertex 1 of several catchment strings are closest to the same pit. Only the closest string will link and the remaining will not be used for any other pit.

After the catchment string link is created, the vertices are re arranged so that the vertex closest to the pit becomes vertex.

26.3 Road string links

Road strings have a name and maximum search distance criteria and there is no limit to how many pits link to a single road string. The closest string matching the string name criteria (in the road string file) and within the search distance will be selected. Note that once the string link is established, changing the criteria will NOT break the link.

26.4 Bypass string links

Pits marked with an Inlet type of Manhole on the DNE->Pit->Main tab ignore the bypass strings. Bypass strings must pass within 1 manhole diameter to be linked..

27.Training Check List

Network plan	- convert from strings
	- insert, append, delete and move pits
Network Vertical	- Set Pit Details and Regrade pipes
	- Pipe alignment modes
Set Pit Names	
Catchments	- draw, label, link and check
	- tc strings
Hydrology	- Area, Tc, C values with defaults
Hydraulics	- setout to grate
	- method (pipe full or HGL)
	- available pipe sizes and box culverts
Services	- over and under drainage pipes
Road strings	- rotate symbols
	- x,y and/or z for setout
	- road grade and crossfall for inlet capacity
	- road chainage from centre line
12d design engine	understanding the reports
Setout pit schedules	- easting, northing and road centre line
Drawings	Long Section plotting
	Plan drawings
Kerb Strings	- select by name
	- crests and sags

THE END

1.0

1.0 Stormwater Design Part 2- Introduction

The **Stormwater Design Course Part 1** and this manual, the **Stormwater Design Part ,** describe the functions and processes of the 12d drainage module. In these documents, the generic term **pit** refers manholes inlets, catch basin and manholes.

The **Stormwater Design Course - Part 1 Notes** contain**:**

- create a super tin for pipe cover and pit cover levels,
- set Defaults and layout a drainage network from CAD and in 12d,
- use the 12d Drainage Network Editor to assign names to the pit/pipes, avoid service clashes, grade pipes, align obverts, minimise depth and many other design tools,
- designate catchment areas and produce catchment plans,
- run the 12d storm rational hydrology and hydraulics engine,
- transfer data to and from electronic spreadsheets to enable the user to easily review the data and add user defined data to the 12d pipe network. This data may include such data as pipe bedding types and trench width,
- create a drainage template containing customised default design parameters,
- create pit setout schedules to export to spreadsheets or word processors for final formatting,
- produce long section drainage profiles including HGL data, flows, invert levels and service crossings,
- create plan drawings with pipe sizes, flows, pit symbols, linestyles for pipe sizes, design parameters for pit and pipes and user defined data,
- locate pits/manholes at exact chainage and offset locations.

This manual, the **Stormwater Design Course - Part 2,** is intended to describe the additional features of 12d Model drainage and discuss the customisation of the package. This will include

- customising the drainage.4d file Drainage Definitions - Manholes and Pipes,
- 12d storm analysis with inlet capacity calculations and bypass flow,
- flooded width analysis and flooding at SAG pits,
- drainage trench excavation volume calculations,
- pipe and pit quantity calculations/reports,
- open channel calculations,
- adjusting pit locations for changes in horiz road geometry
- analysing the major flood events,
- creating drainage symbols with grates and upstream side inlets,
- detailed drainage plan labelling and long sections with hatching under roads.
-

2.0 Starting with a Basic Drainage Network

In this document, the generic term **pit** refers to manholes, inlets, catch basin and manholes. When the term **manhole** is used on the 12d menu system it refers to any type of pit. Pit types, dimensions and inlet capacities of the pits are set in the Drainage.4d setup file/database.

These course notes assume that you have completed the Stormwater Design Course and that you have experience creating 12d Model drainage networks with catchments areas. You may continue this project or begin with a completed drainage design found in the folder

\12d\10.00\Courses\Drainage_Analysis

The project name is Local Road Complete.

3.0 Setup Files and Their Locations

The drainage.4d setup/database file contains the pit and pipe types (RCP, Class 2 etc.). The drainage.4d setup/database file can also control many of the settings in the drainage network editor (DNE). Changing the pit/pipe type in the DNE will result in your favourite settings being applied thus minimising user input errors.

The pit types may optionally include:

- DNE field controls, including internal pit dimensions (diameter or length and width)
- pit wall thicknesses that vary with depth,
- pit connection points (locations where the pipes joint the pit),
- inlet capacities for sag and on grade inlet pits.
- detailed pit type descriptions to be inserted into your pit schedules
- user defined attributes.

The pipe types may optionally include:

- DNE field controls, including internal nominal and actual pipe sizes with wall thickness,
- user defined attributes.

For PCdrain and Drains users there are routines to read your gully pit/database files and create the Drainage.4d setup file/database.

The Drainage.4d setup file/database may be customised for any additional inlet capacity data you may have.**REVIEW THIS DATA CAREFULLY!**

4.0 Bypass Flow

Bypass flow strings are used to trigger the bypass calculations in the network editor and are used as a centre line for flooded width calculations.

The 12d storm analysis, and many of the design programs 12d exports to, allow for bypass flow. Bypass flow involves the calculation of pit inlet capacity for on-grade or sag inlets. These capacities are based on the pit type and may use either ponding depths (sag inlets) or on the road grade and/or crossfall upstream of the inlet (on grade inlets).

4.1 Key Points

1. Draw an bypass flow string in the direction of flow so that it passes within 1 pit diameter of an inlet. At sag locations the string should show the direction of flow during bypass conditions. When bypass flow strings join they must join within 1 pit diameter of an inlet (pits with **inlet config** set to **Manhole** are not considered inlets). If flooded with calculations are to calculated the string should located in the flow channel.
   Enter the model name in the **Bypass flow model** field on the **Global->Utility Models** tab.

   Note: if there is no bypass flow string within 1 pit diameter then 100% of the approach flow will enter into the pit.
2. Many bypass strings may join at an inlet but only one bypass string should leave each inlet.
3. Set the pit type. (**Pit Type** on the **Pit** tab). With a bypass string within 1 pit diameter of the pit centre, no water will enter the pit unless the pit type has inlet capacity data defined in the Drainage.4d setup file/database.
4. Set the **Inlet config** on the **Pit->Main** tab (Manhole, On-grade or Sag pit). This selection will be disabled if cap_config parameter sets the inlet type in the Drainage.4d setup file/database. Manholes have no inlet capacity and are not considered inlets, on-grade inlets capture the water as it passes the inlet while SAG inlets trap the water flowing in from all directions (until the pond depth overflows at the low point of the catchment string).
5. On grade pits may require road grade and/or crossfall data for inlet capacity. They may be entered manually or calculated using the road strings. A setout string link is required to measure road grade. If road crossfall measurement is needed then the centre string is also required. These strings are specified using the Road design file on the **Global->Utility Models** tab (see Stormwater Part 1 manual).
6. Sag inlets require a pond depth either manually entered or calculated by 12d. Pond depth calculations require a link to a catchment string to locate the overflow point and correct grate level (Grate rl mode).
7. Press the **Set Pit Details** button. Road grades, crossfalls, pond depths and bypass pits will now be found on the **Pit->Bypass** tab of the **Network Editor**. Measurement markers are created in the **construction drainage data** model.
8. Bypass pits may be cleared using **Clear Bypass Links** on the **Globals->Utility Models** tab
9. **Storm Analysis** must have **Consider Bypass Flows** selected on the **Main** tab.
10. **Calculate overland flood extents** is optional on the **Flood Extents** tab of **Storm Analysis**.

4.2 Creating Overland Flow Strings using Downhill Strings

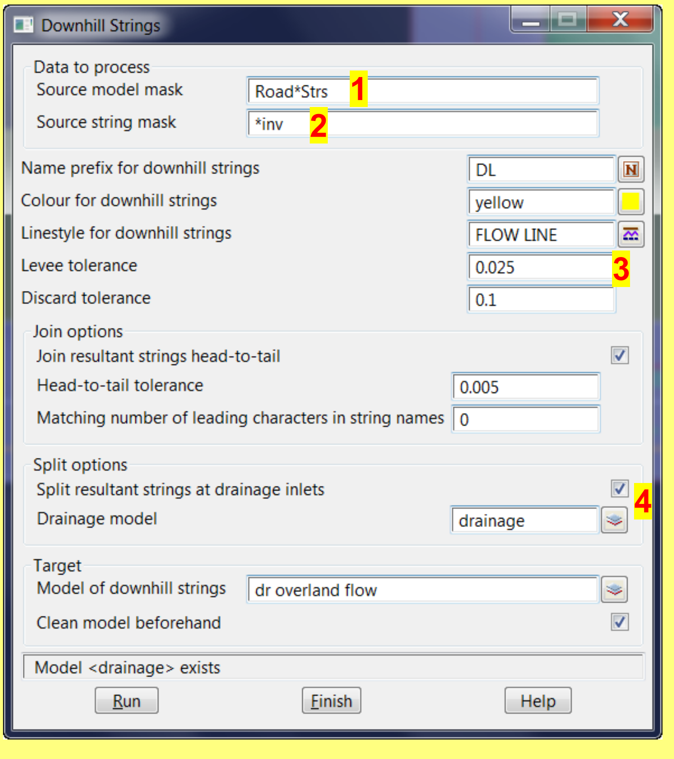
The bypass flow string must be within 1. pit diameter of the drainage pit in be considered on the bypass flow path. If the bypass flow string is to be used for flooded width calculations in the future, the string must also be drawn in the main flow area of the cross section. At sag locations the bypass flow strings indicate the direction the water flows when it overtops the overflow point. This string usually goes uphill to this overflow point.

The downhill string function quickly creates bypass flow strings from road design strings. The remaining tasks will be the bypass flow at intersections and areas off the roadways.

This routine copies strings to a single model using a string name mask to select the strings from a collection of models (using a model name mask). These strings will have their name, line style and colour changed. The line style is usually FLOW LINE so that you may see the direction of the string (downhill). The strings are split at crests (identified using a levee tolerance). They may also optionally be split at drainage string inlet location to allow easy modification using the drainage utility string editor.

From the main menu select

Design->Drainage-Sewer->Downhill strings



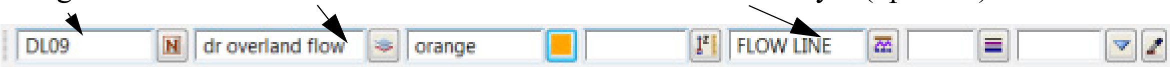
1. type **Road*Strs** (it is case sensitive). These are the models that contain all of the road strings. **Road 1 Strs** for example.
2. type ***inv**. The road string models have a **linv** and **rinv** string for the invert of the kerb and gutters.
3. If the crest height on one side is less than this amount it will not be split.
4. Tick the box and select the **drainage** model. This will help with the drainage utility editor later.
5. type **dr overland flow** to create a model for the new strings
6. Select **Run**
7. add the model **dr overland flow** onto the **roads** view.

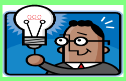
4.3 Creating Overland Flow Strings at Intersection and Sag Locations

Use the **CAD toolbar** to create the overland/bypass flow paths the flow crosses the road.

To use the **CAD toolbar** go to the **CAD data bar** and enter a string name and model name.

Select the line style (optional).

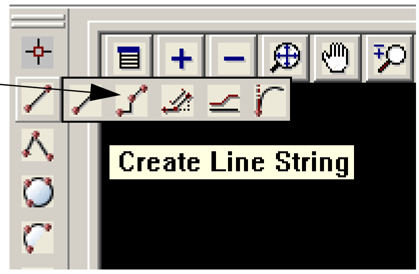


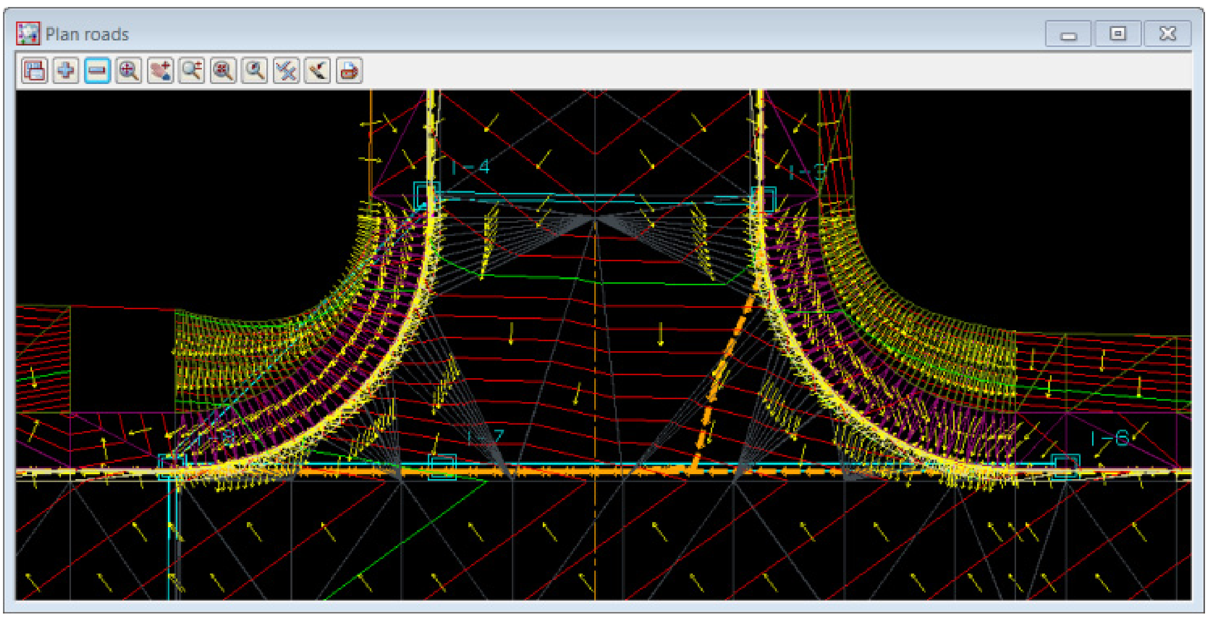


Use the CAD bar eye dropper on the right (same as) and select an existing overland/bypass string. The properties of the selected string will fill the CAD bar fields.

DRAG the **Create line** button and release at the **Create Line String** button.

When finished drawing the string press **ESC.**



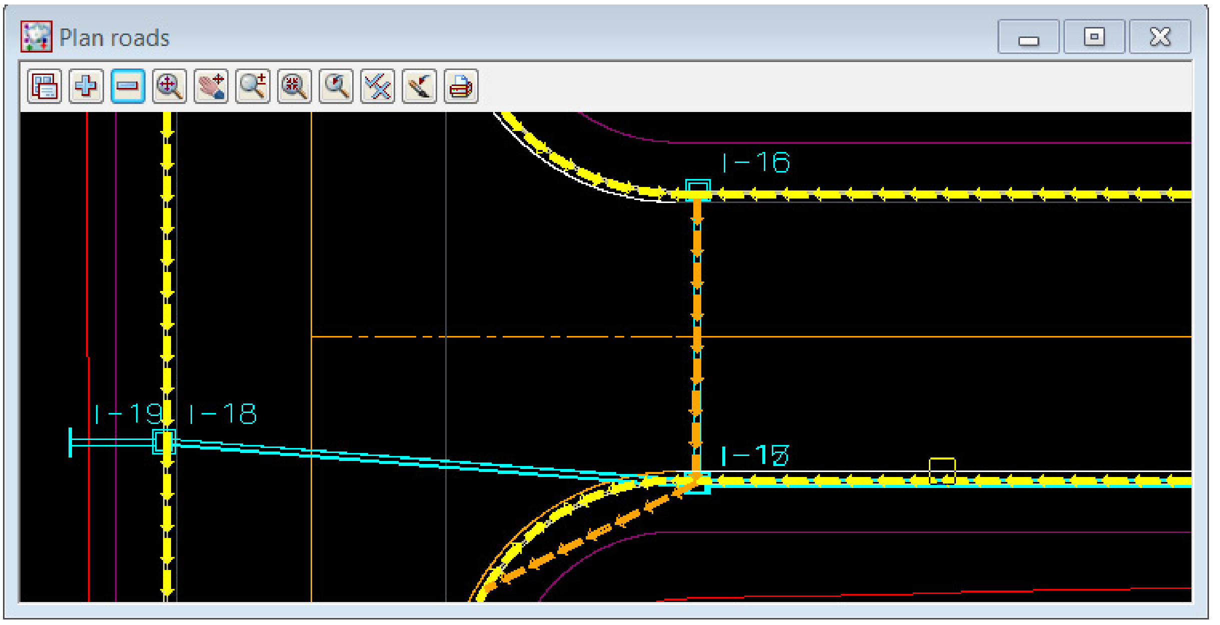


Starting at the upstream end. LB select an insertion point and MB or press return to accept the selection (the line style is shown after the point is accepted).

Do not stop drawing until you reach another sag or ongrade inlet (not a manhole). Overland flow lines often overlap as you see in the drawing above. If you stop short of the inlet, the bypass pit for the upstream inlet will be LOST.

Continue drawing until you reach the end of the flow path. Press **ESC** to finish drawing the string.

Frequently you will draw an overland flow path uphill and away from a sag inlet. It will usually go through the low point on the catchment crest and then downhill to the next inlet. No do this where road 2 intersects road 1 (see below).

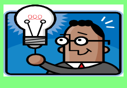


4.4 Creating Overland Flow Strings for Culverts

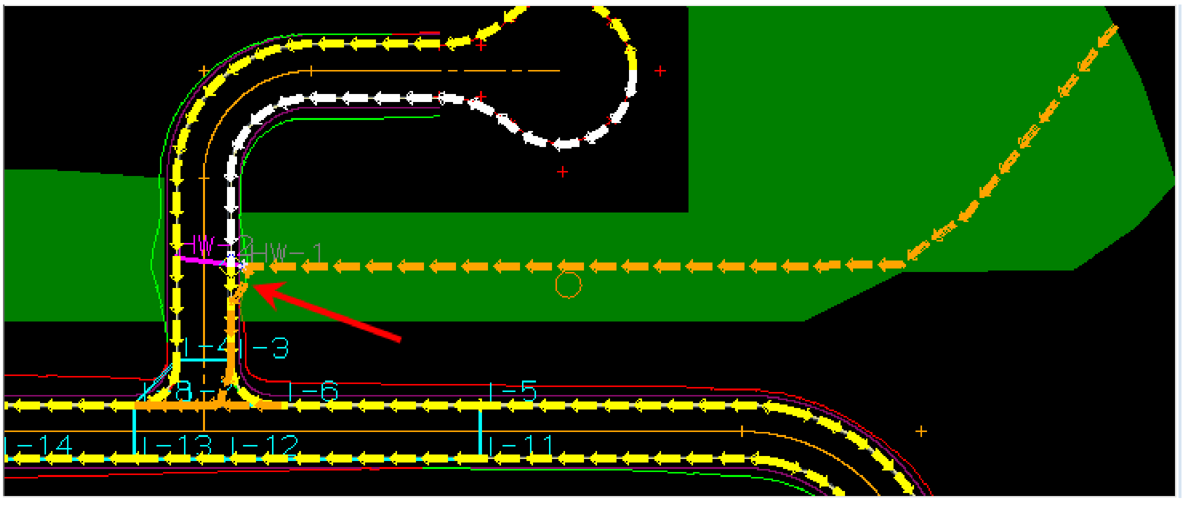
Culverts placed at sag location in the road will generally how the overflow across the crown of the the road and the upstream headwall grate level will be set the sag low point elevation of the road crown. For headwalls, there usually is not grate so this becomes the reference elevation for bypass.

The culvert in our example is not at a sag in the roadways so the water will overflow the southern side of the channel, cross the footpath, and flow down the roadway. We need to set the grate level to the bypass level along this path.

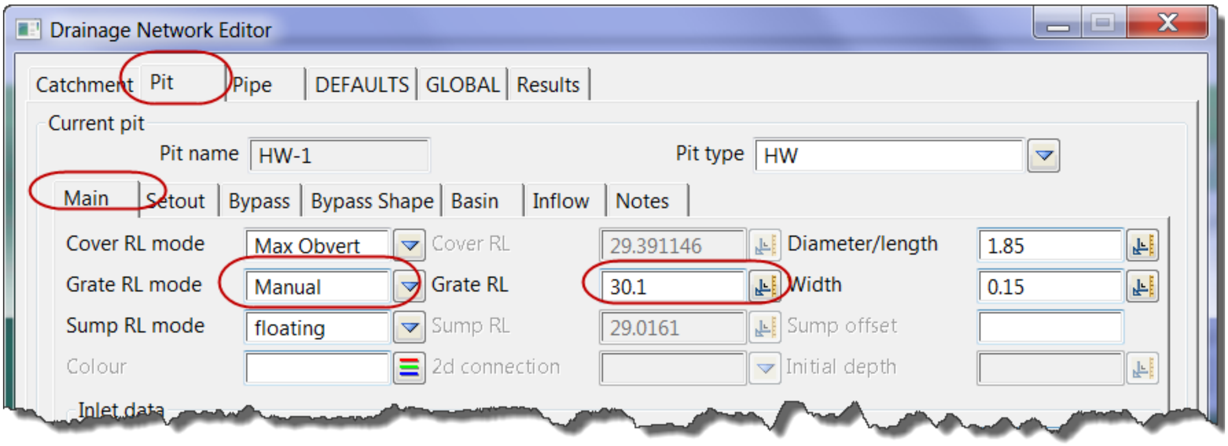
To draw the string, start at the top of the catchment so that the ***flooded width calculations*** in the approach channel will use the correct flows. Finish at I-3!



You could use the **CAD string->Copy** option to copy the tc string that was drawn in the model **dr Catch Reserve tc** and then **CAD vertex->Append** to add onto the downstream end of the new string and then **CAD vertex->Move** to adjust where need.



There is a low point where the south side of the channel joins the road (see red arrow). This overflow area will need to have erosion protection and detailed grading. Lets assume that the overflow area is regraded with a crest RL30.1. The grate level needs to be set to this value.



4.5 Set Inlet type to Sag or On grade

12d supports Manhole, on-grade and SAG configurations. The **inlet configuration** is set on the **Pit->Main**. This setting can be locked to the **Pit type** in the Drainage.4d setup file/database using the cap_config command. The remaining bypass data is found on the **Pit->Bypass** tab.

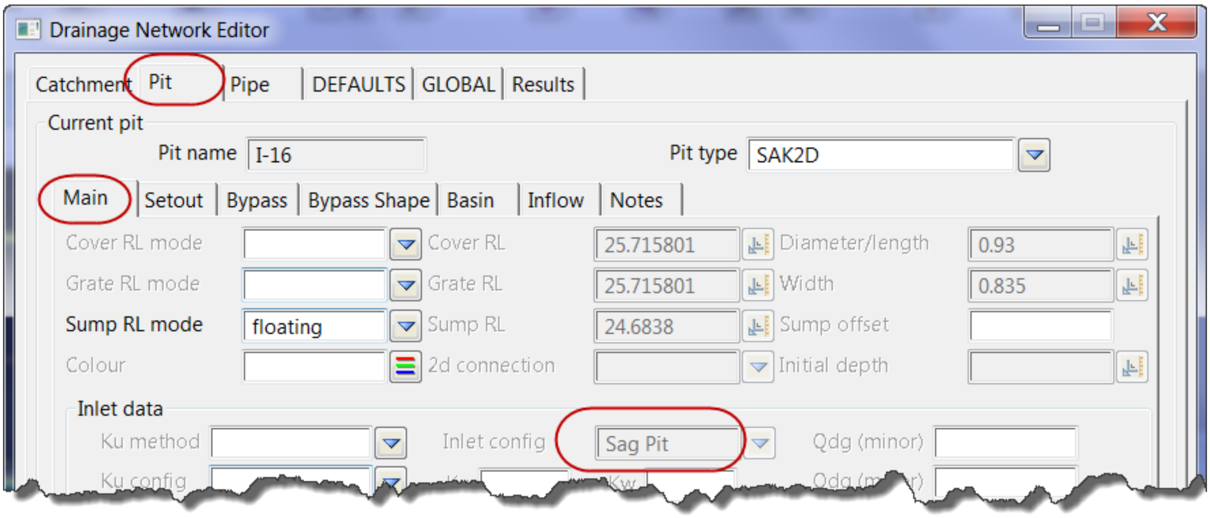
Pits **Inlet config** setting:

- **Manholes** will not receive bypass flow and cannot have catchments assigned to them.
- **On-grade** pits are pits where the water will flow past the pit if not captured (velocity and momentum are important for these inlets). Approach flow, road grade and road crossfall generally determine the inlet capacity equations/curves.
- **Sag pits** are located at sag locations where the water will pond around the pit if there is not enough inlet capacity (generally the water will stop at these locations and flow into the inlet). There is only one inlet capacity curve/equation per inlet and the ponding depth is used to determine the inlet capacity.

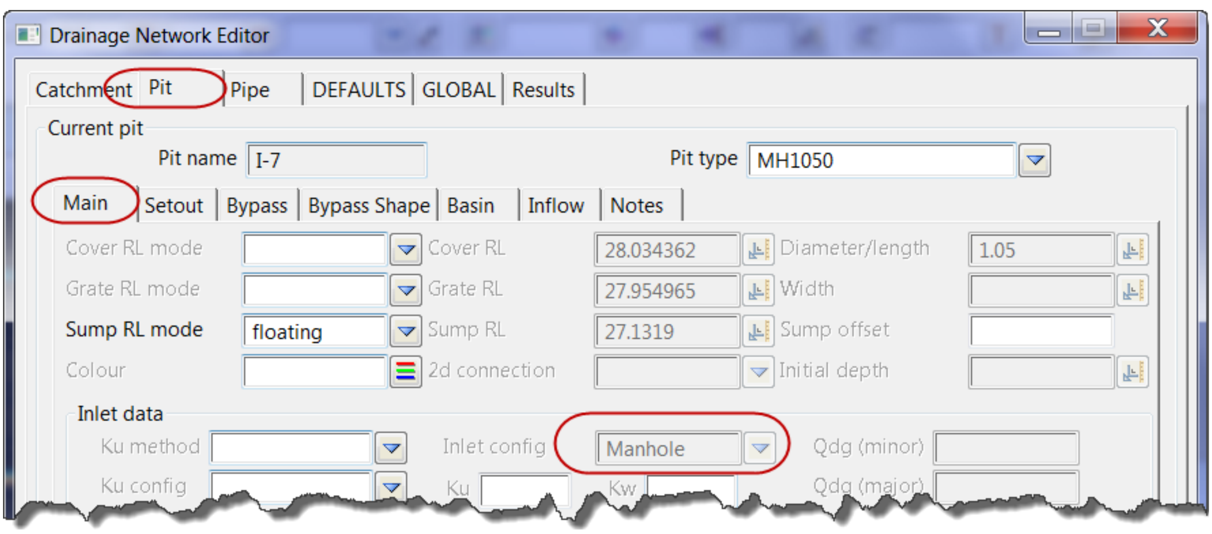
To view the inlet capacity curves used in this project, try Viewing Inlet Capacity Curves.

In this example the pit types have been set to an on grade grated, side entry inlet. We will now change the pit types for the manhole and the sag inlets.

Not all sag locations on a roadway behave hydraulically like a sag inlet. In general the flow should approach the inlet and not bypass over it. Sometimes the inlet acts as a sag inlet at low flows but like an on grade at design flows.



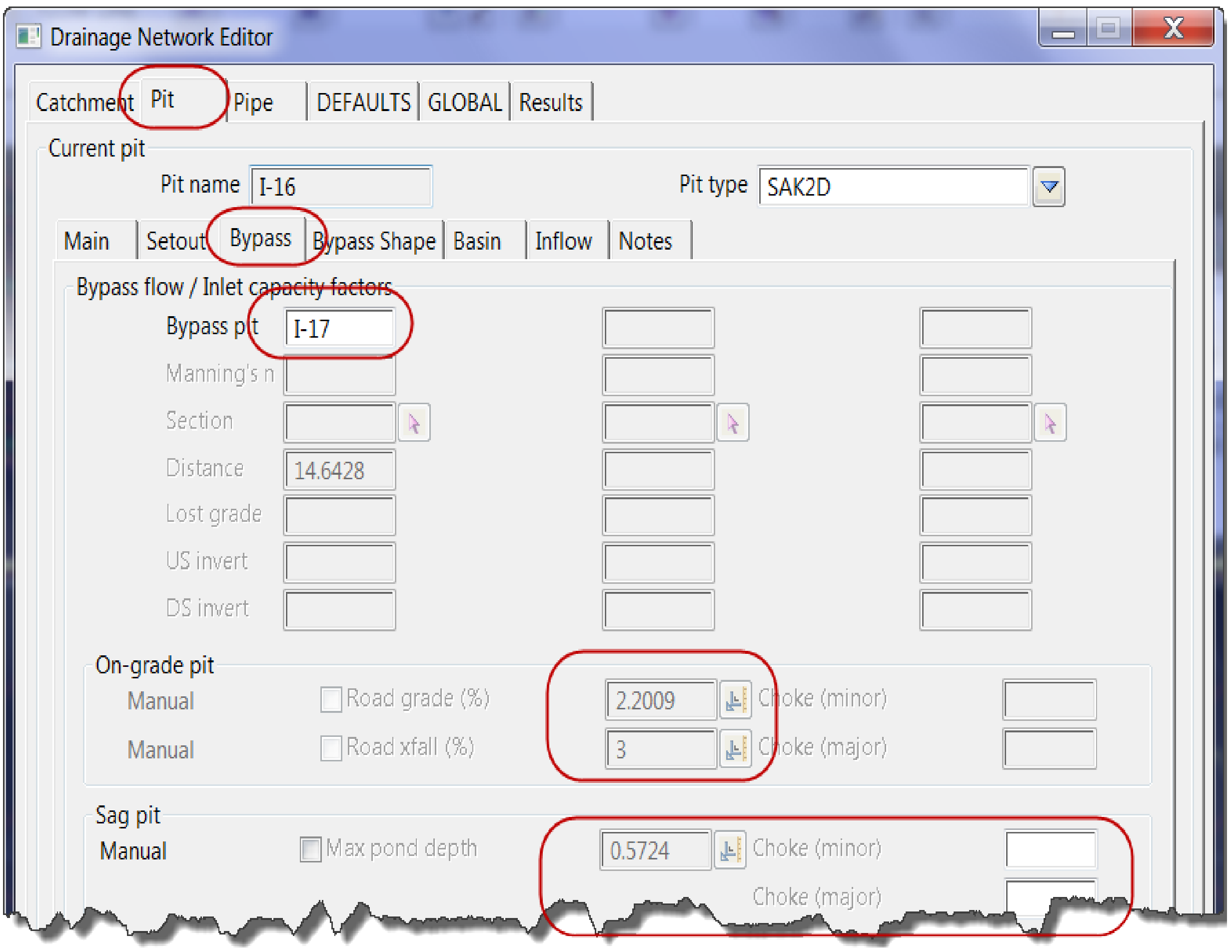
1. Movew to inlet marked I-16 above and select **SAK2D** for the pit type (you may be asked to confirm the changes from the drainge.4d file).
2. Note the **Inlet config** has been changed to **Sag pit** and it cannot be changed. This has been set in the drainage.4d file.



1. Select **MH1050** for the Wipit type (you may be asked to confirm the changes from the drainge.4d file).
2. Note the **Inlet config** has been changed to **Manhole**, the **Diameter/length** to 1.05 and the **Width** has been cleared. These changes are set in the drainage.4d file.

4.6 Set Pit Details - Calculate the Bypass Flow Data

Once the first Key Points of bypass flow are complete, you are ready to calculate the bypass flow data. Select **Set Pit Details** and then select a pit on a bypass flow string. The bypass data is found on the **Pit Main** and **Pit->Bypass** tabs.



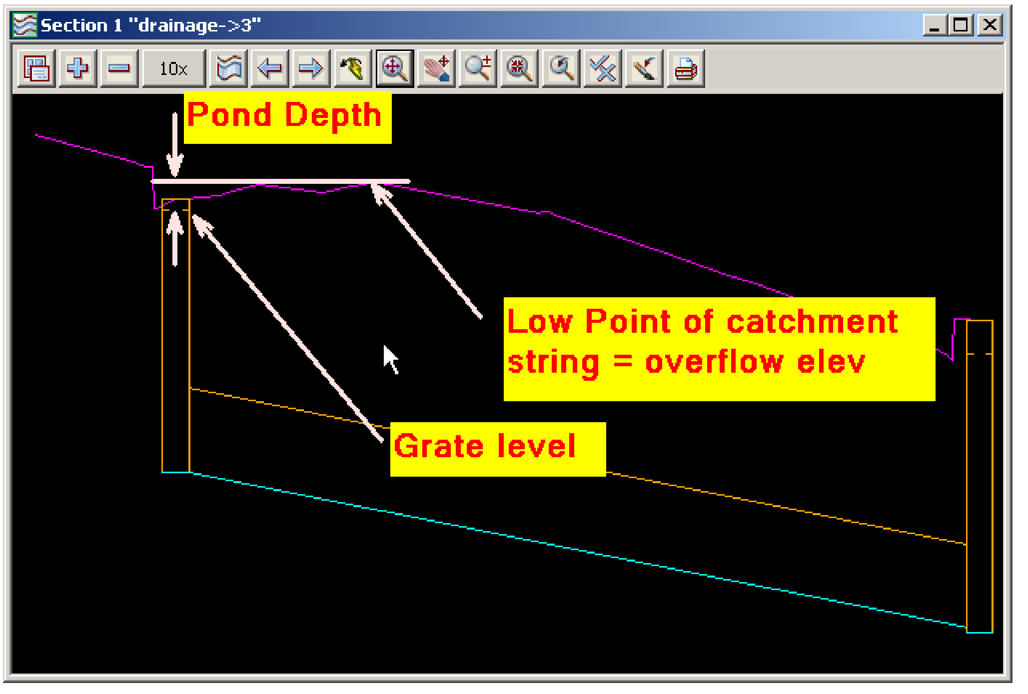
The downstream pit will now show in the **Bypass pit** field. If the bypass string does not go to another inlet (the network outlet is never an inlet) then the **Bypass pit** will be marked as **LOST**.

Always confirm bypass pits set to **LOST** are correct. An incorrect bypass to LOST would mean water is leaving your system when it should be included in the calculations.

If a setout string was found (see **Pit->Setout** tab), the **Road grade** will appear in grey. You may override this value by selecting the **Manual** tick box beside the value and entering your own value. If a road centre string was selected, the **Road xfall** field will also have a value.

If **Sag pit** was selected and a catchment string was selected, the **Max pond depth** will be displayed. The catchment strings from all 3 sets are draped onto the finish surface tin and the low point located. The **Max pond depth** is calculated as

Max pond depth = catchment string low point - Grate RL.



Negative Pond Depths

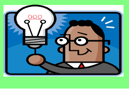
Negative pond depths are usually caused by one of two errors in input. The first may be that the grate level is too high. Often this happens when the Grate RL mode on the **Pit->Main** tab has not be set correctly or if **Sz + setout string** option is used the **Sz** value on the **Pit->Setout** tab has been entered correctly.

The second common error is that the catchment string has not been drawn around the crest of the catchment. The lowest section of the catchment string must be drawn carefully because it is the lowest point on the string that determines the overflow elevation. If in doubt, profile the catchment string with the design tin shown in the section view. Double check were the low point is.

4.7 Checking locations for measuring road grade, crossfall and max ponding depth

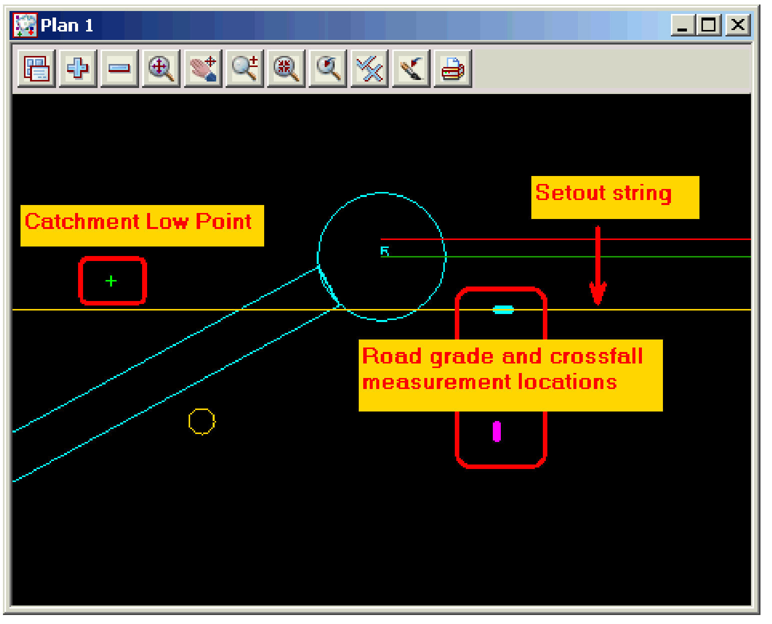
Strings indicating the location of inlet capacity parameters are generating in the **construction drainage data** model when **Set Pit Details** is selected.

The location of the low point, from all 3 catchment set polygons, is shown as a green vertex (plus sign).



Trouble finding the small green +? Select **Models->String info table** and select the sag marker line in the table. A large cross hair will show you the location in plan views.

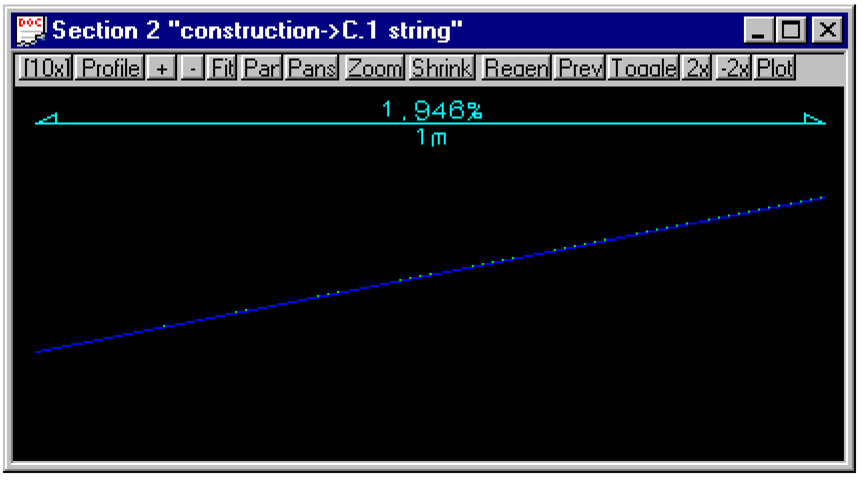
Verification strings in the same model confirm the locations where the road grade and crossfall have been measured. To check these strings add the **construction drainage data** model to the plan view. The following image shows a close up of the verification strings at a pit.



The green line indicates where road grade was measured and the magenta line indicates where the road cross fall was measured.

The default location of the road grade measurement is one pit diameter upstream of the setout point, along the setout string. The road crossfall is measure one pit diameter away from the setout point towards the road centreline. These are 3d super strings and therefore you may profile them in the section view. With the grades toggled on (check under **Toggle**) you can verify the slopes.

The location of the road grade and crossfall measurements can be changed in the **road design file**. The distance upstream to measure the road grade is controlled by the **Grade offset** column one the rows where the setout strings are defined. The road crossfall is controlled by the **Xfall offset** column on the rows where the road centre lines are defined. The distance the measurements are taken is controlled by the **Slope measurement distance**.



No flow into the pipe?

Pits with a bypass pit entered but no inlet capacity defined in the drainge.4d file will have a zero inlet capacity.

With Inlet config set to Manhole, the pit has no inlet capacity.

Important Notes

1. If no bypass flow string is supplied for a pit, the inlet capacity is set to 100%.
2. If you have a problem with the inlet capacity calculations, check the **Storm Analysis** hydrology report for details.

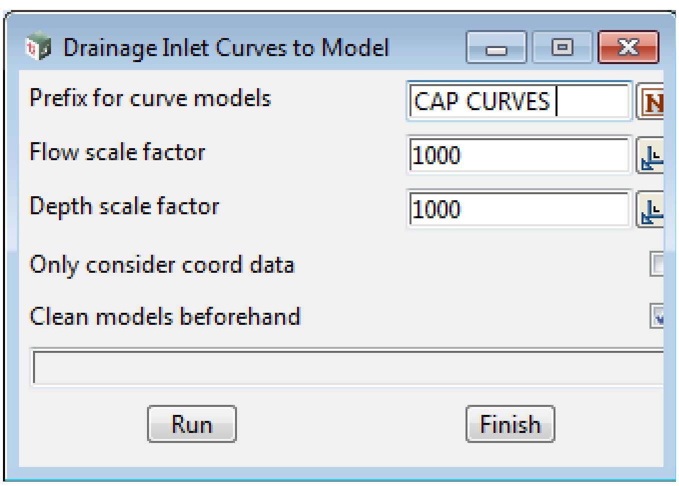
4.8 Viewing Inlet Capacity Curves

The inlet capacity information is stored in the drainage.4d file. We will now plot the inlet capacity curves in 12d. This will confirm what is in the drianage.4d but changing these plots will not change the inlet capacity calculations.

Put a **z** before **CAP CURVES** in the panel below. This will generally put these files at the end of your model list so they are not in the way when you are working.

From the main menu select

Design->Drainage-Sewer->More->Write inlet curves to model



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Prefix for curves models	model box	CAP CURVES	

All inlet curves for a manhole type will be placed in a model named with this prefix and the manhole type.

Flow scale factor	input	1000	

Inlet and bypass flow values will be multiplied by this value before creating the strings.

Depth scale factor input 1000

Depth values for sag curves will be multiplied by this value before creating the strings.

Only consider coord data tick box off

Inlet capacity curves may be expressed as formulas, curve coordinate or both. Selecting this box stops curves with only formulas from being plotted as strings.

Clean models before hand tick box on

The curve models are cleaned before the new strings are created

Run button

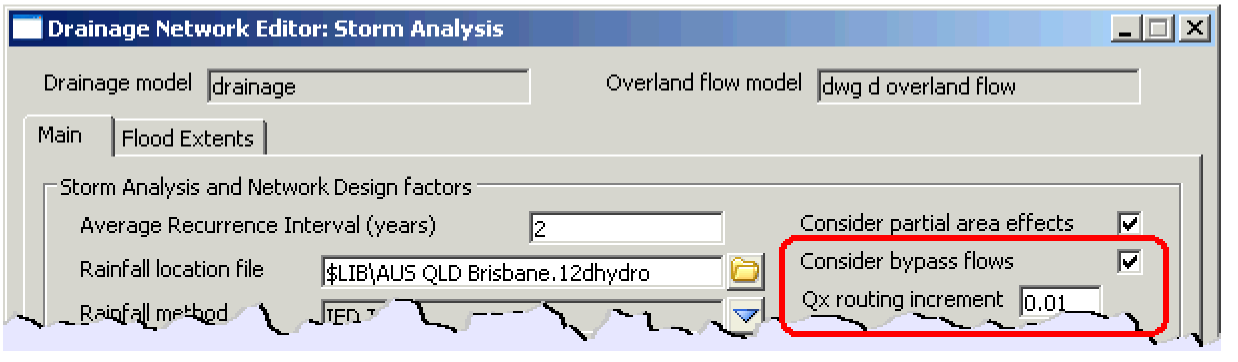
Models and strings are created from the drainage.4d file

Finish button

remove the panel from the screen

5.0 12d Storm Analysis Bypass and Flooded Width Calculations

Once the bypass pits are selected and the Drainage.4d setup file/database has been setup for bypass flow the storm analysis engine must have this feature enabled. Select **Consider bypass flow**.



Excess Flow

The **Qx** value controls how excess flow is handled in the bypass flow calculations. If the hgl at the pit reaches the grate level then no more water can enter the pit even if there is inlet capacity. The flow that will not enter the pit is considered excess flow. When a value greater than zero is entered here, the inlet will initially have its inlet capacity restricted by this value. Upstream inlets are done first as this may reduce the hgl in the downstream system. The system is automatically rerun adjusting the flows by this amount each time.

If the inlet capacity is reduced to zero and the hgl is still above the pit then water is removed from the pit and considered as Qs (surcharge flow). In the hydraulic reports this value is found as a negative **Inlet Flow Qi**.

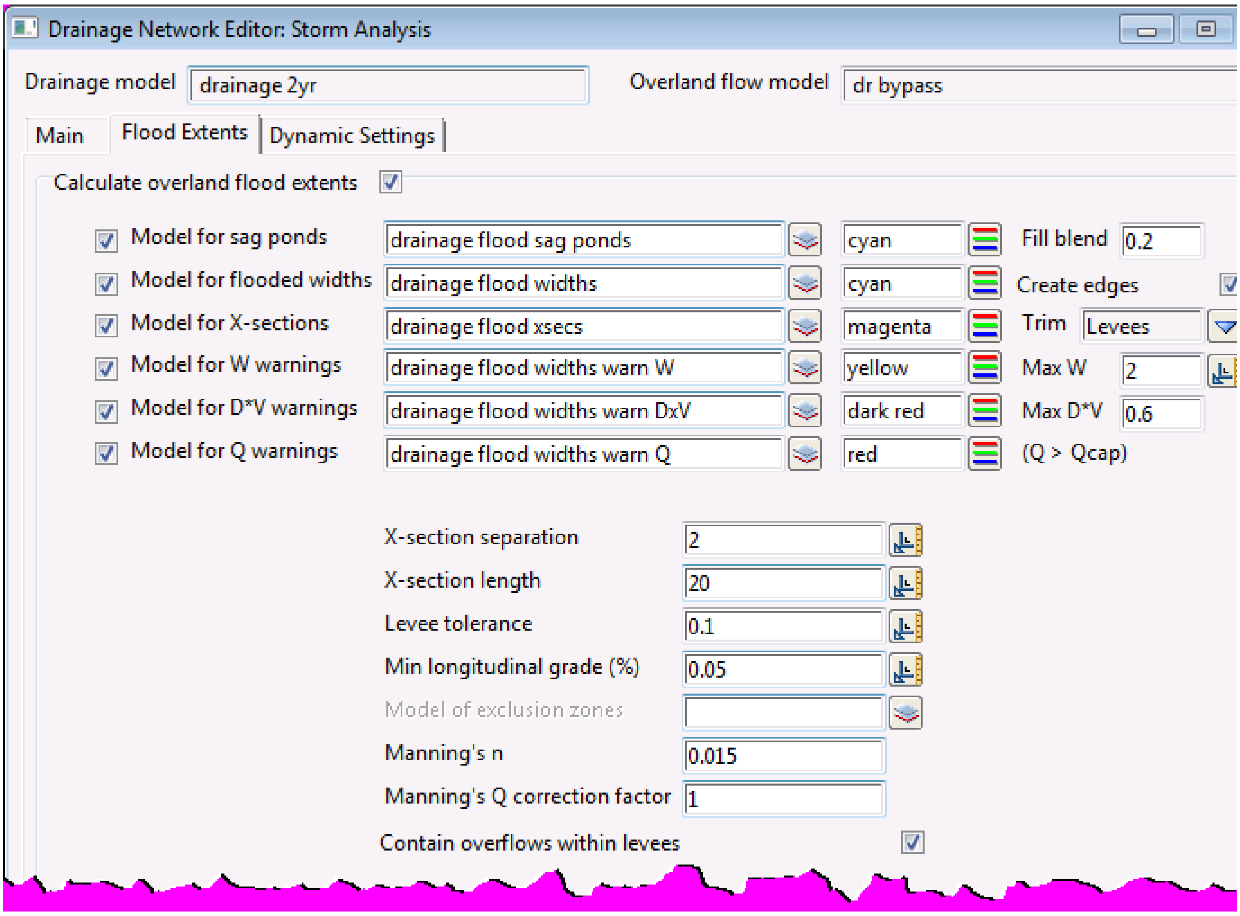
Overland Flow Calculations

The storm analysis engine will calculated flooded widths from normal depths along the flow path and ponding extents at SAG inlets. A bypass flow model (**Global-Utility Models** tab) is required for these calculations.

The **Utility String Editor** should be used where bypass flow strings combine at a pit. Without using this editor it is assumed that 100% of the catchment flow flows down each bypass flow string thereby overestimating the flooded widths.

The **Utility String Editor** could be used to change the Manning's along the string (approach channels to culvert for example) or the maximum flooded width warning limit (before a pedestrian crossing or a highway off-ramp).

The models and the default input data for these calculations are entered on the **Flood Extents** tab.

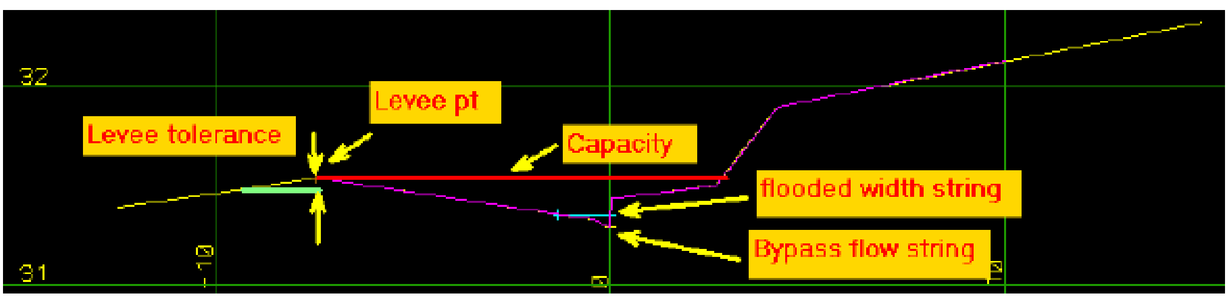


Select **Calculate overland flooded extents** to active the fields on the panel.

Model for sag ponds is used to hold strings that indicate the extent of flooding at the pits marked as SAG inlets. The total approach flow is used with the cap_curve_sag in the Drainage.4d setup file/database to determine the depth of flooding above the grate level. A closed contour at this flood elevation is then selected near the centre of the inlet. A super string is then created at this level with the colour and **fill blend** transparency selected (1.0 is solid).

Model for flooded width holds the strings indicating the normal depth - flooded width calculation results. These strings are created with a fixed elevation of the flood level. The strings will have string attributes with all of the calculations details.

Calculations



1. Cross sections are cut perpendicular to the overland flow string at the interval **X-section sepa-**

ration with a length of **X-section length**. No calculations are done for sect of the bypass flow string with in **Model of exclusion zones** polygon.

2. The x-section string is then trimmed using the **Trim** settings.The section above had a x-section length of 20 and was trimmed at the levee. The **levees** setting has the routine search for levee (high points) on either side of the low point near the bypass flow string. The levee points are found if sections drops more than the **Levee tolerance** value after the high point is found.
3. The flow for the section is calculated using the bypass flow string chainage to interpolate between the upstream bypass flow and the downstream approach flow. The percentage of catchment flow used in the calculation of the approach flow for this bypass string may be changed using the **Utility String Editor.**
4. The slope of the surface near the bypass flow line is measured for each section. If the slope is less than **Min longitudinal grade (%)** then this cross section is skipped in the calculations.
5. **Manning n** value is the default roughness used in the calculations. This value may be changed at any vertex along the bypass flow string using the **Utility String Editor**.
6. The flow's calculated at the indicated level are multiplied by the **Manning's Q correction factor**. A factor of 0.8 would cause the flooded widths to increase and the road capacity to decrease.
7. The maximum depth calculated will be at the point where water overflows the edges of the section (**Road capacity)** unless **Contain overflow within levees** is selected. This will cause frictionless vertical walls to be placed at the ends of the section so that higher water level may be calculated when the flow is greater that the road capacity.

Warning Models

Model for W warnings will contain copies of the flooded width strings with the colour selected if the flooded width is greater than the default **Max W** value.

Model for D*V warnings will contain copies of the flooded width strings with the colour selected if the depth * velocity is greater than the **Max D*V** value.

Model for Q warnings will contain copies of the flooded width strings with the colour selected if the flow is greater than the capacity of the section. The capacity is the flow where the water over tops one of the sides of the section.

6.0 Utility String Editor

Position of option on menu: **Design =>Drainage-Sewer =>Utility String Editor**

This editor is used to edit properties of the strings used by the DNE. Properties always change at an existing vertex.

Cross Section Strings (Manning's n) - set the left and right bank n values. the centre n value is assigned by the DNE.

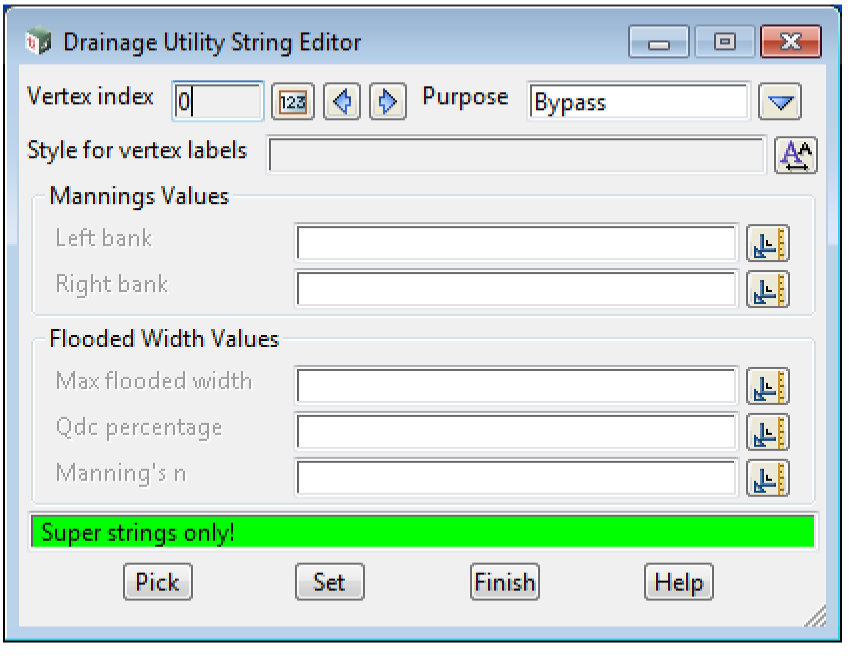
Flooded Width Values on Bypass Flow Strings - the default setting of the bypass flow strings when used to calculate flooded widths in the Drainage Analysis.Changes are in effect until the end of the string or it has been re specified at vertex at a higher chainage.

Vertex labels are created whenever properties are set (Texstyle is required).

Usage

First the string is selected at the vertex where the values are to be assigned. Next select the purpose of the string (cross section or bypass) to unlock the appropriate fields. A textstyle favourite is required as the vertex is labelled with the assigned values. Enter the values into the fields and then select Set to set the values as vertex attributes and create the label as a vertex annotation.

On selecting the **Utility String Editor** option, the **Drainage Utility String Editor** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Vertex index	vertex	selected vertex	

Once the string is selected use this to move between vertices

Style for Vertex Labels textstyle favourite textstyles

A label is created on the vertex using this textstyle favourite

Left bank Input

mannings n value, to delete clear and select set

Right bank Input

mannings n value, to delete clear and select set

Max Flooded with Input

This changes the threshold were warning bars are created during flooded width calculations. It remains in effect till the next change or the end of the bypass flow string.

Qdc percentage Input

This changes the percentage of the Qdirect+Qcatchment that is used to interpolate the discharges during flooded width calculations. It is generally set on the first vertex past upstream pit and remains in effect till the next change or the end of the bypass flow string. The flow changes from the bypass flow at the upstream pit to this value/100(Qdirect+Qcatchment).*

Manning's n Input

This changes the Manning's n value used for during flooded width calculations. It remains in effect till the next change or the end of the bypass flow string.

Pick button

Use this button to select the string. Select near the vertex you want assign the values to.

Set button

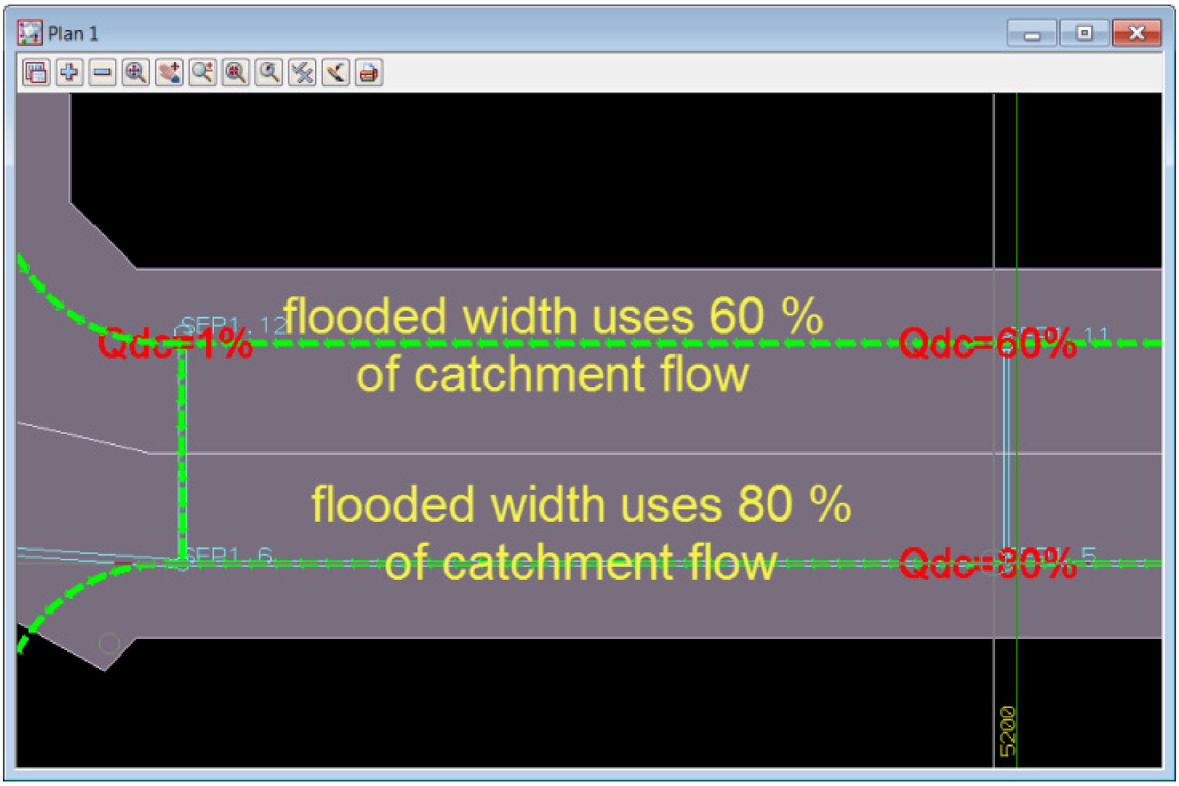
Creates the attributes and the label on the vertex.

Finish button

Removes the panel from the screen.

Help button

Launches the 12d help



In the example above, The Qdc is set to 80% at the eastern end of the catchment. The Qdc is set to 2% for the water overtopping the road. The 18% of the area is not considered large enough to do the flooded width calculation for. If desired, another bypass flow string approaching from the east could be drawn but the final bypass string to the west (direction during bypass should remain).

7.0 Drainage.4d setup file/database

The drainage.4d file controls many of the settings for the pit and pipes types inside 12d. This section details the format and features of the drainage.4d file. Changes to this file take effect only after 12dmodel has been restarted.

Press the **F1** button for 12d help. Select the **index** tab and type **drainage.4d** and select **drainage.4d** from the list. Full documentation of the drainage.4d file is found here.

7.1 Editing the drainage.4d file

When 12d Model starts up, it checks to see if an environment variable called DRAINAGE_4D exists and if it does, then the file it points to is used to provide the available types of manholes (maintenance holes) and pits. If the environment variable is not set, then 12d Model searches for a file called **drainage.4d** in the standard 12d Model search sequence for set up files.

12d model ships a **drainage.4d** file in the “program files\12d\12dmodel\10.00\set_ups” folder. **Do NOT** change this file. Copy it into your user folder “\12d\10.00\User” and edit it there. Files in the user folder are used by preference and they are never over written by a 12d update.

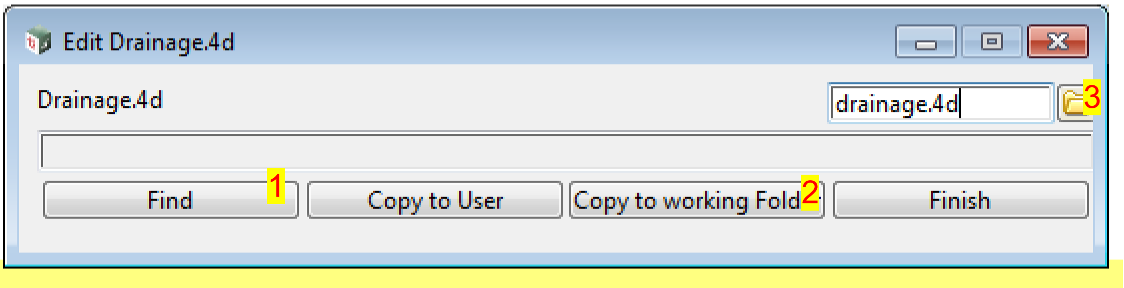
File Format

1. spaces in text - any text string that includes spaces or only numbers, must be enclosed in double quotes **"**.
2. comments - anything after // until the end of the line is ignored.
3. blank lines - blank lines are ignored

Duplicate definitions are not allowed. ie. you cannot have 2 manhole types with the same name.

From the main menu select

Design->Drainage-Sewer->More->Edit drainage.4d



1. Select the **Find** button to search the 12d path for the current drainage.4d file.
2. If the file is found in the setups folder, select **Copy to working folder** and then **Find** again.
3. Now that the file is found in the working folder, select the More info button (the folder) and then **Open** to edit the file.

You must restart 12d for these changes to become active. Select Project->Restart!

At startup, refer to the output window for any error messages. 12d will print the line number where it gave up. When looking in the long list of files, the drainage.4d file is loaded after the shp files. Sometimes one error will result in many other errors to follow so fix the first one, save your changes and restart again.

7.2 Creating and Modifying Manhole Types

We will now create a new manhole type for an enlarged grated inlet. For training purposes we will do it in steps restarting 12d model between each step.

1. Create a new manhole type
2. Add manhole commands to set the size and manhole thickness
3. Add manhole commands to create pit connection points where 2 pipes may be connected on the long side.
4. Add manhole commands to set the manhole level modes
5. Add a manhole command to create a user defined attribute.

7.2.1 Create a new manhole type

The order the manhole commands appear in the file is the order they appear in the type drop down list in the DNE.

1. Go to the bottom of the file and type the following.
   Manhole "AK2D enlarged" {
   }
2. Save the file
3. In 12d model close the Edit drainage.4d panel.
4. From the 12d model main menu select **Project=>Restart**
5. Check the output window for errors (just below the shp files.. there are easy to find as there are so many of them. Your should see the message
   File found <C:\12d\10.00\Courses\drainage\drainage.4d>
6. With the DNE, change inlet I-5 to the new manhole type **AK2D enlarged**. Note that you are free to change the diameter/length, width and many of the other DNE Pit field.s

Manhole must be capitalised.

Use “ (double quotes) not the single ‘ OR 2 single side by side ‘’.

Use {} NOT () or []

7.2.2 Add manhole commands to set the size and manhole thickness

The **mhsize** command sets the DNE length and width fields (rectangular manhole) and locks them from been changed.

mhdiam is used to set the diameter (circular manhole) and clear sthe width,

The **mhthickness** command block sets the mnahole wall thickness. The **diam_thickness** command has the manhole depth followed by the wall thickness. The line **diam_thickness** may be repeated if you want the pit wall thickness to change if the depth becomes greater than the specified value. Depth is measured from the cover level to the sump level.

The indenting is not required but makes the commands easier to read.

1. add a few blank lines between the braces in your new manhole definition and type the new commands shown in bold.
   ```
   Manhole "AK2D enlarged" {
       mhsize 1.86 0.835
       mhthickness {
           diam_thickness 0.000 0.150
       }
   }
   ```
2. Save the file
3. From the 12d model main menu select **Project=>Restart**
4. Check the output window for errors.

7.2.3 Add manhole commands to create extra connection points

With connection points enables on the DNE->Global tab, a connection point is created at the mid points of the internal sides of the manhole. Additional connection points may be added with their offsets from the manhole centre.

1. add a few blank lines before the last brace in your new manhole definition and type the new commands shown in bold.
   ```
   Manhole "AK2D enlarged" {
       mhsize 1.86 0.835
       mhthickness {
           diam_thickness 0.000 0.150
       }

        con_points "AL2D" {
           con_point  0.930   0.000
           con_point  -.465   0.4175
           con_point   .465   0.4175
           con_point -0.930   0.000
           con_point  -.465  -0.4175
           con_point   .465  -0.4175
       }
   }
   ```
2. Save the file
3. From the 12d model main menu select **Project=>Restart**
4. Check the output window for errors.
5. Use **Strings->Points Edit->Move** to move the end of the pipe that connects to I-5. You will see that it will snap to 2 points on the long side and 1 point on the short sides.

7.2.4 Add manhole commands to set the manhole level modes

We will now set the construction setout point and grate level to be obtained from the road setout string, the cover level to be 0.150 higher than the grate and a sump depth of 200mm.

1. add a few blank lines before the last brace in your new manhole definition and type the new commands shown in bold.

```
Manhole "AK2D enlarged" {
    mhsize 1.86 0.835
    mhthickness {
        diam_thickness 0.000 0.150
    }

     con_points "AL2D" {
        con_point   0.930    0.000
        con_point   -.465    0.4175
        con_point    .465    0.4175
        con_point  -0.930    0.000
        con_point   -.465   -0.4175
        con_point    .465   -0.4175
    }
attribute_integer "cover rl mode"        8
attribute_real    "setout adjustment z"   0.150
attribute_integer "grate rl mode"        1
attribute_real    "sump offset"          -0.200

attribute_integer "setout xy mode"       1
attribute_integer "setout z mode"        1

}
```

2. Save the file
3. From the 12d model main menu select **Project=>Restart**
4. Check the output window for errors.
5. Use the DNE to select the i-5 inlet. You will be prompted to verify that you want the design parameters to change for each attribute you have added.

7.2.5 Add manhole commands to create a user defined attribute

We will now create a manhole text attribute called “reference dwg” that we can add our drainage plan plot.

1. add a few blank lines before the last brace in your new manhole definition and type the new commands shown in bold.

```
Manhole "AK2D enlarged" {
    mhsize 1.86 0.835
    mhthickness {
        diam_thickness 0.000 0.150
    }

     con_points "AL2D" {
        con_point   0.930    0.000
        con_point  -.465    0.4175
        con_point   .465    0.4175
        con_point  -0.930    0.000
        con_point  -.465   -0.4175
        con_point   .465   -0.4175
    }
attribute_integer "cover rl mode"        8
attribute_real    "setout adjustment z"    0.150
attribute_integer "grate rl mode"         1
attribute_real    "sump offset"          -0.200

attribute_integer "setout xy mode"        1
attribute_integer "setout z mode"         1

attribute_text "reference dwg" "Dwg 47.1 Rev A"

}
```

2. Save the file
3. From the 12d model main menu select **Project=>Restart**
4. Check the output window for errors.
5. Use the **DNE** to select the pit. This will set the attribute for this manhole. The **Set Pit details** would do this for all manholes in the model.
6. Select **Apply** or more to another pit.
7. Use **Strings->Properties->Attributes** to select the manhole and on the pit table you will find the new attributes “reference dwg”.

7.3 Pit Inlet Capacities

The pit inlet capacity tables contained within the Drainage.4d setup file/database are used by the 12d drainage modules and exported to other design packages in different ways but with a common philosophy.

These tables may be plotted in 12d Model using Viewing Inlet Capacity Curves.

7.3.1 On grade pits

The grade and crossfall values for the tables are threshold values, i.e. the next set of capacity factors/curves will not be used until the measured crossfall and grade are equal to or exceed the threshold values for the curves. Curves with the same crossfall threshold are grouped together and the correct group is selected first. The road grade is then used to select the curve within the crossfall group.

Each inlet selects one road grade curve from the drainage.4d file. 12d Model does not interpolate between the curves.

Some simple sample pit definitions follow to demonstrate how the pit inlet capacities are calculated. The drainage.4d file supplied in the library has extensive curve data from model testing.

```
Manhole "SEP 25" {

  mhsize  1.200 0.900
  mhdesc  "SEP with 25 l/s"
  mhnotes ""
  mhgroup "SA"

  cap_multi   1.0
  cap_fixed   0.025

}

Manhole "SEP 50 percent" {

  mhsize  1.200 0.900
  mhdesc  "SEP with 50%"
  mhnotes ""
  mhgroup "SA"

  cap_multi   1.0
  cap_percent 50.

}

Manhole "SEP Grade x 10" {

  mhsize  1.200 0.900
  mhdesc  "SEP with 25 l/s"
  mhnotes ""
  mhgroup "SA"

  cap_curve_grade "curve 1" {
    road_grade 0.0
    cap_multi   1.0
    cap_fixed   0.010
  }

  cap_curve_grade "curve 2" {
    road_grade  2.0
    cap_multi   1.0
    cap_fixed   0.020
```

```
  }

  cap_curve_grade "curve 3" {
    road_grade  3.0
    cap_multi   1.0
    cap_fixed   0.030
  }

  cap_curve_sag "curve sag" {
    cap_multi   1.0
    coord 0.0  0.000
    coord 0.1  0.010
  }
}
```

7.3.2 Exporting Inlet Capacity to External Programs

Drains Version 1 and ILSAX

The cap1, cap2, cap3 and cap4 values are used to describe the inlet capacity of the pit as described in their user manuals.

Drains Version +

The 12d inlet curve names are exported to Drains as the pit family.

xpswmm, xpstorm and RAT-HGL

If cap2, cap3 and cap4 are all equal to zero then a fixed inlet capacity equal to cap1 will be exported to RAT-HGL. If the sum of these three values is greater than zero then a pit type will be created in the format of **pit_type-crossfall-roadgrade**. For example SA2-3-2 for a SA2 pit with a road crossfall of 3% and a road grade of 2%. A rating curve with this name will have to exist inside RAT-HGL. 12d has no way of transferring the rating curve itself into RAT-HGL.

PC Drain

Similar to RAT-HGL, PC Drain has it own rating curves defined internally. The road grade is sent as a separate piece of data to PC Drain so that the pit inlet capacity may be determined.

PC Drain places a suffix code in the pit type to specify that the pit is a SAG pit. For example an 9S.03 indicates that pit type 9 is a sag pit and the maximum depth before bypassing is 30mm. 12d model does this automatically when exporting.

7.3.3 Add manhole commands for inlet capacity

Manhole inlet configuration and bypass pit entries determine if these inlet capacity commands are used. Both may be set in the Drainage Network Editor.

cap_config

The inlet configuration may be set via the following command

cap_config x

Mode	**x**
Manhole	m
Ongrade	g
Sag	s

Inlet Capacity Equation

The inlet capacity equation is built up with 3 optional components (**single polynomial** + **curve polynomial** + **curve coordinates**). Generally, only one of the 3 components is used for each manhole type but they may all be used if desired.

inlet capacity = inlet efficiency * inlet multiplier *
[**single polynomial** + curve multiplier (**curve polynomial** + **curve coordinates**)]

An inlet efficiency (choke factor) is specified in the Drainage Network Editor. An inlet efficiency (choke factor) of 0 would stop all water from entering the inlet.

Curve Coordinates (On grade and SAG)

For on-grade and sag inlets, the inlet capacity may be determined by entering coordinates along the inlet capacity curve. These coordinates are usually obtained from hydraulic model studies or analytical methods such as HEC-22.

For on grade inlets, the coordinates are Qapproach and Qin, and the curves may change with road grade and cross fall threshold values. The inlet capacity curves are never extrapolated.

```
Example

Manhole "Ongrade Inlet A" {
        cap_config G
        cap_curve_grade "0.5G" {
                road_grade  0
                coord 0.000 0.000
                coord 0.060 0.060
                coord 0.140 0.112
                coord 0.260 0.174
                coord 0.430 0.244
                coord 0.500 0.270
        }
        cap_curve_grade "1G" {
                road_grade  0.75
                coord 0.000 0.000
                coord 0.060 0.060
                coord 0.140 0.108
                coord 0.260 0.164
                coord 0.430 0.227
                coord 0.500 0.248
        }
}
```

For sag inlets, the coordinates are Depth (base units) and Qin, and there is only one curve. Each curve has a curve multiplier specified with a cap_multi parameter (discussed below).

Example

```
Manhole "SAG Inlet A" {
        cap_config S
        cap_curve_sag "SAG" {
                coord 0.000 0.040
                coord 0.045 0.101
                coord 0.070 0.151
                coord 0.095 0.245
                coord 0.120 0.302
                coord 0.170 0.347
                coord 0.220 0.371
                coord 0.270 0.391
        }
}
```

Inlet Curve Block Commands

Inlet curve blocks may be specified for both on-grade or sag inlets. Inside the curve block you may include the Polynomial Inlet Capacity Commands and Coordinate Inlet Capacity Commands.

```
cap_curve_grade "unique name for the pit type" {
        road_grade x.xx
        road_xfall x.xx
}
```

Inside the cap_curve_grade block the road grade and road crossfal threshold values (percent) may be set. The road grade and crossfall are calculated by the Drainage Network Editor. When the 12d analysis engine selects the inlet curve, all curves with the same road_xfall are grouped together and then within the crossfall group the road_grade curves is selected.The inlet curve with the maximum grade threshold that is less than or equal to the road grade is selected.

Rules for 'cap_curve_grade' entries:

Only applicable to on-grade pits.
All cap_curve_grade names must be unique within a Manhole block
If both 'road_grade' and 'road_xfall' entries are omitted, only one cap_curve_grade entry is allowed within a pit.
The cap_curve_grade 'coord' entries (if used) must be in order of increasing Qa.

```
cap_curve_sag "unique name for the pit type" {

}
```

Rules for 'cap_curve_sag' entries:

Only applicable to sag pits.
Only one cap_curve_sag entry is allowed within a pit, and it must have a valid name.

Coordinate Inlet Capacity Commands

The coord command must be used inside the cap_curve_grade or cap_curve_sag grouping

coord x.xx y.yy

x.xx must be in increasing order.

For cap_curve_grade group, the coord command has the parameters Qapproach and Qin

For cap_curve_sag group, the coord command has the parameters Depth and Qin

Please continue to the next section Drainage Definitions - Pipe Types.

7.4 Drainage Definitions - Pipe Types

Pipe types may be used to set the following pipe properties via the DNE.

- pipe nominal/actual diameters and thickness
- roughness method and value
- rational method design mode and design percent depth
- minimum pipe height for the rational design engine
- user defined pipe attributes

Each definition (pipe block) in the file begins with the key word **Pipe**, followed by the pipe type and then curly braces **{ }**. The order that the definitions appear in the file determines the order they appear in the drop down lists inside 12d Model.

The minimum requirement for a pipe type definition is

Pipe “name” {

}

7.4.1 Pipe Thickness

```
pipethickness           {
            diam_thickness   x.xxx y.yyy a.aaa b.bbb c.ccc d.ddd
        }
```

x.xxx	nominal diameter choices will appear in the DNE->Pipe->Diameter drop down
y.yyy	internal diameter (base units) will be entered into the DNE->Pipe->Diameter field
a.aaa	optional top thickness (base units) 0.000 if omitted
b.bbb	optional bottom thickness (base units) top thickness if omitted
c.ccc	optional left thickness in direction of chainage (base units) top thickness if omitted
d.ddd	optional right thickness in direction of chainage (base units) top thickness if omitted

An example pipe definition follows.

```
Pipe "2" {

 // HUMES class 2 rubber ring joint pipe thicknesses
 pipethickness {
  // nominal_diam   internal_diam  top   [bottom] [left] [right]  (looking in the direction of chainage)
  diam_thickness  0.225    0.229     0.025
  diam_thickness  0.300    0.300     0.031
  diam_thickness  0.375    0.375     0.035
  diam_thickness  0.450    0.450     0.042
  diam_thickness  0.525    0.534     0.041
  diam_thickness  0.600    0.610     0.044
  diam_thickness  0.675    0.685     0.048
  diam_thickness  0.750    0.760     0.052
  diam_thickness  0.825    0.838     0.054
  diam_thickness  0.900    0.910     0.066
  diam_thickness  1.050    1.070     0.075
  diam_thickness  1.200    1.220     0.076
  diam_thickness  1.350    1.370     0.077
  diam_thickness  1.500    1.524     0.095
  diam_thickness  1.650    1.676     0.095
  diam_thickness  1.800    1.828     0.102
 }

}
```

7.4.2 Pipe Attributes

attribute_integer	"attribute name1" x	x is an integer value (no decimal, stored exactly by computers
attribute_real	"attribute name2" x.xxx	x is a real value (used to store numbers with decimals or very large or very small numbers
attribute_text	"attribute name3" "text"	text is a series of words or numbers not intended for calculations

The following special attribute commands create/modify an attribute as described above but these attributes also control calculations performed by the set pit details button on the Drainage Network Editor. The DNE fields will be locked when these attributes are defined for the selected pipe type. If these attribute are not defined for the selected pipe type the DNE field will not be locked and remain unchanged.

roughness_n	x.xx>	DNE field ->Pipe=>Main=>Roughness DNE field ->Pipe=>Main=>Roughness type (set to Manning)

roughness_k	x.xx	DNE field ->Pipe=>Main=>Roughness		
	Colebrook k roughness value in millimetres			
		DNE field ->Pipe=>Main=>Roughness type (set to Colebrook)		
attribute_real	x.xx	DNE field ->Pipe=>Design=>Min pipe height		
	min height in base units			
attribute_integer	"design size mode"		x	DNE field ->Pipe=>Design=>Design mode
	Mode			**x**
	Pressurised Pipe: Freeboard design			0
	Part-full Pipe: Freeboard design			1
	Part-full Pipe: Flow depth design			2
	Open Channel: Freeboard design			3

attribute_real	"design percent depth"	x.xx	DNE field Pipe=>Design=>Flow-depth at pipe entrance

An example of these setting follows:

```
Pipe "CHNL GRASS EXISTING 1" {  //Open Channel 1
  roughness_n 0.040
  attribute_integer "design size mode"     3  // open channel mode
}
```

8.0 Open Channel Flow

12d can model flow in open channels a trapezoidal sections. Suggestions for drainage network editor settings are listed below.

Generally, simple channels leading into a pipe network are often not modelled as open channels. The flooded width calculations can perform flooded width calcs (normal depth however) along the channel and the depth at the headwall is calculated using the headwall ku method.

If gradually varied depth/backwater calculations are desired then the channel will need to be modelled as a drainage string. Also if the pipes discharge into a channel then this channel will also need to be modelled as a drainage string.

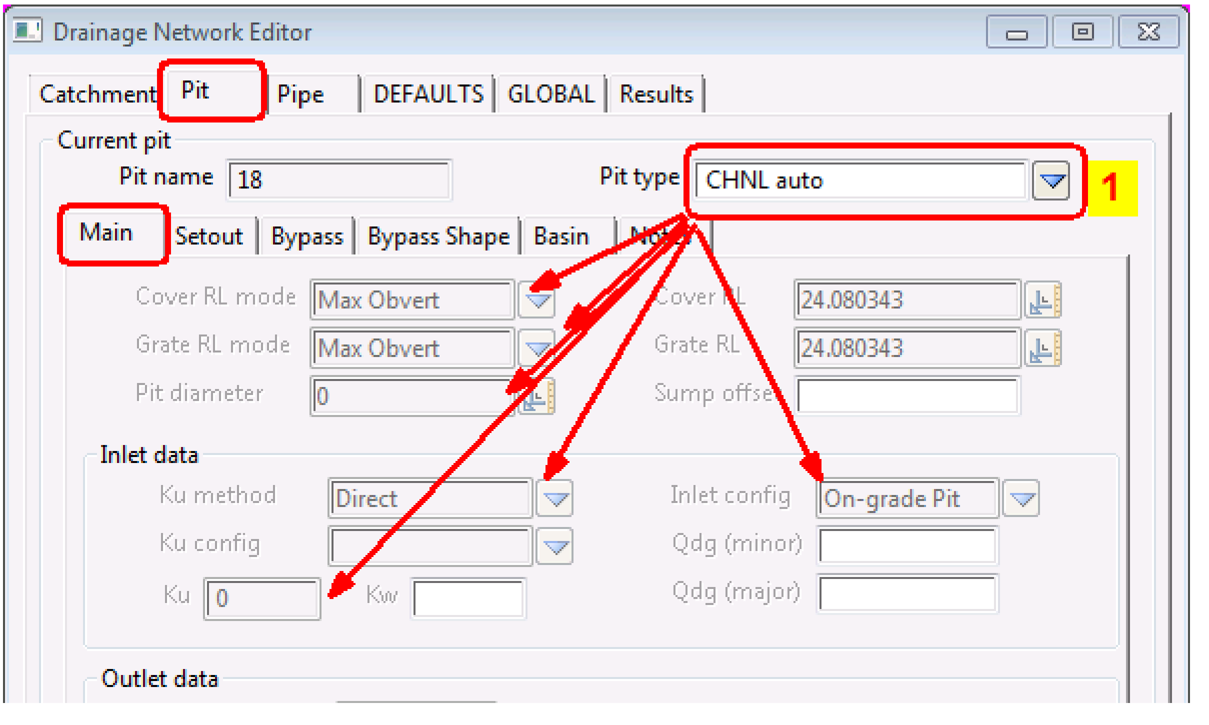
Key points

The pit grate level must always be at or above the top of the open channel conduit.

Changes in vertical grade and horizontal alignment require a pit in the drainage string. The pit diameter is usually set to zero.

Channels may exist in the tin or you may be proposing a channel to be cut into the tin. The grading and grate level modes will be different for each case. These channel type will be referred to as "Existing" or "Proposed"

Pit-Main Tab

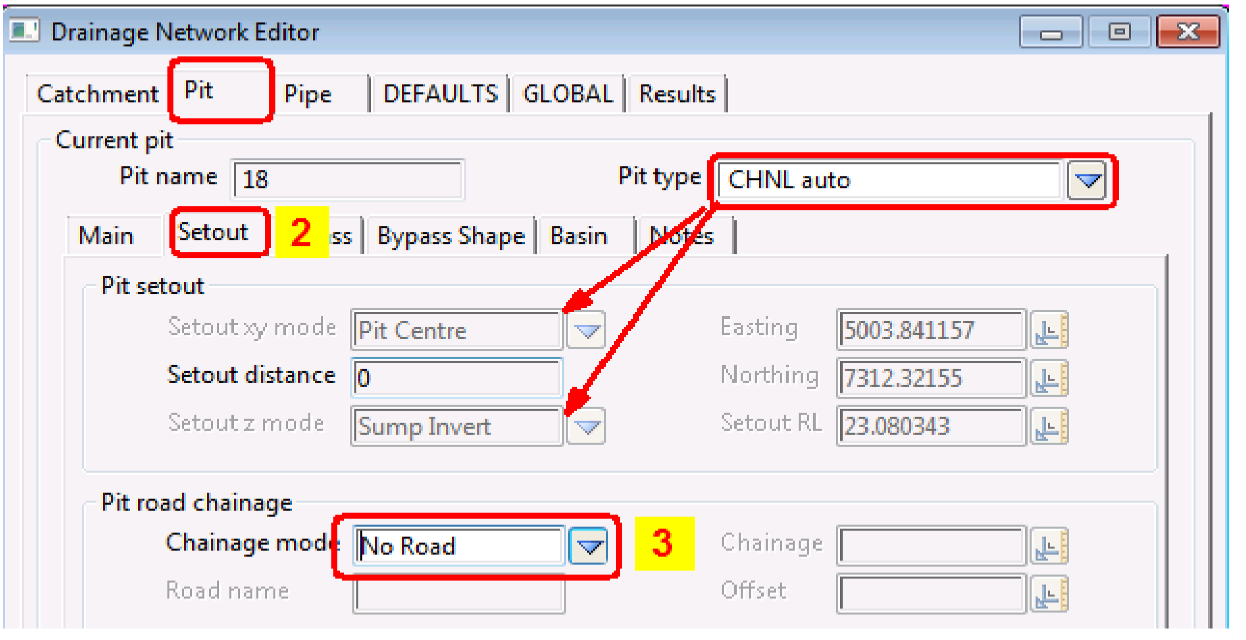


1. Selecting the **CHNL auto** pit typechanges many setting in the DNE.
   Cover RL mode is changed to **Max obvert**.
   Grate RL mode is changed to **Max obvert**.
   Pit diameter of diameter of zero. This will result in a single line on the drainage long sections instead of a pit.
   Ku method is set to **Direct** and a **Ku** of 0 for the channel change of grade points (pits).
   Inlet config - On-grade pit - it has a 200% inlet capacity. The 200% inlet capacity is in case a

choke factor is accidentally applied.

2. Select the Pit Setout tab

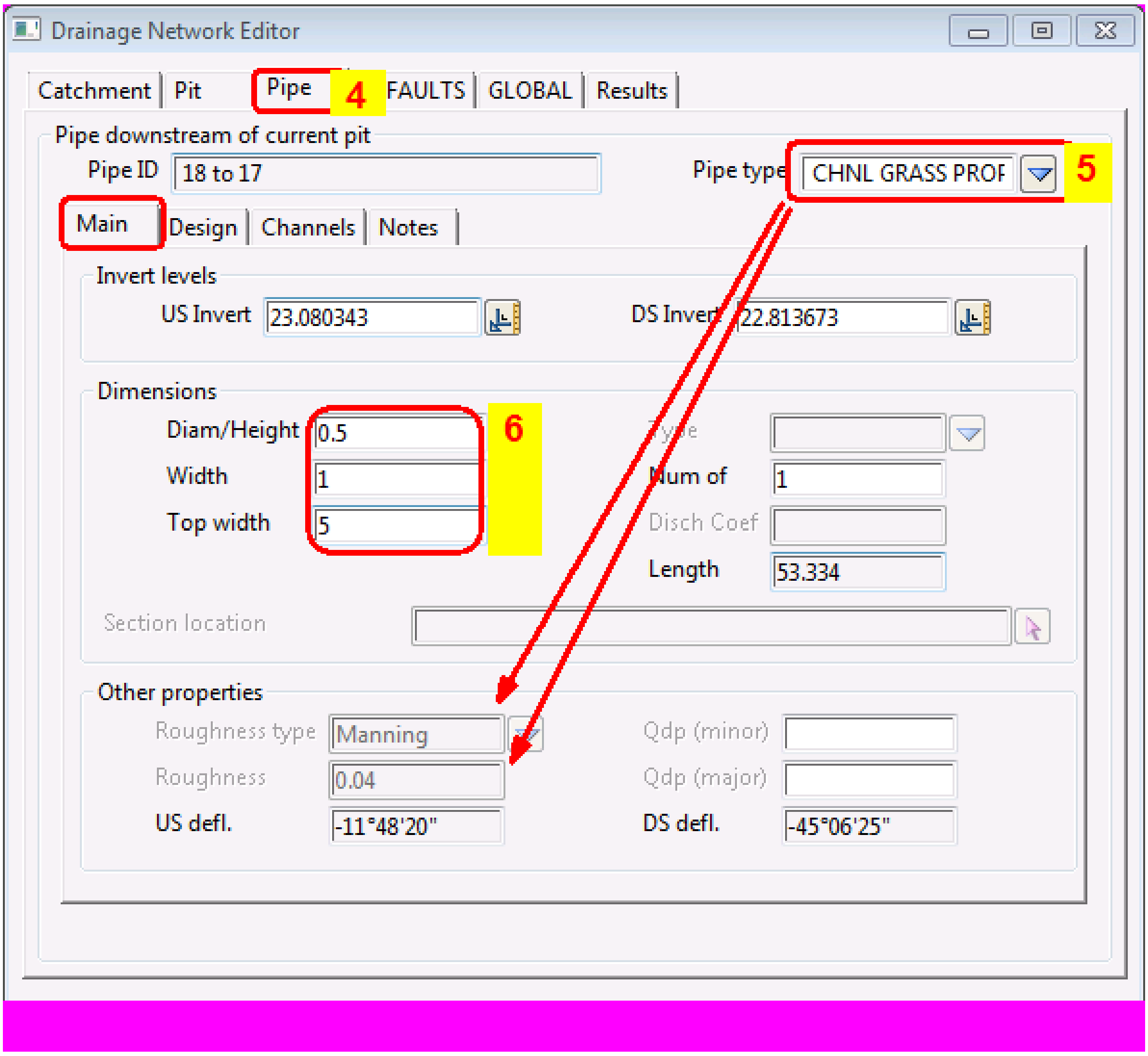
Pit-Setout Tab



Setout xy mode is set to **Pit centre**
Setout z mode is set to **Sump Invert** which is the lowest of the channel inverts if **Sump offset** = 0(Main tab).

3. The **Chainage mode** may be be changed to **No Road** if the channel centre line is not to be used for setout.If the default was **Centre string** this will stop the Problem message saying the centreline string is not found.

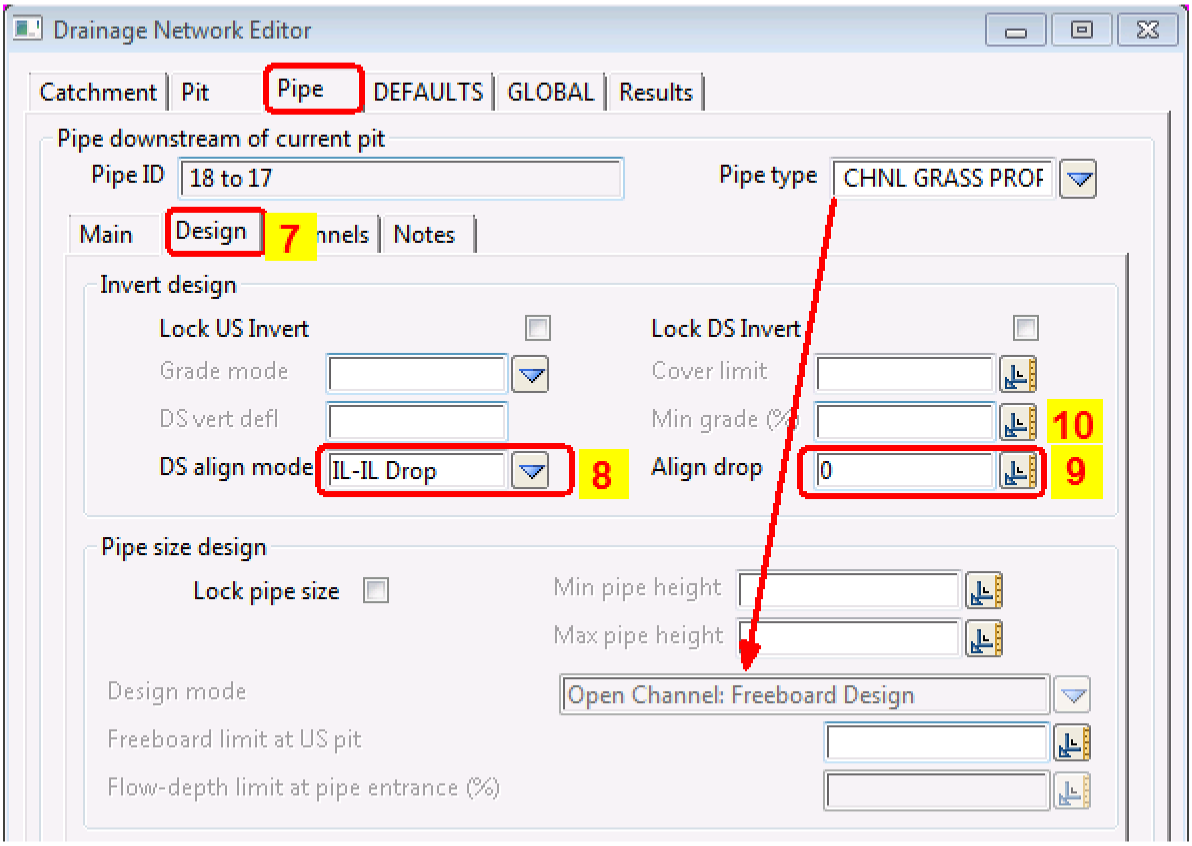
Pipe-Main Tab



4. Change to the **Pipe->Main** tab
5. Change the **Pipe type** to **CHNL GRASS PROPOSED**. This changes the following:
   the **Roughness type** to **Manning**
   the **roughness** for the open channel to 0.040
6. Set the **Height, Top width** and **Bottom width** of the channel.

Note: The pipe type can now be changed back to channel if you needed to alter any of the locked settings. Selecting the **CHNL GRASS PROPOSED** first set most of the setting correctly as a good starting point.

Pipe-Design Tab



7. Change to the **Pipe->Design** tab
8. Set the **Align mode** to “IL-IL drop”
9. Set the **Align drop** value to 0 (unless you are designing drop structures for your channel.
10. **Min Grade %** for open channels is usually much less than pipes.

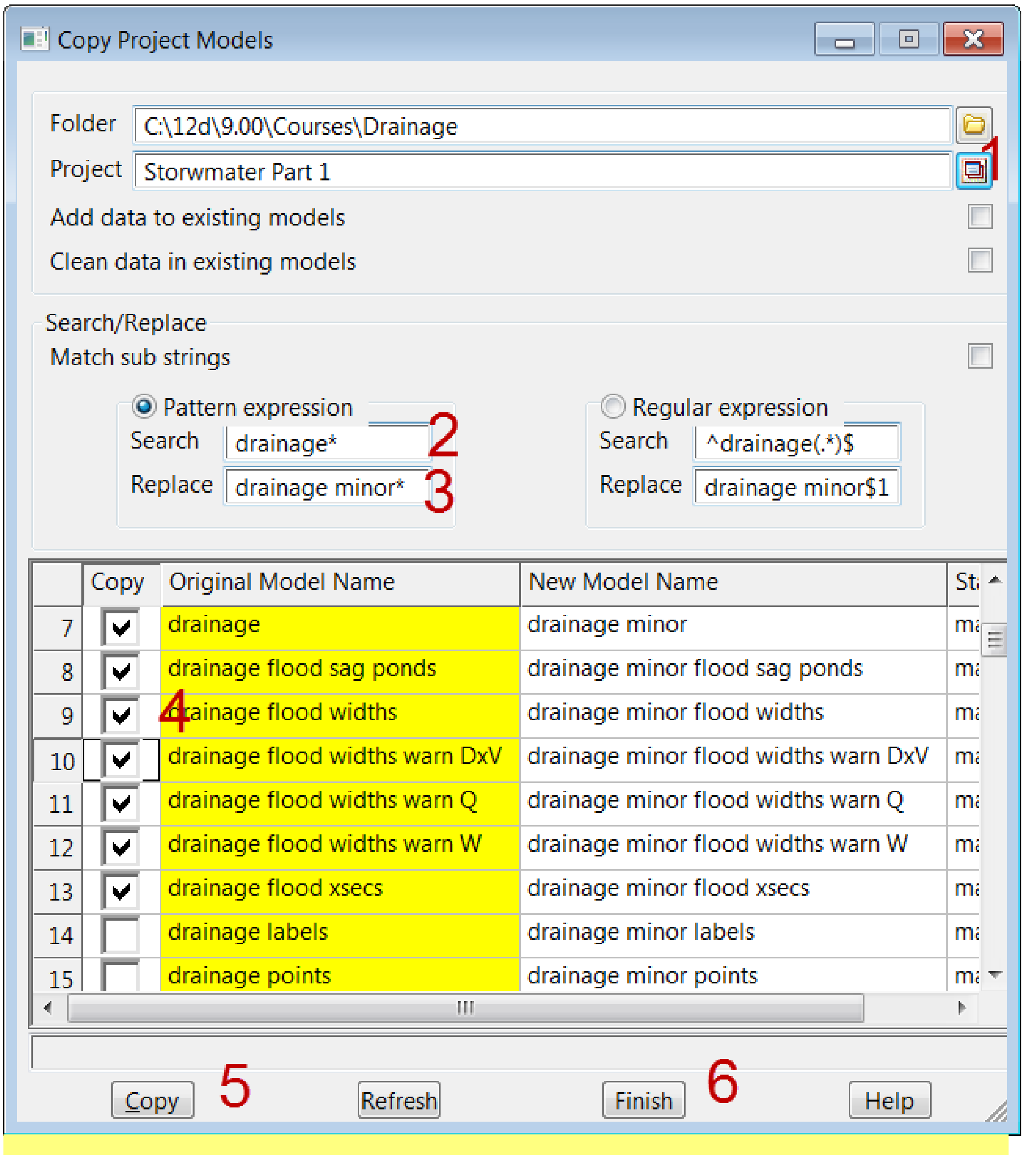
Note: The **Design mode** has been changed to “**Open Channel: Freeboard Design**”. This will allow supercritical flow to continue through the channel junctions. Often the Freeboard limit will be different for a channel than the pipe system.

9.0 Major Flood Events

To keep the results file from the minor event copy the drainage models using

Save your project. The following routine copies the models that are saved on disk!

Models->Utilities->Copy Project Models



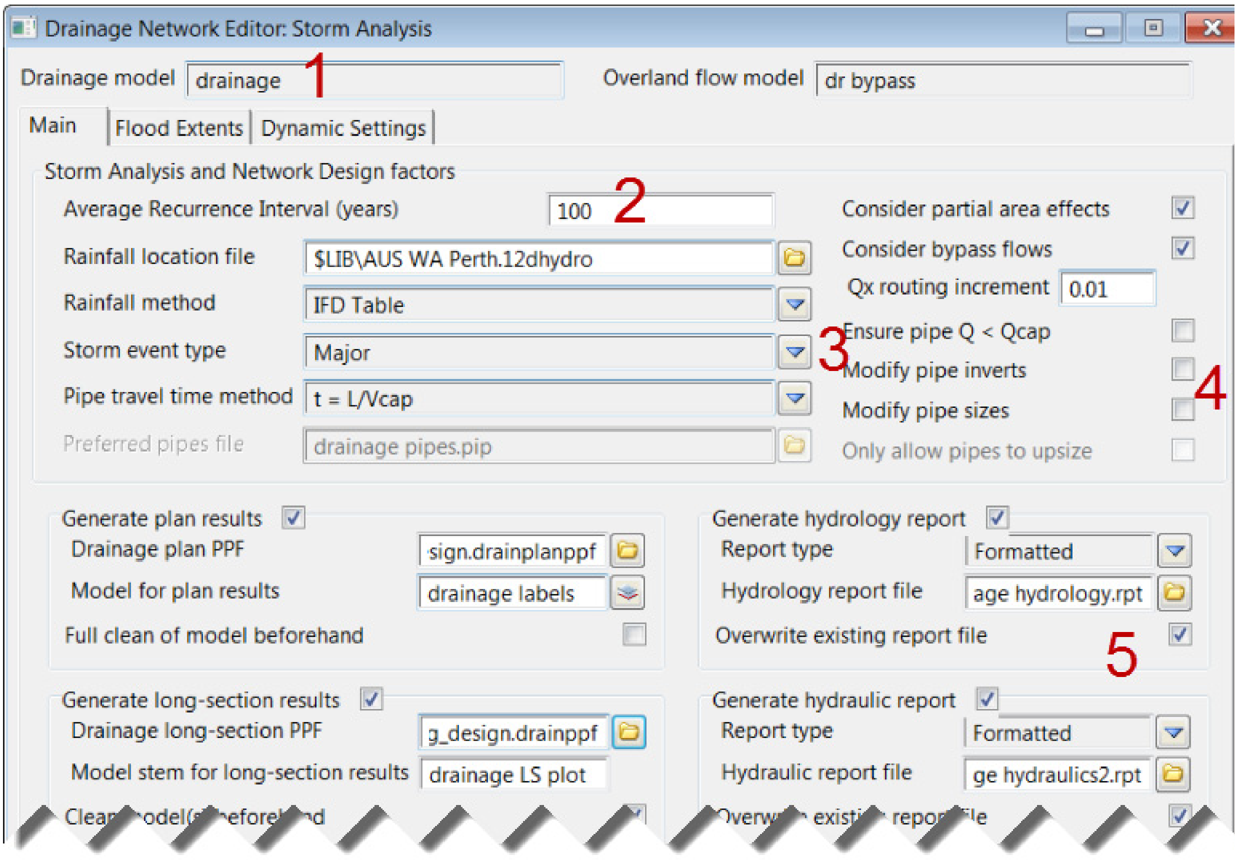
1. Select your current project
2. **Type** the name of your drainage model followed by the wildcard*
3. **Type** the new name for your minor results followed by the wildcard*
4. Tick the models that you want to copy
5. Select **Copy**
6. Select **Finish**

Now that the minor results are saftley stored away, edit the drainage model using the DNE.

For major flood events the user may desire to use an alternative set of values for

Catchment tc,
Catchment C,
Pit direct inflow (Qdi),
Pipe direct inflow (Qpi),
Pit choke factor for ongrade inlets,
Pit choke factor for SAG inlets,
Outlet tailwater levels.

These controls are found on the Storm Analysis Panel, **Main** tab.



1. Ensure you have selected the **drainage** model and then go to **Storm analysis**.
2. Type the new return period.
3. Select the **major** storm setting.
4. turn off the **Modify pipe inverts** and **Modify pipe sizes**.
5. Change the name of your report files by adding **100** after drainage.

You are now ready to analyse the major event and check the surface flooding condtions.

10.0 Excavation Quantities

Sample templates are included in the 12d library (pipe template.tpl). The templates from this template library may be added to your project using

Design=>Templates=>Utilities=>Input.

Usage

Access this report from the menu selection

Design => Drainage => Reports => Excavation Quantities

This routine uses 12d templates to calculate the excavation volume for all of the drainage strings in a model. An option to create section for a tin on top of the pipe is also available so that the drainage long sections can include hatching between the obvert of the pipe and the design tin under roads.

Templates with names set to the pipe diameters (times 1000) are used for the calculations, thus trench shapes can be customised and over excavation for bedding materials can be included. Net area calculations to exclude pipe area are not supported.

Key points

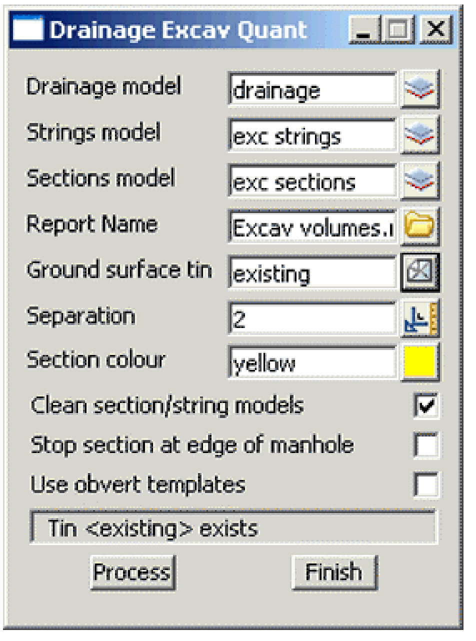
1. One template for each pipe size (mm)
2. If obvert templates are used, add the prefix “obvert “ to the pipe size
3. Carefully consider the tin selected.

A template must exist for each pipe size in the model (pipe size x 1000). For example a 0.3m pipe will require a template to exist named 300. A 0.5ft pipe would require a template named 500. A sample template library is included in the 12d library in the file **pipe_template.tpl**.

The templates are run along the strings and the total volumes are reported. Volumes for each strings are given in the report file.

If a tin is created from these strings then volumes by depth can be determined using **Design=>Volumes=>Exact=>Tin to tin**

On selecting the **Excavation quantities** option, the **Drainage Excavation Quantities** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Drainage model input box

Model to contain all of the pit and pipe network to be worked on.

Strings model model box

Strings generated from the templates will be stored in this model

Sections model model box

Sections generated from the templates will be stored in this model

Report name input box

cut and fill volumes will ne sent to this report

Ground Surface Tin tin box

tins from which the volumes will be calculated

Separation real box

distance between the sections

Sections colour colour box

Sections generated from the templates will be assigned this colour (strings colours are defined in the templates)

Clean section/strings model tick box

Delete the strings in these models before processing.

Stop section at edge of pit tick box

*Template are run from pit centre to centre if this is not selected. The templates stop at the edge of the pit if selected. This is often selected with the following option **Use obvert templates.***

Use obvert templates tick box

*Templates must be named with the prefix "obvert". i.e. **obvert 300.** The template is still run along the invert of the pipe but the user now has a section "set" of templates that can be used to create a tin on top of the pipe as well as below.*

An example report file follows.

```
---------------------------- BEGIN APPLY TEMPLATE REPORT -----------------------------

 apply template to string report -

   string         E
   tin            design
   separation         10.000
   left template  375
   right template 375
   cut volumes and areas are negative
   fill volumes and areas are positive


chainage- ------sectional  information------  ------intermediate  information----    ---------accumulative  information-------
         ---------- ---cut area --fill area ------------ ----cut vol ---fill vol    -cut volume-- -fill volume- ---balance---


    0.000                -1.434       0.000                                                0.000         0.000         0.000
                                                             -0.771       0.000
    0.550                -1.367       0.000                                               -0.771         0.000        -0.771
                                                            -14.222       0.000
   10.000                -1.642       0.000                                              -14.992         0.000       -14.992
                                                            -15.293       0.000
   20.000                -1.416       0.000                                              -30.286         0.000       -30.286
                                                             -1.845       0.000
   21.313                -1.393       0.000                                              -32.130         0.000       -32.130
                                                             -0.794       0.000
   21.863                -1.493       0.000                                              -32.924         0.000       -32.924

total cut                        -32.924
total fill                         0.000
balance                          -32.924
ie excess of cut over fill        32.924

---------------------------- END APPLY TEMPLATE REPORT -----------------------------
```

11.0 Network Quantities Report

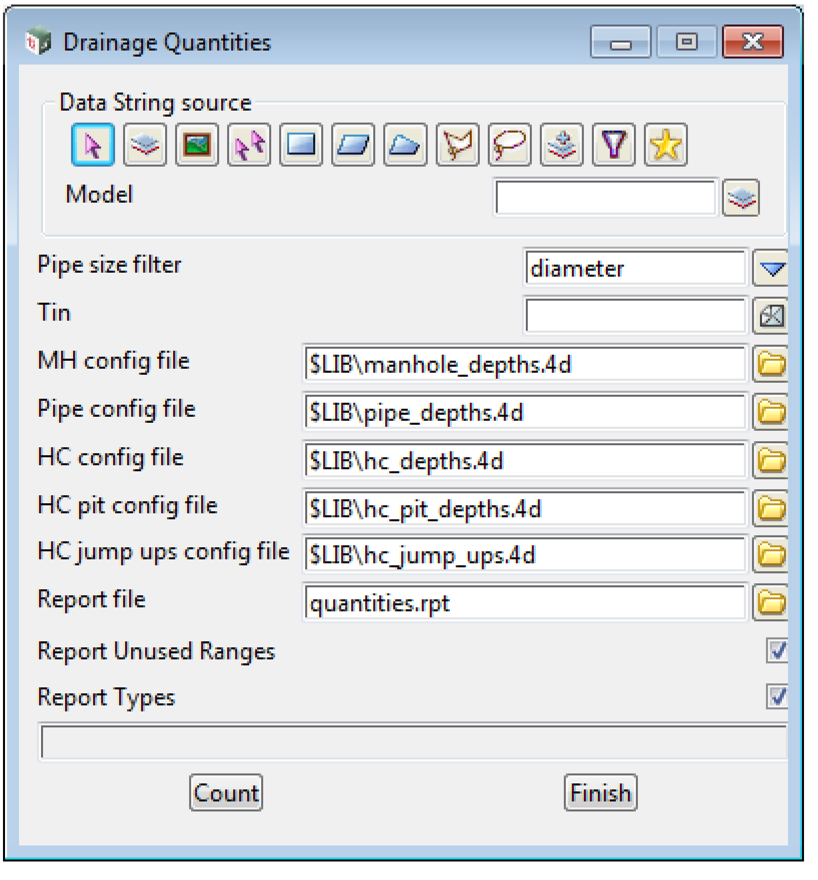
This panel is accessed from the menu selection

Design => Drainage Sewer => Reports=> Network Quantities

Key points

1. Items are counted/totalled by depth and optionally type.
2. The routine will not "double count" items even if the ranges overlap.
3. Types are case sensitive, types with spaces in the name must be enclosed in quotes and the wild card * may be used.
4. Use vertically offset tins and "banded" depth ranges to get quantities under roads, foot paths etc. This is discussed later in detail.
5. Erase count file fields if the items are not to be counted.

On selecting the **Network quantities** option, the **Drainage quantities** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data String Source	Choice		

usually the entire model is selected but view is also available for combining models

Pipe size filter	Choice		diameter or pipe size attribute

*The second item in a count line is used to filter by **diameter** (in meters/ft) or the **pipe size attribute**. The* *__diameter__ can only be used when no box culverts or trapezoidal channels are used. The __pipe size attribute__ is the label generally used in the plan plots. This is the pipe size in mm/inches (375), for box culverts width x height (750x375) and for trapezoidal channels TopwidthBottomwidthxHeight (T5000B1000x500).*

Tin tin box

This tin will be used for the pipe and pit depths.

MH config file file box

This file specifies the types and depth ranges for the pits. Details of this file are contained below.

Pipe config file file box

This file specifies the types and depth ranges for the pipes. Details of this file are contained below.

HC config file file box

This file specifies the types and depth ranges for the house connections. Details of this file are contained below.

HC pit config file file box

This file specifies the types and depth ranges for the HC pits. Details of this file are contained below.

HC jump ups file file box

This file specifies the types and depth ranges for the house connections jump ups. Details of this file are contained below.

Report file file box

a sample report file is given below.

Report unused ranges tick box

*the depth ranges for the pit/pipe/house connections are defined in the *.4d files. Selecting this option will cause the depth ranges in the file to be printed even if there are no pit/pipe/house connections in these depth ranges (zero quantity values will be shown).*

Report types tick box

*Selecting this option will cause the pit/pipe/house connection types used in the model types to be listed (even if quantities are not requested in the *.4d files). Since this is a complete of the type used in the model, the list informs the user what types have not been included in the quantity calculation.*

Count button

executes the option.

Finish button

removes the dialogue from the screen

The *.4d files listed above are contained in the 12d **library** directory. Each line is the file performs a count (count lines). No items are counted twice. Therefore, if an item is counted its type and then a count line is found the wild card is used for the type, the type already counted will not be included in the count.

The format for a count line is three or four values (space delimited) per line. Size is optional.

```
<type (from drainage.4d)>  <size> <starting depth>   <ending depth>
```

Notes:

All **types** with spaces in the name must be enclosed in quotes The wild card * may be used.

The **size** is optional and if omitted the all sizes will be counted in this group (do not use the * for a wild card).

The **starting depth** and **ending depth** are required for all count lines.

Quantities Under Roads and Footpaths

By creating super tins with vertically offset sections, quantities under roads, footpaths etc. can be determined. for example.

Offset your road design tin up by 1000m (**Tins->Utility->Translate/Copy)** and then use the depth range 1000-1999 for pipes under roads.

Create a tin from the footpaths only, null by angle length with a small length to remove the road and then offset it vertically by 2000m. the depth range 2000-2999 is not the quantities under the footpath.

Sample count lines

```
// sum concrete cover manholes is various ranges

"CONC COVER" 0.0 1.6
"CONC COVER" 1.5 3.0
"CONC COVER" 3.0 999.9  // this is expected to be zero
"CONC COVER" -999.0 0.0 // trap errors

// any that are not Concrete cover will be counted here

* 0.0 1.6
* 1.6 3.0
* 3.0 999.9
```

```
Manhole Quantities
====================================

CONC COVER           0.00     1.60     13       16.506
CONC COVER           1.60     3.00      1        1.510
CONC COVER           3.00    999.9      0        0.000
CONC COVER         -999.0      0.0      0        0.000

*                      0.00  1.60       0        0.000
*                      1.60  3.00       0        0.000
*                      3.00 999.9       0        0.000

  total length = 18.016


  Types Used
  ----------

  CONC COVER

  Diameters Used
  ----------

  1.100
```

Since the **Report unused ranges** tick box was selected, these lines were printed even though there were no pits in the data ranges.

This data results from selecting the **Report types** tick box.

Sample count lines for pipes follow.

```
// sum class 2 pipes by diameter and for various ranges

// count 375

2 0.375 0.0 2.0
2 0.375 2.0 5.0
2 0.375 5.0 999.

// count 450

2 0.450 0.0 2.0
2 0.450 2.0 5.0
2 0.450 5.0 999.

// count 525

2 0.525 0.0 2.0
2 0.525 2.0 5.0
2 0.525 5.0 999.

// count pipe sizes that were missed

2 * 0.0 2.0
2 * 2.0 5.0
2 * 5.0 999.

// count all other missed pipes

* 0.0 999.
```

12.0 Exporting to Drainage Design Software Packages

12d contains most of the data required for your drainage design packages. However, each packages has specific design variables that 12d does not have access too. The design process is intended to export your data from 12d to the design package, design the drainage system and then read the results back into 12d for your long sections.

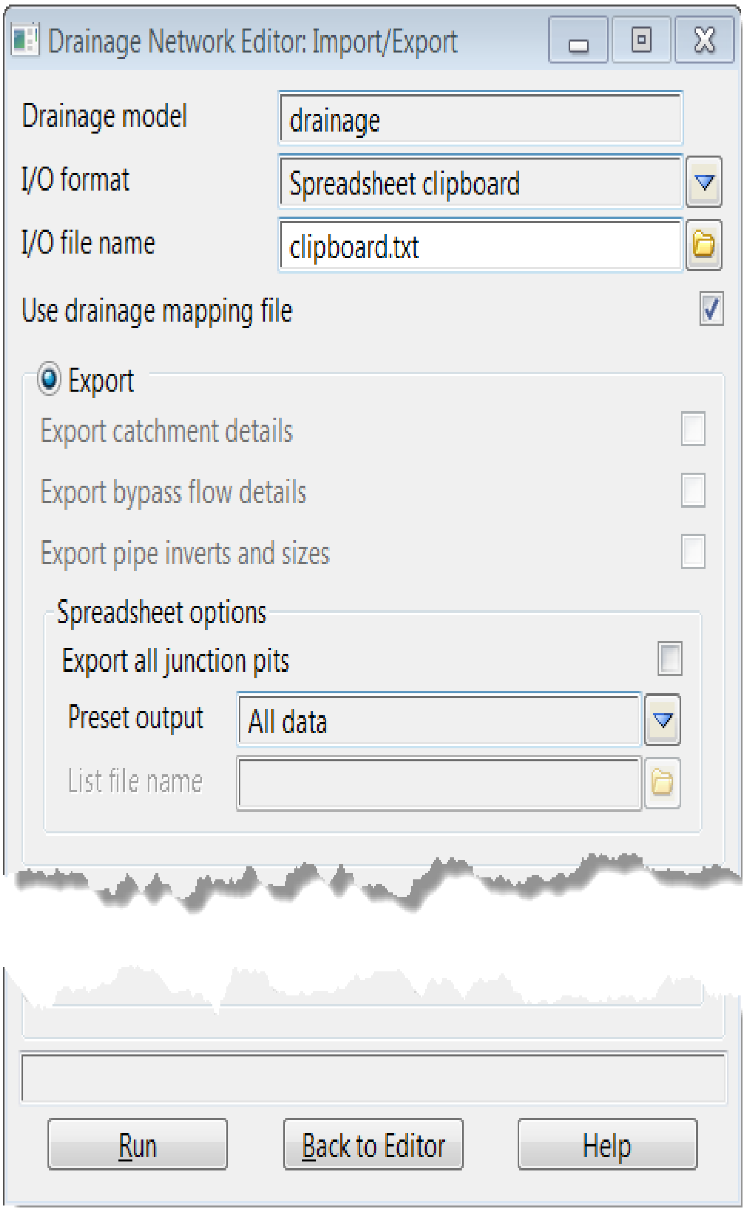
If pits/pipes are to be added/deleted from your network during the design process you are safest to add/delete the pit/pipe to 12d and to your design package separately.

Not recommended and as a poor alternative, you have the option of reading the results back into 12d, adding/deleting the pits/pipes and then exporting the data to a new drainage project in your drainage design software. **As 12d does not have access to all of the data in the design packages this method is not recommended!**

Some of the drainage design programs offer a third option that allows you to import data "on top of" an existing project thereby merging and over writing the existing data. Be sure to contact the drainage software supplier to obtain exact details of how the merging process is performed.

The interface is run by selecting **Import/Export** from the **Drainage** Network Editor

Design->Drainage-Sewer->Network Editor



The **Drainage model** is the model currently being edited.

The **I/O format** selects which external program the 12d is interfacing with. Some programs use the windows clipboard and others use files. If the clipboard is used the data will also be written to a file by 12d in case you need to take the data to another computer.

Export enables the export fields below and exports when **Run** is selected.

The **Export options** have slightly different effects depending on the **I/O format** (program) selected above. Therefore they will be discussed later with the various formats.

Export pipe diameters and inverts is generally select for existing systems only. If your design program will set invert levels and pipe sizes then turn this tick box off for new systems.Some design programs will require initial inverts and pipe sizes. In this case this box should be selected on the first export.

Export default catchment/pit parameters is generally selected for the first export. For subsequent exports turn this selection off and then only the catchment areas (if the model is supplied above) will be exported.

13.0 Drainage Data Input and Output to Spreadsheets

Spreadsheets are an effective method to manage the numerous variables urban drainage designers create in the modelling process. Spreadsheet data can be transferred to and from 12d in tab delimited files and stored within 12d as “user definable attributes”. These attributes are linked to the pit and pipes within a network. Drainage long section plots can display the pipe attributes in the “arrows” data area and pit attributes in the bubbles area. Drainage plan drawing can also show these pit and pipe attributes.

Drainage strings will be created if they do not exist in the model but pits cannot be added to existing strings.

See also

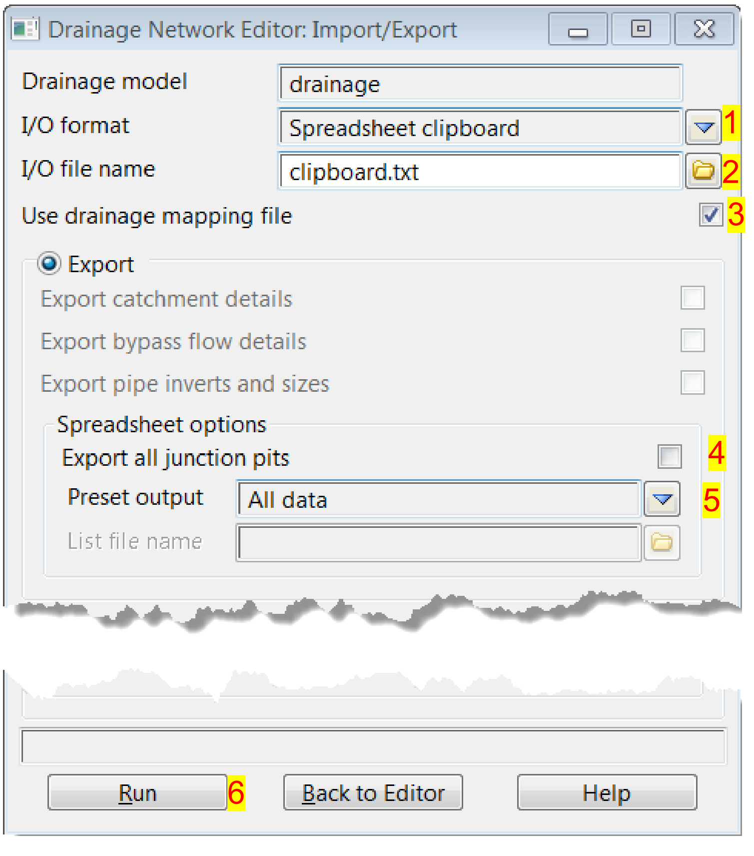
12d to spreadsheet transfers

Spreadsheet to 12d update and create

Spreadsheet options

13.1 12d to spreadsheet transfers

This interface is accessed the **Import/Export** button on the Drainage **Network Editor**.

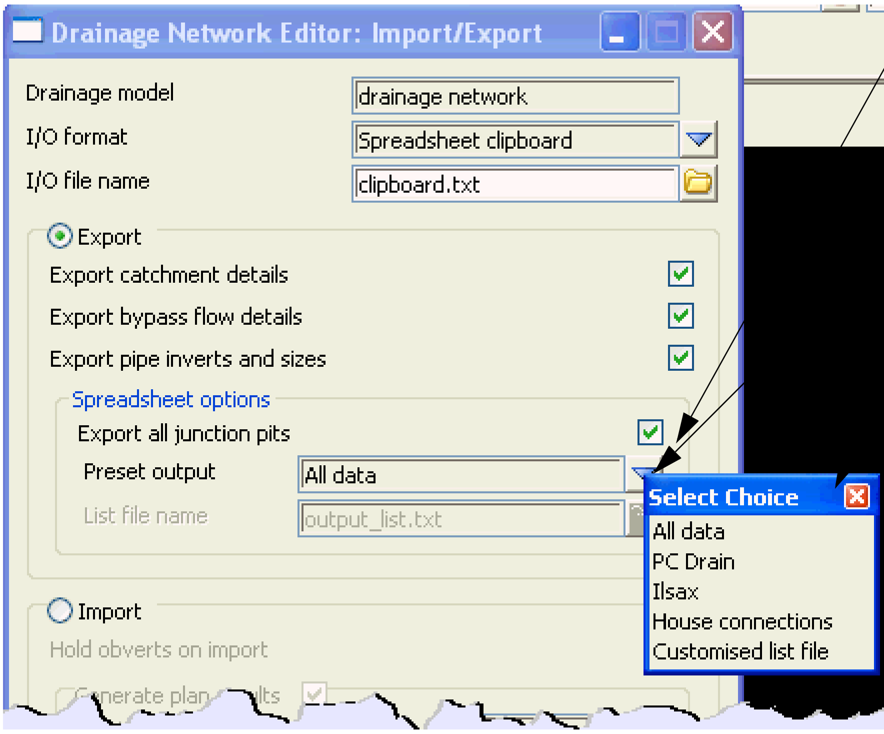


1. Select Spreadsheet clipboard
2. leave as **clipboard.txt** to send the data to the windows clipboard as well as this file.
3. Mapping files are the most current 12d technology. Leave this selected.

   These options are not used for spreadsheet export.
4. Usually leave this off! Select to export the junction pit at the end of all drainage lines (very rarely needed).
5. You may also select to limit the output if desired. If you like using spreadsheets for data entry, the PCdrain data and ILSAX data formats are useful for adding data for the first time for either program.
6. Select Run to place the data on the clipboard.

13.1.1 Options

The **Spreadsheet Options** section allows the user to define the amount of data exported.



Export all junction pits when selected will repeat the data for the junction pit at the end of each branch line.

Preset Output this choice box offers the following choices:

All Data: All of the 12d drainage string data and the user defined attributes will be exported to the clipboard in a tab delimited format. The 12d data names and the user defined attribute names will appear at the top of the spreadsheets columns.

ILSAX: For the ILSAX program, the spreadsheet column headings will change depending on the pipe and catchment indicators (P2 card) and the inlet type (P3 card). Therefore, use the ILSAX pipe editor macro to set up one pit/catchment for the type of data you wish to enter. Now when you export the pipe network data the column headings will include the names of the relevant parameters.

User defined below: The **Customised list file name** is used to define the drainage values, their order and format you desire.

The **customised list file** is a text file where each line contains a drainage variable or a spreadsheet IO command (blank lines are ignored unless preceded by the header command). The spreadsheet IO commands are all lower case and listed below:

header	to define a line of text to be exported
blank	to leave a blank column in the output
pit data	the following attributes are for the pit.
downstream pit data	the following attributes are for the downstream pit.
upstream pit data	the following attributes are for the upstream pit(s).
pipe data	the following attributes are for the pit's outlet pipe
downstream pipe data	the following attributes are for the downstream pipe(s)
upstream pipe data	the following attributes are for the upstream pipe(s)
variable name	a 12d drainage variable names
factor	the following variable is multiplied by this factor
decimals	the following variable will export with these decimal places

The simplest way to create your own customised tab delimited file is to set the **Preset Output** field to **All data** and leave the **customised list file name** field blank. Selecting **Set**, **Finish** and then **Copy** from the main dialogue. The data will be placed on the clipboard and a **customised list file**, named **output_list.txt** will be created containing the names of all of the drainage variables in the 12d model. Use a text editor to add/or delete the variable names, change their order and/or add spreadsheet IO commands. **Save the file with a new name!** The **output_list.txt** file is overwitten on every export.

A listing of a customised list file follows. Note the words in the header file have a "tab" between them so that they will be spaces across the spreadsheet columns.

```
header
Pipe Details
    header
    Name Length   U/S IL   D/S IL   Slope(%)   Class   Dia I.D.   Rough   Pipe Is   No. Pipes

    pit data
    *pit name

    pipe data
    *length
    low ch invert
    high ch invert

    factor
    100
    *grade

    pipe type

    factor
    1000
    diameter
```

After creating your customised list file, select **Options** again and change the **Preset Output** field to **User Defined below** and enter the new **customised list file** name that you saved above. Select **Set** then **Finish** and finally **Copy** to put the formatted data onto the clipboard.

The data can be pasted into a spreadsheet program for checking or additional formatting.

CUSTOM FORMATED DATA MIGHT NOT BE PASTED BACK INTO 12d!

The data must be in the "12d drainage spreadsheet" format to be read into 12d.

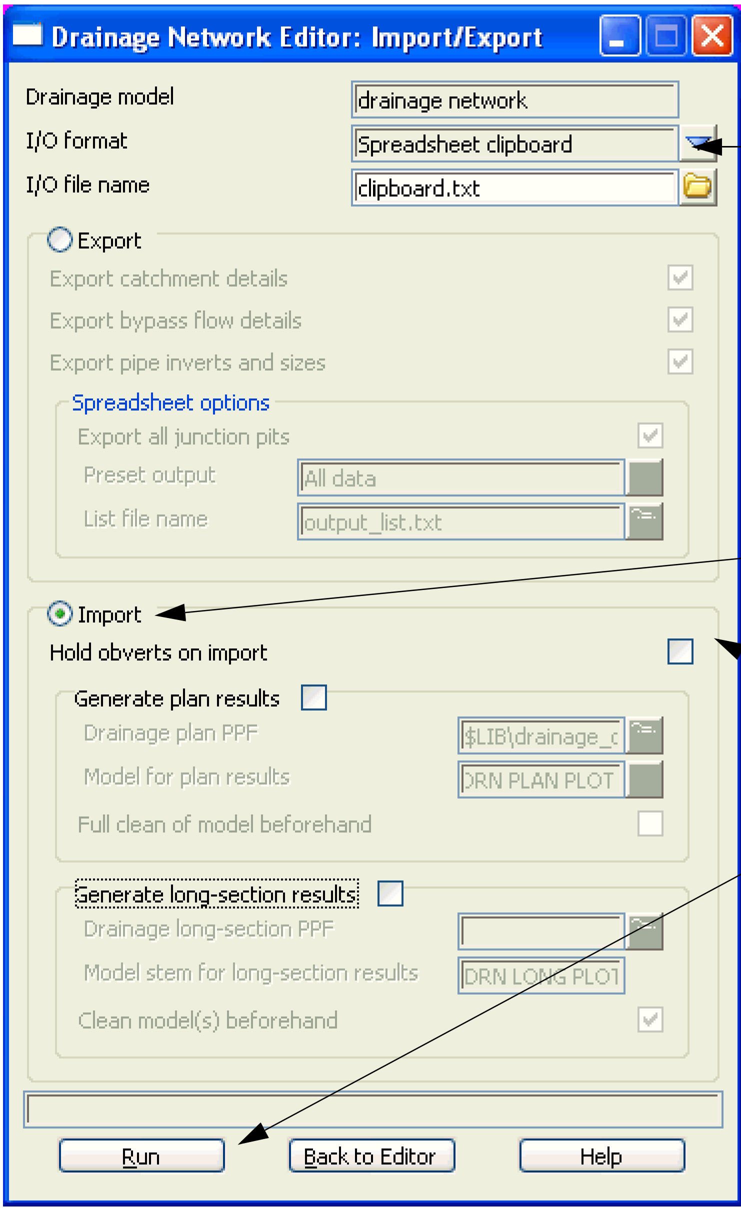
Caution with pit names in the form 1-1 or 1/1. Some spreadsheets will interpret these values as dates. If you use these formats for your pit names you will have to paste command them in once, format the columns that contain the pits names as text data and then paste the information in again.

One final word on using the copy/paste commands in the Microsoft Excel program. The Paste Special command using the "Skip Blanks" option will allow you to copy a large block of 12d data (with blanks in it) on top your data so that your data is preserved where it coincides with the blanks. To use this option paste the data into a blank spreadsheet and then select copy again. The Paste special option with "Skip Blanks" will now be available.

13.1.2 Spreadsheet to 12d transfers

This item is accessed from the **Import/Export** button on the **Drainage Network Editor**.

The following panel will appear.



Select the **Spreadsheet clipboard** format**.**

Select **Import**

This is ignored on spreadsheet imports.

Select **Run** to update the drainage model.

Tab delimited, “12d drainage spreadsheet” format or “from to” format data must be on the clip-board in order to update a 12d drainage model or create a new model. These format are described below.

13.1.3 Updating an Existing Model

The data usually is generated by 12d using the **Export** option, pasted into a spreadsheet and then copied back to the clipboard so that 12d can be updated.

When 12d exports the drainage model to a spreadsheet it includes a column for the unique string identifier and a unique pit identifier (unique to the drainage model not the 12d project). The names of the strings and pits may be changed via the spreadsheet if these columns are present at import time.

If the pit id column is not present, 12d will search the drainage model for a matching pit name. When the pit is a junction between drainage lines, only the trunk line will be the data updated.

13.1.4 Creating a New Model

It is possible to create a new string or an entire drainage network using this format. However, pits cannot be added to an existing string. The entire drainage string must be created at once. Two formats are available, the “from-to pit” format and the “12d drainage spreadsheet” format.

At present the network editor must select a drainage string to become active. Therefore, if you are not adding strings to a network, you will have to great a drainage network with one “dummy” pit. Select this one “dummy” pit to activate the editor. After importing the data and the new drainage lines are created the “dummy” pit may be deleted.

12d drainage spreadsheet Format

The top left cell in the clipboard data must be the text “12d” to specify this format. The minimum amount of data required to create a new string is the string name, pit name, x and y coordinates. You can add as much additional data as you have available. This would include pipe diameters inverts etc. The pits must be listed from upstream to downstream order. If the string is to join a trunk line, the junction pit must be included for both the tributary and the trunk line.

An example file exists called **new_network.txt** is supplied in the library. Open this file in a spreadsheet or a text editor and copy it to the clipboard. Set the **I/O Action** to **Import** and select **Run**. The new drainage lines will exist in the model currently being edited.

From-to Pit Format

The top left cell in the clipboard data must be the text “from to” to specify this format. The minimum amount of data required to create a new string is the upstream pit name "*pit name), the downstream pit name (*ds pit name) and the x(x location) and y(y location) coordinates of the upstream pit. If the string is to join a trunk line, the junction pit must be included for both the tributary and the trunk line.

An optional column for the pit cover elev (cover elev) may be specified. Once the network has been created additional pipe and pit data may be added using the “12d drainage spreadsheet” format described above.

An example file exists called **new_from_to_network.txt** is supplied in the library. It is shown below. Open this file in a spreadsheet or a text editor and copy it to the clipboard. Enter a new model name in the **Drainage model** field and select paste. The new drainage model will now exist.

from to	pit	pit	pit	pit
*pit name	*ds pit name	x location	y location	cover elev
text	text	real	real	real
E/1	A/3	5309.458	7336.935993	29.2173
D/1	A/4	5277.189	7336.935989	28.5071
C/1	B/3	5251.238738	7423.99485	31.5257
A/1	A/2	5354.629222	7336.935998	30.2115
A/2	A/3	5340.019987	7322.035996	29.89
A/3	A/4	5293.458002	7322.035991	28.8652
A/4	A/5	5250.182625	7322.035986	27.9127
A/5	A/6	5217.194202	7322.035983	27.1867
A/6	A/7	5183.458002	7322.035979	26.4442
A/7		5152.698693	7322.035975	25.7672
B/1	B/2	5289.42875	7422.289079	32.7197
B/2	B/3	5264.638564	7393.947083	30.7948
B/3	B/4	5249.738564	7384.207593	30.4187
B/4	B/5	5249.738564	7351.201545	29.1444
B/5	A/5	5233.426685	7336.935984	27.544

13.2 "12d drainage spreadsheet" Format

Each column of data is used for a 12d drainage variable or a user defined attribute. Each row represents a pit and the downstream pipe (controlled by the direction of flow variable) within the drainage network. A sample is shown below.

12d	pit	pit	pit	pit	pit	pit
*string Name	*pit name	pit type	pit low ch invert	pit high ch invert	pit id	string id
text	text	text	real	real	integer	integer
E	E/1	SA2	28.108	28.108	1	67389
E	A/3	SA2	27.7559	27.7559	2	67389
D	D/1	SA2	27.3961	27.3961	3	68100
D	A/4	SA2	26.8018	26.8018	4	68100
C	C/1	SA2	30.67	30.67	5	72072
C	B/3	SA2	29.563	29.563	6	72072
A	A/1	SA2	29.1026	29.1026	7	82469
A	A/2	SA2	28.7811	28.7311	8	82469
A	A/3	SA2	27.7652	27.7059	9	82469
A	A/4	SA2	26.8127	26.7518	10	82469
A	A/5	SA2	26.0867	26.0244	11	82469
A	A/6	SA2	25.3442	25.2942	12	82469
A	A/7	SA2	24.6672	24.6672	13	82469
B	B/1	SA2	31.2759	31.2759	14	192066
B	B/2	SA2	29.351	29.301	15	192066
B	B/3	SA2	29.123	29.073	16	192066
B	B/4	SA2	28.0444	27.8951	17	192066
B	B/5	SA2	26.3447	26.2947	18	192066
B	A/5	SA2	26.0744	26.0744	19	192066

Duplicate Definitions

Strings Variables such as “direction” are may be defined for numerous pits on the same string. Searching in a top down direction through the file, the last definition found for the string will be set.

Invert levels may be set via pipe data or pit data or combined. It is recommended that the user only use one method and not combine them. Both are exported so delete the ones you are not going to use. The variables are processed from left to right, so if duplicate definitions of an invert level or found the right most data will be set.

The format definition

1. Row1, column 1 must contain either “12d”, or “from to”. Therefore, the first column must be a 12d drainage variable (cannot be a user defined attribute).
2. Row 1. The text <pit> at the top of the column indicates the column contains a user defined pit attribute and similarly <pipe> indicates a user defined pipe attribute.
3. Row 2. This row contains the names of the 12d drainage variable names and the pit/pipe attributes. All names are case sensitive so be careful where you use capital letters. A list of 12d drainage variables is found below.

   Names beginning with an asterix (*) will not be processed (except pit/string names when unique identifiers are present in the data). 12d drainage variables names beginning with an asterix indicate that this data was calculated at export time and cannot be read back into 12d (for example, pipe length, pipe grade and deflection angle).

   Prefixing an user defined attribute name with “DELETE ” (no quotes, note the space after the DELETE) will cause the attribute to be deleted from all pits/pipes within the model.
4. Row 3. The text in this row define the type of attribute to be stored within 12d. The only valid choices are;

   integer
   real
   text

   If you want to change an attribute type you must delete the attribute and create it again. If you simply change the attribute type in the third row then that attribute will not be updated.
5. Blank lines may be inserted as desired.
6. You are not required to fill in all of the cells in the spreadsheets. Blank cells are ignored (you must use a space to remove all data from text attributes (the space will not be stored).
7. Pipe names are included in the data so that they can be changed but they are **not** used to identify the pipe. Pipe data will always be assigned to the pipe following the pit in the direction of **ascending** chainage. If flow directions is ascending then the pipe data will be for the downstream pipe. If the flow direction is descending then the pipe data will apply to the upstream pipe.

13.3 12d Drainage Variable Names

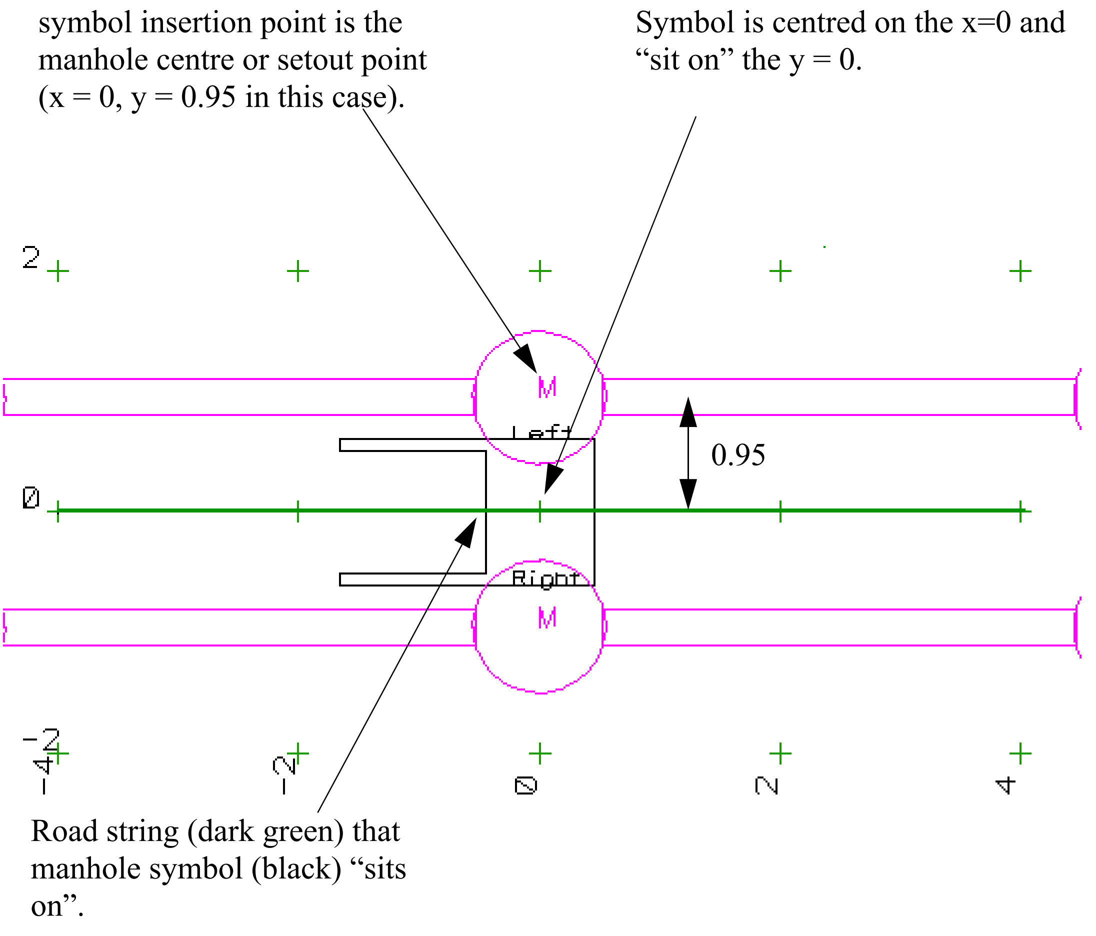
Manhole Variables	Pipe Variables	String Variables
*string Name	pipe name	direction
*pit name	pipe type	[illegible]
pit type	low ch invert	[illegible]
pit diameter	high ch invert	string id
pit low ch invert	diameter	
pit high ch invert	*length	
pit road chainage	*grade	
pit road name	low hgl	
*pit angle	high hgl	
*pit drop	pit hgl	
*pit depth	flow	
*pit chainage	velocity	
x location		
y location		
cover elev		
*fs elev		
*ns elev		
pit id		

14.0 Detailed Drainage Plan Drawing - Creating MH Symbols

There are many methods to create the drainage symbols and one is presented here. The pit symbol is drawn so that the symbol “sits on” the road string that it aligns with (the road string is the y reference). For drawing lintels and grates, assume the road is downhill to the right. See diagram below.

Two symbols are required, one for the left side of the road and one for the right side (referred to as the mirror symbol in the plan ppf). 12d checks the downhill direction of the road strings when using these symbols to determine which is the left and right side of the road.

Either draw your own symbol or import the symbols using **File IO->Data Input->4ds/12da data** and selecting the file **inlet symbols.12da**

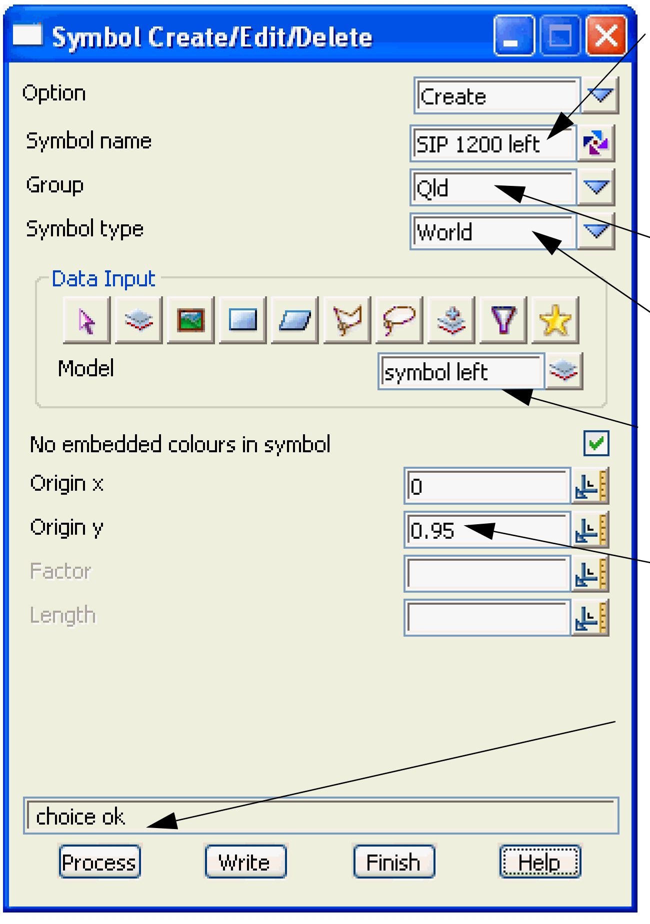


To create the symbol in 12d draw the left pit symbol in a model by itself. The pit should “sit on” the zero “y” grid line. A 900 wide by 600 long inlet with a 1.2m lintel is shown above.

From the main menu, select

Project->Tree

Expand **Project** then expand **Symbols** and finally select **Create Symbol**.



Enter the **Symbol Name**. Usually the name includes the size.

Select the **Group** the symbol is to appear in the symbol drop down menu and select **World** as the symbol type.

Select the **model** that contains the manhole drawing.

Enter the x,y location of the manhole centre in your drawing (0 is different then blank!).

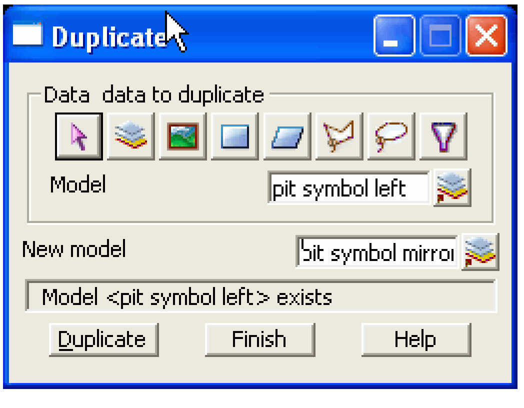
Process will create the symbol in the project.

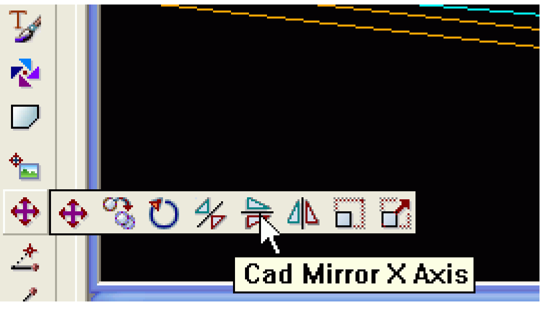
Change the **symbol name**, **origin y** and press **Process** for each offset required.

Write will add the new symbols to the symbols.4d file so that they will be available to other projects.

Leave this panel open as we will need it after we create the mirror image of the pit.

To create the mirror image of the symbol about the x axis use the CAD mirror about X axis command. 12d will require you do this one string at a time. Version 7 does not mirror into a new model so you will want to duplicate the pit symbol model first using **Utilities->A-G->Duplicate.**





Now repeat the process of creating the right side mirrored symbol. Note that the location of the pit centre is now **negative** and add the suffix **mirrored** to the symbol name.

Do not forget to select **Write** again to save the symbols to the symbols.4d file.

These symbols can now be used in the **Maintenance hole** tab of the Drainage Plan Plot ppf editor. If your version of 12d supports, **native-size symbol**, in the ppf editor then use this mode. If not, select scalable symbol and enter the size equal to the full width of the symbol (height if it is greater).

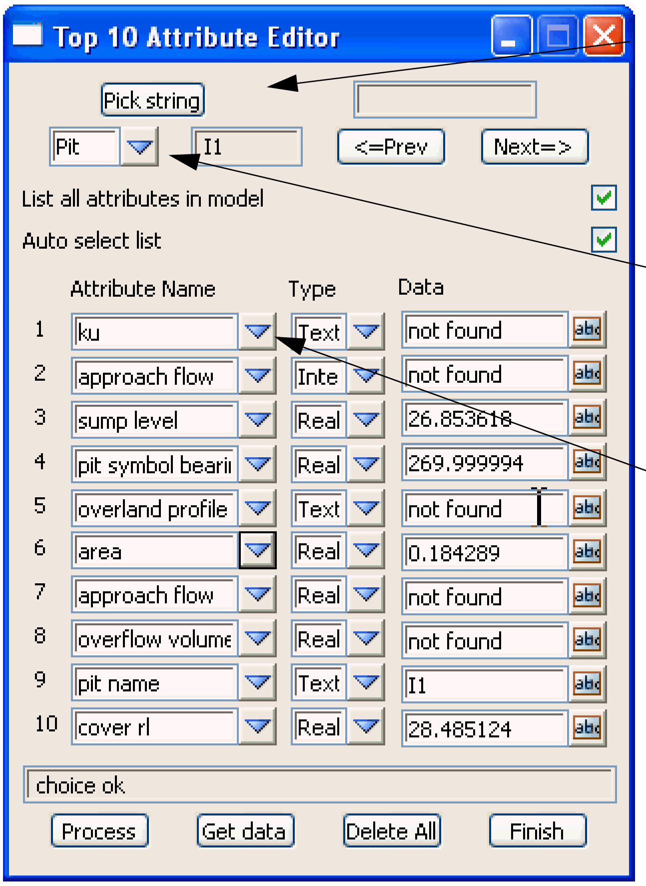
15.0 User Defined Attributes

These drainage attributes are automatically created by 12d when required but you are free to change them or add more as desired. The attributes may be exported to a spreadsheet and edited and then imported back into 12d. To edit/add the user defined attributes within 12d select either

Strings=>Properties=>Attributes or

Strings=>User=>Attribute Editor. This second editor is described below.

.



First Select **Pick** to select the string that contains the user attributes (the drainage string). The strings will be highlighted in white when they are selected.

All catchment data is store with the pits in drainage strings. To access the pit attributes, select the drop down icon and then select **pit**. A circle will be drawn around the pit selected. **Next** and **Prev** will now move you from pit to pit.

Select the drop down icon and then select the **Attribute Name** from the list of existing user defined attributes. These attributes include all of the attributes in the model that the string exists in.

They may not be defined for the pit you are editing. **Not found** will be displayed in the **Data** field if the pit does not have that attribute defined.

To change the value for the attribute enter the new value in the **data** field. If the attribute does not exist, deleting the **not found** text and adding data will create it. The message on the right will be displayed whenever you are creating a new attribute.

15.1 Drainage Pit attributes2010

Pit attributes are created and/or updated when the user selects **Set Pit Details.**

pit length	real	0	mhsize (first value) from drainage.4d file
pit width	real	0	mhsize (second value) from drainage.4d file
pit group	text		mhgroup from drainage.4d file
cover rl	real	446.685248	
grate level	real	446.685248	
setout z	real	446.685248	
setout x	real	299643.648	
setout y	real	6563620.716	
setout distance	real	0	
pit name	text	1-3	
pit type	text	SA2	
pit diameter	real	1.1	
pit chainage	real	118.61441375	
pit centre x	real	299643.648	
pit centre y	real	6563620.716	
pit centre fs level	real	446.685248	
pit centre ns level	real	446.685248	
ds invert	real	445.307	upstream invert level of exit pipe
ds pit	text	1-2	ds pit along the pipe network
sump level	real	445.307	
pit depth	real	1.378	

Extra Atttributes from Pit - Main Tab

cover rl mode	integer	1
design freeboard	real	0.4
direct flow	real	0.02
grate rl mode	integer	1
inlet type	integer	0
ku	real	0.2
ku config	integer	1
ku method	integer	1
kw	real	0.22
sump offset	real	-0.2

Extra Attributes from Pit-Setout Tab

road chainage	real	10
road chainage mode	integer	2
road name	text	My Road
road offset	real	12
setout adjustment	real	-1
setout adjustment z	real	0.045

setout xy mode	integer	0	
setout z mode	integer	8	

Extra Attributes from Pit-Bypass Tab

bypass pit	text	12.4P	
choke major	real	0.8	
choke minor	real	0.7	
choke pog major	real	0.8	
choke pog minor	real	0.7	
inlet capacity curve	text	Sutherland - 3% crossfall	
manual pit grade	integer	1	
manual pit xfall	integer	1	
pit grade	real	1	
pit xfall	real	3	

Extra Attributes from Pit-Notes Tab

reamrks	text	constructed by others	

Setout string selected

design model id	uid	52	
design string id	uid	61	
pit symbol angle	real	81.48609728	
pit symbol bearing	real	8.51390272	
pit symbol bearing dms	text	8°30'50"	
pit grade	real	4.00	if bypass pit present

Centre line string selected (with bypass and setout)

pit xfall	real	3.00	if bypass pit present
centre model id	uid	52	
centre string id	uid	92	
mirror pit	integer	1	mirror symbol required for plotting

Bypass Pit entered

bypass pit	text	1-2	next pit along bypass string
inlet type	integer	1	
choke major	real	0.8	ongrade or sag choke (see sag setting)
choke minor	real	1	ongrade or sag choke (see sag setting)
inlet capacity curve	text	SA2 3% Grade	
			calculate if pit grade and/or pit xfall present (pit type - inlet cap curves determines which are required)
pit grade	real	4.00	if setout string selected
pit xfall	real	3.00	if centre line string selected

The following require the bypass string (manual entry of bypass pit is not enough).

bypass distance	real	33.995	distance to bypass pit
gutter length	real	99.79	dist. up the bypass string to next pit or end of string (longest bypass route if multiple)
gutter grade	real	2.70	(us pit fs levels - ds pit fs levels) / gutter length)

Catchment Tab data

area real 0.1
percent imperviousreal 60

Pervious Area only
c major pervious real 0.9
c minor pervious real 0.8

For both pervious and impervious (change pervious to impervious)
catchment grade perviousreal1
catchment length perviousreal900
catchment roughness perviousreal0.1
tc major perviousreal 5
tc method pervioustext Kinematic Wave
tc method pervioustext Direct
tc minor perviousreal 5

Export Pit Atttributes (calculated when Export selected on Import/Export button))

inlet type	integer	updated to include 5 for headwalls
ds pit string id	uid	drainage string id for the ds pit
ds pit index	integer	index number of the ds pit (counter along the string)
area impervious	real	%impervious * area for set 1
area pervious	real	%pervious * area for set 1
area impervious2	real	%impervious * area for set 2
area pervious2	real	%pervious * area for set 2
area impervious3	real	%impervious * area for set 3
area pervious3	real	%pervious * area for set 3
pcdrain pit type	type	pit type with the pcdrain suffix (S + pond depth)

Export Pipe Atttributes (calculated when Export selected on Import/Export button))

locked diameter	real	pipe diameter (exists only if pipe locked)
locked invert us	real	pipe us invert (exists only if us invert locked)
locked invert ds	real	pipe ds invert (exists only if ds invert locked)
windes diameter	real	if pipe type is WINDES, value is neg diameter
roughness n	real	if “roughness type” is Manning then roughness value otherwise it is 0

roughness k	real	if "roughness type" is Colebrook then roughness value otherwise it is 0

15.2 Drainage Pipe attributes

Set Pit Details

invert us	real	28.47
invert ds	real	28.422
diameter	real	0.225
pipe size	text	225
pipe type	text	2
roughness text	text	n=0.010
calculated pipe length	real	9.58175349
calculated pipe grade	real	0.50095215
calculated pipe grade 1 in	real	199.6198644
calculated us deflection	real	71.75414547
calculated ds deflection	real	-36.3032794
pipe name	text	12.5P to 12.4P
minimum cover	real	1.04113728
calculated drop	real	0.03

Additional Pipe Attributes created via Pipe Tab setting

design alignment mode	integer	0
design cover	real	0.4
design cover mode	integer	0
design drop	real	0.03
design grade	real	0.1
design size mode	integer	0
diameter	real	0.225
direct pipe flow	real	0.01
lock ds il	integer	1
lock size	integer	1
lock us il	integer	1
max height	real	0.6
min height	real	0.3
remarks	text	extra pipe notes
roughness	real	0.012
roughness type	text	Manning
width	real	0.225
width top	real	6

15.3

16.0 Detailed Drainage Plan Drawing

12d drainage has the capability to create detailed drainage plan drawings with the following features:

screate symbols at the inlets and the type of symbol is controlled by the inlet type (given when you create the inlet).

screate text labels for inlet types and user defined pit attributes

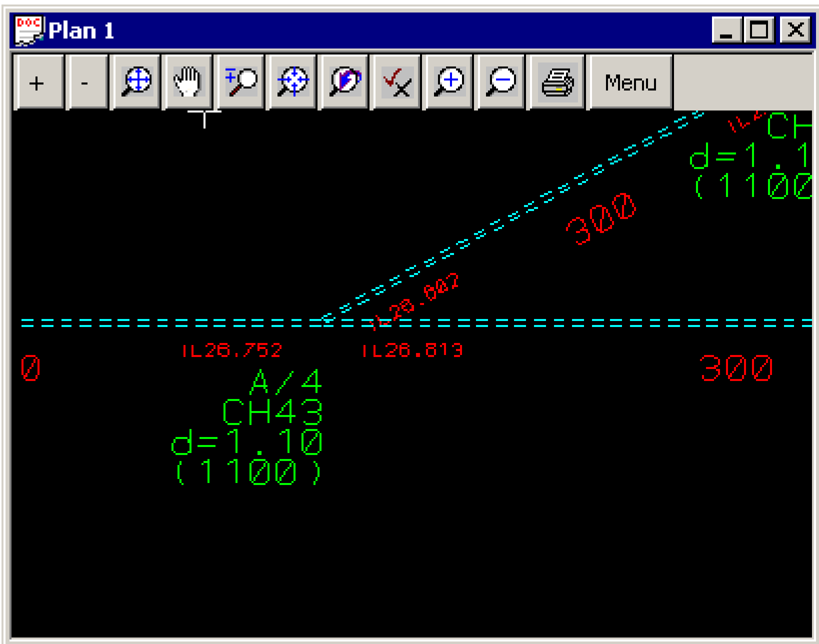
sdraw lines with line styles and colours to represent pipe sizes

screate text labels for pipe diameters, inverts and user defined pipe attributes

screate text labels for house connection types, invert levels and chainages

sindicate direction of flow on pipes.

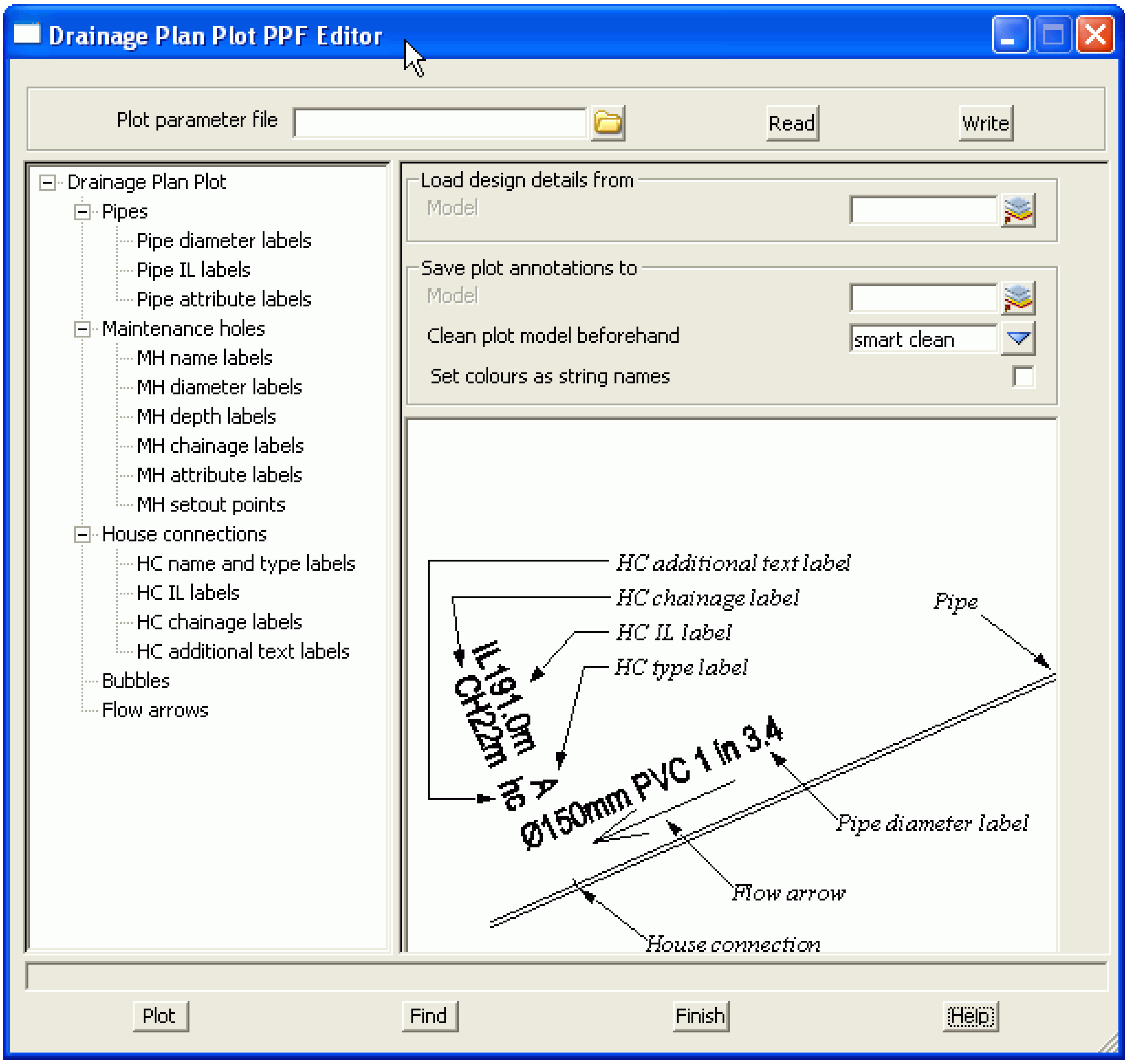
An example is shown below



Usage

This option is accessed from the menu selection

Design => Drainage-Sewer => Plots => Plan Annotations



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Plot parameter file	file box		

Optional - no dpf is required. The default settings will create a schematic drainage drawing. A custom dpf may be selected if desired.

Load design details from model box

data source for drainage strings to be labelled

Save plot annotations to modelmodel box

labels to be created are stored here, Undo will remove the labels created

Clean plot model before hand choice box *smart clean will update text that has been moved and clean the rest. Full clean will all text from the plan annotations model.*

Set colour as string name tick box

when selected the string colour will be used for the string name (to be used for DWG/DXF export using map files)

Plot button

Creates the labels in the model specified

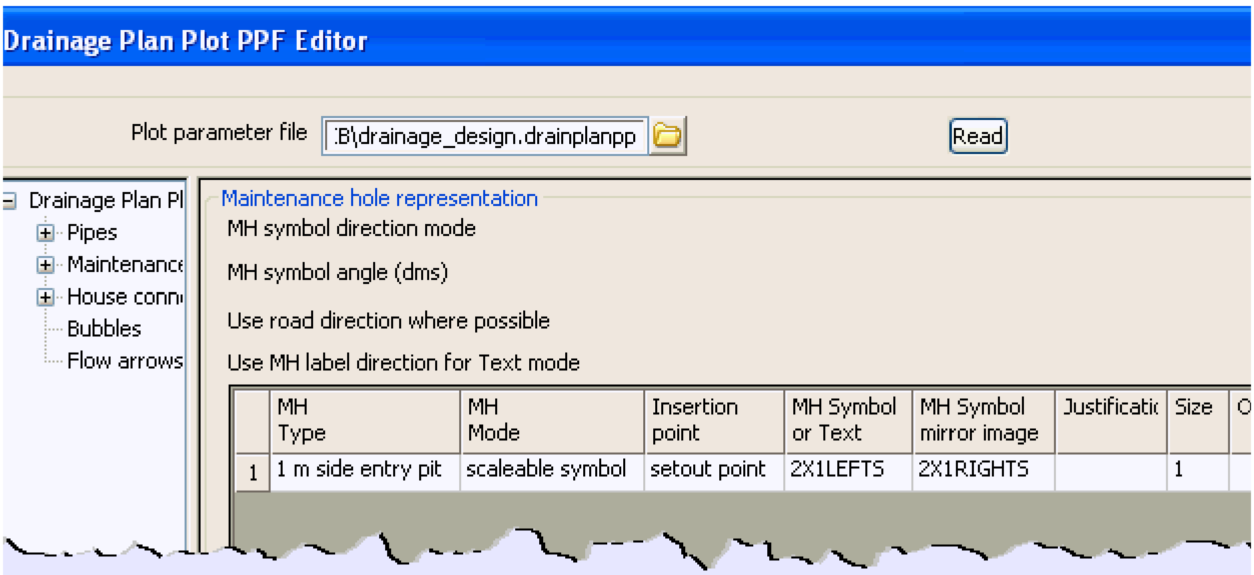
Find button

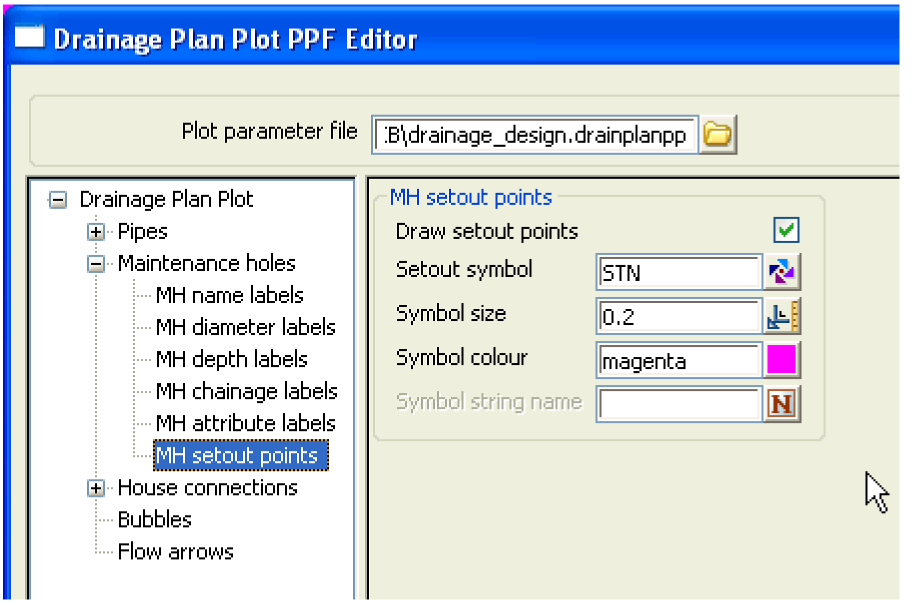
used to locate plot parameter input boxes using Version 5 plot parameter names

IMPORTANT! to turn off any data change the text height to zero.

Select **Maintenance holes** from the tree to set the symbols to be used for the various pit types.

Select the MH type as desired and use **scalable symbol** and **setout point**. For the symbols enter the names that you used when you created the symbols.





Plotting a symbol at the pit setout point is a good confirmation of the data printed in the setout reports. The settings to create this symbol are shown on the left.

17.0 Drainage Long Section Plotting - Hatching Under Roads

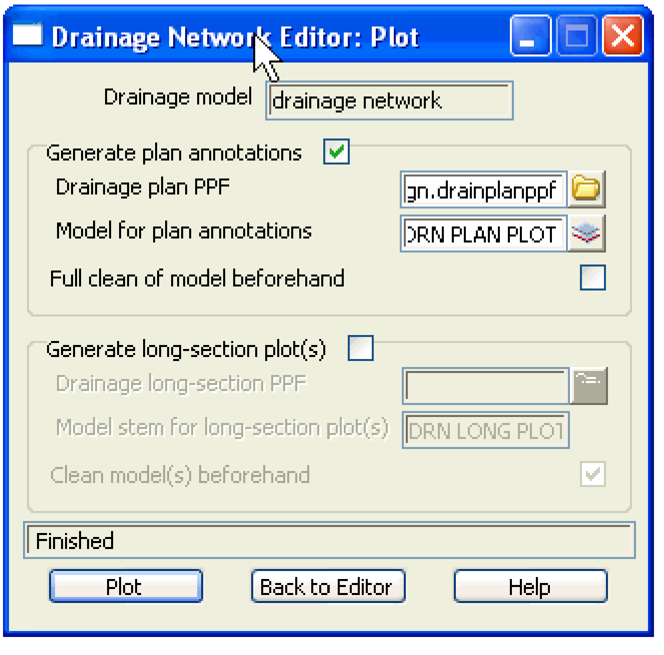
The drainage long section plotting has been discussed in the Intro Drainage Course. Here we will discuss the technique for hatching under roads and/or footpaths.

The following steps are required.

1. Run the Excavation volumes routine using the obvert templates to create sections and strings for a tin on top of the pipe (obvert tin). Select **Stop section at end of manhole** to prevent the pits from being hatched
2. Create the obvert tin from the strings and sections then and null by angle length with a small length value so that the tin is nulled near the pits.
3. Create a design tin that extends to the limit of the roads.
4. Use the hatching section of the drainage long section plot to select the hatching style.

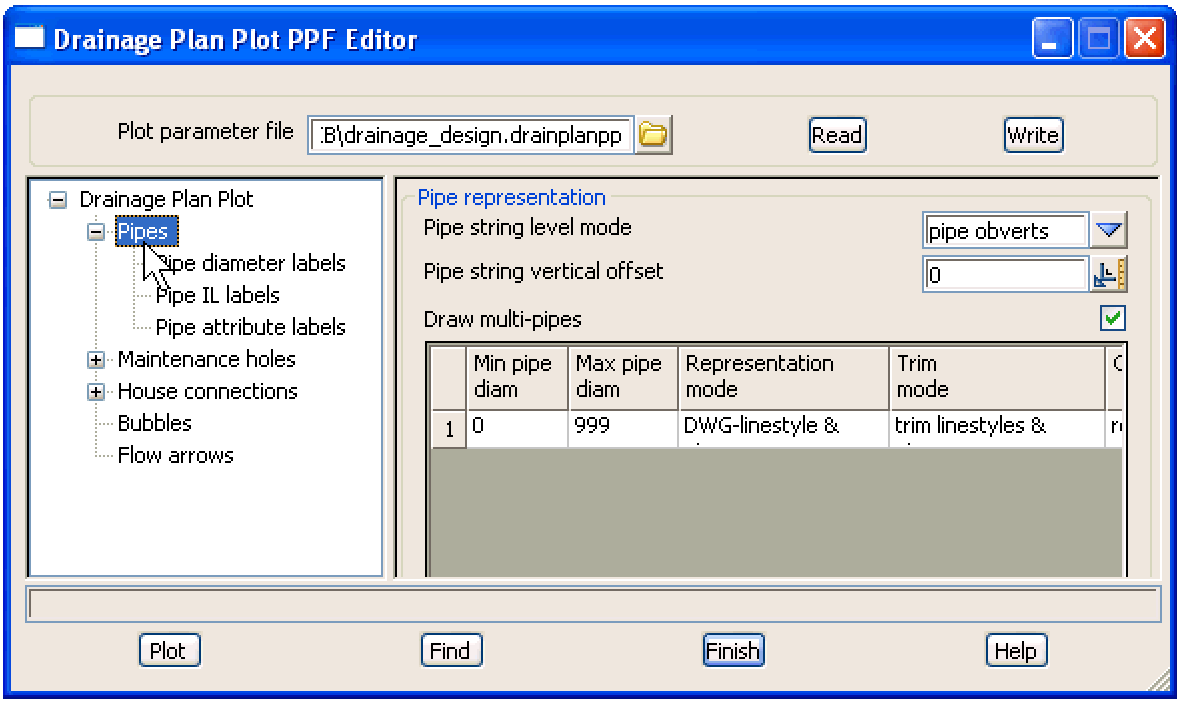
17.1 Creating the Obvert strings.

Run the Drainage Plan Plot by selecting Plot from the Drainage Network Editor.



Select the drainage_design ppf file from the library.

Select the folder icon then **Open** the ppf.

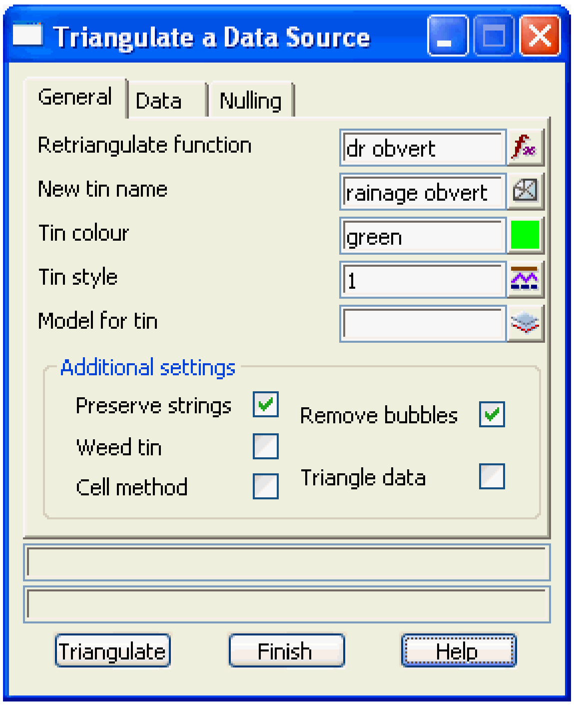


Confirm that the **Pipe string level mode** is set to **pipe obverts** and that all symbols with elevations are not used. This include:

- MH setout points,
- Flow arrows,
- pit symbols

Now plot the drawing and we will use the pipe strings at the obvert level to create a tin.

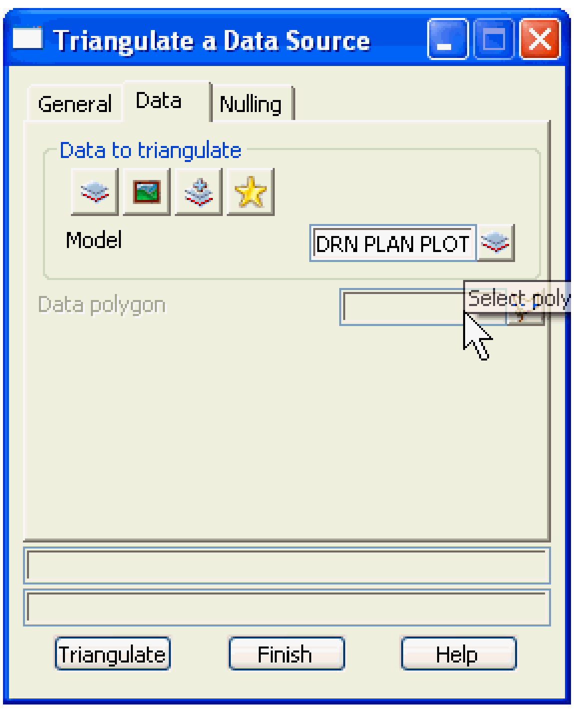
17.2 Create and Null the Obvert tin

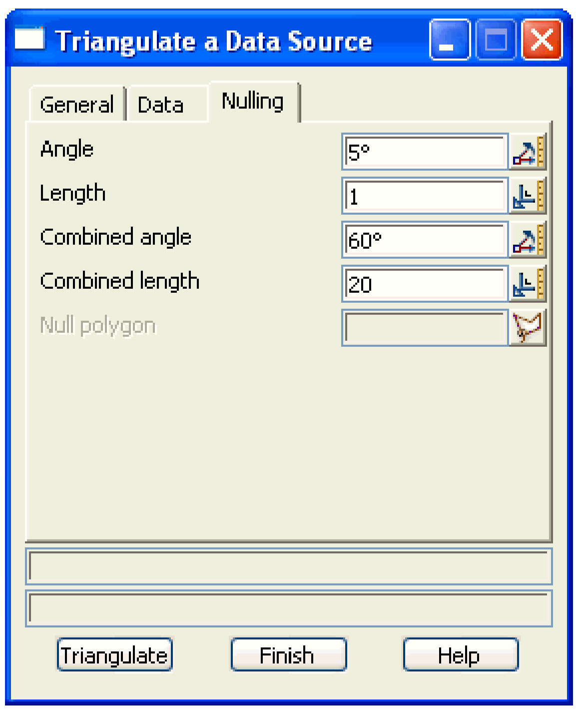


To create the obvert tin select.

Tins->Create->Triangulate Data

Enter a **New tin name** as desired.





Select the **Data** tab.

Select the plot model created above.

Select the Nulling tab

Enter a **Length** that is less than the pit diameter and greater than the largest pipe.

Select **Triangulate**

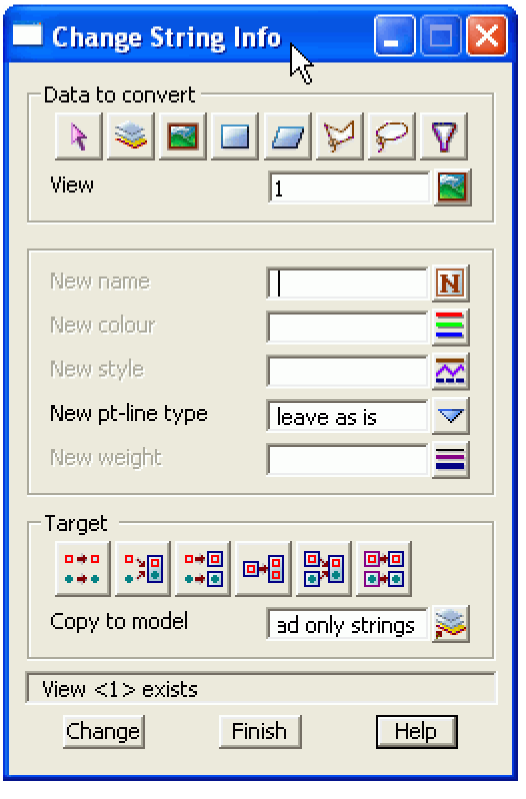
The tin will now only exist on top of the pipe and it is ready to use for hatching.

17.3 Create a Roads Only Tin

The design tin is already nulled to remove the long triangles. If the footpaths were to be excluded from the tin then they should be removed from the road design strings. You could run a template that did not include the footpath and has a **Final Maximum slope width** of zero or just copy the desired road strings to one model and remove unwanted strings. We will use the later techniques.

Add all of the road string and kerb return models onto one view and then select

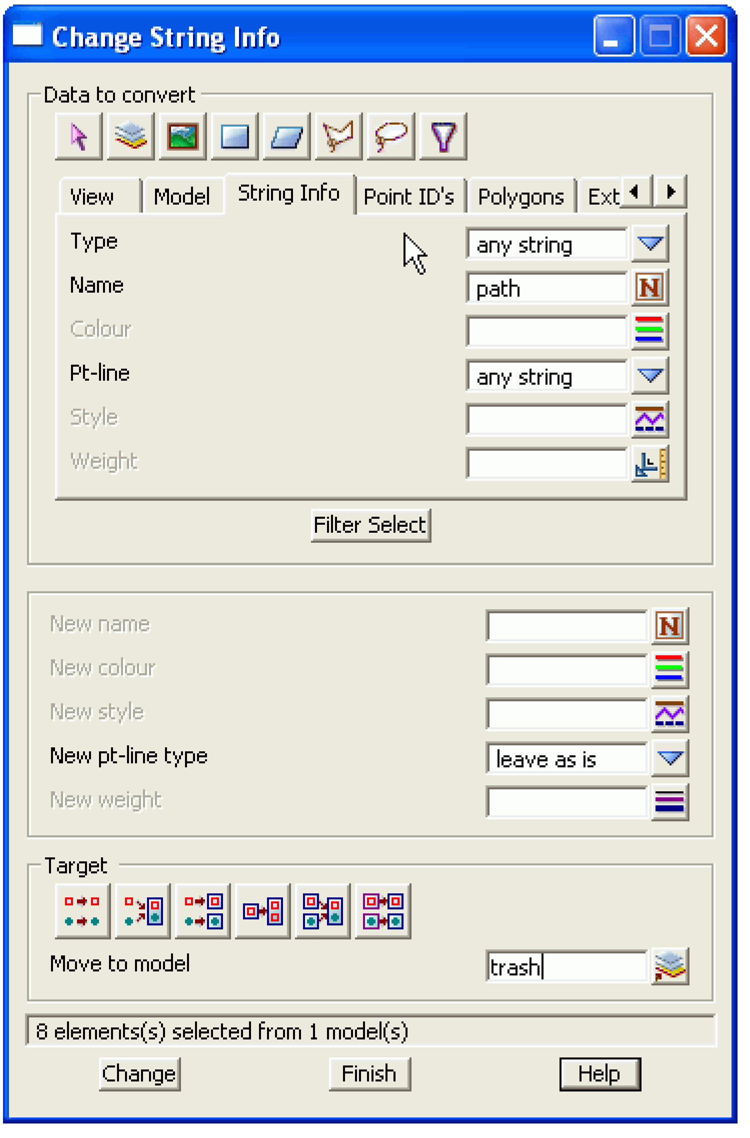
Utilities->A-G->Change



View select the view that contains all of the road strings

Copy to model type the name of a model for all of the road strings.

Select **Change** copy the strings.



Select the **Filter** icon at the end of the selection strip. Now select the **Model** tab and select the **Name** of the model that holds all of your road and kerb strings **road strings only** (tab not shown here).

Select the **String Info** tab and enter the name of the strings that you want to remove from the road only model. In this case it is **path.**

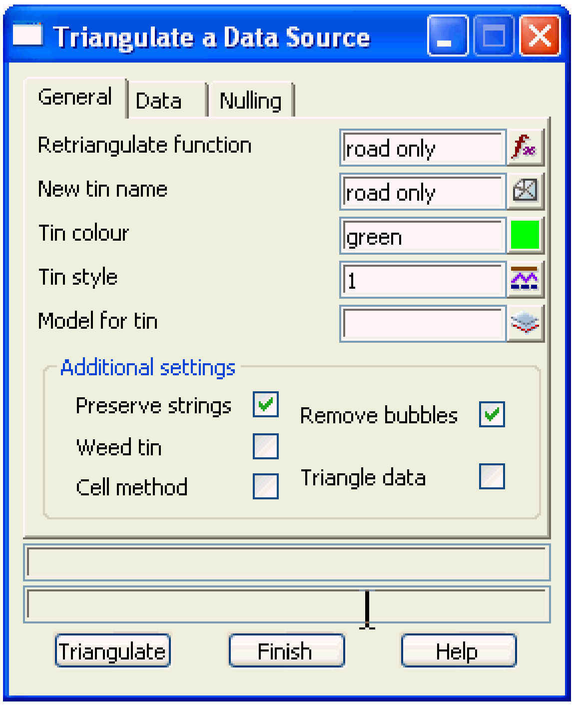
Select **Filter Select** to select these strings.

Select the second **Target** button **Move to model** and select the model **trash** and then select **Change**.

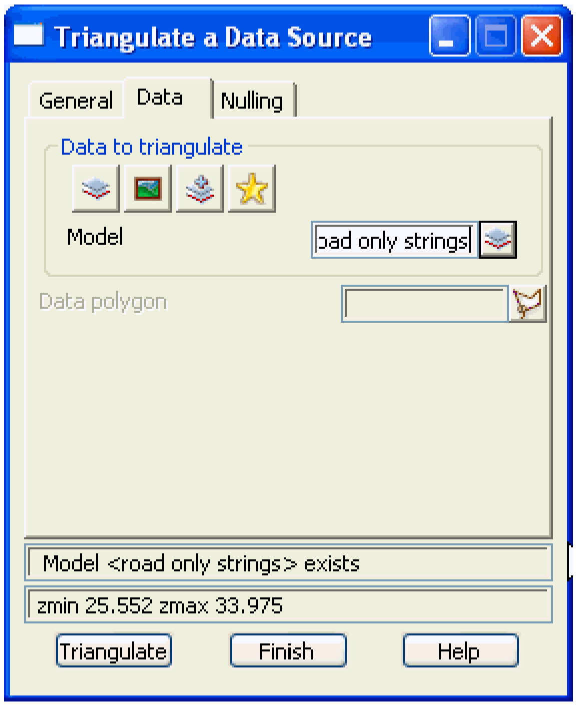
Repeat this for string names **int.**

We are now ready to triangulate the road only tin.

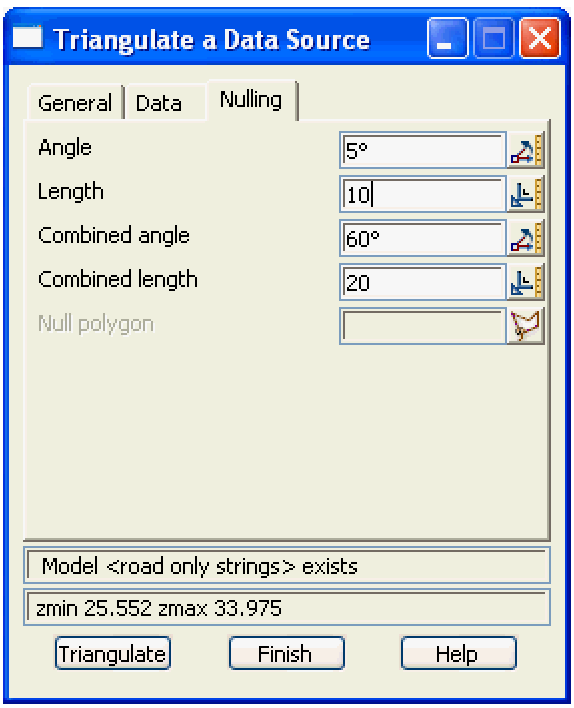
Now to create the tin select **Tins->Create->Triangulate data**



Enter a function name (optional) and **New tin name** and then select the **Data** tab.



Model - Select the model containing the road only strings.



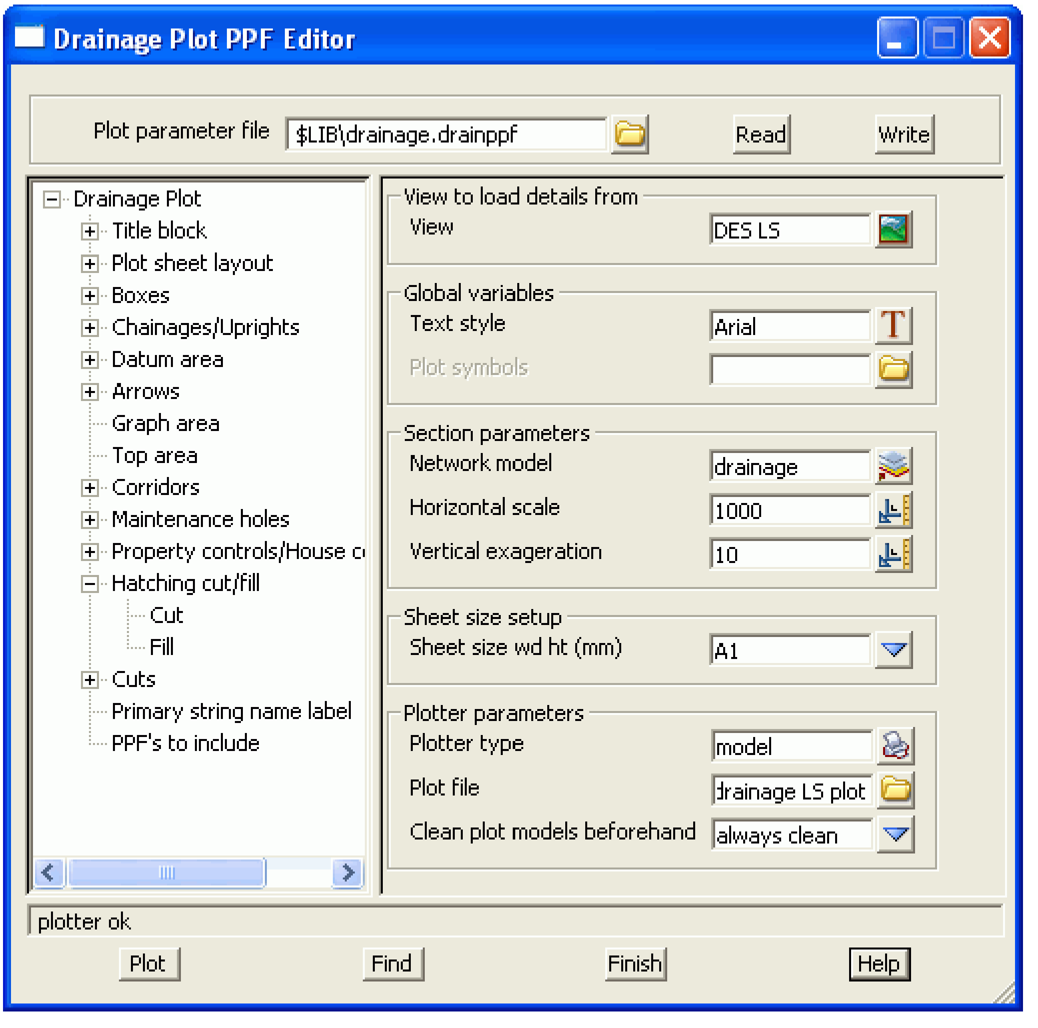
Length - Select a length just greater than the width of half the road so that the tin will not be nulled from the end of the roads.

Select **Triangulate.**.

17.4 Use the Hatching Feature in Drainage Longsections

We are now ready to create the drainage longsection plots. Set up a section view with the a string in the drainage model profiled, the vertical exaggeration set, the desired tins displayed and the service models added. From the main menu select,

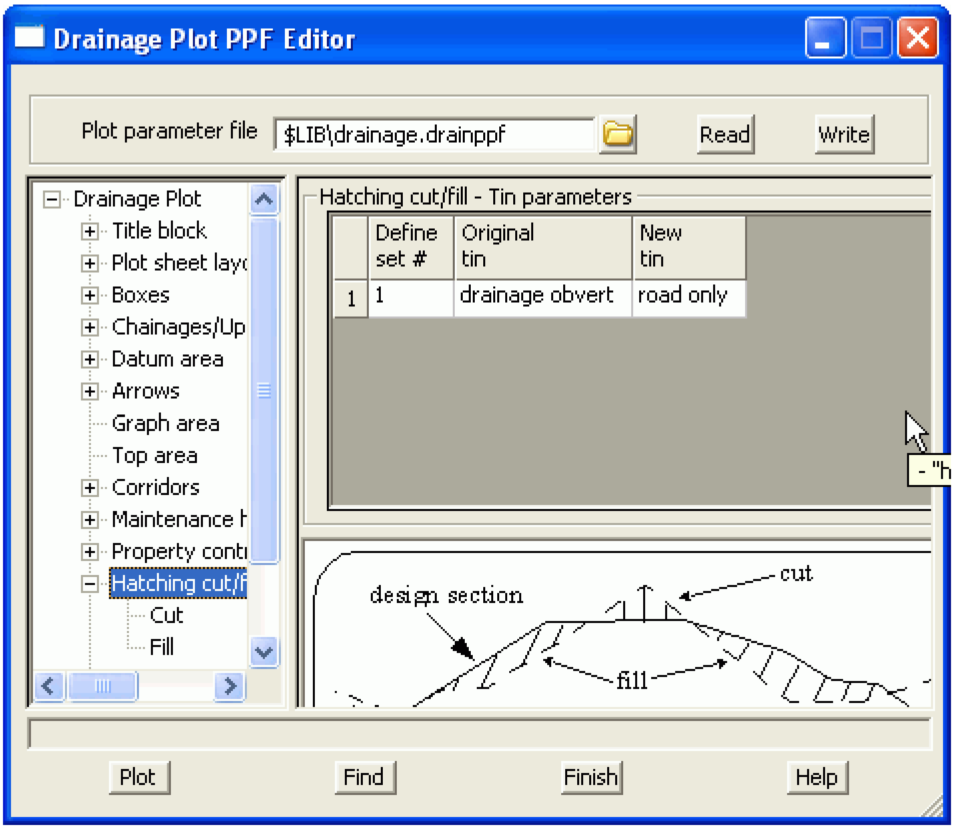
Design->Drainage-Sewer->Plots->Longsections



From the **Plot parameter file** field select a drainage longsection ppf from the library and then select **Read.**

In the **View to load details from** field select the section view you have setup for the drainage long section.

Now select **Hatching cut/fill**

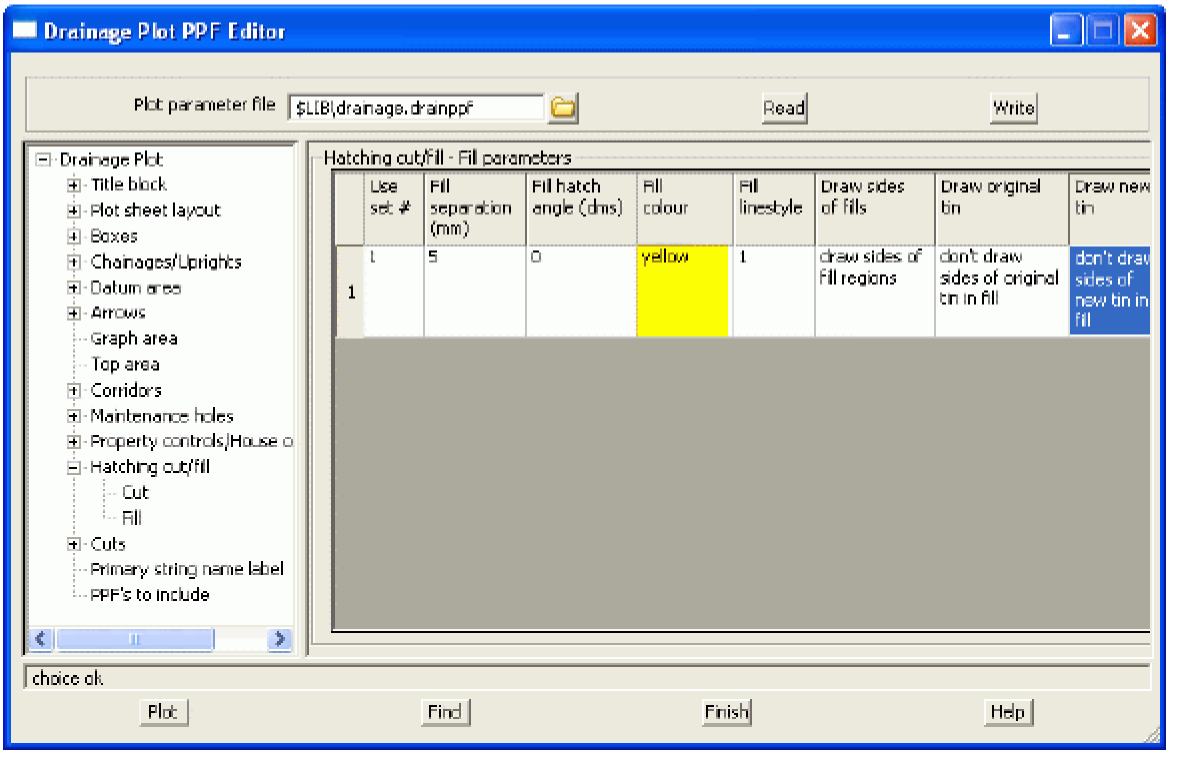


Define set is set to 1.

Original tin is set to the obvert tin.

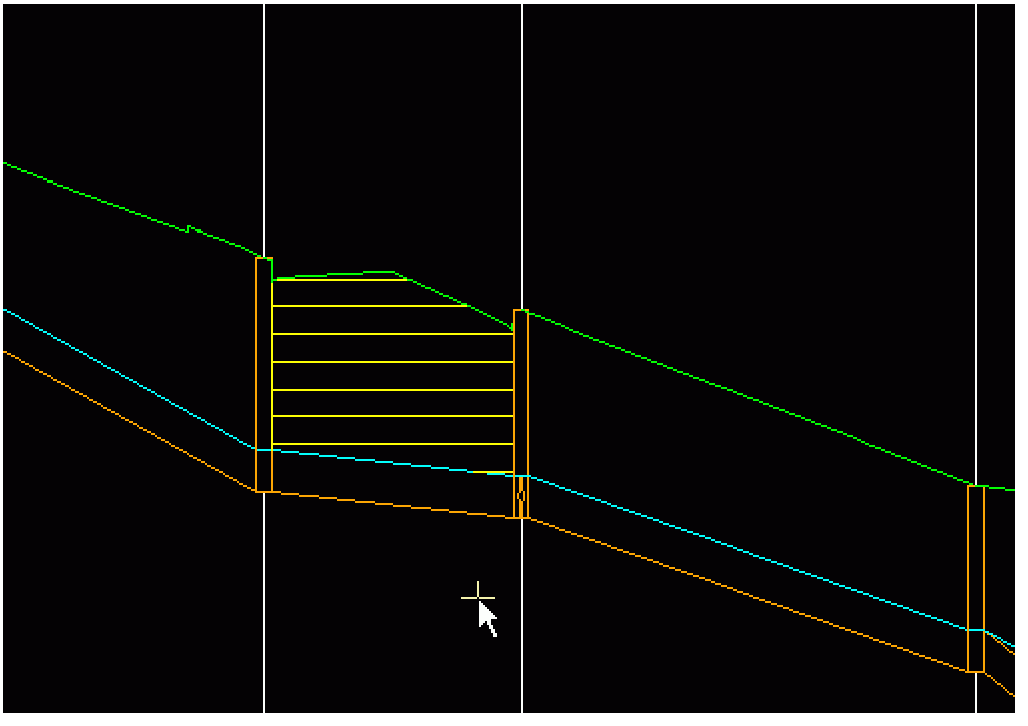
New tin is set to the road only tin.

Now select the + beside the **Hatching cut/fill** and then select **fill.**



Use set # is entered as 1. **Fill separation, hatch angle, colour and linestyle** selected as desired. **Draw sides of tin, Draw original tin** and **Draw new tin** are not required unless you want these extra line in the drawing.

Select **Plot** and add the plot model **drainage LS plot1** onto a PLAN view to preview the drawing.



The hatching is shown to the left at a 2mm spacing with the tins and sides not drawn.

18.0 Flooded Width Analysis and HEC RAS

There are 2 flooded width methods in the drainage module. The first can be used with any of the external drainage packages and the second is part of the 12d storm analysis. The first is found on the main menu

Design->Drainage-Sewer->Calc flooded widths

The name of the overland flow string becomes the name of the HECRAS project.

1.0

1.0 Dynamic Stormwater Design

This manual, the **Dynamic Stormwater Design,** describes how to use 12d model to calculate catchment hydrographs and perform unsteady flow hydraulic computations on drainage networks.

The rainfall runoff and loss methods supported include the time area unit hydrograph (ILSAX) with Horton infiltration (from runoff), and the New Zealand SCS hydrology method.

The starting point for this manual is a 12d drainage network already constructed with catchment areas assigned, inlet and manhole properties set and vertical grading applied. These tasks are described in detail in the **Stormwater Part 1 Design Course.**:The network will have open channels, bypass flow routes and pits with inlet capacity settings as described in **Stormwater Part 2Design Course**

This manual, **Dynamic Stormwater Design Course** contains**:**

- review and modifying the 12d hydro file with respect to IFD data, rainfall temporal patterns, infiltration loss methods (horton and SCS curve numbers),
- selecting the hydrological runoff method (Ilsax2 or NZ SCS) and the infiltration loss method,
- using cross section shapes cut from the tin for bypass flow paths and open channels
- running the analysis and reviewing the graphical output, plan and long section plots and the custom spreadsheet reports.
- create parallel pipes with different sizes and invert levels,
- adding and initial sizing of storage basin with fixed discharge,
- discuss graphical basin links,
- create outlet structures including, culverts, weir spillways, orifice controls, bubble up pits,
- underground storage tanks,
- bypass over road crown
-

2.0 Starting with a Basic Drainage Network

In this document, the generic terms **pit** or **manhole** refer to inlets, catch basins and manholes. The exception to this is when referring to inlet types in the Drainage Network Editor (DNE). The term **manhole** specifically refers to an inlet with zero inlet capacity. Pit types, dimensions and inlet capacities are set in the drainage.4d file.

These course notes assume that you have completed the Stormwater Design Course and that you have experience creating 12d model drainage networks with catchments areas. This course will begin with a completed drainage design found in the directory

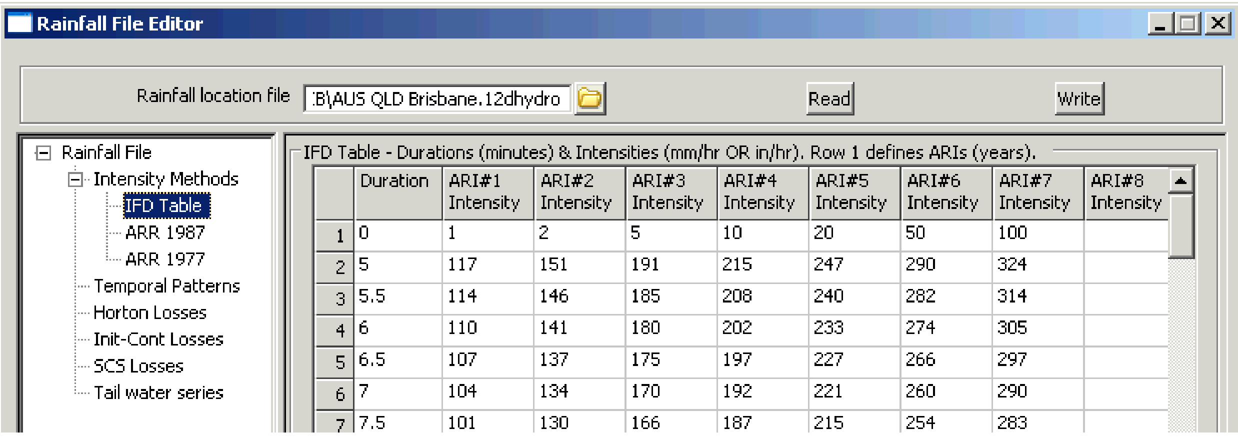
\12d\9.00\Courses\drainage\basins

The project name is basins.

3.0 Hydro-Rainfall File with Infiltration

Access the rainfall editor from the main menu

Design->Drainage-Sewer->Rainfall Editor



Key Points

1. Several rainfall files are shipped in the 12d library as examples.
2. One of the 3 Intensity method MUST be entered. Do not leave blank rows in the IFD Table.
3. All dynamic drainage runs require temporal patterns.
4. Dynamic drainage (ILSAX 2) requires Horton losses
5. Dynamic drainage (NZ SCS) requires SCS losses
6. Rainfall data is in mm/hr NOT mm (important for NZ)
7. Tailwater series are time versus elevation data for outlets.
8. Initial and continuing losses are not supported in release V9c1e

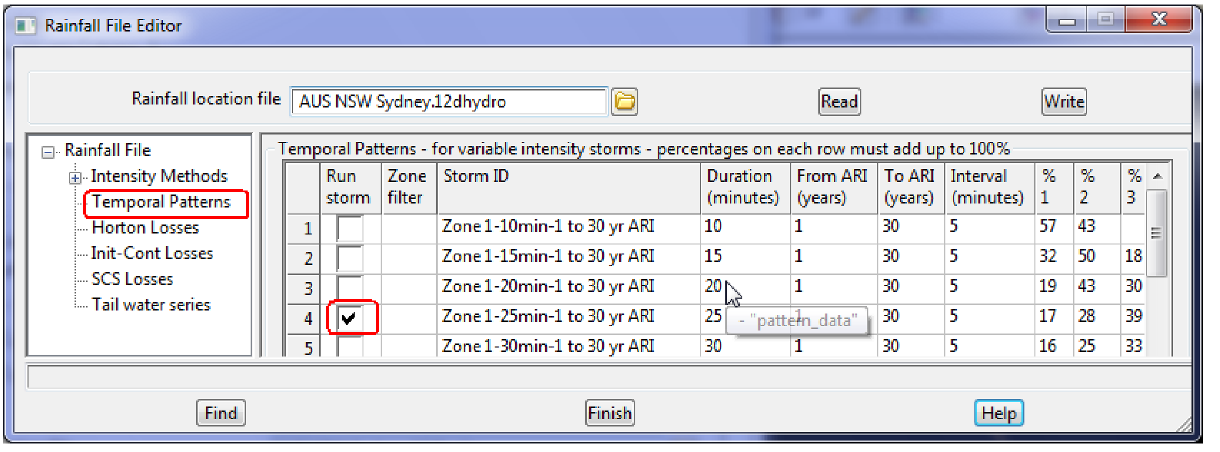
3.1 Intensity Frequency Duration (IFD) Data

The **IFD table** method is the most common (refer to figure above). The rainfall intensity is entered in mm/hr for metric data and in/hr for US units. The top row is the return period in years and the first column is the rainfall duration in minutes. The top left corner cell is always zero.

When using the TP 108 method for the Auckland region, the 24hr rainfall depth is extracted from the Appendix A and then converted to mm/hr before entering it into the IFD table. Thus only 2 lines may exist in the 12d grid.

Note that areal reduction factors must be accounted for by the user as they are not included in the 12d rational hydrology engine.

3.2 Rainfall Temporal Patterns



Temporal patterns are referred to as storms in dynamic drainage. Several example hydro files are included in the 12d library. These examples have the temporal patterns for the minor and major storms in the 8 zones of Australia.

Temporal patterns are not required for the SCS NZ method as the standard 24 hour temporal pattern from TP108 is built into the 12d analysis engine.

The **Run storm**, **Zone filter, From ARI** and **To ARI** columns are used determine which storms are analysed (run). The **Run storm** column must be checked for that temporal pattern to be analsyed. Many storms may be selected.

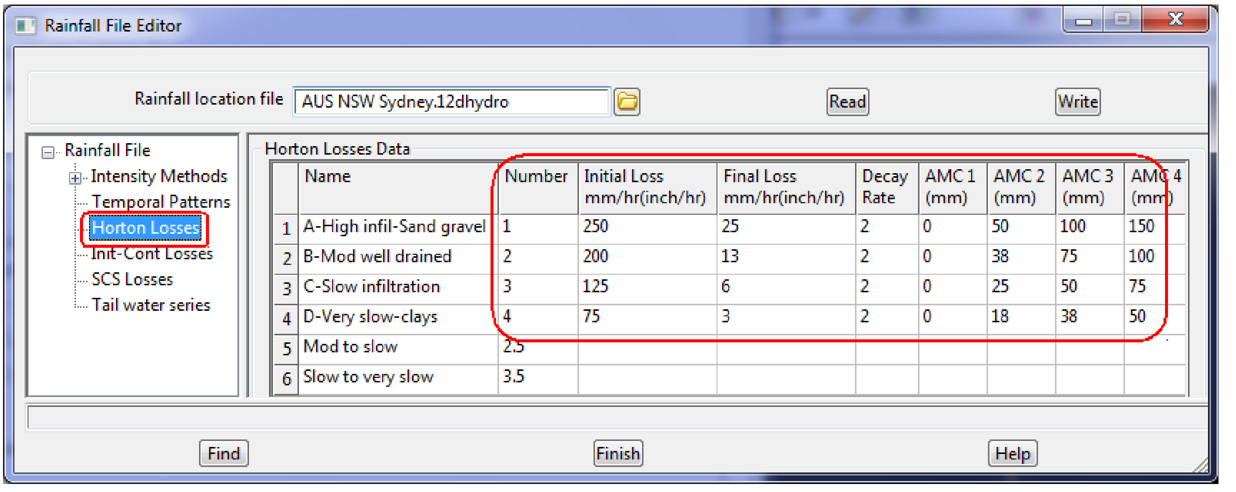
The **Zone filter** is optional. Entering a value here will allow the selected storms to be further filterd. A **Zone filter** field (accepts wild card characters) is found on the DNE Global tab that is used to determine which of the selected storms (paragraph above) are analysed.

The **ARI** field on the Run panel is used with the **From ARI** and **To ARI** columns. The value on the run panel must be within the From-To range for the storm to be analysed.

The **Duration** column determines the total length of the storm. This value divided by the **Interval** must be a whole number and this number determines the number of **%** values to be entered to the right of the **Interval column.** The total of the percentage must equal 100.

3.3 Horton Losses

The pervious portion of the catchments used in the **ILSAX 2** analysis will have a loss type defined describing the soil type. The loss type is defined in the DNE Default->catchment and catchment tabs.



These soil types use the classifications of Terstriep and Stall (1974), based on the system developed by the U.S. Department of Agriculture. The default values entered from the library represent the soil types of

1. Type A - low runoff potential, high infiltration rates (consists of sand and gravel
2. Type B - moderate infiltration rates and moderately well-drained
3. Type C - slow infiltration rates (may have layers that impede downward movement of water)
4. Type D - high runoff potential, very slow infiltration rates (consists of clays with a permanent high water table and a high swelling potential)

Numbers are assigned to each soil type to allow interpolation between the defined soil types. When interpolated values are used they must be included in the list (2.5 and 3.5 for example). Interpolated values do not need loss data entered. If loss data is entered for the interpolated names then this data will be used rather than an interpolation occurring. If any loss data is entered then all of the values must be entered.

Four preset AMC points are defined in the rainfall file to mark AMC conditions ranging from dry (AMC1) to saturated (AMC4). The required data for each line is the **Initial loss rate, Final loss rate, decay rate** and 4 antecedent moisture conditions **(AMCs).** The AMC values are entered in depth of rainfall (mm) they represent the total rainfall prior to the start of the temporal pattern.

The AMC point numbers are set once for all catchments on the DNE Global tab. Value between 1 and 4 (decimal value are permitted) are entered for the minor and major events.

3.4 SCS NZ Losses (Initial Abstraction and Curve Numbers)

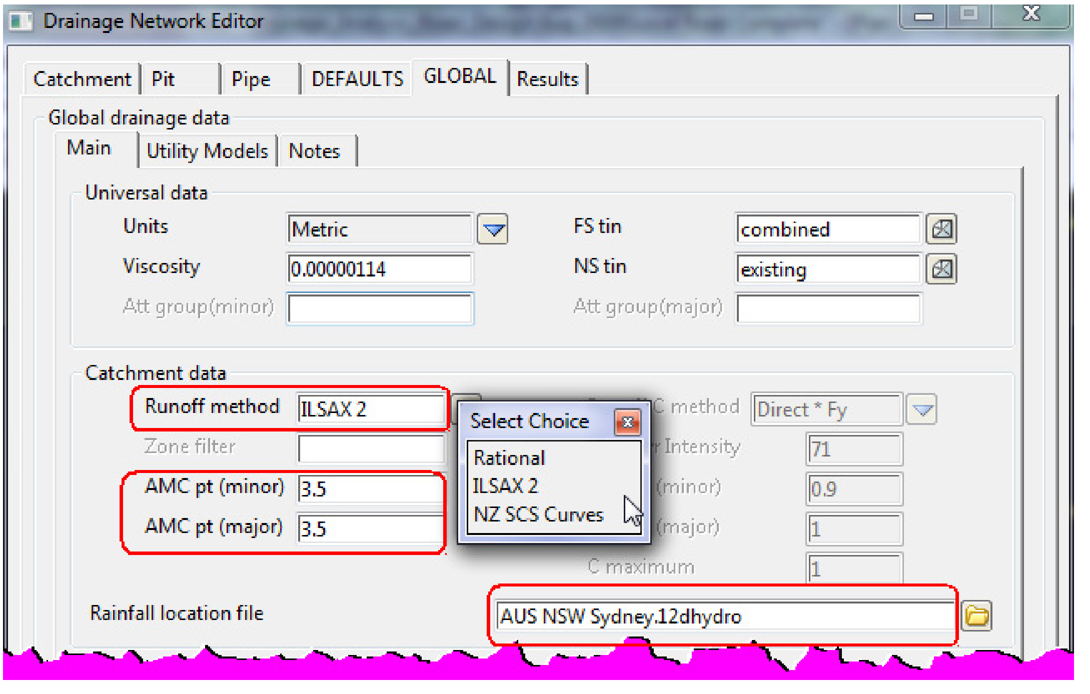
The SCS NZ method uses Initial abstraction (Ia) and the curve number (CN) to determine the losses for the catchments. Names are given to the SCS curve numbers in the rainfall file. These names and Ia (entered as storage values) and selected in the DNE catchment data.

A curve number of 0 results in zero runoff while a CN=100 results in 100% runoff. TP 108 recommends the selection of the curve number by identifying 1) the soil type and 2) the land use. A CN=98 an Ia=0 are recommended for impervious areas.

4.0 Rational to Dynamic ILSAX2 in the DNE

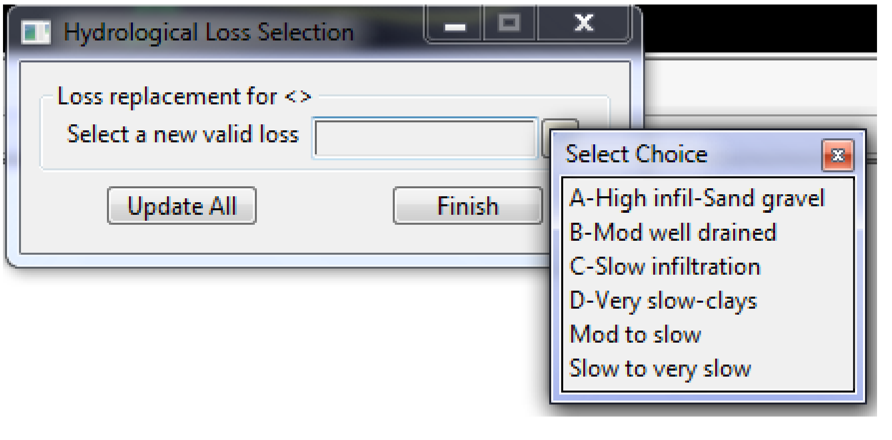
The user does not have to start with the rational hydrology method but since this method has a pipe sizing algorithm it is often run first. The following steps will prepare a drainage network for dynamic hydrology.

Changing to ILSAX 2 is done on the DNE Global->Main tab.



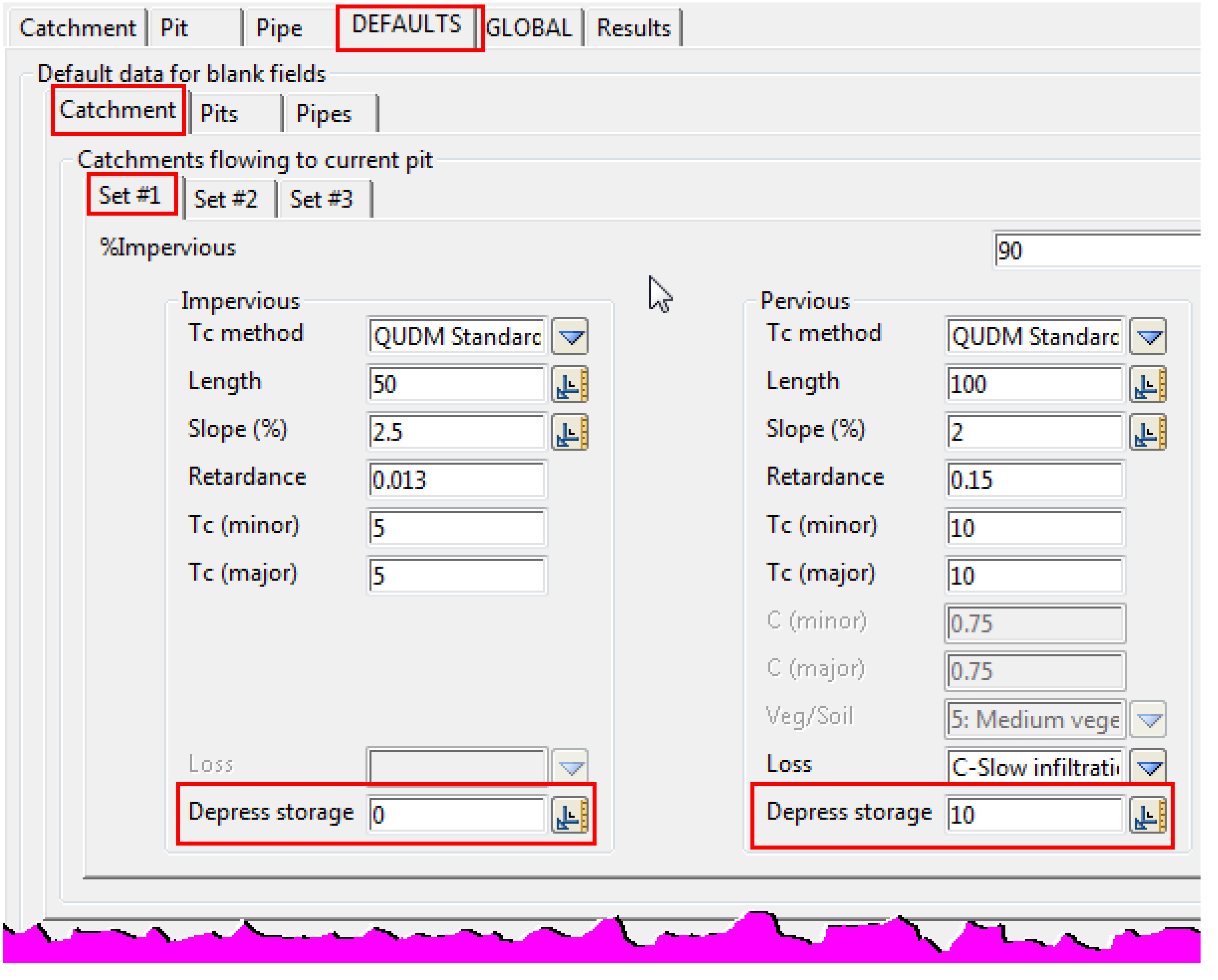
4.1 Infiltration Losses

If this is the first time in this project you have switched to ILSAX2 you will be prompted for the default infiltration loss. These loss types are retrieved from the rainfall location file (Horton losses) specified above. Select one of these and then select **Update All**. These losses are only applied to the previous percentage of the catchment.



4.2 Storage Losses

Both the impervious and pervious portions of the catchments have storage losses. This amount of rainfall depth (in mm) is removed from the start of the rainfall pattern. The default values are entered on the **Defaults Catchments** Tab.



5.0 Rainfall Runoff Calculations

12d currently supports 2 rainfall runoff methods.

See ILSAX 2
See SCS NZ

5.1 ILSAX 2

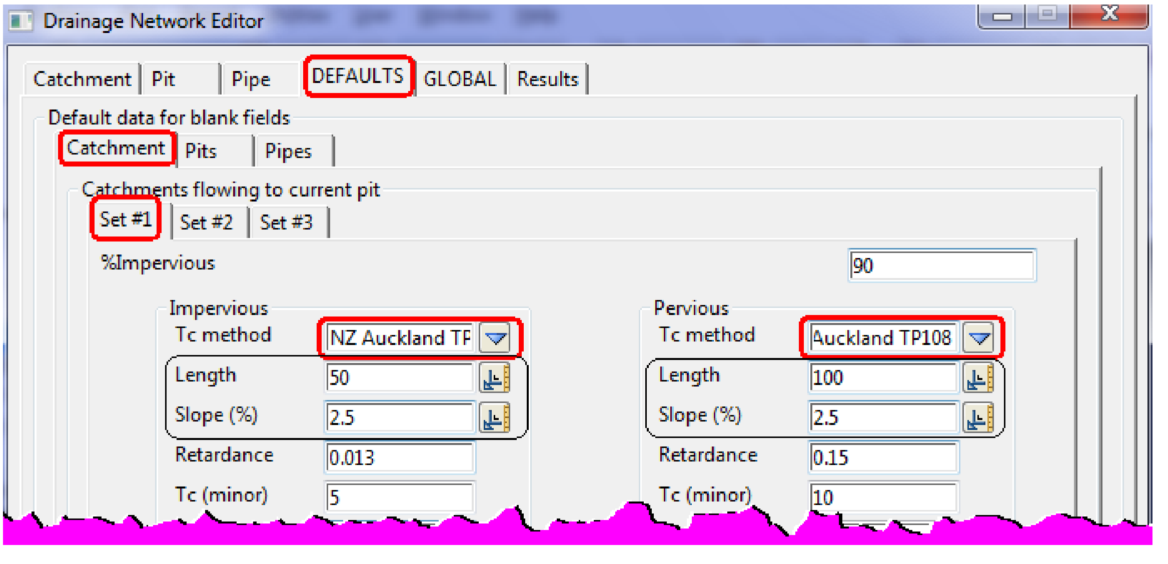
The ILSAX mode 2 uses a triangular shaped unit hydrograph with the time to peak equal to the recession time. The time to peak equals the tc value throughout the 12d DNE.

5.2 SCS NZ

The SCS NZ uses a unit hydrograph as described in TP 108 Table 4.1 (hydrograph number of 3/4). 12d calculates the time to peak value (tp) = 2/3 the time of concentration (tc).

The %impervious value is used to create a composte CN and a composite initial abstraction Ia

One of the optional tc methods uses the equation derived from a regression analysis of Auckland catchments (BCHF, 1999c). On the **Defaults->Catchments->Set 1** tab set the Impervious and Pervious Tc methods to **NZ Auckland TP108**. The default length and slope values will be used unless they are specified for each catchment.



6.0 Dynamic Hydraulics

As with the rational method hgl calculations it is essential that the pit grate levels are entered correctly. With no bypass specified, water is lost from the analysis when the hgl exceeds the grate level. When bypass is specified the water will not be lost until the level exceeds the highest bank of the bypass section shape (see below).

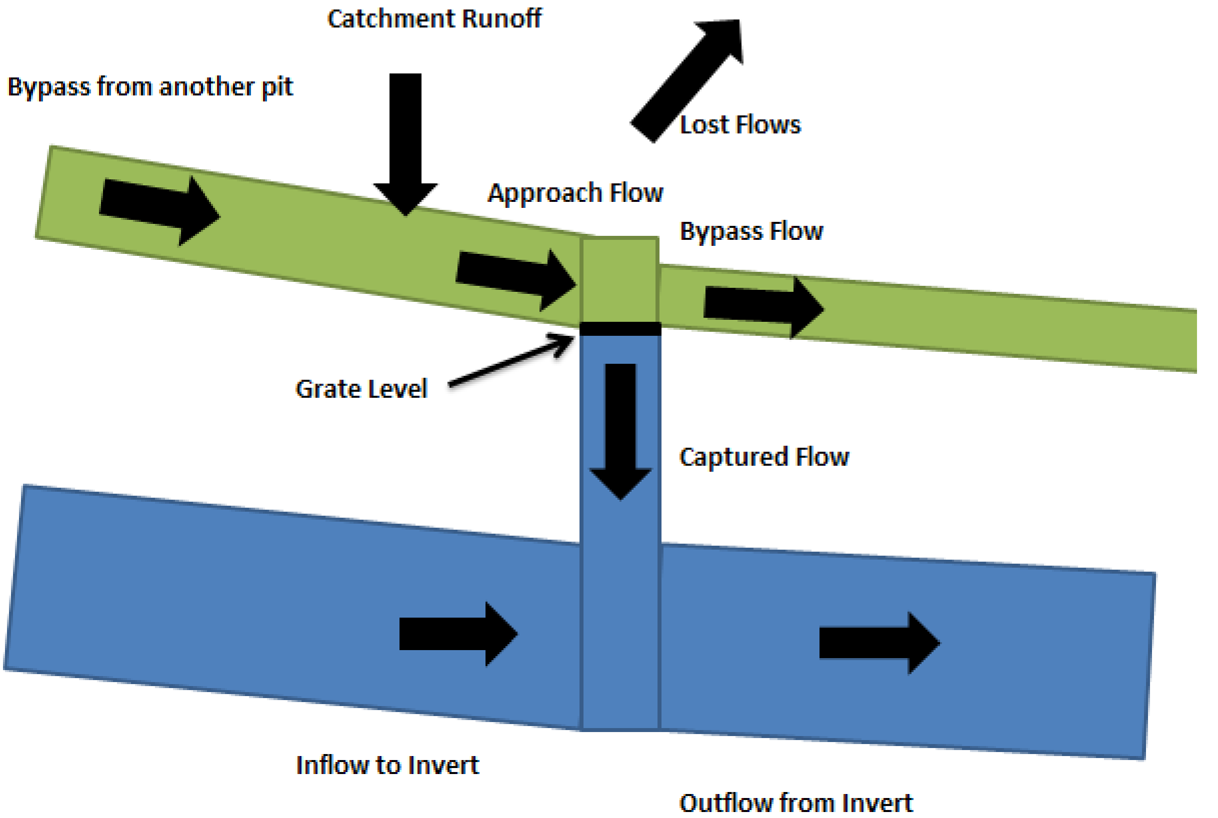
12d uses the St Venant unsteady flow and continuity equations to solve for the flows in the pipe network. With these equations it is the water level in the pits that determines how much flow travels through the conduits. The hgl is not calculated from the flows. Viewing the results from this perspective will help immensely.

6.1 Flows in and out of the Pits

The diagram below shows the possible flows through a pit. When bypass is used, there are 2 hgl levels; one level for all of the conduits with inverts at or above the grate level and another for conduits with inverts below the grate level.

The pit inlet capacity and choke factors determine the amount of approach flow that is captured. If the HGL of the subsurface reaches the grate level, the captured flow becomes zero and only the surface hgl is used to determine the flows in all conduits. When the hgl drops below the grate again, 2 hgl levels will again be used.

The continuity balance is monitored during analysis and the results of this check are stored as a model attribute “dynamic/xx/calculated hydraulic continuity error percent” where xx is the duration of the storm. A detailed output file is also created for each storm analysed. The file name begins with the name of the drainage model and ends with the storm duration (plus the extension rpt).



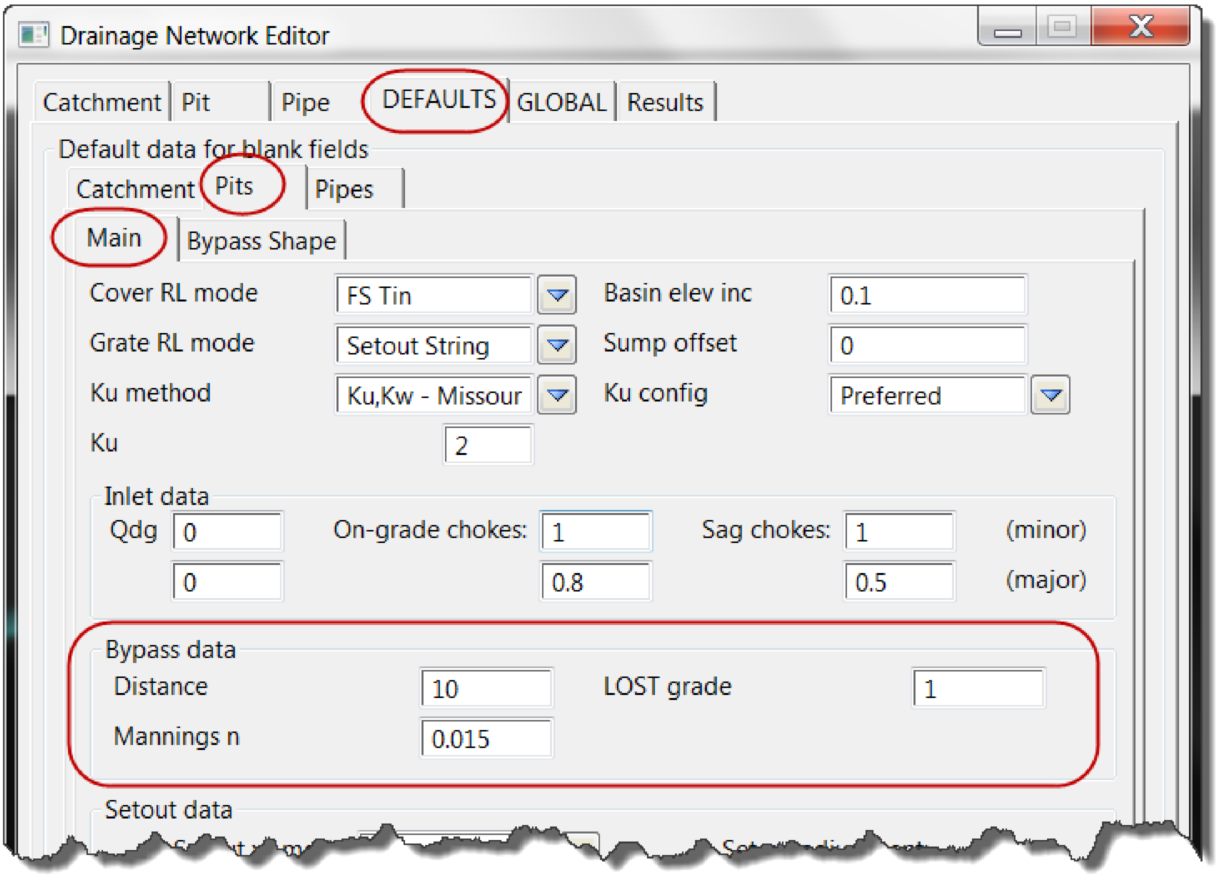
6.2 Default Bypass Section Shapes and Slopes

In the rational method, bypass flows (water that could not enter the inlet because of either inlet constrictions or hgl levels) were bypassed to the **Bypass pit**. With the dynamic analysis, a channel shape is required so that the hgl levels and the storage effects can be modelled. If all of the bypass routes have the same shape then you need only cut one default shape and it will be used for all routes. However you can enter a shape string for every bypass route if desired.

The bypass **distance** is determined from the length of the bypass flow string and the channel has a constant grade. If a changing grade or channel shape is required then the bypass should be to a drainage line modelled as a natural channel (see section further in the notes).

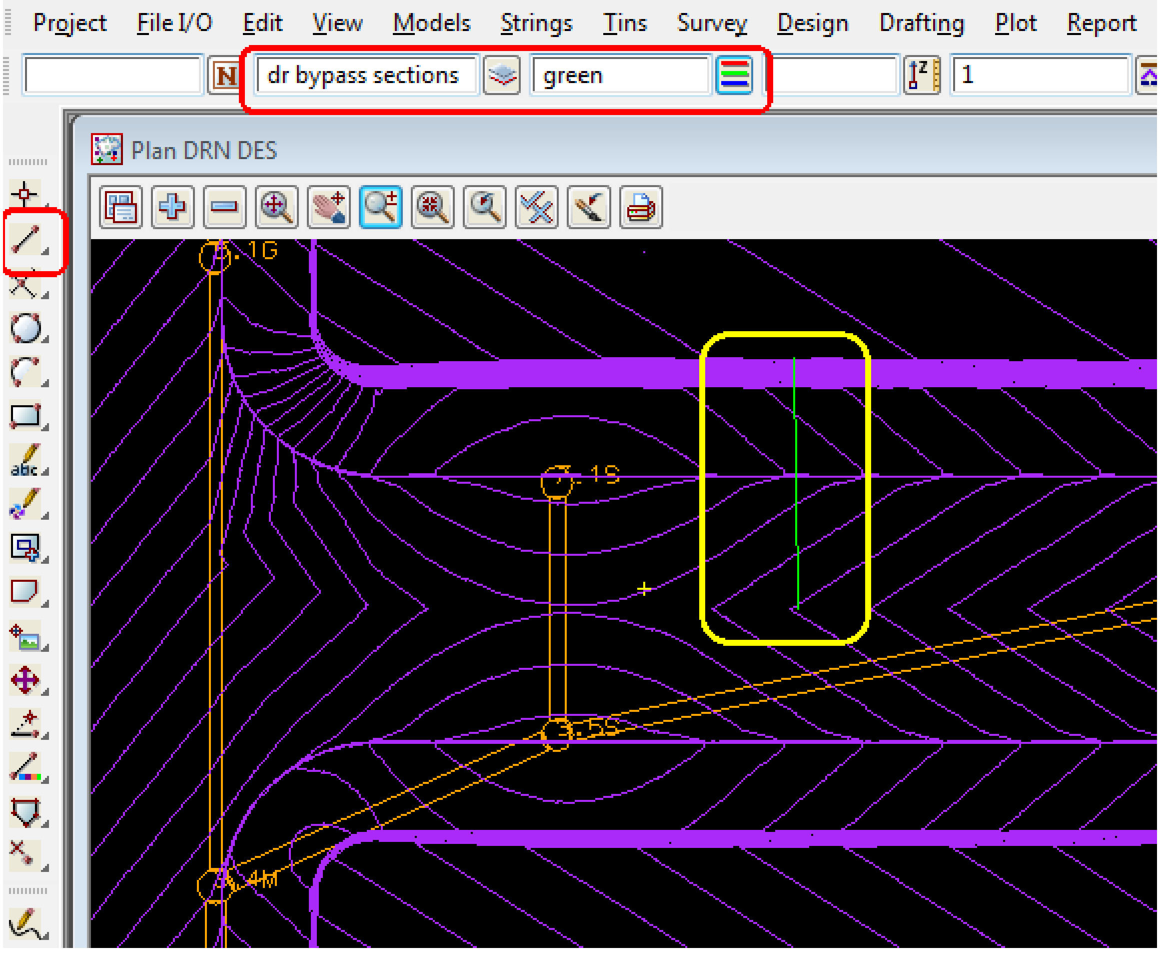
The invert elevations of the bypass channel are set by the grate elevation of the upstream and downstream pits. However, when the bypass is to the pit LOST then there is no downstream grate level so the slope of the bypass channel will be defined. The **Lost grade** value is specified here as 1%. 12d needs this to determine the water forces try to hold the water into the bypass channel.

The **Manning's n** value is the default for the entire section. If left and right bank n values are desired they are entered on the **Bypass Shape** tab.



6.3 Create and Editing Channel Shapes

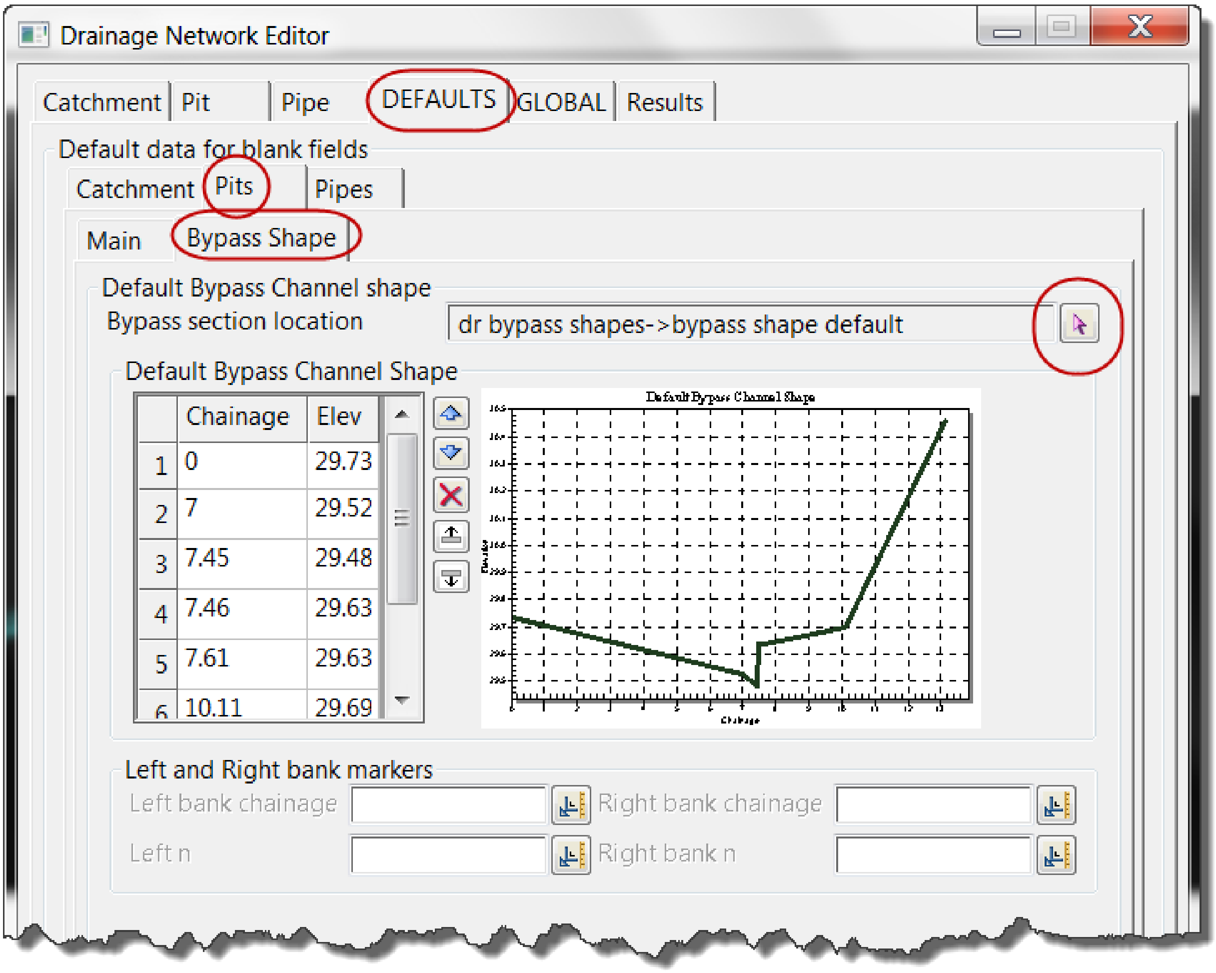
The channel shape may be entered manually or created by drawing a string perpendicular to the flow path and 12d will cut section from the finished surface tin. It is recommended that you keep you section shape strings in one model for data management purposes. The strings are drawn from left to right looking in the direction the bypass string (the direction of flow). To create this string enter the new **model name** in the **CAD Control Bar**. Entering a name for the string is optional. If only 2 points are to be used for the shape string then use the **Create Line** button from the **CAD** tool bar.



After drawing the string it needs to be selected via the **Bypass section location** pick button (see below). Once picked and accepted, select **Set Pit Details** to have the bypass shape cut from the finished surface tin specified on the Global tab. This shape may be seen on the **Bypass Shape** tab and the chainage elevation data displayed.

The elevations on this graph are not used. The data is relative to the lowest elevation on the section. This becomes level 0.0 for the shape and the channel inverts added to determine the final elevation.

Left and right bank n values may also be entered if desired. They **Left n** is used to the left of the **Left bank chainage** and the **Right n** is used to the right of the **Right bank chainage**. The centre chainage is entered on the **Default->Pits->Main** tab.

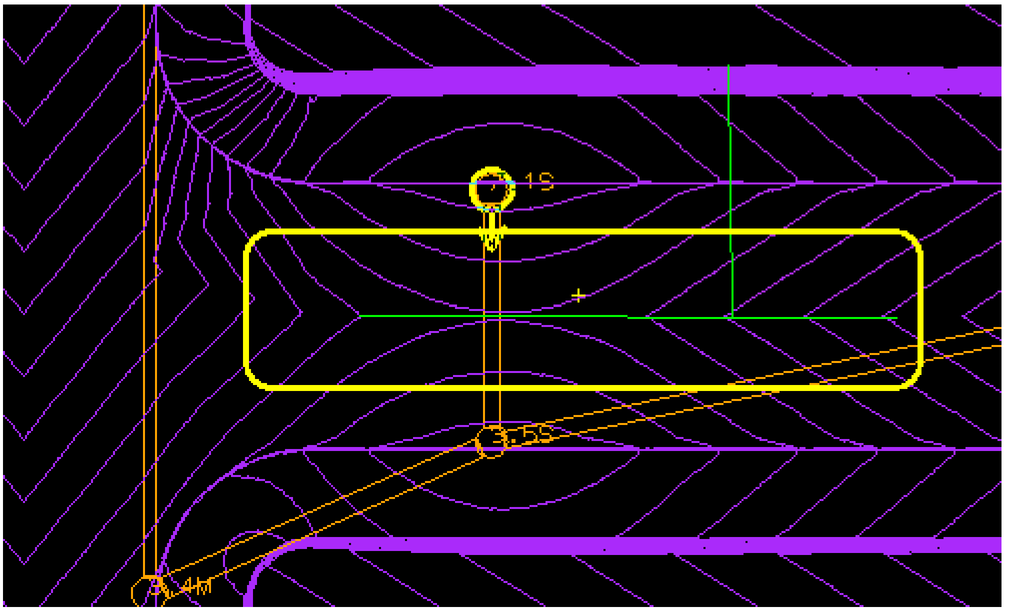


If you manually change any of the chainages and elevations they will be lost the next time you select **Set Pit Details**. To prevent this RB select the **Bypass section location** and select **Clear**. Now there is no string so the data cannot be recalculated.

6.4 Explicit Bypass Section Shapes and Slopes

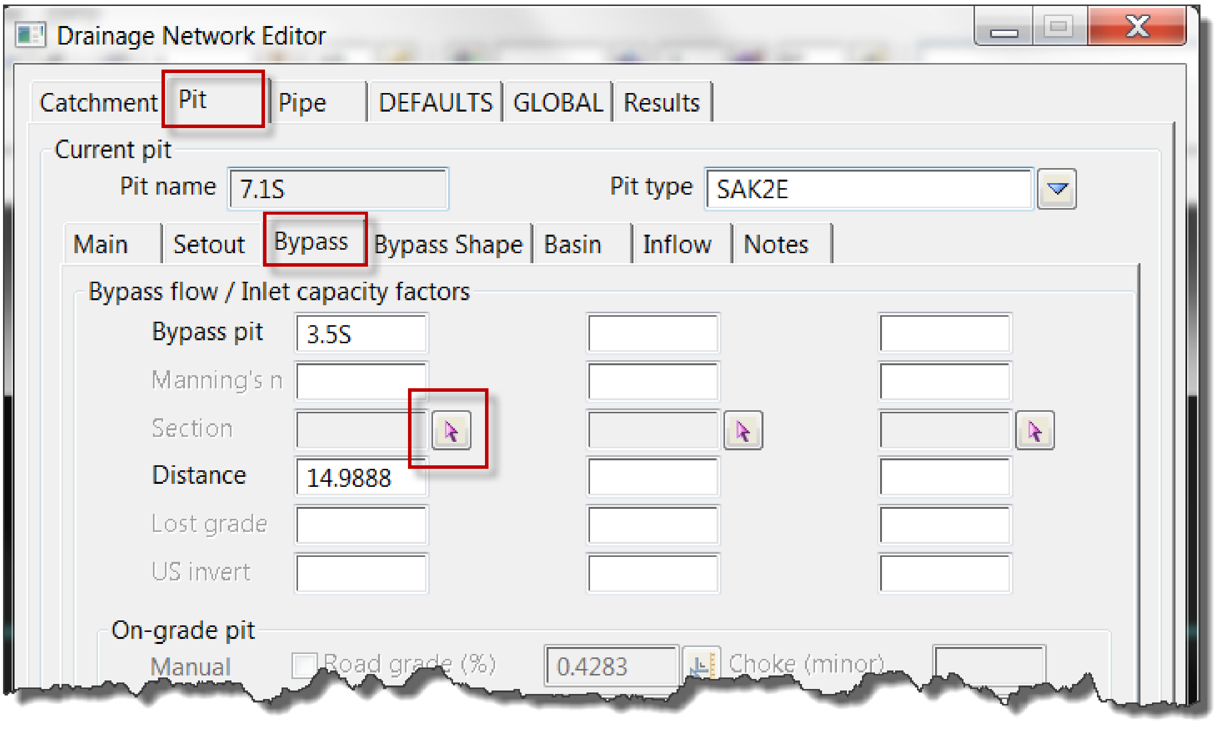
We will now explicitly define a bypass flow shape for bypass link. Water flow across the crown of the road should not use a kerb and gutter shape. Instead we need the shape and invert elevation for the road crown that the water will overtop.

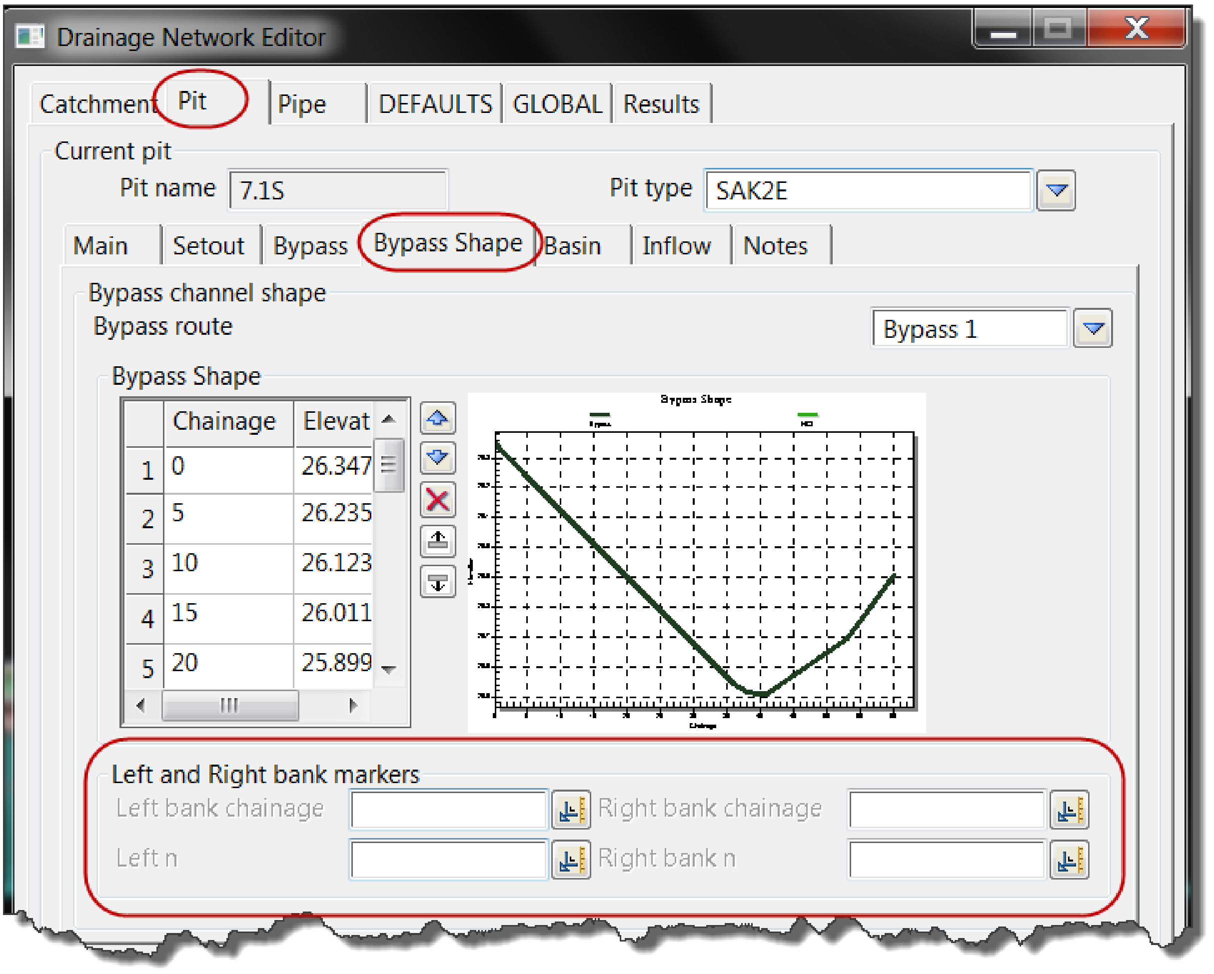
To create a shape string for the bypass across road 2, draw the string from east to west (left to right looking in the direction of flow).



Now select the bypass shape for crossing the crown of the road. Select the **Set Pit Details** button to have 12d calculate the bypass shape from the finished surface tin. Change to the Bypass Shape tab to see the shape. Note the **bypass pit** and **Distance** fields will already be completed from the bypass flow string calculations. Entering a manning's n value is optional.

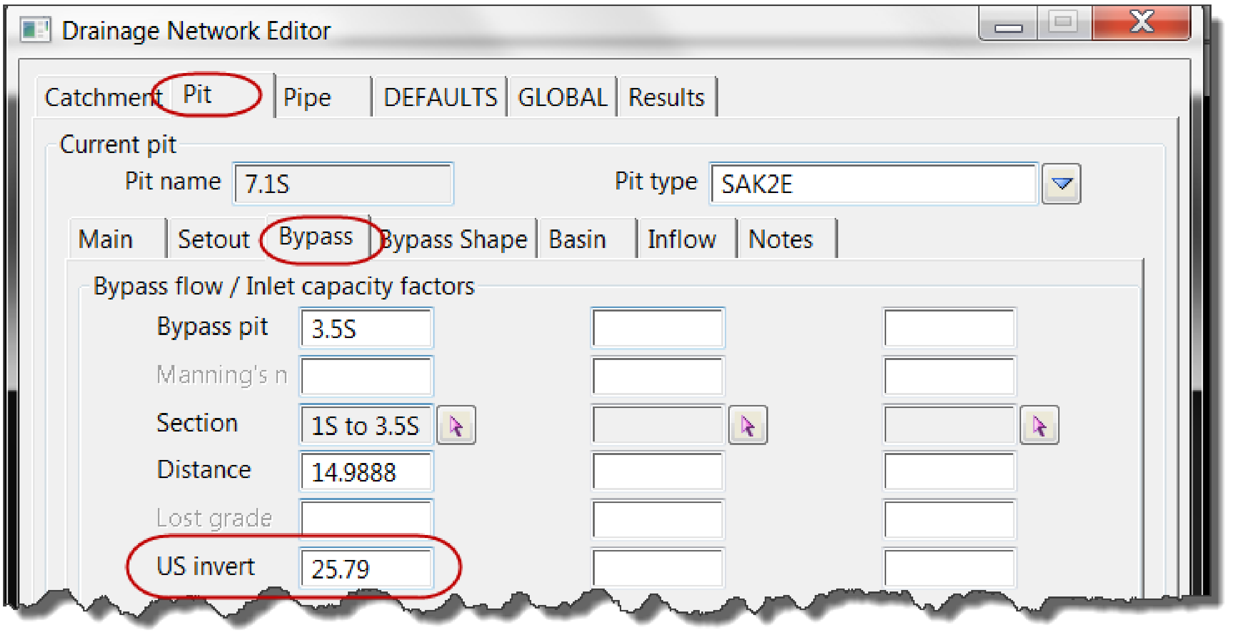
You will also note that there are 2 other bypass flow routes that can be used. To use these the bypass pit and distances will need to me manually set. The hgl above the grate determines the head that will drive the water down the various channels.





The enlarge the plot RB select inside the graph area and select **Maximise** from the menu. Note the lowest level on the section shape is approximately 25.5. This is the grate level of the pit. Press **Esc** or select the title area to close the graph. Left and right channel n values with their chainages may be entered in the fields below the graph. If these values are left blank the n value from the centre of the channel will be used, NOT the n values from the default bypass channel shape.

If this bypass shapes was to be used in the model now, the bypass channel would go from one grate level to the other without any reference to the crown of the road level. To model the flow path from the crown of the road to the pit 3.5S we will raise the upstream invert level to the road crest level (25.790).



Select the **Apply** button to store the changes and note that the lowest point of the bypass shape graph is now raise the 25.79.

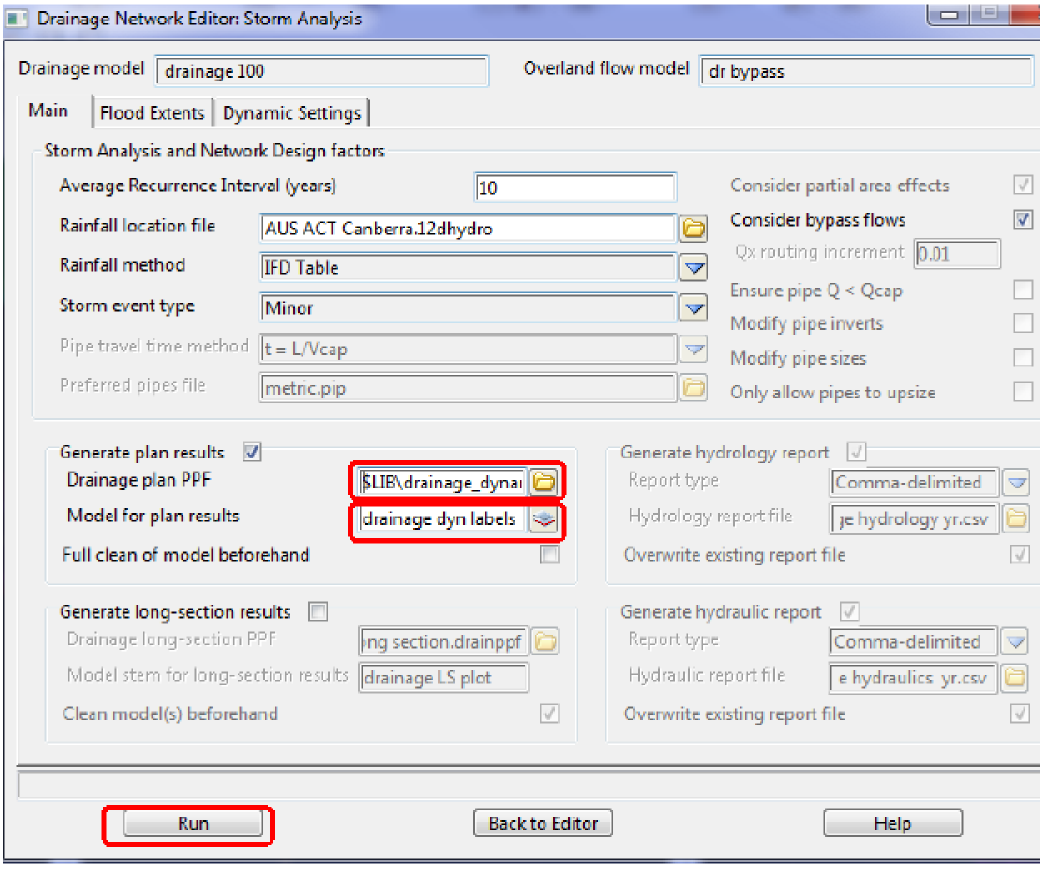
See Create and Editing Channel Shapes for details about manually entering data or changing the shape.

7.0 Running Dynamic Drainage

The dynamic engine analyses the network but does not design the components. Multiple storm patterns may be run simultaneously and the results from individual storms as well as the worst case from all currently analysed storms is saved. The results may be viewed graphically or in user defined reports. If your computer has a multiple core processor each storm will be processed by different processors and 12d will accumulate the results as the storms finish analysing.

7.1 Storm Analysis

Select the **Storm Analysis** button. To generate plan labels for the worst case dynamic results select the **drainage_dynamic_design** ppf file from the library. Also change the name of the **Model for plan results** so that you can keep the results from previous rational hydrology runs. Select the **Run** button.



A process window will be launched for each storm analysed and the status will be printed in the output window. **Always watch this window for messages**. Below are message from a sample run,

==> Checking Storm: 1

==> Storm: 1 OK

==> Updating Catchment Attributes

==> Updating Node Attributes

==> Updating Link Attributes

==> Updating Model Attributes

Elapsed Time: 6.67 secs

Run finished normally.

A run with errors would have a message like the following:

Dynamic Drainage Analysis (Build 5.1.026)

Dynamic Runoff Method: ILSAX 2

ari0.0

zone: 0

1 TP_Durations: 25.00

==> Solving Storm: 1

ERROR: Bypass section not defined for conduit: 7/A-SEP_to_6/A-SEP(S)

Model failed to solve.

7.2 Reviewing Results - Plan Labels

As with the rational method, the plan plots indicate the peak values. If multiple storms are run then this ppf file will print the max from all storms run.

So that the dynamic results are not mixed with the rational results, the worst case results are stored with the prefix **dynamic**. Results from individual storms have the prefix dynamic/xx where xx is the max time of the rainfall event.

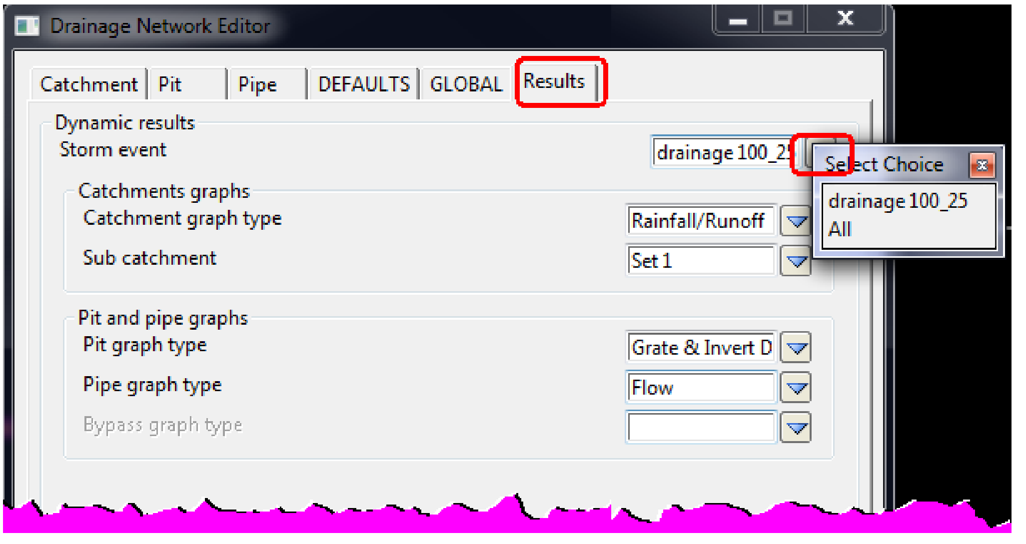
For example the pipe attribute for the maximum pipe flow

calculated pipe max flow	rational hydrology result
dynamic/calculated pipe max flow	worst case dynamic result
dynamic/calculated pipe max flow critical storm	storm where above results occurred
dynamic/25/calculated pipe max flow	25 min storm dynamic result
dynamic/calculated pipe max flow	worst case dynamic result

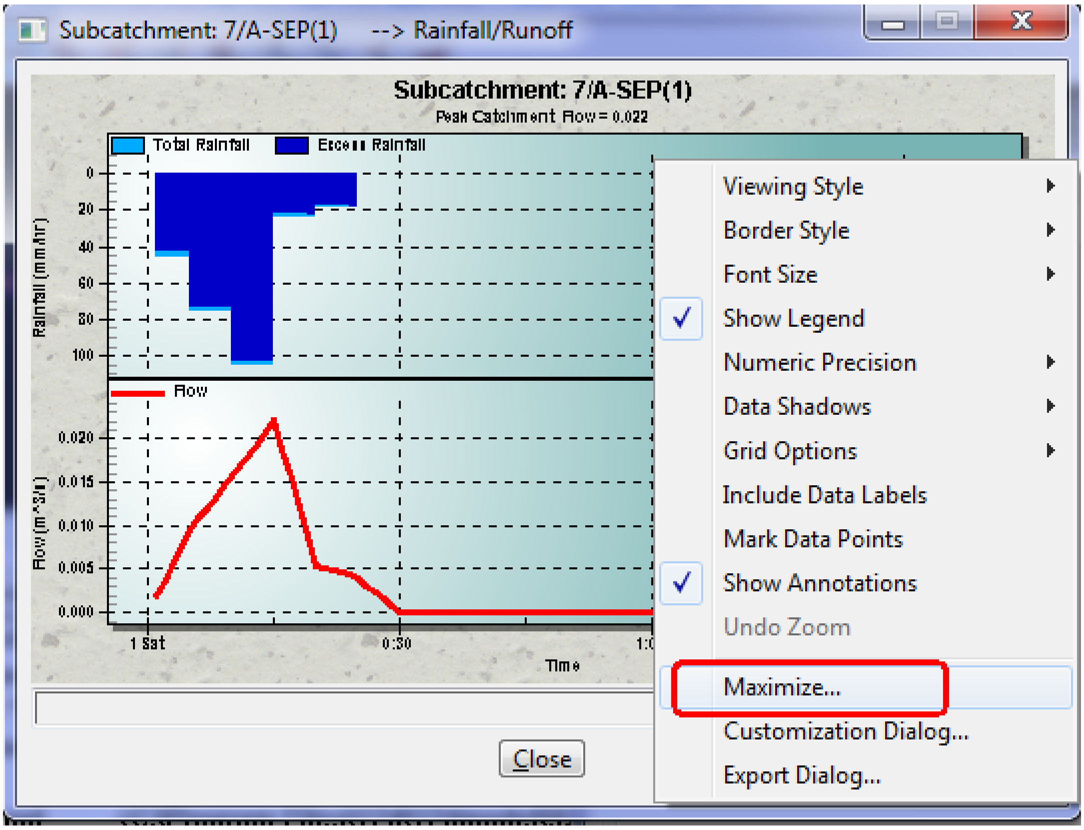
7.3 Reviewing Results - Graphs

Graphs indicating the results are created on the **Graphs** tab of the **DNE**. These graphs are a sampling of the time step results and usually does not include all time steps. Therefore, if the peak indicated in the attributes occurs between 2 plot points the peak may not be on the graph.

Select the **Storm event** to view. Nothing is displayed yet. When **All** is selected, the results from all the storms analysed in the last run are displayed. Graphs types that display more than one line will not be displayed when all is selected. There will be far too many lines on the plot to properly identify them.



Now select the data to view using the drop down boxes or by selecting the below and pressing enter. For example the **Rainfall/Runoff**.



The panel cannot be resized but it can be maximised via the right mouse menu. ESC returns to smaller view. The peak from the plot data (not the attribute) is indicated at the top of the graph.

You can zoom into the graph areas by dragging a rectangle inside the graph area. You must right mouse select and choose **Undo Zoom** to return to the full plot.

The available graphs are:

Catchment graph types (used in conjunction with **Sub catchment** selection)

Rainfall/runoff

The **Total rainfall** is the rainfall intensity calculated from the IFD data, the return period and the temporal patterns.

Excess rainfall (ILSAX 2) is the water that is left over after the rainfall losses infiltrate into the soil.

For the impervious area, excess rainfall = total rainfall - storage

For the pervious area, excess rainfall = total rainfall - storage - infiltration

Runoff starts when the excess rainfall depth exceeds the storage depth and the intensity is greater than the infiltration. Most frequently runoff starts from the impervious area first, (no infiltration, small storage and short tc) and then from the pervious area.

Runoff The runoff component of the **Rainfall/runoff** graph (good for all storms).

Rainfall The rainfall component of the **Rainfall/runoff** graph (good for all storms).

Losses These are the horton infiltration losses that are subtracted from the pervious component of the catchment (good for all storms).

Pit graph types

12d analysis has 2 systems for conveying flow. The **surface system** created with bypass flow strings (open channels with their inverts at the pit grate levels) and the **link system** created with the 12d drainage strings.

Grate & Invert Depths:
The top is the **Grate depth** showing the depth of flow in the surface channel measured from the grate level. The bottom is the **Invert depth** showing the depth in the link system above the pit invert.

Grate & Invert Elevations:
Shows both the bypass and link system water elevations. Good to see when the link elevation reaches the grate level to stop the inflow of water through the grate.

All Inflows & Outflows: (Inflows are + and outflows are - (S means surface)
Flows for each link and bypass are listed separately.
Local is the total **Catchment** flow from all 3 catchment sets plus direct flow,
In: Links and surfaces flows entering the pit.
Out: Links and surfaces flows leaving the pit.

Total Inflows and Outflows:
Local: the total **Catchment** flow from all 3 catchment sets plus direct flow,
Inflow to Invert: the sum of **In:** (links) + **Captured flow** This is the total inflow into the pit for the link system.
Outflow from Invert: the sum of **Out:** (links) This is the total outflow from the pit for the link system.
Approach Flow: the sum of the **Local** + **In:** (surface)
Captured Flow: determined from the **Approach Flow** (on grade pits) or **Grate Depth** (sag pits)
Bypass Flow: Approach flow - Captured Flow

Depth above Invert see invert depth above

Depth above Grate see grate depth above

Elevation - see Invert elevation above

Elevation Bypass Flow - see Grate elevation above

Basin volume - only available on pits with basin curve data.

Catchment Flow - Total of **Runoff** for Sets 1, 2 and 3.

Link Inflow and Captured Inflow: surface and link pit elevations, **Link Inflow** (see **Inflow to invert** above) and **Captured Inflow** (see **Captured Flow** above)

Link Outflow: see **Outflow from Invert** above

Approach Flow: see above

Bypass flow: see above

Inflow: see **Link Inflow** above

Overflow - the flow that is lost from the drainage system when the hgl exceeds the maximum level at the pit.
The maximum level at the pit is largest of
1) the cover level,

2) the grate level + depth of bypass channel
3) the maximum storage level in the basin storage data.

Ku - the link flow, upstream velocity, Ku and head loss of the primary outgoing link.

Pipe graph types (link system)

Dynamic Section: a section view of the entire string showing the hgl in the links and surface channels. The lower dashed line connect the grate levels and the upper shows the top of the surface channels. If the hgl goes above the top of the surface channel water will be lost from the system.

All Link Results 3 charts: The colour of the right axis label is the same as the graph line it represents.
HGL US, HGL DS (left axis)
Depth (right axis) depth at the mid point of the link
Capacity (left axis) the ratio of the current pipe flow / (pipe full, HGL at grade capacity)
Froude Number (right axis) - Froude number at the mid point of the link
Flow (left axis) flow in the link
Velocity (right axis) - velocity at the mid point of the link

Bypass graph types (bypass system)

same results graphs as for the **pipe graph types** above

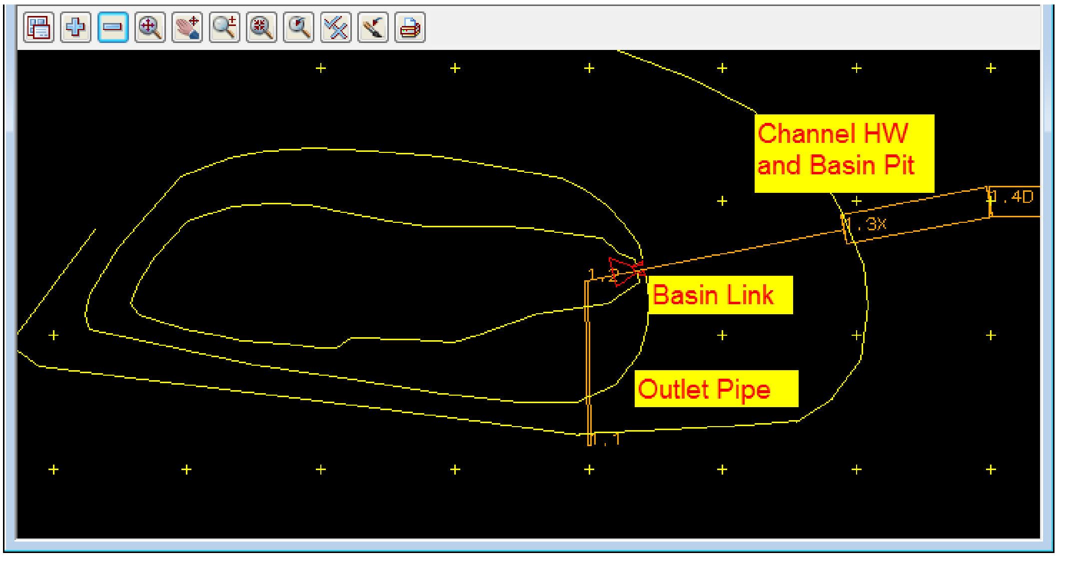
8.0 Adding Storage Basins

Storage basins are created by adding an elevation versus area curve at a pit. Without the area curve the default area of a pit is defined by the diameter or the length, width settings in the drainage.4d file.

The elevation area curve may be entered manually for initial estimates or measured from the finished surface tin (inside a string defining the extent of the basin).

The outlet structure is the link exiting the basin pit. This is often a weir, orifice or discharge limiting pipe.

An optional basin link may be used to separate the outflow structure from the basin pit. This allows the user to place the outlet structures in their correct position. The basin link is only a graphical link and often has its diameter set to zero to help indicate that it is not a hydraulic component of the system.

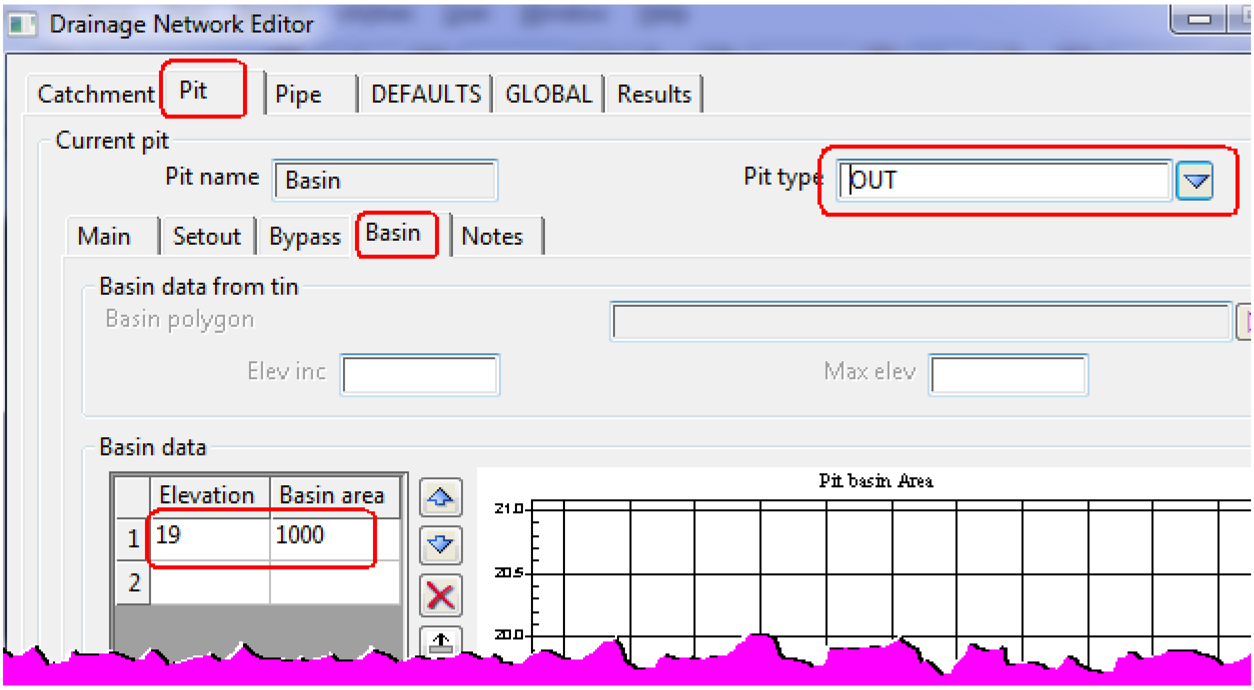


8.1 Selecting the Basin Location

The node modelling the basin is the outlet of the pipe or channel discharging into the basin. The pit becomes a basin when the user enters elevation versus area curve for the basin or selects a polygon around the top of the basin. With the polygon method, 12d will create the elevation versus area curve from the finished surface tin. We will use an existing string from the survey data but you may easily create your own polygon around the top of the basin.

For an initial sizing you only need to enter the basin invert elevation and area into the grid.

If the outlet pipe has an upstream invert level below the lowest basin curve entry, this lower section of the pit will use the default pit area.



8.2 The Outlet Structure and the Basin Link

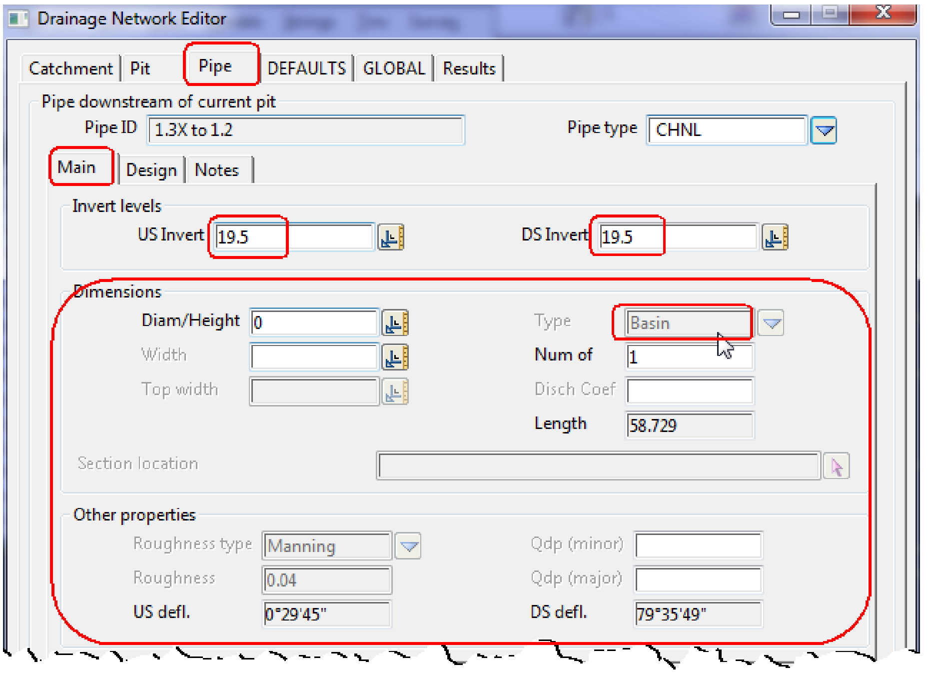
The basin outlet generally includes more than one structure. Typical arrangements include a low flow pipe that restricts the outflow from the basin and a high flow spillway used in larger events.

Every outlet structure is linked to the basin pit with a basin link. Often the outlet structure is a fair distance from the basin pit and we want the outlet structures drawn in their correct location. IF the diameter of the basin link is set to zero it joins the basin pit to the outlet structure but preforms no hydraulic function in the model.

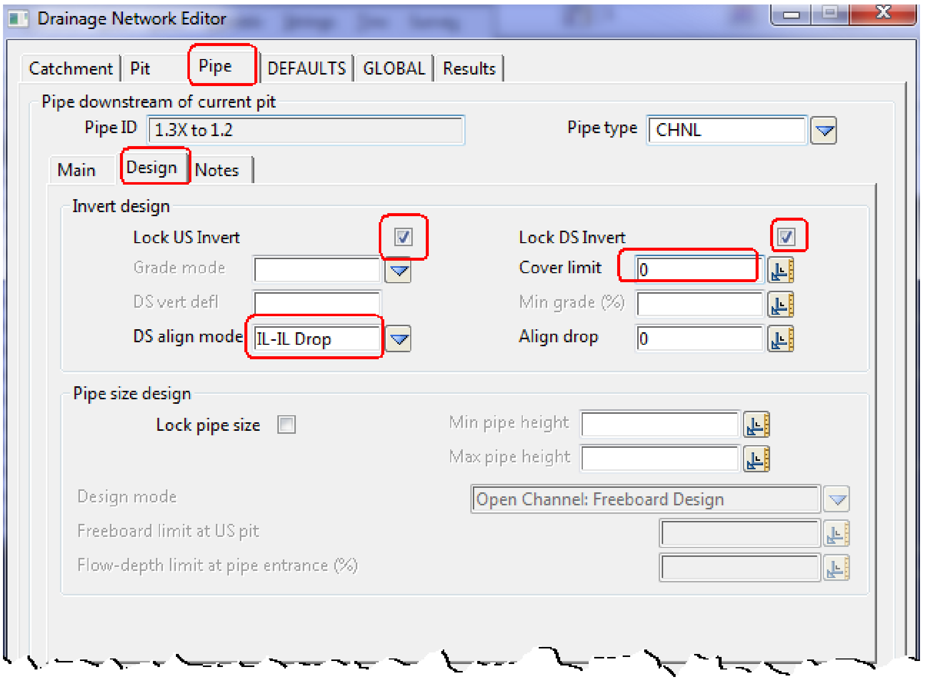
The minimum elevation in the elevation data becomes bottom of the basin and the outlet inverts cannot be below this level. If there is a drop pit for the outlet then ensure that the base of the drop pit is first elevation in the basin data. The area will be the area of the pit. The second entry will be the top of pit level with the same pit area.

8.3 Creating the Basin link

The pipe downstream of a basin pit is a basin link. When the diameter is set to zero, the data set on this link is for graphical presentation and has no hydraulic function.

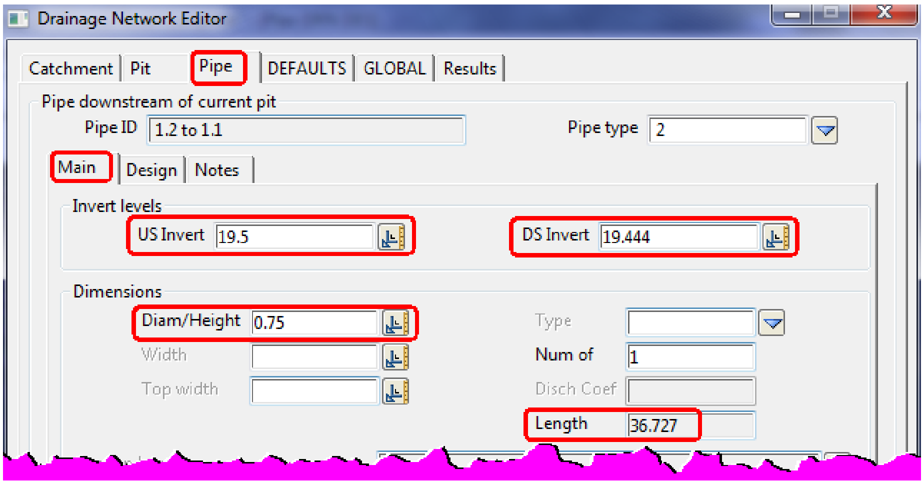


For clarity in the section view we will set the levels to the bottom elevation of the basin and set a zero pipe diameter to make the link as unobtrusive as possible.



Locking the link inverts will ensure the invert elevations are not changed with a regrade pipe selection. Remember this is a graphical link and the values do not affect the hydraulic calculations.

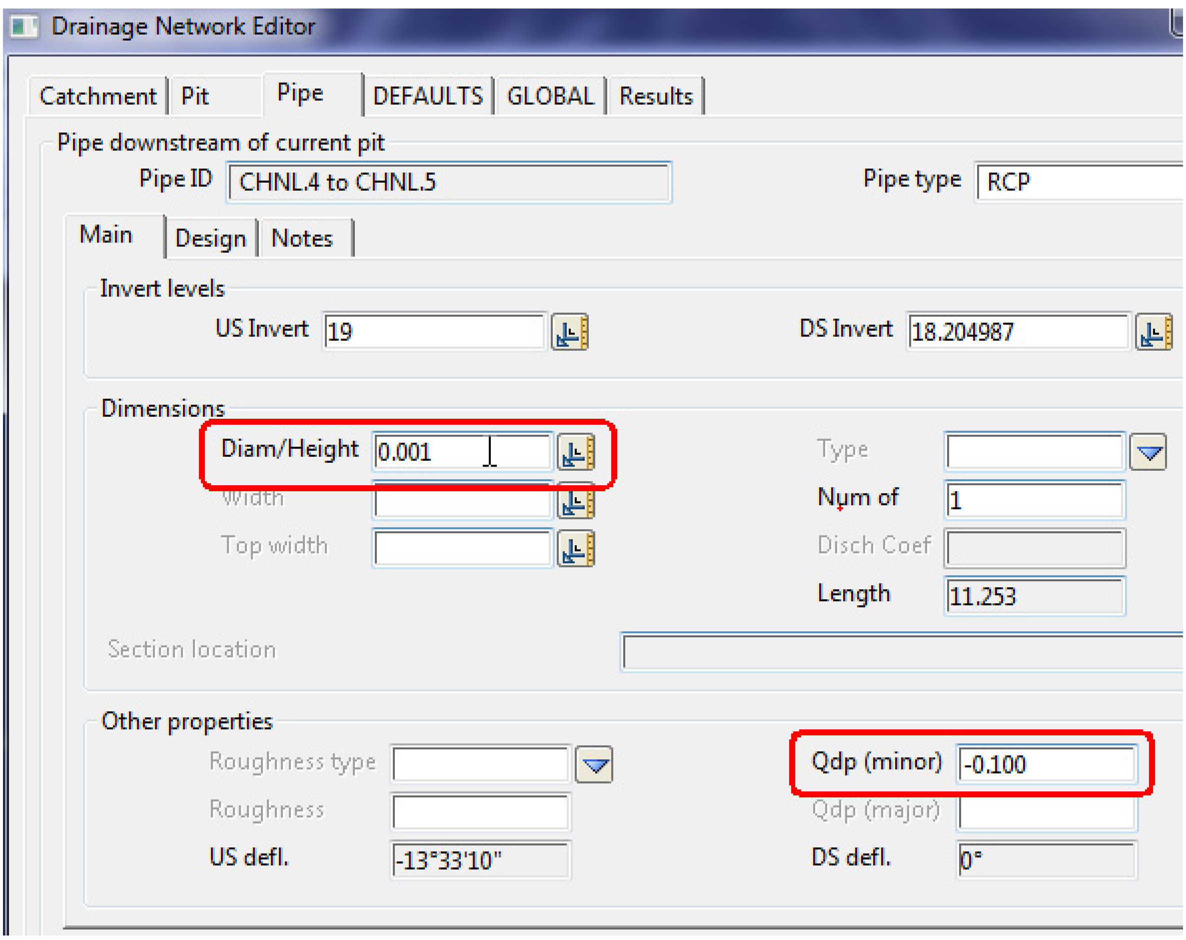
8.4 Data for the Outlet Control Pipe



The upstream invert is being set to the bottom of the basin and then we will lock this invert. We will leave the engine to set the DS invert level.

The length will depend of your drawing.

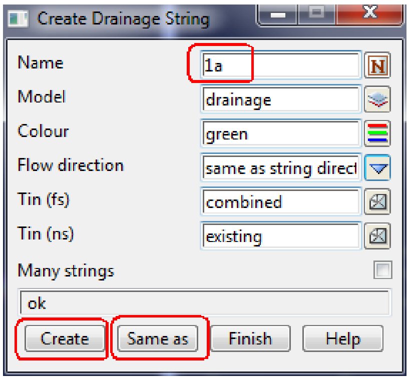
For initial sizing of the basin you can set the diameter of the pipe to 0.001m so that no flow goes through it. Now enter a negative pipe direct flow and this will become the constant outflow from the basin (usually set to your pre-development peak discharge). Once you size the basin, return here, remove the negative flow and select a pipe size to yield similar results.



8.5 Adding a Second Outlet Structure (Spillway Weir)

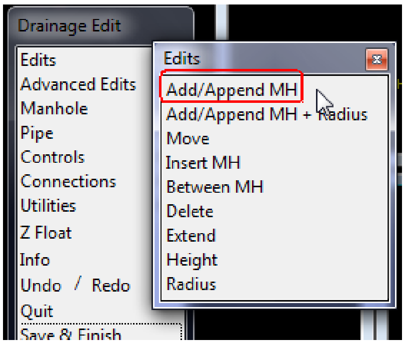
A second drainage string will be needed for the spillway. Again it will start with a basin link and then have a weir section. From the main menu select

Design->Drainage-Sewer->Create

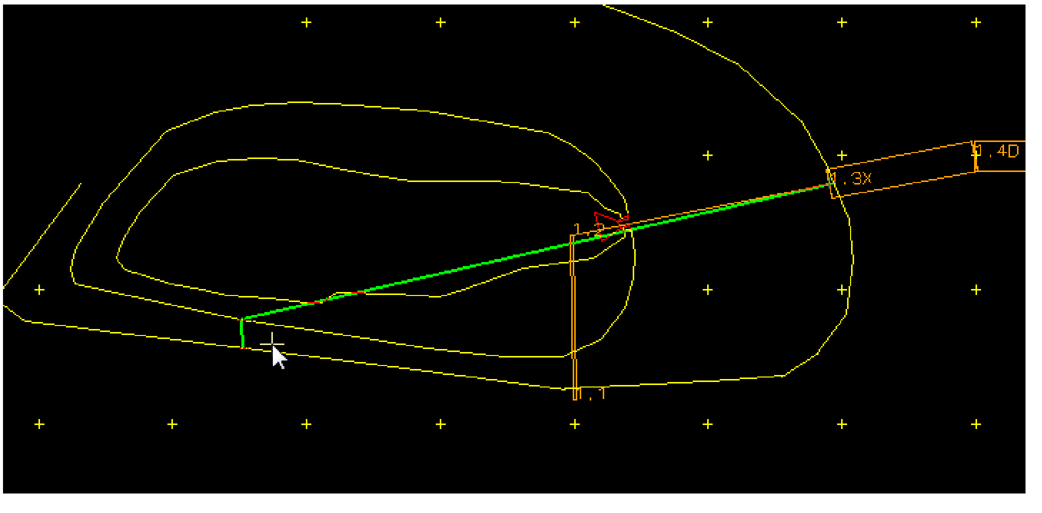


Select the Same as button and pick a drainage string in the model. This will fill in all of fields. Name must be unique so change this data. Finally select Create.

Then select Add/Append MH from the menu.

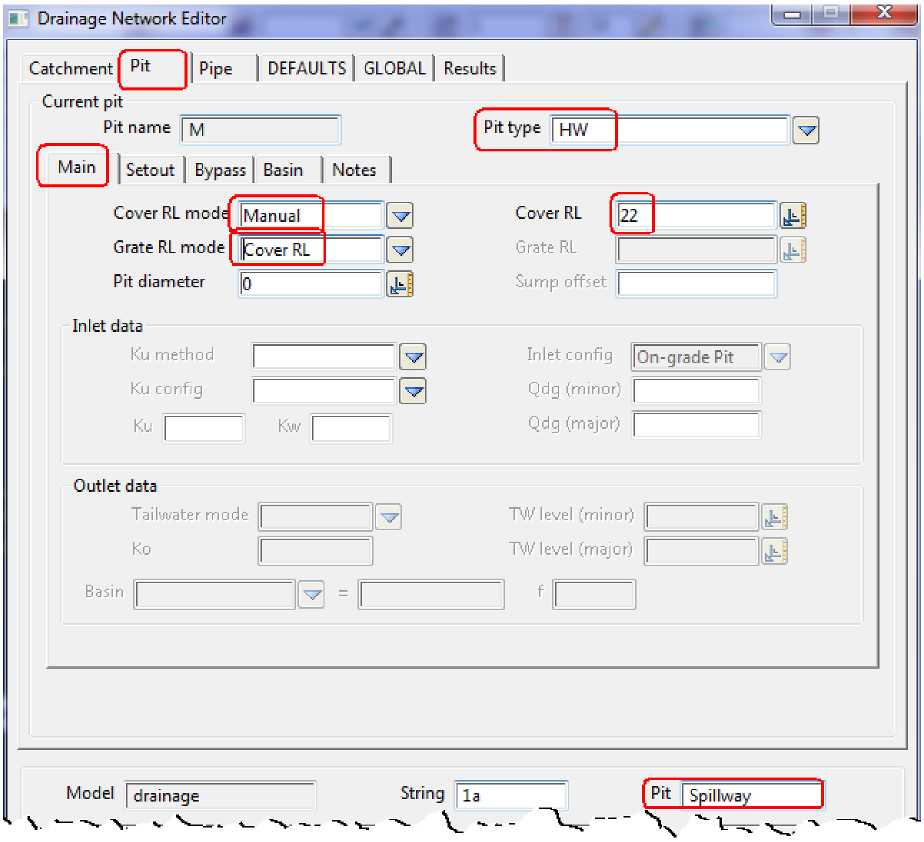


The plan view below shows the basin link and the spillway weir location. Draw the string in the



direction of flow and ensure you start with a point snap on end of the channel.

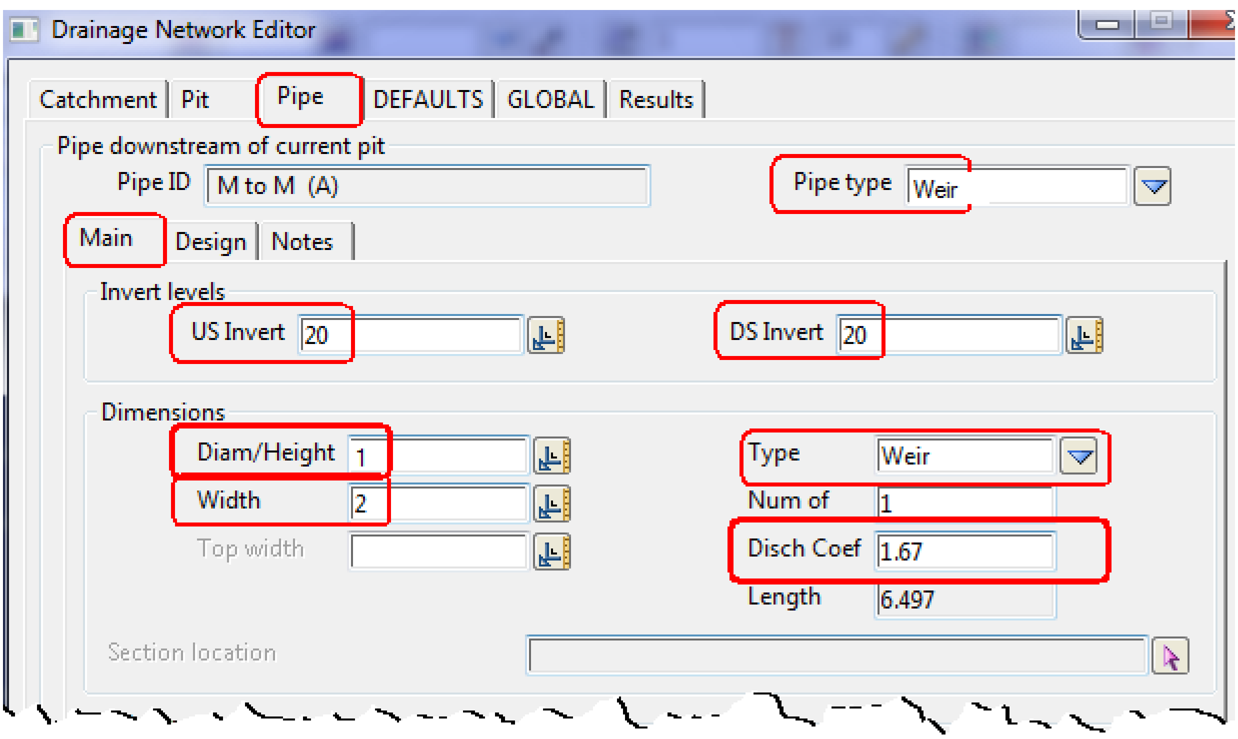
Once the string has been created the pit data at the upstream side of the weir need to be set.



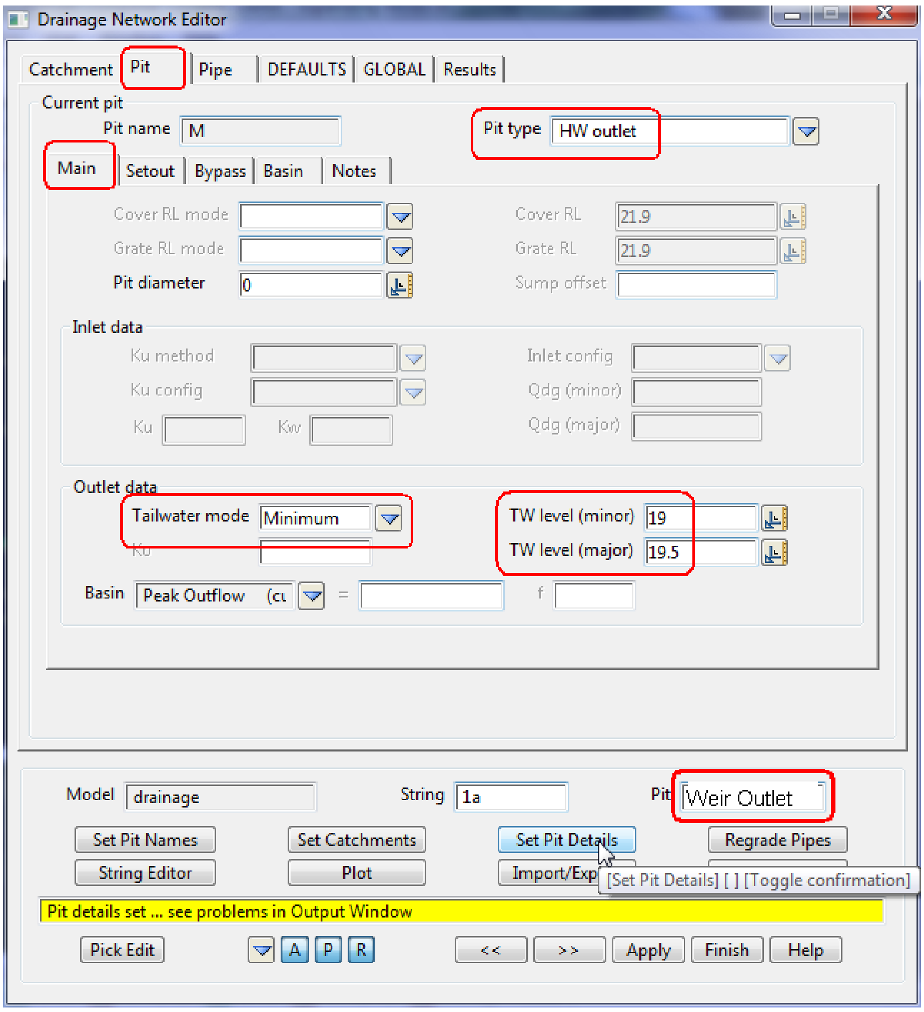
On the Pit->Main tab we are going to set the cover level at the highest expected water level in the basin. The grate level will have the same level as the cover level.

The **Pit type** does not affect the calculations for the weir.

We have also assigned a name to this pit.



On the pipe-main tab we will see the **US invert** level to weir crest level (20.0). The **DS Invert** is set the same but will have no affect on the calculations. The **Diam/Height** determines when the weir will start acting as an orifice. Generally used in underground storage tanks. In this case we will set is to a value higher than the expect flow depth (1.0 m). The **Width** is the dimension of the weir perpendicular to the flow. The Disch Coef is optional as the default will be used if it is blank.



The tailwater levels are now set to the expected levels for the minor and major events.

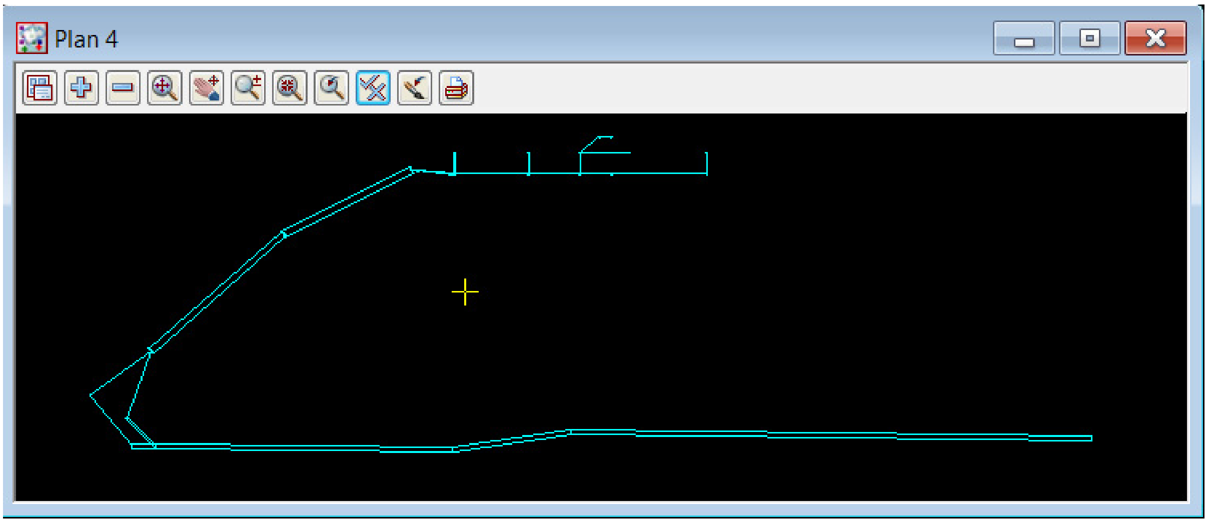
Enter a name for the outlet.

9.0 Natural Section Shapes

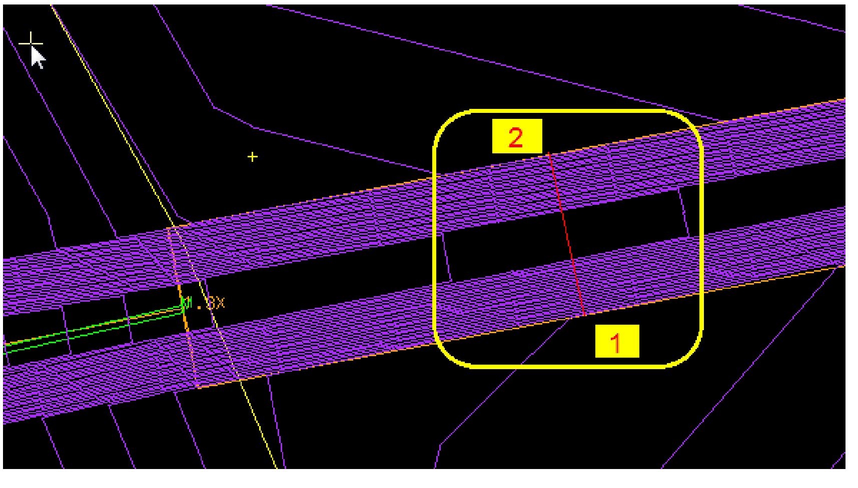
The channel shape may be cut from the design tin by drawing a string at the desired location. It is a standard convention to draw it from left to right looking in the direction of flow.

In this example we are going to connect the end of the culvert to a channel down to the spillway and then eastward to the end of the project. Set the Defaults, **Pit Type** to **chnl auto** and **Pipe Type** to **CHNL GRASS EXISTING 1** with a **height** of 4.0. This height will not affect the calculations but will show the approximate channel top in the section view. This will mean you no not have to change them later.

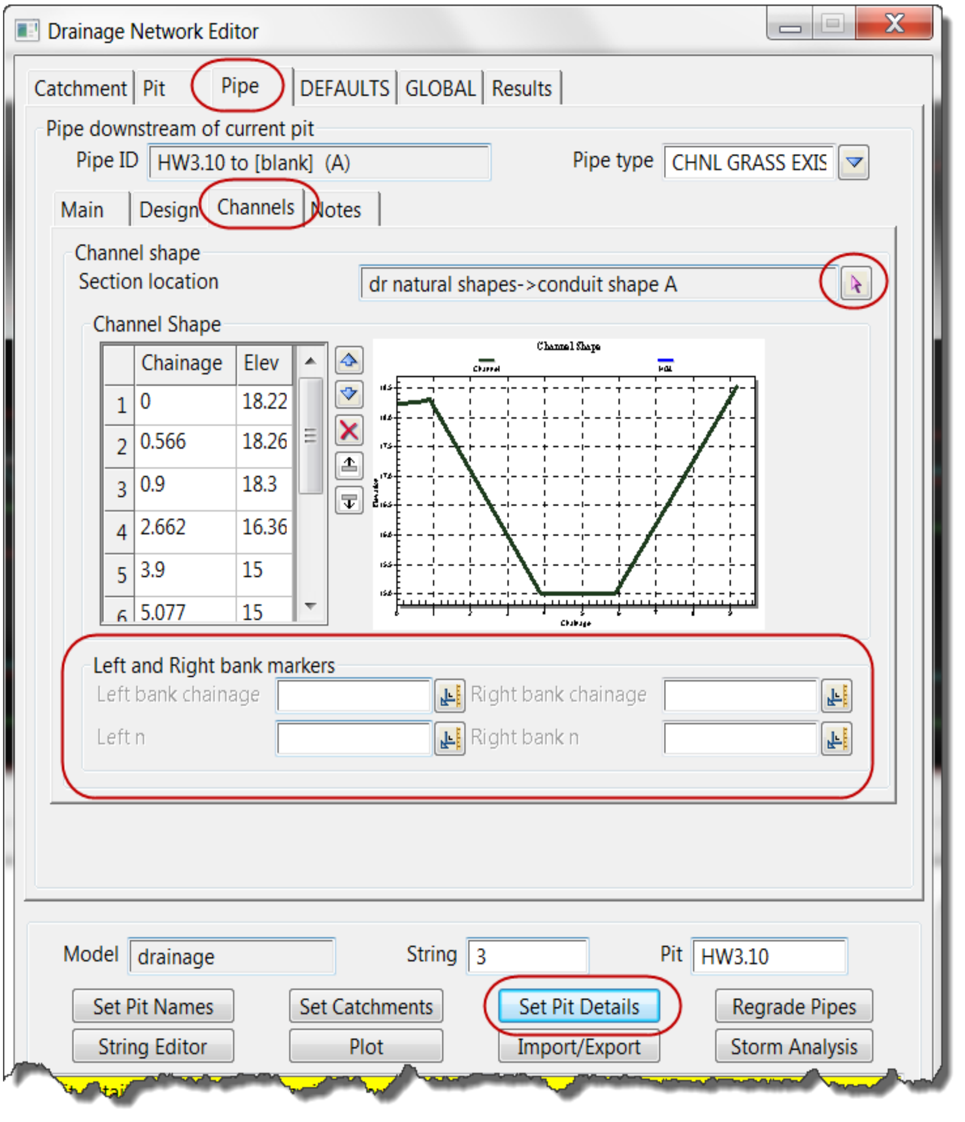
Use the **Points Edit->Append** to add the extra links as shown below. Ensure that the centre of the drainage string is in the lowest part of the channel. Finally, extent the spillway to the channel.



Enter the model name **dr natural shapes** into the CAD toolbar model field before drawing this string.



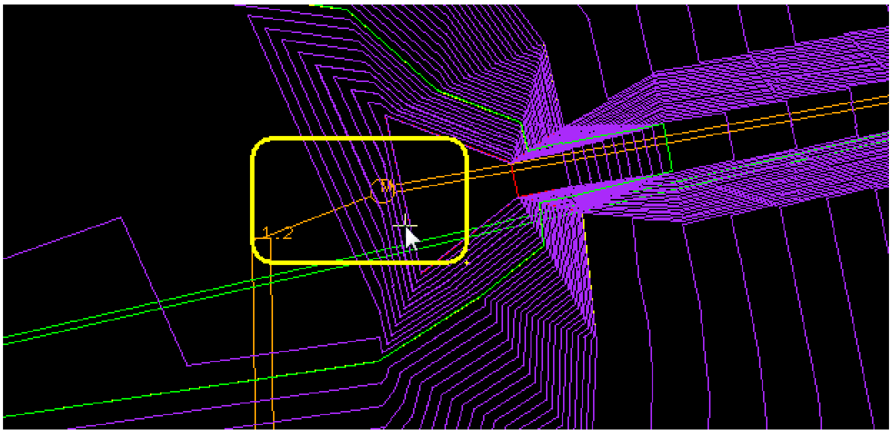
Once you have drawn the string you can select it from the **Pipe->Channels** tab. Selecting **Set Pit Details** will have the shape cut from the design tin and displayed in the grid and graph area. If you do not want the shape cut from the tin you can just enter the shape in the grid area. Once again the shape obtains its elevation by setting the lowest point onto the invert elevations.

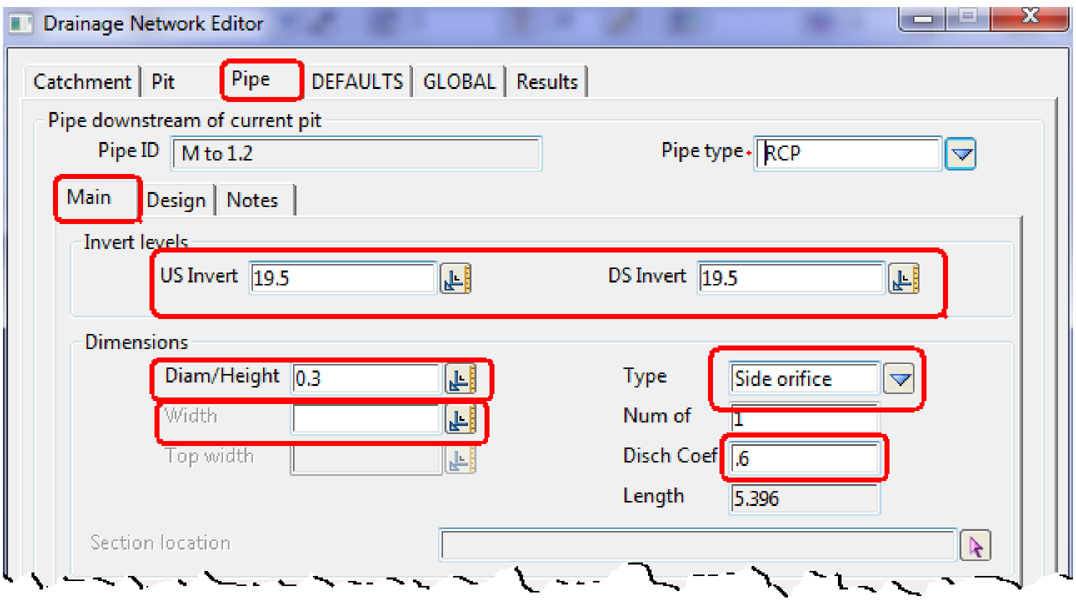


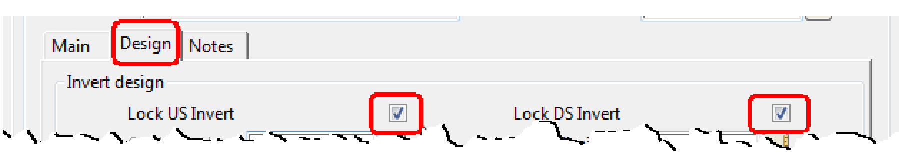
Left and **right bank chainages** and n values may be entered if desired. If blank the link's n value will be used. There is no default left and right bank n values.

10.0 Orifice Control on a Pipe

Use the Strings->Points Edit->Insert command to add a pit upstream of the outlet pipe. This new pipe will be set to an orifice to control the flow entering the pipe.







Lock the invert levels so that the orifice will not move in elevation,

11.0 Storage Tanks as Basin or Pipe

A storage tank may be modelled as a box culvert or as a basin but do not do both as this will double count the storage. To model the tank using basin areas Set conduit type to a Basin link and set the pipe diameter to a value that is meaningful for the long section appearance. Once it is marker as a basin link it will not affect the calculations.

11.0.1 Drainage Utility Program

Position of option on menu: Design => Drainage-Sewer =>More => Misc. utilities

The Drainage utility program contains functions to significantly reduce the time required to perform drainage tasks. These tasks include

Assign Pit names

To use the export routines, every pit in 12d must have a pit name. This selection automatically creates the pit names for the entire model or selected strings. Examples of pit names are 1,2,3... A1,A2,A3....Pit 3-A, Pit 3-B.

Reset pit cover levels

This selection sets the cover levels for the manholes to the design tin or design strings. The user will be prompted for each manhole to select a tin level, a string level (if a design string model is supplied) or keep a manually set level. The tin/string/manual selection will be stored and the levels reset now and whenever the pit/pipe interfaces exports the data (unless this last option has been manually turned off).

Regrade pipe levels

The selection applies the default grading rules (cover level) to reset the pipe invert levels for the entire network. Manhole cover levels are not changed during this function

Label Catchments and Label drainage network

This selection quickly creates labels for a drainage and catchment plan. The pits are labelled with their name, the pipes with their diameters and the catchments with their area and the pit they drain to. These labels must be updated using this selection whenever the catchment or network is changed. To turn off the automatically drawn pit names in the current view select **Menu=>Settings=>Text=>Toggle** and select the drainage model.

Analyse Flooded Width

This is the only hydraulic calculation that takes place in 12d. The normal depth along the bypass flow paths is calculated using discharges imported from hydrology/hydraulic packages including spreadsheets. This flooded width is drawn to scale at intervals along the bypass flow path and colour coded (blue if less than a specified limit and red if greater than the limit).

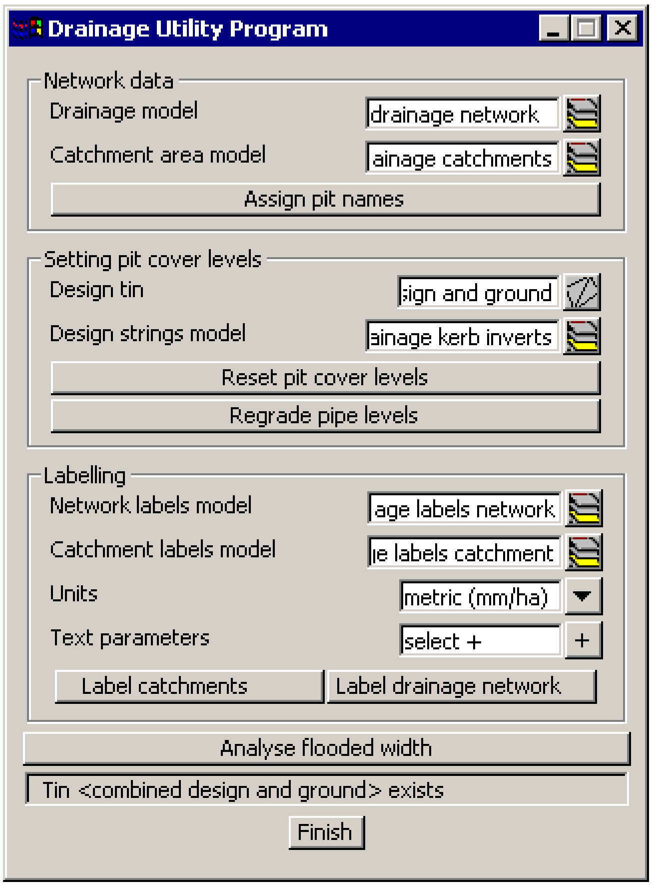
See Also

Drainage overview

Usage

This panel is accessed from the menu selection

Design => Drainage-Sewer => More => Misc. utilities



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up

drainage model input box

Model to contain all of the pit and pipe network to be worked on.

catchment area model input box

Model to contain the catchments strings for the ***Drainage Model*** *above*

Assign pit names button

This button launches a dialogue for setting the names for the pit and pipes in the network model.

design tin tin box

This optional field allows the user to specify a new tin for the surface levels of the pits. If you are using road grade and cross fall for pit inlet capacity it is preferable leave this blank and to link the pits to a string using the ***design model*** *field below. If some or all of the pits are linked to strings in the design model below, they will still use the strings selected.* more about design strings

design model model box

This model contains the strings for the pit cover levels (the string directions are also used for road grade and cross fall). If this model is changed you will be prompted to select new strings to link the pits to. more about design strings

Reset pit cover levels button

*This button will reset the pit levels to the design strings in **design model** or to the tin specified in **design tin** above. The first time this is selected you will be prompted to choose whether to set the pit cover level to the design tin or the a string in the design model.* more about design strings

Regrade pipe levels button

The selection applies the default grading rules (cover level) to reset the pipe invert levels.

network labels model model box drainage labels network

*Model to contain the network labels for the **Drainage Model** above. This model is cleaned out each time **Label drainage network** is selected.*

catchment labels model model box drainage labels catchment

*Model to contain the catchment labels for the **Catchment Area Model** above. This model is cleaned out each time **Label catchments** is selected*

catchment units choice box ha, acres

Conversion factors of 10,000 will be used for ha and 43560 for acres.Not that not all design packages support both units.

text parameters input box

*Select the + to access the text parameters (colour, size, alignment etc.) for the text created with **Label Catchment** or **Label drainage network***

label catchments button

This selection creates labels indicating the catchment name and area in the units specified above.

***CAUTION:** If you change you catchment strings or rename you pits you must run this routine to update the labels.*

label drainage network button

This selection creates labels indicating the pipe size and pit name for the network model.

***CAUTION:** If you change you change the pipe sizes or rename you pits you must run this routine to update the labels.*

analyse flooded width button

This selection launches the dialogue for analysing the flood along bypass flow paths. The bypass flows must be imported from your design package/spreadsheet before running this selection.

11.0.1.1 Set Pit Names

The **Set pit name** selection names the pits and pipes for the drainage model (both pits and pipes can be named). Three numbering schemes are available:

drainage strings names are used as the prefix followed by the pit number (ex A-1)

drainage strings names are used as the suffix followed by the pit number (ex 1/A)

the pits are sequentially numbered. (ex 1,2,3....)

The names can have pre-text added to the beginning of the pit name and a separator between the string name and the pit number (/- etc.) if desired.

Design Program Notes:

PCdrain users: Since catchments contain only 3 characters do not use separators. Using the letters A-Z for strings and numbers 1-99 will give you 26 strings and up to 99 pits on each string. Using the numbered stem works very well in PC Drain.

Micro Drainage users: Pits and pipes are numbered separately in Micro drainage. The pipes must use the numbered sequence with the most upstream pipes numbered with the smallest numbers. Number of digits must be set to 3.

ILSAX users: You must use alphabetic characters for your string names and no more than 3 characters

See Also

Labelling a drainage network

Displaying the Auto Pit Names

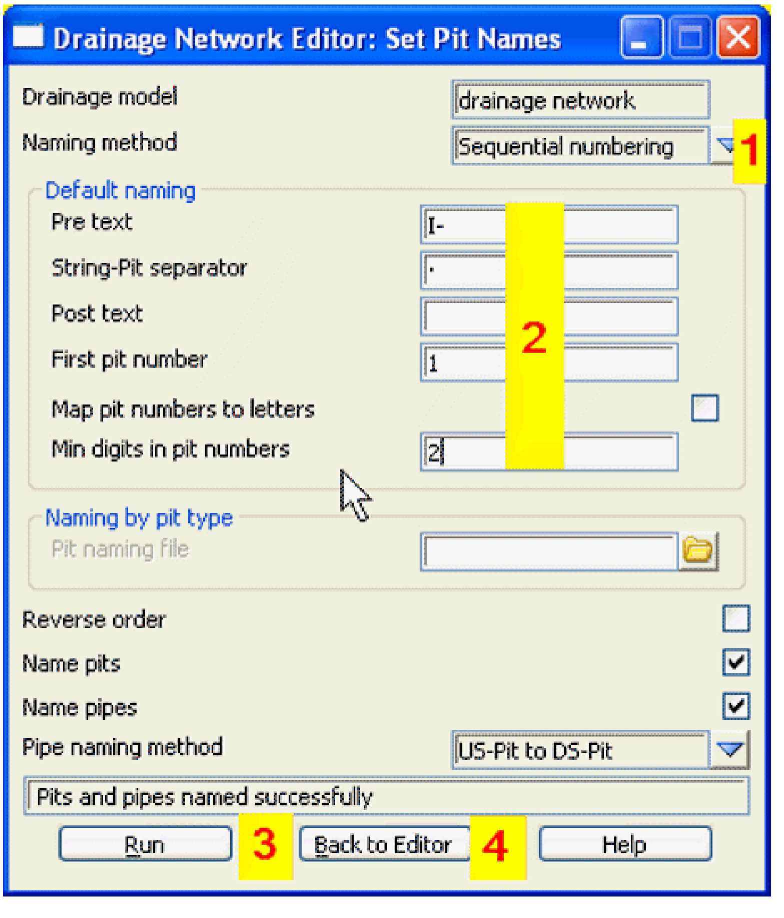
Drainage overview

Usage

The selection is found on the drainage **network editor** and is accessed through the main menu by selecting

Design=>Drainage-Sewer=>Drainage Network=>Drainage=>Network editor

Select the **Set Pit Names button** and the following panel will appear.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Model	Model box		**drainage model from main dialogue**
Naming Method	Choice box		

__Pit num - String name__ for string A pit 1 the name is 1A
__String name - Pit num__ for string A pit 1 the name is A1
__Sequential numbering__ the strings names are sorted alpha-numerically and the pits are numbered starting at __First pit number__

Pre text input box

this text will precede the pit name (ex Pit A01)

String/pit separator input box

the character that separates the string name from the pit number. For example if pit names A/01, A/02 etc. where desired a **"/"** *would be entered. You may omit the separator if none is desired.*

First Pit Number integer box

the starting pit number on the drainage lines. For drainage strings with the flow direction set as descending chainage the first pit is the junction pit.

Min digits in pit numbers integer box

a non zero value will pad the pit numbers with leading zeros. (ex if 2 is entered, pit 1 is not A/1 but rather A/01)

Reverse numbering order tick box

Unselected, pit __First pit number__ is at the low chainage end of the line. Selected, pit __First pit number__ is the second pit from the high chainage end of the line.

Number pits tick box

the pits are assigned names. This allows the pit and pipes to be names differently.

Number pipes tick box

the pipes are assigned names. This allows the pit and pipes to be names differently.

Run button

name the pits/pipes. The pit at the high chainage of the string is not labelled. The only time you will need to name this pit is when it is the outlet from the system.

Back to Editor button

return to the network editor.

Help button

display this page.

11.0.2 Drainage Input/Output Interface

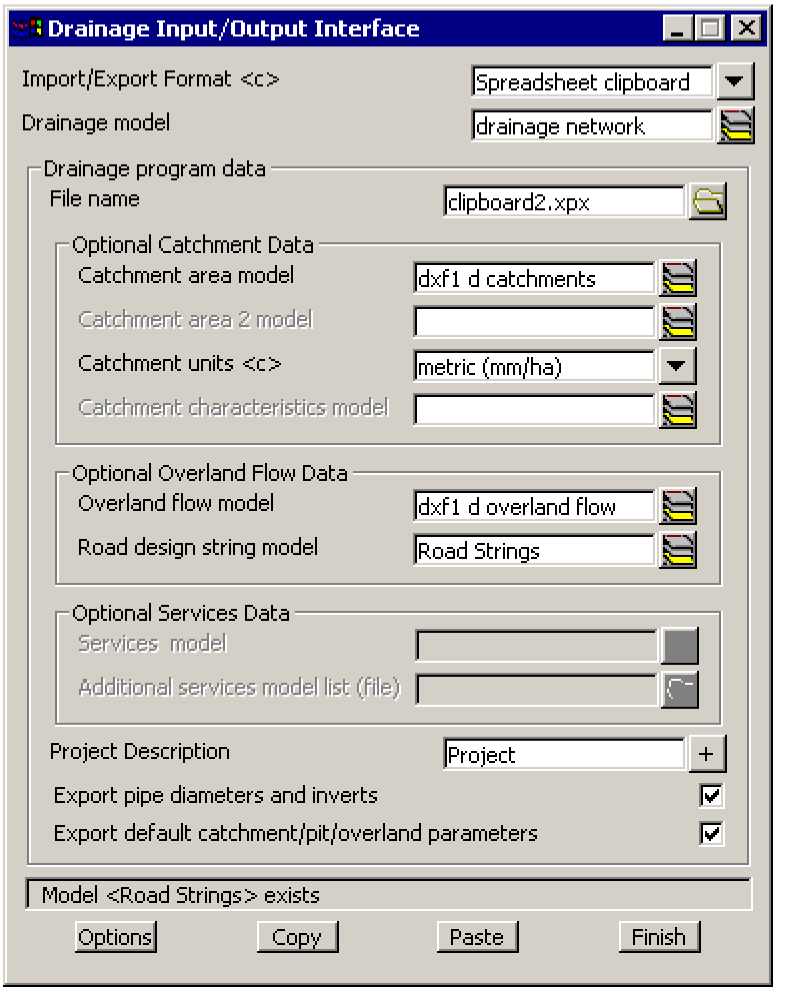
Position of option on menu: Design => Drainage => More=> pit/pipe design interface

See Also

Drainage overview
Drainage Misc Utilities
Spreadsheet clipboard
Running Drains
Running PCdrain (Windows)
Running Micro Drainage - Win DES
Running XPSWMM
Running RAT2000

Usage

This panel is accessed from the menu selection **Design => Drainage => More=>Version 6=>Pit/pipe design interface**



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
import/export format <c>	input box	XP SWMM	**Spreadsheet clipboard**,

Drains clipboard
PCdrain Int (windows)
RAT2000
XP SWMM
Micro Drainage
Drains
PC Drain (DOS)
RAT HGL
ILSAX

select the drainage design program to export/import to.

drainage model input box

***Required** Model to contain all of the pit and pipe network to be exported/imported. Also see* Drawing the Drainage Network

file name input box

***Required** file to be read or written to. If a clipboard format is chosen for* **import/export format** above then the data will also be written to this file on a **Copy** selection

catchment area model input box

***Optional** model to contain the catchments strings for the **Drainage Model** above. Also see* Designating Catchment Areas

catchment units <c> input box ha, acres

Conversion factors of 10,000 will be used for ha and 43560 for acres.Not that not all design packages support both units.

catchment characteristics modelinput box

***Optional** strings in this model will be used to define the catchment slope and width (XP SWMM only)*

bypass flow model input box

***Optional** model to contain the bypass flow strings for the **Drainage Model** above. Also see* Creating Bypass Flow Strings

road design string model input box

***Required** if* **bypass flow model** is specified above. Also see Pit Inlet Capacity, road grade/crossfall and Bypass routes

services model input box

***Optional** if the strings in this model cross the drainage network the crossing data (drainage chainage, invert elevation and thickness) will be sent to the design package*

additional services model list (file)input box

***Optional** if your services lie in more than one model then enter a text file name here and then select edit from the fields file icons. Type the names of all of the service models and then save the file.*

project description input box

***Optional** this description will be sent to the design program*

export pipe diameters and invertstick box

Select this tick box to export the pipe diameters and inverts. see also Quick Check Lists for Drainage Design Software

export default catchment/pit parameterstick box tick

selected all default catchment parameters are exported
not selected only catchment area is exported if **catchment area model** *is specified above. see also* Quick Check Lists for Drainage Design Software

Options button

several calculations are preformed before all exports. Advanced users may turn some off for large models (100's of pits). The option to re-link your pits to new design strings is also included here. More options

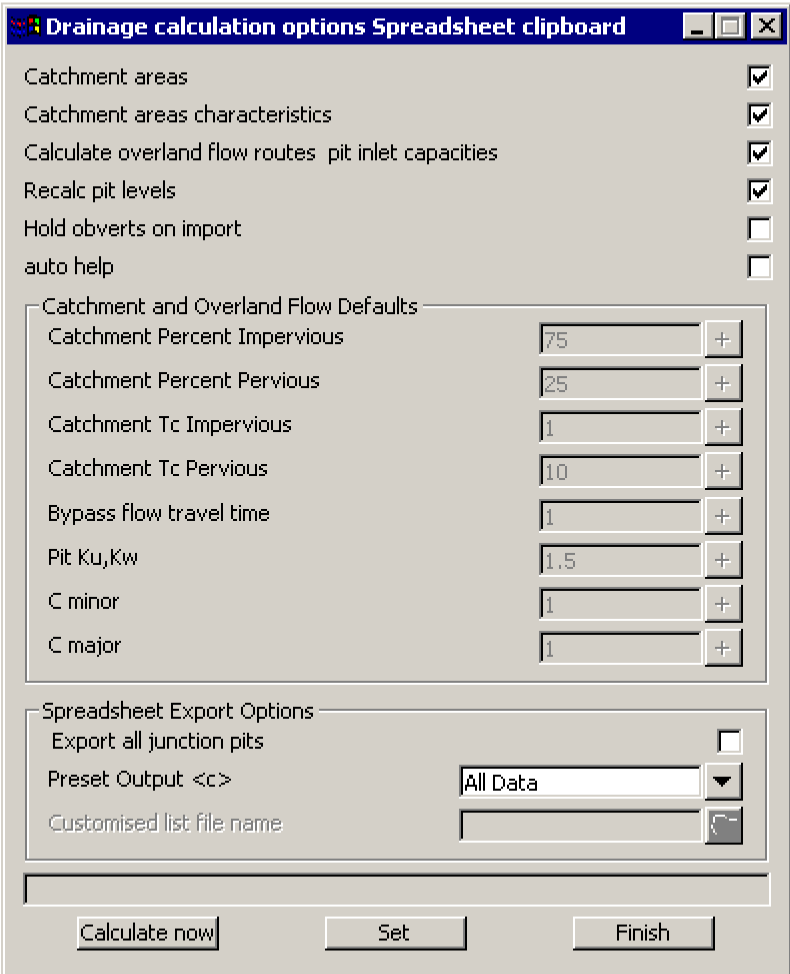
Write/Copy button

This will create/over write the file specified above in **file name**. *If* ***Copy*** *button is present the data will also be placed on the windows clipboard as Tab delimited text.*

Read/Paste button

Read *will read the file specified above in* **file name**. ***Paste*** *read the data from the windows clipboard. Both selections will update the drainage strings in the model specified above in* **drainage model**.*If the* ***Spreadsheet clipboard*** *import is selected and the strings are not present in the model they will be created.*

11.0.2.1 Options



Selecting the Options button brings up the dialogue to the left. Additional choices may be present at the bottom of the dialogue depending on the **Import/ Export Format** that you selected on the main dialogue.:

Catchment areas when selected the catchment areas are linked to the drainage pits and the areas recalculated. See also Designating Catchment Areas

Re-link pits to road strings-tin when selected a dialogue for each pit will be presented asking which design string or tin to link the pit to. See also Selecting design string or tin

Calculate bypass flow routes when selected will calculate the downstream bypass pit, road grade and crossfall and inlet capacities. See also Pit Inlet Capacity, road grade/ crossfall and Bypass routes

Calculate now will calculate the selected option immediately and return the user to the main dialogue.

Set will set the options but no calculations will be performed until a **Write** or **Copy** is selected from the main dialogue.

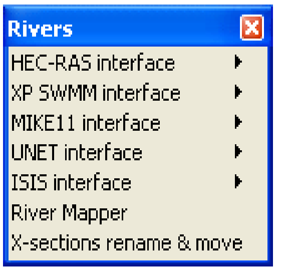
Finish removes the panel from the screen.

21 Rivers

Position of menu: **Design=>Rivers**

The **Rivers** option is used to prepare data for analysis packages and examine the results from the analysis.

The **Rivers** walk-right menu is:



See Also

Frequently Asked Questions (Rivers)

River Interface Models

HEC-RAS Interface

XP-SWMM Interface

XP SWMM Culverts

MIKE11 Interface

UNET Interface

ISIS Interface

River Mapper

Presenting Water Level Results

How to for Rivers

Rivers beta menu

21.1 River Interface Models

The River interface strings are split into different models depending on the strings function. Also the string names are used to define names of entities such as cross sections and reservoirs when they are exported. Following is a list of the river interface string types:

River strings

Source strings

Reservoir Strings

Spill Strings

21.1.1 River Strings

The river centre line and bank are defined by the strings in the **River strings** model.

The centre line string is used to

- measure the centre line distance between the sections,
- mark to zero chainage (or starting chainage) on the cross section, and
- if automatic source strings are created they will be perpendicular to this string
- **define culvert locations** and sizes (XP SWMM only)

The left and right bank strings are used to

- measure the bank distances between the sections and
- mark the cross section chainage where the conveyance (usually roughness) changes

The names of the strings must be **left bank, right bank** and **centre line *river name,reach name*** (centre may be spelled center).

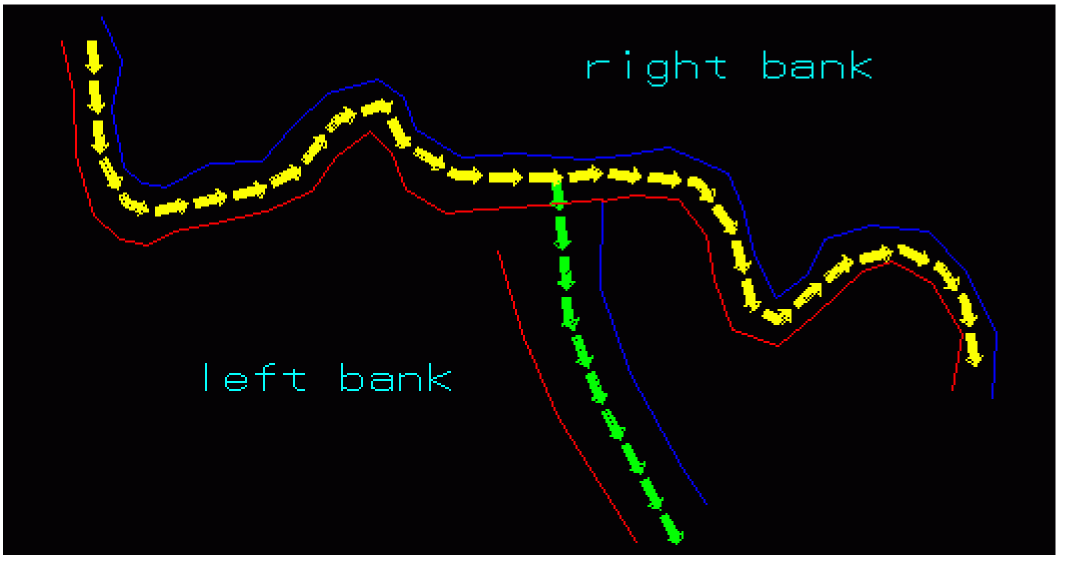
The left bank is on the left side of the river looking downstream and the right bank is on the right (looking down stream). These strings need not touch each other and may extend from one river reach to another.

HEC RAS and UNET

The centre line string must begin at the downstream end of the river. The modelling convention for these programs is to have the low chainage at the downstream end. You may use super, 2D,3D or alignment strings in this model. If you put other strings in this model you will receive warning messages saying that these strings will not be used.

Each reach of the river **MUST** have its own centre line string and they must "touch" each other to create a river confluence.

A sample of a river strings model for **HEC RAS** with one confluence is shown below. The line style for the centre line is not required. It is used only to show the direction of the centre line string.



Notes:

left bank strings are shown in red, right bank in blue and centre line strings in yellow and green. The line style for the centre line strings is *Drainage_4D->Flow line*. This is not required but shows the direction of the string. The string labels were created with Strings->Label->User-

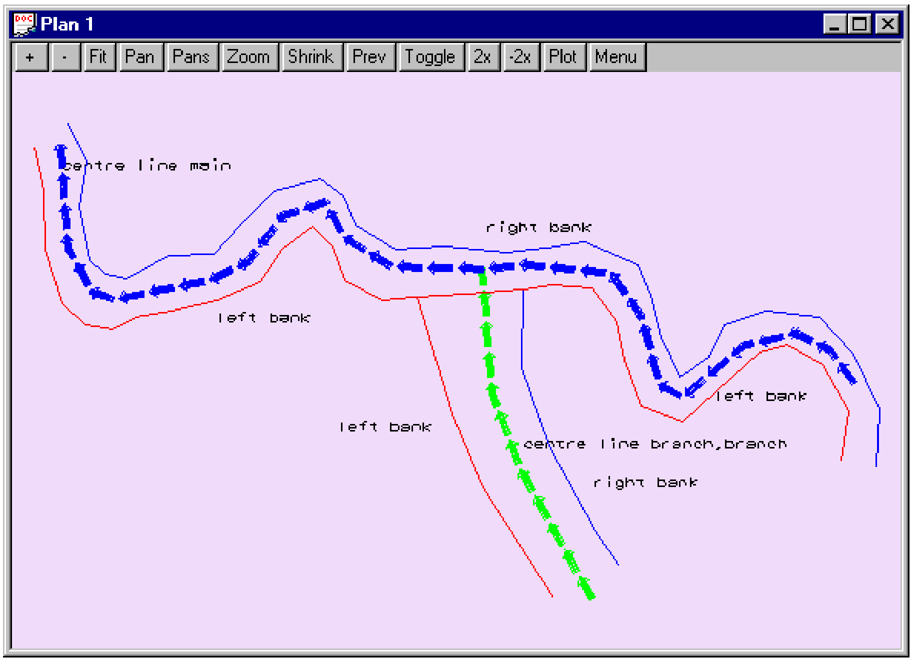
>Label strings with names.

Centre line string direction is very important!

HEC RAS, UNET, XP SWMM	start at the **DOWN**stream end of the river
ISIS and Mike 11	start at the **UP**stream end of the river

ISIS and MIKE11

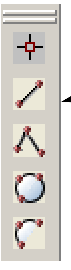
The centre line strings are drawn starting **upstream** for the ISIS and Mike 11 programs.



21.1.2 Source Strings

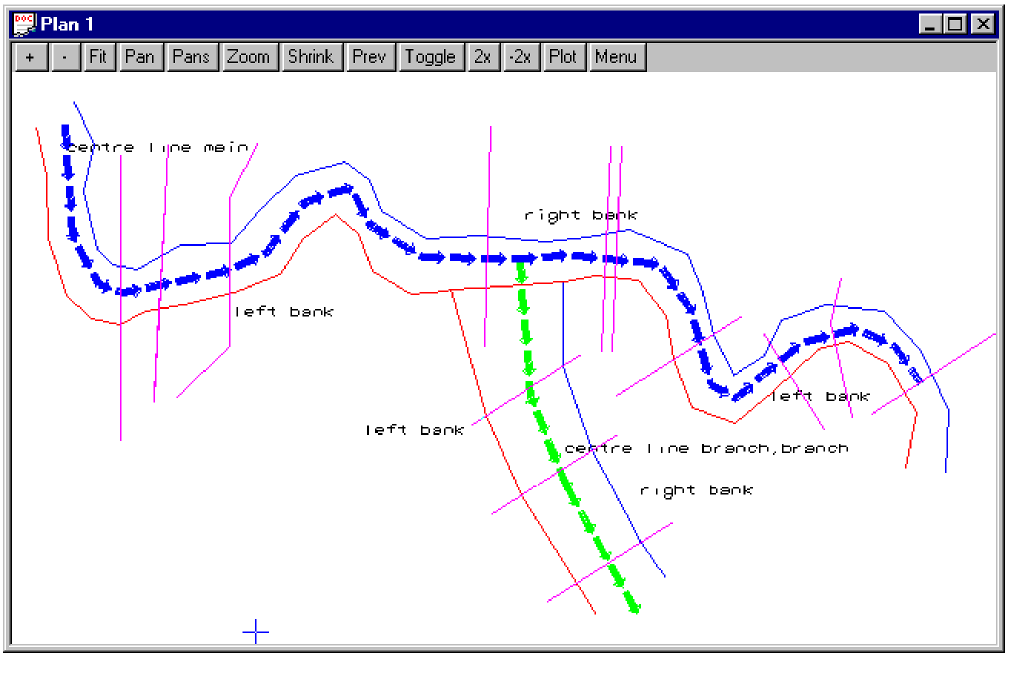
Cross sections are created at the location of the **source strings**. These source strings are initially created at a user defined spacing and section length using any one of the river interfaces writers. The user may alter these sections as desired. These may be shortened if they intersect at sharp bends in the river; they may be extended at extremely wide river sections or extra points may be added so that the section is no longer a straight line.

Source strings can be deleted and additional sections can be added by creating new source strings. The **Create source strings** tick box on the interface panel must **NOT** be selected to use the customised strings.



A quick way to manually create 2 point string is via the CAD tool bar.

A sample of source strings (shown in magenta) is presented below. The source strings may run in any direction except for Mike11. In Mike11, the cross sections will be created in the same direction as the source strings. For all other interfaces the low chainage will be used for the left bank (section viewed looking downstream).



21.1.3 Reservoir Strings

The **reservoir strings** may define inline reservoirs or offline storage. The elevation of the first point on the string sets the maximum level to be used in the stage storage curve. 12d will determine the minimum level inside the reservoir string and then calculate the volumes at a 1m (2 foot for imperial units) increment. The default increment may be changed by creating a string attribute "stage increment" (see **stage increment**).

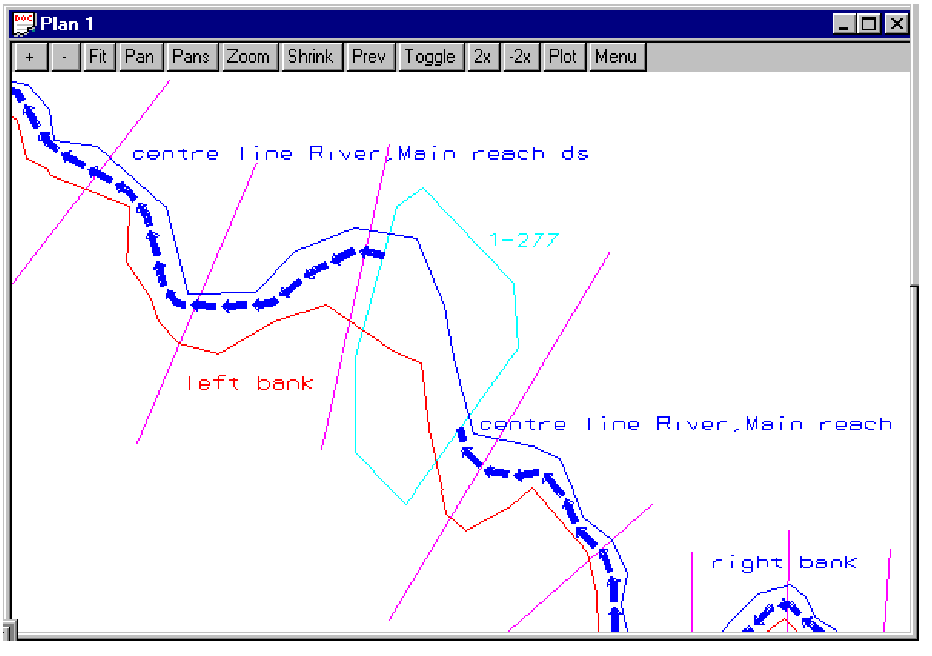
Inline reservoirs are "touched" by centre line strings both upstream and downstream. Offline storage areas are linked to the cross sections via **Spill strings.**

Other features of the reservoir strings are

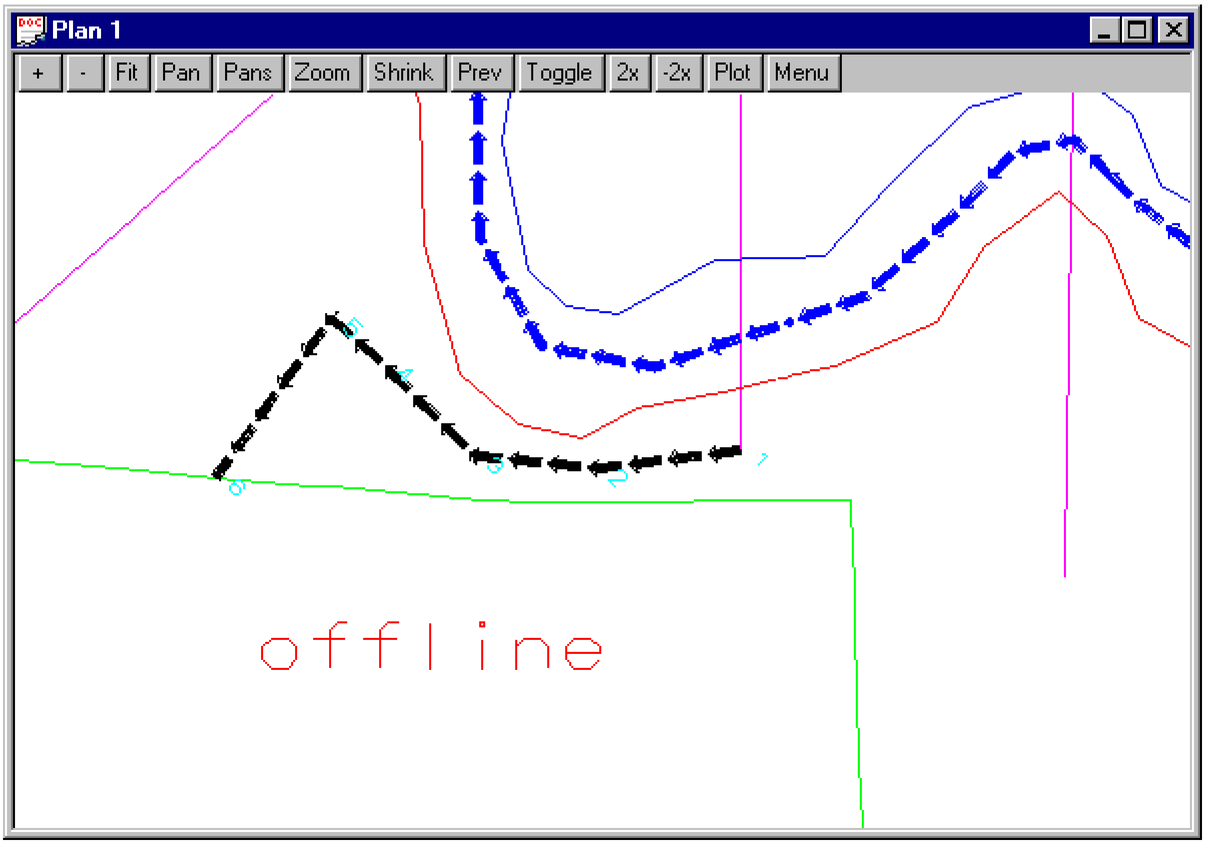
Auto height

XP SWMM catchment parameters

A sample drawing showing an inline reservoir in cyan (1-277) follows. Note that the river centre line touches the reservoir string both upstream and downstream. The left and right bank strings may continue straight through the reservoir.



A sample drawing of an offline reservoir follows (the reservoir string is shown in green). Note that the spill string (black) starts at the source string (point 1), then follows the section line to be cut and exported (points 2 to 5) and ends by touching the off line reservoir string at point 6. The rivers strings do not touch the offline storage strings.


Plan 1
+
-
Fit
Pan
Pans
Zoom
Shrink
Prev
Toggle
2x
-2x
Plot
Menu
offline

21.1.4 Spill strings

Spill strings are strings that link offline storage areas to a cross section (see drawing above). The string must begin by "touching" the source string and then proceed to the first point on the spill section. During the export the first point will NOT be exported as part of the spill section. After defining the end of the spill section the last point on the string must "touch" the reservoir string. Again this last point will NOT be exported as part of the spill section.

The default roughness for the spill section is the "left n" from the main export panel, the default slope is 0.5% and the default length is 100. See the manual setting table for **manual override settings**.

XPSWMM Only. 2 point culvert strings (6 max) may be draw across the spill string to create a multi conduit. The culvert strings should be drawn in the same direction as the spill string so that the upstream and downstream inverts follow in the same direction.

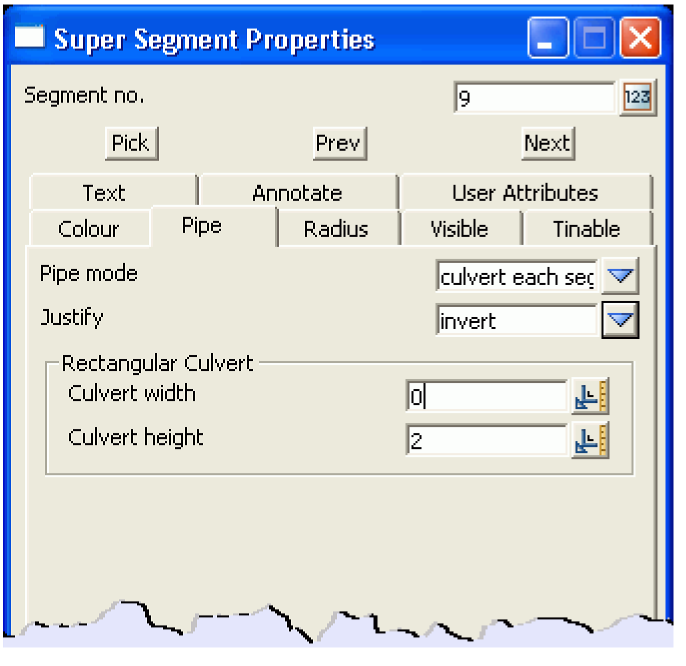
21.1.5 Define Culvert Locations

12d allows culverts to be included as a segment of a super string. The **XP SWMM** rivers interface allows culverts in 2 models. In the spill string model, multiple two point superstring mays cross a spill string to indicate a culvert through the embankment. In the river string model a segment of the river centre line may marked as a culvert. For parrallel culverts, segment attributes must be used.If a source string crosses the same segment than a mulit-link with both the culvert and the natural section will be exported.

Important note: Version 10.0 of xpswmm (at the time of writing) will only use inlet control curves on culverts with positive slopes.

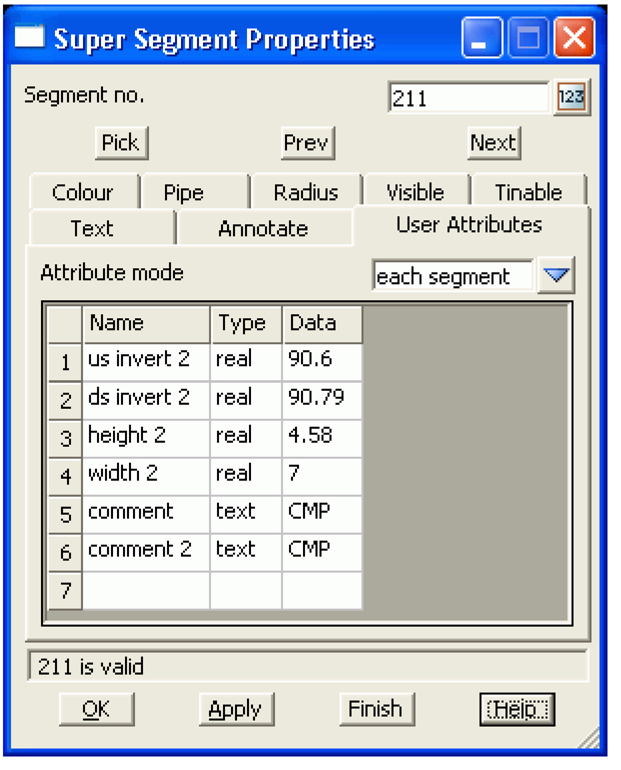
Adding a Culvert to a super string

1. **Super strings only!**
   Convert string to a super string (if required)
   Strings->Convert
2. **Mark the ends of the culvert.**
   Add a vertex at the upstream and downstream end of the culvert (Toggle vertices on if desired)
   String->Points Edit->Insert
3. **Mark the segemnt as a culvert or pipe**
   Use the super string segment editor to set the culvert data.
   Strings->Properties->Segments (all) and change Segment properties to use **culvert each segment** (box) or **pipe each segment** (round). Note that if there is one box culvert on the centre line then all must be box culverts. If you need pipes and box culverts combined, set the width to zero for pipes.
   Also set the Justify mode to indicate the type culvert levels you wish to specify.

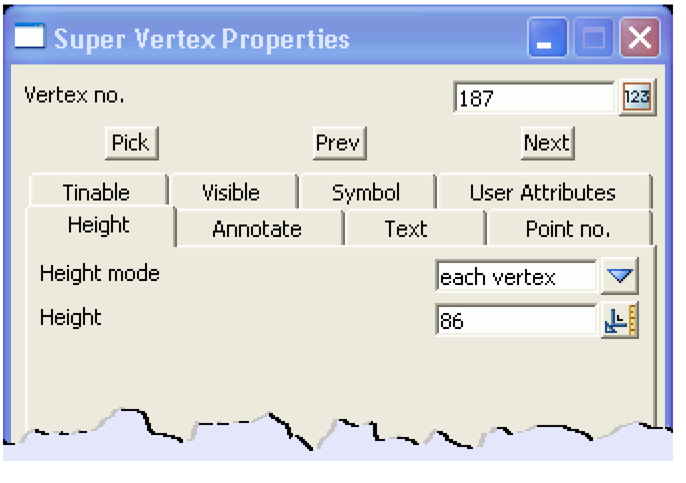


If there are two (max is 7) culverts at this location that have different inverts or sizes add the following user defined attributes to the segment. The comment attributes are printed by the Culvert Table routine and are used to label the culvert as well. If there are a number of identical culverts, the attributes "number of pipes" (integer) can be set to a value greater than 0. Additional attributes to customuse the culvert are listed at the end of this section. Default mannings n are 0.024 for pipes (corrigat metal) and 0.012 for box (concrete).

4. Conduit Factors are automatically set. Circular pipes are set to "Headwall (circular corrigated metal)" and box culverts are set to "45 deg Wingwall Flares (Rect, Conc)". Expansion and contraction energy loss coeficients have default values of 1.0 and 0.5 respectively but these may be changed using the attributes indicated below.



5. **Set the upstream and downstream levels**
   Select a vertex at the end of segment, ensure the **height mode** is **each vertex** and set the height. Use the **Next** or **Prev button** to move to the other end of the culvert and set its height as well.



Additional Culvert Segment Attributes (Spill and River strings)

Purpose	Attribute Name	Type	Typical Data Value
Culvert entrance loss	entrance loss	Real	0.5
Culvert exit loss	exit loss	Real	1.0
Mannings n	roughness	Real	0.014
Multiple identical culverts	number of pipes	Integer	2

Centre Line Culvert Segment Attributes (River strings)

The attribute name has the culvert number as a suffix. i.e. (height 2)

Purpose	Attribute Name	Type	Typical Data Value
Diameter/height	height n	Real	0.5
Box culvert width	width n	Real	1.0
Upstream invert	us invert n	Real	20.2
Downstream invert	ds invert n	Real	20.1
Culvert length	length n	Real	10.2

21.2 Culvert Table

Position of option on menu: Design=>Rivers=>XP SWMM interface=>Culvert table

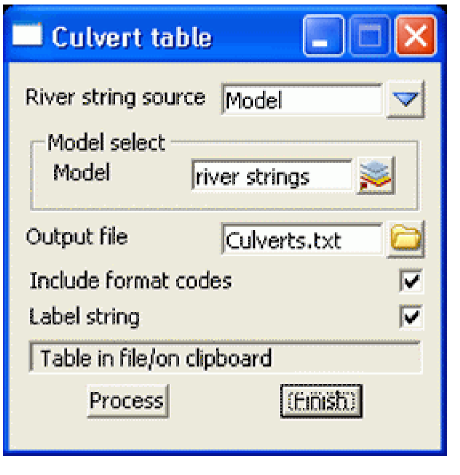
This option is used to create a table of the culverts created along the centre line of the river and to label the invert levels and comment for the first culvert on each segment (some segments may include attributes for more than on culvert). A sample is shown below.

Culvert Listing										
chainage	Number	US Invei	DS Inver	Height	Width	Quantity	X location	Y locatior	Comment	Segment
Centre lin	center line 1									
88444	1	125.28	124.95	11	7	1	683236	1389459		705
87269	1	127.58	127.24	6		1	683059	1390605	72/cmp	695
85983	1	123.13	123.72	6		1	682871	1391834	inv-72/cm	682
84317	1	124.15	123.47	6		1	681757	1392779		666
84317	2	124.59	125.15	6	0	1	681757	1392779	6ft cmp	666
83854	1	123.07	123.28	5		1	681319	1392820	60 inch cc	659
83854	2	123.54	123.42	5		1	681319	1392820	60 inch cc	659

See Also

Defining culvert locations

Usage



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
River string source	input	view	string,model,view

the imported cross section strings will be stored in this model

String,Model,View various

centre line string selection

Output file File box

text file to contain the table

Include format codes tick box

format codes are used to format tables in 12d...not desired for other programs

Label string tick box

The invert labels and the pipe size will be added as vertex and segement text

Process button

executes the option.

Finish button

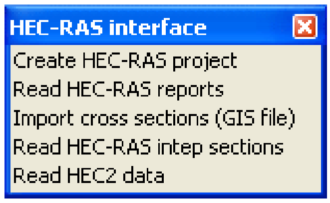
removes the dialogue from the screen

21.3 HEC-RAS Interface

Position of menu: **Design=>Rivers=>HEC-RAS interface**

The HEC-RAS interface creates the HEC-RAS project files ready to open and run. This includes the project, plan, flow and geometry files. Water levels are read back into 12d where they may be viewed in a 3D perspective view to easily identify extents of flooding.

The HEC-RAS walk right menu is,



See also

River and Source Strings

Create HEC-RAS files

Read HEC-RAS results

Presenting Water Level Results

Import cross sections (GIS file)

Read HECRAS Interp sections

Read HEC2 Data

How to for Rivers

Frequently Asked Questions (Rivers)

Exporting to HEC-RAS

The HEC-RAS project is created from a surface tin (representing the river bed and overbanks) and a model containing strings identified by their names “left bank”, “right bank and the name prefix, “centre line”. Any additional strings in the specified model will be ignored (warning messages will be given when you run the macro that any additional strings are being ignored). The low chainage (often zero) of the centre line strings must be at the downstream end of the reaches.

Cross sections are created at the location of the **source strings**. These source strings are initially created using the HEC-RAS option at a user defined spacing and section length. The user may alter these sections as desired. These may be shortened if they intersect at sharp bends in the river; they may be extended at extremely wide river sections or extra points may be added so that the section is no longer a straight line.

Source strings can be deleted and additional sections can be added by creating new source strings. The **Create source strings** tick box on the interface panel must **NOT** be selected to use the customised strings.

Presenting HEC-RAS Results in 12d

After the HEC-RAS analysis is complete the water level results are read back into 12d. Water level strings are created with the plan shape of the cross sections at the elevation retrieved from the HEC-RAS results. These strings are then triangulated to create a water surface tin from which the water level boundaries are determined. These results can then be shown in plan, long section, cross section and in 3D perspectives.

More details

21.3.1 HEC-RAS Write Panel

Position of option on menu: Design=> Rivers=>HEC-RAS interface=>Create HEC-RAS project

The HEC-RAS interface creates the HEC-RAS project files ready to open and run. This includes the project, plan, flow and geometry files. Water levels are read back into 12d where they may be viewed in a three dimension perspective view to easily identify extents of flooding.

See also

River and Source Strings

HEC-RAS Interface overview

Read HEC-RAS results

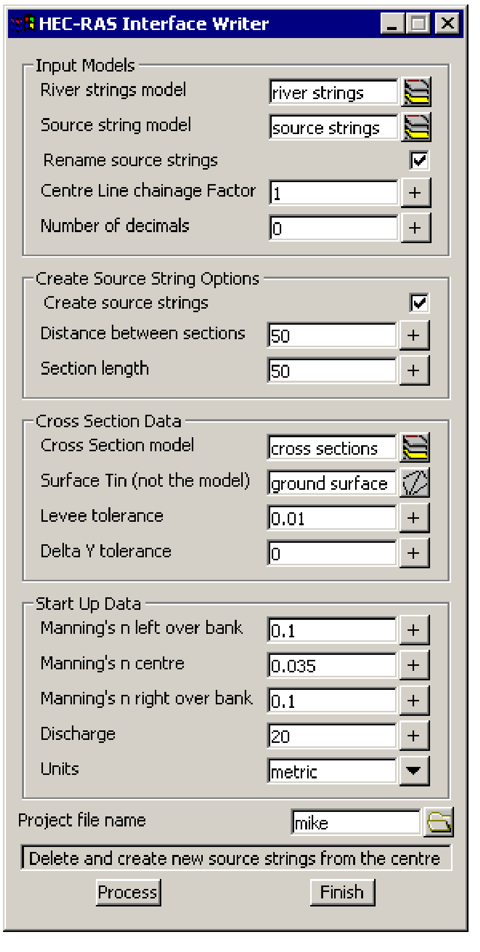
Presenting Water Level Results

How to for Rivers

Frequently Asked Questions (Rivers)

Usage

The HEC-RAS panel for creating the HEC-RAS project follows.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up

INPUT MODELS

River strings model Model box

*Model containing the centre line, left bank and right bank strings. **The centre line strings must begin downstream and proceed upstream.** The name of the river may follow the words "Centre line " (note the ending space). For example the centre line string may be named "Centre line Parramatta River, downstream reach". The comma separates the river name from the reach name. If no comma is included then the river name is repeated for the reach name.*

Confluences are modelled by a using a separate string for all reaches. Thus a system with a branch is modelled with three strings. The branch will be one string and the main reach will have a downstream string and an upstream string. The reaches must touch at the confluence.

The distance from the start of the upstream strings to the first cross section is used to model the confluence length.

The left and right bank strings need not be separate strings (see figure below).

Source string model Model box

*New source strings will be created in this model or existing source strings are contained in the model. See **Create source strings** tick box below.*

Rename Source Strings Tick box not selected

*If selected the source strings will be a name using the chainage along the river centre line. The **number of decimals** and the **Centre line chainage factor** (ft to miles or m to km) are specified below.*

Centre Line Chainage Factor Real box

The cross section names are created by dividing the chainage on the centre line by this factor. Typically 1000 is used to convert metres to kilometres and 5280 to convert feet to miles.

Number of decimals Real box

*When **Rename source strings** is selected, the source strings name will have the specified number of decimals.*

CREATE SOURCE STRNGS OPTIONS

Create Source Strings Tick box not selected

*When selected existing source strings are deleted and new ones created perpendicular to the centre line at the specified spacing and length. Once you have created the sources strings they can be easily modified. On the **Strings->Points Edit** menu you will find the selections **Move** (to move the end points), **Insert** (to insert additional points).*

Distance between sections Real box

The distance between the cross sections. At present no check is made for overlapping cross sections around river bends.

Section Length Real box

The length of the cross section with zero chainage at the mid point.

CROSS SECTION DATA

Cross section model Model box

The cross sections created and exported are stored in this model.

Surface Tin (not the model) Tin box

Tin or super tin to create the cross sections from (remember a tin is like a string. It is placed in a model.).

Levee Tolerance Real box

If the surface level drops more than this amount while moving away from the channel centre line then the crest is marked as a levee. A value of zero means that no levees are marked.

Delta Y tolerance Real box

This value filters out points on the cross section. Imagine a tube of this diameter passing over the cross section. The tube is elongated until one point lies outside the tube. The tube is shortened to the previous point and then all points inside the tube are deleted from the cross section. The tube then moves on to the next point. The filtered (smoothed) and original sections are kept for comparison. ***The final water tin is created from the ground tin and therefore the boundary string is located using the unfiltered section.***

STARTUP DATA

Manning's n Real box

Manning's n values for the left, right and centre channel sections.

Discharge Real box

This discharge is used at the upstream end of all reaches. If you have multiple river branches, you can set the flow for each branch inside HEC-RAS or inside 12d. This can be changed at each section See manual settings

Units Choice box

This selection will set the default units for the project being created.

Project file name Input box

The HEC-RAS project name. Remember HEC-RAS (2.0 or earlier) is limited to 8 characters only. If the total path name is too long HEC-RAS will not analyse the project.

21.3.2 HEC-RAS Read Panel

Position of option on menu: Design=>Rivers=>HEC-RAS interface=>Read HEC-RAS reports

After the HEC-RAS analysis is complete the water level results are read back into 12d. Water level strings are created with the plan shape of the cross sections at the elevation retrieved from the HEC-RAS results. These strings are then triangulated to create a water surface tin.

See also

HEC-RAS Interface overview

Create HEC-RAS files

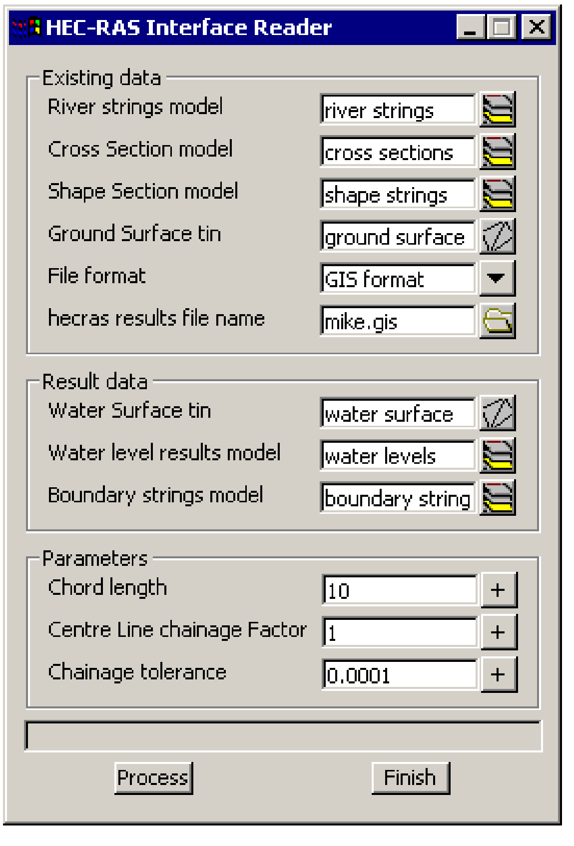
Presenting Water Level Results

How to for Rivers

Frequently Asked Questions (Rivers)

Usage

The HEC-RAS read panel follows.



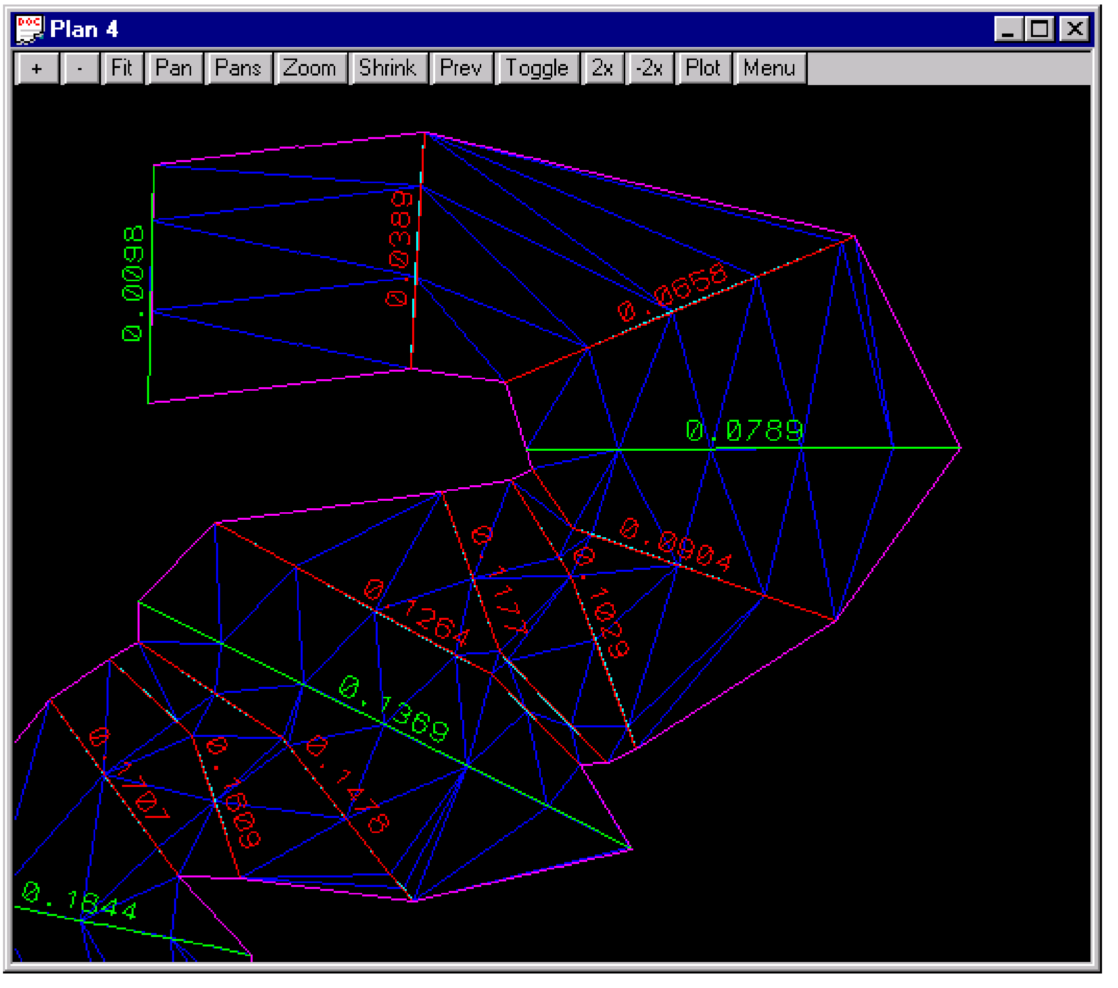
EXISTNG DATA

River strings model Model box

The river strings model specified in the write panel.

Cross Section model Model box

*The cross section model specified in the write panel. This must be specified but is only used when reading the *.rep file types (see **File format** below). The interface will search the string names in this model for the cross sections specified in the HEC-RAS report. A match is successful if the HEC-RAS* *cross section chainage and the string name are within the tolerance specified below in **Chainage tolerance**.*



Shape string model Model box

For meandering rivers, the cross sections (shown in green above) may not be at a close enough spacing to create a water surface that follows the river. 2D shape strings (shown in red above) can be created to create a water surface (shown in blue above) to follow the river. Note that water levels are extended when the shape strings are in a junction area or past the end of a reach.

Ground surface tin Tin box

*If a **boundary string model** is specified below, the intersection of this ground surface and the water surface will be determined. The strings will be stored in the model from the **boundary string model** field.*

File format Choice box

The GIS format should be used in most cases. The rep format is used for reading the HECRAS report files with the "Standard Table 1" selected in HECRAS.

HEC-RAS report file name File box

The HEC-RAS report file (GIS format) is created using the HEC-RAS menu selection

File=>Export GIS data

The HEC-RAS report file (REP format) created using the HEC-RAS menu selection

File=>Generate Report

The report must include the Standard Table 1.

RESULTS DATA

Water surface tin model Model box

The model to contain the new water surface tin.

Water surface tin Tin box

The name of the water surface tin to be created.

Water level results model Model box

The model where the water surface strings will be created at each cross section and shape string.

Boundary string model Model box

The model to contain the intersection strings between the water and ground surfaces specified above. If left blank no intersection strings will be calculated.

PARAMETERS

Chord Length Model box

This value set the spacing for the points on the water level strings (both cross section and shape strings). It is recommended that you use a length of no more than half of your average cross section and shape string lengths. A large value in this field may result in unexpected water level profiles for meandering rivers.

Centre line chainage factor Model box

This data is only required if the ***Shape string model*** *is used. The shape string names are created by dividing the chainage on the centre line by this factor. Typically 1000 is used to convert metres to kilometres and 5280 to convert feet to miles.*

Chainage Tolerance Real box

This is the tolerance used when the cross section chainage from the HEC-RAS report is compared with the cross section string names. A value of 0.00001 is excellent if you have not altered the cross section names in 12d or HEC-RAS.

However, if you have altered chainage names then you may have to increase the value of the tolerance. Suppose the tolerance is set to 0.001 and the water level for section 0.056 is read from the HEC-RAS report file. The interface will search for the first string with a name between 0.055 and 0.057. If you chose to great of a tolerance then more than one water level result will match a 12d cross section and a warning message will be given.

If you have one specific cross section that you would like to have a different tolerance set for (maybe only one section is giving you troubles), use the Attribute Editor

(Strings->User->Attribute Editor), select the cross section string and create a real type attribute named tolerance set to the tolerance desired.

21.3.3 Import cross sections (GIS file)

Position of option on menu: **Design=>Rivers=>HEC-RAS interface=>Import cross sections (gis file)**

This option is used to import HEC RAS cross sections, interpolated cross sections, source strings (Cut lines), river centre lines and storage area boundaries into 12d. Note that the imported data will be given the X,Y coordinates used in the HECRAS project.

See Also

River and Source Strings

HEC-RAS Interface

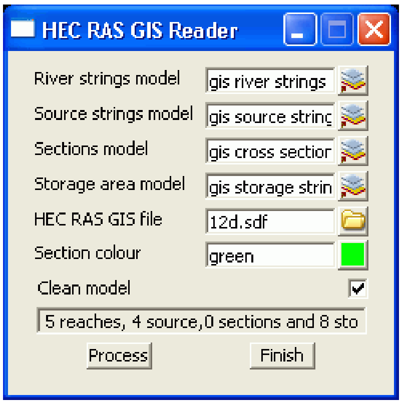
How to for Rivers

Frequently Asked Questions (Rivers)

Usage

This panel is accessed from the menu selection

Design => Rivers => HEC-RAS Interface=>Import cross sections (GIS file)



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
River strings model	model box		

the imported centre line strings will be stored in this model (REACH:CENTERLINE: data). If blank these string will not be imported.

Source string model model box

the imported source strings will be stored in this model (CROSS-SECTION:CUT LINE: data) If blank these string will not be imported.

Cross section model model box

the imported cross section strings will be stored in this model (CROSS-SECTION:SURFACE LINE: data). If blank these string will not be imported.

Storage area model model box

storage area boundaries will be stored in this model (strings not supported in 12d V7 interface). If blank these string will not be imported.

HEC RAS GIS File File box

the GIS file generated from HEC RAS (.RASexport.sdf)*

Section colour colour box

Cross sections will be created using this colour unless the section name contained illegal characters

Clean model tick box

if selected all strings in the Cross section model will be deleted before the cross sections are imported

Process button

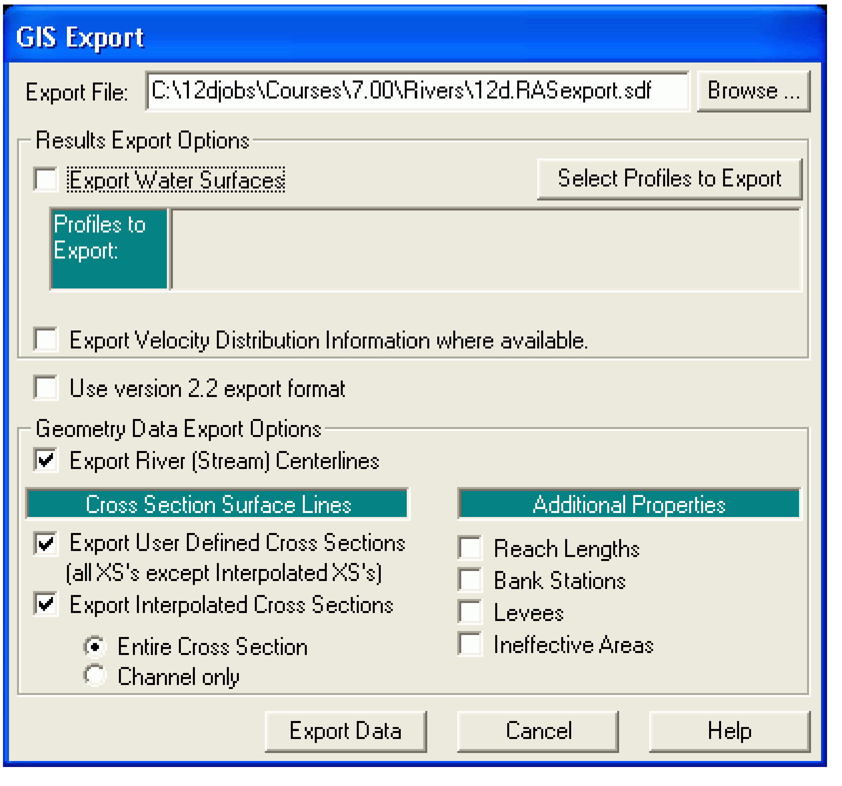
executes the option and the number of string imported will be reported in the message area.

Finish button

removes the dialogue from the screen

Notes:

The RASexport.sdf file is create from HEC RAS using File->Export GIS Data. From this panel the desired data to be exported may be selected. In this example all three Geometry tick boxes have been selected.



21.3.4 Read HEC-RAS interp sections

Position of option on menu: Design=>Rivers=>HEC-RAS interface=>Read HEC-RAS intep section

This option is used to import HEC RAS interpolated cross sections (names begin with *) based on their distance along the left and right bank strings.

The low chainage must be at the downstream end for both the "left bank" and "right bank" strings.

See Also

River and Source Strings

HEC-RAS Interface

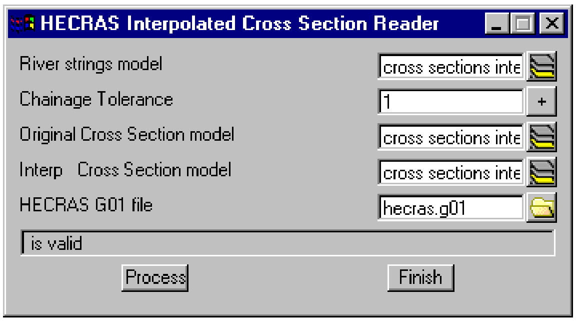
How to for Rivers

Frequently Asked Questions (Rivers)

Usage

This panel is accessed from the menu selection

Design => Rivers => HEC-RAS Interface=>Read HEC RAS Interp sections



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

River strings model model box

the centre line, left and right bank strings must exist in this model

Chainage tolerance Real box

existing cross sections must be located in 12d model. This value is the tolerance used when the HEC RAS cross section names are compared to the 12d string names.

Original cross section model model box

the HEC RAS cross section names will be compared with the string names in this model

Interp cross section model model box

the imported cross section strings will be stored in this model

HEC RAS G01 File File box

*the G01 (geometry) file used by HEC RAS to store the data in raw format. This is **not** the GIS format.*

Process button

executes the option.

Finish button

removes the dialogue from the screen

21.3.5 Read HEC2 Data

Position of option on menu: Design=>Rivers=>HEC-RAS interface=>Read HEC2 data

This option is used to import HEC2 cross sections into 12d. Since the file does not contain any easting and northing data, the cross sections will be lined up vertically and will have to be placed manually inside 12d.

See Also

River and Source Strings

HEC-RAS Interface

Moving Strings ragg

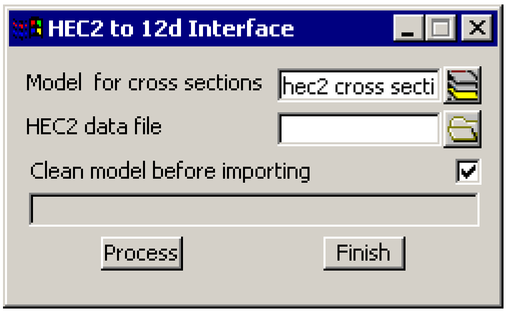
How to for Rivers

Frequently Asked Questions (Rivers)

Usage

This panel is accessed from the menu selection

Design => Rivers => HEC-RAS Interface=>Read HEC2 Data



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Cross section model	model box		

the cross section created will be placed in this model

HEC2 cross section	File boxbox		

the HEC2 cross section data file to be read

Process	button		

read the data and create the cross sections.

Finish	button		

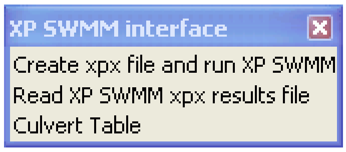
removes the dialogue from the screen

21.4 XP-SWMM Interface

Position of menu: Design=>Rivers=>XP SWMM interface

The XP-SWMM interface launches your XP-SWMM program and loads either a default xp file from the set_ups path (master_rivers.xp) or your existing project. The 12d data is automatically imported in the XP-SWMM project (via a *.xpx file) to either create a new river reach or update an existing reach. Water levels are read back into 12d where they may be viewed in a 3D perspective view to easily identify extents of flooding.

The XP-SWMM walk right menu is,



See also

River and Source Strings

Defining Centre Line Culverts

XP SWMM Culverts

Spill strings

Hydrology Data

Create XP-SWMM files

Read XP-SWMM results

Presenting Water Level Results

How to for Rivers

Frequently Asked Questions (Rivers)

Exporting to XP-SWMM

The XP-SWMM project is created from a surface tin (representing the river bed and overbanks) and a model containing strings identified by their names "left bank", "right bank and the name prefix, "centre line". Any additional strings in the specified model will be ignored (warning messages will be given when you run the macro that any additional strings are being ignored). The low chainage (often zero) of the centre line strings must be at the downstream end of the reaches.

Cross sections are created at the location of the source strings. Source strings can be deleted and additional sections can be added by creating new source strings. The **Create source strings** tick box on the interface panel must **NOT** be selected to use the customised strings.

Presenting XP-SWMM Results in 12d

After the XP-SWMM analysis is complete the water level results are read back into 12d. Water level strings are created with the plan shape of the cross sections at the elevation retrieved from the XP-SWMM results (xpx file). These strings are then triangulated to create a water surface tin from which the water level boundaries are determined. These results can then be shown in plan, long section, cross section and in 3D perspectives.

More details

21.4.1 XP-SWMM Write Panel

Position of option on menu: Design=>Rivers=>XP SWMM interface=>Create xpx file and run XP SWMM

The XP-SWMM interface creates the XP-SWMM project files ready to open and run. This includes the project, plan, flow and geometry files. Water levels are read back into 12d where they may be viewed in a three dimension perspective view to easily identify extents of flooding.

See also

River and Source Strings

Hydrology Data

XP-SWMM Interface overview

Read XP-SWMM results

Presenting Water Level Results

How to for Rivers

Frequently Asked Questions (Rivers)

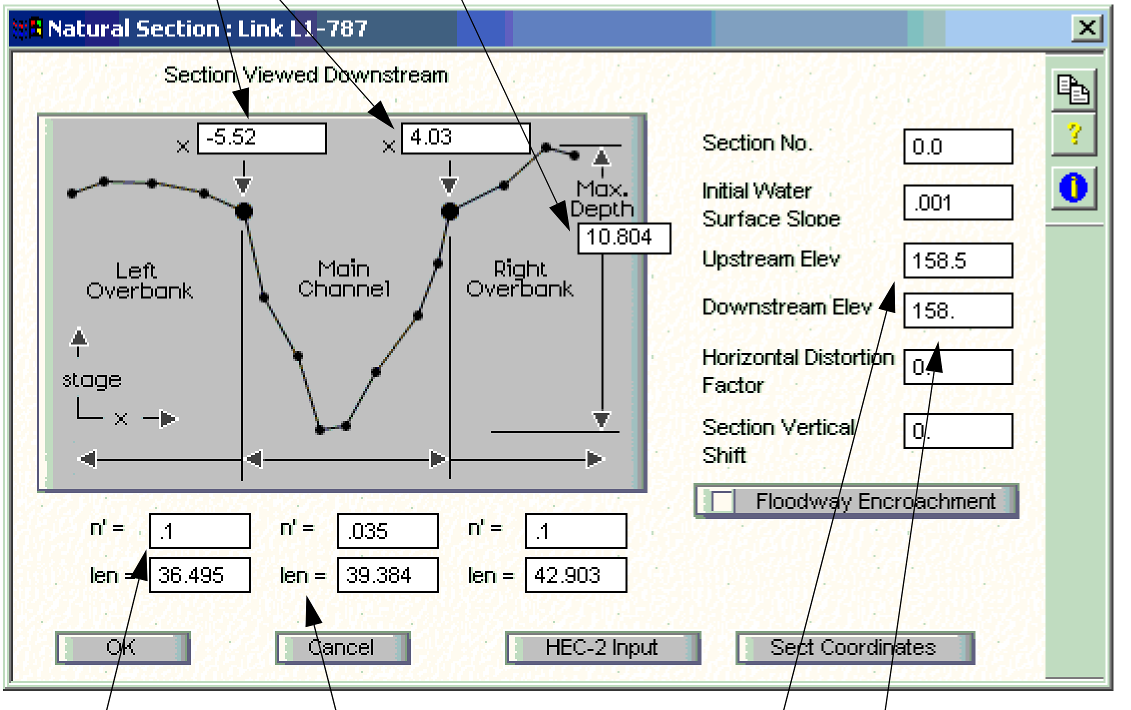
The source strings define the location where cross sections are cut from the ground surface TIN. A node is created in XP-SWMM at this location and the cross section shape is used to the next downstream node (no cross section is created at the most downstream source string).

The centre line and overbank channel lengths are measured in the downstream direction. XP SWMM link and node input panels are displayed below with the relationship to the ***12d Model*** described.

XP SWMM Link Data

cross section chainage at intersection of left and right bank strings

max elev - min elev at upstream cross section



n values from 12d panel

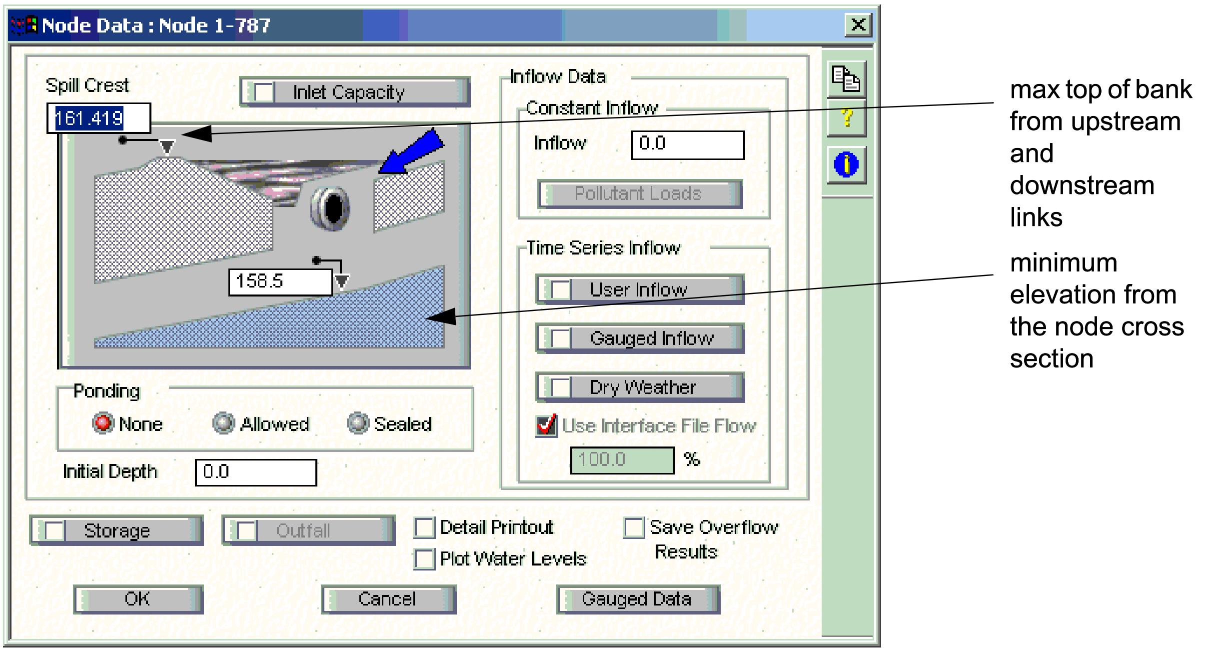
channel lengths measured along river strings

Minimum cross section elev from upstream node.

Minimum cross section elev from downstream node

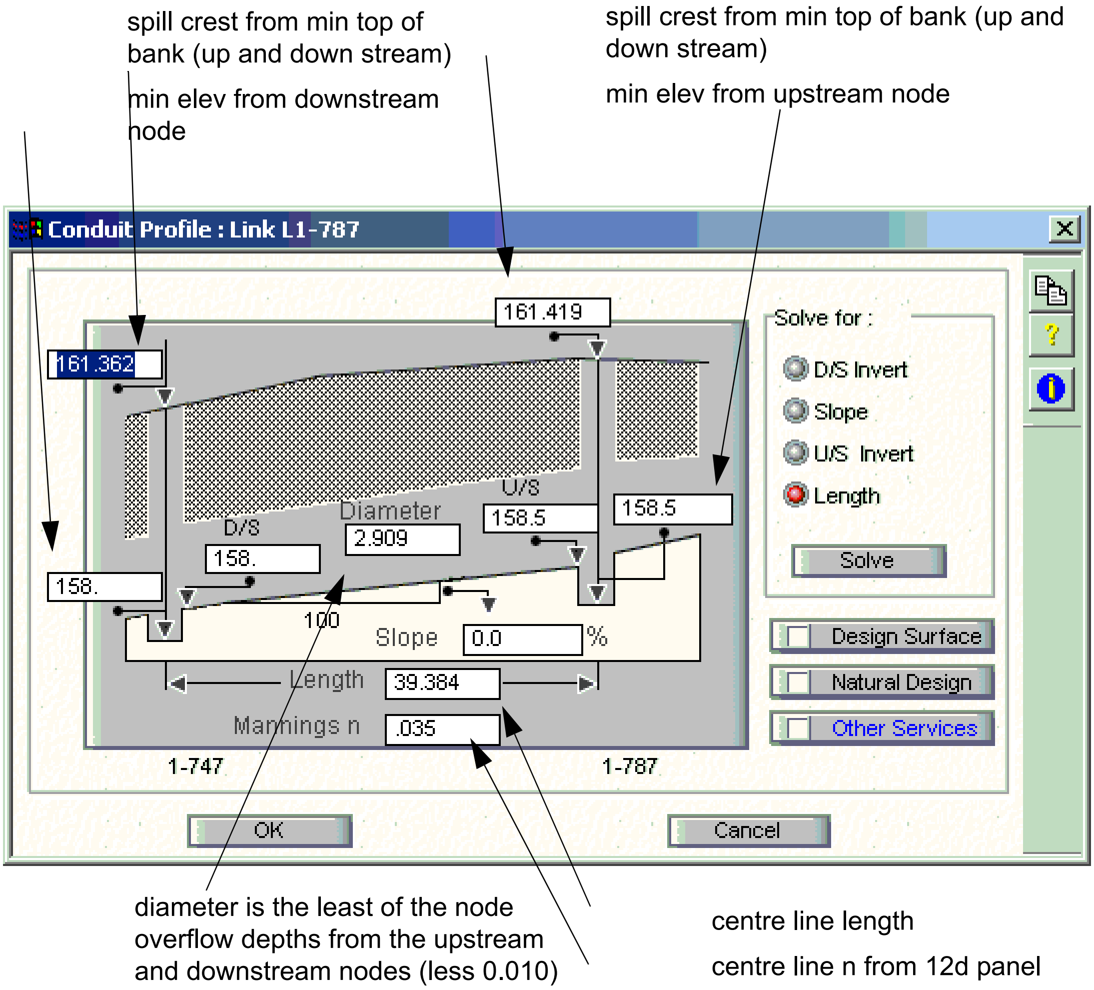
.

XP SWMM Node Data



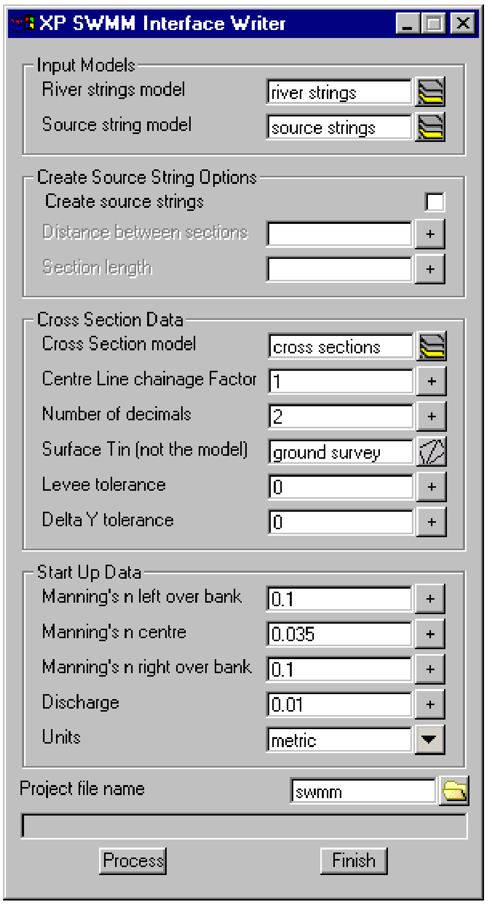
XP SWMM Conduit Data

The slope of the conduit/link is calculated by assuming that its invert matches vertically with the invert of the downstream link. The overflow levels at the nodes are calculated by determining the lowest of the bank levels for the upstream and downstream links. The least of the overflow depths at the nodes is then used as the link diameter.



Usage

The XP-SWMM panel for creating the XP-SWMM project follows.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
River strings model	Model box		

Model containing the centre line, left bank and right bank strings. ***The centre line strings must begin downstream and proceed upstream.*** *The name of the river may follow the words "Centre line " (note the ending space). For example the centre line string may be named "Centre line Parramatta River, downstream reach". The comma separates the river name from the reach name. If no comma is included then the river name is repeated for the reach name.*

The left and right bank strings need not be separate strings (see figure below).

Source string model Model box

New source strings will be created in this model or existing source strings are contained in the model. See ***Create source strings*** *tick box below.*

Create Source Strings Tick box not selected

When selected existing source strings are deleted and new ones created perpendicular to the centre line at the specified spacing and length. Once you have created the sources strings they can be easily modified. On the ***Strings->Points Edit*** *menu you will find the selections* ***Move*** *(to move the end points),* ***Insert*** *(to insert additional points).*

Distance between sections Real box

The distance between the cross sections. At present no check is made for overlapping cross sections *around river bends.*

Section Length Real box

The length of the cross section with zero chainage at the mid point.

Cross section model Model box

The cross sections created and exported are stored in this model.

Centre Line Chainage Factor Real box

The cross section names are created by dividing the chainage on the centre line by this factor. Typically 1000 is used to convert metres to kilometres and 5280 to convert feet to miles.

Number of decimals Integer box

*The cross section names are created with this many decimals. CAUTION XP SWMM only allows 10 characters for the names and each link name begins with "**Lx-**" That leaves 7 characters for the chainages.*

Surface Tin (not the model) Tin box

Tin or super tin to create the cross sections from (remember a tin is like a string. It is placed in a model.).

Levee Tolerance Real box

Not implemented in XP-SWMM.

Manning's n Real box

Manning's n values for the left, right and centre channel sections.

Delta Y tolerance Real box

This value filters out points on the cross section. Imagine a tube of this diameter passing over the cross section. The tube is elongated until one point lies outside the tube. The tube is shortened to the previous point and then all points inside the tube are deleted from the cross section. The tube then moves on to the next point. The filtered (smoothed) and original sections are kept for comparison. ***The final water tin is created from the ground tin and therefore the boundary string is located using the unfiltered section.***

Discharge Real box

This discharge is used at the upstream end of all reaches. If you have multiple river branches, you can set the flow for each branch inside XP-SWMM or inside 12d. This can be changed at each section See manual settings

Units Choice box

This selection will set the default units for the project being created.

Project file name Input box

*The XP-SWMM *.xp name to be created or updated. An xpx file will also be created using this name to transfer the data.*

After selecting the **Process** button the cross section strings will be created and exported to a xpx file (using the project file name as the stem. **Never included ".xpx" in the project file name**) The xpx file will be over written without any warning. This is only used as a temporary transfer file.

12d needs to know where XPSWMM program and the XP SWMM working directory are. The system file that contains these locations is called XPSWMM.4d. (See System file path for its location). The file contents are as follows:

Line 1the XP SWMM working directory
Line 2the full path to the XP SWMM program
Line 3the destination of the hydro.ini file when it is created

IMPORTANT: include the final "\" on lines 1 and 3.

The **XPSWMM.4d** file as supplied follows;

c:\xps\xp-swmm\work\
c:\xps\xp-swmm\xpswmp32.exe
c:\xps\xp-swmm\

Next 12d will check for an XP project in your XP SWMM work directory (project file name + ".xp"). If it does not exist the file **Master_rivers.xp** will be copied from the System file path to your XP SWMM working directory. If a wp file already exists in your working directory you will see the following options dialogue.

The default is to update the wp file. Nodes and links in the file with the same names as in the export will be updated. If they do not exist they will be added. The import will NOT delete any nodes or links inside XP SWMM. This option is often used to merge rivers data with urban drainage data.

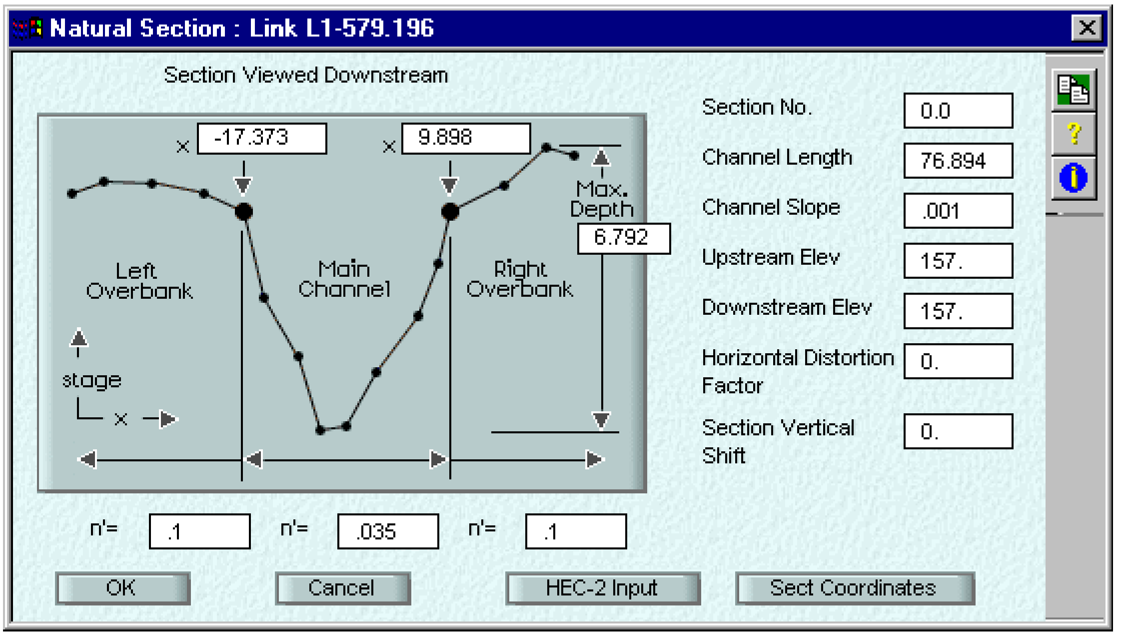
The over write option will copy the Master_river.xp file into the working directory and over write the existing file.

In either of the two cases the xpx file will be imported after XP SWMM has started. When you exit your XP SWMM session regardless of whether or not you save the XP file the XPX file will be exported over writing the original export file from 12d.

Your third option if not to run XP-SWMM at all and work with the XPX file as you see fit.

21.4.1.1 Export Details

The following diagram displays how the values from 12d are exported to XP-SWMM



At every source string location a node is created in XP-SWMM. The cross section geometry is then used for the downstream reach and is exported looking in the downstream direction.

The node name is set manually or calculated inside 12d by obtaining the centre line chainage at the intersection with section string. This value is divided by the **Centre line chainage factor** and rounded to the **Number of decimals** specified in the 12d export dialogue box. This value is then prefixed with the reach index followed by a dash. The downstream link name is set to the node name prefixed with "L". If you are planning to merge the river file with am XP SWMM drainage network then you will want to manually set the node names to the drainage pit names for the most upstream and most downstream source strings.

When selecting the **Number of decimals** in the 12d export dialogue, ensure that the total link name generated does not exceed the XP limit of 10 characters.

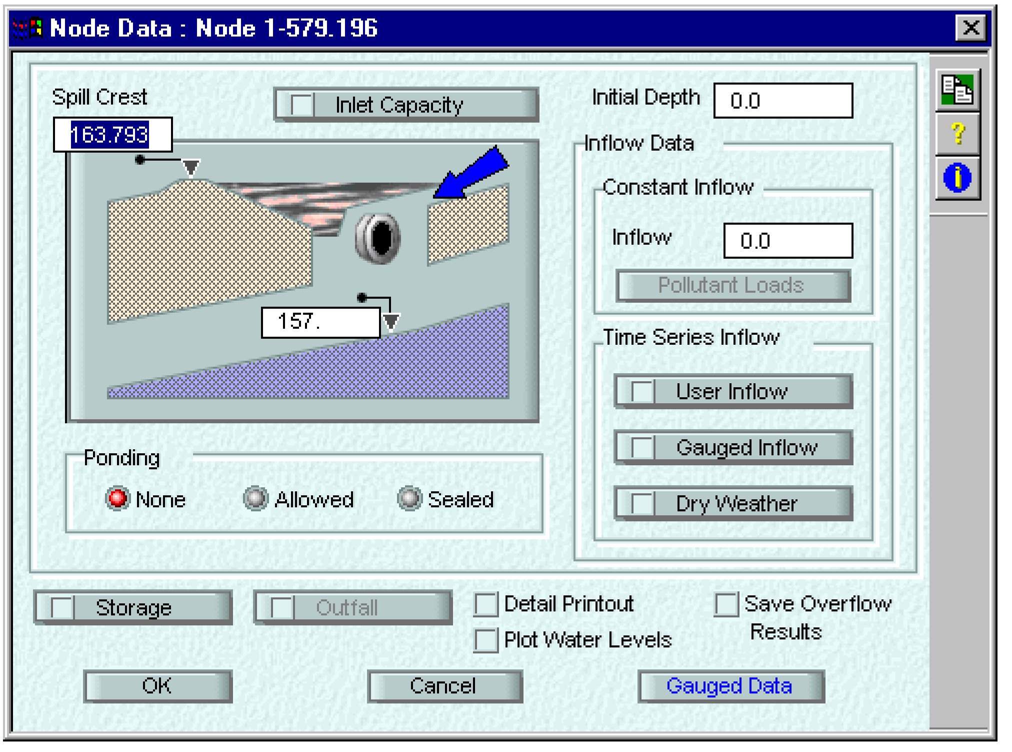
The section number variable is not exported to XP-SWMM.

Where the centre line cross the section in 12d is marked as zero chainage in XP-SWMM. The left overbank (-17.373) and the right overbank (9.898) are determined by the intersection of the left and right bank strings with the section. The n values are the values that are entered in the 12d export dialogue box. The maximum depth is calculated by subtracting the lowest elevation from the greatest elevation for both this cross section and the downstream cross section. The minimum of the two values is used.

The channel length is determined by subtracting the downstream section's centre line chainage from the current section centre line chainage. The **Centre line chainage factor** is **NOT** used in this calculation.

Upstream elevation is obtained by calculating the minimum elevation at the current section and the downstream elevation is the same value for the downstream cross section. The channel slope is calculated using these elevations and the channel length above.

The horizontal distortion factor and section vertical shift are set to zero. A section coordinate point in generated every time the source string crosses a 12d triangle edge, at the centre line and the left/right bank locations.



The spill crest level at the node is set to the maximum level on the cross section. Constant inflow is zero unless the node is the most upstream section on the reach are the flow value for the source string has been set manually see manual settings.

21.4.1.2 Hydrology Data

Hydrolgy data may be exported to the runoff layer of XP SWMM for both the reservoir strings and the source strings. There are 2 requirements to enable the hydrology.

21.4.1.2.1 Key Points

1. If you plan to use storage areas, you need to draw reservoir strings (all in one model) and each should have a name (to become the XP SWMM node name).
2. You must create a file named "hydrology.txt" and store it in the project working folder. This file will contain the default xpx variable names. These defaults may be over riden using the same xpx variable name as a string attribute on the source/reservoir string (see format below).
3. The reservoir or source string must have an integer string attribute named "xpx r rfcmnt" with a value greater than 0 (max of 5). Use the Strings->Properties->Attributes or Strings->User->Attribute Editor to create/change this attribute.
4. Total area for the reservoir strings is calculated at each export time.

21.4.1.2.2 Hydology.txt file format

The file is tab delimted and each line consists of three pieces of data: the XPSWMM variable name, the default value, and the type of data (integer, real or text).

You enter the xpx variable name

Any of these default values may be specified using a string attribute on the reservoir or source string. The attribute name must be of the same type as defined in the hdrology.txt file. If data for a second XP SWMM catchment is desired add a " 2" to the end of the attribute name. Up to 5 catchments are allowed in XP SWMM.

A sample hydrology.txt file follows:

```
// SCS hydrology screen

R_CN        85          real
R_TC        60          real
R_SHF       256         integer
R_SHAPE     0           integer

R_IADEPTH0.04           real
R_IAFRACT0.2            real
R_IA        1           integer
CNTLS       5           integer
R_FSCS      1           integer

// sub catchment screen

R_RAINSEL"SCS Type II FL Mod."text
R_INFILSEL              ""text
R_GWTAG     0           integer

// runoff node screen

R_WAREA     1.          real
R_WIMP      1.          real
R_WIDTH     1.          real
R_WSLOPE1.              real
```

21.4.1.2.3 Attribute name format for string attributes.

The attribute name is the XP variable name prefixed by "xpx " with the underscore character in the xpx variable replaced by a space. For example the xpx variable "R_RAINSEL" would be set using the attribute name "xpx R RAINSEL".

21.4.1.3 Exporting River Junctions

River junctions are defined by the intersection of the centre line strings in 12d. The cross section immediately downstream of the intersection becomes the junction node. It is used as the downstream cross section for all upstream branches. The channel length for the last link on the tributary is set to the distance the centre line string extends beyond the cross section. The distance downstream along the main branch is **NOT INCLUDED!**

21.4.2 XP-SWMM Read Panel

Position of option on menu: Design=>Rivers=>XP SWMM interface=>Read XP SWMM xpx results file

After the XP-SWMM analysis is complete and the XP-SWMM program is closed the water level data is written to an xpx file that 12d will read. Water level strings are created with the plan shape of the cross sections at the elevation retrieved from the XP-SWMM results. These strings are then triangulated to create a water surface tin.

See also

XP-SWMM Interface overview

Create XP-SWMM files

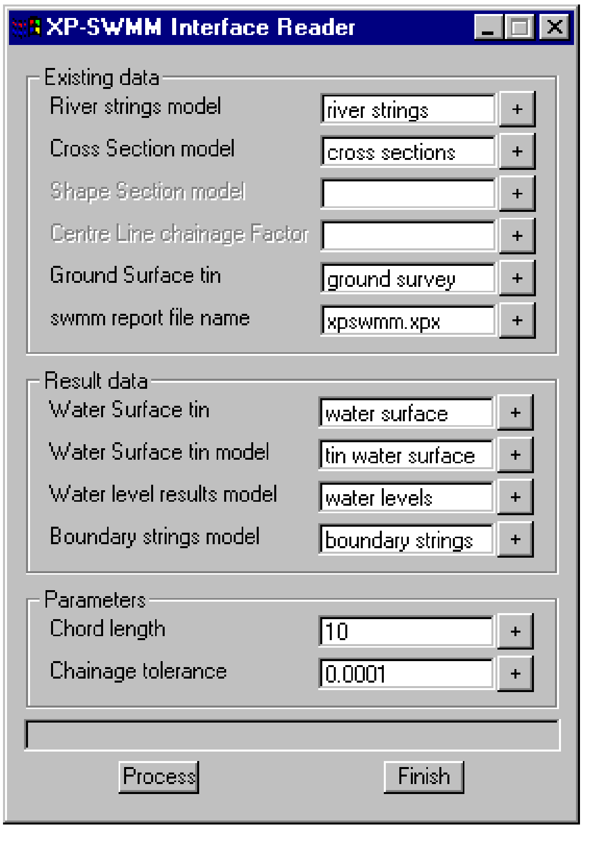
Presenting Water Level Results

How to for Rivers

Frequently Asked Questions (Rivers)

Usage

The XP-SWMM read panel follows.

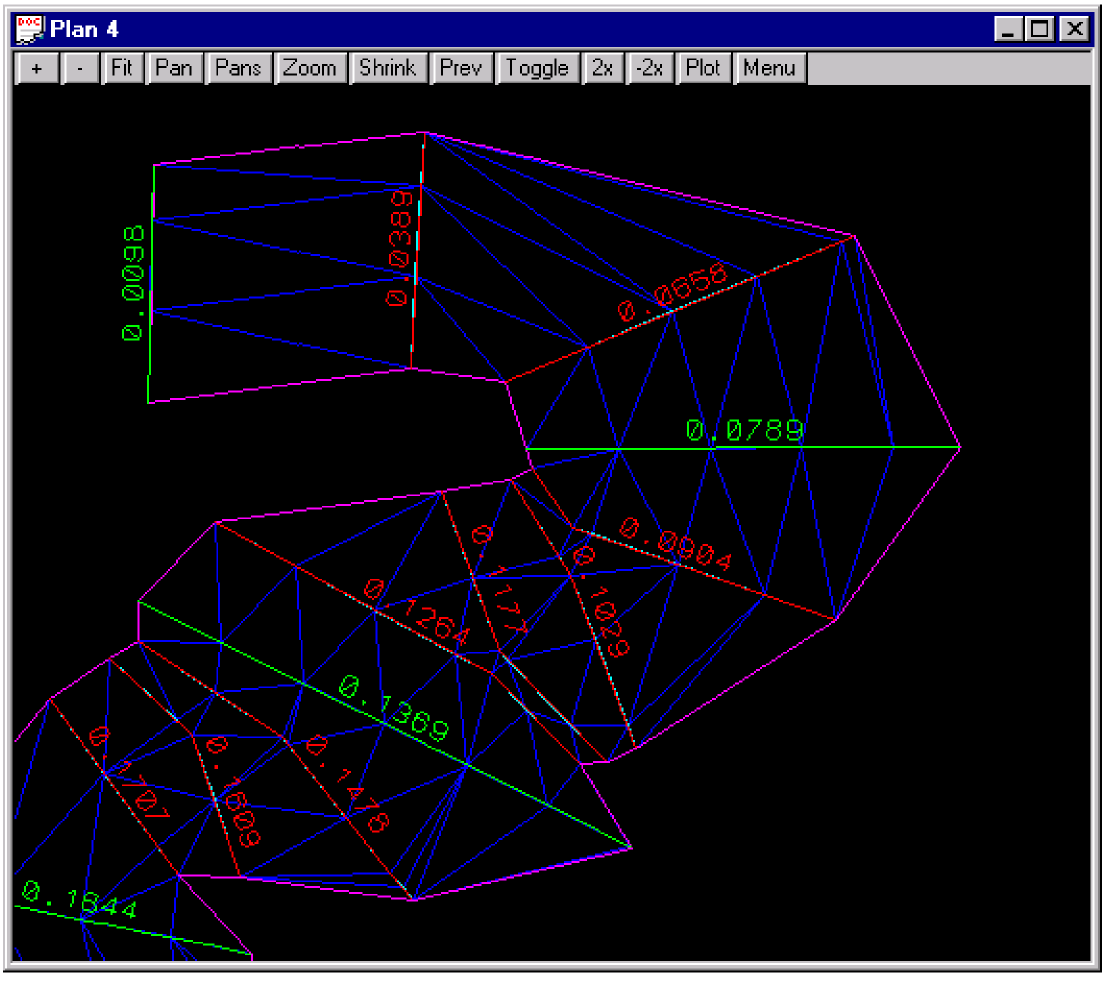


River strings model Model box

The river strings model specified in the write panel.

Cross Section model Model box

*The cross section model specified in the write panel. The interface will search the string names in this model for the cross sections specified in the XP-SWMM report. A match is successful if the XP-SWMM cross section chainage and the string name are within the tolerance specified below in **Chainage tolerance**.*



Shape string model Model box

For meandering rivers, the cross sections (shown in green above) may not be at a close enough spacing to create a water surface that follows the river. 2D shape strings (shown in red above) can be created to create a water surface (shown in blue above) to follow the river. Note that water levels are extended when the shape strings are in a junction area or past the end of a reach.

Centre line chainage factor Model box

This data is only required if the ***Shape string model*** *is used. The shape string names are created by dividing the chainage on the centre line by this factor. Typically 1000 is used to convert metres to kilometres and 5280 to convert feet to miles.*

Ground surface tin Tin box

If a ***boundary string model*** *is specified below, the intersection of this ground surface and the water surface will be determined. The strings will be stored in the model from the* ***boundary string model*** *field.*

XP-SWMM report file name File box

The XP-SWMM xpx file created automatically when closing XP-SWMM or via the XP-SWMM menu selection

File=>Export Data

Water surface tin model Model box

The model to contain the new water surface tin.

Water surface tin Tin box

The name of the water surface tin to be created.

Water level results model Model box

The model where the water surface strings will be created at each cross section and shape string.

Boundary string model Model box

The model to contain the intersection strings between the water and ground surfaces specified above. If left blank no intersection strings will be calculated.

Chord Length Model box

This value set the spacing for the points on the water level strings (both cross section and shape strings). It is recommended that you use a length of no more than half of your average cross section and shape string lengths. A large value in this field may result in unexpected water level profiles for meandering rivers.

Chainage Tolerance Real box

This is the tolerance used when the cross section chainage from the XP-SWMM report is compared with the cross section string names. A value of 0.00001 is excellent if you have not altered the cross section names in 12d or XP-SWMM.

However, if you have altered chainage names then you may have to increase the value of the tolerance. Suppose the tolerance is set to 0.001 and the water level for section 0.056 is read from the XP-SWMM report file. The interface will search for the first string with a name between 0.055 and 0.057. If you chose to great of a tolerance then more than one water level result will match a 12d cross section and a warning message will be given.

If you have one specific cross section that you would like to have a different tolerance set for (maybe only one section is giving you troubles), use the Attribute Editor

(Strings->User->Attribute Editor), select the cross section string and create a real type attribute named tolerance set to the tolerance desired.

21.5 MIKE11 Interface

Position of menu: **Design=>Rivers=>Mike11 interface**

The MIKE 11 interface creates the MIKE 11 project files including *.bnd11, *.hd11, *.nwk11, sim11 and the cross section data text file (to be imported into *.xns11). Water levels are read back into 12d where they may be viewed in a 3D perspective view, on cross sections and on river profiles. Plan drawings easily identify extents of flooding and all data can be plotted onto engineering drawings.

The Mike11 walk right panel is,

See also

River and Source Strings

MIKE11 Interface Overview

MIKE 11 Write Panel

MIKE 11 Read Panel

Presenting Water Level Results

How to for Rivers

Frequently Asked Questions (Rivers)

Exporting to MIKE 11

The MIKE 11 project is created from a surface tin (representing the river bed and overbanks) and a model containing river centre line strings (identified by the name prefix, “centre line”). The low chainage (often zero) of the centre line strings must be at the upstream end of the reaches.

Cross sections are created at the location of the **source strings**. These source strings are initially created using the MIKE 11 option at a user defined spacing and section length, imported from drawing packages or manually created in 12d. The user may alter these sections as desired. They may be shortened if they intersect at sharp bends in the river; they may be extended at extremely wide river sections or extra points may be added so that the section is no longer a straight line.

Source strings can be deleted and additional sections can be added by creating new source strings. The Create source strings tick box on the interface panel must NOT be selected to use the customised strings.

Presenting MIKE 11 Results in 12D

After the MIKE 11 analysis is complete the maximum water level results are read back into 12d. Water level strings are created with the plan shape of the cross sections at the elevation retrieved from the MIKE 11 results. These strings are then triangulated to create a water surface tin.

How the water level boundaries are determined

21.5.1 MIKE 11 Write Panel

Position of option on menu: Design=>Rivers=>Mike11 interface=>Create Mike11 files

The MIKE 11 interface creates the MIKE 11 project files including *.bnd11, *.hd11, *.nwk11, *.sim11 and the cross section data text file (to be imported into *.xns11). Water levels are read back into 12d where they may be viewed in a 3D perspective view to easily identify extents of flooding.

See also

River and Source Strings

MIKE 11 Read Panel

Presenting Water Level Results

How to for Rivers

Frequently Asked Questions (Rivers)

Usage

12d creates most of the files necessary to run MIKE11. The exception to these are the time series files (both water level and discharge data). The simulation file created assumes that these files will have the same name as the centre line strings used in 12d. For example, if you name your centre line string *centre line Major River* then your time series files will be named *Major River-H.DFS0* for the tail water conditions and *Major River-q.dfs0* for your discharges.

The *.bnd11, *.hd11, *.nwk11 and *.sim11 files are created by appending 12d data to default data found in the following files.

cross_sections.4d

hd11-end.4d

hd11-header.4d

nwk11-header.4d

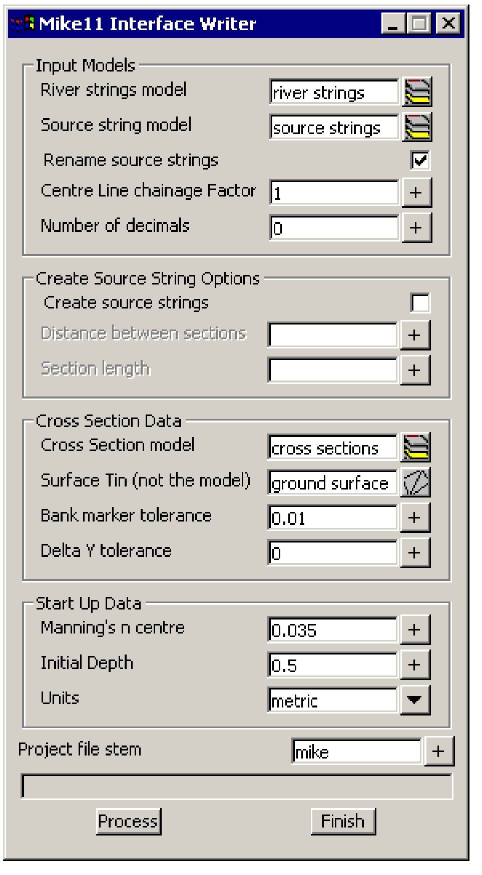
nwk11-options.4d

sim11-header.4d

sim11-period.4d

The user need not modify these files unless they would like to change the default values used when first creating the MIKE11 project. If you plan to modify these files, they are found in the 12d *setups* directory. Before modifying they should be copied to the 12d *user* directory (global defaults) or into the current project directory if they are project specific.

The MIKE 11 panel for creating the MIKE 11 project follows.



The fields and buttons used in this panel have the following functions:

Field Description   Type   Defaults   Pop-Up

River strings model   Model box

Model containing the river centre line strings. ***The centre line strings must begin upstream and proceed downstream.*** *The name of the river must follow the words "centre line " (note the ending space). The name of the river may follow the words "Centre line " (note the ending space). For example the centre line string may be named "Centre line Parramatta River, downstream reach". The comma separates the river name from the reach name. If no comma is included then the river name is repeated for the reach name.*

Confluences are modelled by a using a separate string for all reaches. Thus a system with a branch is modelled with three strings. The branch will be one string and the main reach will have a downstream string and an upstream string. The reaches must touch at the confluence.

Source string model   Model box

New source strings will be created in this model or existing source strings are contained in the model. See ***Create source strings*** *tick box below.*

Create Source Strings   Tick box   not selected

When selected existing source strings are deleted and new ones created perpendicular to the centre line at the specified spacing and length. Once you have created the sources strings they can be easily modified. On the ***Strings->Points Edit*** *menu you will find the selections* ***Move*** *(to move the end points),* ***Insert*** *(to insert additional points).*

Distance between sections Real box

The distance between the cross sections. At present no check is made for overlapping cross sections around river bends.

Section Length Real box

The length of the cross section with zero chainage at the mid point.

Cross section model Model box

The cross sections created and exported are stored in this model.

Centre Line Chainage Factor Real box

The cross section names are created by dividing the chainage on the centre line by this factor. Typically 1000 is used to convert metres to kilometres and 5280 to convert feet to miles.

Surface Tin (not the model) Tin box

Tin or super tin to create the cross sections from (remember a tin is like a string. It is placed in a model.).

Bank Marker Tolerance Real box

If the surface level drops more than this amount while moving away from the channel centre line then the crest is used as a Bank Marker. A value of zero means that no bank marks are created.

Delta Y tolerance Real box

This value filters out points on the cross section. Imagine a tube of this diameter passing over the cross section. The tube is elongated until one point lies outside the tube. The tube is shortened to the previous point and then all points inside the tube are deleted from the cross section. The tube then moves on to the next point. ***The final water tin is created from the ground tin and therefore the boundary string is located using the unfiltered section.***

Manning's n Real box

Manning's n values for the channel sections.

Initial depth Real box

This depth is added to the minimum elevation on the cross section and is used as the starting water level for the cross section.

Units Choice box

This selection will set the default units for the project being created.

Project file name Input box

The MIKE11 project name. All of the MIKE11 files will begin with this name and the appropriate extensions added.

21.5.2 Running MIKE11

Three steps are required to run MIKE11 with the files 12d creates.

1. Create you time series files.
2. Inside MIKE11, create a new cross sections file and import the cross sections.
3. Open the simulation file, and load the network file to have the grid points calculated.

21.5.3 Creating Time Series Files

Your time series files must be named with the prefix of the river string name. For example if your centre line string in 12d was named *centre line Major* River your time series files need to named *Major River-H.DFS0* for the tail water conditions and *Major River-q.dfs0* for your discharges.

The standard time series dates are from 12:00 to 12:30 on 01 January 2000 with a one minute time step. If other periods are desired, you can either change the file *sim11-period.4d* in the 12d library before running the interface or change the dates inside MIKE11 after you read in the data. **DO NOT USE THE ORIGINAL FILES! Copy the file you are changing into the 12d user directory and modify it there. 12d will look for the file here first.**

21.5.4 Importing Cross Sections

From the MIKE11 main menu select **File->new** and then under Mike11 select **cross sections** from the dialogue box.

From the main menu select **File->Import->Import Raw data & Recompute**. Select the *.txt file with the **Project file stem** you specified in the 12d-Mike11 Write Panel. Now save this file with the same **Project file stem** (MIKE11 adds the .xns11 extension).

21.5.5 Calculating Grid Points

From the main menu select **File->Open** and select the *.sim11 file with the **Project file stem** you have specified in 12d. On the **Input** tab property sheet select **Edit** beside the **Network** file. Press **Ctrl+T** to take you into the table editing mode and then select the **Grid Points** tab property sheet. On the sheet select **Generate Grid Points** and then save the file.

You should now get the “Green lights” on the **Start** property sheet of the simulation file editor.

21.5.6 MIKE 11 Read Panel

Position of option on menu: Design=>Rivers=>Mike11 interface=>Read Mike11 results (max)
After the MIKE 11 analysis is complete the water level results are read back into 12d. Water level strings are created with the plan shape of the cross sections at the elevation retrieved from the MIKE 11 results (maximum water level).

Water levels are interpolated to create water level strings at the shape string locations. Note that water levels are extended when the shape strings are in a junction area or past the end of a reach. These strings are then triangulated to create a water surface tin.

The MIKE11 executable file, *res11read.exe* must be found in the directory *c:/mikezero/bin* so that 12d can read the Mike11 binary data files.

See also

MIKE11 Interface Overview

Presenting Water Level Results

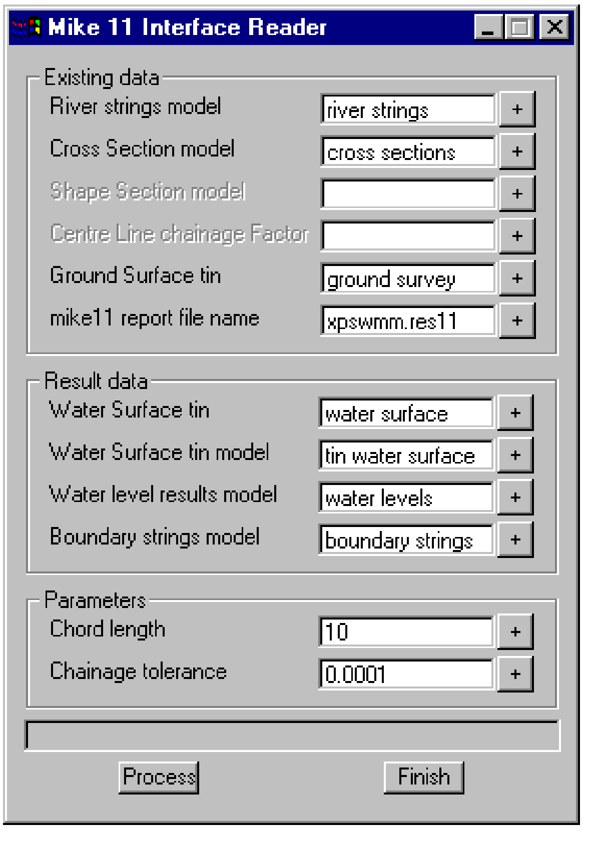
MIKE 11 Write Panel

How to for Rivers

Frequently Asked Questions (Rivers)

Usage

The MIKE 11 read panel follows.



The fields and buttons used in this panel have the following functions:

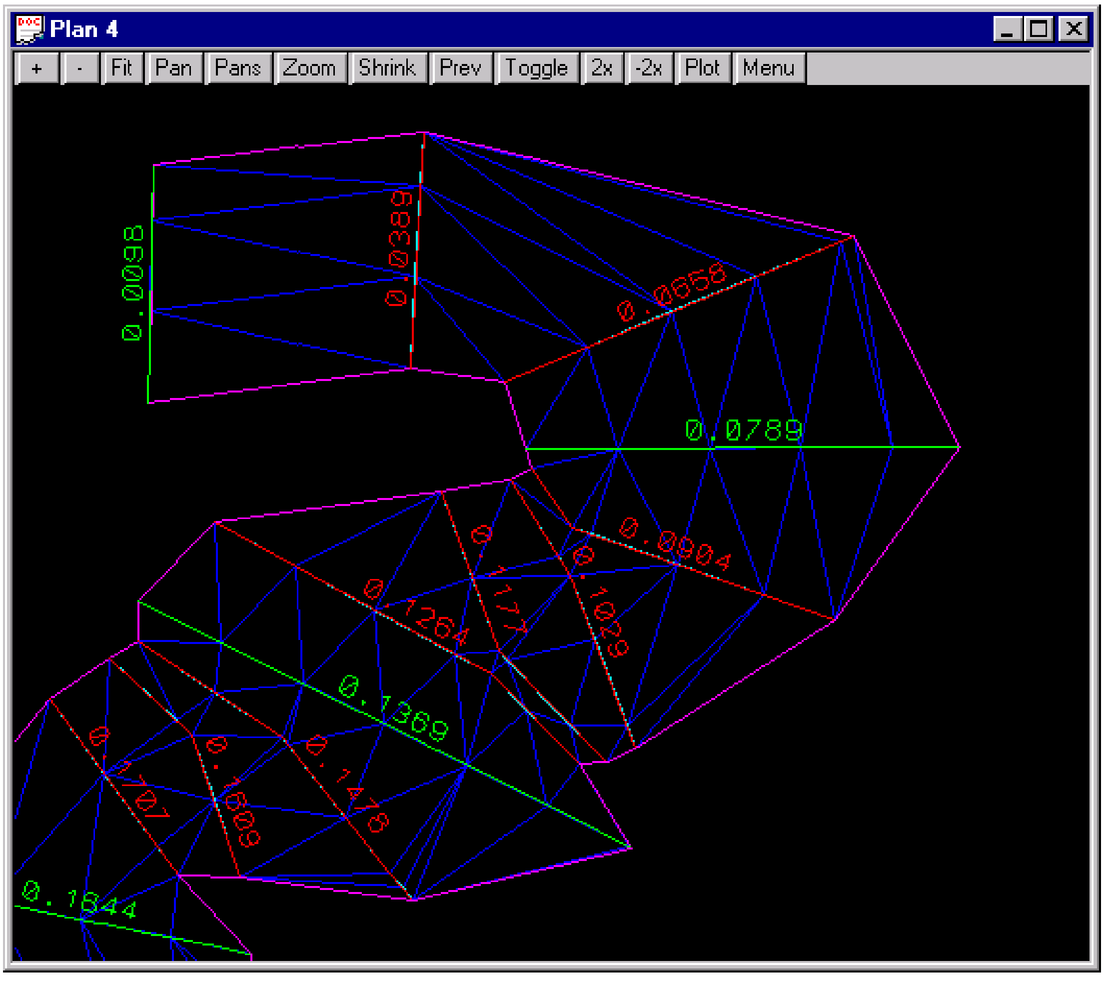
Field Description	Type	Defaults	Pop-Up

River strings model Model box

The river strings model specified in the write panel.

Cross Section model Model box

The cross section model specified in the write panel. The interface will search the string names in this model for the cross sections specified in the MIKE 11 report. A match is successful if the MIKE 11 cross section chainage and the string name are within the tolerance specified below in ***Chainage tolerance****.*



Shape string model Model box

For meandering rivers, the cross sections (shown in green above) may not be at a close enough spacing to create a water surface that follows the river. 2D shape strings (shown in red above) can be created to create a water surface (shown in blue above) to follow the river. Note that water levels are extended when the shape strings are in a junction area or past the end of a reach.

Centre line chainage factor Model box

This data is only required if the ***Shape string model*** *is used. The shape string names are created by dividing the chainage on the centre line by this factor. Typically 1000 is used to convert metres to kilometres and 5280 to convert feet to miles.*

Ground surface tin Tin box

If a ***boundary string model*** *is specified below, the intersection of this ground surface and the water surface will be determined. The strings will be stored in the model from the* ***boundary string model*** *field. Super tins cannot be used for this function. A composite tin is required for the tin-tin intersect.*

MIKE 11 report file name Input box

This is the binary data file that will converted to an ASCII text file and read by 12d. The maximum water levels at each cross section will be extracted from the file.

Water surface tin Tin box

The name of the water surface tin to be created.

Water surface tin model Model box

The model to contain the new water surface tin.

Water level results model Model box

The model where the water surface strings will be created at each cross section and shape string.

Boundary string model Model box

The model to contain the intersection strings between the water and ground surfaces specified above. If left blank no intersection strings will be calculated.

Chord Length Model box

This value set the spacing for the points on the water level strings (both cross section and shape strings). It is recommended that you use a length of no more than half of your average cross section and shape string lengths. A large value in this field may result in unexpected water level profiles for meandering rivers.

Centre Line Chainage Factor Real box

The cross section names are created by dividing the chainage on the centre line by this factor. Typically 1000 is used to convert metres to kilometres and 5280 to convert feet to miles.

Chainage Tolerance Real box

This is the tolerance used when the cross section chainage from the MIKE 11 results file is compared with the cross section string names. If a ***Centre line chainage factor*** *of 1 is used a value of 0.1 is appropriate. If a* ***Centre line chainage factor*** *of 1000 or 5280 is used a* ***Chainage*** *tolerance of 0.00001 is more appropriate.*

However, if you have altered chainage names then you may have to increase the value of the tolerance. Suppose the tolerance is set to 0.001 and the water level for section 0.056 is read from the MIKE 11 report file. The interface will search for the first string with a name between 0.055 and 0.057. If you chose to great of a tolerance then more than one water level result will match a 12d cross section and a warning message will be given.

If you have one specific cross section that you would like to have a different tolerance set for (maybe only one section is giving you troubles), use the Attribute Editor (Strings->User->Attribute Editor), select the cross section string and create a real type attribute named tolerance set to the tolerance desired.

21.6 Presenting River Water Level Results

Topics

See also

River and Source Strings

XP-SWMM Interface

HEC-RAS Interface

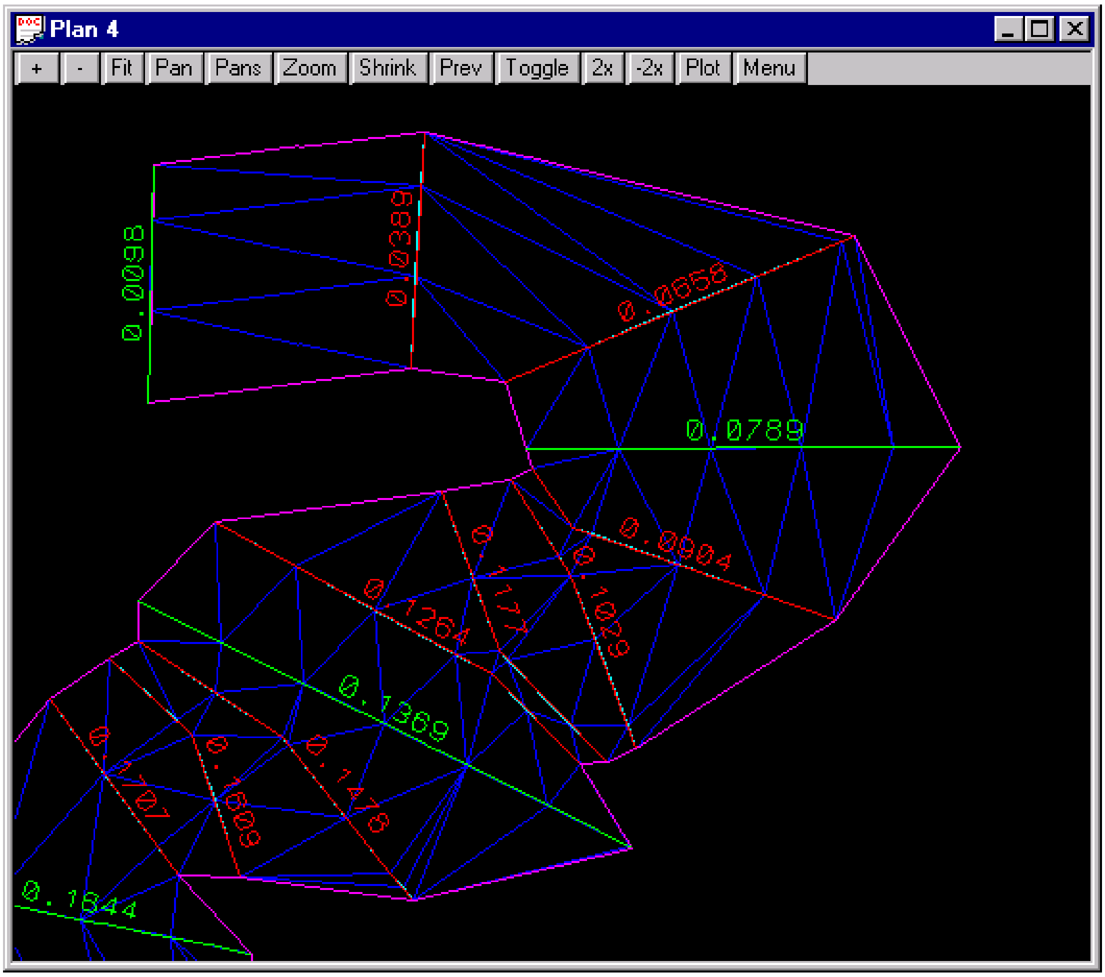
MIKE11 Interface

ISIS Interface

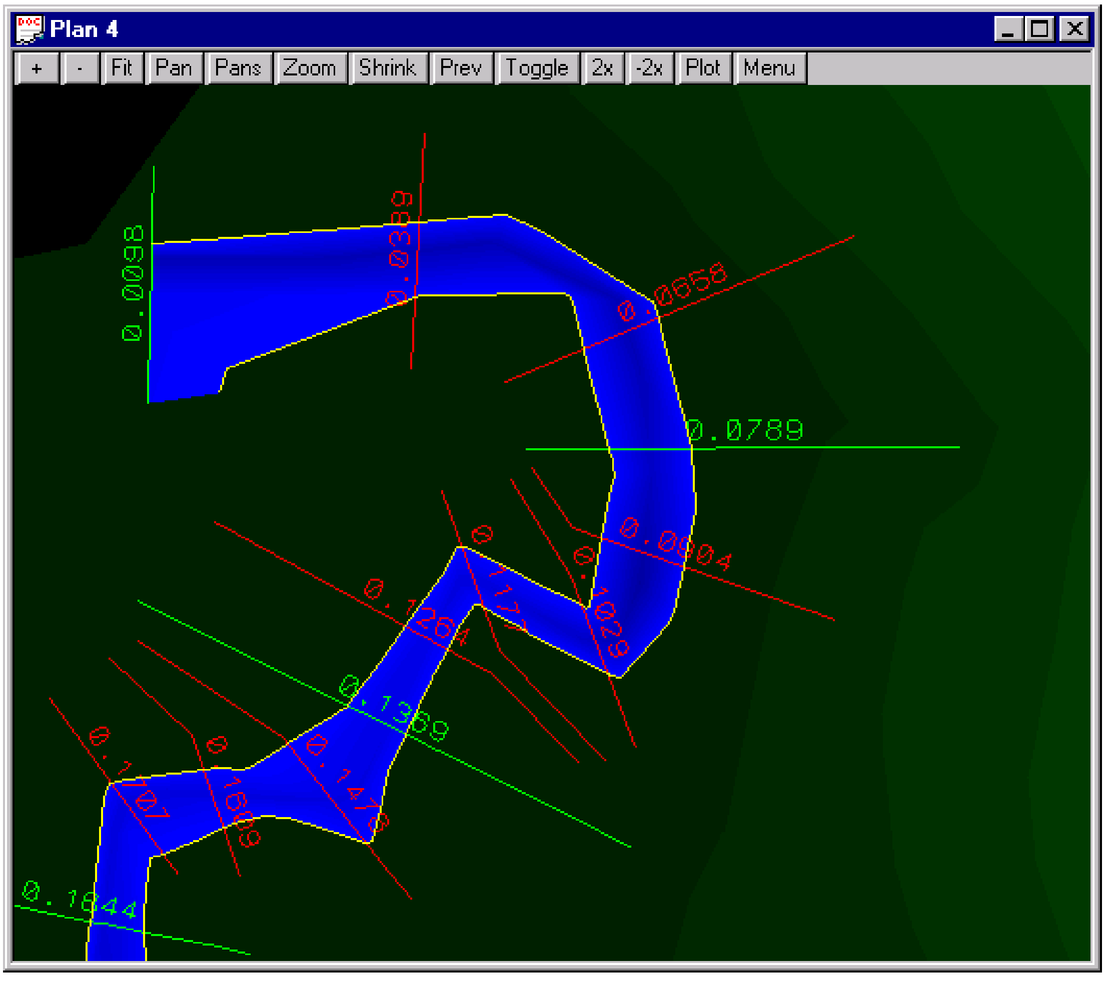
How to for Rivers

Frequently Asked Questions (Rivers)

21.6.1 How the water level boundaries are determined



For meandering rivers, the cross sections (shown in green above) may not be at a close enough spacing to create a water surface that follows the river. 2D shape strings (shown in red above). Note that water levels are extended when the shape strings are in a junction area or past the end of a reach. can be created to create a water surface (shown in blue above) to follow the river.



The water surface is draped over the ground surface and the boundary strings (strings defining the edges of the water surface – shown in yellow above) are created. They may be used to trim the water surface or shade your ground surface tin for flood inundation mapping. Boundary strings also include islands! Shading the river bed blue, in a 3D perspective view, is an effective way to show the water level extents and still view the shape of the river bottom (it has the effect of very clean water that you can see through!).

21.6.2 Sample Presentations and Drawings

The water surface may be

s •contoured (elevation),

s • depth contours created,

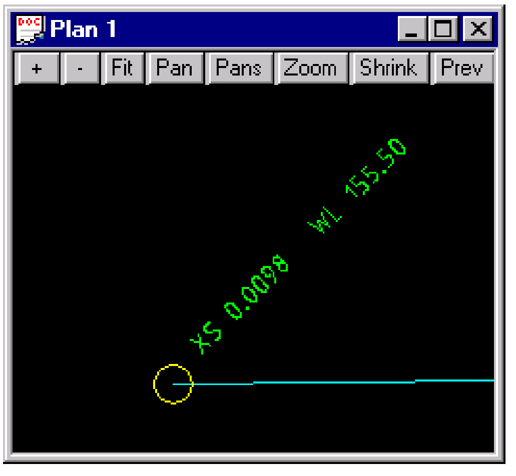
s • water surface coloured by depth (shown above),

s • cross sections plotted

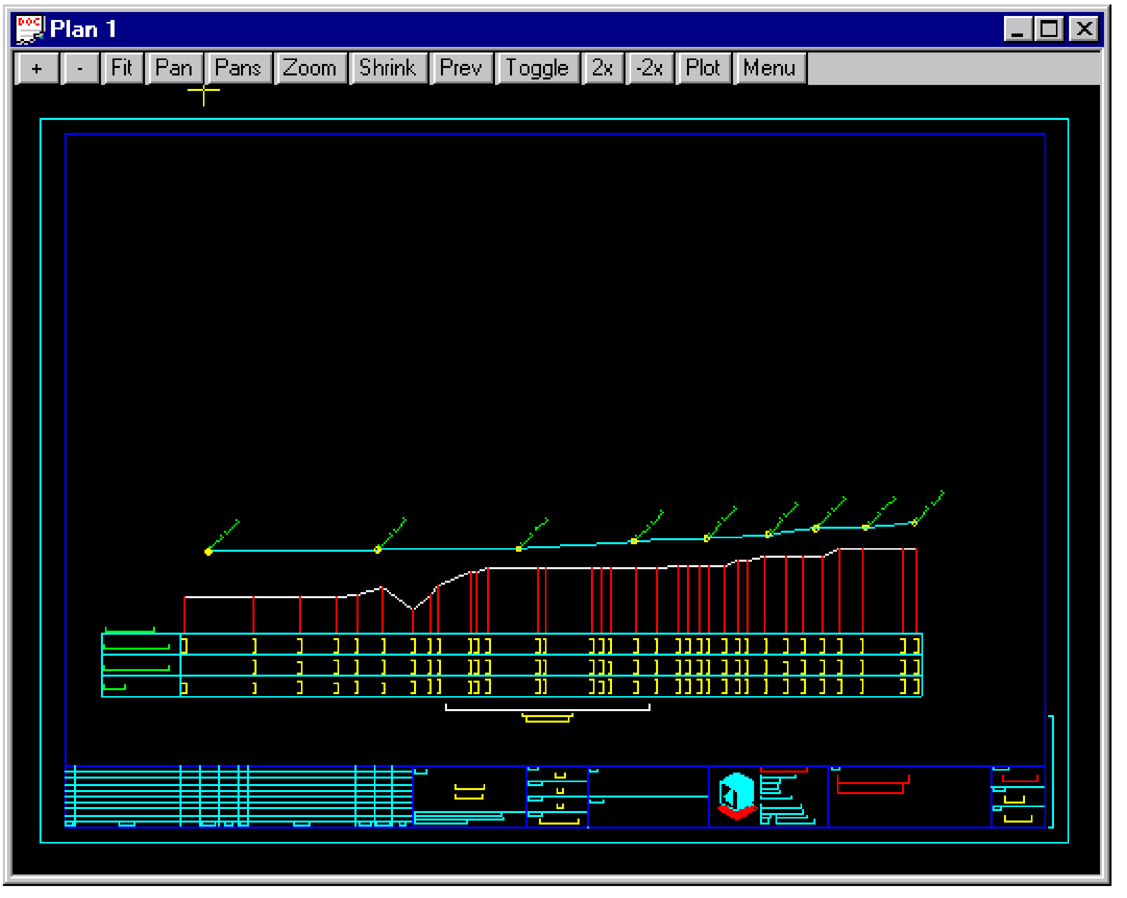
s • and longitudinal profiles drawn (shown below).

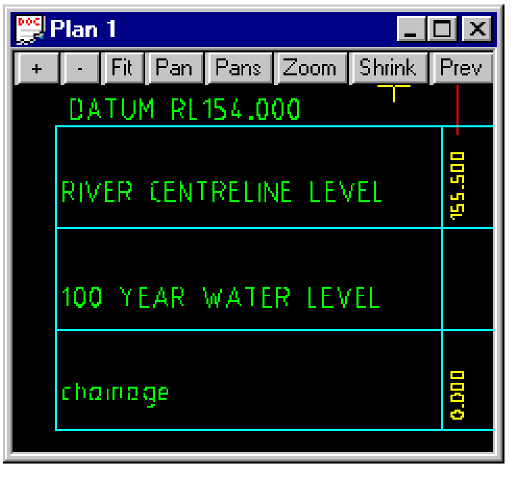
s •All of these results can be plotted complete with your customised drawing sheets.

s •Finally, you may walk down the water course in the perspective view (and record this to an Windows AVI file).

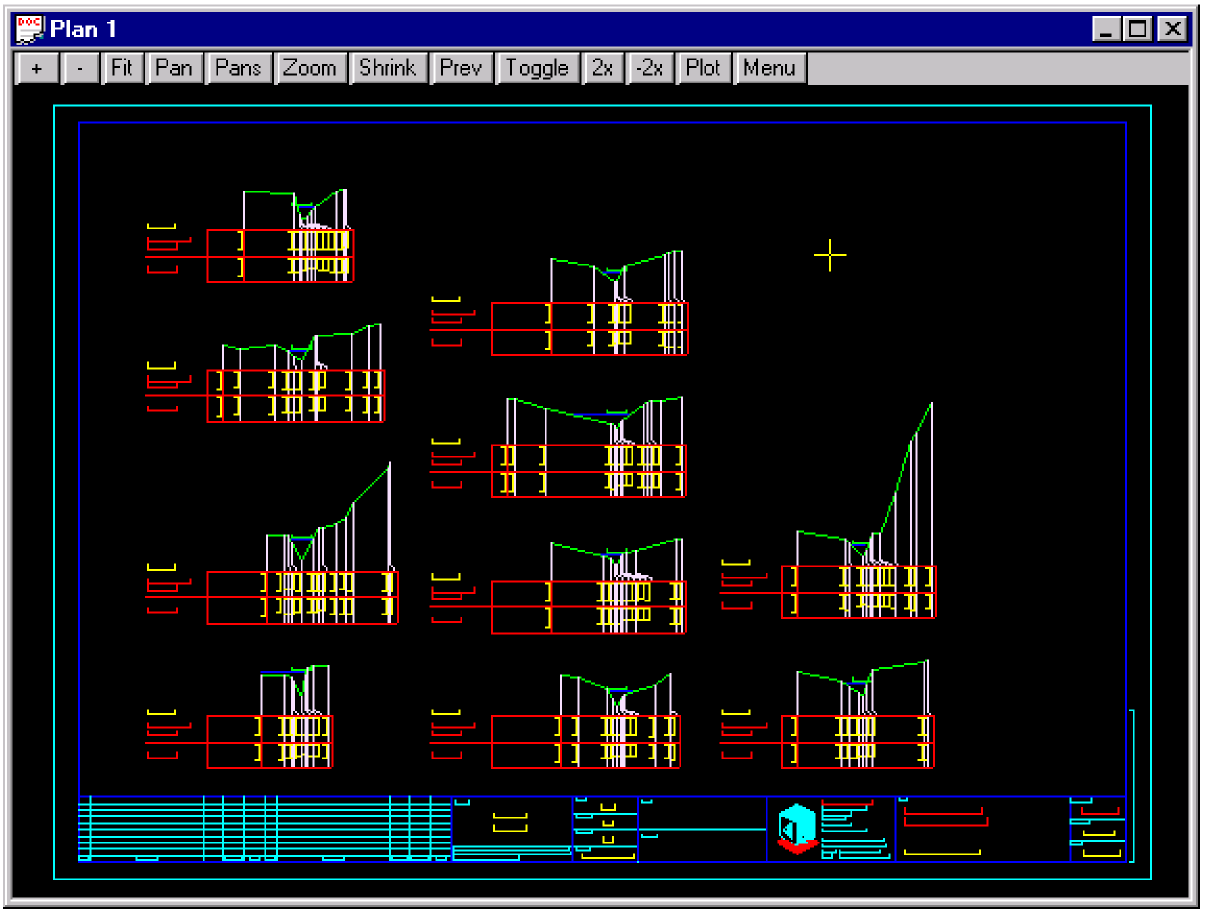


Close up view of cross section labels with water levels

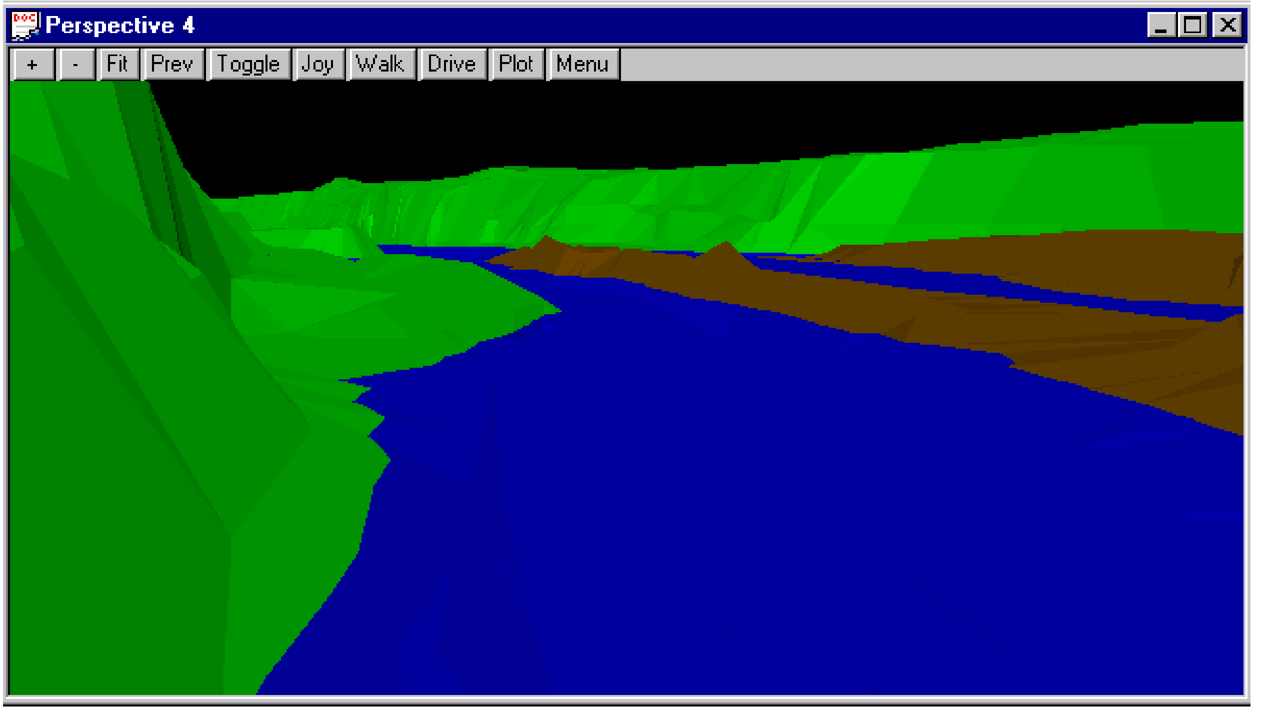

Plan 1
+
-
Fit
Pan
Pans
Zoom
Shrink
Prev
Toggle
2x
-2x
Plot
Menu


Plan 1
+
-
Fit
Pan
Pans
Zoom
Shrink
Prev
DATUM RL154.000
RIVER CENTRELINE LEVEL
155.500
100 YEAR WATER LEVEL
chainage
0.000

Close up of text (user chooses wording, size, colour, text style etc.)



The tin created can be viewed in a perspective view and sections taken where desired. A sample perspective view follows.



Example of cross sections long sections and depth colouring see the HEC-RAS Interface topic.

21.6.3 Defining the Water Surface Boundaries

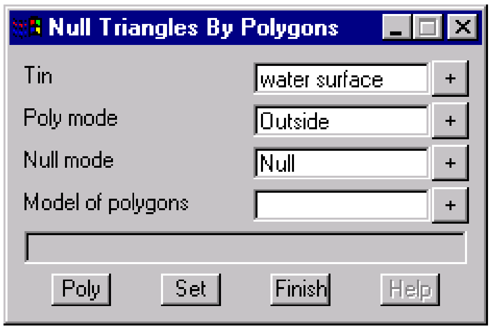
The water surface and its boundary is created by 12d. The first step is to trim the water surface back to the boundary strings. Since the water boundary does not generally form a closed polygon (the left and right river boundaries will need to be joined at one end of the river) use the **Strings=>Strings Edit=>Join** to connected strings.

If the water surface reached the edges of you cross sections then there will be numerous breaks in the boundary string. The best solution is to extend the cross sections and/or add additional shape strings. If it is not a major error in modelling then the break in the boundary can be joined using the **Strings=>Strings Edit=>Join.**

If boundary strings are created outside that water level boundaries they should not be used in this area. Boundary strings should only be used inside the area defined by your water level results.

21.6.4 Trimming the Water Surface Tin and Islands

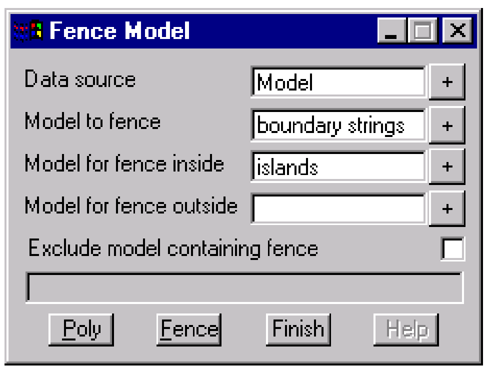
Use the **Triangle=>Null=>by polygons** selection to null triangles inside the polygon you have created above.



Select your water surface tin and change the **Poly mode** to **Outside**. Now select the **Poly** button and pick the boundary string. The triangles outside the polygon will now be nulled.

If you have some islands in the model then change the **Poly mode** to **Inside** and select the islands.

If you have numerous islands, say more than 10 it may be easier to copy all of the islands into one model using the fence command (Utilities->Fence->Fence).

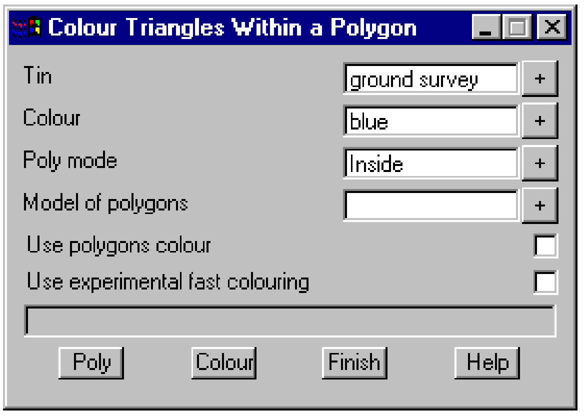


The **Model to fence** is the boundary strings model and the **Model for fence inside** is the new model to contain the islands. The **Exclude model containing fence** should not be ticked. Select **Poly** and then pick the boundary string. All of the islands inside the boundary string will not be copied to the islands model.

21.6.5 Colouring the Ground Surface with Flood Zones

The boundary strings can also be used to colour the ground surface. Colouring the river bed blue is an effective way to show the water level extents and still view the shape of the river bottom (the effect of very clean water that you can see through!).

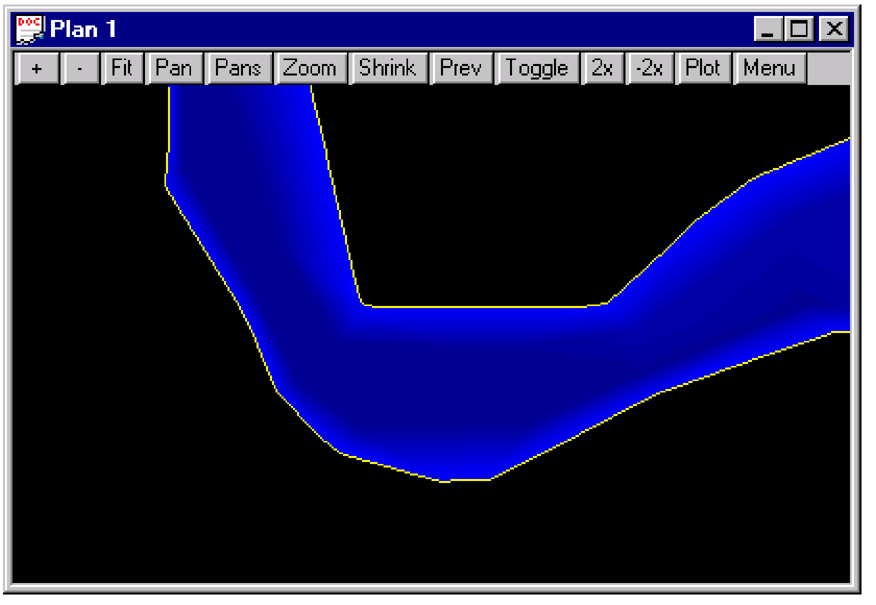
First, define the water surface boundaries (see above). Next re-triangulate the ground tin so that it includes the boundary string model (**Triangles=>Edit=>Tin**). Next select **Triangles=>Colour=>Colour within polygon**. The following panel will appear.



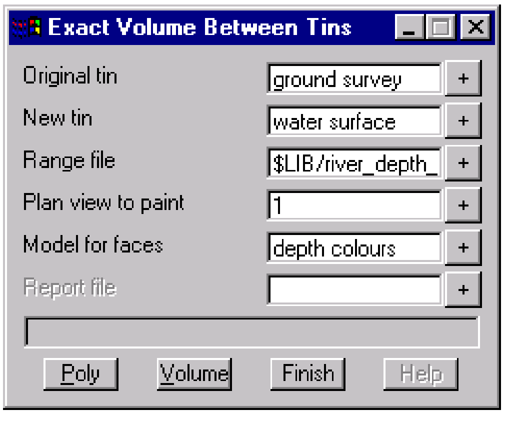
Select your ground tin and the desired colour. Next select the **Poly** button and pick the boundary string to be coloured inside. If islands exist change, change the colour (to a ground colour) and then select the island string.

21.6.6 Colour by Depth

The water surface can be coloured by depth. This function calculates the depth between the water surface and the ground surface and creates “faces” of different colours. The colours to be used are specified in a depth range file. This option requires the purchase of the Volumes Option



From the main menu select Options->Volumes->Exact->Tin to tin.

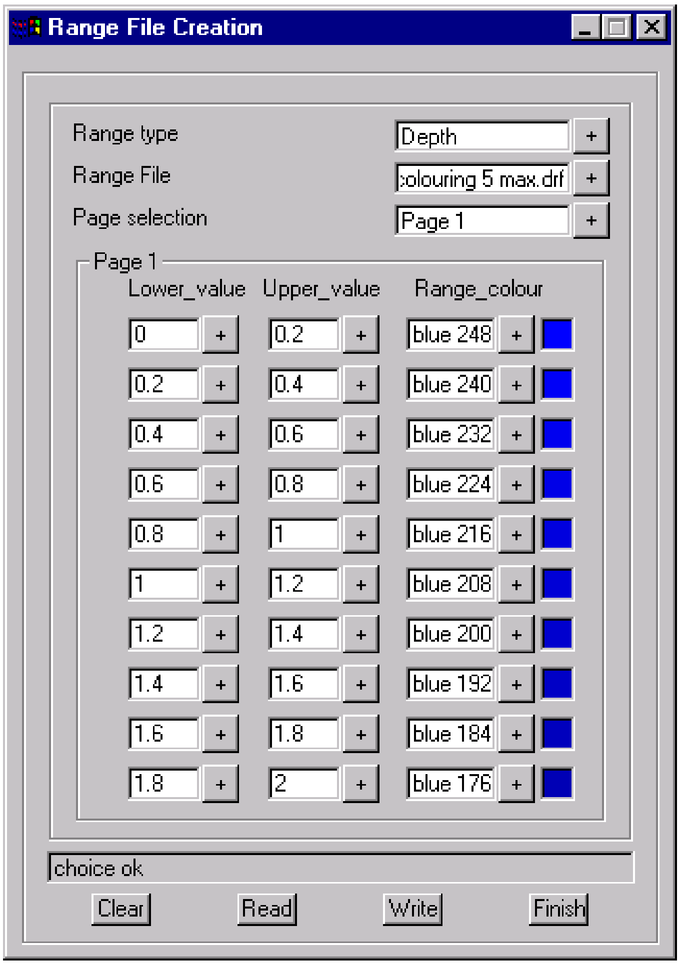


Original tin this is your ground survey tin

New tin this is your water surface tin

Range file two range files are supplied. One with a range from 0 to 5 and another from 0 to 50. The library contains a spreadsheet the can quickly create other range files or you may edit the using the built in range editor.

To use the built in range file editor select **+** beside you range file then **edit**.



You may change the range value and the colours as desired. Be sure to select **Write** to save the changes before selecting **Finish.**

See also Range File Creation

Plan View to paint you can paint a current view without saving the face data. This is a good option if you wish to take a quick look at the depth colours in one area.

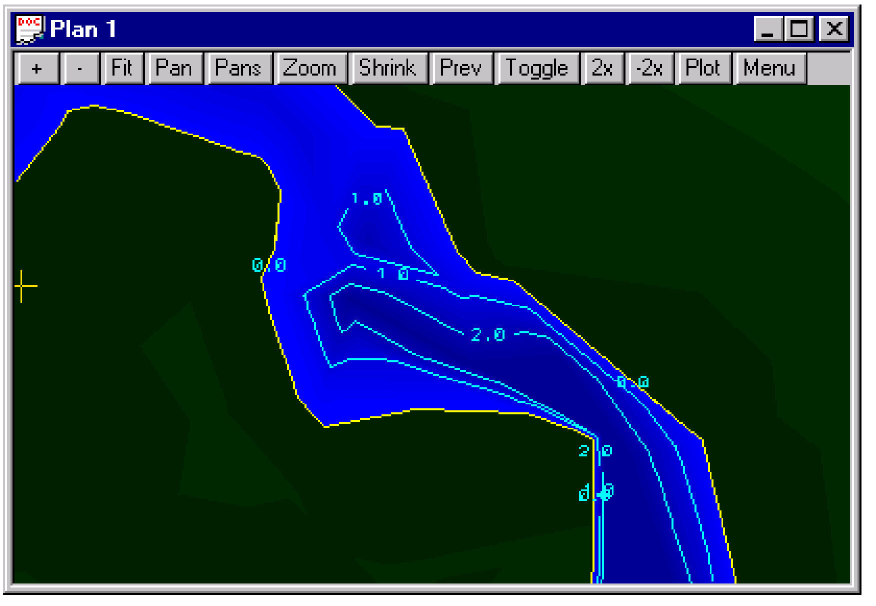
Model for faces the faces can be stored in a model. Note that faces consume a great deal of hard disk space. Therefore you may consider colouring one area at a time using the **Poly** option.

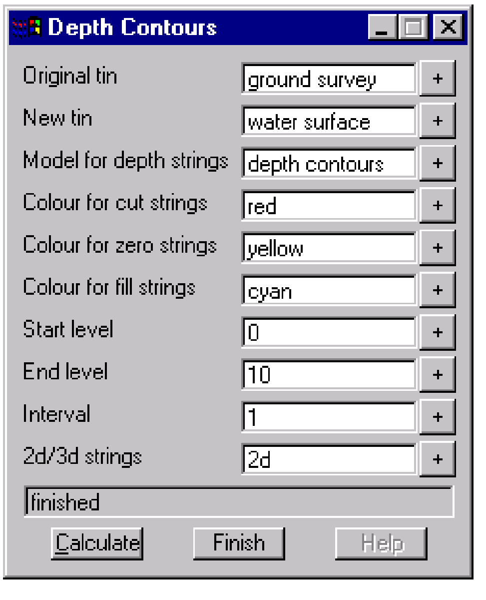
Poly If your water surface tin is very large than you may want to only colour a portion of the tin. You must create a polygon (Strings->Create->2d) and then pick this polygon. When your select **Volume** only the area inside the polygon will be coloured.

Volume select this button to colour the surface.

21.6.7 Depth Contours

Once you have the tin coloured by depth you might want to add depth contours. From the main menu select **Options->Tin->Analysis->Depth Contours**. This option requires the purchase of the tin Analysis module.





Original tin this is your ground survey tin

New tinthis is your water surface tin

Model for depth strings this is the model for your depth contours

Colour for cut strings this must be selected but we will not use this colour

Colour for zero strings the zero string will be the same as our boundary strings

Colour for fill strings this is the colour for the depth contours

Start level enter a zero for this value

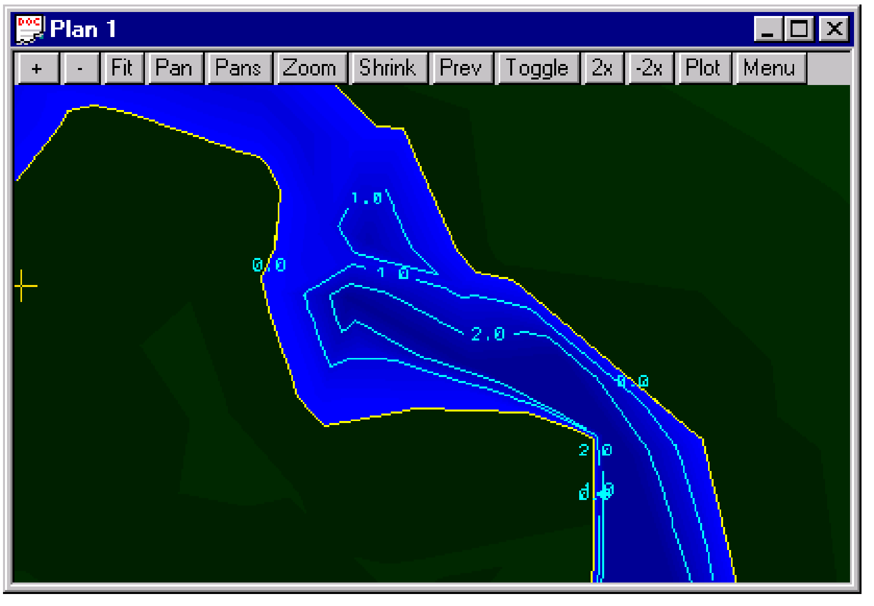
End level enter a level grater than the greatest depth

Interval enter the contour interval as desired

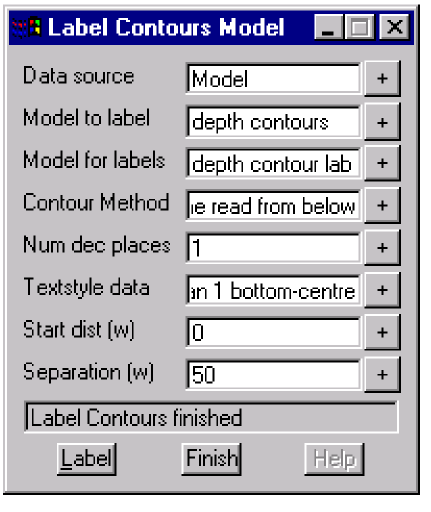
2d/3d strings 2d strings

Calculate this will create you contour strings

Adding Values to the Contours



Since there are numerous ways to label the contours it is performed as a separate step. From the main menu select **Strings->Label->Contours.**



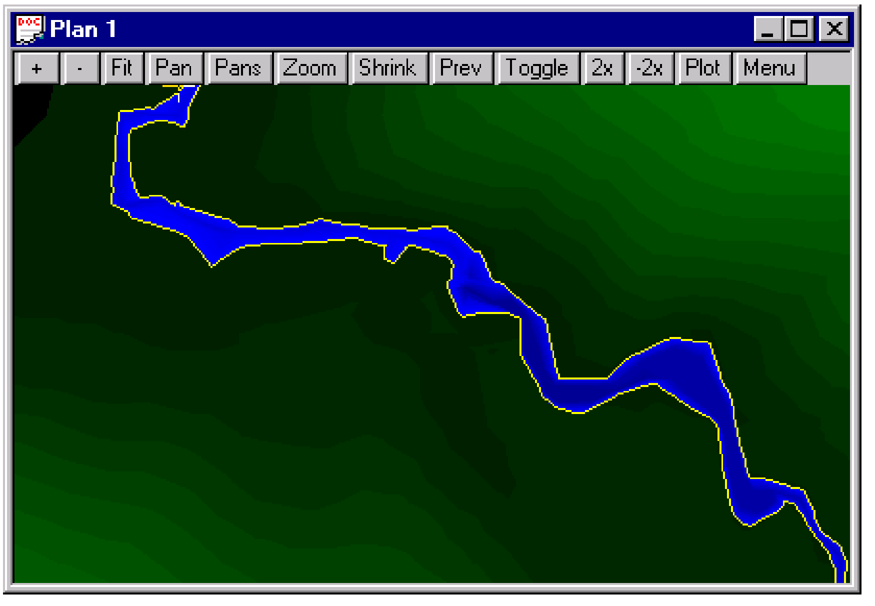
Contour method the example above uses *Line removal and Centred line read from below.* This copies the contour lines themselves and inserts a break in the line.

Start distance this is usually left as zero. It changes the start point for the first label on each contour.

Separation this determines the spacing of the contour labels.

21.6.8 Colour the ground surface by elevation

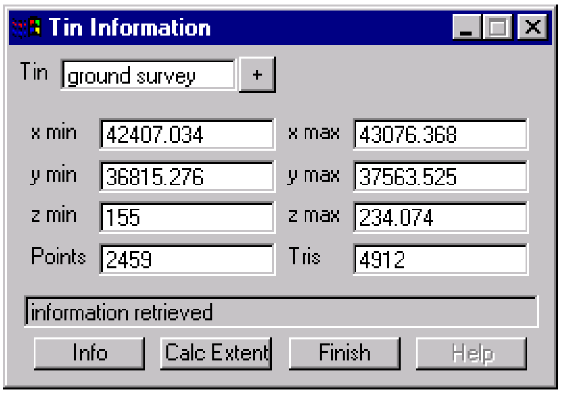
With the water surface coloured by depth, you may want to colour the ground surface by elevation.



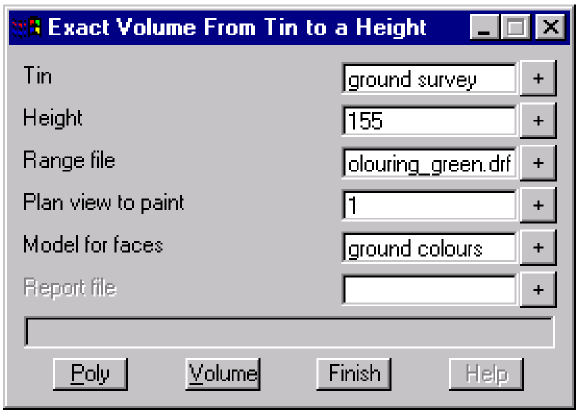
This colouring may be done from either From the main menu select **Options->Volumes->Exact->Tin to height** or **Triangles->Colour->Tin height colour.**

Options->Volumes->Exact->Tin to height

Options->Volumes->Exact->Tin to height requires the purchase of the Volumes module. First select **Triangles->Tin info** from the main menu and then select you ground surface tin. You will want to copy the minimum z level from this pane.



Now from the main menu select **Options->Volumes->Exact->Tin to height**



Select your ground surface to colour and enter the minimum elevation (from above) into the **Height** field. The **Range file** "*$LIB/ground_colouring_green.drf*" is found in the library.

Plan View to paint you can paint a current view without saving the face data. This is rarely used in this case

Model for faces the faces can be stored in a model. Note that faces consume a great deal of hard disk space. Therefore you may consider colouring one area at a time using the **Poly** option.

Poly If your ground surface tin is very large than you may want to only colour a portion of the tin. You must create a polygon (Strings->Create->2d) and then pick this polygon. When your select **Volume** only the area inside the polygon will be coloured.

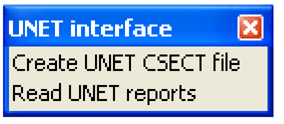
Volume select this button to colour the surface.

21.7 UNET Interface

Position of menu: **Design=>Rivers=>UNET interface**

The UNET interface creates the *.cs file. Water levels are read back into 12d where they may be viewed in a 3D perspective view to easily identify extents of flooding.

The UNET walk right menu is,



See also

River and Source Strings

Reservoir Strings

Create UNET files

Read UNET results

Presenting Water Level Results

How to for Rivers

Frequently Asked Questions (Rivers)

21.7.1 UNET Write Panel

Position of option on menu: **Design=>Rivers=>UNET interface=>Create UNET CSECT file**

The UNET interface creates the UNET *.cs file. Water levels are read back into 12d where they may be viewed in a three dimension perspective view to easily identify extents of flooding.

See also

River and Source Strings

UNET Interface Overview

Read UNET results

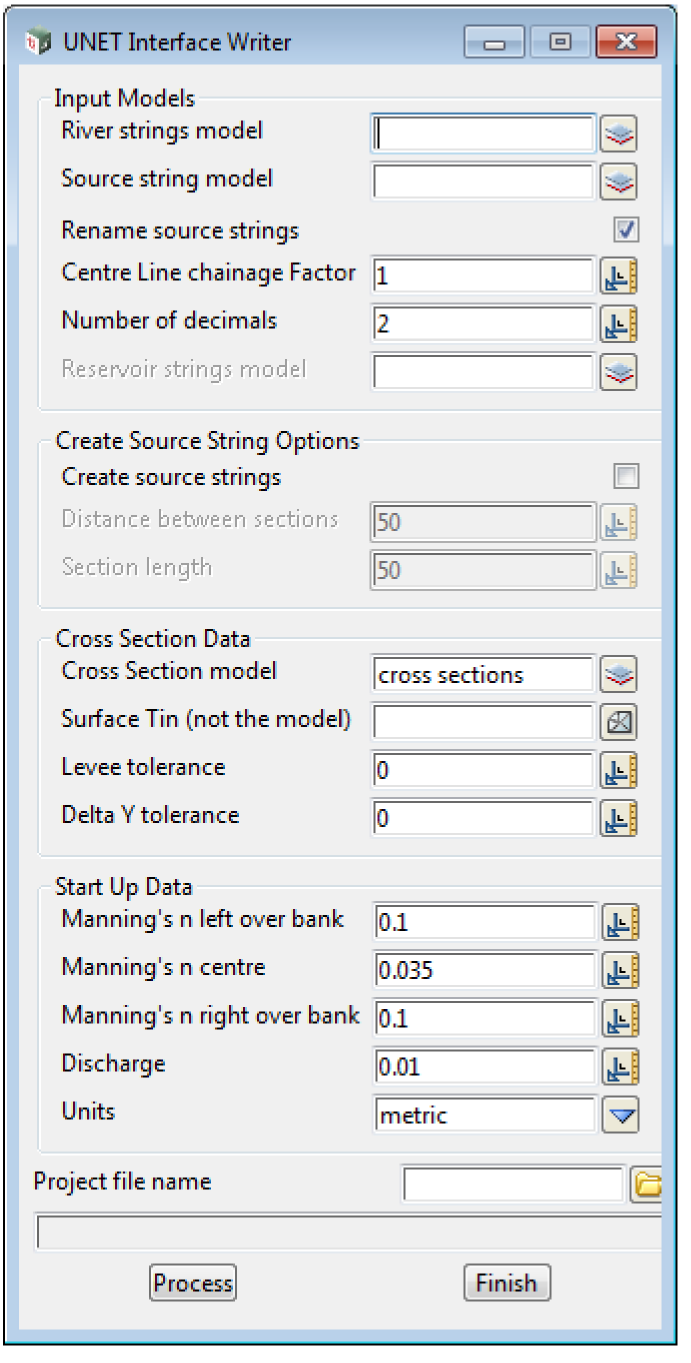
Presenting Water Level Results

How to for Rivers

Frequently Asked Questions (Rivers)

Usage

The UNET panel for creating the UNET project follows.


UNET Interface Writer
Input Models
River strings model
Source string model
Rename source strings
Centre Line chainage Factor
1
Number of decimals
2
Reservoir strings model
Create Source String Options
Create source strings
Distance between sections
50
Section length
50
Cross Section Data
Cross Section model
cross sections
Surface Tin (not the model)
Levee tolerance
0
Delta Y tolerance
0
Start Up Data
Manning's n left over bank
0.1
Manning's n centre
0.035
Manning's n right over bank
0.1
Discharge
0.01
Units
metric
Project file name
Process
Finish

21.7.2 UNET Read Panel

Position of option on menu: **Design=>Rivers=>UNET interface=>Read UNET reports**

After the UNET analysis is complete the water level results are read back into 12d. Water level strings are created with the plan shape of the cross sections at the elevation retrieved from the UNET results. These strings are then triangulated to create a water surface tin.

See also

UNET Interface Overview

Create UNET files

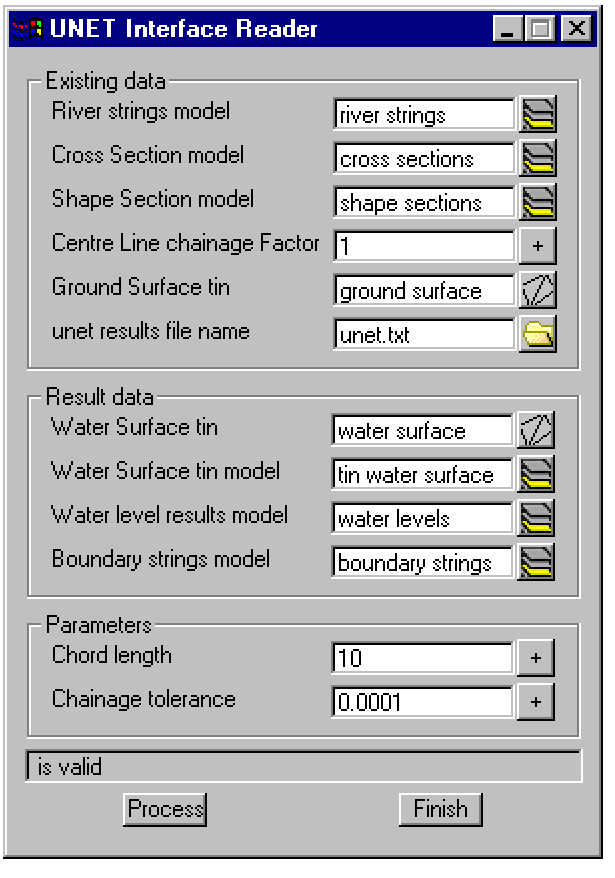
Presenting Water Level Results

How to for Rivers

Frequently Asked Questions (Rivers)

Usage

The UNET read panel follows.



The UNET file format consists of a line number, section name and elevation separated by at least 1 space. An example follows.

1	0.25	118.24
2	0.50	118.25
3	0.75	118.30
4	1.00	118.40

21.8 ISIS Interface

Position of menu: **Design=>Rivers=>ISIS interface**

The ISIS interface creates *.dat input file. Water levels are read back into 12d where they may be viewed in a 3D perspective view to easily identify extents of flooding.

The ISIS walk right menu is,

See also

River and Source Strings

Reservoir Strings

Spill strings

Create ISIS files

Read ISIS results

Presenting Water Level Results

How to for Rivers

Frequently Asked Questions (Rivers)

Exporting to ISIS

The ISIS project is created from a surface tin (representing the river bed and overbanks) and a model containing river strings identified by their names "left bank", "right bank and the name prefix, "centre line". Any additional strings in the specified model will be ignored (warning messages will be given when you run the macro that any additional strings are being ignored). The low chainage (often zero) of the centre line strings must be at the upstream end of the reaches.

Cross sections are created at the location of the source strings. These source strings are initially created using the ISIS option at a user defined spacing and section length. The user may alter these sections as desired. These may be shortened if they intersect at sharp bends in the river; they may be extended at extremely wide river sections or extra points may be added so that the section is no longer a straight line.

Source strings can be deleted and additional sections can be added by creating new source strings. The **Create source strings** tick box on the interface panel must **NOT** be selected to use the customised strings.

Reservoir Strings are 2d strings that define the extents of the reservoir. The volume is calculated in increments of 1 or the value set in the attribute stage increment for this string. The volumes start from a level with zero volume to the level set for the 2d reservoir string.

The **reservoir strings** may define inline reservoirs or offside storage. Inline reservoirs are "touched" by centre line strings both upstream and downstream. Offline storage areas are linked to the cross sections via **spill strings.**

Spill strings are 2d strings that link offline storage areas to a cross section. The string must begin by "touching" the source string and then proceed to the first point on the spill section. During the export the first point will NOT be exported as part of the spill section. After defining the end of the spill section the last point on the string must "touch" the reservoir string. Again this last point will NOT be exported as part of the spill section.

Presenting ISIS Results in 12d

After the ISIS analysis is complete the water level results are read back into 12d. Water level strings are created with the plan shape of the cross sections at the elevation retrieved from the ISIS *.zzr file. These strings are then triangulated to create a water surface tin from which the water level boundaries are determined. These results can then be shown in plan, long section, cross section and in 3D perspectives.

More details

21.8.1 ISIS Write Panel

Position of option on menu: **Design=>Rivers=>ISIS interface=>Create ISIS file**

The ISIS interface creates the ISIS project files ready to open and run. This includes the project, plan, flow and geometry files. Water levels are read back into 12d where they may be viewed in a three dimension perspective view to easily identify extents of flooding.

See also

River and Source Strings

ISIS Interface overview

Read ISIS results

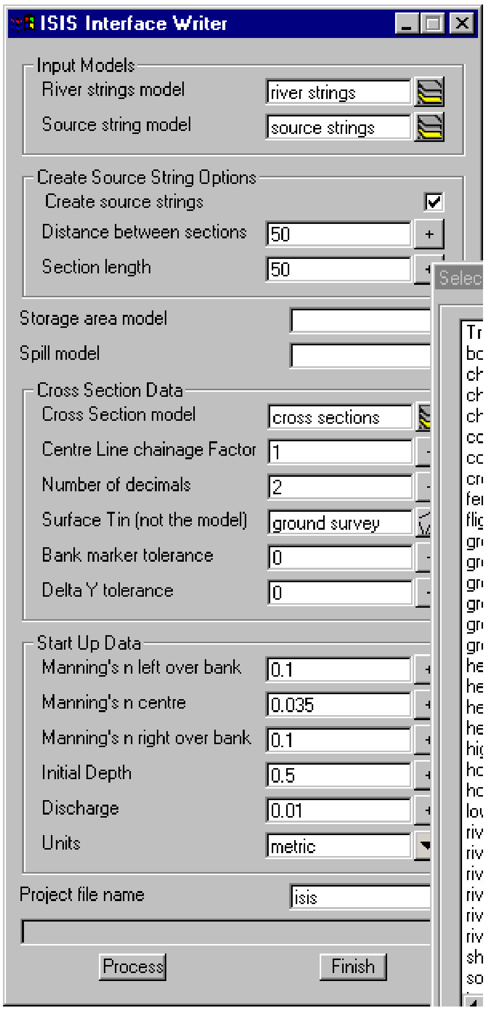
Presenting Water Level Results

How to for Rivers

Frequently Asked Questions (Rivers)

Usage

The ISIS panel for creating the ISIS project follows.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up

River strings model Model box

*Model containing the centre line, left bank and right bank strings. **The centre line strings must begin downstream and proceed upstream.** The name of the river may follow the words "Centre line " (note the ending space). For example the centre line string may be named "Centre line Parramatta River, downstream reach". The comma separates the river name from the reach name. If no comma is included then the river name is repeated for the reach name.*

Confluences are modelled by a using a separate string for all reaches. Thus a system with a branch is modelled with three strings. The branch will be one string and the main reach will have a downstream string and an upstream string. The reaches must touch at the confluence.

The distance from the start of the upstream strings to the first cross section is used to model the confluence length.

The left and right bank strings need not be separate strings (see figure below).

Source string model Model box

*New source strings will be created in this model or existing source strings are contained in the model. See **Create source strings** tick box below.*

Create Source Strings Tick box not selected

*When selected existing source strings are deleted and new ones created perpendicular to the centre line at the specified spacing and length. Once you have created the sources strings they can be easily modified. On the **Strings->Points Edit** menu you will find the selections **Move** (to move the end points), **Insert** (to insert additional points).*

Distance between sections Real box

The distance between the cross sections. At present no check is made for overlapping cross sections around river bends.

Section Length Real box

The length of the cross section with zero chainage at the mid point.

Storage strings Real box

These 2d strings define the extents of the storage area. For more details see Reservoir strings.

Spill strings Real box

These 2d strings define the location to cut the spill section and start end points define the cross section and reservoir, respectively. For more details see Spill strings

Cross section model Model box

The cross sections created and exported are stored in this model.

Centre Line Chainage Factor Real box

The cross section names are created by dividing the chainage on the centre line by this factor. Typically 1000 is used to convert metres to kilometres and 5280 to convert feet to miles.

Number of decimals Integer box

*The cross section names are created with this many decimals. CAUTION XP SWMM only allows 10 characters for the names and each link name begins with "**Lx-**" That leaves 7 characters for the chainages.*

Surface Tin (not the model) Tin box

Tin or super tin to create the cross sections from (remember a tin is like a string. It is placed in a model.).

Bank Marker Real box

Not currently implemented in ISIS

Delta Y tolerance Real box

This value filters out points on the cross section. Imagine a tube of this diameter passing over the cross section. The tube is elongated until one point lies outside the tube. The tube is shortened to the previous point and then all points inside the tube are deleted from the cross section. The tube then moves on to the next point. The filtered (smoothed) and original sections are kept for comparison. ***The final water tin is created from the ground tin and therefore the boundary string is located using the unfiltered section.***

Manning's n Real box

Manning's n values for the left, right and centre channel sections.

Initial depth Real box

Not currently implemented in ISIS.

Discharge Real box

This discharge is used at the upstream end of all reaches. If you have multiple river branches, you can set the flow for each branch inside ISIS or inside 12d. This can be changed at each section See manual settings

Units Choice box

This selection will set the default units for the project being created.

Project file name Input box

The ISIS project name. The extension ".dat" will automatically be added for you.

21.8.2 ISIS Read Panel

Position of option on menu: **Design=>Rivers=>ISIS interface=>Read ISIS results**

After the ISIS analysis is complete the water level results are read back into 12d. Water level strings are created with the plan shape of the cross sections at the elevation retrieved from the ISIS results. These strings are then triangulated to create a water surface tin.

See also

ISIS Interface overview

Create ISIS files

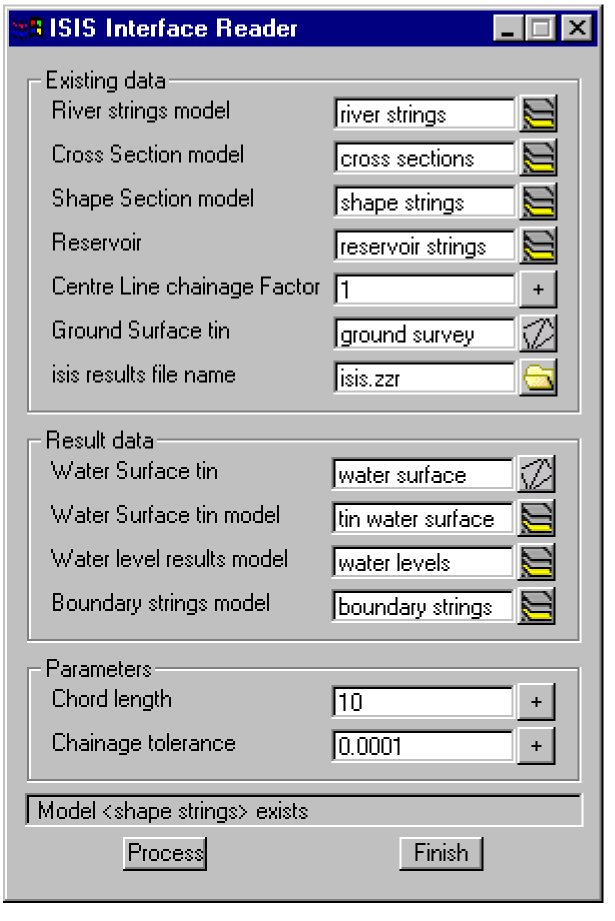
Presenting Water Level Results

How to for Rivers

Frequently Asked Questions (Rivers)

Usage

The ISIS read panel follows.



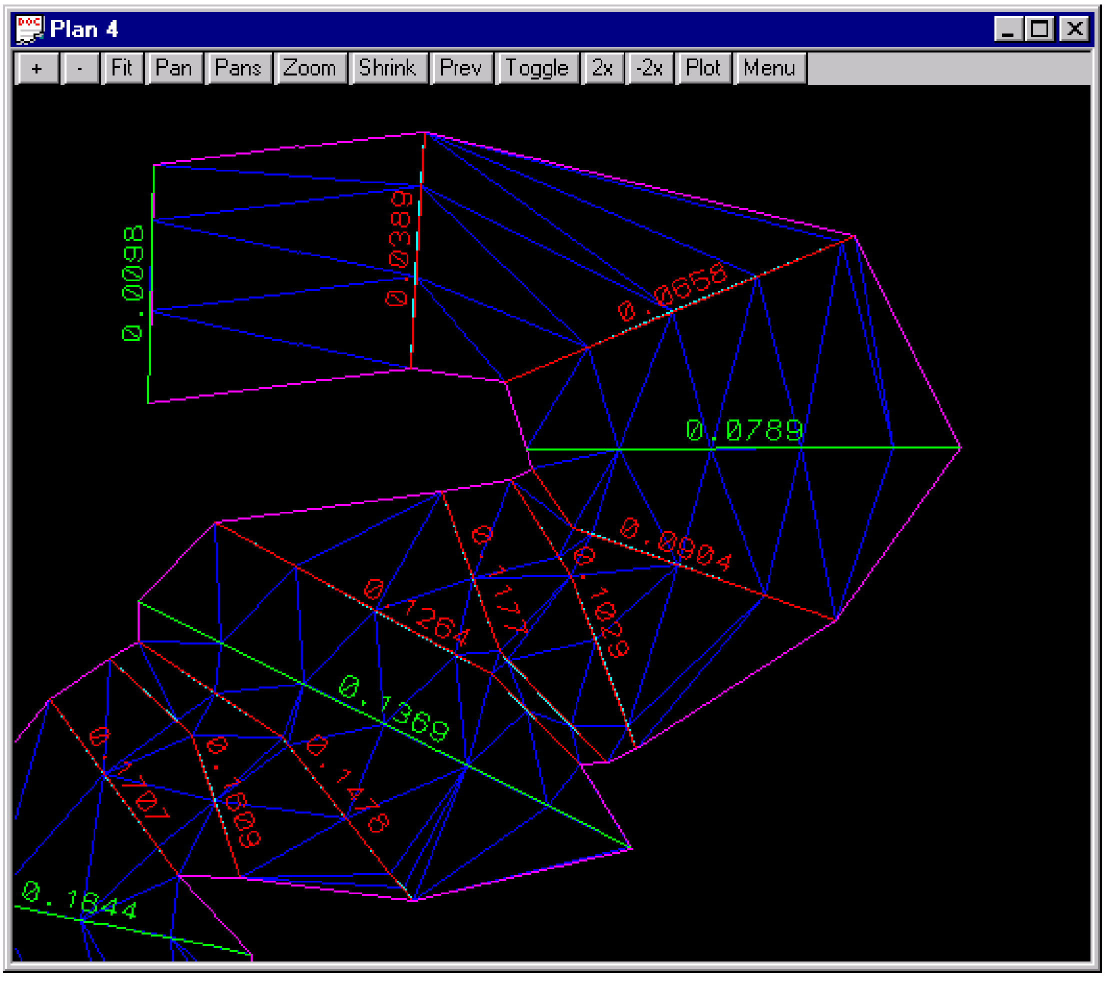
River strings model Model box

The same river strings model specified in the write panel. The river strings are used to sort the cross section and the shape strings. The left and right bank strings are used to check the direction of the strings.

Cross Section model Model box

The same cross section model specified in the write panel. The interface will match the ISIS label and *the string name (without the "SECT "). If the ISIS labels were created by 12d then an exact match should result.*

However, if the ISIS file was not created by 12d, the file can still be mapped by manually creating cross section strings (2d or 3d strings). See Mapping non 12d and historical water level data.



Shape string model Model box

For meandering rivers, the cross sections (shown in green above) may not be at a close enough spacing to create a water surface that follows the river. 2D shape strings (shown in red above) can be created to create a water surface (shown in blue above) to follow the river. Note that water levels are extended when the shape strings are in a junction area or past the end of a reach.

Reservoir Model box

Reservoir strings are assigned elevations from the ISIS zzr file. This model is the same as the ***storage area stings*** *specified in the write panel.*

Centre line chainage factor Model box

This data is only required if the Cross sections have been manually created and have no names yet. See Mapping non 12d and historical water level data.

ISIS results file name File box

The ISIS zzr file is automatically generated by ISIS.

Ground surface tin Tin box

*If a **boundary string model** is specified below, the intersection of this ground surface and the water surface will be determined. The strings will be stored in the model from the **boundary string model** field.*

Water surface tin model Model box

The model to contain the new water surface tin.

Water surface tin Tin box

The name of the water surface tin to be created.

Water level results model Model box

The model where the water surface strings will be created at each cross section and shape string.

Boundary string model Model box

The model to contain the intersection strings between the water and ground surfaces specified above. If left blank no intersection strings will be calculated.

Chord Length Model box

This value set the spacing for the points on the water level strings (both cross section and shape strings). It is recommended that you use a length of no more than half of your average cross section and shape string lengths. A large value in this field may result in unexpected water level profiles for meandering rivers.

Chainage Tolerance Real box

This tolerance is not used for the standard 12d cross section names. However, if the cross section names have been created manually and they can be converted to a real number then this is the tolerance used to match the cross section label from the ISIS report to the cross section string names. A value of 1.0 to 10. is common. See Mapping non 12d and historical water level data *for more details.*

A sample of the *.zzr file that is read by 12d follows.

A sample of the zzr file that is read follows.

Maxima and minima of all variables from

time 0.000 hours to time 30.000 hours

****

maxima of all variables

****

Label12		Flow	Stage	Froude no	Velocity	Umode	Ustate
WY201013	y	0.470	87.305	0.033	0.076	7.916	0.000
WY202013	y	0.470	87.288	0.162	0.297	2.335	0.000
WY203013	y	0.470	87.054	0.682	0.794	4.000	0.800
WY204013	y	0.470	86.804	0.671	1.002	0.562	0.000

21.9 How to for Rivers

This section lists specific tasks for the rivers interface. Worked examples are contained in the courses directory of 12jobs and training manuals are found in the documentation directory on the ***12d Model*** distribution CD.

21.9.1 Manual Override settings

Many of the automatic settings can be overridden using string attributes via the String Attribute Editor. To use the editor

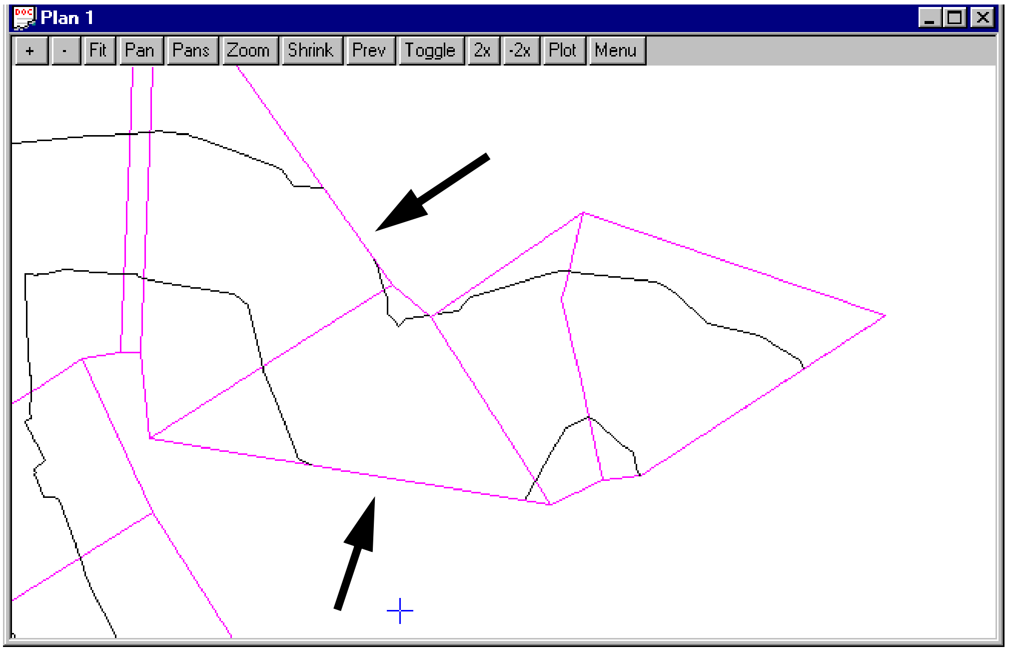
1. Select the river or source string to add/edit the attribute for
2. In the **Attribute Name** column use the selection list to find the attribute to change. If it is not listed then type the attribute name into on of the boxes.
3. Change the attribute **Type** to Integer/Real/Text as required
4. Type the attribute value into the **Data** field (erase the **not found** if required).
5. Select **Process, Next, Previous** or **Pick string.** Selecting **Finish** will NOT save the attribute.

You will be prompted that you will create a new attribute if you have typed in a new attribute for that string

Purpose	String Type	Attribute Name	Type	Typical Data Value
Node/cross section name	source strings	node name	Text	A1
Set a local inflow for this section	source string	flow	Real	10.2
Storage curve increment	storage area strings	stage increment	Real	1.0
Water level string tolerance when comparing numeric string names to data file names	cross sections before importing	tolerance	Real	0.01
Custom n values	source strings	left n, right n, centre n	Real	0.03
Culvert n value	centre line and culverts in spill	roughness	Real	.013
Culvert entrance loss	centre line and culverts in spill	entrance loss	Real	0.5
Culvert exit loss	centre line and culverts in spill	exit loss	Real	1.0
Culvert length	centre line and culverts in spill	length	Real	8.0
Multiple identical culverts	centre line and culverts in spill	number of pipes	Integer	2
Chanel length of a spill string	spill strings	length	Real	30
Channel roughness	spill strings	roughness	Real	0.02
Channel slope %	spill strings	slope	Real	0.5

21.9.2 Boundary Strings are broken. How do I stop this?

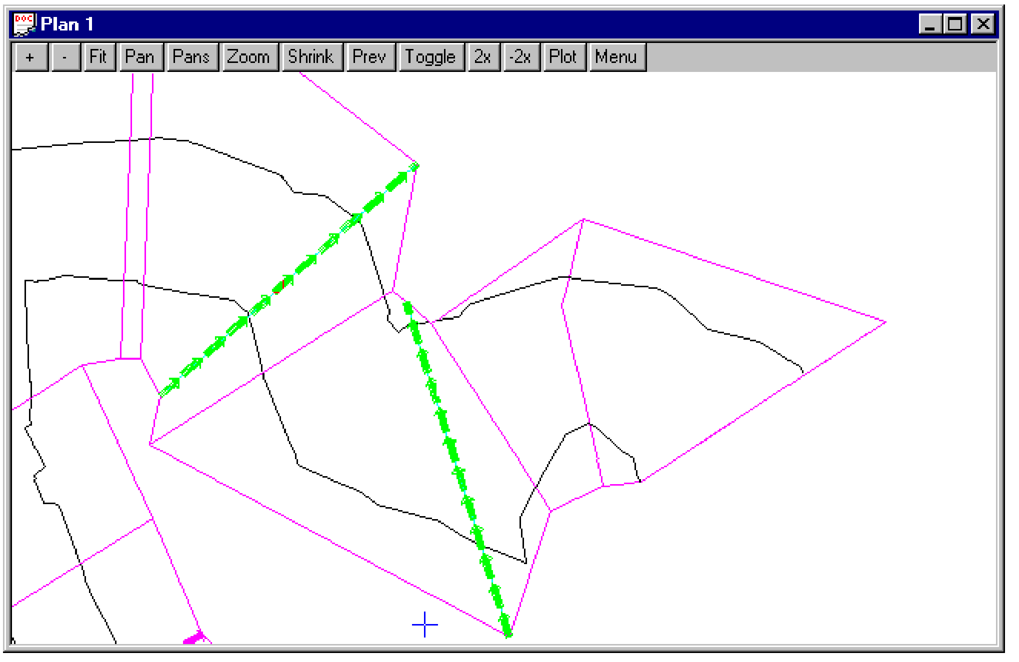
The boundary strings will be broken into sections when the boundary string goes outside the area defined by the cross sections. See below



In these two cases the water level has not exceeded the extents of the cross sections but the boundary strings (black) has gone outside the limits of the water level strings (magenta). **Shape strings** are used to expand the area of the water levels between the sections. The water level assigned to the shape string is a linear interpolation between the upstream and downstream cross section water levels. The interpolation is prorated using the distance along the centre line of the river.

Important! water levels are extended when the shape strings are in a junction area or past the end of a reach.

Shape strings are 2d strings. The direction is not important except for Mike11 models. For Mike11 they must be in the same direction as the cross section strings (usually left bank to right bank). The following drawing shows the boundary string when 2 shape strings have been added (green).


Plan 1
+
-
Fit
Pan
Pans
Zoom
Shrink
Prev
Toggle
2x
-2x
Plot
Menu

21.10 Mapping non 12d and historical water level data

12d can map water levels from river engineering models that were not created by 12d. If the water level results are in the standard HECRAS GIS, HECRAS report, ISIS zzr, UNET, XP SWMM xpx or Mike11 binary file then the standard 12d readers can be used. If the data is historical or not in any of the standard format the **River Mapper** feature can be used.

21.10.1 River Mapper

Position of option on menu: **Design=>Rivers=>River Mapper**

River cross section water level data in an text file may be mapped and displayed using the 12d River Mapper interface. 2d strings are created by the user with the plan shape of the cross sections. 12d assigns water levels from a text file to these strings. These strings are then triangulated to create a water surface tin.

See also

Presenting Water Level Results

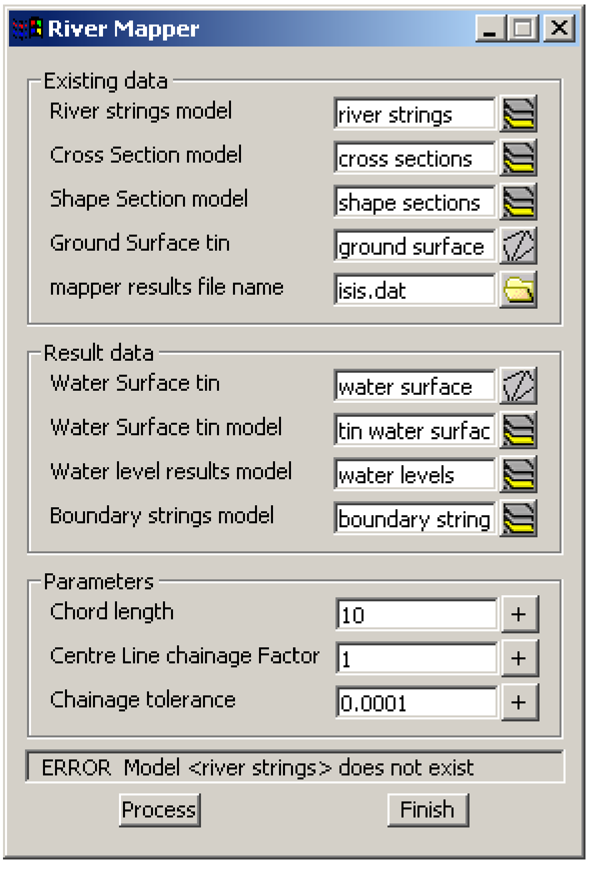
How to for Rivers

Frequently Asked Questions (Rivers)

Usage

The steps are as follows:

1. Create the river strings (centre line, left bank and right bank).
2. Create 2d source strings. Use your name for the cross section as the string name. 12d will match the water levels in the file to this string using the name as the key. If you do not assign a name the interface will assign names to strings according to the chainage along the centre-line and the value of .**Centre line chainage factor.**
3. From the **Design->Rivers** menu select **River Mapper**.
4. Fill in the fields in the dialogue for the rivers strings (step 1 above) and the cross sections (step 2 above).



River strings model Model box

The river strings model specified in the write panel.

Cross Section model Model box

The cross section model specified in the write panel. The interface will search the string names in this model for the cross sections specified in the HEC-RAS report. A match is successful if the HEC-RAS cross section chainage and the string name are within the tolerance specified below in ***Chainage tolerance****.*

Shape string model Model box

For meandering rivers, the cross sections may not be at a close enough spacing to create a water surface that follows the river. 2D shape strings can be created (automatically or manually) to create a water surface to follow the river. Note that water levels are extended when the shape strings are in a junction area or past the end of a reach.

Centre line chainage factor Model box

If you assigned the source/cross section string a name in Step 2 this field is not used. Otherwise, the centre line chainage and the Centre line chainage factor are used to name your source string/cross sections in 12d. The cross sections will be named in the same units as the 12d data if a Centre line chainage factor of 1 is specified. If your existing data uses cross section names in your data file are in kilometres and your 12d data in meters, you will want to use a Centre line chainage factor of 1000. If your existing data cross section names are in miles and your 12d data is in feet then a Centre line chainage factor of 5280 would be used.

Mapper results file name File box

The report is a tab (or space) delimited text file (usually created by a text editor or spreadsheet). The format is

cross section name <tab> water level elevation

each section name with its water level must be on a separate line. For example.

Section A	*2.31*
Section B	*2.32*

If the names contain letters (abc..) then the section name must match the 12d string names exactly (case sensitive). If the names are real numbers than a tolerance for the matching (global and string specific) can be set (see parameters).

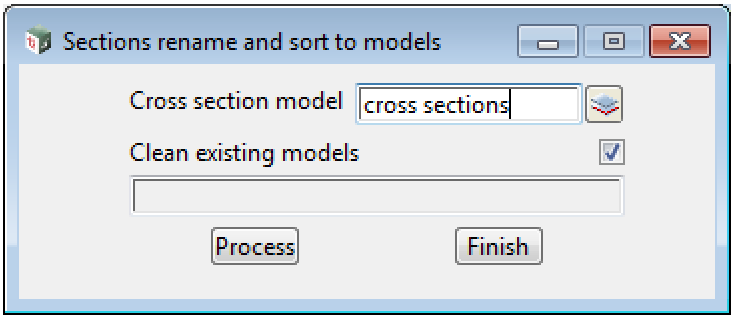
1.0 Rivers Rename and Move Cross Sections

Position of option on menu: **Design =>Rivers=>Xsections rename and move**

River cross section names begin with the river reach number followed by the cross section chainage. To plot these x sections they need to be renamed to the 12d chainage standard.

This routine gets the cross section model name moves the cross section to a model with the name as a prefix followed by the reach number. The reach number is removed from all of the cross section name.

On selecting the Xsections rename and move option, the panel is displayed.



These defaults are used when creating a manhole in a drainage string. The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Cross section model	model box		

this model holds the cross sections created by one of the river interface routines. The name of this model will used as the prefix for the new models to hold the renamed cross sections.

Clean existing models	tick box		

when selected the models that the cross sections are to be moved to will be cleaned before the supplied cross sections are moved into them.

Process	button		

moves and renames the cross sections

Finish	button		

Remove the panel

1.1 12d System Path

When looking for system files 12d first checks your local project directory (the directory that hold the *.project folder), and then the system folders. Unless you have changed these folders in your env.4d file the path is first "program files\12d\12dmodel\5.00\user" and then "program files\12d\12dmodel\5.00\set_ups". In summary:

1. local project directory (the directory that hold the *.project folder
2. program files\12d\12dmodel\5.00\user
3. program files\12d\12dmodel\5.00\set_ups

Never modify the files in set_ups. Copy them to the user directory and make your changes there. Here they are safe from being changed during future 12d updates. The only time you will want files in the local project directory is when you have some project specific data. Master_drainage.xp and Master_rivers.xp are good examples of such cases.

More on System files
Attribute Editor

1.1.1 FAQ Rivers

22 Pipeline

Position of menu: Design =>Pipeline

The **Pipeline** option is used to place large diameter pipelines.

The pipeline is very similar to an alignment string in that it is defined by separate horizontal and vertical geometries.

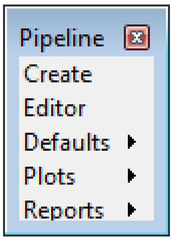
The major differences between a pipeline and an alignment is that a pipeline has a diameter and normally uses circular curves in the vertical geometry rather than the parabolic curves used for alignments.

The pipeline is entered by placing the invert (bottom) of the pipe. Hence in a plan view, the horizontal geometry places the centre of the pipeline. The vertical geometry of the pipeline is for the invert (bottom) of the pipeline.

Note

A **pipeline** string is not the same as a **pipe** string. A **pipe** string is only a 3d string with a diameter.

The **Pipeline** walk-right menu is



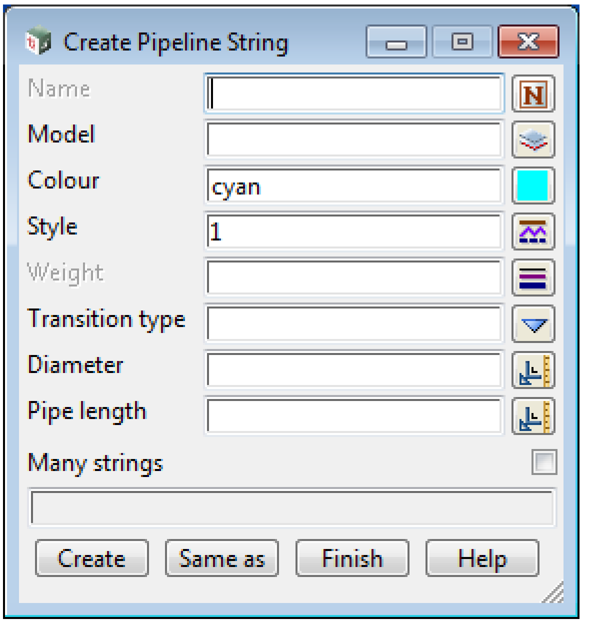
For the option *Create*,go to	22.1 Create
Editor	22.2 Editor
Defaults	22.4 Defaults
Plots	22.5 Plots
Reports	22.6 Reports

22.1 Create

Position of option on menu: Design =>Pipeline =>Create

The **Create** option is used to produce **new** pipelines. If a pipeline string already exists, the **Edit** option is used to modify it.

On selecting the **Create** option, the **Create Pipeline String** panel is displayed.



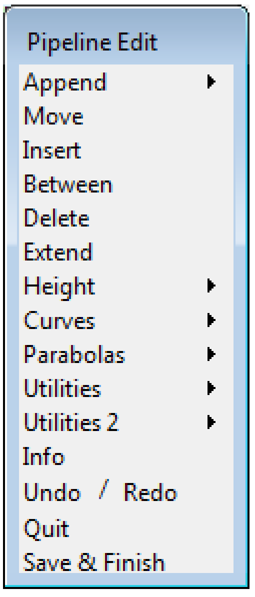
To create a new pipeline string, the pipeline's name, model, colour, style, spiral type and pipe diameter, plus the standard length of pipes used to make up the pipeline are entered into the appropriate fields and the **Create** button selected.

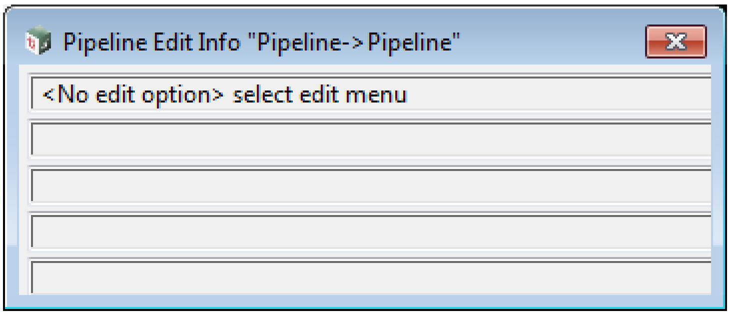
As for a 3d string, to create a new pipeline string with some of the **same** name, colour, model, style etc. **as** an existing string (not necessarily a pipeline), the **same as** button is chosen and the appropriate string selected.

The **Create Pipeline String** panel is then removed and the **Pipeline Edit** menu and **Pipeline Edit Info** panel fired up.

The **Pipeline Edit** menu contains all the options for creating/editing a pipeline string. The **Pipeline Edit Info** panel contains information areas.

The **Pipeline Edit** menu and **Pipeline Edit Info** panel are





As for an alignment string, to create a new pipeline string the user selects the **Append=>HIP's** option from the **Pipeline Edit** menu and start placing points in a plan view.

Since the **Pipeline Edit** menu and **Pipeline Edit Info** panel are the same as those used when editing a pipeline string, the options will be discussed in the following sections under the pipeline **Edit** option.

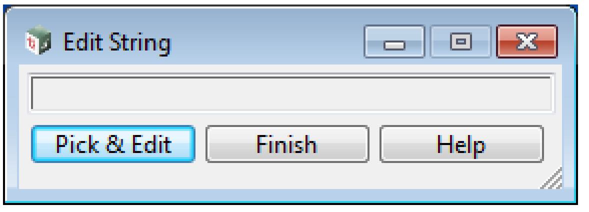
For information on pipeline editing, please continue to the section 22.3 Pipeline Editor.

22.2 Editor

Position of option on menu: Design =>Pipeline =>Editor

This is the same option as **Editor** from the **Strings** walk-right menu on the ***12d Model*** menu.

The string editor is used to modify any ***12d Model*** strings. After selecting the **Editor** option, the **Edit String** panel is placed on the screen to record any error messages.



The option is already in the **Pick** mode (the **Pick & Edit** button only needs to be selected if the pick was cancelled) and the user simply picks and accepts the string to be edited.

From the picked string's type, the editor is able to determine the edits that apply to the string and list them in the string's **Edit** menu.

If a pipeline string is selected, the **Pipeline Edit** menu and **Pipeline Edit Info** panel (as shown in the previous section) are placed on the screen.

The individual edit operations for a pipeline string will now be discussed in detail.

Please continue to the next section 22.3 Pipeline Editor.

22.3 Pipeline Editor

Position of option on menu: Design =>Pipeline =>Editor

A pipeline string is defined by specifying both the horizontal and vertical geometry of the invert (bottom of the pipeline) as separate operations.

The horizontal geometry consists of a series of (x,y) points (called horizontal intersection points) and circular curves applied to the intersection points.

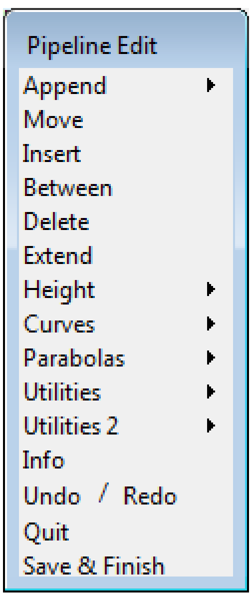
Vertical geometry also consists of a series of points but they are defined with respect to the plan length of the string (chainage) and height. Hence, the vertical geometry is defined by a series of (chainage,height) points (called vertical intersection points) and circular curves applied to the vertical intersection points.

The horizontal geometry is usually defined in a plan view and the vertical geometry in a section view. Hence the pipeline string editor can edit information for the string on both plan and section views.

Note: the string can only be edited on a section view if the string is also profiled on the section view.

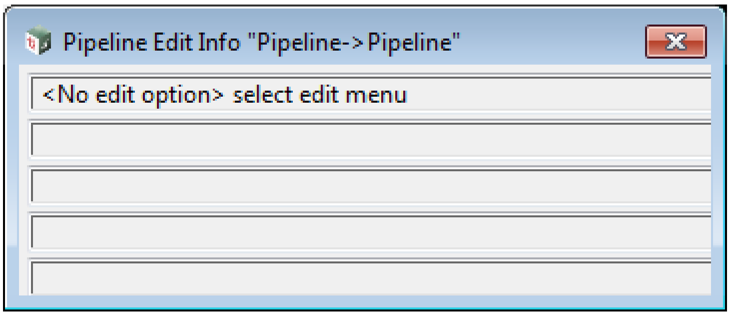
On selecting a pipeline string the **Pipeline Edit** menu and **Pipeline Edit Info** panel are placed on the screen.

The **Pipeline Edit** menu is



append or prepend an IP
move an IP or TP
insert an IP
insert on IP-IP line
delete an IP
extend an IP
modify the strings z-value
circular curves, spirals
VG only, radius or length
interval, start chainage, validate, clear etc.
set diameter and pipe length
toggle edit info panel
undo and redo
quit the edit session
finish the create

and the **Pipeline edit info** panel is



To edit the selected pipeline string, simply select the required edit option from the list of all possible edits in the **pipeline edit** menu.

The edit is **cancelled** by selecting the **cancel** option. The edited string is restored to its pre-edit state and the option terminated.

The edit is **finished** and the new string placed in the given model when the **finish** option is chosen. The edit option then terminates.

When either **quit** or **finish** is selected, a **yes-no-cancel** panel is displayed and the user must confirm the selection.

The **pipeline edit info** panel is used to display information and messages during editing of the pipeline string.

The main message area indicates the purpose of the mouse buttons at each step.

Message area 1 displays the current pipeline edit option and message area 5 indicate the next step in the edit option.

Message areas 2 and 3 are used to display information about the string as the cursor is moved near the string and the string points. For example, for the closest IP, the (x,y,z) position and radius of the IP is displayed in area 2, and in area 3 the bearing-in, bearing-out and deflection angle for the adjacent IPs.

Message area 4 displays the cover above the pipeline and the vertical, horizontal and combined joint deflection.

Most of the options are identical to the edit options for an alignment string and will not be discussed in detail again. The options that are only applicable to a pipeline string will be fully described.

22.3.1 Append

22.3.1.1 HIPs

In a plan view, the **append=>HIPs** option is used to create the first horizontal point in a new pipeline string, to append a new horizontal intersect point to the end of the string or to prepend a new horizontal intersection point to the beginning of the string. The horizontal geometry is for the centre of the pipe in plan.

22.3.1.2 VIPs

In a section view, the **append=>VIPs** option can be used to create and edit the vertical geometry of the picked string if the string is a primary string on any section view. This can be achieved by using either the **VG edit** or the **profile** option from the section view **View ops** menu. The vertical geometry is for the invert (bottom) of the pipe.

The **append** option is the same as **append** for an alignment string and will not be described further.

22.3.2 Move

The **move** option can be used for moving individual horizontal and vertical intersection and tangent points.

The option acts on the horizontal geometry if the point to move is selected in a plan view, or the vertical geometry if the point to move is selected in a section view.

The **move** option is the same as **move** for an alignment string and will not be described again.

22.3.3 Insert

The **insert** option is designed to place a new intersection point in a string between two adjacent horizontal or vertical intersection points (note that the inserted point does not have to be on the line joining the two intersection points).

The **insert** option is the same as **insert** for an alignment string and will not be described again.

22.3.4 Between

The **between** option is similar to the **insert** option except the inserted point **does** have to be on the line joining the two intersection points. To accomplish this, the cursor position is automatically projected onto the IP-IP line to give the new IP point position.

The **between** option is the same as **between** for an alignment string and will not be described again.

22.3.5 Delete

The **delete** option is used to delete horizontal or vertical intersection points from the string.

The **delete** option is the same as **delete** for an alignment string and will not be described again.

22.3.6 Extend

The extend option is used to move an intersection point along the line joining the intersection point to its neighbouring intersection point.

The **extend** option is the same as **extend** for an alignment string and will not be described again.

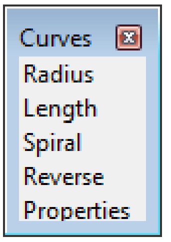
22.3.7 Height

The height option is used to modify the height of a vertical intersection point in the string.

The **height** option is the same as **height** for an alignment string and will not be described again.

22.3.8 Curves

The **curves** option allows the user to add circular curves of a given radius to a either horizontal or vertical intersection points.

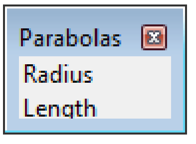


The **curve** option is the same as **curve** for an alignment string and will not be described again.

22.3.9 Parabolas

The **parabolas** option is for adding parabolic curves to the **vertical geometry**.

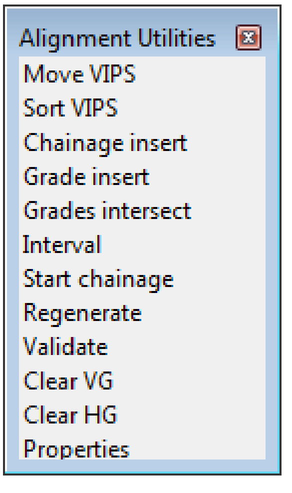
For pipelines, circular curves are normally used on vertical curves.



The **parabolas** option is the same as **parabolas** for an alignment string and will not be described again.

22.3.10 Utilities

The **Alignment Utilities** walk-right menu contains a number of useful miscellaneous options for the pipeline string. The menu is

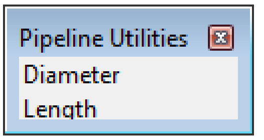


Each of these options is the same as for an alignment string and will not be discussed again.

22.3.11 Utilities 2

The **utilities 2** walk-right menu contains options to modify the diameter of the pipeline and to set the standard length of the pipes used to construct the pipeline.

The **utilities 2** walk-right brings up the **Pipeline Utilities** menu:



22.3.12 Diameter

The **Diameter** option is used to change the diameter of the pipeline string.

After selecting the option, an enter value typed-input box is displayed on the screen containing the pipeline's current diameter.

The new diameter is entered into the typed-input box, terminated with <return>.

The typed-input box then disappears and the option terminates.

22.3.13 Length

The **Length** option is used to change the length of the standard pipe used to construct the pipeline string.

The length of the pipes used in constructing the pipeline string is used for calculating joint deflections along the pipeline.

After selecting the option, an enter value typed-input box is displayed the pipeline's current standard pipe length.

The new length is entered into the typed-input box, terminated with <return>.

The typed-input box then disappears and the option terminates.

22.3.14 Info

The **pipeline edit info** panel is toggled **on/off** by the **info** option in the **pipeline edit** menu.

22.3.15 Quit and Finish

Even after points are created for the new string, the create process can be **aborted** by selecting the **quit** option from the **pipeline edit** menu. The option then terminates and **no** string is created.

The create process is completed and the new string created and placed in the given model when the **finish** option is chosen from the **pipeline edit** menu.

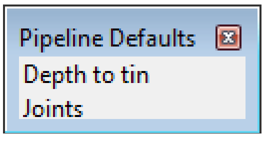
When either the **finish** or **quit** option is selected, a **yes-no-cancel** panel is displayed and the user must confirm the selection.

22.4 Defaults

Position of menu: Design =>Pipeline =>Defaults

The **defaults** menu sets default depth to tin and joint deflection angle.

The **default** walk-right menu is

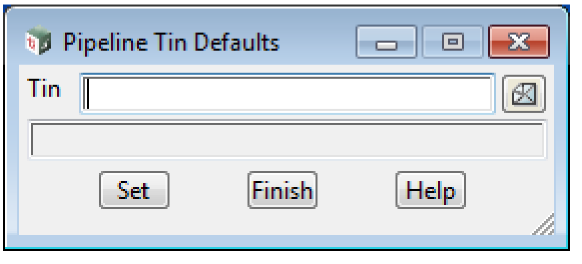


For the option Depth to tin, go to 22.4.1 Depth to Tin
Joints 22.4.2 Joints

22.4.1 Depth to Tin

Position of option on menu: Design =>Pipeline =>Defaults =>Depth to tin

On selecting the **tin** option, the **pipeline tin defaults** panel is displayed.

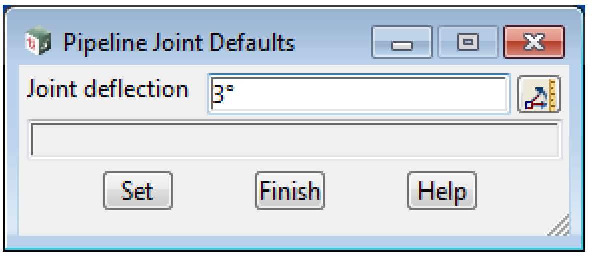


This panel is for setting the default terrain tin which is used to calculated the cover above the pipeline.

22.4.2 Joints

Position of option on menu: Design =>Pipeline =>Defaults =>Joints

On selecting the **joints** option, the **pipeline joint defaults** panel is displayed.



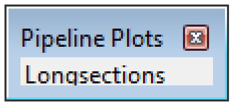
This panel is for setting the maximum joint deflection for the individual pipes that make up the pipeline.

22.5 Plots

Position of menu: Design =>Pipeline =>Plots

Currently the **plots** menu contains only one option and it is used for producing longsection plots of major pipelines.

The **plots** walk-right brings up the **pipeline plots** menu:



This option will now be described in more detail.

Please continue to the next section 22.5.1 Longsections.

22.5.1 Longsections

Position of option on menu: Design =>Pipeline =>Plots

Position of option on menu: Plot =>Pipeline Plot PPF Editor

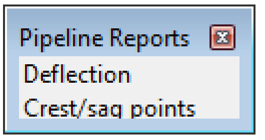
For more information please go to the section 26.7 Pipeline Plot PPF Editor

Continue to the next section 22.6 Reports.

22.6 Reports

Position of menu: Design =>Pipeline =>Reports

The **Reports** walk-right menu contains reports for joint deflections along the pipeline and crest and sag points.



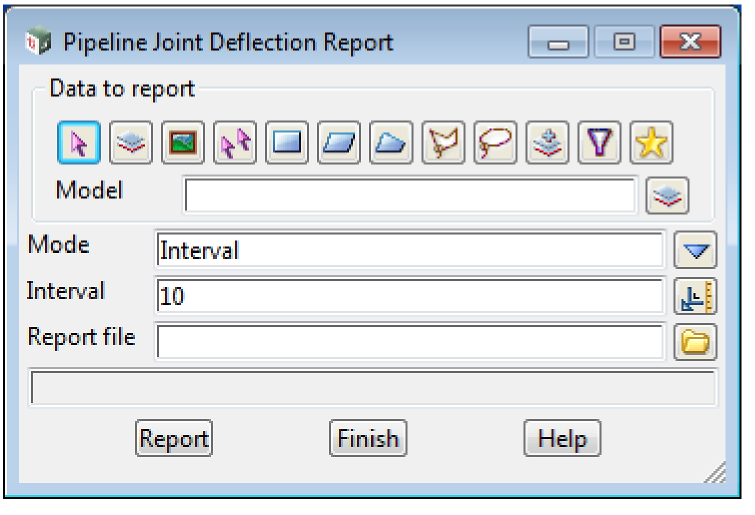
Fo the option Deflection, go to 22.6.1 Deflection
Crest/ sag points 22.6.2 Crest/Sag Points

22.6.1 Deflection

Position of option on menu: Design =>Pipeline =>Reports=>Deflection

The deflection report gives the horizontal, vertical and combined joint deflections along the pipeline.

After selecting the **Deflection** option, the **Pipeline Joint Deflection Report** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Report file	input		

name of the file for the report

Field Description	Type	Defaults	Pop-Up
Mode	input	interval	interval, critical, summary

if interval - reports the joint deflection at the interval given plus any horizontal or vertical intersection points without curves on them

If critical - only reports points where the deflection is greater than the value given in the **pipeline joint defaults** *panel.*

If summary - gives the maximum joint deflections

Field Description	Type	Defaults	Pop-Up
Interval	input	10	

the chainage distance between points along the pipeline where joint deflections are to be reported

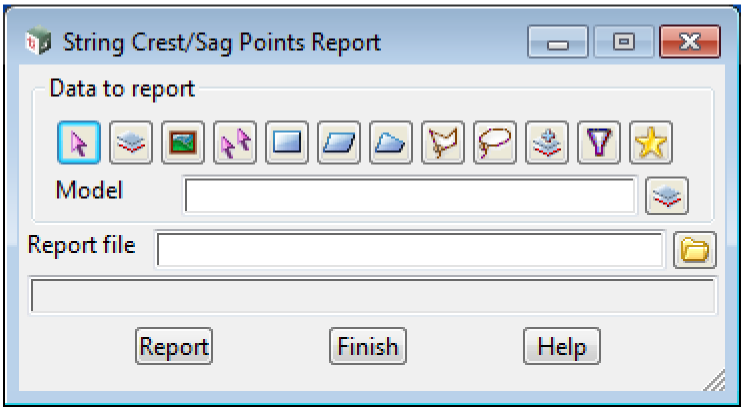
Pick & Report button

pick the pipeline string to report on.

22.6.2 Crest/Sag Points

Position of option on menu: Design =>Pipeline =>Reports=>Crest/Sag Points

After selecting the **Crest/sag points** option, the **String Crest/Sag Points Report** panel is displayed.



This is the same option as the **Reports=>Strings=>Crest/sag points**.

For more information please go to 22.6.2 Crest/Sag Points.

For each selected string, the string's crest/sag points are written to the report file given in the report file field of the panel.

23 Volumes

Position of menu: **Design =>Volumes**

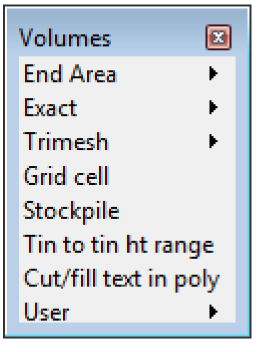
For tins, ***12d Model*** provides two distinct methodologies for calculating volumes: End Area and Exact. See 23.1 Definition of Cut and Fill and 23.2 End Area and Exact Volumes

For both the end area and exact cases, the volumes are broken up into cut and fill volumes.

By using an apporximation method, volumes can be calculated between two tins for a given height range.

Exact volumes can also be calculated for closed trimeshes.

The **Volumes** walk-right menu is



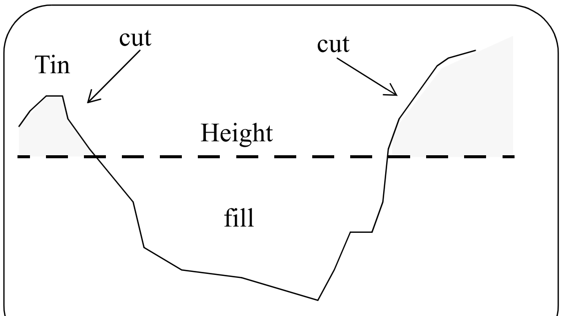
For the options see:

End area	23.3 End Area
Exact	23.4 Exact
Trimesh	23.5 Trimesh
Grid cell	23.6 Grid Cell
Stockpile	23.7 Stockpile
Tin to tin ht range	23.8 Tin to Tin by Height Range
Cut/fill text in poly	23.9 Create Cut/Fill Text Within Polygon

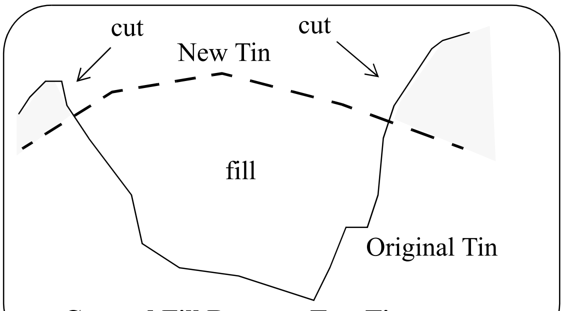
23.1 Definition of Cut and Fill

Cut is defined to be where ever the new surface or given height is **below** the original surface. That is, you have to **cut** to get from the original surface to the new surface. The sign of cut quantities is set via the **Cut volume sign** field from the **Default settings** tab of the Defaults panel (**Utilities=>Defaults**) and the default for *cut volume sign* is positive.

Fill is defined to be where ever the new surface or given height is **above** the original surface. That is, you have to **fill** to get from the original surface to the new surface. The sign of fill quantities is the opposite to the value for cut and cut is set via the **Cut volumes sign** field from the **Default settings** tab of the Defaults panel (**Utilities=>Defaults**). The default for fill is *negative*.



Cut and Fill Between a Tin and a Height



Cut and Fill Between Two Tins

Go to the next section 23.2 End Area and Exact Volumes or return to 23 Volumes.

23.2 End Area and Exact Volumes

In the **End Area** methods using tins, the sections are *automatically* created through the tins. From the sections, cut and fill areas for the sections and the cut and fill volumes between the sections are calculated. End Area is an approximation method. For more information on the end area volume calculations, go to the section 23.2.1 Theory of End Area Volumes

For **Exact** volumes, the volume between a tin and a plane or between two tin is determined by working out the exact geometrical shapes between the tins and summing their volumes. Unlike the end area methods which are approximations, the Exact method calculates the exact volume. For more information on the exact volume calculations, go to the section 23.2.2 Theory of Exact Volumes

The only method that is not an approximation is **Exact** Volumes.

But even these volumes should always be double checked by using an End Area with parallel sections. And a couple of different angles of the sections should be used to make doubly sure that the End Area method is good.

The next best after **Exact** are the **End Area** method with **parallel sections**.

The worst approximations are **End Area** methods where **the sections are NOT parallel**. For example when calculating volumes using the sections along an alignment with curves.

With the End Area approximation methods, how good the approximation also depends on what the data is like. If the calculation is along an alignment then how good the approximation is will also vary depending on how tight the curves on the alignment are, the distribution of cut and fill with respect to the alignment, whether the alignment is in the middle of the sections etc.

However with all the End Area section methods, you should always look at the generated sections through the tins to make sure they have all worked correctly.

Exact volumes are more accurate than End Area volumes. However, for most End Area calculations with parallel sections and a small distance between sections, the results are the same to any reasonable degree of accuracy.

The reason for having both methods available in 12d Model is

(a) to provide an independent check on the volumes produced

(b) each method produces a different breakup of the volumes.

In the End Area method, cut and fill volumes are produced on a section by section basis. This is needed for applications such as mass-haul calculations.

For the Exact method, cut and fill volumes are easily calculated between different **depth** values. This is useful for applications such as depth (isopach) analysis.

For both the end area and the exact method, 12d Model provides options to calculate the volume

(a) between a surface (tin) and a fixed height (z-value)

and

(b) the volume between two surfaces (tins) - an original surface (tin) and a new surface (tin).

NOTE: For the ***end area volumes***, sections are automatically generated and used for the calculations. You do ***not*** need to generate the sections separately.

For checking purposes, each of the End Area volumes options can keep the generated sections so that they can be checked.

For more information on the End Area volume calculations, go to the section 23.2.1 Theory of End Area Volumes.

For information on the End Area options, please go to the section 23.3 End Area.

For more information on the Exact volume calculations, go to the section 23.2.2 Theory of Exact Volumes.

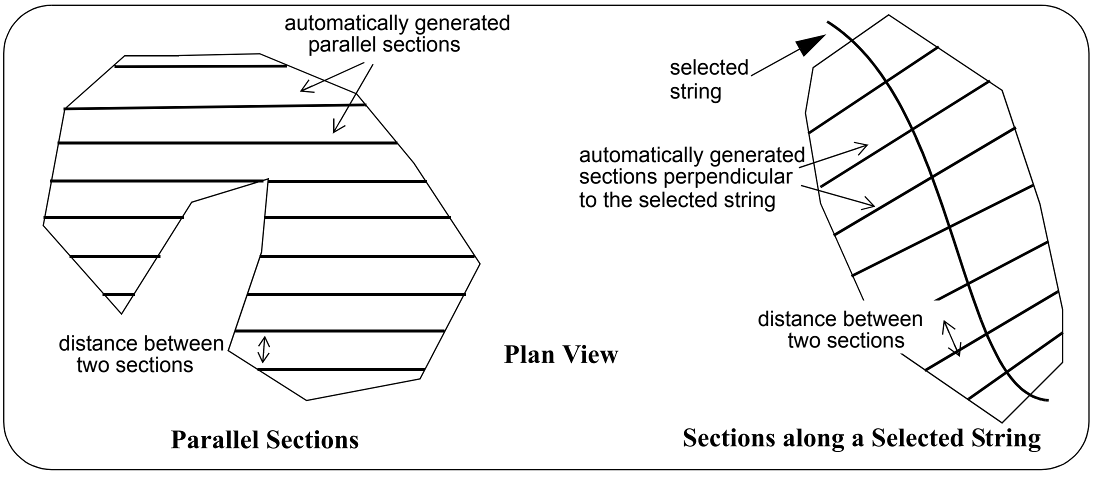
For information on the Exact volume options, please go to the section 23.4 Exact.

23.2.1 Theory of End Area Volumes

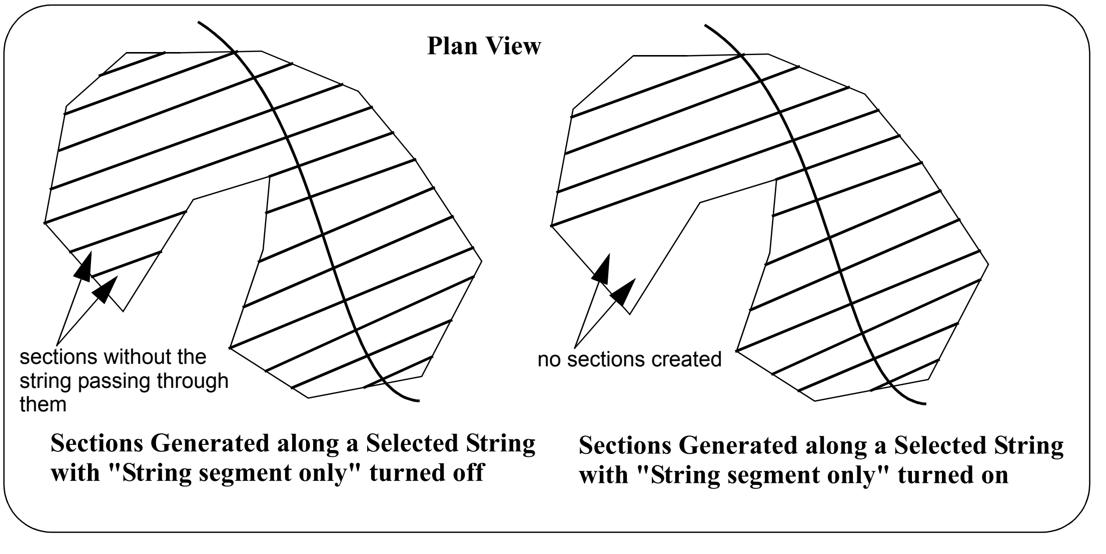
In the **end area** method of calculating volumes, sections are automatically generated through the tins. For each section, cut and fill areas are calculated and then cut and fill volumes are calculated from the cut and fill areas. For the definition of cut and fill, please go to the section 23.1 Definition of Cut and Fill.

For the end area method, 12d Model provides options to calculate the volume between a surface (tin) and a fixed height (z-value) and the volume between two surfaces (tins).

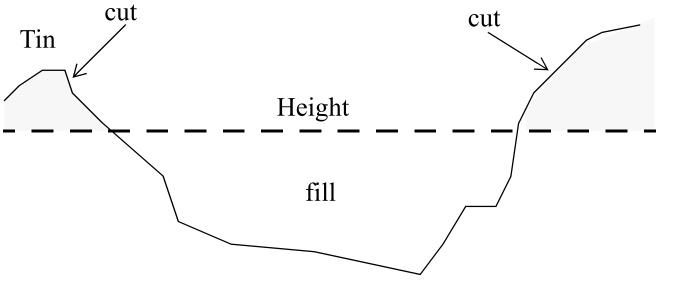
In the **end area** methods of calculating volumes, sections can be generated through a tin or tins at either a *given angle* or at right angles to a ***selected*** string.



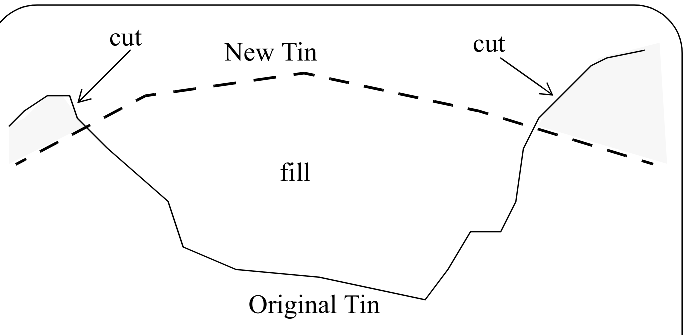
When using the ***sections along a string*** options, it is possible for a generated section to be broken into more than one part and a parameter exists to only use those sections that contain the selected string.



For each generated section, the *cut* and *fill* areas are calculated and then the volumes between the two sections using either the ***average end*** (see 23.2.1.1 Average End Area Formula) or the ***two section prismoidal*** methods (see 23.2.1.2 Two Section Prismoidal Formula).



Section view - Cut and Fill Areas Between a Tin and a Height



Section view - Cut and Fill Areas Between Two Tins

NOTE: For the ***end area volumes***, sections are automatically generated and used for the calculations. You do ***not*** need to generate the sections separately.

See

23.2.1.1 Average End Area Formula

23.2.1.2 Two Section Prismoidal Formula

23.2.1.1 Average End Area Formula

Using the *Average End Area* method the *volume* between two *consecutive* sections is calculated by the formula

Volume of cut between the two sections = D x (C1 + C2) / 2

where

D = distance between the two sections
C1 = area of cut for the first section
C2 = area of cut for the second section

The **Total cut volume** is the sum of the cut volumes between each pair of sections.

Similarly

Volume of fill between the two sections = D x (F1 + F2) / 2

where

D = distance between the two sections
F1 = area of fill for the first section
F2 = area of fill for the second section

The **Total fill volume** is the sum of the fill volumes between each pair of sections.

Go to the next section 23.2.1.2 Two Section Prismoidal Formula or return to 23.2.1 Theory of End Area Volumes.

23.2.1.2 Two Section Prismoidal Formula

The *two section prismoidal* method, as its name suggests, is a modification of the standard prismoidal method but it uses only two sections rather than the three sections required by the standard prismoidal method.

Using the *Two Section Prismoidal* method the *volume* between two *consecutive* sections is calculated by the formula

Volume of cut between the two sections = D x (C1 + C2 + Sqrt(C1 x C2)) / 3

where

D = distance between the two sections
C1 = area of cut for the first section
C2 = area of cut for the second section

The **Total cut volume** is the sum of the cut volumes between each pair of sections.

Similarly

Volume of fill between the two sections = D x (F1 + F2 + Sqrt(F1 + F2)) / 3

where

D = distance between the two sections
F1 = area of fill for the first section
F2 = area of fill for the second section

The **Total fill volume** is the sum of the fill volumes between each pair of sections.

For the definition of cut and fill, please go to the section 23.1 Definition of Cut and Fill.

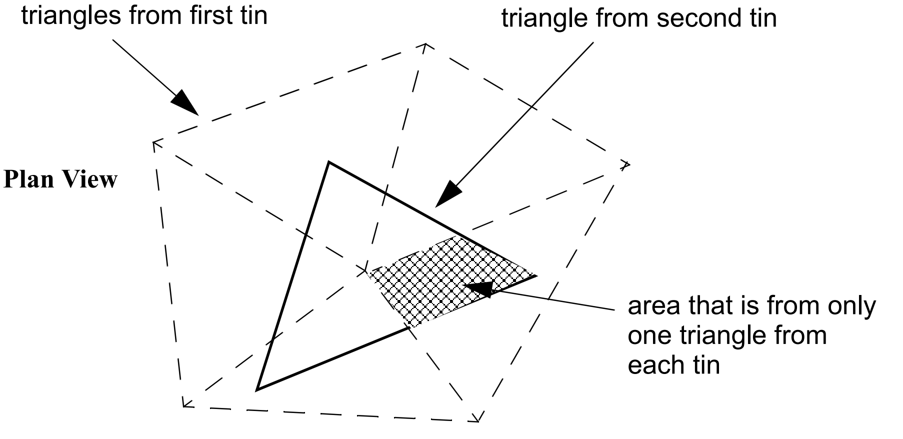
For information on the end area options, please go to the section 23.3 End Area.

For information on the theory for exact volumes, please continue to the section 23.2.2 Theory of Exact Volumes.

23.2.2 Theory of Exact Volumes

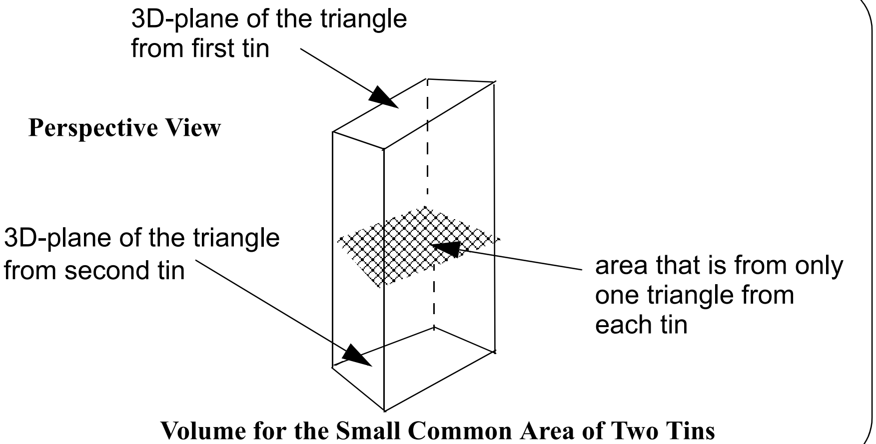
This method calculates the exact volume between two triangulated surfaces (tins), or between one triangulated surface and a plane of constant height.

The volumes are calculated by mathematically subdividing the triangles from both tins into areas that contain only one triangle from each tin.



Subdividing the Triangles for the Exact Method of Calculating Volumes

In three dimensions, these areas represent well defined polygons with the plane of the triangle from one tin as its top and the plane of the triangle from the other tin as its bottom.



Volume for the Small Common Area of Two Tins

For each small area, the cut and fill volume between the two tins can be calculated exactly.

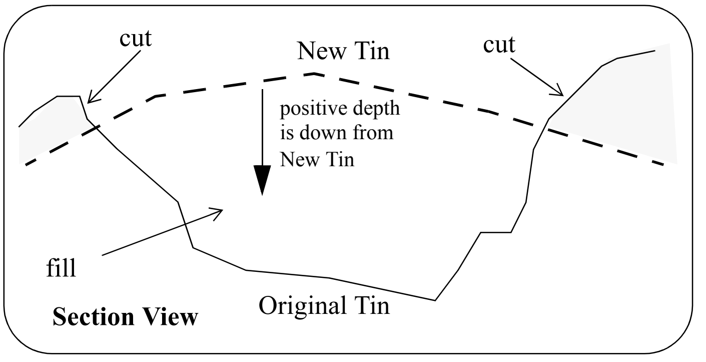
The total *cut* and *fill* volume between the two tin is simply the sum of the cut and fill volumes for each small area.

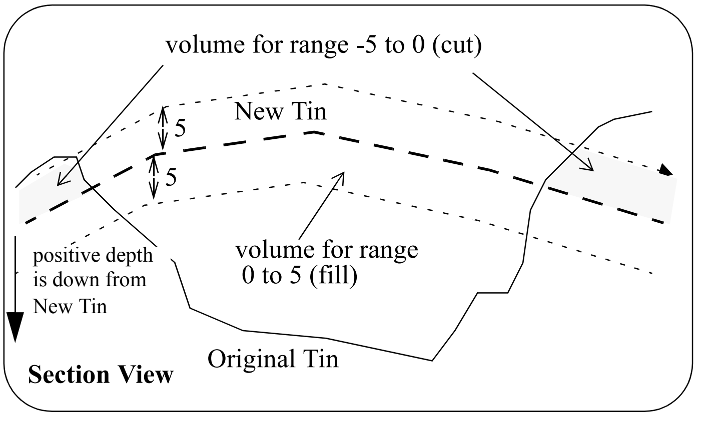
For the definition of cut and fill, please go to the section 23.1 Definition of Cut and Fill.

For information on the exact volume options, please go to the section 23.4 Exact.

The calculated volumes by the exact method can be produced and reported on over *user supplied depth ranges*. Similarly, a view can be coloured on a depth basis using the same **range** file.

IMPORTANT NOTE - the ***volume totals*** are **only** calculated for the ***depths in the range file***. If the depth ranges ***does not cover*** the entire depth difference between the tins then the totals which are the sum of the different depth ranges will ***not*** be the same as the volumes between the two tins.





For information on the theory for end area volumes, please go to the section 23.2.1 Theory of End Area Volumes.

23.3 End Area

Position of menu: **Design =>Volumes =>End area**

These volumes are calculated using the end area method. For more information on the end area volume calculations, go to the section 23.2.1 Theory of End Area Volumes.

The options **tin to height** and **tin to tin** use parallel sections defined at a user supplied angle and separation distance, to calculate the end areas.

The options **string tin to height** and **string tin to tin** use sections defined at right angles to a user selected string, for their end area volume calculations. The distance between sections is supplied by the user.

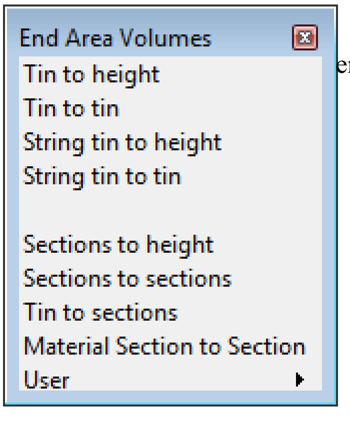
The options **sections to height** and **sections to sections** use sections already created to calculate volumes using end areas.

In all the volume options, the sections are restricted to a user defined polygon.

12d Model reports on the cut, fill and balance for each of the sections used in the calculations and also the total cut, fill and balance.

In the report, the sign for cut (negative or positive) is given by the cut volume sign from the **Defaults** panel (fill will have the opposite sign).

The **End area** walk-right menu is



end area volumes between surface and height
between surfaces
along string between surface and height
along string between surfaces

between a model of sections and height
between two models of sections
between a model of sections and a tin
for one or more material layer(s)

Each option in this menu will now be described.

For the options see

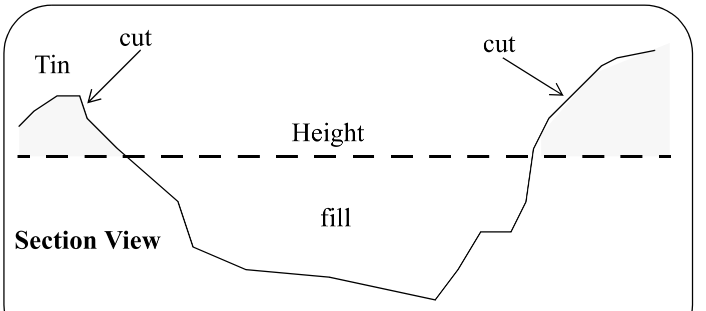
23.3.1 End Area - Tin to Height
23.3.2 End Area - Tin to Tin
23.3.3 End Area - String Tin to Height
23.3.4 End Area - String Tin to Tin
23.3.5 End Area - Sections to Height
23.3.6 End Area - Sections to Sections
23.3.7 End Area - Tin to Sections
23.3.8 End Area - Material Section to Section

For more information on the end area volume calculations, go to the section 23.2.1 Theory of End Area Volumes.

23.3.1 End Area - Tin to Height

Position of option on menu: **Design =>Volumes =>End Area =>Tin to height**

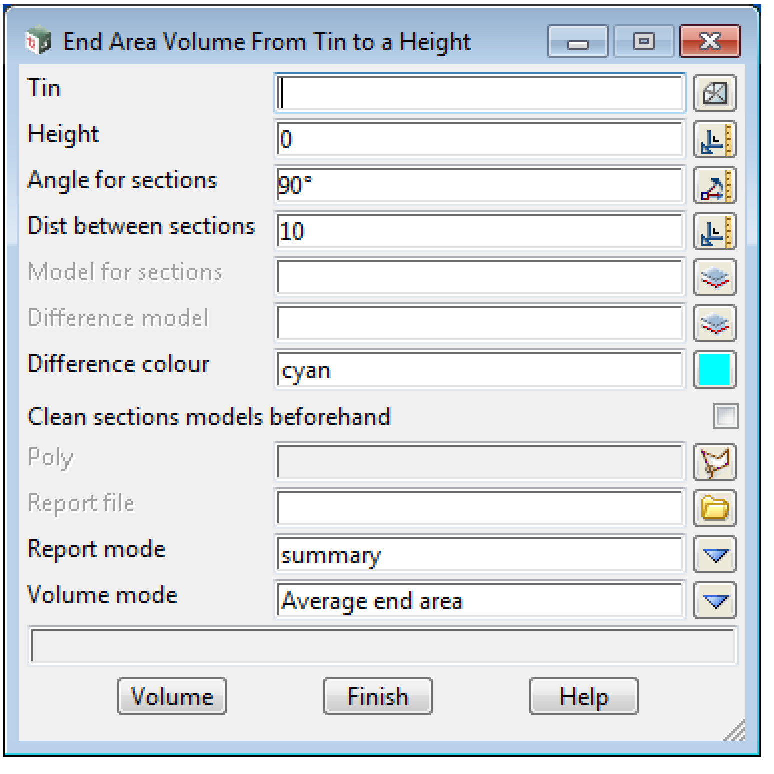
The **Tin to height** option is used to find volumes between an existing surface and a user supplied height. The volumes can only be calculated within a user supplied polygon.



Cut and Fill Areas Between and a Tin and a Height

For more information on the end area volume calculations, go to the section 23.2.1 Theory of End Area Volumes.

On selecting the **Tin to height** option, the **End Area Volume From Tin to a Height** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	input		available tins

name of the tin for which the volume between it and a height (z value) will be calculated.

Field Description	Type	Defaults	Pop-Up
Height	input		

the volume is calculated between the tin and the value in this field.

Angle for sections input 90.0 0,45,90

angle (in degrees) of the lines to section along.

Dist between sections input 10.0 1,10,100

distance between the lines to section along

Model for sections input available models

if non-blank, the sections though the tin used for the end area calculations are retained and placed in the model given in this field. The sections are given the same colour as the tin.
If blank, the sections are not kept.

Difference model input available models

if non-blank, the sections which are the difference between the tin sections and the height are retained and placed in the model given in this field.
If blank, the sections are not kept.

Difference colour input available colours

colour for the difference sections strings

Clean sections models beforehand tick box

if tick, the model of sections and difference sections are cleaned out before the option runs.

Poly poly string-select

if selected, the string, rectangle or lasso used as the bounding polygon for the volume calculations.

Report file input *.rpt

name of the file to contain the volume report. If the file already exists, the report will be appended to the file. If no name is given, no report is produced.

Report mode input summary summary, full

*if **full**, the cut and fill details for every section are included.*
*If **summary**, just the cut and fill totals are given.*

Volume mode choice box Average end area average end area,

prismoidal - 2 sections

the two methods in 12d for calculating volumes using areas of sections. For more information on each method, go to 23.2.1.1 Average End Area Formula

Volume button

The volume between the tin and the height (z-value) within the selected bounding polygon is calculated by the end area method. The sections through the polygon will be calculated along straight lines at the angle given by the angle *field and at a separation given by the* dist *field. The sections are made against the tin given in* tin *field.*

How to Use the Panel and Panel Messages

(a) Select the bounding string by choosing **Poly** and picking the required string.

(b) Volume processing begins on selecting the **Volume** button.

Progress messages - sent to the panel message area

calculating volumes

Completion message - sent to the panel message area

c *cut volume* f *fill volume* bal *total (balance) volume*

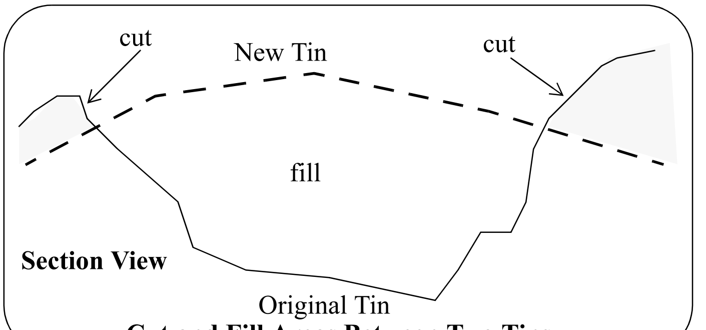
<Esc> can be used to terminate the option during volume calculations.

Continue to the next section 23.3.1 End Area - Tin to Height or return to 23.3 End Area.

23.3.2 End Area - Tin to Tin

Position of option on menu: **Design =>Volumes =>End Area =>Tin to tin**

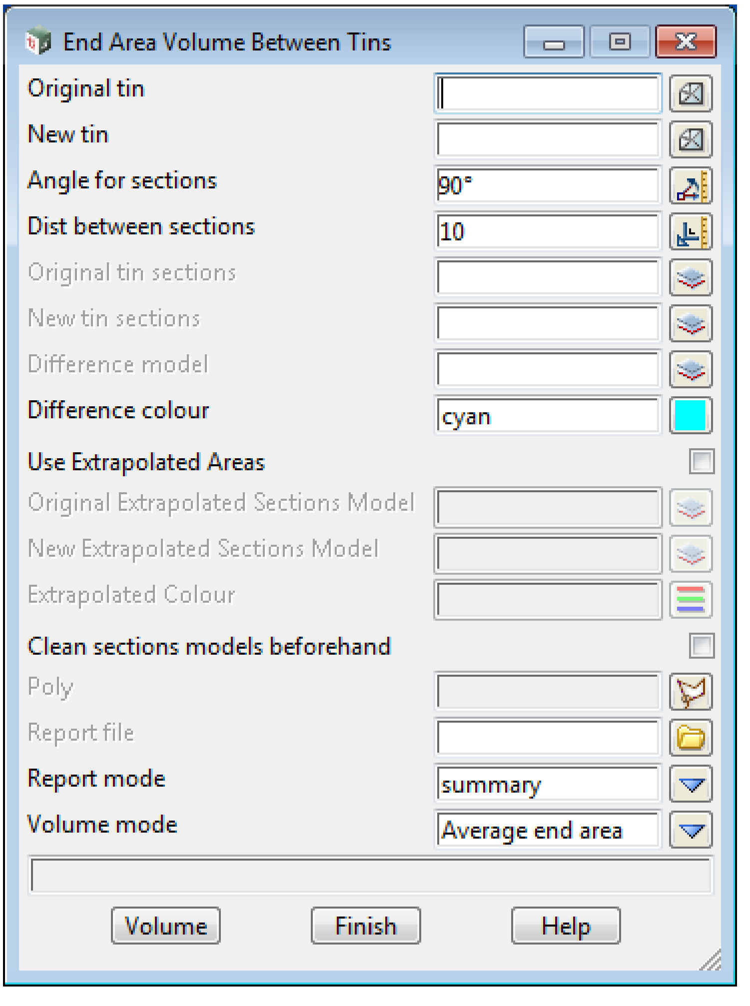
The **Tin to tin** option is used to find volumes between an existing and a new surface. The volumes can only be calculated within a user supplied polygon.



Cut and Fill Areas Between Two Tins

For more information on the end area volume calculations, go to the section 23.2.1 Theory of End Area Volumes.

On selecting the **Tin to tin** option, the **End Area Volume Between Tins** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Original tin	input		available tins

name of the original tin for determining volumes.

New tin	input		available tins

name of the new tin for determining volumes.

Angle for sections	input	90.0	0,45,90

angle (in degrees) of the lines to section along.

Dist between sections	input	10.0	1,10,100

distance between the lines to section along

Original tin sections	input		available models

*if **not blank**, the sections though the original tin used for the end area calculations are retained and placed in the model given in this field. The sections are given the same colour at the original tin.
If blank, the sections are not kept.*

New tin sections	input		available models

*if **not blank**, the sections though the new tin used for the end area calculations are retained and placed in the model given in this field. The sections are given the same colour as the new tin.
If blank, the sections are not kept.*

Difference model	input		available models

*if **not blank**, sections which are the depth from the original tin to the new tin are created and placed in the model given in this field. That is, the value on the difference section is:*

z- diff = z-value of original section - z-value of new tin.

Hence triangulating the difference sections and contouring them will give you the depths isopachs between the two tins.

If blank, the sections are not kept.

Difference colour	input		available colours

colour for the difference sections strings

Use extrapolated areas	tick box	not tick	

*if **not tick**, areas (and hence volumes) are only calculated where **both** sections exist. Hence the sections are limited to where both occur.*

*If **tick**, when the sections are not the same length, the end points of the above and below sections are connected thereby extrapolating the smaller section. This method is not recommended since data does not exist.*

Original extrapolated sections model	model box	available models

*if non-blank, the extrapolated sections created from the Original sections are placed in this model.
If blank, the extrapolated sections for the original sections are not kept.*

New extrapolated sections model	model box	available models

*if non-blank, the extrapolated sections created from the New sections are placed in this model.
If blank, the extrapolated sections for the new sections are not kept.*

Extrapolated colour	colour box	available colours

colour for the extrapolated sections

Clean sections models beforehand	tick box

if tick, the model of sections and difference sections are cleaned out before the option runs.

Poly	poly string-select		

if selected, the string, rectangle or lasso used as the bounding polygon for the volume calculations.

Report file	input		*.rpt

name of the file to contain the volume report. If the file already exists, the report will be appended to the file. If no name is given, no report is produced.

Report mode	input	summary	summary, full

*if **full**, the cut and fill details for every section are included.*
*If **summary**, just the cut and fill totals are given.*

Volume mode	choice box	Average end area	average end area, prismoidal - 2 sections

the two methods in 12d for calculating volumes using areas of sections. For more information on each method, go to 23.2.1.1 Average End Area Formula

Volume	button		

The volume between the new and the original tins within the selected polygon string is calculated by the end area method. The sections through the polygon selected by **poly** *will be calculated along straight lines at the angle given by the* angle *field and at a separation given by the* dist *field. The sections are made against the tins given by the original and the new tin fields.*

How to Use the Panel and Panel Messages

(a) Select the bounding polygon string by choosing **Poly** and picking the required string.

(b) Volume processing begins on selecting the **volume** button.

Progress messages - sent to the panel message area

calculating volumes

Completion message - sent to the panel message area

c *cut volume* f *fill volume* bal *total (balance) volume*

<Esc> can be used to terminate the option during volume calculations.

Continue to the next section 23.3.3 End Area - String Tin to Height or return to 23.3 End Area.

23.3.3 End Area - String Tin to Height

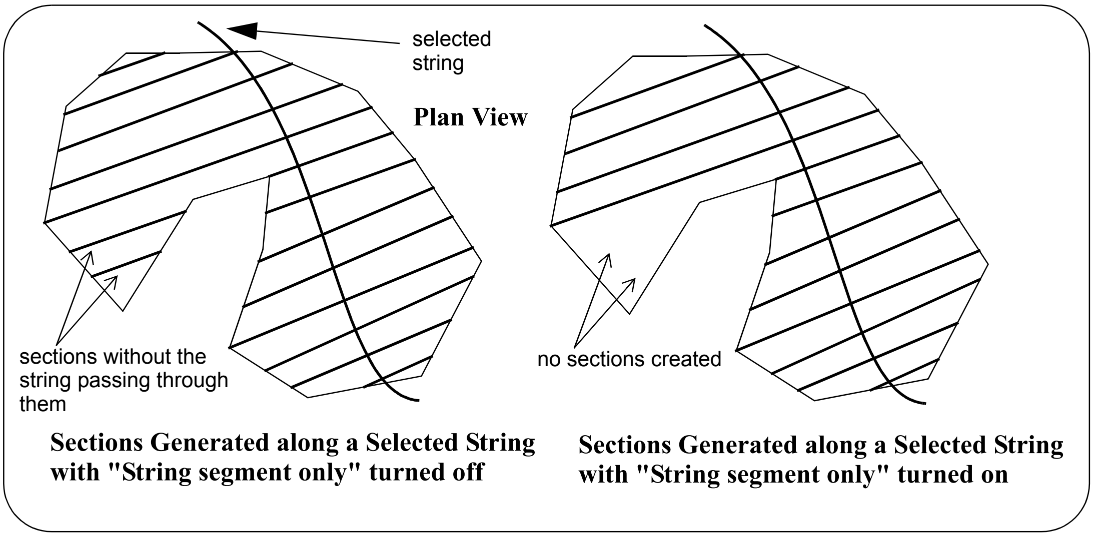
Position of option on menu: **Design =>Volumes =>End Area =>String tin to height**

The volume options already described calculate the volumes by forming parallel sections through the tin and then using the end area method between adjacent sections.

In many situations, the sections to be used are defined to be at right angles to a user selected string. This method is commonly used in road calculations.

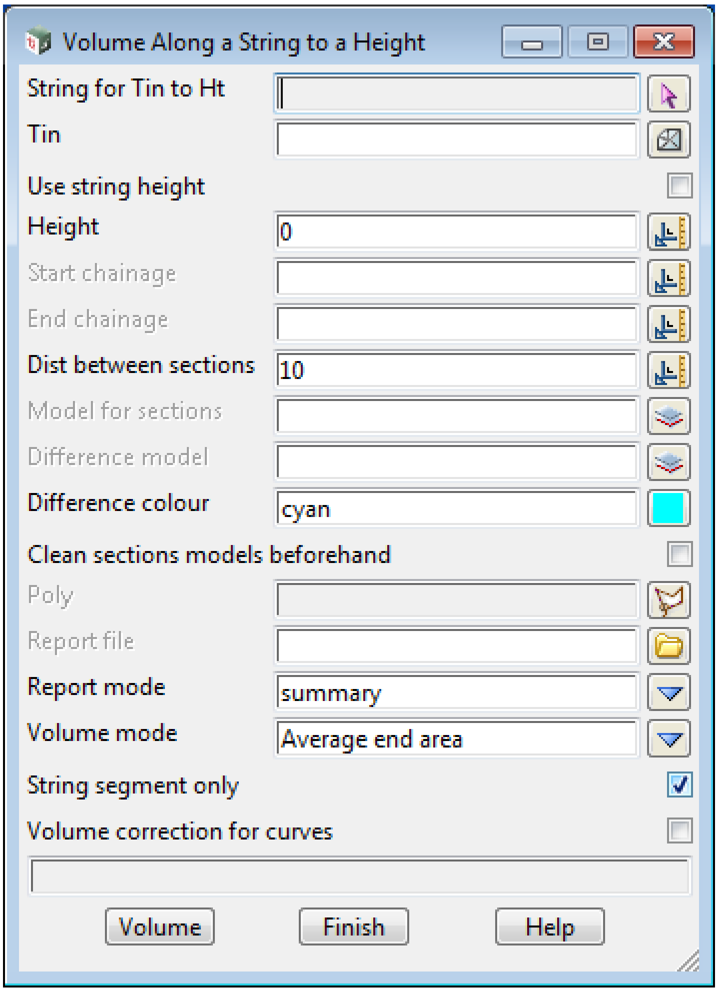
The options **String tin to height** and **String tin to tin** both use sections defined at right angles to a user selected string for the end area volume calculations. For both options, the volumes can only be calculated within a user supplied polygon.

It optionally uses the whole section within the polygon or only the part of the section that contains the string.



For more information on the end area volume calculations, go to the section 23.2.1 Theory of End Area Volumes.

On selecting the **String tin to height** option, the **Volume Along a String to a Height** panel is displayed.



The fields and buttons in this panel are used as follows.

Field Description	Type	Defaults	Pop-Up
String for Tin to Ht	string select		

a string is selected to be used to define the sections used in the end area calculations. The sections are taken at right angles to this string.

Tin	input		available tins

name of the tin for which the volume between it and a height (z-value) will be calculated.

Use string height	tick box		

*if **tick**, the height used for the area calculations is taken at each section from the user selected string.*
*If **not tick**, the height given in the* height *panel field is used in the area calculations.*

Height	input	0	

if Use string height *is set to **no tick**, the volume is calculated between the tin and the value in this field.*

Start chainage	input		

if non blank, the chainage of the first section to use for volume calculations.
If blank, start with the section with the lowest chainage.

End chainage	input		

if non blank, the chainage of the last section to use for volume calculations.
If blank, end with the section with the highest chainage.

Dist between sections input 10 1,10,100

distance between the sections taken at right angles down the user selected string.

Model for sections input available models

if non-blank, the sections though the tin used for the end area calculations are retained and placed in the model given in this field. The sections are given the same colour as the tin.
If blank, the sections are not kept.

Difference model input available models

if non-blank, the sections which are the difference between the tin sections and the height are retained and placed in the model given in this field.
If blank, the sections are not kept.

Difference colour input available colours

colour for the difference sections strings

Clean sections models beforehand tick box

if tick, the model of sections and difference sections are cleaned out before the option runs.

Report file input *.rpt

name of the file to contain the volume report. If the file already exists, the report will be appended to the file. If no name is given, no report is produced.

Report mode input summary summary, full

*if **full**, the cut and fill details for every section are included.*
*If **summary**, just the cut and fill totals are given.*

Volume mode choice box Average end area average end area,

prismoidal - 2 sections

the two methods in 12d for calculating volumes using areas of sections. For more information on each method, go to 23.2.1.1 Average End Area Formula

String segment only tick box tick

*if **tick**, only the part of the section within the polygon that contains the selected string is used.*
*If **not tick**, the whole section within the polygon is used (see* 23.2.1 Theory of End Area Volumes*).*

Volume correction for curves tick box

if tick, volume corrections are made when going around curves.

Poly poly string-select

if selected, the string, rectangle or lasso used as the bounding polygon for the volume calculations.

Volume button

The volume between the tin and the height (z-value) within the selected polygon is calculated by the end area method. The sections through the tin are calculated at right angles to the selected string at a separation given by the dist *field. The section strings are placed in the model given in the* model *field.*

How to Use the Panel and Panel Messages

(a) Select the bounding polygon string by choosing **Poly** and picking the required string.

(b) Select a string to define the sections string by choosing **String** button

(c) Volume processing begins on selecting the **volume** button.

Progress messages - sent to the panel message area

calculating volumes

Completion message - sent to the panel message area c *cut volume* f *fill volume* bal *total (balance) volume*

<Esc> can be used to terminate the option during volume calculations.

Continue to the next section 23.3.4 End Area - String Tin to Tin or return to 23.3 End Area.

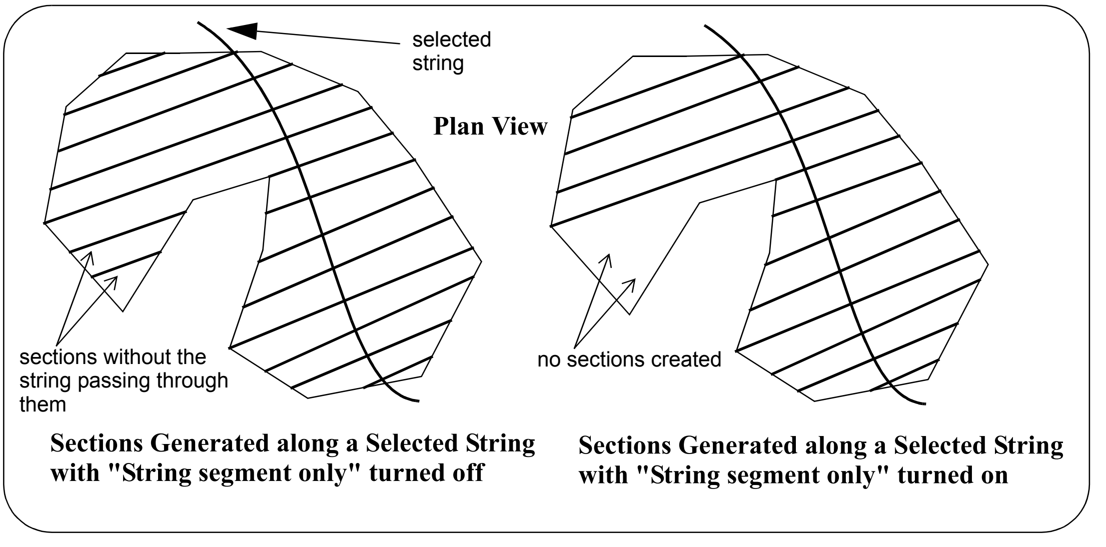
23.3.4 End Area - String Tin to Tin

Position of option on menu: **Design =>Volumes =>End Area =>String tin to tin**

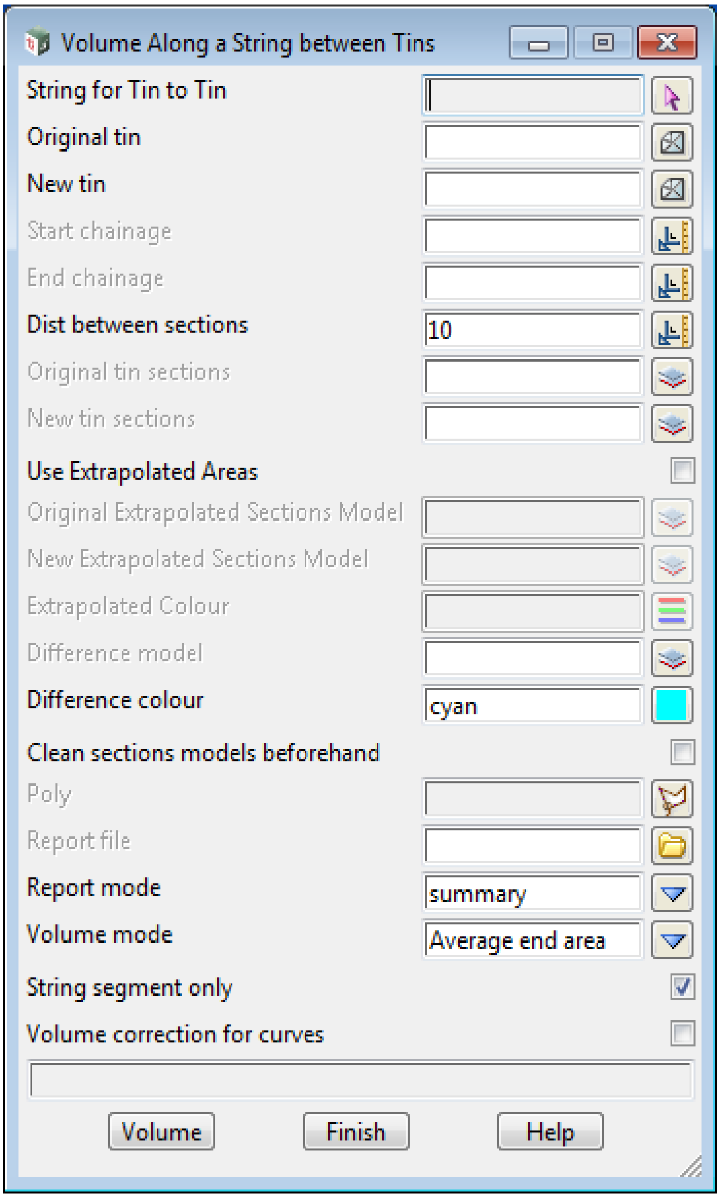
In this option, the sections generated for collating areas and volumes are defined to be at right angles to a *user selected string*. This method is commonly used in road calculations.

It optionally uses the *entire* section within the polygon or only the sections or part of sections that contain the user selected string.

For more information on the end area volume calculations, go to the section 23.2.1 Theory of End Area Volumes.



On selecting the **String Tin to Tin** option, the **Volume Along a String Between Tins** panel is displayed.



The fields and buttons in this panel are used as follows.

Field Description   Type   Defaults   Pop-Up

String for tin to tin   string select

the string for defining the sections used in the end area calculations. The sections are taken at right angles to this string.

Original tin   input   available tins

name of the original tin for determining volumes.

New tin   input   available tins

name of the new tin for determining volumes.

Start chainage   input

if non blank, the chainage of the first section to use for volume calculations.
If blank, start with the section with the lowest chainage.

End chainage   input

if non blank, the chainage of the last section to use for volume calculations.
If blank, end with the section with the highest chainage.

Dist between sections input 10.0 1,10,100

distance between the lines to section along

Original tin sections input available models

if non-blank, the sections though the original tin used for the end area calculations are retained and placed in the model given in this field. The sections are given the same colour at the original tin.
If blank, the sections are not kept.

New tin sections input available models

if non-blank, the sections though the new tin used for the end area calculations are retained and placed in the model given in this field. The sections are given the same colour as the new tin.
If blank, the sections are not kept.

Use extrapolated areas tick box not ticked

*if **not ticked**, areas (and hence volumes) are only calculated where **both** sections exist. Hence the sections are limited to where both occur.*

*If **tick**, when the sections are not the same length, the end points of the above and below sections are connected thereby extrapolating the smaller section. This method is not recommended since data does not exist.*

Original extrapolated sections model model box available models

if non-blank, the extrapolated sections created from the Original sections are placed in this model.
If blank, the extrapolated sections for the original sections are not kept.

New extrapolated sections model model box available models

if non-blank, the extrapolated sections created from the New sections are placed in this model.
If blank, the extrapolated sections for the new sections are not kept.

Extrapolated colour colour box available colours

colour for the extrapolated sections

Difference model input available models

if non-blank, the sections which are the difference of the original and the new sections are retained and placed in the model given in this field.
If blank, the sections are not kept.

Difference colour input available colours

colour for the difference sections strings

Clean sections models beforehand tick box

if ticked, the model of sections and difference sections are cleaned out before the option runs.

Report file input *.rpt

name of the file to contain the volume report. If the file already exists, the report will be appended to the file. If no name is given, no report is produced.

Report mode input summary summary, full

*if **full**, the cut and fill details for every section are included.*
*If **summary**, just the cut and fill totals are given.*

Volume mode choice box Average end area average end area,
prismoidal - 2 sections

the two methods in 12d for calculating volumes using areas of sections. For more information on each *method, go to* 23.2.1.1 Average End Area Formula

String segment only tick box ticked

*if **ticked**, only the part of the section within the polygon that contains the selected string is used. If **not ticked**, the whole section within the polygon is used (see* 23.2.1 Theory of End Area Volumes*).*

Volume correction for curves tick box

if ticked, volume corrections are made when going around curves.

Poly poly string-select

if selected, the string, rectangle or lasso used as the bounding polygon for the volume calculations.

Volume button

The volume between the original and new tins within the selected bounding polygon is calculated by the end area method. The sections through the tins are calculated at right angles to the selected string at a separation given by the dist *field.*

How to Use the Panel and Panel Messages

(a) Select the bounding polygon string by choosing **Poly** and pick the required string.

(b) Select the string to define the sections string by choosing **String for tin to tin**

(c) Volume processing begins on selecting the **Volume** button.

Progress messages - sent to the panel message area

calculating volumes

Completion message - sent to the panel message area

c *cut volume* f *fill volume* bal *total (balance) volume*

<Esc> can be used to terminate the option during volume calculations.

Continue to the next section 23.3.4 End Area - String Tin to Tin or return to 23.3 End Area.

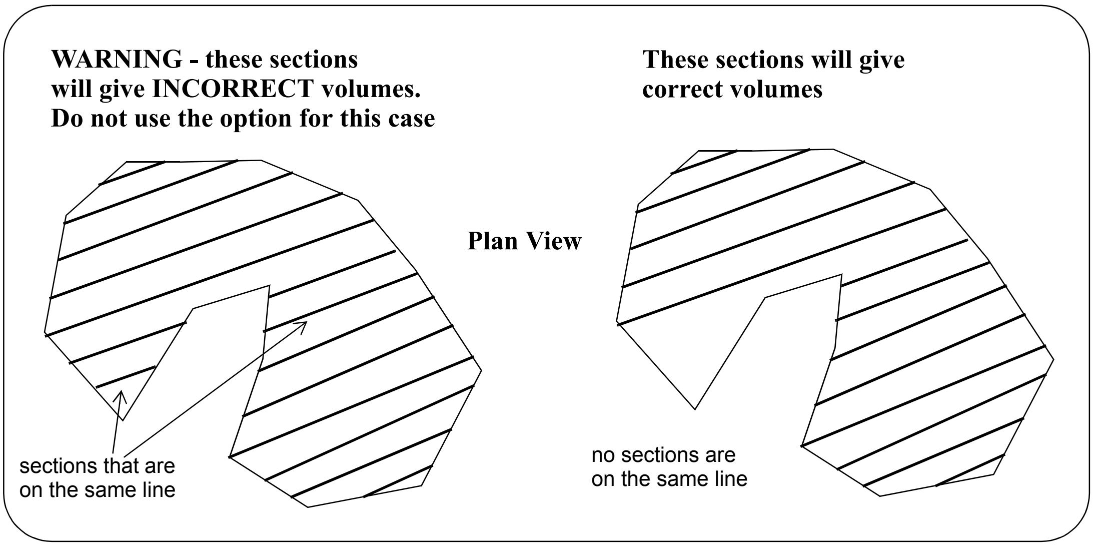
23.3.5 End Area - Sections to Height

Position of option on menu: **Design =>Volumes =>End Area =>Sections to height**

The **Sections to height** option is used to find volumes between sections that already exist (in a given model) and a user supplied height. The volumes can be restricted to be within a user supplied polygon.

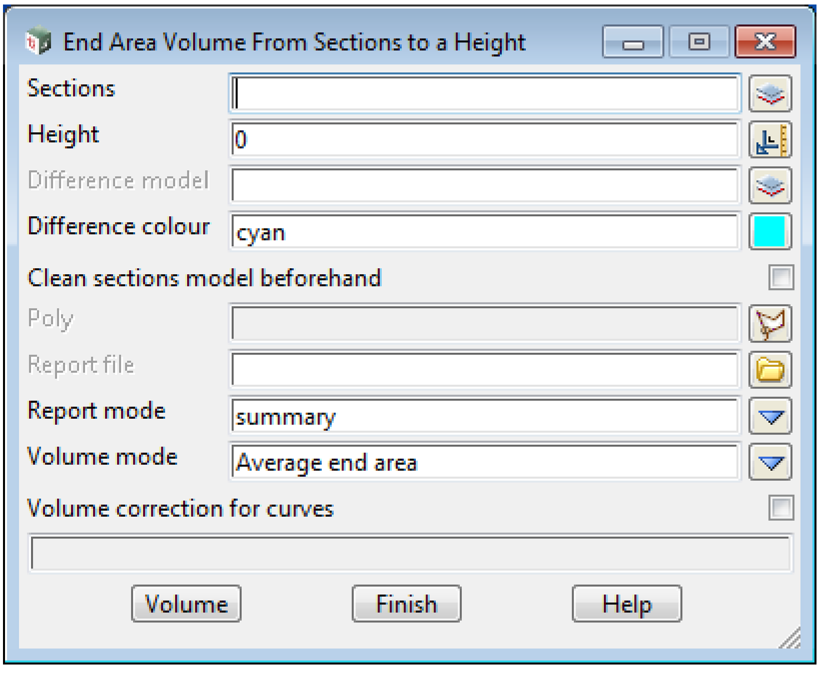
Normally the ***Sections to Height*** method is not used since if the tin exists, the option
Design =>Volumes =>End Area =>Tin to height
automatically generates sections through the tin at the required interval.

WARNING: The ***Sections to Height*** method can ***not*** be used when there are cases of two sections being on the same line.



For more information on the end area volume calculations, go to the section 23.2.1 Theory of End Area Volumes.

On selecting the **Sections to height** option, the **End Area Volume from Tin to a Height** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Sections input available models

name of the model containing the sections for which the volume between them and a height (z value) will be calculated.

Height input

the volume is calculated between the sections and the value in this field.

Difference model input available models

if non-blank, the sections which are the difference between the given sections and the height are retained and placed in the model given in this field.
If blank, the sections are not kept.

Difference colour input available colours

colour for the difference sections strings

Clean sections models beforehand tick box

if ticked, the model of sections and difference sections are cleaned out before the option runs.

Poly poly string-select

if selected, the string, rectangle or lasso used as the bounding polygon for the volume calculations.

Report file input *.rpt

name of the file to contain the volume report. If the file already exists, the report will be appended to the file. If no name is given, no report is produced.

Report mode input summary summary, full

*if **full**, the cut and fill details for every section are included.*
*In **summary**, just the totals are given.*

Volume mode choice box Average end area average end area, prismoidal - 2 sections

the two methods in 12d for calculating volumes using areas of sections. For more information on each method, go to 23.2.1.1 Average End Area Formula

Volume correction for curves tick box

if ticked, volume corrections are made when going around curves.

Volume button

the volume between the model of sections and the height (z-value) within the selected bounding polygon is calculated by the end area method.

<Esc> can be used to terminate the option during volume calculations.

Continue to the next section 23.3.6 End Area - Sections to Sections or return to 23.3 End Area.

23.3.6 End Area - Sections to Sections

Position of option on menu: **Design =>Volumes =>End Area =>Sections to sections**

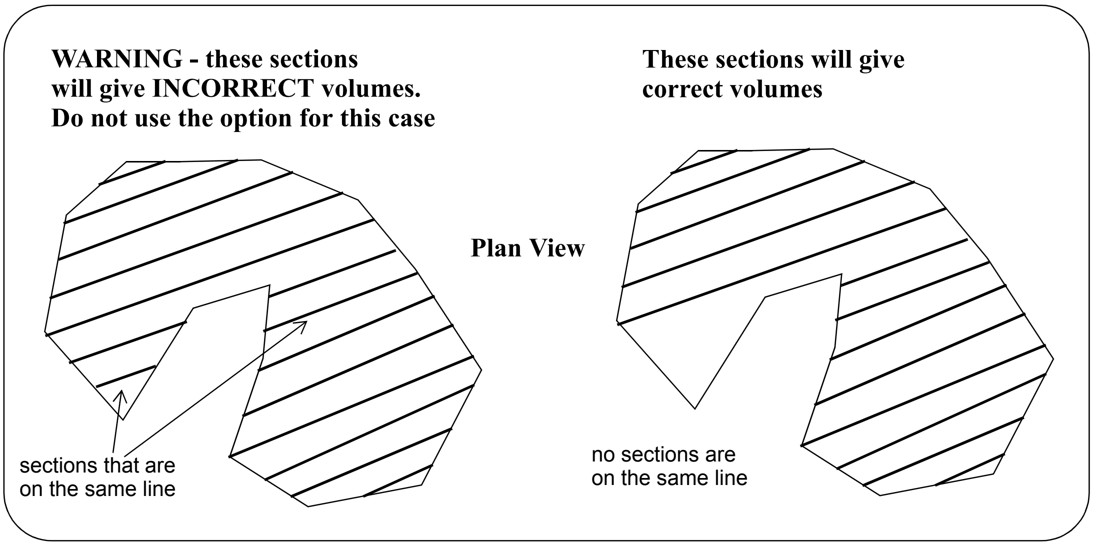
The **Sections to sections** option is used to find volumes between two models of sections. The volumes can be restricted to be within a user supplied polygon. Note that there needs to be matching sections in the two models of section.

Normally the ***Section to Sections*** method is not used since if the tin exists, the option

Design =>Volumes =>End Area =>Tin to tin

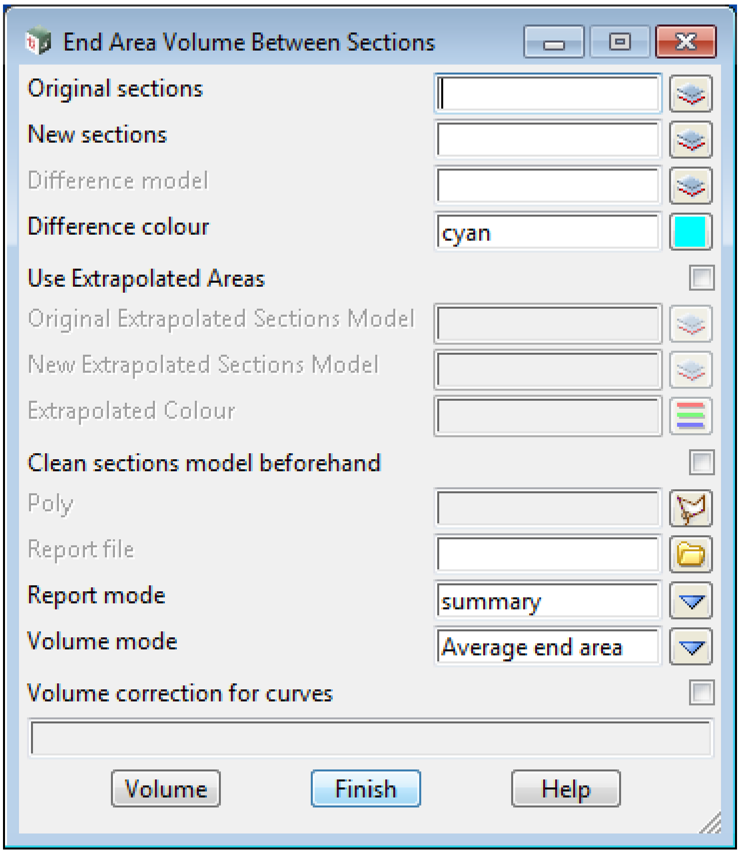
automatically generates sections through the two tins at the required interval.

WARNING: ***The Section to Sections*** method can ***not*** be used when there are cases of two sections being on the same line.



For more information on the end area volume calculations, go to the section 23.2.1 Theory of End Area Volumes.

On selecting the **Sections to sections** option, the **End Area Volume Between Sections** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Original sections	input		available models

name of the original model of sections for determining volumes.

New sections	input		available models

name of the new model of sections for determining volumes.

Difference model	input		available models

if non-blank, the sections which are the difference of the original and the new sections are retained and placed in the model given in this field. If blank, the sections are not kept.

Difference colour	input		available colours

colour for the difference sections strings

Use extrapolated areas	tick box	not tick	

*if **not tick**, areas (and hence volumes) are only calculated where **both** sections exist. Hence the sections are limited to where both occur.*

*If **tick**, when the sections are not the same length, the end points of the above and below sections are connected thereby extrapolating the smaller section. This method is not recommended since data does not exist.*

Original extrapolated sections model	model box	available models

if non-blank, the extrapolated sections created from the Original sections are placed in this model. If blank, the extrapolated sections for the original sections are not kept.

New extrapolated sections model	model box	available models

if non-blank, the extrapolated sections created from the New sections are placed in this model. If blank, the extrapolated sections for the new sections are not kept.

Extrapolated colour colour box available colours

colour for the extrapolated sections

Clean sections models beforehand tick box

if ticked, the model of sections and difference sections are cleaned out before the option runs.

Poly poly string-select

if selected, the string, rectangle or lasso used as the bounding polygon for the volume calculations.

Report file input

name of the file to contain the volume report. If the file already exists, the report will be appended to the file. If no file name is given, the report is not produced.

Report mode input summary summary, full

*if **full**, the cut and fill details for every section are included.*
*If **summary**, just the cut and fill totals are given.*

Volume mode choice box Average end area average end area,

prismoidal - 2 sections

the two methods in 12d for calculating volumes using areas of sections. For more information on each method, go to 23.2.1.1 Average End Area Formula

Volume correction for curves tick box

if ticked, volume corrections are made when going around curves.

Volume button

The volume between the original model of sections and the new model of sections within the selected bounding polygon is calculated by the end area method.

<Esc> can be used to terminate the option during volume calculations.

Continue to the next section 23.3.7 End Area - Tin to Sections or return to 23.3 End Area.

23.3.7 End Area - Tin to Sections

Position of option on menu: **Design =>Volumes =>End Area =>Tin to sections**

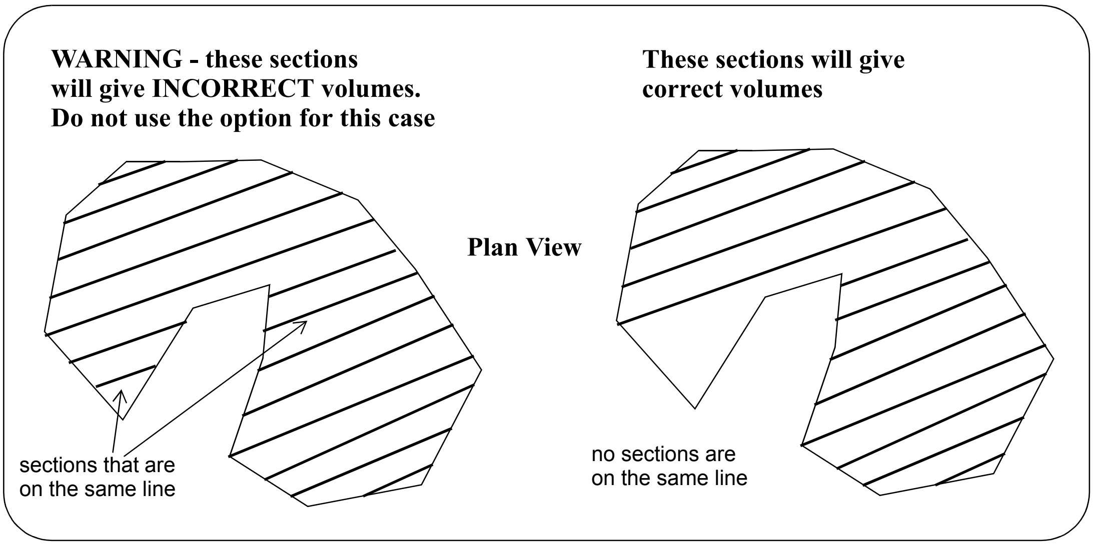
The **Tin to sections** option is used to find volumes between a tin and a model of sections. The volumes can be restricted to be within a user supplied polygon.

Normally the ***Tin to Sections*** method is not used since if the tin for the sections exists, the option

Design =>Volumes =>End Area =>Tin to tin

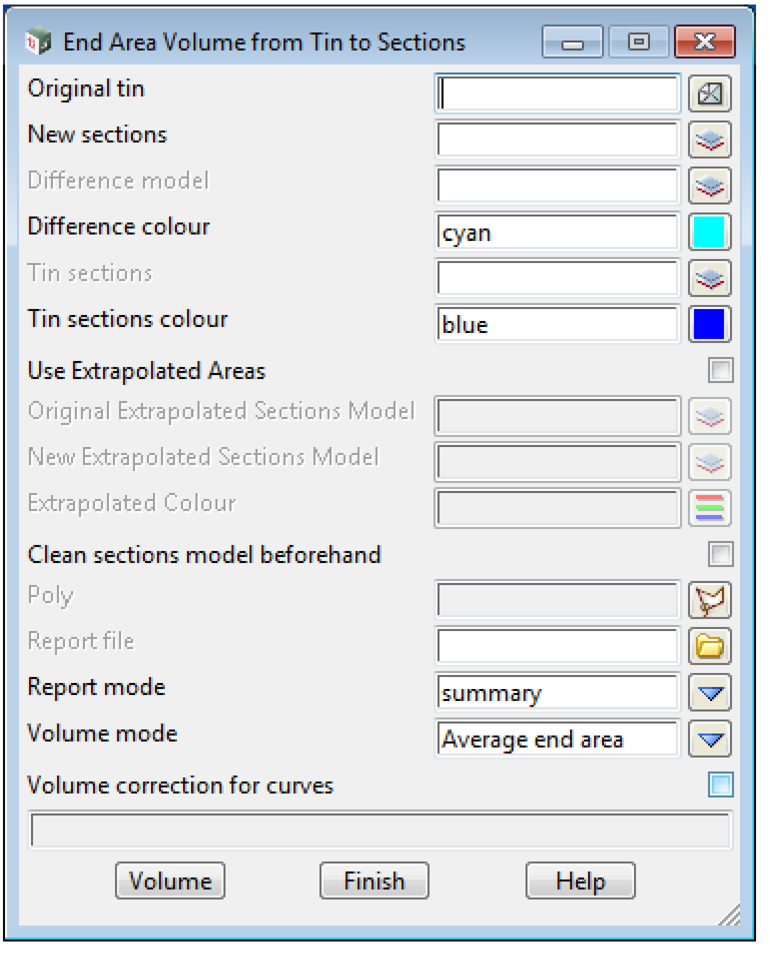
automatically generates sections through the two tins at the required interval.

WARNING: ***The Tin to Sections*** method can ***not*** be used when there are cases of two sections being on the same line.



For more information on the end area volume calculations, go to the section 23.2.1 Theory of End Area Volumes.

Selecting **Tin to sections** brings up the **End Area Volume from Tin to Sections** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Original Tin	tin box		available tins

name of the original tin to use for determining volumes.

New sections	model box		available models

name of the new model of sections for determining volumes.

Difference model	model box		available models

if non-blank, the sections which are the difference of the original and the new sections are retained and placed in the model given in this field. If blank, the sections are not kept.

Difference colour	colour box		available colours

colour for the difference sections strings

Tin sections	model box		available models

if non-blank, the sections cut through the Original tin are placed in this model.
If blank, the sections through the Original tin are not kept.

Tin sections colour	colour box		available colours

colour for the sections cut through the Original tin

Use extrapolated areas	tick box	not ticked	

*if **not ticked**, areas (and hence volumes) are only calculated where **both** sections exist. Hence the* *sections are limited to where both occur.*

*If **ticked**, when the sections are not the same length, the end points of the above and below sections are connected thereby extrapolating the smaller section. This method is not recommended since data does not exist.*

Original extrapolated sections model model box available models

if non-blank, the extrapolated sections created from the Original sections are placed in this model. If blank, the extrapolated sections for the original sections are not kept.

New extrapolated sections model model box available models

if non-blank, the extrapolated sections created from the New sections are placed in this model. If blank, the extrapolated sections for the new sections are not kept.

Extrapolated colour colour box available colours

colour for the extrapolated sections

Clean sections models beforehand tick box

if ticked, the model of sections and difference sections are cleaned out before the option runs.

Poly poly string-select

if selected, the string, rectangle or lasso used as the bounding polygon for the volume calculations.

Report file input

name of the file to contain the volume report. If the file already exists, the report will be appended to the file. If no file name is given, the report is not produced.

Report mode input summary summary, full

*if **full**, the cut and fill details for every section are included.*
*If **summary**, just the cut and fill totals are given.*

Volume mode choice box Average end area average end area,

prismoidal - 2 sections

the two methods in 12d for calculating volumes using areas of sections. For more information on each method, go to 23.2.1.1 Average End Area Formula

Volume correction for curves tick box

if ticked, volume corrections are made when going around curves.

Volume button

The volume between the original model of sections and the new model of sections within the selected bounding polygon is calculated by the end area method.

<Esc> can be used to terminate the option during volume calculations.

Continue to the next section 23.3.8 End Area - Material Section to Section or return to 23.3 End Area.

23.3.8 End Area - Material Section to Section

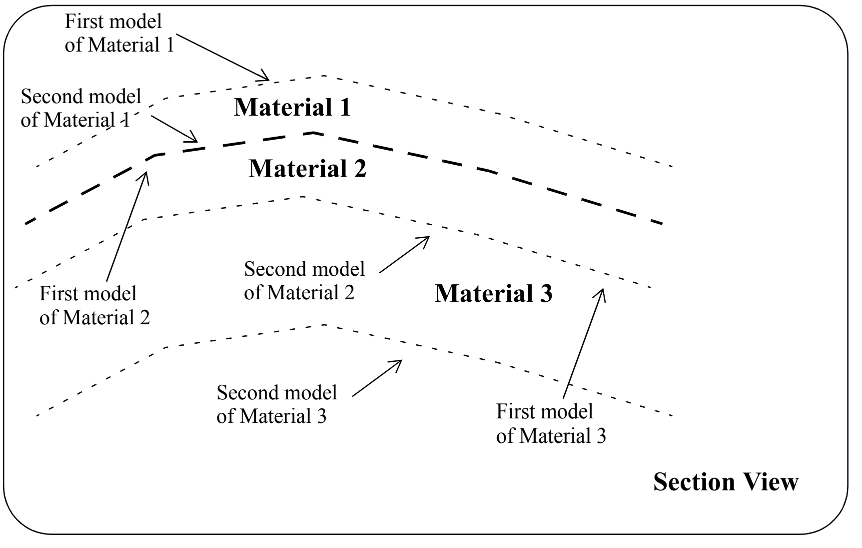
Position of option on menu: **Design =>Volumes =>End Area =>Material section to section**

The **Material section to section** option calculates the volumes for one or more material layers where a material layer is given by two models of sections representing the top (First model) and the bottom (Second model) of the material layer.

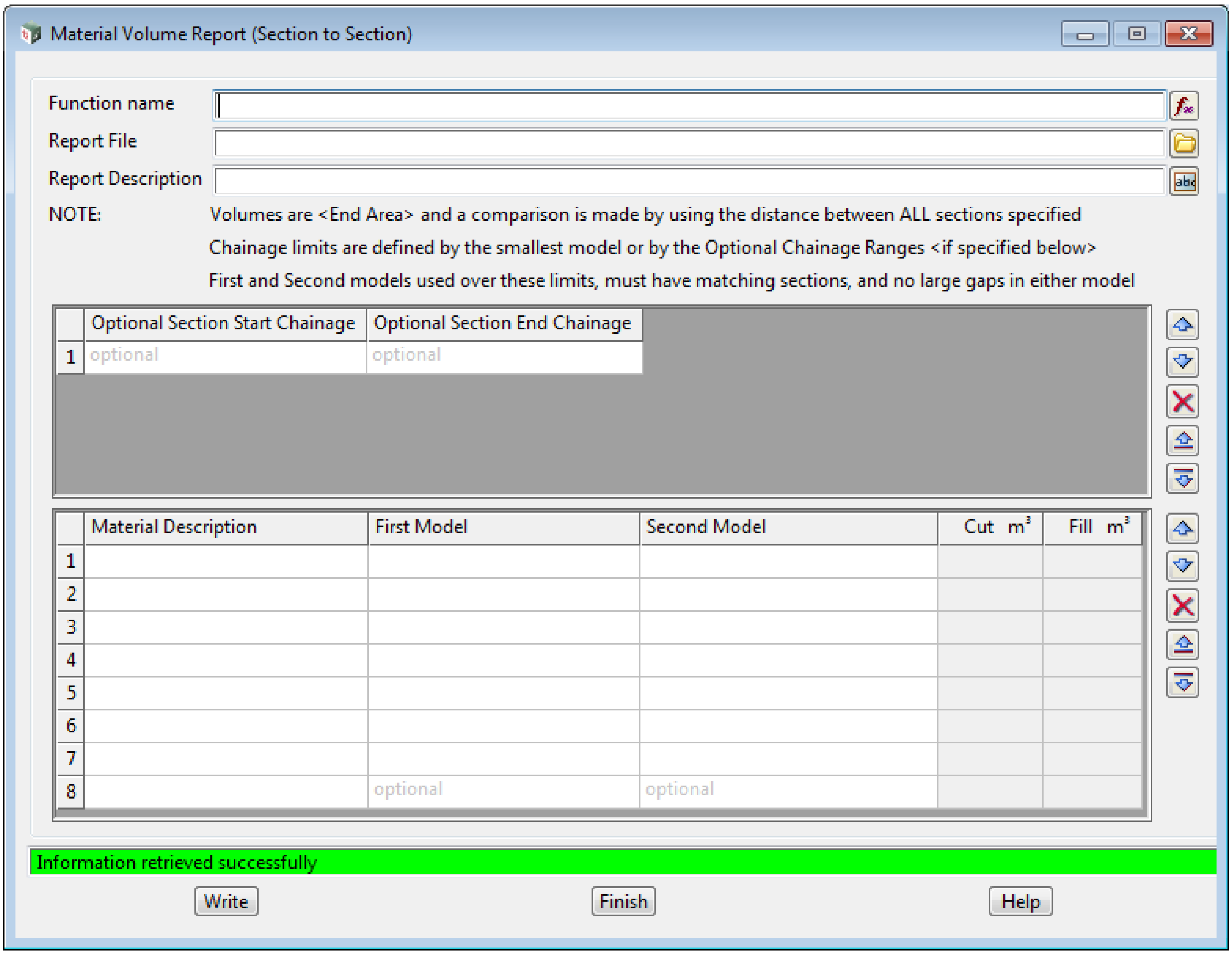
The cut and fill volumes for each material layer is written to a table on the panel, and optionally to a report.

This option uses models of sections already created by other options such as *Apply MTF* or **Tins =>Sections =>X sections**

The cut and fill volumes are calculated using the ***End Area*** method (see 23.2.1 Theory of End Area Volumes).



Selecting **Material section to section** brings up the **Material Volume Report (Section to Section)** panel



Volumes are End Area and a comparison is made by using the distance between ALL the sections specified.

The chainage limits for the volume calculations are defined by the smallest chainages on the sections in the ***First*** and ***Second Models,*** or by the ***Optional Section Start Chainage*** and ***Optional Section End Chainage*** ranges given in the panel.

The ***First*** and ***Second Models*** used over these limits, must have matching sections, and no large gaps in either model.

If the chainage limit in the smaller model falls within the limits of the chainages in the larger model and no optional chainages are used, then the chainage limit of the smaller model is used.

In this case no interpolation is done up to the start or after the end of the smaller model chainage limits, in relation to the larger model.

However, if there is a large chainage gap in the smaller model then some interpolation will be done between the first and last section in this area, in relation to the larger model.

The latter is not desirable and the optional chainage ranges should be used to take into account the chainage gap in the smaller model

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function Name	function box		

if non-blank entry used as the name of a function that can be used in chains etc

Report File	file box		available *.csv files

if non-blank entry used as the report file in a csv format

Report Description input

if non-blank entry used a descriptive heading in the report

Optional Chainages Grid

Section Start Chainage real

if non-blank entry used as the start chainage in the report

Section End Chainage real

if non-blank entry used as the end chainage in the report

Report Options Grid

Material Description input

if non-blank entry used as a description for a material, such as "Asphalt Surface" in the report

First Model model

if non-blank entry this section model is used in the volume calculations

Second Model model

if non-blank entry this section model is used in the volume calculations

Cut m^3

field for display of results only, after the "Write" button is used.

If more than one <Optional Chainage Grids> are used, then the values displayed represents the accumulative cut for that material

Fill m^3

field for display of results only, after the "Write" button is used.

If more than one <Optional Chainage Grids> are used, then the values displayed represents the accumulative fill for that material

Write button

the end area volumes is run for each of the Optional Chainage ranges or for the extents of the smaller model, if no optional chainages are specified

The volumes are calculated for each material using a chain from the Library. This can be verified in the output window.

Starting chain: Vol Sect to Sect.chain

Return to 23.3 End Area.

23.4 Exact

Position of menu: **Design =>Volumes =>Exact**

These volumes are calculated by dividing the tins up into small prisms whose volumes can be calculated exactly. For more information on the exact volume calculations, go to the section 23.2.2 Theory of Exact Volumes

In the exact volume options, the volumes calculations can be restricted to a user defined polygon or if no polygon is provided, the volumes will be only be calculated for the regions where both the triangulations exist.

With the exact method, volumes can easily be calculated for depth bands from

(a) the given height to the tin for "tin to height" volumes

or

(b) the new tin to the original tin for "tin to tin" volumes.

Hence for the calculations, a range file can be supplied giving depth pairs and 12d Model will report the cut, fill and balance for each of the depth pairs and also the total of the cut, fill and balance for all the pairs.

In the report, the sign for cut (negative or positive) is given by the cut volume sign from the **default settings** panel (fill will have the opposite sign).

The range file consists of a list of depth ranges and colours, one set per line, in the format

lower_depth   upper_depth   depth_colour

This line represents all depths satisfying

lower_depth <= depth < upper_depth.

For ***each range*** in the file, the total cut and fill volumes for the depth range will be reported on, and the depth_colour can be used to colour all areas on a plan view satisfying the range.

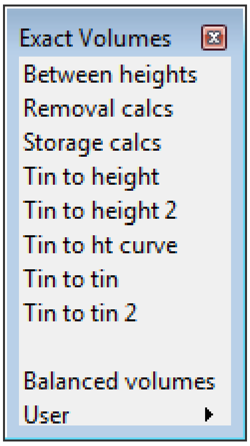
IMPORTANT NOTE - when are depth file is used, the ***volume totals*** are **only** calculated for the ***depths in the range file***. If the depth ranges ***does not cover*** the entire depth difference between the tins then the totals which are the sum of the different depth ranges will ***not*** be the same as the volumes between the two tins or the volumes from a tin to a height.

Example of a Depth Range File

// depth range file
// format: lower_depth upper_depth colour_for_depth_range
// depth is measured positive down and negative up.

-20 -10 red // colour red where the depth is greater or equal to -20 and less than -10
-10 0 magenta // colour magenta where the depth is greater or equal to -10 and less than 0
0 10 green // colour green where the depth is greater or equal to 0 and less than 10
10 20 "dark green" // colour dark green where depth is >=10 and less than 20

The **Exact** walk-right menu is



volumes between two heights
cut/slope vol calcs for height range
fill/slope vol calcs for height range
volumes between surface and a height
" " "

volumes between surfaces
" " "

The options in this menu will now be described.

For the option see

23.4.1 Exact - Between Heights
23.4.2 Exact - Removal Calcs
23.4.3 Exact - Storage Calcs
23.4.4 Exact - Tin to Height
23.4.5 Exact - Tin to Height 2
23.4.6 Exact - Tin to Height Curve
23.4.7 Exact - Tin to Tin
23.4.8 Exact - Tin to Tin 2
23.4.9 Exact - Balanced Volumes

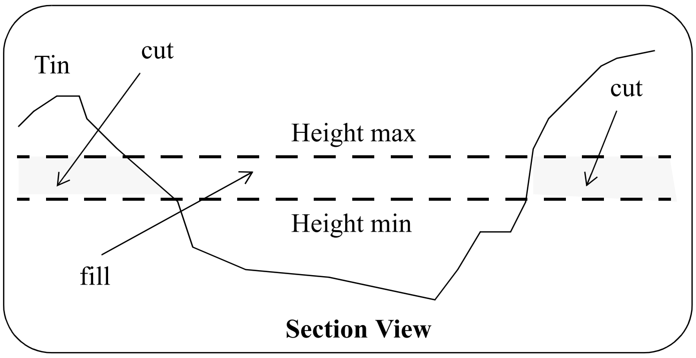
For more information on the exact volume calculations, go to the section 23.2.2 Theory of Exact Volumes

23.4.1 Exact - Between Heights

Position of option on menu: **Design =>Volumes =>Exact =>Between heights**

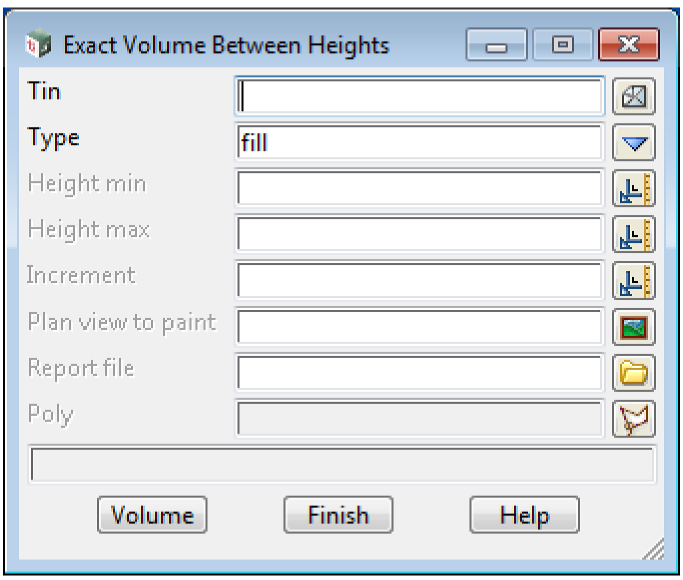
This panel is used to calculate the volume between two heights on a given tin.

The calculations can be restricted to within a polygon, or if no polygon is selected, the entire tin is used.



For more information on the exact volume calculations, go to the section 23.2.2 Theory of Exact Volumes

On selecting the **Between heights** option, the **Exact Volume Between Heights** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	input		available tins

name of the tin for which the volume between two heights will be calculated.

Field Description	Type	Defaults	Pop-Up
Type	input	fill	cut, fill

*if **fill**, the fill volume between the two heights is calculated.*
*If **cut**, the cut volume between the two heights is calculated.*

Height min/max input

volume is calculated between the height min *and the* height max *values.*

Increment input

the volumes between the height min *and* height max *are reported on and is broken up into intervals given by the* increment *value.*

Plan view to paint	input	available views

*if non-blank, the region used for calculations will be painted in the given plan view. The colour will be green if type is **fill**, or red if type is **cut**.*

Report file	input	*.rpt

name of the file to contain the volume report. If the file already exists, the report will be appended to the file. If no name is given, no report is produced.

Poly	poly string-select	

if selected, this string is used as the bounding polygon for the volume calculations.

Volume	button	

The volume between the height min *and* height max *for the tin within the selected bounding polygon is calculated by the exact method.*

<Esc> can be used to terminate the option during volume calculations.

Continue to the next section 23.4.2 Exact - Removal Calcs or return to 23.4 Exact.

23.4.2 Exact - Removal Calcs

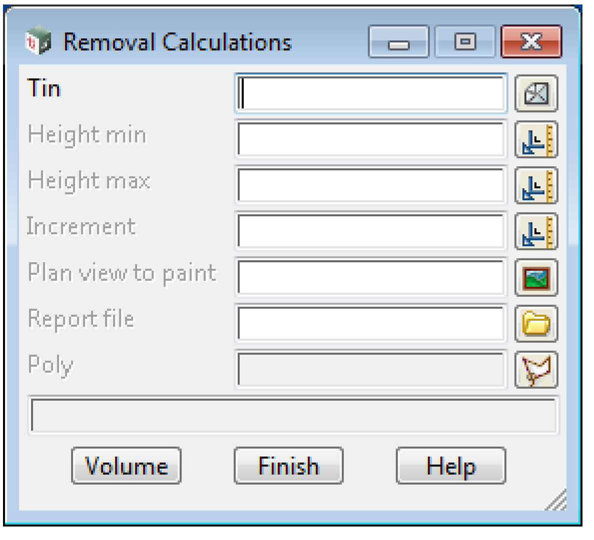
Position of option on menu: **Design =>Volumes =>Exact =>Removal calcs**

For a user specified range of heights, this option will calculate the cut volume from each height to a given tin, plus the plan and slope areas for the region of the tin exposed by cutting to the height.

The calculations can be restricted to within a polygon, or if no polygon is selected, the entire tin is used.

For more information on the exact volume calculations, go to the section 23.2.2 Theory of Exact Volumes

On selecting the **Removal calcs** option, the **Removal Calculations** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	input		available tins
name of the tin used in the volume calculations.			
Height min/max	input		
if non-blank, the minimum/maximum value of the height range to calculate volumes to. if blank, the tin's minimum/maximum z-value is used.			
Increment	input		
increment between the heights to calculate the volumes to.			
Plan view to paint	input		available views
if non-blank, the region used for calculations will be painted red in the given plan view.			
Report file	input		*.rpt
name of the file to contain the volume report. If the file already exists, the report will be appended to the file. if no name is given, no report is produced.			
Poly	poly string-select		
if selected, this string is used as the bounding polygon for the volume calculations.			
Volume	button		

calculate the cut volumes from the tin to the height for the required heights in the height min *and* height max *range.*

<Esc> can be used to terminate the option during volume calculations.

Continue to the next section 23.4.3 Exact - Storage Calcs or return to 23.4 Exact.

23.4.3 Exact - Storage Calcs

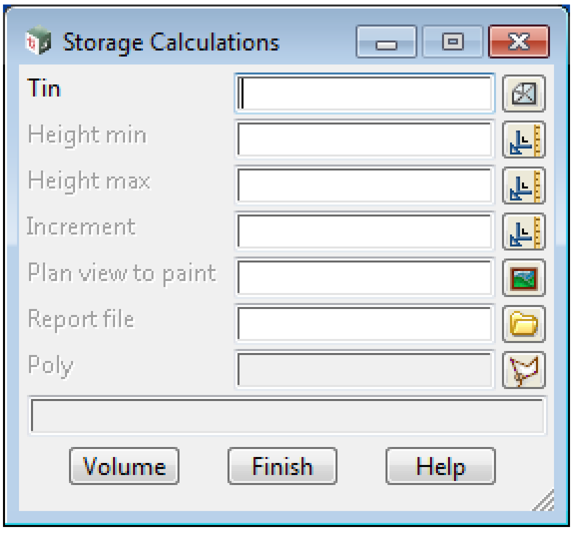
Position of option on menu: **Design =>Volumes =>Exact =>Storage calcs**

For a user specified range of heights, this option will calculate the fill volume from each height to a given tin, plus the plan and slope areas for the region of the tin covered by filling to the height.

The calculations can be restricted to within a polygon, or if no polygon is selected, the entire tin is used.

For more information on the exact volume calculations, go to the section 23.2.2 Theory of Exact Volumes

On selecting the **Storage calcs** option, the **Storage Calculations** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin used in the volume calculations.

Height min/max	input		

if non-blank, the minimum/maximum value of the height range to calculate volumes to.
if blank, the tin's minimum/maximum z-value is used.

Increment	input	1	

increment between the heights to calculate the volumes to.

Plan view to paint	view box		available views

if non-blank, the region used for calculations will be painted green in the given plan view.

Report file	file box		*.rpt

name of the file to contain the volume report. If the file already exists, the report will be appended to the file. If no name is given, no report is produced.

Poly	poly string-select		

if selected, this string is used as the bounding polygon for the volume calculations.

Volume	button		

calculate the fill volumes from the tin to the height for the required heights in the height min *and* height max *range.*

<Esc> can be used to terminate the option during volume calculations.

Continue to the next section 23.4.4 Exact - Tin to Height or return to 23.4 Exact.

23.4.4 Exact - Tin to Height

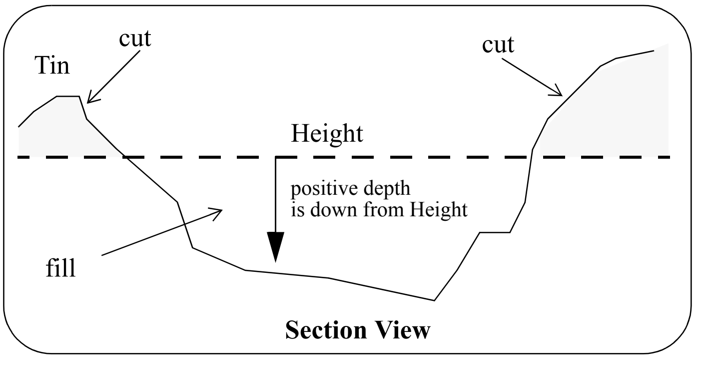
Position of option on menu: **Design =>Volumes =>Exact =>Tin to height**

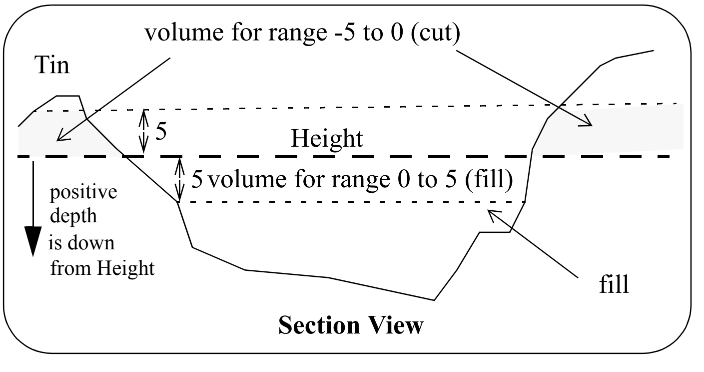
This panel is used to calculate the volume between a tin and a given height.

The calculations can be restricted to within a polygon, or if no polygon is selected, the entire tin is used.

The calculated volumes can be produced and reported over user supplied depth ranges. Similarly, a view can be coloured on a depth basis using the same **range** file.

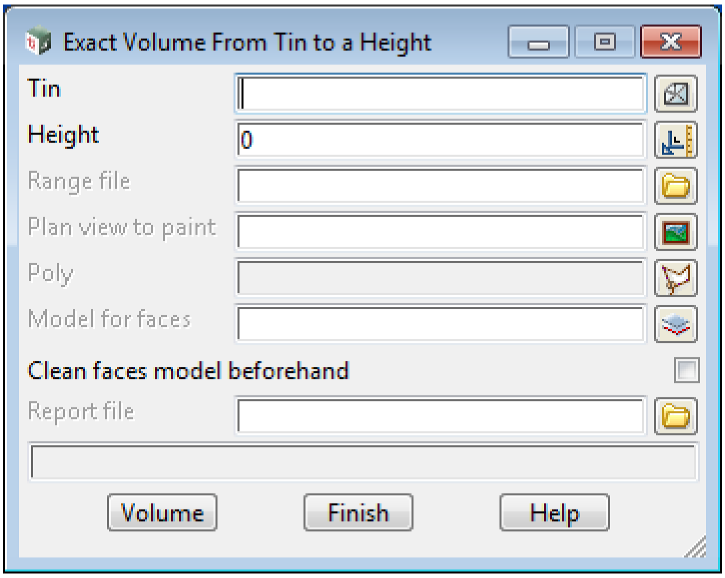
IMPORTANT NOTE - the ***volume totals*** are **only** calculated for the ***depths in the range file***. If the depth ranges ***does not cover*** the entire depth difference between the tin and the height then the totals which are the sum of the different depth ranges will ***not*** be the same as the volume between the tin and the height.





For more information on the exact volume calculations, go to the section 23.2.2 Theory of Exact Volumes

On selecting the **Tin to height** option, the **Exact Volume From Tin to a Height** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	input		available tins

name of the tin for which the volume between it and a height (z value) will be calculated.

Height	input		

the volume is calculated between the tin and the value in this field.

Range file	input		*.drf

if non-blank, the user supplied depth range file is used to split up the volumes report and define the depth colours used for painting a view.

Plan view to paint	input		available views

if non-blank, the given plan view will be painted according to the depth colours given in the range file.

Poly	poly string-select		

if selected, this string is used as the bounding polygon for the volume calculations.

Model for faces	input		available models

if non-blank, faces will be created with colours according to the range colours given in the range file.

Clean faces models beforehand	tick box		

if ticked, the model of faces is cleaned out before the option runs.

Report file	input		*.rpt

name of the file to contain the volume report. If the file already exists, the report will be appended to the file. If no name is given, no report is produced.

Volume	button		

The volume between the tin and the height (z-value) for the given range file within the selected bounding polygon is calculated by the exact method. If a range file does not exist, the volumes between the tin and the height is calculated.

<Esc> can be used to terminate the option during volume calculations.

Continue to the next section 23.4.5 Exact - Tin to Height 2 or return to 23.4 Exact.

23.4.5 Exact - Tin to Height 2

Position of option on menu: **Design =>Volumes =>Exact =>Tin to height 2**

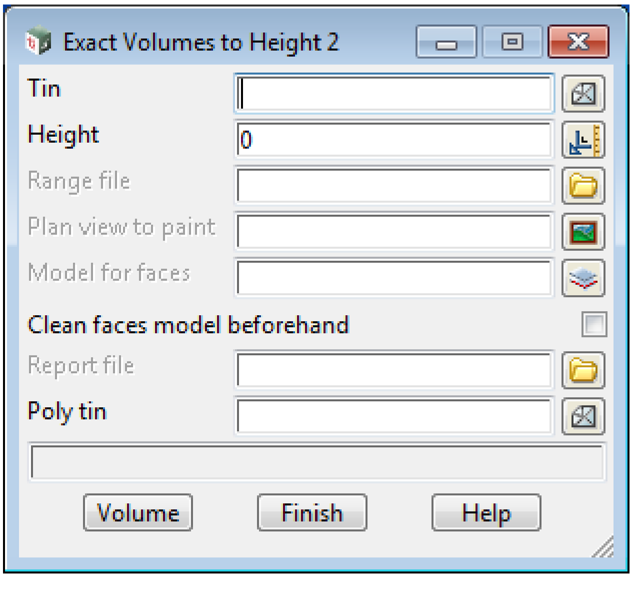
This panel is used to calculate the volume between a tin and a given height but the calculations are restricted to only the regions defined by another tin rather than to a polygon.

The calculated volumes can be produced and reported over user supplied depth ranges. Similarly, a view can be coloured on a depth basis using the same **range** file.

IMPORTANT NOTE - the ***volume totals*** are **only** calculated for the ***depths in the range file***. If the depth ranges ***does not cover*** the entire depth difference between the tin and the height then the totals which are the sum of the different depth ranges will ***not*** be the same as the volume between the tin and the height.

For more information on the exact volume calculations, go to the section 23.2.2 Theory of Exact Volumes

On selecting the **Tin to height 2** option, the **Exact Volumes to Height 2** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin for which the volume between it and a height (z value) will be calculated.

Height	input		

the volume is calculated between the tin and the value in this field.

Range file	input		*.drf

if non-blank, the user supplied depth range file is used to split up the volumes report and define the depth colours used for painting a view.

Plan view to paint	view box		available views

if non-blank, the given plan view will be painted according to the depth colours given in the range file.

Model for faces	model box		available models

if non-blank, faces will be created with colours according to the range colours given in the range file.

Clean faces models beforehand tick box

if ticked, the model of faces is cleaned out before the option runs.

Report file file box *.rpt

name of the file to contain the volume report. If the file already exists, the report will be appended to the file. If no name is given, no report is produced.

Poly tin tin box available tins

name of the tin to define the regions that the volume calculations are restricted to.

Volume button

The volume between the tin and the height (z-value) for the given range file within the selected bounding polygon is calculated by the exact method. If a range file does not exist, the volumes between the tin and the height is calculated.

<Esc> can be used to terminate the option during volume calculations.

Continue to the next section 23.4.6 Exact - Tin to Height Curve or return to 23.4 Exact.

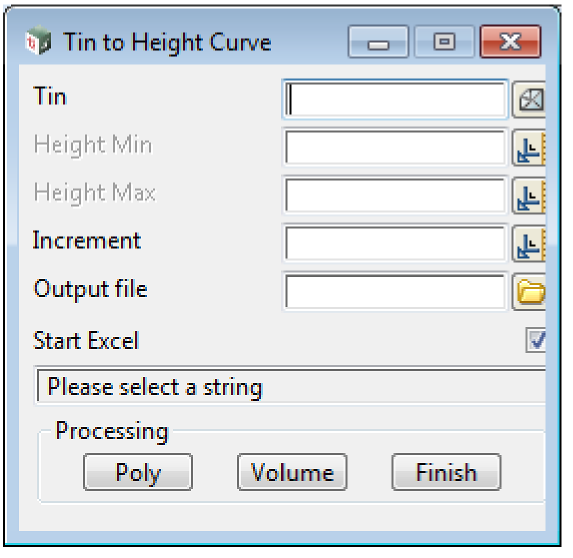
23.4.6 Exact - Tin to Height Curve

Position of option on menu: **Design =>Volumes =>Exact =>Tin to ht curve**

This panel is used to calculate the volume between a tin between heights and writes out a report on the volumes.

For more information on the exact volume calculations, go to the section 23.2.2 Theory of Exact Volumes

On selecting the **Tin to ht curve** option, the **Tin to Height Curve** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins

name of the tin for which the volume between it and a height (z value) will be calculated.

Height min input

the start height for calculation volumes.

Height max input

the end height for calculation volumes.

increment input

the height increment.

Output file file box

the file for the volumes report.

Start Excel tick box tick

if ticked, start Excel up with the report loaded into it.

Poly select

select the polygon which is used in plan only to restrict the area for calculating volumes over.

Volume button

calculate the volume between heights differing by the given increment, and write out a report.

Continue to the next section 23.4.7 Exact - Tin to Tin or return to 23.4 Exact.

23.4.7 Exact - Tin to Tin

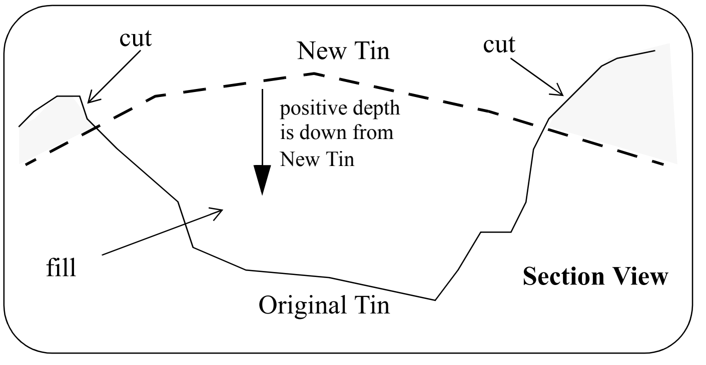
Position of option on menu: **Design =>Volumes =>Exact =>Tin to tin**

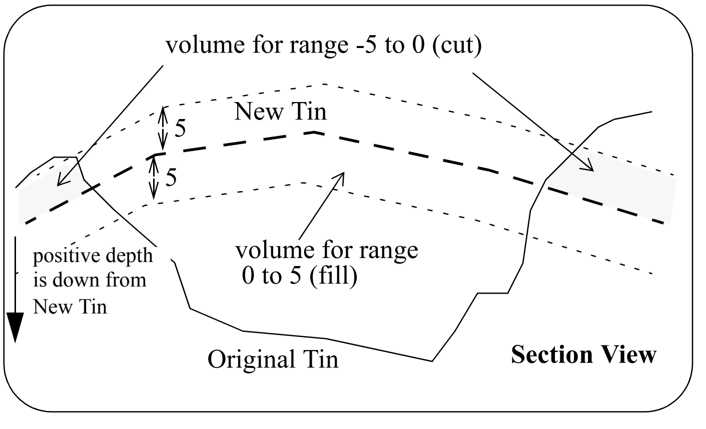
This panel is used to calculate the volume between two tins using the exact method.

The calculations can be restricted to within a polygon, or if no polygon is selected, the overlapping sections of the two tins is used.

The calculated volumes can be produced and reported on over user supplied depth ranges. Similarly, a view can be coloured on a depth basis using the same **range** file.

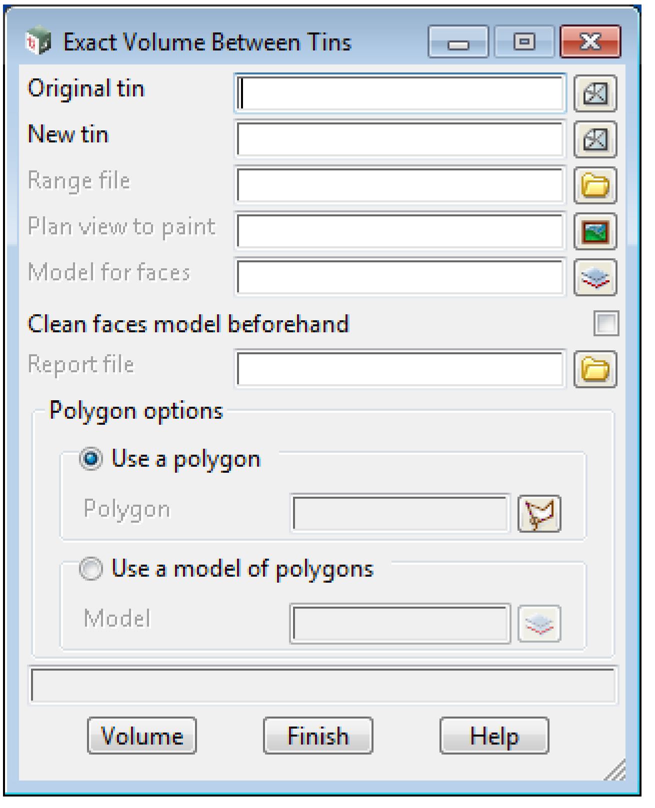
IMPORTANT NOTE - the ***volume totals*** are **only** calculated for the ***depths in the range file***. If the depth ranges ***does not cover*** the entire depth difference between the tins then the totals which are the sum of the different depth ranges will ***not*** be the same as the volumes between the two tins.





For more information on the exact volume calculations, go to the section 23.2.2 Theory of Exact Volumes

On selecting the **Tin to tin** option, the **Exact Volume Between Tins** panel is displayed.



Cut is defined to be where ever the new tin is **below** the original tin. Fill is defined to be where ever the new tin is **above** the original tin.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Original/New Tin	tin box		available tins

name of the original/new tin for determining volumes.

Range file	input		*.drf

if non-blank, the user supplied range file is used to split up the volumes report and define the depth colours used for painting a view.

Plan view to paint	input		available views

if non-blank, the given plan view will be painted according to the depth colours given in the range file.

Model for faces	input		available models

if non-blank, faces will be created with colours according to the range colours given in the range file.

Clean faces model beforehand	tick box		

if ticked, the model of faces is cleaned out before the option runs.

Report file	input		*.rpt

name of the file to contain the volume report. If the file already exists, the report will be appended to the file. If no name is given, no report is produced.

Polygon options

Use a polygon	radio button		
Polygon	poly string select		

if Use a polygon *is selected, then the selected string is used as the bounding polygon for the volume calculations.*

Use a model of polygons radio button

Model model box available models

if Use a model of polygons *is selected, then this model is used and each string in the model is used as a bounding polygon and the volume calculated.*

Volume button

The volume between the two tins for the given range file within the selected bounding polygon is calculated by the exact method. If a range file does not exist, the volumes between the two tins is calculated.

<Esc> can be used to terminate the option during volume calculations.

Continue to the next section 23.4.8 Exact - Tin to Tin 2 or return to 23.4 Exact.

23.4.8 Exact - Tin to Tin 2

Position of option on menu: **Design =>Volumes =>Exact =>Tin to tin 2**

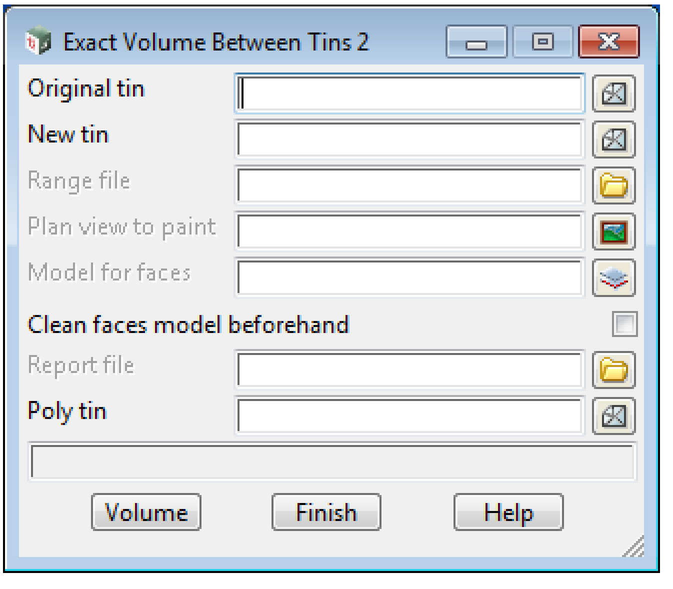
This panel is used to calculate the volume between two tins using the exact method but the calculations are restricted to only the regions defined by another tin rather than to a polygon.

The calculated volumes can be produced and reported on over user supplied depth ranges. Similarly, a view can be coloured on a depth basis using the same **range** file.

IMPORTANT NOTE - the ***volume totals*** are **only** calculated for the ***depths in the range file***. If the depth ranges ***does not cover*** the entire depth difference between the tins then the totals which are the sum of the different depth ranges will ***not*** be the same as the volumes between the two tins.

For more information on the exact volume calculations, go to the section 23.2.2 Theory of Exact Volumes

On selecting the **Tin to tin** option, the **Exact Volume Between Tins 2** panel is displayed.



Cut is defined to be where ever the new tin is **below** the original tin. Fill is defined to be where ever the new tin is **above** the original tin.

The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Original/New Tin tin box available tins

name of the original/new tin for determining volumes.

Range file input *.drf

if non-blank, the user supplied range file is used to split up the volumes report and define the depth colours used for painting a view.

Plan view to paint input available views

if non-blank, the given plan view will be painted according to the depth colours given in the range file.

Model for faces input available models

if non-blank, faces will be created with colours according to the range colours given in the range file.

Clean faces model beforehand tick box *if ticked, the model of faces is cleaned out before the option runs.*

Report file	file box	*.rpt

name of the file to contain the volume report. If the file already exists, the report will be appended to the file. If no name is given, no report is produced.

Poly tin	tin box	available tins

name of the tin to define the regions that the volume calculations are restricted to.

Volume	button	

The volume between the two tins for the given range file within the selected bounding polygon is calculated by the exact method. If a range file does not exist, the volumes between the two tins is calculated.

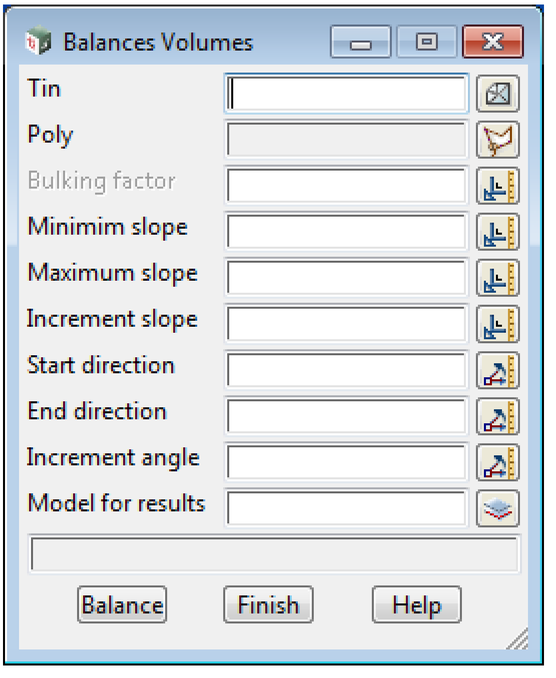
<Esc> can be used to terminate the option during volume calculations.

Continue to the next section 23.4.9 Exact - Balanced Volumes or return to 23.4 Exact.

23.4.9 Exact - Balanced Volumes

Position of option on menu: **Design =>Volumes =>Exact =>Balanced volumes**

On selecting the **Balanced volumes** option, the **Balances Volumes** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box		available tins
Poly	polygon box		
Bulking factor	measure box		available measures
Minimum slope	measure box		available measures
Maximum slope	measure box		available measures
Increment slope	measure box		available measures
Start direction	measure box		available measures
End direction	measure box		available measures
Increment angle	measure box		available measures
Model for results	model box		available models
Balance	button		

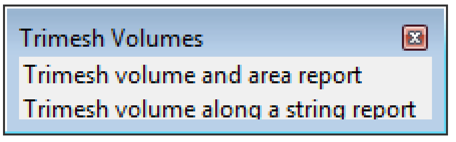
Return to 23.4 Exact.

23.5 Trimesh

Position of menu: **Design =>Volumes =>Trimesh**

These volumes are calculated for trimeshes.

The **Trimesh** walk-right menu is



volumes of trimeshes

volumes of trimeshes for chainge interval along a string

For the options see:

[23.5.1 Trimesh Volume and Area Report]

[23.5.2 Trimesh Volume Along a String]

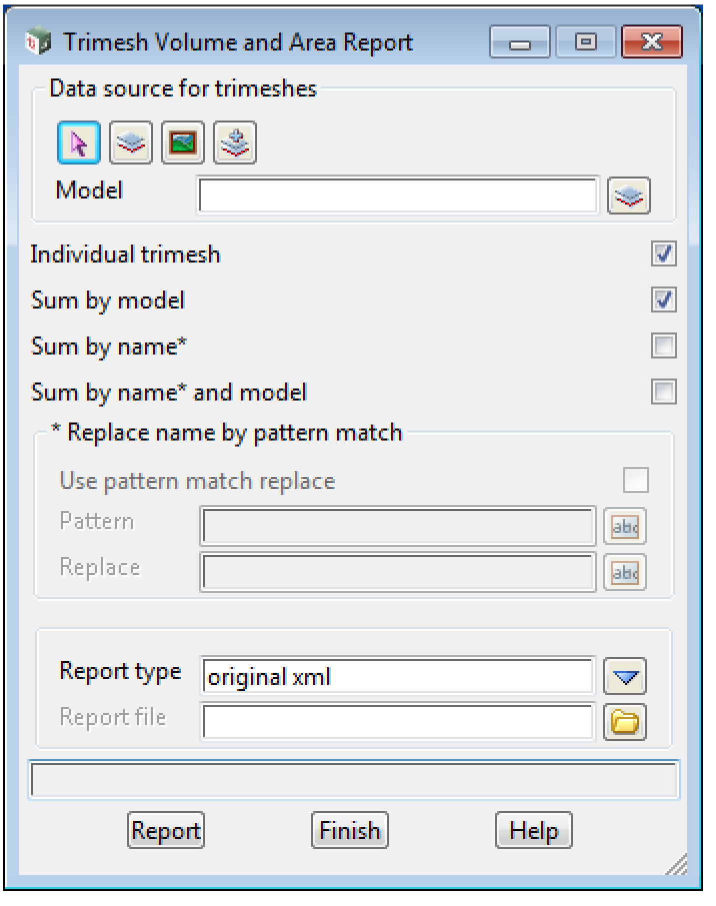
Or Return to [23 Volumes].

23.5.1 Trimesh Volume and Area Report

Position of option on menu: **Design =>Volumes =>Trimesh =>Trimesh volume report**

This option calculate the volume of closed trimeshes, and user selected combinations of closed trimeshes.

On selecting **Trimesh volume report**, the **Trimesh Volume Report** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

Data to search - for a full description go to 4.19.3 Data Source.

Data source	input		

source of trimeshes to calculate the volumes for.

Individual trimesh tick box

if ***ticked****, the volume for each individual trimesh will be written to the XML file.*

Sum by name tick box

if ***ticked****, the sum of the volumes for trimeshes with the same name will be written to the XML file.*

Sum by model tick box

if ***ticked****, the sum of the volumes for trimeshes in the same model will be written to the XML file.*

Sum by name and model tick box

if ***ticked****, the sum of the volumes for trimeshes in the same model will be written to the XML file.*

Report type choice box original xml, <Customize>

output format for the report.

For information on setting up custom reports from the generated XML file using xslts, see 4.30 Setting Up XML Reports.

Report file file box

an XML file will be created and a report of this name, and of the type given by **Report type** *will be generated from the XML file. If the file already exists, a panel with* **Replace** *and* **Cancel** *buttons will be displayed.*

Report button

the volumes of the various combinations of trimeshes are calculated and written to the XML file.

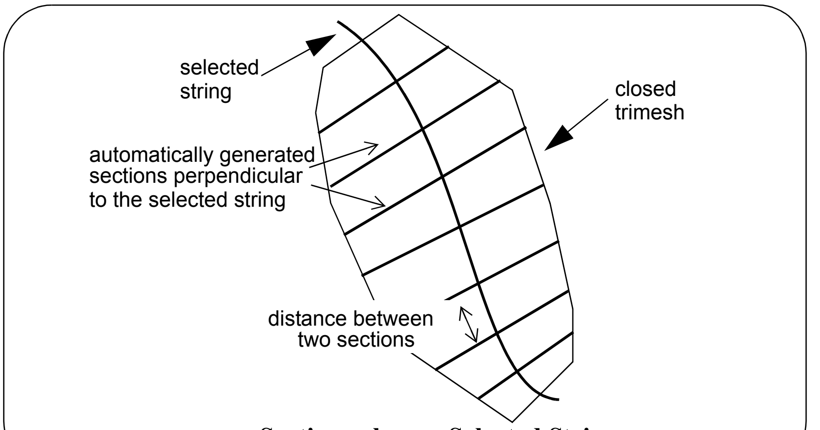
Continue to the next section 23.5.2 Trimesh Volume Along a String or return to 23.5 Trimesh.

23.5.2 Trimesh Volume Along a String

Position of option on menu: **Design =>Volumes =>Trimesh =>Trimesh volumes along a string**

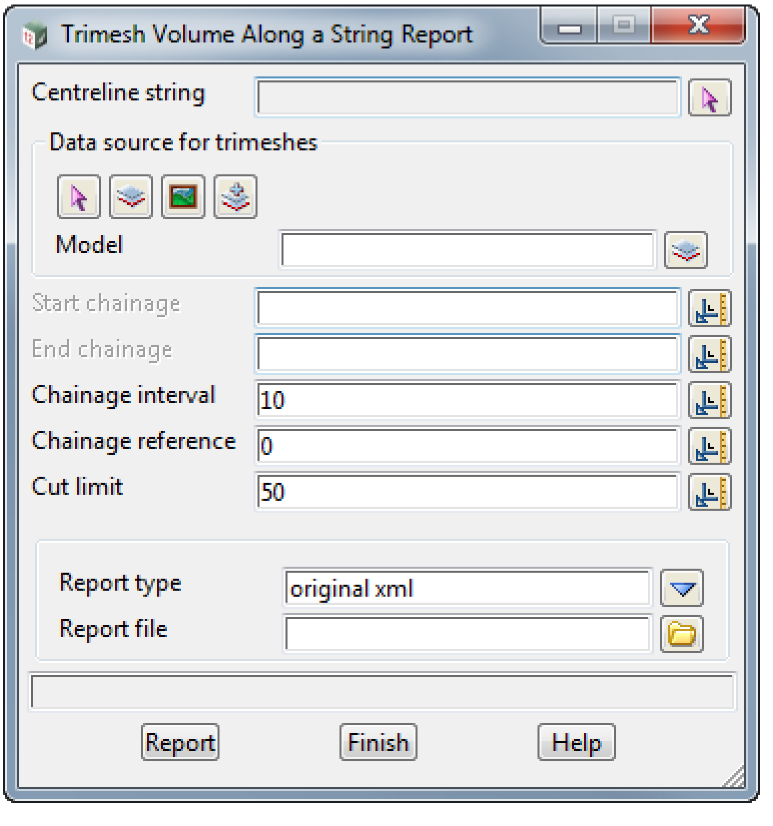
This option calculates the volume of closed trimeshes between chainages on a selected string.

Warning: The volumes can only be calculated for trimeshes that in plan, the section line only cuts the trimesh twice. So in plan, the trimesh can't be a U or an S shape.



Sections along a Selected String

Selecting **Trimesh volumes along a string report** displays the **Trimesh Volumes Along a String Report** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Centreline string	string select		

the string for defining the sections to be used in the volume calculations. The sections are taken at right angles to this string.

Data source type Model

Data to search for trimeshes - for a full description go to 4.19.3 Data Source.

Data source input

source of trimeshes to calculate the volumes along a string for.

Start chainage input

*if **not blank**, the chainage of the first section to use for volume calculations.*
*If **blank**, start at the beginnng of the selected string.*

End chainage input

*if **not blank**, the chainage of the last section to use for volume calculations.*
*If **blank**, end at the end of the selected string.*

Chainage interval real box 10

the chainage distance between the lines to section along.

Cut limit real box 50

the plan distance that the section goes out from the selected string.

Report type choice box original xml, <Customize>

output format for the report.

For information on setting up custom reports from the generated XML file using xslts, see 4.30 Setting Up XML Reports.

Report file file box

an XML file will be created and a report of this name, and of the type given by **Report type** *will be generated from the XML file. If the file already exists, a panel with* **Replace** *and* **Cancel** *buttons will be displayed.*

Report button

the volume of each trimesh is calculated between each pair of sections and written to the XML file.

Continue to the next section 23.6 Grid Cell or return to 23.5 Trimesh.

23.6 Grid Cell

Position of option on menu: **Design =>Volumes =>Grid cell**

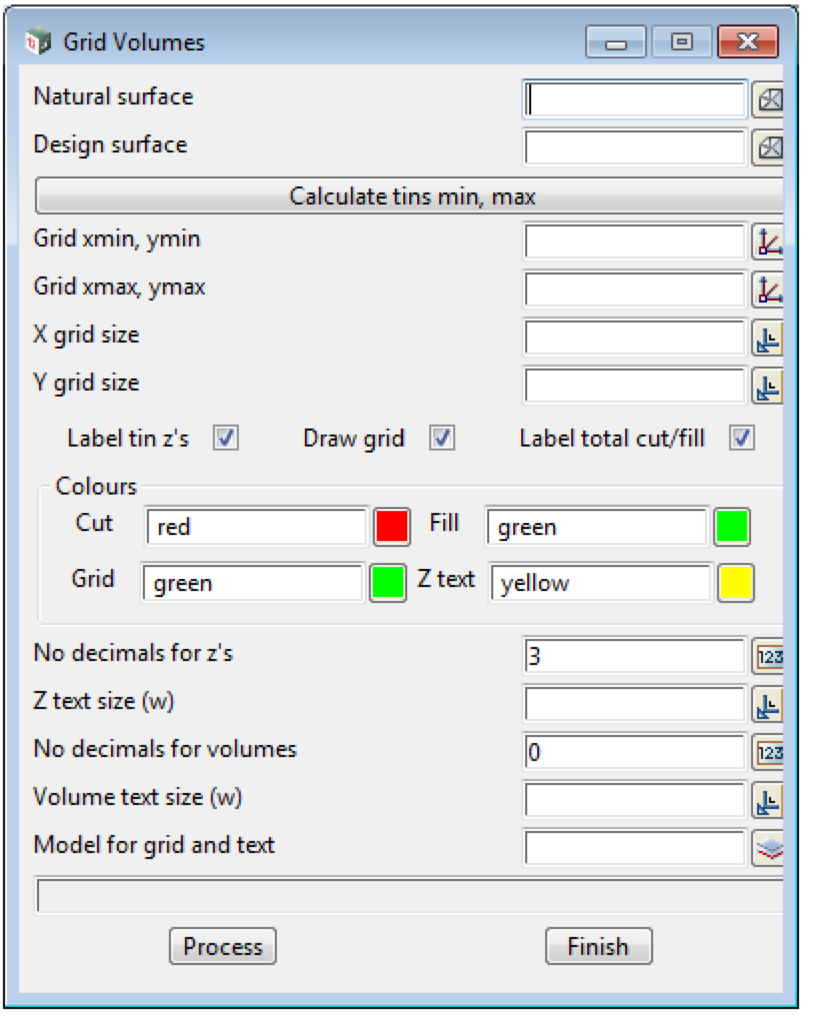
This panel is used to calculate the cut and fill volume between two tins using the exact method and then creating text for each rectangular cells of user defined sides covering the tins.

The rectangular cells (grid) can be drawn and the text of the cut and fill values for each grid cell and z-values at the cell corners. The total cut and fill can also be created at the bottom of the grid.

Cut is defined to be where ever the *design* tin is **below** the *natural surface* tin. ***Fill*** is defined to be where ever the *design* tin is **above** the *natural surface* tin.

For more information on the exact volume calculations, go to the section 23.2.2 Theory of Exact Volumes

On selecting the **Grid cell** option, the **Grid Volumes** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Natural surface/Design surface	tin box		available tins

name of the original/new tin for determining volumes.

Calculate tins min, max button

when selected, the minimum and maximum x and y values to define the grid cells are calculated and written to the grid x and y minimum and maximum panel fields.

Grid xmin xmax input

the minimum and maximum x values to define the grid cells.The two values are separated by one or more spaces.

Grid ymin ymax input

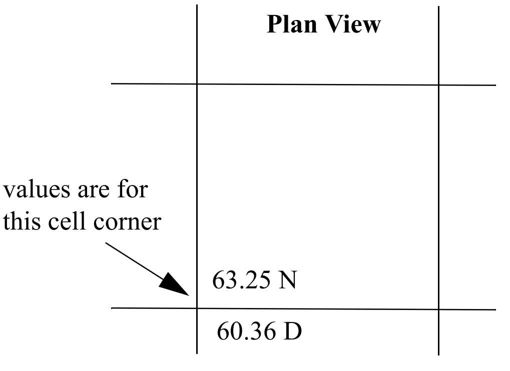
the minimum and maximum y values to define the grid cells.The two values are separated by one or more spaces.

X/Y grid size input

the X/Y size of the grid cell.

Label tin z's tick box ticked

*if ticked, create text values for the z-values from the tins for each of the grid cell corners. The natural surface z value is followed by a **N** and is placed above and to the right of the corner point. The design surface z value is followed by a **D** and is placed below and to the right of the corner point.*



Draw grid tick box ticked

if ticked, the grid is drawn.

Label total cut/fill tick box ticked

if ticked, label the total cut, fill and balance at the left hand bottom corner of the grid.

Cut/Fill/Grid/Z text colours colour box

colours to use for the cut, fill and z-values and the grid lines.

Num decimals for z's input 3

number of decimal places in the z values.

Z text size (w) input

size in world units for the text of the z values.

Num decimals for volumes input 0

number of decimal places in the cut and fill volumes.

Volume text size (w) input

size in world units for the text of the volumes.

Model for grid and text model box available models

model to place all the text and grid lines in.

Report file input *.rpt

name of the file to contain the volume report. If the file already exists, the report will be appended to the file. If no name is given, no report is produced.

Process button

calculate the grid volumes and grid lines.

Undo button

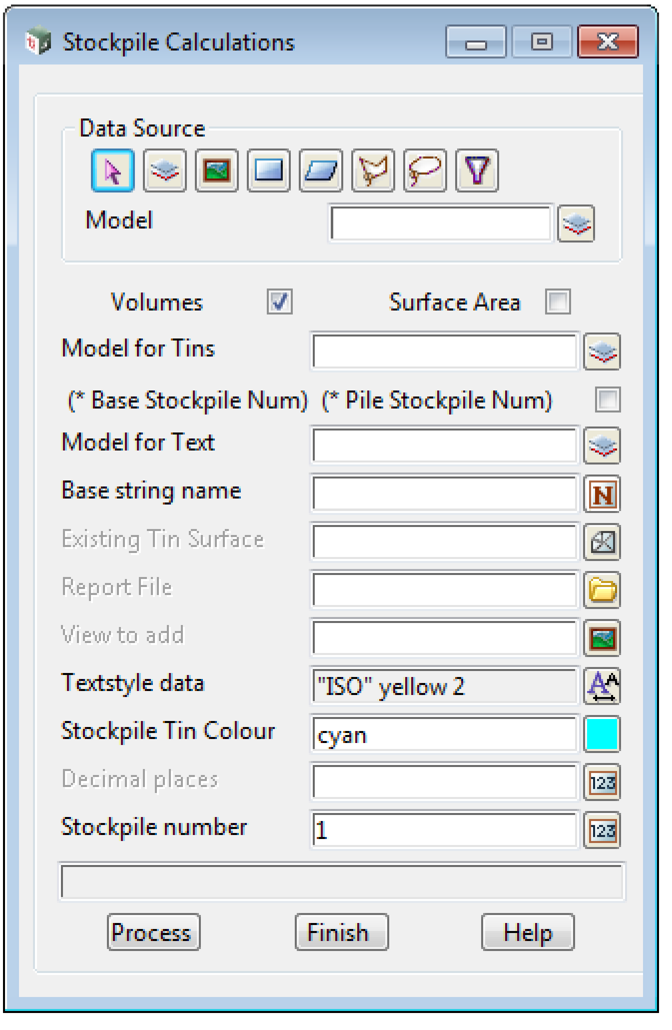
undo the last set of text and grid lines created whilst the panel was up.

Continue to the next section 23.7 Stockpile or return to 23 Volumes.

23.7 Stockpile

Position of option on menu: **Design =>Volumes =>Stockpile**

On selecting the **Stockpile** option, the **Stockpile Calculations** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Data source type Model

data selection type - for a full description go to 4.19.3 Data Source.

Data source

data source for strings to create the stockpile.

Volumes tick box ticked

if ticked, calculate and report the volumes of the stockpiles.

Surface area tick box

if ticked, calculate and report the surface area of the stockpiles

Model for tins model box available models *name of the model to place the stockpile tins in.*

Model for text	model box		available models

name of the model to place the text in.

Base string name	name box	bsp	available names

Name of the string to use as the base of the stockpile.

Report file	file box		*.rpt

name of the file to contain the volume report.

View to add	view box		available views

view to add the model of text to.

Textstyle data	textstyle data box		text favourites

text setup - for a full description go to 4.6.12 Textstyle Data and Textstyle Info.

Decimal places number box

number of decimal places for the volumes in the report.

Stockpile number number box

number of stockpile to use in the report.

Process button

calculate the volume and/or surface area of the stockpile.

Continue to the next section 23.8 Tin to Tin by Height Range or return to 23 Volumes.

23.8 Tin to Tin by Height Range

Position of option on menu: **Design =>Volumes =>Tin to tin ht range**

This option is used to calculate the cut and fill volume between two tins breaking the *cut* and *fill* volumes up by height ranges.

An approximate method is used which uses the z-value of the tin at a point as the representative of the height for a cell of user specified size "delta x" by "delta y" centred on the point. By making delta x and delta y small, the volumes become more accurate but the computation time increases.

Please note that the volume totals are **only** for the heights in the range file. If the height ranges don't cover the entire height difference, then the totals will not be the volumes between the two tins.

Cut is defined to be where ever the new tin is **below** the original tin. **Fill** is defined to be where ever the new tin is **above** the original tin.

In the volumes report, the sign for cut (negative or positive) is given by the cut volume sign from the **default settings** panel (fill will have the opposite sign).

The *height range file* consists of a list of height ranges and colours, one set per line, in the format

lower_height upper_height height_colour

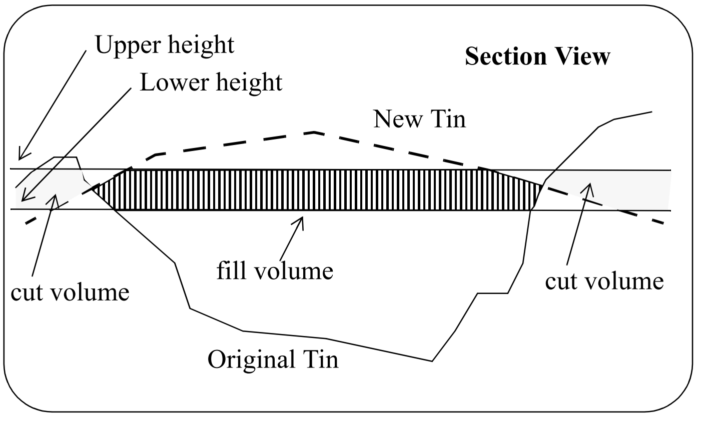
This line represents all heights satisfying

lower_height <= height < upper_height.

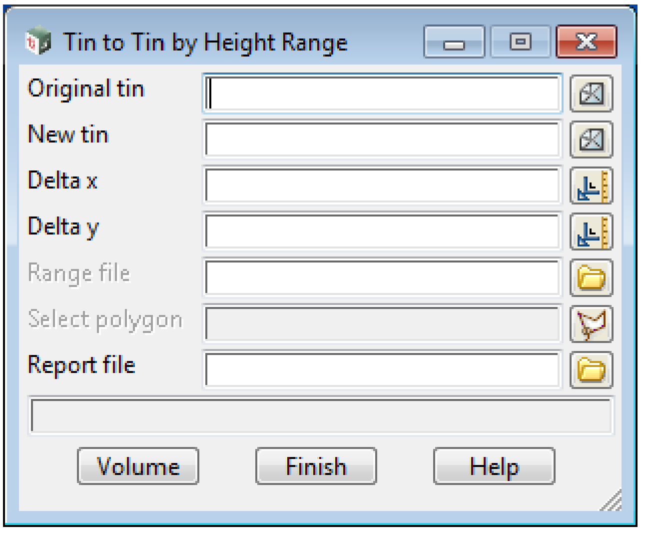
For each range in the file, the total cut and fill volumes for the height range will be reported on. The colour is ignored in this option.

Example of a Height Range File

```
//   height range file
// format:   lower_height      upper_height      colour_for_height_range

110    120    green      // colour green where the height is greater or equal to 110 and less than 120
120    220    "dark green"        // colour dark green where height is >= 120 and less than 220
```



On selecting the **tin to tin ht range** option, the **tin to tin by height range** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Original/New tin	tin box		available tins

name of the original/new tin for determining volumes.

Delta x/Delta y	input	1	

x/y size of the cell to consider to be of the height.

Select polygon	string select		

optional. If selected, the volumes are restricted to being inside this polygon.

Range file	input		*.drf

*if non-blank, the user supplied height range file is used to split up the volumes report. Note that the volume totals are **only** for the heights in the range file.*

Report file	report file		*.rpt

*if non-blank, the name of the file to contain the volume report. Note that the volume totals are **only** for the heights in the range file*

Volume	button		

calculate the volumes.

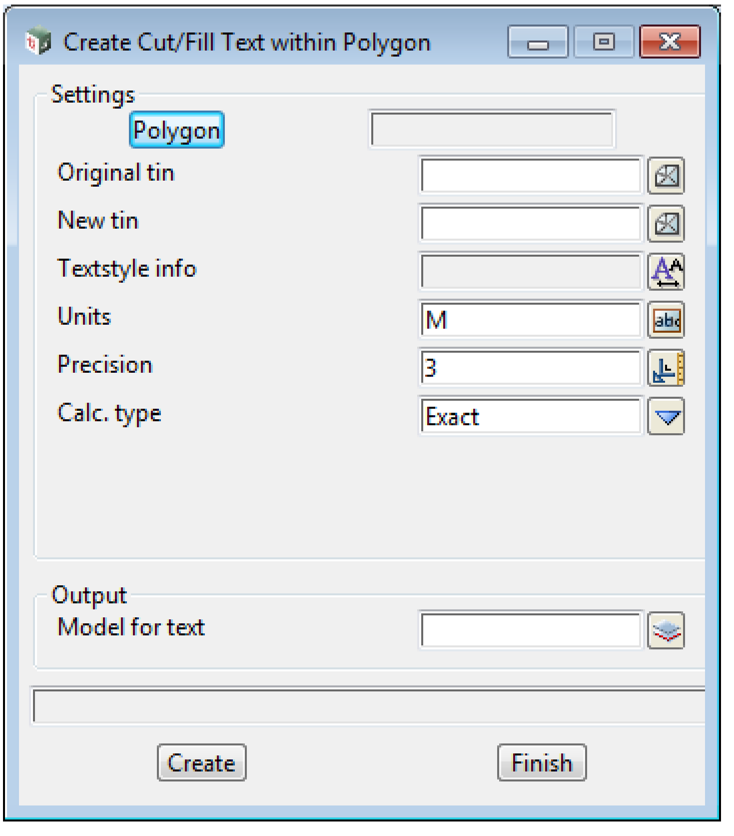
Continue to the next section 23.9 Create Cut/Fill Text Within Polygon or return to 23 Volumes.

23.9 Create Cut/Fill Text Within Polygon

Position of option on menu: **Design =>Volumes =>Cut/fill text in poly**

This option is used to calculate the cut and fill volume between two tins within a polygon and writing the cut and fill values to a model. The Exact or End Area method can be used for the volume calculations.

Selecting **Cut/fill text in poly** brings up the **Create Cut/Fill Text within Polygon** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Polygon	poly string select		
the volumes are restricted to being inside this polygon.			
Original/New tin	tin box		available tins
name of the original/new tin for determining volumes.			
Textstyle info	textstyle info box		
textstyle info to use for the cut and fill text			
Units	input		
text to place after the cut/fill values			
Precision	input	3	
number of decimal places to use for the cut/fill values			
Calc. type	choice box	Exact	Exact, End Area
use either Exact or End Area for the volume calculations			

Angle input

angle for sections when End Area is used

Separation input

distance between sections when End Area is used

Model for text model box available models

model to place to cut and fill text in.

Create button

calculate the volumes and text.

Return to 23 Volumes.

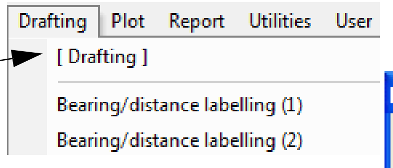
24 Drafting

The **Drafting** menu contains options to create data for plots.

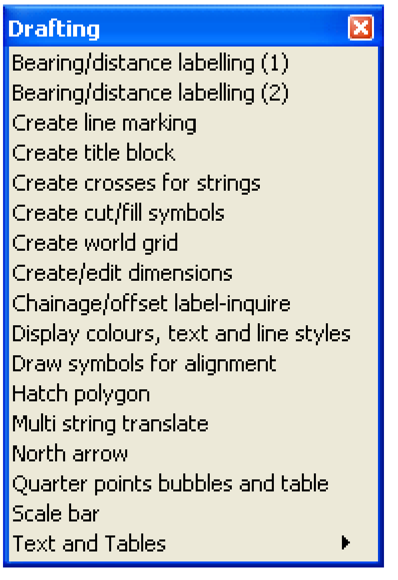
The **Drafting** menu is

on Main menu

create floating **Drafting** menu



on 12d Model menu and floating Drafting menu

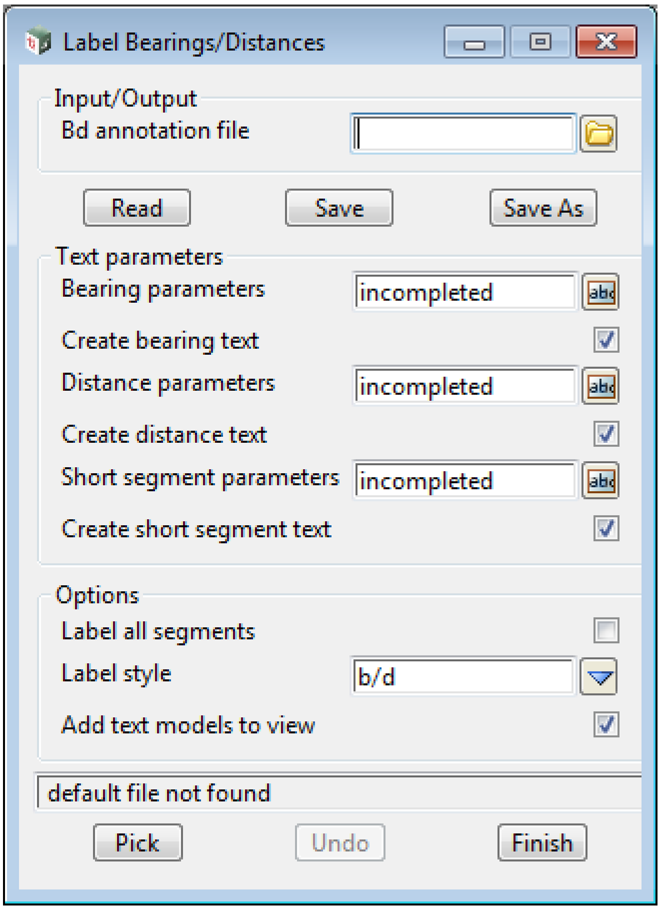


For *Bearing/distance labelling* go to	24.1 Bearing/Distance Labelling
Create line marking	12.13.35.5 Line Marking
Create title block	25.7.1 Create/Edit Title Block File
Create crosses for strings	24.4 Create Crosses at String Points
Create cut/fill symbols	24.5 Create Cut/Fill Symbols
Create world grid	24.6 Create World Grid
Create/edit dimensions	24.7 Create/Edit Dimensions
Chainage/offset label inquire	24.8 Chainage/Offset Label Inquire.
Display colours, text, linestyles	24.9 Display Colours, Textstyles and Linestyles.
Draw symbols for alignment	24.10 Draw Symbols for Alignment String
Hatch polygon	24.11 Polygon Hatching
Multi string translate	24.16 Text and Tables
North arrow	24.13 North Point Insertion
Quarter points bubbles and table	17.18.3 Setout Lip Line
Scale bar	24.15 Scalebar
Text and tables	24.16 Text and Tables

24.1 Bearing/Distance Labelling

Position of option on menu: **Drafting =>Bearing/distance labelling**

This option is used to label a selected line or arc segment with bearing, distance information and arc length information. If the line or arc length is below a given value, the label information is added to a short segment table.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
bd annotation file	file box		*.bdf files

file containing values for the text parameters.

Read	button		

read the file given in the "bd annotation file" field.

Save	button		

save the current text parameter settings to the file given in the "bd annotation file" field.

Save as	button		

save the current text parameter settings in a file.

Bearing parameters	input box	none/complete incomplete	bearing parameters

if none, don't label bearings.

If complete, all the bearing parameters have been set.

If incomplete, not all the bearing parameters have been set so the labels can't be created.

Create bearing text	tick box	tick	

if tick, bearings are labelled.

Distance parameters	input box	none/complete incomplete	bearing parameters

if none, don't label distances.

If complete, all the distance parameters have been set.

If incomplete, not all the distance parameters have been set so the labels can't be created.

Create distance text	tick box	tick	

if tick, distances are labelled.

Label all segments	tick box		

if tick, all segments of the selected element are labelled.

If not-tic, only the selected segment is labelled.

label style	input box	b/d, d/b, /bd /db, bd, db	b/d

defines how to label the bearing and distances. For example, d/b means distance above the line and bearing below the line. Bearing or distance is only labelled if the appropriate flags are set.

Add text models to view	tick box	tick	

if tick, all models containing text are added to the view containing the selected segment.

Minimum length	input box	0	

if non-zero, then if the length of the segment is less than this value, then it is given a short segment number and added to the short segment table.

Next short segment no.	input box	1	

the next number to use for short segments.

Pick	button		

select the segment to be labelled.

Undo	button		

undo the last set of created labels.

24.2 Line Marking

Position of option on menu: **Drafting =>Create line marking**

This option creates a title block file from a model of data. It has already been documented as

View => Visualisation => Line marking

in the section 12.13.35.5 Line Marking.

24.3 Create Titleblock

Position of option on menu: **Drafting =>Create title block**

This option creates a title block file from a model of data. It has already been documented as

Plot => Create title block file

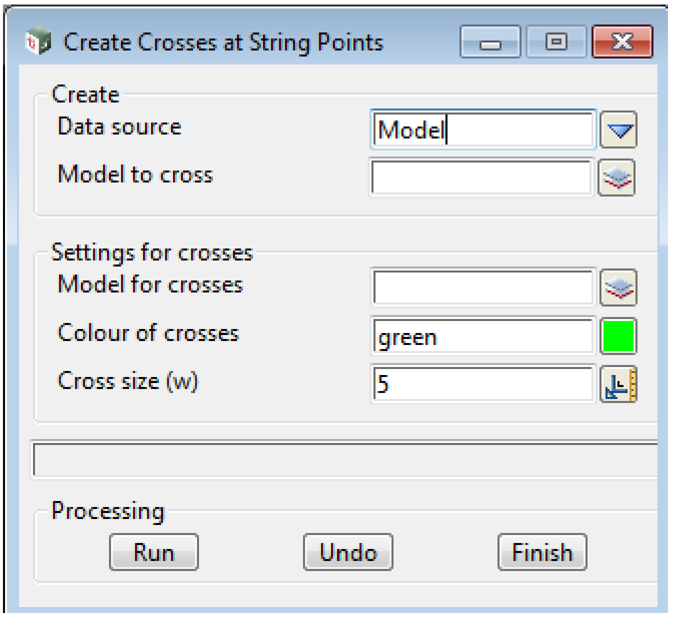
in the section 25.7.1 Create/Edit Title Block File.

24.4 Create Crosses at String Points

Position of option on menu: **Drafting =>Create crosses for strings**

This panel is used to create crosses at the points of strings. The cross is made up of two 2d strings with the intersection of the two strings being the (x,y) position of the point in the string.

The option is mainly used to created crosses to send to packages that don't support a cross at a point.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Data source	data source	Model	String, Model, View

type of data source.

Model/View/String	data source		

data source to create crosses for.

Model for crosses	model box		available models

model for the created crosses

Colour for crosses	colour box		available colours

colour for the crosses

Cross size (w)	input box		5

size of the cross in world units.

Run	button		

run the option

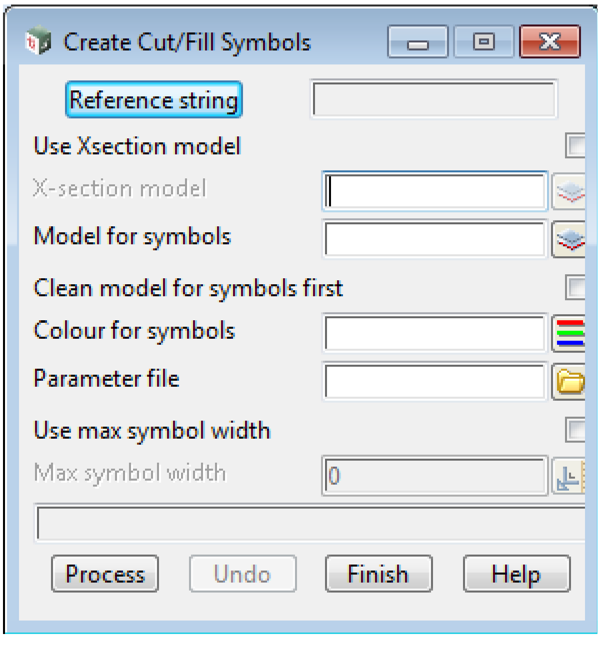
Undo	button		

undo the last set of crosses created whilst the panel has been up.

24.5 Create Cut/Fill Symbols

Position of option on menu: **Drafting =>Create cut/fill symbols**

This option is used to create cut and fill symbols (tadpoles) between strings or the names of the points across x-sections. The choice of symbols and the points they go between is controlled by a parameter file.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Reference string	string select		

string used to define chainage and perpendicular

Use X-section model	tick box	not ticked	

if ticked, the points on the cross-sections in the model are used as the names to create symbols between.

X-section model	model box		available models

model of cross sections used if "Use X-section model" is ticked.

Model for symbols	input		available models

model for the created cut-fill symbols

Clean model for symbols first	tick box	not ticked	

if ticked, the Model for symbols *will be cleaned before the new symbols are created and added.*

Colour for symbols	input		available colours

colour for the created cut-fill symbols

Parameter file	input		

file specifying what symbols are drawn between what strings (or points on x-sections).

Use max symbol width	tick box		

if tick, the symbols are only stretched to the maximum size given in the "Max symbol width" field.

Max symbol width	input		

maximum distance in world units to stretch the symbol.

Process button

Run the option.

Undo button

undo the last set of cut-fill symbols created by running the panel.The "Undo" function is lost once the option is finished.

Notes:

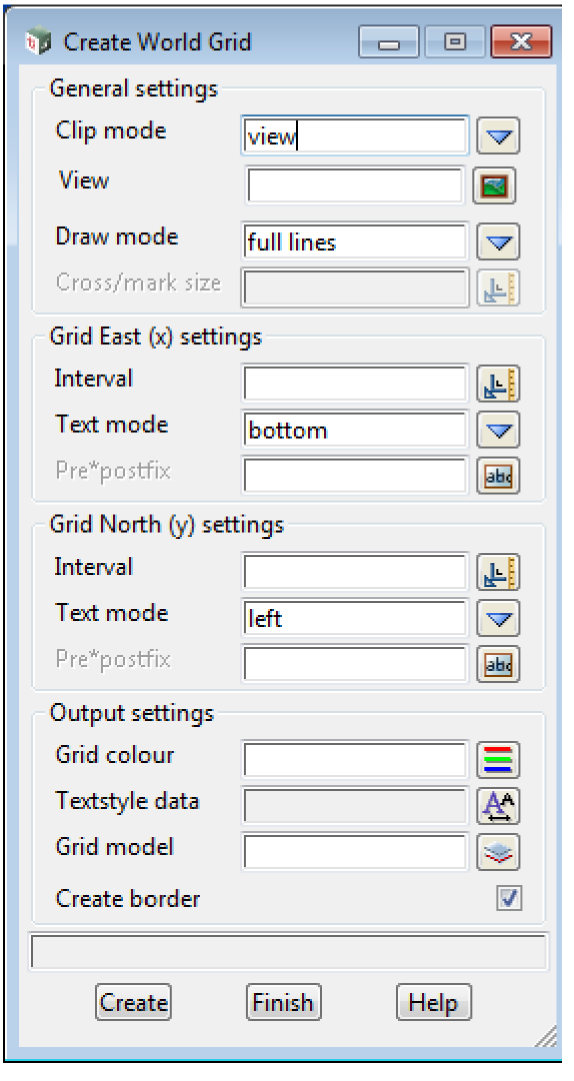
```
// Sample file for plotting symbols to denote cut/fill slopes
// All lines starting with // are comments. Blank lines are ignored
//
// The file layout is:

// ch_start ch_end ch_inc side str1   str2 symbol1 %1 symbol2   %2
//
// eg 110 150 10 L "trial1->c1" "trial->c2" tadpole 100 tadpole 50
//
// ch_start defines the start chainage
// ch_end defines the end chainage
// ch_inc defines the chainage increment
// side is one of L, R or LR (left, right, both)
// str1 is the name of the string (or point on x-section) to start tadpole
// str2 is the name of the string (or point on x-section) to end tadpole

// If strings are used, the string model name must be contained in the
// string definition i.e. in the format "model->string_name"
//
// The symbols drawn between the strings (or points) alternate between
// symbol1 and symbol2 and the symbols are drawn as the
// given percentages %1 and %2 respective of the distance
//  between the strings (points).

// The symbol can be one of "
// tadpole      - predefined tadpole shape
// line        - predefined batter tick
// "model=xxxxx" - where model "xxxxx" contains 3d strings
//             defining the symbol shape. This is as per
//             user-defined batter symbols in 4D.
//             NB. only 3d strings are supported in the macro

// Example

0 700 10 LR "road str->c2" "road str->c3" tadpole 100 tadpole 50
0 700 10 LR "road str->c4" "road str->c5" line 90 line 50
0 700 10 LR "road str->f2" "road str->f3" "model=symbol" 100 tadpole 50
0 700 10 LR "road str->f4" "road str->f5" line 90 line 50
```

24.6 Create World Grid

Position of option on menu: **Drafting =>Create world grid**

This option is used to create a grid, identical to the one drawn on a plan view, but in world units. This can then be output to CAD systems that do not allow grids to be easily created.

Selecting the **Create world grid** option, brings up the **Create World Grid** panel:



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Clip mode	choice box		rectangle, view

*if **rectangle**, icons appear for drawing a rectangle, a rotated rectangle or selecting a plot frame.*

*selecting the **rectangle** and **rotated rectangle** icons require the user to draw the rectangle to define the area to produce the grid for.*

*selecting the **plot frame** icon requires the user to select a plot frame and the view area from the plot frame defines the area to produce the grid for.*

*If **view**, a View Box appears and a view is selected to define the area to produce the grid for.*

Draw mode	choice box	full lines	full lines, crosses, marks

the type of the grid being drawn - full lines, crosses or marks.
*If **full lines**, then solid lines are drawn at the grid x and y spacing.*
*If **crosses**, then crosses of size **cross/mark size** are drawn at the intersection of the grid x and y spacing.*
*If **marks**, then solid lines size **cross/mark size** are drawn at the beginning/end of the grid at the x and y spacing.*

Cross/mark size	input

size in world units of the grid crosses/marks.

Grid East (x)

Interval	input

the distance between the East (x) grid lines. If this value is zero, the x grid lines will not be drawn.

Text mode	input	bottom	off, top, bottom, top and bottom

the x (East) grid lines can be labelled with their x value. If the grid lines are labelled, the labels can be drawn on the top, the bottom or both ends of the x-grid lines.

*if **off**, no text is created for the x-grid lines.*

*If **bottom**, text is created only at the bottom of the x-grid lines.*

*If **top**, text is created only at the top of the x-grid lines.*

*If **top & bottom**, text is created at the top and bottom of the x-grid lines.*

Pre*postfix	input

*if **non blank**, pre*post text to use for the labels on the x-grid lines.*
*prefix/postfix (pre*post) to be applied to the value of x (East). If pretext only, just give the text. If post text is required, precede it by a *. For example E*m will place E before the x value and m after the number.*

Grid North (y)

Interval	input

the distance between the North (y) grid lines. If this value is zero, the y grid lines will not be drawn.

Text mode	input	bottom	off, top, bottom, top and bottom

the y (North) grid lines can be labelled with their y value. If the grid lines are labelled, the labels can be drawn on the top, the bottom or both ends of the y-grid lines.

*if **off**, no text is created for the y-grid lines.*

*If **left**, text is created only at the left of the y-grid lines.*

*If **right**, text is created only at the right of the y-grid lines.*

*If **left & right**, text is created at the left and right of the y-grid lines.*

Pre*postfix	input

*if **non blank**, pre*post text to use for the labels on the y-grid lines.*
*prefix/postfix (pre*post) to be applied to the value of y (North). If pretext only, just give the text. If post text is required, precede it by a *.*

Output settings

Grid colour	colour box	available colours

colour for the grid lines.

Textstyle data	textstyle box	

textstyle data (height, angle etc) for the text for the grid labels.

Grid model	model box	available models

model for the grid lines and text.

Create border	tick box	tick

if tick, a rectangle around the area to create the grid for, is created.

Create	button	

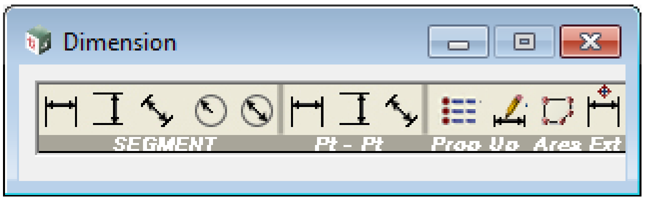
create the grid according to the values given in the panel.

24.7 Create/Edit Dimensions

Position of option on menu: **Drafting =>Create/edit dimensions**

This option is used to create and edit dimensions.

This option is currently under development.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up

read in an existing range file.

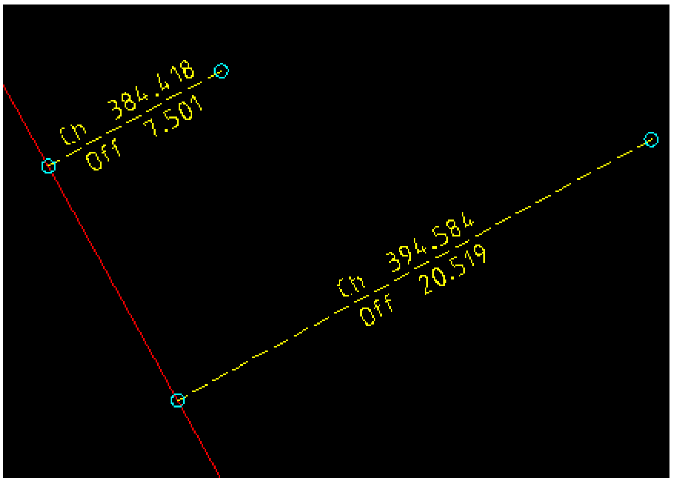
Write button

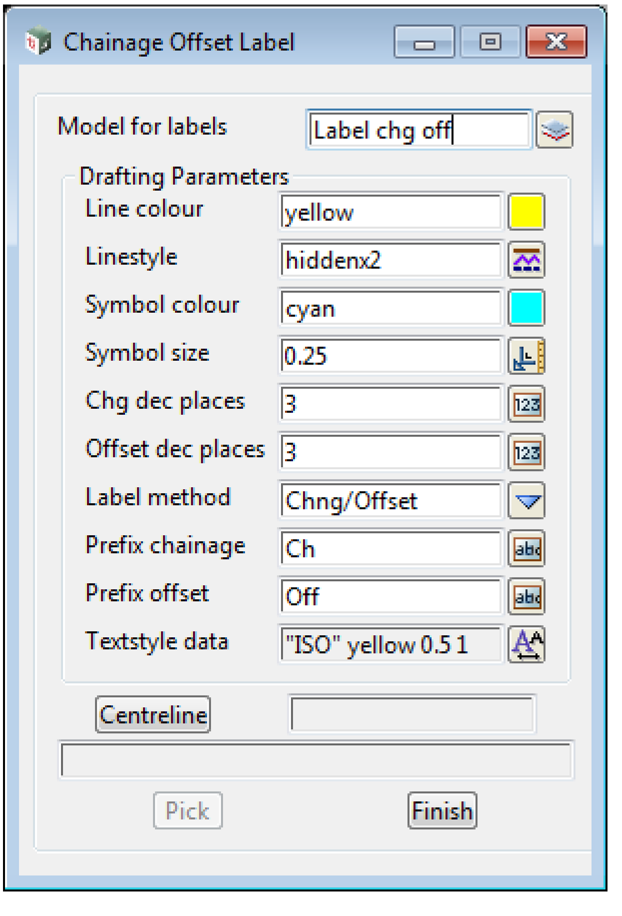
write out the data in the pages to the given range file.

24.8 Chainage/Offset Label Inquire

Position of option on menu: **Drafting =>Chainage/offset label inquire**

This option is used to calculate and label the perpendicular distance from a selected point to a selected centreline, and the chainage of the point dropped perpendicularly onto the centreline.





The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Model for labels model box Label chg off available models

if non-blank, create text for the calculated offset and chainage values and place it in this model.

Line colour yellow available colours

colour of the line drawn.

Linestyle linestyle box hidden2 available linestyles

text style for the text.

Symbol colour cyan available colours

colour of the symbol drawn at the point.

text style for the text.

Symbol size 0.5

size (in world units) of the symbol drawn at the point.

Label method choice box Chng/Offset Chng/Offset, Offset/Chng

method of labelling the line from a selected point to the dropped point.

Prefix chainage Ch

prefix for the chainage value.

Prefix offset Off

prefix for the offset value.

Textstyle data textstyle data box available text data

textstyle, size, colour etc. for the text.

Centreline string select

select the string to use for the calculation of chainage and offset.

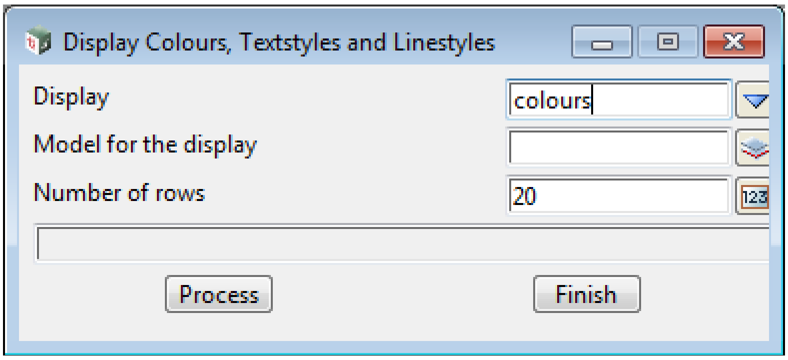
Pick button

pick a position and the offset from the position to the position dropped perpendicularly onto the selected Centreline *is calculated, plus the chainage on the Centreline of the dropped point. The values are reported in the panel's message area and if* Model for labels *is non-blank, text of the values is created and place in the model.*

24.9 Display Colours, Textstyles and Linestyles

Position of option on menu: **Drafting =>Display colours, text and line styles**

This option creates a model to display either all the colours, linestyles or textstyles defined for a project.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Display	choice	colours	colours, textstyles, linestyles

type of data to display.

Model for the display	model box		available modes

model to create the display in.

Number of rows	input	20	

the display is created by first doing a column of the given "Number of rows" and then moves on to another column until all the data is drawn.

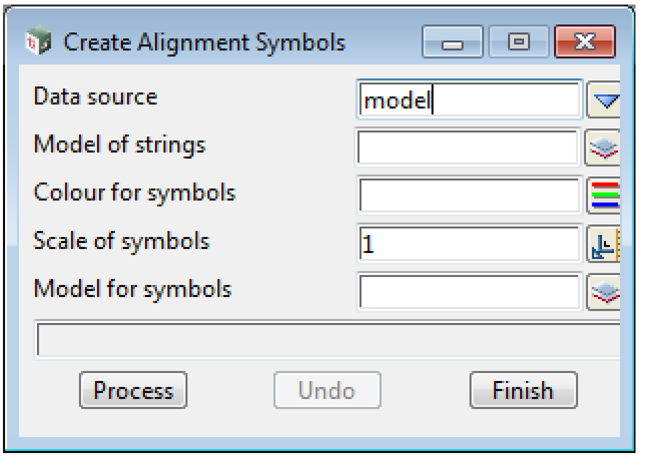
Process	button		

run the option.

24.10 Draw Symbols for Alignment String

Position of option on menu: **Drafting =>Draw symbols for alignment**

This panel is used to create symbols (butterflies and dumbbells) at the spiral-tangent, spiral-curve and curve tangent points.



The fields and buttons used in this panel have the following functions:

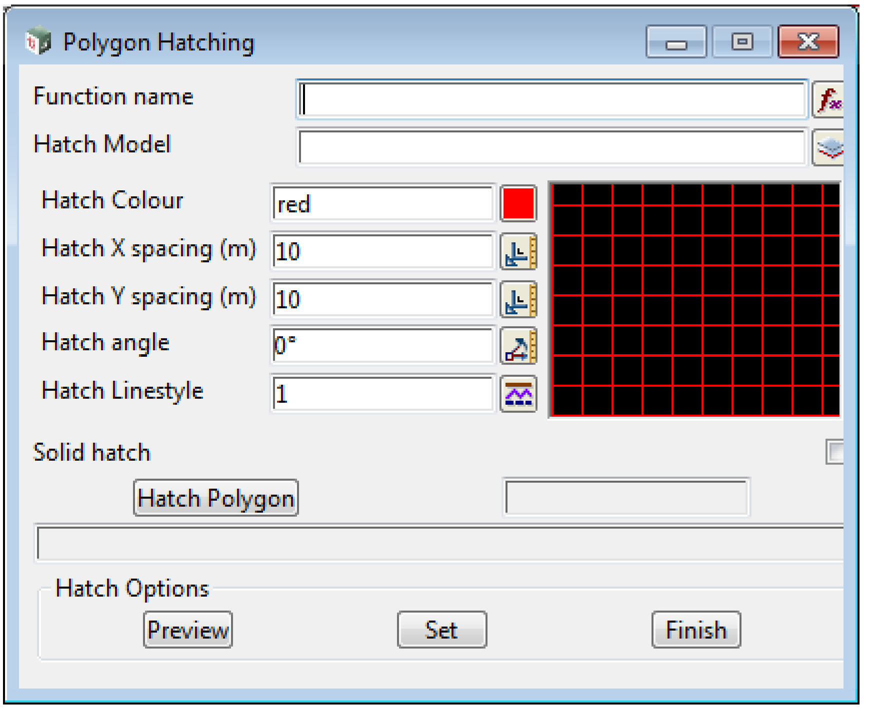
Field Description	Type	Defaults	Pop-Up
Data source	input box		
type of data to create symbols for.			
Model/View/String	input box		
data source to create symbols for.			
Colour of symbols	input box		
colour to use for the created symbols.			
Scale of symbols	input box	1	
scale of the created symbols.			
Model for symbols	input box		
model to put the created symbols into.			
Process	button		
run the option.			
Undo	button		
undo the last set of symbols created whilst the panel has been up.			

24.11 Polygon Hatching

Position of option on menu: **Drafting =>Hatch polygon**

This option is used to create hatching within a user selected string. A function is created so that the hatching can be recalced if the string is modified.

Selecting **Hatch polygon** brings up the Polygon Hatching panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function name	input box		
name of the hatching function.model			
Hatch model	input box		
model to place the hatching in.			
Hatch colour	colour box	red	
colour for the hatch lines			
Hatch x/y spacing (m)	input box	10	
distance in world units between the x/y hatching lines.			
Hatch angle	input box	0	
angle of the hatch lines.			
Hatch linestyle	input box	1	
linestyle for the hatch lines.			
Solid hatch	tick box		
if ticked, a solid fill of the given hatch colour is used.			
Hatch polygon	string select		

select the string to hatch.

Preview button

draw the hatching to see if it is correct - the hatching function has not been defined.

Set button

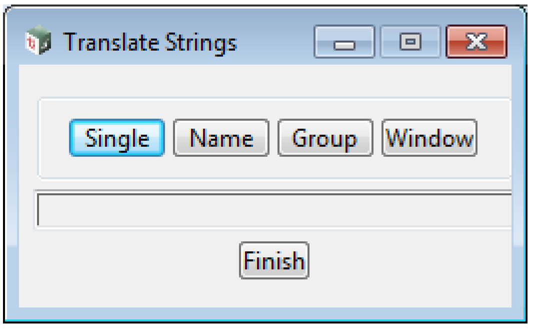
define the hatching function with the values given in the panel fields.

24.12 Translate Strings

Position of option on menu: **Drafting =>Multi string translate**

This panel is used to translate strings selected by name group of a window.

Selecting **Multi string translate** brings up the **Translate Strings** panel.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Single	button		

select a single string to translate.

Name	button		

select a string and all strings with the same name on the view will be translated.

Group	button		

select a string and all strings with the same group on the view will be translated.

Window	button		

click and release to define the first point of the window and then move the cursor and click and release to define the second point of the window. A position is then selected and as the cursor is moved, all strings in the view that are totally inside the window are translated and placed at the next selected position.

Undo	button		

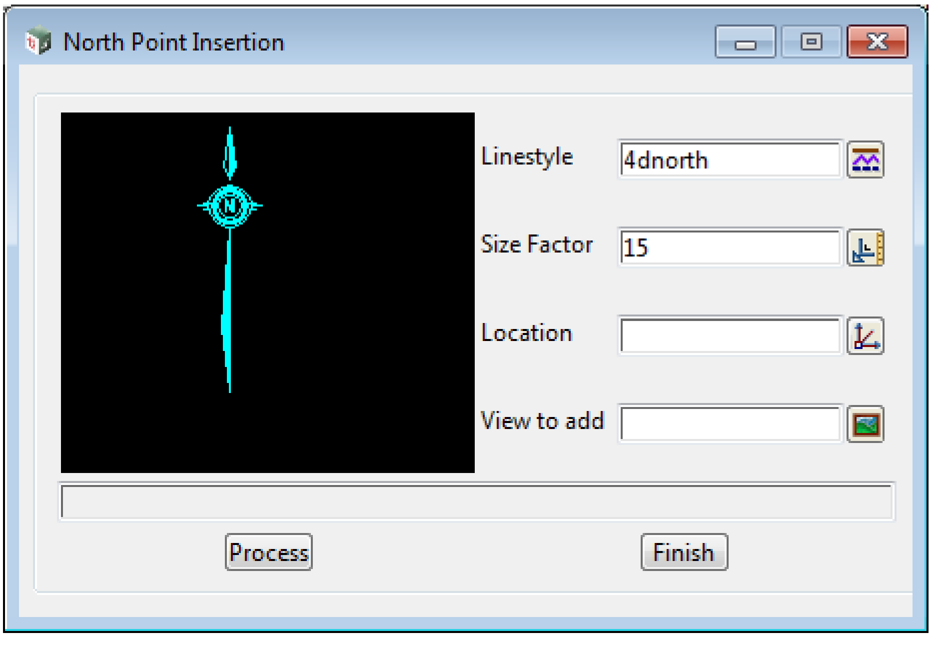
undo the last set of strings translated whilst the panel has been up.

24.13 North Point Insertion

Position of option on menu: **Drafting =>North arrow**

This option creates a point with a textstyle of the north point arrow.

Selecting **North arrow** brings up the North Point Insertion panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Linestyle	input box	4dnorth	

Linestyle for the north point arrow - leave as 4dnorth.

Size factor	input box	15	

factor the north arrow by this value.

Location	input box		

either type in an x y position or click on "+" and use Pick xyz to locate the north arrow.

View to add	input box		

view to add the arrow to.

Process	button		

*create the north point arrow in the model **North Point**.*

24.14 Quarter Points

Position of option on menu: **Drafting =>Quarter points bubbles and table**

This option is used to create bubbles and/or a report for the critical horizontal and vertical points and quarter points (by chord or by chainage) for any arcs in an alignment string.

This option has already been documented as

Survey=>Setout=>Setout lip line

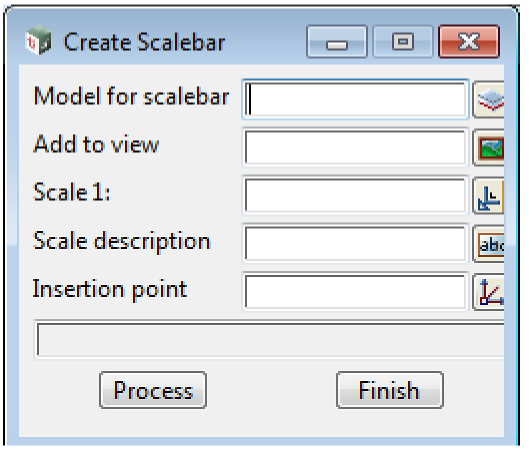
in the section 17.18.3 Setout Lip Line.

24.15 Scalebar

Position of option on menu: **Drafting =>Scale bar**

This option is used to create a scale bar.

Selecting **Scale bar** brings up the Create Scalebar panel.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Model for scalebar	model box		available models

model to place the scalebar in.

Add to view	input box		available views

if non-blank, the model is added to this view.

Scale 1: input box

scale to create the scale bar at.

Scale description input box

description to write under the scale bar.

Insertion point	pick ops		pick ops

position to place the left hand bottom corner of the scale bar at.

Process button

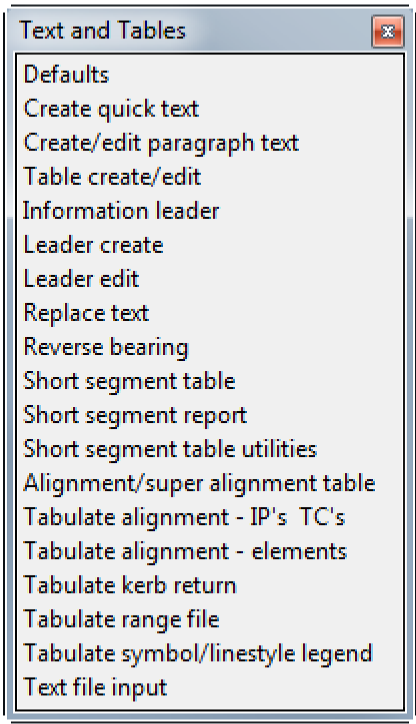
run the option.

24.16 Text and Tables

Position of menu: **Drafting =>Text and Tables**

The **Text and Tables** menu contains options to create text data and tables for plots.

The **Text and Tables** menu is

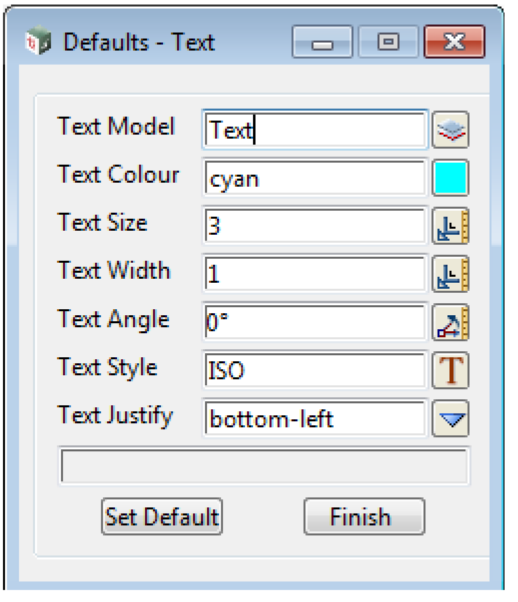


For the option *Defaults*, go to 24.16.1 Defaults - Text
Create quick text 24.16.2 Quick Text Input
Create/edit paragraph text 24.16.3 Text Creation/Edit
Table create/edit 24.16.4 Table Create/Edit
Information Leader 24.16.5 Leader Create (Information)
Leader create 24.16.6 Leader Text Creation
Leader edit 24.16.7 Leader Edit
Replace text 24.16.8 Replace Text
Reverse bearing 24.16.10 Reverse Bearing
Short segment table 24.16.11 Short Segments Table
Short segment report 24.16.12 Short Segment Report
Short segment table utilities 24.16.13 Short Segments Table Utilities
Alignment/super alignment table 24.16.14 Alignment & Super Alignment Table
Tabulate alignment - IP's, TC's 24.16.15 Tabulate Alignment - IP's and CT's
Tabulate alignment - elements 24.16.16 Tabulate Alignment - Elements
Tabulate kerb return 24.16.17 Tabulate Kerb Return
Tabulate range file 24.16.18 Tabulate Range File
Tabulate symbols/linestyle legend 24.16.19 Symbol / Linestyle Legend - Tabulation
Text file input same option as 24.16.3 Text Creation/Edit

24.16.1 Defaults - Text

Position of option on menu: **Drafting =>Text and Tables =>Defaults**

This option is used to project text defaults that are used with the "Fonts" button on many of the text options.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Text model	model box	Text	available models

default model to place the text in.

Text colour	colour box	cyan	available colours

default colour for text.

Text size	input box	3	

default size for text.

Text width	input box	1	

default x-factor text.

Text angle	input box	0	

default angle for text.

Text style	input box	iso	

default text style.

Text justify	input box	bottom-left	

default text justification.

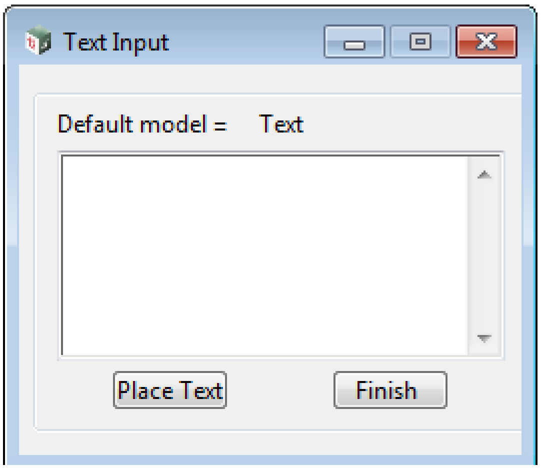
Set defaults	button		

set the project defaults.

24.16.2 Quick Text Input

Position of option on menu: **Drafting =>Text and Tables =>Create quick text**

The quick text option creates lines of text using the text defaults. The text can be edited and modified using the create/edit paragraph text.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Text box	button		

type the text to be created into the text box.

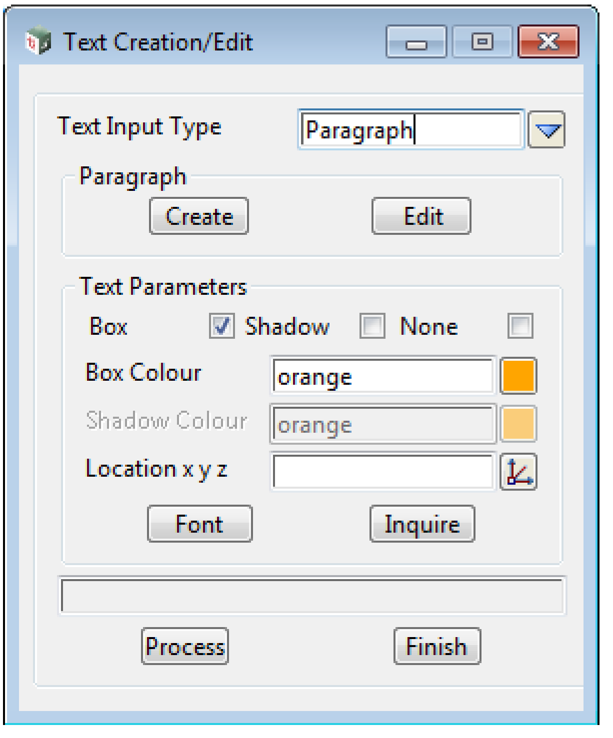
Place text	button		

pick a position to become the justification point for the first line of text.

24.16.3 Text Creation/Edit

Position of option on menu: **Drafting =>Text and Tables =>Create/edit paragraph text**

The option creates and edits a paragraph of text created with this option or the quick text option.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Text input type	choice box	paragraph	paragraph, file

method of creating text - type in a paragraph or read from a file

if *Paragraph* selected

Create	button		

select create to bring up the text box to type text into.

Edit	button		

select edit to pick and edit existing text.

if *File* selected

Text file	file box		

name of the file to read in.

Text parameters

Box/Shadow/None	radio button	box	

if box is ticked, draw a box around the text.

if shadow is ticked, draw shadowed box around the text.

if none is ticked, don't draw a box around the text.

Box colour	colour box	orange	

Colour of the box around the text

Shadow colour colour box orange

Colour of the shadow around the box around the text

Location x y z x,y,z box pick ops menu

pick the position for the text.

Font button

bring up the values for the text - e.g. model, colour, size etc.

Inquire button

Process button

create the text and place it at the given location.

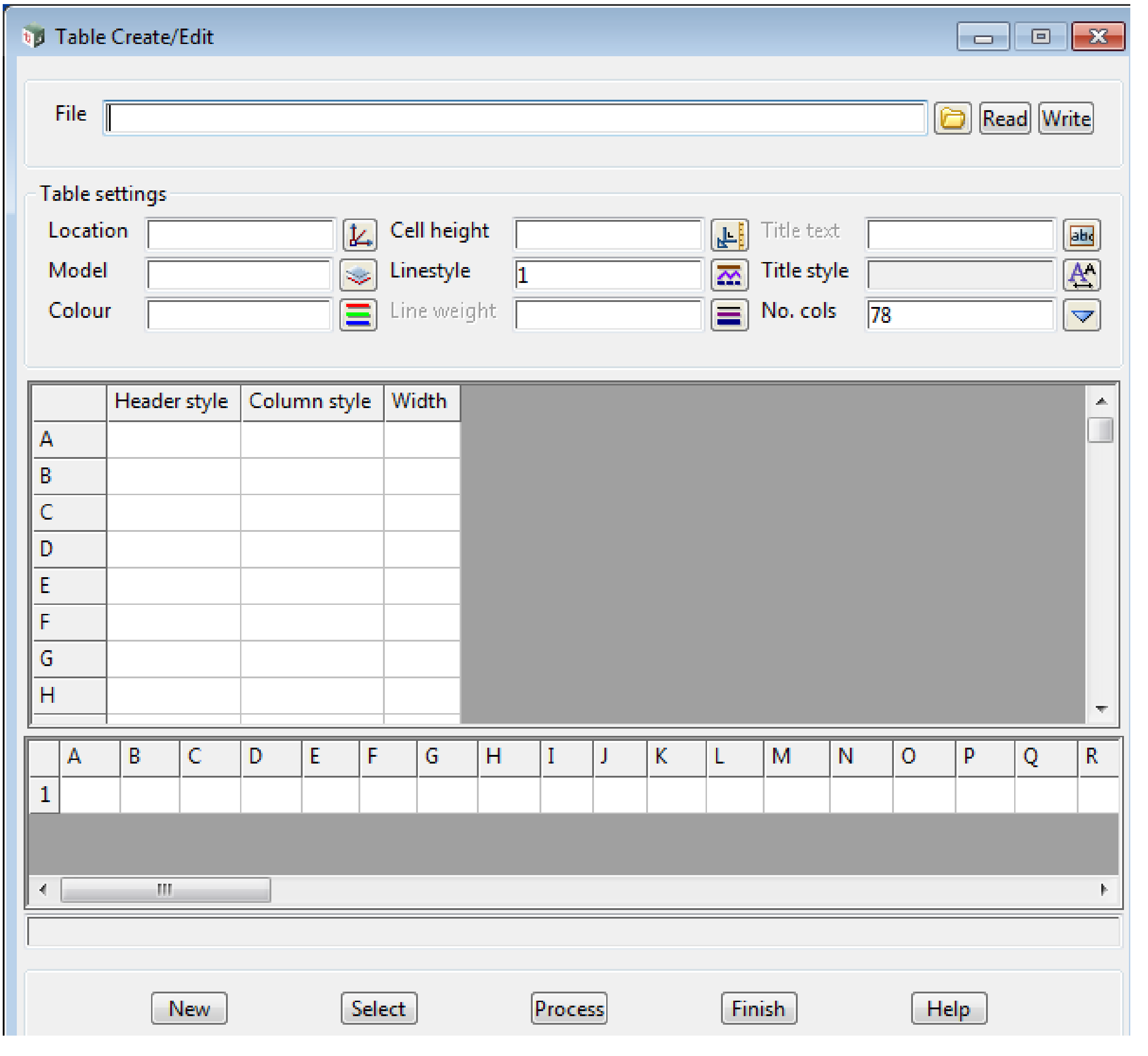
24.16.4 Table Create/Edit

Position of option on menu: **Drafting =>Text and Tables =>Table Create/edit**

This option is currently under development

This option reads in a csv file and loads the data into a grid on the panel. The data can be edited in the grid and then a table produced in a model.

An existing table in a model can also be selected and the data read into the grid on the panel. The data can then be edited and the table updated.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

File file

file to read in and display in the grid on the panel.

Read button

read the specified file in and display it in the grid on the panel.

Write button

write the data in the grid on the panel to the specified file.

Table settings

create the text and place it at the given location.

24.16.5 Leader Create (Information)

Position of option on menu: **Drafting =>Text and Tables =>Information leader**

This option creates a leader that can extract and list certain information about the string selected.

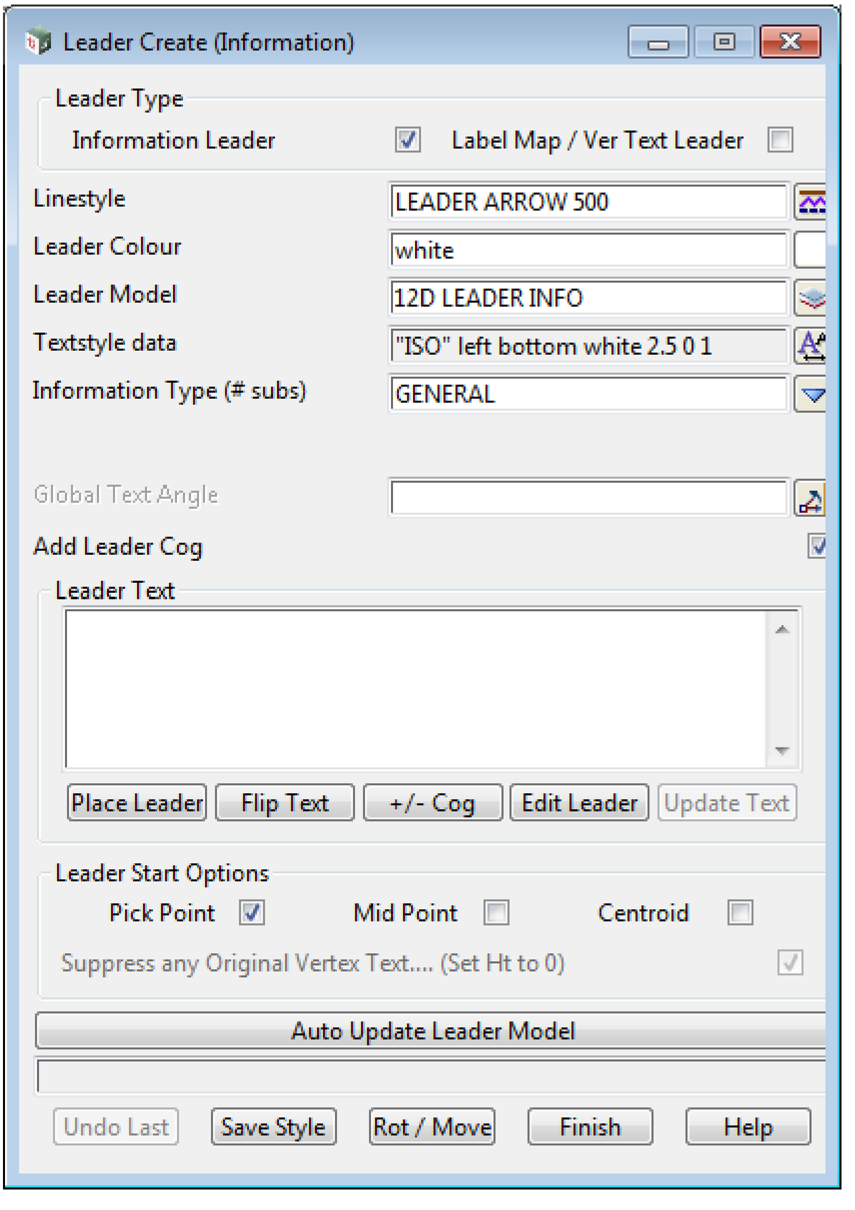
Information such as AREA, STRING LENGTH, SEGMENT LENGTH and GRADE can be displayed.

A choice list is available and can be partially user defined.

General notes can also be placed that do not extract information from any string selected.

Leaders can be moved along the string they relate to, and subsequent information values updated.

Selecting the **Information leader** option brings up the Leader Create (Information) panel:



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up

Leader Type

Information Leader tick box ticked

If ticked, information or general notes can be created.

Label Map / Ver Text Leader tick box not ticked

If ticked, any vertex text or text created via the Label Map File can be converted into a leader, allowing the text to be moved but still have the leader at the original location.

Linestyle linestyle box LEADER ARROW 500 available linestyles

If non-blank, entry used as leader arrow scale.

Leader Colour colour box white available colours

If non-blank, entry used as leader colour.

Leader Model model box 12D LEADER INFO available models

If non-blank, entry used as leader model.

Textstyle data input "ISO" white 2.5 available Textdata

If non-blank, entry used as text properties.

Information Type (# subs) choice box GENERAL available choices

If non-blank, entry used as type of information leader. The list is generated from a defaults file in the library.

Global Text Angle real GENERAL

If non-blank, entry used as angle for text during placement of leader.

Add Leader Cog tick box ticked

If ticked, leader cog will be added.

Leader Text

From the choice above, default text is placed in the Leader Text box on the panel.

Any calculated information value, such as AREA, LENGTH will be substituted for the # that may appear in the Leader Text.

Place Leader button

Select a position along a string that suits the type of information selected from the choice box (segment for SEGMENT LENGTH, a vertex for XYZ COORDS, etc.).

General notes (i.e. notes without the # in the Leader Text) may be placed using the cursor position as well.

Placement of the leader is controlled by the ***Leader Start Options****.*

Flip Text button

Select an information leader already placed. Upon accepting the string, the text will be flipped to the opposite side of the leader.

+/- Cog button

Select an information leader already placed. Upon accepting the string, the cog will be added or removed.

Edit Leader button

Select an information leader already placed to edit the text. The text will be displayed in the edit box.

Update Text button

If selected, the text will be updated on the selected leader.

Leader Start Options

Pick Point tick box

If ticked, the start of the leader will be at the selection point.

Mid Point tick box

If ticked, the start of the leader will be at the mid point of the string selected or segment selected, depending on the Information Type.

Centroid tick box

If ticked, the start of the leader will be at the centroid of the string selected.

Suppress any Original Vertex Text (Set Ht to 0) tick box

*Available only for the leader option **Label Map / Ver Text Leader**, where the vertex text on the original string selected can be invisible by setting its height to zero.*

Auto Update Leader Model button

If selected and the string the leader is linked to has changed, and/or the leader is moved to another position on the string, information in the leader relating to the string will be updated.

Undo Last button

If selected, will undo the last leader placed.

Save Style button

Writes out a default file to the .project folder using the current Information type displayed (e.g. STRING_LENGTH) as part of the file name.

STRING_LENGTH_LEADER_TEXTSTYLE.def

Example data written to file:

Text_Style "ISO"

Text_Colour 7

Text_Whiteout_Colour 0

Text_Border_Colour 0

Text_Size 2.500

Text_Units paper

Text_Angle 0.0000

This file can be copied to a user_lib area for use in other projects.

Rot / Move button

This option is two-fold depending on where you select the leader.

If the leader is selected at the arrow or first vertex, then the Move command is activated.

*If moving the leader from, say, one radius segment to another radius segment, then the leader only needs to be placed close to the segment, as the **Auto Update Leader Model** will lock it onto the segment and in this case update the radius value.*

If the leader is selected anywhere else, then the Rotate command is activated.

The leader rotates about the arrow insertion point.

Example of Library Defaults file:

Info_Leader_Create_Panel.def

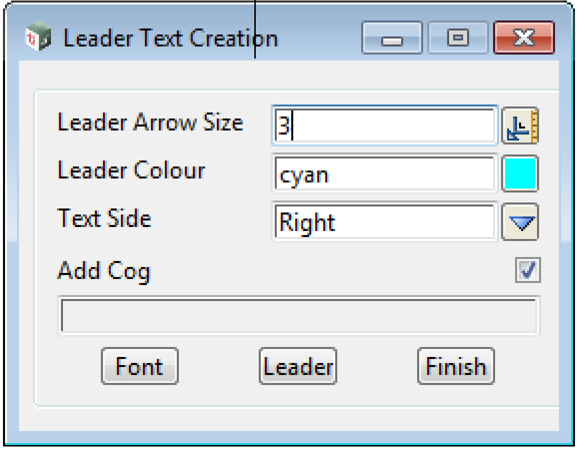
```
Leader_Style              "LEADER ARROW 500"
Leader_Colour              7
Leader_Model              "12D LEADER INFO"
Text_Style                ISO
Text_Colour               7
Text_Whiteout_Colour      -1
Text_Border_Colour        -1
Text_Units                "Paper"
Text_Size                 "2.500"
Text_Angle                "0"
AREA                      "Area #"              "2"
CENTROID_X_Y              "Centroid:\\n#"       "3"
CIRCLE_WITH_TEXT          "1"
GRADE_1_IN                "Grade: 1 in #"       "2"
GRADE_PERCENT             "Grade: #"            "2"
LEVEL                     "S.L. #"              "3"
SEGMENT_BEARING           "Bearing #"
SEGMENT_LENGTH_BEARING    "#"                   "2"
SEGMENT_LENGTH            "Seg Length #m"       "3"
SEGMENT_LENGTH_3D         "Seg Length 3d #m"    "3"
SEGMENT_RADIUS            "R #"                 "2"
STRING_LENGTH             "Length #m"           "3"
STRING_LENGTH_3D          "Length 3d #m"        "3"
STRING_NAME               "Control Line #"
TIN_LEVEL                 "S.L. #"              "3"
TIN_DEPTH                 "Depth #"             "3"
XY_COORDS                 "Coords:\\n#"         "3"
XYZ_COORDS                "Coords:\\n#"         "3"
USER_STRING_ATTRIBUTE     "#"
USER_VERTEX_ATTRIBUTE     "#"
USER_SEGMENT_ATTRIBUTE    "#"
USER_VERTEX_TEXT          ""
USER_SEGMENT_TEXT         ""
NOTE_PROP_MEDIAN          "Proposed Median"
NOTE_PROP_ROUNDABOUT      "Proposed Roundabout"
NOTE_PROP_ISLAND          "Proposed Traffic Island"
NOTE_REMAIN               "Existing Kerb\\nto remain"
NOTE_TAKE_UP              "Take up existing\\nkerb and channel"
NOTE_REMOVE               "Remove existing\\npipes"
NOTE_KERB_STD_SWG         "Refer to Std Dwg\\nfor Kerb Types"
NOTE_REMOVE_PP            "Remove existing power pole"
NOTE_PROP_CULVERT         "Proposed Culvert"
NOTE_MATCH_EXISTING       "Match into existing"
```

```
//STRING_ATTRIBUTES
Str_Att_1  <example only>

//VERTEX_ATTRIBUTES
Ver_Att_1  <example only>

//SEGMENT_ATTRIBUTES
Seg_Att_1  <example only>


Area syntax special:

AREA             "Area #"                             "2"

Drop down       Display Text which can be set as "Area *ha" to define
hectares        Decimal places

TIN_DEPTH             "Depth #"                       "2"

Drop down        Display Text which can be set as "Depth *mm" to
define millimetres    Decimal places


Attributes syntax special:

USER_STRING_ATTRIBUTE       "#"
USER_VERTEX_ATTRIBUTE   "#"
USER_SEGMENT_ATTRIBUTE  "#"

Drop down        Display Text of "#" that will be substituted with
attr value

Note: An attribute with the name "Leader" can be set on any string,
for any of the above attributes.

It will appear automatically in the choice box along with any others
set here
```

24.16.6 Leader Text Creation

Position of option on menu: **Drafting =>Text and Tables =>Leader create**

This option is used to create leader lines and text.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Leader arrow size	input box	3	

length (in world units) of the head of the arrow of the leader line.

Leader colour	colour box	cyan	

colour of the leader line or arrow

Text side	choice box	Right	Right, Left

side of the leader line to create the text.

Add cog	tick box	tick	

if tick, add a cog to the leader line.

Font	button		

brings up panel for defining information about the text (size, model colour etc.).

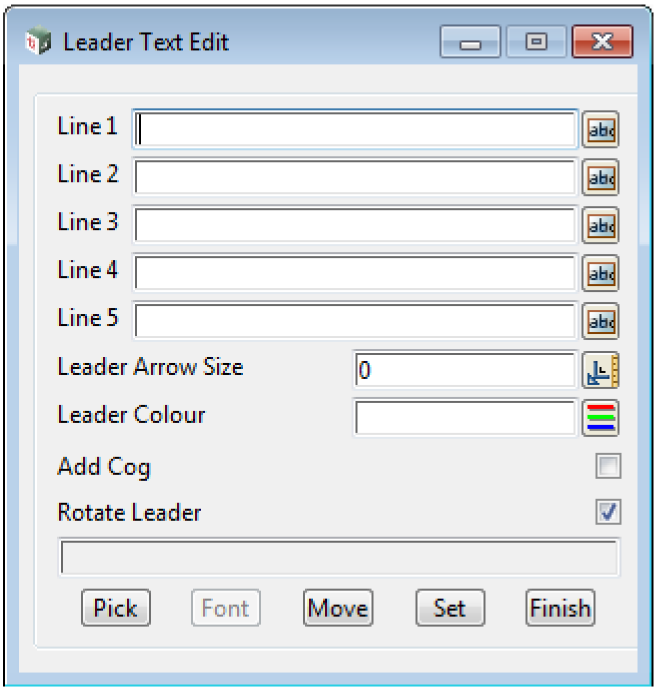
Leader	button		

create the leader by firs picking the point for the arrow end and then the point for the opposite end.where the text goes. Once the second point is place, the "Leader Text Input" panel is displayed for entering five lines of text.

24.16.7 Leader Edit

Position of option on menu: **Drafting =>Text and Tables =>Leader edit**

This option is used to edit leader lines and text.



The fields and buttons used in this panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Line 1-5 input box

lines of text from the selected leader.

Leader arrow size input/output box

length (in world units) of the head of the arrow of the leader line.

Leader colour colour box cyan

colour of the leader line or arrow

Text side choice box Right Right, Left

side of the leader line to create the text.

Add cog tick box tick

if tick, add a cog to the leader line.

Rotate leader tick box tick

if tick, the leader arrow remains fixed when the leader is moved.

If not tick, the whole leader line and text is moved.

Pick button

select the leader to be modified.

Font button

brings up panel for defining information about the text (size, model colour etc.).

Move button

move the select leader and its associated text. If "rotate leader" is ticked, then the leader arrow head is left fixed.

Set button

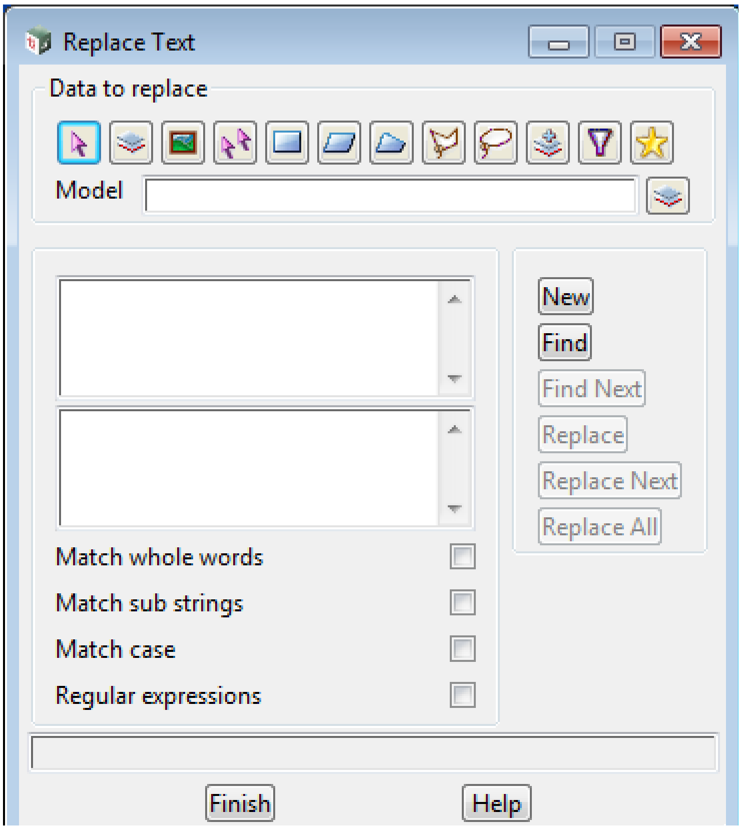
redefine the leader line and text with the current values in the panel fields.

24.16.8 Replace Text

Position of option on menu: **Drafting =>Text and Tables =>Replace text**

Position of option on menu: **Utilities =>H-Z =>Find/replace text**

This option is used to find and replace text in text strings, 4d string, super strings for vertex and segment text.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Data source type Model

source of data to be processed - for a full description go to 4.19.3 Data Source.

Data source

data source for text to search and change.

Find what input box

text to search for.

Replace with input box

text to replace the searched text with

Match whole words tick box

if not tick, text is selected if the "find text" is only part of the text.

If tick, only find the text when the text is exactly the same as the "find text", not just a part of the text.

Match case tick box

if not tick, case is ignored when trying to find a match.

If tick then case is not ignored when trying to find a match.

Regular expressions tick box

if tick then the Find what *text can contain regular expressions.*

If not tick, the Find what *text is not considered to contain regular expressions.*

New button

after selecting new the parameters for the find and replace can be changed.

Find button

find the search text. The text is highlighted and centred in the view.

Find Next button

find the next occurrence of the search text.

Replace button

replace the text.

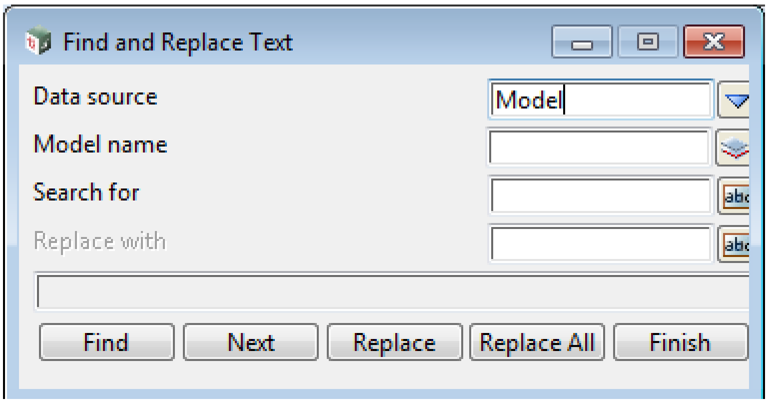
Replace all button

replace all occurrences of the text.

24.16.9 Find and Replace Text

Position of option on menu: **Drafting =>Text and Tables =>Replace text**

This option is used to find and replace text.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Data source type	input box		string, model, view

data source type.

Data source — input box

source of data to change text.

Search for — input box

text to search for.

Replace with — input box

if non-blank, text to replace the searched text with

Find — button

find the search text

Next — button

find the next occurrence of the search text.

Replace — button

replace the text.

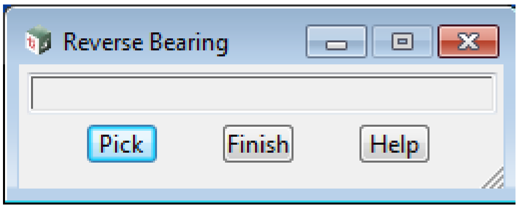
Replace all — button

replace all occurrences of the text.

24.16.10 Reverse Bearing

Position of option on menu: **Drafting =>Text and Tables =>Reverse bearing**

This section of documentation is a work in progress and will be updated in subsequent releases.

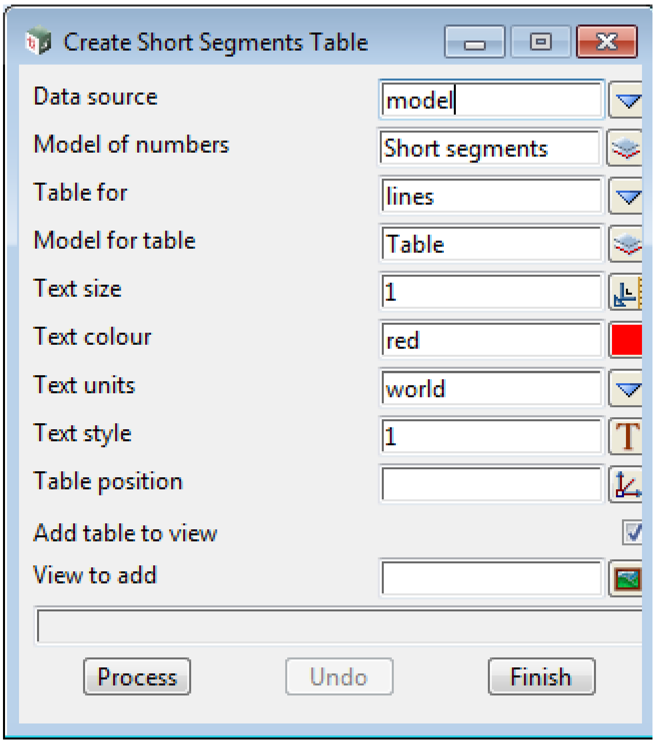


24.16.11 Short Segments Table

Position of option on menu: **Drafting =>Text and Tables =>Short segment table**

This panel is used to create a short segment table.

This option is still under development.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Data source	input box	model	model, string, view

type of data to process

Model/string/View of numbers

data to process and build a short segment table for.

Table for	choice box	lines	lines, arcs, lines and arcs

type of short elements to tabulate.

Model for table	model box	Table	

model to place the short segment table in.

Text size	input box	1	

text size.

Text colour	colour box	red	

text colour

Text units	choice box	world	world, pixels

units for the text size - default is world units.

Text style	input box	1	

textstyle for the created text.

Table position input box

location for the top left hand corner of the table. Either type in an x y position or click on "+" and use "Pick xyz" to locate the table.

Add table to view tick box tick

if tick, add the model containing the table to the view.

View to add input box

view to add the table to.

Process button

create and place the short segment table

Undo button

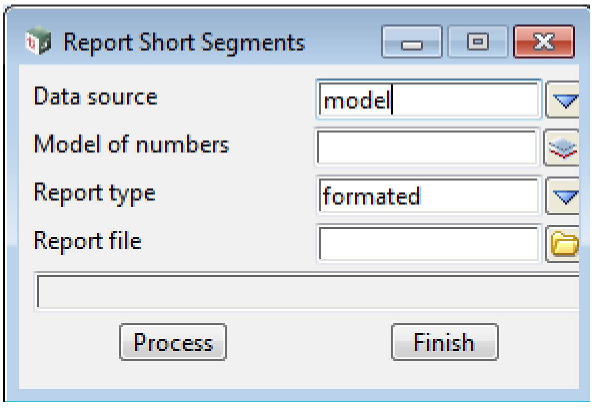
undo the last short segment table created whilst the panel has been up.

24.16.12 Short Segment Report

Position of option on menu: **Drafting =>Text and Tables =>Short segment report**

This panel is used create a report on the short segments.

This option is still under development



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Data source	input box	model	model, string, view

type of data to process

Model/string/View of numbers

data to process and report on the short segments.

Report file report box

name of the report file.

Process button

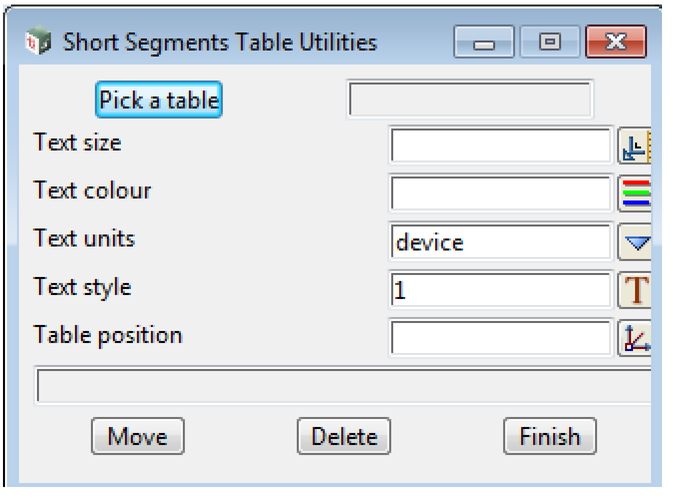
create the short segment report

24.16.13 Short Segments Table Utilities

Position of option on menu: **Drafting =>Text and Tables =>Short segment table utilities**

This panel is used modify an existing short segments table.

This option is still under development.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Pick a table	string select		

select the short segments table to modify.

Text size	input box	1	

text size.

Text colour	colour box	red	

text colour

Text units	choice box	world	world, pixels

units for the text size - default is world units.

Text style	input box	1	

textstyle for the created text.

Table position	input box		

location for the top left hand corner of the table. Either type in an x y position or click on "+" and use "Pick xyz" to locate the table.

Move	button		

move the selected table to the new location.

Delete	button		

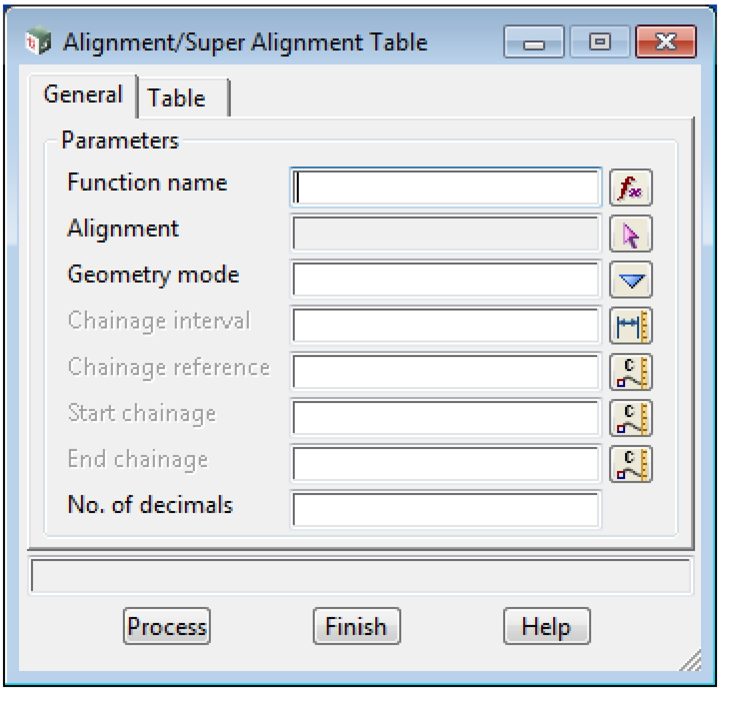
delete the selected table.

24.16.14 Alignment & Super Alignment Table

Position of option on menu: **Drafting =>Text and Tables =>Alignment/Super alignment table**

This option is used to create a report and/or a table in a model for an alignment and super alignment string in an IP or element format.

This option is under development.

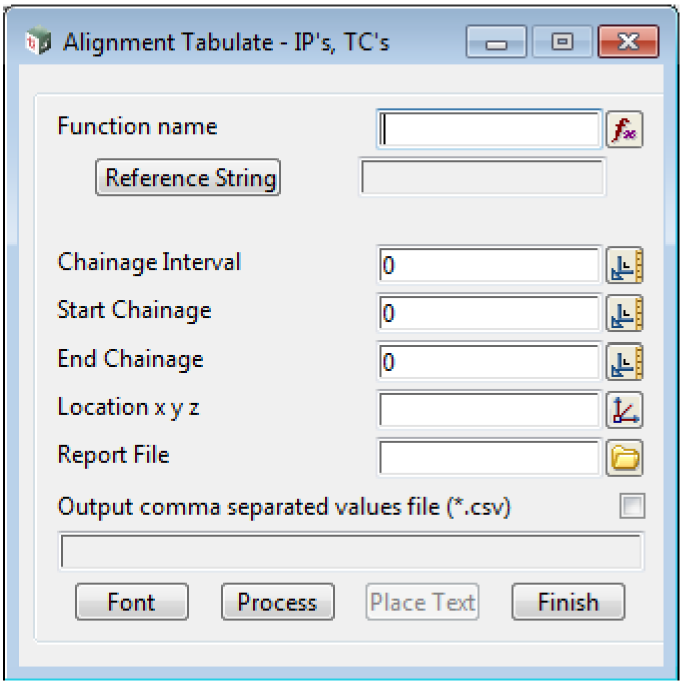


24.16.15 Tabulate Alignment - IP's and CT's

Position of option on menu: **Drafting =>Text and Tables =>Tabulate alignment - IP's TC's**

This option is used to create a report and/or a table in a model of an alignment string in an IP and tangent point format.

The table is created by a function so that if the alignment string changes, the table can be easily recalced.



The fields and buttons used in this panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Function name input box

name of the function

Reference string string select

select the string to create the table and/or report for.

Chainage interval input box 0

if zero, only information on the tangent and intersection points is given.

If non-zero, the chainage interval to use to report the information about the reference string. The horizontal tangent and intersection points are included.

Start/End chainage input box 0

start/end chainage to use. If 0, the start/end chainage is used.

Location x y z input box

location for the top left hand corner of the table. Either type in an x y position or click on "+" and use "Pick xyz" to locate the table.

Report file file box

if non blank, name of the report file.

If blank, don't create the report.

Font button

brings up panel for defining information about the text (size, model colour etc.).

Process button

process the reference string and create the report. The table is not yet created.

Place text button

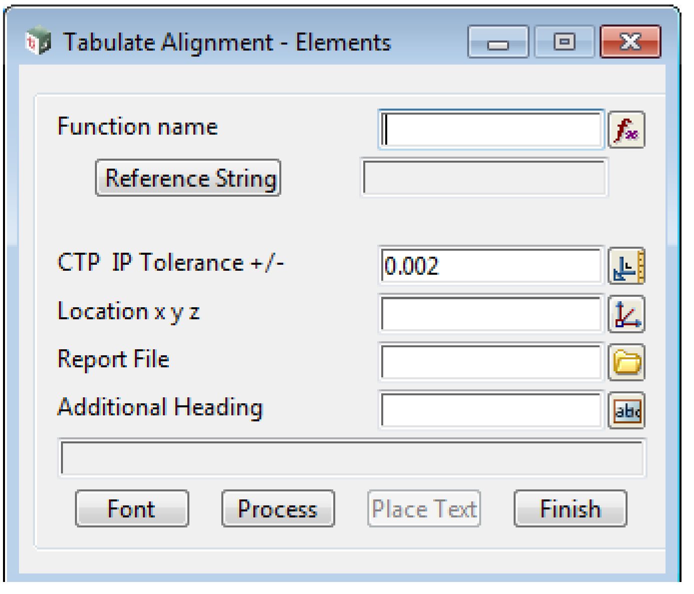
create the table and place it at the given location.

24.16.16 Tabulate Alignment - Elements

Position of option on menu: **Drafting =>Text and Tables =>Tabulate alignment - elements**

This option is used to create a report and/or a table in a model of an alignment string in an horizontal element format.

The table is created by a function so that if the alignment string changes, the table can be easily recalced.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function name	input box		

name of the function

Reference string	string select		

select the string to create the table and/or report for.

CTP IP tolerance +/-	input box	0.002	

xxxx.

Location x y z	input box		

location for the top left hand corner of the table. Either type in an x y position or click on "+" and use "Pick xyz" to locate the table.

Report file	file box		

if non blank, name of the report file.

If blank, don't create the report.

Additional heading	input box		

additional information for the heading above the table.

Font	button		

brings up panel for defining information about the text (size, model colour etc.).

Process	button		

process the reference string and create the report. The table is not yet created.

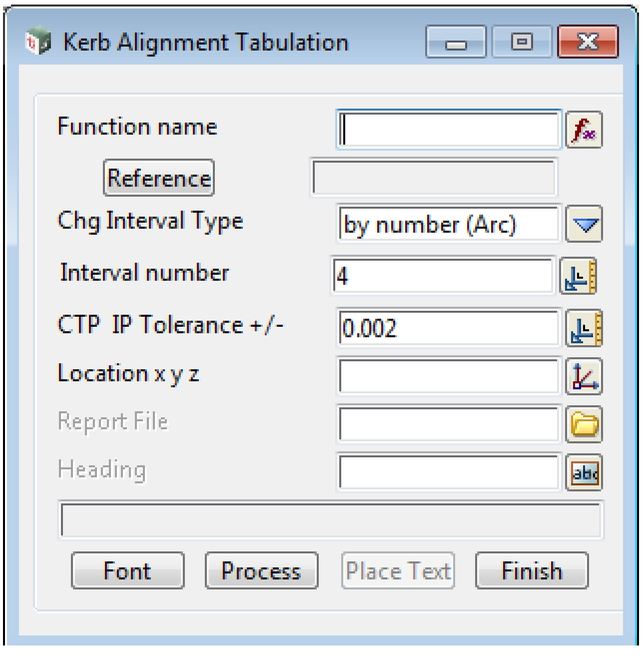
Place text button

create the table and place it at the given location.

24.16.17 Tabulate Kerb Return

Position of option on menu: **Drafting =>Text and Tables =>Tabulate kerb return**

This option creates a table for a kerb return.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function name	input box		
name of the function			
Reference	string select		
select the string to create the kerb return table and/or report for.			
Chg interval type	choice box	by number (Arc)	by number (Arc) by distance (Arc) by special chg file by number (Chord)
Interval number	input box	4	
number of points to divide the alignment into.			
CTP IP tolerance +/-	input box	0.002	
xxxx.			
Interval number	input box	4	
xxxx.			
Location x y z	input box		
location for the top left hand corner of the table. Either type in an x y position or click on "+" and use "Pick xyz" to locate the table.			
Report file	file box		
if non blank, name of the report file. *If blank, don't create the report.*			
Heading	input box		

heading above the table.

Font button

brings up panel for defining information about the text (size, model colour etc.).

Process button

process the reference string and create the report. The table is not yet created.

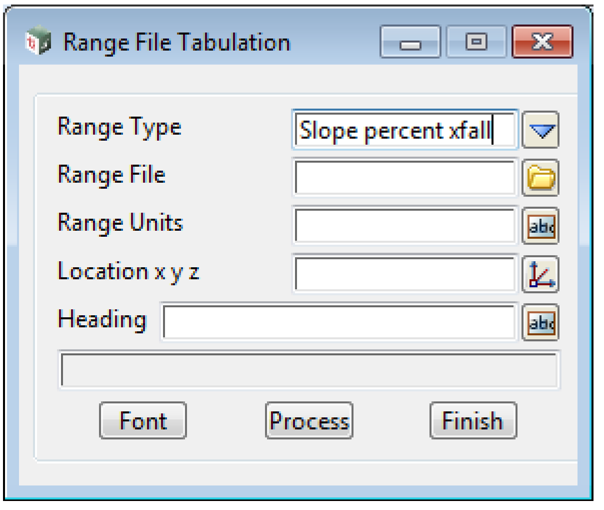
Place text button

create the table and place it at the given location.

24.16.18 Tabulate Range File

Position of option on menu: **Drafting =>Text and Tables =>Tabulate range file**

This option creates a table for a range file.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Range type	choice box	slope percent xfall	slope percent xfall slope 1 v in, slope degrees depth, aspect, height

type of range file to be created.

Range file file box

name of the range file.

Range units input box

text to place beside each line of upper and lower values.

Heading input box

heading to place at the top of the table.

Location x y z input box

location for the top left hand corner of the table. Either type in an x y position or click on "+" and use "Pick xyz" to locate the table.

Font button

brings up panel for defining information about the text (size, model colour etc.).

Process button

create and place the range table.

24.16.19 Symbol / Linestyle Legend - Tabulation

Position of option on menu: **Drafting =>Text and Tables =>Tabulate symbol/linestyle legend**

This option can create a legend showing features and their associated linestyle, colour, symbol *etc.*

There are 3 distinct parts to the macro. Once a data selection is chosen, the data can be analysed by either using a map file or the data features themselves.

Map File Extraction:

If specified, the 12d Map File is read and data sorted by string name (*Name* column).

If the **Extract Data** button is clicked, the **Map File Extraction** grid is populated.

The *Description* column is used in the legend and is controlled by the *Mode* selected, either *Model Name* or *Comment.*

Data only Extraction:

If no map file is specified, then the data is sorted in relation to *Name*, *Linestyle, Colour* and *Weight*.

If the **Extract Data** button is clicked, the **Data Only Extraction** grid is populated.

The *Description* column is used in the legend and is controlled by the *Mode* selected: *Model Name, String Name, Linestyle, Symbol, Ver attr + Name, Name +Ver Attr.*

Once a mode is selected, clicking the Update button will update the *Description* field.

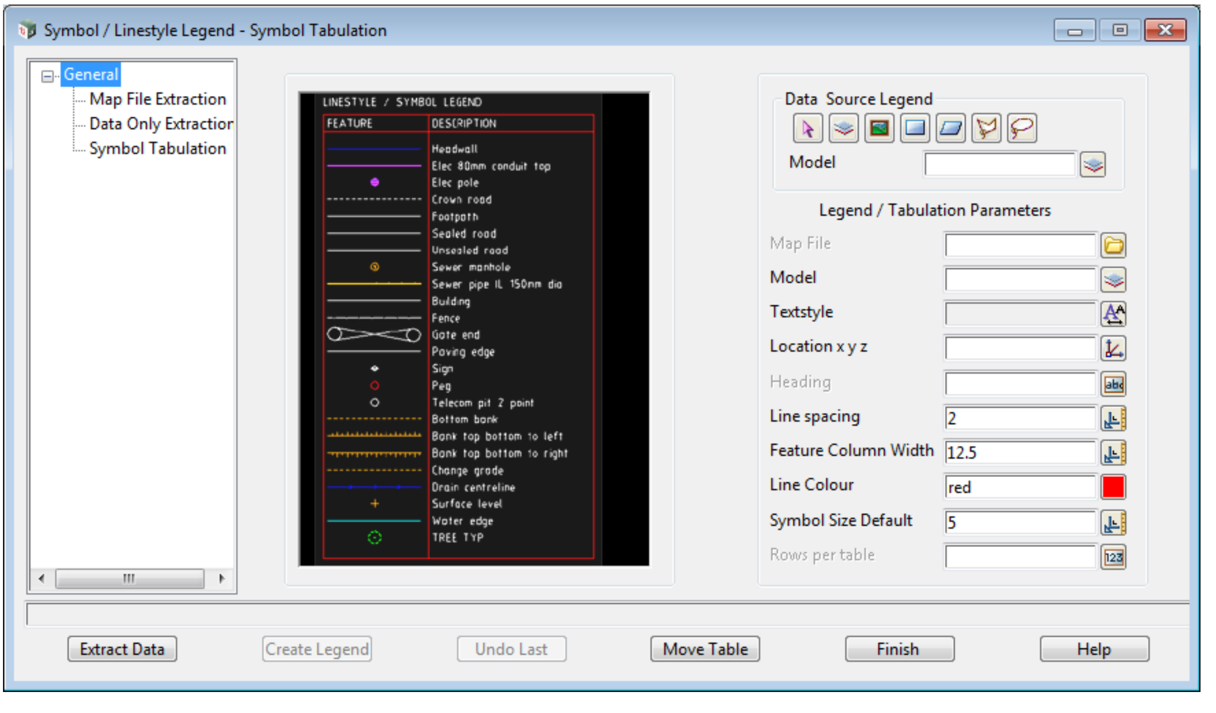
Symbol Tabulation:

This tabulation is for symbols, *e.g.* survey station points.

The table is in the format of Name, Easting, Northing & Level.

The title of the Name column can be changed.

The data to be tabulated is given by the Tabulation by choice box. It can be either tabulating the *Point ID* or the *Vertex Text.*



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Legend / Tabulation Parameters			
Data Source Legend	selection		model, view, string
selection data to extract legend information			
Map File	input		
if non-blank, then the map file is used with the data selection to sort and populate the < Map File Extraction > grid			
Model	input		
model name for legend tabulation			
Textstyle	input		
text parameters for all text			
Location x y z	xyz box		
insertion point for top left corner of legend			
Heading	input		
if non-blank, heading placed at top of legend			
Line spacing	real	2	
spacing for adjustment of lines of text			
Feature Column Width	real	12.5	
width for display of sample line and symbol in legend			
Line Colour	input	red	available colours
the colour of separation lines in the tabulation			
Symbol Size Default	real	5	
default size for symbols in table			
Rows per table	integer		
if blank, the legend table will be created vertically *if non-blank, more than one legend may be created, arrayed in either horizontal or vertical* *the format is: +ve number for Horizontal array -ve number for Vertical array*			
Extract Data	button		
Sorts the source data and populates either the Map File Extraction grid or Data only Extraction grid			
Create Legend	button		
Creates the tabulation using all parameters set in the general tab			
Undo Last	button		T
his undo is only available while the panel is still open and only works on the last table created. *Once the panel is finished all tables can be accessed via the normal 12d undo.*			
Move Table	button		
Any Legend Tabulation can be moved for clarity (Esc or rmb to cancel)			
Symbol Tabulation			
Data Source Tabulation	selection		

selection data to extract tabulation information

Model model box

model name for symbol tabulation

Textstyle textstyle box

text parameters for all text

Location x y z XYZ box

insertion point for top left corner of tabulation

Heading input

if non-blank, heading placed at top of tabulation

Line colour colour box red available colours

the colour of separation lines in the tabulation

Line spacing real 2

spacing for adjustment of lines and text

Station name input STN

maximum 8 characters

Tabulation by choice box Point ID Point ID / Vertex Text

value used to define symbol e.g. Survey Station may use point ID < 901 >

Process button

processes the data

Create Tab button

creates the symbol tabulation using the above parameters.

24.16.20 Text File Input

Position of option on menu: **Drafting =>Text and Tables =>Text file input**

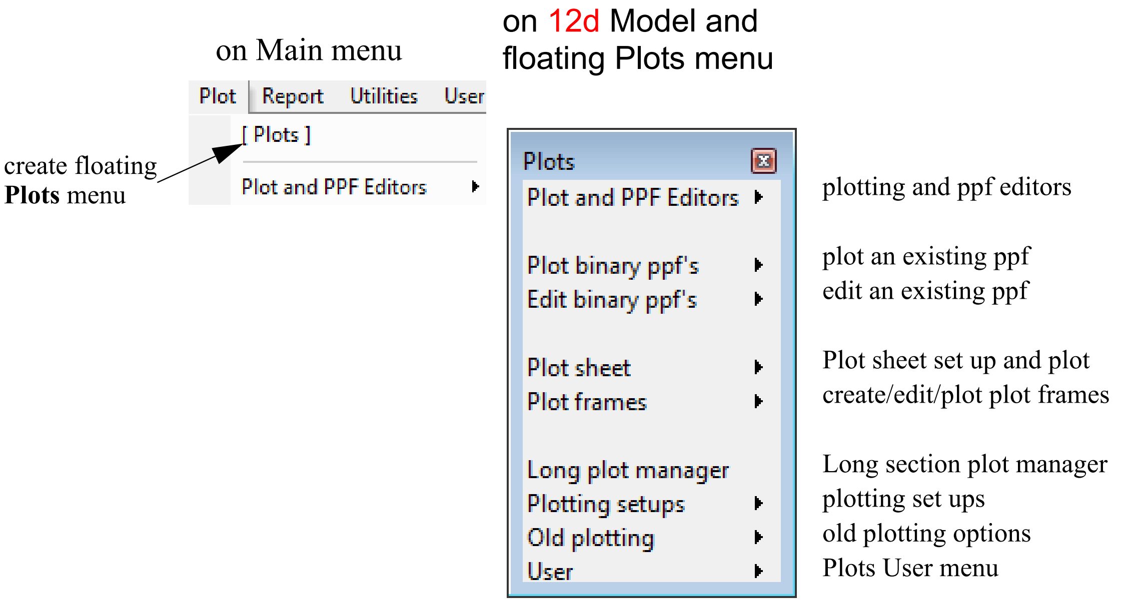
This option has already been documented as 24.16.3 Text Creation/Edit

Drafting =>Text and Tables =>Create/edit paragraph text

25 Plots

Position of menu: **It is on the main menu as Plot**

The **Plots** walk-right menu is



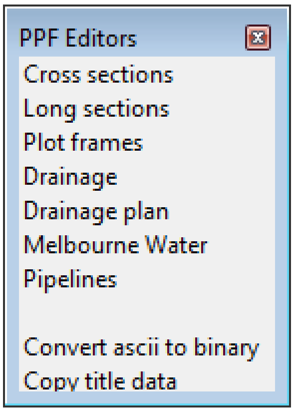
The options 25.2 Plot Binary PPFs and 25.3 Edit Binary PPFs are both walk-right menus which list all the binary PPFs in the project and the when the PPF is double-clicked, it is Plotted/Edited.

For the options see:

25.1 Plot and PPF Editors

Position of menu: **Plot =>Plot and PPF editors**

The **Plot and PPF editors** walk-right menu is



cross section PPF editor
long section PPF editor
creating/editing plot frames
drainage long section PPF editor
drainage plan plot PPF editor
Melbourne Water sewer long section PPF editor
pipeline long section PPF editor

convert old PPF files to new PPF files
copy title data

For the options see:

26.3 Section X Plot PPF Editor
26.4 Long Plot PPF Editor
26.9 Plot Frame and PPF Editor
26.5 Drainage Long Plot PPF Editor
26.8 Drainage Plan Plot PPF Editor
26.6 Melbourne Water Plot PPF Editor
26.7 Pipeline Plot PPF Editor
26.10 Convert Ascii PPF to Binary
26.11 Copy Title Data

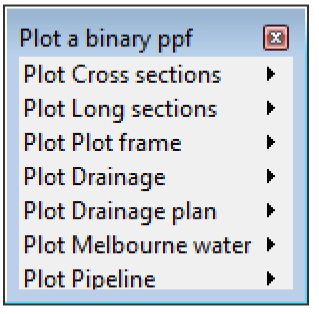
25.2 Plot Binary PPFs

The binary PPFs can have enough information in them to totally generate the plot from the ppf.

The options in this section will run the selected (binary) ppfs and generate the plot as defined by all the plot parameters in the ppf file.

The plot the ppf, just walk right on the appropriate plot type and a walk-right menu will appear with the existing binary ppfs included in the project, and a **[Lib]** and **[User Lib]** to display the ppfs in these two library areas.

When a file is selected from the list, it is then used to generate a plot.



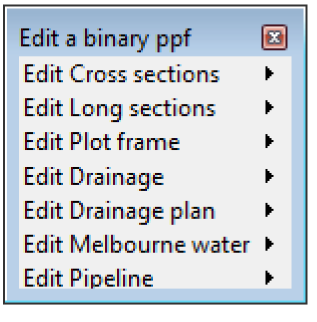
Go to the next section 25.3 Edit Binary PPFs or return to 25 Plots.

25.3 Edit Binary PPFs

The options in this section will bring up the selected (binary) ppfs into the appropriate PPF Editor.

The edit the ppf, just walk right on the appropriate plot type and a walk-right menu will appear with the existing binary ppfs included in the project, and a **[Lib]** and **[User Lib]** to display the ppfs in these two library areas.

When a file is selected from the list, it is then opened in the appropriate PPF Editor.



Go to the next section 25.4 Plot Sheet or return to 25 Plots.

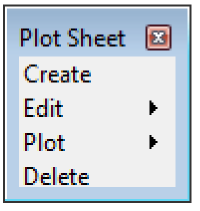
25.4 Plot Sheet

Position of menu: **Plot =>Plot sheet**

The ***Plot sheet*** option is for defining the layout of plotting areas on a sheet of paper and to produce a pdf file of what is on the sheet of paper.

That is ***Plot Sheet*** is for setting up plot compositions. For V11 it is only for setting up areas from plan views.

The **Plot Sheet** menu is:



See

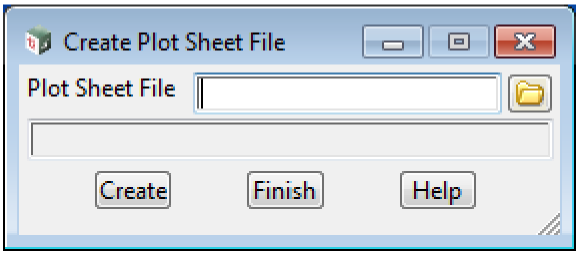
25.4.1 Plot Sheet Create
25.4.2 Plot Sheet Edit
25.4.3 Plot Sheet Plot
25.4.4 Plot Sheet Delete

25.4.1 Plot Sheet Create

Position on menu: **Plot =>Plot sheet =>Create**

Plot sheet create is for creating a new plot sheet file.

Selecting **Create** brings up the **Create Plot Sheet File** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Plot sheet file	file box		*.12dpsf files

name for the file to create.
This file name must not already exist in the working folder.

Create	button		

when pressed, a new plot sheet file is created with the name given in the **Plot sheet file** *field and then the* **Plot Sheet Create/Editor** *panel is started with that file as the Plot Sheet File being edited. See* 25.4.2.1 Plot Sheet Create/Edit.

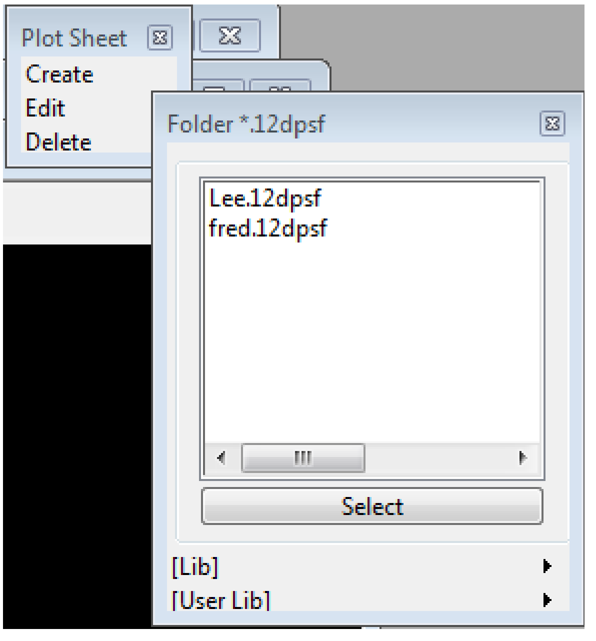
Go to the next section 25.4.2 Plot Sheet Edit or return to 25.4 Plot Sheet.

25.4.2 Plot Sheet Edit

Position on menu: **Plot =>Plot sheet =>Edit**

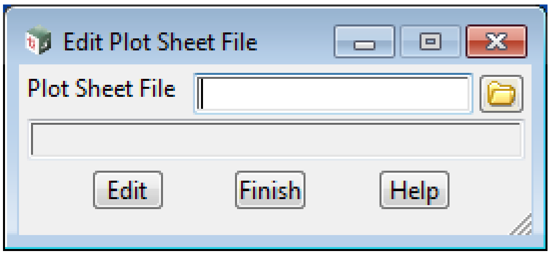
The **Edit** option works in two ways - you can click on **Edit** itself or walk right on **Edit** and see a list of all the plot sheet files in the working folder.

The **Edit** walk right lists all the plot sheet files is the working folder.



Selecting a plot sheet file from the list, or from another area, brings up the **Plot Sheet Create/ Editor** panel already loaded with the selected plot sheet file. *See* 25.4.2.1 Plot Sheet Create/ Edit.

By clicking on **Edit** itself, the **Edit Plot Sheet File** panel is brought up.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Plot sheet file	file box		*.12dpsf files

name for the plot sheet file to edit.
This file name must exist in the working folder.

Edit	button		

when pressed, the existing plot sheet file with the name given in the **Plot sheet file** *field is edited and the* **Plot Sheet Create/Editor** *panel is started with that file as the Plot Sheet File being edited. See* 25.4.2.1 Plot Sheet Create/Edit.

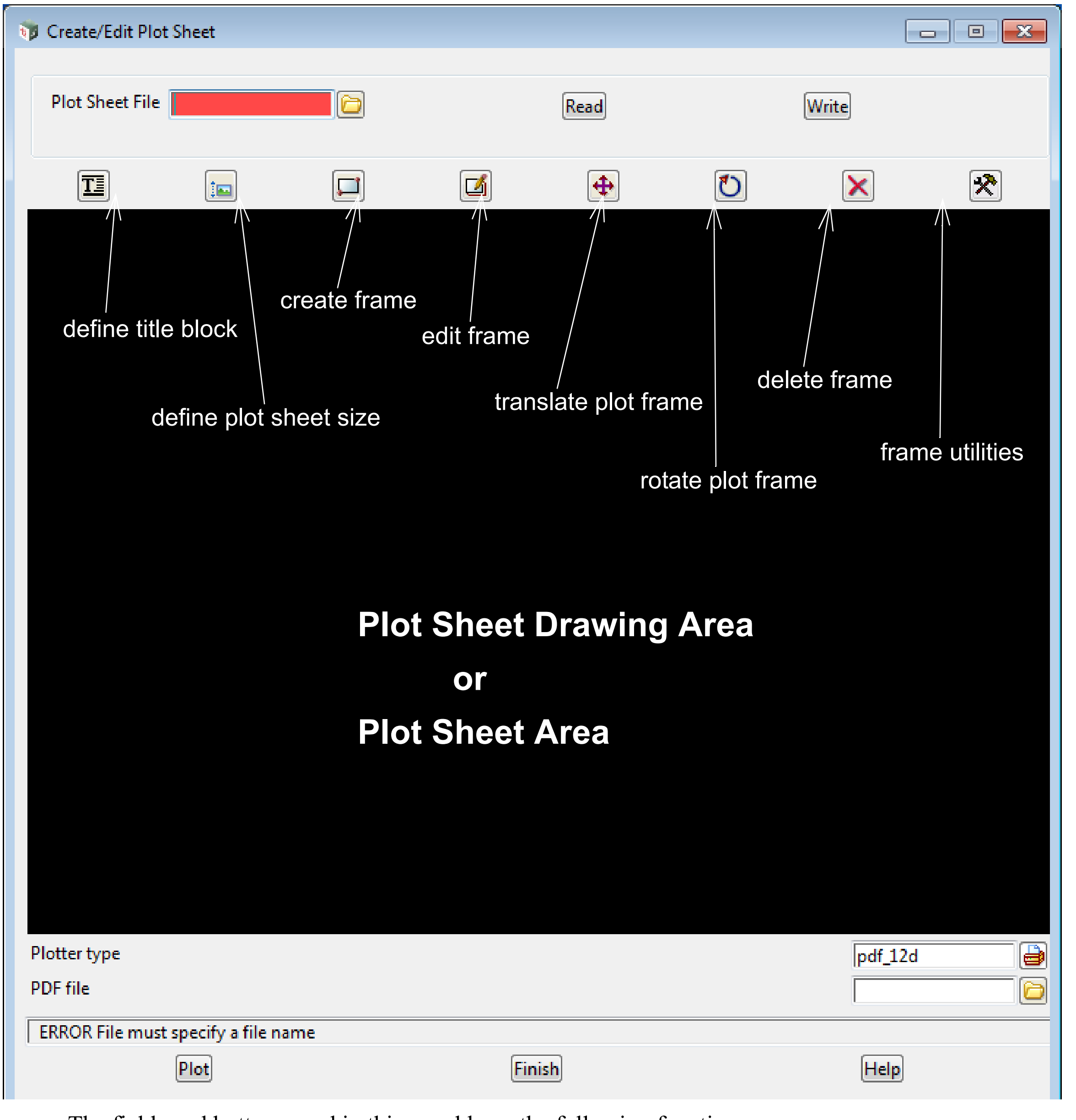
Go to the next section 25.4.2.1 Plot Sheet Create/Edit or return to 25.4 Plot Sheet.

25.4.2.1 Plot Sheet Create/Edit

The **Plot Sheet Create/Edit** panel is for creating/editing the layout of plotting areas on a sheet of paper, and then to produce a pdf file of what is on the sheet of paper.

If the **Plot Sheet Create** option was used, then the **Plot Sheet Edit** panel is brought up with the new file loaded but there is nothing defined for the Plot Sheet.

If the **Plot Sheet Edit** option was used, then the **Plot Sheet Edit** panel is brought up with the existing file loaded so there will already be some information defined on the Plot Sheet.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Plot sheet file	file box		*.12dpsf files

name for the file to save/read in the information for a plot sheet layout.

Read button

read in the plot sheet file given in the **Plot sheet file** *field and load the information into the panel. For any Frames in the file, a Plot Frame is created, and a model and the Plot Frame added to the* *model. If a view has been recorded for that model when the plot sheet file was written, and that view currently exists, then the model is added to that view.*

Write button

write out the current information defined in the panel to a plot sheet file with the name given in the **Plot sheet file** *field.*

Plot Sheet Area

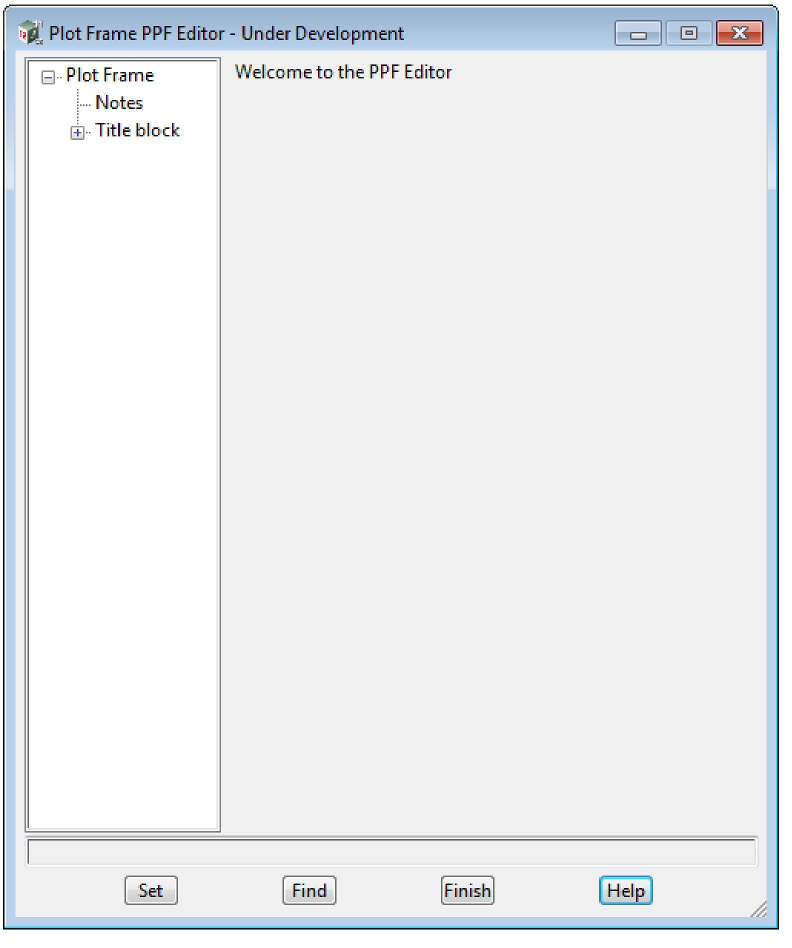
*the black area represents the sheet of paper being plotted (the **Plot Sheet**) with (0,0) in the bottom left hand corner of the plot sheet.*

*The areas where plotting will occur on the Plot Sheet are defined by **Frames** and each created Frame is displayed in the black area. A Frame contains all the information for producing the plot inside the area of the Frame.*

*For example, a Frame may be a **Plan Frame** and defines an area of a Plan View to plot. The Frame then needs has a Scale to give the Frame a world size, a rotation and a World Coordinate to fully define a region to plot. Using these values, a Plot Frame is then created and linked to the Frame. The Plot Frame is added to a Plan View to define which data is to be plotted inside the Frame.*

Title block button

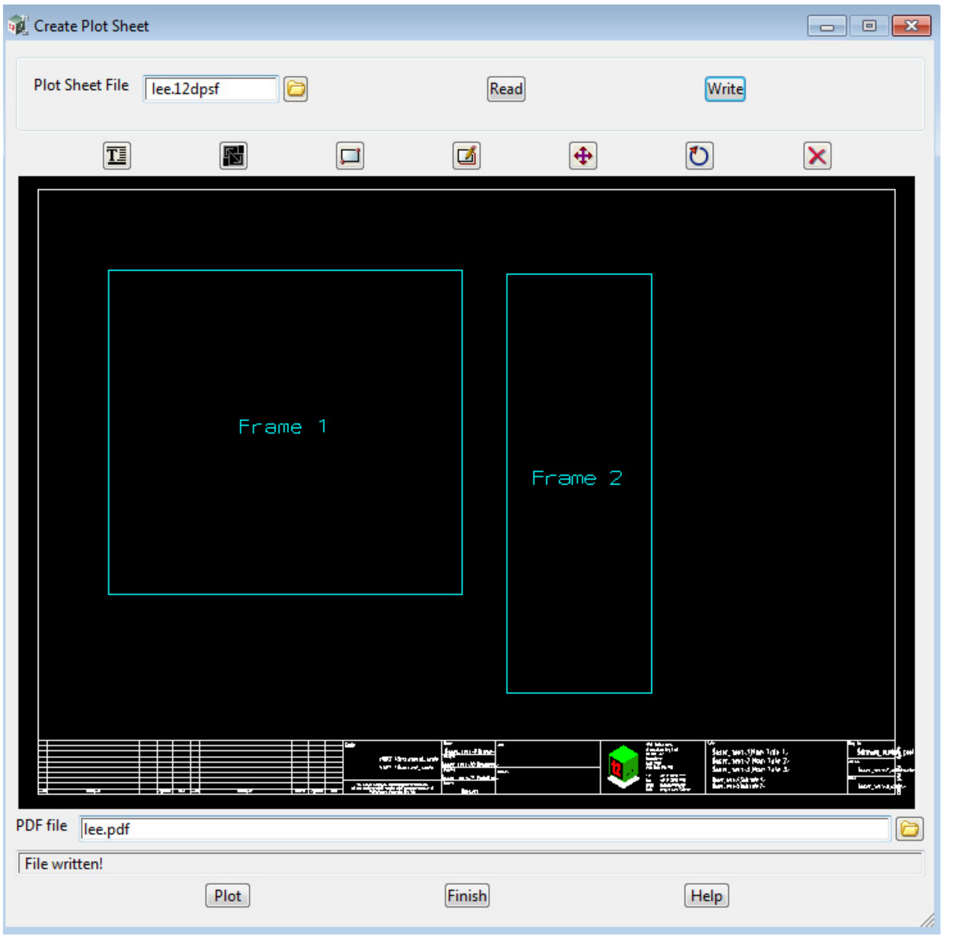
brings up the Title Block section of a Plot Frame.



After the title block information has been filled in, the **Set** *button must be pressed and then* **Finish** *to remove the panel.*

If the title block file (tbf file) includes the Sheet size, then that size is automatically used to set the sheet size for the **Sheet Size** *button.*

If a user title block has been selected, the text and line work in the title block will be drawn in the Plot Sheet Area so that you see where to add Frames to the Plot Sheet without overwriting the title block. No *substitution is made for the $ text variables in the title block file.*



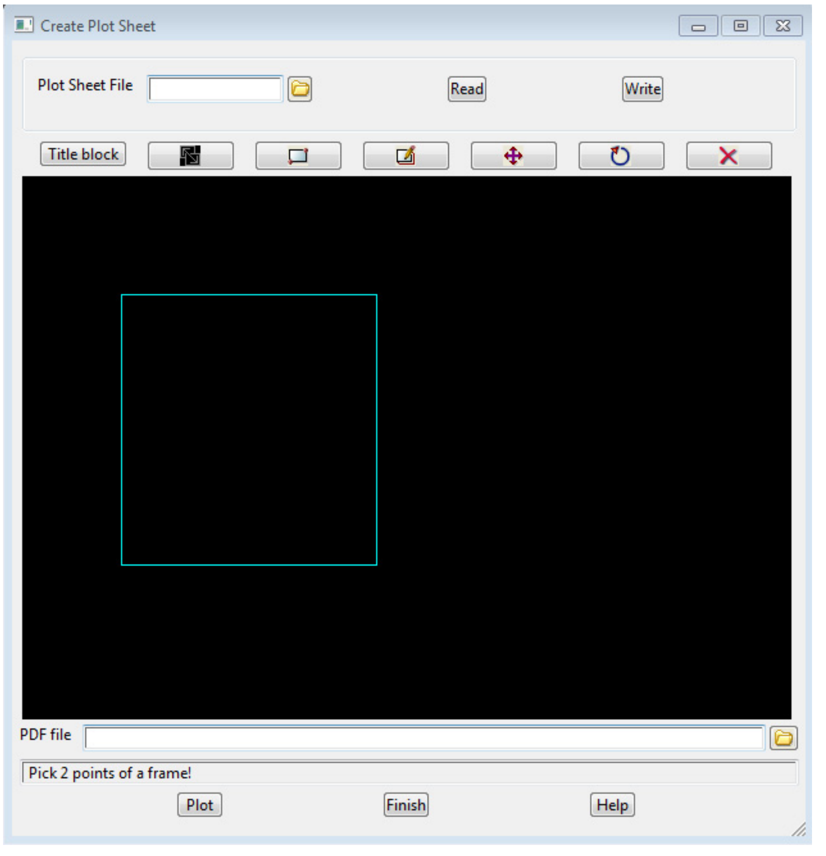
Sheet size button

brings up a **Sheet size** *box for entering the size of the size of the Plot Sheet.*
Once a size has been typed in, or selected from the pop up, the Plot Sheet Area is drawn with an aspect ratio to match the selected paper size.

If there is a Sheet size in the selected user title block file, the Sheet size is automatically set to that size and the **Sheet Size** *button does not need to be used.*

Create frame button

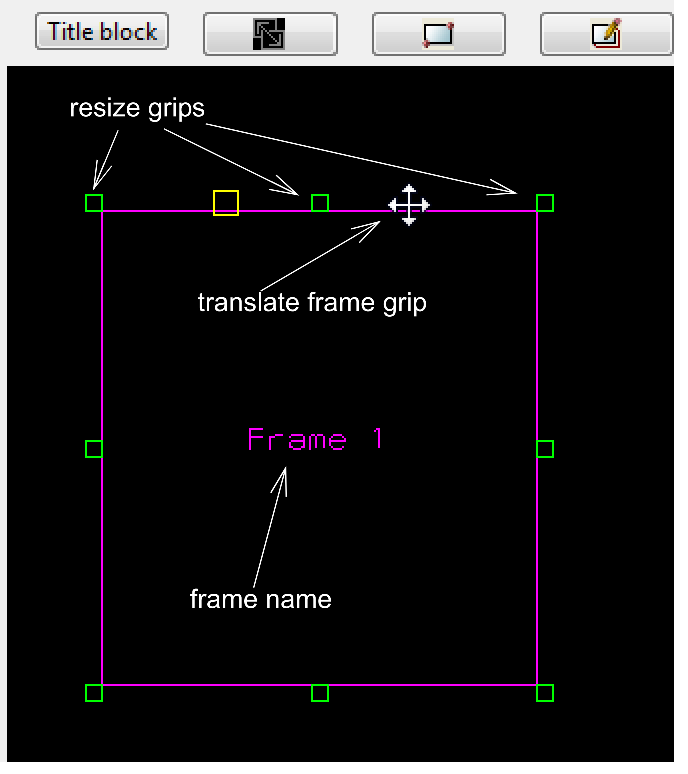
this button is used to define a Frame in the plot sheet area.
After clicking the button, two points are selected in the plot sheet area to define the area on the plot sheet that the Frame will plot to.



Currently all created Frames are Plan Frames so after the Frame is created, the **Create Frame for Plan Plot** *panel is displayed. See* 25.4.2.1.1 Create Frame for Plan Plot.

Edit frame button

this button is used to edit an existing Frame in the plot sheet area.
After clicking the button, a Frame is selected by clicking LB ***anywhere inside*** *the Frame in the Plot Sheet Area.*
When the Frame has been tentatively picked with the LB but ***not*** *yet* ***accepted***, ***Resize*** *grips are displayed on the four corners and on the four sides of the Frame. These grips are used to resize the Frame, and hence the associated plotting area.*
Plus when the cursor is over the edge of a Frame, a ***Translate*** *grip is displayed. The* ***Translate*** *grip is used to move the Frame around in the Plot Sheet Area.*



*How the **Resize** grips extend the plotting area depends on* **Make centre of frame the origin**.

> *If* **Make centre of frame the origin** *is **ticked** then no matter which resize grip is used, the resulting plotting area is still **centred** on the* **World Coordinate**.
>
> *If* **Make centre of frame the origin** *is **not ticked** then the plotting area is extended in the direction indicated by the Resize grip and the* **World Coordinate** *may be changed.*

If you haven't accepted the Frame after the Resizing and Translating, then you can pick another Frame to resize or translate by simply clicking LB inside the new Frame.

When a Frame is accepted after a tentatively pick, then the **Create Frame for Plan Plot** *panel is displayed so that the Frame properties can be modified. See* 25.4.2.1.1 Create Frame for Plan Plot.

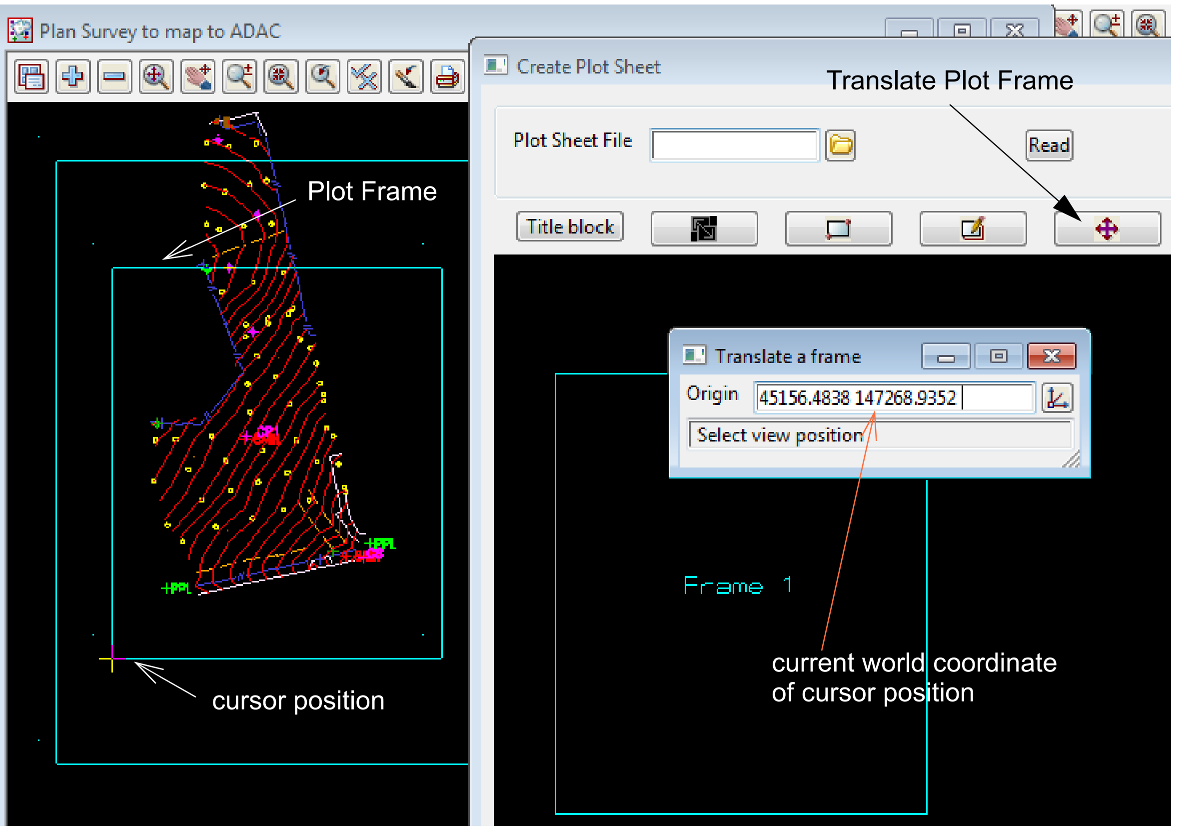
After the **Create Frame for Plan Plot** *panel has been finished, the* **Edit** *is still active and another Frame can be selected to edit.*

Translate Plot Frame button

this button is used to change the value of the **World Coordinate** *of a Frame by dynamically translating the **Plot Frame** that corresponds to the Frame.*

*After clicking the button, a Frame is selected by clicking LB **anywhere inside** the Frame in the Plot Sheet Area and then **accepting** the pick.*

When the Frame is accepted, a **Translate a Frame** *panel is displayed showing the value of the* **World coordinate** *for the Frame. Then when you move the cursor onto a Plan View which has the **Plot Frame** associated with the selected Frame on it, the* **World Coordinate** *of the Plot Frame and Frame are changed to the current cursor position. This continues until a new* **World Coordinate** *is selected on the Plan View.*



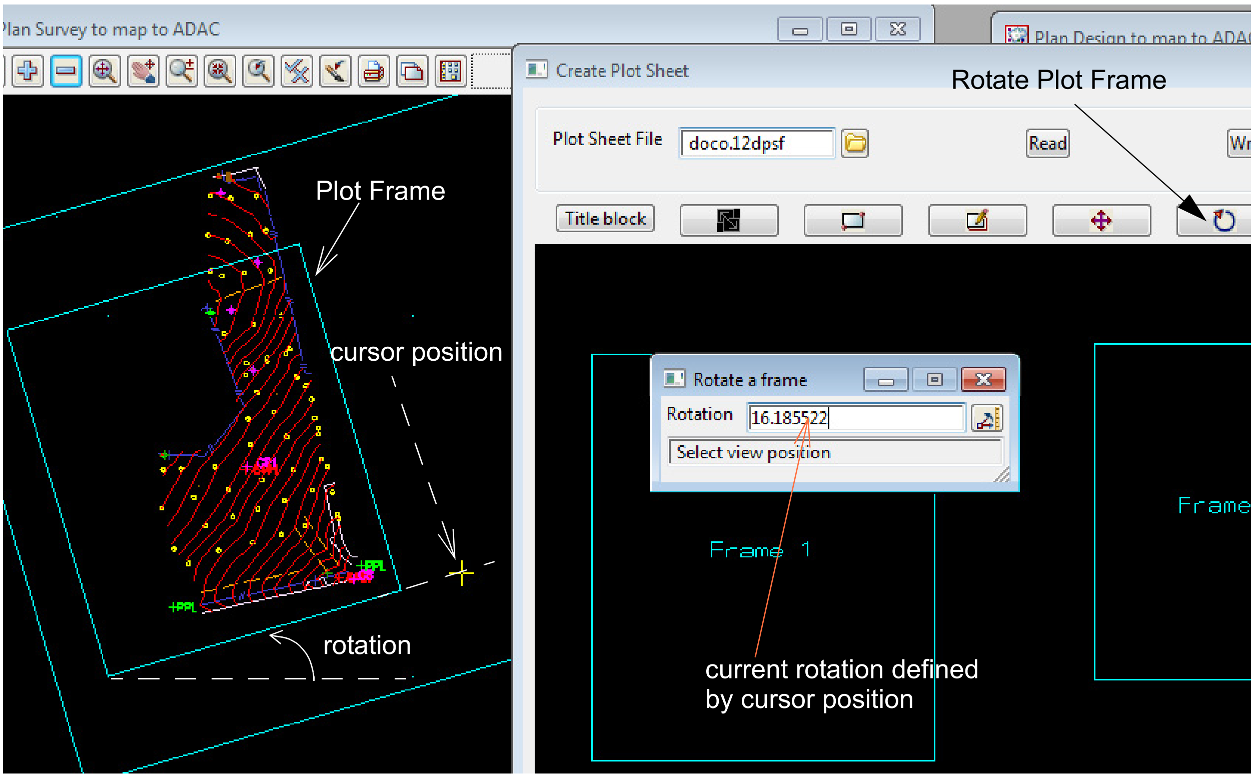
Once the Frame is translated then you can pick another Frame to translate.

Rotate Plot Frame button

this button is used to change the value of the **Rotation** *of a Frame by dynamically rotating the **Plot Frame** that corresponds to the Frame.*

*After clicking the button, a Frame is selected by clicking LB **anywhere inside** the Frame in the Plot Sheet Area and then **accepting** the pick.*

When the Frame is accepted, a **Rotate a Frame** *panel is displayed showing the value of the* **Rotation** *for the Frame. Then when you move the cursor onto a Plan View which has the **Plot Frame** associated with the selected Frame on it, the* **Rotation** *of the Plot Frame and Frame are changed to the rotation defined by the current cursor position. This continues until a cursor position is accepted and that defines the new* **Rotation** *for the Plot Frame and Frame.*



Once the Frame is rotated then you can pick another Frame to rotate.

Delete Frame button

this button is used to delete a Frame in the ***Plot Sheet.*** *The associated Plot Frame and model containing the Plot Frame are also deleted.*

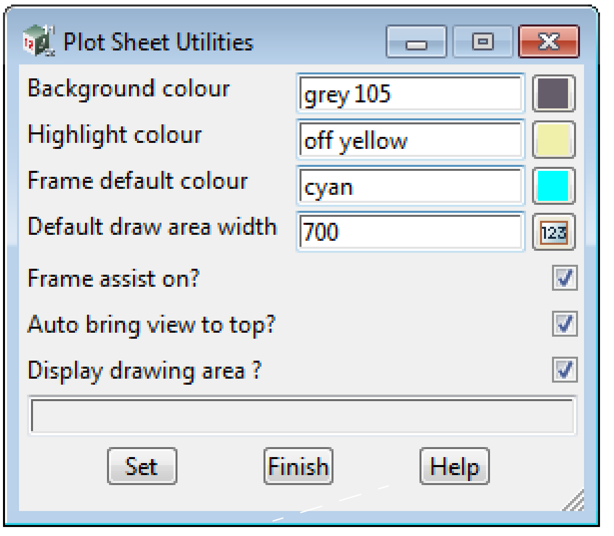
After clicking the button, a Frame is selected by clicking LB ***anywhere inside*** *the Frame in the Plot Sheet Area and then accepting the pick.*

When the Frame is accepted, the Frame is deleted, and its associated Plot Frame and the model containing the Plot Frame are also deleted.

Once a Frame is deleted, another Frame can then be selected to delete.

Utilities button

this button brings up the **Plot Sheet Utilities** *panel to change the settings for the Plot Sheet.*



Background colour colour box grey 105 available colours

the colour for the background of the Plot Sheet drawing area. "Grey 105" is a good colour to use because both blank and white strings are visible when drawn on "Grey 105".

Highlight colour colour box white available colours

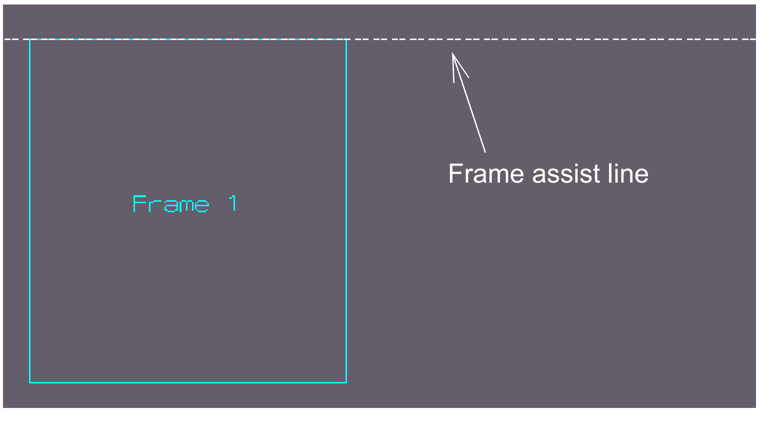
the colour to highlight the Frames in the Plot Sheet Drawing Area when they are selected.

Frame default colour colour box cyan available colours

the colour to draw the Frames borders in the Plot Sheet drawing area.

Frame assist on tick box ticked

*if **ticked**, when a new Frame is being created or an existing Frame moved in the Plot Sheet Area, when the cursor it is near the side of another Frame then guide lines are drawn in the Plot Sheet Area to help in placing the new Frame to be on the same horizontal or vertical line as the sides of existing frames.*



Auto bring view to top? tick box ticked

*if **ticked**, when the buttons Edit Frame or Delete Frame, or Plot Frame Move or Plot Frame Rotate, are selected and a Frame then selected, the View that the corresponding Plot Frame is on is brought to the top so that the View is not obscured by other Views.*

Display drawing area tick box ticked

*if **ticked** and the title block file has a drawing area defined, the n it is shown in the Plot Sheet Area.*

The drawing area of the title block when the buttons Edit Frame or Delete Frame, or Plot Frame

Fields and Buttons Under the Plot Sheet Area

PDF file	file box	*.pdf files

name of the file to write the pdf plot out to.

Plot	button

generate a pdf plot with the name given in the **PDF file** *field.*

Finish	button

when the **Finish** *button is selected, all the Plot Frames and the models containing the plot frames, that have been created by* ***Plot Sheet*** *are deleted.*

Go to the next section 25.4.2.1.1 Create Frame for Plan Plot or return to 25.4 Plot Sheet.

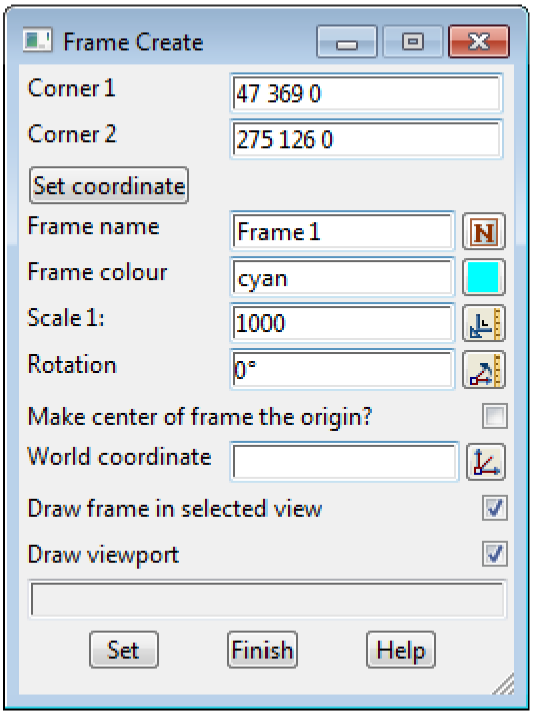
25.4.2.1.1 Create Frame for Plan Plot

The **Frame Create** panel defines the information required to create a plan plot in the Frame.

The size of the Frame in ***millimetres*** is known from the Plot Sheet and so with a ***Scale***, the Frame then defines a rectangle in world units (metres).

A ***World Coordinate*** for one point in the Frame and a ***Rotation*** then fully defines a world rectangle.

A ***Plot Frame*** of a given name can then be created and added to a view to represent the information to be plotted in the Frame.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Frame name	text box		

the name for the frame. It must be unique amongst all the Frames for the Plot Sheet.
A name is automatically generated but it can be changed as long as it is still unique for the Plot Sheet.

*A **Plot Frame** will be created with the name of the plot sheet file (without the .12dpsf) followed by a space and the* **Frame name**.

*A **model** will also be created with the name **"PF "** followed by the Plot Frame Name. The Plot Frame will be added to this model.*

Frame colour	colour box		available colours

the colour for the frame, and also for the Plot Frame that is created.

Scale 1:	measure box		available measures

scale for the plan plot.

Rotation	measure box	0 degrees	available measures

angle of rotation for the plot frame. The angle is in degrees and is measured in a counter clockwise direction from the positive x axis.

Make centre of frame the origin?	tick box		not ticked

*if **ticked**, the* **World coordinate** *is taken to be the world coordinate for the **centre** of the Frame.*

*If **not ticked**, the* **World coordinate** *is taken to be the world coordinate for the **bottom left hand corner** of the Frame.*

World coordinate input **xyz ops** menu

x_origin y_origin z_origin

this is a point in world units to be used as either the centre, or the bottom left hand corner, of the Frame. The values can either be typed in as three values separated by spaces, or selected by clicking LB on the **xyz ops** *icon and then picking a point in a plan view.*

When the **World coordinate** *is selected from a plan view, then after the* **Set** *button is clicked, the plot frame is created, added to the created model, and the model added to the view that the* **world coordinate** *was selected from.*

Draw viewport tick box ticked

*if **ticked**, when a plot is generated, a box is drawn around the Frame.*

Set button

when **Set** *is pressed, a Plot Frame of the name plot sheet file (without the .12dpsf) followed by a space and the* **Frame name** *is created. A model, with the name "PF " followed by the Plot Frame name, is then created and the Plot Frame is added to this model.*

The model containing the Plot Frame is then added to the plan view that the **World coordinate** *was selected from.*

Go to the next section 25.4.3 Plot Sheet Plot or return to 25.4 Plot Sheet.

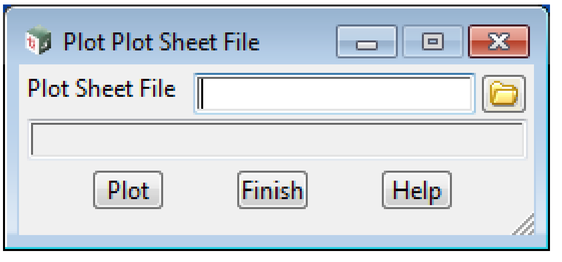
25.4.3 Plot Sheet Plot

Current position on menu: **Plot =>Plot sheet =>Plot**

Plot sheet plot is for plotting a plot sheet file that is on disk without.having to bring up the plot sheet file in the **Plot Sheet Editor**.

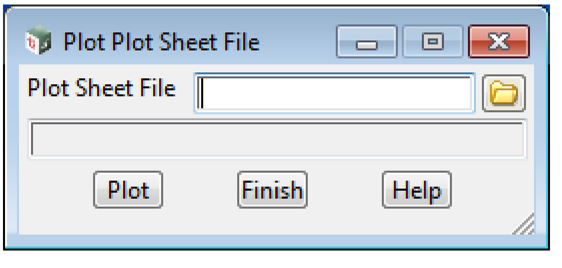
The **Plot** walk right operates in three ways.

1. The **Plot** walk-right displays a list of all the plot sheet files (*.12dpsf files) and by double clicking on a plot sheet file in the list, the plot sheet file will be run and the associated plots produced.
2. If the walk right ***Plot =>Plot sheet =>Plot*** is being displayed from the **Main Menu** on the top of ***12d Model***, then clicking on the name **Plot** on the **top of the walk right** will bring up the **Plot Plot Sheet File** panel.



Selecting the name of the plot sheet file in the field **Plot sheet file** and then pressing **Plot** will generate the plot.

3. If the walk right is being displayed from a pinned menu, then clicking on **Plot** on the **Plot Sheet menu without walking right** will also bring up the **Plot Plot Sheet File** panel.



Again as in the previous case, selecting the name of the plot sheet file in the field **Plot sheet file** and then pressing **Plot** will generate the plot.

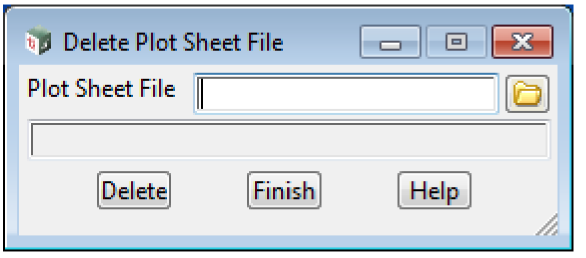
Go to the next section 25.4.3 Plot Sheet Plot or return to 25.4 Plot Sheet.

25.4.4 Plot Sheet Delete

Position on menu: **Plot =>Plot sheet =>Delete**

Plot sheet ***Delete*** is for deleting a plot sheet file from disk.

Selecting **Delete** brings up the **Delete Plot Sheet File** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Plot sheet file	file box		*.12dpsf files

name for the file to be deleted.

Delete	button		

delete the file given in the **Plot sheet file** *field.*

Return to 25.4 Plot Sheet.

25.5 Plot Frames

Position of menu: **Plot =>Plot frames**

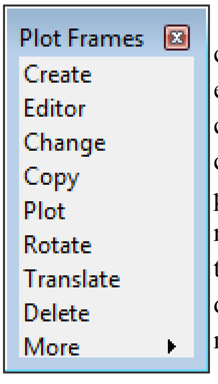
Although plan view plots can be generated in 12d Model by plotting the contents of the plan view with the **Plot** option, the plot will only have a simple title block.

To create plan plots of an arbitrary size, rotation and scale and with complex title blocks, 12d Model uses **plot frames**.

Basically a plot frame consists of a sheet size (in mm), margins within the sheet and a scale for the plot. This will define a plotting area in world coordinates (the plot frame).

The plot frame can be arbitrarily positioned and rotated on a plan view.

The **Plot frames** walk-right menu is



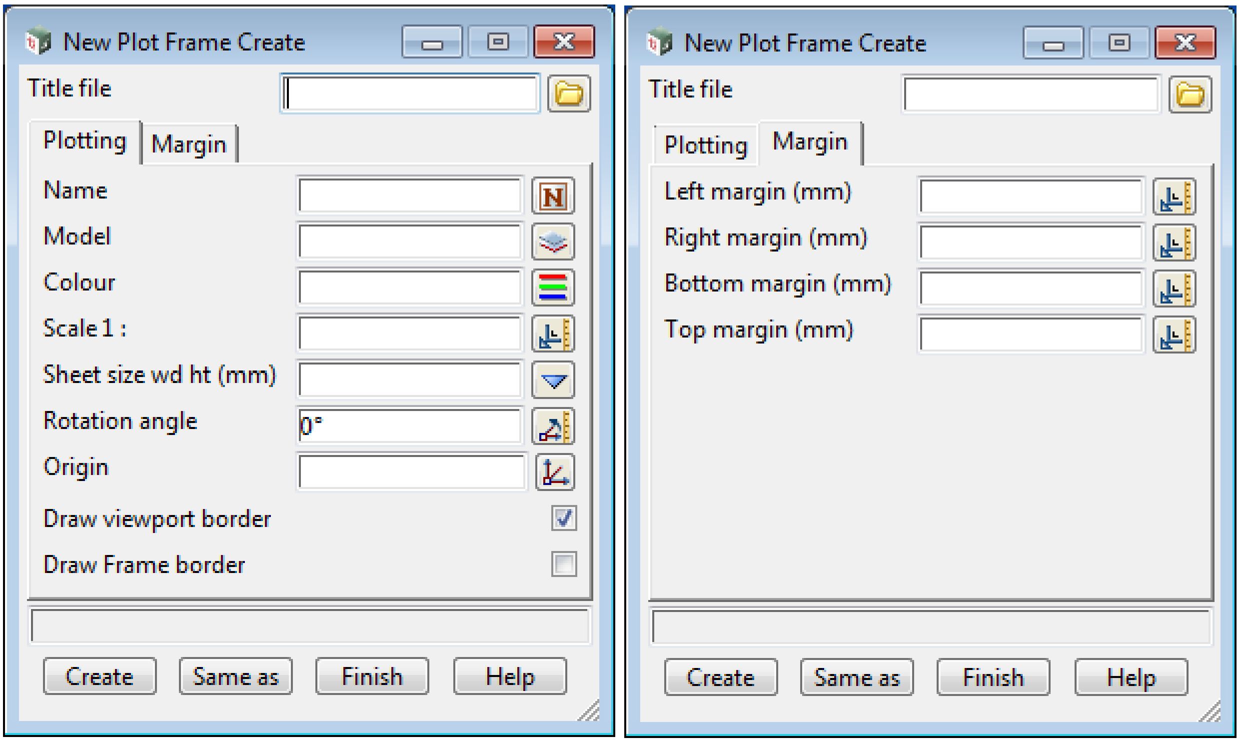
create a plot frame
edit an existing plot frame
change name, model and colour of a plot frame
copy a frame
plot one or many plot frames
rotate a plot frame
translate a plot frame
delete a plot frame
more plot frame options

For the option *Create*, go to	25.5.1 Plot Frame Create
Editor	25.5.2 Plot Frame Editor
Change	25.5.3 Plot Frame Change
Copy	25.5.3 Plot Frame Change
Plot	26.9 Plot Frame and PPF Editor
Rotate	25.5.6 Plot Frame Rotate
Translate	25.5.7 Plot Frame Translate
Delete	25.5.8 Plot Frame Delete
More	25.5.9 More Plot Frames

25.5.1 Plot Frame Create

Position of option on menu: **Plot =>Plot frames =>Create**

On selecting the **Create** option, the **New Plot Frame Create** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Title file	title file box		all title files

name for the title file to use when plotting the plot frame. The title block file can contain default information for the rest of the panel fields. See 26.2.6.5 Values Used for Defaults when Creating Plot Frames

Name	name box		available names

name for the plot frame.

Model	model box		available models

the model for the plot frame.

Colour	colour box		available colours

the colour for the plot frame.

Scale 1:	measure box		available measures

scale for the plot.

Sheet size wd ht (mm)	choice box		available sheet sizes

the width and height values (separated by spaces) or the name of a user defined sheet size.

Rotation angle	measure box	0 degrees	available measures

angle of rotation for the plot frame.

Origin	input		**xyz ops** menu

x_origin y_origin z_origin

origin (in world units) for the corner of the plot frame - given as three values separated by spaces. The values can either be typed in, or selected by clicking LB in the **origin** *panel view and getting up the* **xyz ops** *menu and selecting the* **pick xyz** *option.*

Draw viewport border tick box ticked

*if **ticked**, plot the box around the plotting area (viewport).*

Draw Frame border tick box not ticked

*if **ticked**, plot the box around the frame.*

Create button

create the plot frame and then put up the plot frame edit panel for the created plot frame.

Same as button

after picking the **Same as** *button, an existing plot frame is selected and its information is used to fill in the above panel fields.*

Margin tab

Left/right margin (mm) 10.0

the left/right hand side margin between the sheet and the plotting area inside the sheet.

Bottom/top margin (mm) 10.0

the bottom/top margin between the sheet and the plotting area inside the sheet.

How To Use the Option

The plot frame contains all the information required to define the physical area in real world units to be plotted, the position on the sheet of the area being plotted, the plotter type and the plot file name.

Once the frame is created, the **New Plot Frame Edit** panel for the created plot frame is automatically placed on the screen so that the frame's position can be adjusted using the **Translate** and **Rotate** buttons.

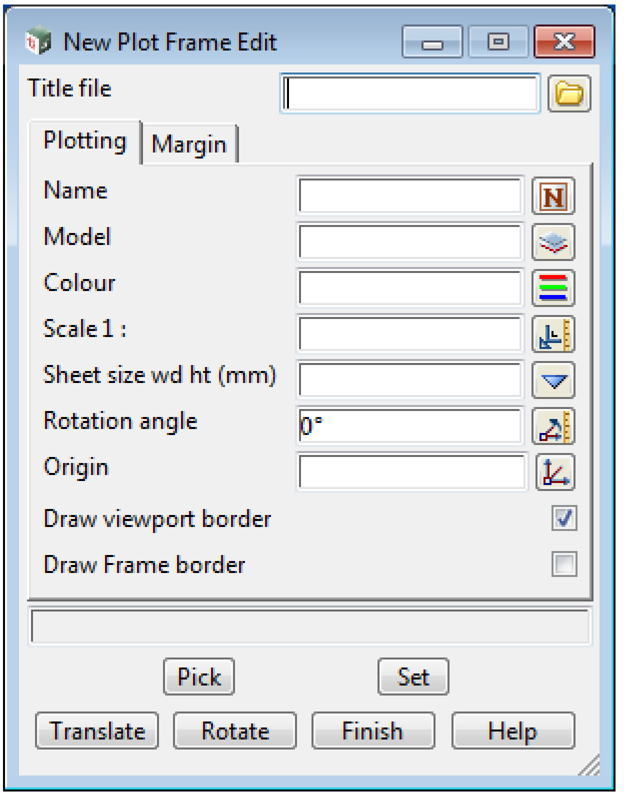
Go to the next section 25.5.2 Plot Frame Editor or return to 25.5 Plot Frames.

25.5.2 Plot Frame Editor

Position of option on menu: **Plot =>Plot frames =>Editor**

The plot frame editor is used to edit existing plot frames. It is automatically placed on the screen when a plot frame is created so that the frame's position can be adjusted using the **Translate** and **Rotate** buttons.

On selecting the **Editor** option, the **New Plot Frame Edit** panel is displayed.



Most of the fields and buttons used in this panel are the same as the **New Plot Frame Create** panel.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Title file	title file box		all title files

name for the title file to use when plotting the plot frame. The title block file can contain default information for the rest of the panel fields. See 26.2.6.5 Values Used for Defaults when Creating Plot Frames

Name	name box		available names

name for the plot frame.

Model	model box		available models

the model for the plot frame.

Colour	colour box		available colours

the colour for the plot frame.

Scale 1:	measure box		available measures

scale for the plot.

Sheet size wd ht (mm) choice box available sheet sizes

the width and height values (separated by spaces) or the name of a user defined sheet size.

Rotation angle measure box 0 degrees available measures

angle of rotation for the plot frame.

Origin input **xyz ops** menu

x_origin y_origin z_origin

origin (in world units) for the corner of the plot frame - given as three values separated by spaces. The values can either be typed in, or selected by clicking LB in the **origin** *panel view and getting up the* **xyz ops** *menu and selecting the* **pick xyz** *option.*

Draw viewport border tick box ticked

if ***ticked****, plot the box around the plotting area (viewport).*

Draw Frame border tick box not ticked

if ***ticked****, plot the box around the frame.*

Pick button

select the plot frame to be edited.

Set button

update the plot frame with the information in the above panel fields.

Translate button

after picking the **Translate** *button, the plot frame will move with the cursor until a point is selected to give the final position of the plot frame.*

Rotate button

after picking the **Rotate** *button, the plot frame is further rotated around the left hand corner of the sheet, by the angle made between the rotation point and the current cursor position.The plot frame will continue to rotate until a point is selected to fix the rotation angle.*

Margin tab

Left/right margin (mm) 10.0

the left/right hand side margin between the sheet and the plotting area inside the sheet.

Bottom/top margin (mm) 10.0

the bottom/top margin between the sheet and the plotting area inside the sheet.

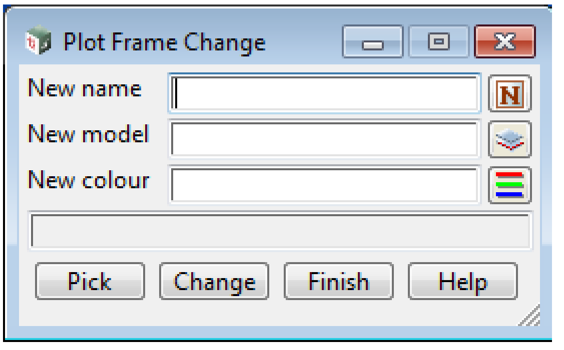
Go to the next section 25.5.3 Plot Frame Change or return to 25.5 Plot Frames.

25.5.3 Plot Frame Change

Position of option on menu: **Plot =>Plot frames =>Change**

The change option is used to modify a plot frame’s name, model or colour.

On selecting the **change** option, the **Plot Frame Change** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
New name	name box		available names

if non-blank, then the name of the selected plot frame will be changed to the name given in the **new name** *field.*

New model	model box		available models

if non-blank, then the selected plot frame will be moved to the model given in the **new model** *field.*

New colour	colour box		available colours

if non-blank, then the colour of the selected plot frame will be changed to the colour given in the **new colour** *field.*

Pick	button		

select the plot frame to be changed.

Change	button		

update the plot frame with the information in the above panel fields.

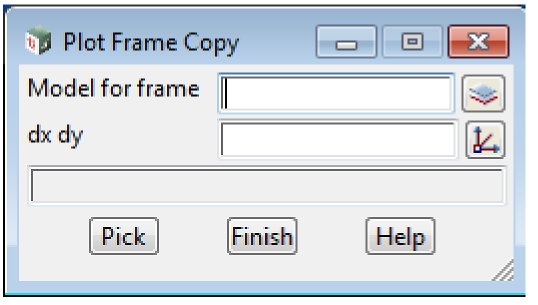
Go to the next section 25.5.4 Plot Frame Copy or return to 25.5 Plot Frames.

25.5.4 Plot Frame Copy

Position of option on menu: **Plot =>Plot frames =>Copy**

The **copy** option is used to make a copy of a plot frame.

On selecting the **Copy** option, the **Plot Frame Copy** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model for frame	model box		available models

*if **blank**, then the model of the copied plot frame will be the same as that of the picked frame.*
*if **non-blank**, then the copied plot frame will be placed in the model given in the* **model for frame** *field.*

dx dy	input		**xyz ops** menu

dx_val   dy _val
translation in world units to apply to the copied plot frame - given as two values separated by spaces.

Pick	button		

select the plot frame to be copied - the copy will be made as soon as the selected plot frame is accepted.

Go to the next section 25.5.5 Plot Frame Plot or return to 25.5 Plot Frames.

25.5.5 Plot Frame Plot

Position of option on menu: **Plot =>Plot frames =>Plot**

The **Plot** option is an alternative way of bringing up the **Plot Frame PPF Editor** panel.

See 26.9 Plot Frame and PPF Editor.

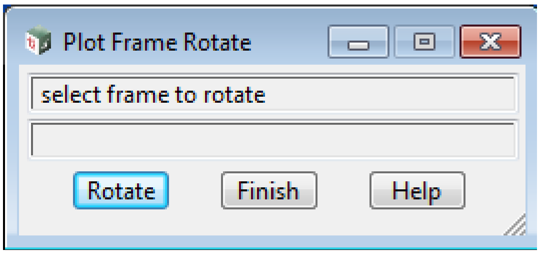
Go to the next section 25.5.6 Plot Frame Rotate or return to 25.5 Plot Frames.

25.5.6 Plot Frame Rotate

Position of option on menu: **Plot =>Plot frames =>Rotate**

The **Rotate** option is used to rotate a plot frame about a selected point.

On selecting the **Rotate** option, the **Plot Frame Rotate** panel is displayed.



The buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Rotate	button		

after picking the **Rotate** *button, the user is first asked to select a plot frame to rotate, and then to selected a point to rotate about. Once the rotation point has been selected, the plot frame is further rotated by the angle made between the rotation point and the current cursor position.The plot frame will continue to rotate until a point is selected to fix the rotation angle.*

The option then repeats.

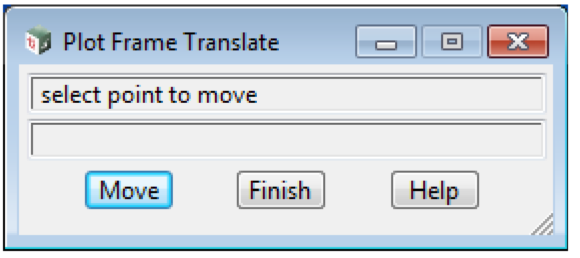
Go to the next section 25.5.7 Plot Frame Translate or return to 25.5 Plot Frames.

25.5.7 Plot Frame Translate

Position of option on menu: **Plot =>Plot frames =>Translate**

The **Translate** option is used to translate a plot frame.

On selecting the **Translate** option, the **Plot Frame Translate** panel is displayed.



The button used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Move	button		

after picking the **Move** *button, the user is first asked to select a plot frame to translate. Once the plot frame has been selected, the plot frame will move with the cursor until a point is selected to give the final position of the plot frame.*

The option then repeats.

Go to the next section 25.5.8 Plot Frame Delete or return to 25.5 Plot Frames.

25.5.8 Plot Frame Delete

Position of option on menu: **Plot =>Plot frames =>Delete**

The **Delete** option just gets up the standard **String Delete** panel which can be used to delete plot frames as well as strings. See 14.19 Delete

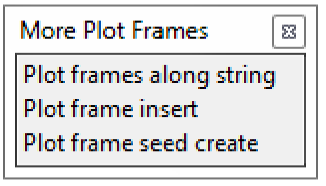
Go to the next section 25.5.9 More Plot Frames or return to 25.5 Plot Frames.

25.5.9 More Plot Frames

Position of menu: **Plot =>Plot frames =>More**

This menu contains options to create plot frame seeds, insert plot frame seeds into a view and create plot frames along a string.

The **More Plot Frames** walk-right menu is



insert a plot frame seed
create plot frames along a string
create a plot frame seed

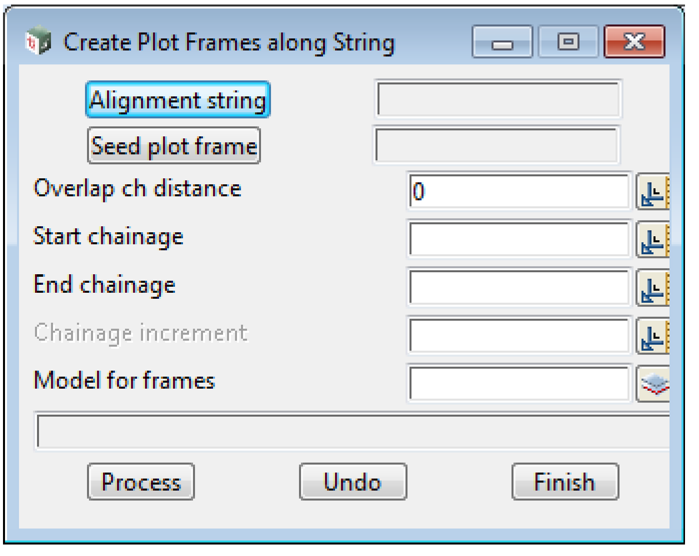
Each option in this menu will now be described.

For the option *Plot frames along string*, go to 25.5.9.1 Create Plot Frames along String
Plot frame insert 25.5.9.2 Insert Plot Frame
Plot frame seed create 25.5.9.3 Create Plot Frame Seed

25.5.9.1 Create Plot Frames along String

Position of option on menu: **Plot =>Plot frames =>More =>Plot frames along string**

On selecting the **plot frames along string** option, the **create plot frames along string** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Alignment string string select

string to create plot frame along.

Seed plot frame plot frame select

plot frame to create along the selected string.

Overlap chainage distance input 0

if non-zero, the next plot frame is started so that it includes this chainage distance of the selected string from previous plot frame.

Start/End chainage input 0

start/end chainage to restrict the amount of the selected string to place plot frames along.

Chainage increment input

if non-blank, the start of the next plot frame is placed this chainage distance along the string from the start of the previous plot frame. Used when you want to match the plot frames with the chainage interval on a long section plot.

If blank, the next plot frame is placed at the chainage point at the end of the previous plot frame (minus the **overlap chainage distance***).*

Model for frames model box available models

model to place the created plot frames in.

Process button

create the plots frames along the selected string.

Undo button

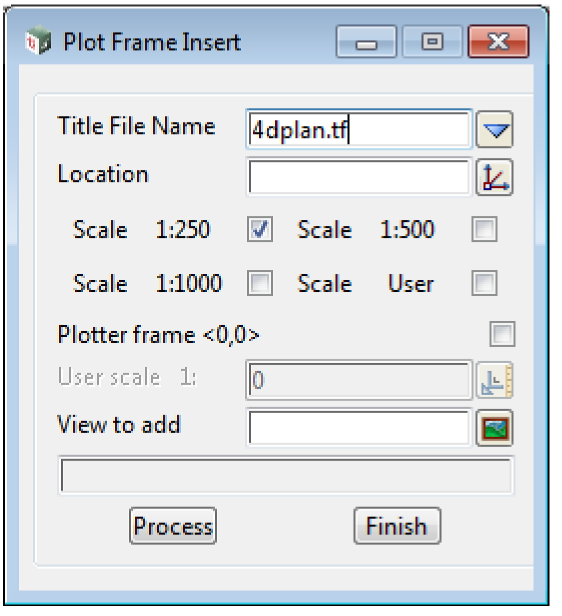
undo the last set of plot frames created whilst this panel is up.

Go to the next section 25.5.9.2 Insert Plot Frame or return to 25.5.9 More Plot Frames or 25.5 Plot Frames.

25.5.9.2 Insert Plot Frame

Position of option on menu: **Plot =>Plot frames =>More =>Plot frame insert**

On selecting the **Plot frame insert** option, the **Plot Frame Insert** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Plot frame to insert			plot frame seeds

plot frame from the seed library to insert.

Location	xyz box		xyz ops menu

the bottom left hand corner of the plot frame in inserted at this location.

Scale	radio buttons	1: 250	1: 250, 1: 500, 1:1000, User

scale to use for the inserted plot frame.

Plotter frame <0,0>	tick box	not ticked	

if ticked, the bottom left hand corner of the plot frame is placed at (0,0).

User scale 1:	measure box		

*if **User** is selected for **Scale**, then the required scale is entered here.*

View to add	view box		available views

if non-blank, add to plot frame to this view.

Process	button		

insert the selected seed plot frame at the given location at the given scale.

How To Use the Option

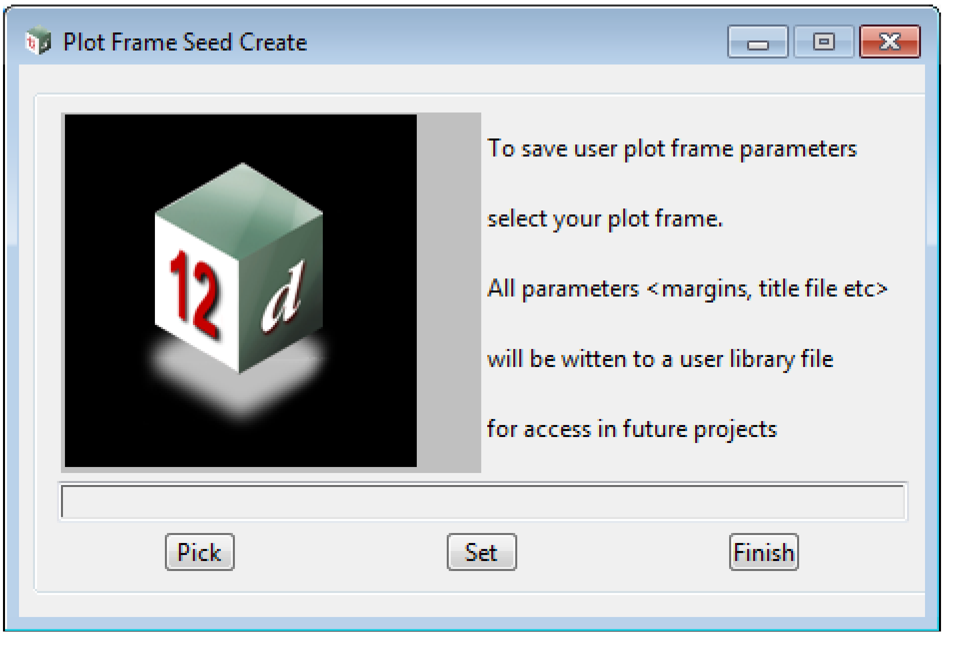
The selected plot frame is inserted at the given location with the given scale.

Go to the next section 25.5.9.3 Create Plot Frame Seed or return to 25.5.9 More Plot Frames or 25.5 Plot Frames.

25.5.9.3 Create Plot Frame Seed

Position of option on menu: **Plot =>Plot frames =>More =>Plot frame seed create**

On selecting the **Plot frame seed create** option, the **Plot Frame Seed Create** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick	string select		

pick the plot frame to add to the plot frame seed library.

Set	button		

add the selected plot frame to the plot frame seed library.

How To Use the Option

The plot frame contains all the information required to define the physical area in real world units to be plotted, the position on the sheet of the area being plotted, the plotter type and the plot file name.

The **Plot Frame Seed Create** panel adds selected plot frames to a library of plot frame seeds.

The seeds from the library can be inserted in a plan view using the **Insert plot frame seed** option.

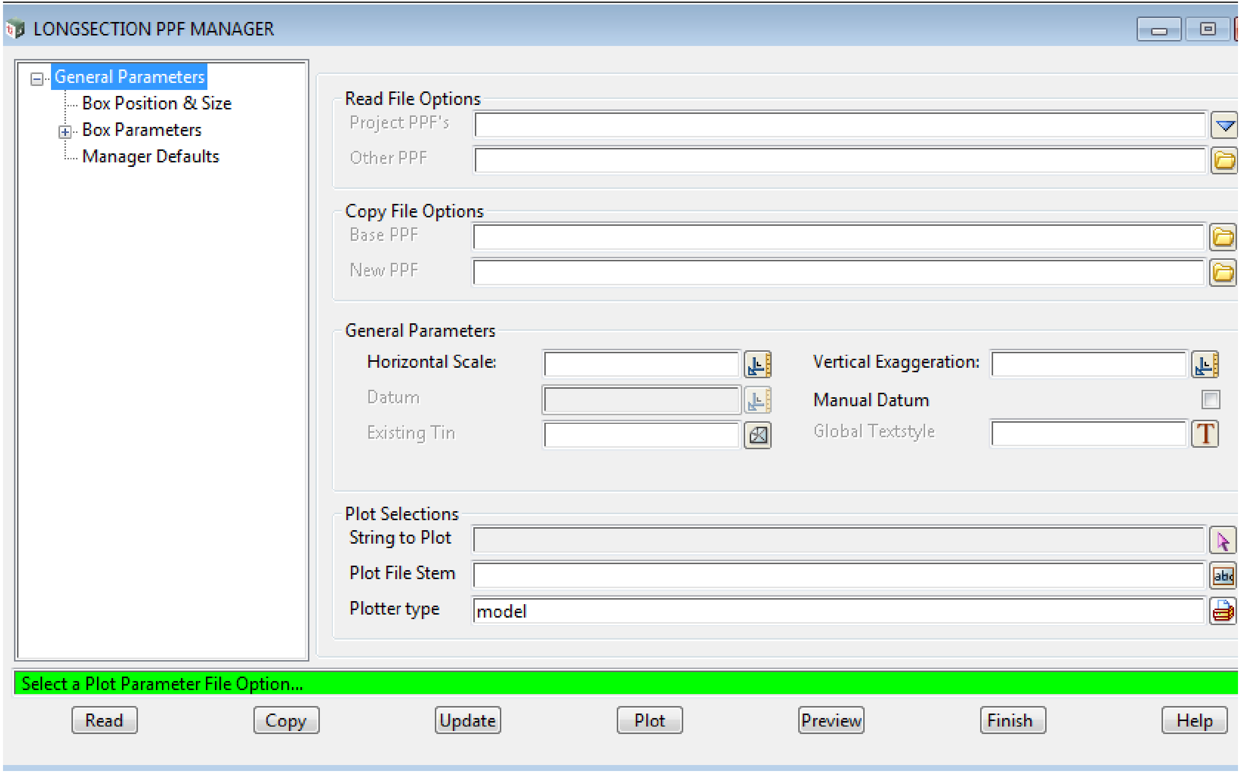
Return to 25.5.9 More Plot Frames or 25.5 Plot Frames.

25.6 Long Plot Manager

Position on menu: **Plot =>Long plot manager**

This section of documentation is a work in progress and will be updated in subsequent releases.

Selecting **Long plot manager** brings up the **Longsection PPF Manager** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
General Parameters			
Read File Options			
Project PPFs	choice box		
Other PPF	file box		
Copy File Options			
Base PPF	file box		
New PPF	file box		
General Parameters			
Horizontal Scale	measure box		
Vertical Exaggeration	measure box		
Datum	measure box		
Manual Datum	tick box	not ticked	
Existing Tin	tin box		

Global Textstyle	textstyle box
Plot Selections	
String to Plot	select
Plot File Stem	input
Plotter type	plotter box

General Parameters - Box Position & Size

Position Choice

Def Box Size

Plot Size

Active

Title 1

Title 2

Box Parameters

Global Title and Values Text Parameters

Title Text Textstyle	textstyle box
Title Text Colour	colour box
Title Text size (mm)	measure box
Values Text Textstyle	textstyle box
Values Text Colour	colour box
Values Text size (mm)	measure box
Values Decimal places	

Datum/Box Parameters

Datum Name	name box
Datum Textstyle	textstyle box
Datum Colour	colour box
Datum Text size (mm)	measure box
Datum Decimal places	
Box Colour	colour box
Uprights Colour	colour box

Box Parameters - Chainage & Design

Chainages - Title Parameters

Title line 1	name box
Title line 2	name box
Textstyle	textstyle box
Colour	colour box
Text size (mm)	measure box

Chainages - Value Parameters

Decimal places

Textstyle textstyle box

Colour colour box

Text size (mm) measure box

Chainage Plot Interval measure box

Design - Title parameters

Title line 1 name box

Title line 2 name box

Textstyle textstyle box

Colour colour box

Text size (mm) measure box

Design - Value Parameters

Decimal places

Textstyle textstyle box

Colour colour box

Text size (mm) measure box

Box Parameters - Natural Surface & Depths

Natural Surface Tin - Title parameters

Title line 1 name box

Title line 2 name box

Textstyle textstyle box

Colour colour box

Text size (mm) measure box

Natural Surface Tin - Value Parameters

Decimal places

Textstyle textstyle box

Colour colour box

Text size (mm) measure box

Depth Cut/Fill - Title Parameters

Title line 1 name box

Title line 2 name box

Textstyle textstyle box

Colour colour box

Text size (mm) measure box

Depth Cut/Fill - Value Parameters

Decimal places	
Textstyle	textstyle box
Colour	colour box
Text size (mm)	measure box
Tin Plot Colour	colour box

Box Parameters - Volumes Cut/Fill

Volume Titles - Title Parameters

Title line 1	name box
Textstyle	textstyle box
Colour	colour box
Text size (mm)	measure box

Volume Titles - Cut Title Parameters

Title line 1	name box
Textstyle	textstyle box
Colour	colour box
Text size (mm)	measure box

Volume Titles - Fill Title Parameters

Title line 1	name box
Textstyle	textstyle box
Colour	colour box
Text size (mm)	measure box

Volume Values - Cut Values Parameters

Textstyle	textstyle box
Colour	colour box
Text size (mm)	measure box
Decimal places	

Volume Values - Fill Values Parameters

Textstyle	textstyle box
Colour	colour box
Text size (mm)	measure box
Decimal places	
Vol Report	file box
Sample Interval	measure box

Box Parameters - Offset Strings

Offset String 1 - Title Parameters

Title line 1	name box

Title line 2	name box
Textstyle	textstyle box
Colour	colour box
Text size (mm)	measure box

Offset String 1 - Value Parameters

Decimal places	
Textstyle	textstyle box
Colour	colour box
Text size (mm)	measure box

Offset String 2 - Title Parameters

Title line 1	name box
Title line 2	name box
Textstyle	textstyle box
Colour	colour box
Text size (mm)	measure box

Offset String 2 - Value Parameters

Decimal places	
Textstyle	textstyle box
Colour	colour box
Text size (mm)	measure box
Offset String 1	select
Offset String 2	select

Box Parameters - X & Y Coords

X Coords - Title Parameters

Title line 1	name box
Title line 2	name box
Textstyle	textstyle box
Colour	colour box
Text size (mm)	measure box

X Coords - Value Parameters

Decimal places	
Textstyle	textstyle box
Colour	colour box
Text size (mm)	measure box

Y Coords - Title Parameters

Title line 1	name box

Title line 2	name box
Textstyle	textstyle box
Colour	colour box
Text size (mm)	measure box

Y Coords - Value Parameters

Decimal places	
Textstyle	textstyle box
Colour	colour box
Text size (mm)	measure box

Box Parameters - Super & Offset String Chgs

Superelevation - Title Parameters

Title line	name box
Textstyle	textstyle box
Colour	colour box
Text size (mm)	measure box

Superelevation - Selection Parameters

Super Draw Mode

X Sect Model

Super Left Hinge

Super Left Edge

Super Right Hinge

Super Right Edge

Offset String (Shift) - Title Parameters

Title line 1	name box
Title line 2	name box
Textstyle	textstyle box
Colour	colour box
Text size (mm)	measure box

Offset String (Shift) - Value Parameters

Decimal places	
Textstyle	textstyle box
Colour	colour box
Text size (mm)	measure box

Chainage Offset String	select

Manager Defaults

Position Choice

Def Box Size	
Title 1	
Title 2	
Read Def	button
Write Def	button

Buttons at bottom

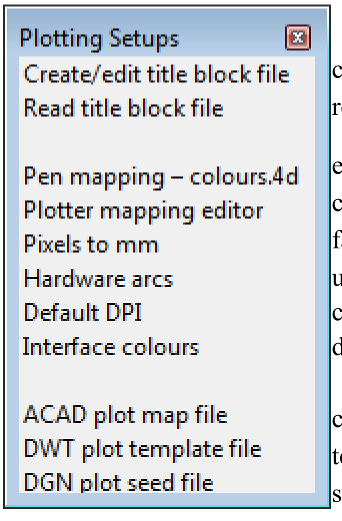
Read	button
Copy	button
Update	button
Plot	button
Preview	button

Go to the next section 25.7 Plotting Setups or return to 25 Plots.

25.7 Plotting Setups

Position of menu: **Plot =>Plotting setups**

The **Plotting setups** walk-right menu is



create/edit a title block file
read in a title block file

edit colours.4d for colours to pens
create/edit plotter mapping file
factor for pixel size to mm on plots
use/not use hardware arcs
change cut/fill colours
default DPI for raster plots

create/edit AutoCAD map file
template file for writing .dwt
seed file for writing .dgn

For the options, go to

25.7.1 Create/Edit Title Block File
25.7.2 Read Title Block

25.7.3 Pen Mapping
25.7.4 Plotter Mapping File Editor
25.7.5 Pixels to mm
25.7.6 Hardware Arcs
25.7.11 DGN Plot Seed File
25.7.8 Default DPI

25.7.9 ACAD Plot Map File
25.7.10 DWT Plot Template File
25.7.11 DGN Plot Seed File

25.7.1 Create/Edit Title Block File

Position of option on menu: **Plot =>Plotting setups =>Create/edit title block file**

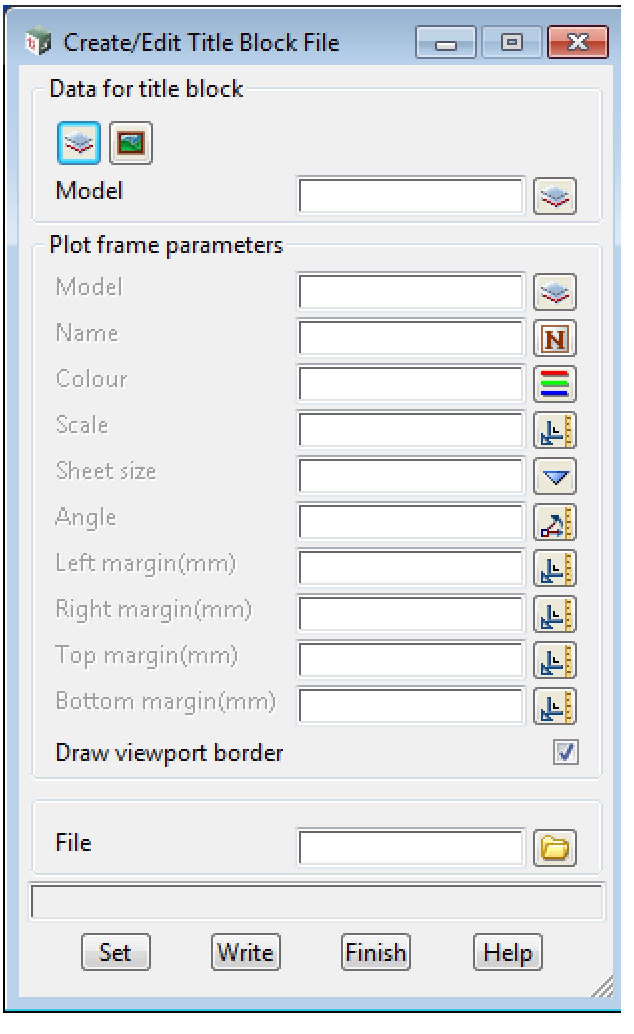
Position of option on menu: **Drafting => Create title block file**

This option is used to create a title block file from string data in a model or a view.

If it is a model, the model attributes can also contain information that is used for creating plot frames. For example, Sheet size, left/right margins, *etc.*).

The data in the model/view should be set up with (0,0) representing the left hand bottom of the plotting sheet and the units in model/view represent ***millimetres*** on the paper. So text should be defined with ***World*** units and a metre in world units is taken to be a millimetre in the title block itself. That is, a text size 5 world units will plot as 5 mm.

For more information and the format of the title block file, go to the section 26.2.6 Title Block



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Data Model/View	model/view box		available models/views

the data in this model/view is written to the title block file to be used as the linework and text in a title block.
The **Set** *button is used to create model attributes for the information in the Plot frame parameters*

section, so if the **Set** *button is to be used, then only a* **Data Model** *can be used here.*

Plot frame parameters

*these parameters are written to an info block in the title block file and are only used when defining a plot frame using the option **Plot =>Plot frames =>Create**. The parameters can also be written as attributes in the* **Data Model** *using the* **Set** *button.*

Name

if not blank, the name to use for the plot frame.

Model model box available models

if not blank, model for the plot frame.

Colour colour box available colours

if not blank, colour for the plot frame.

Scale 1: input

if not blank, scale for the plot frame

Sheet size input available sheet sizes

if not blank, sheet size for the plot frame

Angle angle box

if not blank, angle of the plot frame - measured anticlockwise from the positive x-axis

Left margin, Right margin, Top margin, Bottom margin

margins for the plot frame

Draw viewport border tick box tick

if ticked, a border is drawn around the view port in the colour of the plot frame.

Title block file file box *.tbf files

name of the file to contain the title block information.

Set button

save the plot frame parameters as attributes in the Data model.

Write button

run the option – create the title block file.

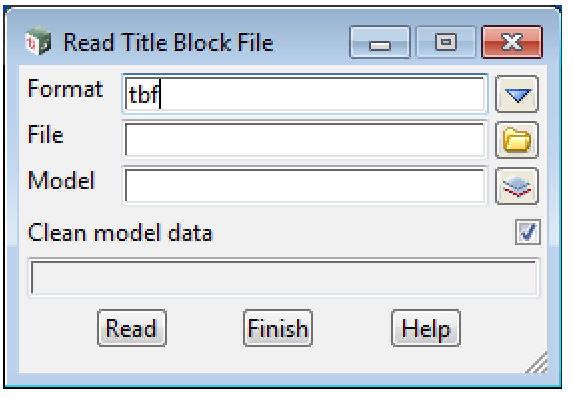
Go to the next section 25.7.2 Read Title Block or return to 25.7 Plotting Setups or 25 Plots.

25.7.2 Read Title Block

Position of option on menu: **Plot =>Plotting setups =>Read title block file**

The **Read title block file** option is for reading in a title block file and placing the lines, arcs and text defined in the title block file into a given model.

On selecting the **Read title block file** option, the **Read Title Block File** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Format	choice box		tbf, tf

specify if the format of the file is the V7 title block file (.tbf) or the older format (.tf).

File	file box		

name of title block file to read in.

Model	model box		available models

model to place the title block in when it is read. Any plot frame parameters in the info block of the title block file are set as attributes for this model. Hence these parameters can be written out again to a new title block file if the model in used as the Data Model when creating a title block file.

Clean model data	tick box	not ticked	

if ticked, the data in the model given in the **Model** *field will be cleaned prior to reading in the title block file.*

If not ticked, the data in the model given in the **Model** *field will not be cleaned prior to reading in the title block file. The data is appended to the existing model data.*

Read	button		

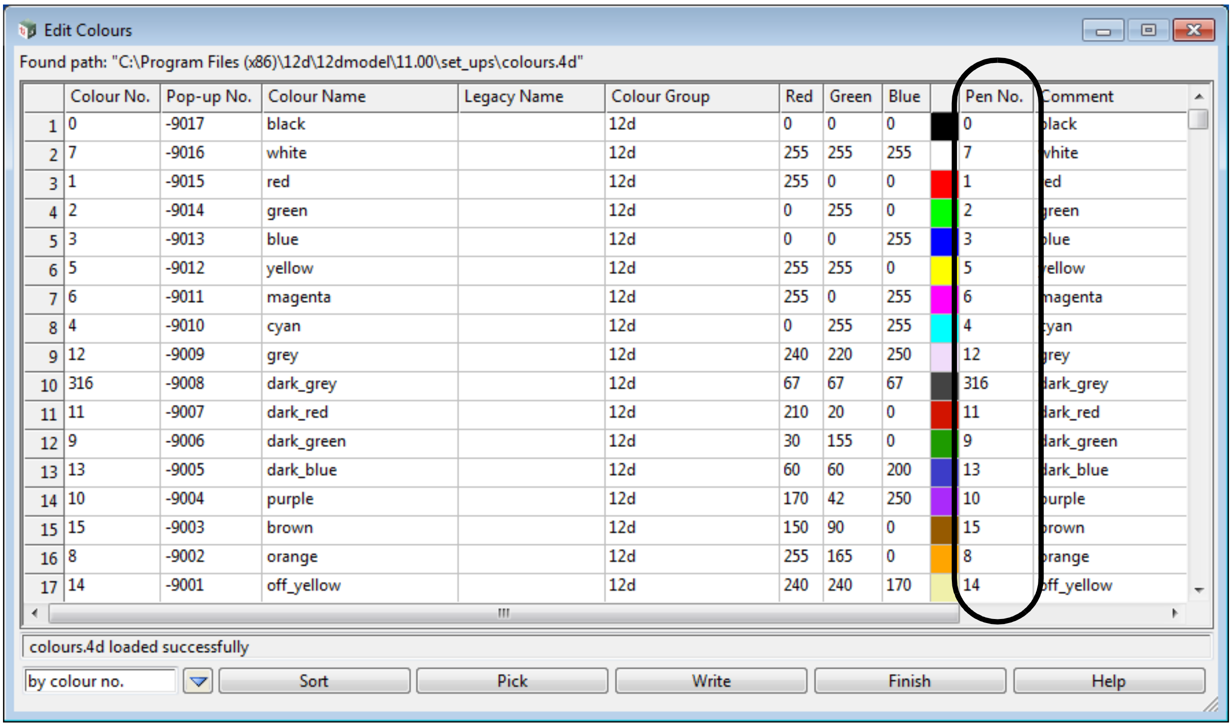
read in the title block file and place the information in the given model.

Go to the next section 25.7.3 Pen Mapping or return to 25.7 Plotting Setups or 25 Plots.

25.7.3 Pen Mapping

This option brings up the **Edit Colours** panel which has a column **Pen No.** that is used as the default pen number that a colour is mapped to.

The individual **colour-to-pen numbers** may be overridden by a **pen mapping file**.


Edit Colours

Found path: "C:\Program Files (x86)\12d\12dmodel\11.00\set_ups\colours.4d"

	Colour No.	Pop-up No.	Colour Name	Legacy Name	Colour Group	Red	Green	Blue	Pen No.	Comment
1	0	-9017	black		12d	0	0	0	0	lack
2	7	-9016	white		12d	255	255	255	7	vhite
3	1	-9015	red		12d	255	0	0	1	ed
4	2	-9014	green		12d	0	255	0	2	reen
5	3	-9013	blue		12d	0	0	255	3	lue
6	5	-9012	yellow		12d	255	255	0	5	ellow
7	6	-9011	magenta		12d	255	0	255	6	nagenta
8	4	-9010	cyan		12d	0	255	255	4	yan
9	12	-9009	grey		12d	240	220	250	12	rey
10	316	-9008	dark_grey		12d	67	67	67	316	lark_grey
11	11	-9007	dark_red		12d	210	20	0	11	lark_red
12	9	-9006	dark_green		12d	30	155	0	9	lark_green
13	13	-9005	dark_blue		12d	60	60	200	13	lark_blue
14	10	-9004	purple		12d	170	42	250	10	urple
15	15	-9003	brown		12d	150	90	0	15	rown
16	8	-9002	orange		12d	255	165	0	8	range
17	14	-9001	off_yellow		12d	240	240	170	14	ff_yellow

colours.4d loaded successfully

by colour no. Sort Pick Write Finish Help

For more information on this option, go to the section 4.22 Colours.

Go to the next section 25.7.4 Plotter Mapping File Editor or return to 25.7 Plotting Setups or 25 Plots.

25.7.4 Plotter Mapping File Editor

Position of option on menu: **Plot =>Plotting Setups =>Plotter mapping file editor**

The **Plotter Mapping Editor** is used to create a ***plotter mapping file*** (pmf file) which is used when making a plot from ***12d Model***. The *plotter mapping file* defines:

(a) the mapping from ***12d Model*** colours to the plotter pens

(b) the rgb (colours) and weights for the plotter pens

When defining Plotters Different *plotter mapping files* are used for different plotter configurations.

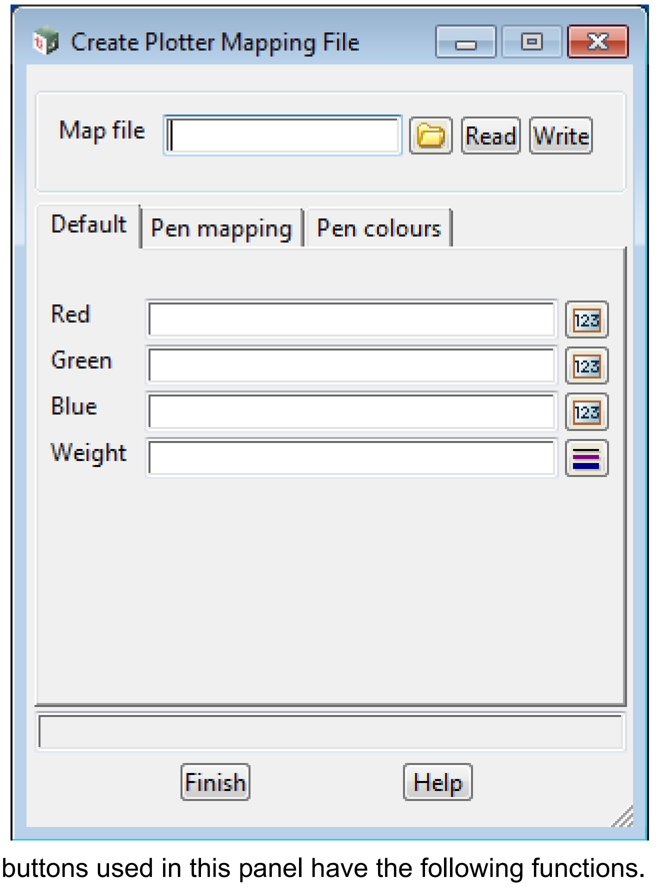
For example, different plotter mapping files would be used when plotting the same plot in ***12d Model*** to a mono-colour plotter or to a colour plotter.

For plotting to a mono-colour plotter, ***12d Model*** colours would be mapped to the one black pen but with possible different weights for different colours. So only one pen colour would be needed and its colour would be set to black (Red 0, Green 0 and Blue 0).

For plotting to a colour plotter, ***12d Model*** colours could be mapped to different colours and weights on the plotter.

The text definition of the plotter mapping file is given in the next section 25.7.5 Pixels to mm.

On selecting the **Plotter mapping editor** option, the **Create Plotter Mapping File** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Map file	file box		*.pmf file

the plotter mapping file to be created/edited.

Read button

read in the plotter mapping file given in **Map file**

Write button

write out to plotter mapping information in the panel to the file given in **Map file**.

Default tab

Red/Green/Blue input

the value between 0 and 255 of red/green/blue to be used for the default pen colour.

Weight input

the default pen weight.

12d Colour mapping tab

12d Colour colour box

the 12d colour to define the pen mapping for.

Use Pen Number integer box

the number of the pen on the plotter for the 12d colour to be mapped to.

Weight real number box

the weight to use for the plotter pen for the given 12d colour.

Define Pen Colours tab

Pen Number integer box

the number of the plotter pen to define the colour for.

Red/Green/Blue input

the value between 0 and 255 of red/green/blue to be used for the colour of the plotter pen.

Go to the next section 25.7.5 Pixels to mm or return to 25.7 Plotting Setups or 25 Plots.

25.7.5 Pixels to mm

Position of option on menu: **Plot =>Plotting setups =>Pixels to mm**

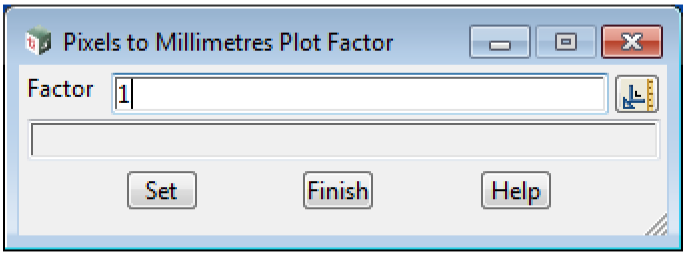
To plot an object, it must be possible to calculate the size in millimetres that the object will have in the plot.

However, in ***12d Model*** it is possible for text and linestyles to be defined in the screen unit pixels, and since the size of a pixel varies between screens, there is no direct millimetre equivalent for a pixel.

To allow text and linestyles with only a pixel size to be plotted, a factor to convert pixels to millimetres is used.

When plotting, all text and linestyles defined only in pixel units are **multiplied** by the pixels-to-millimetre factor to determine its size in the plot.

On selecting the **Pixel to mm** option, the **Pixels to Millimetres Plot Factor** panel is displayed.



To define a new factor, simply type the value into the **Factor** field and select **Set**.

The pixel-to-millimetres value is stored for the project.

For a new project, the initial value is loaded from the **Defaults** file and is defined in the file by

PIXELS TO MM PLOT FACTOR *value*

For information on the Defaults file, see 39.2.7.3 Defaults File (defaults.4d).

Go to the next section 25.7.6 Hardware Arcs or return to 25.7 Plotting Setups or 25 Plots.

25.7.6 Hardware Arcs

Position of option on menu: **Plot =>Plotting setups =>Hardware arcs**

When plotting arcs, the arc can either be broken into a series of straight lines within ***12d Model***, or plotted using the arc command (if it exists) for the relevant plotter.

The initial default for hardware arcs is set by the environment variable HARDWARE_ARCS_4D. but this can be overridden by the option **Plots =>Plotter setups =>Hardware arcs.**

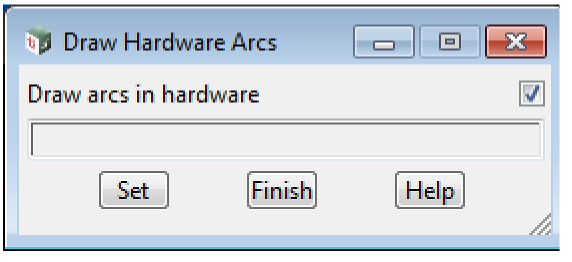
This option allows the user to choose whether

(a) the plotter arc command is used (hardware arcs)

or

(b) before plotting, the arc is broken into a series of lines.

On selecting the **hardware arcs** option, the **Draw Hardware Arcs** panel is displayed.



To use hardware arcs in plots, simply change the **Draw arcs in hardware** tick box to **tick** and select **Set**.

Go to the next section 25.7.7 Interface Colours or return to 25.7 Plotting Setups or 25 Plots.

25.7.7 Interface Colours

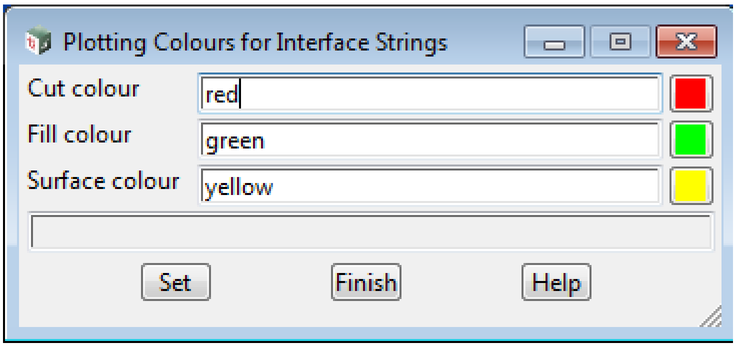
Position of option on menu: **Plot =>Plotting setups =>Interface colours**

When drawing interface strings in ***12d Model,*** cut areas are denoted in red, fill areas in green and sections on the surface in yellow (and yellow if the section became undefined).

However, when plotting it is convenient to map the interface colours to other colours.

This panel is used to define new colours for the interface colours, and the new colours are then mapped to pens using the pen mapping table or plotter mapping file.

On selecting the **interface colours** option, the **plotting colours for interface string** panel is displayed.



To define new colours, simply type the values into the appropriate panel fields and select **Set**.

Note - the new interface colours are not used for displaying interface strings in ***12d Model***, only for plotting.

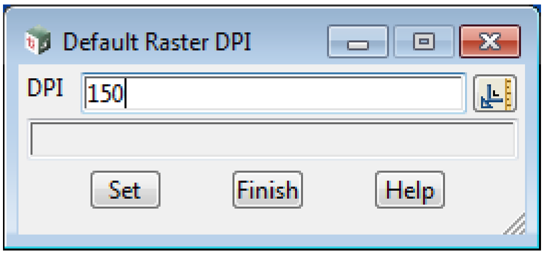
Go to the next section 25.7.8 Default DPI or return to 25.7 Plotting Setups or 25 Plots.

25.7.8 Default DPI

Position of option on menu: **Plot =>Plotting setups => Default DPI**

The *DPI* value give the default dots per inch that is used for plotting rasters.

On selecting **Default DPI**, the **Default Raster DPI** panel is displayed.



To define a new Default DPI, simply type the value into the **DPI** panel fields and select **Set**.

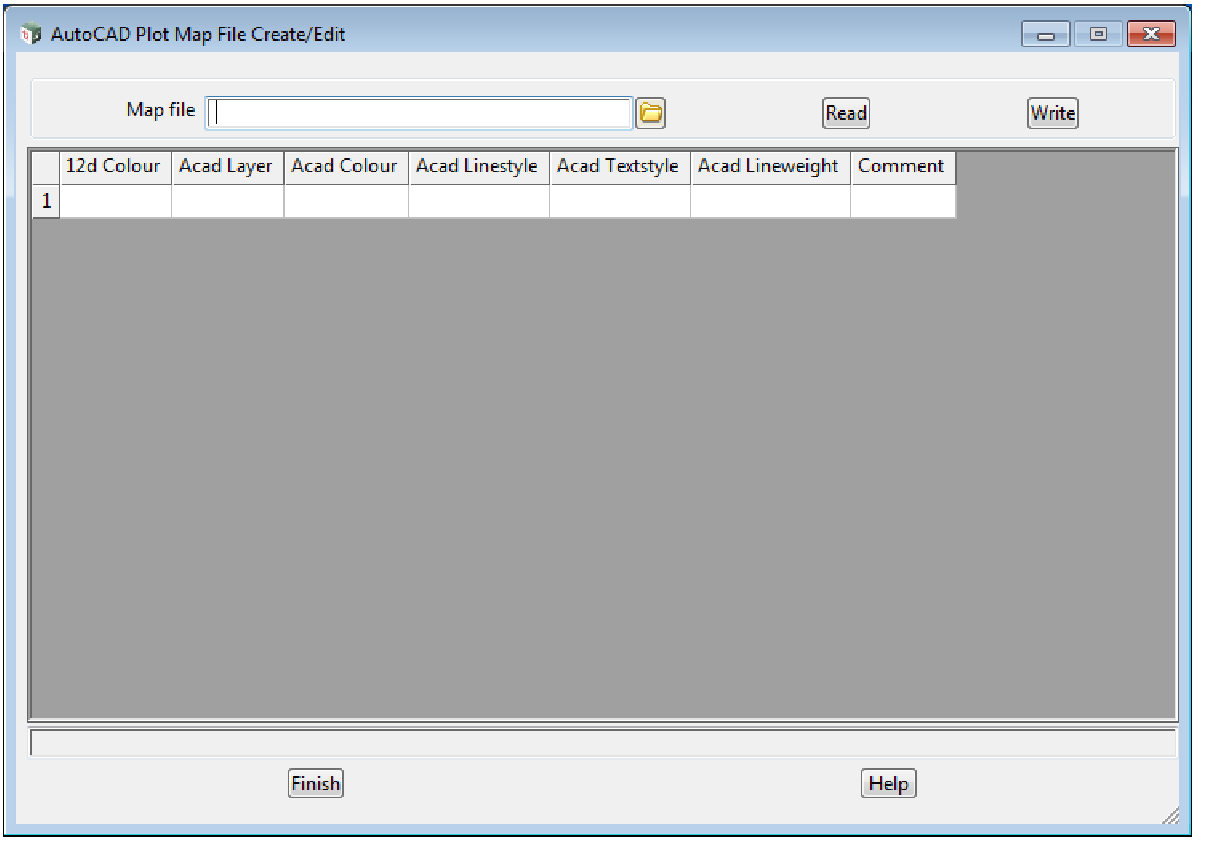
Go to the next section 25.7.9 ACAD Plot Map File or return to 25.7 Plotting Setups or 25 Plots.

25.7.9 ACAD Plot Map File

Position of option on menu: **Plot =>Plotting setups =>ACAD plot map file**

When plotting to AutoCAD, an Autocad map file can be used which uses the ***12d Model*** colour of an entity in the plot as a key to mapping the entity into AutoCAD.

On selecting the **ACAD plot map file** option, the **AutoCAD Map File Create/Edit** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Map file	file box		*.amf files

name for the file to save/read in an AutoCAD mapping file.

Read button

read in the amf file given in the **Map file** *field and load the information into the panel.*

Write button

write out the current information defined in the panel to an amf file with the name given in the **Map file** *field.*

Grid

the information for each line of the amf file. The definition of the amf file is given in 43.2.1.2 Definition and Example of a .tbl File

12d Colour colour box

*the 12d colour to match against. This can include wild cards * and wild characters ?.*

Acad Layer text box

the name of the AutoCAD layer to map the data to.

Acad Colour text box

the AutoCAD colour. This can only be a number between 0 and 256 or BYLAYER.

This can be ***BYLAYER*** *which will be for AutoCAD BYLAYER.*

Use ***** *for the 12d Model colour.*

Acad linestyle text box

the AutoCAD line type.

This can be ***BYLAYER*** *which will be for AutoCAD BYLAYER.*

Use ***** *for the 12d Model linestyle.*

Acad Textstyle text box

Not currently used. The AutoCAD text style.

Use ***** *for the 12d Model text style.*

Comment text box

comment about this line of data.

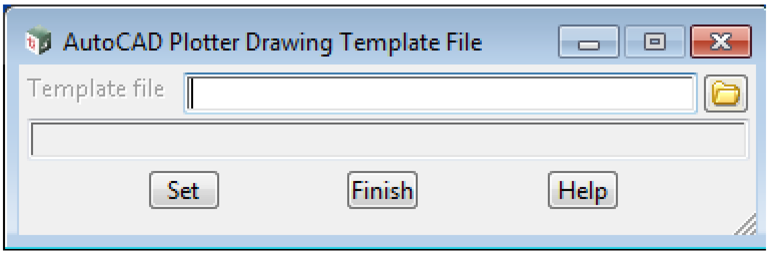
Go to the next section 25.7.10 DWT Plot Template File or return to 25.7 Plotting Setups or 25 Plots.

25.7.10 DWT Plot Template File

Position of option on menu: **Plot =>Plotting Setups =>DWG plot template file**

When creating plot files in AutoCAD format, an AutoCAD template file can be used.

On selecting the **DWT plot template file** option, the **AutoCAD Plotter Drawing Template File** panel is displayed.



To define an AutoCAD plot seed file to use for AutoCAD plotting, simply select either *DWG* or *DXF* for the **File format**, type the file name into the **plot seed file** panel field and select **set**.

25.7.11 DGN Plot Seed File

Position of option on menu: **Plot =>Plotting Setups =>DGN plot seed file**

When creating plot files in Microstation DGN format, a Microstation DGN seed file can be used.

A DGN seed file can be pointed to by the environment variable DGN_PLOT_SEED_FILE_4D

A folder containing the seed files can be pointed to with the environment variable MS_SEEDFILES_4D.

The Microstation seed file can also set by this option.

On selecting the **DGN plot seed file** option, the **DGN Plot Seed File** panel is displayed.

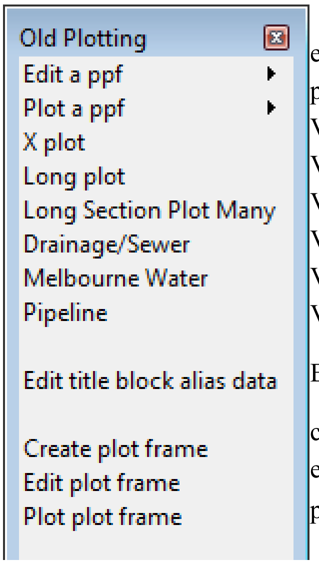
To define a Microstation IGDS plot seed file to use for Microstation plotting, simply type the file name into the **Plot seed file** panel field and select **Set**.

Return to 25.7 Plotting Setups or 25 Plots.

25.8 Old Plotting

Position of menu: **Plot =>Old plotting**

The **Old plotting** walk-right menu is



edit a V5 text PPF
plot a V5 text PPF
V5 x-section plotting
V5 long section plotting
V5 plotting many long section plotting
V5 drainage long section plot
V5 Melbourne Water sewer long section plot
V5 pipeline long section plot

Edit title block alias data for V5 PPFs

create a V5 plot frame
edit a V5 plot frame
plot a V5 plot frame

For the option *Edit a ppf*, go to	25.8.1 Edit a ppf
Plot a ppf	25.8.2 Plot a ppf
X plot	25.8.3 X Plot
Long plot	25.8.4 Long Plot
Long section plot many	25.8.5 Plot Many Long Sections
Drainage/sewer	22.9.2 Drainage Longsections
Melbourne Water	22.9.3 Melbourne Water.
Pipeline	22.5 Plots.
Edit title block alias data	25.8.9 Edit Title Block Alias Data
Create plot frame	25.8.10 Create Old Plot Frame
Edit plot frame	25.8.11 Editor Old Pot Frame
Plot plot frames	25.8.12 Plotting Old Plot Frames

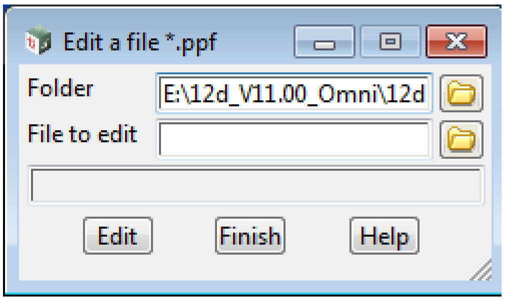
25.8.1 Edit a ppf

Position of option on menu: **Plot =>Old plotting =>Edit a ppf**

The **Edit a ppf** option is used to edit V5 text plot parameter files (*.ppf) with the editor pointed to by the EDITOR_4D environment variable.

The **Edit a ppf** walk-right menu provides a list all the plot parameter files (files ending in .ppf) in the current folder.

When a file is selected from the list, it is loaded into the editor pointer to by the environment variable EDITOR_4D.



25.8.2 Plot a ppf

Position of option on menu: **Plot =>Old plotting =>Plot a ppf**

The V5 text PPFs for long section, cross section and plot frames can have enough information in them to totally generate the plot from the ppf. The **plot a ppf** option is used to plot such a plot parameter file (*.ppf).

The **Plot a ppf** walk-right menu provides a list all the plot parameter files (files ending in .ppf) in the current folder.

When a file is selected from the list, it is then used to generate a plot.

Note - the plot parameter file is defined in 44 Text Plot Parameters.

25.8.3 X Plot

Position of option on menu: **Plot =>Old plotting =>X plot**

The **X plot** option is used to make the traditional pages of cross-section plots, that is, stacked x-sections, with offsets and elevations labelled for each cross section string, and elevations displayed for each of the tins on the section view.

For ***12d Model*** V5, each string to be plotted is assumed to be on a straight line in plan. Hence the cross section is taken as a straight line from the first point of the string to the last point of the string. The chainage of the first point of the string is used as the offset.

For ***12d Model*** V6 and above, sections do not have to be a straight line in plan. The chainage of the first point of the string is used as the offset.

The format of the x-section plot is partially controlled by fields in the **Section X Plot** panel but more extensive control is possible using a **plot parameter file** (*.ppf). The plot parameter file is fully documented in the 44.2 Cross Section Plot Parameter File section of the Appendix, 44 Text Plot Parameters.

The cross section plot is tailored by using the plot parameter file (.ppf file) given in the **plot parameters** field. A default .ppf file is set by pointing to it with the environment variable

X_SECTION_PPF_4D pathname of default .ppf file

In the pathname to the default .ppf file, $LIB is used to stand for the library folder set by LIB_4D. For example, "$LIB/cross.ppf" is the file cross.ppf in the library area.

Which x-sections are to be plotted is specified by giving the model containing the appropriate cross sections. Sections are also drawn through any triangulations and service items on the section view. Vertical exaggeration, services and corridor settings are taken from a plot parameter file or from the settings for the section view.

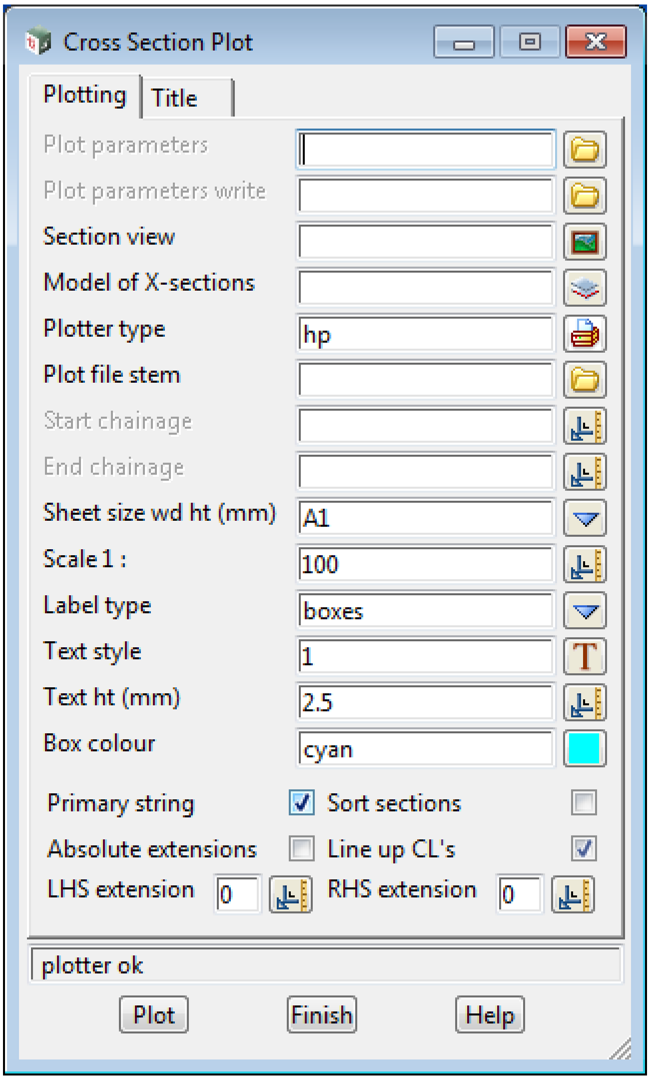
X-sections can be labelled with either the offset from the centreline and heights for each of the x-section points, or simply the position of the centre-line.

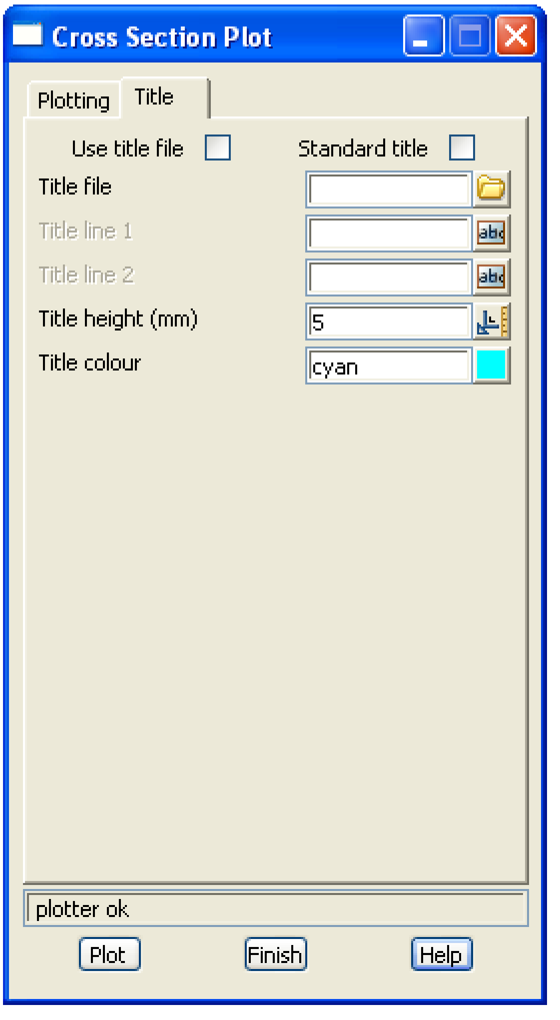
The required page size is given, plus the scale (the vertical exaggeration is taken from the view settings) and the sections are then plotted in columns on the plotter page. Once a page is full, a new plot page is automatically begun.

The stacked x-section plots in each column can be lined up by their centre-lines.

After selecting the **X plot** option, the **Cross Section Plot** panel is displayed.

The **Title** button on the bottom of this panel controls the use of a title block file and the plotting of a border and two lines of title. If the **title** button is selected, the **Cross Section Plot Title** panel is displayed.


Cross Section Plot
Plotting
Title
Plot parameters
Plot parameters write
Section view
Model of X-sections
Plotter type
hp
Plot file stem
Start chainage
End chainage
Sheet size wd ht (mm)
A1
Scale 1 :
100
Label type
boxes
Text style
1
Text ht (mm)
2.5
Box colour
cyan
Primary string
Sort sections
Absolute extensions
Line up CL's
LHS extension
0
RHS extension
0
plotter ok
Plot
Finish
Help



The fields and buttons used in the first panel have the following functions.

Field Description Type Defaults Pop-Up

Plot parameters file box X_SECTION_PPF.4D available *.ppf files

file of plot parameters used for extra control of the x section plot. The default .ppf file is pointed to by the environment variable X_SECTION_PPF_4D.

For more information on the x section plot parameters, please go to the section 44.2 Cross Section Plot Parameter File *in the Appendix* 44 Text Plot Parameters

Plot parameters write file box available *.ppf files

file to write a copy of the plot parameters necessary to regenerate the plot.

Section view input/output current view available views

section view to use for plot definitions such as the tins to profile, the models of services, corridor width and vertical exaggeration.

Model of X-sections model box available models

the model containing all the x-sections to be plotted. The x-sections are plotted in the same order as they are in the model.

Each string to be plotted is assumed to be on a straight line in plan. Hence the cross section is taken as a straight line from the first point of the string to the last point of the string. The chainage of the first point of the string is used as the offset.

Plotter type	input	hp	hp, dxf, postscript etc.

file format for the plot output.

Plot file stem	input

since more that one plot page may be produced, the plot file stem plus a plot page sequence number followed by the ending for the plotter type is used as the plot file names.

Start/end chainage	input

the plots cover the chainage range for the cross sections given by the start and end chainage fields. If the **start/end chainage** *is blank, the star/end chainage of the x-section strings are used.*

Sheet size wd ht (mm)	choice box		available sheet sizes

must contain the width and height values (separated by space) or the name of a user defined sheet size.

Scale 1:	input

horizontal scale for plotting the x-sections.

Label type	choice box	boxes	boxes, centreline

if ***boxes****, the offset from the centreline and heights for each point in the x-section will be plotted in the traditional offset/height boxes.*
if ***centre line****, only the position of the centre-line and the height of the section at that point will be drawn.*

Text style (mm)	input	1

text style to be used in the x section plot boxes or centreline labelling.

Text ht (mm)	input	3

height (in millimetres) to plot the offset and heights in the boxes in the plots of the x-sections

Offset colour	colour box	red	available colours

colour used for the offset text and the boxes.

Primary string	tick box	ticked

if ***ticked****, the string from the* **model of x-sections** *(the primary string) is plotted.*

Sort sections	tick box	ticked

if ***ticked****, the strings from the* **model of x-sections** *are sorted by chainage along the design string.*

Absolute extensions	tick box	not ticked

If ***ticked****, the sections are only drawn from the centreline out to the left and right extension distances.*
if ***not ticked****, the left and right extension values are added to the section left and right widths, and the section is taken between the extended values*

Line up CL's	tick box	ticked

if ***ticked****, the centre-lines are lined up under each other for each plot in a column.*

LHS extension	input	0

distance to extend the x-section to the left.

RHS extension	input	0

distance to extend the x-section to the right.

Plot	button

write out the plots for the x-sections given in the **model of x-sections** *field.*

Title tab

panel for defining information in the title block.

The fields and buttons in *Title* tab are:

Use title file	tick box	not ticked	

*if **ticked,** a user defined title block file is used.*

Standard Title	tick box	ticked	

*if **ticked**, the standard 4D Solutions border and two lines of title are placed on the bottom of the plot*

Title file	file box		available *.tf files

*if **non-blank** and* **use title file** *is set to **tick**, then the file given in this field is used to generate a user defined title block for the plot.*

Title line 1/2	input		

first/second line of title information

Title height (mm)	input	5	

height (in millimetres) to draw the characters in the two lines of title information.

Title colour	input	cyan	available colours

colour used for the border and the title information.

25.8.4 Long Plot

Position of option on menu: **Plot =>Old plotting =>Long plot**

The **Long plot** option is used to make traditional long section plots with string chainages and elevations labelled for the primary string and elevations and depths displayed for each of the tins on the section view.

The format of the long section plot is partially controlled by fields in the section long plot panel but more extensive control is possible using a **plot parameter file** (*.ppf). The plot parameter file is fully documented in the 44.3 Long Section Plot Parameter File section.

After selecting the **Long plot**, the **Long Section Plot** panel is displayed.

The long section plot is tailored by using the plot parameter file (.ppf file) given in the **plot parameters** field. A default .ppf file is set by pointing to it with the environment variable

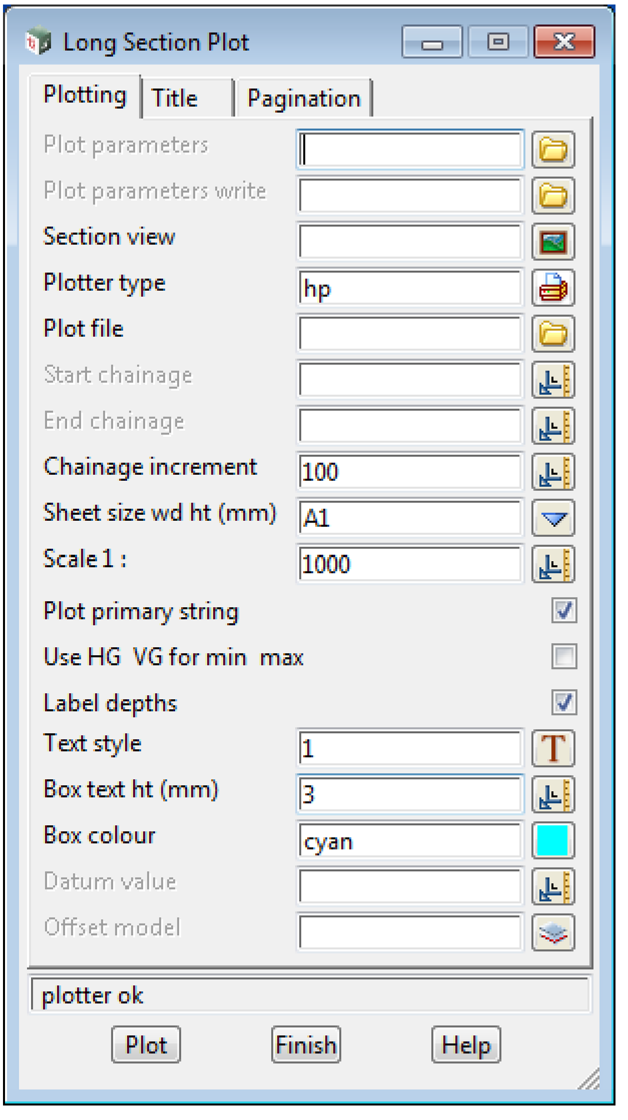
LONG_SECTION_PPF_4D pathname of default .ppf file

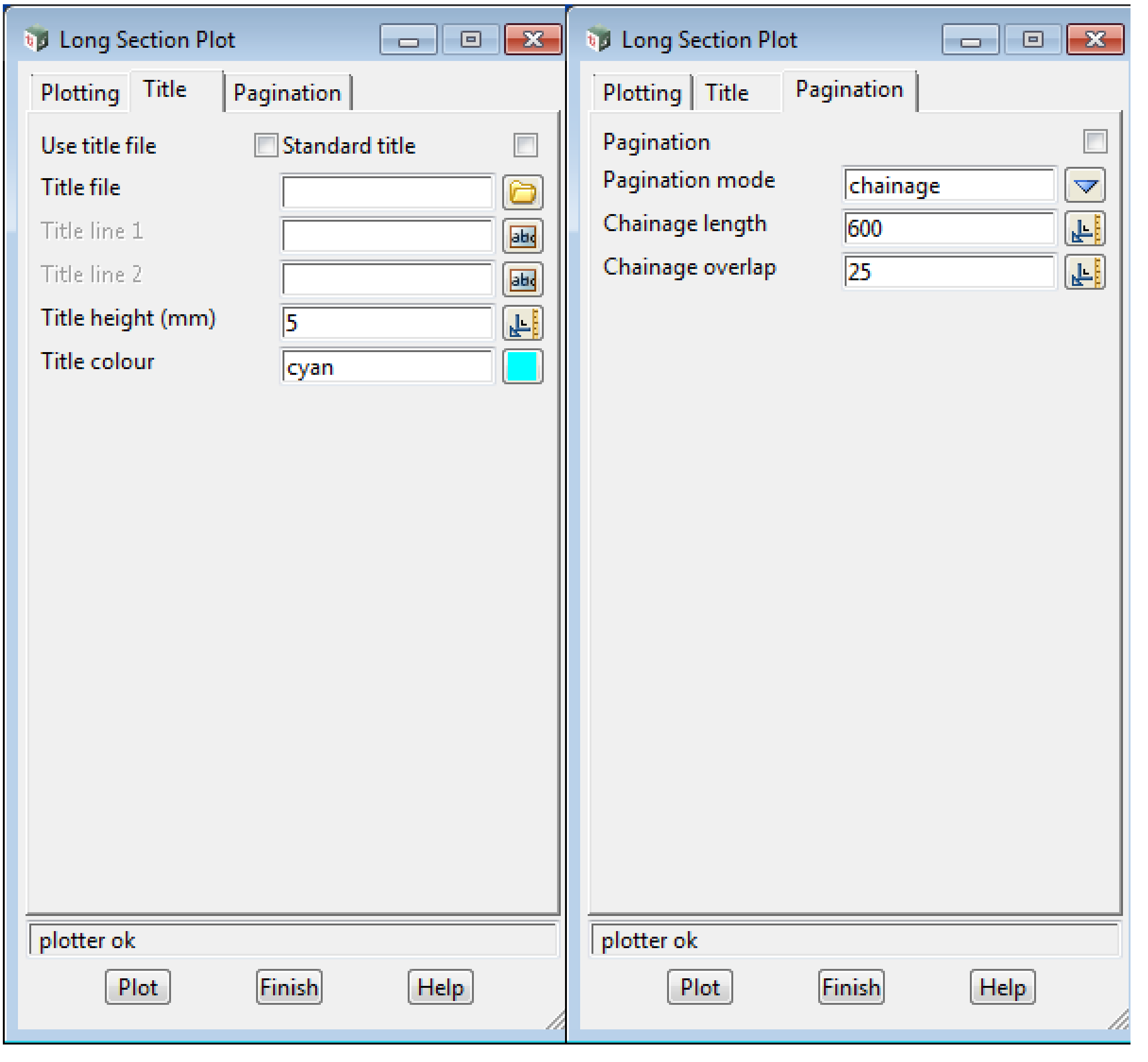
In the pathname to the default .ppf file, $LIB is used to stand for the library folder set by LIB_4D. For example, "$LIB/long.ppf" is the file long.ppf in the library area.

The **title** button on the bottom of this panel controls the use of a title block file and the plotting of a border and two lines of title. If the **title** button is selected, the **section long plot title** panel is displayed.

The **pagination** button on the bottom of this panel controls whether the long section plot is broken into pages, by a chainage length or millimetres of plot.

If the **pagination** button is selected, the **Section long plot pagination** panel is displayed.


Long Section Plot
Plotting
Title
Pagination
Plot parameters
Plot parameters write
Section view
Plotter type
hp
Plot file
Start chainage
End chainage
Chainage increment
100
Sheet size wd ht (mm)
A1
Scale 1 :
1000
Plot primary string
Use HG VG for min max
Label depths
Text style
1
Box text ht (mm)
3
Box colour
cyan
Datum value
Offset model
plotter ok
Plot
Finish
Help



The fields and buttons used in the two panels have the following functions.

Field Description	Type	Defaults	Pop-Up
Plot parameters	file box	LONG_SECTION_PPF_4D*.ppf	

file of plot parameters used for extra control of the long section plot. The default .ppf file is pointed to by the environment variable LONG_SECTION_PPF_4D.
For more information on the long section plot parameters, please go to the section 44.3 Long Section Plot Parameter File *in the Appendix* 44 Text Plot Parameters

Plot parameters write	file box		*.ppf

file to write a copy of the plot parameters necessary to regenerate the plot.

Section view	input/output	current view	available section views

section view to use for plot definitions such as the tins to profile, the models of services, corridor width and vertical exaggeration.

Plotter type	input	hp	hp, dxf, postscript etc.

file format for the plot information.

Plot file	file box		*.hp etc.

name of the file to write out the long section plot to.

Start/end chainage input

the plot covers the chainage range given by the start and end chainage fields. If the **start/end chainage** *is blank, the star/end chainage of the primary string is used.*

Chainage increment input 100

chainage increment to label the long plot with chainage and height values.

Sheet size wd ht (mm) input available sheet sizes

The width and height values (separated by space) or the name of a user defined sheet size.

Scale 1: input

horizontal scale for the plot. The vertical scale is taken from the ppf or the section view.

Plot primary string tick box ticked

if ***ticked****, the primary string is plotted along with any horizontal geometry displayed in the section view.*

Use HG VG for min, max tick box ticked

if ***ticked****, the chainage range available for plotting is from the minimum of the horizontal geometry (HG) and the vertical geometry (VG) to the maximum of the HG and VG. Useful for kerb returns.*
If ***not ticked****, the chainage range available for plotting is from the minimum of the horizontal geometry (HG) to the maximum of the HG.*

Label depths tick box ticked

if ***ticked****, label in boxes at the bottom of the plot, the distance between the primary string and the tins.*

Text style input 1

text style to be used in the long section plot boxes.

Box text ht (mm) input 3

height (in millimetres) to plot the chainage and elevations in the boxes in the long section plot.

Box colour colour box cyan available colours

colour used for the chainage text and the boxes.

Datum value input

if ***non-blank****, the value to be used as a datum. If* ***blank****, then a suitable datum value will be calculated.*

Offset model input available models

if ***non-blank****, then all the strings in the offset model will be projected onto the primary string and drawn and labelled on the long section plot.*

Title tab

brings up the section long plot title panel.

Use title file tick box not ticked

if ***ticked,*** *a use supplied title block file is used.*

Standard Title tick box ticked

if ***ticked****, the standard 4D Solutions border and two lines of title are placed on the bottom of the plot*

Title file file box available *.tf files

if ***non-blank*** *and* **use title file** *is set to* ***tick****, then the file given in this field is used to generate a user defined title block for the plot.*

Title line 1/2 input

first/second line of title information

Title height (mm) input 5

height (in millimetres) to draw the characters in the two lines of title information.

Title colour colour box cyan available colours

colour used for the border and the title information.

Pagination tab

brings up the section long plot pagination panel.

Pagination tick box ticked

*if **ticked,** the long section plot is broken into separate plots of length given in the* **length** *field, from the start chainage to the end chainage. If the* **overlap** *field is non-zero, then each page of the long section plot also includes the* **overlap** *value of the end of the previous page of plot.*

The units for **length** *and* **overlap** *can be either **chainage distance** or **millimetres on the plot page**.*

Hence apart from the first page and possibly the last page, the plot will have a total length given by the sum of the **length** *and the* **overlap** *panel fields.*

Pagination mode input chainage chainage, millimetres

the units for **length** *and* **overlap** *can be either **chainage distance** or **millimetres on the plot page**.*

If **pagination mode** *is set to **chainage**, the units are chainage distance.*
If **pagination mode** *is set to **millimetres**, the units are millimetres on the plot.*

Length measure box 600

if **pagination** *is set to **ticked**, the length of new plot to be included in the current plot.*

Overlap measure box 25

if **pagination** *is set to **ticked**, the length of the previous plot page to be included in the current plot.*

Plot button

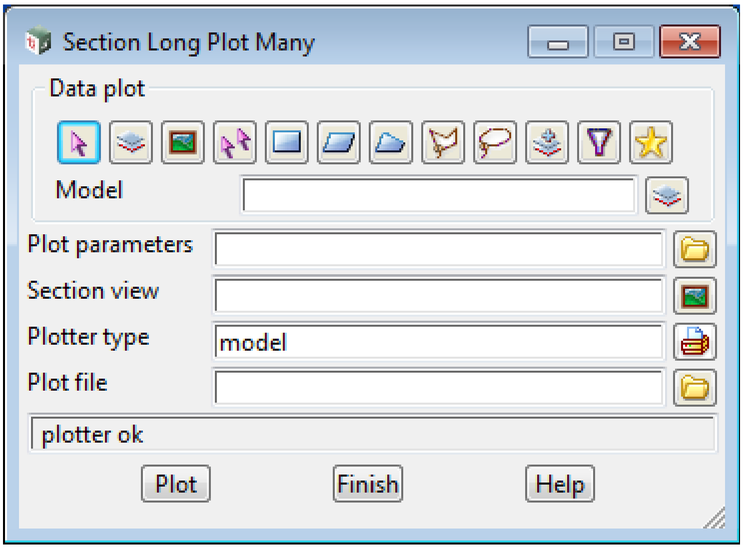
write out the plot for the long section plot between the chainages given in the start and end chainage fields. The format of the file is given by the **plotter type**

25.8.5 Plot Many Long Sections

Position of option on menu: **Plot =>Old plotting =>Long Section Plot Many**

The **Long Section Plot Many** option is used to plot individual long section plots for each string selected by the Data source.

On selecting the **Long Section Plot Many** option, the **Section Long Plot Many** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Data to plot - Data source type

data source type.

Data source

when the **plot** *button is selected, long section plots are created for all the strings selected by the data source.*

Plot parameters file box *.ppf files

file of plot parameters to be used for the long section plots.

Section view view box available views

name of the section view for defining information for the long section plots. For example, tins to plot.

Plotter type input hp available plotters

type of plotter to plot to.

Plot file input

stem of the name to use for the long section plots.

Plot button

when the **Plot** *buttons is selected, individual long section plots will be created for all the strings selected in the data source field.*

25.8.6 Drainage/Sewer Plot

Position of option on menu: **Plot =>Old plotting =>Drainage/Sewer**

The **Drainage/Sewer** option is for creating drainage and/or sewer long-section plots.

The option has already been described under **Design=>Drainage-sewer =>Plots**.

See 22.9.2 Drainage Longsections.

25.8.7 Melbourne Water Sewer Plot

Position of option on menu: **Plot =>Old plotting =>Melbourne Water**

The **Melbourne Water** option is for creating sewer long-section plots to Melbourne Water specifications.

The option has already been described under **Design =>Drainage-sewer =>Plots**.

See 22.9.3 Melbourne Water.

25.8.8 Pipeline Plot

Position of option on menu: **Plot =>Old plotting =>Pipeline**

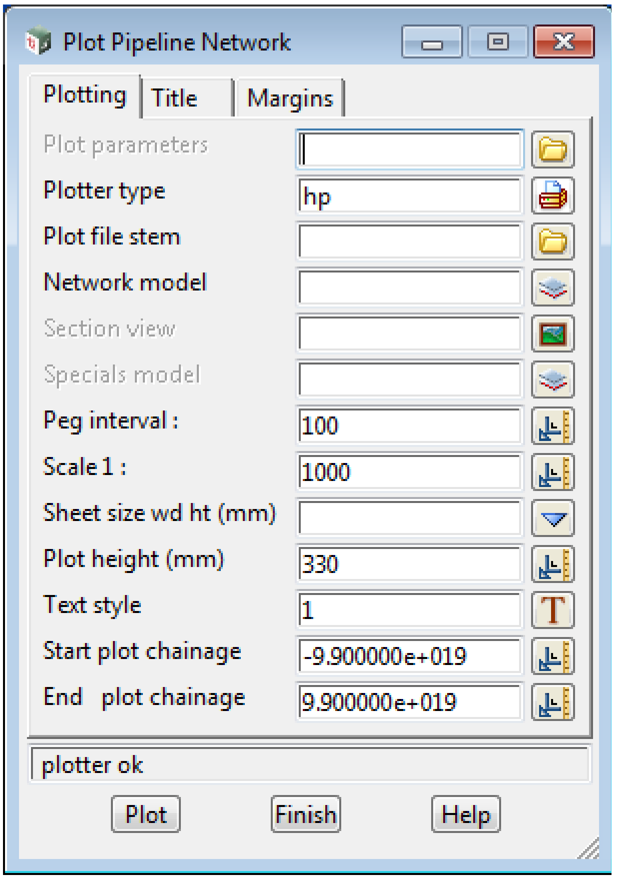
The **Pipeline** option is for creating pipeline long-section plots.

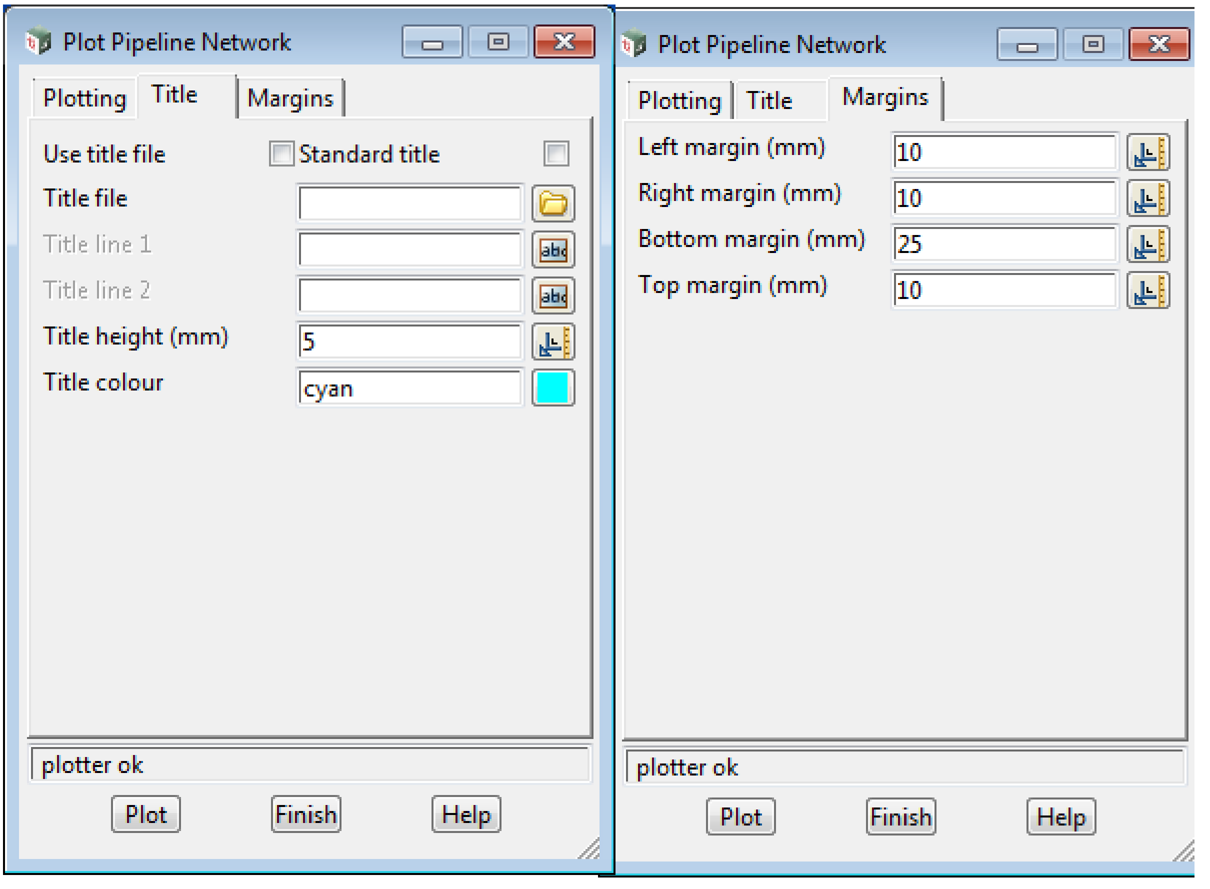
Given the plot sheet size and the horizontal and vertical scales, the longsections for the pipelines are plotted starting at the top of the sheet and moving across the sheet. Once one row is full, if there is room the plot moves down the page and begins a new row. When a plot sheet is full, a new plot sheet is automatically begun.

The pipeline plots include

- the horizontal and vertical joint deflection at any vertical intersection points which have no curves on them
- the natural surface height and chainage at any pegs or positions from the specials model, plus the invert level and depth to pipe at the position
- for any services which cuts the pipeline, the name, diameter, invert level, the distance along the pipeline of the cut point plus the invert level and depth of the invert for the pipeline and the cut point.

After selecting the **Pipeline** option, the **Plot Pipeline Network** panel is displayed.





The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Plot parameters	input		*.ppf

file of plot parameters used for extra control of the long section plot.

Plotter type	input	hp	hp, dxf, postscript etc.

format for the plot output.

Network model input

the model containing the pipelines to be plotted.

Section view input

the section view to be used to define the vertical exaggeration, corridor widths, tins to section through, services models to section etc.

Plot file stem input

*since more that one plot page may be produced, the plot file stem plus a plot page sequence number followed by **.plt** is used as the plot file names.*

Specials model input

model of text strings which are used as extra labels for the plots.

Peg interval input 50

if the **specials model** *is blank, the pipelines are labelled with the text **peg** at the peg interval along the* *pipeline.*

Scale 1: input

horizontal scale for plotting the pipeline long sections. The vertical exaggeration is taken from the section view given in the **section view** *field.*

Sheet size wd ht (mm) input available sheet sizes

*if **non-blank**, the width and height values in millimetres (separated by space) or the name of a user defined sheet size.*
*If **blank**, the sheets size is calculated to fit the long section plot.*

Pipeline line ht (mm) input

the maximum allowable height for a longsection plot for a pipeline line. Datum breaks are applied to any part of the longsection that will not fit into the **pipeline line ht***.*

Start plot chainage input

If blank, the start chainage of the pipelines is used; if non blank, the plots start at this chainage for each pipeline.

End plot chainage input

If blank, the end chainage of the pipelines is used; if non blank, the plots finishes at this chainage for each pipeline.

Plot button

write out the pipeline longsection plots for the pipelines in the model given in the **network model** *field.*

The fields and buttons for *margin* tab:

Left/right margin (mm) 10.0

the left/right hand side margin between the sheet and the plotting area inside the sheet.

Bottom/top margin (mm) 10.0

the bottom/top margin between the sheet and the plotting area inside the sheet.

The fields and buttons for the *title* tab:

Use title file tick box

*if **ticked,** a user defined title block file is used.*

Use Standard title tick box tick

*if **ticked**, the standard border and two lines of title are placed on the bottom of the plot*

Title file input *.tf

*if **non-blank** and* **use title file** *is set to **tick**, then the file given in this field is used to generate a user defined title block for the plot.*

Title line 1/2 input

first/second line of title information

Title height (mm) input 5

height (in millimetres) to draw the characters in the two lines of title information.

Title colour input cyan available colours

colour used for the border and the title information.

25.8.9 Edit Title Block Alias Data

Position of option on menu: **Plot =>Old Plotting =>Edit title block data**

NO LONGER USED IN V7

This panel is used to easily change to data in your drawing title blocks (set values for the user_text_n field in a title file).

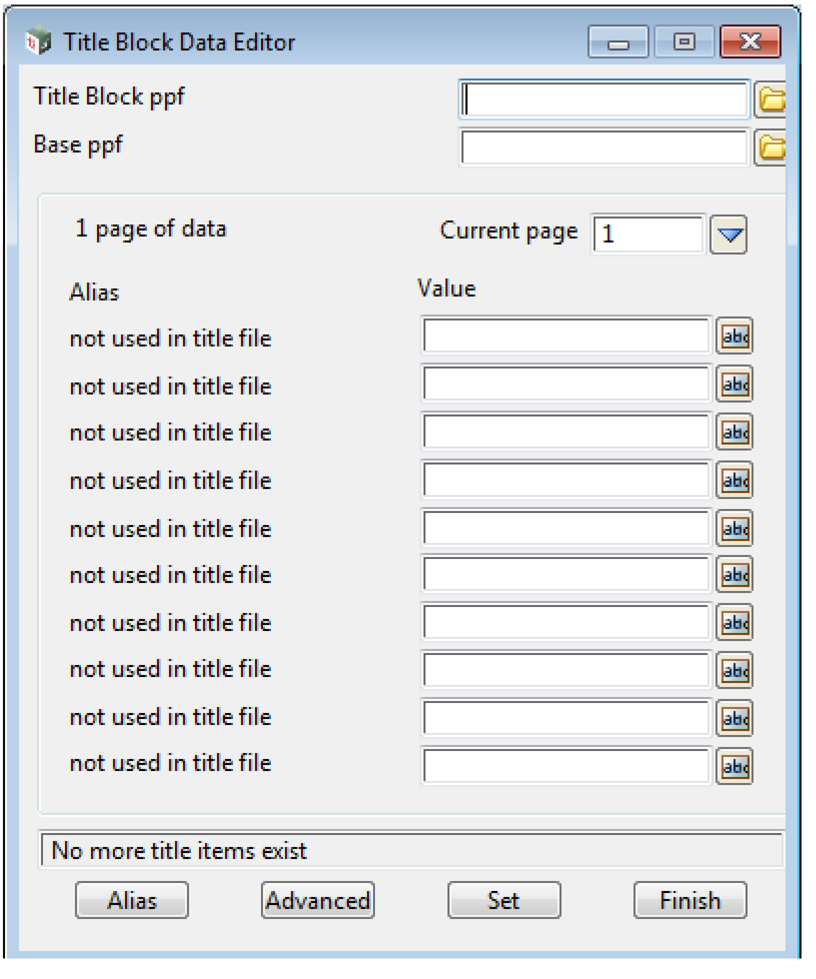
The ***Title block ppf*** is the ppf that is created by the panel. You may want to create a title block ppf for every drawing you create. This way you can easily re-plot a drawing with the same title block data. This is the ppf name that you will use in the future plot panels when you are ready to plot using the title block.

The ***Base ppf*** field is an existing ppf that includes all of your default plot parameters for creating your drawing. It must reference a title file.

12d marks the location of your title block data using the variables *user_text_1, user_text_2 etc*. Since it is difficult to remember what *user_text_1* represents in the title block, 12d allows you to use a meaningful name reference (Main title for example) instead of *user_text_1.* The Alias button allows you to set these alias names for the title file referenced in your base ppf.

The Advanced button is only used if the same text data is to be used in another Title Block ppf (share the data between drawing).

The Set button creates/updates the title block ppf and the title file values that you entered.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Title block ppf

this is the ppf that is created by the panel. This is the ppf name that you will use in the future plot panels when you are ready to plot using the title block. The Title Block ppf is usually stored locally (in the current folder).

Base ppf

this ppf must exist (often in library or your user library). It will contain all of your default plot parameters for creating your drawing. It must reference a title file (see notes below).When you press enter, the ppf file is read to find the name of the title file you reference. Once the title file is found, the alias names are read from the title alias file if available.

To convert an existing ppf file into a Base ppf file use a text editor to delete all references to *user_text_n* variables in the file.

Current page choice box available pages

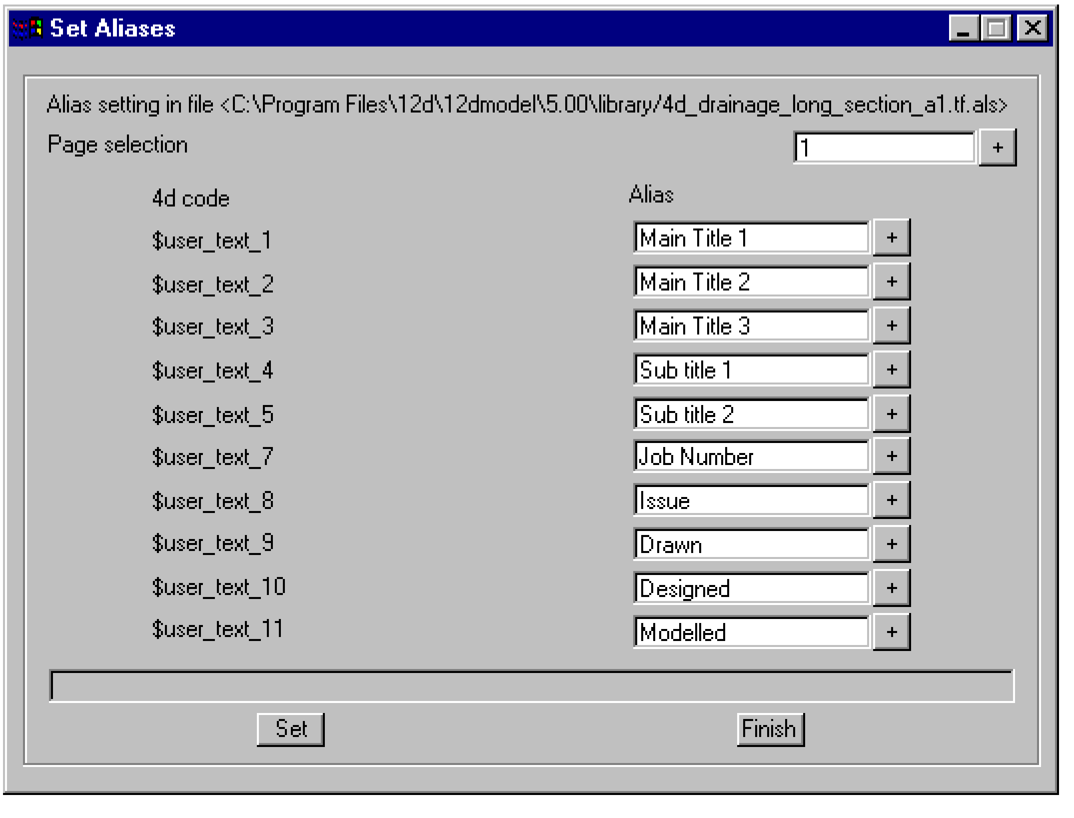
if more than one page of data (10 fields) are required additional pages will be listed here.

Value input box

the data to be shown into your drawing title block is entered here. The description to the left is either *$user_text_n* (Note that there may be gaps in the numbering sequence) or an alias that has been set up using the alias button below.

Alias button

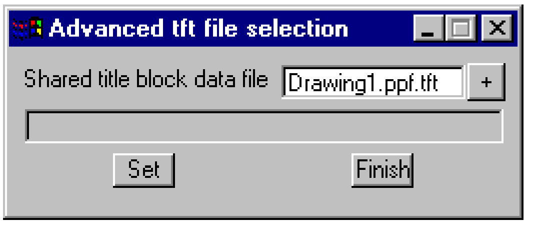
if the fields have the names *$user_text_1, $user_text_2 etc.* use the button to change the names to a more meaningful description. These new names will be stored in the same folder as the with the title file specified in your base ppf file (the extension ***als*** will be added). The following panel will be displayed.



Note that there may be gap in the *$user_text_n* sequence. Only the variable used in the title file will be listed.

Advanced button

this feature allows you to share text data between different *title block ppf* files. The actual text data in stored in a title file txt (*.tft) file. Selecting the button will bring up the following panel and allow you to change the *.tft file to be used. Generally you will select a *.tft file previously set up for another title block file.



Select the **Set** button to set your select. The text data will be read and the main panel updated when you select the **Finish** button.

Set button

creates/updates the Title block ppf file and *.tft file.

Notes:

Short-cuts:

***.als** and ***.tft** files can be copies, renamed and then edited using the Title Block Edit Panel or your own text editor.

Alias files (*.als) are kept in the same folder as the your title files. Once you have one alias file set up copy the alias file and rename the copy the same name as the title file but add the additional extension .als. For example:

title file 4d_drainage_long_section_a1.tf
alias file 4d_drainage_long_section_a1.tf.als

CAUTION: The *user_text_n* variables should represent the same data in both title files. For example *user_text_1* should be the main title in both files.

Title file text files (*.tft see advanced) files are kept in the same folder as the your Title block ppf. If you choose not to share a tft between Title block ppf files you can simply copy and rename them. Once you have saved one *.tft file, copy the file and rename the copy the same name as the title block file but add the additional extension .tft

Title Block ppf file drawing 1.ppf
title file text filedrawing 1.ppf.tft

Title Block ppf

The title block ppf file does not contain any data itself. Rather it references a *.tft file containing the text data and your base ppf file (file containing all of your default plot parameters). An example ppf file follows:

The Title Block ppf file is a ppf file with the Base ppf and user text values (in the tft file) referenced by #include statements.

```
#define ALIAS 1
section_long_plot "plot 2" {
// TITLE BLOCK PPF FILE
#include "Base_ljg.ppf"
#include "Title_block.ppf.tft"
}
```

Base ppf

The Base ppf can be anywhere including local, Library and User Library. It must contain a reference to a title file in the *use_title_file* parameter. The *user_title_file* parameter must be in the file and can't be in an #include file.

The following is an example title file reference.

use_title_file	yes
title_file	"your title file.tf "

Note: If you already have user_text variables defined in your Base ppf these values will not be used and a warning message will be issued at plot time. It is best to remove all user_text variables from your existing ppfs before attempting to use them with this new 4d facility.

The panel scans the title file reference and records all of the *user_text_n* entries that are used. **Only those used are displayed in the panel.** Next a file with the (is located. If found the alias values are used in the panel instead of *$user_text_1, $user_text_2 etc.*

If an alias file (same name as the title file PLUS the extension .als) has been previously created using the Alias button, the aliases for the user_text's are used on the left hand side of the panel. If an alias file has not been found, they can be created by simply selecting the Alias button.

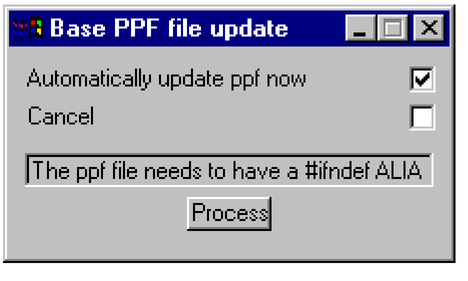
The Base ppf is also checked to see if a

#ifndef ALIAS

and

#endif

exist around the opening "xxxxxxxx {" and closing "}" braces. If not, a Base PPF file update panel is displayed.



Select *Automatically update ppf now* and select the Process button. The extra lines then are added to the Base ppf.

For example, the beginning and end of the updated file will look like:

BEGINNING

#ifndef ALIAS

section_long_plot "plot 2" {

#endif

END

#ifndef ALIAS

}

#endif

Clicking on the Alias button brings up the Set Aliases panel. Simply fill in the names to use instead of the words $user_texts_n in the Title Block - User Text Data Editor panel. Selecting Set writes out an aliases file which has the same names as the title file but with the appended ending .als.

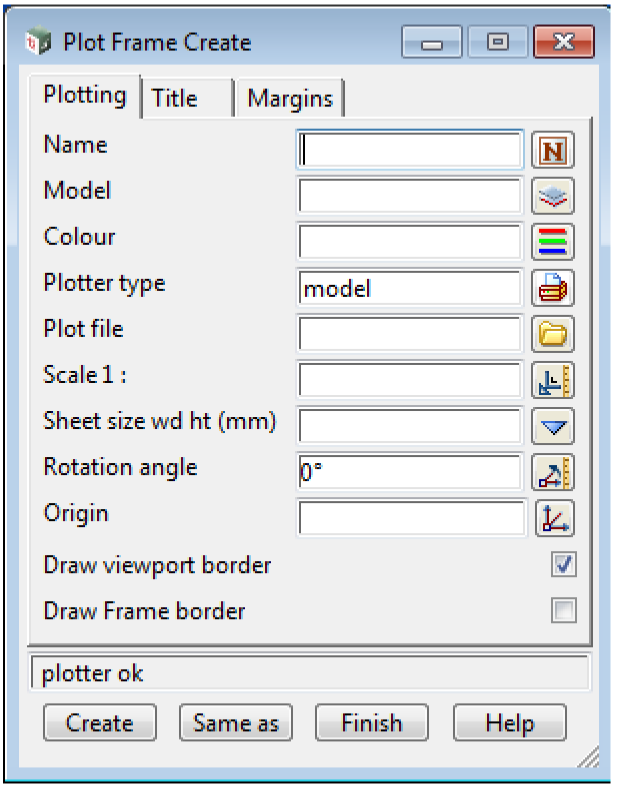
Set

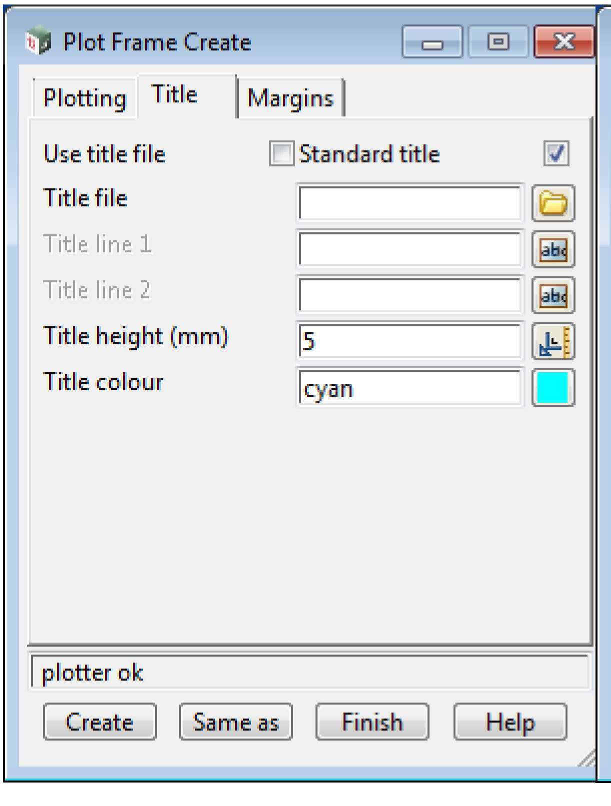
After selecting Set, the values for the user_texts are written to a file specified under **Advanced** (if used) or of the same name as the Title block ppf but with .tft appended.

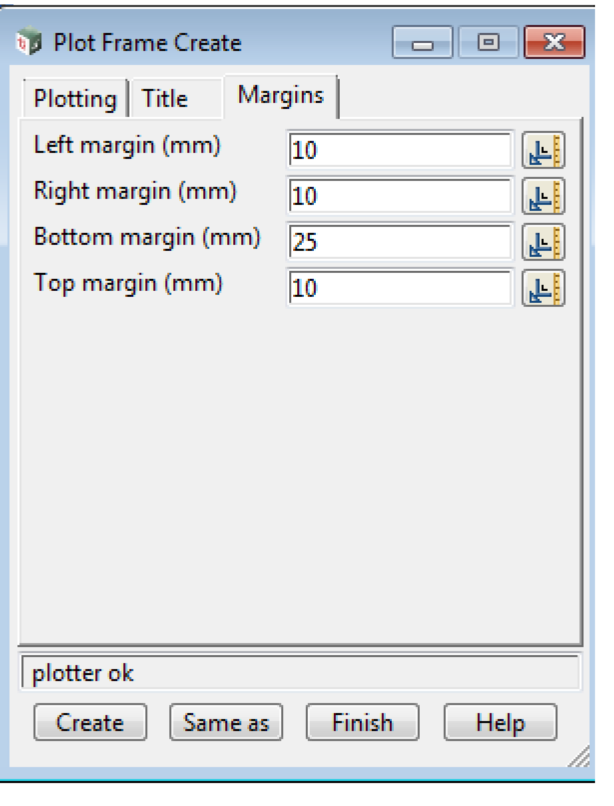
25.8.10 Create Old Plot Frame

Position of option on menu: **Plot =>Old plotting =>Create plot frame**

On selecting the **Create old** option, the **plot frame create** panel is displayed.







The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Name	input		

name for the plot frame.

Model	input		available model

the model for the plot frame.

Colour	input		available colours

the colour for the plot frame.

Plotter type	input	hp	hp, dxf, dwg, postscript etc.

format for the plot file.

Plot file	input		*.hp etc.

name of the file to write the plot of the plot frame to.

Scale 1:	input		

scale for the plot.

Sheet size wd ht (mm)	input		

the width and height values (separated by spaces) or the name of a user defined sheet size.

Rotation angle	input	0	

angle of rotation for the plot frame.

Origin	input		**xyz ops** menu

x_origin y_origin z_origin

origin (in world units) for the corner of the plot frame - given as three values separated by spaces. The values can either be typed in, or selected by clicking LB in the **origin** *panel view and getting up the* **xyz ops** *menu and selecting the* **pick xyz** *option.*

Draw viewport border	tick box	tick	

*if **ticked**, plot the box around the plotting area (viewport).*

Create	button		

create the plot frame and then put up the plot frame edit panel for the created plot frame.

Same as	button		

after picking the **same as** *button, an existing plot frame is selected and its information is used to fill in the above panel fields.*

Title tab

Use title file	tick box		

*if **ticked**, use the file given in the* **title file** *field to generate a title block.*

Standard Title	tick box	tick	

*if **ticked**, plot the standard 12d Model title block with the text, height and colour given in the following panel fields.*

Title line 1/2	input		

first/second line of title information in the 12d titleblock or for the user defined title block.

Title height (mm)	input	10	

*height (in **millimetres**) to draw the two lines of title information in the 12d titleblock.*

Title colour	input	cyan	available colours

colour used for the border and the title information in the 12d titleblock.

Title tab

Left/right margin (mm) 10.0

the left/right hand side margin between the sheet and the plotting area inside the sheet.

Bottom/top margin (mm) 10.0

the bottom/top margin between the sheet and the plotting area inside the sheet.

How To Use the Option

The plot frame contains all the information required to define the physical area in real world units to be plotted, the position on the sheet of the area being plotted, the plotter type and the plot file name.

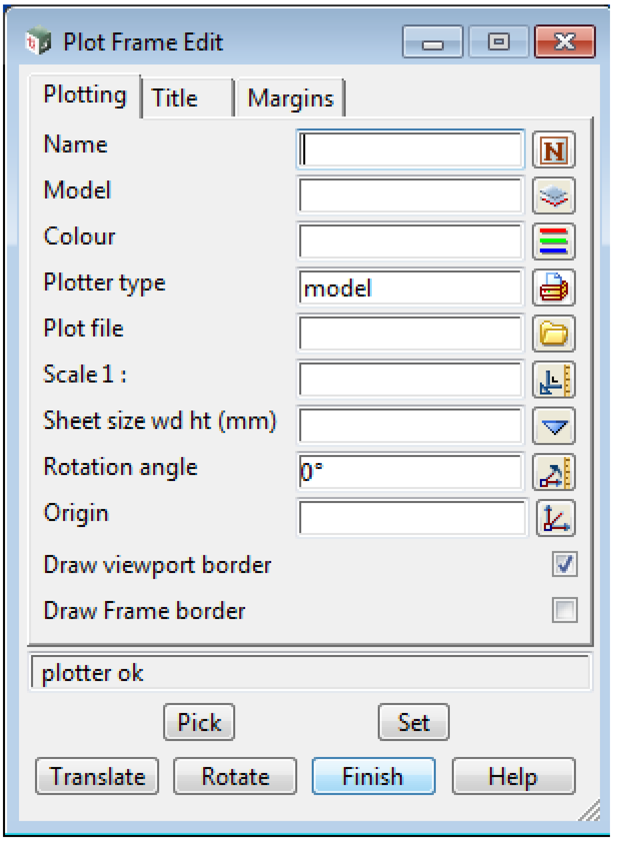
Once the frame is created, the **plot frame edit** panel for the created plot frame is automatically placed on the screen so that the frame's position can be adjusted using the **Translate** and **Rotate** buttons.

25.8.11 Editor Old Pot Frame

Position of option on menu: **Plot =>Old plotting =>Editor plot frame**

The plot frame editor is used to edit existing plot frames. It is automatically placed on the screen when a plot frame is created so that the frame's position can be adjusted using the **Translate** and **Rotate** buttons.

On selecting the **Editor old** option, the **plot frame edit** panel is displayed.



Most of the fields and buttons used in this panel are the same as the **plot frame create** panel - see the previous section 25.8.10 Create Old Plot Frame. The different ones are

Field Description	Type	Defaults	Pop-Up

Pick button

select the plot frame to be edited.

Set button

update the plot frame with the information in the above panel fields.

Translate button

after picking the **translate** *button, the plot frame will move with the cursor until a point is selected to give the final position of the plot frame.*

Rotate button

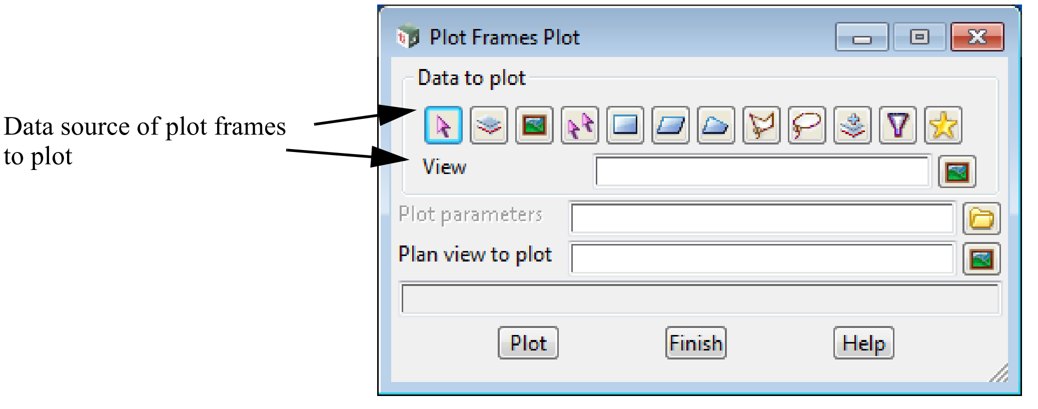
after picking the **rotate** *button, the plot frame is further rotated around the left hand corner of the sheet, by the angle made between the rotation point and the current cursor position.The plot frame will continue to rotate until a point is selected to fix the rotation angle.*

25.8.12 Plotting Old Plot Frames

Position of option on menu: **Plot =>Old plotting =>Plot plot frame**

The **plot old** option is used to make a plot of the data on a given plan view for all the selected plot frames (selected by the data source). Note that the plots are of a given plan view and the plot frames do not have to be on the plan view.

On selecting the **plot old** option, the **plot frames plot** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field DescriptionType Defaults Pop-Up

Data to plot - Data source type

data source type.

Data source

when the **plot** *button is selected, all the plot frames in the data source will be plotted.*

Plot parameters input *.ppf files

file of plot parameters to be used for the plot frames.

For more information on the plot frame plot parameters, please go to the section 44.1 Plot Frame Plot Parameter File *in the Appendix* 44 Text Plot Parameters

Plan view to plot input available views

the name of the view that is plotted when the plot frames are selected for plotting. Note that the plot frames ***do not*** *have to be on the view.*

Plot button

when the **plot** *buttons is selected, all the plot frames in the data source field will be plotted.*

25.8.13 Plotter Mapping Table

Position of option on menu: **Plot =>Plotting setups =>Plotter Mapping**

The **Plotter mapping file** can be used with pen plotters but is more specifically designed for electostatic and inject plotters and allows the user to

(a) map the full 10,240 12d Model colours to particular plotter pens and also specify a width (or weight) to be used forPCL5, HPGL2 and postscript plotters.

(b) specify the red, green and blue mix for pens on HPGL2 and colour postscript plotters

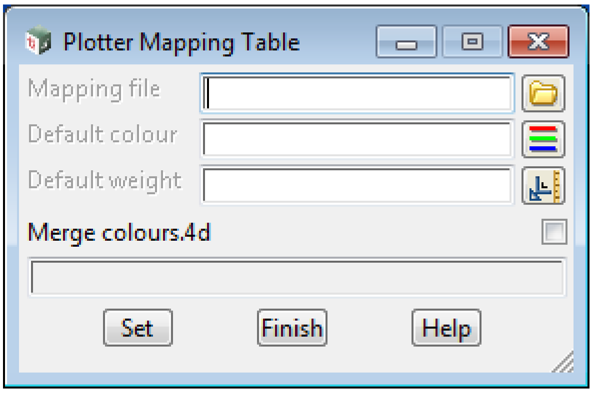
There are also two special formats of the plotter mapping file which are used for plotting to Autocad dxf/dwg and Intergraph dgn. The special mapping files use the 12d Model colour as a key to tables which control how the information is passed to DXF/DWG and DGN.

(c) .tbl file used with plotting to Intergraph dgn

(d) .amf used with plotting to Autocad dxf and dwg

The .tbl and .amf files are documented in the 43.3 Mapping 12d Colours to Pens and RGBs and 43.2.1.3 Definition and Example of a .amf File sections in the Appendix 43 Plotters and Plotting.

On selecting the **plotter mapping** option, the **Plotter Mapping Table** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Mapping file | *pmf box | | *.pmf file

the plotter mapping file to be used when plotting.

If no plotter mapping file is given, the pen mapping table in colours.4d is used.

Default colour | colour box | | available colours

*if **not blank**, the rgb of this colour is used as the rgb for the default pen colour.*

Default weight | input

*if **not blank**, the default pen weight.*

Merge colours.4d | tick box | ticked

*If **ticked**, the pen_mapping section is populated with the rgb columns from the colours.4d file are used to define the pen_colours for the colour_number row, and the pen mapping number is used to construct the pen_mapping table.*

Set | button

set the information in the panel

25.8.13.1 Initialization of the Plotter Mapping File

25.8.13.1.1 When using the Plotter Mapping Table panel:

When used with the **Plotter Mapping Table** panel, the *pen_mapping* and *pen_colours* sections are initialised before the plotter mapping file is used. This is to ensure that any colours and/or pens not mentioned in the file but used in a plot, do have valid definitions.

The initialisation sequence for the plotter mapping file used with the **Plotter Mapping Table** panel is:

Before the plotter mapping file is used:

(a) a default *pen_mapping* table is constructed for the colours in *colours.4d* by:

colour n goes to pen n with the **default weight** from the **Plotter Mapping Table** panel.

(b) a default *pen_colours* table for pens 0 to 10,240 is set up by:

pen n *default_red* *default_green* *default_blue*

where *default_red*, *default_green*, *default_blue* are the red, green and blue of the **default colour** from the **Plotter Mapping Table** panel.

If the **merge colours.4d** flag is set in the **Plotter Mapping Table** panel, the rgb columns from the *colours.4d* file are used to define the *pen_colours* and the *pen mapping number* is used to construct the pen_mapping table.

The **plotter mapping file** is then processed and over writes any of the above initial mapping values.

25.8.13.1.2 When Using a Plotter from Plotters.4d:

A *plotter mapping file* can be set for a user defined plotter in the file *plotters.4d*.

In this case, the **Plotter Mapping Table** panel is **not** used and hence the initialisation sequence is slightly different.

For a **user defined plotter**, the initialisation sequence for the plotter mapping file is:

Before the plotter mapping file is used

(a) a default *pen_mapping* table is constructed from the colours in the *colours.4d* file by:

colour n goes to pen n *default_weight*

where **default weight** is the *default_weight* if one is given in the specified plotter mapping file or zero if the default_weight does not exist.

(b) a default *pen_colours* table for pens 0 to 10,240 is set up by:

pen n *default_red* *default_green* *default_blue*

where *default_red*, *default_green*, *default_blue* are the red, green and blue of the *default_colour* if one is given in the specified plotter mapping file, or rgb 0 0 0 if the default_colour does not exist.

The **plotter mapping file** is then processed and over writes any of the above initial mapping values.

25.8.13.2 An Example of a Plotter Mapping File

An example of a plotter mapping file (.pmf) is:

```
default_colour        100     0     100
default_weight        0.1

pen_mapping {                       // pen mapping and weight table
```

```
//          colour      pen         weight

            0           1           0.15        // map colour 0 to pen 1 with weight 0.15
            1           2           0.25        // map colour 1 to pen 2 with weight 0.25
            3           1           0.5         // map colour 3 to pen 1 with weight 0.5
}
pen_colours {           // pen colours table

//pen       red         green       blue     values 0-255
   0        0           0           0           // pen 0 is black
   1        255         0           0           // pen 1 is red
   3        0           255         0           // pen 3 is green
}
```

25.8.14 Pen Mapping Old

Position of option on menu:

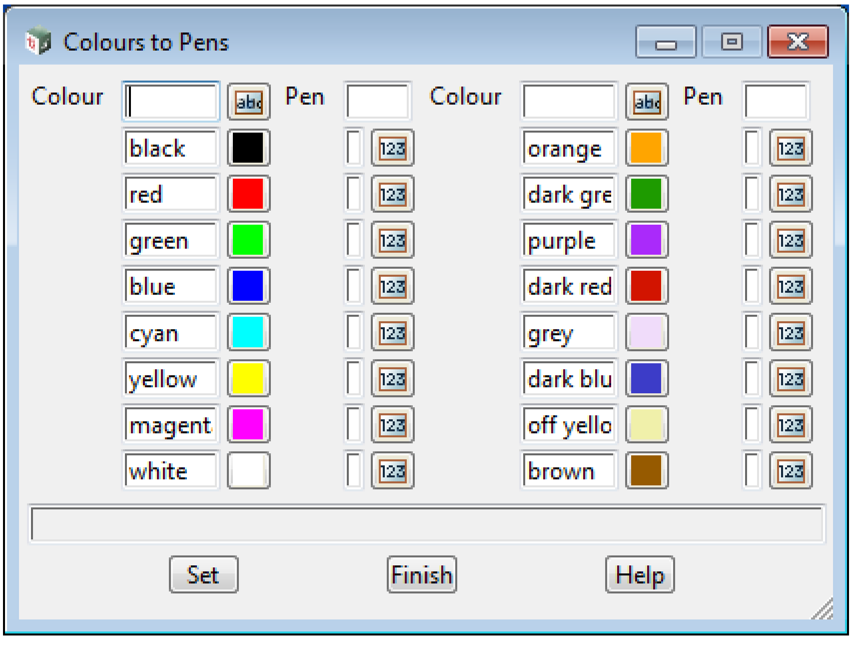
NO LONGER USED

Although 12d Model uses up to 10,240 different colours, most plotters have a smaller number of pens. Consequently for any plot, it is necessary to define what pen number corresponds to each 12d Model colour.

For all plotters types except Eagle, the **pen mapping** option can be used to define the correspondence between 12d Model colours and plotter pens for the first sixteen colours. The other colours are given in the colour_map.def file. For Eagle plots, the **eagleplt.pmf** file is used to map colours to pens.

The pen mapping option is only used if a plotter mapping is not being used (see next section).

On selecting the **pen mapping** option, the current pen mapping table is displayed in the **colours to pens** panel.



When starting a new project or entering an existing project, the default pen mapping table is defined by the **colours.4d** file.

Once inside the project, any pen number in the table can be modified by changing the appropriate **pen** field and then selecting the **set** button. The modified table is stored until 12d Model is quit or the user changes to another project.

The pen mappings for colours above sixteen are given in the **colours.4d** file but can not be displayed or modified by the **colour to pens** panel.

Whenever a plot is created using non-eagle plotter types, the colours are mapped to the pen numbers given in the **colours to pen** table before writing out the plot file.

For Eagle plots, the **eagleplt.pmf** file is used to map colours to pens.

Note - when 12d Model is started up again or the user changes to a new project, the table reverts to its default settings as given in the colours.4d file.

26 PPF Editors

26.1 General Information on PPF Editors

For the special plot types:

> Cross sections, General Long sections., Plot Frames, Drainage Long sections, Drainage plan, Melbourne Water long sections and Pipeline Long sections.

there are plot parameters for controlling what items are drawn on the plots, what colours are use, what the text is and its colour and sizes, what the title block, what information is in the title block including user supplied values that can vary with each plot, *etc*.

The plots themselves are considered to be drawn onto a sheet of paper with the origin (0,0) in the bottom left hand corner and with drawing units of millimetres.

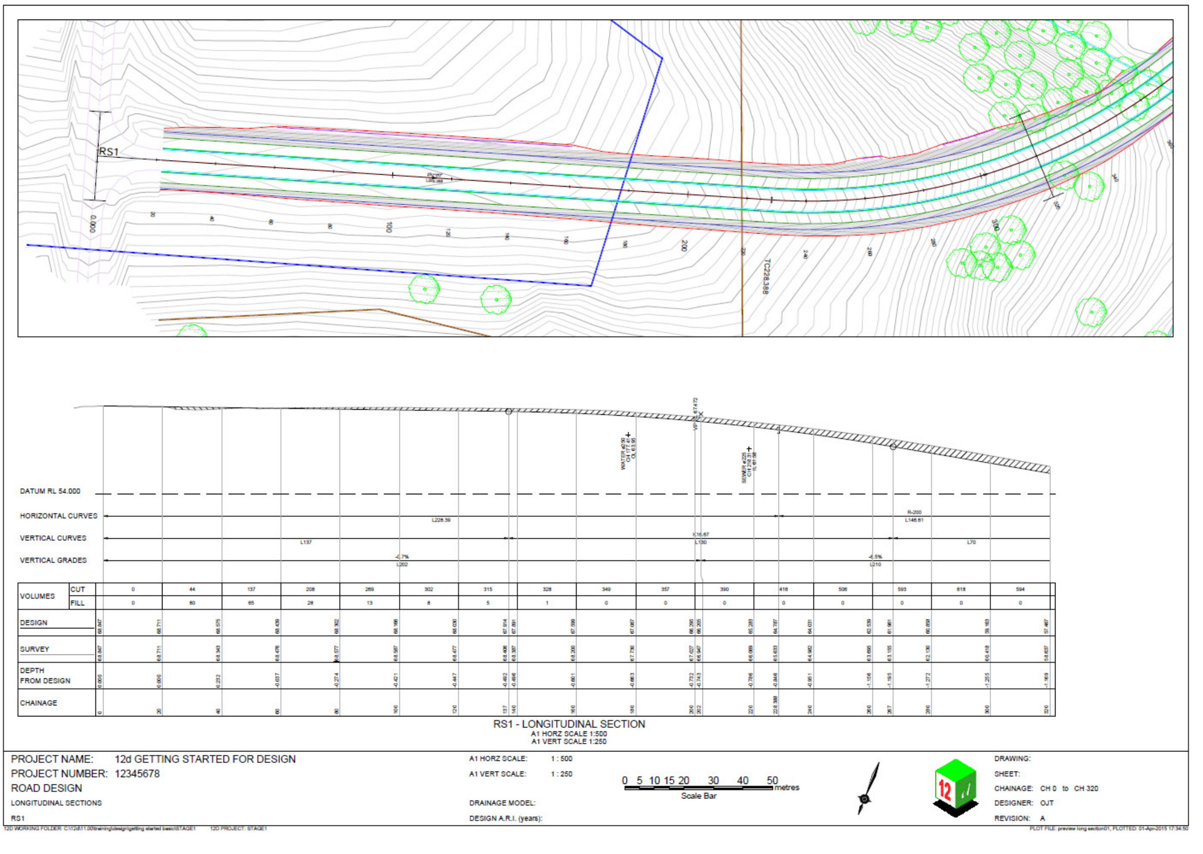
So when any of the plots is generated from ***12d Model***, the data is translated and scaled so that it fits onto the positive quadrant with (0,0) in the left hand corner. This means that title blocks and overlay models can be set up knowing exactly where they will sit in relation to a plot.

For each of the special plot types, there is a ***PPF Editor*** that allows the user to interactively change any of the plots parameters, specify the type of plotter to use (pdf, model, Windows *etc*) and actually produce the plot.

All this information can be saved away in a plot parameter file (ppf) for future reuse.

In fact the ppf has all the information required to generate the plot and can even be used to create a plot without any interaction from the user.

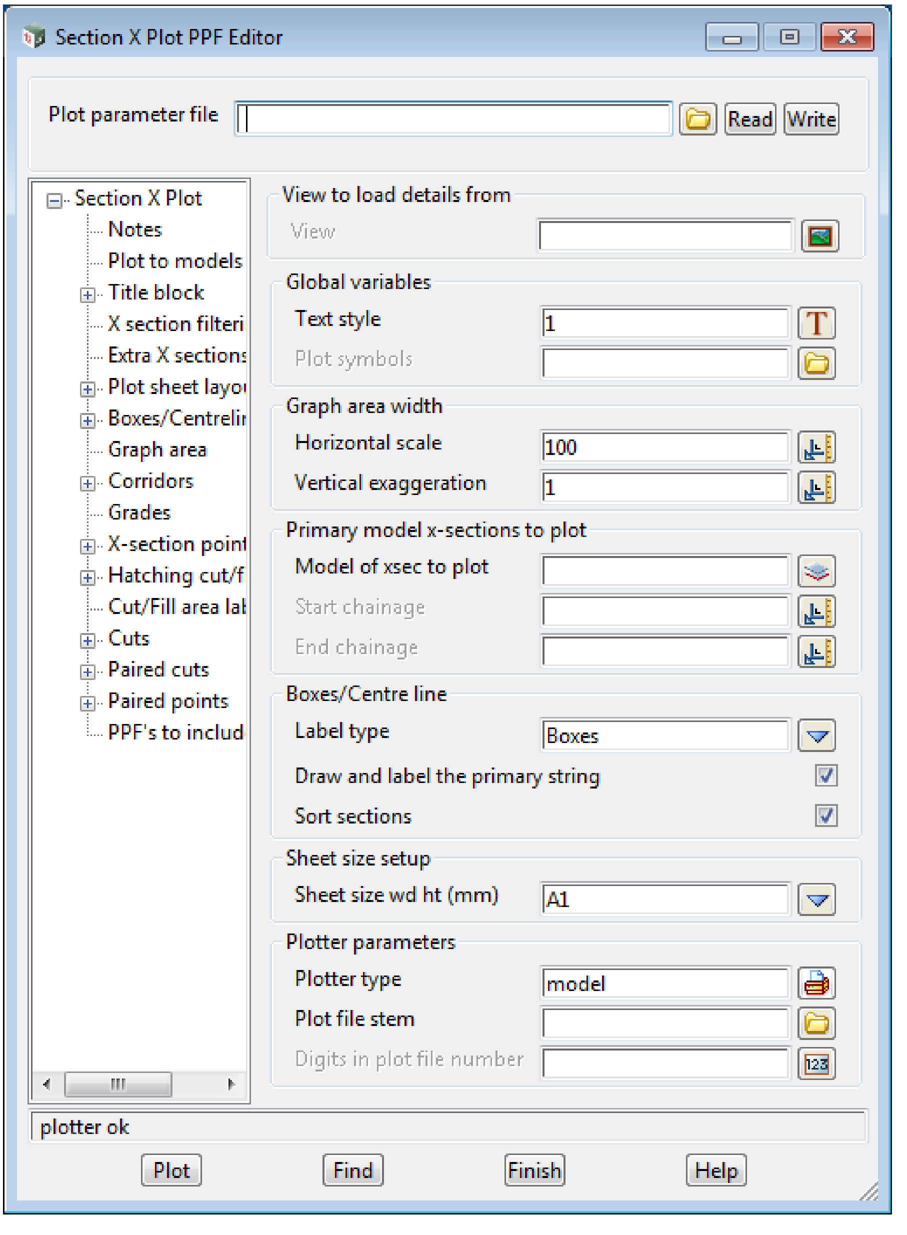
An example of a long section plot created in the ***Getting Started for Design*** manual is:



This plot, and the other three sheets of the set, are generated straight from ***12d Model*** with no manual drafting at all.

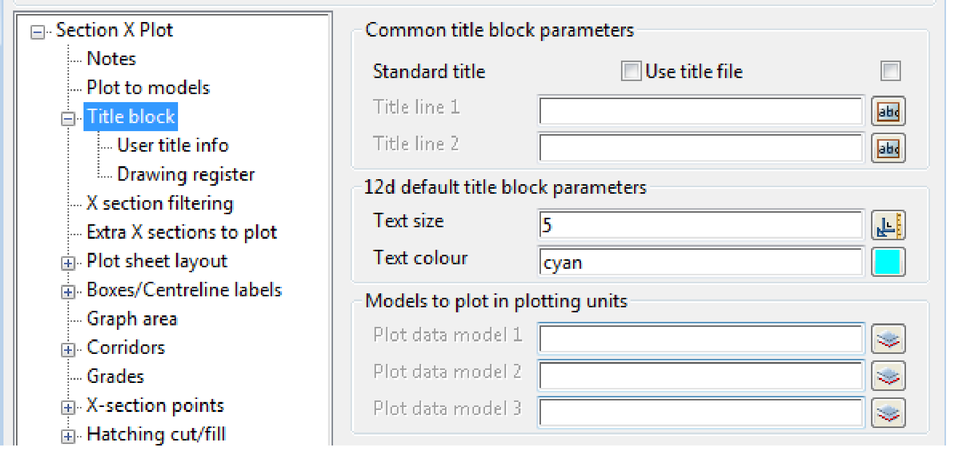
For each of the ***PPF Editors***, the parameters are displayed in a tree structure so they can be grouped together to make it easy to find any parameters to edit and modify.

For example, the **Section X Plot PPF Editor** looks like:



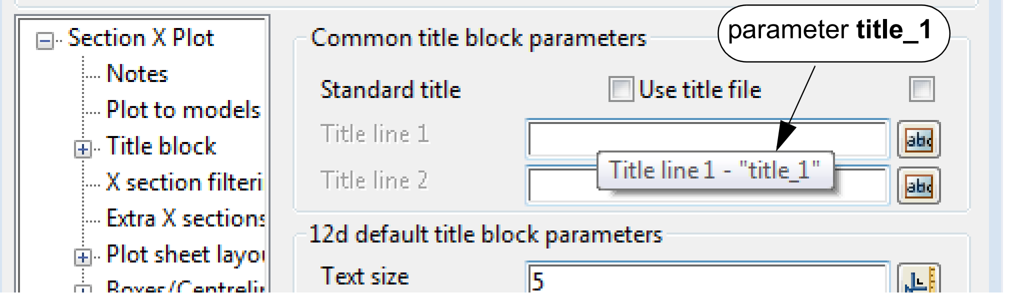
Clicking on a **+** will expand the node and on a **-** will collapse the node.

Clicking on a **node** will display information on the right hand side of the panel. This is where all the parameter values are entered.



Every plot parameter actually has a name but it is rare that will ever need to know it.

For most parameters you can see what the actual parameter name is by hovering over the parameter itself - a description and the name of the parameter is then displayed:



For the user, the parameter names usually only seen in diagrams explaining what the parameters control but when a plot is generated, the PPF Editor file is actually converted to a readable text file with the parameters and there values in it, and that text file that is used for generating the plot.

This all occurs in the background it is useful to know because the PPF Editor files are often referred to as **binary PPF**s and the text ones as **Ascii PPF**s and there is even a converter to go from the Ascii PPF to a Binary PPF.

However throughout this manual, PPF will refer to the Binary PPF's unless stated otherwise.

Note: Binary PPFs are normally stored *within* a project.

Many of the section of the ***PPF Editors*** are common and as much as possible, will be described in the one place.

For the items common to all the PPF Editors that are documented in the section 26.2 Nodes Common to the PPF Editors, see

26.2.1 Plot Parameter File
26.2.2 View to Load and Global Variables
26.2.3 Sheet Size and Plotter Parameters
26.2.4 Notes
26.2.5 Plot to models
26.2.6 Title Block
26.2.10 PPFs To Include
26.2.11 Buttons at the Bottom of the PPF Editors

For the individual PPF Editors, see

The first section of the documentation is 26.2 Nodes Common to the PPF Editors.

26.2 Nodes Common to the PPF Editors

For the items common to all the PPF Editors, see

26.2.1 Plot Parameter File
26.2.2 View to Load and Global Variables
26.2.3 Sheet Size and Plotter Parameters
26.2.4 Notes
26.2.5 Plot to models
26.2.6 Title Block
26.2.10 PPFs To Include
26.2.11 Buttons at the Bottom of the PPF Editors

26.2.1 Plot Parameter File

The Plot Parameter file is common to all the PPF Editors and reads and writes (binary) plot parameter files for all values on the PPF Editor.

Panel field **Type**

Plot parameter file file box

name for the binary plot parameter file to read in or write out.

If a name is selected the Read button still needs to be pressed to read the file in and load up the plot parameters in the PPF Editor.

Read button

read the given plot parameter file in and load up the plot parameters in the PPF Editor from any values in the file.

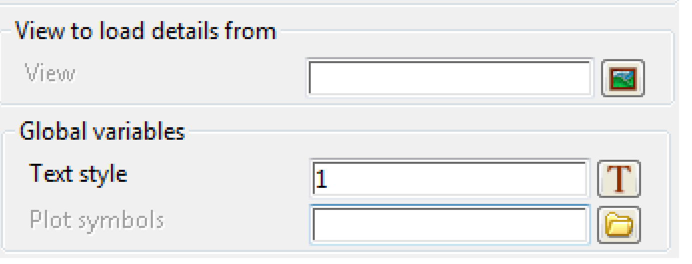
Write button

write out the plot parameters to the given plot parameter file.
A text ppf file is also written out.

Continue to the next section 26.2.2 View to Load and Global Variables or return to 26 PPF Editors.

26.2.2 View to Load and Global Variables

The following fields are common to the top node for most of the PPF Editors.



The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: View to load details from

View **section view box** **all section views**

on selection of an existing section view, the vertical exaggeration, model of x-sections to plot, corridor model and corridor settings from the section view are loaded into the PPF editor.

Section: Global variables

Text style global_textstyle text box

default textstyle to use if there is no parameter given for an optional text style parameter.

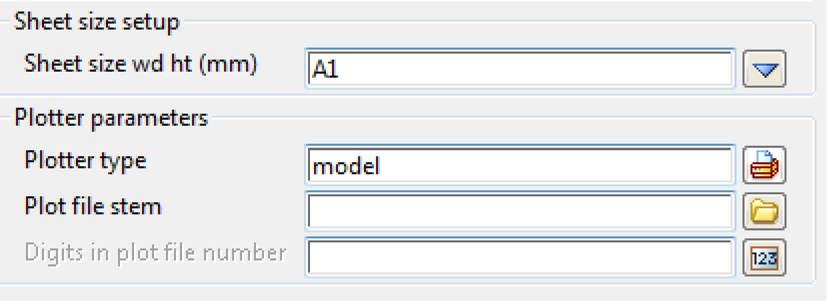
Plot symbols plot_symbols file box

if not blank, this file of plot symbols is used instead of the symbol file currently being used for the project.

Continue to the next section 26.2.3 Sheet Size and Plotter Parameters or return to 26 PPF Editors.

26.2.3 Sheet Size and Plotter Parameters

The following fields are common to the top node for most of the PPF Editors.



The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Sheet size setup

Sheet size wd ht (mm) sheet_size sheet size box

a valid sheet size is selected.

Section: Plotter parameters

Plotter type plotter_type plotter box

a valid plotting option is selected.

Plot file stem plot_stem plotter box

plot file name. The appropriate extension is added depending on the plotter type selected.

Digits in plot file number file_num_digits number box

*leaving **Digits in plot file number** blank or setting a value of 1 or less, will result in file names like we've always known them.*

*Setting **Digits in plot file number** to a value of 2 or more, ensures there will be at least that number of digits in the page numbers built into the plot file names, and if zero-padding is used where necessary to get that number of digits.*

*For example, if **Digits in plot file number** is 3 then plot number 7 will have the number 007.*

Continue to the next section 26.2.4 Notes or return to 26 PPF Editors.

26.2.4 Notes

Section: Document your PPF here ...

Region for typing ppf_notes

Type in any notes as documentation for this PPF file.

This information is not plotted.

Continue to the next section 26.2.5 Plot to models or return to 26 PPF Editors.

26.2.5 Plot to models

The ***Plot to models*** node controls whether models are cleaned before the plot is written to them, and whether the plots go to separate models for each page or all into one model.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Options for plotting to models

Clean plot models beforehand plot_model_clean choice box do not clean
prompt for clean
always clean

when plotting to a model, it defines whether to clean (delete the elements in) any resultant plot models that may already exist, before generating the plot(s).

This parameter is only applicable if plotting to a model or models.

Note that if the models are cleaned using this parameter, any non-plot or locked elements found in the models will ***not*** *be cleaned from the models, and the plot job will be cancelled.*

Section: Translate and merge models

Whenever a PPF is set to produce multiple sheets of plot models, the ***Translate and merge plot models*** *group of parameters, offer the ability to produce plot models arranged neatly in a row or column of sheets, optionally merged into the single plot model created for the first sheet.*

Note*: The* ***Origin X*** *and* ***Origin Y*** *parameters are useful when building up larger arrays of plot sheets -- perhaps entire drawing sets -- from multiple PPFs, where each PPF may be specified with a different origin.*

Mode plot_model_translate_mode choice box do not translate
row wise translation
column wise translation

The entire group of parameters is activated by setting the Mode to either "row-wise translation" or "column-wise translation". This determines whether the multiple sheets will appear as a row or a column of sheets. If unspecified, "do not translate or merge" is assumed, and the remaining parameters are ignored

Spacing (mm) plot_model_translate_spacing real box

determines how far to translate each sheet from the previous sheet. Practically, it should be at least as large as the sheet width or height. For an A1 sheet size, 1000 mm works well as a nice round number

Origin X (mm) plot_model_translate_origin_x real box

determines the x-coordinate of the bottom-left corner of the first sheet.

If let blank then 0 mm is assumed.

Origin Y (mm) plot_model_translate_origin_y real box

determines the y-coordinate of the bottom-left corner of the first sheet.

If let blank then 0 mm is assumed.

Merge plot_model_merge tick box

if ***ticked****, all sheets produced by the PPF, will be created in a single model and that model is the plot model created for the first sheet.*

Note: The **Origin X** and **Origin Y** parameters are useful when building up larger arrays of plot sheets -- perhaps entire drawing sets -- from multiple PPFs, where each PPF may be specified with a different origin.

Continue to the next section 26.2.6 Title Block or return to 26 PPF Editors.

26.2.6 Title Block

12d Model plots are considered to be drawn onto a sheet of paper with the origin (0,0) in the bottom left hand corner and with drawing units of millimetres. So when any plot is generated from ***12d Model***, the data is translated and scaled so that it fits onto the positive quadrant with (0,0) in the left hand corner.

This means that linework and text to make up title blocks and other data can be accurately positioned to overlay a ***12d Model*** plot.

For most plots, users can select to have a Standard ***12d*** title block, or use their own Title blocks that have been stored in ***title block files*** (tbf).

The format for the title block information is a subset of the ***12da*** format with the special header of ***title_block*** and the units inside the tbf are considered to be millimetres.

So inside the ***tbf*** file, the set of all title block drawing commands is enclosed within a set of curly brackets **{ }** with the special header

title_block

before the curly brackets.

That is,

```
title_block   {
              title block drawing commands in 12da syntax

}
```

The easiest way to create a ***tbf*** file is to create the title block information is to use ***12d Model*** and draw up the title block in a model, or a view of models, and then write the models out to a ***tbf*** file using the **Create/Edit Title Block File** panel (*Plot =>Plotting setups =>Create/edit title block file* or *Drafting => Create title block file. See* 25.7.1 Create/Edit Title Block File).

Because the units for plots are millimetres, the units for the title block drawing commands are taken to be millimetres**.**

So for the purpose of drawing up a title block in a ***12d Model*** model, the (0,0) for the ***model*** corresponds to the **origin** (0,0) of the plot, and the distances are taken to represent millimetres so you do everything in World units and multiply all the coordinates and distances as the millimetre value multiplied by 1000.

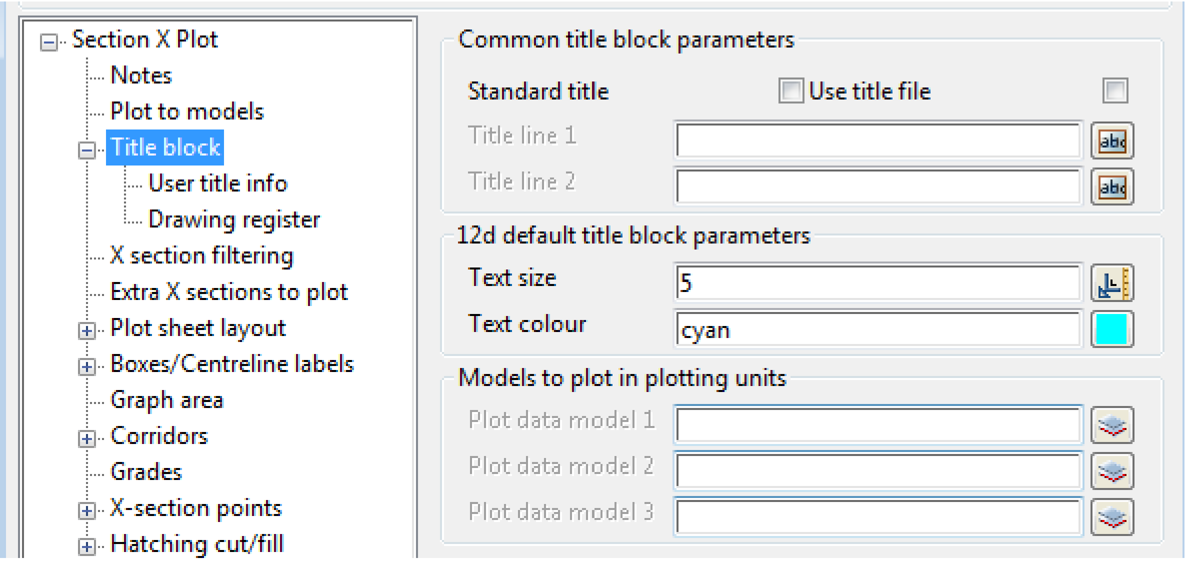
So if text is to be 5 mm high in the title block, then draw text in world units and of size 5. Do not use paper size. If a line is to be 200 mm long, then draw it as 200 World Units.

So for any lines and text drawn up in a model as part of a title block, think of the World units as representing millimetres rather than metres.

Continue to the next section 26.2.6.1 Title Block Section in PPF Editors or return to 26 PPF Editors.

26.2.6.1 Title Block Section in PPF Editors

Each of the **PPF Editors** has a **Title Block** section which is the same for each of them.



See:

26.2.6.1.1 Title Block Node
26.2.6.1.3 Title block - User title info
26.2.6.1.4 Title Block - Drawing Register

For general information about a title block, see 26.2.6 Title Block.

26.2.6.1.1 Title Block Node

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Common title block parameters

Standard title	plot_border	tick box	

if ticked, a standard 12d title block will be used.

Use title file	use_title_file	tick box	

if ticked, a user defined title file is used.

Title line 1	title_1	text box	

if **Standard title** *is ticked,* **Title line 1** *is the first line of title text. If* **Use title file** *is ticked,* **Title line 1** *is substituted for the title block variable* $title_1.

Title line 2	title_2	text box	

if **Standard title** *is ticked,* **Title line 2** *is the second line of title text. If* **Use title file** *is ticked,* **Title line 2** *is substituted for the title block variable* $title_2.

Section: 12d default title block parameters

Text size	title_text_size	measure box	

text size of title text.

Text colour	title_colour	colour box	available colours

colour of title text.

Section: Model to plot in plotter units

Plot data model 1, 2, 3	plot_model	model box	available models

if not blank, the models are plotted as part of the plot.

The Units for the **Plot data model** *are to be in World Units but each unit* ***is taken to represent a millimetre*** *when plotted. So a height of* ***5*** *represents 5mm on the plot. (0,0) in the* **Plot data model** *corresponds to (0,0) in the plot. Paper Units should not be used in Plot data models.*

For more information on plot data models, see 26.2.6.1.2 Plot Data Models.

Continue to:

26.2.6.1.3 Title block - User title info
26.2.6.1.4 Title Block - Drawing Register

Or return to 26.2.6 Title Block or 26 PPF Editors.

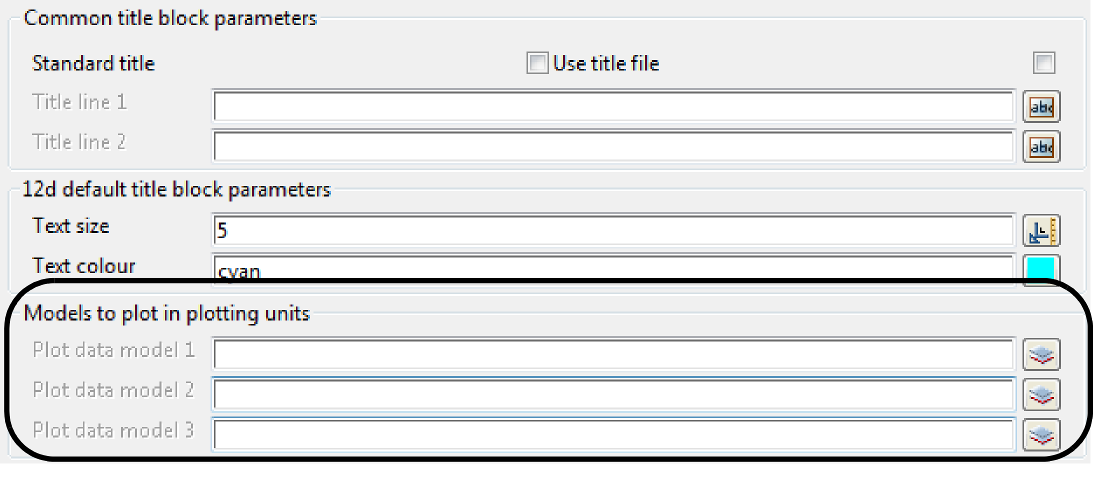
26.2.6.1.2 Plot Data Models

A sheet of paper on a plotter has (0,0) in the bottom left hand corner and the units of millimetres.

So when any plot (plan, plot frame, long section, cross section, drainage network longsection, Melbourne Water longsection or pipeline longsection) is generated from ***12d Model***, the data is effectively translated and scaled so that it fits onto a sheet of paper with (0,0) in the left hand corner, and the drawing units are millimetres.

If a title block is drawn, either using either the *Standard Title* or taking the title drawing information from a *User Defined Title File*, the title block is also drawn onto the sheet of paper with the drawing units of millimetres.

In each of the ***Title Block*** nodes in the plot ***PPF Editors***, models can be given in the ***Plot data model*** fields and these models are also drawn onto the plot sheet.



Because of the special nature of the plot sheet, the Origin (0,0) of a ***Plot data model*** is matched to the origin of the plot sheet, and the ***World Units*** in the ***Plot data model*** are taken to be ***millimetres*** on the plot sheet. This applies to strings, text, linestyles and symbols in World units. So a height of **5** world units represents 5mm on the plot. (0,0).

For text, linestyles and symbols defined in ***Paper units*** (millimetres), one Paper unit in a ***Plot data model*** still ends up as one millimetre on the plot.

There is also ***no clipping*** done when plotting the ***Plot data model.***

So like the Title Block, the ***Plot data models*** are simply an overlay that is plotted on top of the paper plan/long section/x section plot that has just been generated.

Special Notes

1. If the ***Plot data model*** is added to a *Plan View* (for creating, editing *etc*), then the ***Plotting scale*** for the *Plan view* needs to be ***1000*** so that any text, linestyles and symbols defined in *Paper Units* are drawn at the correct size to in relation to the other *World Units* data in the model, and hence as they would appear on a final plot.
2. When ***plotting to a model:***

(a) for all the data ***other than*** the ***Plot Data Models, 12d Model*** multiplies the millimetre size by 1000 and all text, linestyles and symbols are converted to World Units. So that lengths (distances) in the "model of the plot" represent millimetres on the plot page.

Hence if you plot the "model of the plot", you need to plot with a ***scale of 1:1000*** to get the correct size on the paper plot.

(b) for the ***Plot Data Models***, a copy of the ***Plot data model*** is simply added to the model containing the plot

Because the ***Plot data model*** is simply an overlay with no change of scale and is not clipped, then in the special case of ***plotting to a model***, a copy of the ***Plot data model*** is simply added to the model containing the plot. So World text, linestyles and symbols stay as World text, linestyles and symbols. And text, linestyles and symbols in Paper units stay as text, linestyles and symbols in Paper units.

3. When you ***plot to a model*** and then display the "model of the plot" on a *Plan View*, the ***Plotting scale*** for the *Plan View* needs to be **1000** so that text, linestyles and symbols that have come from the ***Plot data model*** and are defined in *Paper Units* display as they would appear in a final paper plot.
4. If you ***plot to a model***, and then add the model of the plot to a Plan View and plot the "model of the plot", ***all*** the linestyles and symbols will be exploded into lines and arcs.

When creating data in a ***Plot data model***, to save confusion it is easiest to create all the data, text, linestyles and symbols in World units, and to think of the World unit as being millimetres.

However many ***12d Model*** users have many of their linestyles and symbols defined in Paper units and then can be used in the ***Plot data model*** as long as the ***Special Notes*** are remembered.

Return to 26.2.6.1.1 Title Block Node.

26.2.6.1.3 Title block - User title info

The parameters shown below are subject to the appropriate title block variables existing in the title file. For more information on these variables see the section 26.2.6.2 Title Block Variables.

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: User title block parameters

Title file title_file file box available title files

specifies the name of the title file to use. If a valid title file exists, the specified **Name** *values will be filled out in the grid using the title block variable* $user_text_n.

Name output

the alias for the nth user text specified in the title file.

Value user_text_n text box

the alias text to be substituted in for the user text specified.

Time format time_format text box

the time format relates to the $time *title block variable. For more information on the time formats see the section* 26.2.6.6 Specifying the Format for $time.

Start page number start_page_number integer box

used as the starting value for the title block variable $page_number.

If missing, $page_number *starts at 1.*

Start drawing number start_drawing_number integer box

used as the starting value for the title block variable $drawing_number. *If missing,* $drawing_number *starts at 1.*

Drawing number prefix drawing_number_prefix text box

the value entered is used for the title block variable $drawing_number_prefix.

Drawing number postfix drawing_number_postfix text box

the value entered is used for the title block variable $drawing_number_postfix.

Continue to 26.2.6.1.4 Title Block - Drawing Register or return to 26.2.6.1 Title Block Section in PPF Editors or 26 PPF Editors.

26.2.6.1.4 Title Block - Drawing Register

These parameters are for using a **Drawing Register** to be used to provide information for the title block. More detailed information is given below after defining the parameters on the node.

The columns for the fields documented in the sections are for.

Panel Field | **Parameter name** | **Type** | **Pop-Up**

Section: Drawing register parameters

Register type | drawing_register_type | choice box | Excel

the type of file that is being used for the drawing register.

Register file | drawing_register_file | file box

name of the register file.

Worksheet | drawing_register_excel_sheet | text box

*if **not blank** and the* **Register type** *is **Excel**, name of the register file, the name of the Worksheet of the spread sheet to use.*

Grid

name of the register file.

Name | text grid cell | text entry

name of the register file.

Value | text grid cell | text entry

name of the register file.

Many people use database tables, such as an Excel spreadsheet, to provide information for the title blocks of the plots. For example, for special drawing numbers and drawing revision. This database is often known as a ***Drawing Register***.

For example, in an Excel spreadsheet, the columns of the spreadsheet would have unique names to refer to the information in each column, and each row of the spreadsheet holds the information for a single, specific plot.

	A	B	C	D
1	PLOT FILE	DRG#	TITLE	REV
2	plot_xs_MC01_1	12345678-5-1	CROSS SECTIONS MC01	C
3	plot_xs_MC01_2	12345678-5-2	CROSS SECTIONS MC01	C
4	plot_xs_MC01_3	12345678-5-3	CROSS SECTIONS MC01	C
5	plot_xs_MC02_1	12345678-5-4	CROSS SECTIONS MC02	C
6	plot_xs_MC02_2	12345678-5-5	CROSS SECTIONS MC02	C
7	plot_ls_MC01_1	12345678-6-1	LONGITUDINAL SECTION MC01	C
8	plot_ls_MC01_2	12345678-6-2	LONGITUDINAL SECTION MC01	C
9	plot_ls_MC02_1	12345678-6-3	LONGITUDINAL SECTION MC02	C
10	plot_dr_ls_1	12345678-7-1	DRAINAGE LONGITUDINAL SECTIONS	C
11	plot_dr_ls_2	12345678-7-2	DRAINAGE LONGITUDINAL SECTIONS	C
12	plot_dr_ls_3	12345678-7-3	DRAINAGE LONGITUDINAL SECTIONS	C
13				
14				

information to use for the one plot

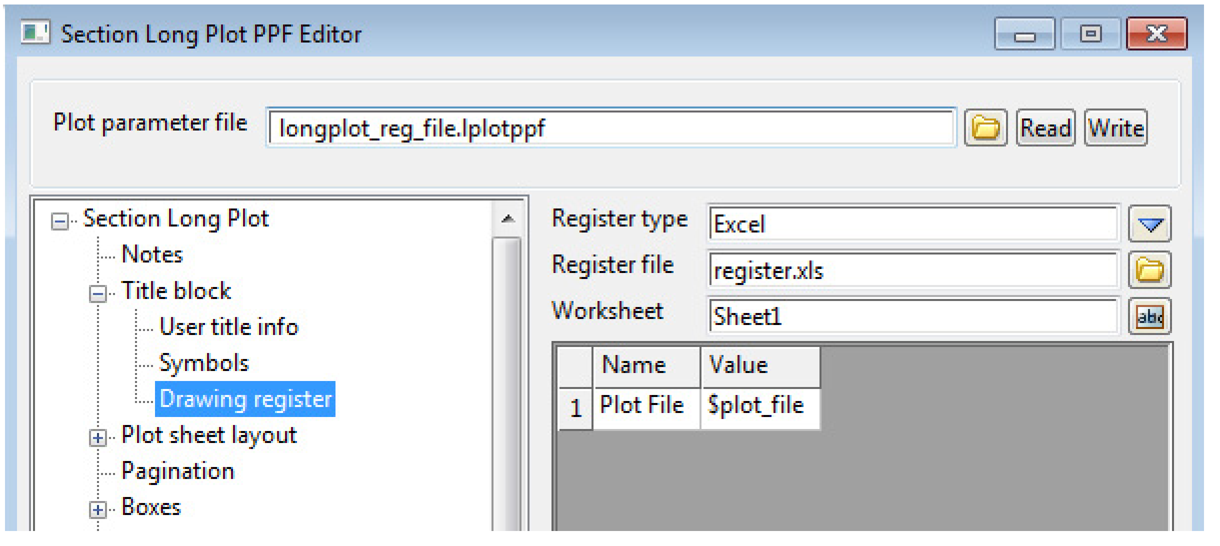
The **Drawing Register** node in the ***PPF Editor*** specifies the type and name of the database to be used to retrieve information from, and enough information to determine what row of the database table is to be used to supply information for the title block of a specific plot.

For example in the spreadsheet above, the *plot file name* will identify a particular plot, and the information in each of the columns from the rest of the row is the drawing number (**DRG#)**, Title (**TITLE**) and Revision (**REV**) for that plot.

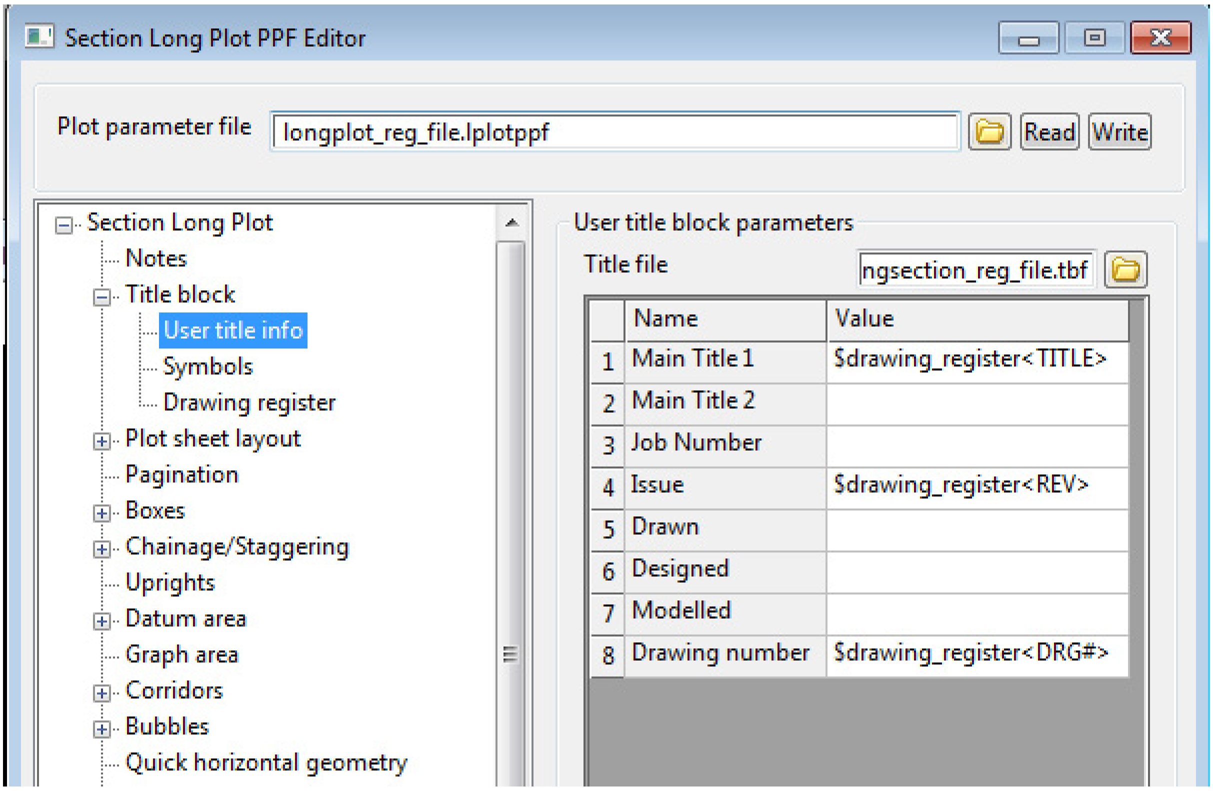
Once the ***Drawing Register*** node has been set up to identify a row of the spreadsheet, the title block variable to extract the information for column ***column_name*** from that row of the drawing register is

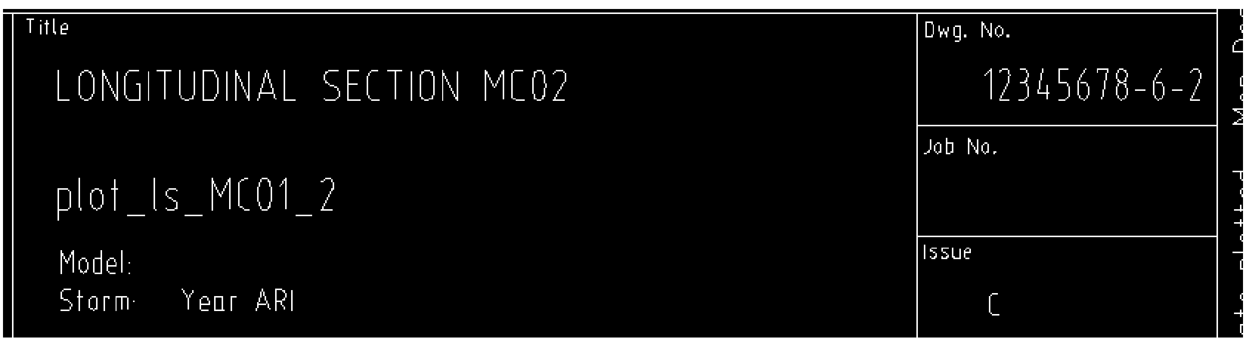
$drawing_register<column_name>

So the ***Drawing register*** node of a *Long Section Plot PPF* could be



and this could be used in a **User tile info** node as:





Continue to 26.2.6.2 Title Block Variables or return to 26.2.6.1 Title Block Section in PPF Editors or 26 PPF Editors.

26.2.6.2 Title Block Variables

For title blocks, the **text** drawing command has been extended so that they can use special information such as the project name, project details, current date and time, plotting scales *etc*. can be automatically inserted into the title block at plot time.

To achieve this, special **title block variables** have been defined and wherever these variables appear in a title block **text**, they are expanded to their defined value at the time of plotting.

Hence the title block variables are simply place markers which have text values substituted for them when the title block file is used in a plot.

The title block variables all begin with a **$** and are followed by either another title block variable or a single space. For example, in the 12da string definition, the line is

text “user text ***$variable*** more user text”

or

text “user text ***$variable_1*** more user text ***$variable_2*** more user text ”

For example

text "Project $project"

will write out the word "Project " followed by the name of the project.

Note - colour, style, xfactor, slant, offset *etc*. are all assigned as part of the ***12da*** string definition.

List of Title Block Variables

See:

Project Name
Project Details
Folder Details
Plot Details
Drawing Register Details
PPF Details
Model Details - for the Drainage Long-section Plot only

Project Name

$project // the current project

Project Details

Project details which are entered by the user for each project (and defined by the option **Project =>Details =>Details**) can be included in the title block and are specified as *title block variables* by beginning with ***$project_detail*** and then including the *project detail* ***name*** as part of the $project_detail.

$project_detail_*project_detail_name*

where project_detail_name is the name of the project detail as given in the **Projects Detail Editor** (and not the **Display name** which is shown in the **Setup Projects Details** or **Edit Project Details** panels)

For example

text "Client: $project_detail_Client"

will write out the word "Client: " followed by the text in the project detail with the name "Client".

Folder Details

The full path name of the working folder

$folder

Plot Details

The following title block variables take values which are passed down from the plot itself:

$plot_file	// the current plot file
$scale	// for plot frames, the current scale
$horizontal_scale	// for section plots, the current horizontal scale
$vertical_scale	// for section plots, the current vertical scale
$start_chainage	// for section plots, the start chainage for the plot // sheet
$end_chainage	// for section plots, the end chainage of the plot sheet

Drawing Register Details

$drawing_register<column_name>

When a ***Drawing Register*** node has been set up to identify a row of the spreadsheet, the title block variable **$drawing_register** is used to extract the information for column ***column_name*** from that row of the drawing register.

For information on setting up a ***Drawing Register*** node, see 26.2.6.1.4 Title Block - Drawing Register.

PPF Details

The following title block variables take values which are passed down from the **plot parameter file** for the plot

(plot parameter files are described in more detail in the rest of this manual):

$time

The current date and time. The format for the date and time is given by the plot parameter time_format. See the next section for the description of the format.

$title_1, $title_2

$title_1 takes the value from the panel field **title line 1** from the **section x plot title**, **section long plot title** or **plot frame title** panels, or from the parameter **title_ 1** in the plot parameter file.
If title_1 doesn't exist, then $title_1 is blank.

Similarly for $title_2.

$user_text_n where n=1,2,... 1000

$user_text_n takes the value of the parameter user_text_n in the plot parameter file. If user_n doesn't exist, then $user_text_n is blank.

Aliases can also be defined for the $user_text_n which are used in the PPF editors. See the next section 26.2.6.4 Aliases for $User Text on aliases.

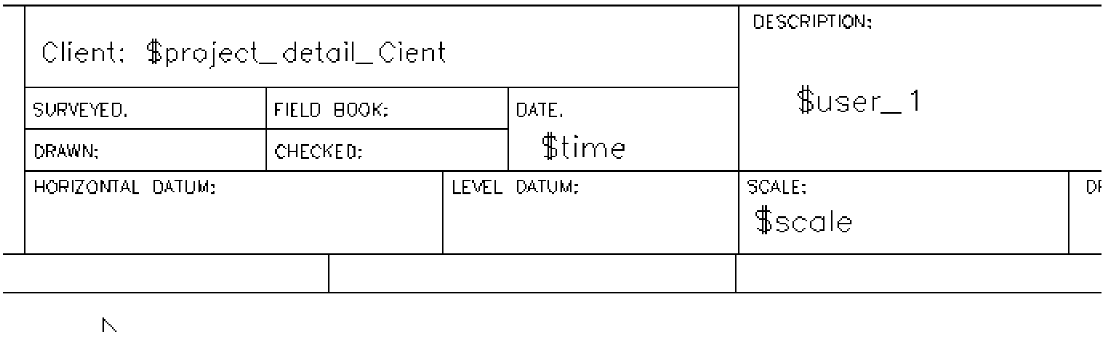
$page_number

$page_number has the starting value one, or the value given by the parameter **start_page_number** from the plot parameter file, and is incremented by one, for each plot produced by the plot option (for example, for each page of a long section plot, for each page of x-section plots or each plot generated from a model of plot frames).

$drawing_number_prefix, $drawing_number, $drawing_number_postfix

The values for $drawing_number_prefix and $drawing_number_postfix are passed down from the plot parameter file by the parameters **drawing_number_prefix** and **drawing_number_postfix** respectively.

$drawing_number has the starting value one, or the value given by the parameter **start_drawing_number** from the plot parameter file, and is incremented by one for each plot produced by the plot option (for example, for each page of a long section plot, for each page of x-section plots or each plot generated from a model of plot frames).



Example of Model with Title Block Variables

Notes

1. For plot frames, the horizontal_scale and vertical_scale are given the current scale, and $start_chainage and $end_chainage are ignored.
2. For section plots, $scale is ignored.

Model Details - for the *Drainage Long-section Plot* only

The following title block variables take values from the model containing the drainage network.

$model_name
The name of the drainage model

$model_event_type
Type of event of last storm analysed ("Minor" or "Major")

$model_return_period
ARI of last storm analysed.

Continue to 26.2.6.3 Ad Hoc Queries or return to 26.2.6 Title Block or 26 PPF Editors.

26.2.6.3 Ad Hoc Queries

It is possible to extract information from some external data sources to use in ***User Title Block*** variables, or directly in the title block file, to place in the title blocks of plots.

Data can be extracted from

(a) Excel spreadsheets

(b) XML files

(c) Databases supporting ODBC

The external data can be accesses in a completely user definable way, known as an **Ad Hoc Queries**.

For each different data source, there is a different method to specify how to identify the bit of information needed from the database.

See

26.2.6.3.1 EXCEL
26.2.6.3.2 XML
26.2.6.3.3 ODBC (Open Database Connectivity)

26.2.6.3.1 EXCEL

The Excel spreadsheet to extract information from **must** have the **first row as a header row** with the names of each column in it so row (n+1) is only the n'th row of actual information.

The format to retrieve the cell X(n+1) from the Sheet ***sheet_n*** is

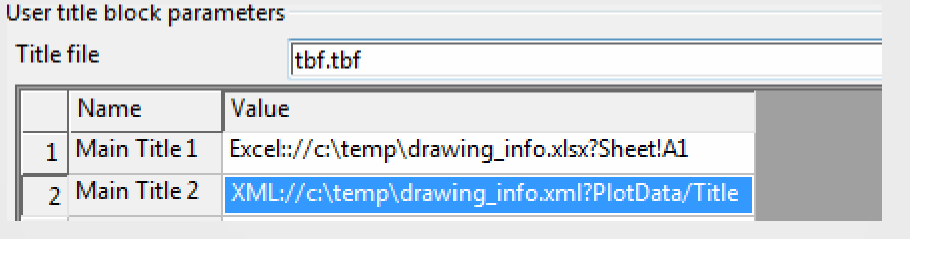
Excel://Full_Path_To_Excel_File?sheet_n?Xn

It is **Xn** is because the first row of the spreadsheet is the header row so Xn is referring to the n'th row of actual information but that is the (n+1) row in the spreadsheet.

For example,

Excel://c:\temp\drawing.info.xslx?Sheet1?A1

will open the Excel file ***c:\temp\drawing.info.xslx*** and get Cell **A2** from the sheet **Sheet 1**.



Return to 26.2.6.3 Ad Hoc Queries

26.2.6.3.2 XML

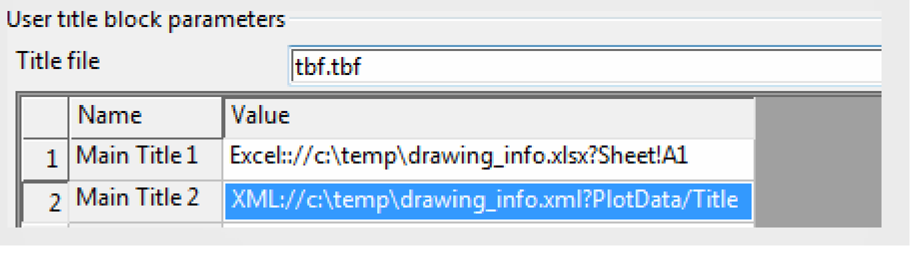
The format to retrieve the value of the element **sub_elt** inside the element **elt** is

XML://Full_Path_To_XML_File?elt/sub_elt

For example,

XML://c:\temp\drawing_info.xml?PlotData/Title

will open the XML file ***c:\temp\drawing_info.xml*** and extract the value of the Title element below the element PlotData.



So from the xml file

<PlotData>

<Title>My Title</Title>

<Client>12d</Client>

etc

</PlotData>

it will get the value "My Title"

Return to 26.2.6.3 Ad Hoc Queries

26.2.6.3.3 ODBC (Open Database Connectivity)

The format to retrieve information from a database is

ODBC://ConnectionString?Query

Unfortunately the format of the **ConnectionString** and the **Query** to pull data from a database is totally dependent on the database system.

But it would resemble something like

ODBC://DSN=MyDBDsn?SELECT Title FROM PlotData WHERE PlotType = ‘long’

ODBC is for advanced users only, and the user must be familiar with the requirements of the end database system.

Return to 26.2.6.3 Ad Hoc Queries

26.2.6.4 Aliases for $User Text

12d Model user defined title blocks can include up to 1,000 markers for placing *user defined text* ($user_text_n where n can be from 1 to 1000) which is only substituted for actual text at plotting time. Because of the difficulty of knowing exactly what *$user_text_n* stood for when the title block is being plotted, the concept of aliases has been introduced.

For example, instead of referring to $user_text_4, an alias can be defined to use the "Engineers name:" instead.

The definition of *$user_text_n* in a title block file has been ***extended*** so that it also includes any ***alias*** for the *$user_text_n.*

To define an alias for the $user_text_n in the title block file, simply replace the

"_n" by "<n, for the alias>"

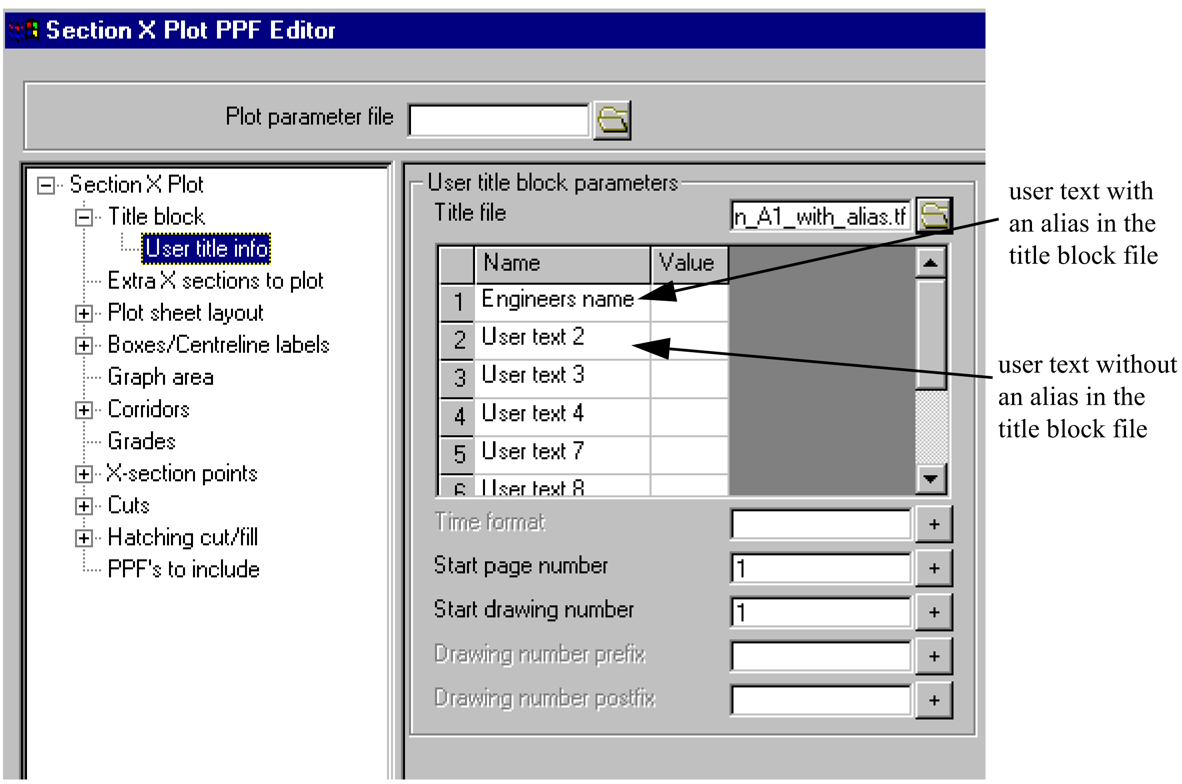
For example, if *$user_text_1* is to have the alias "Engineers name", the title block file would now contain:

text "$user_<1,Engineers name:>"

The title file and aliases work seamlessly with the interactive PPF editors and Plotters (which are described in their own sections). In the *Title block* section of the interactive PPF editors and Plotters, the user defined title block file is *scanned* for any *$user_text* in it and it is presented as a grid containing all the $user_user_text's in the title block with a column to fill in the values that the user wants to be plotted.

If the format *$user_text_n* is in the title block file, it is displayed as *"User text n" in the PPF editor*. If an *alias* exists, then the *alias* is displayed instead of "User text n".

For example, using the *$user_text<1,Engineers name>* in a title block file will give:



Continue to 26.2.6.5 Values Used for Defaults when Creating Plot Frames or return to 26.2.6 Title Block or 26 PPF Editors.

26.2.6.5 Values Used for Defaults when Creating Plot Frames

There are special parameters that are written in an **info block** at the top of title block file that are used as defaults for fields in the option **Plots=>Plot frames=>Create.**

They are

```
info {
   text  "plot_frame_name"              plot_frame_name
   text  "plot_frame_model"             plot_frame_model
   text  "plot_frame_sheet_size"        plot_frame_model
   text  " plot_frame_colour"           plot_frame_colour
   text  " plot_frame_scale"            real_number
   text  "plot_frame_angle"             real_number
   text  "plot_frame_left_margin"       real_number
   text  "plot_frame_right_margin"      real_number
   text  "plot_frame_top_margin"        real_number
   text  "plot_frame_bottom_margin"     real_number
   text  "plot_frame_viewport"          0 or 1
}
```

Continue to 26.2.6.6 Specifying the Format for $time or return to 26.2.6 Title Block or 26 PPF Editors.

26.2.6.6 Specifying the Format for $time

The format for $time is passed down by the plot parameter **time_format**.

time_format *format* // format for $time

The format consists of one or more codes and the formatting codes are preceded by a percent sign (%). Characters that do not begin with a % are copied unchanged.

%a	abbreviated weekday name
%A	full weekday name
%b	abbreviated month name
%B	full month name
%c	date and time representation for locale
%d	day of month as decimal number (01 - 31)
%H	hour in 24-hour format (00 - 23)
%I	hour in 12-hour format (01 - 12)
%j	day of year as decimal number (001 - 366)
%m	month as decimal number (01 - 12)
%M	minute as decimal number (00 - 59)
%p	current locale's A.M./P.M. indicator for 12-hour clock
%S	second as decimal number (00 - 59)
%U	week of year as decimal number, with Sunday as first day of week (00 - 51)
%w	weekday as decimal number (0 - 6; Sunday is 0)
%W	week of year as decimal number, with Monday as first day of week (00 - 51)
%x	date representation for current locale
%X	time representation for current locale
%y	year without century, as decimal number (00 - 99)
%Y	year with century, as decimal number
%z, %Z	time-zone name or abbreviation; no characters if time zone is unknown
%%	percent sign

The # flag may prefix any formatting code and the meaning of the format code is changed as follows

Format Code	Meaning
%#c	long date and time representation, appropriate for current locale. For example, "Tuesday, March 16, 1993,12:41:29"
%#x	long date, appropriate for current locale. For example, "Tuesday, March 16"
%#d, %#H, %#I, %#, %#m, %#M, %#S, %#U, %#W, %#y, %#Y	Remove leading zeros (if any).
%#a, %#A, %#b, %#B, %#p, %#X, %#z, %#Z, %#%	# flag is ignored

Examples

The format to give the date in the form dd/mm/yy (06/09/97) is "%d/%m/%y"

If you want to remove leading zeros from the day and month (6/9/97) "%#d/%#m/%y"

Continue to 26.2.6.7 Example of a Title Block File or return to 26.2.6 Title Block or 26 PPF Editors.

26.2.6.7 Example of a Title Block File

```
title_block {
  name ""
  info {        // Info block for plot frames
    text    "plot_frame_name"   "test 1"
    text    "plot_frame_model"   "test"
    text    "plot_frame_colour"   "blue"
    real    "plot_frame_scale"   1000
    text    "plot_frame_sheet_size"   "A0"
    real    "plot_frame_angle"   0
    real    "plot_frame_left_margin"   5
    real    "plot_frame_right_margin"   5
    real    "plot_frame_top_margin"   5
    real    "plot_frame_bottom_margin"   5
    integer "plot_frame_viewport"   1
  }
// text and line work for title block
  data {

    string text {
      name      "white"
      chainage  0
      breakline line
      colour    black
      style     "1"
      worldsize   1.91800944
      textstyle   "ROMANS"
      angle       0
      x_factor    1
      slant       0
      offset      0
      raise       0
      text_colour black
      justify     "middle-left"
      x         644.91367043
      y         54.28362874
      z         -999
      text "DATE:"
    }

    string text {
      name      "white"
      chainage  0
      breakline line
      colour    black
      style     "1"
      worldsize   3.32407138
      textstyle   "ROMANS"
      angle       0
      x_factor    1
      slant       0
      offset      0
      raise       0
      text_colour black
      justify     "bottom-left"
      x         689.63937012
      y         53.9997359
```

```
  z        -999
  text "$user_1"
}

string text {
  name      "white"
  chainage  0
  breakline line
  colour    black
  style     "1"
  worldsize   1.91800944
  textstyle   "HELV"
  angle       0
  x_factor    1
  slant       0
  offset      0
  raise       0
  text_colour black
  justify     "bottom-left"
  x         504.81297445
  y         42.5174265
  z         -999
  text "WEB: www.12d.com"
}

string text {
  name      "white"
  chainage  0
  breakline line
  colour    black
  style     "1"
  worldsize   3
  textstyle   "ROMANS"
  angle       0
  x_factor    1
  slant       0
  offset      0
  raise       0
  text_colour black
  justify     "middle-left"
  x         568.83213949
  y         63.44111597
  z         -999
  text "Client: $project_detail_Cient"
}

string text {
  name      "white"
  chainage  0
  breakline line
  colour    black
  style     "1"
  worldsize   3.32407138
  textstyle   "ROMANS"
  angle       0
  x_factor    1
  slant       0
  offset      0
```

```
      raise       0
      text_colour black
      justify     "bottom-left"
      x        648.7669706
      y        46.62569771
      z        -999
      text "$time"
    }

    string text {
      name      "white"
      chainage  0
      breakline line
      colour    black
      style     "1"
      worldsize   3.32407138
      textstyle   "ROMANS"
      angle       0
      x_factor    1
      slant       0
      offset      0
      raise       0
      text_colour black
      justify     "bottom-left"
      x        683.00792635
      y        34.87576264
      z        -999
      text "$scale"
    }

  string super {
      name      "white"
      chainage  0
      breakline line
      colour    black
      style     "1"
      closed 0
      data_3d {
        801 594 0
        841 594 0
        841 554 0
      }
    }
  }
```

Return to 26.2.6 Title Block or 26 PPF Editors.

26.2.7 Corridors

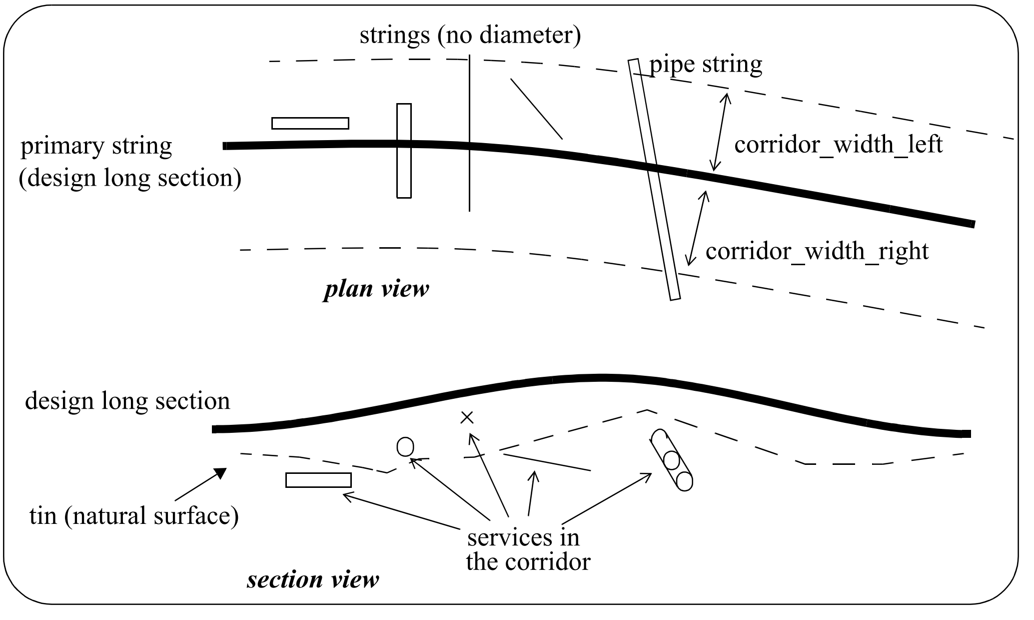
A corridor around the primary string is defined by giving a left and right corridor width.

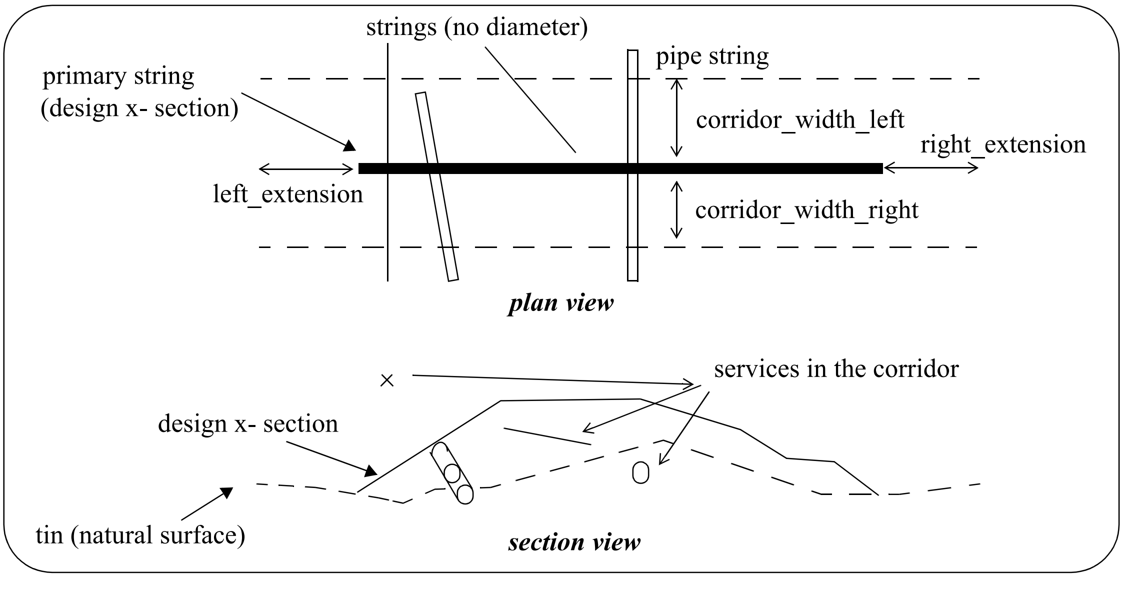
Any string in a model added to the section view is checked to see if it appears in the corridor, and if it does, it is drawn on the long section plot.

To be drawn, strings do not have to cross the primary string, but just be in the corridor.

For a **long section plot**, the primary string is the string being plotted.

For a **x section plot**, the primary string is the string currently being plotted in the x section sub plot.

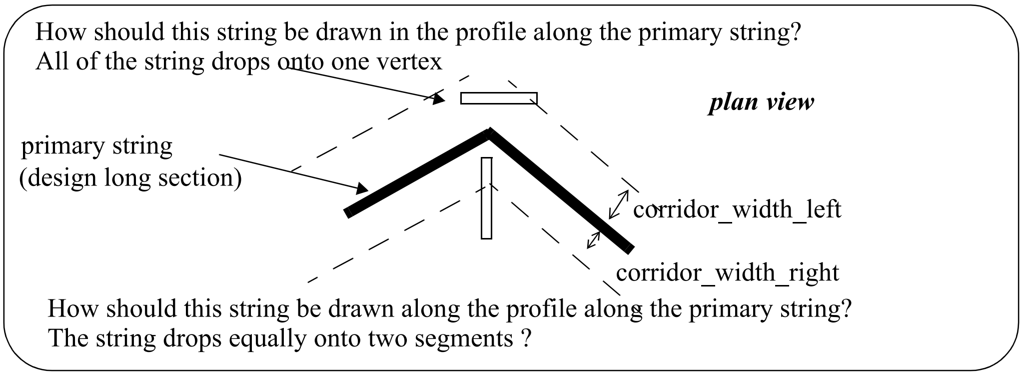




Warnings:

When the primary string is not a straight line in plan, then how the strings in the corridor project onto the primary string in the plot is not always well defined. Where every possible the ends of the string are dropped perpendicularly onto the primary string.

However if the primary string have sharp angles in it in plan, then dropping perpendicular may not be possible and dropping to the closed vertex is all that could be done. This means that many points on a string may drop onto the sam point on the primary string.



The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Corridor parameters

Left corridor width (world units) corridor_width_left real box
left corridor width

Right corridor width (world units) corridor_width_right real box
right corridor width

Left corridor overlap (world units) corridor_overlap_left real box
left corridor overlap

Right corridor overlap (world units) corridor_overlap_right real box
right corridor overlap

Chord-arc tolerance (world units) corridor_chord_arc real box
chord-arc tolerance used near any bends in the corridor.

Corridors - Model Selection

Section: Corridor - Model selection

Corridor model grid

the grid contains the models of tins and strings (service strings) to be in the corridor and hence draw onto the long section, There can be up to 100 models.

For the parameter corridor_model_n, n is the line number on the grid. that is, n = 1, 2, ... ,100.

Corridor model corridor_model_n model box

models containing tins and service strings to be drawn on the section. There can be up to 100 models.

For the parameter corridor_model_n, n is the row number of the cell in the grid.

Continue to the next section 26.2.8 Hatching Cut/Fill or return to 26 PPF Editors.

26.2.8 Hatching Cut/Fill

This option is used to hatch cut and/or fill areas between sets of tins.

For each set, the name of the two tins, the hatch linestyle, colour and separation and whether cut and/or fill regions are required are all user definable.

Up to twenty (20) separate sets of tins may be hatched.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Hatching cut/fill - Tin parameters

Define Set # integer box

set number to be used to define different original/new tin sets.

Original tin hatch_original_tin_n tin box

tin_name for original surface

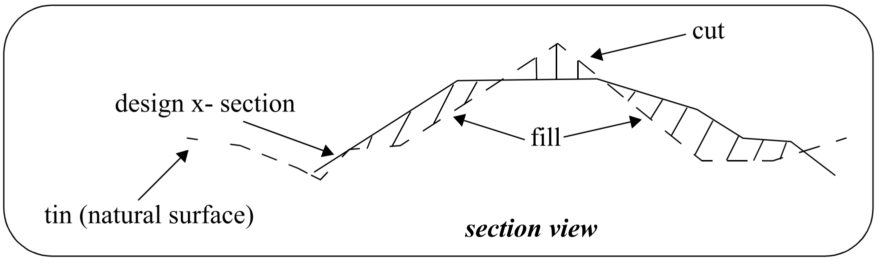
New tin hatch_new_tin_n tin box

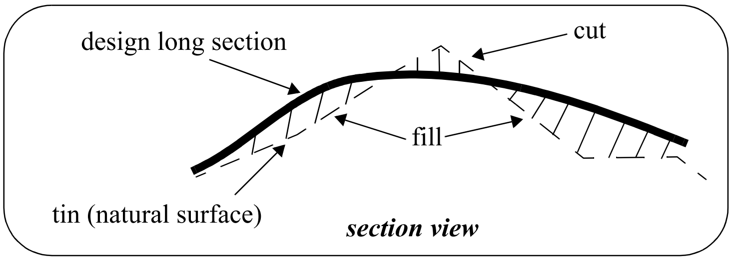
tin_name for final surface

Notes

(a) cut is when the new tin is below the original tin.
fill is when the new tin is above the original tin.

(b) cut hatching is turned off by setting *hatch_cut_separation_n* to 0.0.
fill hatching is turned off by setting *hatch_fill_separation_n* to 0.0.





See

26.2.8.1 Hatching Cut/Fill - Cut
26.2.8.2 Hatching Cut/Fill - Fill

Or return to 26 PPF Editors.

26.2.8.1 Hatching Cut/Fill - Cut

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Hatching cut/fill - Cut parameters

Use Set # **number box**

set number as specified in the **Define set#**.

Cut separation (mm) hatch_cut_separation_n real box

distance between cut hatch lines. If 0, no hatching.

Cut hatch angle (dms) hatch_cut_angle_n real box

angle of hatching.

Cut colour hatch_cut_colour_n colour box

colour of the hatching.

Cut linestyle hatch_cut_linestyle_n linestyle box

linestyle of the hatching.

Draw sides of cuts hatch_cut_draw_sides_n choice box

draw mode for sides of cut regions.

Draw original tin hatch_cut_draw_original_n choice box

draw mode for sides of original tin in cut.

Draw new tin hatch_cut_draw_new_n choice box

draw mode for sides of new tin in cut.

Continue to the next section 26.2.8.2 Hatching Cut/Fill - Fill or return to 26.2.8 Hatching Cut/Fill.

26.2.8.2 Hatching Cut/Fill - Fill

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Hatching cut/fill - Fill parameters

Use Set # integer box

set number as specified in the **Define set#***.*

Fill separation (mm) hatch_fill_separation_n real box

distance between fill hatch lines. If 0, no hatching.

Fill hatch angle (dms) hatch_fill_angle_n angle box

angle of hatching in degrees, minutes and seconds in hp notation.

Fill colour hatch_fill_colour_n colour box

colour of the hatching.

Fill linestyle hatch_fill_linestyle_n linestyle box

linestyle of the hatching.

Draw sides of fills hatch_fill_draw_sides_n choice box

draw mode for sides of fill regions.

Draw original tin hatch_fill_draw_original_n choice box

draw mode for sides of original tin in fill.

Draw new tin hatch_fill_draw_new_n choice box

draw mode for sides of new tin in fill.

Continue to the next section 26.2.9 Cuts or return to 26 PPF Editors.

26.2.9 Cuts

The cuts that the primary string makes though strings in any user-specified model, can be automatically labelled on the x-section/long section plots.

The **height**, **offset/chainage name**, **attributes**, **diameter**, **x** and **y coordinates**, **3d length of CL until cut point** of the cut string can be labelled as well as a **symbol** drawn. The height of tins at the same offset value can also be labelled.

For a **x section plot,** the **offset position** for the labelling is the offset on the primary string where it cuts the string.

For a **long section plot**, the **chainage position** for the labelling is the chainage on the primary string where it cuts the string.

The height position for the labelling can be specified as the:

(a) top of the boxes

(b) above the maximum height of the strings on the plot,

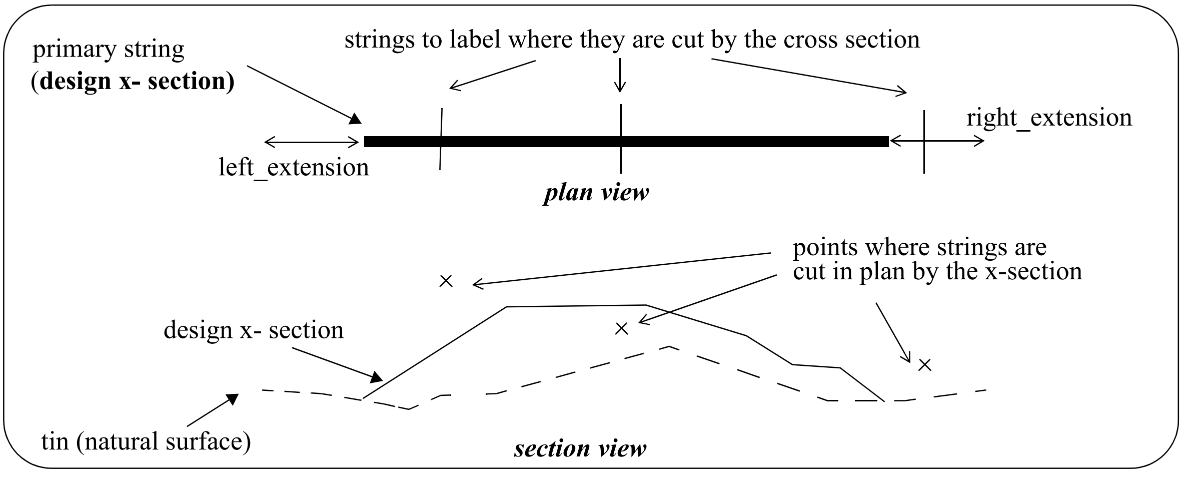
(c) height value of the cut string,

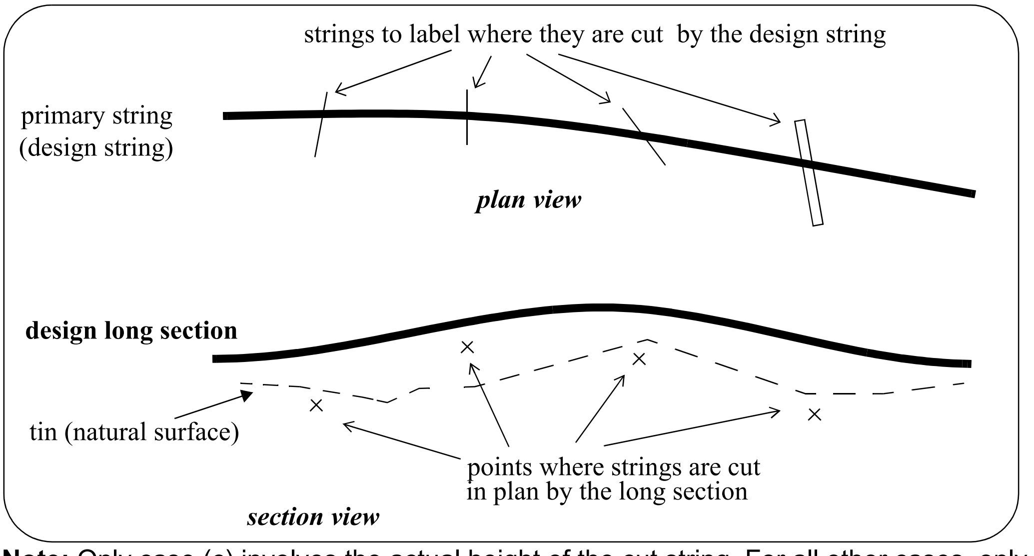
(d) diameter value of the cut string,

(e) height of the primary string,

(f) height of a tin.

The actual position of the label is defined relative to the above point.





Note: Only case (c) involves the actual height of the cut string. For all other cases, only the offset/chainage of the cut string is used. Hence, for all cases except (c), the string does need to have a sensible height to be used for cuts through strings. For example, a boundary string may have null heights but only the offset is required and the height of the tin at that offset can be used as the height (case (e)).

The method for specifying which strings are to be checked for cuts is by first specifying the **model** which contains the strings, and then a **name mask** which is used to restrict the strings in the model to only those whose names match the name mask.

Up to twenty five different sets of models and name masks can be used so that different cut sets can be labelled in different ways.

See

26.2.9.1 Cuts - Model/Name mask parameters
26.2.9.2 Cuts - Offsets - X Section Only
26.2.9.3 Cuts - Chainage - Long Sections Only
26.2.9.5 Cuts - Diameters
26.2.9.6 Cuts - Labels
26.2.9.7 Cuts - Symbols

26.2.9.1 Cuts - Model/Name mask parameters

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Cuts - Model/Name mask parameters

Define Set # *number box*

set number to be used to define different model/mask sets.

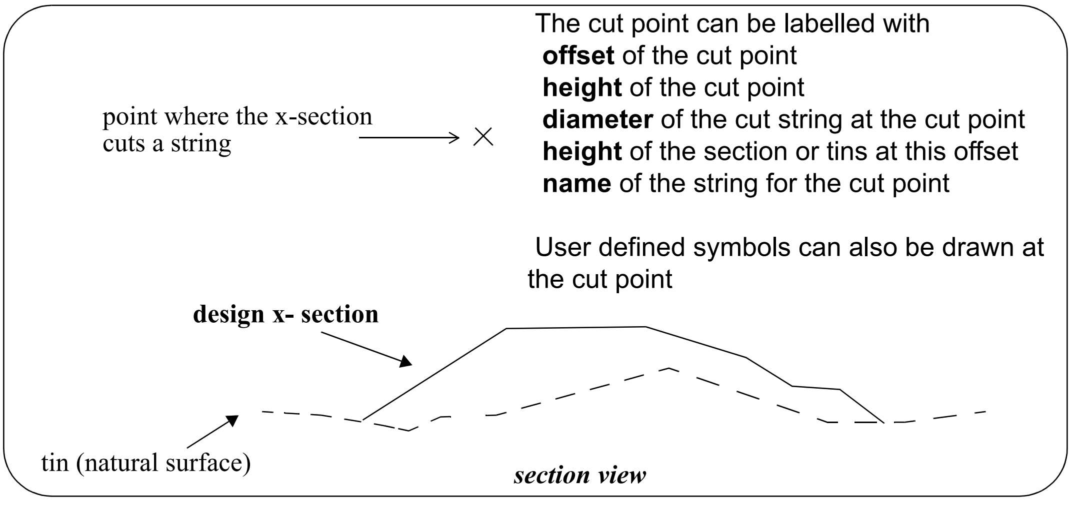
Model cuts_n_model model box

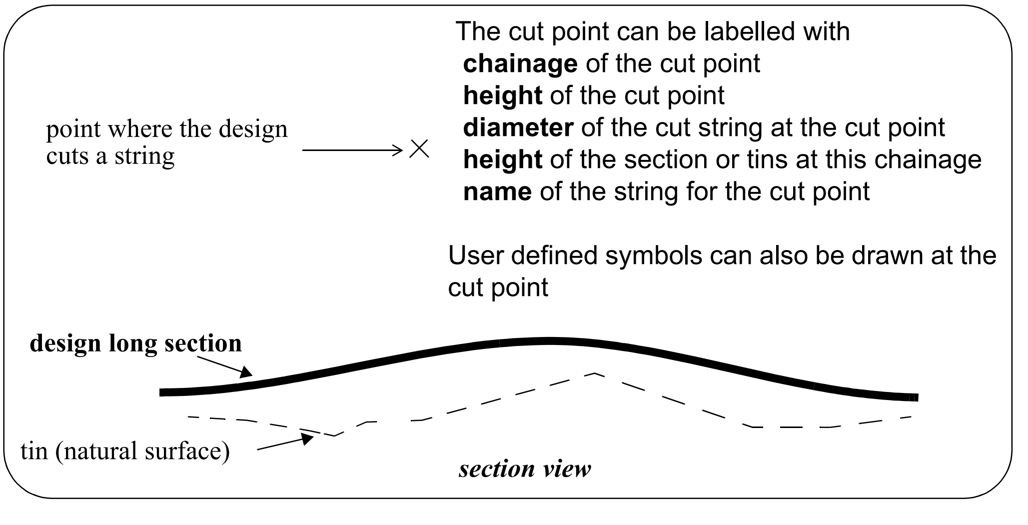
model from which cut masks are derived

Name mask cuts_n_mask text box

text string containing the name masks, each separated by one or more spaces, to test the string name against. Each mask can include wild cards and wild characters.
For example: "ke" or,*
"?bank" or, if both masks are required,*
"ke ?bank*"*
If cuts_n_mask is blank, then all strings in the model are used. This is equivalent to name mask being set to "".*

All strings in the model *cuts_n_model* whose name satisfy the name mask *cuts_n_mask* are then checked for cuts with the x-sections/longsection, and if a cut occurs, the cut point will be labelled according to the rest of the parameters in the nth set.





Continue to 26.2.9.2 Cuts - Offsets - X Section Only or 26.2.9.3 Cuts - Chainage - Long Sections Only or return to 26.2.9 Cuts.

26.2.9.2 Cuts - Offsets - X Section Only

This is for Section X Plots Only

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Cuts - Offset parameters

Use Set #		integer box	

set number as specified in the **Define set#***.*

Position	cuts_offset_n_position	choice box	at cut string height above top of boxes above top of graph area to primary string to tin 1 to tin 2 to tin 3 to tin 4 to tin 5 to tin 6 to tin 7 to tin 8 to tin 9 to tin 10

position of offset label.

X (mm)	cuts_offset_n_x	real box	

horizontal adjustment to position of offset text.

Y (mm)	cuts_offset_n_y	real box	

vertical adjustment to position of offset text.

Angle (dms)	cuts_offset_n_angle	angle box	

rotation of offset text about position in degrees, minutes and seconds in hp notation.

Colour	cuts_offset_n_colour	colour box	

colour of offset text.

Size (mm)	cuts_offset_n_size	real box	

size of offset text. A value of 0 means no label.

Textstyle	cuts_offset_n_textstyle	text box	

textstyle of offset text.

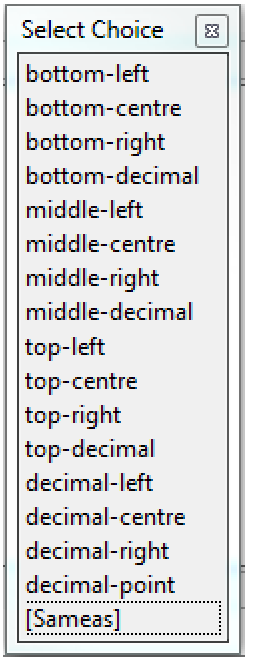
Pre-text	cuts_offset_n_pre_text	text box	

text before offset text.

Post-text	cuts_offset_n_post_text	text box	

text after offset text.

Justification cuts_offset_n_justification justification box



justification of the offset text.

Decimals cuts_offset_n_no_decimals integer box

number of decimals in offset.

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Continue to 26.2.9.3 Cuts - Chainage - Long Sections Only or return to 26.2.9 Cuts.

26.2.9.3 Cuts - Chainage - Long Sections Only

This is for the Long section Plots Only,

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Cuts - Chainage parameters

Use Set # integer box

set number as specified in the **Define set#**.

Position cuts_chainage_n_position choice box above cut string height value
above top of boxes
to primary string
to tin 1
to tin 2
to tin 3
to tin 4
to tin 5
to tin 6
to tin 7
to tin 8
to tin 9
to tin 10

position of chainage label.

X (mm) cuts_chainage_n_x real box

horizontal adjustment to position of chainage text.

Y (mm) cuts_chainage_n_y real box

vertical adjustment to position of chainage text.

Angle (dms) cuts_chainage_n_angle real box

rotation of chainage text about position in degrees, minutes and seconds in hp notation.

Colour cuts_chainage_n_colour colour box

colour of chainage text.

Size (mm) cuts_chainage_n_size real box

size of chainage text. A value of 0 means no label.

Textstyle cuts_chainage_n_textstyle text box

textstyle of chainage text.

Pre-text cuts_chainage_n_pre_text text box

text before chainage text.

Post-text cuts_chainage_n_post_text text box

text after chainage text.

Justification cuts_chainage_n_justification justification box bottom-left
bottom-centre
bottom-right
bottom-decimal
middle-left
middle-centre
middle-right
middle-decimal
top-left
top-centre
top-right

top-decimal
decimal-left
decimal-centre
decimal-right
decimal-point

justification of the chainage text.

Decimals cuts_chainage_n_no_decimals integer box

number of decimals in chainage.

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Continue to 26.2.9.4 Cuts - Heights or return to 26.2.9 Cuts.

26.2.9.4 Cuts - Heights

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Cuts - Height parameters

Panel Field	Parameter name	Type	Pop-Up
Use Set #		integer box	

set number as specified in the **Define set#**.

Panel Field	Parameter name	Type	Pop-Up
Mode	cuts_height_n_mode	choice box	use height of cut point use real height above boxes height of primary string use height of tin 1 use height of tin 2 use height of tin 3 use height of tin 4 use height of tin 5 use height of tin 6 use height of tin 7 use height of tin 8 use height of tin 9 use height of tin 10

determines which height value is labelled.

Panel Field	Parameter name	Type	Pop-Up
Position	cuts_height_n_position	choice box	at cut string above top of boxes above top of graph area to primary string to tin 1 to tin 2 to tin 3 to tin 4 to tin 5 to tin 6 to tin 7 to tin 8 to tin 9 to tin 10

position of height label.

Panel Field	Parameter name	Type	Pop-Up
X (mm)	cuts_height_n_x	real box	

horizontal adjustment to position of height text.

Panel Field	Parameter name	Type	Pop-Up
Y (mm)	cuts_height_n_y	real box	

vertical adjustment to position of height text.

Panel Field	Parameter name	Type	Pop-Up
Angle (dms)	cuts_height_n_angle	angle box	

rotation of height text about position in degrees, minutes and seconds in hp notation.

Panel Field	Parameter name	Type	Pop-Up
Colour	cuts_height_n_colour	colour box	

colour of height text.

Panel Field	Parameter name	Type	Pop-Up
Size (mm)	cuts_height_n_size	real box	

size of height text. A value of 0 means no label.

Panel Field	Parameter name	Type	Pop-Up
Textstyle	cuts_height_n_textstyle	text box	

textstyle of height text.

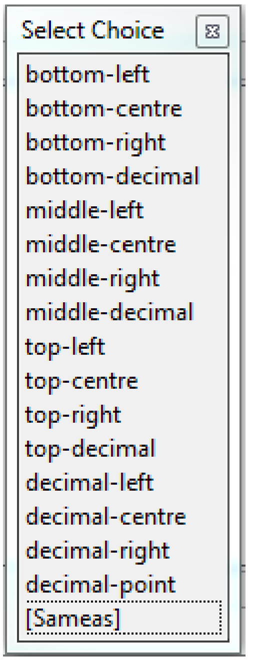
Panel Field	Parameter name	Type	Pop-Up
Pre-text	cuts_height_n_pre_text	text box	

text before height text.

Post-text cuts_height_n_post_text text box

text after height text.

Justification points_height_n_justification justification box



justification of the height text.

Decimals cuts_height_n_no_decimals integer box

number of decimals in height.

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Continue to 26.2.9.5 Cuts - Diameters or return to 26.2.9 Cuts.

26.2.9.5 Cuts - Diameters

The columns for the fields documented in the sections are for.

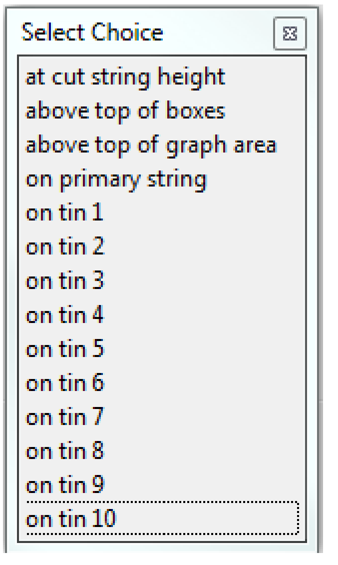
Panel Field **Parameter name** **Type** **Pop-Up**

Section: Cuts - Diameter parameters

Use Set # integer box

set number as specified in the **Define set#***.*

Position cuts_diam_n_position choice box



position of diameter label.

X (mm) cuts_diam_n_x real box

horizontal adjustment to position of text.

Y (mm) cuts_diam_n_y real box

vertical adjustment to position of text.

Angle (dms) cuts_diam_n_angle angle box

rotation of diameter text about position in degrees, minutes and seconds in hp notation.

Colour cuts_diam_n_colour colour box

colour of height text.

Size (mm) cuts_diam_n_size real box

size of diameter text. A value of 0 means no label.

Textstyle cuts_diam_n_textstyle text box

textstyle of diameter text.

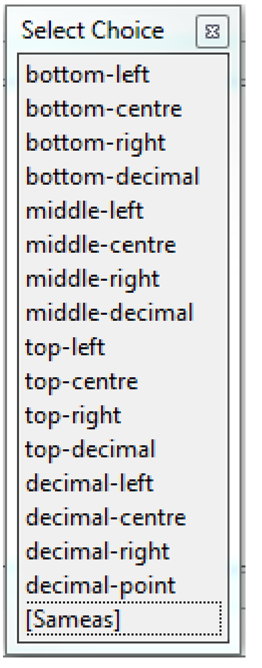
Pre-text cuts_diam_n_pre_text text box

text before diameter text.

Post-text cuts_diam_n_post_text text box

text after diameter text.

Justification points_diam_n_justification justification box



justification of the diameter text.

Factor cuts_diam_n_factor real box

factor to multiply the diameter by. For example 1000 if you want the label to be in millimetres.

Decimals cuts_diam_n_no_decimals integer box

number of decimals in diameter value.

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Continue to 26.2.9.6 Cuts - Labels or return to 26.2.9 Cuts.

26.2.9.6 Cuts - Labels

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop Up**

Section: Cuts - Label parameters

Use Set # integer box

set number as specified in the **Define set#***.*

Position cuts_label_n_position choice box at cut string height
above top of boxes
above top of graph area
to primary string
to tin 1
to tin 2
to tin 3
to tin 4
to tin 5
to tin 6
to tin 7
to tin 8
to tin 9
to tin 10

position of label text.

Mode cuts_label_n_mode choice box don't include string name
include cut string name

determines whether the label includes the cut string name.

X (mm) cuts_label_n_x real box

horizontal adjustment to position of label.

Y (mm) cuts_label_n_y real box

vertical adjustment to position of label.

Angle (dms) cuts_label_n_angle angle box

rotation of label about position in degrees, minutes and secons in hp notation.

Colour cuts_label_n_colour colour box

colour of label.

Size (mm) cuts_label_n_size real box

size of label. A value of 0 means no label.

Textstyle cuts_label_n_textstyle text box

textstyle of label.

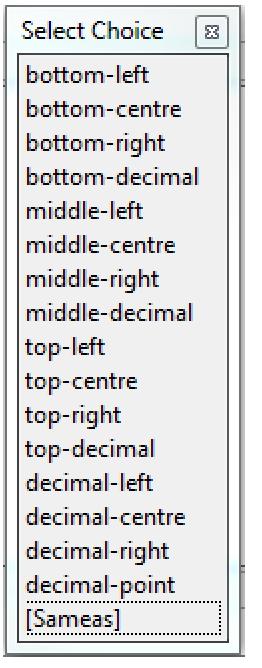
Pre-text cuts_label_n_pre_text text box

text before label.

Post-text cuts_label_n_post_text text box

text after label.

Justification points_label_n_justification justification box



justification of the label.

Continue to 26.2.9.7 Cuts - Symbols or return to 26.2.9 Cuts.

26.2.9.7 Cuts - Symbols

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop Up

Section: Cuts - Symbol parameters

Use Set #		integer box	

set number as specified in the **Define set#**.

Mode	cuts_symbol_n_mode	choice box	cross (0) up from centre of box (1) up&down from box centre(2) square (3) triangle, base at bottom (4) circle (5) use a plot symbol

symbol mode.

predefined symbols 0 to 5

Note: *If a plot symbol is to be used, the* cuts_symbol_n_style *parameter must be specified.*

Symbol	cuts_symbol_n_style	plot symbols	

a valid plot symbol can be selected.

Position	cuts_symbol_n_position	choice box	at cut string height above top of boxes above top of graph area to primary string to tin 1 to tin 2 to tin 3 to tin 4 to tin 5 to tin 6 to tin 7 to tin 8 to tin 9 to tin 10

position of symbol.

X (mm)	cuts_symbol_n_x	real box	

horizontal adjustment to position of symbol.

Y (mm)	cuts_symbol_n_y	real box	

vertical adjustment to position of symbol.

Angle (dms)	cuts_symbol_n_angle	input	

rotation of symbol about point in degrees, minutes and secons in hp notation.

Colour	cuts_symbol_n_colour	colour box	

colour of symbol.

Size (mm)	cuts_symbol_n_size	real box	

size of symbol. A value of 0 means no symbol.

Return to 26.2.9 Cuts.

26.2.10 PPFs To Include

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up
PPF files		file box	

by including pre-existing PPF files, the user can build up a modified version without having to set all the parameters.

Continue to the next section 26.2.11 Buttons at the Bottom of the PPF Editors or return to 26 PPF Editors.

26.2.11 Buttons at the Bottom of the PPF Editors

PPF_Editor_Find_ParameterThe columns for the fields documented in the sections are for.

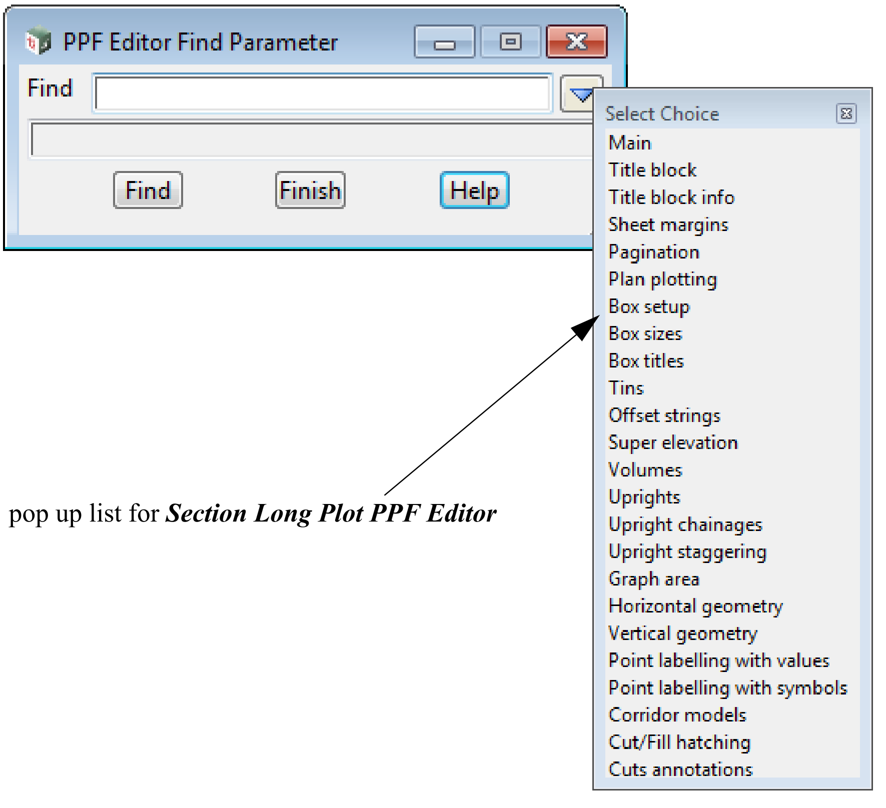
Panel Field **Parameter name** **Type** **Pop-Up**

Plot button

use the plot parameters from the panel to create the plot.

Find button

when clicked it brings up the **PPF Editor Find Parameter** *panel.*



pop up list for ***Section Long Plot PPF Editor***

If a valid plot parameter name is keyed in followed by the ***Find*** *button, the cursor will be placed in the appropriate field on the appropriate page.*

If you click on the ***Choice*** *icon, then a list of regularly used selections that are built into the choice box will appear (these are different for each PPF Editor). Double clicking on a choice in the list will take you to the relevant section of the tree and the relevant parameter on the right side of the panel.*

Return to 26 PPF Editors.

26.3 Section X Plot PPF Editor

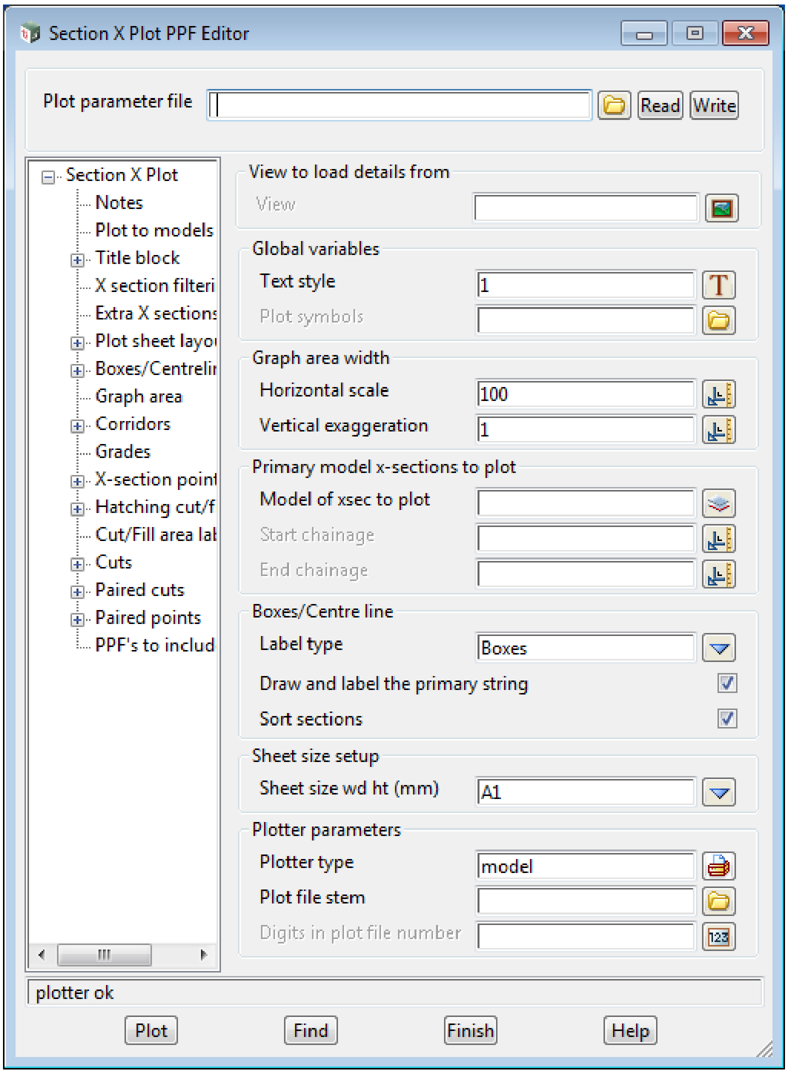
Position of option on menu: **Plot =>Plot and PPF Editors =>Cross sections**

The **Cross Sections PPF Editor** is for creating and/or editing a (binary) X-section PPF and for creating a cross section plot.

Note: Binary PPFs are stored *within* the project (not in the folder containing the project as the text PPFs were).

On selecting the **Cross sections** option, the **Section X Plot PPF Editor** panel is displayed.

The plot parameters for controlling the cross section plots are accessed by expanding the appropriate node in the ***Section X Plot*** tree (click on the **+** to expand to node or **-** to collapse the node) and then clicking on the required node, and the information to fill in is displayed on the right hand side of the panel.



For information on all the different nodes see:

26.3.1 General Information on Cross Section Plots
26.3.2 Section X Plot - Front Page
26.3.3 Notes - X-Section
26.3.5 Title Block - X Section
26.3.6 X Section Filtering

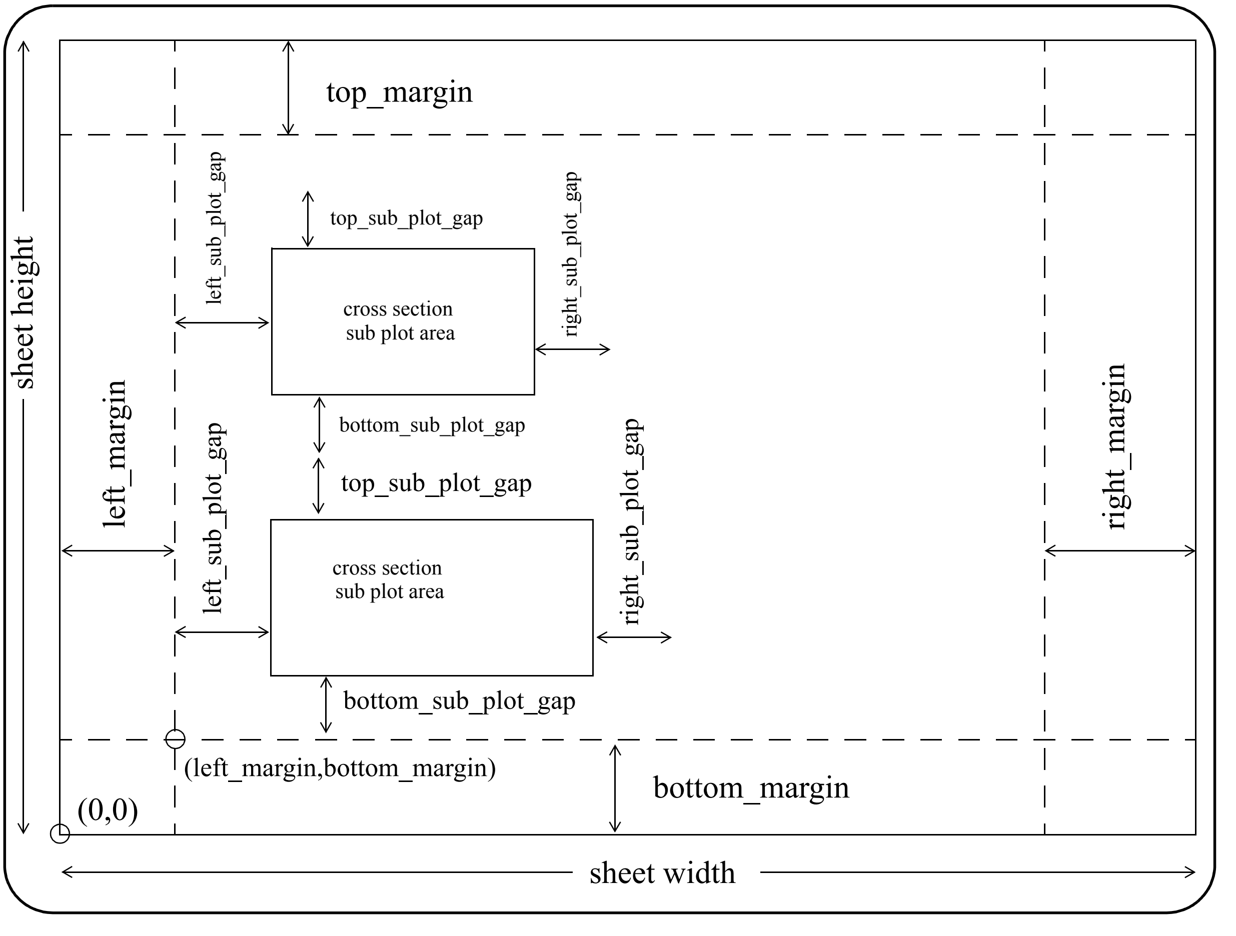
26.3.1 General Information on Cross Section Plots

X-sections are normally generated at chainages along a given centreline and this centreline chainage is stored as part of the name of each x-section string.

The start chainage of the x-section string is set so that zero chainage is where the Centre line cuts the x-section in plan. So the chainage along a x-section string is usually referred to as **offsets** from the centreline position rather than x-section chainages.

The x-sections along the centreline are stored in the one model (the primary model) which is then used to generate the cross section plot.

Each x-section from the primary model of x-sections generates its own **sub-plot** for which a x-section is the primary string. Hence the cross section plot consists of many individual plots drawn on one or more plot sheets.



The x-sections are plotted in the order they occur in the x-section model or can be sorted by the centreline chainage that is part of the x-section name.

The x-sections start being plotted at the bottom left hand corner of the cross section plotting area. The individual x-section plots are then drawn going up the column, and when the column is full, start from the bottom of the next column.

When a sheet is full, a follow on sheet is created.

If required, all the sub-plots in a column can be automatically positioned up so that the zero offsets (the centrelines) of each x-section line up.

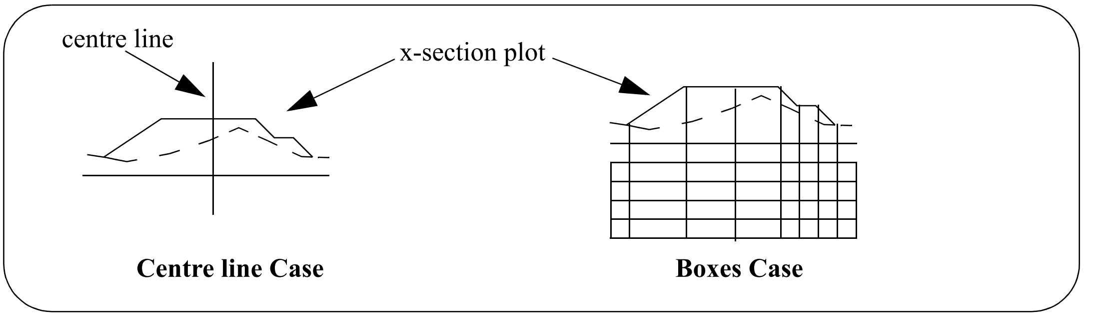
The x-section sub plots can be drawn and labelled with the two choices of:

(a) centreline case

where the x-section and tins are plotted and an upright at the zero offset, and labels for the offset and height values at the zero offset (normally the centre line position)

(b) **boxes case**

where the x-section is plotted and the heights of the x-section and the tins at all the x-section points are labelled in boxes under the plot of the x-section.

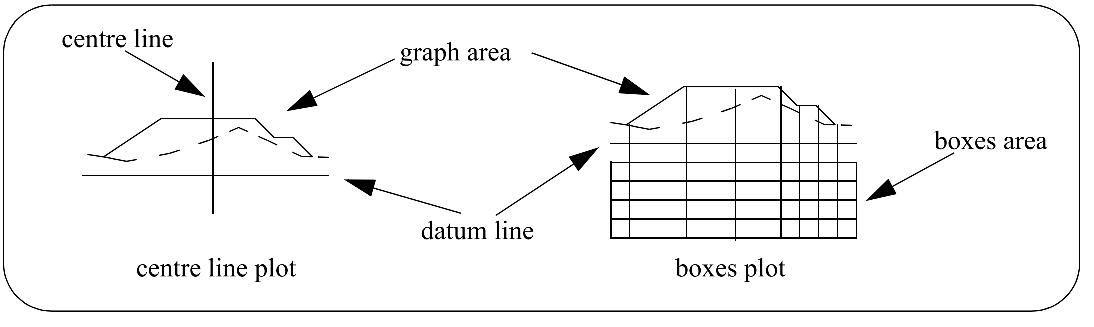


The x-section sub-plot itself consists of the three regions - graph, datum and boxes.

The **graph area** is the area where the actual plots of the strings are drawn. This exists for both the centreline and boxes case.

The **datum area** is the region between graph area and the datum line. This exists for both the centreline and boxes cases.

The **boxes area** is where the offset values and the heights for the strings drawn on the x-section plot are labelled. This only exists for the boxes case.



For both cases, the x-section sub-plot can be labelled with other information including

(a) grades across the x-section

(b) points across the x- section

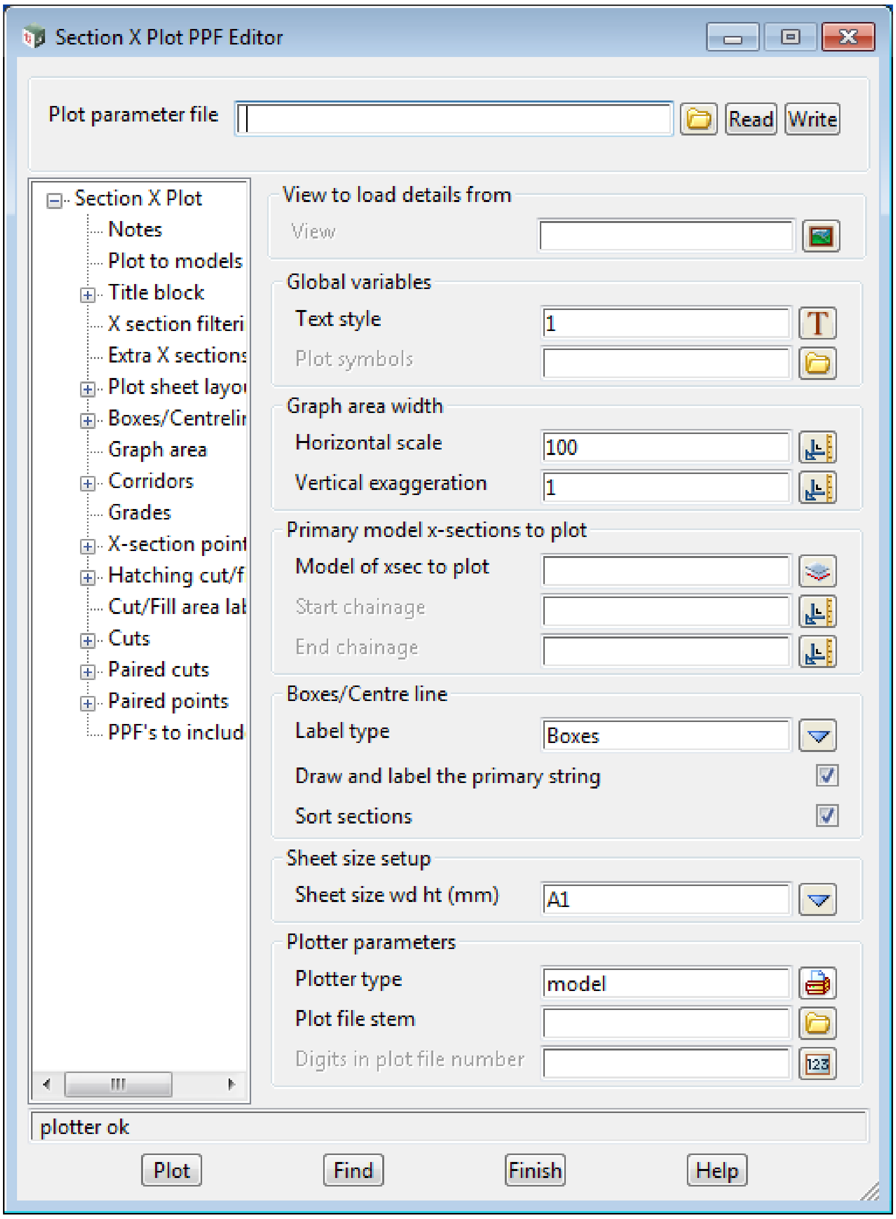
(c) cuts the x-section makes through strings

(d) cut and fill areas

All the required parameters for controlling the cross section plot are set up in the ***Section X Plot PPF Editor*** and will be described in detail in the following sections.

Please continue to the next section 26.3.2 Section X Plot - Front Page or return to 26.3 Section X Plot PPF Editor.

26.3.2 Section X Plot - Front Page



Section: Plot parameter file

*a **plot parameter file** can be used to load values into the fields of the PPF Editor, or as a file to write out all current values in a PPF Editor to.*

This section is documented for all the PPF Editors in 26.2.1 Plot Parameter File.

Section: Section: View to load details from AND Global variables

*a **section view** can be selected to load certain values into fields of the PPDF Editor. For example, Vertical exaggeration and Corridor models. And there are variables to use if some values aren't given.*

These sections are documented for all the PPF Editors in 26.2.2 View to Load and Global Variables.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Graph area width

Horizontal Scale **scale** **measure box** **available measures**

horizontal scale to be used for X section plots

Vertical exaggeration **vertical_exaggeration** **measure box** **available measures**

vertical scale to be used for X section plots

Section: Primary model x-sections to plot

Model of xsec to plot **model_to_plot** **model box**

model of X sections to plot

Start chainage **start_chainage** **input**

start chainage of X sections to plot

End chainage **end_chainage** **input**

start chainage of X sections to plot

Section: Boxes/Centre line

Label type label_type choice box centre *line*
boxes

the label type to be used for plotting.

Draw and label the primary string primary_string tick box

if ticked, the primary string will be labelled and drawn.

Sort sections **sort_sections** **tick box**

if ticked, the X sections will be sorted in increasing chainage. This is important if individual cross sections are added after an apply function for example.

Section: Sheet size setup and Plotter parameters

these sections define the size of the "paper" to plot on, the type of plotter to use and the naming to use for the plot files.

These sections are documented for all the PPF Editors in 26.2.3 Sheet Size and Plotter Parameters.

Please continue to the next section 26.3.3 Notes - X-Section.

26.3.3 Notes - X-Section

This is documented for all the PPF Editors in 26.2.4 Notes.

Please continue to the next section 26.3.4 Plot to Models - X-Section.

26.3.4 Plot to Models - X-Section

This is documented for all the PPF Editors in 26.2.5 Plot to models.

Please continue to the next section 26.3.5 Title Block - X Section.

26.3.5 Title Block - X Section

This is documented for all the PPF Editors in 26.2.6.1 Title Block Section in PPF Editors.
For more general information about a title block, see 26.2.6 Title Block.

Please continue to the next section 26.3.6 X Section Filtering.

26.3.6 X Section Filtering

The model of cross sections usually contains all the sections required to accurately model the object.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: X-Section Filtering

For plotting purposes, often only a subset of all the cross sections is required to be plotted so the filtering section contains methods to select only those cross sections that are to be plotted.

Filter cross-sections filter_sections tick box

*if **tick**, the fields in this tab are used to filter cross sections in the X-sections model.*

Filtered sections colour filter_colour colour box available colours

colour of the filtered cross sections

Created by function filter_function function box available functions

a model of sections can contain sections created by more than one function. If a function is given, only cross sections created by that function are selected.

Regular filtering interval filter_interval input

regular interval to use for filtering the cross sections

Regular culling tolerance filter_tolerance input 0

tolerance to use when selecting a cross section

Include start section filter_start tick box

*if **tick**, a section at the start chainage is included even if the start chainage is not a regular interval*

Include end section filter_end tick box

*if **tick**, a section at the end chainage is included even if the end chainage is not a regular interval*

Include chainage equality sections filter_equalities tick box

*if **tick**, include sections where there is a chainage equality*

Reference string filter_reference string select

*if **tick**, a string is selected to define which sections occur at H/V tangent points, crest and sags.*

Include H tangent sections filter_tangent tick box

*if **tick**, a sections at the horizontal tangent points of the reference string are included even if they are not a regular interval*

Include V tangent sections filter_tangent_vertical tick box

*if **tick**, a sections at the vertical tangent points of the reference string are included even if they are not a regular interval*

Include V crest/sag sections filter_crestsag tick box

*if **tick**, a sections at the crest and sag points of the reference string are included even if they are not a regular interval*

Special chainage file filter_spc_file file box *.spc files

if non blank, a file of chainages to include sections at even if they are not a regular interval

Please continue to the next section 26.3.7 Extra X Sections To Plot.

26.3.7 Extra X Sections To Plot

The order and centreline chainages of the x-section subplots for the x-section plot are defined by the sections from the primary model of cross sections.

The sections through any specified triangulations and service models, and offsets for labelling are fully defined by these primary x-sections.

However it is also possible to plot extra x-sections on each of the sub-plots by supplying extra models of x-sections which are at the same plan positions as the primary x-section strings.

Only those x-sections from the extra models that are within a user specified tolerance of a primary x-section are plotted.

The extra models of x-sections and the plan tolerance for checking that the extra cross-sections are the same position as the primary x-sections are

The columns for the fields documented in the sections are for.

Panel Field	**Parameter name**	**Type**	**Pop-Up**

Section: Extra models of x-sections to plot

Model tolerance **extra_model_tolerance_1** **input**

user defined distance

Model name extra_model_n **input**

extra models of x-sections *to plot.*

Please continue to the next section 26.3.8 Plot Sheet Layout.

26.3.8 Plot Sheet Layout

X-sections are normally generated at chainages along a given centreline. This centreline chainage is stored with each x-section string.

The chainages of the actual x-sections are referred to as **offsets** from the centreline position rather than x-section chainages. The offsets of the x-section are set up so that the zero offset occurs where the x-section crossed the centreline string.

The x-sections along the centreline are stored in the one model (the primary model) which is then used to generate the cross section plot.

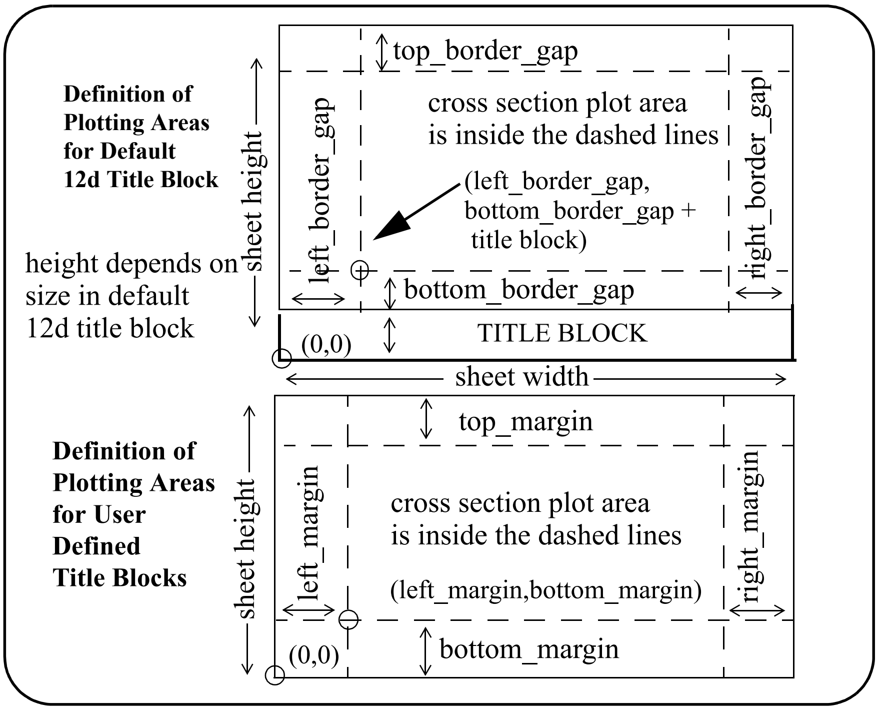
The x-sections are plotted in the order they occur in the x-section model and start being plotted at the bottom left hand corner of the sheet (left_margin,bottom_margin).

The individual x-section plots are then drawn going up the column, and when the column is full, start from the bottom of the next column.

When a sheet is full, a follow on sheet is created.

Each x-section from the primary model of x-sections generates its own sub-plot for which the x-section is the primary string. Hence the cross section plot consists of many individual plots drawn on one or more plot sheets.

Each plot sheet is considered to have only positive co-ordinates with the origin (0,0) in the left hand corner. The units for the plot are millimetres.



The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Plot width parameters

The width of the plot can be a fixed distance left or right of the centre line (zero offset) or for the full section plus an extra left and right distance:

Absolute extensions absolute_extensions tick box

if ticked, the section goes from the left_extension offset on the left to the right_extension offset on the *right. If unticked the section goes for the entire section length plus the left and right extension distances.*

Left extension left_extension input

The left_extension value in world units.

Right extension right_extension input

The right_extension value in world units

Section: Align section parameters

If required, all the sub-plots in a column can be automatically positioned so that the zero offsets (the centrelines) of each x-section line up.

Line up centrelines line_up_cl tick box

if ticked, each sub plot will be aligned on the plot using the centreline of each x section.

See

26.3.8.1 Plot Sheet Layout - Margins
26.3.8.2 Plot Sheet Layout - Sub Plot Gaps

Or return to 26.3 Section X Plot PPF Editor.

26.3.8.1 Plot Sheet Layout - Margins

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Margins for standard 12d title file - Border gaps

If the default 12d title block is used, then the size of the bottom of title block depends on the size. The following parameters are used in the default title block case and the bottom_border_gap is added to the calculated height of the bottom of the title block.

Left (mm) left_border_gap input

left border gap (in millimetres).

Right (mm) right_border_gap input

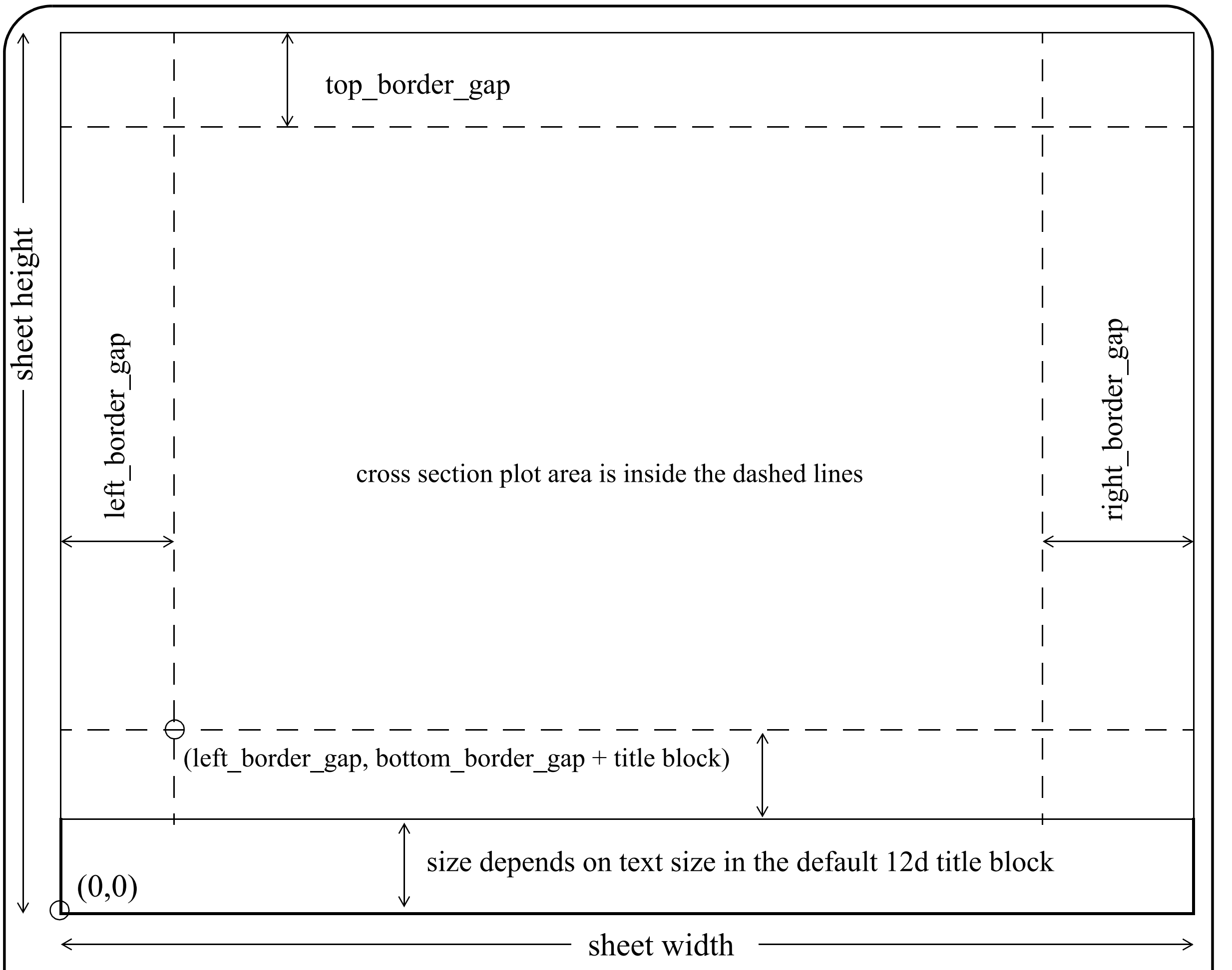
right border gap (in millimetres).

Top (mm) top_border_gap input

top border gap (in millimetres).

Bottom (mm) bottom_border_gapinput

bottom border gap (in millimetres).



Definition of Plotting Areas for Default 12d Title Block

Section: Margins for user title file

Left (mm)	left_margin	real box

left margin (in millimetres).

Right (mm)	right_margin	real box

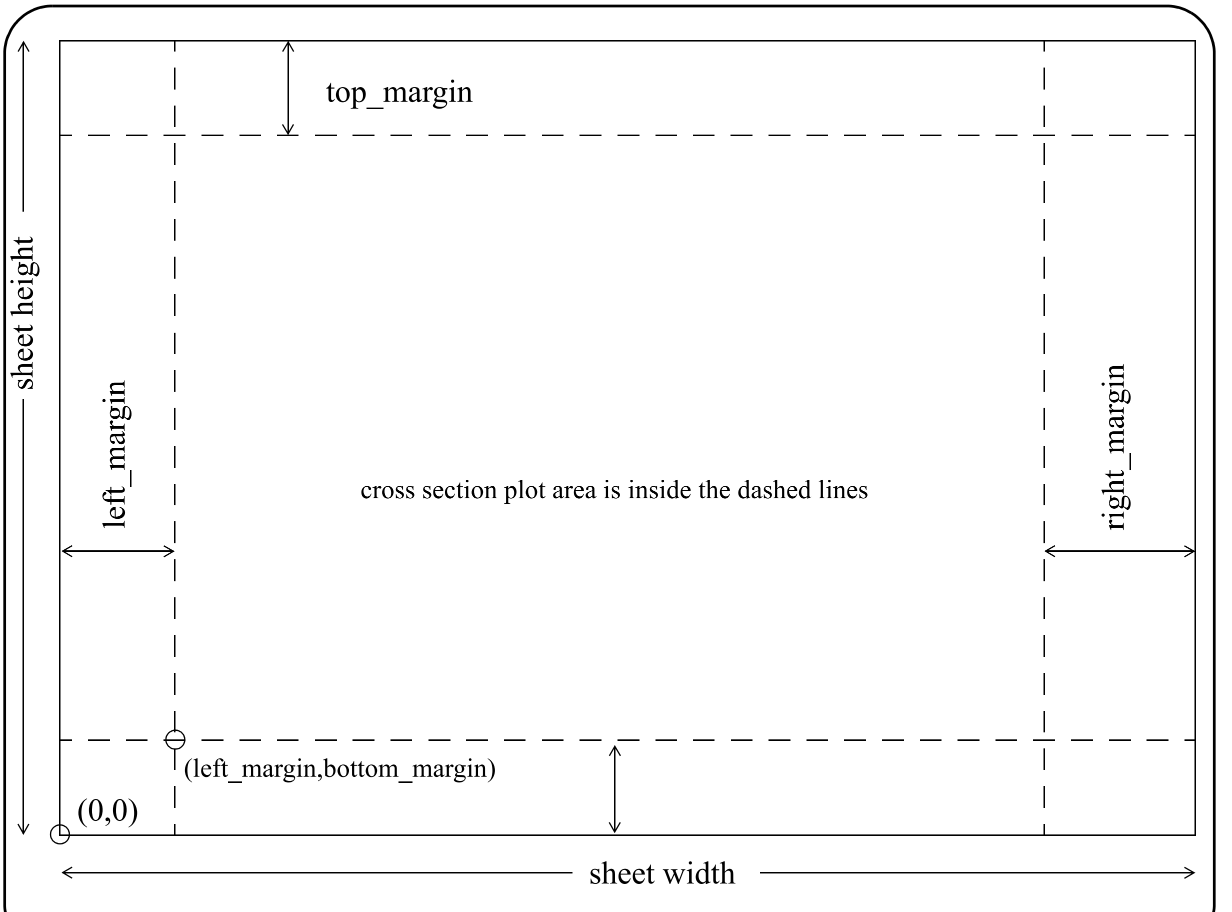
right margin (in millimetres).

Top (mm)	top_margin	real box

top margin (in millimetres).

Bottom (mm)	bottom_margin	real box

bottom margin (in millimetres).



Definition of Plotting Areas for User Defined Title Blocks

Because the user can easily select from the plotting panel whether a User Defined Title Block or the default 12d title block is used, both sets of margin and gap parameters can exist in the one plot parameter file.

The x-sections are plotted in the order they occur in the x-section model and start being plotted at the bottom left hand corner of the cross section plotting area.

The individual x-section plots are then drawn going up the column, and when the column is full, start from the bottom of the next column.

When a sheet is full, a follow on sheet is created.

Continue to the next section 26.3.8.2 Plot Sheet Layout - Sub Plot Gaps or return to 26.3.8 Plot

Sheet Layout.

26.3.8.2 Plot Sheet Layout - Sub Plot Gaps

The gaps between the sub plots is restricted to those specified below.

Section: Sub plot gaps parameters

The fields and buttons used in this section have the following functions.

Panel Field | **Parameter name** | **Type** | **Pop-Up**

Left (mm) left_sub_plot_gap real box

left margin (in millimetres).

Right (mm) right_sub_plot_gap real box

right margin (in millimetres).

Top (mm) top_sub_plot_gap real box

top margin (in millimetres).

Bottom (mm) bottom_sub_plot_gap real box

bottom margin (in millimetres).

The x-section sub-plot itself consists of the three regions - boxes, datum and graph.

The **boxes area** is where the offset values and the heights for the strings drawn on the x-section plot are labelled.

The **datum area** is the region between the boxes area and the graph area.

The **graph area** is the area where the actual plots of the strings are drawn.

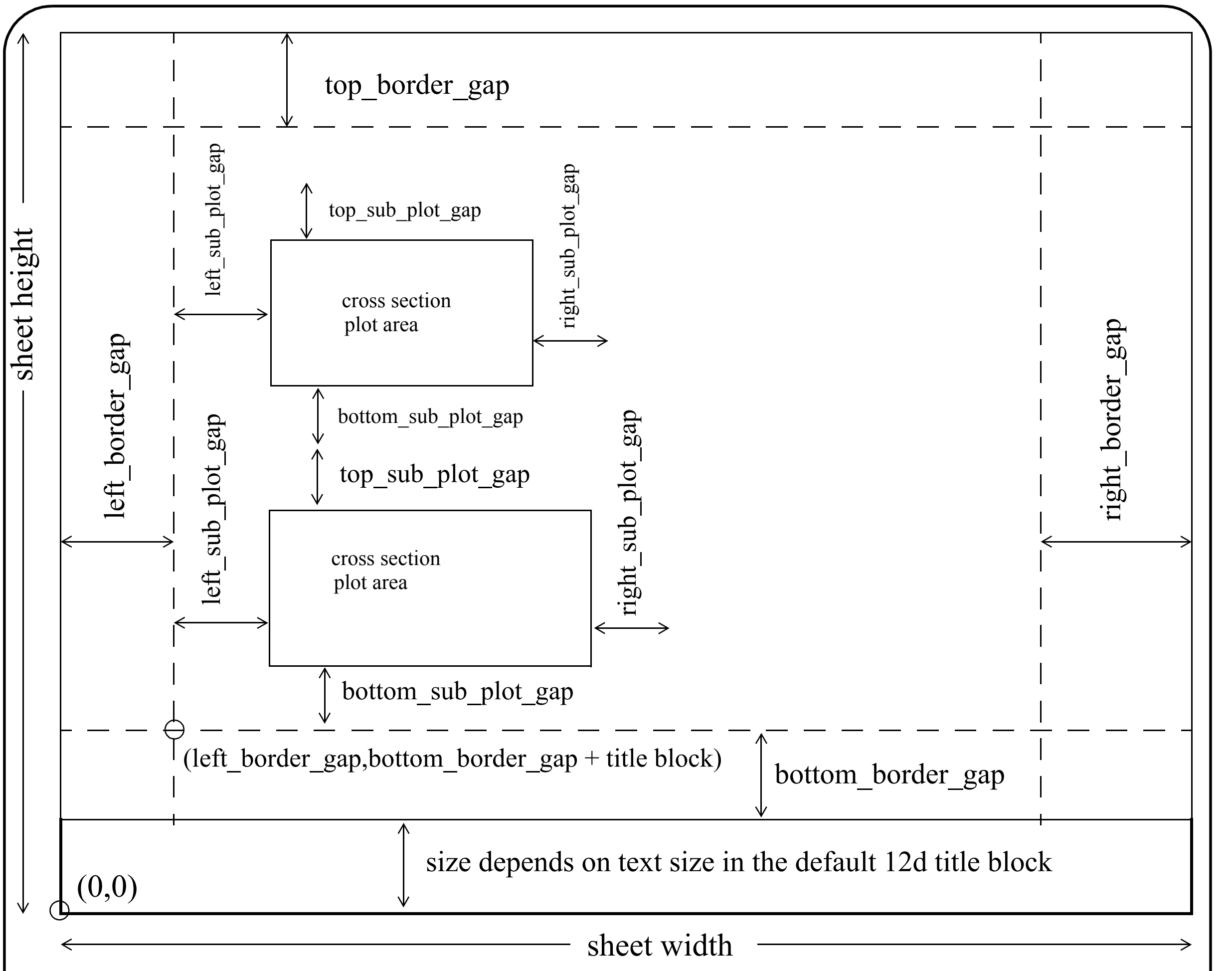
Apart from information labelled in the boxes area, the x-section sub-plot can label other information such as

(a) grades across the x-section

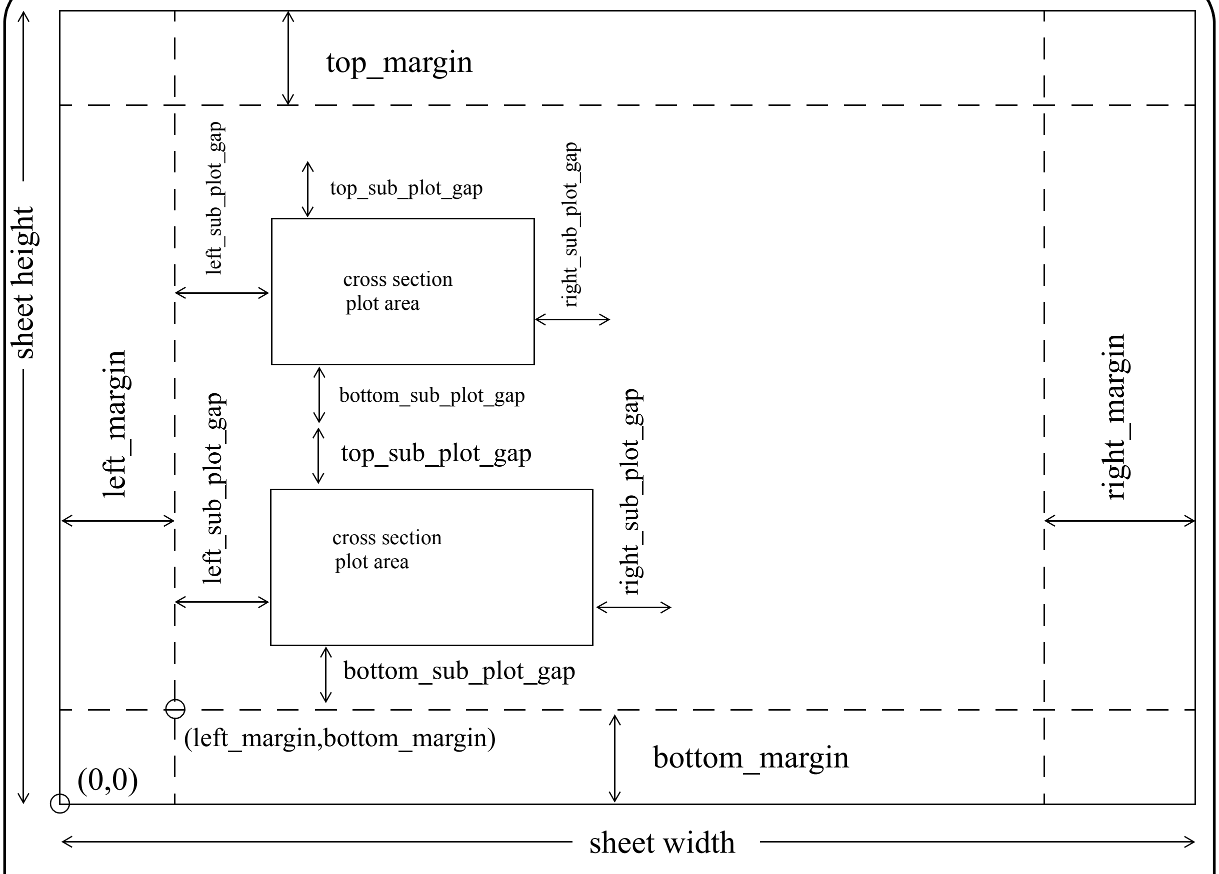
(b) points across the x- section

(c) cuts the x-section makes through strings

(d) cut and fill areas



Definition of Plotting Areas for Default 12d Title Block



Definition of Plotting Areas for User Defined Title Blocks

Please continue to the next section 26.3.9 Boxes/ Centreline Labels or return to 26.3.8 Plot Sheet Layout.

26.3.9 Boxes/ Centreline Labels

The x-section can be labelled with either

(a) an upright, and the offset and height value at the zero offset (normally the centre line position) - Centre line

(b) the heights of the x-section and the tins at all the x-section points. - Boxes

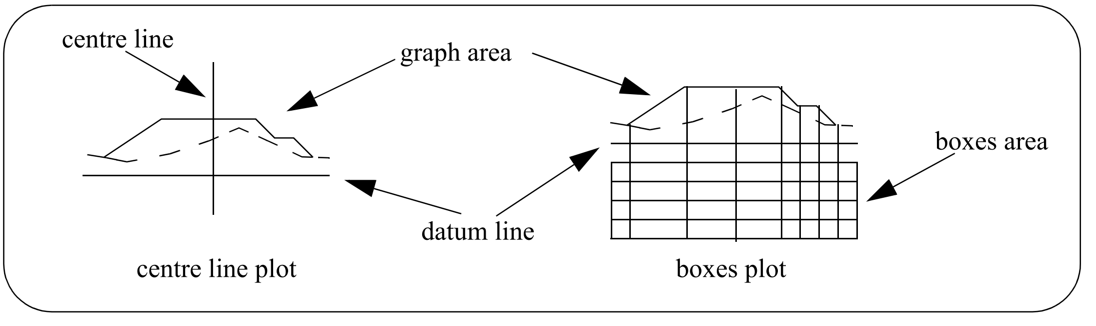
The choice is given by the parameter ***label_type*** defined in the section Section: Boxes/Centre line.

A datum line exists for both cases.

For the boxes case, a box area for the offset and heights is created below the datum line. The available parameters for tailoring the box area will be given after describing the datum line parameters.

For the centre line case, the centre line and offset and height of the centre line are shown.

In both cases, the actual cross section plot is drawn above the datum line in the graph area.



The parameters defined below, are common to both boxes and centreline label types.

See

26.3.9.1 Boxes/Centreline Labels - Common Parameters
26.3.9.2 Boxes/Centreline Labels - Boxes
26.3.9.3 Boxes/Centreline Labels - Centreline

Or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.1 Boxes/Centreline Labels - Common Parameters

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Common Datum Line label parameters

Datum roundoff datum_roundoff input

value to roundoff the datum value to e.g. 0.5, 0.2, 1.0 (default 1.0)

The roundoff for the datum value is specified by the user (default 1.0) and the datum is automatically calculated for each sub-plot, and labelled.

Decimal places for datum datum_decimals input

number of decimal places to display the datum value (default 1).

*If > 0, trailing zeros are **removed** after the decimal point.*
If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Datum linestyle datum_linestyle linestyle box

datum line linestyle (default solid)

Datum name datum_name input

text to write before the datum value

Datum textstyle datum_textstyle text box

textstyle for datum information

Datum text size (mm) datum_text_size input

size of datum text and value (mm)

Datum colour datum_colour colour box

colour of the datum text

Datum line colour datum_line_colour colour box

colour of the datum line

See

26.3.9.1.1 Boxes/Centreline Labels - Common Parameters - Tins To Label
26.3.9.1.2 Boxes/Centreline Labels - Common Parameters - Centreline Chainage
26.3.9.1.3 Boxes/Centreline Labels - Com Params - Centreline Equality Chainage
26.3.9.1.4 Boxes/Centreline Labels - Com Params - X Coordinate At 0 Offset
26.3.9.1.5 Boxes/Centreline Labels - Com Params - Y Coordinate At 0 Offset
26.3.9.1.6 Boxes/CL Labels - Com Par - Height Of Primary String At 0 Offset
26.3.9.1.7 Boxes/Centreline Labels - Com Params - Text Placement
26.3.9.1.8 Boxes/Centreline Labels - Com Params - Position of ch,x,y,ht,text

26.3.9.1.1 Boxes/Centreline Labels - Common Parameters - Tins To Label

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Common parameters for tins to label

Define tin set #		input	

where n = 1 to.... The set enables the specification of a number of parameters for a number of specified tin names.

Tin name	tin_n_name	tin box	available tins

the name of the nth tin to be used for labelling.

If a tin of the name given by ***tin_n_name*** **does** not exist, then the plot is not produced and an error message is given.

Continue to the next section 26.3.9.1.2 Boxes/Centreline Labels - Common Parameters - Centreline Chainage or return to 26.3.9 Boxes/ Centreline Labels.

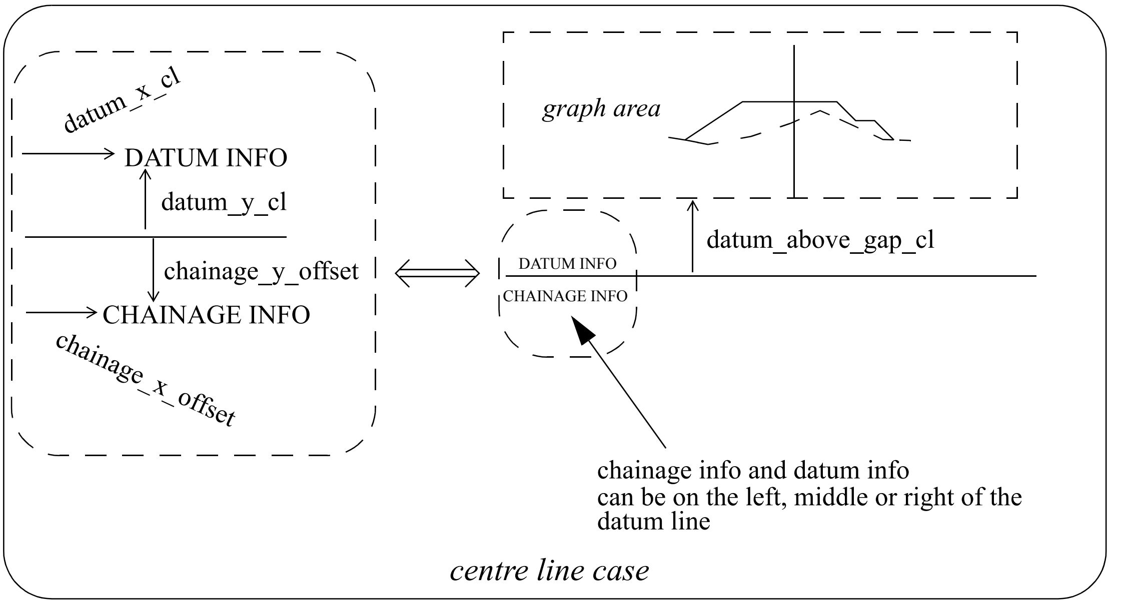
26.3.9.1.2 Boxes/Centreline Labels - Common Parameters - Centreline Chainage

Each x-section sub-plot can be labelled with the centreline chainage of the x-section string.

This CHAINAGE INFO label is made up of the text strings:

chainage_title followed by the *chainage-value*

and is drawn under the datum line.



The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Common centreline chainage label parameters

Label subplot with centreline chainage chainage_label tick box

if ticked, label sub-plot with centreline

Chainage title chainage_title input

text before the chainage value

Chainage Decimal places chainage_decimalsinput

number of decimals in the chainage value.

*If > 0, trailing zeros are **removed** after the decimal point.*
If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Chainage colour chainage_colour colour box

colour of the text

Chainage text size (mm) chainage_size input

size of the text

Chainage textstyle chainage_textstyle text box

textstyle for the chainage label

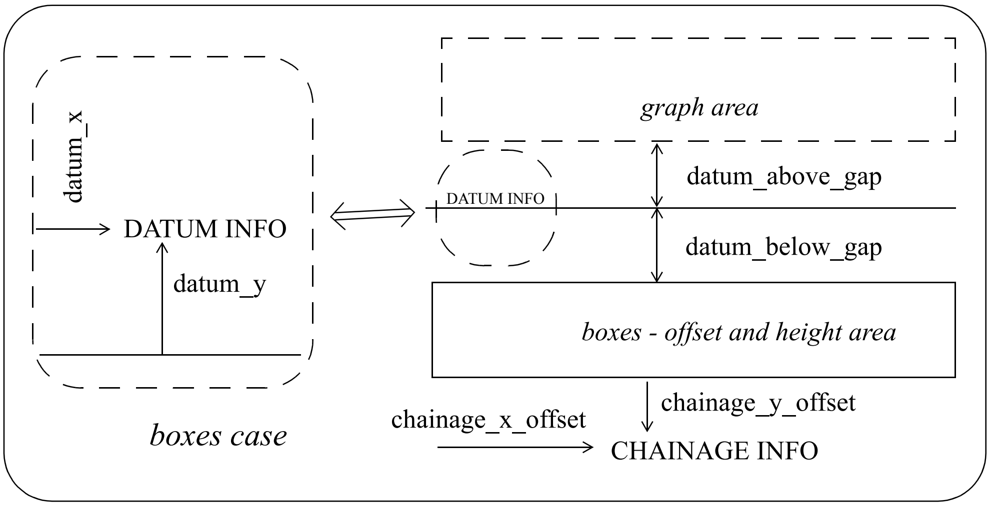
Chainage text x position (mm) chainage_x_offset input *x position of text (mm)*

Chainage text y position (mm) chainage_y_offset input

y position of text (mm)

The *chainage_x_offset* is measured from the beginning of the datum line.

The *chainage_y_offset* is measured from the bottom of the datum line with positive being **down**.



The values of the height and X and Y co-ordinates of the **primary string** (usually the design cross section) at the zero offset can be labelled. Note that zero offset is normally where the alignment string cuts the cross section.

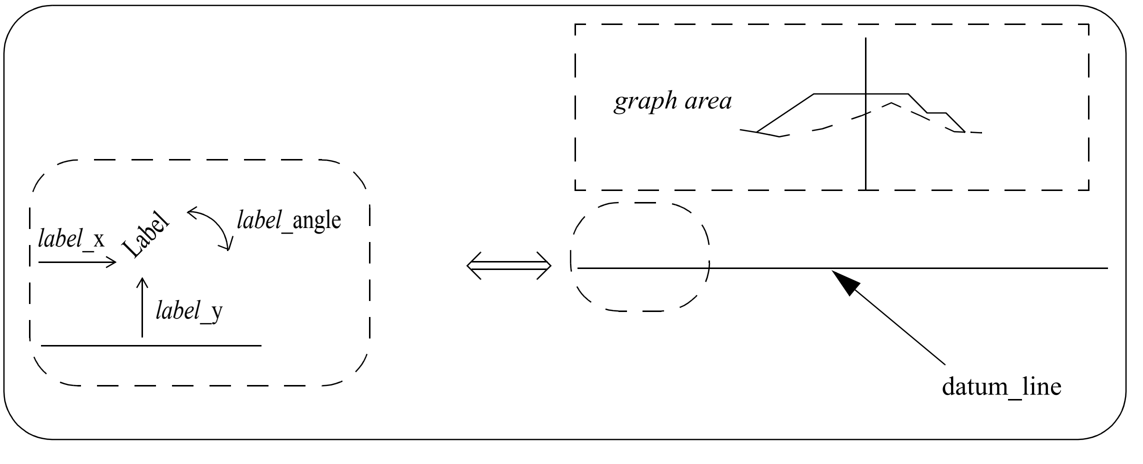
The heights of any tins (such as the natural surface) at the zero offset can also be labelled.

The labels are made up of:

pre_ text *value* post_text

where *value* is either a height or a co-ordinate.

The label is positioned at either the left, right or middle of the datum line, with an x and y adjustment and a rotation.



Chainage text justification chainage_text_justification_cl justification box

justification of the chainage text.

Continue to the next section 26.3.9.1.3 Boxes/Centreline Labels - Com Params - Centreline Equality Chainage or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.1.3 Boxes/Centreline Labels - Com Params - Centreline Equality Chainage

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Common centreline equality chainage label parameters

Label sub-plot with centreline equality chainage chainage_label_eq tick box

Equality chainage title chainage_title_eq tick box

Decimal places chainage_decimals_eq

1000s separator (non K-post) chainage_thousands_separator_eq

Zero-pack digits after 1000s separator chainage_zero_pack_eq tick box

Show K-post at non-zero offset (if defined) chainage_name_include_eq tick box

K-post pre text chainage_name_pre_eq

K-post post text chainage_name_post_eq

Space before offset of K-post chainage_offset_space_eq tick box

Plus sign before positive offset of K-post chainage_plus_eq tick box

Show equality zone (if defined) chainage_zone_include_eq tick box

Space before equality zone chainage_zone_space_eq tick box

Equality zone pre text chainage_zone_pre_eq

Equality zone post text chainage_zone_post_eq

Before/after equality separator chainage_zone_pre_eq

Colour chainage_colour_eq

Text size (mm) chainage_size_eq

Textstyle chainage_textstyle_eq

Text x position chainage_x_offset_eq

Text y position chainage_y_offset_eq

Text justification chainage_text_justification_cl_eq

Continue to the next section 26.3.9.1.4 Boxes/Centreline Labels - Com Params - X Coordinate At 0 Offset or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.1.4 Boxes/Centreline Labels - Com Params - X Coordinate At 0 Offset

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Common labelling for X coord at 0 offset

Draw Label primary_x0_draw_mode tick box

if ticked draw the label -default

Pre text primary_x0_pre_text input

pre- text for label - def " "

Post text primary_x0_post_text input

post-text for label - def " "

Decimal places primary_x0_decimals input

number of decimal places to display - def 1.

*If > 0, trailing zeros are **removed** after the decimal point.*
If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

X adjustment (mm) primary_x0_x input

x adjustment to position of label - def 0

Y adjustment (mm) primary_x0_y input

y adjustment to position of label - def 0

Angle (dms) primary_x0_angle angle box

angle of label - def 0

Colour primary_x0_colour colour box

colour of the label

Size (mm) primary_x0_size input

size (in mm) of the label

Textstyle primary_x0_textstyle text box

textstyle of the label

Justification primary_x0_justify justification box

justification of the label

Continue to the next section 26.3.9.1.5 Boxes/Centreline Labels - Com Params - Y Coordinate At 0 Offset or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.1.5 Boxes/Centreline Labels - Com Params - Y Coordinate At 0 Offset

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Common labelling for Y coord at 0 offset

Draw Label primary_y0_draw_mode tick box

if ticked draw the label -default

Pre text primary_y0_pre_text input

pre-text for label - def " "

Post text primary_y0_post_text input

post-text for label - def " "

Decimal places primary_y0_decimals input

number of decimal places to display - def 1.

*If > 0, trailing zeros are **removed** after the decimal point.*
If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

X adjustment (mm) primary_y0_x input

x adjustment to position of label - def 0

Y adjustment (mm) primary_y0_y input

y adjustment to position of label - def 0

Angle (dms) primary_y0_angle angle box

angle of label - def 0

Colour primary_y0_colour colour box

colour of the label

Size (mm) primary_y0_size input

size (in mm) of the label

Textstyle primary_y0_textstyle text box

textstyle of the label

Justification primary_y0_justify justification box

justification of the label

Continue to the next section 26.3.9.1.6 Boxes/CL Labels - Com Par - Height Of Primary String At 0 Offset or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.1.6 Boxes/CL Labels - Com Par - Height Of Primary String At 0 Offset

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Common labelling for height of primary string at 0 offset

Draw Label primary_height_draw_mode tick box

if ticked draw the label -default

Pre text primary_height_pre_text input

pre-text for label - def " "

Post text primary_height_post_text input

post-text for label - def " "

Decimal places primary_height_decimals input

number of decimal places to display - def 1.

If > 0, trailing zeros are ***removed*** *after the decimal point.*
If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

X adjustment (mm) primary_height_x input

x adjustment to position of label - def 0

Y adjustment (mm) primary_height_y input

y adjustment to position of label - def 0

Angle (dms) primary_height_angle angle box

angle of label - def 0

Colour primary_height_colour colour box

colour of the label

Size (mm) primary_height_size input

size (in mm) of the label

Textstyle primary_height_textstyle text box

textstyle of the label

Justification primary_height_justify justification box

justification of the label

Continue to the next section 26.3.9.1.7 Boxes/Centreline Labels - Com Params - Text Placement or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.1.7 Boxes/Centreline Labels - Com Params - Text Placement

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Common parameters for text labelling

Draw Label extra_text_draw_mode tick box

if ticked draw the label -default

Text extra_text input

text for label - def " "

X adjustment (mm) extra_text_x input

x adjustment to position of label - def 0

Y adjustment (mm) extra_text_y input

y adjustment to position of label - def 0

Angle (dms) extra_text_angle angle box

angle of label - def 0

Colour extra_text_colour colour box

colour of the label

Size (mm) extra__size input

size (in mm) of the label

Textstyle extra_text_textstyle text box

textstyle of the label

Justification extra_text_justify justification box

justification of the label

Continue to the next section 26.3.9.1.8 Boxes/Centreline Labels - Com Params - Position of ch,x,y,ht,text or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.1.8 Boxes/Centreline Labels - Com Params - Position of ch,x,y,ht,text

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Centreline - Chainage Text Position

Panel Field	Parameter name	Type	Pop-Up
Position of chainage value	chainage_side_cl	choice box	Middle of datum line Left of datum line Right of datum line

side of centreline to position chainage text.

Panel Field	Parameter name	Type	Pop-Up
Justification of chainage text	chainage_text_justification_cl	just. box	bottom-left bottom-centre bottom-right bottom-decimal middle-left middle-centre middle-right middle-decimal top-left top-centre top-right top-decimal decimal-left decimal-centre decimal-right decimal-point

justification of the chainage text.

Section: Labelling for X Coord At 0 Offset

Panel Field	Parameter name	Type	Pop-Up
Position of text	primary_x0_position	choice box	Middle of datum line Left of datum line Right of datum line

side of centreline to position x coord text.

Section: Labelling for Y Coord At 0 Offset

Panel Field	Parameter name	Type	Pop-Up
Position of text	primary_y0_position	choice box	Middle of datum line Left of datum line Right of datum line

side of centreline to position y coord text.

Section: Labelling for height of primary string at 0 offset

Panel Field	Parameter name	Type	Pop-Up
Position of text	primary_height_position	choice box	Middle of datum line Left of datum line Right of datum line

side of centreline to position height text.

Section: Parameters for text labelling

Panel Field	Parameter name	Type	Pop-Up
Position of text	extra_text_position	choice box	Middle of datum line Left of datum line Right of datum line

side of centreline to position text.

Continue to the next section 26.3.9.2 Boxes/Centreline Labels - Boxes or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.2 Boxes/Centreline Labels - Boxes

When **boxes** is selected for ***label_type***, the primary string (usually the design cross section) and each tin in the x-section sub-plot can be labelled with one or two lines of title, and the height at the offset position for each point in the primary string.

The **title** for the strings, is drawn in the **title area** of the **boxes area**.

The offsets/heights are drawn in the **heights area** of the **boxes area**.

Consequently the boxes area is made up of rows of text consisting of:

string/tin titles followed by the **offset/height values** across the string/tin.

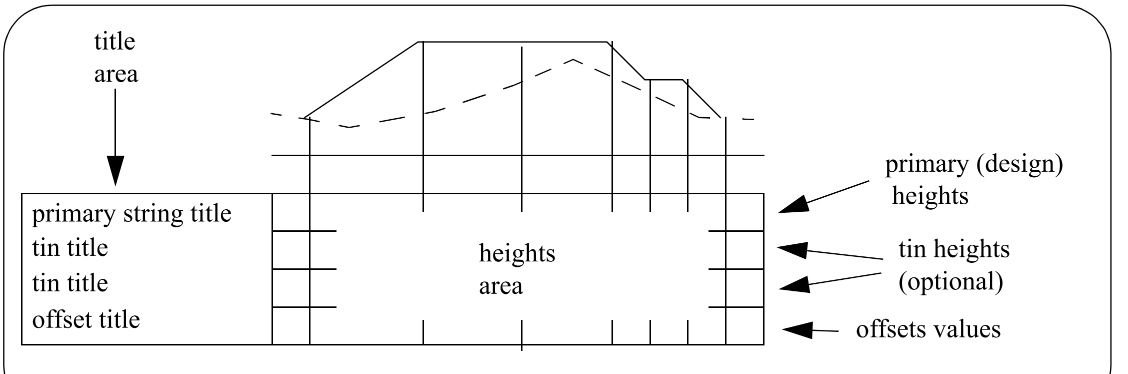
Each row is surrounded by lines to form a box.

The default order of the boxes from the bottom up is

(a) offset title and values

(b) tin title and heights - natural surface *etc*. (optional)

(c) primary string title and heights - design x-section (optional)



boxes cross section plot

The **title area** starts at the relative position (left_sub_plot_gap,bottom_sub_plot_gap).

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Default box parameters

Colour box_colour colour box

the default box colour

Insert 0 offset offset_insert_zero tick box

if ticked, a zero offset will be calculated if not already defined on the x section.

Draw box mode draw_box_mode **choice box** **Do not draw any box lines**
Draw box lines around
height area only
Draw box lines around
title and height area only
Use parameters defined in
boxes - heights

mode for drawing the boxes for the x section plot.

See

26.3.9.2.1 Boxes/Centreline Labels - Boxes - Datum Lines
26.3.9.2.2 Boxes/Centreline Labels - Boxes - Title Area
26.3.9.2.3 Boxes/Centreline Labels - Boxes - Heights Area
26.3.9.2.4 Boxes/Centreline Labels - Boxes - Outside Linework
26.3.9.2.5 Boxes/Centreline Labels - Boxes - Inside Linework
26.3.9.2.6 Boxes/Centreline Labels - Boxes - Extension Mode
26.3.9.2.7 Boxes/Centreline Labels - Boxes - Offset Titles/Values
26.3.9.2.8 Boxes/Centreline Labels - Boxes - Primary String Titles/Values
26.3.9.2.9 Boxes/Centreline Labels - Boxes - Tin Titles/Values
26.3.9.2.10 Boxes/CL Labels - Boxes - Upright Offset/Staggering Parameters
26.3.9.2.11 Boxes/CL Labels - Boxes - Upright Off/Stag Params - Uprights

Or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.2.1 Boxes/Centreline Labels - Boxes - Datum Lines

For the boxes case, the **datum line** is positioned the distance ***datum_below_gap*** above the top of the boxes area and the graph area is then positioned the distance ***datum_above_gap*** above the datum line.

Hence the graph area is distance (datum_below_gap + datum_above_gap) above the top of the boxes area.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Boxes - datum line parameters

Graph area to datum line gap (mm) datum_above_gap input

dist from datum line to bottom of the

Datum line gap to top of boxes (mm) datum_below_gap input

dist from datum line to the top of the boxes

X adjustment (mm) datum_x input

distance to move the datum text along the datum line

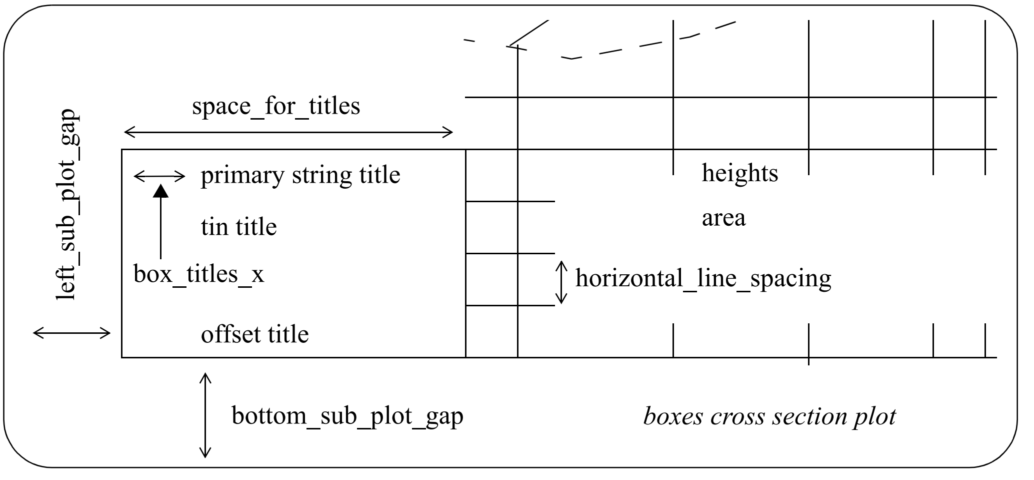
Y adjustment (mm) datum_y input

distance to raise the datum text above the datum line (used to be called datum_offset) The datum_x and datum_y can be positive, zero or negative.

Continue to the next section 26.3.9.2.2 Boxes/Centreline Labels - Boxes - Title Area or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.2.2 Boxes/Centreline Labels - Boxes - Title Area

Many distance definitions in the plot parameter file are given in terms of distance above the **top** of the boxes area.



The size of the title text is given by the ***title_box_text_size*** parameter.

The width of the title area is either given by the *space_for_titles* parameter, or if omitted, the required width is automatically calculated.

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Boxes - Title area parameters

Size of titles (mm) title_box_text_sizeinput

size of offset, height label and values

Space for titles (mm) space_for_titles input

size of title area.

There can be two lines of title text and the title text, textstyle and colour can be set independently for the primary string and each tin.

The x position of the title text is the same for all the lines of title text and can be set to be a fixed distance from the left hand side of the boxes.

X adjustment (mm) box_titles_x input

distance to move the title text from the left hand side of the boxes

The text in the title area is the same for each cross section plot on the sheet so it is possible to restrict the title area to be only on the first cross section on the sheet or the first column of cross sections.

Title area mode	label_first_only	choice box	on all sections
			on first x-sec of page only
			on x-sec's in first column

it is possible to restrict the title area to be only on the first cross section on the sheet or the first column of cross sections

Continue to the next section 26.3.9.2.3 Boxes/Centreline Labels - Boxes - Heights Area or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.2.3 Boxes/Centreline Labels - Boxes - Heights Area

The **heights area** starts at the end of the title area.

The height text is written at right angles to the bottom of the boxes. It can be either top or bottom justified with respect to the box (*box_text_justification*).

The number of decimal places and the size of the heights text can also be specified.

The height of each individual box area is either given by the *horizontal_line_spacing* parameter, or if omitted, the required height is automatically calculated.

Section: Boxes - Height area parameters

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up
Number of decimals	**number_of_decimals**	**input**	

number of decimal places in the offset height boxes. If <0, the absolute value is taken as the number of decimal places i.e. no trailing zeros are removed for the values in the offset, heights area.

Panel Field	Parameter name	Type	Pop-Up
Text size (mm)	**text_size**	**input**	

text size of the height values

Panel Field	Parameter name	Type	Pop-Up
Justification	**box_text_justification**	**choice box**	**bottom of individual boxes** top of individual boxes

justification of box text.

Horizontal line spacing (mm) **horizontal_line_spacinginput**

height of each individual box area of title area. Calculated if omitted

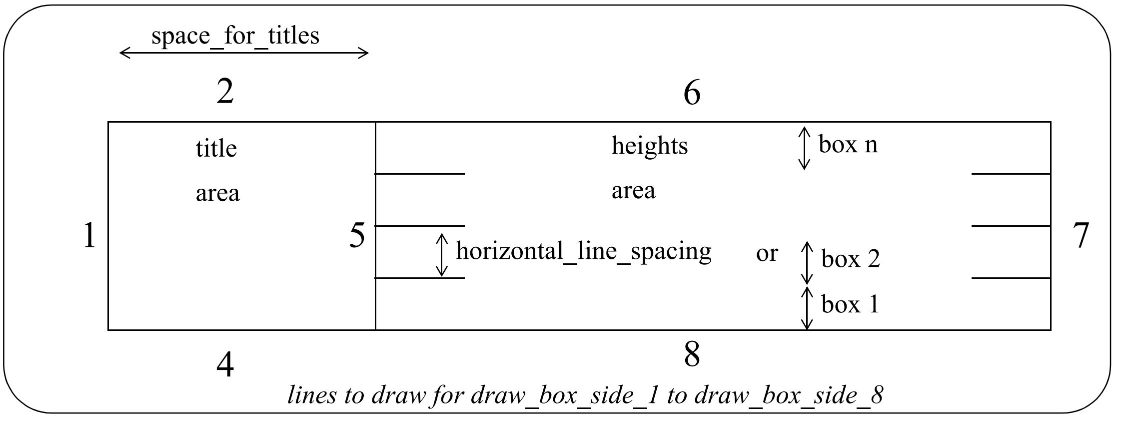
The total height of the boxes area is simply given by number of boxes drawn multiplied by the height of one box (they all have the same height).

The **width** of the heights area is determined by the number of chainages to be labelled and whether the values are staggered to prevent over writing.

Hence the total width of the boxes area is the width of the labels area plus the width of the heights area.

Continue to the next section 26.3.9.2.4 Boxes/Centreline Labels - Boxes - Outside Linework or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.2.4 Boxes/Centreline Labels - Boxes - Outside Linework



Section: Boxes - Outside linework parameters

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up
Draw the left side of title area(1)			
	draw_box_side_1	**tick box**	

if ticked, draw the left side of the title area (def).

Draw the top line of title area(2)			
	draw_box_side_2	**tick box**	

if ticked, draw the top line of the title area (def).

Draw the bottom line of title area(4)			
	draw_box_side_4	**tick box**	

if ticked, draw the bottom line of the title area (def).

Draw the left side of heights area(5)			
	draw_box_side_5	**tick box**	

if ticked, draw the left side of the heights area (def).

Draw the top line of heights area(6)			
	draw_box_side_6	**tick box**	

if ticked, draw the top line of the heights area (def).

Draw right side of heights area(7)			
	draw_box_side_7	**tick box**	

if ticked, draw right side of the heights area (def).

Draw bottom side of heights area(8)			
	draw_box_side_8	**tick box**	

if ticked, draw bottom side of the heights area (def).

Colour (1)	box_side_colour_1	**colour box**	

colour to draw left side of title area

Colour (2)	box_side_colour_2	**colour box**	

colour to draw top of title area

Colour (4) box_side_colour_4 **colour box**

colour to draw bottom of title area

Colour (5) **box_side_colour_5** **colour box**

colour to draw left side of heights area

Colour (6) box_side_colour_6 **colour box**

colour to draw top of heights area

Colour (7) box_side_colour_7 **colour box**

colour for right side of heights area

Colour (8) box_side_colour_8 **colour box**

colour for bottom of heights area

Continue to the next section 26.3.9.2.5 Boxes/Centreline Labels - Boxes - Inside Linework or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.2.5 Boxes/Centreline Labels - Boxes - Inside Linework

The lines at the top of the individual boxes inside the title area and heights area (separation lines) are controlled by the parameters *box_line_draw_mode* and *box_line_mode_n*.

The separation lines can be drawn just in the title area, just in the heights area or in both areas.

The parameter *box_line_draw_mode* can be set to control all the separation lines but there are additional parameters, *box_line_mode_n*, which override *box_line_draw_mode* for each of the individual boxes where n = 1, ... number of boxes -1.

The top of the top box is not controlled by box_line_mode_n but is controlled by the parameters *draw_box_side_2* and *draw_box_side_6.*

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Boxes - Inside linework parameters

Panel Field	Parameter name	Type	Pop-Up
Separation line mode	**box_line_draw_mode**	**choice box**	**Do not draw any separation lines** **Draw the separation lines in both areas** **Draw the separation lines in the title area only** **Draw the separation lines in the heights area only**

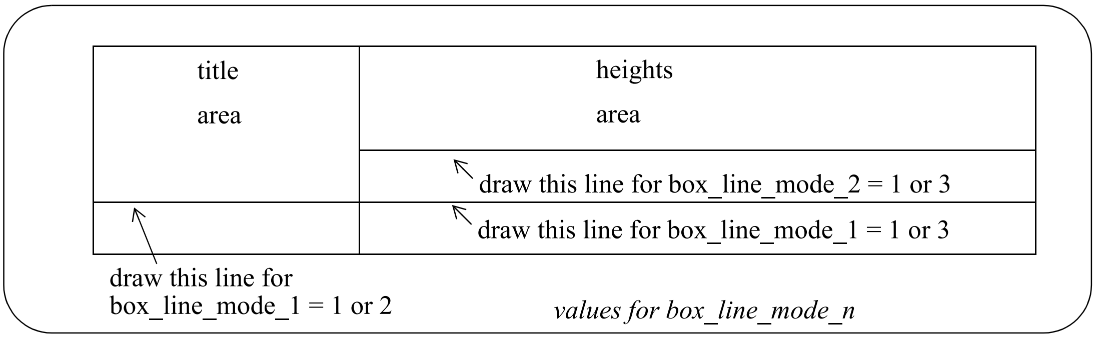
the separation line draw mode.

Panel Field	Parameter name	Type	Pop-Up
Box #		**input**	

Where box # = 1 to number of boxes(n). Box 1 is the bottom box, increasing upwards.

Panel Field	Parameter name	Type	Pop-Up
Line mode	**box_line_mode_n**	**choice box**	**No top line for title or height area (mode=0)** **Draw top line for title or height area (mode=1)** **Draw top line for title area only (mode=2)** **Draw top line for height area only (mode=3)**

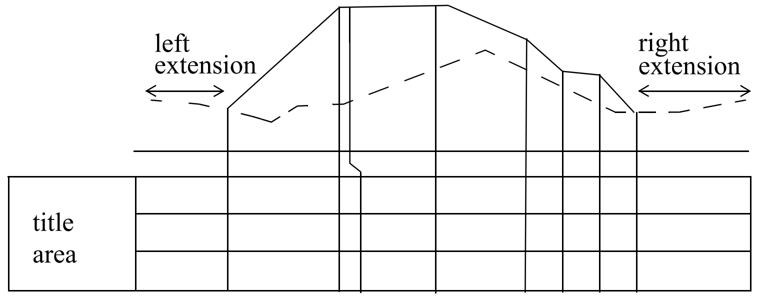
line mode for top of title and height areas for box number specified.



Continue to the next section 26.3.9.2.6 Boxes/Centreline Labels - Boxes - Extension Mode or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.2.6 Boxes/Centreline Labels - Boxes - Extension Mode

The right hand end of the boxes can stop at the end of the design x-section or extend to the end of the right_extension distance.



if box_extension_mode = 1 then the boxes extend to the end of the right extension

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Boxes - Extension mode parameters

Extension mode	**box_extension_mode**	**choice box**	**extend boxes to end of design x-section (mode=0)**
			extend boxes to end of
right			
			extension distance (mode=1)

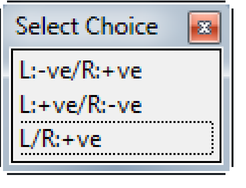
Continue to the next section 26.3.9.2.7 Boxes/Centreline Labels - Boxes - Offset Titles/Values or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.2.7 Boxes/Centreline Labels - Boxes - Offset Titles/Values

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Boxes - Offset title/value parameters

Panel Field	Parameter name	Type	Pop-Up
First line of offset title *first line of offset title*	offset_title	**input**	
Second line of offset title *second line of offset title*	offset_title_2	**input**	
Offset title textstyle *offset title textstyle*	offset_title_textstyle	**text box**	
Offset title colour *offset title colour*	offset_title_colour	**colour box**	
Offset title size (mm) *offset title size*	offset_title_size	**input**	
Offset value mode *offset value mode*	offset_value_mode	**choice box**	
Offset value colour *offset value colour*	offset_colour	**colour box**	
Offset value textstyle *offset value textstyle*	offset_textstyle	**text box**	
Offset value size (mm) *offset value size*	offset_size	**input**	

Continue to the next section 26.3.9.2.8 Boxes/Centreline Labels - Boxes - Primary String Titles/Values or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.2.8 Boxes/Centreline Labels - Boxes - Primary String Titles/Values

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Boxes - Primary string title parameters

First line primary_title real box

first line of primary string title

Second line primary_title_2 real box

second line of primary string title

Textstyle primary_title_textstyle text box

primary string title textstyle

Colour primary_title_colour colour box

primary string title colour

Size (mm) primary_title_size real box

primary string title size

Y position (mm) primary_title_y_pos real box

if set, the height in mm above the bottom of all the boxes that the primary string title text is drawn. If not set, then the text is placed at a height that puts it inside the default box for the primary string title.

Section: Boxes - Primary string value parameters

Colour primary_colour colour box

primary string value colour

Textstyle primary_textstyle text box

primary string value textstyle

Size (mm) primary_size real box

primary string value size

Y position (mm) primary_y_pos real box

if set, the height in mm above the bottom of all the boxes that the primary string value text is drawn. If not set, then the text is placed at a height that puts it inside the default box for the primary string values.

Decimal places primary_decimals integer box

number of dec places in primary string height.

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Continue to the next section 26.3.9.2.9 Boxes/Centreline Labels - Boxes - Tin Titles/Values or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.2.9 Boxes/Centreline Labels - Boxes - Tin Titles/Values

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Tin titles/values - Tin parameters

Panel Field	Parameter name	Type	Pop-Up
Use tin set #		**input**	

set number to be used to define sets of tin parameters i.e. n value

Panel Field	Parameter name	Type	Pop-Up
Tin draw mode	**tin_n_draw_mode**	**choice box**	**Draw the tin** **Do not draw the tin**

draw tin mode for the nth tin specified by set #.

Panel Field	Parameter name	Type	Pop-Up
Tin colour	**tin_n_draw_colour**	**colour box**	

tin colour mode for the nth tin specified by set #.

Panel Field	Parameter name	Type	Pop-Up
Tin label mode	**tin_n_label**	**choice box**	**Label the tin** **Do not label the tin**

tin label mode for the nth tin specified by set #.

See

26.3.9.2.9.1 Boxes/Centreline Labels - Boxes - Tin Titles/Values - Tin Titles
26.3.9.2.9.2 Boxes/Centreline Labels - Boxes - Tin Titles/Values - Tin Heights
26.3.9.2.9.3 Boxes/Centreline Labels - Boxes - Tin Titles/Values - Tin Depths

26.3.9.2.9.1 Boxes/Centreline Labels - Boxes - Tin Titles/Values - Tin Titles

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Tin title parameters

Use tin set # **input**

set number to be used to define sets of tin parameters i.e. n value

First line of tin title **tin_n_title** **input**

first line of nth tin title

Second line of tin title **tin_n_title_2** **input**

second line of nth tin title

Tin title textstyle **tin_n_title_textstyle** **text box**

nth tin title textstyle

Tin title colour **tin_n_title_colour** **colour box**

nth tin title colour

Tin title size (mm) **tin_n_title_size** **input**

nth tin title size

Title Y position (mm) **tin_n_title_y_pos** **input**

if set, the height in mm above the bottom of all the boxes that the nth tin title text is drawn. If not set, then the text is placed at a height that puts it inside the default box for the tin heights.

Continue to the next section 26.3.9.2.9.2 Boxes/Centreline Labels - Boxes - Tin Titles/Values - Tin Heights or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.2.9.2 Boxes/Centreline Labels - Boxes - Tin Titles/Values - Tin Heights

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Tin height parameters

Use tin set # **input**

set number to be used to define sets of tin parameters i.e. n value

Decimals **tin_n_decimals** **input**

number of dec places in nth tin height.

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Textstyle **tin_n_textstyle** **text box**

textstyle of nth tin height

Colour **tin_n_colour** **colour box**

nth tin height and depth colour

Size (mm) **tin_n_size** **input**

nth tin height size

Y pos (mm) **tin_n_y_pos** **input**

if set, the height in mm above the bottom of all the boxes that the tin height text is drawn.If not set, then the text is placed at a height that puts it inside the default box for the tin heights.

Continue to the next section 26.3.9.2.9.3 Boxes/Centreline Labels - Boxes - Tin Titles/Values - Tin Depths or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.2.9.3 Boxes/Centreline Labels - Boxes - Tin Titles/Values - Tin Depths

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Tin depth parameters

Use tin set # **input**

set number to be used to define sets of tin parameters i.e. n value

Depth label mode tin_n_depth_label **choice box** **Label depths**

Do not label depths

depth label mode for the nth tin specified by set #.

First line of tin depth title tin_n_depth_title **input**

first line of nth tin depth title

Second line of tin depth title

tin_n_depth_title_2 **input**

second line of nth tin depth title

Depth title textstyle tin_n_depth_title_textstyle **text box**

nth tin depth title textstyle

Depth title colour tin_n_depth_title_colour **colour box**

nth tin depth title colour

Depth title size (mm) tin_n_depth_title_size **input**

nth tin depth title size

Depth title Y position (mm) tin_n_depth_title_y_pos **input**

if set, the height in mm above the bottom of all the boxes that the tin depth title text is drawn. If not set, then the text is placed at a height that puts it inside the default box for the tin heights.

Decimals in depth value tin_n_depth_decimals **input**

number of dec places in tin height.

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Depth value textstyle tin_n_depth_textstyle **text box**

nth tin depth textstyle

Depth value colour tin_n_depth_colour **colour box**

nth tin depth colour

Depth value size (mm) tin_n_depth_size **input**

nth tin depth size

Depth value Y position (mm) tin_n_depth_y_pos **input**

if set, the height in mm above the bottom of all the boxes that the tin depth text is drawn. If not set, then the text is placed at a height that puts it inside the default box for the tin heights.

Multiplier for positive depths **depth_positive_factor** **input**

if set, this value will be used to multiply positive depth values.

Multiplier for negative depths **depth_negative_factor** **input**

if set, this value will be used to multiply negative depth values.

The depth from the primary string to a tin, at a particular offset is defined as

depth = tin height value - height of the primary string

That is, the depth that the primary string is **below** the tin.

Before plotting, the value of depth is multiplied by either the *depth_positive_factor* or *depth_negative_factor*.

if (depth >= 0) plotted_depth_value = depth * depth_positive_factor

if (depth < 0) plotted_depth_value = depth * depth_negative_factor

Hence the definition of depth can be modified by the parameters:

depth_positive_factor *value* // multiplier for positive depths

depth_negative_factor *value* // multiplier for negative depths

For example, if the opposite sign is required for depth, that is,

depth = height of the primary string - tin height value

simply set

depth_positive_factor -1

depth_negative_factor- 1

Continue to the next section 26.3.9.2.10 Boxes/CL Labels - Boxes - Upright Offset/Staggering Parameters or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.2.10 Boxes/CL Labels - Boxes - Upright Offset/Staggering Parameters

For each sub-plot, the offsets of the points across the x-section string (primary string) are used for positioning uprights (leader lines), and the offset and height labels for the uprights.

By default, there is an upright at each point across the primary string.

However, if the primary string is a 4d string, then the text at the points on the 4d string can be used as a key to **suppress** the labelling and upright at that point. See Section: Exclude uprights at nominated x-sec points

If the real offset position is used for the horizontal position of the offset/height text, text over writing can easily occur. To prevent over writing, the text is automatically **staggered**.

When staggering occurs, the real offset position is then indicated by the offset markers which are drawn at the top of the text boxes from the staggered text position back to the actual offset position of the upright.

The size and position of the staggers are given by:

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Upright offset/staggering parameters

Offset label tolerance offset_label_tolerance **input**

weed out offset values closer together (in offset units) than this value.

Top of stagger to boxes distance (mm) stagger_gap_top **input**

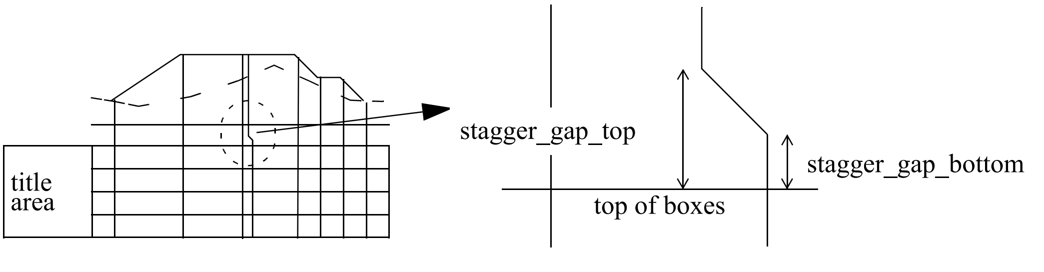
distance from boxes to top of stagger.

Bottom of stagger to boxes distance (mm) **stagger_gap_bottom** **input**

distance from boxes to bottom of stagger.

Stagger gap factor stagger_gap_factor **input**

*distance between staggers is box_text_size * stagger_gap_factor*



boxes cross section plot

Continue to the next section 26.3.9.2.10.1 Boxes/Centreline Labels - Boxes - Upright off/ staggering params - Uprights to exclude or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.2.10.1 Boxes/Centreline Labels - Boxes - Upright off/staggering params - Uprights to exclude

For each sub-plot, the offsets of the points across the x-section string (primary string) are used for positioning uprights (leader lines), and the offset and height labels for the uprights.

By default, there is an upright at each point across the primary string.

However, if the primary string is a 4d string, then the text at the points on the 4d string can be used as a key to **suppress** the labelling and upright at that point.

The parameters to **stop** labelling and uprights are:

mask_name_n	4d_string_point_ text	where n=1,100

After any *name masks* have been applied, it is often desirable to weed out offset values that are too close together before doing any labelling.

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Exclude uprights at nominated x-sec points

Set #		**input**	

set number to be used to define a number of uprights to suppress

X-sec points to exclude	**mask_name_n**	**input**	

*the string name to be excluded. **Note**: the name can include wild cards (*) and characters (?). n is the value from the Set # provided.*

Continue to the next section 26.3.9.2.11 Boxes/CL Labels - Boxes - Upright Off/Stag Params - Uprights or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.2.11 Boxes/CL Labels - Boxes - Upright Off/Stag Params - Uprights

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Uprights

Upright draw mode **uprights_draw_mode** **choice box** ***none***
ticks to stagger height
to uprights_y above boxes
to primary string
to tin 1
to tin 2
to tin 3
to tin 4
to tin 5
to tin 6
to tin 7
to tin 8
to tin 9
to tin 10

Upright draw mode above boxes.

Uprights Y distance (mm) uprights_y **input**

distance to draw the uprights for uprights_draw_mode = "to uprights_y above boxes"

Uprights colour **uprights_colour** **colour box**

uprights colour. Default is box_colour

Uprights bottom mode uprights_bottom_mode **choice box** **stop at top of boxes**
draw to bottom of boxes
draw to uprights_bottom_y below top of boxes
draw to uprights_bottom_y above bottom of boxes
ticks at chainage

Upright draw mode below top of boxes.

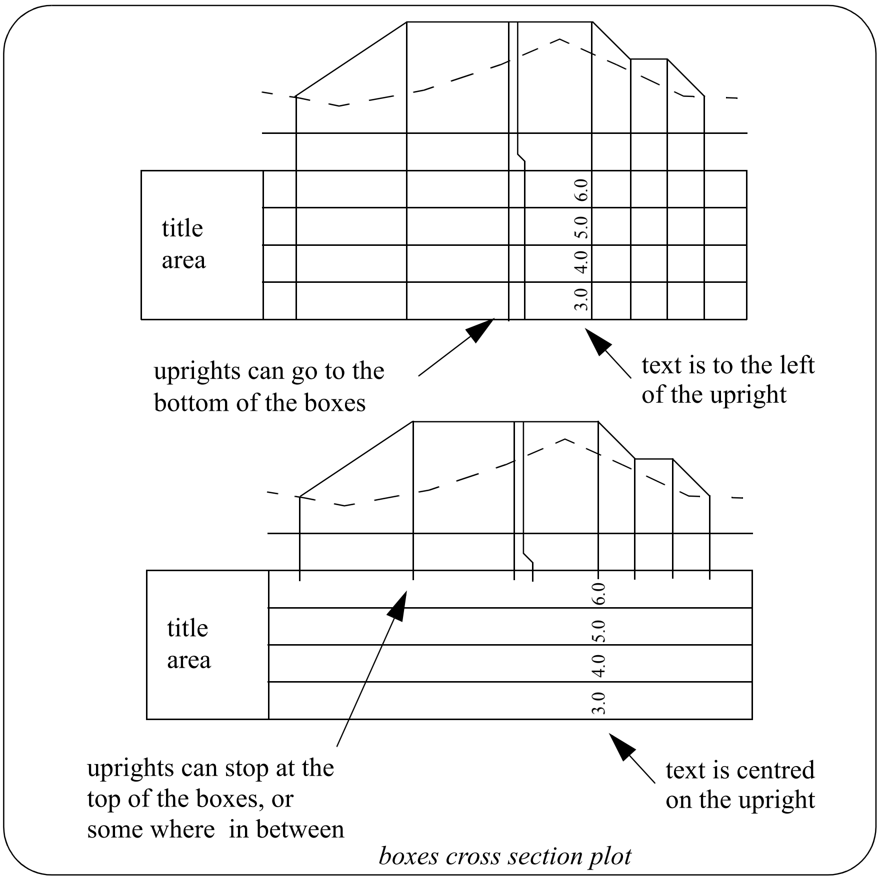
Uprights bottom Y distance (mm) uprights_bottom_y **input**

distance in mm.

Uprights text offset factor uprights_text_offset_factor **input**

*move the text by this factor*size.*

When uprights go below the top of the boxes, the height and offset text is moved to the left so that the upright does not go through the text. The left hand side of the heights boxes also moves to the left to leave room for the height text.



boxes cross section plot

Continue to the next section 26.3.9.3 Boxes/Centreline Labels - Centreline or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.3 Boxes/Centreline Labels - Centreline

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Centreline linestyle

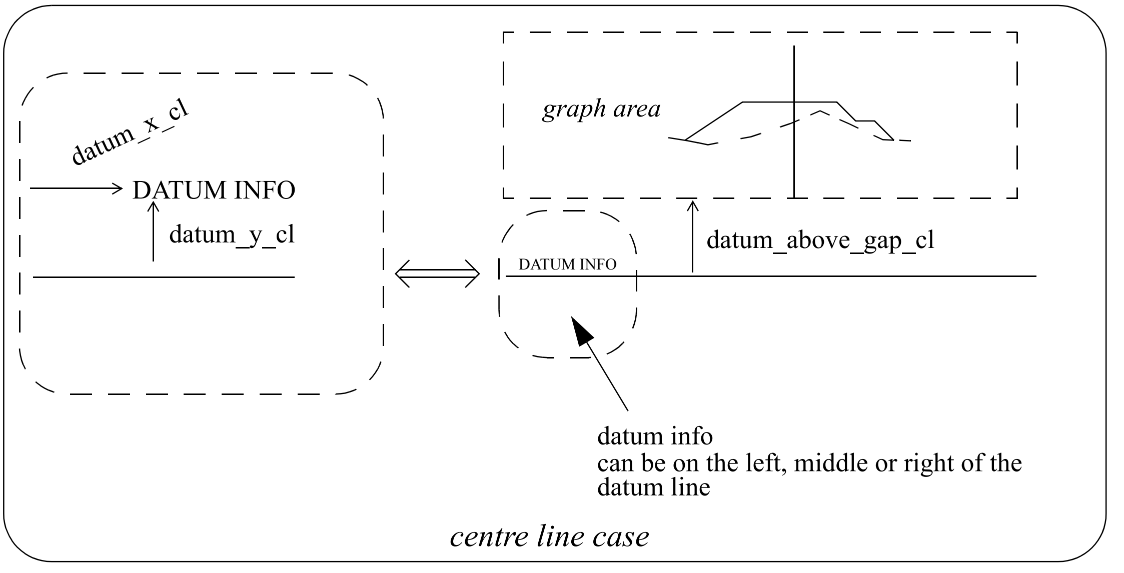
Linestyle of centreline	cl_linestyle	linetype box	

linestyle to be used for drawing centreline

See

26.3.9.3.1 Boxes/Centreline Labels - Centreline - Datum Lines
26.3.9.3.2 Boxes/Centreline Labels - Centreline - Height Of Tin At 0 Offset
26.3.9.3.3 Boxes/CL Labels - Centreline - Labelling Offset/Height (Superseded)
Or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.3.1 Boxes/Centreline Labels - Centreline - Datum Lines



For the centreline case, the graph area is positioned the distance datum_above_gap_cl above the datum line.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Centreline - datum line parameters

Graph area - datum line gap (mm) datum_above_gap_cl **input**

dist from datum line to bottom of the graph area

Position of text datum_side_cl choice box Middle of datum line
Left of datum line
Right of datum line

side of centreline to position text.

Text justification datum_text_justification_cl **just. box** **bottom-left**
bottom-centre
bottom-right
bottom-decimal
middle-left
middle-centre
middle-right
middle-decimal
top-left
top-centre
top-right
top-decimal
decimal-left
decimal-centre
decimal-right
decimal-point

justification of the datum text. ***NOTE****: - this is not normally required since by default the text justification is set to match datum_side_cl.*

Datum text position along datum line (mm)
datum_x_cl input

distance to move the datum text along the datum line

Datum text position above datum line (mm)

datum_y_cl input

distance to move the datum text above the datum line

The datum_x_cl and datum_y_cl can be positive, zero or negative.

Continue to the next section 26.3.9.3.2 Boxes/Centreline Labels - Centreline - Height Of Tin At 0 Offset or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.3.2 Boxes/Centreline Labels - Centreline - Height Of Tin At 0 Offset

The columns for the fields documented in the sections are for.

Panel Field | **Parameter name** | **Type** | **Pop-Up**

Section: Parameters for labelling the height of tin at 0 offset

Use tin set # **input**

set number to be used to define sets of tin parameters i.e. n value

Draw label **tin_n_height_draw_mode** **choice box** **Do not draw the label**

Draw the label

label mode for the nth tin specified by set #.

Position of text tin_n_height_position choice box Middle of datum line

Left of datum line
Right of datum line

side of datum line to position text.

Pre text tin_n_height_pre_text **input**

pre text for label

Post text tin_n_height_post_text **input**

post text for label

Decimals tin_n_height_decimals **input**

number of decimal places to display - def 1.

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

X adjustment (mm) tin_n_height_x **input**

x adjustment to position label

Y adjustment (mm) tin_n_height_y **input**

y adjustment to position label

Angle (dms) tin_n_height_angle **angle box**

angle of label

Colour tin_n_height_colour **colour box**

colour of label

Size (mm) tin_n_height_size **input**

size of label

Textstyle tin_n_height_textstyle **text box**

textstyle of label of label

Continue to the next section 26.3.9.3.3 Boxes/CL Labels - Centreline - Labelling Offset/Height (Superseded) or return to 26.3.9 Boxes/ Centreline Labels.

26.3.9.3.3 Boxes/CL Labels - Centreline - Labelling Offset/Height (Superseded)

In the centre line case, the value of the height of the **primary string** (usually the design cross section) at the zero offset can be labelled. This is normally where the alignment string cuts the cross section.

The label is made up of the texts:

primary_title offset_title offset_value height_text height_value

Note: These parameters have now been superseded. The parameters for labelling the height of the primary string at zero offset covers this case. To use the zero offset parameters instead, the *primary_height_pre_text* would include all text required for the Primary_title, offset_title, offset_value and height_ text (offset_value is always 0.0).

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Centreline - Labelling offset/height parameters

Include primary title in label

primary_mode_cl **Tick box**

*The parameter primary_mode_cl controls whether the **primary_title** is included in the label.*

Include height in label

height_mode_cl **Tick box**

*The parameter height_mode_cl controls whether the **height_title** and **height_value** are included in the label.*

Include offset in label

offset_mode_cl **Tick box**

*The parameter offset_mode_cl controls whether the **offset_title** and **offset_value** are included in the label.*

Position of text offset_height_side_cl **Choice** **0**

Choice **1**

Choice **2**

The label can be placed on the left, centre of right side of the datum line.

Justification of text offset_height_text_justification_cl

*justification of the offset height text. **NOTE**: - this is not normally required since by default the text justification is set to match offset_height_side_cl*

Height text label height_text **Input**

height text.

Distance to move text position along datum line (mm)

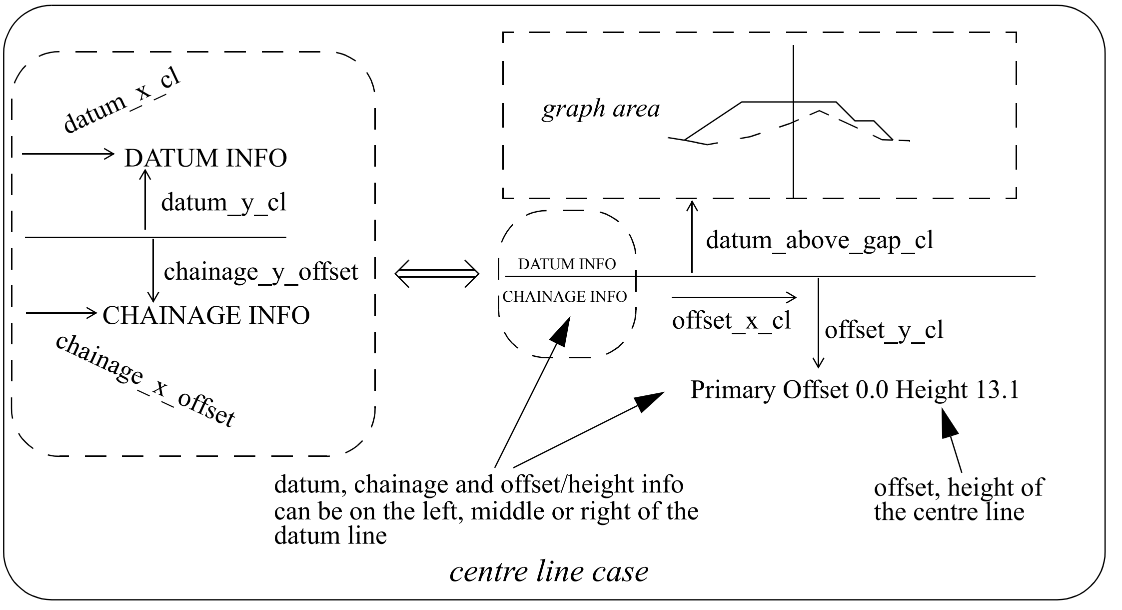
offset_x_cl **Input**

distance to move the text along the datum line.

Distance to move text position below datum line (mm)

offset_y_cl **Input**

distance to move the text below the datum line.



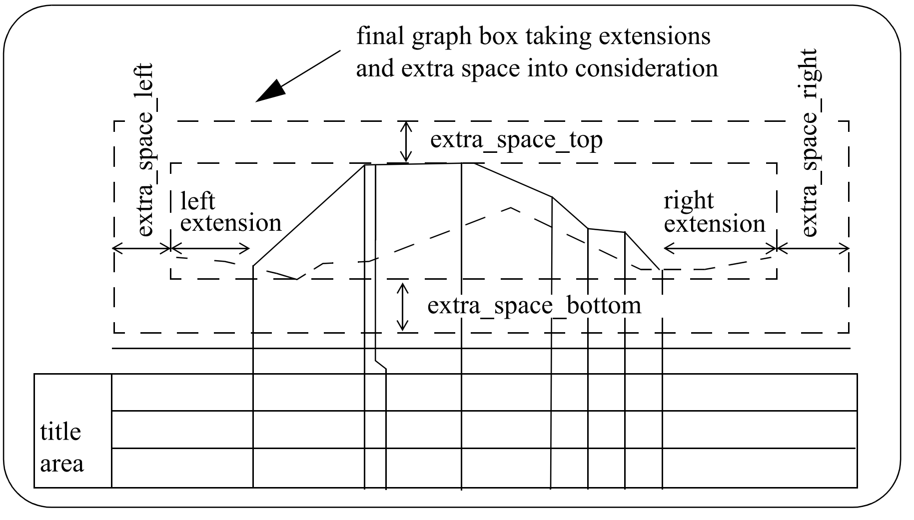
Continue to the next section 26.3.10 Graph Area or return to 26.3.9 Boxes/ Centreline Labels.

26.3.10 Graph Area

The **graph area** for each section sub-plot is the area where the actual plot of the x-section string is drawn.

The **width** of the graph area is determined by the width of the x-section string being plotted, the left and right extensions and horizontal scale (scale) given by the parameters.

The size of the graph area can be extended to allow for symbols by the following parameters:



The types of strings that are drawn in the graph area of a cross section plot are:

(a) primary string the x-section string from the x-section model that is being drawn. Called the primary string and is usually the design x-section.

(b) tins sections of the primary string through any tins in models in the corridor models.

(c) services parts of strings (from models in corridor models) that cut the defined corridor.

The **colour** of the strings in the plot is the actual string colour for cases (a) and (c), and the colour of the tin used for the section in case (b).

Although all the strings are plotted, the plot parameter file can be used to select which ones are labelled with heights.

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Graph area - Extra space parameters

Extra space units	extra_space_units	**choice box**	**world units**
			millimetres

units for specifying extra space.

Extra space left (units)	extra_space_left	**input**

distance to subtract from left of plot area

Extra space right (units)	extra_space_right	**input**

distance to add to right of plot area

Extra space top (units)	extra_space_top	**input**

distance to add to top of plot area

Extra space bottom (units)	extra_space_bottom	**input**

distance to subtract from bottom of plot area

Please continue to the next section 26.3.11 Corridors - X Section.

26.3.11 Corridors - X Section

A corridor around the primary string is defined by giving a left and right corridor width.

Any string in a model added to the section view is checked to see if it appears in the corridor, and if it does, it is drawn on the cross-section plot.

To be drawn, strings do not have to cross the primary string, but just be in the corridor.

This is documented for all the PPF Editors in 26.2.7 Corridors.

Please continue to the next section 26.3.12 Grades.

26.3.12 Grades

The plot of the x-section string is made up of straight lines joining the individual points of the x-section.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Grade parameters

Label grades grade_label **tick box**

if ticked, label grades.

Either the individual lines can be used as the segments to be labelled for grade, or adjacent lines of the same grade can be considered to be just one segment and labelled only once.

Hence the segments to be labelled for grade can be the individual lines of the x-section, or the segments defined by changes of grade.

Label change of grade segments

grade_change_only **tick box**

if ticked, label change of grade segments. If not, label individual lines.

It is also possible to ignore segments smaller that a given minimum width on the plot.

Minimum segment 3d length grade_minimum_width **input**

segments smaller than specified value (in mm) are not labelled

The grade labels are drawn parallel to the segment, centred about the segments end points, and a distance *grade_offset* above the segment. The size, colour and number of decimal places can all be set.

Minimum segment plan length grade_minimum_width_2d **input**

Minimum slope (m/m) grade_minimum_**slope** **input**

Maximum slope (m/m) grade_maximum_slope **input**

Decimal places for grades grade_decimals **input**

number of decimal places in grade.

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Size for grade text (mm) grade_size **input**

size of the grade label

Textstyle for grades **grade_textstyle** **text box**

textstyle of the grade label

Colour for grades grade_colour **colour box**

number of decimal places in grade

Grade offset (mm) grade_offset **input**

distance above the segment for label

Show grade sign grade_offset **tick box**

if ticked, the sign of the grade is labelled.

The grade can be labelled as percent cross-fall, 1 in slope, m/m or VicRoads x:1.

Also a threshold value can be set and any grades whose absolute value are below the threshold can be labelled in one way, and those above the threshold labelled a different way.

Hence, if the absolute value of the grade is less that or equal to the absolute value of *grade_threshold*, then ***grade_mode*** is used, otherwise ***grade_upper_mode*** is used.

Grade mode	grade_mode	**choice box**	**% grade** 1 in **m/m** **VicRoads x:1**

type of grade value

Grade threshold	grade_threshold	**input**

grade threshold for type of grade labelling

Grade threshold mode	grade_threshold_mode	**choice box**	**% grade** 1 in **m/m** **VicRoads x:1**

type of grade value

Grade upper mode	grade_upper_mode	**choice box**	**% grade** 1 in m/m VicRoads x:1

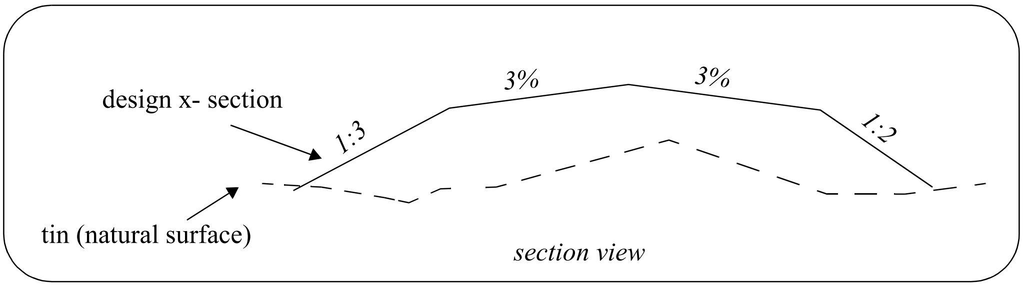
type of grade value

Decimal places for grades above threshold value

grade_upper_decimals **input**

number dec places in grades above threshold value

*If > 0, trailing zeros are **removed** after the decimal point.*
If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.



Please continue to the next section 26.3.13 X-Section Points.

26.3.13 X-Section Points

The points across each x-section can be automatically labelled on the x-section plots.

The **offset, height** and **name** of the point can be labelled as well as a **symbol** drawn. The height of tins at the same offset value can also be labelled.

The offset position for the labelling is the offset of the point.

The height position for the labelling can be specified as the

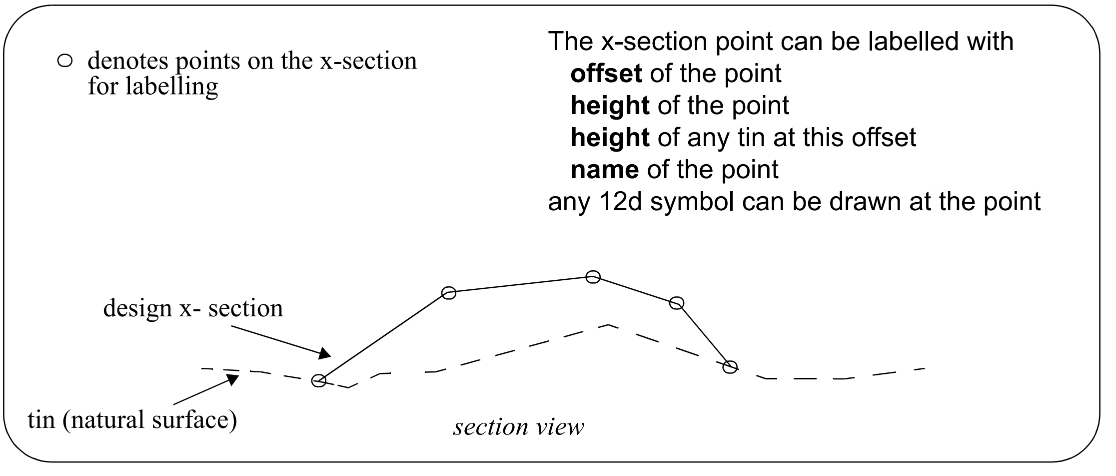
(a) top of the boxes for the x-section

(b) above the maximum height of the strings on the plot

(c) height of the point on the x-section string (primary string)

(d) height of a tin.

The actual position of the label is defined relative to the above point.



Note:

Only case (b) involves the actual height of the point on the cross section string. For all other cases, only the offset of the cut string is used.

Other heights, for example, the height of the tin at that offset can be used as the height (case (d)).

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: X-section point mask parameters

Define Set # **input**

set number to be used to define a number of points to include

Point mask **points_n_mask** **input**

The points of the x-section to be labelled for the nth set of parameters is restricted to all the points whose name satisfying the points_n_mask. For example

points_1_mask = "ke"*

or *points_1_mask = "?bank*"*

or, if both masks are required,

points_1_mask = "ke ?bank*"*

See

Or return to 26.3 Section X Plot PPF Editor.

26.3.13.1 X Section Points - Offsets

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: X-section - Offset parameters

Use Set # **input**

set number as specified in the **Define set#**

Position points_offset_n_position **choice box** **above point value**
above top of boxes
above top of graph area
to primary string
to tin 1
to tin 2
to tin 3
to tin 4
to tin 5
to tin 6
to tin 7
to tin 8
to tin 9
to tin 10

above point value.

X (mm) points_offset_n_x **input**

horizontal adjustment to position of offset text.

Y (mm) points_offset_n_y **input**

height adjustment to position of offset text.

Angle (dms) points_offset_n_angle **input**

rotation of offset text about point.

Colour points_offset_n_colour **colour box**

colour of offset text

Size (mm) points_offset_n_size **input**

size of offset text. A value of 0 = no label

Textstyle points_offset_n_textstyle **text box**

textstyle of offset text

Pre-text points_offset_n_pre_text **input**

text before offset text

Post-text points_offset_n_post_text **input**

text after offset text

Justification points_offset_n_justification

justification box **bottom-left**
bottom-centre
bottom-right
bottom-decimal
middle-left
middle-centre
middle-right
middle-decimal
top-left

top-centre
top-right
top-decimal
decimal-left
decimal-centre

decimal-right

decimal-point

justification of the offset text.

Decimals points_offset_n_no_decimals **input**

number of decimals in offset text.

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Continue to the next section 26.3.13.2 X Section Points - Heights or return to 26.3.13 X-Section Points.

26.3.13.2 X Section Points - Heights

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: X-section - Height parameters

Use Set # **input**

set number as specified in the **Define set#**

Mode points_height_n_mode **choice box** **use height of cut point**
use real height above boxes
height of primary string
use height of tin 1
use height of tin 2
use height of tin 3
use height of tin 4
use height of tin 5
use height of tin 6
use height of tin 7
use height of tin 8
use height of tin 9
use height of tin 10

height mode above point value.

Position points_height_n_position **choice box** **above point value**
above top of boxes
above top of graph area
to primary string
to tin 1
to tin 2
to tin 3
to tin 4
to tin 5
to tin 6
to tin 7
to tin 8
to tin 9
to tin 10

above point position.

X (mm) points_height_n_x **input**

horizontal adjustment to position of height text.

Y (mm) points_height_n_y **input**

height adjustment to position of height text.

Angle (dms) points_height_n_angle **input**

rotation of height text about point.

Colour points_height_n_colour **colour box**

colour of height text

Size (mm) points_height_n_size **input**

size of height text. A value of 0 = no label

Textstyle points_height_n_textstyle **text box**

textstyle of height text

Pre-text points_height_n_pre_text **input**

text before height text

Post-text	points_height_n_post_text	**input**	

text after height text

Justification	points_height_n_justification	just. box	bottom-left bottom-centre bottom-right bottom-decimal middle-left middle-centre middle-right middle-decimal top-left top-centre top-right top-decimal decimal-left decimal-centre decimal-right decimal-point

justification of the height text.

Decimals	points_height_n_no_decimals	**input**	

number of decimals in height text.

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Continue to the next section 26.3.13.3 X Section Points - Labels or return to 26.3.13 X-Section Points.

26.3.13.3 X Section Points - Labels

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: X-section - Label parameters

Use Set # **input**

set number as specified in the **Define set#**

Position points_label_n_position **choice box** **above point value**
above top of boxes
above top of graph area
to primary string
to tin 1
to tin 2
to tin 3
to tin 4
to tin 5
to tin 6
to tin 7
to tin 8
to tin 9
to tin 10

above point position.

Mode points_label_n_mode **choice box** **don't include point name**
include point name in label

label mode.

X (mm) points_label_n_x **input**

horizontal adjustment to position of label.

Y (mm) points_label_n_y **input**

height adjustment to position of label.

Angle (dms) points_label_n_angle **input**

rotation of label about point.

Colour points_label_n_colour **colour box**

colour of label

Size (mm) points_label_n_size **input**

size of label. A value of 0 = no label

Textstyle points_label_n_textstyle **text box**

textstyle of label

Pre-text points_label_n_pre_text **input**

text before label

Post-text points_label_n_post_text **input**

text after label

Justification points_label_n_justification just. box bottom-left
bottom-centre
bottom-right
bottom-decimal
middle-left
middle-centre
middle-right

middle-decimal
top-left
top-centre
top-right
top-decimal
decimal-left
decimal-centre
decimal-right
decimal-point

justification of the label.

Continue to the next section 26.3.13.4 X Section Points - Symbols or return to 26.3.13 X-Section Points.

26.3.13.4 X Section Points - Symbols

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

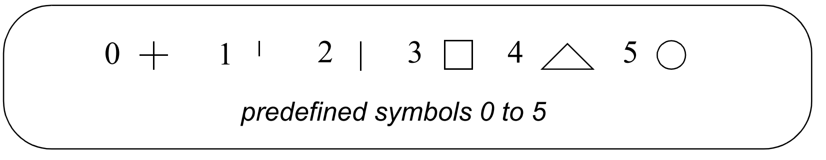
Section: X-section - Symbol parameters

Use Set # **input**

set number as specified in the **Define set#**

Mode points_symbol_n_mode **choice box** **cross (0)**
up from centre of box (1)
up and down from centre of box (2)
square (3)
triangle, base at bottom (4)
circle (5)
use a plot symbol

symbol mode.



predefined symbols 0 to 5

if a plot symbol is to be used, the points_symbol_n_style parameter must be specified.

Symbol points_symbol_n_style **plot symbols**

a valid plot symbol can be selected.

Position points_symbol_n_position **choice box** **above point value**
above top of boxes
above top of graph area
to primary string
to tin 1
to tin 2
to tin 3
to tin 4
to tin 5
to tin 6
to tin 7
to tin 8
to tin 9
to tin 10

above point position.

X (mm) points_symbol_n_x **input**

horizontal adjustment to position of symbol.

Y (mm) points_symbol_n_y **input**

height adjustment to position of symbol.

Angle (dms) points_symbol_n_angle **input**

rotation of symbol about point.

Colour points_symbol_n_colour **colour box**

colour of symbol

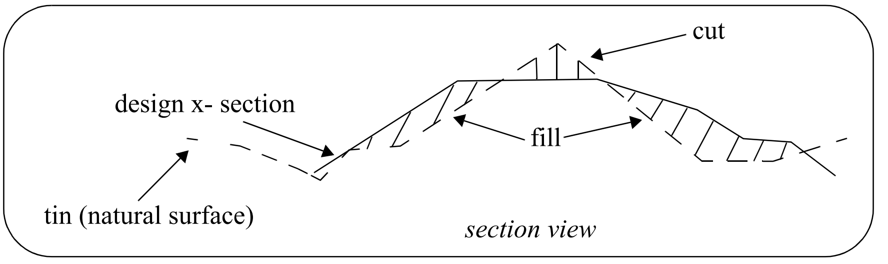
Size (mm) points_symbol_n_size **input**

size of symbol. A value of 0 = no symbol

Please continue to the next section 26.3.14 Hatching Cut/Fill.

26.3.14 Hatching Cut/Fill

This option is used to hatch cut and/or fill areas between sets of tins.



The **Hatching Cut/Fill** section is common to the **PPF Editors** and is fully documented in 26.2.8 Hatching Cut/Fill.

Section: Hatching cut/fill - Tin parameters - *see* 26.2.8 Hatching Cut/Fill

Section: Hatching cut/fill - Cut parameters - *see* 26.2.8.1 Hatching Cut/Fill - Cut

Section: Hatching cut/fill - Fill parameters- *see* 26.2.8.2 Hatching Cut/Fill - Fill

Please continue to the next section 26.3.15 Cut/Fill Area Labels.

26.3.15 Cut/Fill Area Labels

If the cross section were produced by the Apply MTF or Apply Function, each section will have as following string attributes with values calculated at the time that the cross sections were created.

LHS Cut Area

LHS Fill Area

LHS Strip Area

RHS Cut Area

RHS Fill Area

RHS Strip Area

From these, the cut left, cut right and cut total, fill left, fill right and fill total, and balance left, balance right and balance total can be labelled on the plot.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Cut/Fill are labels

this section defines the pre and post text, textstyle, size and colour for displaying selected cut/fill areas.

The fields for each row form one set of controls for the placement of the given areas on the plot.

Cut/Fill areas Grid

Label cut/fill tick box

*if **ticked**, this row is used and the selected cut/fill value is labelled.*
*If **not ticked**, this row is ignored.*

Label mode choice box cut left, cut right, cut total
fill left, fill right, fill total
balance left, balance right, balance total

the type of area to be labelled for this row.

Text size (mm)

the text size for the area label.

Label position choice box middle
left
right

the area text is positioned with respect to the datum line.

*If the **Boxes/Centreline** **Label type** is **Boxes**, then the area text is position at the left hand side of the boxes at the Datum line and the* **Label position** *is ignored.*

*If the **Boxes/Centreline** **Label type** is **Centre line**, on the Datum line but positioned horizontally at the left hand side, right hand side, or middle of the Datum line depending on* **Label position**.

X off (mm) real box

the horizontal adjustment to the selected label position.

Y off (mm) real box

the vertical adjustment to the selected label position.

Angle (dms) angle box

the angle of the attribute label in degrees minutes and seconds in hp notation.

Pre text text box

text to go before the attribute label.

Post text text box

text to go after the attribute label.

Decimal places integer box

number of decimal places to use when it is a real attribute.

*If > 0, trailing zeros are **removed** after the decimal point.*
If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Units factor real box

the area is multiplied by the **Units factor***.*

*If **blank** the value is 1.0*

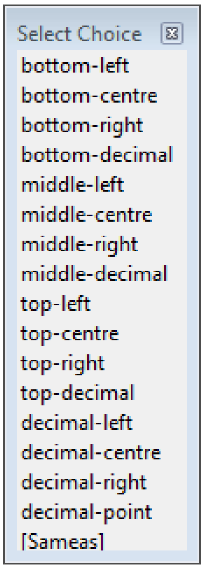
Colour colour box

the colour of the area label.

Textstyle textstyle box

the textstyle of the area label.

Justification justification box



the justification of the area label.

Please continue to the next section 26.3.16 Cuts.

26.3.16 Cuts

The cuts that each x-section string makes though strings in any user-specified model, can be automatically labelled on the x-section plots.

The **Cuts** section is common to the **PPF Editors** and is fully documented in 26.2.9 Cuts.

Section: Cuts - Model/Name mask parameters *- see* 26.2.9 Cuts.

Section: Cuts - Offset parameters *- see* 26.2.9.2 Cuts - Offsets - X Section Only

Section: Cuts - Height parameters *- see* 26.2.9.4 Cuts - Heights

Section: Cuts - Diameter parameters *- see* 26.2.9.5 Cuts - Diameters

Section: Cuts - Label parameters *- see* 26.2.9.6 Cuts - Labels

Section: Cuts - Symbol parameters *- see* 26.2.9.7 Cuts - Symbols

Continue to the next section 26.3.17 Paired Cuts - X Sections.

26.3.17 Paired Cuts - X Sections

Paired Cuts uses pairs of strings and where both strings cut each x-section, the cuts on the x-plot can be labelled with information such as

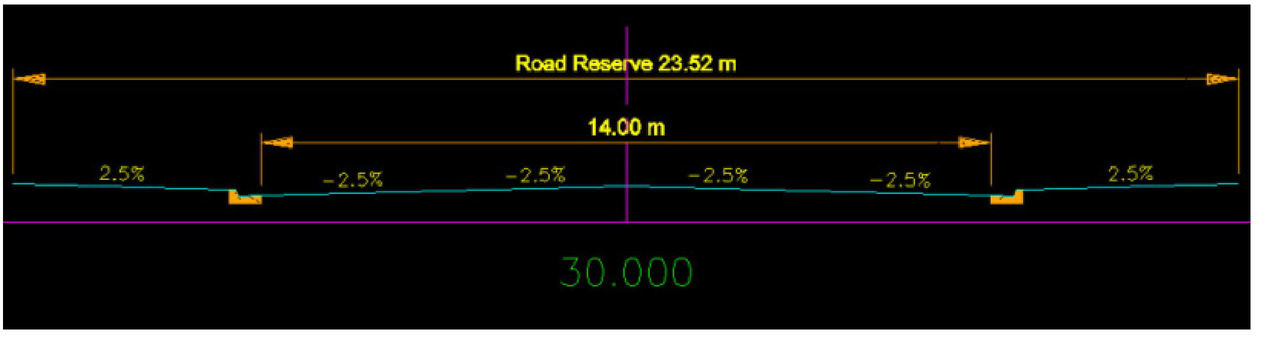
(a) the name of the first and second cut strings

(b) attributes from the first and second cut string

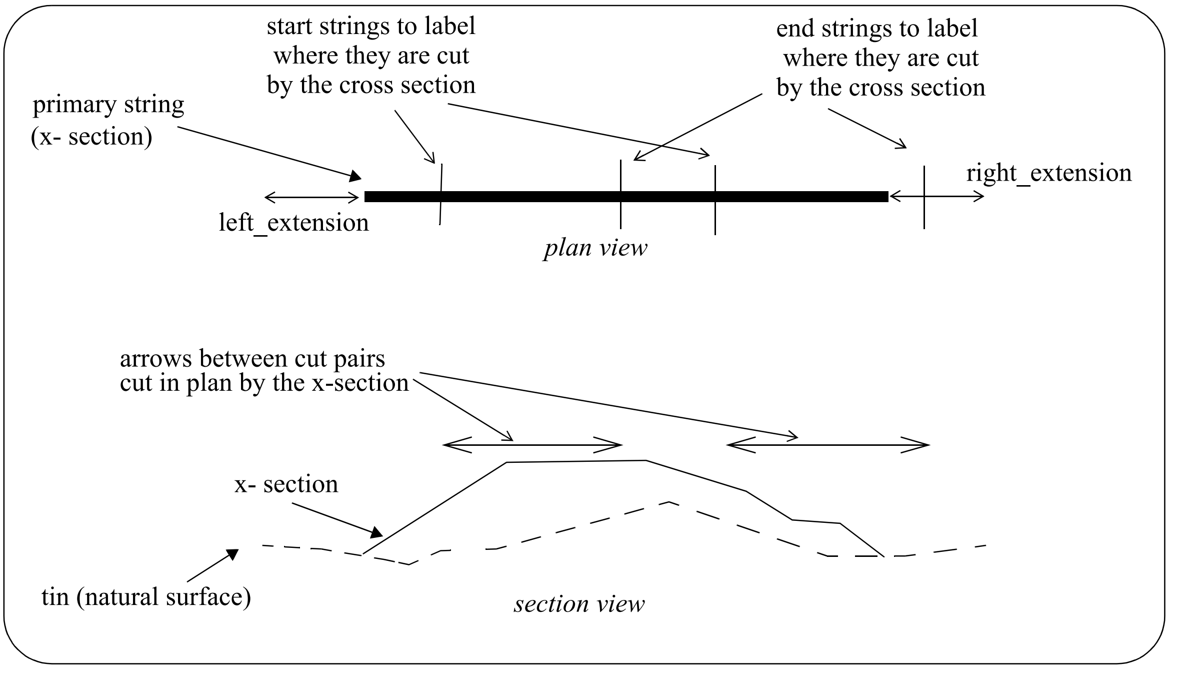
(c) offset distance between the two cuts of the pair

(d) 3d length between the two cuts of the pair

(e) the offsets on the x-section of the first and second cuts of the pair



Symbols can be drawn at the offsets of the first and second cuts (at a height specified when defining the Cut sets), and a line drawn between the symbols. Using both the line and a symbol of an arrow head makes an arrow between the two cuts.



The method for specifying which strings are to be checked for paired cuts is by first specifying the **models** (using wild cards and characters) that contains the strings, and then a **start name mask** to select the all the strings that are to be the first strings in a cut pair, and an **end name mask** to select all the strings that are the second strings in a cut pair.

For a x-section, all the cuts of the selected **start** strings are found and the cuts ordered by the offset of the cut on the x-section. Then all the cuts of the selected **end** strings are found and the cuts ordered by the offset of the cut on the x-section.

The start cuts are then processed and each start cut is paired with the next end cut with a larger offset than the start cut. It is possible that there are end cuts before the first start cut (orphaned end cuts) and start cut with no following end cut (orphaned start cuts).

Up to twenty five different sets of models and name masks can be used so that different paired cut sets can be labelled in different ways.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type**

Section: Pairing - Model/Name mask parameters

Define Set # **input**

set number to be used to define different model/mask sets.

Model pairing_n_model **model box**

models (the name can include wild cards (*) and wild characters (?)) from which start and end cut masks are derived.

Start name mask pairing_n_start_mask **input**

text string containing the name masks to select the start strings, each separated by one or more spaces, to test the string names against. If the name include spaces then it must be enclosed in the quotes **"** *(eg* "bench 2"*).*
Each mask can include wild cards (*) and wild characters (?).
For example: *"ke*"* *or,*
"?bank"* *or, if both masks are required,*
"ke ?bank*"*

End name mask pairing_n_end_mask **input**

text string containing the name masks to select the end strings, each separated by one or more spaces, to test the string names against. If the name include spaces then it must be enclosed in the quotes **"** *(eg* "bench 2"*)*
Each mask can include wild cards and wild characters.
For example: *"ke*"* *or,*
"?bank"* *or, if both masks are required,*
"ke ?bank*"*

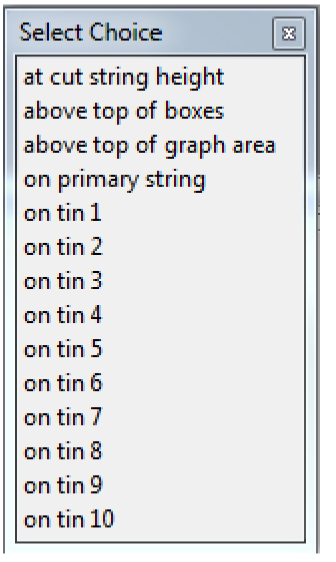
Note

If ***Start name mask*** *is blank and* ***End name mask*** *is not blank, then all strings in the models that are not used as end strings, are used as start string.*

If ***End name mask*** *is blank and* ***Start name mask*** *is not blank, then all strings in the models that are not used as start strings, are used as end string.*

This is currently not operational: If ***Start name mask*** *and* ***End name mask*** *are both blank, then all strings in the models are used. All the strings are cut by the x-section are ordered by offset, and successive pairs of cuts taken to be the cut pairs. There could be one remaining cut point (it has the largest offset) and if so, it is taken to be orphaned start point.*

Position **pairing_n_position** **choice box**



where to get the z-values to use for the two cut positions. This is used in conjunction with the choice in the **Adopted height** *field.*

If left blank, ***at cut string height*** *is used.*

Adopted height **pairing_n_height** **choice box** **lower position, higher position**

start position, end position

for the choice selected in the **Position** *field, there will be a z-value at the first cut and another z-value at the end cut. The* **Adopted height** *says which of the two z-value to use for placing symbols, lines between cuts, text from Attributes and the Offsets of the first cut and the second cut.*

If lower (higher) position, the smaller (greater) of the two z-values will be used.
If start (end), the z-value of the first (second) cut of the cut point pair will be used.

If left blank, ***lower position*** *is used.*

Include orphaned start ? pairing_n_orphan_start **choice box** **yes, no**

if ***Yes*** *and there are start cuts after the last end cut (and so can't be paired), then the last start cut (i.e. the one with the largest offset) is taken as an orphaned start and it is labelled as though there is a matching end at the end of the x-section.*
if ***No*** *and there are start cuts after the last end cut, then those start cut are ignored.*

If left blank, ***No*** *is used.*

Include orphaned ends ? pairing_n_orphan_end **choice box** **yes, no**

if ***Yes*** *and there are end cuts before the first start cut, then the first end cut (the one with the smallest offset) is taken as an orphaned end and it is labelled as though there is a matching start at the beginning of the x-section.*
if ***No*** *and there are end cuts before the first start cut, then those end cut are ignored.*

If left blank, ***No*** *is used.*

See

26.3.17.1 Paired Cuts - Lines and Symbols
26.3.17.2 Paired Cuts - Labels
26.3.17.3 Paired Cuts - Start Offset
26.3.17.4 Paired Cuts - End Offset

Or return to 26.3 Section X Plot PPF Editor.

26.3.17.1 Paired Cuts - Lines and Symbols

Symbols can be placed at the start and end cut offsets (at the z-value given by the **Position** and **Adopted height** columns when defining the Sets), and a line can also be drawn between the cuts at that z-value. Having both the line and a symbol forms an arrow between the cuts pair.

The columns for the fields documented in the sections are for.

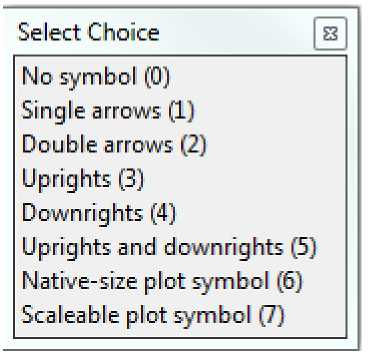
Panel Field **Parameter name** **Type**

Section: Pairing - Lines and symbol parameters

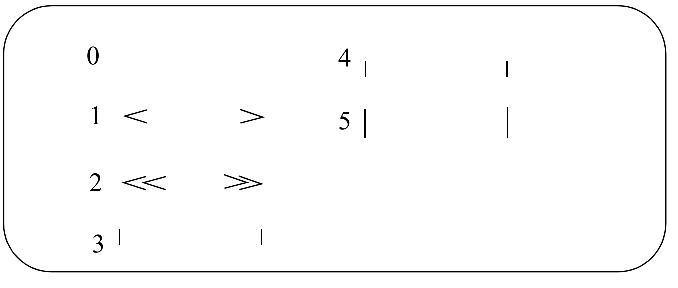
Use Set # **input**

set number as specified in the **Define set #**.

Symbol type **pairing_line_symbols_n_symbol** **choice box**



type of symbol at each end of the cut pair.



for cases 6 and 7, a plot symbol is to be used and is given in the Symbol field

The height used for the symbol at both ends of the cuts pair is given by the **Position** *and* **Adopted height** *column in the Set definition.*

Symbol **pairing_line_symbols_n_user_symbol** **plot symbols**

for Symbol type 6 and 7, the plot symbol to be used at each cut point.

For Symbol type 0 to 5, this field is not used.

The height used for the symbol at both ends is given by the **Position** *and* **Adopted height** *columns in the Set definition.*

Symbol colour **pairing_line_symbols_n_symbol_colour** **colour box**

colour of the symbol

Draw line **pairing_line_symbols_n_line** **choice box** **No line, Draw line**

if Draw line, draw a line between the start and end cuts for each cut pair.
If No line, no line is drawn between the start and end cuts for each cut pair.

The height used for the line at both ends is given by the **Position** *and* **Adopted height** *columns in the Set definition.*

Line colour **pairing_line_symbols_n_line_colour** **colour box**

colour of the line between the start and end cuts for each cut pair

Left symbol rotation **pairing_line_symbols_n_left_rotate** **measure box**

angle (measured counterclockwise from the positive x-axis with units of degrees in 4.17.1 HP Notation*) to rotate the symbol at the start cut point for the cut point pair.*

If there is no value then the rotation is 0.

Right symbol rotation **pairing_line_symbols_n_right_rotate** **measure box**

angle (measured counterclockwise from the positive x-axis with units of degrees in 4.17.1 HP Notation*) to rotate the symbol at the end cut point for the cut point pair.*

If there is no value then the rotation is 0.

X (mm) **pairing_line_symbols_n_x** **measure box**

horizontal adjustment to the position of the symbol.

Y (mm) **pairing_line_symbols_n_y** **measure box**

vertical adjustment to the position of the symbol.

Size **pairing_line_symbols_n_size** **measure box**

size of symbol. A value of 0 means no symbol.

Continue to the next section 26.3.17.2 Paired Cuts - Labels or return to 26.3.17 Paired Cuts - X Sections.

26.3.17.2 Paired Cuts - Labels

The columns for the fields documented in the sections are for.

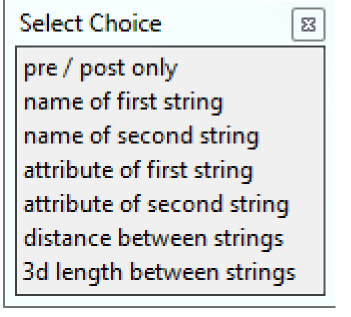
Panel Field **Parameter name** **Type**

Section: Pairing - Label parameters

Use Set # **input**

set number as specified in the **Define set #**.

Mode **pairing_label_n_mode** **choice box**



type of label to write between the start and end points of the cuts pair

The height used for the label is given by the **Position** *and* **Adopted height** *columns in the Set definition.*

Attribute name **pairing_label_n_attribute** **text box**

if **Mode** *is **attribute of first string** or **attribute of second string**, this is the name of the attribute to use as the label*

Offset **pairing_label_n_offset** **measure box**

the text is raised by this amount above the line between the cut pair

Colour **pairing_label_n_colour** **colour box**

colour of the text

Size (mm) **pairing_label_n_size** **measure box**

size of the text in millimetres. A value of 0 means no text.

Textstyle **pairing_label_n_textstyle** **textstyle box**

the textstyle for the text

Pre-text **pairing_label_n_pre_text** **text box**

text to draw before the label. This can include spaces, including one or more spaces after the last non blank character of **Pre-text**.

Post-text **pairing_label_n_post_text** **text box**

text to draw after the label. This can include spaces, including one or more spaces before the first non blank character of **Post-text**.

Decimals **pairing_label_n_no_decimals** **number box**

if the label is a number, then **Decimals** *is the number of decimal places to write the number out to.*

*If > 0, trailing zeros are **removed** after the decimal point.*
If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Leave gap for text pairing_label_n_?? **tick box**

if ticked then a gap in the line between the cuts pair is left large enough for the label and the pre-text and post-text.
If not ticked then no gap is left in the line between the cuts pair.

Rotate text to fit pairing_label_n_roate_text **tick box**

if ticked, then if the label and the pre-text and post-text will not fit between the start cut and end cut of the cut pair, the text is rotated through ninety degrees.

If not ticked, then the label text is drawn even though it will run over the ends of the cut pair.

Continue to the next section 26.3.17.3 Paired Cuts - Start Offset or return to 26.3.17 Paired Cuts - X Sections.

26.3.17.3 Paired Cuts - Start Offset

The Offset (on the x-section) of the first point of the cut pair can be labelled.

The text for the label consists of pre-text followed by Offset value followed by post-text.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type**

Section: Pairing - Start Offset parameters

Use Set # **input**

set number as specified in the **Define set #**.

The height used for the text is given by the **Position** *and* **Adopted height** *columns in the Set definition. This is then adjusted by the field* **Y (mm).**

X (mm) **pairing_start_offset_n_x** **measure box**

for placing the text: horizontal adjustment from the offset of the first cut.

Y (mm) **pairing_start_offset_n_y** **measure box**

for placing the text: the vertical adjustment to the height given by the **Position** *and* **Adopted height** *columns in the Set definition*

Angle (dms) **pairing_start_offset_n_angle** **measure box**

angle (measured counterclockwise from the positive x-axis with units of degrees in 4.17.1 HP Notation*) to rotate the text*

Colour **pairing_start_offset_n_colour** **colour box**

colour of the text

Size (mm) **pairing_start_offset_n_size** **measure box**

size of the text in millimetres. A value of 0 means no text.

Textstyle **pairing_start_offset_n_textstyle** **textstyle box**

the textstyle for the text

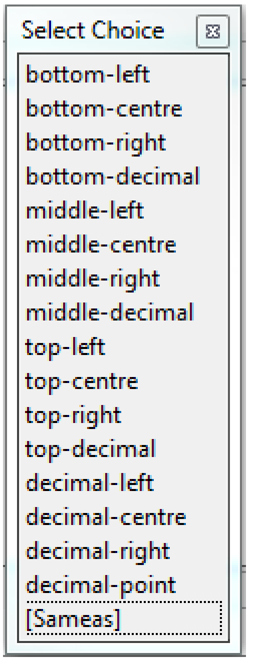
Pre-text **pairing_start_offset_n_pre_text** **text box**

text before the Offset value. This can include spaces, including one or more spaces after the last non blank character of **Pre-text**.

Post-text **pairing_start_offset_n_post_text** **text box**

text after the Offset value. This can include spaces, including one or more spaces before the first non blank character of **Post-text**.

Justification **pairing_start_offset_n_justification** **justification box**



justification of the text.

Decimals **pairing_start_offset_n_no_decimals** **number box**

the number of decimal places for the offset value

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Continue to the next section 26.3.17.4 Paired Cuts - End Offset or return to 26.3.17 Paired Cuts - X Sections.

26.3.17.4 Paired Cuts - End Offset

The Offset on the x-section of the second point of the cut pair can be labelled.

The label consists of pre-text then the Offset value followed by post-text.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type**

Section: Pairing - End Offset parameters

Use Set # **input**

set number as specified in the **Define set #**.

The height used for the text is given by the **Position** *and* **Adopted height** *columns in the Set definition. This is then adjusted by the field* **Y (mm).**

X (mm) **pairing_end_offset_n_x** **measure box**

for placing the text: horizontal adjustment from the offset of the second cut.

Y (mm) **pairing_end_offset_n_y** **measure box**

for placing the text: the vertical adjustment to the height given by the **Position** *and* **Adopted height** *columns in the Set definition*

Angle (dms) **pairing_end_offset_n_angle** **measure box**

angle (measured counterclockwise from the positive x-axis with units of degrees in 4.17.1 HP Notation*) to rotate the text*

Colour **pairing_end_offset_n_colour** **colour box**

colour of the text

Size (mm) **pairing_end_offset_n_size** **measure box**

size of the text in millimetres. A value of 0 means no text.

Textstyle **pairing_end_offset_n_textstyle** **textstyle box**

the textstyle for the text

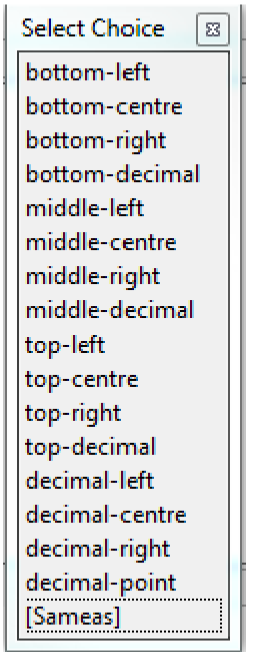
Pre-text **pairing_end_offset_n_pre_text** **text box**

text before the Offset value. This can include spaces, including one or more spaces after the last non blank character of **Pre-text**.

Post-text **pairing_end_offset_n_post_text** **text box**

text after the Offset value. This can include spaces, including one or more spaces before the first non blank character of **Post-text**.

Justification **pairing_start_offset_n_justification** **justification box**



justification of the text.

Decimals **pairing_start_offset_n_no_decimals** **number box**

the number of decimal places for the Offset value

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Please continue to the next section 26.3.18 Paired Points.

26.3.18 Paired Points

Most cross sections are created with the vertices having vertex names which correspond to the strings created by template links, or MTF modifiers, or the names of the cut strings when cross sections are created by cutting through strings.

The ***Paired Points*** option enables a user to link pairs of these vertex names from the cross section being plotted (to create a points pair), and the points from the pair can be labelled with information such as

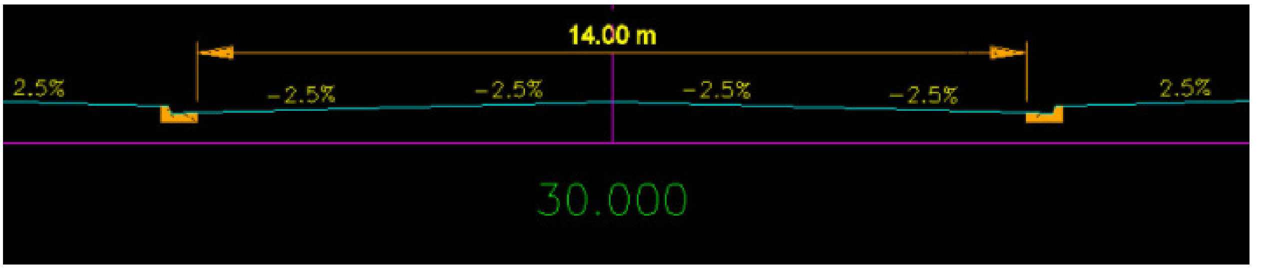
(a) the name of the first and second point of the pair

(b) vertex attributes of the cross section from the first and second points of the pair

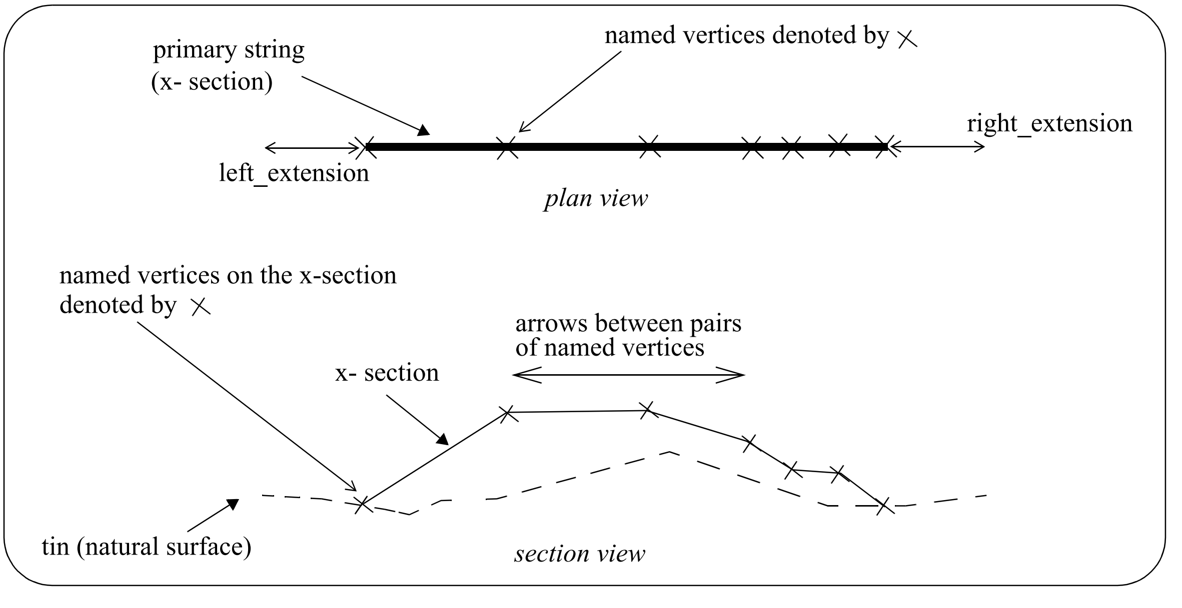
(c) distance between the two points in a pair

(d) 3d length between the two points in a pair

(e) the offsets on the x-section of the first and second points in a pair



Symbols can be drawn at the offsets of the first and second points of a pair (at a height specified when defining the Cut sets), and a line drawn between the symbols. Using both the line and a symbol of an arrow head makes an arrow between the two cuts in a pair.



The method for specifying which named vertices are to be used for paired points is by a **start name mask** (using wild cards and characters) to select the all the vertices that are to be the first points in a pair, and an **end name mask** (using wild cards and characters) to select all the vertices that are the second points in a pair.

For a x-section, all the named vertices of the selected **start points** are found and the points ordered by the offset of the point on the x-section. Then all the named vertices of the selected

end points are found and the points ordered by the offset of the point on the x-section.

The start points are then processed and each start point is paired with the next end point with a larger offset than the start point. It is possible that there are end points before the first start point (orphaned end points) and start points with no following end point (orphaned start points).

Up to twenty five different sets of name masks can be used so that different paired point sets can be labelled in different ways.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type**

Section: Pairing - Model/Name mask parameters

Define Set # input

set number to be used to define different name mask sets.

Start name mask input

text string containing the name masks to select the start points, each separated by one or more spaces, to test the x-section vertex names against. If the name includes spaces then it must be enclosed in the quotes " *(eg* "bench 2"*).*
Each mask can include wild cards (*) and wild characters (?).
For example: "ke" or,*
"?bank" or, if both masks are required,*
"ke ?bank*"*

End name mask input

text string containing the name masks to select the end points, each separated by one or more spaces, to test the x-section vertex names against. If the name include spaces then it must be enclosed in the quotes " *(eg* "bench 2"*)*
Each mask can include wild cards and wild characters.
For example: "ke" or,*
"?bank" or, if both masks are required,*
"ke ?bank*"*

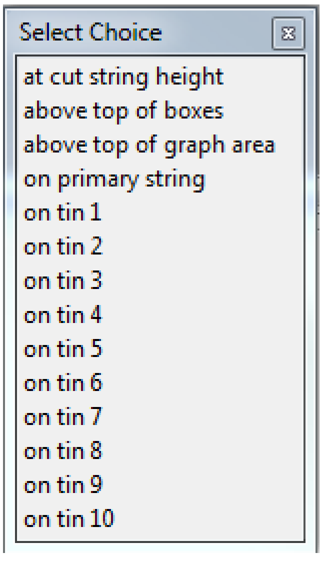
Note

*If **Start name mask** is blank and **End name mask** is not blank, then all vertices on the x-section that are not used as end points, are used as start points.*

*If **End name mask** is blank and **Start name mask** is not blank, then all vertices on the x-section that are not used as start points, are used as end points.*

*If **End name mask** and **Start name mask** are both blank, then all vertices on the x-section are used. Successive pairs of vertices taken to be the point pairs. There could be one remaining point (it has the largest offset) and if so, it is taken to be orphaned start point.*

Position **choice box**



where to get the z-values to use for the two paired points. This is used in conjunction with the choice in the **Adopted height** *field.*

If left blank, ***at point height*** *is used.*

Adopted height **pairing_n_height** **choice box** **lower position, higher position**

start position, end position

for the choice selected in the **Position** *field, there will be a z-value at the first point and another z-value at the second point of the pair. The* **Adopted height** *says which of the two z-value to use for placing symbols, lines between paired points, text from Attributes and the Offsets of the first point and the second point of a point pair.*

If lower (higher) position, the smaller (greater) of the two z-values will be used.
If start (end), the z-value of the first (second) cut of the cut point pair will be used.

If left blank, ***lower position*** *is used.*

Include orphaned start ? pairing_n_orphan_start **choice box** **yes, no**

if ***Yes*** *and there are start points after the last end point (and so can't be paired), then the last start point (i.e. the one with the largest offset) is taken as an orphaned start and it is labelled as though there is a matching end at the end of the x-section.*

if ***No*** *and there are start points after the last end point, then those start points are ignored.*

If left blank, ***No*** *is used.*

Include orphaned ends ? pairing_n_orphan_end **choice box** **yes, no**

if ***Yes*** *and there are end points before the first start point, then the first end point (the one with the smallest offset) is taken as an orphaned end and it is labelled as though there is a matching start at the beginning of the x-section.*
if ***No*** *and there are end points before the first start point, then those end points are ignored.*

If left blank, ***No*** *is used.*

See

26.3.18.1 Paired Points - Lines and Symbols
26.3.18.2 Paired Points - Labels
26.3.18.3 Paired Points - Start Offset
26.3.18.4 Paired Points - End Offset

Or return to 26.3 Section X Plot PPF Editor.

26.3.18.1 Paired Points - Lines and Symbols

Symbols can be placed at the start and end point offsets (at the z-value given by the **Position** and **Adopted height** columns when defining the Sets), and a line can also be drawn between the paired points at that z-value. Having both the line and a symbol forms an arrow between the paired points.

The columns for the fields documented in the sections are for.

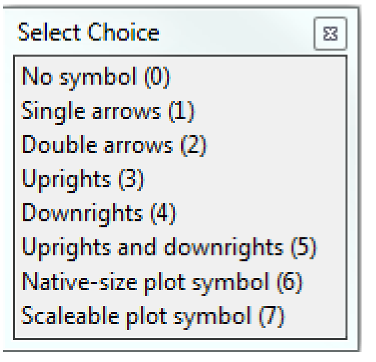
Panel Field **Parameter name** **Type**

Section: Pairing - Lines and symbol parameters

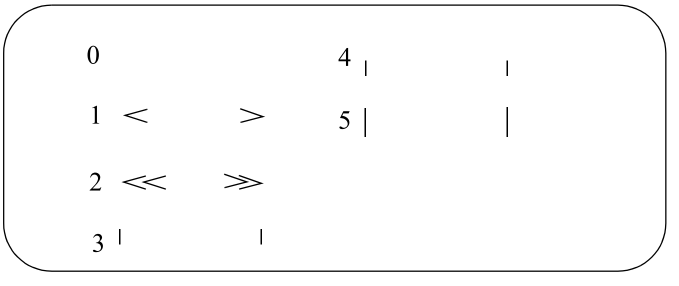
Use Set # **input**

set number as specified in the **Define set #***.*

Symbol type **pairing_line_symbols_n_symbol** **choice box**



type of symbol at each end of the points pair.



for cases 6 and 7, a plot symbol is to be used and is given in the Symbol field

The height used for the symbol at both ends is given by the **Position** *and* **Adopted height** *column in the Set definition.*

Symbol **pairing_line_symbols_n_user_symbol** **plot symbols**

for Symbol type 6 and 7, the plot symbol to be used at each paired point.

For Symbol type 0 to 5, this field is not used.

The height used for the symbol at both ends is given by the **Position** *and* **Adopted height** *columns in the Set definition.*

Symbol colour **pairing_line_symbols_n_symbol_colour** **colour box**

colour of the symbol

Draw line **pairing_line_symbols_n_line** **choice box** **No line, Draw line**

if Draw line, draw a line between the start and end points for each point pair.
If No line, no line is drawn between the start and end points for each point pair.

The height used for the line at both ends is given by the **Position** *and* **Adopted height** *columns in the Set definition.*

Line colour **pairing_line_symbols_n_line_colour** **colour box**

colour of the line between the start and end points for each point pair

Left symbol rotation pairing_line_symbols_n_left_rotate measure box

angle (measured counterclockwise from the positive x-axis with units of degrees in 4.17.1 HP Notation*) to rotate the symbol at the start point of the point pair.*

If there is no value then the rotation is 0.

Right symbol rotation pairing_line_symbols_n_right_rotate measure box

angle (measured counterclockwise from the positive x-axis with units of degrees in 4.17.1 HP Notation*) to rotate the symbol at the end point of the point pair.*

If there is no value then the rotation is 0.

X (mm) pairing_line_symbols_n_x measure box

horizontal adjustment to the position of the symbol.

Y (mm) pairing_line_symbols_n_y measure box

vertical adjustment to the position of the symbol.

Size pairing_line_symbols_n_size measure box

size of the symbol. A value of 0 means no symbol.

Continue to the next section 26.3.18.2 Paired Points - Labels or return to 26.3.18 Paired Points.

26.3.18.2 Paired Points - Labels

The columns for the fields documented in the sections are for.

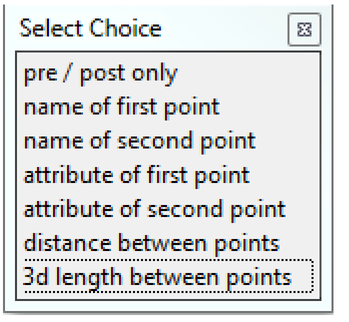
Panel Field **Parameter name** **Type**

Section: Pairing - Label parameters

Use Set # **input**

set number as specified in the **Define set #**.

Mode **pairing_label_n_mode** **choice box**



type of label to write between the start and end points of the points pair

The height used for the label is given by the **Position** *and* **Adopted height** *columns in the Set definition.*

Attribute name **pairing_label_n_attribute** **text box**

if **Mode** *is **attribute of first point** or **attribute of second point**, this is the name of the attribute to use as the label*

Offset **pairing_label_n_offset** **measure box**

the text is raised by this amount above the line between the point pair

Colour **pairing_label_n_colour** **colour box**

colour of the text

Size **pairing_label_n_size** **measure box**

size of the text. A value of 0 means no text.

Textstyle **pairing_label_n_textstyle** **textstyle box**

the textstyle for the text

Pre-text **pairing_label_n_pre_text** **text box**

text to draw before the label. This can include spaces, including one or more spaces after the last non blank character of the Pre-text.

Post-text **pairing_label_n_post_text** **text box**

text to draw after the label. This can include spaces, including one or more spaces before the first non blank character of the Post-text.

Decimals **pairing_label_n_no_decimals** **number box**

if the label is a number, then **Decimals** *is the number of decimal places to write the number out to.*

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Leave gap for text pairing_label_n_?? **tick box**

if ticked then a gap in the line between the points pair is left large enough for the label and the pre-text and post-text.

If not ticked then no gap is left in the line between the points pair.

Rotate text to fit pairing_label_n_roate_text **tick box**

if ticked, then if the label and the pre-text and post-text will not fit between the start point and end point of the point pair, the text is rotated through ninety degrees.

If not ticked, then the label text is drawn even though it will run over the ends of the point pair.

Continue to the next section 26.3.18.3 Paired Points - Start Offset or return to 26.3.18 Paired Points.

26.3.18.3 Paired Points - Start Offset

The offset (on the x-section) of the first point of the point pair can be labelled.

The text for the label consists of pre-text followed by Offset value followed by post-text.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type**

Section: Pairing - Start Offset parameters

Use Set # **input**

set number as specified in the **Define set #**.

The height used for the text is given by the **Position** *and* **Adopted height** *columns in the Set definition. This is then adjusted by the field* **Y (mm).**

X (mm) **pairing_start_offset_n_x** **measure box**

for placing the text: horizontal adjustment from the offset of the first cut.

Y (mm) **pairing_start_offset_n_y** **measure box**

for placing the text: the vertical adjustment to the height given by the **Position** *and* **Adopted height** *columns in the Set definition*

Angle (dms) **pairing_start_offset_n_angle** **measure box**

angle (measured counterclockwise from the positive x-axis with units of degrees in 4.17.1 HP Notation*) to rotate the text*

Colour **pairing_start_offset_n_colour** **colour box**

colour of the text

Size (mm) **pairing_start_offset_n_size** **measure box**

size of the text in millimetres. A value of 0 means no text.

Textstyle **pairing_start_offset_n_textstyle** **textstyle box**

the textstyle for the text

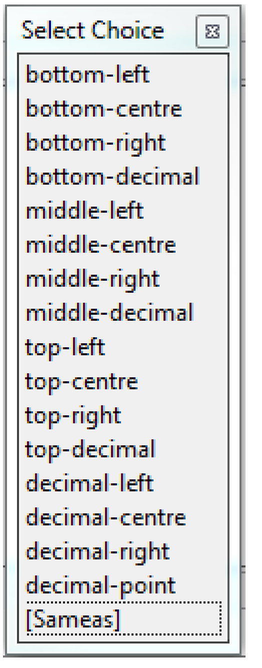
Pre-text **pairing_start_offset_n_pre_text** **text box**

text before the Offset value. This can include spaces, including one or more spaces after the last non blank character of **Pre-text**.

Post-text **pairing_start_offset_n_post_text** **text box**

text after the Offset value. This can include spaces, including one or more spaces before the first non blank character of the **Post-text**.

Justification **pairing_start_offset_n_justification** **justification box**



justification of the text.

Decimals **pairing_start_offset_n_no_decimals** **number box**

the number of decimal places for the Offset value

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Continue to the next section 26.3.18.4 Paired Points - End Offset or return to 26.3.18 Paired Points.

26.3.18.4 Paired Points - End Offset

The Offset (on the x-section) of the second point of the point pair can be labelled.

The label consists of pre-text then the Offset value followed by post-text.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type**

Section: Pairing - End Offset parameters

Use Set # **input**

set number as specified in the **Define set #**.

The height used for the text is given by the **Position** *and* **Adopted height** *columns in the Set definition. This is then adjusted by the field* **Y (mm).**

X (mm) **pairing_end_offset_n_x** **measure box**

for placing the text: horizontal adjustment from the Offset of the second point.

Y (mm) **pairing_end_offset_n_y** **measure box**

for placing the text: the vertical adjustment to the height given by the **Position** *and* **Adopted height** *columns in the Set definition*

Angle (dms) **pairing_end_offset_n_angle** **measure box**

angle (measured counterclockwise from the positive x-axis with units of degrees in 4.17.1 HP Notation*) to rotate the text*

Colour **pairing_end_offset_n_colour** **colour box**

colour of the text

Size (mm) **pairing_end_offset_n_size** **measure box**

size of the text in millimetres. A value of 0 means no text.

Textstyle **pairing_end_offset_n_textstyle** **textstyle box**

the textstyle for the text

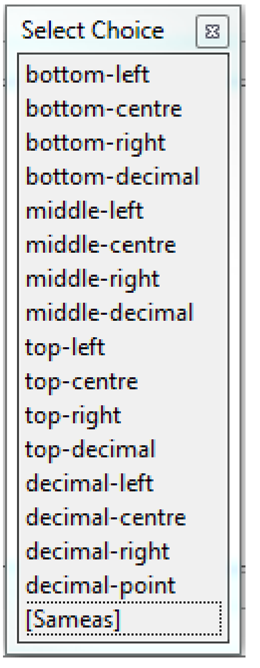
Pre-text **pairing_end_offset_n_pre_text** **text box**

text before the Offset value. This can include spaces, including one or more spaces after the last non blank character of **Pre-text.**

Post-text **pairing_end_offset_n_post_text** **text box**

text after the Offset value. This can include spaces, including one or more spaces before the first non blank character of **Post-text**.

Justification **pairing_start_offset_n_justification** **justification box**



justification of the text.

Decimals **pairing_start_offset_n_no_decimals** **number box**

the number of decimal places for the Offset value

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Please continue to the next section 26.3.19 PPFs To Include.

26.3.19 PPFs To Include

This is documented for all the PPF Editors in 26.2.10 PPFs To Include.

Please continue to the next section 26.3.20 Buttons at the Bottom of X-Plot Panel.

26.3.20 Buttons at the Bottom of X-Plot Panel

This is documented for all the PPF Editors in 26.2.11 Buttons at the Bottom of the PPF Editors.

Return to 26.3 Section X Plot PPF Editor.

26.4 Long Plot PPF Editor

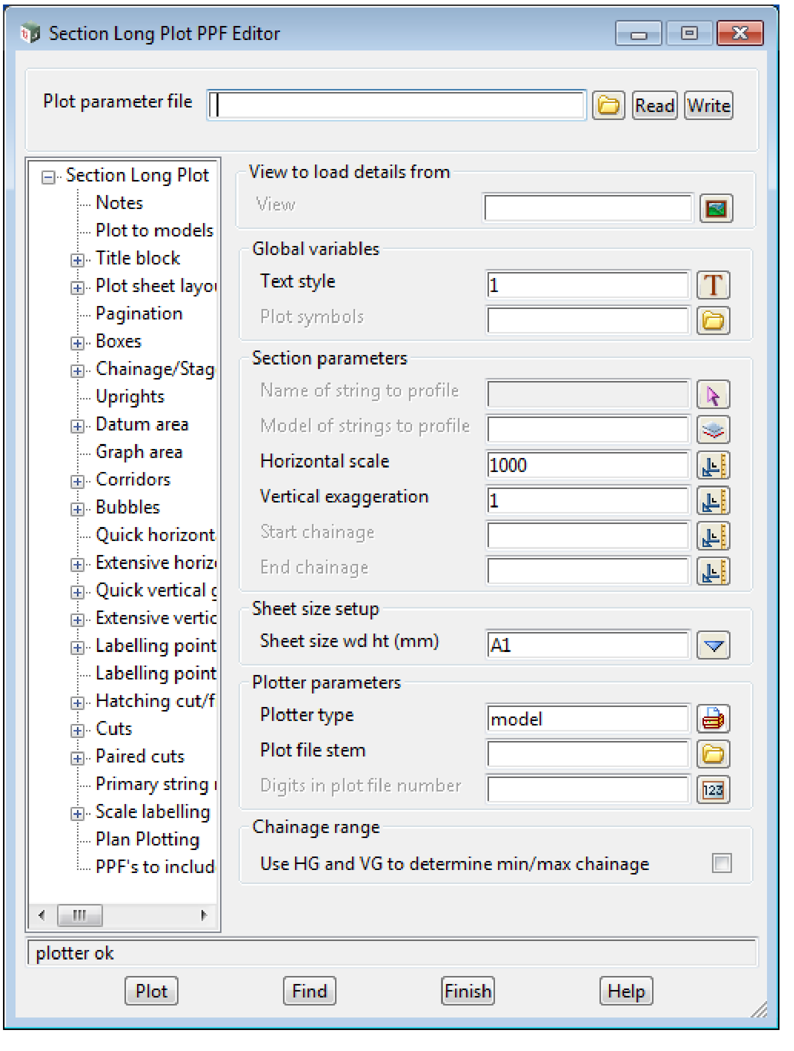
Position of option on menu: **Plot =>Plot and PPF Editors =>Long sections**

The **Long section PPF editor** is for creating and/or editing a (binary) long-section PPF and for creating a long section plot.

Note: Binary PPFs are stored *within* the project (not in the folder containing the project as the text ppf's were).

On selecting the **Long sections** option, the **Section Long Plot PPF Editor** panel is displayed.

The plot parameters for controlling the cross section plots are accessed by expanding to the appropriate node in the ***Section Long Plot*** tree (click on the **+** to expand to node or **-** to collapse the node) and then clicking on the required node, and the required information to fill in is displayed on the right hand side of the panel.



For information on all the different nodes see:

26.4.1 General Information on Long Section Plots

26.4.2 Section Long Plot - Front Page

26.4.3 Notes - Long Section

26.4.4 Plot to models - Long Section

26.4.5 Title Block - Long Section

26.4.1 General Information on Long Section Plots

A long section is normally generated along a given centreline and includes sections along the centre line through tins and services.

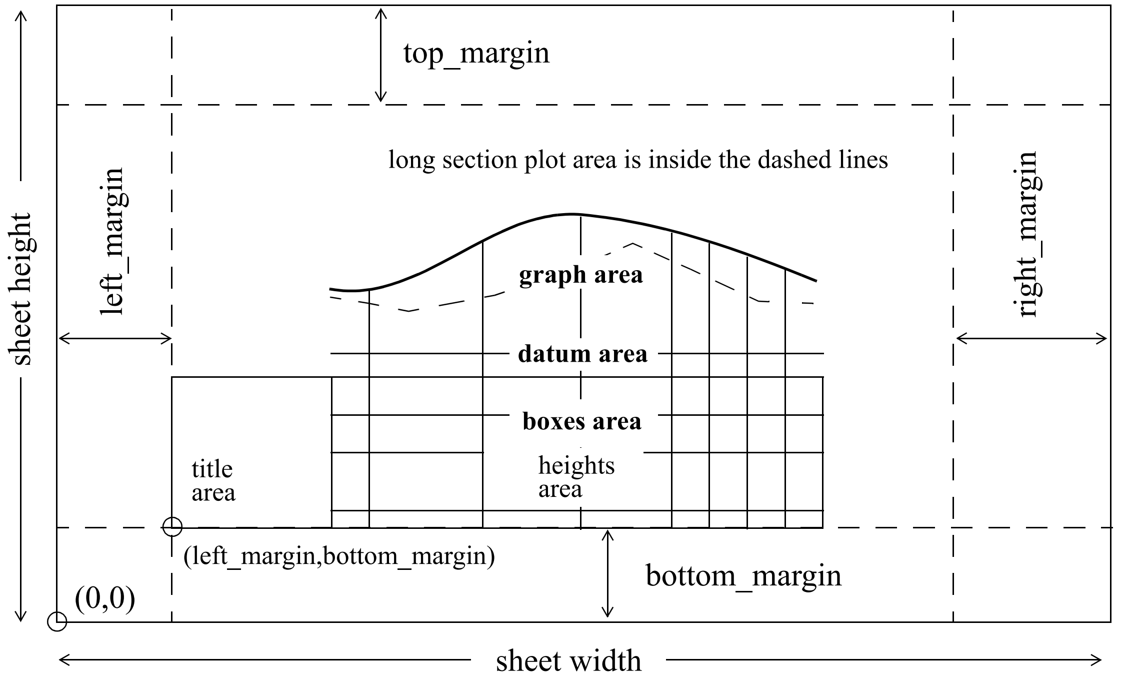
If the long section plot is too long to fit on one page, it can be broken into a number of pages (sheets) with a user defined overlap between the sheets.

The long plot on each page consists of the three regions - boxes, datum and graph areas.

The **boxes area** is where the titles and the chainage values and the heights/depths for the strings drawn on the long plot are labelled.

The **datum area** is the region between the boxes area and the graph area.

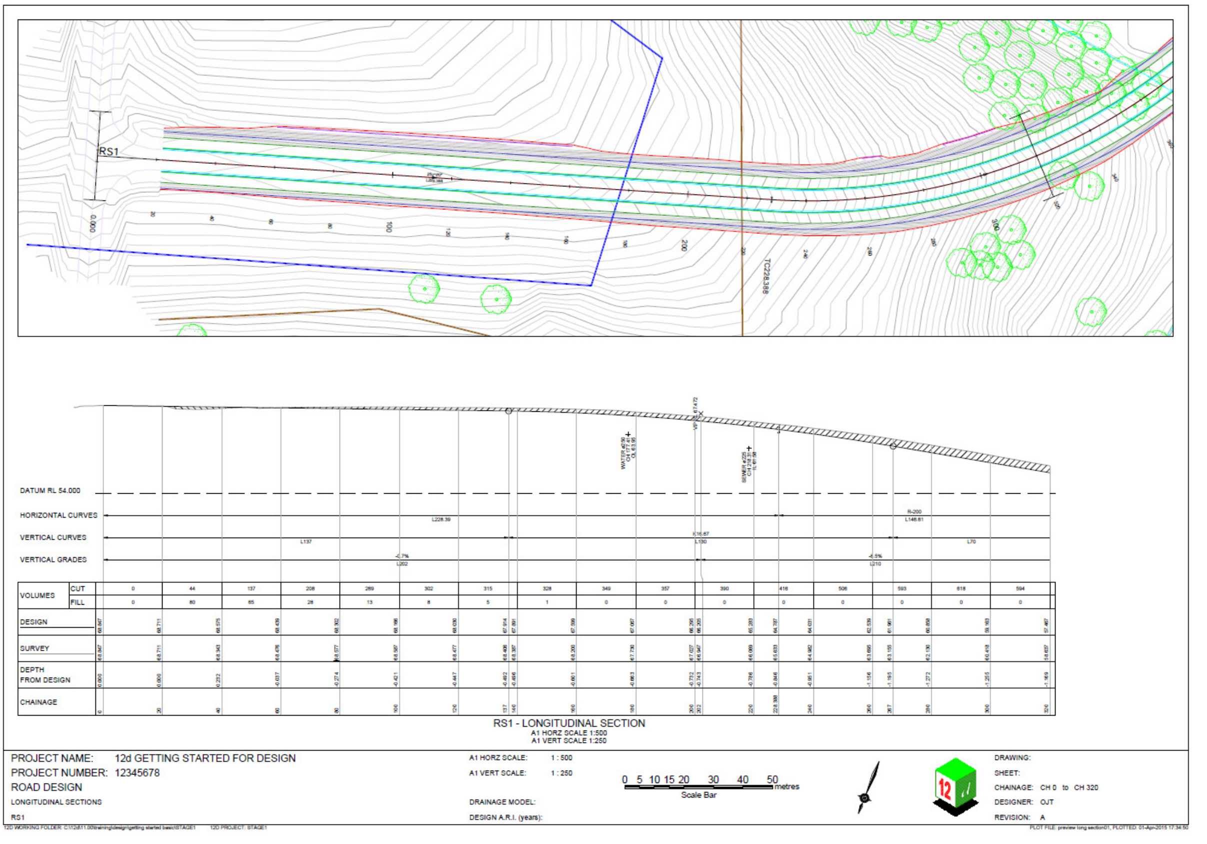
The **graph area** is the area where the actual plots of the strings are drawn.



Apart from information labelled in the boxes area, the long section plot can label other information such as

(a) horizontal geometry
(b) vertical geometry
(c) chainage, heights at special points
(d) symbols at special points
(e) bubbles at special chainages
(f) cuts the design string makes through strings
(g) cut and fill areas

A plot of the plan area that the long section traverses can also be automatically included on each page of the long section plot.



All the required parameters for controlling the long section plot are set up in the ***Long Plot PPF Editor*** and will be described in detail in the following sections.

Please continue to the next section 26.4.2 Section Long Plot - Front Page or return to 26.4 Long Plot PPF Editor.

26.4.2 Section Long Plot - Front Page

The long plot itself consists of the three regions - boxes, datum and graph areas.

The **boxes area** is where the titles and the chainage values and the heights/depths for the strings drawn on the long plot are labelled.

The **datum area** is the region between the boxes area and the graph area.

The **graph area** is the area where the actual plots of the strings are drawn.

Apart from information labelled in the boxes area, the long section plot can label other information including

(a) horizontal geometry

(b) vertical geometry

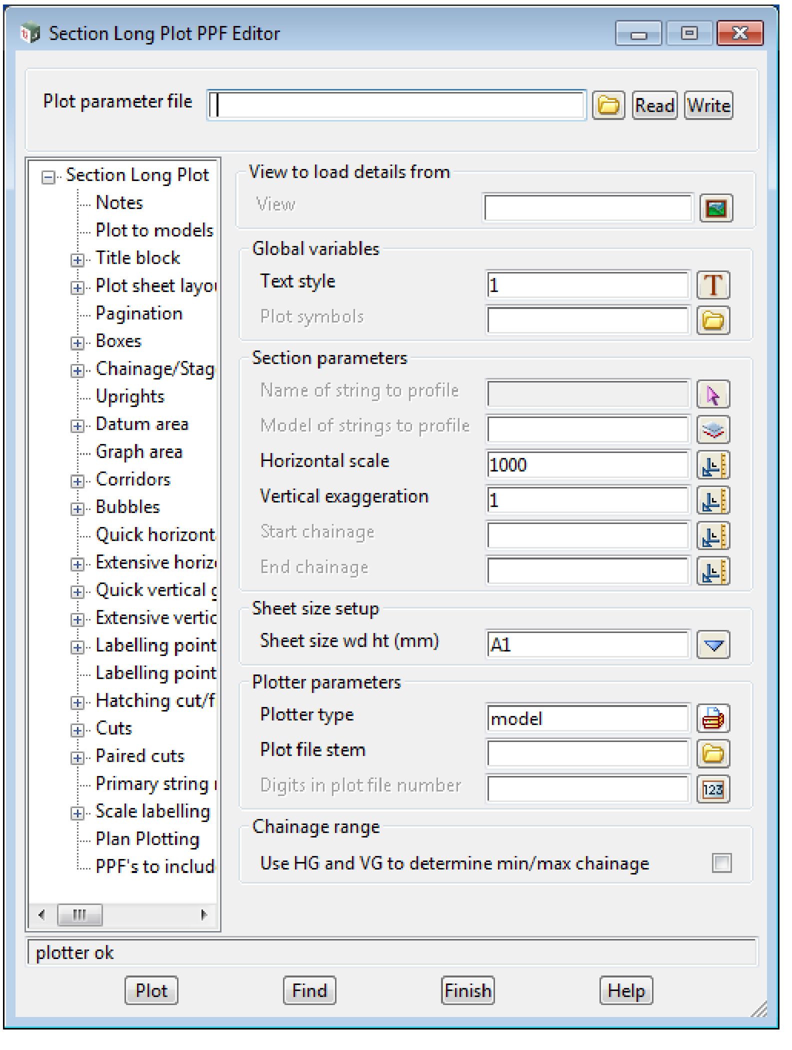
(c) chainage, heights at special points

(d) symbols at special points

(e) bubbles at special chainages

(f) cuts the design string makes through strings

(g) cut and fill areas



Section: Plot parameter file

*a **plot parameter file** can be used to load values into the fields of the PPF Editor, or as a file to write out all current values in a PPF Editor to.*

This section is documented for all the PPF Editors in 26.2.1 Plot Parameter File.

Section: Section: View to load details from AND Global variables

*a **section view** can be selected to load certain values into fields of the PPDF Editor. For example, Vertical exaggeration and Corridor models. And there are variables to use if some values aren't given.*

These sections are documented for all the PPF Editors in 26.2.2 View to Load and Global Variables.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Section parameters

Name of string to profile **string_to_plot** **string select box**

if selected, a long section plot is produced for this string.

Model of strings to profile **model_to_plot** **model box**

if not blank, a long section plot is produced for each string in the model.

Horizontal Scale **scale** **input**

horizontal scale to be used for the long section plots

Vertical exaggeration **vertical_exaggeration** **input**

vertical scale to be used for the long section plots

Start chainage **start_chainage** **input**

start chainage of the string to start the long section plots at

End chainage **end_chainage** **input**

end chainage of the string to end the long section plots at.

Section: Sheet size setup and Plotter parameters

these sections define the size of the "paper" to plot on, the type of plotter to use and the naming to use for the plot files.

These sections are documented for all the PPF Editors in 26.2.3 Sheet Size and Plotter Parameters.

Section: Chainage range

Use HG VG for min, max determines whether the horizontal geometry (HG) and the vertical geometry (VG) are both used to determine the minimum and maximum chainages for drawing. This allows the vertical geometry to be plotted when it is outside the horizontal geometry (e.g. kerb returns).

Use HG and VG to determine min/max chainage

use_hg_vg_for_min_max **tick box**

if ticked, the min/max chainage will be determined by the Horizontal and Vertical geometry.

Please continue to the next section 26.4.3 Notes - Long Section.

26.4.3 Notes - Long Section

This is documented for all the PPF Editors in 26.2.4 Notes.

Please continue to the next section 26.4.4 Plot to models - Long Section.

26.4.4 Plot to models - Long Section

This is documented for all the PPF Editors in 26.2.5 Plot to models.

Please continue to the next section 26.4.5 Title Block - Long Section.

26.4.5 Title Block - Long Section

This is documented for all the PPF Editors in 26.2.6.1 Title Block Section in PPF Editors.

For more general information about a title block, see 26.2.6 Title Block.

This is documented for all the PPF Editors in 26.2.6 Title Block.

Title Block - Symbols

Panel field	Parameter name	Type	Pop-Up
Symbol			
Scale mode			
Scale			
Rotate with plot			
Rotation			
Colour			
X			
Y			

Please continue to the next section 26.4.6 Plot Sheet Layout - Long Section.

26.4.6 Plot Sheet Layout - Long Section

Plot Sheet Layout - Margins

Section: Margins for standard 12d title file - Border gaps

If the default 12d title block is used, then the size of the bottom of title block depends on the text size. The following parameters are used in the default title block case and the bottom_border_gap is added to the calculated height of the bottom of the title block.

Left (mm) left_border_gap real box

left border gap (in millimetres).

Right (mm) right_border_gap real box

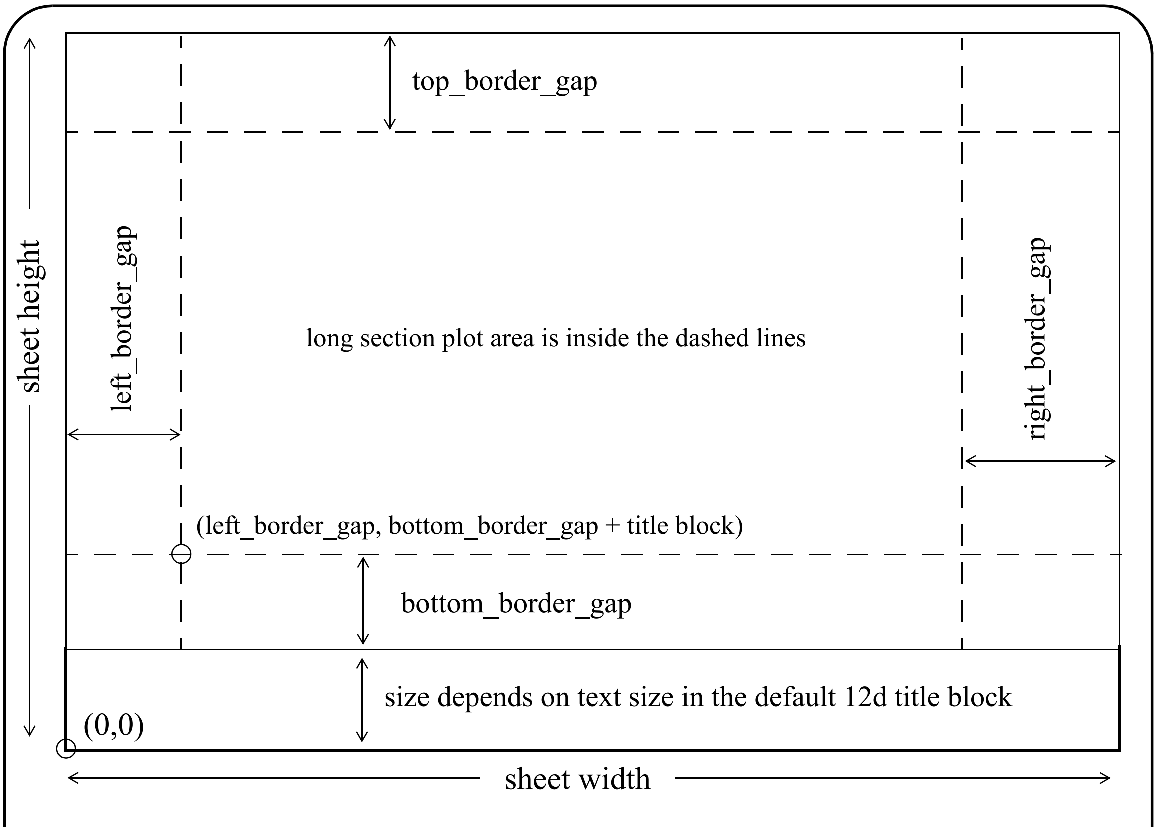
right border gap (in millimetres).

Top (mm) top_border_gap real box

top border gap (in millimetres).

Bottom (mm) bottom_border_gap real box

bottom border gap (in millimetres).



Definition of Plotting Areas for Default 12d Title Block

Section: Margins for user title file

Left (mm) left_margin input

left margin (in millimetres).

Right (mm) right_margin input

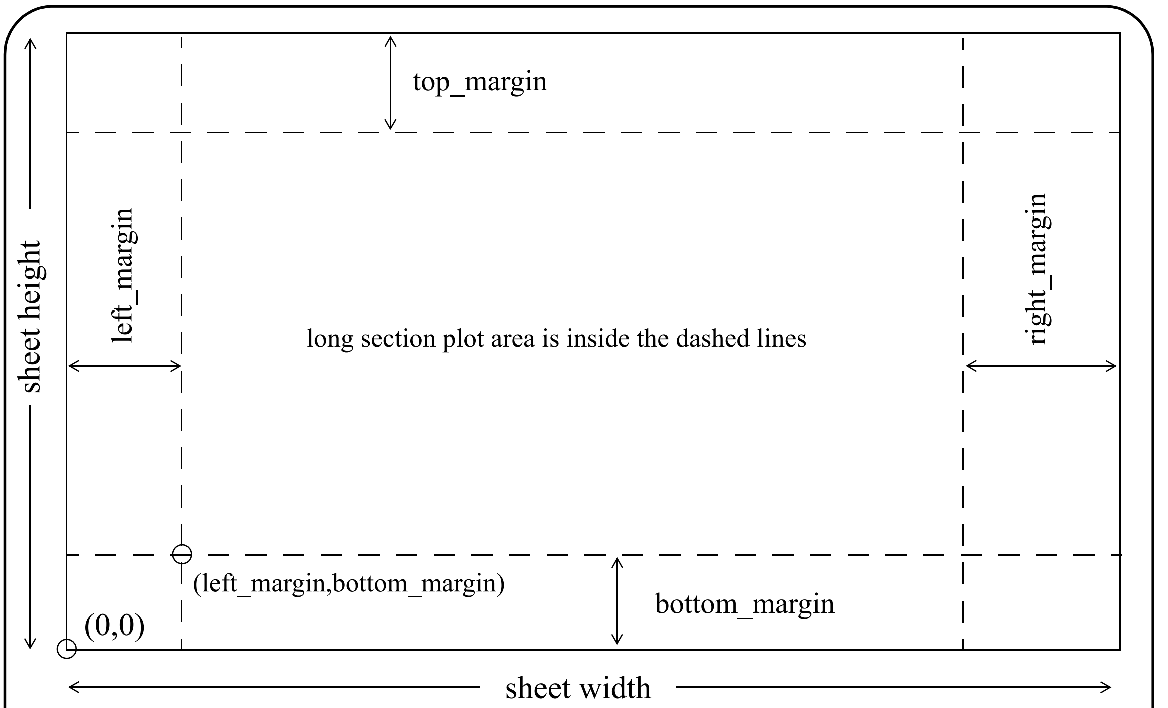
right margin (in millimetres).

Top (mm) top_margin input

top margin (in millimetres).

Bottom (mm) bottom_margin input

bottom margin (in millimetres).



Definition of Plotting Areas for User Defined Title Blocks

Because the user can easily select from the plotting panel whether a User Defined Title Block or the default 12d title block is used, both sets of margin and gap parameters can exist in the one plot parameter file.

The (left_margin,bottom_margin) defines the left hand corner position of the long plot on the plot sheet.

The *right_margin* and *top_margin* need not be set and if missing, will be calculated from the other plot parameters defining the plot layout.

If the sheet sizes are missing, they will also be automatically calculated.

Please continue to the next section 26.4.7 Pagination.

26.4.7 Pagination

If the long section plot is too long to fit on one page, it can be broken into a number of pages (sheets).

The parameter, *pagination_length*, controls the amount of new chainage length on each plot page. Each page of the long section plot can also included a set chainage amount from the end of the **previous** plot.

Hence apart from the first page and possibly the last page, the plot will have a chainage length given by the sum of the *pagination_length* and *pagination_overlap*.

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Pagination parameters

Panel Field	Parameter name	Type	Pop-Up
Use pagination	**pagination**	**tick box**	
if ticked, then break the plot into pages.			
Pagination mode	pagination_mode	choice	chainage millimetres
length and overlap units.			
Chainage length	**pagination_length**	**input**	
new chainage range on each page.			
Chainage overlap	**pagination_overlap**	**input**	
overlap between pages.			

Note:
The *pagination_length* and *pagination_overlap* are given in chainage units **or** millimetres.The user must choose values which will fit on the selected sheet size or the end of each plot will be truncated by the sheet.

Please continue to the next section 26.4.8 Boxes.

26.4.8 Boxes

Each string in the long section plot can be labelled with one or two lines of title, and the chainages/heights/depths at the user specified chainages for the strings.

The **title** for the strings, is drawn in the **title area** of the **boxes area**.

The **chainages/heights/depths** are drawn in the **heights area** of the **boxes area**.

Consequently the boxes area is made up of rows of text consisting of:

string/tin titles followed by the **chainage/height/depth** values along the string.

Each row is surrounded by lines to form a box.

The default order of the boxes from the bottom up is

(a) optional super-elevation diagram

(b) chainage values

(c) can be primary string heights - user choice

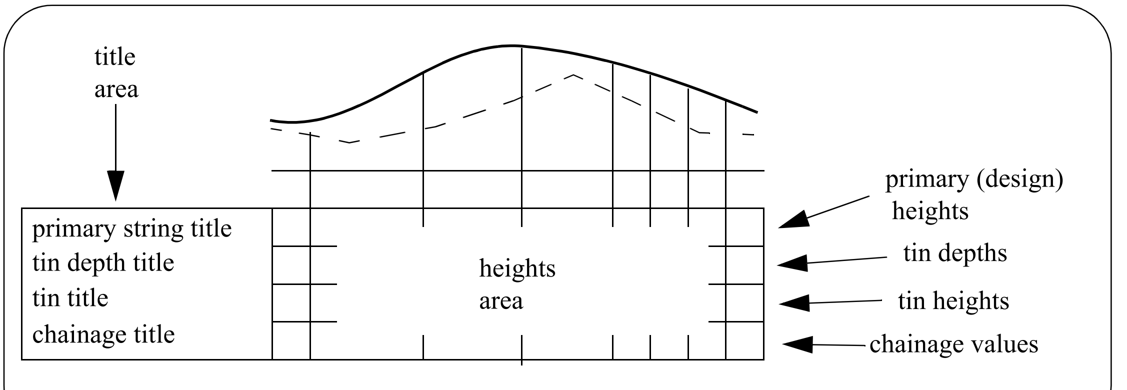
(d) tin heights and depths

(e) offset heights and depths

(f) optional volumes or earth works

(g) can be primary string heights - user choice

(h) zero or more blank boxes



boxes for long section plot

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Linework parameters

Panel Field	**Parameter name**	**Type**	**Pop-Up**
Draw box mode	**draw_box_mode**	**choice box**	**Do not draw any box lines** Draw box lines around height area only **Draw box lines around** title and height area only **Use parameters defined in** boxes - heights

mode for drawing the boxes for the long section plot.

Default box colour **box_colour** **colour box**

default box colour

Number of blank boxes **number_of_blank_boxesinput**

number of boxes to be left blank

The blank boxes are used to place other information in (such as horizontal or vertical geometry) or for other user supplied information.

Section: Primary string parameters

The primary string (the design string) is used to define

(a) the design long section

(b) the chainage positions for labelling heights and drawing uprights

(c) the section line used for sectioning through tins

(d) the section line for defining the corridor for services

Although the primary string is used to set up most of the information for the long section plot, it doesn't have to be drawn on the long section.

The drawing or not drawing of the primary string on each cross section plot is controlled by the parameter *primary_string*.

The **colour** of the primary string in the plot is the actual primary string colour.

Sometimes the primary string is only required to define chainages or the horizontal path for tin sections. In this case, the primary string would not be drawn on the long section plot.

If the primary string is drawn, it is automatically labelled.

Also the drawing of crosses at the vertical intersection points can be controlled from the ppf file.

Draw and label the primary string **primary_string** **tick box**

if ticked, the primary string will be drawn and labelled.

Draw crosses at VIPs **primary_draw_vips** **tick box**

if ticked, crosses will be draw at the vertical intersection points.

The position of the primary string label box can be either straight after the chainage box, or after the tin and offset boxes but before the blank boxes.

Placing the primary string label before the bank boxes or just after the chainage box in controlled by the parameter *primary_label_mode*.

Primary string location primary_label_mode choice box in last box before blank boxes

in first box above chainages

position of the primary string label.

Although the order of the boxes may appear to be fixed, in practice they can be in any order. This is possible because for each box, there is a parameter to set the height in millimetres from the bottom of all the boxes that the text in the box is drawn at (the *_y_pos* parameters).

The roundoff for the datum value is specified by the user (default 1.0) and the datum is automatically calculated for each sub-plot, and labelled.

See

26.4.8.1 Boxes - Title Area
26.4.8.2 Boxes - Values Area
26.4.8.3 Boxes - Outside Linework
26.4.8.4 Boxes - Inside Linework
26.4.8.5 Boxes - Chainage Titles/Values
26.4.8.6 Boxes - Primary String Titles/Heights

Or return to 26.4 Long Plot PPF Editor.

26.4.8.1 Boxes - Title Area

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Boxes - Title area parameters

The **title area** starts at the co-ordinate (left_margin,bottom_margin).

The size of the title text is given by the ***title_box_text_size*** parameter.

The **width** of the title area is either given by the *space_for_titles* parameter, or if omitted, the required width is automatically calculated.

Size of titles (mm) **title_box_text_size** **input**

size of offset, height label and values

Space for titles (mm) **space_for_titles** **input**

size of title area.

The x position of the title text is the same for all the lines of title text and can be set to be a fixed distance from the left hand side of the boxes.

X adjustment (mm) **box_titles_x** **input**

distance to move the title text from the left hand side of the boxes

The y position of the title text can be set separately for each type of title. The parameters are given later under each of the title types (*e.g.* chainages, primary string, tins, depths *etc.*).

After the title area there can be a user defined gap, followed by the heights area

Distance between title and values area (mm) box_gap input

distance between title and values area in mm.

The height text is written at right angles to the bottom of the boxes. It can be either top or bottom justified with respect to the box (*box__justification*).

Continue to the next section 26.4.8.2 Boxes - Values Area or return to 26.4.8 Boxes.

26.4.8.2 Boxes - Values Area

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Boxes - Value area parameters

The **values area** starts at the end of the title area.

The height text is written at right angles to the bottom of the boxes. It can be either top or bottom justified with respect to the box (*box_text_justification*).

The number of decimal places (*number_of_decimals*), and the size of the heights text (*box_text_size*) can be specified globally but there is a parameter for each box which overrides these defaults for each box of heights.

Justification box_text_justification choice box bottom of individual boxes
top of individual boxes

justification of height text.

Number of decimals **number_of_decimals** **input**

number of decimal places in the height boxes. If <0, the absolute value is taken as the number of decimal places i.e. no trailing zeros are removed for the values in the heights area.

Text size (mm) **box_text_size** **input**

distance to move the title text from the left hand side of the boxes

Horizontal line spacing (mm)

horizontal_line_spacing input

height of the individual height boxes.

A global height for the individual boxes is either given by the *horizontal_line_spacing* parameter, or if omitted, a height to fit the largest height or depth value is calculated and used as the default box height.

However, the height of each box can be individually set by parameters *box_size_n* where the boxing numbering, n. starts from the bottom box. The value of *horizontal_line_spacing* is used for any of the *box_size_n* parameters not specified.

Label tin/offset string heights by default

label_heights tick box

Label tin/offset string depths by default

label_depths tick box

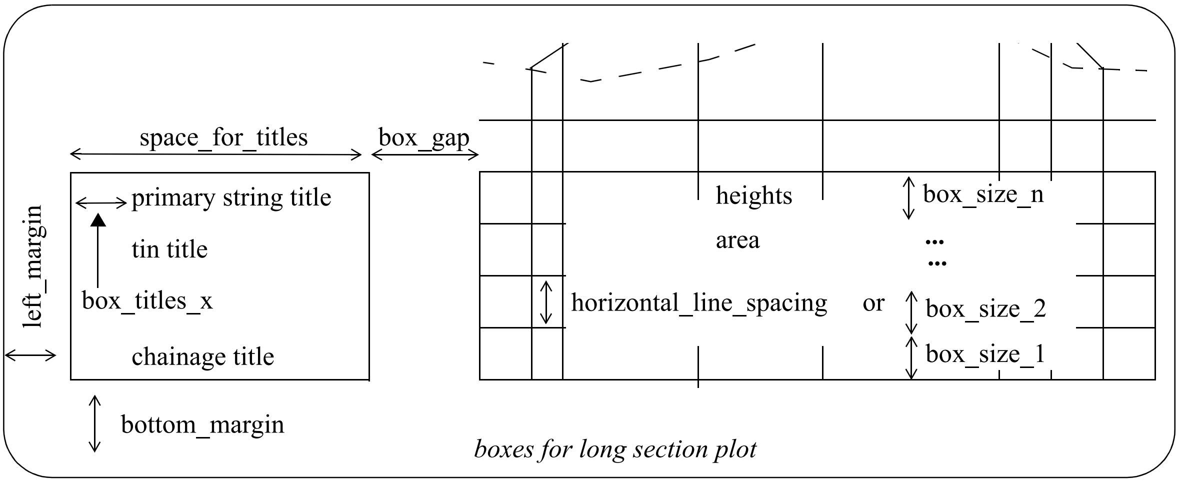
Box # **input**

specifying the nth value

Size (mm) **box_size_n** **input**

height of the nth box, numbered from bottom up.

The total height of the boxes area is simply given by the sum of the heights of each box.



The **width** of the **heights area** is determined by the number of chainages to be labelled and whether the values are staggered to prevent over writing.

Hence the total width of the boxes area is the width of the **title area** plus box_gap, plus the width of the **heights area**.

Many distance definitions in the plot parameter file are given in terms of distance above the **top** of the boxes area so that the distances are independent of the number of boxes and box sizes.

Continue to the next section 26.4.8.3 Boxes - Outside Linework or return to 26.4.8 Boxes.

26.4.8.3 Boxes - Outside Linework

The drawing of the box line work, the box colour, the position of the primary string labels and the number of blank boxes are all set by parameters.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Boxes - Outside linework parameters

Draw the left side of title area(1)

draw_box_side_1 **tick box**

if ticked, draw the left side of the title area (def).

Draw the top line of title area(2)

draw_box_side_2 **tick box**

if ticked, draw the top line of the title area (def).

Draw the top line of title area(3)

draw_box_side_3 **tick box**

if ticked, draw right side of the title area (def).

Draw the bottom line of title area(4)

draw_box_side_4 **tick box**

if ticked, draw the bottom line of the title area (def).

Draw the left side of heights area(5)

draw_box_side_5 **tick box**

if ticked, draw the left side of the heights area (def).

Draw the top line of heights area(6)

draw_box_side_6 **tick box**

if ticked, draw the top line of the heights area (def).

Draw right side of heights area(7)

draw_box_side_7 **tick box**

if ticked, draw right side of the heights area (def).

Draw bottom side of heights area(8)

draw_box_side_8 **tick box**

if ticked, draw bottom side of the heights area (def).

Colour (1) **box_side_colour_1** **colour box**

colour to draw left side of title area

Colour (2) **box_side_colour_2** **colour box**

colour to draw top of title area

Colour (3) **box_side_colour_3** **colour box**

colour to draw right side of title area

Colour (4) **box_side_colour_4** **colour box**

colour to draw bottom of title area

Colour (5) **box_side_colour_5** **colour box**

colour to draw left side of heights area

Colour (6) **box_side_colour_6** **colour box**

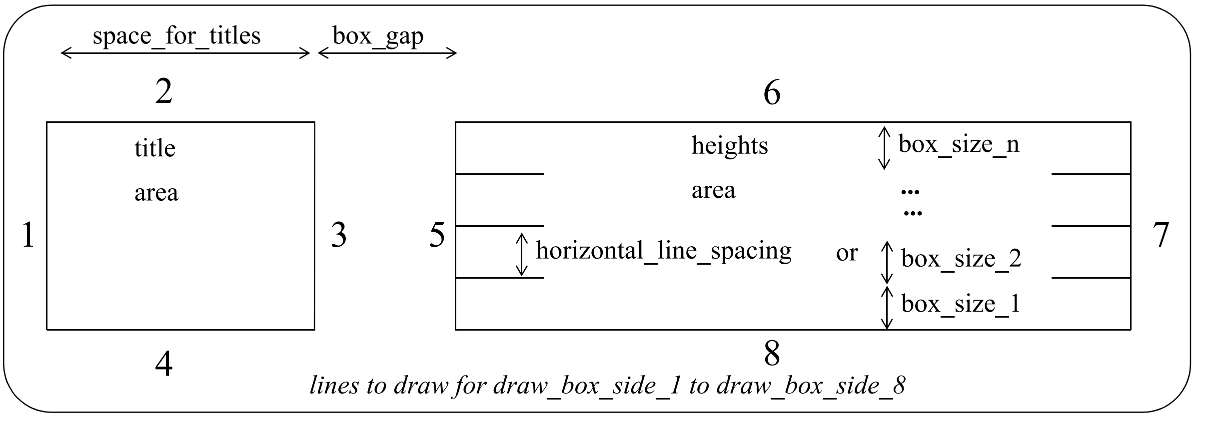
colour to draw top of heights area

Colour (7) **box_side_colour_7** **colour box**

colour for right side of heights area

Colour (8) **box_side_colour_8** **colour box**

colour for bottom of heights area



Continue to the next section 26.4.8.4 Boxes - Inside Linework or return to 26.4.8 Boxes.

26.4.8.4 Boxes - Inside Linework

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Boxes - Inside linework parameters

The lines at the top of the individual boxes inside the title area and heights area (separation lines) are controlled by the parameters *box_line_draw_mode* and *box_line_mode_n*.

The separation lines can be drawn just in the title area, just in the heights area or in both areas.

The parameter *box_line_draw_mode* can be set to control all the separation lines but there are additional parameters, *box_line_mode_n*, which override *box_line_draw_mode* for each of the individual boxes where n = 1, ... number of boxes -1.

The top of the top box is not controlled by box_line_mode_n but is controlled by the parameters *draw_box_side_2* and *draw_box_side_6.*

Separation line mode **box_line_draw_mode** **choice box** **Do not draw any separation**

lines

Draw the separation lines in

both areas

Draw the separation lines in

the title area only

Draw the separation lines in

the heights area only

the separation line draw mode.

Box # **input**

Where box # = 1 to number of boxes(n). Box 1 is the bottom box, increasing upwards.

Line mode **box_line_mode_n** **choice box** **No top line for title or height**

area (mode=0)

Draw top line for title or

height area (mode=1)

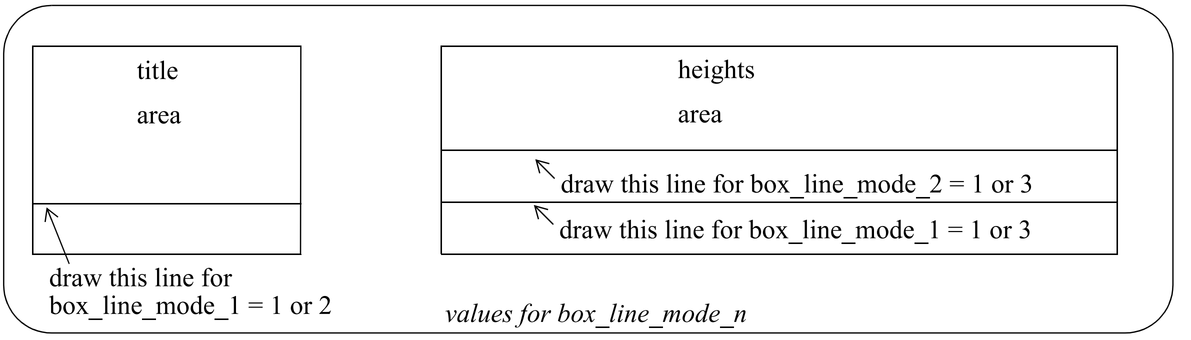
Draw top line for title area

only (mode=2)

Draw top line for height area

only (mode=3)

line mode for top of title and height areas for box number specified.



Continue to the next section 26.4.8.5 Boxes - Chainage Titles/Values or return to 26.4.8 Boxes.

26.4.8.5 Boxes - Chainage Titles/Values

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Chainage title parameters

Title line 1 **chainage_title** **input**

1st line of chainage title.

Title line 2 **chainage_title_2** **input**

second line of chainage title.

Textstyle **chainage_title_textstyle** **text box**

Textstyle for the chainage title

Colour **chainage_title_colour** **colour box**

colour of the chainage title

Text size (mm) **chainage_title_size** **input**

size of the chainage title

Text y position (mm) chainage_title_y_pos input

if set, the height in mm above the bottom of all the boxes that the chainage title text is drawn. If not set, then the text is placed at a height that puts it inside the default box for the chainage.

Section: Chainage value parameters (raw/running chainage)

Decimal places **chainage_decimals** **input**

number of decimal places for chainages.

If > 0, trailing zeros are ***removed*** *after the decimal point.*
If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Textstyle **chainage_textstyle** **text box**

textstyle for the chainage values

Colour **chainage_colour** **colour box**

colour of the chainage values

Text size (mm) **chainage_size** **input**

size of the chainage values

Text y position (mm) **chainage_y_pos** **input**

if set, the height in mm above the bottom of all the boxes that the chainage values are drawn. If not set, then the text is placed at a height that puts it inside the default box for the chainage.

Continue to the next section 26.4.8.5.1 Boxes - Chainage Title/Values - Equality Chainage Values or return to 26.4.8 Boxes.

26.4.8.5.1 Boxes - Chainage Title/Values - Equality Chainage Values

Section: Equality chainage values

Label with equality chainages chainage_label_eq tick box

Decimal places chainage_decimals_eq input

1000s separator (non K-post) chainage_thousands_separator_eq input

Zero-pack digits after 1000s separator chainage_zero_pack_eq tick box

Show K-Post at non-zero offset (if defined) chainage_name_include_eq tick box

K-post pre text chainage_name_pre_eq input

K-post post text chainage_name_post_eq input

Space before offset of K-post chainage_offset_space_eq tick box

Plus sign before positive offset of K-post chainage_plus_eq tick box

Show equality zone (if defined) chainage_zone_include_eq tick box

Space before equality zone chainage_zone_space_eq tick box

Equality zone pre text chainage_zone_pre_eq input

Equality zone post text chainage_zone_post_eq input

Before/after equality separator chainage_equals_separator_eq input

Textstyle chainage_textstyle_eq input

Colour chainage_colour_eq colour box

Text size (mm) chainage_size_eq input

Text y position (mm) chainage_y_pos_eq input

Continue to the next section 26.4.8.6 Boxes - Primary String Titles/Heights or return to 26.4.8 Boxes.

26.4.8.6 Boxes - Primary String Titles/Heights

Sometimes the primary string is only required to define chainages or the horizontal path for tin sections. In this case, the primary string would not be drawn on the long section plot.

If the primary string is drawn, it is automatically labelled.

Also the drawing of crosses at the vertical intersection points can be controlled from the ppf file.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Primary string title parameters

Title line **primary_title** **input**

1st line of string title.

Title line 2 **primary_title_2** **input**

second line of string title. Default is primary string name.

Textstyle **primary_title_textstyle** **text box**

textstyle for the primary string title

Colour **primary_title_colour** **colour box**

colour of the primary string title

Text size (mm) **primary_title_size** **input**

size of the string title

Text y position (mm) **primary_title_y_pos** **input**

if set, the height in mm above the bottom of all the boxes that the primary string title text is drawn. If not set, then the text is placed at a height that puts it inside the default box for the primary string.

Section: Primary string height parameters

Decimal places **primary_decimals** **input**

number of decimal places for height values.

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Textstyle **primary_textstyle** **text box**

textstyle for the height values

Colour **primary_colour** **colour box**

colour of the height values

Text size (mm) **primary_size** **input**

size of the height values

Text y position (mm) **primary_y_pos** **input**

if set, the height in mm above the bottom of all the boxes that the primary string height values are drawn. If not set, then the text is placed at a height that puts it inside the default box for the primary string.

Search dist when no z (m) **primary_ch_tolerance** **input**

for a given chainage, if no z value exists, this distance is added to/subtracted from the chainage to search for a valid z value.

Continue to the next section 26.4.8.7 Boxes - Tin Titles/Heights/Depths or return to 26.4.8 Boxes.

26.4.8.7 Boxes - Tin Titles/Heights/Depths

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Tin titles/heights/depths parameters

Label depth default	**label_depths**	**tick box**	

if ticked, the depths are labelled.

Define tin set #		**input**	

where n = 1 to.... The set enables the specification of a number of parameters for a number of specified tin names.

Tin name	**tin_n_name**	**tin box**	**available tins**

the name of the nth tin to be used for labelling.

If a tin of the name given by ***tin_n_name*** **does** not exist, then the plot is not produced and an error message is given.

Tin draw mode	**tin_n_draw_mode**	**choice box**	**Draw the tin**
			Do not draw the tin

draw tin mode for the nth tin specified by tin set #.

Tin colour	**tin_n_draw_colour**	**colour box**	

tin colour mode for the nth tin specified by tin set #.

Tin label mode	**tin_n_label**	**choice box**	**Label the tin**
			Do not label the tin

tin label mode for the nth tin specified by tin set #.

See

26.4.8.7.1 Boxes - Tin Titles/Heights/Depths - Titles
26.4.8.7.2 Boxes - Tin Titles/Heights/Depths - Heights
26.4.8.7.3 Boxes - Tin Titles/Heights/Depths - Depths

Or return to 26.4.8 Boxes.

26.4.8.7.1 Boxes - Tin Titles/Heights/Depths - Titles

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Tin - Title parameters

Use tin set # **input**

set number to be used to define sets of tin parameters i.e. n value

First line of tin title **tin_n_title** **input**

first line of nth tin title

Second line of tin title **tin_n_title_2** **input**

second line of nth tin title

Tin title textstyle **tin_n_title_textstyle** **text box**

nth tin title textstyle

Tin title colour **tin_n_title_colour** **colour box**

nth tin title colour

Tin title size (mm) **tin_n_title_size** **input**

nth tin title size

Title Y position (mm) **tin_n_title_y_pos** **input**

if set, the height in mm above the bottom of all the boxes that the nth tin title text is drawn. If not set, then the text is placed at a height that puts it inside the default box for the tin heights.

Continue to the next section 26.4.8.7.2 Boxes - Tin Titles/Heights/Depths - Heights or return to 26.4.8 Boxes.

26.4.8.7.2 Boxes - Tin Titles/Heights/Depths - Heights

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Tin - Height parameters

Use tin set # **input**

set number to be used to define sets of tin parameters i.e. n value

Decimals **tin_n_decimals** **input**

number of dec places in nth tin height.

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Textstyle **tin_n_textstyle** **text box**

textstyle of nth tin height

Colour **tin_n_colour** **colour box**

nth tin height and depth colour

Size (mm) **tin_n_size** **input**

nth tin height size

Y pos (mm) **tin_n_y_pos** **input**

if set, the height in mm above the bottom of all the boxes that the tin height text is drawn.If not set, then the text is placed at a height that puts it inside the default box for the tin heights.

Continue to the next section 26.4.8.7.3 Boxes - Tin Titles/Heights/Depths - Depths or return to 26.4.8 Boxes.

26.4.8.7.3 Boxes - Tin Titles/Heights/Depths - Depths

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Tin depth parameters

Use tin set # **input**

set number to be used to define sets of tin parameters i.e. n value

Depth label mode **tin_n_depth_label** **choice box** **Label depths**

Do not label depths

depth label mode for the nth tin specified by set #.

First line of tin depth title **tin_n_depth_title** **input**

first line of nth tin depth title

Second line of tin depth title **tin_n_depth_title_2** **input**

second line of nth tin depth title

Depth title textstyle **tin_n_depth_title_textstyle** **text box**

nth tin depth title textstyle

Depth title colour **tin_n_depth_title_colourcolour box**

nth tin depth title colour

Depth title size (mm) **tin_n_depth_title_size** **input**

nth tin depth title size

Depth title Y position (mm) **tin_n_depth_title_y_posinput**

if set, the height in mm above the bottom of all the boxes that the tin depth title text is drawn. If not set, then the text is placed at a height that puts it inside the default box for the tin heights.

Decimals in depth value **tin_n_depth_decimals input**

number of dec places in tin height.

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Depth value textstyle **tin_n_depth_textstyle** **text box**

nth tin depth textstyle

Depth value colour **tin_n_depth_colour** **colour box**

nth tin depth colour

Depth value size (mm) **tin_n_depth_size** **input**

nth tin depth size

Depth value Y position (mm) **tin_n_depth_y_pos** **input**

if set, the height in mm above the bottom of all the boxes that the tin depth text is drawn. If not set, then the text is placed at a height that puts it inside the default box for the tin heights.

Multiplier for positive depths **depth_positive_factorinput**

if set, this value will be used to multiply positive depth values.

Multiplier for negative depths **depth_negative_factorinput**

if set, this value will be used to multiply negative depth values.

The depth from the primary string to a tin, at a particular offset is defined as

depth = tin height value - height of the primary string

That is, the depth that the primary string is **below** the tin.

Before plotting, the value of depth is multiplied by either the *depth_positive_factor* or *depth_negative_factor*.

if (depth >= 0) plotted_depth_value = depth * depth_positive_factor

if (depth < 0) plotted_depth_value = depth * depth_negative_factor

Hence the definition of depth can be modified by the parameters:

depth_positive_factor *value* // multiplier for positive depths

depth_negative_factor *value* // multiplier for negative depths

For example, if the opposite sign is required for depth, that is,

depth = height of the primary string - tin height value

simply set

depth_positive_factor -1

depth_negative_factor- 1

Continue to the next section 26.4.8.7.4 Boxes - Offset String Titles/Heights/Depths or return to 26.4.8 Boxes.

26.4.8.7.4 Boxes - Offset String Titles/Heights/Depths

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Offset string titles/heights/depths parameters

Each string in the offset model given can be automatically projected onto the primary string and drawn on the long section plot.

The offset model has been modified to allow strings to be projected onto the primary string to be specified by model and name, rather than just projecting all the strings in a given model. The string can also be labelled with either the string name, its model name or both.

The string is specified by

offset_n_mask *"model_name->string_name"*

For each n, a set of plot parameters determine if the string's heights and/or depths from the primary string are labelled in the boxes area.

If only the *string_name* is given, then the model given in the **Offset model** field of the **Section Long Plot** panel.

If there is **more than one string** with the given model and name, then it will be considered to be one string and at any primary string chainage, the closest of the strings will be the part used for projecting. For example, strings of the same name on either side of a road intersection will be considered to be the one string for projecting.

If any *offset_n_mask* is used, then the Offset option expects all the strings to be specified by an offset_n_mask. Then if no mask exists for any value of n, then that parameter set is ignored and no string projected.

If no offset masks are used (that is, no *offset_n_mask*'s are used), then the plot parameters determine whether each string in the offset model is drawn, and if the string's heights and depths from the primary string are labelled in the boxes area. The order that the strings are then plotted and labelled is the same as the order of the strings in the Offset model.

For the following sets of parameters, n takes the value 1 to 100 and specifies that the parameter set applies to the nth string given by *offset_n_mask* or, if no offset masks are given, the nth string in the *Offset model*.

Panel Field	Parameter name	Type	Pop-Up
Default offset model	**offset_model**	**model box**	
default offset model.			
Define set #		**input**	
where n = 1 to.... The set enables the specification of a number of parameters for a number of specified tin names.			
Offset mask	**offset_n_mask**	**input**	
the value specified by model_name->string_name.			
Offset draw mode	offset_n_draw_mode	choice box	Draw the offset string Do not draw the offset string
draw mode for offset string.			
Datum difference from primary		**input**	
Left search distance	**offset_n_lw**	**input**	
default 1000. Distance to search to the left of the primary string for the offset string. If 0, don't search to the left.			
Right search distance	**offset_n_rw**	**input**	

default 1000. Distance to search to the right of the primary string for the offset string. If 0, don't search to the right.

Offset colour	**offset_n_draw_colour**	**colour box**	

default offset string colour.

Offset label mode	offset_n_label	choice box	Label the strings heights
			Do not label the strings heights

offset string label mode.

See

26.4.8.7.5 Boxes - Offset String Titles/Heights/Depths - Titles
26.4.8.7.6 Boxes - Offset String Titles/Heights/Depths - Heights
26.4.8.7.7 Boxes - Offset String Titles/Heights/Depths - Depths

Or return to 26.4.8 Boxes.

26.4.8.7.5 Boxes - Offset String Titles/Heights/Depths - Titles

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Offset string - Title parameters

The default title for each set each parameter set, is the string name but this can be changed to just the model name or the model and string name:

Offset string mode **offset_title_mode** **choice box** **Label with string name**

Label with model name

Label with model->string

as

name

offset string title mode.

Use set # **input**

set number to be used to define sets of offset parameters i.e. n value

or, for any set, the *offset_title_mode* can be replaced by two lines of user defined title:

First line of offset title **offset_n_title** **input**

first line of nth offset title

Second line of offset title **offset_n_title_2** **input**

second line of nth offset title

Title textstyle **offset_n_title_textstyle** **text box**

nth offset title textstyle

Title colour **offset_n_title_colour** **colour box**

nth offset title colour

Title size (mm) **offset_n_title_size** **input**

nth offset title size

Title Y position (mm) **offset_n_title_y_pos** **input**

if set, the height in mm above the bottom of all the boxes that the offset title text is drawn. If not set, then the text is placed at a height that puts it inside the default box for the offset heights.

Continue to the next section 26.4.8.7.6 Boxes - Offset String Titles/Heights/Depths - Heights or return to 26.4.8 Boxes.

26.4.8.7.6 Boxes - Offset String Titles/Heights/Depths - Heights

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Offset string - Height parameters

Use set # **input**

set number to be used to define sets of offset parameters i.e. n value

Decimals **offset_n_decimals** **input**

number of dec places in nth offset height.

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Textstyle **offset_n_textstyle** **text box**

Textstyle of nth offset height

Colour **offset_n_colour** **colour box**

nth offset height and depth colour

Size (mm) **offset_n_size** **input**

nth offset height size

Y pos (mm) **offset_n_y_pos** **input**

if set, the height in mm above the bottom of all the boxes that the offset height text is drawn.If not set, then the text is placed at a height that puts it inside the default box for the offset heights.

Continue to the next section 26.4.8.7.7 Boxes - Offset String Titles/Heights/Depths - Depths or return to 26.4.8 Boxes.

26.4.8.7.7 Boxes - Offset String Titles/Heights/Depths - Depths

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Offset string - Depth parameters

Use set # **input**

set number to be used to define sets of offset parameters i.e. n value

Depth label mode **offset_n_depth_label** **choice box** **Label depths**

Do not label depths

depth label mode for the nth offset specified by set #.

First line of depth title **offset_n_depth_title** **input**

first line of nth offset depth title

Second line of depth title **offset_n_depth_title_2** **input**

second line of nth offset depth title

Depth title textstyle **offset_n_depth_title_textstyle** **text box**

nth offset depth title textstyle

Depth title colour **offset_n_depth_title_colour** **colour box**

nth offset depth title colour

Depth title size (mm) **offset_n_depth_title_sizeinput**

nth offset depth title size

Depth title Y position (mm) **offset_n_depth_title_y_pos** **input**

if set, the height in mm above the bottom of all the boxes that the offset depth title text is drawn. If not set, then the text is placed at a height that puts it inside the default box for the offset heights.

Decimals in depth value **offset_n_depth_decimalsinput**

number of dec places in offset height.

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Depth value textstyle **offset_n_depth_textstyle box**

nth offset depth textstyle

Depth value colour **offset_n_depth_colour** **colour box**

nth offset depth colour

Depth value size (mm) **offset_n_size** **input**

nth offset depth size

Depth value Y position (mm) **offset_n_depth_y_posinput**

if set, the height in mm above the bottom of all the boxes that the offset depth text is drawn. If not set, then the text is placed at a height that puts it inside the default box for the offset heights.

Continue to the next section 26.4.8.8 Boxes - Offset String Chainages or return to 26.4.8 Boxes.

26.4.8.8 Boxes - Offset String Chainages

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Offset string - Chainage parameters

Use String **chainage_n_offset_mode** **tick box**

if ticked, use offset string.

Offset String **chainage_n_offset_stringselect box**

the value specified by model_name->string_name.

First line of chainage title **chainage_n_title** **input**

first line of offset chainage title.

Second line of chainage title **chainage_n_title_2** **input**

second line of offset chainage title.

Chainage title textstyle **chainage_n_title_textstyle** **text box**

textstyle for the offset chainage title

Chainage title colour **chainage_n_title_colourcolour box**

colour of the offset chainage title

Chainage title size (mm) **chainage_n_title_size** **input**

size of the offset chainage title

Chainage title Y position (mm) chainage_n_title_y_pos **input**

if set, the height in mm above the bottom of all the boxes that the offset chainage title text is drawn. If not set, then the text is placed at a height that puts it inside the default box for the chainage.

Decimals in chainage value **chainage_n_decimals** **input**

number of decimal places for offset chainages.

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Chainage value textstyle **chainage_n_textstyle** **text box**

textstyle for the offset chainage values

Chainage value colour **chainage_n_colour** **colour box**

colour of the offset chainage values

Chainage value size (mm) **chainage_n_size** **input**

size of the offset chainage values

Chainage value Y position (mm) **chainage_n_y_pos** **input**

if set, the height in mm above the bottom of all the boxes that the offset chainage values are drawn. If not set, then the text is placed at a height that puts it inside the default box for the chainage.

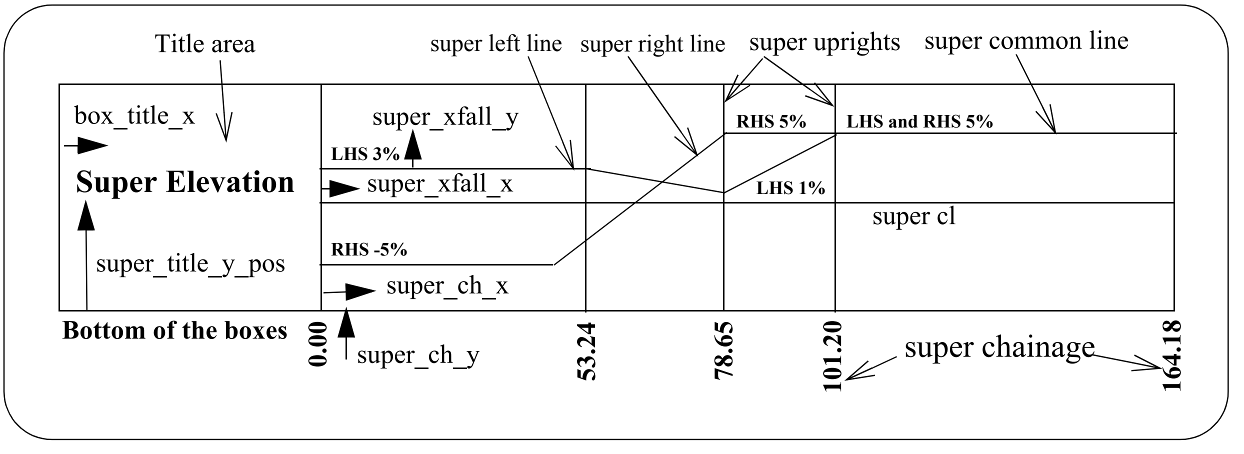
Continue to the next section 26.4.8.9 Boxes - Super Elevation Diagram or return to 26.4.8 Boxes.

26.4.8.9 Boxes - Super Elevation Diagram

The optional *super-elevation diagram* draws the values of the cross-fall (x-fall) between two strings using the primary string as the reference string. The cross-fall at a given chainage on the primary string is calculated by sectioning perpendicular to the reference string at that chainage and cutting the two strings. The cross-fall is defined as the cross-fall between the two cuts points on the strings.

The diagram has levels for the cross fall for a pair of strings on the left of the primary string, and a pair of strings on the right of the primary string. The diagram also has uprights in the super-elevation box with chainage values at the change of super values.

For the left hand side, the cross-fall is calculated at right angles to the primary string between the user given left hinge string and the left edge string.



The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Super elevation diagram parameters

Super elevation diagram draw mode

	super_draw_mode	**choice box**	**don't draw diagram**
		mode = 1	draw diagram using super sample interval
		mode = 2	sample super using chainages of x-sec's
		mode = 3	sample super using chainages from a string

super elevation diagram draw mode.

Super sample interval **super_sample_interval** **input**

if mode = 1, *Chainage distance to sample x-fall. (default 20)*

Super sample name **super_sample_name** **input**

name to be used in sampling. If mode = 2, name = model_name.

If mode = 3, name = model_name->string_name

Super tolerance **super_tolerance** **input**

If super_draw_mode is 1, the change of super is greater than super_tolerance, *draw uprights*

See

- 26.4.8.9.1 Boxes - Super Elevation Diagram - Titles
- 26.4.8.9.2 Boxes - Super Elevation Diagram - X-Fall
- 26.4.8.9.4 Boxes - Super Elevation Diagram - Left Side
- 26.4.8.9.5 Boxes - Super Elevation Diagram - Right Side
- 26.4.8.9.6 Boxes - Super Elevation Diagram - Centre Line
- 26.4.8.9.7 Boxes - Super Elevation Diagram - Common X-Fall
- 26.4.8.9.8 Boxes - Super Elevation Diagram - Uprights

Or return to 26.4.8 Boxes.

26.4.8.9.1 Boxes - Super Elevation Diagram - Titles

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Super elevation diagram - Title parameters

First line of super title **super_title** **input**

1st line of the super title

Super title textstyle **super_title_textstyle** **text box**

textstyle of super title

Super title colour **super_title_colour** **colour box**

colour of super title

Super title size (mm) **super_title_size** **input**

size of super title

Super title Y position (mm) **super_title_y_pos** **input**

if set, the height in mm above the bottom of all the boxes that the super title text is drawn. If not set, then the text is placed at a height that puts it inside the default box for the super.

Continue to the next section 26.4.8.9.2 Boxes - Super Elevation Diagram - X-Fall or return to 26.4.8 Boxes.

26.4.8.9.2 Boxes - Super Elevation Diagram - X-Fall

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Super elevation diagram - X-fall parameters

X-fall textstyle **super_xfall_textstyle** **text box**

textstyle of super title

X-fall colour **super_xfall_colour** **colour box**

colour of super title

X-fall size (mm) **super_xfall_size** **input**

size of super title

X-fall X position (mm) **super_xfall_x** **input**

the distance to the right from the super chainage uprights to the start of the x-fall text.

X-fall Y position (mm) **super_xfall_y** **input**

if set, the height in mm above the bottom of all the boxes that the super x-fall text is drawn. If not set, then the text is placed at a height that puts it inside the default box for the super.

Continue to the next section 26.4.8.9.3 Boxes - Super Elevation Diagram - Chainage or return to 26.4.8 Boxes.

26.4.8.9.3 Boxes - Super Elevation Diagram - Chainage

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Super elevation diagram - Chainage parameters

Chainage textstyle **super_ch_textstyle** **text box**

textstyle of super chainage title

Chainage colour **super_ch_colour** **colour box**

colour of super chainage title

Chainage size (mm) **super_ch_size** **input**

size of super chainage title

Chainage X position (mm) **super_ch_x** **input**

the distance to the right from the super chainage uprights to the start of the super chainage text.

Chainage Y position (mm) **super_ch_y** **input**

if set, the height in mm below the bottom of all the boxes that the super chainage is finished.

Chainage decimal places **super_ch_decimals** **input**

number of dec places in super chainage

Continue to the next section 26.4.8.9.4 Boxes - Super Elevation Diagram - Left Side or return to 26.4.8 Boxes.

26.4.8.9.4 Boxes - Super Elevation Diagram - Left Side

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Super elevation diagram - Left side parameters

Left hinge string **super_hinge_name_left** **select box**

left hinge string. (model_name->string_name)

Left edge string **super_edge_name_left** **select box**

left edge string. (model_name->string_name)

Super left linestyle **super_left_linestyle** **linestyle box**

linestyle of super left line.

Super left line colour **super_left_line_colour** **colour box**

colour of super left line.

Super left decimal places **super_left_decimals** **input**

number of dec places in super left x-fall value

Text before left X-fall value **super_left_pre** **input**

text before super left x-fall value.

Text after left X-fall value **super_left_post** **input**

text after super left x-fall value.

Continue to the next section 26.4.8.9.5 Boxes - Super Elevation Diagram - Right Side or return to 26.4.8 Boxes.

26.4.8.9.5 Boxes - Super Elevation Diagram - Right Side

The columns for the fields documented in the sections are for.

Panel Field Parameter name Type Pop-Up

Section: Super elevation diagram - Right side parameters

Right hinge string super_hinge_name_rightselect box

right hinge string. (model_name->string_name)

Right edge string super_edge_name_rightselect box

right edge string. (model_name->string_name)

Super right linestyle super_right_linestyle linestyle box

linestyle of super right line.

Super right line colour super_right_line_colourcolour box

colour of super right line.

Super right decimal places super_right_decimals input

number of dec places in super right x-fall value

Text before right X-fall value super_right_pre input

text before super right x-fall value.

Text after right X-fall value super_right_post input

text after super right x-fall value.

Continue to the next section 26.4.8.9.6 Boxes - Super Elevation Diagram - Centre Line or return to 26.4.8 Boxes.

26.4.8.9.6 Boxes - Super Elevation Diagram - Centre Line

The columns for the fields documented in the sections are for.

Panel Field Parameter name Type Pop-Up

Section: Super elevation diagram - Centre line parameters

Super centre linestyle super_cl_linestyle linestyle box

super centre line linestyle.

Super centre line colour super_cl_line_colour colour box

super centre line colour.

Continue to the next section 26.4.8.9.7 Boxes - Super Elevation Diagram - Common X-Fall or return to 26.4.8 Boxes.

26.4.8.9.7 Boxes - Super Elevation Diagram - Common X-Fall

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Super elevation diagram - Common X-fall parameters

Text before common X-fall value

super_common_pre **input**

text before super common x-fall value.

Text after common X-fall value

super_common_post **input**

text after super common x-fall value.

Super common linestyle **super_common_linestylelinestyle box**

super common linestyle.

Super common line colour

super_common_line_colour **colour box**

super common colour.

Super common decimal places

super_common_decimals **input**

number of decimal places in super common x-fall value

Continue to the next section 26.4.8.9.8 Boxes - Super Elevation Diagram - Uprights or return to 26.4.8 Boxes.

26.4.8.9.8 Boxes - Super Elevation Diagram - Uprights

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Super elevation diagram - Upright parameters

Upright mode super_upright_mode choice box for left string only

for right string only

for both strings

upright mode for super elevation diagram.

Super uprights colour **super_upright_colour** **colour box**

super upright colour.

Continue to the next section 26.4.8.10 Boxes - Volume Cut/Fill or return to 26.4.8 Boxes.

26.4.8.10 Boxes - Volume Cut/Fill

The columns for the fields documented in the sections are for.

Panel Field | **Parameter name** | **Type** | **Pop-Up**

Section: Volume cut/fill parameters

The values of calculated cut and fill volumes (cut and fill earth works) can be read from a file and then interpolated to produce cut and fill volumes for a given interval along the primary string.

Draw a volume diagram volume_draw_mode tick box

if ticked, draw a volume diagram.

Cut and fill text position volume_text_centre_mode choice box along the uprights
between the uprights

position of cut and fill text relative to the uprights.

Sample interval volume_sample_interval input

interval to display volumes.

Report file volume_file_name report box

name of volumes report file.

Box Y position (mm) volume_y_pos input

if set, the volume box is positioned at this height from the bottom of the first box

Box size (mm) volume_box_size input

if set, upright, volumes and titles box height.

Horizontal cut/fill dividing line linestyle

volume_cl_linestyle linestyle box

linestyle of the cut/fill dividing line.

Horizontal cut/fill dividing line colour

volume_cl_line_colour colour box

colour of the cut/fill dividing line.

See

26.4.8.10.1 Boxes - Volume Cut/Fill - Titles
26.4.8.10.2 Boxes - Volume Cut/Fill - Values
26.4.8.10.3 Boxes - Volume Cut/Fill - Uprights/Sub Uprights

Or return to 26.4.8 Boxes.

26.4.8.10.1 Boxes - Volume Cut/Fill - Titles

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Volumes cut/fill - Title parameters

First line of volume title **volume_title** **input**

1st line of the volume title

Volume title textstyle **volume_title_textstyle** **text box**

textstyle of volume title

Volume title colour **volume_title_colour** **colour box**

colour of volume title

Volume title size (mm) **volume_title_size** **input**

size of volume title

Volume title Y position (mm) **volume_title_y_pos** **input**

if set, the height in mm above the bottom of all the boxes that the volume title text is drawn. If not set, then the text is placed at a height that puts it inside the default box for the volumes.

See

26.4.8.10.1.1 Boxes - Volume Cut/Fill - Titles - Cut
26.4.8.10.1.2 Boxes - Volume Cut/Fill - Titles - Fill

Or return to 26.4.8 Boxes.

26.4.8.10.1.1 Boxes - Volume Cut/Fill - Titles - Cut

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Volumes cut/fill - Title cut parameters

First line of volume cut title volume_cut_title input

1st line of the cut volume title

Volume cut title textstyle volume_cut_title_textstyle text box

textstyle of cut volume title

Volume cut title colour volume_cut_title_colourcolour box

colour of cut volume title

Volume cut title size (mm) volume_cut_title_size input

size of cut volume title

Volume cut title Y position (mm) volume_cut_title_y input

if set, the height in mm above the default position. If not set, then the text is placed at a height that puts it inside the default box for the cut volumes.

Continue to the next section 26.4.8.10.1.2 Boxes - Volume Cut/Fill - Titles - Fill or return to 26.4.8 Boxes.

26.4.8.10.1.2 Boxes - Volume Cut/Fill - Titles - Fill

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Volumes cut/fill - Title fill parameters

First line of volume fill title volume_fill_title input

1st line of the fill volume title

Volume fill title textstyle volume_fill_title_textstyle text box

textstyle of fill volume title

Volume fill title colour volume_fill_title_colour colour box

colour of fill volume title

Volume fill title size (mm) volume_fill_title_size input

size of fill volume title

Volume fill title Y position (mm) volume_fill_title_yinput

if set, the height in mm above the default position. If not set, then the text is placed at a height that puts it inside the default box for the fill volumes.

Continue to the next section 26.4.8.10.2 Boxes - Volume Cut/Fill - Values or return to 26.4.8 Boxes.

26.4.8.10.2 Boxes - Volume Cut/Fill - Values

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Volumes cut/fill - Cut value parameters

Textstyle **volume_cut_textstyle** **text box**

textstyle of the cut values.

Colour **volume_cut_text_colourcolour box**

colour of cut values

Size (mm) **volume_cut_text_size** **input**

size of cut values

X position (mm) **volume_cut_text_x** **input**

the x distance to move the cut text from the default cut text position, def 0, it is not used if volume_text_centre_mode= between the uprights

Y position (mm) **volume_cut_text_y** **input**

the y distance to move the cut text from the default cut text position

Decimal places **volume_cut_decimals** **input**

number of dec places in cut values.

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Section: Volumes cut/fill - Fill value parameters

Textstyle **volume_fill_textstyle** **textstyle box**

textstyle of the fill values.

Colour **volume_fill_text_colour** **colour box**

colour of fill values

Size (mm) **volume_fill_text_size** **input**

size of fill values

X position (mm) **volume_fill_text_x** **input**

the x distance to move the fill text from the default fill text position, def 0, it is not used if volume_text_centre_mode= between the uprights

Y position (mm) **volume_fill_text_y** **input**

the y distance to move the fill text from the default fill text position

Decimal places **volume_fill_decimals** **input**

number of dec places in fill values.

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Continue to the next section 26.4.8.10.3 Boxes - Volume Cut/Fill - Uprights/Sub Uprights or return to 26.4.8 Boxes.

26.4.8.10.3 Boxes - Volume Cut/Fill - Uprights/Sub Uprights

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Volumes cut/fill - Upright parameters

Draw cut/fill uprights volume_uprights_draw_mode

choice box **don't draw cut/fill uprights**

draw cut/fill uprights

upright draw mode.

Colour of uprights **volume_uprights_line_colour** **colour box**

colour of uprights

Section: Volumes cut/fill - Sub upright parameters

Colour of sub uprights **volume_sub_upright_colourcolour box**

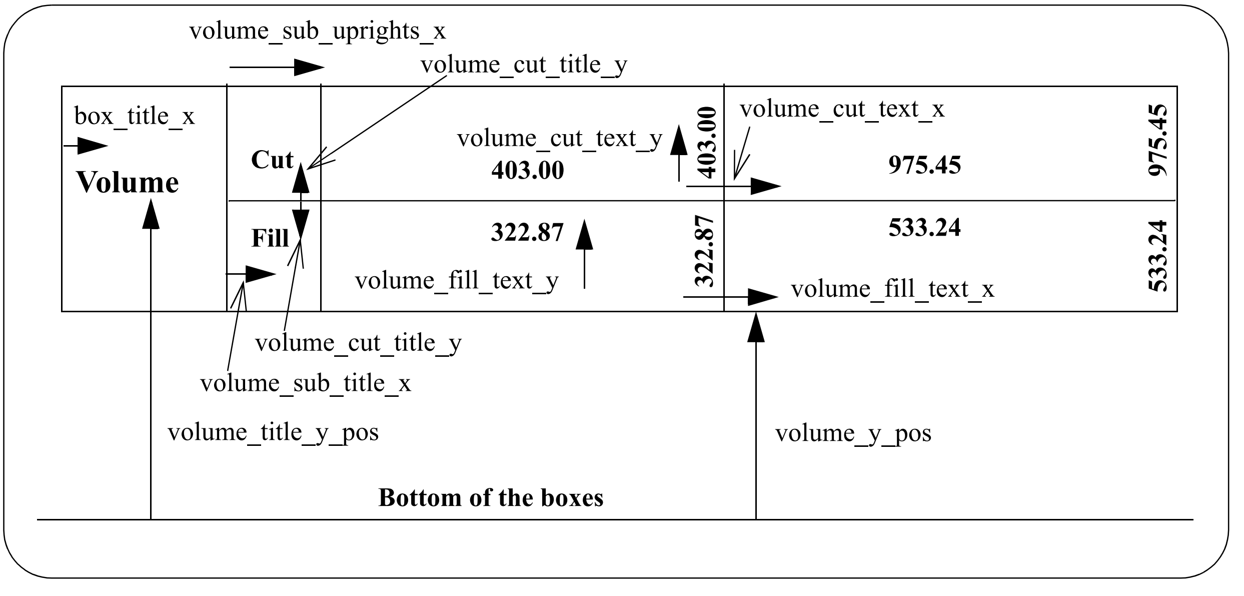
colour of sub uprights line

Sub upright X position (mm) **volume_sub_uprights_x** **input**

the x distance to move from the default sub uprights position.

Sub upright title X position (mm) **volume_sub_title_x** **input**

the distance to move the sub title text from the volume sub uprights



Continue to the next section 26.4.8.11 Boxes - X,Y or return to 26.4.8 Boxes.

26.4.8.11 Boxes - X,Y

The values of the X and Y coordinate for selected chainages can be labelled as separate boxes.

The chainages specified are limited to the base set of chainages already defined in the chainage selection and staggering section. i.e. an upright must exist for the X and Y value to be labelled. Not all of the uprights have to be labelled, just those specified in this section.

There are plot parameters to control all aspects of the X and Y labelling.

Chainages are used for positioning X and Y labels.

The chainages for the long section plot relate to the primary string and are controlled by a set of parameters.

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: X, Y parameters

X, Y order xy_order choice box X before Y / Y before X

*order to place X and Y boxes. X before Y indicates that the X box will appear closest to the bottom of the boxes and the Y box will be on top of it. **Note**: The boxes are placed from bottom to top.*

Chord/arc chainage mode xy_chord_arc choice box Do not use chord/arc chainages / Use chord/arc chainages

mode to include chord/arc chainages.

Start chainage **xy_start_chainage** **input box**

start chainage of chainage range to be labelled.

End chainage **xy_end_chainage** **input box**

end chainage of chainage range to be labelled.

Chainage interval **xy_interval** **input**

the regular interval (0=no regulars).

Include tangents, spirals **xy_label_hcp** **tick box**

if ticked, include the tangents and spirals.

Include hip points **xy_label_hip** **tick box**

if ticked, include horizontal intersection points.

Include vip points **xy_label_vip** **tick box**

if ticked, include vertical intersection points.

Include tangent points **xy_label_vtp** **tick box**

if ticked, include tangent points.

Include crest points **xy_label_crest** **tick box**

if ticked, include crest points.

Include sag points **xy_label_sag** **tick box**

if ticked, include sag points.

Include change of vertical grade

xy_label_grade_change **tick box**

if ticked, include changes in vertical grade.

Include chord/arc tolerance chainages

xy_chord_arc **tick box**

Chainage weeding tolerance

xy_label_tolerance **input**

if >0 then use as a weeding tolerance, if <=0 don't weed.

Files of special chainages

xy_special_n_file **file box**

n = 1 to 20 - include chainages from the file (one chainage per line)

See

26.4.8.11.1 Boxes - X,Y - X Parameters
26.4.8.11.2 Boxes - X,Y - Y Parameters

Or return to 26.4.8 Boxes.

26.4.8.11.1 Boxes - X,Y - X Parameters

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: X parameters

Label X values/title **x_label** **tick box**

if ticked, the X values/titles will be labelled.

Section: X title parameters

Title line 1 **x_title** **input**

1st line of X title.

Title line 2 **x_title_2** **input**

second line of X title.

Textstyle **x_title_textstyle** **text box**

textstyle for the X title

Colour **x_title_colour** **colour box**

colour of the X title

Text size (mm) **x_title_size** **input**

size of the X title

Text y position (mm) x_title_y_pos input

if set, the height in mm above the bottom of all the boxes that the X title text is drawn. If not set, then the text is placed at a height that puts it inside the default box for the X labels.

Section: X value parameters

Decimal places **x_label_decimals** **input**

number of decimal places for X values.

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Textstyle **x_label_textstyle** **text box**

textstyle for the X values

Colour **x_label_colour** **colour box**

colour of the X values

Text size (mm) **x_label_size** **input**

size of the X values

Text y position (mm) **x_label_y_pos** **input**

if set, the height in mm above the bottom of all the boxes that the X values are drawn. If not set, then the text is placed at a height that puts it inside the default box for the X labels.

Continue to the next section 26.4.8.11.2 Boxes - X,Y - Y Parameters or return to 26.4.8 Boxes.

26.4.8.11.2 Boxes - X,Y - Y Parameters

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Y parameters

Label Y values/titles **y_label** **tick box**

if ticked, the Y values/titles will be labelled.

Section: Y Title Parameters

Title line 1 **y_title** **input**

1st line of Y title.

Title line 2 **y_title_2** **input**

second line of Y title.

Textstyle **y_title_textstyle** **text box**

textstyle for the Y title

Colour **y_title_colour** **colour box**

colour of the Y title

Text size (mm) **y_title_size** **input**

size of the Y title

Text y position (mm) y_title_y_pos input

if set, the height in mm above the bottom of all the boxes that the Y title text is drawn. If not set, then the text is placed at a height that puts it inside the default box for the Y labels.

Section: Y value parameters

Decimal places **y_label_decimals** **input**

number of decimal places for Y values.

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Textstyle **y_label_textstyle** **text box**

textstyle for the Y values

Colour **y_label_colour** **colour box**

colour of the Y values

Text size (mm) **y_label_size** **input**

size of the Y values

Text y position (mm) **y_label_y_pos** **input**

if set, the height in mm above the bottom of all the boxes that the Y values are drawn. If not set, then the text is placed at a height that puts it inside the default box for the Y labels.

Please continue to the next section 26.4.9 Chainage/Staggering.

26.4.9 Chainage/Staggering

Chainages are used for positioning height labels, uprights (leader lines) and bubbles. The chainages for the long section plot relate to the primary string

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Chainage/Staggering - Chainage parameters

Chord/arc chainage mode chord_arc choice box Do not use chord/arc chainages
Use chord/arc chainages

mode to include chord/arc chainages.

Chainage interval **chainage_interval** **input**

the regular interval (0=no regulars).

Include start and end chainages
chainage_label_ends **tick box**

if ticked, include the start and end chainages.

Include tangents, spirals **chainage_label_hcp** **tick box**

if ticked, include the tangents and spirals.

Include horizontal intersection points **chainage_label_hip** **tick box**

if ticked, include horizontal intersection points.

Include horizontal tangent points **chainage_label_hcp** **tick box**

if ticked, include horizontal tangent points.

Include vertical intersection points **chainage_label_vip** **tick box**

if ticked, include vertical intersection points.

Include vertical tangent points **chainage_label_vtp** **tick box**

if ticked, include vertical tangent points.

Include crest points **chainage_label_crest** **tick box**

if ticked, include crest points.

Include sag points **chainage_label_sag** **tick box**

if ticked, include sag points.

Include change of vertical grade **chainage_label_grade_change** **tick box**

if ticked, include changes in vertical grade.

Include chord/arc tolerance chainages **chainage_label_grade_change** **tick box**

Chainage weeding tolerance **chainage_label_tolerance** **input**

if >0 then use as a weeding tolerance, if <=0 don't weed.

Merge in bubble chainages **chainage_merge_bubbles** **tick box**

if ticked, merge in bubble chainages.

Weeding tolerance after bubble merge **chainage_merge_tolerance** **input**

if > 0, weed after merge using specified weeding tolerance. If <= 0, don't weed after merge.

Files of special chainages **chainage_special_n_file** **file box**

n = 1 to 20 - include chainages from the file (one chainage per line)

Continue to the next section 26.4.9.1 Chainage/Staggering - Staggering or return to 26.4.9

Chainage/Staggering.

26.4.9.1 Chainage/Staggering - Staggering

If the real chainage position is used for the horizontal position of the chainage/height/depth text, text over writing can easily occur.

To prevent over writing, the text can be **staggered**.

If the ***stagger_mode*** parameter is set to 1, the text position is adjusted so that the text does not over write.

The real chainage position is then indicated by the chainage markers which are drawn at the top of the text boxes from the staggered text position back to the actual chainage position of the text.

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Chainage/Staggering - Staggering parameters

Stagger mode	stagger_mode	choice box	no staggering, allow over writing Stagger text

stagger mode.

Distance from boxes to top of staggers (mm)

	stagger_gap_top	input	

distance from the top of boxes to top of staggers in mm.

Distance from boxes to bottom of staggers (mm)

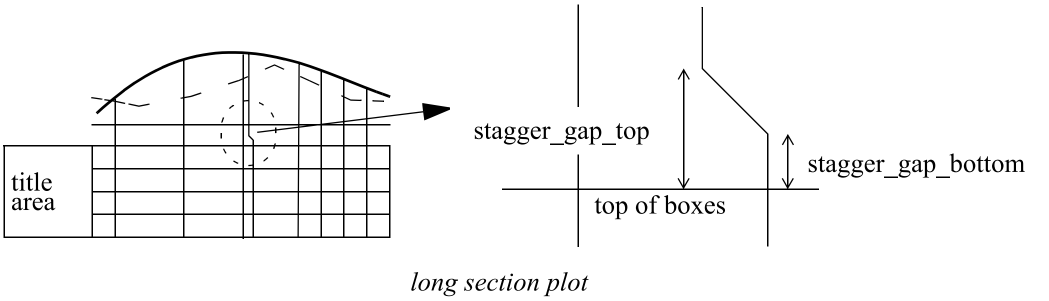
	stagger_gap_bottom	input	

distance from the top of boxes to bottom of staggers in mm.

Stagger gap factor	stagger_gap_factor	input	

*distance between staggers is box_text_size * stagger_gap_factor*

When staggering occurs, it is possible for the heights area to be longer than the graph area.



long section plot

Please continue to the next section 26.4.10 Uprights.

26.4.10 Uprights

Uprights, or leader lines, can be drawn from the top of the staggers to the strings drawn on the plot.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Upright parameters

Upright draw mode uprights_draw_mode choice box *none*
max string height
to stagger height
uprights_y above boxes
to primary string
tin 1
tin 2
tin 3
tin 4
tin 5
tin 6
tin 7
tin 8
tin 9
tin 10
to offset 1
to offset 2
to offset 3
to offset 4
to offset 5
to offset 6
to offset 7
to offset 8
to offset 9
to offset 10

upright draw mode.

Uprights Y distance (mm) **uprights_y** **input**

distance to draw the uprights for uprights_draw_mode = "to uprights_y above boxes"

Uprights colour **uprights_colour** **colour box**

uprights colour. Default is box_colour

Uprights bottom mode **uprights_bottom_mode** **choice box** **stop at top of boxes**
draw to bottom of boxes
draw to uprights_bottom_y below top of boxes
draw to uprights_bottom_y above bottom of boxes
ticks at chainage

Upright draw mode below top of boxes.

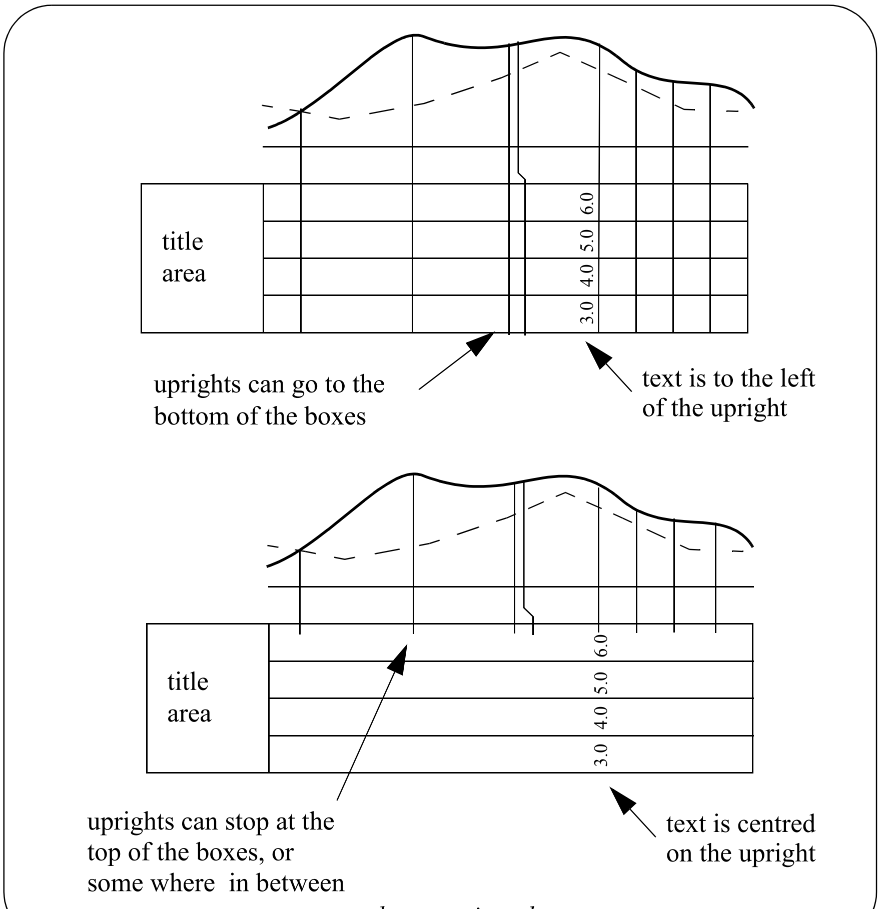
Uprights bottom Y distance (mm) **uprights_bottom_y** **input**

distance in mm.

Uprights text offset factor **uprights_text_offset_factor** **input**

move the text by this factor*size.

When uprights go below the top of the boxes, the height and offset text is moved to the left so that the upright does not go through the text. The left hand side of the heights boxes also moves to the left to leave room for the height text.



long section plot

Please continue to the next section 26.4.11 Datum Area.

26.4.11 Datum Area

The **datum area** is the region between the boxes area and the graph area.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Datum area - datum value mode

Use manual datum **manual_datum** **tick box**

if ticked the datum will be calculated internally, else use the datum_value parameter for the datum.

Datum value **datum_value** **input**

the value to be used for the datum.

Datum roundoff **datum_roundoff** **input**

the value to roundoff the datum value.

Decimal places for datum **datum_decimals** **input**

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Section: Datum area - other parameters

Datum name **datum_name** **input**

text to write before the datum value

Graph area to datum line gap (mm) datum_above_gap real box

distance from the bottom of the graph area to the datum line.

Datum line gap to top of boxes (mm) datum_below_gap real box

distance from datum line to top of boxes.

The **datum line** is positioned the distance ***datum_below_gap*** above the top of the boxes area and the graph area is positioned the distance ***datum_above_gap*** above the datum line.

Hence the graph area is distance (datum_below_gap + datum_above_gap) above the top of the boxes area.

The *datum_below_gap* and *datum_above_gap* can be zero or positive.

Datum linestyle datum_linestyle linestyle box

datum line linestyle (default solid)

Datum textstyle datum_textstyle text box

textstyle for datum information

Datum text size (mm) datum_text_size input

size of datum text and value (mm)

Datum colour datum_colour colour box

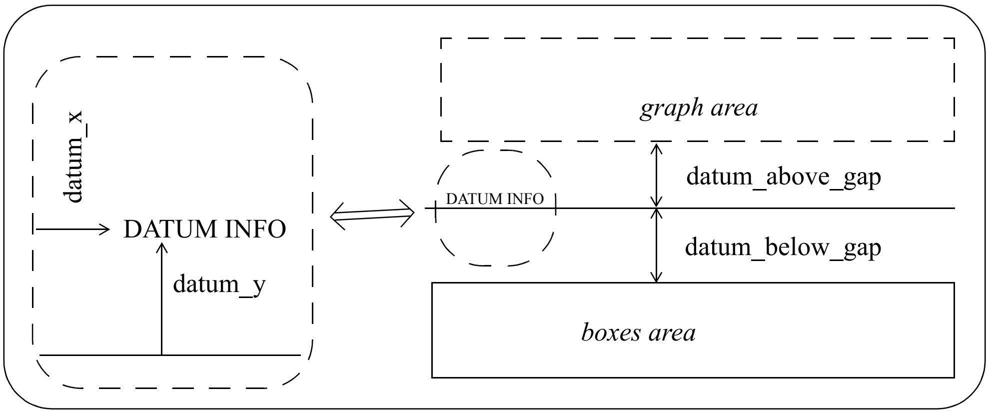
colour of the datum text and line.

X adjustment (mm) **datum_x** **input**

distance to move the datum text along the datum line

Y adjustment (mm) **datum_y** **input**

distance to raise the datum text above the datum line



Continue to the next section 26.4.11.1 Datum Area - Offset String Datum Labels or return to 26.4.11 Datum Area.

26.4.11.1 Datum Area - Offset String Datum Labels

Section: Offset string datum area label parameters

Offset string

Datum name

Datum colour

X adjustment (mm)

Y adjustment (mm)

Please continue to the next section 26.4.12 Graph Area.

26.4.12 Graph Area

The graph area sits on top of the boxes and datum areas, so there may not be enough room left on the sheet for the full plot height. In this case, the plot will be truncated at the top of the allowed graph area.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Graph area parameters

Left/Right extensions (world units)

if not zero, the long section goes for this chainage distance from before the primary string starts to after the primary string ends. The primary string is extended by continuing at the instantaneous bearing at the ends of the primary string.

Extra space units **extra_space_units** **choice box** **world units**

millimetres

units for specifying extra space to be added to the plot area.

Extra space left (units) **extra_space_left** **input**

extra distance to add to the end of title area before the plot area starts.

Extra space right (units) **extra_space_right** **input**

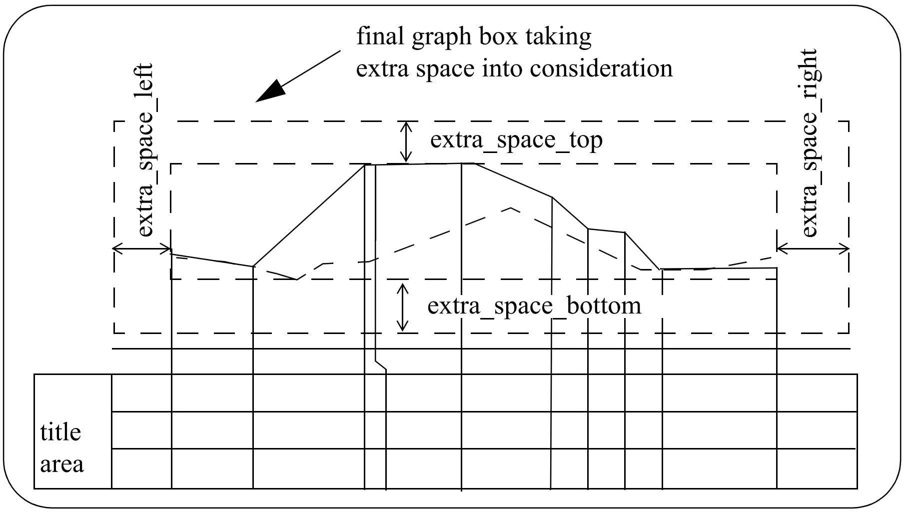
extra distance to add to the end of plot area for the end of the boxes to draw to.

Extra space top (units) **extra_space_top** **input**

extra distance to add to the top of plot area.

Extra space bottom (units) **extra_space_bottom** **input**

extra distance to add before the bottom of plot area starts.



Please continue to the next section 26.4.13 Corridors - Long Section.

26.4.13 Corridors - Long Section

A corridor around the primary string is defined by giving a left and right corridor width.

Any string in a model added to the section view is checked to see if it appears in the corridor, and if it does, it is drawn on the cross-section plot.

To be drawn, strings do not have to cross the primary string, but just be in the corridor.

This is documented for all the PPF Editors in 26.2.7 Corridors.

Please continue to the next section 26.4.14 Bubbles.

26.4.14 Bubbles

Circles with the string name and a unique number (**bubbles**) can be drawn on the long section plot. Bubbles are normally used for lip profiles.

The chainages used for the bubbles are given by a set of parameters similar to the chainage parameters. The resulting set of bubbles are sequentially numbered (starting with one) in chainage order.

Although many bubbles can be defined by the bubble parameters, a bubble is only drawn on the plot if there is a labelled chainage to draw it above.

Hence not all bubbles given by the bubble chainage parameters are drawn but for the ones that are drawn, the bubble number is taken from the full bubble set.

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Bubble definition parameters

Bubble draw mode	chainage_bubbles	choice box	Do not draw bubbles Draw bubbles
draw bubble mode			
Bubble radius	bubble_radius	input	
radius of the bubbles			
Bubble colour	bubble_colour	colour box	
colour of the bubbles			
Bubble upright mode	bubble_draw_upright	choice box	Do not draw bubble upright Draw bubble upright
draw bubble upright mode			
Bubble start chainage	bubble_start_chainage	input	
start chainage bubbles			
Bubble end chainage	bubble_end_chainage	input	
end chainage bubbles			
Chord/arc chainage mode	bubble_chord_arc	choice box	Do not use chord/arc chainages Use chord/arc chainages
chord/arc chainage mode			
Bubble interval	bubble_interval	input	
include regular interval for bubbles			

See

26.4.14.1 Bubbles - Text
26.4.14.2 Bubbles - Label

Or return to 26.4 Long Plot PPF Editor.

26.4.14.1 Bubbles - Text

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Bubble definition - Text parameters

Bubble text string name mode

bubble_text_string_name_mode

choice box Do not label with string name
Label with string name
Label with model->string name

mode of bubble text string name.

Pre text **bubble_pre_text** **input**

pre text for label

Post text **bubble_post_text** **input**

post text for label

Textstyle **bubble_textstyle** **text box**

textstyle of label of label

Size (mm) **bubble_text_size** **input**

size of label

Colour **bubble_text_colour** **colour box**

colour of label

Offset (mm) **bubble_text_offset** **input**

offset value for bubbles.

Upright distance (mm) **bubble_upright_distanceangle box**

distance bubbles are above boxes/uprights

Bubble upright distance mode **bubble_mode** **choice box** **Distance is above boxes**
Distance is above uprights

mode of bubble upright distance.

Continue to the next section 26.4.14.2 Bubbles - Label or return to 26.4.14 Bubbles.

26.4.14.2 Bubbles - Label

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Bubble definition - Label parameters

Include tangents, spirals **bubble_label_hcp** **tick box**

if ticked, include the tangents and spirals.

Include horizontal intersection points **bubble_label_hip** **tick box**

if ticked, include horizontal intersection points.

Include horizontal tangent points **bubble_label_hcp** **tick box**

if ticked, include horizontal tangent points.

Include vertical intersection points **bubble_label_vip** **tick box**

if ticked, include vertical intersection points.

Include vertical tangent points **bubble_label_vtp** **tick box**

if ticked, include vertical tangent points.

Include crest points **bubble_label_crest** **tick box**

if ticked, include crest points.

Include sag points **bubble_label_sag** **tick box**

if ticked, include sag points.

Include change of vertical grade

bubble_label_grade_change **tick box**

if ticked, include changes in vertical grade.

Include chord/arc tolerance chainages

bubble_chord_arc **tick box**

Bubble weeding tolerance

bubble_label_tolerance **input**

if >0 then use as a weeding tolerance, if <=0 don't weed.

Files of special chainages

bubble_special_n_file **file box**

n = 1 to 20 - include chainages from the file (one chainage per line)

Please continue to the next section 26.4.15 Quick Horizontal Geometry.

26.4.15 Quick Horizontal Geometry

The standard horizontal geometry arrows can be drawn at a given distance above the top of the boxes area.

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Quick horizontal geometry

Distance above boxes (mm) horizontal_geometry_yinput

distance above boxes for drawing of arrows

Section: Quick horizontal geometry - Left side label parameters

Text horizontal_geometry_label_text input

text for label

Textstyle horizontal_geometry_label_textstyle text box

textstyle of label

Text colour horizontal_geometry_label_text_colour colour box

colour of label

Text size (mm) horizontal_geometry_label_text_size input

size of label

Section: Quick horizontal geometry - Arrow and text parameters

Text colour horizontal_geometry_arrow_text_colour colour box

colour of arrow text

Textstyle horizontal_geometry_arrow_textstyle text box

textstyle of arrow text

Text size (mm) horizontal_geometry_arrow_text_size input

size of arrow text

Decimal places horizontal_geometry_label_decimals input

number of decimal places in arrow text.

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Arrow colour horizontal_geometry_arrow_colour colour box

colour of arrow

Arrow height (mm) horizontal_geometry_arrow_height input

size of arrow

Please continue to the next section 26.4.16 Extensive Horizontal Geometry.

26.4.16 Extensive Horizontal Geometry

The standard horizontal geometry arrows can be drawn at a given distance above the top of the boxes area.

For complicated horizontal geometry labelling, there are sets of horizontal geometry labelling parameters which give tight control over the position and types of labels.

It is also possible to label the horizontal geometry of ***alignment*** strings other than the primary string. To plot such a string on the same plot, the chainage position of the horizontal geometry for the non-primary alignment strings is *projected* onto the primary string to give a primary string chainage for plotting. The values of the horizontal geometry (such as radius and spiral length) that are plotted are taken from the other string. Independently graded offset strings (such as a left and right kerbs) are the type of additional alignment strings that may need to be plotted on the same long section plot as the reference string (primary string).

For plotting horizontal geometry, the user can give up to twenty sets of these labels and they can be used to label spirals, curves and tangent information for the primary string and/or additional alignment strings.

Each label set consists of three parts:

(a) a text label on the left hand side of the plot
(b) an arrow
(c) text on the arrows.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Extensive horizontal geometry parameters

Define set # **input**

where n = 1 to....20. The set enables the specification of a number of parameters for a number of specified extensive horizontal geometry.

For the following parameters, n takes the value 1 to 20 and specifies the nth parameter set.

Panel Field	Parameter name	Type	Pop-Up
Geometry to label	h_g_n_type	choice box	label spirals
			label horizontal curves
			label horizontal tangents

specifies what geometry is to be labelled for the nominated set. If h_g_n_type is missing, then the set is ignored.

Panel Field	Parameter name	Type	Pop-Up
Value to label	h_g_n_value_mode	choice box	nothing
			length
			radius (for curve labelling)
			or radius*length (for spiral)

specifies the value to label.

Y offset (mm) **h_g_n_label_y** **input**

distance of arrow line above top of the boxes.

If the set of parameters is to apply to the horizontal geometry of an alignment string *other* than the primary string, then the offset string can be specified.

Offset string **h_g_n_offset_string** **select box**

the name of the non-primary string. i.e.

h_g_n_offset_string *model->string_name*

or

h_g_n_offset_string *string_name*

and the model is the *defined by offset_model*. *offset_model has been* defined in the section 26.4.8.7.4 Boxes - Offset String Titles/Heights/Depths.

If the *h_g_n_offset_string* parameter does not exist, then the set of horizontal geometry parameters is applied to the primary string.

See

26.4.16.1 Extensive Horizontal Geometry - Left Hand Labels
26.4.16.2 Extensive Horizontal Geometry - Arrow Type
26.4.16.3 Extensive Horizontal Geometry - Arrow Text

Or return to 26.4 Long Plot PPF Editor.

26.4.16.1 Extensive Horizontal Geometry - Left Hand Labels

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Extensive horizontal geometry - Left hand label parameters

The fields and buttons used in this section have the following functions.

Field Description **Parameter name** **Type** **Pop-Up**

Use set # **input**

set #, as specified by the **define set #** *parameter.*

For the following parameters, n takes the value 1 to 20 and specifies the nth parameter set.

Label X (mm) **h_g_n_label_x** **input**

distance from the left hand side of the labels area to start the left hand label text.

Offset (mm) **h_g_n_label_offset** **input**

distance to raise the left hand label text above arrow line.

Text size (mm) **h_g_n_label_text_size** **input**

size of the left hand label text.

Text colour **h_g_n_label_text_colour** **colour box**

size of the left hand label text.

Text **h_g_n_label_text** **input**

left hand label text.

Textstyle **h_g_n_label_textstyle** **text box**

textstyle for the left hand label text.

Continue to the next section 26.4.16.2 Extensive Horizontal Geometry - Arrow Type or return to 26.4.16 Extensive Horizontal Geometry.

26.4.16.2 Extensive Horizontal Geometry - Arrow Type

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Extensive horizontal geometry- Arrow type parameters

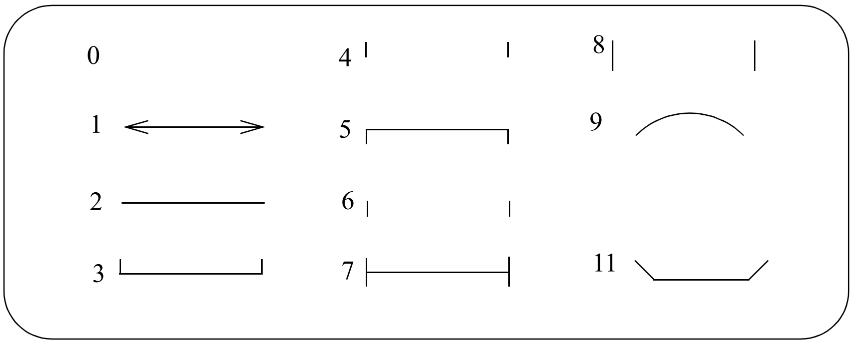
Use set # **input**

set #, as specified by the **define set #** *parameter.*

For the following parameters, n takes the value 1 to 20 and specifies the nth parameter set.

Arrow type mode h_g_n_draw_mode choice box arrow (1)
line (2)
line with uprights at ends (3)
uprights with no lines (4)
line with downrights (5)
downrights with no lines (6)
line with up and downrights at ends (7)
up and downrights with no line (8)
draw curve (9)
radius*length curve (11)

specifies the arrow type to be drawn.



Left arrow gap (mm) **h_g_n_left_gap** **input**

size of gap for left side of arrow.

Right arrow gap (mm) **h_g_n_right_gap** **input**

size of gap for right side of arrow.

Arrow colour **h_g_n_colour** **colour box**

colour of arrow text

Arrow height (mm) **h_g_n_height** **input**

height of arrow in mm.

Leave gap in arrow for text **h_g_n_gap** **tick box**

if ticked, a gap will be left for text.

Continue to the next section 26.4.16.3 Extensive Horizontal Geometry - Arrow Text or return to 26.4.16 Extensive Horizontal Geometry.

26.4.16.3 Extensive Horizontal Geometry - Arrow Text

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Extensive horizontal geometry - Arrow text parameters

Use set # **input**

set #, as specified by the **define set #** *parameter.*

Text colour **h_g_n_text_colour** **colour box**

colour of arrow text

Text size (mm) **h_g_n_text_size** **input**

size of arrow text

Text offset (mm) **h_g_n_text_offset** **input**

distance to raise the text above the arrow line.

Arrow text pre-text **h_g_n_pre_text** **input**

text before the arrow text

Arrow text post-text **h_g_n_post_text** **input**

text after the arrow text

Textstyle **h_g_n_textstyle** **text box**

textstyle of arrow text

Decimal places **h_g_n_no_decimals** **input**

number of decimal places in arrow text.

If > 0, trailing zeros are ***removed*** *after the decimal point.*
If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Rotate text to fit **h_g_n_rotate** **tick box**

if ticked, the text on the arrows will be rotated to fit.

Please continue to the next section 26.4.17 Quick Vertical Geometry.

26.4.17 Quick Vertical Geometry

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Quick vertical geometry - Grade

Distance above boxes (mm)

vertical_geometry_grade_y input

distance above boxes. If 0, don't draw.

Section: Quick vertical geometry - Length

Distance above boxes (mm)

vertical_geometry_length_y input

distance above boxes. If 0, don't draw.

Section: Quick vertical geometry - Common parameters

Arrow mode vertical_geometry_arrow_mode

input **ticks**
arrows

arrow mode.

Arrow colour **vertical_geometry_arrow_colour**

colour box

arrow colour.

Arrow height (mm) **vertical_geometry_arrow_height**

input

height of arrow in mm.

See

26.4.17.1 Quick Vertical Geometry - Grade Labels
26.4.17.2 Quick Vertical Geometry - Length Labels

Or return to 26.4 Long Plot PPF Editor.

26.4.17.1 Quick Vertical Geometry - Grade Labels

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Quick vertical geometry - Grade left hand label parameters

Text **vertical_geometry_label_grade_text**
input

text for left hand grade label.

Textstyle **vertical_geometry_label_grade_textstyle**
textstyle box

textstyle for left hand grade label.

Text colour **vertical_geometry_label_grade_text_colour**
colour box

colour of left hand grade label text

Text size (mm) **vertical_geometry_label_grade_text_size**
input

size of left hand grade label text

Section: Quick vertical geometry - Grade arrow and text parameters

Grade mode **vertical_geometry_grade_mode**
choice box **%**
1 in

mode of grade for grade arrow text.

Grade decimal places **vertical_geometry_label_grade_decimals**
input

number of decimal places for grade arrow text.

Grade arrow textstyle **vertical_geometry_arrow_grade_textstyle**
textstyle box

textstyle for grade arrow text.

Grade arrow text colour **vertical_geometry_arrow_grade_text_colour**
colour box

colour of grade arrow text.

Grade arrow text size (mm)
vertical_geometry_arrow_grade_text_size
input

size of grade arrow text

Continue to the next section 26.4.17.2 Quick Vertical Geometry - Length Labels or return to 26.4.17 Quick Vertical Geometry.

26.4.17.2 Quick Vertical Geometry - Length Labels

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Quick vertical geometry - Length left hand label parameters

Text **vertical_geometry_label_length_text**

input

text for left hand length label.

Textstyle **vertical_geometry_label_length_textstyle**

textstyle box

textstyle for left hand length label.

Text colour **vertical_geometry_label_length_text_colour**

colour box

colour of left hand length label

Text size (mm) **vertical_geometry_label_length_text_size**

input

size of left hand length label

Section: Quick vertical geometry - Length arrow and text parameters

Length mode **vertical_geometry_length_mode**

choice box **length**
radius
k value
mixed - length for parabolic,
radius for circular

mode of length for length labels.

Length decimal places **vertical_geometry_label_length_decimals**

input

number of decimal places for length arrow labels.

Length arrow textstyle **vertical_geometry_arrow_length_textstyle**

textstyle box

textstyle for length arrow labels.

Length arrow text colour **vertical_geometry_arrow_length_text_colour**

colour box

colour of length arrow labels.

Length arrow text size (mm)

vertical_geometry_arrow_length_text_size

input

size of length arrow labels

Please continue to the next section 26.4.18 Extensive Vertical Geometry.

26.4.18 Extensive Vertical Geometry

For complicated vertical geometry labelling of the *primary alignment* string, there are sets of vertical geometry labelling parameters which give tight control over the position and types of labels.

It is also possible to label the vertical geometry of ***alignment*** strings other than the primary string. To plot such a string on the same plot, the chainage position of the vertical geometry for the non-primary alignment strings is *projected* onto the primary string to give a primary string chainage for plotting. The values of the vertical geometry (such as grade and curve length) that are plotted are taken from the other string. Independently graded offset strings (such as a left and right kerbs) are the type of additional alignment strings that may need to be plotted on the same long section plot as the reference string (primary string).

For plotting vertical geometry, the user can give up to twenty sets of these labels and they can be used to label grades or vertical curve information for the primary string and/or additional alignment strings.

Each label set consists of three parts:

(a) a text label on the left hand side of the plot
(b) an arrow
(c) text on the arrows.

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Extensive vertical geometry parameters

Define set # **input**

where n = 1 to....20. The set enables the specification of a number of parameters for a number of specified extensive horizontal geometry.

For the following parameters, n takes the value 1 to 20 and specifies the nth parameter set.

Geometry to label	v_g_n_type	choice box	label grades **label vg curve information**

specifies what geometry is to be labelled for the nominated set. If v_g_n_type *is missing, then the set is ignored.*

Value to label	v_g_n_value_mode	choice box	nothing %grade |or| parabola length, arc length *1 in grade |or| radius* *mm grade |or| K value* *ch length between curve points* *per chord(QR) |or| curve* constant (QR).

specifies the value to label. This will be dependant on the v_g_n_type chosen.

Vertical curve points to draw the arrows between (for grade labelling only)

Points to draw arrows between

	v_g_n_between_mode	**choice box**	**between chainages at the** vips between chainages at the vtps

specifies where to draw the arrows between (for grade labelling only).

Y offset (mm) **v_g_n_label_y** **input**

distance of arrow line above top of the boxes. This value can be negative

If the set of parameters is to apply to the vertical geometry of an alignment string *other* than the primary string, then the following parameter can define the other alignment string

Offset string **v_g_n_offset_string** **select box**

the name of the non-primary string. i.e.

v_g_n_offset_string *model->string_name*

or

v_g_n_offset_string *string_name*

and the model is the *defined by offset_model*. *offset_model has been* defined in the section 26.4.8.7.4 Boxes - Offset String Titles/Heights/Depths.

If the *v_g_n_offset_string* parameter does not exist, then the set of vertical geometry parameters is applied to the primary string.

See

26.4.18.1 Extensive Vertical Geometry - Left Hand Labels
26.4.18.2 Extensive Vertical Geometry - Arrow Type
26.4.18.3 Extensive Vertical Geometry - Arrow Text

Or return to 26.4 Long Plot PPF Editor.

26.4.18.1 Extensive Vertical Geometry - Left Hand Labels

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Extensive vertical geometry - Left hand labels parameters

Use set # **input**

set #, as specified by the **define set #** *parameter.*

For the following parameters, n takes the value 1 to 20 and specifies the nth parameter set.

Label X (mm) **v_g_n_label_x** **input**

distance from the left hand side of the labels area to start the left hand label text.

Offset (mm) **v_g_n_label_offset** **input**

distance to raise the left hand label text above arrow line.

Text size (mm) **v_g_n_label_text_size** **input**

size of the left hand label text.

Text colour **v_g_n_label_text_colour** **colour box**

size of the left hand label text.

Text **v_g_n_label_text** **input**

left hand label text.

Textstyle **v_g_n_label_textstyle** **text box**

textstyle for the left hand label text.

Continue to the next section 26.4.18.2 Extensive Vertical Geometry - Arrow Type or return to 26.4.18 Extensive Vertical Geometry.

26.4.18.2 Extensive Vertical Geometry - Arrow Type

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Extensive vertical geometry - Arrow type parameters

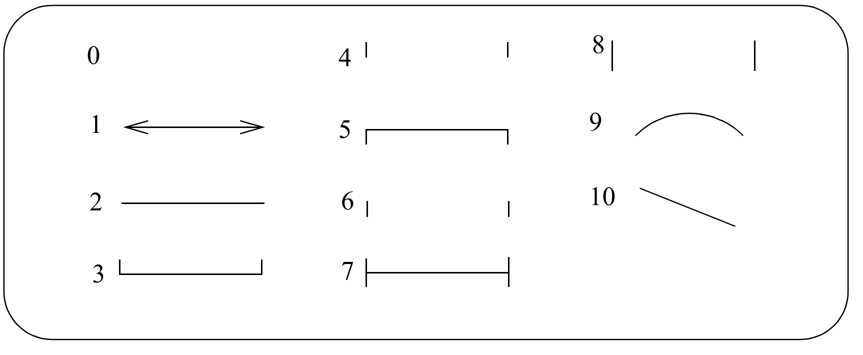
Use set # **input**

set #, as specified by the **define set #** *parameter.*

For the following parameters, n takes the value 1 to 20 and specifies the nth parameter set.

Arrow type mode v_g_n_draw_mode choice box arrow (1)
line (2)
line with uprights at ends (3)
uprights with no lines (4)
line with downrights (5)
downrights with no lines (6)
line with up and downrights at ends (7)
up and downrights with no line (8)
draw curve (9)
draw grade (10)

specifies the arrow type to be drawn.



Left arrow gap (mm) **v_g_n_left_gap** **input**

size of gap for left side for arrow.

Right arrow gap (mm) **v_g_n_right_gap** **input**

size of gap for right side of arrow.

Arrow colour **v_g_n_colour** **colour box**

colour of arrow text

Arrow height (mm) **v_g_n_height** **input**

height of arrow in mm.

Leave gap in arrow for text **v_g_n_gap** **tick box**

if ticked, a gap will be left for text.

Continue to the next section 26.4.18.3 Extensive Vertical Geometry - Arrow Text or return to 26.4.18 Extensive Vertical Geometry.

26.4.18.3 Extensive Vertical Geometry - Arrow Text

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Extensive vertical geometry - Arrow text parameters

Use set # **input**

set #, as specified by the **define set #** *parameter.*

Text colour **v_g_n_text_colour** **colour box**

colour of arrow text

Text size (mm) **v_g_n_text_size** **input**

size of arrow text

Text offset (mm) **v_g_n_text_offset** **input**

distance to raise the text above the arrow line.

Arrow text pre-text **v_g_n_pre_text** **input**

text before the arrow text

Arrow text post-text **v_g_n_post_text** **input**

text after the arrow text

Textstyle **v_g_n_textstyle** **text box**

textstyle of arrow text

Decimal places **v_g_n_no_decimals** **input**

number of decimal places in arrow text.

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Rotate text to fit **v_g_n_rotate** **tick box**

if ticked, the text on the arrows will be rotated to fit.

Please continue to the next section 26.4.19 Labelling Points With Chainage/Height/Grade/Deflection.

26.4.19 Labelling Points With Chainage/Height/Grade/Deflection

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Labelling points with chainage/height/grade/deflection

Define set #		**input**	

where n = 1 to....20. The set enables the specification of a number of parameters for a number of specified labels.

For the following parameters, n takes the value 1 to 20 and specifies the nth parameter set.

Label Type	label_n_type	choice box	chainage of vip, height of vip chainage of vip, height of primary crest sag vtp hcp change of grade mid-ordinate of the vertical curve

The chainage and/or height values for certain points (given by label_n_type) can be labelled.

If *label_n_type* is missing, then the set is ignored.

Label height mode	**label_n_y_mode**	**choice box**	**height above boxes (mm)** height above height value **(mm)** height above primary height **(mm)**

specifies which reference point the label_n_y *distance is measured from.*

See

26.4.19.1 Labelling Points - Label Position
26.4.19.2 Labelling Points - Label Text Type
26.4.19.3 Labelling Points - Label Text

Or return to 26.4 Long Plot PPF Editor.

26.4.19.1 Labelling Points - Label Position

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Ch/Ht/Grade label position parameters

Use set # **input**

set #, as specified by the **Define set #** *parameter.*

For the following parameters, n takes the value 1 to 20 and specifies the nth parameter set.

Distance above point (mm) **label_n_y** **input**

distance above point.

Angle (dms) **label_n_angle** **angle box**

rotation about point.

Angle on grade **tick box**

Distance along from point (mm) **label_n_x** **input**

the distance along from the point in mm.

Text raise height (mm) **label_n_offset** **input**

the distance to raise the text in mm.

Justification **label_n_justification** **choice box** **left end**
middle
end

justification of the label.

Continue to the next section 26.4.19.2 Labelling Points - Label Text Type or return to 26.4.19 Labelling Points With Chainage/Height/Grade/Deflection.

26.4.19.2 Labelling Points - Label Text Type

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Ch/Ht/Grade label text type parameters

Use set # — — **input**

set #, as specified by the **Define set #** *parameter.*

For the following parameters, n takes the value 1 to 20 and specifies the nth parameter set.

Size (mm) **label_n_size** **input**

size of the text.

Colour **label_n_colour** **colour box**

colour of text.

Textstyle **label_n_textstyle** **text box**

the textstyle of the text.

Continue to the next section 26.4.19.3 Labelling Points - Label Text or return to 26.4.19 Labelling Points With Chainage/Height/Grade/Deflection.

26.4.19.3 Labelling Points - Label Text

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Ch/Ht/Grade label text parameters

Use set # **input**

set #, as specified by the **Define set #** *parameter.*

For the following parameters, n takes the value 1 to 20 and specifies the nth parameter set.

Label type	**label_n_value_mode**	**choice box**	**no values labelled** val 1 = chainage val 1 = height **val 1 = chainage, val2 =** height **val 1 = height, val2 =** chainage

the value label mode.

Pre-text **label_n_pre_text** **input**

text before the label text

Mid-text **label_n_mid_text** **input**

text at mid position i.e. between val 1 and val 2.

Post-text **label_n_post_text** **input**

text after the label text

Decimal places val 1 **label_n_no_decimals_1** **input**

number of decimal places in val 1.

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Decimal places val 2 **label_n_no_decimals_2** **input**

number of decimal places in val 2.

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Deflection mode **input**

Please continue to the next section 26.4.20 Labelling Points With Symbols.

26.4.20 Labelling Points With Symbols

Symbols can be placed at certain points given by *symbol_n_type*.

The symbol is drawn in a square box centred on (0,0) with sides of length two millimetres. That is, the box co-ordinates are (-1,-1), (1,1), (1,-1), (-1,-1).

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Labelling points with symbols

Set # — **input**

where n = 1 to....20. The set enables the specification of a number of parameters for a number of specified symbols.

For the following parameters, n takes the value 1 to 20 and specifies the nth parameter set.

Symbol Type	symbol_n_type	choice box	chainage of vip, height of vip chainage of vip, height of primary crest sag vtp hcp change of grade mid-ordinate of the vertical curve

If *symbol_n_type* is missing, then the set is ignored.

Label height mode	symbol_n_y_mode	choice box	height above boxes (mm) height above height value (mm) height above primary height (mm)

specifies which reference point the label_n_y *distance is measured from.*

Distance above point (mm) — **symbol_n_y** — **input**

distance above point given by mode.

Angle (dms) — **symbol_n_angle** — **angle box**

rotation about point.

Distance along from point (mm) — **symbol_n_x** — **input**

the distance along from the point in mm.

Size (mm) — **symbol_n_size** — **input**

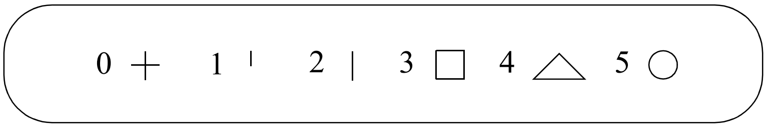
the symbol size in mm.

Colour — **symbol_n_colour** — **colour box**

the symbol size in mm.

Symbol draw mode	symbol_n_draw_mode	choice box	height above boxes (mm) cross (0) upright from centre of box (1) up and downright from centre of box (2) square (3) triangle, base at bottom (4) circle (5)

draw mode for symbol

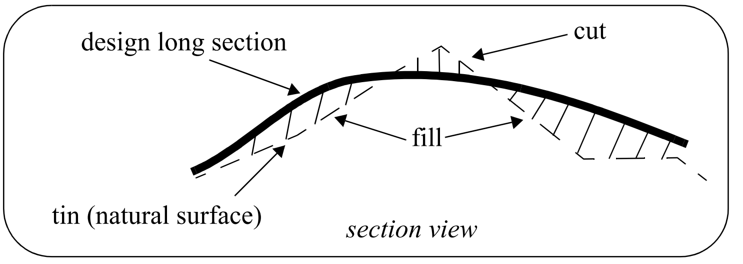


Symbol **input**

Please continue to the next section 26.4.21 Hatching Cut/Fill.

26.4.21 Hatching Cut/Fill

This option is used to hatch cut and/or fill areas between sets of tins.



The **Hatching Cut/Fill** section is common to the **PPF Editors** and is fully documented in 26.2.8 Hatching Cut/Fill.

Section: Hatching cut/fill - Tin parameters - *see* 26.2.8 Hatching Cut/Fill

Section: Hatching cut/fill - Cut parameters - *see* 26.2.8.1 Hatching Cut/Fill - Cut

Section: Hatching cut/fill - Fill parameters- *see* 26.2.8.2 Hatching Cut/Fill - Fill

Please continue to the next section 26.4.22 Cuts.

26.4.22 Cuts

The cuts that the longsection string makes though strings in any user-specified model, can be automatically labelled on the x-section plots.

The **Cuts** section is common to the **PPF Editors** and is fully documented in 26.2.9 Cuts.

Section: Cuts - Model/Name mask parameters - *see* 26.2.9 Cuts.

Section: Cuts - Chainage parameters - *see* 26.2.9.3 Cuts - Chainage - Long Sections Only

Section: Cuts - Height parameters - *see* 26.2.9.4 Cuts - Heights

Section: Cuts - Diameter parameters - *see* 26.2.9.5 Cuts - Diameters

Section: Cuts - Label parameters - *see* 26.2.9.6 Cuts - Labels

Section: Cuts - Symbol parameters - *see* 26.2.9.7 Cuts - Symbols

Continue to the next section 26.4.23 Paired Cuts - Long Section.

26.4.23 Paired Cuts - Long Section

Paired cuts uses pairs of strings and where both strings cut the primary string, the cuts on the long plot can be labelled with information such as

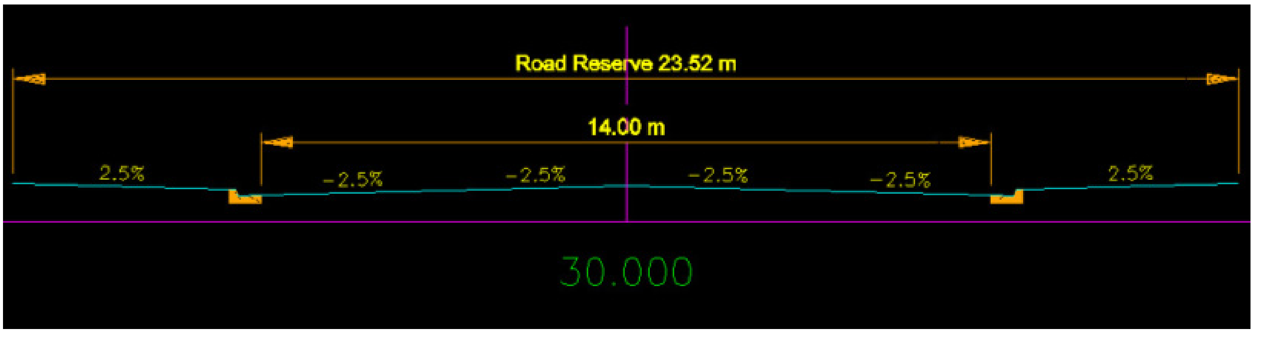
(a) the name of the first and second cut strings

(b) attributes from the first and second cut string

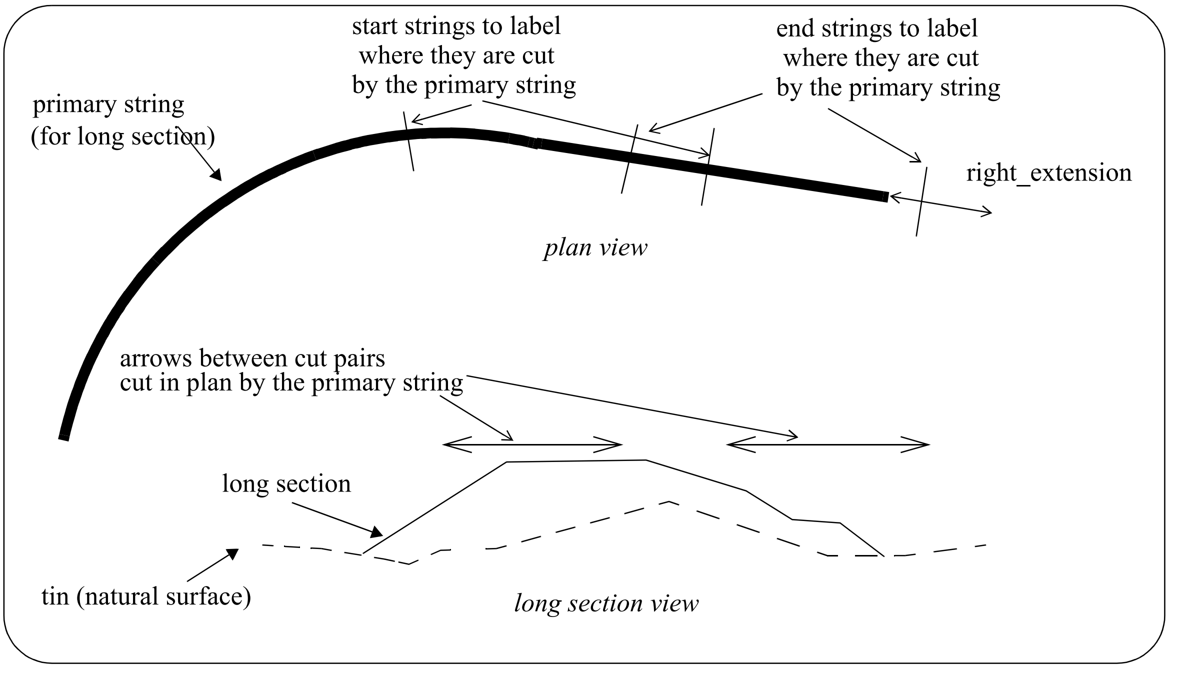
(c) plan distance between the two cuts of the pair

(d) 3d length between the two cuts of the pair

(e) the chainge on the long section of the first and second cuts of the pair



Symbols can be drawn at the chainages of the first and second cuts (at a height specified when defining the Cut sets), and a line drawn between the symbols. Using both the line and a symbol of an arrow head makes an arrow between the two cuts.



The method for specifying which strings are to be checked for paired cuts is by first specifying the **models** (using wild cards and characters) that contains the strings, and then a **start name mask** to select the all the strings that are to be the first strings in a cut pair, and an **end name mask** to select all the strings that are the second strings in a cut pair.

For a long section, all the cuts of the selected **start** strings are found and the cuts ordered by the chainage of the cut with the primary string of the long section. Then all the cuts of the selected

end strings are found and the cuts ordered by the chainage of the cut with the primary string of the long section.

The start cuts are then processed and each start cut is paired with the next end cut with a larger chainage than the start cut. It is possible that there are end cuts before the first start cut (orphaned end cuts) and start cut with no following end cut (orphaned start cuts).

Up to twenty five different sets of models and name masks can be used so that different paired cut sets can be labelled in different ways.

See

26.4.23.1 Paired Cuts (Long Section) - Front Page
26.4.23.2 Paired Cuts (Long Section) - Lines and Symbols
26.4.23.3 Paired Cuts (Long Section) - Labels
26.4.23.4 Paired Cuts (Long Section) - Start Chainage
26.4.23.5 Paired Cuts (Long Section) - End Chainage

Or return to 26.4 Long Plot PPF Editor.

26.4.23.1 Paired Cuts (Long Section) - Front Page

Section: Pairing - Model/Name mask parameters

Panel Field	Parameter name	Type
Define Set #		**input**

set number to be used to define different model/mask sets.

Model	pairing_n_model	**model box**

models (the name can include wild cards (*) and wild characters (?)) from which start and end cut masks are derived.

Start name mask	pairing_n_start_mask	**input**

text string containing the name masks to select the start strings, each separated by one or more spaces, to test the string names against. If the name include spaces then it must be enclosed in the quotes " *(eg* "bench 2"*).*
Each mask can include wild cards (*) and wild characters (?).

For example:	*"ke*"*	*or,*
	"?bank"*	*or, if both masks are required,*
	"ke ?bank*"*	

End name mask	pairing_n_end_mask	**input**

text string containing the name masks to select the end strings, each separated by one or more spaces, to test the string names against. If the name include spaces then it must be enclosed in the quotes " *(eg* "bench 2"*)*
Each mask can include wild cards and wild characters.

For example:	*"ke*"*	*or,*
	"?bank"*	*or, if both masks are required,*
	"ke ?bank*"*	

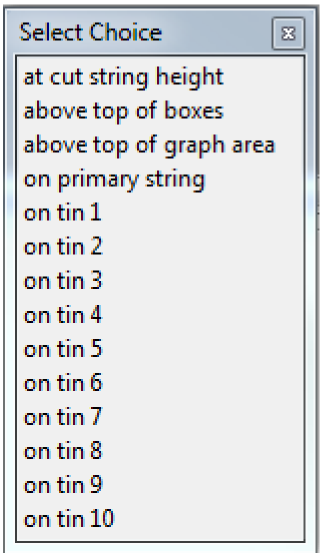
Note

*If **Start name mask** is blank and **End name mask** is not blank, then all strings in the models that are not used as end strings, are used as start string.*

*If **End name mask** is blank and **Start name mask** is not blank, then all strings in the models that are not used as start strings, are used as end string.*

*This is currently not operational: If **Start name mask** and **End name mask** are both blank, then all strings in the models are used. All the strings that cut the primary string of the long section are ordered by offset, and successive pairs of cuts taken to be the cut pairs. There could be one remaining cut point (it has the largest chainage) and if so, it is taken to be orphaned start point.*

Position **pairing_n_position** **choice box**



where to get the z-values to use for the two cut positions. This is used in conjunction with the choice in the **Adopted height** *field.*

If left blank, ***at cut string height*** *is used.*

Adopted height **pairing_n_height** **choice box** **lower position, higher position**

start position, end position

for the choice selected in the **Position** *field, there will be a z-value at the first cut and another z-value at the end cut. The* **Adopted height** *says which of the two z-value to use for placing symbols, lines between cuts, text from Attributes and the Offsets of the first cut and the second cut.*

If lower (higher) position, the smaller (greater) of the two z-values will be used.
If start (end), the z-value of the first (second) cut of the cut point pair will be used.

If left blank, ***lower position*** *is used.*

Include orphaned start ? pairing_n_orphan_start **choice box** **yes, no**

if ***Yes*** *and there are start cuts after the last end cut (and so can't be paired), then the last start cut (i.e. the one with the largest chainage) is taken as an orphaned start and it is labelled as though there is a matching end at the end of the long section.*
if ***No*** *and there are start cuts after the last end cut, then those start cut are ignored.*

If left blank, ***No*** *is used.*

Include orphaned ends ? pairing_n_orphan_end **choice box** **yes, no**

if ***Yes*** *and there are end cuts before the first start cut, then the first end cut (the one with the smallest chainage) is taken as an orphaned end and it is labelled as though there is a matching start at the beginning of the long section.*
if ***No*** *and there are end cuts before the first start cut, then those end cut are ignored.*

If left blank, ***No*** *is used.*

Continue to the next section 26.4.23.2 Paired Cuts (Long Section) - Lines and Symbols or return to 26.4.23 Paired Cuts - Long Section.

26.4.23.2 Paired Cuts (Long Section) - Lines and Symbols

Symbols can be placed at the start and end cut chainages (at the z-value given by the **Position** and **Adopted height** columns when defining the Sets), and a line can also be drawn between the cuts at that z-value. Having both the line and a symbol forms an arrow between the cuts pair.

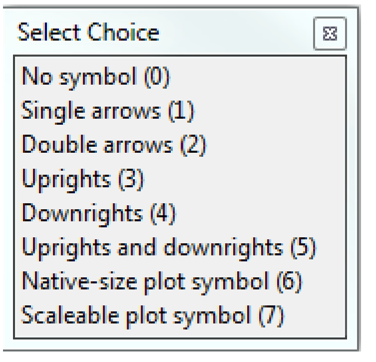
Section: Pairing - Lines and symbol parameters

Panel Field **Parameter name** **Type** **Pop-Up**

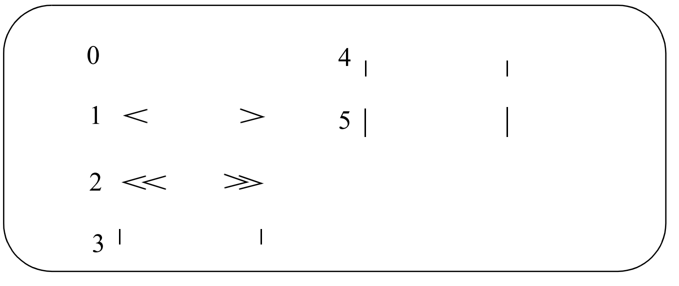
Use Set # **input**

set number as specified in the **Define set #**.

Symbol type **pairing_line_symbols_n_symbol** **choice box**



type of symbol at each end of the cut pair.



for cases 6 and 7, a plot symbol is to be used and is given in the Symbol field

The height used for the symbol at both ends of the cuts pair is given by the **Position** *and* **Adopted height** *column in the Set definition.*

Symbol **pairing_line_symbols_n_user_symbol** **plot symbols**

for Symbol type 6 and 7, the plot symbol to be used at each cut point.

For Symbol type 0 to 5, this field is not used.

The height used for the symbol at both ends is given by the **Position** *and* **Adopted height** *columns in the Set definition.*

Symbol colour **pairing_line_symbols_n_symbol_colour** **colour box**

colour of the symbol

Draw line **pairing_line_symbols_n_line** **choice box** **No line, Draw line**

if Draw line, draw a line between the start and end cuts for each cut pair.
If No line, no line is drawn between the start and end cuts for each cut pair.

The height used for the line at both ends is given by the **Position** *and* **Adopted height** *columns in the Set definition.*

Line colour **pairing_line_symbols_n_line_colour** **colour box**

colour of the line between the start and end cuts for each cut pair

Left symbol rotation **pairing_line_symbols_n_left_rotate** **measure box**

angle (measured counterclockwise from the positive x-axis with units of degrees in 4.17.1 HP Notation*) to rotate the symbol at the start cut point for the cut point pair.*

If there is no value then the rotation is 0.

Right symbol rotation **pairing_line_symbols_n_right_rotate** **measure box**

angle (measured counterclockwise from the positive x-axis with units of degrees in 4.17.1 HP Notation*) to rotate the symbol at the end cut point for the cut point pair.*

If there is no value then the rotation is 0.

X (mm) **pairing_line_symbols_n_x** **measure box**

horizontal adjustment to the position of the symbol.

Y (mm) **pairing_line_symbols_n_y** **measure box**

vertical adjustment to the position of the symbol.

Size **pairing_line_symbols_n_size** **measure box**

size of symbol. A value of 0 means no symbol.

Continue to the next section 26.4.23.3 Paired Cuts (Long Section) - Labels or return to 26.4.23 Paired Cuts - Long Section.

26.4.23.3 Paired Cuts (Long Section) - Labels

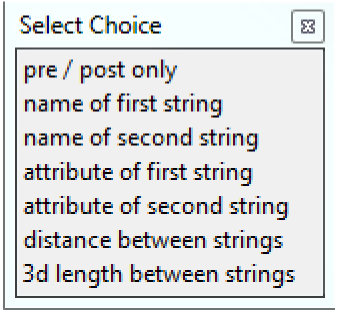
Section: Pairing - Label parameters

Panel Field **Parameter name** **Type** **Pop-Up**

Use Set # **input**

set number as specified in the **Define set #***.*

Mode **pairing_label_n_mode** **choice box**



type of label to write between the start and end points of the cuts pair

The height used for the label is given by the **Position** *and* **Adopted height** *columns in the Set definition.*

Attribute name **pairing_label_n_attribute** **text box**

if **Mode** *is* ***attribute of first string*** *or* ***attribute of second string****, this is the name of the attribute to use as the label*

Offset **pairing_label_n_offset** **measure box**

the text is raised by this amount above the line between the cut pair

Colour **pairing_label_n_colour** **colour box**

colour of the text

Size (mm) **pairing_label_n_size** **measure box**

size of the text in millimetres. A value of 0 means no text.

Textstyle **pairing_label_n_textstyle** **textstyle box**

the textstyle for the text

Pre-text **pairing_label_n_pre_text** **text box**

text to draw before the label. This can include spaces, including one or more spaces after the last non blank character of **Pre-text***.*

Post-text **pairing_label_n_post_text** **text box**

text to draw after the label. This can include spaces, including one or more spaces before the first non blank character of **Post-text***.*

Decimals **pairing_label_n_no_decimals** **number box**

if the label is a number, then **Decimals** *is the number of decimal places to write the number out to.*

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Leave gap for text pairing_label_n_?? **tick box**

if ticked then a gap in the line between the cuts pair is left large enough for the label and the pre-text and post-text.

If not ticked then no gap is left in the line between the cuts pair.

Rotate text to fit pairing_label_n_roate_text **tick box**

if ticked, then if the label and the pre-text and post-text will not fit between the start cut and end cut of *the cut pair, the text is rotated through ninety degrees.*

If not ticked, then the label text is drawn even though it will run over the ends of the cut pair.

Continue to the next section 26.4.23.4 Paired Cuts (Long Section) - Start Chainage or return to 26.4.23 Paired Cuts - Long Section.

26.4.23.4 Paired Cuts (Long Section) - Start Chainage

The Chainge (on the long section) of the first point of the cut pair can be labelled.

The text for the label consists of pre-text followed by Chainage value followed by post-text.

Section: Pairing - Start Chainage parameters

Panel Field **Parameter name** **Type** **Pop-Up**

Use Set # **input**

set number as specified in the **Define set #**.

The height used for the text is given by the **Position** *and* **Adopted height** *columns in the Set definition. This is then adjusted by the field* **Y (mm).**

X (mm) **pairing_start_chainage_n_x** **measure box**

for placing the text: horizontal adjustment from the chainage of the first cut.

Y (mm) **pairing_start_chainage_n_y** **measure box**

for placing the text: the vertical adjustment to the height given by the **Position** *and* **Adopted height** *columns in the Set definition*

Angle (dms) **pairing_start_chainage_n_angle** **measure box**

angle (measured counterclockwise from the positive x-axis with units of degrees in 4.17.1 HP Notation*) to rotate the text*

Colour **pairing_start_chainage_n_colour** **colour box**

colour of the text

Size (mm) **pairing_start_chainage_n_size** **measure box**

size of the text in millimetres. A value of 0 means no text.

Textstyle **pairing_start_chainage_n_textstyle** **textstyle box**

the textstyle for the text

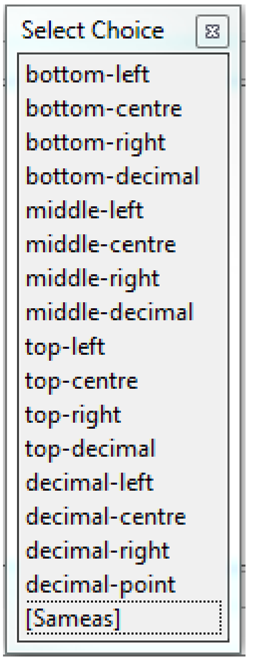
Pre-text **pairing_start_chainage_n_pre_text** **text box**

text before the Chainage value. This can include spaces, including one or more spaces after the last non blank character of **Pre-text**.

Post-text **pairing_start_chainage_n_post_text** **text box**

text after the Chainage value. This can include spaces, including one or more spaces before the first non blank character of **Post-text**.

Justification **pairing_start_chainage_n_justification** **justification box**



justification of the text.

Decimals **pairing_start_chainage_n_no_decimals** **number box**

the number of decimal places for the offset value

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Continue to the next section 26.4.23.5 Paired Cuts (Long Section) - End Chainage or return to 26.4.23 Paired Cuts - Long Section.

26.4.23.5 Paired Cuts (Long Section) - End Chainage

The Chainage on the long section of the second point of the cut pair can be labelled.

The label consists of pre-text then the Chainage value followed by post-text.

Section: Pairing - End Chainage parameters

Panel Field **Parameter name** **Type** **Pop-Up**

Use Set # **input**

set number as specified in the **Define set #**.

The height used for the text is given by the **Position** *and* **Adopted height** *columns in the Set definition. This is then adjusted by the field* **Y (mm).**

X (mm) **pairing_end_chainage_n_x** **measure box**

for placing the text: horizontal adjustment from the offset of the second cut.

Y (mm) **pairing_end_chainage_n_y** **measure box**

for placing the text: the vertical adjustment to the height given by the **Position** *and* **Adopted height** *columns in the Set definition*

Angle (dms) **pairing_end_chainage_n_angle** **measure box**

angle (measured counterclockwise from the positive x-axis with units of degrees in 4.17.1 HP Notation*) to rotate the text*

Colour **pairing_end_chainage_n_colour** **colour box**

colour of the text

Size (mm) **pairing_end_chainage_n_size** **measure box**

size of the text in millimetres. A value of 0 means no text.

Textstyle **pairing_end_offset_n_textstyle** **textstyle box**

the textstyle for the text

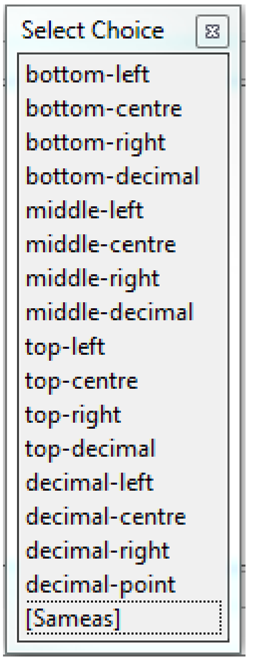
Pre-text **pairing_end_chainage_n_pre_text** **text box**

text before the Offset value. This can include spaces, including one or more spaces after the last non blank character of **Pre-text**.

Post-text **pairing_end_chainage_n_post_text** **text box**

text after the Offset value. This can include spaces, including one or more spaces before the first non blank character of **Post-text**.

Justification **pairing_start_chainage_n_justification** **justification box**



justification of the text.

Decimals **pairing_start_chainage_n_no_decimals** **number box**

the number of decimal places for the Offset value

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Please continue to the next section 26.4.24 Primary String Name Label.

26.4.24 Primary String Name Label

The plot can be labelled with a name under the boxes area.

The name is made up of a concatenation of the text string names.

Section: Primary string name label parameters

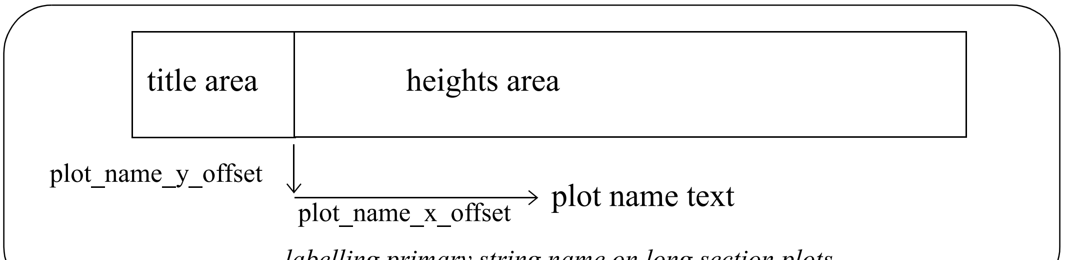
Panel Field	Parameter name	Type
Name mode	**plot_name_string_name**	**choice box**
determines what text is placed in the name labels.		
Pre-text	**plot_name_pre_text**	**input**
text before label.		
Post-text	**plot_name_post_text**	**input**
text after label.		
Textstyle	**plot_name_textstyle**	**text box**
textstyle of label.		
Size (mm)	**plot_name_size**	**input**
size of label.		
Colour	**plot_name_colour**	**colour box**
colour of label.		
X offset (mm)	**plot_name_x_offset**	**input**
horizontal adjustment to position of label.		
Y offset (mm)	**plot_name_y_offset**	**input**
vertical adjustment to position of label.		

The plot name is positioned under the boxes.

The *plot_name_x_offset* is measured from the beginning of the height boxes.

The default for *plot_name_x_offset* is centred on heights area.

The *plot_name_y_offset* is measured from the bottom of the box area with positive being down.



labelling primary string name on long section plots

Please continue to the next section 26.4.25 Scale Labelling.

26.4.25 Scale Labelling

The plot can be labelled with the horizontal and vertical scale under the boxes area.

The scale label is made up of a concatenation of the text strings:

scale_horizontal_pre_text horizontal scale value scale_horizontal_post_text

and

scale_vertical_pre_text vertical scale value scale_vertical_post_text

The horizontal scale value is the value given by the scale parameter.

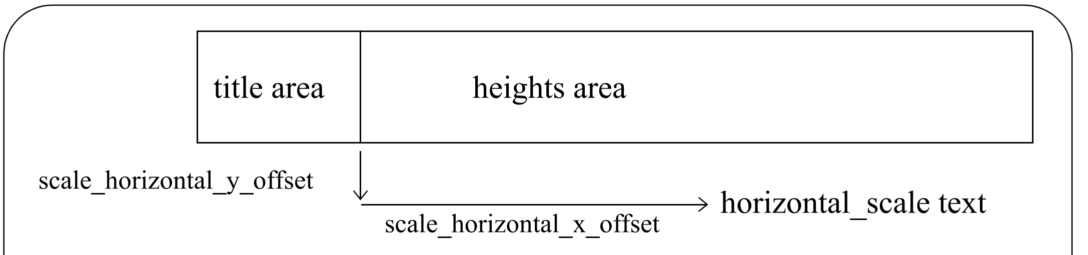
The scales are positioned under the boxes.

See

26.4.25.1 Scale Labelling - Horizontal
26.4.25.2 Scale Labelling - Vertical

Or return to 26.4 Long Plot PPF Editor.

26.4.25.1 Scale Labelling - Horizontal



labelling of scales on the long section plots

Section: Scale labelling - Horizontal parameters

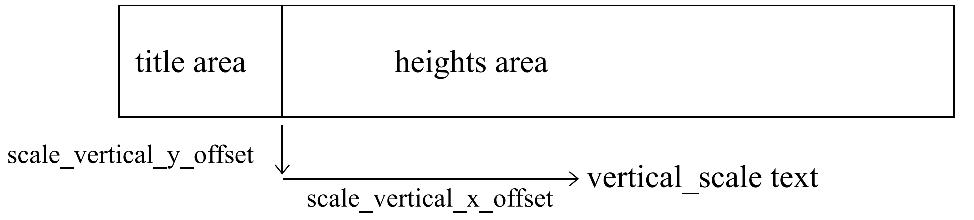
The fields and buttons used in this section have the following functions.

Panel Field	Parameter name	Type	Pop-Up
Pre-text	scale_horizontal_pre_text	text box	
text before scale label			
Post-text	scale_horizontal_post_text	text box	
text after scale label			
Textstyle	scale_horizontal_textstyle	textstyle box	
textstyle of scale label			
Size (mm)	scale_horizontal_size	real box	
size of scale label			
Colour	scale_horizontal_colour	colour box	
colour of scale label			
X offset (mm)	scale_horizontal_x_offset	real box	
horizontal adjustment to position the label. It is measured from the beginning of the heights area.			
Y offset (mm)	scale_horizontal_y_offset	real box	
vertical adjustment to position the scale label. It is measured down from the bottom of the boxes.			
Decimal places	scale_horizontal_decimals	integer box	
number of decimal places in the scale label.			

*If > 0, trailing zeros are **removed** after the decimal point.*
If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Continue to the next section 26.4.25.2 Scale Labelling - Vertical or return to 26.4.25 Scale Labelling.

26.4.25.2 Scale Labelling - Vertical



labelling of scales on the long section plots

Section: Scale labelling - Vertical parameters

The fields and buttons used in this section have the following functions.

Field Description	Parameter name	Type	Pop-Up
Pre-text	scale_vertical_pre_text	text box	

text before label

Post-text	scale_vertical_post_text	text box	

text after label

Textstyle	scale_vertical_textstyle	textstyle box	

textstyle of label

Size (mm)	scale_vertical_size	real box	

size of label

Colour	scale_vertical_colour	colour box	

colour of label

X offset (mm)	scale_vertical_x_offset	real box	

horizontal adjustment to the position of label. This is measured from the beginning of the heights area.

Y offset (mm)	scale_vertical_y_offset	real box	

vertical adjustment to the position of label. This is measure in a downward direction from the bottom of the boxes.

Decimal places	scale_vertical_decimals	number box	

number of decimal places in the label.

*If > 0, trailing zeros are **removed** after the decimal point.*
If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Please continue to the next section 26.4.26 Plan Plotting.

26.4.26 Plan Plotting

Section: Plan parameters

The *Plan Plotting* section is for generating a combined long section and plan plot on the one plot sheet. The chainage interval for the long section plots (as set up in the ***Pagination*** section 26.4.7 Pagination) is used to define the chainage interval for the plan plot.

The total sheet size is given by the ***Sheet size*** parameter (see the section 26.4 Long Plot PPF Editor), and the position of the long section plot on the sheet is defined in the section 26.4.6 Plot Sheet Layout - Long Section. What is left to be defined is the area on the sheet for the plan plot.

Hence sheets of long sections and the associated plan plots are automatically generated.

The fields and buttons used in this section have the following functions.

Field Description | **Parameter name** | **Type** | **Pop-Up**

Use plan plotting | **plan_plotting** | **tick box**

*if **tick**, a plan plot to match the long section plot, is created on the same sheet.*
*If **not tick**, no plan plot is created - only the long section.*

Left margin (mm) | **plan_left_margin** | **input**

left margin of the area of the plan plot

Right margin (mm) | **plan_right_margin** | **input**

right margin of the area of the plan plot

Top margin (mm) | **plan_top_margin** | **input**

top margin of the area of the plan plot

Bottom margin (mm) | **plan_bottom_margin** | **input**

bottom margin of the area of the plan plot

Gap from left margin to start chainage (mm) | **plan_left_margin_gap** | **input**

Draw border around plan plot | **plan_draw_border** | **tick box**

*if **ticked**, a border is drawn around the plan plot area*

Border colour | **plan_border_colour** | **colour box**

colour of border

Symbol for start chainage | **plan_start_symbol** | **symbol box**

symbol to place in the plan area on the primary string at the start chainage of the long section plot

Symbol for end chainage | **plan_end_symbol** | **symbol box**

symbol to place in the plan area on the primary string at the end chainage of the long section plot

Symbol size (mm) | **plan_symbol_size** | **input**

size of the symbols

Symbol colour | **plan_symbol_colour** | **colour box**

colour of the symbols

View to plot | **plan_view_name** | **view box**

view containing the information to be plotted in the plan area

Please continue to the next section 26.4.27 PPFs To Include - Long Section.

26.4.27 PPFs To Include - Long Section

This is documented for all the PPF Editors in 26.2.10 PPFs To Include.

Please continue to the next section 26.4.28 Buttons at Bottom of Panel - Long Section.

26.4.28 Buttons at Bottom of Panel - Long Section

This is documented for all the PPF Editors in 26.2.2 View to Load and Global Variables.

Return to 26.4 Long Plot PPF Editor.

26.5 Drainage Long Plot PPF Editor

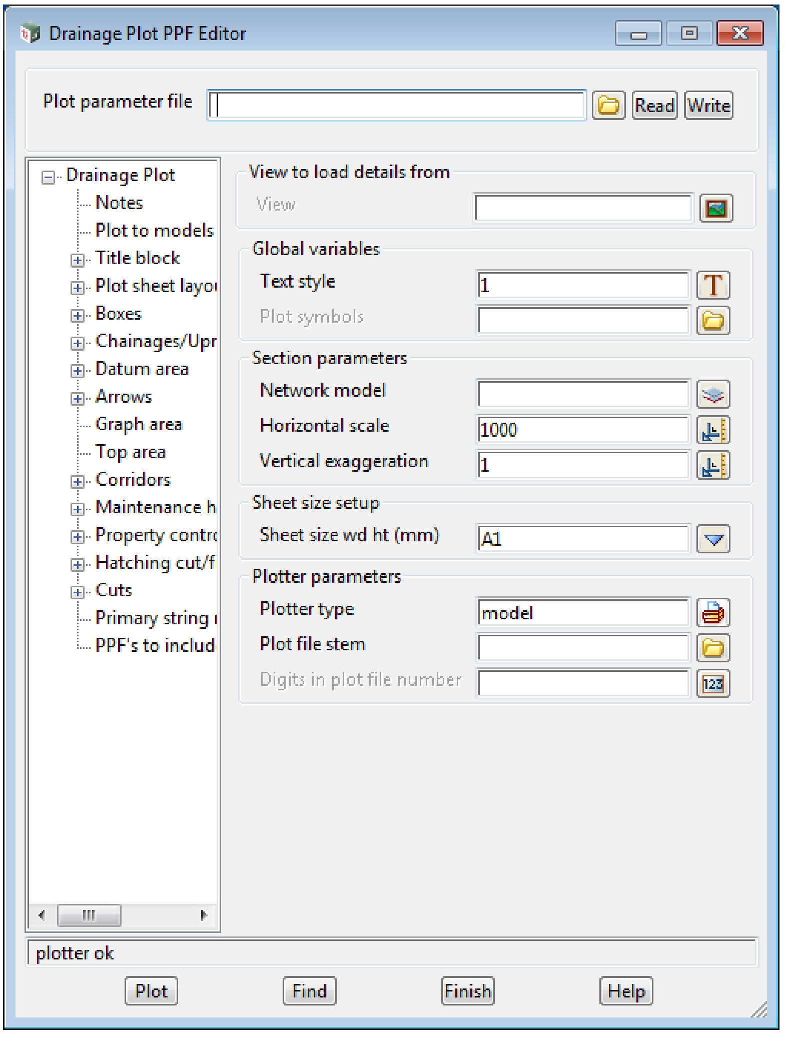
Position of option on menu: **Plot =>Plot and PPF Editors => Drainage long**

The **Drainage Plot PPF Editor** is for creating and/or editing a (binary) drainage and sewer long section ppf file and for creating a drainage and/or sewer long section plot.

Note: Binary and text PPFs are stored *within* the project (not in the folder containing the project).

On selecting the **Drainage long** option, the **Drainage Plot PPF Editor** panel is displayed.

The plot parameters for controlling the cross section plots are accessed by expanding to the appropriate node in the ***Drainage Plot*** tree (click on the **+** to expand to node or **-** to collapse the node) and then clicking on the required node, and the required information to fill in is displayed on the right hand side of the panel.



For information on all the different nodes see:

[26.5.1 General Information on Drainage Long Section Plots]

[26.5.2 Drainage Plot - Front Page]

[26.5.3 Notes - Drainage Long Plot]

[26.5.5 Title Block - Drainage Long Plot]

[26.5.6 Plot Sheet Layout]

[26.5.7 Boxes]

[26.5.8 Chainages/Uprights]

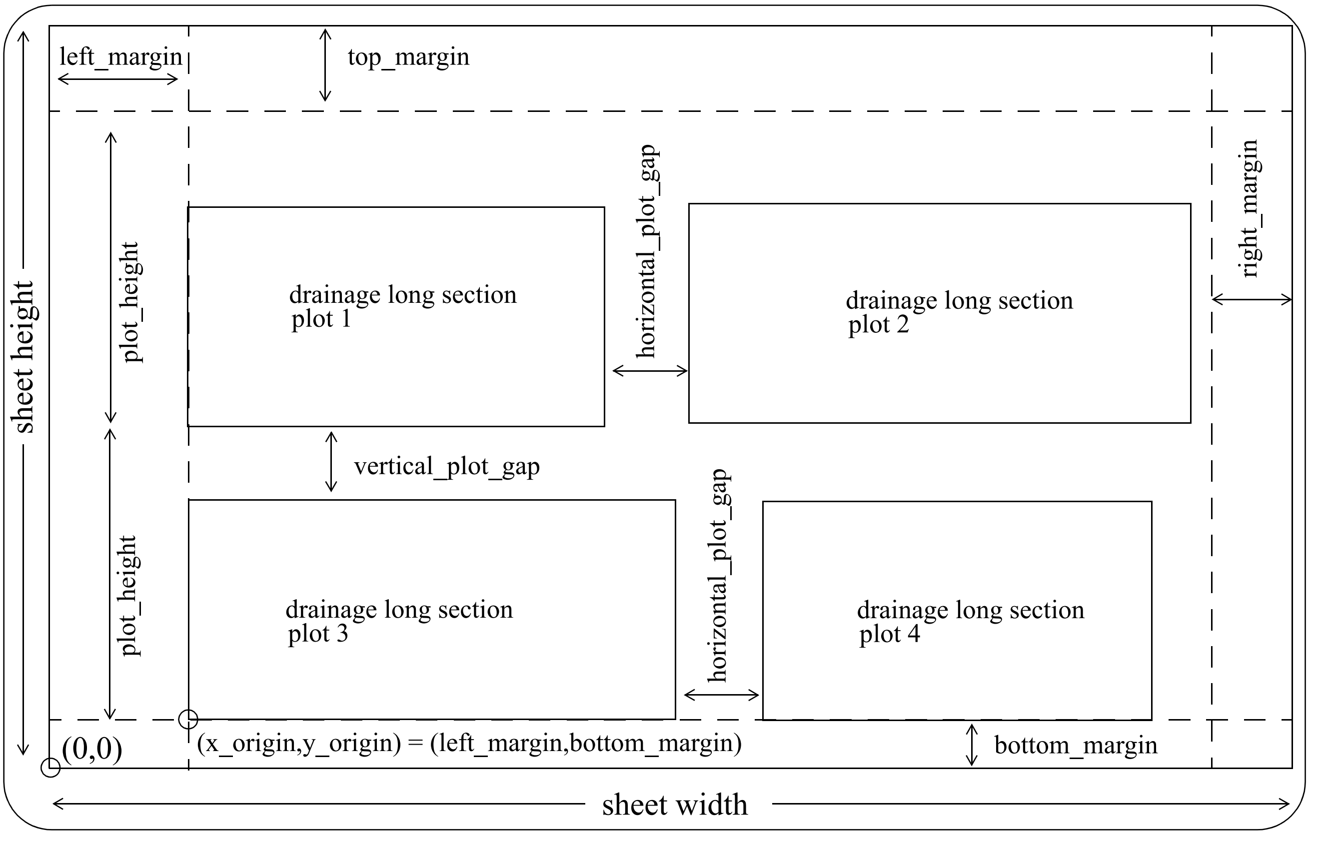
26.5.1 General Information on Drainage Long Section Plots

The Drainage Long Section Plot generates the longsection plots for all lines in a drainage network.

Given the plot sheet size and the horizontal and vertical scales, the long sections for the drainage lines are plotted in string name alphabetical order starting at the top of the sheet and moving across the sheet. Once one row is full, if there is room the plot moves down the page and begins a new row. When a plot sheet is full, a new plot sheet is automatically begun.

Hence the drainage lines are plotted one after another on one or more plot sheets.

.



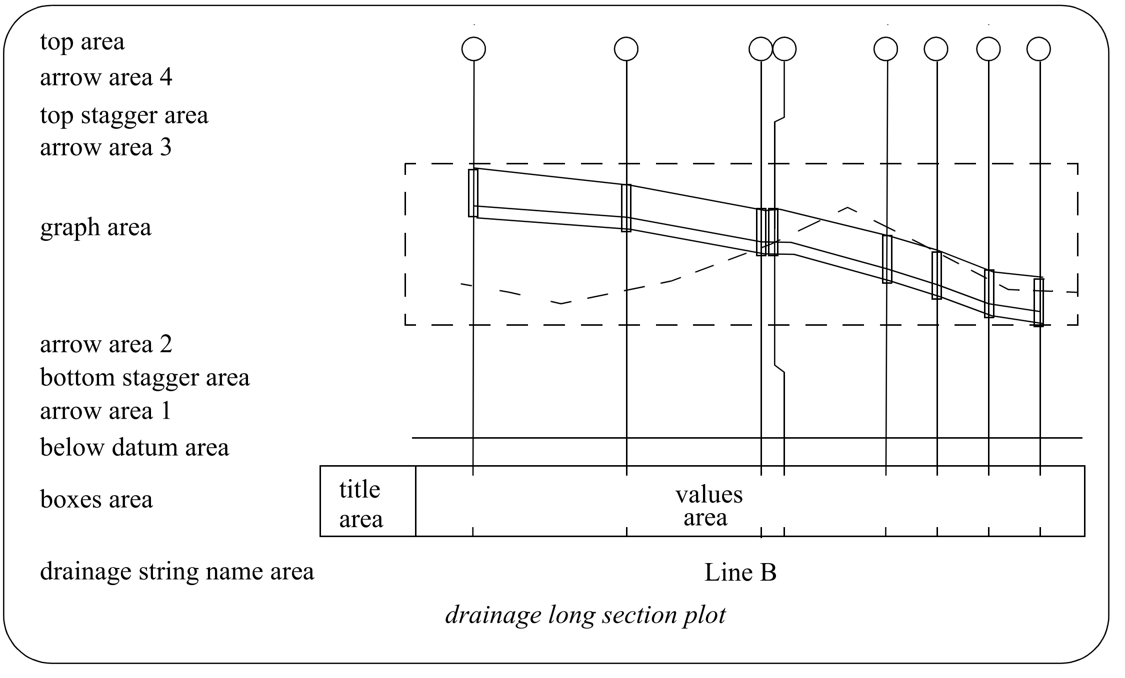
Each individual drainage long section plot consists of eleven areas.

From the bottom up, they are

1. The **drainage string name area** is where the name of the drainage string can be plotted.
2. The **boxes area** is where the chainages and various values for the drainage strings are labelled.
3. The **below datum area** is a region between the boxes area and the datum line.
4. The **arrow 1 area** is for drawing arrows where the arrows go between the staggered uprights and below the graph area. The datum line is at the bottom of the arrow 1 area.
5. The **bottom stagger area** is where the upright line staggers occur before going up from the boxes area to the graph area.
6. The **arrow 2 area** is for drawing arrows below the graph but where the arrows go between non-staggered uprights.
7. The **graph area** is the area where the actual plots of the strings are drawn.
8. The **arrow 3 area** is for drawing arrows above the graph area and where the arrows go between non-staggered uprights.

9. The **top stagger area** is where the upright line staggers occur above the graph area.
10. The **arrow 4 area** is for drawing arrows where the arrows go between the staggered uprights and above the graph area.
11. The **top area** is an annotation area above the arrow 4 area and is used for bubbles, manhole names (pit names), junctions, deflection angles *etc*.

The areas and the information in them will now be described in more detail.



The drainage longsection plot includes

(a) the manholes, drainage pipe and any house connections
(b) the height of the finished surface at the manhole
(c) manhole names and cover types
(d) distances between manholes
(e) the invert depth of the pipe on either side of a manhole
(f) the grades and types of the pipes
(g) any services in the corridor - including their name, invert level and distance from the nearest downstream manhole
(h) if the information exists, the velocity, flow, HGL values and diagram

All the required parameters for controlling the drainage long section plot are set up in the ***Drainage Plot PPF Editor*** and will be described in detail in the following sections.

Please continue to the next section 26.5.2 Drainage Plot - Front Page or return to 26.5 Drainage Long Plot PPF Editor.

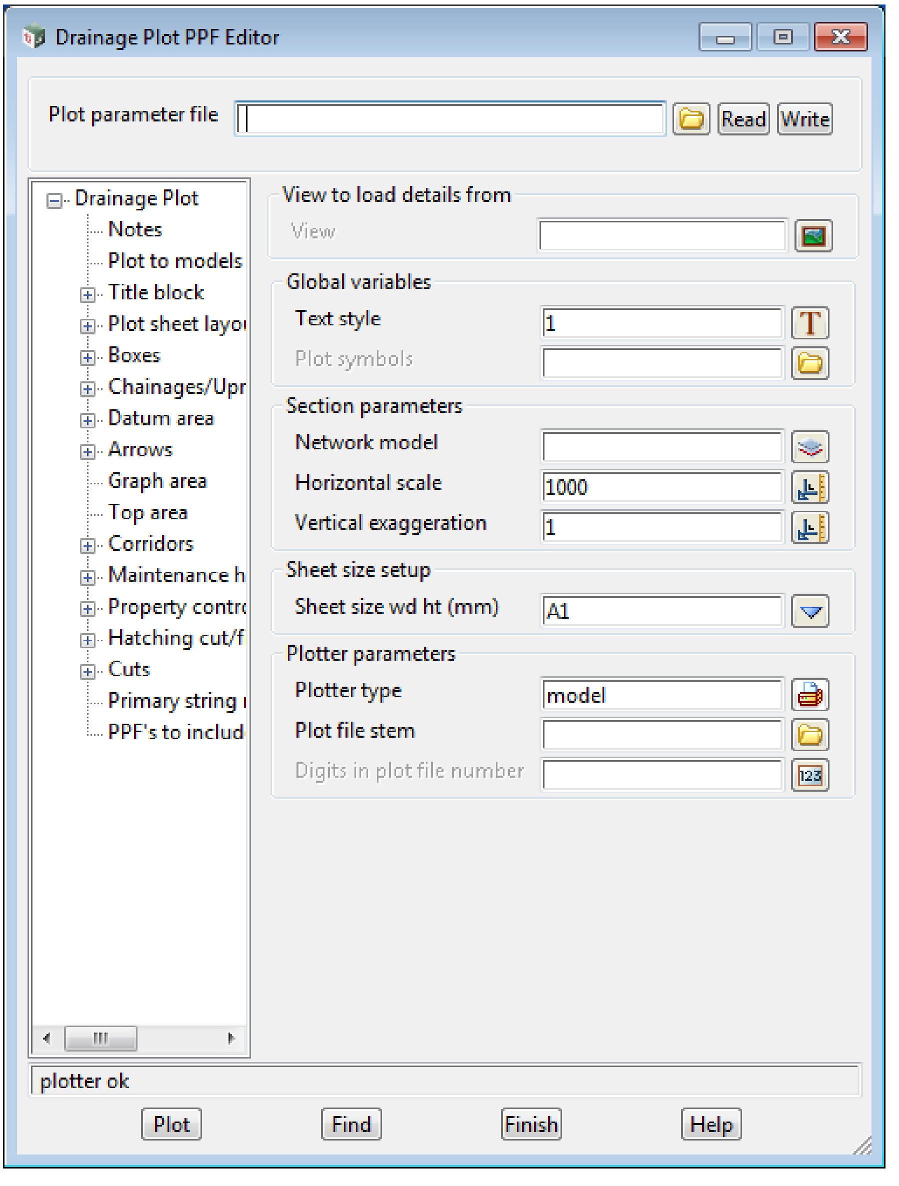
26.5.2 Drainage Plot - Front Page

The drainage long plot itself consists of the three regions - boxes, datum and graph areas.

The **boxes area** is where the titles and the chainage values and the heights/depths for the strings drawn on the long plot are labelled.

The **datum area** is the region between the boxes area and the graph area.

The **graph area** is the area where the actual plots of the strings are drawn.



Section: Plot parameter file

*a **plot parameter file** can be used to load values into the fields of the PPF Editor, or as a file to write out all current values in a PPF Editor to.*

This section is documented for all the PPF Editors in 26.2.1 Plot Parameter File.

Section: Section: View to load details from AND Global variables

*a **section view** can be selected to load certain values into fields of the PPDF Editor. For example,* *Vertical exaggeration and Corridor models. And there are variables to use if some values aren't given.*

These sections are documented for all the PPF Editors in 26.2.2 View to Load and Global Variables.

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Section parameters

Network model | **network_model** | **model box** | **available models**

model containing the drainage/sewer strings to draw in long section on the plot.

Horizontal Scale | **scale** | **input**

horizontal scale to be used for the long section plots

Vertical exaggeration | **vertical_exaggeration** | **input**

vertical scale to be used for the long section plots

Section: Sheet size setup and Plotter parameters

these sections define the size of the "paper" to plot on, the type of plotter to use and the naming to use for the plot files.

These sections are documented for all the PPF Editors in 26.2.3 Sheet Size and Plotter Parameters.

Please continue to the next section 26.5.3 Notes - Drainage Long Plot.

26.5.3 Notes - Drainage Long Plot

This is documented for all the PPF Editors in 26.2.4 Notes.

Please continue to the next section 26.5.4 Plot to Models - Drainage Long Plot.

26.5.4 Plot to Models - Drainage Long Plot

This is documented for all the PPF Editors in 26.2.5 Plot to models.

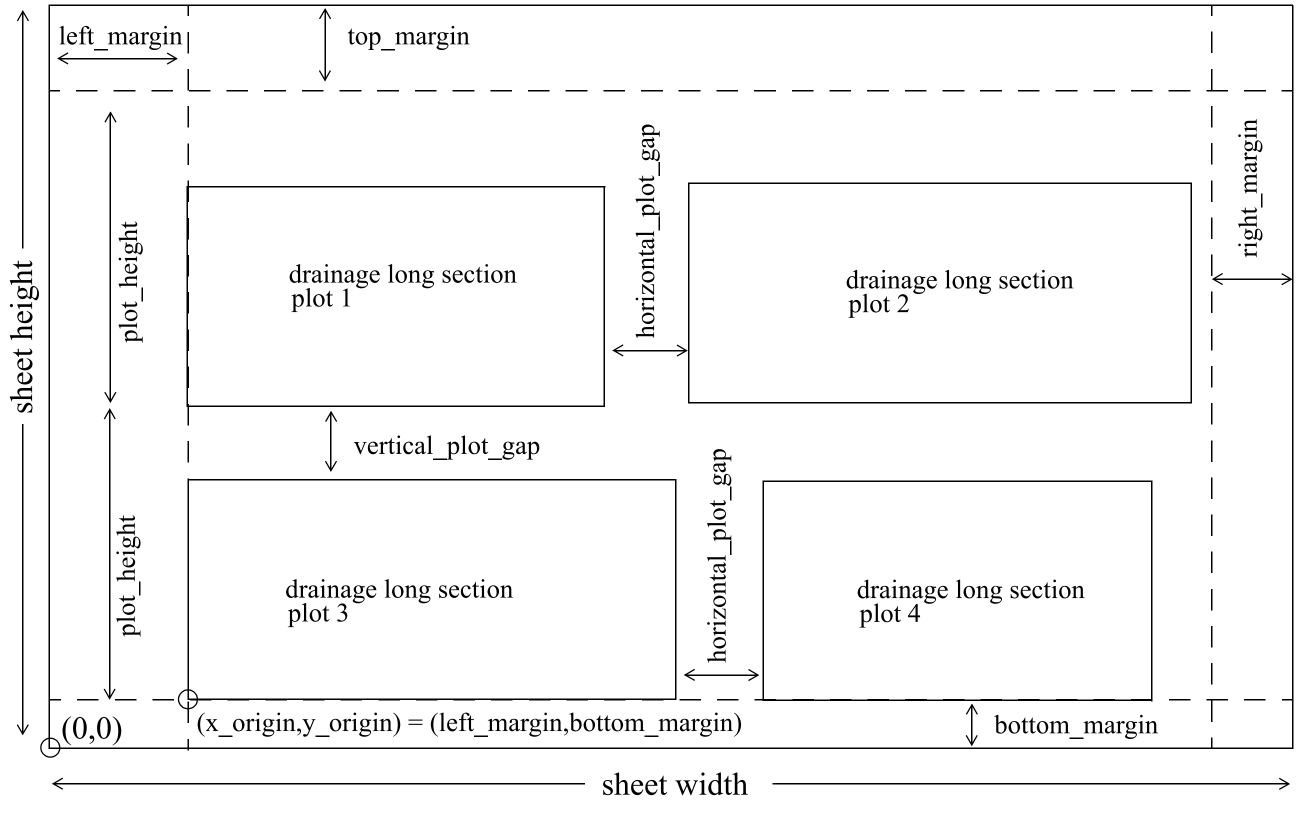
Please continue to the next section 26.5.5 Title Block - Drainage Long Plot.

26.5.5 Title Block - Drainage Long Plot

This is documented for all the PPF Editors in 26.2.6.1 Title Block Section in PPF Editors.
For more general information about a title block, see 26.2.6 Title Block.

Please continue to the next section 26.5.6 Plot Sheet Layout.

26.5.6 Plot Sheet Layout

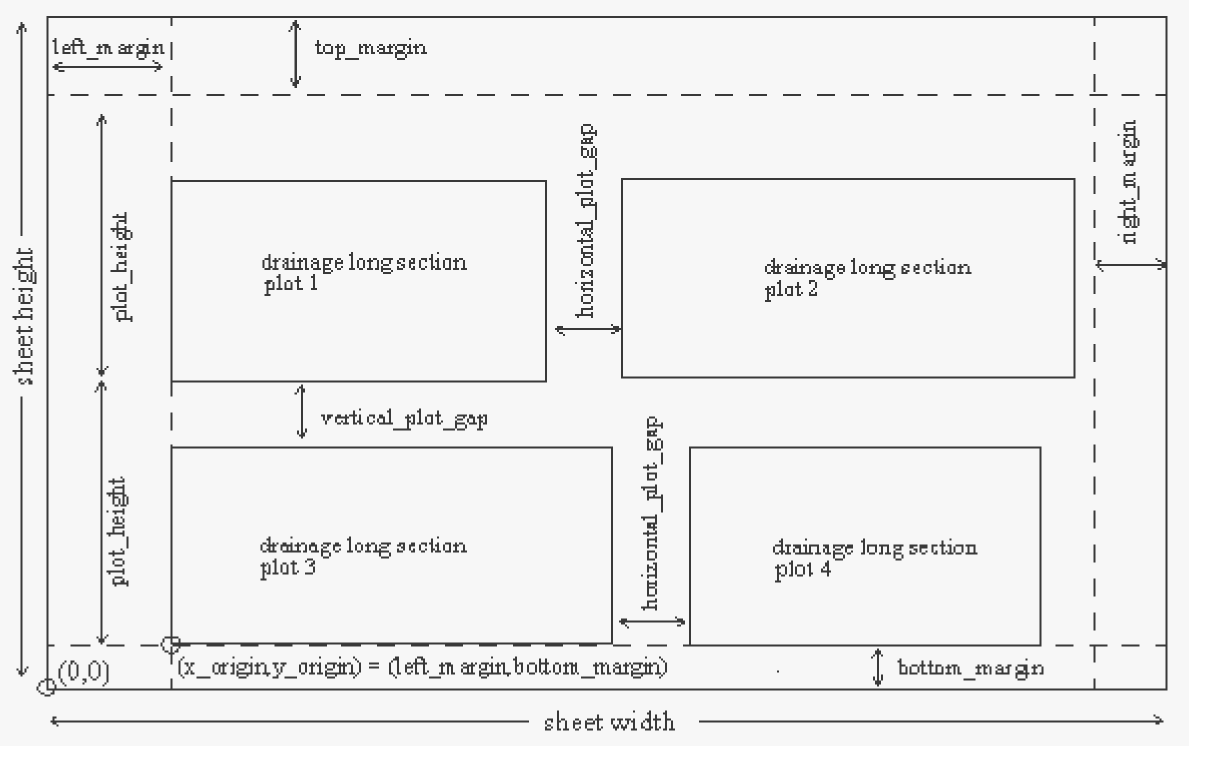


See

26.5.6.1 Plot Sheet Layout - Margins
26.5.6.2 Plot Sheet Layout - Other Parameters

Or return to 26.4 Long Plot PPF Editor.

26.5.6.1 Plot Sheet Layout - Margins



Section: Plot sheet layout - Margins

Panel field	Parameter name	Type
Left (mm)	**left_border_gap**	**input**

left border gap (in millimetres).

Right (mm)	**right_border_gap**	**input**

right border gap (in millimetres).

Top (mm)	**top_border_gap**	**input**

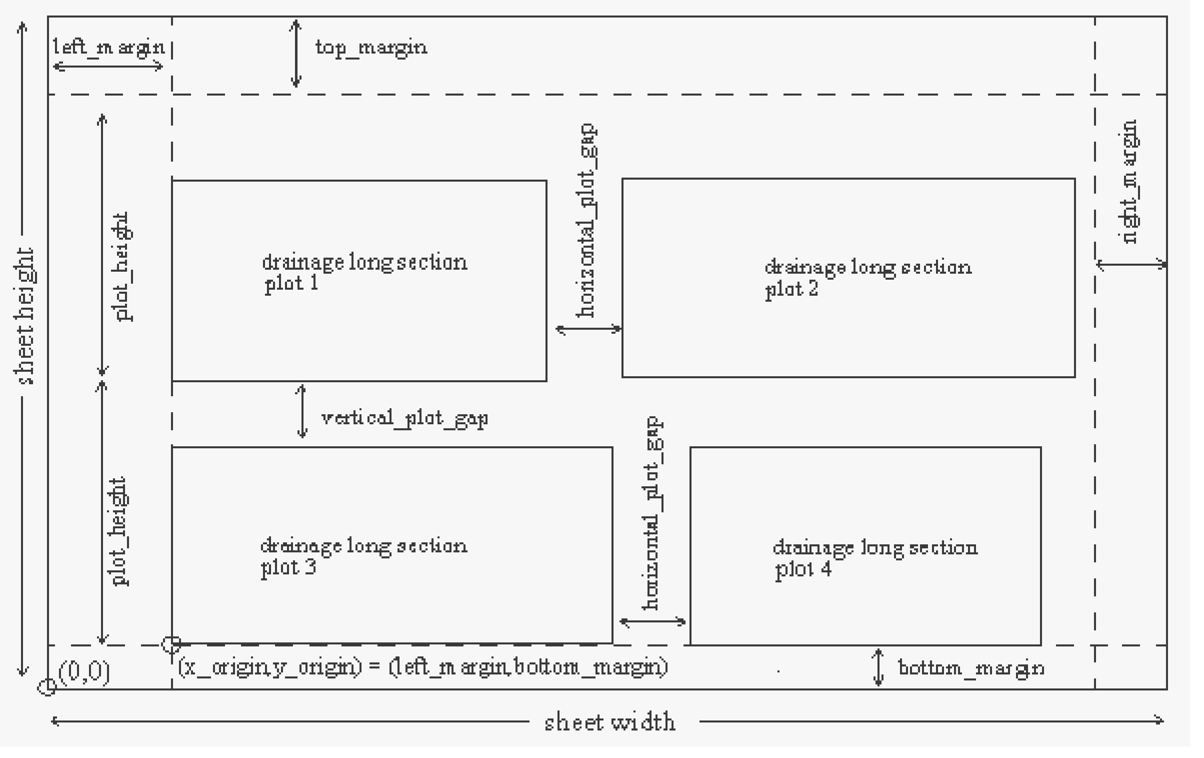
top border gap (in millimetres).

Bottom (mm)	**bottom_border_gap**	**input**

bottom border gap (in millimetres).

Continue to the next section 26.5.6.2 Plot Sheet Layout - Other Parameters or return to 26.5.6 Plot Sheet Layout.

26.5.6.2 Plot Sheet Layout - Other Parameters



Section: Plot sheet layout - Other parameters

Panel field	Parameter name	Type	Pop-Up
Row per sheet mode	**only_one_line**	**choice box**	**more than one row on sheet**
sheet			
			only one row on sheet

whether to allow multiple rows on a sheet.

Plot height (mm)	**plot_height**	**input**	

height of plot row (in millimetres).

Horizontal plot gap (mm)	**horizontal_plot_gap**	**input**	

gap between plot columns (in millimetres).

Vertical plot gap (mm)	**vertical_plot_gap**	**input**	

gap between plot rows (in millimetres).

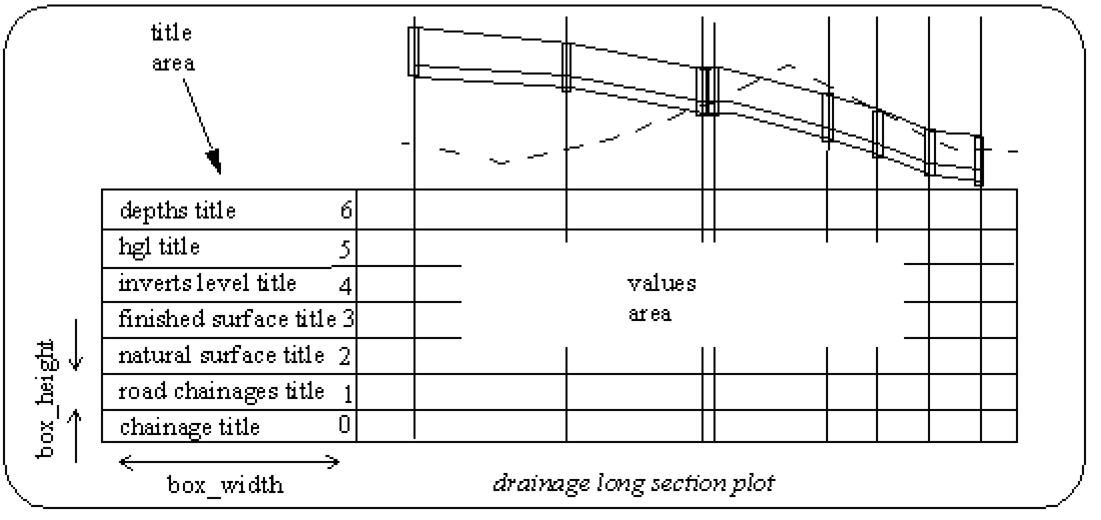
Start new sheet/row for strings longer than (world units) string_length_for_break input

Continue to the next section 26.5.7 Boxes or return to 26.5.6 Plot Sheet Layout.

26.5.7 Boxes

The Boxes Area on the drainage long section plot is used to label points on the drainage strings with the following information: chainages, road centre line chainages, natural surface heights, finished surface heights, before pit and after pit invert levels, before pit and after pit hgl values, and before pit and after pit depths to inverts.

The default order of the boxes is as shown in the diagram, below.



The *chainage box* is hardwired to always appear first (at the bottom) in the order of boxes. If, you wish to change the order of any of the other boxes, or wish to omit any of the other boxes, you can do so by specifying the following box order parameters.

Panel Field	Parameter name	Type	Pop-Up
Box #	**box_n_set**	**input**	

order (from 1 to 6) of the nth box type.

Panel Field	Parameter name	Type	Pop-Up
List #	**box_n**	**choice box**	**road centre line chainages** **natural surface heights** **finished surface heights** **invert levels** **hgl values** **depths to inverts**

box type for the nth set of box order parameters.

See

26.5.7.1 Boxes - Title Area
26.5.7.2 Boxes - Values Area
26.5.7.3 Boxes - Chainage
26.5.7.4 Boxes - Road Centre Line Chainage
26.5.7.5 Boxes - Natural Surface
26.5.7.6 Boxes - Finished Surface
26.5.7.7 Boxes - Invert Levels
26.5.7.8 Boxes - HGL
26.5.7.9 Boxes - Depths
26.5.7.10 Boxes - Outer linework

Or return to 26.4 Long Plot PPF Editor.

26.5.7.1 Boxes - Title Area

Section: Boxes - Title area parameters

Panel Field	Parameter name	Type	Pop-Up
Box width (mm)	box_width	input	
width of the box title area.			
Box height (mm)	box_height	input	
default height of each box.			
Colour of box linework	box_colour	colour box	available colours
colour of box line work.			
Size of titles (mm)	title_box_text_size	input	
default text size of box titles.			
Colour of titles	title_box_text_colour	colour box	available colours
default text colour of box titles.			
Suppress title left linework	suppress_title_box_line_left	tick box	
Suppress title top linework	suppress_title_box_line_top	tick box	
Draw linestyle legend beneath TIN & HGL titles	draw_linestyle_legends	tick box	
Draw linestyle legend beneath IL title	draw_linestyle_legend_il	tick box	

Continue to the next section 26.5.7.2 Boxes - Values Area or return to 26.5.7 Boxes.

26.5.7.2 Boxes - Values Area

Section: Boxes - Value area parameters

Panel Field	Parameter name	Type	Pop-Up
Size of values (mm)	box_text_size	input	
default text size of box values.			
Colour of values	box_text_colour	colour box	available colours
default text colour of box values.			
Justification mode	box_text_left_justify	choice box	top justify values bottom justify values
whether to justify the (vertical) value text with the top or the bottom of the box.			
Text side mode	box_text_side	choice box	left right centre
whether to place the (vertical) value text to the left of, to the right of, or on the chainage uprights.			
Decimal places of values	**number_of_decimals**	number box	
default number of decimal places for box values.			

*If > 0, trailing zeros are **removed** after the decimal point.*
If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Suppress value top linework	suppress_value_box_line_top	tick box

Continue to the next section 26.5.7.3 Boxes - Chainage or return to 26.5.7 Boxes.

26.5.7.3 Boxes - Chainage

Panel Field **Parameter name** **Type** **Pop-Up**

Box height (mm) **chainage_box_size** **input**

height of chainage box (only required if different from default box_height*).*

Section: Title parameters

Line 1 **plot_title_chainage_name** **input**

first line of text for chainage box title.

Line 2 **plot_title_chainage_name_2** **input**

optional second line of text for chainage box title.

Colour **chainage_title_colour** **colour box** **available colours**

text colour of chainage box title (required only if different from default title_box_text_colour*).*

Text size (mm) **chainage_title_text_sizeinput**

text size of chainage box title (required only if different from default title_box_text_size*).*

Textstyle **chainage_title_textstylefont box** **available fonts**

font of chainage box title (required only if different from default global_textstyle*).*

Suppress title bottom linework **suppress_title_box_line_ch** **tick box**

Section: Value parameters

Colour **chainage_text_colour** **colour box** **available colours**

text colour of chainage box values (required only if different from default box_text_colour*).*

Text size (mm) **chainage_text_size** **input**

text size of chainage box values (required only if different from default box_text_size*).*

Textstyle **chainage_textstyle** **font box** **available fonts**

font of chainage box values (required only if different from default global_textstyle*).*

Decimal places **chainage_decimals** **input**

decimal places for chainage box values (required only if different from default number_of_decimals*)*

If > 0, trailing zeros are ***removed*** *after the decimal point.*
If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Suppress value bottom linework **suppress_value_box_line_ch** **tick box**

Continue to the next section 26.5.7.4 Boxes - Road Centre Line Chainage or return to 26.5.7 Boxes.

26.5.7.4 Boxes - Road Centre Line Chainage

Panel Field **Parameter name** **Type** **Pop-Up**

Include centre line chainage box
draw_centre_chainage **tick box**

whether to include the road centreline chainage box in the boxes area.

Box height (mm) **centre_chainage_box_size**
input

height of road centreline chainage box (only required if different from default box_height*).*

Section: Title parameters

Line 1 **plot_title_centre_chainage_name**
input

first line of text for road centreline chainage box title.

Line 2 **plot_title_centre_chainage_name_2**
input

optional second line of text for road centreline chainage box title.

Colour **chainage_title_colour_clcolour box available colours**

colour of road centreline chainage box title (required only if different from default title_box_text_colour*).*

Text size (mm) **chainage_title_text_size_cl**
input

text size of road centreline chainage box title (required only if different from default title_box_text_size*).*

Textstyle **chainage_title_textstyle_cl**
font box **available fonts**

font of road centreline chainage box title (required only if different from default global_textstyle*).*

Suppress title bottom linework **suppress_title_box_line_rc** **tick box**

Section: Value parameters

Colour **chainage_text_colour_clcolour box available colours**

text colour of road centreline chainage box values (required only if different from default box_text_colour*).*

Text size (mm) **chainage_text_size_cl** **input**

text size of road centreline chainage box values (required only if different from default box_text_size*).*

Textstyle **chainage_textstyle_cl** **font box** **available fonts**

font of road centreline chainage box values (required only if different from default global_textstyle*).*

Decimal places **chainage_cl_decimals** **input**

decimal places for road centreline chainage box values (required only if different from default number_of_decimals*).*

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Suppress value bottom linework **suppress_value_box_line_rc** **tick box**

Continue to the next section 26.5.7.5 Boxes - Natural Surface or return to 26.5.7 Boxes.

26.5.7.5 Boxes - Natural Surface

Panel Field	Parameter name	Type	Pop-Up
Draw text mode	draw_ns_text	choice box	don't include ns box **include ns box**

whether to include the natural surface box in the boxes area.

Box height (mm)	**surface_box_size**	**input**	

height of natural surface box (only required if different from default box_height*).*

Section: Title parameters

Line 1 **plot_title_surface_name** **input**

first line of text for natural surface box title.

Line 2 **plot_title_surface_name_2** **input**

optional second line of text for natural surface box title.

Colour	**ns_title_colour**	**colour box**	**available colours**

text colour of natural surface box title (required only if different from default title_box_text_colour*).*

Text size (mm)	**ns_title_text_size**	**input**	

text size of natural surface box title (required only if different from default title_box_text_size*).*

Textstyle	**ns_title_textstyle**	**font box**	**available fonts**

font of natural surface box title (required only if different from default global_textstyle*).*

Suppress title bottom linework **suppress_title_box_line_ns** **tick box**

Section: Value parameters

Colour	**ns_text_colour**	**colour box**	**available colours**

text colour of natural surface box values (required only if different from default box_text_colour*).*

Text size (mm)	**ns_text_size**	**input**	

text size of natural surface box values (required only if different from default box_text_size*).*

Textstyle	**ns_textstyle**	**font box**	**available fonts**

font of natural surface box values (required only if different from default global_textstyle*).*

Decimal places	**ns_decimals**	**input**	

decimal places for natural surface box values (required only if different from default number_of_decimals*).*

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Suppress value bottom linework **suppress_value_box_line_ns** **tick box**

Continue to the next section 26.5.7.6 Boxes - Finished Surface or return to 26.5.7 Boxes.

26.5.7.6 Boxes - Finished Surface

Panel Field	Parameter name	Type	Pop-Up
Draw text mode	draw_fs_text	choice box	don't include fs box **include fs box**

whether to include the finished surface box in the boxes area.

FS tin/Top of MH mode	draw_fs_mode	choice box	height from fs tin **height from top of MH**

whether the finished surface height values come from the finished surface tin, or from the tops of the maintenance holes.

Box height (mm) **f_surface_box_size** **input**

height of finished surface box (only required if different from default box_height*).*

Section: Title parameters

Line 1 **plot_title_finished_name** **input**

first line of text for finished surface box title.

Line 2 **plot_title_finished_name_2** **input**

optional second line of text for finished surface box title.

Colour **fs_title_colour** **colour box** **available colours**

text colour of finished surface box title (required only if different from default title_box_text_colour*).*

Text size (mm) **fs_title_text_size** **input**

text size of finished surface box title (required only if different from default title_box_text_size*).*

Textstyle **fs_title_textstyle** **font box** **available fonts**

font of finished surface box title (required only if different from default global_textstyle*).*

Suppress title bottom linework **suppress_title_box_line_fs** **tick box**

Section: Value parameters

Colour **fs_text_colour** **colour box** **available colours**

text colour of finished surface box values (required only if different from default box_text_colour*).*

Text size (mm) **fs_text_size** **input**

text size of finished surface box values (required only if different from default box_text_size*).*

Textstyle **fs_textstyle** **font box** **available fonts**

font of finished surface box values (required only if different from default global_textstyle*).*

Decimal places **fs_decimals** **input**

decimal places for finished surface box values (required only if different from default number_of_decimals*).*

*If > 0, trailing zeros are **removed** after the decimal point.*
If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Suppress value bottom linework **suppress_value_box_line_fs** **tick box**

Continue to the next section 26.5.7.7 Boxes - Invert Levels or return to 26.5.7 Boxes.

26.5.7.7 Boxes - Invert Levels

Panel Field **Parameter name** **Type** **Pop-Up**

Box height (mm) **invert_height_box_size** **input**

height of invert level box (only required if different from default box_height*).*

Section: Title parameters

Line 1 **plot_title_invert_name** **input**

first line of text for invert level box title.

Line 2 **plot_title_invert_name_2input**

optional second line of text for invert level box title.

Colour **il_title_colour** **colour box** **available colours**

text colour of invert level box title (required only if different from default title_box_text_colour*).*

Text size (mm) **il_title_text_size** **input**

text size of invert level box title (required only if different from default title_box_text_size*).*

Textstyle **il_title_textstyle** **font box** **available fonts**

font of invert level box title (required only if different from default global_textstyle*).*

Suppress title bottom linework **suppress_title_box_line_il** **tick box**

Section: Value parameters

Colour **il_text_colour** **colour box** **available colours**

text colour of invert level box values (required only if different from default box_text_colour*).*

Text size (mm) **il_text_size** **input**

text size of invert level box values (required only if different from default box_text_size*).*

Textstyle **il_textstyle** **font box** **available fonts**

font of invert level box values (required only if different from default global_textstyle*).*

Decimal places **il_decimals** **input**

decimal places for invert level box values (required only if different from default number_of_decimals*).*

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Suppress value bottom linework **suppress_value_box_line_il** **tick box**

Continue to the next section 26.5.7.8 Boxes - HGL or return to 26.5.7 Boxes.

26.5.7.8 Boxes - HGL

Panel Field	Parameter name	Type	Pop-Up
Draw HGL box	**draw_hgl_value**	**tick box**	

whether to include the HGL box in the boxes area.

Box height (mm)	**hgl_box_size**	**input**	

height of HGL box (only required if different from default box_height*).*

Section: Title parameters

Line 1	**plot_title_hgl_name**	**input**	

first line of text for HGL box title.

Line 2	**plot_title_hgl_name_2**	**input**	

optional second line of text for HGL box title.

Colour	**hgl_title_colour**	**colour box**	**available colours**

text colour of HGL box title (required only if different from default title_box_text_colour*).*

Text size (mm)	**hgl_title_text_size**	**input**	

text size of HGL box title (required only if different from default title_box_text_size*).*

Textstyle	**hgl_title_textstyle**	**font box**	**available fonts**

font of HGL box title (required only if different from default global_textstyle*).*

Suppress title bottom linework	**suppress_title_box_line_hg**	**tick box**	

Section: Value parameters

Colour	**hgl_text_colour**	**colour box**	**available colours**

text colour of HGL box values (required only if different from default box_text_colour*).*

Text size (mm)	**hgl_text_size**	**input**	

text size of HGL box values (required only if different from default box_text_size*).*

Textstyle	**hgl_textstyle**	**font box**	**available fonts**

font of HGL box values (required only if different from default global_textstyle*).*

Decimal places	**hgl_decimals**	**input**	

decimal places for HGL box values (required only if different from default number_of_decimals*).*

If > 0, trailing zeros are ***removed*** *after the decimal point.*
If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Suppress value bottom linework	**suppress_value_box_line_hg**	**tick box**	

Continue to the next section 26.5.7.9 Boxes - Depths or return to 26.5.7 Boxes.

26.5.7.9 Boxes - Depths

Panel Field	Parameter name	Type	Pop-Up
Depth box mode	depth_mode	choice box	don't include depths box **depths from fs tin** **depths from top of MH** **depths from ns tin**

whether to include the depths (to inverts) box in the boxes area, and if so, whether the depths are measured from the finished surface tin, the tops of the maintenance holes, or the natural surface tin.

Box height (mm) **invert_depth_box_size** **input**

height of depths box (only required if different from default box_height*).*

Section: Title parameters

Line 1 **plot_title_depth_name** **input**

first line of text for depths box title.

Line 2 **plot_title_depth_name_2input**

optional second line of text for depths box title.

Colour **depth_title_colour** **colour box** **available colours**

text colour of depths box title (required only if different from default title_box_text_colour*).*

Text size (mm) **depth_title_text_size** **input**

text size of depths box title (required only if different from default title_box_text_size*).*

Textstyle **depth_title_textstyle** **font box** **available fonts**

font of depths box title (required only if different from default global_textstyle*).*

Suppress title bottom linework **suppress_title_box_line_dp** **tick box**

Section: Value parameters

Colour **depth_text_colour** **colour box** **available colours**

text colour of depths box values (required only if different from default box_text_colour*).*

Text size (mm) **depth_text_size** **input**

text size of depths box values (required only if different from default box_text_size*).*

Textstyle **depth_textstyle** **font box** **available fonts**

font of depths box values (required only if different from default global_textstyle*).*

Decimal places **depth_decimals** **input**

decimal places for depths box values (required only if different from default number_of_decimals*).*

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Suppress value bottom linework **suppress_value_box_line_dp** **tick box**

Continue to the next section 26.5.7.10 Boxes - Outer linework or return to 26.5.7 Boxes.

26.5.7.10 Boxes - Outer linework

Section: Boxes - Outer linework

Panel Field	Parameter name	Type	Pop-Up
Title area box mode	draw_box_mode	choice box	do not draw title area box **draw title area box, no lines** **draw title area box with lines**

whether to draw the line work around the titles area, and if so, whether to draw lines separating each title as well.

Panel Field	Parameter name	Type	Pop-Up
Chainage box mode	chainage_box_mode	choice box	don’t draw the chainage box **draw the chainage box**

whether to draw the line work around the chainage box.

Please continue to the next section 26.5.8 Chainages/Uprights.

26.5.8 Chainages/Uprights

Uprights, or leader lines, can be drawn through the points on the drainage strings, anywhere from the top area to the bottom of the boxes area.

Section: Upright parameters

Panel Field	Parameter name	Type	Pop-Up
Top of uprights mode	uprights_top_mode	choice box	stop at top of pit **go to top area**

determines how high on the plot, the uprights are drawn.

Panel Field	Parameter name	Type	Pop-Up
Bottom of uprights mode	uprights_bottom_mode	choice box	stop at top of boxes **go to bottom of boxes**

determines how low on the plot, the uprights are drawn.

Leave gap in uprights through pit uprights_gap_mode tick box

Panel Field	Parameter name	Type	Pop-Up
Colour of uprights to pits	**manhole_line_colour**	**colour box**	**available colours**

colour of the uprights.

Section: Chainage parameters

Panel Field	Parameter name	Type	Pop-Up
Chainage values at uprights			
	chainage_mode	**choice box**	**pipe length chainage** **running chainage** **both pipe length and**

running

whether the chainage values represent the individual pipe lengths, the running chainages, or both.

Panel Field	Parameter name	Type	Pop-Up
Pipe length label position	centre_pipe_length	choice box	0 at start pit, length at end pit **centre of pipe length**

determines how pipe length chainages are displayed. ***Note****: if the* centre of pipe length *option is chosen, the pipe length chainage text is drawn horizontally, rather than vertically.*

See

26.5.8.1 Chainages/Uprights - Services
26.5.8.2 Chainages/Uprights - Finished Surface/Top of MH Values
26.5.8.3 Chainages/Uprights - Staggering

Or return to 26.4 Long Plot PPF Editor.

26.5.8.1 Chainages/Uprights - Services

Any services crossing the drainage strings (defined by specifying *corridor service models* in the *Corridors* section) can be automatically labelled with an upright, chainage and invert level labels (of the drainage string, placed within the appropriate boxes in the Boxes area), and a name label (that includes the crossing service string name, its diameter, and its invert level at the point of crossing).

Note: crossing services may also be labelled (independently) by utilising the parameters in the *Cuts* section.

Panel Field **Parameter name** **Type** **Pop-Up**

Service chainage values at uprights
service_chainage_mode **choice box** **pipe length chainage**
running chainage
both pipe length and running

whether the crossing service chainage values represent the chainages from the previous pits, the running chainages, or both.

Draw uprights to crossing services tick box

Colour of uprights to crossing services
service_line_colour **colour box** **available colours**

colour of the crossing service uprights.

Section: Service chainage text parameters

Panel Field **Parameter name** **Type** **Pop-Up**

Colour **service_ch_text_colour** **colour box** **available colours**

text colour of crossing service chainage box values (required only if different from default box_text_colour*).*

Text size (mm) **service_ch_text_size** **input**

text size of crossing service chainage box values (required only if different from default box_text_size*).*

Textstyle **service_ch_textstyle** **font box** **available fonts**

font of crossing service chainage box values (required only if different from default global_textstyle*).*

Decimal places **service_ch_decimals** **input**

decimal places for crossing service chainage box values (required only if different from default number_of_decimals*).*

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Section: Service invert level text parameters

Panel Field **Parameter name** **Type** **Pop-Up**

Colour **service_il_text_colour** **colour box** **available colours**

text colour of crossing service invert level box values (required only if different from default box_text_colour*).*

Text size (mm) **service_il_text_size** **input**

text size of crossing service invert level box values (required only if different from default box_text_size*).*

Textstyle **service_il_textstyle** **font box** **available fonts**

font of crossing service invert level box values (required only if different from default global_textstyle*).*

Decimal places **service_il_decimals** **input**

decimal places for crossing service invert level box values (required only if different from default number_of_decimals*).*

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Section: Service label text parameters

Panel Field	Parameter name	Type	Pop-Up
Show name	**service_name_show_name**	**tick box**	
Show size	**service_name_show_size**	**tick box**	
Show IL	**service_name_show_il**	**tick box**	
Show clearance	**service_name_show_clr**	**tick box**	
X adjustment (mm)	service_name_x	input	
Y adjustment (mm)	service_name_y	input	
Stagger	service_name_stagger	tick box	
Y position	service_name_y_mode	choice box	below crossing point
			above crossing point
			below top staggers
			above bottom staggers

Distance of label below crossing point (mm)
service_name_y **input**

distance (in millimetres) of crossing service name labels below crossing points.

Colour **service_name_text_colour**
colour box **available colours**

text colour of crossing service name labels (required only if different from default box_text_colour*).*

Text size (mm) **service_name_text_sizeinput**

text size of crossing service name labels (required only if different from default box_text_size*).*

Textstyle **service_name_textstyle** **font box** **available fonts**

font of crossing service name labels (required only if different from default global_textstyle*).*

Decimal places **service_name_decimalsinput**

decimal places for the crossing service invert levels that make up part of the crossing service name labels (required only if different from default number_of_decimals*).*

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Continue to the next section 26.5.8.2 Chainages/Uprights - Finished Surface/Top of MH Values or return to 26.5.8 Chainages/Uprights.

26.5.8.2 Chainages/Uprights - Finished Surface/Top of MH Values

In addition to the finished surface level labels within the Boxes area, more detailed vertical labels representing finished surface levels at the pits, can be added to accompany the pit uprights within the Graph area.

Panel Field	Parameter name	Type	Pop-Up
Finished surface draw mode			
	draw_fs_vertical	**choice box**	**do not draw fs values** **always draw fs values** **only draw if different**

whether to label the finished surface levels within the Graph area. Additional control is provided to only draw the labels if the finished surface level is different from the level at the top of the maintenance hole.

Panel Field	Parameter name	Type	Pop-Up
Finished surface value mode			
	draw_fs_vertical_mode	**choice box**	**value from fs tin** **value from top of mh**

whether the finished surface level label values come from the fs tin or the top of the maintenance holes.

Pre text fs_vertical_pre_text input

label text before the finished surface level.

Post text fs_vertical_post_text input

label text after the finished surface level.

Decimal places **fs_vertical_decimals** **input**

decimal places for finished surface level values (required only if different from default number_of_decimals*).*

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Colour **fs_vertical_colour** **colour box** **available colours**

text colour of finished surface level labels (required only if different from default box_text_colour*).*

Size (mm) **fs_vertical_size** **input**

text size of finished surface level labels (required only if different from default box_text_size*).*

Textstyle **fs_vertical_textstyle** **font box** **available fonts**

font of finished surface level labels (required only if different from default global_textstyle*).*

X adjustment (mm) fs_vertical_x input

horizontal adjustment to position of finished surface level labels.

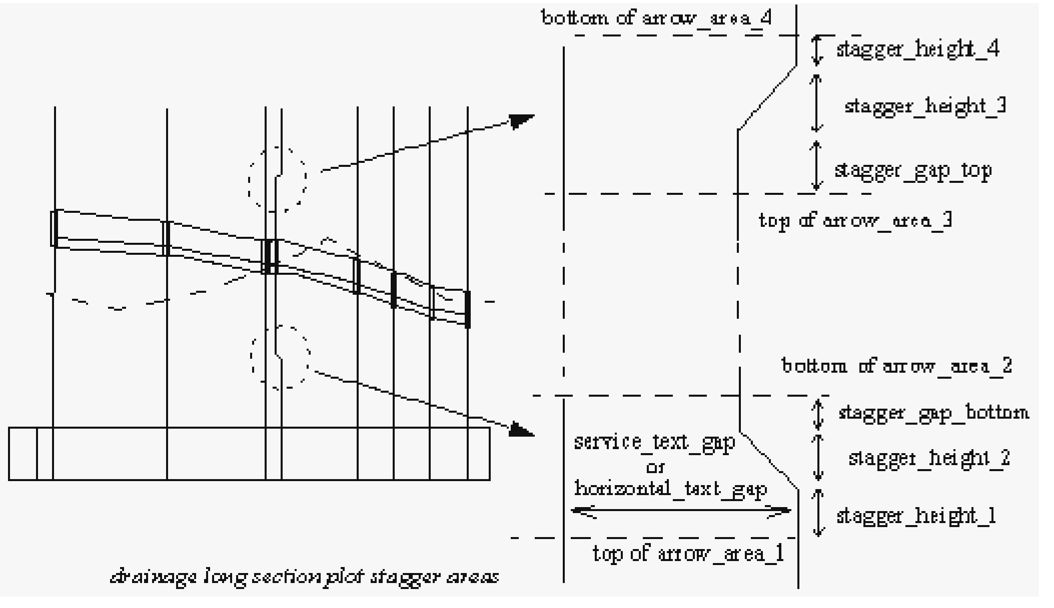
Y adjustment (mm) fs_vertical_y input

vertical adjustment to position of finished surface level labels.

Continue to the next section 26.5.8.3 Chainages/Uprights - Staggering or return to 26.5.8 Chainages/Uprights.

26.5.8.3 Chainages/Uprights - Staggering

Because the pit and service uprights are sometimes too close together to fit all the desired information in the Boxes and Arrows areas, the uprights can be staggered (widened) in such a way as to provide more space in these areas, below and above the Graph area, whilst still maintaining the correct upright spacing within the Graph area itself. Specifically, the staggering of the uprights occurs below the Graph area between Arrow areas 1 and 2, and above the Graph area between Arrow areas 3 and 4. As such, it is recommended that Arrow areas 2 and 3 be used only for information requiring minimal space.



Section: Stagger area below graph area parameters

Panel Field | **Parameter name** | **Type**

Top of arrow area 1 to start of the staggers (mm)
stagger_height_1 **input**

refer to stagger_height_1 *on above diagram.*

Distance over which stagger occurs (mm)
stagger_height_2 **input**

refer to stagger_height_2 *on above diagram.*

End of the staggers to bottom of arrow area 2 (mm)
stagger_gap_bottom **input**

refer to stagger_gap_bottom *on above diagram.*

Section: Stagger area above graph area parameters

Panel Field | **Parameter name** | **Type**

Top of arrow area 3 to start of the staggers (mm)
stagger_gap_top **input**

refer to stagger_gap_top *on above diagram.*

Distance over which stagger occurs (mm)
stagger_height_3 **input**

refer to stagger_height_3 *on above diagram.*

End of the staggers to bottom of arrow area 4 (mm)
stagger_height_4 **input**

refer to stagger_height_4 *on above diagram.*

Section: Stagger offset parameters

Panel Field	Parameter name	Type
Minimum distance between pit uprights (mm)	**horizontal_text_gap**	**input**

refer to horizontal_text_gap *on above diagram.*

Panel Field	Parameter name	Type
Minimum distance before first service upright (mm)	**service_text_gap_before**	**input**
Minimum distance after service uprights (mm)	**service_text_gap**	**input**

refer to service_text_gap *on above diagram.*

Please continue to the next section 26.5.9 Datum Area.

26.5.9 Datum Area

The Datum area is where each long section plot's datum value is drawn. The datum line for each long section plot is defined at the bottom of Arrow area 1.

Panel Field	Parameter name	Type
Below datum area gap (mm)	**datum_gap**	**input**

distance from the top of the Boxes area to the bottom of Arrow area 1.

Panel Field	Parameter name	Type
Extend datum line into title area	**datum_extend**	**tick box**

See

26.5.9.1 Datum Area - Name
26.5.9.2 Datum Area - Values

Or return to 26.4 Long Plot PPF Editor.

26.5.9.1 Datum Area - Name

The datum name (or title) is drawn at the datum line, above the Boxes title area.

Panel Field	Parameter name	Type	Pop-Up
Datum name	datum_name	input	

title for the datum area.

Panel Field	Parameter name	Type	Pop-Up
X offset (mm)	datum_title_x	input	

horizontal adjustment to position of datum name.

Panel Field	Parameter name	Type	Pop-Up
Y offset (mm)	datum_title_y	input	

vertical adjustment to position of datum name.

Panel Field	Parameter name	Type	Pop-Up
Textstyle	**datum_title_textstyle**	**font box**	**available fonts**

font of datum name (required only if different from default global_textstyle*).*

Panel Field	Parameter name	Type	Pop-Up
Text size (mm)	**datum_title_text_size**	**input**	

text size of datum name (required only if different from default box_text_size*).*

Panel Field	Parameter name	Type	Pop-Up
Colour	**datum_title_colour**	**colour box**	**available colours**

text colour of datum name (required only if different from default box_text_colour*).*

Continue to the next section 26.5.9.2 Datum Area - Values or return to 26.5.9 Datum Area.

26.5.9.2 Datum Area - Values

The datum values are drawn at the datum lines, above the Box values areas.

Panel Field | **Parameter name** | **Type** | **Pop-Up**

Decimal places datum_decimals input

number of decimal places for datum values.

*If > 0, trailing zeros are **removed** after the decimal point.*
If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

X offset (mm) datum_x input

horizontal adjustment to position of datum values.

Y offset (mm) datum_y input

vertical adjustment to position of datum values.

Textstyle datum_textstyle font box available fonts

font of datum values (required only if different from default global_textstyle*).*

Text size (mm) datum_text_size input

text size of datum values (required only if different from default box_text_size*).*

Colour datum_text_colour colour box available colours

text colour of datum values (required only if different from default box_text_colour*).*

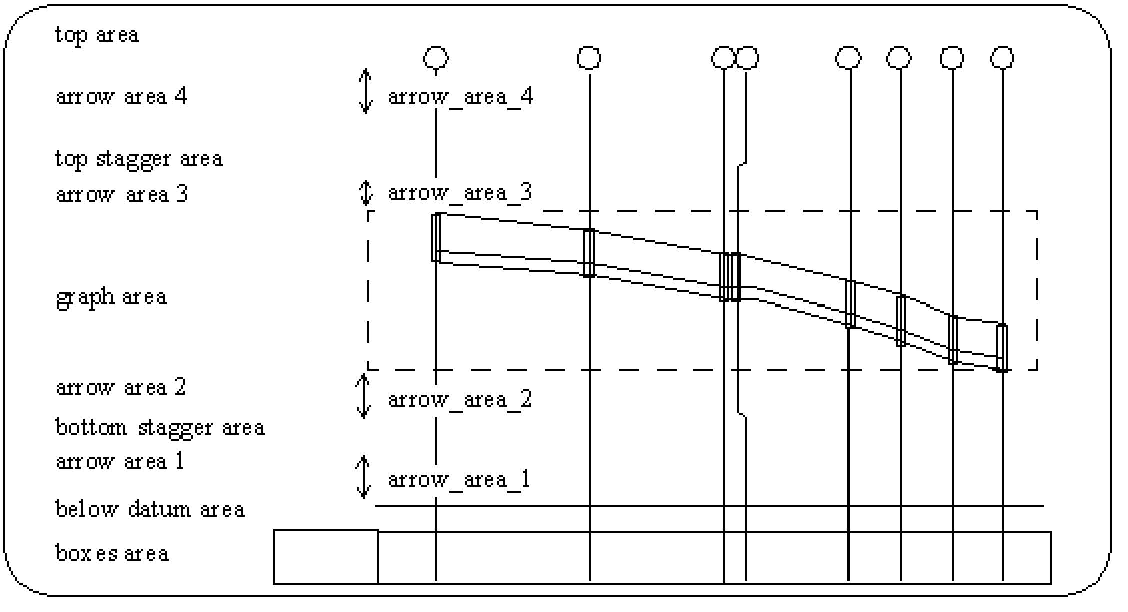
Please continue to the next section 26.5.10 Arrows.

26.5.10 Arrows

The Arrow areas on the drainage long section plot, are used to label segments (individual pipes) on the drainage strings with the following information: grades, diameters and types, flow velocities, flow volumes, drainage line names, and any user-defined pipe attribute values.

There are four separate Arrow areas on the plot, any of which can be used to label the above information. From plot bottom to top, the Arrow areas are:

Arrow area 1: staggered arrows below the Graph area,
Arrow area 2: unstaggered arrows below the Graph area,
Arrow area 3: unstaggered arrows above the Graph area,
Arrow area 4: staggered arrows above the Graph area.



Panel Field **Parameter name** **Type**

Height of arrow area 1 (mm)
arrow_area_1 **input**

refer to arrow_area_1 *on the above diagram.*

Height of arrow area 2 (mm)
arrow_area_2 **input**

refer to arrow_area_2 *on the above diagram.*

Height of arrow area 3 (mm)
arrow_area_3 **input**

refer to arrow_area_3 *on the above diagram.*

Height of arrow area 4 (mm)
arrow_area_4 **input**

refer to arrow_area_4 *on the above diagram.*

See

26.5.10.1 Arrows - Grades
26.5.10.2 Arrows - Diameters
26.5.10.3 Arrows - Velocities
26.5.10.4 Arrows - Flow

Or return to 26.4 Long Plot PPF Editor.

26.5.10.1 Arrows - Grades

Panel Field	Parameter name	Type	Pop-Up
Draw pipe grade	draw_pipe_grade	check box	

whether to draw the pipe grades in one of the Arrow areas.

Arrow area for the arrow	pipe_grade_arrow_area	choice box	Area 1 **Area 2** **Area 3** **Area 4**

the desired Arrow area for the pipe grade information.

Y offset (mm)	pipe_grade_y	input	

vertical offset of pipe grade information, measured upwards from the bottom of the selected Arrow area.

Grade mode	percentage_grade	choice box	1 in grade **% grade**

determines the format of the pipe grade information.

See

26.5.10.1.1 Arrows - Grades - Title Text
26.5.10.1.2 Arrows - Grades - Arrow type
26.5.10.1.3 Arrows - Grades - Arrow Text

Or return to 26.5.10 Arrows.

26.5.10.1.1 Arrows - Grades - Title Text

Panel Field	Parameter name	Type	Pop-Up
Text	pipe_grade_title	input	

title for the pipe grade information.

Textstyle	**pipe_grade_title_textstyle**	**font box**	**available fonts**

font of pipe grade title.

Size (mm)	**pipe_grade_title_text_size**	**input**	

text size of pipe grade title.

Colour	**pipe_grade_title_text_colour**	**colour box**	**available colours**

text colour of pipe grade title.

Offset (mm)	pipe_grade_title_offset	input	

vertical adjustment to position of pipe grade title.

X offset (mm)	pipe_grade_title_x	input	

horizontal adjustment to position of pipe grade title.

Draw title bottom linework	pipe_grade_title_line	tick box	
Height of title left linework (mm)	pipe_grade_title_height	input	

Continue to the next section 26.5.10.1.2 Arrows - Grades - Arrow type or return to 26.5.10 Arrows.

26.5.10.1.2 Arrows - Grades - Arrow type

Panel Field	Parameter name	Type	Pop-Up
Arrow type	pipe_grade_arrow_mode	choice box	available arrow types

arrow type (0 to 8) for the pipe grade information.

Colour	**pipe_grade_arrow_colour**	**colour box**	**available colours**

arrow colour for pipe grade information.

Size (mm)	**pipe_grade_arrow_size**	**input**	

arrow size for pipe grade information.

Arrow gap mode	pipe_grade_arrow_gap	choice box	no gap in arrow **leave gap in arrow for text**

whether to put text gaps in the arrows, for pipe grade information.

Continue to the next section 26.5.10.1.3 Arrows - Grades - Arrow Text or return to 26.5.10 Arrows.

26.5.10.1.3 Arrows - Grades - Arrow Text

Panel Field	Parameter name	Type	Pop-Up
Pre text	pipe_grade_arrow_pre_text	text box	

label text before the pipe grade values.

Post text	pipe_grade_arrow_post_text	**text box**	

label text after the pipe grade values.

Decimal places	pipe_grade_arrow_decimals	integer box	

decimal places for pipe grade values.

*If > 0, trailing zeros are **removed** after the decimal point.*
If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Colour	pipe_grade_arrow_text_colour	colour box	available colours

text colour for pipe grade values.

Size (mm)	pipe_grade_arrow_text_size	input	

text size for pipe grade values.

Textstyle	pipe_grade_arrow_textstyle	font box	available fonts

font of pipe grade values.

Offset (mm)	pipe_grade_arrow_text_offset	real box	

vertical adjustment to position of pipe grade values.

Continue to the next section 26.5.10.2 Arrows - Diameters or return to 26.5.10 Arrows.

26.5.10.2 Arrows - Diameters

Panel Field	Parameter name	Type	Pop-Up
Pipe diameter scale factor	**pipe_diameter_scale_factor**	**input**	

multiplier of pipe diameter values.

Panel Field	Parameter name	Type	Pop-Up
Draw pipe diameter	draw_pipe_diameter	choice box	don't draw pipe diameter **draw pipe diameter**

whether to draw the pipe diameters in one of the Arrow areas.

Panel Field	Parameter name	Type	Pop-Up
Arrow area for the arrow	pipe_diameter_arrow_area	**choice box**	**Area 1** **Area 2** **Area 3** **Area 4**

the desired Arrow area for the pipe diameter information.

Panel Field	Parameter name	Type	Pop-Up
Y offset (mm)	pipe_diameter_y	input	

vertical offset of pipe diameter information, measured upwards from the bottom of the selected Arrow area.

See

26.5.10.2.1 Arrows - Diameters - Title Text
26.5.10.2.2 Arrows - Diameters - Arrow type
26.5.10.2.3 Arrows - Diameters - Arrow Text

Or return to 26.5.10 Arrows.

26.5.10.2.1 Arrows - Diameters - Title Text

Panel Field	Parameter name	Type	Pop-Up
Text	pipe_diameter_title	input	

title for the pipe diameter information.

Textstyle	**pipe_diameter_title_textstyle**	**font box**	**available fonts**

font of pipe diameter title.

Size (mm)	**pipe_diameter_title_text_size**	**input**	

text size of pipe diameter title.

Colour	**pipe_diameter_title_text_colour**	**colour box**	**available colours**

text colour of pipe diameter title.

Offset (mm)	pipe_diameter_title_offset	**input**	

vertical adjustment to position of pipe diameter title.

X offset (mm)	pipe_diameter_title_x	input	

horizontal adjustment to position of pipe diameter title.

Draw title bottom linework	pipe_diameter_title_line	tick box	
Height of title left linework (mm)	pipe_diameter_title_height	input	

Continue to the next section 26.5.10.2.2 Arrows - Diameters - Arrow type or return to 26.5.10 Arrows.

26.5.10.2.2 Arrows - Diameters - Arrow type

Panel Field	Parameter name	Type	Pop-Up
Arrow type	pipe_diameter_arrow_mode	**choice box**	**available arrow types**

arrow type (0 to 8) for the pipe diameter information.

Colour	**pipe_diameter_arrow_colour**	**colour box**	**available colours**

arrow colour for pipe diameter information.

Size (mm)	**pipe_diameter_arrow_size**	**input**	

arrow size for pipe diameter information.

Arrow gap mode	pipe_diameter_arrow_gap	**choice box**	**no gap in arrow** **leave gap in arrow for text**

whether to put text gaps in the arrows, for pipe diameter information.

Continue to the next section 26.5.10.1.3 Arrows - Grades - Arrow Text or return to 26.5.10 Arrows.

26.5.10.2.3 Arrows - Diameters - Arrow Text

Panel Field	Parameter name	Type	Pop-Up
Pre text	pipe_diameter_arrow_pre_text	**input**	

label text before the pipe diameter values.

Panel Field	Parameter name	Type	Pop-Up
Post text	pipe_diameter_arrow_post_text	**input**	

label text after the pipe diameter values.

Panel Field	Parameter name	Type	Pop-Up
Pipe type mode	**pipe_type_mode**	**choice box**	**don't include the pipe type** **include the pipe type**

whether to include the pipe type after the post text in the pipe diameter labels.

Panel Field	Parameter name	Type	Pop-Up
Units factor	**pipe_diameter_arrow_factor**	**input**	
Decimal places	**pipe_diameter_arrow_decimals**	**input**	
Colour	**pipe_diameter_arrow_text_colour**	**colour box**	**available colours**

text colour for pipe diameter values.

Panel Field	Parameter name	Type	Pop-Up
Size (mm)	**pipe_diameter_arrow_text_size**	**input**	

text size for pipe diameter values.

Panel Field	Parameter name	Type	Pop-Up
Textstyle	**pipe_diameter_arrow_textstyle**	**font box**	**available fonts**

font of pipe diameter values.

Panel Field	Parameter name	Type	Pop-Up
Offset (mm)	pipe_diameter_arrow_text_offset	**input**	

vertical adjustment to position of pipe diameter values.

Continue to the next section 26.5.10.3 Arrows - Velocities or return to 26.5.10 Arrows.

26.5.10.3 Arrows - Velocities

Panel Field	Parameter name	Type	Pop-Up
Draw pipe velocity	draw_pipe_velocity	check box	

whether to draw the flow velocities in one of the Arrow areas.

Panel Field	Parameter name	Type	Pop-Up
Arrow area for the arrow	pipe_velocity_arrow_area	choice box	Area 1 **Area 2** **Area 3** **Area 4**

the desired Arrow area for the flow velocity information.

Panel Field	Parameter name	Type	Pop-Up
Y offset (mm)	pipe_velocity_y	input	

vertical offset of flow velocity information, measured upwards from the bottom of the selected Arrow area.

See

26.5.10.3.1 Arrows - Velocities - Title Text
26.5.10.3.2 Arrows - Velocities - Arrow type
26.5.10.3.3 Arrows - Velocities - Arrow Text

Or return to 26.5.10 Arrows.

26.5.10.3.1 Arrows - Velocities - Title Text

Panel Field	Parameter name	Type	Pop-Up
Text	pipe_velocity_title	input	

title for the flow velocity information.

Textstyle	**pipe_velocity_title_textstyle**	**font box**	**available fonts**

font of flow velocity title.

Size (mm)	**pipe_velocity_title_text_size**	**input**	

text size of flow velocity title.

Colour	**pipe_velocity_title_text_colour**	**colour box**	**available colours**

text colour of flow velocity title.

Offset (mm)	pipe_velocity_title_offset	input	

vertical adjustment to position of flow velocity title.

X offset (mm)	pipe_velocity_title_x	input	

horizontal adjustment to position of flow velocity title.

Draw title bottom linework	pipe_velocity_title_line	tick box	
Height of title left linework (mm)	pipe_velocity_title_height	input	

Continue to the next section 26.5.10.3.2 Arrows - Velocities - Arrow type or return to 26.5.10 Arrows.

26.5.10.3.2 Arrows - Velocities - Arrow type

Panel Field	Parameter name	Type	Pop-Up
Arrow type	pipe_velocity_arrow_mode	**choice box**	**available arrow types**

arrow type (0 to 8) for the flow velocity information.

Colour	**pipe_velocity_arrow_colour**	**colour box**	**available colours**

arrow colour for flow velocity information.

Size (mm)	**pipe_velocity_arrow_sizeinput**		

arrow size for flow velocity information.

Arrow gap mode	pipe_velocity_arrow_gap	choice box	no gap in arrow **leave gap in arrow for text**

whether to put text gaps in the arrows, for flow velocity information.

Continue to the next section 26.5.10.3.3 Arrows - Velocities - Arrow Text or return to 26.5.10 Arrows.

26.5.10.3.3 Arrows - Velocities - Arrow Text

Panel Field	Parameter name	Type	Pop-Up
Pre text	pipe_velocity_arrow_pre_text	**input**	

label text before the flow velocity values.

Post text	pipe_velocity_arrow_post_text	**input**	

label text after the flow velocity values.

Units factor	**pipe_velocity_arrow_factor**	**input**	
Decimal places	**pipe_velocity_arrow_decimals**	**input**	

decimal places for flow velocity values.

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Colour	**pipe_velocity_arrow_text_colour**	**colour box**	**available colours**

text colour for flow velocity values.

Size (mm)	**pipe_velocity_arrow_text_size**	**input**	

text size for flow velocity values.

Textstyle	**pipe_velocity_arrow_textstyle**	**font box**	**available fonts**

font of flow velocity values.

Offset (mm)	pipe_velocity_arrow_text_offset	**input**	

vertical adjustment to position of flow velocity values.

Continue to the next section 26.5.10.4 Arrows - Flow or return to 26.5.10 Arrows.

26.5.10.4 Arrows - Flow

Panel Field	Parameter name	Type	Pop-Up
Draw pipe flow	draw_pipe_flow	check box	

whether to draw the flow volumes in one of the Arrow areas.

Arrow area for the arrow	pipe_flow_arrow_area	choice box	Area 1 **Area 2** **Area 3** **Area 4**

the desired Arrow area for the flow volume information.

Y offset (mm)	pipe_flow_y	input	

vertical offset of flow volume information, measured upwards from the bottom of the selected Arrow area.

See

26.5.10.4.1 Arrows - Flow - Title Text
26.5.10.4.2 Arrows - Flow - Arrow Type
26.5.10.4.3 Arrows - Flow - Arrow Text

Or return to 26.5.10 Arrows.

26.5.10.4.1 Arrows - Flow - Title Text

Panel Field	Parameter name	Type	Pop-Up
Text	pipe_flow_title	input	

title for the flow volume information.

Textstyle	**pipe_flow_title_textstyle**	**font box**	**available fonts**

font of flow volume title.

Size (mm)	**pipe_flow_title_text_size**	**input**	

text size of flow volume title.

Colour	**pipe_flow_title_text_colour**	**colour box**	**available colours**

text colour of flow volume title.

Offset (mm)	pipe_flow_title_offset	input	

vertical adjustment to position of flow volume title.

X offset (mm)	pipe_flow_title_x	input	

horizontal adjustment to position of flow volume title.

Draw title bottom linework	pipe_flow_title_line	tick box	
Height of title left linework (mm)	pipe_flow_title_height	input	

Continue to the next section 26.5.10.4.2 Arrows - Flow - Arrow Type or return to 26.5.10 Arrows.

26.5.10.4.2 Arrows - Flow - Arrow Type

Panel Field	Parameter name	Type	Pop-Up
Arrow type	pipe_flow_arrow_mode	choice box	available arrow types

arrow type (0 to 8) for the flow volume information.

Colour	**pipe_flow_arrow_colour**	**colour box**	**available colours**

arrow colour for flow volume information.

Size (mm)	**pipe_flow_arrow_size**	**input**	

arrow size for flow volume information.

Arrow gap mode	pipe_flow_arrow_gap	choice box	no gap in arrow **leave gap in arrow for text**

whether to put text gaps in the arrows, for flow volume information.

Continue to the next section 26.5.10.4.3 Arrows - Flow - Arrow Text or return to 26.5.10 Arrows.

26.5.10.4.3 Arrows - Flow - Arrow Text

Panel Field	Parameter name	Type	Pop-Up
Pre text	pipe_flow_arrow_pre_text	**input**	

label text before the flow volume values.

Post text	pipe_flow_arrow_post_text	**input**	

label text after the flow volume values.

Units factor	**pipe_flow_arrow_factor**	**input**	
Decimal places	**pipe_flow_arrow_decimals**	**input**	

decimal places for flow volume values.

*If > 0, trailing zeros are **removed** after the decimal point.*
If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Colour	**pipe_flow_arrow_text_colour**	**colour box**	**available colours**

text colour for flow volume values.

Size (mm)	**pipe_flow_arrow_text_size**	**input**	

text size for flow volume values.

Textstyle	**pipe_flow_arrow_textstyle**	**font box**	**available fonts**

font of flow volume values.

Offset (mm)	pipe_flow_arrow_text_offset	**input**	

vertical adjustment to position of flow volume values.

Continue to the next section 26.5.10.5 Arrows - Drainage Line Name or return to 26.5.10 Arrows.

26.5.10.5 Arrows - Drainage Line Name

Panel Field	Parameter name	Type	Pop-Up
Draw drainage line	draw_drainage_line	check box	

whether to draw the drainage line names in one of the Arrow areas.

Panel Field	Parameter name	Type	Pop-Up
Arrow area for the arrow	drainage_line_arrow_area	**choice box**	**Area 1** **Area 2** **Area 3** **Area 4**

the desired Arrow area for the drainage line name information.

Panel Field	Parameter name	Type	Pop-Up
Y offset (mm)	drainage_line_y	input	

vertical offset of drainage line name information, measured upwards from the bottom of the selected Arrow area.

See

26.5.10.5.1 Arrows - Drainage Line Name - Title Text
26.5.10.5.2 Arrows - Drainage Line Name - Arrow Type
26.5.10.5.3 Arrows - Drainage Line Name - Arrow Text

Or return to 26.5.10 Arrows.

26.5.10.5.1 Arrows - Drainage Line Name - Title Text

Panel Field	Parameter name	Type	Pop-Up
Text	drainage_line_title	input	
title for the drainage line name information.			
Textstyle	**drainage_line_title_textstyle**	**font box**	**available fonts**
font of drainage line name title.			
Size (mm)	**drainage_line_title_text_size**	**input**	
text size of drainage line name title.			
Colour	**drainage_line_title_text_colour**	**colour box**	**available colours**
text colour of drainage line name title.			
Offset (mm)	drainage_line_title_offset	input	
vertical adjustment to position of drainage line name title.			
X offset (mm)	drainage_line_title_x	input	
horizontal adjustment to position of drainage line name title.			
Draw title bottom linework	drainage_line_title_line	tick box	
Height of title left linework (mm)	drainage_line_title_height	input	

Continue to the next section 26.5.10.5.2 Arrows - Drainage Line Name - Arrow Type or return to 26.5.10 Arrows.

26.5.10.5.2 Arrows - Drainage Line Name - Arrow Type

Panel Field	Parameter name	Type	Pop-Up
Arrow type	drainage_line_arrow_mode	**choice box**	**available arrow types**
arrow type (0 to 8) for the drainage line name information.			
Colour	**drainage_line_arrow_colour**	**colour box**	**available colours**
arrow colour for drainage line name information.			
Size (mm)	**drainage_line_arrow_size**	**input**	
arrow size for drainage line name information.			
Arrow gap mode	drainage_line_arrow_gap	choice box	no gap in arrow **leave gap in arrow for text**
whether to put text gaps in the arrows, for drainage line name information.			

Continue to the next section 26.5.10.5.3 Arrows - Drainage Line Name - Arrow Text or return to 26.5.10 Arrows.

26.5.10.5.3 Arrows - Drainage Line Name - Arrow Text

Panel Field	Parameter name	Type	Pop-Up
Pre text	drainage_line_arrow_pre_text	**input**	

label text before the drainage line names.

Post text	drainage_line_arrow_post_text	**input**	

label text after the drainage line names.

Colour	**drainage_line_arrow_text_colour**	**colour box**	**available colours**

text colour for drainage line names.

Size (mm)	**drainage_line_arrow_text_size**	**input**	

text size for drainage line names.

Textstyle	**drainage_line_arrow_textstyle**	**font box**	**available fonts**

font of drainage line names.

Offset (mm)	drainage_line_arrow_text_offset	**input**	

vertical adjustment to position of drainage line names.

Continue to the next section 26.5.10.6 Arrows - User Defined Pipe Attributes or return to 26.5.10 Arrows.

26.5.10.6 Arrows - User Defined Pipe Attributes

User defined pipe attribute data can be drawn in the Arrow areas.

The following fields are required for each desired pipe attribute (set). Each set of fields forms one row of a grid.

Panel Field	Parameter name	Type	Pop-Up
Set #		input	

a unique integer n, greater than 0, to identify the nth set of pipe attribute plot parameters.

Draw pipe attribute mode	draw_pipe_attr_n	choice box	don't draw pipe attribute **draw pipe attribute**

whether to draw the nth set of pipe attribute data in one of the Arrow areas.

Pipe attribute name	pipe_attr_n_name	input	

name of the nth pipe attribute.

Arrow area for the arrow	pipe_attr_n_arrow_area	choice box	Area 1 **Area 2** **Area 3** **Area 4**

the desired Arrow area for the nth set of pipe attribute data.

Y offset (mm)	pipe_attr_n_y	input	

vertical offset of the nth set of pipe attribute data, measured upwards from the bottom of the selected Arrow area.

See

26.5.10.6.1 Arrows - User Defined Pipe Attributes - Title Text
26.5.10.6.2 Arrows - User Defined Pipe Attributes - Arrow Type
26.5.10.6.3 Arrows - User Defined Pipe Attributes - Arrow Text

Or return to 26.5.10 Arrows.

26.5.10.6.1 Arrows - User Defined Pipe Attributes - Title Text

The following fields are required for each desired pipe attribute (set). Each set of fields forms one row of a grid.

Panel Field	Parameter name	Type	Pop-Up
Set #		input	

a unique integer n, greater than 0, to identify the nth set of pipe attribute plot parameters. The Set # should match one of the Set #'s from the grid in the parent node: ***User defined pipe attributes.***

Text	pipe_attr_n_title	input	

title for the nth set of pipe attribute data.

Textstyle	**pipe_attr_n_title_textstyle**	**font box**	**available fonts**

font of the title for the nth set of pipe attribute data.

Size (mm)	**pipe_attr_n_title_text_size**	**input**	

text size of the title for the nth set of pipe attribute data.

Colour	**pipe_attr_n_title_text_colour**	**colour box**	**available colours**

text colour of the title for the nth set of pipe attribute data.

Offset (mm)	pipe_attr_n_title_offset	input	

vertical adjustment to position of the title for the nth set of pipe attribute data.

X offset (mm)	pipe_attr_n_title_x	input	

horizontal adjustment to position of the title for the nth set of pipe attribute data.

Draw title bottom linework		tick box	
Height of title left linework (mm)		input	

Continue to the next section 26.5.10.6.2 Arrows - User Defined Pipe Attributes - Arrow Type or return to 26.5.10 Arrows.

26.5.10.6.2 Arrows - User Defined Pipe Attributes - Arrow Type

The following fields are required for each desired pipe attribute (set). Each set of fields forms one row of a grid.

Panel Field	Parameter name	Type	Pop-Up
Set #		input	

a unique integer n, greater than 0, to identify the nth set of pipe attribute plot parameters. The Set # should match one of the Set #'s from the grid in the parent node: ***User defined pipe attributes****.*

Arrow type	pipe_attr_n_arrow_mode	choice box	available arrow types

arrow type (0 to 8) for the nth set of pipe attribute data.

Colour	**pipe_attr_n_arrow_colour**	**colour box**	**available colours**

arrow colour for the nth set of pipe attribute data.

Size (mm)	**pipe_attr_n_arrow_size**	**input**	

arrow size for the nth set of pipe attribute data.

Arrow gap mode	pipe_attr_n_arrow_gap	choice box	no gap in arrow **leave gap in arrow for text**

whether to put text gaps in the arrows, for the nth set of pipe attribute data.

Continue to the next section 26.5.10.6.3 Arrows - User Defined Pipe Attributes - Arrow Text or return to 26.5.10 Arrows.

26.5.10.6.3 Arrows - User Defined Pipe Attributes - Arrow Text

The following fields are required for each desired pipe attribute (set). Each set of fields forms one row of a grid.

Panel Field	Parameter name	Type	Pop-Up
Set #		input	

a unique integer n, greater than 0, to identify the nth set of pipe attribute plot parameters. The Set # should match one of the Set #'s from the grid in the parent node: ***User defined pipe attributes****.*

Pre text pipe_attr_n_arrow_pre_text input

label text before the values of the nth set of pipe attribute data.

Post text pipe_attr_n_arrow_post_text input

label text after the values of the nth set of pipe attribute data.

Units factor

Decimal places pipe_attr_n_arrow_decimals input

decimal places the values of the nth set of pipe attribute data.

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Colour pipe_attr_n_arrow_text_colour colour box available colours

text colour for the values of the nth set of pipe attribute data.

Size (mm) pipe_attr_n_arrow_text_size input

text size for the values of the nth set of pipe attribute data.

Textstyle pipe_attr_n_arrow_textstyle textstyle box available textstyles

font for the values of the nth set of pipe attribute data.

Offset (mm) pipe_attr_n_arrow_text_offset input

vertical adjustment to position of values of the nth set of pipe attribute data.

Please continue to the next section 26.5.11 Drainage Graph Area.

26.5.11 Drainage Graph Area

The Graph area is where the diagram of the drainage line, in long-section, is drawn. In addition, if Hydraulic Grade Line (HGL) information is set, the HGL can also be drawn in the Graph area.

Panel Field	Parameter name	Type	Pop-Up
Left/Right extensions (world units)	profile_extension	input	
Drainage string colour	drainage_colour	colour box	available colours
Draw finished surface tin	draw_fs_tin	tick box	
Finished surface tin colour	fs_tin_colour	colour box	available colours
Draw natural surface tin	draw_ns_tin	tick box	
Natural surface tin colour	ns_tin_colour	colour box	available colours

Draw HGL draw_hgl_diag tick box

whether to draw the HGL in the Graph area.

HGL colour hgl_colour colour box available colours

colour of HGL.

HGL linestyle hgl_linestyle choice box

linestyle of HGL.

Please continue to the next section 26.5.12 Top Area.

26.5.12 Top Area

Panel Field	Parameter name	Type	Pop-Up
Line at the bottom of the top area mode			
	draw_top_line	choice box	don't draw line draw line

whether to draw a horizontal line at the bottom of the Top area.

Please continue to the next section 26.5.13 Corridors - Drainage Long.

26.5.13 Corridors - Drainage Long

A corridor around the primary string is defined by giving a left and right corridor width.

Any string in a model added to the section view is checked to see if it appears in the corridor, and if it does, it is drawn on the cross-section plot.

To be drawn, strings do not have to cross the primary string, but just be in the corridor.

This is documented for all the PPF Editors in 26.2.7 Corridors.

Please continue to the next section 26.5.14 Maintenance Holes.

26.5.14 Maintenance Holes

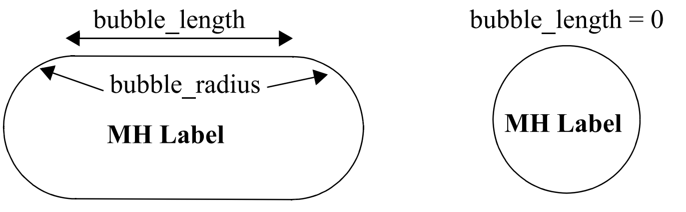
The **drainage string name**, the **maintenance hole** name (pit number) and **maintenance hole type** (pit type) can be drawn on the drainage long section plot as **MH label,** with or without a *bubble* around it.

The MH label text is made up of

pre_text line_name / pit_name   pit_type post_text

where each bit of the MH label is controlled by parameters.

The *bubble* may or may not exist



The **MH label** (and bubble) can be positioned at the bottom of the **top area**, or can be placed a fixed distance above the top of the corresponding manhole.

If the **bubble text** is drawn at the bottom of the top area, the upright is automatically drawn up to the **bubble text.**

If the **bubble text** is drawn above the pit, it is positioned by the *distance_above_pit* parameter **plus** the top stagger distances, *stagger_gap_top*, *stagger_height_3* and *stagger_height_4*.

This is necessary because the **bubble text** may need to be staggered.

The columns for the fields documented in the sections are for.

Panel Field	**Parameter name**	**Type**	**Pop-Up**

Section: Maintenance hole names/types/bubbles

Include line name mode	draw_line_name	choice box	don't include line name include line name

whether to include the drainage line name in the MH name label.

Include pit name mode	draw_pit_name	choice box	don't include pit name include pit name

whether to include the MH name in the MH name label.

Include pit type mode	draw_pit_type	choice box	don't include pit type include pit type

whether to include the MH type in the MH name label.

Size (mm)	bubble_text_size	real box	

text size of MH name label.

Colour	bubble_text_colour	colour box	available colours

text colour of MH name label.

Textstyle	bubble_textstyle	textstyle box	available textstyles

font of MH name label.

X adjustment (mm)	bubble_text_x	real box	

horizontal adjustment to position of MH name label.

Y adjustment (mm)	bubble_text_y	real box	

vertical adjustment to position of MH name label.

Pre text	bubble_pre_text	text box	

text to go before MH name label.

Post text	bubble_post_text	text box	

text to go after MH name label.

Text angle (dms)	bubble_text_angle	angle box	

text angle of MH name label.

Text justification	bubble_text_justify	justification box	

text justification of MH name label.

Bubble radius (mm)	bubble_radius	real box	

radius of bubble around MH name label. Zero (0) for no bubble.

Bubble colour	bubble_colour	colour box	available colours

colour of bubble around MH name label.

Bubble length (mm)	bubble_length	input	

length of bubble around MH name label. Zero (0) for circular bubble.

Bubble text position	draw_text_at_pit	choice box	draw bubble&text in top area draw bubble&text above pit

whether to position the MH name labels and bubbles in the Top area, or directly above the pits.

Distance to add to place bubble above the pit (mm) distance_above_pit real box

for draw_text_at_pit *set to* draw bubble & text above pit*, distance to add between top of pit and bottom of bubble.*

See

26.5.14.1 Maintenance Holes - Line/Junction Deflection Angles
26.5.14.2 Maintenance Holes - Symbols
26.5.14.3 Maintenance Holes - Attribute Labels

Or return to 26.5 Drainage Long Plot PPF Editor.

26.5.14.1 Maintenance Holes - Line/Junction Deflection Angles

If there is a change of direction of the pipes of the plotted drainage string going through the pit, the **deflection angle (pit angle)** is drawn above the MH label.

The **deflection angle text** is made up:

angled_pre_text   deflection angle   angled_post_text

Also **any junctions** at a pit in the plotted drainage string can be labelled with the name of the drainage strings coming into the pit, and the angle of the pipes at the junction.

Hence the **junction text** for each junction at the pit is made up of **two** lines:

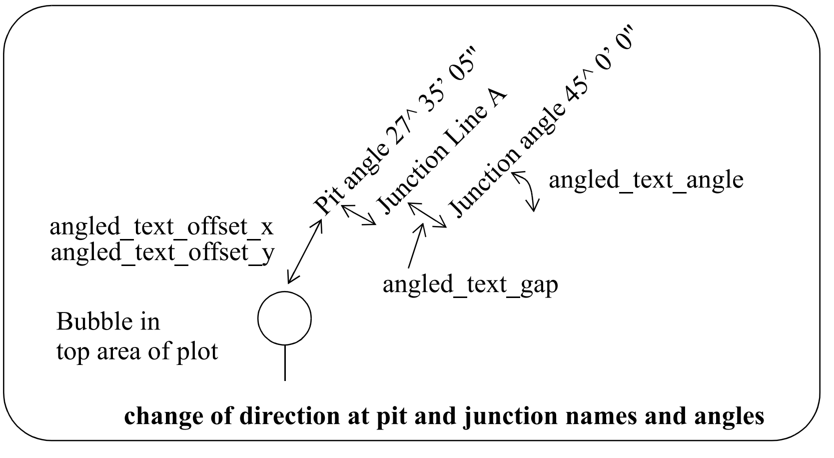
junction_pre_text   line_name   junction_post_text
junction_angle_pre_text   junction angle   junction_angle_post_text

Note that there may be more than one junction at the pit.

If bubbles are drawn around the MH label, the deflection angle and junction name and angle are drawn at the distance (***angled_text_x_offset, angled_text_y_offset)*** above the bubble and with the distance ***angled_text_gap*** between the lines of information.

If bubbles are not drawn, **MH label** and the deflection angle and junction names and angles are placed the distance (***angled_text_x_offset, angled_text_y_offset)*** above the bottom of the top area. Each line of text is separated by the distance ***angled_text_gap***.

The deflection angle and junction name and angle are drawn at an angle of ***angled_text_angle***.



The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Maintenance hole line/junction deflection angles

this section defines where the lines of text will be, and the angle of the text and distance between the lines of text.

Angled text x offset (mm) angled_text_x_offset real box

distance in mm that the text is offset in x from the centre of the top of the bubble.

Angled text y offset (mm) angled_text_offset real box

distance in mm that the text is offset in y from the centre of the top of the bubble.

Angled text horiz gap (mm) angled_text_gap real box

perpendicular distance in mm between the pit angle text and the junction texts.

Angled text angle (dms) angled_text_angle angle box

angle of the text in degrees minutes and seconds in hp notation.

Angled text justification angled_text_justification choice box left, right

justification of the pit angle text and the junction texts. It can only be "left" and "right".

Degrees mode angled_text_degrees_mode choice box DDDMMSS.s
DDDMM
DDD
Decimal degrees

*if **DDMMSS.s** the angle is given in degrees, minutes and seconds and tenths of a second*
*if **DDMMSS.s** the deflection angle is given in degrees, minutes and seconds and tenths of a second.*
*If **DDDMM** the deflection angle is given in degrees and minutes.*
*If **DDD** the deflection angle is given in degrees.*
*If **Decimal degrees** the deflection angle is given in decimal degrees with the number of decimal places given by* **Degrees decimals**.

Degrees decimals angled_text_degrees_decimals integer box

*if **Degrees mode** is Decimal degrees the it will be written out t this many decimal places.*

Section: Drainage line deflection angles

this section defines the pre and post text, textstyle, size and colour for the deflection angle (pit angle) text.

Pre text angled_pre_text text box

text to go before the pit deflection angle text.

Post text angled_post_text text box

text to go after the pit deflection angle text.

Colour angled_text_colour colour box available colours

colour of the pit deflection angle text.

Size (mm) angled_text_size real box

size in millimetres of the pit deflection angle text.

Textstyle angled_textstyle textstyle box available textstyles

textstyle of the pit deflection angle text.

See

26.5.14.1.1 Maint Holes - Line/Junction Deflection Angles - Junction Names
26.5.14.1.2 Maint Holes - Line/Junction Defl Ang - Junction Deflection Angles

Or return to 26.5.14 Maintenance Holes.

26.5.14.1.1 Maint Holes - Line/Junction Deflection Angles - Junction Names

The columns for the fields documented in the sections are for.

Panel Field | **Parameter name** | **Type** | **Pop-Up**

Section: Maintenance hole junction name

this section defines the pre and post text, textstyle, size and colour for the junction name text.

Junction name mode junction_name_mode choice box don't include junction string name
include junction string name

Pre text junction_pre_text text box

text to go before the junction name text.

Post text junction_post_text text box

text to go after the junction name text.

Colour junction_text_colour colour box available colours

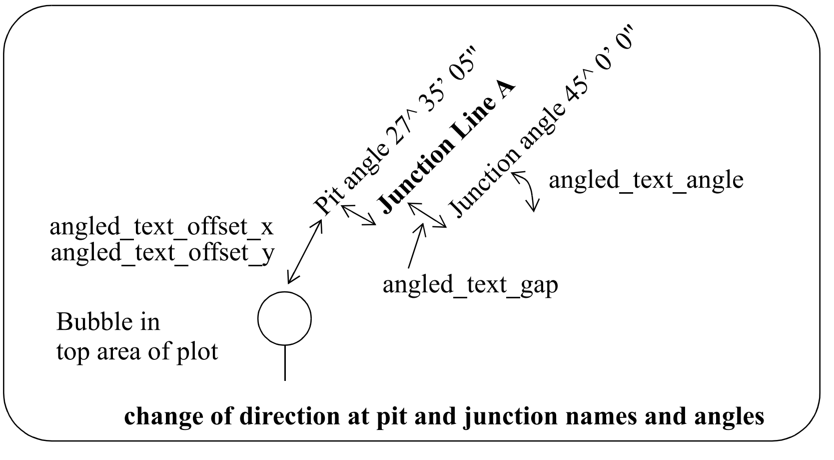
colour of the junction name text.

Size (mm) junction_text_size real box

size in millimetres of the junction name text.

Textstyle junction_textstyle textstyle box available textstyles

textstyle of the junction name text.



Continue to the next section 26.5.14.1.2 Maint Holes - Line/Junction Defl Ang - Junction Deflection Angles or return to 26.5.14 Maintenance Holes.

26.5.14.1.2 Maint Holes - Line/Junction Defl Ang - Junction Deflection Angles

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Maintenance hole junction deflection angles

this section defines the pre and post text, textstyle, size and colour for the junction angle text.

Append mode junction_angle_append choice box
label junction name and angle on separate lines
append angle to junction name
append name to junction angle

Pre text junction_angle_pre_text text box

text to go before the junction angle text.

Post text junction_angle_post_text text box

text to go after the junction angle text.

Colour junction_angle_text_colour colour box available colours

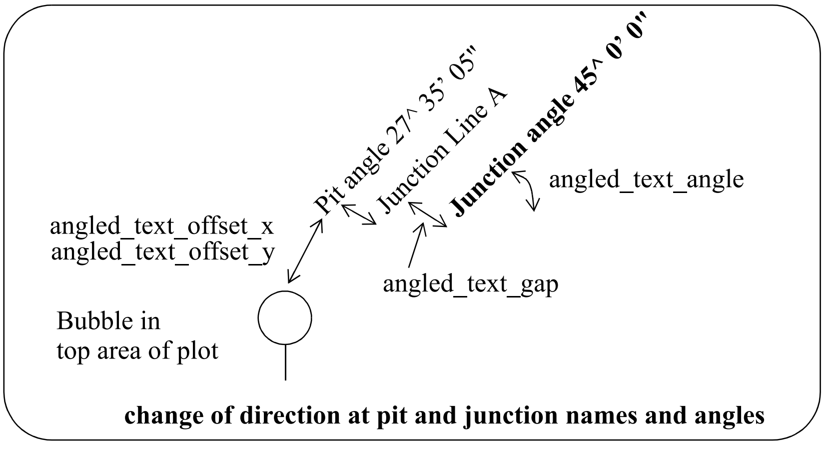
colour of the junction angle text.

Size (mm) junction_angle_text_size real box

size in millimetres of the junction angle text.

Textstyle junction_angle_textstyle textstyle box available textstyles

textstyle of the junction angle text.



Continue to the next section 26.5.14.2 Maintenance Holes - Symbols or return to 26.5.14 Maintenance Holes.

26.5.14.2 Maintenance Holes - Symbols

Symbols representing the different types of maintenance holes used, can be drawn on the plots.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Maintenance hole symbols

this section defines the pre and post text, textstyle, size and colour for the junction name text.

The fields for each row form one set of controls for the placement of MH symbols on the plot. Normally, the grid will contain one row for each MH type/symbol used.

Symbols Grid

All MH types tick box

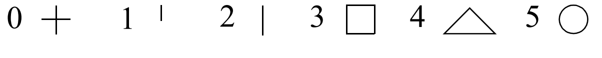
*if **ticked**, then all MH types are labelled with the fields in this row.*
*If **not ticked**, then all maintenance holes with the given* **MH Type** *are labelled with the fields in this row.*

MH type manhole_symbol_n_type choice box available MH types

maintenance holes with this **MH type** *are labelled with the fields in this row.*

MH symbol mode manhole_symbol_n_mode choice box cross (0)
up from centre of box (1)
up&down from box centre(2)
square (3)
triangle, base at bottom (4)
circle (5)
use a 12d symbol (6)

for the MH type for this row, whether to use one of the hard-wired symbols (1-5), or a symbol from the 12d symbols list.(6)



predefined symbols for* manhole_symbol_n_modes *0 to 5

MH symbol position manhole_symbol_n_position
choice box at top of mh
above top of boxes
above highest point
to primary string
to first found tin

for the MH type for this row, the desired position of the symbol.

Stagger tick box

Size (mm) manhole_symbol_n_size real box
for the MH type for this row, the symbol size.

X (mm) manhole_symbol_n_x real box

for the MH type for this row, the horizontal adjustment to the selected MH symbol position.

Y (mm) manhole_symbol_n_y real box

for the MH type for this row, the vertical adjustment to the selected MH symbol position.

Symbol manhole_symbol_n_style symbol box available symbols

for the MH type for this row, and with Manhole symbol mode *set to* use a symbol*, the symbol to use from the 12d symbols list.*

Angle (dms) manhole_symbol_n_angle angle box

for the MH type for this row, the symbol angle in degrees minutes and seconds in hp notation.

Colour manhole_symbol_n_colour colour box available colours

for the MH type for this row, the symbol colour.

Continue to the next section 26.5.14.3 Maintenance Holes - Attribute Labels or return to 26.5.14 Maintenance Holes.

26.5.14.3 Maintenance Holes - Attribute Labels

The text for Maintenance Holes attributes can be drawn on the plots.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Maintenance hole attribute labels

this section defines the pre and post text, textstyle, size and colour for displaying the attributes as text.

The fields for each row form one set of controls for the placement of the given attribute on the plot.

Attributes Grid

Label MH attribute tick box

*if **ticked**, this row is used and the attribute is labelled.*
*If **not ticked**, this row is ignored.*

Attribute name

the full path name of the attribute to be labelled.

From primary string at junction pits (non-standard) tick box

*if **ticked**,.*

Text size (mm)

the text size for this attribute label.

Label position choice box at top of mh
at ds il of mh
at sump of mh
at first found tin
at top of boxes
at highest point

position to place attribute label

Stagger tick box

the text size for this attribute label.

X off (mm) real box

the horizontal adjustment to the selected label position.

Y off (mm) real box

the vertical adjustment to the selected label position.

Angle (dms) angle box

the angle of the attribute label in degrees minutes and seconds in hp notation.

Pre text text box

text to go before the attribute label.

Post text text box

text to go after the attribute label.

Decimal places integer box

number of decimal places to use when it is a real attribute.

*If > 0, trailing zeros are **removed** after the decimal point.*
If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Units factor real box

for real and integer attributes, the value is multiplied by the **Units factor**.

Colour colour box

the colour of the attribute label.

Textstyle textstyle box

the textstyle of the attribute label.

Justification justification box

the justification of the attribute label.

Please continue to the next section 26.5.15 Property Controls/House Connections.

26.5.15 Property Controls/House Connections

Section: Property control parameters

Draw property control mode	draw_property_controls	choice box	draw property controls don't draw property controls

Section: House connection parameters

Draw house connections mode	draw_house_connections	choice box	draw house connections don't draw house connections
House connection width (mm)	house_connection_width	real box	

26.5.15.1 Property Controls/House Connections - House Connection Labels

26.5.15.1.1 PC/HC - House Connection Labels - Symbols

26.5.15.1.2 PC/HC - House Connection Labels - House Connection Type

26.5.15.1.3 PC/HC - House Connection Labels - Distance

26.5.15.1.4 PC/HC - House Connection Labels - Depths

26.5.15.1.5 PC/HC - House Connection Labels - Finished Surface

26.5.15.1.6 PC/HC - House Connection Labels - Invert Levels

26.5.15.2 Property Controls/House Connections - Property Control Labels

26.5.15.2.1 PC/HC - Property Control Labels - Symbols

26.5.15.2.2 PC/HC - Property Control Labels - Distance

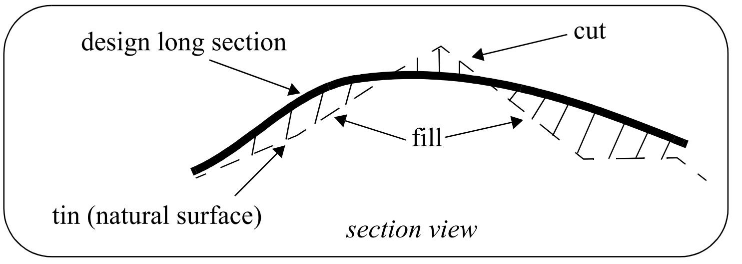
26.5.15.2.3 PC/HC - Property Control Labels - Finished Surface

26.5.15.2.4 PC/HC - Property Control Labels - Invert Levels

Please continue to the next section 26.5.16 Hatching Cut/Fill.

26.5.16 Hatching Cut/Fill

This option is used to hatch cut and/or fill areas between sets of tins.



The **Hatching Cut/Fill** section is common to the **PPF Editors** and is fully documented in 26.2.8 Hatching Cut/Fill.

Section: Hatching cut/fill - Tin parameters - *see* 26.2.8 Hatching Cut/Fill

Section: Hatching cut/fill - Cut parameters - *see* 26.2.8.1 Hatching Cut/Fill - Cut

Section: Hatching cut/fill - Fill parameters- *see* 26.2.8.2 Hatching Cut/Fill - Fill

Please continue to the next section 26.5.17 Cuts - Drainage/Sewer Long.

26.5.17 Cuts - Drainage/Sewer Long

The cuts that the primary string (drainage or sewer line) makes though strings in any user specified model can be automatically labelled on the drainage/sewer long section plots.

The **Cuts** section is common to the **PPF Editors** and is fully documented in 26.2.9 Cuts.

Section: Cuts - Model/Name mask parameters - *see* 26.2.9 Cuts.

Section: Cuts - Chainage parameters - *see* 26.2.9.3 Cuts - Chainage - Long Sections Only

Section: Cuts - Height parameters - *see* 26.2.9.4 Cuts - Heights

Section: Cuts - Diameter parameters - *see* 26.2.9.5 Cuts - Diameters

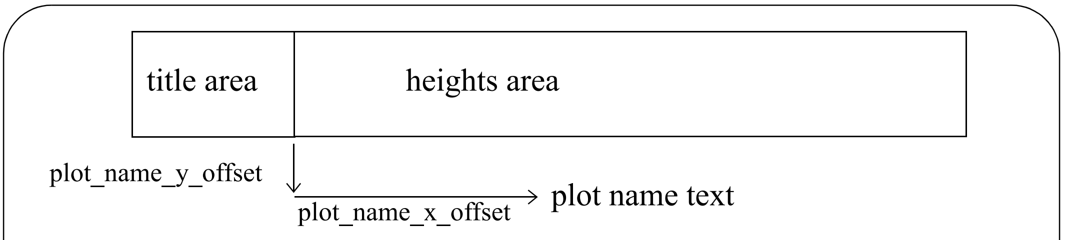
Section: Cuts - Label parameters - *see* 26.2.9.6 Cuts - Labels

Section: Cuts - Symbol parameters - *see* 26.2.9.7 Cuts - Symbols

Continue to the next section 26.5.18 Primary String Name Label.

26.5.18 Primary String Name Label

The plot can be labelled with a name under the boxes area.



labelling primary string name on long section plots

The name is made up of a concatenation of the text string names:

Panel Field	Parameter name	Type
Name mode	plot_name_string_name	choice box
determines what text is placed in the name labels.		
Pre-text	plot_name_pre_text	text box
text before label.		
Post-text	plot_name_post_text	text box
text after label.		
Textstyle	plot_name_textstyle	textstyle box
textstyle of label.		
Size (mm)	plot_name_size	real box
size of label.		
Colour	plot_name_colour	colour box
colour of label.		
X offset (mm)	plot_name_x_offset	real box
horizontal adjustment to position of label.		
Y offset (mm)	plot_name_y_offset	real box
vertical adjustment to position of label.		

The plot name is positioned under the boxes.

The *plot_name_x_offset* is measured from the beginning of the height boxes.

The default for *plot_name_x_offset* is centred on heights area.

The *plot_name_y_offset* is measured from the bottom of the box area with positive being down.

Please continue to the next section 26.5.19 PPFs to include - Drainage Long Section.

26.5.19 PPFs to include - Drainage Long Section

This is documented for all the PPF Editors in 26.2.10 PPFs To Include.

Please continue to the next section 26.5.20 Buttons at Bottom of Panel.

26.5.20 Buttons at Bottom of Panel

This is documented for all the PPF Editors in 26.2.2 View to Load and Global Variables.

Return to 26.5 Drainage Long Plot PPF Editor.

26.6 Melbourne Water Plot PPF Editor

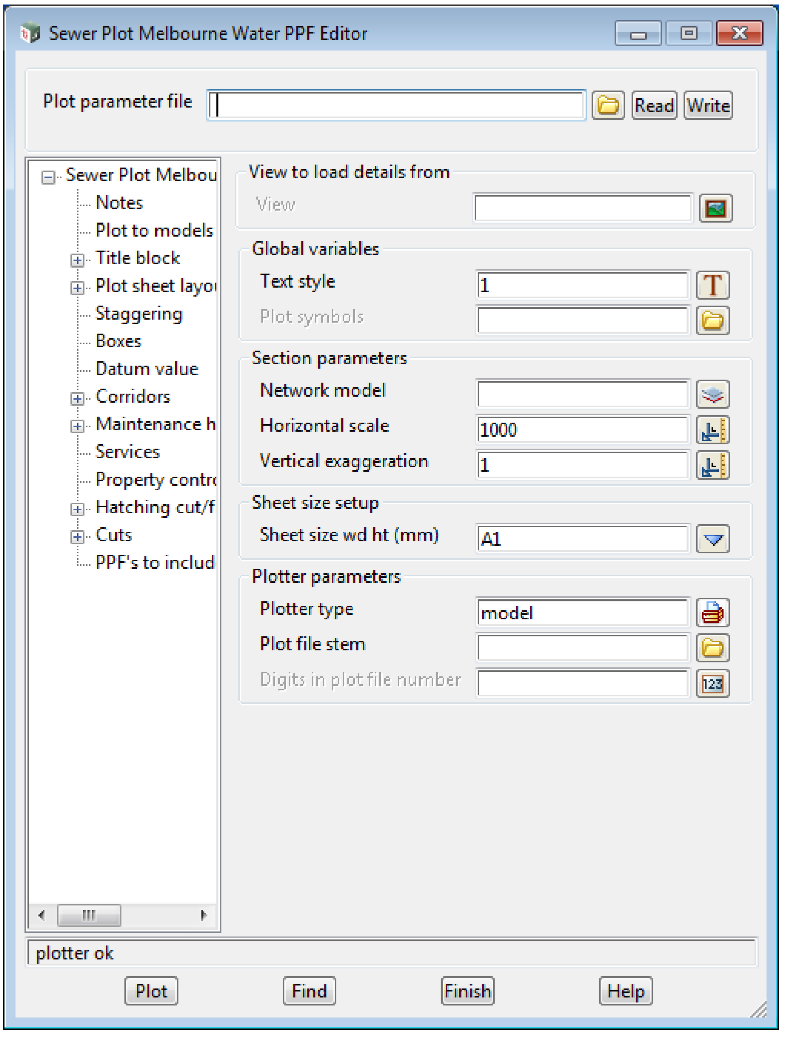
Position of option on menu: **Plot =>Plot and PPF Editors =>Melbourne Water**

The **Melbourne Water plot PPF editor** is for creating and/or editing a (binary) Melbourne Water sewer long section PPF and for creating a Melbourne Water sewer long section plot. An text version of the file is also produced.

Note: Binary and text PPFs are stored *within* the project (not in the folder containing the project).

On selecting the **Melbourne Water** option, the **Sewer Plot Melbourne Water PPF Editor** panel is displayed.

The plot parameters for controlling the cross section plots are accessed by expanding to the appropriate node in the ***Sewer Plot Melbourne Water*** tree (click on the **+** to expand to node or **-** to collapse the node) and then clicking on the required node, and the required information to fill in is displayed on the right hand side of the panel.



For information on all the different nodes see:

[26.6.1 Melbourne Water Plot - Front Page](#)
[26.6.2 Notes - Melbourne Water](#)
[26.6.3 Plot to Models - Melbourne Water](#)
[26.6.4 Title Block - Melbourne Water](#)

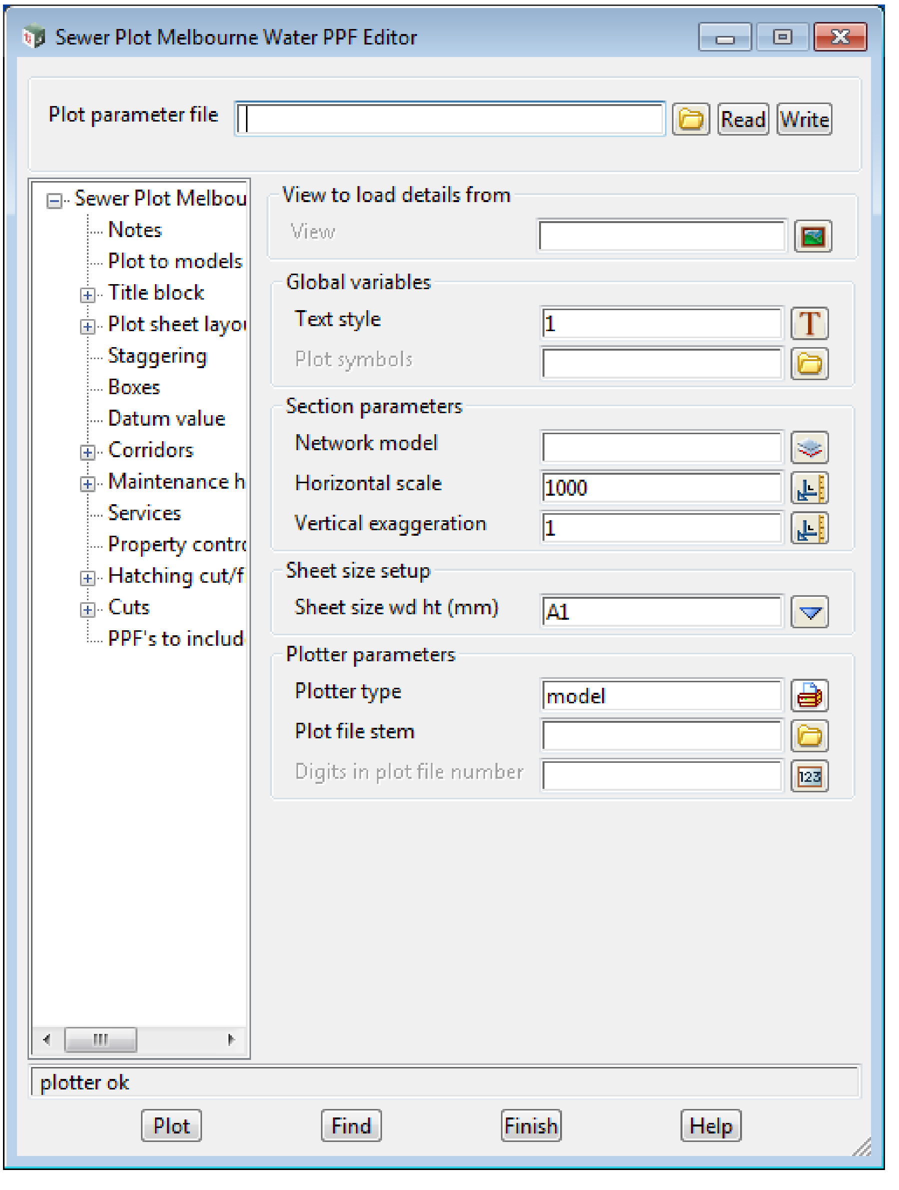
26.6.1 Melbourne Water Plot - Front Page

The Melbourne Water sewer long plot itself consists of the three regions - boxes, datum and graph areas.

The **boxes area** is where the titles and the chainage values and the heights/depths for the strings drawn on the long plot are labelled.

The **datum area** is the region between the boxes area and the graph area.

The **graph area** is the area where the actual plots of the strings are drawn.



Section: Plot parameter file

a ***plot parameter file*** *can be used to load values into the fields of the PPF Editor, or as a file to write out all current values in a PPF Editor to.*

This section is documented for all the PPF Editors in 26.2.1 Plot Parameter File.

Section: Section: View to load details from AND Global variables

*a **section view** can be selected to load certain values into fields of the PPDF Editor. For example, Vertical exaggeration and Corridor models. And there are variables to use if some values aren't given.*

These sections are documented for all the PPF Editors in 26.2.2 View to Load and Global Variables.

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Section parameters

Network model	network_model	model box	available models

model containing the sewer strings to draw in long section on the plot.

Horizontal Scale	scale	real box	

horizontal scale to be used for the long section plots

Vertical exaggeration	vertical_exaggeration	real box	

vertical scale to be used for the long section plots

Section: Sheet size setup and Plotter parameters

these sections define the size of the "paper" to plot on, the type of plotter to use and the naming to use for the plot files.

These sections are documented for all the PPF Editors in 26.2.3 Sheet Size and Plotter Parameters.

Continue to the next section 26.6.2 Notes - Melbourne Water.

26.6.2 Notes - Melbourne Water

This is documented for all the PPF Editors in 26.2.4 Notes.

Continue to the next section 26.6.3 Plot to Models - Melbourne Water.

26.6.3 Plot to Models - Melbourne Water

This is documented for all the PPF Editors in 26.2.5 Plot to models.

Continue to the next section 26.6.4 Title Block - Melbourne Water.

26.6.4 Title Block - Melbourne Water

This is documented for all the PPF Editors in 26.2.6.1 Title Block Section in PPF Editors.

For more general information about a title block, see 26.2.6 Title Block.

Continue to the next section 26.6.5 Plot Sheet Layout - Melbourne Water.

26.6.5 Plot Sheet Layout - Melbourne Water

The information on this node is the same as that for Drainage Long Sections - see 26.5.6 Plot Sheet Layout

Continue to the next section 26.6.6 Staggering - Melbourne Water.

26.6.6 Staggering - Melbourne Water

Continue to the next section 26.6.7 Boxes - Melbourne Water.

26.6.7 Boxes - Melbourne Water

Continue to the next section 26.6.8 Datum Value - Melbourne Water.

26.6.8 Datum Value - Melbourne Water

Continue to the next section 26.6.9 Corridors - Melbourne Water.

26.6.9 Corridors - Melbourne Water

A corridor around the primary string is defined by giving a left and right corridor width.

Any string in a model added to the section view is checked to see if it appears in the corridor, and if it does, it is drawn on the cross-section plot.

To be drawn, strings do not have to cross the primary string, but just be in the corridor.

This is documented for all the PPF Editors in 26.2.7 Corridors.

Continue to the next section 26.6.10 Maintenance Holes - Melbourne Water.

26.6.10 Maintenance Holes - Melbourne Water

Continue to the next section 26.6.11 Services - Melbourne Water.

26.6.11 Services - Melbourne Water

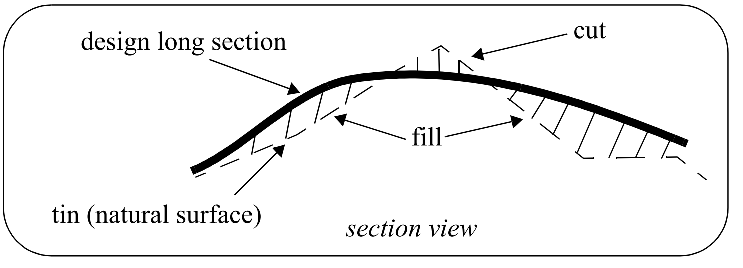
Continue to the next section 26.6.12 Property Controls/House Connections - Melbourne Water.

26.6.12 Property Controls/House Connections - Melbourne Water

Continue to the next section 26.6.13 Hatching Cut/Fill - Melbourne Water.

26.6.13 Hatching Cut/Fill - Melbourne Water

This option is used to hatch cut and/or fill areas between sets of tins.



The **Hatching Cut/Fill** section is common to the **PPF Editors** and is fully documented in 26.2.8 Hatching Cut/Fill.

Section: Hatching cut/fill - Tin parameters - *see* 26.2.8 Hatching Cut/Fill

Section: Hatching cut/fill - Cut parameters - *see* 26.2.8.1 Hatching Cut/Fill - Cut

Section: Hatching cut/fill - Fill parameters- *see* 26.2.8.2 Hatching Cut/Fill - Fill

Continue to the next section 26.6.14 Cuts - Melbourne Water Long.

26.6.14 Cuts - Melbourne Water Long

The cuts that the primary string (drainage or sewer line) makes though strings in any user specified model can be automatically labelled on the drainage/sewer long section plots.

The **Cuts** section is common to the **PPF Editors** and is fully documented in 26.2.9 Cuts.

Section: Cuts - Model/Name mask parameters - *see* 26.2.9 Cuts.

Section: Cuts - Chainage parameters - *see* 26.2.9.3 Cuts - Chainage - Long Sections Only

Section: Cuts - Height parameters - *see* 26.2.9.4 Cuts - Heights

Section: Cuts - Diameter parameters - *see* 26.2.9.5 Cuts - Diameters

Section: Cuts - Label parameters - *see* 26.2.9.6 Cuts - Labels

Section: Cuts - Symbol parameters - *see* 26.2.9.7 Cuts - Symbols

Continue to the next section 26.6.16 PPFs to include - Melbourne Water.

26.6.15 Paired Cuts - Melbourne Water

Paired cuts uses pairs of strings and where both strings cut the primary string, the cuts on the long plot can be labelled with information such as

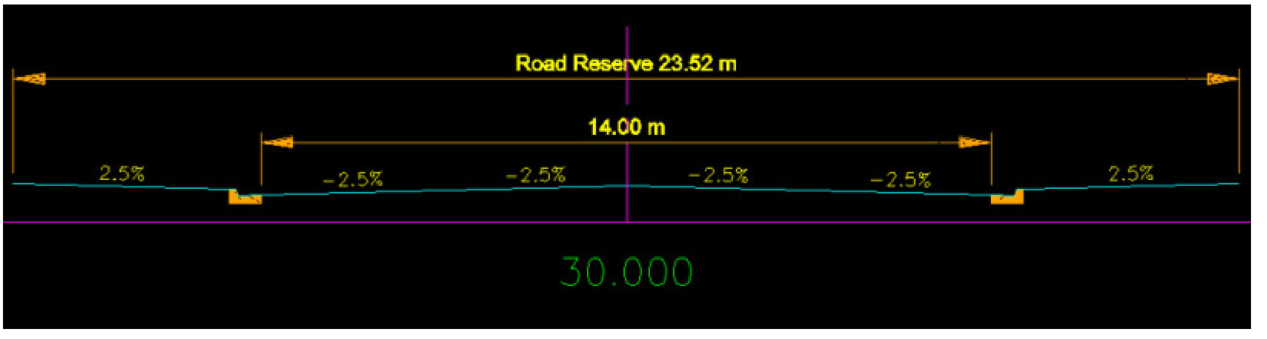
(a) the name of the first and second cut strings

(b) attributes from the first and second cut string

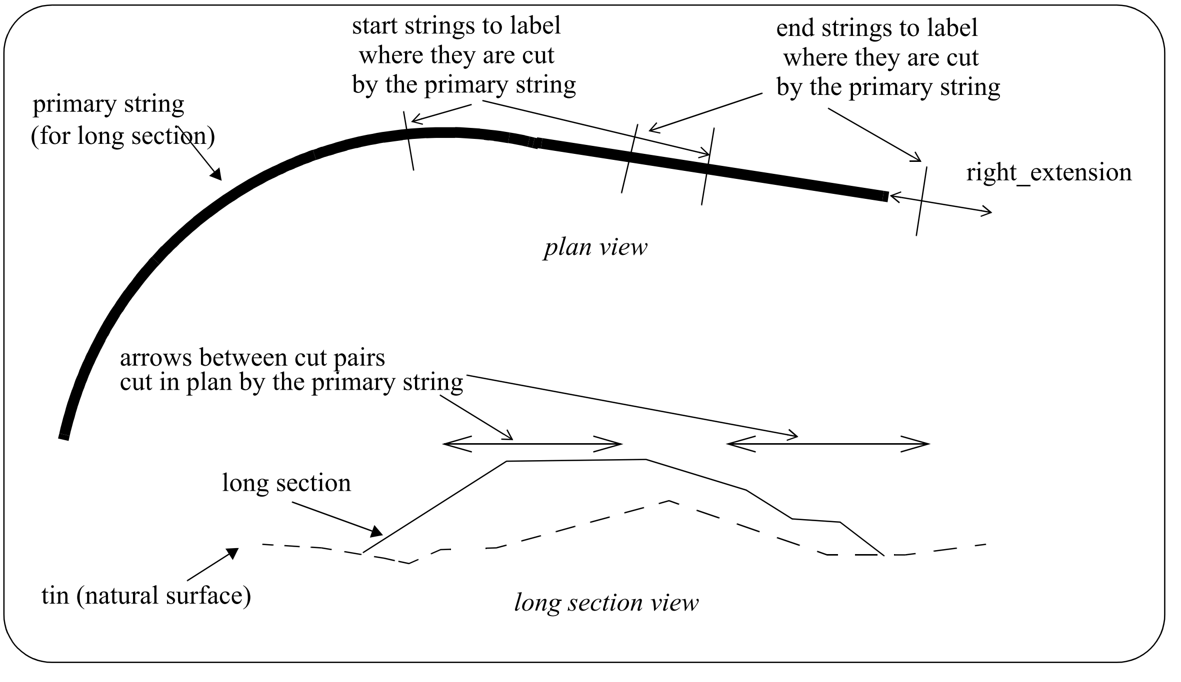
(c) plan distance between the two cuts of the pair

(d) 3d length between the two cuts of the pair

(e) the chainge on the long section of the first and second cuts of the pair



Symbols can be drawn at the chainages of the first and second cuts (at a height specified when defining the Cut sets), and a line drawn between the symbols. Using both the line and a symbol of an arrow head makes an arrow between the two cuts.



The method for specifying which strings are to be checked for paired cuts is by first specifying the **models** (using wild cards and characters) that contains the strings, and then a **start name mask** to select the all the strings that are to be the first strings in a cut pair, and an **end name mask** to select all the strings that are the second strings in a cut pair.

For a long section, all the cuts of the selected **start** strings are found and the cuts ordered by the chainage of the cut with the primary string of the long section. Then all the cuts of the selected **end** strings are found and the cuts ordered by the chainage of the cut with the primary string of the long section.

The start cuts are then processed and each start cut is paired with the next end cut with a larger chainage than the start cut. It is possible that there are end cuts before the first start cut (orphaned end cuts) and start cut with no following end cut (orphaned start cuts).

Up to twenty five different sets of models and name masks can be used so that different paired cut sets can be labelled in different ways.

26.6.15.1 Paired Cuts - Front Page

The information on this section is the same as that for Long Sections, please see 26.4.23.1 Paired Cuts (Long Section) - Front Page

26.6.15.1.1 Paired Cuts - Lines and Symbols

The information on this section is the same as that for Long Sections, please see 26.4.23.2 Paired Cuts (Long Section) - Lines and Symbols

26.6.15.1.2 Paired Cuts - Labels

The information on this section is the same as that for Long Sections, please see 26.4.23.3 Paired Cuts (Long Section) - Labels

26.6.15.1.3 Paired Cuts - Start Chainage

The information on this section is the same as that for Long Sections, please see 26.4.23.4 Paired Cuts (Long Section) - Start Chainage

26.6.15.1.4 Paired Cuts - End Chainage

The information on this section is the same as that for Long Sections, please see 26.4.23.5 Paired Cuts (Long Section) - End Chainage

26.6.16 PPFs to include - Melbourne Water

This is documented for all the PPF Editors in 26.2.10 PPFs To Include.

Continue to the next section 26.6.17 Buttons at Bottom of Panel - Melbourne Water.

26.6.17 Buttons at Bottom of Panel - Melbourne Water

This is documented for all the PPF Editors in 26.2.11 Buttons at the Bottom of the PPF Editors.

Return to 26.6 Melbourne Water Plot PPF Editor.

26.7 Pipeline Plot PPF Editor

Position of option on menu: **Plot => Plot and PPF Editors => Pipelines**

Position of option on menu: **Design =>Pipeline =>Plots**

The **Pipeline Plot PPF Editor** is for creating and/or editing a (binary) pipeline long section ppf file and for creating a pipeline long section plot for all the pipeline strings in a given model.

The pipeline long section plot will break an individual plot up if it doesn't fit across the sheet. There can be one or more rows of plot on the same sheet.

The top row is done first, followed by the second top row, then the third and so on until the bottom row. If there is only one row, it is considered to be the bottom row.

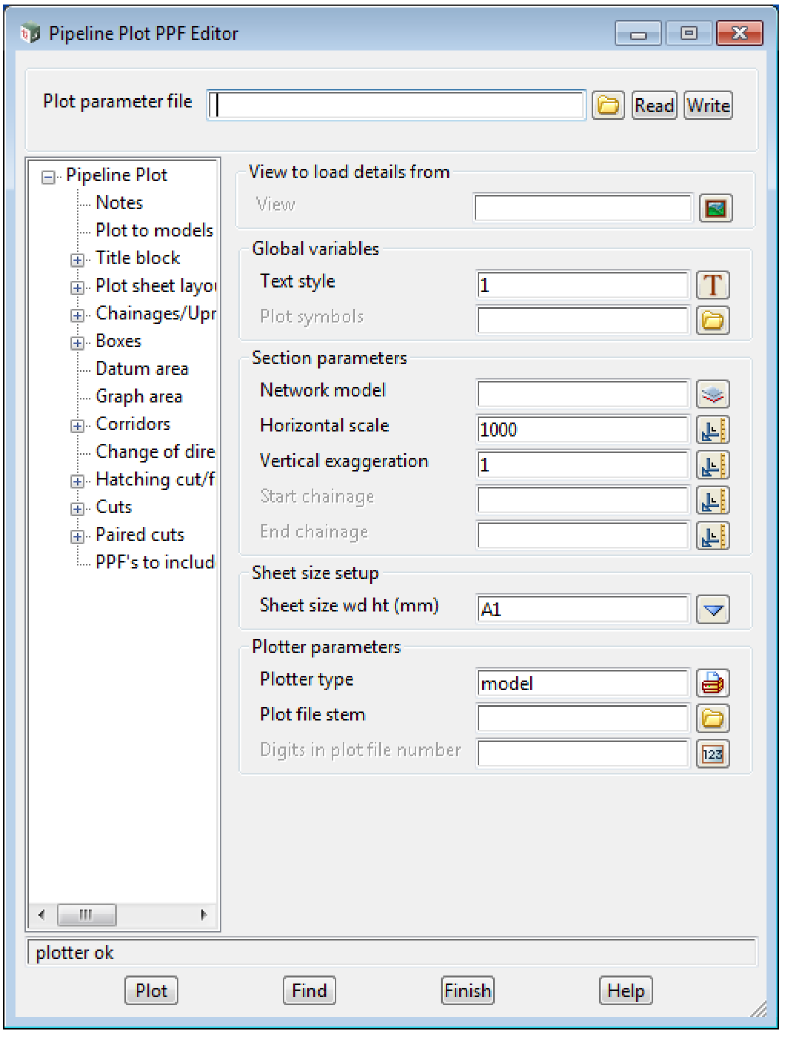
When a sheet is full, a follow on sheet is created.

As soon as one pipeline string is completed, the next pipeline string in the network model is plotted beginning on the same row as the previous pipeline string and with a horizontal gap of size horizontal_plot_gap between the plots. If there is not enough room on the row to start the next plot, it will begin on a new row.

Note: Binary PPFs are stored *within* the project (not in the folder containing the project as the text PPFs were).

On selecting the **Pipelines** option, the **Pipeline Plot PPF Editor** panel is displayed.

The plot parameters for controlling the cross section plots are accessed by expanding to the appropriate node in the ***Pipeline Plot*** tree (click on the **+** to expand to node or **-** to collapse the node) and then clicking on the required node, and the required information to fill in is displayed on the right hand side of the panel.



For information on all the different nodes see:

26.7.1 General Information on Pipeline Long Section Plots
26.7.2 Pipeline Longsection Plot - Front Page
26.7.3 Notes - Pipeline
26.7.4 Plot to Models - Pipeline
26.7.5 Title Block - Pipeline
26.7.6 Plot Sheet Layout - Pipeline
26.7.7 Chainage/Uprights - Pipeline
26.7.8 Boxes - Pipeline
26.7.9 Datum Value - Pipeline
26.7.10 Graph Area - Pipeline
26.7.11 Corridors - Pipeline
26.7.12 Change_of_Direction_at_Intersection_Point
26.7.13 Hatching Cut/Fill - Pipeline
26.7.14 Cuts - Pipeline
26.7.15 Paired Cuts - Pipeline
26.7.16 PPFs to include - Pipeline
26.7.17 Buttons at Bottom of Panel - Pipeline

26.7.1 General Information on Pipeline Long Section Plots

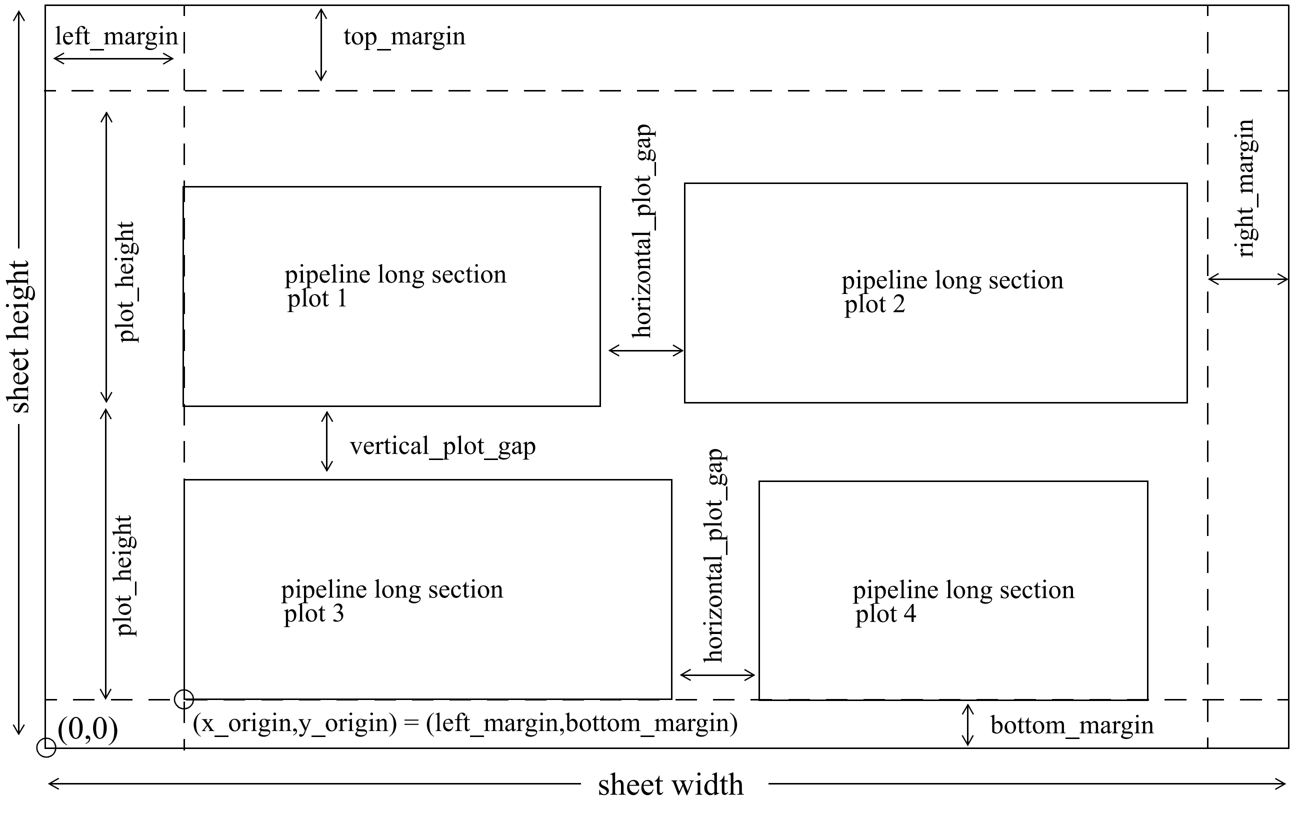
The pipeline plot makes special long section plots for a **network** of pipeline strings.

The pipeline long section plot will break an individual plot up if it doesn't fit across the sheet. There can be one or more rows of plot on the same sheet.

The top row is done first, followed by the second top row, then the third and so on until the bottom row. If there is only one row, it is considered to be the bottom row.

When a sheet is full, a follow on sheet is created.

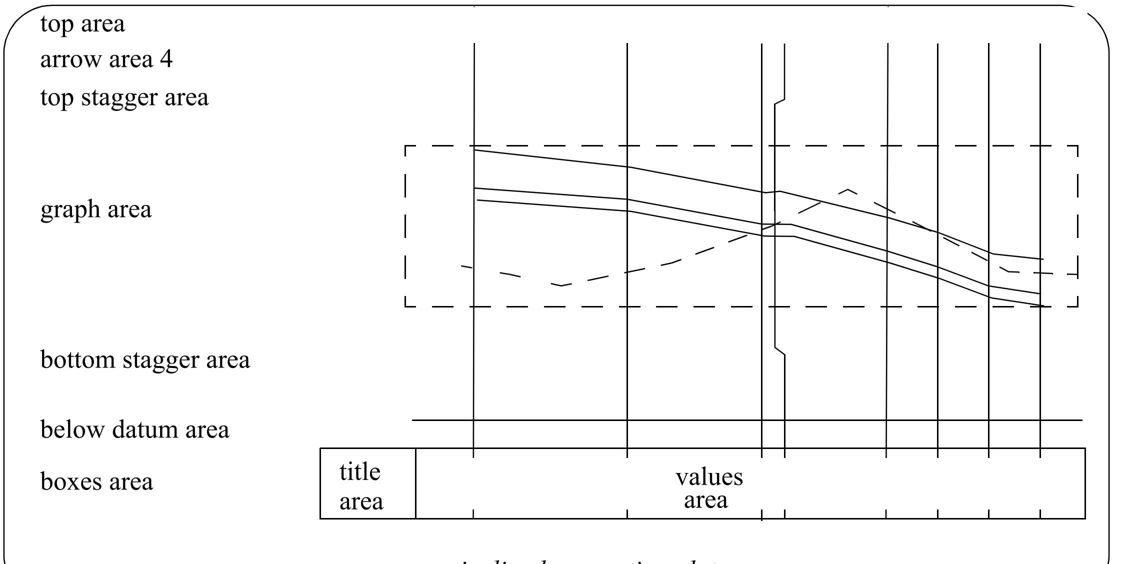
As soon as one pipeline string is completed, the next pipeline string in the network model is plotted beginning on the same row as the previous pipeline string and with a user given horizontal gap between the plots. If there is not enough room on the row to start the next plot, it will begin on a new row.



Each pipeline long plot consists of seven areas. From the bottom up, they are

1. The **boxes area** is where the chainages and various values for the pipeline strings are labelled.
2. The **below datum area** is a region between the boxes area and the datum line.
3. The **bottom stagger area** is where the upright line staggers occur before going up from the boxes area to the graph area.
4. The **graph area** is the area where the actual plots of the strings are drawn.
5. The **top stagger area** is where the upright line staggers occur above the graph area.
6. The **arrow 4 area** is an extension of the uprights above the top stagger area to allow for the drawing of arrows where the arrows go between the staggered uprights above the graph area.
7. The **top area** is an annotation area above the arrow 4 area (the top of the plot) and is used for

pegs and deflection angles.



pipeline long section plot

Apart from information labelled in the boxes and top areas, the pipeline long section plot can place arrows between ips for other information such as

(a) pipe grade
(b) vertical geometry
(c) horizontal geometry

Also the chainages where services cross the pipeline line are automatically labelled.

For each pipeline plot, the labelling of pipe invert levels, depth to invert, pipe grade and natural surface level are done at the chainages:

(a) pegs given by a user given peg interval
(b) horizontal and vertical defection points
(c) points in the specials model
(d) crossing services

All the required parameters for controlling the pipeline long section plot are set up in the ***Pipeline Plot PPF Editor*** and will be described in detail in the following sections.

Please continue to the next section 26.4.2 Section Long Plot - Front Page or return to 26.4 Long Plot PPF Editor.

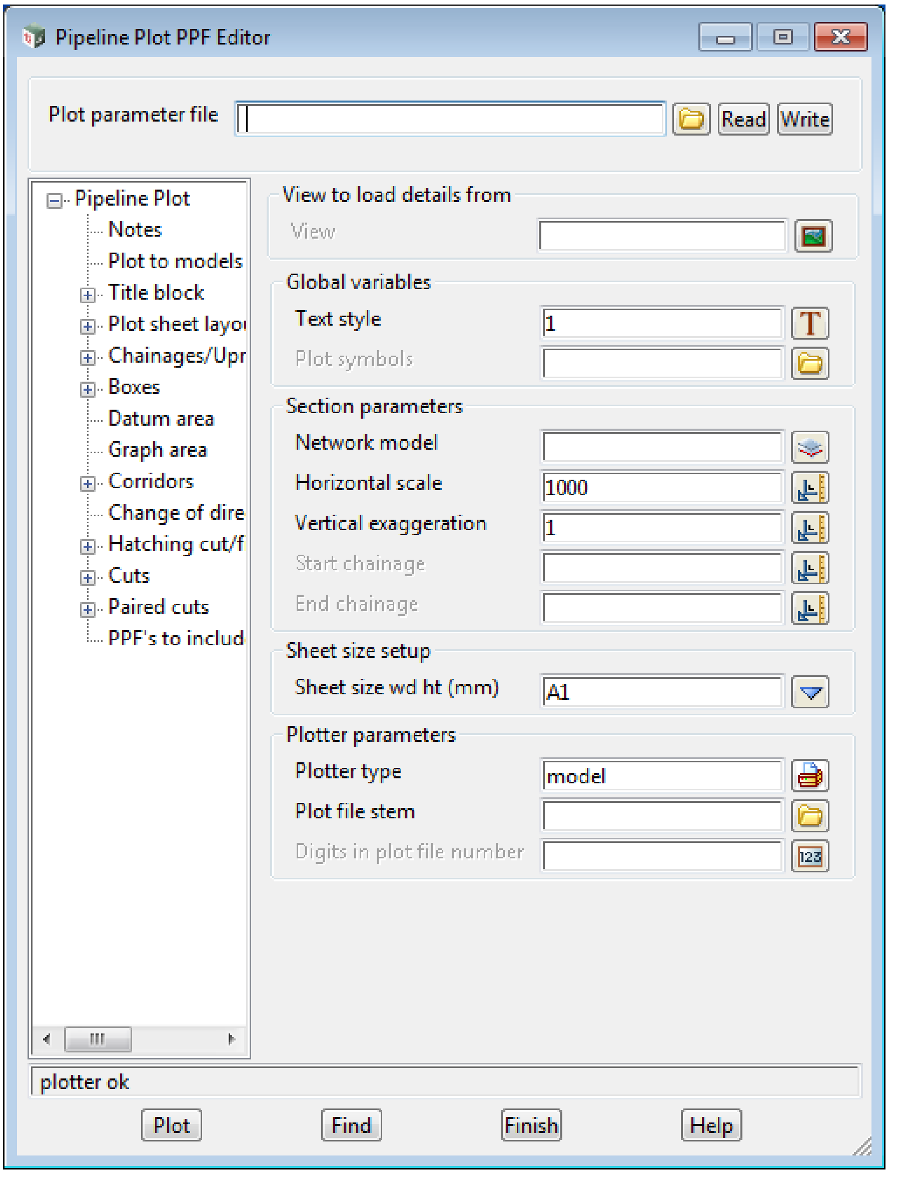
26.7.2 Pipeline Longsection Plot - Front Page

The pipeline long plot itself consists of the three regions - boxes, datum and graph areas.

The **boxes area** is where the titles and the chainage values and the heights/depths for the strings drawn on the long plot are labelled.

The **datum area** is the region between the boxes area and the graph area.

The **graph area** is the area where the actual plots of the strings are drawn.



Section: Plot parameter file

*a **plot parameter file** can be used to load values into the fields of the PPF Editor, or as a file to write out all current values in a PPF Editor to.*

This section is documented for all the PPF Editors in 26.2.1 Plot Parameter File.

Section: Section: View to load details from AND Global variables

*a **section view** can be selected to load certain values into fields of the PPDF Editor. For example,* *Vertical exaggeration and Corridor models. And there are variables to use if some values aren't given.*

These sections are documented for all the PPF Editors in 26.2.2 View to Load and Global Variables.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Section parameters

Panel Field	Parameter name	Type	Pop-Up
Network model	network_model	model box	available models

model containing the pipeline strings to draw in long section on the plot.

Panel Field	Parameter name	Type	Pop-Up
Horizontal Scale	scale	real box	

horizontal scale to be used for the long section plots

Panel Field	Parameter name	Type	Pop-Up
Vertical exaggeration	vertical_exaggeration	real box	

vertical scale to be used for the long section plots

Panel Field	Parameter name	Type	Pop-Up
Start chainage	start_chainage	real box	

start chainage of the string to start the long section plots at

Panel Field	Parameter name	Type	Pop-Up
End chainage	end_chainage	real box	

end chainage of the string to end the long section plots at.

Section: Sheet size setup and Plotter parameters

these sections define the size of the "paper" to plot on, the type of plotter to use and the naming to use for the plot files.

These sections are documented for all the PPF Editors in 26.2.3 Sheet Size and Plotter Parameters.

Please continue to the next section 26.7.3 Notes - Pipeline.

26.7.3 Notes - Pipeline

This is documented for all the PPF Editors in 26.2.4 Notes.

Continue to the next section 26.7.4 Plot to Models - Pipeline.

26.7.4 Plot to Models - Pipeline

This is documented for all the PPF Editors in 26.2.5 Plot to models.

Continue to the next section 26.7.5 Title Block - Pipeline.

26.7.5 Title Block - Pipeline

This is documented for all the PPF Editors in 26.2.6.1 Title Block Section in PPF Editors.

For more general information about a title block, see 26.2.6 Title Block.

Continue to the next section 26.7.6 Plot Sheet Layout - Pipeline.

26.7.6 Plot Sheet Layout - Pipeline

The information on this node is the same as that for Drainage Long Sections - see 26.5.6 Plot Sheet Layout

Continue to the next section 26.7.7 Chainage/Uprights - Pipeline.

26.7.7 Chainage/Uprights - Pipeline

Continue to the next section 26.7.8 Boxes - Pipeline.

26.7.8 Boxes - Pipeline

Continue to the next section 26.7.9 Datum Value - Pipeline.

26.7.9 Datum Value - Pipeline

Continue to the next section 26.7.10 Graph Area - Pipeline.

26.7.10 Graph Area - Pipeline

Continue to the next section 26.7.11 Corridors - Pipeline.

26.7.11 Corridors - Pipeline

A corridor around the primary string is defined by giving a left and right corridor width.

Any string in a model added to the section view is checked to see if it appears in the corridor, and if it does, it is drawn on the cross-section plot.

To be drawn, strings do not have to cross the primary string, but just be in the corridor.

This is documented for all the PPF Editors in 26.2.7 Corridors.

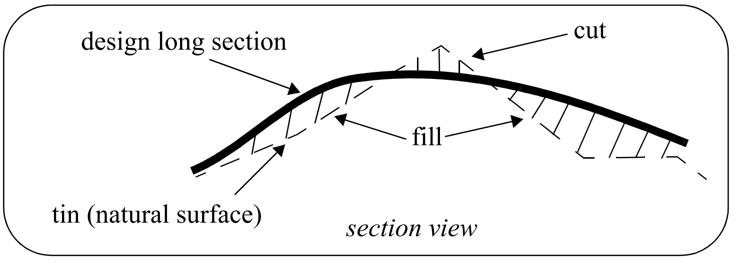
Continue to the next section 26.7.12 Change_of_Direction_at_Intersection_Point.

26.7.12 Change_of_Direction_at_Intersection_Point

Continue to the next section 26.7.13 Hatching Cut/Fill - Pipeline.

26.7.13 Hatching Cut/Fill - Pipeline

This option is used to hatch cut and/or fill areas between sets of tins.



The **Hatching Cut/Fill** section is common to the **PPF Editors** and is fully documented in 26.2.8 Hatching Cut/Fill.

Section: Hatching cut/fill - Tin parameters - *see* 26.2.8 Hatching Cut/Fill

Section: Hatching cut/fill - Cut parameters - *see* 26.2.8.1 Hatching Cut/Fill - Cut

Section: Hatching cut/fill - Fill parameters- *see* 26.2.8.2 Hatching Cut/Fill - Fill

Continue to the next section 26.7.14 Cuts - Pipeline.

26.7.14 Cuts - Pipeline

The cuts that the primary string (drainage or sewer line) makes though strings in any user specified model can be automatically labelled on the drainage/sewer long section plots.

The **Cuts** section is common to the **PPF Editors** and is fully documented in 26.2.9 Cuts.

Section: Cuts - Model/Name mask parameters - *see* 26.2.9 Cuts.

Section: Cuts - Chainage parameters - *see* 26.2.9.3 Cuts - Chainage - Long Sections Only

Section: Cuts - Height parameters - *see* 26.2.9.4 Cuts - Heights

Section: Cuts - Diameter parameters - *see* 26.2.9.5 Cuts - Diameters

Section: Cuts - Label parameters - *see* 26.2.9.6 Cuts - Labels

Section: Cuts - Symbol parameters - *see* 26.2.9.7 Cuts - Symbols

Continue to the next section 26.7.15 Paired Cuts - Pipeline.

26.7.15 Paired Cuts - Pipeline

Paired cuts uses pairs of strings and where both strings cut the primary string, the cuts on the long plot can be labelled with information such as

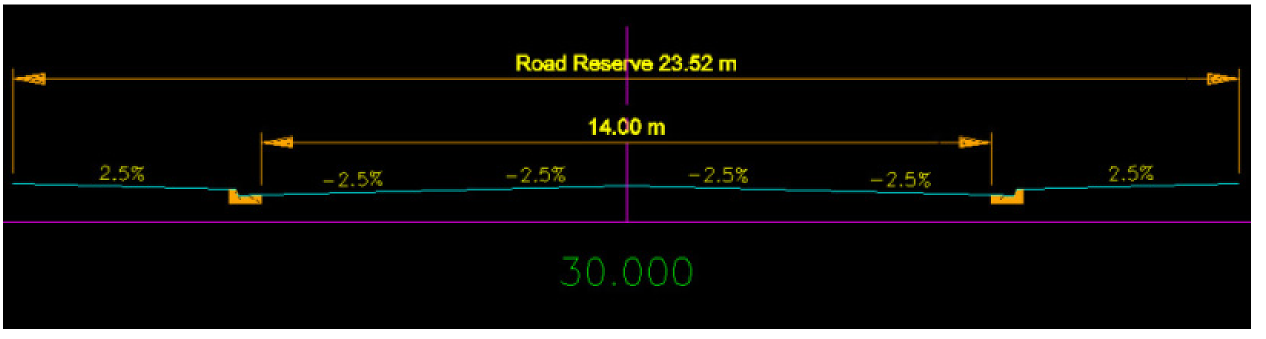
(a) the name of the first and second cut strings

(b) attributes from the first and second cut string

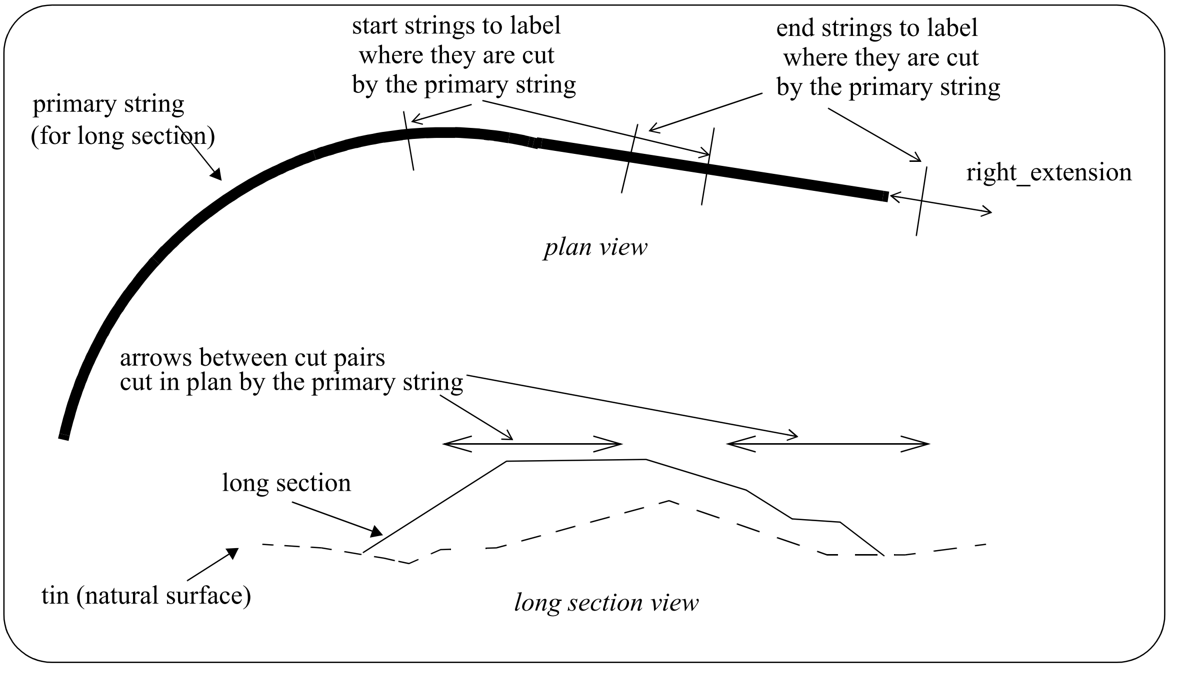
(c) plan distance between the two cuts of the pair

(d) 3d length between the two cuts of the pair

(e) the chainge on the long section of the first and second cuts of the pair



Symbols can be drawn at the chainages of the first and second cuts (at a height specified when defining the Cut sets), and a line drawn between the symbols. Using both the line and a symbol of an arrow head makes an arrow between the two cuts.



The method for specifying which strings are to be checked for paired cuts is by first specifying the **models** (using wild cards and characters) that contains the strings, and then a **start name mask** to select the all the strings that are to be the first strings in a cut pair, and an **end name mask** to select all the strings that are the second strings in a cut pair.

For a long section, all the cuts of the selected **start** strings are found and the cuts ordered by the chainage of the cut with the primary string of the long section. Then all the cuts of the selected **end** strings are found and the cuts ordered by the chainage of the cut with the primary string of the long section.

The start cuts are then processed and each start cut is paired with the next end cut with a larger chainage than the start cut. It is possible that there are end cuts before the first start cut (orphaned end cuts) and start cut with no following end cut (orphaned start cuts).

Up to twenty five different sets of models and name masks can be used so that different paired cut sets can be labelled in different ways.

26.7.15.1 Paired Cuts - Front Page

The information on this section is the same as that for Long Sections, please see 26.4.23.1 Paired Cuts (Long Section) - Front Page

26.7.15.1.1 Paired Cuts - Lines and Symbols

The information on this section is the same as that for Long Sections, please see 26.4.23.2 Paired Cuts (Long Section) - Lines and Symbols

26.7.15.1.2 Paired Cuts - Labels

The information on this section is the same as that for Long Sections, please see 26.4.23.3 Paired Cuts (Long Section) - Labels

26.7.15.1.3 Paired Cuts - Start Chainage

The information on this section is the same as that for Long Sections, please see 26.4.23.4 Paired Cuts (Long Section) - Start Chainage

26.7.15.1.4 Paired Cuts - End Chainage

The information on this section is the same as that for Long Sections, please see 26.4.23.5 Paired Cuts (Long Section) - End Chainage

Continue to the next section 26.7.16 PPFs to include - Pipeline.

26.7.16 PPFs to include - Pipeline

This is documented for all the PPF Editors in 26.2.10 PPFs To Include.

Continue to the next section 26.7.17 Buttons at Bottom of Panel - Pipeline.

26.7.17 Buttons at Bottom of Panel - Pipeline

This is documented for all the PPF Editors in 26.2.2 View to Load and Global Variables.

Return to 26.7 Pipeline Plot PPF Editor.

26.8 Drainage Plan Plot PPF Editor

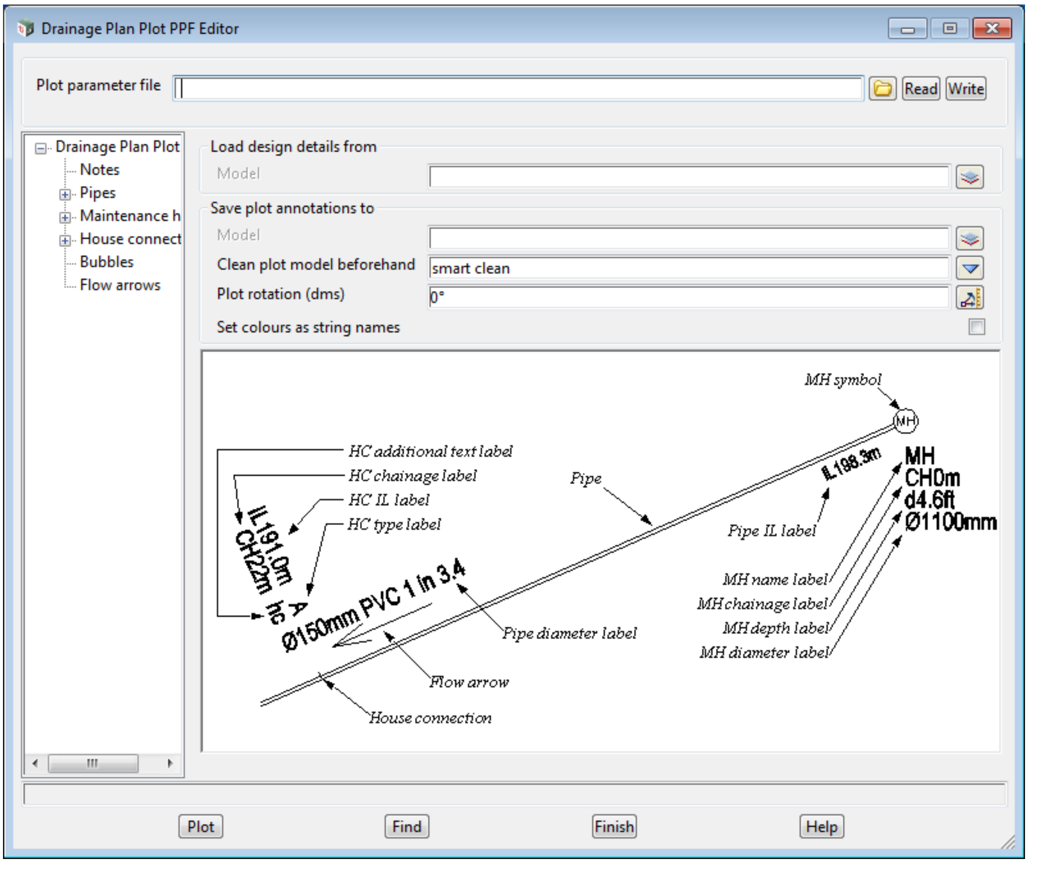
Position of option on menu: **Plot => Plot and PPF Editors =>Drainage Plan**

The **Drainage Plan Plot PPF Editor** is for creating and/or editing a (binary) drainage and sewer plan plot ppf file and for creating a drainage and/or sewer plan annotation overlay model. A text version of the file is also produced.

Note: Binary and text PPFs are stored *within* the project (not in the folder containing the project).

On selecting the **Drainage plan** option, the **Drainage Plan Plot PPF Editor** panel is displayed.

The plot parameters for controlling the cross section plots are accessed by expanding to the appropriate node in the ***Drainage Plan Plot*** tree (click on the **+** to expand to node or **-** to collapse the node) and then clicking on the required node, and the required information to fill in is displayed on the right hand side of the panel.

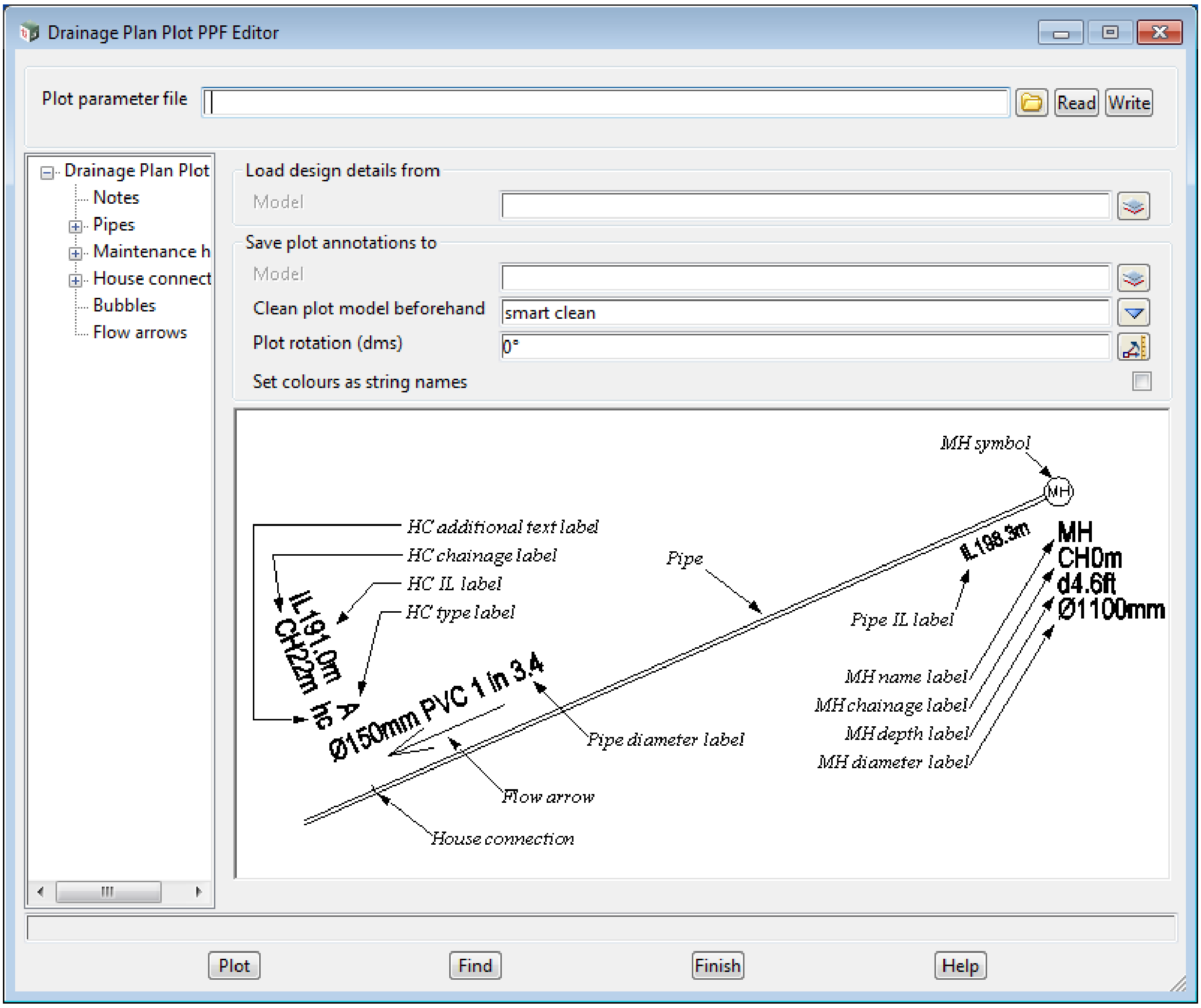


For information on all the different nodes see:

[26.8.1 Drainage Plan Plot - Front Page](#)
[26.8.2 Notes - Drainage Plan Plot](#)
[26.8.3 Pipes](#)
[26.8.4 Maintenance Holes](#)
[26.8.5 House Connections](#)
[26.8.6 Bubbles](#)
[26.8.7 Flow Arrows](#)
[26.8.8 Buttons at Bottom of Panel](#)

26.8.1 Drainage Plan Plot - Front Page

***Note**: When creating Drainage Plan Plot overlays, all distances and sizes (including text sizes, offsets, rises, etc) are specified in world units.*



Section: Plot parameter file

*a **plot parameter file** can be used to load values into the fields of the PPF Editor, or as a file to write out all current values in a PPF Editor to.*

This section is documented for all the PPF Editors in 26.2.1 Plot Parameter File.

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Load design details from

Model	model box	model	available models

model of drainage design strings to create Annotion for.

Section: Save plot annotations to

Model	model box	model	available models

model to write the drainage annotations to.

Clean plot model beforehand	plot_model_clean	choice box	smart clean

full clean

determines how to clean (delete the elements in) the output model, if it exists, before generating the plot annotations.

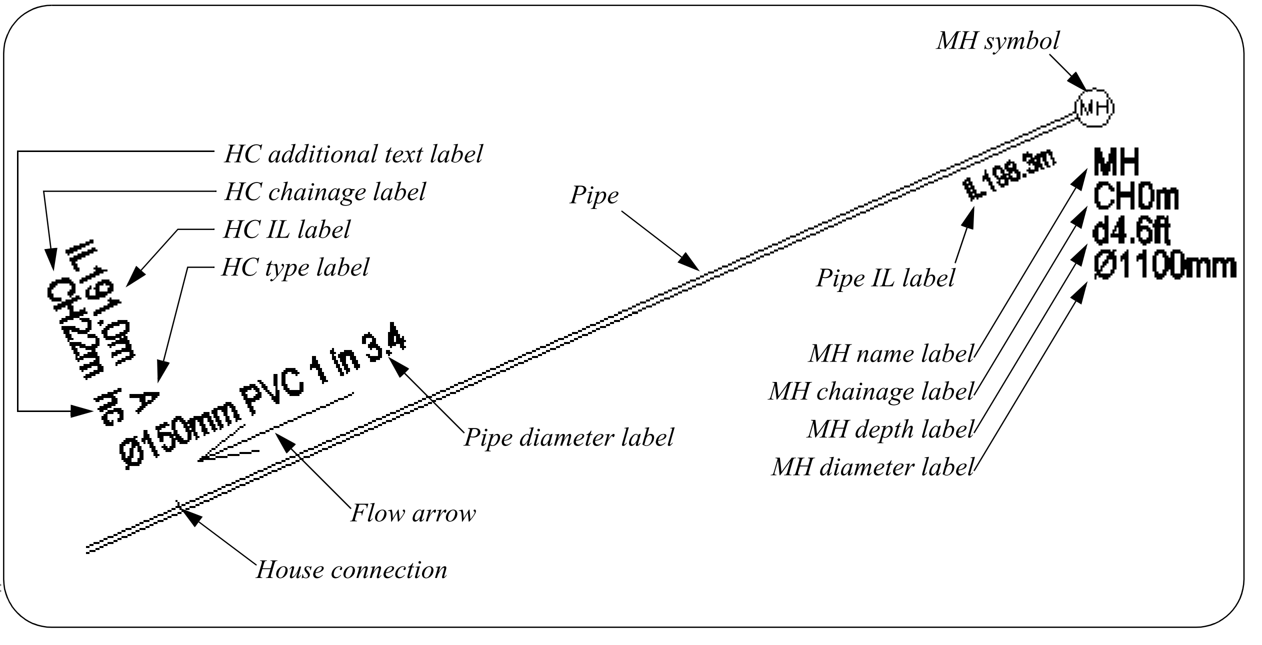
For **smart clean**, *any existing text, flow arrow and bubble annotation elements found that match the annotations about to be generated, will retain their pre-existing positions.*

For **full clean**, *no information about the pre-existing plot annotation is used, and the new plot annotation is created from scratch.*

Plot rotation (dms) plot_rotation measure box At Point, Point to Point, String from Point, String to Point

Set colours as string names **naming_mode** tick box

if ticked, the names all created plot annotation strings will be their colour names. This will override any user specified string names defined in other sections.



Please continue to the next section 26.8.2 Notes - Drainage Plan Plot.

26.8.2 Notes - Drainage Plan Plot

This is documented for all the PPF Editors in 26.2.4 Notes.

Continue to the next section 26.8.3 Pipes or return to 26.8 Drainage Plan Plot PPF Editor

26.8.3 Pipes

The pipes in a plan of a drainage network are often not drawn as they appear but with different linestyles *etc* to represent different pipe diameters *etc*.

The pipe diameter, invert levels and attributes can all be labelled in plan.

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Pipe representation

Pipe string level mode pipe_level_mode choice box pipe inverts, pipe obverts
mh cover levels, pipe hgl,
normal flow depth,
critical flow depth
zero

.

Pipe string vertical offset pipe_vertical_offset measure box available measures

the distance.

Draw multi pipes pipe_multi tick box

*if **ticked**,*

Pipe Diameters Label grid

The fields in this section are grid column fields that may each have up to 20 rows (sets) defined.

All pipes with diameters that fall between the minimum diameter **Min pipe diam** *and the maximum diameter* **Max pipe diam** *for a row (set), will be represented according to the values of the other fields in this row. Note that the Pipe diameters are defined in the drainage design strings.*

Min pipe diam (m) pipe_dia_min_n input

minimum pipe diameter in metres for the nth row of pipe representations.

Max pipe diam (m) pipe_dia_max_n input

maximum pipe diameter in metres for the nth row of pipe representations.

All pipes with diameters that fall between the min and max diameters of this row, will be represented according to the values of the other fields in this row.

Pipe type mask text box

Representation mode pipe_dia_mode choice box none
linestyle
pipe
linestyle and pipe

determines whether to represent pipes as linestyles and/or solid pipes for the nth row of pipe representations.

Trim mode trim_edge_mode choice box don't trim strings
trim pipes
trim linestyles
trim pipes and linestyles

determines whether to trim pipe and/or linestyle strings around maintenance holes for the nth row of pipe representations.

Multi-pipe spacing real box

the distance to space the pipes when multiple pipes are drawn.

Colour pipe_dia_colour_n colour box available colours

colour to use for the nth row of pipe representations.

Linestyle	pipe_dia_linestyle_n	linestyle box	available linestyles

linestyle to use for the nth row of pipe representations.

Linestyle string name	pipe_dia_name_n	input	

string name of linestyles for the nth row of pipe representations.

Pipe string name	pipe_edge_name_n	input	

string name of pipes for the nth row of pipe representations.

Section: Pipe labels

Label Rise mode	rise_mode	choice box	rise from centre of pipe rise from edge of pipe

determines where all pipe label text rise values are measured from.

See

26.8.3.0.1 Pipes - Pipe Diameter Labels
26.8.3.0.2 Pipes - Pipe IL Labels
26.8.3.0.3 Pipes - Pipe Attribute Labels

Or return to 26.8 Drainage Plan Plot PPF Editor.

26.8.3.0.1 Pipes - Pipe Diameter Labels

The diameter of the pipes can be labelled in plan.

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: Pipe diameter labels

Draw diameter labels	pipe_label_draw	tick box	

if ***ticked****, the pipe diameters are labelled.*

Label mode	pipe_label_mode	choice box	none pre dia post pre dia post grade pre dia post type grade

determines what text is to be placed for the pipe diameter labels. Pipe types and grades can be included, in addition to diameters. Pipe diameters, types and grades are all defined in the drainage design strings.

Note: *if a pipe attribute of real type named* **width** *exists and is greater than zero, the pipe is considered to be rectangular and is labelled* <width x height>*, where the height takes the value of the pipe diameter.*

Text size	pipe_label_size	real box	

text size of the pipe diameter labels. The labels will only be created if the size is greater than zero.

Text offset	pipe_label_offset	real box	

distance to move the pipe diameter labels along the pipe, from the mid position of the pipe segment. Positive values are to the right.

Text rise	pipe_label_rise	real box	

distance to move the pipe diameter labels away from the pipe centreline or edge (depending on setting of rise_mode*). A negative value will place the labels on the other side of the pipes.*

Text colour	pipe_label_colour	colour box	available colours

colour of the pipe diameter labels.

Text style	pipe_label_textstyle	textstyle box	available fonts

font of the pipe diameter labels.

Pre text	pipe_label_pre_text	text box	

text to be placed before the diameter in the pipe diameter labels.

Post text	pipe_label_post_text	text box	

text to be placed after the diameter in the pipe diameter labels.

Units factor	pipe_label_factor	real box	

multiplier of diameter values. A Units factor of 1000 will result in the diameters being labelled in mm.

Decimal places	pipe_label_decimals	integer box	

number of decimal places in labelled diameter values.

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Text string name	pipe_label_name	name box	

string name of the pipe diameter labels.

Continue to the next section 26.8.3.0.2 Pipes - Pipe IL Labels section or return to 26.8.3 Pipes.

26.8.3.0.2 Pipes - Pipe IL Labels

The invert levels of both ends of the pipes can be labelled in plan.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Pipe IL labels

Draw IL labels pipe_text_draw_il tick box

*if **ticked**, the pipe invert levels are labelled.*

Text size pipe_text_size_il real box

text size of the pipe IL (Invert Level) labels. The labels will only be created if the size is greater than zero.

Text offset pipe_text_offset_il real box

distance to move the pipe IL labels along the pipe, from the ends of the pipe. Positive values are towards the mid position of the pipe segment.

Text rise pipe_text_rise_il real box

distance to move the pipe IL labels away from the pipe centreline or edge (depending on setting of rise_mode*). A negative value will place the labels on the other side of the pipes.*

Text colour pipe_text_colour_il colour box available colours

colour of the pipe IL labels.

Text style pipe_text_textstyle_il textstyle box available textstyles

textstyle of the pipe IL labels.

Pre text pipe_text_pre_text_il text box

text to be placed before the IL in the pipe IL labels.

Post text pipe_text_post_text_il text box

text to be placed after the IL in the pipe IL labels.

Units factor pipe_text_factor_il real box

multiplier of IL values. A Units factor of 3.281 will result in the ILs being labelled in feet.

Decimal places pipe_text_decimals_il integer box

number of decimal places in labelled IL values.

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Text string name pipe_text_name_il name box

text string name of the pipe IL labels.

Continue to the next section 26.8.3.0.3 Pipes - Pipe Attribute Labels section or return to 26.8.3 Pipes.

26.8.3.0.3 Pipes - Pipe Attribute Labels

The values of any pipe attributes can be labelled in plan.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Pipe attribute labels

Pipe attribute label grid

each row of the grid defines how a pipe attribute is labelled. There may be up to 20 rows.

Label pipe attribute tick box

*if **ticked**, the pipe attribute for this row will be labelled.*

*If **not ticked**, the pipe attribute for this will **not** be labelled.*

Attribute name text box

full path name of the pipe attribute for this row that may be labelled.

Text size pipe_attribute_text_size_n real box

text size of the pipe attribute for this row.

Text offset real box

distance to move the pipe attribute labels for this row along the pipe, from the ends of the pipe. Positive values are towards the mid position of the pipe segment.

Text rise real box

distance to move the pipe attribute labels for this row away from the pipe edge. A negative value will place the labels on the other side of the pipes.

Text colour colour box available colours

colour of the pipe attribute labels for this row.

Text style textstyle box available textstyles

textstyle of the pipe attribute labels for this row.

Pre text text box

text to be placed before the pipe attribute labels for this row.

Post text text box

text to be placed after the pipe attribute labels for this row.

Units factor real box

if the pipe attribute for this row is real or integer, it is multiplied by the **Units factor** *for this row.*

Decimal places **I** integer box

if the pipe attribute for this row is real, this is the number of decimal places in the pipe attribute label for this row.

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Text string name pipe_attribute_text_name_n name box

name of the text string of the pipe attribute label for this row.

Continue to the next section 26.8.4 Maintenance Holes or return to 26.8.3 Pipes or 26.8 Drainage Plan Plot PPF Editor

26.8.4 Maintenance Holes

The maintenance holes in a plan of a drainage network are often not drawn as they appear but with different symbols *etc* to represent different manhole types *etc*.

The manhole name, diameter, depth, chainage, attributes and setout points can all be labelled in plan.

The columns for the fields documented in the sections are for.

Panel Field | **Parameter name** | **Type** | **Pop-Up**

Section: Maintenance hole representation

MH symbol direction mode mh_symbol_dir_mode choice box
use symbol angle
left
right
average

determines what mode of direction is used to place the MH symbols.

MH symbol angle (dms) mh_symbol_angle angle

angle of MH symbol placement in degrees, minutes and seconds in hp notation. This is only used depending on the setting of mh_symbol_dir_mode.

Use road direction where possible tick box

Use MH label direction for Text mode tick box

Group symbols with text labels tick box

if ***ticked****,*

MH Grid

the rows in the grid contain information for placing symbols at maintenance holes. There can be up to 20 rows in the grid.

MH type mh_type_n MH cell available MH types

MH type that will be drawn by information in this row of the grid. All MHs of this type will be represented according to the values of the other fields in this row. MH types are defined in the drainage design strings.

MH mode mh_mode_n choice cell none
text
symbol
circle

determines whether a text string, a symbol, or a circle is to be used to represent the maintenance holes in this row.

Insertion point choice cell mh centre, setout point

where to place the representation of the maintenance hole.

MH Symbol or Text mh_symbol_n symbol cell available symbols

symbol or text string to use to represent the nth set of maintenance holes.

If mh_mode_n *is set to* **text***, a text string can be typed in.*
If mh_mode_n *is set to* **symbol***, the symbol name can be chosen from the pop-up list.*
If mh_mode_n *is set to* **circle***, then it is ignored.*

MH Symbol mirror image

.

Justification	mh_symbol_justify_n	choice cell	left bottom left middle left top centre bottom centre middle centre top right bottom right middle right top

justification mode of the MH text for this row of maintenance holes. It is relative to the centres of the MHs. This will only be relevant if mh_mode_n *is set to* **text**.

Size	mh_symbol_size_n	real cell	

size of the MH symbols or text strings for this row of maintenance holes. The symbols, circles or text strings will only be created if the size is greater than zero.

In the case of a circle, the size is determined to match the diameter of the MH.

Offset	mh_symbol_offset_n	real cell	

distance to move the MH symbols, circles or text strings to the left or right of the MH centres, for this row of maintenance holes. Positive values are to the right. The direction of movement is parallel to the angle of the MH symbols (see mh_symbol_dir_mode *and* mh_symbol_angle*).*

Rise	mh_symbol_rise_n	real cell	

distance to move the MH symbols, circles or text strings above or below of the MH centres, for this row of maintenance holes. Positive values are upwards. The direction of movement is perpendicular to the angle of the MH symbols (see mh_symbol_dir_mode *and* mh_symbol_angle*).*

Colour	mh_symbol_colour_n	colour box	available colours

colour of the MH symbols, circles or text strings, for this row of maintenance holes.

Text style	mh_symbol_textstyle_n	textstyle box	available fonts

font of the MH text for this row of maintenance holes. This will only be relevant if mh_mode_n *is set to **text**.*

MH Symbol string name	mh_symbol_name_n	text cell	

string name of the MH symbols, circles or text strings for this row of maintenance holes.

Section: Maintenance Hole Labels

MH label direction mode	mh_text_mode	choice box	use label angle use pipe angle

determines what angle is to be used to position the maintenance hole label text.

MH label angle	mh_text_angle	angle	

angle of MH label text in degrees, minutes and seconds using hp notation.

This is only used depending on the setting of mh_text_mode.

MH label justification	mh_text_justify	choice box	left bottom left middle left top centre bottom centre middle centre top right bottom right middle right top

justification mode of all MH label text strings, relative to the MH centres.

See

26.8.4.1 Maintenance Holes - MH Name Labels
26.8.4.2 Maintenance Holes - MH diameter Labels
26.8.4.3 Maintenance Holes - MH depth Labels
26.8.4.4 Maintenance Holes - MH Chainage Labels
26.8.4.5 Maintenance Holes - MH Attribute Labels
26.8.4.6 Maintenance Holes - MH Setout Points

Or return to 26.8 Drainage Plan Plot PPF Editor.

26.8.4.1 Maintenance Holes - MH Name Labels

The maintenance holes names are defined in the drainage strings.

The columns for the fields documented in the sections are for.

Panel Field	Parameter name	Type	Pop-Up

Section: MH name labels

*this section defines how a **MH name** is labelled.*

Panel Field	Parameter name	Type	Pop-Up
Draw name labels		tick box	

*if **ticked** the name of the maintenance hole is labelled.*

*If **not ticked** the name of the maintenance hole is not labelled.*

Panel Field	Parameter name	Type	Pop-Up
Text size	mh_text_size	real box	

*if **>0**, the text size of the MH name labels.*
*If **zero**, the MH name will not be labelled.*

Panel Field	Parameter name	Type	Pop-Up
Text offset	mh_text_offset	real box	

distance to move the MH name labels to the left or right of the MH centres. Positive values are to the right. The direction of movement is parallel to the angle of the text, specified by parameters mh_text_mode *and* mh_text_angle. *The justification of the text is specified by parameter* mh_text_justify.

Panel Field	Parameter name	Type	Pop-Up
Text rise	mh_text_rise	real box	

distance to move the MH name labels above or below the MH centres. Positive values are above. The direction of movement is perpendicular to the angle of the text, specified by parameters mh_text_mode *and* mh_text_angle. *The justification of the text is specified by parameter* mh_text_justify.

Panel Field	Parameter name	Type	Pop-Up
Text colour	mh_text_colour	colour	available colours

colour of the MH name labels.

Panel Field	Parameter name	Type	Pop-Up
Text style	mh_text_textstyle	textstyle	available textstyles

textstyle of the MH name labels.

Panel Field	Parameter name	Type	Pop-Up
Pre text	mh_text_pre	text box	

text to be placed before the MH name in the MH name labels.

Panel Field	Parameter name	Type	Pop-Up
Post text	mh_text_post	text box	

text to be placed after the MH name in the MH name labels.

Panel Field	Parameter name	Type	Pop-Up
Text string name	mh_text_name	text box	

text string name of the MH name labels.

Continue to the next section 26.8.4.2 Maintenance Holes - MH diameter Labels or return to 26.8.4 Maintenance Holes.

26.8.4.2 Maintenance Holes - MH diameter Labels

The diameter of each maintenance hole is defined in the drainage strings.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: MH diameter labels

*this section defines how a **MH diameter** is labelled.*

Draw diameter labels tick box

*if **ticked** the diameter of the maintenance holes are labelled.*

*If **not ticked** the diameter of the maintenance holes are not labelled.*

Sub Section: Only label if diameter is outside range:

Min mh_text_min_dia real box

Max mh_text_max_dia real box

range of MH diameters for which the MH diameter labels will not be created.
Both Min and Max are entered in metres.

Text size mh_text_size_dia real box

*if **>0**, the text size of the MH diameter labels.*
*If **zero**, the MH diameter will not be labelled.*

Text offset mh_text_offset_dia real box

distance to move the MH diameter labels to the left or right of the MH centres. Positive values are to the right. The direction of movement is parallel to the angle of the text, specified by parameters mh_text_mode *and* mh_text_angle*. The justification of the text is specified by parameter* mh_text_justify.

Text rise mh_text_rise_dia real box

distance to move the MH diameter labels above or below the MH centres. Positive values are above. The direction of movement is perpendicular to the angle of the text, specified by parameters mh_text_mode *and* mh_text_angle*. The justification of the text is specified by parameter* mh_text_justify.

Text colour mh_text_colour_dia colour box available colours

colour of the MH diameter labels.

Text style mh_text_textstyle_dia textstyle box available fonts

font of the MH diameter labels.

Pre text mh_text_pre_dia text box

text to be placed before the diameter in the MH diameter labels.

Post text mh_text_post_dia text box

text to be placed after the diameter in the MH diameter labels.

Units factor **mh**_text_factor_dia real box

multiplier of diameter values. A Units factor of 1000 will result in the diameters being labelled in mm.

Decimal places mh_text_decimals_dia integer box

number of decimal places in labelled diameter values.

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Text string name mh_text_name_dia text box

string name of the MH diameter labels.

Continue to the next section 26.8.4.3 Maintenance Holes - MH depth Labels or return to 26.8.4 Maintenance Holes.

26.8.4.3 Maintenance Holes - MH depth Labels

The depth of each maintenance hole is defined in the drainage strings.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: MH depth labels

*this section defines how a **MH depth** is labelled.*

Draw depth labels tick box

*if **ticked** the depth of the maintenance holes are labelled.*

*If **not ticked** the depth of the maintenance holes are not labelled.*

Text size mh_text_size_depth real box

*if **>0**, the text size of the MH depth labels.*
*If **zero**, the MH depths are not labelled.*

Text offset mh_text_offset_depth real box

distance to move the MH depth labels to the left or right of the MH centres. Positive values are to the right. The direction of movement is parallel to the angle of the text, specified by parameters mh_text_mode *and* mh_text_angle. *The justification of the text is specified by parameter* mh_text_justify.

Text rise mh_text_rise_depth real box

distance to move the MH depth labels above or below the MH centres. Positive values are above. The direction of movement is perpendicular to the angle of the text, specified by parameters mh_text_mode *and* mh_text_angle. *The justification of the text is specified by parameter* mh_text_justify.

Text colour mh_text_colour_depth colour box available colours

colour of the MH depth labels.

Text style mh_text_textstyle_depth textstyle available fonts

font of the MH depth labels.

Pre text mh_text_pre_depth text box

text to be placed before the depth in the MH depth labels.

Post text mh_text_post_depth text box

text to be placed after the depth in the MH depth labels.

Units factor mh_text_factor_depth real box

multiplier of depth values. A Units factor of 3.281 will result in the depths being labelled in feet.

Decimal places mh_text_decimals_depth integer box

number of decimal places in labelled depth values.

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Text string name mh_text_name_depth name box

string name of the MH depth labels.

Continue to the next section 26.8.4.4 Maintenance Holes - MH Chainage Labels or return to 26.8.4 Maintenance Holes.

26.8.4.4 Maintenance Holes - MH Chainage Labels

The chainage of each maintenance hole is defined in the drainage strings.

The columns for the fields documented in the sections are for.

Panel Field | **Parameter name** | **Type** | **Pop-Up**

Section: MH chainage labels

*this section defines how a **MH chainage** is labelled.*

Draw chainage labels | | tick box |

*if **>0**, the text size of the MH chainage labels.*
*If **zero**, the MH depths are not labelled.*

Chainage mode | mh_text_mode_ch | choice box | running chainage, previous chainage, next chainage

determines what the chainage is that is labelled at the MHs.

If **running chainage***, the chainage is given with reference to the start of the drainage design string.*

If **previous chainage***, the chainage is from the previous maintenance hole.*

If **next chainage***, the chainage is from the next maintenance hole.*

Text size | mh_text_size_ch | real box |

*if **>0**, the text size of the MH chainage labels.*
*If **zero**, the MH depths are not labelled.*

Text offset | mh_text_offset_ch | real box |

distance to move the MH chainage labels to the left or right of the MH centres. Positive values are to the right. The direction of movement is parallel to the angle of the text, specified by parameters mh_text_mode *and* mh_text_angle*. The justification of the text is specified by parameter* mh_text_justify.

Text rise | mh_text_rise_ch | real box |

distance to move the MH chainage labels above or below the MH centres. Positive values are above. The direction of movement is perpendicular to the angle of the text, specified by parameters mh_text_mode *and* mh_text_angle*. The justification of the text is specified by parameter* mh_text_justify.

Text colour | mh_text_colour_ch | colour box | available colours

colour of the MH chainage labels.

Text style | mh_text_textstyle_ch | textstyle | available fonts

font of the MH chainage labels.

Pre text | mh_text_pre_ch | text box |

text to be placed before the chainage in the MH chainage labels.

Post text | mh_text_post_ch | text box |

text to be placed after the chainage in the MH chainage labels.

Units factor | mh_text_factor_ch | real box |

multiplier of chainage values. A Units factor of 3.281 will result in the chainages being labelled in feet.

Decimal places | mh_text_decimals_ch | integer box |

number of decimal places in labelled chainage values.

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Text string name | mh_text_name_ch | name box |

string name of the MH chainage labels.

Continue to the next section 26.8.4.5 Maintenance Holes - MH Attribute Labels or return to 26.8.4 Maintenance Holes.

26.8.4.5 Maintenance Holes - MH Attribute Labels

The values of any maintenance hole attributes can be labelled in plan.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: MH attribute labels

MH attribute label grid

each row of the grid defines how a MH attribute is labelled. There may be up to 20 rows.

Label MH attribute tick box

*if **ticked**, the MH attribute for this row will be labelled.*

*If **not ticked**, the MH attribute for this row will **not** be labelled.*

Attribute name text box

full path name of the MH attribute for this row that may be labelled.

Text size real box

text size of the MH attribute for this row.

Text offset real box

distance to move the MH attribute labels for this row to the right of the centre of the MH.

Text rise real box

distance to move the MH attribute labels for this above the centre of the MH.

Text colour colour box available colours

colour of the MH attribute labels for this row.

Text style textstyle box available textstyles

textstyle of the MH attribute labels for this row.

Pre text text box

text to be placed before the MH attribute labels for this row.

Post text text box

text to be placed after the MH attribute labels for this row.

Units factor real box

if the MH attribute for this row is real or integer, it is multiplied by the **Units factor** *for this row.*

Decimal places integer box

if the MH attribute for the this row is real, this is the number of decimal places in the MH attribute label for this row.

*If > 0, trailing zeros are **removed** after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Text string name name box

name of the text string of the MH attribute label for this row.

Continue to the next section 26.8.4.6 Maintenance Holes - MH Setout Points or return to 26.8.4 Maintenance Holes.

26.8.4.6 Maintenance Holes - MH Setout Points

Symbols can be placed at the setout points for each maintenance hole.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: MH setout points

Draw setout points tick box

*if **ticked**, symbols are drawn at the MH set out points.*
*If **not ticked**, symbols are **not** drawn at the MH set out points.*

Setout symbol symbol box available symbols

the symbol to draw at each setout point.

Symbol size real box

size of the symbol for the setout points.

Setout points string name name box

name of the string that has the symbol.

Include MH name in string name tick box

*if **ticked**, the MH name is included in the setout points string name.*
*If **not ticked**, the MH name is **not** included in the setout points string name.*

Continue to the next section 26.8.5 House Connections or return to 26.8 Drainage Plan Plot PPF Editor

26.8.5 House Connections

The maintenance holes in a plan of a drainage network are often not drawn as they appear but with different symbols *etc* to represent different manhole types *etc*.

The manhole name, diameter, depth, chainage, attributes and setout points can all be labelled in plan.

The columns for the fields documented in the sections are for.

Panel Field **Parameter name** **Type** **Pop-Up**

Section: Maintenance hole representation

Panel field **Parameter name** **Type** **Pop-Up**

Section: Maintenance hole representation

Draw house connections **hc_draw_mode** **tick box**

HC linestyle **hc_line_style** **linestyle** **available linestyles**

linestyle with which to represent the house connections to the pipes. The HCs will be represented as lines perpendicular to the pipe, from the pipe centreline, to the sides of the pipe that the HCs are on.

HC colour **hc_colour** **colour** **available colours**

colour of the house connection linestyles

HC string name hc_line_name input

string name of the HC lines.

The remaining fields in this section are grid column fields that may each have up to 6 sets (rows) defined.

Panel field **Parameter name** **Type** **Pop-Up**

HC Type **hc_types_n** **input**

*HC type of the nth set of house connections. All HCs of this type will be represented according to the values of the other fields in this set (row). HC types are defined in the drainage design strings. There are only 6 different HC types used in **12d Model**.*

HC Type label **hc_type_text_n** **input**

text label to be associated with the HC type of the nth set of house connections.

Label HC Name and Type **hc_type_tp_n** **tick box**

label the names and types of the HCs of the nth set of house connections. Depending on the HC type, it may not be desired to label the type.

Label HC IL **hc_type_il_n** **tick box**

label the ILs of the HCs of the nth set of house connections. Depending on the HC type, it may not make sense to label the IL.

Label HC Chainage **hc_type_ch_n** **tick box**

label the chainages of the HCs of the nth set of house connections. Depending on the HC type, it may not make sense to label the chainages.

Label HC Additional Text **hc_type_at_n** **tick box**

label the Additional text of the HCs of the nth set of house connections. Depending on the HC type, it may not be desired to label the Additional text.

See

26.8.5.1 House Connections - HC Name and Type Labels
26.8.5.2 House Connections - HC IL Labels
26.8.5.3 House Connections - HC Chainage Labels

26.8.5.4 House Connections - HC Additional Text Labels

Or return to 26.8 Drainage Plan Plot PPF Editor.

26.8.5.1 House Connections - HC Name and Type Labels

Panel field	Parameter name	Type	Pop-Up
Label mode	**hc_type_mode**	**choice box**	**available choices**
Text size	**hc_type_size**	**input**	

text size of the HC type labels. The labels will only be created if the size is greater than zero and the parameter hc_type_tp_n *is set to* **on** *for the nth set of house connections. The HC type labels are associated with the HC types are defined by the parameter set* hc_type_text_n.

Text offset	**hc_type_offset**	**input**	

distance to move the HC type labels away from the pipe centreline. Negative values are converted to positive. HC label text is always placed perpendicular to the pipe (or parallel to the HC line) and on the same side of the pipe as the HC. The text will be automatically left justified or right justified, depending on the side.

Text rise	**hc_type_rise**	**input**	

distance to move the HC type labels above or below the HC lines. Positive values are above. The direction of movement is parallel to the pipe (or perpendicular to the HC line). The text will be automatically bottom justified.

Text colour	**hc_type_colour**	**colour**	**available colours**

colour of the HC type labels.

Text style	**hc_type_textstyle**	**textstyle**	**available fonts**

font of the HC type labels.

Text string name	hc_type_name	input	

string name of the HC type labels.

Continue to the next section 26.8.5.2 House Connections - HC IL Labels or return to 26.8.5 House Connections.

26.8.5.2 House Connections - HC IL Labels

Panel field	Parameter name	Type	Pop-Up
Text size	**hc_il_size**	**input**	

text size of the HC IL (Invert Level) labels. The labels will only be created if the size is greater than zero and the parameter hc_type_il_n *is set to* **on** *for the nth set of house connections.*

Text offset	**hc_il_offset**	**input**	

distance to move the HC IL labels away from the pipe centreline. Negative values are converted to positive. HC label text is always placed perpendicular to the pipe (or parallel to the HC line) and on the same side of the pipe as the HC. The text will be automatically left justified or right justified, depending on the side.

Text rise	**hc_il_rise**	**input**	

distance to move the HC IL labels above or below the HC lines. Positive values are above. The direction of movement is parallel to the pipe (or perpendicular to the HC line). The text will be automatically bottom justified.

Text colour	**hc_il_colour**	**colour**	**available colours**

colour of the HC IL labels.

Text style	**hc_il_textstyle**	**textstyle**	**available fonts**

font of the HC IL labels.

Pre text	**hc_il_pre**	**input**	

text to be placed before the IL in the HC IL labels.

Post text	**hc_il_post**	**input**	

text to be placed after the IL in the HC IL labels.

Units factor	**hc_il_factor**	**input**	

multiplier of IL values. A Units factor of 3.281 will result in the ILs being labelled in feet.

Decimal places	**hc_il_decimals**	**input**	

number of decimal places in labelled IL values.

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Text string name	hc_il_name	input	

string name of the HC IL labels.

Continue to the next section 26.8.5.3 House Connections - HC Chainage Labels or return to 26.8.5 House Connections.

26.8.5.3 House Connections - HC Chainage Labels

Panel field	Parameter name	Type	Pop-Up
Chainage mode	**hc_chain_mode**	**choice box**	**hc lot offset** **running chainage** **previous chainage** **next chainage**

determines what chainage values will be labelled at the HCs. The **running chainage** *option refers to the start of the drainage design string. The* **previous chainage** *and* **next chainage** *options refer to the previous and next MHs respectively. The* **hc lot offset** *option is a special one that looks for a user-defined attribute called* "<hc_name> hc lot offset" *in the house connections along the drainage design strings. If the attribute is found, its value is set as the HC chainage label. If the attribute is not found, the value of the HC chainage label is set to a default HC Lot Offset of 5.0.*

Text size — **hc_chain_size** — **input**

text size of the HC chainage labels. The labels will only be created if the size is greater than zero and the parameter hc_type_ch_n *is set to* **on** *for the nth set of house connections.*

Text offset — **hc_chain_offset** — **input**

distance to move the HC chainage labels away from the pipe centreline. Negative values are converted to positive. HC label text is always placed perpendicular to the pipe (or parallel to the HC line) and on the same side of the pipe as the HC. The text will be automatically left justified or right justified, depending on the side.

Text rise — **hc_chain_rise** — **input**

distance to move the HC chainage labels above or below the HC lines. Positive values are above. The direction of movement is parallel to the pipe (or perpendicular to the HC line). The text will be automatically bottom justified.

Text colour — **hc_chain_colour** — **colour** — **available colours**

colour of the HC chainage labels.

Text style — **hc_chain_textstyle** — **textstyle** — **available fonts**

font of the HC chainage labels.

Pre text — **hc_chain_pre** — **input**

text to be placed before the chainage in the HC chainage labels.

Post text — **hc_chain_post** — **input**

text to be placed after the chainage in the HC chainage labels.

Units factor — **hc_chain_factor** — **input**

multiplier of chainage values. A Units factor of 3.281 will result in the chainages being labelled in feet.

Decimal places — **hc_chain_decimals** — **input**

number of decimal places in labelled chainage values.

If > 0, trailing zeros are ***removed*** *after the decimal point.*

If <0, the absolute value is taken as the number of decimal places to report i.e. no trailing zeros are removed. For example -3 means that there is always 3 figures after the decimal place.

Text string name — hc_chain_name — input

string name of the HC chainage labels.

Continue to the next section 26.8.5.4 House Connections - HC Additional Text Labels or return to 26.8.5 House Connections.

26.8.5.4 House Connections - HC Additional Text Labels

Panel field	Parameter name	Type	Pop-Up

Additional text **hc_text** **input**

the additional text to place near the HCs.

Text size **hc_text_size** **input**

text size of the HC additional text labels. The labels will only be created if the size is greater than zero and the parameter hc_type_at_n *is set to* **on** *for the nth set of house connections.*

Text offset **hc_text_offset** **input**

distance to move the HC additional text labels away from the pipe centreline. Negative values are converted to positive. HC label text is always placed perpendicular to the pipe (or parallel to the HC line) and on the same side of the pipe as the HC. The text will be automatically left justified or right justified, depending on the side.

Text rise **hc_text_rise** **input**

distance to move the HC additional text labels above or below the HC lines. Positive values are above. The direction of movement is parallel to the pipe (or perpendicular to the HC line). The text will be automatically bottom justified.

Text colour **hc_text_colour** **colour** **available colours**

colour of the HC additional text labels.

Text style **hc_text_textstyle** **textstyle** **available fonts**

font of the HC additional text labels.

Text string name hc_text_name input

string name of the HC additional text labels.

Continue to the next section 26.8.6 Bubbles or return to 26.8 Drainage Plan Plot PPF Editor

26.8.6 Bubbles

Panel field	Parameter name	Type	Pop-Up

Section: Bubbles

Panel field	Parameter name	Type	Pop-Up
Draw bubbles	**bubble_draw_mode**	**tick box**	

draw bubbles (circles) around the MH name labels.

Panel field	Parameter name	Type	Pop-Up
Bubble colour	**bubble_colour**	**colour box**	**available colours**

colour of the bubbles.

Panel field	Parameter name	Type	Pop-Up
Bubble factor	**bubble_factor**	**input**	

the initial radius of the bubble is calculated as half the text length of the MH name label. The bubble factor is used to multiply the initial radius to give the final bubble radius. Factors greater than 1 will increase the bubble size.

Panel field	Parameter name	Type	Pop-Up
Bubble string name	**bubble_name**	**input**	

string name of the bubbles.

Continue to the next section 26.8.7 Flow Arrows or return to 26.8 Drainage Plan Plot PPF Editor

26.8.7 Flow Arrows

Panel field	Parameter name	Type	Pop-Up

Section: Flow Arrows

Draw flow arrows **flow_arrow_mode** **tick box**

draw flow arrows near the pipe diameter labels, showing the direction of flow in the pipes.

Arrow colour **flow_arrow_colour** **colour box**

colour of the flow arrows.

Solid fill arrows **flow_arrow_fill** **tick box**

Arrow length **flow_arrow_length** **input**

length of the flow arrows. Setting the length to a negative number will place the arrow on the other side of the pipe (whilst still maintaining the correct arrow direction).

Arrow rise **flow_arrow_rise** **input**

Arrow string name **flow_arrow_name** **input**

string name of the flow arrows.

Continue to the next section 26.8.8 Buttons at Bottom of Panel or return to 26.8 Drainage Plan Plot PPF Editor

26.8.8 Buttons at Bottom of Panel

This is documented for all the PPF Editors in 26.2.11 Buttons at the Bottom of the PPF Editors.

Return to 26.8 Drainage Plan Plot PPF Editor

26.9 Plot Frame and PPF Editor

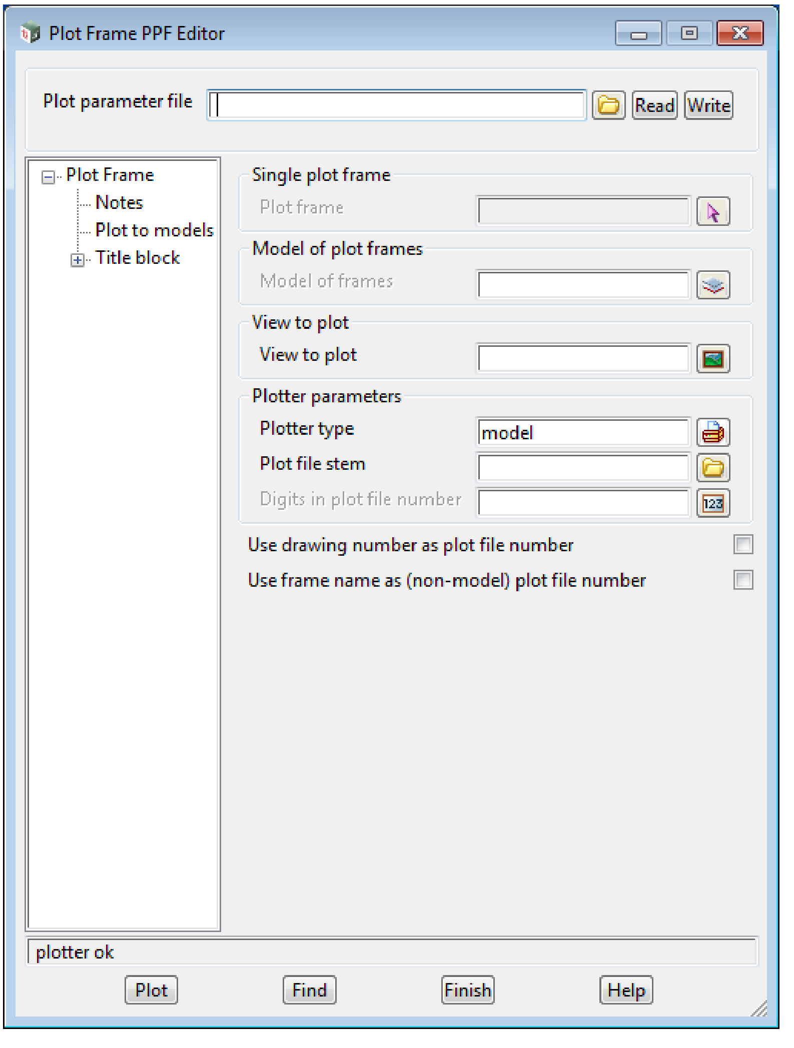
Position of option on menu: **Plot =>Plot and PPF Editors =>Plot frames**

The **Plot Frame PPF Editor** is for creating and/or editing a (binary) plot frame PPF file and for creating a cross section plot.

Note: Binary PPFs are stored *within* the project (not in the folder containing the project as the text ppf's were).

On selecting the **Plot frames** option, the **Plot Frame PPF Editor** panel is displayed.

The plot parameters for controlling the cross section plots are accessed by expanding to the appropriate node in the ***Plot Frame*** tree (click on the **+** to expand to node or **-** to collapse the node) and then clicking on the required node, and the required information to fill in is displayed on the right hand side of the panel.



See

26.9.1 Plot Frame - Front Page
26.9.2 Notes - Plot Frame
26.9.4 Plot Frame Title Block
26.9.5 Buttons at Bottom of Plot Frame PPF Editor

26.9.1 Plot Frame - Front Page

Section: Single plot frame

The fields and buttons used in this section have the following functions.

Field Description	Parameter name	Type	Pop-Up
Plot frame	**single_frame**	**string select box**	

select the plot frame to be plotted.

Section: Model of plot frames

The fields and buttons used in this section have the following functions.

Field Description	Parameter name	Type	Pop-Up
Model of frames	**model_of_frames**	**model box**	

model of plot frames to plot.

Section: View to plot

The fields and buttons used in this section have the following functions.

Field Description	Parameter name	Type	Pop-Up
View to plot	**view_name**	**model box**	

view in which data to plot resides.

Section: Plotter parameters

The fields and buttons used in this section have the following functions.

Field Description	Parameter name	Type	Pop-Up
Plotter type	**plotter_type**	**plotter box**	

a valid plotting option is selected.

Field Description	Parameter name	Type	Pop-Up
Plot file stem	**plot_stem**	**plotter box**	

plot file name. The appropriate extension is added dependant on the plotter type selected.

Clean plot models beforehand

Field Description	Parameter name	Type	Pop-Up
	plot_model_clean	**choice box**	**do not clean** **prompt for clean** **always clean**

whether to clean (delete the elements in) any resultant plot models that may already exist, before generating the plot(s). This parameter is only applicable if plotting to a model or models. Note that if the models are cleaned using this parameter, any non-plot or locked elements found in the models will not be cleaned from the models, and the plot job will be cancelled.

Use drawing numbers in plot file names use_drawing_numbers_in_filenames tick box

Please continue to the next section 26.9.2 Notes - Plot Frame.

26.9.2 Notes - Plot Frame

This is documented for all the PPF Editors in 26.2.4 Notes.

Please continue to the next section 26.9.3 Plot to Models - Plot Frame.

26.9.3 Plot to Models - Plot Frame

This is documented for all the PPF Editors in 26.2.5 Plot to models.

Please continue to the next section 26.9.4 Plot Frame Title Block.

26.9.4 Plot Frame Title Block

This is documented for all the PPF Editors in 26.2.6.1 Title Block Section in PPF Editors.
For more general information about a title block, see 26.2.6 Title Block.

Return to 26.9 Plot Frame and PPF Editor.

26.9.5 Buttons at Bottom of Plot Frame PPF Editor

This is documented for all the PPF Editors in 26.2.2 View to Load and Global Variables.

Return to 26.9 Plot Frame and PPF Editor.

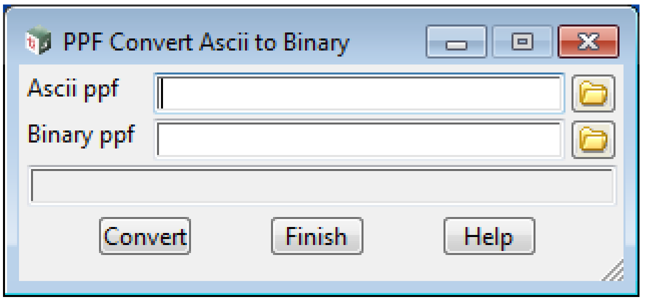
26.10 Convert Ascii PPF to Binary

Position of option on menu: **Plot =>Plot and PPF Editors =>Convert Ascii to Binary**

The **Convert ascii to binary** option is for converting the old text plot parameter files to the new binary plot parameter file format.

Note: Binary PPFs are stored *within* the project (not in the folder containing the project as the Text PPFs were).

Selecting the **Convert ascii to binary** displays the **PPF Convert Ascii to Binary** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Ascii ppf	file box		

name of the Text plot parameter file to convert.

Binary ppf	button		

convert the Text PPF to the equivalent binary PPF. The binary PPF file will be stored in the current project.

Convert	button		

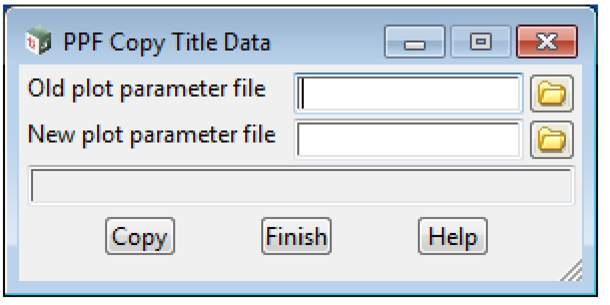
convert the Text PPF to a binary PPF.

26.11 Copy Title Data

Position of option on menu: **Plot =>Plot and PPF Editors =>Copy title data**

The **Copy title data** option is for copying the title data in one binary plot parameter to another binary plot parameter file format.

On selecting the **Copy title data** option, the **PPF Copy Title Data** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description		Type		Defaults		Pop-Up

Old plot parameter file	file box

name of the plot parameter file to copy the title data from.

New plot parameter file	button

name of the plot parameter file to copy the title data to.

Copy	button

copy the title data from one PPF file to another.

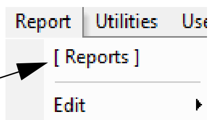
27 Reports

Position of menu: **It is on the main menu as Report**

The **Reports** menu contains options to report on the strings in a model, polygon details (area, centroid), string set outs, string names, crest and sag points and x-falls.

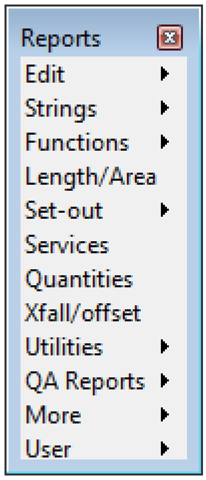
The **Reports** menu is

on Main menu



create floating **Reports** menu

on 12d Model menu and floating Reports menu



edit a .rpt file
report on strings, crest/sag points, names
report on functions
length, area, centroid of strings
point and string set-out information

calculations of quantities
report x-fall between two strings
XML report utilities
QA reports
further reports
Reports User menu

For the option/menu *Edit*, go to 27.1 Edit
Strings 27.2 Strings
Functions 27.3 Functions
Length/area 27.4 Length and Area
Set-out 27.5 Set-Out Reports
Services 27.6 Services
Quantities 27.7 Quantities
X-fall 27.8 X-Fall and Offset Report
Utilities 27.9 Report Utilities
QA reports 27.10 QA Reports
More 27.11 More Reports

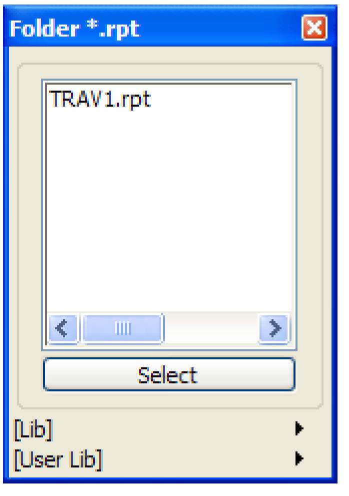
Each menu option will now be discussed.

27.1 Edit

Position on menu: **Report =>Edit**

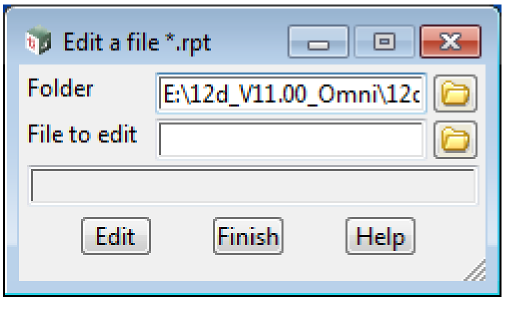
The **Edit** option is used to edit any (printer) reports ending in *.rpt.

On walking right on *Edit*, the list of files ending in *.rpt are displayed.



Double clicking on the report name will bring up the report in the system editor.

Clicking on the *Edit* at the top of the menu brings up the **Edit a file** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Folder	folder box	current folder	folder browser

the folder to look for the report file.

Field Description	Type	Defaults	Pop-Up
File to edit	file box		*.rpt

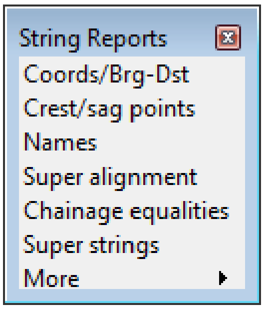
name of the file to bring up in the system editor.

27.2 Strings

Position of menu: **Report =>Strings**

The **Strings** menu contains options to report on selected strings, report the crest/sag points for a string and create a list of all the unique string names in a model or view.

The **Strings** walk-right menu is



For the option *Coords/Brg-Dst*, go to 27.2.1 Coordinates or Bearing-Distance Report

Crest/sag points 27.2.2 Crest/Sag Points Report

Names 27.2.3 Names Report

Super alignment 27.2.4 Super Alignment Report

Chainage Equalities 27.2.5 Chainage Equalities Report

Super strings 27.2.6 Report for Super Strings

More 27.2.7 More Strings Reports.

27.2.1 Coordinates or Bearing-Distance Report

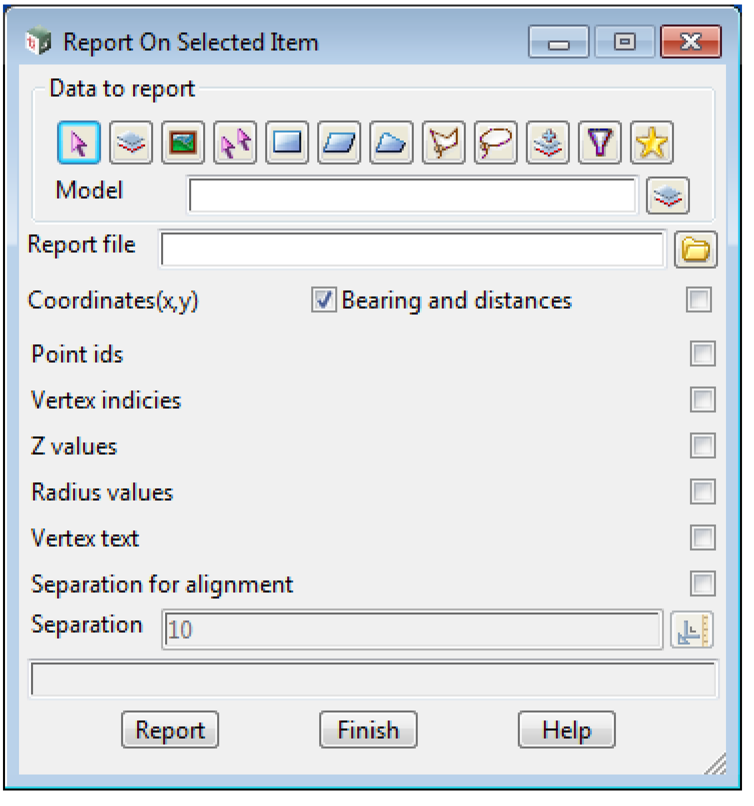
Position of option on menu: **Report =>Strings =>Coords/Brg-Dst**

The **Coords/Brg-Dst** option is used to generate a report on selected items defined by a source box. The report can be of the form of coordinates of each point of a string, or the co-ordinates of the first point of a string and the bearings and distances from each other point of the string to the first point of the string.

For a a Super Alignment string it will report the solved horizontal and vertical geometry (the horizontal and vertical Segments that make up the Super Alignment). If there is a non-zero separation value, then at each chainage that is a multiple of the separation value, the report also includes the chainage, x, y, z, and in and out bearings at he chainage.

For an Alignment string it will report on the HIP's and VIPS. If there is a non-zero separation value, then at each chainages that are multiples of the separation value, the report also includes the chainage, x, y, z, bearing for horizontal and chainage, height (z), grade for vertical.

On selecting the **Coords/Brg-Dst** option, the **Report on Selected Item** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Report file	input		*.rpt

name of the file to report to.

Coordinates(x,y)	tick box	tick	

*if **tick**, the coordinate report parameters will be displayed which will determine what other information as well as the coordinate values will be placed in the report. **NOTE:** If an **alignment string** is part of the selection, the report will include **all** relevant details about the alignment. In this case the report will not use the coordinate report parameters as specified, for the alignment string.*

Bearing and Distances tick box

*if **tick**, the bearing and distance report parameters will be displayed which will determine what other information as well as the bearing and distance values will be placed in the report.*

Coordinate report parameters

if the **Coordinates(x,y)** option is selected, the following parameters will be displayed

Point ids tick box

if selected, any valid point ids will be shown in the report.

Vertex indices tick box

if selected, vertex indices will be shown in the report.

Z Values tick box

if selected, any valid z values will be shown in the report.

Radius values tick box

if selected, any valid radius values will be shown in the report.

Vertex Text tick box

if selected, any valid vertex text values will be shown in the report.

Separation for Alignment tick box

if selected, the separation for alignment box will be enabled.

Separation input box 10

if enabled by ticking the previous tick box, the alignment string information will be reported at a chainage interval given by the separation value.

Report button

run the option and create the report.

For a Super Alignment, the report is the same as **Reports =>Strings => Super alignment** (see 27.2.4 Super Alignment Report).

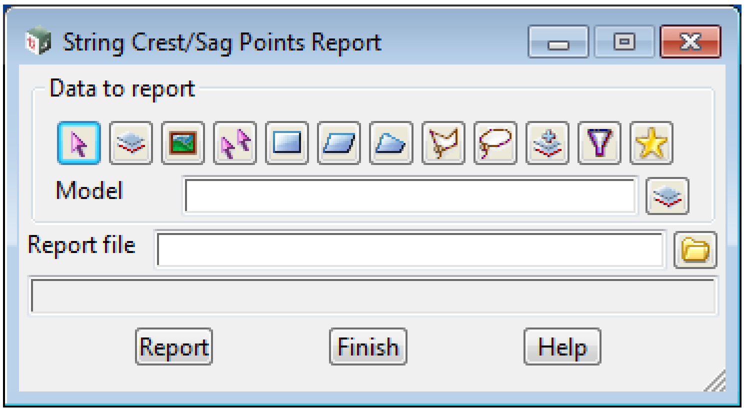
27.2.2 Crest/Sag Points Report

Position of option on menu: **Report =>Strings =>Crest sag points**

The **crest/sag points** option is used to generate a (printer) report on the crests and sags points of a string.

A number of strings can be reported on by selecting them in turn

On selecting the **crest/sag points** option, the **string crest/sag points report** panel is displayed.



The fields and buttons in this panel are used as follows

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source

type of data source.

Report file	input		*.rpt

name of the file to contain the reports on the crest and sag points of strings.

Pick & Report	button		

as strings are selected, the crest and sag points report is generated and appended to the report file.
The cycle is terminated by clicking RB to raise the **pick ops** *menu and selecting* **cancel** *from it.*

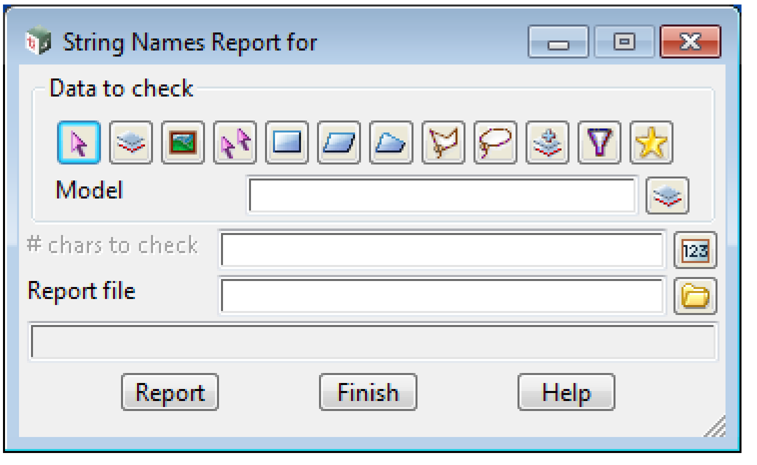
27.2.3 Names Report

Position of option on menu: **Report =>Strings =>Names**

The **names** option is used to report on all the unique strings in a model or on a view.

If the **report** button is selected, then all the strings in the model/view given by the model/view to report field will be checked and all unique names, plus a frequency count, will be written to the report file.

Selecting **Names** displays the **String Names Report for** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source

type of data source.

# chars to check	input		

*if **non blank**, the number of characters to be used in the string name when checking for uniqueness.*
*If **blank**, then the entire string name is used.*

Report file	file box		*.rpt files

name of the file for the report.

Report	button		

report on all the unique string names in model/view.

27.2.4 Super Alignment Report

Position of option on menu: **Report =>Strings =>Super alignment**

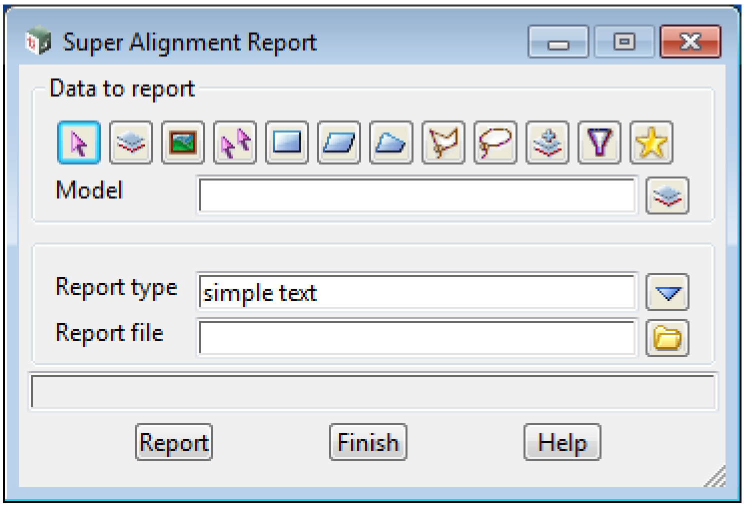
Position of option on menu: **Strings =>SA tools =>Report**

Report on the Super Alignment that gives information on the horizontal and vertical components.

For a Super Alignment string it will report the solved horizontal and vertical geometry (the horizontal and vertical Segments that make up the Super Alignment).

If there is a non-zero separation value, then at each chainages that are multiples of the separation value, the report also includes the chainage, x, y, z, in and out bearing for horizontal vertices, and chainage, height (z) and in and out grade for vertical vertices.

On selecting the **Super alignment** option, the **Super Alignment Report** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	
data selection type - for a full description go to 4.19.3 Data Source.			
Data source			
type of data source.			
Report type	choice box	simple text	html tables, pdf tables, original xml, csv ip tables, simple text, crests and sags
type of report			
Report file	input		*.rpt
name of the file to report to. See below for definition of the report and an example.			
Report	button		
run the option and create the report.			

The Super Alignment report is made up of 5 parts:

(a) General Information

The type of transition curve.

Whether the super alignment is closed or not.

The start chainage for the string and the chainage length (2d length) of the string.

(b) Horizontal Vertex Data

The **Horizontal Vertex Data** section of the report gives information about each of the *horizontal vertices* of the underlying ***horizontal segment geometry***.

Note: "Solving" a super alignment takes all the constraints and produces a string of vertices with segments of type Line, Arc or Transition between each pair of adjacent vertices.

So the **Horizontal Vertex Data** report gives the (x,y,z) coordinates of each of the vertices of the solved horizontal geometry with the type of segment going into the vertex (**In Seg**), and the type of segment going out of the vertex (**Out Seg)**.

The **In Direction** is the angle at the vertex of the segment going into the vertex, and the **Out Direction** is the angle at the vertex of the segment going out of the vertex.

If the *In* and *Out Direction* are the same, then the two segments are *tangential* at the vertex and there will be a *Yes* in the **Tan** column, otherwise there will be a *No* in the **Tan** column.

In Seg

The type of the segment going into the critical point (In Segment) at the chainage - Line, Arc, Leading Spiral (LSpiral), Trailing Spiral (TSpiral)

Out Seg

The type of the segment going out of the critical point (Out Segment) at the chainage - Line, Arc, Leading Spiral (LSpiral), Trailing Spiral (TSpiral)

Tan - yes or no

yes if the In Segment and Out Segment are tangential. That is, the In Direction and Out Direction are the same.

no if the In Segment and Out Segment are not tangential. That is, the In Direction and Out Direction are different.

Chainage

chainage of the critical point

X, **Y** and **Z Coordinates** at that chainage

In Direction

the angle of the In Segment at the critical point. Measured in dms, counterclockwise from the positive x-axis.

Out Direction

the angle of the Out Segment at the critical point. Measured in dms, counterclockwise from the positive x-axis.

In Radius

the radius of the In Segment at the critical point (or blank if on a Line).

Out Radius

the radius of the Out Segment at the critical point (or blank if on a Line).

(c) Horizontal Geometry Data

How each of the horizontal parts is defined.

(d) Vertical Vertex Data

The **Vertical Vertex Data** section of the report gives information about each of the *vertical vertices* of the underlying ***vertical segment geometry***.

Note: "Solving" the vertical of a super alignment takes all the constraints and produces a 2d string in the (Chainage, height) plane, of vertices with segments of type Line, Arc or Parabola between each pair of adjacent vertices.

So the **Vertical Vertex Dat**a report gives the (ch,z) coordinates of each of the vertices of the solved vertical geometry with the type of segment going into the vertex (**In Seg**), and the type of segment going out of the vertex (**Out Seg)**.

The **In Grade** is the grade at the vertex of the segment going into the vertex, and the **Out Grade** is the grade at the vertex of the segment going out of the vertex.

If the *In* and *Out Grade* are the same, then the two segments are *tangential* at the vertex and there will be a *Yes* in the **Tan** column, otherwise there will be a *No* in the **Tan** column.

In Seg

The type of the segment going into the critical point (In Segment) at the chainage - Line, Arc, Parabola

Out Seg

The type of the segment going out of the critical point (Out Segment) at the chainage - Line, Arc, Parabola

Tan - yes or no

yes if the In Segment and Out Segment are tangential. That is, the In Grade and Out Grade are the same.

no if the In Segment and Out Segment are not tangential. That is, the In Grade and Out Grade are different.

Chainage

chainage of the critical point

Level

height (z value) at that chainage

In Grade

the grade of the In Segment at the critical point.

Out Grade

the grade of the Out Segment at the critical point.

In VCL Radius

the radius of the parabola or arc of the In Segment at the critical point (or blank if on a Line).

Out VCL Radius

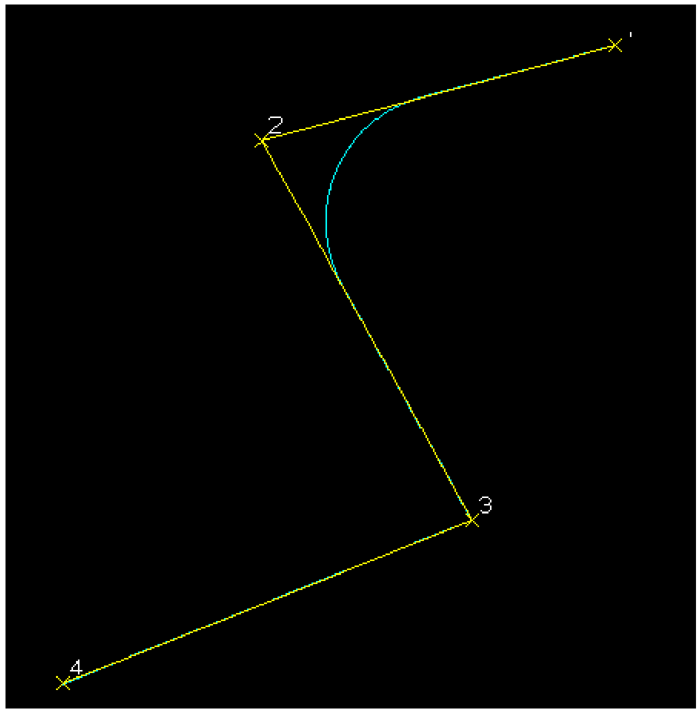
the radius of the parabola or arc of the Out Segment at the critical point (or blank if on a Line).

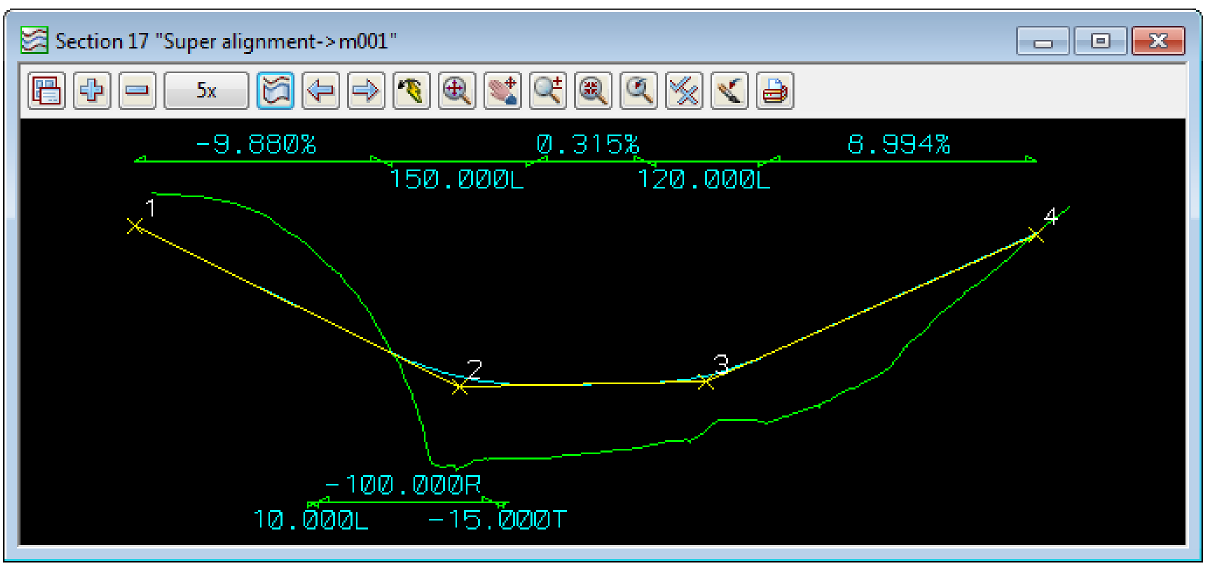
(e) Vertical Geometry Data

How each of the vertical parts is defined.

Super Alignment to Report

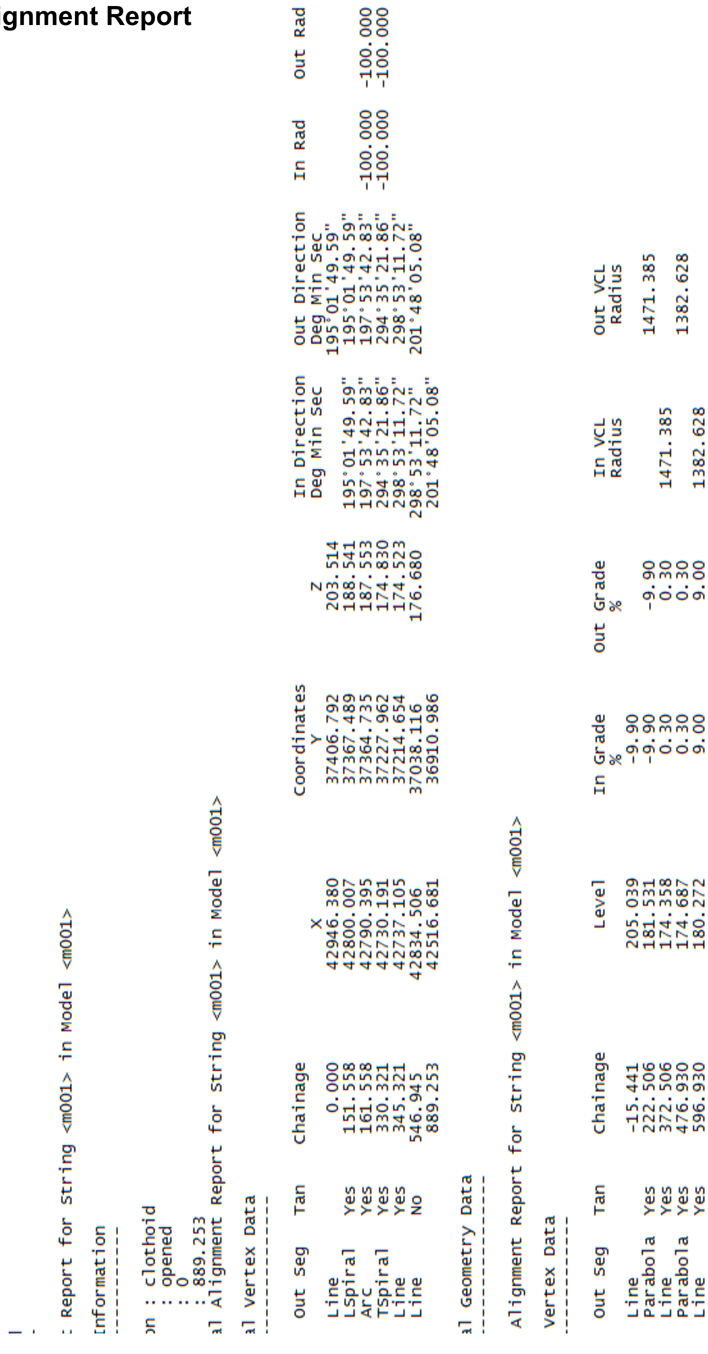
Horizontal Geometry





Vertical Geometry

Super Alignment Report



Report for String <m001> in Model <m001>

Information
----

on : clothoid
: opened
: 0
: 889.253

al Alignment Report for String <m001> in Model <m001>

al Vertex Data

Out Seg	Tan	Chainage	Coordinates X	Coordinates Y	Z	In Direction Deg Min Sec	Out Direction Deg Min Sec	In Rad	Out Rad
Line		0.000	42946.380	37406.792	203.514		195°01'49.59"		
LSpiral	Yes	151.558	42800.007	37367.489	188.541	195°01'49.59"	195°01'49.59"		
Arc	Yes	161.558	42790.395	37364.735	187.553	197°53'42.83"	197°53'42.83"	-100.000	-100.000
TSpiral	Yes	330.321	42730.191	37227.962	174.830	294°35'21.86"	294°35'21.86"	-100.000	-100.000
Line	Yes	345.321	42737.105	37214.654	174.523	298°53'11.72"	298°53'11.72"		
Line	No	546.945	42834.506	37038.116	176.680	298°53'11.72"	201°48'05.08"		
		889.253	42516.681	36910.986		201°48'05.08"			

al Geometry Data

Alignment Report for String <m001> in Model <m001>

Vertex Data

Out Seg	Tan	Chainage	Level	In Grade %	Out Grade %	In VCL Radius	Out VCL Radius
Line		-15.441	205.039	-9.90			
Parabola	Yes	222.506	181.531	-9.90	-9.90		1471.385
Line	Yes	372.506	174.358	0.30	0.30	1471.385	
Parabola	Yes	476.930	174.687	0.30	0.30		1382.628
Line	Yes	596.930	180.272	9.00	9.00	1382.628	
		855.532	203.531		9.00		

Old Report - V9

The Super Alignment report is made up of 5 parts:

(a) General Information

The date the report was produced and the string and model name for the string.

The type of transition curve.

Whether the super alignment is closed or not.

The start chainage for the string and the chainage length (2d length) of the string.

(b) A report for the critical points, and the horizontal chainage increment of the string giving the

Point Type at the chainage (eg Straight, Tangent-Curve, Curve, Spiral-Curve etc)

Chainage at that point.

X, **Y** and **Z Coordinates** at that chainage

Instantaneous **Bearing** at that chainage

Instantaneous **Radius** at that chainage (or blank if on a straight)

(c) Horizontal Vertex Data

The **Horizontal Vertex Data** section of the report gives information about each of the *horizontal vertices* of the underlying ***horizontal segment geometry***.

Note: "Solving" a super alignment takes all the constraints and produces a string of vertices with segments of type Line, Arc or Transition between each pair of adjacent vertices.

So the **Horizontal Vertex Dat**a report gives the (x,y,z) coordinates of each of the vertices of the solved horizontal geometry with the type of segment going into the vertex (**In Seg**), and the type of segment going out of the vertex (**Out Seg)**.

The **In Bearing** is the bearing at the vertex of the segment going into the vertex, and the **Out Bearing** is the bearing at the vertex of the segment going out of the vertex.

If the *In* and *Out Bearings* are the same, then the two segments are *tangential* at the vertex and there will be a *Yes* in the **Tan** column, otherwise there will be a *No* in the **Tan** column.

(d) Horizontal Geometry Details

How each of the horizontal parts is defined.

(e) Vertical Vertex Data

The **Vertical Vertex Data** section of the report gives information about each of the *vertical vertices* of the underlying ***vertical segment geometry***.

Note: "Solving" the vertical of a super alignment takes all the constraints and produces a 2d string in the (Chainage, height) plane, of vertices with segments of type Line, Arc or Parabola between each pair of adjacent vertices.

So the **Vertical Vertex Dat**a report gives the (ch,z) coordinates of each of the vertices of the solved vertical geometry with the type of segment going into the vertex (**In Seg**), and the type of segment going out of the vertex (**Out Seg)**.

The **In Grade** is the grade at the vertex of the segment going into the vertex, and the **Out Grade** is the grade at the vertex of the segment going out of the vertex.

If the *In* and *Out Grade* are the same, then the two segments are *tangential* at the vertex and there will be a *Yes* in the **Tan** column, otherwise there will be a *No* in the **Tan** column.

(f) Horizontal Geometry Details

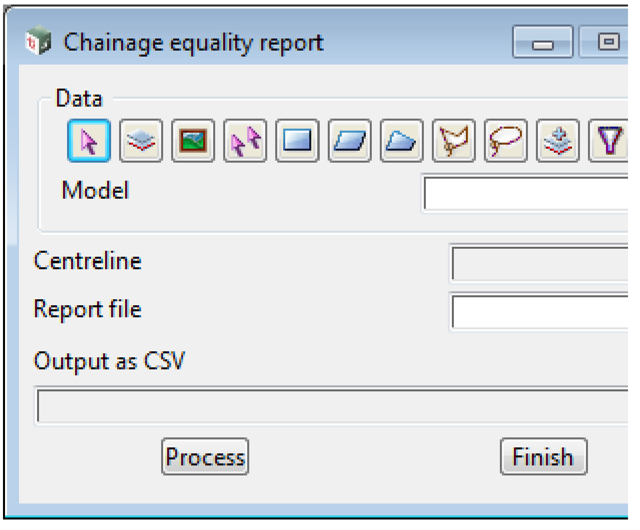
How each of the vertical parts is defined.

27.2.5 Chainage Equalities Report

Position of option on menu: **Report =>Strings =>Chainage equalities**

This option is under development.

On selecting the **Chainage equalities** option, the **Chainage Equality Report** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source

type of data source.

Centreline string select

name of the super alignment to report on.

Report File file select

name of the file to report to.

Output as CSV tick box

if tick, write the report out as a CSV file

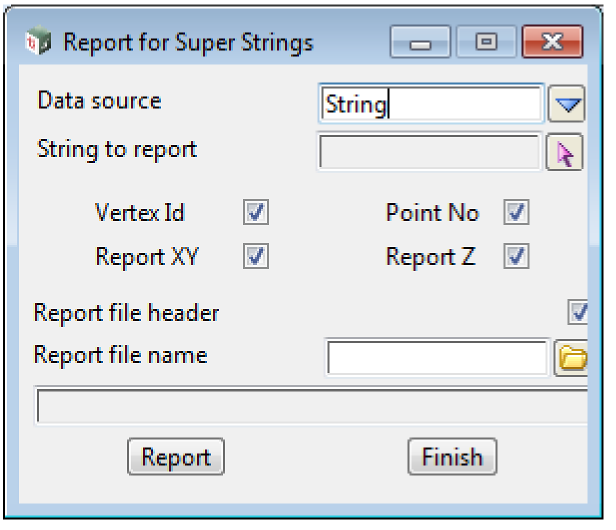
Report button

run the option and create the report.

27.2.6 Report for Super Strings

Position of option on menu: **Report =>Strings =>Super strings**

This option reports on information in the super string.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source

type of data source.

Vertex index	tick box	tick	

if tick, vertex indices are reported.

Point id	tick box	tick	

if tick, point ids are reported.

Report XY	tick box	tick	

if tick, x and y values are reported.

Report Z	tick box	tick	

if tick, z values are reported.

Report file header	input box	tick	

if tick, a header including the date and model name and string names is included.

Report file name	input box		

name of the report.

Report	button		

Run the option.

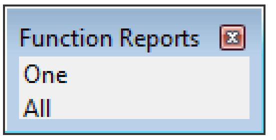
27.2.7 More Strings Reports

There are currently no options on this menu.

27.3 Functions

Position of menu: **Report =>Functions**

A report can be made on a function. The **function** walk-right menu is

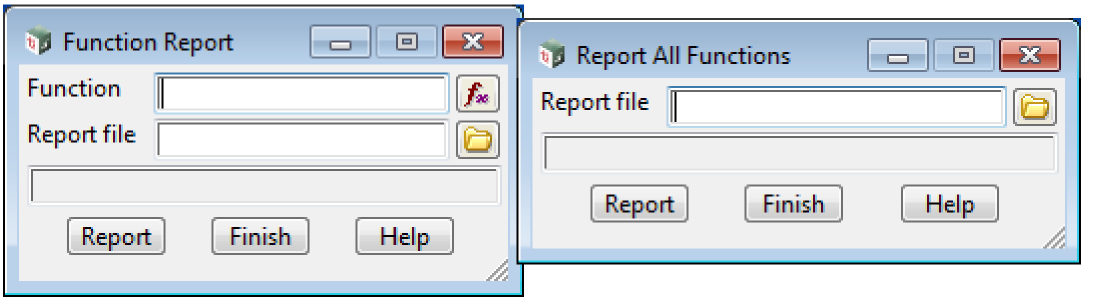


27.3.1 Report a Function, Report all Functions

Position of option on menu: **Report =>Functions =>One**

Position of option on menu: **Report =>Functions =>All**

On selecting the **report** or **report all** option, the **function report/ report all functions** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function	function box		available functions

name of the function to be reported.

Report file	file box		*.rpt files

name of the file to write the report to.

Report	button		

after selecting this button, a report on the function/ all functions is produced.

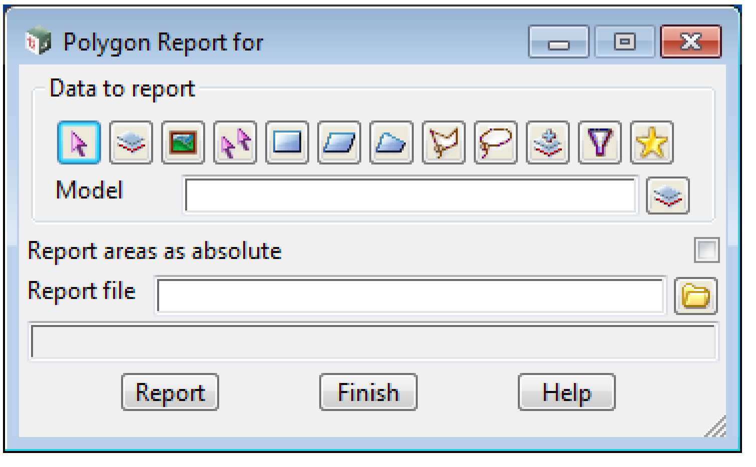
27.4 Length and Area

Position of option on menu: **Report =>Length/Area**

The **Length/area (Polygons)** option creates a report containing the following information for each selected string:

- string name
- if the string is closed or not
- length of the string
- area of the string (if not closed, join first and last points). A polygon area is positive if the vertices of the polygon go in a clockwise direction, or negative if the vertices go in an anti-clockwise direction.
- centroid co-ordinates

On selecting the Length/Area option, the Polygon Report for panel is displayed.



The fields and buttons in this panel are used as follows

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source

type of data source.

Report areas as absolute tick box

A polygon area is positive if the vertices of the polygon go in a clockwise direction, or negative if the vertices go in an anti-clockwise direction.

if ticked, report all areas as positive.

Report file	input		*.rpt

file for the polygon reports.

Report	button		

produce a polygon report for all strings selected in the Data source.

27.5 Set-Out Reports

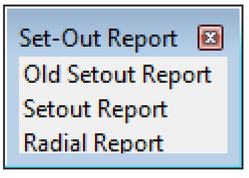
Position of menu: **Report =>Set-out**

The **Set out** report options are for reporting the bearing and distance from a selected instrument station to individual points, strings, models of strings or views of strings.

In the *Setout report*, a backsight stations can be selected and the bearing of the **line** from the instrument station to the backsight station reported. Individual backsight points or strings of backsight points can be selected for reporting.

In the *Radial report*, the report can be sorted by point number, bearing or distance.

The **set outs** walk-right menu is



old set out report from V5.0
set out report for strings
set out report sorted by name, distance or bearing

For the option *Old setout report*, go to 27.5.1 Old Setout Report
Setout report 27.5.2 Setout Report
Radial report 27.5.3 Radial Report

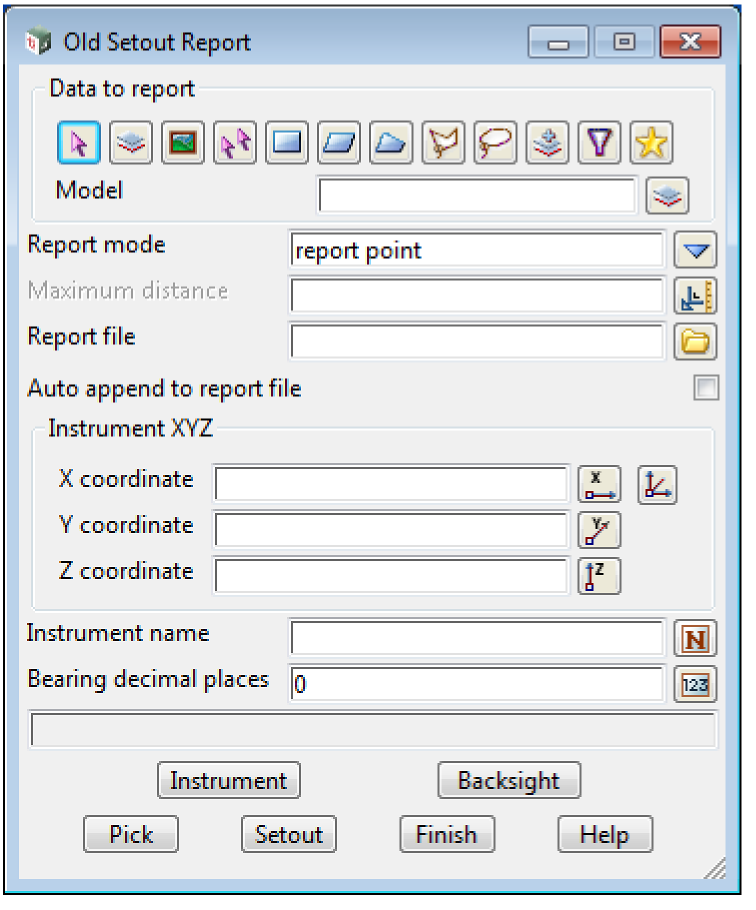
The menu options will now be discussed.

27.5.1 Old Setout Report

Position of option on menu: **Report =>Set out =>Old setout report**

The setout report from ***12d Model*** V5. This option has been rewritten for ***12d Model*** V6 and is called **Setout Report**.

On selecting the **Old setout report** option, the **Old Setout Report** panel is displayed.



The fields and buttons in this panel are used as follows

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source

type of data source.

Report mode	input		report points, report string

*if **report points**, report selected points. Otherwise report on all points in the selected string.*

Maximum distance	input		

*if **non-blank**, any points further that the maximum distance from the instrument station will not be reported on. If **blank**, all selected points will be reported on*

Report file	input		

file for the backsight and setout reports

Instrument XYZ/name output

co-ordinates/name of the current instrument station

Bearing decimal places input

number of decimal places to report the bearings to

Instrument button

the required instrument station is selected and its name and position are reported in the instrument XYZ *and* instrument name *panel fields.*

Backsight button

a backsight is selected and written to the report file.

Pick button

*After pick is chosen, a set out report is written to the report file for any selected strings. If the report mode is **report point**, only the individual point is reported. If the report mode is **report string**, all the points in the string are reported.*
*The cycle is terminated by clicking RB to raise the **pick ops** menu and selecting **cancel** from it.*

Setouts button

produce a set out report for all the strings in the model/view given in the model/view *field.*

How to Use the Panel and Panel Messages

(a) Enter the report file name and maximum distance.

(b) The instrument station is chosen by activating the **station** button and selecting the required station point. It will then be reported in the file

(c) Any backsight points to be reported can then be chosen by picking the **backsight** button and selecting the backsight point.

(d) To report on individual strings or points, set the report mode to either point or string. The **pick** button is then activated and the individual points or strings selected. The cycle is terminated by clicking RB to raise the **pick ops** menu and selecting **cancel** from it.

To report on all the strings in a model/view, give the model/view in the model/view field, and select the **setout** button.

Note

The instrument station can be changed at any time by selecting the **station** button and picking the new station.

All backsight and set outs will then be taken from the new instrument station.

27.5.2 Setout Report

Position of option on menu: **Report =>Set out =>Setout report**

In the *Setout report*, a backsight stations can be selected and the bearing of the **line** from the instrument station to the backsight station reported. Individual backsight points or strings of backsight points can be selected for reporting.

The instrument station can be changed at any time and new backsight and set outs reported.

The **set out** option prints the following information:

instrument station

string name, point number, x, y, z

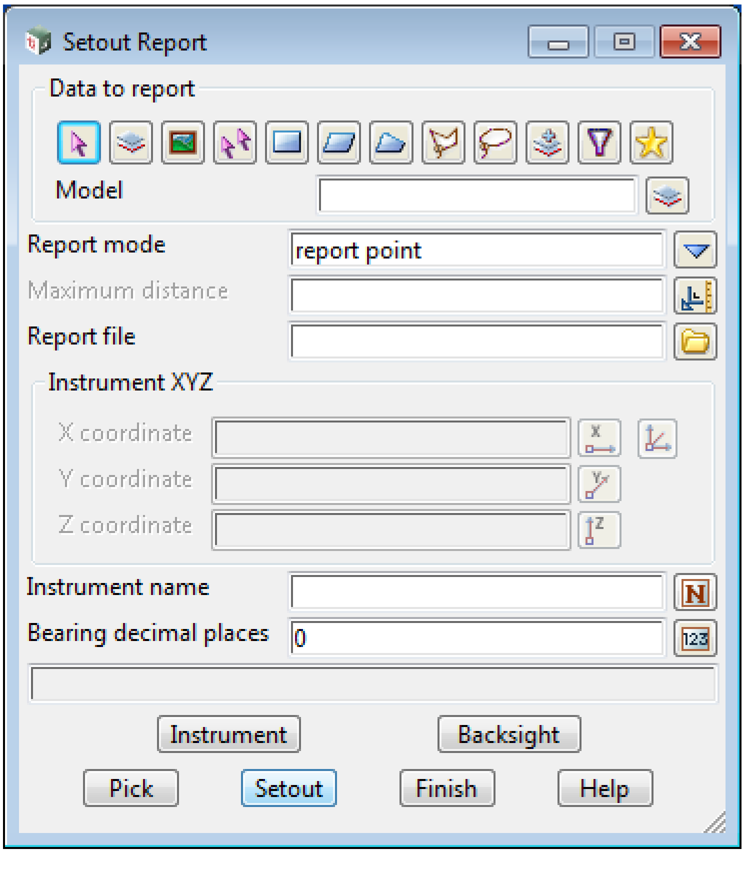
backsight point

string name, point number, x, y, z, bearing to the line joining the instrument and backsight, horizontal distance, height difference between instrument station and the backsight

point set-out point

string name, point number, x, y, z, bearing, horizontal distance, height difference between instrument station and the point

On selecting the **Setout report** option, the **Setout Report** panel is displayed.



The fields and buttons in this panel are used as follows

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source

type of data source.

Report mode choice box report points, report string

*if **report points**, report selected points. Otherwise report on all points in the selected string.*

Maximum distance input

*if **non-blank**, any points further that the maximum distance from the instrument station will not be reported on. If **blank**, all selected points will be reported on*

Report file file box

file for the backsight and setout reports

Instrument XYZ/name output

co-ordinates/name of the current instrument station

Bearing decimal places input

number of decimal places to report the bearings to

Instrument button

the required instrument station is selected and its name and position are reported in the instrument XYZ *and* instrument name *panel fields.*

Backsight button

a backsight is selected and written to the report file.

Pick button

*After pick is chosen, a set out report is written to the report file for any selected strings. If the report mode is **report point**, only the individual point is reported. If the report mode is **report string**, all the points in the string are reported.*
The cycle is terminated by clicking RB to raise the **Pick ops** *menu and selecting* ***Cancel*** *from it.*

Setouts button

produce a set out report for all the strings in the model/view given in the model/view *field.*

How to Use the Panel and Panel Messages

(a) Enter the report file name and maximum distance.

(b) The instrument station is chosen by activating the **station** button and selecting the required station point. It will then be reported in the file

(c) Any backsight points to be reported can then be chosen by picking the **backsight** button and selecting the backsight point.

(d) To report on individual strings or points, set the report mode to either point or string. The **pick** button is then activated and the individual points or strings selected. The cycle is terminated by clicking RB to raise the **pick ops** menu and selecting **cancel** from it.

To report on all the strings in a model/view, give the model/view in the model/view field, and select the **setout** button.

Note

The instrument station can be changed at any time by selecting the **station** button and picking the new station.

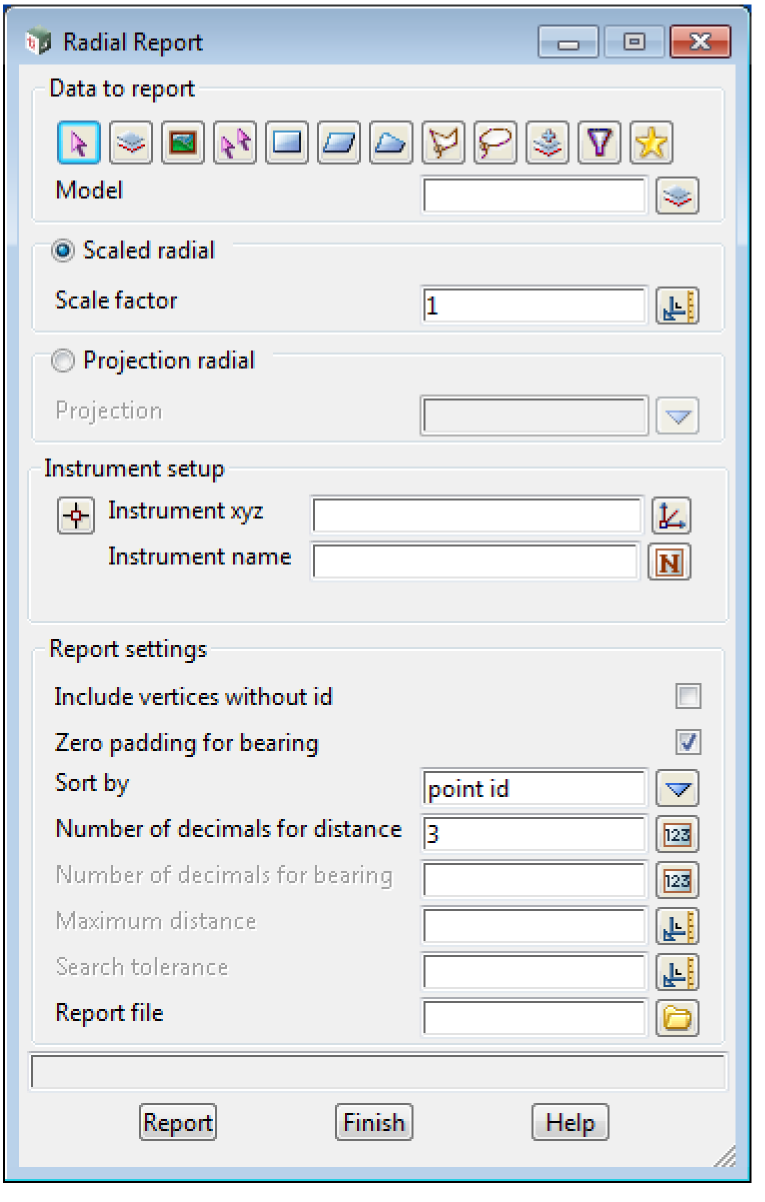
All backsight and set outs will then be taken from the new instrument station.

27.5.3 Radial Report

Position of option on menu: **Report =>Set out =>Radial report**

The radial report reports the bearing and distance from the Instrument to selected vertices. the report can be sorted by point id, bearing or distance.

Selecting **Radial report** brings up the **Radial Report** panel.



The fields and buttons in this panel are used as follows

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source

type of data source.

Scaled radial radio button

if selected, the report uses a fixed scale factor.

Scale factor input 1

*if **scaled radial is** selected, the fixed scale factor to use*

Projection radial radio button

if selected, the report uses a projection for calculating distances

Projection projection box available projections

*if **projection radial** is selected, the selected projection is used when calculating distances*

Instrument setup

*when the instrument point is selected, the **Instrument co-ordinates** and **name** is displayed n the* Instrument xyz *and* Instrument name *fields.*

Report settings

Include points without id tick box

*if **tick**, vertices with a non blank point id are included in the report.*
*if **not tick**, only vertices with non blank point id are included in the report.*

Zero padding for bearing tick box

*if **tick**, include extra zeros for minutes and seconds.*

Sort by choice box point id point id, bearing, distance

*if **point id**, the report is sorted by the point id's on the vertex.*
***bearing** , the report is sorted by the bearing to the point.*
***distance** , the report is sorted by the distance to the point.*

Number of decimals input 3

number of decimal places for reporting co-ordinates

Maximum distance input

*if **non-blank**, any vertices further that the maximum distance from the instrument station will not be reported on. If **blank**, all vertices points will be reported on*

Search tolerance input

when a vertex is found, any vertices with the same point id within the search distance of the vertex are not reported. Also no vertices closer to the instrument point that this distance will be reported.

Report file file box

file for the setout reports

Report button

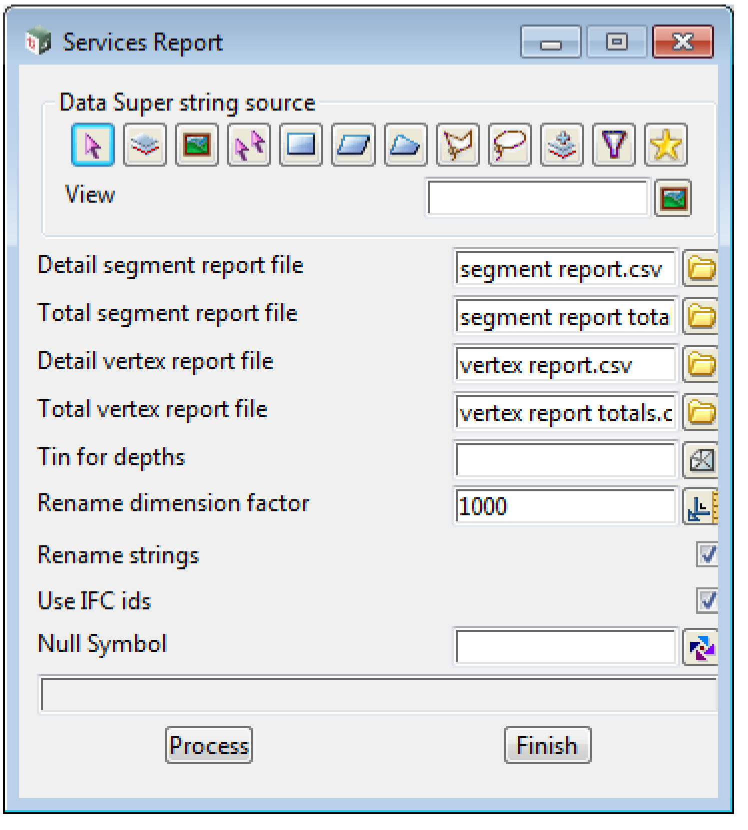
produce the report

27.6 Services

Position of option on menu: **Report =>Services**

Selecting **Services** brings up the **Services** panel.

This section of documentation is a work in progress and will be updated in subsequent releases.



The fields and buttons in this panel are used as follows

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source

type of data source.

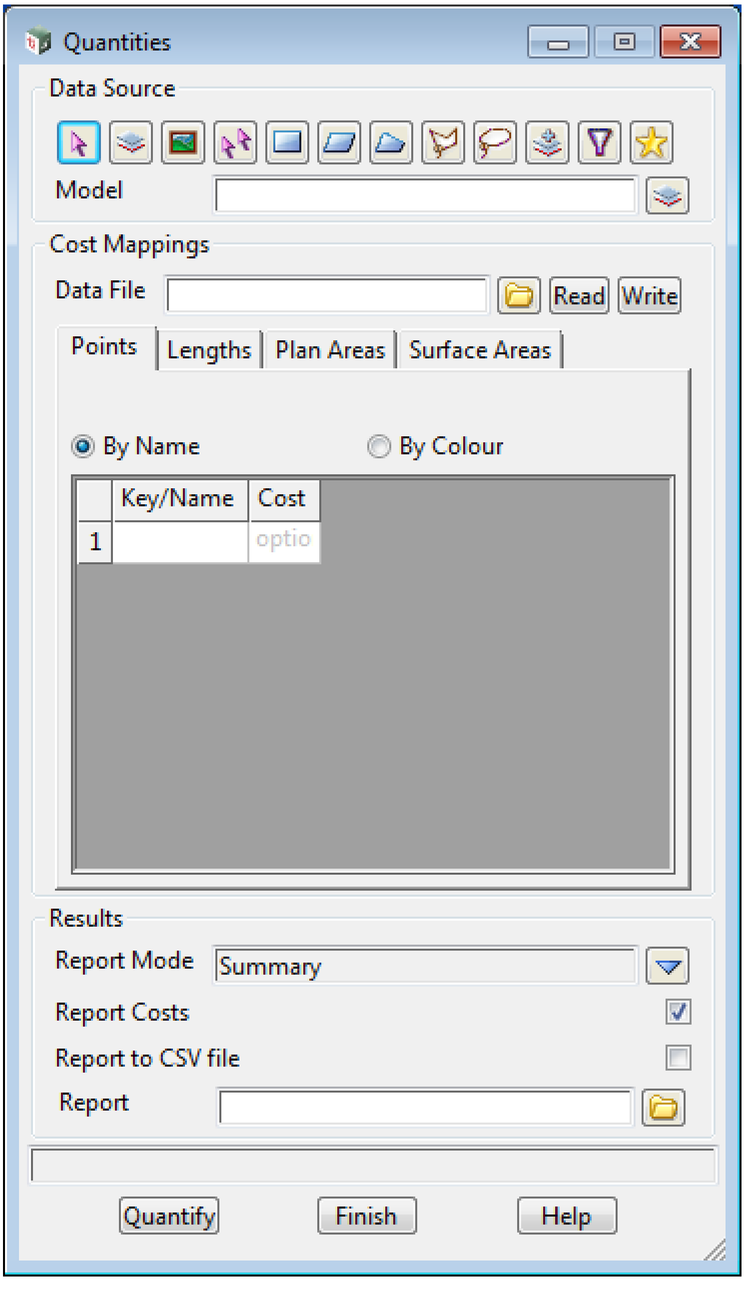
Field Description	Type	Defaults	Pop-Up
Data segment report file	file box	segment report.csv	available .csv files
Total segment report file	file box	segment report totals.csv	available .csv files
Detail vertex report file	file box	vertex report.csv	available .csv files
Total vertex report file	file box	vertex report totals.csv	available .csv files
Tin for depths	tin box		available tins
Rename dimension factor	measure box	1000	At Point, Point to Point, String from Point, String to Point

Rename strings	tick box	ticked	
Use IFS ids	tick box	ticked	
Null symbol	symbol box		available symbols

27.7 Quantities

Position of option on menu: **Report =>Quantities**

Selecting **Quantities** brings up the **Quantities** panel.



The fields and buttons in this panel are used as follows

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source

type of data source.

Data file file box

set up file for the costing data - file is read/written using the ***Read/Write*** *buttons.*

Costing tabs

Points tab and grid

By name **By colour**

for any string selected by the key/name or colour, the number of vertices in the string is multiplied by the cost and the total added to the quantities report.

Lengths tab and grid

By name **By colour**

for any string selected by the key/name or colour, the plan or 3d length of the string is multiplied by the cost and the total added to the quantities report.

Plan Areas tab and grid

By name **By colour**

for any string selected by the key/name or colour, the plan area of the string is multiplied by the cost and the total added to the quantities report.

Surface Areas tab and grid

By name **By colour**

for any string selected by the key/name or colour, the surface area of the string within the given tin is multiplied by the cost and the total added to the quantities report.

Report mode choice box Summary Summary, Full

produce the quantity report.

Report costs tick box

if tick, the cost used for each calculation is included in the report.

Report to CSV file tick box

if tick, the report is written as a CSV file

Report file box

name of the quantities report file

Quantify button

produce the quantity report.

27.8 X-Fall and Offset Report

Position of option on menu: **Report =>Xfall/offset**

The **x-fall** report option reports on the horizontal and vertical offsets and the cross-fall between two strings.

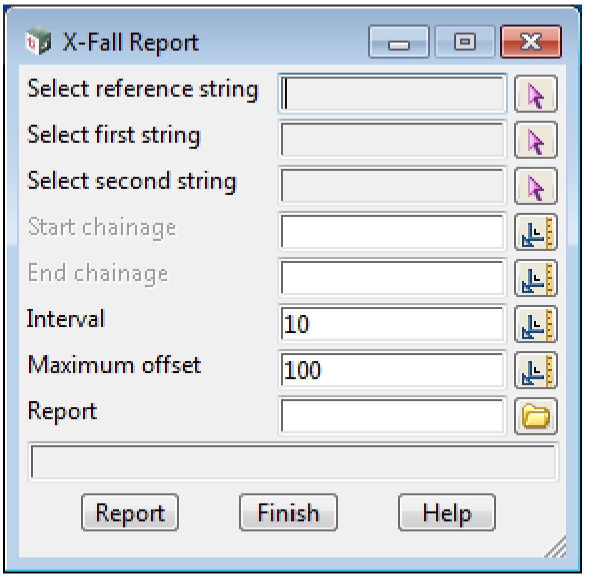
The offsets are calculated as follows:

(a) a reference string is selected which is used to define chainages.

(b) lines perpendicular to the reference string are taken at regular chainages and intersected (in plan) with the first and second strings.

(c) the horizontal and vertical offsets and the cross-fall between the two strings is calculated at the intersection points.

On selecting the **X-fall** option, the **X-Fall Report** panel is displayed.



The fields and buttons in this panel are used as follows

Field Description	Type	Defaults	Pop-Up
Start/End chainage	input		

if blank, the start/end chainage of the reference string is used.
if non-blank, the given chainage is used as the start/end chainage.

Interval	input		

chainage interval to calculate values at.

Maximum offset	input		

if non-blank, the maximum distance to search from the reference string to find the 1st and 2nd strings.

Report file	file box		*.rpt

file for the x-fall report

Ref/1st/2nd	button		

select the reference/first/second string.

Report	button		

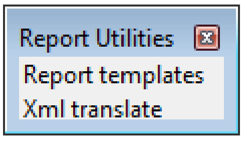
produce a offset and x-fall report between the first and second strings.

27.9 Report Utilities

Position of menu: **Report =>Utilities**

This section of documentation is a work in progress and will be updated in subsequent releases.

The **Utilities** walk-right menu is



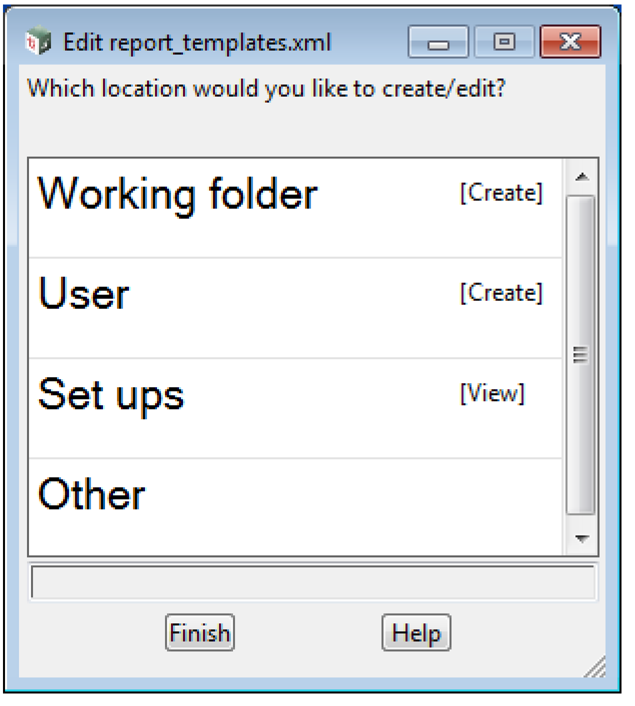
For the option *Report templates*, go to	27.9.1 Report Templates
XML translate	27.9.2 XML Translate

27.9.1 Report Templates

Position of option on menu: **Report =>Utilities =>Report templates**

This section of documentation is a work in progress and will be updated in subsequent releases.

On selecting **Report templates**, the **Edit** panel is displayed.

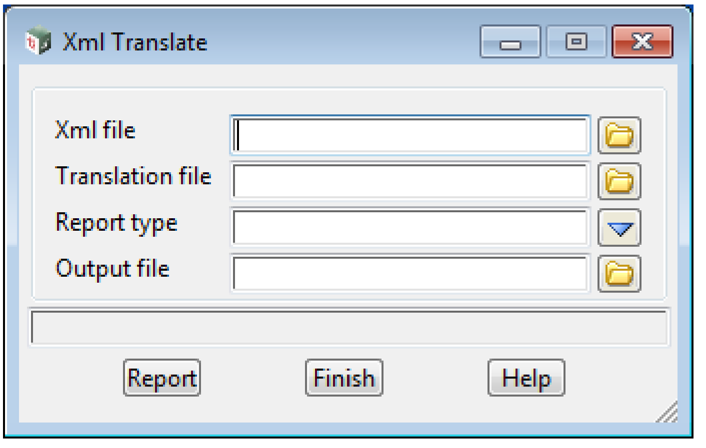


27.9.2 XML Translate

Position of option on menu: **Report =>Utilities =>XML translate**

This section of documentation is a work in progress and will be updated in subsequent releases.

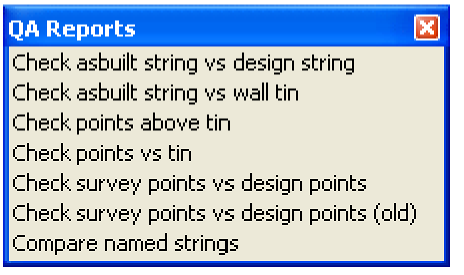
On selecting **XML translate**, the **XML Translate** panel is displayed.



27.10 QA Reports

Position of menu: **Report =>QA Reports**

The QA options are for checking surveyed strings against design strings or tins.



For *Check asbuilt string vs design string*,go to 27.10.1 Check As built String vs Design String

Check asbuilt string vs wall tin 27.10.2 Check Asbuilt String vs Wall Tin

Check points above tin 27.10.3 Check Points Above a Tin

Check points vs tin 27.10.4 Check Points vs Tin

Check survey points vs design points 27.10.6 Check Survey Points vs Design Points (3).

Check survey points vs design points 27.10.5 Check Survey Points vs Design Points (Old).

Compare name strings 27.10.7 Compare Named Strings

27.10.1 Check As built String vs Design String

Position of option on menu: **Report =>QA Reports =>Check asbuilt string vs design string**

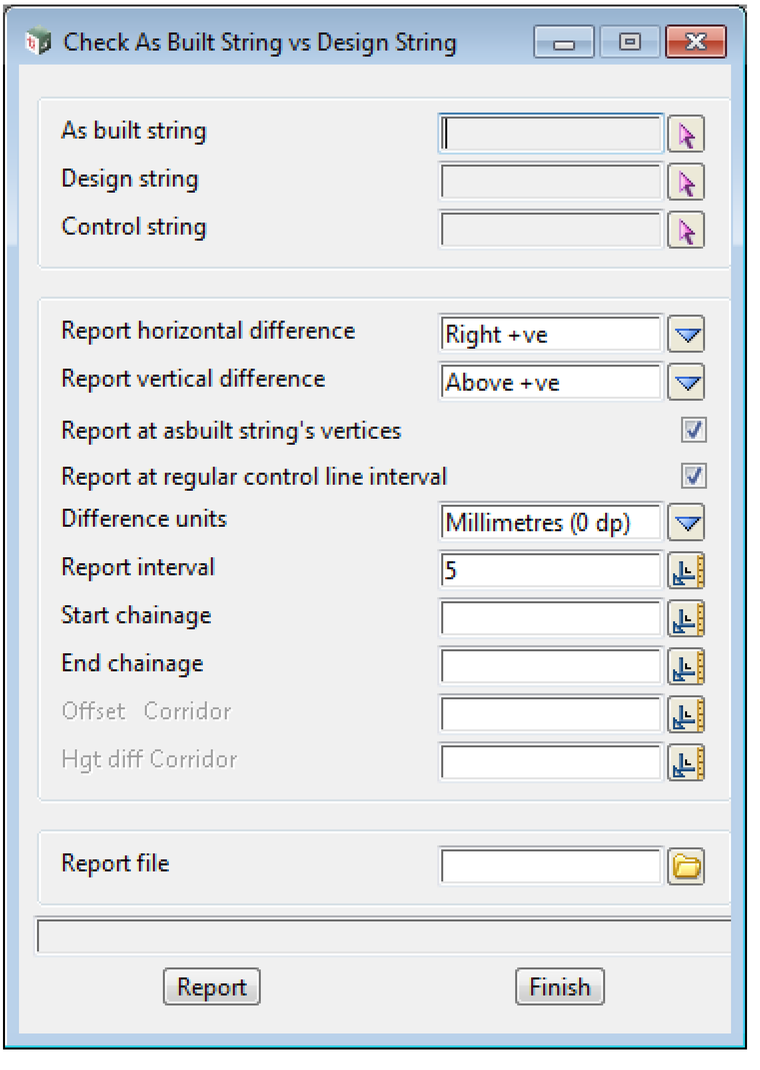
This panel is used to check the horizontal and vertical differences between two strings (usually "as built" and design). A control string is selected to provide the chainage and line to cut the two strings and calculate the difference along. The control string can be one of the two strings.

The corridors are an optional setting for where multiple cuts of the as built and design strings are possible.

E.g. If string crosses over itself or turns through more than 90° a section normal to the control string could cut it more than once.

If the user sees warning messages such as "> 1 cut" then use these settings to filter out the multiple cuts.

Selecting the **Check asbuilt string vs design string** brings up the **Check As Built String vs Design String** panel.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
As built string	string select		
string to compare			
Design string	string select		

string to compare

Control string string select

string to use to define chainage and right angles. The line at right angles cuts the "as built" and "design" strings and is used to calculate the horizontal and vertical differences.

Report horizontal difference choice box Right +ve None, Right +ve, Right -ve

*if **None*** *Don't report horizontal distances.*

*if **Right +ve*** *Horizontal differences are reported with differences to the right reported as +ve*

*if **Right -ve*** *Horizontal differences are reported with differences to the right reported as -ve*

Report vertical difference choice box Above +ve None, Above +ve, Above -ve

*if **None*** *Don't report horizontal distances.*

*if **Above +ve*** *Vertical differences are reported with differences above reported as +ve*

*if **Above -ve*** *Vertical differences are reported with differences above reported as -ve*

Report at asbuilt string's verticestick box tick

if tick, the differences are reported at the vertices of the as built string.

Report at regular control line intervaltick box tick

if tick, the differences are reported at the chainage interval given in the "Report interval" box.

Difference units choice box Millimetres (0 dp) Millimetres (0 dp)
Millimetres (1 dp)
Metres (3 dp)
Metres (4 dp)

*if **Millimetres (0 dp)** In the report, differences are printed out as Millimetres with 0 decimal places*

*if **Millimetres (1 dp)** In the report, differences are printed out as Millimetres with 1 decimal place*

*if **Metres (3 dp)*** *In the report, differences are printed out as Metres with 3 decimal places*

*if **Metres (4 dp)*** *In the report, differences are printed out as Metres with 4 decimal places*

Report interval input box 5

chainage interval of the control string to report the differences at.

Start chainage input box

start chainage of the control string to start reporting differences.

End chainage input box

end chainage of the control string to stop reporting differences.

Offset Corridor input box

If this value is set a cut offset greater than this from the design string is ignored.

Hgt diff Corridor input box

If this value is set a cut hgt difference greater than this from the design string is ignored

Report file input box

name of the report file.

Report button

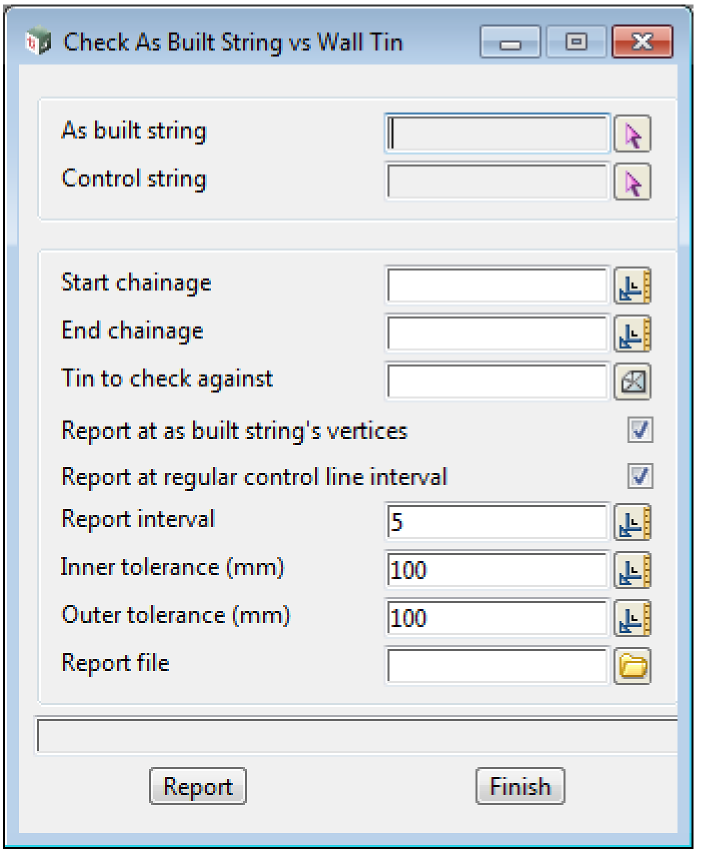
run the option.

27.10.2 Check Asbuilt String vs Wall Tin

Position of option on menu: **Report =>QA Reports =>Check asbuilt string vs wall tin**

This panel is used to check the horizontal difference between a string (usually the “as built”) and a tin (usually a wall tin). A control string is selected to provide the chainage and bearing to project a horizontal line to cut the string and the tin and calculate the horizontal difference along. The control string can be the “as built” string.

Points that are different by a given inner and outer tolerance are flagged in the report.



The fields and buttons used in this panel have the following functions:

Field Description | Type | Defaults | Pop-Up

As built string string select

string to compare against the wall tin.

Control string string select

string to use to define chainage and right angles. The horizontal line at right angles cuts the “as built” string and “tin to check against” tin that is used to calculate the horizontal differences.

Start chainage input box

start chainage of the control string to start reporting differences.

End chainage input box

end chainage of the control string to stop reporting differences.

Tin to check against input box available tins

tin to compare the as built string with.

Report at as built string's verticestick box tick

if tick, the differences are reported at the vertices of the as built string.

Report at regular control line intervaltick box tick

if tick, the differences are reported at the chainage interval given in the "Report interval" box.

Report interval input box 5

chainage interval of the control string to report the differences at.

Inner tolerance (mm) input box 100

if the inner horizontal distance between the string and the wall is greater that this value (in units times 1000) then it is flagged in the report.

Outer tolerance (mm) input box 100

if the outer horizontal distance between the string and the wall is greater that this value (in units times 1000) then it is flagged in the report.

Report File input box

name of the report file.

Report button

run the option.

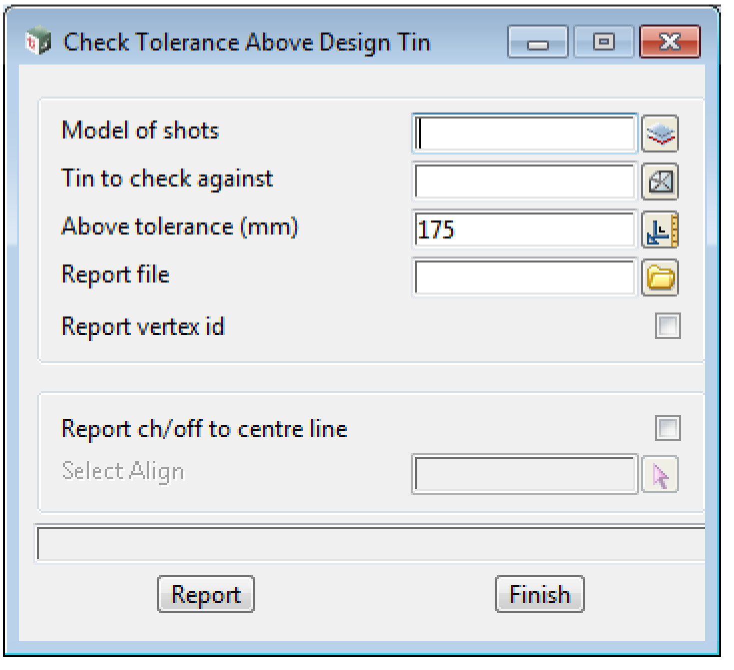
27.10.3 Check Points Above a Tin

Position of option on menu: **Report =>QA Reports =>Check points above tin**

This panel is used to check the z-value of points against the z-value of a tin at the same (x,y) locations and only report on those above the tin. To check points above and below a tin, go to the next section 27.10.4 Check Points vs Tin.

A z-tolerance above the tin is given and all points above the tin and outside the tolerance are especially flagged in the report.

Optionally the report can also include the chainage and offset of the points from a selected alignment string.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Model of shots	model box		

model of points to check the z-value against the tin.

Tin to check against	tin box		

tin to check the z-values against

Above tolerance (mm)	input box	175	

*if the z-value of the point is **above** the tin and the difference of the z-value of the point and the tin (times 1000), is greater than this amount then it is flagged in the report.*

Report file	file box		

name of the report file.

Report ch/off to centre line	tick box		

if tick, the chainage and offset of the points from the selected alignment string are included in the report.

Select align	string select		

select the alignment string to calculate offset and chainage from.

Report

run the option.

27.10.4 Check Points vs Tin

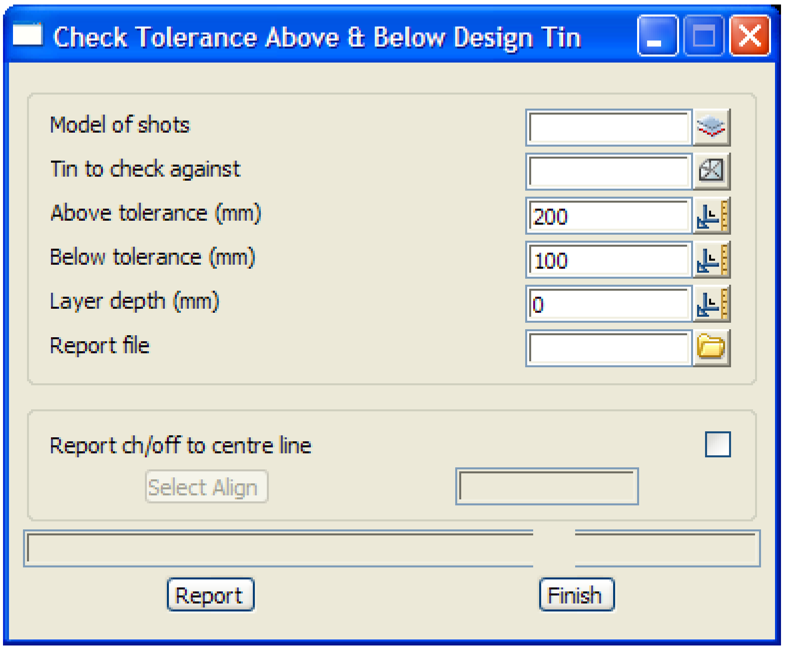
Position of option on menu: **Report =>QA Reports =>Check points vs tin**

For **Check points above tin**, go to 27.10.3 Check Points Above a Tin.

This panel is used to check the z-value of points against the z-value of a tin at the same (x,y) locations, or if a **Layer depth** is given, the z-values of points are checked against the tin *minus* the layer depth.

Separate z-tolerances are supplied for ***above*** and ***below*** the tin and points outside either tolerance are especially flagged in the report.

Optionally the report can also include the chainage and offset of the points from a selected alignment string.



The fields and buttons used in this panel have the following functions:

Field Description   Type   Defaults   Pop-Up

Model of shots   model box

model of points to check the z-value against the tin.

Tin to check against   tin box

tin to check the z-values against

Above tolerance (mm)   input box   175

*if the z-value of the point is **above** the tin and the difference of the z-value of the point and the tin (times 1000), is greater than this amount then it is flagged in the report.*

Below tolerance (mm)   input box   175

*if the z-value of the point is **below** the tin and the absolute value of the difference of the z-value of the point and the tin (times 1000), is greater than this amount then it is flagged in the report.*

Layer depth   input box   0

this value is subtracted form the z-value of the tin before the comparison is made with the z-value of the point.

Report file   file box

name of the report file.

Report ch/off to centre linetick box

if tick, the chainage and offset of the points from the selected alignment string are included in the report.

Select align string select

select the alignment string to calculate offset and chainage from.

Report

run the option.

27.10.5 Check Survey Points vs Design Points (Old)

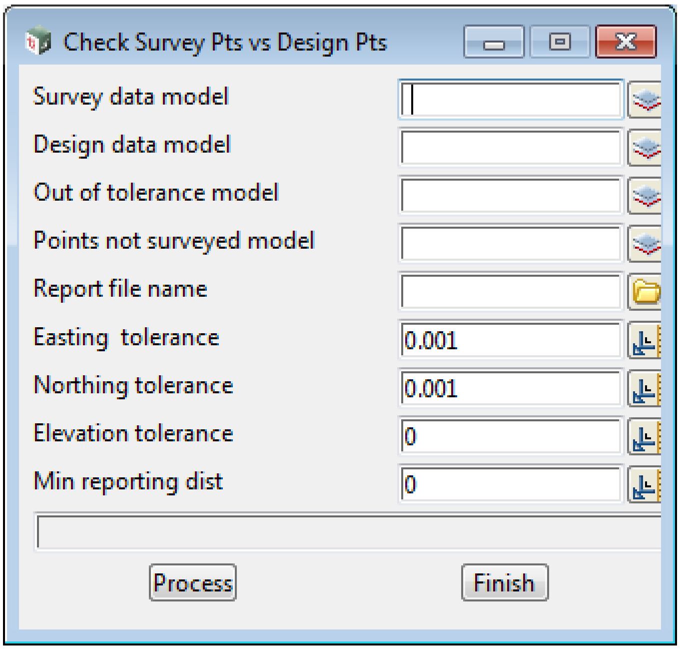
Position of option on menu: **Report =>QA Reports =>Check survey points vs design points (old)**

This option is used to check surveyed points against the design points.

Separate tolerances are given for Eastings (x), Northings (y) and Elevations (z).

Any design points that have not been surveyed are noted and copies of points out of tolerance and/or points not surveyed can be automatically made.

NOTE - if a ***radial*** search distance is required in (x,y), go to the next section 27.10.6 Check Survey Points vs Design Points (3).



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Survey data model	model box		available models

model of points to be compared against the design points.

Design data model	model box		available models

points to be compared against.

Out of tolerance model	model box		available models

copies of any points out of tolerance are added to this model.

Points not surveyed model input box

copies of any points from the design model that have not been surveyed.

Report file name	input box		

name of the report.5

Easting tolerance	input box	0.001	

tolerance in the x direction.

Northing tolerance	input box	0.001	

tolerance in the y-direction.

Elevation tolerance	input box	0

tolerance in the z-direction.

Min reporting dist	input box	0

Process	button

run the option.

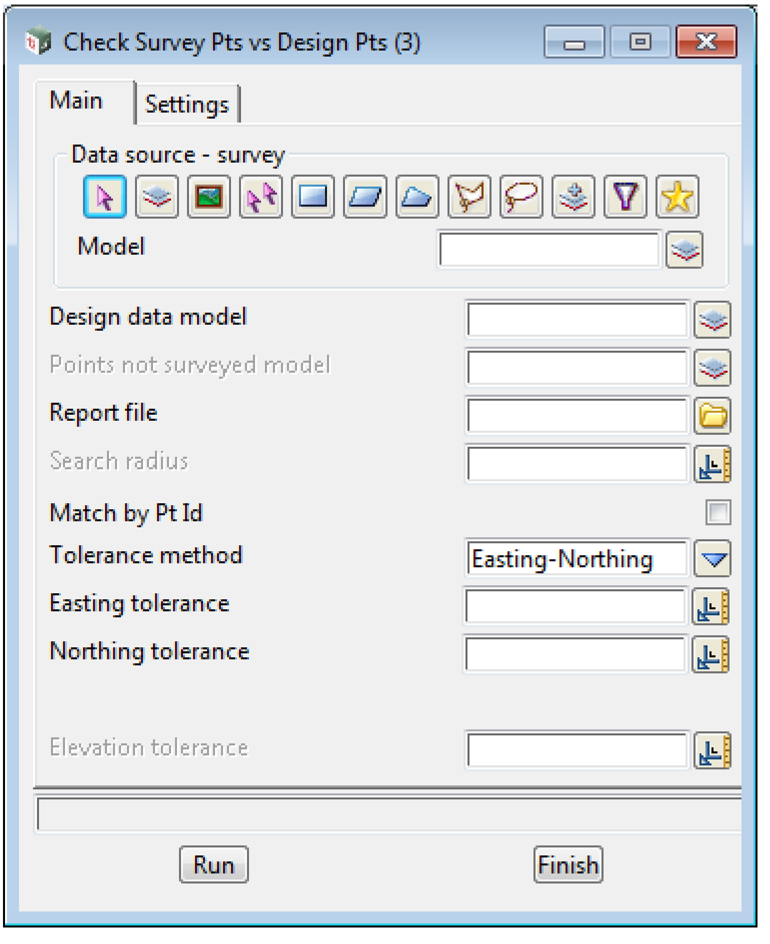
27.10.6 Check Survey Points vs Design Points (3)

Position of option on menu: **Report =>QA Reports =>Check survey points vs design points**

This option is used to check surveyed points against the design points.

The (x,y) search distance can be either one radial distance, a separate x and y distance or separate chainage and offset distances.

Any design points that have not been surveyed are noted and can be optionally copied to a model.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up

Main Tab

Data source type - survey — Defaults: Model

data selection type - for a full description go to 4.19.3 Data Source.

Data source — Type: input

source of data to be processed.

Design data model — Type: model box — Pop-Up: available models

Model containing design points for comparison against.

Points not surveyed model	input box		

Copies of any points from the design model that have not been surveyed.

Report file	file box		

Name of report file for results

Search radius	measure box		At Point, Point to Point, String from Point,String to Point

if set surveyed/design points are matched when less than this distance apart.

Match by Pt Id	tick box		

if ticked surveyed/design points are matched by point id.

Tolerance method	choice box	Easting-Northing	Easting-Northing, Distance,Chainage-Offset

*if **Chainage-Offset**, separate chainage and offset tolerances relative to a selected string are given to check against.*

*if **Easting-Northing**, separate Easting (x) and Northing (y) tolerances are given to check against.*

*if **Distanc**e, the test is for a radial distance, not a separate tolerances.*

Easting tolerance	measure box		At Point, Point to Point, String from Point, String to Point

The tolerance in the x-direction

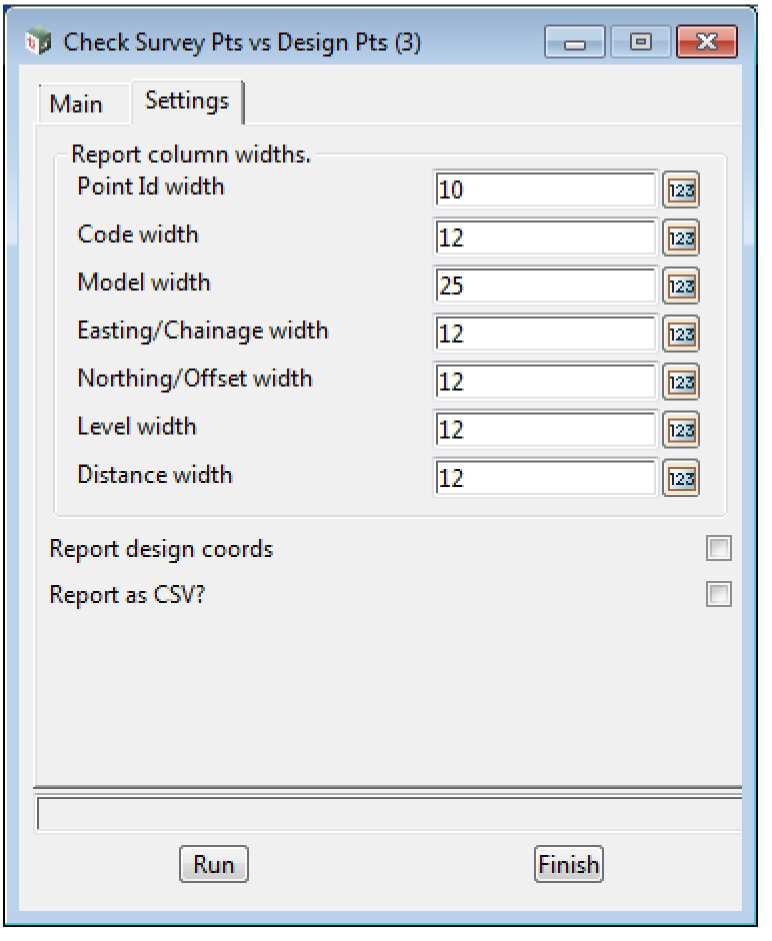
Northing tolerance	measure box		At Point, Point to Point, String from Point, String to Point

The tolerance in the y-direction

Elevation tolerance	measure box		At Point, Point to Point, String from Point, String to Point

The tolerance in the z-direction

Settings Tab



This tab allows some settings in controlling the report.

Report Column Widths

Point Id width input 10

Width of point id column.

Code width input 12

Width of code column, (0 removes column)

Model width input 25

Width of model column, (0 removes column)

Easting /Chainage width input 12

Width of easting/chainage column.

Northing /Offset width input 12

Width of northing/offset column.

Level width input 12

Width of level column.

Distance width input 12

Width of distance column.

Report design coords

if ticked design coordinates will be written to the report file

Report as CSV?

If **Report as CSV?** *is ticked then the column width settings are ignored and the report written as a comma delimited 'CSV' file suitable for exports to spreadsheets and others.*

If **Report as CSV?** *is not ticked then the reported is formatted with the column width figures in mind.*

Run button

run the option.

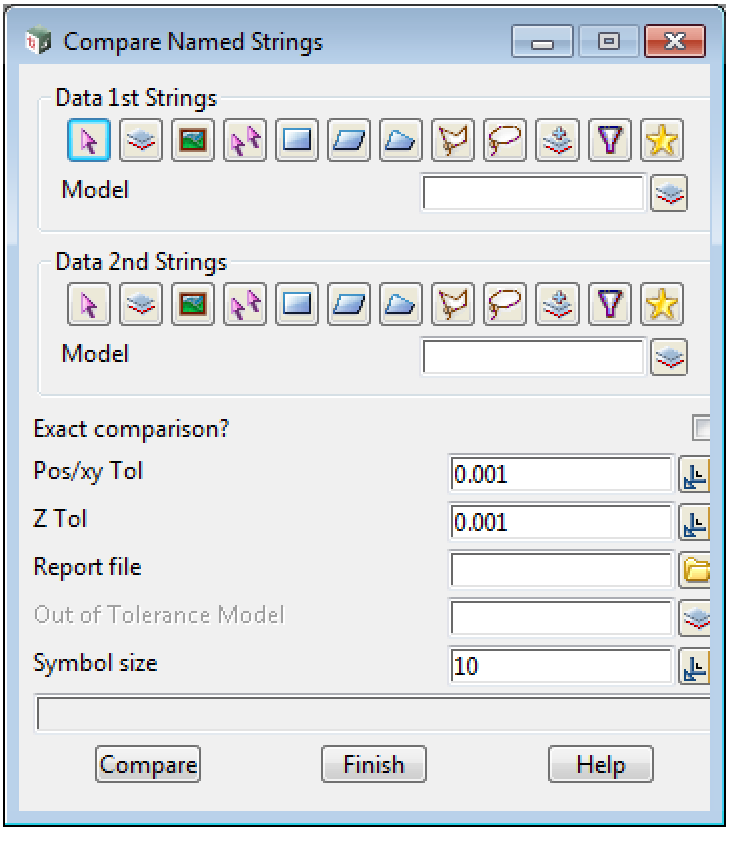
27.10.7 Compare Named Strings

Position of option on menu: **Report =>QA Reports =>Compare named strings**

The **compare named strings** panel is a utility to check for changes in a string and highlights these changes. It would typically be used when a revised design is released to highlight where the actual changes are.

Strings are matched by name, there must only be one string of the same name in each of the data sources.

Selecting **Compare named strings** brings up the **Compare Named Strings** panel.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Data 1st Strings

the data source for the 1st set of strings.

data selection type - for a full description go to 4.19.3 Data Source.

Data 2nd Strings

the data source for the set of strings to compare to the 1st set of strings.

data selection type - for a full description go to 4.19.3 Data Source.

Exact comparison? tick box

if ticked the strings are compared vertex by vertex

if unticked the points in the 2nd string are dropped to the first string and the comparison is done by offset and height difference.

Pos/xy Tol measure box

The distance between vertices or the difference in offset before the point is considered different.

Z Tol measure box

The height difference between the strings before the point is considered different.

Report file file box

The file to which the results of the comparison are written, contains the strings compared, the number of name matches if not matched or unique and the details of differing points.

Out of Tolerance Model model box

Any points outside the nominated tolerances can be written to this model.

Symbol size input

Points in the ***out of tolerance model*** *have a diamond of this real world size drawn around them to highlight the points.*

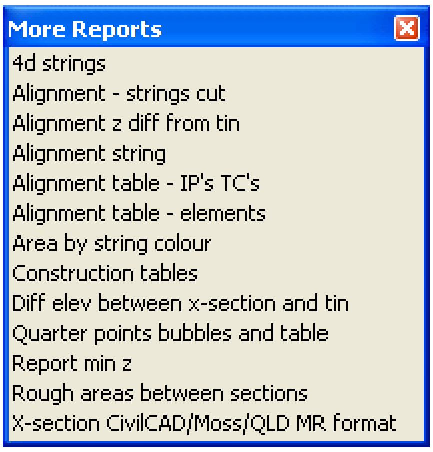
Compare button

Compares the two data sources

27.11 More Reports

Position of menu: **Report =>More**

More miscellaneous reports.



report on 4d strings
report the strings that an alignment cuts
give the differences from a tin to a string
report on an alignment string
create alignment report and/or table giving IP's and CT's
create alignment report and/or table giving elements
give plan areas summed by string colour
give the differences from the points on a section to a tin
report and/or table of quarter points for an arc
report the minimum values on a x-section
report of areas between sections

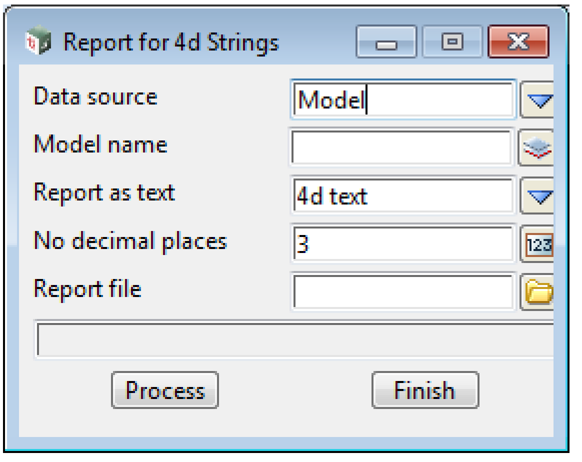
For the option/menu *4d string*, go to	27.11.1 Report for 4d Strings
Alignment - strings cut	27.11.2 Alignment - Strings Cut
Alignment z diff from tin	27.11.3 Report Z Differences from Alignment to Tin
Alignment string	27.11.4 Alignment Report
Alignment table - IP's TC's	24.16.15 Tabulate Alignment - IP's and CT's
Alignment table - elements	24.16.16 Tabulate Alignment - Elements
Area by string colour	27.11.7 Total of Plan Area by String Colour
Diff elev between x-sections and tin	27.11.8 Report Z Differences From X-Sections to a Tin
Quarter points bubble and table	17.18.3 Setout Lip Line
Report min z	27.11.10 Report Minimums Z-Value on a Cross Section
Rough areas between sections	27.11.11 Report Rough Areas
Super string	27.2.6 Report for Super Strings
X-section CivilCAD/Moss/Qld MRD format	27.11.12 X-Sections Report

The menu options will now be discussed.

27.11.1 Report for 4d Strings

Position of option on menu: **Report =>More =>Report for 4d strings**

This panel is used to create a report for 4d strings where the x y and z values are printed out for each point plus either the 4d text for the point or the name of the string (repeated for each point).



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Data source	choice box	model	string, model, view

data source type.

Model/View/String	data source		

Data source to do report of 4d strings on.

Report as text	file	4d text	string name 4d text

if 4d text, the 4d text for the point is written out after the x, y and z values.

If string name, the string name is written out after the x, y and z values.

No decimal places	input	3	

number of decimal places to use in the x, y and z values.

Report file	file		

name of the report file.

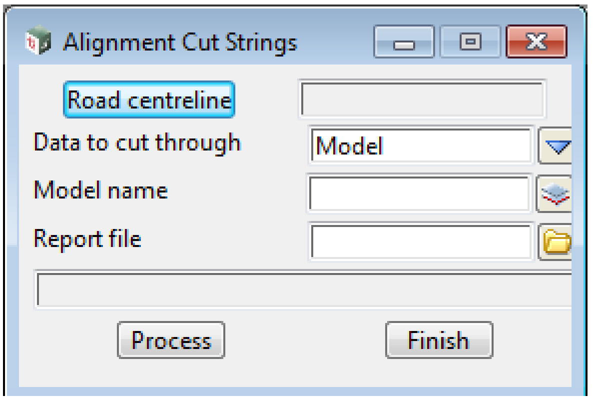
Process	button		

run the option.

27.11.2 Alignment - Strings Cut

Position of option on menu: **Report =>More =>Alignment - strings cut**

This option is used to create a report of the position and angle that strings make where they cut a selected alignment string.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Road centreline	string select		

select the alignment string to report the cuts for.

Data source	choice box	Model	String/Model/View

source of the data

Data

method a specifying data

Report file	input box		*.rpt files

name of the report file.

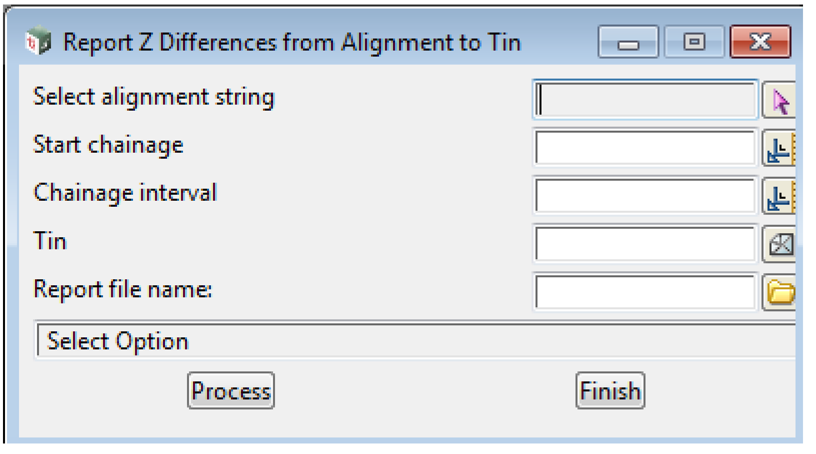
Process	button		

run the option.

27.11.3 Report Z Differences from Alignment to Tin

Position of option on menu: **Report =>More =>Report z differences from alignment to tin**

This panel is used to create a report of the difference in z-values between an alignment string and a tin. The differences are reported at a given chainage interval along the alignment string.



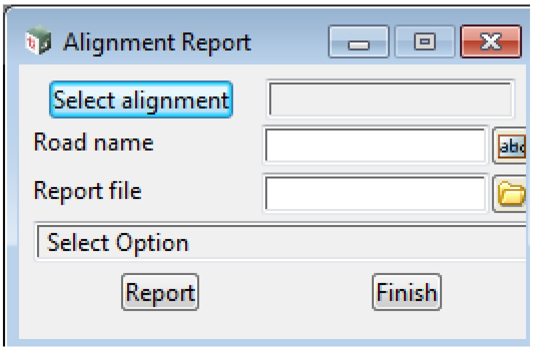
The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Select alignment string	string select		
select the alignment string to report z-differences for.			
Chainage interval	input box		
interval to report on.			
Tin	input box		available tins
tin to get z-values from.			
Report file name	input box		*.rpt files
name of the report file.			
Process	button		
run the option.			

27.11.4 Alignment Report

Position of option on menu: **Report =>More =>Alignment string**

This panel is used to report the tangent points and centres of arcs for an alignment string.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Select alignment string	string select		

select the alignment string to report on.

Report file name	input box		

name of the report file.

Report	button		

run the option.

27.11.5 Alignment Table - IP's, TC's

Position of option on menu: **Report =>More =>Alignment table - IP's and TC's**

Create table of IP and CT points for an alignment string.

This panel has already been documented.

See 24.16.15 Tabulate Alignment - IP's and CT's in 24 Drafting

27.11.6 Alignment Table - Elements

Position of option on menu: **Report =>More =>Alignment table - elements**

Create table of horizontal elements for an alignment string.

This panel has already been documented.

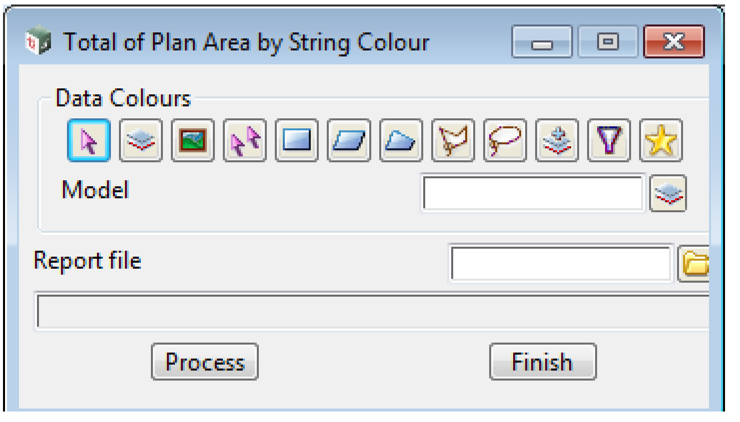
See 24.16.16 Tabulate Alignment - Elements in 24 Drafting

27.11.7 Total of Plan Area by String Colour

Position of option on menu: **Report =>More =>Area by string colour**

This panel is used to calculate the sum of the plan area of all strings of the same colour. This is especially useful for calculating the plan areas of faces produced for slope areas, volumes etc.

A report giving the total plan area for each different colour is produced.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Data source	data source	model	model, view

Type of data to calculate areas for.

Model/View data source

Data source to calculate the areas and sum by colour.

Report file input box

Report giving the total plan areas of the strings for each colour.

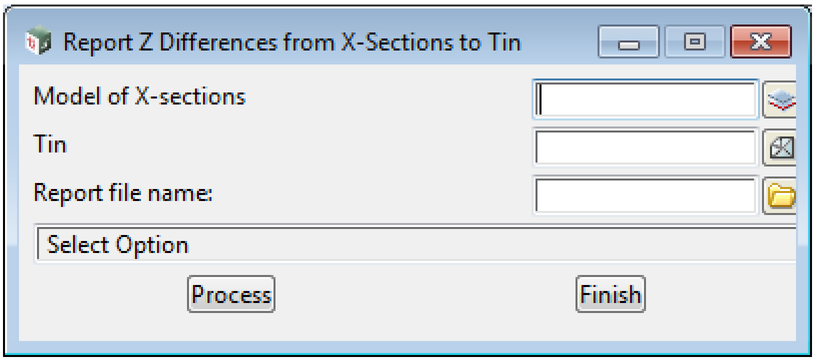
Process button

Run the option

27.11.8 Report Z Differences From X-Sections to a Tin

Position of option on menu: **Report =>More =>Diff elev between x-section and tin**

This panel is used to report the z differences between the points on cross sections and the corresponding z-values from a tin.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Model of X-Sections	model box		available models

model of cross sections to report on.

Tin	tin box		

tin to get z-values from.

Report file name	file box		

name of the report file.

Process	button		

run the option.

27.11.9 Quarter Points Report

Position of option on menu: **Report =>More =>Quarter points bubbles and table**

This option is used to create bubbles and/or a report for the critical horizontal and vertical points and quarter points (by chord or by chainage) for any arcs in an alignment string.

This option has already been documented as

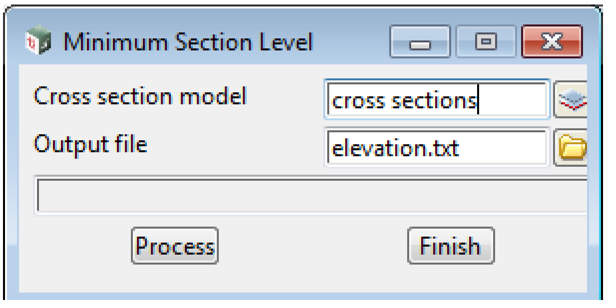
Survey =>Setout =>Setout lip line

in the section 17.18.3 Setout Lip Line.

27.11.10 Report Minimums Z-Value on a Cross Section

Position of option on menu: **Report =>More =>Report min z**

This panel creates a report giving the minimum z-value on a cross section.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Cross section model	model box		

model of cross sections to report on.

Output file	file box		*.txt files

file to write section names and minimum z-values to.

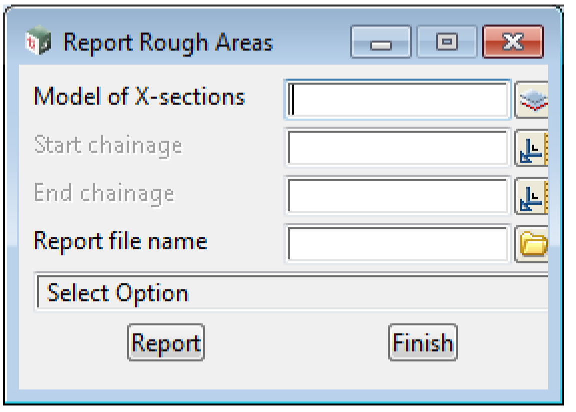
Process	button		

run the option.

27.11.11 Report Rough Areas

Position of option on menu: **Report =>More =>Rough areas between sections**

This panel is used to



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Model of x-sections	input box		

xxxxx

Start chainage	input box		

xxxxx

End chainage	input box		

xxxxx

Report file name	input box		

name of the report file.

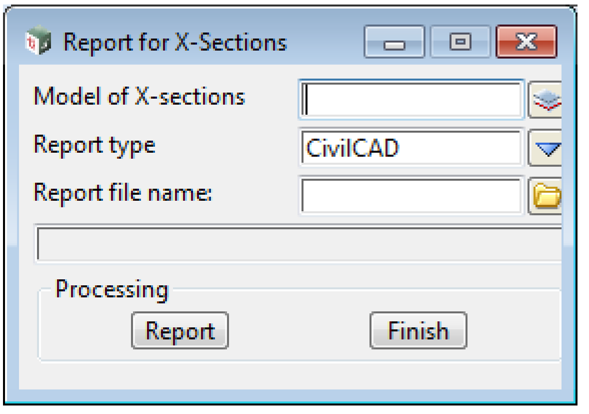
Report	button		

run the option.

27.11.12 X-Sections Report

Position of option on menu: **Report =>More =>X-section CivilCAD/Moss/QLD MR format**

This panel is used to write out cross sections in a variety of formats.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Model of x-sections	input box		available models

model of cross sections to write out.

Report type	file box	CivilCAD	CivilCAD, Moss MainRoad Qld, Brief

format of the report.

Report file name	input box		

name of the report file.

Report	button		

run the option.

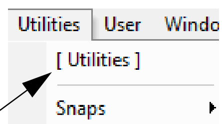
28 Utilities

Position of menu: **It is on the main menu as Utilities**

The **Utilities** menus contains a collection of useful operations that don't easily fit under any of the other menus.

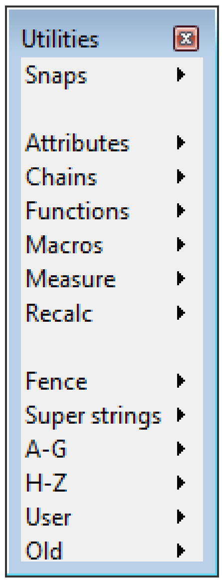
For ease of selection, the frequently used utilities are placed on the first level of the menu. All the other utilities are split alphabetically between two walk-right menus - the utilities walk-right menus for A-G and H-Z.

on Main menu



create floating **Utilities** menu

on 12d Model menu and floating Utilities menu



snaps

edit, create, delete, *etc.* for attributes
edit, create, delete, *etc.* for chains
edit, create, delete, *etc.* for functions
compiling and running macros.
measure distances, angles, *etc.*
re-calc functions

create inside and outside of a polygon
options to modify super strings
utilities from A to G
utilities from H to Z
Utilities User menu
old options

For the option/menu

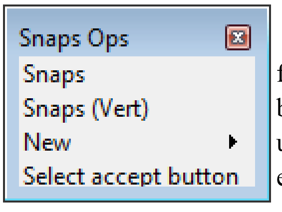
Snaps, go to	28.1 Snaps
Attributes	28.2 Attributes
Chains	28.3 Chains
Functions	28.4 Functions
Macros	28.5 Macros
Measure	28.6 Measure
Recalc	28.7 Recalc
Fence	28.8 Fence
Super strings	14.12.9 Super Strings
A-G	28.9 Utilities A-G
H-Z	28.10 Utilities H-Z
Old	28.11 Old

28.1 Snaps

Position of menu: **Utilities =>Snaps**

The **snaps** option allows the user to set snapping modes for use during pick operations. The snapping modes have been discussed in detail in the 4.13 Snaps section.

The **Snaps** walk-right menu provides a full snap menu, abbreviated snap menus laid out either horizontally or vertically, and a snaps cogo menu for creating special point positions using co-ordinate geometry commands.



full snaps menu
brief snaps laid out vertically
under development
equivalent to middle mouse button (accept)

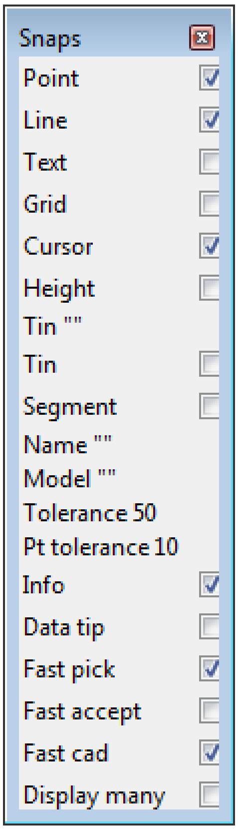
For the option

Snaps, go to	28.1.1 Snaps
Snaps (Vert)	28.1.2 Snaps (Vert) and H Toolbar
Cogo	28.1.3 Snaps Cogo
New	28.1.5 Snaps New
Select accept button	28.1.4 Select accept button

28.1.1 Snaps

Position of option on menu: **Utilities =>Snaps =>Snaps**

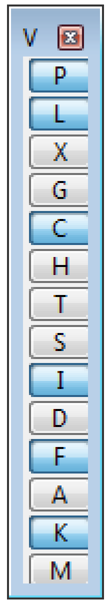
The **Snaps** menu provides tick boxes to turn the various snap modes on or off and options to set tin, string and model names. Any combination of snap modes and names can be set.



Snaps menu

point snap on/off
line snap on/off
text snap on/off
grid snap on/off
allow cursor point on/off
ask for z value at each point in edits
name of tin to snap to
tin snap on/off
turn segment snap on/off
restriction on string names
restriction on model names
set snap tolerance
set point snap tolerance
bring up info panel on/off
turns data tips on/off
turns fast pick on/off
turns fast accept on/off
K - accept results of snaps CAD without MB
picks bring up selection grid

Snaps V menu



The snaps are set on or off using the toggles on the **Snaps H** toolbar, or the tick boxes on the **Snaps** or **Snaps V** menu.

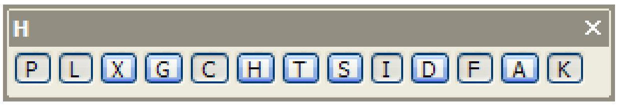
For more information on what all the snaps in the Snaps menu are, go to the 4.13 Snaps and 4.13.2 Setting Snaps section.

28.1.2 Snaps (Vert) and H Toolbar

Position of option on menu: **Utilities =>Snaps =>Snaps (Vert)**

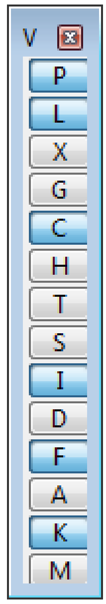
The **Snaps (vert)** menu provides the same tick boxes as the **Snaps** menu to turn the various snap modes on or off, however the snap descriptions are abbreviated to one character to reduce the menu size. There is also the **H** toolbar which is a horizontal version of the Snaps menu.

The **H** toolbar and Snaps (vert) menu are H toolbar which is



The **Snaps** are

P Point snap (on)
L Line snap (on)
X Text snap (off)
G Grid snap (off)
C Cursor snap (on)
H Height snap (off)
T Tin snap (off)
S Segment snap (off)
I Info snap (on)
D Data tip (off)
F Fast accept (on)
A Fast accept (off)
K Construction snap (on)
M Show selection grid on picks



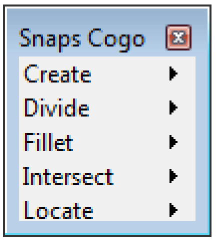
For more information on snaps, go to the 4.13 Snaps and 4.13.2 Setting Snaps section.

28.1.3 Snaps Cogo

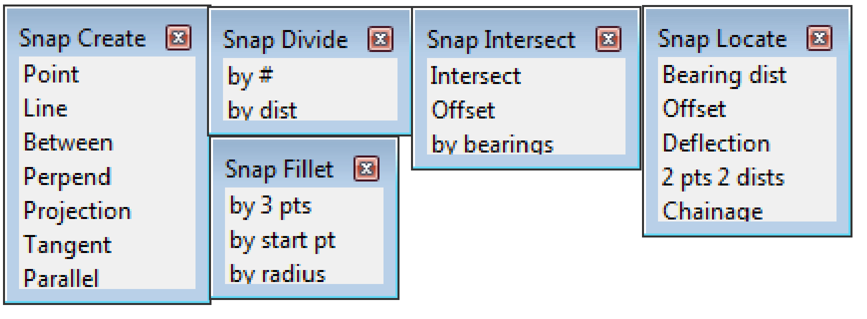
Position of menu: **Utilities =>Snaps =>Cogo**

The **Snaps cogo** options are used to create the point required for any 12d Model select operation. For example, selecting the next point in **append** in an editor. The result of the **Snaps cogo** is returned to the **select operation** as the selected point.

Walking right on the **cogo** option brings up the **Snaps cogo** menu



and the walk-right menus for each of the **Snaps cogo** menu items are



Note on Combining Snaps Cogos

When using a snaps cogo option, another select is often required and it is permissible to use other snaps cogo option to create the point for that select.

For example, the required point is the point halfway between two other points **dropped perpendicularly** onto an arc.

Such a combined result can be achieved using **Snaps cogo** however the order of selecting the **snaps cogo** options may at first seem to be the reverse of what is required.

For the above example, the snaps cogo **perpen** must be selected before the **between** option.

This is because when the **perpen** snap cogo option is selected, it asks the user to **select** the point to drop and it is then that the snaps cogo **between** option is used to find the midpoint of two selected point. The result of the **between** then becomes the point to drop for the **perpen**.

If the **between** option had been selected first, the resultant midpoint would have been returned as the selected point and that select completed.

28.1.4 Select accept button

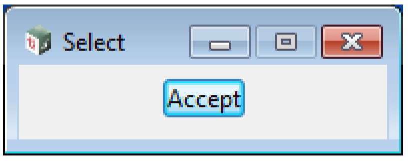
Position of menu: **Utilities =>Snaps =>Select accept button**

Instead of selecting (MB) to accept a string selection, you can click (LB) on the **Accept** button.

This option is usually used for tablet computers without a middle mouse button.

LB = the left mouse button

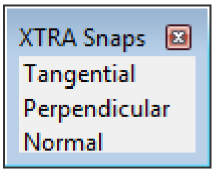
MB = the middle mouse button



28.1.5 Snaps New

Position of menu: **Utilities =>Snaps =>New**

The **XTRA Snaps** menu is currently under development.

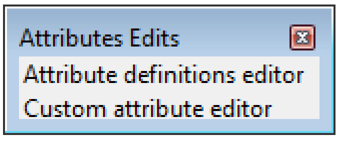


28.2 Attributes

Position of menu: **Utilities =>Attributes**

This section of documentation is a work in progress and will be updated in subsequent releases.

The **Attributes** walk-right menu is:



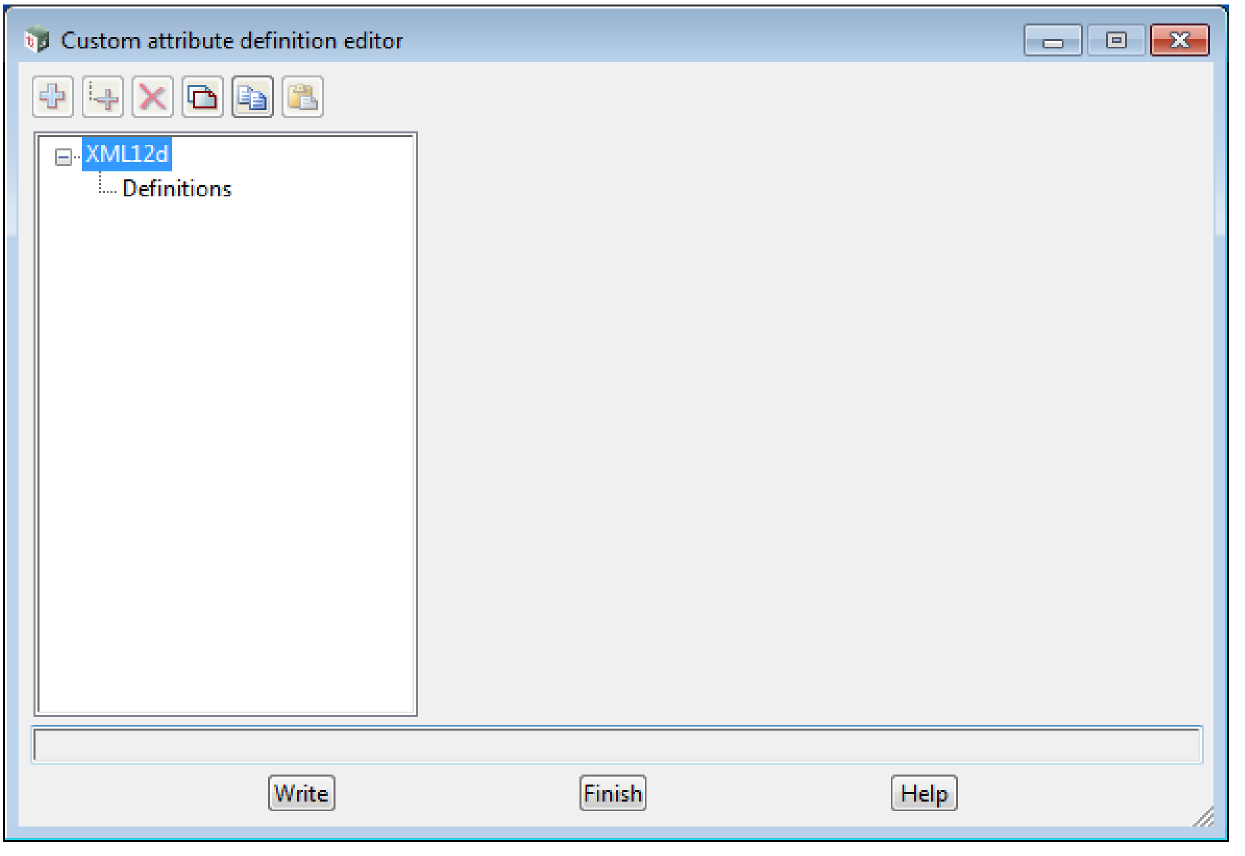
edit attribute definitions
edit custom attributes

For the *Attribute definitions editor*, go to 28.2.1 Attribute Definitions Editor
Custom attribute editor 28.2.2 Custom Attribute Editor

28.2.1 Attribute Definitions Editor

Position of option on menu: **Utilities =>Attributes =>Attributes Edits =>Attribute definitions editor**

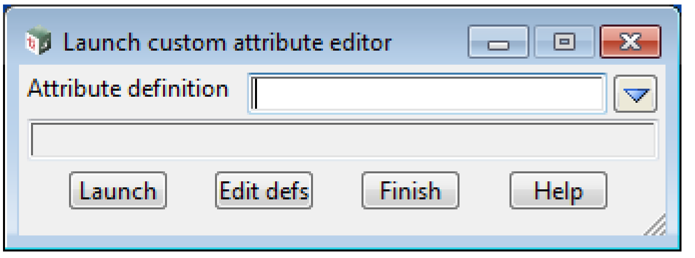
This section of documentation is a work in progress and will be updated in subsequent releases.



28.2.2 Custom Attribute Editor

Position of option on menu: **Utilities =>Attributes =>Attributes Edits =>Custom attribute editor**

This section of documentation is a work in progress and will be updated in subsequent releases.



28.3 Chains

Position of menu: **Utilities =>Chains**

12d Model has many different built in objects, from simple strings, through to complex super alignments with computators, drainage networks, super tins and Apply MTF functions.

However as complex as these objects are, no software can foresee every situation met in surveying and civil design, both now and into the future.

12d Model provides two tools to help users extend the supplied ***12d Model*** functionality and automate simple and complex tasks within a company. They are the

(a) the ***12d Model Programming Language*** (12dPL) - a full programming language so users can write complex tailored options and functions (called macros)

and

(b) ***chains***.

Chains are a relative simple method for users to combine and run existing ***12d Model*** functions and options. Anyone who can use ***12d Model*** can create and run a chain.

In its simplest form, a ***Chain*** is a list of items that are run in the order that they appear in the list. Things that can be run in a *Chain* include:

(a) ***12d Model*** functions

(b) other chains

(c) super alignment resolves

(d) plot parameter files

(e) most ***12d Model*** options

(f) most screen layout files

(g) most macros, and certainly those that don't have a panel

(h) an external command line

(i) some special hardwired chain commands such as delete a tin, delete all tins, delete a model, delete all models, clean a model, delete a file and compare files.

For more sophisticated applications, chains can also

(a) create ***12d Model*** views and add and remove models from the views

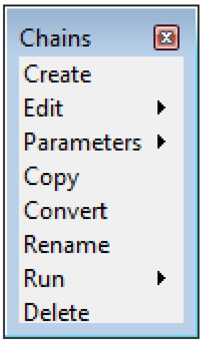
(b) minimise, maximise, redraw and delete views

(c) include flow control using tests, labels and go to's

(d) be parametrised and run with user supplied parameter values

So a *chain* can be as simple as running a list of functions.or as complex and making decisions inside the chain, thus creating user defined ***Super Objects***.

The **Chains** walk-right menu is



create a chain
edit a chain
parameterise a chain
copy a chain
convert a chain
rename a chain
run a chain
delete a chain

For the option

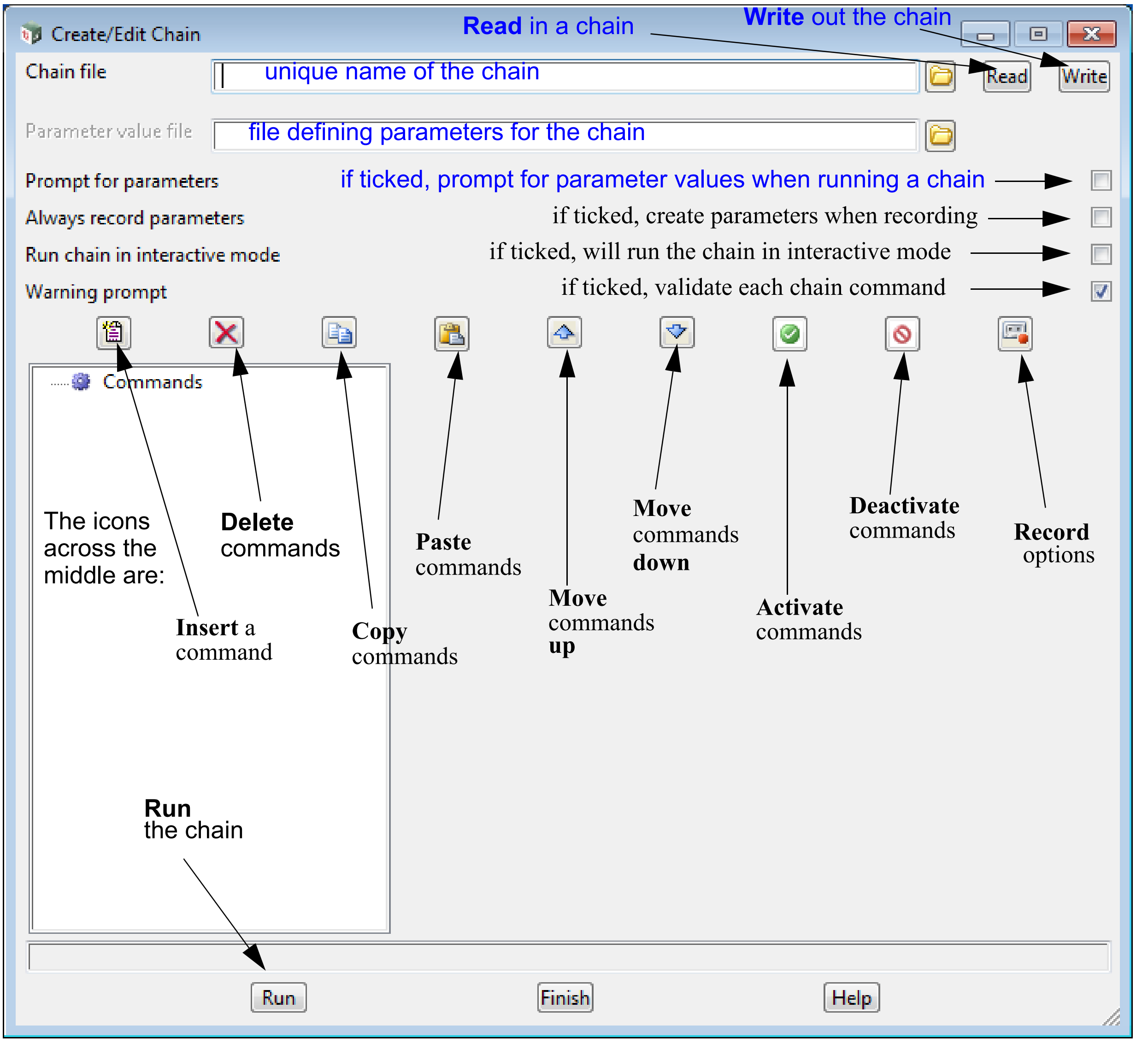
Create	28.3.1 Create/Edit a Chain
Edit	28.3.3 Edit a Chain
Parameters	28.3.4 Chain Parameters
Copy	28.3.6 Copy a Chain
Convert	28.3.7 Chain Conversion
Rename	28.3.8 Rename a Chain
Run	28.3.9 Run a Chain
Delete	28.3.10 Delete a Chain

28.3.1 Create/Edit a Chain

Position of option on menu: **Utilities =>Chains =>Create**

The **Create/Edit** option creates new chains but can also edit an existing chain, or run an existing chain.

Selecting **Create** brings up the **Create/Edit Chain** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Chain file	file box	blank	available *.chain, *.rcn files

name of the file for the chain.

*To **create** a new chain, a unique name is typed into the **Chain file** box and the **Write** button clicked.*

*To **edit** an existing chain, enter the name of the existing chain into the **Chain file** box and then click on the **Read** button.*

*The chain is then built up by inserting commands using the **Insert** icon.*

Read	button

when pressed, read in the given chain file

Write button

when presses, write out all the chain commands in the list to the given chain file.

*After each item is added to the chain, click on **Write** to save the updated chain.*

Parameter value file file box blank existing parameter files

*if **not blank**, the file to use for mapping parameters and values in the Chain.*

*If **blank**, no parameter file is used*

For more information, go to the section 28.3.4 Chain Parameters.

Prompt for parameters tick box

*if **ticked**, when a with parameters chain is run, the table of parameters is displayed on the screen before running the chain so that any required modifications can be made before the chain is run.*

*If **not ticked,** any values for parameters are taken from the parameter value file.*

Always record parameters tick box

*If **ticked** and the user records panels or options, the recorder will attempt to record parameters.*

Run chain in interactive mode tick box

*if **ticked**, run the chain in **Interactive Mode**. See* 28.3.2 Interactive Mode.
*If **not ticked**, the chain runs normally.*

Important Note and Warning

***Run chain in interactive mode** has no connection to the* **Run in interactive mode** *that is available when many panels are recorded in a chain. When a recorded panel in the chain has* **Run in interactive mode** *ticked on, then when that panel runs in the chain, the panel is displayed and the user can change values in the panel fields and then run/not run the panel. When the panel is closed the chain continues running.*

Warning prompt tick box

*if **not ticked**, validations of the inserted commands are not performed straight away. For example, if you press **Insert** many times, you won't get the warning prompt "Some changes you have made are invalid. Do you wish to change nodes and discard your changes?". Also a confirmation prompt does not come up when you delete commands from the chain.*

*if **ticked**, inserted commands are validated and warning messages and error messages will be generated.*

Icons

Insert icon

inserts commands into the chain. For more information on the commands, go to 28.3.1.1 Inserting a Chain Command

Delete icon

deletes the highlighted commands in the chain

Copy icon

copies the highlighted commands in the chain so that they can be pasted back into this chain or another chain.

Paste icon

pastes the copied commands into this chain, or into another opened chain.

Move up icon

move the highlighted commands up one position in the chain

Move down icon

move the highlighted commands down one position in the chain

Activate icon

activate any of the highlighted commands that are deactivated

Deactivate icon

deactivate all of the highlighted commands

Record icon

record one or more options. For more information on Record, go to 28.3.1.2 Record Icon and the Recorder Toolbar

Button at Bottom

Run button

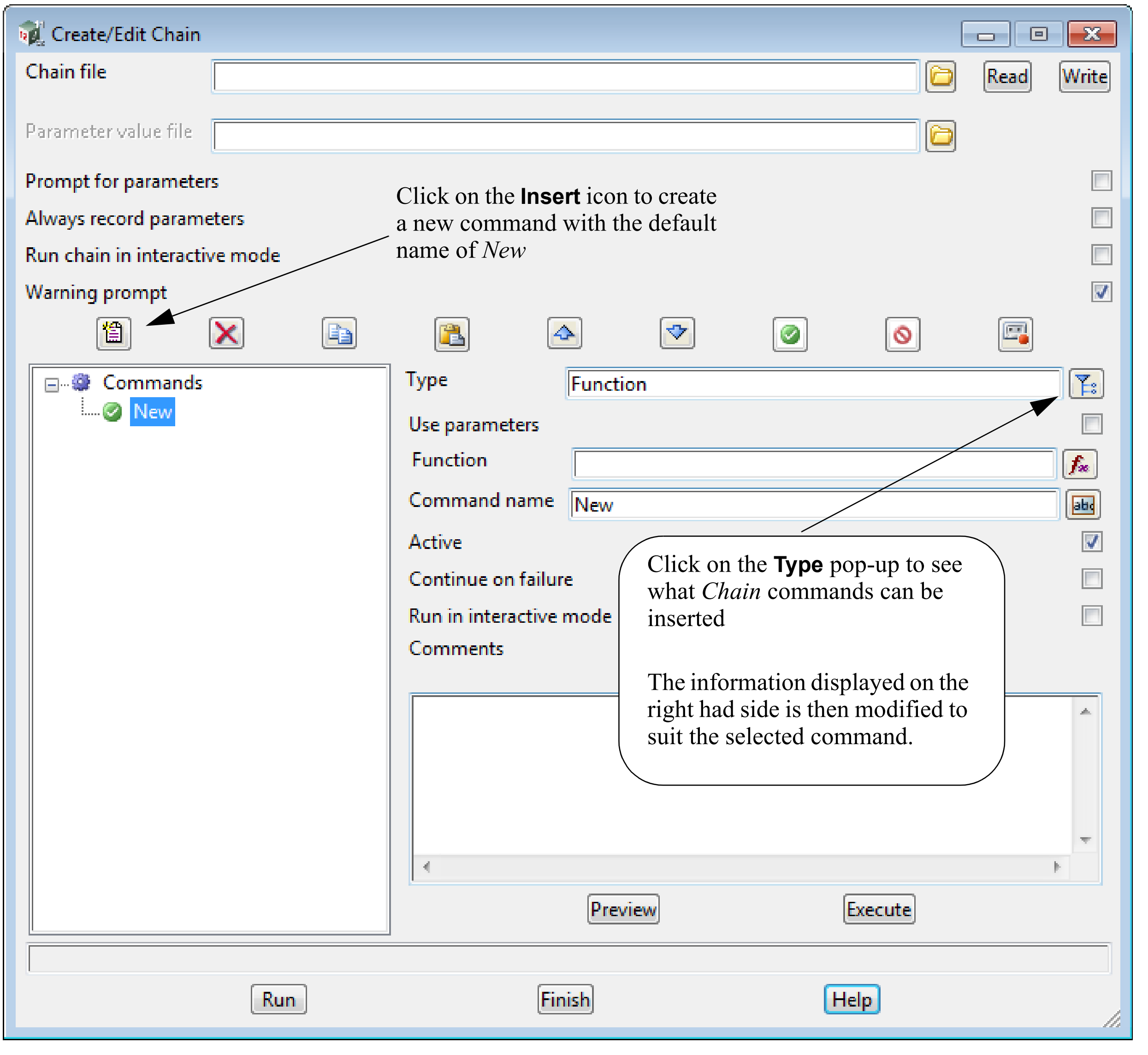
run the chain.

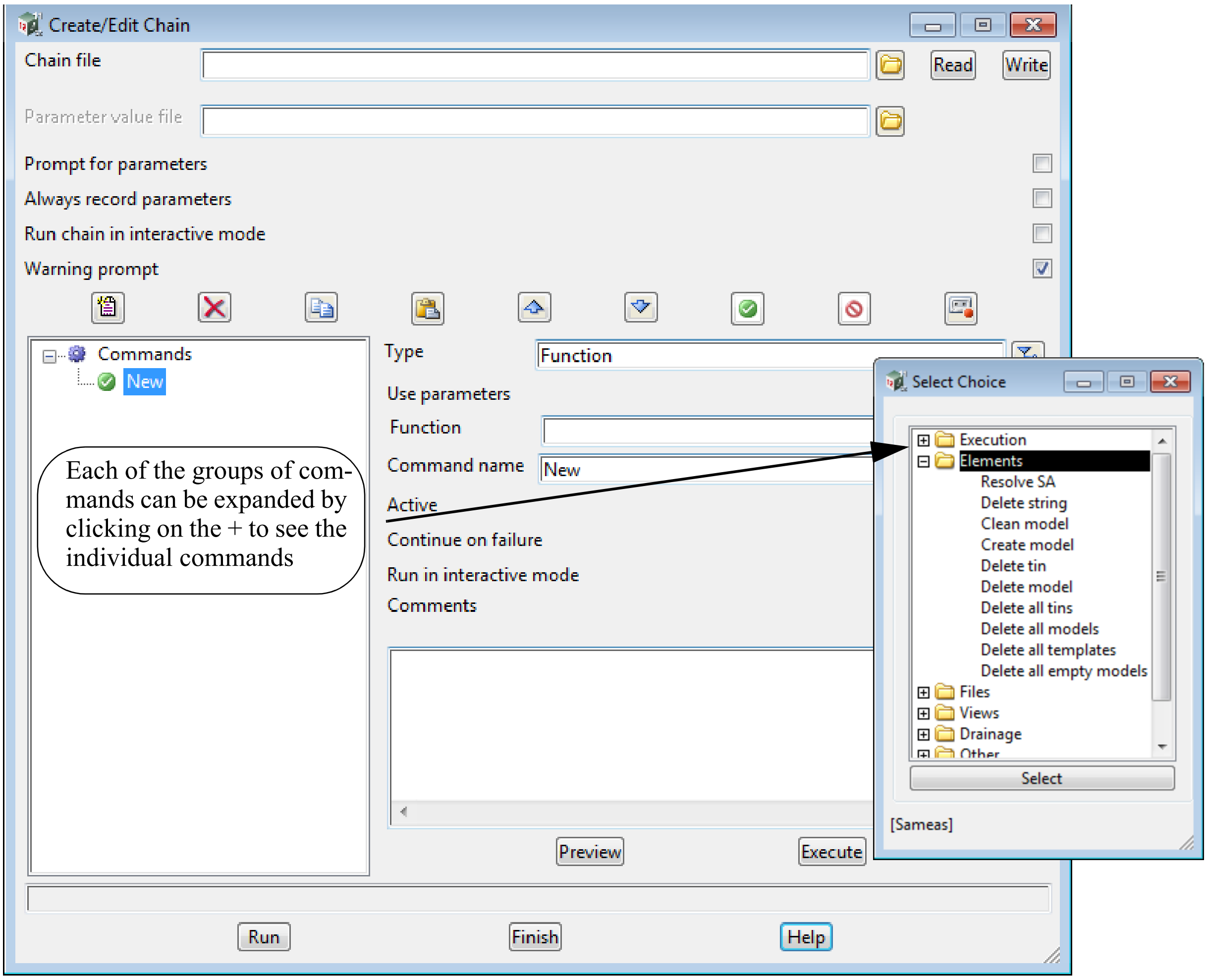
Continue to the next subsection 28.3.1.1 Inserting a Chain Command or the next section 28.3.3 Edit a Chain or return to 28.3 Chains.

28.3.1.1 Inserting a Chain Command

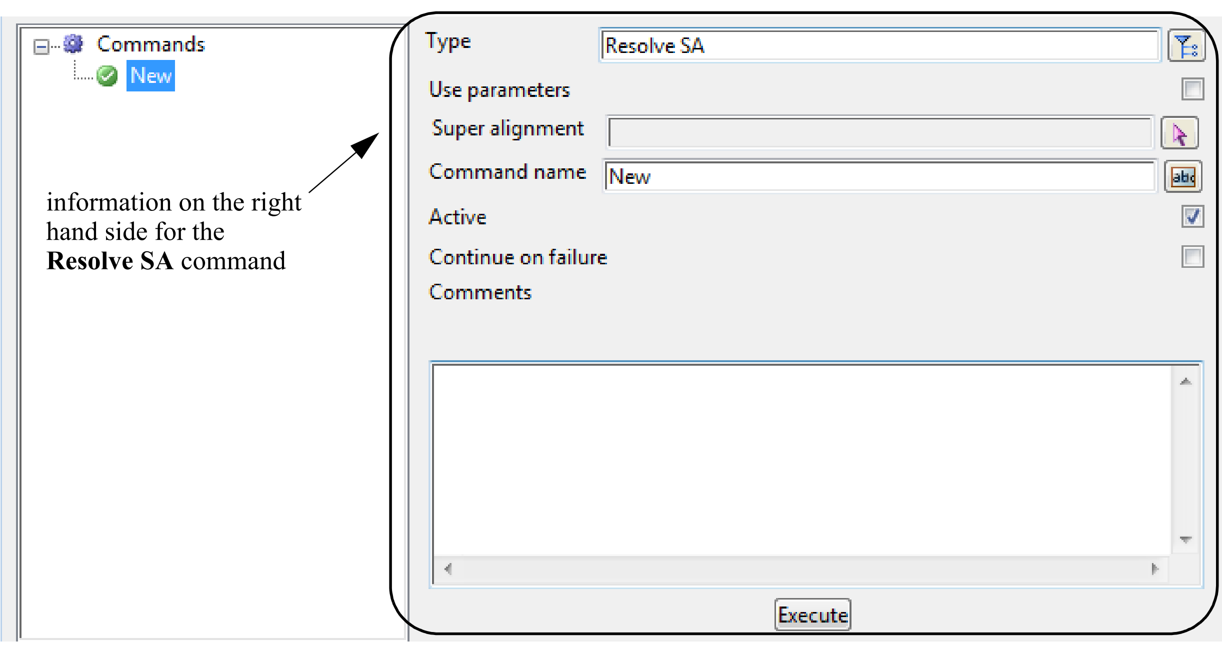
The chain is built up by *inserting commands* using the ***Insert*** icon.

Clicking on the ***Insert*** icon inserts a new command with the default name of **New** into the chain. The ***Type*** of the command is then selected on the right hand side of the panel. The specific information needed for defining that type of new command is then displayed on the right hand side of the panel.





Selecting a command will update the right hand side of the panel with information for that particular command. For example, for the **Resolve SA** command:



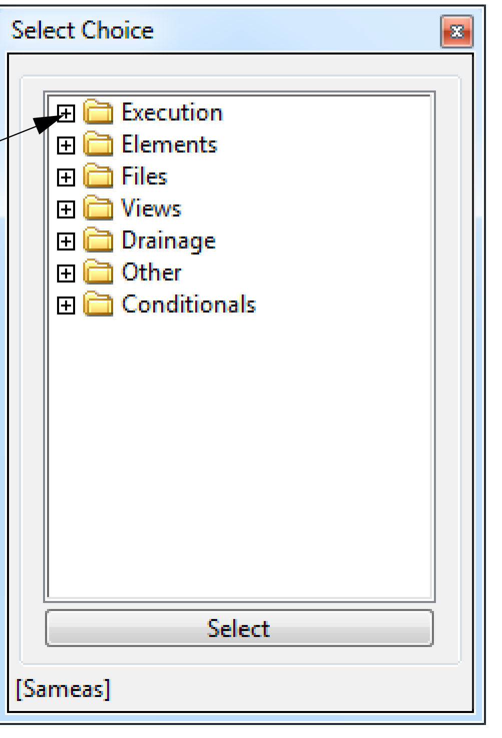
After each Command is added to the chain, or a Command modified, click on **Write** to save the updated chain.

For all the available **Chain Commands** that can be **Inserted**, see 28.3.1.1.1 Chain Commands.

Continue to the next section 28.3.1.1.1 Chain Commands or return to 28.3 Chains.

28.3.1.1.1 Chain Commands

Click on the + to expand the Command group



For the *Chain* commands under the group

Execution, go to	28.3.1.1.1.2 Execution
Elements	28.3.1.1.1.3 Elements
Files	28.3.1.1.1.4 Files
Views	28.3.1.1.1.5 Views
Drainage	28.3.1.1.1.6 Drainage
Other	28.3.1.1.1.7 Other
Conditionals	28.3.1.1.1.8 Conditionals

28.3.1.1.1.1 Common Fields in Commands

The common fields used in the right hand side of the **Create/Edit Chain** panel for the Chain commands are:

Field Description Type Defaults Pop-Up

Use parameters tick box

*If **ticked**, parameters can be used within the commands.*

Note that only some chain commands have a **Use Parameters** *tick.*

Command name text box

the name to give the command - this is what is displayed in the Chain list.

As information is filled in for a command, a default Command name composed of the command and some extra information from the command fields (for example the name of the super alignment) is created and written into the Command name field.

The Command name does not have to be unique and can be changed at any time.

*Clicking on the **abc** icon on the end of the* **Command name** *field will regenerate the automatic Command name and write it in the* **Command name** *field.*

*When you **rerecord** a command, the* **Command name** *is **not** modified but left as it was before rerecording.*

Active tick box

*if **ticked**, the command is run when the chain is run.*

*If **not ticked**, the command is not run when the chain is run.*

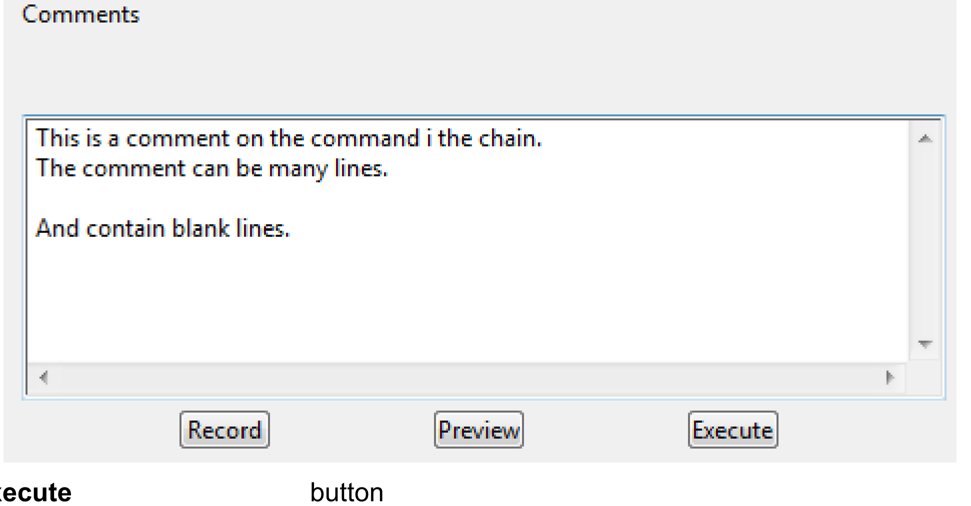
Continue on failure tick box

*if **ticked** and there is an error when running the command, processing continues to the next command in the Chain.*

*If **not ticked** and there is an error when running the command then the Chain terminates.*

Comments text box

any information typed into this box is kept as a Comment for current Chain command being displayed.



Execute button

run the current chain command. This button is not on all commands.

When a chain is running, messages and error messages are written to the ***Output window***.

Important Note - each time a command is inserted or modified, the **Write** button must be clicked to write out the modified chain.

Return to 28.3.1.1.1 Chain Commands or 28.3 Chains.

28.3.1.1.1.2 Execution

For the description of the common fields on most Chain Command panels, see 28.3.1.1.1.1 Common Fields in Commands*. These include the fields* **Use Parameters**, **Command name**, **Active**, **Continue on failure** *and* **Comments**, *and the button* **Execute**.

Option

run a **12d Model** *option. Note that not all options can be run in a chain.*

Option *is also used to record the option that it runs.*

For more information, go to 28.3.1.1.2.1 Execution >Option.

Option - Manual

run a **12d Model** *option in Manual Mode. Note that not all options can be run in a chain.*

Option - Manual *is also used to record the option that it runs.*

For more information, go to 28.3.1.1.2.2 Execution >Option - Manual.

Screen layout file

run an existing screen layout file. Note - not all slf's can be run.

For more information, go to 28.3.1.1.2.3 Execution >Screen Layout File.

Function

recalc a 12d function

For more information, go to 28.3.1.1.2.4 Execution >Function.

Delete function

delete a function.

It will optionally ***clean*** *all the strings created by the function.*

Delete all functions

delete all functions

It will optionally ***clean*** *all the strings created by all the functions.*

Chain

run an existing chain.

Macro

run a **12d Model** *macro that doesn't have a panel*

For more information, go to 28.3.1.1.2.5 Execution >Macro.

Plot

run a plot parameter file.

For more information, go to 28.3.1.1.2.6 Execution >Plot.

Shell command

run an external program, Microsoft command line etc.

For more information, go to 28.3.1.1.2.7 Execution >Shell Command.

Continue to the next section 28.3.1.1.1.3 Elements or return to 28.3.1.1.1 Chain Commands or 28.3.1.1 Inserting a Chain Command.

28.3.1.1.1.3 Elements

For the description of the common fields on most Chain Command panels, see 28.3.1.1.1.1 Common Fields in Commands. *These include the fields* **Use Parameters**, **Command name**, **Active**, **Continue on failure** *and* **Comments**, *and the button* **Execute**.

Resolve SA

resolve a selected super alignment.

Clicking on the Resolve command highlights the selected super alignment.

Delete string

delete a selected string

Clean model

clean the given model

Create model

creates a new model of the specified name

Delete tin

delete the given tin

Delete model

delete the given model

Delete all tins

delete all tins in the project

Delete all models

delete all models in the project

Delete all templates

delete all templates in the project

Delete all empty models

delete all empty models in the project

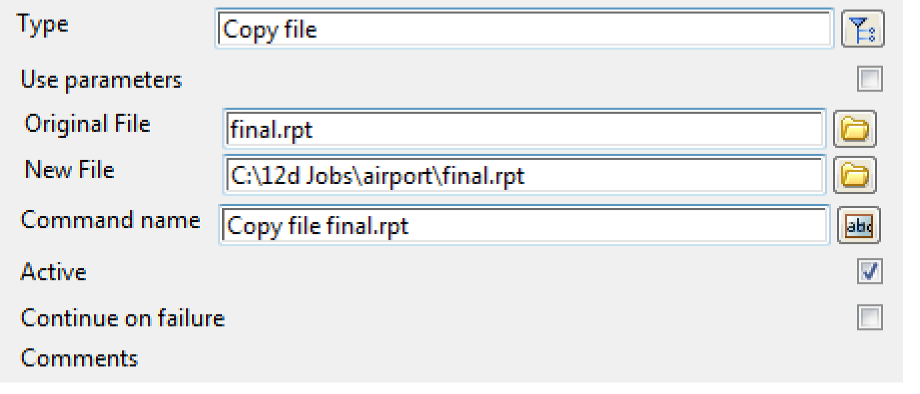
Continue to the next section 28.3.1.1.1.4 Files or return to 28.3.1.1.1 Chain Commands or 28.3.1.1 Inserting a Chain Command.

28.3.1.1.1.4 Files

For the description of the common fields on most Chain Command panels see [28.3.1.1.1.1 Common Fields in Commands]. *These include the fields* **Use Parameters**, **Command name**, **Active**, **Continue on failure** *and* **Comments**, *and the button* **Execute**.

Copy file

copy the given file to a user given file name (which included the full path name).



The file being copied will overwrite an existing file.

Delete file

deletes the specified file

Compare

compare two text files and write the results out to a **Report** *file.*

If the Report file already exists, the **Write mode** *determines if it is overwritten or appended to.*

Continue to the next section [28.3.1.1.1.5 Views] or return to [28.3.1.1.1 Chain Commands] or [28.3.1.1 Inserting a Chain Command].

28.3.1.1.1.5 Views

For the description of the common fields on most Chain Command panels see 28.3.1.1.1.1 Common Fields in Commands*. These include the fields* **Use Parameters***,* **Command name***,* **Active***,* **Continue on failure** *and* **Comments***, and the button* **Execute***.*

Redraw all views

redraws all views.

Redraw a view

redraws a given view.

Fit view

do a fit on a given view.

Minimise view

minimise a given view.

Maximise view

maximise a given view.

Restore a view

restore a minimised view.

Create view

creates a new view of the given name and optionally position in the View Area.

For more information, go to 28.3.1.1.2.8 Views >Create view.

Delete view

delete a view of a given name.

Add model to view

adds the given model to a given view.

Remove model from view

remove a given model from a given view.

Remove all from view

removes all the models from a given view.

Dump view image

dump an image of a given view.

For more information, go to 28.3.1.1.2.9 Views >Dump view image.

Change view colour

change the background colour of a given view.

Continue to the next section 28.3.1.1.1.6 Drainage or return to 28.3.1.1.1 Chain Commands or 28.3.1.1 Inserting a Chain Command.

28.3.1.1.1.6 Drainage

For the description of the common fields on most Chain Command panels see 28.3.1.1.1.1 Common Fields in Commands*. These include the fields* **Use Parameters***,* **Command name***,* **Active***,* **Continue on failure** *and* **Comments***, and the button* **Execute***.*

Drainage analysis

runs drainage analysis on a given model

Drainage analysis- dynamic

runs dynamic drainage analysis on a given model

Set pit details

sets the pit details for a given model

Regrade pipes

regrades the pipes in a given model

Continue to the next section 28.3.1.1.1.7 Other or return to 28.3.1.1.1 Chain Commands or 28.3.1.1 Inserting a Chain Command.

28.3.1.1.1.7 Other

For the description of the common fields on most Chain Command panels see 28.3.1.1.1.1 Common Fields in Commands*. These include the fields* **Use Parameters***,* **Command name***,* **Active***,* **Continue on failure** *and* **Comments***, and the button* **Execute***.*

Comment

creates a cosmetic comment - no action is performed.

Halt

stop running the chain

Prompt

places a Prompt and returns a value to a chain parameter.

For more information, go to 28.3.1.1.2.10 Other >Prompt.

Sleep

puts the chain asleep for a given number of seconds.

Write output window

writes a one line message to the Output Window.

Clear output window

clears the Output Window.

Dump slx image

writes out the image of the panel than an SLX creates.

For more information, go to 28.3.1.1.2.11 Other >Dump slx image.

Print parameter

prints out information about a chain parameter to the Output Window.

For more information, go to 28.3.1.1.2.12 Other >Print parameter.

Save project

save the current project to disk.

Continue to the next section 28.3.1.1.1.8 Conditionals or return to 28.3.1.1.1 Chain Commands or 28.3.1.1 Inserting a Chain Command.

28.3.1.1.1.8 Conditionals

How the Conditionals work is described in 28.3.1.1.1.9 Chain Flow Control and Conditional Commands.

Label

create a labelled line in the chain. See 28.3.1.1.1.9 Chain Flow Control and Conditional Commands.

Goto

jump to a labelled line. See 28.3.1.1.1.9 Chain Flow Control and Conditional Commands.

If Model Is Empty

test if a model of a given name is empty.

If Model Exists

test if a model of a given name exists

If Tin Exists

test if a tin of a given name exists

If Function Exists

test if a function of a given name exists

If File Exists

test if a file of a given name exists

If View Exists

test if a view of a given name exists

If Parameter Equals

test if a given parameter has a certain value

For more information, go to 28.3.1.1.2.13 Conditionals >If Parameter Equals.

Return to 28.3.1.1.1 Chain Commands or 28.3.1.1 Inserting a Chain Command.

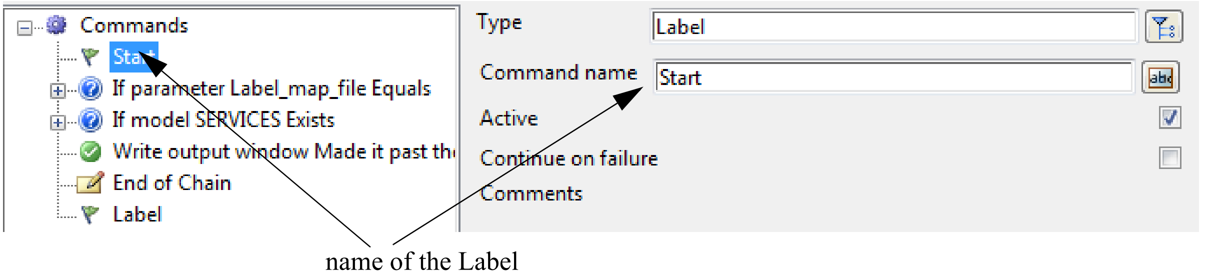
28.3.1.1.1.9 Chain Flow Control and Conditional Commands

The commands in a Chain are normally processed in the order that they are listed in the Chain.

However this order can be changed by using the **Goto** and **Label** commands, and the commands given in the **Conditionals** group (see 28.3.1.1.1.8 Conditionals).

Label

The **Label** command has a user given name and acts as a reference line in a chain. Goto's and Conditionals can change the processing order of the Chain so that the next command processed is the one after the ***Label*** and processing continues on from there.

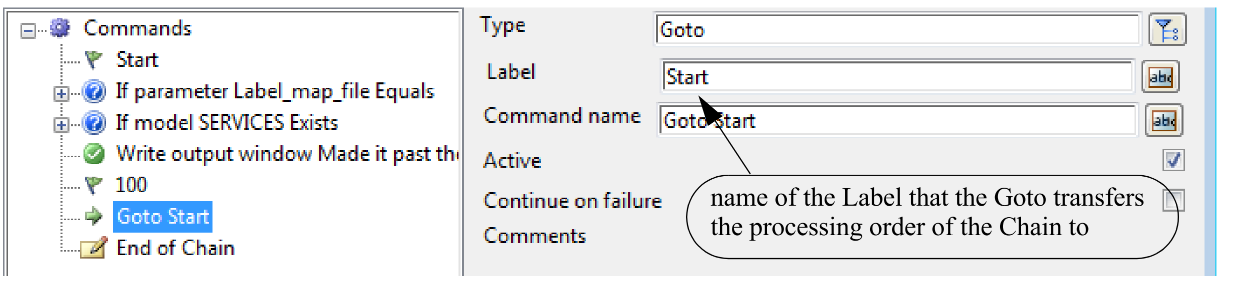


The name of a *Label* must be unique within the Chain.

When run the *Label* command doesn't do anything and processing moves onto the command after the *Label* command.

Goto

The **Goto** command changes the processing order of the Chain so that next command processed is the one after the given ***Label***.



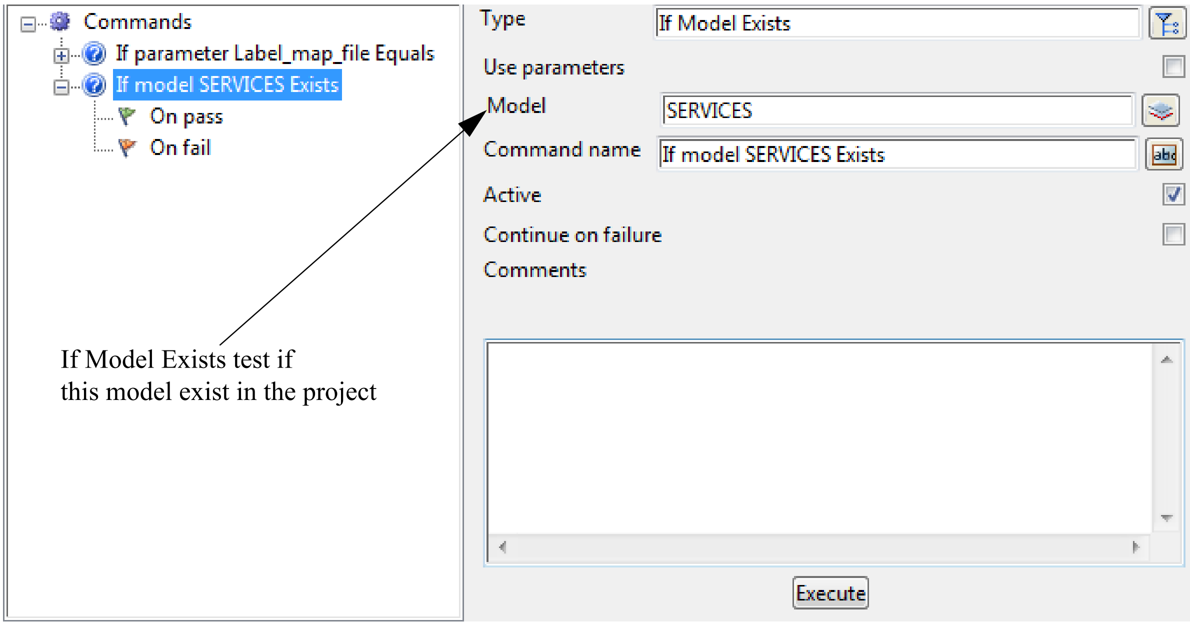
Conditionals

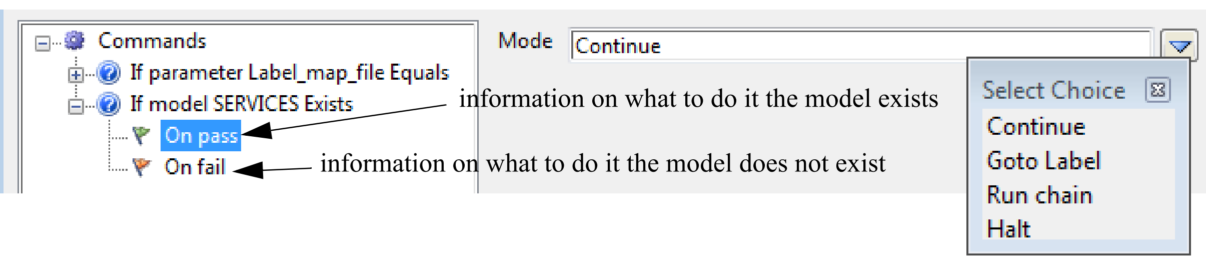
A **Conditional** consists of a test and the two nodes **On pass** and **On fail.**

Each **Conditional** commands performs a specific test and

(a) if the test is passed then the action on the **On pass** line is performed

(b) if the tests fails then the action on the **On fail** line is performed

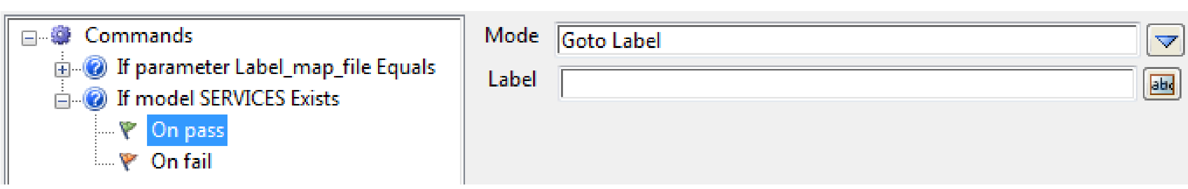




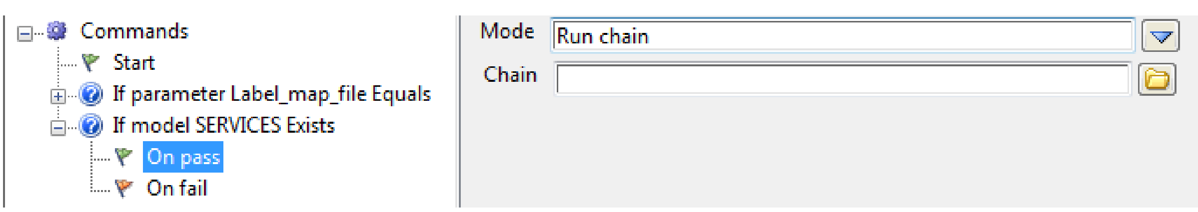
The choices for both the **On pass** and **On fail** are the same and are:

Continue *- run the next command in the Chain*

Goto Label *- the next command to run is the one after the given Label.*



Run chain *- run the given Chain. The next command in the Chain is then run.*



Halt *- stop running the chain.*

Return to 28.3.1.1.1 Chain Commands or 28.3.1.1 Inserting a Chain Command.

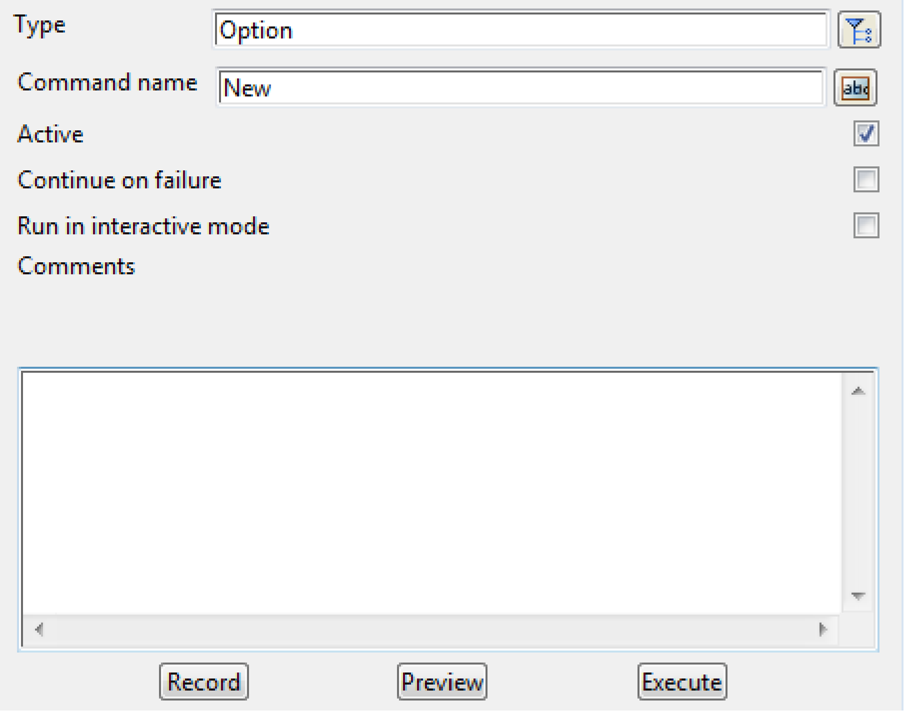
28.3.1.1.2 Additional Information on Some Chain Commands

See

- 28.3.1.1.2.1 Execution >Option
- 28.3.1.1.2.2 Execution >Option - Manual
- 28.3.1.1.2.3 Execution >Screen Layout File
- 28.3.1.1.2.4 Execution >Function
- 28.3.1.1.2.5 Execution >Macro
- 28.3.1.1.2.6 Execution >Plot
- 28.3.1.1.2.7 Execution >Shell Command
- 28.3.1.1.2.8 Views >Create view
- 28.3.1.1.2.9 Views >Dump view image
- 28.3.1.1.2.10 Other >Prompt
- 28.3.1.1.2.11 Other >Dump slx image
- 28.3.1.1.2.13 Conditionals >If Parameter Equals

28.3.1.1.2.1 Execution >Option

The **Option** command records any ***12d Model*** panel or **12d** PL's (macros) that can be recorded.



After this command is selected, the **Record** button must then be pressed to start a "one option" record cycle. The option to be recorded is then started up and run so that it is recorded.

The fields and buttons have the following functions.

Field Description	Type	Defaults	Pop-Up
Type	chain command box		

*the chain command - this command is **Option***

Run in interactive mode	tick box		

*if **ticked**, when the command runs, the panel will be displayed.*

At this point all the data in the panel field may be changed. Once the panel is closed, the chain will continue.

Note that it is the responsibility of the user to only make sensible changes.

*If **not ticked**, the command will run without being displayed and allowing for changes to be made.*

Important Note and Warning

*The **Run in interactive mode** for a **panel** has no connection to the* **Run chain in interactive mode** *for the Chain.*

For the description of the fields **Use Parameters**, **Command name**, **Active**, **Continue on failure** *and* **Comments**, *and the button* **Execute** *see* 28.3.1.1.1.1 Common Fields in Commands.

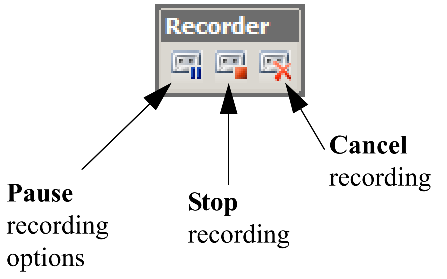
Buttons at bottom

See Record button
Preview button
Execute button

Record button

Clicking on the **Record** *button starts a "one option" record cycle.*

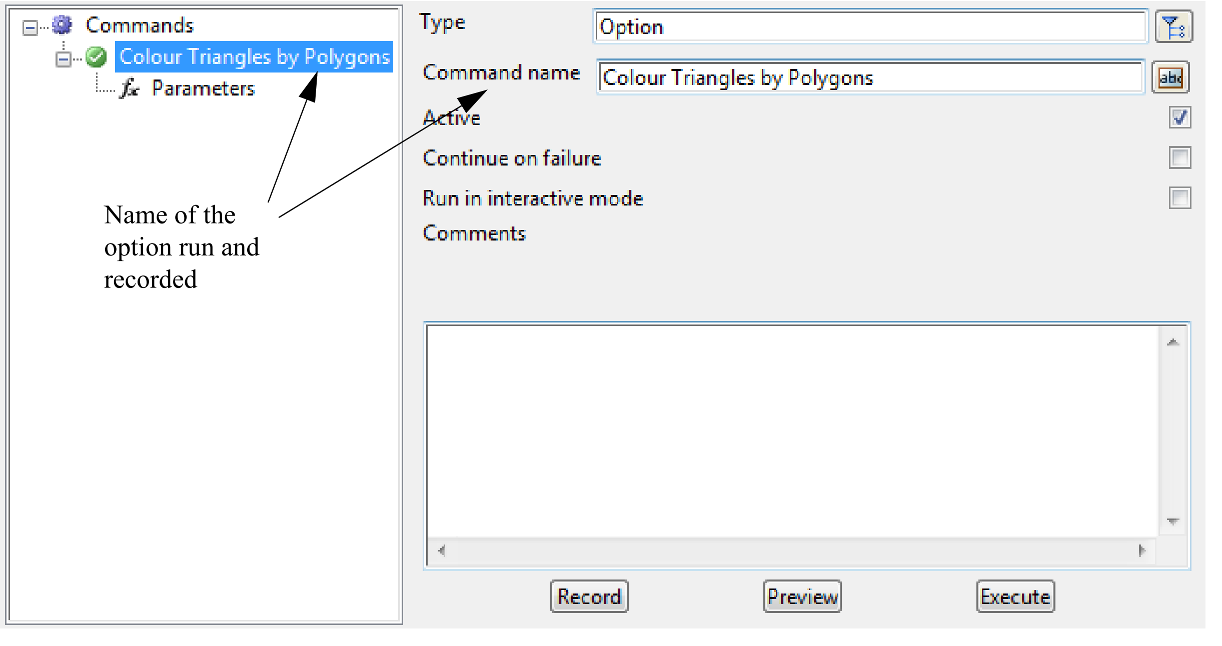
The **Recorder** *toolbar is placed on the screen but no button has to be pressed because the Record is already running. However the buttons* **Stop** *and* **Cancel** *can be pressed to terminate the recording.*



Now ***run*** *the* ***12d Model*** *panel option or macro that you wish to record.*

Once the option has run, the **Recorder** *panel is automatically removed, and the recorded option's details recorded in the* **Option** *command and the* **Command name** *of the* **Option** *is set to the name of the option that was run.*

The **Create/Edit Chain** *panel is then opened and displayed the* **Option** *command that was just recorded.*



Click on the **Write** *button at the bottom of the* **Create/Edit Chain** *panel to save the modified chain.*

Important Notes on Recording

1. CAD commands ***can not*** *be recorded.*

2. Many Macros ***can not*** *be recorded. But if* **Option** *won't work, then* **Option - Manual** *may. See* 28.3.1.1.2.2 Execution >Option - Manual.

3. Not all 12d panels can be recorded. For example, if a screen layout file can not be created for an option, then it can't be recorded. When recording, the ***12d Model*** *panel option must be filled in and run. This may restrict what options can be recorded. But if* **Option** *won't work,* **Option - Manual** *may. See* 28.3.1.1.2.2 Execution >Option - Manual. *If you still find a panel won't record, please leave a note about that panel on the 12d Model Forum.*

Preview button

the **Preview** *button brings up the panel with all the information filled in for those panel fields* ***that can be validated****. So sometimes* **Preview** *can not display all the command information.*

Preview *is for display only and on its own can't be used to modify the details of the command.*

However, if it is a ***recorded option*** *and there is a* **Record** *button on the command, it can be done in conjunction with the* **Record** *button.*

So to change the values in a ***recorded*** *option, select* **Preview** *to bring up the recorded option with all the information filled in.*

Then hit the **Record** *button on the* **Option** *command, make the required changes to the displayed option, and run the panel again.*

This will record the new information ***in place of the existing information****.*

Click on the **Write** *button at the bottom of the* **Create/Edit Chain** *panel to save the modified chain.*

Execute button

run the current chain command.

Option Parameters

To define or modify parameters used in an option, click on the **+** beside the option name in the chain (or double click on the option name) and the Parameters node for the option will be displayed.

For more details on option parameters, go to 28.3.4.3 How to Create Chain Parameters.

Special Notes on Recording Options

1. Use Unique String Names for String Selects

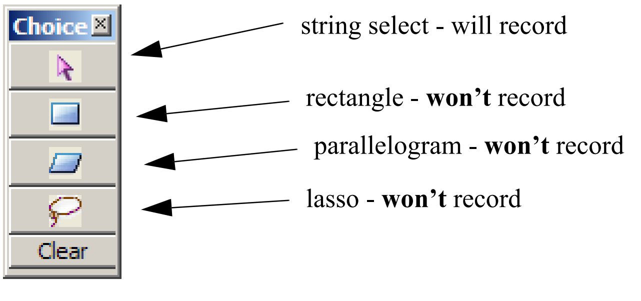
String selects on panels record the ID and the name of the string, and the model ID and the name of the model containing the string.

When the option is run in the chain, the string is first searched for using the model and string id's but if that fails then a string of the recorded name is searched for in the recorded model name.

If the string name is unique in the model then the correct string will be selected. If there is more than one string with the same name in the model, then the first string of that name that is found will be used.

2. Polygon Selects

String selects can be recorded for a *Polygon box* but ***Rectangle***, ***Parallelogram*** **or** ***Lasso*** can **not** be recorded.



Continue to the next section 28.3.1.1.2.2 Execution >Option - Manual or return to 28.3.1.1.1 Chain Commands or 28.3.1.1 Inserting a Chain Command.

28.3.1.1.2.2 Execution >Option - Manual

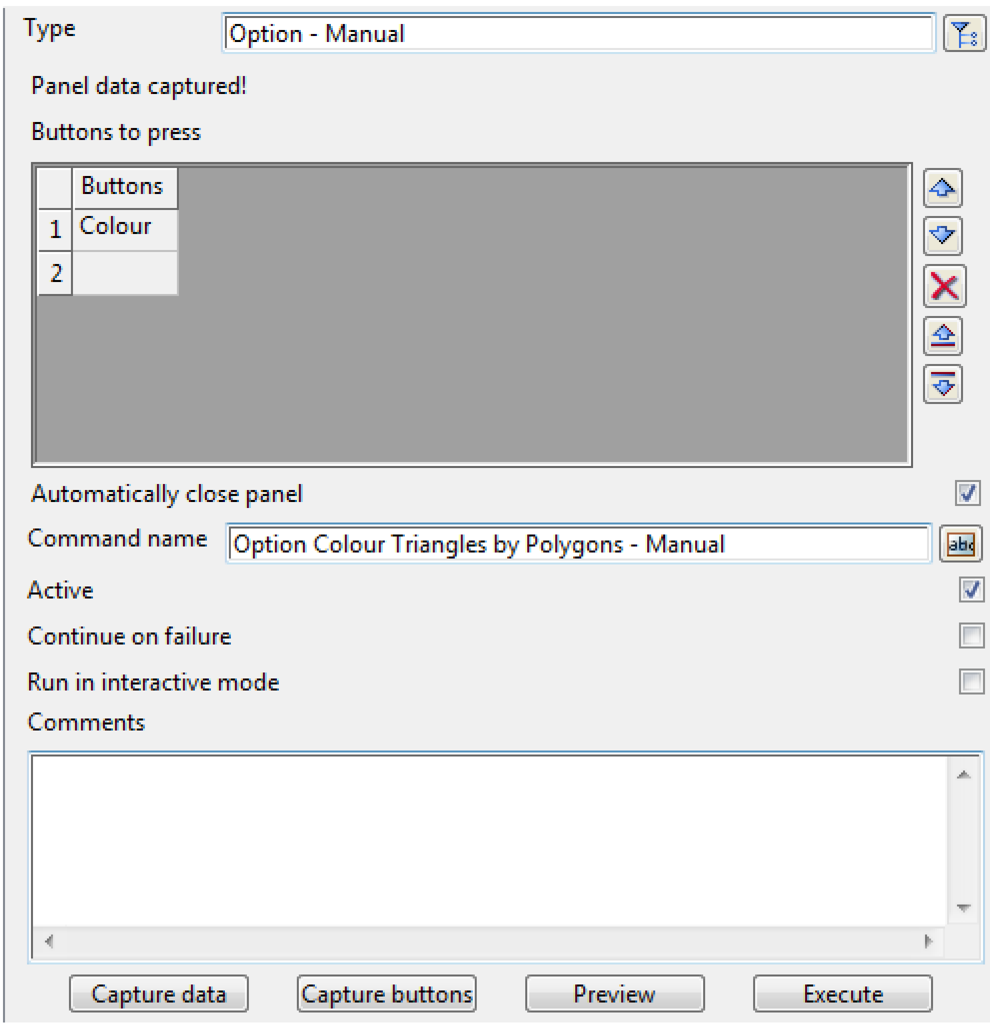
The **Option - Manual** command records an option but with additional information that is used when the command is run in the chain. For example, recording that certain buttons are to be pressed in a certain order.

This is useful for more complex panels, or panels that **require multiple button presses**. It is also useful for **recording macros**.

Unlike the **Option** command, the panel must already be on the screen with the appropriate fields filled in **before** it can be recorded.

The **Option - Manual** command records an option in two parts:

1. The panel to use (the data)
2. The buttons to press and the order they should be pressed



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Type	chain command box		

the chain command - this command is ***Option - Manual***

Buttons to press	grid		

displays the list of buttons in the panel that are to be pressed and in the order that they are to be pressed. The buttons are added to the list by using the **Capture button** *button.*

For information about grids, see 4.19.6 Grids in Panels.

Automatically close panel tick box

*if **ticked**, when the panel is run in the chain, the panel is automatically closed after all the captured buttons are pressed.*

*if **not ticked**, the panel is not automatically closed. If a **Finish** button has not been captured and hence not pressed when the panel is run in the chain, the panel will remain on the screen (open) after the command is finished.*

Run in interactive mode tick box

*if **ticked**, when the command runs, the panel will be displayed.*

At this point all the data in the panel field may be changed. Once the panel is closed, the chain will continue.

Note that it is the responsibility of the user to only make sensible changes.

*If **not ticked**, the command will run without being displayed and allowing for changes to be made.*

Important Note and Warning

*The **Run in interactive mode** for a **panel** has no connection to the* **Run chain in interactive mode** *for the Chain.*

For the description of the fields **Use Parameters**, **Command name**, **Active**, **Continue on failure** *and* **Comments**, *and the button* **Execute** *see* 28.3.1.1.1.1 Common Fields in Commands.

Capture data button

after pressing **Capture data**, *the cursor is moved over the panel that is to be captured. The panel will be highlighted with a yellow border to indicate which panel will be selected.*

Clicking LB will select the highlighted panel and the panel and all the panel field values will be recorded.

Unlike the **Option** *command, the panel must already be on the screen with the appropriate fields filled in **before** pressing* **Capture data**.

Also unlike the **Option** *command the panel is not run.*

*Instead after the panel has been selected, the panel **buttons** that need to be pressed for running the panel are recorded in the order that they need to be pressed.*

Capture buttons button

after pressing **Capture buttons**, *the cursor is moved over the panel buttons on the selected panel, and the panel button will be highlighted with a yellow border to indicate which button will be selected.*

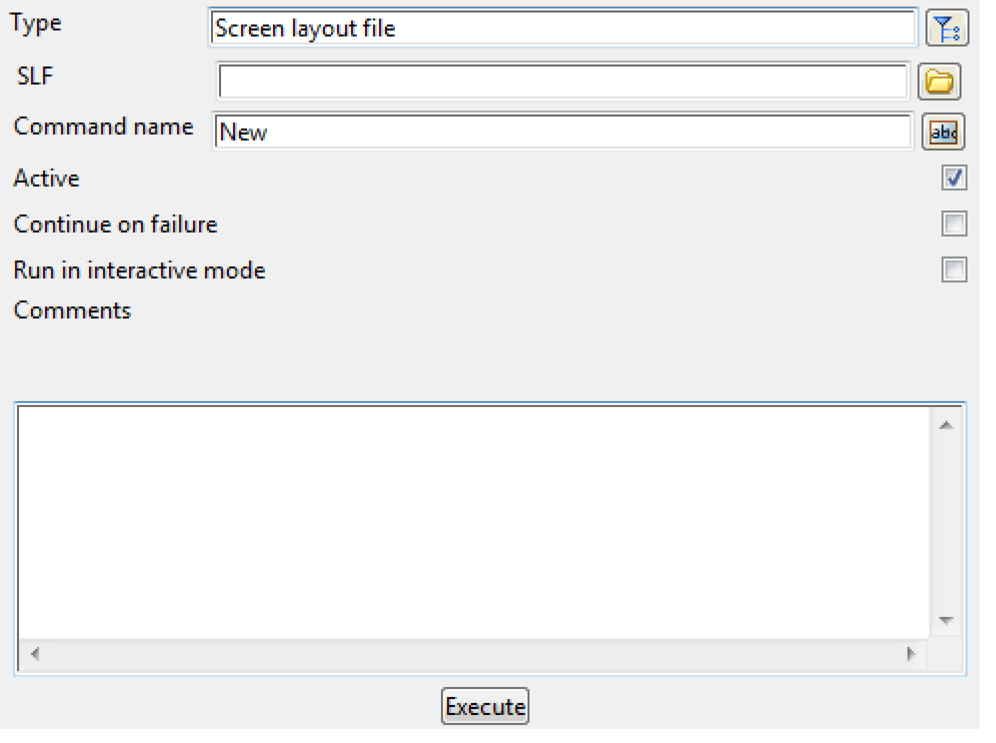
Clicking LB will select the highlighted button and the name of the button will be added to the bottom of the **Buttons to press** *grid.*

To select another panel button, click on **Capture buttons** *again.*

Continue to the next section 28.3.1.1.2.3 Execution >Screen Layout File or return to 28.3.1.1.1 Chain Commands or 28.3.1.1 Inserting a Chain Command.

28.3.1.1.2.3 Execution >Screen Layout File

The **Screen layout file** command displays the first panel stored in an SLF file and runs it by pressing the panels designated '**run button**'.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Type	chain command box		

*the chain command - this command is **Screen layout file.***

SLF	file		

the slf file to run the first panel from.

Run in interactive mode	tick box		

*if **ticked**, when the command runs, the panel will be displayed.*

At this point all the data in the panel field may be changed. Once the panel is closed, the chain will continue.

Note that it is the responsibility of the user to only make sensible changes.

*If **not ticked**, the command will run without being displayed and allowing for changes to be made.*

Important Note and Warning

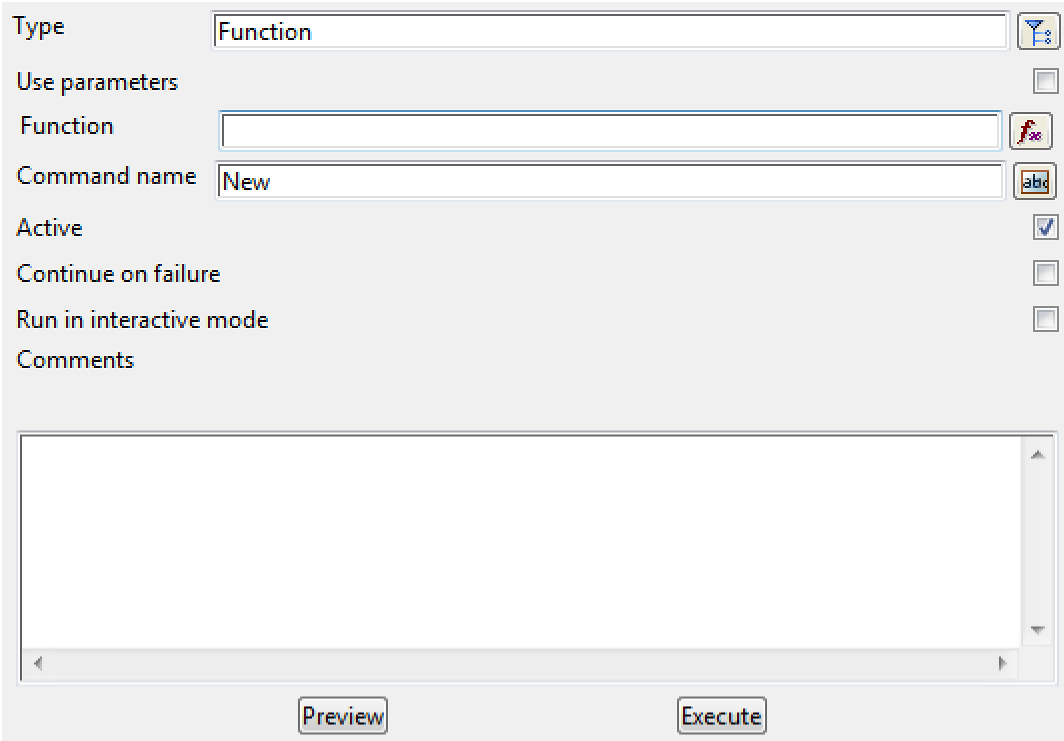
*The **Run in interactive mode** for a **panel** has no connection to the* **Run chain in interactive mode** *for the Chain.*

For the description of the fields **Use Parameters**, **Command name**, **Active**, **Continue on failure** *and* **Comments**, *and the button* **Execute** *see* 28.3.1.1.1.1 Common Fields in Commands.

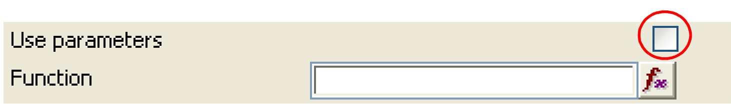
Continue to the next section 28.3.1.1.2.4 Execution >Function or return to 28.3.1.1.1 Chain Commands or 28.3.1.1 Inserting a Chain Command.

28.3.1.1.2.4 Execution >Function

The **Function** command recalcs an existing function.



If **Use parameters** is ***not ticked***, the **Function** field is displayed. This requires the selection of an existing function to recalc.



If **Use parameters** is ***ticked***, the **Function parameter** field is displayed. This requires the selection of a function parameter. See 28.3.4 Chain Parameters for more details.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Type	chain command box		

*the chain command - this command is **Function.***

Field Description	Type	Defaults	Pop-Up
Function	function box		available functions

the function to recalc.

Field Description	Type	Defaults	Pop-Up
Run in interactive mode	tick box		

*if **ticked**, the function will not be recalced. Instead, the editor for the function will be displayed (if one is available).*

At this point all the data in the panel fields may be changed. Once the panel is closed, the chain will continue.

Note that it is the responsibility of the user to only make sensible changes.

*If **not ticked**, the command will run without being displayed and allowing for changes to be made.*

Important Note and Warning

*The **Run in interactive mode** for a **panel** has no connection to the* **Run chain in interactive mode** *for the Chain.*

Preview button

the **Preview** *button brings up the panel with all the information filled in for those panel fields in the function **that can be validated**. So sometimes* **Preview** *can not display all the command information.*

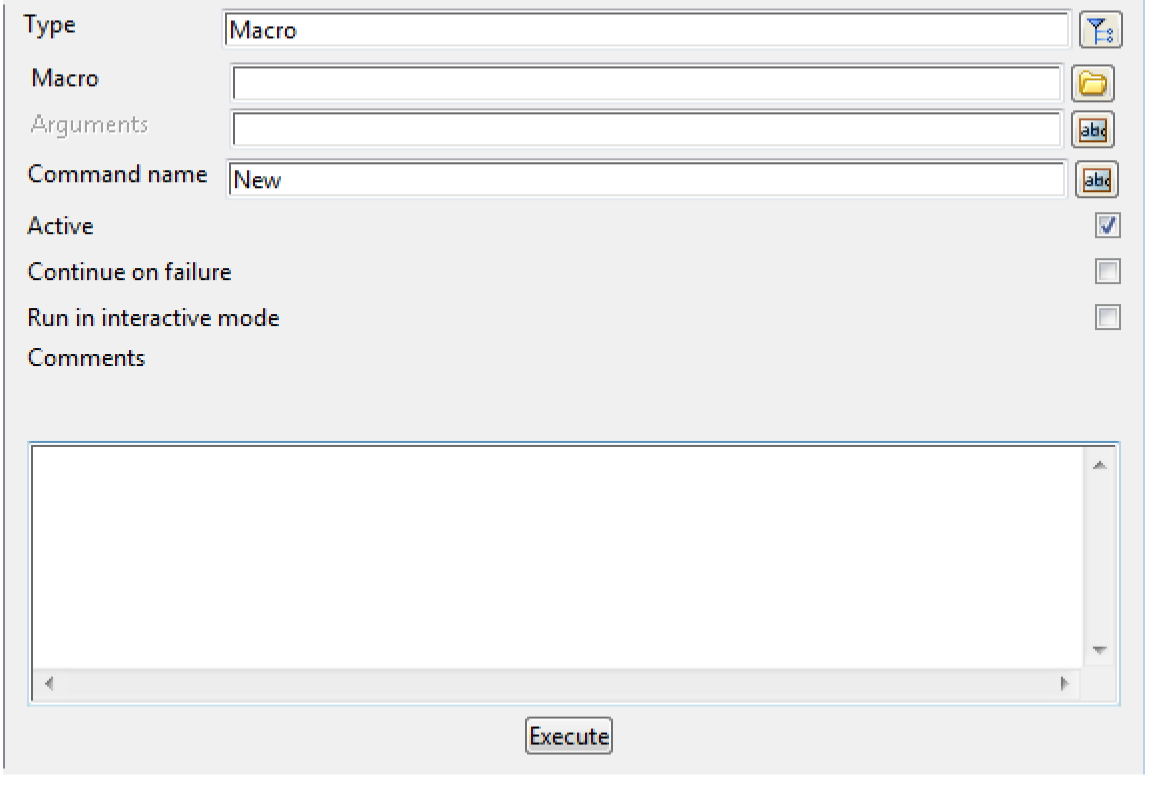
Preview *is for display only and on its own can't be used to modify the details of the command.*

For the description of the fields **Use Parameters***,* **Command name***,* **Active***,* **Continue on failure** *and* **Comments***, and the buttons* **Execute** *see* 28.3.1.1.1.1 Common Fields in Commands.

Continue to the next section 28.3.1.1.2.5 Execution >Macro or return to 28.3.1.1.1 Chain Commands or 28.3.1.1 Inserting a Chain Command.

28.3.1.1.2.5 Execution >Macro

The **Macro** command runs a macro with a set of optional arguments



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Type	chain command box		

*the chain command - this command is **Macro**.*

Macro	file box		*.4do files

the macro to run.

Arguments	input		

an optional set of arguments to pass to the macro.

***Note** that Chain Parameters can be passed as argument to the Macro. See* 28.3.5.5 Chain Parameters as Macro (12dPL's) Arguments

Run in interactive mode	tick box		

if the macro creates a panel, it must be run in interactive mode.

*If **ticked**, if it is a panel macro (i.e. brings up a panel), the macro will be run in Interactive mode and the panel will be displayed. If it is not a panel macro, the macro will run as normal.*

At this point all the data in the panel fields may be changed. Once the panel is closed, the chain will continue.

Note that it is the responsibility of the user to only make sensible changes.

*If **not ticked**, the command will run without being displayed and allowing for changes to be made.*

Important Note and Warning

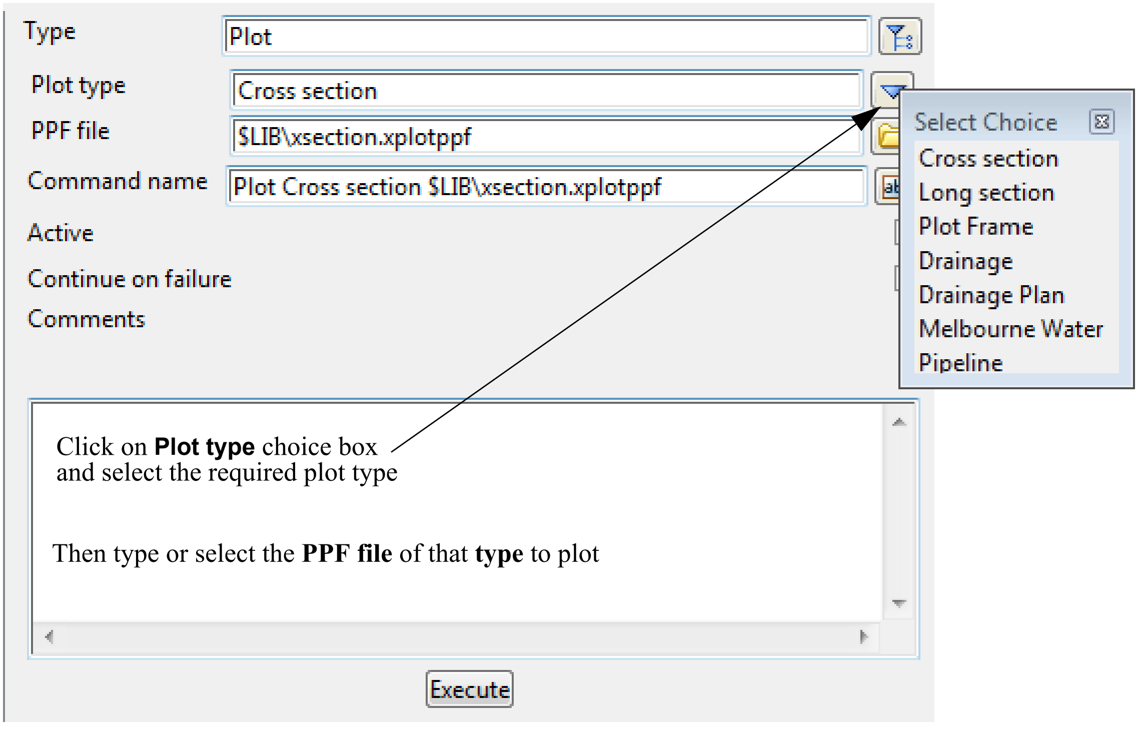
*The **Run in interactive mode** for a **panel** has no connection to the* **Run chain in interactive mode** *for the Chain.*

For the description of the fields **Use Parameters***,* **Command name***,* **Active***,* **Continue on failure** *and* **Comments***, and the button* **Execute** *see* 28.3.1.1.1.1 Common Fields in Commands*.*

Continue to the next section 28.3.1.1.2.6 Execution >Plot or return to 28.3.1.1.1 Chain Commands or 28.3.1.1 Inserting a Chain Command.

28.3.1.1.2.6 Execution >Plot

The **Plot** command plots a given ppf file.

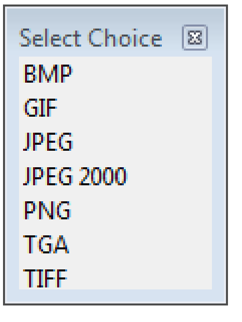


The fields and buttons have the following functions.

Field Description	Type	Defaults	Pop-Up
Type	chain command box		

*the chain command - this command is **Plot***

Plot type	Plot type box		



type of plot to create.

Plot type	plot type box		

name of the view to create. If the name already exists then it is an error.

PPF file	PPF file box		available PPF's for the Plot type

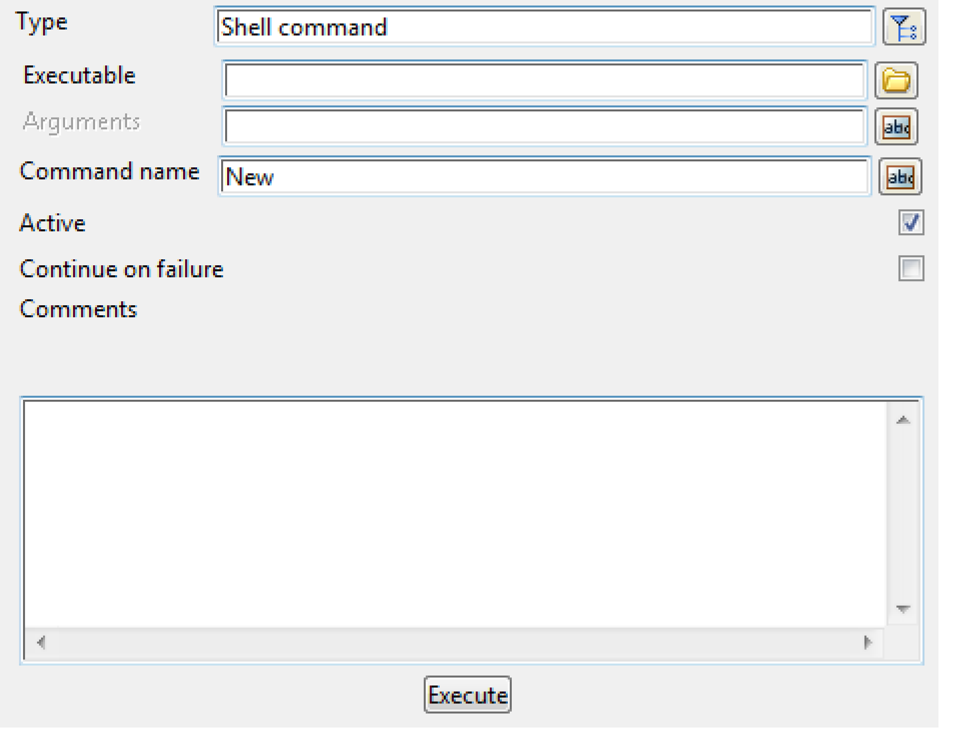
the PPF to use to create the plot.

For the description of the fields **Use Parameters**, **Command name**, **Active**, **Continue on failure** *and* **Comments**, *and the button* **Execute** *see* 28.3.1.1.1.1 Common Fields in Commands.

Continue to the next section 28.3.1.1.2.7 Execution >Shell Command or return to 28.3.1.1.1 Chain Commands or 28.3.1.1 Inserting a Chain Command.

28.3.1.1.2.7 Execution >Shell Command

The **Shell command** commands run an external program, Microsoft command line *etc*.



The fields and buttons have the following functions.

Field Description	Type	Defaults	Pop-Up
Type	chain command box		

the chain command - this command is ***Shell command***

Executable	file browse box		

name of the file to execute.

Arguments	text box		

if ***not blank****, the text is passed as arguments to the* **Executable**.

Also the value of a Chain parameter can be used as an argument passed to the **Executable** *program.*

*The parameter name is enclosed in square brackets (**[** and **]**) and then surrounded by double quotes (**"**)*

"[parameter_name]"

and this is entered in the **Arguments** *field in the place of the argument of the* **Executable** *that it replaces.*

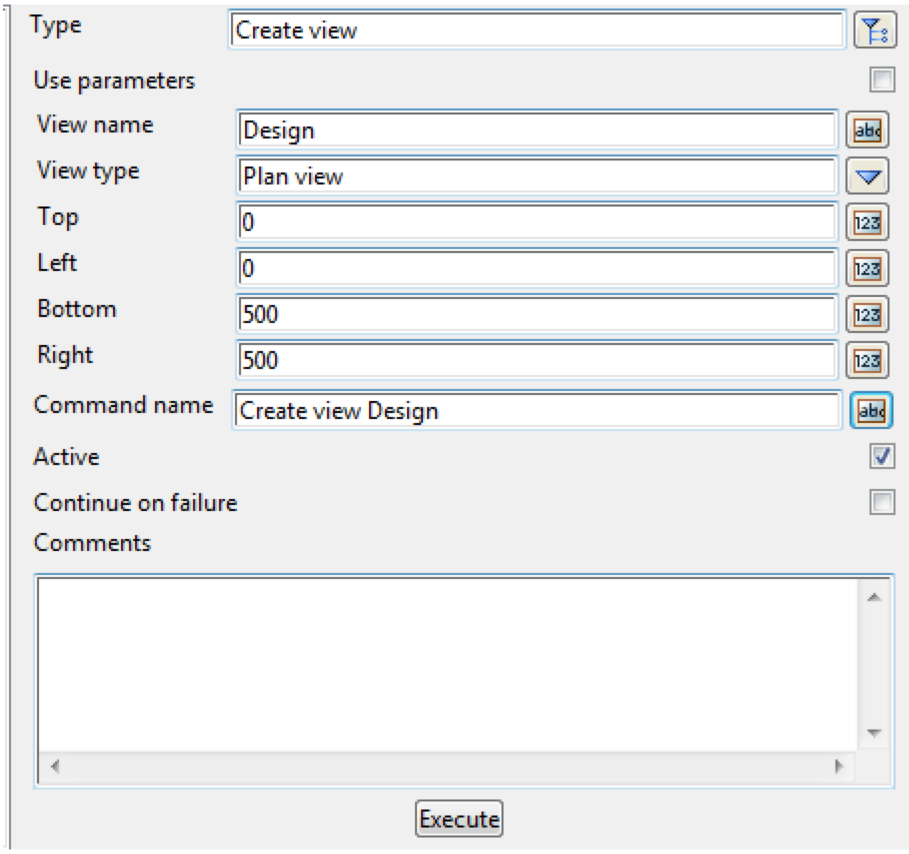
For the description of the fields **Use Parameters**, **Command name**, **Active**, **Continue on failure** *and* **Comments**, *and the button* **Execute** *see* 28.3.1.1.1.1 Common Fields in Commands.

Continue to the next section 28.3.1.1.2.8 Views >Create view or return to 28.3.1.1.1 Chain Commands or 28.3.1.1 Inserting a Chain Command.

28.3.1.1.2.8 Views >Create view

The **Create view** command creates view of a given name and type.

It can also optionally give the Top Left and Bottom Right to position and size the view.



The fields and buttons have the following functions.

Field Description	Type	Defaults	Pop-Up
Type	chain command box		

*the chain command - this command is **Create view***

View name	text box		

name of the view to create. If the name already exists then it is an error.

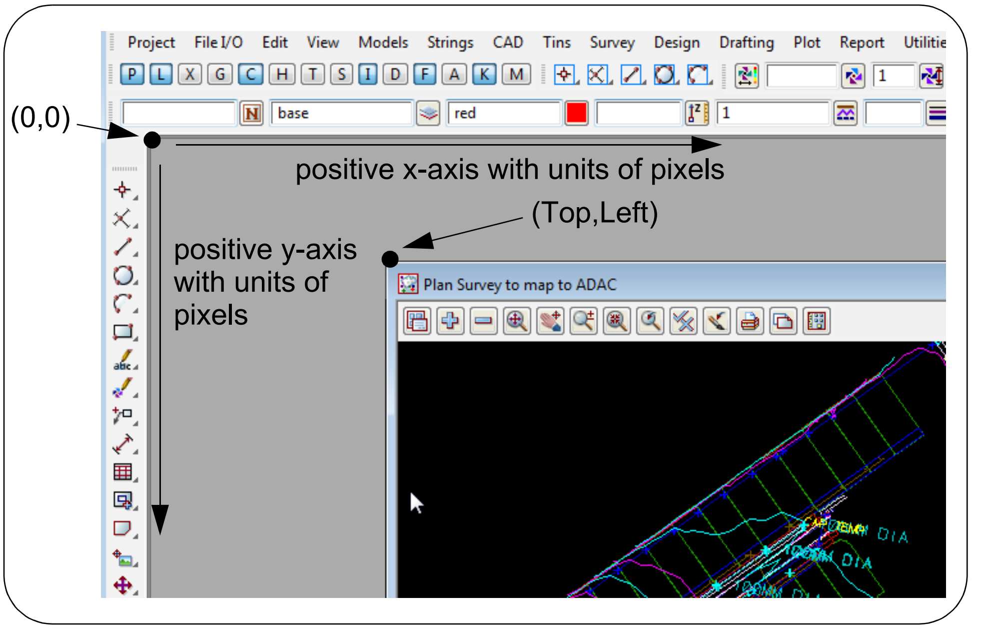
View type	view types box		available view types

the type of view to create

Top/Left/Bottom/Right	integer box		

if not blank, the value in pixels of the top left hand and bottom right corners of the of the created view.

*The coordinate system has (0,0) at the Top Left of the **12d Model** View area, and the X-axis going across the screen and the Y-axis going down the screen. The units are pixels.*

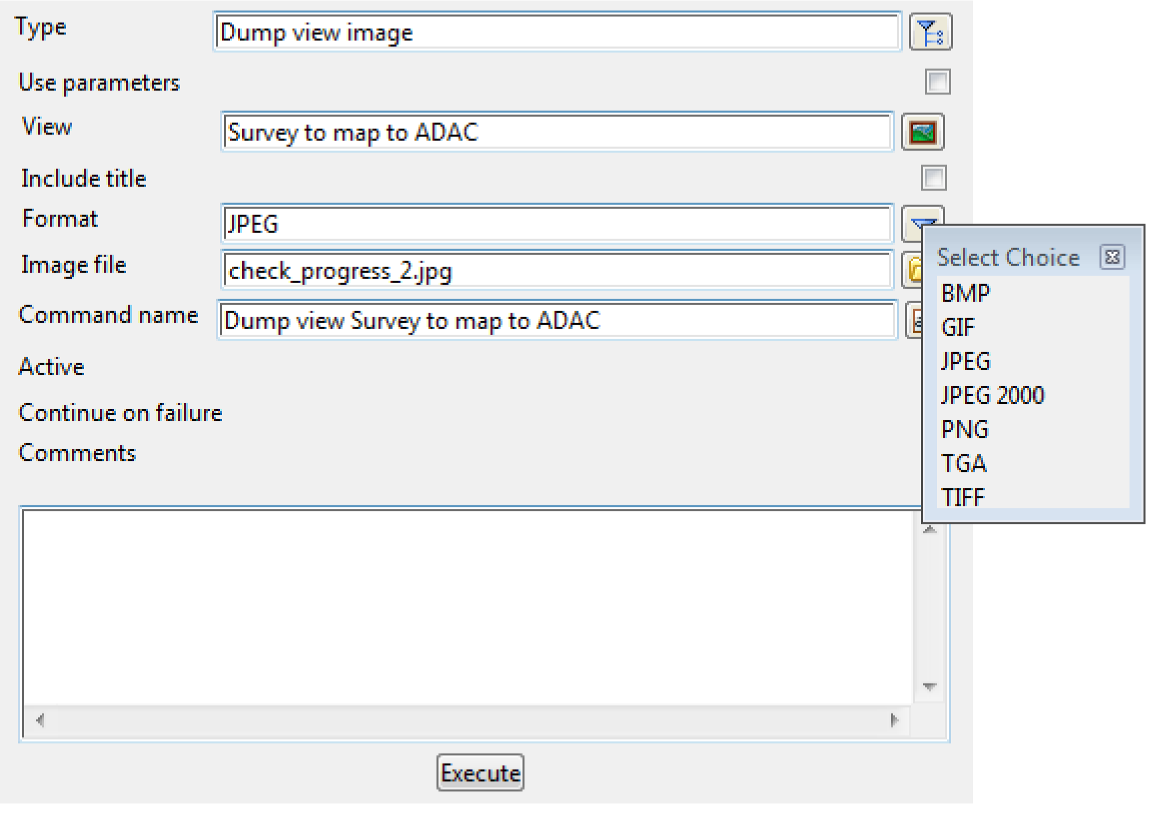


For the description of the fields **Use Parameters**, **Command name**, **Active**, **Continue on failure** *and* **Comments**, *and the button* **Execute** *see* 28.3.1.1.1.1 Common Fields in Commands.

Continue to the next section 28.3.1.1.2.9 Views >Dump view image or return to 28.3.1.1.1 Chain Commands or 28.3.1.1 Inserting a Chain Command.

28.3.1.1.2.9 Views >Dump view image

The **Dump view image** command write out an image of the given view.



The fields and buttons have the following functions.

Field Description	Type	Defaults	Pop-Up
Type	chain command box		

*the chain command - this command is **Dump view image***

View	View box		available views

name of the view to write out an image of file of.

Include title	tick box		

*if **ticked**, the title of the view and the icons on the view are included in the image.*

*If **not ticked**, the title of the view and the icons on the view are **not** included in the image.*

Format	choice box		

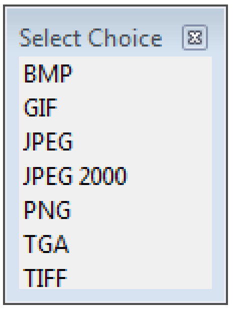


image format to use.

Image file	file box		

the image is written out to this file.

For the description of the fields **Use Parameters***,* **Command name***,* **Active***,* **Continue on failure** *and* **Comments***, and the button* **Execute** *see* 28.3.1.1.1.1 Common Fields in Commands*.*

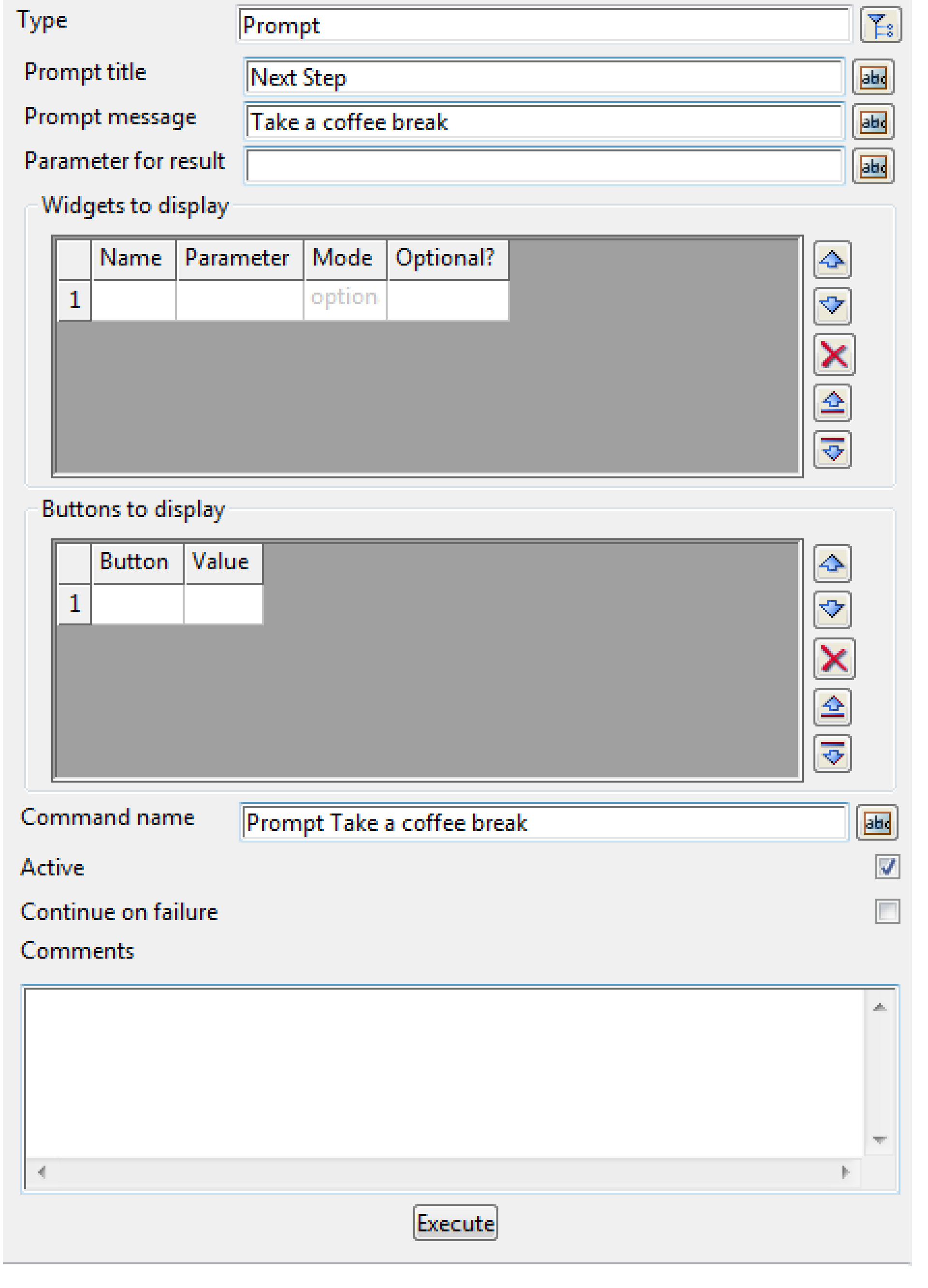
Continue to the next section 28.3.1.1.2.10 Other >Prompt or return to 28.3.1.1.1 Chain Commands or 28.3.1.1 Inserting a Chain Command.

28.3.1.1.2.10 Other >Prompt

The **Prompt** command is used to display a user defined Prompt or panel, with user defined text, optional widgets and buttons.

Each button assigns a value to a given parameter, which may be used later in the chain to provide flow control or change data.

All Prompts come up in the same place.



The fields and buttons have the following functions.

Field Description	Type	Defaults	Pop-Up
Type	chain command box		

*the chain command - this command is **Prompt***

Prompt title	text box		

the title to display at the top of the Prompt

Prompt message text box

the message to display

Parameter for result text box

the parameter to store the result of the Prompt in

Widgets to display - grid

An optional list of widgets to display on the panel to capture more information. The widget must be stored in an existing parameter, defined in the chain's PVF file. The type of parameter will determine the type of widget displayed.

For information about grids, see 4.19.6 Grids in Panels.

Name text box

the name of the widget

Parameter parameter box list of parameters for the chain

the parameter that the widget is created for and will set once the prompt is closed. The parameter must be one of the parameters for the chain.

Mode

Optional? tick box

*if **yes**, the widget will be optional and the user will not need to fill it in when the prompt is displayed.*

*if **no** or **blank**, the widget will not be optional and the user will be required to fill in the widget.*

For the description of the fields **Use Parameters**, **Command name**, **Active**, **Continue on failure** *and* **Comments**, *and the button* **Execute** *see* 28.3.1.1.1.1 Common Fields in Commands.

Buttons to display - grid

a set of buttons to display on the Prompt panel, each with a name and a return value.

For information about grids, see 4.19.6 Grids in Panels.

Name

the name of the button

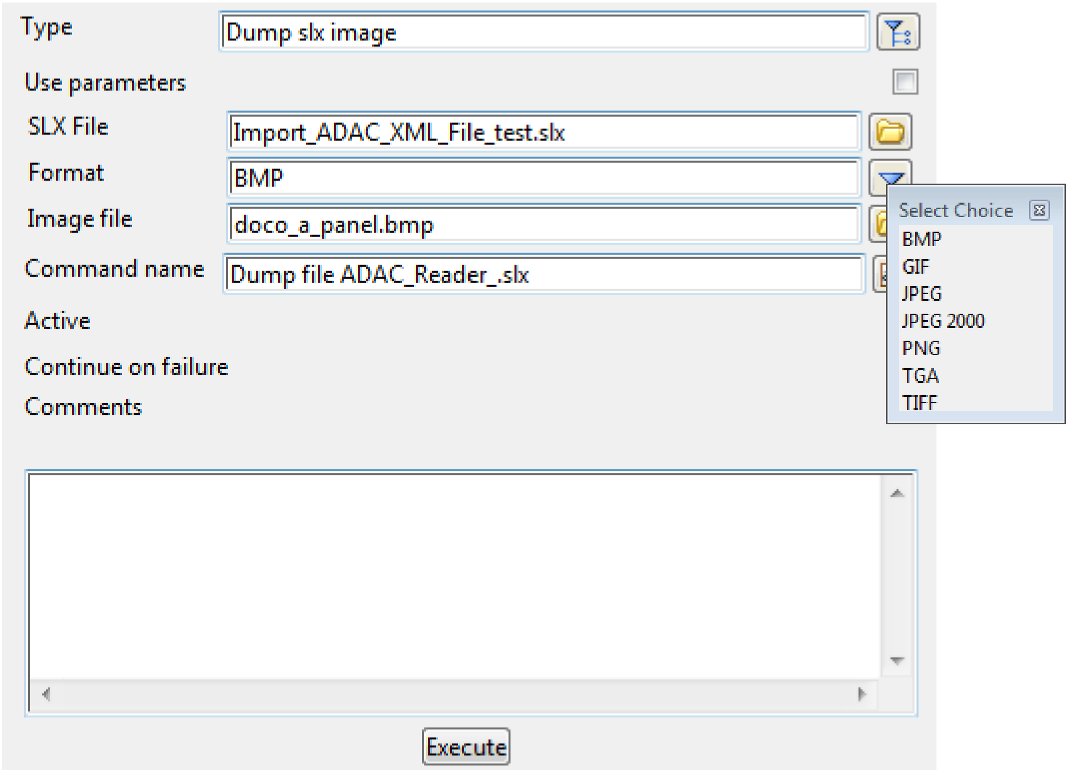
Value

if the button is pressed, the value to assign to the parameter **Parameter for result**.

Continue to the next section 28.3.1.1.2.11 Other >Dump slx image or return to 28.3.1.1.1 Chain Commands or 28.3.1.1 Inserting a Chain Command.

28.3.1.1.2.11 Other >Dump slx image

The **Dump slx image** command write out an image of the panel that the SLX creates.



The fields and buttons have the following functions.

Field Description Type Defaults Pop-Up

Type chain command box

*the chain command - this command is **Dump slx image***

SLX File file box available *.slx commands

name of the slx file to read in and display the panel it describes, and then write out an image for.

Format choice box

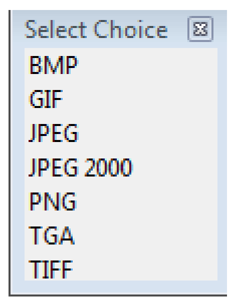


image format to use.

Image file file box

the image is written out to this file.

For the description of the fields **Use Parameters**, **Command name**, **Active**, **Continue on failure** *and* **Comments**, *and the button* **Execute** *see* 28.3.1.1.1.1 Common Fields in Commands.

Continue to the next section 28.3.1.1.2.12 Other >Print parameter or return to 28.3.1.1.1 Chain Commands or 28.3.1.1 Inserting a Chain Command.

28.3.1.1.2.12 Other >Print parameter

The **Print parameter** command writes out information about a chain parameter to the Output Window.

It prints out in the format

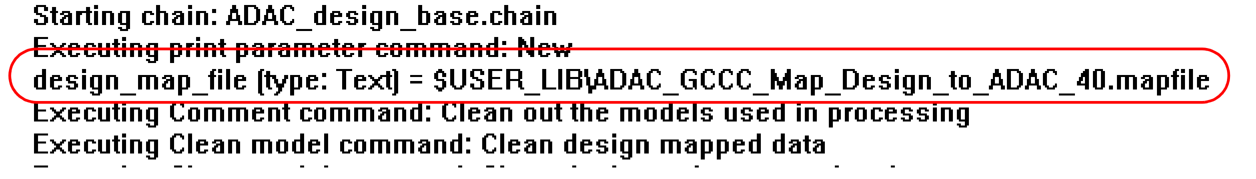
parameter_name [type:*parameter_type*] = *parameter_value*

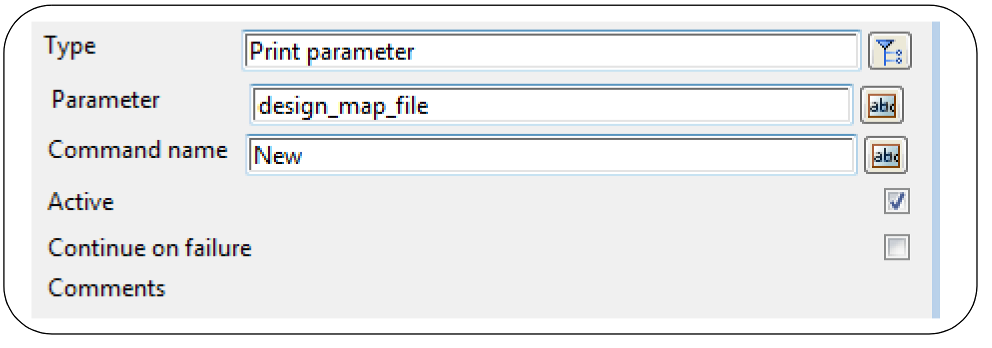
(a) the name of the parameter

(b) the type of the type

(c) the value of the parameters where the *Print parameter* command is processed in the Chain

As an example, for a parameter *design_map_file,* this would print to the ***Output Window***:





The fields and buttons have the following functions.

Field Description	Type	Defaults	Pop-Up
Type	chain command box		

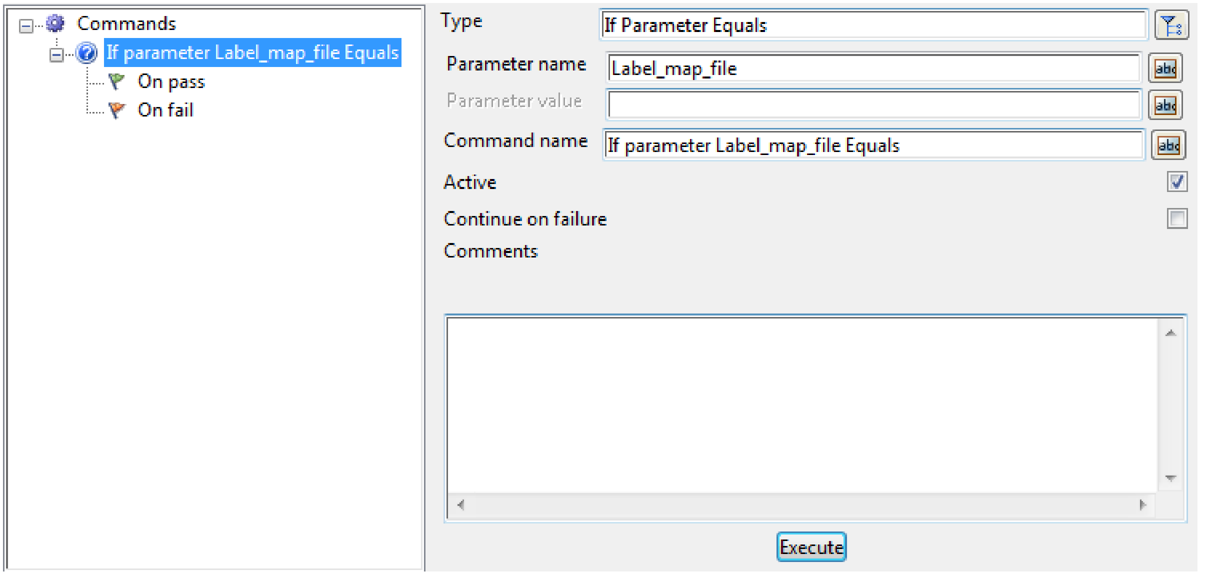
*the chain command - this command is **Print parameter***

For the description of the fields **Command name**, **Active**, **Continue on failure** *and* **Comments**, *and the button* **Execute** *see* 28.3.1.1.1.1 Common Fields in Commands.

Continue to the next section 28.3.1.1.2.13 Conditionals >If Parameter Equals or return to 28.3.1.1.1 Chain Commands or 28.3.1.1 Inserting a Chain Command.

28.3.1.1.2.13 Conditionals >If Parameter Equals

The **If Parameter equals** command tests if a Chain Parameter has a certain value.



The fields and buttons have the following functions.

Field Description	Type	Defaults	Pop-Up
Type	chain command box		

*the chain command - this command is **If Parameter Equals***

Parameter name	parameter box		parameters in this chain

*the name of a parameter from the **Parameter value file** that is to be tested for having a user given value. The pop-up lists all the parameters in the **Parameter value file** for the Chain.*

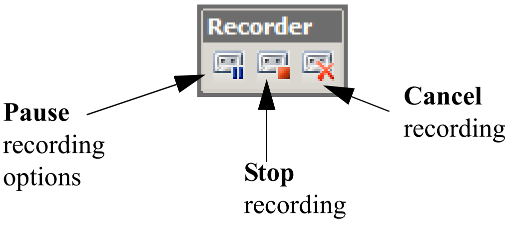
The action of the **On Pass** *and* **On fail** *nodes are described in* 28.3.1.1.1.9 Chain Flow Control and Conditional Commands.

For the description of the fields **Command name**, **Active**, **Continue on failure** *and* **Comments**, *and the button* **Execute** *see* 28.3.1.1.1.1 Common Fields in Commands.

Return to 28.3.1.1.1 Chain Commands or 28.3.1.1 Inserting a Chain Command.

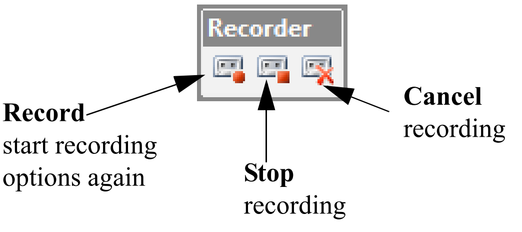
28.3.1.2 Record Icon and the Recorder Toolbar

Clicking on the **Record** icon removes the **Create/Edit Chain** panel and brings up the **Recorder toolbar** with icons to ***Pause***, ***Stop*** and ***Cancel*** the recording.

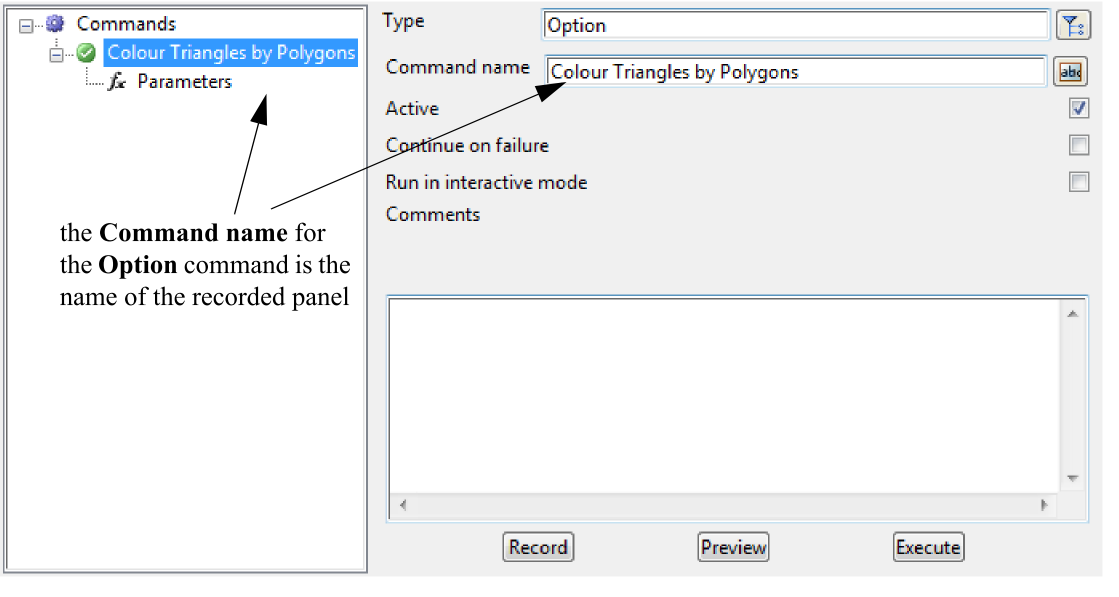


The ***Recording*** process is already underway and you simply ***run*** the ***12d Model*** panels and recordable macros that you wish to record and then press the **Stop** icon.

Pressing **Pause** causes the recording process is interrupted and the **Pause** icon on the Recorder toolbar changes to **Record**. Other options can be run and recording only resumes when **Record** is pressed on the **Recorder** toolbar.



For each recorded panel or macro, an ***Option*** command is added to the list of commands in the chain and its **Command name** is the name of the recorded panel or macro.



When ***Stop*** is pressed, the **Recorder** toolbar is **removed** and the **Create/Edit Chain** panel for the chain is opened on the screen with the new ***Option*** commands added to the command list.

Click on the **Write** button at the bottom of the **Create/Edit Chain** panel to then update the chain.

Note that in **Create/Edit Chain** panel the **Command name** can be modified by clicking on the command in command list in the **Create/Edit Chain** panel and then changing its ***Command name*** field. Click on the **Write** button at the bottom of the **Create/Edit Chain** panel to update the chain.

For more information on recording panel and the buttons **Record**, **Preview** and **Execute**, go to 28.3.1.1.2.1 Execution >Option.

28.3.2 Interactive Mode

Chains support an **Interactive Mode**, meaning that the user is allowed to influence the flow of commands or the data input into the chain.

There are two types of Interactive Mode: **Chain Interactive Mode** and **Command Interactive Mode:**

(a) **Chain Interactive Mode**:

This is toggled via the Run chain in inter.eactive mode tick box on the **Chain Edit/Create** panel. See 28.3.1 Create/Edit a Chain

Then when the chain is run, the **Interactive toolbar** is displayed and the chain will pause for user input. See 28.3.2.1 Interactive Toolbar and Chain Interactive Mode for more details.

(b) **Command Interactive Mode**:

Several commands now support and interactive mode which means that instead of running without the user seeing the command, a command with interactive mode ticked on will pause the chain, display the panel, and allows the user to change panel values and be run or not tun. This applies in the case of panels, functions and macros.

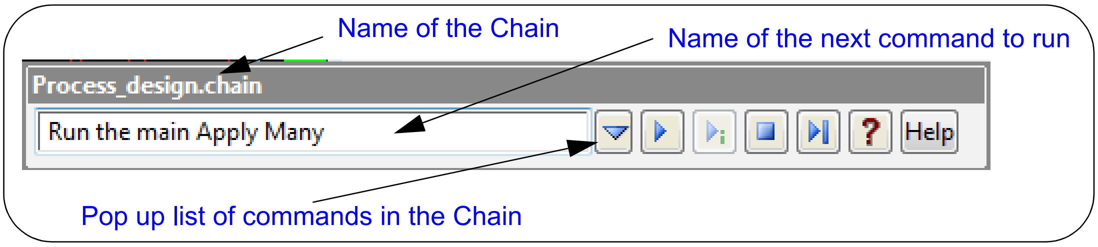
So in these cases, any panels associated with them, and the data they were recorded with, will be displayed and the panel will wait for user input before continuing. See 28.3.2.2 Commands Interactive Mode.

Important Note

The **Chain Interactive Mode** and the **Command Interactive Mode** are totally separate and not related in any way. However when running a **Chain in Interactive Mode**, for each command that supports **Command Interactive Mode**, the user can run the command as is, or **force it** to run in **Command Interactive Mode**.

28.3.2.1 Interactive Toolbar and Chain Interactive Mode

When running a ***Chain*** in **Interactive Mode**, the **Interactive toolbar** is placed on the screen and the chain pauses and ***waits for the user*** to say run the command. The user can even choose a different command to run, and continue running from that command on.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Choice box	choice box		all commands in the Chain

choose the command from the Chain that is placed in the **Interactive** *toolbar ready to be run next.*

Play button button

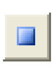
plays the command displayed in the **Interactive** *toolbar.*

Play + i button button

plays the command displayed in the **Interactive** *toolbar in the* ***Command Interactive Mode*** *regardless of the commands own* **Run in interactive mode** *field. See* 28.3.2.2 Commands Interactive Mode.

This button is only available when the command can be run in Command Interactive Mode. For example a ***Comment*** *command can* ***not*** *be run in Command Interactive Mode.*

Stop button button



stops running the Chain but the **Create/Edit Chain** *panel for the Chain is left on the screen.*

Fast forward button button

runs the chain to the end. The **Create/Edit Chain** *panel for the Chain is left on the screen.*

Fx button

 *bring up the* **Chain Parameters Editor** *panel with the* ***Parameter value file*** *for the Chain loaded into it.*

Help button

display the context sensitive help.

28.3.2.2 Commands Interactive Mode

Running a command in **Command Interactive Mode** will do one of several things, depending on the type of the command:

1. If it is a recorded **Option**, the panel will be presented to the user and the user can change values in the panel fields.

   The user either runs the panel or doesn't run the panel, and then selects the **Finish** button or closes the panel with the X icon.

   If the chain is being run from the **Interactive** toolbar, the Chain then moves on and displays the next command in the Chain in the **Interactive** toolbar.

   If the chain is not being run from the **Interactive** toolbar, the Chain then moves on and runs the next command in the Chain.

2. If it is a **Function** command, the chain will attempt to open the function editor.

   The user either runs the function or doesn't run the function, and then selects the **Finish** button or closes the panel with the X icon.

   If the chain is being run from the **Interactive** toolbar, the Chain then moves on and displays the next command in the Chain in the **Interactive** toolbar.

   If the chain is not being run from the **Interactive** toolbar, the Chain then moves on and runs the next command in the Chain.

28.3.3 Edit a Chain

Position of option on menu: **Utilities =>Chains =>Edit**

Position of option on menu: **Utilities =>Recalc =>Edit chain**

Editing an existing chain on either menu can be done two ways.

The **Edit** walk-right menu displays all the chains in the project and double clicking on a chain in the list will bring up the **Create/Edit Chain** panel (see 28.3.1 Create/Edit a Chain for more information).

Another way of editing a chain is by using the **Recalc Chain** panel (yes the **Recalc Chain** panel) that comes up by:

(a) If the **Chain** menu is being displayed by walking right on **Utilities =>Chains =>Edit** on the top **Main Menu** then the **Edit** menu will be displaying with **[Edit]** written on the top. Clicking on **[Edit]** will bring up the **Recalc Chain** panel

If the **Chain** menu is being displayed by walking right on **Utilities =>Recalc =>Edit chains** on the top **Main Menu** then the **Edit** menu will be displaying with **[Edit chain]** written on the top. Clicking on **[Edit chain]** will bring up the **Recalc Chain** panel

(b) If the **Chain** or **Recalc** menu has been torn from the **Main Menu** on the top then the **Recalc Chain** panel comes up if you click on **Utilities =>Chains =>Edit** without walking right or **Utilities =>Recalc =>Edit chain** without walking right.

The **Recalc Chain** panel can run an existing chain, edit an existing Chain or create a new chain.

For information on the **Recalc Chain** panel see 28.3.9 Run a Chain.

Continue to the next section 28.3.4 Chain Parameters or return to 28.3 Chains.

28.3.4 Chain Parameters

The basic usage of chains is to record and playback a set of panels and other chain commands.

The user can have interactive with the command running in the chain by running the Chain in Interactive Mode (see 28.3.2 Interactive Mode) and/or running selected commands in Command Interactive Model (see 28.3.2.2 Commands Interactive Mode).

However even though these options allow you to change values in some panels as the chain runs, this has to be done again every time the chain is run.

To allow for more power and in the way values in the panels fields are changed, **Chain Parameters** were introduced.

Chain Parameters can be used in place of actual values in the panels and the parameters can be given different values each time the chain is run.

For example, you may have a chain that triangulates a data set and creates and labels contours for the created tin. And to make the chain more flexible, you want to give the name of the created tin and contour models each time the chain is run but don't want to do it by hand.

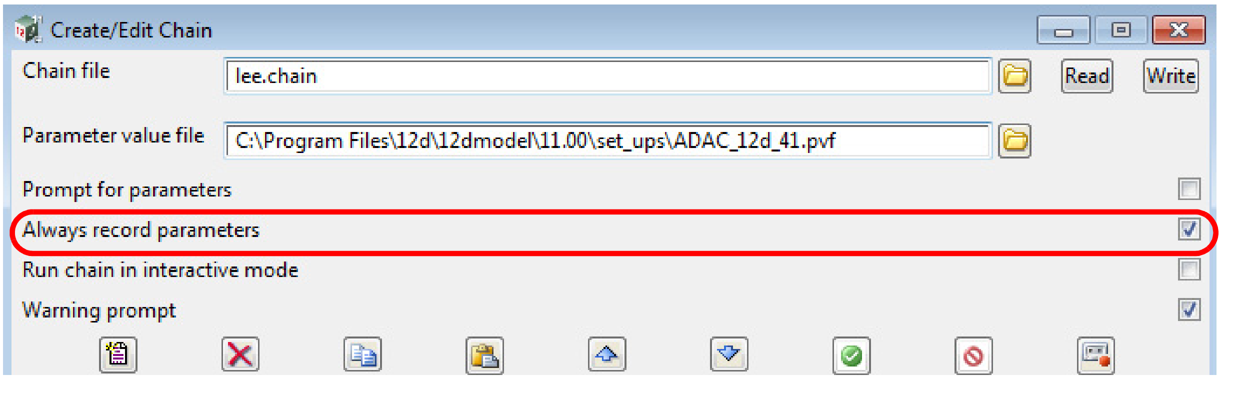
Or you may want to just give the name of the tin and have the names of the contour models derived from the given tin name.

Chain Parameters are used to give the chain this added flexibility.

When Chain Parameters are created, the actual values for the parameters used are stored in a separate **parameter value file** (*.pvf) and so the values of all the parameters can be changed by running the chain with a ***different*** parameter value file.

The Chain Parameters and their values can be defined when the panel is created using the **Option** command, or done separately within the chain itself. Or a commination of both.

To create parameters straight away, have the **Always record parameters** field ticked **on** before starting to record an Option.

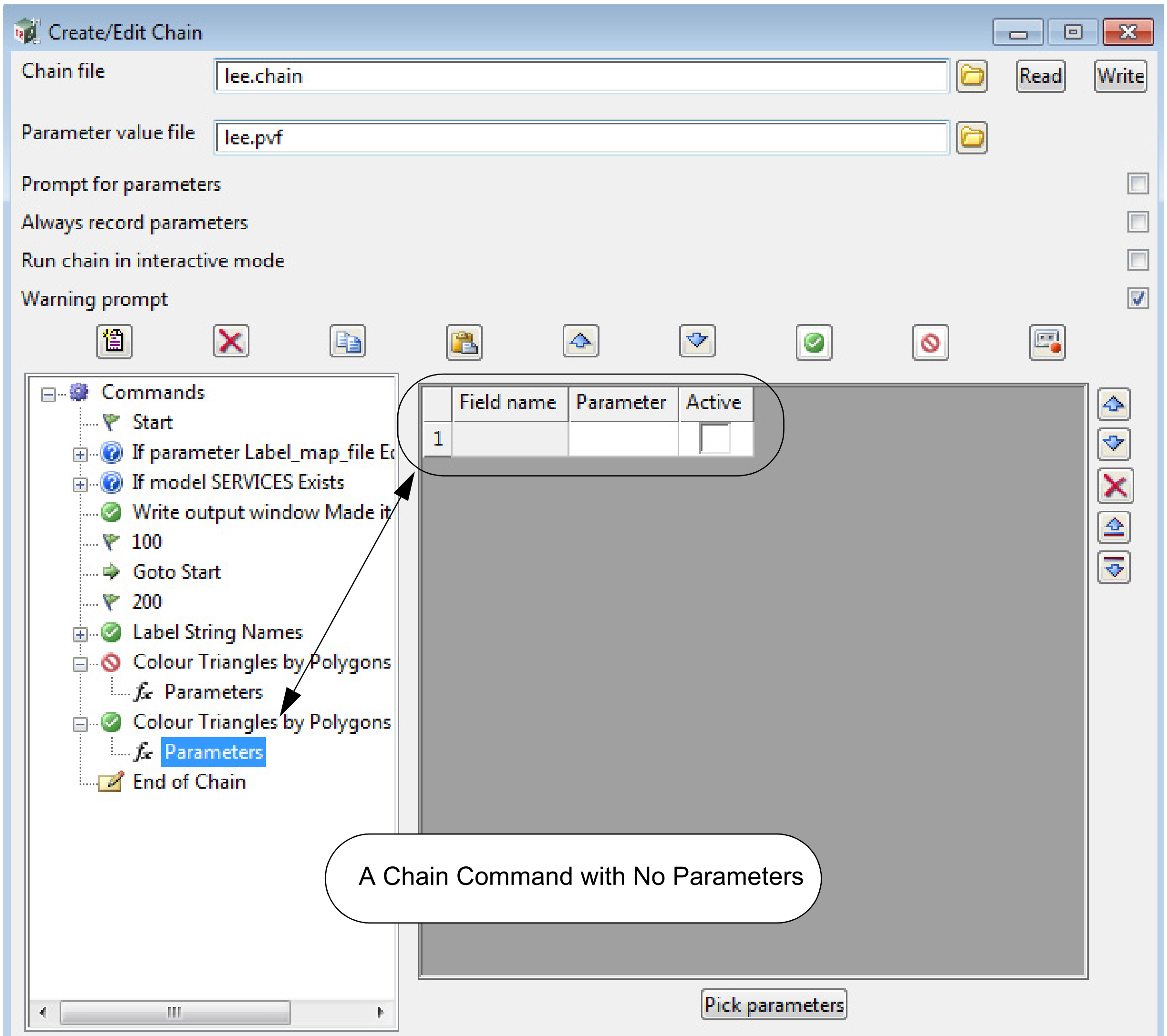


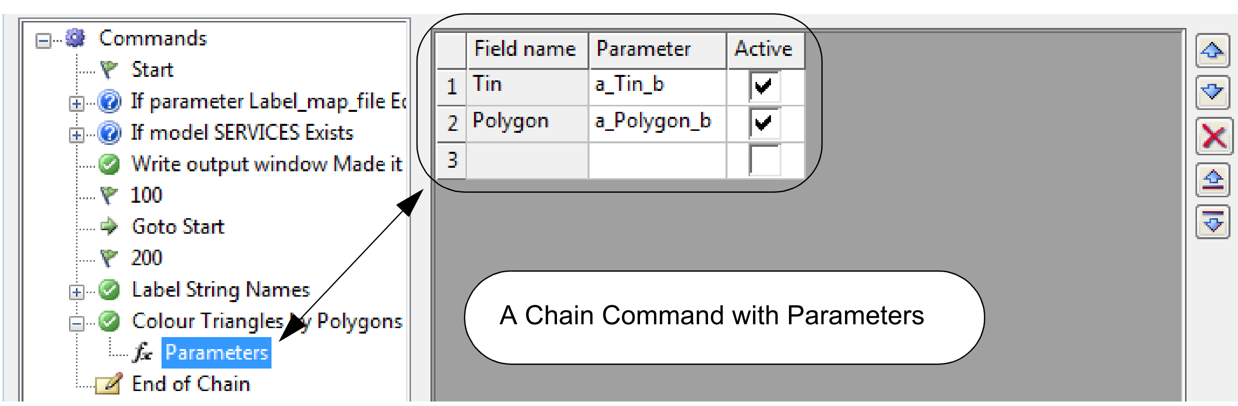
After the option is finished and the recording stops, the **Record Option Parameters** panel is displayed and is used to select the fields you wish to parameterise. See 28.3.4.2 Record Option Parameters Panel.

For information on how to create parameters separately to the recording process, or how to modify existing parameters, see 28.3.4.1 Creating and Modifying Parameters Outside the Recording Process.

28.3.4.1 Creating and Modifying Parameters Outside the Recording Process

For a command in the **Create/Edit Chain** panel, to create parameters separately to the recording process, or to modify existing parameters, click on the **+** to the left of the command in the command list and then on the *Parameter line*. The Parameters are then displayed on the right hand side of the panel.





The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Parameter grid

the parameter grid holds information about each defined parameter in the chain.

Field name display only

the name of the panel field to be parameterised. This is written to when a field is selected by the **Pick parameters** *buttons. This is read only.*

Parameter text box

the name the user gives the parameter. This can also be modified.

Note that the parameter name can only be alphanumeric and underscores. No spaces are allowed in the name

Active tick box

if ***ticked****, this row is used.*
If ***not ticked****, the row is not used and acts like a comment.*

Buttons at Bottom

Pick parameters

this is used to define more parameters for the option by picking a field from the recorded panel to use as a parameter.

To create a new parameter for the command, click on the **Pick parameters** button. This will bring up the **Record Options Parameter** panel which is used to pick new parameters.

For a description of the **Record Options Parameter** panel and information on how to use it to pick parameters, see 28.3.4.2 Record Option Parameters Panel.

Highlighted rows in the grid can be deleted using the X icon.

New rows are created using the **Pick parameters** button.

If a *Parameter* is not to be used at this time, ***untick*** the **Active** column.

Parameters within options are defined as *Parameter Mappings.*

A parameter mapping maps from a specific field to a named parameter. When an option command runs and fills out a particular field, it will look for the appropriate parameter mapping to that field to use as the value for the field.

To create parameters without recording, or modify them, click on the Parameters under the expanded Option command in the chain editor. All the mappings from parameters to the fields they represent will be listed here.

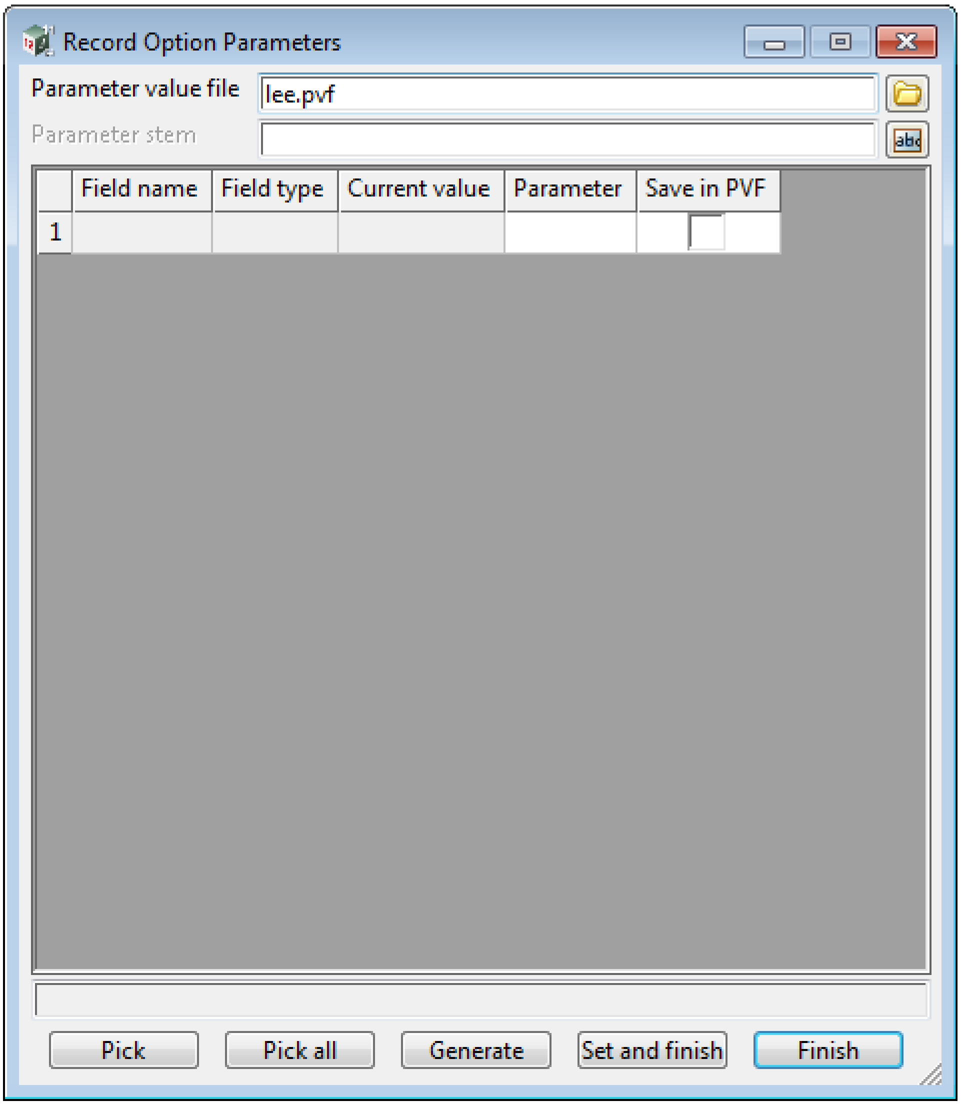
28.3.4.2 Record Option Parameters Panel

The **Record Option Parameters** panel is displayed:

(a) as soon as a recording is finished if the **Always record parameters** field ticked **on** in the **Create/Edit Chain** panel.

or

(b) by clicking on the **Pick parameters** button when the *Parameters* line is clicked on for a command in the **Create/Edit Chain** panel.



The fields and buttons for this panel will now be described but how you use the panel to select parameters is given in the section 28.3.4.3 How to Create Chain Parameters

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Parameter value file	parameter value file box		available *.pvf files

the name of the field on the panel that is being parameterised.

Parameter stem	text box		

*pretext and postext written in the form pre_text*post_text and it is used on the value that is going to be written in the* **Parameter** *column in the grid when it is created by the* **Generate** *button.*

*Note that if there are any spaces in the pre_text*post_text, then each space will be replaced by an underscore in the parameter name.*

Parameter grid name

the parameter grid holds information about each defined parameter in the chain.

Field name display only

the name of the panel field to be parameterised. This is written to when a field is selected by either the **Pick** *or* **Pick all** *buttons. This is read only.*

Field type display only

the type of the parameterised field. This is written to when a field is selected by either the **Pick** *or* **Pick all** *buttons This is read only.*

Current value display only

the value that the panel field had at the time of recording. This is read only.

Parameter text box

the name of the parameter. If **Generate** *is pressed, then this is automatically created from the* **Parameter stem** *and the name of the Field in the panel.*

Note that the parameter name can only be alphanumeric and underscores. Not spaces are allowed in the name

Save in PVF tick box

*if **ticked**, this row is saved in the* **parameter value file**.
*If **not ticked**, the row is not saved in the* **parameter value file**.

However in both cases the information is saved with the chain.

Buttons at Bottom

Pick

pick a field from the recorded panel to use as a parameter

Pick all

automatically pick all fields to be used as parameters

Generate

generate names for the parameters, using the optional parameter stem

Set and finish

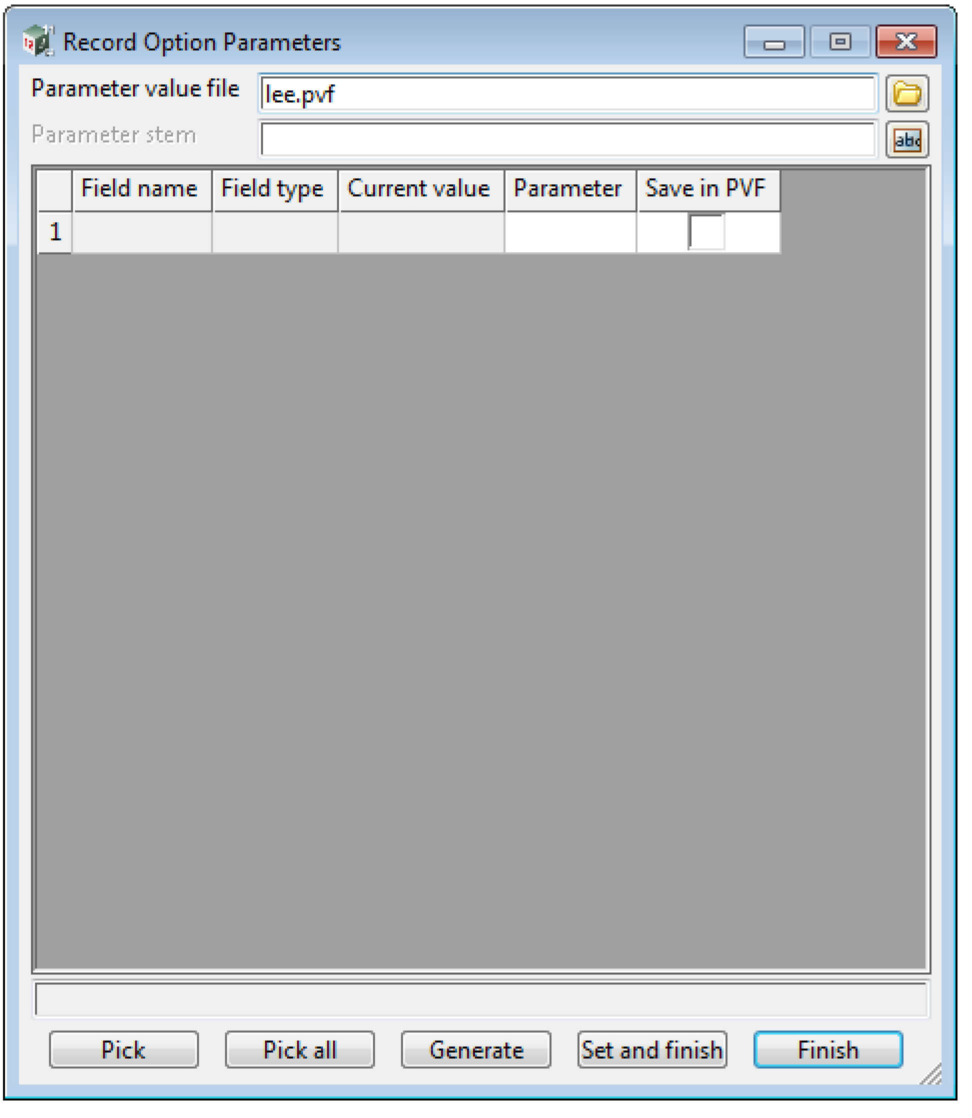
save the recorded parameters and finish the panel.

For information on how to use this panel to select parameters is given in the section 28.3.4.3 How to Create Chain Parameters.

28.3.4.3 How to Create Chain Parameters

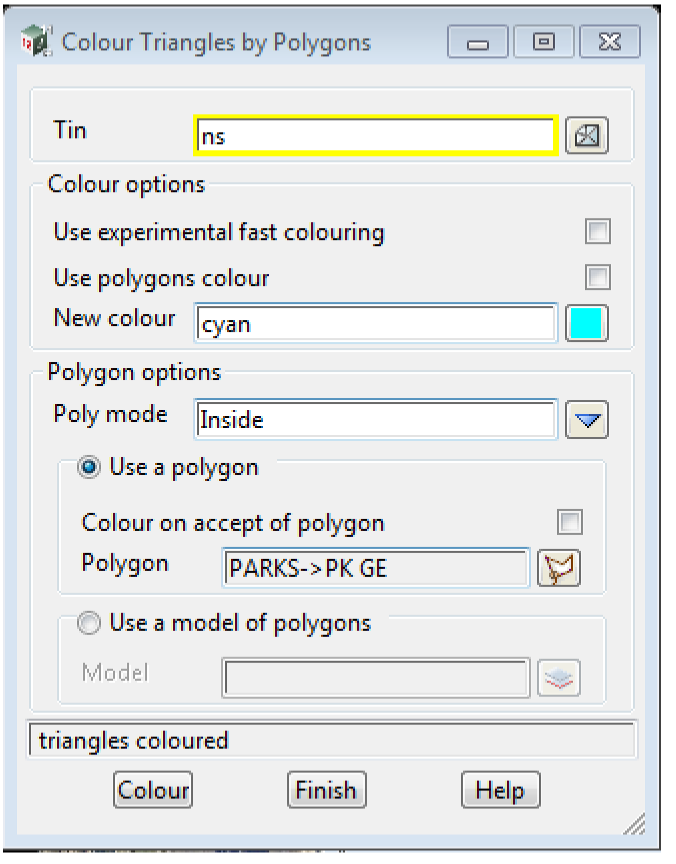
The **Record Option Parameters** is used to create ***Chain Parameters*** and this section describes that process in detail.

Note - information on each field and button in the panel is given in the section 28.3.4.2 Record Option Parameters Panel.

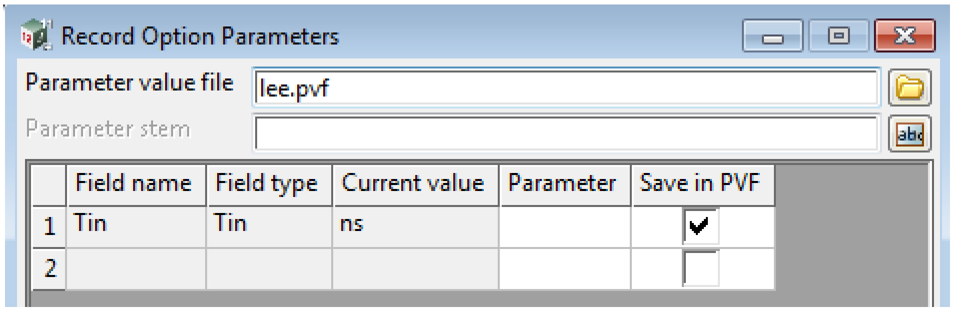


The **Pick** or the **Pick All** buttons can then be used to select the fields that are to be parameterised.

After **Pick** is pressed, as you move over the panel that was just recorded, the fields of the panel that the cursor is over is highlighted with a yellow rectangle.



Clicking when a particular field is highlighted writes the name, type and current value of the field to the next blank line in the grid of the **Record Option Parameters** panel.



The name that you want to call the parameter is then typed into the **Parameter** column and the **Save in PVF** column ticked or not ticked.

If another panel field is to be parameterised, then press the **Pick** button and repeat the process.

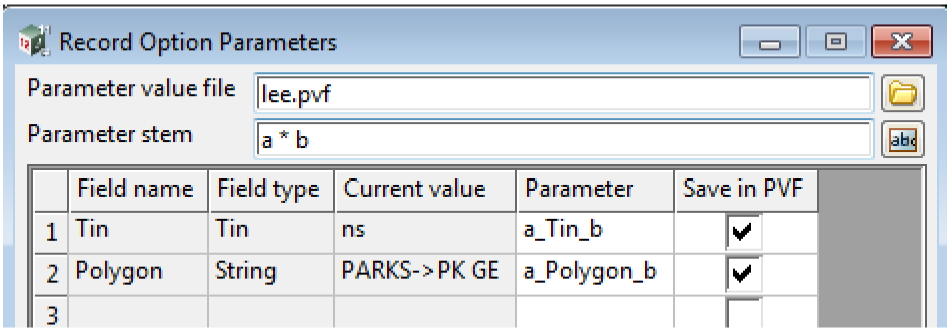
When all the fields to be parameterised have been selected, click on **Set and finish**.

The **Record Option Parameters** panel is then removed from the screen. Click on Finish to remove the panel that was recorded.

Click on the **Write** button on the top right hand side of the **Create/Edit Chain** panel to update the chain in the file, and also if there is one, update the **Parameter value file**.

Note that sometimes the name of the command is not updated until **Write** is pressed.

Another way to create the **Parameter** names is to click on **Generate**. The **Parameter stem** is then applied to the name of the field on the panel to create the **Parameter** name.

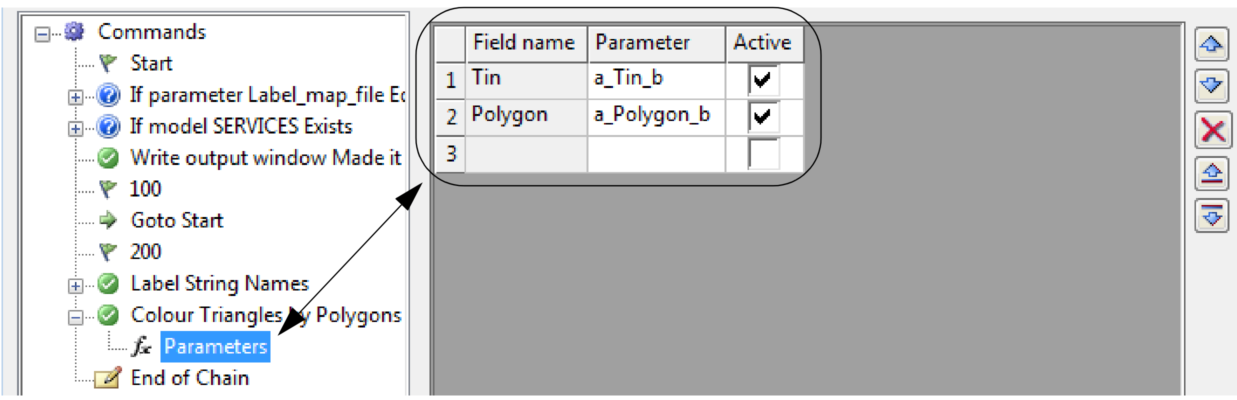


Note that a parameter name can not contain spaces so any spaces are replaced by underscores.

Finally if all panel fields are to be selected as parameters then press the **Generate all** button.

Also if you want to do most parameters but not all, it is often quicker to click on **Generate all** and then delete the few that you don't want from the **Record Option Parameters** panel.

The parameters for a command can be seen by clicking on the + to the left of the command in the command list and then clicking on the Parameter line that is displayed. The Parameters are then displayed in the right hand side of the panel.



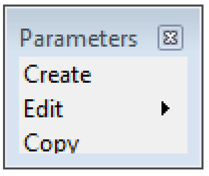
If the *Parameter* is not to be used at this time, ***untick*** the **Active** column.

Return to 28.3.4.2 Record Option Parameters Panel or 28.3.4.3 How to Create Chain Parameters.

28.3.5 Creating/Editing Parameter Value Files

Position of option on menu: **Utilities =>Chains =>Parameters**

These options Create, Edit and Copy Chain pvf files (parameter value files).



selecting Create opens the parameter editor
walk right on Edit, lists parameter files to edit
used to copy a Chain Parameter Value File from one file to another.

For the option:

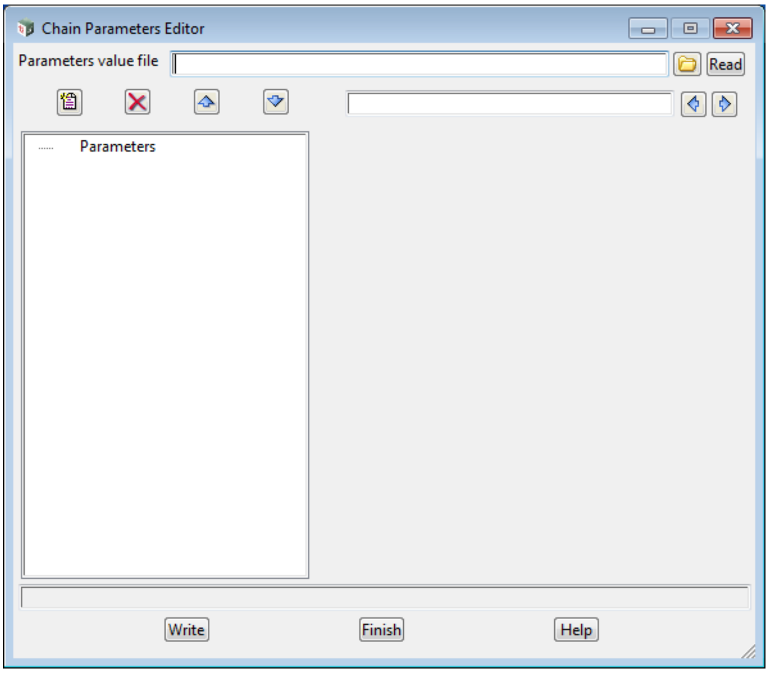
Create, go to	28.3.5.1 Creating a Chain Parameters Value File
Edit	28.3.5.2 Editing a Chain Parameters Value File
Copy	28.3.5.3 Copy a PVF

Chains can also be used

28.3.5.1 Creating a Chain Parameters Value File

Position of option on menu: **Utilities =>Chains =>Parameters =>Create**

Selecting **Create** brings up the **Chain Parameters Editor** panel which is used to both create and edit chain parameters.



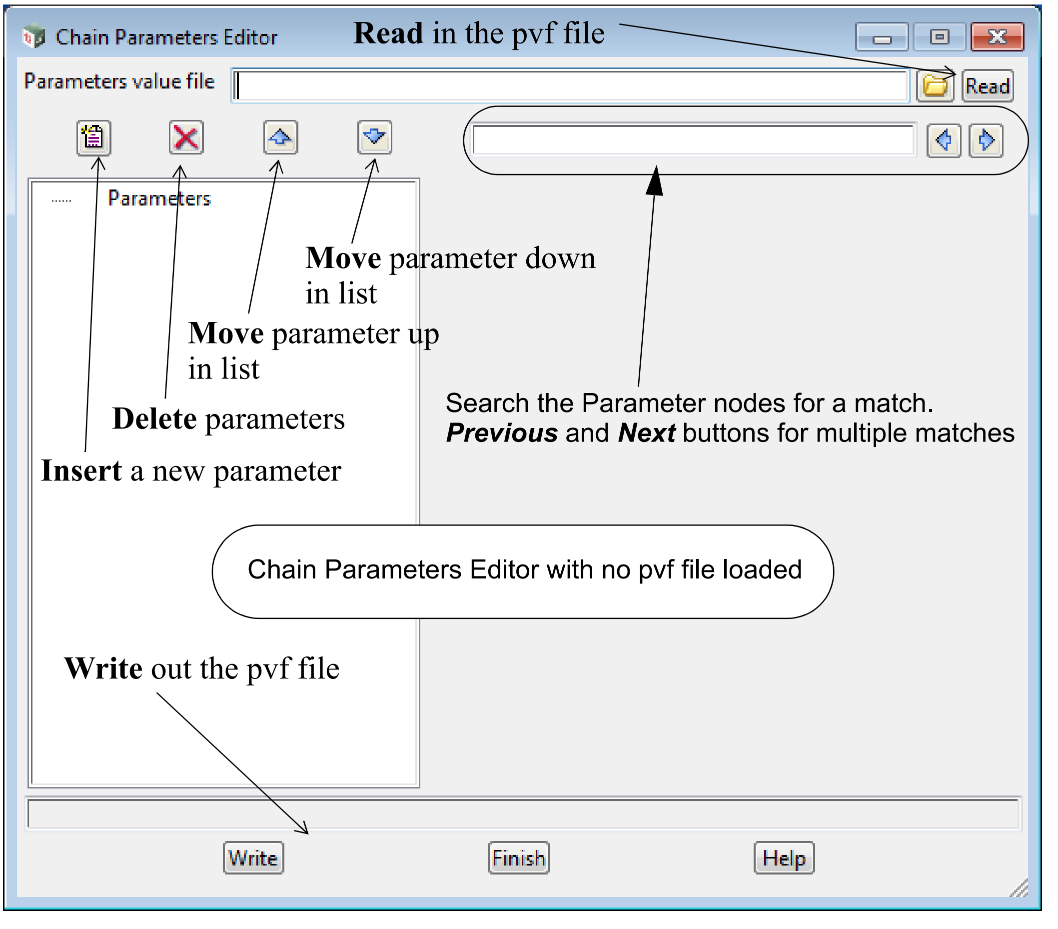
For information about this panel, see 28.3.5.2 Editing a Chain Parameters Value File

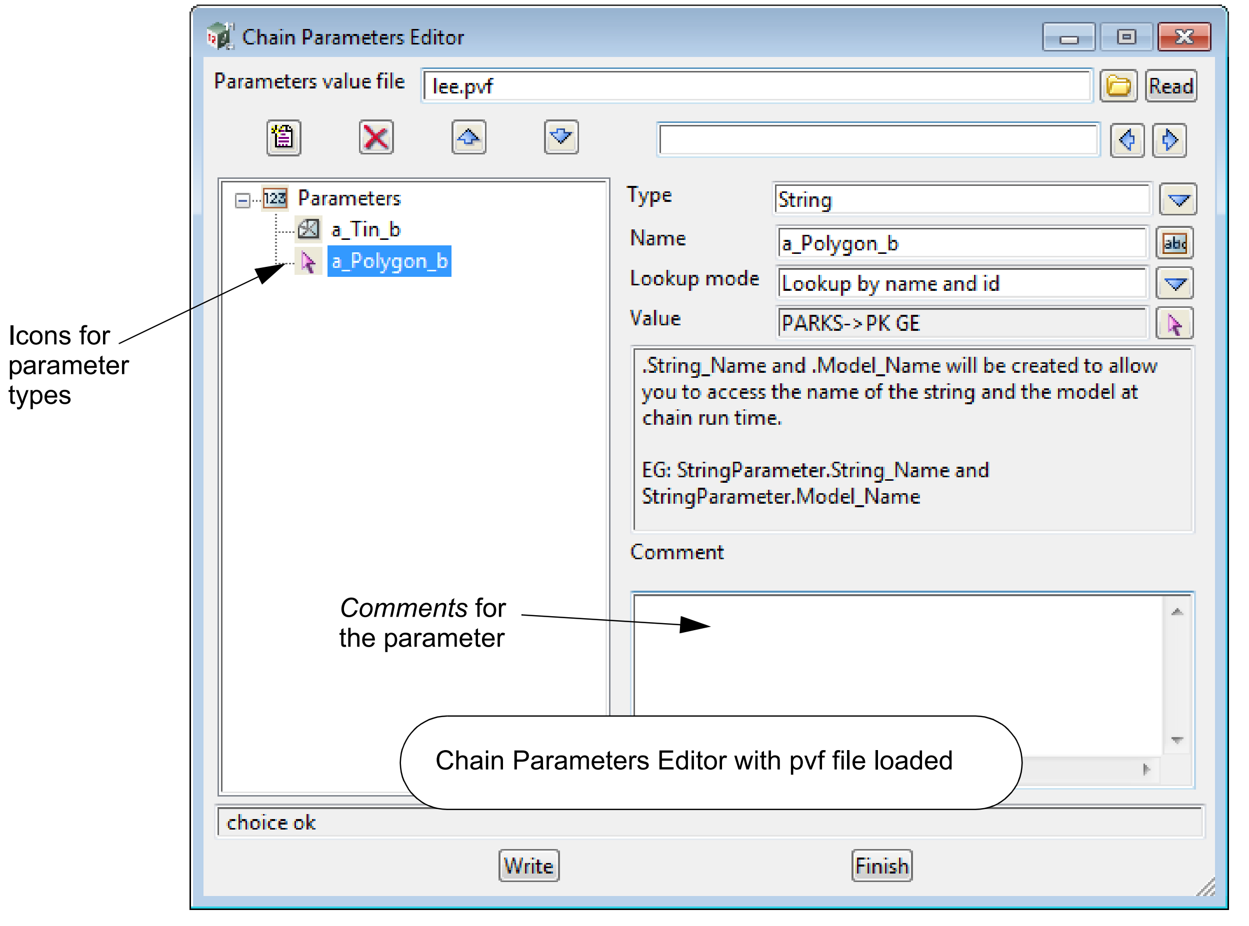
28.3.5.2 Editing a Chain Parameters Value File

Position of option on menu: **Utilities =>Chains =>Parameters =>Edit**

The **Edit** option is a walk-right which lists the existing parameter files. Clicking on a file in this list, brings it up in the **Chain Parameters Editor** panel with the selected pvf file already loaded into the panel.

Or if the **Parameter** menu is a floating menu, clicking on **Edit** without walking right will bring up the **Chain Parameters Editor** panel without a pvf file already loaded. This can then be used to create a new pvf file or selected and load an existing pvf file.





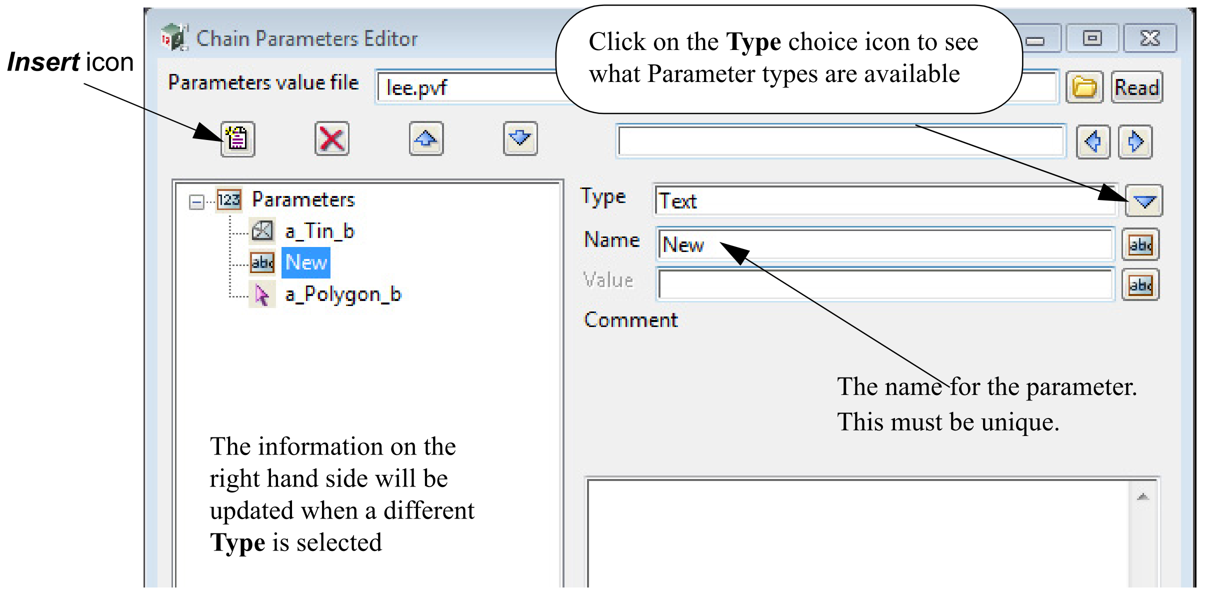
Chain Parameters Editor with pvf file loaded

See:

Inserting a parameter 28.3.5.2.1 Inserting a Parameter
Searching 28.3.5.2.2 Searching for Parameter Values

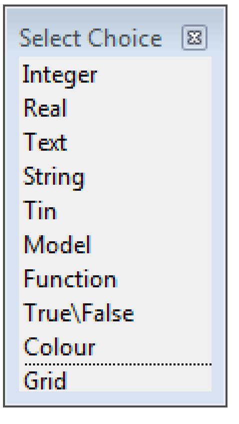
28.3.5.2.1 Inserting a Parameter

Clicking on the ***Insert*** icon insert a new parameter into the list of parameters under the current highlighted parameter. The new parameter is given the default **Type** Text and name **New**.



The correct type for the new parameter is then selected in the **Type** field and the information required for that type of parameter is then displayed in the right hand side of the panel.

The types of parameters available are:

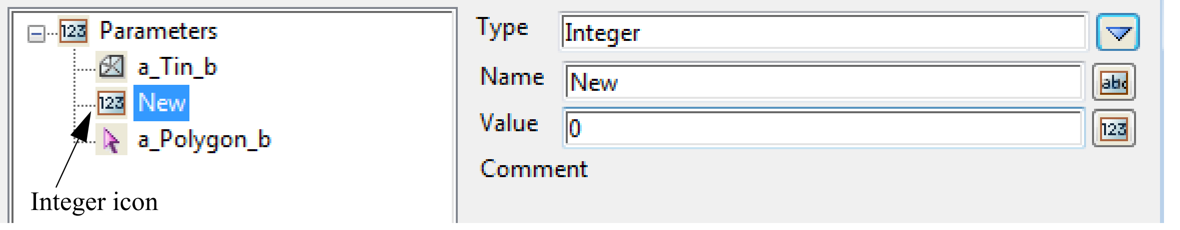


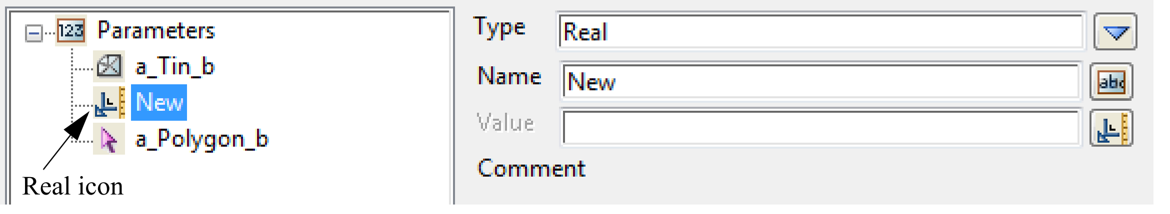
See:

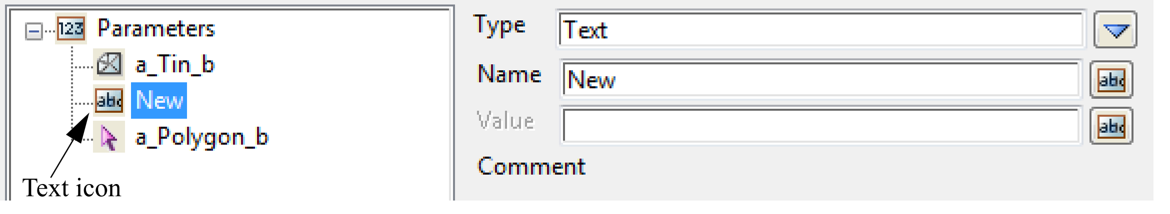
Integer	28.3.5.2.1.1 Integer, Real, Text, Function, True/False and Colour Parameters
Real	28.3.5.2.1.1 Integer, Real, Text, Function, True/False and Colour Parameters
Text	28.3.5.2.1.1 Integer, Real, Text, Function, True/False and Colour Parameters
String	28.3.5.2.1.2 String Parameter
Tin	28.3.5.2.1.3 Tin Parameter
Model	28.3.5.2.1.4 Model Parameter
Function	28.3.5.2.1.1 Integer, Real, Text, Function, True/False and Colour Parameters
True\False	28.3.5.2.1.1 Integer, Real, Text, Function, True/False and Colour Parameters
Colour	28.3.5.2.1.1 Integer, Real, Text, Function, True/False and Colour Parameters
Grid	28.3.5.2.1.5 Grid Parameter

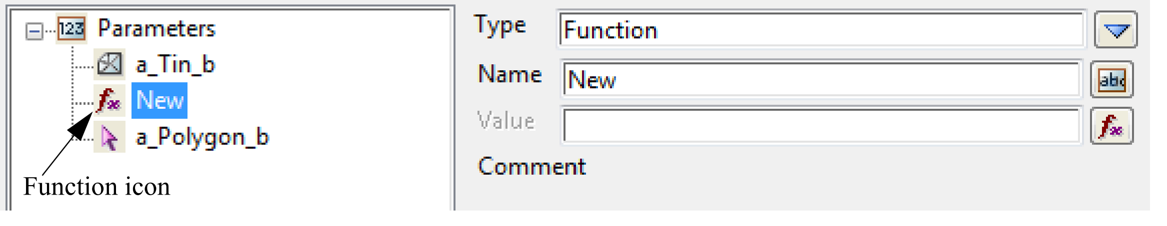
28.3.5.2.1.1 Integer, Real, Text, Function, True/False and Colour Parameters

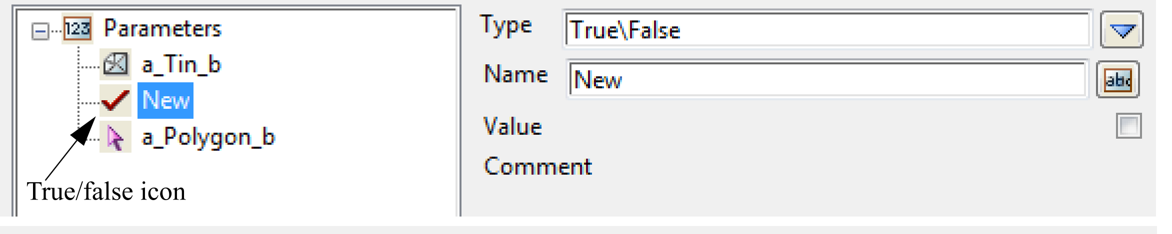
The information needed for all these parameter types is virtually the same.

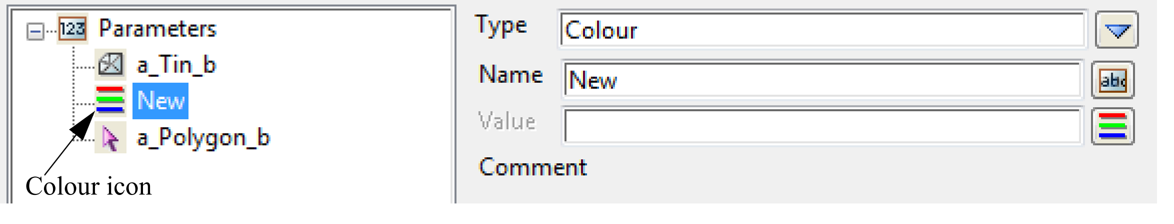












The fields and buttons used in the right hand side of the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Type	choice box		parameter types

*the type of the parameter. In this case it will be **Integer**, **Real**, **Text**, **Function, True/False** or **Colour**.*

*The icon displayed on the left of the parameter name in the Parameter list under the Parameter node will match the **Type**.*

Name text box

the name of the parameter. This must be unique within the pvf file.
The name becomes the name displayed in the Parameter list under the Parameter node.

The name in this list may not change until the **Write** *button is pressed.*

Value integer, real, text, function, tick box, colour

the value of the parameter. This must match the type of the parameter (integer, real, text, function, true/false tick box, colour).

For **Type** ***True/False***, **Value** *is a tick box and* ***ticked*** *means* ***True*** *and* ***unticked*** *means* ***False***.

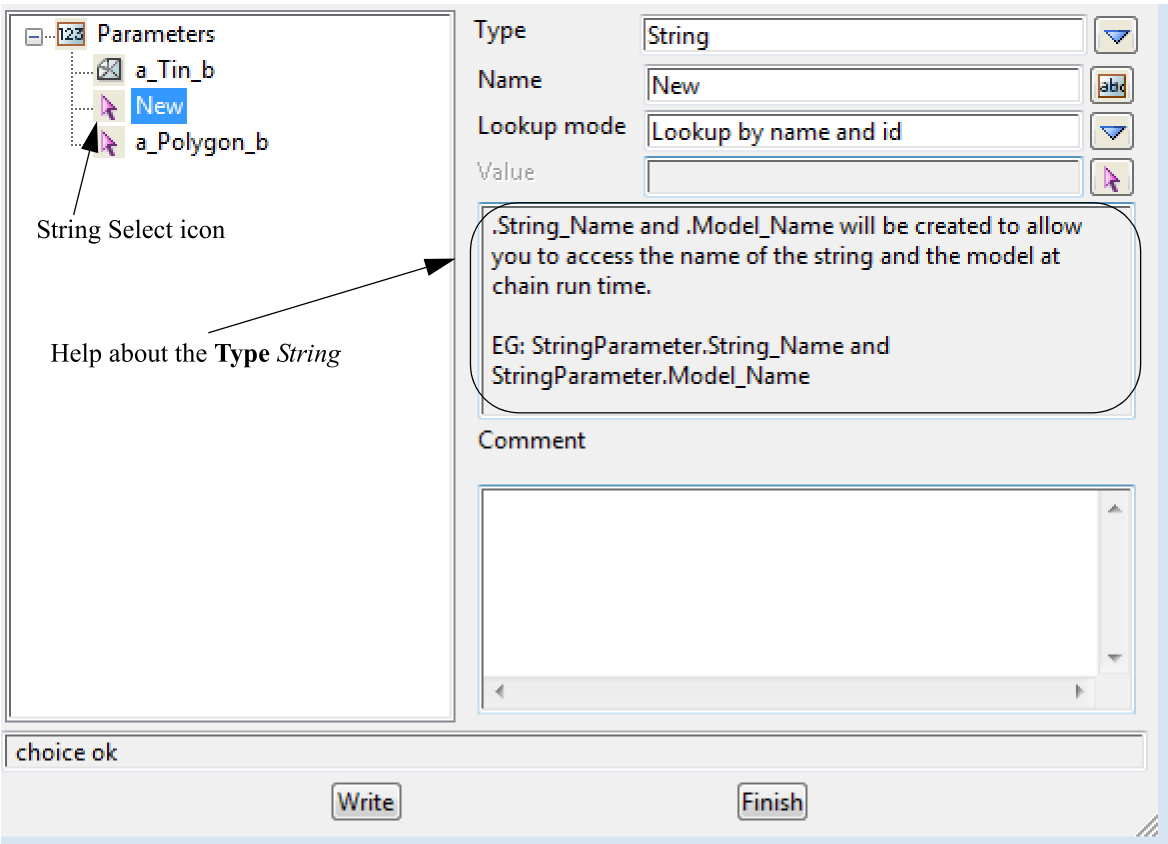
Comment comment box

user comments to be stored with the parameter.

Continue to the next section 28.3.5.2.1.2 String Parameter or return to 28.3.5.2.1 Inserting a Parameter.

28.3.5.2.1.2 String Parameter

The information needed for the parameter of type **String** is given on the right hand side of the panel when ***String*** is selected for **Type**.



The fields and buttons used in the right hand side of the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Type	choice box		parameter types

*the type of the parameter. In this case it is **String**.*

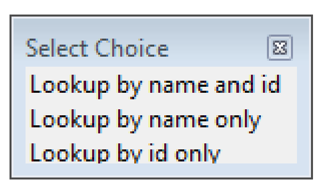
*The **Select String** icon displayed on the left of the parameter name in the Parameter list under the Parameter node.*

Name	text box		

the name of the parameter. This must be unique within the pvf file.
The name becomes the name displayed in the Parameter list under the Parameter node.

The name in this list may not change until the **Write** *button is pressed.*

Lookup mode	choice box		



*if **Lookup by name and id**, the string to use is searched for by string id first and if the string id does not exist, then by string name. If the search becomes by name and there is more than one string with the name then the first one found is selected. THIS NEEDS TO BE EXTENDED FOR name and model*

*if **Lookup by name only**, the string to use is searched for by string name only. If there is more than one string with the name then the first one found is selected. THIS NEEDS TO BE EXTENDED FOR name* *and model*

*if **Lookup by id only**, the string to use is searched for by id only. THIS NEEDS TO BE EXTENDED FOR name and model*

Value string select box

a string is selected by clicking on the string select icon on the right of the field and then selecting a string. The string name and id, and the model name and id that the string is in, are saved with the parameter.

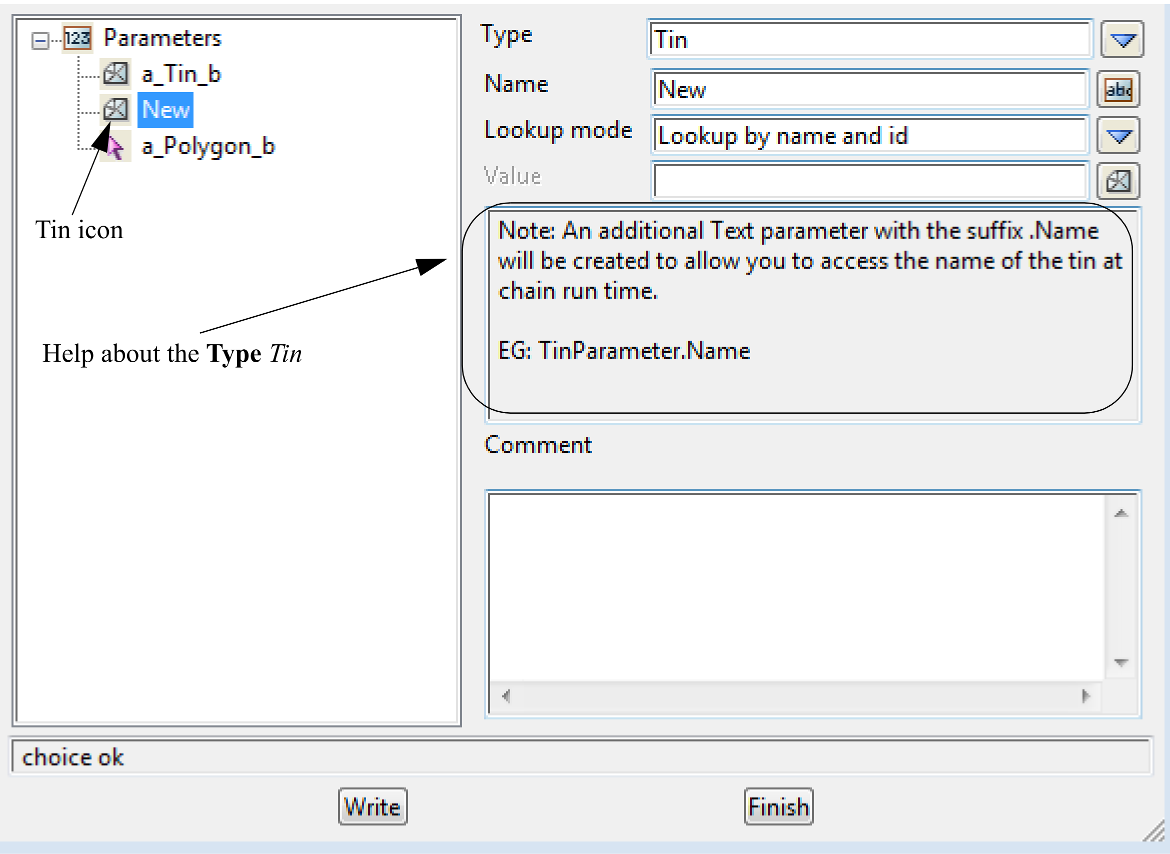
Comment comment box

user comments to be stored with the parameter.

Continue to the next section 28.3.5.2.1.3 Tin Parameter or return to 28.3.5.2.1 Inserting a Parameter.

28.3.5.2.1.3 Tin Parameter

The information needed for the parameter of type **Tin** is given on the right hand side of the panel when ***Tin*** is selected for **Type**.



The fields and buttons used in the right hand side of the panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Type | choice box | | parameter types

*the type of the parameter. In this case it is **Tin**.*

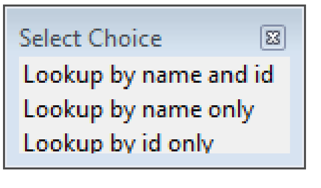
*The **Tin** icon displayed on the left of the parameter name in the Parameter list under the Parameter node.*

Name | text box

the name of the parameter. This must be unique within the pvf file.
The name becomes the name displayed in the Parameter list under the Parameter node.

The name in this list may not change until the **Write** *button is pressed.*

Lookup mode | choice box



*if **Lookup by name and id**, the tin to use is searched for by tin id first and if the tin id does not exist, then by tin name.*

*if **Lookup by name only**, the tin to use is searched for by tin name only.*

*if **Lookup by id only**, the tin to use is searched for by id only.*

Value tin box

the name of the tin. The tin name and id are saved with the parameter.

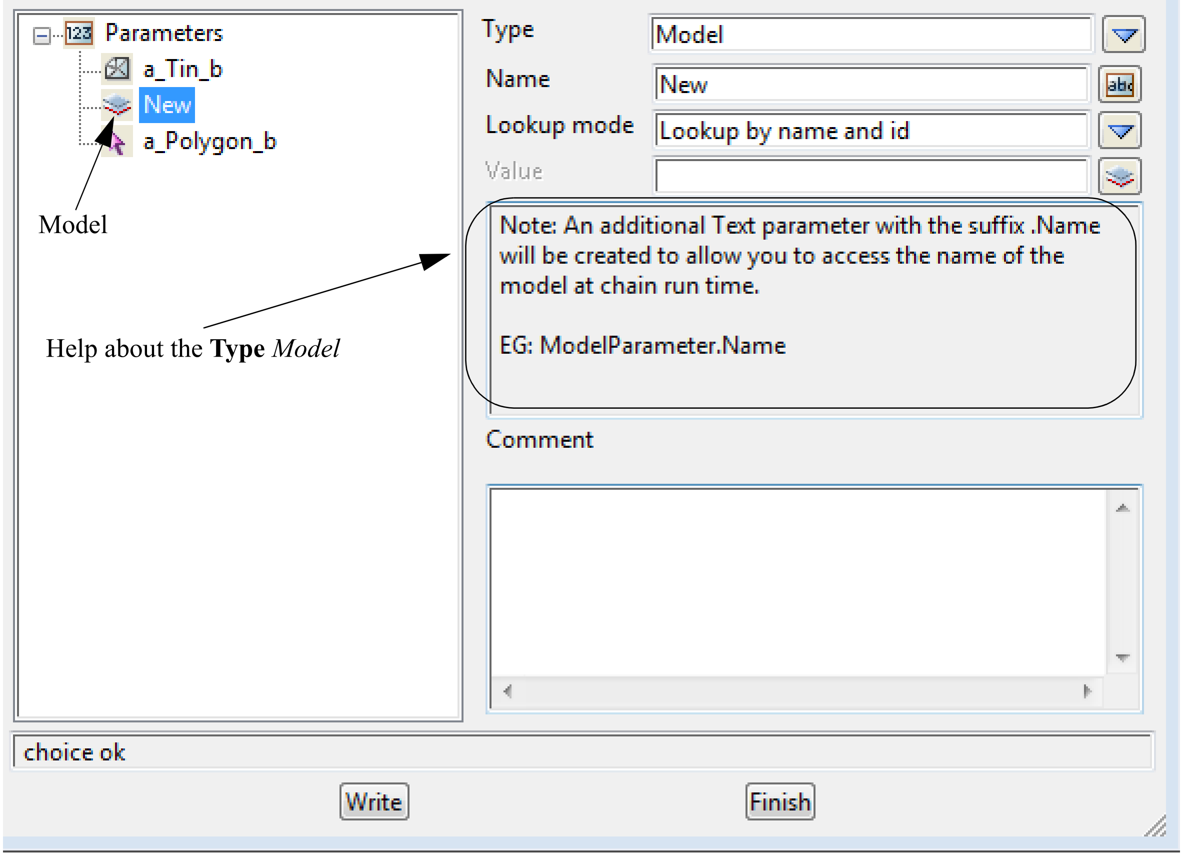
Comment comment box

user comments to be stored with the parameter.

Continue to the next section 28.3.5.2.1.4 Model Parameter or return to 28.3.5.2.1 Inserting a Parameter.

28.3.5.2.1.4 Model Parameter

The information needed for the parameter of type **Model** is given on the right hand side of the panel when ***Model*** is selected for **Type**.



The fields and buttons used in the right hand side of the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Type	choice box		parameter types

*the type of the parameter. In this case it is **Model**.*

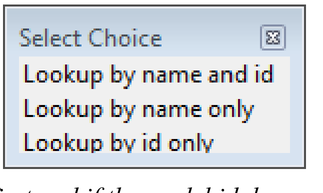
*The **Model** icon displayed on the left of the parameter name in the Parameter list under the Parameter node.*

Name	text box		

the name of the parameter. This must be unique within the pvf file.
The name becomes the name displayed in the Parameter list under the Parameter node.

The name in this list may not change until the **Write** *button is pressed.*

Lookup mode	choice box		



*if **Lookup by name and id**, the model to use is searched for by model id first and if the model id does not exist, then by model name.*

*if **Lookup by name only**, the model to use is searched for by model name only.*

*if **Lookup by id only**, the model to use is searched for by id only.*

Value tin box

the name of the model. The model name and id are saved with the parameter.

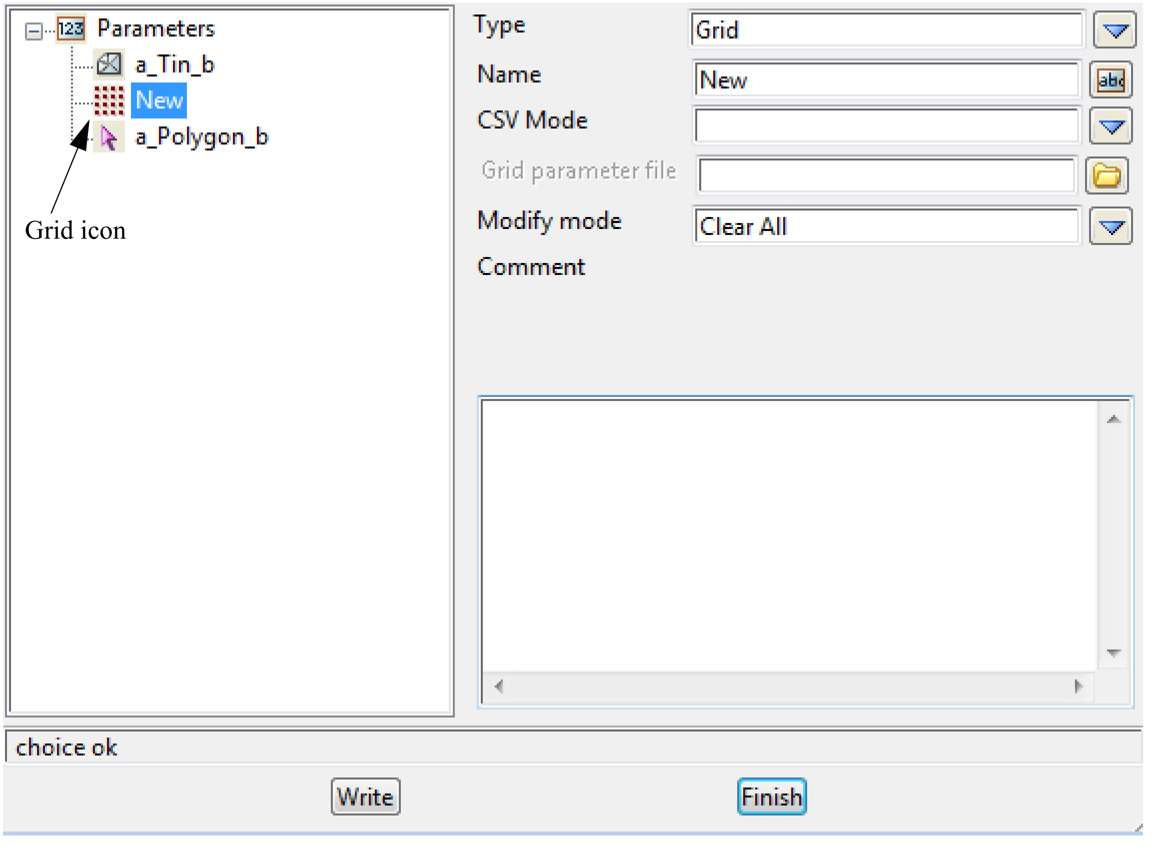
Comment comment box

user comments to be stored with the parameter.

Continue to the next section 28.3.5.2.1.5 Grid Parameter or return to 28.3.5.2.1 Inserting a Parameter.

28.3.5.2.1.5 Grid Parameter

The information needed for the parameter of type **Grid** is given on the right hand side of the panel when ***Grid*** is selected for **Type**.



The fields and buttons used in the right hand side of the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Type	choice box		parameter types

*the type of the parameter. In this case it is **Grid**.*

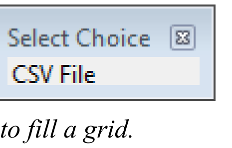
*The **Grid** icon is displayed on the left of the parameter name in the Parameter list under the Parameter node.*

Name	text box		

the name of the parameter. This must be unique within the pvf file.
The name becomes the name displayed in the Parameter list under the Parameter node.

The name in this list may not change until the **Write** *button is pressed.*

CSV mode	choice box		

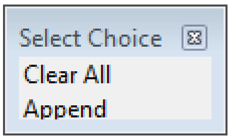


*if **CSV file**, a CSV (comma separated value) file, which defines the values to fill a grid.*

Grid parameter file	file box		all files in working area

the name of the CSV file.

Modify mode choice box



*if **Clear All**, all the existing values in the grid are deleted and are replaced by those in the CSV file.*

*If **Append**, any existing value in the grid in the panel that has the same name as one in the CSV file is replaced by the one in the CSV file. All values in the CSV file that do not already exist in the grid in the panel are added to the grid in the panel.*

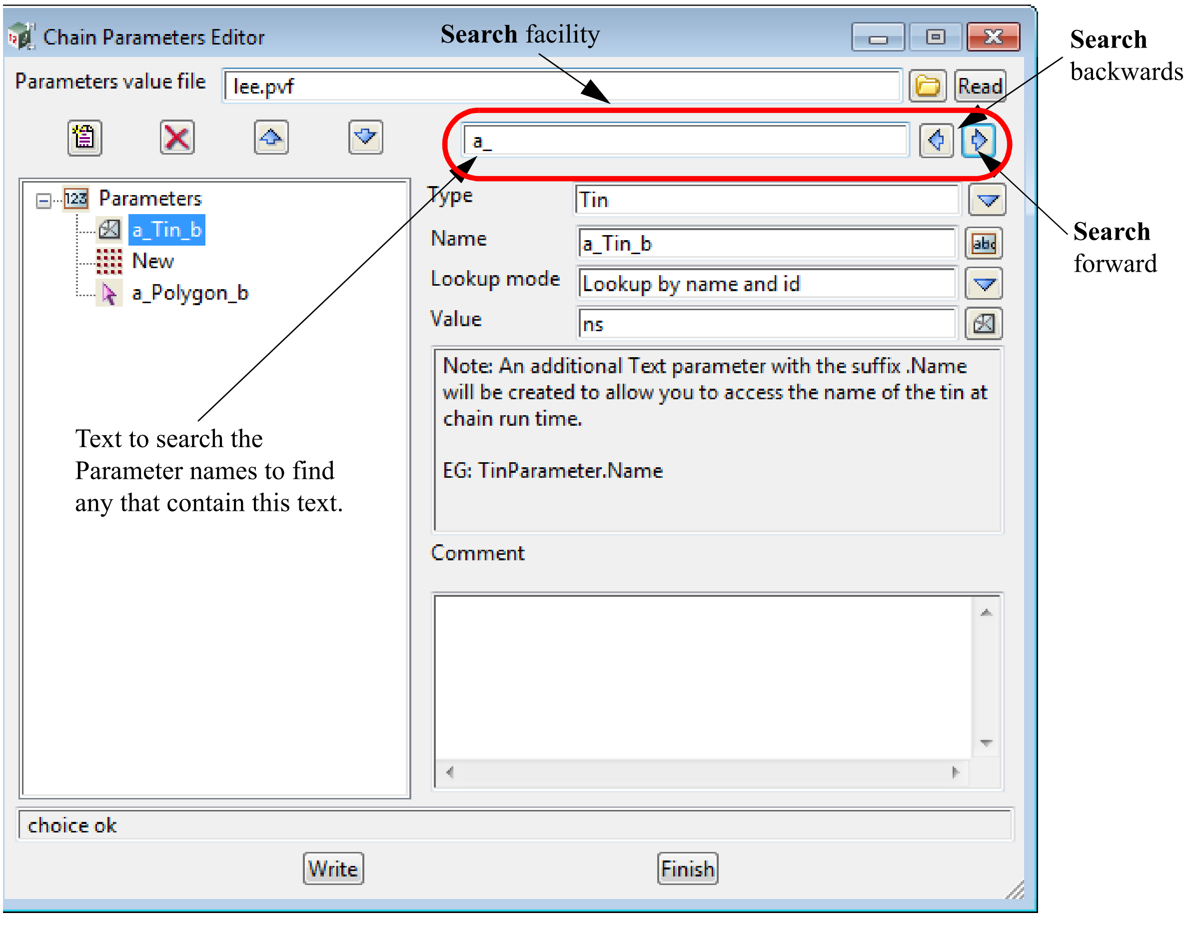
Comment comment box

user comments to be stored with the parameter.

Return to 28.3.5.2.1 Inserting a Parameter.

28.3.5.2.2 Searching for Parameter Values

The **Chain Parameters Editor** panel has a search facility to find any parameter names that contain some user given text.



Text is entered into the search field and if the button for:

Backward search is pressed then the parameter names are searched backwards from the current highlighted node until a name containing the text is found. The parameter node of the found name is highlighted so that information for it is displayed in the right had side of the panel. The search stops when the beginning of the parameter nodes is reached.

Forward search is pressed then the parameter names are searched forward from the current highlighted node until a name containing the text is found. The parameter node of the found name is highlighted so that information for it is displayed in the right had side of the panel. The search stops when the end of the parameter nodes is reached.

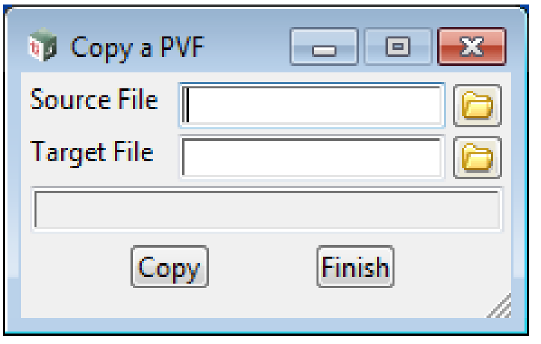
Return to 28.3.5.2 Editing a Chain Parameters Value File.

28.3.5.3 Copy a PVF

Position of option on menu: **Utilities =>Chains =>Parameters=>Copy**

Copy a PVF is used to copy a ***Chain Parameter Value File***.

Selecting **Copy** brings up the **Copy a PVF** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Source File	file box		available *.pvf files
the original PVF file to copy.			
Target File	file box		available *.pvf files
the name of the new copied PVF file.			
Copy	button		
copies the PVF file.			

Continue to the next section 28.3.5.4 Running Chains from User Menus and Toolbars or return to 28.3.5 Creating/Editing Parameter Value Files.

28.3.5.4 Running Chains from User Menus and Toolbars

Chain can be run from a user menu or a user toolbar by using the **Command**:

Command "chain *chain_file_name*"

where *chain_file_name* is the name of the chain.

For User Menus and Toolbars, it is also possible to **substitute** the name of the **parameter value file (pvf)** to use for the chain *chain_file_name* **in place of the pvf mentioned in the chain** *chain_file_name* itself.

The command is

Command "chain *-pvf pvf_file_name chain_file_name*"

where ***pvf_file_name*** is the name of the **pvf** file to use when the chain *chain_file_name* is run from this User Menu or Toolbar.

Note that the

-pvf *pvf_file_name*

is still inside the double quotes (").

Using the ***-pvf*** means that the one base chain can be easily used in user menus and toolbars but with different pvf files.

For example, the chain *ADAC_Survey_base.chain* occurs many times in the "ADAC Survey 4.0 Chains" menu but each time it is run with a different pvf file.

```
Menu "ADAC BRC 4.0 Chains" {
   Button "BRC Survey to ADAC 40 chain" {
      Command "chain -pvf $USER/BRC/ADAC_BRC_40.pvf $LIB/ADAC_survey_base.chain"
   }
   Button "BRC Design to ADAC 40 chain" {
      Command "chain -pvf $USER/BRC/ADAC_BRC_40.pvf $LIB/ADAC_design_base.chain"
   }
}
```

different pvf files

Same chain ADAC_survey_base.chain Run but With Different pvf Files

For more information on User Menus and User Toolbars, see 41.2 User Defined Menus and 41.3 User Defined Toolbars in the Appendix 41 Functions Keys, Menus, Toolbars.

Continue to the next section 28.3.5.5 Chain Parameters as Macro (12dPL's) Arguments or return to 28.3.5 Creating/Editing Parameter Value Files.

28.3.5.5 Chain Parameters as Macro (12dPL's) Arguments

When the Chain Command

Execution >Macro

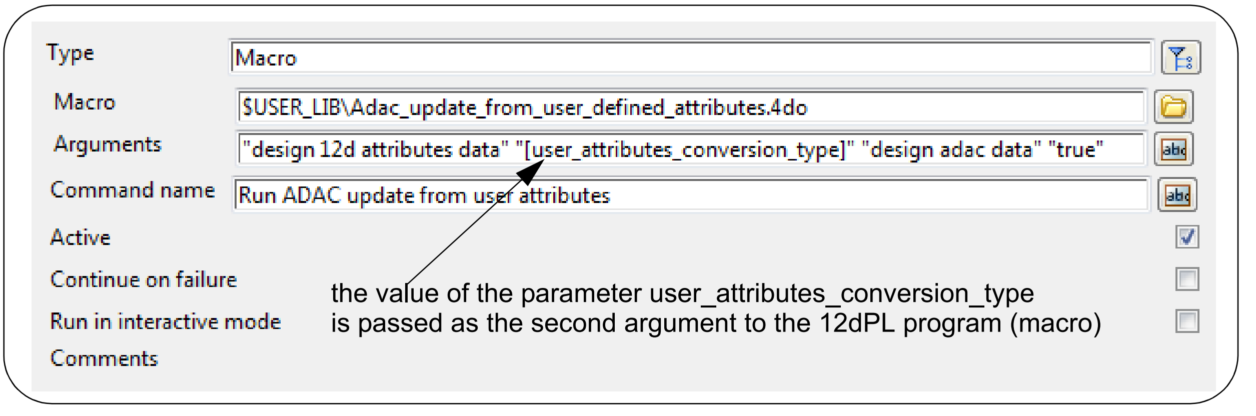
is used to run a 12dPL program (also known as 12PL's or ***12d Model*** macros), the **value** of a **Chain parameter** can be used as an argument passed to a 12dPL program.

This is done via the **Arguments** field in the Macros Command.

The Parameter Name is enclosed in square brackets (**[** and **]**) and then surrounded by double quotes (")

"[parameter_name]"

and this is entered in the **Arguments** field in the place of the argument that it replaces.



For information on the chain command **Macro**, see 28.3.1.1.2.5 Execution >Macro.

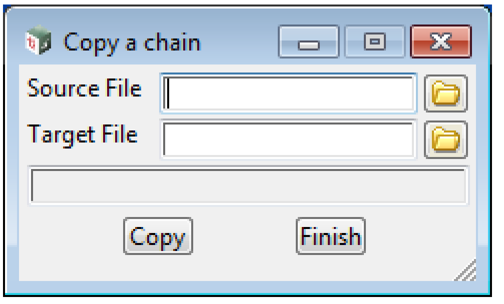
Return to 28.3.5 Creating/Editing Parameter Value Files or 28.3 Chains.

28.3.6 Copy a Chain

Position of option on menu: **Utilities =>Chains =>Copy**

This option copies a chain.

Selecting **Copy** bring up the **Copy a Chain** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Source file	file		available *.chain, *.rcn files

name of the chain to be copied.

Target file	file		

new name for the copied chain.

Copy	button		

Copy the chain.

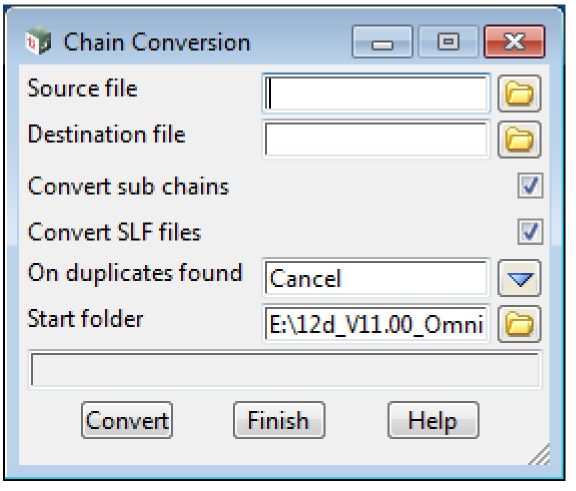
Continue to the next section 28.3.7 Chain Conversion or return to 28.3 Chains.

28.3.7 Chain Conversion

Position of option on menu: **Utilities =>Chains =>Convert**

This option converts a pre-v10 chain **rcn** file to the ***.chain*** format.

Selecting **Convert** brings up the **Chain Conversion** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Source file	file		available *.rcn files

the name of the chain in ***rcn*** *format*

Destination file	file		available *.chain files

the name converted chain in the ***.chain*** *format*

Convert sub chains tick box

if ***ticked****, all sub chains will be converted.*

if ***not ticked****, no sub chains will be converted.*

Convert SLF files

if ***ticked****, all slf files included in the chain will be converted to the new SLX format.*

if ***not ticked****, no slf files will be converted.*

Convert button

converts the source ***rcn*** *file into the new .chain file format.*

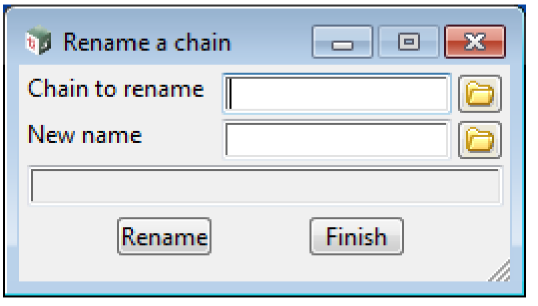
Continue to the next section 28.3.8 Rename a Chain or return to 28.3 Chains.

28.3.8 Rename a Chain

Position of option on menu: **Utilities =>Chains =>Rename**

This option renames an existing chain.

Selecting **Rename** brings up the **Rename a Chain** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Chain to rename	file box		available *.chain, *.rcn files
chain to be renamed.			
New name	text box		available *.chain, *.rcn files
new name for the chain.			
Rename	button		
Change the name of the chain.			

Continue to the next section 28.3.9 Run a Chain or return to 28.3 Chains.

28.3.9 Run a Chain

Position of option on menu: **Utilities =>Chains =>Run**

Position of option on menu: **Utilities =>Recalc =>Run chain**

These options can run an existing chain and also create a new chain or edit an existing chain.

Running an existing chain on either menu can be done two ways.

The **Run** or **Run chain** walk-right menu displays all the chains in the project and double clicking on a chain in the list will run the chain.

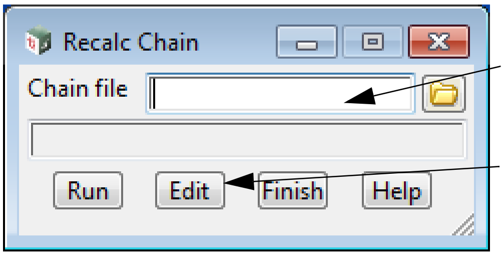
Another way of running a chain is by using the **Recalc Chain** panel that comes up by:

(a) If the menu is being displayed by walking right on **Utilities =>Chains =>Run** on the top **Main Menu** then the **Run** menu will be displaying with **[Run]** written on the top. Clicking on **[Run]** will bring up the **Recalc Chain** panel

If the menu is being displayed by walking right on **Utilities =>Recalc =>Run chain** on the top **Main Menu** then the **Run chain** menu will be displaying with **[Run chain]** written on the top. Clicking on **[Run chain]** will bring up the **Recalc Chain** panel

(b) If any of the menus getting to the **Chain** or **Recalc** menus have been torn from the top **Main Menu** then the **Recalc Chain** panel comes up by clicking on **Utilities =>Chains =>Run** without walking right or **Utilities =>Recalc =>Run chain** without walking right.

The **Recalc Chain** panel can run an existing chain, or edit an existing chain, or create a new chain.



to ***create*** a chain, type in a new name for the chain (the **.chain** will be automatically added)

Clicking on ***Edit*** brings up the **Create/Edit Chain** panel.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Chain file	file box		available *.chain and *.rcn files

the name of the chain file to create/edit or run.

If the file does not exist, a new Chain of that name will be created.

Run	button		

if the chain exist then the chain will be run.

Edit	button		

if the chain exist then it will be brought up in the **Create/Edit Chain** *panel for editing.*

Position of option on menu: **Utilities =>Chains =>Edit**

Position of option on menu: **Utilities =>Recalc =>Edit chain**

Editing an existing chain on either menu can be done two ways.

The **Edit** walk-right menu displays all the chains in the project and double clicking on a chain in the list will bring up the **Create/Edit Chain** panel (see 28.3.1 Create/Edit a Chain for more information).

Another way of editing a chain is by using the **Recalc Chain** panel that comes up by:

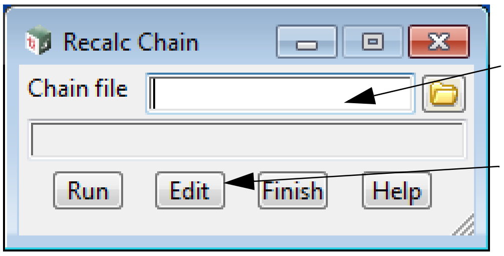
(a) If the **Chain** menu is being displayed by walking right on **Utilities =>Chains =>Edit** on the top

Main Menu then the **Edit** menu will be displaying with **[Edit]** written on the top. Clicking on **[Edit]** will bring up the **Recalc Chain** panel

If the **Chain** menu is being displayed by walking right on **Utilities =>Recalc =>Edit chains** on the top **Main Menu** then the **Edit** menu will be displaying with **[Edit chain]** written on the top. Clicking on **[Edit chain]** will bring up the **Recalc Chain** panel

(b) If the **Chain** or **Recalc** menu has been torn from the **Main Menu** on the top then the **Recalc Chain** panel comes up if you click on **Utilities =>Chains =>Edit** without walking right or **Utilities =>Recalc =>Edit chain** without walking right.

The **Recalc Chain** panel can edit an existing Chain or create a new chain.



to ***create*** a chain, type in a new name for the chain (the **.chain** will be automatically added)

Clicking on ***Edit*** brings up the **Create/Edit Chain** panel.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Chain file	file box		available *.chain, *.rcn files

the name of the chain file to create.

Run	button		

if the chain exist then the chain will be run.

Edit	button		

if the chain in **Chain file** *exist then it will be brought up in the* **Create/Edit Chain** panel for editing.

If the chain in **Chain file** *does not exist then it is created and then brought up in the* **Create/Edit Chain** panel for editing.

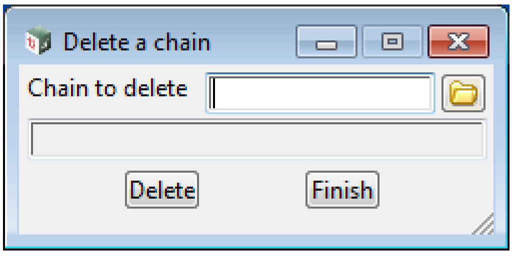
Continue to the next section 28.3.10 Delete a Chain or return to 28.3 Chains.

28.3.10 Delete a Chain

Position of option on menu: **Utilities =>Chains =>Delete**

This option deletes a chain.

Selecting **Delete** brings up the **Delete a Chain** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Chain to delete	folder box		available *.chain,*.rcn files

name of the chain to delete.

Delete	button		

delete the chain given in the **Chain to delete** *field.*

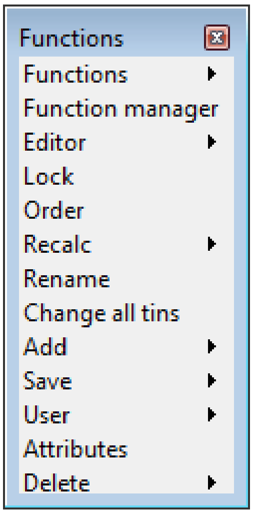
Return to 28.3 Chains.

28.4 Functions

Position of menu: **Utilities =>Functions**

Functions are used in the template and interface options to collect special data together. This allows 12d Model to recognise when some of the data has been modified and that some of the data is no longer valid. It is then possible to re-run the functions (re-calc the functions) and update the modified data.

The **Functions** walk-right menu is



list of functions in the project

modify an existing function
lock a function so it won't recalc
modify the recalc order of functions
recalculate functions
rename a function
change the name of tins in functions
add functions to the project
save a functions to disk

edit attributes of a function
delete functions from disk

The **Functions** walk-right simply lists all the functions that have been defined in the project. The other options in this menu will now be described in more detail.

For the option *Editor*, go to	28.4.1 Editor
Lock	28.4.2 Lock
Order	28.4.3 Order
Recalc	28.4.4 Recalc
Rename	28.4.5 Rename
Change all tins	28.4.6 Change all Tins
Add	28.4.7 Add
Save	28.4.8 Save
Attributes	28.4.9 Attributes
Delete	28.4.10 Delete
Delete many	28.4.10.2 Delete Many Functions

28.4.1 Editor

On walking right on the **Editor** menu option, a list of defined functions appears.

By selecting the highlighted name of a function to be edited, the appropriate function panel filled with the information from the selected function, is displayed

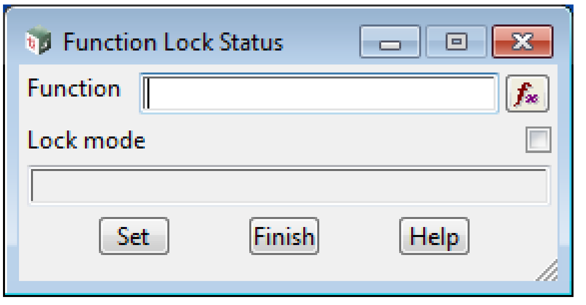
The information in the selected function can then be modified.

Important Note
The defined functions menu also has a **[Same as]** option which is used to select the function to be edited by simply picking **any string** that was created by the function.

28.4.2 Lock

Position of option on menu: **Utilities =>Functions =>Lock**

The function lock option is used to *lock* a function so that it can't be re-calculated.
On selecting the **Lock** option, the **Function Lock Status** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function	input		available functions

name of the function to lock/unlock.

Lock model	tick box		

display and modifies the lock status for the given function.

If ticked, the function is locked an won't re-calculate.

Set	button		

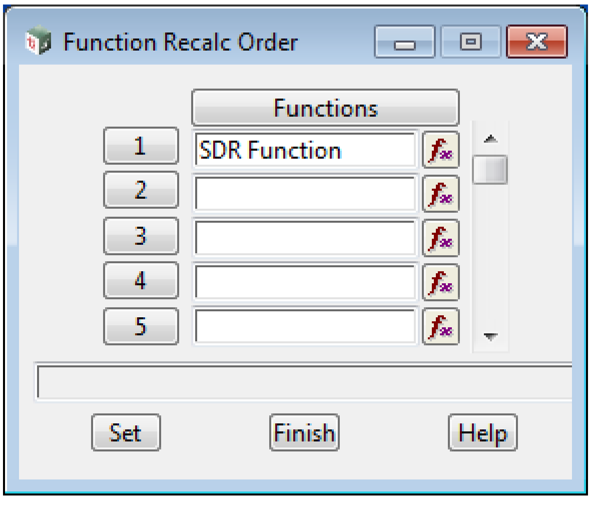
set the lock mode for the selected function.

28.4.3 Order

Position of option on menu: **Utilities =>Functions =>Order**

The default recalculation order for functions is the order in which the functions were created. However, the **Order** option is used to **modify** the function recalc order.

On selecting the **Order** option, the **Function Recalc Order** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Functions	table		available functions

list of functions in their recalculation order. Functions should only appear once in the list.

Set	button		

record the order of functions in the table.

Note - all functions must still exist somewhere in the re-arranged recalc order.

28.4.4 Recalc

Position of option on menu: **Utilities =>Functions =>Recalc**

The **recalc** menu is the same as the menu.

Utilities =>Recalc

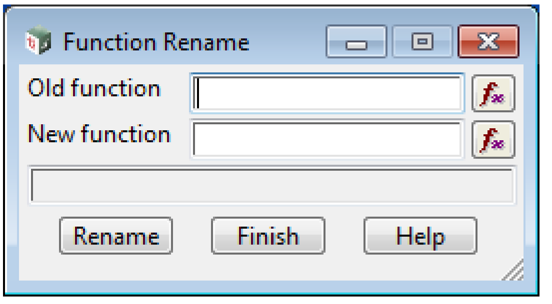
For more information, please go to the section 28.7 Recalc.

28.4.5 Rename

Position of option on menu: **Utilities =>Functions =>Rename**

On selecting the **rename** option, the **function rename** panel is displayed.

This panel can be used to change the names of existing functions



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Old function	input		available functions

name of the function to be renamed.

New function	input

new name for the function

Rename	button

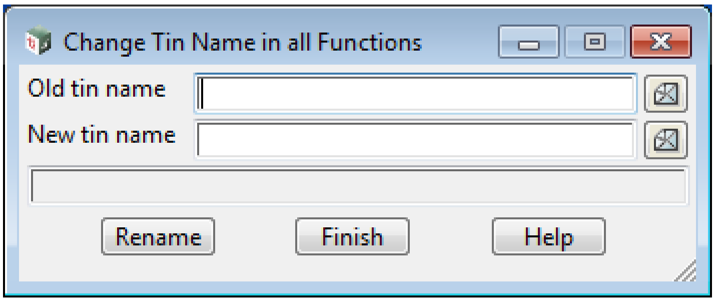
Change the name of the function in the **old function** *field to the name given in the* **new function** *field.*

28.4.6 Change all Tins

Position of option on menu: **Utilities =>Functions =>Change all tins**

Selecting the **Change all tins** option, displays the panel **Change Tin Name in All Functions**.

This panel is used to change the name of a tin in all functions to another tin name.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Old tin name	Tin box		available tins

name of the tin to be renamed.

New tin name	Tin box		available tins

new name for the tin in all functions.

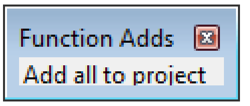
Rename button

Change the name of the tin in the **Old tin name** *field in all functions to the name given in the* **New tin name** *field.*

28.4.7 Add

Position of menu: **Utilities =>Functions =>Add**

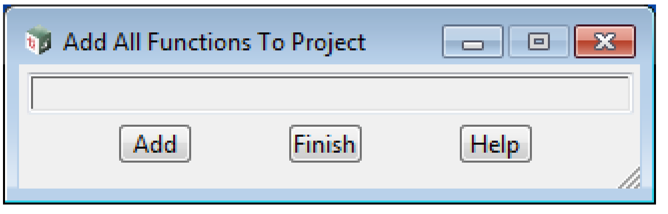
The function **add** option brings up the walk-right menu, **function adds**



Position of option on menu: **Utilities =>Functions =>Add =>Add all to project**

On selecting the **Add all to project** option, the **Add All Functions To Project** panel is displayed.

This panel is used to add all removed functions to the project.

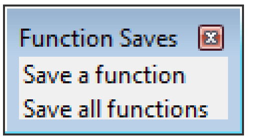


28.4.8 Save

Position of menu: **Utilities =>Functions =>Save**

Functions can be saved on disk so that they can be used for future project work.

The **Function save** walk-right menu is



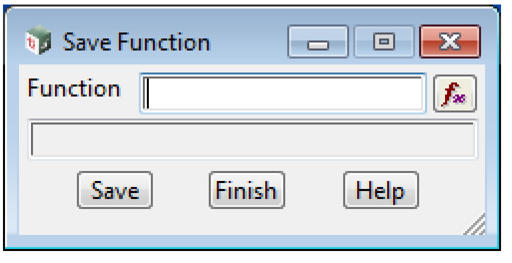
Note - Modified or new functions are automatically saved when a project is saved.

For the option Save a function, go to 28.4.8.1 Save a Function
Save all functions 28.4.8.2 Save All Functions

28.4.8.1 Save a Function

Position of option on menu: **Utilities =>Functions =>Save =>Save a function**

On selecting the **Save a function** option, the **Save Function** panel is displayed.



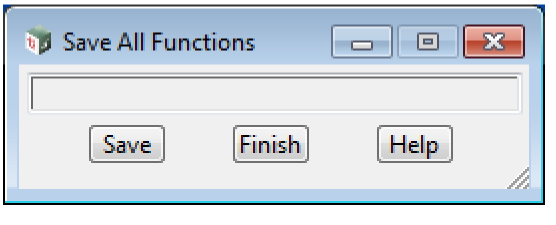
The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function	input		available functions
name of the function to be saved.			
Save	button		
after selecting this button, the function given in the function field will be saved to disk.			

28.4.8.2 Save All Functions

Position of option on menu: **Utilities =>Functions =>Save =>Save all functions**

On selecting the **Save all** option, the **Save All Functions** panel is displayed.

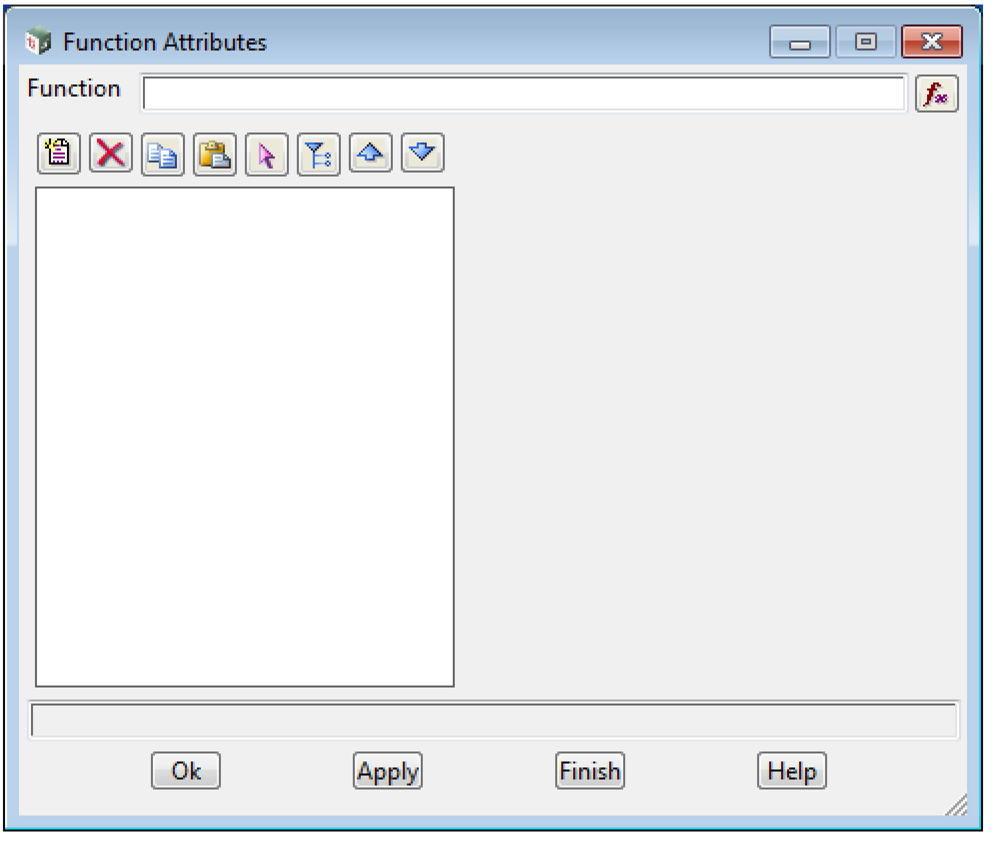


By selecting the **save** button, all functions in the working project will be saved to disk. Unless an error occurs, the panel will be removed after the saving is completed.

28.4.9 Attributes

Position of menu: **Utilities =>Functions =>Attributes**

This section of documentation is a work in progress and will be updated in subsequent releases.



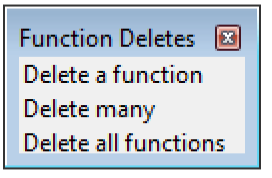
28.4.10 Delete

Position of menu: **Utilities =>Functions =>Delete**

Using the **Delete** option, functions can be deleted from disk so that they no longer can be accessed or take up disk space.

To help protect the user against disasters, when a function is selected for deletion, a **yes-no** pop-up menu is used to confirm that the user did intend deleting the function.

If deletion is confirmed, the selected function is removed from the project and deleted from the disk.
The **Function Delete** walk-right menu is



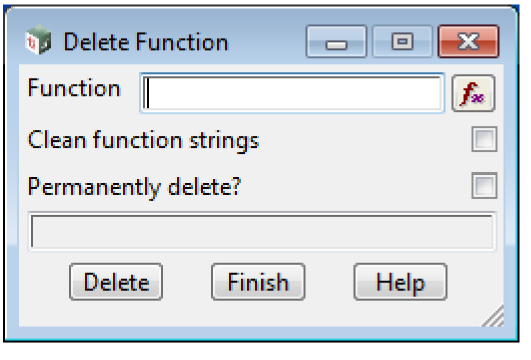
For the option *Delete a function*, go to 28.4.10.1 Delete a Function
Delete many 28.4.10.2 Delete Many Functions
Delete all functions 28.4.10.3 Delete All Functions

28.4.10.1 Delete a Function

Position of option on menu: **Utilities =>Functions =>Delete =>Delete a function**

The **Delete a function** option can be used to delete a function in the working project.

On selecting the **Delete a function** option, the **Delete Function** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function	function box		all project functions

name of the function to be deleted.

Clean function strings	tick box		

if ***ticked****, delete all the strings created by the function*

Delete	button		

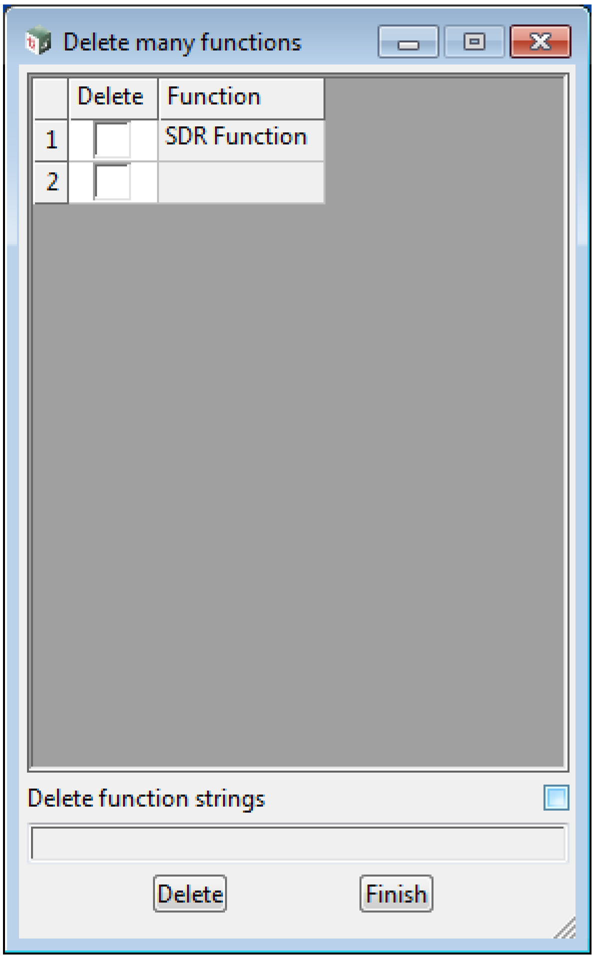
after selecting this button, the function given in the function field will be deleted from the computer disk. A **yes-no** *pop-up is used to confirm that deletion is required.*

28.4.10.2 Delete Many Functions

Position of menu: **Utilities =>Functions =>Delete many**

The **Delete many** option deletes one or more functions in the working project.

Selecting the **Delete many**, displays the **Delete many functions** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Delete	tick box		

*if **ticked**, delete the selected function listed after the tick*

Function	input		all project functions

name of all the functions in the project

Delete function strings	tick box		

*if **ticked**, delete all the strings created by the selected functions*

Delete	button		

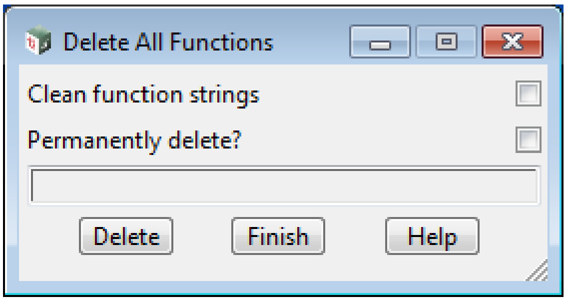
*delete all the **ticked** functions in the function field*

28.4.10.3 Delete All Functions

Position of option on menu: **Utilities =>Functions =>Delete =>Delete all functions**

The **Delete all functions** option will delete all of the functions in the working project.

On selecting the **Delete all functions** option, the **Delete All Functions** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Clean function strings	tick box		

*if **ticked**, delete all the strings created by all the function*

After selecting the **Delete** button, a **yes-no** pop-up is used to confirm that deletion is required. If it is, all functions in the working project will be deleted from disk and unless an error occurs, the panel will be removed.

28.5 Macros

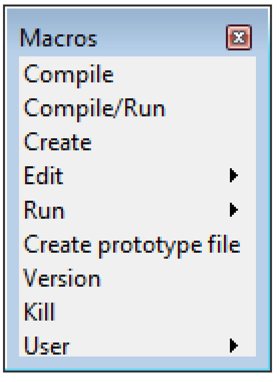
Position of menu: **Utilities =>Macros**

The 12d Solutions Programming Language (12dPL) is a powerful programming language designed to run from within 12d Model. Its main purpose is to allow users to enhance the existing 12d Model package by writing their own programs (12dPLs or macros).

12dPL programs can be compiled from within or outside 12d Model but can only be run from within 12d Model.

The Macros menu has options to compile macros, compile and run or just run macros. A full description of macros is given in the 12d Model Programming Language manual.

The **Macros** walk- right menu is



compile a macro
compile and run a macro
create a macro
edit a macro
run a macro
create file of macro prototypes
version of macro executable
terminates the running of macros

For the option *Compile*, go to	28.5.1 Compile
Compile/run	28.5.2 Compile and Run
Create	28.5.3 Create
Edit	28.5.4 Edit
Run	28.5.5 Run
Create prototypes file	28.5.6 Create Prototype File
Version	28.5.7 Version
Kill	28.5.8 Kill

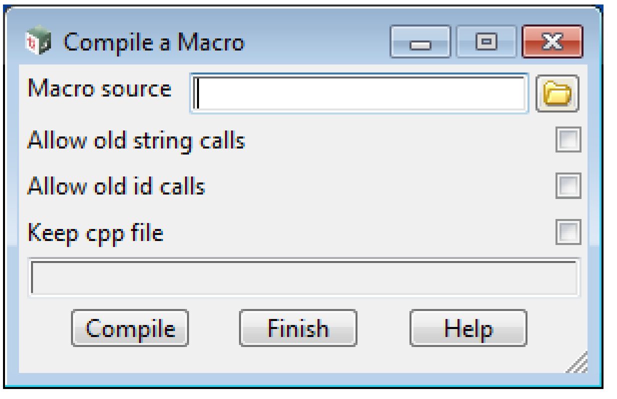
The options will now be described.

28.5.1 Compile

Position of option on menu: **Utilities =>Macros =>Compile**

The **Compile** option is used to compile the 12dPL program source code into a executable program which can then be run from within 12d Model.

On selecting the option, the **Compile a Macro** panel is placed on the screen.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Macro source	input		*.4dm files

the name of the 12dPL program (macro) source file.

Compile	button		

compile the code file given in the **Macro source** *field.*

28.5.2 Compile and Run

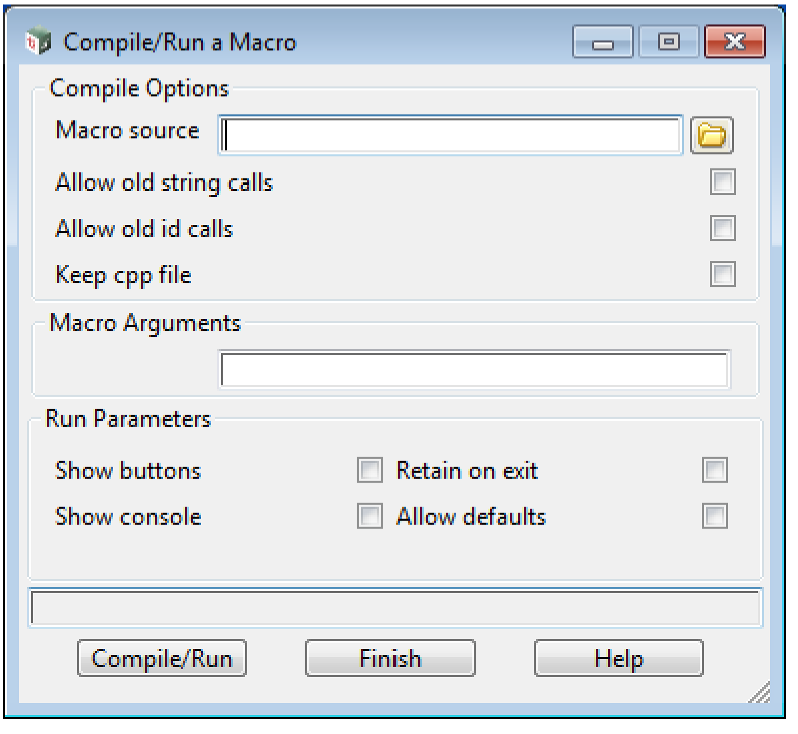
Position of option on menu: **Utilities =>Macros =>Compile/Run**

The **Compile/Run** option compiles the 12dPL program (macro) source code into a executable program, and if there are no errors, runs the program.

When a macro is run, a **macro console** panel is placed on the screen to provide an i/o and message area for the macro. It is possible to bring up the macro console with or without the **restart**, **abort** and **finish** buttons.

When the macro finishes, the macro console can be left on the screen or removed.

Selecting the **Compile/Run** menu option, brings up the **Compile/Run a Macro** panel:



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Macro source	input		*.4dm files

the name of the macro source file.

Macro arguments input

if non-blank, a text line which is passed to the program as an argument line.

Show buttons tick box

if ticked, the macro console panel has **abort**, **restart** *and* **finish** *buttons on it.*

Retain on exit tick box tick

if ticked, the macro console panel remains on the screen after the macro has finished running.

Show console tick box tick

if ticked, the macro console panel is placed on the screen.
if not ticked, the macro console panel is not displayed - used mainly with panels.

Allow defaults tick box

if ticked, the default value for a prompt is displayed in the macro console panel.

Compile/Run button

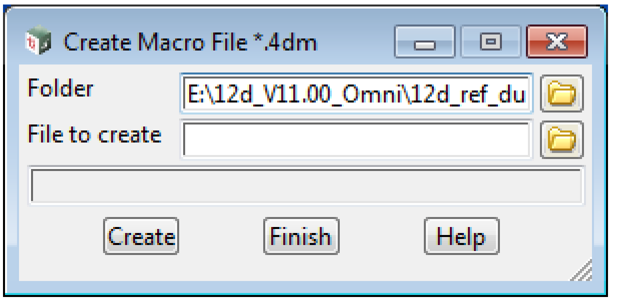
compile the code file given in the **macro source** *field and if the compile is successful, run the program.*

28.5.3 Create

Position of option on menu: **Utilities =>Macros =>Create**

The **Macros=>Create** option is used to create macro files (*.4dm) with the editor pointed to by the EDIT_4D environment variable.

Selecting **Macros=>Create** brings up the **Create Macro File *.4dm** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Folder	input	current folder	

name of the folder to create the .4dm file in.

File to create	input		*.4dm files

name of the file, in **folder***, to create.*

Create	button		

create the file given by the **folder** *and* **file to create** *panel fields. A macro header is set up in the file.*

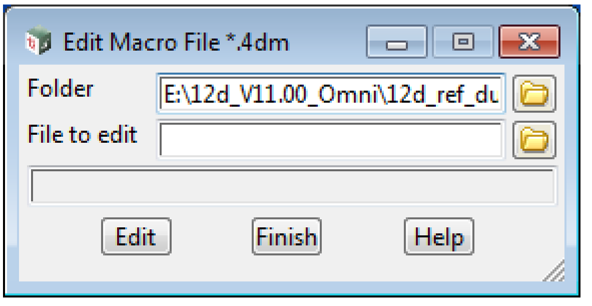
28.5.4 Edit

Position of option on menu: **Utilities =>Macros =>Edit**

The **Macros=>Edit** option is used to edit macro files (*.4dm) with the editor pointed to by the EDIT_4D environment variable.

The **Macros=>Edit** option has two modes of operation - selecting the **macros=>edit** itself, or by activating the **Macros=>Edit** option's walk-right menu, **folder *.4dm**.

Selecting **Macros=>Edit** itself brings up the **Edit Macro File *.4dm** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Folder	input	current folder	

name of the folder for the .4dm file.

File to edit	input		*.4dm files

name of the file, in **folder***, to edit.*

Edit	button		

edit the file given by the **folder** *and* **file to edit** *panel fields. If the file given in the* **file to edit** *field does not exist, then a new file is created which already has each of the macro header set up.*

The **Macros=>Edit** walk-right menu provides a list all the many template files (files ending in .mtf) in the current folder. When a file is selected from the list, it is automatically loaded into the editor.

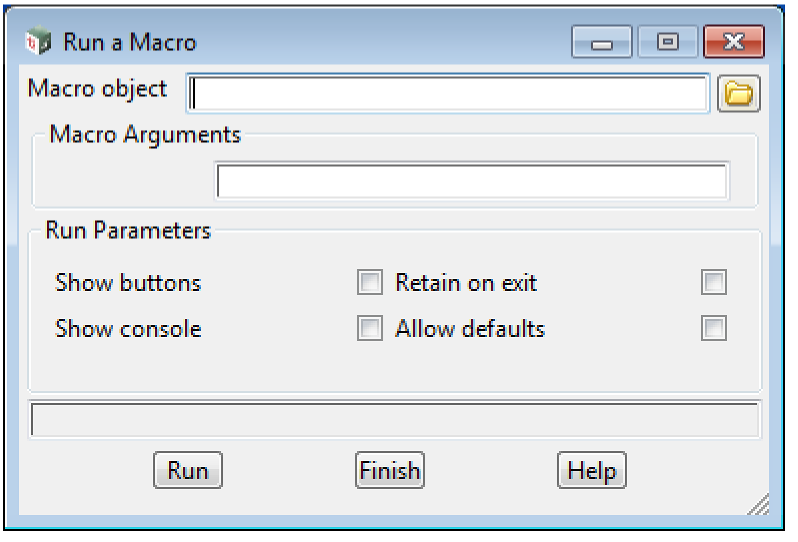
28.5.5 Run

Position of option on menu: **Utilities =>Macros =>Run**

The **Run** option runs an existing macro executable program (produced by the compile process).

Like the **Compile/Run** option, the Run options can run the macro executable with or without buttons on the macro console, and leave or remove the macro console once the macro has been executed.

Selecting the **Run** menu option brings up the **Run a Macro** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Macro object	input		*.4dm files

the name of the macro object or executable program.

Macro arguments	input		

if non-blank, a text line which is passed to the program as an argument line.

Show buttons	tick box		

if ticked, the macro console panel has **abort**, **restart** *and* **finish** *buttons on it.*

Retain on exit	tick box		tick

if ticked, the macro console panel remains on the screen after the macro has finished running.

Show console	tick box	tick	

if ticked, the macro console panel is placed on the screen.
if not ticked, the macro console panel is not displayed - used mainly with panels.

Allow defaults	tick box		

if ticked, the default value for a prompt is displayed in the macro console panel.

Run	button		

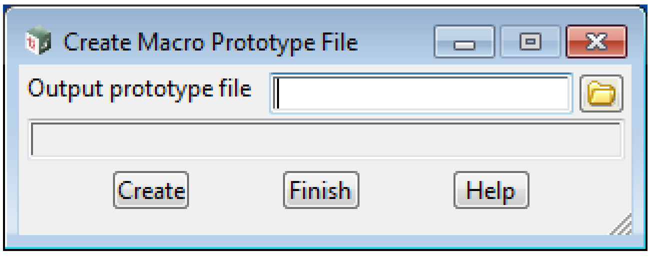
run the executable given in the **macro object** *field.*

28.5.6 Create Prototype File

Position of option on menu: **Utilities =>Macros =>Create prototype file**

The **Create prototype file** option creates a list of the function prototypes for all the ***12dPL*** functions in the ***12d Programming Language*** compiler that is available on your computer.

Selecting the **Create prototypes file** brings up the **Create Macro Prototype File** panel



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Output prototype file	File Box		*.txt files

the name of the file to write the macro function prototypes out to.

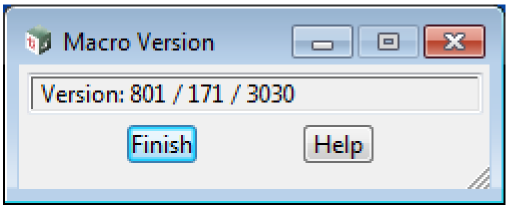
Create	button		

create the file of macro function prototypes.

28.5.7 Version

Position of option on menu: **Utilities =>Macros =>Version**

The **Version** option simply brings up a panel giving information about the macro process.



28.5.8 Kill

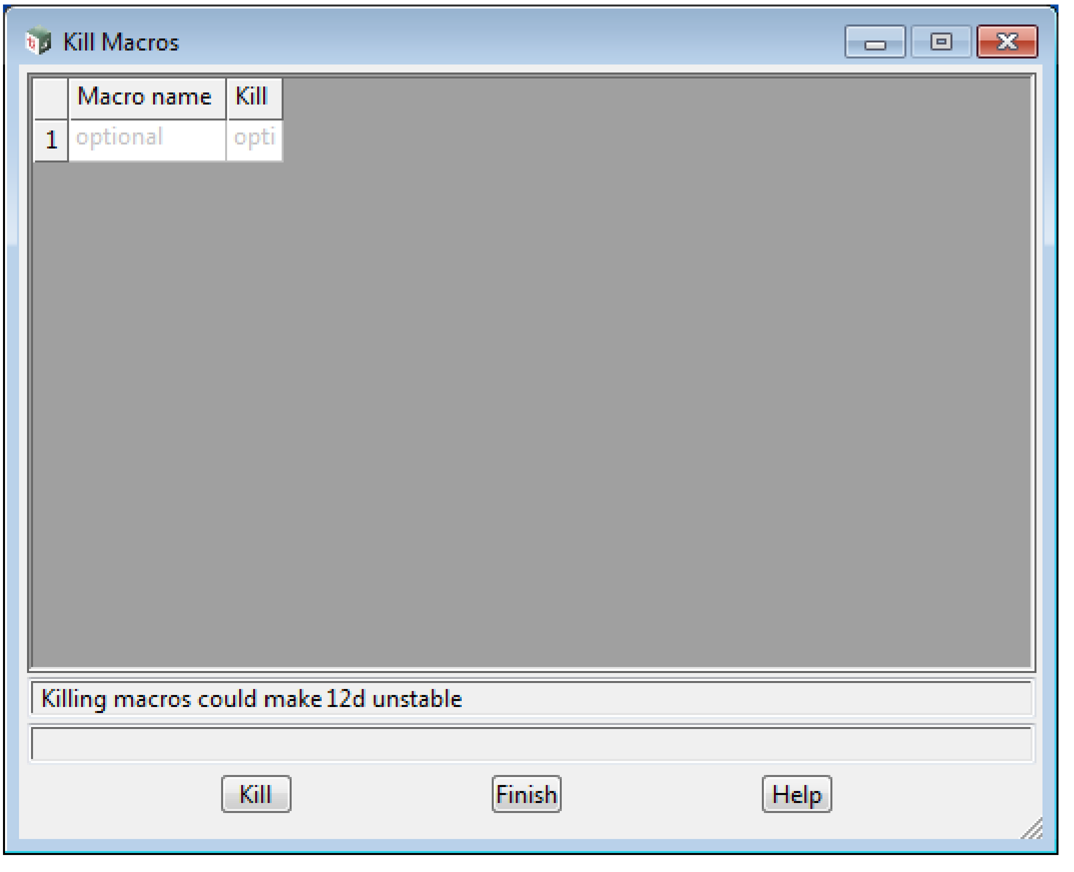
Position of option on menu: **Utilities =>Macros =>Kill**

The **Kill** option is used to terminate selected running macros.

This option is useful in situations such as when the macro writer has omitted a **Finish** button on the macro or has unintentionally made the macro panel invisible.

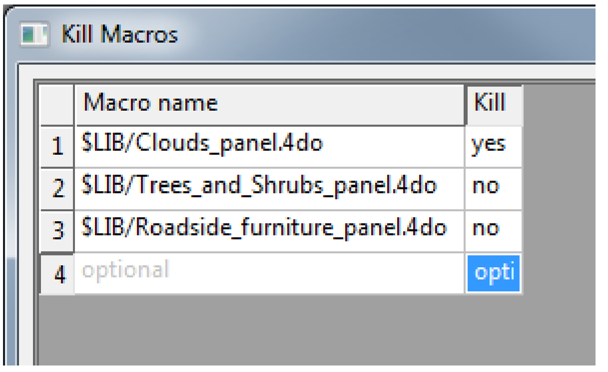
It cannot be used to terminate a macro in an infinite loop because there is no way of getting to the **Kill** option.

On selecting **Kill**, the **Kill Macros** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Macro Grid			
Macro name			
the name of a running macro.			
Kill	choice cell	yes	yes, no
*if **yes**, the macro will be terminated when the **Kill** button is selected.*			



Kill button

*terminates all macros in the grid with **yes** in the* **Kill** *column.*

28.6 Measure

Position of menu: **Utilities =>Measure**

The **Measure** menu contains two options to display the

delta x, delta y, horizontal distance, angle and bearing

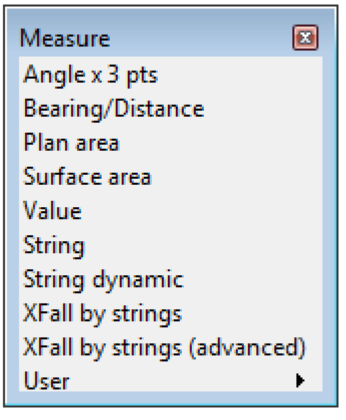
between points, and for a section view, the

delta chainage, delta height, slope distance and % grade

between points are calculated and displayed.

There are also options to calculate the plan area of polygons, the surface area of a tin within a polygon, run measures and dynamic measures and dynamically calculate and display the x-fall between two strings.

The **Measure** walk- right menu is



measure angle by 3 points
distance/bearing with scale option
plan areas
surface area for a tin
any measures
information from a point to a string
information from a position to a string
x-fall between two strings
more information & x-fall between strings

Note

The **Measure** option has another mode of operation. Rather than moving onto the walk-right arrow, if LB is clicked when the **Measure** button is highlighted on the **12d Model=>utilities** menu, the **distance/bearing** panel is displayed on the screen.

For the option *Angle x 3 pts*, go to	28.6.1 Angle by 3 Points
Bearing/distance	28.6.2 Bearing and Distance
Plan area	28.6.3 Plan Area
Surface area	16.11.4 Surface Area
Value	28.6.5 Value
String	28.6.6 String
String dynamic	28.6.7 String Dynamic
Xfall by strings	28.6.8 X Fall by Strings
Xfall by strings (advanced)	28.6.9 X Fall by Strings (Advanced)

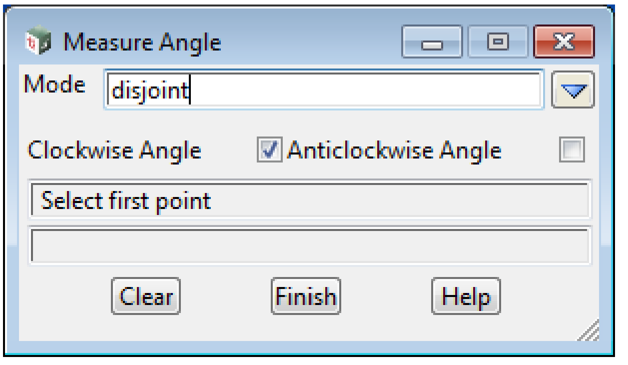
The options in the menu will now be described.

28.6.1 Angle by 3 Points

Position of option on menu: **Utilities =>Measure =>Angle x 3 pts**

The **Angle by 3 pts** option is used to calculate the angle formed by selected three positions. the angle can be expressed as a clockwise or anti-clockwise angle.

On selecting the **Angle x 3 pts** option, the **Measure Angle** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Mode	input	disjoint	disjoint, continuous

In disjoint mode, only the angle between the three selected positions is reported. After the angle is reported, the user can select the 1st point in the sequence.

In continuous mode, after the initial reporting of the angle, the user is prompted for the 3rd point of the new angle to be measured. By default, the previous 2nd point becomes the first point and the previous 3rd point becomes the 2nd point. This allows just one point (the third point) to be selected and the angle reported will change accordingly.

Clockwise angle tick box

if ticked, then the angle is measured in the clockwise (forward) direction.

Anti-clockwise angle tick box

if ticked, then the angle is measured in the anti-clockwise direction.

Clear button

when this button is selected, the sequence is re-initialised.

Note - after two positions are selected, the angle to the cursor position is dynamically calculated and displayed as the cursor moves around the view.

28.6.2 Bearing and Distance

Position of option on menu: **Utilities =>Measure =>Bearing/Distance**

The **Bearing/Distance** option is used to calculate

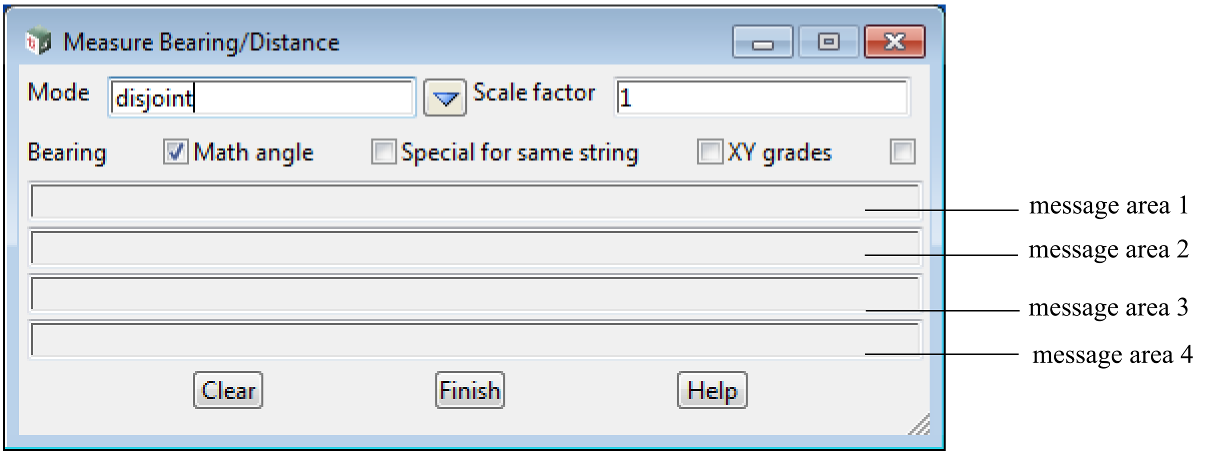
(a) the bearing measured in a clockwise direction between the positive y-axis and the (imaginary) line joining two selected points (when the bearing tick box is selected)

or

the bearing measured in an anticlockwise direction between the positive x-axis and the (imaginary) line joining two selected points (when the **Math angle** tick box is selected)

(b) the distance between two user selected points.

On selecting the **Bearing/Distance** option, the **Measure Bearing/Distance** panel is displayed.



This panel is principally used to display the distances between, and bearing of the line joining, pairs of user selected points.

message area 1 : brg = plane dist = ellipsoid dist =

where brg is the angle measured clockwise from north and is calculated using the coordinate values (Inverse value)

plane dist is the distance calculated using the coordinate values (Inverse value)

ellipsoid dist is the distance calculated by dividing the plane distance by the current scale factor

message area 2: dx = dy = dht = (if applicable)

where dx is the difference in x value between the two coordinates

dy is the difference in y value between the two coordinates

dz is the difference in z value between the two coordinates

If valid height values exist for two consecutive points the grade and slope is also reported

message area 3: grade % = slope =

If the measurement mode is set to continuous, the sum distances will be shown for the route taken

message area 4: Sum plane dist = Sum ellip dist =

where Sum plane dist, is the accumulative length of the route taken calculated using plane distances.

Sum ellip dist, is the Sum plane dist / scale factor.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Mode	input	disjoint	disjoint, continuous

In disjoint mode, only the distance between the two points and the bearing (in degrees, minutes and seconds) of the (imaginary) line connecting the two points are displayed.

In continuous mode, after the initial reporting of the bearing/distance, the user is prompted to select the next point (2nd point). In this case the previously selected second point becomes the first point.

Scale	input/output	1.0	

the scale factor which will be used to convert plane distances to ellipsoid distances.

i.e ellipsoid distance = plane distance / scale factor

Bearing	tick box	ticked	

if ticked, the measured vectors can be displayed as bearings (angles measured from north, clockwise)

Math angle	tick box	un-ticked	

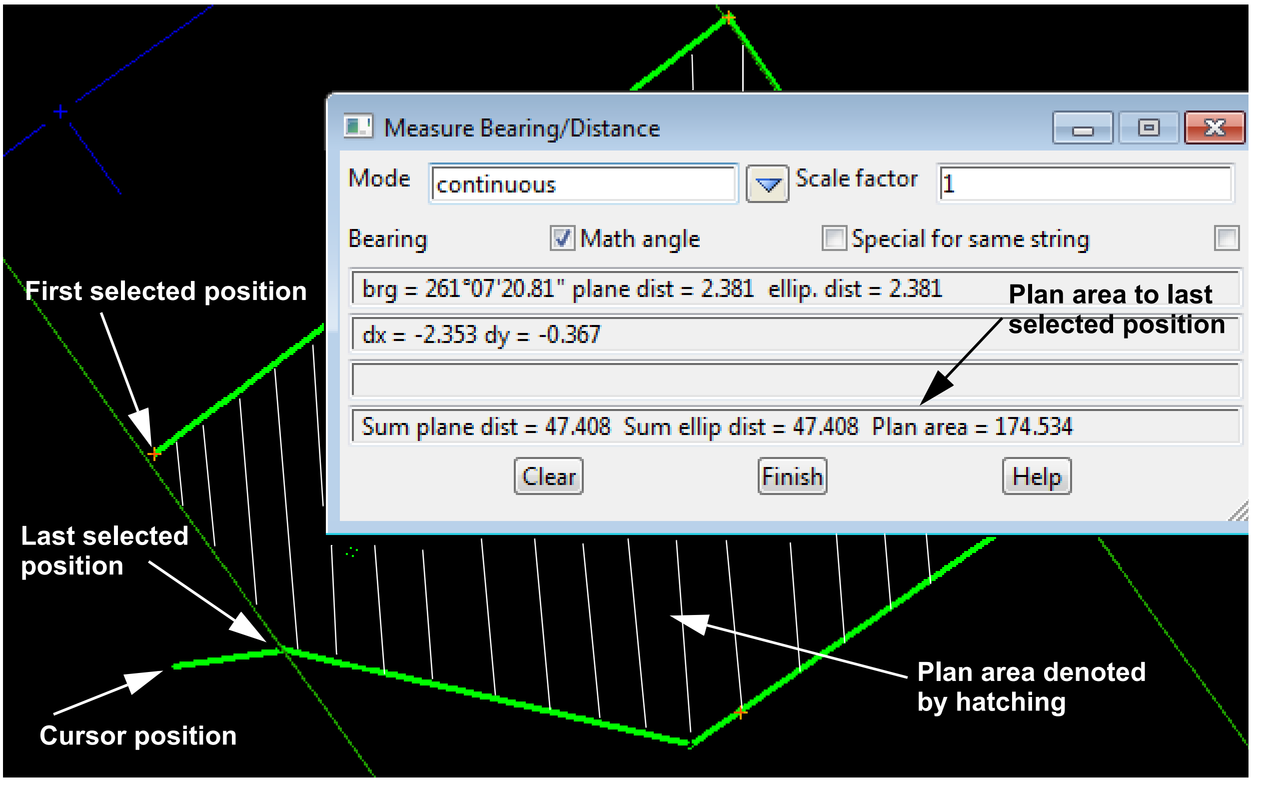
if ticked, the measured vectors can be displayed as a mathematical angle (measured from east in a anti-clockwise direction.

Clear	button		

when this button is selected, the selection sequence is re-initialised.

Note

when in **continuous** mode, each time the next screen position is added to the highlighted figure, the **plan area** is calculated for the closed figure that is constructed by joining the first and last selected screen positions.



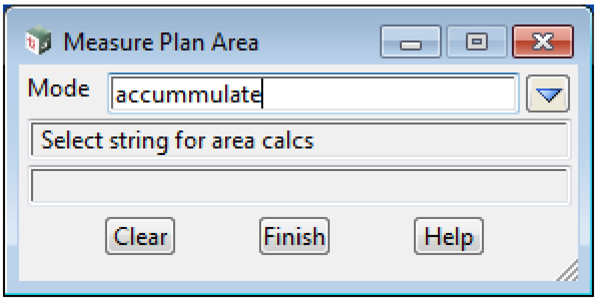
28.6.3 Plan Area

Position of option on menu: **Utilities =>Measure =>Plan area**

The **Plan area** option is used to calculate the plan areas enclosed by strings.

If a string is not closed, the first and last points are considered to be connected to form a polygon.

On selecting the option, the **Measure Plan Area** panel is placed on the screen.



This panel is principally used to display the plan areas of strings selected by the user.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Mode	input	disjoint	disjoint, continuous

*If **disjoint**, only the area of the selected string is displayed. If **continuous**, the total of the areas is also displayed.*

Clear	button

when this button is selected, the total-to-date of the area is reset to zero and the selection sequence begun.

How to Use the Panel and Panel Messages

The measure area cycle consists is as follows:

(a) the string is selected with the mouse.

message area 1	select string for area calcs
message area 2	

(b) repeat step (a) to find more areas and accumulate them. If the accumulator is to be zeroed, simply select the **clear** button again.

The area of the selected string is displayed in message area 2.

If the mode is set to continuous, the accumulated areas (sum) of the selected strings is also displayed.

message area 1	select string for area calcs
message area 2	area = *value* sum = *value*

The cycle can then be repeated for as many strings as needed without leaving the option.

The accumulated total is reset to zero (cleared) by selecting the **clear** button again.

28.6.4 Surface Area

Position of option on menu: **Utilities =>Measure =>Surface area**

The **Surface area** option calculates and reports the surface area of a tin restricted to a user specified polygon.

This option has already been documented as **Tins=>Tin analysis=>Surface area.**

See the section 16.11.4 Surface Area.

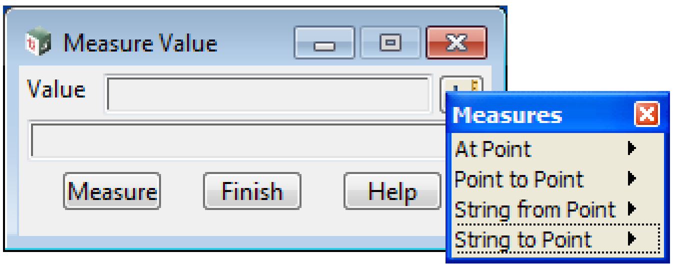
28.6.5 Value

Position of option on menu: **Utilities =>Measure =>Value**

The **Value** option is used to calculate and display any of the measures available in the **Measures** pop-up menu.

The **Measures** are documented under 4.21 Measures.

On selecting the option, the **Measure Value** panel and the **Measures** menu are placed on the screen.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Value	output		

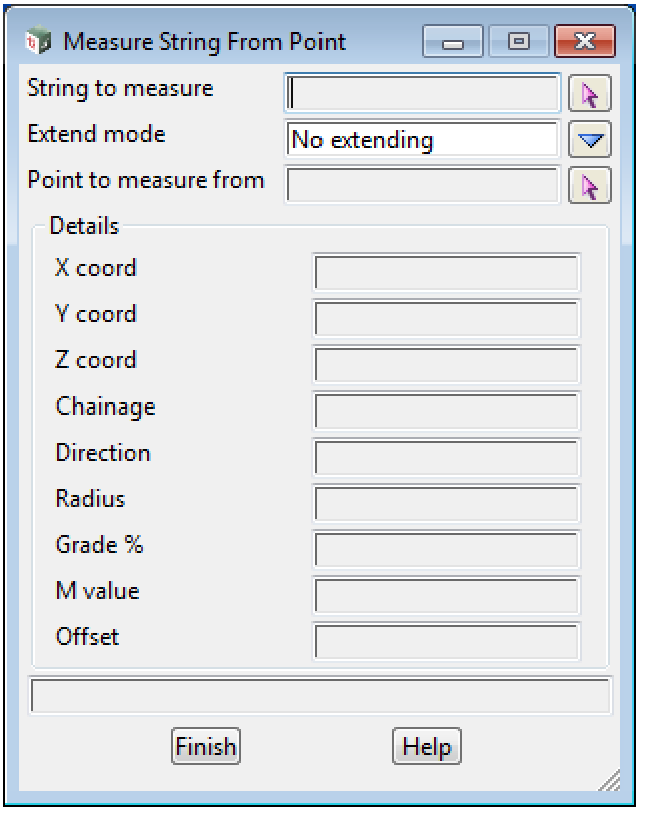
the value calculated by the selected measure option.

28.6.6 String

Position of option on menu: **Utilities =>Measure =>String**

This section of documentation is a work in progress and will be updated in subsequent releases.

On selecting **String**, the **Measure String From Point** panel is placed on the screen.

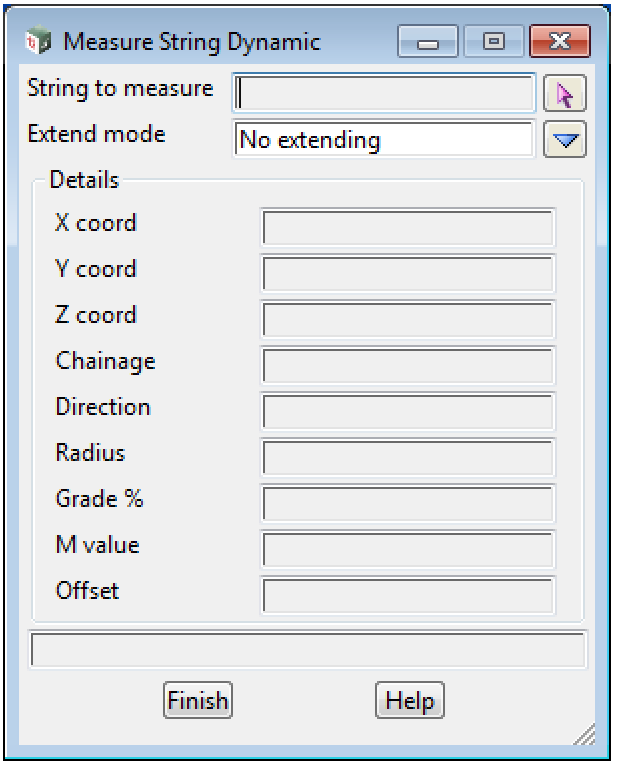


28.6.7 String Dynamic

Position of option on menu: **Utilities =>Measure =>String dynamic**

This section of documentation is a work in progress and will be updated in subsequent releases.

On selecting **String dynamic**, the **Measure String Dynamic** panel is placed on the screen.



28.6.8 X Fall by Strings

Position of option on menu: **Utilities =>Measure =>XFall by strings**

In this option, the user selects a reference string to define what is meant by chainage and right angles, and then selects two strings to calculate the x-fall between.

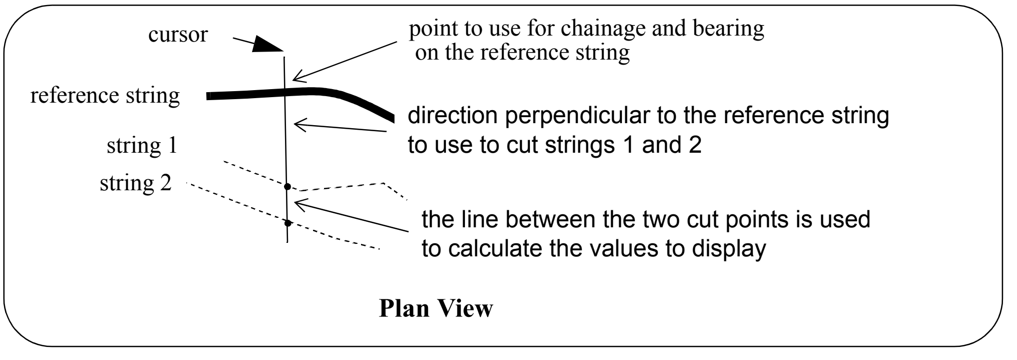
Once the three strings are selected, as the cursor moves around in a plan view, the cursor position is dropped perpendicularly onto the reference string.

At the dropped chainage, a line is taken at right angles to the reference point and extended until it cuts the two selected strings. The xfall, delta width and delta z is then calculated between the cut points on the strings.

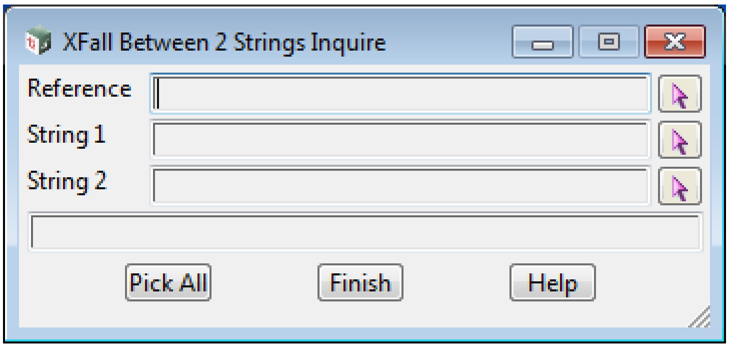
Then the chainage and the xfall are dynamically displayed in the panel message area.

The line joining the two strings is dynamically displayed on the plan view.

The chainage, xfall, delta width and delta z are dynamically displayed in the message area.



On selecting the **Xfall by strings** option, the **XFall Between 2 strings Inquire** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Reference	string-select		

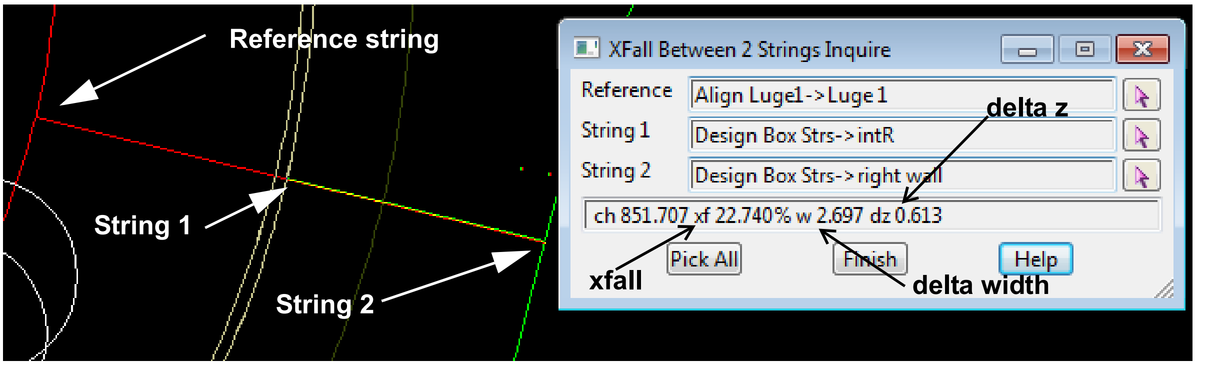
the selected string is used to defined the meaning of chainage and bearing for the inquire. The cursor is dropped perpendicularly onto the reference string and at the dropped chainage, a line is taken at right angles to the reference point and extended until it cuts **string 1** *and* **string 2**. *The xfall, delta width and delta z between the two cut points are then calculate and displayed.*

String 1/2	string-select		

the two strings to calculate x-fall between.

Pick all	button		

if **pick all** *is selected, the user is asked to sequentially select the three strings: reference string, string 1 and string 2. The strings are automatically assigned to the appropriate string-select panel fields.*



28.6.9 X Fall by Strings (Advanced)

Position of option on menu: **Utilities =>Measure =>XFall by strings (advanced)**

In this option, the user selects a reference string to define what is meant by chainage and right angles, and then selects two strings to calculate a variety of values for the reference string and the two strings, and the differences between the values for the two strings.

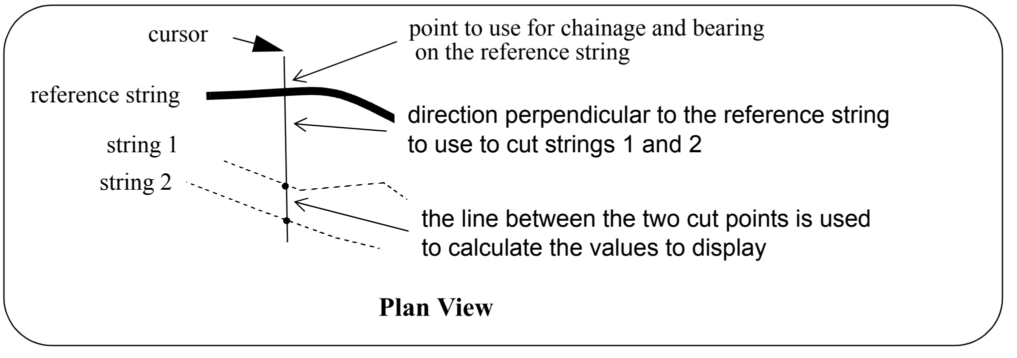
Once the three strings are selected, as the cursor moves around in a plan view, the cursor position is dropped perpendicularly onto the reference string.

At the dropped chainage, a line is taken at right angles to the reference point and extended until it cuts the two selected strings. The xfall is then calculated between the cut points on the strings.

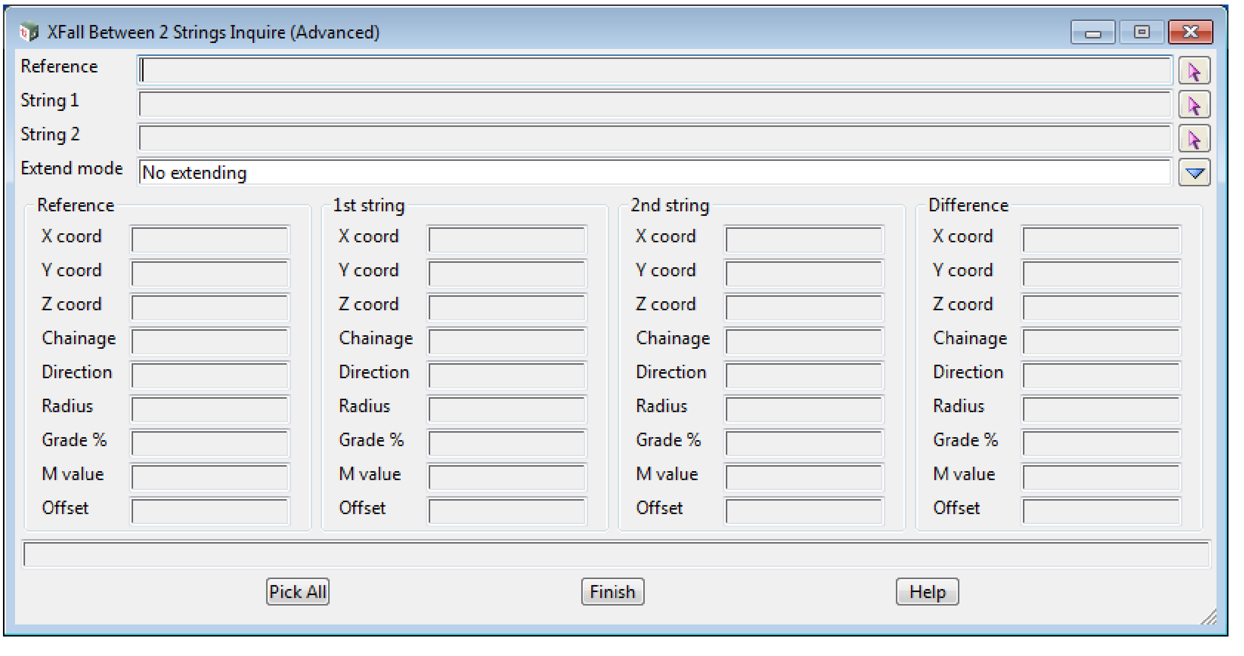
Then the chainage and the xfall are dynamically displayed in the panel message area.

The line joining the two strings is dynamically displayed on the plan view.

Various values for the cut points on the two strings, and the differences of the values between the strings dynamically displayed in the grid.



Plan View



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Reference	string-select		

the selected string is used to defined the meaning of chainage and bearing for the inquire. The cursor is *dropped perpendicularly onto the reference string and at the dropped chainage, a line is taken at right angles to the reference point and extended until it cuts* **string 1** *and* **string 2***. The values are then calculate and displayed for the two cut points and the difference between the values of the cut points.*

String 1/2 string-select

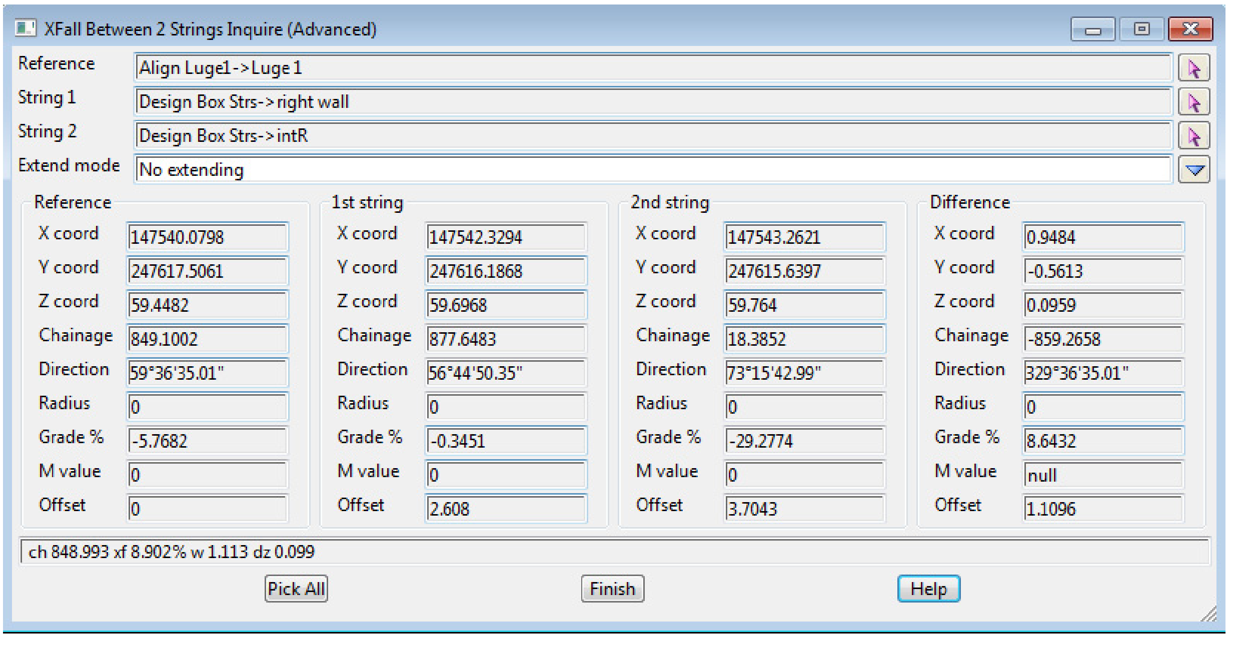
the two strings to calculate value on, and the differences between the values.

Grid output

grid to display the values or the cut point on the two strings and the differences between the values.

Pick all button

For example,



28.7 Recalc

Position of menu: **Utilities =>Recalc**

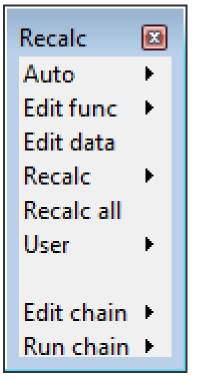
Once a function has been applied, the calculated information is only correct whilst the initial data is not modified in any way. If any modifications are made to the data, then the function must be re-run and all the associated information recalculated and redrawn, and any perspective views associated with the function set with the new eye and target positions, and redrawn.

There are date and time stamps for all string, tin and template information in 12d Model and this enables 12d Model to determine what information has been modified since a function was last run and what re-calculations are required.

The **Recalc** option is designed to **re-run** functions

If LB is clicked whilst the **Recalc** menu option is highlighted, all the template functions with **auto recalc** set **on**, and that have had their initial data modified since their last re-run, will be recalculated and all appropriate information updated. The data will be redrawn on any views that is was on before the recalc.

The **Recalc** walk-right menu has four options:



For the option *Auto*, go to 28.7.1 Auto
Edit func 28.7.2 Editor
Edit data 28.7.3 Edit Data from String
Recalc 28.7.4 Recalc
Recalc all 28.7.5 Recalc All

Edit chain 28.7.6 Edit Chain
Run chain 28.7.7 Run Chain

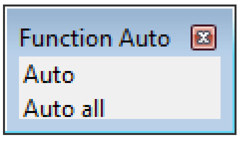
28.7.1 Auto

Position of option on menu: **Utilities =>Recalc =>Auto**

The **Auto** option allows the user to specify whether a function is automatically recalced if another function, tin, template or string that it depends upon, is modified or recalced.

Individual functions or all functions can have auto-recalc mode turned on or off and the default for a new function is **off**.

The **Function Auto** menu is

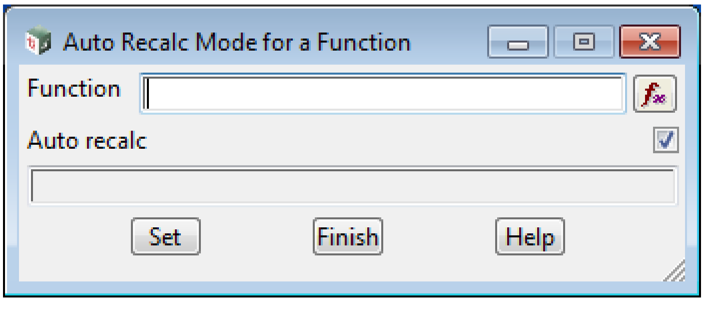


For the option Auto, go to 28.7.1.1 Auto
Auto all 28.7.1.2 Auto all

28.7.1.1 Auto

Position of option on menu: **Utilities =>Recalc =>Auto =>Auto**

Selecting **Auto** displays the **Auto Recalc Mode for a Function** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Function	input		available functions

name of the function to be have auto recalc mode set on/off.

Auto recalc	tick box	tick	

If the tick is on, the specified function will have its ***auto recalc*** *value changed to on.*

If the tick is off, the specified function will have its ***auto recalc*** *value changed to off.*

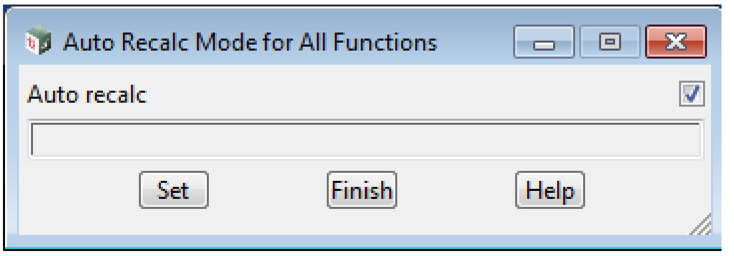
Set	button		

Set the **auto recalc** *mode for the selected function.*

28.7.1.2 Auto all

Position of option on menu: **Utilities =>Recalc =>Auto =>Auto all**

Selecting **Auto all** displays the **Auto Recalc Mode For All Functions** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Auto recalc	tick box	tick	

*If the tick is on, all functions within the project will have their **auto recalc** value changed to on.*

*If the tick is off, all functions within the project will have their **auto recalc** value changed to off.*

Note: the tick is not a global setting within 12d. The use of the option changes every function within the project.

Set	button		

Set the **auto recalc** *mode for all functions.*

28.7.2 Editor

Position of option on menu: **Utilities =>Recalc =>Editor**

The **Recalc=>Editor** option is used to edit functions.

The **Recalc=>Editor** walk-right menu provides a list all the defined functions in the project.

When a function is selected from the list, it is automatically loaded into the appropriate function panel for the selected function, and displayed

The information in the selected function can then be modified.

Important Note
The defined functions menu also has a **[Same as]** option which is used to select the function to be edited by simply picking **any string** that was created by the function. This is documented in the next section 28.7.2.1 Edit Function from String.

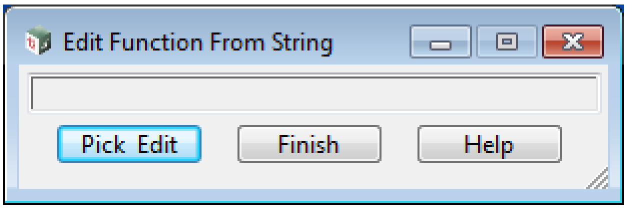
28.7.2.1 Edit Function from String

Position of option on menu: **Utilities =>Recalc =>Editor =>[Same as]**

Position of option on menu: **Utilities =>Recalc =>Recalc =>[Same as]**

The **Edit Function from String** option allows the user to start up the *Function editor* by simply selecting a string that was created by the function.

On selecting the option, the **Edit Function From String** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick Edit	button		

select data that was created by a function and the appropriate function editor is then opened.

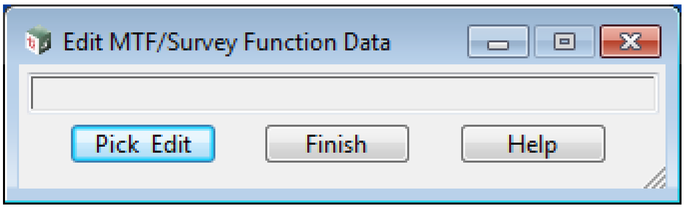
28.7.3 Edit Data from String

Position of option on menu: **Utilities =>Recalc =>Edit data**

The **Edit data** option allows the user to start up the *MTF editor* by simply selecting a string that was created using an Apply Templates function (and hence an MTF file). The MTF file used in the *Apply Templates* function is automatically loaded into MTF editor (go to 20.4.1.1 MTF Edit).

Similarly the **Edit data** option also allows the user to start up the *Survey Field Data editor* by simply selecting a string that was created using a *Survey Data Reduction* function. All the data produced by the *Survey Data Reduction* function is automatically loaded into the *Survey Field Data* editor (go to the section 17.7.5 Survey Field Data Editor).

On selecting the **Edit data** option, the **Edit MTF/Survey Function Data** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Pick Edit	button		

select data that was created by applying an MTF file or a Survey Data Reduction function. The appropriate MTF editor or Survey Field Data editor that created the selected data, is then opened.

For information on the *Survey Field Data Editor*, go to the section 17.7.5 Survey Field Data Editor.

For information on the *MTF Editor*, go to 20.4.1.1 MTF Edit.

28.7.4 Recalc

Position of menu: **Utilities =>Recalc =>Recalc**

The **Recalc** option is used to recalculate a user selected function, and all functions that depend on the function that have auto recalc set on.

The **Recalc** walk-right menu lists all the existing functions and by selecting a function name, the function will be forced to re-run and all appropriate information and views for that function will be updated.

The existing functions menu also has a **[Changed]** option which recalcs all changed functions.

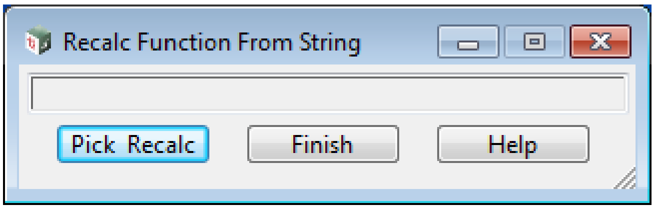
The existing functions menu has a **[Same as]** option which is used to select the function to be recalced by simply picking **any string** that was created by the function. This is documented in the next section 28.7.4.1 Recalc Function from String

28.7.4.1 Recalc Function from String

Position of option on menu: **Utilities =>Recalc =>Recalc =>[Same as]**

The **Recalc Function from String** option allows the user to recalc a function by simply selecting a string that was created by the function.

On selecting the option, the **Recalc Function From String** panel is displayed.



The fields and buttons used in this panel have the following functions.

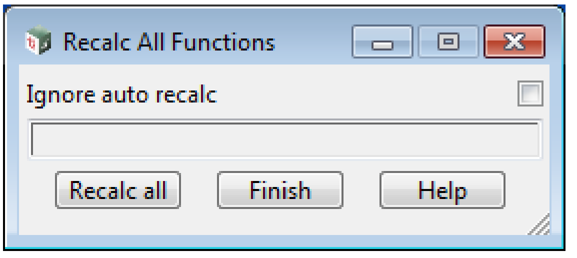
Field Description	Type	Defaults	Pop-Up
Pick Edit	button		

select data that was created by a function and the appropriate function will be recalced.

28.7.5 Recalc All

Position of option on menu: **Utilities =>Recalc =>Recalc all**

The **Recalc all** option re-runs all functions regardless of their date and time stamps.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Ignore auto recalc	tick box		

*if **not ticked**, all functions that have the auto-recalc flag set on are re-calculated.*

*If **ticked**, the auto-recalc flag is ignored and all functions are re-calculated.*

28.7.6 Edit Chain

Position of option on menu: **Utilities =>Recalc =>Edit chain**

The **Recalc=>Edit chain** option is used to edit chains.

The **Recalc=>Edit chain** walk-right menu provides a list all the defined chains in the project.

When a chain is selected from the list, it is automatically loaded into the **Edit/Create Chain** panel ready for editing.

28.7.7 Run Chain

Position of option on menu: **Utilities =>Recalc =>Run chain**

The **Recalc=>Run chain** option is used to run chains.

The **Recalc=>Run chain** walk-right menu provides a list all the defined chains in the project.

When a chain is selected from the list, it is automatically run.

28.8 Fence

Position of menu: **Utilities =>Fence**

It is often necessary to divide strings into those parts that are within a certain region and those parts that are outside the region. In 12d Model, the **Fence** option is the method of achieving this result.

Given a string to be used as a fence, the 12d Model fencing options will process strings in a model/view against the fence and break the strings into the parts inside the fence string and those parts outside the fence string.

Any 2d, 3d or interface line-strings that cut the fence polygon will have points inserted at the intersections with the fence polygon. The z-value at the intersection point will be interpolated from the adjacent points in the line-string.

Point strings do not have any extra points added to them.

Restrictions

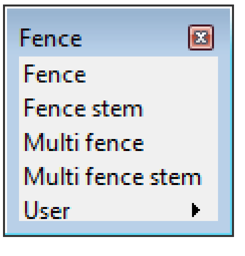
- only 2d, 3d and interface strings can be split using the fence option.
- 4d strings are treated as though they were point strings.
- arcs are considered to be **inside** the fence if the minimum bounding rectangles of the fence and arc overlap.
- alignment, pipeline, drainage, sewer strings are not processed by the fence option

Notes

1. if the fence string is not closed, the first and last points are joined together to form a closed polygon as the fence.
2. a string may be broken into a number of pieces if it wanders in and out of the fence string.
3. the fence string is not processed against itself even if it is in the model or view being fenced.
4. unpredictable results will occur if any strings being fenced lie on top of the fence. If problems occur, parallel the fence in or out by a millimetre.

The **fence** options allows the user to choose a single string to be used as the polygon fence or a number of strings to be used as polygons and process the data against all the strings in one operation.

The **fence** walk-right menu is



For the option *Fence*, go to 28.8.1 Fence
Fence stem 28.8.2 Fence Stem
Multi fence 28.8.3 Multi Fence
Multi fence stem 28.8.4 Multi Fence Stem

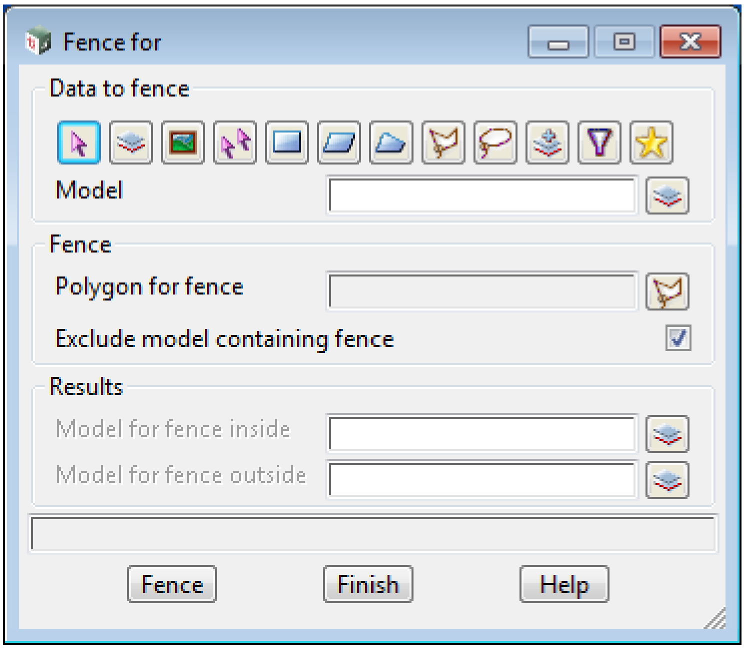
28.8.1 Fence

Position of option on menu: **Utilities =>Fence =>Fence**

In this option, the user selects a single string to be used as a polygon fence. If the string is not closed, the first and last points are joined together to form a polygon.

The **Fence** option will process a selected string or all the strings in a model/view/string against this fence and break the strings into the parts inside the fence string and those parts outside the fence string.

On selecting the **Fence** option and then the appropriate **Data Source** in the panel, the **Fence for** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Data source type

data selection type - for a full description go to 4.19.3 Data Source.

Data source

data source for strings to fence.

Polygon for fence string-select

the selected string will be used as the polygon to act as a fence to separate data into an inside and an outside.

Exclude model containing fence tick box ticked

if ***ticked****, no data in the model containing the fence string will be fenced.*
if ***not ticked****, all the appropriate data in the model containing the fence string will be fenced.*

Model for fence inside input available models

if ***non-blank****, name of the model to contain the data inside of the fence string.*
If ***blank****, then no fence inside will be calculated.*

Model for fence outside input available models

if ***non-blank****, name of the model to contain the data outside of the fence string.*
If ***blank****, then no fence outside will be calculated.*

Fence button

on selecting **fence***, the appropriate strings in the data source will be divided by the fence string (selected by the* **poly** *button) into the parts of the strings inside the fence string and the parts of the string outside the fence string. Any line-strings that cut the fence string will have a point inserted at the intersection with the fence string and the z-value for this point interpolated from the points on either side of it in the line-string.*

How to Use the Panel and Panel Messages

(a) Select a fence polygon string by choosing **Polygon to fence**

(b) Fence processing begins on selecting the **Fence** button.

Progress messages - sent to the panel message area
calculating inside-outside
calculating inside
calculating outside
fencing *string name*

Completion message - sent to the panel message area
finished fence

<esc> can be used to abort the fence option.

28.8.2 Fence Stem

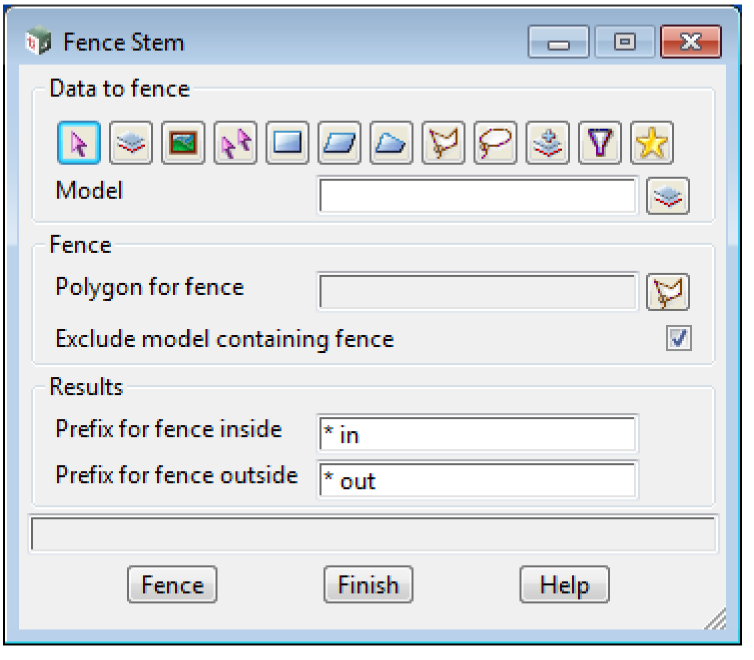
Position of option on menu: **Utilities =>Fence =>Fence stem**

As for the **Fence** option, the user selects a single string to be used as a polygon fence. If the string is not closed, the first and last points are joined together to form a polygon.

The **Fence Stem** option will process a selected string or all the strings in a model/view against this fence and break the strings into the parts inside the fence string and those parts outside the fence string. The inside and outside will then be put into models of the same name as the original model modified by a prefix and/or postfix.

This means that unlike the **Fence** option, when fencing a view with the **Fence Stem** option, the inside/outside for each model on the view goes to distinct models using the prefix/postfix for the inside/outside model names.

On selecting the **Fence Stem** option and then the appropriate **Data Source** in the panel, the **Fence Stem** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source

data source for strings to fence.

Prefix for fence inside	input		available models

*if **non-blank**, prefix*postfix to adjust the data model name to be the model to contain the data inside of the fence string.*

*If **blank**, then no fence inside will be calculated.*

Prefix for fence outside	input		available models

*if **non-blank**, prefix*postfix to adjust the data model name to be the model to contain the data outside of the fence string.*

*If **blank**, then no fence outside will be calculated.*

Polygon for fence string-select

the selected string will be used as the polygon to act as a fence to separate data into an inside and an outside.

Exclude model containing fence tick box tick

*if **ticked**, no data in the model containing the fence string will be fenced.*
*if **not ticked**, all the appropriate data in the model containing the fence string will be fenced.*

Fence button

on selecting **Fence***, the appropriate strings in the data source will be divided by the fence string into the parts of the strings inside the fence string and the parts of the string outside the fence string. Any line-strings that cut the fence string will have a point inserted at the intersection with the fence string and the z-value for this point interpolated from the points on either side of it in the line-string.*

<esc> can be used to abort the fence option.

28.8.3 Multi Fence

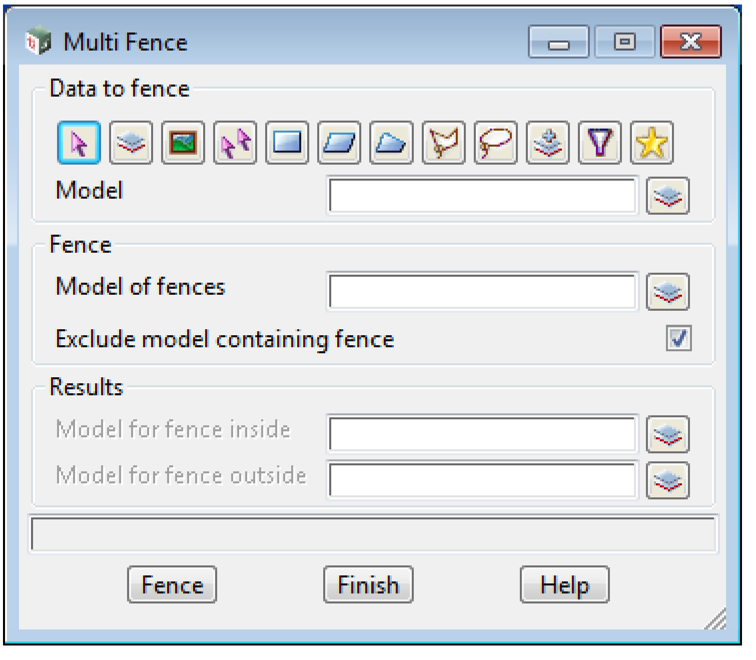
Position of option on menu: **Utilities =>Fence =>Multi fence**

It is often necessary to fence data against more than one polygon.

In the **Multi fence** option, the user provides a model containing all the polygons to be used as fences. The **Multi fence** option will process the selected string or all the strings in a model/view against all the polygons in the polygon model.

The data will be clipped and the **outside** is considered to be those bits are **outside all** of the polygons and the inside is the remainder, *i.e.* the bits **inside any one** of the polygon.

On selecting the **Multi fence** option and then the appropriate **Data Source** in the panel, the **Multi Fence String/Model/View** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source

data source for strings to fence.

Model of fences	input		available models

name of the model that contains the polygons to be used as fences.

Model for fence inside	input		available models

name of the model to contain the data that is inside any one of the fences. If this field is blank, then no fence inside will be calculated.

Model for fence outside	input		available models

name of the model to contain the data outside all of the fences. If this field is blank, then no fence outside will be calculated.

Exclude model containing fence	tick box	tick	

if ***ticked****, no data in the model containing the fence string will be fenced.*
if ***not ticked****, all the appropriate data in the model containing the fence string will be fenced.*

Fence button

the appropriate strings in the data source will be divided by the fence polygons (from the model of fences) into the parts of the strings inside any one fence and the parts of the string outside all of the fences. Any line-strings that cut the fences will have a point inserted at the intersection with each fence and the z-value for this point interpolated from the points on either side of it in the line-string.

<Esc> can be used to abort the multi-fence option.

28.8.4 Multi Fence Stem

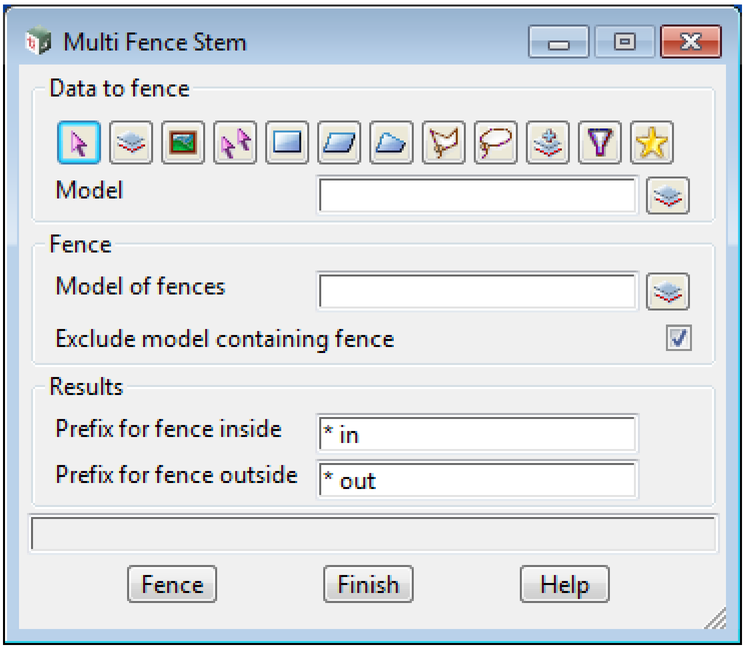
Position of option on menu: **Utilities =>Fence =>Multi fence stem**

It is often necessary to fence data against more than one polygon.

In the **Multi fence stem** option, the user provides a model containing all the polygons to be used as fences. The **Multi fence stem** option will process selected strings or all the strings in a model/view against all the polygons in the polygon model.

The data will be clipped and the **outside** is considered to be those bits are **outside all** of the polygons and the **inside** is the remainder, *i.e.* the bits **inside any one** of the polygon.

On selecting the **Multi fence stem** option and then the appropriate **Data Source** in the panel, the **Multi Fence Stem String/Model/View** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source

data source for strings to fence.

Model of fences	input		available models

name of the model that contains the polygons to be used as fences.

Prefix for fence inside	input		available models

*if **non-blank**, prefix*postfix to adjust the data model name to be the model to contain the data inside of the fence string.*
*If **blank**, then no fence inside will be calculated.*

Prefix for fence outside	input		available models

*if **non-blank**, prefix*postfix to adjust the data model name to be the model to contain the data outside of the fence string.*
*If **blank**, then no fence outside will be calculated.*

Exclude model containing fence tick box tick

*if **ticked**, no data in the model containing the fence string will be fenced.*
*if **not ticked**, all the appropriate data in the model containing the fence string will be fenced.*

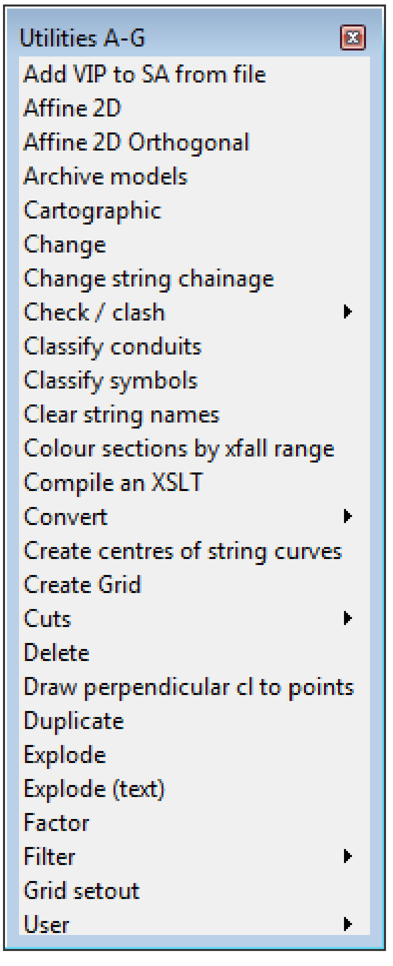
Fence button

the appropriate strings in the data source will be divided by the fence polygons (from the model of fences) into the parts of the strings inside any one fence and the parts of the string outside all of the fences. Any line-strings that cut the fences will have a point inserted at the intersection with each fence and the z-value for this point interpolated from the points on either side of it in the line-string.

<Esc> can be used to abort the multi-fence option.

28.9 Utilities A-G

Position on menu: **Utilities =>A-G**



add a vertical intersection point to a super alignment from a file
perform 2D affine transformation
perform 2D affine transformation with no skewing of axes
save all models matching a specified model mask as 12da
perform cartographic projections
change string colour
set the start chainage of selected strings
check breaklines, strings, 3d clashes, polygons
set attributes on conduit strings

remove string names
colour sections using a x-fall colour file
compile an XSLT file
convert 2d to 3d, 3d to 4d etc
create centres of curves
create a world grid
calculate sections by cuts through strings
delete strings
draw perpendicular from selected points to centreline
duplicate strings
explode strings into lines and arcs
explode text into lines and arcs
multiply the x,y,z values by factors
data filters
create a setout Grid
User Utilities A-G menu

For *Add VIP to SA from file* go to	28.9.2 Add VIP to SA From File
Affine 2D	28.9.3 2D Affine
Affine 2D Orthogonal	28.9.4 2D Affine - Orthogonal
Archive models	28.9.5 Archive Models
Cartographic	28.9.6 Cartographic Projections
Change	28.9.7 Change
Change string chainage	28.9.8 Change String Chainage
Check/clash	28.9.9 Check/Clash
Classify conduits	28.9.10 Classify Conduits
Classify symbols	
Clear string names	28.9.11 Clear String Names
Colour X-Sections by Xfall range	20.14.1 Colour Sections by XFall Range
Convert	28.9.12 Convert
Create centres of string curves	17.18.1 Create Centre Points for Curves of Strings
Create Grid	24.6 Create World Grid
Cuts	28.9.13 Cuts
Delete	28.9.14 Delete
Draw perpendicular LC to points	28.9.15 Draw a Perpendicular from Centreline to a Point

Duplicate	28.9.16 Duplicate
Explode	28.9.17 Explode
Explode (text)	28.9.18 Explode Text
Factor	28.9.19 Factor
Filter	28.9.20 Filter
Grid setout	28.9.21 Grid setout

For general information on ***Affine*** and ***Helmert*** Transformations, go to 28.9.1 Coordinate Transformations - Helmert and Affine

28.9.1 Coordinate Transformations - Helmert and Affine

A ***2D Helmert*** transformation is a two dimensional linear transformation consisting of a scaling, a rotation (of both the x-axis and y-axis about the vertical axis) and a 2D- translation (shift) of data. Hence there are four parameters to be specified.

A ***3D Helmert*** transformation is a three dimensional linear transformation consisting of a scaling, three rotations (about the 3 axes), and a 3D-translation (shift) of data. Hence there are 7 parameters to be specified.

A ***2D affine transformation*** is a two dimensional linear transformation consisting of a 2D-translation (shift) of the data, a scaling of the data in two different directions and rotations of the x-axis (about the vertical axis) and of the y-axis (about the vertical axis). Hence there are six parameters to be specified. Unlike the Helmert transformations, rectangles can be skewed by an affine transformation.

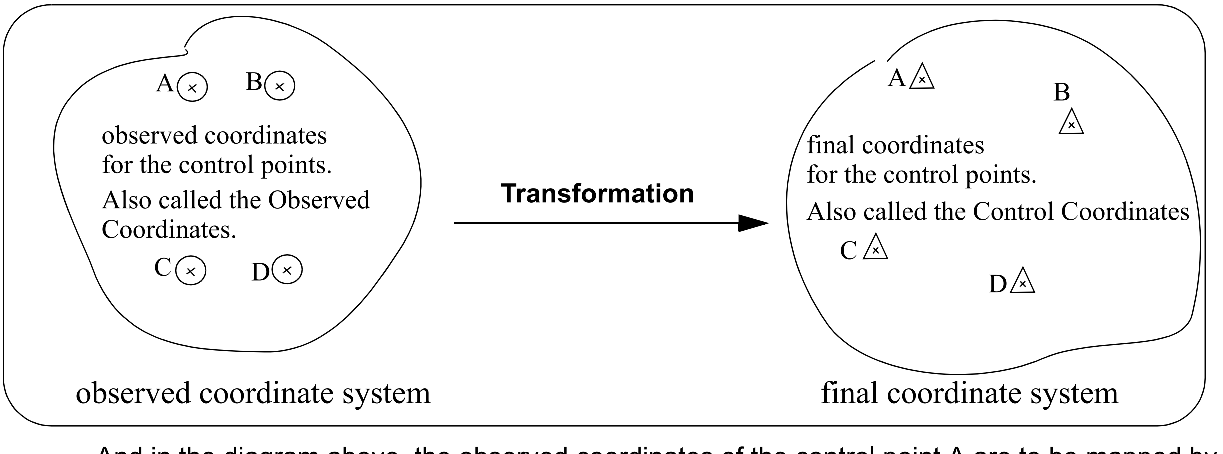
A ***2D orthogonal affine transformation*** is a two dimensional linear transformation consisting of a 2D-translation (shift) of the data, a scaling of the data in two different directions and the same rotation (of both the x-axis and y-axis like the Helmert). Hence there are five parameters to be specified. Unlike the standard affine transformations, rectangles are not skewed by the orthogonal affine transformation.

These types of transformation are often used in civil projects for converting data between a local coordinate system (observed coordinates) and another coordinate system.

The most frequent method of defining all these transformations is not by explicitly giving the transformation parameters but by having both the observed **and** the final coordinates for three or more points (known as the **control points**) and then trying to calculate the transformation parameters from these sets of coordinates.

If the transformation parameters can be successfully calculated from the observed-final coordinate pairs for the control points, then the transformation parameters are then be used to convert the coordinates of any other object in the observed coordinate system to the new final coordinates.

So for this method, the first step is to select **control points** for which the coordinates are known in the final coordinate system and in the local (observed) coordinate system.



And in the diagram above, the observed coordinates of the control point A are to be mapped by the transformation to the coordinates of the control point A in the final coordinate system. And similarly for the points B, C and D.

Note that A is the same point on the ground - it just has a different coordinate in a different coordinate system and hence will appear in a different place in a diagram for each coordinate system.

Usually each control point is selected because

(a) its final coordinates are already known

(b) there is a marker on the ground indicating the exact position of the control point

and

(c) the control point is the area that is being surveyed (observed).

The control point is then surveyed to get its observed coordinates.

Because the final coordinates of the control points are known in advance, the final coordinates of the control point are referred to as the **Control Coordinates**, or **Control Easting** and **Control Northing** and the control point coordinates often referred to as the **Control Point**. The surveyed coordinates of the control point are called as the **Observed Coordinates** or **Observed Easting** and **Observed Northing** (of the control point).

IMPORTANT NOTE - the control points must be selected so that they are spread throughout the area, and the control points must not all be on the one straight line.

The number of parameters to be determined for the transformation ranges from three in a fixed scale 2D Helmert up to seven for a 3D Helmert so there must be enough control points to provide sufficient independent equations to be able to solve for the parameters. Also all survey measurements have errors in them and it is best to have more than the minimum number of required control points to help control errors.

However when there are four or more control-observed point pairs, the system will be over determined. That is, by taking different subsets of the control-observed point pairs, different transformation parameters will result.

In this situation, ***12d Model*** uses **Least Squares** to calculate the transformation parameters from the over determined systems.

Once the transformation parameters have been calculated by Least Squares, for each of the control points the transformation can be applied to the observed coordinates of the control point. The difference between the Control Coordinates and the transformed Observed Coordinates for a control point is called the **residuals** for the transformation.

The ***residuals*** for the least squares solution gives a measure of how good the Least Squares solution was, and for checking purposes, the residuals are displayed in the transformation panel for each of the control-observed point pairs.

As an example of the meaning of the transformation parameters, we'll look at the 2D Helmert transformation.

The 2D Helmert has four parameters:

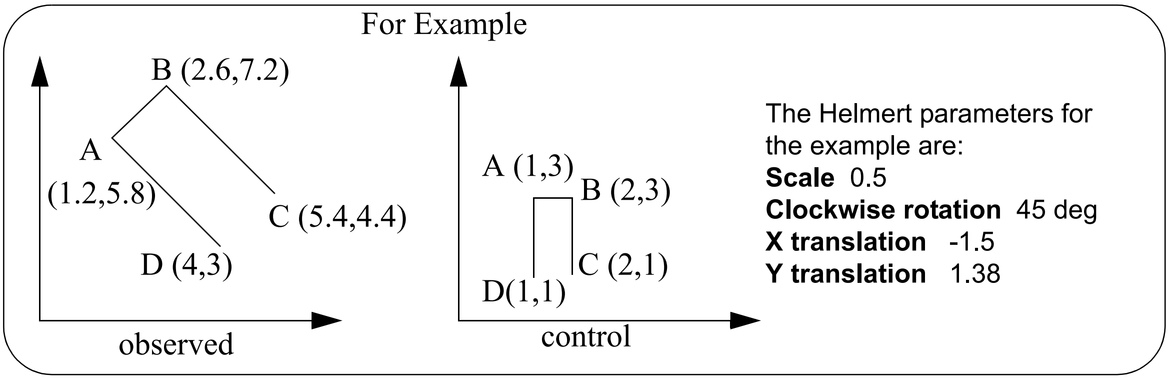
(a) the scale factor for the existing coordinates with respect to the new coordinates

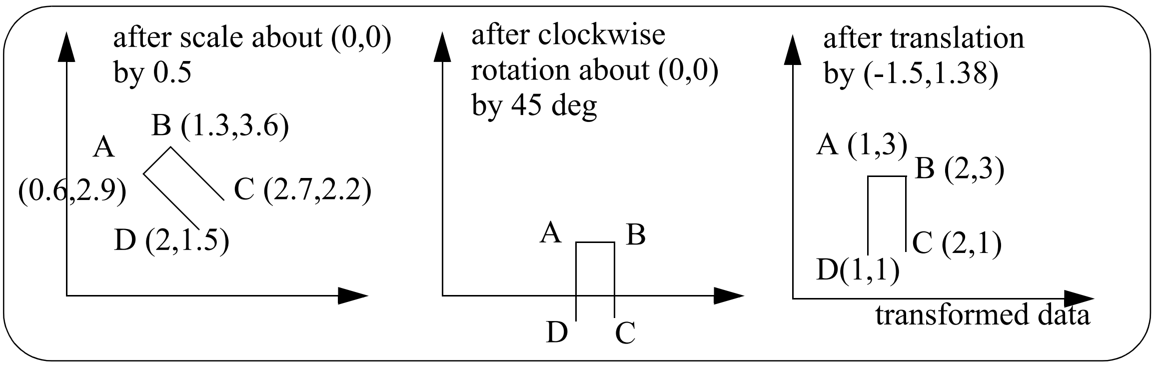
(b) the (z axis) rotation of the existing coordinates to align it with the new coordinates axis

(c) the two (x and y) translations of the existing coordinates after the scalings and rotations have been applied

and the operations to be performed on observed coordinates are:

1. Scale the observed coordinates about (0,0) by the *Scale* factor
2. Then rotate the resulting coordinates about (0,0) in a clockwise direction by the *Clockwise rotation* degrees.
3. Then translate the resulting coordinates by ((X translation, Y translation).



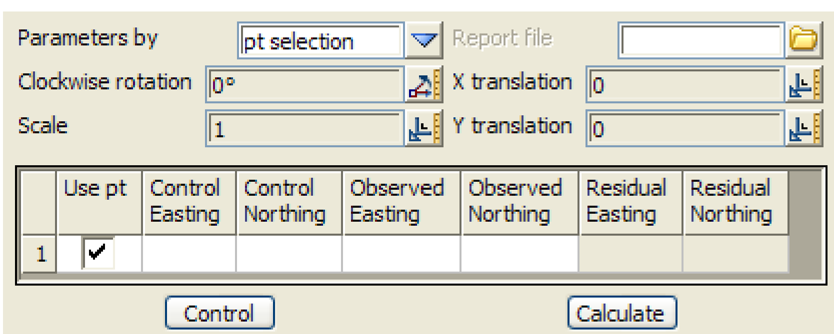


For information about how to select control and observed points in the Affine and Helmert panels, go to the next section 28.9.1.1 Selecting Control and Observed Points for the Helmert and Affine Transformations

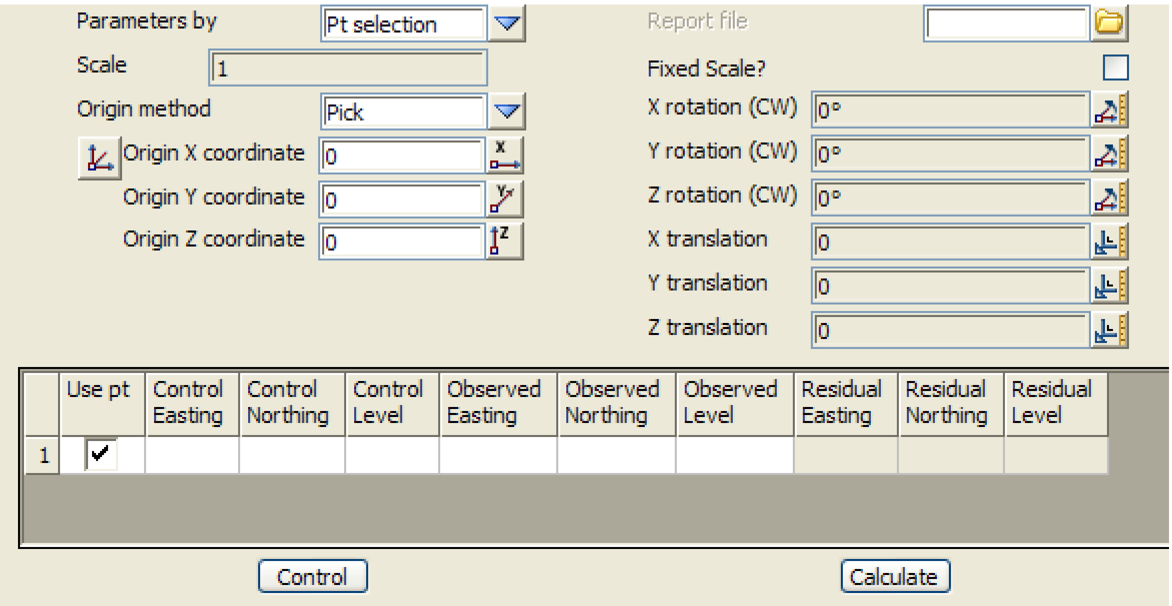
28.9.1.1 Selecting Control and Observed Points for the Helmert and Affine Transformations

For the **Affine** and **Helmert** transformations, the parameters can be either typed in or calculated from control-observed point pairs using the method of least squares.

If the **Parameters by** choice box is set to ***Pt Selection*** then control-observed point pairs are selected by the user and the pairs are written to the grids on the panels.



Grid for 2D transformations



Grid for 3D transformations

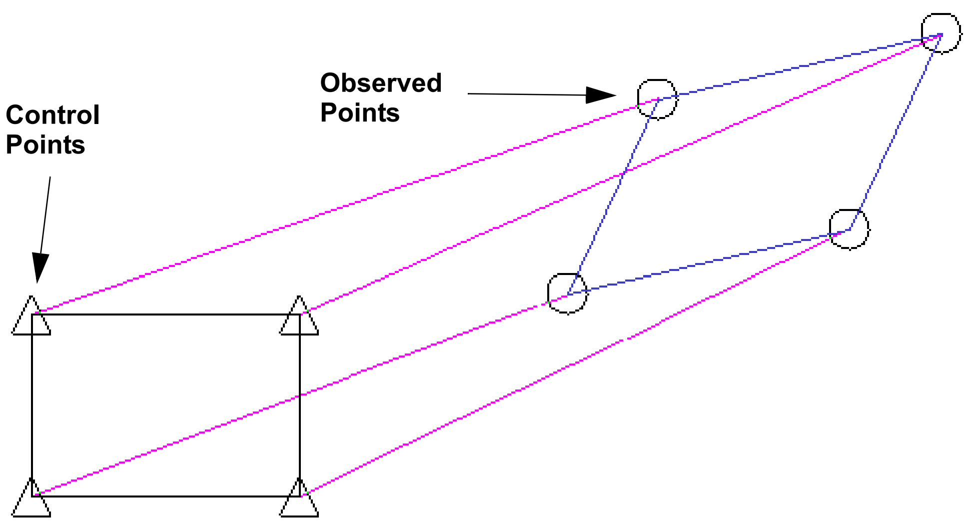
If the **Pt selection** method is chosen, the selection of points is started by selecting the **Control** button.

Note - the default is to select the ***Control Point*** *(the coordinates of the control point in the transformed coordinate system)* and then its associated ***Observed Point*** *(the coordinates of the control point in the observed system)*.

Neither point is accepted until the second point is selected.

However the ***order*** of selection can be ***reversed*** by setting the environment variable **PICK_ORDER_OBSERVED_FIRST_4D** (see 7.6.3 env.4d). The order can then be ***Observed Point first*** and then its associated ***control point with Control Coordinates***.

To help reduce errors, after a control point - observed point pair is selected, the position of the Control Point is shown as a triangle and the position of the associated Observed Point as a circle with a line drawn between the control-observed point pair for clarity



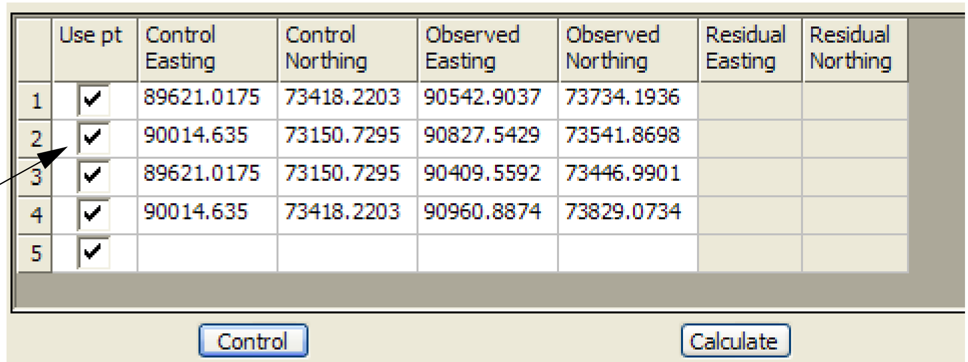
After selecting the **Control** button, the selection starts with the Control (Observed) Point, accepting that point then selecting the corresponding Observed (Control) Point.

After selecting the first pair of points, the second pair is then selected and so on.

The picking is terminated by clicking **RB** and selecting **Cancel** when the next control (observed) point is to be selected.

The coordinates of the Control Points and Observed Points are added to the grid on the selection of two valid points. The **Use pt** field is ticked by default.

Use point column



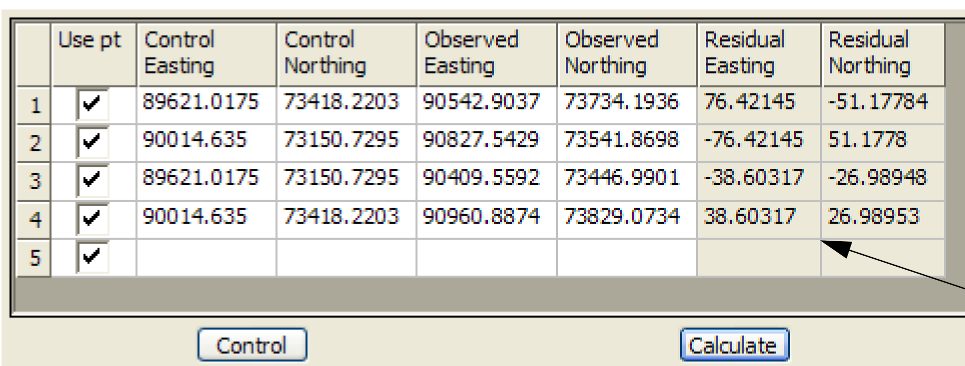
	Use pt	Control Easting	Control Northing	Observed Easting	Observed Northing	Residual Easting	Residual Northing
1	✔	89621.0175	73418.2203	90542.9037	73734.1936		
2	✔	90014.635	73150.7295	90827.5429	73541.8698		
3	✔	89621.0175	73150.7295	90409.5592	73446.9901		
4	✔	90014.635	73418.2203	90960.8874	73829.0734		
5	✔						

After the selection process is terminated, additional control-observed point pairs can be selected at any time by simply clicking on the **Control** button again. The new control-observed point pairs are added after the existing control-observed point pairs.

If ***Use pt*** is set to tick, the corresponding control-observed point pair is used in the calculations. If ***Use pt*** is not ticked, the corresponding control-observed point pair is ignored and not used.

The **Calculate** button calculates the transformation parameters based on all the control-observed point paris in the grid with the **Use pt** tick box ticked on.

The residuals are placed into the residual columns of the grid and the calculated parameters are placed into the parameters section of the panel.



	Use pt	Control Easting	Control Northing	Observed Easting	Observed Northing	Residual Easting	Residual Northing
1	✔	89621.0175	73418.2203	90542.9037	73734.1936	76.42145	-51.17784
2	✔	90014.635	73150.7295	90827.5429	73541.8698	-76.42145	51.1778
3	✔	89621.0175	73150.7295	90409.5592	73446.9901	-38.60317	-26.98948
4	✔	90014.635	73418.2203	90960.8874	73829.0734	38.60317	26.98953
5	✔						

Residuals columns

If the **Parameters by** choice box is set to **Direct entry**, the transformation parameters are manually entered into the panel and the grid section is not used.

For information on how the **Affine Transformation works with Rasters,** go to the section 28.9.1.2 Affine and Helmert Transformations Applied to Rasters

28.9.1.2 Affine and Helmert Transformations Applied to Rasters

12d Rasters (including ECW rasters) may have transforms applied with any transformation panel with the following constraints:

1. "Replace existing data" is the only output option supported.
2. The set of parameters supplied for the transformation, when applied to the four corners of the

original raster, results in a rectangle.

3. The raster cannot be selected as a string to be transformed, it should be placed in a model on its own and the model selected as the **'Data to change'**.

The raster images themselves will not be altered, only the model information which maps them into world coordinates is changed.

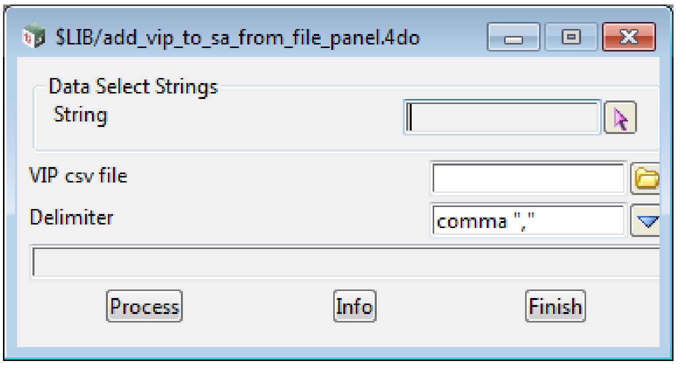
For the Affine panel, this means that different x and y scaling factors may be used, however the two axis rotations must be identical.

28.9.2 Add VIP to SA From File

Position of option on menu: **Utilities =>A-G =>Add VIP to SA from file**

This section of documentation is a work in progress and will be updated in subsequent releases.

On selecting **Add VIP to SA from file**, the **Add VIP to SA from file** panel is displayed.



28.9.3 2D Affine

Position of option on menu: **Utilities =>A-G =>Affine**

An affine transformation is a two dimensional linear transformation consisting of a 2D-translation (shift) of the data, a scaling of the data in two perpendicular directions and a rotation of the x-axis and the y-axis. Hence there are six parameters to be specified. Unlike the Helmert transformations, rectangles can be skewed by an affine transformation.

So the affine parameters are

(a) the x-scale factor for the observed co-ordinates with respect to the control coordinates

(b) the y-scale factor for the observed co-ordinates with respect to the control coordinates

(c) the rotation of the x-axis of the observed co-ordinates to align it with the control x co-ordinates axis

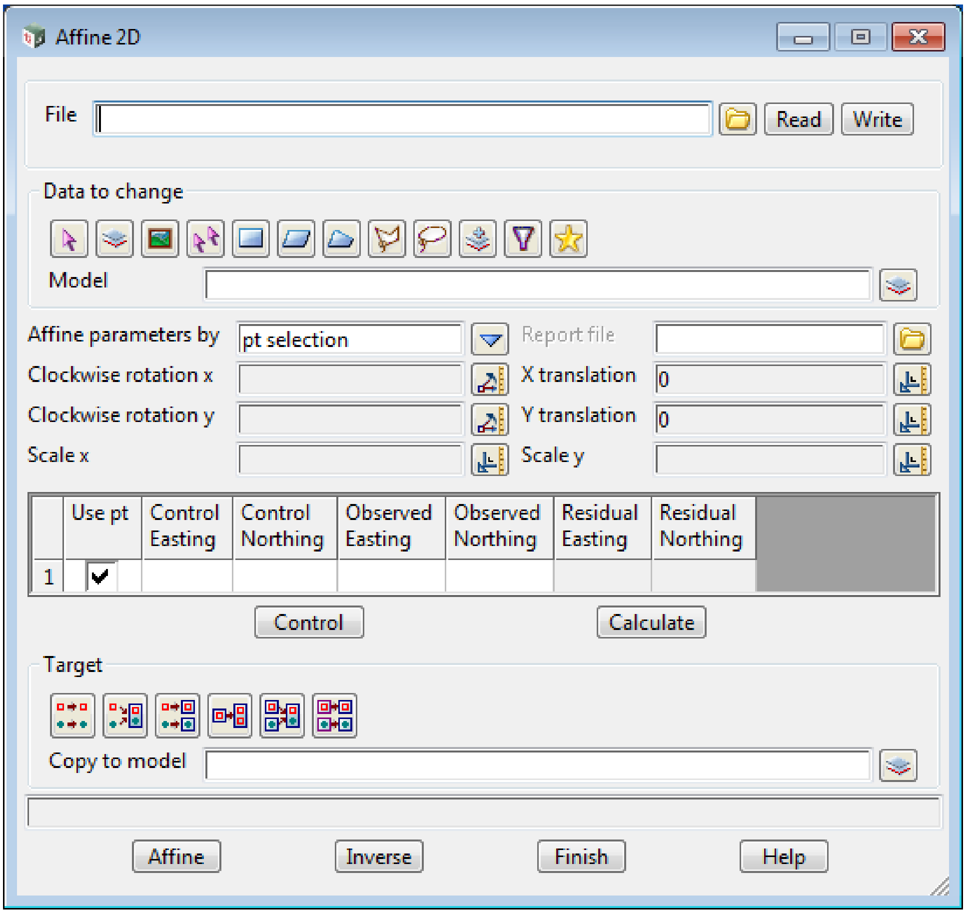
(d) the rotation of the y-axis of the observed co-ordinates to align it with the control y co-ordinates axis

(e) the x and y translation of the existing co-ordinates after the scalings and rotations have been applied

For information on the affine transformation, please go to the section 28.9.1 Coordinate Transformations - Helmert and Affine

For information on how the **Affine Transformation works with Rasters,** go to the section 28.9.1.2 Affine and Helmert Transformations Applied to Rasters

On selecting **Affine 2D**, the **Affine 2d** panel is displayed.



If the **Parameters by** choice box is set to ***Pt Selection*** then control-observed point pairs are selected by the user and the pairs are written to the grid on the panel. See the section 28.9.1.1 Selecting Control and Observed Points for the Helmert and Affine Transformations for information on picking control and observed points.

If the **Parameters by** choice box is set to **Direct entry**, the transformation parameters are manually entered into the panel and the grid section is not used.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
File	file box		*.aaf files

a filename can be specified for reading or writing a file to restore the panel contents for re-use.

Read button

if a valid file exists, the file contents can be loaded into the panel.

Write button

if a valid name is specified, the user can write the input data to a file.

Data to change

Data source type Model

data selection type - for a full description go to 4.19.3 Data Source.

Data source input

source of data to be processed.

Affine parameters by	choice	Pt selection	Pt selection Direct entry

the parameter entry method.

***Pt selection** - the control points and their matching observed points are selected by the user after clicking the control button.*

***Direct entry**- the transformation parameter fields are used to type in the parameter data.*

Report file *.rpt files

if not blank, write a report on the Affine transformation.

Rotation x angle box

the rotation of the x-axes of the existing points with respect to the transformed x-axis.

Rotation y angle box

the rotation of the y-axes of the existing points with respect to the transformed y-axis.

Scale x input/output

the x-scale factor for the existing coordinates with respect to the transformed coordinates

Scale y input/output

the y-scale factor for the existing coordinates with respect to the transformed coordinates

X Translation input/output

the x translation of the existing coordinates with respect to the transformed coordinates

Y Translation input/output

the y translation of the existing coordinates with respect to the transformed coordinates

Control button

starts the selection process for choosing more control points. The process is terminated when the calculate button is used.

Calculate button

calculate the Affine parameters from the selected points shown in the grid.

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info input

extra information required for the target.

Affine button

apply the affine transformation to the data specified in the source box, and put it into the appropriate target area.

28.9.4 2D Affine - Orthogonal

Position of option on menu: **Utilities =>A-G =>Affine 2D orthogonal**

A ***2D orthogonal affine transformation*** is a two dimensional linear transformation consisting of a 2D-translation (shift) of the data, a scaling of the data in two different directions and the same rotation of both the x-axis and y-axis (about the z axis, like the Helmert). Hence there are five parameters to be specified. Unlike the standard affine transformations, rectangles can not be skewed by the orthogonal affine transformation.

Hence the orthogonal affine parameters are

(a) the x-scale factor for the observed co-ordinates with respect to the control coordinates

(b) the y-scale factor for the observed co-ordinates with respect to the control coordinates

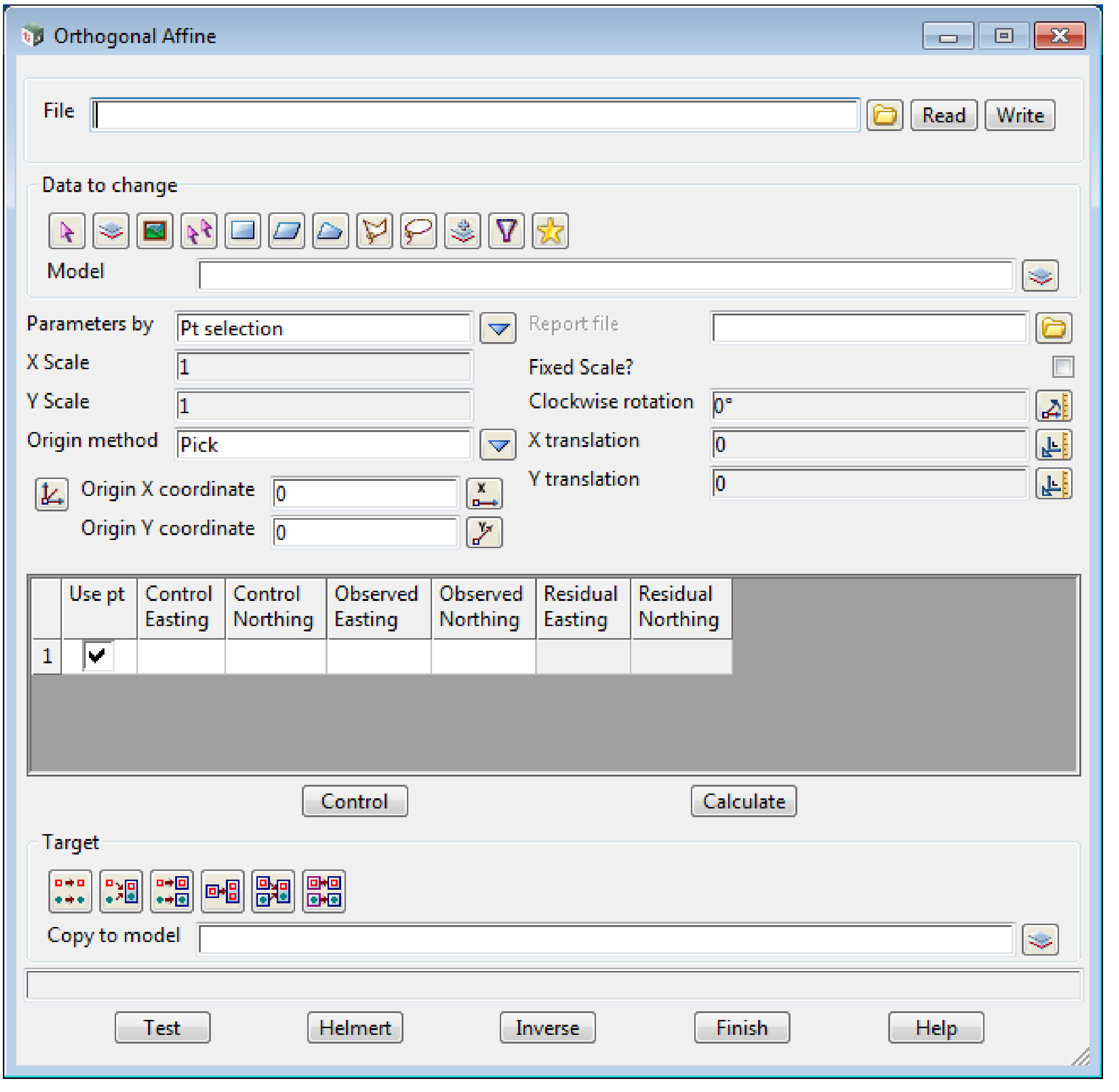
(c) the same rotation of the x-axis and y-axis of the observed coordinates to align it with the control x and coordinate axis

(d) the x and y translation of the observed coordinates after the scalings and rotations have been applied

For information on the affine transformation, please go to the section 28.9.1 Coordinate Transformations - Helmert and Affine

For information on how the **Affine Transformation works with Rasters,** go to the section 28.9.1.2 Affine and Helmert Transformations Applied to Rasters

On selecting **Affine 2D orthogonal**, the **Orthogonal Affine** panel is displayed.



If the **Parameters by** choice box is set to ***Pt Selection*** then control-observed point pairs are selected by the user and the pairs are written to the grids on the panels. See the section 28.9.1.1 Selecting Control and Observed Points for the Helmert and Affine Transformations for information on picking control and observed points.

If the **Parameters by** choice box is set to **Direct entry**, the transformation parameters are manually entered into the panel and the grid section is not used.

The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
File	file box		*.ortho_aaf files

a filename can be specified for reading or writing a file to restore the panel contents for re-use.

Read	button		

if a valid file exists, the file contents can be loaded into the panel.

Write	button		

if a valid name is specified, the user can write the input data to a file.

Data to change

Data source type

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Parameters by	choice	Pt selection	Pt selection Direct entry

the parameter entry method.

Report file			*.rpt files

if not blank, write a report on the orthogonal affine transformation.

Clockwise rotation	angle box		

the rotation of both x and y axes of the existing points with respect to the transformed x and y axes.

Fixed scale?	tick box		

if ticked, the x and y scales are typed into the ***X/Y Scale*** *fields and are held fixed when the orthogonal affine parameters are calculated from the selected control-observed point pairs.*

Scale x	input/output		

the x-scale factor for the existing co-ordinates with respect to the transformed coordinates

Scale y	input/output		

the y-scale factor for the existing coordinates with respect to the transformed coordinates

X Translation	input/output		

the x translation of the existing coordinates with respect to the transformed coordinates

Y Translation	input/output		

the y translation of the existing coordinates with respect to the transformed coordinates

Origin method	choice box	Pick	Pick, Obs'd centroid Ctrl centroid

the translation can be reported against another origin than the default (0,0).

Origin X/Y coordinate	input		

the translation parameters are written out with respect to this origin.

Control	button		

restarts the selection process for choosing more control points.

Calculate	button		

calculate the orthogonal affine parameters from the selected points shown in the grid.

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info	input		

extra information required for the target.

Affine	button		

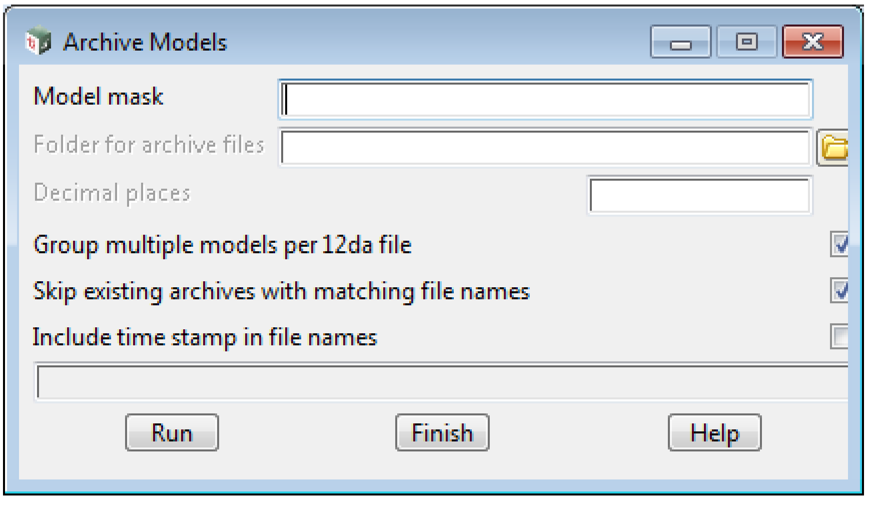
apply the orthogonal affine transformation to the data specified in the source box, and put it into the appropriate target area.

28.9.5 Archive Models

Position of option on menu: **Utilities =>A-G =>Archive models**

This option saves all models matching a specified model mask, to 12da files named in the style: <model name/prefix>[YYYYMMDD HHMMSS].12da

On selecting the **Archive models** option, the **Archive models** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults
Model mask	input text	

specifies the model(s) to archive. Use '' and '?' wildcards to specify multiple models.*

Folder for archive files	folder	

a pre-existing folder to write the 12da files to. If unspecified, will write to backups.4d or working folder, depending on the environment variable $USE_BACKUPS_4D_FOLDER_4D.

Decimal places **integer**

Number of decimal places to use when writing the 12da files. If unspecified, 15 decimal places are used.

Group multiple models per 12da file	tick box	ticked

if ticked, multiple models may be archived per 12da file, based on the matching first word of each model name. (The first word is the prefix of the model name, before the first space.) If not ticked, one model is archived per 12da file.

Skip existing archives with matching file names	tick box	ticked

if ticked, existing 12da files with matching names are left untouched and not re-archived. If not ticked, existing files are overwritten.

Include time stamp in the file names	tick box	not ticked

if ticked, include archive date and time stamp in the file names, in the style:

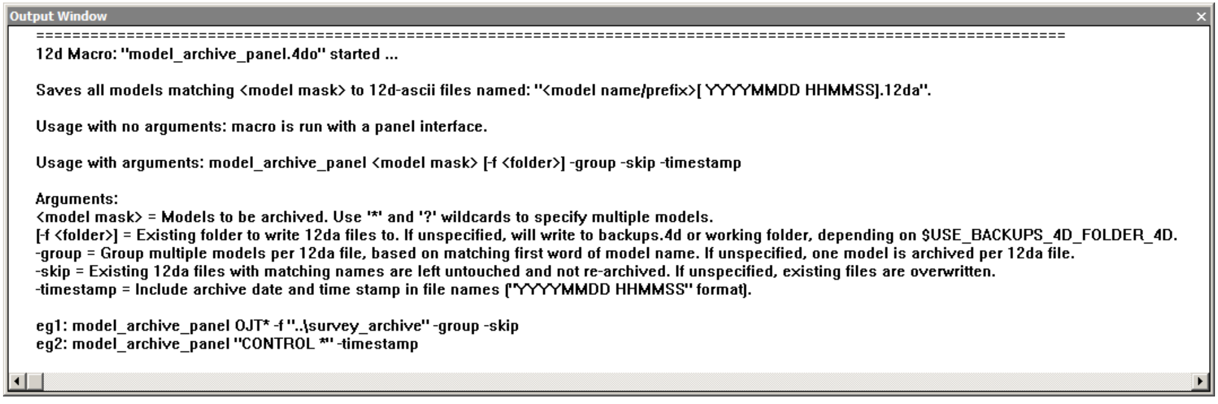
YYYYMMDD HHMMSS

Run	button	

runs the option

Special Note

The **Archive Models** option is a macro, which may optionally be run from a chain via arguments, rather than via an interactive panel. The details of the argument syntax are written to the Output Window, whenever the macro is run without valid arguments, namely:



28.9.6 Cartographic Projections

Position of option on menu: **Utilities =>A-G =>Cartographic**

The **Cartographic** option is used to transform data based on the same datum between

(a) two different cartographic projections (based on the same datum)

(b) longitude and latitude and a cartographic projection (based on the same datum)

(c) a cartographic projection and longitude and latitude (based on the same datum).

The *Cartographic* option has already been documented in the section 17.10.2 Cartographic

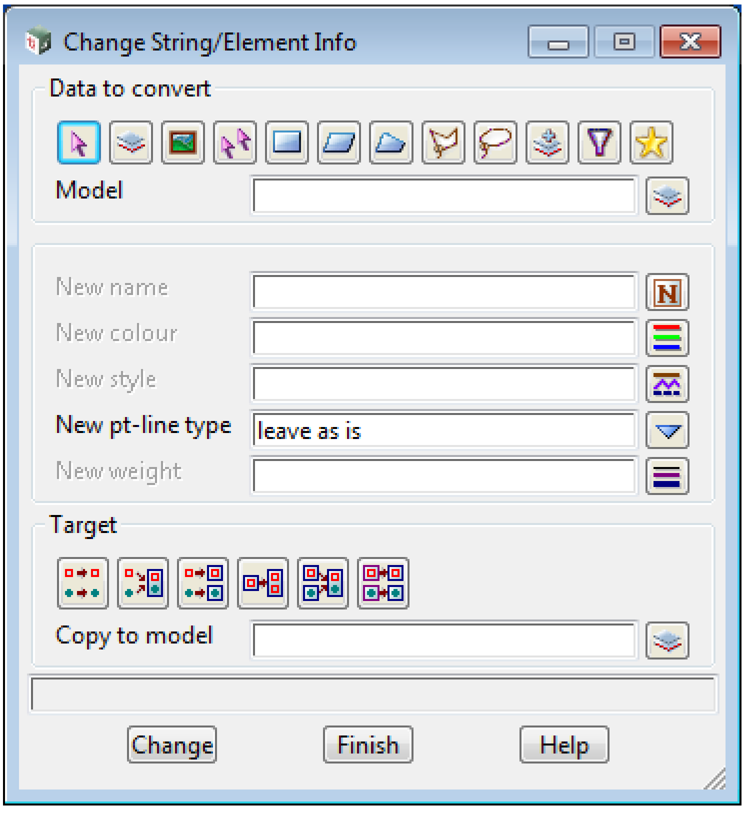
28.9.7 Change

Position of option on menu: **Utilities =>A-G =>Change**

This **Change** option is similar to the **Change** option from the **Strings edits** menu.

For the selected strings, **Change** can modify the colour, name, breakline type, style and model of each string.

On selecting the **Change** option, the **Change String Info** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be changed.

New name	input		

if non-blank, then the name of the selected strings will be changed to the name given in the **new name** *field.*

New colour	input		available colours

if non-blank, then the colour of the selected strings will be changed to the colour given in the **new colour** *field.*

New style	input		

if non-blank, then the linestyle of the selected strings will be changed to the

given in the **new style** *field.*

New pt-line type input leave as is leave as is, point, line

*if **leave as is**, then the point-line type of the selected string will be not be changed.*
*If **point string** or **line string**, then the point-line type of the selected string will be changed to that type.*

New weight input leave as is leave as is, point, line

if non-blank, then the weight of the selected strings will be changed to the

given in the **new weight** *field.*

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info input

extra information required for the target.

Change button

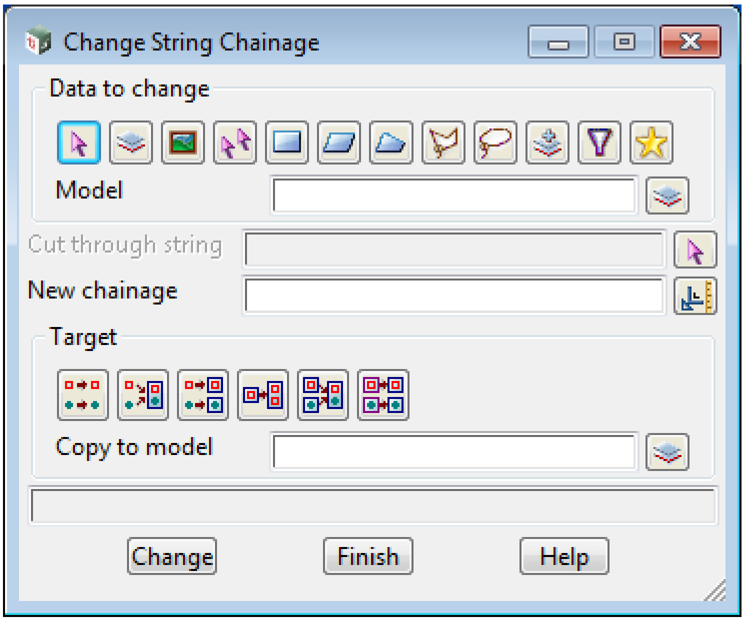
process the Data source of selected strings.

28.9.8 Change String Chainage

Position of option on menu: **Utilities =>A-G =>Change string chainage**

Change string chainage sets the start chainage of the selected strings to a given chainage value.

Selecting the **Change string chainage** brings up the **Change string chainage** panel.



The fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

Data to change

Data source type Model

data selection type - for a full description go to 4.19.3 Data Source.

Data source input

source of data to be processed.

Cut through string string select

if selected, the position on each string from the ***"Data to change"*** *source box intersects the selected string will have the chainage from* ***"New Chainage"*** *at that position.*

if no string is selected, the field is displayed as optional and the ***"New Chainage"*** *value becomes the start chainage of each string selected from the source box.*

New chainage input

value to set the start chainage of all the selected strings to.

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info input

extra information required for the target.

Change button

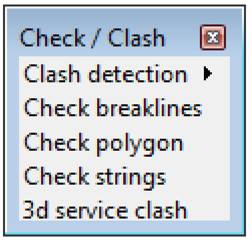
process the selected Data source of strings

28.9.9 Check/Clash

Position of menu: **Utilities =>A-G =>Check/clash**

Check/clash options run checks on strings, and clashes between strings.

The **Check/Clash** walk-right menu is



check for clashing pipes, drainage strings, *etc.*
check for crossing breaklines, duplicate vertices, *etc.*
run checks on a polygon
check strings against a Map File, *etc.*
check strings clashing in 3d

For

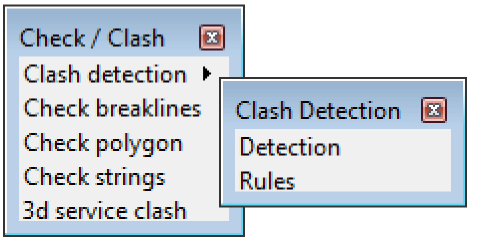
Clash detection	28.9.9.1 Clash Detection
Check breaklines, go to	16.3 Check Breaklines, Duplicate Vertices, Identicals
Check polygon	28.9.9.2 Check Polygon
Check strings	28.9.9.3 Check Strings
3d service clash	28.9.9.4 Strings Clash in 3d

28.9.9.1 Clash Detection

Position of menu: **Design =>Check/clash => Clash detection**
Utilities =>A-G =>Check/clash =>Clash detection

Clash detection checks whether strings clash with each other, and trimeshes of the clash zones can be created. **Rules** can be defined specifying what is a clash.

The **Check/Clash** menu is:



For the options, see

Detection 28.9.9.1.1 Clash Detection
Rules 28.9.9.1.2 Clash Detection Rules

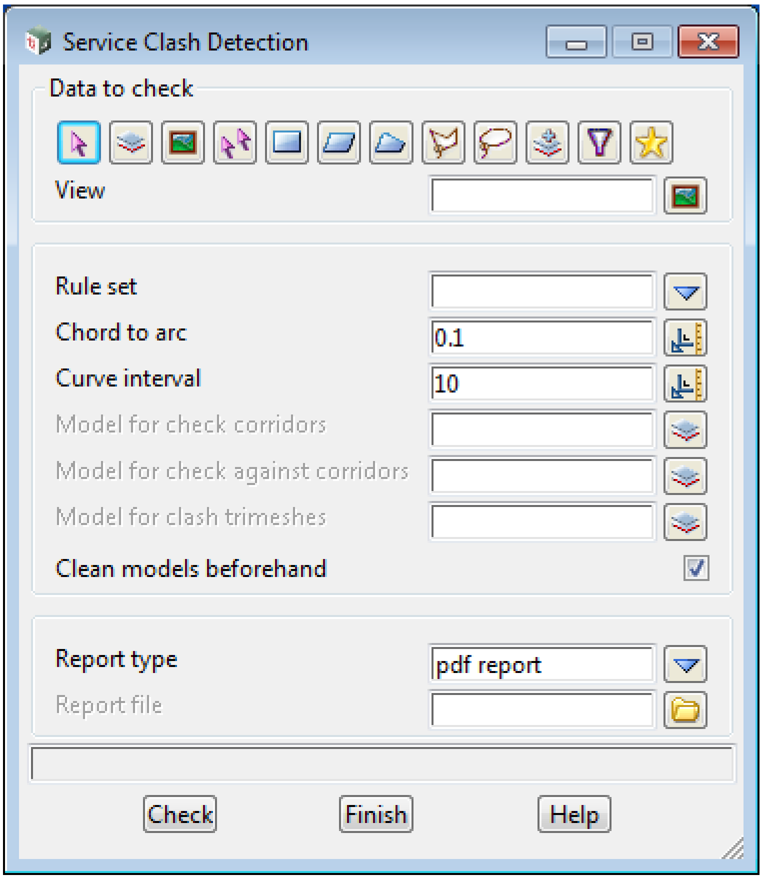
28.9.9.1.1 Clash Detection

Position of option on menu: **Design =>Check/clash => Clash detection =>Detection**
Utilities =>A-G =>Check/clash => Clash detection =>Detection

Service Clash detection checks whether strings clash with each other, and trimeshes of the clash zones can be created. **Rules** can be defined specifying what is a clash. For example, you may only want to know about a horizontal clash, but not a vertical clash, or you may want to know if anything clashes not only with a string but within a user given corridor around the string.

Clashes can be detected between strings of the types Super Pipe with straight line segments and Drainage with straight line segments.

Selecting **Detection** brings up the **Service Clash** panel



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Rule set	choice box		available clash rules

the rules to use in the clash detection. The rules define which models are checked against which models. See 28.9.9.1.2 Clash Detection Rules.

Curve interval	real box	10	

the interval to break curves up when calculating clashes.

Chord to arc real box 0.1

the chord to arc tolerance to use when breaking curves up when calculating clashes.

Model for check corridors model box

*if **not blank**, for all strings in the **Model(s) to check** given in the Rule Set, strings will be created representing the defined corridor.*

Model for check against corridors model box

*if **not blank**, for all strings in the **Model(s) to check against** given in the Rule Set, strings will be created representing the defined corridor.*

Model for clash trimeshes model box available models

*if **not blank**, trimeshes are created to indicate where the clashes occur, and are placed in this model.*

Clean models beforehand tick box

*if **ticked**, the models for check corridors, check against corridors and clash trimesh models are cleaned before processing.*

Report type choice box available xml formats for this panel

the type of report to be generated from the xml file.

New customised report formats can be added by the user. See 4.30 Setting Up XML Reports

Report file file box

*if **not blank**, a report file of this name is created.*

*If **blank**, no report is created.*

Check button

perform the clash checks. Error messages are written to the Output Window as Intelligent Log Lines to help find the problems.

28.9.9.1.2 Clash Detection Rules

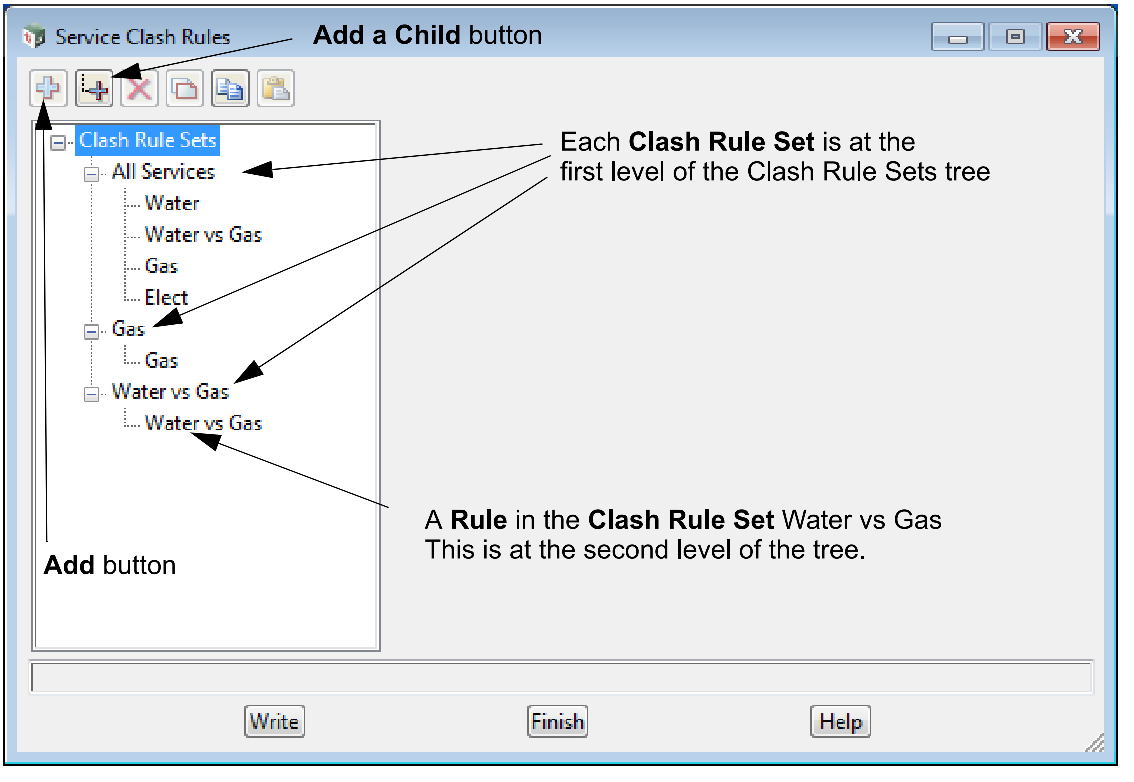
Position of option on menu: **Design =>Check/clash => Clash detection =>Rules**
Utilities =>A-G =>Check/clash => Clash detection =>Rules

The **Service Clash Rules** define what is meant by a "clash".

The **Service Clash Rules** is a tree of Clash Rule Sets and each set has a unique name. The **Service Clash Detection** panel specifies which of the **Clash Rule Sets** is to be used for a particular clash detection run.

The ***Service Clash Rules*** are loaded from a file called ***service_clash_rules.xml*** that is searched for as a standard ***Set_Ups*** file. See 39.2 Files for Setting Up 12d.

Selecting **Rules** brings up the **Service Clash Rules** panel



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up

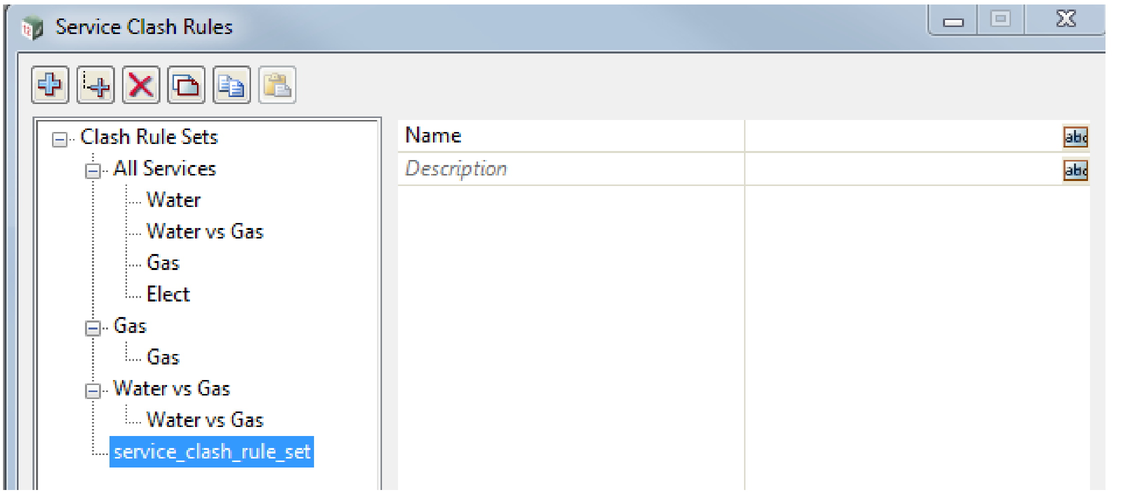
Clash Rule Sets Tree

(a) Clash Rule Set

A **Clash Rule Set** is defined at the first level of the **Clash Rule Sets** tree and each **Clash Rule Set** must have a unique name. One **Clash Rule Set** is selected in the **Service Clash Detection** panel to provide the rules to be used for a particular clash detection run

A new *Clash Rule Set* is created by clicking and highlighting the top level *Clash Rules Sets* and then pressing the **Add Child** button, or by clicking on and highlighting a first level *Clash Rule Set* and then pressing the **Add** button.

A new *Clash Rules Set* with the dummy name **service_clash_rule_set** is then created and a new name must be entered in the **Name** field on the right hand side of the tree. The **Description** is optional.

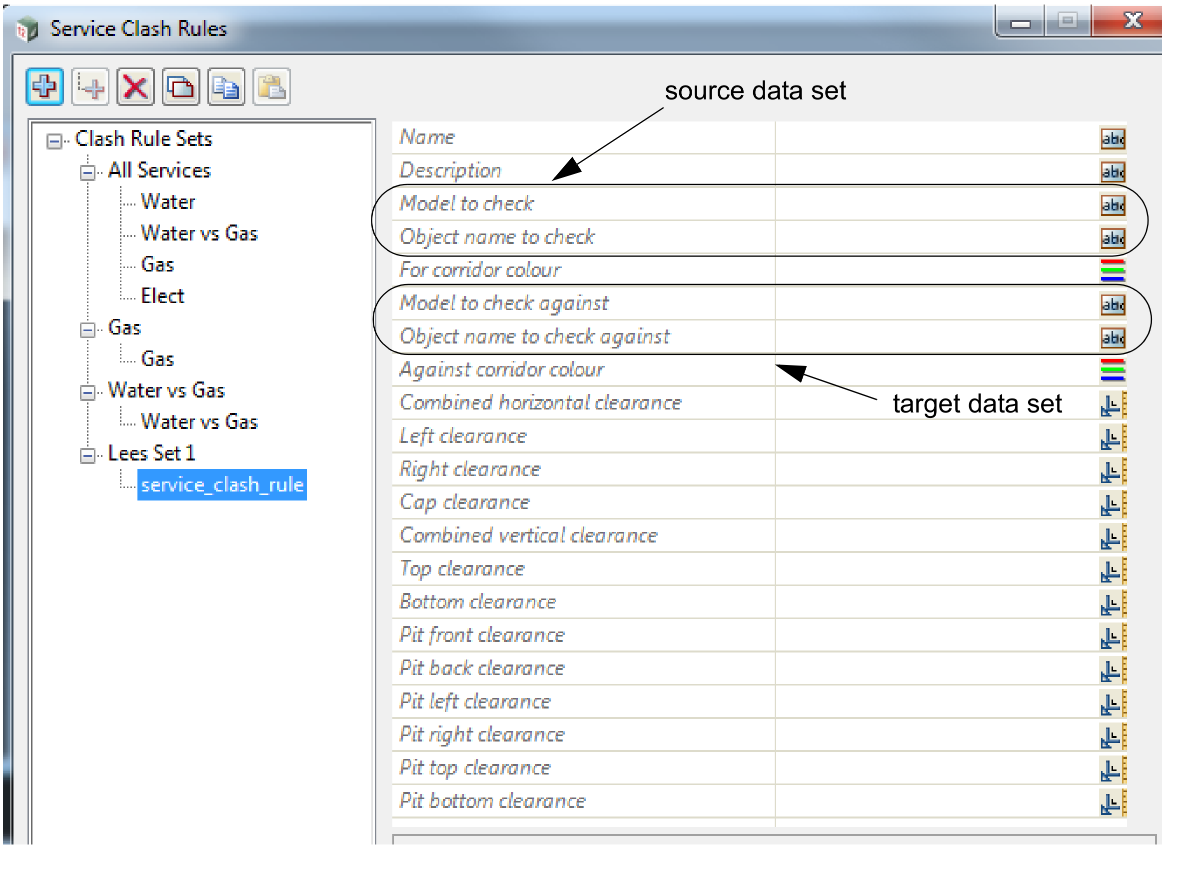


(b) Rules in a Clash Rule Set

The rules for a named Clash Rule Set as defined as second level items under the *Clash Rule Set*. The rules define what is meant by a particular clash.

A new *Rule* is created by clicking and highlighting the named *Clash Rules Set* and then pressing the **Add Child** button, or by clicking on and highlighting a Rule already in the named *Clash Rule Set* and then pressing the **Add** button.

A new *Rule* with the dummy name **service_clash_rule** is then created and a new name must be entered in the **Name** field on the right hand side of the tree. The **Description** is optional.



There must be at least one clash rule in a Clash Rule Set otherwise there is nothing to do. If such a Clash Rule Set is used then no results will be reported.

The Rule defines two separate datasets for each rule - the **Source dataset** containing the elements being checked, and the **Target dataset** containing the elements being checked against. Elements cannot exist in both. They are in either one or the other or neither.

The **Model to check** and **Object name to check** defines the elements that are in the Source data set. Both fields can contain wild cards and wild characters and blank fields will match all elements. That is, blank fields are taken as the wild character *.

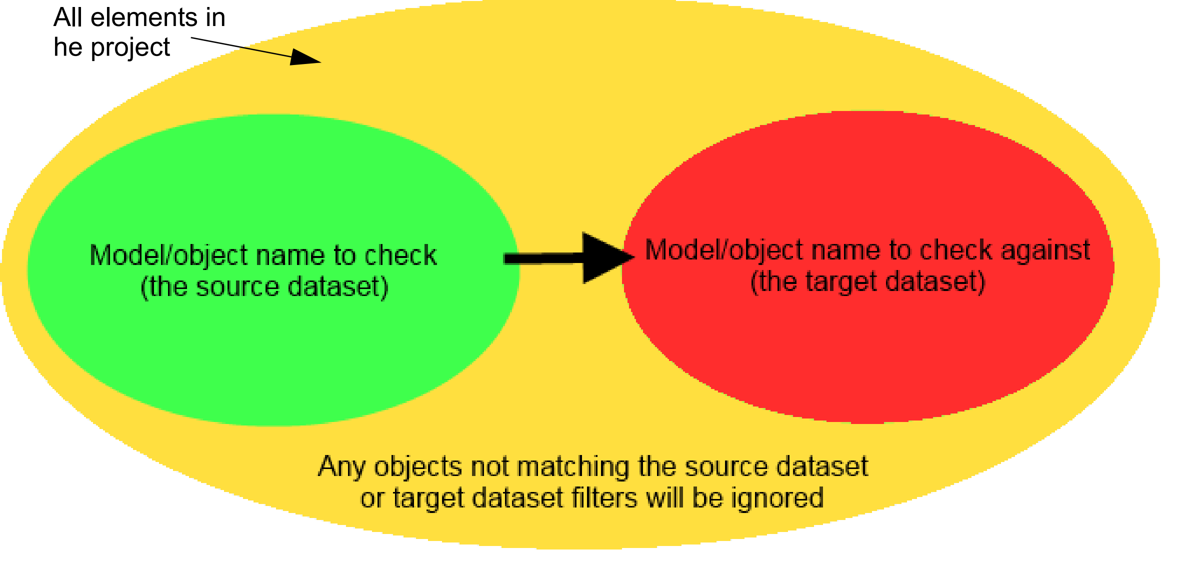
The **Model to check against** and **Object name to check against** defines which of the remaining elements (where remaining = all elements minus source data) are in the **Target dataset**. Both fields can contain wild cards and wild characters.

Elements in the *Source dataset* will **only be checked** against elements in the T*arget dataset.* Any other objects not in the *Source dataset* or *Target dataset* are ignored

So you should be specific for the **Source dataset** and cannot be too generic, since being too generic will mean that there are few elements left to be in the **Target dataset** leaving little or nothing to check against

Hence you should be fairly specific in your **Source dataset** filters, but can be totally generic in the **Target dataset** filters.

To see this in a diagram, if the yellow region represents all the elements in the project and the green region the elements in the *Source dataset*, then the Target dataset (red region) can only be totally outside the green region. So if the green region is very large, then there are not many elements left to be in the red region. The elements in the yellow region that aren't in the red or green regions are the objects in the project that are totally ignored by that rule.



Recommendations for Using the Service Clash Detection:

(a) Model each type of utility in its own model, *e.g.* GAS, WATER, ELEC, TELECOM.

(b) As a minimum create a rule for each service type.

(c) If there are different clearances for different target services being checked, create a rule for each case. *e.g.* Gas vs Water, Gas vs Elec, Gas vs Other.

(d) If you need to check for clashes between utilities of the same type, place these in different models or name them differently so that they can be selected by name, *e.g.* WATER TRUNK and WATER CONNECTIONS or ELEC LEFT and ELEC RIGHT.

(e) Be as narrow and specific with the Source filters as you can be. There is a trade-off, however, between being specific and needing to create multiple rules. Find a balance.

(f) Use standard naming for models and strings and develop standard clash detection rule sets.

(g) You can use map files to help filter out or map strings to certain models to match existing rule sets.

Write Button

When the **Write** *button is selected, a* **Write Setup File** *panel comes up to specify where the* ***services_clash_rules.xml*** *file is to be written out to.*

For the choices on the panel, see 39.2.6 Writing Out Setup Files

A project restart is required for the new file to take effect

28.9.9.2 Check Polygon

Position of option on menu: **Utilities =>A-G =>Check/clash =>Check polygon**

Check polygon checks to see if a polygon passes the polygon checks performed by a polygon box.

The checks are to try and ensure that the polygon is suitable for use in some ***12d*** options.

For example, a polygon that crosses over itself, or has duplicate vertices, or has edges partially on top of other edges, or in parts consists of a single line, are not usable in most options requiring a polygon (fence) for calculations.

The ***Exact Volumes*** options in particular are very sensitive to having a well formed polygon whereas other options can use polygons that are unsuitable for *Exact Volumes*.

Some of the things that make a polygon unsuitable (and will produce the message "bad polygon" or "bad poly") are:

(a) self intersects in the polygon

(b) multiple vertices at the same (x,y) location

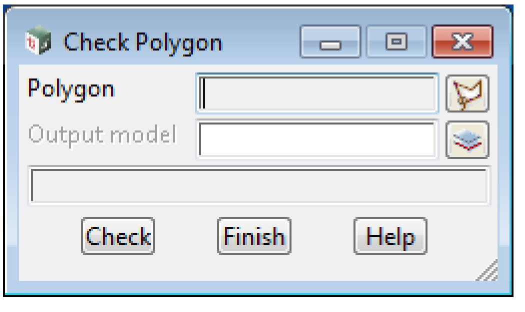
(c) sides being partially on top of other sides

(d) corners just touching other sides or vertices

(e) parts of the polygon that contain no area. For example, a loop at one end and then a line going out from the loop. The line does not enclose any area. This may be created by contouring a surface that contains has a ridge line of constant height.

Errors in the polygon are reported in the ***Output Window*** with intelligent log lines to help find the section of the polygon with the error.

Selecting **Check polygons** brings up the **Check Polygon** panel



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Polygon	polygon box		

select a polygon to check

Output model

*if **non blank**, the resolved polygon is added to this model*

Check button

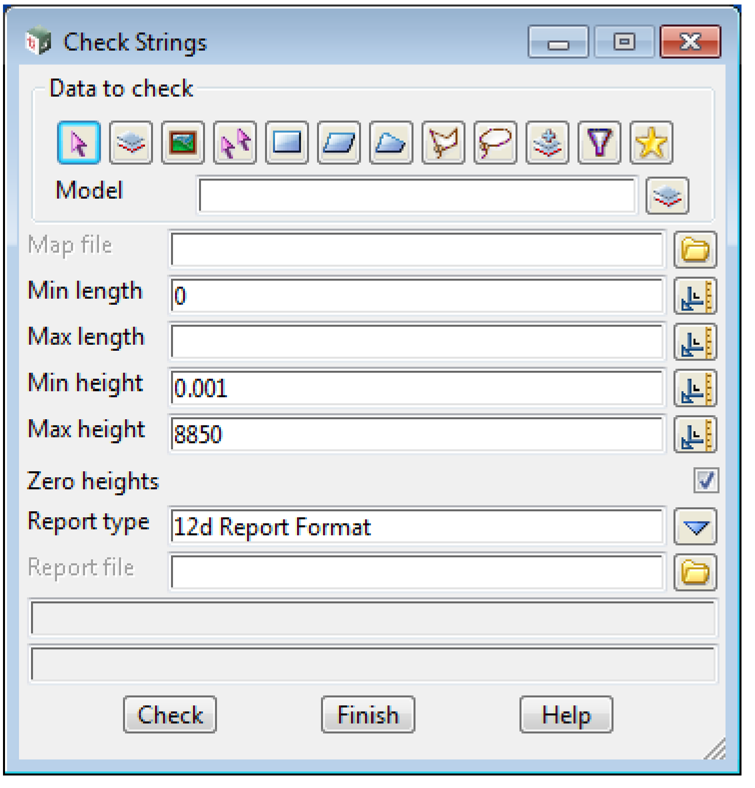
perform the polygon checks. Error messages are written to the Output Window as Intelligent Log Lines to help find the problems.

28.9.9.3 Check Strings

Position of option on menu: **Utilities =>A-G =>Check/clash =>Check strings**

Check strings compares selected strings against a ***Map File*** and also checks distances between vertices on strings.

Selecting **Check strings** brings up the **Check Strings** panel.



The fields and buttons used in the panel have the following functions.

Field Description   Type   Defaults   Pop-Up

Data source type   Model

data selection type - for a full description go to 4.19.3 Data Source.

Data source   input

source of data to be processed.

Map file   map file box

if a map file is picked then each selected string is compared against the map file keys and if any matches occur, the string attributes must agree with the map file fields of the matched key. Any differences will be reported to the Output window and the Report file.

Min length   input

report if any distance between adjacent vertices in a string is less than Min length.

Max length   input

report if any distance between adjacent vertices in a string is greater than Max length.

Min height   input

report if any vertex height is less than Min height.

Max height   input

report if any vertex height is greater than Max height.

Zero heights tick box

if ticked, any vertex heights equal to 0.0 will be reported. It is vary rare that any height is exactly 0.0

Report file

*if **non blank**, any failures are reported in this file.*

Check button

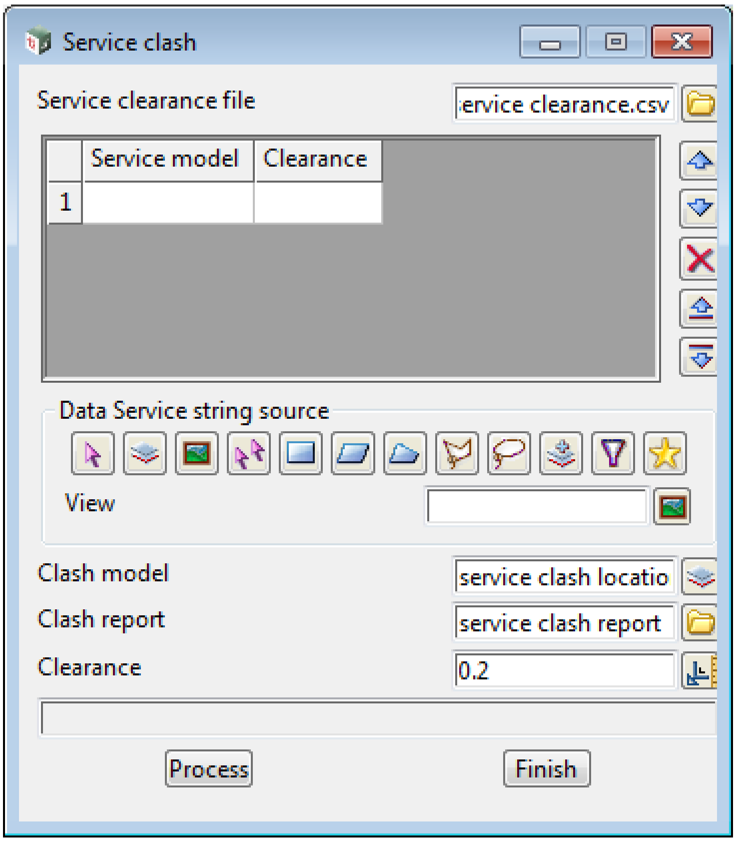
process the selected strings

28.9.9.4 Strings Clash in 3d

Position of option on menu: **Utilities =>A-G =>Check/clash =>3d service clash**

3d service clash compares selected models of strings and checks to see if they intersect each other.

Selecting **3d service clash** brings up the **Service Clash** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Service clearance file	file box		*.csv files

csv file containing a list of model names and clearances. This file is read in as soon as the option starts and fills out the Service and Clearance grid.

Grid - Service model and Clearance

For a row in the grid:

*any string in the model of name **Service model** must have the clearance given in the **Clearance** column. That is, any other string must be at least the **Clearance** distance from the string from the **Service** model.*

When the option starts, the grid is loaded from the **Service clearance file**. *The grid can be modified and whenever the* **Process** *button is selected, the contents of the grid is written to the file name given by* **Service clearance file.**

Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Each string in the data source is processed against all other strings selected by the Data source.

If a string is from a model listed in the ***Service model and Clearance*** *grid, then the corresponding clearance is used for the string, otherwise the default clearance given in the* **Clearance** *panel field is used.*

Clash model model box available models

the lines of clash between any two strings is placed in the **Clash model**.

Clash report

a report is generated for any clashes

Clearance input

the default clearance to use for a string that is ***not*** *in any of the Service models given in the* ***Service model and Clearance*** *grid*

Process button

process the selected strings and check for clashes

28.9.10 Classify Conduits

Position of option on menu: **Utilities =>A-G =>Classify conduits**

This option sets attributes on “conduit strings”, *i.e.* strings of the following type only: *Super* (with conduit dimensions), *Drainage*, *Pipeline*, *Pipe*.

String attributes set are:

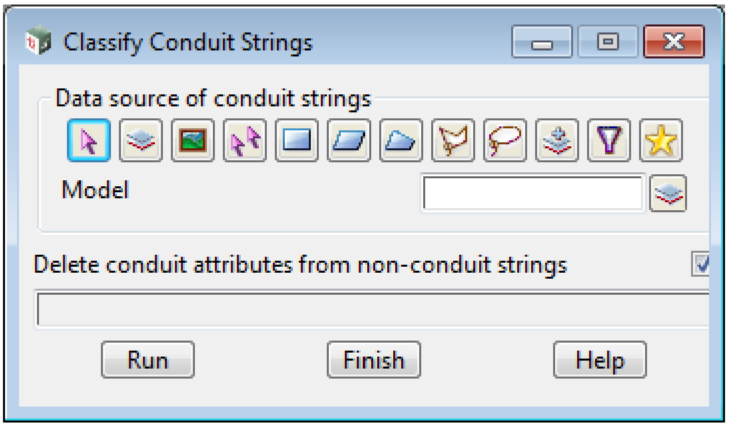
"justification" (text) on all strings - values set to "invert", "obvert" or "centre".
"pipe size" (text) on constant-size Super string conduits, Pipeline and Pipe strings.
"diameter" (real) on constant-size Super string conduits, Pipeline and Pipe strings.
"width" (real) on constant-size Super string box conduits.

Segment attributes set are:

"pipe size" (text) on all Super string conduit segments.
"diameter" (real) on all Super string conduit segments.
"width" (real) on all Super string box conduit segments.

Notes:
1) *Drainage* and *Pipeline* strings are always invert justified.
2) *Pipeline* and *Pipe* strings are always circular with a constant diameter.
3) *Super* string conduits may be circular or box. *Drainage* strings may be circular, box, vee or trapezoid.
4) For non-circular conduit shapes, attribute "diameter" represents height of conduit.
5) Attribute "pipe size" is in mm (base units x 1000) and supports shapes: circular, box, vee, trapezoid - eg: "600", "1200x600", "V2000x600", "T2000B1200x600".
6) *Drainage* strings already have Pipe attributes "diameter" and "pipe size" (and potentially "width" and/or "top width") set via the *Drainage Network Editor*.

On selecting the **Classify conduits** option, the **Classify Conduit Strings** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source of conduit strings input

source of data to be processed

Delete conduit attributes from non-conduit strings tick box ticked *whether to delete the above mentioned string and segment attributes (if present) from non-conduit string types*

Run button

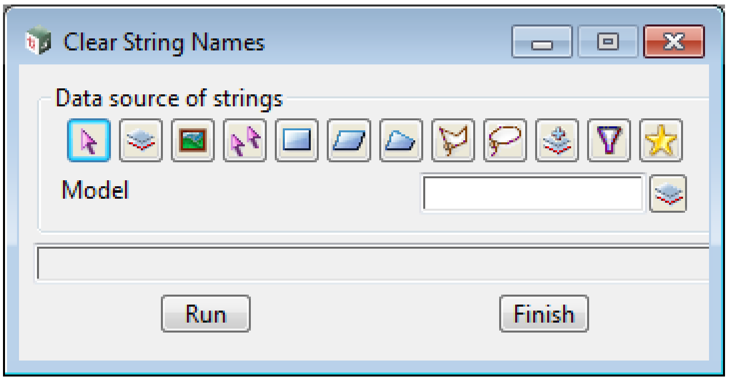
runs the option

28.9.11 Clear String Names

Position of option on menu: **Utilities =>A-G =>Clear string names**

Clear string names removes the names for all the selected strings.

Selecting **Clear string names** brings up the **Clear String Names** panel.



The fields and buttons used in the panel have the following functions.

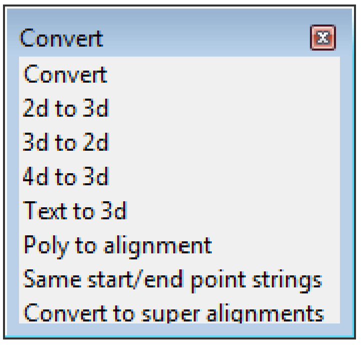
Field Description	Type	Defaults	Pop-Up
Data source type		Model	
data selection type - for a full description go to 4.19.3 Data Source.			
Data source	input		
source of data to be processed.			
Run	button		
remove/delete the string names from the selected strings			

28.9.12 Convert

Position of menu: **Utilities =>A-G =>Convert**

The **convert** options are designed to convert large numbers of strings between string types. There is also a **global convert** option which will convert even more string types.

The **convert** walk-right menu is



convert bulk strings
convert 2d strings to 3d strings
convert 3d strings to 2d strings
convert 4d strings to 3d strings
convert text to 3d points
convert polylines to alignments
convert closed strings to closed super strings
convert strings to super alignments

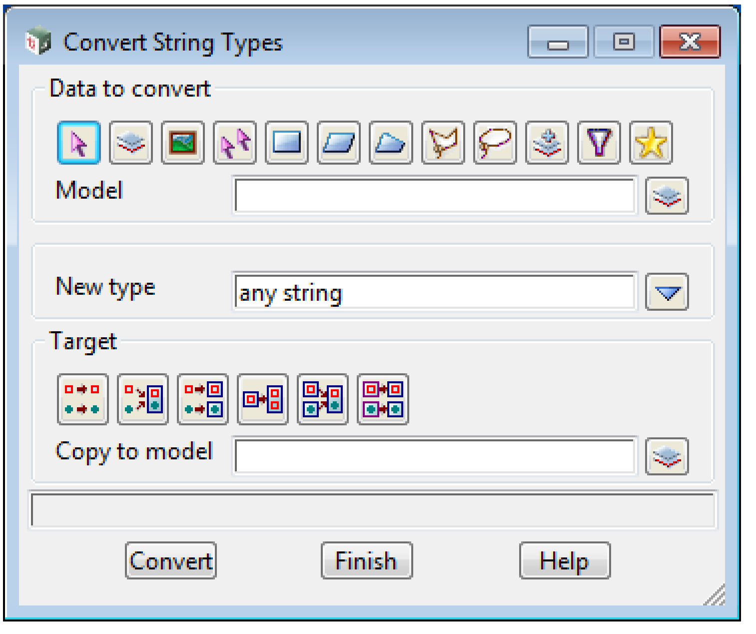
For the option *Convert*, go to the section 28.9.12.1 Convert
2d to 3d 28.9.12.2 2d to 3d
3d to 2d 28.9.12.3 3d to 2d
4d to 3d 28.9.12.4 4d to 3d
Text to 3d 28.9.12.5 Text to 3d
Poly to alignment 28.9.12.6 Poly to Alignment
Same start/end point strings 28.9.12.7 Same Start/End Point Strings
Convert to super alignments 28.9.12.8 Convert to Super Alignment

28.9.12.1 Convert

Position of option on menu: **Utilities =>A-G =>Convert =>Convert**

The **Convert** option converts strings to a selected string type.

On selecting the **Convert** option, the **Convert String Types** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Data source type | | Model |

data selection type - for a full description go to 4.19.3 Data Source.

Data source | input

source of data to be processed.

New type | string type box | | all string types

the type of string to try and convert the data to.

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info | input

extra information required for the target.

Convert | button

convert all the selected strings to the type given in the **New type** *field.*

28.9.12.2 2d to 3d

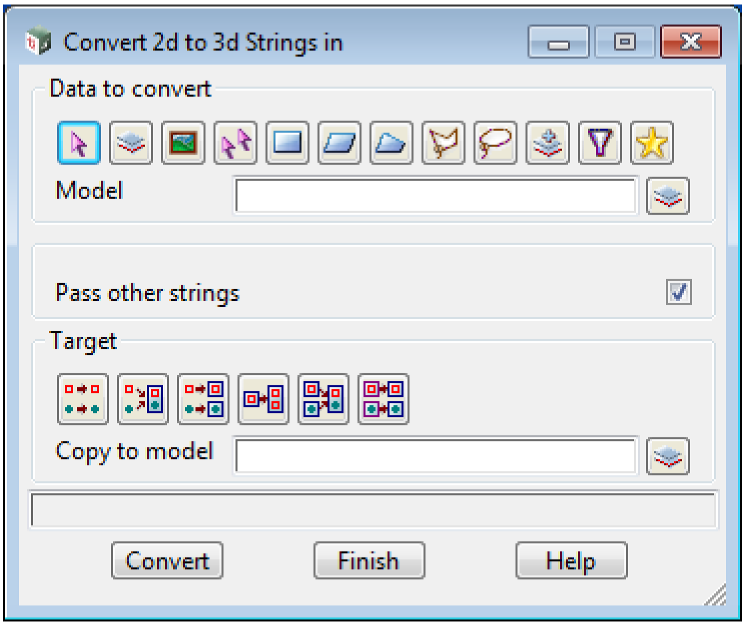
Position of option on menu: **Utilities =>A-G =>Convert =>2d to 3d**

The main difference between a 2d and a 3d string is that all the points in a 2d string have the same height. Hence to modify the z-values at individual points, a string must be 3d.

The **2d to 3d** option is used to turn 2d (contour) strings into 3d strings.

This option is especially useful when a large number of strings have been read in with constant z-values (for example, cadastre) and the user then wants to modify the heights at each point on the string.

On selecting **2d to 3d** and then the appropriate **Data Source** in the panel, the **Convert 2d to 3d Strings in** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Pass other strings	tick box	tick	

*if **ticked**, copies of all other strings are passed through to the model for converted strings*

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info	input		

extra information required for the target.

Convert	button		

convert all the 2d string in the Data source to 3d strings.

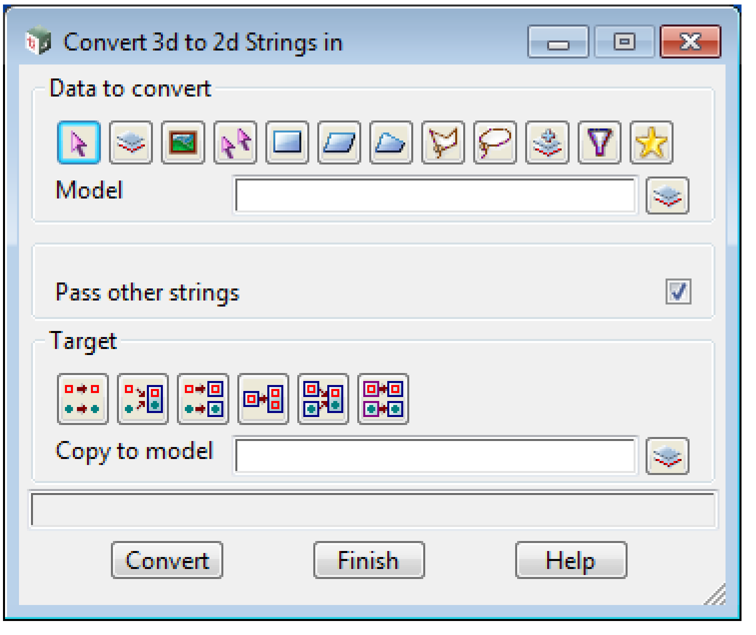
28.9.12.3 3d to 2d

Position of option on menu: **Utilities =>A-G =>Convert =>3d to 2d**

The difference between a 3d and a 2d string is that all the points in a 2d string have the same height. The **3d to 2d** option turns 3d string with a constant height into 2d strings. That is, it only converts 3d strings that have the **same** height at each point.

This option is useful when contours strings have come from another system as 3d strings, not 2d strings.

On selecting **3d to 2d** and then the appropriate **Data Source** in the panel, the **Convert 3d to 2d Strings in** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Pass other strings	tick box	tick	

*if **ticked**, copies of all other strings are passed through to the model for 2d strings*

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info	input		

extra information required for the target.

Convert	button		

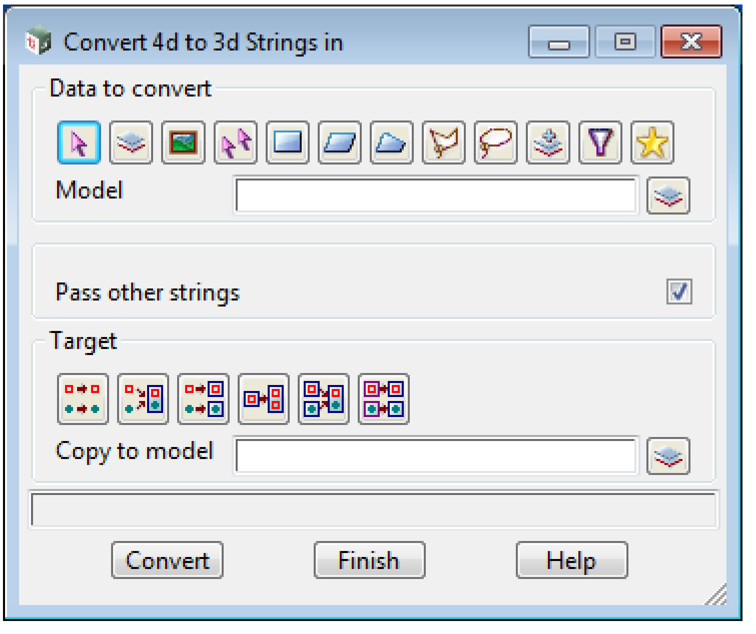
*convert all the 3d strings with **constant** height in the Data source to 2d strings.*

28.9.12.4 4d to 3d

Position of option on menu: **Utilities =>A-G =>Convert =>4d to 3d**

The difference between a 4d and a 3d string is that a 4d string can have a text label at each point on the string. The **4d to 3d** option turns 4d string into 3d strings by stripping off the text labels at each point of the 4d string.

On selecting **4d to 3d** and then the appropriate **Data Source** in the panel, the **Convert 4d to 3d Strings in** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source input

source of data to be processed.

Pass other strings tick box tick

*if **ticked**, copy all other strings and pass them through to the model for 3d strings*

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info input

extra information required for the target.

Convert button

convert all the 4d strings in the Data source to 3d strings.

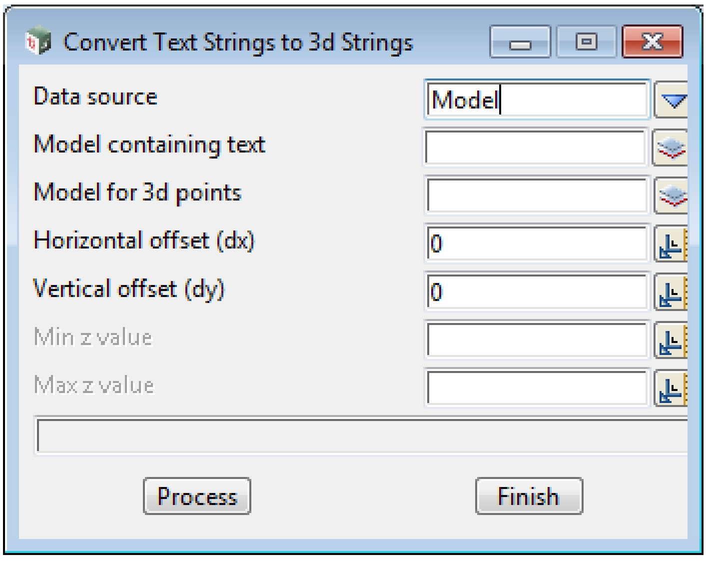
28.9.12.5 Text to 3d

Position of option on menu: **Utilities =>A-G =>Convert =>Text to 3d**

Sometimes when data for points is received from a CAD systems, all that is received is the text of the z-value of the point and no actual point.

The *Text to 3d* option will find text representing a z-value and create a new point using the justification point of the text as the (x,y) position and the value of the text as the z-value.

On selecting **Text to 3d**, the **Convert Text Strings to 3d Strings** panel is displayed.



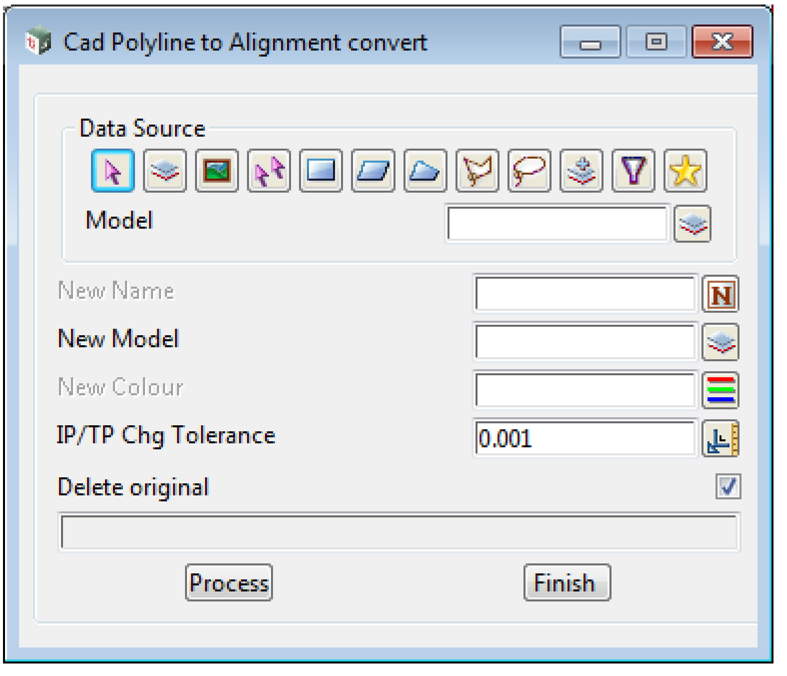
The fields and buttons used in this panel have the following functions

Field Description	Type	Defaults	Pop-Up
Data source type		Model	
data selection type - for a full description go to 4.19.3 Data Source.			
Data source	input		
source of data of the text to be processed.			
Model for 3d points	model box		available models
model to place the 3d points in.			
Horizontal offset (dx)	input	0	
the x-value for created point is the text justification point less the horizontal offset.			
Vertical offset (dy)	input	0	
the y-value for created point is the text justification point less the vertical offset.			
Min z value	input		
if non-blank, only convert text with z-value greater than **Min z value**.			
Max z value	input		
if non-blank, only convert text with z-value less than **Max z value**.			
Process	button		
convert all the selected text strings to 3d strings.			

28.9.12.6 Poly to Alignment

Position of option on menu: **Utilities =>A-G =>Convert =>Poly to alignment**

Selecting **Poly to alignment** brings up the **CAD Polyline to Alignment Convert** panel.



The fields and buttons used in this panel have the following functions

Field Description	Type	Defaults	Pop-Up
Data source type		Model	
data selection type - for a full description go to 4.19.3 Data Source.			
Data source	input		
source of data of the text to be processed.			
New name	name box		available names
*if **non blank**, new name for the converted strings*			
New model	model box		available models
new model for the converted strings			
New colour	colour box		available colours
*if **non blank**, new colour for the converted strings*			
IP/TP chg tolerance	input		0.001
Delete original	tick box		tick
*if **ticked**, the original strings are deleted.*			
Process	button		
convert all the selected alignments.			

28.9.12.7 Same Start/End Point Strings

Position of option on menu: **Strings =>Properties =>Same start/end point strings**

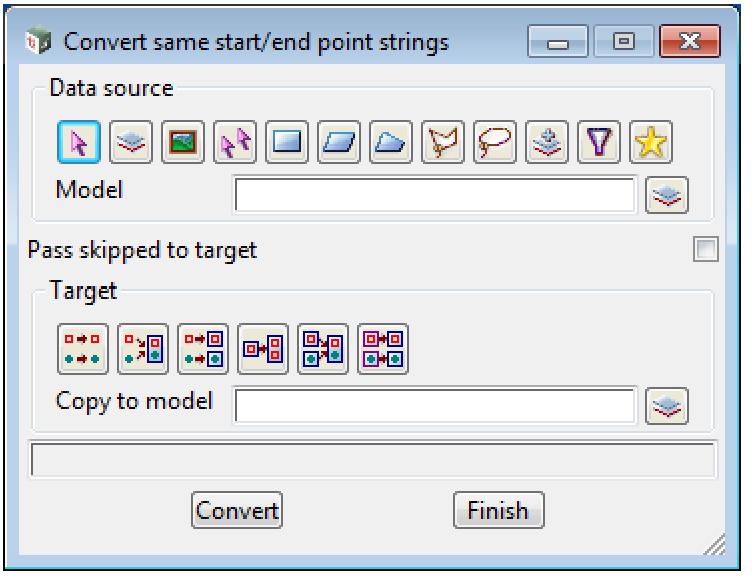
Position of option on menu: **Utilities =>A-G =>Convert =>Same start/end point strings**

Position of option on menu: **Utilities =>Super strings =>Same start/end point strings**

When non super string strings are closed, an extra vertex identical to the first vertex is added to the end of the string. For super strings, there is a ***closed string*** flag and no duplication of the first and last vertices is required.

This options converts closed non super strings to closed super strings and deletes the duplicated vertex.

Selecting **Same string/end point strings** brings up the **Convert Same Start/End Point Strings** panel.



The fields and buttons used in this panel have the following functions

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Pass other strings	tick	tick	

*if **ticked**, copy all other strings and pass them through to the target model*

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info	input		

extra information required for the target.

Convert	button		

convert all the closed strings to closed super strings with duplicated end point.

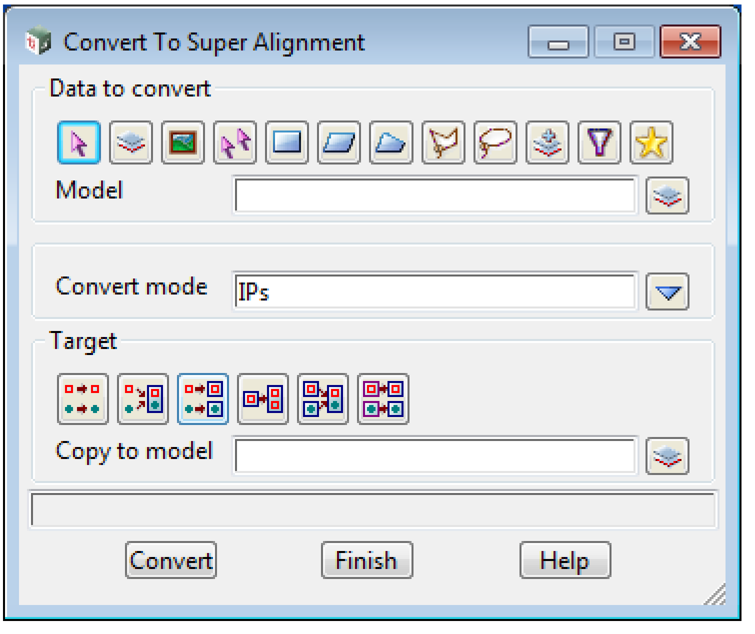
28.9.12.8 Convert to Super Alignment

Position of option on menu: **Utilities =>A-G =>Convert =>Convert to super alignments**

The **Convert** option converts strings to super alignments, either with just IPs or using Elements.

Note - conversion to *Elements* is only possible if you have the ***Alignment*** module.

Selecting **Convert to super alignments** brings up the **Convert to Super Alignment** panel.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

Data source type Model

data selection type - for a full description go to 4.19.3 Data Source.

Data source input

source of data to be processed.

Convert mode choice box IPs, Elements

*if **IPs**, the strings are converted to super alignments with IPs.*

*If **Elements**, the strings are converted to super alignments using Elements.*

Note - if the user does not have the Alignment module, then the **Convert mode** *is automatically set to IPs.*

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info input

extra information required for the target.

Convert button

convert all the selected strings to either super alignments with IPs or super alignments with Elements.

28.9.13 Cuts

Position of menu: **Utilities =>A-G =>Cuts**

Cuts is used to create intersections through string data rather than tins.

That is, it calculates the cuts that a plan string makes with each of the strings in a model or view, at creates cut points with the z-value from the cut strings.

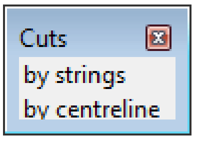
The cut points that the plan string creates are connected as a 4d string - the x-section of cuts through a model or view - with the z value coming from the cut string, and the text at the point being the name of string that was cut.

There are two methods for generating the cuts x-sections

(a) by taking sections along selected strings or all the strings in a model or view

(b) by taking sections at regular intervals and perpendicular to a selected centre line string.

The **Cuts** walk-right menu contains these two methods



For the option *by strings*, go to the section 28.9.13.1 Cuts by Strings.
by centreline 28.9.13.2 Cuts by Centreline.

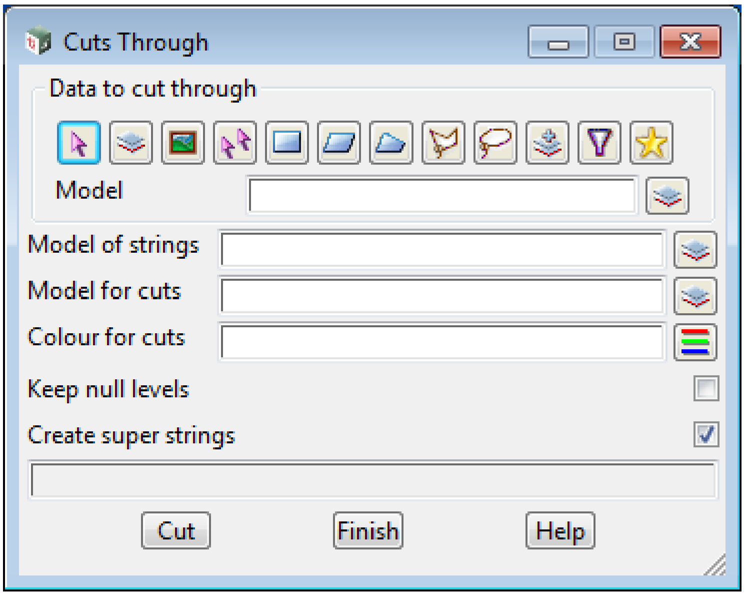
28.9.13.1 Cuts by Strings

Position of option on menu: **Utilities =>A-G =>Cuts =>by strings**

by strings generates cuts through a Data source of strings, for all the strings in a user specified model.

For each string in the **Model of strings**, the plan representation of the string is cut through the all the string in ***Data to cut through***, to generate cut x-sections.

Selecting **by strings** displays the **Cuts Through** panel.



The fields and buttons used in this panel have the following functions

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source input

source of data to be cut through.

Model of strings input available models

each string in this model will be processed against all the strings in the Data source ***Data to cut through.***

Model for cuts input available models

model to place the cut x-sections into.

Colour for cuts input available colours

colour for the cut x- section strings

Cut button

each string in the **Model of strings** *is processed against all the strings in the Data source* ***Data to cut through****. The 4d strings of cuts are placed in the* **Model for cuts**.

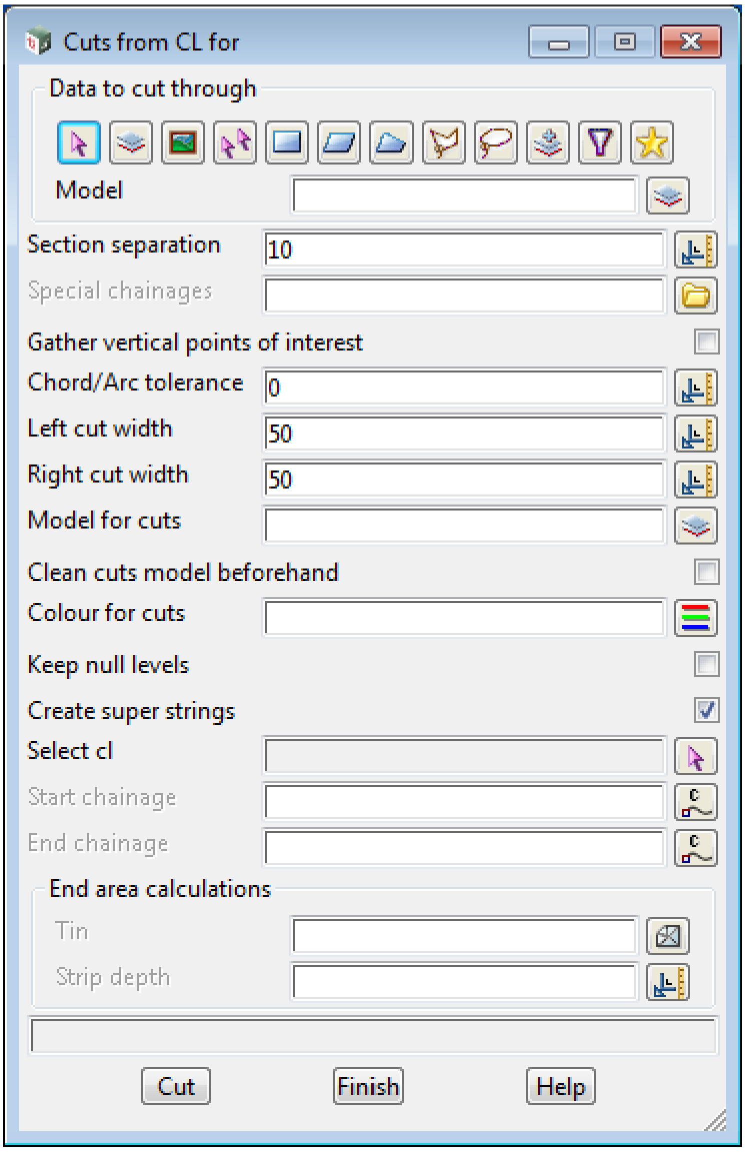
28.9.13.2 Cuts by Centreline

Position of option on menu: **Utilities =>A-G =>Cuts =>by centreline**

For **by centreline**, a centre line string is selected by the user and temporary plan lines are created at regular intervals perpendicular to the selected string.

Cut x-sections are then generated for the temporary plan section lines by taking cuts through the Data source **Data to cut through**.

Selecting **by centreline** displays the **Cuts from CL for** panel.



The fields and buttons used in this panel have the following functions

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Section separation	input	10.0	

the distance along the selected centre line to generate plan section lines to be used to cut through the *data source.*

Special chainages input *.spf files

a file containing chainages, one per line, that are also used as chainages to create cross sections at.

Gather vertical points of interest tick box

whether or not to create cuts at vertical points of interests (sags, crests, vertical tangent points etc)

Chord/arc tolerance input default chord/arc tolerance

the chord to arc tolerance to use on the selected string for determining how many plan sections are created around horizontal curves.

Left/Right cut width input 50

the left/right distance to go out from the centre line for creating a section to cut through the strings.

Model for cuts input available models

model to place the cut x-section strings into.

Clean cuts model beforehand tick box

*if **ticked**, the model is cleaned of all data before new sections are created.*

Colour for cuts colour box available colours

colour for the cut section strings

Keep null levels tick box

*if **ticked**, then if a null level exists on the string being cut, then a null level point is created.*

Select cl string-select

the selected cl string is used to create plan section lines at regular intervals and perpendicular to the centre line string. These strings are cut though the model/view of strings.

Start/End chainage input

if non-blank then sections for the cuts are restricted to between the given start and end chainage of the selected cl string.

End Area Calculations

Tin tin box available tins

*if **non-blank**, the cut and fill areas for the section against the tin are calculated and added as attributes to the section.*

Strip depth input

*if **non-zero**, then a strip depth is removed from the tin before the cut and fill areas for the section against the tin are calculated.*

Cut button

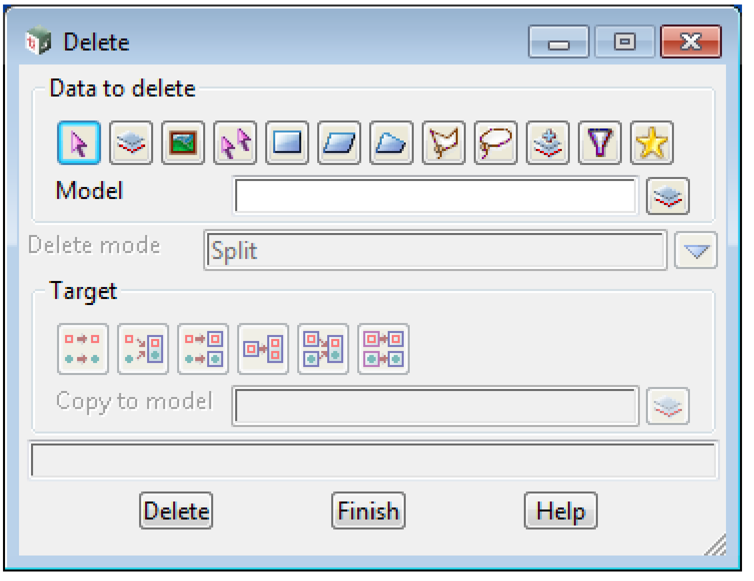
each plan section line generated down the selected centre line string is processed against all the strings in the Data source. The 4d strings of cuts are placed in the **Model for cuts**.

28.9.14 Delete

Position of option on menu: **Utilities =>A-G =>Delete**

The **Delete** option deletes all the selected strings.

On selecting the **Delete** option, the **Delete** panel is displayed.



The fields and buttons used in this panel have the following functions

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source	input		

source of data to be processed.

Delete	button		

delete the selected strings.

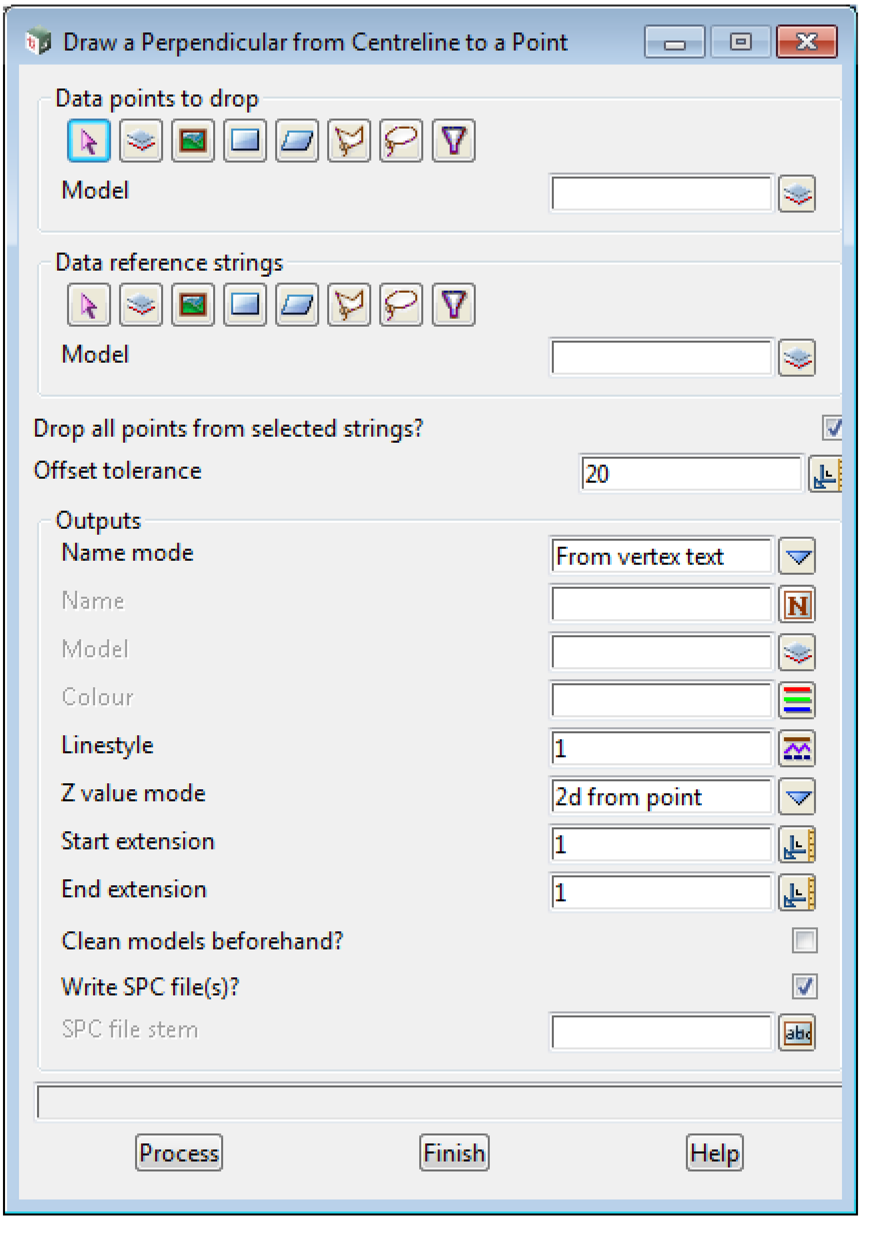
28.9.15 Draw a Perpendicular from Centreline to a Point

Position of option on menu: **Utilities =>A-G =>Draw perpendicular cl to points**

This option creates strings drawn perpendicular from reference alignments to selected points, and output a Special Chainage (SPC) file of the reference chainages that the points are perpendicular to. The user has the choice of either output, or both.

Typical uses are to create a model of cut strings which can be used to for cut labelling on long section plots where features such as property access locations occur along the road corridor, or to write SPC files that can be used to create cross sections through the dropped points when running **Apply MTF** functions.

Selecting **Draw perpendicular cl to points** displays the **Draw a Perpendicular from Centreline to a Point** panel.



The fields and buttons used in this panel have the following functions

Field Description	Type	Defaults	Pop-Up

Data Points To Drop

Gathers a list of all the points on the selected strings.

For a full description of data source go to 4.19.3 Data Source

Data Reference Strings

Selects the reference strings that the above points will be dropped onto. Reference strings must be of type Alignment or Super Alignment.

For a full description of data source go to 4.19.3 Data Source

Drop all points from chosen strings? tick box ticked

The default is to calculate chainages and/or strings from every point in the selected strings.

*Turning this tick box off will restrict the calculation to the end points only. e.g. It can be used to create cut strings for use with the **paired cuts** functionality in long section plotting.*

Offset Tolerance input

Will restrict the search to points that fall within the specified width from the reference string.

Outputs

Name mode	choice box	From point text	From point text, From string names, Typed

*For the **From point text** mode, the name of the new string is set by the text attached to the points.*

*For the **From string names** mode, the name of the new string is set by the selected strings.*

*For the **Typed** mode, the name is given in the* **Name** *field.*

Name	name box		names.4d file

*This only needs to be filled in when using the **Typed** name mode. This field is not required for the other* **Name mode** *choices.*

Model	model box		available models

The model to store the new strings on. This field is optional and no strings will be created if left blank. e.g. if you only want Special Chainage Files.

Colour	colour box		available colours

The colour to draw the new strings with.

Linestyle	linestyle box	1	available linestyles

The linestyle to draw the new strings with.

Z value mode	choice box	2d from point	2d from point, 2d from reference, 3d reference to point

The method by which z values are assigned to the points on the new string.

*For the **2d from point** mode, all points have the same z value as the point being dropped.*

*For the **2d from reference** mode, all points have the same z value as the reference string.*

*For the **3d reference to point** mode, the new string is drawn from the z value at the reference to the z value of the dropped point.*

Start extension	measure box	1	At point, Point to Point, String from Point, String to Point

End extension	measure box	1	At point, Point to Point, String from Point, String to Point
Write SPC Files(s)?	tick box	ticked	

The writing of Special Chainage Files is optional.

If ticked, the names of the files are automatically derived from the combination of Model name and Reference string name and a user defined file stem (given in the **SPC file stem** *field) may be added to the name.*

SPC File Stem text box

An optional user entered text value that is added to the automatically derived file name.

Process button

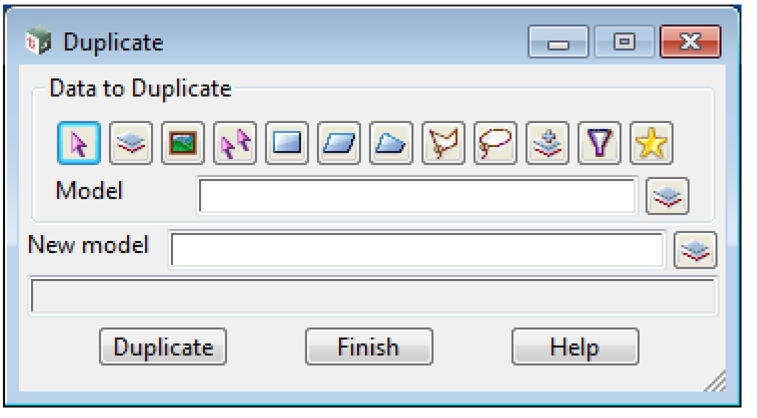
run the option.

28.9.16 Duplicate

Position of option on menu: **Utilities =>A-G =>Duplicate**

Duplicate makes duplicates of strings.

Selecting **Duplicate** displays the **Duplicate** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source

Data source	input		

source of data to be processed.

New model	input		available models

name of the model to place the duplicated strings into.

Duplicate	button		

duplicate all the strings in the data source.

<esc> can be used to abort the duplicate option.

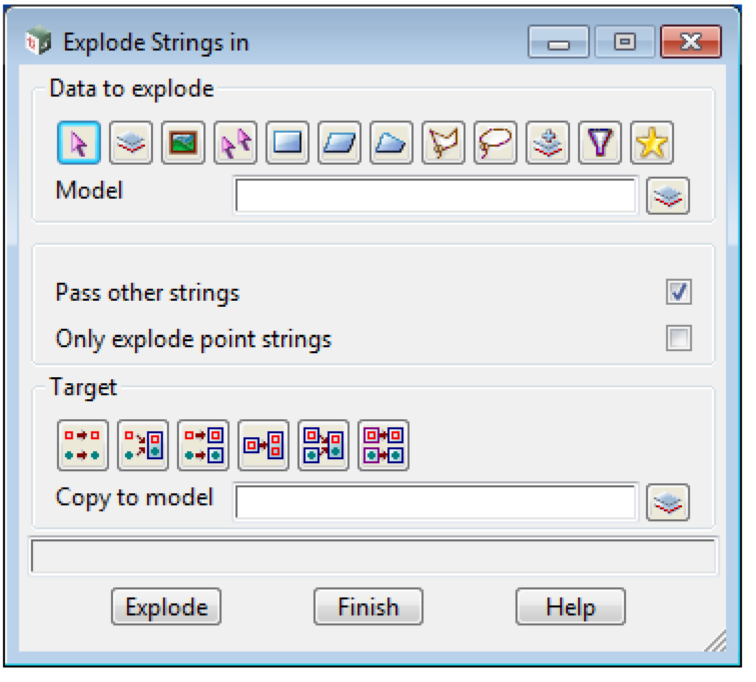
28.9.17 Explode

Position of option on menu: **Utilities =>A-G =>Explode**

The **Explode** option explodes point strings into individual one vertex strings and line strings into individual segments.

Super alignment, Alignment, pipeline, sewer and drainage strings are not exploded.

On selecting **Explode** and then the appropriate **Data Source** in the panel, the **Explode Strings in** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description Type Defaults Pop-Up

Data source type Model

data selection type - for a full description go to 4.19.3 Data Source

Data source input

source of data to be processed.

Pass other strings tick box tick

if ***ticked****, copies of all other strings and passed through to the exploded model.*

Only explode point strings tick box

if ***ticked****, only point strings are exploded. Line strings are left unexploded.*

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info input

extra information required for the target.

Explode button

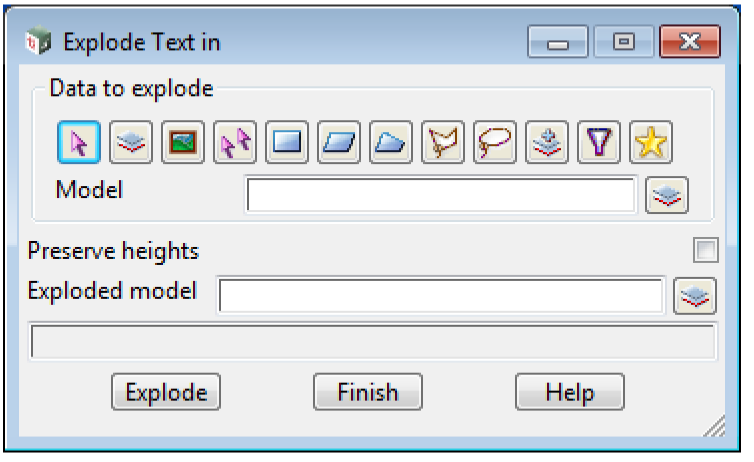
explode all the strings in the data source into one vertex strings and segments.

28.9.18 Explode Text

Position of option on menu: **Utilities =>A-G =>Explode (text)**

Explode (text) explodes text in text strings and the text from 4d strings, into its component arcs and lines. The exploded arcs and lines are placed in one new model.

Selecting **Explode (text)** displays the **Explode Text in** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source

Data source	input		

source of data to be processed.

Exploded model	input		available models

name of the model to place the exploded text strings and text of 4d strings into.

Explode	button		

explode all the selected text and add them to the model given in the **Exploded model** *field.*

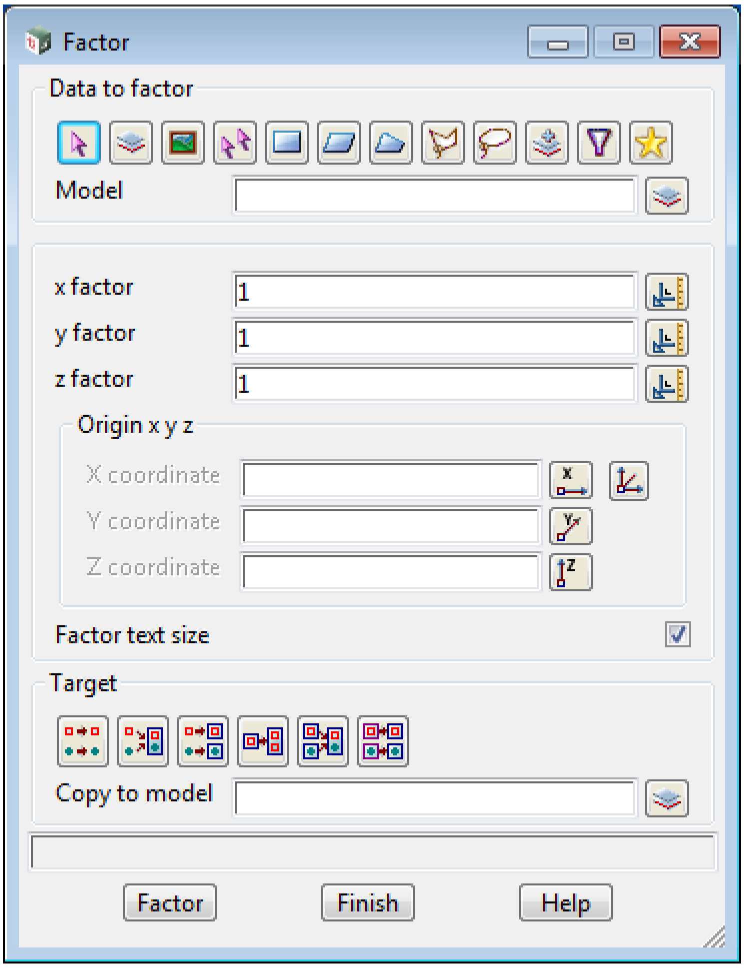
28.9.19 Factor

Position of option on menu: **Utilities =>A-G =>Factor**

Factor multiplies the x,y and z values of selected strings by user supplied factors.

The option is useful for changing the units of any data in a model (*e.g.* Imperial to metric).

Selecting **Factor** displays the **Factor** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source

Data source	input		

source of data to be processed.

x/y/z factor	input	1.0	

factor to multiply the x/y/z-values of a string by.

Origin x y z	input		**xyz ops** menu

field with x y z value to be used as (x,y,z) origin for the factor.
*if **blank**, an origin of (0,0,0) is used.*

Factor text size tick box tick

*if **ticked**, the size of text is factored.*

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info input

extra information required for the target.

Factor button

multiply the (x,y,z) values of the selected strings by the x,y and z factors given in the appropriate panel fields.

WARNING - if the x factor and y factor are not equal, then arcs in arcs, circles, alignment and polyline strings can not be factored. For these cases, the y factor will be set the x-factor and then applied to the arcs.

28.9.20 Filter

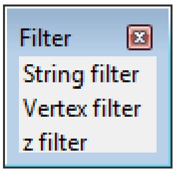
Position of menu: **Utilities =>A-G =>Filter**

The **Filter** options are used to remove surplus points from strings.

Three types of filters are currently supported in 12d Model:

- a string filter, which removes points from 2d and/or 3d line strings that do not deviate by more than a specified offset tolerance from straight lines joining successive string points (in three dimensions for 3d strings)
- a vertex-filter which removes adjacent vertices closer than a given user defined (x,y) and z distance
- a z-filter which eliminates points not contained within a specified z-range

The **Filter** walk-right menu is



filter 2d and 3d strings
removes close adjacent vertices
filter z values

For the option *String filter*, go to 28.9.20.1 String Filter
Vertex filter 28.9.20.2 Vertex Filter
z filter 28.9.20.3 Z Filter

28.9.20.1 String Filter

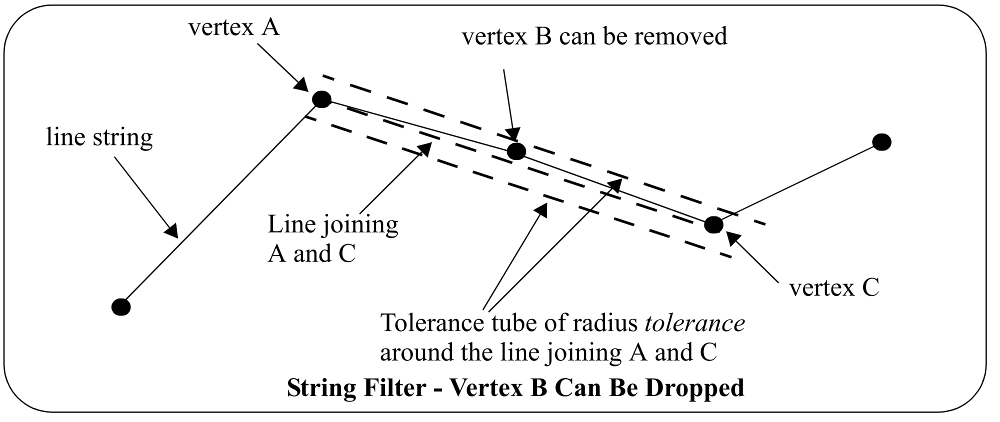
Position of option on menu: **Utilities =>A-G =>Filter =>String filter**

The **String filter** option is used to remove surplus vertices from 2d strings (contours) and 3d strings.

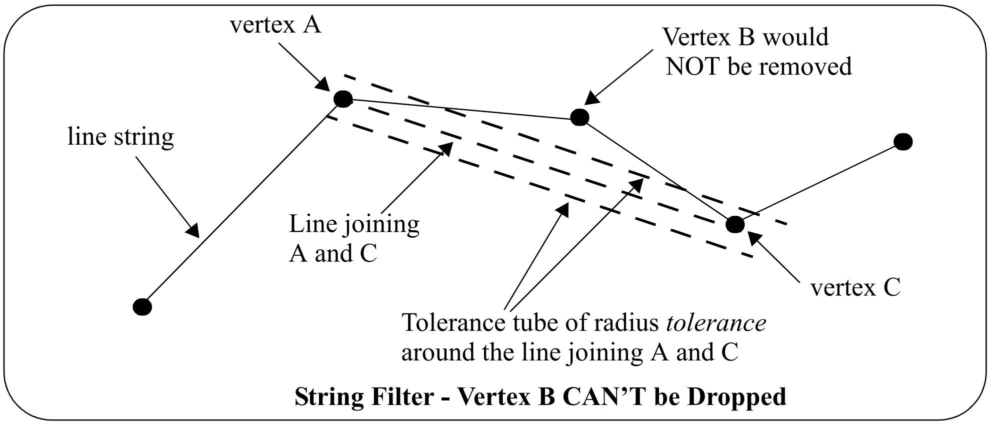
The string filter option tries to drop out vertices that if left out, don't make "too much difference".

So for a user defined tolerance:

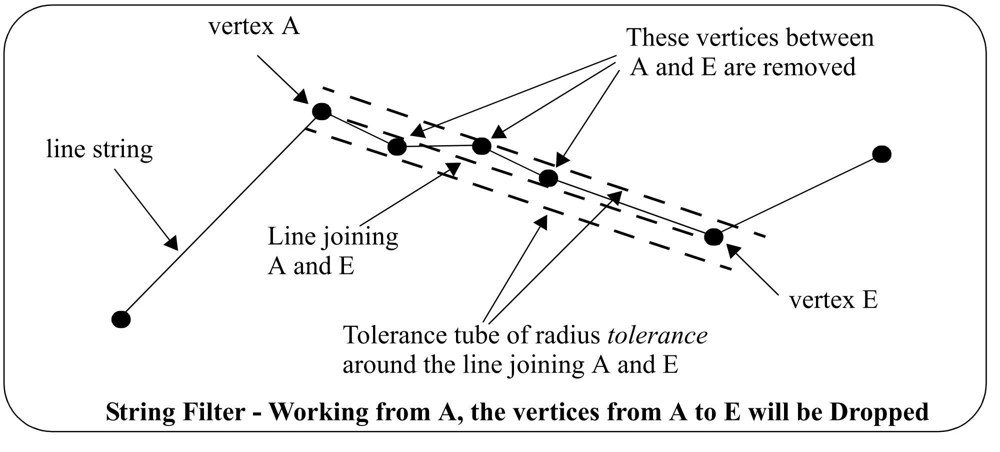
> if you have successive vertices A, B and C, and by leaving out vertex B, then the lines AB and BC don't go outside a tube of radius tolerance around the lines AC, then vertex B is dropped.



String Filter - Vertex B Can Be Dropped



String Filter - Vertex B CAN'T be Dropped



String Filter - Working from A, the vertices from A to E will be Dropped

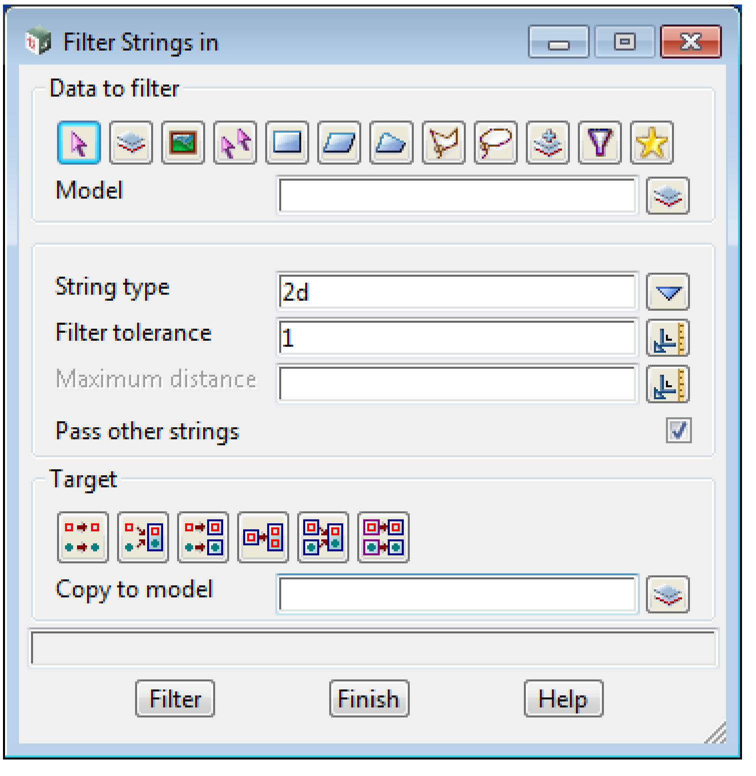
The filter starts at the first vertex of the string and looks ahead, dropping out vertices until it finds one that can't be removed. That vertex then become the new second vertex and the process is repeated but starting at the new second vertex. This process is repeated for the new third vertex and so on until the end of the string.

The *tolerance* value used normally depends on the data set and the job that the data is being used for.

So the string filter removes vertices from 2d and/or 3d line strings that do not deviate by more than a specified offset tolerance from straight lines joining successive string vertices (in three dimensions for 3d strings).

NOTE - string filer does not apply to point strings.

Selecting **String filter** displays the **Filter Strings in** panel.



The fields and buttons used in this panel have the following function

Field Description	Type	Defaults	Pop-Up
Data source type		Model	
data selection type - for a full description go to 4.19.3 Data Source			
Data source	input		
source of data to be processed.			
String type	input	2d	2d, 2d & 3d
string types to be filtered			
Filter tolerance	input	1	
tolerance to be used in filtering			
Pass other strings	tick box	tick	
*if **ticked**, pass all other strings through to the model for filtered strings*			
Target type			
Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target			

Target info input

extra information required for the target.

Filter button

filter all the selected strings.

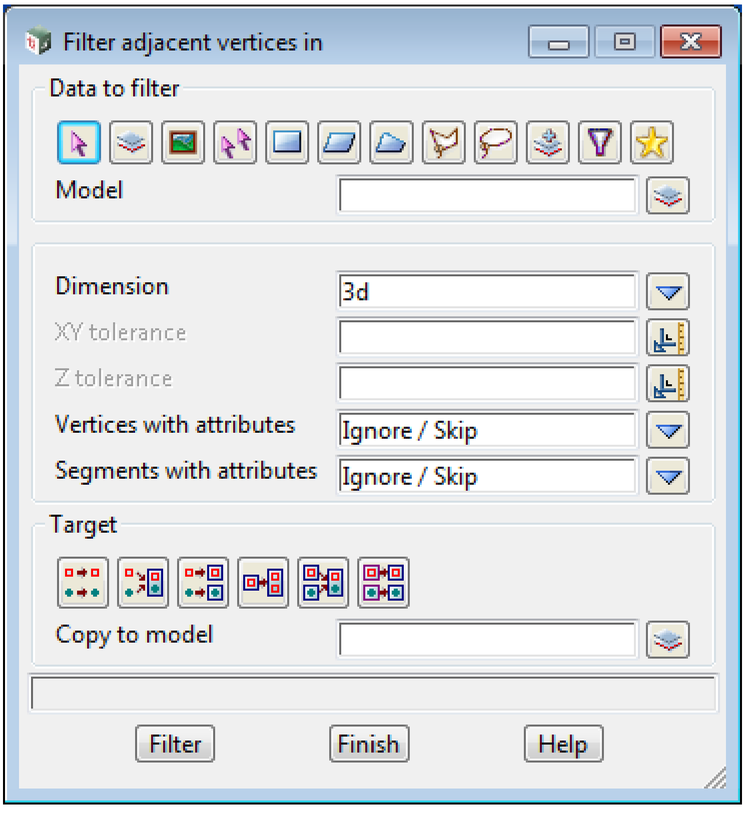
28.9.20.2 Vertex Filter

Position of option on menu: **Utilities =>A-G =>Filter =>Vertex filter**

The **Vertex filter** option is used to remove close adjacent vertices from selected strings. The strings can be *point* or *line* strings.

The option can remove adjacent string vertices that are equal to a given tolerance either in plan position only (*i.e.* have similar x and y co-ordinates) or equal to a given tolerance for x, y and z co-ordinates.

Selecting **Vertex filter** displays the **Filter adjacent vertices in** panel.



The fields and buttons used in this panel have the following function

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source

Data source	input		

source of data to be processed.

Dimension	input	2d	2d,3d

*if **2d**, adjacent string vertices with equal x and y co-ordinates are filtered.*
*If **3d**, adjacent string vertices with equal x, y and z co-ordinates are filtered.*

xy tolerance	input		

the distance to use for checking if adjacent vertices are too close in (x,y)

z tolerance	input		

the delta-z to use for checking if adjacent vertices are too close in z

Vertices with attributes	choice box	Ignore/skip	Ignore/skip, Merge attributes

Lose attributes

*if **Ignore/skip**, don't drop any vertices with attributes.*

*If **Merge attributes**, merge the attributes from a dropped vertex with the kept vertex.*

*If **Lose attributes**, drop the attributes from a dropped vertex.*

Segments with attributes choice box Ignore/skip Ignore/skip, Merge attributes Lose attributes

*if **Ignore/skip**, don't drop any vertices of segments with attributes.*

*If **Merge attributes**, merge the attributes from a dropped segment with the kept segment.*

*If **Lose attributes**, drop the attributes from a dropped segment.*

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info input

extra information required for the target.

Filter button

filter the selected strings.

28.9.20.3 Z Filter

Position of option on menu: **Utilities =>A-G =>Filter =>Z filter**

It is often necessary to remove data which is not within a certain z range. For example, all points with z-values below 0.0. The **z filter** option allows the user to define a z-range by specifying a minimum and maximum z value.

Strings of type 2d, 3d and 4d can then be processed against the z-range and new strings created containing only those points that are

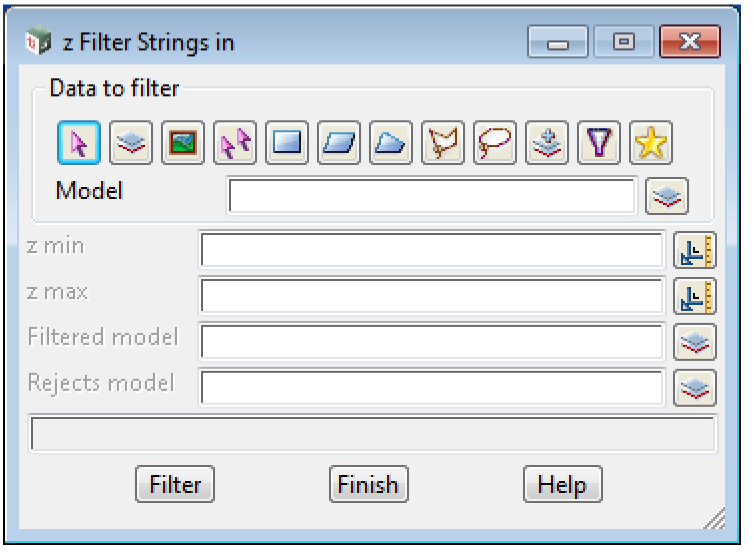
(a) within the z-range - accepted points

or

(b) outside the z-range - rejected points

For all other string types, the entire string is accepted if any part of the string is within the z-range, otherwise it is rejected.

Selecting **Z filter** displays the **Z Filter Strings in** panel:



The fields and buttons used in this panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Data source type | | Model |

data selection type - for a full description go to 4.19.3 Data Source

Data source | input

source of data to be processed.

z min | input

minimum z value to be accepted, If this field is blank, all string z-values pass the minimum test.

z max | input

maximum z value to be accepted, If this field is blank, all string z-values pass the maximum test.

Filtered model | input | | available models

if non-blank, the name of the model to place the accepted string points in. The string has the same name as the original string. If blank, the accepted point strings are not saved.

Rejects model | input | | available models

if non-blank, then strings containing the rejected points are placed in this model. The strings of rejected points have the same name as the original strings. If blank, the rejected point strings are not saved.

Filter button

z filter the selected strings by the z minimum and z maximum values.

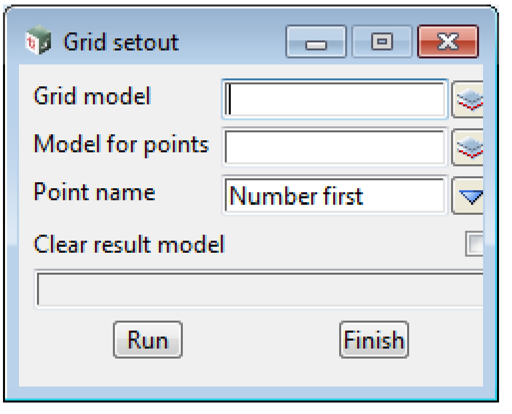
28.9.21 Grid setout

Position of option on menu: **Utilities =>A-G =>Grid setout**

Grid setout creates a set of points from the intersections of lines and/or arc strings. The strings to intersect normally form a grid.

It is assumed that the strings to intersect are named with letters along one axis of the grid and numbers in the other so that a point may be named in the following manner: 1a or a1.

Selecting **Grid setout** displays the **Grid Setout** panel.



The fields and buttons used in this panel have the following function

Field Description	Type	Defaults	Pop-Up
Grid model	model box		Select Model

The model of strings to intersect with each other

Field Description	Type	Defaults	Pop-Up
Model for points	model box		Select Model

the model to contain the points of intersection for the strings in **Grid model**.

Field Description	Type	Defaults	Pop-Up
Point name	choice box	Number first	Number first, Letter first, Any order

Method used to determine how to name the created points based on names of the strings being intersected:

Number first: use the numbered string first

Letter first: use the alphabetical string first

Any order: use the order in which the strings are processed

Field Description	Type	Defaults	Pop-Up
Clear result model	tick box		

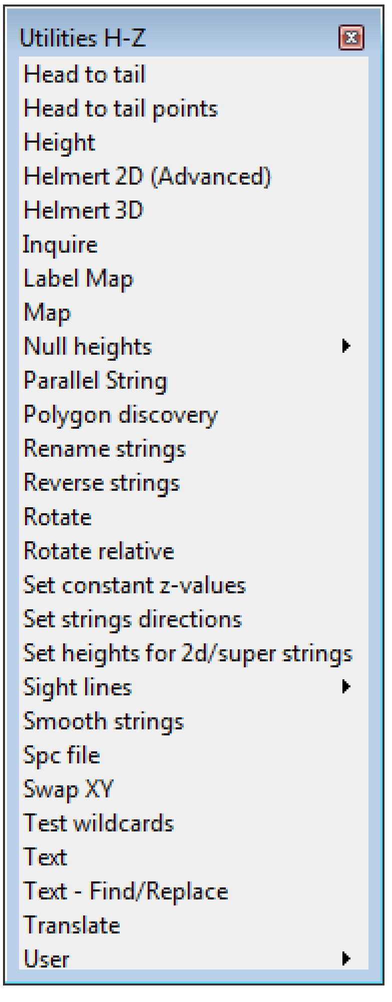
if tick, clean out the **Model for points** *before finding the intersections.*

Field Description	Type	Defaults	Pop-Up
Run	**button**		

run the option.

28.10 Utilities H-Z

Position of menu: **Utilities =>H-Z**



join strings with common ends
join one point strings into multi-point strings
give strings new height
2D Helmert, also allowing fixed scale
3D Helmert transformation
highlight selected strings
apply a Label Map File on strings
apply a Map File on strings
set heights to null value
parallel a string
find enclosing polygon from strings surrounding selected point
rename strings
reverse string directions
rotate strings about one point
various methods of rotating strings
give strings a constant z value
give strings same direction as selected string
set heights for 2d/3d super strings
options using sight lines
smooth strings
create special chainage points file
swap X and Y coordinates of strings
testing pattern and expression matching
modify text parameters
find and replace text
translate strings in model/view
User Utilities H-Z menu

For the option *Head to tail*, go to		28.10.1 Head to Tail
	Head to tail points	28.10.2 Head to Tail Point Strings
	Height	28.10.3 Set Height
	Helmert 2D (Advanced)	28.10.4.1 2D Helmert (Advanced)
	Helmert 3D	28.10.4.2 3D Helmert
	Inquire	28.10.5 Inquire
	Label map	8.7.2 Apply a Label Map File
	Map	8.8.5 Apply Map File
	Null heights	28.10.6 Null Heights
	Parallel	28.10.7 Parallel String
	Polygon discovery	28.10.8 Polygon Discovery
	Rename strings	28.10.9 Rename Strings
	Reverse strings	28.10.10 Reverse Strings
	Rotate	28.10.11 Rotate
	Rotate Relative	28.10.12 Rotate Relative
	Set constant z-values	28.10.13 Set Constant Z-Values
	Set strings directions	28.10.14 Set Strings Directions

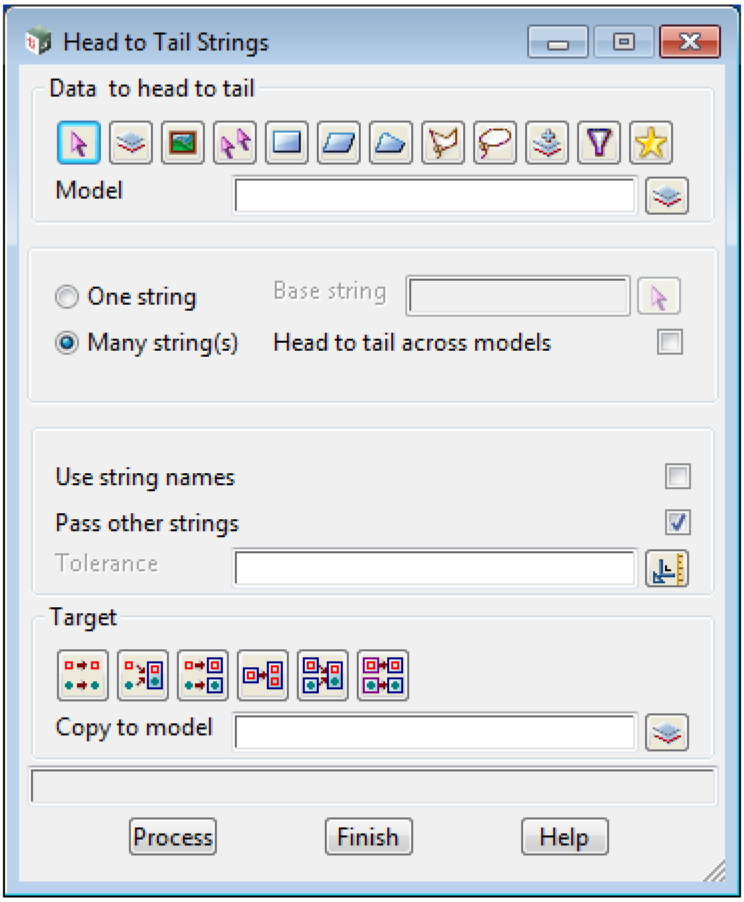
28.10.1 Head to Tail

Position of option on menu: **Utilities =>H-Z =>Head to tail**

Head to tail is used to join strings with common end points together.

If requested, strings are only joined if they have matching names as well as matching end points.

Selecting **Head to tail,** displays the **Head to Tail Strings** panel.



The fields and buttons used in this panel have the following functions

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source

Data source input

source of data to be processed.

One string or **Many strings** radio button

*if **One String** is selected, then a Base string is selected and only the head-to-tailed string containing the Base string is created.*

*If **Many strings** is selected, then head to tailing occurs for all the strings in the Data source.*

Head to tail across models tick box

*if **ticked**, then strings can be created from head to tailing strings from any of the selected models.*

*If **not ticked**, then strings will only be head to tailed with strings from the same model.*

Use string names tick box

*if **ticked**, the string names must also match for the strings to be joined.*

Pass other strings tick box tick

*if **ticked**, any string not joined will also copied to the joined model.*

Tolerance input 0.0005

If the distance between two end points is less than **tolerance***, then the points are considered the same and the strings may be joined.*

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info input

extra information required for the target.

Process button

process all the selected strings

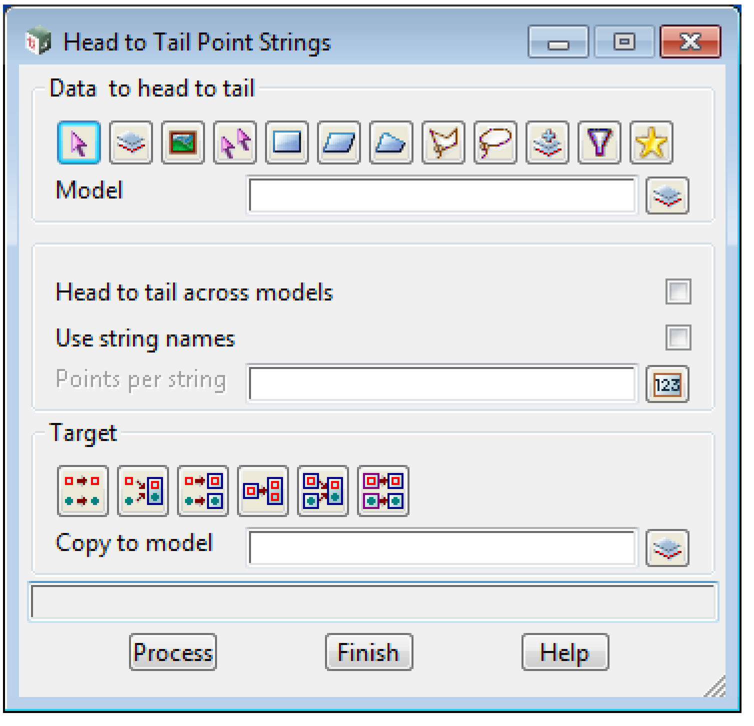
28.10.2 Head to Tail Point Strings

Position of option on menu: **Utilities =>H-Z =>Head to tail point strings**

Head to tail point string is used to joins point strings into larger point strings. This can speed up drawing and processing when there are large numbers of single point strings.

If requested, strings are only joined if they have matching names as well as being point strings.

Selecting **Head to tail points** displays the **Head to Tail Point Strings** panel.



The fields and buttons used in this panel have the following functions

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source

Data source input

source of data to be processed.

Head to tail across models tick box

*if **ticked**, then strings can be created from joining point strings from any of the selected models.*

*If **not ticked**, then strings will only joined with strings from the same model.*

Use string names tick box

*if **ticked**, the string names must also match for the strings to be joined.*

Points per string integer box

the maximum number of vertices to have in the joined strings.

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info input

extra information required for the target.

Process button

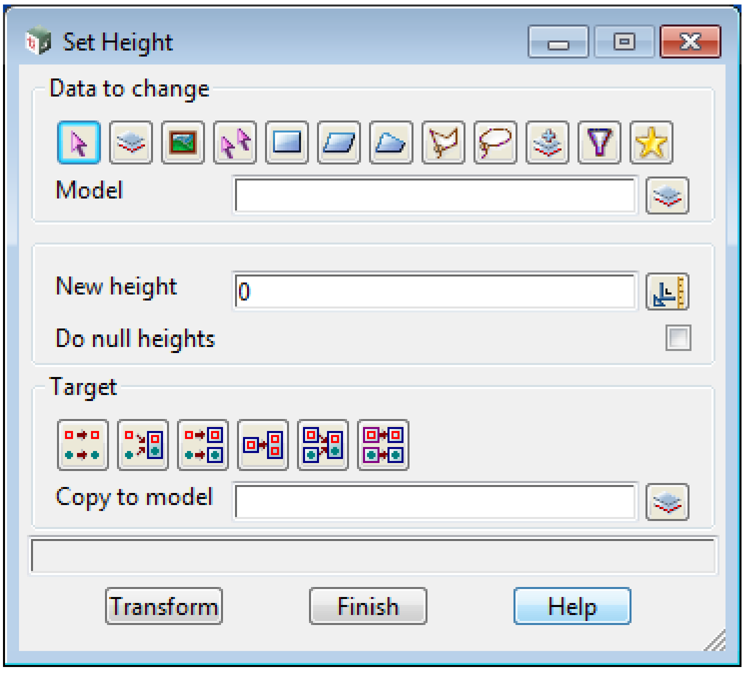
process all the selected point strings

28.10.3 Set Height

Position of option on menu: **Utilities =>H-Z =>Height**

For the selected strings, **Height** will set all the z-values to the given height.

Selecting **Height** displays the **Set Height** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source

Data source	input		

source of data to be processed.

New height	input	0	

the new height to set all the z-values in the selected strings to.

Do null heights	tick box		

*if **ticked**, null heights are set to the new height.*
*If **not ticked**, null heights are not modified.*

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info	input		

extra information required for the target.

Set	button		

process the selected string

28.10.4 Helmert Transformations

See 28.10.4.1 2D Helmert (Advanced)
28.10.4.2 3D Helmert

28.10.4.1 2D Helmert (Advanced)

Position of option on menu: **Utilities =>H-Z =>Helmert 2D (Advanced)**

A ***2D Helmert*** transformation is a two dimensional linear transformation consisting of a scaling, rotation and a 2D- translation (shift) of data. That is, there are **four** parameters to be specified.

Hence the 2D Helmert parameters are

(a) the one scale factor for both the x and y axes

(b) the same rotation of the x-axis and y-axis of the existing coordinates to align it with the control x and coordinate axis

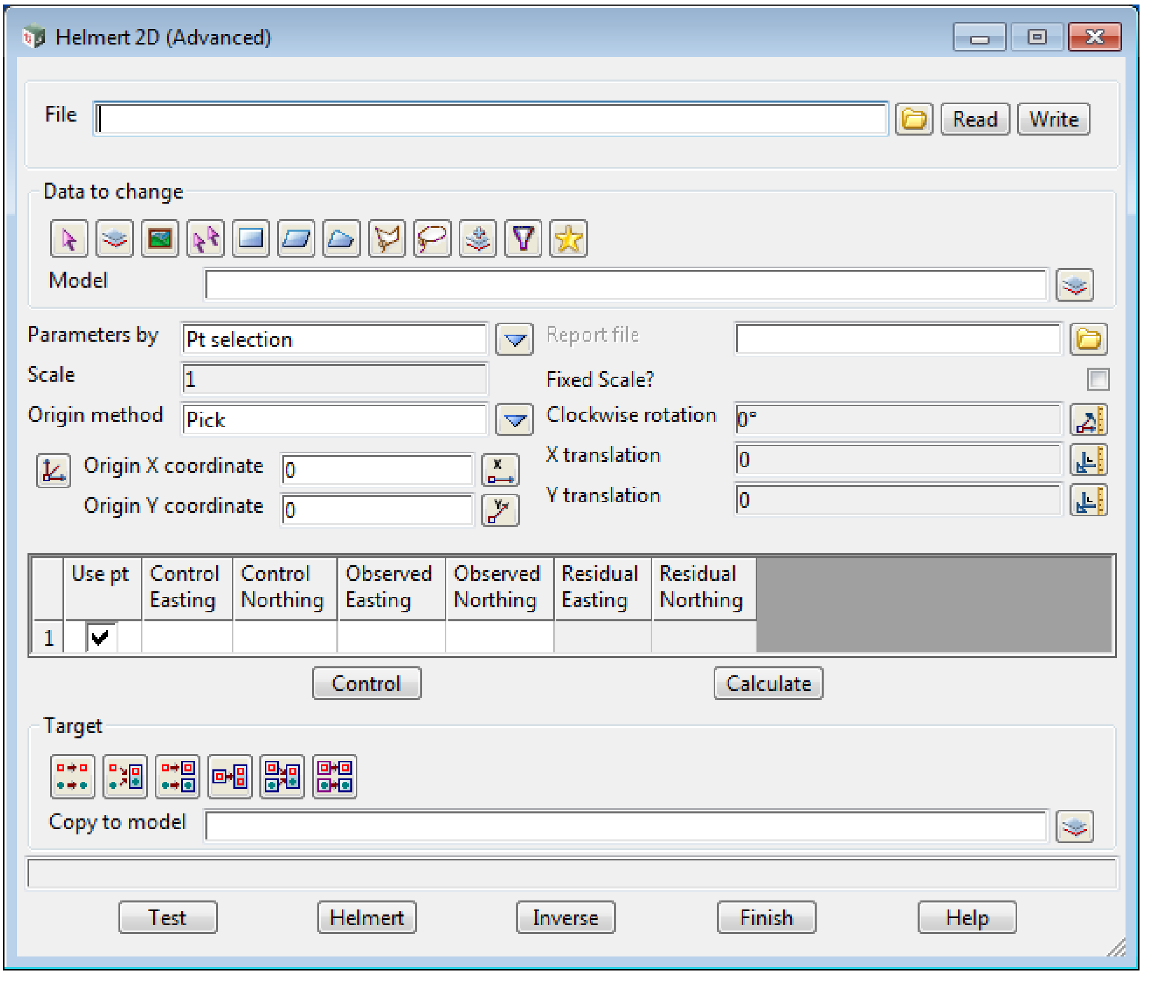
(c) the x and y translation of the observed coordinates after the scalings and rotations have been applied

The **2D Helmert (Advanced)** option calculated and applied a Helmert transformation, and also allows the user to ***fix the scale*** for the transformation. That is, the scale is user defined and not determined by the least squared calculations.

For information on the 2D Helmert transformation, please go to the section 28.9.1 Coordinate Transformations - Helmert and Affine

For information on how the **Affine Transformation works with Rasters,** go to the section 28.9.1.2 Affine and Helmert Transformations Applied to Rasters

Selecting **2D Helmert (Advanced)** brings up the **Helmert 2D (Advanced)** panel.



If the **Parameters by** choice box is set to ***Pt Selection*** then control-observed point pairs are selected by the user and the pairs are written to the grids on the panels. See the section 28.9.1.1 Selecting Control and Observed Points for the Helmert and Affine Transformations for information on picking control and observed points.

If the **Parameters by** choice box is set to **Direct entry**, the transformation parameters are manually entered into the panel and the grid section is not used.

The fields and buttons used in this panel have the following functions

Field Description	Type	Defaults	Pop-Up
File	file box		*.hel_adv, *.hel files

the name of a 2d Helmert parameter file to read in/write out

Read button

read the contents of the .hel_adv (or .hel) file given in **File** *and load the parameters saved in the file into the panel fields.*

Note*: the *.hel file format is from an earlier Helmert panel that no longer exists in* ***12d Model****.*

Write button

write the parameters and data in the panel fields to the file given in **File**.

Data source type Model

data selection type - for a full description go to 4.19.3 Data Source

Data source	input		

source of data to be processed.

Parameters by	choice	Pt selection	Pt selection Direct entry

the parameter entry method.

Report file			*.rpt files

if not blank, write a report on the Helmert transformation.

Scale input/output

the scale factor for the existing coordinates with respect to the transformed coordinates

Fixed scale? tick box

*if ticked, the scale is entered into the **Scale** field and it is held fixed when the Helmert parameters are calculated from the selected control-observed point pairs.*

Clockwise rotation input/output

the rotation of the axes of the existing points with respect to the transformed axes.

Origin method	choice box	Pick	Pick, Obs'd centroid Ctrl centroid

the translation can be reported against another origin than the default (0,0).

Origin X/Y coordinate input

the translation parameters are written out with respect to this origin.

X Translation input/output

the x translation of the existing coordinates with respect to the transformed x coordinates

Y Translation input/output

the y translation of the existing coordinates with respect to the transformed y coordinates

Control button

restarts the selection process for choosing more control points.

Calculate button

calculate the helmert parameters from the selected points shown in the grid.

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target.

Target info input

extra information required for the target.

Helmert button

apply the helmert transformation to the data specified in the source box, and put it into the appropriate target area.

Inverse button

applies the inverse transformation to the one defined in the panel.

Finish button

end the option, remove the panel from the screen.

28.10.4.2 3D Helmert

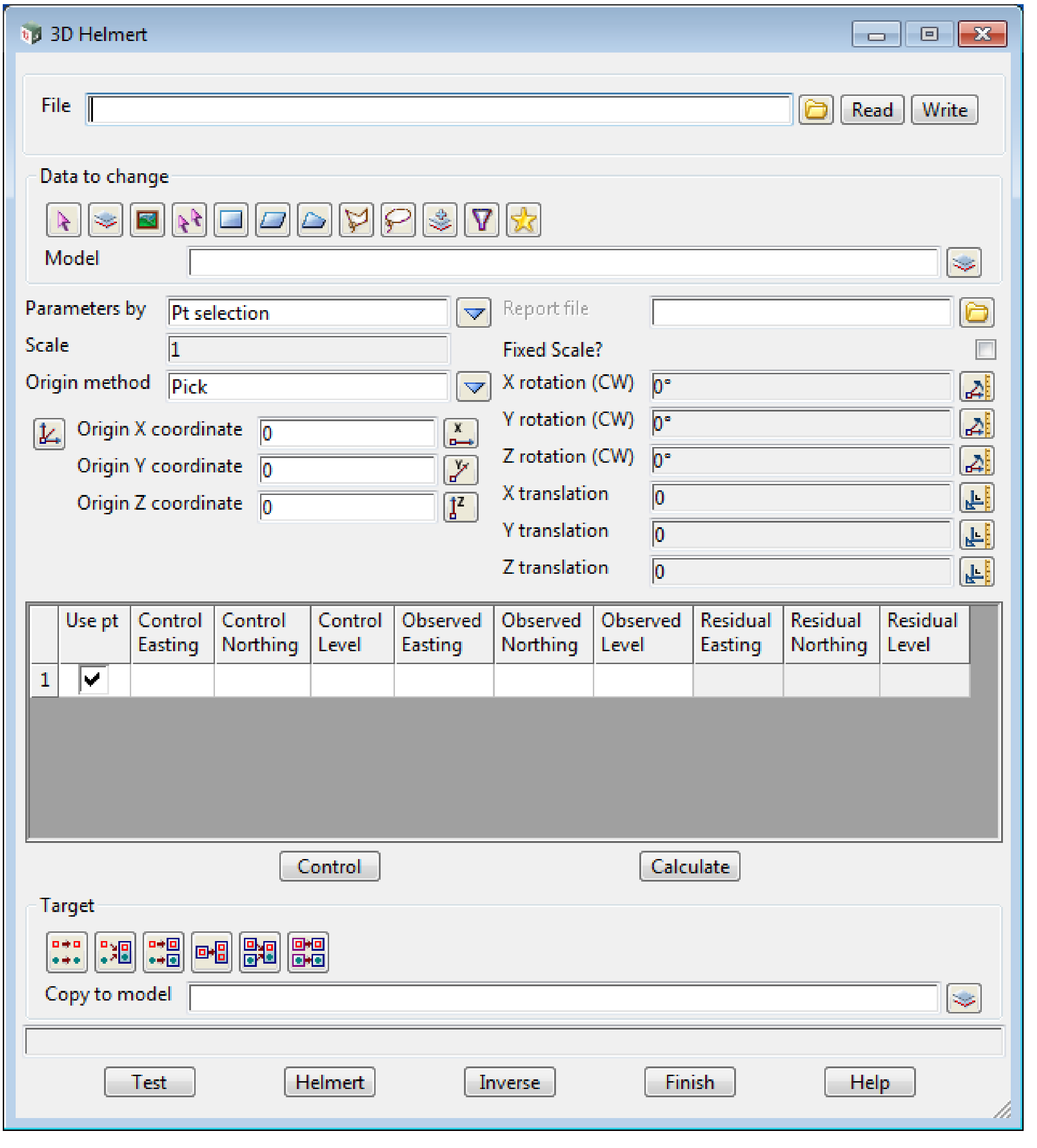
Position of option on menu: **Utilities =>H-Z =>Helmert 3D**

A ***3D Helmert*** transformation is a three dimensional linear transformation consisting of a scaling, three rotations, and a 3D-translation (shift) of data. Hence there are 7 parameters to be specified.

For information on the 3D Helmert transformation, please go to the section 28.9.1 Coordinate Transformations - Helmert and Affine

For information on how the **Affine Transformation works with Rasters,** go to the section 28.9.1.2 Affine and Helmert Transformations Applied to Rasters

Selecting **Helmert 3D** brings up the **3D Helmert** panel.



If the **Parameters by** choice box is set to ***Pt Selection*** then control-observed point pairs are selected by the user and the pairs are written to the grids on the panels. See the section 28.9.1.1 Selecting Control and Observed Points for the Helmert and Affine Transformations for information on picking control and observed points.

If the **Parameters by** choice box is set to **Direct entry**, the transformation parameters are manually entered into the panel and the grid section is not used.

The fields and buttons in the panel have the following functions

Field Description	Type	Defaults	Pop-Up
File	file box		*.hel_3d

the name of a 3d Helmert parameter file to read in/write out

Read button

read the contents of the .hel_3d file given in **File** *and load the parameters saved in the file into the panel fields.*

Write button

write the parameters and data in the panel fields to the file given in **File**.

Data source type — Default: Model

data selection type - for a full description go to 4.19.3 Data Source

Data source input

source of data to be processed.

Field Description	Type	Defaults	Pop-Up
Parameters by	choice	Pt selection	Pt selection Direct entry

the parameter entry method.

Report file — Pop-Up: *.rpt files

if not blank, write a report on the 3D Helmert transformation.

Scale input/output

the scale factor for the existing coordinates with respect to the transformed coordinates

Fixed scale? tick box

if ticked, the scale is entered into the ***Scale*** *field and it is held fixed when the Helmert parameters are calculated from the selected control-observed point pairs.*

Field Description	Type	Defaults	Pop-Up
Origin method	choice box	Pick	Pick, Obs'd centroid Ctrl centroid

the translation can be reported against another origin than the default (0,0,0).

Origin X/Y/Z coordinate input

the translation parameters are written out with respect to this origin.

X rotation (Clockwise rotation

the rotation in the (y,z) plan about the x-axis of the axes of the existing points with respect to the transformed axes.

Y rotation (Clockwise rotation

the rotation in the (z,x) plane of the axes of the existing points with respect to the transformed axes.

Z rotation (Clockwise rotation

the rotation in the (x,y) plane of the axes of the existing points with respect to the transformed axes.

X/Y/Z Translation input/output

the x/y/z translation of the existing co-ordinate with respect to the transformed x/y/z coordinates

Control button

restarts the selection process for choosing more control points.

Calculate button

calculate the helmert parameters from the selected points shown in the grid.

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info input

extra information required for the target.

Helmert button

apply the Helmert transformation to the data specified in the source box, and put it into the appropriate target area.

Inverse button

applies the inverse transformation to the one defined in the panel.

Finish button

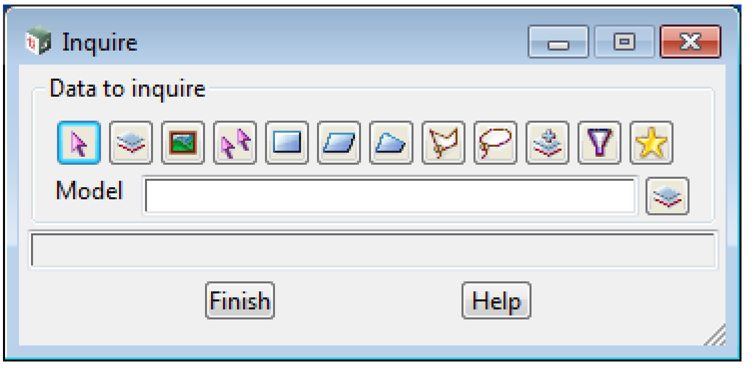
end the option, remove the panel from the screen.

28.10.5 Inquire

Position of option on menu: **Utilities =>H-Z =>Inquire**

Inquire highlights all the selected strings.

On selecting the **Inquire** option, the **Inquire** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source

Data source	input		

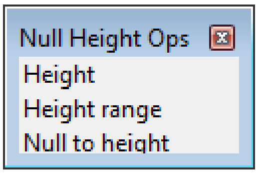
source of data to be highlighted.

28.10.6 Null Heights

Position of option on menu: **Utilities =>H-Z =>Null heights**

A null value is used as a z-value (height) when no actual z-value exists at a vertex. That is, the vertex has valid x and y coordinates but no valid z-value.

The **null heights** walk-right menu is



For the option *Height*, go to 28.10.6.1 Heights

Height range 28.10.6.2 Height Range

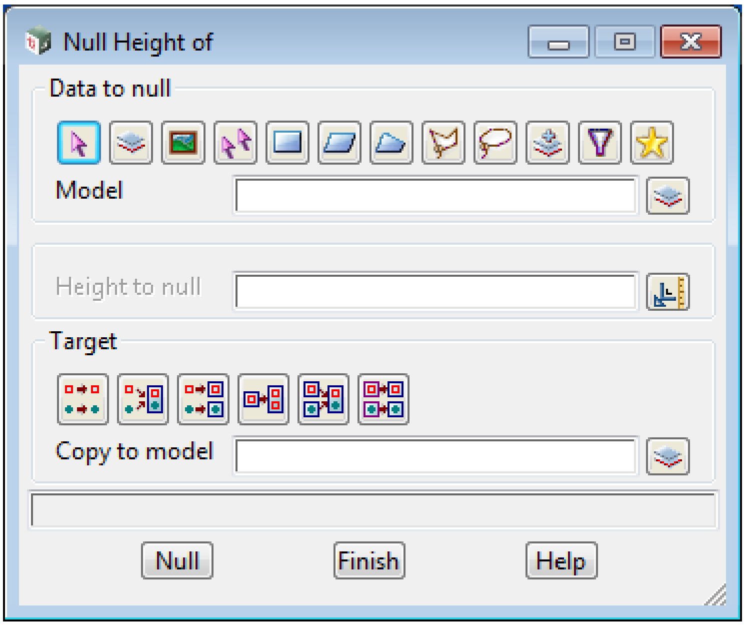
Null to height 28.10.6.3 Null to Height

28.10.6.1 Heights

Position of option on menu: **Utilities =>H-Z =>Null heights =>Height**

Null heights is used to set vertices with a given z-value to ***12d Model***'s null value.

Selecting **Heights** displays **Null Height of** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source

Data source	input		

source of data to be processed.

Height to null	input		

height value to set to the null value.

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info	input		

extra information required for the target.

Null	button		

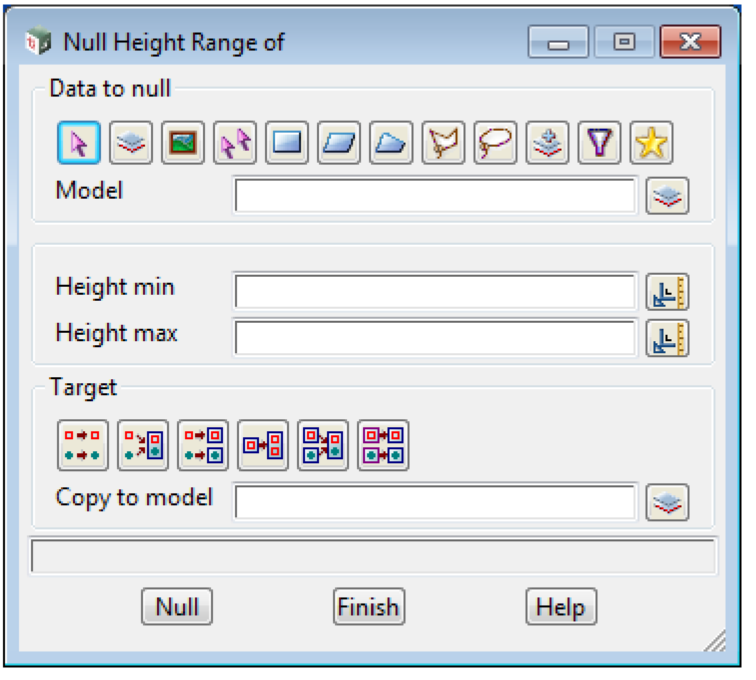
test all the z-values in the selected strings and if the value is equal to the **Height to null** *value, set the z-value to null.*

28.10.6.2 Height Range

Position of option on menu: **Utilities =>H-Z =>Null heights =>Null range**

Null height range sets the vertices with z-values between a given height minimum and height maximum, to ***12d Model***'s null value.

Selecting **Height range** displays the **Null Height Range of** panel.



The fields and buttons used in this panel have the following functions.

Field Description | Type | Defaults | Pop-Up

Data source type | | Model |

data selection type - for a full description go to 4.19.3 Data Source

Data source | input

source of data to be processed.

Height min | input

minimum value of the heights to be set as null values.

Height max | input

maximum value of the heights to be set as null values.

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info | input

extra information required for the target.

Null | button

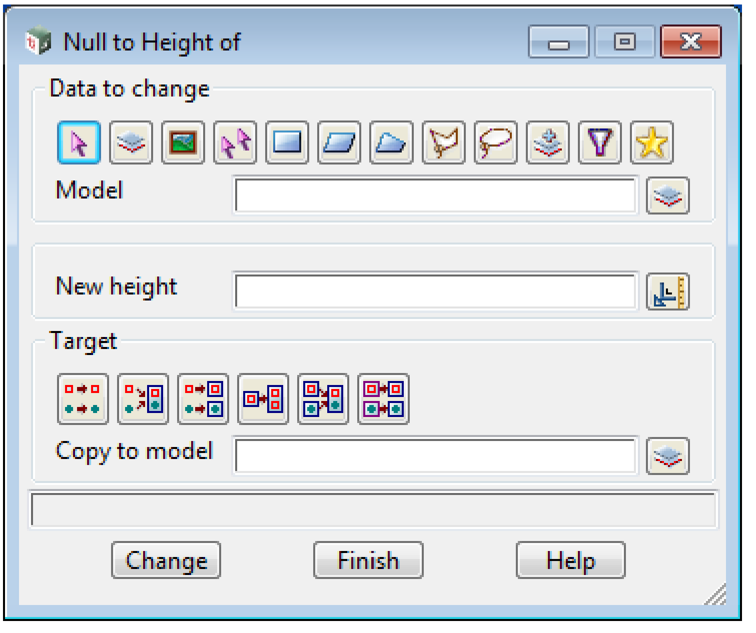
test all the z-values in the selected strings and if the value is between the **Height min** *and* **Height max** *value, set the z-value to null.*

28.10.6.3 Null to Height

Position of option on menu: **Utilities =>H-Z =>Null heights =>Null to height**

Null to height is used to set vertices with a null z-value to a user given value.

Selecting **Null to height range** displays the **Null to Height of** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source

Data source input

source of data to be processed.

New height input

height value to change null values to.

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info input

extra information required for the target.

Change button

Change all the null z-values in the selected strings to the value in the **New height** *field.*

28.10.7 Parallel String

Position of option on menu: **Utilities =>H-Z =>Parallel string**

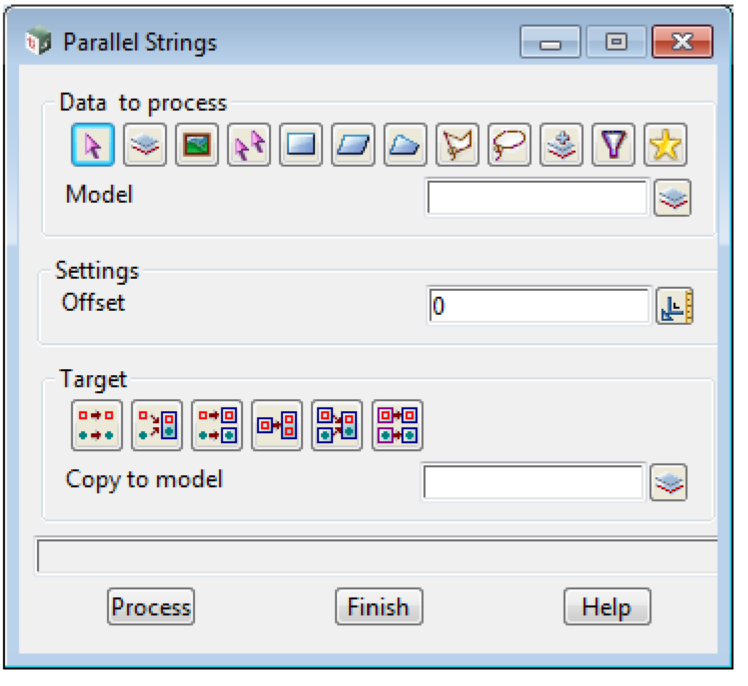
This option parallels horizontally one or more strings (the option **Strings => Strings edit => Parallel** only parallels one string at a time).

It is a macro and runs in two ways*:*

1. from the menu or
2. via a command line interface that does not require user interaction.

Note: the command line interface is available so that the option can be run in a chain if the macro can not be recorded in a chain. If the macro is called without any command line arguments, then it will assume it is being run in the graphical, interactive manner and will display a panel to the user. Otherwise, the macro will parse the relevant command line arguments and run without user interaction.

Selecting **Parallel** brings up the **Parallel Strings** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	
data selection type - for a full description go to 4.19.3 Data Source			
Data source	input		
source of data to be processed.			
Offset	measure box	0	available measures
the distance to parallel the source element.			
The direction of the source string is used to determine the left and right side of the string in the parallel operation. A positive offset parallels the string to the right with respect to the string direction. A negative offset parallels to the left with respect to the string direction.			
Target type			

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info input

extra information required for the target.

Process button

processes the selected data.

Note When Using the Option With Command Line Arguments:

When calling the macro using the command line interface, it must be called according to the following format:

parallel_strings_panel.4do “SOURCE MODEL” offset “TARGET MODEL”

The first argument passed must be the name of a model containing the data to process. The source model of data to process must be the first argument provided to the macro.

The second argument must be the offset distance to parallel the source strings. This can be negative or positive, but it must be a valid real number (*e.g.* 1.234, -567.00, 0).

The third and final argument must be the name of the model in which to place the resultant paralleled strings.

If any of the fields contain spaces, they must be enclosed in quotes
For example: “SOURCE DATA”

When using the command line interface, there must be 3 and only 3 arguments. Any fewer or more will be considered an error and will stop the macro from progressing any further.

Refer to the description of the fields in the Graphical User Interface section for further details, including restrictions on fields and expected behaviour

The macro uses the **Parallel()** function made available through the macro language. As such, the string types supported and behaviour of the macro is dependent on the implementation of that function.

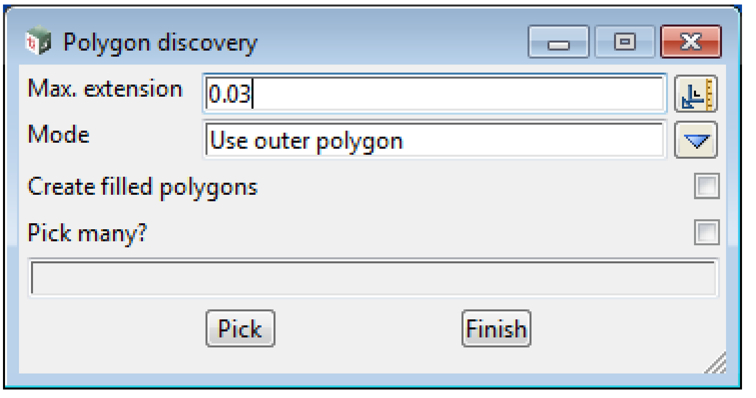
28.10.8 Polygon Discovery

Position of option on menu: **Utilities =>H-Z =>Polygon discovery**

Polygon discovery takes a selected position/point on a plan view and from the data displayed on the view, finds the polygon enclosing the selected position/point from the line and arc segments in strings around the position/point.

The name, colour and model for the created polygon are taken from the *CAD* Tool bar.

Selecting **Polygon discovery** brings up the **Polygon Discovery** panel.

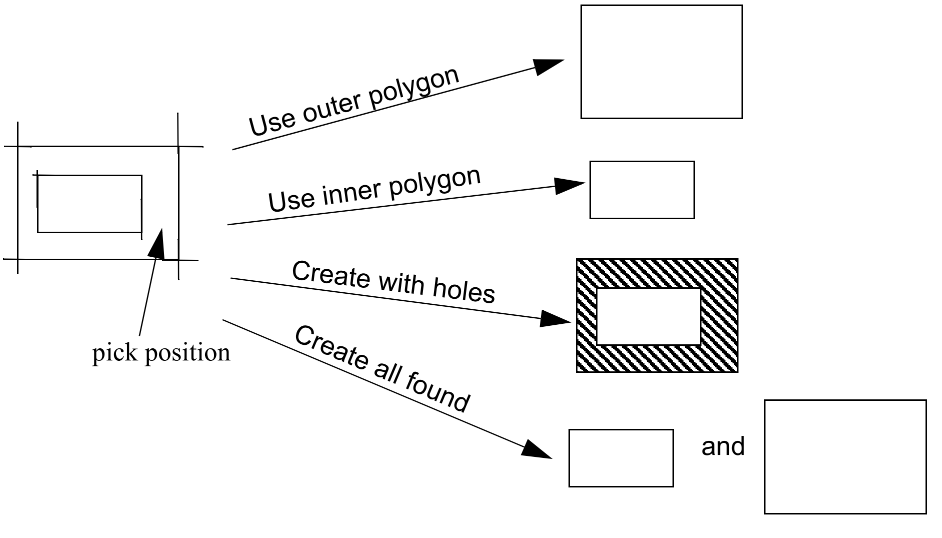


The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Max. extension	real value box	0.03	measures

maximum distance to extend segments to get an intersection to help define the surrounding polygon.

Field Description	Type	Defaults	Pop-Up
Mode ,	choice box	Use outer polygon	Use outer polygon User inner polygon Create with holes Create all found



Field Description	Type	Defaults	Pop-Up
Create filled polygons	tick box		

if ticked, the created polygons are given a fill colour.

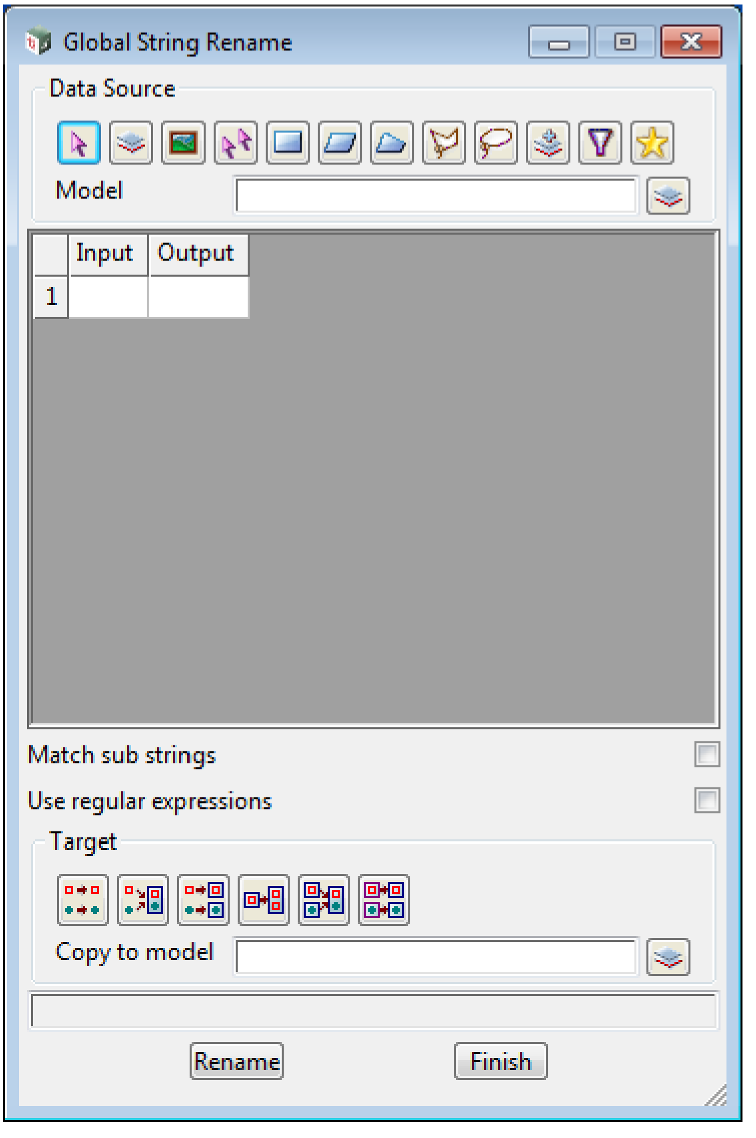
Pick button

pick the position/point that will be used to find an enclosing polygon for.

28.10.9 Rename Strings

Position of option on menu: **Utilities =>H-Z =>Rename strings**

Selecting **Rename strings** brings up the **Global String Rename** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source

Data source input

source of data to be processed.

Input

expression to match the selected string names against.

If a match occurs then the string name is modified as per the **Output** *field.*

Output

expression for renaming the strings that matched **Input**.

Match sub strings tick box not ticked

if ticked, only a part of the string name needs to match the expression in **Input**.

Use regular expressions tick box not ticked

if ticked, **Input** *is a regular expression.*

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info input

extra information required for the target.

Rename button

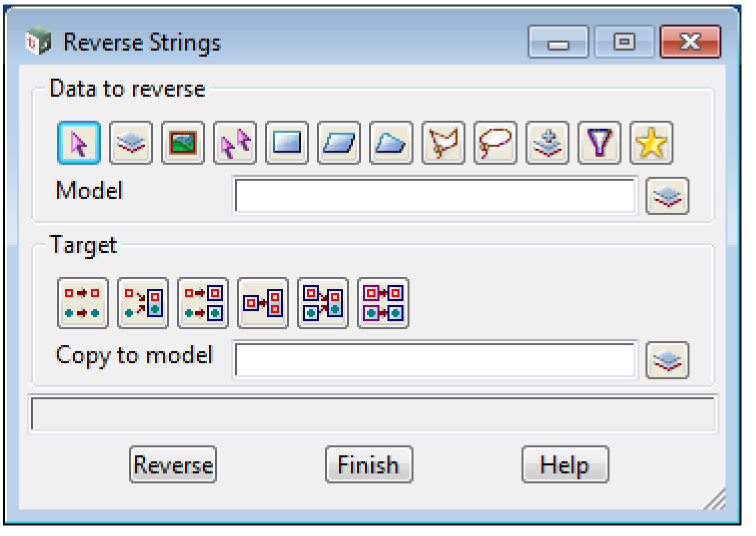
rename the selected data.

28.10.10 Reverse Strings

Position of option on menu: **Utilities =>H-Z =>Reverse strings**

Reverses the direction of the selected strings.

Selecting **Reverse strings** brings up the **Reverse Strings** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source

Data source	input		

source of data to be processed.

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info	input		

extra information required for the target.

Reverse	button		

reverse the direction of the selected strings.

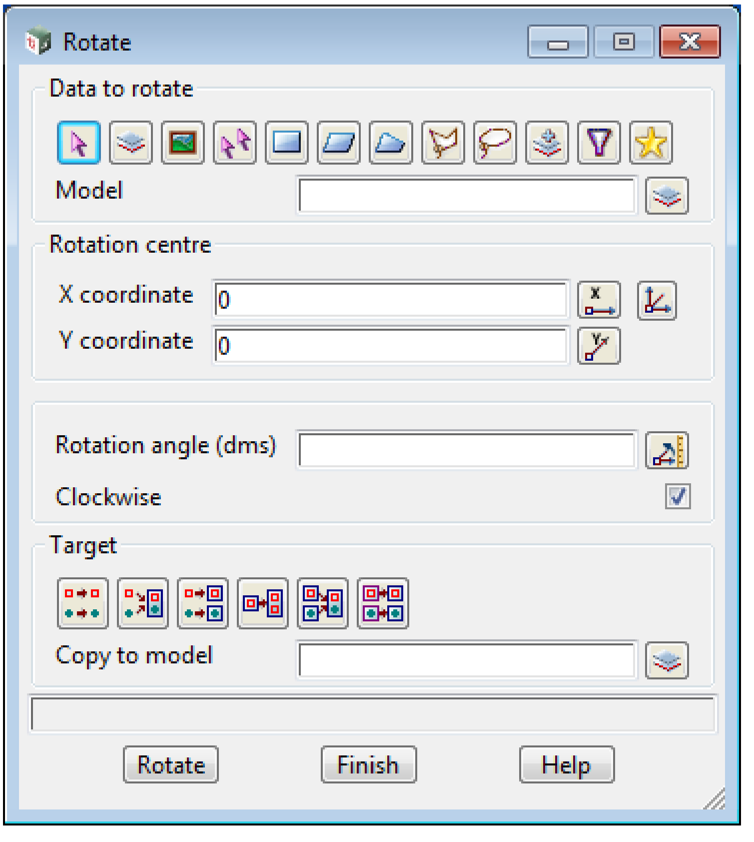
28.10.11 Rotate

Position of option on menu: **Utilities =>H-Z =>Rotate**

Rotate rotates all the selected strings about a central point (the rotation centre). The strings are rotated through a user supplied angle and can be moved or copied to their new rotated position.

The difference between a copy and a move is that if a string is **copied**, a copy of the string is rotated and the original string left untouched, whereas for a **move**, the actual string is moved from its original position to the new rotated position.

Selecting **Rotate** displays the **Rotate** panel.



The centre of the rotation is defined by selecting the **Centre** button and picking a point as the new centre. The centre can be changed at any time by using the **Centre** button to select another centre.

The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source

Data source	input		

source of data to be processed.

Rotation centre			xyz ops

select the centre of the rotation. The centre can be changed at any time.

Rotation angle (dms)	angle box		

the angle in degrees to rotate the string through. The angle is measured about the rotation centre point in either a clockwise or anticlockwise direction.

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info input

extra information required for the target.

Rotate button

rotate the selected data.

28.10.12 Rotate Relative

Position of option on menu: **Utilities =>H-Z =>Rotate relative**

Rotate Relative allows multiple strings to be rotated in the XY plane by a user given angle and about various base points (rotation centres).

The simpler Rotate option (**Utilities =>H-Z =>Rotate**) only rotates the selected strings about a **single** point.

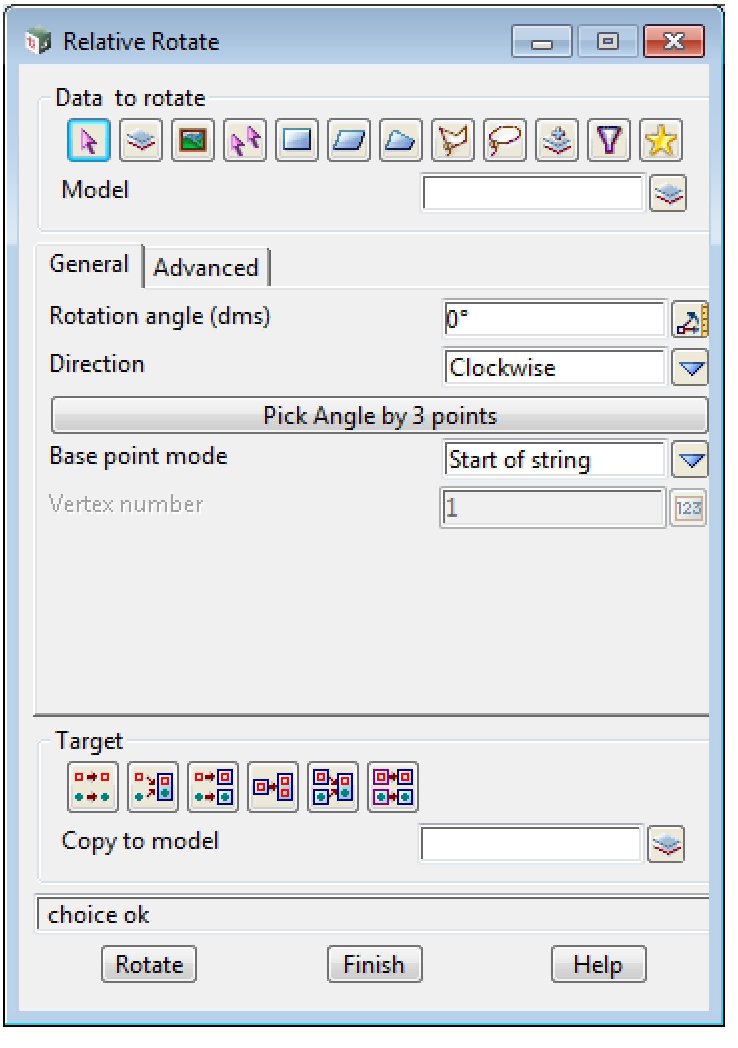
Rotate Relative is more flexible and has many methods for defining the base points (rotation centres), even allowing individual strings to have their own base point. In this case, the base point for rotation is calculated for each string in the Data Source and the rotation applied to each string individually. For example, Relative Rotate can rotate strings where the base point for each string is the start vertex of each string.

Allowing various base point modes allows greater flexibility in rotating many strings at once.

Rotate Relative performs a 2d rotation of strings. That is, the rotation is only in the XY plane.

The strings can be **moved** or **copied** to their new rotated position. The difference between a copy and a move is that if a string is copied, a copy of the string is rotated and the original string left untouched, whereas for a move, the actual string is moved from its original position to the new rotated position.

Selecting **Rotate Relative** displays the **Relative Rotate** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type	source box	Model	

data selection type - for a full description go to 4.19.3 Data Source

Data source	input		

source of data is to be processed.

Rotation angle (dms)	angle box	0°	

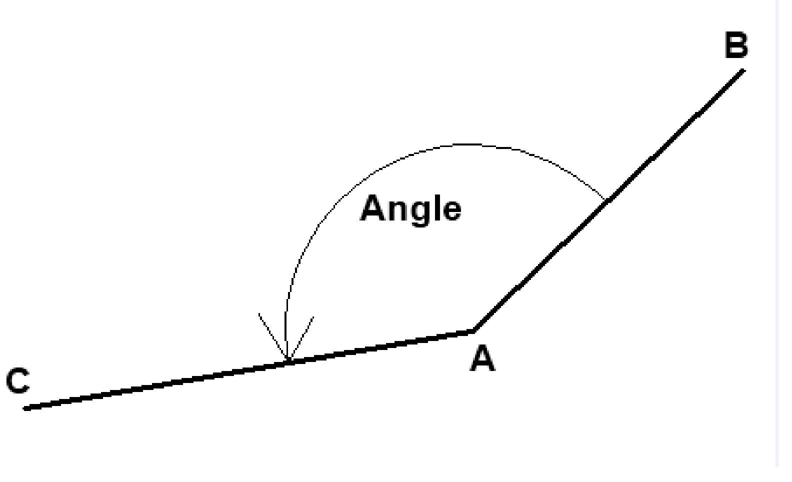
the angle in degrees to rotate the string through. The angle is measured about the rotation centre point in either a clockwise or anticlockwise direction.

Direction	choice box	Clockwise	Clockwise, Counter-clockwise

the direction in which to apply the angle, either clockwise or counter-clockwise.

Pick Angle by 3 points	button		

allows the user to measure the angle for rotation by picking 3 points on the screen.



Upon clicking the button, the user will be asked to pick the:

1. *Base point about which the angle is measured (A);*
2. *Reference point (B); and*
3. *New point (C).*

The Rotation angle and Direction will be calculated based on the 3 picked points.

Base point mode	choice box	Start of string	Start of string, End of string, Middle of string, Middle Vertex, Vertex Number, Chainage, Chainage & Offset, Coordinates, Angle by 3 pts Vertex, Justification point (Text only!)

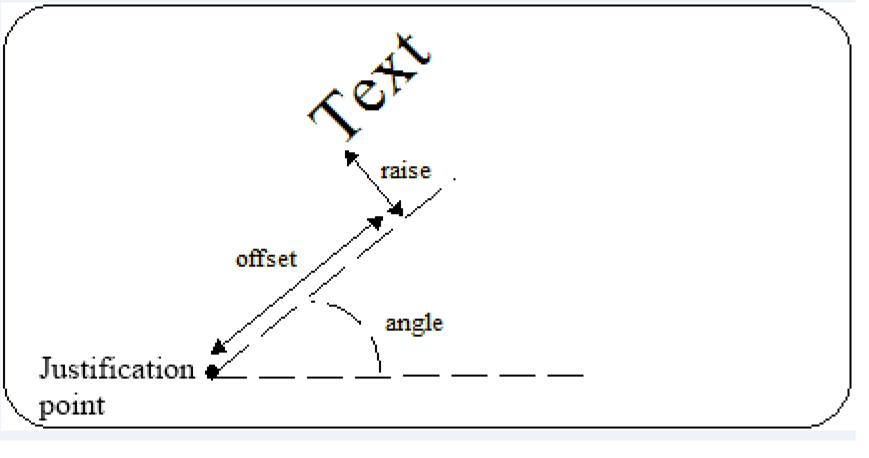
allows the user to define how the base point for the rotation is determined for the data. Valid options include:

• *Start of string – the base point is the start of each string*

• *End of string – the base point is the end of each string*

• *Middle of string – the base point is the middle (based on chainage) of each string*

- *Middle Vertex – the base point is the middle vertex of each string. In the case of a string with an even number of vertices, the behaviour of this routine can be varied. Refer to the Middle Vertex Mode in the Advanced tab.*
- *Vertex Number – the base point is a specific vertex of each string, specified by the vertex number. The start vertex is 1, the next vertex 2 and so on for a string with n vertices. The vertex number may also be specified as a negative number to count vertices from the end of the string back towards the start. In this case, for a string with n vertices, a vertex number of -1 is the last vertex (nth vertex), -2 the 2nd last vertex (n-1), and so on until the first vertex number is –n. If the specified vertex number exceeds the string extents, then an error will be produced and the string skipped.*
- *Chainage – the base point is calculated at a specific chainage along each string. Where the chainage is beyond either end of a string, the behaviour of this routine can be varied. Refer to the Base point chainage extend mode in the Advanced tab.*
- *Chainage & Offset – the base point is calculated at a specific chainage along and offset from each string. The standard convention of a negative offset to the left of the string and positive offset to the right of the string applies.*
- *Coordinates – the base point is specified by a set of coordinates. In this mode, the routine behaves in the same way as the normal Rotate routine- the same base point is used for all strings. The Z coordinate is not optional and must be entered, even if it is zero. The Z coordinate, however, does not affect the rotation.*
- *Angle by 3pts Vertex – the base point is the set of coordinates selected when the Angle by 3 pts was picked. This is the coordinates of point A in the diagram above.*
- *Justification point (Text only!) – the base point is the justification point of a Text object. Note that this only applies to Text objects- all other objects will be skipped in this mode.*



The fields displayed on the panel will change depending on the base point mode chosen.

Vertex number integer box 1

the vertex number to be used as a base point. Enabled for Vertex Number mode only.

Chainage real box 0

the chainage to be used for calculating the base point. Enabled for Chainage and Chainage & Offset modes only.

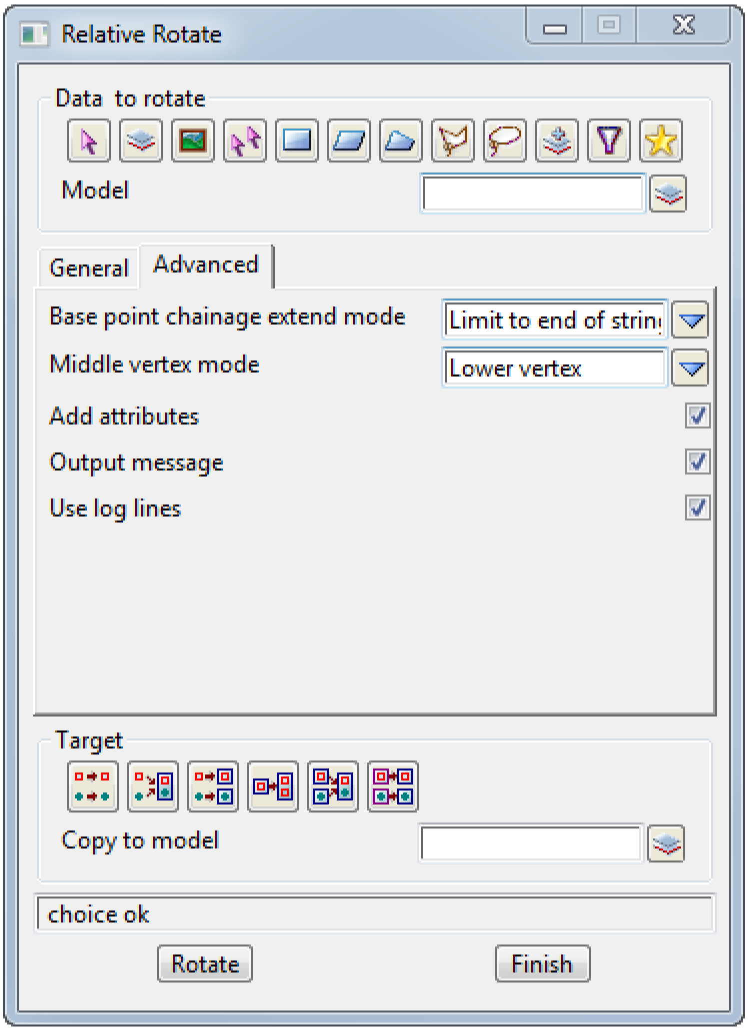
Offset real box 0

the offset to be used for calculating the base point. Enabled for Chainage & Offset mode only.

Base Point xyz box 0,0,0

the coordinates (X, Y, Z) to be used as the base point. Enabled for Coordinates mode only.

Advanced Tab

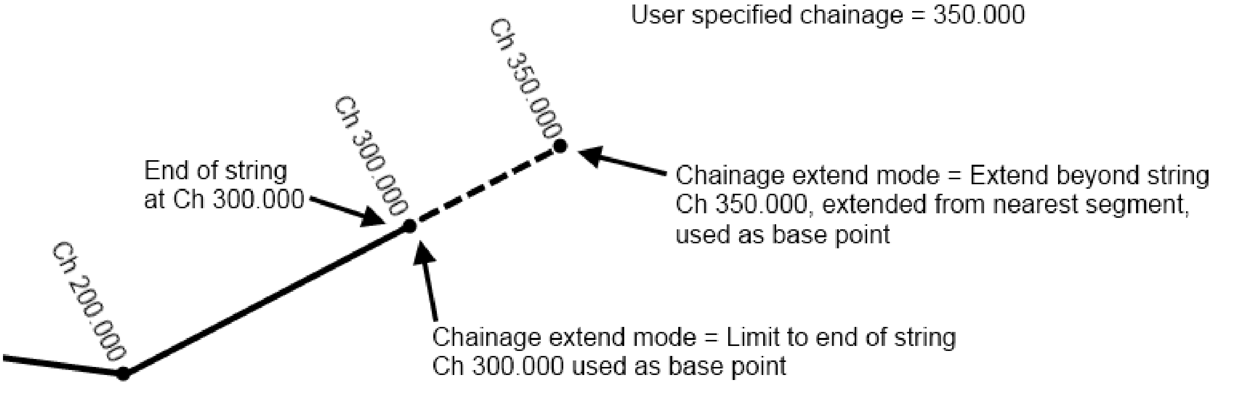


The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Base point chainage extend mode	choice box	Limit to end of string	Skip with error, Limit to end of string, Extend beyond string

controls the behaviour of the routine when the base point mode is Chainage and the specified chainage is beyond the string extents.

- *Skip with error – the string will be skipped, with an error written to the output window*
- *Limit to end of string – the base point will use the chainage of the nearest end point*
- *Extend beyond string – the base point will be calculated based on a projection beyond the string to the specified chainage and based on the bearing of the nearest end.*



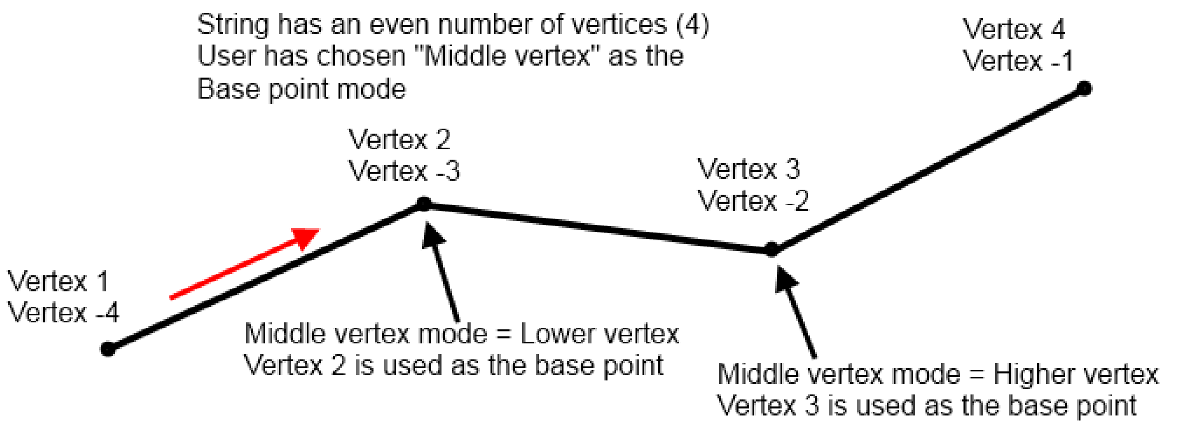
Middle vertex mode	choice box	Lower vertex	Skip with error, Lower vertex< Higher vertex

controls the behaviour of the routine when the base point mode is Middle Vertex and the string has an even number of vertices. Where the string has an even number of vertices, the middle vertex can be one of two possibilities.

Skip with error – the string will be skipped, with an error written to the output window

Lower vertex – the lower-numbered vertex will be used as the middle vertex

Higher vertex – the higher-numbered vertex will be used as the middle vertex



Add attributes	tick box	ticked

writes the various relative rotate settings to the rotated string as string attributes.

Output messages	tick box	ticked

writes various output messages when run. If not ticked, no output messages will be produced by the routine, including critical warning and error messages.

Use log lines	tick box	ticked

use intelligent log lines for the certain output messages. If not ticked and output messages are enabled, the messages will be normal messages.

Target type

data selection type - for a full description go to 4.19.3 Data Source

Target info input

extra information required for the target.

Rotate button

rotate the selected data.

28.10.13 Set Constant Z-Values

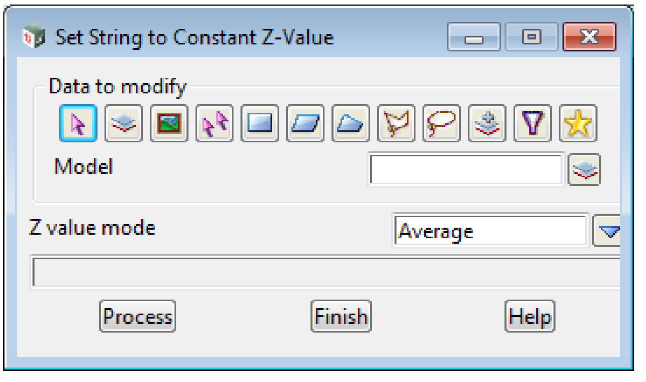
Position of option on menu: **Utilities =>H-Z =>Set constant z-values**

This option sets all the points on a string to the same elevation - the average, highest, or lowest z-value of each selected string's z-values.

For example, the option can be used to set a starting value for building pads.

The user can drape a string then average the z-values to get close to a volumes balance. Alternatively, the user can opt for the highest z-value calculated by the drape, *i.e.* the entire platform is in fill, or select the lowest z-value so the entire platform is in cut.

Selecting **Set constant z-values** brings up the **Set Constant Z-Value** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source

Data source	input		

source of data to be processed.

Z value mode	choice box	Average	Average, Highest, Lowest

if Average, the string is given the Z value that is the average of all the vertex Z values.

If Highest, the string is given the Z value that is the highest of all the vertex Z values.

If Lowest, the string is given the Z value that is the lowest of all the vertex Z values.

Process	button		

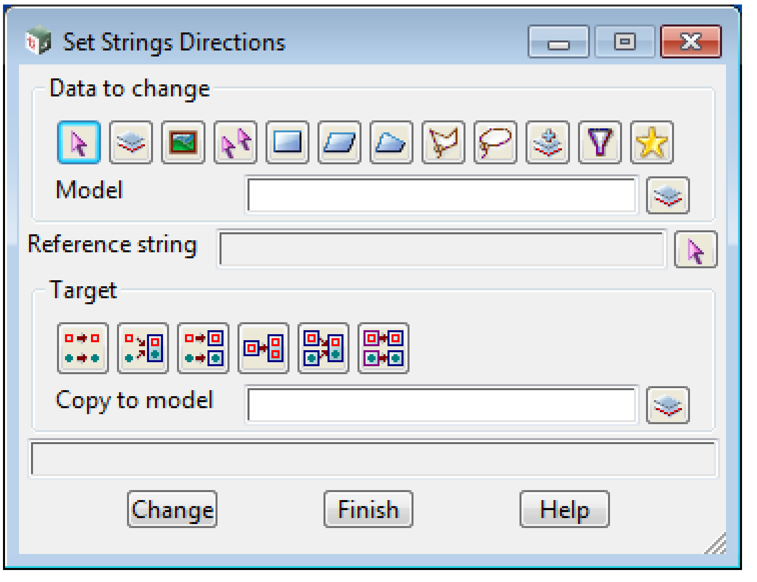
sets the constant z-value for the selected strings.

28.10.14 Set Strings Directions

Position of option on menu: **Utilities =>H-Z =>Set strings directions**

Set strings directions attempts to set the direction of select strings to be the same as the direction of a selected reference string.

On selecting **Set strings directions**, the **Set Strings Directions** panel is displayed.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source

Data source	input		

data source to be select strings to change direction.

Reference string	string select		

the string that is used as the reference direction. This is the direction that, if possible, the selected strings are given.

Target type

where to put the processed string - for a full description go to 4.19.4 Data Target

Target info	input		

extra information required for the target.

Change	button		

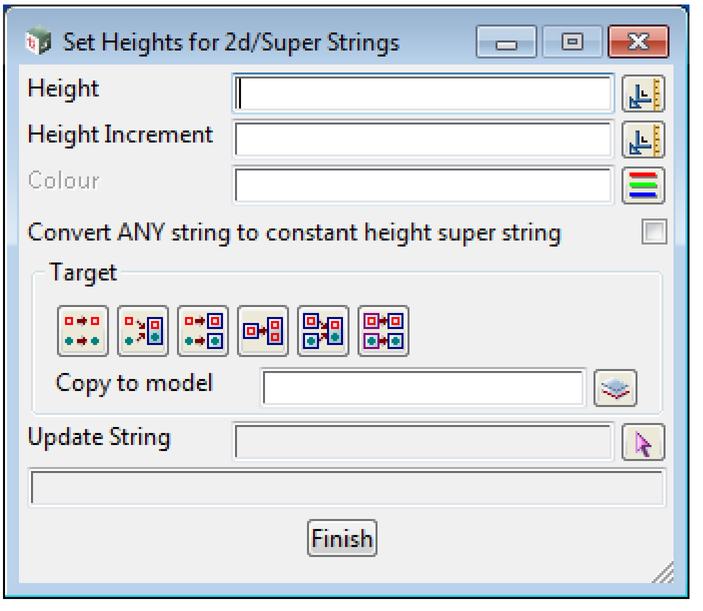
change the direction of the selected strings.

28.10.15 Set Heights for 2d/Super Strings

Position of option on menu: **Utilities =>H-Z =>Set heights for 2d/super strings**

Set heights for 2d/super strings is used to quickly change the height of selected 2d strings or 2d super strings (super strings with a constant z value). Is is especially useful when you have a lot of contours without z-values and you wish to give them z-values.

An initial height and increment is given and as each string is selected, it is given the current height and the height then incremented for use when picking the next string.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Height	input		

Height *to used as the height for the next selected string. This is incremented by the* **Height increment** *after each string is accepted.*

Height increment	input		

As each string is accepted and given the **Height**, **Height** *is incremented by this value.*

Colour	input		available colours

if not blank, the selected strings are given this colour when their z-value is changed.

If blank, the selected strings keep their original colour.

Target type

where to put the processed string - for a full description go to 4.19.4 Data Target

Target info	input		

extra information required for the target.

Update string	string select		

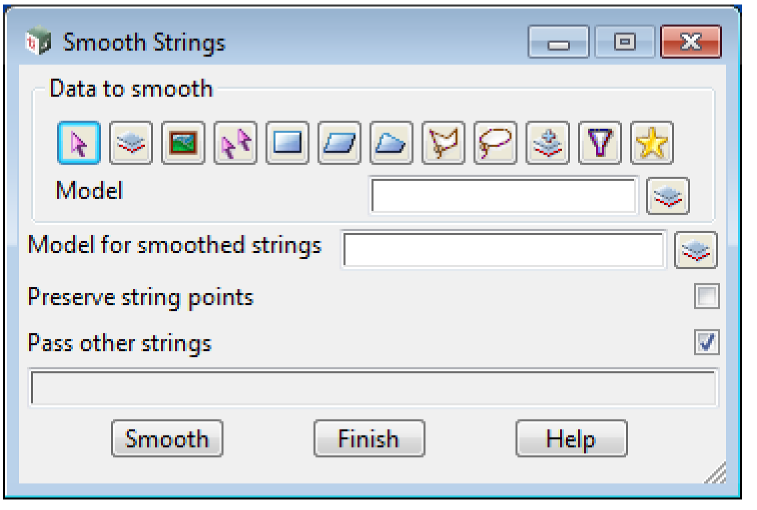
After clicking on the **Update string** *arrow icon, 2d strings are selected and when accepted, have their z-value changed to the value in* **Height**. *The height value is then automatically incremented by* **Height increment**. *The next string is then selected and accepted (without needing to click on* **Pick** *again). This continues until* ***Cancel*** *is selected from the* **Pick Ops** *menu.*

28.10.16 Smooth Strings

Position of option on menu: **Utilities =>H-Z =>Smooth strings**

The **Smooth** options are used to add extra points into 2d or 3d strings to make a smoother string.

Selecting **Smooth strings** displays the **Smooth Strings** panel.



The fields and buttons used in this panel have the following function

Field Description	Type	Defaults	Pop-Up
Data source type		Model	
Data source	input		
Model for smoothed strings	input		available models
Preserve string points	tick box		
Pass other strings	tick box	tick	
Smooth	button		

Data source type — Model

data selection type - for a full description go to 4.19.3 Data Source.

Data source — input

data source to be smoothed.

Model for smoothed strings — input — available models

model for the smoothed strings to go to

Preserve string points — tick box

*if **ticked**, the smoothed strings will still contain all the original points.*
*if **not ticked**, the smoothed strings may deviate from the original string points*

Pass other strings — tick box — tick

*if **ticked**, any strings that can't be smoothed are copied and added to the* **smoothed strings** *model.*

Smooth — button

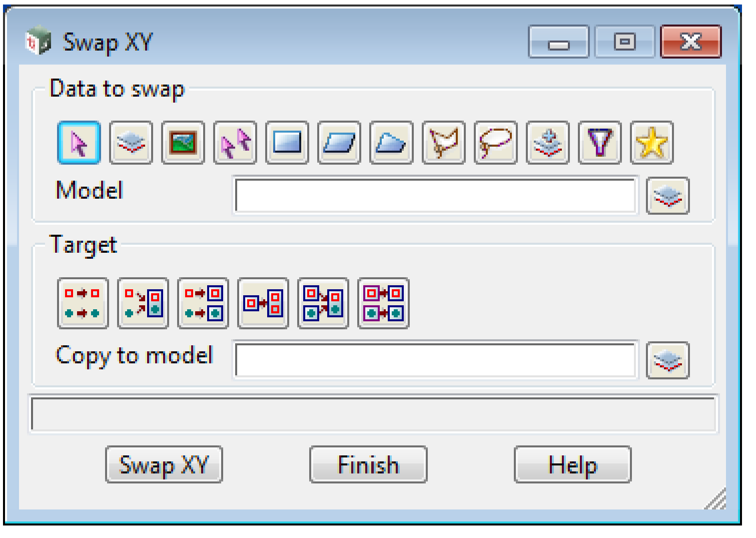
smooth the selected strings.

28.10.17 Swap XY

Position of option on menu: **Utilities =>H-Z =>Swap XY**

The **Swap xy** option is used to swap the (x,y) coordinates of selected strings.

Selecting **Swap XY** displays the **Swap XY String/Model/View** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source input

source of data to be have the XY coordinates swapped.

Target type

Data target - where to put the processed string. For a full description go to 4.19.4 Data Target

Target info input

extra information required for the target.

Swap XY button

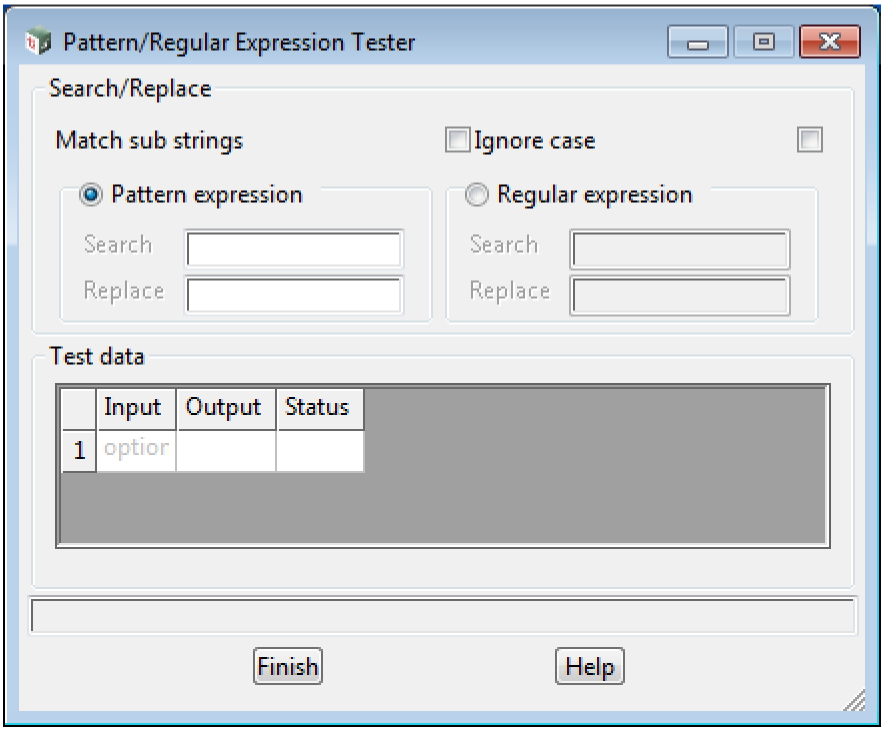
swap the selected strings and add them to the target.

28.10.18 Test Wildcards

Position of option on menu: **Utilities =>H-Z =>Test wildcards**

Test wildcards is used to test the effect of pattern expressions and regular expressions on text.

Selecting **Test wildcards** displays the **Pattern/Regular Expression Tester** panel



The **Search/Replace** criteria and the **Pattern expression** or **Regular expression** are applied to the data in the **Input** column and the results are placed in the **Output** column.

28.10.19 Text

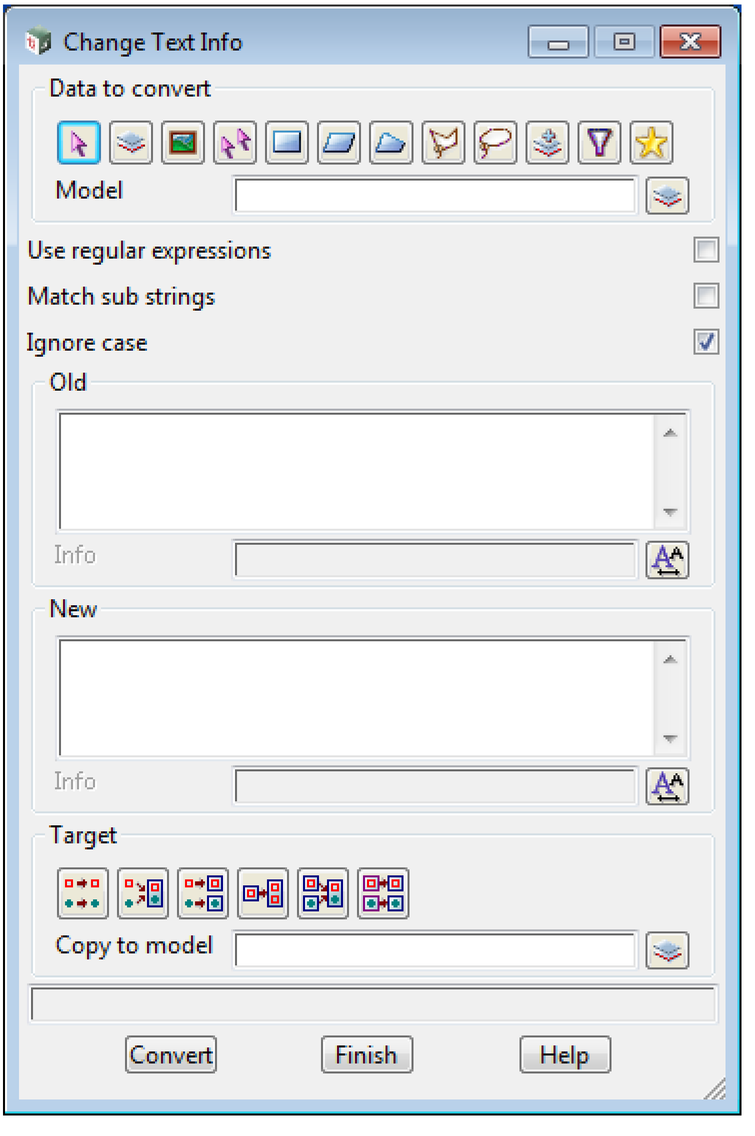
Position of option on menu: Utilities =>H-Z =>Text

The Text option is used to change the text style, units, height, offset, justification, angle and text for text, 4d strings and super strings.

For this option the selection process is extended from the standard Data Source.

After the data is selected as defined by the Data Source, the two extra parameters, **Old text** and the Textdata parameters in **Info**, are used to further specify which text is to be selected and modified.

Selecting Text displays the **Change Text Info** panel.



The fields and buttons used in the panel have the following functions.

Field Description	Type	Default
Data source type	Model	

data selection type - for a full description go to 4.19.3 Data Source.

Data source input

source of data to be processed.

Use regular expressions tick box not ticked

if ticked, then ***Regular*** *expressions given in the* **Old** *and* **New** *fields are used to modify the text.*

Ignore case tick box ticked

if ticked, the search for **Old** *text will make a match independent of upper or lower case.*

If not ticked, the search for **Old** *text will treat upper or lower case as being different.*

Any matched text is replaced by new text.

Old text input

if the **Old text** *field is non-blank, then any text in 4d or text strings will be checked for a match against this value. Wild cards and characters can be used.*

Info input

if non-blank, then only strings which match these Textdata parameters will be selected.

New text input

if non-blank, then any selected text the selected strings will be modified to this value. Wild cards and characters can be used.

Info input

if non-blank, then the select text on the selected strings will be given these Textdata parameters.

Target type

Data target - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info input

extra information required for the target.

Convert button

run the option and change the selected text.

28.10.20 Translate

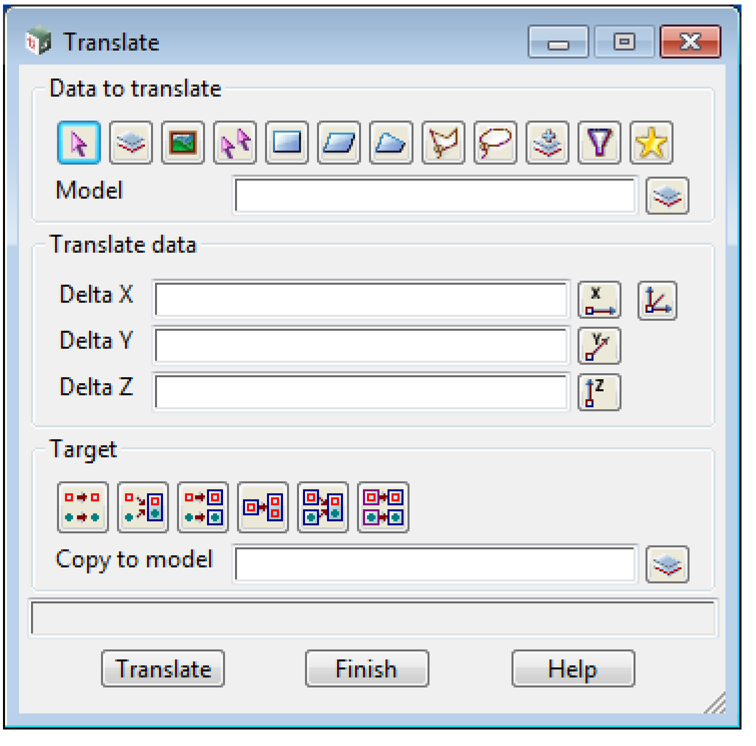
Position of option on menu: **Utilities =>H-Z =>Translate**

Translate is used to translate the selected strings through a user supplied translation vector (dx,dy,dz).

The strings can be translated in the x,y and z directions and be moved or copied to their new translated positions.

The difference between a **copy** and a **move** is that if a string is copied, a **copy** of the string is translated and the original string left untouched, whereas for a **move**, the actual string is moved from its original position to the new translated position.

Selecting **Translate** displays the **Translate** panel.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source

Data source input

source of data to be processed.

dx dy dz input

the translation vector (dx,dy,dz). This vector is added to all points in the strings.

The translation (dx dy dx) can be typing into the **dx dy dz** *field or by picking the* **dx dy dx** *button and then selecting two points with the cursor. When two points are selected, the difference vector of the two selected points is taken as the translation vector (dx,dy,dz). The (dx,dy,dz) value is piped into the* **dx dy dz** *field.*

Target type

Data target - where to put the processed string. For a full description go to 4.19.4 Data Target

Target info input

extra information required for the target.

Translate button

translate the selected strings.

How to Use the Panel

(a) Enter the **Data source** for the strings are to translated.

(b) Define the translation vector (dx dy dz) by either typing the value into the **dx dy dz** field or by selecting the **dx dy dz** button and picking two points to define the translation vector. The resulting (dx dy dz) values are piped into the **dx dy dz** field.

(c) Select the **target** mode.

(d) Selecting the **Translate** button then does the required translations (moves or copies).

28.11 Old

Position of option on menu: **Utilities =>Old**

The **Old Utilities** contains superseded options. They should no longer be used.

For *Check breaklines*, go to 28.11.1 Check Breaklines
Check breaklines (advanced) 28.11.2 Check Breaklines (Advanced)
Helmert 2D 28.11.3 2D Helmert

28.11.1 Check Breaklines

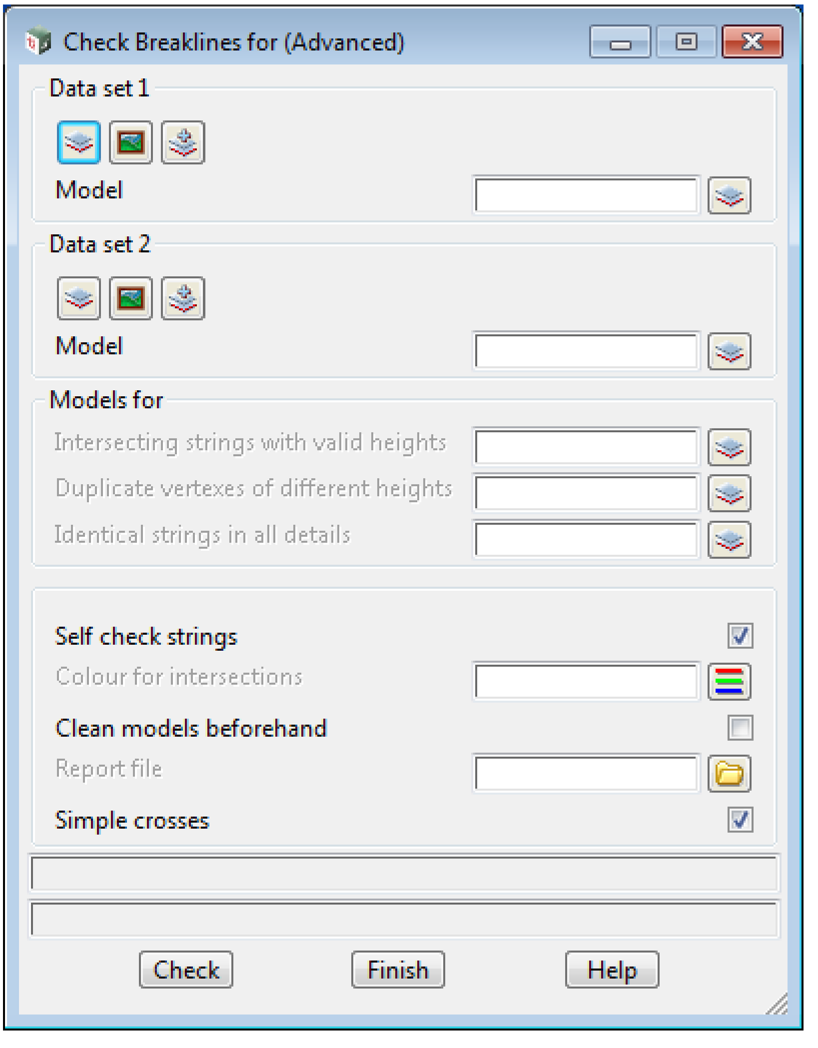
Position of option on menu: **Utilities =>Old=> Check breaklines**

For information on this option please go to the section 16.4 Check Breaklines - Old

28.11.2 Check Breaklines (Advanced)

Position of option on menu: **Utilities =>Old=> Check breaklines (advanced)**

This section of documentation is a work in progress and will be updated in subsequent releases.



The fields and buttons in the panel have the following functions.

Field Description	Type	Defaults	Pop-Up

Data set 1

*if **Data set 2** is ticked **off**, then all the selected strings in Data set 1 are checked against all the other selected strings from Data set 1.*

*If **Data set 2** is ticked **on**, then all the selected strings in Data set 1 are checked against all the selected strings in Data set 2, BUT the strings in Data set 1 are not checked against each other and the strings in Data set 2 are not checked against each other.*

Data set 1 source type Model

data selection type - for a full description go to 4.19.3 Data Source.

Data set 1 source input

data to be processed.

Data set 2

Data set 2 source type Model

data selection type - for a full description go to 4.19.3 Data Source.

Data set 2 source input

data to be processed.

Intersecting strings with valid heights model box available models

if non-blank, check for any crossing line segments and place either a copy of the crossing segments into the model given in this field or a diamond depending on the state of Simple crosses.

Duplicate vertices of different heights model box available models

if non-blank, ***copies*** *of any strings that are completely duplicated are placed in the model given in this field and circles are placed in this model at any duplicate points that aren't from an entire string.*

Identical strings in all details model box available models

if non-blank, the duplicates of any strings are ***moved*** *to model given in this field. That is, if any strings are identical in all ways, then the second and subsequent identical strings are moved to the this model. This is especially for the case when a second copy of some data has been supplied.*

Self check strings tick box tick

if ticked, a string is not check against itself for crossing breaklines etc.
If not ticked, a string is not checked against itself for crossing breaklines. This speeds up processing.

Colour for intersections colour box available colours

colour for the copies of the crossing segments.

Clean models beforehand tick box tick

if ***ticked****, the models for intersecting string, duplicate points, identicals are cleaned before the option is run.*

Report file file box

if non-blank, a report file of this name is created giving details of all the crossing breaklines and duplicate plan points with different z-values.

Simple crosses tick box tick

if ***ticked****, create diamonds at the position where strings cross, otherwise create a string in the shape of a cross with parts of the crossing strings.*

Check button

after selecting the **check** *button, all the strings in the model/view are tested for any crossing line segments or duplicate points. If requested, a report is generated.*

<esc> can be used to abort the checking option.

28.11.3 2D Helmert

Position of option on menu: **Utilities =>Old=> 2D Helmert**

NOTE - this option has been superseded and should not be used. It is for upward computability only.

A ***2D Helmert*** transformation is a two dimensional linear transformation consisting of a scaling, rotation and a 2D- translation (shift) of data. Hence there are flour parameters to be specified.

Hence the 2D Helmert parameters are

(a) the one scale factor for both the x and y axes

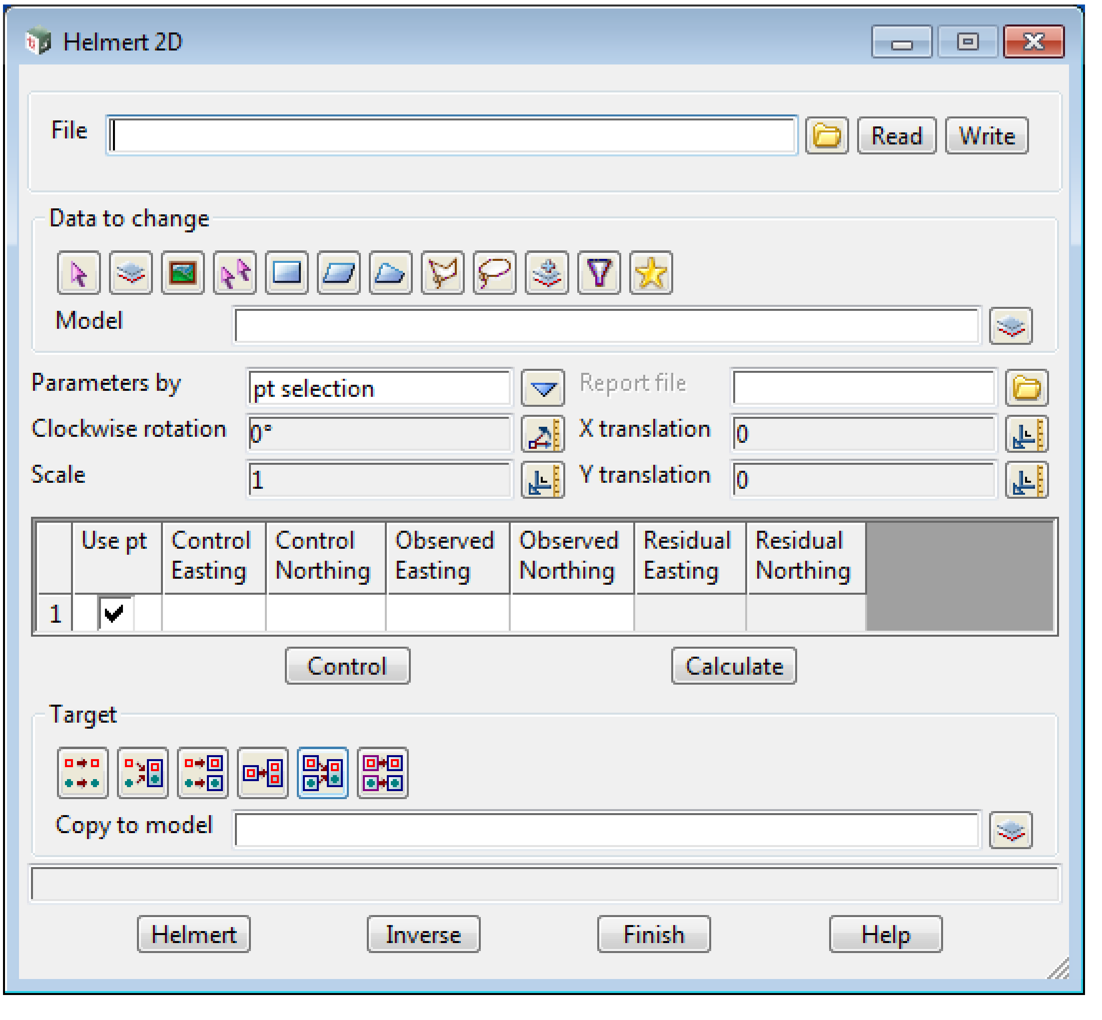
(b) the same rotation of the x-axis and y-axis of the existing coordinates to align it with the control x and coordinate axis

(c) the x and y translation of the observed coordinates after the scalings and rotations have been applied

For information on the 2D Helmert transformation, please go to the section 28.9.1 Coordinate Transformations - Helmert and Affine

This case of a fixed scale is not allowed in this option but is in the 28.10.4.1 2D Helmert (Advanced) option.

Selecting **Helmert** brings up the **Helmert 2D** panel.



If the **Parameters by** choice box is set to ***Pt Selection*** then control-observed point pairs are selected by the user and the pairs are written to the grids on the panels. See the section 28.9.1.1 Selecting Control and Observed Points for the Helmert and Affine Transformations for information on picking control and observed points.

If the **Parameters by** choice box is set to **Direct entry**, the transformation parameters are manually entered into the panel and the grid section is not used.

The fields and buttons used in this panel have the following functions

Field Description	Type	Defaults	Pop-Up
File	file box		*.hel files

a file name can be specified for reading or writing a file.

Read	button		

if a valid file exists, the file contents can be loaded into the panel.

Write	button		

if a valid name is specified, the user can write the input data to a file.

Data source type		Model	

data selection type - for a full description go to 4.19.3 Data Source

Data source	input		

source of data to be processed.

Helmert parameters by	choice	Pt selection	Pt selection Direct entry

the parameter entry method.

Report file			*.rpt files

if not blank, write a report on the Helmert transformation.

Rotation	input/output		

the rotation of the axes of the existing points with respect to the transformed axes.

Scale	input/output		

the scale factor for the existing coordinates with respect to the transformed coordinates

X Translation	input/output		

the x translation of the existing coordinates with respect to the transformed x coordinates

Y Translation	input/output		

the y translation of the existing coordinates with respect to the transformed y coordinates

Control	button		

restarts the selection process for choosing more control points.

Calculate	button		

calculate the helmert parameters from the selected points shown in the grid.

Target type

Data target type - where to put the processed strings. For a full description go to 4.19.4 Data Target

Target info	input		

extra information required for the target.

Helmert	button		

apply the helmert transformation to the data specified in the source box, and put it into the appropriate *target area.*

Inverse button

applies the inverse transformation to the one defined in the panel.

Finish button

end the option, remove this panel and the **Helmert control points table** *from the screen.*

29 User

12d Model allows options to be placed on User menus.

Go to

29.1 User Defined Menus

To help customise ***12d Model***, there is a pull down menu **User** on the Main menu and **User** menus on each of the pull down menus on the Main menu.

The **User** menus can run 12d Model macros, external programs, chains, and bring up ***12d Model*** screen layout files, panels and menus.

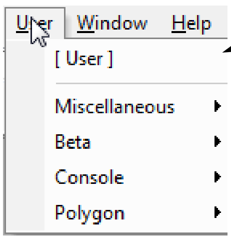
For the full definition and format for user defined menus, see 41.2 User Defined Menus in the Appendix 41 Functions Keys, Menus, Toolbars.

29.2 User on the Main Menu

All the options under **User** and **User menus** are either customers own macros or unsupported macros from ***12d Solutions*** which may not be documented. For more information on defining *User* menus, go to 29.1 User Defined Menus

The **User** walk-right menu on the Main menu is

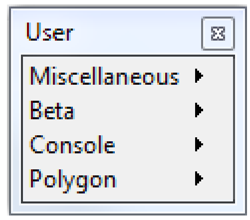
on Main menu



create floating
User menu

create floating User menu
miscellaneous macros
beta macros
console macros
polygon macros

on 12d Model menu and floating User menu

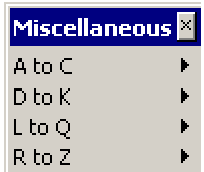


For the option Miscellaneous, go to 29.3 Miscellaneous
Beta 29.4 Beta
Console 29.5 Console
Polygon 29.6 Polygon

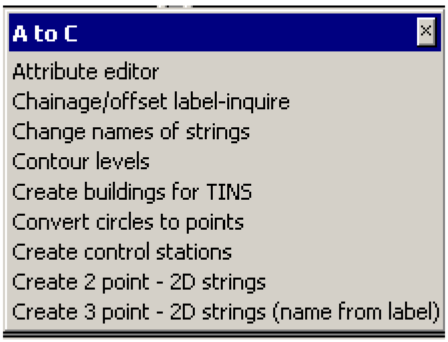
29.3 Miscellaneous

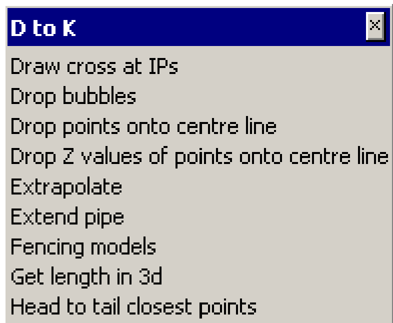
All the options under **User** and **User menus** are either customers own macros or unsupported macros from 12D Solutions which may not be documented. For more information on *User* menus, go to 29.1 User Defined Menus

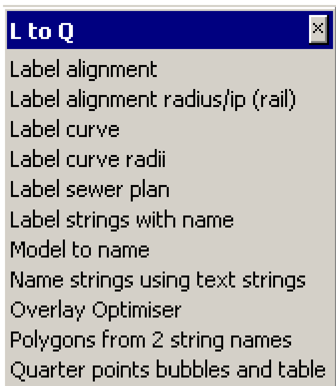
The walk rights **User =>Miscellaneous** menu is:

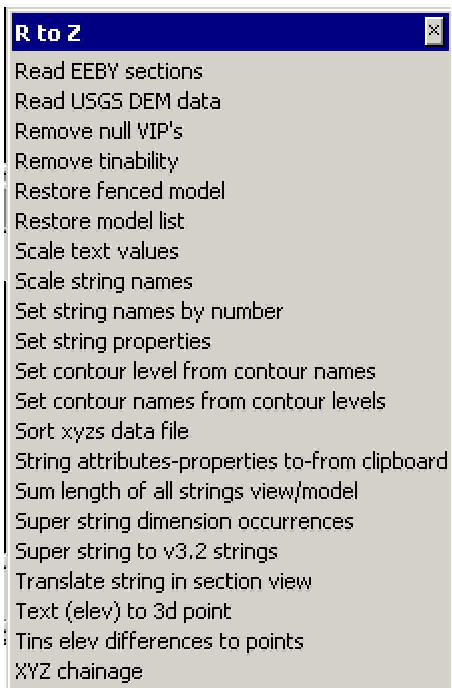


and the walk rights on the **Miscellaneous** menus are:









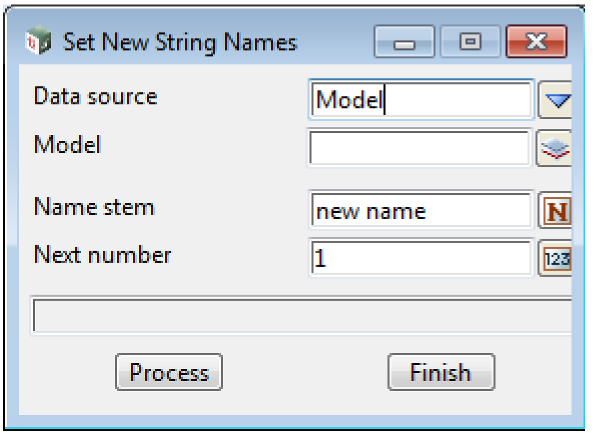
For *Attribute Editor*, go to the section	14.18.1 Attribute Editor
Chainage/offset label inquire	*this is a console macro*
Change names of strings	29.3.1 XXX New Name
Contour levels	29.3.2 Set Heights for 2d (contour) Strings
Create buildings for TINS	29.3.3 XXX Create Building for a Tin
Create points for circle centres	29.3.4 Create Points for Centre of Circles

Create control stations	29.3.5 XXX Create Control Stations
Create 2 point - 2D strings	29.3.6 XXX Create 2d String from Two Points
Create 3 point - 2D strings (name from label)	29.3.7 XXX Create 3 Point 2d Strings
Draw cross at IPs	24.4 Create Crosses at String Points
Drop bubbles	29.3.8 XXX Drop Bubbles
Drop points onto centreline	29.3.9 XXX Drop Points onto Alignment
Drop Z values of points onto centre lines	29.3.10 XXX Drop z-value onto Centreline
Extrapolate	29.3.11 Extrapolate Point
Extend pipe	29.3.12 XXX Create VicRoad Pipe
Fencing models	29.3.13 XXX Fencing Models
Get length in 3d	29.3.14 Length in 3d
Head to tail closest points	29.3.15 XXX Head to Tail Closest Points
Label alignment	*this is a console macro*
Label alignment radius/ip (rail)	29.3.17 XXX Label Alignment Radius
Label curve	*this is a console macro*
Label curve radii	*this is a console macro*
Label sewer plan	29.3.18 XXX Label Sewer Plan (old)
Label strings with name	29.3.19 XXX Name Strings by Nearby Text
Model to name	29.3.20 XXX Name Strings by Model Name
Name strings using text strings	29.3.21 XXX Name Section Strings by Picking Text
Overlay Optomiser	29.3.22 XXX Overlay Optimiser
Polygon from 2 string names	20.14.5 Polygons from Sections
Quarter points bubbles and table	17.18.3 Setout Lip Line
Read EEBY sections	29.3.23 XXX Read EEBY Sections
Read USGS DEM data	29.3.24 XXX Read USGS DEM Data
Remove null VIPs	29.3.25 Remove Null VIP Points
Remove tinability	29.3.26 XXX Remove Tinability
Restore fenced model	29.3.27 XXX Restore Fenced Model
Restore model list	12.10 Models Save/Restore
Scale text values	29.3.28 XXX Scale Text Values
Scale string names	29.3.29 XXX Scale String Names
Set string names by number	29.3.30 XXX Set String Names by Number
Set string properties	29.3.31 XXX String Operations
Set contour levels from contour names	29.3.32 Set Contour Levels from String Names
Set contour names from contour levels	29.3.33 Set Contour Names From Contour Levels
Sort xyzs data file	29.3.34 XXX Sort XYZs Data File
String attributes properties to-from clipboard	29.3.35 String Clipboard
Sum length of all strings model/view	29.6 Polygon
Super string dimension occurrences	29.3.37 XXX Super String Dimension Occurrences
Super string to V3.2 strings	29.3.38 XXX Transform V4 to V3.2
Translate string in section view	29.3.39 XXX Section Move
Text (elev) to 3d point	28.9.12.5 Text to 3d
Tin elev differences to points	16.14.15 Z Differences from Tins
XYZ chainage	29.3.40 XXX Output XYZ and Chainage

29.3.1 XXX New Name

Position of option on menu: **User =>Miscellaneous =>A to C =>Change name of strings**

Note - This is an unsupported option which may not be fully documented.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Data source		Model	string, model, view
Model/View/String of circles			

type of source of the data to process.

Model for centres	model box		available models

source of data to process.

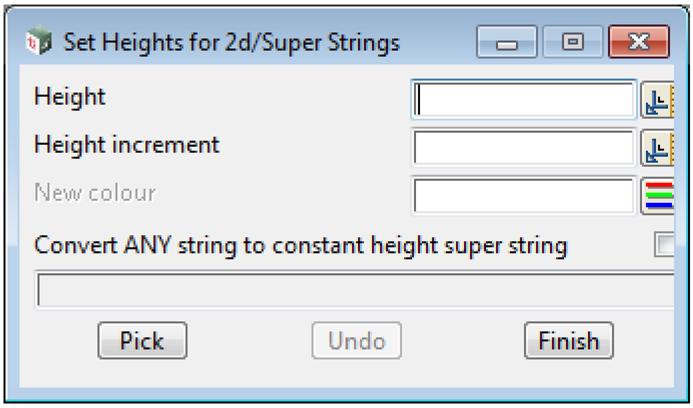
Name stem	input		
Process	button		

29.3.2 Set Heights for 2d (contour) Strings

Position of option on menu: **User =>Miscellaneous =>A to C =>Contour levels**

Note - This is an unsupported option which may not be fully documented.

This panel is used to quickly change the height of contour strings. An initial height and increment is given. As each string is accepted, it is given the height and the height then incremented.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Height	input		

Height to use for the next selected string. This is incremented by the "Height increment" after each string is accepted.

Height increment	input		

As each string is accepted, the "Height" value is incremented by this value.

New colour	input		available colours

If non-blank, the selected strings are given this colour when their z-value is changed.

Pick	input		

After clicking on Pick, 2d strings are selected and when accepted, have their z-value changed to the value in the Height field. The height value is then automatically incremented. The next string is then selected and accepted (without needing to click on Pick again). This continues until Cancel is selected from the "Pick Ops" menu.

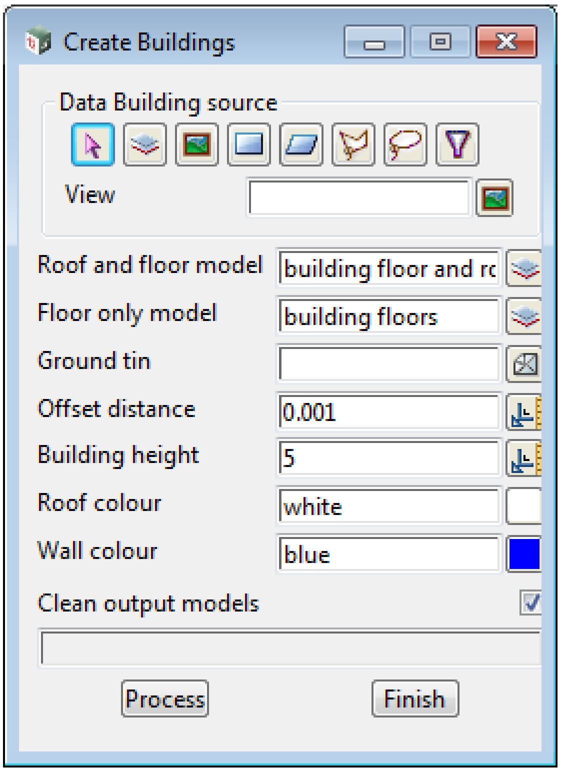
Undo	button		

Undo the last height change. This can be selected up to 200 times.

29.3.3 XXX Create Building for a Tin

Position of option on menu: **User =>Miscellaneous =>A to C =>Create buildings for tin**

Note - This is an unsupported option which may not be fully documented.

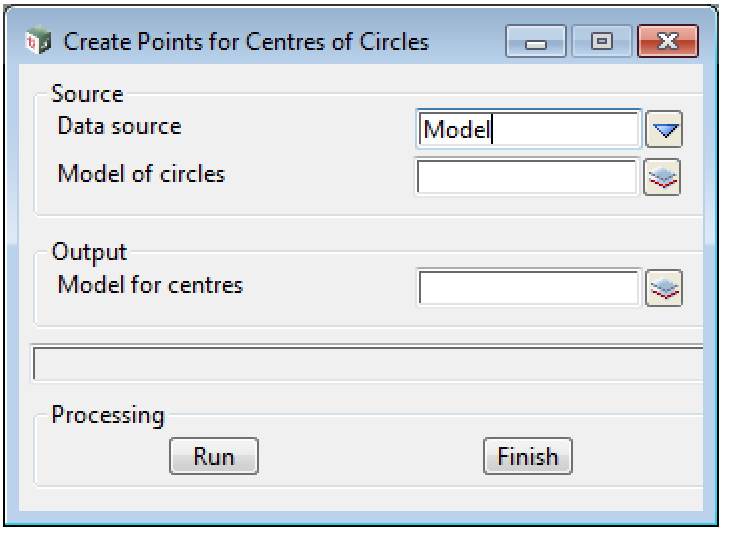


29.3.4 Create Points for Centre of Circles

Position of option on menu: **User =>Miscellaneous =>A to C =>Convert circles to points**

Note - This is an unsupported option which may not be fully documented.

This options creates points at the centres of circles.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Data source		Model	string, model, view
Model/View/string of circles			

source of the circles to create centre points for.

Model for centres	model box		available models

model to place the created centre points in.

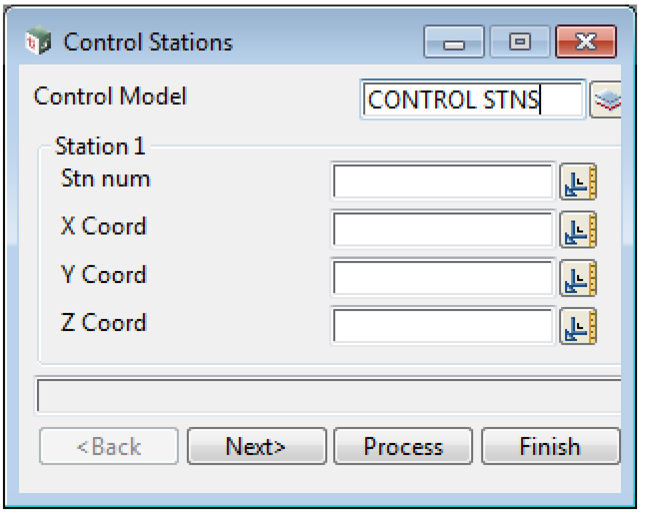
Run	button		

run the option.

29.3.5 XXX Create Control Stations

Position of option on menu: **User =>Miscellaneous =>A to C =>Create control stations**

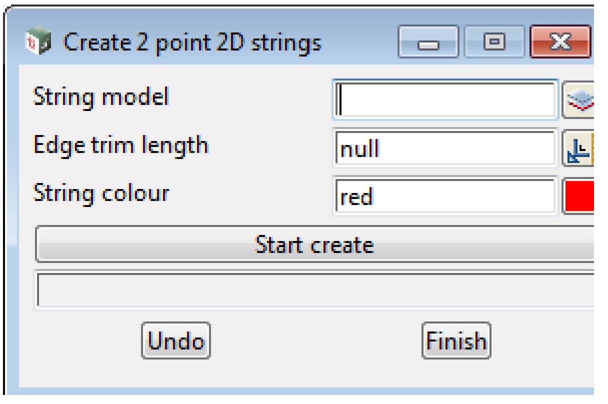
Note - This is an unsupported option which may not be fully documented.



29.3.6 XXX Create 2d String from Two Points

Position of option on menu: **User =>Miscellaneous =>A to C =>Create 2 point - 2d strings**

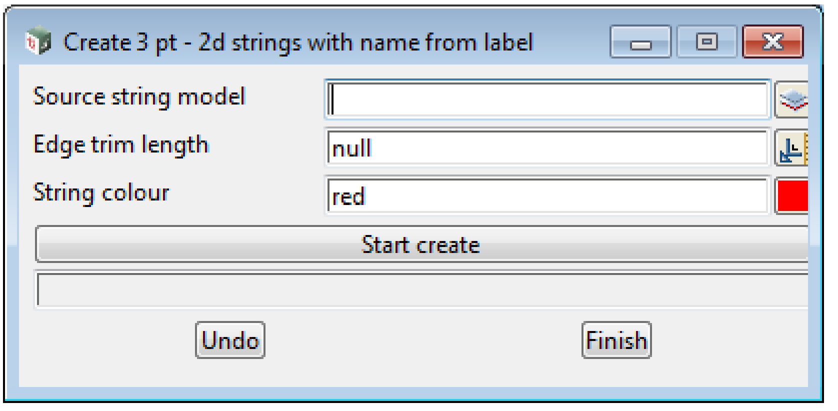
Note - This is an unsupported option which may not be fully documented.



29.3.7 XXX Create 3 Point 2d Strings

Position of option on menu: **User =>Miscellaneous =>A to C =>Create 3 point - 2d strings**

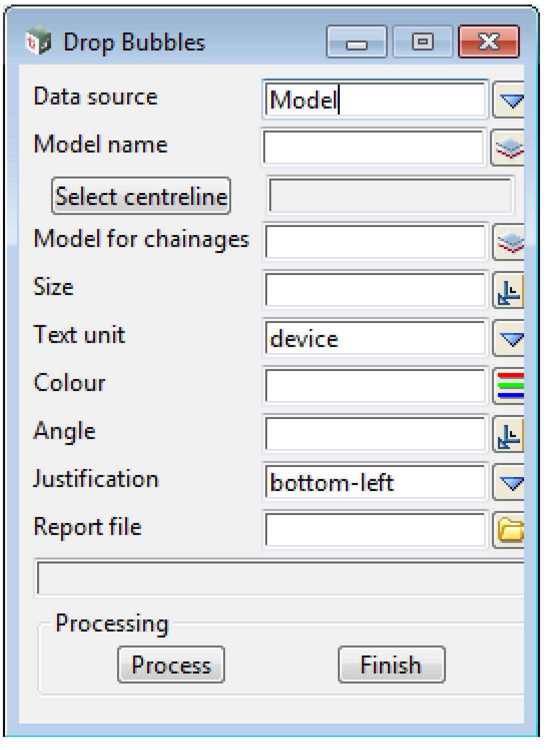
Note - This is an unsupported option which may not be fully documented.



29.3.8 XXX Drop Bubbles

Position of option on menu: **User =>Miscellaneous =>D to K =>Drop bubbles**

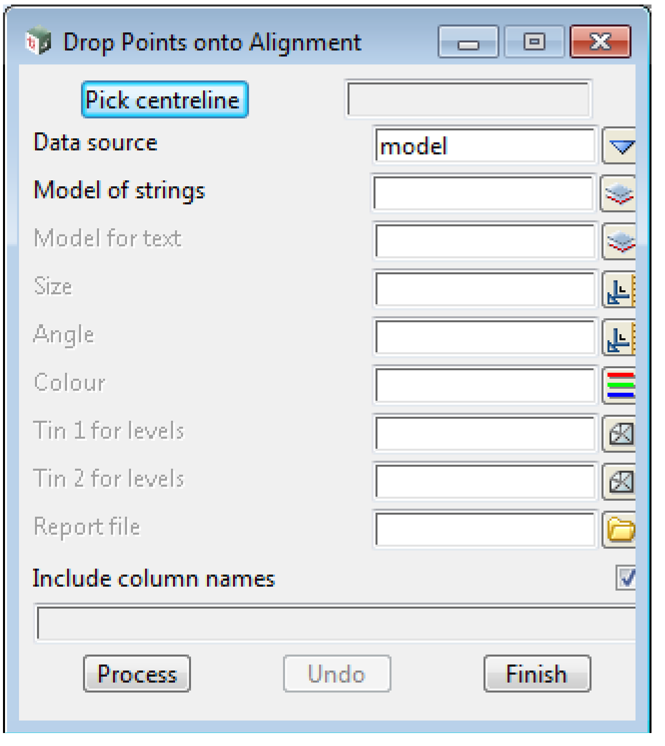
Note - This is an unsupported option which may not be fully documented.



29.3.9 XXX Drop Points onto Alignment

Position of option on menu: **User =>Miscellaneous =>D to K =>Drop points onto alignment**

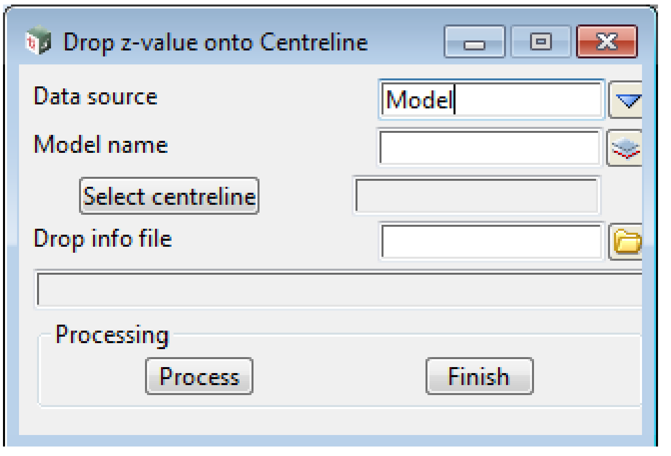
Note - This is an unsupported option which may not be fully documented.



29.3.10 XXX Drop z-value onto Centreline

Position of option on menu: **User =>Miscellaneous =>D to K =>Drop points on centreline**

Note - This is an unsupported option which may not be fully documented.

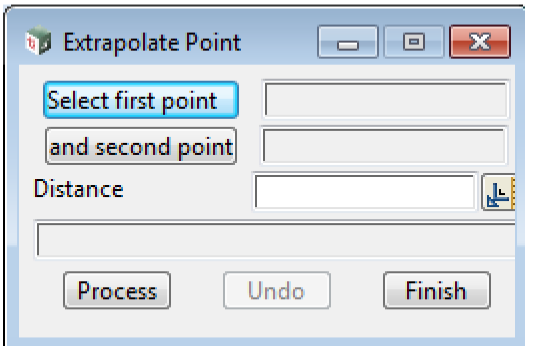


29.3.11 Extrapolate Point

Position of option on menu: **User =>Miscellaneous =>D to K =>Extrapolate**

Note - This is an unsupported option which may not be fully documented.

This option is used to select two point and then creates a new point a given distance from the second point along the line from the first point to the second point. The z-value for the point is the extrapolated z-value from the two selected points.



The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

Select first point string select

select the first point to use in the extrapolation.

and second point string select

select the second point to use in the extrapolation.

Distance output box

distance to project from the second point along the line from the first to the second point.

Process button

run the option.

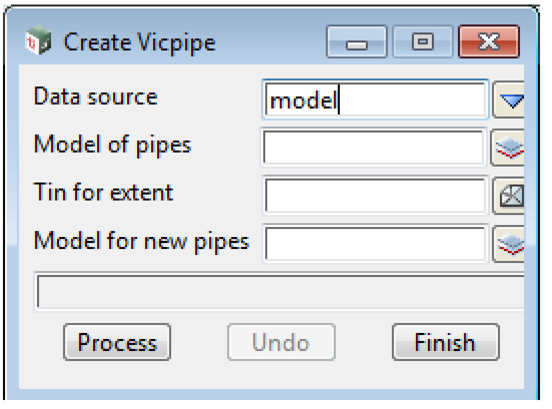
Undo button

undo the last extrapolated point created since the panel has been up.

29.3.12 XXX Create VicRoad Pipe

Position of option on menu: **User =>Miscellaneous =>D to K =>Extend pipe**

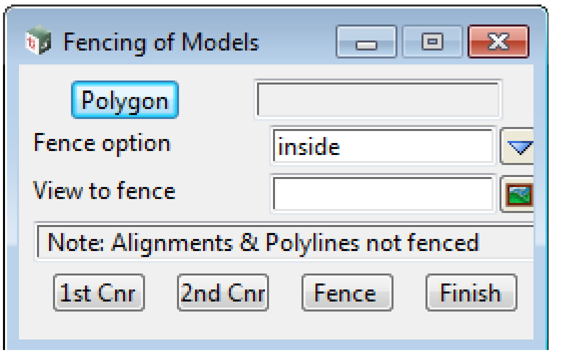
Note - This is an unsupported option which may not be fully documented.



29.3.13 XXX Fencing Models

Position of option on menu: **User =>Miscellaneous =>D to K =>Fencing models**

Note - This is an unsupported option which may not be fully documented.

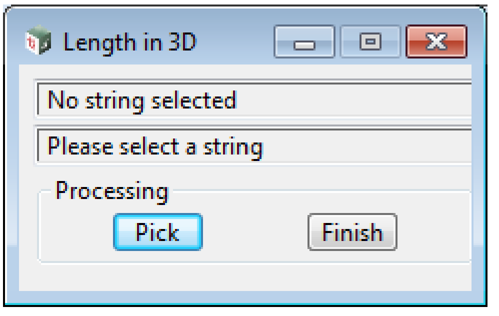


29.3.14 Length in 3d

Position of option on menu: **User =>Miscellaneous =>D to K =>Get length in 3d**

Note - This is an unsupported option.

This panel is used to calculate the 3d length of a selected string.



The fields and buttons used in this panel have the following functions:

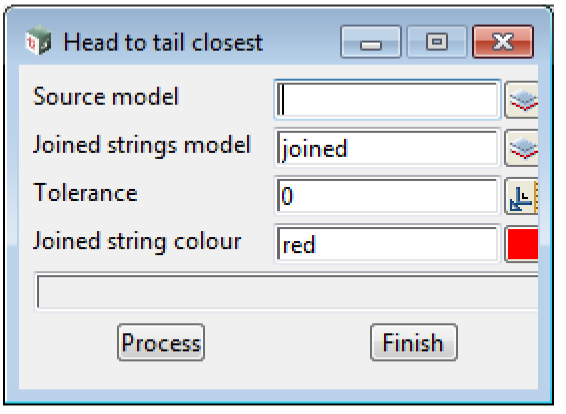
Field Description	Type	Defaults	Pop-Up
Pick	button		

select the string to calculated the 3d length for.

29.3.15 XXX Head to Tail Closest Points

Position of option on menu: **User =>Miscellaneous =>D to K =>Head to tail closest points**

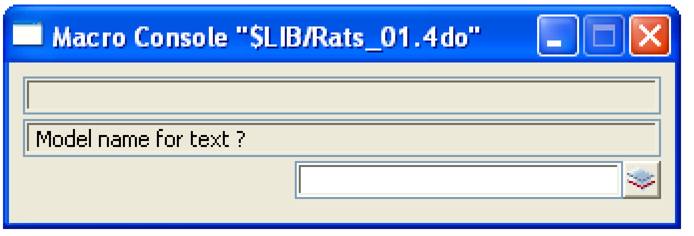
Note - This is an unsupported option which may not be fully documented.



29.3.16 XXX Label Alignment

Position of option on menu: **User =>Miscellaneous =>L to Q =>Label alignment**

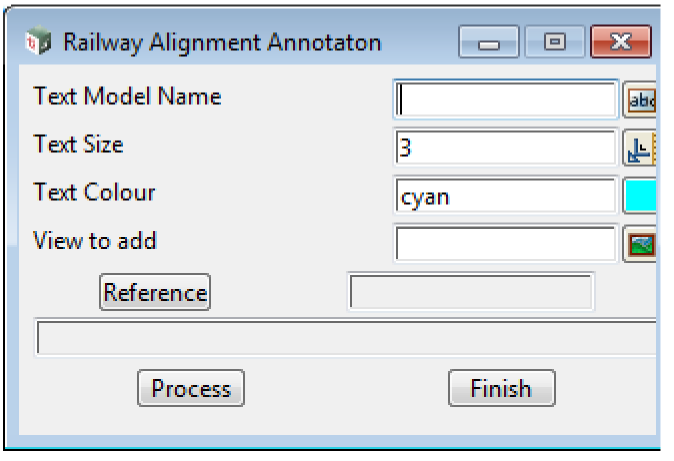
Note - This is an unsupported option which may not be fully documented.



29.3.17 XXX Label Alignment Radius

Position of option on menu: **User =>Miscellaneous =>L to Q =>Label alignment radius/ip (rail)**

Note - This is an unsupported option which may not be fully documented.

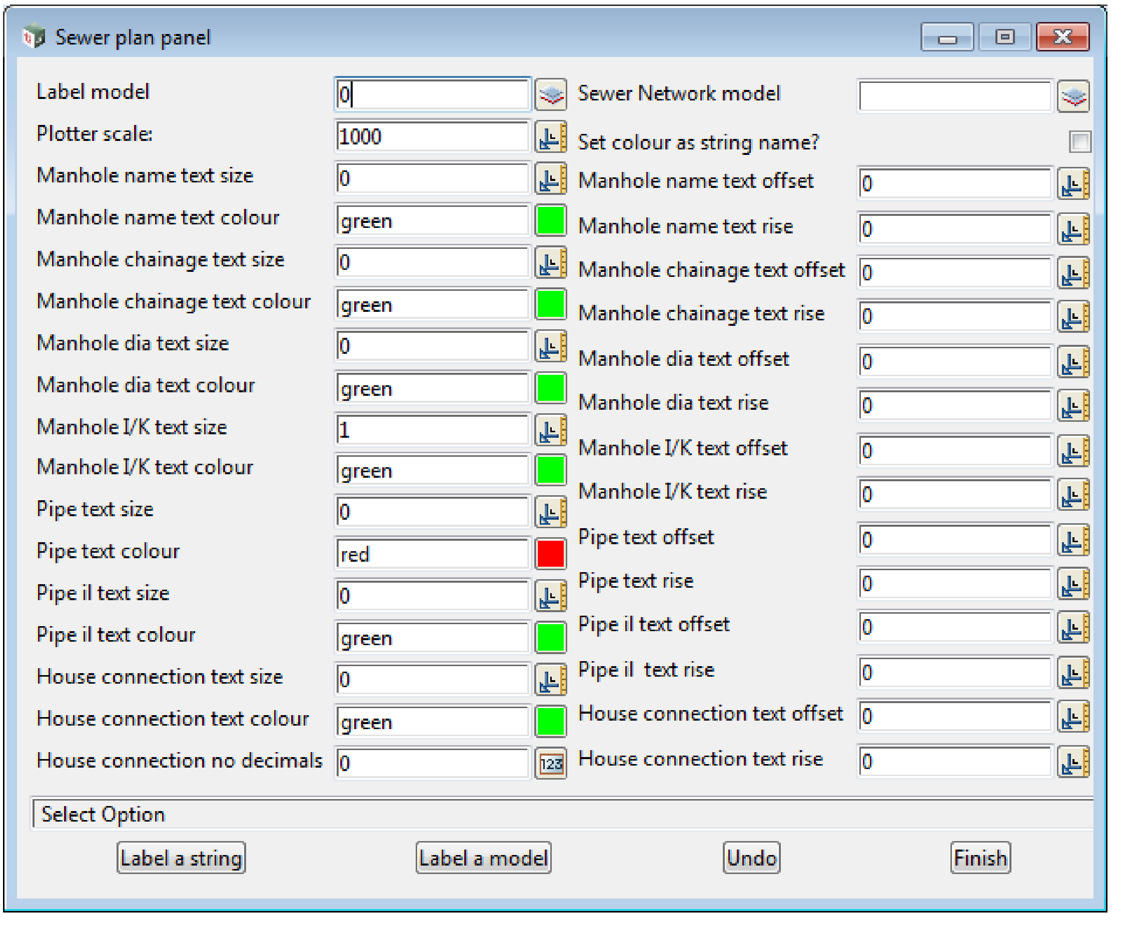


29.3.18 XXX Label Sewer Plan (old)

Position of option on menu: **User =>Miscellaneous =>L to Q =>Label sewer plan**

Note - This is an unsupported option which may not be fully documented.

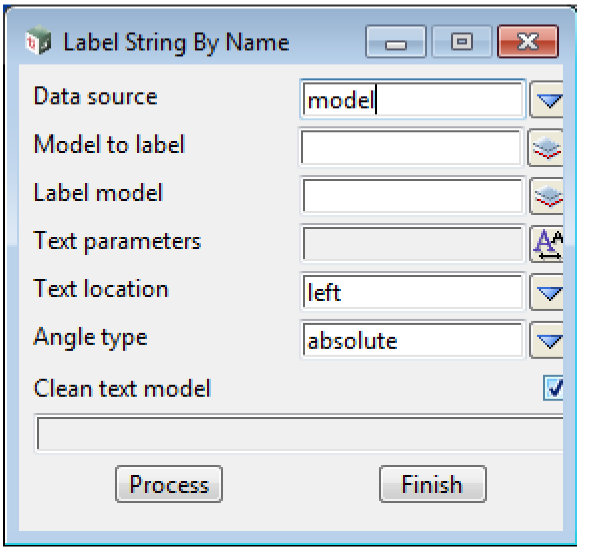
This option has been superseded.



29.3.19 XXX Name Strings by Nearby Text

Position of option on menu: **User =>Miscellaneous =>L to Q =>Label strings with name**

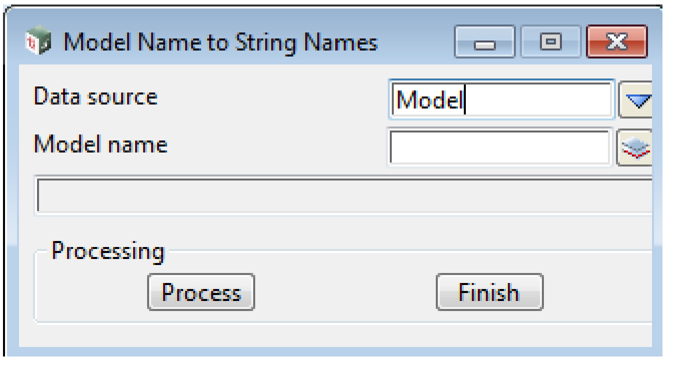
Note - This is an unsupported option which may not be fully documented.



29.3.20 XXX Name Strings by Model Name

Position of option on menu: **User =>Miscellaneous =>L to Q =>Model to name**

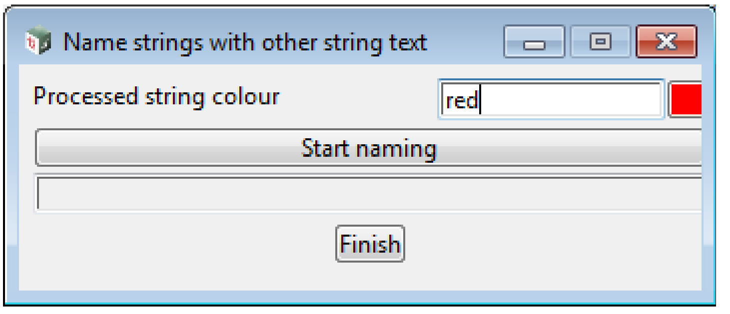
Note - This is an unsupported option which may not be fully documented.



29.3.21 XXX Name Section Strings by Picking Text

Position of option on menu: **User =>Miscellaneous =>L to Q =>Name strings using text strings**

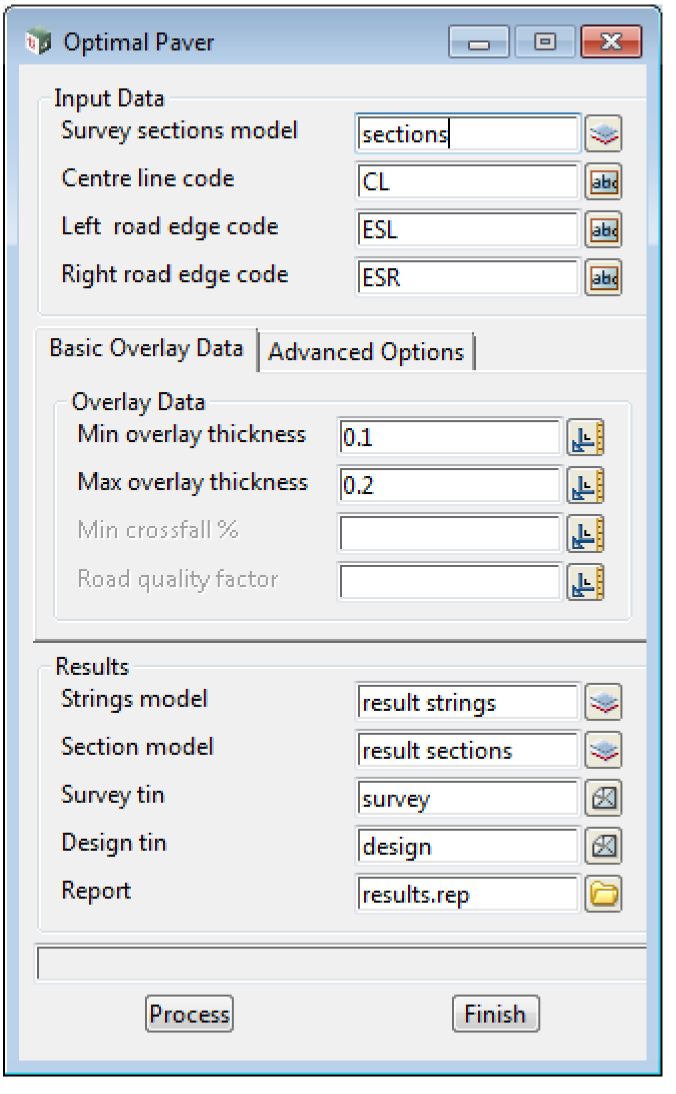
Note - This is an unsupported option which may not be fully documented.



29.3.22 XXX Overlay Optimiser

Position of option on menu: **User =>Miscellaneous =>L to Q =>Overlay optimiser**

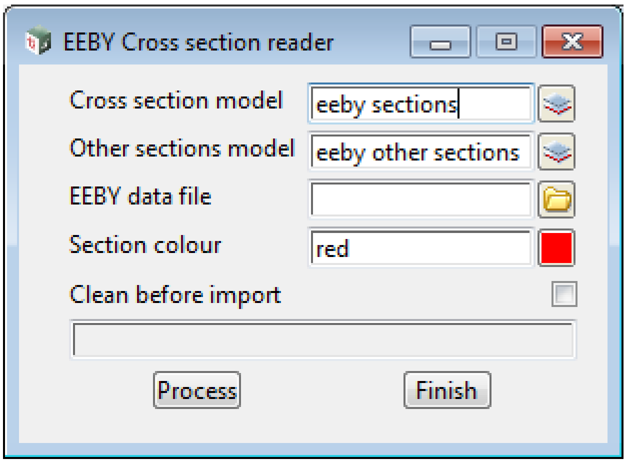
Note - This is an unsupported option which may not be fully documented.



29.3.23 XXX Read EEBY Sections

Position of option on menu: **User =>Miscellaneous =>R to Z =>Read EEBY sections**

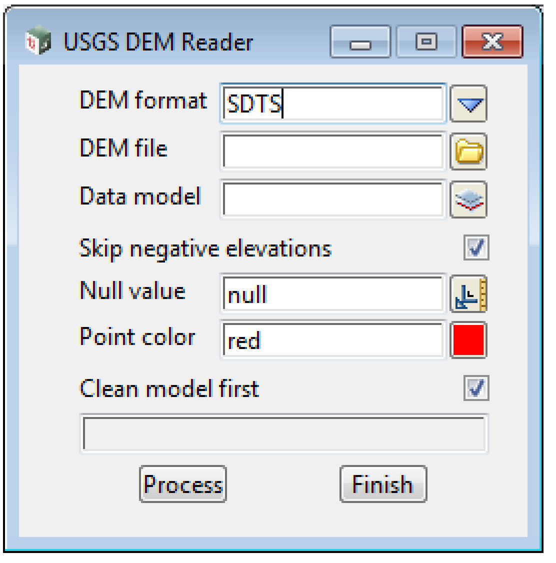
Note - This is an unsupported option which may not be fully documented.



29.3.24 XXX Read USGS DEM Data

Position of option on menu: **User =>Miscellaneous =>R to Z =>Read USGS DEM data**

Note - This is an unsupported option which may not be fully documented.

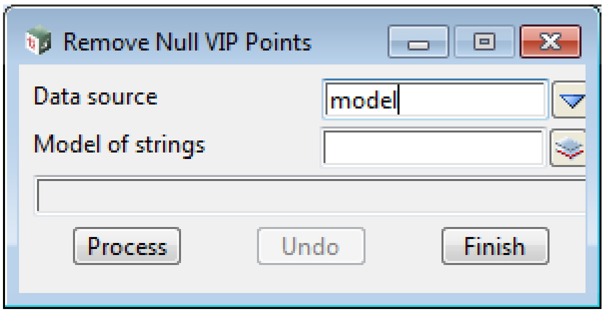


29.3.25 Remove Null VIP Points

Position of option on menu: **User =>Miscellaneous =>R to Z =>Remove null VIP’s**

Note - This is an unsupported option which may not be fully documented.

This option is used to remove any null vertical intersection points from alignment strings.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Data source	data source	Model	String, Model, View

Type of data to remove null vips for.

Model/View/String	data source

Data source for all the alignment strings to remove all null vips from.

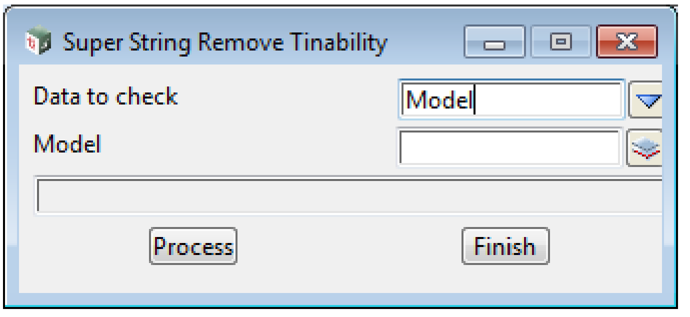
Process	button

Run the option.

29.3.26 XXX Remove Tinability

Position of option on menu: **User =>Miscellaneous =>R to Z =>Remove tinability**

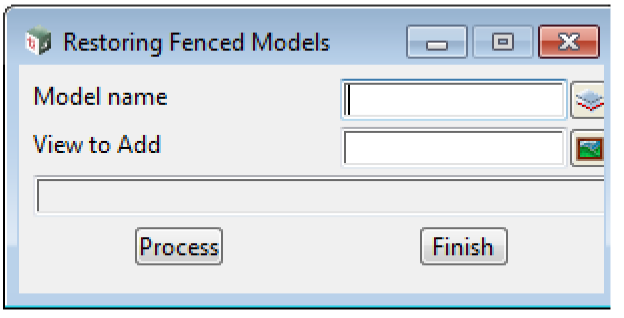
Note - This is an unsupported option which may not be fully documented.



29.3.27 XXX Restore Fenced Model

Position of option on menu: **User =>Miscellaneous =>R to Z =>Restore fenced model**

Note - This is an unsupported option which may not be fully documented.



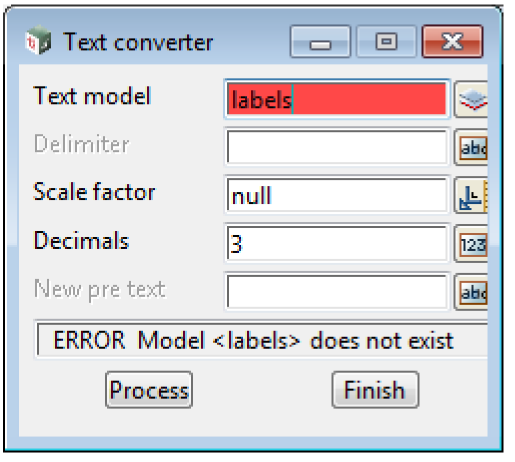
The fields and buttons used in this panel have the following functions:

Field Description Type Defaults Pop-Up

29.3.28 XXX Scale Text Values

Position of option on menu: **User =>Miscellaneous =>R to Z=>Scale text values**

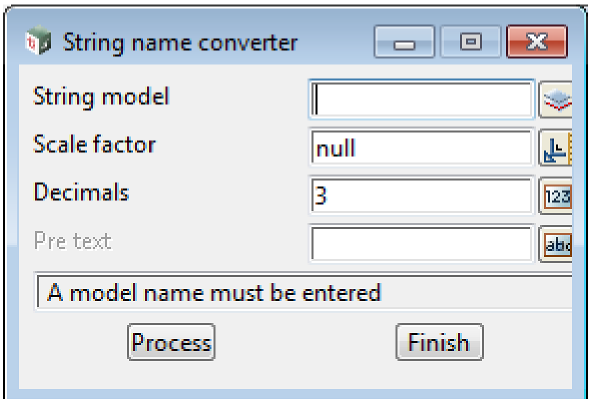
Note - This is an unsupported option which may not be fully documented.



29.3.29 XXX Scale String Names

Position of option on menu: **User =>Miscellaneous =>R to Z=>Scale string names**

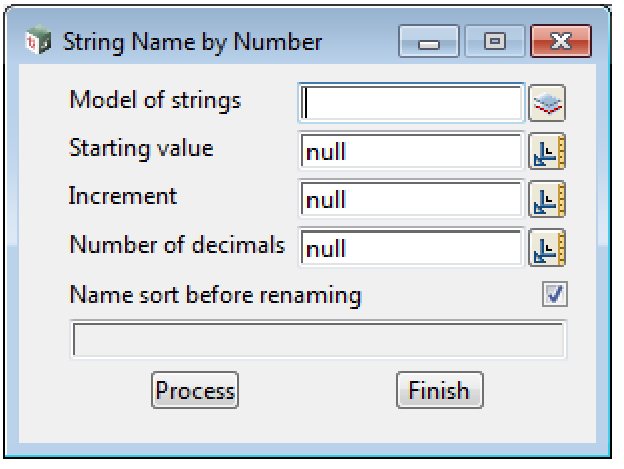
Note - This is an unsupported option which may not be fully documented.



29.3.30 XXX Set String Names by Number

Position of option on menu: **User =>Miscellaneous =>R to Z=>Set string names by number**

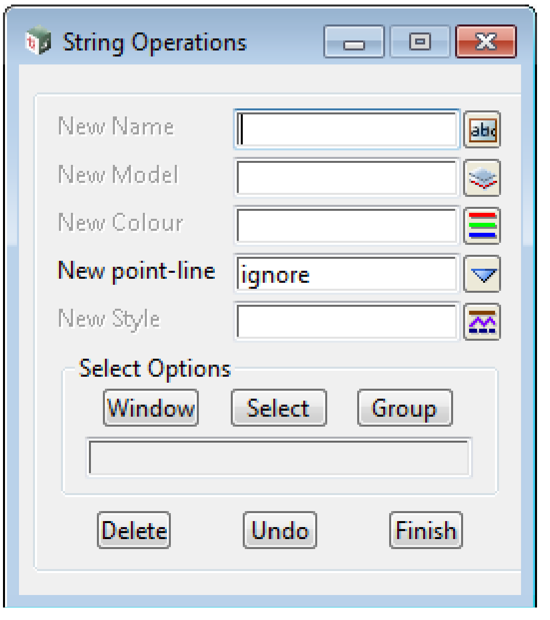
Note - This is an unsupported option which may not be fully documented.



29.3.31 XXX String Operations

Position of option on menu: **User =>Miscellaneous =>R to Z=>Set string properties**

Note - This is an unsupported option which may not be fully documented.

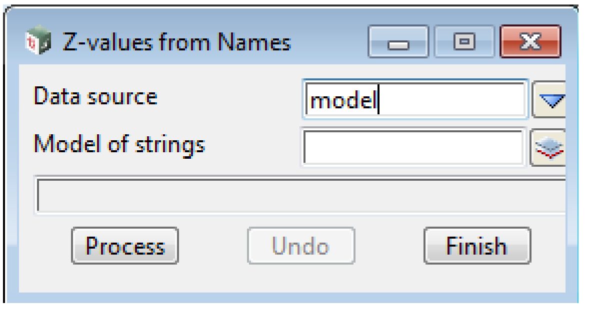


29.3.32 Set Contour Levels from String Names

Position of option on menu: **User =>Miscellaneous =>R to Z=>Set contour level from contour name**

Note - This is an unsupported option which may not be fully documented.

This operates on contour strings and sets the contour z-value to be the value of the name of the contour string. The z-values may need to be factored to bring them into the correct units.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Data source	data source	Model	Model, View

Type of data to set contour z-values for.

Model/View	data source		

Data source to of contours to set z-values for.

Process	button		

Run the option

Undo	button		

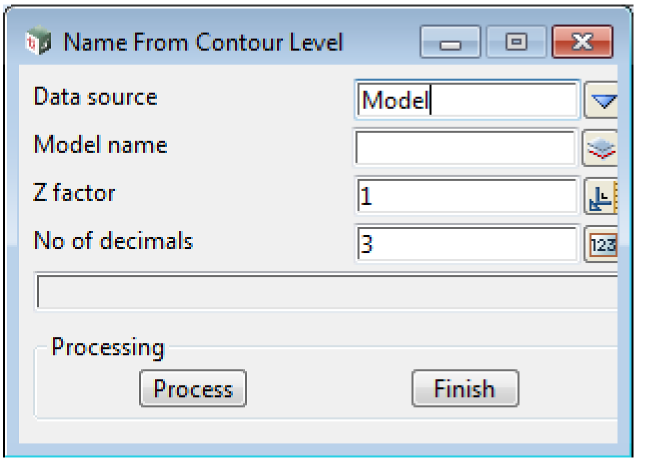
Undo the last set of contours labelled since the panel was up.

29.3.33 Set Contour Names From Contour Levels

Position of option on menu: **User =>Miscellaneous =>R to Z=>Set contour names from contour levels**

Note - This is an unsupported option which may not be fully documented.

This option operates on contours only and sets the string name to be the contour z value multiplied by the given factor.



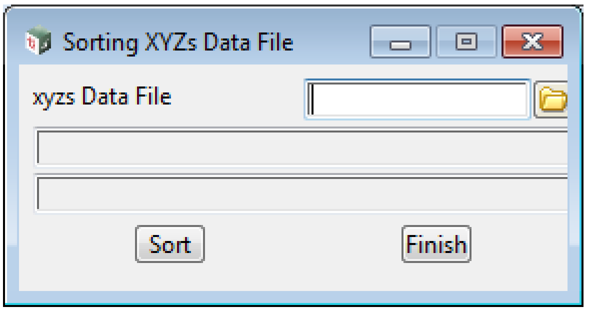
The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Data source	data source	Model	String, Model, View
Type of data to label contours strings for.			
Model/View/String	data source		
Data source to calculate label the contours for.			
Z factor	input	1	
Value to multiply the z values by before creating the string name.			
No of decimals	input	3	
Number of decimals places to use in the string name.			
Process	button		
Run the option.			

29.3.34 XXX Sort XYZs Data File

Position of option on menu: **User =>Miscellaneous =>R to Z=>Sort xyz data file**

Note - This is an unsupported option which may not be fully documented.



29.3.35 String Clipboard

Position of option on menu: **User =>Miscellaneous =>R to Z=>String attributes-properties to-from clipboard**

Position of option on menu: **File I/O=>User =>String attributes-properties to-from clipboard**

Note - This is an unsupported option which may not be fully documented.

On selecting the **String attributes-properties to-from clipboard** option, the **String clipboard** panel is displayed.

This panel can be used to export string properties and attributes to the windows clipboard and import the properties to update the same strings.

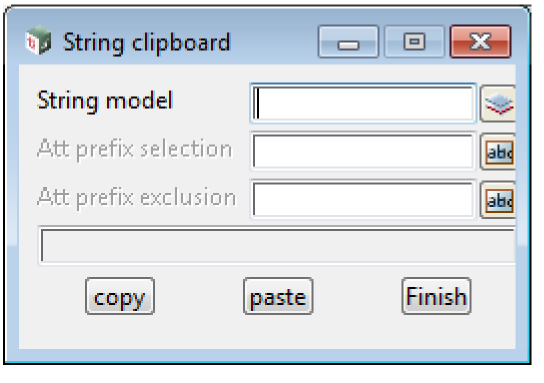
The following properties are always exported when **copy** is selected:

String id	Name	Type	Breakline	Style	Colour	Length
1	3	3	1	3	1	3
3322187D1		Drainage	1	1	1004	238.358

String id, **Type** and **Length** are for export only and cannot be changed on an import. **Name, Breakline, Style** and **Colour** may be changed to update the string when **paste** is selected. **Style** (linestyle order number from linestyl.4d) and **Colour** (colours.4d) numbers are used.

Starting in the 8th column, all of the string attributes are exported unless the **Att prefix selection** and/or the **Att prefix exclusion** filters are used. The first row is the string attribute name and the second row contains a number indicating the type of attribute. Attribute type numbers are

1	integer numbers (no decimals)
2	real numbers (decimals allowed)
3	text (characters allowed)



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
String model	model box		

name of the model to select strings from.

Att prefix selection	input	last data used	

(copy only) if blank all attributes will be selected prior to the exclusion. If text is entered then only attributes with this exact text (case sensitive) will be selected prior to the exclusion.

Att prefix exclusion	input	last data used	

(copy only) if blank all attributes matching the selection above will be exported. If text is entered then attributes with this exact text (case sensitive) will be excluded from the export.

Copy button

copies the data from the string to the windows clipboard

Paste button

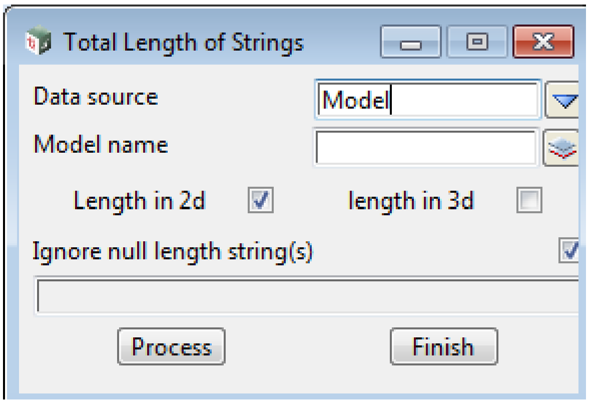
pastes the data from the windows clipboard. The string id is used to locate the string in the given model and then the properties/attributes are created and/or updated. If the attribute name is prefixed by DELETE followed by a space this string attribute will be deleted from all strings listed in the import.

29.3.36 Total Length of Strings

Position of option on menu: **User =>Miscellaneous =>R to Z=>Sum length of all strings**

Note - This is an unsupported option which may not be fully documented.

This panel is used to calculate the total length of the strings in a selected model or view.



The fields and buttons used in this panel have the following functions:

Field Description	Type	Defaults	Pop-Up
Data source			model, view, string

data source type

Data source

data source to calculate lengths for.

Length in 2d/3d	radio button	2d

if 2d, the lengths are only calculated in 2d.

If 3d, the lengths are calculated in 3d.

Ignore null length string(s)	tick box	ticked

if ticked, strings of null length are ignored in the string count and hence don't affect the average length.

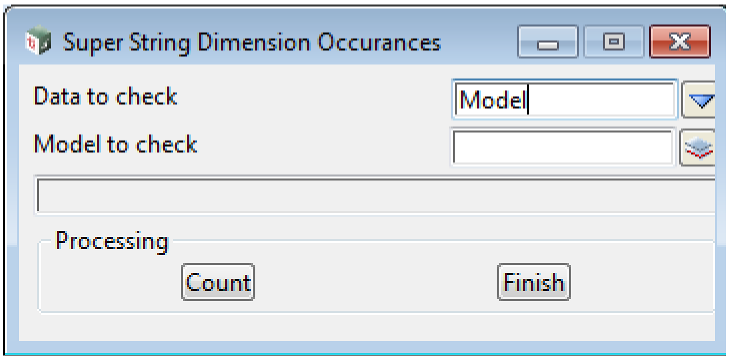
Process	button

run the option

29.3.37 XXX Super String Dimension Occurrences

Position of option on menu: **User =>Miscellaneous =>R to Z=>Super string dimension occurrences**

Note - This is an unsupported option which may not be fully documented.

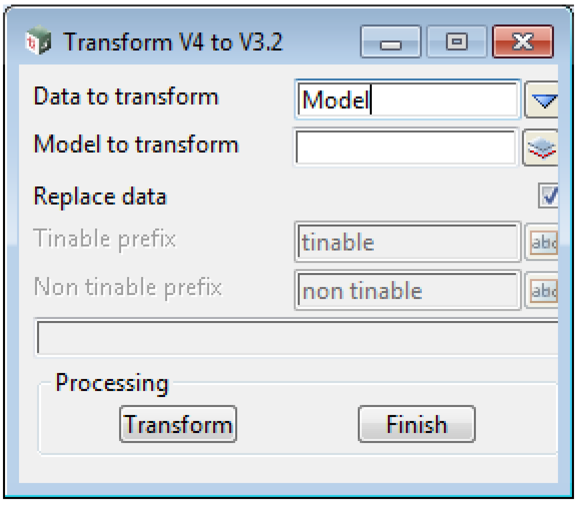


29.3.38 XXX Transform V4 to V3.2

Position of option on menu: **User =>Miscellaneous =>R to Z=>Super string to V3.2 strings**

Note - This is an unsupported option which may not be fully documented.

This panel convert a super string from ***12d Model*** V4.0 format to V3.2 strings.

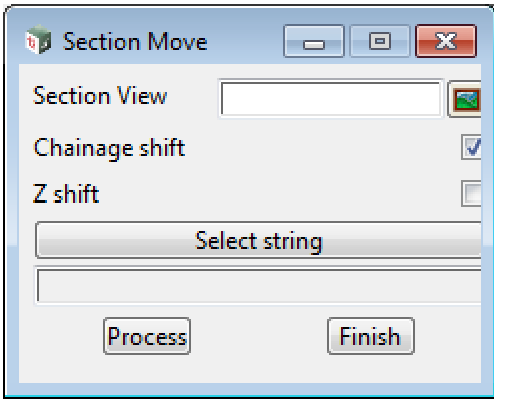


29.3.39 XXX Section Move

Position of option on menu: **User =>Miscellaneous =>R to Z=>Translate string in section view**

Note - This is an unsupported option which may not be fully documented.

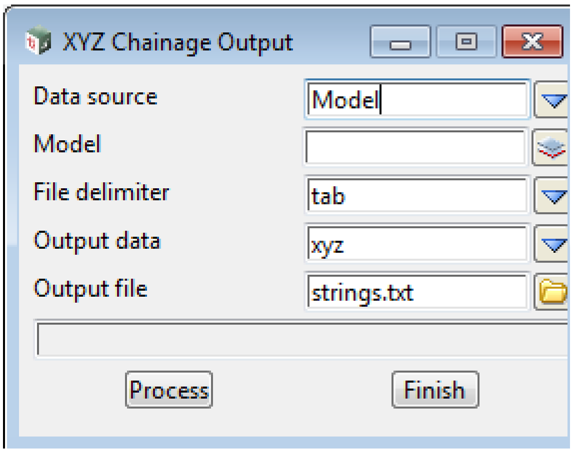
This panel is used to translate a string on a section view.



29.3.40 XXX Output XYZ and Chainage

Position of option on menu: **User =>Miscellaneous =>R to Z=>XYZ chainage**

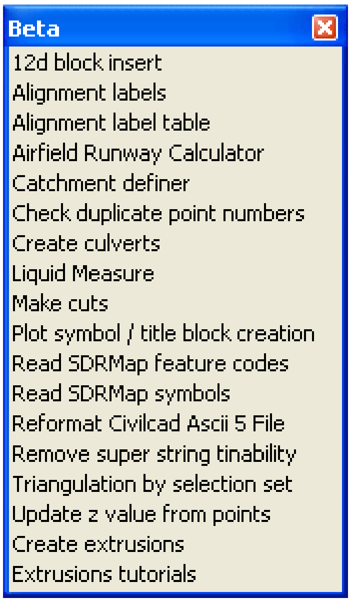
Note - This is an unsupported option which may not be fully documented.



29.4 Beta

All the options under **User** and **User menus** are either customers own macros or unsupported macros from 12D Solutions which may not be documented. For more information on *User* menus, go to 29.1 User Defined Menus

The walk rights **Beta** menu is:

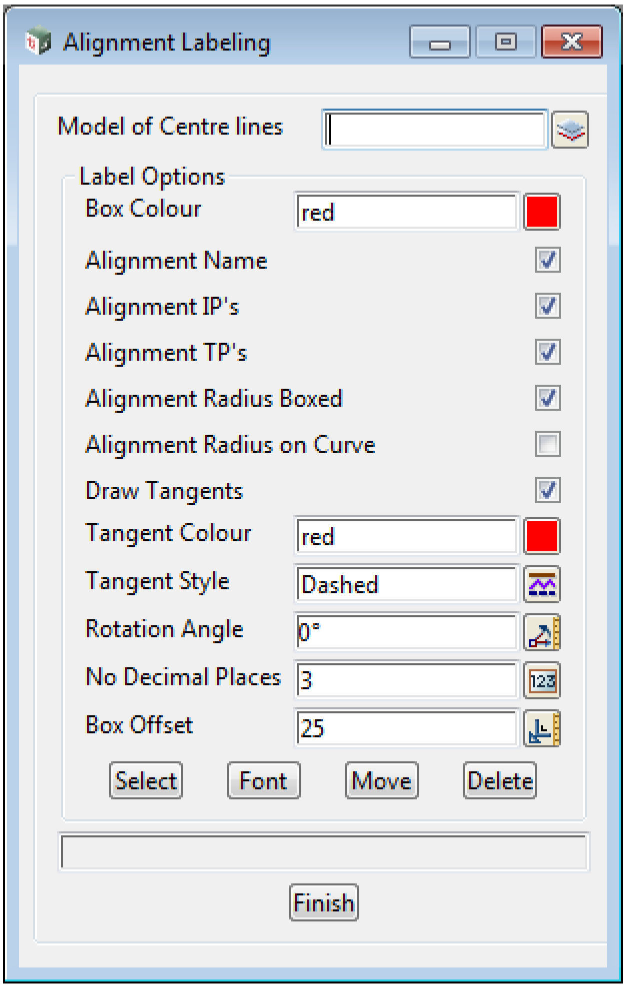


For the option *Alignment labels*, go to	29.4.1 XXX Alignment Labels
Alignment label table	29.4.2 XXX Table of Alignment Labels
Airfield Runway Calculator	29.4.3 Airfield Runway Calculator
Catchment definer	29.4.4 XXX Catchment Definer
Check duplicate point numbers	29.4.5 XXX Check Duplicate Point Numbers
Create culverts	29.4.6 XXX Create Culvert
Liquid measure	29.4.7 XXX Liquid Measure
Make cuts	29.4.8 XXX Make Cuts Through Strings
Plot symbol/title block	29.4.9 XXX Plot Symbol and Title Block Creation (old)
Read SDRMap feature codes	29.4.10 XXX Read SDRMap Feature Code
Read SDRMap symbols	29.4.11 XXX Read SDRMap Symbols
Reformat CivilCAD Ascii 5 file	29.4.12 XXX Reformat CivilCAD Ascii 5 file
Remove super string tinability	29.4.13 XXX Remove Super String Tinability
Triangulate by selection set	29.4.14 XXX Triangulate by Selection Set
Update z values from points	29.4.15 XXX Update Z Values from Points

29.4.1 XXX Alignment Labels

Position of option on menu: **User =>Beta =>Alignment labels**

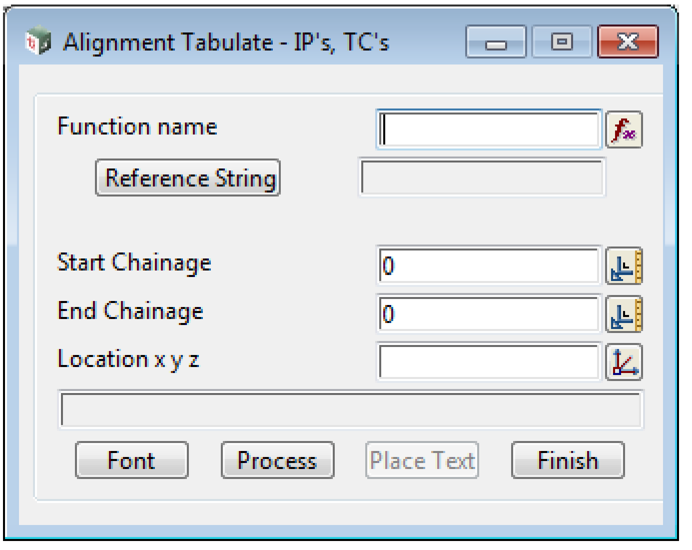
Note - This is an unsupported option which may not be fully documented.



29.4.2 XXX Table of Alignment Labels

Position of option on menu: **User =>Beta =>Alignment label table**

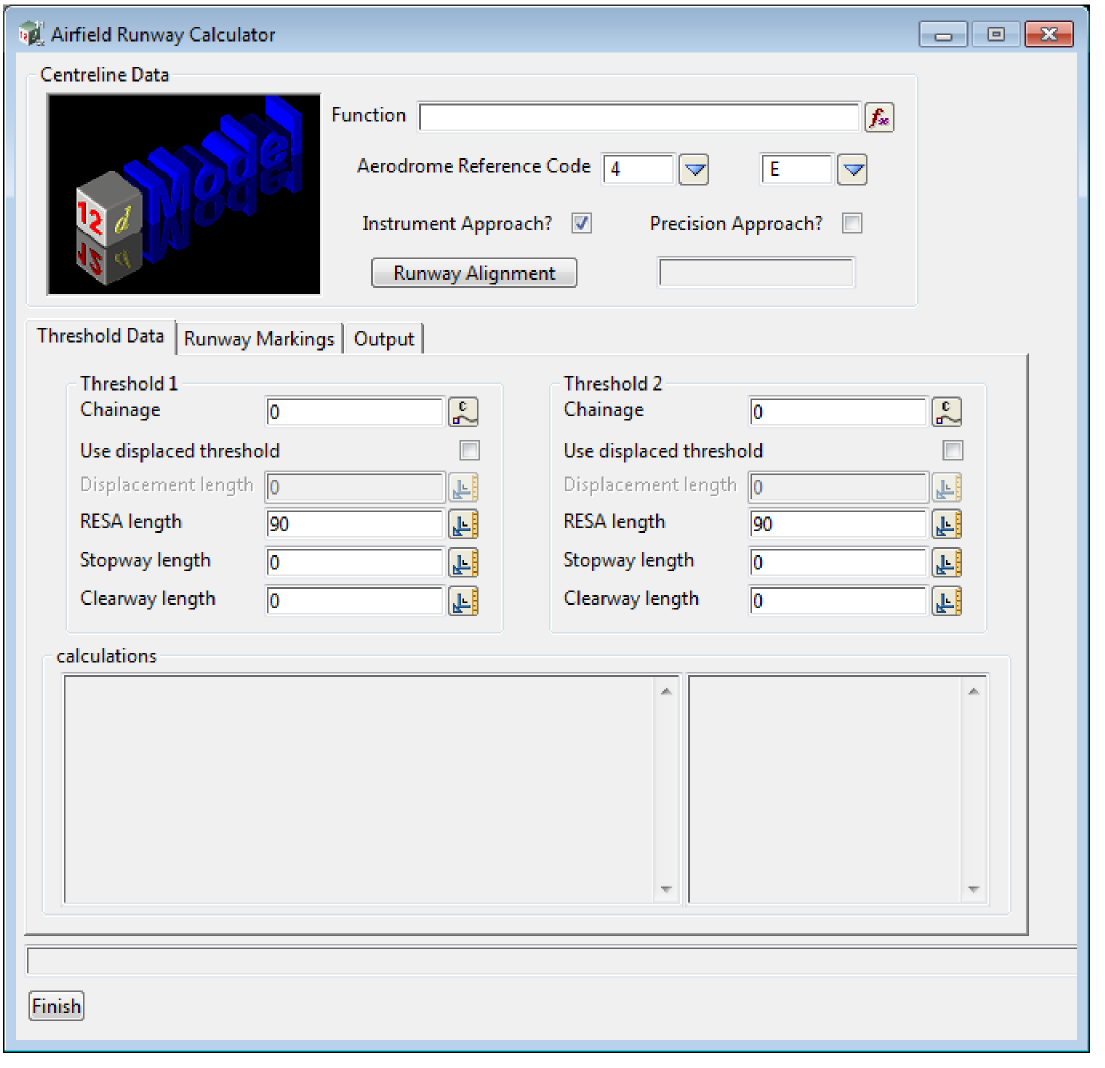
Note - This is an unsupported option which may not be fully documented.



29.4.3 Airfield Runway Calculator

Position of option on menu: **User =>Beta =>Airfield Runway Calculator**

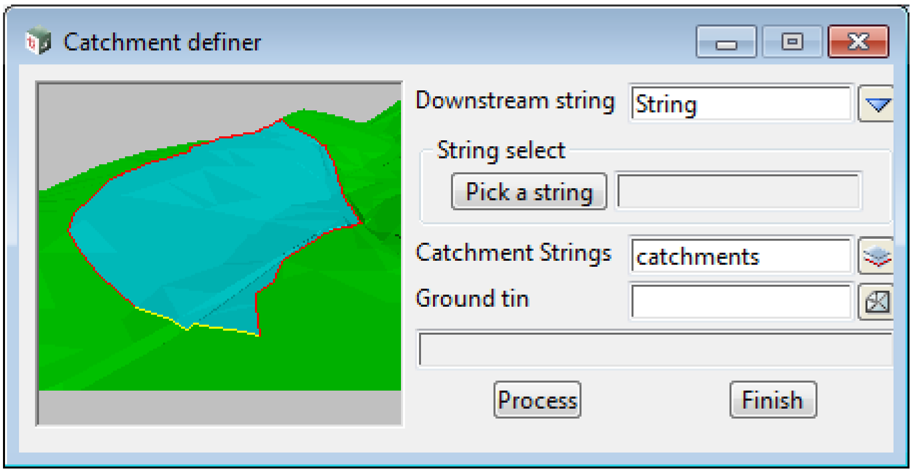
This option is under development



29.4.4 XXX Catchment Definer

Position of option on menu: **User =>Beta =>Catchment definer**

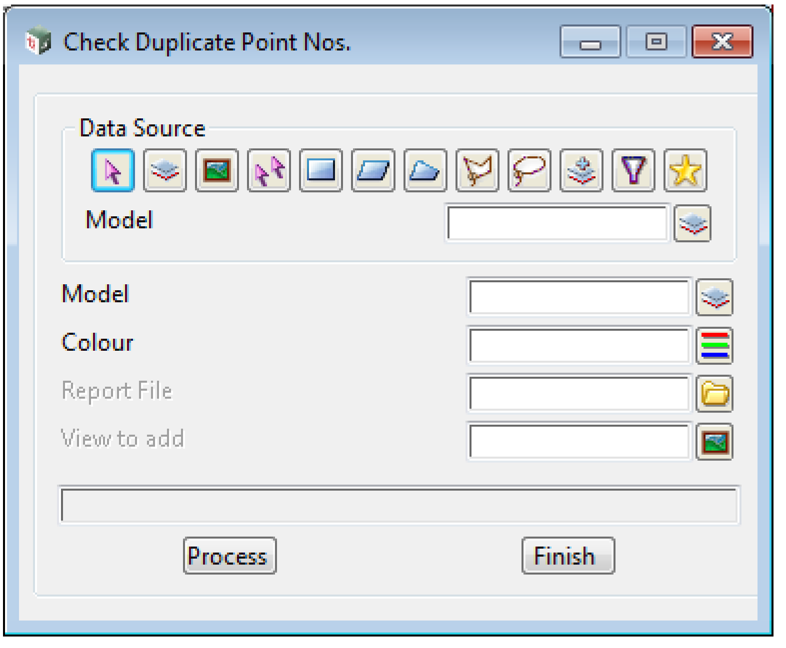
Note - This is an unsupported option which may not be fully documented.



29.4.5 XXX Check Duplicate Point Numbers

Position of option on menu: **User =>Beta =>Check duplicate point numbers**

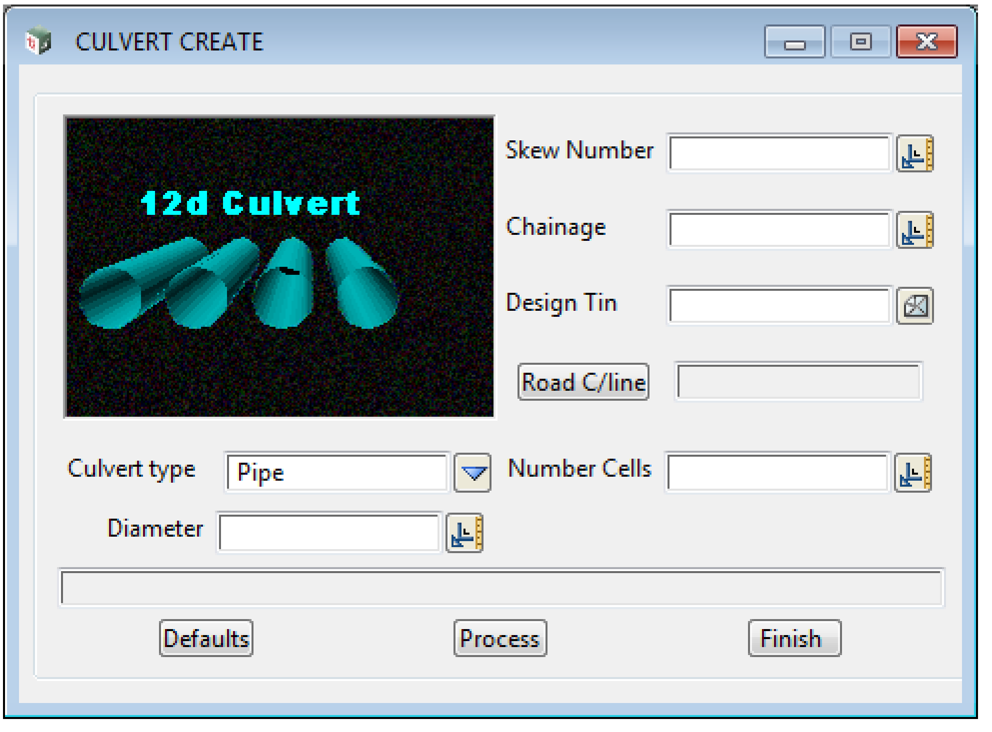
Note - This is an unsupported option which may not be fully documented.



29.4.6 XXX Create Culvert

Position of option on menu: **User =>Beta =>Create culvert**

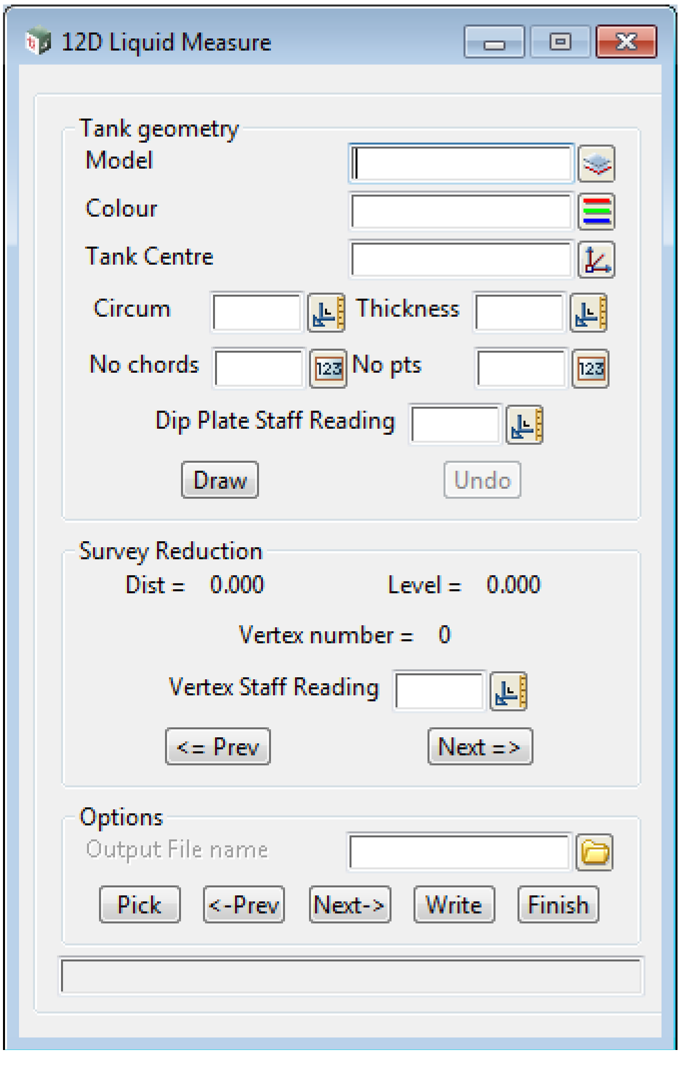
Note - This is an unsupported option which may not be fully documented.



29.4.7 XXX Liquid Measure

Position of option on menu: **User =>Beta =>Liquid measure**

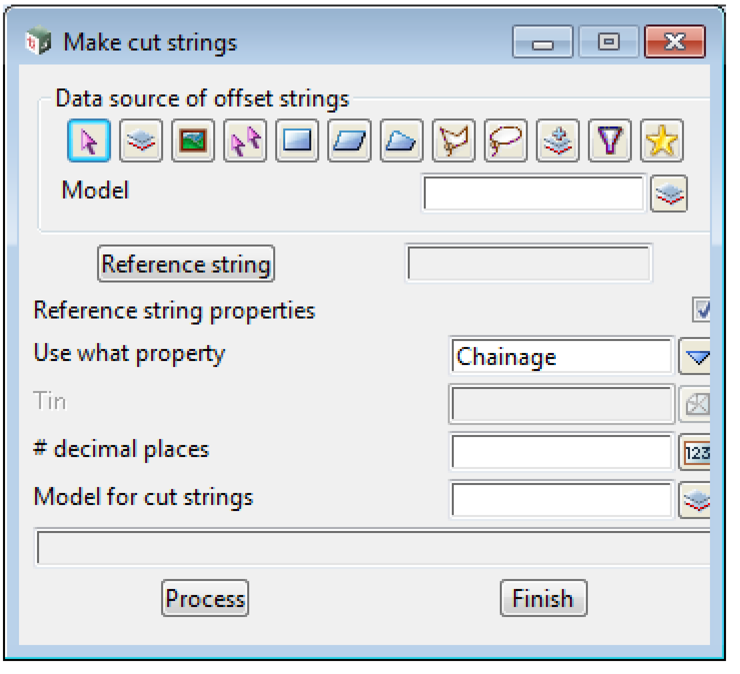
Note - This is an unsupported option which may not be fully documented.



29.4.8 XXX Make Cuts Through Strings

Position of option on menu: **User =>Beta =>Make cuts**

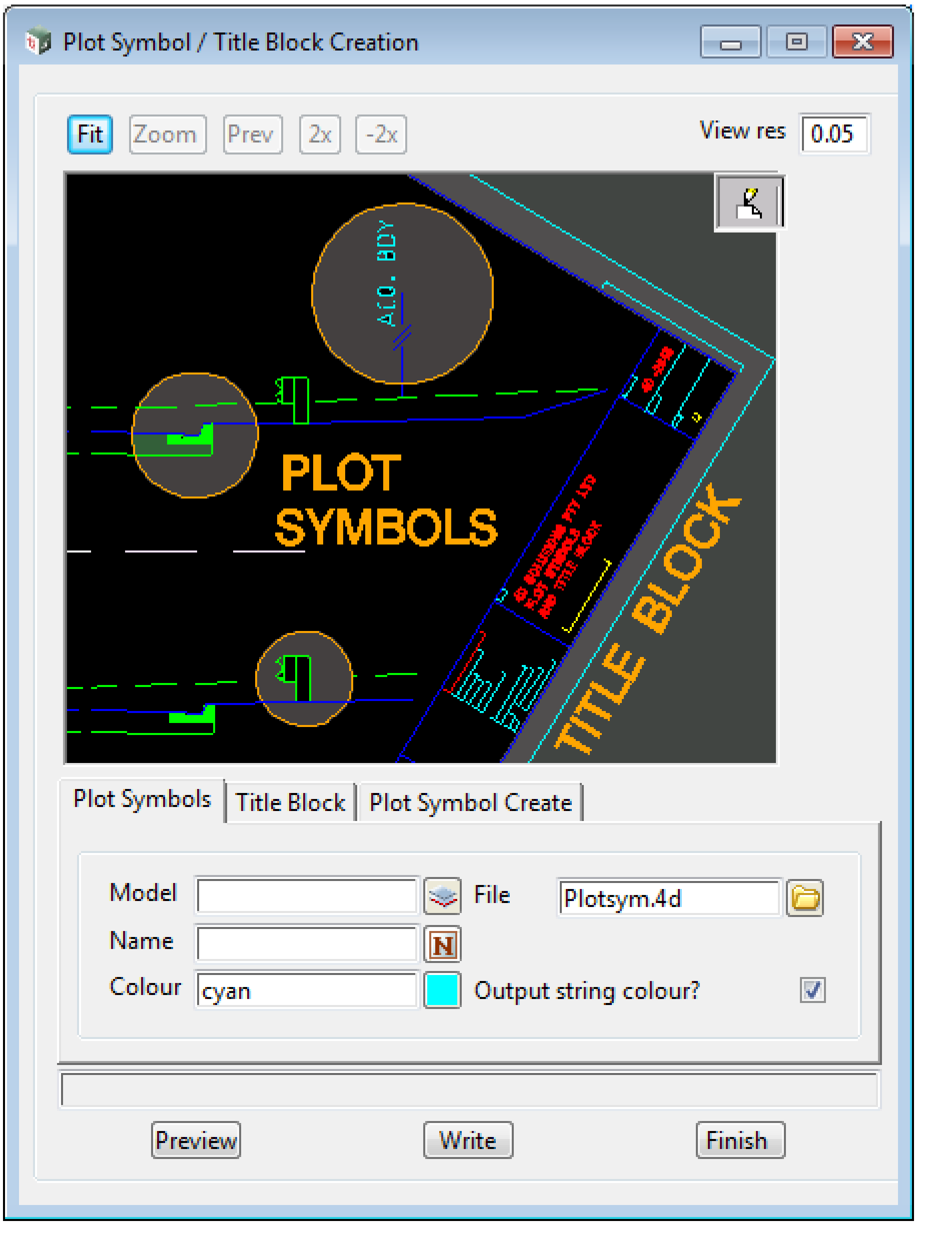
Note - This is an unsupported option which may not be fully documented.



29.4.9 XXX Plot Symbol and Title Block Creation (old)

Position of option on menu: **User =>Beta =>Plot symbol/title block creation**

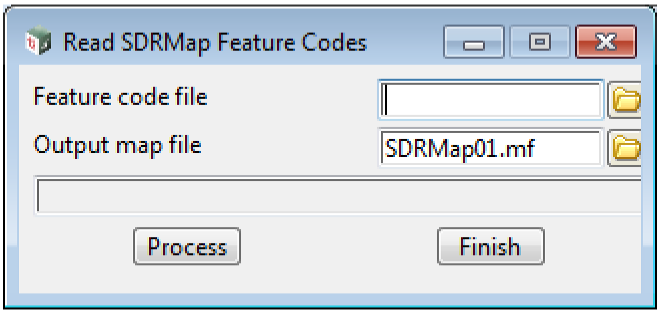
Note - This is an unsupported option which may not be fully documented.



29.4.10 XXX Read SDRMap Feature Code

Position of option on menu: **User =>Beta =>Read SDRMap feature codes**

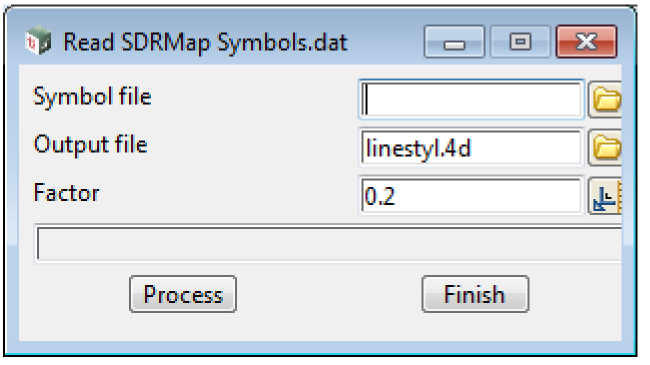
Note - This is an unsupported option which may not be fully documented.



29.4.11 XXX Read SDRMap Symbols

Position of option on menu: **User =>Beta =>Read SDRMap symbols**

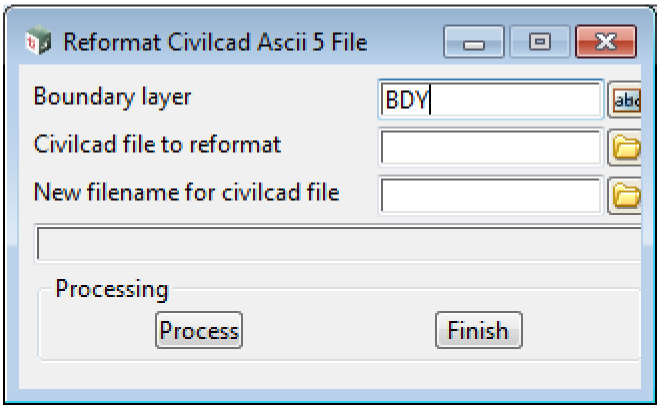
Note - This is an unsupported option which may not be fully documented.



29.4.12 XXX Reformat CivilCAD Ascii 5 file

Position of option on menu: **User =>Beta =>Reformat CivilCAD Ascii 5 file**

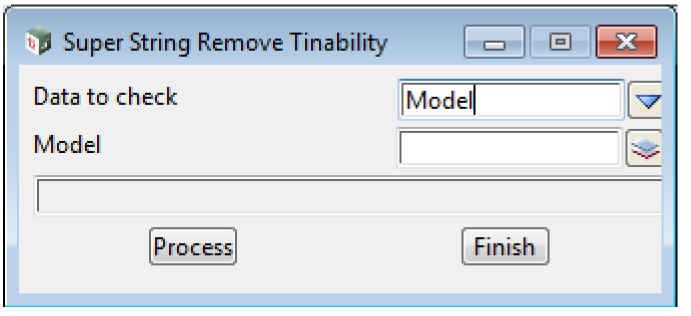
Note - This is an unsupported option which may not be fully documented.



29.4.13 XXX Remove Super String Tinability

Position of option on menu: **User =>Beta =>Remove super string tinability**

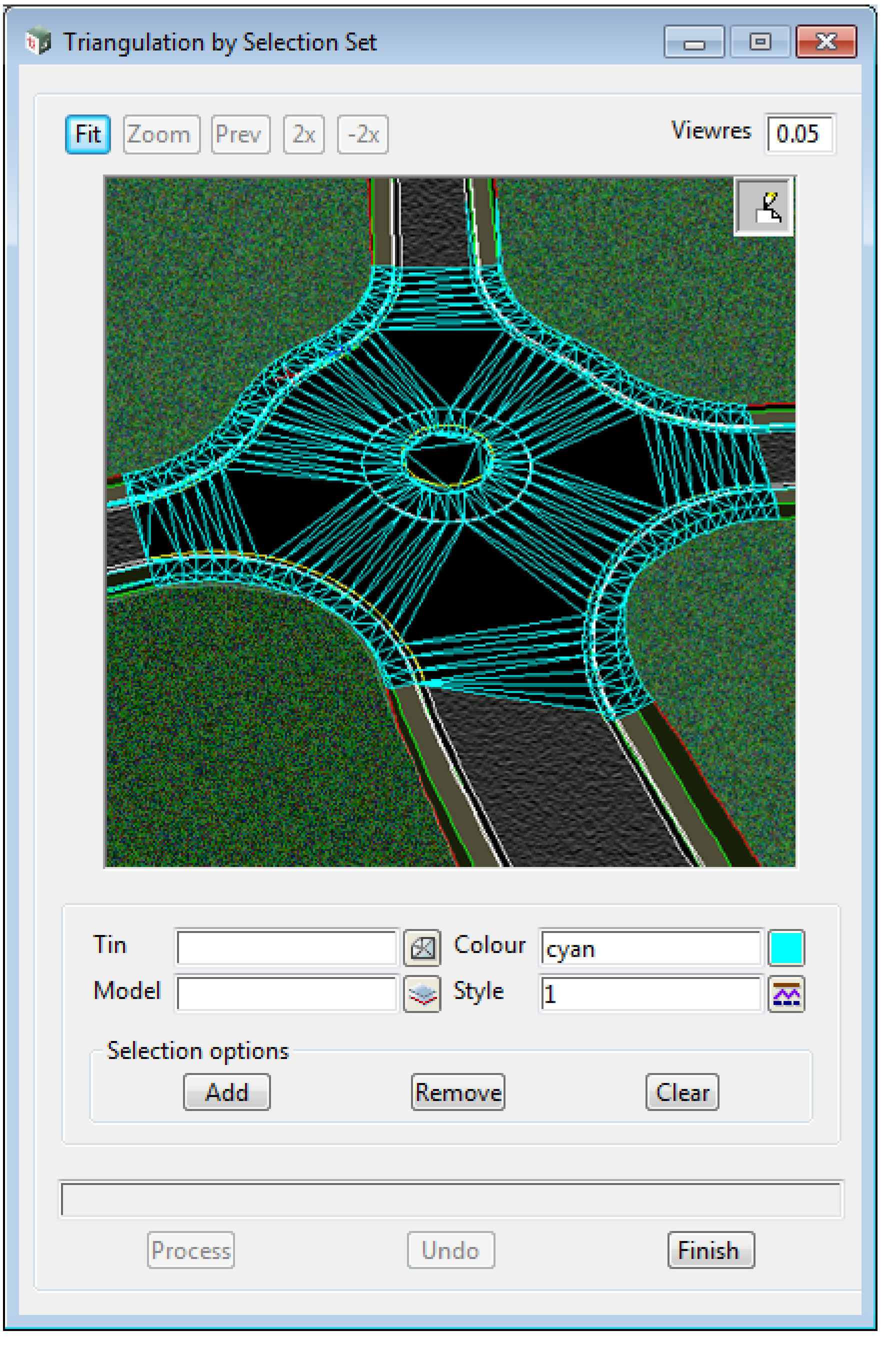
Note - This is an unsupported option which may not be fully documented.



29.4.14 XXX Triangulate by Selection Set

Position of option on menu: User =>Beta =>Triangulation by selection set

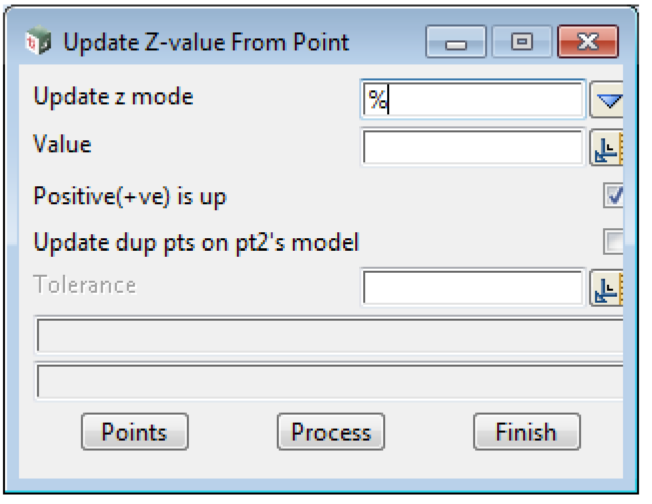
Note - This is an unsupported option which may not be fully documented.



29.4.15 XXX Update Z Values from Points

Position of option on menu: **User =>Beta =>Update z value from points**

Note - This is an unsupported option which may not be fully documented.



29.5 Console

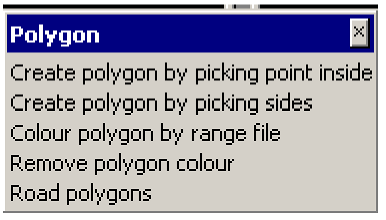
All the options under **User** and **User menus** are either customers own macros or unsupported macros from 12D Solutions which may not be documented. For more information on *User* menus, go to 29.1 User Defined Menus

All the macros on the **Console** menu only use the console panel and so do not support a F1 key for help.

The **User =>Console** rights menu is:

29.6 Polygon

The **Polygon** walk-right menu is



create polygon by picking point inside
create polygon by picking sides
colour polygon by range file
remove polygon colour

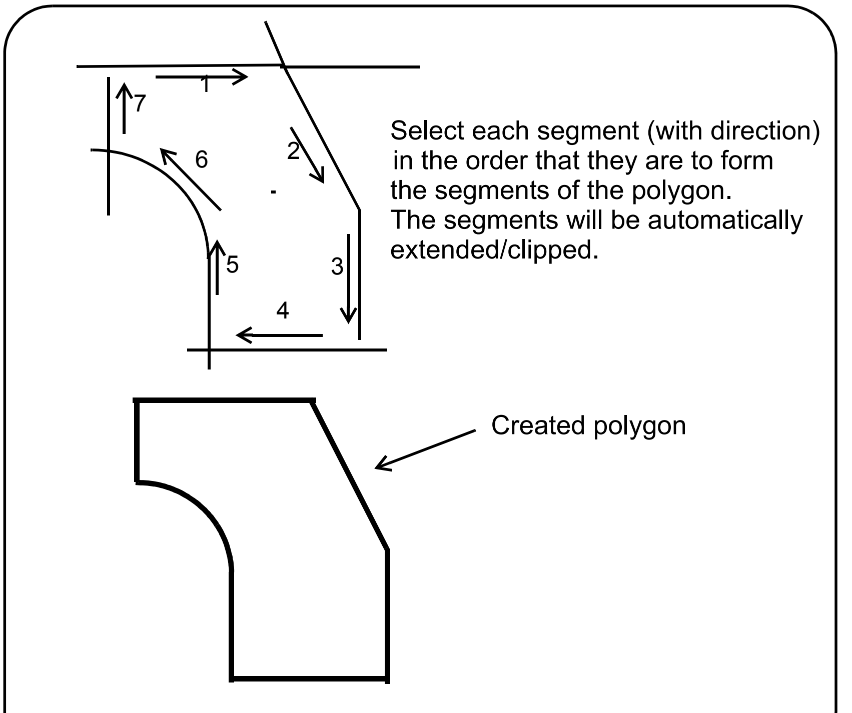
and each option in this menu will now be discussed.

For *Create polygon by picking point inside area*, go to 14.6.18 Surrounding Polygon
Create polygon by picking sides 29.6.1 Create Polygon by Picking Sides
Colour polygons from range file 29.6.2 Colour Polygons by Range File
Remove polygon colour 29.6.3 Remove Polygon Colour
Road polygons 20.8.12.3 Road Polygons

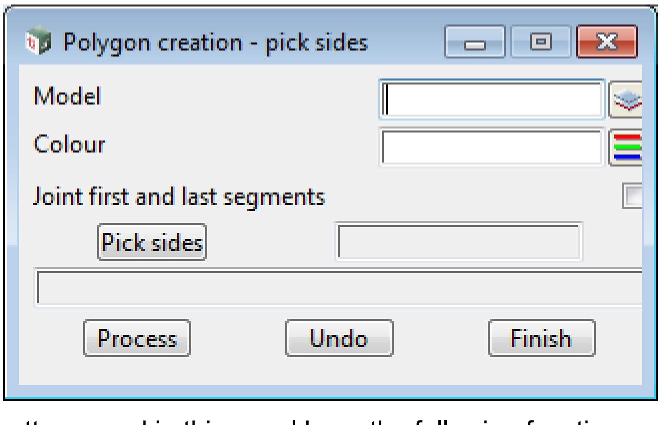
29.6.1 Create Polygon by Picking Sides

Position of option on menu: **User =>Polygon =>Create polygon by picking sides**

This option creates a polygon by a picking each segment in its order (and with direction) around the polygon. Segments will be automatically extended or clipped to form the polygon.



Creating a Polygon by Picking String Segments



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model for polygon	model box		available models
model for the created polygon			
Lot colour	colour box		available colours
colour for the created polygon			
Join first and last segment	tick box		

if tick, join the end of the last selected segment to the start of the first selected segment to form the final *side of the polygon. The segments may be automatically trimmed/extended.*

Pick sides string select

pick, with direction, the segments to be joined together to form the sides of the polygon.

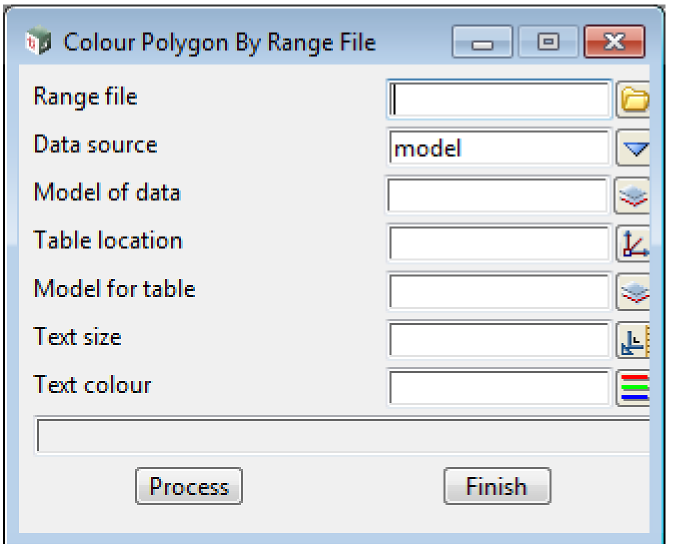
Process button

create the polygon from the selected segments.

29.6.2 Colour Polygons by Range File

Position of option on menu: **User =>Polygon =>Colour polygons by range file**

This option creates a polygon by a picking each segment in its order (and with direction) around the polygon. Segments will be automatically extended or clipped to form the polygon.



The fields and buttons used in this panel have the following functions.

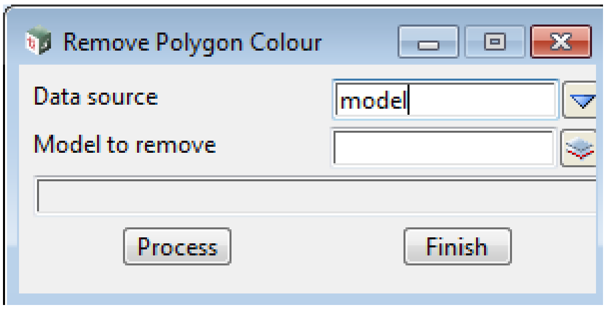
Field Description	Type	Defaults	Pop-Up
Range file	file box		*.lrf
range file for colouring polygons.			
Data source type		model	model, view
data source type.			
Data source			
data source for polygons to be coloured.			
Table location	position select box		
position of the table.			
Model for table	model box		available models
model for the table of statistics on the polygon areas.			
Text colour	colour box		available colours
colour of the text.			
Text size (w)	double box		
size of the text for the table.			
Process	button		
select all the polygons given by the Data source *and colour them according to the lot range file.*			

29.6.3 Remove Polygon Colour

Position of option on menu: **Design =>Estate/Lots =>Lot utilities =>Remove polygon colour**

Note - This is an unsupported option which may not be fully documented.

This option removes the colour of all the selected polygons.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Data source type		model	model, view

data source type.

Data source

data source for polygons to have their colour removed.

Process button

select all the polygons given by the Data source *and remove the polygon colour.*

29.7 Undocumented User Menus

The options on a **User** menu are either customer's macros or unsupported macros from 12D Solutions and may not be documented.

To return to the start of the documentation on the **User** menu, click on 29 User.

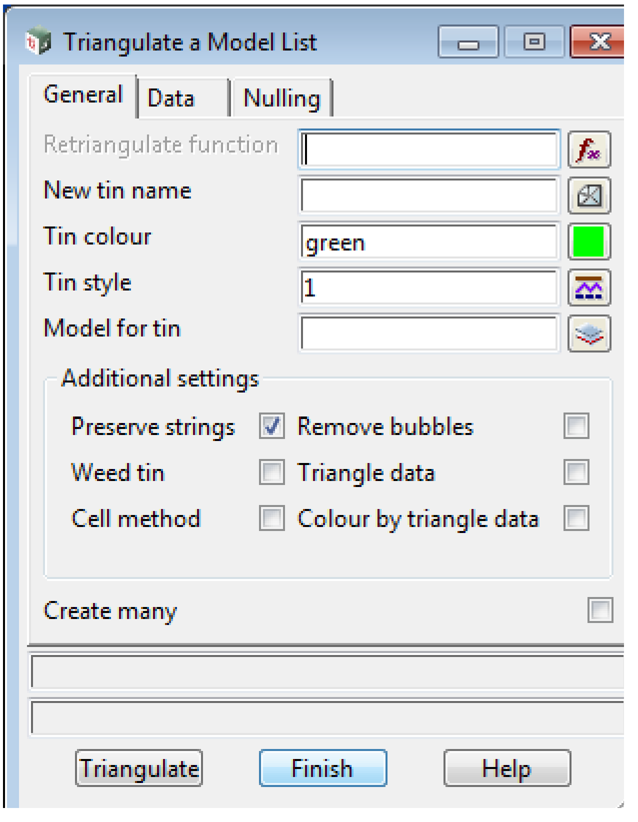
29.8 Obsolete Options

These options are now obsolete and may not be documented.

To return to the start of the documentation on the **User** menu, click on 29 User.

29.8.1 Triangulate a Model List

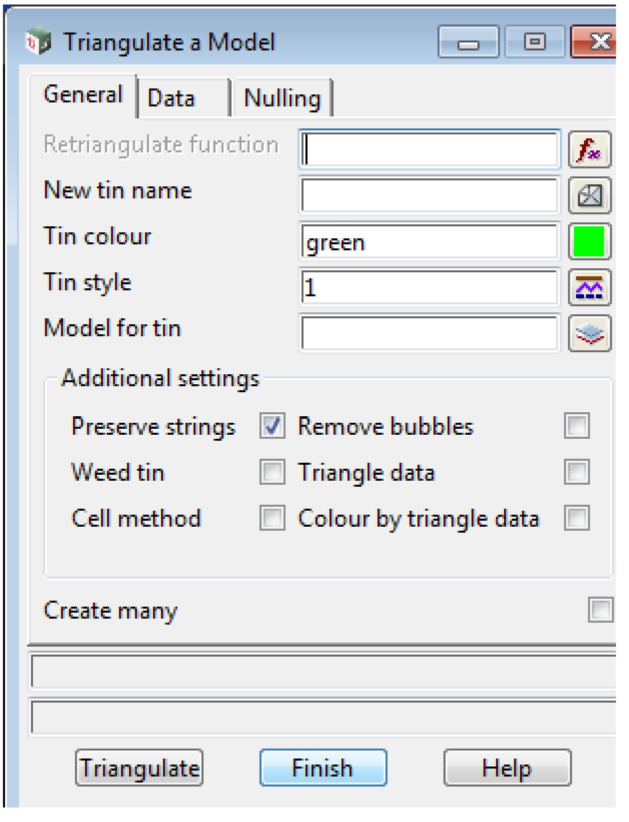
This is an obsolete option which is not documented.



To return to the start of the documentation on the **User** menu, click on 29 User.

29.8.2 Triangulate a Model

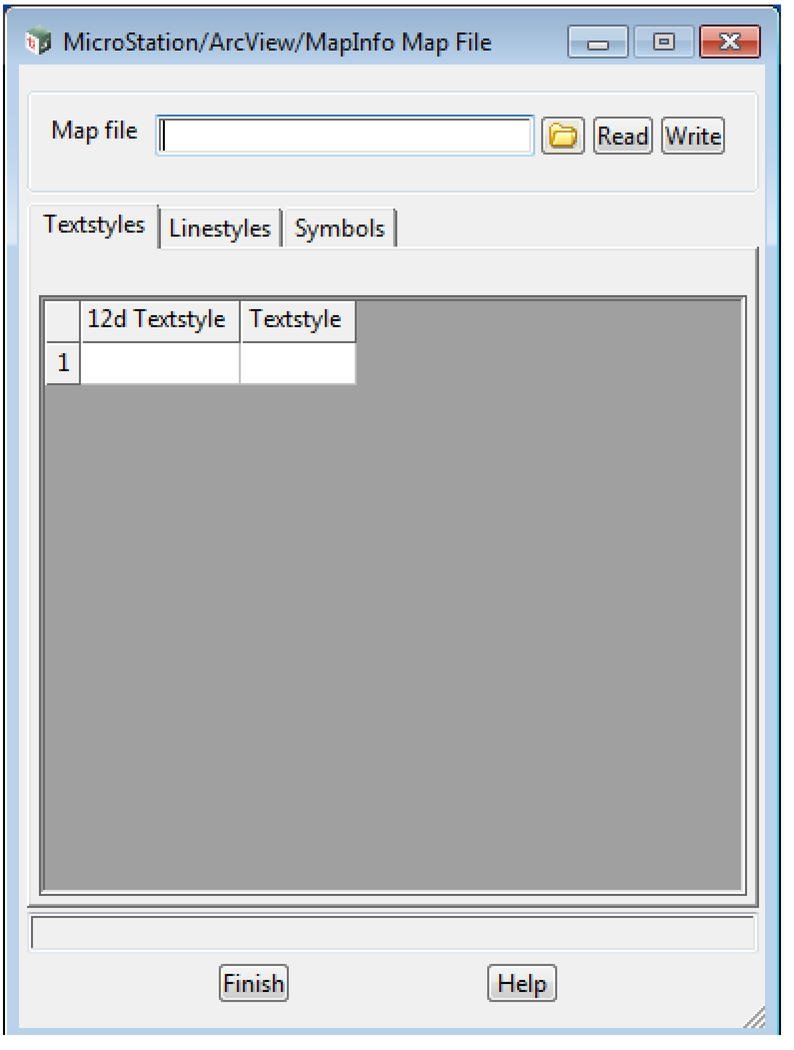
This is an obsolete option which is not documented.



To return to the start of the documentation on the **User** menu, click on 29 User.

29.8.3 MicroStation/ArcView/MapInfo Map File

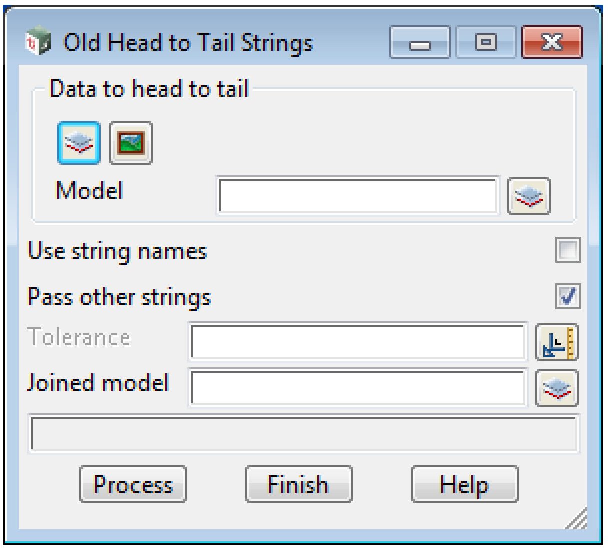
This is an obsolete option which is not documented.



To return to the start of the documentation on the **User** menu, click on 29 User.

29.8.4 Old Head to Tail Strings

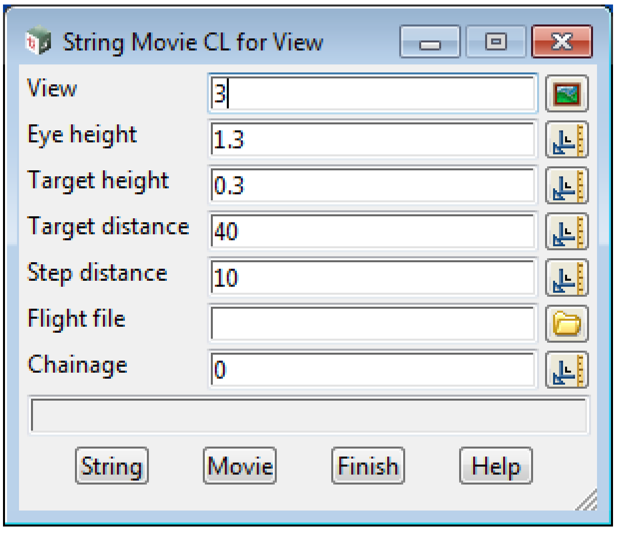
This is an obsolete option which is not documented.



To return to the start of the documentation on the **User** menu, click on 29 User.

29.8.5 String Movie CL for View

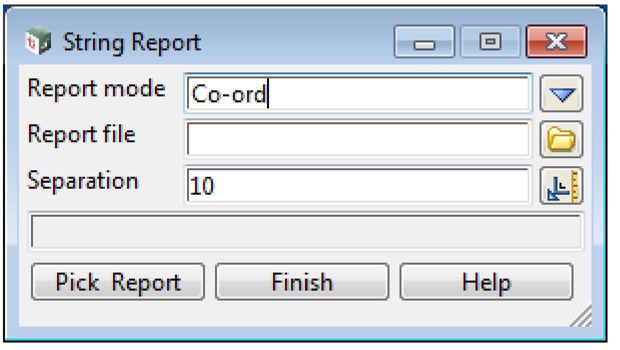
This is an obsolete option which is not documented.



To return to the start of the documentation on the **User** menu, click on 29 User.

29.8.6 String Report

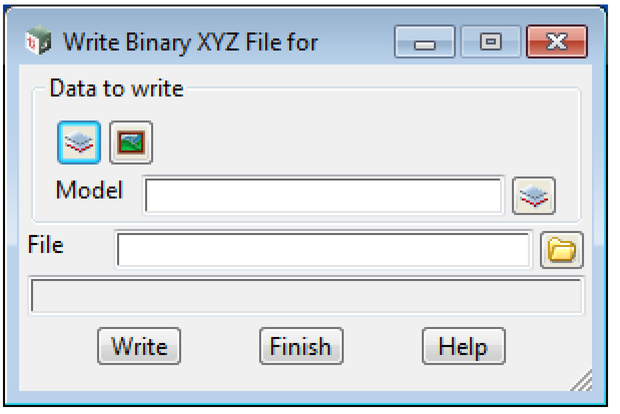
This is an obsolete option which is not documented.



To return to the start of the documentation on the **User** menu, click on 29 User.

29.8.7 Write Binary XYZ File For

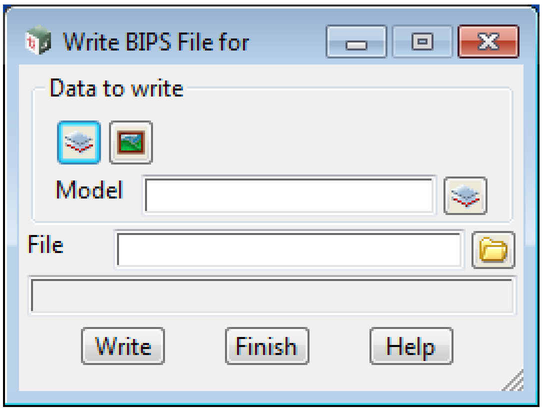
This is an obsolete option which is not documented.



To return to the start of the documentation on the **User** menu, click on 29 User.

29.8.8 Write BIPS File For

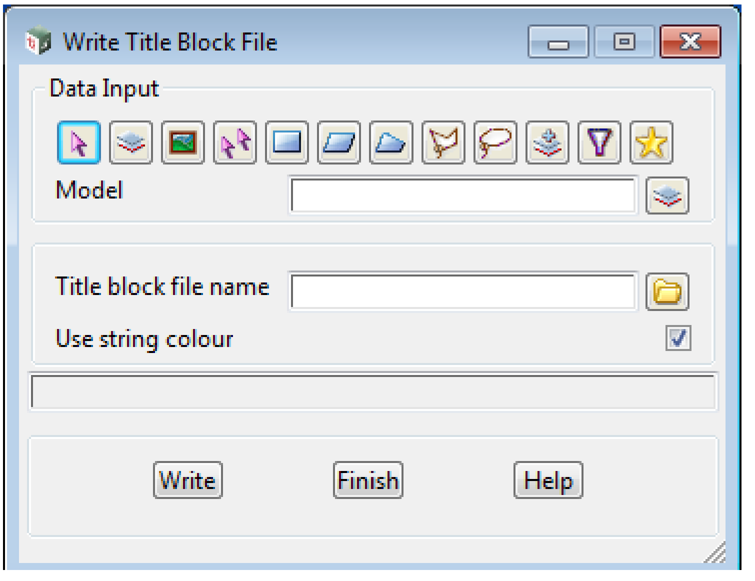
This is an obsolete option which is not documented.



To return to the start of the documentation on the **User** menu, click on 29 User.

29.8.9 Write Title Block File

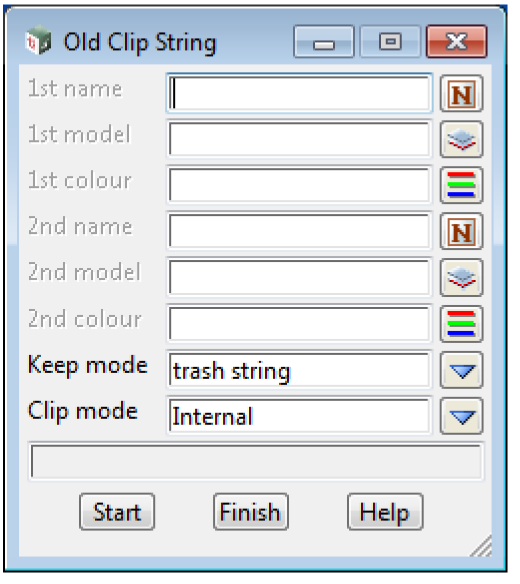
This is an obsolete option which is not documented.



To return to the start of the documentation on the **User** menu, click on 29 User.

29.8.10 Old Clip String

This is an obsolete option which is not documented.



To return to the start of the documentation on the **User** menu, click on 29 User.

29.9 Drainage Options

These options are part of the Drainage-Sewer menu.

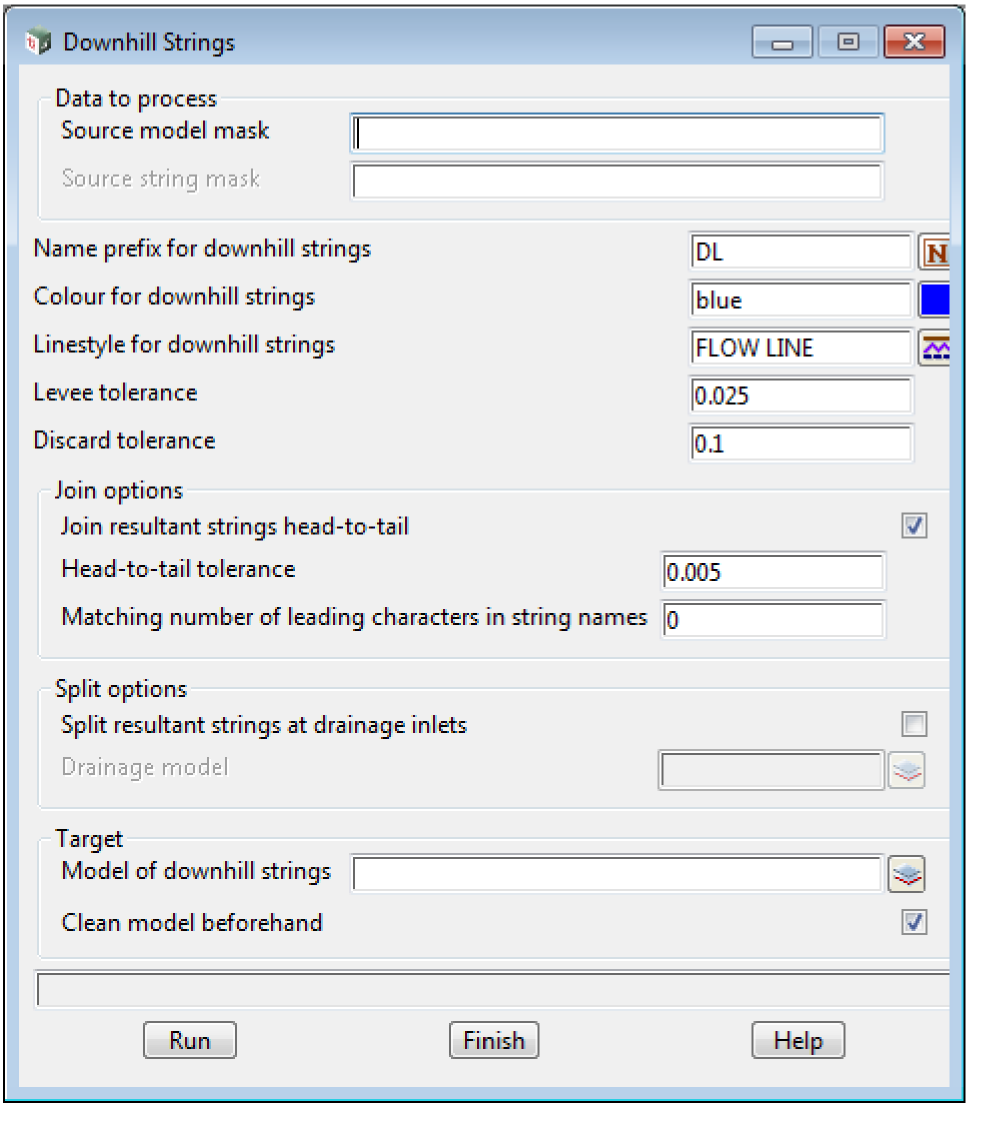
For the option	*Downhill strings, go to*	29.9.1 Downhill Strings
	Raindrop	29.9.2 Raindrop/Teardrop
	Aquaplaning risk	29.9.3 Aquaplaning Risk Assessment
	Extract sewer controls	29.9.4 Extract Sewer Controls
	Convert to pts and lines	29.9.5 Convert to Points and Lines

29.9.1 Downhill Strings

Position of option on menu: **Design=>Drainage-Sewer=>Downhill strings**

This option copies selected Super strings to a target model, where the strings are split at their crests and sags, and set in the downhill direction.

On selecting the **Downhill strings** option, the **Downhill Strings** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Source model mask	input text		

Specifies the model(s) from which to select Super strings to process. Use '' and '?' wildcards to specify multiple models. Note: the target model cannot be specified.*

Source string mask input text

Specifies the string name(s) to process from within the selected model(s). Use '' and '?' wildcards to specify multiple string names. Note: if blank, the option will match all string names in the selected model(s).*

Colour for downhill strings colour box available colours

Colour to apply to all strings in the target model (optional).

Linestyle for downhill strings linestyle box "FLOW LINE" available linestyles

Linestyle to apply to all strings in the target model (optional).

Levee tolerance input real 0.025

Localised uphill portions will be allowed to rise up within this vertical tolerance.

Discard tolerance input real 0.1

Resultant strings shorter than this tolerance will be discarded from the target model.

Join resultant strings head-to-tail tick box ticked

Whether to join strings in the target model in a head-to-tail fashion.

Head-to-tail tolerance input real 0.005

If joining strings head-to-tail, the head must be within this tolerance of the tail.

Matching number of leading characters in string names input integer0

If joining strings head-to-tail, this number of leading characters in both string names must match.

Split resultant strings at drainage inlets tick not ticked

Whether to split strings in the target model at nearby drainage inlets.

Drainage model model box available models

If splitting strings at drainage inlets, strings will be split at all nearby drainage inlet pits found in this model. Drainage inlets (i.e. on-grade and sag pits only) must be within one pit diameter of the string, and may not be closer than the Discard tolerance from either end of the string.

Model of downhill strings model box available models

Target model. All matching source strings are copied to and processed in this model.

Clean model beforehand tick box ticked

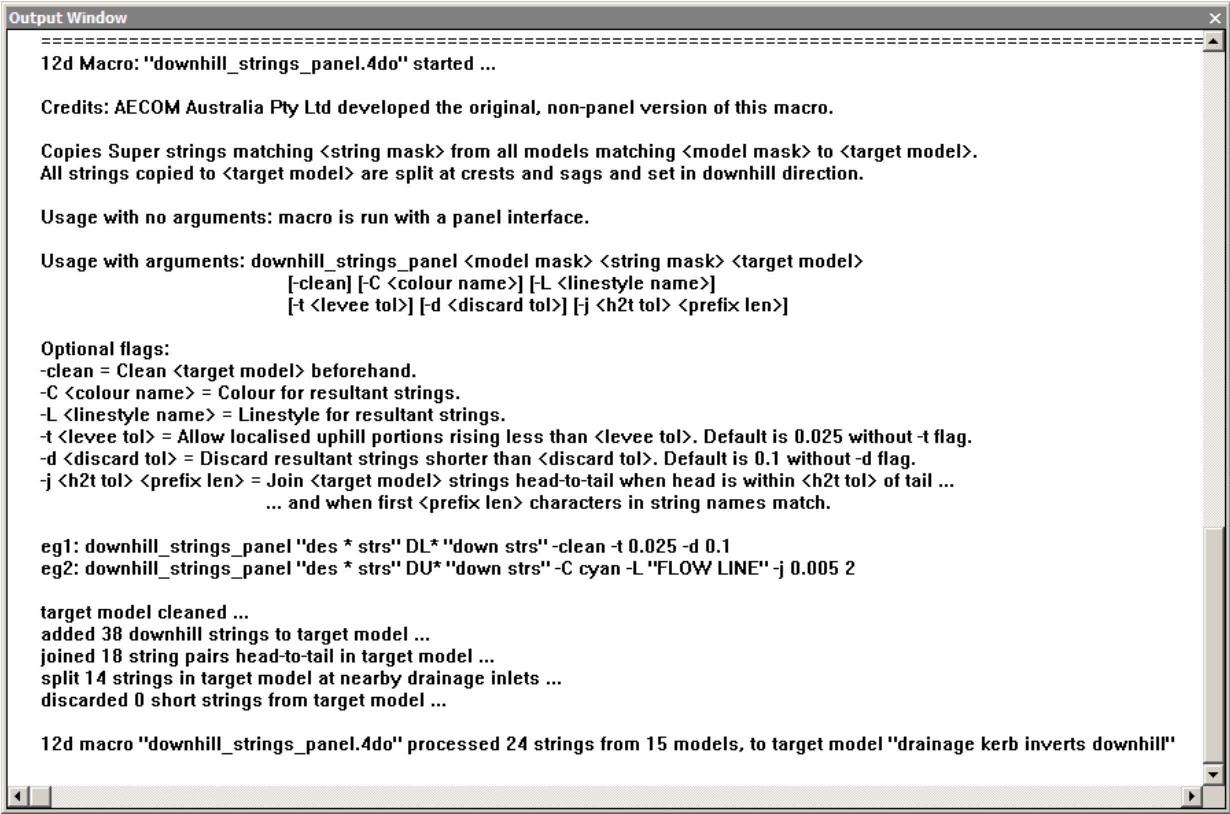
Whether to clean the target model beforehand.

Run button

Runs the option.

Special Note:

The **Downhill Strings** option is a macro, that may optionally be run from a chain via arguments, rather than via a panel. The details of the argument syntax are written to the Output Window, whenever the macro is run without valid arguments, viz:



```
Output Window
===========================================================================================================
12d Macro: "downhill_strings_panel.4do" started ...

Credits: AECOM Australia Pty Ltd developed the original, non-panel version of this macro.

Copies Super strings matching <string mask> from all models matching <model mask> to <target model>.
All strings copied to <target model> are split at crests and sags and set in downhill direction.

Usage with no arguments: macro is run with a panel interface.

Usage with arguments: downhill_strings_panel <model mask> <string mask> <target model>
                      [-clean] [-C <colour name>] [-L <linestyle name>]
                      [-t <levee tol>] [-d <discard tol>] [-j <h2t tol> <prefix len>]

Optional flags:
-clean = Clean <target model> beforehand.
-C <colour name> = Colour for resultant strings.
-L <linestyle name> = Linestyle for resultant strings.
-t <levee tol> = Allow localised uphill portions rising less than <levee tol>. Default is 0.025 without -t flag.
-d <discard tol> = Discard resultant strings shorter than <discard tol>. Default is 0.1 without -d flag.
-j <h2t tol> <prefix len> = Join <target model> strings head-to-tail when head is within <h2t tol> of tail ...
                    ... and when first <prefix len> characters in string names match.

eg1: downhill_strings_panel "des * strs" DL* "down strs" -clean -t 0.025 -d 0.1
eg2: downhill_strings_panel "des * strs" DU* "down strs" -C cyan -L "FLOW LINE" -j 0.005 2

target model cleaned ...
added 38 downhill strings to target model ...
joined 18 string pairs head-to-tail in target model ...
split 14 strings in target model at nearby drainage inlets ...
discarded 0 short strings from target model ...

12d macro "downhill_strings_panel.4do" processed 24 strings from 15 models, to target model "drainage kerb inverts downhill"
```

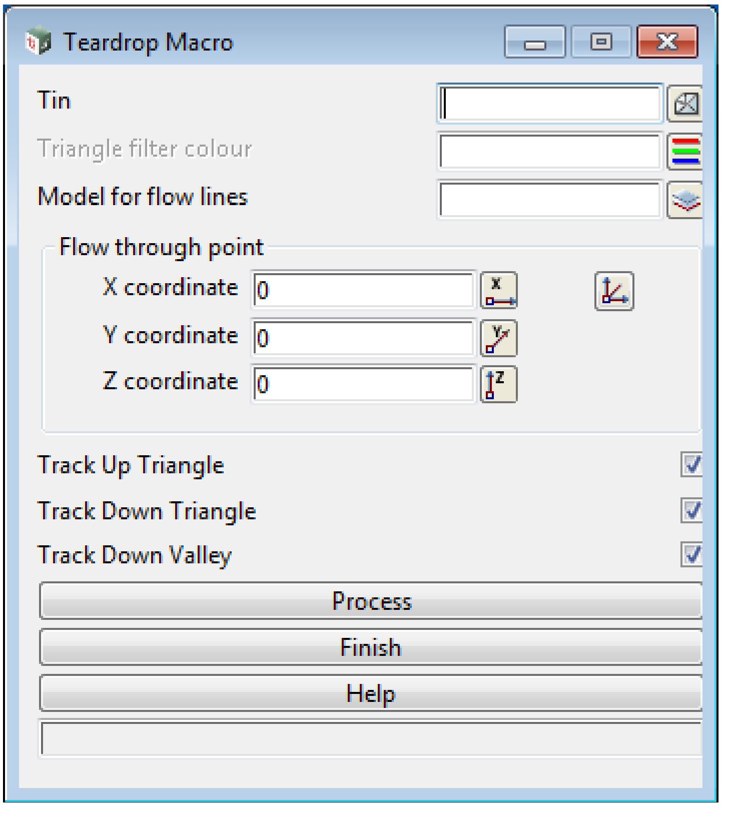
29.9.2 Raindrop/Teardrop

Position of option on menu: **Design=>Drainage-Sewer=>Raindrop**

Position of option on menu: **Tins=>Tin Analysis=>Rain drop**

This section of documentation is a work in progress and will be updated in subsequent releases.

On selecting the **Raindrop** option, the **Teardrop Macro** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Tin	tin box	GROUND	available tins
Triangle filter colour	colour box		available colours
Model for flow lines	model box	FLOWMODEL	available models

Flow through point

X coordinate	measure box	0	available measures
Y coordinate	measure box	0	available measures
Z coordinate	measure box	0	available measures
Track Up Triangle	tick box	ticked	
Track Down Triangle	tick box	ticked	
Track Down Valley	tick box	ticked	
Process	button		

29.9.3 Aquaplaning Risk Assessment

Position of option on menu: **Design=>Drainage-Sewer=>Aquaplaning risk**

This option performs an aquaplaning risk assessment using the *Gallaway Equation*. The user need only supply flow path strings as 2d Super strings in the areas of concern on the road pavement tin (typically at the transitions in the road cross-fall). The option then re-creates these flow path strings in 3d, with evenly spaced vertices, then applies the *Gallaway Equation* to determine a water film depth at each vertex (where all parameters considered are set as vertex attributes). It then assesses the risk of aquaplaning at each vertex by associating a risk level and segment colour with the water film depth calculated at each vertex. In addition, a report file is written to the *Windows* clipboard, and optionally to file.

The *Gallaway Equation* is defined as follows:

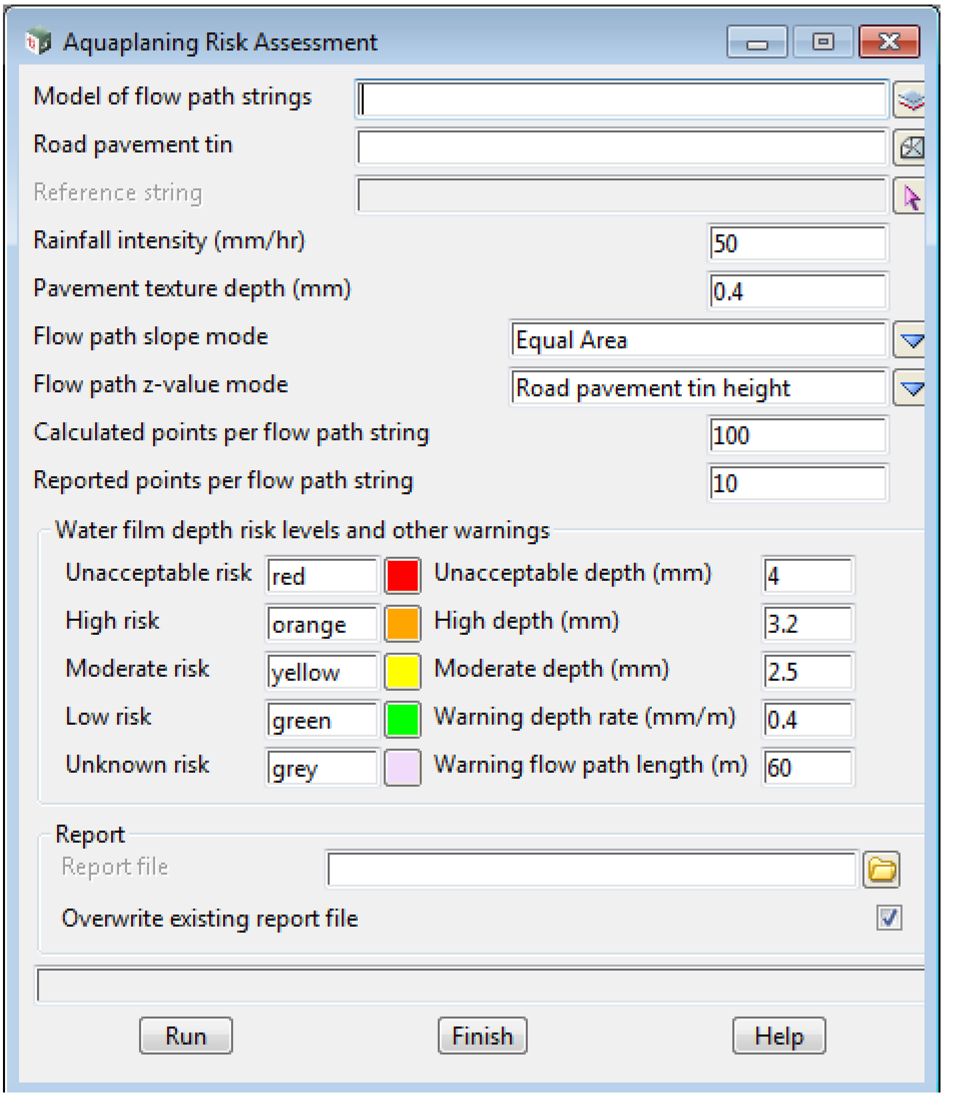
Gallaway Equation

$$d = \frac{0.103\, T^{0.11} L^{0.43} I^{0.59}}{S^{0.42}} - T$$

d = water film depth above top of pavement texture (mm)
L = length of flow path (m)
S = slope of the flow path (%)
T = average pavement texture depth (mm)
I = rainfall intensity (mm/hr)

Source:
Gallaway, B. M., et. al., *"Pavement and Geometric Design Criteria for Minimizing Hydroplaning", Federal Highway Administration, Report No. FHWARD-79-31, (1979).*

On selecting the **Aquaplaning risk** option, the **Aquaplaning Risk Assessment** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Model of flow path strings	model box		available models

All Super strings in this model will be analysed and updated with results.

Road pavement tin	tin box		available tins

Used to define the vertical profile of each flow path string.

Reference string	string select		

Used to locate, identify, rename and sort the flow path strings (optional).

Rainfall intensity (mm/hr)	input	50	

Typically 50 mm/hr for aquaplaning checks.

Pavement texture depth (mm)	input	0.4	

Average texture depths can range from about 0.2 to 4 mm for different pavement materials.

Flow path slope mode	choice box	Equal Area	Equal Area, Average

Determines how the slope of the flow path string is calculated at each vertex. Available modes are "Equal Area" and "Average" slope, from the start of the string to each vertex.

Flow path z-value mode choice box Road pavement tin height Road pavement tin height, Water film depth (mm)

Determines what the z-values of the flow path strings will represent. Available modes are "Road pavement tin height" and "Water film depth (mm)".

Calculated points per flow path string input 100

Number of evenly spaced points along each flow path string at which to calculate water film depths.

Reported points per flow path string input 10

Number of evenly spaced points along each flow path string at which to report water film depths.

Water film depth risk levels and other warnings

Unacceptable risk colour box red available colours

Colour to associate with an unacceptable risk.

Unacceptable depth (mm) input 4

The lowest water film depth associated with an unacceptable risk.

High risk colour box orange available colours

Colour to associate with a high (acceptable) risk.

High depth (mm) input 3.2

The lowest water film depth associated with a high (acceptable) risk.

Moderate risk colour box yellow available colours

Colour to associate with a moderate (acceptable) risk.

Moderate depth (mm) input 2.5

The lowest water film depth associated with a moderate (acceptable) risk.

Low risk colour box green available colours

Colour to associate with a low (acceptable) risk.

Unknown risk colour box grey available colours

Colour for strings that cannot be assessed for risk.

Warning depth rate (mm/m) input 0.4

Warn if <water film depth>/<flow path length> is excessive (optional).

Warning flow path length (m) input 60

Warn if flow path length is excessive (optional).

Report

Report file file box

Report will be written in tab-delimited format to the Windows clipboard, and optionally to this file.

Overwrite existing report file tick box ticked

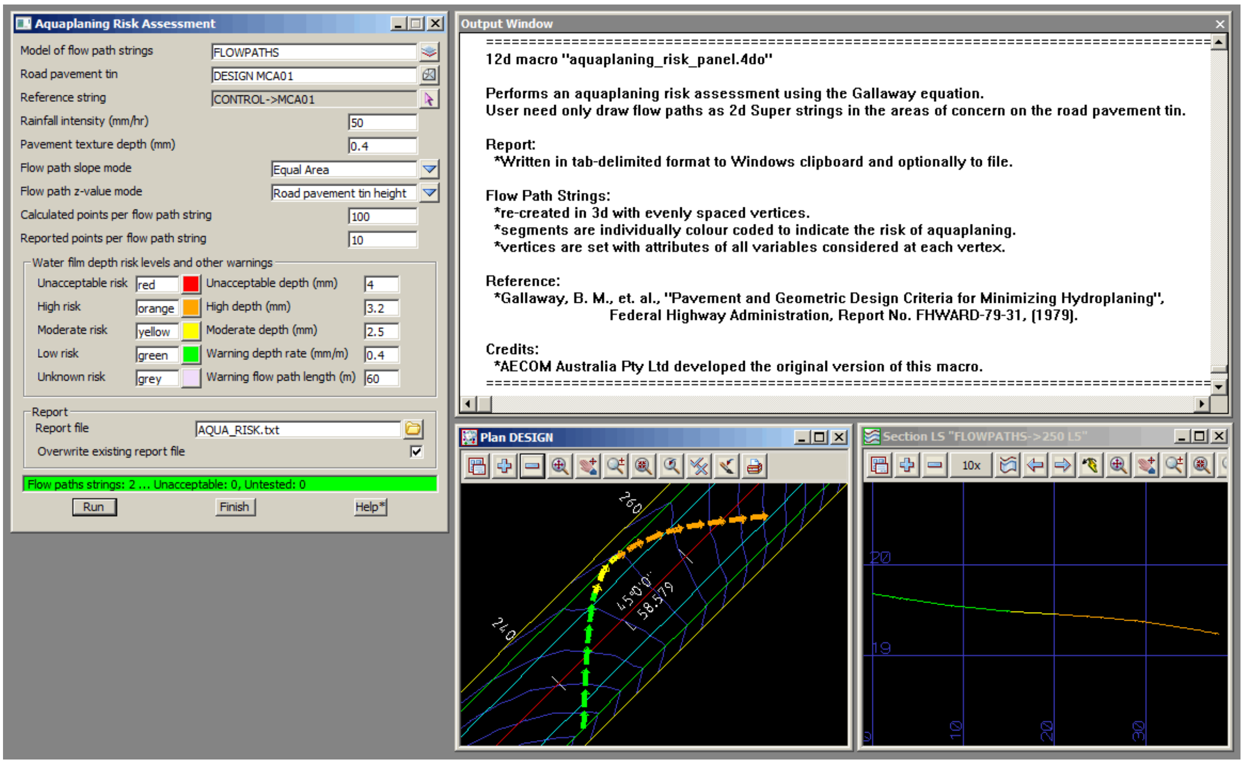
Whether to overwrite or append to an existing report file.

Run button

Runs the option.

Additional Notes:

1) The **Aquaplaning Risk Assessment** option writes auxiliary help information to the *Output Window* every time it is run, *viz*:



2) The tab-delimited report is always written to the *Windows* clipboard, even if a report file is not specified. A pre-formatted *Excel* report template is installed and available in the library – **$LIB\aquaplaning_report_template.xlsx** – and after running the aquaplaning option, the contents of the clipboard may be pasted directly into a copy of this template file, as shown here:

12D AQUAPLANING RISK ASSESSMENT

12D MODEL VERSION 10.0 Beta 9

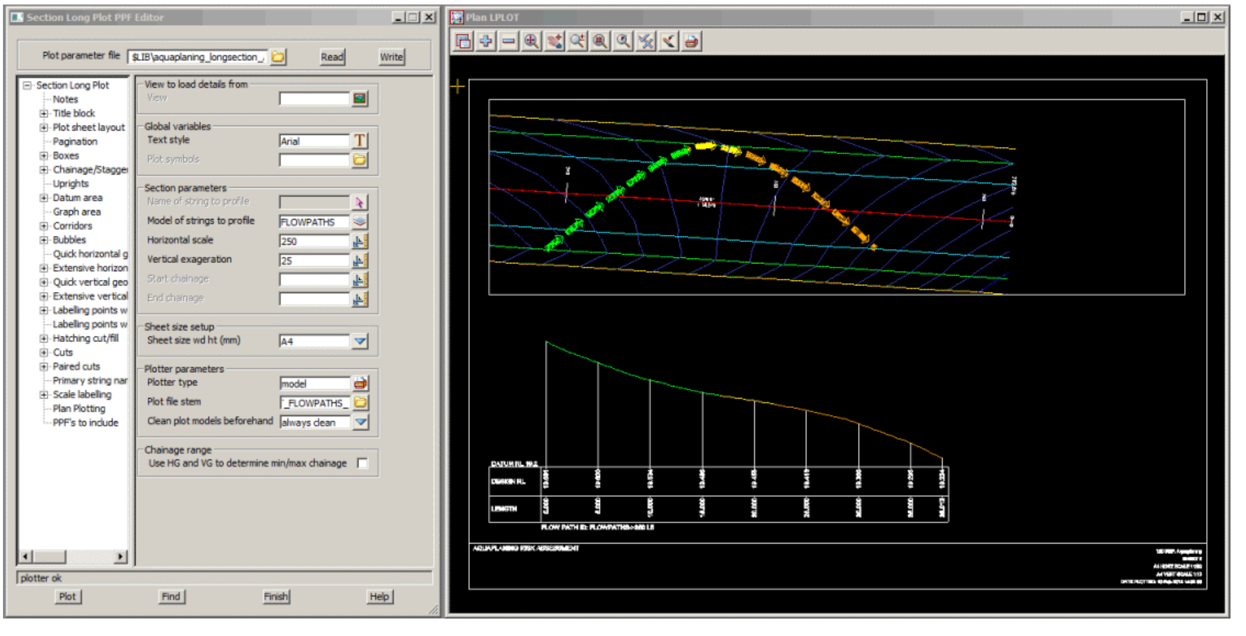
12D PROJECT:	AQUAPLANING
12D WORKING FOLDER:	C:\12d\10.00\Jobs\AQUAPLANING
DATE:	29-MAR-2012 16:57:38
ASSESSOR:	OWEN THORNTON
VERIFIER:	
MODEL OF FLOW PATHS:	FLOWPATHS
ROAD PAVEMENT TIN:	DESIGN MCA01
REFERENCE STRING:	CONTROL->MCA01
RAINFALL INTENSITY (mm/hr):	50.0
PAVEMENT TEXTURE DEPTH (mm):	0.4
FLOW PATH SLOPE:	EQUAL-AREA SLOPE

WATER FILM DEPTH RISKS		
UNACCEPTABLE RISK >=	4.0	mm
HIGH (ACCEPTABLE) RISK >=	3.2	mm
MODERATE (ACCEPTABLE) RISK >=	2.5	mm
LOW (DESIRABLE) RISK: <	2.5	mm

WATER FILM DEPTH PREDICTION

FLOW PATH ID	POINT	LENGTH (m)	DESIGN RL (m)	SLOPE (%)	DEPTH (mm)	DEPTH RISK	DEPTH RATE (mm/m)	WARNING
250 L5	1	3.801	19.617	1.59	0.97	LOW	0.26	
250 L5	2	7.602	19.562	1.49	1.49	LOW	0.20	
250 L5	3	11.404	19.519	1.28	2.01	LOW	0.18	
250 L5	4	15.205	19.485	1.11	2.48	LOW	0.16	
250 L5	5	19.006	19.460	0.94	3.00	MODERATE	0.16	
250 L5	6	22.807	19.434	0.87	3.42	HIGH	0.15	
250 L5	7	26.608	19.404	0.85	3.71	HIGH	0.14	
250 L5	8	30.410	19.360	0.92	3.81	HIGH	0.13	
250 L5	9	34.211	19.305	1.03	3.82	HIGH	0.11	
250 L5	10	38.012	19.234	1.19	3.76	HIGH	0.10	

3) Combined plan and profile plots of the analysed flow path strings, for inclusion in reports, may be generated from the **Section Long Plot PPF Editor**. A PPF file set specifically for this purpose is installed and available in the library – **$LIB\aquaplaning_longsection_A4.lplotppf** – as shown here:

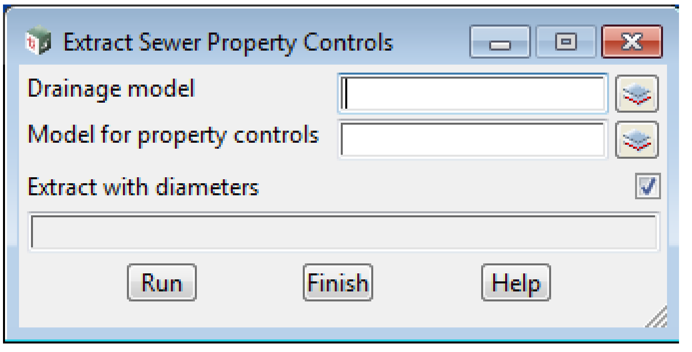


29.9.4 Extract Sewer Controls

Position of option on menu: **Design=>Drainage-Sewer=>Extract sewer controls**

The sewer property control strings are a sub string of the drainage string and therefore may only be profiled using a right mouse click of the profile button. To include these control strings on plots or export to other packages they need to be converted to super strings.

On selecting the **Extract Sewer Controls** option, the **Extract Sewer Property Controls** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Drainage model	model box		available models

All drainage strings in this model will have their property controls strings duplicated as super strings.

Field Description	Type	Defaults	Pop-Up
Model for property controls	model box		available models

The super strings for the property controls are placed in this model.

Field Description	Type	Defaults	Pop-Up
Extract with diameters	tick box	ticked	

The super strings will have the constant pipe dimension set to the control diameter.

Field Description	Type	Defaults	Pop-Up
Run	button		

Create the super strings representing the property controls.

29.9.5 Convert to Points and Lines

Position of option on menu: **Design=>Drainage-Sewer=>Convert to pts and lines**

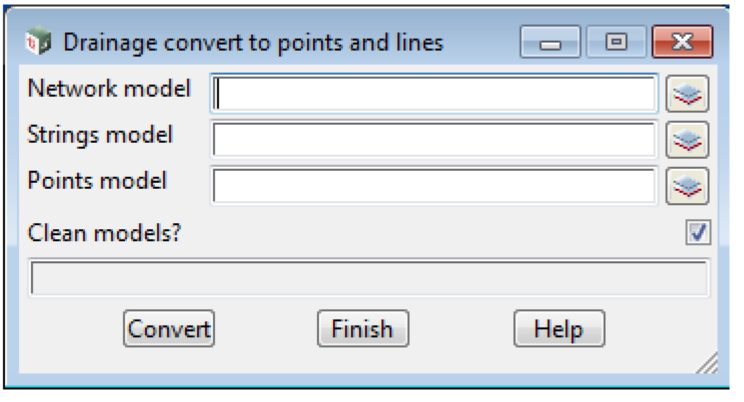
This option converts a drainage network into simple points and lines.

Pits will be created as single point super strings with the pit attributes set as string attributes.

Pipes will be created as single or multiple segment super strings with pipe attributes set as string attributes.

This may be useful for uploading to external GIS systems via the GIS module.

Selecting **Convert to pts and lines** brings up the **Drainage convert to points and lines** panel.



The fields and buttons used in this panel have the following functions.

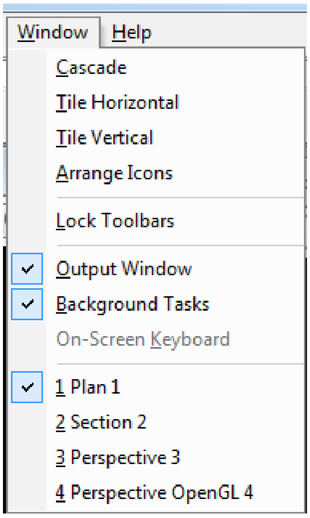
Field Description	Type	Defaults	Pop-Up
Network model	model box		available models
the source model of the drainage network			
Strings model	model box		available models
the model to output pipe strings to			
Points model	model box		available models
the model output pit point strings to			
Copy string attributes to pits and pipes?	tick box	not ticked	
if ticked, the string attributes for points are copied to to pit attributes and string attributes for lines are copied to the pipe attributes. *If not ticked, no attributes are copied.*			
Clean models	tick box	ticked	
whether or not to clean the models first			
Convert	button		
performs the conversion			

30 Window

The 12d Model **Window** menu contains options to cascade, tile horizontally and vertically. The ***Output window*** can also be toggled on and off.

The **Window** on main menu and walk-right menus are

on Main menu



cascade the non-minimized views
horizontally tile the non-minimized views
vertically tile the non-minimized views
place all minimized view icons at the bottom of the views area

lock toolbars so that they can't be moved

toggle the output window on/off
toggle the background tasks windo on/off
shows if on-screen keyboard is enabled

list of existing views

31 Help

Position of menu: **It is on the main menu as Help**

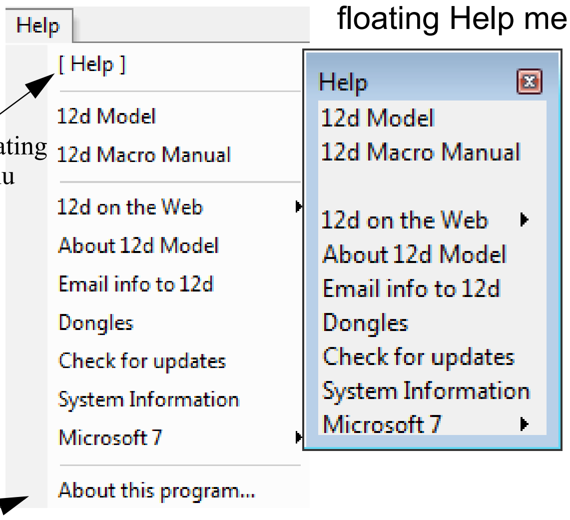
The ***12d Model*** **help** menu contains options to access Microsoft's Help on Microsoft's Help, the ***12d Model Help*** and the ***12d Model Macro Programming Language Help***, plus links to the 12d Solutions web site www.12d.com., information about the option in the current ***12d Model*** being run, an email information to 12d Solutions, dongle testing routines, checking for updates to the ***12d Model*** exe, system information and for Windows 7, the download for using WinHlp.

The **help** on main menu and walk-right menus are

on Main menu

on 12d Model menu and floating Help menu

create floating **Help** menu



contents of ***12d Model*** help
help on 12d macro programming language

links to www.12d.com.
dongle number, authed ***12d Model*** modules
email ***12d Model*** information to 12d
dongle testing panel
check for updates to the ***12d Model*** exe
information on your computer system
For WIn 7,8 - links to the WinHlp32.exe

show ***12d Model*** version and build number

See:

- 31.1 12d Model Help
- 31.2 12d Model Macro Manual
- 31.3 12d on the Web
- 31.4 About 12d Model
- 31.5 Email Info to 12d
- 31.6 Dongles Administration
- 31.7 Check for Updates
- 31.8 Microsoft 7

Continue to the next section 31.1 12d Model Help.

31.1 12d Model Help

Position of option on menu: **Help =>12d Model**

Using the *context sensitive* ***12d Model*** *Help* is documented in the chapter 6 12d Model Help.

Continue to the next section 31.1 12d Model Help or return to 31 Help.

31.2 12d Model Macro Manual

Position of option on menu: **Help =>12d Macro manual**

12d Model includes a powerful Programming Language (12dPL) for writing user programs (also known as macros).

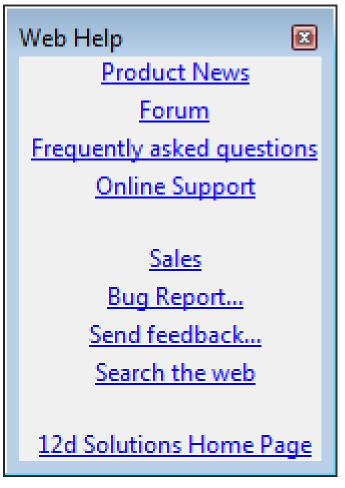
This is the context sensitive version of the ***12d Model*** Programming Language manual.

Continue to the next section 31.3 12d on the Web or return to 31 Help.

31.3 12d on the Web

Position of menu: **Help =>12d on the web**

The **12d on the web** walk-right menu contains links to the 12d Solutions web site www.12d.com.



link to ***12d Model*** forum
link to FAQs in the 12d Forum
support contacts

sales contacts
on-line bugs and enhancement form
send email to 12d Solutions
link to *Google*

12d Solutions home page

Continue to the next section 31.4 About 12d Model or return to 31 Help.

31.4 About 12d Model

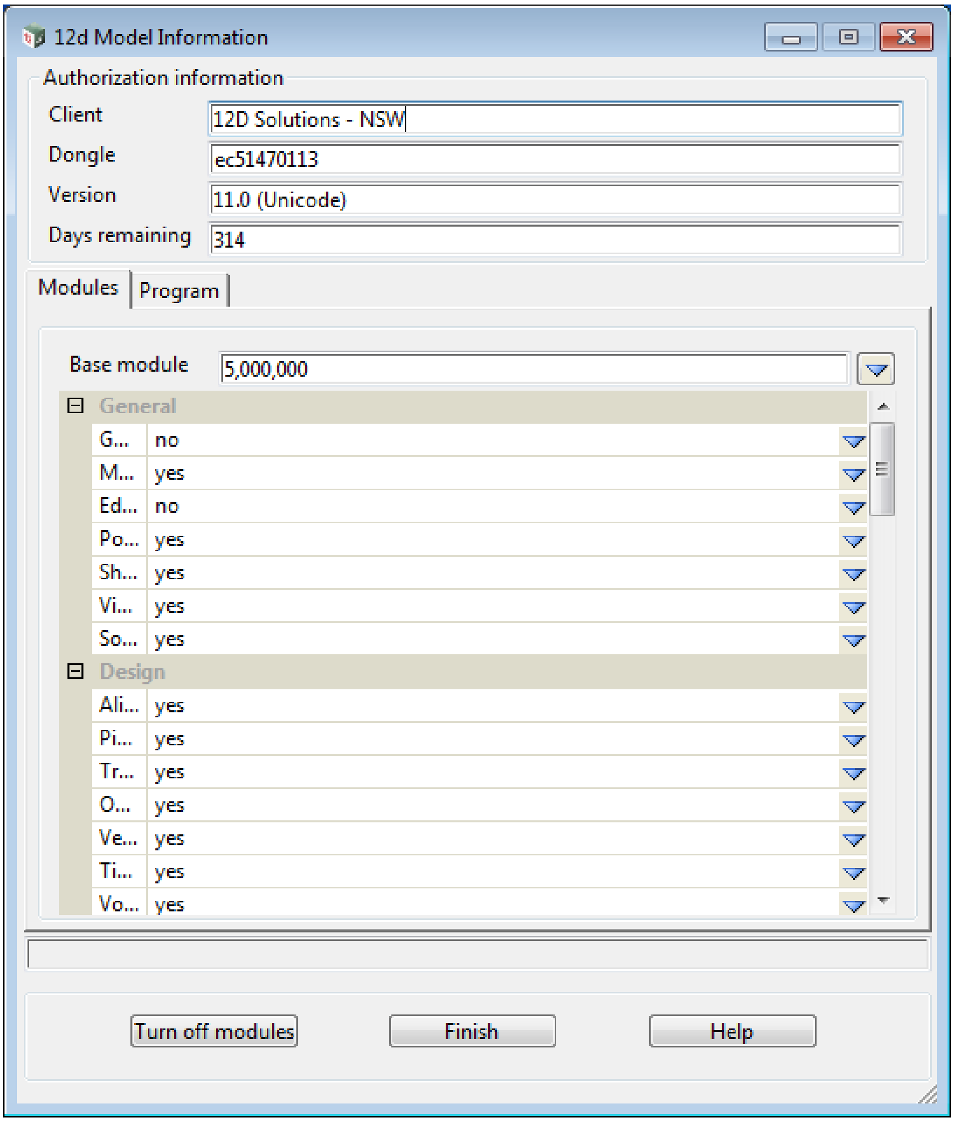
Position of option on menu: **Help =>About 12d Model**

The **About 12d Model** option displays information about the current ***12d Model*** authorization including such items as the Client name, dongle number and authorized modules.

The existing ***12d Model*** modules are shown on the **Modules** and **Extras** tabs with a tick indicating that the current licence has the module.

The **Program** tab has extra information about the current***12d Model*** executable and project.

On selecting the **About 12d Model** option, the **12d Model Information** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Client	output only		

name of the authorized client.

Dongle output only

number of the dongle for this licence.

Version output only

***12d Model** version number.*

Days remaining output only

number of days left for the authorisation.

Modules tab

Base module output only

number of allowed points in the Base module. If this number is exceeded then the project cannot be saved until enough points are deleted to bring the point count below the allowed number of points.

Alignment, Pipeline, Volumes, *etc.* choice boxes yes, no, maintenance.

*if **yes**, the module is authorised. If it is **yes** then it can be changed to **no** and when the* **Turn off modules** *button is pressed, the module will no longer be authorised until you exit the project and restart it.*

*If **no**, the module is not authorised.*

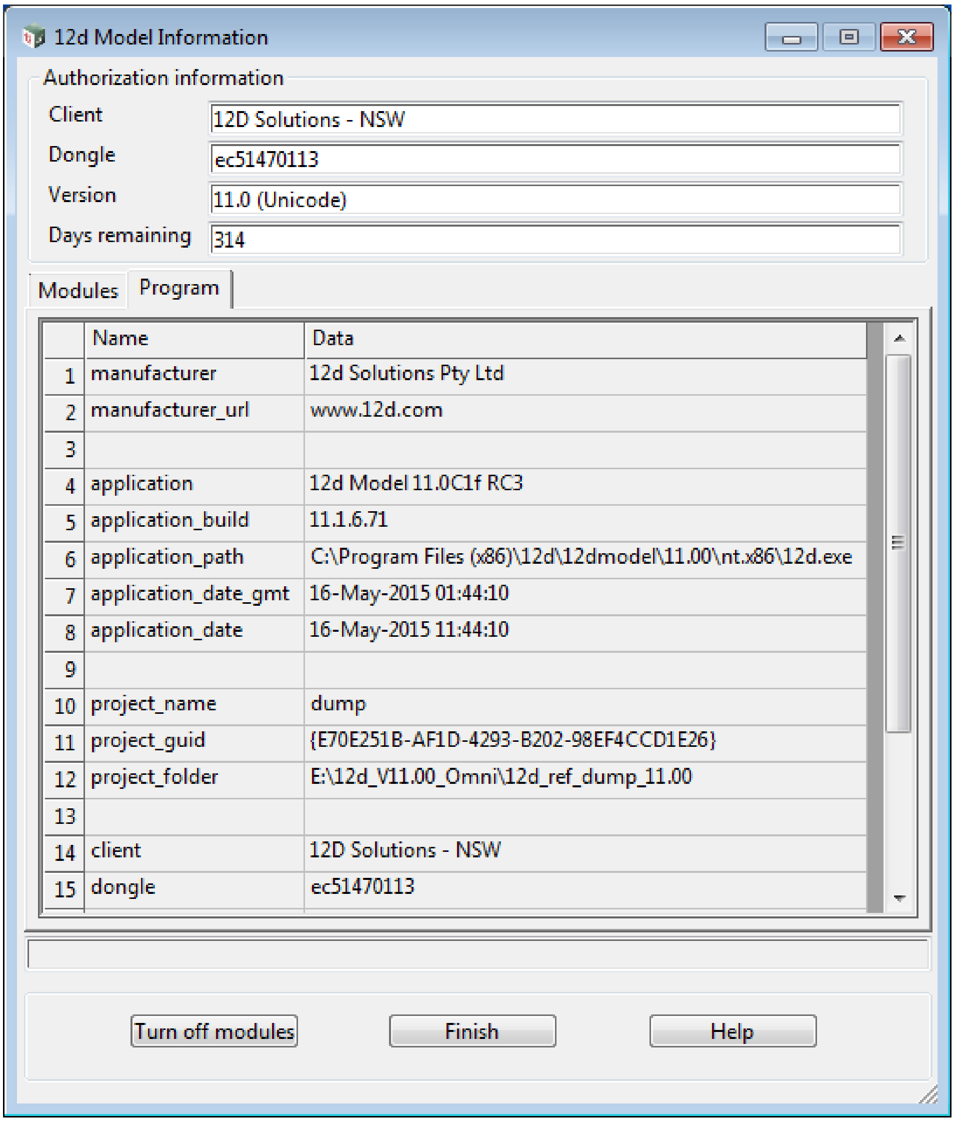
*If **Maintenance**, the module is included whilst you are on Maintenance.*

Turn off modules **button**

*for any modules with **yes**, you can **turn off** modules by setting the choice to **no** and then clicking this button.*

You can NOT turn modules back on. To get the module back you need to restart the project and then the modules that are authorised in the nodes.4d file will again have yes displayed, and hence be turned on.

Program tab



Name, Data grid

information about the 12d Model program and current project.

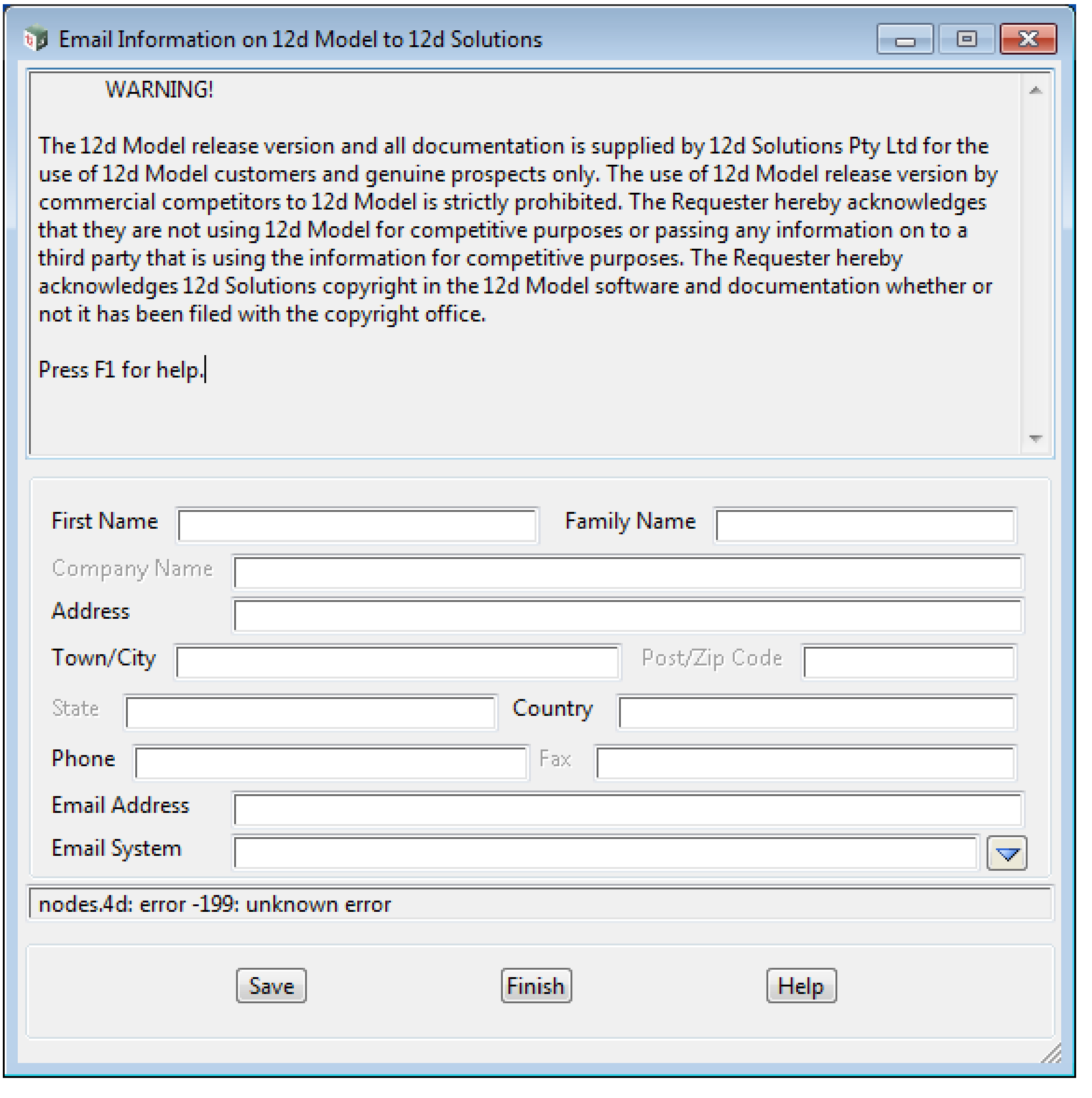
Continue to the next section 31.5 Email Info to 12d or return to 31 Help.

31.5 Email Info to 12d

Position of option on menu: **Help =>Email info to 12d**

The **Email info to 12d** option emails information about the current version of ***12d Model*** to 12d Solutions Pty Ltd. This is usually used for debugging authorization problems.

On selecting the **Email info to 12d** option, the ***12d Model*** **Information** panel is displayed.



The fields and buttons used in this panel have the following functions.

Field Description Type Defaults Pop-Up

First name/Last name/Company name ....

information about the user and the Company owning the license of ***12d Model****.*

Email button

try to send an email containing information about ***12d Model*** *and the user to 12d Solutions Pty Ltd.*

Save button

if the **Email** *button does not send an email, the* **Save** *button can be used to write the information out to a file called* ***12d_auth.txt****. This file can then be emailed to* ***support@12d.com****.*

Continue to the next section 31.6 Dongles Administration or return to 31 Help.

31.6 Dongles Administration

Position of option on menu: **Project =>Management =>Dongles =>Administration**
Position of option on menu: **Help =>Dongles**

This option is documented in 7.6.5.2 Dongles Administration.

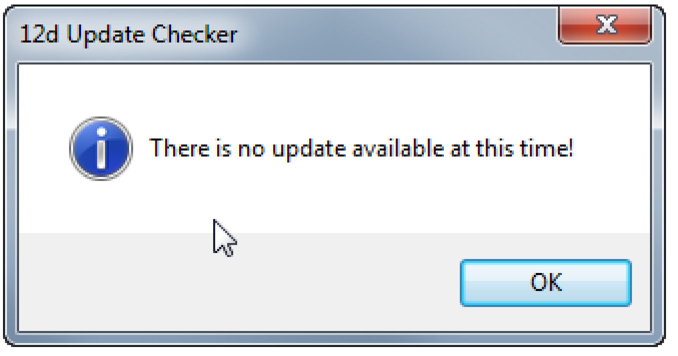
Continue to the next section 31.7 Check for Updates or return to 31 Help.

31.7 Check for Updates

Position of option on menu: **Help =>Check for updates**

Selecting **Check for updates** will check if there are any newer versions and/or builds of ***12d Model*** than the one currently being run.

The **12d Update Checker** panel will give a link to the new versions or display the message that there is no update available if the current version and build of ***12d Model*** is the latest one.



Continue to the next section 31.8 Microsoft 7 or return to 31 Help.

31.8 Microsoft 7

Position of option on menu: **Help =>Windows 7**

12d Model has a context sensitive reference help system which can be accessed directly from most menus and panels:

(a) For most menus in ***12d Model***, when the menu is on the screen and has the Windows focus, simply press F1 and the ***12d Help*** will open automatically at that menu

(b) For most panels in ***12d Model***, when the panel is on the screen and has the Windows focus, simply press F1 and the ***12d Help*** will open automatically at that panel

(c) Most panels in ***12d Model*** have a **Help** button on them and clicking on the **Help** button opens the ***12d Help*** at that panel

(d) By clicking on the option ***Help =>12d Model***

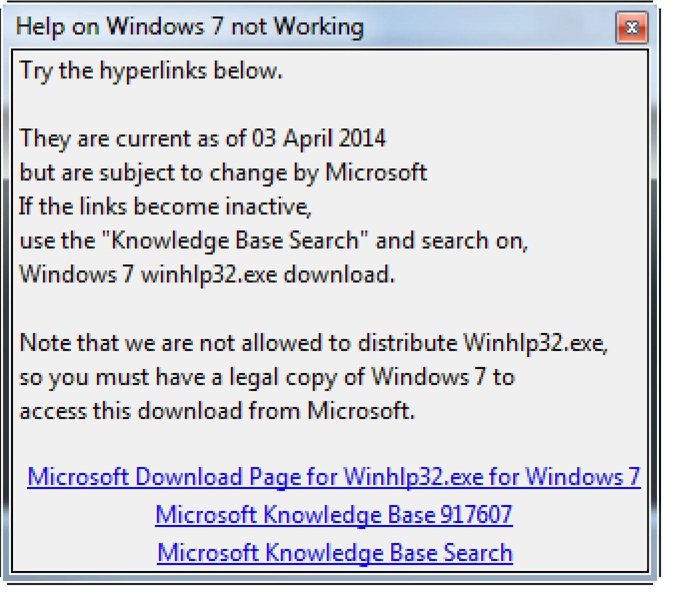
The ***12d Model*** context sensitive help uses Microsoft's ***WinHlp***.

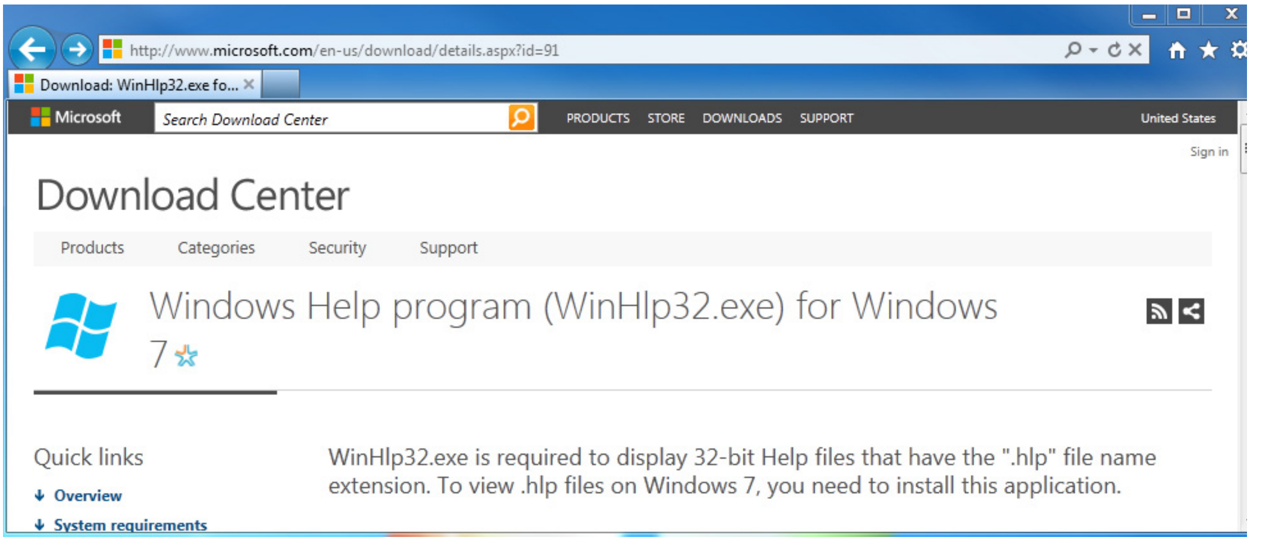
For *Win 7 and 8*, Microsoft no longer ships the WinHlp.exe executable and Microsoft will no longer allow it to be installed by the ***12d Model*** installation DVD.

So if you are running Win 7 or Win 8, you need to download and install WinHlp.exe from Microsoft's website to access the ***12d Model*** *Help* files.

The link to the area for the download is

Walking right on **Microsoft 7** displays the **Help on Windows 7 Not Working** panel which has a link to the Microsoft web site to download Winhel32.exe.





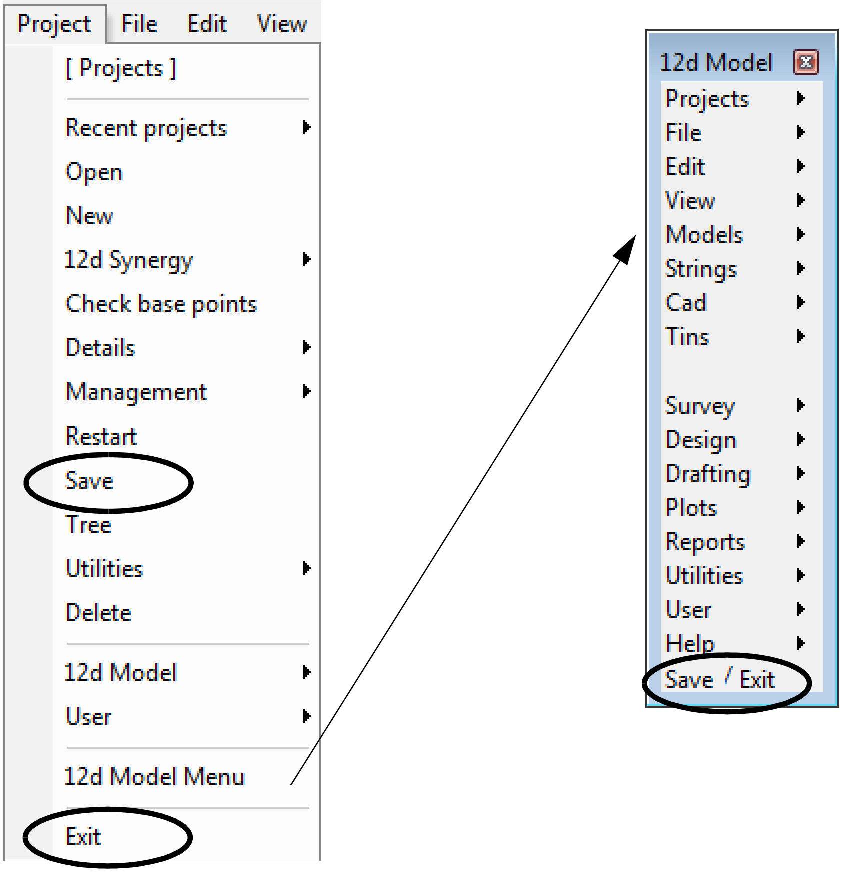
To return to the start of the documentation on the **Help** menu, click on 31 Help.

32 Save and Exit

Save and **Exit** are available from both the **Main Menu** under **Project** or on the floating ***12d Model*** menu.

on Main menu

floating **12d Model** menu

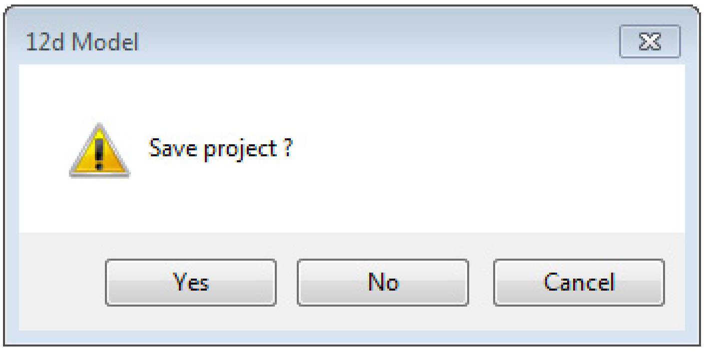


32.1 Save and Exit from Main Menu

Selecting **Save** from the **Main Menu=>Projects** menu simply saves all data in the project modified since the last save.

When **Exit** is selected from the bottom of the **Main Menu=>Projects** menu and a save is not needed, 12d Model exits the project.

If **Exit** is selected from the bottom of the **Main Menu=>Projects** menu and a save is needed, then the **Save Project ?** yes-no panel is placed on the screen.



If **Yes** is selected, then 12d Model saves the project and exits.

If **No** is selected, then 12d Model does not save the project and exits.

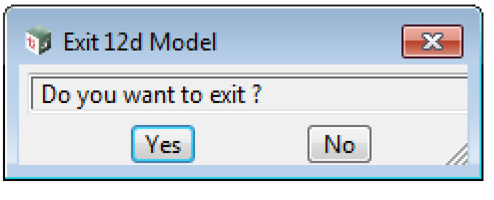
If **Cancel** is selected, then the Exit is aborted and 12d Model stays in the project.

Note: when exiting 12d Model, the user is reminded if any ***mtf*** files modified by the 12d Model *mtf editor* have not been saved, or any string editors are still running.

32.2 Save/Exit from Floating12d Model Menu

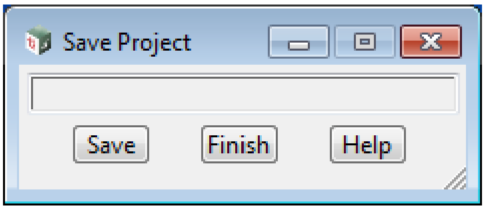
When **Save** is selected from the **left hand side** of the bottom of the ***12d Model*** menu (**Save / Exit** menu option), all the information modified since the last save is written to disk.

When **Exit** is selected from the **right hand side** of the bottom of the 12d Model menu (**Save / Exit** menu option), an **Exit12d Model** panel is fired up.



Selecting **No** removes the **yes-no** pop-up and leaves the user in ***12d Model***.

If **Yes** is selected and a modification to the project has been made since the last project save, a **Save Project** panel is fired up.



If **yes** is selected, the project is **saved** and 12d Model terminates.

If **no** is selected, the project is **not saved** and 12d Model terminates.

If **cancel** is selected, the **exit** option is **aborted** and the user is left in 12d Model.

Note: When exiting 12d Model, the user is reminded if any ***mtf*** files modified by the 12d Model *mtf editor* have not been saved, or any string editors are still running.

Selecting **No** removes the **yes-no** pop-up and leaves the user in ***12d Model***.

If **Yes** is selected and a modification to the project has been made since the last project save, a **Save Project** panel is fired up.

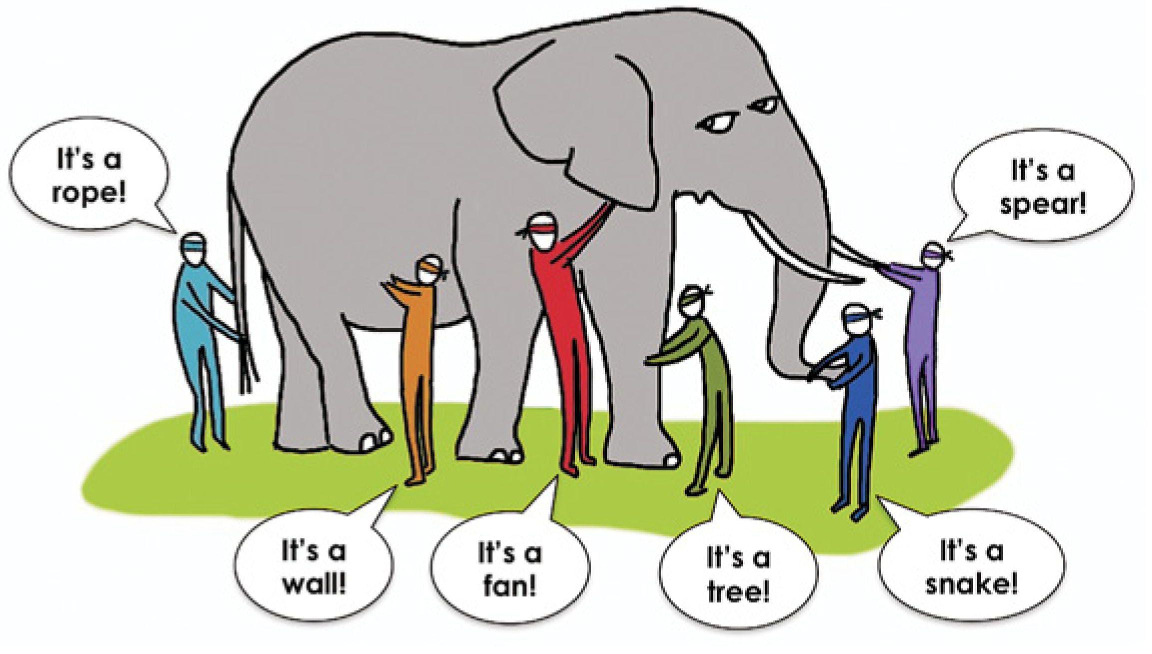
Appendices

33 BIM

See

33.1 What is BIM ?

What BIM is seems to depend on who you are talking to. And the various participants in the design, construction and management of the asset (building or infrastructure) also emphasise different aspects of BIM.



If a wide search of the literature for BIM and ISO Standards is made, and all the marketing fluff is ignored, one quickly sees that almost universally the **BI** in BIM is **Building Information** and although BIM is about the process, all the objects are for a Building on a site. The other constant is that the International Standard for transferring BIM data is the vendor independent format of IFCs. For more information on IFCs, see 8.2.19 IFC Output.

In 2011 the BIM movement received a major boost when the UK Government released the **Government Construction Strategy** which aimed to reduce he cost of government construction projects by 15-20% and BIM was identified as one of the strategies to achieve help the savings.

The **Strategy** contained an **Action Plan** that included the objective for BIM to *introduce a progressive programme of mandated use of fully collaborative Building Information Modelling for Government projects by 2016.*

One early outcome of the **Strategy** was for BuildingSMART *to develop a national BIM standard with the principles of interoperability. This will be key to future delivery of Level 3 "Open & Shared" BIM.*

buildingSMART is now the International group looking after the development of the **Industry Foundation Classes** (IFCs) which is the data model for the Open Standard for BIM. See http://www.buildingsmart-tech.org.

For more information, see 33.3 The UK Government and BIM and for some common BIM terms and the ***Levels of BIM*** outlined by the UK Government, see 33.4 Common Terms in BIM.

Under the umbrella of **buildingSMART** there are a number of International Committees currently grappling with the problem of extending IFCs from Buildings to Bridges, an Alignment for Roads, and for Roads and Railways in general, and to move IFCs from uncoordinated flat space with (0,0) in the corner, to a GIS and 12d environment with world coordinates and projections.

Beware that some groups talk BIM but then insist on data being supplied in an unpublished, encrypted proprietary format. In Australia this is often Revit, Navisworks, DWG or DGN.

This is diametrically opposed to the International Open BIM concept but apart from pointing that out to your client, you are caught between a rock and a hard place.

However for those users caught in the encrypted proprietary format trap, we are working to help solve your problems. For an overview of where we are currently at, please see the information in the next section 33.2 The 12d Approach to BIM.

Continue to 33.2 The 12d Approach to BIM or return to 33 BIM.

33.2 The 12d Approach to BIM

12d Solutions is actively involved in, and supporting, the International effort on IFCs.

12d Solutions is:

(a) a founding member of the **Open BIM Alliance** which supports the transferring of BIM data using the ISO Standard for IFCs.

(b) working with **buildingSMART** (the International Group looking after IFCs) on how non-Building data should be transferred using IFCs. See http://www.buildingsmart-tech.org.

(c) working with **buildingSMART** on how IFCs can be extended to Alignments, Roads and Railways.

(d) working on the **Precinct Information Model** project, part of the **CRC** for **Low Carbon Living**.

This project is looking at the problem of extending BIM from a few building on a site, to a larger area and including roads, drainage, sewerage etc.

For more information on ***12d Model*** and IFCs, see 8.2.19 IFC Output

For those users caught in the encrypted proprietary format trap, although we do not have access to the formats, **12d Solutions** will continue to look for ways to make that data accessible to our users.

To this end, the approaches we are pursuing with our users to write out BIM type data from ***12d Model*** are:

(a) Writing Out Fully Structured IFCs

Using IFCs correctly means that the data to be output must have a fully defined hierarchical structure (known as the Spatial Structure).

This is the approach of the ***12d Model* IFC Writer**. See 8.2.19 IFC Output.

(b) Writing Out Unstructured IFCs

The IFC format could also be used in an informal way as a method of transferring data where there is no concern for the spatial structure but just after the object shapes.

12d is currently working on an alternate IFC Writer to do this.

(c) Writing out OBJ and STL Files

Drainage and sewer strings, super strings with round or rectangular sections, extrusions and trimeshes can be written out as OBJ files (see 8.2.17 Export OBJ) or STL files (8.2.18 Export STL).

(d) Writing out Polyface Mesh in DWG

Drainage and sewer strings, super strings with round or rectangular sections, extrusions and trimeshes can be written out to DWG files as Polyface Meshes. See 8.2.7 Output DWG/DFX/DXB Files.

(e) Revit Routines to Read in 12d Model Data.

EXDS have developed routines that run from within Revit to read in ***12d Model*** data in 12d.XML format See www.EXDS.com.au.

The approaches we are using to bring data into ***12d Model*** are:

(a) Reading in OBJs

OBJ files can be read into to ***12d Model*** as trimeshes. See 8.1.18 Wavefront OBJ Input

(b) Reading in Polyface Mesh in DWG

Polyface mesh in a DWG file can be read into to ***12d Model*** as trimeshes. See 8.1.10 DWG/ DXF Input.

(c) Getting Revit Data into 12d Model

Although the Revit database itself is unreadable, Revit can write data out in the Autodesk fbx format and there is an option in ***12d Model*** to read in an fbx file trimeshes. See 8.1.11 FBX Input.

Also if data can be written out from Revit as Polyface Meshes to a DWG file then again that can be read in into ***12d Model*** as trimeshes. See 8.1.10 DWG/DXF Input.

Continue to 33.3 The UK Government and BIM or return to 33 BIM.

33.3 The UK Government and BIM

In 2011, the UK Government released the **Government Construction Strategy** which was the framework for a range of work streams, all of which have the ultimate aim of reducing the cost of government construction projects by 15-20% (https://www.gov.uk/government/publications/government-construction-strategy).

One problems was identified in Section 2.30:

2.30 *A lack of compatible systems, standards and protocols, and the differing requirements of clients and lead designers, have inhibited widespread adoption of a technology which has the capacity to ensure that all team members are working from the same data, and that:*

- *the implications of alternative design proposals can be evaluated with comparative ease;*
- *projects are modelled in three dimensions (eliminating coordination errors and subsequent expensive change);*
- *design data can be fed direct to machine tools, creating a link between design and manufacture and eliminating unnecessary intermediaries; and*
- *there is a proper basis for asset management subsequent to construction.*

and the remedy, BIM, was mandated in Section 2.31:

2.31 *The Cabinet Office will coordinate Government's drive to the development of standards enabling all members of the supply chain to work collaboratively through Building Information Modelling (BIM). This will be a phased process working closely with industry groups, in order to allow time for industry to prepare for the development of new standards and for training*

The **Strategy** also included an **Action Plan** with a timetable, and Section 7 was for Building Information Modelling (BIM):

*7(i)**Objective**: To introduce a progressive programme of mandated use of fully collaborative Building Information Modelling for Government projects by 2016.*

From the **Government Construction Strategy: One Year On and Action Plan Update** it reported for BIM:

A reciprocal Memorandum of Agreement has been reached with buildingSMART US to develop a national BIM standard with the principles of interoperability. This will be key to future delivery of Level 3 "Open & Shared" BIM.

Continue to 33.4 Common Terms in BIM or return to 33 BIM.

33.4 Common Terms in BIM

Here is a list of some need-to-know BIM terms and their definitions taken from the **NBS** website http://www.thenbs.com.

NBS is part of RIBA Enterprises Ltd, which is wholly owned by the Royal Institute of British Architects (RIBA) and was contracted by the UK Government to build the National BIM Library.

1. . 4D, 5D, 6D

First there was 2D CAD, then 3D CAD – now there are extra dimensions to refer to the linking of the BIM model with time-, cost- and schedule-related information (although the precise order hasn't to date been agreed across the whole industry).

2. . Asset Information Model (AIM), Building Information Model (BIM)

Not only is there the 'Building' information model, but the 'Asset' information model – which is the name given to the same model post-construction, i.e. supplemented with the data needed to assist in the running of the completed asset. Note that 'asset' can also refer to civil engineering and infrastructure work.

3. . Common Data Environment (CDE)

This is a central information repository that can be accessed by all stakeholders in a project. Whilst all the data within the CDE can be accessed freely, ownership is still retained by the originator.

The scope and requirements for a CDE are defined in PAS 1192-2 (see 8. . PAS 1192).

4. Level 0 BIM, Level 1 BIM, Level 2 BIM, Level 3 BIM

The move to 'full' collaborative working via distinct and recognisable milestones, in the form of 'levels'. These have been defined within a range from 0 to 3, and, whilst there is some debate about the exact meaning of each level, the broad concept is:

Level 0 – no collaboration.

2D CAD drafting only. Output and distribution is via paper or electronic prints, or a mixture of both.

Level 1 – a mixture of 3D CAD for concept work, and 2D for drafting of statutory approval documentation and Production Information.

CAD standards are managed to BS 1192:2007, and electronic sharing of data is carried out from a common data environment (CDE), often managed by the contractor.

There is no collaboration between different disciplines – each publishes and maintains its own data.

Level 2 – collaborative working – all parties use their own 3D CAD models.

Design information is shared through a common file format, which enables any organisation to be able to combine that data with their own in order to carry out interrogative checks on it.

Hence any CAD software that each party used must be capable of exporting to a common file format.

This is the method of working that has been set as a minimum target by the UK government for all public-sector work, by 2016.

Level 3 – integrated working between all disciplines by using a single, shared project model which is held in a common data environment.

All parties can access and modify that same model, removing the final layer of risk for conflicting information. This is known as ‘Open BIM’ (see 7. Open BIM), and the UK government’s target date for public-sector working is 2018, although the precise requirements have yet to be determined.

Note that the definition of BIM maturity Level 2 was originally developed as part of the UK Government strategy in 2011. It is also defined in PAS 1192-2, with reference to best practice and the adoption tools and standards.

It is also worth noting, though, PAS 1192-2 acknowledges that, given the early stages of adoption of managed methods of working in BIM at the time the PAS was drafted, it can be expected that Level 2 practices will continue to evolve, and that the scope of information sharing and exchange will vary from project to project.

Therefore, PAS 1192-2 anticipates that the definition of Level 2 BIM will continue to evolve around the core principles of the shared use of individually authored models in a CDE.

5. Construction Operations Building Information Exchange (COBie)

COBie is a data schema which is delivered in a spreadsheet data format, and contains a ‘subset’ of the information in the building model (all except graphical data, and hence a subset of IFC (see 6. Industry Foundation Class (IFC)), for FM handover. It was originally devised by the US Army Engineering Corps.

Over the course of a project, data can be added to it from a range of sources (besides CAD programs), relating to brief, design, construction, operation, refurbishment or demolition, as the case may be.

The UK Government’s Level 2 - mandated requirement is for COBie-compliant information exchange. BS 1192-4 documents best practice for the implementation of COBie.

6. Industry Foundation Class (IFC)

IFC is an object-based format, to enable exchange of information between different software. Developed by **buildingSMART**, a global alliance specialising in open standards for BIM, IFC is an official standard, BS ISO 16739, and contains geometric as well as other data.

7. Open BIM

An open-source approach to collaborative design, realisation and operation of buildings, based on open standards and work flows.

Open BIM is an initiative of several leading software vendors using the buildingSMART Data Model, which incorporates data to ISO 16739 (via the IFC file format), terms to ISO 12006-3 (using the International Framework for Dictionaries, which maps different technical terms that have the same meaning) and process to ISO 29481-1 (the Information Delivery Manual).

8. . PAS 1192

The PAS 1192 framework sets out the requirements for the level of model detail (the graphical content), model information (non-graphical content, such as specification data), model definition (its meaning) and model information exchanges:

(a) PAS 1192-2 deals with the construction (CAPEX) phase, and specifies the requirements for Level 2 maturity; sets out the framework, roles & responsibilities for collaborative BIM working; builds on the existing standard of BS 1192, and expands the scope of the Common Data Environment (see 3. . Common Data Environment (CDE)).

(b) PAS 1192-3 deals with the operational (OPEX) phase, focussing on use & maintenance of the Asset Information Model, for Facilities Management.

(c) BS 1192-4 documents best practice for the implementation of COBie.

(d) PAS 1192-5 is currently under development, and will cover security of data.

Continue to 33.5 What is Wrong with Local Coordinates ? or return to 33 BIM.

33.5 What is Wrong with Local Coordinates ?

What we mean by Local Coordinates is the placing of (0,0) somewhere on the site, with a direction for the x or y axis, NOT a cartographic projection with (0,0) defined on the site.

There is nothing wrong with using a Local Coordinate system as long as you are aware of the restrictions and consequences. As Einstein said for General Relativity "space is locally flat".

So as long as you are working on a small site (say under 5 kilometres long) and everyone is able to reliably measure back to your (0,0) and use your definition of angle of the coordinate axes, then there should be no major problems.

But if you are working on a large project, or trying to combine data from a variety of sources, or want to coordinate work with a GPS, then you need to know more about how and why there are problems with Local Coordinates.

One of the problems is that the earth is not flat.

See

33.5.1 An Exact Position for Everything on the Earth
33.5.2 Map (Cartographic) Projections - Eastings and Northings
33.5.3 Why Isn’t MGA Used on a Small Building Site ?

33.5.1 An Exact Position for Everything on the Earth

To give everything on earth a unique coordinates that fit in with GNSS systems (GPS, GLONASS, BeiDou and Galileo), a **reference ellipsoid** has been defined with:

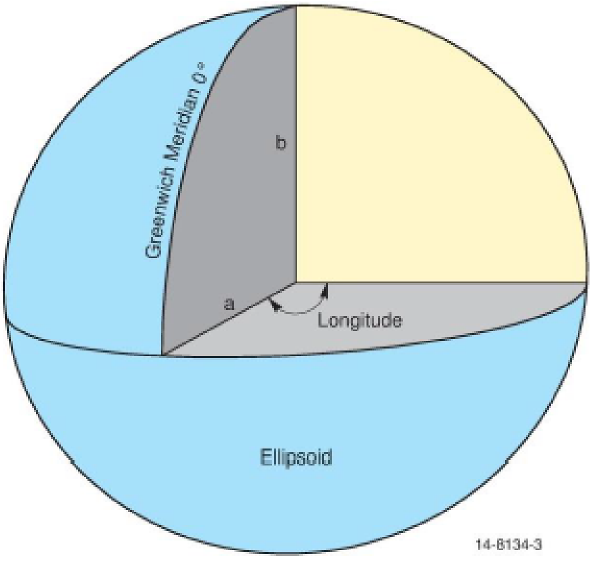
(a) (0,0,0) at the Mass centre of the earth (geocentre), a semi major axis of 6,378,137 m, and an inverse flattening of 298.257222101. This is known as **GRS80**.

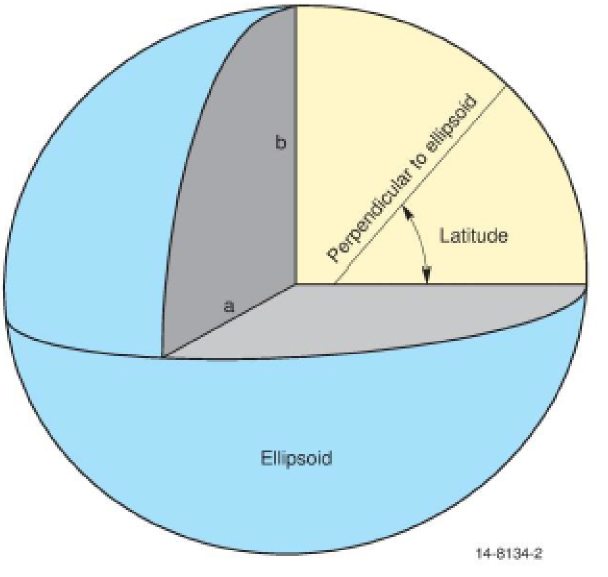
This is because satellites orbit around the mass centre of the earth.

(b) Longitude is an angular quantity measured from the Greenwich meridian.

(c) Latitude is an angular quantity measured from the equatorial plane, to the plane defined by the point position and the plumb line to the ellipsoid surface.

(d) Ellipsoid height is the height above the reference ellipsoid.





Using a GNSS system, the Geodetic coordinates of (Latitude, Longitude, Ellipsoid height) can be directly obtained for any position on earth that can see the satellites.

Note: Heights in Australia are not usually quoted as Ellipsoid heights h but instead in AHD (Australian Height Datum) H which is defined in terms of Means Sea Level. The difference between the ellipsoid height and the AHD is known as the **N (AHD Value**.

AHD (Geoid height) = Ellipsoid height - N (AHD) Value

Continue to 33.5.2 Map (Cartographic) Projections - Eastings and Northings or return to 33.5 What is Wrong with Local Coordinates ? or 33 BIM.

33.5.2 Map (Cartographic) Projections - Eastings and Northings

Although Geodetic coordinate of Latitude and Longitude give unique coordinates for a point, they are not very appropriate for drawing up civil engineering projects. They are usually represented on a map by mathematically "projecting" them on to a surface, which can be laid flat.

A Map or Cartographic Projection is a transformation of the latitudes and longitudes of locations on the surface of an ellipsoid into locations on a plane. The projection coordinates are usually referred to as Eastings (x) and Northings (y).

The important thing is that the Cartographic projection can be reversed. That is, has an inverse. So the Cartographic projection

(a) maps a (Latitude, Longitude) to a (Easting, Northing)

and

(b) the inverse projection maps an (Easting, Northing) to a (Latitude, Longitude).

Consequently if you have the (Easting, Northing) then can calculate the equivalent (Latitude, Longitude) and vice versa.

The most commonly used Cartographic projection for civil works is the **Transverse Mercator** projection.

In mathematical speak, **Transverse Mercator** projections are **conformal** projections and preserve angles locally, implying that they map infinitesimal circles of constant size anywhere on the Earth to infinitesimal circles of **varying sizes** on the map. In contrast, mappings that are not conformal distort most such small circles into ellipses of distortion. An important consequence of conformality is that relative angles at each point of the map are correct, and the local scale (although varying throughout the map) **in every direction around any one point is constant**.

Transverse Mercator projections are commonly used for road projects by the Road Authorities in each State of Australia. They are also used for the Meridional Circuits in New Zealand.

The Transverse Mercator projection is also used as the basis of the **UTM** (Universal Transverse Mercator). The UTM is not a single map projection but divides the Earth into sixty zones, each a six-degree band of longitude, and uses a Transverse Mercator projection.

In Australia, **MGA** coordinates (Map Grid of Australia) are based on the UTM using the ellipsoid GRS80 previously defined.

So an MGA coordinate (Easting, Northing) has a unique (Latitude, Longitude).

And a reading from a GNSS device (commonly known as GPS) uniquely converts to a MGA coordinate, and conversely a MGA coordinate converts to (Latitude, Longitude) for use in a GPS without knowing anything about a local coordinate system.

Continue to 33.5.3 Why Isn’t MGA Used on a Small Building Site ? or return to 33.5 What is Wrong with Local Coordinates ? or 33 BIM.

33.5.3 Why Isn't MGA Used on a Small Building Site ?

It is very easy to gloss over one very important point about Transverse Mercator projections as used in MGA.

Transverse Mercator projections are **conformal projections** and although they **preserve angles**, they do **not preserve distances**.

So a small circle will transform to a small circle but the radius will vary depending on where you are in an MGA zone.

The amount of the distortion is known as a **scale factor** and it varies from point to point. Over a small distance the scale factor is almost a constant but that constant value **is rarely 1**.

What that means in practise is if you draw everything up in MGA coordinates, the actual distance between two points in **NOT the standard square root distance between the two points**.

This means that on a plan in MGA coordinates, you can't simply use a ruler to measure he distance between two points. You need to allow for the scale factor. Over a small distance the scale factor will be a constant (for example 0.9996) but over a large distance you need to need to do the correct Geodetic calculations.

This also means that in a standard drafting system, the usual distance between two points that is used in conventional drafting dimensioning options, will not give the correct value.

For those working in a local coordinate project you may have noticed the scale factor effect when you have identified two points a good distance apart and you also have their MGA coordinates. If you measure between the points in your local coordinate system and then calculate the square root distance between the two MGA points using the actual MGA coordinates then you won't get the same number.

This is probably why most projects defined in local coordinates only give the MGA coordinates for o**ne point** and a **rotation of the axes**.

The result of all this is that if you only have a small building site then you can obtain fairly accurate MGA coordinates by applying a 2D Helmert transformation to the local coordinates. That is, a translation, a rotation or the axis and a **constant multiplication factor** (which could be the Point Scale Factor of the given MGA point).

However, as the size of the project increases, the larger errors will be in the MGA coordinates for points further away from the origin.

Similarly if you apply a translation, rotating and scale to convert data given in MGA coordinates (say from a GIS system or the State Road Authority) to local coordinates, then there will be errors the further away from the "best fitted" point.

Note

If you are not a surveyor you will probably not be aware that in ***12d Model***, you can specify the Cartographic projection that the coordinates are in and can then obtain the (Latitude, Longitude) for selected points, measure between points and get the square-root distance and also the **ellipsoid distance** between the points.

Return to 33.5 What is Wrong with Local Coordinates ? or 33 BIM.

34 12d Archive File Format

The ***12d Archive*** file format (called 12d ascii in Version 10 and earlier) is a text file definition from 12d Solutions which is used for reading and writing out string data from 12d Model. 12d Archive files normally end in '**.**12da' and are often referred to as 12da files.

Unlike the earlier 12d Ascii files in Version 9, from Version 10 onwards the 12d Archive file is a **Unicode** file.

This document is for the ***12d Archive file*** format used in ***12d Model*** *Version 11*.

For General Comments about 12da, see 34.1 General Comments about a 12da File

For the 12da definitions:

Attributes	34.2 Attributes
Commands	34.3 Commands
Each string type	34.4 12da Definition for each String Type
Tin	34.3.7 Tin
Super Tin	34.3.8 Super Tin

For documenation on the ***12d XML*** file format, see 35 12d XML File Format.

34.1 General Comments about a 12da File

Unicode - 12d Archive file is a Unicode file.

//

Anything written on a line after // is ignored. This is used to place comments in the file.

Blank lines

Unless they are part of a text string, blank lines are ignored.

Spaces

Unless enclosed in quotes ("), more than one consecutive space or tab is treated as one space. Except when it is the delimiter after a //, an end of line (<enter>) is also considered a space.

Spaces and special characters in text strings

Any text string that includes spaces and any characters other than a to z, A to Z or 0 to 9 (alphanumeric), must be enclosed in double quotes. In text strings, double quotes " and backslash \ must be preceded by a \. For example, \" and \\ define a " and a \ respectively in a text string.

Names of models, tins, styles, colours and attributes

Models, tins, styles (linestyles), colours and attributes can include the characters a to z, A to Z, 0 to 9 (alphanumeric characters) and space. Leading and trailing spaces are ignored. The names can be up to 255 characters in length. If the name includes spaces, the name must be enclosed in double quotes (").

The names for models, tins, styles, colours or attributes can not be blank.

The names for models, tins, styles and colours can contain upper and lower alpha characters which are stored, but the set of model names, tin names, style names, colour names or attribute names for an object *must be unique when case is ignored*. For example, the model name "Fred" will be stored as "Fred" but "FRED" is considered to be the *same* model name as "Fred".

String names

String names can include the characters a to z, A to Z, 0 to 9 (alphanumeric characters), space, decimal point (.), plus (+), minus (-), comma (,), open and closed round brackets and equals (=). Leading and trailing spaces are ignored. String names can be up to 255 characters in length. If the string name includes anything other than alphanumeric characters, then the name must be enclosed in double quotes (").

String names can contain upper and lower alpha characters which are retained but case is ignored when selecting by string name. That is, the string name **Fred** will be stored as **Fred** but **FRED** is not considered to be a different string name.

String names do not have to be unique and can be blank.

Continue to the next section 34.2 Attributes or return to 34 12d Archive File Format.

34.2 Attributes

Many *12d* ***Model*** objects (models and elements such as individual strings and tins) can have an unlimited number of named **attributes** of type integer (numbers), real and text.

The attributes for an object are given in an ***attributes block*** which consists of the keyword **attributes** followed by the definitions of the *individual attributes* enclosed in start and end curly braces { and }. That is, an ***attributes_block*** is

attributes {

attribute_1

attribute_2

...

attribute_n

}

where the attribute definitions for the individual attributes *attribute_i* consists of

attribute_type ***attribute_name*** ***attribute_value***

where

attribute_type	is integer, real or text
attribute_name	is the unique attribute name for the object.

If the attribute name includes spaces then the text of the name must be enclosed in double quote character (")

and

attribute_value	is the appropriate value of the integer, real or a text.

Within an object, the attribute names are case sensitive and must be unique. That is, for attribute names, upper and lower case alphabet characters are considered to be different characters.

If the *text* for a text attribute includes spaces then the text must be enclosed in double quote characters ("). It the text is blank, it is given as "".

An example of and ***attribute block*** defining four attributes named "pole id", "street", "pole height" and "pole wires" is:

```
attributes {
  text              "pole id"           "QMR-37"
  text               street             "477 Boundary St"
  real              "pole height"        5.25
  integer           "pole wires"         3
}
```

Continue to the next section 34.3 Commands or return to 34 12d Archive File Format.

34.3 Commands

Commands consist of a ***keyword*** followed by a space and then a ***value*** (a keyword and its value is often referred to as a ***keyword pair***). A *value* must always exist.

keyword *value* // a keyword pair

There can be more than on command keyword pair per line as long as each keyword pair is separated by a space. In fact, the *keyword* can be on one line and the *value* on the next line.

Although the names of commands are only shown in lower case in these notes, commands are case insensitive and all combinations of case are recognised as the same command. That is **model**, **MODEL** and **ModeL** are all recognised as the command **model**.

For the definition of the commands in the 12da file *see*:

34.3.1 Model
34.3.2 Colour
34.3.3 Style
34.3.4 Breakline
34.3.5 Null
34.3.6 String
34.3.7 Tin
34.3.8 Super Tin

Or return to 34 12d Archive File Format.

34.3.1 Model

There are two formats for the **model** command:

(a) model command when there are no attributes for the model

model *model_name*

All elements (strings, tins, plot frames etc) following until the next **model** keyword are placed in the model *model_name*. This can be overridden for an element by a **model** command inside the element definition.

The default model name used for elements when no model name has been specified is ***data***.

(b) model command when there are model attributes

If the model includes attributes, the following form of the *model* command must be used.

model {

name *model_name*

attributes_block

}

where the *attributes_block* is defined in 34.2 Attributes.

For example:

```
model {
  name        "telegraph poles"

  attributes {
    text          "pole id"          "QMR-37"
    text          "street"           "477 Boundary St"
    real          "pole height"       5.25
    integer       "pole wires"        3
  }
}
```

Continue to the next section 34.3.2 Colour or return to 34.3 Commands or 34 12d Archive File Format.

34.3.2 Colour

The format of the **colour** command is:

colour *colour_name*

When reading a 12da file, there is a ***current colour,*** which has the default value of ***red***, and when a **colour** command is read, the ***current colour*** is set to *colour_name.*

When strings are read in a 12da file, they are given the *current colour*.

This can be overridden for a string by a *string colour command* inside the string command defining that string. For the definition of the string command, see 34.3.6 String.

Continue to the next section 34.3.3 Style or return to 34.3 Commands or 34 12d Archive File Format.

34.3.3 Style

The format of the **style** command is:

style *linestyle_name*

When reading a 12da file, there is a ***current linestyle,*** which has the default value of ***1***, and when a **style** command is read, the ***current linestyle*** is set to *linestyle_name.*

When strings are read in a 12da file, they are given the *current linestyle*.

This can be overridden for a string by a *string style command* inside the string command defining that string. For the definition of the string command, see 34.3.6 String.

Continue to the next section 34.3.4 Breakline or return to 34.3 Commands or 34 12d Archive File Format.

34.3.4 Breakline

The format of the **breakline** command is:

breakline *breakline_type*

where *breakline_type* is *point* or *line*.

When reading a 12da file, there is a ***current breakline type,*** which has the default value of ***point***, and when a **breakline** command is read, the ***current breakline type*** is set to *breakline_type.*

When strings are read in a 12da file, they are given the *current breakline type*.

This can be overridden for a string by a *string breakline command* inside the string command defining that string. For the definition of the string command, see 34.3.6 String.

Continue to the next section 34.3.5 Null or return to 34.3 Commands or 34 12d Archive File Format.

34.3.5 Null

The format of the **null** command is:

null *null_value*

When reading a 12da file, there is a ***current null value,*** which has the default value of ***-999***, and when a **null** command is read, the ***current null value*** is set to *null_value.*

When strings are read in a 12da file and the string has z-values equal to *null_value*, then the z-value is replaced by the ***12d Model*** null value.

This can be overridden for a string by a *null_value command* inside the string command defining that string. For the definition of the string command, see 34.3.6 String.

Continue to the next section 34.3.6 String or return to 34.3 Commands or 34 12d Archive File Format.

34.3.6 String

The format of the **string** command is:

string *string_type* {

attributes_block

string_command_1

string_command_2

...

string_command_n

}

The ***string_type*** is compulsory and must be followed by all the string information enclosed in curly braces { and }.

So if a ***string type,*** or possibly information inside the string is not recognised, the 12da reader has a chance of being able to jump over the string by looking for the end curly brace }.

Inside the braces are **string commands** as keyword pairs defining information for the string.

There can be more than one *string command* keyword pair per line as long as each keyword pair is separated by a space. In fact, the *keyword* can be on one line and the *value* on the next line.

Any unrecognised *string commands* are ignored.

The *string command keyword pairs* include **model**, **colour**, **style** and **breakline,** which are all *optional* inside the string definition. However if any of them exist inside a string definition, then the *string command keyword* overrides the current value for **model**, **colour**, **style** or **breakline** *commands* but the override is only for that particular string.

Not all string types can have an *attributes_block*.

For some string types (e.g. super string) there is more data required than just the *string command* keyword pairs.

This extra data is contained is blocks consisting of a *keyword* followed by the required information enclosed in the curly braces **{** and **}**. For example attributes for all *string types* and (x,y,z) data for a super string.

For all string types, if there is not enough recognised information to define the string, the string is ignored.

For the definition for each *string type* and the allowed *string commands* and extra data that is required for that *string type, see* 34.4 12da Definition for each String Type.

Note: if the string does not have any attributes then the ***attributes_block*** can be left out entirely (see 34.2 Attributes for the definition of *attributes_block*).

Continue to the next section 34.3.7 Tin or return to 34.3 Commands or 34 12d Archive File Format.

34.3.7 Tin

Tins (***t****riangulated* ***i****rregular* ***n****etworks*) and *Super Tins* can be written out and read in from a 12da.

```
tin {
  name    tin_name        // MANDATORY name of the tin when created in 12d Model

  time_created text              // optional - time tin first created
  time_updated text              // optional - time tin last modified


// Attributes Block:

// The attributes style, faces, null_length, null_angle, null_combined_value
// and null_combined_angle are special attributes that has extra information used by
// 12d Model to create the tin. These special attributes should not be deleted.
//
// The attributes in this block and the Attributes block itself are optional.
// When a tin is read into 12d Model from a 12da file, the style is used
// as the Tin style.

  attributes {
    text "style"            text        // name of line style for the tin
    integer "faces          0/1          // 0 non triangle data, 1 triangle data
    real "null_length"      value         // values for null by angle/length
    real "null_angle"       value    // angle in radians
    real "null_combined_length"    value
    real "null_combined_angle"      value          // angle in radians

//                  any other attributes
  }                                                  // end of attributes block


// Points Block
//
// Co-ordinates of the points at the vertices of the triangles
// The points are implicitly numbered by the order in the list (starting at point 1).
//
// The Points Block is MANDATORY

  points {                         // x y z for each point in the tin
    x-value    y-value    -value           // point 1
        "         "         "              // point 2
       "          "           "
  }                                // end of points block

// Triangles Block
//
//    Each triangle is given as a triplet of the point numbers that make up
//      the triangle vertices (the point numbers are the implicit position of the points
//      given in the Points Block.
// The order of the triangles is unimportant
//
// The Triangles Block is MANDATORY
```

```
  triangles {                         // points making up each triangle
    T1-1      T1-2      T1-3          // point numbers of the 3 vertices of first triangle.
    T2-1      T2-2      T-33          // point numbers of the 3 vertices of second triangle.
       "         "
       "         "
  }                       // end of triangles block

// Base Colour
//    The tin has a base colour that is the default colour for all triangles

  colour tin_base_colour              // optional - base colour of the tin

// Colours Block
//
// Triangles can be given colours other than the base colour by including
// a colours block. The colour for each triangle in then individually given
// (-1 means base colour). The order is the same as the order of the triangles in
// the Triangles Block.
//
// If all the triangles are the base colour, then simply omit the Colours Block

  colours {
    C1     C2     C3                  // colour for each triangle given in triangle order
    C4   C5   C6   C7                 // colour "-1" means use the base tin colour.
       "    "    "
       "    "    "
  }                       // end of colours block

// Input Block
//
// More information about how the tin was created by 12d Model.
// None of this information is needed when reading a tin into 12d Model.
// This block can be omitted

  input {                                      // data for reconstructing tin from strings
    preserve_strings     true/false            // if true, preserve breaklines etc.
    remove_bubbles       true/false            //
    weed_tin             true/false
    triangle_data        true/false
    sort_tin             true/false
    cell_method          true/false

    models {
      "model_name_1"                  // name of the first model making up the tin
      "model_name_2"                  // name of the second model making up the tin
       "    "    "
       "    "    "
    }                                 // end of models block

  }                                   // end of input block
}                                     // end of tin 12a definition
```

Continue to the next section 34.3.8 Super Tin or return to 34.3 Commands or 34 12d Archive File Format.

34.3.8 Super Tin

Super Tins, which consists of a number of **tins** (***t***riangulated ***i***rregular ***n***etworks), can be written out and read in from a 12da.

```
super_tin {
  name   tin_name        // MANDATORY name of the super tin

  time_created text              // optional - time super tin first created
  time_updated text              // optional - time super tin last modified


// Attributes Block:

// This is mainly information used by 12d Model to create the super tin.
// The attributes in this block and the Attributes block itself are optional.
//   When a super tin is read into 12d Model from a 12da file, the style is used
//   as the Super Tin style.

  attributes {
        text "style"          text       // name of line style for the tin
//             any other attributes
  }                                            // end of attributes block


// Super Tin Colour
//      The super tin has a base colour

  colour tin_base_colour         // optional - base colour of the super tin

// Tins Block
//
// This is the list of tins that make up the super tin.
// This block is MANDATORY

  tins {                                   // list of tins for the super tin
    "tin_name_1"                 // name of the first tin making up the super tin
    "tin_name_2"                 // name of the second tin making up the super tin
    "    "    "
    "    "    "
  }                              // end of tins block

}                                // end of super tin 12a definition
```

Note that the tins that make up the super tin must exist in ***12d Model*** for the super tin to be fully defined.

Continue to the next section 34.4 12da Definition for each String Type or return to 34.3 Commands or 34 12d Archive File Format.

34.4 12da Definition for each String Type

For the 12da definition of each string type, see:

34.4.1 Arc String
34.4.2 Circle String
34.4.3 Drainage String
34.4.4 Face String
34.4.5 Feature String
34.4.6 Interface String
34.4.7 Plot Frame String
34.4.8 Super String
34.4.9 Super Alignment String
34.4.10 Text String

And for the superceded strings, see:

34.4.11 2d String
34.4.12 3d String
34.4.13 4d String
34.4.16 Alignment String
34.4.14 Pipe String
34.4.17 Pipeline String
34.4.15 Polyline String

Or return to 34 12d Archive File Format.

34.4.1 Arc String

```
string arc {
  model model_name  name string_name
  colour colour_name  style style_name
  chainage start_chainage  interval value  radius value
  xcentre value  ycentre value  zcentre value
  xstart value  ystart value  zstart value
  xend value  yend value  zend value
}
```

Continue to the next section 34.4.2 Circle String or return to 34.4 12da Definition for each String Type or 34 12d Archive File Format.

34.4.2 Circle String

```
string circle {
  model model_name  name string_name
  colour colour_name  style style_name
  chainage start_chainage  interval value  radius value
  zcentre value  xcentre value  ycentre value
}
```

Continue to the next section 34.4.3 Drainage String or return to 34.4 12da Definition for each String Type or 34 12d Archive File Format.

34.4.3 Drainage String

```
string drainage {
  chainage   start_chainage
  model  model_name   name  string_name
  colour  colour_name   style  style_name
  breakline  point or line
  attributes {
    text  Tin   finished_surface_tin
    text  NSTin   natural_surface_tin
    integer   "_floating"   1|0              // 1 for floating, 0 not floating
  }
  outfall     outfall_value                  // z-value at the outfall
  flow_direction    0|1                      // 0 drainage line is defined from downstream
                                             // to upstream

  data {                                     // key word - geometry of the drainage string
    x-value     y-value     z-value     radius   bulge
      "          "          "
      "          "          "
  }
  pit {                            // pit/manhole - one pit record for each pit/manhole
                                   // in the order along the string
      name            text                   // pit name
      type            text                   // pit type
      road_name       text                   // road name
      road_chainage        chainage          // road chainage
      diameter             value             // pit diameter
      floating        yes|no                 // is pit floating or not
      chainage        pit_chainage           // internal use only
     ip               value                  // internal use only
      ratio           value                  // internal use only
      x               x-value                // x-value of top of pit
      y               y-value                // y-value of top of pit
      z               z-value                // z-value of top of pit
  }
  pipe {                           // one pipe record for each pipe connecting pits/manholes
                                   // in the order they occur along the string
      name            text                   // pipe name
      type            text                   // pipe type
      diameter        value                  // pit diameter
      us_level        value                  //
      ds_level        value                  //
      us_hgl          value                  //
      ds_hgl          value                  //
      flow_velocity        value             //
      flow_volume          value             //
  }
  property_control {
    name              text                   // lot name
    colour            colour_name
    grade             value                  // grade of pipe in units of "1v in"
    cover             value                  // cover of the of pipe
    diameter          value                  // diameter of the of pipe
    boundary          value                  // boundary trap value
    chainage          chainage               // internal use only
    ip                value                  // internal use only
    ratio             value                  // internal use only
```

```
    x                 x-value              // x value of where pipe connects to sewer
    y                 y-value              // y value of where pipe connects to sewer
    z                 z-value              // internal use only

    data {                        // key word - geometry of the property control
    x-value    y-value    z-value    radius   bulge
       "          "          "
       "          "          "
  }
  house_connection {   // warning - house connections may change in future versions
    name              text                 // house connection name
    hcb               integer              // user given integer
    colour            colour_name
    grade             value                // grade of connection in units of "1v in"
    depth             value
    diameter          value
    side              left or right
    length            value
    type              text                 // connection type
    material          text                 // material type
    bush              text                 // bush type
    level             value
    adopted_level     value
    chainage          chainage             // internal use only
    ip                value                // internal use only
    ratio             value                // internal use only
    x                 x-value              // x value of where pipe connects to sewer
    y                 y-value              // y value of where pipe connects to sewer
    z                 z-value              // internal use only
  }

}              // end of drainage-sewer data
```

Continue to the next section 34.4.4 Face String or return to 34.4 12da Definition for each String Type or 34 12d Archive File Format.

34.4.4 Face String

```
string face {
  model   model_name   name   string_name
  colour   colour_name   style   style_name
  chainage start_chainage     breakline point or line
  hatch_angle value
  hatch_distance value
  hatch_colour colour
  edge_colour colour
  fill_mode 0 or 1
  edge_mode 0 or 1
  data {                                        // keyword
     x-value    y-value    z-value
       "          "          "
  }
}
```

Continue to the next section 34.4.5 Feature String or return to 34.4 12da Definition for each String Type or 34 12d Archive File Format.

34.4.5 Feature String

```
string feature {
  model  model_name  name  string_name
  colour  colour_name  style  style_name
  chainage start_chainage  interval value  radius value
  zcentre  value  xcentre value  ycentre value
}
```

Continue to the next section 34.4.6 Interface String or return to 34.4 12da Definition for each String Type or 34 12d Archive File Format.

34.4.6 Interface String

```
string interface {
  chainage start_chainage
  model model_name   name string_name
  colour colour_name   style style_name
  breakline point or line
  data {                                                // keyword
    x-value    y-value    z-value       mode
       "          "          "            "              // mode = -1 cut
        "          "          "           "              //         0 surface
  }                                                     //         1 fill
}
```

Continue to the next section 34.4.7 Plot Frame String or return to 34.4 12da Definition for each String Type or 34 12d Archive File Format.

34.4.7 Plot Frame String

Plot frames can be written out and read in from a 12da file.

```
string plot_frame {
  name             frame_name
  title_file       filename
  border           0 or 1
  viewport         0 or 1
  user_title_file  0 or 1
  title_1          text
  title_2          text
  plot_file        filename
  text_size        mm
  sheet_code       text
  width            value
  height           value
  scale            value
  rotation         value
  xorigin          value
  yorigin          value
  left_margin      mm
  right_margin     mm
  top_margin       mm
  bottom_margin    mm
  plotter          text
  colour           colour
  textstyle        textstyle_name
}
```

Continue to the next section 34.4.8 Super String or return to 34.4 12da Definition for each String Type or 34 12d Archive File Format.

34.4.8 Super String

Because the super string is so versatile, its 12da format looks complicated but it is very logical and actually quite simple.

In its most primitive form, the super string is simply a set of (x,y) values as in a 2d string, or (x,y,z) values as in a 3d string, or (x,y,z,radius,bulge_flag) as for a polyline string or even lines, arcs and transitions (spirals and non-spiral transitions).

Additional blocks of information can extend the definition of the super string. For example, text, pipe diameters and visibility.

Some of the properties of the super string extend what were constant properties for the entire string in other string types. For example, *breakline* type for the string extends to *tinability* of *vertices* and *segments*. One colour for the string extends to individual colours for each segment.

Other properties such as vertex id's (point numbers), visibility and culvert data are entirely new.

For user attributes, the super string still has the standard user attributes defined for the entire string, but user attributes for each vertex and segment are also supported.

The definition of a closed string has been refined for polyline and super strings. For other string types, closing a string simply meant having the first vertex the same as the last vertex. Hence the vertex was duplicated.

For a super string, being closed is a property of the string and no extra vertex is needed. That is, the first and the last vertices are not the same for a closed super string and the super string knows there is an additional segment from the last vertex back to the first vertex.

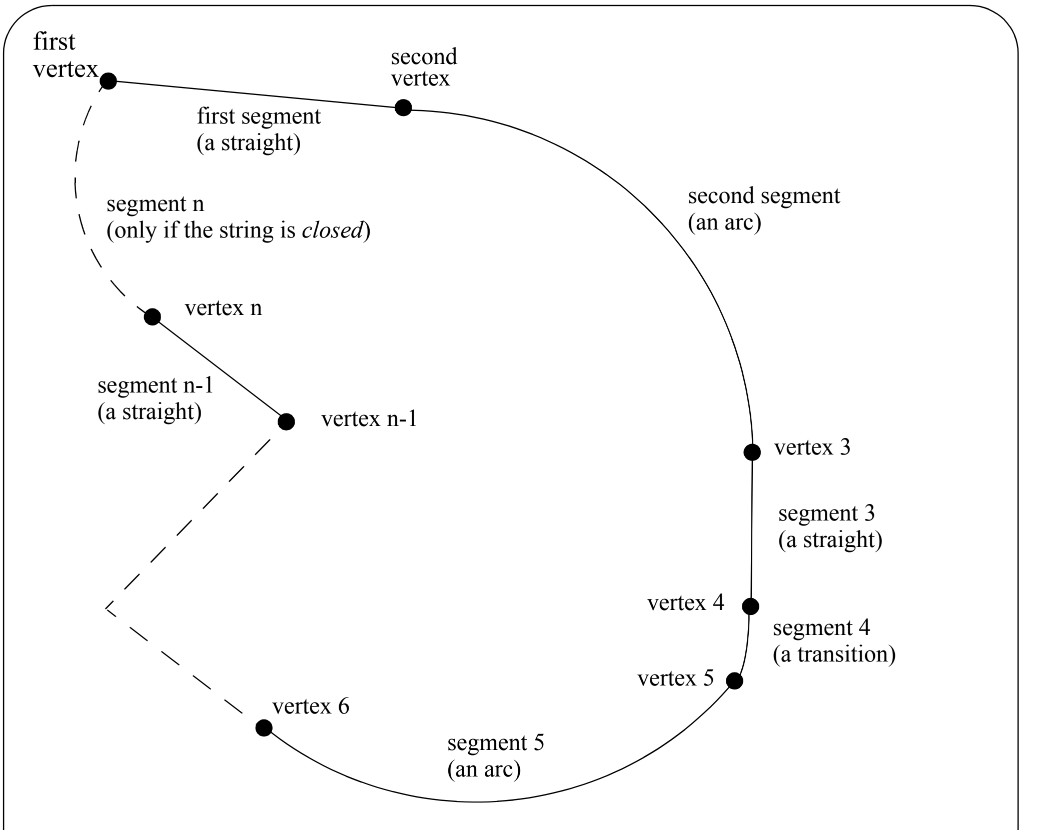
Hence in the 12da format, there is a *closed* flag for the super string:

closed *true* or *false*

where *true* can be 1 or T or t or Y or y (or words starting with T, t, Y or y))
and *false* is 0 or F or f or N or n (or words starting with F, f, N or n.

Thus if a string has *n* vertices, then an open string has *n-1* segments joining the vertices and a closed string has *n* segments since there is an additional segment from the last to the first vertex.

With the additional data for vertices and segments in the super string, the data is in vertex or segment order. So for a string with *n* vertices, there must be *n* bits of vertex data. For segments, if the string is open then there only needs to be *n-1* bits of segment data but for closed strings, there must be *n* bits of data. For an open string, *n* bits of segment data can be specified and the *nth* bit will be read in and stored. If the string is then closed, the *nth* bit of data will be used for the extra segment.



Vertices and Segments Forming the Super String

The full 12da definition of the super string is:

```
string super {
  chainage start_chainage
  model model_name  name string_name
  colour colour_name  style style_name
  breakline point or line
  closed true or false
  interval {
    chord_arc    value           // chord-to-arc tolerance for curves
    distance     value           // chainage interval to break the geometry up
  }

  block of info {
  }
  block of info {
  }
  block of info {
  }
}
```

The blocks of info can be broken up into four types.

(a) blocks defining the position of the vertices in z, y and z

data_2d **or** *data_3d*

(b) blocks defining the geometry of the segments

radius_data and *major_data* **or** *geometry_data*

(c) a superseded block defining vertices and segment geometry

data

(d) extra information for the vertices and/or segments

pipe diameters - *diameter_value or diameter_data*
culvert dimensions - *culvert_value or culvert_data*
pipe/culvert justification - *justify*
colour - *colour or colour_data*
vertex ids (point numbers) at each vertex- *point_data*
tinability - *breakline or vertex_tinability_data and segment_tinability_data*
visibility - *vertex_visible_data and segment_visible_data*
vertex text and annotation - *vertex_text_data and vertex_annotation_data*
segment text and annotation - *segment_text_data and segment_annotation_data*
symbols at vertices - *symbol_value or symbol_data*
vertex attributes - *vertex_attribute_data*
segment attributes - *segment_attribute_data*
extrudes
image data
holes

The definition for the blocks of each type now follows.

(a) Blocks Defining the Position of the Vertices

For (x, y) Values with a Constant z

If there is only (x,y) values at each vertex (like a 2d string):

```
data_2d {                                       // keyword
            x-value    y-value
               "          "
               "          "
}
```

and if there is a non-null constant z for the string

```
z    value
```

For (x,y,z) Values

If there is (x,y,z) values at each vertex (like a 3d string):

```
data_3d {                                       // keyword
   x-value    y-value    z-value
      "          "          "
      "          "          "
}
```

(b) Blocks Defining the Geometry of the Segments

Straights and Arcs Only for the Segments

If data_2d or data_3d was used, it is possible to add radius and bulge_flag data:

```
radius_data {                                   // keyword
                 radius for first segment
                 radius for second segment
```

```
                    . . .
              radius for last segment
}

major_data {                                     // keyword
              bulge flag for first segment
              bulge flag for second segment
                    . . .
              bulge flag for last segment
}
```

Straights, Arcs and Transitions (Spiral and non-Spiral Transitions) for the Segments

If data_2d or data_3d was used, it is possible to specify if the segments are straight, arcs or transitions using a *geometry_data* block.

```
geometry_data {
  segment_info_1 {
    information on the first segment
  }
  segment_info_2 {
    information on the second segment
  }
      "    "
      "    "
  segment_info_n-1 {        // the last segment if it is open
    information on the (n-1) segment
  }
  segment_info_n {          // the last segment if it is closed
    information on the n-th segment
  }
}
```

where the *segment_info* blocks are from the following:

(a) Straight

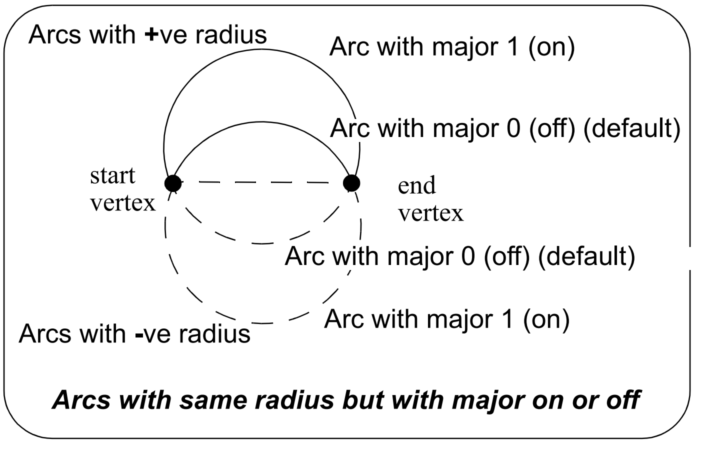
No parameters are needed for defining a straight segment. The *straight* block is simply:

```
straight {                  // no parameters are needed for a straight
}
```

(b) Arc

There are four possibilities for an arc of a given radius placed between two vertices.

We use *positive* and *negative* radius, and a flag *major* which can be set to 1 (on) or off (0) to differentiate between the four possibilities.



So the *arc* block is:

```
arc {
  radius value     // radius of the arc (+ve is above the line connecting the vertices)
  major  0 or 1    // 0 is the smaller arc, 1 the larger arc).
}
```

(c) Spiral - this covers both spiral and non-spiral transitions

There can be a partial transition between adjacent vertices. The partial transition is defined by the parameters

l1 length of the full transition up to the start vertex
r1 radius of the transition at the start vertex
a1 angle in decimal degrees of the tangent to the transition at the start vertex
l2 length of the full transition up to the end vertex
r2 radius at the end vertex
a2 angle in decimal degrees of the tangent to the transition at the end vertex

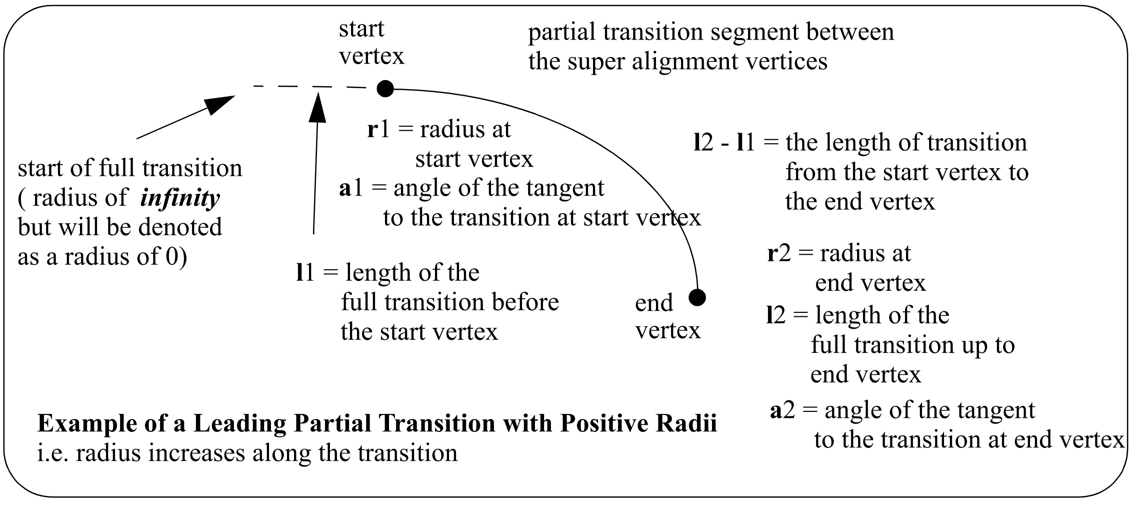
Since a radius can not be zero, a radius of infinity is denoted by *zero*.

The transition is said to be a *leading* transition if the absolute value of the radius is increasing along the direction of the transition (the transition will *tighten*). Otherwise it is a *trailing* transition.

If a leading transition is a full transition then **r**1 = 0 and **l**1 = 0. Similarly if a trailing transition is a full transition then **r**2 = 0 and **l**2 = 0.

For a partial transition, if the coordinates of the start of the full transition are needed then they can be calculated from l1,r1,a1, l2,r2,a2 and the co-ordinates of the start and end vertices.

Note that the radii can be positive or negative. If the radii's are positive then a leading transition will curl to the right (and will be above the line joining the start and end vertices).



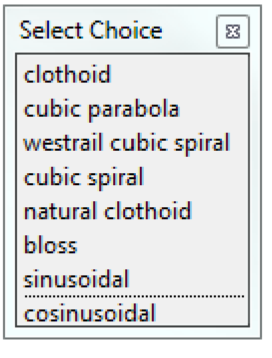
Example of a Leading Partial Transition with Positive Radii
i.e. radius increases along the transition

The parameters for the *spiral* block are:

```
spiral {
  type       value        // type can be clothoid, cubic parabola, westrail-cubic,
                          // cubic spiral, natural clothoid, bloss,
                          // bloss, sinusoidal, cosinusoidal
  leading    1 or 0       // 1 denotes a leading transition, 0 a trailing transition
  l1         value        // length of the full transition at start vertex
  r1         value        // radius at the start vertex
  a1         value        // angle in decimal degrees of the tangent to the transition
                          // at the start vertex
  l2         value        // length of the full transition at end vertex
  r2         value        // radius at end vertex
  a2         value        // angle in decimal degrees of the tangent to the transition
                          // at the end vertex
}
```

Notes

1. The ***spiral*** block covers both spiral and non-spiral transitions.
2. The transitions/spirals supported by ***12d Model*** are:



Clothoid - spiral approximation used by Australian road authorities and Queensland Rail.

Cubic parabola – special transition curve used by NSW railways. Not a spiral.

Westrail cubic – spiral approximating used by WA railways.

Cubic spiral – low level spiral approximation. Only ever used in surveying textbooks.

Natural Clothoid – the proper Euler spiral. Not used by any authority.

Bloss – special transition used by Deutsche Bahn. Not a spiral.

Sinusoidal - special transition. Not a spiral.

Cosinusoidal - special transition. Not a spiral.

(c) Block Defining the Vertices and Segments

For compatibility with the polyline, the *data* block gives the (x,y,z,radius,bulge) values at each vertex of the string and so defines both the vertices and the geometry of the segments in the one block.

```
data {                                                        // keyword
         x-value    y-value    z-value    radius   bulge
           "          "          "
           "          "          "
}
```

(d) Other Blocks

Pipe Diameters

There can be one pipe diameter value for the entire super string or the pipe diameter varies for each segment of the super string.

```
diameter_value value
```

or

```
diameter_data {                                               // keyword
         pipe diameter for first segment
         pipe diameter for second segment
                          . . .
         pipe diameter for last segment
}
```

Culvert Dimensions

There can be one culvert width and height for the entire super string or the culvert width and height vary for each segment of the super string.

```
culvert_value {
   width    value
   height    value
}
```

or

```
culvert_data { properties {width   value      // width and height for first segment
                           height  value
                          }
               properties {width   value      // width and height for second segment
                           height  value
                          }
                    . . .
               properties {width   value      // width and height for last segment
                           height  value
                          }
}
```

Justification for Pipe or Culverts

There can be only one justification for the pipe or culvert for the entire super string.

```
justify   justification            // bottom or invert
                                   // top or obvert
                                   // centre (default)
```

Colour

There can be one colour for the entire super string which is given by the `colour` command at the beginning of the string definitions (before the blocks of information) or the colour varies for each segment of the super string and is specified in a colour_data block.

```
colour_data {                                        // keyword
                colour for first segment
                colour for second segment
                    . . .
                colour for last segment
}
```

Vertex Id’s (Point Numbers)

Each vertex can have a vertex id (point number). This is not the order number of the vertex in the string but is a separate id which is usually different for every vertex in every string. The vertex id can be alphanumeric.

```
point_data {                                         // keyword
                vertex id or first vertex            // alphanumeric
                vertex id for second vertex
                    . . .
                vertex id for last vertex
}
```

Tinability

For a *super string*, the concept of breakline has been extended to a property called **tinable** which can be set independently for each vertex and each segment of the super string.

If a vertex is tinable, then the vertex is used in triangulations. If the vertex is not tinable, then the vertex is ignored when triangulating.

If a segment is tinable, then the segment is used as a side of a triangle during triangulation. This may not be possible if there are *crossing* tinable segments.

```
vertex_tinable_data {                                // keyword
                tinable flag for first vertex        // 1 for tinable
                tinable flag for second vertex       // 0 for not tinable
                    . . .
                tinable flag for last vertex


segment_tinable_data {                               // keyword
                tinable flag for first segment       // 1 for tinable
                tinable flag for second segment      // 0 for not tinable
                    . . .
                tinable flag for last segment
}
```

Note that even if a segment is set to tinable, is can only be used if both its end vertices are also tinable.

Visibility

For a *super string*, the concept of visibility and invisibility for vertices and segments has been introduced.

```
vertex_visible_data {                                  // keyword
                    visibility flag for first vertex   // 1 for visible
                    visibility flag for second vertex  // 0 for invisible
                          . . .
                    visibility flag for last vertex
}

segment_visible_data {                                  // keyword
                    visibility flag for first segment   // 1 for visible
                    visibility flag for second segment  // 0 for invisible
                          . . .
                    visibility flag for last segment
}
```

Vertex Text and Vertex Annotation

There can be the same piece of text for every vertex in the super string or a different text for each vertex of the super string. How the text is drawn is specified by vertex annotation values. Note that in vertex annotations, all vertices must be either worldsize or all vertices papersize. That is, worldsize and papersize can not be mixed - the first one found is used for all vertices.

```
vertex_text_value       text
```

or

```
vertex_text_data {                                  // keyword
                  text for first vertex             // text string, enclose
                  text for second vertex            // by " " if there are any
                    . . .                           // spaces in the text string
                  text for last vertex
}

vertex_annotate_value {                             // keyword
                  angle value   offset value   raise value
                  textstyle textstyle_name   slant degrees   xfactor value
                  worldsize value or papersize value or screensize value
                  justify   "top|middle|bottom-left|centre|right"
                  colour colour_name
}
```

or

```
vertex_annotate_data {                              // keyword
          properties {  angle value   offset value   raise value
                        textstyle textstyle   slant degrees   xfactor value
                        worldsize value or papersize value or screensize value
                        justify   "top|middle|bottom-left|centre|right"
                        colour colour_name
          }
          properties {    text properties second vertex
          }
          properties {       ...
          }
          properties {    text properties for last vertex
          }
}
```

Segment Text and Segment Annotation

There can be the same piece of text for every segment in the super string or a different text for each segment of the super string. How the text is drawn is specified by segment annotation values. Note that in segment annotations, all segments must be either worldsize or all segments papersize. That is, worldsize and papersize can not be mixed - the first one found is used for all segments. However, vertex text and segment text do not both have to be papersize or worldsize.

```
    segment_text_value          text
or
    segment_text_data {                              // keyword
                    text for first segment           // text string, enclose
                    text for second segment          // by " " if there are any
                      . . .                          // spaces in the text string
                   text for last segment
     }

    segment_annotate_value {                         // keyword
                    angle value  offset value  raise value
                    textstyle textstyle  slant degrees  xfactor value
                    worldsize value or papersize value or screensize value
                    justify  "top|middle|bottom-left|centre|right"
                    colour colour_name
     }
or
    segment_annotate_data {                          // keyword
                properties { angle value  offset value  raise value
                      textstyle textstyle  slant degrees  xfactor value
                    worldsize value or papersize value or screensize value
                      justify  "top|middle|bottom-left|centre|right"
                      colour colour_name
                }
                properties {  text properties second segment
                }
                properties {  ...
                }
                properties {   text properties for last segment
                }
     }
```

Symbols

There can be the same symbol (defined as a linestyle) for every vertex in the super string or a different symbol for each vertex of the super string. If a symbol does not have a colour, then it uses the string colour or the segment colour.

```
     symbol_value {                                  // keyword
                    style linestyle_name  colour colour_name  size value
                    rotation value                     // in dms
                    offset value  raise value
     }
or
    symbol_data {                                    // keyword
                properties { style linestyle_name  colour colour_name  size value
                    style linestyle  colour colour  size value
                    rotation value                     // in dms
                    offset value  raise value
                }
                properties {   symbol and properties for second vertex
```

```
            }
            properties  {   ...
            }
            properties  {   symbol and properties for last vertex
            }
}
```

Vertex Attributes

Each vertex can have one or more user defined named attributes.

```
vertex_attribute_data {                              // key word
            attributes  {   attribute_type   attribute_name   attribute_value
                            attribute_type   attribute_name   attribute_value
                                  . . .
                            attribute_type   attribute_name   attribute_value
            }
            attributes  {   named attributes for second vertex
            }
            attributes  {   ...
            }
            attributes  {   named attributes for last vertex
            }
}
```

Segment Attributes

Each segment can have one or more user defined named attributes.

```
segment_attribute_data {                             // keyword
            attributes  {   attribute_type   attribute_name   attribute_value
                            attribute_type   attribute_name   attribute_value
                                . . .
                            attribute_type   attribute_name   attribute_value
            }
            attributes  {   named attributes for second segment
            }
            attributes  {   ...
            }
            attributes  {   named attributes for last segment
            }
 }
```

Continue to the next section 34.4.9 Super Alignment String or return to 34.4 12da Definition for each String Type or 34 12d Archive File Format.

34.4.9 Super Alignment String

In an *alignment* string, only the intersection point method (IP's) could be used to construct the horizontal and vertical geometry. The IP definition is actually a *constructive* definition and the tangents points and segments between the tangent points (lines, arcs, transitions etc.) are calculated from the IP definition. For an alignment string, only the IP definitions are included in the 12da file.

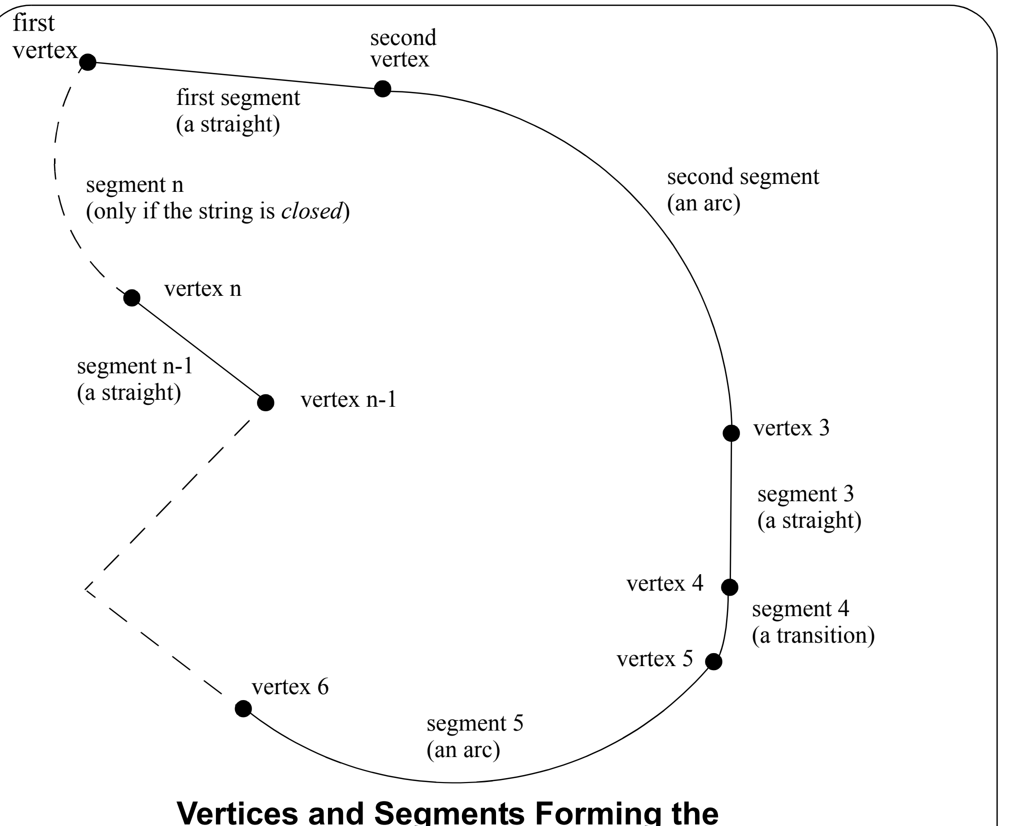
For a ***super alignment***, the horizontal and vertical geometry are also defined separately and with construction definitions but the construction definition can be much more complex than just IP's. For example, an arc could be defined as being tangential to two offset elements, or constrained to go through a given point.

If the horizontal construction methods are consistent then the horizontal geometry can be solved, and the horizontal geometry expressed in terms of consecutive segments (lines, arcs, transitions) that are easily understood and drawn.

Similarly if the vertical construction methods are consistent then the vertical geometry can be solved, and the vertical geometry expressed in terms of consecutive segments (lines, arcs, parabolas) that are easily understood and drawn.

Unlike the *alignment*, the *super alignment* stores ***both*** the ***construction methods*** (the ***parts***) and the resulting ***vertices*** and ***segments*** (lines, arcs, transitions etc.) that make up the horizontal and vertical geometry (the ***data***).

For many applications such as uploading to survey data collectors or machine control devices, only the ***horizontal data*** and the ***vertical data*** are required, not the *construction* methods (i.e. the ***horizontal*** and ***vertical parts***). When reading the 12da of a *super alignment*, only the ***horizontal*** and ***vertical data*** needs to be read in and the constructive methods (the ***horizontal*** and ***vertical parts***) can be skipped over.



Vertices and Segments Forming the Horizontal Data for a Super Alignment

Notes

1. Just using the horizontal and vertical data is valid ***as long as the super alignment geometry is consistent*** (and solves) and the horizontal and vertical parts can be created.

   There are flags in the 12da of the super alignment to say that the horizontal and vertical geometry is consistent and solves.

2. Segments meeting at a common vertex do not have to be tangential although for most road and rail applications, they should be.

The full 12da definition of the *super alignment* is:

```
string super_alignment {
//
  name              string_name
  chainage          start_chainage
  colour            colour_name
  style             style_name
  breakline         point or line
  closed            true or false
  spiral_type       transition_type          // the spiral_types are clothoid,
                                             // cubic parabola, westrail-cubic,
cubic spiral,
                                             // natural clothoid, bloss, sinusoidal
and
                                             // cosinusoidal. Note that some
spiral_type's
                                             // are non-spiral transitions

  valid_horizontal    true or false          // if true then the horizontal geometry
                                             // is consistent and solves
  valid_vertical      true or false          // if true then the horizontal geometry
                                             // is consistent and solves

  block of info {
             }
  block of info {
             }
  block of info {
             }

}                                            // end of super alignment
```

where the block of info can be one of more of:

attributes, horizontal_parts, horizontal_data, vertical_parts, vertical_data.

The *attributes* block has been described in the earlier section 34.2 Attributes.

The structure of the blocks *horizontal_parts*, *horizontal_data which define the horizontal geometry, and vertical_parts* and *vertical_data* which define the vertical geometry will now be described in more detail.

For information on *horizontal geometry,* go to Horizontal Geometry
vertical geometry Vertical Geometry

Horizontal Geometry

The horizontal geometry is described by two blocks - the *horizontal_parts* block and the *horizontal_data* block.

The ***horizontal_parts*** block contains the ***methods*** to construct the horizontal geometry such as float (fillet) an arc of a certain radius between two given lines or create a transition (spiral or non-spiral transition) between a line and an arc.

If the horizontal construction methods are consistent, then they can be solved to form a string made up of lines, arcs and transitions. The ***horizontal_data*** block is simply a list of the vertices and segments (lines, arcs etc.) that make up the ***solved*** geometry.

If the geometry in the *horizontal_parts* can be solved and produces a valid *horizontal_data* block, then the flag *valid_horizontal* in the super_alignment block is set to *true*.

```
valid_horizontal    true or false    //true if the horizontal geometry can be solved and
                                     // hence create a valid horizontal_data
horizontal_parts {/                  / methods for creating the horizontal geometry
    ....
}
horizontal_data {                    // the horizontal segments that make up the solved geometry
    ....
}
```

For information on *horizontal_parts,* go to the section Horizontal_parts
horizontal_data Horizontal_data

Horizontal_parts

The *horizontal_parts* block describes the methods used to construct the horizontal geometry of the super alignment. The parts that make up the horizontal geometry are defined in chainage order from the start to the end of the super alignment.

```
horizontal_parts {    // methods for creating the horizontal geometry
    blocks defining the sequential parts
    making up the horizontal geometry
}
```

Apart from the special case of parts defined by horizontal intersection points and their accompanying transitions and arcs, the other parts in the *horizontal_parts* block are not documented.

Horizontal_parts for defined by IP Method Only

For a horizontal intersection point (HIP) with no transitions or arc defined at that HIP, the part is defined by:

```
ip {
      id   value   // part id - a number that is unique for each horizontal and vertical part,
                   // and the value of part id is a multiple of 100
      x    value   // x co-ordinate of the horizontal intersection point
      y    value   // y co-ordinate of the horizontal intersection point
}
```

For a horizontal intersection point (HIP) with an arc but no transitions defined at that HIP, the part is defined by

```
    arc {
        id   value   // part id - a number that is unique for each horizontal and vertical part,
                     // and the value of part id is a multiple of 100
        r    value   // radius of the arc at the HIP
        x    value   // x co-ordinate of the HIP
        y    value   // y co-ordinate of the HIP
    }
```

For a horizontal intersection point (HIP) with an arc and transitions defined at that HIP, the part is defined by

```
    spiral {
        id   value   // part id - a number that is unique for each horizontal and vertical part,
                     // and the value of part id is a multiple of 100
        r    value   // radius of the arc at the HIP
        l1   value   // length of the leading transition at the HIP
        l2   value   // length of the trailing transition at the HIP
        x    value   // x co-ordinate of the HIP
        y    value   // y co-ordinate of the HIP
    }
```

Note that the *transition* used in the ***spiral*** block is given by *spiral_type* in the super_alignment block.

Hence a super alignment with horizontal geometry defined by IP methods only would consist of a horizontal_parts section with only the above ip, arc and spiral blocks in it.

```
horizontal_parts {

  ip_spiral_arc {
                 values          // values defining the ip_spiral_arc block
                 "
                 values
               }
      ....
  ip_spiral_arc {
                 values          // values defining the ip_spiral_arc block
                 "
                 values
}
```

For example,

```
horizontal_parts {
  ip {
      id 100
      x 42606.66161172
      y 37239.28824481
  }

  ip {
      id 200
      x 43134.36832349
      y 37330.26705997
  }

  spiral {
      id 300
      r 50
      l1 30
      l2 40
      x 43336.6595
      y 37469.2563
  }

  arc {
      id 400
      r 75
      x 43481.15324268
      y 37331.6431906
  }

  ip {
      id 500
      x 43627.02308964
      y 37544.94343852
  }
}
```

1st HIP
HIP only

Unique Part id
incrementing by 100

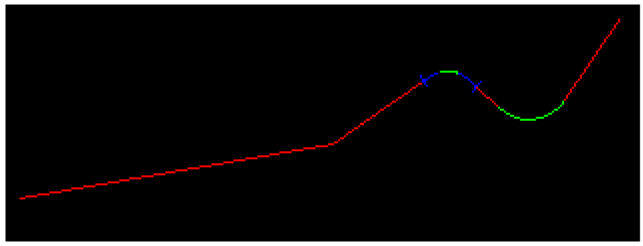
2nd HIP
HIP only

3rd HIP
HIP with arc and leading and trailing transitions

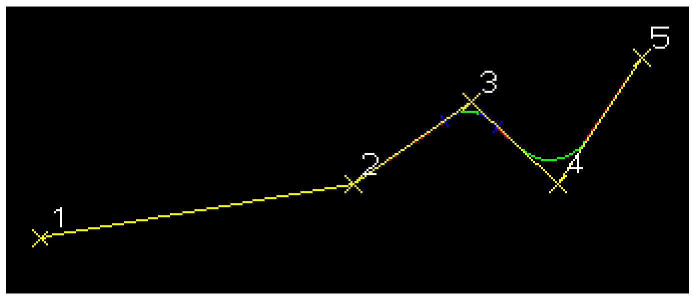
4th HIP
HIP with arc only

5th HIP
HIP only

Plan View of Super Alignment



Super Alignment Being Edited



Horizontal Parts with IP Methods Only

Horizontal_data

The *horizontal_data* block contains the ***solved*** horizontal geometry of the super alignment.

The *solved horizontal geometry* is made up of a series of (x,y) vertices given in a *data_2d* block followed by a *geometry_data* block specifying the geometry of the segments between adjacent vertices. The segment can be a straight line, an arc, a transition (e.g. a spiral) or a partial transition.

If the horizontal geometry has n vertices, then there will be (n-1) segments for an *open* super alignment or n segments if the super alignment is *closed*.

The format of the *horizontal_data* block is:

```
horizontal_data {
  name        ""
  chainage    value
  breakline   line or point
  colour      colour
  style       linestyle
  closed      0 or 1               // 0 if the string is open, 1 if it is closed
```

```
interval {
  chord_arc     value          // chord-to-arc tolerance for curves
  distance      value          // chainage interval to break the geometry up
}

data_2d {
           x1-value   y1-value          // co-ordinates of the first vertex
           x2-value   y2-value          // co-ordinates of the second vertex
              "          "
              "          "
           xn-value   yn-value          // co-ordinates of the n-th vertex
}

geometry_data {
  segment_info_1 {
    information on the first segment
  }
  segment_info_2 {
    information on the second segment
  }
      "    "
      "    "
  segment_info_n-1 {           // the last segment if it is open
    information on the (n-1) segment
  }
  segment_info_n {             // the last segment if it is closed
    information on the n-th segment
  }
}
```

where the *segment_info* blocks are from the following:

(a) Straight

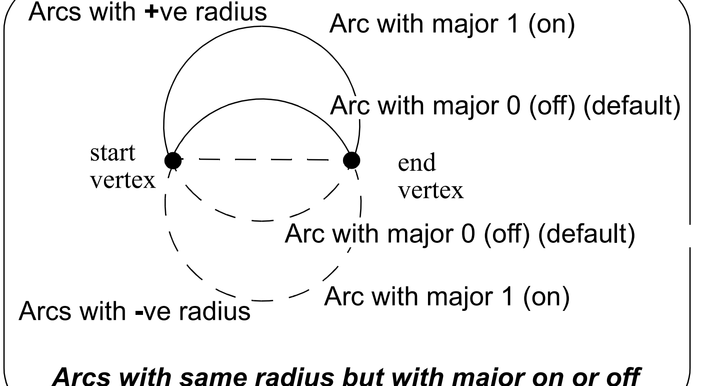
No parameters are needed for defining a straight segment. The *straight* block is simply:

```
straight {                     // no parameters are needed for a straight
}
```

(b) Arc

There are four possibilities for an arc of a given radius placed between two vertices.

We use *positive* and *negative* radius, and a flag *major* which can be set to 1 (on) or off (0) to differentiate between the four possibilities.



Arcs with same radius but with major on or off

So the *arc* block is:

```
arc {
  radius value       // radius of the arc (+ve is above the line connecting the vertices)
  major  0 or 1      // 0 is the smaller arc, 1 the larger arc).
}
```

(c) Spiral - this covers both spiral and non-spiral transitions

There can be a partial transition between adjacent vertices. The partial transition is defined by the parameters

l1 length of the full transition up to the start vertex
r1 radius of the transition at the start vertex
a1 angle in decimal degrees of the tangent to the transition at the start vertex
l2 length of the full transition up to the end vertex
r2 radius at the end vertex
a2 angle in decimal degrees of the tangent to the transition at the end vertex

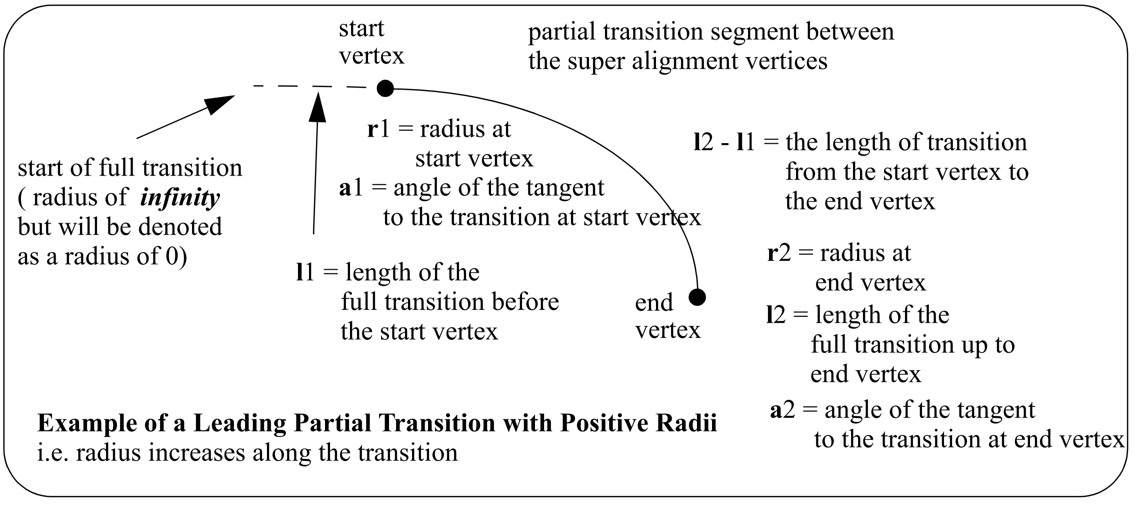
Since a radius can not be zero, a radius of infinity is denoted by *zero*.

The transition is said to be a *leading* transition if the absolute value of the radius is increasing along the direction of the transition (the transition will *tighten*). Otherwise it is a *trailing* transition.

If a leading transition is a full transition then **r**1 = 0 and **l**1 = 0. Similarly if a trailing transition is a full transition then **r**2 = 0 and **l**2 = 0.

For a partial transition, if the coordinates of the start of the full transition are needed then they can be calculated from l1,r1,a1, l2,r2,a2 and the co-ordinates of the start and end vertices.

Note that the radii can be positive or negative. If the radii's are positive then a leading transition will curl to the right (and will be above the line joining the start and end vertices).



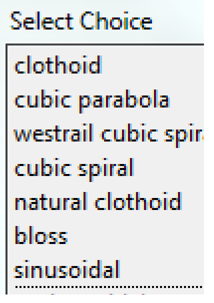
The parameters for the *spiral* block are:

```
spiral {
  type       transition_type    // any of the transitions supported in 12d
  leading    1 or 0             // 1 denotes a leading transition, 0 a trailing transition
  l1         value              // length of the full transition at start vertex
  r1         value              // radius at the start vertex
  a1         value              // angle in decimal degrees of the tangent to the transition
                                // at the start vertex
  l2         value              // length of the full transition at end vertex
```

```
    r2          value                    // radius at end vertex
    a2          value                    // angle in decimal degrees of the tangent to the transition
                                         // at the end vertex
}
```

Notes

1. The ***spiral*** block covers both spiral and non-spiral transitions.
2. The transitions/spirals supported by ***12d Model*** are:



Clothoid - spiral approximation used by Australian road authorities and Queensland Rail.

Cubic parabola – special transition curve used by NSW railways. Not a spiral.

Westrail cubic – spiral approximating used by WA railways.

Cubic spiral – low level spiral approximation. Only ever used in surveying textbooks.

Natural Clothoid – the proper Euler spiral. Not used by any authority.

Bloss – special transition used by Deutsche Bahn. Not a spiral.

Sinusoidal - special transition. Not a spiral.

Cosinusoidal - special transition. Not a spiral.

Vertical Geometry

The *vertical* geometry is described by two blocks - the *vertical_parts* block and the *vertical_data* block.

The ***vertical_parts*** block contains the ***methods*** to construct the vertical geometry such as float (fit) a parabola of a certain length between two given lines.

If the vertical construction methods are consistent, then they can be solved to form a string made up of lines, parabolas and arcs. The ***vertical_data*** block is simply a list of the vertices and segments (lines, parabolas and arcs) that make up the *solved* geometry.

If the geometry in the *vertical_parts* can be solved and produces a valid *vertical_data* block, then the flag *valid_vertical* in the super_alignment block is set to *true*.

```
valid_vertical   true or false///true if the vertical geometry can be solved and
                             // hence create a valid vertical_data
vertical_parts {             // methods for creating the vertical geometry
    ....
}
vertical_data {              // the vertical geometry
    ....
}
```

For information on *vertical_parts,* go to the section Vertical_parts
vertical_data Vertical_data

Vertical_parts

The *vertical_parts* block describes the methods used to construct the vertical geometry of the super alignment. The parts that make up the vertical geometry are defined in chainage order from the start to the end of the super alignment.

```
vertical_parts {             // methods for creating the vertical geometry
    blocks defining the sequential parts
    making up the vertical geometry
}
```

Apart from the special case of parts defined by vertical intersection points and their accompanying parabolas and arcs, the other parts in the *vertical_parts* block are undocumented.

Vertical_parts for defined by IP Method Only

For a vertical intersection point (VIP) with no parabola or arc defined at that VIP, the part is defined by:

```
ip {
        id   value    // part id - a number that is unique for each horizontal and vertical part,
                      // and the value of part id is a multiple of 100
        x    value    // chainage co-ordinate of the VIP
        y    value    // height co-ordinate of the VIP
}
```

For a vertical intersection point (VIP) with a parabola defined by a k value at that VIP, the part is defined by

```
kvalue {
        id   value    // part id - a number that is unique for each horizontal and ver-
```

```
tical part,
                          // and the value of part id is a multiple of 100
            k    value    // k-value of the parabola at the VIP
            x    value    // chainage co-ordinate of the VIP
            y    value    // height co-ordinate of the VIP
    }
```

For a vertical intersection point (VIP) with a parabola defined by length at that VIP, the part is defined by

```
    length {
            id   value    // part id - a number that is unique for each horizontal and ver-
tical part,
                          // and the value of part id is a multiple of 100
            l    value    // length of the parabola at the VIP
            x    value    // chainage co-ordinate of the VIP
            y    value    // height co-ordinate of the VIP
    }
```

For a vertical intersection point (VIP) with a parabola defined by an effective radius at that VIP, the part is defined by

```
    radius {
            id   value    // part id - a number that is unique for each horizontal and ver-
tical part,
                          // and the value of part id is a multiple of 100
            r    value    // effective radius of the parabola at the VIP
            x    value    // chainage co-ordinate of the VIP
            y    value    // height co-ordinate of the VIP
    }
```

For a vertical intersection point (VIP) with an asymmetric parabola defined by the start and end lengths at that VIP, the part is defined by

```
    length {
            id   value    // part id - a number that is unique for each horizontal and ver-
tical part,
                          // and the value of part id is a multiple of 100
            l1   value    // start length of the asymmetric parabola at the VIP
            l2   value    // end length of the asymmetric parabola at the VIP
            x    value    // chainage co-ordinate of the VIP
            y    value    // height co-ordinate of the VIP
    }
```

For a vertical intersection point (VIP) with an arc defined by a radius at that VIP, the part is defined by

```
    arc {
            id   value    // part id - a number that is unique for each horizontal and ver-
tical part,
                          // and the value of part id is a multiple of 100
            r    value    // radius of the arc at the VIP
            x    value    // chainage co-ordinate of the VIP
            y    value    // height co-ordinate of the VIP
    }
```

Hence a super alignment with vertical geometry defined by IP methods only would consist of a vertical_parts section with only the above ip, parabola and arc blocks in it.

```
vertical_parts {

  ip_parabola_arc {
                    values          // values defining the ip_parabola_arc block
                    "
                    values
  }
     ....
  ip_parabola_arc {
                    values          // values defining the ip_parabola_arc block
                    "
                    values
  }
}
```

For example,

```
vertical_parts {
  ip {
    id 600
    x -50.8459652
    y 159.79764161
  }
  kvalue {
    id 700
    k 1.25
    x 38.4627
    y 179.2126
  }
  length {
    id 800
    l 50
    x 172.61694837
    y 154.72967932
  }
  asymmetric {
    id 900
    l1 25
    l2 75
    x 270.0182
    y 208.1493
  }
  arc {
    id 1000
    r 1000
    x 424.2402
    y 196.5637
  }
  radius {
    id 1100
    r 200
    x 526.7263
    y 201.5302
  }
  ip {
    id 1200
    x 637.69216273
    y 198.71894484
  }
}
```

1st VIP
VIP only

Unique Part id
incrementing by 100

2nd VIP
Parabola defined
by k value

3rd VIP
Parabola defined
by length

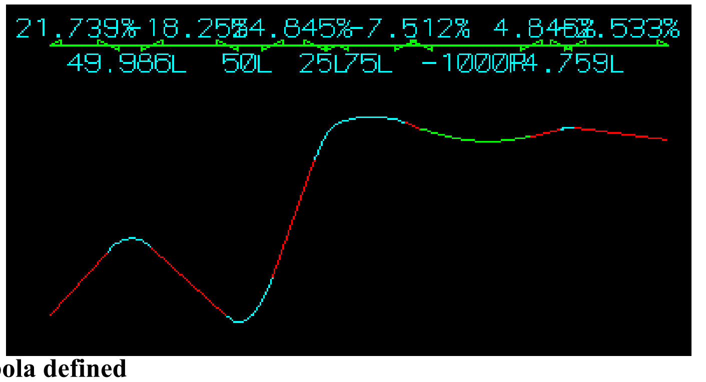
4th VIP
Asymmetric parabola defined
by two lengths

5th VIP
Arc with radius

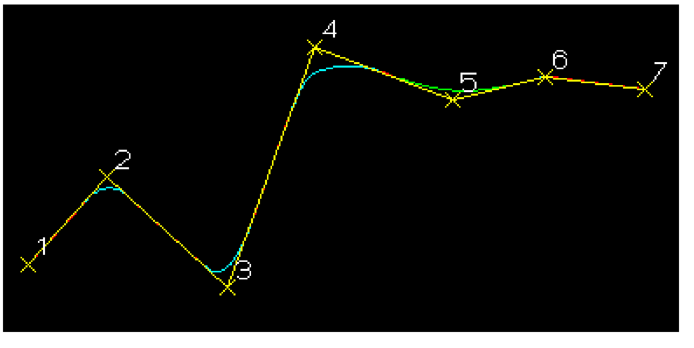
6th VIP
Parabola defined
by effective radius

7th VIP
VIP only

Section View of Super Alignment



Vertical Geometry Being Edited



Vertical Parts with IP Methods Only

Vertical_data

The *vertical_data* block contains the *solved* vertical geometry of the super alignment.

The *solved vertical geometry* is made up of a series of (chainage,height) vertices given in a *data_2d* block followed by a *geometry_data* block specifying the geometry of the segments between adjacent vertices. The segment can be a straight line, a parabola or an arc.

If the vertical geometry has n vertices, then there will be (n-1) segments for an *open* super alignment or n segments if the super alignment is *closed*.

The format of the *vertical_data* block is:

```
vertical_data {
```

```
name       ""
chainage   value
breakline  line or point
colour     colour
style      linestyle
closed     0 or 1                 // 0 if the string is open, 1 if it is closed
interval {
  chord_arc    value              // chord-to-arc tolerance for curves
  distance     value              // chainage interval to break the geometry up
}

data_2d {
           ch1-value   ht1-value        // co-ordinates of the first vertex
           ch2-value   ht2-value        // co-ordinates of the second vertex
              "           "
              "           "
           chn-value   htn-value        // co-ordinates of the n-th vertex
}

geometry_data {
  segment_info_1 {
    information on the first segment
  }
  segment_info_2 {
    information on the second segment
  }
      "    "
      "    "
  segment_info_n-1 {              // the last segment if it is open
    information on the (n-1) segment
  }
  segment_info_n {                // the last segment if it is closed
    information on the n-th segment
  }
}
```

where the *segment_info* blocks are from the following:

(a) Straight

No parameters are needed for defining a straight segment. The *straight* block is simply:

```
straight {                        // no parameters are needed for a straight
}
```

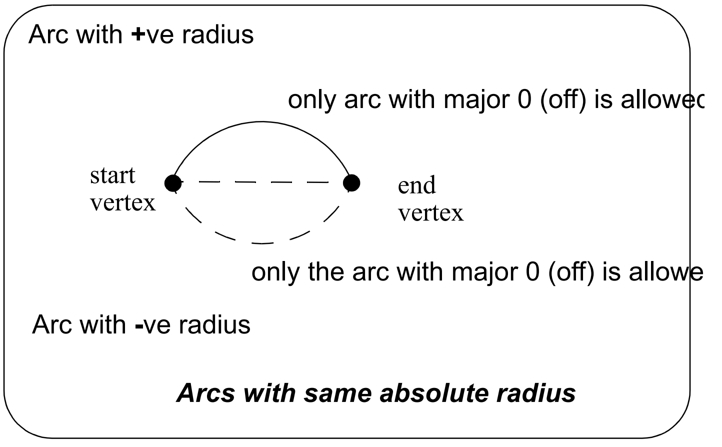
(b) Arc

Since vertical geometry can't go backwards in chainage value, the majors arcs can not be used and hence there are only possibilities for an arc of a given radius placed between two vertices.

We use *positive* and *negative* radius to differentiate between the four possibilities.

So the *arc* block is:

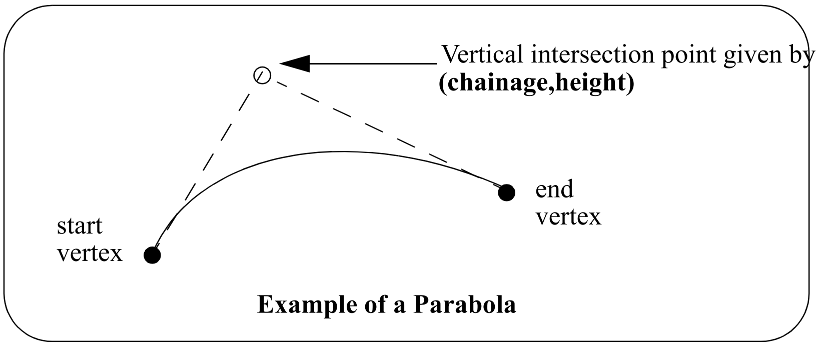
```
arc {
  radius value     // radius of the arc (+ve is above the line connecting vertices)
  major  value     // this is ignored since only minor arcs are used
}
```



(c) Parabola

There can be a parabola between adjacent vertices. The parabola is defined by giving the co-ordinates of the vertical intersection point for the parabola

chainage	chainage of the VIP of the parabola
height	height of the VIP of the parabola



The parameters for the *parabola* block are:

```
parabola {
  chainage      value        // chainage of the VIP of the parabola
  height        value        // height of the VIP of the parabola
}
```

Continue to the next section 34.4.10 Text String or return to 34.4 12da Definition for each String Type or 34 12d Archive File Format.

34.4.10 Text String

```
string  text {
  x value   y value   z value
  model model_name   name string_name   colour colour_name
  text text_value
  angle value   offset value   raise value
  textstyle textstyle_name   slant degrees   xfactor value
  worldsize value or papersize value or screensize value
  justify   "top|middle|bottom-left|centre|right"
}
```

The string types in the following sections have been superceded.

Continue to the next section 34.4.11 2d String or return to 34.4 12da Definition for each String Type or 34 12d Archive File Format.

34.4.11 2d String

The 2d string has been superceded and has been replaced by the super string (see 34.4.8 Super String).

```
string 2d {
  z   value    chainage   start_chainage
  model   model_name  name   string_name
  colour   colour_name    style   style_name
  breakline point or line
  data {                                        // keyword
    x-value    y-value
       "          "
       "          "
  }
}
```

Continue to the next section 34.4.12 3d String or return to 34.4 12da Definition for each String Type or 34 12d Archive File Format.

34.4.12 3d String

The 3d string has been superceded and has been replaced by the super string (see 34.4.8 Super String).

```
string 3d {
  chainage start_chainage
  model model_name   name   string_name
  colour colour_name   style style_name
  breakline point or line
  data {                                        // keyword
    x-value     y-value     z-value
       "           "           "
       "           "           "
  }
}
```

Continue to the next section 34.4.13 4d String or return to 34.4 12da Definition for each String Type or 34 12d Archive File Format.

34.4.13 4d String

The 4d string has been superceded and has been replaced by the super string (see 34.4.8 Super String).

```
string 4d {
  angle value   offset value   raise value
  worldsize value or papersize value or screensize value
  chainage start_chainage
  model model_name   name string_name
  colour colour_name   style style_name
  breakline point or line
  textstyle text   slant degrees   xfactor value
  justify    "top|middle|bottom-left|centre|right"
  data {                                          // keyword
    x-value   y-value    z-value    text          // text can not be blank
      "         "          "          "           // use "" for no text.
      "         "          "          "
  }
}
```

Continue to the next section 34.4.14 Pipe String or return to 34.4 12da Definition for each String Type or 34 12d Archive File Format.

34.4.14 Pipe String

The pipe string has been superceded and has been replaced by the super string (see 34.4.8 Super String).

```
string pipe {
  diameter value  chainage start_chainage
  model model_name  name string_name
  colour colour_name  style style_name
  breakline point or line
  data {                                        // keyword
    x-value  y-value  z-value
       "        "        "
       "        "        "
  }
}
```

Continue to the next section 34.4.15 Polyline String or return to 34.4 12da Definition for each String Type or 34 12d Archive File Format.

34.4.15 Polyline String

The polyline string has been superceded and has been replaced by the super string (see 34.4.8 Super String).

The definition of a closed string has been refined for polyline and super strings. For other string types, closing a string simply meant having the first vertex the same as the last vertex. Hence the vertex was duplicated.

For a polyline string, being closed is a property of the string and no extra vertex is needed - the first and the last vertices are not the same and the polyline string knows there is an additional segment from the last vertex back to the first vertex.

In the 12da format, there is a new *closed* flag for the polyline string:

closed *true* or *false*

where *true* can be 1 or T or t or Y or y (or words starting with T, t, Y or y))
and *false* is 0 or F or f or N or n (or words starting with F, f, N or n.

```
string polyline {
  chainage start_chainage
  model model_name   name string_name
  colour colour_name   style style_name
  breakline point or line
  closed true or false

  data {                                         // keyword
    x-value    y-value    z-value    radius  bulge_flag
       "          "          "
       "          "          "
  }
}
```

Continue to the next section 34.4.16 Alignment String or return to 34.4 12da Definition for each String Type or 34 12d Archive File Format.

34.4.16 Alignment String

The alignment string has been superceded and has been replaced by the super alignment (see 34.4.9 Super Alignment String).

In an alignment string the horizontal and vertical geometry are given separately and both can only be defined by the intersection point method (IP's).

For the horizontal geometry, the (x,y) position of the horizontal intersection points (HIPs) are given in the order that they appear in the string, plus the circular radius and left and right transition lengths on each HIP.

Hence a horizontal intersection point is given by either

```
x-value    y-value    radius        // circular curve, no transition
```

or

```
x-value  y-value  radius  spil1  left-transition-length   spil2  right-transition-length
```

radius, left-transition-length, right-transition-length can be zero (meaning they don't exist).

For the vertical geometry, the (chainage,height) position of the vertical intersection points (VIPs) are given in increasing chainage order, plus either the radius of the circular arc or the length of the parabolic curve on each VIP.

Hence for a vertical intersection point is given by either

```
ch_value    z-value    length   parabola
```

or

```
ch_value    z-value     radius    circle
```

where

the word *parabola* is optional. *length* and *radius* can be zero, meaning that the parabola or arc doesn't exist.

```
string alignment {
  model model_name   name string_name
  colour colour_name   style style_name
  chainage start_chainage   interval value
  draw_mode value              // 1 to draw crosses at HIPs and VIPs, 0 don't draw
  spiral_type text             // spiral_type covers both spiral and non-spiral transitions.
                               // For an alignment string, the supported transition types
                               // are clothoid, cubic parabola, westrail-cubic, cubic spiral
                               // More transition are supported in the super alignment
                               //
  hipdata {                    //  some hips must exist and precede the VIP data
    x-value y-value   radius                                   // or
    x-value y-value   radius   spil1  left-transition-length   spil2   right-transition-length
      "        "        "        "      "                       "         "
  }
  vipdata {                                                    // vips optional
    ch_value    z-value    parabolic-length                    // or
    ch_value    z-value    parabolic-length         parabola   // or
    ch_value    z-value    radius                   circle
       "           "          "           "
  }
}
```

Continue to the next section 34.4.17 Pipeline String or return to 34.4 12da Definition for each String Type or 34 12d Archive File Format.

34.4.17 Pipeline String

The pipeline string has been superceded and has been replaced by the super alignment (see 34.4.9 Super Alignment String).

This is the same as an alignment string except that it has the additional keywords

diameter, which gives the diameter of the pipeline in world units

and

length of the typical pipe making up the pipeline (used for deflections).

```
string pipeline {
  model model_name   name string_name
  colour colour_name   style style_name
  diameter diameter   length pipe-length
  chainage start_chainage   interval value
  spiral_type text            // spiral_type covers both spiral and non-spiral transitions
                              // supported by 12d. For an alignment string, the
                              // supported transition types are clothoid, cubic parabola,
                              // westrail-cubic, cubic spiral. Other transition types
                              // are supported in the super alignment
  hipdata {                   // some hips must exist and precede vips
   x-value y-value   radius                       // or
   x-value y-value   radius   spil1 left-transition-length   spil2   right-transition-length
      "       "        "        "    "                 "         "
  }
  vipdata {                                       // vips optional
    ch-value   z-value   parabolic-length         // or
    ch-value   z-value   parabolic-length   parabola // or
    ch-value   z-value   radius   circle
       "         "         "        "
  }
}
```

Return to 34.4 12da Definition for each String Type or 34 12d Archive File Format.

35 12d XML File Format

Extensible Markup Language (***XML***) is a markup language that defines a set of rules for encoding documents in a format which is both human-readable and machine-readable. It is defined by the ***World Wide Web Consortium's*** (***W3C***) XML Specifications which are free open standards.[

The 1***2d XML*** file format is a text file definition from ***12d Solutions*** which is used for reading and writing out string data from ***12d Model***. 12d XML files normally end in **.12dxml**.

The ***12d XML*** file is a **Unicode** file.

This document is for the ***12d XML*** file format used in ***12d Model*** *Version 11*.

For general comments see:

For the 12d XML definitions see:

For documentation on the ***12d Archive*** (***12da***) file format, see 34 12d Archive File Format.

35.1 General Information about XML

(Unicode) Character

By definition, an XML document is a string of characters. Almost every legal Unicode character may appear in an XML document.

Markup and Content

The characters making up an XML document are divided into ***markup*** and ***content***, which may be distinguished by the application of simple syntactic rules.

Generally, strings that constitute markup either begin with the character **<** and end with a **>**, or they begin with the character **&** and end with a **;**.

Strings of characters that are not markup are content.

However, in a CDATA section, the delimiters <![CDATA[and]]> are classified as markup, while the text between them is classified as content. In addition, whitespace before and after the outermost element is classified as markup.

Characters "<", ">" and "&"

The characters "<", ">" and "&" are key syntax markers and may never appear in content outside a CDATA section. They need to be represented by special escape sequences:

&lt represents "<"

&gt represents ">"

&amp represents "&"

Tag

An XML tag is a markup construct that begins with **<** and ends with **>**.

Tags come in three flavours:

(a) start-tags - for example: <section>

(b) end-tags - for example: </section>

(c) empty-element tags - for example: <line-break />

XML Element

A logical document component which either begins with a start-tag and ends with a matching end-tag or consists only of an empty-element tag.

The characters between the start- and end-tags, if any, are the element's content, and may contain markup, including other elements, which are called child elements.

An example of an element is **<Greeting>**Hello, world.**</Greeting>**.

Another is **<**line-break **/>**.

Note: Because elements are ***12d Model*** items that are in a model, in the documentation of 12d XML we will refrain from using element for the element in XML. Instead we will use the words ***keyword block*** to refer to special XML Elements in 12d XML.

Empty XML Elements <keyword/>

When an XML element has no content it is called an **empty** element.

For example <name> </name>

There is special shorthand for empty elements:

<keyword/> is shorthand for **<keyword></keyword>**

XML Attribute

A markup construct consisting of a name/value pair that exists within a start-tag or empty-element tag. In the example (below) the element **img** has two attributes, **src** and **alt**:

<img src="madonna.jpg" alt='Foligno Madonna, by Raphael' **/>**

Another example would be

<step number="3">Connect A to B.**</step>**

where the name of the attribute is "number" and the value is "3".

An XML attribute can only have a single value and each attribute can appear at most once on each element.

Note: Because attributes are fundamental ***12d Model*** items, in the documentation of 12d XML the word attribute will refer to ***12d Model*** attributes.

The words **XML attribute** will always be used when there is need to refer to an XML attribute.

XML declaration

XML documents may begin by declaring some information about themselves, as in the following example:

<?xml version="1.0" encoding="UTF-8"?>

Escaping

XML provides escape facilities for including characters which are problematic to include directly. For example:

There are five predefined entities:

&lt represents "<"

&gt represents ">"

&amp represents "&"

&apos represents '

&quot represents "

**
** represents a new line.

XML Comments

Comments may appear anywhere in a document outside other markup. Comments cannot appear before the XML declaration.

Comments start with "**<!--**" and end with "**-->**".

For compatibility with SGML, the string "--" (double-hyphen) is not allowed inside comments; this means comments cannot be nested.

The ampersand has no special significance within comments, so entity and character references are not recognized as such, and there is no way to represent characters outside the character set of the document encoding.

An example of a valid comment: "<!--no need to escape <code> & such in comments-->"

Continue to the next section 35.2 General Information about a 12d XML File or return to 35 12d XML File Format.

35.2 General Information about a 12d XML File

Unicode

12d XML file is a Unicode file.

Blank lines

Unless they are part of a string of characters making up text, blank lines are ignored.

Names of models, tins, styles, colours and attributes

Models, tins, styles (linestyles), colours and attributes can include the characters a to z, A to Z, 0 to 9 (alphanumeric characters) and space. Leading and trailing spaces are ignored. The names can be up to 255 characters in length.

The names for models, tins, styles, colours or attributes can not be blank.

The names for models, tins, styles and colours can contain upper and lower alpha characters which are stored, but the set of model names, tin names, style names, colour names or attribute names for an object *must be unique when case is ignored*. For example, the model name "Fred" will be stored as "Fred" but "FRED" is considered to be the *same* model name as "Fred".

String names

String names can include the characters a to z, A to Z, 0 to 9 (alphanumeric characters), space, decimal point (.), plus (+), minus (-), comma (,), open and closed round brackets and equals (=).

Leading and trailing spaces are ignored.

String names do not have to be unique and can be blank.

String names can contain upper and lower alpha characters which are retained but case is ignored when selecting by string name. That is, the string name **Fred** will be stored as **Fred** but **FRED** is not considered to be a different string name.

Keywords Blocks

There are many regularly used blocks of information in 12d XML that are identified and documented by keywords.

The keyword and its block consist of a starting **<keyword>**, followed by the information in the keyword block, and ending in **</keyword>**

That is

<keyword> information in the keyword block **</keyword>**

Continue to the next section 35.3 Regularly Used Keyword Blocks or return to 35 12d XML File Format.

35.3 Regularly Used Keyword Blocks

In the documentation of ***12d XML*** the term ***keyword* block** refers to a ***<keyword>*** followed by various information then a ***</keyword>****.*

For the definition of some of the regularly used keyword blocks used in the 12d XML *see*:

Or return to 35 12d XML File Format.

35.3.1 Name

The format of the **name** keyword block is:

<name>*name_text*</name>

where ***name_text*** is a string of characters.

What characters can be in the name depends on where the name is used. See Names of models, tins, styles, colours and attributes and String names.

Continue to the next section 35.3.2 Colour or return to 35.3 Regularly Used Keyword Blocks or 35 12d XML File Format.

35.3.2 Colour

The format of the **colour** keyword block is:

<colour>*colour_name*</colour>

where ***colour_name*** is a string of characters that is to be the name of a colour or the colour number.

When reading a 12d XML file, there is a ***current colour,*** which has the default value of ***red***, and when a **colour** command is read, the ***current colour*** is set to *colour_name.*

When strings are read in a 12d XML file, they are given the *current colour*.

This can be overridden for a string by a *string colour command* inside the string command defining that string. For the definition of the string commands, see 35.6.3 String Header Block.

Continue to the next section 35.3.3 Line Style or return to 35.3 Regularly Used Keyword Blocks or 35 12d XML File Format.

35.3.3 Line Style

The format of the **line style** keyword block is:

<style>*line_style_name*</style>

where *line_style_name* is the name of a line style. It is a string of characters.

When reading a 12d XML file, there is a ***current linestyle,*** which has the default value of ***1***, and when a **style** command is read, the ***current linestyle*** is set to *linestyle_name.*

When strings are read in a 12d XML file, they are given the *current linestyle*.

This can be overridden for a string by a *string style command* inside the string command defining that string. For the definition of the string command, see 35.6.3 String Header Block.

Continue to the next section 35.3.4 Chainage or return to 35.3 Regularly Used Keyword Blocks or 35 12d XML File Format.

35.3.4 Chainage

The format of the **chainage** keyword block is:

<chainage> *chainage_real* </chainage>

where *chainage_real* is a real value.

Continue to the next section 35.3.5 Weight or return to 35.3 Regularly Used Keyword Blocks or 35 12d XML File Format.

35.3.5 Weight

The format of the **weight** keyword block is:

<weight> *weight_real* </weight>

where *weight_real* is a real value.

Continue to the next section 35.3.6 Interval or return to 35.3 Regularly Used Keyword Blocks or 35 12d XML File Format.

35.3.6 Interval

For all elements other than the super string, the format of the **interval** keyword block is:

<interval> *interval_real* </interval>

where *interval_real* is a real value.

For a super string, the format of the **interval** keyword block is:

<interval>

<chord_arc> chord_arc_real</chord_arc>

<distance> distance_real</chord_arc>

</interval>

where *chord_arc_real and distance_real* are real values.

Continue to the next section 35.3.7 Time Created or return to 35.3 Regularly Used Keyword Blocks or 35 12d XML File Format.

35.3.7 Time Created

The format of the **time_created** keyword block is:

<time_created>*time_text*</time_created>

where ***time_text*** is a string of characters in the W3C time format.

DD-*MMM*-*YYYY*T*hh*:*mm*:*ss*Z

and

dd in the day of the month

MMM in the first three letters of the month

YYYY in the year

hh in the hour in the 24-hour clock

mm in the number of minutes

ss in the number of seconds

For example, 28-Apr-2015T06:42:45Z

Continue to the next section 35.3.8 Time Updated or return to 35.3 Regularly Used Keyword Blocks or 35 12d XML File Format.

35.3.8 Time Updated

The format of the **time_updated** keyword block is:

<time_updated>*time_text*</time_updated>

where ***time_text*** is a string of characters in the W3C time format.

DD-*MMM*-*YYYY*T*hh*:*mm*:*ss*Z

and

dd in the day of the month

MMM in the first three letters of the month

YYYY in the year

hh in the hour in the 24-hour clock

mm in the number of minutes

ss in the number of seconds

For example, 28-Apr-2015T06:42:45Z

Continue to the next section 35.3.9 Breakline or return to 35.3 Regularly Used Keyword Blocks or 35 12d XML File Format.

35.3.9 Breakline

The format of the **breakline** keyword block is:

<breakline> *breakline_type_text* </breakline>

where ***breakline_type_text*** is text and can only have the values ***point*** or ***line***.

When reading a 12d XML file, there is a ***current breakline type,*** which has the default value of ***point***, and when a **breakline** command is read, the ***current breakline type*** is set to *breakline_type_text.*

When strings are read in a 12d XML file, they are given the *current breakline type*.

This can be overridden for a string by a *string breakline command* inside the string command defining that string. For the definition of the string command, see 35.6.3 String Header Block.

Continue to the next section 35.3.10 Null or return to 35.3 Regularly Used Keyword Blocks or 35 12d XML File Format.

35.3.10 Null

NOT CERTAIN ABOUT NULL ??

The format of the **null** command is:

null *null_value*

When reading a 12d XML file, there is a ***current null value,*** which has the default value of ***-999***, and when a **null** command is read, the ***current null value*** is set to *null_value.*

When strings are read in a 12d XML file and the string has z-values equal to *null_value*, then the z-value is replaced by the ***12d Model*** null value.

This can be overridden for a string by a *null_value command* inside the string command defining that string. For the definition of the string command, see 35.6.3 String Header Block.

Continue to the next section 35.3.11 Radius or return to 35.3 Regularly Used Keyword Blocks or 35 12d XML File Format.

Continue to the next section 35.4 Attributes or return to 35 12d XML File Format.

35.3.11 Radius

The format of the **radius** keyword block is:

<radius> *radius_real* </radius>

where *radius_real* is a real value.

Continue to the next section 35.3.12 data_2d or return to 35.3 Regularly Used Keyword Blocks or 35 12d XML File Format.

35.3.12 data_2d

For some strings, there is a constant z for the entire string, or even no z value at all. For such strings only the (x,y) coordinates are required for each vertex and no space is taken up by redundant z values. Vertex data with no z-values is written out in a ***data_2d*** block.

The definition of a ***data_2d*** block is:

<data_2d>

<p>*x_value_1 y_value_1***</p>**

<p>*x_value_2 y_value_2***</p>**

...

<p>*x_value_n y_value_n***</p>**

</data_2d>

where (x_value_i, y_value_i) are the 2D coordinates of the i'th vertex.

Continue to the next section 35.3.13 data_3d or return to 35.3 Regularly Used Keyword Blocks or 35 12d XML File Format.

35.3.13 data_3d

For most string, the z value can vary for each vertex along the string and so the (x,y,z) values are required for each vertex. This vertex data is written out as a ***data_3d*** block.

The definition of a ***data_3d*** block is:

```
<data_3d>
    <p>x_value_1    y_value_1    z_value_1</p>
    <p>x_value_2    y_value_2    z_value_2</p>
        ...
    <p>x_value_n   y_value_n    z_value_n</p>
</data_3d>
```

where (x_value_i, y_value_i, z_value_i) are the 3D coordinates of the i'th vertex.

For example, for a string of 5 vertices

```
<data_3d>
  <p>42578.27649249 37366.79821468 null</p>
  <p>42523.36402317 37252.26649295 null</p>
  <p>42575.1386371 37043.59910954 null</p>
  <p>42826.16706828 37026.34090489 null</p>
  <p>42766.49603263 37412.54781911 61.53707464</p>
</data_3d>
```

35.3.14 radius_data and major_data

If there are only straight and arc segments for the string, then for either **data_2d** or **data_3d**, it is possible to add a radius and major/minor arc flag for each segment of the string using the ***radius_data*** and **major_data** blocks respectively.

The order of the entries in the *radius_data* and *major_data* blocks must match the order of the segments in the string (which is also the order in the data_2d or data_3d block).

So there is exactly one entry for each segment.

Note: If there are n vertices in the super string, then there are (n-1) segments for a open string (not closed) and n segments for a closed string.

For each segment there are five possibilities for an arc going between the vertices and these are specified by using **positive**, **zero** or **negative** values for the **radius**, and **1** or **0** for the **major flag**.

1. Straight segment - radius = 0. Major flag can be 1 or 0.
2. Positive radius and major flag 0

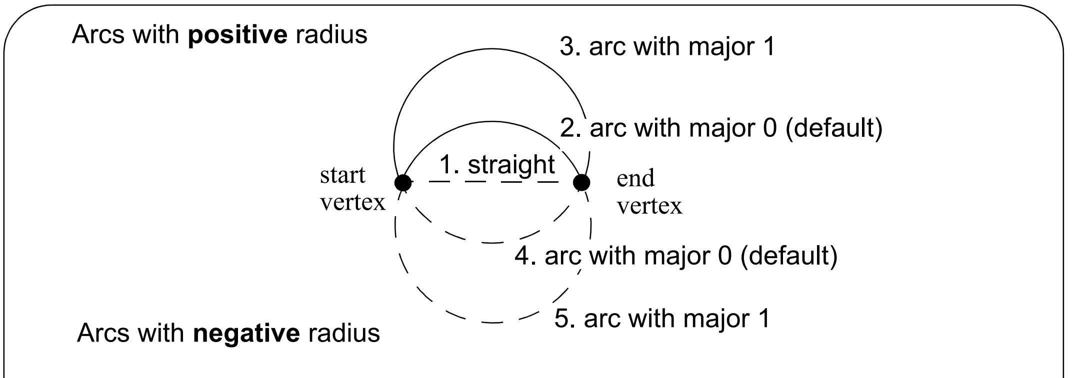
   The arc is above the straight line joining the two vertices but the arc is the smaller of the two possibilities (minor arc).
3. Positive radius and major flag1

   The arc is above the straight line joining the two vertices but the arc is the larger of the two possibilities (major arc).
4. Negative radius and major flag 0

   The arc is below the straight line joining the two vertices but the arc is the smaller of the two possibilities (minor arc).
5. Negative radius and major flag1

   The arc is below the straight line joining the two vertices but the arc is the larger of the two possibilities (major arc).

,



Arcs with same absolute radius but with major 1 or 0

The ***radius_data*** block is

<radius_data>

radius_for_segment_1

radius_for_segment_2

...

radius_for_segment_m

</radius_data>

where

radius_for_segment_i is the radius for the i'th segment and can be positive, zero or negative, and

m = n-1 for an open string or m = n for a closed string.

If the ***radius_block*** is missing then the radius is taken to be 0 and all the segments are straight lines.

The ***major_data*** block is

<major_data>

major_flag_for_segment_1

major_flag_for_segment_2

...

major_flag_for_segment_m

</major_data>

where

major_flag_for_segment_i for the i'th segment is 1 or t if the arc is a major arc, and 0 or f if it is a minor arc, and

m = n-1 for an open string or m = n for a closed string.

If the ***major_block*** is missing then the major flag is taken to be 0 and any segments with arcs are always the minor arcs.

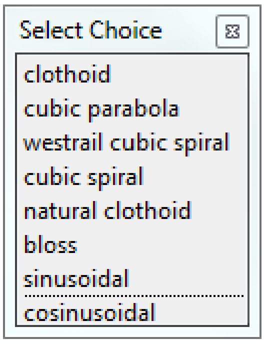
For example, for a closed string of five vertices

```
<radius_data>
    100 -300 0 0 0
</radius_data>
<major_data>
    f f f f f
```

```
</major_data>
```

35.3.15 Available Transition Types

The transition that are available are



where

clothoid, (or **spiral**) is the spiral approximation used by Australian road authorities and Queensland Rail

cubic parabola (or **state wide rail nsw**) is a special transition curve used by NSW Railways. It is not a spiral.

westrail cubic spiral (or **westrail-cubic)** is a spiral approximation used by WA railways.

cubic spiral (or **spiral**) is a low level spiral approximation. Mainly only used in surveying textbooks.

natural clothoid (or landxml spiral or clothoid landxml) is the full Euler spiral. Not used by any Authority in Australia or New Zealand.

bloss is a Bloss curve.

sinusoidal is a sinusoidal curve.

cosinusoidal is a cosinusoidal curve.

35.4 Attributes

Many *12d* ***Model*** objects (models and elements such as individual strings and tins) can have an unlimited number of named **attributes** of type integer (numbers), real and text. Super strings and drainage strings can also have attributes on each vertex and segment.

The attributes for an object are given in an ***attributes block*** which consists of the keyword **attributes** followed by the definitions of the *individual attributes* enclosed in start and end curly braces { and }. That is, an ***attributes_block*** is

<attributes>

attribute_1

attribute_2

...

attribute_n

</attributes>

where the attribute definitions for the individual attributes *attribute_i* consists of

<attribute_type>

<name> *attribute_name* ***</name>*** ***<value> attribute_value </value>***

</attribute_type>

where

attribute_type is **integer**, **real** or **text**

attribute_name is the unique attribute name for the object.

and

attribute_value is the appropriate value of the integer, real or a text.

OR

where *attribute_type* is **group**

<group>

<name> *group_name* ***</name>*** ***attributes_block***

</group>

where

group_name is the unique name of the group at this level

and

attributes_block is another attributes_block.

Note that the definition of **<group>** includes an attribute_block which can contain another **<group>** so the definition is recursive.

Hence you can have a hierarchy or tree of attributes going down to any level.

Within an object, the attribute names are case sensitive and must be unique. That is, for attribute names, upper and lower case alphabet characters are considered to be different characters.

An example of and ***attribute block*** defining four attributes named "pole id", "street", "pole height" and "pole wires" is:

```
<attributes>
  <text> <name>pole id</name> <value>QMR-37</value> </text>
  <text> <name>street</name> <value>477 Boundary St</value> </text>
```

```
  <real> <name>pole wires</name> <value>3</value> </text>
</attributes>
```

Continue to the next section 35.5 Model or return to 35 12d XML File Format.

35.5 Model

Within a ***12d Model*** project, information is collected in units called **MODELS**. The items that can be stored in a model are called elements and elements include strings, tins, super tins, grid tins, trimeshes and plot frames.

Each model has a unique user-defined text name, ***model_name***, of up to two hundred alphanumeric characters and spaces.

The format for the **model** keyword block is:

<model>

<name>*model_name*</name>

attribute_block

time_created_block

time_updated_block

<children>

element_data_1

...

element_data_n

</children>

</model>

where:

model_name is a string of characters for the model name. For the characters allowed, see

attribute_block is option. For *attributes_block* see 35.4 Attributes.

time_created_block is optional. See 35.3.7 Time Created.

time_updated_block is optional. See 35.3.8 Time Updated.

element_data_i is an element stored in the model. See35.6 Elements Contained in Models.

The ***children*** block is ***optional*** and is mainly there so that in an xml editor, the *element_data_i* items can be collapsed into the ***children*** section.

An example of a model with no elements and no *children* block:

```
<model>
  <name>telegraph poles,/name>

  <attributes>
    <text> <name>pole id</name> <value>QMR-37</value> </text>
    <text> <name>street</name><value>477 Boundary St</value></text>
    <real> <name>pole wires</name> <value>3</value> </text>
  </attributes>
</model>
```

Continue to the next section 35.6 Elements Contained in Models or return to 35 12d XML File Format.

35.6 Elements Contained in Models

See

35.6.1 Tin

A *tin* (***t****riangulated* ***i****rregular* ***n****etworks*) is an element that may, or may not, be in a model.

Each tin has text name, ***tin_name***, of up to two hundred alphanumeric characters and spaces and the names of each tin and super tin in the project must be unique.

The format for the **tin** element is:

<tin>

<name>*tin_name***</name>**

attribute_block

time_created_block

time_updated_block

colour_block

points_block

triangles_block

colours_block

input_block

</tin>

where

tin_name

is a string of characters for the tin name and can't be blank. This must be unique in a project.

For the characters that can make up a tin_name, see Names of models, tins, styles, colours and attributes.

time_created_block

is the time the tin was originally created, This is optional. For the syntax see 35.3.7 Time Created.

time_updated_block

is the last time the tin was last modified, This is optional. For the syntax see 35.3.7 Time Created.

colour_block

this colour number is the primary (base) colour for all the triangles in the tin. A triangle in the tin will have this colour unless it is overridden by a *colours_block*. For the syntax of colour_block, see 35.3.2 Colour.

attribute_block is optional: For the syntax of an *attributes_block* see 35.4 Attributes.

The attributes in this block and the attributes_block itself are optional.

The attributes **Style**, **Weed, Faces**, **Boundary_String**, **null_length**, **null_angle**, **null_combined_length** and **null_combined_angle** are special attributes that have extra information used by ***12d Model*** to create the tin. These special attributes should not be deleted.

The format of the special attributes inside the **<attributes> ... </attributes>** is:

```
<text> <name>Style</name> <value>style_name</value> </text>
<integer> <name>Weed</name> <value>weed_value</value> </integer>
<integer> <name>Faces</name> <value>faces_value</value> </integer>
<text> <name>Boundary_String</name><value>full_string_name</value></text>
<real> <name>null_length</name> <value>null_len_val</value> </real>
<real> <name>null_angle</name> <value>null_angle_rad</value> </real>
<real> <name>null_combined_length</name> <value>null_com_ln/value> <real>
```

```
<real> <name>null_combined_angle</name><value>null_com_rad</value></real>
```

where

style_name is the style for the tin

weed_value is 0 or 1

faces_value is 0 if the data is not from triangles, 1 if the data is from triangles

full_string_name is the name of a polygon for nulling outside. This is optional.

null_len_val is value for nulling by angle

null_angle_rad is in radians value for nulling by angle

null_com_ln is for nulling by combined angle and length

null_com_rad is in radians for nulling by combined angle and length

points_block

This gives the coordinates of the points that will be vertices of the triangles in the tin. The points are implicitly numbered by the order in the list (starting at point 1). The Points Block is MANDATORY.

<points>

<p>x_value_1 y_value_1 z_value_1</p>

<p>x_value_2 y_value_2 z_value_2</p>

...

<p>x_value_m y_value_m z_value_m</p>

</points>

where (x_value_j, y_value_j, z_value_j) are the coordinates of the j'th point.

triangles_block

This gives the triangles that make up the tin.

Each triangle is given as a triplet of the point numbers in the Points block that are the triangle vertices. The order of the triangles is unimportant. The Triangles Block is MANDATORY

<triangles>

<t>t1_pt_1 t1_pt_2 t1_pt_3</t>

<t>t2_pt_1 t2_pt_2 t2_pt_3</t>

...

<t>tn_pt_1 tn_pt_2 tn_pt_3</t>

</triangles>

where *tk_pt_1 tk_pt_2 tk_pt_3* are point numbers from the points_block of the three vertices of the k'th triangle.

colours_block

Triangles can be given colours other than the base colour by including a Colours Block. The colour for each triangle in then individually given where -1 means use the base colour. The order of the entries in the colours block must match the order of the triangles in the Triangles Block. So there is exactly one entry for each triangle

If all the triangles are the base colour, then the *Colours Block* is omitted.

<colours>

c1 c2 **...** *c15 c16*

c17 c18 **...** *c31 c32*

...

cn-2 cn-1 cn

</colours>

where **ck** is the colour number of the k'th triangle in the *triangles_block*.

ck equals -1 when there is no special colour set and the triangle is drawn in the base colour.

input_block

The *input_block* gives more information about how the tin was created by 12d Model.

None of this information is needed when reading a tin into 12d Model and the *input_ block* can be omitted.

<input>

<**preserve_strings**> *pres_str_text_logical* **</preserve_strings>**

<**remove_bubbles**> *rem_bub_text_logical* **</remove_bubbles>**

<**weed_tin**> *weed_tin_text_logical* **</weed_tin>**

<**triangle_data**> *triangle_data_text_logical* **</triangle_data>**

<**sort_tin>** *sort_tin_text_logical* **</sort_tin>**

<**cell_method>** *cell_method_text_logical* **</cell_method>**

<models>

model_name_1

model_name_2

...

model_name_p

</models>

</input>

where

pres_str_text_logical, rem_bub_text_logical, weed_tin_text_logical, triangle_data_text_logical, sort_tin_text_logical and *cell_method_text_logical* are text and can only have the values **true** or **false**.

<models> ... </models> is the list of models in the tin where

model_name_i is the name of the i'th model making up the tin.

Continue to the next section 35.6.2 Super Tin or return to 35.3 Regularly Used Keyword Blocks or 35 12d XML File Format.

35.6.2 Super Tin

A *Super Tins* consists of a number of **tins** (***t***riangulated ***i***rregular ***n***etworks).

Each super tin has text name, ***tin_name***, of up to two hundred alphanumeric characters and spaces and the names of each tin and super tin in the project must be unique.

The format for the **super_tin** element is:

<super_tin>

<name>tin_name</name>

attribute_block

time_created_block

time_updated_block

colour_block

exact_block

tins_block

</super_tin>

where

tin_name

is a string of characters for the super tin name and can't be blank. This must be unique in a project.

For the characters that can make up a tin_name, see Names of models, tins, styles, colours and attributes.

time_created_block

is the time the super tin was originally created, This is optional. For the syntax see 35.3.7 Time Created.

time_updated_block

is the last time the super tin was last modified, This is optional. For the syntax see 35.3.7 Time Created.

colour_block

this colour number is the primary (base) colour for the super tin. For the syntax of colour_block, see 35.3.2 Colour.

attribute_block is optional: For the syntax of an *attributes_block* see 35.4 Attributes.

The attributes in this block and the attributes_block itself are optional.

The attribute **Style** is a special attribute that is used by ***12d Model*** to create the super tin. This special attribute should not be deleted.

The format of the **Style** attribute inside the <attributes> **...** </attributes> is:

<text> <name>**Style**</name> <value>*style_name*</value> </text>

where

style_name is the style for the super tin

exact_block

<exact> *exact_text_logical* **</exact>**

where

exact_text_logical is text and can only have the value **true** or **false**.

tins_block

This gives the tins that make up the super tin within the keyword block **tins**.

<**tins**>

tin_info_1

tin_info_2

...

tin_info_p

</**tins**>

where

there are p tins in the super tin and *tin_info_i* is information about the i'th tin. The information about a tin is contained in a **tin** block.

<**tin**>

<name> tin_name_i</name>

<active> active_text_logical</active>

<mode> mode_text_logical</mode>

</**tin**>

where

tin_name_i is the name of the i'th tin making up the super tin.

active_text_logical and *mode_text_logical* are text and can only have the value **true** or **false**.

For example

```
<super_tin>
  <name>super tin</name>
  <colour>green</colour>
  <attributes>
    <text> <name>Style</name> <value>1</value> </text>
  </attributes>
  <time_created>28-Apr-2015 06:42:45</time_created>
  <time_updated>28-Apr-2015 06:42:45</time_updated>
  <exact>true</exact>
  <tins>
    <tin>
      <name>DESIGN ALL</name>
      <active>true</active>
      <mode>replace</mode>
    </tin>
    <tin>
      <name>HILL</name>
      <active>true</active>
      <mode>replace</mode>
    </tin>
  </tins>
</super_tin>
```

Note that the tins that make up the super tin must exist in ***12d Model*** for the super tin to be fully defined.

Continue to the next section 35.6.3 String Header Block or return to 35.3 Regularly Used Keyword Blocks or 35 12d XML File Format.

35.6.3 String Header Block

Strings are special types of elements that reside in a model.

Strings have common header information and this will be documented in this one spot as a ***string_header_block***.

The format for the string_header_block is:

string_name_block

chainage_block

colour_block

style_block

weight_block

interval_block

time_created_block

time_updated_block

attribute_block

where

string_name_block

The format of the **string_name_block** is:

<name> *string_name_text* </name>

where

string_name_text is a string of allowable characters that is the name of the string.

For the characters that can make up a string_name, see String names.

Any leading and trailing spaces will be removed in the string name.

string_name can be blank.

An example of a string name is:

```
<name> design 100.0 </name>
```

chainage_block

is the start chainage of the string. This is optional. For the syntax see 35.3.4 Chainage.

colour_block

the colour name is the primary colour for the string. For the syntax of colour_block, see 35.3.2 Colour.

style_block

is the line style of the string. This is optional. For the syntax of style_block see 35.3.3 Line Style.

weight_block

is the weight (thickness) of the string. This is optional. For the syntax of weight_block see 35.3.5 Weight.

interval_block

the chainage interval to temporarily introduce extra vertices into the string when the string is in a triangulation to form a tin. For the syntax of interval_block, see 35.3.6 Interval.

time_created_block

is the time the super tin was originally created, This is optional. For the syntax of time_created_block see 35.3.7 Time Created.

time_updated_block

is the last time the super tin was last modified, This is optional. For the syntax of time_updated_block see 35.3.8 Time Updated.

attribute_block

The string attributes are in this block. For the syntax of an *attributes_block* see 35.4 Attributes

The attributes_block is optional.

For example

string_header_block

```
<string_arc>
  <name>arc</name>
  <chainage>0</chainage>
  <breakline>line</breakline>
  <colour>yellow</colour>
  <style>1</style>
  <weight>2</weight>
  <time_created>28-Apr-2015 07:46:57</time_created>
  <time_updated>28-Apr-2015 07:46:57</time_updated>
  <interval>10</interval>
  <centre>1067.40263766 530.14953857 0</centre>
  <radius>226.6814323</radius>
  <chord_arc>0.1</chord_arc>
  <start>867.42825529 423.40349345 0</start>
  <end>1118.02452861 751.10631241 0</end>
</string_arc>
```

Continue to the next section 35.6.4 Text Information or return to 35.6 Elements Contained in Models or 35 12d XML File Format.

35.6.4 Text Information

See

35.6.4.1 Vertex Annotation Information
35.6.4.2 Segment Annotation Information

35.6.4.1 Vertex Annotation Information

The *vertex_annotation_information* is

<worldsize> *world_size_real* **</worldsize>**
<textstyle> *textstyle_name* **</textstyle>**
<angle> *angle_dec_deg_real* **</angle>**
<x_factor> *x_factor_real* **</x_factor>**
<slant> *slant_dec_deg_real* **</slant>**
<offset> *offset_real* **</offset>**
<raise> *raise_real* **</raise>**
<text_colour> text_colour_name **</text_colour>**
<justify> *text_justification_text* **</justify>**

where

world_size_real is the size of the text in world units.

textstyle_name is the name of the textstyle for the text.

angle_dec_deg_real is the angle of the text. The value is in decimal degrees and is measured in a counter clockwise direction from the positive x-axis.

x_factor_real is the factor to apply to the width of the text.

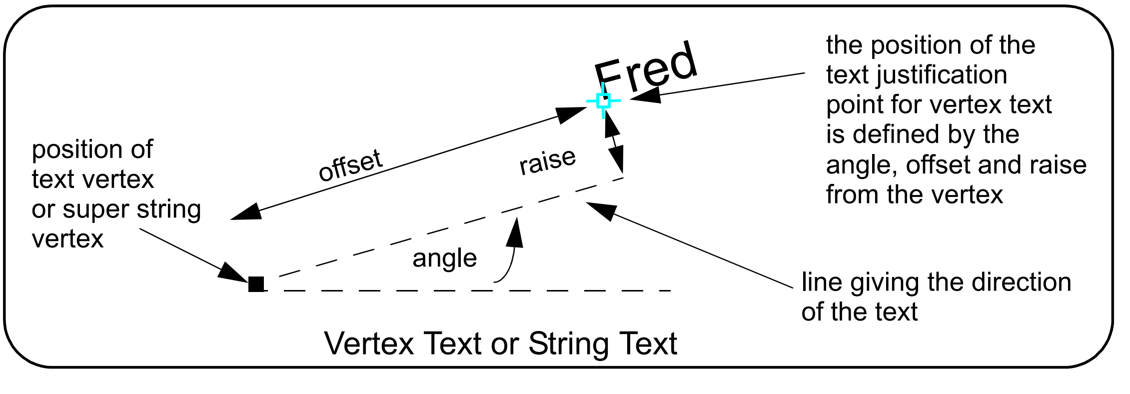
slant_dec_deg_real is the angle the text is slanted from the vertical. The value is in decimal degrees and is measured in a clockwise direction from the positive y-axis.

offset_real is distance to offset the text from the text vertex.

raise_real is the perpendicular distance the text is off the direction line of the text.

text_colour_name is the colour of the text. This should be the same as the colour in the *string_header_block*.For the syntax of colour_block, see 35.3.2 Colour.

text_justification_text is the text giving the justification point of the text.



35.6.4.2 Segment Annotation Information

The *segment_annotation_information* is

<worldsize> *world_size_real* **</worldsize>**

<textstyle> *textstyle_name* **</textstyle>**

<angle> *angle_dec_deg_real* **</angle>**

<x_factor> *x_factor_real* **</x_factor>**

<slant> *slant_dec_deg_real* **</slant>**

<offset> *offset_real* **</offset>**

<raise> *raise_real* **</raise>**

<text_colour> text_colour_name **</text_colour>**

<justify> *text_justification_text* **</justify>**

where

world_size_real is the size of the text in world units.

textstyle_name is the name of the textstyle for the text.

angle_dec_deg_real is the angle of the text. The value is in decimal degrees and is measured in a counter clockwise direction from the segment.

x_factor_real is the factor to apply to the width of the text.

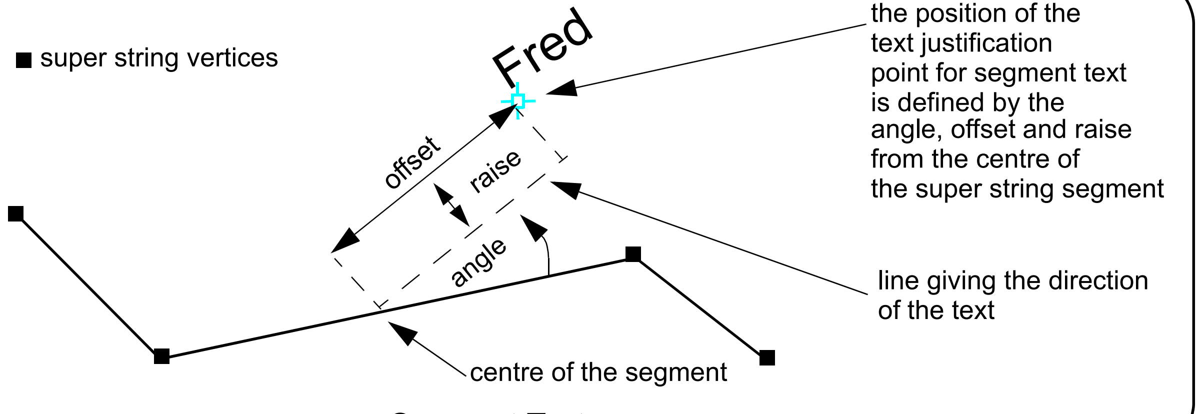
slant_dec_deg_real is the angle the text is slanted from the vertical. The value is in decimal degrees and is measured in a clockwise direction from the positive y-axis.

offset_real is distance to offset the text from the centre of the segment.

raise_real is the perpendicular distance the text is off the direction line of the text.

text_colour_name is the colour of the text. This should be the same as the colour in the *string_header_block*.For the syntax of colour_block, see 35.3.2 Colour.

text_justification_text is the text giving the justification point of the text.



Segment Text

35.6.5 Arc String

The format for the **string_arc** element is:

<string_arc>

string_header_block

centre_block

radius_block

chord_arc_block

start_block

end_block

</string_arc>

where

string_header_block

the common header block for each string. for the contents and the syntax, see 35.6.3 String Header Block.

centre_block

The format of the **centre_block** is:

<centre> *x_centre_real y_centre_real z_centre_real* </centre>

where

(*x_centre_real,y_centre_real,z_centre_real*) is the centre of the arc.

radius_block

the radius of the arc. For the syntax of radius_block, see 35.3.9 Breakline.

A positive radius means that the arc goes from the start point in a clockwise direction (goes to the right) and a negative radius means that the arc goes is in a counter clockwise direction (goes to the left).

chord_arc_block

The format of the **chord_arc_block** is:

<chord_arc>*chord_arc_real* </chord_arc>

where

chord_arc_real is a real number and is the chord to arc tolerance to use to temporarily insert vertices into the arc when the arc is included in a triangulation to form a tin.

start_block

The format of the **start_block** is:

<start> *x_start_real y_start_real z_start_real* </start>

where

(*x_start_real,y_start_real,z_start_real*) is the start coordinate of the arc.

end_block

The format of the **end_block** is:

<end> *x_end_real y_end_real z_end_real* </end>

where

(*x_end_real,y_end_real,z_end_real*) is the end coordinate of the arc.

For example

```
<string_arc>
  <name>arc</name>
  <chainage>0</chainage>
  <breakline>line</breakline>
  <colour>yellow</colour>
  <style>1</style>
  <weight>2</weight>
  <time_created>28-Apr-2015 07:46:57</time_created>
  <time_updated>28-Apr-2015 07:46:57</time_updated>
  <interval>10</interval>
  <centre>1067.40263766 530.14953857 0</centre>
  <radius>226.6814323</radius>
  <chord_arc>0.1</chord_arc>
  <start>867.42825529 423.40349345 0</start>
  <end>1118.02452861 751.10631241 0</end>
</string_arc>
```

Continue to the next section 35.6.6 Circle String or return to 35.6.3 String Header Block or 35 12d XML File Format.

35.6.6 Circle String

The format for the **string_circle** element is:

<string_circle>

string_header_block

centre_block

radius_block

chord_arc_block

</string_arc>

where

string_header_block

the common header block for each string. for the contents and the syntax, see 35.6.3 String Header Block.

centre_block

The format of the **centre_block** is:

<centre> *x_centre_real* *y_centre_real* *z_centre_real* </centre>

where

(*x_centre_real,y_centre_real,z_centre_real*) is the centre of the circle.

radius_block

the radius of the circle. For the syntax of radius_block, see 35.3.9 Breakline.

A positive radius means that the circle goes in a clockwise direction (goes to the right) and a negative radius means that the circle goes is in a counter clockwise direction (goes to the left).

chord_arc_block

The format of the **chord_arc_block** is:

<chord_arc>*chord_arc_real* </chord_arc>

where

chord_arc_real is a real number and is the chord to arc tolerance to use to temporarily insert vertices into the circle when the circle is included in a triangulation to form a tin.

For example

```
<string_circle>
  <name>circle</name>
  <chainage>0</chainage>
  <breakline>line</breakline>
  <colour>yellow</colour>
  <style>1</style>
  <weight>5</weight>
  <interval>10</interval>
  <time_created>28-Apr-2015 07:45:53</time_created>
  <time_updated>28-Apr-2015 07:46:23</time_updated>
  <centre>409.93551 548.76354 null</centre>
  <radius>100</radius>
  <chord_arc>0.1</chord_arc>
</string_circle>string circle
```

Continue to the next section 35.6.7 Drainage String or return to 35.6.3 String Header Block or 35 12d XML File Format.

35.6.7 Drainage String

The full 12dXML definition of the drainage string is:

<string_drainage>

string_header_block

outfall_block

flow_direction_block

use_pit_con_points_block

drainage_sewer_block

data_3d_block

radius_data_block

major_data_block

pit_records

pipe_records

</string_drainage>

where

string_header_block

the common header block for each string. for the contents and the syntax, see 35.6.3 String Header Block.

There are also some special attributes in the string attributes in the String Header Block that provide extra information for the drainage string.

outfall_block

<outfall> *outfall_real* **</outfall>**

where *outfall_real* is the z-value of the outfall (the low end of the string.

flow_direction_block

<flow_direction> *flow_direction_flag* **</flow_direction>**

where *flow_direction_flag* is **1** if the flow is the same as the string direction, or **0** if the flow is opposite to the string direction.

use_pit_con_points_block

<user_pit_con_points> *use_pit_connection_points_logical_text* **</user_pit_con_points>**

where *use_pit_connection_points_logical_text* is **true** if pit connection points are used, or **false** if pit connection points are not being used and hence the pipes go to the centre of he pits.

drainage_sewer_block

<drainage_sewer> *drainage_sewer_choice_text* **</drainage_sewer>**

where *drainage_sewer_choice_text* is **drainage** (storm water) if it is for drainage and **sewer** if it is for sewer (foul water).

data_3d_block, radius_data_block and major_data_block

the drainage string has an underlying string that is used to define locations of the pits and the geometry for the pipes. The underlying string can have straight and arc segments.

The vertex data for the underlying string is given in a **data_3d** block, and if there any arcs, then these are specified in **radius_data** and **major_data** blocks. See 35.3.13 data_3d and 35.3.14 radius_data and major_data.

pit_records

In plan the pits sit on the underlying string and there is one pit record for each pit. The pits do not have to be on a vertex of the underlying string.

There is one **pit** block for each pit in the string and they are in the order that they occur along the string.

The information for each **pit** is:

<pit>

<name> *pit_name_text* **</name>**

<type> *pit_type_text* **</type>**

<chainage> *pit_chainage_real* **</chainage>**

<ip> *pit_ip_text* **</ip>**

<ratio> *pit_ratio_real* **</ratio>**

<x> *pit_x_real* **</x>**

<y> *pit_y_real* **</y>**

<z> *pit_z_real* **</z>**

<road_chainage> *pit_road_chainage_real* **</road_chainage>**

<diameter> *pit_diameter_real* **</diameter>**

<width> *pit_width_real* **</width>**

<sump_level> *pit_sump_level_real* **</sump_level>**

<floating_sump> *pit_floating_sump_flag* **</floating_sump>**

<thickness> *pit_thickness_real* **</thickness>**

<thickness_bottom> *pit_thickness_bottom_real* **</thickness_bottom>**

<thickness_back> *pit_thickness_back_real* **</thickness_back>**

<thickness_left> *pit_thickness_left_real* **</thickness_left>**

<thickness_right> *pit_thickness_right_real* **</thickness_right>**

<con_point_mode> *pit_con_points_mode_text* **</con_point_mode>**

<floating> *pit_floating_logical_text* **</floating>**

<hgl> *pit_hgl_real* **</hgl>**

pit_attributes_block

</pit>

where

pipe_records

In plan the pipes sit on the underlying string and the plan geometry is based on the underlying string. Each pipe goes between two adjacent pits.

There is one **pipe** block for each pipe in the string and they are in the order that they occur along the string.

<pipe>

<name> *pipe_name_text* **</name>**

<type> *pipe_type_text* **</type>**

<colour> *pipe_colour_text* **</colour>**

<diameter> *pipe_diameter_real* **</diameter>**

<nominal_diameter> *pipe_nominal_diameter_real* **</nominal_diameter>**
<width> *pipe_width_real* **</width>**
<top_width> *pipe_top_width_real* **</top_width>**
<thickness> *pipe_thickness_real* **</thickness>**
<thickness_bottom> *pipe_thickness_bottom_real* **</thickness_bottom>**
<thickness_back> *pipe_thickness_back_real* **</thickness_back>**
<thickness_left> *pit_thickness_left_real* **</thickness_left>**
<thickness_right> *pipe_thickness_right_real* **</thickness_right>**
<separation> *pipe_separation_real* **</separation>**
<number_of_pipes> *pipe_number_of_pipes_integer* **</number_of_pipes>**
<us_level> *pipe_us_level_real* **</us_level>**
<ds_level> *pipe_ds_level_real* **</ds_level>**
<us_hgl> *pipe_us_hgl_real* **</us_hgl>**
<ds_hgl> *pipe_ds_hgl_real* **</ds_hgl>**
<flow_velocity> *pipe_flow_velocity_real* **</flow_velocity>**
<flow_volume> *pipe_flow_volume_real* **</flow_volume>**
pipe_attributes_block
</pipe>

```
string drainage {
  chainage   start_chainage
  model   model_name   name  string_name
  colour   colour_name   style  style_name
  breakline  point or line
  attributes {
    text   Tin   finished_surface_tin
    text   NSTin   natural_surface_tin
    integer   "_floating"   1|0             // 1 for floating, 0 not floating
  }
  outfall     outfall_value                 // z-value at the outfall
  flow_direction    0|1                     // 0 drainage line is defined from downstream
                                            // to upstream

  data {                                    // key word - geometry of the drainage string
    x-value     y-value     z-value     radius   bulge
       "           "           "
       "           "           "
  }
  pit {                         // pit/manhole - one pit record for each pit/manhole
                                // in the order along the string
      name            text                  // pit name
      type            text                  // pit type
      road_name       text                  // road name
      road_chainage       chainage          // road chainage
      diameter            value             // pit diameter
      floating        yes|no                // is pit floating or not
      chainage        pit_chainage          // internal use only
```

```
    ip              value                  // internal use only
      ratio         value                  // internal use only
      x             x-value                // x-value of top of pit
      y             y-value                // y-value of top of pit
      z             z-value                // z-value of top of pit
}
pipe {                          // one pipe record for each pipe connecting pits/manholes
                                // in the order they occur along the string
      name          text                   // pipe name
      type          text                   // pipe type
      diameter      value                  // pit diameter
      us_level      value                  //
      ds_level      value                  //
      us_hgl        value                  //
      ds_hgl        value                  //
      flow_velocity     value              //
      flow_volume       value              //
}
property_control {
  name              text                   // lot name
  colour            colour_name
  grade             value                  // grade of pipe in units of "1v in"
  cover             value                  // cover of the of pipe
  diameter          value                  // diameter of the of pipe
  boundary          value                  // boundary trap value
  chainage          chainage               // internal use only
  ip                value                  // internal use only
  ratio             value                  // internal use only
  x                 x-value                // x value of where pipe connects to sewer
  y                 y-value                // y value of where pipe connects to sewer
  z                 z-value                // internal use only

  data {                        // key word - geometry of the property control
  x-value     y-value     z-value     radius   bulge
     "           "           "
     "           "           "
}
house_connection {   // warning - house connections may change in future versions
  name              text                   // house connection name
  hcb               integer                // user given integer
  colour            colour_name
  grade             value                  // grade of connection in units of "1v in"
  depth             value
  diameter          value
  side              left or right
  length            value
  type              text                   // connection type
  material          text                   // material type
  bush              text                   // bush type
  level             value
  adopted_level     value
  chainage          chainage               // internal use only
  ip                value                  // internal use only
  ratio             value                  // internal use only
  x                 x-value                // x value of where pipe connects to sewer
  y                 y-value                // y value of where pipe connects to sewer
  z                 z-value                // internal use only
}
```

```
}                    // end of drainage-sewer data
```

Continue to the next section 35.6.8 Feature String or return to 35.6.3 String Header Block or 35 12d XML File Format.

35.6.8 Feature String

The full 12dXML definition of the drainage string is:

<string_feature>

string_header_block

<radius> *feature_radius_real* **</radius>**

<centre> *x_centre_real y_centre_real   z_centre_real* **</centre>**

</string_feature>

where

string_header_block

the common header block for each string. for the contents and the syntax, see 35.6.3 String Header Block.

There are also some special attributes in the string attributes in the String Header Block that provide extra information for the drainage string.

feature_radius_real is the radius of the feature string.

(xcentre_real, y_centre_real, z_centre_real) is the centre of the feature string.

For example

```
<string_feature>
  <name>Line 1</name>
  <chainage>0</chainage>
  <breakline>line</breakline>
  <colour>cyan</colour>
  <style>1</style>
  <time_created>2015-05-19T08:06:01Z</time_created>
  <time_updated>2015-05-19T08:06:01Z</time_updated>
  <centre>42200.06055 37384.05873 null</centre>
  <radius>20</radius>
</string_feature>
```

Continue to the next section 35.6.9 Plot Frame String or return to 35.6.3 String Header Block or 35 12d XML File Format.

35.6.9 Plot Frame String

The format for the **string_plot_frame** element is:

<string_plot_frame>

info_block

time_created_block

time_updated_block

sheet_details_block

title_block_block

origin_block

scale_block

rotation_block

plotter_details_block

</string_plot_frame>

where

info_block

The format of the **info_block** is:

<info>

<name> *plot_frame_name_text*</name>

<colour> *plot_frame_name_colour_text*</colour>

<plot_file> *plot_file_name_text*</plot_file>

</info>

where

plot_frame_name_text is a string of allowable characters that is the name of the plot file string. For the characters that can make up a string_name, see String names.

plot_frame_colour_text is the colour of the plot frame. For the syntax of colour_block, see 35.3.2 Colour.

plot_file_name_text is the name of file that the plot frame will plot to.

time_created_block

is the time the plot frame was originally created, This is optional. For the syntax of the time_created_block see 35.3.7 Time Created.

time_updated_block

is the last time the plot frame was last modified, This is optional. For the syntax of the time_updated_block see 35.3.8 Time Updated.

sheet_details_block

The format of the **sheet_details_block** is:

<sheet_details>

<sheet_code> *sheet_code_text*</sheet_code>

<width> *sheet_width_real*</width>

<height> *sheet_height_real*</height>

<left_margin> *sheet_left_margin_real*</left_margin>

<right_margin> *sheet_right_margin_real*</right_margin>

<top_margin> *sheet_top_margin_real*</top_margin>

<bottom_margin> *sheet_bottom_margin_real*</bottom_margin>

<border> *sheet_border_text_logical*</border>

<viewport> *sheet_viewport_text_logical*</viewport>

</sheet_details>

where

sheet_code_text is the name of the sheet. This can be blank.

sheet_width_real, sheet_height_real, sheet_left_margin_real, sheet_right_margin_real, sheet_top_margin_real, sheet_bottom_margin_real are all real values and give the size and margins for he sheet that the plot frame will plot. The units for all of them is millimetres.

plot_frame_border_text_logical and *plot_frame_viewport_text_logical* are text and can only have the value **true** or **false**.

origin_block

The format of the **origin_block** is:

<origin> *x_real   y_real   z_real*</origin>

where

(*x_real,y_real,z_real*) is the coordinates of the origin of the plot frame.

scale_block

The format of the **scale_block** is:

<scale> *scale_real*</scale>

where

scale_real is the 1: scale for the plots created by the plot frame.

rotation_block

The format of the **rotation_block** is:

<rotation> *rotation_dec_deg_real*</rotation>

where

rotation_dec_deg_real is rotation of the plot frame. The value is in decimal degrees and is measured in a counter clockwise direction from the positive x-axis.

plotter_details_block

The format of the **plotter_details_block** is:

<plotter_details>

<title_1>*title_1_text*</title_1>

<title_2>*title_2_text*</title_2>

<use_title_file> *title_file_text_logical*</border>

<title_file> *title_file_name_text*</title_file>

<text_size> *title_text_size_real*_mm</text_size>

<textstyle> *title_text_style*</textstyle>

</plotter_details>

where

title_1_text and *title_2_text* are two lines of text for the title block. They can be blank.

use_title_file_text_logical is text and can only have the value **true** or **false**.

title_file is the path name of the file to use as a title block file. This can be blank.

title_text_size_real_mm is the size of the text in the title block. The units are millimetres.

For example

```
<string_plot_frame>
  <info>
    <name>Plot frame</name>
    <colour>green</colour>
    <plot_file>plot</plot_file>
  </info>
  <sheet_details>
    <sheet_code>A0</sheet_code>
    <width>1189</width>
    <height>841</height>
    <left_margin>5</left_margin>
    <right_margin>10</right_margin>
    <bottom_margin>5</bottom_margin>
    <top_margin>10</top_margin>
    <border>true</border>
    <viewport>true</viewport>
  </sheet_details>
  <title_block>
    <title_1>Title 1</title_1>
    <title_2>Title 2</title_2>
    <use_title_file>true</use_title_file>
    <title_file>A0 title.tbf</title_file>
    <text_size>5</text_size>
    <textstyle>1</textstyle>
  </title_block>
  <origin>695.2353 1464.6248</origin>
  <scale>100</scale>
  <rotation>45</rotation>
  <plotter_details>
    <id>9</id>
    <type>model</type>
    <mode>""</mode>
    <names>""</names>
  </plotter_details>
  <time_created>29-Apr-2015 01:11:52</time_created>
  <time_updated>29-Apr-2015 01:11:52</time_updated>
</string_plot_frame>
```

Continue to the next section 35.6.10 Super String or return to 35.6.3 String Header Block or 35 12d XML File Format.

35.6.10 Super String

Because the super string is so versatile, its 12d XML format looks complicated but it is very logical and actually quite simple.

In its most primitive form, the super string is simply a set of (x,y) values as in a 2d string, or (x,y,z) values as in a 3d string.

Additional blocks of information can extend the definition of the super string and only need to be included if they exist. For example, segment arcs or transitions, vertex ids, vertex and segment text, round pipe diameters or box pipes widths and heights and tinability.

Some of the properties of the super string can be constant for the entire string or can vary for each vertex and/or segment. For example, there can be one colour for the entire string or individual colours for each segment.

For user attributes, the super string not only has the standard user attributes defined for the entire string (string attributes), but also can have user attributes for each vertex (vertex attributes) and for each segment (segment attributes).

Being closed or not is another property of the super string and if the super string is closed then the super string knows there is an additional segment going from the last vertex back to the first vertex. This means that there is no duplication of the first and last vertex needed.

Thus if a super string has *n* vertices, then an open super string has *n-1* segments joining the vertices and a closed super string has *n* segments since there is an additional segment from the last to the first vertex.

With the additional data for vertices and/or segments in the super string, the data is in vertex or segment order.

So for a string with *n* vertices, there must be *n* bits of vertex data.

For segments, if the string is open then there only needs to be *n-1* bits of segment data but for closed strings, there must be *n* bits of data.

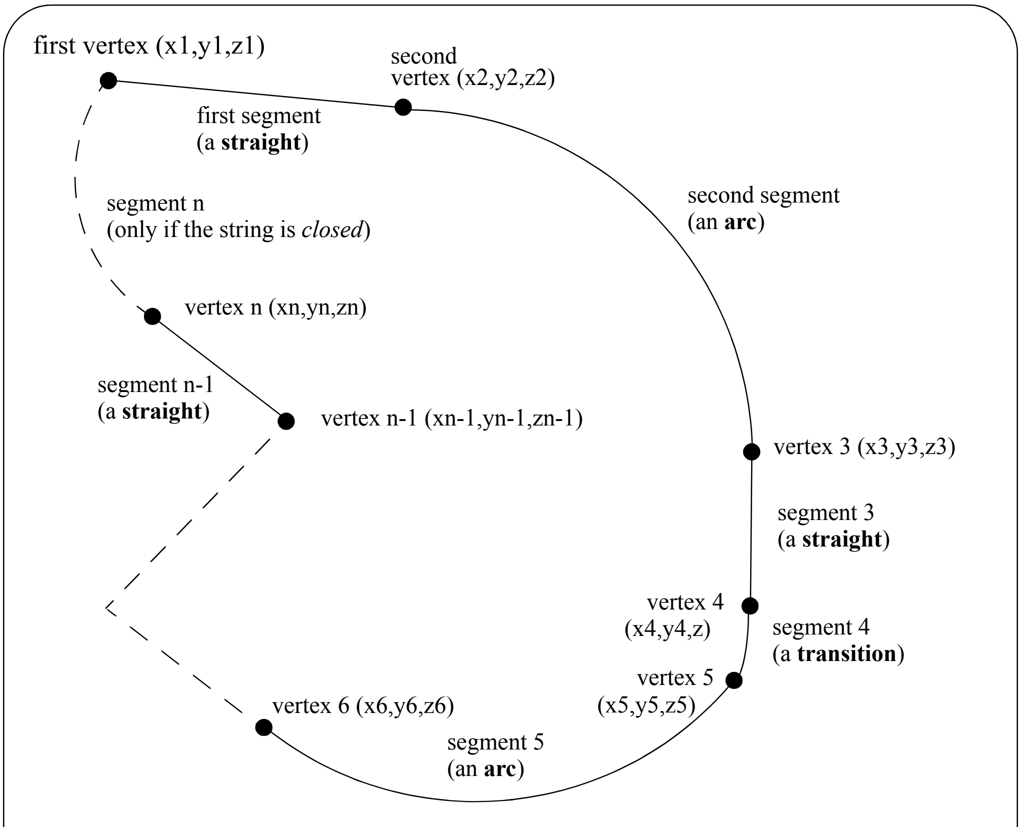
For an open string, *n* bits of segment data can be specified and the *nth* bit will be read in and stored. If the string is then closed, the *nth* bit of data will be used for the extra segment.

Important Note

For a super string, the arcs, transitions and offset transitions are that shape in plan.

Hence an arc with a z-value at each end is actually a **helix** and **NOT** part of a three dimensional circle.

Transitions and offset transitions are helixes with a constantly changing radius.



Vertices and Segments Forming the Super String

The 12dXML definition of the super string is:

<string_super>

string_header_block

closed_block

interval_block

blocks_of_info_1

blocks_of_info_2

...

blocks_of_info_n

</string_super>

where

string_header_block

the common header block for each string. for the contents and the syntax, see 35.6.3 String Header Block.

closed_block

<closed> closed_text_logical **</closed>**

where *closed_text_logical is* **true** if the super string is closed and **false** if the super string is open.

interval_block

The interval_block for a super string has a *distance* (a chainage interval) and a *chord_to_arc*_real

where

the ***distance*** to temporarily introduce extra vertices into the string at the given chainage distance when the string is in a triangulation to form a tin.

chord_arc_real is a real number and is the chord to arc tolerance to use on any arcs in the super string to temporarily insert vertices into the arc when the arc is included in a triangulation to form a tin.

For the syntax of interval_block, see 35.3.6 Interval.

blocks_of_info

The blocks of info can be broken up into four types.

(a) blocks defining the position of the vertices in z, y and z

Each vertex must have at least an (x,y) value but there may be one z-value for the entire string and (x,y) at each vertex (data_2d), **or** an (x,y,z) for each vertex (data_3d).

See 35.6.10.1 Defining the Coordinates of the Vertices

(b) blocks defining the geometry of the segments

Segments can be straights, arcs, transitions or offset transitions.

radius_data and *major_data* **or** *geometry_data.*

See 35.6.10.2 Geometry of the Horizontal Segments

(c) extra information for the vertices and/or segments such as colour, attributes, vertex ids, symbols tinability etc.

The definition for the blocks of each type now follows.

35.6.10.1 Defining the Coordinates of the Vertices

35.6.10.2 Geometry of the Horizontal Segments

35.6.10.3 Colour

35.6.10.4 String, Vertex and Segment Attributes

35.6.10.5 Vertex Id’s (Point Id’s)

35.6.10.6 Symbols at Vertices

35.6.10.7 Tinability

35.6.10.8 Round or Box (Culvert) Pipes

35.6.10.9 Vertex Text and Vertex Annotation

35.6.10.10 Segment Text and Segment Annotation

35.6.10.1 Defining the Coordinates of the Vertices

See

35.6.10.1.1 One Z or No Z for the String
35.6.10.1.2 Varying Z Values along the String

35.6.10.1.1 One Z or No Z for the String

If there is a non-null constant z value for the entire string then the z value is given by a **z** block:

<z> *z_value* **</z>**

where *z_value* is the constant z value for the entire string.

And when there is a constant z, or no z, for the string, then only the (x,y) coordinates are required for each vertex and these are given in a ***data_2d*** block. See 35.3.12 data_2d

35.6.10.1.2 Varying Z Values along the String

If the z value can vary for different vertices along the string then the (x,y,z) values must be given for each vertex and the data is then written out as a ***data_3d*** block. See 35.3.13 data_3d.

35.6.10.2 Geometry of the Horizontal Segments

If the segments are straight lines only then that is the default and no further information is required.

If the segments are only straight lines and arcs, then the ***radius_data*** and **major_data** blocks are used to define a ***radius*** and ***bulge_flag*** data for each segment of the super string. See 35.6.10.2.1 Only Straights and Arcs for Segments.

If any of the segments are transitions or offset transitions then ***geometry_data*** must be used for each segment. ***geometry_data*** can represent a straight, arc, transition or offset transition. See 35.6.10.2.2 Straights, Arcs and Transitions for Segments.

35.6.10.2.1 Only Straights and Arcs for Segments

If there are only straight and arc segments for the string, then for either **data_2d** or **data_3d**, it is possible to add a radius and major/minor arc flag for each segment of the super string using the ***radius_data*** and **major_data** blocks respectively. See 35.3.14 radius_data and major_data.

35.6.10.2.2 Straights, Arcs and Transitions for Segments

When some of the segments are transitions or offset transitions, then the ***geometry_data*** block **must** be used the give the geometry for each segments.

Either a **data_2d** or **data_3d** block defines the coordinates for the vertices and the ***geometry_data*** block defines for each segment whether the segment is a straight, an arc or a transition or offset transition.

The definition of the ***geometry_data*** block is

<geometry_data>

info_for_segment_1_block

info_for_segment_2_block

...

info_for_segment_m_block

</geometry_data>

where

info_for_segment_i_block is the information defining the i'th segment as either a straight, an arc or an offset transition (offset transition or transition), and

m = n-1 for an open string or m = n for a closed string.

For the definition of *info_for_segment_i_block* see:

35.6.10.2.2.1 Straight

35.6.10.2.2.2 Arc

35.6.10.2.2.3 Offset Transitions

35.6.10.2.2.1 Straight

No parameters are needed for defining a straight segment. The *straight* block is simply:

<straight> </straight>

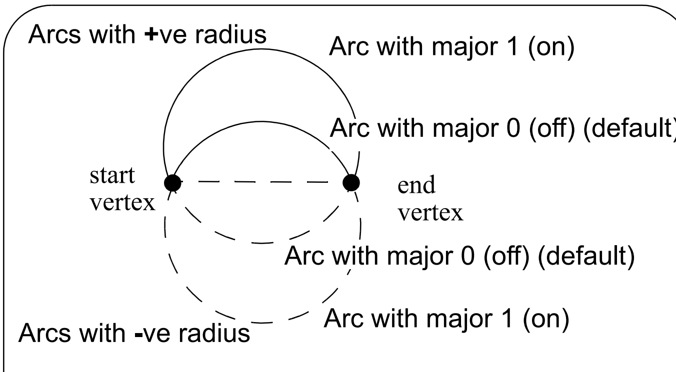
or simply

<straight/>

35.6.10.2.2.2 Arc

There are four possibilities for an arc of a given radius placed between two vertices.

We use *positive* and *negative* radius, and a flag *major* which can be set to 1 (on) or off (0) to differentiate between the four possibilities.



Arcs with same radius but with major on or off

The *arc* block is:

<arc>

<radius> *radius_for_segment*</radius>

<major> *major_flag_for_segment*</major>

</arc>

where

radius_for_segment is the radius for the segment and

major_flag_for_segment is 1 if the arc is a major arc and 0 if it is a minor arc.

35.6.10.2.2.3 Offset Transitions

An ***offset transition*** is a fixed perpendicular offset (***offset_real)*** of a base transition where the base transition is a Euler spiral (or a certain approximation to it) or some other specially defined curve. The base transition has a start point where the absolute radius of the curve is infinity and then has a continuously decreasing absolute radius as you continue along the curve (this may be in a forward or reverse direction).

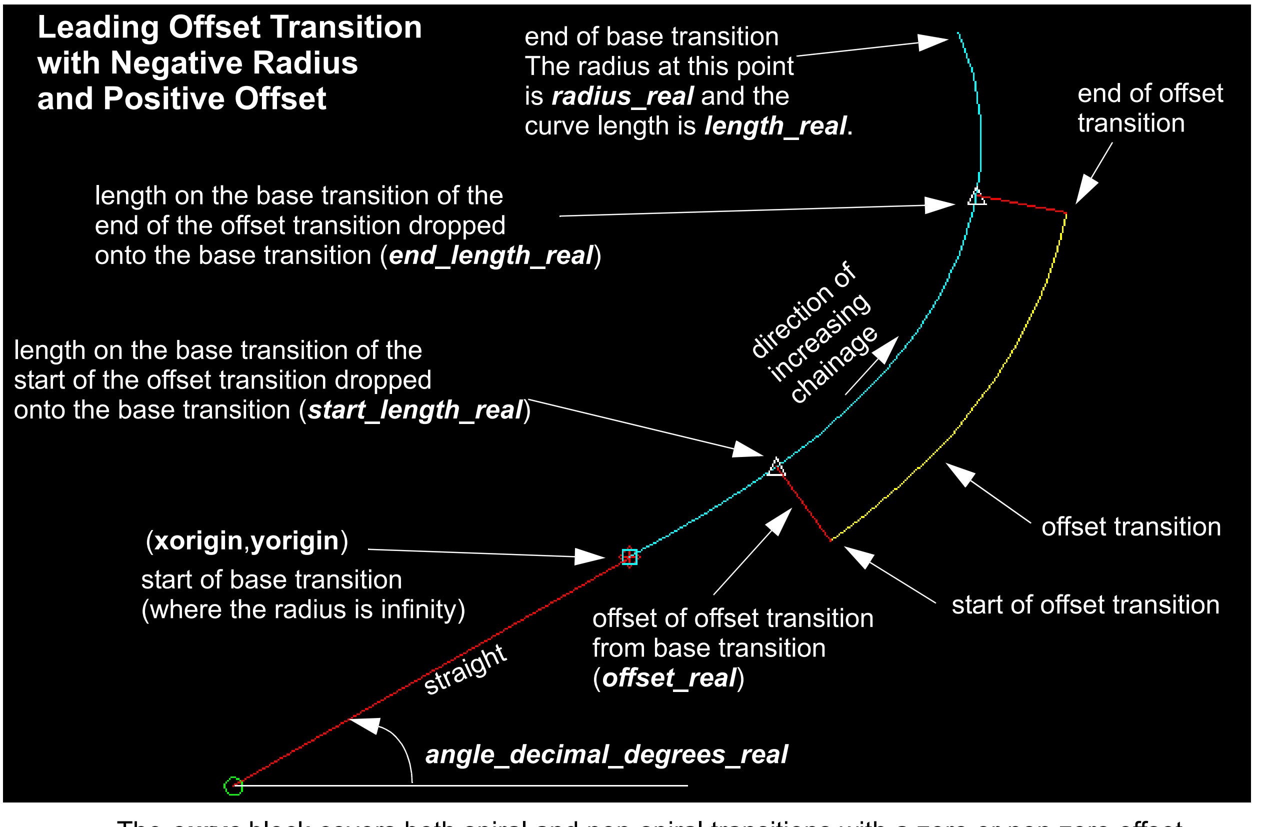
The ***base transition*** is defined by giving its starting point **(xorigin, yorigin)** where the radius is infinity and the angle of the tangential line at the start point is ***angle_decimal_degrees_real*** and the fact that the radius ***radius_real*** occurs at a given curve length ***length_real*** along the base transition.

The ***offset transition*** is a fixed offset (***offset_real***) from the base transition and goes from a start point that is specified by giving the length on the base transition where the start point drops perpendicularly onto the base transition (***start_length_real***) and to the end point that is specified by length on the base transition where the end point drops perpendicularly onto the base transition (***end_length_real***). The offset can be positive or negative.

If you are travelling along the curve in a forward direction (increasing chainage) then the base transition is said to be a **leading transition** if the absolute radius decreases as you go along the curve, and a **trailing transition** if the absolute radius decreases.

The ***end radius*** can be ***positive*** or ***negative***.

If you are travelling along the curve in a forward direction then for a leading transition, if the **end radius** is **positive** then the curve curls to the right, and for a negative end radius, the curve curls to the left.



The ***curve*** block covers both spiral and non-spiral transitions with a zero or non zero offset.

The *curve* block is:

<curve>

<curve_type> *curve_type_text*</curve_type>

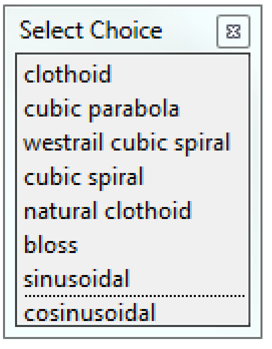
<leading> *leading_logical_text*</leading>

<xorigin> *xorigin_real*</xorigin>
<yorigin> *yorigin_real*</xorigin>
<radius> *radius_real*</radius>
<length> *length_real*</length>
<start> *start_length_real*</start>
<end> *end_length_real*</end>
<angle> *angle_decimal_degrees_real*</angle>
<offset> *offset_real*</offset>
<mvalue> *mvalue_real*</mvalue>

</curve

where

curve_type_text is the type of base transition.



For more information on the choices, see 35.3.15 Available Transition Types.

leading_logical_text is ***true*** if it is a **leading** base **transition** or *false* if is a trailing base transition.

*(**xorigin**, **yorigin**)* is the origin of the base transition. That is, where the radius is infinity.

radius_real is the radius at the end of the base transition. The radius is positive if the curve goes to the right when travelling in decreasing absolute radius.

length_real is the curve length and the end of the base transition and the radius is radius_real.

start_length_real is the curve length on the base transition where the start of the offset transition drops perpendicularly onto the base transition.

end_length_real is the curve length on the base transition where the end of the offset transition drops perpendicularly onto the base transition.

angle_decimal_degrees_real is the angle of the tangent of the base transition at the origin of the base transition. It is measured in decimal degrees in a counter clockwise direction from the positive x-axis.

offset_real is the perpendicular offset distance of the transition from the base transition. For a leading transition, a positive value offsets the base transition to the right and a negative value offsets it to the left as you travel in a forward direction.

mvalue_real *- if the transition is a cubic parabola then mvalue_real* is the mvalue for the cubic parabola, otherwise, *mvalue_real* is zero.

For example, for a string with four segments

```
<geometry_data>
```

```
  <arc>
    <radius>-222.77841769</radius>
    <major>0</major>
  </arc>
  <curve>
    <type>clothoid</type>
    <leading>false</leading>
    <xorigin>114.78632204</xorigin>
    <yorigin>22.22840069</yorigin>
    <radius>222.77841769</radius>
    <length>194.18990415</length>
    <start>50.95749554</start>
    <end>194.18990415</end>
     <angle>174.01773651</angle>
    <offset>0</offset>
    <mvalue>0</mvalue>
  </curve>
  <arc>
    <radius>-848.96871636</radius>
    <major>0</major>
  </arc>
  <straight/>
</geometry_data>
```

35.6.10.3 Colour

There can be one colour for the entire super string which is given by the **<colour>** keyword block in the ***string_header_block***, or the colour varies for each segment of the super string and is then specified in a ***<colour_data>*** block.

The order of the entries in the **<colour_data>** block must match the order of the segments in the super string. So there is exactly one entry for each segment.

If all the segment are the string colour, then simply omit the **<colour_data>** block.

For a super string with **n** vertices

<colour_data>

colour_text_for_segment_1

colour_text_for_segment_2

...

colour_text_segment_m

</colour_data>

where

colour_text_segment_i is the colour name or colour number for the i'th segment OR is **no_colour** when no special colour has been set for the segment and the string colour is then used for that segment. If the name includes spaces then it must be enclosed in **"**, and

m = n-1 for an open string or m = n for a closed string.

For example for a string with four segments

```
<colour_data>
    "off yellow" magenta no_colour no_colour
</colour_data>    <leading>false</leading>
```

35.6.10.4 String, Vertex and Segment Attributes

The super string can have attributes for the entire string (string attributes) but can also have attributes for each vertex (vertex attributes) and attributes for each segment (segment attributes).

See

35.6.10.4.1 String Attributes
35.6.10.4.2 Vertex Attributes
35.6.10.4.3 Segment Attributes

35.6.10.4.1 String Attributes

There can be attributes for the entire string. They are part of the String Header Block and are described in 35.6.3 String Header Block.

For example

```
<string_super>
  <name>Line 1</name>
  <chainage>0</chainage>
  <breakline>line</breakline>
  <colour>cyan</colour>
  <style>1</style>
  <attributes>
    <text>
    <name>Street</name>
    <value>Weemala Road</value>
    </text>
  </attributes>
  <time_created>2015-05-11T09:08:06Z</time_created>
  <time_updated>2015-05-11T11:59:29Z</time_updated>
           ...
```

35.6.10.4.2 Vertex Attributes

Each vertex can have one or more user defined attributes.

For a super string with **n** vertices

<vertex_attribute_data>

vertex_attributes_for_vertex_1_block

vertex_attributes_for_vertex_2_block

...

vertex_attributes_for_vertex_n_block

</vertex_attribute_data>

where

vertex_attributes_for_vertex_j_block is the *attribute_block* for vertex j. The attribute_block is defined in 35.4 Attributes.

For example, for a string with four vertices

```
<vertex_attribute_data>
  <attributes>
    <real>
      <name>Flow</name>
      <value>27.4</value>
    </real>
  </attributes>
  <attributes/>
  <attributes/>
  <attributes/>
</vertex_attribute_data>
```

35.6.10.4.3 Segment Attributes

Each segment can have one or more user defined attributes.

For a super string with **n** vertices

<segment_attribute_data>

segment_attributes_for_segment_1_block

segment_attributes_for_segment_2_block

...

segment_attributes_for_segment_m_block

</segment_attribute_data>

where

segment_attributes_for_segment_j_block is an *attribute_block* for segment j. The attribute_block is defined in 35.4 Attributes, and

m = n-1 for an open string or m = n for a closed string.

For example, for an open string with four vertices

```
<segment_attribute_data>
  <attributes>
    <real>
      <name>Material</name>
      <value>clay</value>
    </real>
  </attributes>
  <attributes/>
  <attributes/>
  <attributes/>
</segment_attribute_data>
```

35.6.10.5 Vertex Id's (Point Id's)

Each vertex can have a vertex id (point id).

This is not the number position of the vertex in the string but is a separate id which is usually different for every vertex in every string.

The ***vertex id*** can be alphanumeric and include spaces.

The definition is:

For a super string with **n** vertices

<point_data>

point_id_text_for_vertex_1

point_id_text_for_vertex_2

...

point_id_text_for_vertex_n

</point_data>

where

point_id_text_for_vertex_i is the point id of the i'th vertex.

m = n-1 for an open string or m = n for a closed string.

point_id_text_for_vertex_i is the actual text enclosed in **"**, even if the text does not include spaces. If the point id has not been set for a vertex, the value should be included as "".

For example "Point 1" or "Point2" or "".

If the ***point_data*** block does not exist then there are no Vertex Ids.

For example, for a string with 4 vertices

```
<point_data>
  "Point 1" "Point2" "" ""
</point_data>
```

35.6.10.6 Symbols at Vertices

There can be no symbols at all, or the same symbol for every vertex in the using the ***symbol_value*** block or the symbol varies for each vertex of the super string using the ***symbol_data*** block.

If a symbol does not have a colour, or there is no colour in the symbol definition, then it uses the string colour.

The definitions are:

<symbol_value>

symbol_properties_block

</symbol_value>

where

symbol_properties_block is the description for the symbol to be used at every vertex of the super string, and

OR

For a super string with **n** vertices

<symbol_data>

symbol_properties_for_vertex_1_block

symbol_properties_for_vertex_2_block

...

symbol_properties_for_vertex_n_block

</symbol_data>

where

symbol_properties_for_vertex_i_block is the description for the symbol at vertex i.

The format of ***symbol_properties_block*** and ***symbol_properties_for_vertex_i_block*** is:

<properties>

<style> *symbol_name_text* </style>

<colour> *symbol_colour_name_text* </colour>

<size> symbol_size_*real* </size>

<rotation> *angle_dec_deg_real*</rotation>

<offset_x> *symbol_offset_x_real* </offset_y>

<offset_y> *symbol_offset_y_real* </offset_y>

</properties>

where

symbol_name_text is the name of the symbol.

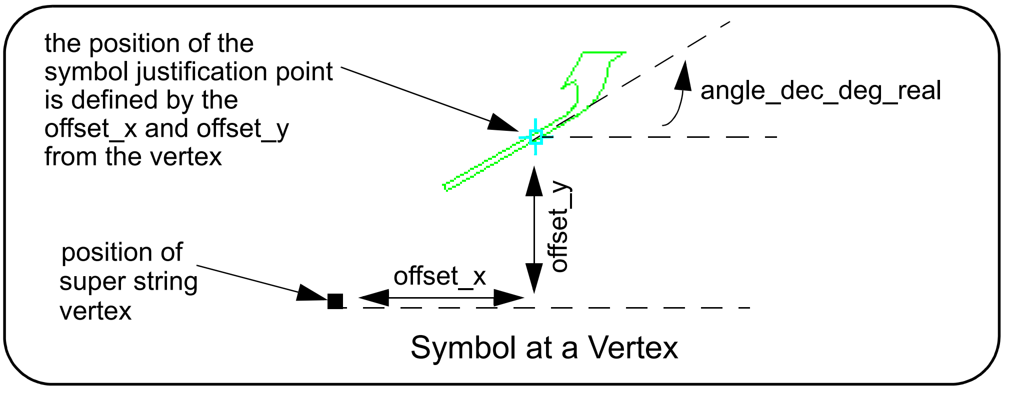
symbol_colour_name is the colour of the symbol is there is no colours in the symbol definition. If the ***colour*** block is missing and there is no colours in the symbol definition then the string colour is used. For the syntax of the ***colour*** block, see 35.3.2 Colour.

symbol_size_real is the size of the symbol in the units of the symbol.

angle_dec_deg_real is the angle of the symbol. The value is in decimal degrees and is measured in a counter clockwise direction from the positive x-axis.

offset_x_real is x distance to offset the symbol from the super string vertex.

offset_y_real is the y distance to offset the symbol from the super string.



35.6.10.7 Tinability

Tinability

For a *super string*, the concept of breakline has been extended to a property called **tinable** which can be set independently for each vertex and each segment of the super string.

If a vertex is tinable, then the vertex is used in triangulations. If the vertex is not tinable, then the vertex is ignored when triangulating.

If a segment is tinable, then the segment is used as a side of a triangle during triangulation. This may not be possible if there are *crossing* tinable segments.

Vertex tinability is given by the ***vertex_tinable_data*** block where for a string of n vertices,

<vertex_tinable_data>

tinable_flag_for_vertex_1

tinable_flag_for_vertex_2

...

tinable_flag_for_vertex_n

</vertex_tinable_data>

where

tinable_flag_for_vertex_i for the i'th vertex is 1 or t if the vertex is tinable, or 0 or f if the vertex is not tinable.

Segment tinability is given by the ***segment_tinable_data*** block where

<*segment_tinable_data*>

tinable_flag_for_segment_1

tinable_flag_for_segment_2

...

tinable_flag_for_segment_m

</*segment_tinable_data*>

where

tinable_flag_for_segment_i for the i'th segment is 1 or t if the segment is tinable, or 0 or f if the segment is not tinable, and

m = n-1 for an open string or m = n for a closed string.

For example, for a open string with four vertices

```
<vertex_tinable_data>
  t t f t
</vertex_tinable_data>
<segment_tinable_data>
  f t t
</segment_tinable_data>
```

Note that even if a segment is set to tinable, it can only be used in a triangulation if both its end vertices are also tinable.

35.6.10.8 Round or Box (Culvert) Pipes

All the super string segments can be round or all box (culvert). That is, some can't be round and others box.

There is also a justification that applies to all round pipes or box pipes in the string.

See

35.6.10.8.1 Pipe Diameters
35.6.10.8.2 Culvert Dimensions
35.6.10.8.3 Justification for Round or Culvert Pipes

35.6.10.8.1 Pipe Diameters

There can be one pipe diameter value for the entire super string using the ***pipe_value*** block **or** the pipe diameter varies for each segment of the super string using the ***pipe_data*** block.

The definitions are:

<pipe_value> *pipe_diameter_real* **</pipe_value>**

where *pipe_diameter_real* is the diameter for ***every*** segment of the string.

OR

For a super string with **n** vertices

<pipe_data>

<properties>

<diameter> pipe_diameter_for_segment_1 </diameter>

</properties>

<properties>

<diameter> pipe_diameter_for_segment_2 </diameter>

</properties>

...

<properties>

<diameter> pipe_diameter_for_segment_m </diameter>

</properties>

</pipe_data>

where

pipe_diameter_for_segment_i is the pipe diameter for the i'th segment, and

m = n-1 for an open string or m = n for a closed string.

35.6.10.8.2 Culvert Dimensions

There can be one culvert width and height for the entire super string using the ***culvert_value*** block **or** the culvert width and height vary for each segment of the super string using the ***culvert_data*** block.

The definitions are:

<culvert_value>

<width> *pipe_width_real* **</width>**

<height> *pipe_height_real* **</height>**

</culvert_value>

where *pipe_width_real* is the width and *pipe_height_real* is the height for every segment of the string.

OR

For a super string with **n** vertices

<culvert_data>

<properties>

<width> pipe_width_for_segment_1 </width>

<height> pipe_height_for_segment_1 </height>

</properties>

<properties>

<width> pipe_width_for_segment_2 </width>

<height> pipe_height_for_segment_2 </height>

</properties>

...

<properties>

<width> pipe_width_for_segment_m </width>

<height> pipe_height_for_segment_m </height>

</properties>

</culvert_data>

where

pipe_width_for_segment_i is the width and *pipe_height_for_segment_i* is the height for the i'th segment and

m = n-1 for an open string or m = n for a closed string.

35.6.10.8.3 Justification for Round or Culvert Pipes

There can be only one justification for all the round or culvert pipe segments in the super string.

The definition is:

<justify> *pipe_justification_text* **</justify>**

where

pipe_justification_text is the justification for the entire pipe and can have the values **centre**, **top**, **obvert**, **bottom** or **invert**.

If the ***justify*** block is missing then the round pipe or culvert pipe is **centre** justified.

35.6.10.9 Vertex Text and Vertex Annotation

See

[35.6.10.9.1 Vertex Text](#)
[35.6.10.9.2 Vertex Annotation](#)

35.6.10.9.1 Vertex Text

There can be not text at each vertex, the same piece of text for every vertex in the super string or a different text for each vertex of the super string.

Note: How the vertex text is drawn is specified by the vertex annotation. See 35.6.10.9.2 Vertex Annotation.

If there is a constant text value for each vertex in the string, then the text value is given by a **vertex_text_value** block:

<vertex_text_value> *text_value_text* **</vertex_text_value>**

where *text_value_text* is the constant text value for each vertex in the string.

For example, for a string of 5 vertices

```
<vertex_text_value>Constant text</vertex_text_value>
```

If there is a different text value for each vertex in the string, then the value of the text for each vertex is given in a ***vertex_text_data*** block.

<***vertex_text_data***>

<p> *text_value_for_vertex_1*</p>

<p> *text_value_for_vertex_2*</p>

...

<p> *text_value_for_vertex_n*</p>

</***vertex_text_data***>

where *text_value_for_vertex_i* is the vertex text for the i'th vertex.

For example, for a string of four vertices

```
<vertex_text_data>
  <p>First vertex</p>
  <p>Second vertex</p>
  <p/>
  <p/>
  </vertex_text_data>
```

35.6.10.9.2 Vertex Annotation

How the vertex text is drawn at each vertex is specified by the vertex annotation.

There can be no vertex annotations at all, or the same vertex annotation is used for every vertex in the string using the ***vertex_annotation_value*** block, or the vertex annotation varies for each vertex of the super string using the ***vertex_annotation_data*** block.

Note that in vertex annotations, the size of the text for all vertices must be either world size or all paper size or all screen size. That is, world size, paper size and screen size can not be mixed. The first one found is used for all vertices.

The definitions are:

<vertex_annotate_value>

vertex_annotation_information

</vertex_annotate_value>

where

vertex_annotation_information is the annotation to be used for drawing the text at every vertex of the super string. For the definition of *vertex_annotation_information* see 35.6.4.1 Vertex Annotation Information.

OR

For a super string with **n** vertices

<vertex_annotation_data>

annotation_for_vertex_1_block

annotation_for_vertex_2_block

...

annotation_for_vertex_n_block

</vertex_annotation_data>

where

annotation_for_vertex_i_block is the description for the vertex annotation for vertex i.

The format of the ***annotation_for_vertex_i_block*** is:

<properties>

vertex_*annotation_information*

<properties>

where

vertex_annotation_information is the annotation for drawing the text at the vertex. For the definition of *vertex_annotation_information* see 35.6.4.1 Vertex Annotation Information.

35.6.10.10 Segment Text and Segment Annotation

See

35.6.10.10.1 Segment Text
35.6.10.10.2 Segment Annotation

35.6.10.10.1 Segment Text

There can be no text on each segment, the same piece of text for every segment in the super string or a different text for each segment of the super string.

Note: How the segment text is drawn is specified by the segment annotation. See 35.6.10.10.2 Segment Annotation.

If there is a constant text value for each segment in the string, then the text value is given by a **segment_text_value** block:

<segment_text_value> *text_value_text* **</segment_text_value>**

where *text_value_text* is the constant text value for each segment in the string.

For example, for a string of 5 vertices

```
<segment_text_value>Constant text</segment_text_value>
```

If there is a different text value for each segment in the string, then the value of the text for each segment is given in a ***segment_text_data*** block.

<***segment_text_data***>

<p>text_value_for_segment_1</p>

<p>text_value_for_segment_2</p>

...

<p> text_value_for_segment_m</p>

<***/segment_text_data***>

where

text_value_for_segment_i is the segment text for the i'th segment, and

m = n-1 for an open string or m = n for a closed string.

For example, for a string of four segments

```
<segment_text_data>
  <p>First segment</p>
  <p>Second Segment&#xa;Two lines</p>
  <p>seg3</p>
  <p/>
</segment_text_data>
```

35.6.10.10.2 Segment Annotation

How the segment text is drawn at each segment is specified by the segment annotation.

There can be no segment annotations at all, or the same segment annotation is used for every segment in the string using the ***segment_annotation_value*** block, or the segment annotation varies for each segment of the super string using the ***segment_annotation_data*** block.

Note that in segment annotations, the size of the text for all segments must be either world size or all paper size or all screen size. That is, world size, paper size and screen size can not be mixed. The first one found is used for all segments.

The definitions are:

<segment_annotate_value>

segment_annotation_information

</segment_annotate_value>

where

segment_annotation_information is the annotation to be used for drawing the text at every segment of the super string. For the definition of *segment_annotation_information* see 35.6.4.2 Segment Annotation Information.

OR

For a super string with **n** vertices

<segment_annotation_data>

annotation_for_segment_1_block

annotation_for_segment_2_block

...

annotation_for_segment_m_block

</vertex_annotation_data>

where

annotation_for_segment_i_block is the description for the annotation at segment i, and.

m = n-1 for an open string or m = n   for a closed string.

The format of the ***annotation_for_segment_i_block*** is:

<properties>

segment_*annotation_information*

<properties>

where

segment_annotation_information is the annotation for drawing the text at the segment. For the definition of *segment_annotation_information* see 35.6.4.2 Segment Annotation Information.

Continue to the next section 35.6.11 Super Alignment String or return to 35.6 Elements Contained in Models or 35 12d XML File Format.

35.6.11 Super Alignment String

The 12d XML for the Super Alignment String has not yet been documented.

In an *alignment* string, only the intersection point method (IP's) could be used to construct the horizontal and vertical geometry. The IP definition is actually a *constructive* definition and the tangents points and segments between the tangent points (lines, arcs, transitions etc.) are calculated from the IP definition. For an alignment string, only the IP definitions are included in the 12d XML file.

For a ***super alignment***, the ***horizontal*** and ***vertical geometry*** are also defined separately and with construction definitions but the construction definition can be much more complex than just IP's. For example, an arc could be defined as being tangential to two offset elements, or constrained to go through a given point.

If the horizontal construction methods are consistent then the horizontal geometry can be solved, and the horizontal geometry expressed in terms of consecutive segments (lines, arcs, transitions) that are easily understood and drawn.

Similarly if the vertical construction methods are consistent then the vertical geometry can be solved, and the vertical geometry expressed in terms of consecutive segments (lines, arcs, parabolas) that are easily understood and drawn.

For the *super alignment* **both** the ***construction methods*** (the ***parts***) and the resulting ***vertices*** and ***segments*** (lines, arcs, transitions etc.) that make up the horizontal and vertical geometry (the ***data***) are written out to the 12d XML file.

For most applications such as uploading to survey data collectors or machine control devices, only the ***horizontal data*** and the ***vertical data*** are required, not the *construction* methods (i.e. the ***horizontal*** and ***vertical parts***). So when reading the 12d XML of a *super alignment*, only the ***horizontal*** and ***vertical data*** needs to be read in and the constructive methods (the ***horizontal*** and ***vertical parts***) can be skipped over.

Consequently only the ***horizontal data*** and the ***vertical data*** is full documented for the super alignment.

Notes

1. Just using the horizontal and vertical data is valid ***as long as the super alignment geometry is consistent*** (and solves) and the horizontal and vertical parts can be created.

   There are flags in the 12d XML of the super alignment to say that the horizontal and vertical geometry is consistent and solves.

2. Segments meeting at a common vertex do not have to be tangential although for most road and rail centre lines, they should be.

3. When ***12d Model*** reads in a 12d XML file and there is only ***horizontal_parts*** and no ***horizontal_data*** then if possible, ***12d Model*** generates the ***horizontal_data*** from the horizontal parts.

   This is very useful if you are creating a 12d XML file for a super alignment string that only uses HIP methods as it is fairly simple to create the ***horizontal_parts*** for such a string and that is fully documented in 35.6.11.2 Horizontal_Parts When Geometry is Defined by IP Method Only.

4. When ***12d Model*** reads in a 12d XML file and there is only ***vertical_parts*** and no ***vertical_data*** then if possible, ***12d Model*** generates the ***vertical_data*** from the vertical parts.

   This is very useful if you are creating a 12d XML file for a super alignment string that only uses VIP methods as it is fairly simple to create the ***vertical_parts*** for such a string and that is fully documented in 35.6.11.5 Vertical_parts When VG is Defined by IP Method Only.

The 1**2d XML** definition of the super alignment string is:

<string_super_alignment>

string_header_block

drawables_block

spiral_type_block

closed_block

valid_horizontal_block

valid_vertical_block

synch_vertical_block

label_style_block

horizontal_parts_block

horizontal_data_block

vertical_parts_block

vertical_data_block

geometry_modifiers_block

</string_super_alignment>

where

string_header_block

the common header block for each string. for the contents and the syntax, see 35.6.3 String Header Block.

drawables_block

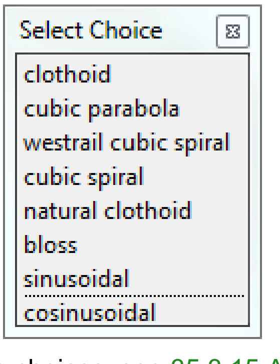
the drawables block contains information on how the super alignment is labelled.

This block is not documented.

spiral_type_block

<spiral_type> *transition_type_text* **</spiral_type>**

where *transition_type_text* is the default transition type use in the super alignment and is one of



For more information on the choices, see 35.3.15 Available Transition Types.

closed_block

<closed> closed_text_logical **</closed>**

where *closed_text_logical is* **true** if the super alignment string is closed and **false** if the super alignment string is open.

valid_horizontal_block

<valid_horizontal> *valid_horizontal_text_logical* **</valid_horizontal>**

where *valid_horizontal_text_logical is* **true** if the super alignment string horizontal geometry solves and **false** if the horizontal geometry does not solve.

If the horizontal geometry does not solve then the *horizontal_data* may be rubbish.

valid_vertical_block

<valid_vertical> *valid_vertical_text_logical* **</valid_vertical>**

where *valid_vertical_text_logical is* **true** if the super alignment string vertical geometry solves and **false** if the vertical geometry does not solve.

If the vertical geometry does not solve then the *vertical_data* may be rubbish.

synch_vertical_block

<synch_vertical> *synch_vertical_text_logical* **</synch_vertical>**

where *synch_vertical_text_logical is* **true** if the super alignment vertical geometry is to be synchronized to the horizontal geometry whenever the horizontal geometry is modified.

This is an internal ***12d Model*** flag.

label_style_block

<label_style> *label_style_text* **</label_style>**

where *label_style_text* is the name of the super alignment label style used for drawing the super alignment.

horizontal_parts_block

the *horizontal_parts* block contains the **methods** to construct the super alignment horizontal geometry. For example float (fillet) an arc of a certain radius between two given lines or create a transition (spiral or non-spiral transition) between a line and an arc.

The parts that make up the horizontal geometry are defined in chainage order from the start to the end of the super alignment.

If the horizontal construction methods are consistent, then they can be solved to form a plan string made up of lines, arcs and transitions and this is given in the ***horizontal_data*** block.

Because the construction methods can be very complex, the horizontal_parts block will only be documented for the case where all the horizontal parts are horizontal intersection points (HIPs) with an arc and leading and trailing transitions. See 35.6.11.2 Horizontal_Parts When Geometry is Defined by IP Method Only.

horizontal_data_block

the *horizontal_data* block contains the segments that define the horizontal geometry.

The *horizontal_data* block needs to be read in.

For the description of the *horizontal_data* block, see 35.6.11.1 Horizontal Data Block.

vertical_parts_block

the *vertical_parts* block contains the **methods** to construct the super alignment vertical geometry. For example float (fillet) an arc of a certain radius between two given lines.

The parts that make up the vertical geometry are defined in chainage order from the start to the end of the super alignment.

If the vertical construction methods are consistent, then they can be solved to form a string in (chainage, offset) space made up of lines, arcs and parabolas and this is given in the ***vertical_data*** block.

Because the construction methods can be very complex, the vertical_parts block will only be documented for the case where all the vertical parts are vertical intersection points (VIPs) with an arc or a parabola on the VIP. See 35.6.11.5 Vertical_parts When VG is Defined by IP Method Only.

vertical_data_block

the *vertical_data* block contains the segments that define the vertical geometry.

The *vertical_data* block needs to be read in.

For the description of the *vertical_data* block, see 35.6.11.3 Vertical Data Block.

geometry_modifiers_block

the *geometry_modifiers_parts* block contains extra construction information for the super alignment.

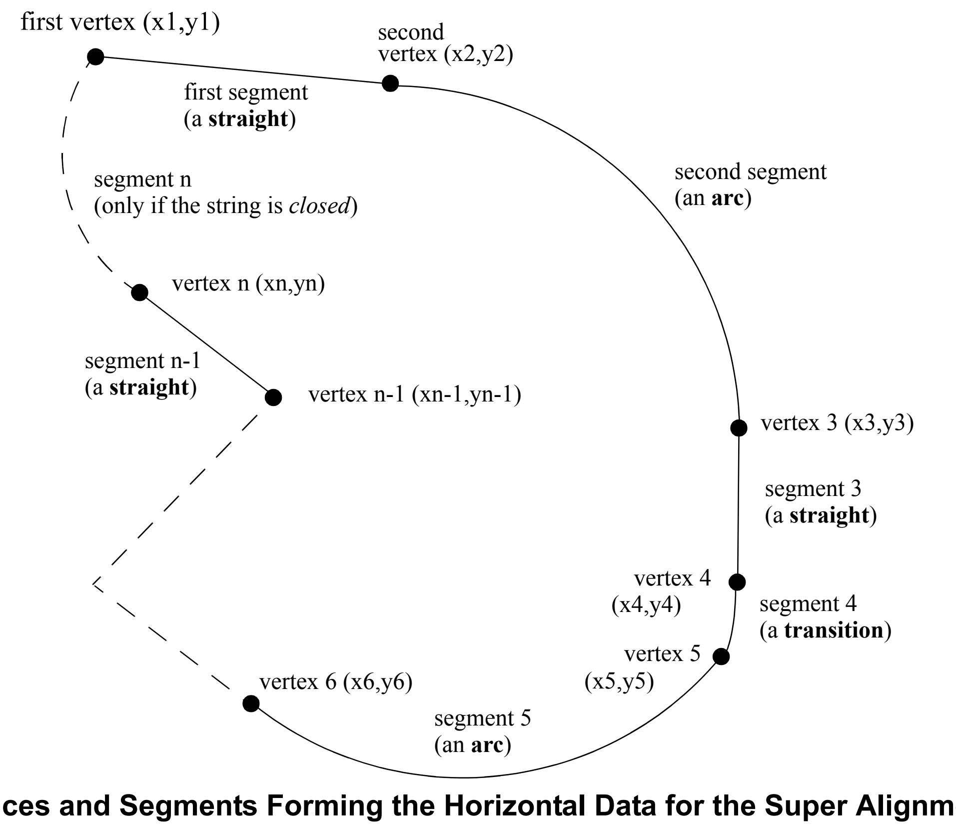
This block is not documented.

35.6.11.1 Horizontal Data Block

The *horizontal_data* block contains the ***solved*** horizontal geometry of the super alignment.

The *solved horizontal geometry* is made up of a series of (x,y) vertices given in a *data_2d* block followed by a *geometry_data* block specifying the geometry of the segments between adjacent vertices. Each segment can be a straight line, an arc, a transition or an offset transition.

If the solved horizontal geometry has **n** vertices, then there will be **n-1 segments** for an ***open*** super alignment or **n segments** if the super alignment is ***closed***.



Vertices and Segments Forming the Horizontal Data for the Super Alignment

The format of the *horizontal_data* block is the same as for the segments of a super string except that the data is only in 2D. Unlike a super string where there is just a z-value at each vertex, the third dimension of the super alignment is given by the *vertical_data* block (see 35.6.11.3 Vertical Data Block).

The definition of the *horizontal_data* block is:

<horizontal_data>

string_header_block

closed_block

interval_block

data_2d_block

geometry_data_block

blocks_of_info_1

blocks_of_info_2

...

blocks_of_info_n

</horizontal_data>

where

string_header_block

the common header block for each string. for the contents and the syntax, see 35.6.3 String Header Block. This provides information such as colour for the horizontal data.

interval_block

The interval_block for a super string has a *distance* (a chainage interval) and a *chord_to_arc_*real

where

the ***distance*** to temporarily introduce extra vertices into the string at the given chainage distance when the string is in a triangulation to form a tin.

chord_arc_real is a real number and is the chord to arc tolerance to use on any arcs in the horizontal data to temporarily insert vertices into the arc when the arc is included in a triangulation to form a tin.

For the syntax of interval_block, see 35.3.6 Interval.

data_2d_block

the *data_2d* block defines the (x,y) value of the vertices that makes up the horizontal data.

For the definition of the data_2d block, see 35.3.12 data_2d.

geometry_data_block

the segments of the horizontal data can be straights, arcs, transitions or offset transitions.

For the definition of the *geometry_data* block*, see* 35.6.10.2 Geometry of the Horizontal Segments

blocks_of_info

extra information for the vertices and/or segments such as colour, attributes, vertex text, vertex uids etc are defined in the same way as for super strings.

35.6.11.2 Horizontal_Parts When Geometry is Defined by IP Method Only

When the horizontal geometry is defined by IP methods only, then the ***horizontal_parts*** is fairly straight forward.

When ***12d Model*** reads in a 12d XML file and there is no ***horizontal_data*** section, then ***12d Model*** will calculate the ***horizontal_parts***. So you are writing a 12d XML with only IP methods for the horizontal geometry then simply leave out the ***horizontal_data*** section and ***12d Model*** will calculate it for you.

For a horizontal geometry is defined by IP methods only, the ***horizontal_parts*** definition is:

<horizontal_data>

info_for_HIP_1_block

info_for_HIP_2_block

...

info_for_HIP_n_block

</horizontal_data>

where ***info_for_HIP_i_block*** is the information about the successive HIPs in the super alignment and is one of:

(a) A horizontal intersection point (HIP) with no arc.

This is defined by:

<ip>

<id> *part_id_integer* **</id>**

time_created_block

time_updated_block

<x> *x_ip_coordinate_real* **</x>**

<y> *y_ip_coordinate_real* **</y>**

</ip>

where

part_id_integer is a number that is unique for each horizontal and vertical part and the value is a multiple of 100.

time_created_block

is the time the super tin was originally created, This is optional. For the syntax see 35.3.7 Time Created.

time_updated_block

is the last time the super tin was last modified, This is optional. For the syntax see 35.3.7 Time Created.

x_ip_coordinate_real is the x coordinates of the HIP.

y_ip_coordinate_real is the y coordinates of the HIP.

(b) A horizontal intersection point (HIP) with an arc of a given radius at the HIP.

This is defined by:

<arc>

<id> *part_id_integer* **</id>**

time_created_block

time_updated_block

<r> *arc_radius_real* **</r>**

<x> *x_ip_coordinate_real* **</x>**

<y> *y_ip_coordinate_real* **</y>**

</arc>

where

part_id_integer is a number that is unique for each horizontal and vertical part and the value is a multiple of 100.

time_created_block

is the time the super tin was originally created, This is optional. For the syntax see 35.3.7 Time Created.

time_updated_block

is the last time the super tin was last modified, This is optional. For the syntax see 35.3.7 Time Created.

arc_radius_real is the radius of the arc on the HIP.

x_ip_coordinate_real is the x coordinate of the HIP.

y_ip_coordinate_real is the y coordinate of the HIP.

(c) A horizontal intersection point (HIP) with an arc of a given length at the HIP

This is defined by:

<length>

<id> *part_id_integer* **</id>**

time_created_block

time_updated_block

<l> *arc_length_real* </l>

<x> *x_ip_coordinate_real* **</x>**

<y> *y_ip_coordinate_real* **</y>**

</length>

where

part_id_integer is a number that is unique for each horizontal and vertical part and the value is a multiple of 100.

time_created_block

is the time the super tin was originally created, This is optional. For the syntax see 35.3.7 Time Created.

time_updated_block

is the last time the super tin was last modified, This is optional. For the syntax see 35.3.7 Time Created.

arc_length_real is the length of the arc on the HIP.

x_ip_coordinate_real is the x coordinate of the HIP.

y_ip_coordinate_real is the y coordinate of the HIP.

(d) A horizontal intersection point (HIP) with an arc and transitions

This is defined by:

<spiral>

<id> *part_id_integer* </id>

time_created_block

time_updated_block

transition_type_block

<r> *arc_radius_real* **</r>**

<l1> *leading_transition_length_real* **</l1>**

<l2> *trailing_transition_length_real* **</l2>**

<x> *x_ip_coordinate_real* **</x>**

<y> *y_ip_coordinate_real* **</y>**

</spiral>

where

part_id_integer is a number that is unique for each horizontal and vertical part and the value is a multiple of 100.

time_created_block

is the time the super tin was originally created, This is optional. For the syntax see 35.3.7 Time Created.

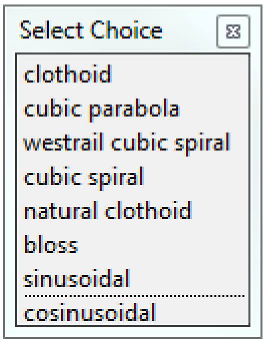
time_updated_block

is the last time the super tin was last modified, This is optional. For the syntax see 35.3.7 Time Created.

transition_type_block

<transition_type> *transition_type_text* **</transition_type>**

where *transition_type_text* is the default transition type use in the super alignment and is one of



This block is optional and if it is missing then the ***default transition type*** for the super alignment is used.

For more information on the choices, see 35.3.15 Available Transition Types.

arc_radius_real is the radius of the arc on the HIP.

leading_transition_length_real is the length of the leading transition on the HIP.

trailing_transition_length_real is the length of the trailing transition on the HIP.

x_ip_coordinate_real is the x coordinate of the HIP.

y_ip_coordinate_real is the y coordinate of the HIP.

Notes

1. A <length> block with *arc_length_real* equal to zero, or a <spiral> block with the *arc_radius_real*, *leading_transition_length_real* and *trailing_transition_length_real* all zero,

will also represent a HIP with no arcs or transitions on it.:

<length>

<id> *part_id_integer* **</id>**

time_created_block

time_updated_block

<l> *0* </l>

<x> *x_ip_coordinate_real* **</x>**

<y> *y_ip_coordinate_real* **</y>**

</length>

OR

<spiral>

<id> *part_id_integer* </id>

time_created_block

time_updated_block

transition_type_block

<r> *0* **</r>**

<l1> *0* **</l1>**

<l2> *0* **</l2>**

<x> *x_ip_coordinate_real* **</x>**

<y> *y_ip_coordinate_real* **</y>**

</spiral>

2. If the HIP is the first HIP or the last HIP then no arc or transitions will be drawn even if the relevant parameters are non zero.

As an example of horizontal_parts with only HIP methods:

```
<horizontal_parts>
  <ip>
    <id> 100 </id>
    <x> 42606.66161172 </x>
    <y> 37239.28824481 </y>
  </ip>

  <ip>
    <id> 200 </id>
    <x> 43134.36832349 </x>
    <y > 37330.26705997</y>
  </ip>

  <spiral>
    <id> 300 </id>
    <r> 50 </r>
    <l1> 30 </l1>
    <l2> 40 </l2>
    <x> 43336.6595 </x>
    <y> 37469.2563 </y>
  </spiral>

  <arc>
    <id> 400 </id>
    <r> 75 </r>
    <x> 43481.15324268 </x>
    <y> 37331.6431906 </y>
  </arc>

  <ip>
    <id> 500 </id>
    <x> 43627.02308964 </x>
    <y> 37544.94343852 </y>
  </ip>
</horizontal_parts>
```

1st HIP
HIP only

Unique Part id
incrementing by 100

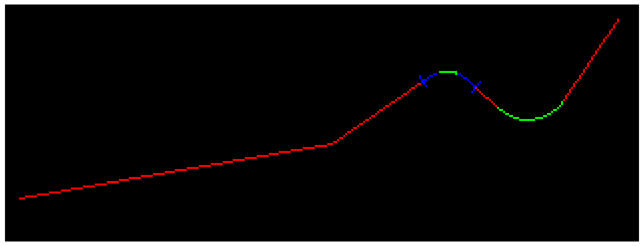
2nd HIP
HIP only

3rd HIP
HIP with arc and leading and trailing transitions

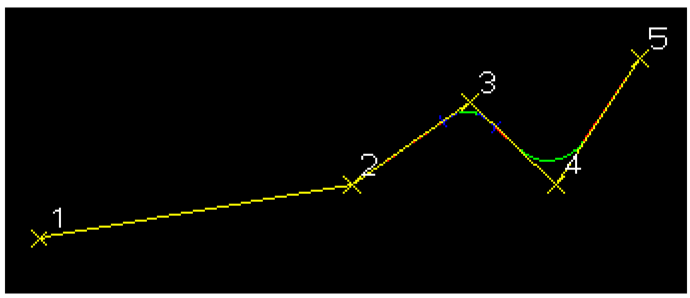
4th HIP
HIP with arc only

5th HIP
HIP only

Plan View of Super Alignment



Super Alignment Being Edited



Horizontal Parts with IP Methods Only

35.6.11.3 Vertical Data Block

The *vertical_data* block contains the ***solved*** vertical geometry of the super alignment.

The *solved vertical geometry* is made up of a series of (chainage,height) vertices given in a *data_2d* block followed by a *geometry_data* block specifying the geometry of the segments between adjacent vertices. The segment can be a straight line, a parabola or an arc.

Note that the ***chainage*** is the chainage of the horizontal geometry defined in the *horizontal_data* block (see 35.6.11.1 Horizontal Data Block).

If the vertical geometry has **n** vertices, then there will be **n-1 segments** for an ***open*** super alignment or **n segments** if the super alignment is ***closed***.

The format of the *vertical_data* block is the same as for the segments in a horizontal_data block except that the data is (chainage, height) rather than (x,y) and there is no transitions but a parabola instead.

The definition of the *vertical_data* block is:

<vertical_data>

string_header_block

closed_block

interval_block

data_2d_block

geometry_data_block

blocks_of_info_1

blocks_of_info_2

.**...**

blocks_of_info_n

</vertical_data>

where

string_header_block

the common header block for each string. for the contents and the syntax, see 35.6.3 String Header Block. This provides information such as colour for the vertical data.

interval_block

The interval_block for a super string has a *distance* (a chainage interval) and a *chord_to_arc*_real

where

the ***distance*** to temporarily introduce extra vertices into the string at the given chainage distance when the string is in a triangulation to form a tin.

chord_arc_real is a real number and is the chord to arc tolerance to use on any arcs in the vertical data to temporarily insert vertices into the arc when the arc is included in a triangulation to form a tin.

For the syntax of interval_block, see 35.3.6 Interval.

data_2d_block

the *data_2d* block defines the (chainage,height) value of the vertices that makes up the vertical data.

For the definition of the data_2d block, see 35.3.12 data_2d where x is chainage and y is height.

geometry_data_block

the segments of the vertical data can be straights, arcs or parabolas.

For the definition of the *geometry_data* block*, see* 35.6.11.4 Geometry of the Vertical Segments

blocks_of_info

extra information for the vertices and/or segments such as colour, attributes, vertex text, vertex uids etc are defined in the same way as for super strings.

35.6.11.4 Geometry of the Vertical Segments

If the segments are straight lines only then that is the default and no further information is required.

If the segments are only straight lines and arcs, then the ***radius_data*** and **major_data** blocks are used to define a ***radius*** and ***bulge_flag*** data for each segment of the super string. See 35.6.11.4.1 Only Straights and Arcs for Segments.

If any of the segments are parabolas then ***geometry_data*** must be used for each segment. ***geometry_data*** can represent a straight, arc, transition or offset transition. See 35.6.11.4.2 Straights, Arcs and Parabolas for Segments.

35.6.11.4.1 Only Straights and Arcs for Segments

If there are only straight and arc segments for the string, then for the **data_2d** it is possible to add a radius and major/minor arc flag for each segment of the super string using the ***radius_data*** and **major_data** blocks respectively. See 35.3.14 radius_data and major_data.

35.6.11.4.2 Straights, Arcs and Parabolas for Segments

When some of the segments are parabolas then the ***geometry_data*** block **must** be used the give the geometry for each segments.

When the vertical_data has **n** vertices, then the definition of the ***geometry_data*** block is

<geometry_data>

info_for_segment_1_block

info_for_segment_2_block

...

info_for_segment_m_block

</geometry_data>

where

info_for_segment_i_block is the information defining the i'th segment as either a straight, an arc or an parabola and m = n-1 for an open string or m = n for a closed string.

For the definition of *info_for_segment_i_block* see:

35.6.11.4.2.1 Straight

35.6.11.4.2.2 Arc

35.6.11.4.2.3 Parabola

35.6.11.4.2.1 Straight

No parameters are needed for defining a straight segment. The *straight* block is simply:

<straight> </straight>

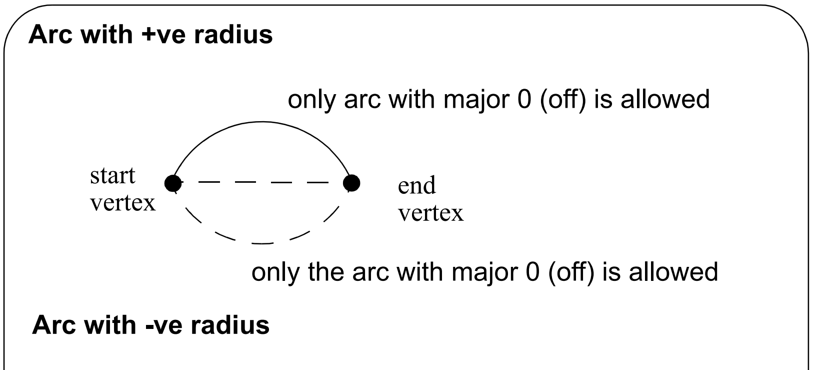
or simply

<straight/>

35.6.11.4.2.2 Arc

Since vertical geometry can't go backwards in chainage value, the majors arcs can not be used and hence there are only possibilities for an arc of a given radius placed between two vertices.

We use *positive* and *negative* radius to differentiate between the four possibilities.



Arcs with same absolute radius

The *arc* block is:

<arc>

<radius> *radius_for_segment*</radius>

<major> *major_flag_for_segment*</major>

</arc>

where

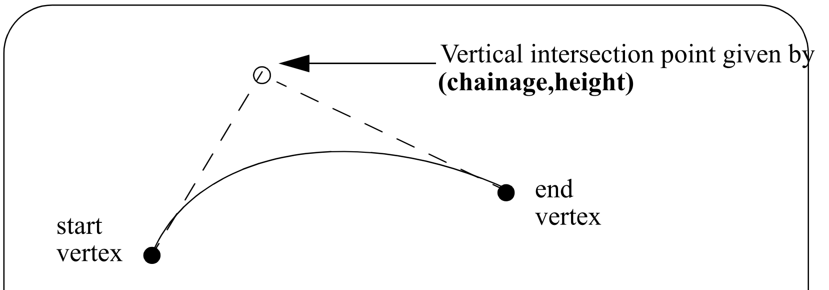
radius_for_segment is the radius for the segment where positive is above the line connecting the vertices.

major_flag_for_segment is ignored because only minor arcs are allowed.

35.6.11.4.2.3 Parabola

There can be a parabola between adjacent vertices. The parabola is defined by giving the coordinates of the vertical intersection point for the parabola

chainage chainage of the VIP of the parabola
height height of the VIP of the parabola



Example of a Parabola

The *parabola* block is:

<parabola>

<chainage> *vip_chainage_real* </chainage>

<height> *vip_height_real* </height>

</parabola

where

vip_chainage_real is the chainage of the VIP of the parabola

vip_height_real is the height of the VIP of the parabola

35.6.11.5 Vertical_parts When VG is Defined by IP Method Only

When the vertical geometry is defined by IP methods only, then the ***vertical_parts*** is fairly straight forward.

When ***12d Model*** reads in a 12d XML file and there is no ***vertical_data*** section, then ***12d Model*** will calculate the ***vertical_parts***. So if you are writing a 12d XML with only VIP methods for the vertical geometry then simply leave out the ***vertical_data*** section and ***12d Model*** will calculate it for you.

For a vertical geometry is defined by VIP methods only, the ***vertical_parts*** definition is:

<vertical_data>

info_for_VIP_1_block

info_for_VIP_2_block

...

info_for_VIP_n_block

</vertical_data>

where ***info_for_VIP_i_block*** is the information about the successive VIPs in the super alignment and is one of:

(a) A vertical intersection point (VIP) with no arc or parabola.

This is defined by:

<ip>

<id> *part_id_integer* **</id>**

time_created_block

time_updated_block

<x> *chainage_ip_coordinate_real* **</x>**

<y> *height_ip_coordinate_real* **</y>**

</ip>

where

part_id_integer is a number that is unique for each horizontal and vertical part and the value is a multiple of 100.

time_created_block

is the time the super tin was originally created, This is optional. For the syntax see 35.3.7 Time Created.

time_updated_block

is the last time the super tin was last modified, This is optional. For the syntax see 35.3.7 Time Created.

chainage_ip_coordinate_real is the chainage of the VIP.

height_ip_coordinate_real is the height of the VIP.

(b) A vertical intersection point (VIP) with an parabola of a given chainage length at the VIP

This is defined by:

<length>

<id> *part_id_integer* **</id>**

time_created_block

time_updated_block

<l> *parabola_chainage_length_real* </l>

<x> *chainage_ip_coordinate_real* **</x>**

<y> *height_ip_coordinate_real* **</y>**

</length>

where

part_id_integer is a number that is unique for each horizontal and vertical part and the value is a multiple of 100.

time_created_block

is the time the super tin was originally created, This is optional. For the syntax see 35.3.7 Time Created.

time_updated_block

is the last time the super tin was last modified, This is optional. For the syntax see 35.3.7 Time Created.

parabola_chainage_length_real is the chainage length of the parabola on the VIP.

chainage_ip_coordinate_real is the chainage of the VIP.

height_ip_coordinate_real is the height of the VIP.

(c) A vertical intersection point (VIP) with an parabola of a given k value at the VIP

This is defined by:

<kvalue>

<id> *part_id_integer* **</id>**

time_created_block

time_updated_block

<k> *parabola_k_value_real* </k>

<x> *chainage_ip_coordinate_real* **</x>**

<y> *height_ip_coordinate_real* **</y>**

</kvalue>

where

part_id_integer is a number that is unique for each horizontal and vertical part and the value is a multiple of 100.

time_created_block

is the time the super tin was originally created, This is optional. For the syntax see 35.3.7 Time Created.

time_updated_block

is the last time the super tin was last modified, This is optional. For the syntax see 35.3.7 Time Created.

parabola_k_value_real is the k value of the parabola on the VIP.

chainage_ip_coordinate_real is the chainage of the VIP.

height_ip_coordinate_real is the height of the VIP.

(d) A vertical intersection point (VIP) with an parabola of a given effective radius value at the VIP

This is defined by:

<radius>

<id> *part_id_integer* **</id>**

time_created_block

time_updated_block

<r> *parabola_effective_radius_value_real* </r>

<x> *chainage_ip_coordinate_real* **</x>**

<y> *height_ip_coordinate_real* **</y>**

</kvalue>

where

part_id_integer is a number that is unique for each horizontal and vertical part and the value is a multiple of 100.

time_created_block

is the time the super tin was originally created, This is optional. For the syntax see 35.3.7 Time Created.

time_updated_block

is the last time the super tin was last modified, This is optional. For the syntax see 35.3.7 Time Created.

parabola_effective_radius_value_real is the effective radius of the parabola on the VIP.

chainage_ip_coordinate_real is the chainage of the VIP.

height_ip_coordinate_real is the height of the VIP.

(e) A vertical intersection point (VIP) with an arc of a given radius at the VIP.

This is defined by:

<arc>

<id> *part_id_integer* **</id>**

time_created_block

time_updated_block

<r> *arc_radius_real* **</r>**

<x> *chainage_ip_coordinate_real* **</x>**

<y> *height_ip_coordinate_real* **</y>**

</arc>

where

part_id_integer is a number that is unique for each horizontal and vertical part and the value is a multiple of 100.

time_created_block

is the time the super tin was originally created, This is optional. For the syntax see 35.3.7 Time Created.

time_updated_block

is the last time the super tin was last modified, This is optional. For the syntax see 35.3.7 Time Created.

arc_radius_real is the radius of the arc on the VIP.

chainage_ip_coordinate_real is the chainage of the VIP.

height_ip_coordinate_real is the height of the VIP.

(f) A vertical intersection point (VIP) with an asymmetric parabola defined by the start and end chainage lengths at that VIP

This is defined by:

<asymmetric>

<id> *part_id_integer* **</id>**

time_created_block

time_updated_block

<l1> *parabola_start_chainage_length_real* **</l1>**

<l2> *parabola_end_chainage_length_real* **</l2>**

<x> *chainage_ip_coordinate_real* **</x>**

<y> *height_ip_coordinate_real* **</y>**

</asymmetric>

where

part_id_integer is a number that is unique for each horizontal and vertical part and the value is a multiple of 100.

time_created_block

is the time the super tin was originally created, This is optional. For the syntax see 35.3.7 Time Created.

time_updated_block

is the last time the super tin was last modified, This is optional. For the syntax see 35.3.7 Time Created.

parabola_start_chainage_length_real is the start chainage length of the asymmetric parabola on the VIP.

parabola_end_chainage_length_real is the end chainage length of the asymmetric parabola on the VIP.

chainage_ip_coordinate_real is the chainage of the VIP.

height_ip_coordinate_real is the height of the VIP.

Notes

1. A <length> block with *arc_length_real* equal to zero, or a <spiral> block with the *arc_radius_real*, *leading_transition_length_real* and *trailing_transition_length_real* all zero, will also represent a HIP with no arcs or transitions on it.:

   <length>

   <id> *part_id_integer* **</id>**

   time_created_block

   time_updated_block

   <l> *0* </l>

   <x> *x_ip_coordinate_real* **</x>**

   <y> *y_ip_coordinate_real* **</y>**

   </length>

   OR

2. If the VIP is the first VIP or the last VIP then no parabola or arc will be drawn even if the relevant parameters are non zero.

As an example of ***vertical_parts*** with only VIP methods:

```
<vertical_parts>
  <ip>
    <id> 600 </id>
    <x>-50.8459652 <x>
    ,<y> 59.79764161 <y>
  </ip>
  <kvalue>
    <id> 700 </id>
    <k> 1.25 </k>
    <x> 38.4627 </x>
    <y> 179.2126 </y>
  </kvalue>
    <length>
    <id> 800 </id>
    <l> 50 </l>
    <x> 172.61694837 </x>
    <y> 154.72967932 <x>
  </length>
  <asymmetric>
    <id> 900 </id>
    <l1> 25 </l1>
    <l2> 75 </l2>
    <x> 270.0182 </x>
    <y> 208.1493 </y>
  </asymmetric>
    <arc>
    <id> 1000 </id>
    <r> 1000 </r>
    <x> 424.2402 </x>
    <y>196.5637 </y>
  </arc>
  <radius>
    <id> 1100 </id>
    <r> 200 </r>
    <x> 526.7263 </x>
    <y> 201.5302 </y>
  </radius>
  <ip>
    <id> 1200 </id>
    <x> 637.69216273 </x>
    <y> 198.71894484 </y>
  </ip>
</vertical_parts>
```

1st VIP
VIP only

Unique Part id
incrementing by 100

2nd VIP
Parabola defined
by k value

3rd VIP
Parabola defined
by length

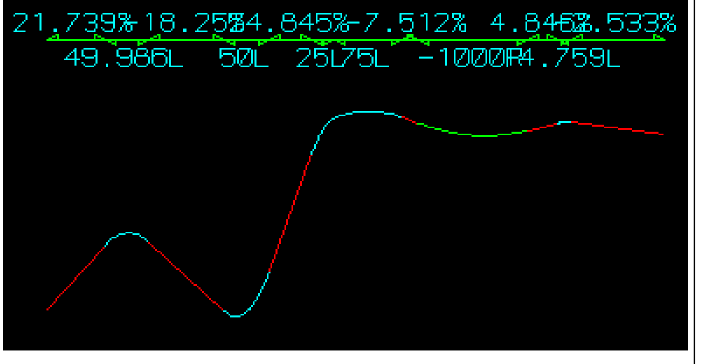
4th VIP
Asymmetric parabola defined
by two lengths

5th VIP
Arc with radius

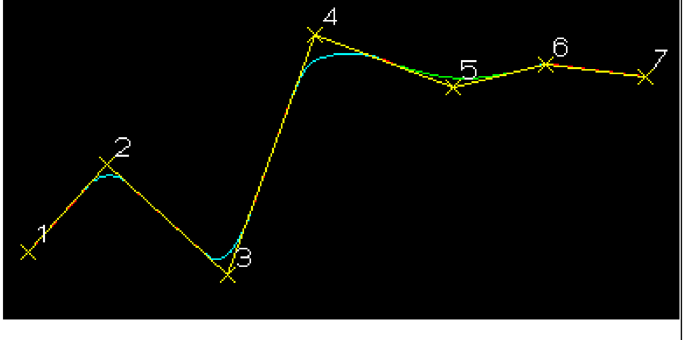
6th VIP
Parabola defined
by effective radius

7th VIP
VIP only

Section View of Super Alignment



Vertical Geometry Being Edited



Vertical Parts with IP Methods Only

Continue to the next section 35.6.12 Text String or return to 35.6 Elements Contained in Models
or 35 12d XML File Format.

35.6.12 Text String

The format for the **string_text** element is:

<string_text>

string_header_block

point_block

vertex_text_value_block

vertex_annotate_value_block

</string_arc>

where

string_header_block

the common header block for each string. for the contents and the syntax, see 35.6.3 String Header Block.

point_block

The format of the **point_block** is:

<point> *x_real y_real z_real* **</point>**

where

(*x_real,y_real,z_real*) is the vertex of the text.

vertex_text_value_block

The text for the text string.

The format of the **vertex_text_value_block** is:

<vertex_text_value**>** *characters_of_the_text* **</**vertex_text_value**>**

where

characters_of_the_text is the characters of the text with the except of some character that are special characters and are replace by something else.

For example **&** in the text is replaced **&amp** and a new line is given by **
**. See Characters "<", ">" and "&" and Escaping.

vertex_annotate_block

These are the setting for displaying text at a vertex.

The format of the **vertex_annotate_block** is:

<vertex_text_value**>**

vertex_annotation_information

</vertex_text_value**>**

where

vertex_annotation_information is the annotation to be used for drawing the text. For the definition of *vertex_annotation_information* see 35.6.4.1 Vertex Annotation Information.

For example

```
<string_text>
    <name>text</name>
    <chainage>0</chainage>
    <breakline>line</breakline>
    <colour>yellow</colour>
    <style>1</style>
    <time_created>28-Apr-2015 07:48:35</time_created>
    <time_updated>28-Apr-2015 07:49:33</time_updated>
```

```
    <point>1230.93054186 517.0328703 null</point>
    <vertex_text_value>First line&#xa;Second line</vertex_text_value>
    <vertex_annotate_value>
        <worldsize>20</worldsize>
        <textstyle>Arial</textstyle>
        <angle>45</angle>
        <x_factor>1</x_factor>
        <slant>0</slant>
        <offset>0</offset>
        <raise>0</raise>
        <text_colour>yellow</text_colour>
        <justify>middle-centre</justify>
    </vertex_annotate_value>
</string_text>
```

Continue to the next section 35.6.13 Trimesh or return to 35.6 Elements Contained in Models or 35 12d XML File Format.

35.6.13 Trimesh

A trimesh is a special type of primitive_3 object.

A trimesh is made up of 3D triangles and can be described by giving the list of ***m*** vertices in the trimesh and the three vertices that make up each of the ***n*** triangular faces.

The 12d XML definition of a trimesh is:

<primitive_3d>

string_header_block

trimesh_3d_block

</primitive_3d>

where

string_header_block

the common header block for each string. For the contents and the syntax, see 35.6.3 String Header Block.

The ***colour*** in the *string_header_block* is the base colour for the trimesh.

trimesh_block

The trimesh block gives the vertices of the trimesh and then the faces of the trimesh in terms of the vertex numbers.

<trimesh_3d>

<***vertices***>

<v> *x_value_1* *y_value_1* *z_value_1* **</v>**

<v> *x_value_2* *y_value_2* *z_value_2* **</v>**

...

<v> *x_value_n* *y_value_n* *z_value_n* **</v>**

</***vertices***>

<***faces***>

<f> *face_1_vertex_1* *face_1_vertex_2* *face_1_vertex_3* **</f>**

<f> *face_2_vertex_1* *face_2_vertex_2* *face_2_vertex_3* **</f>**

...

<f> *face_m_vertex_1* *face_m_vertex_2* *face_m_vertex_3* **</f>**

</***vertices***>

</trimesh_3d>

where

n is the number of vertices and (*x_value_i*, *y_value_i*, *z_value_i*) are the 3D coordinates of the ***i***'th vertex

m is the number of faces in the trimesh and *face_j_vertex_1*, *face_j_vertex_2*, *face_j_vertex_3* are the number of the vertices in the ***vertex*** block for the j'th face.

Return to 35.6 Elements Contained in Models or 35 12d XML File Format

36 12d Survey Guide

The information contained in this appendix outlines the general options, terminology, definitions and methods used by 12d Model for the purpose of the input, reduction and output of survey information. It is a general guide, with the appendices following relating to specific instruments and the interactions with 12d Model.

See

Guide to Survey Reduction in 12d Model
Guide to Survey Coding in 12d Model
Field Coding
Field Templates
Shape field coding
Traverse coding
Field Coding for Leica Instruments
Field Coding for Non Leica Instruments
Data Collector Definitions
The 12d Field File Format
Batch Typed Entry

Please continue to the next section 36.1 Guide to Survey Reduction in 12d Model.

36.1 Guide to Survey Reduction in 12d Model

The ***12d Model*** **survey** options are used to reduce electronically recorded survey information and produce 12d Model strings, a process called **survey reduction**.

In the most general case, there are three steps involved:

1. 12d Model downloads raw data from a data collector and stores it on the computer.
2. The raw data file is converted to a standard 12d Model raw field file format, normally with extension of ".fld". For simplicity, the standard 12d Model raw field file is called the "12d field file".
3. The 12d field file is read into a ***12d Model*** Survey Reduction function (12d Survey function) and reduced. If errors occur, the field data for 12d Survey function can be interactively and/or graphically edited. The reduction produces 12d Model super strings in one or many ***12d Model*** models.

In some circumstances, not all steps are necessary.

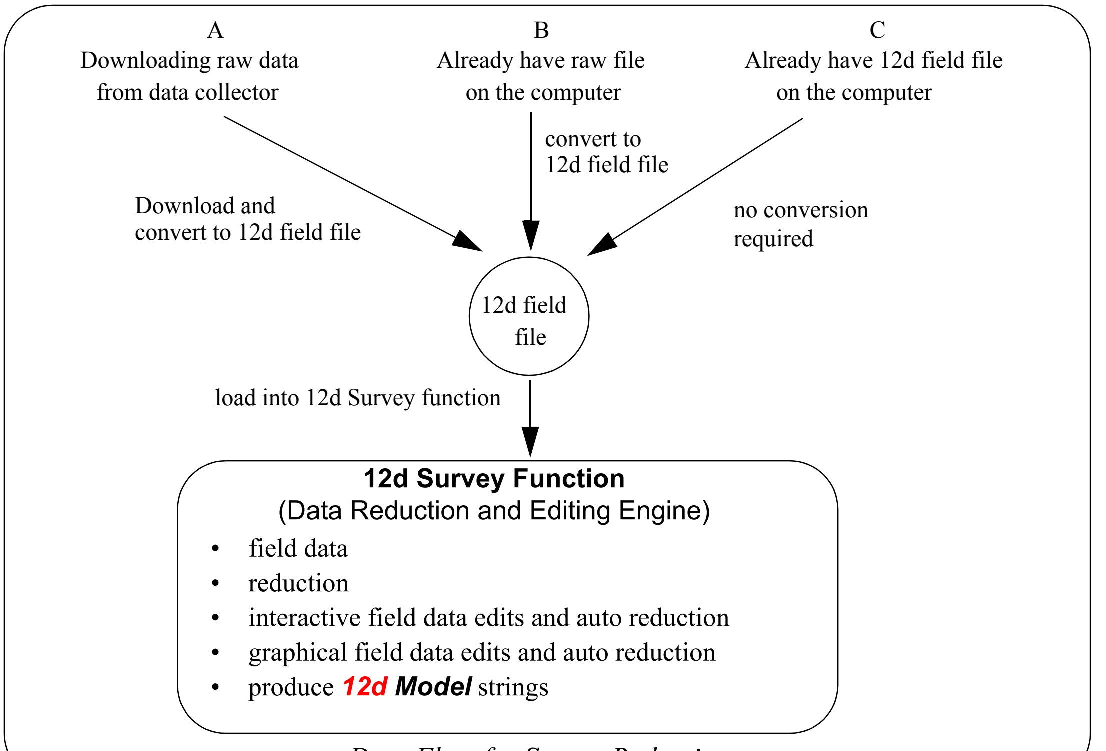
For example, Step 1 is not necessary when the raw data file is already on the computer (and hence doesn't need downloading).

This often occurs when data collectors have PCMCIA cards and PCMCIA card readers on the computer. Another example is when the raw data file has already been downloaded to the computer using another package such as HyperTerminal. Or maybe the raw data file has already been downloaded in a previous 12d Model session.

Steps 1 and 2 are not necessary when the data is already in the form of a 12d field file.

This occurs when other software packages, such as TP Setout, can produce a 12d field file. Or the 12d field file was created in a previous 12d Model session, maybe even on a different computer.

A data flow diagram for the survey reduction process is:



Data Flow for Survey Reduction

Please continue to the next section 36.2 Guide to Survey Coding in 12d Model.

36.2 Guide to Survey Coding in 12d Model

At first the whole process may seem confusing because there are a multitude of methods of setting up coding in the field. Why isn't there just one way of doing things?

Firstly, each brand of survey instrument has a totally different format for recording information. Unfortunately there is no industry standard. In fact, different instruments from the same survey instrument manufacturer can have different formats for recording data.

Secondly, many of our customers were already familiar with another survey package and if possible, wished to continue field coding in the same way. Where possible, ***12d Model*** has tried to accommodate this wish. Since ***12d Model*** has replaced a number of survey packages, this led to yet more possible configurations.

12d Model has also been developing its own preferred method of field coding for each instrument. It is not compulsory to use the 12d method but of course it is the only way of taking advantage of features that are not available in other coding systems. For example, defining and using field templates and recording user-defined attributes on points and segments.

So to make help guide you through survey coding section:

(a) There are field coding concepts used in ***12d Model*** that are common to all instruments. For example, feature codes and string numbers, offset, close strings etc.

These are described in 36.3 Field Coding

(b) There are template field coding concepts used in ***12d Model*** that are common to all instruments. These are described in 36.4 Field Templates

(c) There are shape field coding concepts used in ***12d Model*** that are common to all instruments. These are described in 36.5 Shape field coding

(d) There are traverse field coding concepts used in ***12d Model*** that are common to all instruments. These are described in 36.6 Traverse coding

(e) For leica instruments, the coding methods are different from most other types. These are described in 36.7 Field Coding for Leica Instruments

(f) For non-leica instruments, the concepts used in coding are similar for each type. These are described in 36.8 Field Coding for Non Leica Instruments

36.3 Field Coding

EDM equipment is used to make readings of points in the field.

However, rather than just collecting points, it is usually desirable to add extra information by coding the readings in a way that can be interpreted during the data reduction process and produce more valuable information.

Unfortunately, there is no industry standard for this extra coding and it is software specific.

In 12d Model, the extra information is included in one or both of

- a *feature code* and *string number*
- extra commands called *field codes*

How the *feature codes*, *string number* and *field codes* are added in the field depends on the data collector being used and the coding convention set up by the user in ***12d Model***.

For example, some surveyors like to enter the *feature code* before the *string number*, others like to enter the *feature code* after the *string number*. Other surveyors don't use string numbers at all but prefer to use a *New String* command to start new strings (mainly ex SDR Map users).

In ***12d Model***, a particular *field coding convention* is defined by the user and stored with a ***unique*** name as a *data collector definition* in the file survey.4d. New data collector definitions can be created and existing ones modified using the *Survey Data Collectors* section of *Project=>Browse*.

12d Model converts all the raw data files from the different survey instruments and data recorders with different coding conventions, into the one 12d Model standard field file format before being loaded into a ***12d Model*** Survey Reduction function and reduced. The 12d Model standard field file will simply be referred to as the 12d field file or just the field file.

The use of many of the commands allowed in the 12d field file will now be described. The complete definition of all the 12d field file commands will be described later.

The coding methodology for specific data recorders will be described in separate Sections.

Please continue to the next section 36.3.1 Stringing in the Field.

36.3.1 Stringing in the Field

In the coding convention, it is possible to specify that

(a) *feature codes* and *string numbers* are used
or
(b) just feature codes.

Case (a) ***feature codes*** **and** ***string numbers***

If *feature codes* and *string numbers* are entered with measurements in the field, a coding methodology is used so that strings are automatically created during the reduction process.

To allow this stringing, the **feature code** and **string number** are interpreted in the following manner:

During reduction, 12d Model connects measurement points with the **same** *feature code* **and** *string number* in the order they are measured in. That is, the feature code **and** string number determines which points are joined together to form the vertices of a super string. At the end of the reduction, the *string number* is dropped and just the *feature code* remains as the *name* for the super string.

Hence the *feature code* and *string number* combination allows any number of different super strings with the same name (feature code) to be produced.

If the *string number* is zero, then the point-line type of the super string is set to *point*. If the *string number* is non-zero, the point-line type of the super string is set to *line*.

Finally, during reduction, the *feature code* can be used as the key to a Map File to specify the name, model, colour, point-line type, linestyle, tinability and other details for the super string.

Note that the measurements of points with different feature codes and string numbers can be intertwined. That is, *not* all the points in one super string need to be measured before the points in a different super string.

Hence at the end of the reduction, 12d Model super strings are created for each unique *feature code* and *string number* combination in the input data.

Note - if the *string number* is blank, the *string number* defaults to 0

Case (b) just ***feature codes***

If just *feature codes* are used then a *New String* command is used to start a new super string rather than giving a new string number.

During reduction, 12d Model connects measurement points with the **same** feature code in the order they are measured in until a *New String* command is found. That is, just the feature code determines which points are joined together to form a super string and the *New String* command defines when a new super string begins.

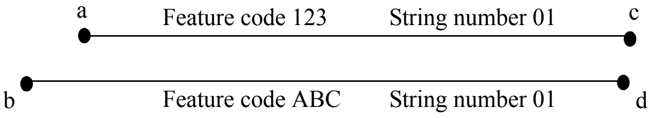
At the end of the reduction, the *feature code* remains as the *name* for the super string.

Also during reduction, the *feature code* can be used as the key to a Map File to specify the name, model, colour, point-line type, linestyle, tinability and other details for the super string.

Note that the measurements of points with different feature codes can be intertwined. That is, *not* all the points in one super string need to be measured before the points in a different super string.

An Example of Coding to String Points Together

Two super strings are to be created, one joining points **a** and **c**, the other joining points **b** and **d**.



Points may be measured and assigned *feature codes* and *string numbers* as follows:

Measurement to point	Feature code	String Number
a	123	01
b	ABC	01
c	123	01
d	ABC	01

Alternatively, the points could have been measured in the order a, c, b, d as long as the correct *feature codes* and *string numbers* were entered.

Measurement to point	Feature code	String Number
a	123	01
c	123	01
b	ABC	01
d	ABC	01

Because the *string numbers* are non-zero, the default *point-line* type for both super strings is *line*.

Please continue to the next section 36.3.2 Offsets.

36.3.2 Offsets

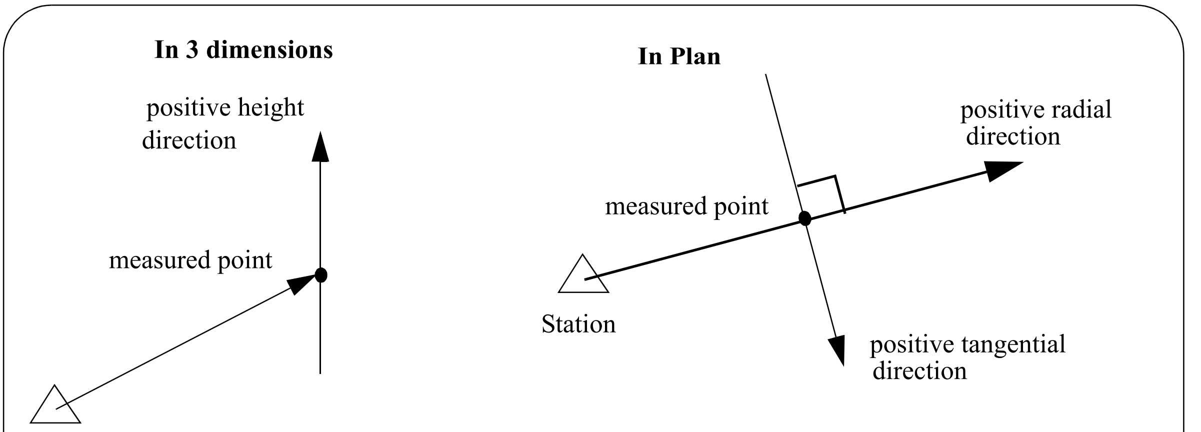
It is not always possible to measure a point directly but it may be possible to measure a point nearby and then measure an offset to adjust the measured point by and so produce the co-ordinates of the required point.

The three offsets that are allowed in the 12d field file are:

Height - The *height offset* adjusts the height of a non-null point. A positive offset adds to the height and a negative offset reduces the height.

Radial - The *radial offset* adjusts the position of the specified point by a *plan* distance from the specified points original position, along the plan line joining the current station to the specified point. A positive offset is away from the station and a negative offset is toward the station.

Tangential - The *tangential offset* adjusts the position of the specified point by a *plan* distance from the specified points original position, at rights angles to the plan line joining the current station to the specified point. A positive offset is to the right (looking from the station to the point) and a negative offset is to the left.



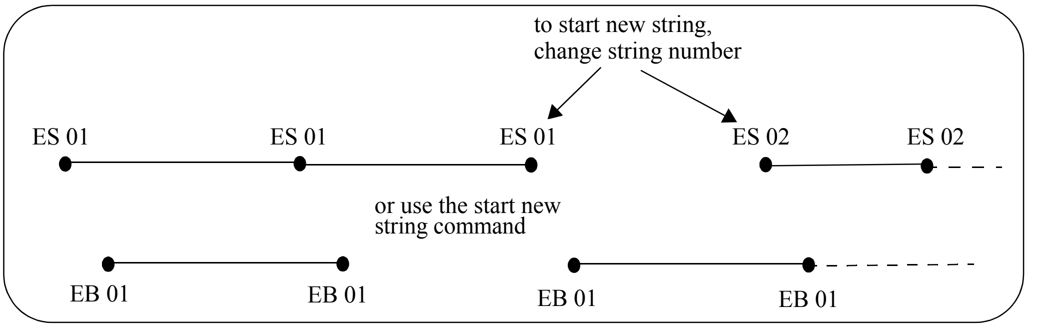
Offset Directions

Please continue to the next section 36.3.3 Start New String.

36.3.3 Start New String

A new string is automatically started whenever a different string number is used. However there is also a ***start new string*** command which begins a new super string even if the string number is the same as the string number for previous points.

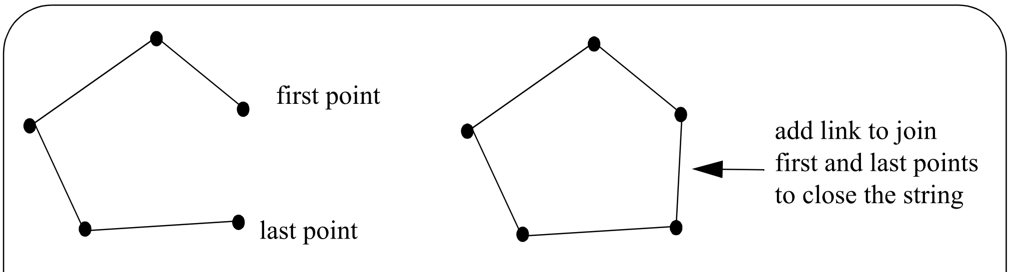
The *New String* command is particularly useful for correcting the field error of forgetting to change the string number.



Please continue to the next section 36.3.4 Close String.

36.3.4 Close String

The close string command closes a super string by joining the first and last points of the super string. If a super string is already closed, then the close has no effect. The string closed command can be given at the recording of any point of the string, and the entire string is closed.



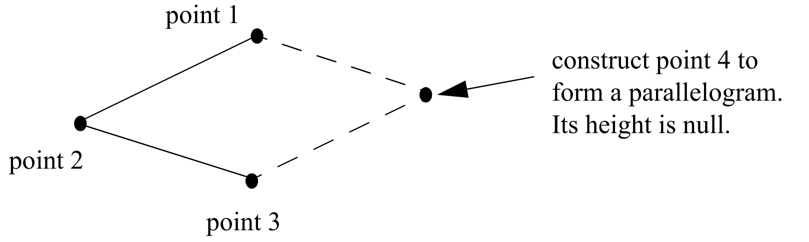
Close String

Please continue to the next section 36.3.5 Rectangle.

36.3.5 Rectangle

The rectangle command acts on last three points of a string and adds a new point after the last point to form a parallelogram (squashed rectangle). The string is then closed.

The height of the added point is set to null.



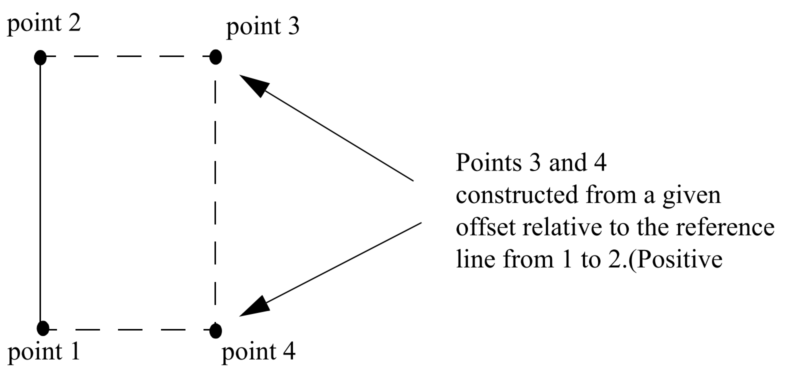
Form Rectangle

Please continue to the next section 36.3.6 Rectangle by 2 Points.

36.3.6 Rectangle by 2 Points

The rectangle by 2 pts command acts on last two points of a string and adds two new points at a given offset after the last point to form a rectangle. The string is then closed.

The height of the added points are set to null.



Form Rectangle By 2 Points

Please continue to the next section 36.3.7 Feature.

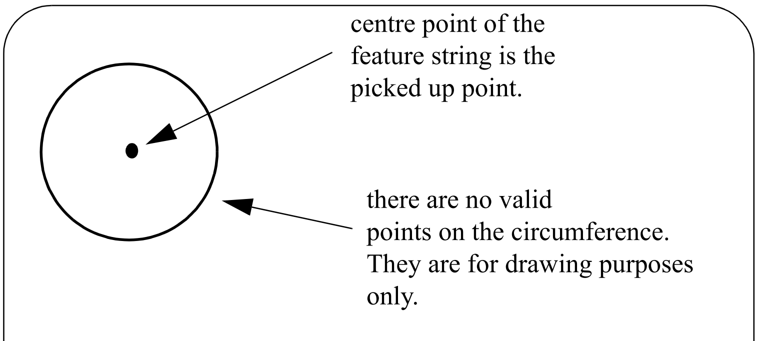
36.3.7 Feature

A *feature string* is a circle with a z-value at the centre but only null values on the circumference of the circle.

The feature commands creates a *feature string* with the picked up point as its centre and the radius/diameter being set by the feature command.

If a feature string is given a ***world*** line style, then the style is centred on the centre point of the feature string and scaled up to the radius of the feature string.

If a feature string is given a ***screen*** or ***paper*** line style, then the style is wrapped around the circumference of the feature string.



Feature String

Please continue to the next section 36.3.8 Joining Strings.

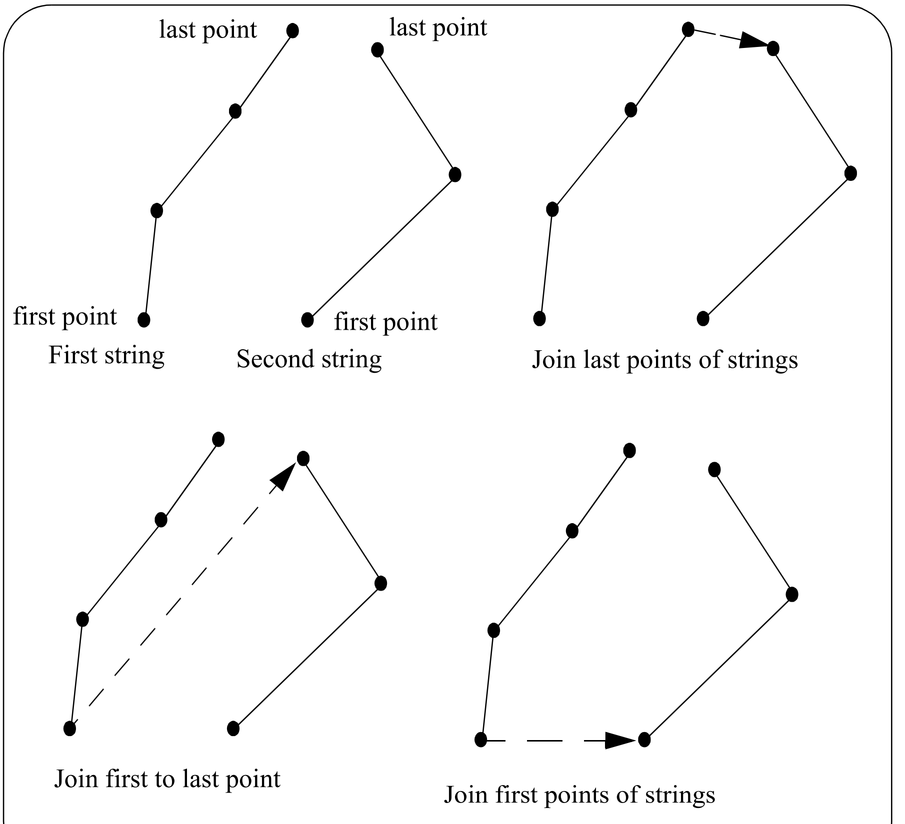
36.3.8 Joining Strings

There are three commands for joining two strings together.

Join last points of strings - the last point of the first string is joined to the last point of the second string. The direction of the final string is along the forward direction of the first string, across to the end of the second string and then in the reversed direction of the second string.

Join first to last points of strings - the first point of the first string is joined to the last point of the second string. The direction of the final string is in the reverse direction of the first string, across to the start of the second string and then in the forward direction of the second string.

Join first points of strings - the first point of the first string is joined to the first point of the second string. The direction of the final string is in the reverse direction of the first string, across to the start of the second string and then in the forward direction of the second string.



Join Strings

Please continue to the next section 36.3.9 Arcs Through Points.

36.3.9 Arcs Through Points

There are a number of commands to fit arcs through sequences of three or more points. Note that this is an **arc** in **plan**, with different z-values at each of the three points. The z-values are linearly interpolated around the arc between the points.

Hence it is a **helix** and **not** a circle in the plane containing the three points. Note that an 3d-arc in a plane not parallel to the x-y plane does **not** project onto an arc in the x-y plane.

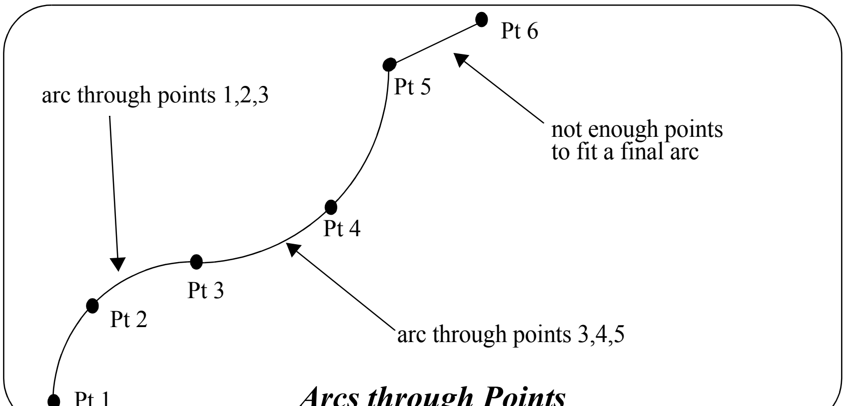
There are arc commands to

(a) fit an arc through the next three points

(b) fit an arc through the previous three points.

(c) fit arcs to sets of three points until stopped.

The first two cases need no explanation but in the final case, a arc is fitted to the first three points, and then another arc to points 3, 4 and 5 and then an arc through 5,6 and 7 and so on. If at any stage there is only one point left, then no arc can be fitted and a straight line is drawn to the final point.



Arcs through Points

Please continue to the next section 36.4 Field Templates.

36.4 Field Templates

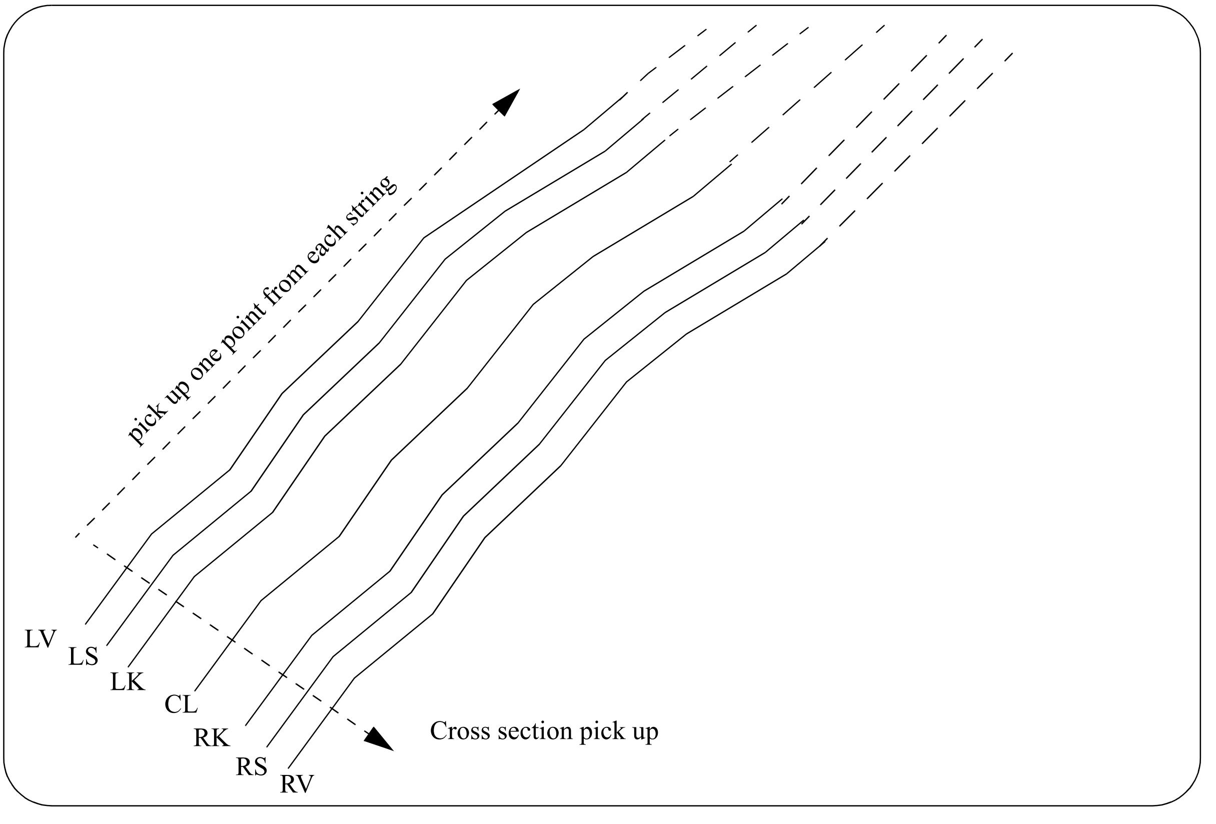
If a series of points are being picked up along the one string then the *feature code* and *string number* only need to be entered once since the default is for a measurement to use the last *feature code* and *string number* if no new ones are given.

However it is often much more efficient to pick up one point from a number of strings before moving onto the next point of each string (this is called a cross section pick up).

For example when picking up a road, it would be preferable to pick up the points for a section across the road and then move onto the next section rather than picking up all of one string at a time.

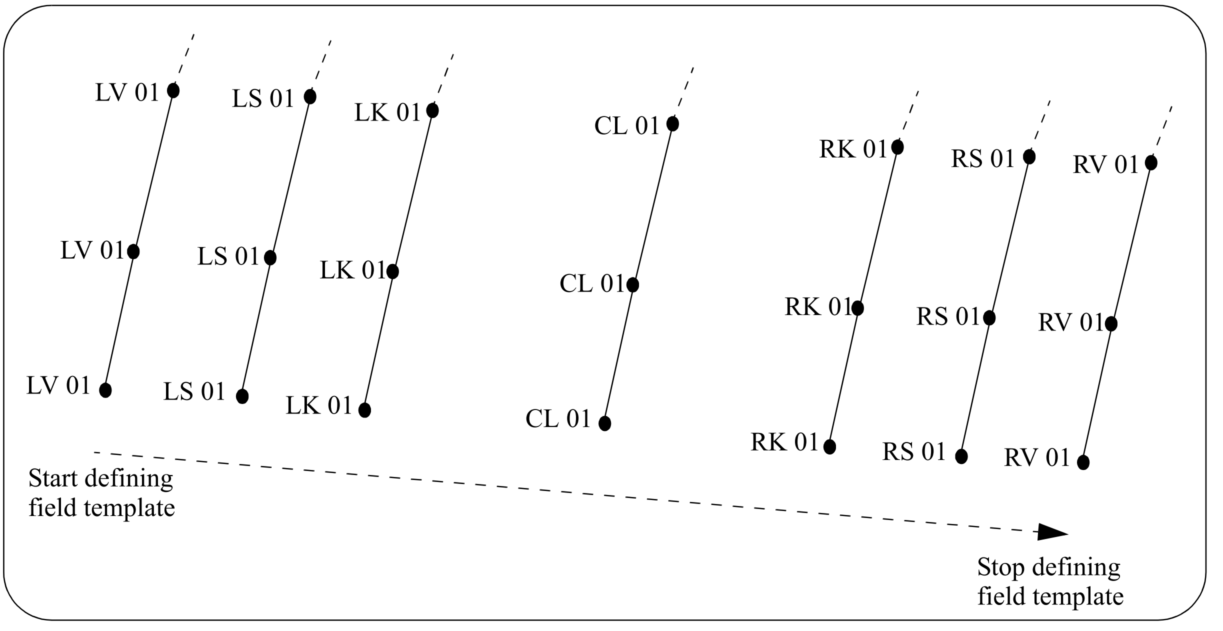
In the diagram below, this means picking up one point from each of the strings LV, LS, LK, CL, RK, RS, RV and then moving onto the next cross section rather than picking up all of LV and then all of LS and so on.

Normally if each measurement is from a different string, then the feature code and string number would need to be re-entered with each measurement which is a very time consuming process. To simplify the coding for section pick up, ***12d* Model** uses ***field templates***.



Basically, a ***12d* Model** *field template* consists of defining a sequence of *feature codes* and s*tring numbers* pairs for the field template. The field template can be given a unique name or have no name at all.

When a field template is used, measurements are taken without entering a *feature code* and *string number* and the *feature code* and *string number* for the measurement come from the *field template definition*.



For example, a field template could be defined as the sequence:

LV 01, LS 01, LK 01, CL 01, RK 01, RS 01, RV 01

When the field template is used, measurements are taken without giving a feature code or string number and the measurements will be sequentially given the codes LV 01, LS 01, LK 01 etc.

To define a ***12d Model*** *field template*, there is a command to *start recording* the field template.

The *feature code*s and *string numbers* for the next series of measurements until the *stop recording* command is given, are stored as the field template. There are also commands to *insert* and *delete* a point in the template when it is being used for picking up points.

When a field template is used, the feature code and string number from the field template can be used:

(a) in the same order as the codes are defined in the field template (forward direction).

(b) in the opposite order to how the codes are defined in the field template (reverse direction).

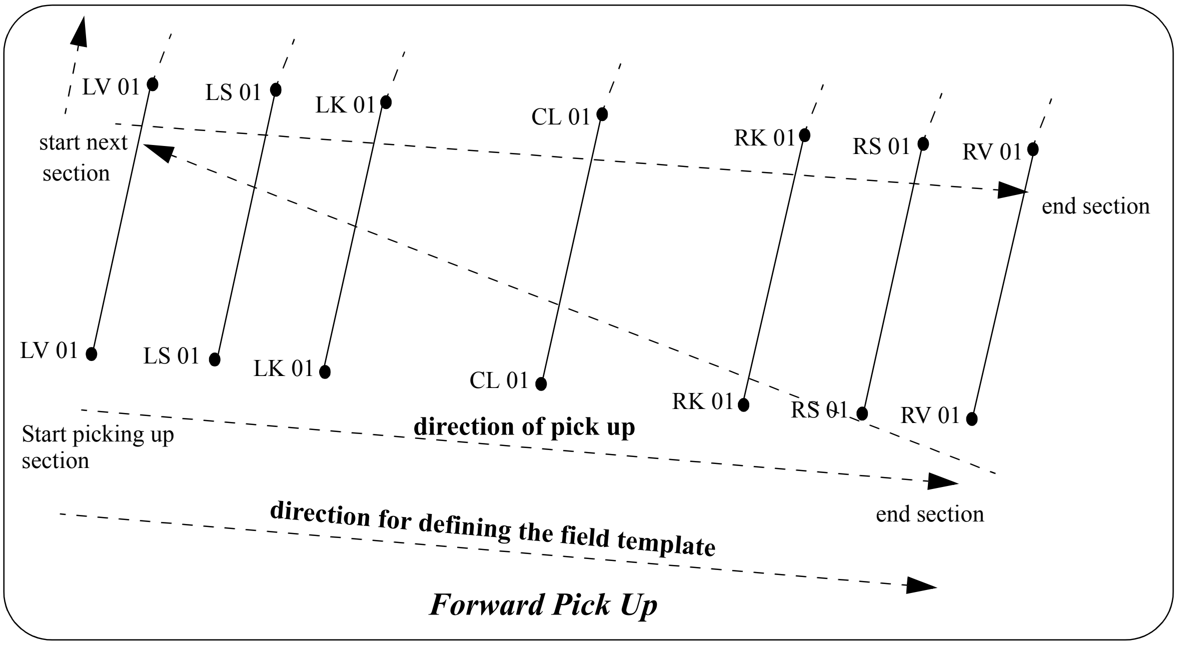
(c) in an alternating same and opposite order that the codes are defined in field template (zig-zag)

These three modes of usage of a field template will be described in the following sections.

Please continue to the next section 36.4.1 Forward Direction.

36.4.1 Forward Direction

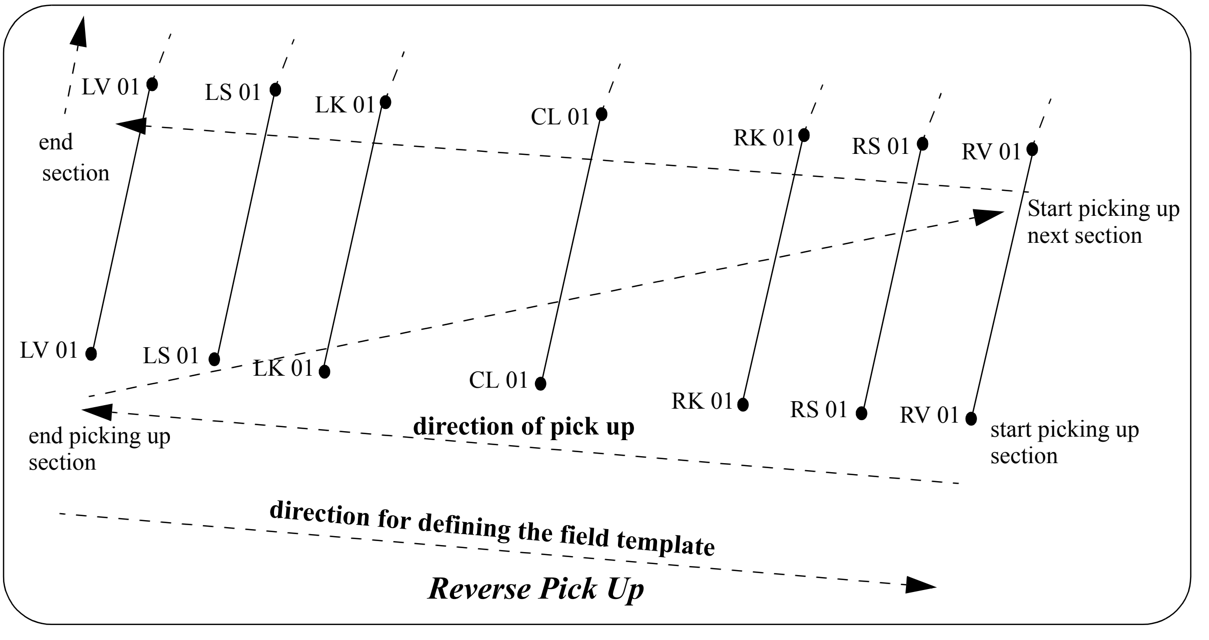
If the field template is used in the *forward* direction, then the feature codes and string numbers are used in the same order that they were defined to be in the field template. Once the end of the field template is reached, the feature codes and string numbers re-start at the beginning of the field template.



Please continue to the next section 36.4.2 Reverse Direction.

36.4.2 Reverse Direction

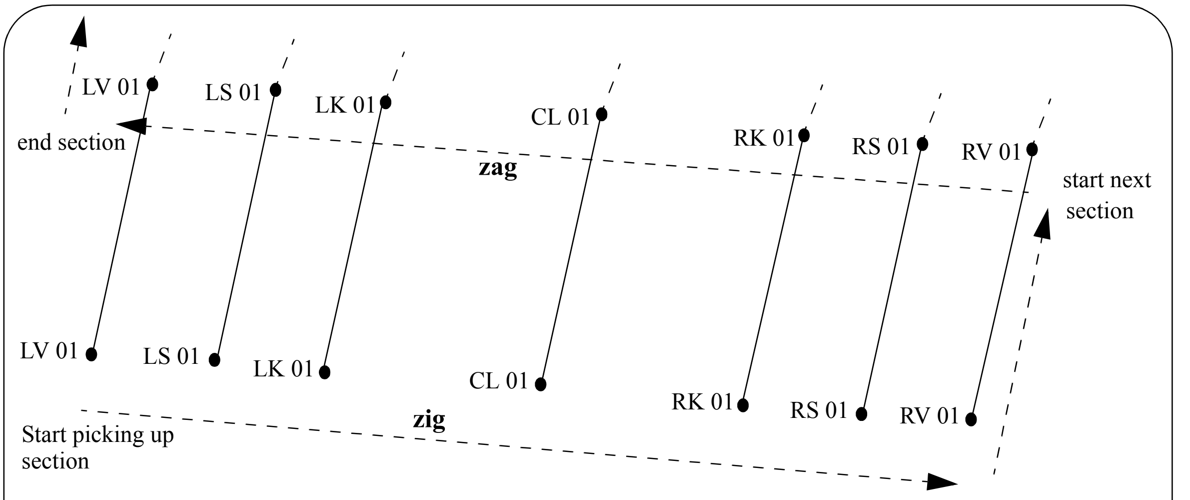
If the field template is used in the *reverse* direction, then the feature codes and string numbers are used in the reverse order to what they were defined to be in the field template. That is the feature codes and string numbers start at the *end* of the field template definition and are used in the reverse order. Once the beginning of the field template is reached, the feature codes and string numbers re-start at the end of the field template and are used in the reverse order.



Please continue to the next section 36.4.3 Zig-Zag.

36.4.3 Zig-Zag

When picking up a road in sections, it is often quickest to pick up the first section going from one side of the road to the other side, and then move onto to the next section point on the other side of the road and pick up points coming back across the road. Hence the points for the second section are in the reverse order to those in the first section. This process is known as zig-zagging.



Zig-Zag Pick Up

This situation can be covered in two ways. A field template could be defined containing all the points for two sections and the field template used in the forward (or reverse) direction. For example, the field template to be used in the forward mode could be defined as:

LV 01, LS 01 LK 01, CL 01, RK 01, RS 01, RV 01, RV 01, RS 01, RK 01, CL 01, LK 01, LS 01, LV 01

However, in ***12d Model*** it is only necessary to define the *one* section

LV 01, LS 01 LK 01, CL 01, RK 01, RS 01, RV 01

and when the field template is used, it is specified that it is being used as a *zig-zag* field template starting on either the *zig* (the forward direction of the field template) or the *zag* (the reverse direction of the field template).

Once a *zig* is completed, ***12d Model*** automatically uses the reverse order of the field template and hence produces a *zag*. Similar, once a zag is completed, ***12d Model*** uses the forward order of the field template and produces a zig.

Thus a zag automatically follows a zig and a zig follows a zag.

Hence if a field template is used in the zig-zag mode, it can be used as either:

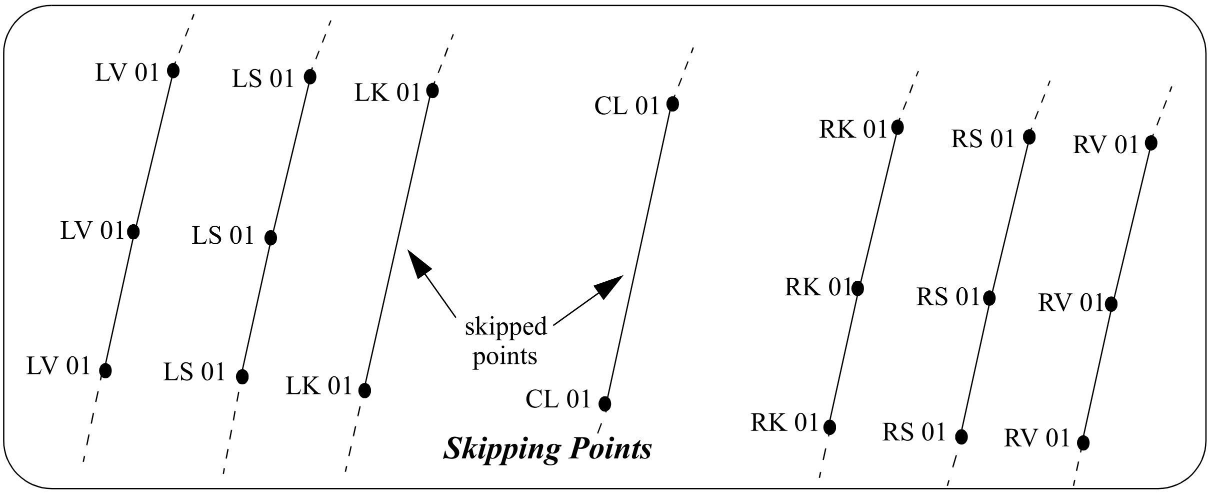
(a) a zig-zag field template starting on the zig

(b) a zig-zag field template starting on the zag.

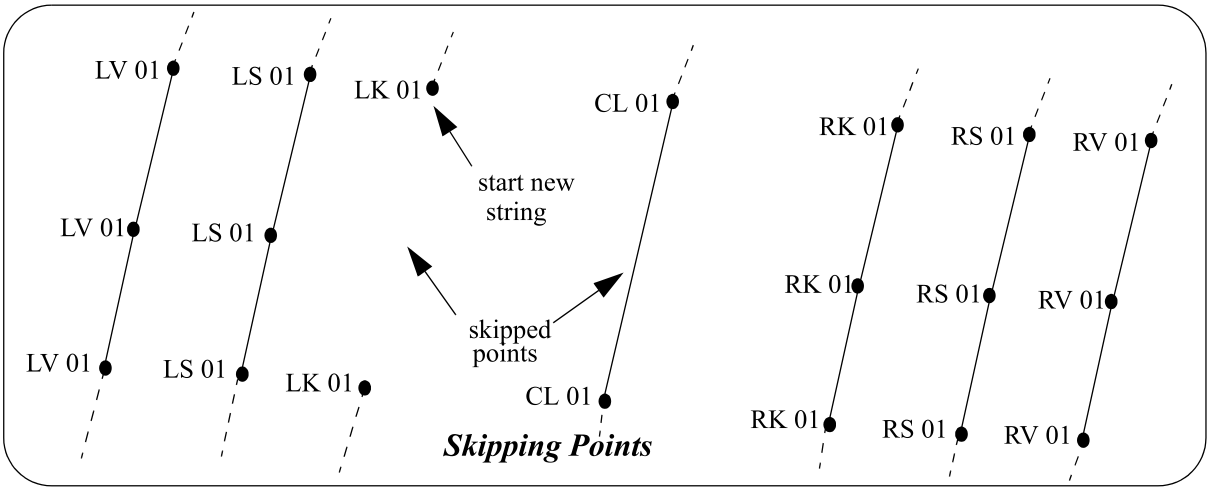
Please continue to the next section 36.4.4 Skipping Field Template Points.

36.4.4 Skipping Field Template Points

When picking up points using a field template, ***12d Model*** allows for one or more points to be skipped. By default, the points on the strings on either side of the skipped points will then be joined together.



By combining skipping points and start new string commands, points can be skipped and new strings started on the other side of the skipped points.



Please continue to the next section 36.4.5 Insert Template Points or Insert Multiple Codes.

36.4.5 Insert Template Points or Insert Multiple Codes

When picking up points using a field template, ***12d Model*** allows for one or more points to be inserted. The inserted points ***change*** the template from that point onwards so that extra strings can be picked up as they arise.

If the insert point command is given after the last point of a template, a flag can be applied to specify which template pick-up the inserted point is to be added to. That is, add it to the last series of points or the next series. This flag is called the "insert special" flag.

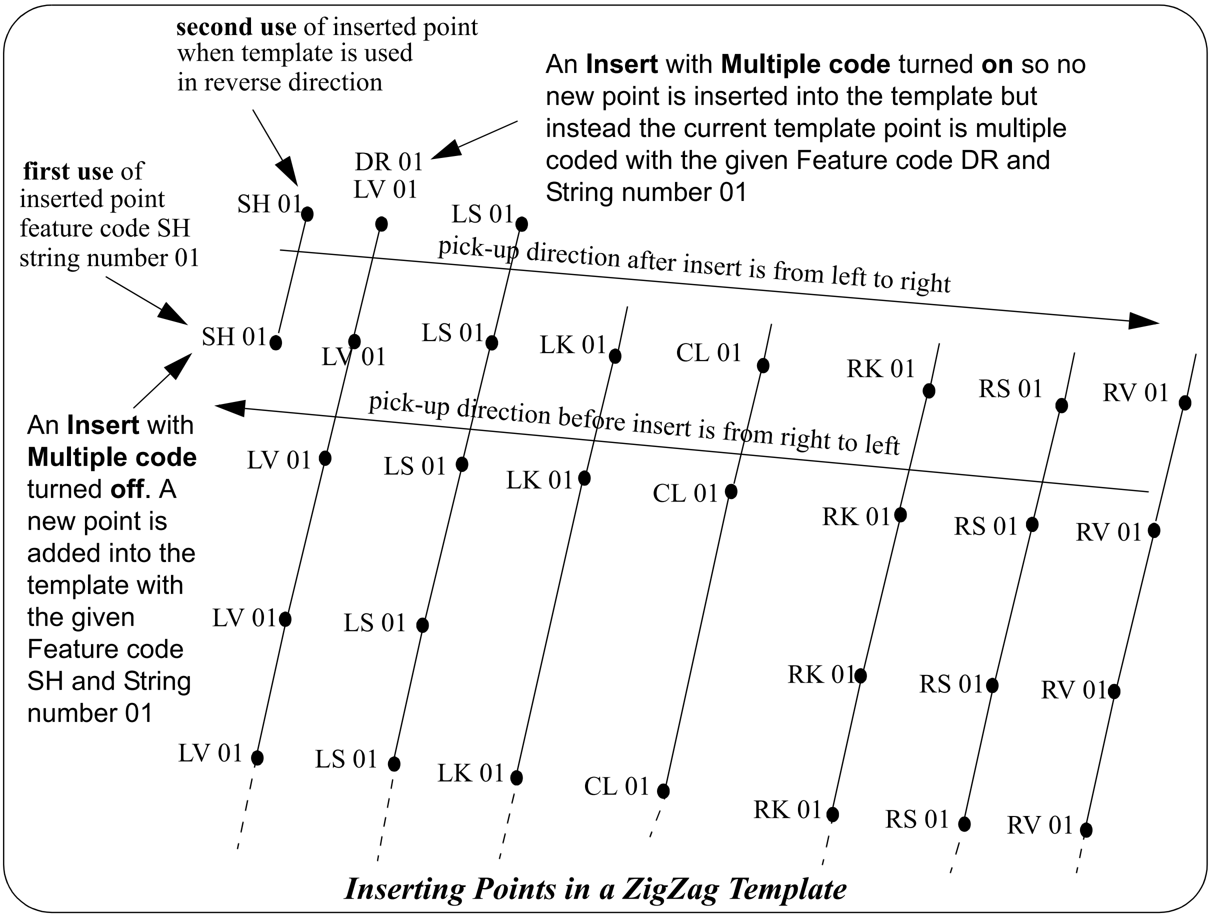
If ticked on in the insert panel, or if the flag given in the offset code is 1, the point will be added to the current series of points in the template being picked up.

In the case shown below where SH 01 is inserted, the insert special flag should be set to on so that the next picked up point will be on the current template. The insert would have been made after the last LV 01 observation in the last pick-up direction. The following pick-up will use the redefined template definition.

Note - in the example below, it is a zig-zag template so SH 01 is then used again straight away as the first point of the pickup when coming from left to right.

With the Insert, rather than insert new template points, it is also possible to give multiple codes to existing points in the template so that more than one code can be assigned to the one pick-up point (insert multiple codes).

In the case shown below, an insert was made on the next pick-up direction after the LV 01 observation. The multiple code tick box or flag was set on so that the last picked up point will be assigned the extra code specified, in this case DR 01. The template will be applied to all subsequent measurements so that the observed LV 01 string will also be coded DR 01.



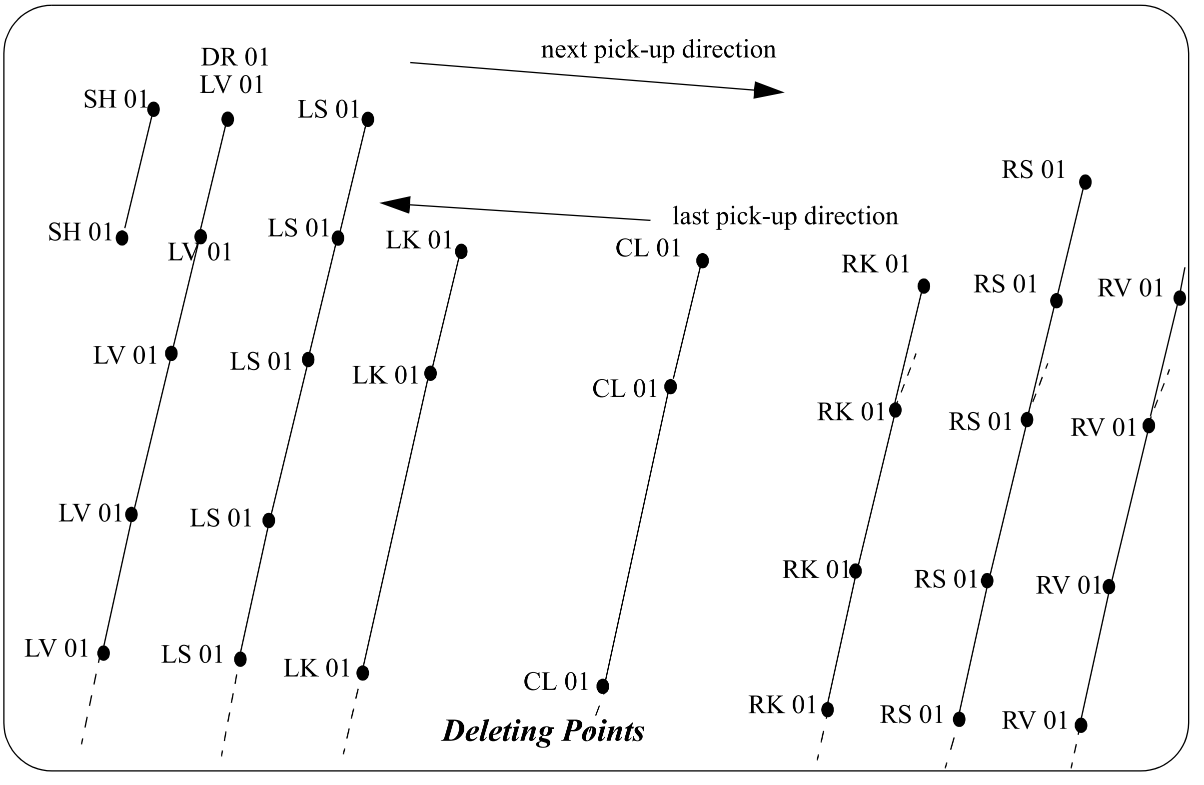
Inserting Points in a ZigZag Template

Please continue to the next section 36.4.6 Delete Template Points.

36.4.6 Delete Template Points

When picking up points using a field template, ***12d Model*** allows for one or more points to be deleted. The deleted points change the template from the next specified number of points inclusive of the current point. i.e. They are removed from the template.

If a template delete command was given after the LS 01 string in the next pick-up direction, and the number of specified points were 3, the template will be altered such that the next observed string will be RS 01.



Deleting Points

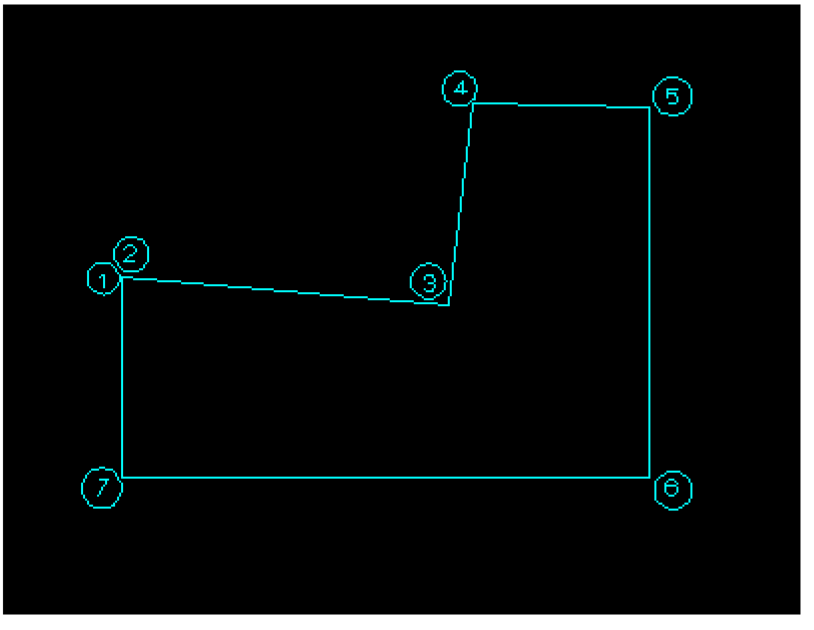
36.5 Shape field coding

If a an object of a standard section is to picked up such as a length of kerb, a shape can be defined and extruded/parallelled along a single pick-up string related to that shape.

For example, a kerb shape can be defined by observing all points on a typical section of the kerb and assigned a shape name. Then when picking up the length of kerb, only one reference string to the shape (defined when recording the shape) has to be picked up e.g. lip of kerb. On reduction, 12d can extrude or parallel the shape such that the strings/shape of the kerb are produced for the entire kerb pick-up.

Normally if each measurement is from a different string e.g back of kerb, lip kerb etc., then the feature code and string number would need to be re-entered with each measurement which is a very time consuming process. To simplify the coding for section pick up, ***12d Model*** uses *shape field coding.*

Basically, a ***12d Model*** *shape* consists of observing a number of points on a given section of an object. The shape can be given a unique name or have no name at all.



For example, a shape can be defined by observations shown in order above

1- being the reference point, 2- 7 being the shape points. In this case, the 1st point of the shape pick-up coincides with the reference point.

To define a ***12d Model*** *shape*, there is a command to *start and end the recording* of the shape. The *feature code*s and *string numbers* for the next series of measurements until the *stop recording* command is given, are stored in the final shape.

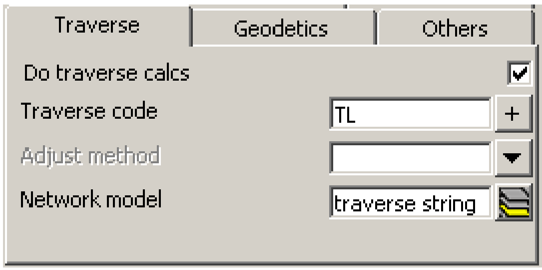
As the shape is defined, other field codes can be used in conjunction such as offset. In this example an observation may be made at 5 for the position of 6 using a vertical offset. Similarly for point 7 using the observation at a point near point 2. In addition, points 6 and 7 can be made non-tinable so that formation of a tin is constrained to the surface of the kerb.

The next step is to pick up the entire length of the kerb at the reference string position using the same *feature code* and *string number* used for the reference string in the shape pick-up. In this example the lip of kerb.

Once completed, the shape can be extruded or parallelled using the extrude or parallel commands. The parallel command will create a number of strings according to the number of points on the shape. The extrude will create a super string with a shape defined like a pipe string.

36.6 Traverse coding

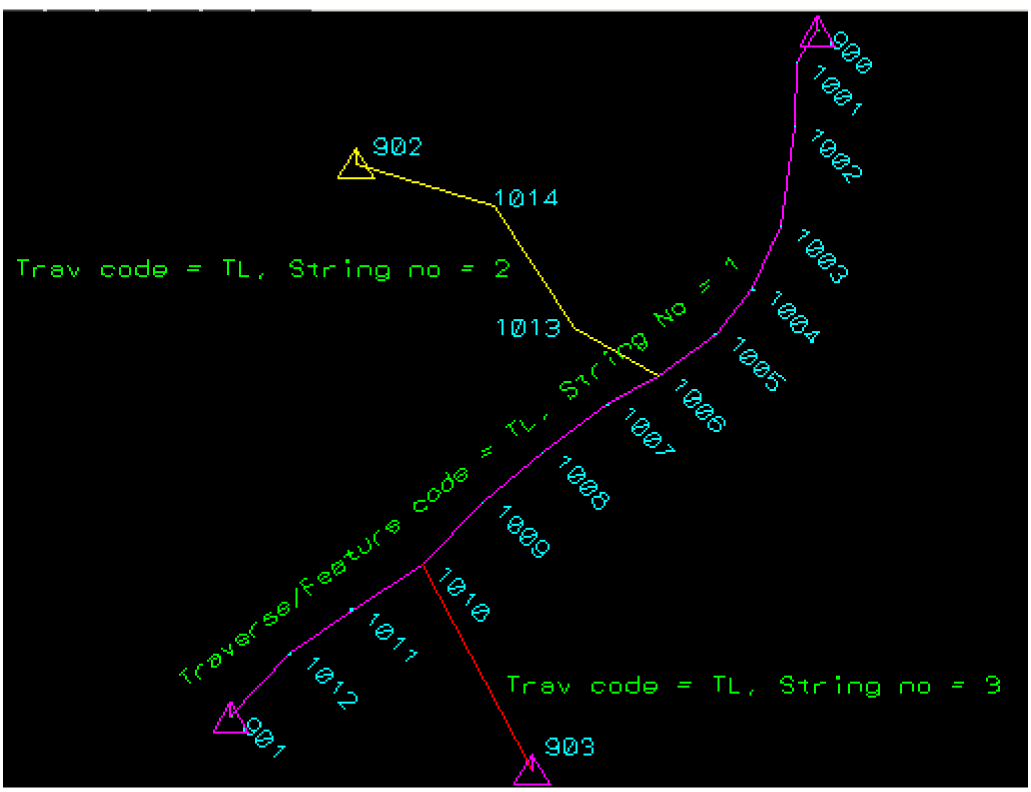
If a traverse is undertaken as part of a survey, a traverse code and string number can be coded so that 12d can extract the traverse information. The specific traverse code can be supplied in the survey reduction panel under the traverse tab. An example is shown below:



In this case, the feature code of TL will be searched in the field file on reduction, so that a traverse string can be extracted. The user is required to nominate the foresight measurement with the TL code in this example. If they also include the TL code in a backsight to a previously defined traverse leg, a reciprocal calculation will be made. It uses the pair of observations (Foresight and Backsight observations of the same line e.g. Foresight 1001 to 1002 and backsight from 1002 to 1001). This reciprocal calculation takes the mean of the distance and vertical angles eliminating the effects of refraction.

A number of separate but interrelated traverses can be extracted using differing string numbers in the field.

The traverse code also allows for adjustments to be made between known stations. This adjustment maybe be chosen in the reduction panel. This field is optional.



36.7 Field Coding for Leica Instruments

For Leica instruments, the Leica GSI format breaks lines of data into fixed length 'words' and cannot use the same encoding method as the other instruments. The ***12d Model*** coding system for the Leica TPS instruments is covered in the section 37.1 12d and Leica TPS Instruments.

36.8 Field Coding for Non Leica Instruments

EDM equipment is used to make readings of points in the field.

Rather than just collecting points, it is usually desirable to add extra information by coding the readings in a way that can be interpreted during the data reduction process and produce more valuable information. Unfortunately, this extra coding is non-standard and software specific.

In ***12d Model***, all the raw data files from different data collectors are converted into the one standard field file format (the ***12d Model*** field file) before being loaded into a ***12d Model*** Survey Reduction function and reduced. Hence the method for coding information in the data collector in the field needs to be well defined so that it can be sensibly converted into a ***12d Model*** field file (the section 36.10 The 12d Field File Format contains the complete description of the 12d field file).

When a measurement is taken, most data collectors (Sokkia, Geodimeter, Topcon, Nikon etc.) allow the user to enter text which is then output with the measurement data. Depending on the data collector, the total number of characters of text may be strictly limited and may also be restricted to only one line of text. In ***12d Model***, a coding system has evolved so the text can be interpreted in a meaningful way.

36.8.1 Blocks and the Block (Command) Delimiter

To allow the one line of text to hold a variety of different information, the line of text is broken up into smaller blocks and each block is processed separately.

The character (or characters) used as the block separator is called the *command delimiter* or *block delimiter*.

Obviously the command delimiter can not appear in any of the commands inside a block.

What is used for the *command delimiter* is user defined in the data collector definition and is found on the *Delimiters* tab of the *Survey.4d Create/Edit* panel in the section 36.9 Data Collector Definitions.

The default command delimiter is * and this will be used in the examples in this appendix.

Hence using the command delimiter, the text line is broken into separate blocks for processing.

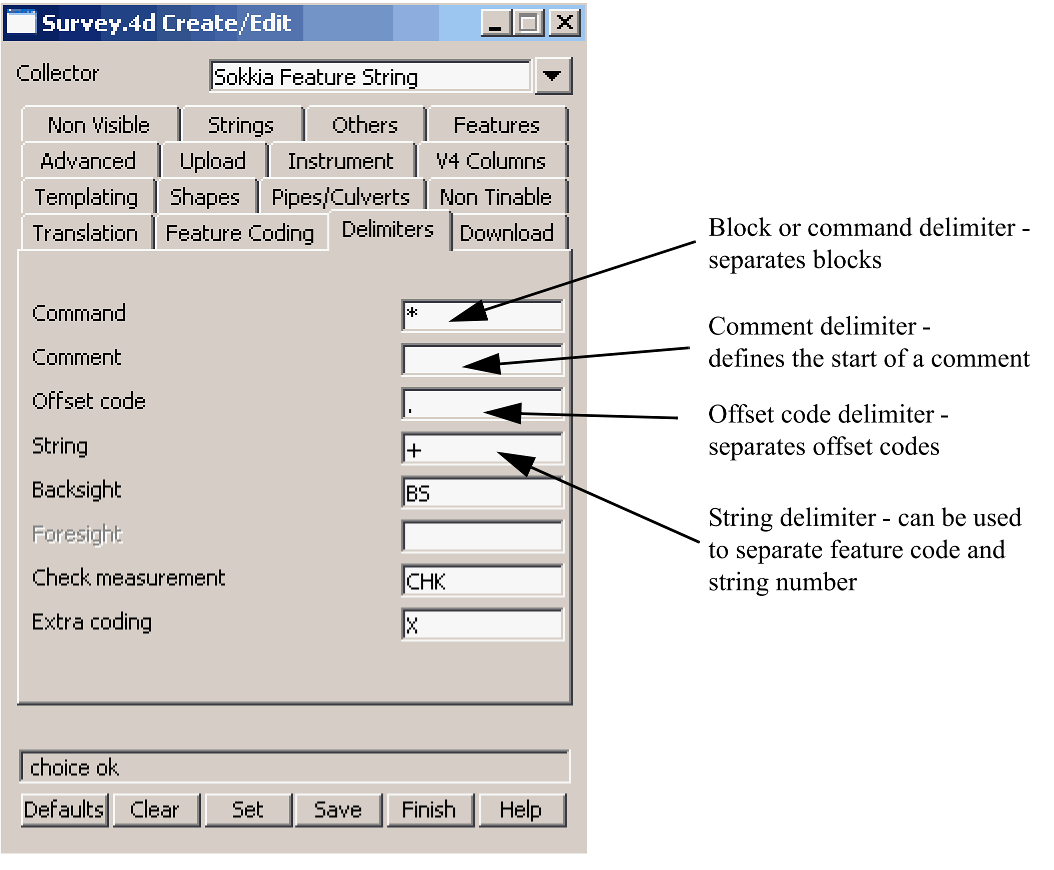
For example, the text string

EB01*XA.road

breaks into the separate blocks.

EB01

XA.road



A block can be either a ***control code block*** or a ***feature code block***.

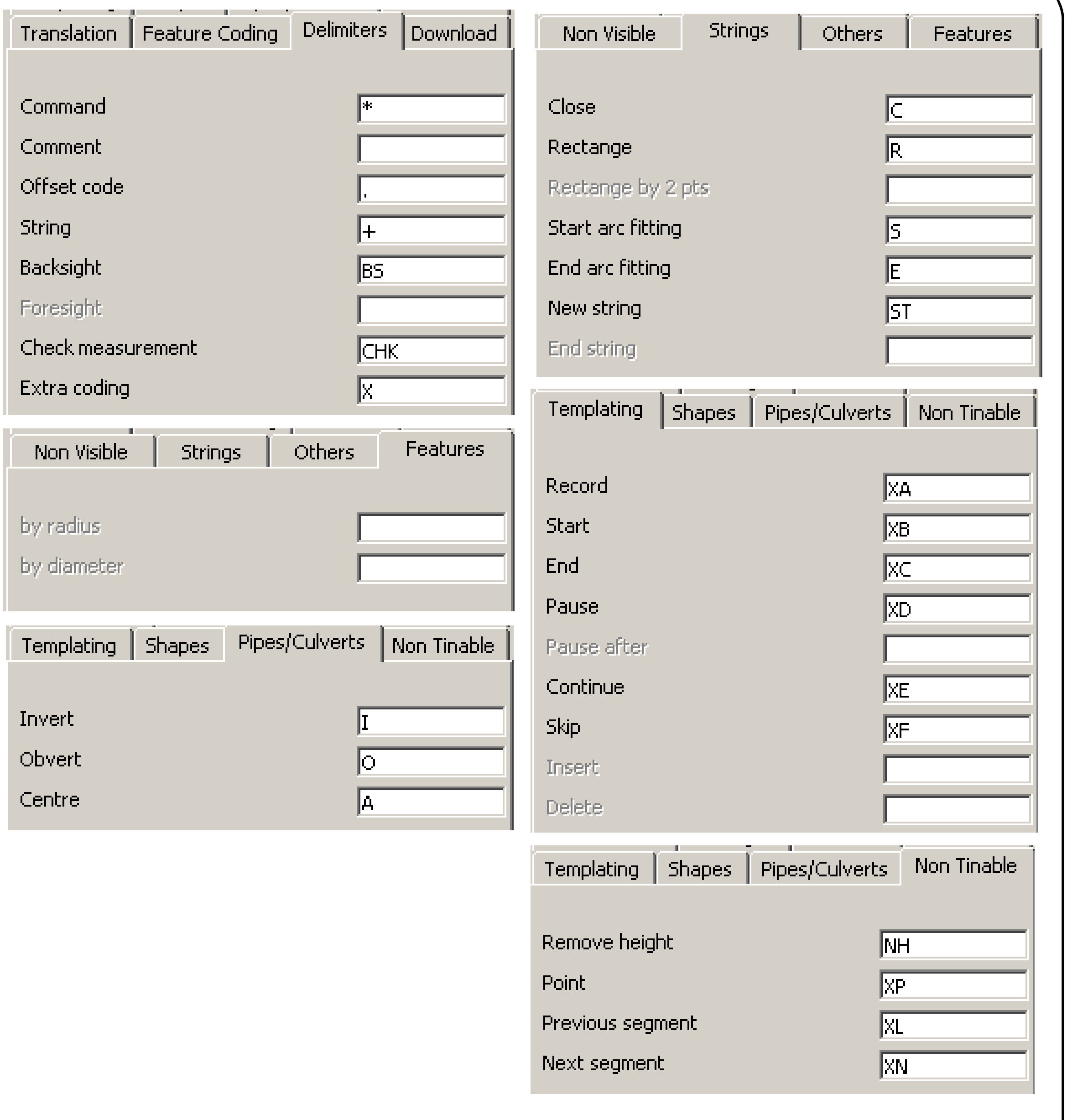
A *control code block* is any block that start with a valid *control code*.

Valid control codes are simply the one or more characters of text designated in the *Delimiters*, *Features*, *Templating*, *Pipes/Culverts*, *Not Tinable* and *Strings* tabs in the **Survey.4d Create/Edit** panel for the selected data collector definition.

If no control code appears at the start of the block, then the block is taken to be a *feature code* *block*.

For example, for the *Sokkia Feature String* data collector definition which is installed with ***12d Model***, the *valid control codes* are *****, dot (**.**), space,**+**, **BS**, **CHK**, **X**, **C**, **R**, **S**, **E**, **ST**, **XA**, **XB**, **XC**, **XD**, **XE**, **XF**, **NH**, **XP**, **XL**, **XN**, **I**, **O** and **A**

Any block starting with any of above control codes is a *control code block*. Blocks that aren't control code blocks are *feature code blocks*.



Tabs on the *Sokkia Feature String* data collector definition which define Control Codes

Blocks can include *comments* and how a comment in specified is given in the next section 36.8.2

Comments in a Block.

The feature code block is discussed in the section 36.8.3 Feature Code Blocks.

The explanation of each control code is given in the section 36.8.4 Control Code Blocks.

Please continue to the next section 36.8.2 Comments in a Block.

36.8.2 Comments in a Block

Comments are supported in a block. There is a *Comment* delimiter and all characters after the *comment* delimiter to the ***end*** of the ***block*** are treated as a comment.

Comments are attached to the measurement point as ***text***.

Hence the *measurement* will create a vertex on a super string and the *comment* will create text at that vertex of the super string.

The default for the *Comment* delimiter is **/** and it is defined by the *Comment* field on the *Delimiters* tab of the *Survey.4d Create/Edit* panel.

For example, if * is the command (block) delimiter and / is the comment delimiter, then in

EB01/first point of a template*XA.road

there are two blocks *EB01/first point of a template* and *XA.road*

and the text "first point of a template" in the first block is a comment and will be added to the super string as vertex text.

Another common character to use as a *comment delimiter* is a space (' '). For example, if * is the command delimiter and space is the comment delimiter, then in

EB01 first point of a template*XA.road

"first point of a template" is a comment.

The *comment* delimiter for the *Sokkia Feature String* data collector is a space

Please continue to the next section 36.8.3 Feature Code Blocks.

36.8.3 Feature Code Blocks

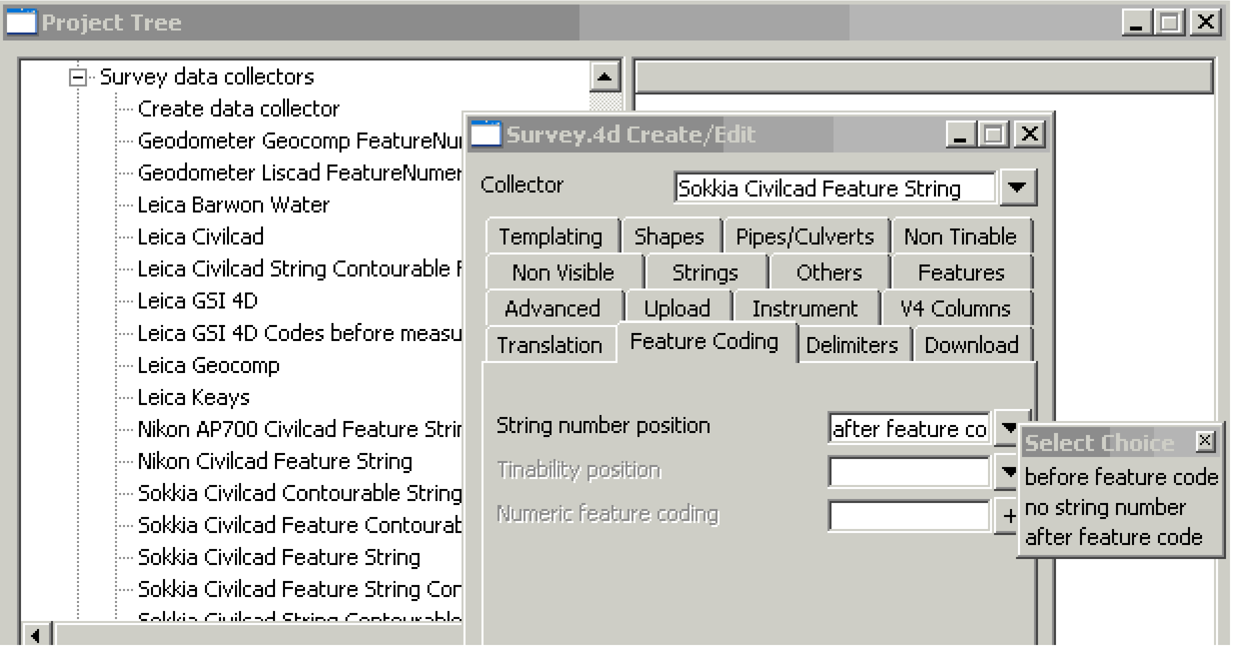
Feature Code, String Number and String Delimiter

In a feature code block, the *feature codes* and *string numbers* are given so that strings are automatically created during the reduction process.

The *feature code* and *string number* appear at the start of the feature code block and can be entered in a variety of ways.

Some surveyors like to enter the *string number* before the *feature code* and other surveyors prefer to enter the *string number after the feature code*. Some surveyors prefer to use *no string numbers* at all (SDRmap users) and use a *new string* control code to start a new string.

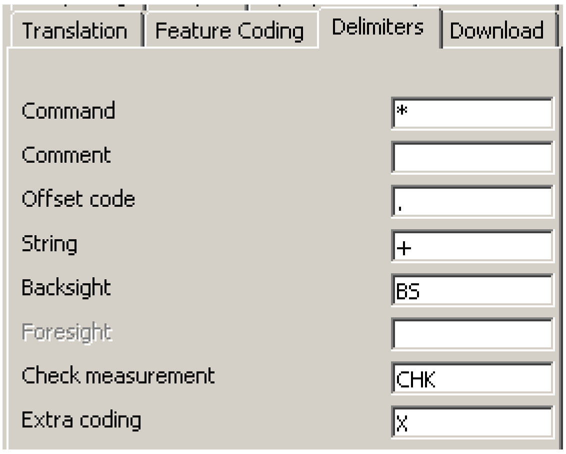
All three methods are supported in 12d. The setting is made on the *String number position* field on the *Feature Coding* tab of the *Survey.4d Create/Edit* panel. The default is 'before feature code'.



If string numbers are going to be used (either *before feature code* or *after feature code*) then there are rules for determining what is the feature code and what is the string number. Note that the *string number* is always *numeric* and can be any length.

Firstly, the *feature code* and *string number*s can be separated by an ***optional String*** delimiter given by the *String* field on the *Delimiters* tab of the *Survey.4d Create/Edit* panel. The default character for the *string delimiter* is **+**.

For example, 20+110KVA is interpreted as string number '20' and feature code '110KVA' if the string number comes before the feature code.



Delimiters Tab on the *Survey.4d Create/Edit* Panel

If a String delimiter is not used, then there are ***some restrictions*** on the feature code so that the feature code and string number can be split apart. Note that the *string number* is always *numeric* and can be any length.

If a string delimiter is not used, then either

(a) the *feature code* must be *purely alpha* and then the feature code can be of any length.

or

(b) the *feature code* is purely numeric. Then the feature code must the ***fixed length*** given by the *Numeric feature coding* field on the *Feature Coding* tab of the *Survey.4d Create/Edit* panel. If the *Numeric feature coding* field is not set and the feature code is numeric, then the whole thing is interpreted as a feature code with a string number of 0.

If a *feature code* is a ***mixture of alpha and numeric***, then a ***string delimiter must be used*** to separate the feature code from the string number. Once a string delimiter is used, feature code can be any length.

Tinability as Part of the String Number

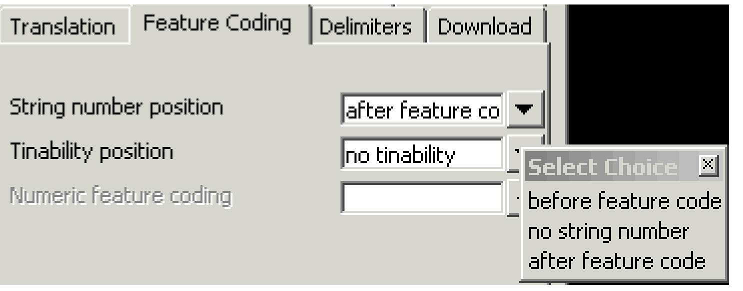
For compatibility with other software coding schemes, the tinability of a point can be included as part of the string number. This is ***not normally recommended*** since 12d has more flexible methods of defining tinability.

The field *Tinability position* on the *Feature Coding* tab of the *Survey.4d Create/Edit* panel controls whether it is used or not. The default is 'no tinability' as part of the string number.

If tinability is used as part of the string number, then it consists of either a **0** or a **1**.

If *Tinability position* is set to *before string*, the first character of the *string number* is stripped off and taken as the tinability flag. A value of 1 means the point is tinable and 0 is non-tinable.

If *Tinability position* is set to *after string*, the last character of the *string number* is stripped off and taken as the tinability flag. A value of 1 means the point is tinable and 0 is non-tinable.



Offset Codes

It is not always possible to measure a point directly but it may be possible to measure a point nearby and then measure an offset to adjust the measured point by and so produce the co-ordinates of the required point. The three offsets that are allowed in ***12d Model*** are height, radial and tangential. See the Section 36.3.2 Offsetsfor more information.

The three offsets height, radial and tangential have the codes H, R and T respectively.

The *offset codes* come after the feature code-string number combination and are separated by the ***Offset code*** delimiter whose default value is a decimal point (**.**) and is defined by the *Offset code* field on the *Delimiters* tab of the *Survey.4d Create/Edit* panel.

After the *offset code delimiter*, the offset code is given as either H, R or T and then a real value for the offset (with no spaces in between). The offset can be positive or negative with the sign only being recorded if negative. A second and third offset code (H, R or T) can follow the first but no offset code delimiter is used for the second or third offset codes. The offset code delimiter is only used to separate the field code-point number from the offset codes.

In the example 20+110KVA**.**H2.1T0.5 there is a horizontal offset of 2.1 and a tangential offset of 0.5.

Multiply Coded Points

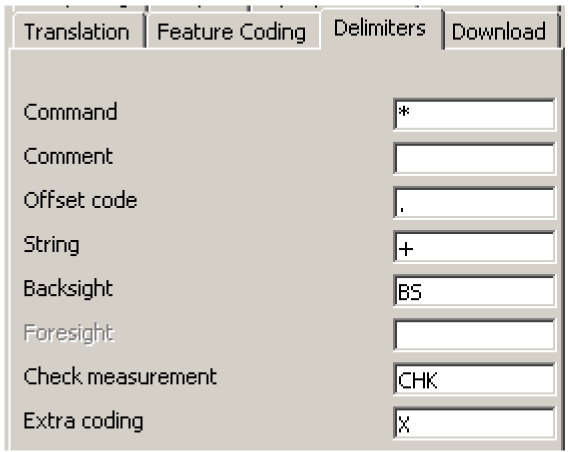
It is possible to have more than one *Feature code block* and this will create a multiply coded point.

For example, if the command delimiter is *, then EB01*PP would have the two feature code blocks ’EB01’ and ’PP’.

Please continue to the next section 36.8.4 Control Code Blocks.

36.8.4 Control Code Blocks

Control Codes on the Delimiters tab



Delimiters Tab on the *Survey.4d Create/Edit* Panel

Extra Coding

This code is only for the instruments supporting the Sokkia SDR format.

The default for the *extra coding* control code is **X** and it is defined by the *Extra coding* field on the *Delimiters* tab of the *Survey.4d Create/Edit* panel.

For some instruments, extra lines of information can be entered after a measurement. For example on a Sokkia by using the note (13NM) to add more information to the previous measurement line (07).

The *extra coding* control code is used to append this additional information to the information on the measurement line. Hence it is then possible to have extra blocks of information on the line (or lines) following the measurement line.

The block containing the *extra coding* control code must be the last block on the measurement line.

For example,

EB01/first point of a template*XA.road*X
/ this is some extra info that happens to be a comment

is the same as

EB01/first point of a template*XA.road*/ this is some extra info that happens to be a comment

Note that there must be a feature code block on the measurement line otherwise a point will be created with no name.

Backsight

The default for the *backsight* control code is **BS** and it is defined by the *Backsight* field on the *Delimiters* tab of the *Survey.4d Create/Edit* panel.

The rest of the block after the *backsight control code* is the name of the station that the backsight was to, or the point number that the backsight was to.

For example, if *BS* is the backsight control code

BSPSM3

would designate a backsight to the station PSM3.

If only the backsight control code exists, the point number from the reading is taken as the point number for the backsight. This may be data collector dependent.

Warning for feature codes starting with the *backsight* control code.

If the feature codes comes before the string number, then feature codes for ordinary measurements ***can not*** start with the backsight control code because they would be interpreted as backsight measurements and no measurement point would be created.

If the feature codes comes after the string number, then whenever a feature codes for ordinary measurements is used that starts with the backsight control code, a string number (which may be zero for a point string) ***must*** be used otherwise the feature code will be interpreted as a backsight measurement and no measurement point would be created.

Foresight

There is no default for the *foresight* control code. It is defined by the *Foresight* field on the *Delimiters* tab of the *Survey.4d Create/Edit* panel.

The rest of the block after the *foresight control code* is the name of the station that the foresight was to, or the point number that the foresight was to.

For example, if *FS* is the backsight control code

FSPSM3

would designate a backsight to the station PSM3.

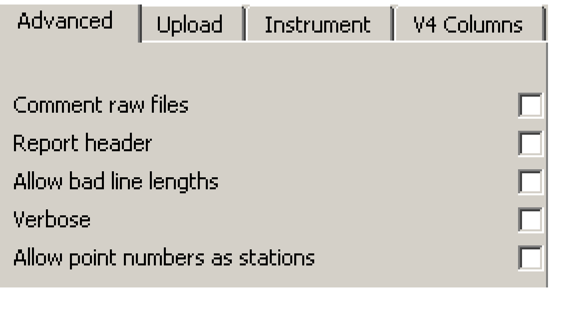
If only the foresight control code exists, the point number from the reading is taken as the point number for the foresight. This may be data collector dependent.

Warning for feature codes starting with the *foresight* control code if the *Allow point numbers as stations* flag is *not set* on the Advanced tab of the *Survey.4d Create/Edit* panel.

If the feature codes comes before the string number, then feature codes for ordinary measurements **can not** start with the foresight control code because they would be interpreted as foresight measurements and no measurement point would be created.

If the feature codes comes after the string number, then whenever a feature codes for ordinary measurements is used that starts with the check measurement control code, a string number (which may be zero for a point string) must be used otherwise the feature code will be interpreted as a foresight measurement.

If the *Allow point numbers as stations* flag is set on, then a point number used as a foresight will create a measurement point for that point number.



Advanced tab on the *Survey.4d Create/Edit* Panel

Check Measurement

The default for the *check measurement* control code is **CHK** and it is defined by the *Check measurement* field on the *Delimiters* tab of the *Survey.4d Create/Edit* panel.

The rest of the block after the *check measurement control code* is the name of the station that the check measurement was to, or the point number that the check measurement was to.

For example, if *CHK* is the check measurement control code

CHKPSM3

would designate a check measurement to the station PSM3.

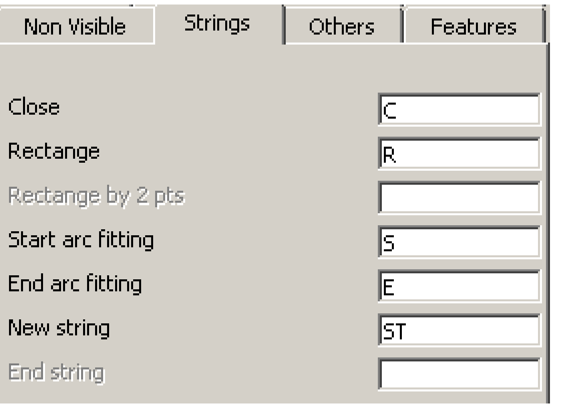
If only the check measurement control code exists, the point number from the reading is taken as the point number that the check measurement is made to. This may be data collector dependent.

Warning for feature codes starting with the *check measurement* control code.

If the feature codes comes before the string number, then feature codes for ordinary measurements ***can not*** start with the check measurement control code because they would be interpreted as check measurements and no measurement point would be created.

If the feature codes comes after the string number, then whenever a feature codes for ordinary measurements is used that starts with the check measurement control code, a string number (which may be zero for a point string) ***must*** be used otherwise the feature code will be interpreted as a check measurement and no measurement point would be created.

Control Codes on the Strings tab



Strings tab on the *Survey.4d Create/Edit* Panel

Close String

The default for the *close string* control code is **C** and it is defined by the *Close* field on the *Strings* tab of the *Survey.4d Create/Edit* panel.

The *close string* control code closes the string that the measurement is a point of.

See the Section 36.3.4 Close String for more information on closing a string.

Rectangle

The default for the *rectangle* control code is **R** and it is defined by the *Rectangle* field on the *Strings* tab of the *Survey.4d Create/Edit* panel.

The *rectangle* control code uses the current point and the previous two points in the same string (three points total) and creates a new point to form a parallelogram.

See the Section 36.3.5 Rectangle for more information on forming a rectangle.

Rectangle by 2 Pts

A *rectangle_2* control code can be defined by the *Rectangle by 2 pts* field on the *Strings* tab of the *Survey.4d Create/Edit* panel.

The *rectangle_2* control code uses the current point, last point and an offset. The rectangle is defined by two points (reference side) and a offset.

If a positive offset value is given, two points will be created to the right of the reference side.

If a negative offset value is given, two points will be created to the left of the reference side.

If no Description is given, the two new points will be joined to the given points in a closed rectangular string, and will have the same feature code as the points given.

If the feature code and string number exist, then a search is made for the last occurrence of two points with the same feature code and string number. If found, then these points are used to define the reference side of the rectangle.

If the point number exists, then a search is made for the last occurrence of two points with the same feature code and string number as the point given by the point number. If found, then these points are used to define the reference side of the rectangle.

Two consecutive rectangles are unable to be defined side by side. In other words if the two points given are part of string of greater than two vertices, the command will only work for sets of two points that are exclusively defined. i.e. For a 5 point string, a rectangle can be defined by points 1 and 2, and 4 and 5.

See the Section 36.3.6 Rectangle by 2 Points for more information on forming a rectangle by 2 points.

Start Arc Fitting

The default for the *start arc fitting* control code is **S** and it is defined by the *Start arc fitting* field on the *Strings* tab of the *Survey.4d Create/Edit* panel.

The *start arc fitting* control code starts arc fitting with the current point.

See the Section 36.3.9 Arcs Through Points for more information on arc fitting.

End Arc Fitting

The default for the *end arc fitting* control code is **E** and it is defined by the *End arc fitting* field on the *Strings* tab of the *Survey.4d Create/Edit* panel.

The *end arc fitting* control code ends arc fitting at the current point. That is, the current point is included in the arc fitting but it is the last point used in the arc fitting.

See the Section 36.3.9 Arcs Through Points for more information on arc fitting.

New String

The default for the *new string* control code is **ST** and it is defined by the *New string* field on the *Strings* tab of the *Survey.4d Create/Edit* panel.

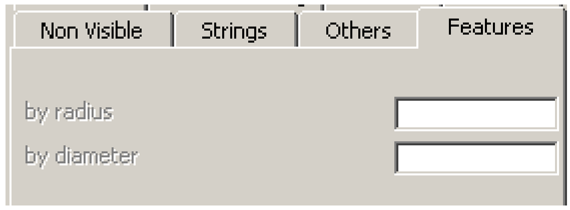
The *new string* control code starts a new string with the current point even if the feature code and string number haven't changed.

See the Section 36.3.3 Start New String for more information on starting a new string.

End String

The *end string* control code ends the current string even if the feature code and string number haven't changed for the next point.

Control Codes on the Features tab



Features tab on the *Survey.4d Create/Edit* Panel

Feature by Radius

There is no default for the *feature by radius* control code. It it is defined by the *by radius* field on the *Features* tab of the *Survey.4d Create/Edit* panel.

The *feature by radius* control code denotes that the feature has a given radius. For example, if RA was used then TRE*RA2 would denote a feature code TRE with a radius of 2 units.

See the Section 36.3.7 Feature for more information on feature strings.

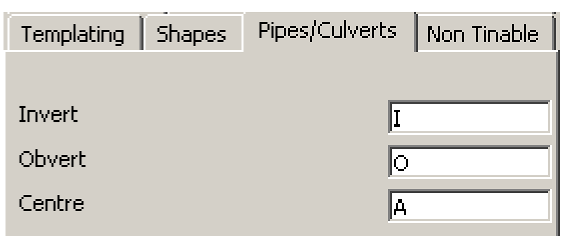
Feature by Diameter

There is no default for the *feature by diameter* control code. It it is defined by the *by diameter* field on the *Features* tab of the *Survey.4d Create/Edit* panel.

The *feature by diameter* control code denotes that the feature has a given diameter. For example, if DI was used then TRE*DI2 would denote a feature code TRE with a diameter of 2 units.

See the Section 36.3.7 Feature for more information on feature strings.

Control Codes on the Pipes/Culverts tab



Pipes/Culvert tab on the *Survey.4d Create/Edit* Panel

Invert Point for Pipe or Culvert

The default for the *invert level* control code is **I** and it is defined by the *Invert* field on the *Pipe/ Culverts* tab of the *Survey.4d Create/Edit* panel.

The *invert level* control code denotes that the point is an invert level (bottom).

The invert commands can also denote the diameter of a pipe or width and height of a culvert. For example, I.3 denotes the measurement was an invert level and it was a pipe of diameter 0.3 units. I.3x.4 denotes the measurement was an invert level and it was a box culvert of width 0.3 and height 0.4 units.

Obvert Point for Pipe or Culvert

The default for the *obvert level* control code is **O** and it is defined by the *Obvert* field on the *Pipe/ Culverts* tab of the *Survey.4d Create/Edit* panel.

The *obvert level* control code denotes that point is an obvert level (top).

The obvert command can also define the diameter of a pipe or width and height of a culvert. For example, O.3 denotes the measurement was an invert level and it was a pipe of diameter 0.3 units. O.3x.4 denotes the measurement was an invert level and it was a box culvert of width 0.3 and height 0.4 units.

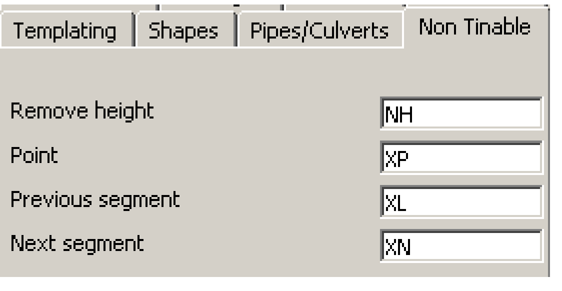
Centre Point for Pipe or Culvert

The default for the *centre level* control code is **A** and it is defined by the *Centre* field on the *Pipe/ Culverts* tab of the *Survey.4d Create/Edit* panel.

The *centre level* control code denotes that point is an centre level (axial).

The centre command can also define the diameter of a pipe or width and height of a culvert. For example, A.3 denotes the measurement was an invert level and it was a pipe of diameter 0.3 units. A.3x.4 denotes the measurement was an invert level and it was a box culvert of width 0.3 and height 0.4 units.

Control Codes on the Non Tinable tab



Non tinable tab on the *Survey.4d Create/Edit* Panel

Remove Height

The control code to set the height of the current point to null.

The default for the *remove height from point* control code is **NH** and it is defined by the *Remove height* field on the *Non-Tinable* tab of the *Survey.4d Create/Edit* panel.

Point

The default for the *make point non-tinable* control code is **XP** and it is defined by the *Point* field on the *Non-Tinable* tab of the *Survey.4d Create/Edit* panel.

The *make point non-tinable* control code makes the current point non-tinable. That is, the point is not used in tins (triangulations).

Previous Segment

The default for the *make previous segment non-tinable* control code is **XL** and it is defined by the *Previous segment* field on the *Non-Tinable* tab of the *Survey.4d Create/Edit* panel.

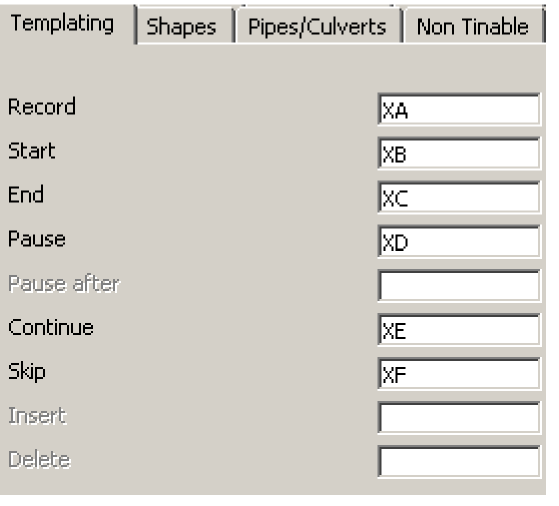
The *make previous segment non-tinable* control code makes the previous segment non-tinable. That is, the segment that the current point is the end of is non-tinable and hence not used as a breakline in tins (triangulations).

Next Segment

The default for the *make next segment non-tinable* control code is **XN** and it is defined by the *Next segment* field on the *Non-Tinable* tab of the *Survey.4d Create/Edit* panel.

The *make next segment non-tinable* control code makes the next segment point non-tinable. That is, the segment that the current point is the start of is non-tinable and hence not used as a breakline in tins (triangulations).

Control Codes on the Templating tab



Templating tab on the *Survey.4d Create/Edit* Panel

Record

The control code to denote that this is the first point of a new field template definition. The field template definition continues until the next *stop recording a field template* control code or a *start using a field template* control code.

If the field template is to have a name, then the *start recording* control code is followed by the *offset code* delimiter and then the name for the field template. If no name is given, then the field template defines the default field template.

For example, if *XA* starts recording for the default template, *XA.road* starts recording the field template called road

The default for the *start recording a field template* control code is **XA** and it is defined by the *Record* field on the *Templating* tab of the *Survey.4d Create/Edit* panel.

See the Section 36.4 Field Templates for more information on field templates.

Start

The control code to denote that a field template is being used. This point takes the feature code and string number from the field template. If it is a named field template, then the control code is followed by the *offset code* delimiter and then the name of the field template.

The field template can also be used as a forward template, a reverse template or as a zig-zag template starting on either a zig or on a zag.

The denote using the field template as a forward template, add the *offset code* delimiter and *for.*

To denote using the field template as a reverse template, add the *offset code* delimiter and *rev.*

To denote zig-zag mode starting on a zig, add the *offset code* delimiter and *zig.*

To denote zig-zag mode starting on a zag, add the *offset code* delimiter and *zag.*

If anything other than ’for’, ’rev’ or ’zag’ is given, the field template is used as a zig-zag template starting on a zig.

For example, if *XB* starts using the default template, *XB.road*.zag starts using the field template called road as a zig-zag field template starting on a zag.

The default for the *start using a template* control code is **XB** and it is defined by the *Start* field on the *Templating* tab of the *Survey.4d Create/Edit* panel.

See the Section 36.4 Field Templates for more information on field templates.

End

The control code to end the use of the current field template after the current point. That is, the current point uses the field template but subsequent measurements don’t.

The default for the *stop using a template* control code is **XC** and it is defined by the *End* field on the *Templating* tab of the *Survey.4d Create/Edit* panel.

See the Section 36.4 Field Templates for more information on field templates.

Pause

The control code to pause using the current field template. The current point does not use the field template.

The default for the *pause using a template* control code is **XD** and it is defined by the *Pause* field on the *Templating* tab of the *Survey.4d Create/Edit* panel.

See the Section 36.4 Field Templates for more information on field templates.

Pause after

The control code to pause using the current field template and the current point does use the field template.

There is no default for the *pause after* control code and it is defined by the *Pause after* field on the *Templating* tab of the *Survey.4d Create/Edit* panel.

See the Section 36.4 Field Templates for more information on field templates.

Continue

The control code to continue the use of a paused field template. The current point uses the field template.

The default for the *continuing using a field template* control code is **XE** and it is defined by the *Continue* field on the *Templating* tab of the *Survey.4d Create/Edit* panel.

See the Section 36.4 Field Templates for more information on field templates.

Skip

The control code to skip point one or more points of the field template. The current point uses the next point *after* the skipped points from field template.

If more than one point is to be skipped then the control code is followed by the *offset code* delimiter and then number of points to be skipped.

For example, if *XF* is the skip points control code, *XF.2* skips two points. Note that just XF skips one point and the ’.1’ is not required.

The default for the *skipping field template points* control code is **XF** and it is defined by the *Skip* field on the *Templating* tab of the *Survey.4d Create/Edit* panel.

See the Section 36.4 Field Templates for more information on field templates.

Insert

A default control code to insert a point in the field template can be added to this field. The point in inserted after the last point.

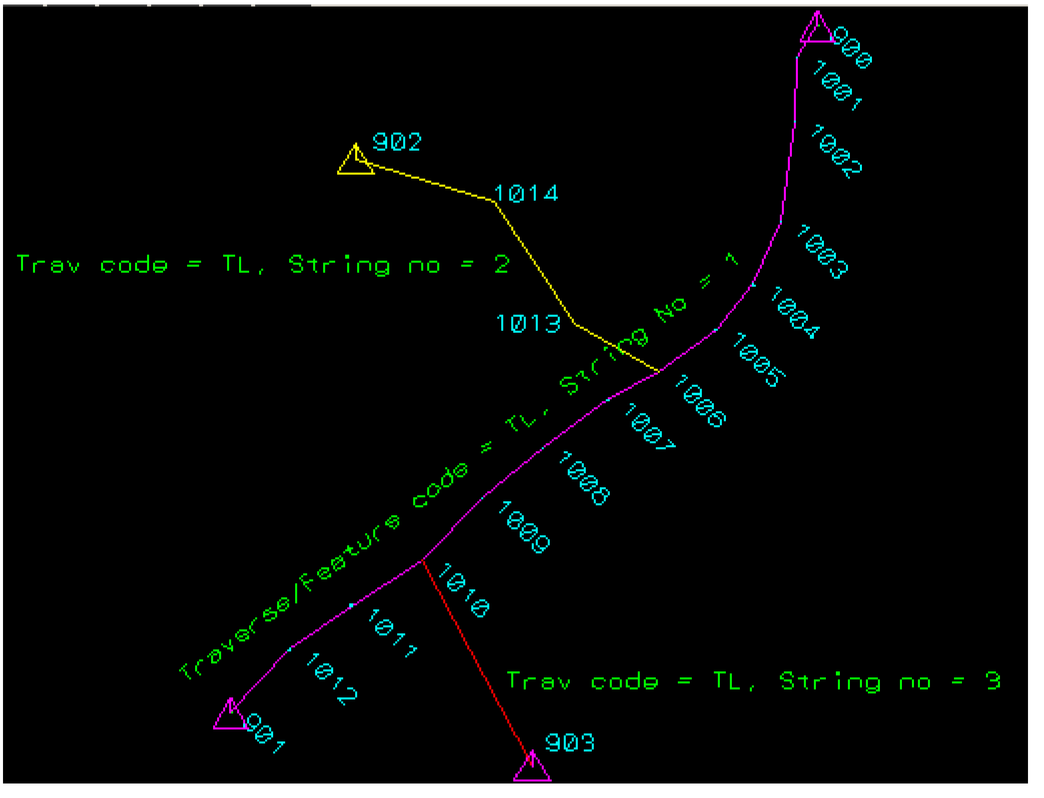
See the Section 36.4 Field Templates for more information on field templates.

Delete

A default control code to delete a point in the field template can be added to this field. a number of points can be nominated for deletion.

See the Section 36.4 Field Templates for more information on field templates.

36.8.5 Feature coding for traverse extraction (Non- Leica instruments)



In the example shown above, three distinct traverses were observed. The main traverse between stations 900 and 901 were given a feature code of TL and a string number of 1 (TL1). A typical observation description follows:

Setup on station 900. A foresight measurement was taken from 900 to 1001. A typical feature code may be:

TL1 FS1001. Where the block delimiter is a space(" ") breaking the feature code up into TL1 and FS1001. This tells the reduction that the observation was a traverse because it has a feature code of TL as specified in the traverse tab of the reduction panel. The foresight control code may be FS, telling the reduction that the observation is a foresight observation to a named point

Setup on station 1001. A backsight to station 900 was observed. A typical feature code may be:

TL1 BS900. Where BS is the backsight control code

A foresight to 1002 may be coded as:

TL1 FS1002.

And so forth.

The observation to station 1013 may be made at the same time as when observing to station 1007 from the setup station 1006. Simply changing the string number to 2 for the observation to 1013 will ensure that it is treated as a different traverse, *i.e.* TL2 FS1013.

The second traverse between stations 1006 and 902 were given a feature code of TL and a string number of 2 whilst the traverse between stations 1010 and 903 were given a feature code of TL and a string number of 3.

The feature code/ string number needs to be present for at least the foresight or measured leg of each of the traverse legs. If they are also present for the backsight, then foresight/backsight pairs will be grouped together and reciprocal calculations done for that leg.

36.9 Data Collector Definitions

Unfortunately, each brand of data collector has its own method of communicating with a computer and a software package.

To allow for a variety of data collectors, ***12d Model*** lets the user create and edit data collector definitions which are simply user defined sets of data collector parameters stored under user specified names.

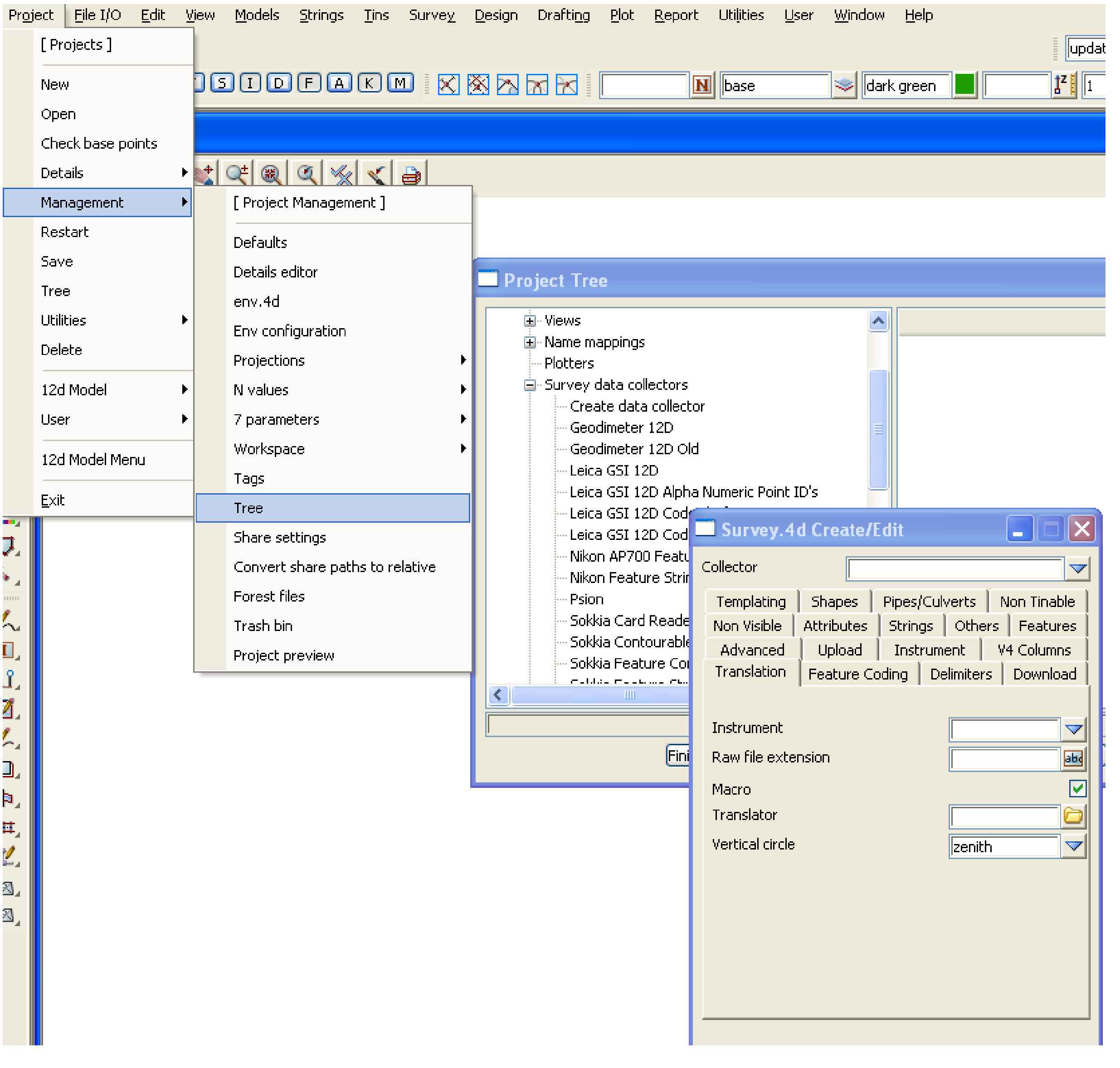
The set of data collector definitions are stored in a file but are created and edited using the ***12d Model*** panel *Survey.4d Create/Edit*.

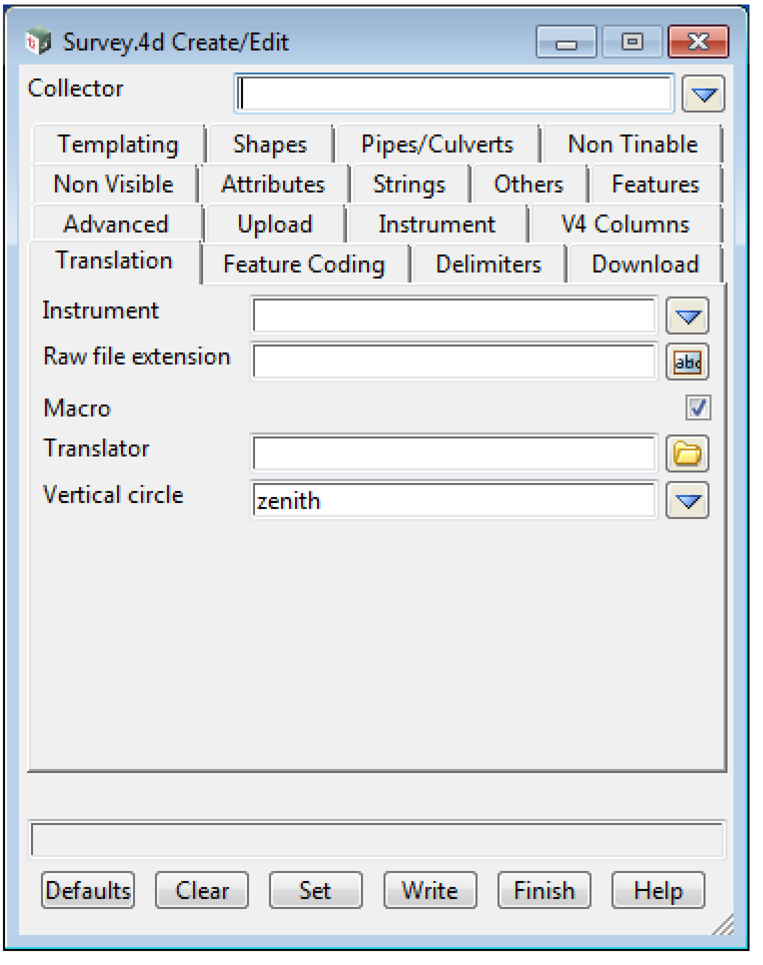
The *Survey.4d Create/Edit panel* is accessed via **Project** => **Management** =>**Tree**.

First select **Project** => **Management** =>**Tree** and then click on the **+** beside the Project name to expand the list of available information for the project. Then click on the **+** beside *Survey data collectors* to see the list of existing 12d data collector definitions.

Double click LB on *Create data collector* to create a new data collector definition, or double click LB on an existing data collector definition to examine and/or modify it.

The **Survey.4d Create/Edit** panel will then appear.





The fields and buttons used in this panel have the following functions.

Field Description	Type	Defaults	Pop-Up
Collector	input		available data collectors

name of the data collector to create/edit.

Defaults button

set all the panel fields to default values.

Clear button

clear the values in the panel fields in all the tabs.

Set button

set the values in the panel fields for the given data collector for this session of **12d Model**.

WARNING - Set does not save the changes to the survey.4d file. To save the changes, click on the 'Save' button.

Save button

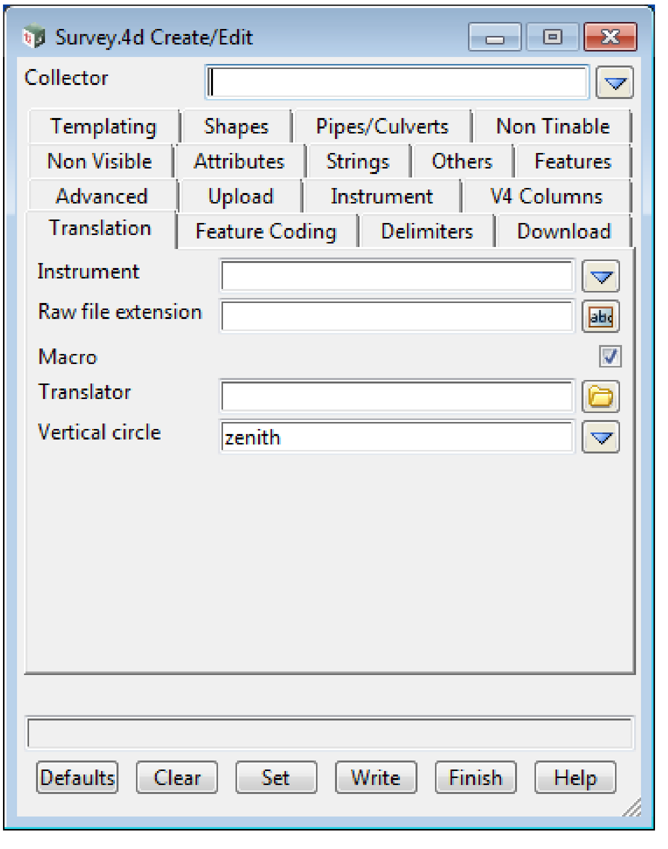
store the data collector definitions to the survey.4d file.

For descriptions of each of the tabs on the panel, go to:

Translation Translation tab
Feature coding Feature Coding tab
Delimiters Delimiters tab
Download Download tab

Advanced	Advanced tab
Upload	Upload tab
Instrument	Instrument tab
V4 Columns	V4 Columns tab
Templating	Templating tab
Shapes	Shapes tab
Pipe/Culverts	Pipes/Culverts tab
Non tinable	Non Tinable tab
Non visible	Non visible tab
Attributes	Attributes tab
Strings	Strings tab
Other	Others tab
Features	Features tab

Translation tab



Instrument input Geodimeter, Leica
Nikon DR1, Nikon AP700, Sokkia 20/33
Topcon GT700, Topcon FC5

type of data collector format.

Raw file extension input

file ending to use for the raw file when it is downloaded.

Macro tick box tick

if tick, then the Translator *pop-up list only shows* **12d Model** *macros.*
If not ticked, then the Translator *pop-up list only shows programs.*

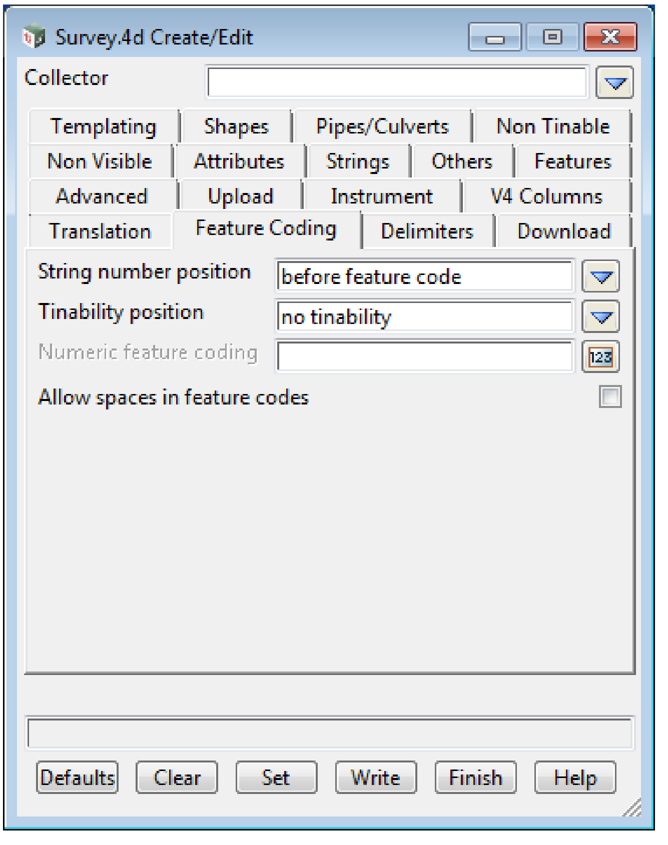
Translator input list of translators in library

name of the macro/program to translate the raw survey file into the 12d field file format.

Vertical circle input zenith zenith, nadir

define the vertical circle zero.

Feature Coding tab



String number position input before before feature code
no string number
after feature code

defines if the string number is before or after the feature code or is not used at all (no string number). When 'no string number' then the New string command is used to start new strings with the same feature code.

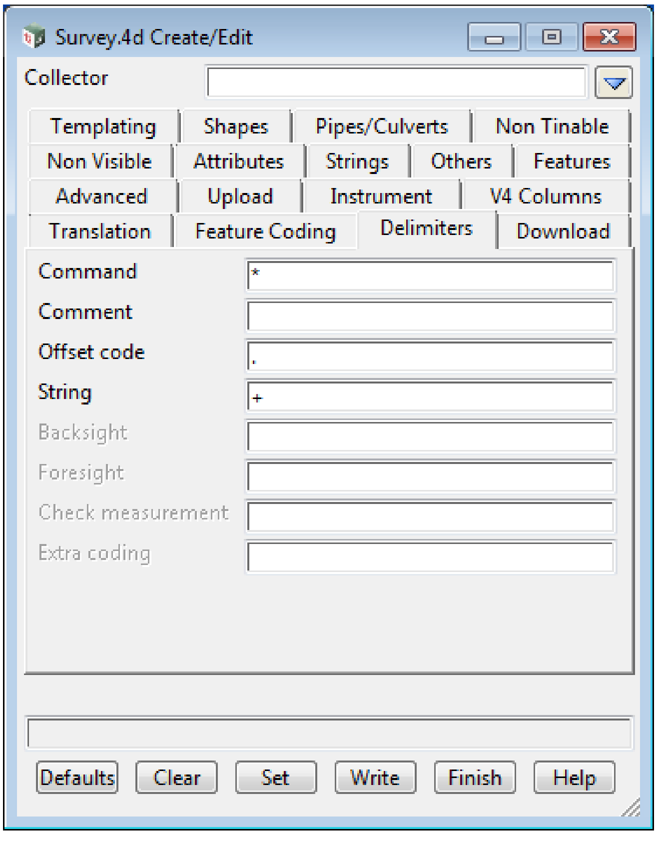
Tinability position input no tinability before string
no tinability
after string

defines the position or absence, of the tinability flag. This should only be used for compatibility with data coded for other systems such as CivilCAD. 12d has more flexible methods of defining tinability.

Numeric feature coding integer

if 'no string number' is given for String number position then Number feature coding is ignored.
If a number is given, then the feature code must be numeric and the given number is the length of the feature code (that is, the number of digits in the feature code).
If no number is given, then feature codes are alphabetical characters only except when a String code is used to separate the feature code and string number.

Delimiters tab



Command input *

defines the separator between the feature code/string number and op code commands. The delimiter characters can not be used in any other part of the command line.

Comment input /

defines the separator for comments. Note that this can be a space.

Offset code input dot (.)

defines the beginning of the offset code block. If the offset code block exists, it must come immediately after the Feature code/string number/tinability block. The offset code used R for radial, T for tangential, H for height.

String input +

separates the feature code from the string number. Used when the feature code includes numbers. For example 110KVA+20   would be a feature code of 110KVA and a string number of 20.

Backsight input BS

if the feature code is this control code then the measurement is to a backsight and not a point.

Foresight input

if the feature code is this control code then the measurement is to a foresight and not a point.

Check measurement input CHK

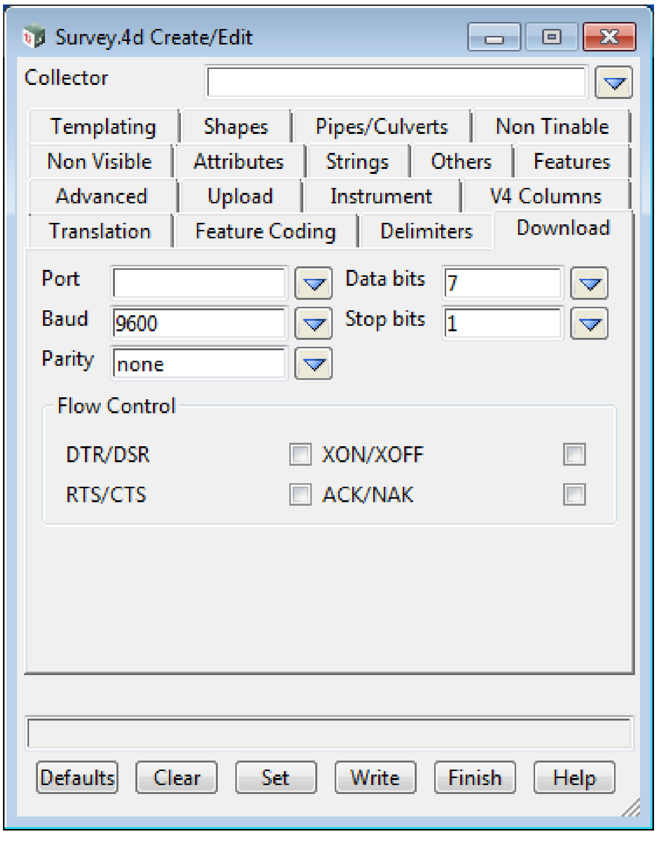
if the feature code is this control code then the measurement is a check measurement and not a point.

Extra coding input X

denotes that the information following is appended to the previous information for the measurement. This allows extra coding than may be allowed for on the measurement line. Mainly for Sokkia using the note (13NM) to add more information to the previous measurement line (07).

A feature code block must be on the measurement line or a measurement with no name will be created. Also a command block can't be split between two lines.

Download tab



Port	input	COM1	COM1 to COM4

port on the computer to use to communicate with the data collector for downloads.

Baud rate	input	9600	110, 300 ... 256000

baud rate to use for the computer port.

Data bits	input	7	5, 6, 7, 8

number of data bits to use.

Stop bits	input	1	0, 1, 1.5, 2

number of stop bits to use.

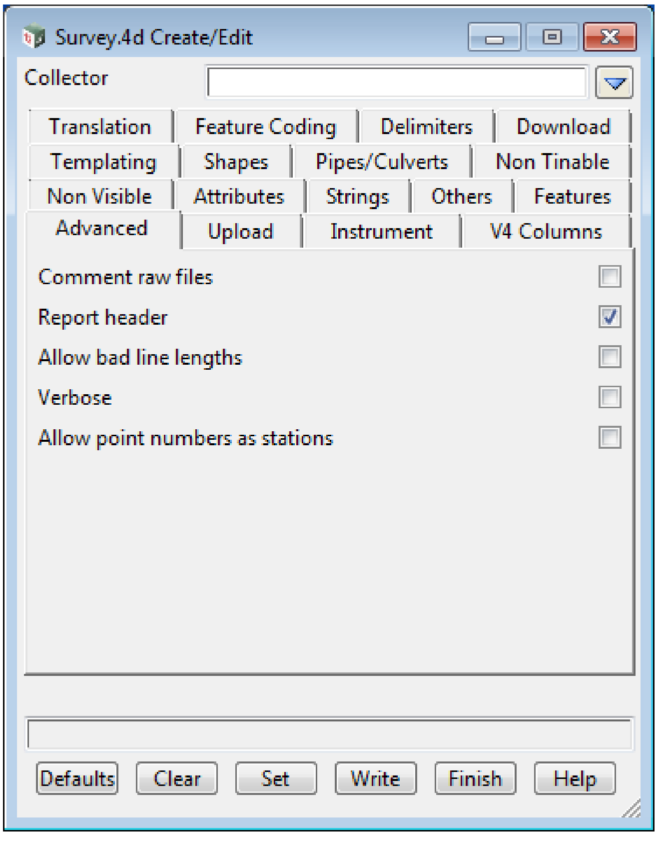
Parity	input	none	none, even, odd, mark, space

parity to use.

DTR/DSR, RTS/CTS, XON/XOFF,ACK/NAK tick boxes

flow control settings.

Advanced tab



Comment raw files tick box

if tick, include the data from the raw survey file as comments in the 12d field file.

Report header tick box

if tick, include some header information as comments in the 12d field file.

Allow bad line lengths tick box

if tick, ignore the fact that the length of the line from the data collector is incorrect. This may lead to other errors so it should be used sparingly.

Verbose tick box

if tick, then extra information is written to the Output window.

Allow point numbers as stations tick box

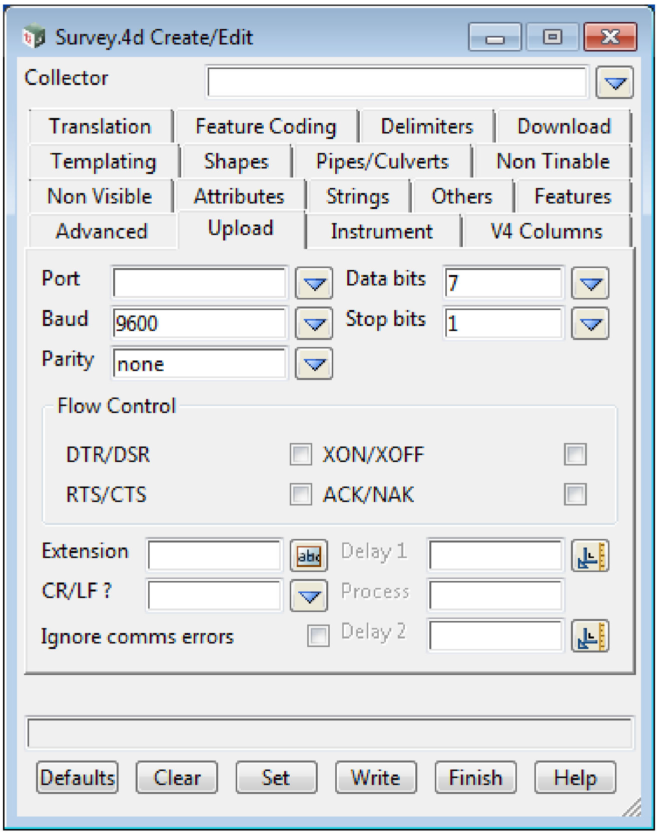
if tick, the point numbers for previous measurements can be used as stations for instrument setups, backsights, foresights and check measurements.

Note

When 12d Model starts up, it checks to see if an environment variable called DATA_COLLECTORS_4D exists and if it does, then the file it points to is used to provide the definitions for the data collectors.

If the environment variable is not set, then 12d Model searches for a file called **survey.4d** (this was **data_collectors_definitions** in V3.0) in the standard 12d Model search sequence for set up files.

Upload tab



Port input COM1 COM1 to COM4

port on the computer to use to communicate with the data collector for uploads.

Baud rate input 9600 110, 300 ... 256000

baud rate to use for the computer port.

Data bits input 7 5, 6, 7, 8

number of data bits to use.

Stop bits input 1 0, 1, 1.5, 2

number of stop bits to use.

Parity input none none, even, odd, mark, space

parity to use.

DTR/DSR, RTS/CTS, XON/XOFF,ACK/NAK tick boxes

flow control settings.

Extension input

ending of the file that will be uploaded. For example, typing in **sdr** means the file to upload ends in ".sdr"

CR/LF ? input cr, lf, crlf, lfcr,none

end of line characters - characters needed by the data recorder to denote the end of a line of data being uploaded to the data recorder.

Delay 1 input

the time between sending lines of data to the data recorder.

Process input

for some data recorders after a data line is sent, a text string needs to be sent to the data recorder to make it process the data line (e.g. GRE3 used a P).

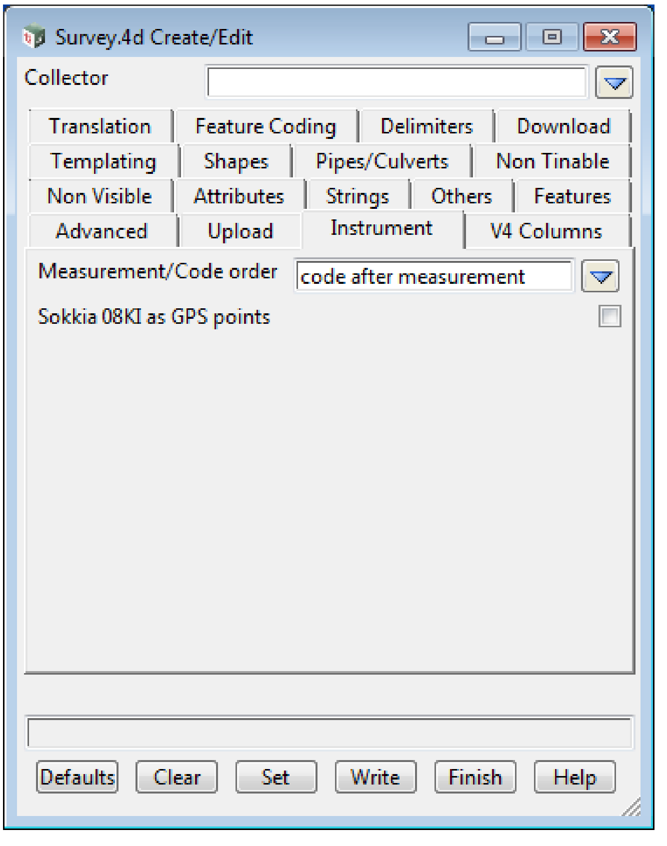
Delay 2 input

the time between sending the process string and sending the next data line.

Ignore comms errors tick box

if tick, ignore any communication errors whilst uploading to the data recorder.

Instrument tab



Measurement/code order input after measurement code before measurement

code after measurement

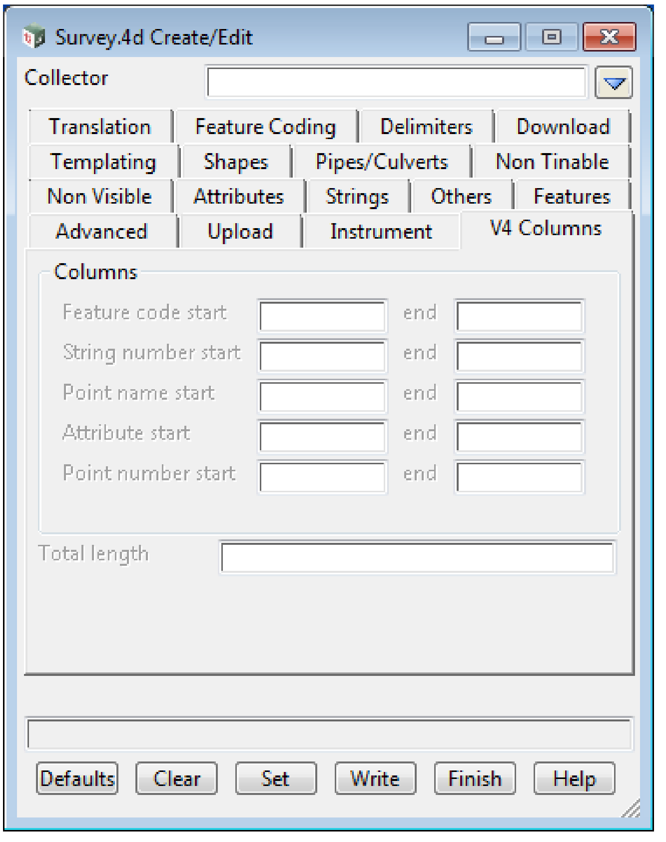
For Leica only where the feature code/string number record is separate to the measurement record. For all other data recorders the feature code block is in the same record as the measurement.

If 'code before measurement', then the feature code/ string number record for a measurement is entered before taking the measurement.
If 'code after measurement', then the feature code/ string number record for a measurement is entered after taking the measurement.

Sokkia 08KI as GPS points tick box

if tick, when converting reading an Sokkia SDR file, the "08KI" records are treated as points and no directly entered station co-ordinates.

V4 Columns tab



The V4 columns tab is for compatibility with ***12d Model*** V4 field files which were of fixed column widths.

Feature code start/end input 1/6

the start and end column position for the feature code in the field file record.

String number start/end input 7/9

the start and end column position for the string number in the field file record.

Point name/Attribute/Point number start/endinput 22/30, 10/16, 17/20

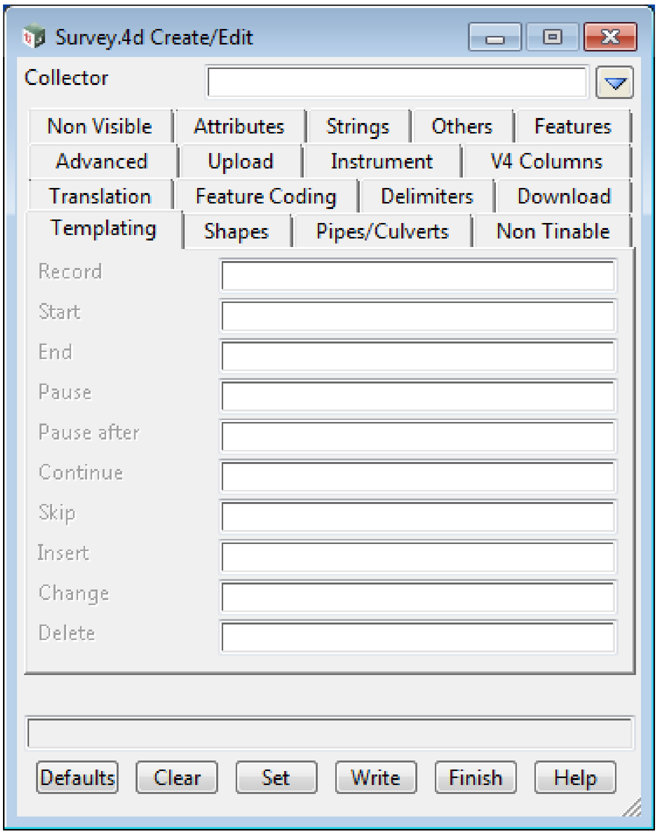
the start and end column position for the point name/Attribute/Point number in the field file record.

Total length input

the total length of line expected.

Templating tab

For information on field templating, see 36.4 Field Templates.



Record input XA text

characters to denote the start of recording a field template. For example XA starts recording for the default template, XA.road starts recording the field template called road. See 55 Start recording a field template.

Start input XB text

characters to denote the start of using a field template. For example XB starts using the blank field template, XB.road starts using the field template called 'road'. XB.road.zig starts using 'road' a zig-zag field template and begins as a zag. See 51 Start using an existing field template.

End input XC text

characters to denote stop using the current field template. See 52 Finish using a field template or finish recording a field template.

Pause input XD text

characters to denote temporarily stop using the current field template to take other measurements. The current point does not use the field template. See 53 Pause using the current field template.

Pause after input XD text

characters to denote temporarily stop using the current field template to take other measurements. The current point does use the field template. See 51 Start using an existing field template.

Continue input XE text

characters to denote continuing to use the current field template after a pause. See 54 Continue the current field template.

Skip input XF text

characters to denote that template points are to be skipped. For example XF means skip one field template point, XF.3 means skip three field template points. See 56 Skipping picking up points when using a field template.

Insert input XI text

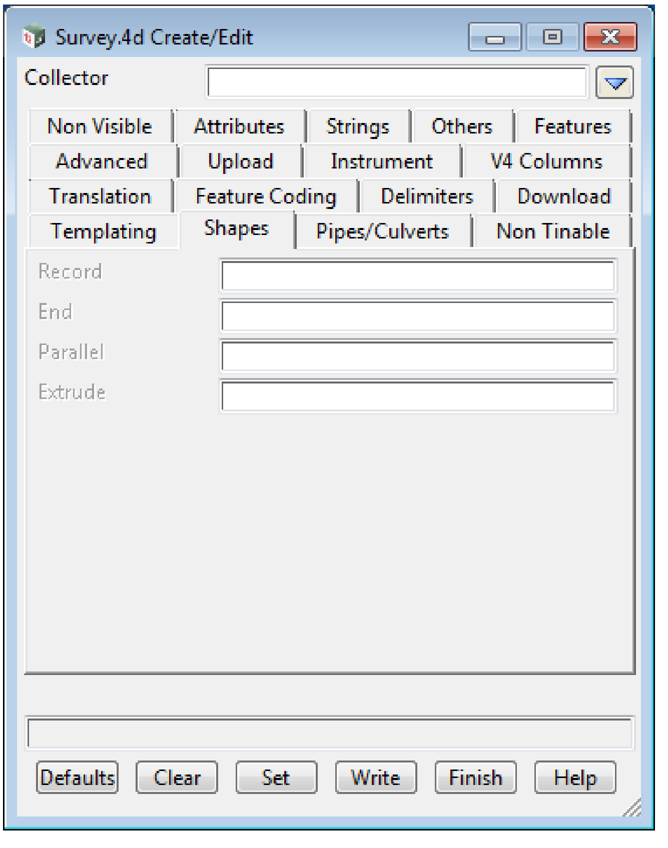
characters to denote that template points are to be inserted. For example XI.FE01 means insert a template point of FE01 into the template after the last measured template point. See 58 Insert points when using a field template - after the measurement of last point.

Delete input XQ text

characters to denote that template points are to be deleted. For example XQ.3 means delete the next 3 pts from the template after the last measured template point. See 57 Delete points on a field template - after the measurement of last point.

Shapes tab

For information on shapes, see 36.5 Shape field coding.



Record input text

characters to denote the start of recording a shape. For example if RS was used, the code RS starts recording for the default shape, RS.shape_name defines the recording of a shape of name shape_name. See 83 Start recording a shape - before the measurement.

End input text

characters to denote stop using the current shape. See 84 Finish using a shape definition or finish recording a shape - after the measurement.

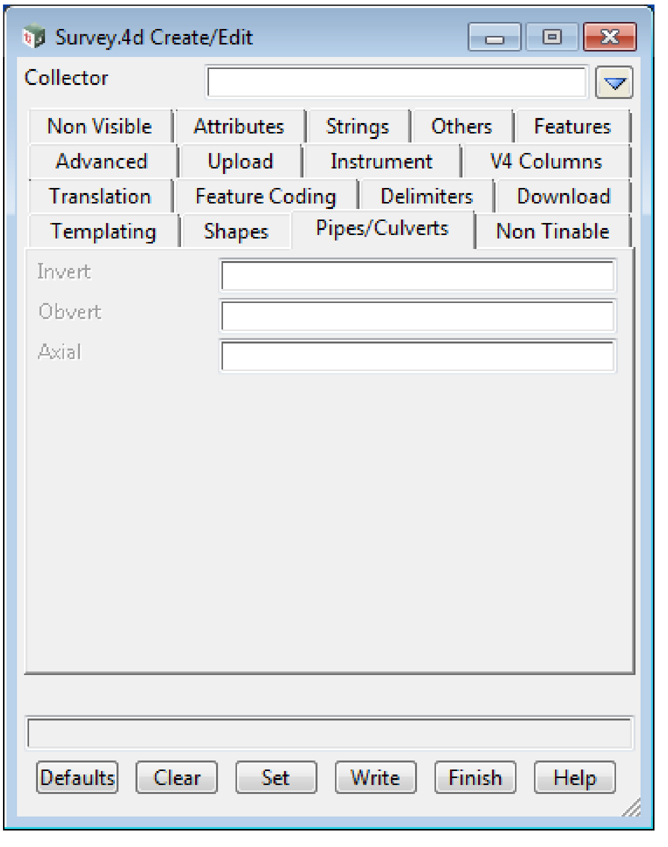
Parallel input text

*characters to denote the application of a shape by parallelling to the shape reference string. The code may be LK01*PS.shape_name*
Where the Parallel command is PS, the shape reference string is LK01 and the shape name = shape_name. See 85 Shape parallel.

Extrude input text

*characters to denote the application of a shape by extrusion to the shape reference string. The code may be LK01*ES.shape_name*
Where the Extrusion command is ES, the shape reference string is LK01 and the shape name = shape_name. See 86 Shape extrude.

Pipes/Culverts tab



Invert input I text

characters to denote that the measurement was the an invert level. See 80 Pipe invert point (bottom of the pipe).

Obvert input O text

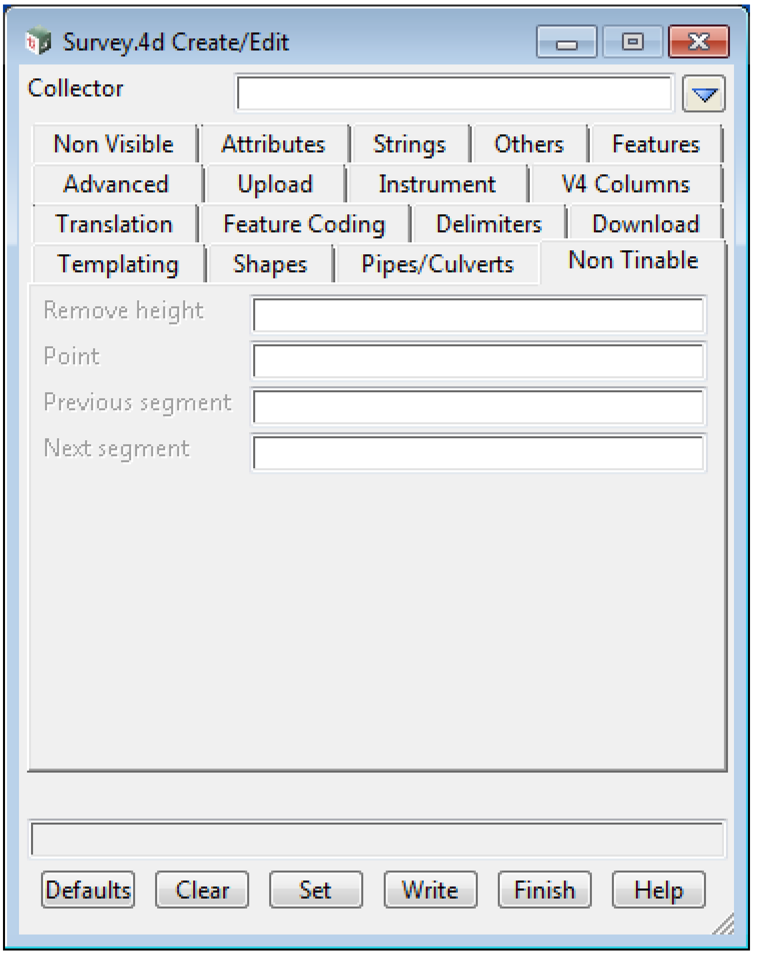
characters to denote that the measurement was the an obvert level. See 82 Pipe obvert point (top of the pipe).

Centre input C text

characters to denote that the measurement was the a centreline (axial) level. See 81 Pipe axial point (centre of the pipe).

Note that these commands can also denote the diameter of a pipe or width and height of a culvert. For example, I.3 denotes the measurement was an invert level and it was a pipe of diameter 0.3 units. I.3x.4 denotes the measurement was an invert level and it was a box culvert of width 0.3 and height 0.4 units.

Non Tinable tab



Remove height

characters to denote that the measurement has a null height. See 30 Remove height from a point - that is make it a null height.

Point

characters to denote that the measurement is a non-tinable point. See 40 Make a point non-tinable

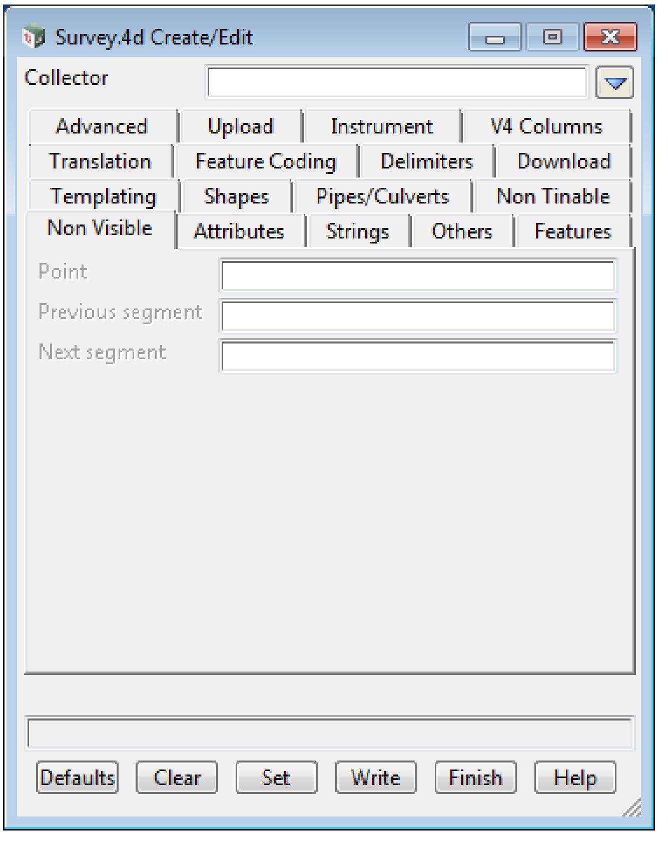
Previous segment

characters to denote that the next segment of the current string is non-tinable. See 38 Make the previous segment non-tinable.

Next segment

characters to denote that the previous segment of the current string is non-tinable. See 39 Make the next segment non-tinable.

Non visible tab



Point input text

characters to denote that the measurement is a non-visible point. That is, the point is not visible and is not included in a triangulation. See 109 Make a point invisible - after the measurement.

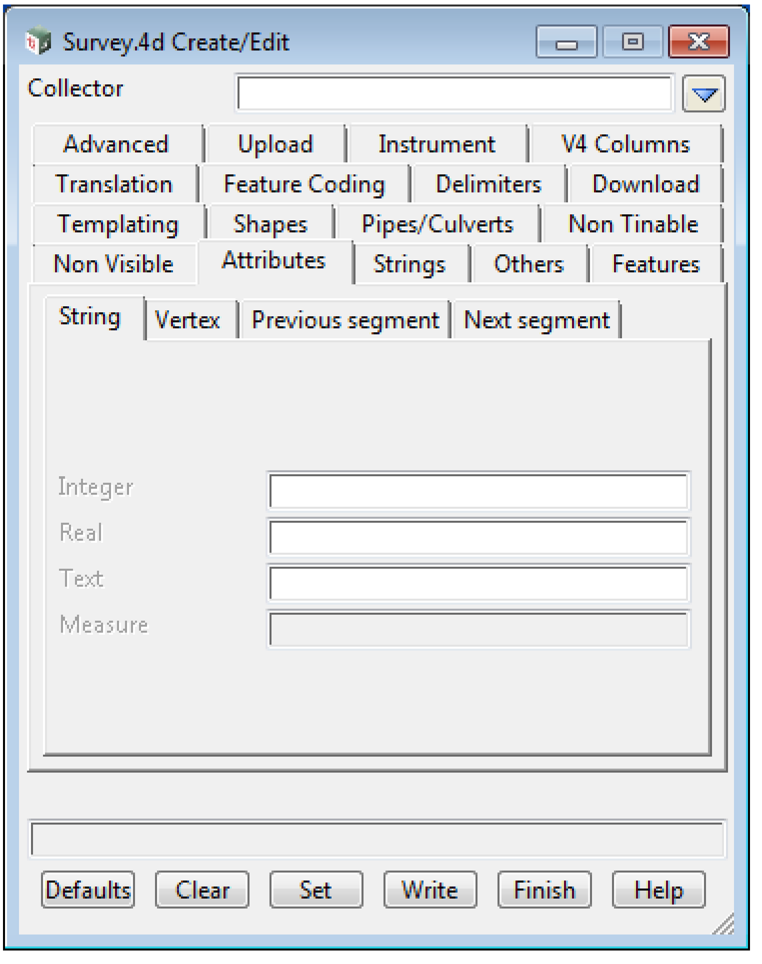
Previous segment input text

characters to denote that the measurement is the end of a non-visible segment. That is, the previous segment is not used as a breakline in a triangulation and it is not visible. See 107 Make the previous segment invisible - after the measurement.

Next segment input text

characters to denote that the measurement is the start of a non-visible segment. That is, the next segment is not used as a breakline in a triangulation and is not visible. See 108 Make the next segment invisible - after the measurement for the first point of the segment.

Attributes tab



String tab (Attributes)

Integer input text

characters to denote that the next text is to be taken as a user defined integer attribute of the current string. See 68 Add an integer user defined attribute to the current string.

Real input text

characters to denote that the next text is to be taken as a user defined real attribute of the current string. See 69 Add a real user defined attribute to the current string.

Text input text

characters to denote that the next text is to be taken as a user defined text attribute of the current string. See 70 Add text user defined attribute to the current string.

Measure input text

Vertex tab (Attributes)

Integer input text

characters to denote that the next text is to be taken as a user defined integer attribute of the current vertex. See 71 Add integer user defined attribute to the current point.

Real input text

characters to denote that the next text is to be taken as a user defined real attribute of the current vertex. See 72 Add real user defined attribute to the current point.

Text input text

characters to denote that the next text is to be taken as a user defined text attribute of the current vertex. See 73 Add text user defined attribute to the current point.

Measure input text

Previous segment tab (Attributes)

Integer input text

characters to denote that the next text is to be taken as a user defined integer attribute of the previous segment. See 77 Add integer user defined attribute to the previous segment.

Real input text

characters to denote that the next text is to be taken as a user defined real attribute of the previous segment. See 78 Add real user defined attribute for the previous segment.

Text input text

characters to denote that the next text is to be taken as a user defined text attribute of the previous segment. See 79 Add text user defined attribute to the previous segment.

Measure input text

Next segment tab (Attributes)

Integer input text

characters to denote that the next text is to be taken as a user defined integer attribute of the next segment. See 74 Add integer user defined attribute to the next segment.

Real input text

characters to denote that the next text is to be taken as a user defined real attribute of the next segment. See 75 Add real user defined attribute to the next segment.

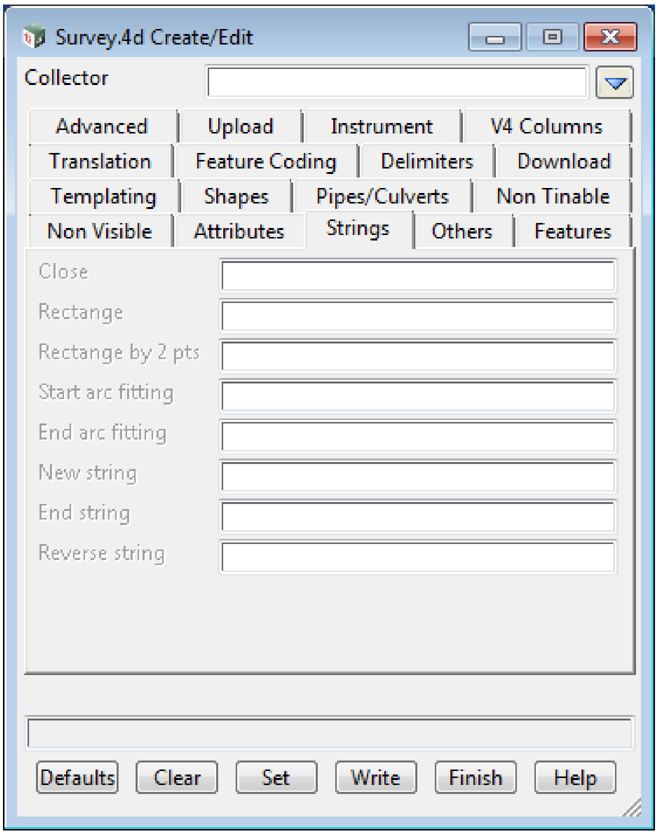
Text input text

characters to denote that the next text is to be taken as a user defined text attribute of the next segment. See 76 Add text user defined attribute to the next segment.

Measure input text

Strings tab

For information on the string commands, see 36.3 Field Coding.



Close input C text

characters to denote that the string is closed. That is the current measurement is joined to the first point of the string. See 20 Close string.

Rectangle input R text

characters to denote that the string is a rectangle. That is the current measurement and the previous two points of the string are three points of a parallelogram and the fourth point is automatically created (and given a null height). See 45 Make a parallelogram from the last three measurement points.

Rectangle by 2 pts input text

characters to denote that the string is a rectangle. That is the previous two points of the string define one side and a offset defines the opposite side. Positive being to the right. The created points are given null height. See 37 Rectangle by two points.

Start arc fitting input S text

characters to denote the start of arc fitting. See 61 Start of arc through sets of three points until end of string, or a 62 occurs.

End arc fitting input E text

characters to denote the end of arc fitting. See 62 End the arcs begun by a 61 command.

New string input ST text

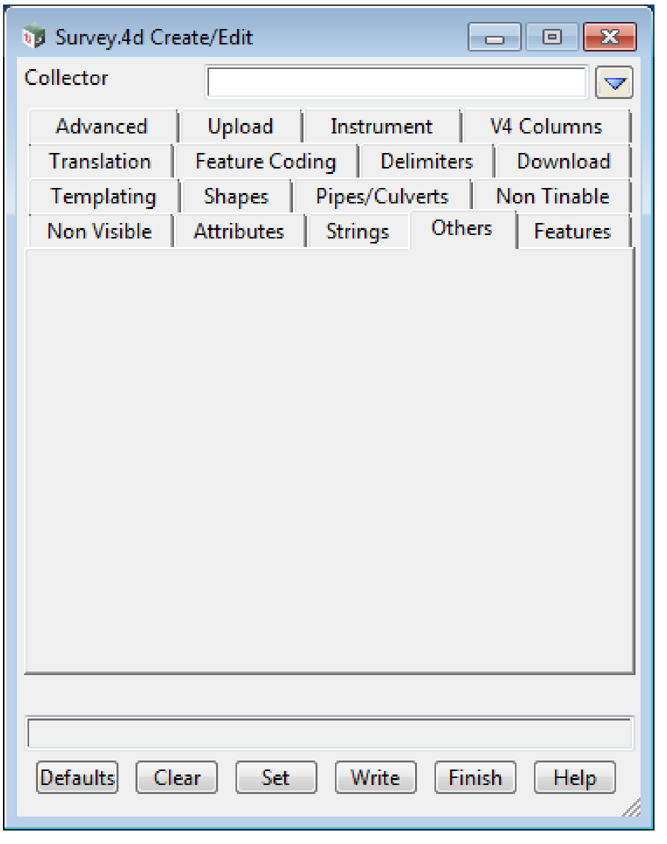
characters to denote the that the current measurement is the start of a new string, even if the string number has not changed. See 47 Start a new string using the same feature code and string number.

End string input text

characters to denote the that the current measurement is at the end of a string, even if the string number has not changed on the next measurement. See 48 End a string.

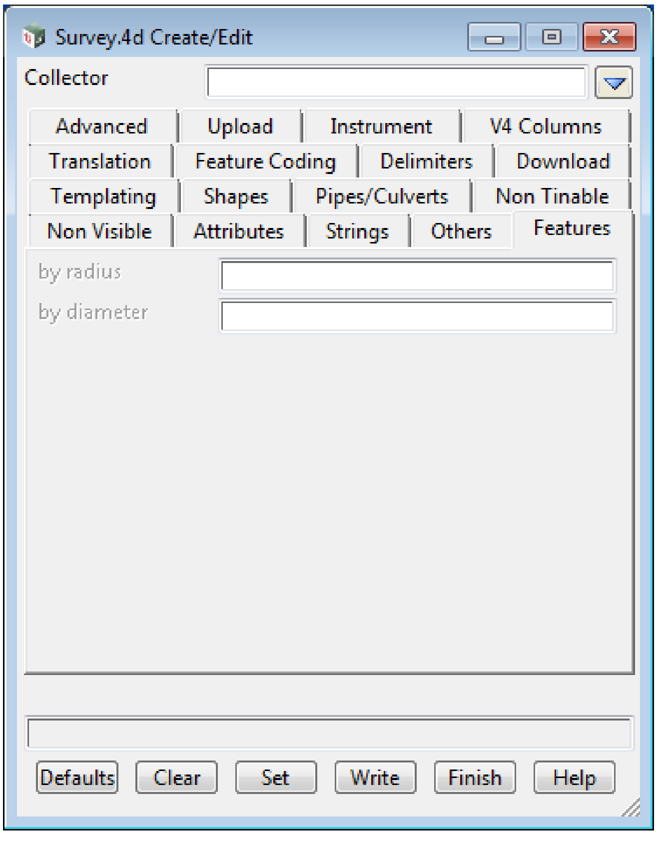
Others tab

Under development.



Features tab

For information on Feature string, see 36.3.7 Feature.



By radius input text

*characters to denote that the feature has a given radius. For example, if RA was used then TRE*RA2 would denote a feature code TRE with a radius of 2 units.*

There can be more than one set of characters that define 'By radius'. The sets are entered into the By radius field separated by a space. For example

RA RAD RD

would specify that RA and RAD and RD can be used to denote that the feature is defined by radius.

By diameter input text

*characters to denote that the feature has a given diameter. For example, if DI was used then TRE*DI2 would denote a feature code TRE with a diameter of 2 units.*

There can be more than one set of characters that define 'By diameter'. The sets are entered into the By diameter field separated by a space. For example

DI DIA DR

would specify that DI and DIA and DR can be used to denote that the feature is defined by diameter.

36.10 The 12d Field File Format

The detailed definition of the standardised 12d Field File will now be given. This is the file that all raw files are converted to before reduction in ***12d Model***.

A few definitions will be given, followed by the complete description of the 12d Field File.

The raw data files from different data collectors are all converted to the 12d field file format.

The 12d field file is text and has a name ending in **.fld**

Each line in the 12d field file is called a record and consists of an ***operation code*** (or *op code* for short) followed by zero or more tabs and pieces of information.

(a) zero fields

What actually follows the fixed header depends on the *op code*.

Hence the 12d field file record is:

op code
99

or

op code	tab	value	...	tab	value
02		abcS			fred

A number of conventions and definitions will be now given, followed by the complete description of the 12d Field File.

Important Note

Op codes are not the same as field codes. *Field codes* are entered by the surveyor in the field and may lead to one or more op codes being created in the 12d field file.

Please continue to the next section 36.10.1 Point Description.

36.10.1 Point Description

The *feature codes* and *string numbers* are entered by the surveyor whilst surveying in the field and the *point numbers* are usually automatically created by the surveying instrument but may sometimes be entered into the instrument by the surveyor.

Other information can also be entered by the surveyor using ***field codes*** and associated information. How each field code and associated information is entered in the field *depends on the data recorder* and the *coding convention* being used. This is set up in the *Data Collector definitions* in ***12d Model***.

The data file from the instrument or data collector (raw file) is first converted in 12d Model to the 12d field file ready for loading into a ***12d Model*** Survey Reduction function for reduction.

In most lines of the **12d field file**, a *feature code*, *string number*, *point number*, *point name* and *point text* are given and are stored next to each other separated by tabs.

The **feature code** and **string number** have already been described.

The **point number** is the EDM tacheometry measurement point number which is not normally entered by the user but is written in the raw data file by the data collector. This is stored as the point number for the vertex of the super string.

The **point name** is used to store the name of a station (see the next section on *named measurements*) and is supplied by the user with certain op codes. It is also used by some op codes that do not require an actual point name but have other special information that needs to be stored.

The **point text** is stored as vertex text for that vertex of the super string.

In the 12d field file, the *feature code*, *string number*, *point number*, *point name* and *point text* are given in this order and are separated by tabs. Each can be up to sixty-three characters in length. If the item is missing then a tab is still needed so there may be two or more sequential tabs.

For simplicity in describing the **12d field file**, the feature code, string number, point number, point name and point text are grouped together and called the **point description**.

Hence the **point description** consists of the following pieces of information separated by tabs:

- feature code (string name)
- string number
- point number
- point name
- point text

So it is convenient to think of the *point description* record as five columns of information:

feature code	string number	point number	point name	text
abc	01	1002	STN 4	freddie

Please continue to the next section 36.10.2 Measurements and Named Measurements.

36.10.2 Measurements and Named Measurements

The 12d field file allows five types of measurements that create points (vertices) in super strings. They are

(a) directly entered co-ordinates measurement

(b) EDM measurement (HA,VA,SD, + point_description.)

(c) EDM measurement VD (HA,HD,VD + point_description.)

(d) EDM measurement HT (HA,HD,HT, + point_description.)

(e) three hair stadia measurement

Each type of measurement creates a new point which is appended to the previous points with the same *feature code* and *string number*.

The last such measurement is referred to as the *current measurement point* or *current point* and the string it is appended to is the *current string*.

If a *point_name* exists in the *point_description* for any of the three types of measurements, then it is called a ***named measurement*** and a one point super string of name *point_name* is created and mapped using the Map File. The vertex text for the one point super string is the station prefix followed by *point_name*. The *point_name* is added to an internal list of named points for searching for co-ordinates.

Please continue to the next section 36.10.3 Existing Station Co-ordinates.

36.10.3 Existing Station Co-ordinates

When setting up a new instrument, measuring to a backsight, doing a check measurement or manually entering a bearing to use as the bearing datum difference, the *point_name* or point number from the *point_description* is used to specify a point. The (x,y,z) co-ordinates for *point_name* are found by searching in the following order:

The difference between *point names* and *point numbers* is that *point names* are usually given by the user and should be a unique identifier for a point whilst for that same physical point a number of measurements (and hence point numbers) may be assigned (usually by the data collector). This may be particularly true of control station measurements where measurements are made to a given point name but each measurement is given a different point number by the data collector.

In most instances, a measurement to a point has a point number (from the data collector) and 12d automatically gives it the same point name as it is rare to measure a non-control point more than once (the point name can be over ridden by the user).

The names allow the reduction routine to search for the details of that point (e.g. coordinates) to allow for the reduction of further measurements. The order in which this searching takes place is as follows:

First search the Control model (if it exists):

1. A search is made of the control model for a string whose name is the same as the specified *point name*. If a string is found, the first point of the string is used for the (x,y,z) co-ordinates.
2. A search is made of the control model for a vertex of a string whose point number is the same as the specified *point name*. If a vertex is found its (x,y,z) co-ordinates are used.
3. If only a *point number* was specified, a search is made of the control model for a vertex of a string whose point number is the same as the specified *point number*. If a vertex is found its (x,y,z) co-ordinates are used.

Next search the already entered directly entered co-ordinates (DEC) in the field file:

4. A search is made of previously entered directly entered co-ordinates in the field file for a directly entered co-ordinate whose *point name* is the same as the specified *point name*. If a DEC is found, its (x,y,z) co-ordinates are used.
5. A search is made of previously entered directly entered co-ordinates in the field file for a directly entered co-ordinate whose *point number* is the same as the specified *point name*. If a DEC is found, its (x,y,z) co-ordinates are used.
6. A search is made of previously entered directly entered co-ordinates in the field file for a directly entered co-ordinate whose *point number* is the same as the specified *point number*. If a DEC is found, its (x,y,z) co-ordinates are used.

Next search the previous measurements in the field file:

7. A search is made of previous measurements in the field file for a measurement whose *point name* is the same as the specified *point name*. If a measurement is found, its (x,y,z) co-ordinates are used.
8. A search is made of previous measurements in the field file for a measurement whose *point number* is the same as the specified *point name*. If a measurement is found, its (x,y,z) co-ordinates are used.
9. A search is made of previous measurements in the field file for a measurement whose *point number* is the same as the specified *point number*. If a measurement is found, its (x,y,z) co-ordinates are used.
10. or finally, the user is asked to type in the (x,y,z) co-ordinates in a **Survey Data Define Station** panel. If a model is specified in the Add to model field of the panel, then a new one point super string is created with the name *point_name*, and as the vertex text for the point, the Station label prefix field value followed by *point_name*.

For a summary of the **12d Field File Op Codes**, go to the section 36.10.5 Summary of 12d Field File Op Codes

For the full description of the **12d Field File Op Codes**, go to the section 36.10.4 Full Description of 12d Field File Op Codes

36.10.4 Full Description of 12d Field File Op Codes

For a summary of the **12d Field File Op Codes**, go to the section 36.10.5 Summary of 12d Field File Op Codes.

The record for each *op code* allowed in the 12d field file will now be described in detail.

For each *op code* record, two lines and a paragraph of description are given:

- The first line consists of the *op code* and a short description of the purpose of the code.
- The second line gives the full syntax of the record for that *op code*.
- The paragraph gives a detailed description of the *op code* record.

Optional information is enclosed in the square brackets **[ ]**

All **angles** in the 12d field file are given in ***decimal*** degrees.

Important Note

Op codes are not the same as field codes. *Field codes* are entered by the surveyor in the field and may lead to one or more op codes being created in the 12d field file.

Op Code **Description of Record**

1 Four 10 character information blocks

01 block1 block2 block3 block4

This information is ignored.

2 Directly entered coordinate measurement

02 Point_description X Y Z

A *measurement* point is created with the f*eature code* and *string number* from the *point_description* and given (x, y,z) co-ordinates. No reduction is needed.

The *point_number* and *text* from the *point_description* are recorded as the point number and text for that vertex of the super string.

If a *point_name* exists in the *point_description*, then it is a ***named measurement*** and a 4d point string of name *point_name* is created and mapped using the Map File. The 4d text is the station prefix followed by *point_name*. The *point_name* is added to the internal list of named points for searching for co-ordinates.

3 New instrument point

03 Point_description instrument_height

Setting up an instrument at the point with name given in the *point_name* section of the *point_description*. The (x,y,z) co-ordinates for *point_name* are found by first searching the control model, then the list of previously named point in the reduction, point numbers of previous measurements and finally if *point_name* is still not found, the user is asked to type in the (x,y,z) co-ordinates. A record is written to the report file.

4 Measurement to backsight

04 Point_description horizontal_circle vertical_circle slope_distance azimuth

Measurement to a backsight whose name is given in the *point_name* section of the *point_description.* If the *Display panel for backsights* field in the **Survey Data Reduce** panel is tick, then the **Survey Data Bearing Datum Difference** panel shows the *bearing datum difference* and the *horizontal distance* *difference*. A record is written to the report file. The units for *horizontal_circle* and *vertical_circle* are decimal degrees. The azimuth may be specified where no coordinate for the backsight point exists in decimal degrees.

5 New target height

05 Target_height

Set a new target height.

6 Check measurement

06 Point_description horizontal_circle vertical_circle slope_distance

A check measurement is made to the station given in the *point_name* section of the *point_description*. A two point super string (with name *point_name*) from the instrument point to the measured point is created in the default model for the check measurement. The instrument point name, the station name and the differences between the measurement point co-ordinates and the station co-ordinates are written as text along the super string. The differences between the measurement and the known point is also written to the report file. The units for *horizontal_circle* and *vertical_circle* are decimal degrees.

7 Measurement - HA, VA, SD

07 Point_description horizontal_circle vertical_circle slope_distance

Measurement made by the instrument. A *measurement* point is created with the *feature code* and *string number* from the *point_description*. The units for *horizontal_circle* and *vertical_circle* are decimal degrees.

The *point_number* and *text* from the *point_description* are recorded as the point number and text for that vertex of the super string.

If a *point_name* exists in the *point_description*, then it is a ***named measurement*** and a 4d point string of name *point_name* is created and mapped using the Map File. The 4d text is the station prefix followed by *point_name*. The *point_name* is added to the internal list of named points for searching for co-ordinates.

9 Scale factor for subsequent distances

09 Scale_factor

Scale factor to apply to subsequent slope distances.

10 Three hair stadia measurement

10 Point_description horizontal_circle vertical_circle bottom middle top

Manual measurement. A *measurement* point is created with the *feature code* and *string numbe*r from the *point_description*. The units for *horizontal_circle* and *vertical_circle* are decimal degrees.

The *point_number* and *text* from the *point_description* are recorded as the point number and text for that vertex of the super string.

If a *point_name* exists in the *point_description*, then it is a ***named measurement*** and a 4d point string of name *point_name* is created and mapped using the Map File. The 4d text is the station prefix followed by *point_name*. The *point_name* is added to the internal list of named points for searching for co-ordinates.

11 Measurement - HA, HD, Height

11 Point_description horizontal_circle horizontal_distance height

Measurement made by the instrument. A *measurement* point is created with the *feature code* and *string number* from the *point_description*. The unit for *horizontal_circle* is decimal degrees.

The *point_number* and *text* from the *point_description* are recorded as the point number and text for that vertex of the super string.

If a *point_name* exists in the *point_description*, then it is a ***named measurement*** and a 4d point string of name *point_name* is created and mapped using the Map File. The 4d text is the station prefix followed by *point_name*. The *point_name* is added to the internal list of named points for searching for co-ordinates.

12 Measurement - HA, HD, Height difference

11 Point_description horizontal_circle horizontal_distance height_difference

Measurement made by the instrument. A *measurement* point is created with the *feature code* and *string number* from the *point_description*. The unit for *horizontal_circle* is decimal degrees.

The *point_number* and *text* from the *point_description* are recorded as the point number and text for that vertex of the super string.

If a *point_name* exists in the *point_description*, then it is a ***named measurement*** and a 4d point string of name *point_name* is created and mapped using the Map File. The 4d text is the station prefix followed by *point_name*. The *point_name* is added to the internal list of named points for searching for co-ordinates.

15 Vertical circle correction

09 Vertical_circle_in_decimal_degrees

The *vertical_circle_in_decimal_degrees* is **subtracted** from the vertical circle value in any measurements. The units for *vertical_circle_in_decimal_degrees* is decimal degrees.

16 Multiply coded point

16 Point_description

Additional coding for the current measurement point created by op codes 02, 07 or 10. A new measurement point is created at the same position as the current measurement point but with the *feature code* and *string number* from the *point_description* for this op code.

The *point_number* and *text* from the *point_description* are recorded as the point number and text for that vertex of the super string.

If a *point_name* exists in the *point_description*, then it is a ***named measurement*** and a 4d point string of name *point_name* is created and mapped using the Map File. The 4d text is the station prefix followed by *point_name*. The *point_name* is added to the internal list of named points for searching for co-ordinates.

17 Arc through previous three points

17 [Point_description]

If no *point_description* is given, then the current measurement point and the two previous points with the same feature code and string number as the current measurement point, are joined by an arc. If there is less than three such points, no arc is fitted.

If a *point_description* exists, then either the *feature code* and *string number* or the *point number* section of the *point_description* can be used.

If the *feature code* and *string number* from the *point_description* exist, the last three previous measurement points of the same *feature code* and *string number* are joined by an arc. If the current measurement point has that feature code and string number, then it is the third of the three points used. If there is less than three points, no arc is fitted.

If *Point_number* exists, then the *feature code* and *string number* are taken from the previous measurement point **with** that point number. That point and the two measurement points previous to the predefined point of the same *feature code* and *string number*, are joined by an arc. If there is less than three points, no arc is fitted

See 36.3.9 Arcs Through Points

18 Circle Feature

18 Radius

Creates a feature string with the given radius and centred on the current measurement point.

See 36.3.7 Feature

20 Close string

20 [Point_description]

If no *point_description* is given, the current string is closed.

If a *point_description* exists, then either the *feature code* and *string number* or the *point number* section of the *point_description* can be used.

If the *feature code* and *string number* from the *point_description* exist, the last previous string with that *feature code* and *string number* is closed.

If the *point number* from the *point_description* exists, then the string containing that point number will be closed.

See 36.3.4 Close String

21 Join last points of strings

21 Feature_code string_number_1 string_number_2

In the final reduction, the last point of the string with the given *feature_code* and *string_number_1* is joined to the last point of the string with given *feature_code* and *string_number_2*. The created string has the given *feature_code* (no string number is needed since it is the final phase of reduction when the string numbers are dropped).

See 36.3.8 Joining Strings

22 Join first to last point of strings

22 Feature_code string_number_1 string_number_2

In the final reduction, the first point of the string with the given *feature_code* and *string_number_1* is joined to the last point of the string with given *feature_code* and *string_number_2*. The created string has the given *feature_code* (no string number is needed since it is the final phase of reduction when the string numbers are dropped).

See 36.3.8 Joining Strings

23 Join first points of strings

23 Feature code string number_1 string number_2

In the final reduction, the first point of the string with the given *feature code* and *string number 1* is joined to the fist point of the string with given *feature code* and *string number 2*. The created string has the given *feature code* (no string number is needed since it is the final reduction when the string numbers are then dropped).

See 36.3.8 Joining Strings

28 Add text to the string name

28 [Point_description] Text

Text is appended to the string name. For example, if 1.200 is entered, “1.200” is appended to the string name.

If no *point_description* is given, *Text* is appended to the string name of the current string.

If a *point_description* exists, then either the *feature code* and *string number* or the *point number* section of the *point_description* can be used.

If the *feature code* and *string number* exist, then the last previous string with that *feature code* and *string number* has *Text* appended to the string name.

If the *point number* exists, then the string containing that point number has *Text* appended to the string name.

29 **Note or memo**

29 Comments

Any text may be entered and will be added to the check measurements model at the position of the current measurement point.

30 **Remove height from a point - that is make it a null height**

30 [Point_description]

If no *point_description* is given, the height of the current measurement point is set to null.

If a *point_description* exists, then either the *feature code* and *string number* or the *point number* section of the *point_description* can be used.

If the *feature code* and *string number* exist*,* then the height of the last point of the previous string with that *feature code* and *string number* is set to null.

If the *point number* exists*,* then the height of the point with that point number is set to null.

37 **Rectangle by two points**

37 [Point_description] offset_in_metres

The rectangle is defined by two points (reference side) and a offset.

If a positive offset value is given, two points will be created to the right of the reference side.

If a negative offset value is given, two points will be created to the left of the reference side.

If no *point_description* is given, the two new points will be joined to the given points in a closed rectangular string, and will have the same feature code as the points given.

If the *feature code* and *string number* exist*,* then a search is made for the last occurrence of two points with the same *feature code* and *string number*. If found, then these points are used to define the reference side of the rectangle.

If the *point number* exists*,* then a search is made for the last occurrence of two points with the same *feature code* and *string number* as the point given by the point number. If found, then these points are used to define the reference side of the rectangle.

Two consecutive rectangles are unable to be defined side by side. In other words if the two points given are part of string of greater than two vertices, the command will only work for sets of two points that are exclusively defined. i.e. For a 5 point string, a rectangle can be defined by points 1 and 2, and 4 and 5.

See 36.3.6 Rectangle by 2 Points

38 **Make the previous segment non-tinable**

38 [Point_description]

If no *point_description* is given, the previous segment containing the current measurement point is set to non-tinable. That is, it will not be treated as a breakline in triangulations.

If a *point_description* exists, then either the *feature code* and *string number* or the *point number* section of the *point_description* can be used.

If the *feature code* and *string number* exist*,* then the last segment of the previous string with that *feature code* and *string number* is set to non-tinable.

If the *point number* exists*,* then the segment containing the point with that point number as an end point, is set to non-tinable.

39 **Make the next segment non-tinable**

39 [Point_description]

If no *point_description* is given, the next segment containing the current measurement point as a starting point is set to non-tinable. That is, it will not be treated as a breakline in triangulations.

If a *point_description* exists, then either the *feature code* and *string number* or the *point number* section of the *point_description* can be used.

If the *feature code* and *string number* exist, then the segment that is created in the future from the last point of the previous string with that *feature code* and *string number* is set to non-tinable.

If the *point number* exists, then the segment containing the point with that point number as a start point, is set to non-tinable.

40 Make a point non-tinable

40 [Point_description]

If no *point_description* is given, the current measurement point is set to non-tinable. That is, it will not be included in triangulations.

If a *point_description* exists, then either the *feature code* and *string number* or the *point number* section of the *point_description* can be used.

If the *feature code* and *string number* exist, then the last point of the previous string with that *feature code* and *string number* is set to non-tinable.

If the *point number* exists, then the point with that point number is set to non-tinable.

41 Add additional text for the current measurement point

41 Text

The given *text* is added to the end of any existing text for the current measurement point. Any spaces from column four onwards will be part of the text.

42 Add a radial offset

42 [Point_description]   Radial_offset_in_metres

The *radial_offset_in_metres* is used to adjust the position of the specified point by a plan distance from the specified points original position, along the plan line joining the current station to the specified point. A positive offset is away from the station, negative is toward the station.

If no *point_description* is given, the offset is used to adjust the position of the current measured point.

If a *point_description* exists, then either the *feature code* and *string number* or the *point number* section of the *point_description* can be used.

If the *feature code* and *string number* exist, then the last point of the previous string with that *feature code* and *string number* is adjusted.

If the *point number* exists, then the point with that point number is adjusted.

See 36.3.2 Offsets

43 Add a tangential offset

43 [Point_description]   Tangential_offset_in_metres

The *tangential_offset_in_metres* is used to adjust the position of the specified point by a plan distance from the specified points original position, at rights angles to the plan line joining the current station to the specified point. A negative offset is to the left (looking from the station), and positive is to the right (looking from the station).

If no *point_description* is given, the offset is used to adjust the position of the current measured point.

If a *point_description* exists, then either the *feature code* and *string number* or the *point number* section of the *point_description* can be used.

If the *feature code* and *string number* exist, then the last point of the previous string with that *feature code* and *string number* is adjusted.

If the *point number* exists, then the point with that point number is adjusted.

See 36.3.2 Offsets

44 Add a height offset

44 [Point_description] Height_offset_in_metres

If the height of the specified point is not null, then the *height_offset_in_metres* adjusts the height of the point. A positive offset adds to the height, a negative offset reduces the height.

If no *point_description* is given, the offset is used to adjust the position of the current measured point.

If a *point_description* exists, then either the *feature code* and *string number* or the *point number* section of the *point_description* can be used.

If the *feature code* and *string number* exist, then the last point of the previous string with that *feature code* and *string number* is adjusted.

If the *point number* exists, then the point with that point number is adjusted.

See 36.3.2 Offsets

45 Make a parallelogram from the last three measurement points

45 [Point_description]

If no *point_description* is given, the current measurement point and the two previous points from the current string are used and a fourth point is created to form a parallelogram (squashed rectangle) and the height of the fourth point is set to null. The string is then closed.

If a *point_description* exists, then either the *feature code* and *string number* or the *point number* section of the *point_description* can be used.

If the *feature code* and *string number* exist, the last three points with that *feature code* and *string number* are used and a fourth point is created to form a parallelogram (squashed rectangle) and the height of the fourth point is set to null. The string is then closed.

If the *point number* exists, then the *feature code* and *string number* of the point with that point number are used and processed as above. Note that the point with the point number is not necessarily used.

See 36.3.5 Rectangle

46 Make the string a breakline or not

46 [Point_description] [mode]

The *point_description* is used to select a string and the *mode* is used specify if the string is a breakline or not.

point_description:

If no *point_description* is given, the current string is selected.

If a *point_description* exists, then either the *feature code* and *string number* or the *point number* section of the *point_description* can be used.

If the *feature code* and *string number* exist, the last string with that *feature code* and *string number* is selected.

If the *point number* exists, then the string containing the point with that point number is selected.

***mode*:**

If no *mode* is given, the selected string is set as a point string (that is, not a breakline).

If *mode* is given, then

if *mode* is 0, the selected string is set to a point string and hence is not a breakline.

if *mode* is 1, the selected string is set to a *line* string and is therefore a breakline

47 Start a new string using the same feature code and string number

47 [Point_description]

If no *point_description* is given, the current string is terminated (without including the current measurement point) and the current measurement point becomes the first point of a new string with the same feature code and string number.

If a *point_description* exists, then either the *feature code* and *string number* or the *point number* section of the *point_description* can be used.

If the *feature code* and *string number* exist, then the last point of the previous string with that feature code and string number becomes the first point of a new string with the same *feature code* and *string number*.

If the *point number* exists, then the previous string containing the point with that point number is terminated *before* the point number point, and the point becomes the first point of a new string with the *same feature code* and *string number*.

See 36.3.3 Start New String

48 End a string

48 [Point_description]

If no *Point_description exist*s, the current string is terminated (including the current measurement point).

If *Feature_code* and *String_number* exist, then the last point of the previous string with that feature code and string number becomes the last point of that string.

If *Point_number* exists, then the previous string containing the point with that point number is terminated *after* the point number point

50 Specify the bearing to correct for true north - used as the bearing datum difference

50 Point_description bearing_in_decimal_degrees

The *bearing_in_decimal_degrees* is used as the bearing datum difference for the current instrument set up. The *point_name* in the *point_description* and the *bearing_in_decimal_degrees* are written to the report file.

51 Start using an existing field template

51 Template_name zig_zag_mode

Start using the field template *Template_name.* If *Template_name* is blank, the default field template is used.
If *mode* is "for", then the field template is used as a *forward* template.
"rev", then the field template is used as a *reverse* template.
"zig", then the field template is used as a zig_zag template and is used in the *forward* definition direction first (that is starts on a zig).
"zag", then the template is used as a zig_zag template and is used in the *reverse* direction first (that is, starts on a zag).
If *mode* is blank, or anything other than "for", "rev", or "zag" then the field template is used as a zig-zag template starting on a *zig*.

See 36.4 Field Templates

52 Finish using a field template or finish recording a field template

52

Stops using the current field template or stops recording a field template.

See 36.4 Field Templates

53 Pause using the current field template

53

Pause using the current field template or defining a field template, until a continue field template (54) or a finish field template (52) code is given.

See 36.4 Field Templates

54 Continue the current field template

54

Continue using or defining the current field template, which has been stopped by a *Pause* field template command (53). The *Continue* command only needs to be given once and applies to all following measurements until another *Pause* or *Finish* command is given.

See 36.4 Field Templates

55 Start recording a field template

55 [Template_name]

Start recording a field template with the name *Template_name*. If *Template_name* is blank, then it is the default field template that is defined. The *feature_code* and *string_number* of the following measurements until a *Finish* code (52) are stored as the field template. There is no limit to the number of points in a field template.

See 36.4 Field Templates

56 Skipping picking up points when using a field template

56 [num_skipped_points]

Allows the user to skip picking up one or more points from the field template currently being used. The next measurement takes the *feature_code* and *string_number* from the next point of the field template definition. If *num_skipped_points* is missing, then only one point is skipped otherwise *num_skipped_points are skipped.*

See 36.4.4 Skipping Field Template Points

57 Delete points on a field template - after the measurement of last point

57 [num_points_to_delete]

Allows the user to delete one or more points from the field template currently being used. The *next measurement* takes the *feature_code* and *string_number* from the next point of the field template definition.

See 36.4 Field Templates

58 Insert points when using a field template - after the measurement of last point

58 Feature_code String_number Multiple_code_flag Insert_special_flag

Allows the user to insert points into the field template currently being used, or give an existing point a multiple code.

If the *Multiple_code_flag = 1, then the feature code will be added to the previous defined template point else if Multiple_code_flag = 0 (default), it will be added to the template as a separate point.*

If the insert is done at the end of a section and the *Insert_special_flag = 1 the point will be added to the end of the current template section else it will be at the start of the next section.*

The *next measurement* takes the *feature_code* and *string_number* from the next point of the field template definition.

See 36.4.5 Insert Template Points or Insert Multiple Codes

60 Arc through next three points

60 [Point_description]

If no *point_description* is given, an arc is inserted through the current measurement point and the next two measured points with the same feature code and string number as the current measurement point. If there is less than three points, no arc is fitted.

If a *point_description* exists, then either the *feature code* and/or *string number* and/or the *point number* section of the *point_description* can be used.

If the *feature code* or *string number* from the *point_description* exist, a search is made for a previously defined measurement with the same feature code or string number. An arc is inserted through this previous measurement and the next two measured points following this previous measurement with the same feature code and string number, as given in *point_description*. If the current point has that feature code and string number, then it is the first of the three points. If there is less than three points, no arc fitted.

If the *point number exists,* then the *feature code* and *string number* are taken from the previous measurement point **with** that point number, and an arc is inserted through that point and the next two measurement points with the same *feature code* and *string number*. If there is less than three points, no arc is fitted

See 36.3.9 Arcs Through Points

61 Start of arc through sets of three points until end of string, or a 62 occurs

61 [Point_description]

If no *point_description* is given, arcs are inserted through the following sets of measurement points with the same feature code and string number as the current measurement point. The current measurement point is the first of the points.

The arcs are fitted as follows - the first arc is fitted through points one, two and three, the next arc through points three, four and five etc. If the current point has that feature code and string number, then it is the first of the points. If there is less than three points, then no arc is fitted.

If a *point_description* exists, then either the *feature code* and *string number* or the *point number* section of the *point_description* can be used.

If the *feature code* and *string number exist*, a search is made for a previously defined measurement with the same feature code or string number. An arc is inserted through the following measured points with the same feature code and string number as given in *point_description*. If the current point has that feature code and string number, then it is the first of the points.

If the *point number* exists*,* then the *feature code* and *string number* are taken from the previous measurement point **with** that point number, and arcs are inserted through that point and the following measured points with the same *feature code* and *string number*.

See 36.3.9 Arcs Through Points

62 End the arcs begun by a 61 command

62 [Point_description]

If no *point_description* is given, then the fitting of arcs through the points of the current string is stopped. The current measurement point is the last of the points used in the arc fitting.

If a *point_description* exists, then either the *feature code* and *string number* or the *point number* section of the *point_description* can be used.

If the *feature code* and *string number* from the *point_description* exist, then the fitting of arcs through the points of the previous string with the same *feature code* and *string number* is stopped. If the current measurement point has that feature code and string number, then it is the last point used in the arc fitting.

If the *point number* from the *point_description* exists*,* then the point with that point number is the last point used in the arc fitting.

If ***12d Model*** encounters an *End Arcs (62*) but no *Start Arcs through sets of three points* (61) command for the string, then a *Start Arcs through sets of three points* (61) is assumed to apply at the beginning of the string and hence arc fitting will be applied to the entire string.

See 36.3.9 Arcs Through Points

There are op codes for adding user defined attributes to:

(a) the current string being measured (i.e. the string containing the current measurement point)

(b) the current measurement point

(c) the next segment from the current measurement point (i.e. the segment joining the current measurement point and the *next* measured point *of the same feature code and string number*)

or

(d) the previous segment to the current measurement point (i.e. the segment joining the current measurement point to the previous measured point *of the same feature code and string number*).

If there is no name for the attribute (name is just spaces or a tab), then the attribute is *unnamed*. The attributes are coded in the following way:

68 Add an integer user defined attribute to the current string

68 Name Integer

Add an user defined integer attribute to the current string.

69 Add a real user defined attribute to the current string

69 Name Real

Add a real (floating point) user defined attribute to the current string.

70 Add text user defined attribute to the current string

70 Name Text

Add a text user defined attribute to the current string.

71 Add integer user defined attribute to the current point

71 Name Integer

Add an integer user defined attribute to the current measurement point.

72 Add real user defined attribute to the current point

72 Name Real

Add a real (floating point) user defined attribute to the current measurement point.

73 Add text user defined attribute to the current point

73 Name Text

Add a text user defined attribute to the current measurement point.

74 Add integer user defined attribute to the next segment

74 Name Integer

Add an integer user defined attribute to the next segment from the current measurement point.

75 Add real user defined attribute to the next segment

75 Name Real

Add a real (floating point) user defined attribute to the next segment from the current measurement point.

76 Add text user defined attribute to the next segment

76 Name Text

Add a text user defined attribute to the next segment from the current measurement point.

77 Add integer user defined attribute to the previous segment

77 Name Integer

Add an integer user defined attribute to the previous segment for the current measurement point.

78 Add real user defined attribute for the previous segment

78 Name Real

Add a real (floating point) user defined attribute to the previous segment for the current measurement point.

79 Add text user defined attribute to the previous segment

79 Name Text

Add a text user defined attribute to the previous segment for the current measurement point.

In addition, extra codes allow special ***12d Model*** pipe strings to be coded in the field

80 Pipe invert point (bottom of the pipe)

80 [Point_description]

If no *point_description* is given, the current measurement point is on the invert (bottom) of a pipe. This is the default for measurements to points on pipe strings. If the point is not part of a pipe string, it is ignored.

If a *point_description* exists, then either the *feature code* and *string number* or the *point number* section of the *point_description* can be used.

If the *feature code* and *string number* exist, the last point of the previous string with the same *feature code* and *string number* as given in *point_description* is on the invert (bottom) of a pipe. If the point is not part of a pipe string, it is ignored.

If the *point number* exists, then the point with that point number is on the invert (bottom) of a pipe. If the point is not part of a pipe string, it is ignored.

81 Pipe axial point (centre of the pipe)

81 [Point_description]

If no *point_description* is given, the current measurement point is on the axis (centre) of a pipe. If the point is not part of a pipe string, it is ignored.

If a *point_description* exists, then either the *feature code* and *string number* or the *point number* section of the *point_description* can be used.

If the *feature code* and *string number* exist, the last point of the previous string with the same *feature code* and *string number* as given in *point_description* is on the axis (centre) of a pipe. If the point is not part of a pipe string, it is ignored.

If the *point number* exists, then the point with that point number is on the axis (centre) of a pipe. If the point is not part of a pipe string, it is ignored.

82 Pipe obvert point (top of the pipe)

82 [Point_description]

If no *point_description* is given, the current measurement point is on the obvert (top) of a pipe. If the point is not part of a pipe string, it is ignored.

If a *point_description* exists, then either the *feature code* and *string number* or the *point number* section of the *point_description* can be used.

If the *feature code* and *string number* exist, the last point of the previous string with the same *feature code* and *string number* as given in *point_description* is on the obvert (top) of a pipe. If the point is not part of a pipe string, it is ignored.

If the *point number* exists, then the point with that point number is on the obvert (top) of a pipe. If the point is not part of a pipe string, it is ignored.

83 Start recording a shape - before the measurement

83 [Shape_name]

Start recording a shape with the name *Shape_name*. If *Shape_name* is non-blank, then the default field Shape is defined by the *feature_code* and *string_number* of the following measurements until a *Finish* code (84) are stored as the shape. There is no limit to the number of points in a shape.

See 36.5 Shape field coding

84 Finish using a shape definition or finish recording a shape - after the measurement

Stops using the current shape or stops recording a shape.

See 36.5 Shape field coding

85 Shape parallel

85 [Shape_name] [Point_description]

Takes all the points on the defined shape of *Shape_name* and parallels them the entire length of the string. Once paralleled, a number of strings are created.

If *Feature_code* and *String_number* exist, the last string with the same *feature code* and *string number* has the shape applied to the entire length of the string.

If *Point_number* exists, then the string containing that point number has the shape applied to the entire length of the string.

See 36.5 Shape field coding

86 Shape extrude

86 [Shape_name] [Point_description]

Takes the defined shape of *Shape_name* and extrudes it along the entire length of the string. Once extruded, only one strings is created which contains all the shape information.

If *Feature_code* and *String_number* exist, the last string with the same *feature code* and *string number* has the shape applied to the entire length of the string.

If *Point_number* exists, then the string containing that point number has the shape applied to the entire length of the string.

See 36.5 Shape field coding

92 Remove all z-values for a string (i.e. make all z-values null)

92 [Point_description]

If no *point_description* is given, all z-values for the current string are removed.

If a *point_description* exists, then either the *feature code* and *string number* or the *point number* section of the *point_description* can be used.

If the *feature code* and *string number* exist, the last string with the same *feature code* and *string number* has all its z-values removed.

A *point-line type* can be embedding as a 0 or 1 in the *point name* part of the *point description* field. A zero value specifies a point string, and a non-zero value specifies a line string. If the field was omitted, a line string is assumed.

The point-line type may be overridden by the Map File.

93 Set the Point-line type

93 [Point_description]

A *point-line type* can be embedding as a 0 or 1 in the *point name* part of the *point description* field. A zero value specifies a point string, and a non-zero value specifies a line string. If the field was omitted, a line string is assumed.

If no *point_description* is given, the point-line type for the current string is set to *line*.

If a *point_description* exists, then either the *feature code* and *string number* or the *point number* section of the *point_description* can be used.

If the *feature code* and *string number* exist*,* the point-line type is set for the last previous string with the same *feature code* and *string number.*

If the *point number* exists*,* then the point-line type is set for the string containing that point number.

The point-line type may be overridden by the Map File.

94 Use name library file/ Map File for vertex text on the string - name mapping

94 [Point_description]

If this op code exists then during reduction, vertex text is creating using either the name library, or if the name library doesn't exist, the map file. If neither exist then the op code is ignored.

> if a name library is used and the feature code of the string is found in the first column of the name library, then the entry from the second column of that row will be used as text for *all* vertices of the string that don't already have vertex text. As a default, the string is set as a point string.
>
> if the map file is used and the feature code of the string is found in the first column of the map file, then the *string name* field of the map file is used as vertex text for all vertices that don't already have text. As a default, the string is set as a point string.

If no *point_description* is given, then name mapping is applied to the current string.

If a *point_description* exists, then either the *feature code* and *string number* or the *point number* section of the *point_description* can be used.

If the *feature code* and *string number* exist*,* then name mapping is applied to the last previous string with the same *feature code* and *string number.*

If the *point number* from the *point_description* exists*,* then name mapping is applied to the string containing that point number.

A *point-line type* can be embedding as a 0 or 1 in the *point name* part of the *point description* field. A zero value specifies a point string, and a non-zero value specifies a line string. If the field was omitted, a line string is assumed.

The point-line type may be overridden by the mapping file.

95 Pipe string

95 [Point_description] diameter

Pipe strings are always line strings and are stored with the justification of the majority of the string points. Individual pipe points are picked up either top (obvert), centre (axial) or bottom (invert) of the pipe using op codes 80, 81 and 82.

If no *point_description* is given, the current string is created as a pipe string with the given diameter.

If a *point_description* exists, then either the *feature code* and *string number* or the *point number* section of the *point_description* can be used.

If the *feature code* and *string number* exist, the last string with the same *feature code* and *string number* is created as a pipe with the given diameter.

If the *point number* exists*,* then the string containing that point number is created as a pipe string with the given diameter.

96 Culvert string

96 [Point_description] width height

Culvert strings are always line strings and are stored with the justification of the majority of the string points. Individual culvert points are picked up either top (obvert), centre (axial) or bottom (invert) of the culvert using op codes 80, 81 and 82.

If no *point_description* is given, the current string is created as a culvert string with the given width and height.

If a *point_description* exists, then either the *feature code* and *string number* or the *point number* section of the *point_description* can be used.

If the *feature code* and *string number* exist, the last string with the same *feature code* and *string number* is created as a culvert with the given width and height.

If the *point number* exists, then the string containing that point number is created as a culvert string with the given width and height.

99 Terminate processing

99

Stop processing the 12d field file at this line. Useful for debugging errors.

107 Make the previous segment invisible - after the measurement

107 [Point_description]

If no *Point_description* exists, the previous segment containing the current measurement point is set to invisible.

If the *Feature_code* and *String_number* exist, then the last segment of the previous string with that *feature code* and *string number* is set to invisible.

If *Point_number* exists, then the segment containing the point with that point number as an end point, is set to invisible.

108 Make the next segment invisible - after the measurement for the first point of the segment

108 [Point_description]

If no *Point_description* exists, the next segment containing the current measurement point as a starting point is set to invisible.

If the *Feature_code* and *String_number* exist, then the segment that is created in the future from the last point of the previous string with that *feature code* and *string number* is set to invisible.

If *Point_number* exists, then the segment containing the point with that point number as a start point, is set to invisible.

109 Make a point invisible - after the measurement

109 [Point_description]

If no *Point_description* exists, the current measurement point is set to invisible.

If *Feature_code* and *String_number* exist, then the last point of the previous string with that *feature code* and *string number* is set to invisible.

If *Point_number* exists, then the point with that point number is set to invisible.

110 Start buildings face observations - before the measurements

110 [Building_name]

Start recording a field template with the name *Building_name*. If *Building_name* is non-blank, then the default building face is defined. The *feature_code* and *string_number* of the following measurements until a *Finish* code (111) are stored as the building face. There is no limit to the number of points in a building face.

111 End building face observations

111 [Building_name]

If no *Building_name exist*s, the current building face observation set is finished (including the current measurement point).

Notes

1. Arc fitting is applied *after* the Joins are processed. Hence the new joined strings are created and then curve fitting is applied according to the arc codes (start arc, end arc, fit arcs, stop fitting arcs etc.) on any vertex of the string.
2. The *point description* has several pieces of information embedded in it and has been described in the previous section. For some op codes, the *point name* section of the *point description* is used to hold other information.

For a summary of the **12d Field File Op Codes**, go to the section 36.10.5 Summary of 12d Field File Op Codes

36.10.5 Summary of 12d Field File Op Codes

Op Code	Description of Record
-2	Comment
-1	Error
01	Header Information
02	Directly entered coordinate measurement
03	New instrument setup point
04	Measurement to backsight
05	New target height
06	Check measurement
07	Measurement - HA, VA, SD
09	Scale factor for subsequent distances
10	Three hair stadia measurement
11	Measurement - HA, HD, Ht
12	Measurement - HA, HD, Ht diff
15	Vertical circle correction
16	Multiply coded point
17	Arc through previous three points
18	Circle Feature
20	Close string
21	Join last points of strings
22	Join first to last point of strings
23	Join first points of strings
28	Add text to the string name
29	Note or memo
30	Remove height from a point - that is make it a null height
37	Rectangle by two points
38	Make the previous segment non-tinable
39	Make the next segment non-tinable
40	Make a point non-tinable
41	Add additional text for the current measurement point
42	Add a radial offset
43	Add a tangential offset
44	Add a height offset
45	Make a parallelogram from the last three measurement points
46	Make the string a breakline or not
47	Start a new string using the same feature code and string number
48	String end
50	Specify the bearing to correct for true north - used as the bearing datum difference
51	Start using an existing field template
52	Finish a field template or finish recording a field template
53	Pause the current field template until a continue op code (54) or a finish template (52)
54	Continue the current field template
55	Start recording a field template
56	Skip picking up one or more points from a field template
57	Delete one or more points from a field template
58	Insert a point into a field template
60	Arc through next three points
61	Start of arc through sets of three points until end of string, or a 62 occurs
62	End the arcs begun by a 61 command
68	Add an integer user defined attribute to the current string
69	Add a real user defined attribute to the current string
70	Add text user defined attribute to the current string
71	Add integer user defined attribute to the current point
72	Add real user defined attribute to the current point
73	Add text user defined attribute to the current point
74	Add integer user defined attribute to the next segment

75 Add real user defined attribute to the next segment
76 Add text user defined attribute to the next segment
77 Add integer user defined attribute to the previous segment
78 Add real user defined attribute for the previous segment
79 Add text user defined attribute to the previous segment
80 Pipe or culvert invert point (bottom of the pipe or culvert)
81 Pipe or culvert axial point (centre of the pipe or culvert)
82 Pipe or culvert obvert point (top of the pipe or culvert)
83 Start recording/measuring a shape
84 End measuring a shape
85 Parallel an existing shape
86 Extrude an existing shape
92 Remove all z-values for a string (i.e. make all z-values null)
93 Set the Point-line type
94 Use name library file/ mapping file for vertex text on the string - name mapping
95 Pipe string
96 Culvert string
99 Terminate processing
107 Last segment of point invisible
108 Next segment of point invisible
109 Point invisible
110 Building face start recording
111 Building face end recording

36.11 Batch Typed Entry

Typed entry can be used to enter survey data into a 12d Field File and reduced using the Survey data Editor.

12d Model uses the following 12d field file ops code:

Op Code	Description of Record
-2	Comment
-1	Error
01	Header Information
02	Directly entered coordinate measurement
03	New instrument setup point
04	Measurement to backsight
05	New target height
06	Check measurement
07	Measurement - HA, VA, SD
09	Scale factor for subsequent distances
10	Three hair stadia measurement
11	Measurement - HA, HD, Ht
12	Measurement - HA, HD, Ht diff
15	Vertical circle correction
16	Multiply coded point
17	Arc through previous three points
18	Circle Feature
20	Close string
21	Join last points of strings
22	Join first to last point of strings
23	Join first points of strings
28	Add text to the string name
29	Note or memo
30	Remove height from a point - that is make it a null height
37	Rectangle by two points
38	Make the previous segment non-tinable
39	Make the next segment non-tinable
40	Make a point non-tinable
41	Add additional text for the current measurement point
42	Add a radial offset
43	Add a tangential offset
44	Add a height offset
45	Make a parallelogram from the last three measurement points
46	Make the string a breakline or not
47	Start a new string using the same feature code and string number
48	String end
50	Specify the bearing to correct for true north - used as the bearing datum difference
51	Start using an existing field template
52	Finish a field template or finish recording a field template
53	Pause the current field template until a continue op code (54) or a finish template (52)
54	Continue the current field template
55	Start recording a field template
56	Skip picking up one or more points from a field template
57	Delete points from a field template
58	Insert points in a field template
60	Arc through next three points
61	Start of arc through sets of three points until end of string, or a 62 occurs
62	End the arcs begun by a 61 command
68	Add an integer user defined attribute to the current string
69	Add a real user defined attribute to the current string

70 Add text user defined attribute to the current string
71 Add integer user defined attribute to the current point
72 Add real user defined attribute to the current point
73 Add text user defined attribute to the current point
74 Add integer user defined attribute to the next segment
75 Add real user defined attribute to the next segment
76 Add text user defined attribute to the next segment
77 Add integer user defined attribute to the previous segment
78 Add real user defined attribute for the previous segment
79 Add text user defined attribute to the previous segment
80 Pipe or culvert invert point (bottom of the pipe or culvert)
81 Pipe or culvert axial point (centre of the pipe or culvert)
82 Pipe or culvert obvert point (top of the pipe or culvert)
83 Start recording/measuring a shape
84 End measuring a shape
85 Parallel an existing shape
86 Extrude an existing shape
92 Remove all z-values for a string (i.e. make all z-values null)
93 Set the Point-line type
94 Use name library file/ mapping file for vertex text on the string - name mapping
95 Pipe string
96 Culvert string
99 Terminate processing
107 Last segment of point invisible
108 Next segment of point invisible
109 Point invisible
110 Building face start recording
111 Building face end recording

Users can enter text for each measurement (observation 09 record or position 08 record) which is appended to the end of the record and this is used as the text of blocks that are interpreted according to the descriptions given in the earlier section 36.8 Field Coding for Non Leica Instruments.

The ’13’ record can also be used after a measurement record to add additional information to the preceding blocks using the *extra coding* control code at the end of the previous line (see Extra Coding in the section 36.8.4 Control Code Blocks

Strictly speaking the Sokkia SDR20/33 formats use fixed length lines and if the lengths are incorrect, an error message will be written to the Output Window. For example,

’Line 248 line incorrect length. required length is 58. received length is 50.’

These messages often appear after a raw file has been manually edited because most editors remove space padding at the end of a line.

For the full description of the **12d Field File Op Codes**, go to the section 36.10.4 Full Description of 12d Field File Op Codes

37 12d and Survey Instruments

See

37.1 12d and Leica TPS Instruments

The Leica GSI data format produced by the Leica TPS is used as a raw data file by 12d Model and is converted into a 12d Field File. The contents of the recorded data can be manipulated by use of the Leica's recording masks. The setup of which is explained in the section 37.1.6 Setup of Leica 1100 instrument for Detail Pickup and use with 12d.

Each line (data block) of the Leica GSI file consists of between 1 and 12 words, with the words containing either 16 (8 characters for data) or 24 (16 characters for data) characters. The two formats will be referred to as the 8 format and 16 format respectively.

Measurement lines in the 8 format start with 11 and code lines start with 41.

For the 16 format, measurement lines start with *11 and code lines start with *41.

For an example of a GSI file in the 8 format, go to the section 37.1.6.7.2 Example of Leica GSI File.

When using a Leica, the Leica screen can be standard or controlled by using a code.hex file on the 1000 series or a Geobasic program or Leica Codelist (.crf file) on the 1100 series.

12d Solutions provide a basic code.hex, Geobasic program and various Codelists(.crf files) on the ***12d Model*** Installation CD in the folder 'Other_Software\Leica'. These can be customised to your requirements - please contact 12d Solutions for more information.

For an example of a Leica screen with standard set-up, Codelist and Geobasic program, please go to the section 37.1.5 Examples of Leica Screens.

See the sections:

37.1.1 Feature Codes and String Numbers

37.1.2 Leica Field Codes

37.1.3 Full Description of Leica Field Codes

37.1.4 Summary of Leica Field Codes

37.1.5 Examples of Leica Screens

37.1.6 Setup of Leica 1100 instrument for Detail Pickup and use with 12d

37.1.6.7.2 Example of Leica GSI File

Please continue to the next section 37.1.1 Feature Codes and String Numbers.

37.1.1 Feature Codes and String Numbers

The *feature code* and *string number* for a measurement are entered in the Leica by giving the *feature code* in the first word and the *string number* as the second word of a code line. The feature codes and string numbers can be alpha or numeric and up to 8 (16) characters long.

For example, the feature code ABC with string number 1 is recorded as:

410003+00000ABC 42....+00000001 for the 8 format

or

*410003+0000000000000ABC 42....+0000000000000001 for the 16 format.

Some surveyors like to define the *feature code* and *string number* ***before*** they make a measurement. Other surveyors prefer to define the *feature code* and *string number* ***after*** they make a measurement.

When defining a *Data Collector setup*, ***12d Model*** allows the user to specify whether the feature code comes before or after a measurement:

code before measurement

The given *feature code* and *string number* applies to the **next** measurement and all subsequent measurements until another *feature code* and *string number* is entered.

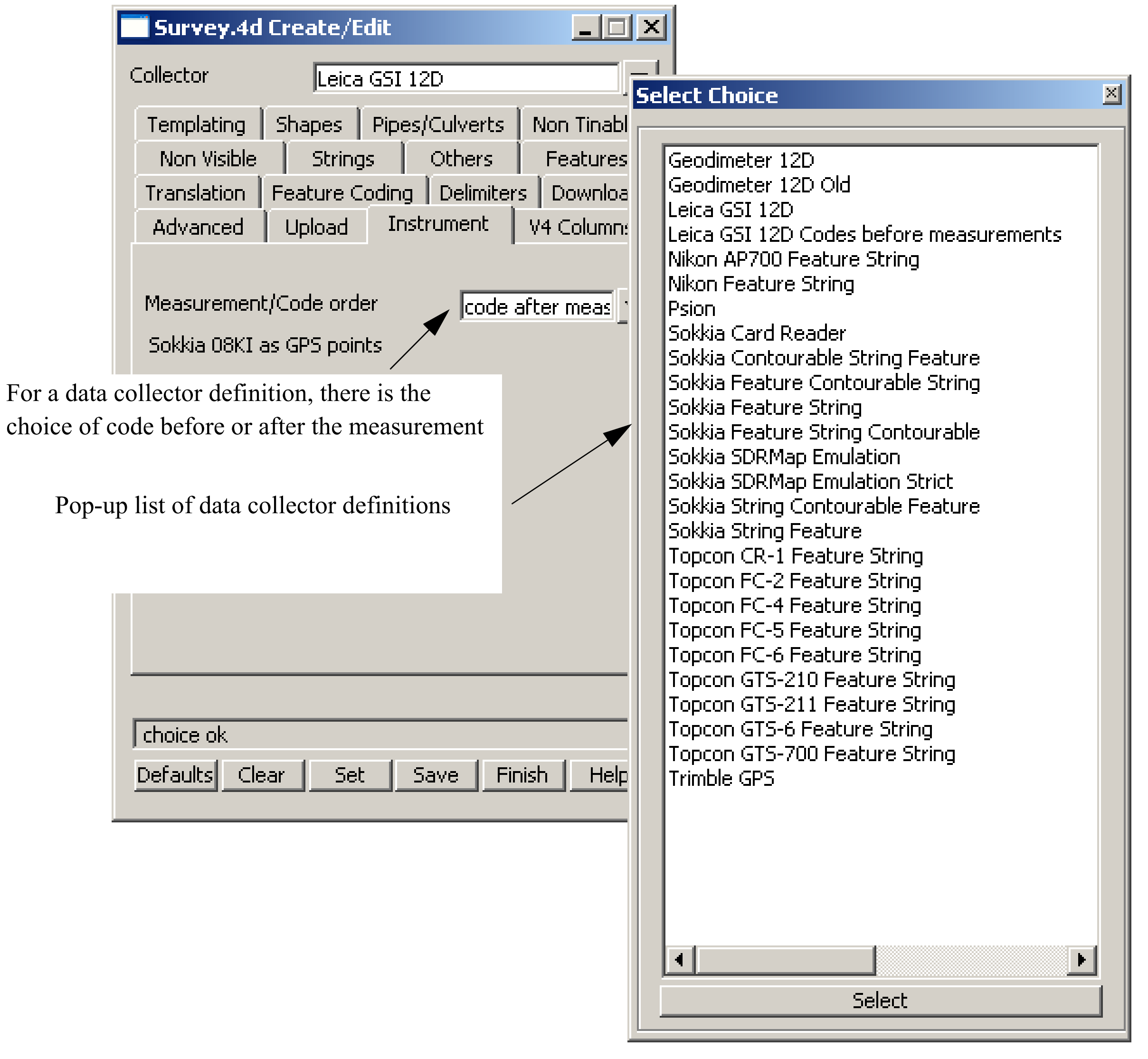
code after measurement

The given *feature code* and *string number* applies to the **last** measurement and all subsequent measurements until another *feature code* and *string number* is entered.

However, for a particular data collector setup, it can be only be defined as one of the two choices. That is you can't change between having feature codes before or after measurements in the one raw file.

Summarising:

Whether feature code definitions come before or after the measurements is defined in the *Survey Data Collector* definition and is set by the *Measurement/Code order* field on the *Instrument* tab of the *Surevy.4d Create/Edit* panel.



Please continue to the next section 37.1.2 Leica Field Codes.

37.1.2 Leica Field Codes

12d Model also uses code lines to supply extra information using field codes.

All *field codes* are entered as the first word of the code line and to differentiated it from a *feature code*, the *field code* is proceeded by a "**.**". Depending on the field code, more information may be required and it is entered in words 2 and above in the code line.

Each *field code* will now be described in detail.

For each *field code*, two lines and a paragraph of description are given:

- The first line consists of the *field code* and a short description of the purpose of the code.
- The second line gives the full syntax of the record for that *field code*.
- The paragraph gives a detailed description of the *field code*.

A 12d Model code list containing 12d Model field codes is available for the TPS to help coding in the field.

Field Code Conventions

In the syntax for the *field codes*, optional Words or groups of Words are enclosed in the square brackets **[ ]**.

However, when the Leica TPS creates the Leica GSI format, the data in all Words is right justified and left filled with zeros. Consequently, 12d Model strips leading zeros from all Words before processing the data.

This means that

(a) an **optional Word** still exists in the file but its data field is simply **filled with zeros**.

(b) any **text** in a Word **cannot start with a zero**. You would need to start with a space and then a zero.

In the *field code* descriptions, the point just measured is referred to as the ***current measurement point***, the ***current point*** or the ***last measurement point***.

The *string* that the current measurement point belongs to is called the ***current string***.

For a summary of the Leica Field Codes, go to the section 37.1.4 Summary of Leica Field Codes

For the full description of the Leica Field Codes, go to the section 37.1.3 Full Description of Leica Field Codes.

For an example of a raw Leica file, please go to the section 37.1.6.7.2 Example of Leica GSI File.

37.1.3 Full Description of Leica Field Codes

For a summary of the Leica Field Codes, go to the section 37.1.4 Summary of Leica Field Codes

Field Code **Description of Record**

.2 Directly entered coordinates

Word 1 .2 Word 2 X Word 3 Y Word 4 Z Word 5 Feature_code

Word 6 String_number [Word 7 Point_number Word 8 Point_name]

A *measurement* point is created with the given f*eature code* and *string number* and given (x, y, z) co-ordinates. No reduction is needed.

If a *Point_name* exists, then it is a ***named measurement*** and in the reduction, a 4d point string of name *Point_name* is created and mapped using the Map File. The 4d text is the station prefix followed by *Point_name*. The *Point_name* is added to the internal list of named points for searching for co-ordinates.

Note that the Leica GSI format includes directly entered coordinates in a code line by data blocks with word indices 81, 82, 83 (Easting, Northing and Elevation).

.3 New instrument point - before the measurement

Word 1 .3 Word 2 Point_name Word 3 Instrument_height

Set up an instrument at the point with name *Point_name*. In the reduction, the (x, y, z) co-ordinates for *Point_name* are found by first searching the control model, then the list of previously named points in the reduction, point numbers of previous measurements and finally if *Point_name* is still not found, the user is asked to type in the (x, y, z) co-ordinates.

Note that the Leica GSI format includes setting up on a station in a measurement line by data blocks with word indices 84, 85, 86 (Station easting, Station northing and Station elevation) and 88 (Instrument height).

.4 Next measurement is a backsight - before the measurement

Word 1 .4 Word 2 Station_name [Word 3 Feature_code Word 4 String_number Word 5 Target_height]

The ***next*** measurement is to a backsight with name *Station_name.*

For traverse reduction, if the traverse code specified in the reduction panel matches the Feature code given in Word 3, the data will be included in a traverse reduction, string number can also be specified. Also the target height to the backsight can be given in the backsight command.

.5 New target height for subsequent measurements - before the measurement

Word 1 .5 Word 2 Target_height

Set a new target height to be used for all subsequent measurement points.

Note that the Leica GSI format can include the target height on each measurement line (word index 97).

.6 Next measurement is a Check measurement - before the measurement

Word 1 .6 Word 2 Station_name

The ***next*** measurement is a check measurement made to the station *Station_name*.

During reduction a two point super string (with name *Station_name*) from the instrument point to the measured point is created in the default model for the check measurement. The instrument point name, the station name and the differences between the measurement point co-ordinates and station co-ordinates are written as text along the super string. The differences between the measurement and the known point is also written to the report file.

.9 Scale factor for subsequent distances

Word 1 .9 Word 2 Scale_factor

Scale factor to apply to subsequent slope distances.

.11 Next measurement is to a named station - before the measurement

Word 1 .11 Word 2 Station_name [Word 3 Feature_code Word 4 String_number Word 5 Target_height]

This field code gives a name to a measurement so that it can be used in other field codes for setting up an instrument on, backsighting to or doing a check measurement to.

The ***next*** measurement locates a new instrument station with the name *Station_name*. A point with vertex text of *Station_name* is created. The *Station_name* is added to the internal list of named points for searching for co-ordinates.

If *Feature_code* exists, the point is given that *Feature_code*, otherwise the current feature code is used.

For traverse reduction, if the traverse code specified in the reduction panel matches the Feature code given in Word 3, the data will be included in a traverse reduction, string number can also be specified. Also the target height to the foresight can be given in this command.

.14 Feature code and string number - before or after measurement depends on the data collector definition

Word 1 .14 Word 2 Feature_code Word 3 String_number [Word 5 text Word 6 text] ...

A field code for entering the feature code and string number rather than using the standard method of using word 1 and word 2 This is for use with Leica quick codes which must put out a fixed first word. Any additional *text* is added to the end of any existing vertex text for the current measurement point. Any leading zeros in the text words will be ignored.

This field code is used *always before* or *always after* measurements depending on the *Measurement/code* field on the *Instrument* tab of the **Survey.4d Create/Edit** panel for the selected data collector definition.

.15 Vertical circle correction for subsequent measurements - before the measurement

Word 1 .15 Word 2 Vertical_circle_in_decimal_degrees

The *vertical_circle_in_decimal_degrees* is **subtracted** from the vertical circle value in any subsequent measurements.

.16 Additional code for point (multiply coded point) - after the measurement

Word 1 .16 Word 2 Feature_code Word 3 String_number [Word 4 Point_name] [Word 5 Point_text]

Additional coding for the current measurement point. A new measurement point is created at the same position as the current measurement point but with the *Feature_code* and *String_number* from this field code. The same point number is used as for the current measurement point.

If *Point_text* exists, it is used as the vertex text for that vertex of the super string.

If *Point_name* exists, then it is a ***named measurement*** and during reduction, a 4d point string of name *Point_name* is created and mapped using the Map File. The 4d text is the station prefix followed by *Point_name*. The *Point_name* is added to the internal list of named points for searching for co-ordinates.

.17 Arc through previous three points - after the measurement of last point of arc

Word 1 .17 [Word 2 Feature_code Word 3 String_number] [Word 4 Point_number]

If only Word 1 exists, then the current measurement point and the two previous points with the same feature code and string number as the current measurement point, are joined by an arc. If there is less than three such points, no arc is fitted.

If the *Feature_code* and *String_number* exist*,* the last three previous three measurement points of the same *feature code* and *string number* are joined by an arc. If the current measurement point has that feature code and string number, then it is the third of the three points used. If there is less than three points, no arc is fitted.

If *Point_number* exists*,* then the *feature code* and *string number* are taken from the previous measurement point **with** that point number, and that point and the two measurement points previous to the pre-defined point of the same *feature code* and *string number*, are joined by an arc. If there is less than three points, no arc is fitted

See the Section 36.3.9 Arcs Through Points for more information on arc fitting.

.18 Circle Feature - after the measurement

Word 1 .18 Word 2 Radius [Word 2 Radius **...***]*

Creates a feature string with the given radius and centred on the current measurement point. If there is more than one radius, then separate feature strings are created each with one of the radii.

.20 Close string - after the measurement

Word 1 .20 [Word 2 Feature_code Word 3 String_number] [Word 4 Point_number]

If only Word 1 exits, the current string is closed.

If the *Feature_code* and *String_number* exist*,* the last previous string with that *feature code* and *string number* is closed.

If *Point_number* exists*,* then the string containing that point number is closed.

See the Section 36.3.4 Close String for more information on closing a string.

.21 Join last points of strings

Word 1 .21 Word 2 Feature_code Word 3 String_number_1 Word 4 String_number_2

In the final reduction, the last point of the string with the given *Feature_code* and *String_number_1* is joined to the last point of the string with given *Feature_code* and *String_number_2*. The created string has the given *Feature_code* (no string number is needed since it is the final phase of reduction when the string numbers are dropped).

See the Section 36.3.8 Joining Strings for more information on joining strings.

.22 Join first to last point of strings

Word 1 .22 Word 2 Feature_code Word 3 String_number_1 Word 4 String_number_2

In the final reduction, the first point of the string with the given *Feature_code* and *String_number_1* is joined to the last point of the string with given *Feature_code* and *String_number_2*. The created string has the given *Feature_code* (no string number is needed since it is the final phase of reduction when the string numbers are dropped).

See the Section 36.3.8 Joining Strings for more information on joining strings.

.23 Join first points of strings

Word 1 .23 Word 2 Feature_code Word 3 String_number_1 Word 4 String_number_2

In the final reduction, the first point of the string with the given *Feature_code* and *String_number 1* is joined to the fist point of the string with given *Feature_code* and *String_number_2*. The created string has the given *Feature_code* (no string number is needed since it is the final reduction when the string numbers are then dropped).

See the Section 36.3.8 Joining Strings for more information on joining strings.

.28 Text appended to a string name

Word 1 .28 Word 2 Text [Word 3 Feature_code Word 4 String_number] [Word 5 Point_number]

Text is appended to the string name. For example, if 1.200 is entered, “ 1.200” is appended to the string name.

If only *Text* is given, *Text* is appended to the string name of the current string.

If the *Feature_code* and *String_number* exist, then the last previous string with that *feature code* and *string number* has *Text* appended to the string name.

If *Point_number* exists, then the string containing that point number has *Text* appended to the string name.

.29 Note or memo - after the measurement

Word 1 .29 [Word 2 text] [Word 3 text] **...**

Any text may be entered and will be added to the check measurements model at the position of the current measurement point.

.30 Remove height from a point - that is make it a null height - after the measurement

Word 1 .30 [Word 2 Feature_code Word 3 String_number] [Word 4 Point_number]

If only Word 1 exists, the height of the current measurement point is set to null.

If *Feature_code* and *String_number* exist, then the height of the last point of the previous string with that *feature code* and *string number* is set to null.

If *Point_number* exists, then the height of the point with that point number is set to null.

.37 Rectangle by two points

Word 1 .37 Word 2 offset_in_metres [Word 3 Feature_code Word 4 String_number] [Word 5 Point_number]

The rectangle is defined by two points (reference side) and a offset.

If a positive offset value is given, two points will be created to the right of the reference side.

If a negative offset value is given, two points will be created to the left of the reference side.

If *Feature_code* and *String_number* exist, the last two points with that *feature code* and *string number* are used and a fourth points are created to form a rectangle. The height of the two new points are set to null. The string is then closed.

If *Point_number* exists, then the *feature code* and *string number* of the point with that point number are used and processed as above. Note that the point with the point number is not necessarily used.

Two consecutive rectangles are unable to be defined side by side. In other words if the two points given are part of string of greater than two vertices, the command will only work for sets of two points that are exclusively defined. i.e. For a 5 point string, a rectangle can be defined by points 1 and 2, and 4 and 5.

See the Section 36.3.5 Rectangle for more information.

.38 Make the previous segment non-tinable (not a breakline) - after the measurement

Word 1 .38 [Word 2 Feature_code Word 3 String_number] [Word 4 Point_number]

If only Word 1 exists, the previous segment containing the current measurement point is set to non-tinable. That is, it will not be treated as a breakline in triangulations.

If the *Feature_code* and *String_number* exist, then the last segment of the previous string with that *feature code* and *string number* is set to non-tinable.

If *Point_number* exists, then the segment containing the point with that point number as an end point, is set to non-tinable.

.39 Make the next segment non-tinable (i.e. not a breakline) - after the measurement for the first point of the segment

Word 1 .39 [Word 2 Feature_code Word 3 String_number] [Word 4 Point_number]

If only Word 1 exists, the next segment containing the current measurement point as a starting point is set to non-tinable. That is, it will not be treated as a breakline in triangulations.

If the *Feature_code* and *String_number* exist*,* then the segment that is created in the future from the last point of the previous string with that *feature code* and *string number* is set to non-tinable.

If *Point_number* exists*,* then the segment containing the point with that point number as a start point, is set to non-tinable.

.40 Make a point non-tinable - after the measurement

Word 1 .40 [Word 2 Feature_code Word 3 String_number] [Word 4 Point_number]

If only Word 1 exists, the current measurement point is set to non-tinable. That is, it will not be included in triangulations.

If *Feature_code* and *String_number* exist*,* then the last point of the previous string with that *feature code* and *string number* is set to non-tinable.

If *Point_number* exists*,* then the point with that point number is set to non-tinable.

.41 Add additional text to the current measurement point - after the measurement

Word 1 .41 Word 2 text [Word 3 text] ...

The given *text* is added to the end of any existing vertex text for the current measurement point. Any leading zeros in the text words will be ignored.

.42 Add a radial offset - after the measurement

Word 1 .42 Word 2 Radial_offset_in_metres [Word 3 Feature_code Word 4 String_number] [Word 5 Point_number]

The *Radial_offset_in_metres* is used to adjust the position of the specified point by a plan distance from the specified points original position, along the plan line joining the current station to the specified point. A positive offset is away from the station, negative is toward the station.

If only Word 1 and 2 *exist*, the offset is used to adjust the position of the current measured point.

If *Feature_code* and *String_number* exist*,* then the last point of the previous string with that *feature code* and *string number* is adjusted.

If *Point_number* exists*,* then the point with that point number is adjusted.

See the Section 36.3.2 Offsets for more information on offsets.

.43 Add a tangential offset - after the measurement

Word 1 .43 Word 2 Tangential_offset_in_metres [Word 3 Feature_code Word 4 String_number] [Word 5 Point_number]

The *Tangential_offset_in_metres* is used to adjust the position of the specified point by a plan distance from the specified points original position, at rights angles to the plan line joining the current station to the specified point. A negative offset is to the left (looking from the station), and positive is to the right (looking from the station).

If only Word 1 and 2 *exist*, the offset is used to adjust the position of the current measured point.

If *Feature_code* and *String_number* exist*,* then the last point of the previous string with that *feature code* and *string number* is adjusted.

If *Point_number* exists*,* then the point with that point number is adjusted.

See the Section 36.3.2 Offsets for more information on offsets.

.44 Add a height offset - after the measurement

Word 1 .44 Word 2 Height_offset_in_metres [Word 3 Feature_code Word 4 String_number] [Word 5 Point_number]

If the height of the specified point is not null, then the *Height_offset_in_metres* adjusts the height of the point. A positive offset adds to the height, a negative offset reduces the height.

If only Word 1 and 2 *exist*, the offset is used to adjust the position of the current measured point.

If *Feature_code* and *String_number* exist*,* then the last point of the previous string with that *feature code* and *string number* is adjusted.

If *Point_number* exists*,* then the point with that point number is adjusted.

See the Section 36.3.2 Offsets for more information on offsets.

.45 Make a parallelogram from the last three measurement points - after the measurement

Word 1 .45 [Word 2 Feature_code Word 3 String_number] [Word 4 Point_number]

If only Word 1 *exist*s, the current measurement point and the two previous points from the current string are used and a fourth point is created to form a parallelogram (squashed rectangle) and the height of the fourth point is set to null. The string is then closed.

If *Feature_code* and *String_number* exist*,* the last three points with that *feature code* and *string number* are used and a fourth point is created to form a parallelogram (squashed rectangle) and the height of the fourth point is set to null. The string is then closed.

If *Point_number* exists*,* then the *feature code* and *string number* of the point with that point number are used and processed as above. Note that the point with the point number is not necessarily used.

See the Section 36.3.5 Rectangle for more information on forming a parallelogram.

.46 Make the entire string a breakline or not (tinable or non-tinable)

Word 1 .46 [Word 2 Breakline_type] [Word 3 Feature_code Word 4 String_number] [Word 5 Point_number]

***Word 2: Breakline_type*:**

If *Breakline_type* is 0, the selected string is set to a point string and hence is not a breakline (non-tinable).

If *Breakline_type* is 1, the selected string is set to a *line* string and is therefore a breakline (tinable).

Words 3, 4, and 5:

If none of Words 3, 4 and 5 exist, the *Breakline_type* is applied to the current string.

If *Feature_code* and *String_number* exist*,* the *Breakline_type* is applied to the last string with that *feature code* and *string number.*

If *Point_number* exists*,* then the *Breakline_type* is applied to the string containing the point with that point number.

.47 Start a new string using the same feature code and string number - after the measurement of the first point of the new string

Word 1 .47 [Word 2 Feature_code Word 3 String_number] [Word 4 Point_number]

If only Word 1 *exist*s, the current string is terminated (without including the current measurement point) and the current measurement point becomes the first point of a new string with the same feature code and string number.

If *Feature_code* and *String_number* exist*,* then the last point of the previous string with that feature code and string number becomes the first point of a new string with the same *feature code* and *string number.*

If *Point_number* exists*,* then the previous string containing the point with that point number is terminated *before* the point number point, and the point becomes the first point of a new string with the *same feature code* and *string number.*

.48 End a string

Word 1 .48 [Word 2 Feature_code Word 3 String_number] [Word 4 Point_number]

If only Word 1 *exist*s, the current string is terminated (including the current measurement point).

If *Feature_code* and *String_number* exist*,* then the last point of the previous string with that feature code and string number becomes the last point of that string.

If *Point_number* exists, then the previous string containing the point with that point number is terminated *after* the point number point

.50 Specify the bearing to correct for true north - used as the bearing datum difference

Word 1 .50 Word 2 Bearing_in_decimal_degrees [Word 3 Text]

The *Bearing_in_decimal_degrees* is used as the bearing datum difference for the current instrument set up. The *Text* and the *Bearing_in_decimal_degrees* are written to the reduction report file.

.51 Start using an existing field template - before the measurement

Word 1 .51 Word 2 [Template_name] [mode]

Start using the template *Template_name.* If *Template_name* is blank, the default template is used
If *mode* is "for", then the field template is used as a *forward* template.
"rev", then the field template is used as a *reverse* template.
"zig", then the field template is used as a zig_zag template and is used in the *forward* definition direction first.
"zag", then the field template is used as a zig_zag template and is used in the *reverse* direction first
If *mode* is blank, or 0, or anything other than "for", "rev", or "zag", then the field template is used as a *zig-zag* field template starting on a *zig*.

See the Section 36.4 Field Templates for more information on field templates.

.52 Finish using a field template or finish recording a field template - after the measurement

Word 1 .52

Stops using the current field template or stops recording a field template.

.53 Pause using the current field template - after the measurement

Word 1 .53

Pause using the current field template or defining a field template, until a continue template (54) or a finish field template (52) code is given.

.54 Continue the current field template - before the measurement

Word 1 .54

Continue using the current field template or defining the current field template, which has been stopped by a *Pause* command (53). The *Continue* command only needs to be given once and applies to all *following* measurements until another *Pause* or *Finish* command is given.

.55 Start recording a field template - before the measurement

Word 1 .55 Word 2 [Template_name]

Start recording a field template with the name *Template_name*. If *Template_name* is non-blank, then the default field template is defined. The *feature_code* and *string_number* of the following measurements until a *Finish* code (52) are stored as the field template. There is no limit to the number of points in a field template.

See the Section 36.4 Field Templates for more information on field templates.

.56 Skipping picking up points when using a field template - after the measurement of last point before skipping points

Word 1 .56 Word 2 [num_skipped_points]

Allows the user to skip picking up one or more points from the field template currently being used. The *next measurement* takes the *feature_code* and *string_number* from the next point of the field template definition. If *num_skipped_points* is missing, then only one point is skipped otherwise *num_skipped_points are skipped.*

.57 Delete points on a field template - after the measurement of last point

Word 1 .57 Word 2 [num_points_to_delete]

Allows the user to delete one or more points from the field template currently being used. The *next measurement* takes the *feature_code* and *string_number* from the next point of the field template definition.

.58 Insert points when using a field template - after the measurement of last point before inserting points

Word 1 .58 Word 2 Feature_code Word 3 String_number Word 4 Multiple_code_flag Word 5 Insert_special_flag

Allows the user to insert points into the field template currently being used. The *next measurement* takes the *feature_code* and *string_number* from the next point of the field template definition.

If the *Multiple_code_flag = 1, then the feature code will be added to the previous defined template point else if Multiple_code_flag = 0 (default), it will be added to the template as a separate point.*

If the insert is done at the end of a section and the *Insert_special_flag = 1 the point will be added to the end of the current template section else it will be at the start of the next section.*

.60 Start of arc through next three points - after the measurement of the first point of the arc

Word 1 .60 [Word 2 Feature_code Word 3 String_number] [Word 4 Point_number]

If only Word 1 *exist*s, an arc is inserted through the current measurement point and the next two measured points with the same feature code and string number as the current measurement point. If there is less than three points, no arc is fitted.

If *Feature_code* and *String_number* exist*,* a search is made for a previously defined measurement with the same feature code or string number. An arc is inserted through the previous measurement and the next two measured points following this previous measurement with the same feature code and string number. If the current point has that feature code and string number, then it is the first of the three point. If there is less than three points, no arc is fitted.

If *Point_number* exists*,* then the *feature code* and *string number* are taken from the previous measurement point **with** that point number, and an arc is inserted through that point and the next two measurement points with the same *feature code* and *string number*. If there is less than three points, no arc is fitted

See the Section 36.3.9 Arcs Through Points for more information on arc fitting.

.61 Start of arc through sets of three points until end of string, or a 62 occurs - after the measurement of the first point of the arc

Word 1 .61 [Word 2 Feature_code Word 3 String_number] [Word 4 Point_number]

The arcs are fitted as follows - the first arc is fitted through points one, two and three, the next arc through points three, four and five etc. If there is less than three points remaining at the end, then no arc is fitted to the end points.

If only Word 1 *exist*s, arcs are inserted through the following sets of measurement points with the same feature code and string number as the current measurement point. The current measurement point is the *first* of the points.

If the current point has that feature code and string number, then it is the first of the points. If there is less than three points, then no arc is fitted.

If the *feature code* and *string number exist*, a search is made for a previously defined measurement with the same feature code or string number. An arc is inserted through the following measured points with the same feature code and string number as given in *point_description*. If the current point has that feature code and string number, then it is the first of the points.

If *Point_number* exists*,* then the *feature code* and *string number* are taken from the previous measurement point **with** that point number, and arcs are inserted through that point and the following measured points with the same *feature code* and *string number*.

If ***12d Model*** encounters a *Start Arcs through sets of three points* but no *End Arcs* command for the string, then an *End Arcs* is assumed at the end of the string.

See the Section 36.3.9 Arcs Through Points for more information on arc fitting.

.62 End the arcs begun by a 61 command - after the measurement of the last point of the arcs

Word 1 .62 [Word 2 Feature_code Word 3 String_number] [Word 4 Point_number]

If only Word 1 *exists*, then the fitting of arcs through the points of the current string is stopped. The current measurement point is the *last* of the points used in the arc fitting.

If *Feature_code* and *String_number* exist, then the fitting of arcs through the points of the previous string with the same *feature code* and *string number* is stopped. If the current measurement point has that feature code and string number, then it is the last point used in the arc fitting.

If *Point_number* exists, then the point with that point number is the last point used in the arc fitting.

If ***12d Model*** encounters an *End Arcs* (62) but no *Start Arcs through sets of three points* (61) command for the string, then an *Start Arcs through sets of three points* (61) is assumed to apply at the beginning of the string and hence arc fitting will be applied to the entire string.

See the section 36.3.9 Arcs Through Points for more information on arc fitting.

There are field codes for adding user defined attributes to:

(a) the current string being measured (i.e. the string containing the current measurement point)

(b) the current measurement point

(c) the next segment from the current measurement point (i.e. the segment joining the current measurement point and the *next* measured point *of the same feature code and string number*)

or

(d) the previous segment to the current measurement point (i.e. the segment joining the current measurement point to the previous measured point *of the same feature code and string number*).

The attributes can be named or unnamed and are coded in the following way:

If Word 2 is missing (i.e. all zeros), then the attribute is an un-named attribute and the rest of the Words on the line is the attribute value. The attribute is given the special name "unnamed attribute n" for n=1, 2, ...

.68 Add an integer user defined attribute to the current string

Word 1 .68 [Word 2 Name] Word 3 Integer

Add an user defined integer attribute to the current string.

.69 Add a real user defined attribute to the current string

Word 1 .69]Word 2 Name] Word 3 Real

Add a real (floating point) user defined attribute to the current string.

.70 Add text user defined attribute to the current string

Word 1 .70 [Word 2 Name] Word 3 Text [Word 4 Text] **...**

Add a text user defined attribute to the current string.

.71 Add integer user defined attribute to the current point

Word 1 .71 [Word 2 Name] Word 3 Integer

Add an integer user defined attribute to the current measurement point.

.72 Add real user defined attribute to the current point

Word 1 .72 [Word 2 Name] Word 3 Real

Add a real (floating point) user defined attribute to the current measurement point.

.73 Add text user defined attribute to the current point

Word 1 .73 [Word 2 Name] Word 3 Text [Word 4 Text] **...**

Add a text user defined attribute to the current measurement point.

.74 Add integer user defined attribute to the next segment

Word 1 .74 [Word 2 Name] Word 3 Integer

Add an integer user defined attribute to the next segment from the current measurement point.

.75 Add real user defined attribute to the next segment

Word 1 .75 [Word 2 Name] Word 3 Real

Add a real (floating point) user defined attribute to the next segment from the current measurement point.

.76 Add text user defined attribute to the next segment

Word 1 .76 [Word 2 Name] Word 3 Text [Word 4 Text] **...**

Add a text user defined attribute to the next segment from the current measurement point.

.77 Add integer user defined attribute to the previous segment

Word 1 .77 [Word 2 Name] Word 3 Integer

Add an integer user defined attribute to the previous segment for the current measurement point.

.78 Add real user defined attribute for the previous segment

Word 1 .78 [Word 2 Name] Word 3 Real

Add a real (floating point) user defined attribute to the previous segment for the current measurement point.

.79 Add text user defined attribute to the previous segment

Word 1 .79 [Word 2 Name] Word 3 Text [Word 4 text] **...**

Add a text user defined attribute to the previous segment for the current measurement point.

In addition, extra codes allow ***12d Model*** pipe strings to be coded in the field.

.80 Pipe invert point (bottom of the pipe) - after the measurement

Word 1 .80 [Word 2 Feature_code Word 3 String_number] [Word 4 Point_number]

If only Word 1 *exist*s, the current measurement point is on the invert (bottom) of a pipe. This is the default for measurements to points on pipe strings. If the point is not part of a pipe string, it is ignored.

If *Feature_code* and *String_number* exist, the last point of the previous string with the same *feature code* and *string number* is on the invert (bottom) of a pipe. If the point is not part of a pipe string, it is ignored.

If *Point_number* exists, then the point with that point number is on the invert (bottom) of a pipe. If the point is not part of a pipe string, it is ignored.

.81 Pipe axial point (centre of the pipe) - after the measurement

Word 1 .81 [Word 2 Feature_code Word 3 String_number] [Word 4 Point_number]

If only Word 1 *exist*s, the current measurement point is on the axis (centre) of a pipe. If the point is not part of a pipe string, it is ignored.

If *Feature_code* and *String_number* exist*,* the last point of the previous string with the same *feature code* and *string number* is on the axis (centre) of a pipe. If the point is not part of a pipe string, it is ignored.

If *Point_number* exists*,* then the point with that point number is on the axis (centre) of a pipe. If the point is not part of a pipe string, it is ignored.

.82 Pipe obvert point (top of the pipe) - after the measurement

Word 1 .82 [Word 2 Feature_code Word 3 String_number] [Word 4 Point_number]

If only Word 1 *exist*s, the current measurement point is on the obvert (top) of a pipe. If the point is not part of a pipe string, it is ignored.

If *Feature_code* and *String_number* exist*,* the last point of the previous string with the same *feature code* and *string number* is on the obvert (top) of a pipe. If the point is not part of a pipe string, it is ignored.

If *Point_number* exists*,* then the point with that point number is on the obvert (top) of a pipe. If the point is not part of a pipe string, it is ignored.

.83 Start recording a shape - before the measurement

Word 1 .83 [Word 2 Shape_name]

Start recording a shape with the name *Shape_name*. If *Shape_name* is non-blank, then the default field Shape is defined by the *feature_code* and *string_number* of the following measurements until a *Finish* code (84) are stored as the shape. There is no limit to the number of points in a shape.

See the Section 36.4 Field Templates for more information on field templates.

.84 Finish using a shape definition or finish recording a shape - after the measurement

Word 1 .84

Stops using the current shape or stops recording a shape.

.85 Shape parallel

Word 1 .85 Word 2 Shape_name [Word 3 Feature_code Word 4 String_number] [Word 5 Point_number]

If *Feature_code* and *String_number* exist*,* the last string with the same *feature code* and *string number* has the shape applied to the entire length of the string.

If *Point_number* exists*,* then the string containing that point number has the shape applied to the entire length of the string.

Parallelling the shape will produce separate strings for each point of the shape.

.86 Shape extrude

Word 1 .86

Word 1 .85 Word 2 Shape_name [Word 3 Feature_code Word 4 String_number] [Word 5 Point_number]

If *Feature_code* and *String_number* exist*,* the last string with the same *feature code* and *string number* has the shape applied to the entire length of the string.

If *Point_number* exists*,* then the string containing that point number has the shape applied to the entire length of the string.

Extruding the shape will produce a single string for the shape.

.92 Remove all z-values for a string (i.e. make all z-values null) - after the measurement

Word 1 .92 [Word 2 Point_line_type] [Word 3 Feature_code Word 4 String_number] [Word 5 Point_number]

If only Word 1 *exist*s, all z-values for the current string are removed.

If *Feature_code* and *String_number* exist*,* the last string with the same *feature code* and *string number* has all its z-values removed.

If *Point_number* exists*,* then the string containing that point number has all its z-values removed.

If *Point_line_type* is 0, the selected string is set to a point string.
If *Point_line_type* is 1, the selected string is set to a *line* string.

.93 Set the Point-line type - after the measurement

Word 1 .93 [Word 2 Point_line_type] [Word 3 Feature_code Word 4 String_number] [Word 5 Point_number]

If only Word 1 *exist*s, the current string is created as a point string.

If *Feature_code* and *String_number* exist*,* the last previous string with the same *feature code* and *string number* has its point-line type modified.

If the *point number* exists*,* then the string containing that point number has its point-line type modified.

If *Point_line_type* is 0, the selected string is set to a point string.
If *Point_line_type* is 1, the selected string is set to a *line* string.

.94 Use name library/mapping file for vertex text on the string - name mapping - after the measurement

Word 1 .94 [Word 2 Point_line_type] [Word 3 Feature_code Word 4 String_number] [Word 5 Point_number]

If this field code exists then during reduction, vertex text is creating using either the name library, or if the name library doesn't exist, the map file. If neither exist then the field code is ignored.

> if a name library is used and the feature code of the string is found in the first column of the name library, then the entry from the second column of that row will be used as text for *all* vertices of the string that don't already have vertex text. As a default, the string is set as a point string.
>
> if the map file is used and the feature code of the string is found in the first column of the map file, then the *string name* field of the map file is used as vertex text for all vertices that don't already have text. As a default, the string is set as a point string.

If only Word 1 *exist*s, then name mapping is applied to the current string.

If *Feature_code* and *String_number* exist*,* then name mapping is applied to the last previous string with the same *feature code* and *string number.*

If *Point_number* exists*,* then name mapping is applied to the string containing that point number.

If *Point_line_type* is 0, the selected string is set to a point string.
If *Point_line_type* is 1, the selected string is set to a *line* string.

.95 Circular Pipe string - after the measurement

Word 1 .95 Word 2 Diameter [Word 3 Feature_code Word 4 String_number] [Word 5 Point_number]

Pipe strings are always line strings and are stored with the justification of the majority of the string points. Individual pipe points are picked up either top (obvert), centre (axial) or bottom (invert) of the pipe using field codes 80, 81 and 82.

If only Word 1 and 2 *exist*s, the current string is created as a pipe string with the given diameter.

If *Feature_code* and *String_number* exist, the last string with the same *feature code* and *string number* is created as a pipe with the given diameter.

If *Point_number* exists, then the string containing that point number is created as a pipe string with the given diameter.

.96 Box Culvert string - after the measurement

Word 1 .95 Word 2 Width Word 3 Height [Word 4 Feature_code Word 5 String_number] [Word 6 Point_number]

Culvert strings are always line strings and are stored with the justification of the majority of the string points. Individual culvert points are picked up either top (obvert), centre (axial) or bottom (invert) of the culvert using field codes 80, 81 and 82.

If only Word 1, 2 and 3 *exist*s, the current string is created as a culvert string with the given width and height.

If *Feature_code* and *String_number* exist, the last string with the same *feature code* and *string number* is created as a culvert with the given width and height.

If *Point_number* exists, then the string containing that point number is created as a culvert string with the given width and height.

.99 Terminate processing

Word 1 .99

Stop processing the field file at this line. Useful for debugging errors.

.107 Make the previous segment invisible - after the measurement

Word 1 .107 [Word 2 Feature_code Word 3 String_number] [Word 4 Point_number]

If only Word 1 exists, the previous segment containing the current measurement point is set to invisible.

If the *Feature_code* and *String_number* exist, then the last segment of the previous string with that *feature code* and *string number* is set to invisible.

If *Point_number* exists, then the segment containing the point with that point number as an end point, is set to invisible.

.108 Make the next segment invisible - after the measurement for the first point of the segment

Word 1 .108 [Word 2 Feature_code Word 3 String_number] [Word 4 Point_number]

If only Word 1 exists, the next segment containing the current measurement point as a starting point is set to invisible.

If the *Feature_code* and *String_number* exist, then the segment that is created in the future from the last point of the previous string with that *feature code* and *string number* is set to invisible.

If *Point_number* exists, then the segment containing the point with that point number as a start point, is set to invisible.

.109 Make a point invisible - after the measurement

Word 1 .109 [Word 2 Feature_code Word 3 String_number] [Word 4 Point_number]

If only Word 1 exists, the current measurement point is set to invisible.

If *Feature_code* and *String_number* exist, then the last point of the previous string with that *feature code* and *string number* is set to invisible.

If *Point_number* exists, then the point with that point number is set to invisible.

.110 Start buildings face observations - before the measurements

Word 1 .110 Word 2 [Building_name]

Start recording a field template with the name *Building_name*. If *Building_name* is non-blank, then the default building face is defined. The *feature_code* and *string_number* of the following measurements until a *Finish* code (111) are stored as the building face. There is no limit to the number of points in a building face.

.111 End building face observations

Word 1 .111 Word 2 [Building_name]

If only Word 1 *exist*s, the current building face observation set is finished (including the current measurement point).

Notes

1. Arc fitting is applied *after* the Joins are processed. Hence the new joined strings are created and then curve fitting is applied according to the arc codes (start arc, end arc, fit arcs, stop fitting arcs etc.) on any vertex of the string.

For a summary of the Leica Field Codes, go to the section 37.1.4 Summary of Leica Field Codes

37.1.4 Summary of Leica Field Codes

Most of the field codes are entered after the measurement is taken and they will have a (AM) after the description. Some of the field codes are entered before the measurement is taken (mainly set up codes) and will have a (BM) after the description.

Field Code	Description of Record
.2	Directly entered coordinate measurement
.3	New instrument setup point (BM)
.4	Measurement to backsight (BM)
.5	New target height (BM)
.6	Check measurement (BM)
.9	Scale factor for subsequent distances (BM)
.10	Three hair stadia measurement
.11	Measurement to a named station (BM)
.15	Vertical circle correction (BM)
.14	A field code for entering Feature code and String number - BM or AM depends on data collector definition
.16	Multiply coded point (AM)
.17	Arc through previous three points (AM of last point of arc)
.18	Circle Feature (AM)
.20	Close string (AM)
.21	Join last points of strings
.22	Join first to last point of strings
.23	Join first points of strings
.28	Add text to the string name (AM)
.29	Note or memo (AM)
.30	Remove height from a point - that is make it a null height (AM)
.37	Rectangle by two points
.38	Make the previous segment non-tinable (AM of end point of segment)
.39	Make the next segment non-tinable (AM of first point of segment)
.40	Make a point non-tinable (AM)
.41	Add additional text (AM)
.42	Add a radial offset (AM)
.43	Add a tangential offset (AM)
.44	Add a height offset (AM)
.45	Make a parallelogram from the last three measurement points (AM)
.46	Make the string a breakline or not
.47	Start a new string using the same feature code and string number (AM of first point of new string).
.48	End a string -
.50	Specify the bearing to correct for true north - used as the bearing datum difference
.51	Start using an existing field template (BM)
.52	Finish using a field template or finish recording a field template (AM)
.53	Pause the current field template until a continue op code (54) or a finish template (52)
.54	Continue the current field template (BM)
.55	Start recording a field template (BM)
.56	Skip picking up one or more points from a field template (AM of last point before skipping)
.57	Delete one or more points from a field template (AM)
.58	Insert a point into a field template (BM)
.60	Arc through next three points (AM of first point of arc)
.61	Start of arc through sets of three points until end of string, or a 62 occurs (AM of first point of arc)
.62	End the arcs begun by a 61 command (AM of last point of arcs)
.68	Add an integer user defined attribute to the current string (AM of any point of the string)
.69	Add a real user defined attribute to the current string (AM of any point of the string)
.70	Add text user defined attribute to the current string (AM of any point of the string)
.71	Add integer user defined attribute to the current point (AM)
.72	Add real user defined attribute to the current point (AM)
.73	Add text user defined attribute to the current point (AM)

.74 Add integer user defined attribute to the next segment (AM of first point of the segment)
.75 Add real user defined attribute to the next segment (AM of first point of the segment)
.76 Add text user defined attribute to the next segment (AM of first point of the segment)
.77 Add integer user defined attribute to the previous segment (AM of last point of the segment)
.78 Add real user defined attribute for the previous segment (AM of last point of the segment)
.79 Add text user defined attribute to the previous segment (AM of last point of the segment)
.80 Pipe invert point (bottom of the pipe) (AM)
.81 Pipe axial point (centre of the pipe) (AM)
.82 Pipe obvert point (top of the pipe) (AM)
.83 Start recording a shape - before the measurement
.84 Finish using a shape definition or finish recording a shape - after the measurement
.85 Shape parallel
.86 Shape extrude
.92 Remove all z-values for a string (i.e. make all z-values null) (AM of point of the string)
.93 Set the Point-line type
.94 Use name library file/ mapping file for vertex text on the string - name mapping
.95 Circular pipe string with diameter (AM)
.96 Box culvert string with dimensions (AM)
.99 Terminate processing
.107 Make the previous segment invisible - after the measurement
.108 Make the next segment invisible - after the measurement for the first point of the segment
.109 Make a point invisible - after the measurement
.110 Start buildings face observations - before the measurements
.111 End building face observations

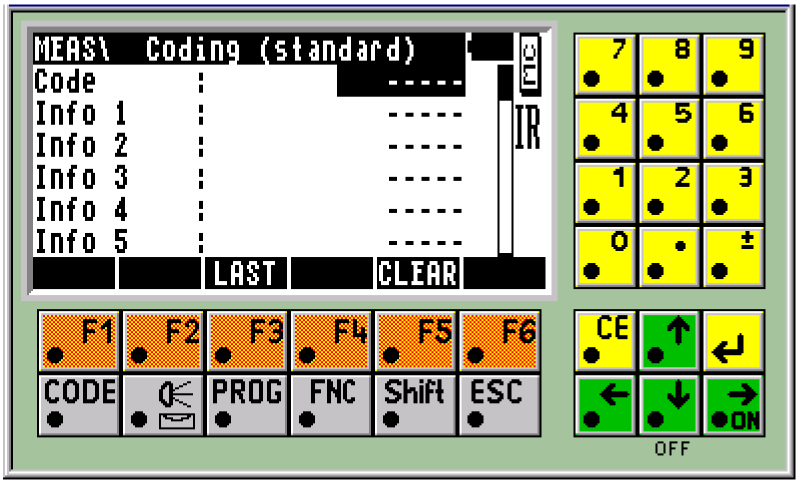
For a full description of the Leica codes, please go to the section 37.1.3 Full Description of Leica Field Codes.

For an example of a raw Leica file, please go to the section 37.1.6.7.2 Example of Leica GSI File.

37.1.5 Examples of Leica Screens

Standard Screen

When using a Leica instrument with no code.hex on the 1000 series, or Geobasic program or Codelist (.crf file) on the 1100 series, hitting the *Code* button brings up the standard Leica screen:

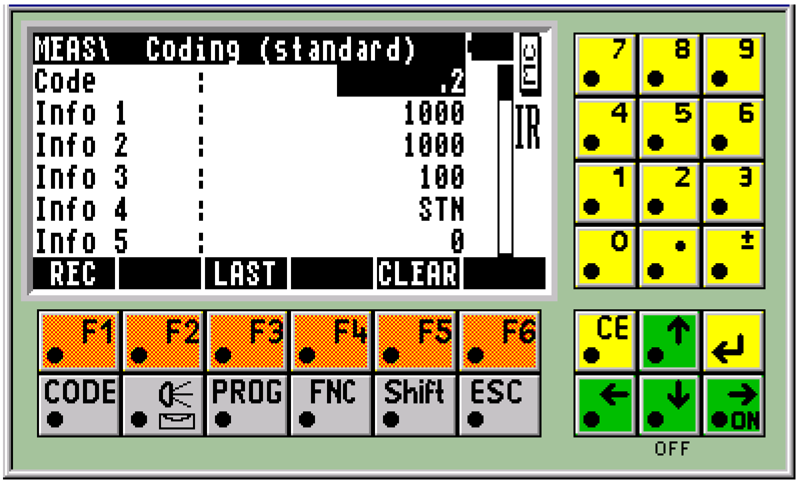


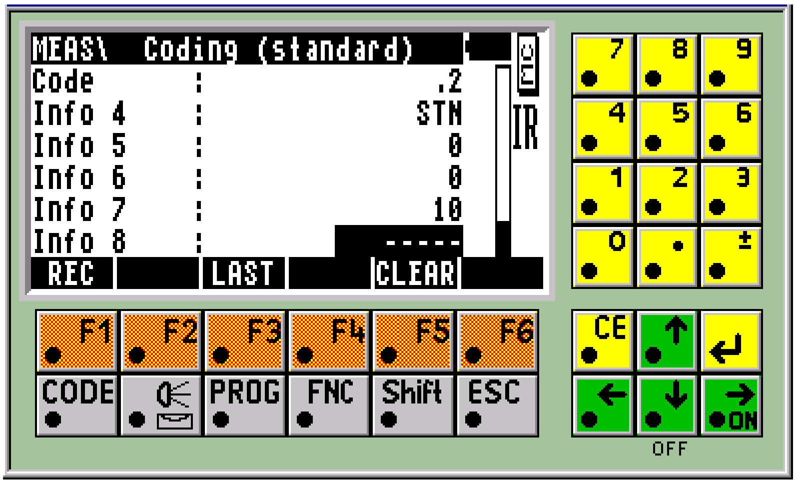
The information entered in *Code* goes to Word 1 of the line written to the Leica GSI file.

The information in *Info 1* to *Info 8* go to Word 2 to Word 9 respectively of the line written to the GSI file.

As an example, to enter a *directly entered co-ordinate* (the 2 field code),

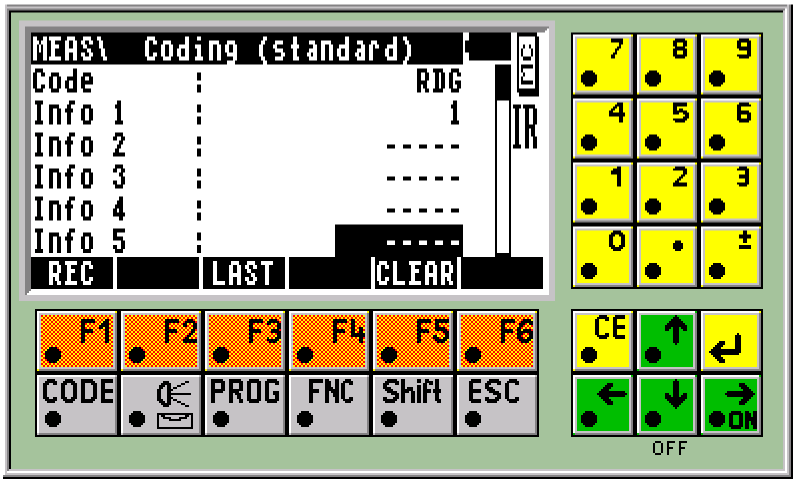
".2" is entered in the *Code* field
X, Y and Z in Info 1, Info 2 and Info 3 respectively.
the feature code (STN) is entered into Info 4
the station name "10" is entered into Info 7





To enter a *feature code and string number* to be used for measurements

the feature code is entered into the *Code* field
the string number is entered into *Info 1*.



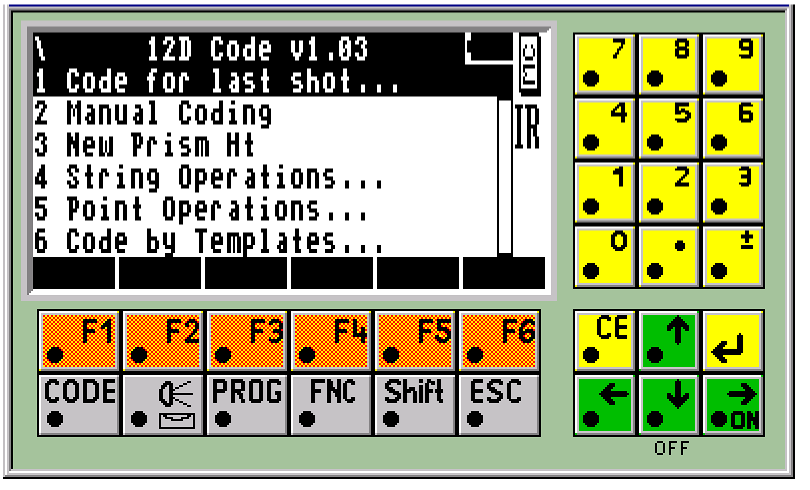
Leica Geobasic Program

A Leica Geobasic program gives the programmer full control over the Leica screen.

WARNING

Only one Geobasic program can exist on the Leica at any one time. Loading a Geobasic program will over write the existing Geobasic program on the instrument. Make sure you have a backup copy of the existing Geobasic program on your computer before loading a new one onto the instrument.

If you do not have a backup copy of the existing Geobasic program, do not load a new one.



The front screen of the 4D Solutions supplied Geobasic program

Leica Codelist

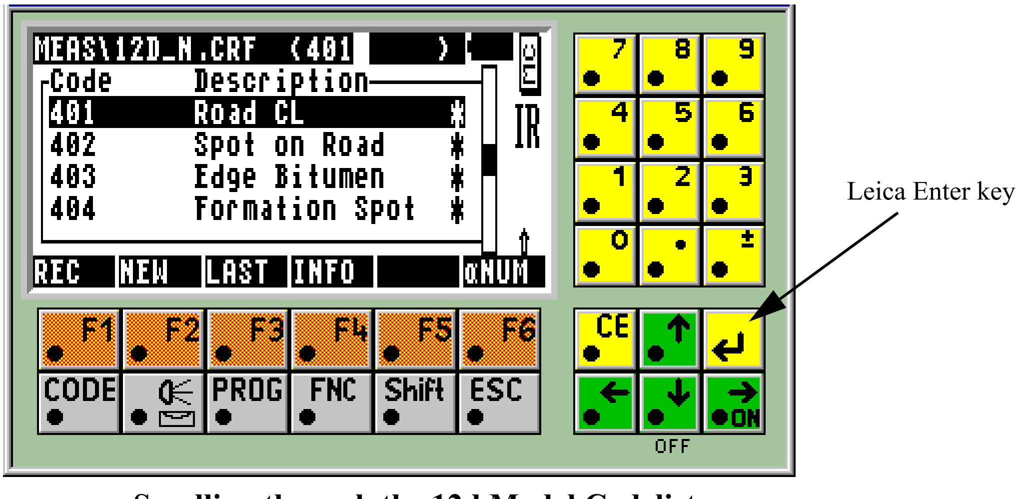
A Leica Codelist (.crf file) controls what is displayed and validated on the Leica screen and what values are recorded whenever the *Code* button is selected. For example, when using the 4D Solutions supplied Codelists , a list of defined feature codes and descriptions are displayed.

The Codelist controls what codes are displayed and whenever a code is selected, what information is required for that code. The Codelist also specifies what information is written out to the GSI file for that code.

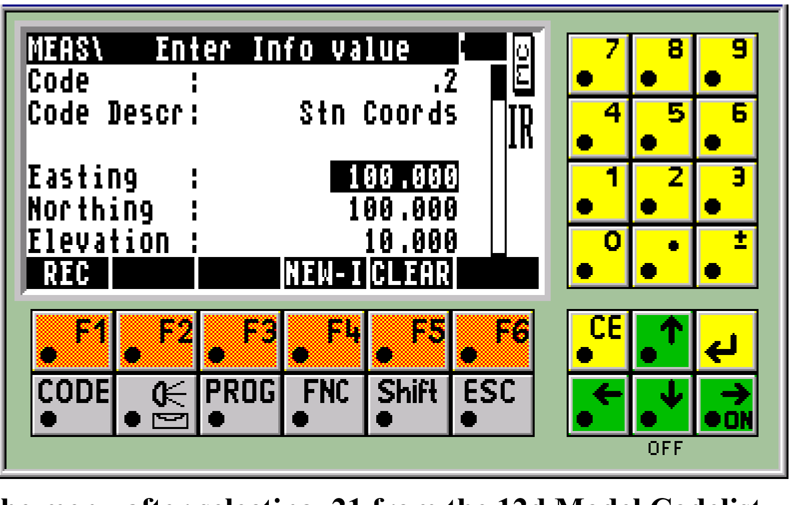
After hitting the *Code* button, the list of codes from the Codelist is displayed in alphanumeric order. Because the ***12d Model*** field codes start with a ".", they will appear at the top of the list.

A code can be typed in or the list can be scrolled through using the arrow keys.

The highlighted code is selected by pressing the Leica Enter Key. The menu for the selected code then appears. The line of data is written out to the GSI file when the F1 (REC) button is selected.



Scrolling through the 12d Model Codelist



The menu after selecting .21 from the 12d Model Codelist

37.1.6 Setup of Leica 1100 instrument for Detail Pickup and use with 12d

The following relates to the setup of a Leica 1100 series instrument for topographic/detail pickup and use with 12d.

37.1.6.1 Setting up the Instrument

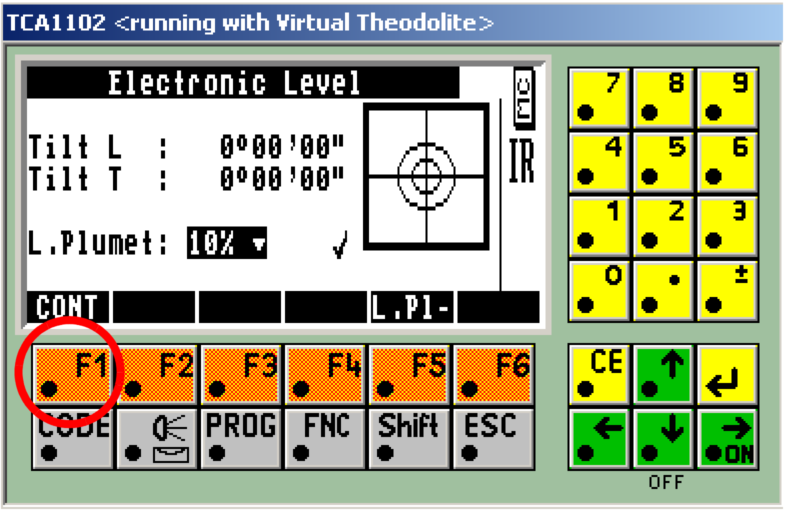
37.1.6.1.1 Copying the CRF file

A number of example .crf files are include on the installation CD. These can be found in the directory Other_Software\Leica\ . For the purpose of this documentation, we will use the file **12D_DS.CRF** which is designed for detail pickup. This file should be copied from the 12d CD to the folder “Code” on the Leica card.

This file contains all of the field codes used in ***12d Model*** along with a sample code listing

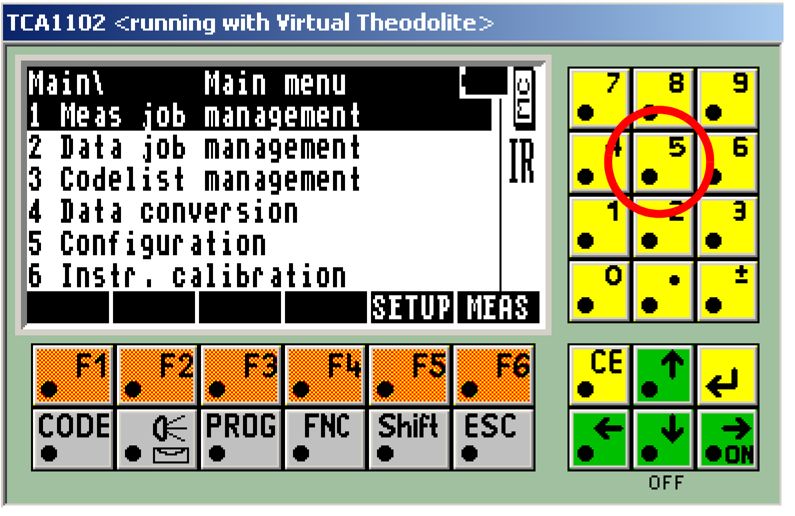
37.1.6.1.2 Set up the display and record masks

Turn on and level instrument

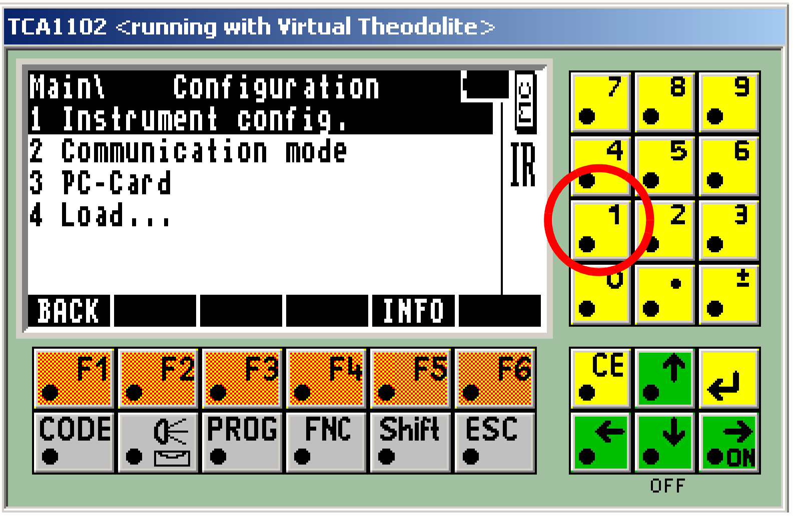


Press F1 to Continue

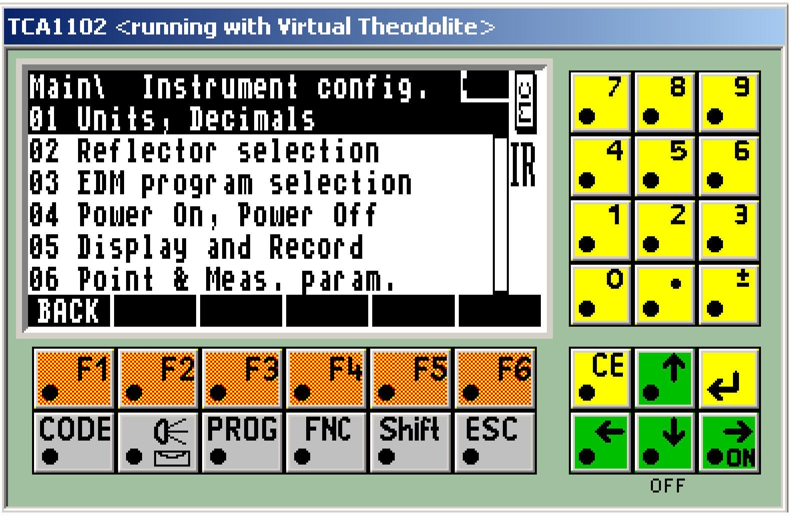
Press 5 to configure the instrument



Press 1 for instrument configuration

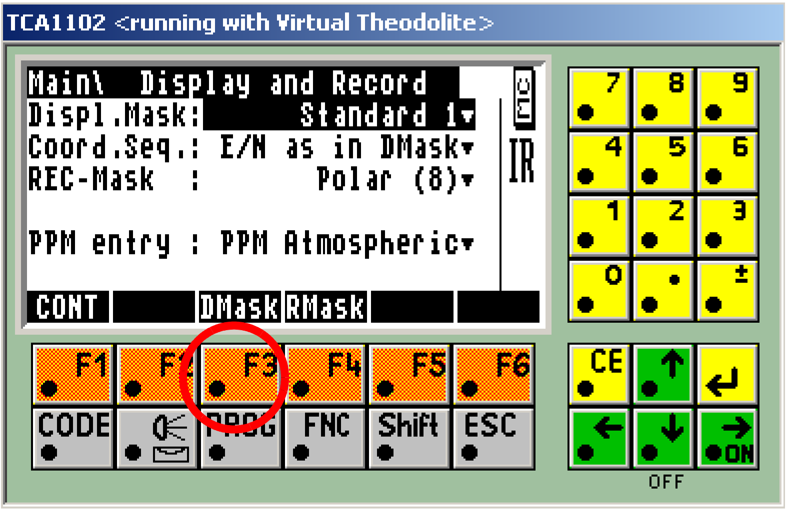


Press 05 for Display and Record mask setup

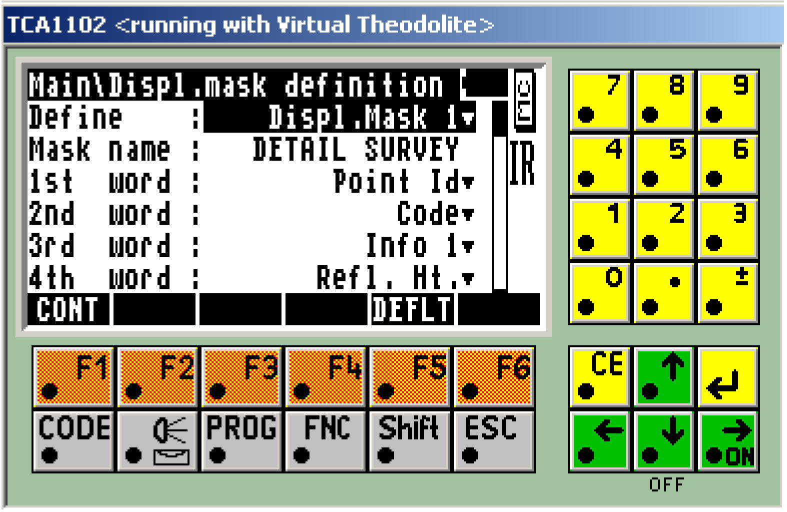


Display mask:

Press F3 to set up display mask



Type in the description DETAIL SURVEY for the Mask name



Select the following settings:

1st Word…….Point Id

2nd Word…….Code

3rd Word…….Info 1

4th Word…….Refl. Ht.

5th Word…….Hz

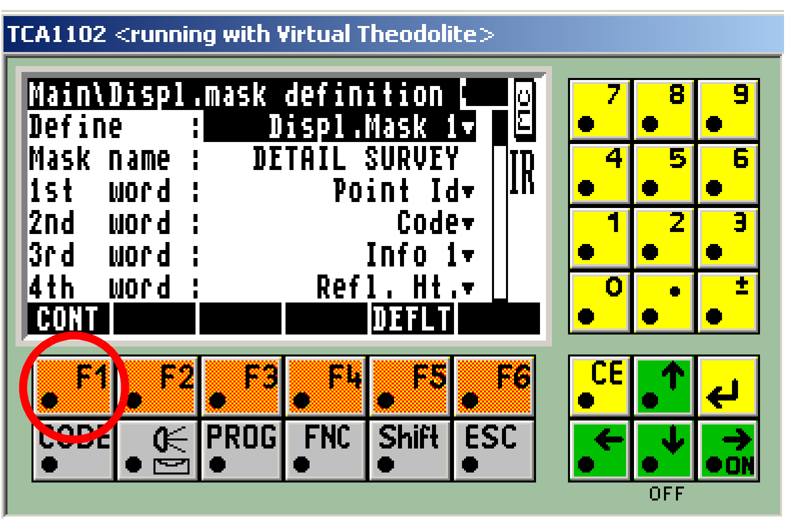
6th Word…….Horiz. Dist

7th Word…….East

8th Word…….North

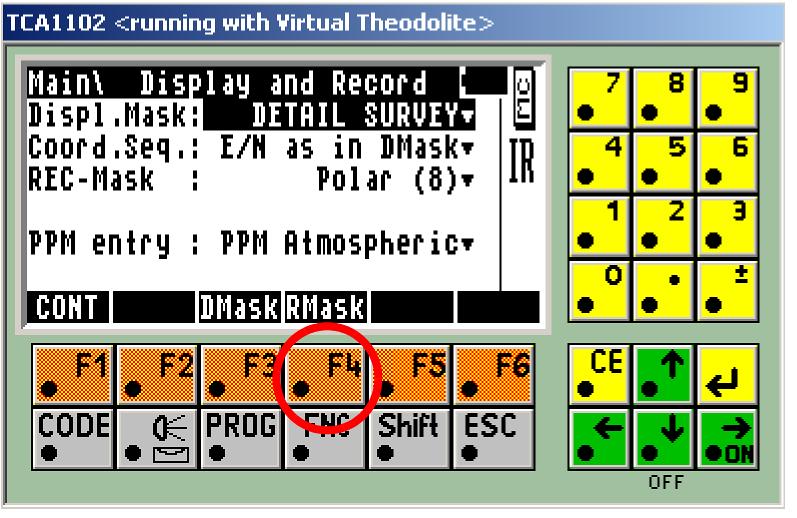
9th Word…….Elev

Press F1

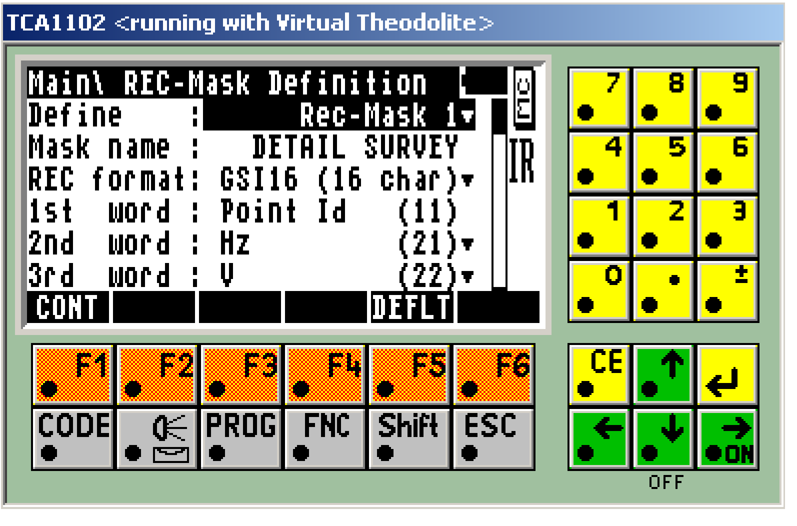


Record mask:

Press F4 to set up record mask



Type in the description DETAIL SURVEY for the Mask name



Select the following settings:

REC format….GSI16 (16 char)

1st Word…….Point Id (11)

2nd Word…….Hz (21)

3rd Word……..V (22)

4th Word…….Slope Dist (31)

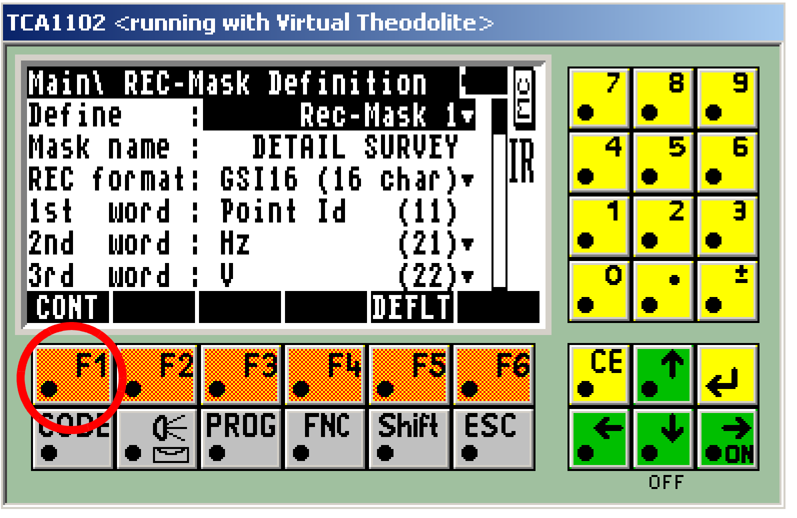
5th Word…….Ref. Ht. (87)

6th Word…….East (81)

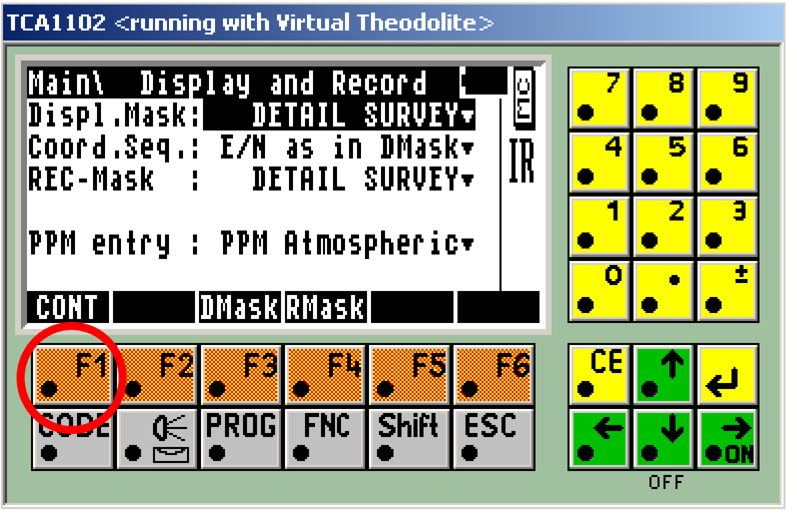
7th Word…….North (82)

8th Word…….Elev (83)

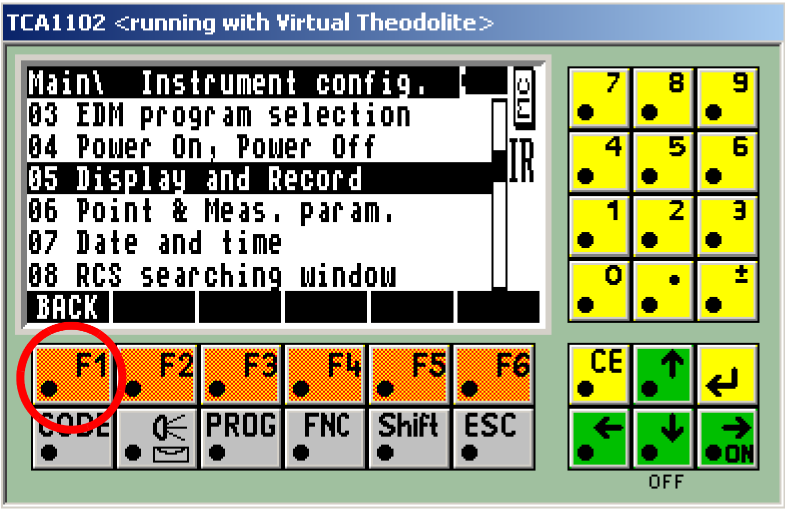
Press F1



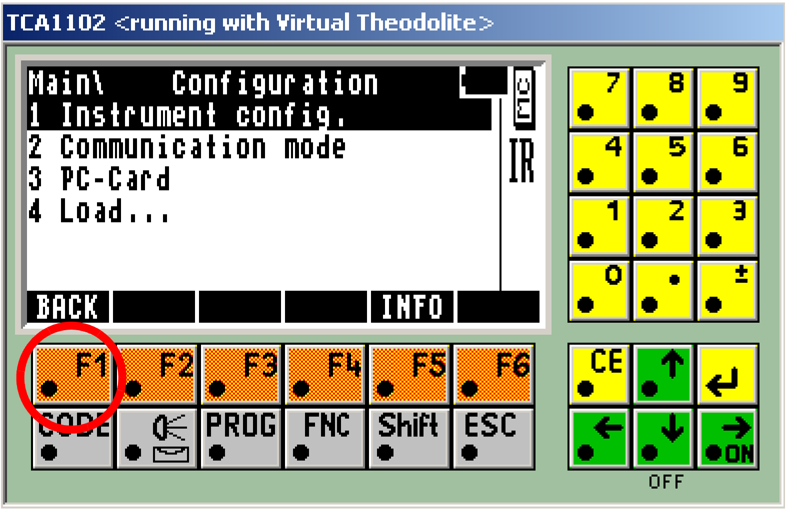
Press F1



Press F1 to exit menu



Press F1 to exit menu



37.1.6.1.3 Leica field codes

It is important to know when the code is entered into the instrument as some codes should precede a reading (BM) and some come after (AM). In this manual if a code is related to the current string then it is placed before the reading.

Below is a list of the commonly used codes to be used

.2	Directly entered coordinates
.3	New instrument station
.4	Backsight reading (BM)
.5	Height of target (BM)
.6	Check measurement (BM)
.9	Scale factor (BM)
.11	Foresight station (BM)
.14	Feature code and string number (Use for Leica quick codes only)
.15	Vertical circle correction (d.dddd) (BM)
.16	Additional code for point (AM)
.17	Arc thru last 3 points (AM)
.18	Circle (AM)
.20	Close string (AM)
.29	Note (AM)
.30	Remove height from point (AM)
.37	Rectangle by two points (AM)
.38	Make last segment non tinable (AM)
.39	Make next segment non tinable (AM)
.40	Make point non tinable (AM)
.42	Add radial offset (AM)
.43	Add tangential offset (AM)
.44	Add height offset (AM)
.45	Create parallelogram from last three points (AM)
.47	Start new string with same code and string no. as previous string (AM)
.48	End string (AM)….Not generally used

Templates

.51	Start template readings (BM)
.52	Finish template readings (AM)
.53	Pause template (AM)
.54	Continue a template (BM)
.55	Start recording template (BM)
.56	Skip points on template (BM)

Arcs

.60	Arc thru next 3 points (AM)…..See also .17
.61	Arc through sets of 3 points until .62 code entered (BM)
.62	End of arc from .61 (AM)

Pipes

.80	Reading taken to invert level of pipe (AM)
.81	Reading taken to centre of pipe (AM)
.82	Reading taken to obvert level of pipe (AM)
.95	Circular pipe diameter (AM)
.96	Box culvert dimensions (AM)

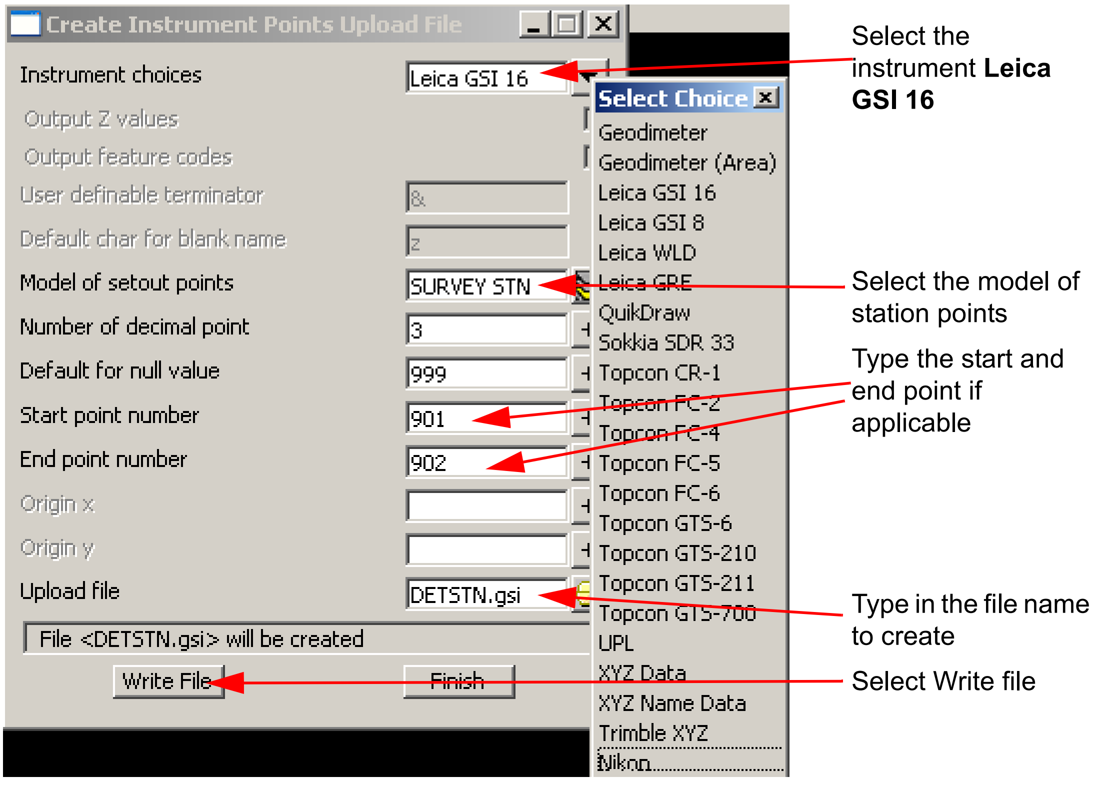
.92	Remove heights from string (AM)
.107	Make previous segment invisible (AM)
.108	Make next segment invisible (AM)

37.1.6.2 Setting up files

The existing station coordinates for a pickup can be entered into the Leica by hand or they can be uploaded using 12d. The 12d upload sequence will be described here.

37.1.6.2.1 Uploading coordinate file

Select option **Survey=>Upload=>Create points upload file**

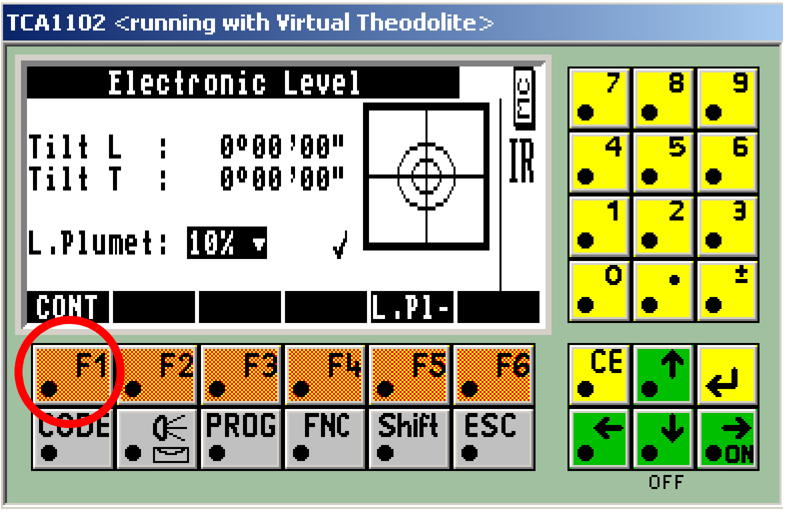


37.1.6.2.2 Copy the upload file

The file upload file should be copied to the folder “GSI” on the Leica card

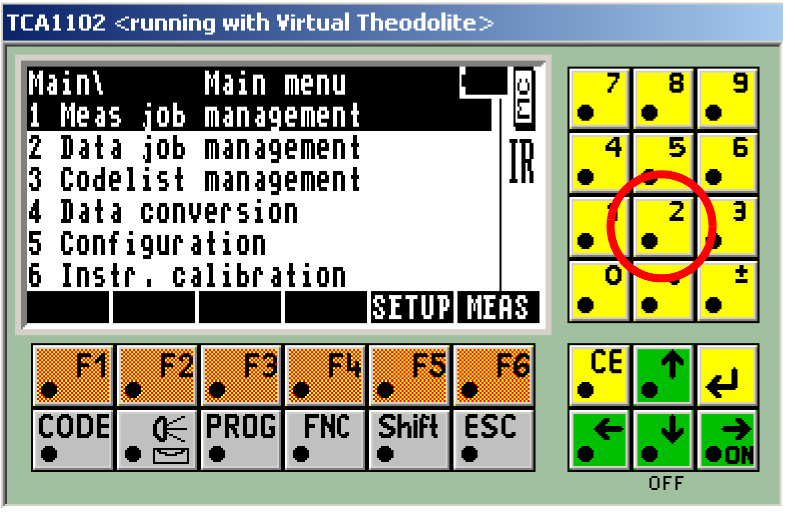
37.1.6.2.3 Data file setup

Turn on and level instrument



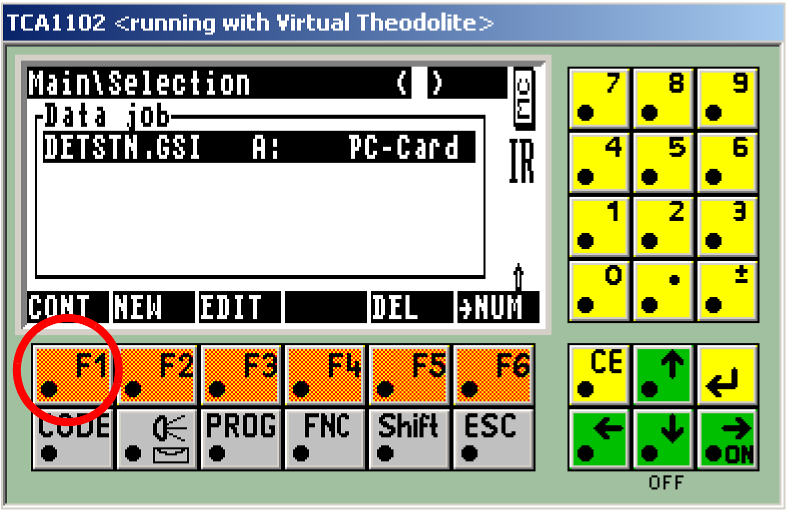
Press F1 to Continue

Press 2 to select Data job file.



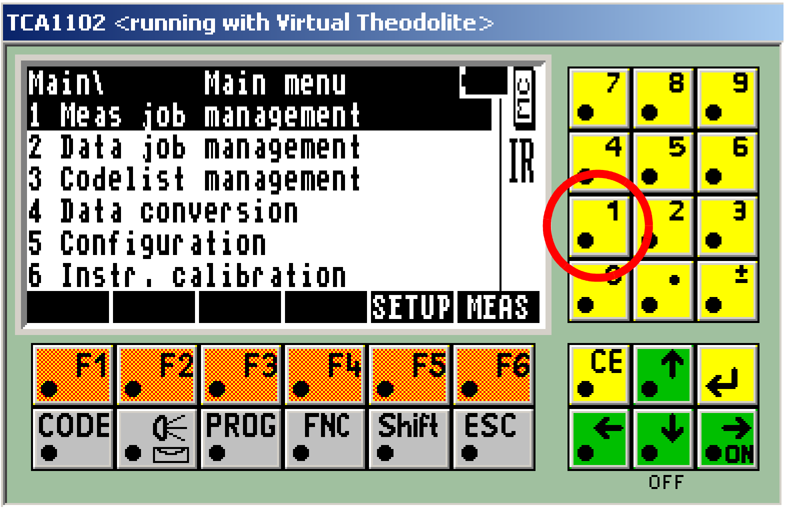
This is the file containing the control points if uploaded or if entered during the survey. Keep the file name separate from the measured data file name

Select the file to use followed by the F1 key or press F2 to create a new file



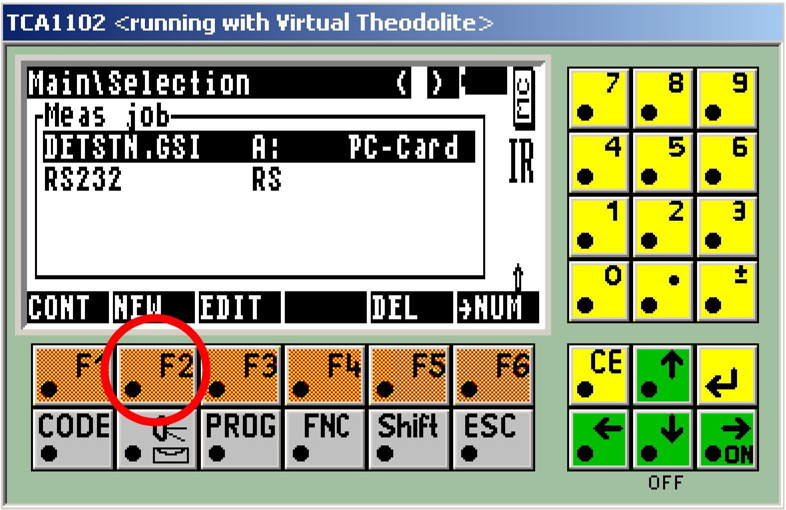
Press F1 to continue

37.1.6.3 Measured file setup

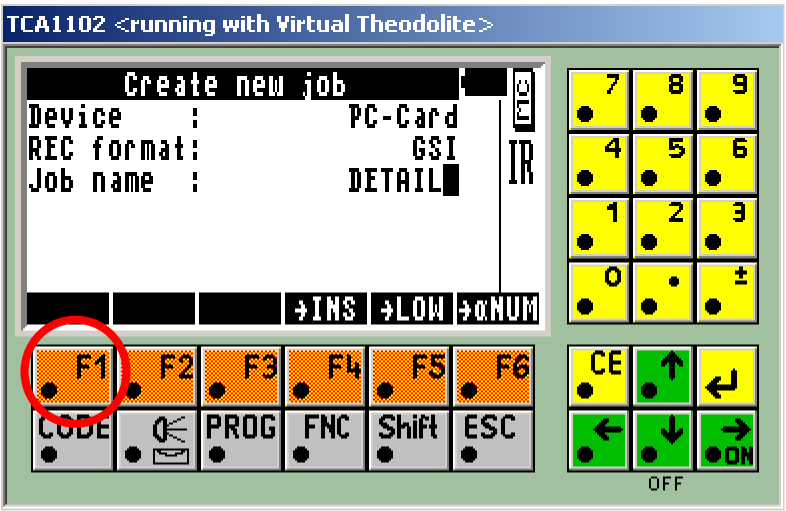


Select 1 to set up Measured job file. This is the file that will contain the detail survey readings

Select F2 to create new file

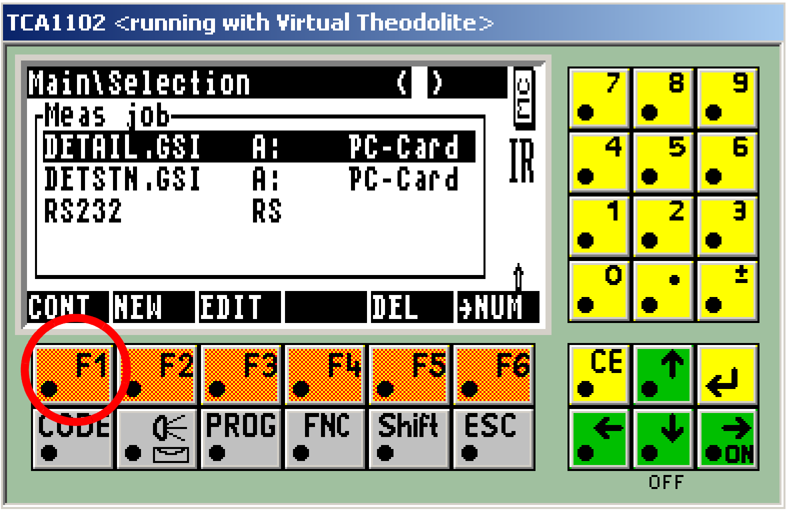


Type in new job name



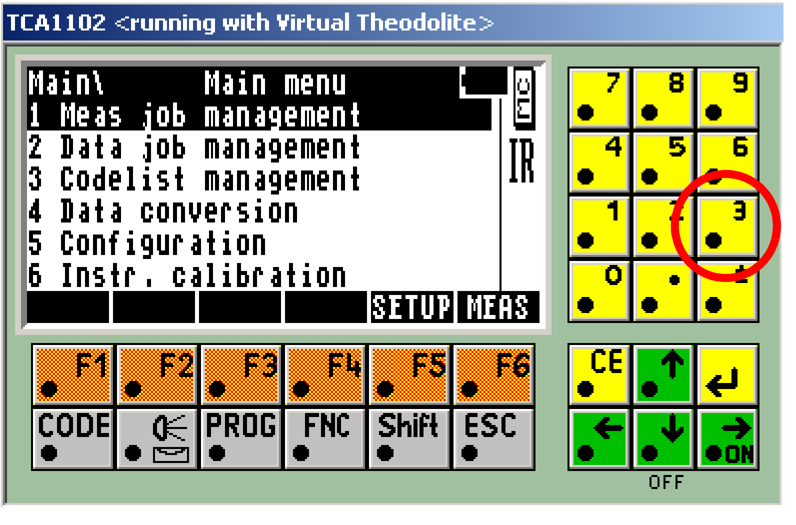
Press F1 to continue

Press F1 to continue

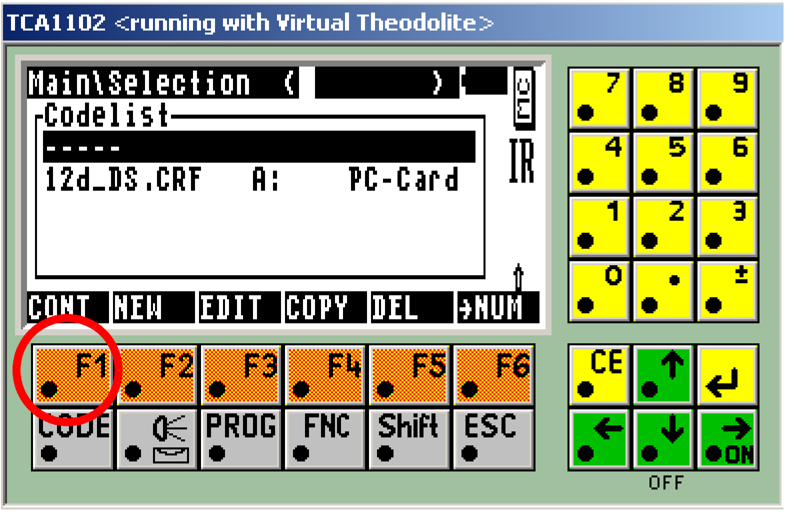


37.1.6.4 Code list selection

Press 3 to select code list



Select the code list 12D_DS.CRF



Press F1 to continue

37.1.6.5 Station setup

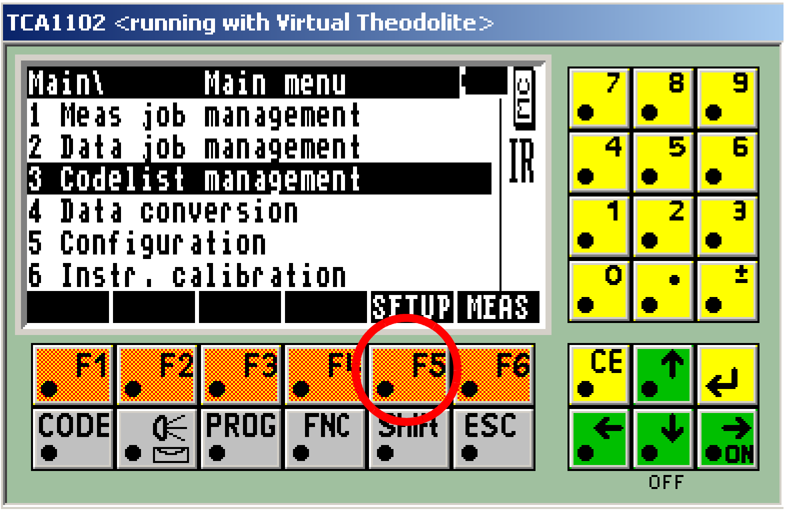
We can use Quick set (See the section37.1.6.6 Setup using Quick set (QSET)) for the station setup or manually select the Station

The following notes assume that the data file contains station coordinates

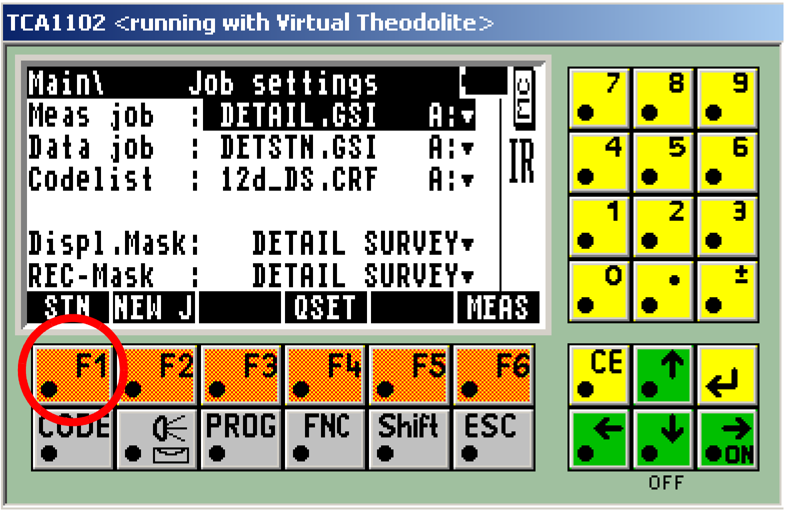
If not the user will have to follow prompts to insert the coordinates

37.1.6.5.1 Manual setup

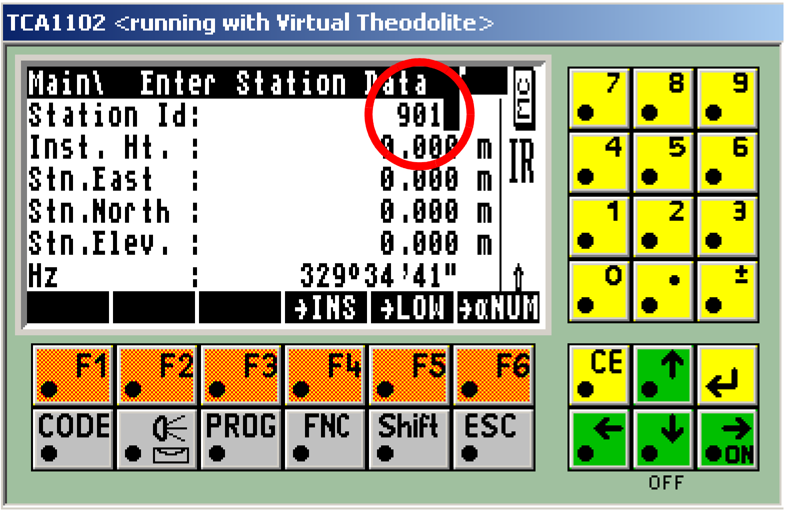
Press F5 for set up



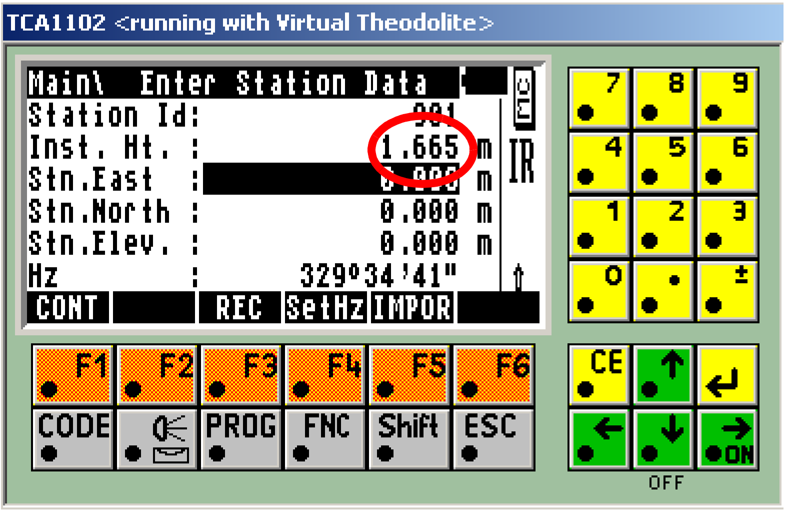
Press F1 to select station



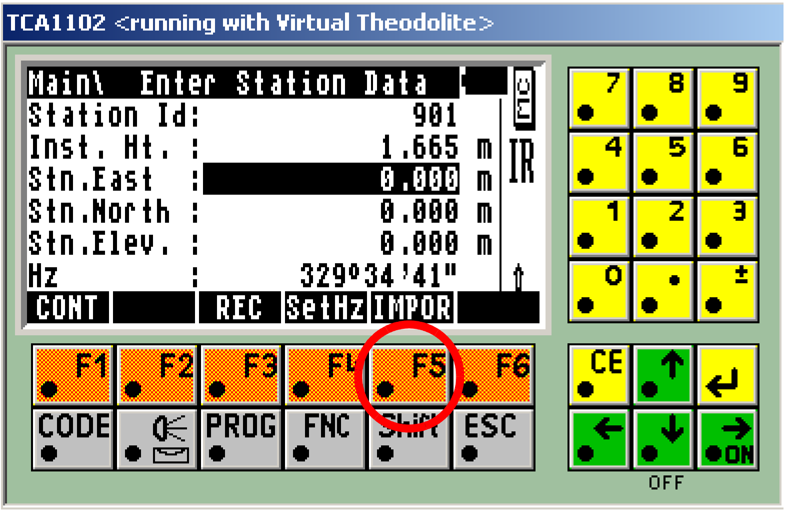
Type in setup station number



Type in the instrument height

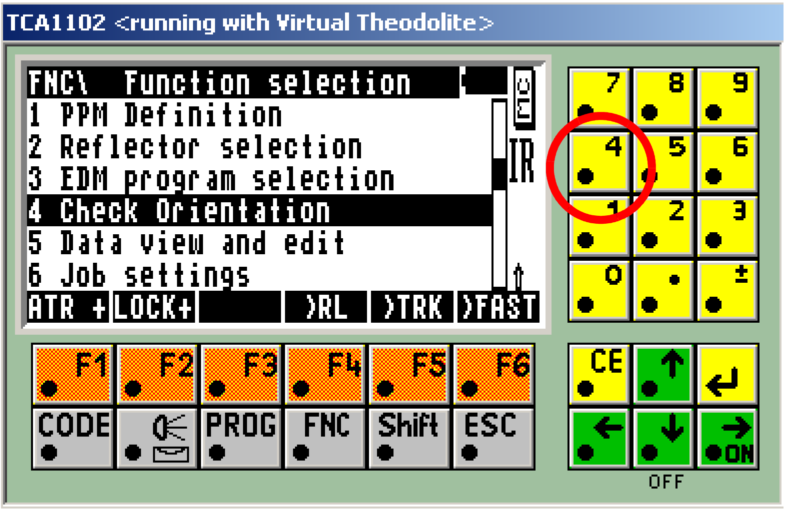


Press F5 to read the coordinates from the data file

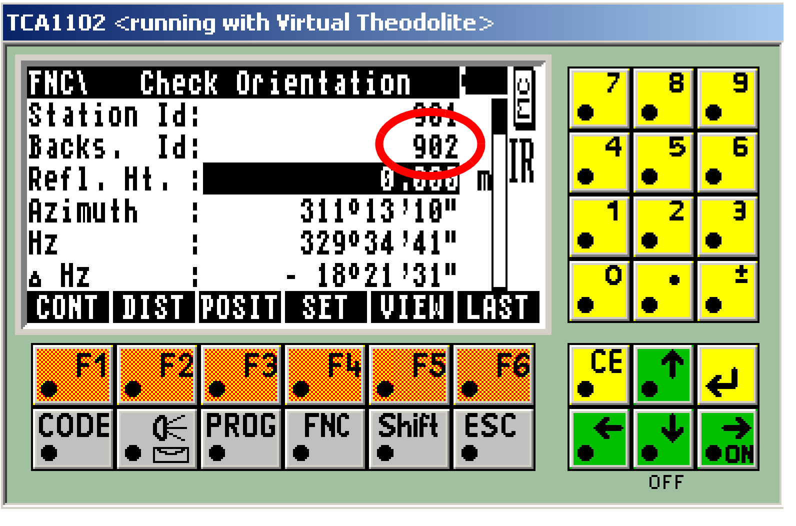


To view the bearing between the setup station and backsight station select FNC

Press 4 to check orientation



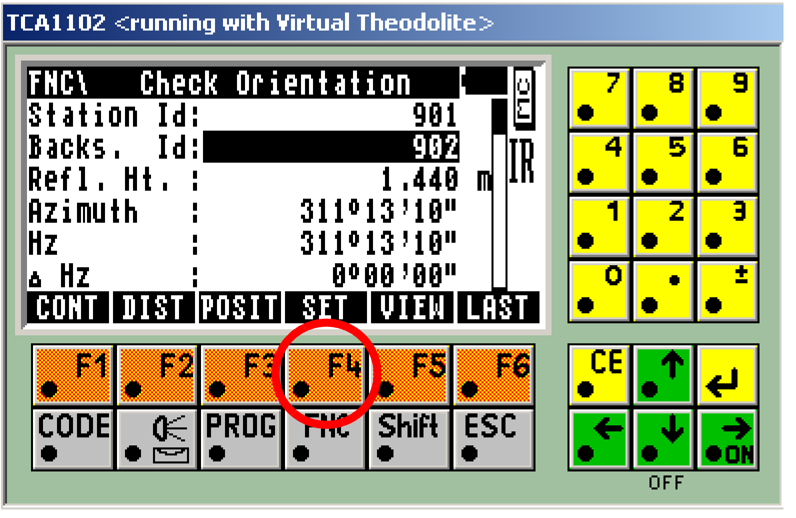
Type in the backsight station number



The Azimuth is displayed

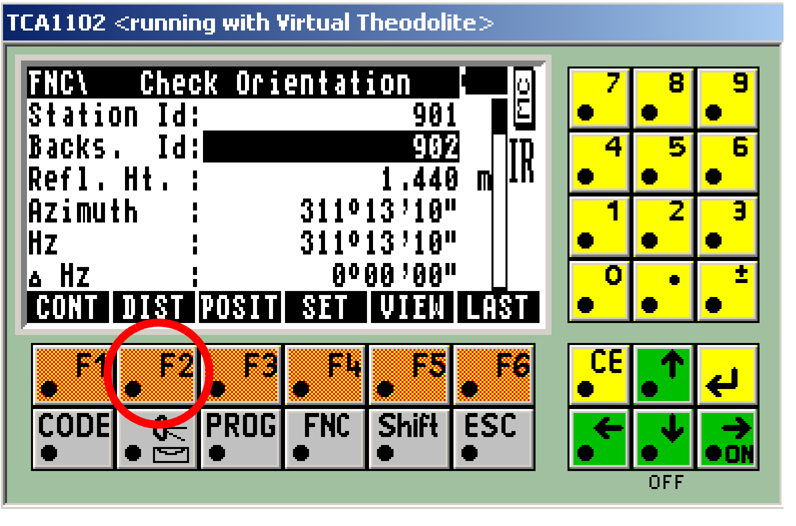
Type in the reflector height of the backsight

Sight to the backsight station and press F4 to set the bearing

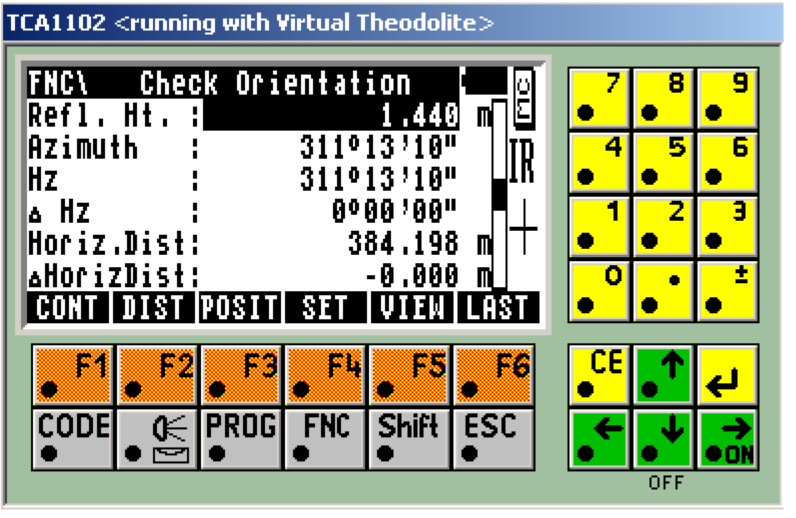


The Bearing is set. Notice that the Azimuth and HZ values are set to the same value.

Press F2 to take a distance to the backsight station

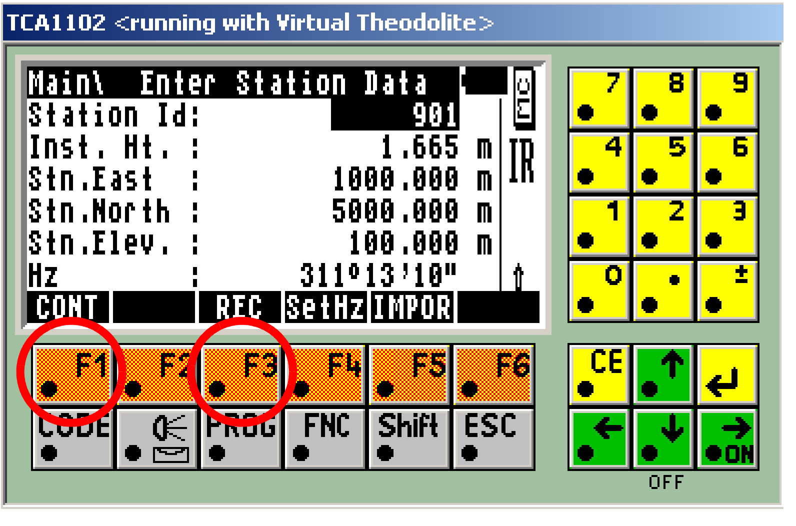


Arrow down to view the distance error



If acceptable press F1 to continue

To save the station information press F3 then F1 to continue

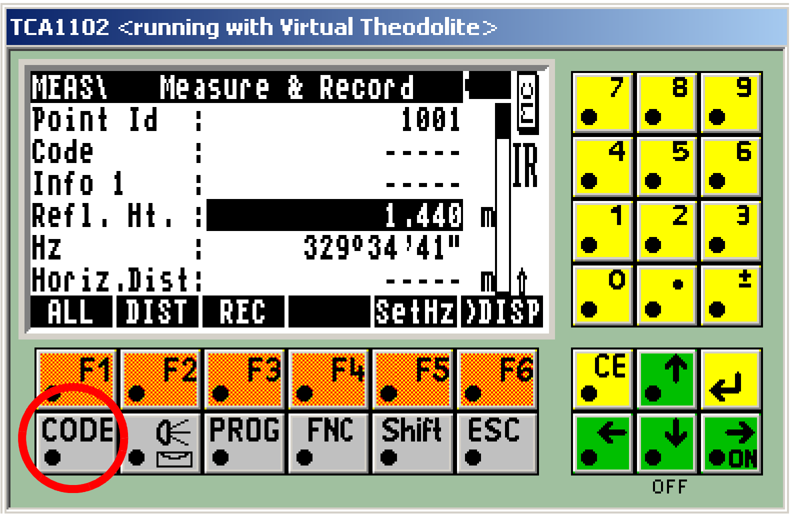


Take Backsight reading:

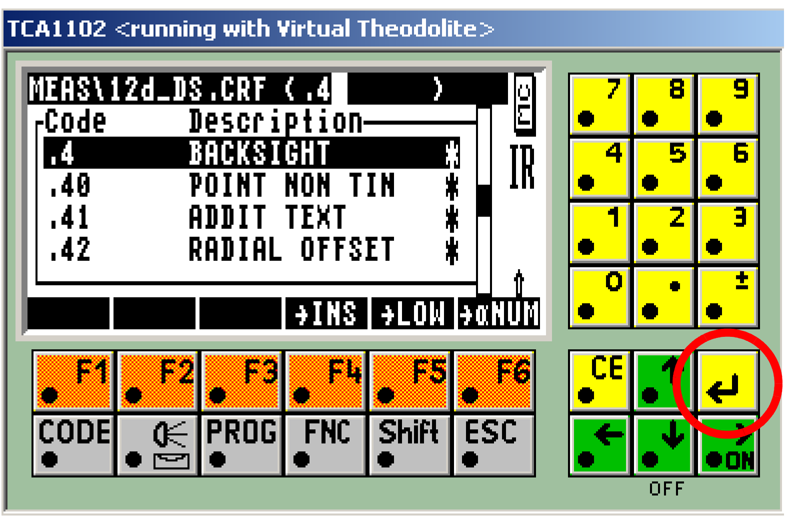
Type in the start point number of the survey

Type in the reflector height. This can also be done as part of the backsight measurement as shown below.

Select CODE

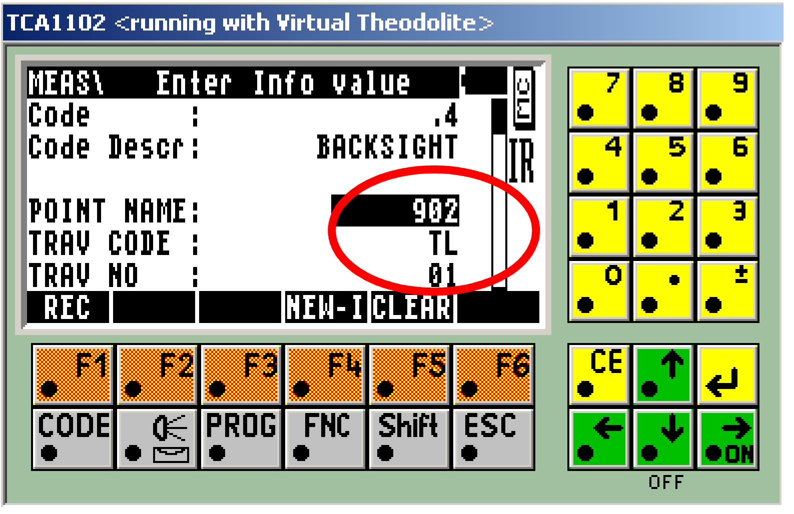


Type in .4 for backsight code followed by the enter key

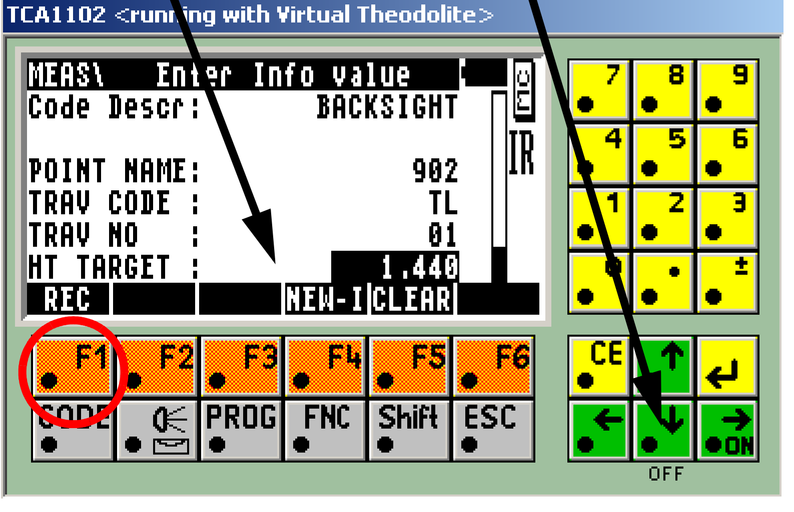


Type in backsight station number

Type in traverse code and traverse no. string number if using traverse coding. For more information about traverse coding see the section 36.6 Traverse coding.

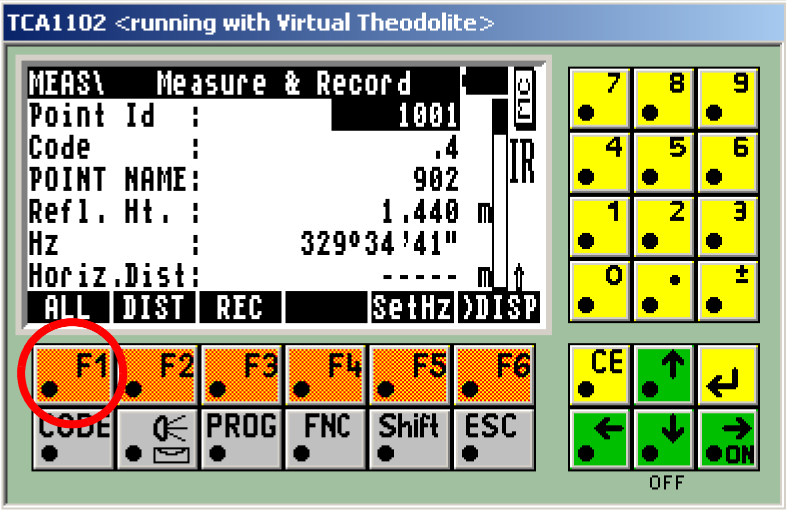


Scroll down using the arrow keys to show other parameters



Press F1 to record the code

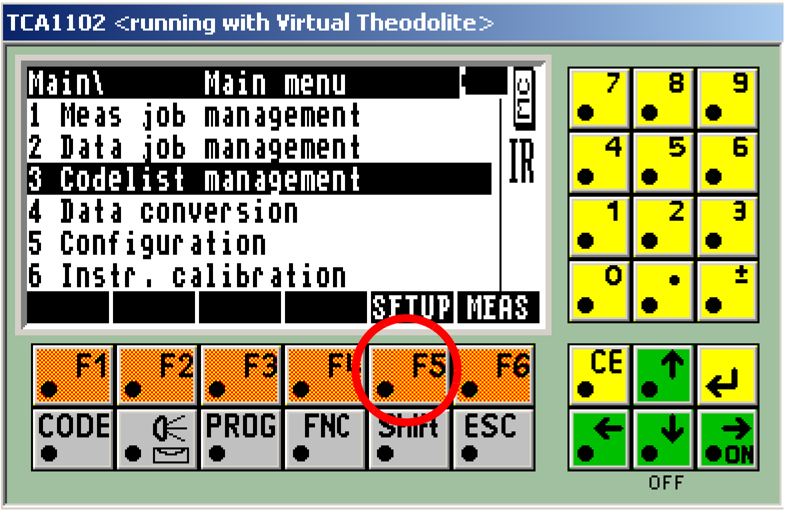
Point to backsight station and press F1 to take and record reading



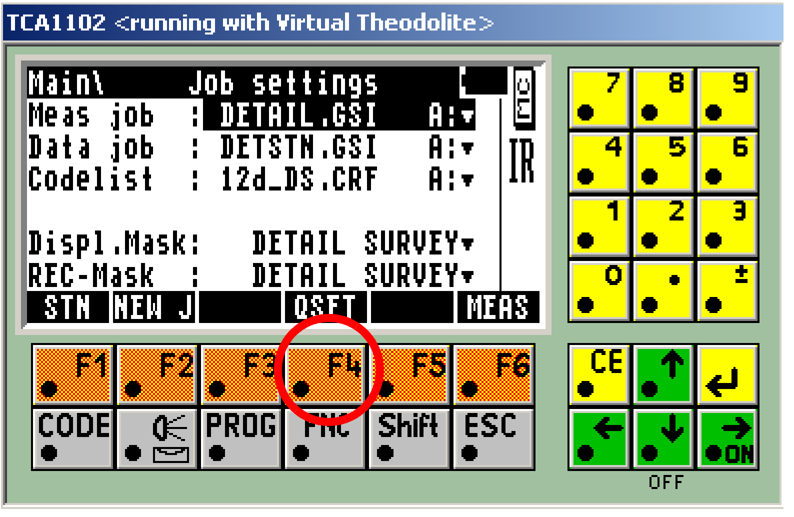
37.1.6.6 Setup using Quick set (QSET)

IMPORTANT NOTE: If using QSET, you MUST take a check reading to your backsight station as described below.

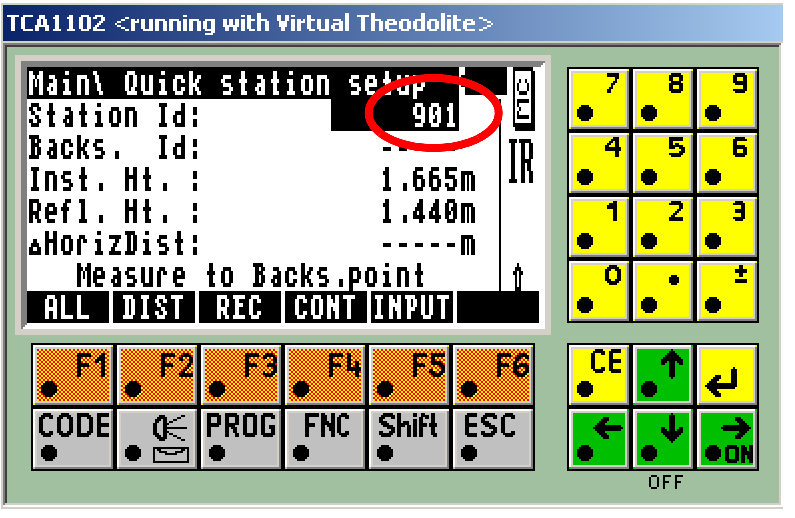
Press F5 for set up



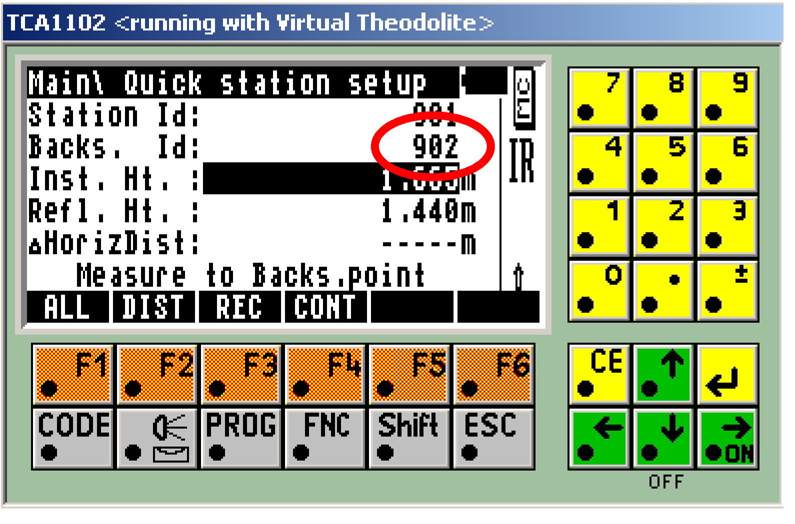
Press F4 for Quick set


TCA1102 <running with Virtual Theodolite>
Main\ Job settings
Meas job : DETAIL.GSI A:
Data job : DETSTN.GSI A:
Codelist : 12d_DS.CRF A:
Displ.Mask: DETAIL SURVEY
REC-Mask : DETAIL SURVEY
STN NEW J QSET MEAS
IR
7 8 9 4 5 6 1 2 3 0 . ±
F1 F2 F3 F4 F5 F6
CODE PROG FNC Shift ESC
CE ON
OFF

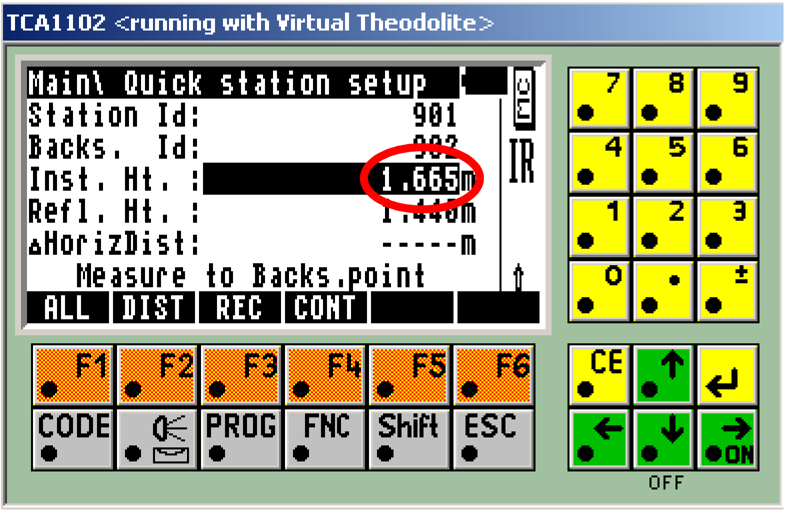
Type in setup station number



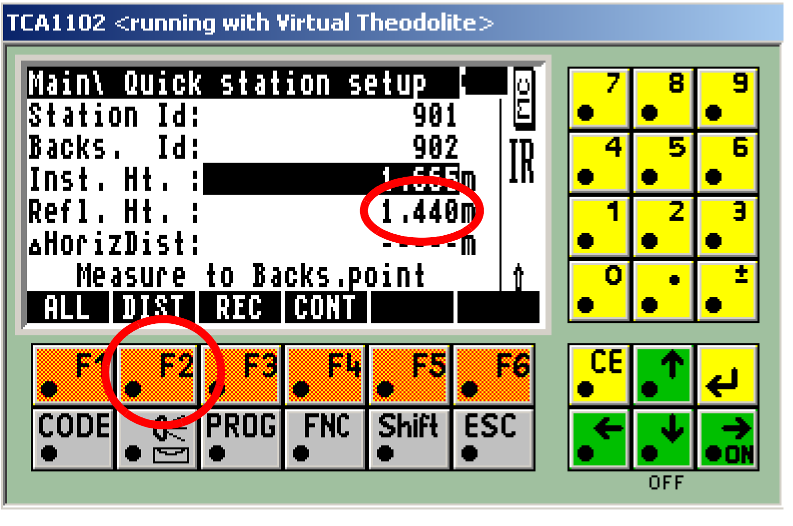
Type in the backsight station number



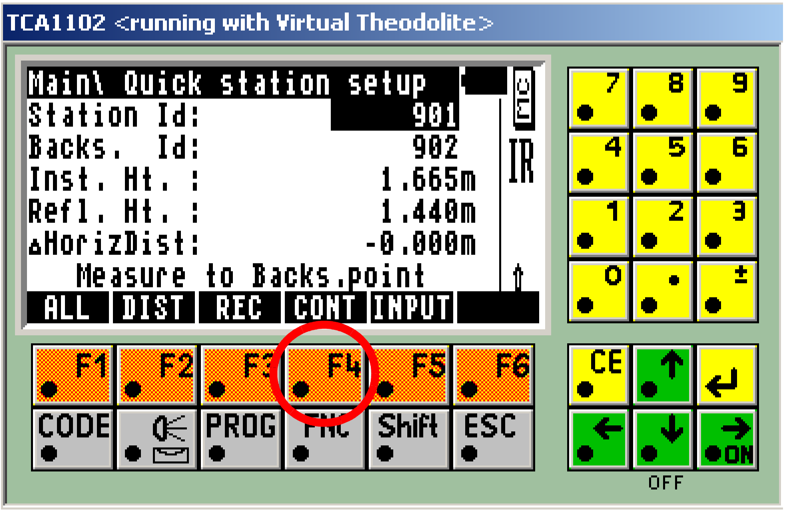
Type in instrument height



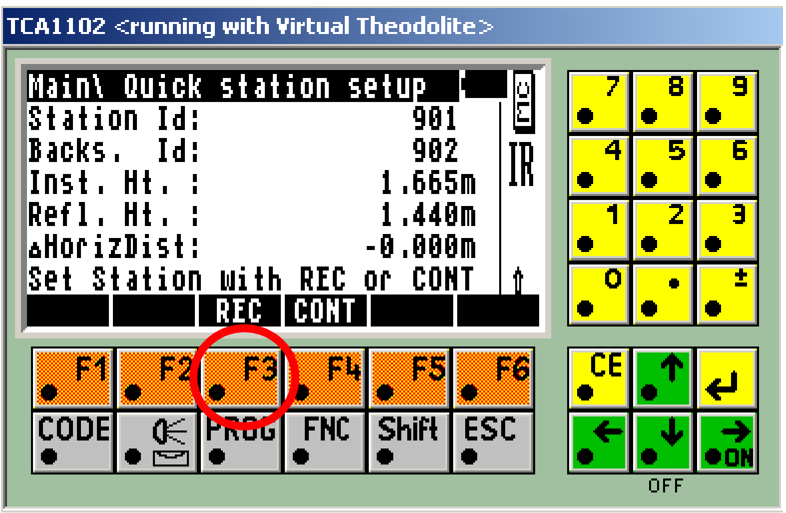
Type in height of target then Press F2 to take reading to backsight to initialise the QSET function.



Press F4 to continue



Press F3 to record the station setup information.

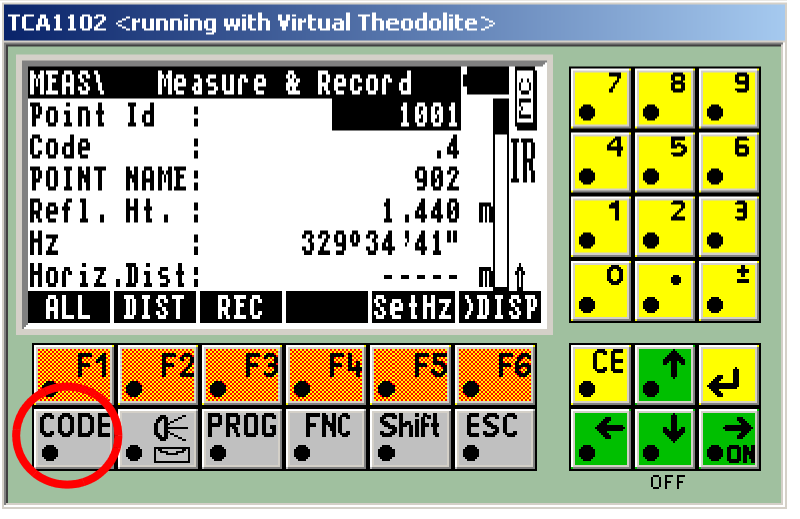


IMPORTANT NOTE: If using QSET, you MUST take a check reading to your backsight station as described below.

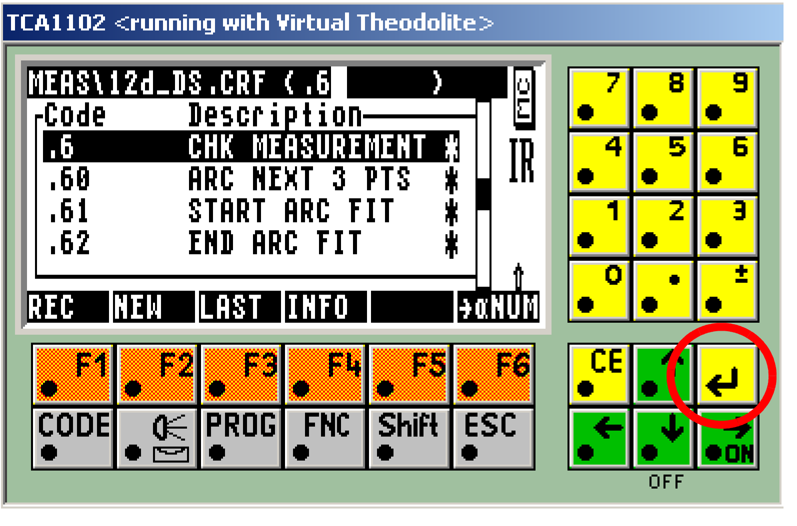
Take Check reading (Mandatory):

Type in reflector height

Select CODE

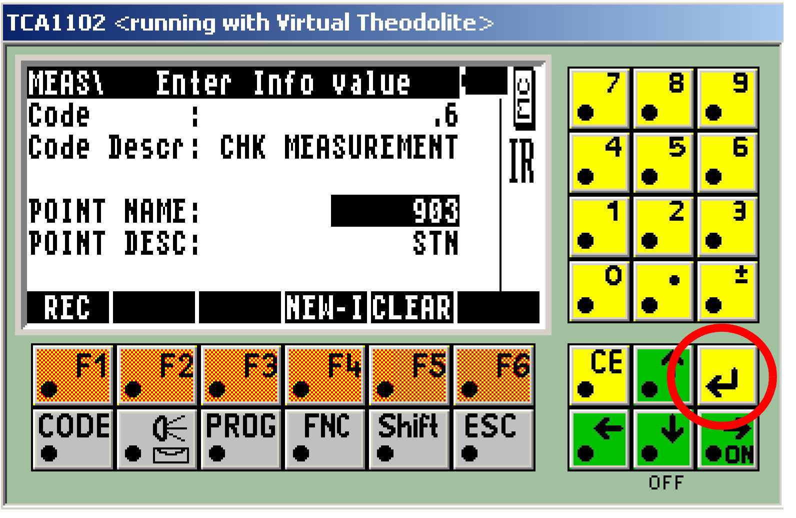


Type in .6 for check measurement code followed by the enter key

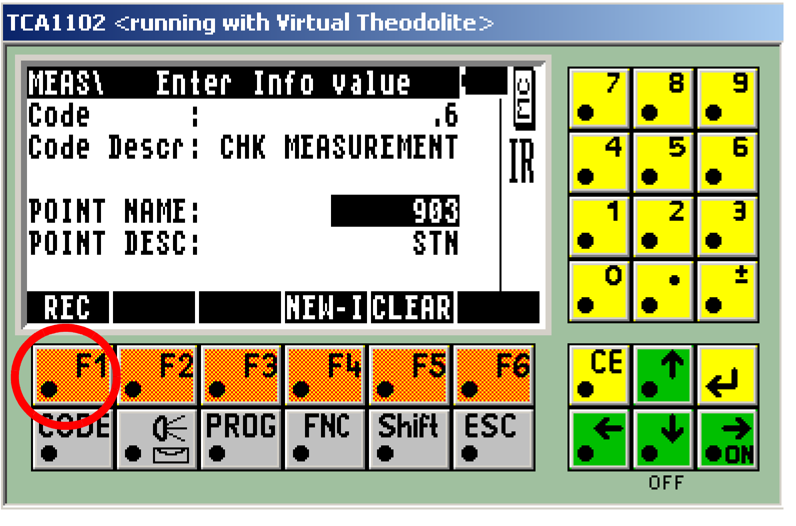


Type in the check station number

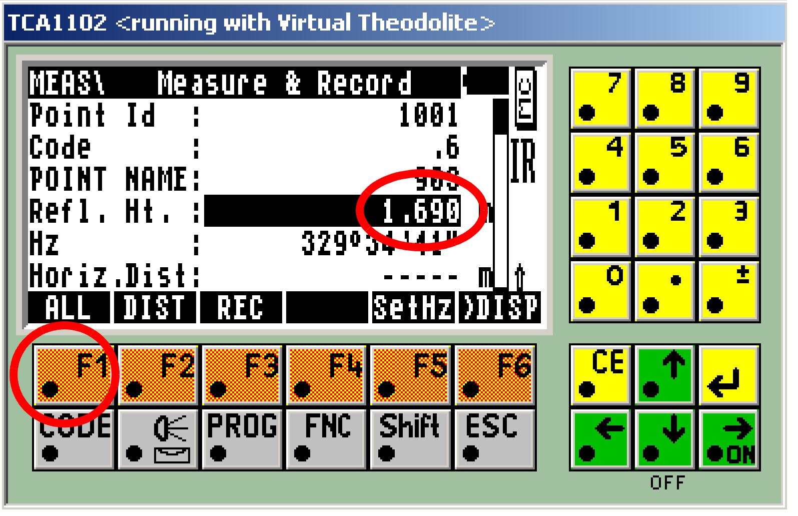
Type in the description



Press F1 to record code



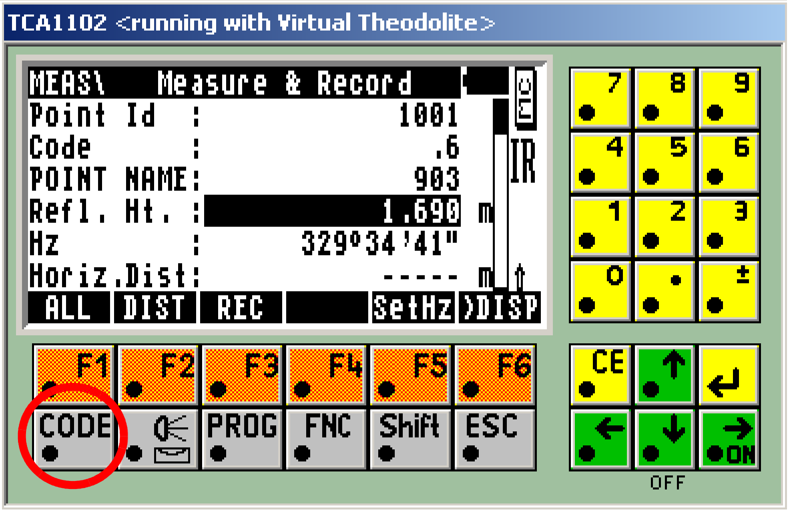
Type in the reflector height



Select F1 to take reading

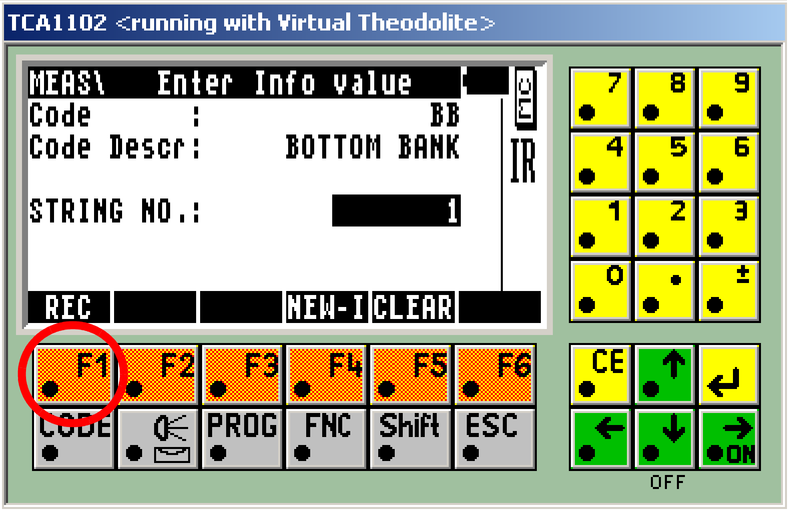
37.1.6.7 Normal readings

Always change the code prior to starting a new feature and ensure the reflector height is correct



37.1.6.7.1 Select Code

Type in the required code and string number

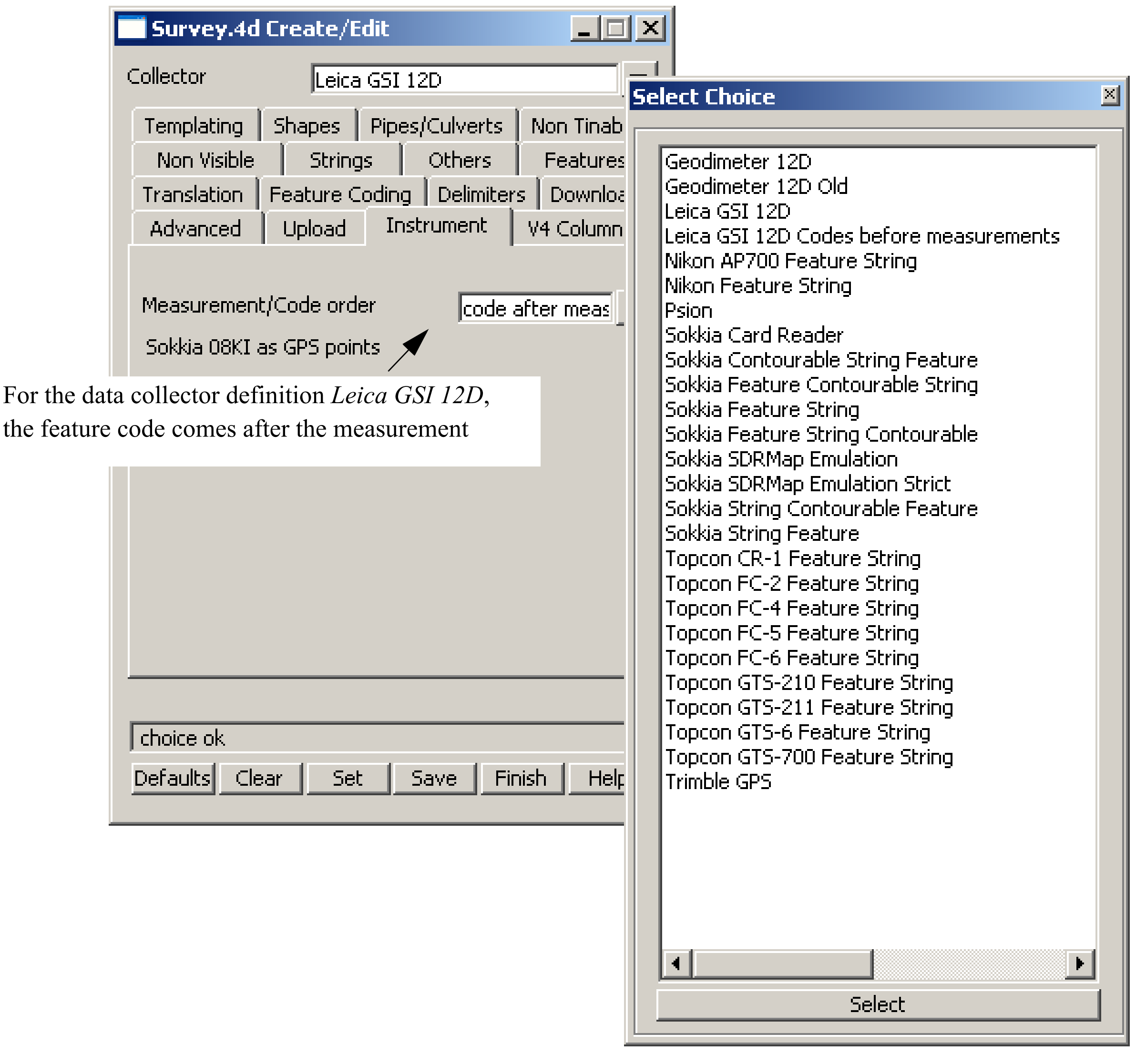


Press F1

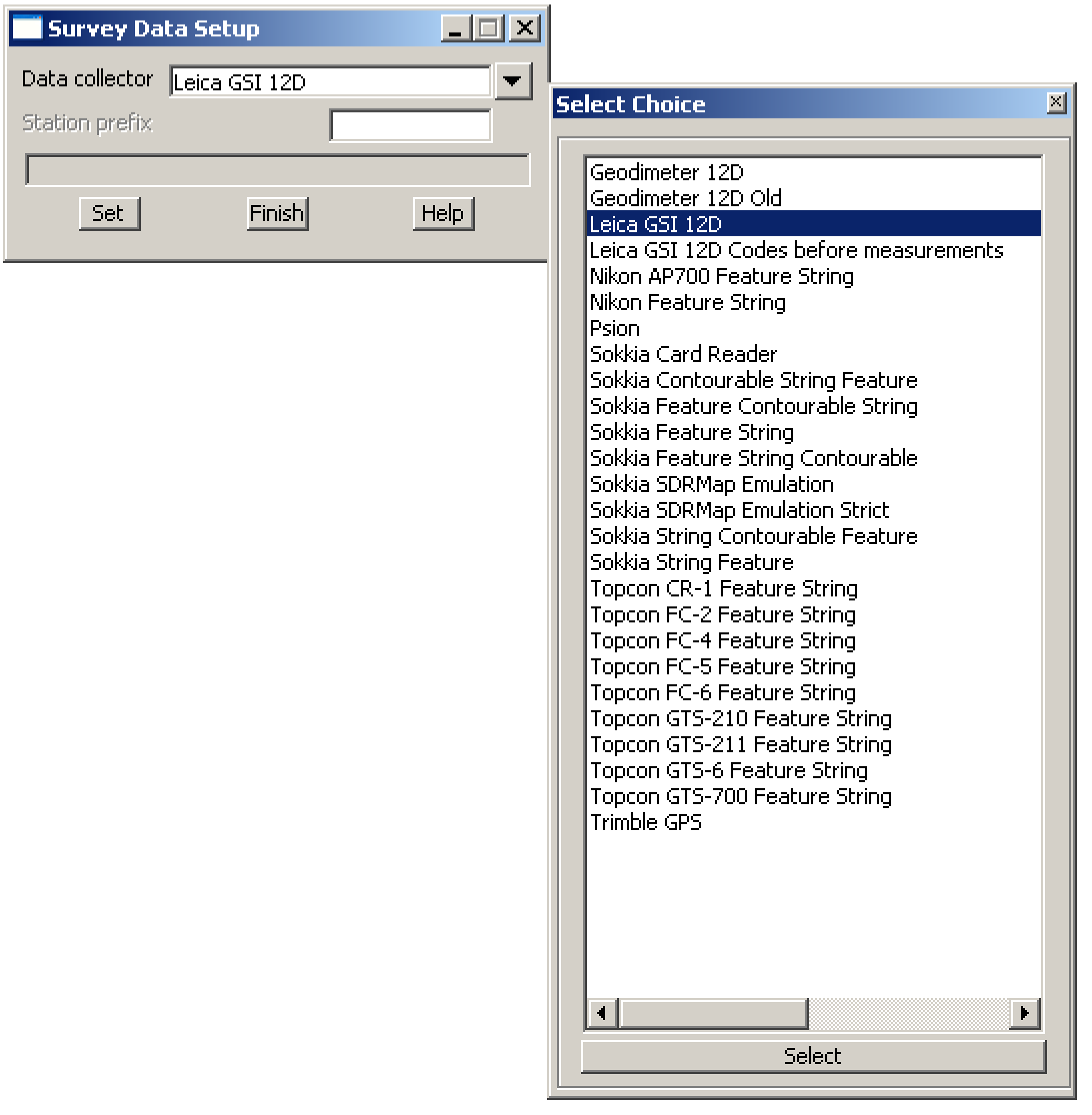
37.1.6.7.2 Example of Leica GSI File

The following GSI file has been coded with the Feature Code being recorded after the measurement. The Feature code definition is then applied to all subsequent measurements until another measurement with a following Feature Code definition is found.

The data collector definition *Leica GSI 12D* which is shipped with ***12d Model***, is an appropriate data collector definition for converting the example GSI file to a 12d Field File ready for processing.



The data collector definition *Leica 12D* which is shipped with ***12d Model*** is selected using the option *Survey=>Setup*


Survey Data Setup
Data collector
Leica GSI 12D
Station prefix
Set
Finish
Help
Select Choice
Geodimeter 12D
Geodimeter 12D Old
Leica GSI 12D
Leica GSI 12D Codes before measurements
Nikon AP700 Feature String
Nikon Feature String
Psion
Sokkia Card Reader
Sokkia Contourable String Feature
Sokkia Feature Contourable String
Sokkia Feature String
Sokkia Feature String Contourable
Sokkia SDRMap Emulation
Sokkia SDRMap Emulation Strict
Sokkia String Contourable Feature
Sokkia String Feature
Topcon CR-1 Feature String
Topcon FC-2 Feature String
Topcon FC-4 Feature String
Topcon FC-5 Feature String
Topcon FC-6 Feature String
Topcon GTS-210 Feature String
Topcon GTS-211 Feature String
Topcon GTS-6 Feature String
Topcon GTS-700 Feature String
Trimble GPS
Select

Description	Leica GSI File in 8 Format using 12d Model Field Coding and Giving Feature Codes After a Measurement
Directly entered co-ords for station 10	410001+000000.2 42....+10000000 43....+10000000 44....+00010000 45....+00000STN 46....+00000000 47....+00000000 48....+00000010
Instrument set-up on 10 with instrument height of 1.715	410002+000000.3 42....+00000010 43....+0001.715
Next measurement is backsight to 931	410003+000000.4 42....+00000931
Measurement to backsight - has point number 1765	110004+00001765 21.324+19135130 22.324+09010170 31..00+00091100 51..1.+0007+000
New target height of 1.338	410005+000000.5 42....+0001.338
Measurement FC DHW SN 0 - Pt no 1766 (the FC and SN is given on the next line)	110006+00001766 21.324+19316250 22.324+08955240 31..00+00074468 51..1.+0007+000
Set feature code DHW string number 0	410007+00000DHW 42....+00000000
Measurement - FC RDG SN 1 has point number 1767	110008+00001767 21.324+21458540 22.324+09150580 31..00+00019501 51..1.+0007+000
Set feature code RDG string number 1	410009+00000RDG 42....+00000001
Measurement - FC RDG SN 1 has point number 1768	110010+00001768 21.324+21820190 22.324+09137140 31..00+00020397 51..1.+0007+000
Measurement - FC RDG SN 1 has point number 1769	110011+00001769 21.324+22104470 22.324+09155360 31..00+00019241 51..1.+0007+000
Measurement - FC RDG SN 1 has point number 1770	110012+00001770 21.324+21747290 22.324+09208580 31..00+00018299 51..1.+0007+000
Close the previous string RDG	410013+00000.20

37.2 Sokkia Instruments (SDR Files)

This Section deals with interfacing 12d with Sokkia SDR20/33 data formats specifically with the SDR33 electronic field book (Controller). Other controllers and instruments such as the powerset range of total stations also use these formats.

A large number of instruments support the Sokkia SDR data format including (Trimble Total Station ACU), not just those from Sokkia.

For the topic:

1. Field Coding see the section 37.2.1 Coding For SDR Files
2. Sending raw file from controller to 12d 37.2.1.1.1 Downloading SDR File To 12d
3. Converting raw data to field file see the section 37.2.1.1.2 Converting SDR Raw File To Field File
4. Creating Points upload file see the section 37.2.1.1.3 Creating SDR Point Upload File
5. Creating Road upload file see the section 37.2.1.1.4 Creating SDR Roads Upload File
6. Creating Tin upload file see the section 37.3.7 Create Tin Upload File

Also, SDRMap users should also read the section 37.2.1.1 Special Notes for SDRmap Users

37.2.1 Coding For SDR Files

The Sokkia SDR20/33 data formats are used as raw data files by 12d Model and are converted into a 12d Field File before reduction. ***12d Model*** uses the following Sokkia records:

01	Record header - SDR format
02	Station details
03	Target height
07	Back bearing details
08	Position - directly entered co-ordinates
09	Observation - measurement
13	Comment or continuation of blocks
15	RTK station details
16	RTK observation - measurement
57	RTK antenna height

Users can enter text for each measurement (observation 09 record or position 08 record) which is appended to the end of the record and this is used as the text of blocks that are interpreted according to the descriptions given in the earlier section 36.8 Field Coding for Non Leica Instruments.

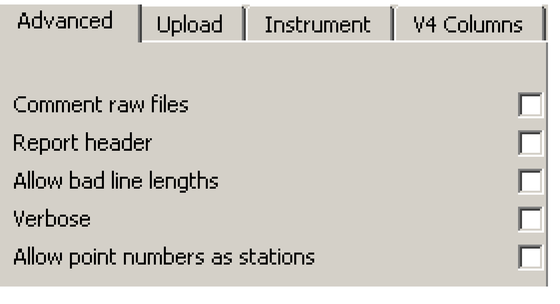
The '13' record can also be used after a measurement record to add additional information to the preceding blocks using the *extra coding* control code at the end of the previous line (see Extra Codingin the section 36.8.4 Control Code Blocks

Strictly speaking the Sokkia SDR20/33 formats use fixed length lines and if the lengths are incorrect, an error message will be written to the Output Window. For example,

'Line 248 line incorrect length. required length is 58. received length is 50.'

These messages often appear after a raw file has been manually edited because most editors remove space padding at the end of a line.

To ignore such error messages, there is a setting *Allow bad line lengths* on the *Advanced* tab of the **Survey.4d Create/Edit** panel. If this is set for a data collector, then lines of incorrect length will not be rejected.



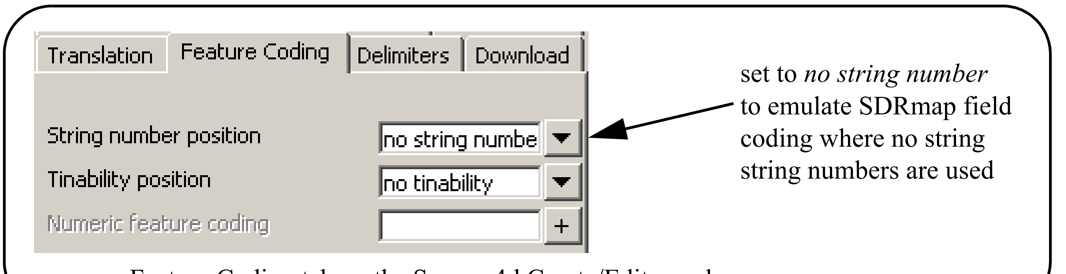
For non SDRmap users, please proceed to the section 37.2.1.1.1 Downloading SDR File To 12d.

SDRmap users, please continue to the next section 37.2.1.1 Special Notes for SDRmap Users.

37.2.1.1 Special Notes for SDRmap Users

SDRmap does not use String numbers - only Feature codes are used and a *New String* command to denote the start of a new string.

12d Model supports ***only using a Feature code*** by setting the *String number position* on the *Feature Coding* tab to ***no string number***.



Feature Coding tab on the Survey.4d Create/Edit panel

If *no string number* is set for the data collector (and so no string numbers are used in the field), then there is no way of telling from the field data whether the string is a *point* string or a *line* string (setting the pt-line type for the string).

For the *no string number* case, the method of defining the point-line type is to:

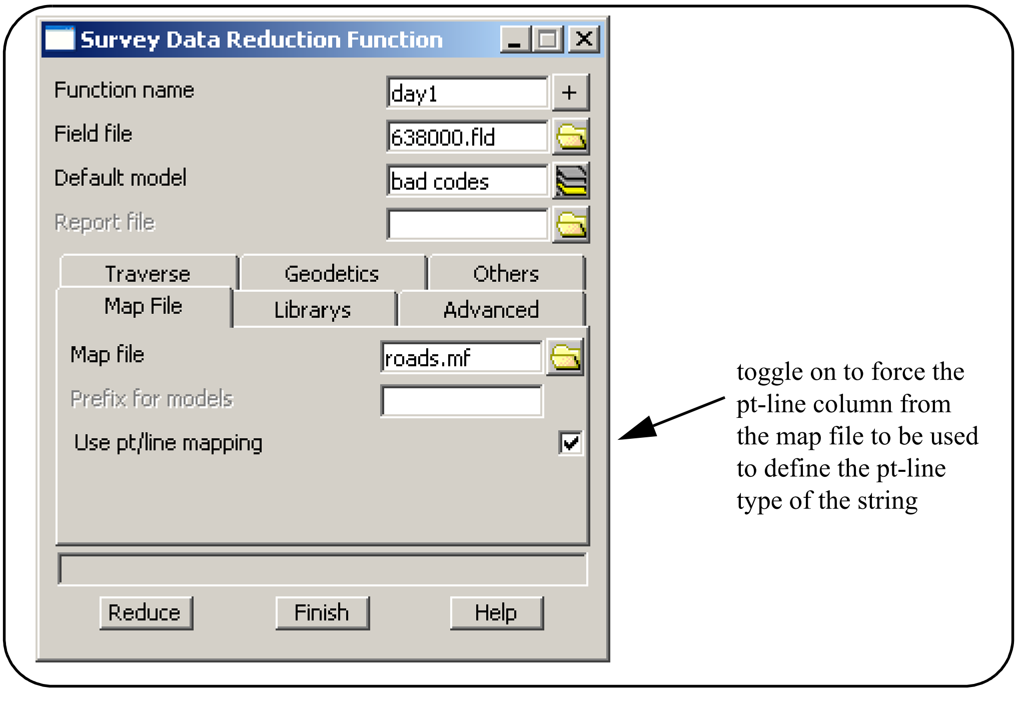
(a) use a map file during the reduction

and

(b) have a key in the map file to match the feature code *and* have the pt-line type set for that key. If the *feature code* does not match any key in the map file, the string defaults to a *point string*.

and

(c) because by default the pt-string column in the map file is *ignored* during reduction (the string number usually defines the pt-line type), the *Use pt/line mapping* tick box in the Survey Data Reduction Function panel must be set to tick to *force* the pt-line column in the mapping file to be used.



Please continue to the next section 37.2.1.1.1 Downloading SDR File To 12d.

37.2.1.1.1 Downloading SDR File To 12d

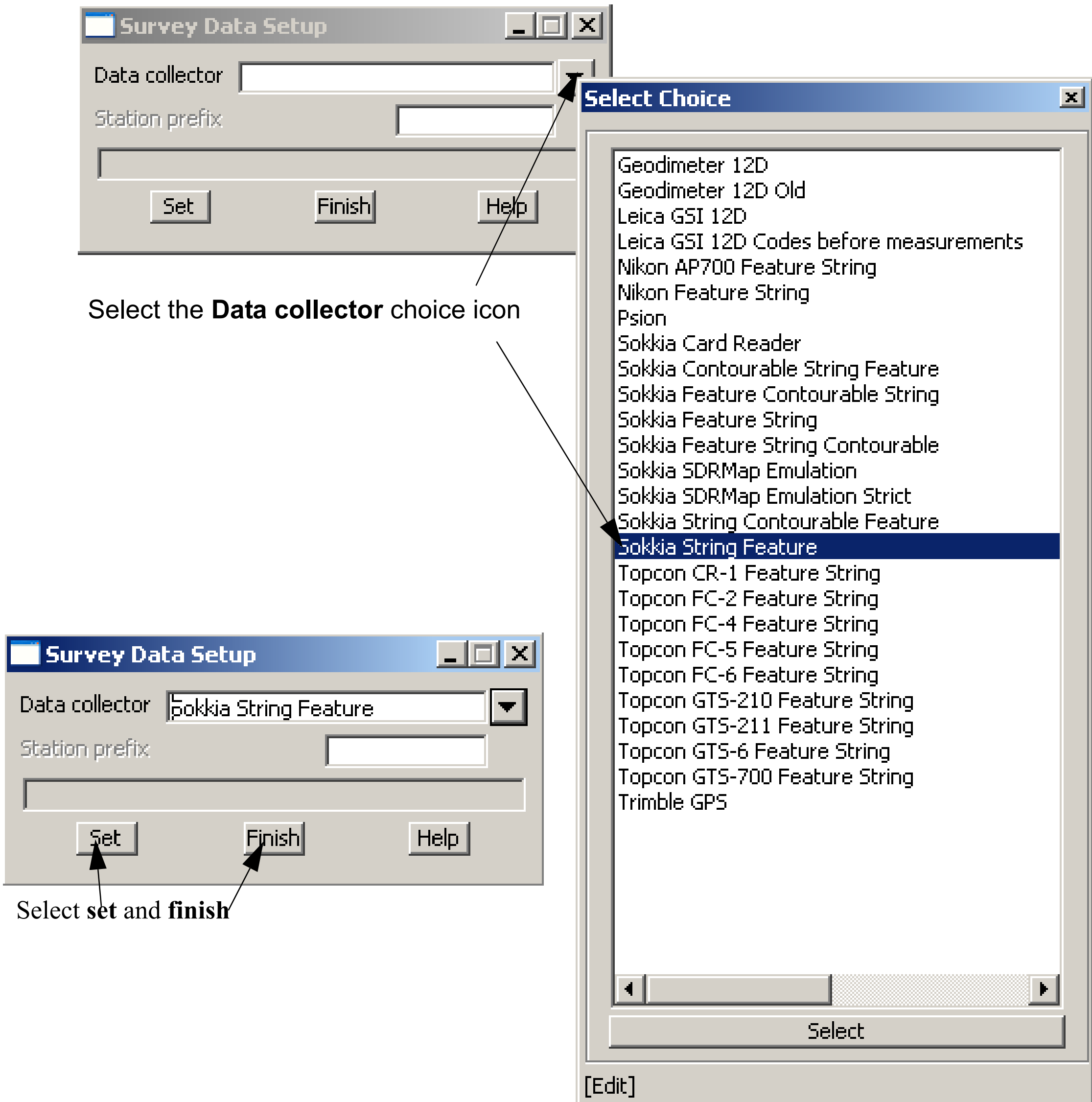
Raw SDR files can be input to 12d for reduction by two basic methods:

1. Download directly from the instrument
2. Copy the SDR file to the working folder directly from a PCMCIA card or by using a data transfer product such as "HyperTerminal" or "Wincomms". For this method, see the following section 37.2.1.1.2 Converting SDR Raw File To Field File.

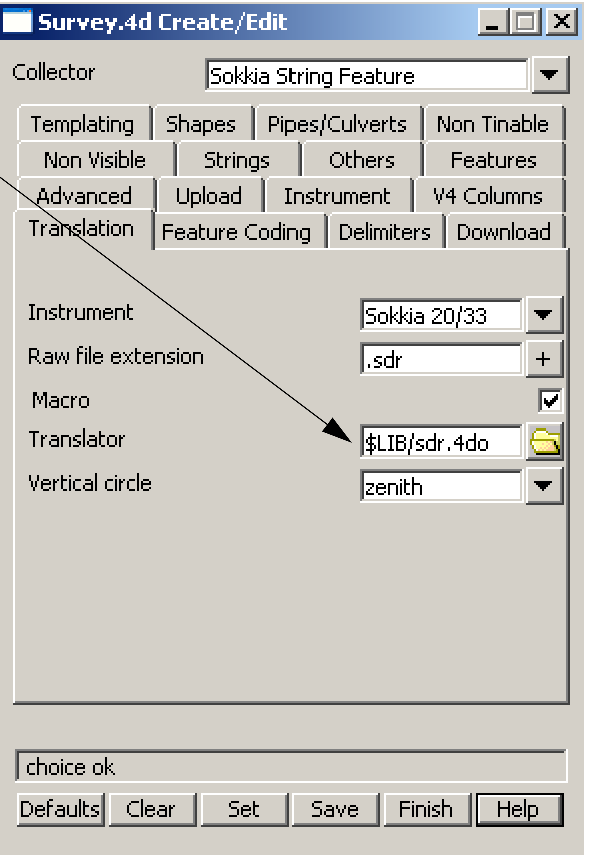
An option exists within 12d to download the raw SDR file from the instrument. This option has the added functionality that is reduces the raw propriety format (SDR file) to a 12d field file format at the same time. This conversion requires that the data collector definition be set before the download of the file is commenced.

Select the data collector type

Select option ***Survey=>Setup***



The raw file will be converted to the 12d field file format using the specified translator macro

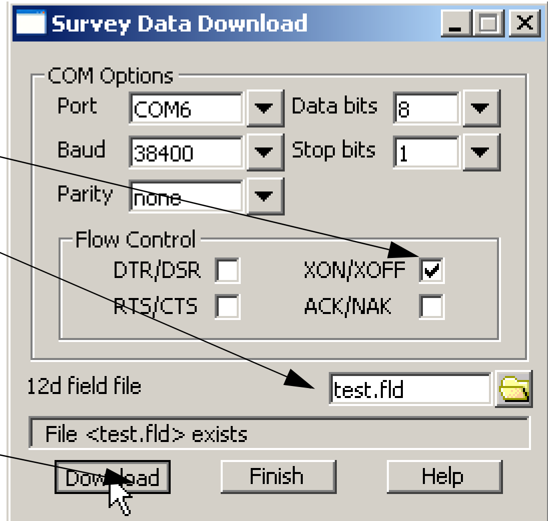


Select ***Survey=>Download***

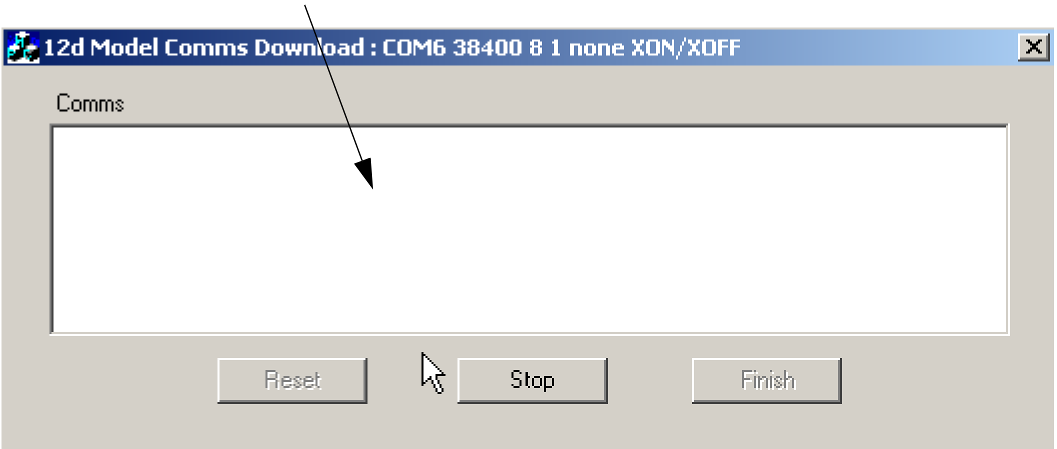
Ensure that the correct values are set for the comms settings i.e. they match the data collector parameters

Enter a name for the field file

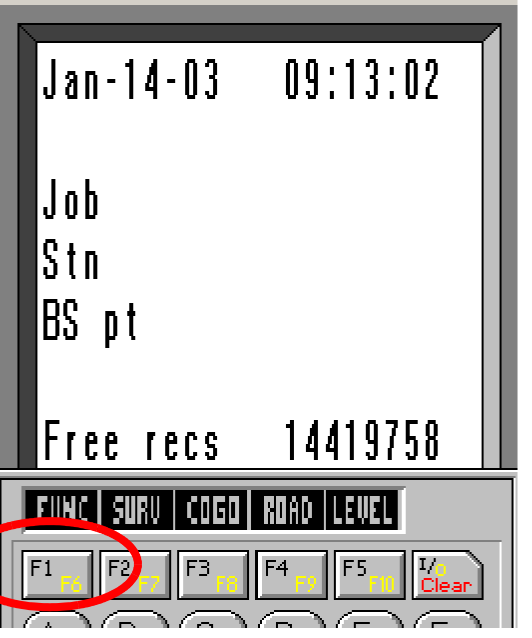
Press **download**



The 12d download window is shown.

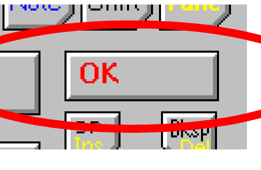


Prepare the data collector for download by connecting to appropriate port, selecting file to download etc.

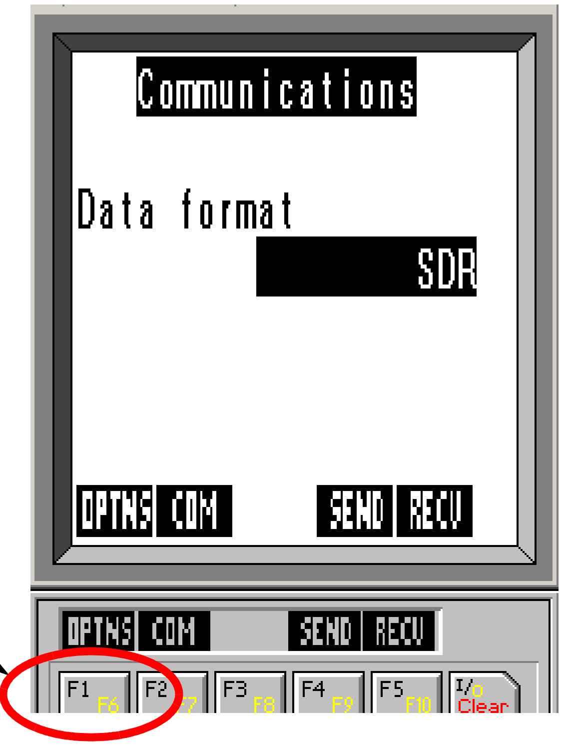


From the Data collector main menu select the **FUNC** (function) menu

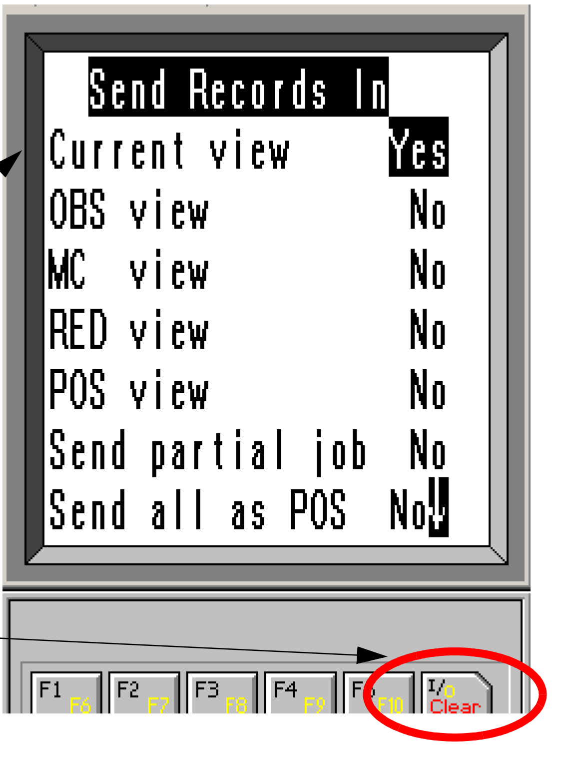
Toggle down to the **Communications** menu



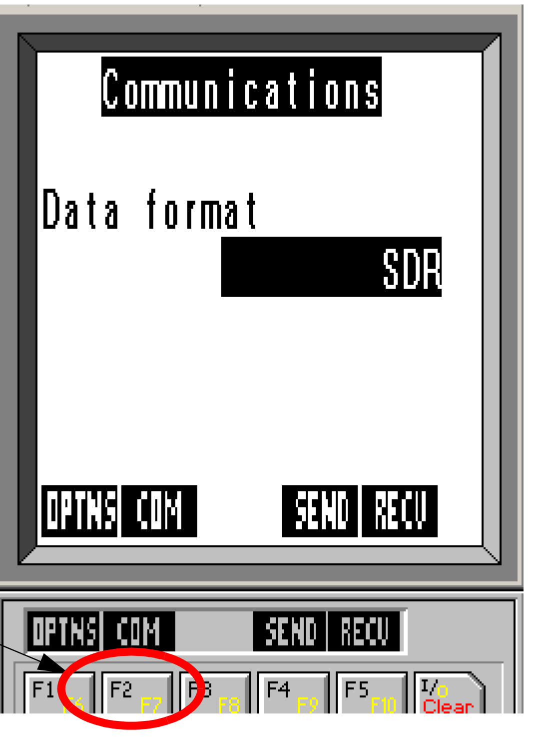
Press **OK** to select menu



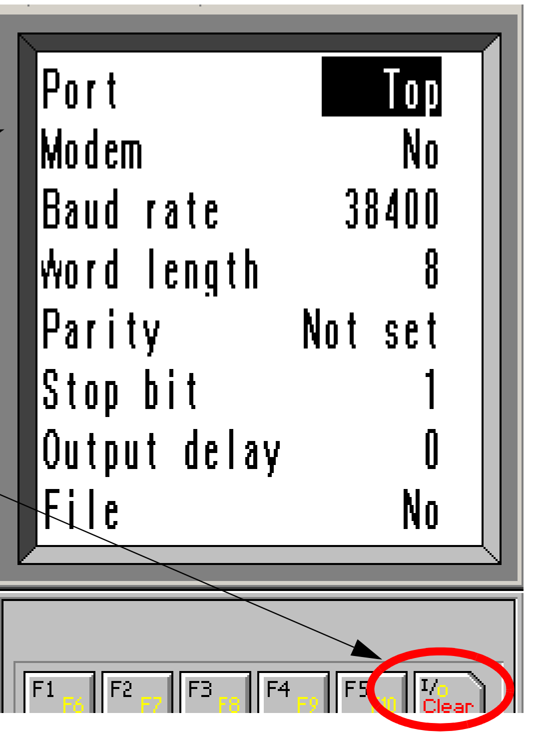
Select the **Options** menu to view the current settings



Select the **Current view** to **Yes** only. If the POS view were set to **Yes** also, the POS points would be sent also and eventually reduced twice in 12d (once from observations and the other from the POS record). Press **Clear** to leave menu



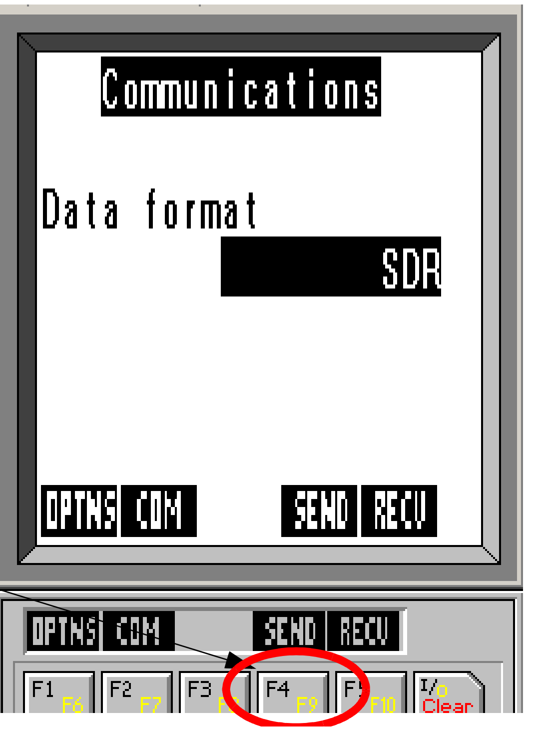
Select the **COM** menu to view the current comms settings



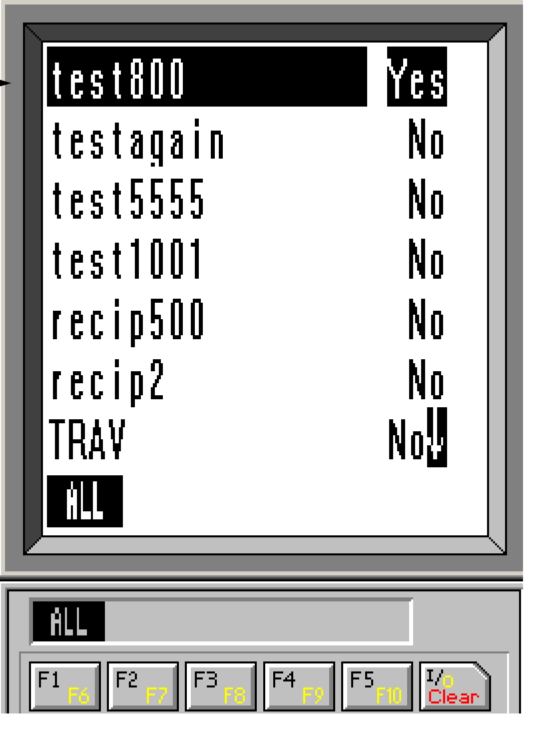
Ensure the comms settings are as per the values set in the 12d download option.

Select **Clear** to leave the menu

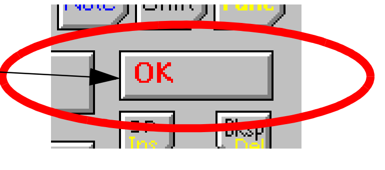
Select the **SEND** menu



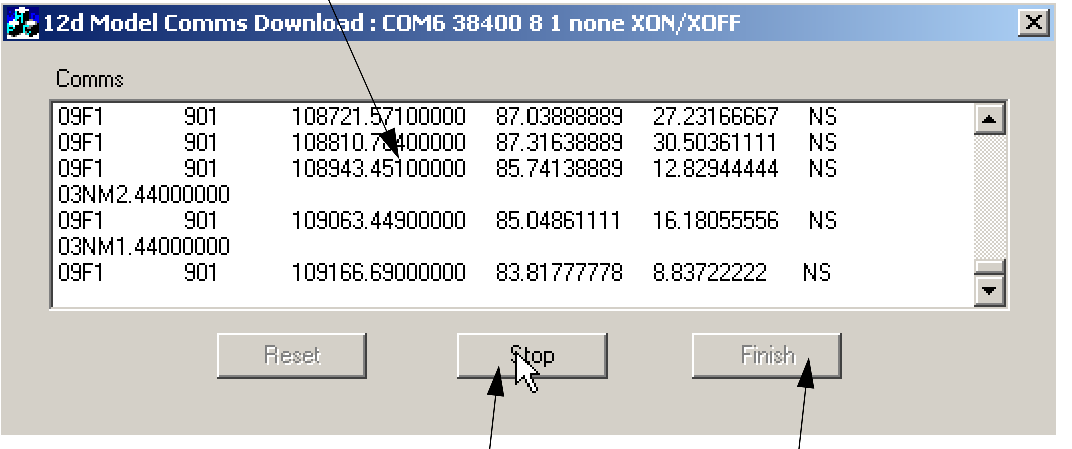
Select the file to download



Press **OK** to send



The file is shown downloading line by line in the 12d download window.



When the download is finished select the **stop** button and then **finish**

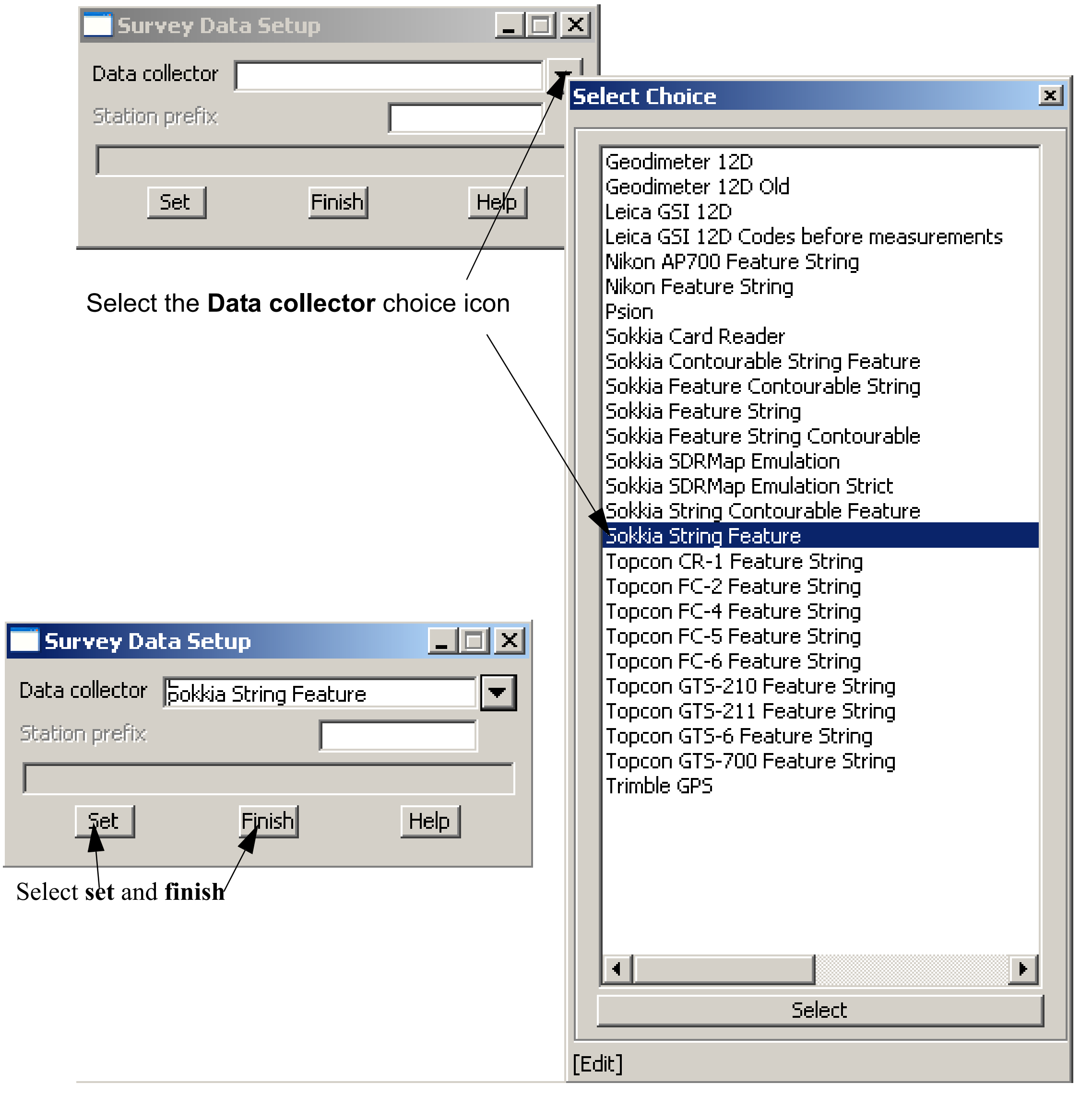
Please continue to the next section 37.2.1.1.2 Converting SDR Raw File To Field File

37.2.1.1.2 Converting SDR Raw File To Field File

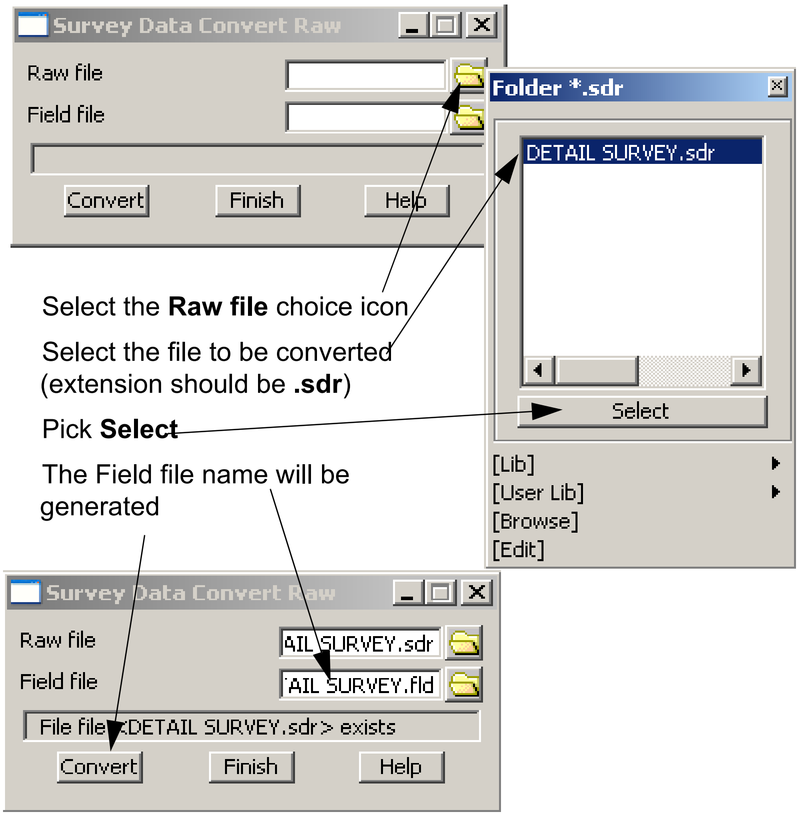
If a raw SDR file is copied to the working folder from a PCMCIA card or other means such as transferring it from a third party software package, it must be converted to a 12d field file for reduction inside 12d.

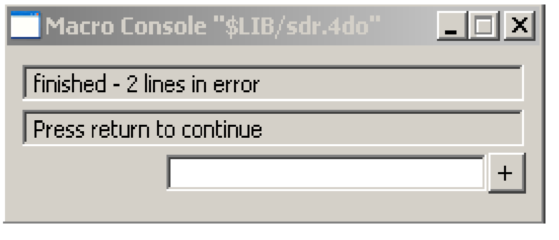
Select the data collector type

Select option ***Survey=>Setup***



Select option ***Survey=>Convert raw***





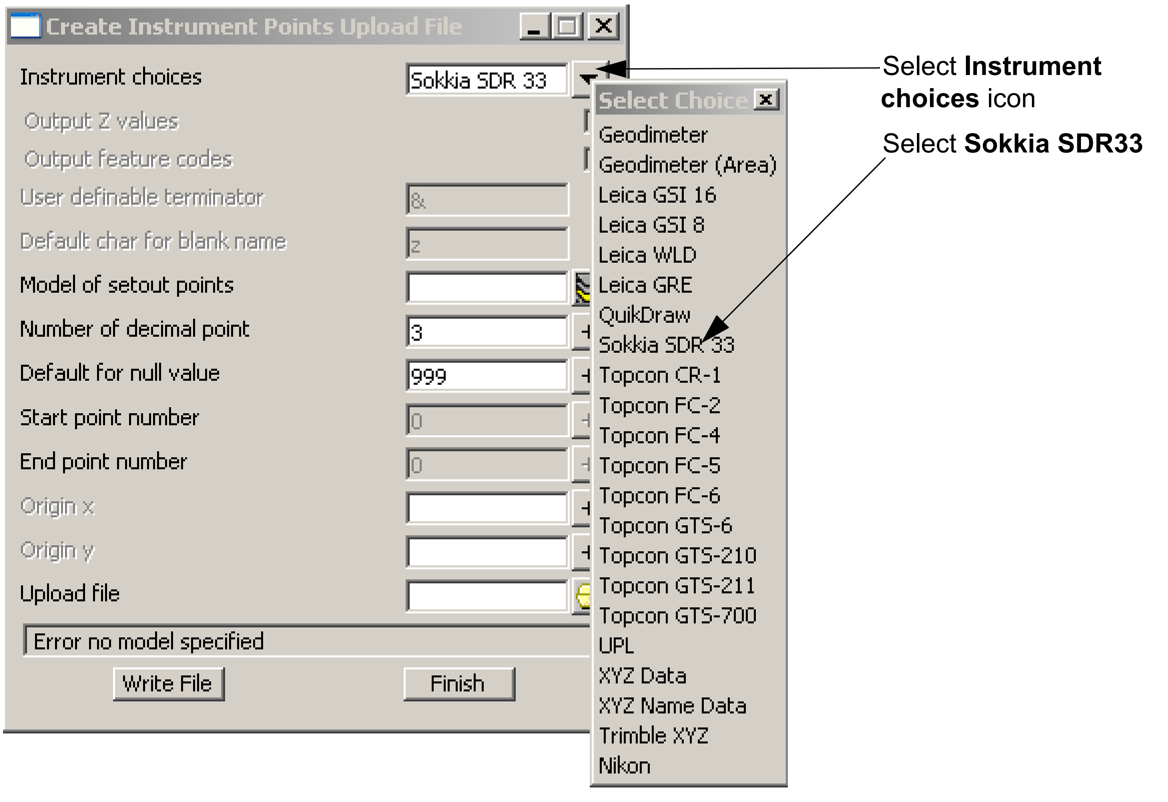
Any errors will be listed in the Output window. If the Output window is not shown, it can be opened using the option ***Window=>Output Window***

37.2.1.1.3 Creating SDR Point Upload File

Create upload file in 12d

After creating the setout points an upload file can be created for the sdr format

Select option ***Survey=>Upload=>Create points upload file***



Fill in the rest of the screen as per normal

Select **Write File**

A SDR33 file is created ready to send to the instrument directly using the upload facility or by copying the file to PCMCIA cards etc.

Loading file into controller using 12d upload facility

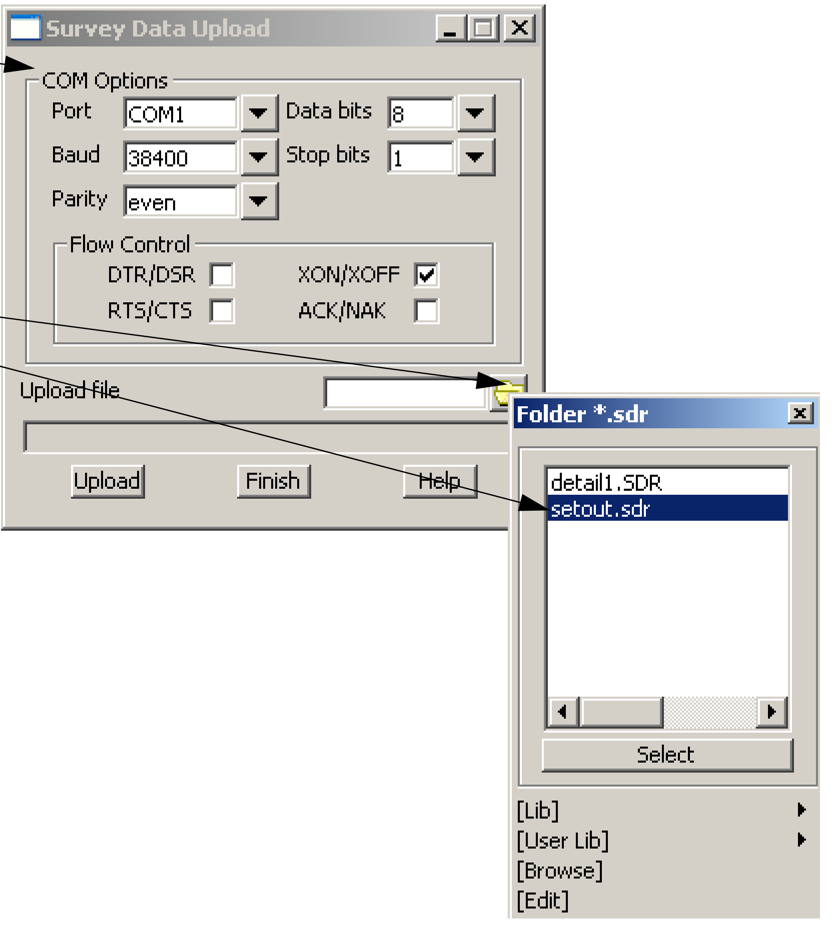
The points upload file can be transferred to the controller for setout using the upload panel.

Connect the controller to the PC

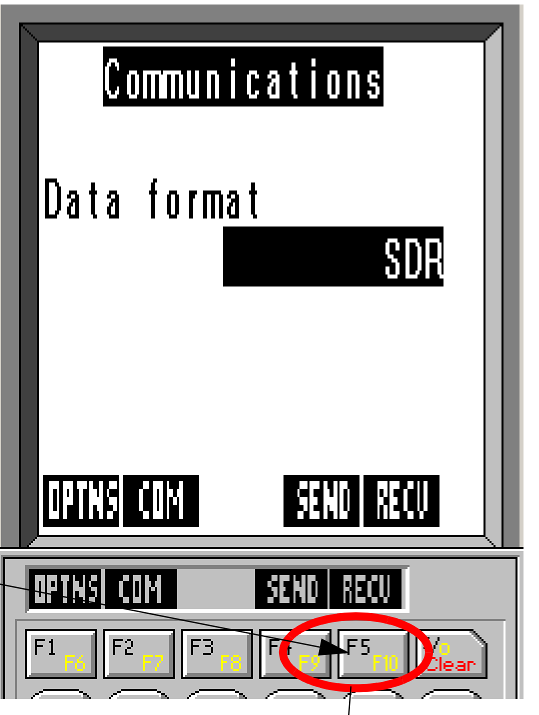
Select option ***Survey=>Upload=>Upload***

Ensure that the correct values are set for the comms settings i.e. they match the data collector parameters

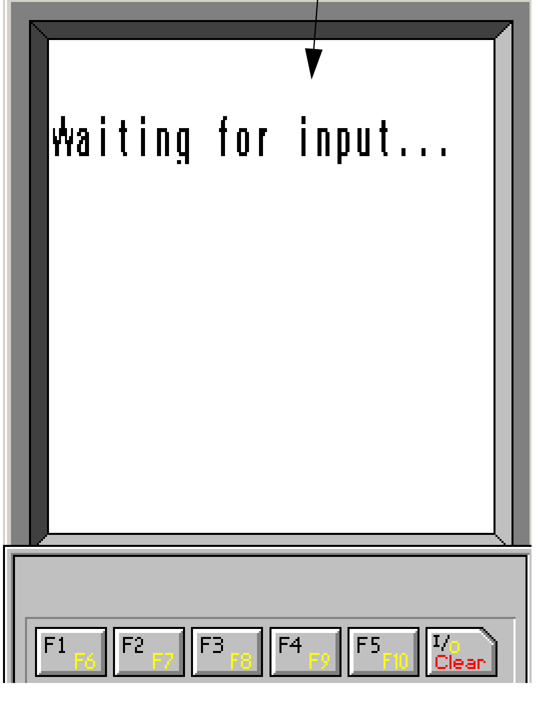
Select the upload file by selecting the folder icon and selecting the appropriate file



Prepare the data collector for uploading.

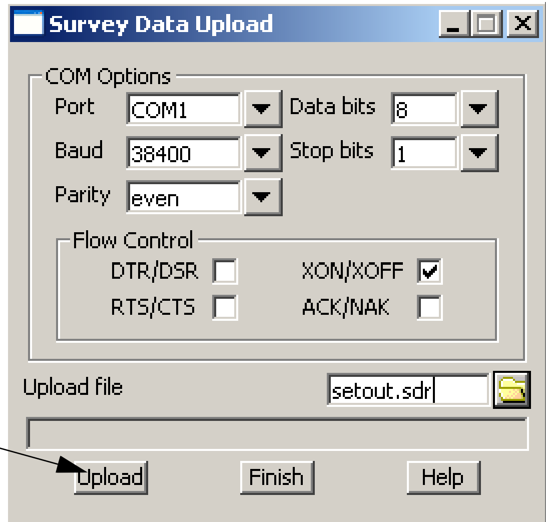


From the communication menu, select the receive button



Once the data collector is ready, the data can be sent from 12d

Select **upload** to start uploading the file



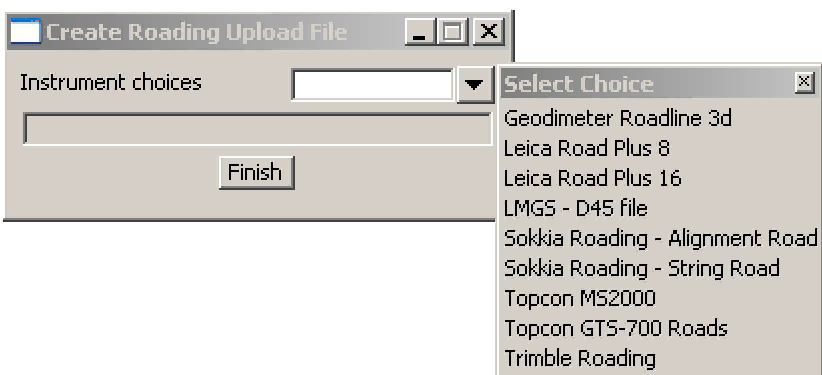
Please continue to the next section 37.2.1.1.3 Creating SDR Point Upload File

37.2.1.1.4 Creating SDR Roads Upload File

Create upload file in 12d

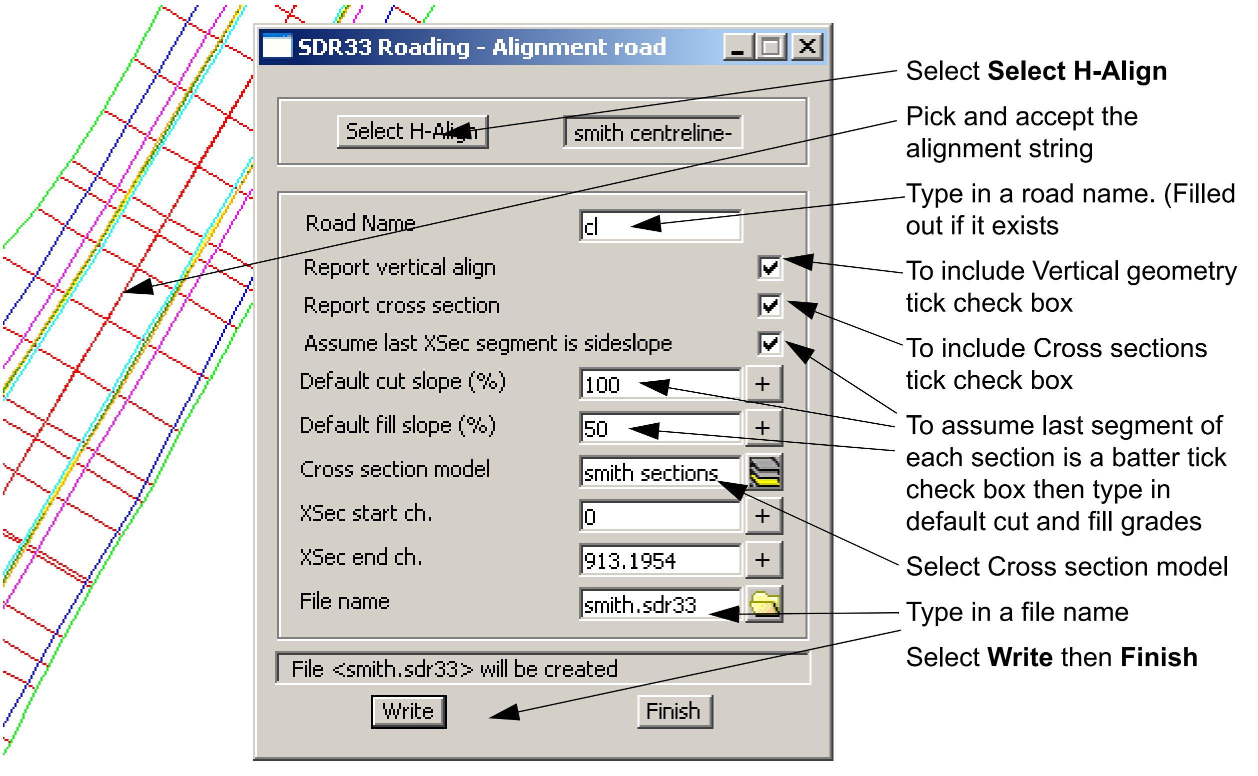
After creating the road alignment an upload file can be created for the SDR33 format

Select option ***Survey=>Upload=>Create roads upload file***



Select **Instrument** choice icon

Select **Sokkia Roading - Alignment Road**



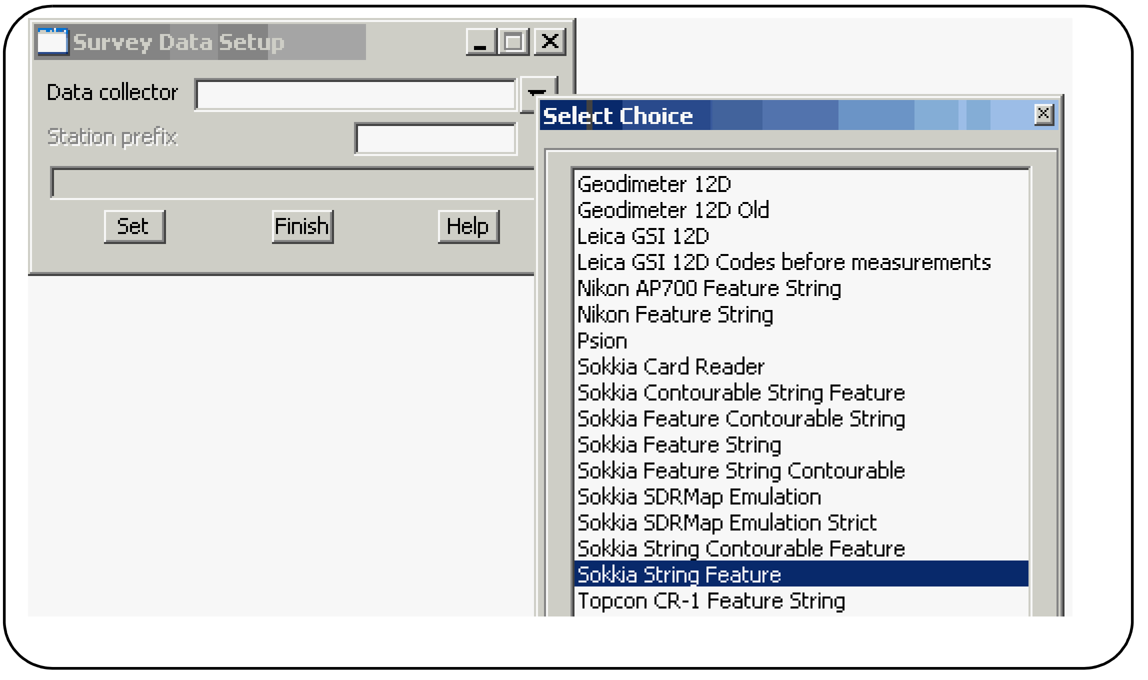
A SDR33 file is created ready to send to the instrument directly using the upload facility or by copying the file to PCMCIA cards etc. The steps involved in uploading the file to the controller using the 12d upload facility can be seen is the previous section Loading file into controller using 12d upload facility.

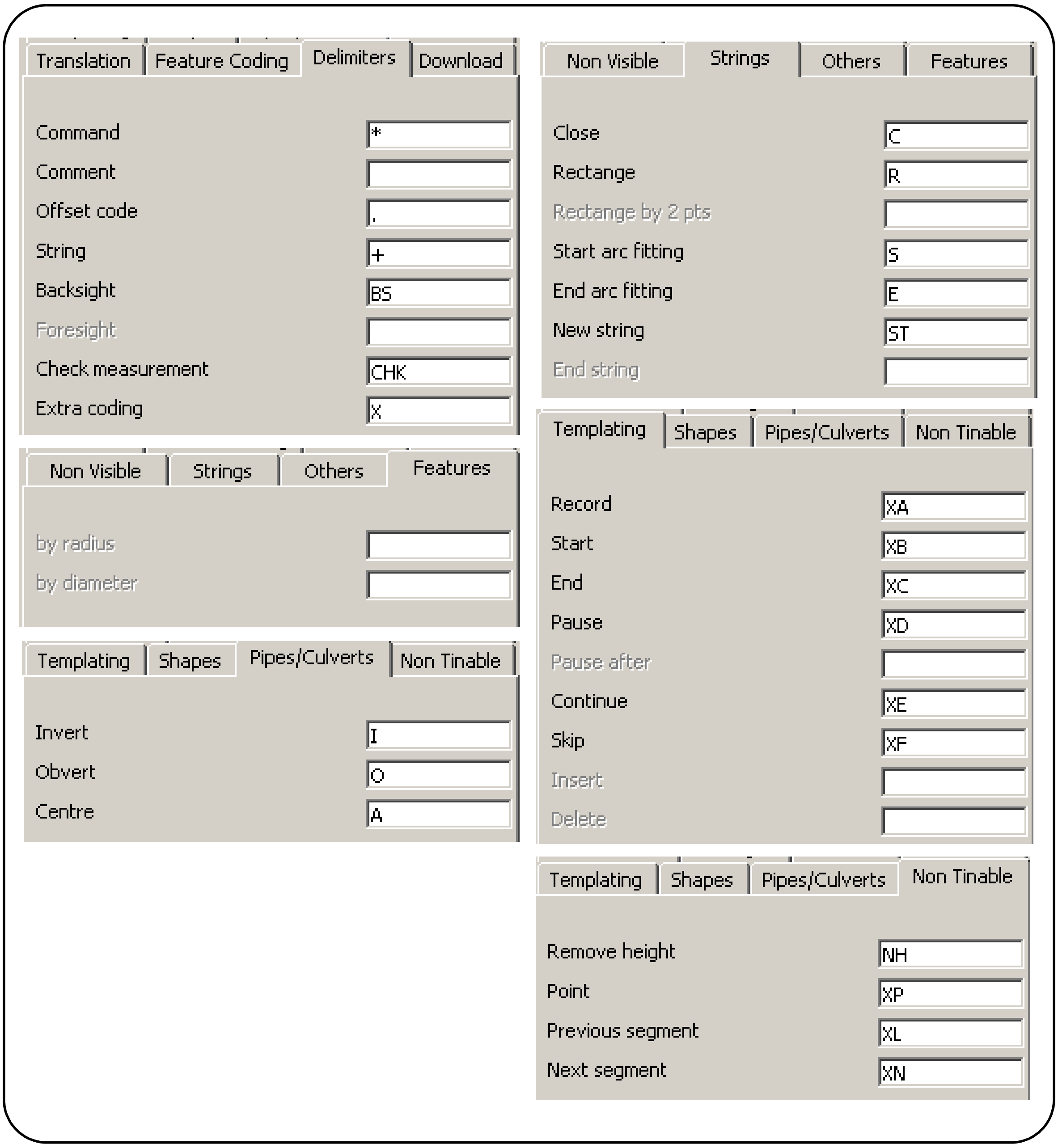
Please continue to the next section 37.2.1.2 Example of Sokkia SDR File

37.2.1.2 Example of Sokkia SDR File

The raw Sokkia SDR file 'detail1.sdr' has been coded in accordance with the data collector definition *Sokkia String Feature* which is shipped with ***12d Model***.

The data collector *Sokkia String Feature* is set using the option *Survey=>Setup*


Survey Data Setup
Data collector
Station prefix
Set
Finish
Help
Select Choice
Geodimeter 12D
Geodimeter 12D Old
Leica GSI 12D
Leica GSI 12D Codes before measurements
Nikon AP700 Feature String
Nikon Feature String
Psion
Sokkia Card Reader
Sokkia Contourable String Feature
Sokkia Feature Contourable String
Sokkia Feature String
Sokkia Feature String Contourable
Sokkia SDRMap Emulation
Sokkia SDRMap Emulation Strict
Sokkia String Contourable Feature
Sokkia String Feature
Topcon CR-1 Feature String



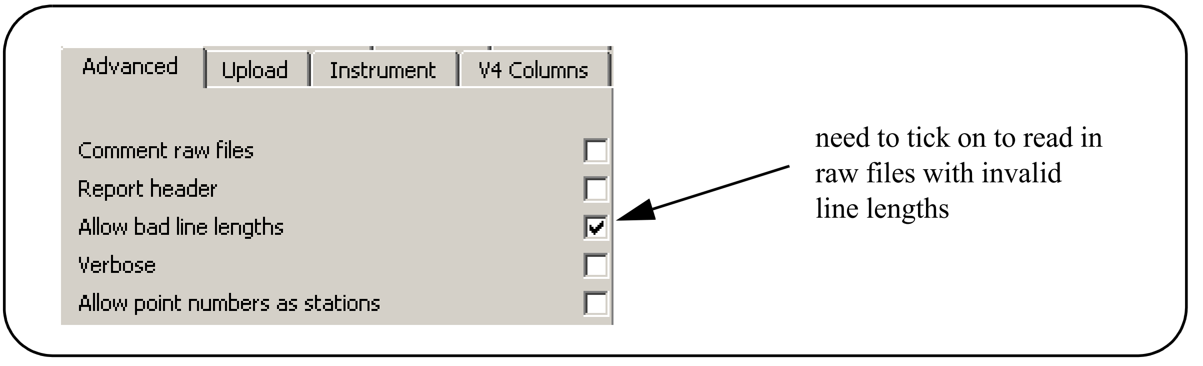
For the *Sokkia String Feature* data collector definition, the *string number* is given before the *feature code.*

Some of the control codes defined for the *Sokkia String Feature* data collector that are used in the example are:

***** is used as the Command (Block) delimiter
space is used as the comment delimiter
S and **E** are used to start arc fitting and end arc fitting respectively
R is the Rectangle (make a parallelogram) command
the template commands **XA**, **XB**, **XC** and **XD** are used
the invert **I** and obvert **O** commands are used.
XN for the next segment to be non-tinable (that is, not a breakline).

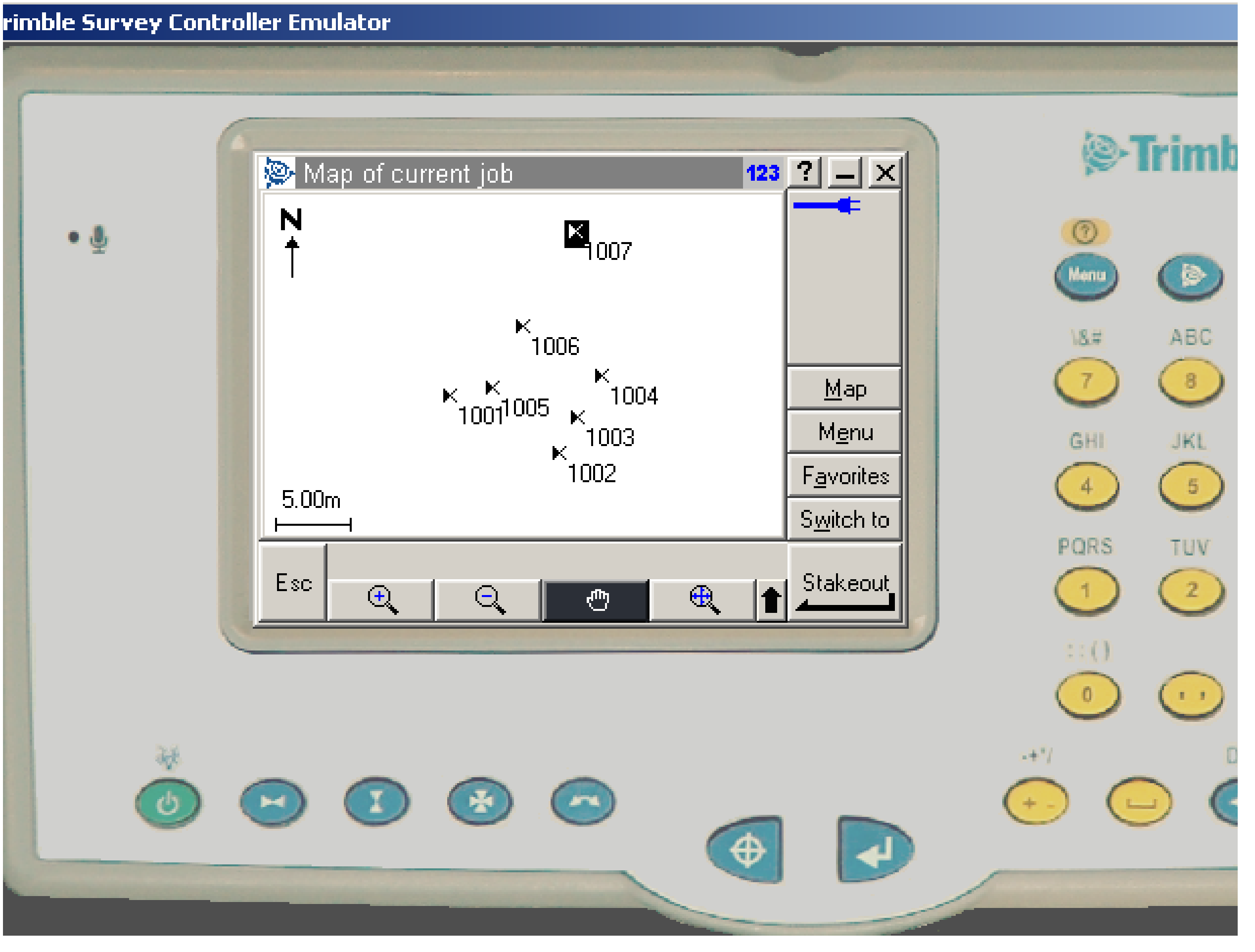
The listing of the raw file 'Detail1.sdr' is now given, followed by a dump of a ***12d Model*** view displaying the job. No Map File has been used in the reduction so no line styles appear on the view.

WARNING - The raw file 'Detail1.sdr' is in the Survey area of the training data but the file has been manually edited and if read in with the standard *Sokkia String Feature* data collector, will have bad line lengths and the data ignored. To read the file in, the *Allow bad line lengths* on the *Advanced* tab of the *Sokkia String Feature* data collector will need to be set before converting the raw file to a 12d field file.



37.3 12d and the Trimble Total Station ACU

This Section deals with interfacing 12d with the Trimble Total Station ACU unit data formats



These notes are not meant to take the place of the operations manuals for the Trimble ACU but rather to act as a guide for ensuring the correct file types are used between the two systems.

See the sections:

37.3.1 Coding

37.3.2 Sending Raw file to 12d

37.3.3 Creating and Setting Data Collector Type

37.3.4 Convert Raw File

37.3.5 Creating Point Upload File

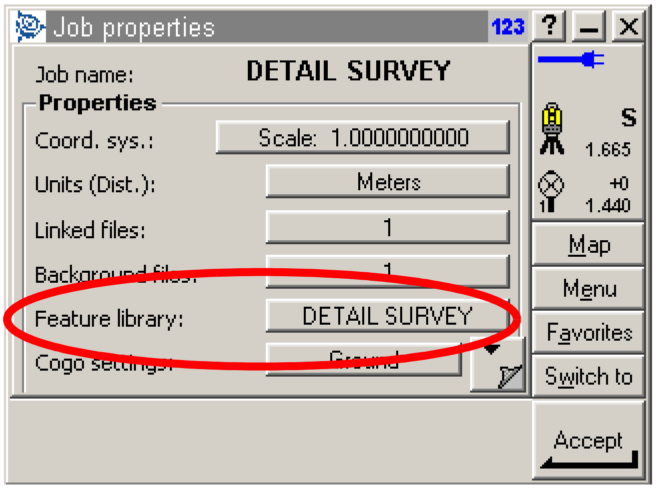
37.3.6 Create Roads Upload File

37.3.7 Create Tin Upload File

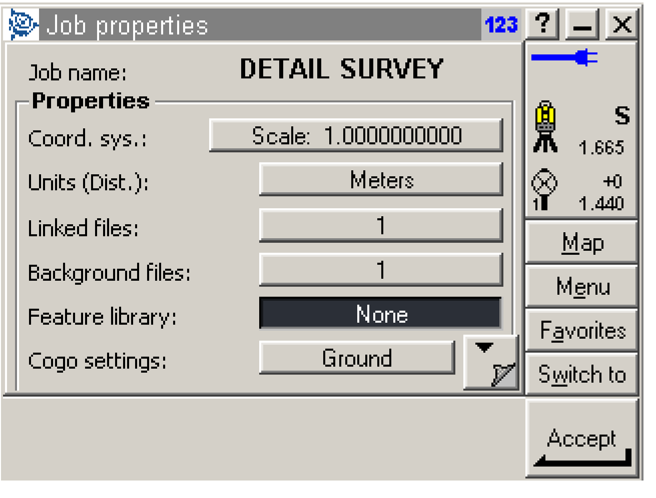
37.3.1 Coding

On the ACU the user can either manually type in a code for field readings or use a code library to pre-define codes which can then be selected from a list

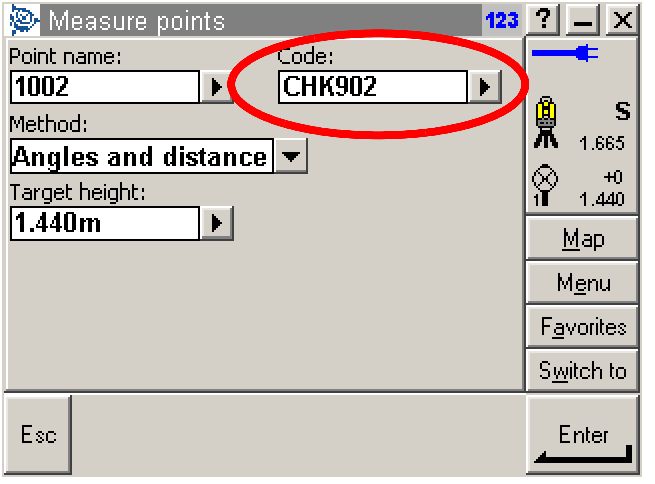
To use a code library the user should select the relevant library when creating the job

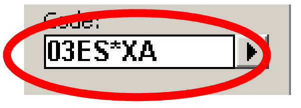


If using string numbers the Feature library can be set to ***None*** and the string number, feature code, field code and delimiters can be manually typed in



Codes are typed in to the ACU in the **MEASURE POINTS** option



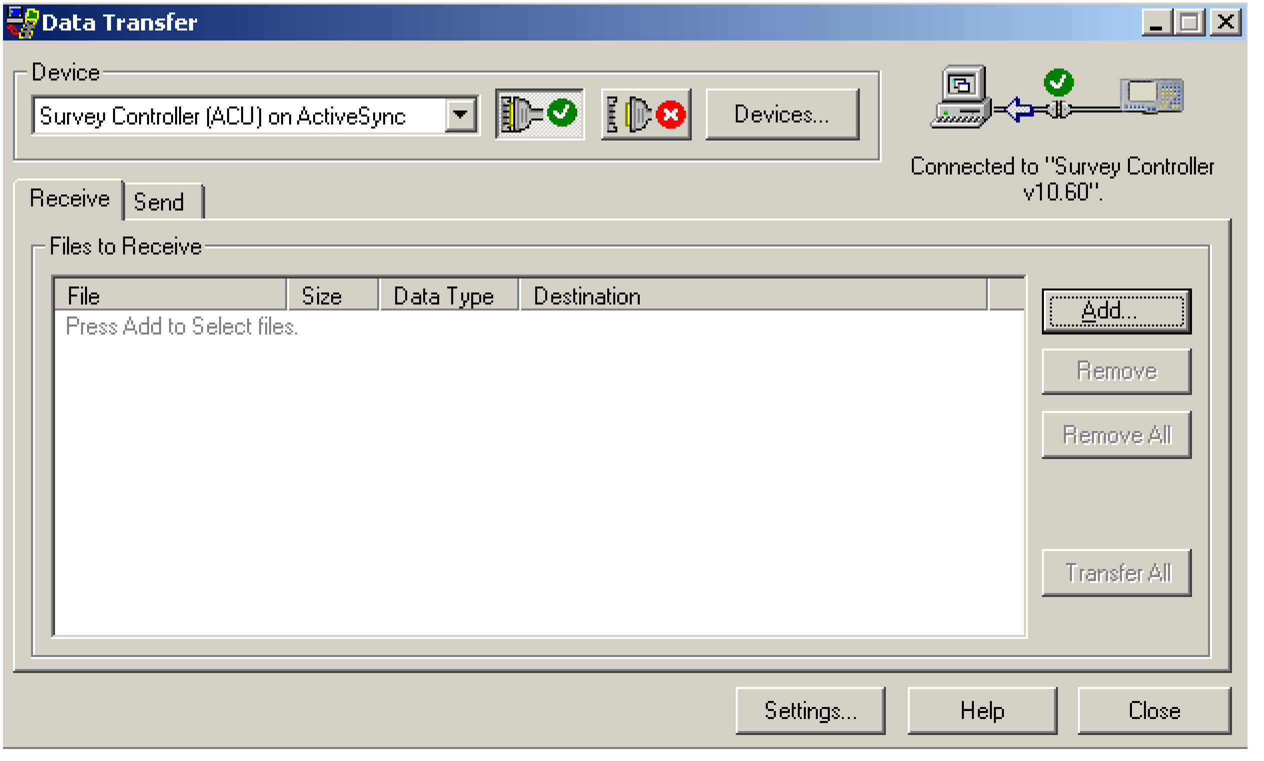


Please continue to the next section 37.3.2 Sending Raw file to 12d

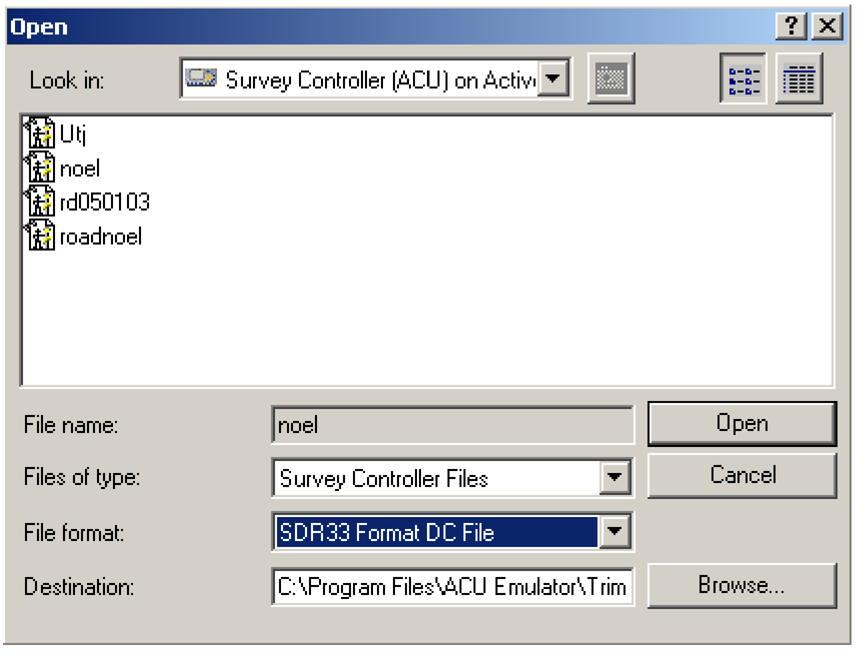
37.3.2 Sending Raw file to 12d

The raw file can be transferred from the ACU as a format similar to the SDR33. This is called the **SDR33 DC** file format.

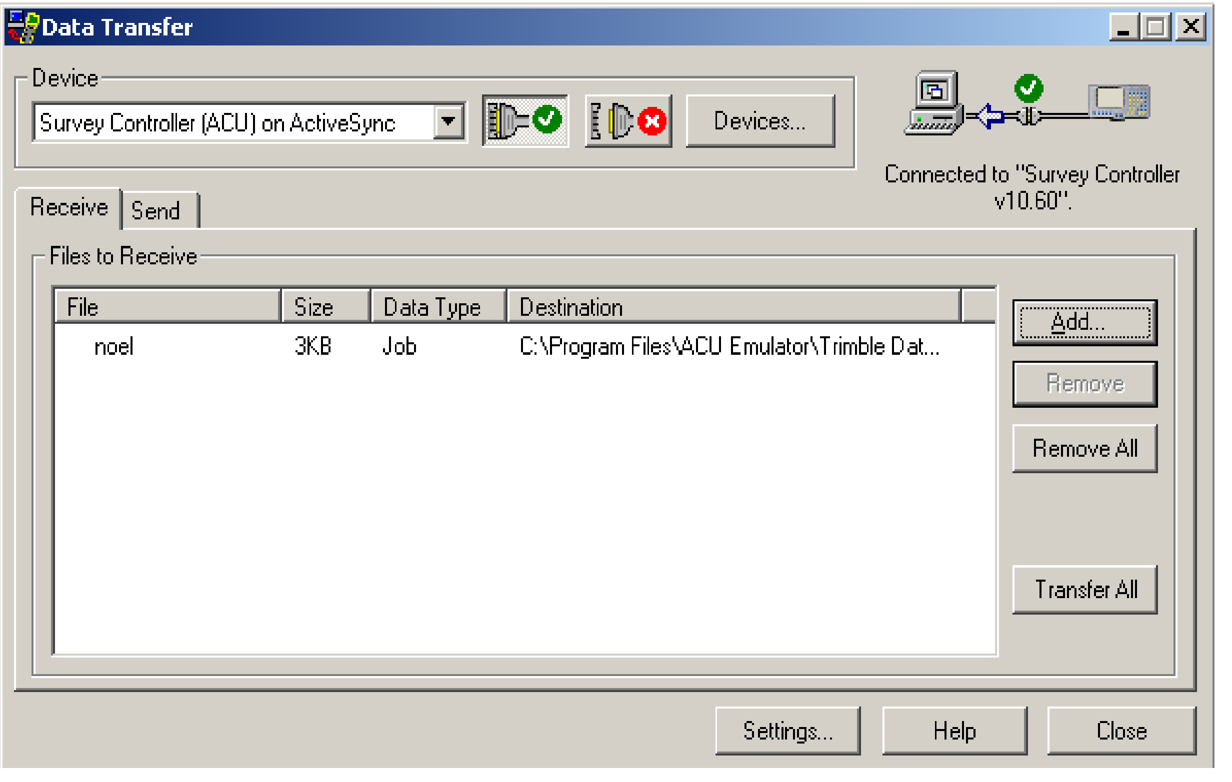
Connect the ACU to the PC using **TRIMBLE DATA TRANSFER**



Select ***Receive*** tab then select ***Add***



Select the Job to download using the format ***SDR33 FORMAT DC FILE***



Select ***Transfer All*** to transfer the file

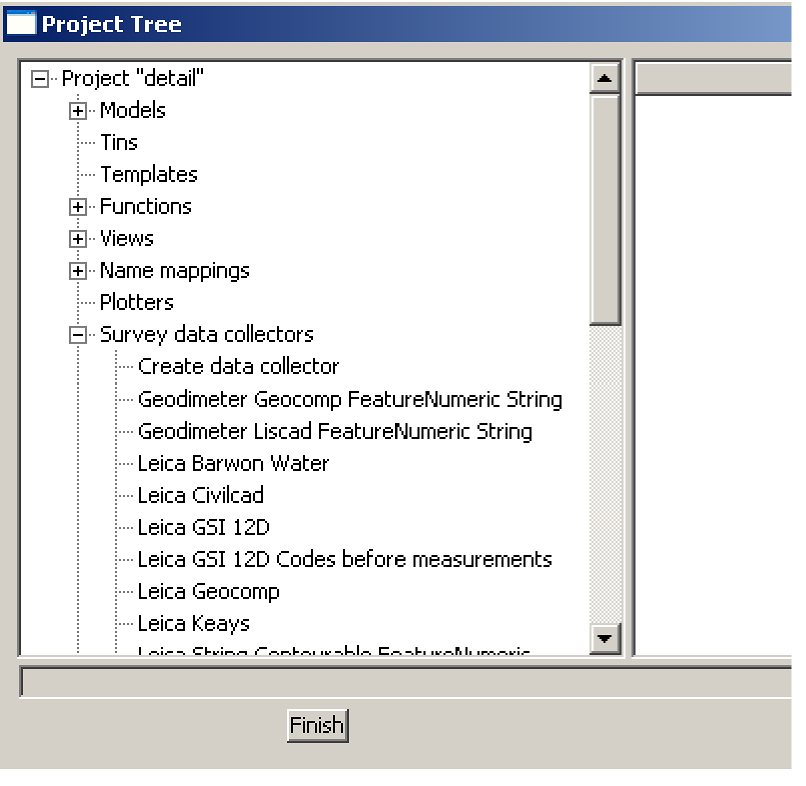
Please continue to the next section 37.3.3 Creating and Setting Data Collector Type

37.3.3 Creating and Setting Data Collector Type

Create new data collector type

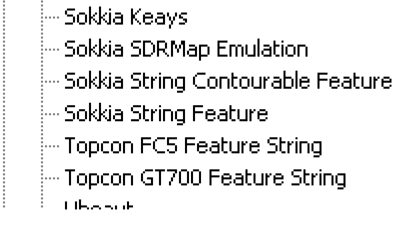
We will create a specific name for the Trimble data collector even though it is similar to the SDR33 format

Select ***Project=>Browse***



Select the Survey data collectors branch

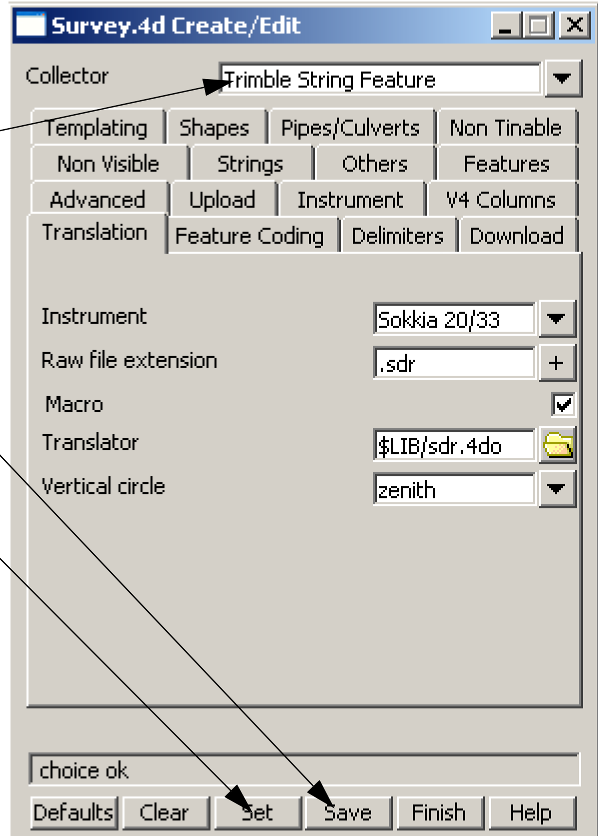
Double click on **Sokkia String Feature**



This loads up the default parameters for the SDR33 format.

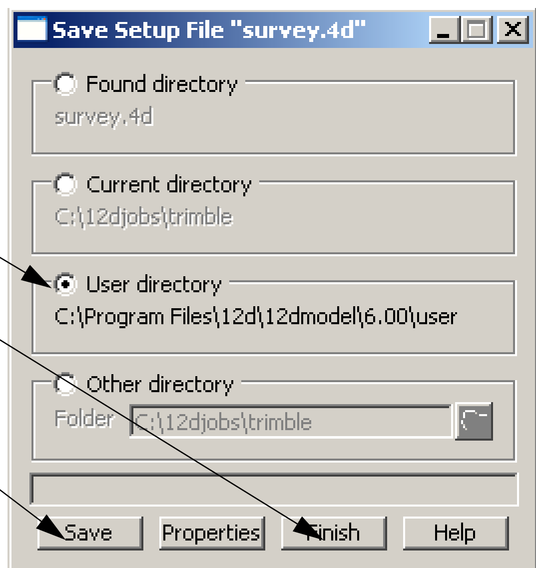
Type in the new name **Trimble String Feature**

Select **Set** then **Save**



Select **User directory**

Select **Save** then **Finish**

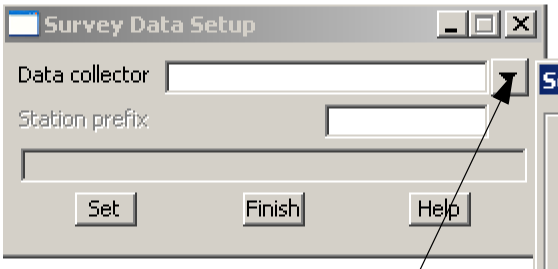


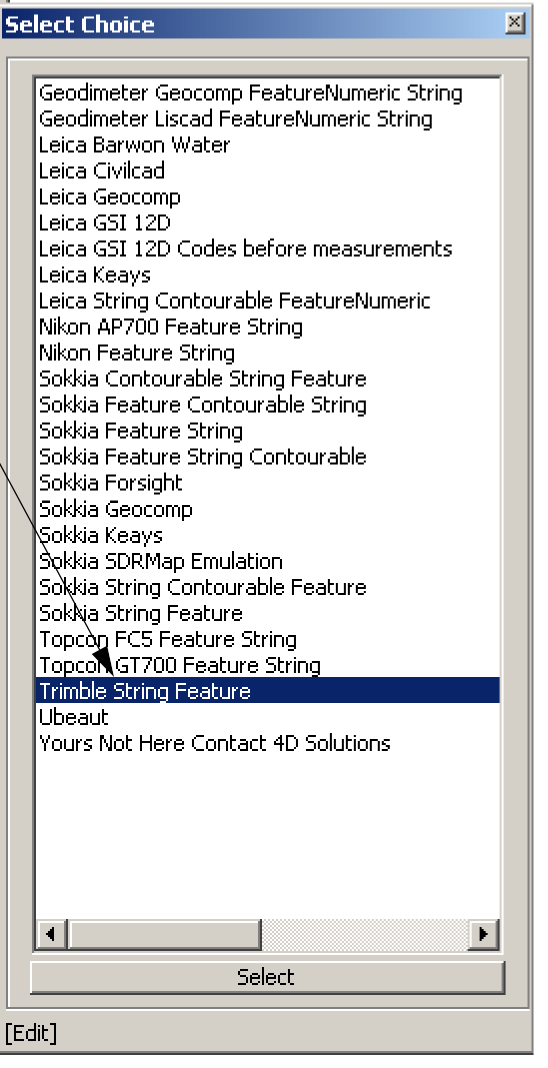
Select **Finish** back in the previous panel

To set the changes the project has to be restarted by selecting ***Project=>Restart***

Select the data collector type

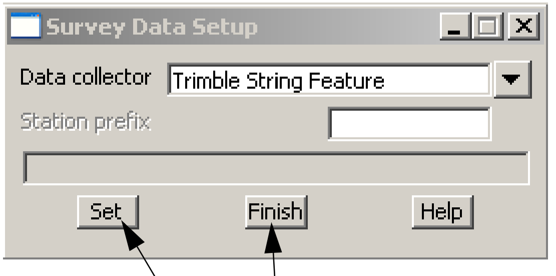
Select option ***Survey=>Setup***





Select the **Data collector** choice icon

Double click on **Trimble String Feature**

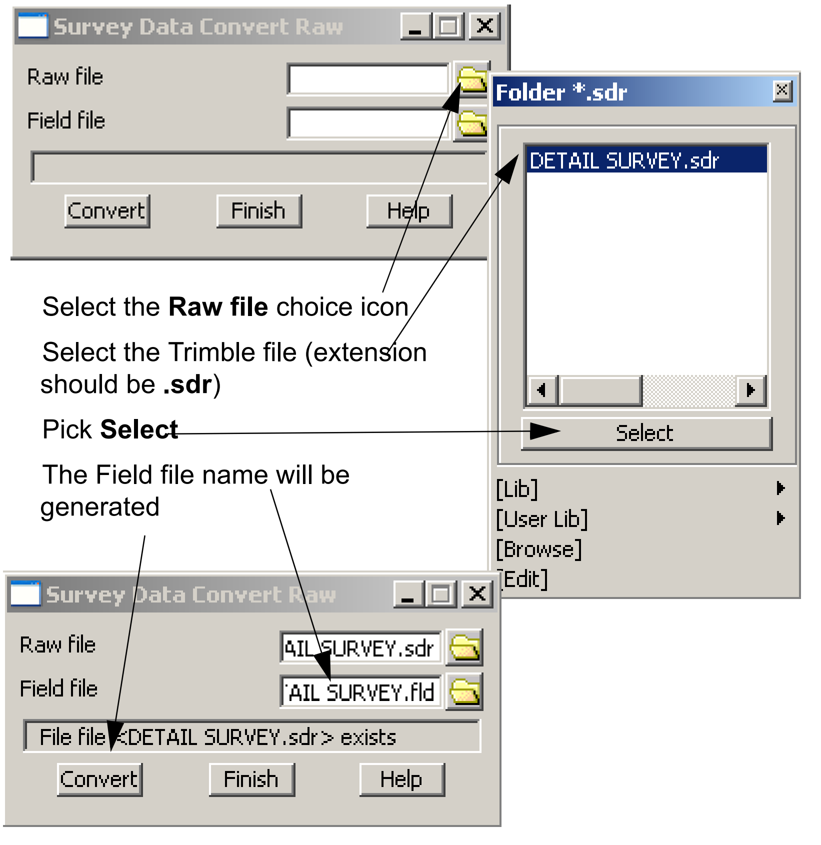


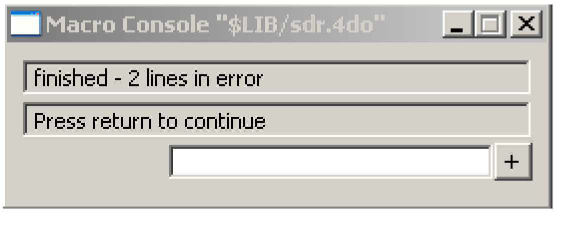
Select **Set** then **Finish**

Please continue to the next section 37.3.4 Convert Raw File

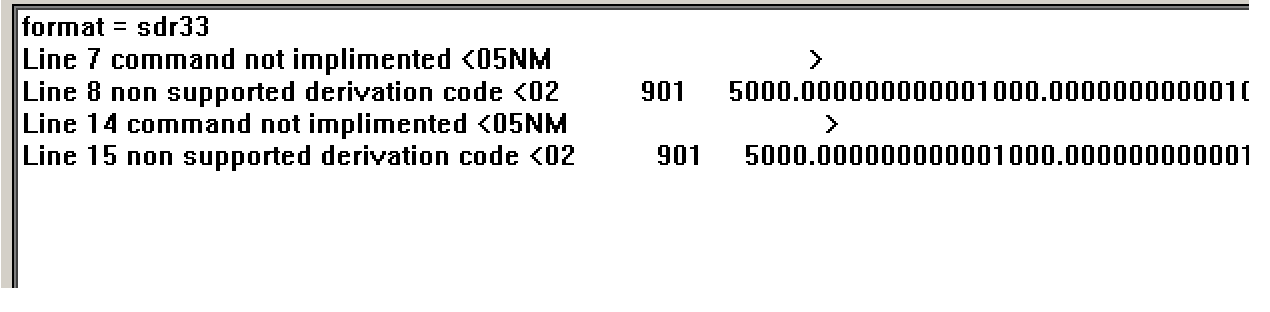
37.3.4 Convert Raw File

To convert raw file to field file select option ***Survey=>Convert raw***

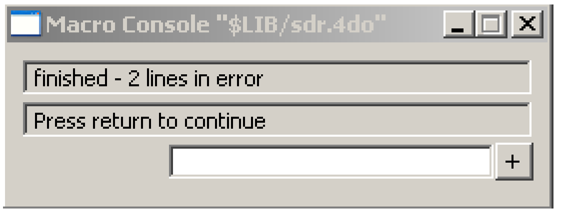




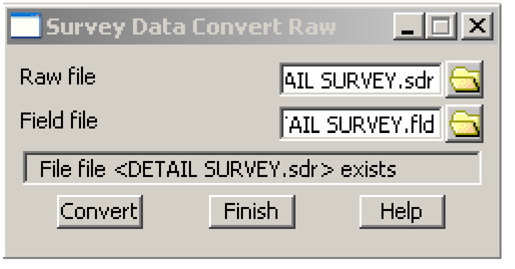
Any errors can be listed by opening the Output window using option ***Window=>Output Window***



format = sdr33
Line 7 command not implimented <05NM >
Line 8 non supported derivation code <02 901 5000.000000000001000.00000000001(
Line 14 command not implimented <05NM >
Line 15 non supported derivation code <02 901 5000.000000000001000.000000000001



Close the Error panel



Select **Finish**

The field file can now be reduced in the normal manner

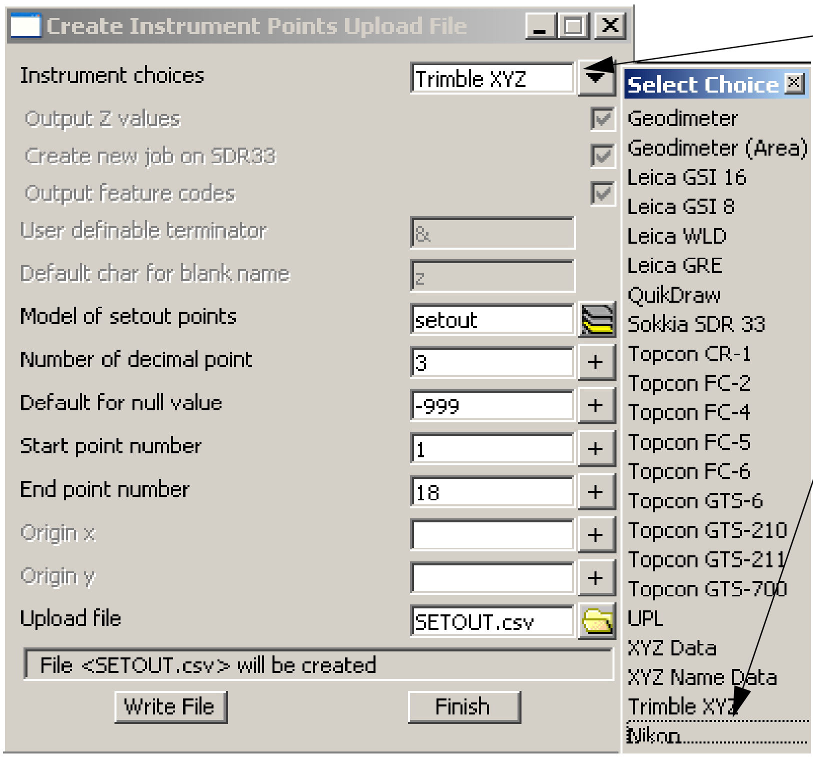
Please continue to the next section 37.3.5 Creating Point Upload File

37.3.5 Creating Point Upload File

Create upload file in 12d

After creating the setout points an upload file has to be created for the Trimble format

Select option ***Survey=>Upload=>Create points upload file***



Select **Instrument choices** icon

Select **Trimble XYZ**

Fill in the rest of the screen as per normal

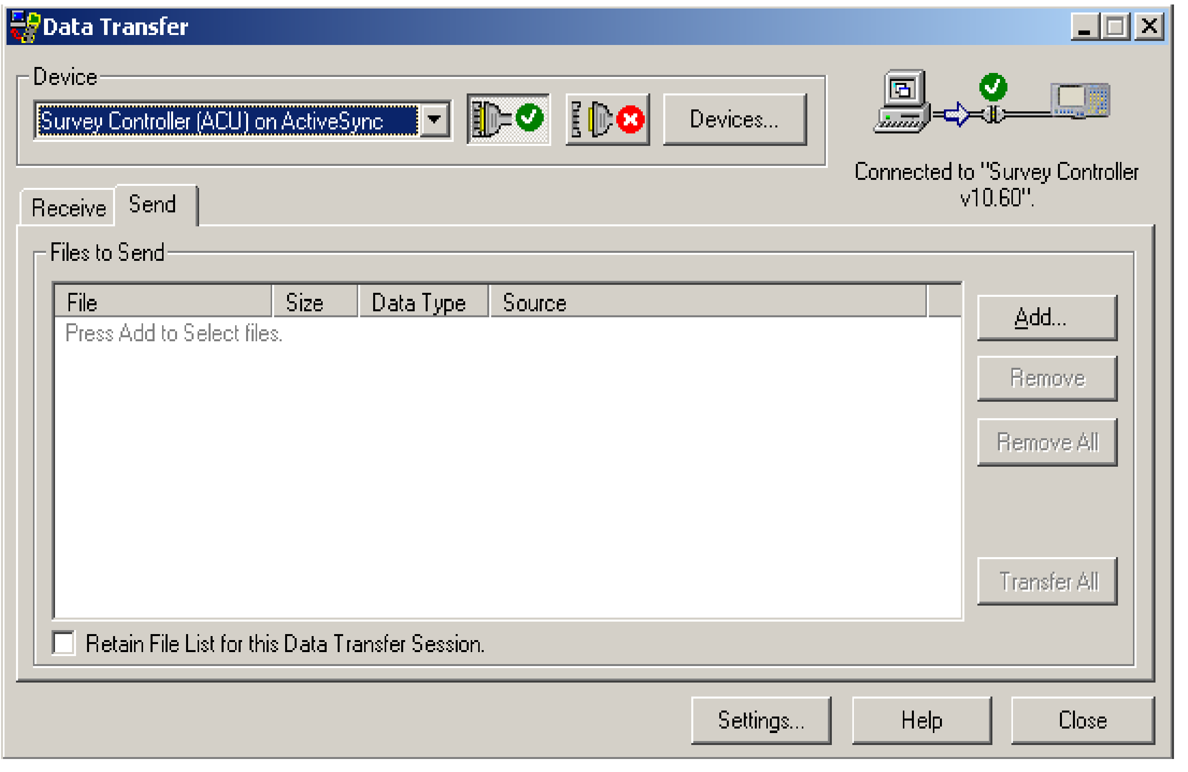
Select **Write File**

A standard comma delimited file is created ready to send to the ACU

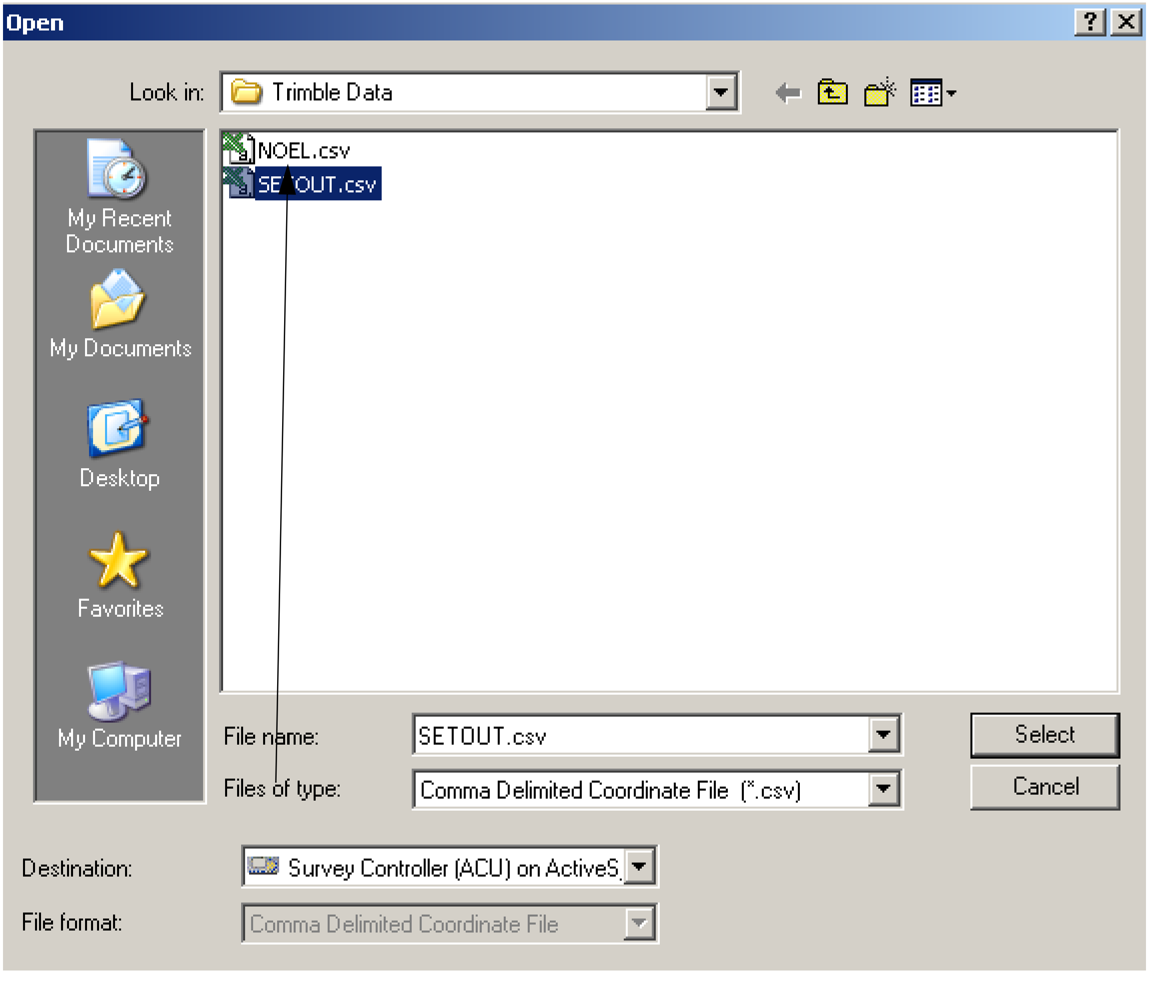
Loading file into job in ACU

The points upload file can be transferred to the ACU for setout.

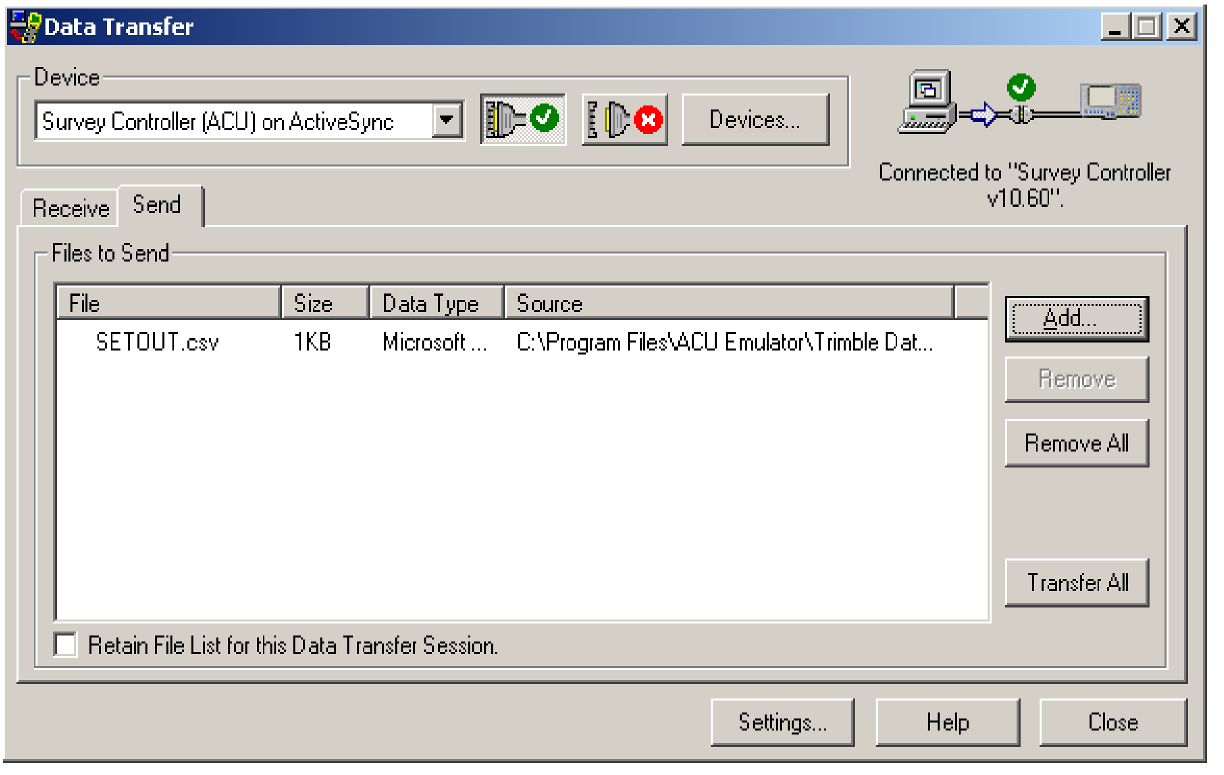
Connect the ACU to the PC using **TRIMBLE DATA TRANSFER**



Select ***Send*** tab then select ***Add***

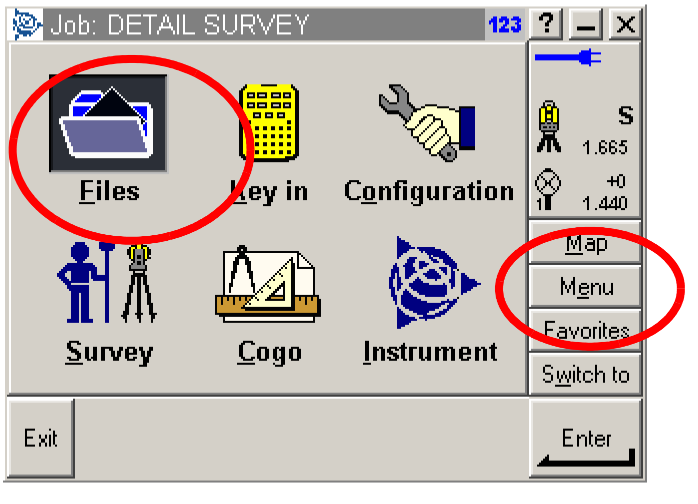


Select the file to upload using format ***Comma delimited coordinate file***

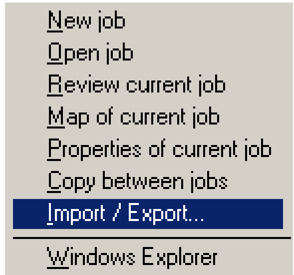


Select ***Transfer All*** to transfer the file

To load the file into the current job on the ACU select ***Menu=>Files***

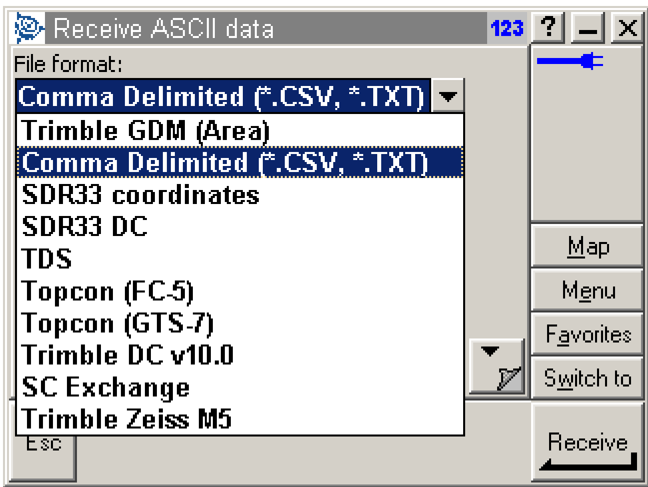


Select ***Import / Export***

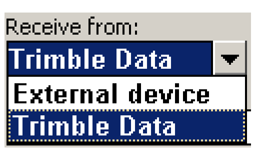


Select ***Receive ASCII data***

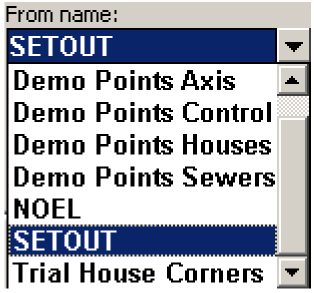
Select **File format** choice icon then pick ***Comma Delimited (*.CSV, *.TXT)***



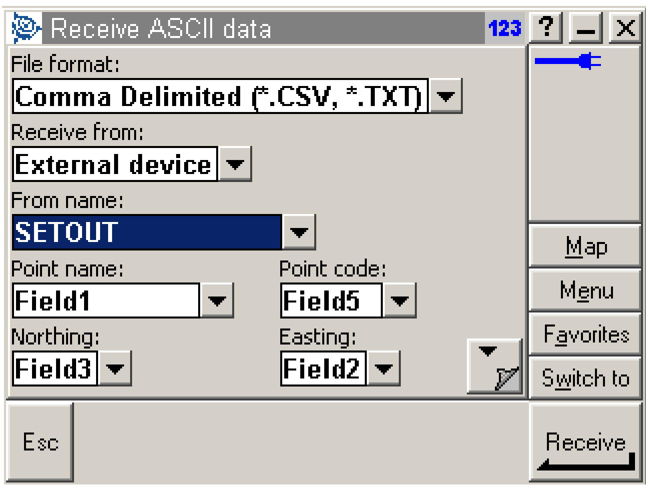
Select **Receive from** choice icon and pick ***Trimble Data***



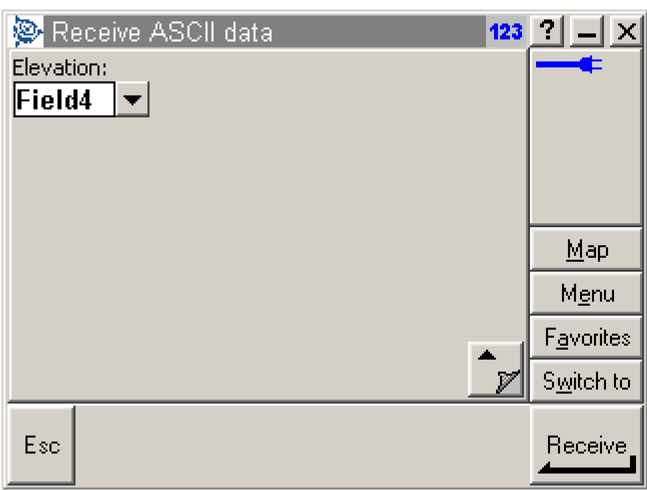
Select **From file** choice icon and select file



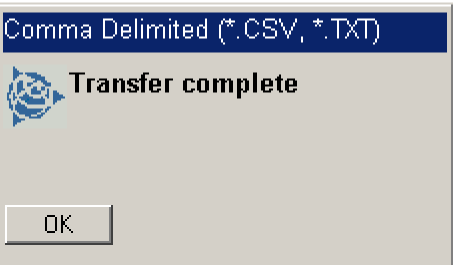
Select the field order for the file as (1) Point number, (2) Easting, (3) Northing, (4) Level and (5) Code



Select the **Next page** icon

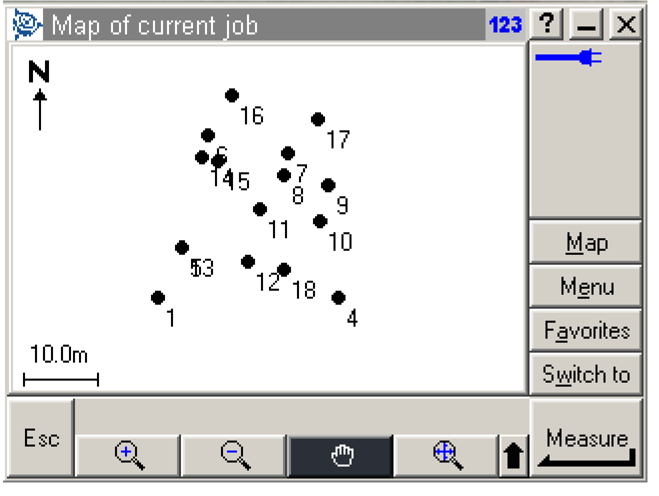


Select **Receive**



Select **OK** to finish

The points can be displayed by selecting **Map** and zooming all of the job



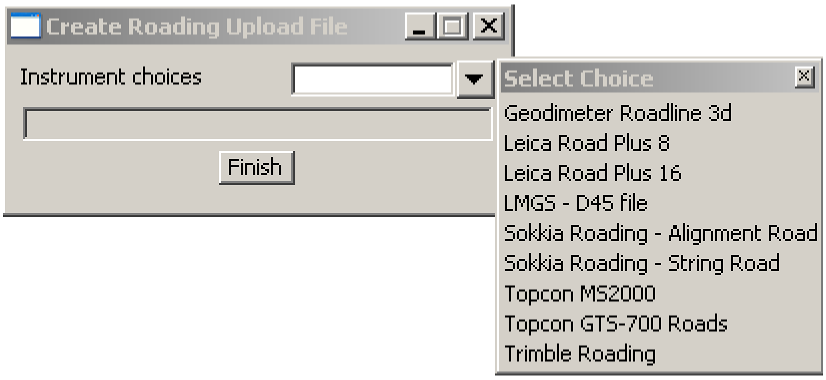
Please continue to the next section 37.3.6 Create Roads Upload File

37.3.6 Create Roads Upload File

Create upload file in 12d

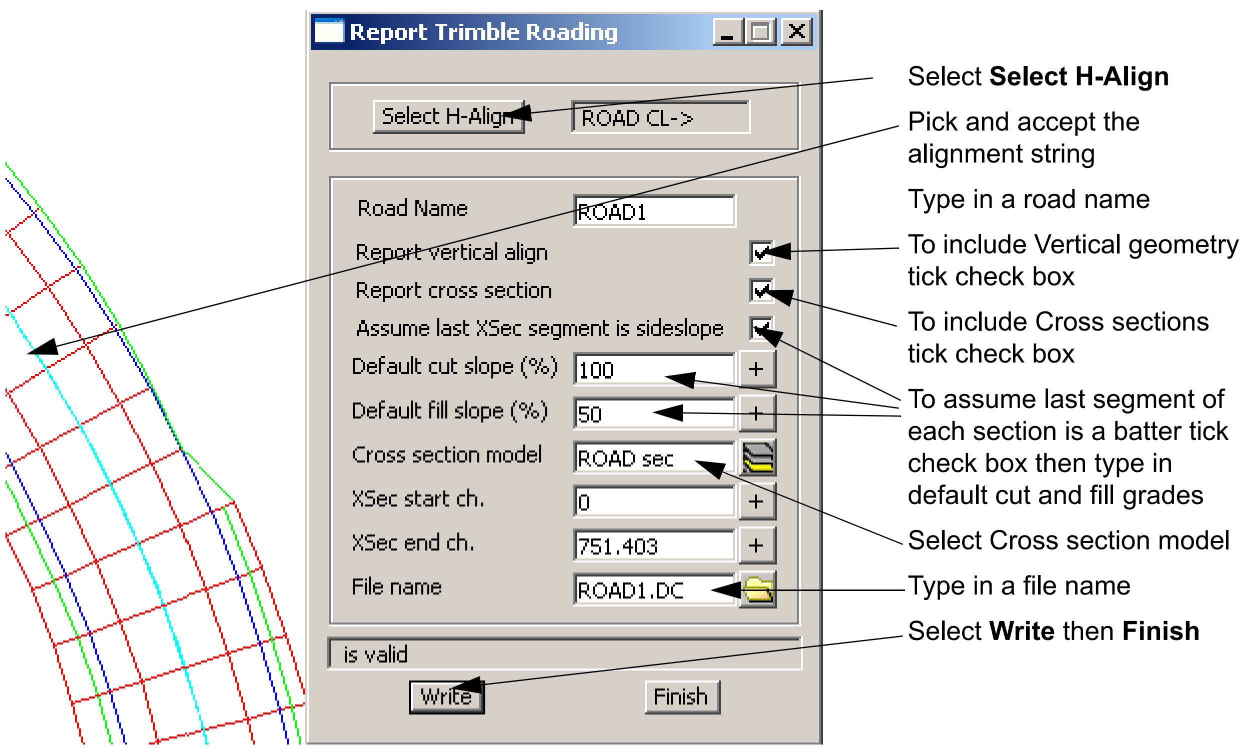
After creating the road alignment an upload file has to be created for the Trimble format

Select option ***Survey=>Upload=>Create roads upload file***



Select **Instrument** choice icon

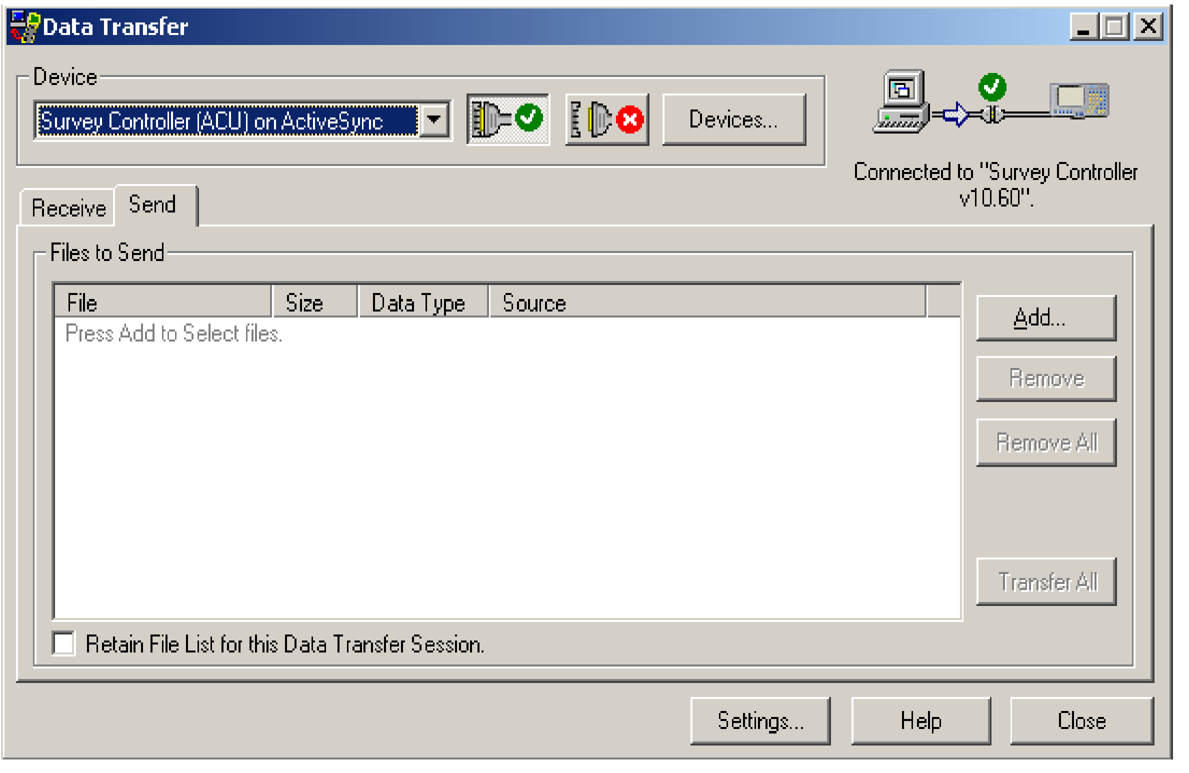
Select **Trimble Roading**



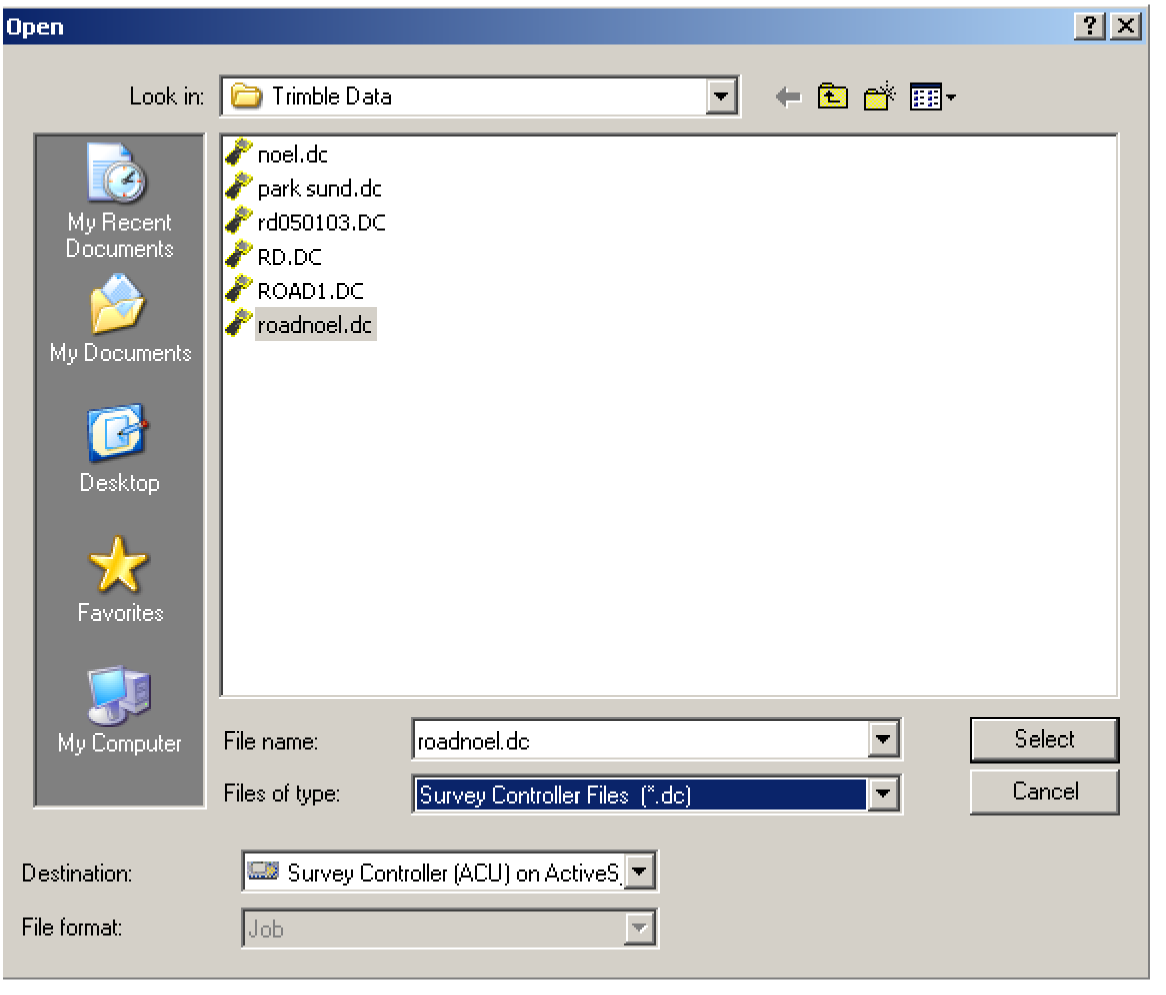
Loading file to the ACU

The road upload file can be transferred to the ACU for setout.

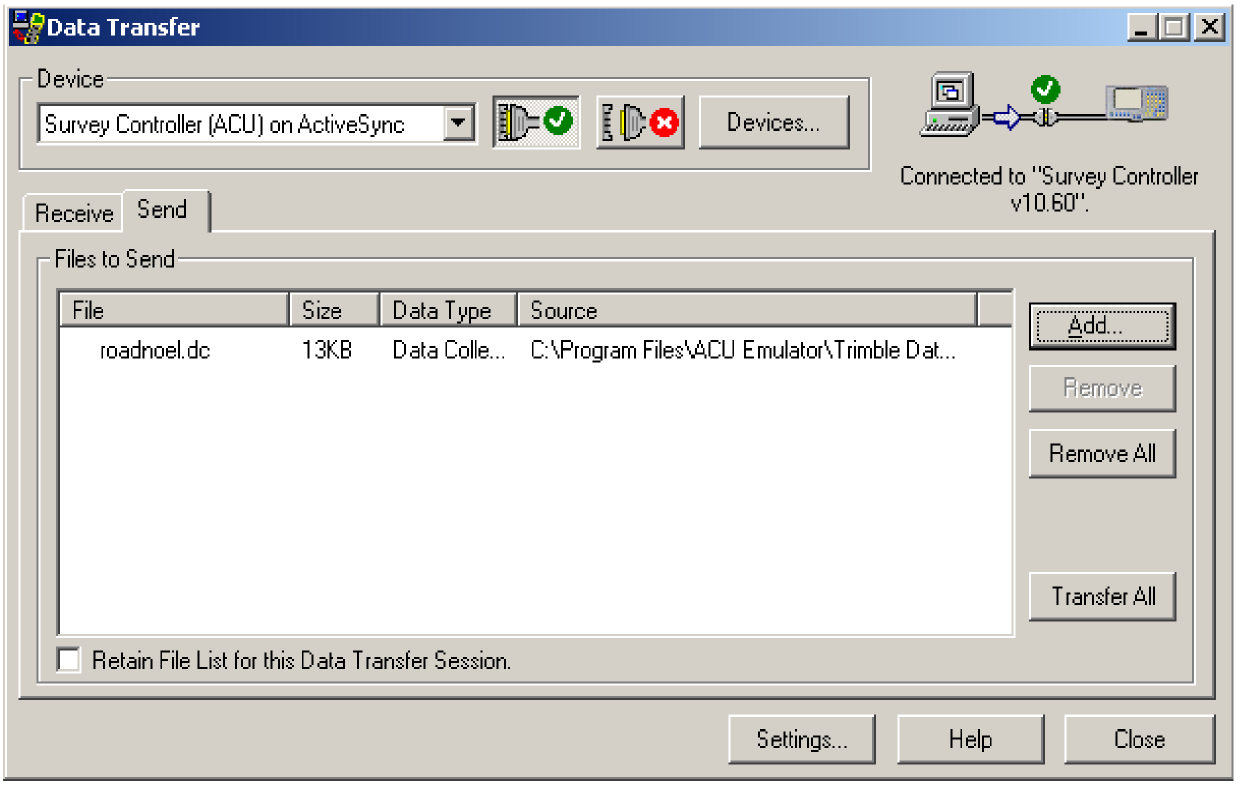
Connect the ACU to the PC using **TRIMBLE DATA TRANSFER**



Select ***Send*** tab then select ***Add***

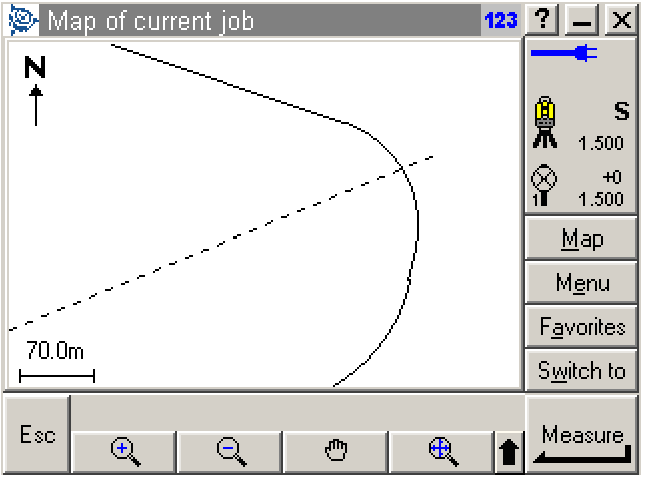


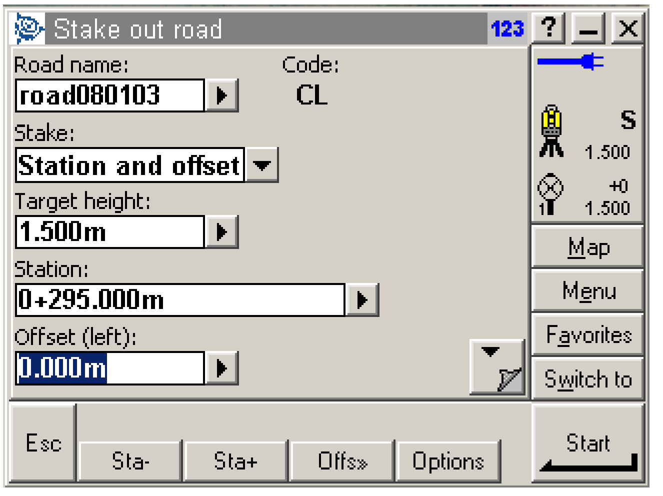
Select the road file to upload using format ***Survey Controller file***

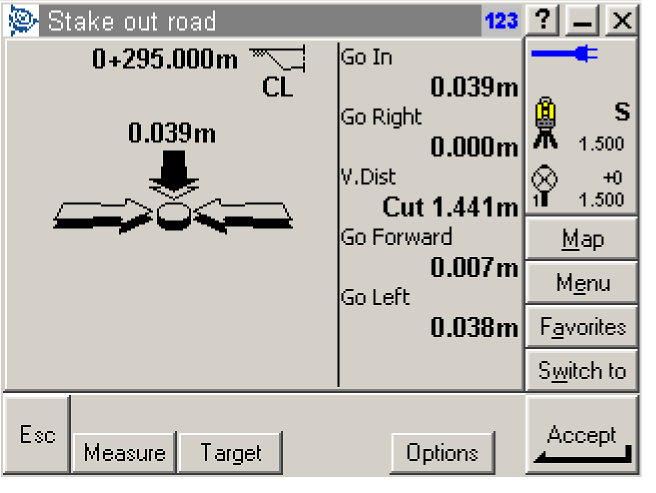


Select ***Transfer All*** to transfer the file

Once the road file has been copied into the ACU it can be setout using option ***Survey=>Stakeout=>Roads***







Please continue to the next section 37.3.7 Create Tin Upload File

37.3.7 Create Tin Upload File

Create upload file in 12d

After creating the triangulation an upload file has to be created for the Trimble format

Select option ***Survey=>Upload=>Create triangle upload file***

Select the **File Type** choice icon and select **Trimble**

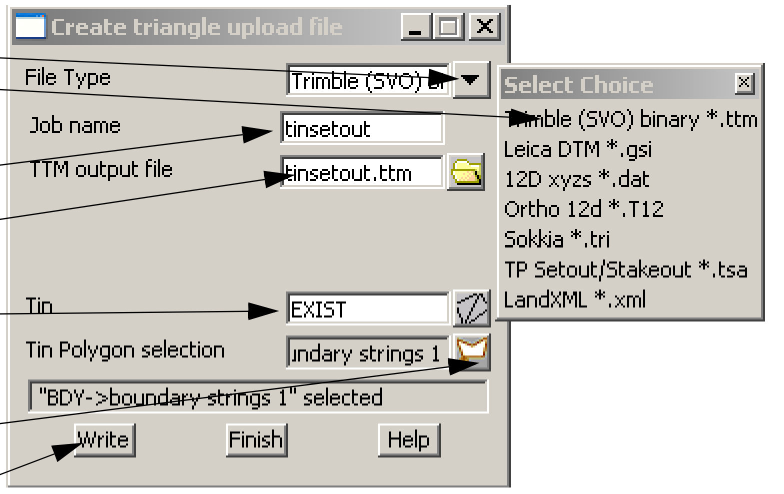
Type in Job name

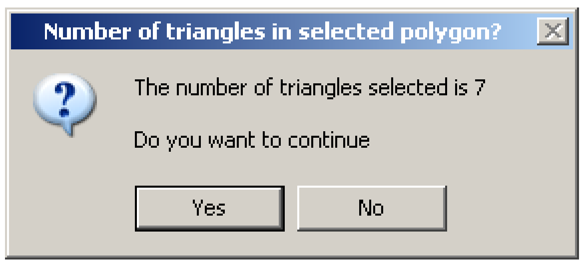
Type in output file name

Select the tin to list

Select the boundary string for the tin

Select **Write** then **Finish**



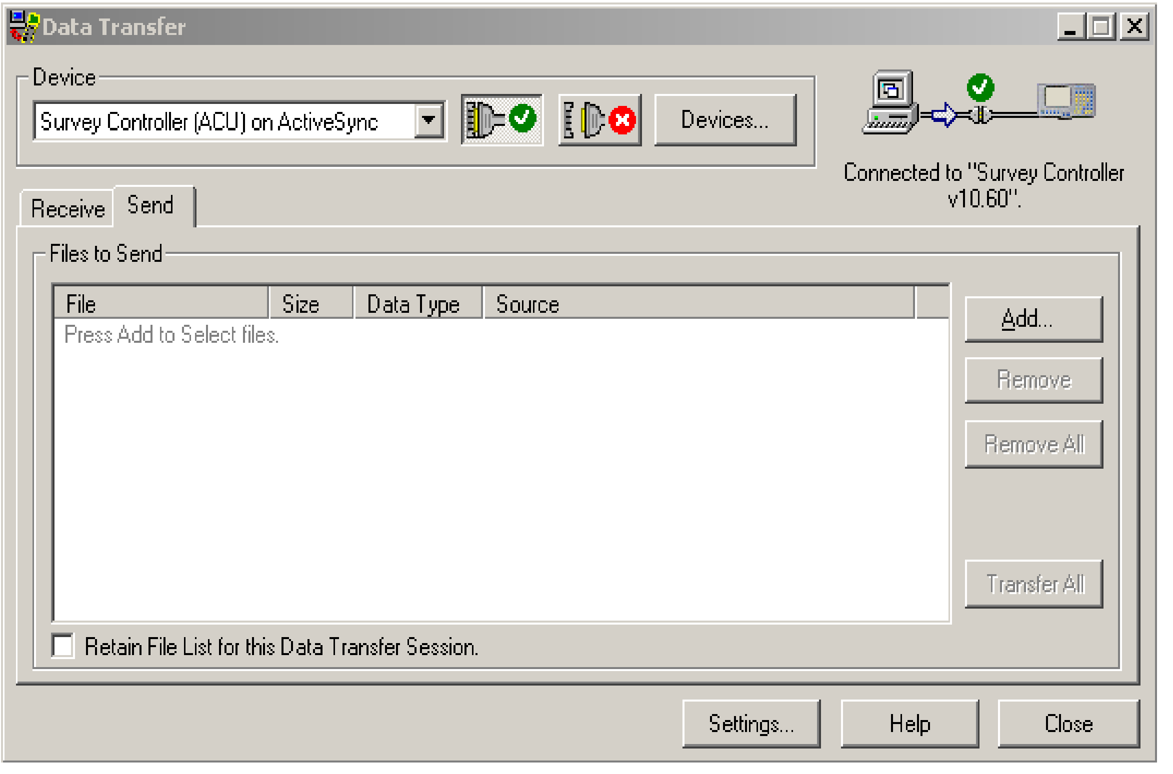


Select **Yes** to create the file

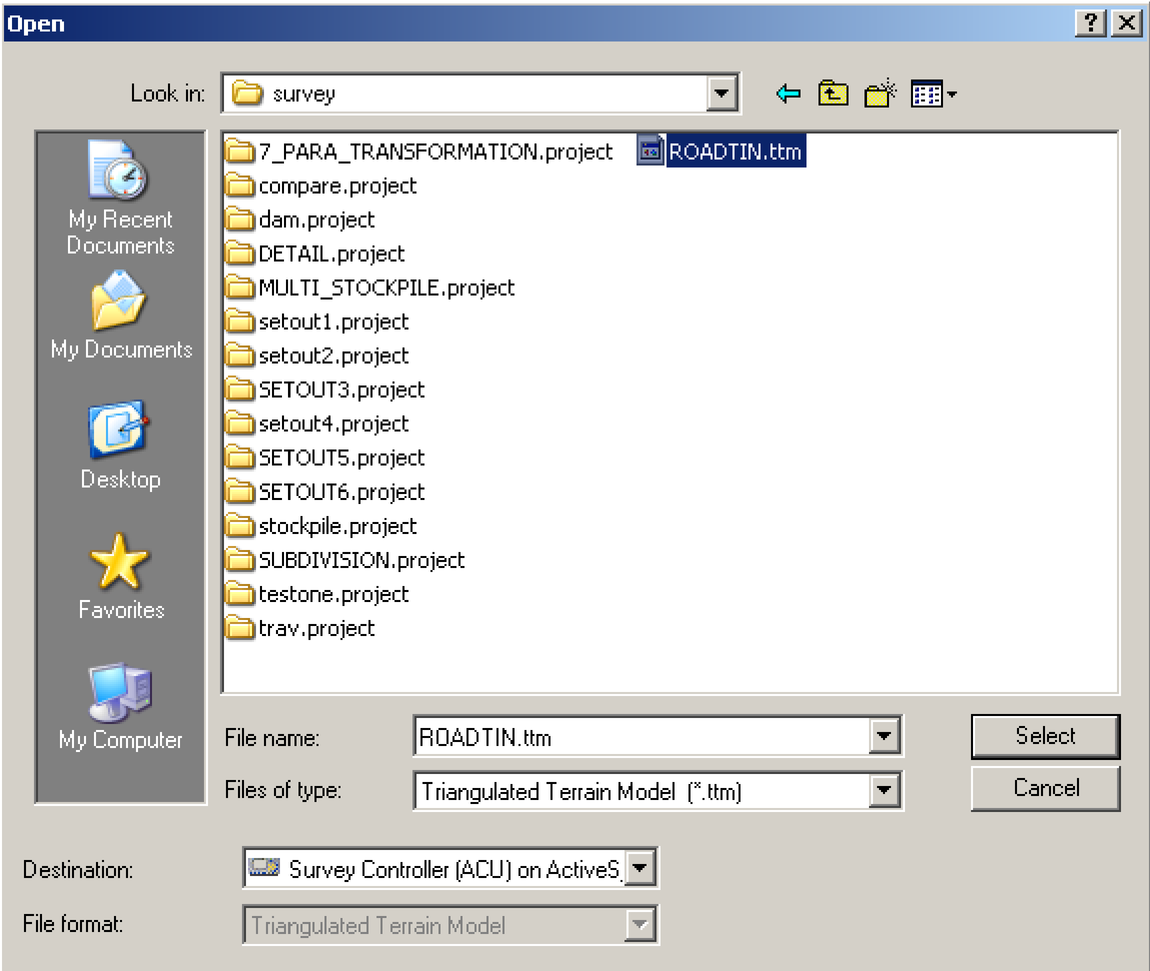
Loading and using the file in the ACU

The tin upload file can be transferred to the ACU for setout.

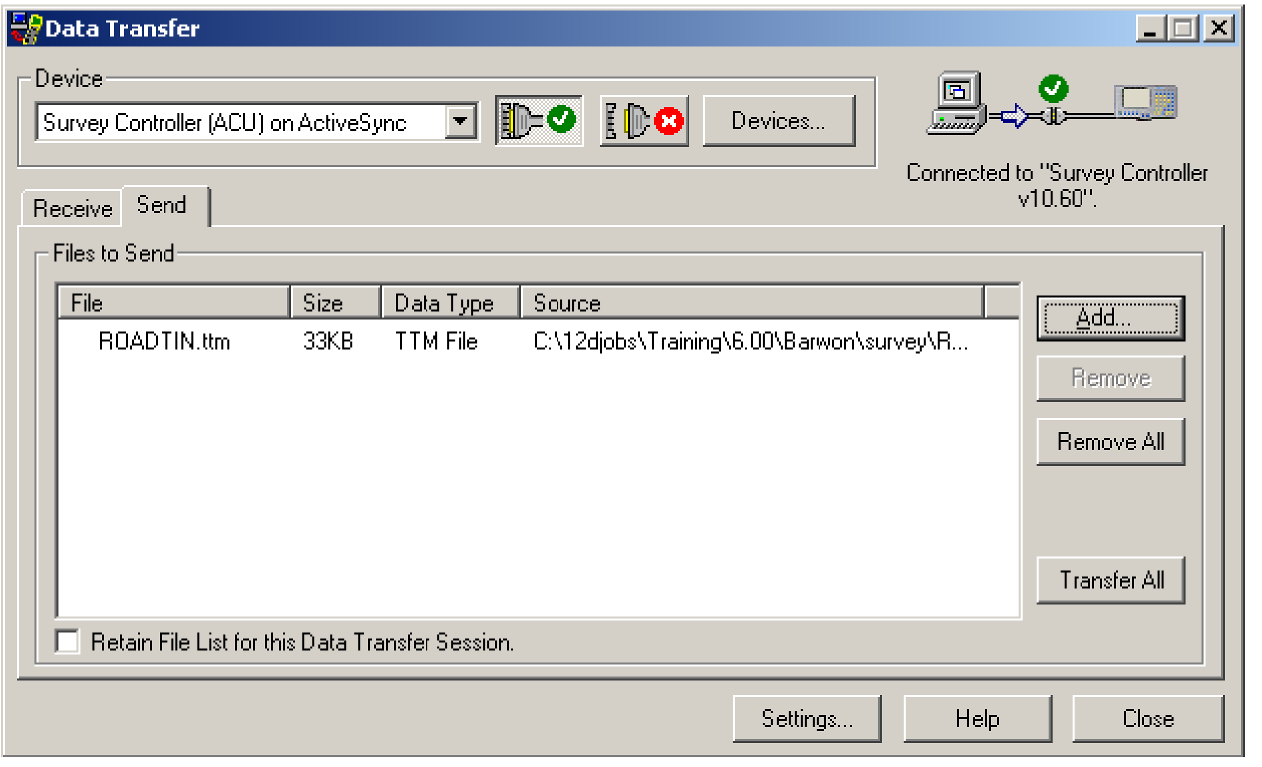
Connect the ACU to the PC using **TRIMBLE DATA TRANSFER**



Select ***Send*** tab then select ***Add***



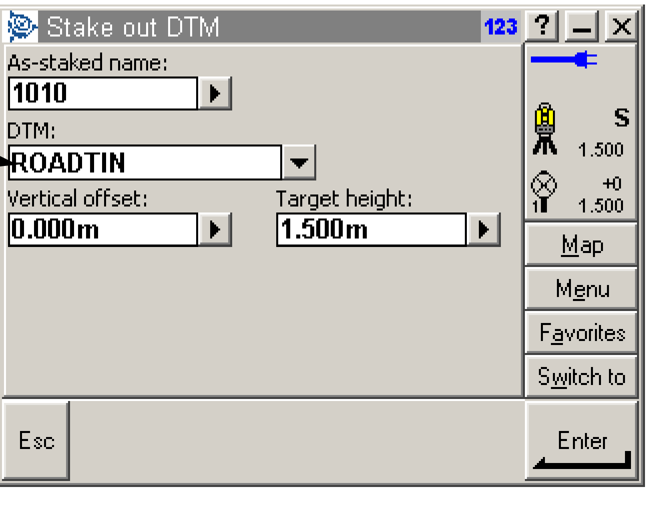
Select the tin file to upload using format ***Triangulated Terrain model***

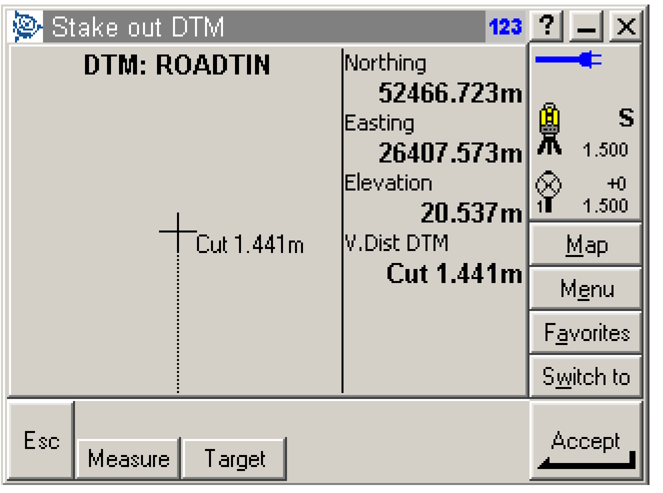


Select ***Transfer All*** to transfer the file

Once the tin has been copied into the ACU it can be called up when using the Stakeout option ***Survey=>Stakeout=>DTM***

Select the uploaded tin file to setout





The cut to the tin is calculated and displayed

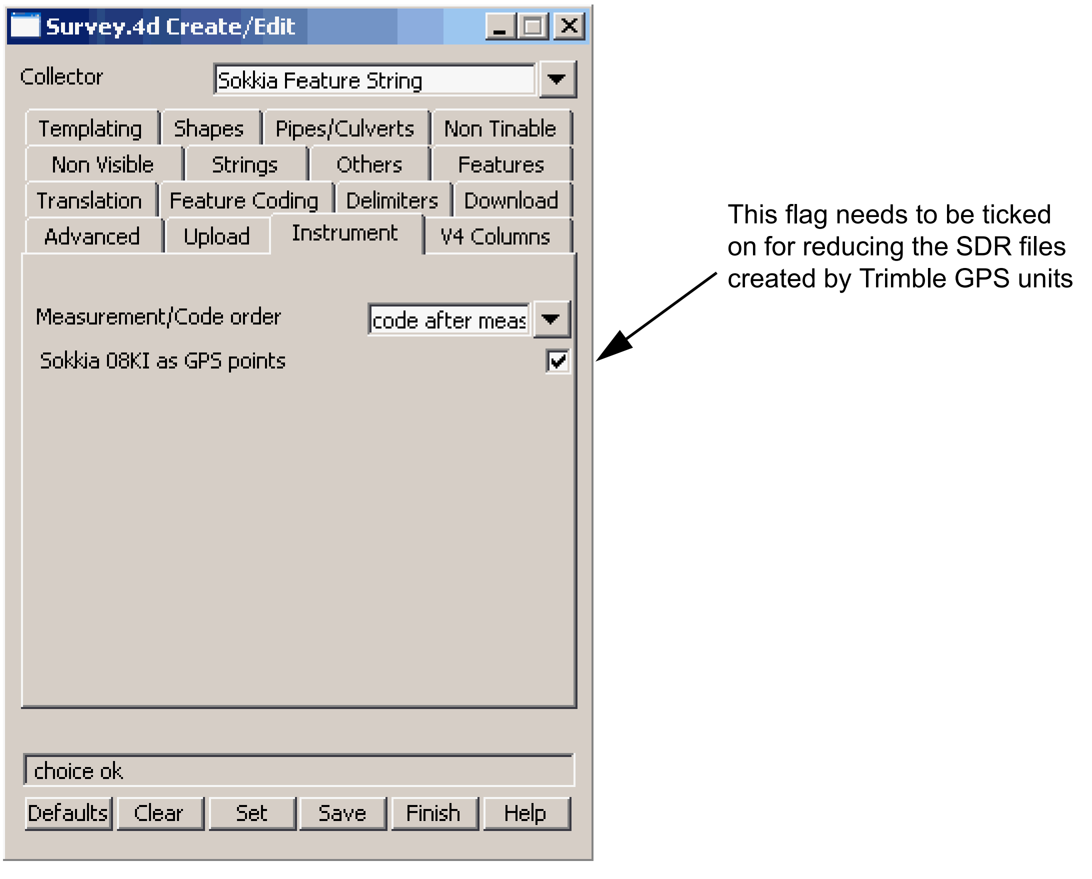
37.4 12d and Trimble GPS Controllers

The Trimble Geomatics Office software which comes with most Trimble GPS units can produce a Sokkia SDR file which is used by 12d Model as a raw data file.

However, the Trimble software writes out the (x,y,z) co-ordinates of GPS points as a "08KI" record which is normally treated by 12d Model as directly entered co-ordinates for a Station.

To overcome this problem, there is a flag in the data collector definitions to treat the Sokkia 08KI records as GPS points.

So before converting a Sokkia SDR file from a Trimble GPS unit, define a data collector which has a tick for the field *Sokkia 08KI as GPS points* on the *Instrument* tab of the **Survey.4d Create/ Edit** panel.



The code information that is entered on the Trimble in the field is appended the Sokkia 08KI record and is processed as blocks of information as given in the section 36.8 Field Coding for Non Leica Instruments and the appendix 37 12d and Survey Instruments.

37.5 12d and Geodimeter Instruments

This appendix deals with interfacing 12d with Geodimeter files.

See the sections:

37.5.1 12d UDS's

The geodimeter allows the recording of information in a format specified by the user. Data for each label can be prompted for and measurements from the instrument can be registered. The user can assign a certain series of labels to a User Defined Sequence or UDS.

For 12d to reduce files from the geodimeter instruments, the recorded information from the instrument must be in a specific format that 12d can understand. A number of Geodimeter UDS files are supplied on the ***12d Model*** Installation CD in the folder '*Other_Software\Geodimeter'.*

For uploading or direct entry of UDS's to your particular instrument see you instrument manual.

These UDS's have been set up so that 12d can convert the raw files from the instrument in a 12d field file. These UDS's should be installed ***prior*** *to undertaking surveys that are to be reduced by 12d. The standard UDS that 12d supply is as follows:*

UDS 1 - used within UDS 8 to record 12d field op codes.

UDS 2 - prompts for additional data for a 12d field op code.

UDS 5 - job administration at the start of a job

UDS 6 - station co-ordinates entry

UDS 7 - station set up - name and instrument height

UDS 8 - data pick-up

UDS 5 is run at the beginning of the job and it runs UDS 6, which runs UDS 7 which runs UDS 8.

UDS 1 can be run as required from within UDS8 to give 12d field file op codes.

UDS 1 - used within UDS 8 to record 12d field op codes

42=1
43=OP-CODE-SELECT
91=1
79=7/2

UDS 2 - used for additional data for 12d field ops codes

42=2
43=OP-CODE DATA
92=1
79=5

UDS 5 - job admin at start of day

42=5
43=TCC-ADMIN
79=10
0=1
53=1
51=0
79=7/6

UDS 6 - station co-ordinates entry

42=6
43=TCC-COORD-ENTRY
79=10
38=1
39=1
37=1
79=7/7

UDS 7 - station set up - name and instrument height

42=7
43=TCC-STN-ID
79=10
3=1
95=1
2=1
4=1
1=1
5=2
6=2
79=7/8

UDS 8 - data pick-up

42=8
43=TCC-DATA-PICKUP
5=4
4=8
6=8
39=8
8=0
9=0
7=0
6=8
5=8
4=8
38=8
37=8
39=8
6=3
4=3
1=3
90=3
79=5

12d standard UDS's for pick-up with Geodimeter

The Geodimeter data format is used as a raw data file by 12d Model and is converted into a 12d Field File before reduction by potentially 2 different macros:

1) geodat4d.4do (default) for use with the 12d standard UDS's.

2) geodat_qmrd.4do (Specialised reduction) for use with Queensland MRD UDS's.

For the geodat4d.4do macro (default) ***12d Model*** uses the following Geodimeter labels:

0=	Information
1=	Attributes
2=	Station name or named point
3=	Instrument height
4=	Blocks
5=	Point number
6=	Target (signal) height
7=	Horizontal angle
8=	Vertical angle
9=	Slope distance
37=	Northing (X) value
38=	Easting (Y) value
39=	Height (Z) value
51=	Date
53=	Operator
90=	End of record, flush buffer
91=	***12d Model*** field file op code
92=	Parameters for the previous 12d op code
95=	

In particular, the Geodimeter PCode label block (4=) is used as the text of blocks that are interpreted according to the descriptions given in the earlier section 36.8 Field Coding for Non Leica Instruments.

A new conversion macro for V6.0, geodat_qmrd.4do recognises geodimeter Program 22 and Program 32 commands **if the following setup/procedures are used**:

1. The geodimeter should be set to record program numbers. This is shown by the label **0=Pn** in the raw file, where **n** is the program number and is assigned by the instrument. To set this up on the instrument use MENU 6 1(Switches) and turn Prg_Num on.

2. For 12d to convert raw files using geodimeter programs such as P22 and P32 the data must contain a label **61=n** each time an UDS or onboard program is started (Where **n** is the program number). This allows 12d to be able to determine when one program is finished and another started. Queensland MRD have created there own UDS's which utilise the **61=** label so that it is prompted for within the UDS's. They have also set up a UDS (UDS 6) which is used to place the label at the commencement of an onboard program such as program 22.

Please continue to the next section 37.5.2 Using 12d Field Ops Codes on the Geodimeter.

37.5.2 Using 12d Field Ops Codes on the Geodimeter

In additional to the commands accessible from the blocks, most of the 12d field file op codes are accessible directly from the Geodimeter.

The label '91=' label is used to specify a 12d field file op code and any following '92=' labels provide any additional information required by the 12d op code.

The 12d field file ops codes apply to the measurement *before* the op code.

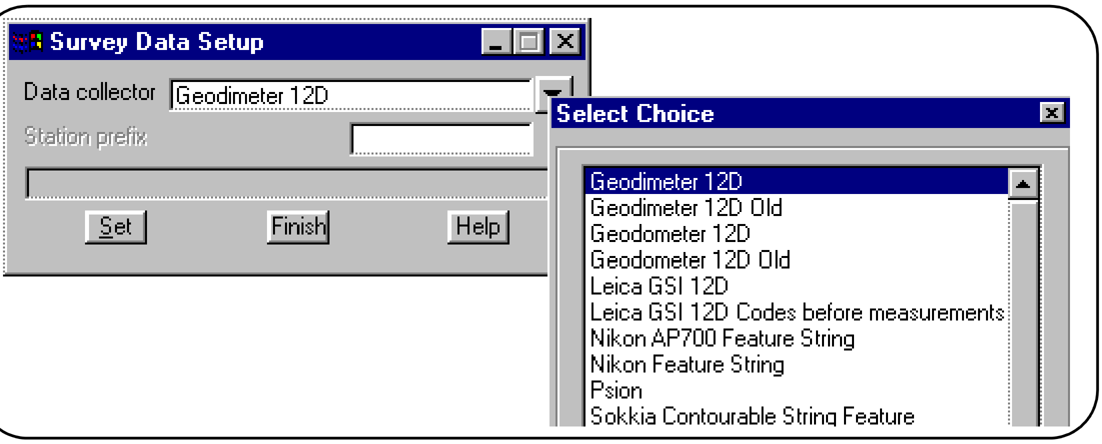
For a complete definition of the ***12d Model*** field file and the ***12d Model*** field file op codes, see the section 36.10 The 12d Field File Format

Please continue to the next section 37.5.3 Example of Geodimeter File.

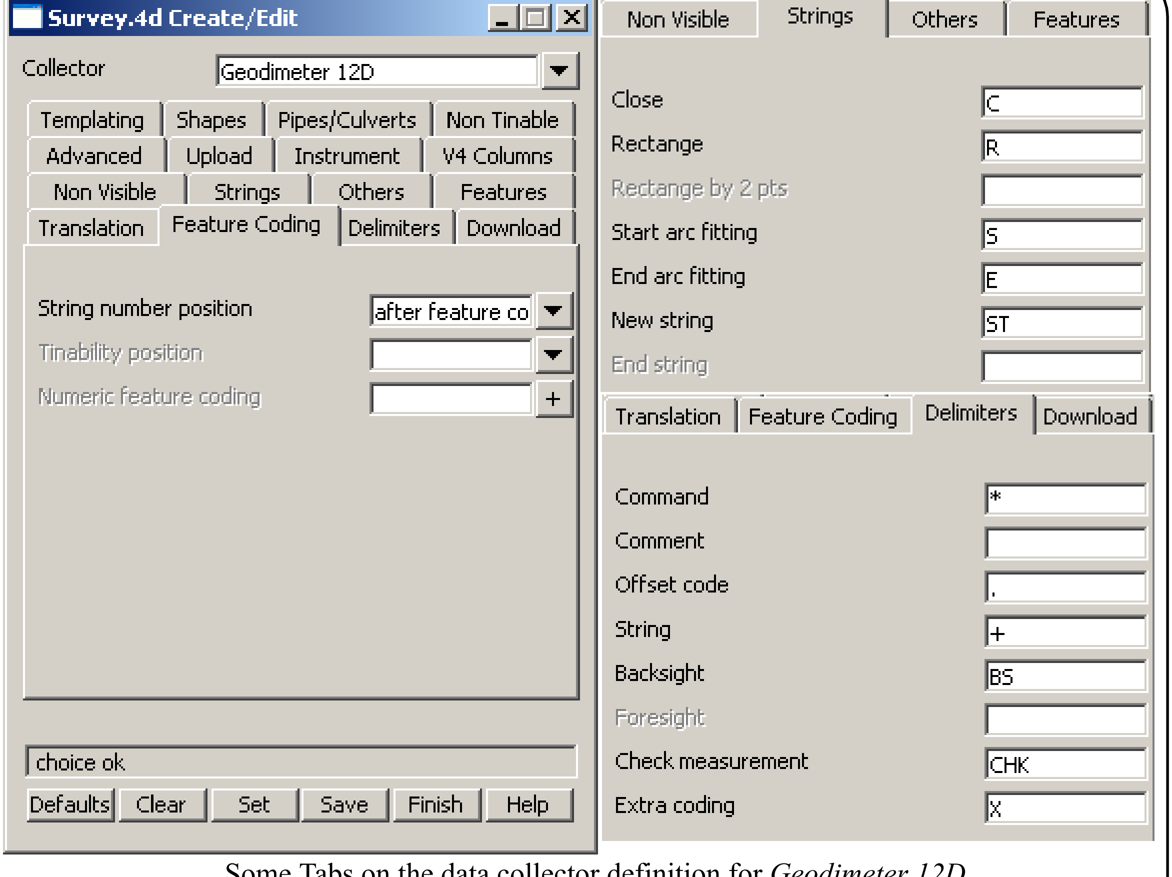
37.5.3 Example of Geodimeter File

The raw file 'Geodimeter.job' has been coded in accordance with the data collector definition *Geodimeter 12D* which is shipped with ***12d Model***. The ***12d Model*** Geodimeter UDS's were used in the field pick-up.

The data collector *Geodimeter 12D* is set using the option *Survey=>Setup*



Some of the tabs on the *Geodimeter 12D* data collection definition as show below. Note that ***** is used as the Command (Block) delimiter, the *string number* is given after the *feature code*, **S** is used to start arc fitting and **R** is the Rectangle (make a parallelogram) command.



Some Tabs on the data collector definition for *Geodimeter 12D*

The listing of the raw file is now given, followed by a dump of a ***12d Model*** view displaying the job with string names (feature codes) and point numbers toggled on. The raw file 'Geodimeter.job' is in the Survey area of the training data. It can be converted into a 12d field file by using the default conversion macro geodat4d.4do.

Description	Geodimeter file
run UDS 5 job name	0=OP CODE TEST
operator	53=DAD
date	51=2001.0525
UDS 6 station	38=5000.000
co-ordinates	39=10.000
	37=10000.000
UDS 7 instrument ht	3=1.654
	95=0
	2=1
Feature code (FC) PISP	4=PISP
	1=Peg
UDS 8 point no 1000	5=1000
vertical angle	8=90.1219
slope distance	9=5.80
horizontal	7=119.4705
target (signal) height	6=1.500
FC PSHT	4=PSHT
***note** that the string no.	1=
(SN) defaults to 0	90=0
point no 1001	5=1001
vertical angle	8=90.1207
slope distance	9=15.80
horizontal	7=119.4708
target height	6=1.500
FC *EP* SN *1* *start arc*	4=EP1*S
	1=
point no 1002	90=0
vertical angle	5=1002
slope distance	8=90.1215
horizontal	9=25.80
target height	7=119.4705
FC *RC* SN *1* *start arc*	6=1.500
	4=RC1*S
	1=
	90=0
	5=1003
	8=90.1158
	9=35.80
	7=119.4705
	6=1.500
	4=EP2
	1=
	90=0
	5=1004
	8=90.1209
	9=45.80
	7=119.4706
	6=1.500
	4=DL1
	1=
	90=0

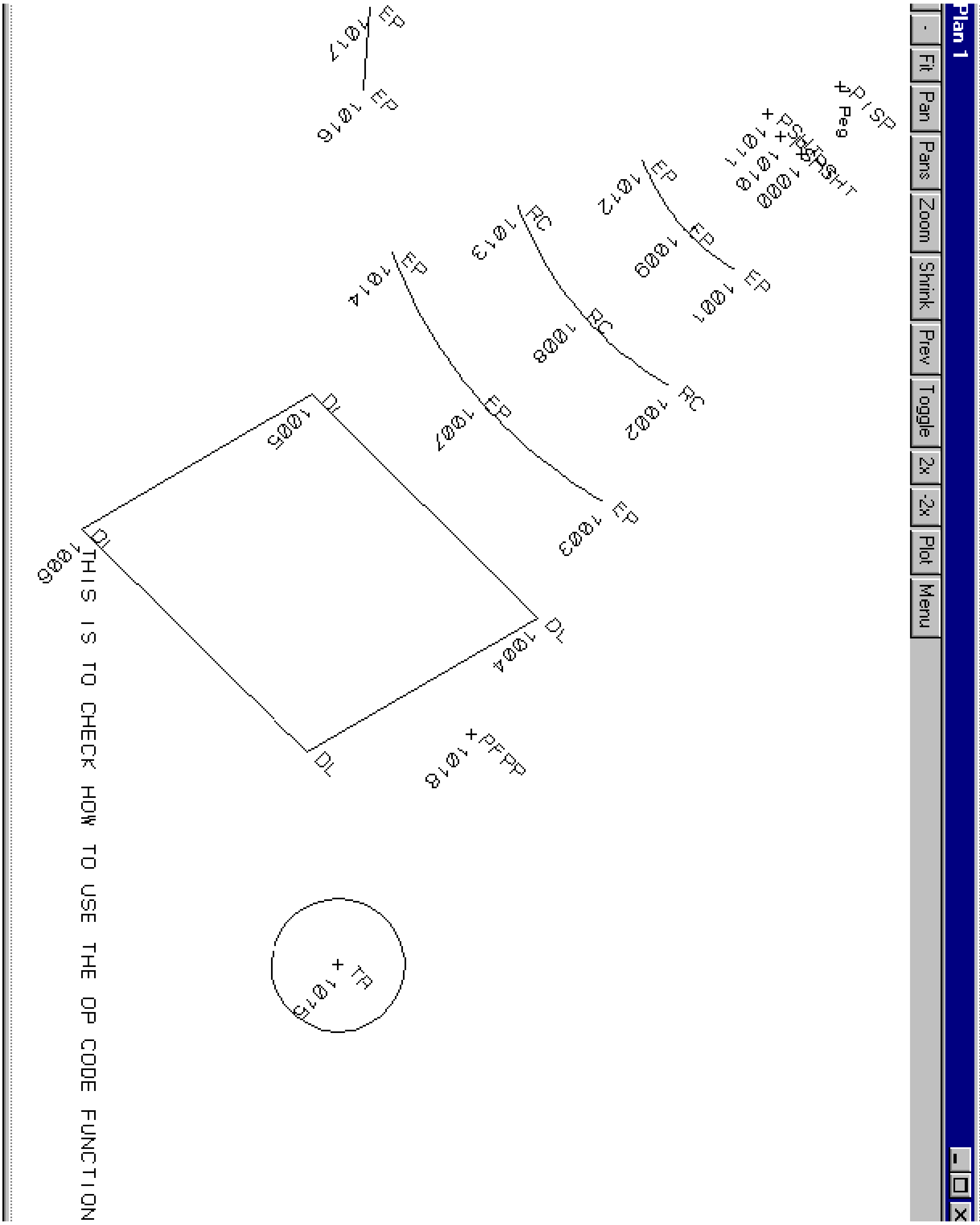
Description	Geodimeter file
point no 1005	5=1005
vertical angle	8=90.1203
slope distance	9=45.80
horizontal	7=149.4706
target height	6=1.500
FC *DL* SN *1*	4=DL1
	1=
	90=0
point no 1006	5=1006
vertical angle	8=90.1159
slope distance	9=65.80
horizontal	7=149.4706
target height	6=1.500
FC *DL* SN*1*	4=DL1
	1=
	90=0
UDS 1 **12d op code 45**	91=45
UDS 1 **12d op code 41**	91=41
UDS 2 parameters	92=THIS IS TO CHECK
for op code 45 (additional text)	92= HOW TO USE THE
UDS 8 point no 1007	92= OP CODE FN
vertical angle	5=1007
slope distance	8=90.1219
horizontal	9=35.80
target height	7=139.4711
FC *EP* SN *2*	6=1.500
	4=EP2
***note** that 12d op code 45 is 'make a parallelogram from the last three points' and it is applies to the measurement before the command. The rectangle R could have been instead as part of the feature code block. That is " 4=DL1*R" instead of the "91=45"	1=
	90=0
	5=1008
	8=90.1214
	9=25.80
	7=139.4706
	6=1.500
	4=RC1
	1=
	90=0
***Also note** that the 12d op code 41 is 'additional text' and the additional text is given by the "92=" lines. Again the op code applies to the measurement before the command..	5=1009
	8=90.1211
	9=15.80
	7=139.4706
	6=1.500
	4=EP1
	1=
	90=0

continue on next page

Description	Geodimeter file
point no 1010	5=1010
vertical angle	8=90.1209
slope distance	9=5.80
horizontal	7=139.4709
target height	6=1.500
FC *PSHT* SN *0*	4=PSHT
	1=
	90=0
	5=1011
	8=90.1141
	9=5.80
	7=159.4714
	6=1.500
	4=PSHT
	1=
	90=0
	5=1012
	8=90.1138
	9=15.80
	7=159.4711
	6=1.500
	4=EP1
	1=
	90=0
	5=1013
	8=90.1141
	9=25.80
	7=159.4709
	6=1.500
	4=RC1
	1=
	90=0
point no 1014	5=1014
vertical angle	8=90.1141
slope distance	9=35.80
horizontal	7=159.4713
target height	6=1.500
FC *EP* SN *2*	4=EP2
	1=
	90=0
UDS 1 **12d op code 17**	91=17
which is 'arc through previous three points'	continued in next column

Description	Geodimeter file
point no 1015	5=1015
vertical angle	8=90.1141
slope distance	9=75.80
horizontal	7=119.4709
target height	6=1.500
FC *TR* SN *1*	4=TR1
	1=
	90=0
UDS 1 **12d op code 18**	91=18
UDS 2 Radius 5	92=5
which is 'circle feature' with a radius of 5	5=1016
	8=90.1146
	9=35.80
	7=179.4713
	6=1.500
	4=EP2
	1=
	90=0
	91=47
	5=1017
	8=90.1145
	9=35.80
	7=189.4714
	6=1.500
	4=EP2
	1=
	90=0
point no 1018	5=1018
vertical angle	8=90.1145
slope distance	9=55.80
horizontal	7=119.4713
target height	6=1.500
FC *PFPP* SN *1*	4=PFPP
	1=
	90=0
UDS 1 **12d op code 40**	91=40
which is 'make a point non-tinable' This applied to the previous point (1018)	

Plan View with of the Reduced Geodimeter Data Displayed
with Point Numbers and String Names (Feature Codes) Turned On



The following raw field file has been taken from an QMRD Geodimeter 610 which has the QMRD UDS's installed. The file can be converted into a 12d field file by using the geodat_qmrd.4do. Note the inclusion of the activity codes (61=) that have been added to the file with the use of the UDS's. It also shows the use program 22 and the traverse extraction facility. For more information on traverse extraction see the section 36.6 Traverse coding.

Description	Geodimeter file
	50=030602
Activity code from UDS	61=1
	6=1.458
	54=ROGER-RANGER STN
	0=OVERLAY
	53=RAF
	51=03-06-2002
Run UDS 6 to place	61=22
activity code (61=) before	0=P22
program 22	2=PSM45026 PBMK
	3=1.561
	5=STN1
foresight to STN1,	4=TL1 FSSTN1
traverse line(TL), string 1	6=1.475
	7=213.0004
	8=89.1413
	9=98.297
	17=33.0014
	18=270.4550
	24=213.0009
	25=89.1417
UDS 6 run to label	61=22
activity code	0=P22
	2=STN1 PISP
	3=1.570
	5=PSM45026
Backsight(RO)	4=TL1 ROPSM45026
reading,trav line 1 to	6=1.468
PSM45026	7=32.5955
	8=90.5246
	9=98.300
	17=212.5956
	18=269.0716
	24=32.5955
	25=90.5249
	5=STN2
foresight to STN2,	4=TL1 FSSTN2
traverse line(TL), string 1	6=1.448
	7=182.1037
and so forth....	8=89.4211
	9=162.715
	17=2.1040
	18=270.1759
	24=182.1032
	25=89.4221
	61=22
	0=P22
	2=STN2 PISP
	3=1.542
	5=STN1

Description	Geodimeter file
	4=TL1 ROSTN1
	6=1.475
	7=10.1654
	8=90.2218
	9=162.710
	17=190.1657
	18=269.3740
	24=10.1652
	25=90.2216
	5=STN3
	4=TL1 FSSTN3
	6=1.486
	7=195.4510
	8=90.0511
	9=257.710
	17=15.4508
	18=269.5452
	24=195.4513
	25=90.0514
	61=22
	0=P22
	2=STN3 PISP
	3=1.581
	5=STN2
	4=TL1 ROSTN2
	6=1.448
	7=187.1306
	8=89.5740
	9=257.710
	17=7.1306
	18=270.0232
	24=187.1306
	25=89.5752
	5=STN4
	4=TL1 FSSTN4
	6=1.400
	7=23.1437
	8=87.3949
	9=173.958
	17=203.1434
	18=272.2031
	24=23.1440
	25=87.4008
	61=22
	0=P22
	2=STN4 PISP
	3=1.546
	5=STN3
	4=TL1 ROSTN3
	6=1.487

continue on next page

Description	Geodimeter file
	7=199.0404
	8=92.2432
	9=173.963
	17=19.0353
	18=267.3552
	24=199.0414
	25=92.2456
	5=STN5
	4=TL1 FSSTN5
	6=1.457
	7=34.4057
	8=88.1846
	9=151.526
	17=214.4059
	18=271.4140
	24=34.4055
	25=88.1912
	61=22
	0=P22
	2=STN5 PISP
	3=1.552
	5=STN4
	4=TL1 ROSTN4
	6=1.452
	7=184.4113
	8=91.4601
	9=151.527
	17=4.4111
	18=268.1402
	24=184.4114
	25=91.4603
	5=STN6
	4=TL1 FSSTN6
	6=1.496
	7=19.0418
	8=89.5731
	9=179.824
	17=199.0420
	18=270.0229
	24=19.0416
	25=89.5730
	61=22

continued in next column

Description	Geodimeter file
	0=P22
	2=STN6 PISP
	3=1.591
	5=STN5
	4=TL1 ROSTN5
	6=1.458
	7=179.5456
	8=90.0633
	9=179.821
	17=359.5457
	18=269.5333
	24=179.5456
	25=90.0639
	5=STN7
	4=TL1 FSSTN7
	6=1.453
	7=7.5532
	8=91.2051
	9=149.687
	17=187.5533
	18=268.3914
	24=7.5531
	25=91.2056
	61=22
	0=P22
	2=STN7 PISP
	3=1.547
	5=STN6
	4=TL1 ROSTN6
	6=1.496
	7=196.4115
	8=88.4357
	9=149.679
	17=16.4116
	18=271.1600
	24=196.4115
	25=88.4353
	5=STN8
	4=TL1 FSSTN8
	6=1.478
	7=8.1420
	8=91.5307
	9=205.948

continue on next page

Description	Geodimeter file
	17=188.1422
	18=268.0657
	24=8.1418
	25=91.5310
	61=22
	0=P22
	2=STN8 PISP
	3=1.574
	5=STN7
	4=TL1 ROSTN7
	6=1.453
	7=181.1643
	8=88.1026
	9=205.939
	17=1.1647
	18=271.4934
	24=181.1638
	25=88.1026
	5=STN9
	4=TL1 FSSTN9
	6=1.517
	7=359.0523
	8=89.1311
	9=199.132
	17=179.0523
	18=270.4648
	24=359.0523
	25=89.1310
	61=22
	0=P22
	2=STN9 PISP
	3=1.614
	5=STN8
	4=TL1 ROSTN8
	6=1.478
	7=173.0014
	8=90.5031
	9=199.132
	17=353.0017
	18=269.0934
	24=173.0011
	25=90.5037
	continued in next column

Description	Geodimeter file
	5=STN10
	4=TL1 FSSTN10
	6=1.400
	7=345.0918
	8=86.2903
	9=221.043
	17=165.0917
	18=273.3103
	24=345.0920
	25=86.2908
	61=22
	0=P22
	2=STN10 PISP
	3=1.583
	5=STN9
	4=TL1 ROSTN9
	6=1.517
	7=165.4152
	8=93.3413
	9=221.053
	17=345.4122
	18=266.2600
	24=165.4221
	25=93.3425
	5=STN11
	4=TL1 FSSTN11
	6=1.460
	7=0.2903
	8=86.1535
	9=215.241
	17=180.2930
	18=273.4438
	24=0.2836
	25=86.1548
	50=ERICC
	61=1
	6=1.443
	54=ROGER-RANGER STN
	0=A
	53=ECB
	51=04-06-2002
	61=22
	continue on next page

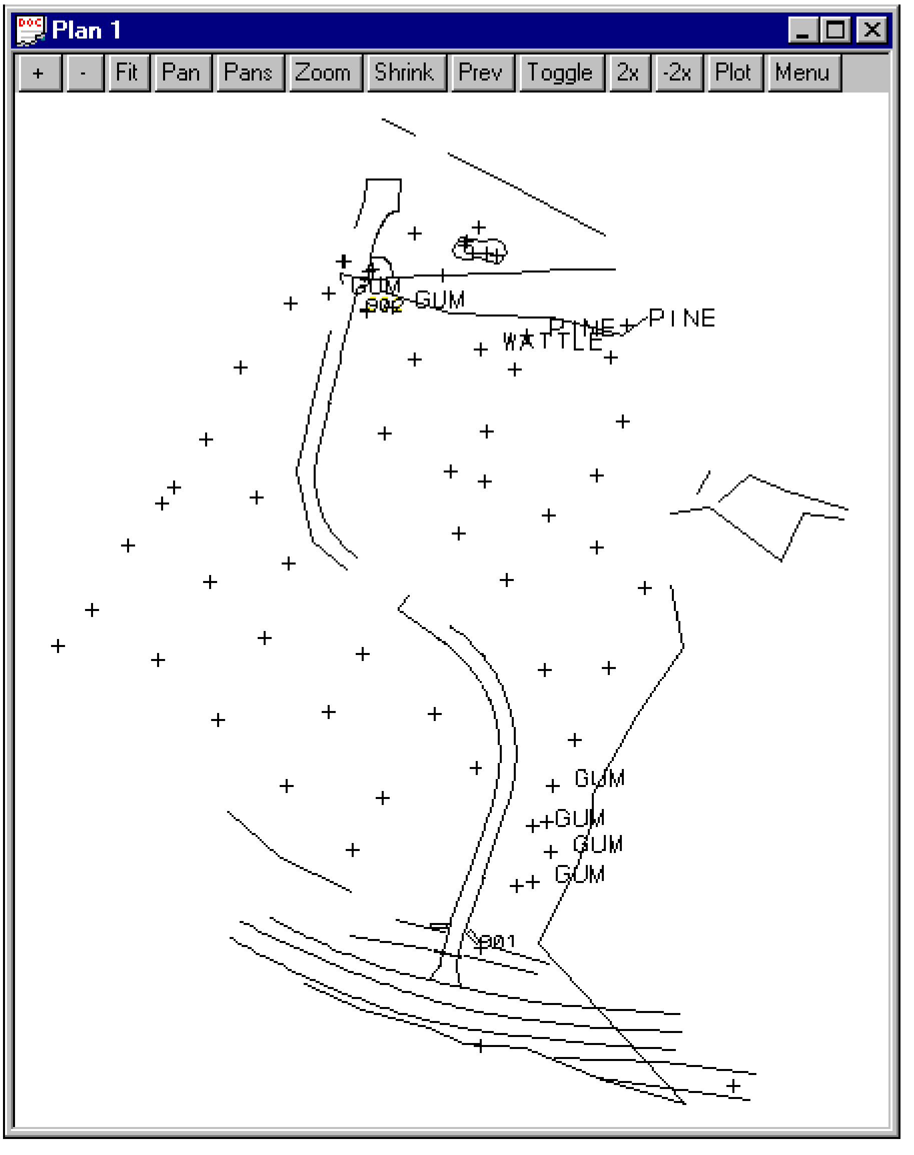
Description	Geodimeter file	Description	Geodimeter file
	0=P22 2=STN11 PISP 3=1.573 5=STN10 4=TL1 ROSTN10 6=1.520 7=131.1633 8=93.4738 9=215.253 17=311.1639 18=266.1222 24=131.1626 25=93.4738 5=STN12 4=TL1 FSSTN12 6=1.465 7=307.3035 8=87.2924 9=191.435 17=127.3037 18=272.3034 24=307.3033 25=87.2922 61=22 0=P22 2=STN12 PISP 3=1.558 5=STN11 4=TL1 ROSTN11 6=1.477 7=11.1452 8=92.3416 9=191.448 17=191.1454 18=267.2611 24=11.1449 25=92.3443 5=STN13 4=TL1 FSSTN13 6=1.443 7=180.1114 8=89.1045 continued in next column		9=222.450 17=0.1114 18=270.4942 24=180.1113 25=89.1113

37.6 12d and Topcon Instruments

The Topcon data format is used as a raw data file by 12d Model and is converted into a 12d Field File before reduction.

Description	Sokkia SDR 20 File
00 - SDR format	00NMSDR20 V03-05 03-Aug-00 09:00 111121
	10NM1697 DETAIL
	13CPSea level crn: N
	13CPC and R crn: N
	13CPAtmos crn: N
	06NM1.00000000
	13OOCurrent view
	13TS02-Aug-00 08:23
	13JS10000
	13TS02-Aug-00 08:25
	01NM: 000000 00000031 0.00000000
	13PCP.C. mm Applied: 0.000
02 - station details for pt 902	02TP09020982.770005096.700000115.7500001.7400000STN
02 - station details for pt 901	02TP09011000.000005000.000000100.000001.66500000STN
07 - back bearing from pt 901 to 902	07TP09010902349.895000349.895000
03 - target height	03NM1.44000000
check shot from pt 901 to pt 902	09F109011001099.43000081.0180555349.893611CHK902
measurement FC TBL SN 1 Pt 1003	09F109011003046.93700089.5425000119.69972201TBL
FC TBL SN 1 Pt 1003	09F109011004037.34200089.7522222125.31527701TBL
	09F109011005026.44500089.9869444138.51611101TBL
FC BB SN 1 Pt 1006	09F109011006045.68200089.8705555114.79527702BB
	09F109011007032.72000090.2769444122.23361102BB
	09F109011008020.06200090.9736111147.16833302BB
FC ES SN 3 Pt 1009 start template	09F109011009033.47400090.0641666118.03777703ES*XA
FC CR SN 4 Pt 1010	09F109011010032.98200089.8936111113.25027704CR
FC ES SN 5 Pt 1011	09F109011011031.88200090.0369444108.80138805ES
start using template in zig mode	09F109011012022.46200090.2811111130.749722XB
using template	09F109011013021.56900090.1400000124.623055
	09F109011014019.30400090.5197222118.459722
	09F109011015011.35800091.0744444136.851388S
	09F109011016013.58000090.5797222144.980000S
	09F109011017015.06100090.5494444154.349722S
	09F109011018012.79200090.7638888201.297500
	09F10901101909.757000091.1033333201.376944
pause template - meas FC PP SN 0	09F109011110043.58500089.6561111118.780000PP*XD
continue with template	09F10901102006.659000092.5538888198.245000XE
start arc through 3 points	09F109011021014.70100091.2202777256.936111S
start arc through 3 points	09F109011022015.55300090.8763888248.602222S
start arc through 3 points	09F109011023018.16300090.6650000240.646111S
	09F109011024027.42900090.5169444260.969444
	09F109011025025.37600090.5413888266.748055
	09F109011026026.05000090.6897222272.995833
end arc through 3 points	09F109011027032.16700090.5105555277.837500E
end arc through 3 points	09F109011028036.58100090.2566666276.025833E
end arc, stop template	09F109011029038.06400090.1986111272.082777E*XC
FC BB SN 6 Pt 1030	09F109011030019.76600091.8072222275.24777706BB
	09F10901103107.245000093.9725000266.13777706BB
FC SWUG SN 7 Pt 1032 obv 0.225 dia	09F10901103207.044000094.6780555265.51250007SWUG*O.225
FC SWUG SN 7 Pt 1033 invert meas	09F10901103303.441000097.8750000242.44805507SWUG*I
FC BB SN 8 Pt 1034	09F10901103402.973000099.2075000239.99916608BB
	09F10901103509.678000092.9816666114.81555508BB
	09F109011036010.50700090.6511111104.50611109TBR
	09F10901103701.155000095.167500082.357777809TBR
	09F10901103805.969000090.5472222291.73500010TBR
	09F109011039013.82100090.8102777288.00027710TBR
FC WA SN 11 Pt 1040	09F10901104008.206000091.0138888290.96333311WA
	09F10901104108.476000091.0427777293.60750011WA

FC WA SN 11 Pt 1042	09F10901104206.213000089.7200000305.83194411WA
FC WA SN 11 Pt 1043- close string	09F10901104306.116000090.3575000300.03777711WA*C
FC WA SN 12 Pt 1044	09F10901104403.100000090.0911111314.07277712WA
FC WA SN 12 Pt 1045	09F10901104502.970000089.6530555327.17166612WA
FC WA SN 12 Pt 1046 - create rect	09F10901104600.760000099.5219444351.28472212WA*R
FC PL SN 13 Pt 1048	09F109011048027.45200090.7930555258.30333313PL
omit part of file	.omit part of file
FC TBL SN 21 Pt 1103	09F109011103021.21900088.8433333293.53222221TBL
FC TR0306 SN 0 - comment GUM	09F109011105012.69000087.209444438.3036111TR0306 GUM
FC TR0309 SN 0 - comment GUM	09F109011106017.95900086.573333336.3436111TR0309 GUM
FC TR0306 SN 0 - comment GUM	09F109011107019.97400086.607777723.2163889TR0306 GUM
FC TR0309 SN 0 - comment GUM	09F109011108026.78400086.450833323.7155556TR0309 GUM
FC TR0308 SN 0 - comment GUM	09F109011109027.38900085.6861111358.734166TR0608 GUM
02 - station details pt 902	02TP09020982.770005096.710000115.7500001.7100000STN
07 - back bearing from pt 902 to 901	07TP090209010169.895000169.895000
03 - target height	03NM1.440000000
check measurement to pt 901	09F109021201099.43000099.2611111169.893611CHK901
FN DW SN 21 Pt 1202	09F109021202038.369000100.2900000182.1266621DW*S
FC DW SN 21 Pt 1203	09F109021203033.80100099.7461111190.31638821DW
omit part of file	.omit part of file
FC FE SN 25 Pt 1346 - non tin segm	09F10902134604.867000085.22555555.8144444025FE*XN
FC FE SN 25 Pt 1347	09F109021347010.66400085.510000060.870277825FE
.omit part of file	.omit part of file
Pt 2135 FC DW SN 25 start arc	09F10902135708.003000085.01361113.5244444027DW*S*28WA
and also FC WA SN 28	
	.omit to end of file



Plan View with of the Reduced Sokkia Data Displayed
Note that no linestyles or symbols have been turned on.

38 Geodetics Summary

Various options in ***12d Model*** use geodetic calculations to present and change data. These options use terminology that are common to the field of geodetics, which will be defined here.

Most of the terminology adopted follows definitions given in the Australian **GDA Technical Manual** which is published by the **Intergovernmental Committee on Surveying and Mapping** (ICSM). This publication is a valuable reference document and the reader is encouraged to obtain a copy for a full understanding of the topic. The document can be accessed on the internet at the following address http://www.icsm.gov.au/gda/tech.html

See

38.1 Shape Of The Earth

38.2 Geodetic Coordinates

38.3 Map (Cartographic) Projections

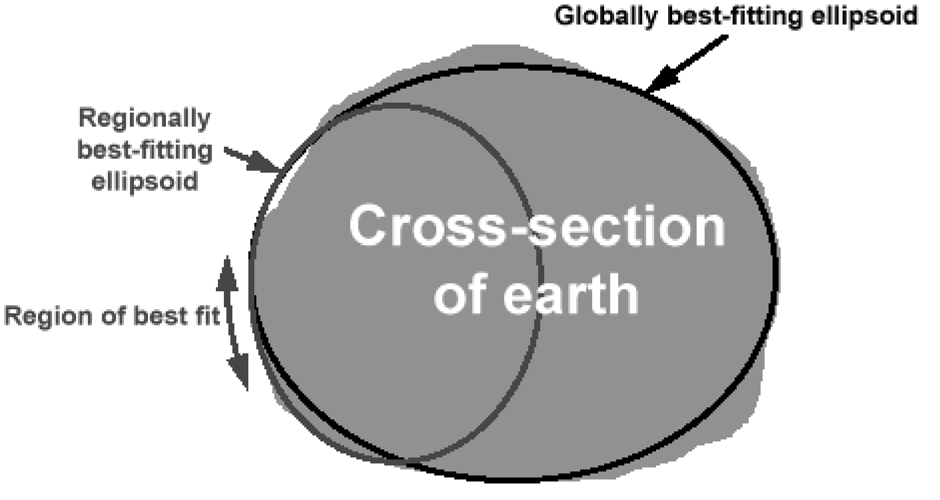
38.4 Terminology

38.5 Converting Between AMG, ISG and MGA

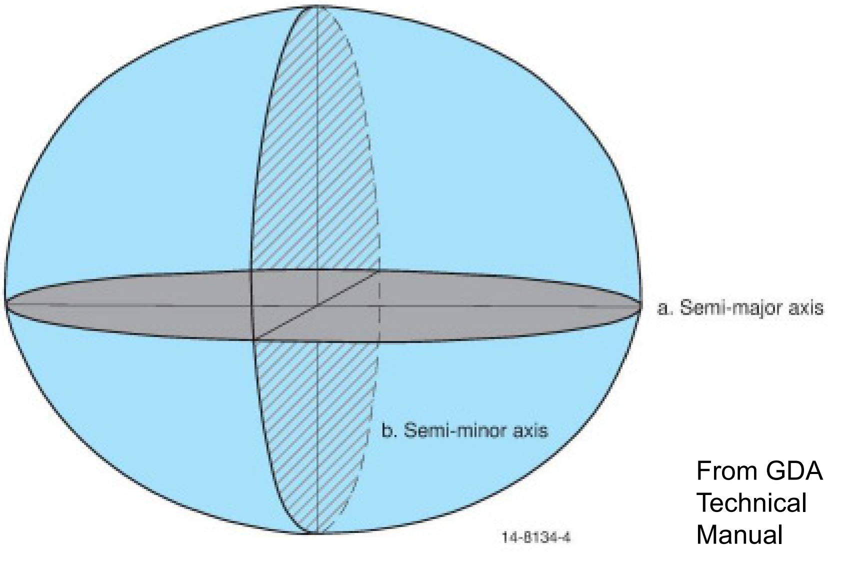
38.1 Shape Of The Earth

The determination of the Earth's shape is a science known as **Geodesy**. Today, it is widely accepted that the Earth's shape best approximates an **ellipsoid** that has been revolved around the Earth's polar axis. Put another way, the shape is a sphere that has been squashed at the north and south poles. The non-spherical shape is due to gravity.

A number of ellipsoids have been calculated to best approximate the Earth's shape at local locations and the earth as a whole. The best fit is concerned with matching the Earth's equipotential gravity field (the **Geoid** that is best approximated by Mean Sea Level), to a geometric ellipsoid shape. As such, there a wide number of definitions.



The most common way of defining an ellipsoid is by describing the **semi-major axis** value and an **inverse flattening** value (this parameter describes the "squashing" of the ellipse).



From GDA Technical Manual

Some commonly used ellipsoids are:

1. GRS80

Ellipsoid	**Semi-major axis**	**Inverse flattening**
GRS80	6,378,137.0	298.257222101

This ellipsoid is used for Australia's GDA definition (Geocentric Datum of Australia GDA 94) used for MGA (Map Grid of Australia) calculations, New Zealand's NZGD2000 datum as well as other geocentric earth model datums around the world.

2. ANS

Ellipsoid	Semi-major axis	Inverse flattening
ANS	6,378,160	298.25

This was the ellipsoid used to define the Australian Geodetic datum (AGD 84) used for AMG (Australian Map Grid) calculations and ISG (Integrated Survey Grid) co-ordinates.

3. NZ Geodetic 49

Ellipsoid	Semi-major axis	Inverse flattening
NZ Geodetic 49	6,378,399.065	297.0

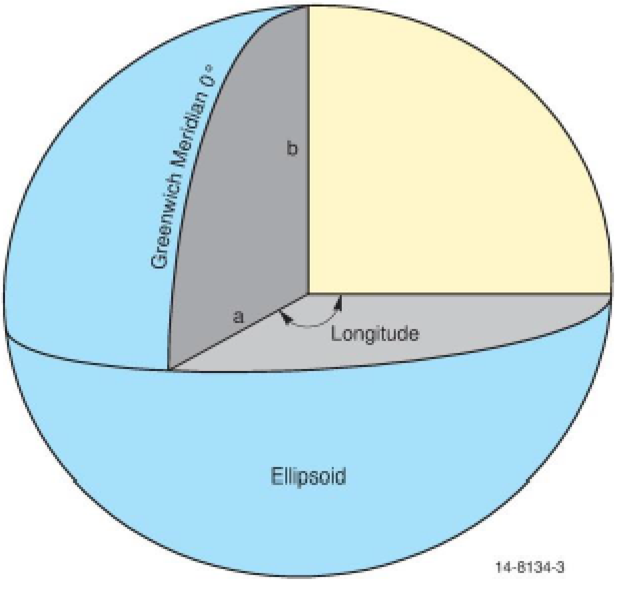
This was the ellipsoid used to define the NZ 1949 Geodetic datum. The semi-major axis given here has been adjusted to compensate for errors in units conversion from links to meters.

Continue to the next section 38.2 Geodetic Coordinates or return to 38 Geodetics Summary.

38.2 Geodetic Coordinates

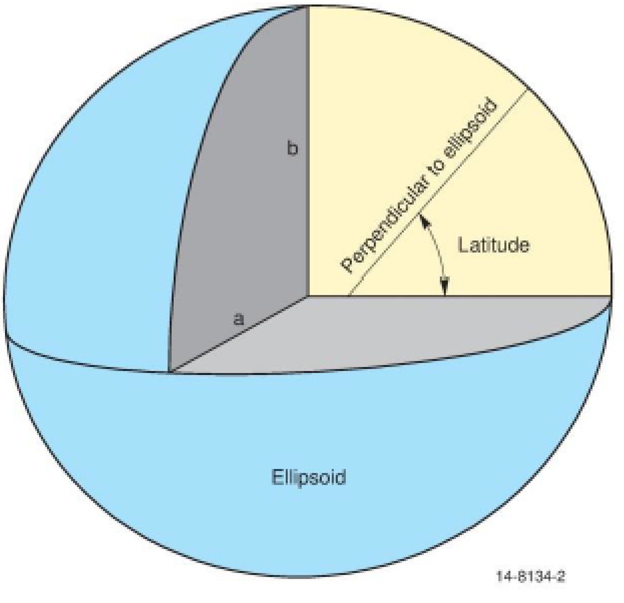
Once an ellipsoid or **Geodetic Datum** is defined, a position on the earths surface can be described in terms of Geodetic coordinates. These coordinates are **Longitude**, **Latitude** and **Ellipsoid height**.

Longitude is a angular quantity measured from the Greenwich meridian. It is most commonly described in terms of degrees, minutes, seconds East or west of the Greenwich meridian.



From GDA Technical Manual

Latitude is a angular quantity measured from the equatorial plane, to the plane defined by the point position and the plumb line to the ellipsoid surface. It is most commonly described in terms of degrees, minutes, seconds South or North to the equator.

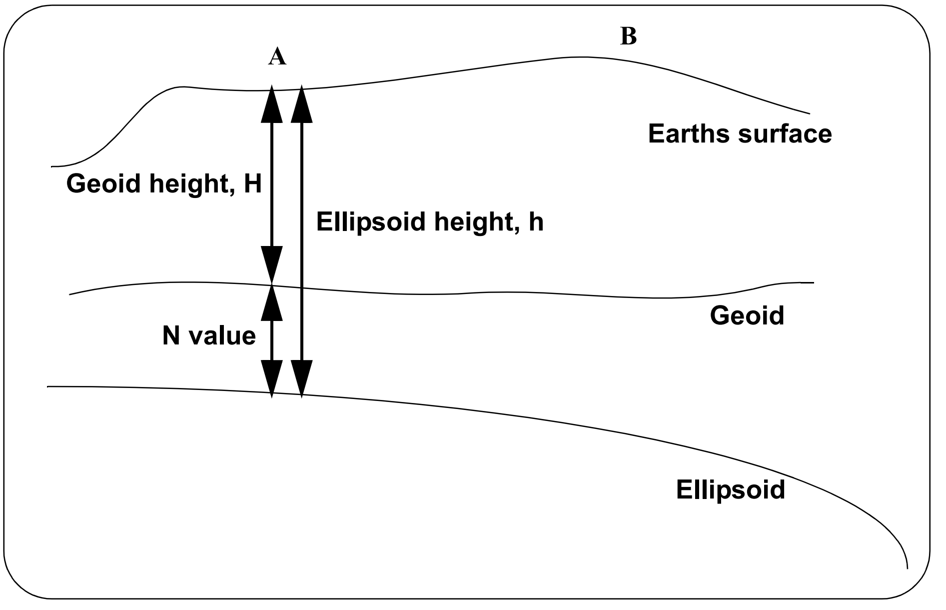


From GDA Technical Manual

The ***ellipsoid height***, h is the height above the reference ellipsoid.

Most height datums are not based on ellipsoid height but are based on the ***geoid***.e.g. **AHD** in Australia.

As such, levels from GPS observations (which are *ellipsoid* heights) need to be corrected to a **geoidal** or **orthometric height**. To do this, we require the separation or gap distance between the two different surfaces. This separation is known as the **N value**.



The **Ellipsoid height** = **Geoid height** + **N** value

or

$$h = H + N$$

N values can be defined in a geoidal model such as **Ausgeoid98** which represents grids of **N** values over all of Australia.

For a given Geodetic coordinate, an N value can be interpolated from the model and applied to the ellipsoid height to give a geoidal height. Similarly, the N value can be used to convert a geoidal height to an ellipsoid height.

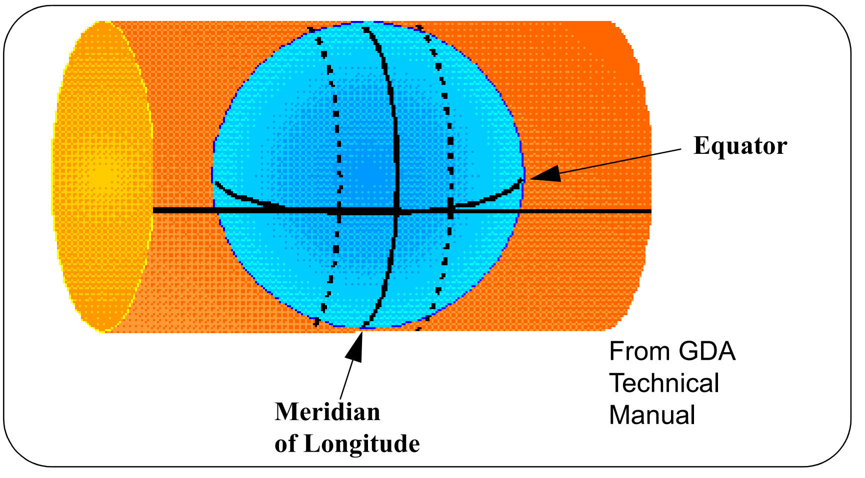
Since most geodetic calculations are made on the ellipsoid, the ***ellipsoid height*** is required for precise calculations.

Continue to the next section 38.3 Map (Cartographic) Projections or return to 38 Geodetics Summary.

38.3 Map (Cartographic) Projections

In order to represent ellipsoid data on a flat surface for mapping, it is necessary to use a projection. A projection enables points on the earths surface to be mathematically projected onto an imaginary developable surface. This surface can then be developed or “rolled flat”. Typically, this surface is a cylinder or cone.

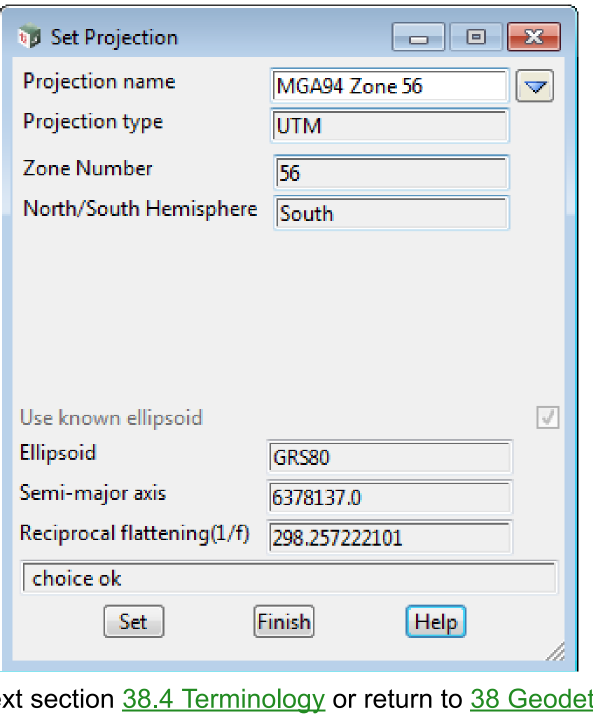
The Transverse Mercator system (TM) projects coordinates onto a cylinder that is tangent to the equator and the entire length of a meridian of Longitude.



12d Model allows a number of projections to be specified including, Universal Transverse Mercator (UTM), Transverse Mercator (TM) and Rectified Skew Orthomorphic (RSO). There are many more available using the **General** type.

Within 12d Model, a projection can be defined that specifies both the ***reference ellipsoid*** and ***projection type***. This then can be used for geodetic calculations.

A projection has various parameters that define it. These are specific to the projection and are clearly defined for major mapping systems. For example MGA94 zone 56 projection is defined as follows:



Continue to the next section 38.4 Terminology or return to 38 Geodetics Summary.

38.4 Terminology

The various geodetic options in 12d Model mostly use standard geodetic terminology as defined in the Australian GDA technical manual. For clarity, they will be defined again here. For some options it is important to note that some terminology used in Australia has quite a differing meaning in other countries.

See

38.4.1 Ellipsoid Distance

The ***ellipsoid distance*** is the reduced distance along the *surface* of the ellipsoid. Standard survey measurements are reduced to the horizontal but require a correction due to the height above the ellipsoid. This is usually done by a **height scale factor** which takes into account the ellipsoid height at each end of the measured line. i.e.

$$HeightScaleFactor = 1 - \frac{h_M}{R + h_M}$$

Where:

$\mathbf{h_M}$ = Mean terrain height (mean of the two ellipsoid heights at either end of the measured line)

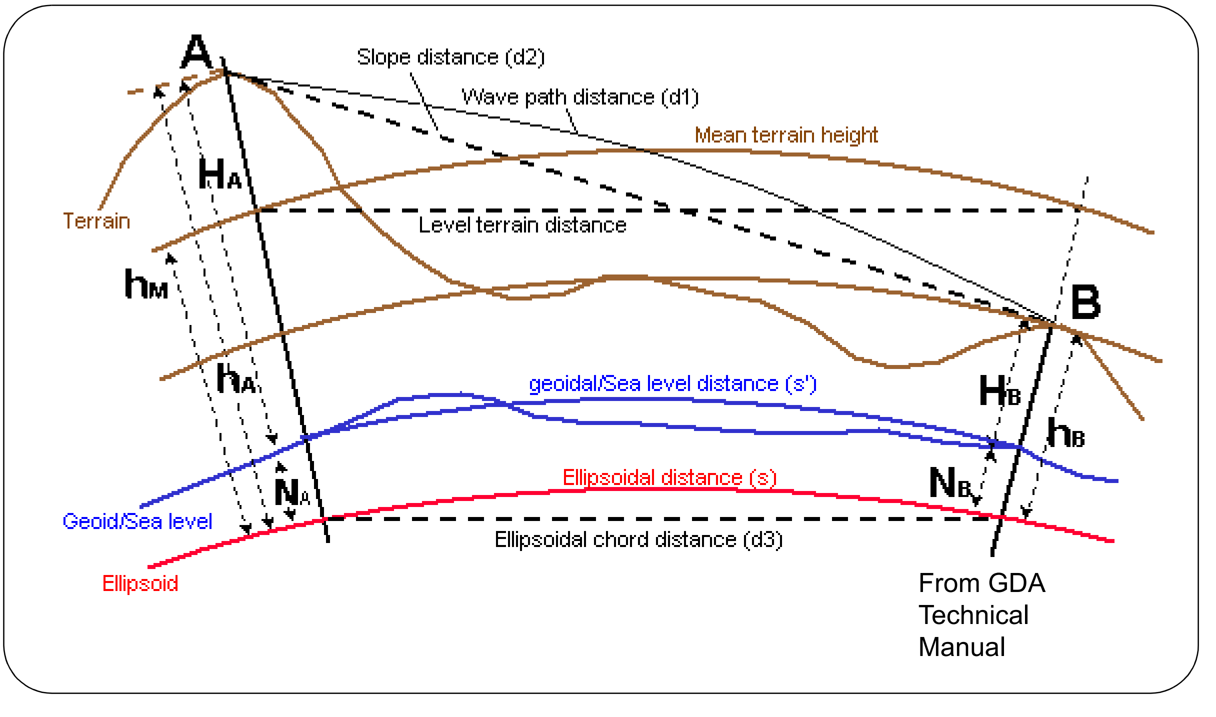
R = Radius of the earth in the azimuth of the line.

Note: An error of 60 meters in the value of $\mathbf{h_M}$ will introduce an error of 10 ppm in the reduced ellipsoid distance. With the introduction of the Australian Geocentric datums, the N values have increased markedly over Australia. So even if a survey is undertaken at sea level (Geoid height approximately 0.0), the ellipsoid heights may very well be greater than 60 meters. As The Ellipsoid height = Geoid height + N value, N values should be considered when reducing measured distances to these datums.

For older Australian ellipsoids, Mean seal level approximated the surface of the ellipsoid (i.e. N value approximately 0.0), so corrections could use geoidal heights to bring the distances down onto the reference surface.

The ellipsoid distance can be calculated using the calculated height factor:

Ellipsoid Distance = Reduced Horizontal Distance x height scale factor.



Continue to the next section 38.4.2 Bearings and Distances - Plane and Grid or return to 38.4 Terminology or 38 Geodetics Summary.

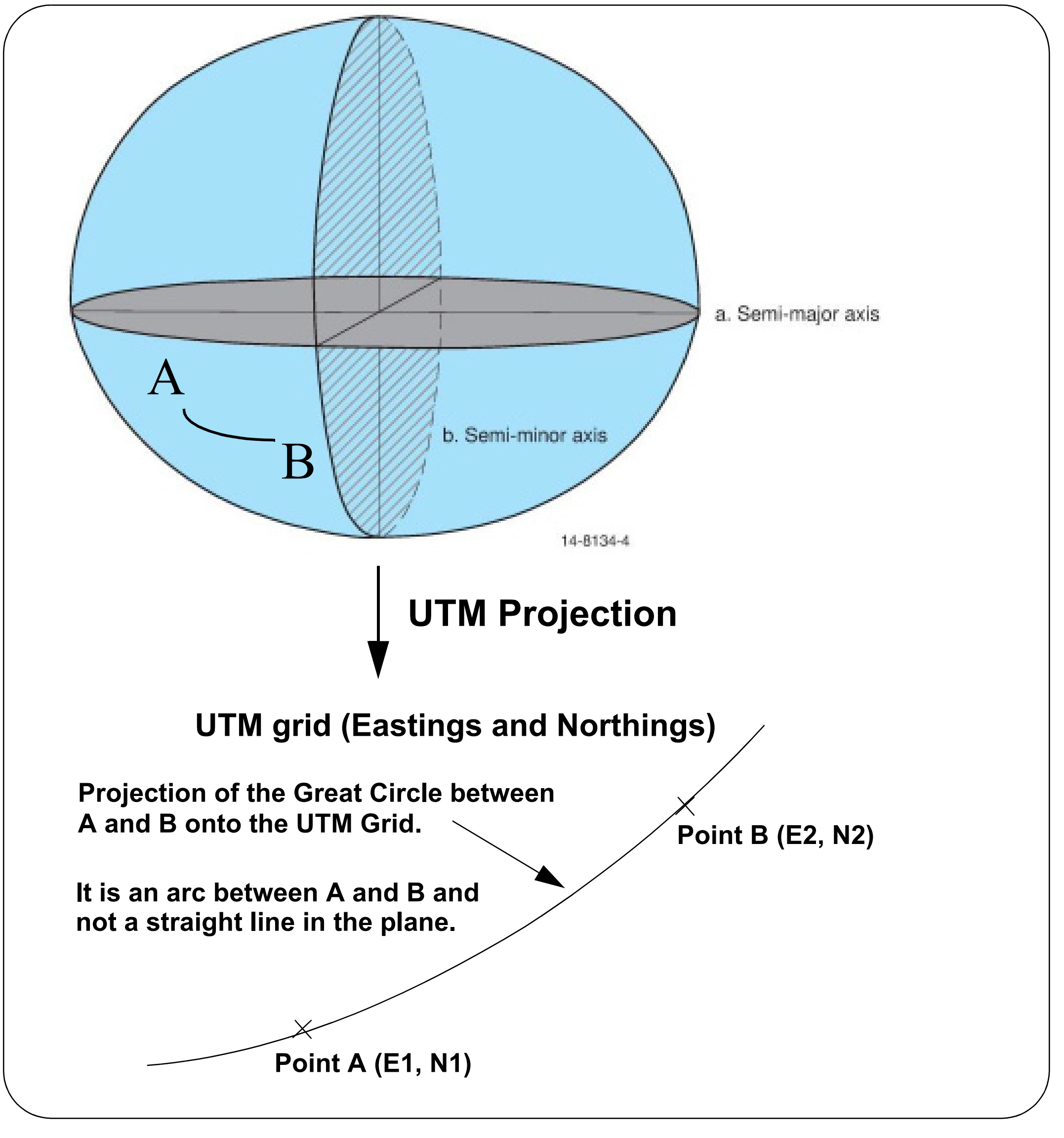
38.4.2 Bearings and Distances - Plane and Grid

If you have a point on an ellipsoid and a Transverse Mercator projection (such as UTM), the projection co-ordinates are known as the grid co-ordinates of the point (Easting and Northing).

On an *ellipsoid*, the ***straight line*** joining two points is the ***Great circle*** between the two points. If each point along the Great circle is projected onto the grid, the path that is traced out is an *arc* between the points.

That is, the *straight lines* on an ellipsoid project onto the TM grid as arcs.

In the diagram below, the Great circle arc through points A and B on the ellipse projects onto the arc shown through Points A and B in grid co-ordinates.



So the *straight line* joining the two points in the **plane** *is different* from the projection of the great circle (straight line on the ellipse) joining the two points on the ellipse.

Continue to the next section 38.4.3 Plane Bearing, Projection Bearing or return to 38.4 Terminology or 38 Geodetics Summary.

38.4.3 Plane Bearing, Projection Bearing

If a straight line is drawn in the plane between two points on a grid, the angle between grid north and this line in equal to the ***plane bearing***.

In other words, if the two point's coordinates are known, standard plane trigonometry can be used to calculate the bearing of the line.

tan (plane bearing) = (E2 - E1) / (N2 - N1)

Projection bearing and *plane* bearing is used interchangeably in ***12d Model***.

Continue to the next section 38.4.4 Australian Grid Bearing, Ellipsoid Bearing or return to 38.4 Terminology or 38 Geodetics Summary.

38.4.4 Australian Grid Bearing, Ellipsoid Bearing

In Australia, the ***grid bearing*** is the angle between grid north and the *tangent* to arc at a point on the arc. e.g Point A. The grid bearing at point A is **not** equal to the reverse grid bearing from point B.

This term is also known as the **ellipsoid bearing** in some countries.

Warning: In some countries, including New Zealand and the US, the term **Grid bearing** is used for the previously defined term **Plane bearing**. However, as defined in the Australian GDA technical manual, the definition of Grid bearing is different to that of a plane bearing.

To avoid confusion, we will use the term ***ellipsoid bearing***.

Continue to the next section 38.4.5 Plane Distance or return to 38.4 Terminology or 38 Geodetics Summary.

38.4.5 Plane Distance

The ***plane distance*** is the length in the plane of the straight line joining two points on a grid - that is, the standard distance between two points in a plane

plane distance = square root [(E2 - E1)*(E2 - E1) + (N2 - N1)*(N2 - N1)]

Continue to the next section 38.4.6 Australian Grid Distance or return to 38.4 Terminology or 38 Geodetics Summary.

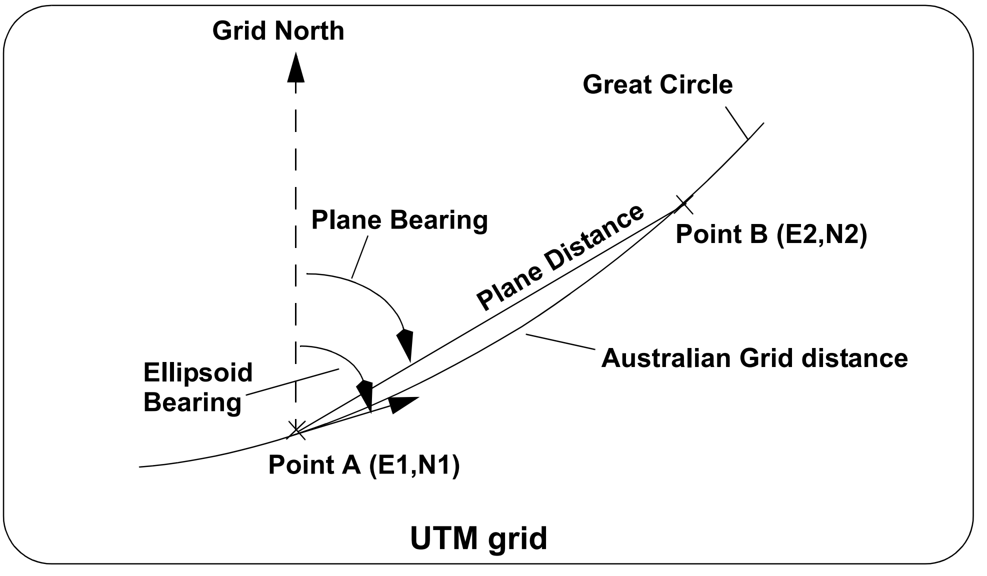
38.4.6 Australian Grid Distance

On an ellipsoid, the *straight line* joining two points on the ellipsoid is the Great circle between the two points. The *straight lines* on an ellipsoid project onto the UTM grid as arcs. In the diagram below, the arc shown through points A and B is the projection of the Great circle arc through points A and B on the ellipse.

In Australia, the ***grid distance*** is the distance on this *arc* from point A to B.

The difference between the plane distance and the grid distance is usually negligible.

Warning: In some countries, the term *grid distance* is used for the definition of plane distance here.



Continue to the next section 38.4.7 Datum or return to 38.4 Terminology or 38 Geodetics Summary.

38.4.7 Datum

Within the Geodetic sections of 12d Model, the term ***datum*** relates to the reference ellipsoid adopted by countries/organisations for mapping projects.

For example AGD is the Australian Geodetic Datum, using the ANS ellipsoid parameters. GDA refers to the Geodetic Datum of australia, using the GRS80 ellipsoid as the basis for defining geodetic coordinates.

In New Zealand, the NZGD49 datum refers to the NZ Geodetic 49 ellipsoid. The NZGD2000 datum is the New Zealand geodetic datum which again refers to the GRS80 ellipsoid.

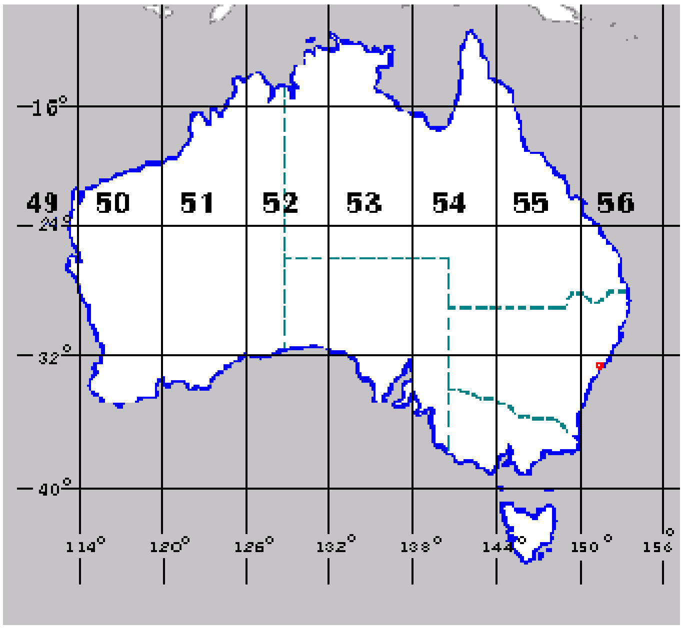
Continue to the next section 38.4.8 Zone or return to 38.4 Terminology or 38 Geodetics Summary.

38.4.8 Zone

The meaning of ***zone*** is specific to the UTM projection type which is commonly used around the world as a mapping projection. This includes AMG and MGA in Australia.

The Universal Transverse Mercator projection splits the world into 60 zones of 6 degrees of longitude. The zone numbering starts at 180 degrees West, longitude. Each zone has a specific central meridian and range of longitude that defines it. As such, by supplying a zone number, a number of parameters about a projection can be deduced.

For example, Australia is covered by the UTM zones 49 to 56.



Continue to the next section 38.4.9 Arc-to-Chord Correction (t-T correction) or return to 38.4 Terminology or 38 Geodetics Summary.

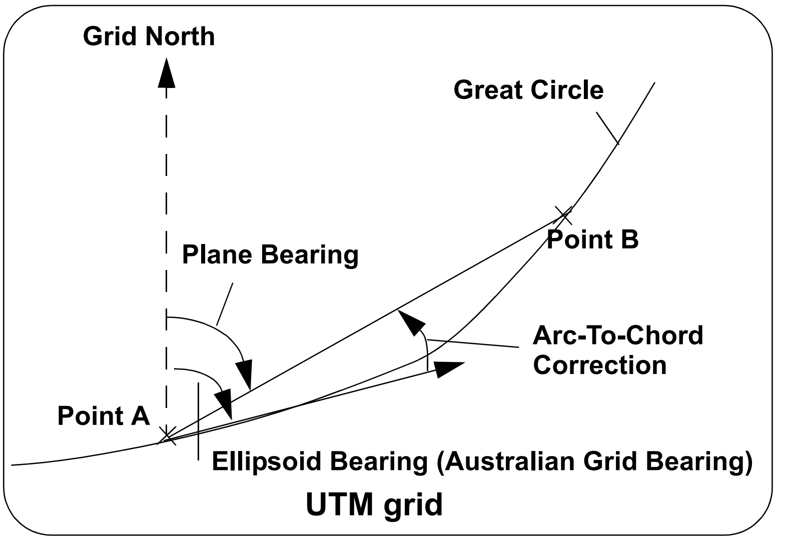
38.4.9 Arc-to-Chord Correction (t-T correction)

The ***arc-to-chord*** correction is the quantity to be added algebraically to a ellipsoid bearing (Australian grid bearing) to obtain a plane bearing.

Plane Bearing = Ellipsoid Bearing + Arc-To-Chord Correction

Plane Bearing = Australian Grid Bearing + Arc-To-Chord Correction

This correction is only really applicable for lines over 10 km but it is included in calculations for completeness. The correction shown in the example below is negative in sign but it can also be positive.



Continue to the next section 38.4.10 Point Scale Factor or return to 38.4 Terminology or 38 Geodetics Summary.

38.4.10 Point Scale Factor

The ***point scale factor*** is the ratio of an infinitesimal plan distance at a point on a grid to a corresponding ellipsoid distance.

It can be used as an approximation to convert (factor) measured ellipsoid distances to plane distances.

Continue to the next section 38.4.11 Line Scale Factor or return to 38.4 Terminology or 38 Geodetics Summary.

38.4.11 Line Scale Factor

The ***line scale factor*** is the ratio of the plane distance on a grid to a corresponding ellipsoid distance. i.e.

$$LineScaleFactor = \frac{PlanDist}{EllipsoidDist}$$

This factor can be used to calculate either the plane or ellipsoid distance given the Line Scale Factor and the other distance. i.e.

Plane distance = Ellipsoid distance x Line scale factor,

Similarly,

Ellipsoid distance = Plane distance / Line scale factor.

Continue to the next section 38.4.12 Combined Point Scale Factor or return to 38.4 Terminology or 38 Geodetics Summary.

38.4.12 Combined Point Scale Factor

The ***combined point scale factor*** is the product of the point scale factor and the height scale factor. i.e.

Combined Scale Factor = Point Scale Factor x Height Scale Factor.

Continue to the next section 38.4.13 Azimuth and Convergence or return to 38.4 Terminology or 38 Geodetics Summary.

38.4.13 Azimuth and Convergence

Azimuth is the horizontal angle measured from an *ellipsoidal meridian clockwise from north* and the great circle between measured points.

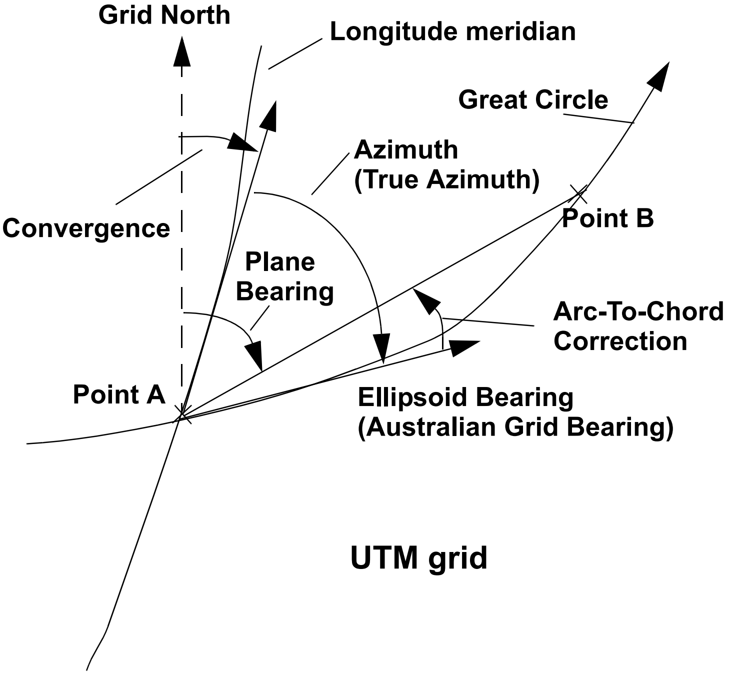
It is also known as the ***True Azimuth***. In general, this value will be calculated internally in 12d Model.

Convergence or ***Grid Convergence*** is the angular quantity to be added algebraically to an *Azimuth* to obtain an ellipsoid bearing (Australian grid bearing). i.e.

Ellipsoid Bearing = Azimuth + Grid Convergence

Australian Grid Bearing = Azimuth + Grid Convergence

Warning: In some countries, the *Grid Convergence* has the opposite sign.



Combining

Ellipsoid Bearing = Azimuth + Grid Convergence

Australian Grid Bearing = Azimuth + Grid Convergence

and

Plane Bearing = Ellipsoid Bearing + Arc-To-Chord Correction

Plane Bearing = Australian Grid Bearing + Arc-To-Chord Correction

produces the equation

Plane Bearing = Azimuth + Grid Convergence + Arc-To-Chord Correction

or

Azimuth = Plane Bearing - Grid Convergence - Arc-To-Chord Correction

True Azimuth = Plane Bearing - Grid Convergence - Arc-To-Chord Correction

Warning: In some countries, the *Grid Convergence* has the opposite sign.

Please continue to the next section 38.5 Converting Between AMG, ISG and MGA or return to 38.4 Terminology or 38 Geodetics Summary.

38.5 Converting Between AMG, ISG and MGA

In Australia the ellipsoid used to represent the shape of the earth used to be defined by AGD 84 but it has now has been changed to an ellipsoid that has its centre at the centre of mass of the earth (geocentric - GDA 94). See 38.1 Shape Of The Earth.

This is to fit in with GPS because satellites orbits are centred on the centre of mass.

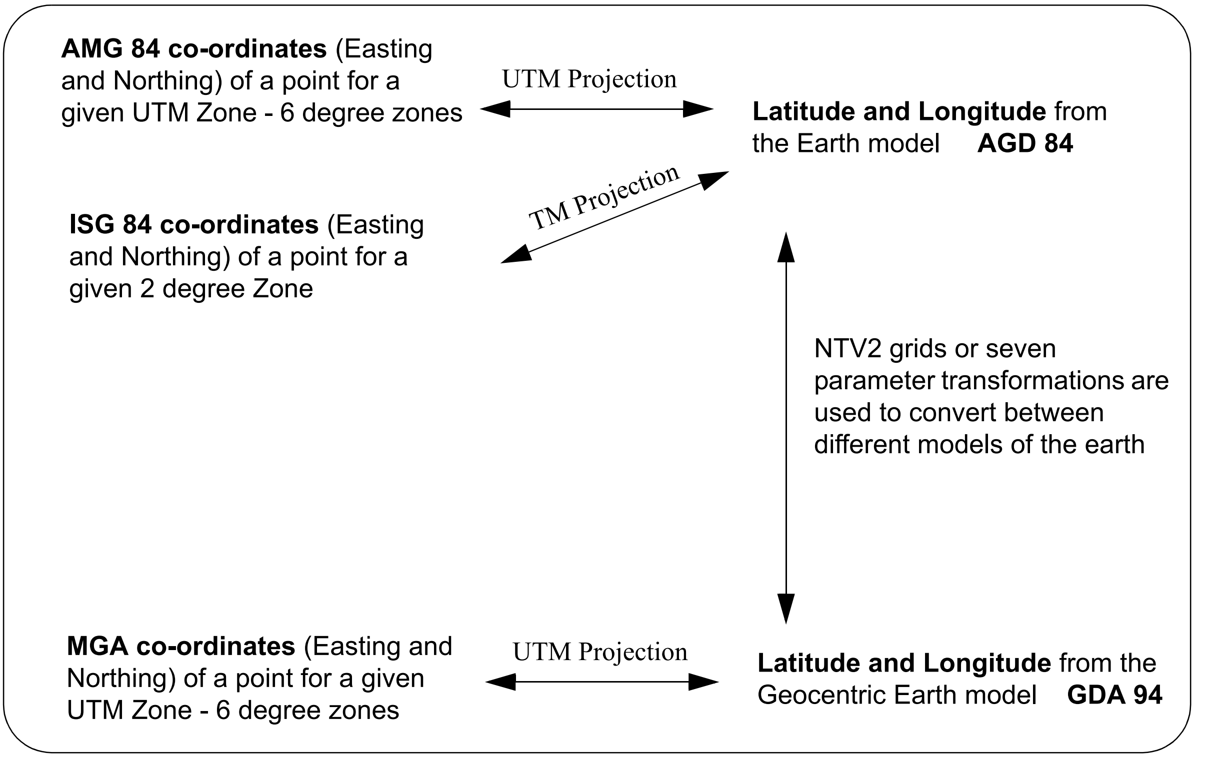
This means that the Latitude and Longitude of every point has changed.

The use of UTM projections (Universal Transverse Mercator - see 38.3 Map (Cartographic) Projections) and Zones still apply but because the latitude and longitude of each point has change, its Easting and Northing for a UTM zone has also changed.

AMG 84 Zones are the co-ordinates for the UTM Projections defined for six degree zones using the ellipsoid defined by AGD 84.

MGA 94 Zones are the co-ordinates for the UTM Projections defined for six degree zones using ellipsoid define by GDA 94.

Note - ISG 84 Zones are the co-ordinates for the TM Projections defined for two degree zones using the ellipsoid defined by AGD 84.



For converting between the ***different datums*** AGD66/84 and GDA94 (i.e. AMG, ISG, Long Lat <-> MGA, Long Lat), use the option

Survey =>Conversions =>AGD66/84 <-> GDA94 (see 17.10.4 AGD66/84 <---> GDA94

For converting between different AMG Zones (AMG <->AMG) or different MGA Zones (MGA<->MGA) use the option

Survey =>Conversions =>Australian conversions. See 17.10.1 Australian Conversions.

For converting between different ISG/AMG Zones (AMG,ISG <->AMG, ISG)

Survey =>Conversions =>Cartographic. See 17.10.2 Cartographic.

Return to38 Geodetics Summary.

39 Setting Up & Configuring 12d

This appendix contains information about files used for setting up and configuring ***12d Model***.

See

39.1 Folder Structure Installed by 12d Model

12d Model has Set Up files to define how new projects are created, define colours, linestyles, symbols etc. and control many other aspects of ***12d Model***.

12d Model comes with defaults for these things but most users wish to customise many of the settings, share them between groups of users across a network etc.

To make customisation easier, all the ***12d Model*** customisation files (called Set Up files) have default names and on installation, are stored in a folder called ***Set_ups***.

To customise ***12d Model***, users only need to which file to modify for that particular customisation.

When *12d Model 11* is installed on a *Windows* computer, the major components are installed in the Microsoft preferred installation folder **Program Files,** in the sub folders **12d\12dmodel\11.00.**

That is, the software and associated files are installed in **Program Files\12d\12dmodel\11.00**

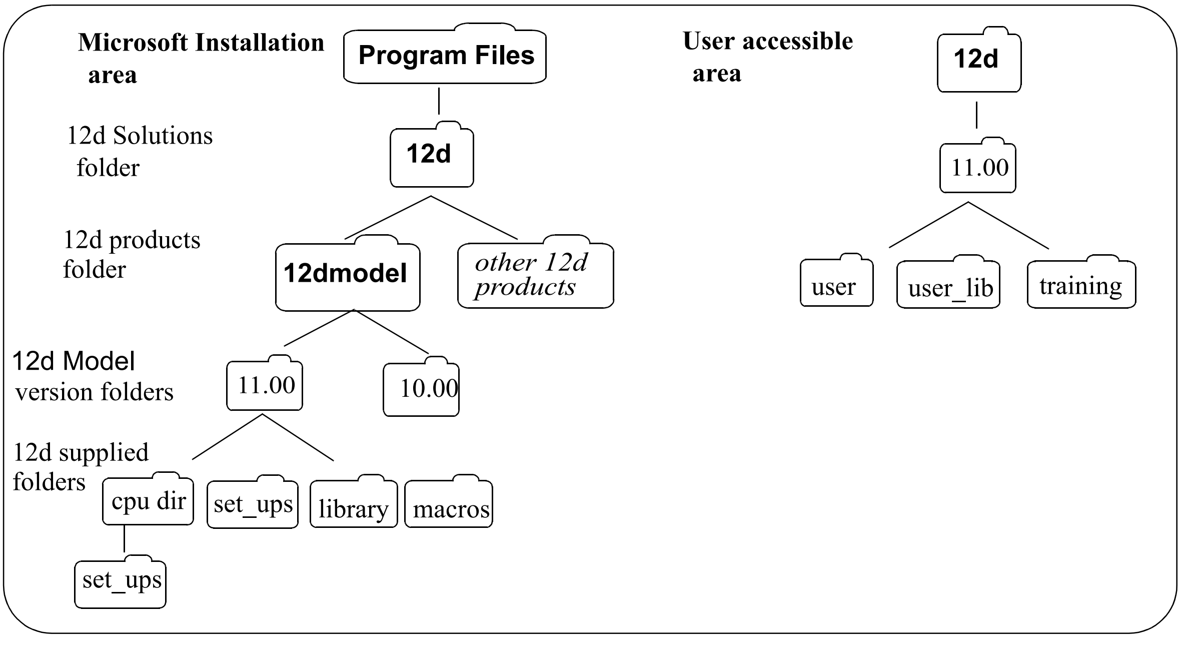
However, without *Administrator* privilege, users do no have write access to **Program Files.**

To allow non-administrator users access to Set Up areas that users can store modified files, *12d Model 11* also installs extra folders in an area that could be accessed by the user. The default folder structure for this area is **C:\12d\11.00**

So the customisation system allows the use of both the areas
Program Files\12d\12dmodel\11.00

and

C:\12d\11.00 for storing Set Up and other files necessary for the tailoring and running of ***12d Model***.



Please continue to the next section 39.2 Files for Setting Up 12d.

39.2 Files for Setting Up 12d

When ***12d Model*** fires up, it looks for special files (called ***Setting Up*** or ***Setup*** files) to define many of its features. Hence the *Setup* files can be used to customize ***12d Model*** for a site, a client or even a user.

When ***12d Model*** creates a new project, or changes to another project, then for each Setting Up file

EITHER

(a) an environment variable for the Setup file exists, and the value of the environment variable is taken as the full path name of the file to be used as that Setup file.

OR if the environment variable does not exist for the Setup file

(b) a file with the default Setup file name is searches for in specially named folders, in a fully defined order.

The search for ***each Setup file*** is independent of any other Setup file. That is, for each Setup file, the folders are searched in the specific order until that file is first found and the searching stops for that Setup file. and then the search is begun again for the next Setting Up file.

Consequently the Setup files do not all have to be in the same folder. This greatly increases the tailorability of ***12d Model***.

The description, default name and any environment variable for each Setup file are given in the sections 39.2.1 Setup Files Only Used for a New Project, 39.2.2 Setup Files Only Used for an Existing Project and 39.2.3 Setup Files Used for New and Existing Projects.

The folders searched for Setup files, and the folder search order, is described in the section 39.2.4 Folders Searched for Setup Files.

See

39.2.1 Setup Files Only Used for a New Project

The two Setup files, **setups.4d** and **defaults.4d**, are only used when creating a **NEW *12d Model*** project, and after the project is created, the information supplied in these two Setup files is ***stored within the 12d Model project*** and is then modified from within ***12d Model*** when in that project.

Similarly the file **macros.4d** is a file of macros or chains (one per line) that are only run when a **new** project is created (**project_macros.4d** is a run when an **existing** project is opened).

defaults.4d

define the defaults for a **new** project.

Modified by the **Defaults** panel from the **Projects =>Management =>Defaults** option.

Once a project is created, the defaults can be modified inside 12d Model.
The defaults are then saved with the project.

The environment variable is DEFAULTS_4D.

For more information go to the section 39.2.7.3 Defaults File (defaults.4d)

setups.4d

define the layout of views on the screen for a **new** project.

Once a project is created, the view layout can be modified inside ***12d Model***.

The view layout is then saved with the project.

The environment variable is SETUPS_FILE_4D.

For more information go to the section 39.2.7.1 Set Ups File (setups.4d)

macros.4d

a file of macros and/or chains (one per line) that are run when a **new** project is created (see 5.9 Running Macros and Chains on Start Up).

The environment variable is RUN_MACROS_FILE_4D.

39.2.2 Setup Files Only Used for an Existing Project

The file **project_macros.4d** is a file of macros or chains (one per line) that are only run when an **existing** project is opened. So project_macros.4d is modified between accesses to the project then the modified version of the file is used for the project.

project_macros.4d

a file of macros (one per line) that are run when an **existing** project is opened (see 5.9 Running Macros and Chains on Start Up).

The environment variable is RUN_PROJECT_MACROS_FILE_4D.

39.2.3 Setup Files Used for New and Existing Projects

The other *Setting Up* files are searched for ***each time a project is created or opened***. So if these files are modified between accesses to the project then the modified versions of the files are used for the project.

Some of the Setup files that ***12d Model*** searches for each time a project is opened are:

colours.4d // sets the colour names and red, green, blue mix; also defines the default pens that the colours map to (as displayed in the panel **colours to pens**). See 39.2.7.2 Colours File (colours.4d).

The environment variable is COLOURS_4D.

digitize.4d // list and defines the digitizers available in 12d Model. See 8.5.7 Digitizer Definitions File.

The environment variable is DIGITIZERS_4D.

env.4d // contains values for environment variables See 39.4 Environment Variables.

The environment variable is ENVIRONMENT_4D.

fonts.4d // defines the fonts used in text styles. See 40.3 Textstyles and Fonts.

The environment variable is FONTS_4D.

gui.4d // define the fonts and colours used in panels and menus, spacing between items in panels and menus and the maximum pop-up length. See 39.2.7.6 GUI.

The environment variable is GUI_4D.

layout.4d // a file containing the screen layout file information (slx/slf) for placing menus or panels on the screen. The layout file is used every time a project is opened. The layout file can have most menus and panels in it. See 42.2.3 Screen Layout File in the appendix 42 Special File Formats.

The environment variable is LAYOUT_FILE_4D.

linestyl.4d // defines the line styles (linestyles, worldstyles, groupstyles and twoptstyles). See 40.1 Line Styles.

The environment variable is LINESTYLES_4D.

names.4d // a special name mapping file which is used to define for given string names, information such as colours, model etc. This is used in some panels to fill out other panels fields after entering the string names in the panel.

The environment variable is NAME_MAPPINGS_4D.

plotters.4d // defines the plotters that appear in the pop-up for the plotter type panel field. See 43.2 Defining Plotters - Plotters.4d.

The environment variable is PLOTTERS_4D.

pmf.4d // plotter mapping file for defining mapping of 12d Model colours to pens with thicknesses, and the rgb for the pens See 43.3.2.1 Definition of a Plotter Mapping File

The environment variable is PLOTTER_MAPPING_4D.

sheets.4d // gives sheet sizes and names used in the pop-up for the sheet size panel field. See 39.2.7.4 Sheet Sizes File (sheets.4d)

The environment variable is SHEET_SIZES_4D.

survey.4d // lists and defines the data collectors available in 12d Model See 7.9.3 Survey Data Collectors

The environment variable is DATA_COLLECTORS_4D.

symbols.4d // defines the symbols. See 40.2 Symbols

The environment variable is SYMBOLS_4D.

textstyl.4d // defines the text styles. See 40.3 Textstyles and Fonts.

The environment variable is TEXTSTYLES_4D.

textstyle_names.4d // defines the textstyle favourites

The environment variable is TEXTSTYLE_MAPPINGS_4D.

toolbars.4d // defines the toolbars See 41.3 User Defined Toolbars

The environment variable is TOOLBARS_4D.

userkeys.4d // defines the action of function keys. See 41.1 User Defined Function Keys

The environment variable is FUNCTION_KEYS_4D.

usermenu.4d // defines the user defined walk-right menus for User on 12d Model menus. See 41.2 User Defined Menus.

The environment variable is USER_OPTIONS_4D.

xtramenu.4d // defines the 12d Solutions defined walk-right menus for User on 12d Model menus. See 41.2 User Defined Menus.

The environment variable is EXTRA_OPTIONS_4D.

The folders searched for Setting Up files, and the folder search order, is described in the next section 39.2.4 Folders Searched for Setup Files.

39.2.4 Folders Searched for Setup Files

When ***12d Model 11*** is installed on a *Windows* computer the major components installed by ***12d Solutions*** are under the Microsoft preferred installation folder **Program Files,** in the sub folders **12d\12dmodel\11.00.** That is **Program Files\12d\12dmodel\11.00.**

Without Administrator privilege, users do no have access to this area.

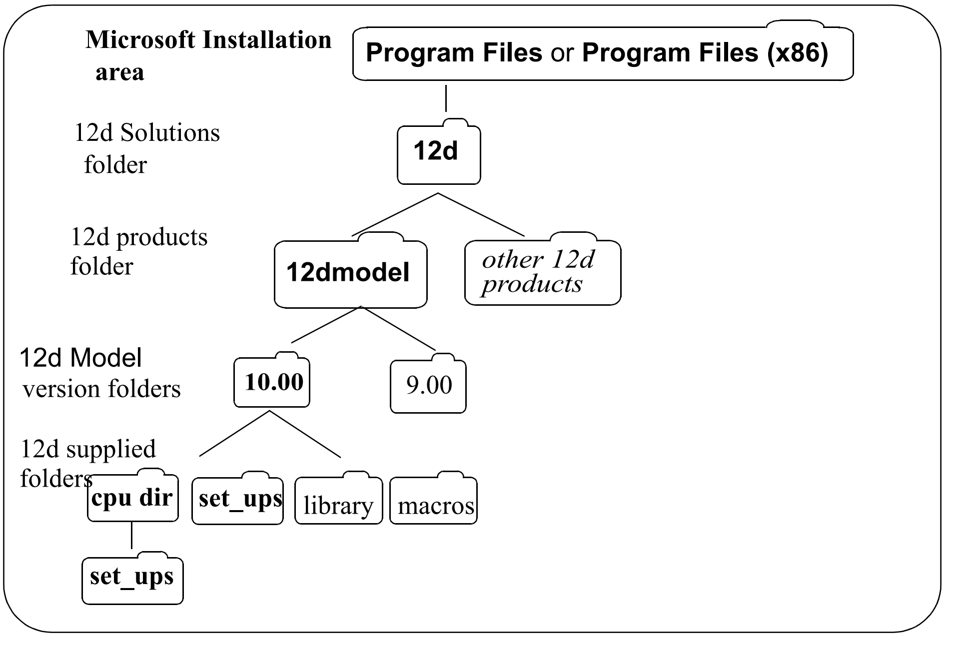
Note: these are the folders used for 32-bit ***12d Model*** on a 32-bit Windows and 64-bit ***12d Model*** on a 64-bit Windows. For a 32-bit ***12d Model*** on a 64-bit Windows, the software is installed in **Program Files (X86)\12d\12dmodel\11.00**

Other information that can be accessed and modified by the user, will be installed in the sub folders **C:\12d\11.00** in an area that the user has access to.

The structure of the folders containing Setting Ups files are **Set_ups** and **User**. How the various folders are searched for Set Up files will be described in the following sections.

set_ups

The ***12d Model*** installation creates a number of specific folders such as the operating system-cpu specific folder (for example **nt.x86** for 32-bit ***12d Model*** on Intel chips and **nt.x64** for 64-bit ***12d Model*** on Intel chips), and areas for ***set up files:*** namely an o/s-cpu specific **set_ups** under the o/s-specific folder and a **set_ups** under **Program files\12d\12dmodel\11.00** (or **Program files (X86)\12d\12dmodel\11.00** for 32-bit ***12d Model*** on 64-bit Windows).



It is also possible to move the set_ups folder **Program files\12d\12dmodel\11.00\set_ups** to another location, and even give it another name, by defining an environment variable SET_UPS_4D which gives the path of the moved folder

SET_UPS_4D folder // 12d supplied folder of Setting Up files

User and Customer_User

Users usually want to customise ***12d Model*** by modify one or more of the Setup files but most users do not have written access to the ***Program Files*** folder, and **Program files\12d\12dmodel\11.00\set_ups**. Also, any files in the **set_ups** folders may be overwritten by future *12d Model* installations.

To overcome these two problem, there are two special folders (reserved for customers) that are searched ***before*** **set_ups** for any Setting Up files.

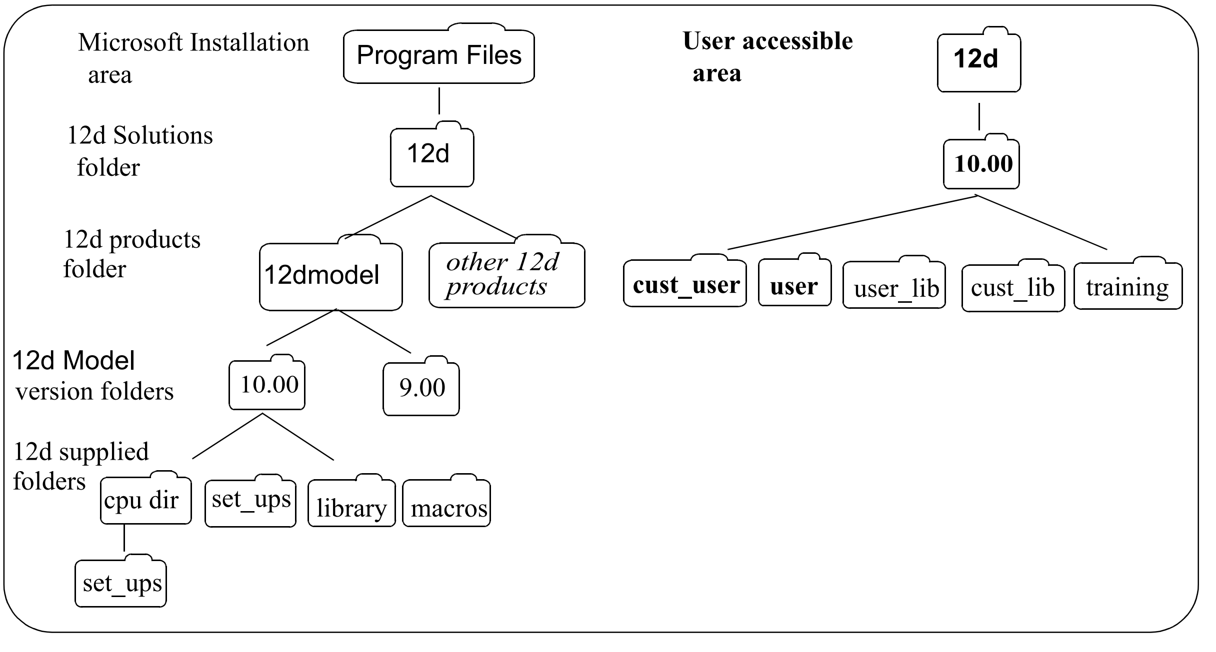
One folder is only looked for when it has been created by a user and is pointed to by the environment variable CUSTOMER_USER_4D. It has no default name but is whatever name has been set up for it. We will call it as **cust_user.**

The other folder has the default name **user**.

The folders **cust_user** and **user** should ***NOT*** be under ***Program Files*** but in **12d\11.0** which the ***12d Model*** installer usually creates in a folder that the ***12d Model*** user has read/write access to. The default folder to contain **user** and **cust_user** is **C:\12d\11.0**

During the ***12d Model*** installation, if **user** does not exist, an empty **user** folder is created.

The folder **cust_user** is not created and if required, must be created by the user and the environment variable CUSTOMER_USER_4D set up to point to it.



To find a Setting Up file, the folder **cust_user** will be searched for before **user**, and **user** is searched for before **set_Ups**.

Consequently any modified Setting Up files should be placed in **cust_user** or **user** and then they will be found before the file of the same name in **set_ups** (**user** may contain o/s-cpu specific folders).

When a new ***12d Model*** version is issued (*e.g.* 11), the contents of **cust_user** and **user** from the previous version of ***12d Model*** (*e.g.* V10) need to be copied over to the new **cust_user** and **user** area. This is **NOT** done automatically during the installation in case there are files that need to be modified for the newer version.

There is also an ***environment variable*** USER_4D which can be set to point to another user defined folder which will be searched for Setting Up files ***before*** the default **user** is searched.

USER_4D folder // users own folder for Setting Up info

This is a common method for having all users on a network share the same Setup files. The common files are placed on a disk drive accessible by all the users and the environment variable USER_4D defined to point to that folder.

Note - both **user** and the folder pointed to by USER_4D can exist.

Current and Home

The folder that a project is in (the **current** folder) and the users **home** folder are also searched for Setup files.

39.2.5 Searching Order for Setup Files

When ***12d Model*** creates a new project, or changes to another project, it searches folders for the Setup files in a well defined order.

Each Setup file has a default name, and unless there is an environment variable for that specific Setup file defining its exact name and path, folders are searched in a specific order until the file is found. The search is begun afresh for each Setup file so different Setup files may come from different folders.

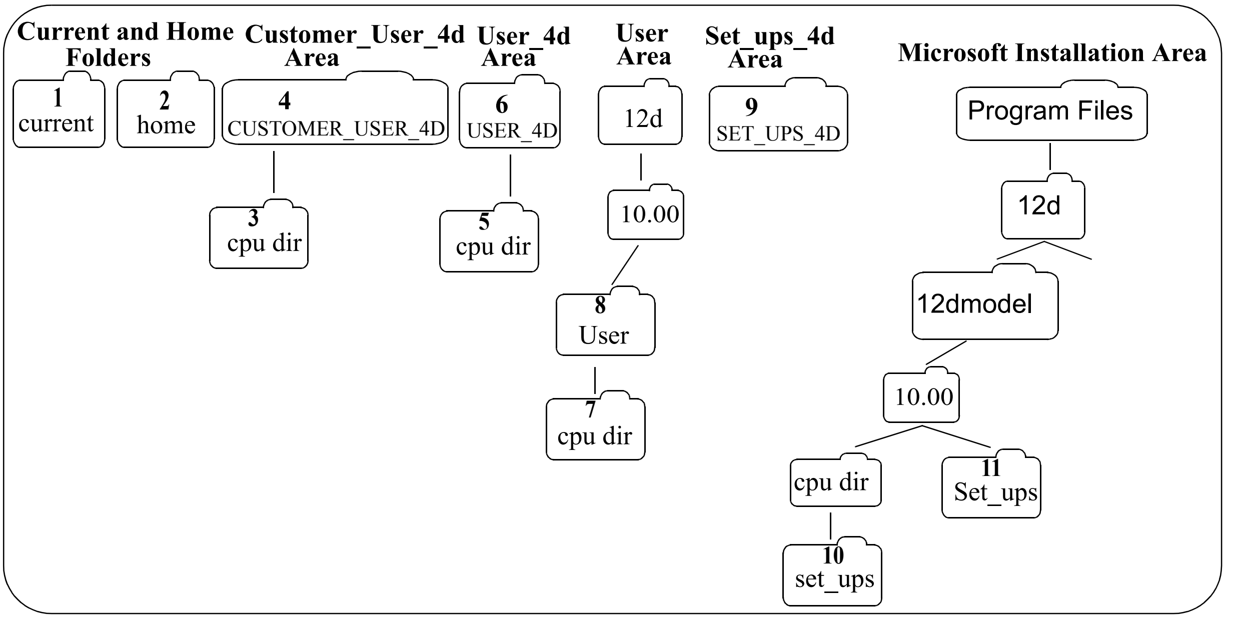
The fixed search order for any Setup files is

1. check for the appropriate environment variable defining the full path to the ***specific*** Setup file.

If the environment variable is not defined, or the file pointed to does not exist, then the folders are searched for in the following order for the specific Setting Up file with its default name (e.g. colours.4d)

2. the projects current folder
3. the users home folder
4. specific cpu folder in CUSTOMER_USER_4D
5. file folder defined by CUSTOMER_USER_4D
6. specific cpu folder in USER_4D e.g. nt.x86 for Intel
7. file folder defined by USER_4D
8. specific cpu folder in **User** under **C:\12d\11.00**
9. **User** under **C:\12d\11.00**
10. the folder defined by SET_UPS_4D
11. **set_ups** folder under the specific cpu folder under **Program Files\12d\12dmodel\11.0**
12. **set_ups** folder under **Program Files\12d\12dmodel\11.00**

Once the appropriate Setup file is found, the search is terminated and that, and only that file, is used.



Notes

(a) When an **existing** project is opened by *12d Model*, the Setting Up files ***setups.4d***,

defaults.4d and ***macros.4d*** are ***not*** used. These files are only used for defining a NEW project.

(b) **project_macros.4d** is only used when opening an EXISTING project and not a new project.

(c) A Setup file such as ***colours.4d*** may exist in one or more of the above folder. However, only the ***first*** occurrence of the file found when using the above search order, is used.

Please continue to the next section 39.2.6 Writing Out Setup Files.

39.2.6 Writing Out Setup Files

Most of the ***Setup*** files can be created and edited within ***12d Model*** using panels such as **Projects =>Management =>Defaults.**

When the **Write** button is selected on these panels, a **Write Setup File** panel comes up to specify where the file is to be written out to.

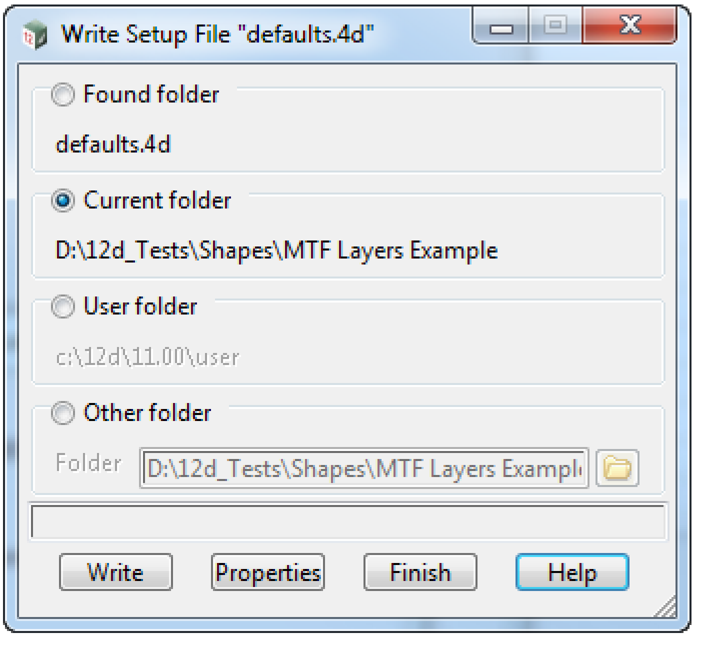
The choices on the panel allow the file to be written out to:

Found folder - the folder where the file currently being used by ***12d Model*** resides. This will be unavailable (greyed out) if the user doesn't have access to the folder.

Current folder - the folder where the project currently being used by ***12d Model*** resides. This will be unavailable (greyed out) if the user doesn't have access to the folder.

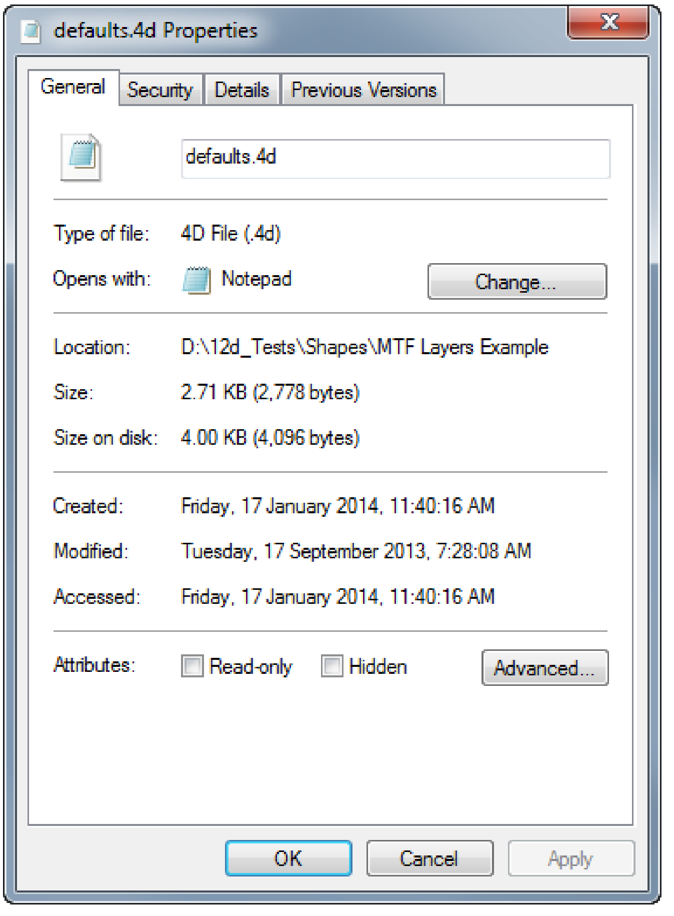
User folder - the **user** folder. This will be unavailable (greyed out) if the user doesn't have access to the folder.

Other folder - any folder can be selected.



A project restart is required for the new file to take effect

If you are having problems writing to the file, click on the **Properties** button to bring up the **Properties** panel for the selected file (if it exists) and check the *Security* for the file.



Please continue to the next section 39.3 Library, User Library, Customer Library.

39.2.7 Some Special Setup Files

See

39.2.7.1 Set Ups File (setups.4d)
39.2.7.2 Colours File (colours.4d)
39.2.7.3 Defaults File (defaults.4d)
39.2.7.4 Sheet Sizes File (sheets.4d)
39.2.7.5 Transitions and Spirals File

39.2.7.1 Set Ups File (setups.4d)

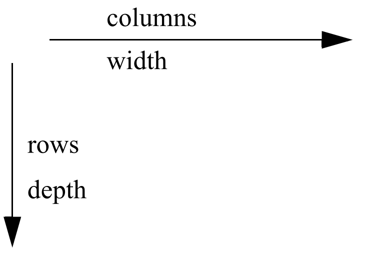
The St Up file, **setups.4d**, is used to define the initial screen set-ups. It can be used for

- the system font used for text
- the size of the initial window
- the position of the main menu, header menu, function menu, function recalc menu, snaps menu, volumes menu, geometry menu, sewer menu, pipeline menu
- the position of the screen message box, xyz message box
- the position of the **save project** panel (now redundant)
- the position, size, name and viewing parameters for the initial views

The co-ordinate system used for specifying the position of the left hand top corner of items on the screen is by column and row. For items which include an area of the screen (such as the window and views), a width and depth are also specified.

The column value is measured from the left hand size of the screen and the row value is measured from the top of the screen.

The units for row, column, width and depth are screen pixels.



Note - If any views are defined in the set_up_file, then the position of the **screen message box** must be also be defined and it must be **before** the definition of any view.

The format of the commands in the file setups.4d is

WINDOW *column_value* *row_value* *width* *depth* // main window

MAIN MENU *column_value* *row_value*

FUNCTION MENU *column_value* *row_value*
FUNCTION RECALC MENU *column_value* *row_value*
GEOMETRY MENU *column_value* *row_value*
PIPELINE MENU *column_value* *row_value*
SEWER MENU *column_value* *row_value*
SURVEY MENU *column_value* *row_value*
VOLUMES MENU *column_value* *row_value*

SNAPS MENU *column_value* *row_value*
SNAPS BUTTONS HORIZONTAL *column_value* *row_value*
SNAPS BUTTONS VERTICAL *column_value* *row_value*

PROJECT SAVE PANEL *column_value* *row_value*

// Views

PLAN VIEW *column_value* *row_value* *width* *depth* *name*

```
SECTION VIEW            column_value     row_value   width     depth   name

PERSPECTIVE VIEW        column_value     row_value   width   depth     name
  reference point       0.0         750.0       120.0       // x  y   z
  view plane normal     1.0           0.0         0.0       // dx  dy  dz
  view up vector        0.0           0.0         1.0       // dx  dy  dz
  view distance         10.0                                // distance

PERSPECTIVE OPENGL VIEW   column_value     row_value   width  depth  name
  reference point       0.0         750.0       120.0       // x  y   z
  view plane normal     1.0           0.0         0.0       // dx  dy  dz
  view up vector        0.0           0.0         1.0       // dx  dy  dz
  view distance         10.0                                // distance
```

The following parameters were used for V5.00 but are ignored for V5.0 onward:

```
SCREEN MESSAGE BOX          column_value          row_value
XYZ MESSAGE BOX             column_value          row_value
```

Continue to the next section 39.2.7.2 Colours File (colours.4d) or return to 39.2.7 Some Special Setup Files.

39.2.7.2 Colours File (colours.4d)

12d Model allows the use of up to 10,240 distinct colours for drawing in any view.

The colours are numbered from 0 to 10,239 and the user can define how the colour appears on the screen by giving the red, green and blue intensities (RGB) for each colour, and what name is used throughout the program when referring to a particular colour number.

The ***colour names*** and ***RGB*** mix associated with each colour number, pop up order etc are defined in the file **colours.4d**.

Each line of the file refers to a particular colour number.

There are nine items on each line, each separated by one of more spaces, with an optional comment at the end which is preceded by //.

The items in their order on a colour line are:

1. *red intensity value - a value between 0 and 255*
2. *green intensity value - a value between 0 and 255*
3. *blue intensity value - a value between 0 and 255*
4. *plotter pen number - 0 or a positive integer*

   When plotting, the **plotter pen number** is the default pen number that the colour is mapped to. This may be overridden by a plotter mapping file.
5. *colour name - the name that can be used instead of the colour number.*

   The **colour name** used to describe the colour is given in quotes. Colour names can be alphanumeric with a decimal point, although upper and lower case are considered the same. Each colour must have a *unique* colour name. If an underscore "_" is used in a colour name in the *colours.4d* file then a space is inserted in ***12d Model***. The colour name must be enclosed within quotes. For example, "dark red".
6. *colour number - 0 or a positive integer. the actual colour number that the line refers to.*

   The **colour number** is the actual colour number that the line refers to and what is actually stored in ***12d Model***. A colour numbers can only occur once in the *colours.4d* file but there can be gaps in the colour numbers.
7. *colour pop-up number*

   The colour pop-up number is an integer that can be negative, zero or positive. Colour pop-up numbers can only occur once in the *colours.4d* file but there can be gaps in the colour pop-up numbers.

   The *colour pop-up number* is used to decide which colours are should in the Colour box. If **Display colours** is set to ***n*** in ***Project =>Management =>Defaults***, then the colours with the *n* smallest pop-up numbers are displayed in the Colour box pop-up. So the order that the colours are chosen to appear in the colour pop-up is independent of the colour number.
8. *colour group*

   The colour group is simply text and does not have to be unique. The colour group is enclosed in quotes. For example, "vis".

   The colour group is not currently used in ***12d Model*** but in the future will be used to give a tree structure to the colours pop-up.
9. *colour legacy name*

   This is another name that the colour number could be known as. This was introduced to allow upward compatibility when going from the colours.4d supplied in v10 to that supplied in v11.
10. *// comment*

    text that is treated as a comment. It must be preceded by a //.

The *colours.4d* file is displayed and modified by using the **Edit Colours** panel brought up from the

[Edit] item on the Colours pop-up, *Select Colour.* For more information, go to the section 4.22 Colours.

Continue to the next section 39.2.7.3 Defaults File (defaults.4d) or return to 39.2.7 Some Special Setup Files.

39.2.7.3 Defaults File (defaults.4d)

The **defaults** file, **defaults.4d**, defines the initial default settings used in the ***12d Model*** **Default** and the **Systems Defaults** panels.

The **defaults** file can contain:

```
// general defaults

DEFAULT COLOUR                              red
DEFAULT POINT COLOUR                        yellow
DEFAULT TIN COLOUR                          brown
DEFAULT CONTOUR COLOUR                      cyan
DEFAULT CONTOUR INDEX COLOUR                magenta
DEFAULT TEXT SIZE                           8

// view settings

DEFAULT CULLING OFF/ON
DEFAULT CULLING SIZE                        1.0
DEFAULT SECTION VIEW EXAGGERATION           10.0
DEFAULT PERSPECTIVE VIEW EXAGGERATION1.0

DEFAULT FAST TEXT CULL SIZE                 4.5        // if not specified assumes 4.5
DEFAULT NONE TEXT CULL SIZE                 2.0        // if not specified assumes 2.0

// highlighting

DEFAULT ANGLE MODE                          BEARINGS/DEGREES
DEFAULT HIGHLIGHT COLOUR                    white
DEFAULT HIGHLIGHT CROSS COLOUR              yellow
DEFAULT HIGHLIGHT CROSS SIZE                2.0

// drawing points

DEFAULT CHORD ARC TOLERANCE                 0.1
DEFAULT POINT CROSS SIZE MMS                2.0
DEFAULT POINT CROSS SIZE PIXELS             3

// trash model and mode

DEFAULT TRASH MODEL          model_name             // model for trash
DEFAULT TRASH MODE           trash string           // send used strings to trash
                             delete string          // delete used strings
                             keep string            // keep used strings

// miscellaneous

DEFAULT NAME SETTINGS        file_name              // names.4d file
```

The information panel for each editor can be toggled on/off and the initial state when a new edit operation is begun is given by

```
DEFAULT EDIT INFORMATION     1          // show info panel on editor start-up
                             0          // don't show info panel
```

When output report are created, the scripts/programs pointed to by EDITOR_4D and PRINTER_4D will be run depending on the values of the defaults **display reports** and **print reports**.

```
DEFAULT DISPLAY REPORT FILES                1          // run EDITOR_4D
```

	0	// don't run "
DEFAULT PRINT REPORT FILES	1	// run PRINTER_4D
	0	// don't run "

Similarly when plots are created, the script/program pointed to by PLOTTER_4D and will be run depending on the value of the default **send plots**.

DEFAULT SEND PLOT FILES	1	// run PLOTTER_4D
	0	// don't run "

The number of minutes between displays of the **save project** yes-no box is given by

DEFAULT SAVE INTERVAL	minutes	// 0 for never

The file defining string colours can have up to 10,240 colours in it however this number is usually inconvenient to display in the standard colour pop-up so there is a setting to set how many colours are displayed from the list.

DEFAULT POPUP COLOURS	number_of_colours	// default 16

The precision for displaying real numbers in the information panel and in boxes and panel fields can be set.

DEFAULT PRECISION	integer	// info panel - default 3
DEFAULT BOX PRECISION	integer	// boxes & panels -default 4

The sign for cut areas and volumes can be positive or negative (fill is the opposite) and is given by

DEFAULT CUT VOLUME SIGN	-1	// negative for cut (default)
	1	// positive for cut

Text in text string, 4d strings and linestyles may be in pixels and must be given a millimetre size for plotting. Pixel text is multiplied by a factor to convert it to a millimetre size. The pixels to mm plot

DEFAULT PIXELS TO MM PLOT FACTOR	real	// default 1.0

Continue to the next section 39.2.7.4 Sheet Sizes File (sheets.4d) or return to 39.2.7 Some Special Setup Files.

39.2.7.4 Sheet Sizes File (sheets.4d)

For plot frames, long and x plots, the overall size of the plot sheet can be given by a pop-up containing defined sheet size.

The sheet size names, width and heights can be specified by the user in a file named **sheets.4d** which is in the normal Set Up areas, or is pointed to by the environment variable

```
SHEET_SIZES_4D      file        // file of plotter sheets sizes
```

The layout of the sheet sizes file is

```
// User definition file for sheets sizes in 12d Model
// Heights and widths are in mm.

//          sheet name          width       height

            A0                  1189        841
            A1                  841         594
            A2                  594         420
            A3                  420         297
            A4                  297         210

            B1                  1000        707
```

Continue to the next section 39.2.7.5 Transitions and Spirals File or return to 39.2.7 Some Special Setup Files.

39.2.7.5 Transitions and Spirals File

A transition is a means of easing from a straight to a curve (full transition) or from one radius curve to another radius curve (partial transition).

There is often confusion between the words spirals and transition curves come in a variety of formulae and spiral is just one type of transition curve.

See

39.2.7.5.1 Spirals

39.2.7.5.2 Left/Right, Leading/Trailing Transitions

39.2.7.5.3 Transitions/Spirals Supported by 12d Model

39.2.7.5.1 Spirals

A spiral curve (Euler spiral) is a special type of transition where the radius of curvature is proportional to the length along the curve.

That is

Radius of curvature at a point x length from start of spiral to that point = Constant

rl = K

Although the above definition fully defines a spiral (or Euler curve), spiral calculations are difficult by hand and so no authority uses the full definition but uses an approximation to the Euler spiral.

A Euler spiral can be uniquely defined in terms of a start tangent vector, a final radius of curvature (R) and a total spiral length (L).

The equation for this spiral can be given in terms of a local co-ordinate system where the origin is at the start of the spiral and the x-direction (abscissa) is along the tangent vector at the start of the spiral. The y-direction (ordinate) is given as the offset from the x-axis of the point on the spiral.

Formulae for the local co-ordinates of a point on the spiral can then be derived in terms of the distance of the point along the spiral (the spiral length to the point) and the given constants L (the total spiral length) and R (the final radius of curvature of the spiral).

These formulae for the local co-ordinates of a point on the spiral are **polynomial series** in terms of the spiral length to that point.

For use in calculations, the local co-ordinates can be **approximated** by restricting the polynomial series for the abscissa and the offset to a fixed number of terms.

So for example, the so called ***clothoid*** spiral used by Australian road authorities is defined as using exactly 5 terms for the abscissa (x) polynomial and exactly 4 terms of the offset (y) polynomial.

Note that a spiral is a transition but not all transitions are spirals.

Continue to the next section 39.2.7.5.2 Left/Right, Leading/Trailing Transitions or return to 39.2.7 Some Special Setup Files.

39.2.7.5.2 Left/Right, Leading/Trailing Transitions

The radius of curvature **R** (or simply the radius) of a transition is allowed to be positive or negative.
To agree convention of going to the left or right when travelling along a road and the sign of the curve radius, the radius of curvature **R** of a transition is allowed to be positive or negative.

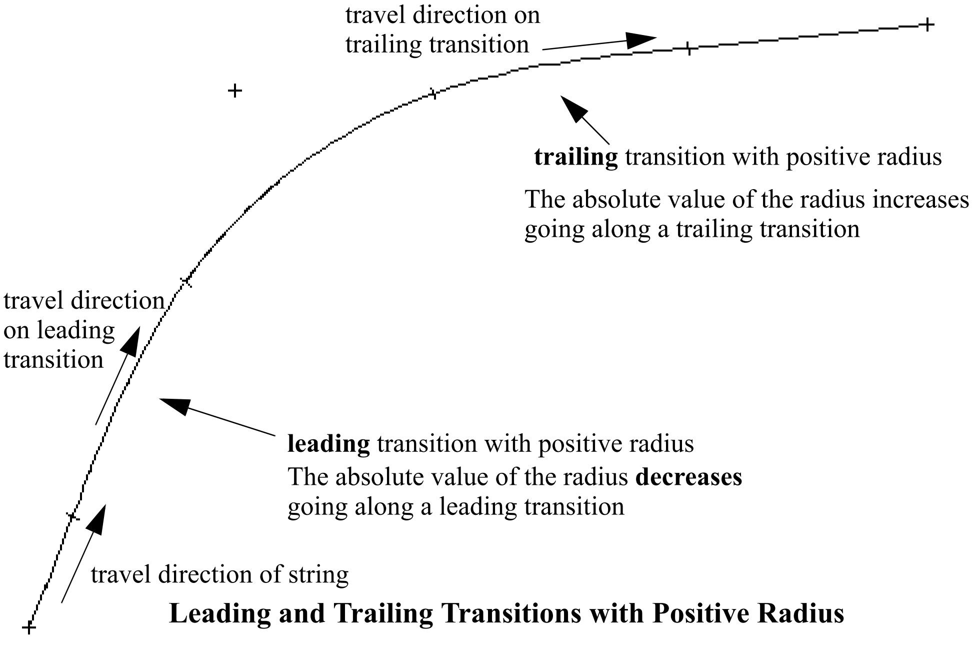
If **R** is **positive**, the transition will then curve to the **right**

If **R** is **negative**, the transition will then curve to the **left**

Also depending on the direction of travel along a transition, the absolute radius of curvature will be increasing or decreasing going along the transition.

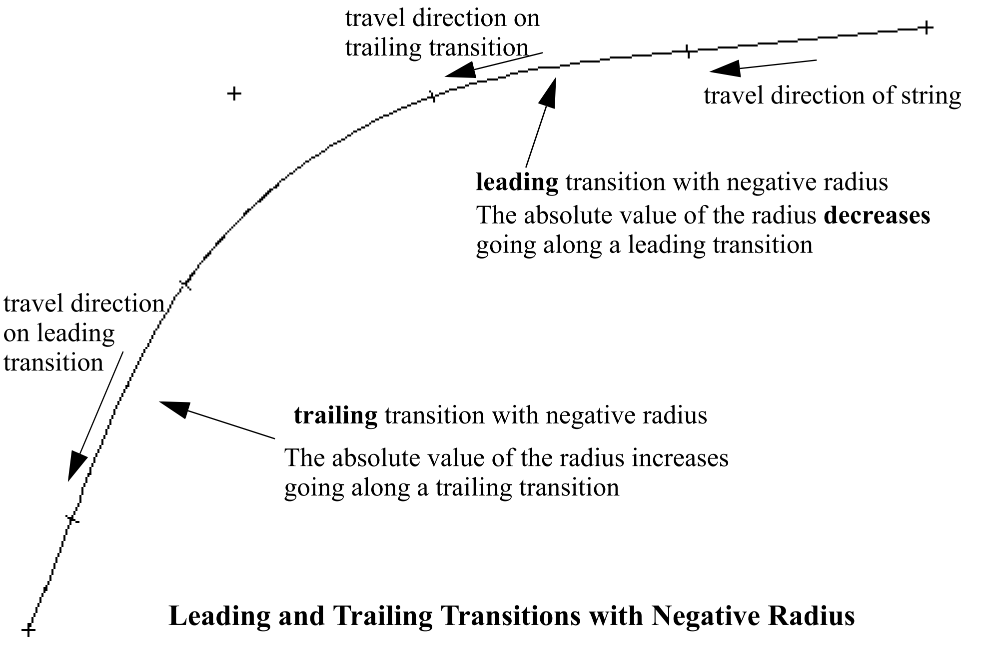
If the radius is decreasing along the transition, it is called a **leading** transition.

If the radius is increasing along the transition, it is called a **trailing** transition.



Leading and Trailing Transitions with Positive Radius

If the direction of the travel along the string is reversed, the radius of the transitions, and the arc, will be negative and the leading transition become trailing transition and the trailing transition becomes a leading transition.



Leading and Trailing Transitions with Negative Radius

Continue to the next section 39.2.7.5.3 Transitions/Spirals Supported by 12d Model or return to 39.2.7 Some Special Setup Files.

39.2.7.5.3 Transitions/Spirals Supported by 12d Model

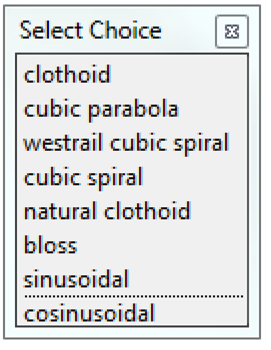
Users must be aware of the exact transition definitions that are to be used in their work.

In Australia, totally different transitions are used by NSW Rail, Queensland Rail and WA Rail.

Luckily all the road authorities in Australia use the same spiral approximation, and that is the same spiral approximation used by Queensland Rail.

So if work is being done in Western Australia or New South Wales, a different transition type is needed in the road alignments from that used in any rail alignments.

The transitions/spirals supported by ***12d Model*** are:



Clothoid - spiral approximation used by Australian road authorities and Queensland Rail.

Cubic parabola – special transition curve used by NSW railways. Not a spiral.

Westrail cubic – spiral approximating used by WA railways.

Cubic spiral – low level spiral approximation. Only ever used in surveying textbooks.

Natural Clothoid – the proper Euler spiral. Not used by any authority.

Bloss – special transition used by Deutsche Bahn. Not a spiral.

Sinusoidal - special transition. Not a spiral.

Cosinusoidal - special transition. Not a spiral.

Return to 39.2.7 Some Special Setup Files.

39.2.7.6 GUI

Note - this section is only for versions earlier than V4.0 and is here for historical reasons

The file **gui.4d**defined the colours used for screen objects (the graphical user interface objects), screen fonts, maximum pop-up length, and spacing in the menus and panels.

The gui.4d file is read every time a project (new or old) is opened.

39.2.7.6.1 GUI Colours

The colours associated with each screen object are given in terms of intensity values of red, green and blue. The intensity values are between 0 and 255, where 0 represents no colour and 255 full colour.

The layout for screen colours in the gui.4d file is:

```
//   object                            RGB values- red green blue

VIEW BACKGROUND COLOUR           0      0       0         // black
VIEW BORDER COLOUR                 255   255    0         // yellow

BUTTON HIGHLIGHT COLOUR        255     255    0           // yellow
BUTTON TEXT COLOUR                127    127    127       // grey
BUTTON BACKGROUND COLOUR         60     60    200         // dark blue

MENU BACKGROUND COLOUR         150     90     0           // brown
MENU BORDER COLOUR                 255      255   0       // yellow

PANEL BACKGROUND COLOUR        150     90     0            // brown
PANEL BORDER COLOUR                255     255   0        // yellow

FRONT SCREEN LOGO COLOUR          255      255    0       // yellow
FRONT SCREEN TEXT COLOUR           0        255     0     // green
FRONT SCREEN BACKGROUND COLOUR    60     60    200        // dark blue

WINDOW BACKGROUND COLOUR 1      64       128       0      // half green
WINDOW BACKGROUND COLOUR 2       0       128      196     // cyan
```

39.2.7.6.2 Maximum Pop-Up Length

The maximum number of items in a pop-up list before splitting into walk-right pop-ups is given by (necessary when using VGA screens on PC's)

```
POPUP LENGTH            integer        //maximum number of items in a pop-up
```

39.2.7.6.3 Fonts for Menus and Panels

A font can be defined for use in the menu and panel titles areas and a separate font for the rest of the text in the menus and panels.

The fonts are defined in the gui.4d file as:

```
SYSTEM TITLE FONT             font_name            // font for titles
SYSTEM FONT                   font_name            // font used elsewhere
```

For X-Windows, the font_name is the name of the required font from the font list given by the command xlsfonts.

For Windows NT, the font_name is made up of the Windows font name plus zero or more parameter values for the font.

The font name and parameter values are given as one text name, font_name, by concatenating the font name and values with only a minus separating them.

If the font name consists of more than one word, the font name is enclosed in double quotes(").

Hence for NT, an example of defining fonts is:

Parameter	**Possible Values**
Font	font name (less than 32 characters)
Height	number
italic	
underline	
strikeout	
Weight	thin extralight light normal medium semibold bold extrabold heavy
Quality	draft proof
Pitch	default_pitch fixed_pitch variable_pitch
Family	decorative modern roman script swiss

Defaults

Height	14
Weights	fw_dontcare (family weight don't care)
Pitch	font default pitch
Family	ff_dontcare (font family don't care)
Quality	default_quality
italic	false
underline	false
strikeout	false

For example

Arial-14-bold-italic	is the fond Arial, of height 14 and bold and italic
"Courier New"-16-italic	is the font Courier New, of height 16 and bold.

Under NT, if the font is not properly defined or doesn't exist, then the system font is used.

Under X-Windows, if the font does not exist, the font fixed is tried. If fixed does not exist, 12d Model will not start up.

39.2.7.6.4 Spacing for Borders and Panels

There are parameter to control the amount of space in border, between items etc. for menus and borders. These are normally set by 12D Solutions and should not need to be modified.

SCREEN TEXT BORDER X	pixels
SCREEN TEXT BORDER Y	pixels
SCREEN TEXT EXTRA X	pixels
SCREEN TEXT EXTRA Y	pixels
EDIT BOX BORDER X	pixels
EDIT BOX BORDER Y	pixels

PANEL BORDER X	pixels
PANEL BORDER Y	pixels
PANEL GAP Y	pixels
VERTICAL BORDER X	pixels
VERTICAL BORDER Y	pixels
VERTICAL GAP Y	pixels

HORIZONTAL BORDER X	pixels
HORIZONTAL BORDER Y	pixels
HORIZONTAL GAP Y	pixels
INPUT BOX BORDER X	pixels
INPUT BOX BORDER Y	pixels

39.3 Library, User Library, Customer Library

Apart from ***Setting Up*** files which are **only accessed** when a project is **opened** or **first created**, there are two or three special library areas that are searched for files with the appropriate file ending whenever the folder icon at the end of a panel file field is selected. For example, clicking on the folder icon for a **Map File** panel field would display a list of files ending in **.mapfile** and **.mf**.

One folder, **Library**, is supplied by *12d Solutions Pty Ltd* and is installed by ***12d Model***, and two other folders, **User_Lib** and **Customer_Lib** are not touched in an install and only contains files placed there by users.

39.3.1 Library

In a manner similar to the folder **set_ups**, the location of **Library** can be modified defining the environment variable

LIB_4D

If the environment variable LIB_4D doesn't exist, or the folder it points to doesn't exist, then the folder **Library** installed by 12d as **Program files\12d\12dmodel\11.00\Library** is used.

The files of the required ending in **library** are listed under the **[Lib]** walk right of pop-ups.

39.3.2 User Library

Because most users do not have write access to **Program files\12d\12dmodel\11.00\Library,** and the fact that any files in that area may be deleted or overwritten by future ***12d Model*** installations, there is a special user library folder called **User_lib** where user library files should be placed.

The user library, is also a folder and is pointed to by the environment variable

USER_LIB_4D

or if the environment variable USER_LIB_4D or the folder that it points to doesn't exist, in a folder called **User_lib** directly under the user accessible area **C:\12d\11.00**

The files from the User library are listed under the **[User Lib]** walk right of the pop-up.

So unlike the files in the folders *set_ups* or *user*, files in the ***12d*** supplied area **Library** and the user file in **User_lib**, are all displayed.

39.3.3 Customer Library

There is one more folder that is only looked for when it has been created by a user and is ***pointed to by the environment variable*** CUSTOMER_LIB_4D. It has no default name but is whatever name the user gave it. For convenience, we will call it **Customer_lib.**

So the customer library, is also a folder and is pointed to by the environment variable

CUSTOMER_LIB_4D

If the environment variable CUSTOMER_LIB_4D or the folder that it points to doesn't exist, then no Customer Library is used.

The files from the Customer library are listed under the **[Customer Lib]** walk right of the pop-up.

So unlike the files in the folders *set_ups* or *user*, files in the ***12d*** supplied area **Library** and the users files in **User_lib** and **Customer_lib**, are all displayed.

Notes

1. The list of special endings for files is given in the Appendix, Special File Formats.
2. A full list of environment variables is given in the later section 39.4 Environment Variables.

Please continue to the next section 39.4 Environment Variables.

39.4 Environment Variables

When ***12d Model*** is invoked for a new or existing project, it uses environment variables to tailor the system.

In Windows, environment variables can be set for each User from the Control Panel, but a much easier method to set the environment variables used by ***12d Model*** is to include them in a special file which is read in each time a project (new or existing) is opened by ***12d Model***.

The environment variables that ***12d Model*** recognises can be broken into two types

(a) mode type where the environment variable is only a flag setting a mode

and

(b) a second type which point to a file, program or folder. For the environment variables of the second type, a default file/program is often searched for if the environment variable is not defined.

The folder search order for the default files for the file group of environment variables is given in the previous sections.

The list of environment variables will be given in alphabetical order but it will be obvious from the documentation which type the environment variable is.

39.4.1 Alphabetical Environment Variables List

For environment variables that simply set a flag or value (the default value is shown in bold), the documentation will be:

Variable Name Value Description Default

For environment variables that point to a file, program or folder, and their associated default files/ programs, the documentation will be:

Environment Variable Name Type Default

Environmental Variables List

ACAD_SEEDFILES_4D

ACAD_SEEDFILES_4D folder no default

if not blank, the full path name of the folder holding Autocad template files.

ADAC_VERSION_4D

ADAC_VERSION_4D text

the ADAC Schema version, *e.g.* 4.1.0 or 4.0.0

ALIGNMENT_CORRIDOR_FIXUP_4D

ALIGNMENT_CORRIDOR_FIXUP_4D value 0 or **1** *default* 1

if not zero, alignment corridor calculations introduced in V8 are used.
If zero, the V7 alignment corridor calculations are used.

ALLOW_ANONYMOUS_FUNCTIONS_4D

ALLOW_ANONYMOUS_FUNCTIONS_4D value 0 or 1

if tick, when a file is read with a File input option, a function is automatically created and named. This function must exist to allow data to be added to a view after it is read in, to be able to delete all the data read in at a later time, and to re-run the function to reread the data file and replace the data read in last time.

ALLOW_NAMED_POINT_ATTRIBUTES_4D

ALLOW_NAMED_POINT_ATTRIBUTES_4D value 0 or 1

ALLOW_OLD_PLOTTING_4D

ALLOW_OLD_PLOTTING_4D value 0 or 1

If not zero, then a menu **Plot =>Old plotting** is included which has all the old text ppf options.

ALLOW_SUPER_TINS_EXACT_CALCS_4D

ALLOW_SUPER_TINS_EXACT_CALCS_4D value 0 or 1

Options using tins are split into two categories:

(1) options that require sections through tins
(2) options that require the triangles of the tin

if ticked, this allows super tins to be used in options where triangles are required for calculations.
if not ticked, this prevents super tins to be used in options where triangles are required for calculations.

Note: ticking this option requires super tins to be updated when a tin within the super tin changes, so

there is a time, storage, and memory cost. It was found that some projects only ever required options that used sections (not triangles), so for these type of projects, there is an efficiency gain by turning off this option.

ALWAYS_VALIDATE_PROJECT_DETAILS_4D

ALWAYS_VALIDATE_PROJECT_DETAILS_4D value 0 or 1

if ALLOW_ANONYMOUS_FUNCTIONS_4D is set to 1, then when a file is read with a File input option, a function is automatically created and named. This function can be used to delete all the data at a later time, to allow the data to be added to a view after it is read in, and to re-run the function to reread in the data file.

ASK_ON_CHAIN_COMMAND_DELETE_4D

ASK_ON_CHAIN_COMMAND_DELETE_4D

AUTHORIZATION_4D

AUTHORIZATION_4D full path name to nodes.4d

The authorization file is normally called **nodes.4d**, and is under the folder c:\12d**12d model**\version_number. However, the file name and path can be set by the user with the AUTHORIZATION_4D environment variable.

The full path-name of the file is given.

AUTO_CERTIFY_DONGLE_4D

AUTO_CERTIFY_DONGLE_4D value **0** or 1 *default* 0

if non zero, then when the warning period is active for the CodeMeter Container (dongle) being used by ***12d Model***, ***12d Model*** will automatically attempt to certify the dongle and if successful, no intervention by the user is required. See 7.6.5.4 Certify CodeMeter.

Note that the computer must have internet access for Certification to take place.

AUTO_DELETE_WALKRIGHTS_4D

AUTO_DELETE_WALKRIGHTS_4D	**0**	Use old method from V3.1	32
	pixels	Distance in pixels to move past the end of the last menu before the cascade collapses	

If not zero, the environment variable AUTO_DELETE_WALKRIGHTS_4D sets the distance in pixels that is used to collapse the cascade of walk-right menus when the cursor moves that distance past the end of the last walk-right menu.

The default value is 32.

AUTO_HIDE_PANEL_SELECTS_4D

AUTO_HIDE_PANEL_SELECTS_4D value 0 or **1** *default* 1

if non zero, when the string select icon is picked on a panel, the panel minimises until a string is selected is accepted.

This mode was introduced so that the panel was not in the way when selecting a string. It is particularly useful on tablets with limited screen area.

AUTO_MODEL_SYNC_4D

AUTO_MODEL_SYNC_4D	**0**	Don't auto sync models	*default 0*
	1	Auto sync models	

if auto-model sync is set to on, the server projects for any shared models added to this project are checked to see if they have been modified (checked every SHARE_CHECK_INTERVAL seconds). If any models have been modified, they are re-copied to this project.

AUTO_PAN_SELECT_4D

AUTO_PAN_SELECT_4D value 0 or **1** *default* 1

if non zero, if you have accepted but not accepted a string and are zoomed in on the strings and type **ch value** where the position at that chainage is off the view, the view will autopan so that the new selection point (at chainage value) is on the view. This applies for all typed selects.

AUTO_RESET_SELECT_4D

AUTO_RESET_SELECT_4D	0	Don't use auto-reset reselect
	1	Use auto-reset reselect

The reset mechanism for picking has ben modified for V5.0 and above. Reset is now done automatically if the cursor is moved a user specified distance (given by AUTO_RESET_TOLERANCE_4D) after a pick (without accepting), and a separate user specified distance (given by AUTO_RESET_DRAG_TOLERANCE_4D) after a directional pick (without accepting).

The default value is 1.

AUTO_RESET_SELECT_DRAG_TOLERANCE_4D

AUTO_RESET_SELECT_DRAG_TOLERANCE_4D *value*

Reset distance for directional picks.

If AUTO_RESET_SELECT_4D is non zero, then AUTO_RESET_SELECT_DRAG_TOLERANCE_4D is the distance in pixels to move the cursor to reset the picking rejection list for a directional pick.

The default value is 50.

AUTO_RESET_TOLERANCE_4D

AUTO_RESET_TOLERANCE_4D *value*

Reset distance for non-directional picks.

If AUTO_RESET_SELECT_4D is non zero, then AUTO_RESET_TOLERANCE_4D is the distance in pixels to move the cursor to reset the picking rejection list for a non-directional pick.

The default value is 5.

AUTO_TIN_SYNC_4D

AUTO_TIN_SYNC_4D	**0**	Don't auto sync tins	*default 0*
	1	Auto sync tins	

if auto-tin sync is set to on, the server projects for any shared tins added to this project are checked to see if they have been modified (checked every SHARE_CHECK_INTERVAL seconds). If any tins have been modified, they are re-copied to this project.

AUTOCAD_PATTERNS_4D

AUTOCAD_PATTERNS_4D folder no default

if non blank, the full path name of the AutoCAD patterns file.

BISECTORS_4D

BISECTORS_4D	**0**	Don't use bisector section use two sections instead.	*default 0*
	1	Use bisector section	

When applying templates to a horizontal intersection point with a sharp change of direction, (that is, it is not and end point and the HIP has no curve on it and there is a change of direction at the HIP), either two sections can be applied at the HIP point (applied perpendicular to the line on either side of the HIP point) or just a single bisector section applied to the bisector of the change of angle through the HIP.

The default value is 0.

BOXING_CONTINUE_ON_FAIL_4D

BOXING_CONTINUE_ON_FAILURE_4D value **0** or 1 *default* 0

if non zero, the boxing stops if there is an error such as not being able to intersect with a tin. If zero, the boxing stops when an error is found.

BOXING_RULES_COLOUR_4D

BOXING_RULES_COLOUR_4D *colour* *default blue*

the colour of the text in the Type column in the Boxing Rules panel for all Boxing command other than Comment, Decision, Goto and Labels.

BOXING_COMMENT_COLOUR_4D

BOXING_COMMENT_COLOUR_4D *colour* *default yellow*

background colour for the Comment boxing command grid row in the Boxing Rules panel. The text in the grid row is black.

BOXING_DECISION_GOTO_COLOUR_4D

BOXING_DECISION_GOTO_COLOUR_4D *colour* *default mauve*

the colour of the text in the Type column for the Decision and Goto commands in the Boxing Rules panel.

BOXING_LABEL_COLOUR_4D

BOXING_LABEL_COLOUR_4D *colour* *default light red*

the colour of the text in the Type column for the Label command in the Boxing Rules panel.

BOXING_REGION_COLOUR_4D

BOXING_REGION_COLOUR_4D *colour*

the colour of the text in the row for the Region command in the Boxing Rules panel.

BOXING_WALL_OFFSET_4D

BOXING_WALL_OFFSET_4D *real* *default 0.000001*

when boxing automatically creates a vertical wall, this is the offset distance between the top and bottom of the vertical wall.

CAD_START_IN_MULTI_PICK_4D

CAD_START_IN_MULTI_PICK_4D tick box

CHECK_FOR_UPDATES_4D

CHECK_FOR_UPDATES_4D integer

if non zero, the 12d web site is checked on startup for any newer updates to ***12d Model***.

If zero, the 12d web site is not checked.

CIVILCAD_PATH_4D

CIVILCAD_PATH_4D folder no default

if non blank, the full path name of the folder holding CivilCAD files.

There is no default.

COLOURS_4D

COLOURS_4D filename default colours.4d

file of colour rgb definitions and names. See 39.2.7.2 Colours File (colours.4d).

The default is colours.4d.

CONSTRUCTION_SNAP_MODEL_4D

CONSTRUCTION_SNAP_MODEL_4D model name

If non blank, the model to use for objects created during construction snaps.

CUBED_CHARACTER_4D

CUBED_CHARACTER_4D *integer (base 10)* *default 179*

The integer (base 10) value of the character to use as the cubed symbol.

CUSTOMER_LIB_4D

CUSTOMER_LIB_4D folder name no default name

if the environment variable exists and the folder exists, then the folder is available when selecting a library file (*see* 39.3 Library, User Library, Customer Library).

There is no default folder name.

CUSTOMER_USER_4D

CUSTOMER_USER_4D folder name no default name

if the environment variable exists and the folder exists, then the folder is part of the search for locating setup files (*see* 39.2 Files for Setting Up 12d).

There is no default folder name.

CUT_HEIGHT_ZONE_COLOUR_4D

CUT_HEIGHT_ZONE_COLOUR_4D *colour* *default dark red*

the colour of the text in the *Type* column for the *Cut* commands involving height, in the *Modifiers* section of the ***Left/Right MTF Modifiers*** *panel*.

CUT_WIDTH_ZONE_COLOUR_4D

CUT_WIDTH_ZONE_COLOUR_4D *colour* *default light red*

the colour of the text in the *Type* column for the *Cut* commands involving width, in the *Modifiers* section of the ***Left/Right MTF Modifiers*** *panel*.

CUT_SLOPE_ZONE_COLOUR_4D

CUT_SLOPE_ZONE_COLOUR_4D *colour* *default dark red*

the colour of the text in the *Type* column for the *Cut* commands involving slope, in the *Modifiers* section of the ***Left/Right MTF Modifiers*** *panel*.

CUT_ZONE_COLOUR_4D

CUT_ZONE_COLOUR_4D *colour* *default red*

the colour of the text in the *Type* column for the *Cut* commands (other than those involving width, height or slope) in the *Modifiers* section of the ***Left/Right MTF Modifiers*** *panel*.

DATA_COLLECTOR_4D

DATA_COLLECTOR_4D data collect name

the data collector that is used if no data collector is set for a project.

There is no default.

DATA_COLLECTORS_4D

DATA_COLLECTORS_4D filename default survey.4d file of definitions of available data collectors

The default is survey.4d.

DATA_TIPS_4D

DATA_TIPS_4D	**0**	Don't show data tips
	1	Show data tips

When non zero, data tips are displayed when the cursor moves over vertices in a plan view.

The default value is 0.

DEBUG_DONGLE_ACCESS_4D

DEBUG_DONGLE_ACCESS_4D   positive integer

time between searches for a dongle

DECISION_ZONE_COLOUR_4D

DECISION_ZONE_COLOUR_4D   *colour*   *default green*

the colour of the text in the *Type* column for the *Decision* commands in the *Modifiers* section of the ***Left/Right MTF Modifiers*** *panel*.

DEFAULTS_4D

DEFAULTS_4D   filename   default defaults.4d

file of defaults - only used for new project

The default is defaults.4d. See 39.2.7.3 Defaults File (defaults.4d).

DEFAULT_PLAN_PLOT_SCALE_4D

DEFAULT_PLAN_PLOT_SCALE_4D   *value*

value is the scale (1:value) for new plan views

If a value is given, then it is used the default scale for any new plan view created in a project. Note that the scale is used in the plan view for displaying text, linestyles and symbols that have been defined in paper units.

DEFAULT_RASTER_DPI_4D

DEFAULT_RASTER_DPI_4D   *real_dots_per_inch*

default dots per inch to use for plotting rasters

The default value is 150.0

DEFAULT_TABLE_WIDTH_4D

DEFAULT_TABLE_WIDTH_4D   *number_of_characters*   *default 32*

Gives the number of characters displayed in tables such as decisionals and mtf editor.

The default value is 32.

DEFAULT_VIEW_COLOUR_4D

DEFAULT_VIEW_COLOUR_4D   *colour_number*   *default 0*

The default background colour for views.

DEGREES_CHARACTER_4D

DEGREES_CHARACTER_4D   *integer (base 10)*   *default 176*

The integer (base 10) value of the character to use as the degrees symbol.

DGN_PLOT_SEED_FILE_4D

DGN_PLOT_SEED_FILE_4D filename

if non blank, the full path name of the file to use as a seed file for Microstation plots.

DIAMETER_LARGE_CHARACTER_4D

DIAMETER_LARGE_CHARACTER_4D *integer (base 10)* *default 216*

The integer (base 10) value of the character to use as the large diameter symbol.

DIAMETER_SMALL_CHARACTER_4D

DIAMETER_SMALL_CHARACTER_4D *integer (base 10)* *default 248*

The integer (base 10) value of the character to use as the small diameter symbol.

DIGITIZERS_4D

DIGITIZERS_4D filename *default digitize.4d*

file of digitizer definitions

The default is digitize.4d. See 8.5.7 Digitizer Definitions File.

DISABLE_MTF_WARNINGS_4D

DISABLE_MTF_WARNINGS_4D	**0**	Display the mtf warnings in the text editor
	1	Write mtf warnings to the background window

Controls whether the mtf warnings go to a file or to the output window.

If non zero, write mtf warnings to the background window
If zero, display the mtf warnings in the text editor.

The default value is 0.

DONGLE_4D - no longer used

Important Note: this environment variable is not used for ***12d Model 11*** and above.

Hardlock dongles will not work for ***12d Model*** 11 and above so this environment variable is ignored by these versions of ***12d Model***.

The environment variable can have a number of parameters to control the use of Hardlock network and stand alone dongles.

(a) DONGLE_4D -local OR -no_local

usually used when there is a network dongle and need to know if any stand alone Hardlock dongle is to be looked for on the computer that the user is on.

if **-local**, then a stand alone Hardlock dongle is looked for on the computer the user is on.
If **-no local**, then no stand alone Hardlock dongle is looked for on the computer the user is on.

For example DONGLE_4D - local or -no local

(b) DONGLE_4D -no network OR -network first OR -network last

need to specify if any network Hardlock network dongle is to be looked for.

if -**no network**, then no Hardlock network dongle is looked for.
If **-network first**, then a Hardlock network dongle is looked for before a stand alone local dongle.
If **-network last**, then a Hardlock network dongle is looked for after looking for a stand alone local dongle.

For example DONGLE_4D -no_network *or* - network_first *or* -network_last

(c) DONGLE_4D -login_retries number of retries to find a Hardlock dongle

the number ***nn*** of retries to make when searching for a Hardlock dongle and it is not found.

For example DONGLE_4D -login_retries 3

(d) DONGLE_4D -login_wait ss

the number of seconds ***ss*** to wait between retries to find a network Hardlock dongle.

For example DONGLE_4D -login_wait 5

(e) DONGLE_4D -no_dongle

if the parameter -no_dongle is found, then no search is made for a Hardlock dongle.

For example DONGLE_4D -no_dongle

(f) DONGLE_4D -debug

if the parameter -debug is found, then Hardlock dongle debug information is written to the

For example DONGLE_4D -debug

See the separate notes on installing a network dongle for more details.

DONGLE_ORDER_4D - no longer used

Important Note: this environment variable is not used for ***12d Model 11*** and above.

Hardlock dongles will not work for ***12d Model*** 11 and above so this environment variable is ignored by these versions of ***12d Model***.

DONGLE_ORDER_4D choice box Hardlock/Wibu, Wibu/Hardlock

the order to search for network dongles.

If **Wibu/Hardlock**, then Wibu networks dongles are searched for before any Hardlock dongles.

If **Hardlock/Wibu**, then Hardlock networks dongles are searched for before any Wibu dongles.

All new dongles are Wibu so most ***12d Model*** sites now only have Wibu dongles.

DONGLES_4D

DONGLES_4D full path name to dongles.4d

The dongles file is normally called **dongles.4d** and is searched for as a Set Up file. However, the file name and path can be set by the user with the DONGLES_4D environment variable.

The full path-name of the file is then given.

DOUBLE_CONFIRM_DELETE_4D

DOUBLE_CONFIRM_DELETE_4D	0	Confirm deletes/cleans without undos once
	1	Confirm deletes/cleans without undos twice

Controls whether or not the user is asked once or twice to confirm that deletes and cleans that do not have undos are to be done.

If non zero, the user is asked twice to confirm for deletes and cleans.
If zero, the user is asked once to confirm for deletes and cleans.

The default value is 1.

DRAFTING_SPLIT_MESSAGE_4D

DRAFTING_SPLIT_MESSAGE_4D	**0**	Confirm deletes/cleans without undos once
	1	Confirm deletes/cleans without undos twice

if non zero, for CAD Dimensions, Leaders and Tables, long messages that are written to the screen message area are broken up into two shorter lines with the first one being displayed. Both the shorter lines have a (m)ore on them to get to the other line. However all the typed commands for the options are still recognised regardless of whether they are visible or not in the split message.

If zero, the messages are not split into two lines.

The default value is 0.

DRAINAGE_4D

DRAINAGE_4D filename default drainage.4d

file of definitions for the drainage option

The default is drainage.4d.

DRAINAGE_EDIT_GRADE_4D

DRAINAGE__EDIT_GRADE_4D tick box

DRAINAGE_PPF_4D

DRAINAGE_PPF_4D *full path name of .ppf file* no default

Points to the file used as the default plot parameter file (.ppf) for the panel **New Plot Drainage Network.**

The path name to the .ppf file can contain $LIB which expands out to the path of the library area. For example, **$LIB/drainage.ppf** points to the file drainage.ppf in the area defined by $LIB_4D.

DRAINAGE_FLOW_DIR_4D

DRAINAGE_FLOW_DIR_4D **0** Flow direction is opposite to the string direction
1 Flow direction is the same as string direction

Defines the default flow direction used in the creating a drainage string.

The environment variable DRAINAGE_FLOW_DIR_4D specified whether the default flow direction that appears in the **Create Drainage String** panel is that the flow direction in the line is in the same direction as the direction of the drainage string (ascending chainage) or that the flow direction in the drainage line is in the opposite direction to the direction of the drainage string (descending chainage).

If non zero, the drainage flow direction is the same as the string direction.
If zero, the drainage flow direction is in the opposite direction to the string direction.

The default value is 0.

DWG_PLOT_SEED_FILE_4D

DWG_PLOT_SEED_FILE_4D filename

if non blank, the full path name of the file to use as a template file for AutoCAD plots.

DWG_PLOT_UNIT_4D

DWG_PLOT_UNIT_4D English OR Metric

if non blank, the units to use for AutoCAD files.

EDIT_BOX_WIDTH_IN_CHARACTERS_4D

EDIT_BOX_WIDTH_IN_CHARACTERS_4D *number* *default 10*

if non blank, the number of characters to make the width of an Edit Box on a panel.

EDITOR_4D

EDITOR_4D script/program 10.00\cpu_area\te

the script or program that is fired up when a report is created. It usually points to an editor.

If the parameter **display reports** in the panel **sys default settings** (given by the menu option **Utilities => Default=>Sys defaults**) is set to **yes**, the **editor displays** each report as it is created.

The default is **10.00\cpu_area\te**.

ENVIRONMENT_4D

ENVIRONMENT_4D *full path name of file* default env.4d

It is possible to set all the environment variables from a file normally called **env.4**d. The environment variable file is searched for in the standard Set Up areas (local, USER_4D, user, set_ups) or is set by the environment variable ENVIRONMENT_4D.

The default is env.4d.

The format of the environment variable file is given in the section 39.4.2 Setting Environment Variables.

EXTRA_OPTIONS_4D

EXTRA_OPTIONS_4D file default xtramenu.4d

If non blank, the full path name of the 12d Solutions supplied file of definitions for some extra menus.

The default is xtramenu.4d. See 41.2 User Defined Menus

EXTRA_START_END_MTF_4D

EXTRA_START_END_MTF_4D 0 extra start/end on *default **1***
1 extra start/end on

Used in the Modifiers section of the MTF.

If EXTRA_START_END_MTF_4D =1, then the **Extra start** and **Extra end** tick boxes on the MTF panels are ticked.

If EXTRA_START_END_MTF_4D =0, then the **Extra start** and **Extra start** tick boxes on the MTF panels are not ticked.

The default value is 1.

EXTRUSIONS_4D

EXTRUSIONS_4D filename default extrusions.4d

This points to the file which contains the definition of the extrusions in the Extrusions Library. Please see 12.13.6 Extrusions for more information about extrusions.

The default is extrusions.4d.

FAST_ACCEPT_4D

FAST_ACCEPT_4D **0** don't set fast accept snap to on for new projects *default 0*
1 set fast accept snap on for new projects

If non zero, the **Fast Accept** snap (**A** snap) is turned on by default for new projects.

If fast accept is turned on, then when an item is picked and there is only one item in the selection list, that item is automatically selected without clicking MB.

FAST_CONSTRUCTION_SNAPS_4D

FAST_CONSTRUCTION_SNAPS_4D **0** don't set construction snap to on for new projects
1 set construction snap to on for new projects

If non zero, the **Fast Construction** snap (**K** snap) is turned on by default for new projects.

The default value is 0.

FAST_ZOOM_PAN_4D

FAST_ZOOM_PAN_4D

FLD_IGNORE_XTRA_WORDS_4D

FLD_IGNORE_XTRA_WORDs_4D **0**,1

If non zero, trailing tabs are not considered words in a field file. Mainly for Leica when it is writing a format file which can't suppress trailing tabs when writing a 12d Field file.

The default value is 0.

FILE_READ_ADD_TO_VIEW_4D

FILE_READ_ADD_TO_VIEW_4D choice box

if anonymous function are set, the data created by the File input option can be added to a view. The choices are:

Do nothing - don't add the data read in to any view

Add to current view - add the data read in to the current view (the current view is the view that highlighted)

Add to new view - automatically create a new view and add the data read in to that view

Add to named view - add the data read in to the view given by FILE_READ_ADD_TO_VIEW_NAME_4D

FILE_READ_ADD_TO_VIEW_NAME_4D

FILE_READ_ADD_TO_VIEW_NAME_4D text box

name of the view to use if FILE_READ_ADD_TO_VIEW_4D is set to **Add to named view**.

If the view does not exist then it is created.

FILL_HEIGHT_ZONE_COLOUR_4D

FILL_HEIGHT_ZONE_COLOUR_4D *colour* *default dark green*

the colour of the text in the *Type* column for the *Fill* commands involving height, in the *Modifiers* section of the ***Left/Right MTF Modifiers*** *panel*.

FILL_SLOPE_ZONE_COLOUR_4D

FILL_SLOPE_ZONE_COLOUR_4D *colour* *default dark green*

the colour of the text in the *Type* column for the *Fill* commands involving slope, in the *Modifiers* section of the ***Left/Right MTF Modifiers*** *panel*.

FILL_WIDTH_ZONE_COLOUR_4D

FILL_WIDTH_ZONE_COLOUR_4D *colour* *default light green*

the colour of the text in the *Type* column for the *Fill* commands involving width, in the *Modifiers* section of the ***Left/Right MTF Modifiers*** *panel*.

FILL_ZONE_COLOUR_4D

FILL_ZONE_COLOUR_4D *colour* *default green*

the colour of the text in the *Type* column for the *Fill* commands (other than those involving width, height or slope) in the *Modifiers* section of the ***Left/Right MTF Modifiers*** *panel*.

FINAL_CUT_SLOPE_ZONE_COLOUR_4D

FINAL_CUT_SLOPE_ZONE_COLOUR_4D *colour* *default black*

the colour of the text in the *Type* column for the *Final* command for Final Cut slope, in the *Modifiers* section of the ***Left/Right MTF Modifiers*** *panel*.

FINAL_FILL_SLOPE_ZONE_COLOUR_4D

FINAL_FILL_SLOPE_ZONE_COLOUR_4D *colour* *default black*

the colour of the text in the *Type* column for the *Final* command for Final Fill slope, in the *Modifiers* section of the ***Left/Right MTF Modifiers*** *panel*.

FINAL_NO_CUT_SLOPE_ZONE_COLOUR_4D

FINAL_NO_CUT_SLOPE_ZONE_COLOUR_4D *colour* *default black*

the colour of the text in the *Type* column for the *Final* command for No Cut slope, in the *Modifiers* section of the ***MTF Modifiers*** *panel*.

FINAL_NO_CUT_FILL_SLOPE_ZONE_COLOUR_4D

FINAL_NO_CUT_FILL_SLOPE_ZONE_COLOUR_4D *colour* *default dark brown*

the colour of the text in the *Type* column for the *Final* commands for No cut/fill, in the *Modifiers* section of the ***MTF Modifiers*** *panel*.

FINAL_NO_FILL_SLOPE_ZONE_COLOUR_4D

FINAL_NO_FILL_SLOPE_ZONE_COLOUR_4D *colour* *default black*

the colour of the text in the *Type* column for the *Final* command for No Fill slope, in the *Modifiers* section of the ***MTF Modifiers*** *panel*.

FINAL_WIDTH_ZONE_COLOUR_4D

FINAL_WIDTH_ZONE_COLOUR_4D *colour* *default light brown*

the colour of the text in the *Type* column for the *Final* commands involving width, in the *Modifiers* section of the ***MTF Modifiers*** *panel*.

FINAL_ZONE_COLOUR_4D

FINAL_ZONE_COLOUR_4D *colour* *default brown*

the colour of the text in the *Type* column for the *Final* commands (other than those involving width or slope) in the *Modifiers* section of the ***MTF Modifiers*** *panel*.

FIXED_DECISION_ZONE_COLOUR_4D

FIXED_DECISION_ZONE_COLOUR_4D *colour*

the colour of the text in the Type column for the Fixed Decision commands in the **Left/Right MTF Modifiers** panel.

FIXED_HEIGHT_ZONE_COLOUR_4D

FIXED_HEIGHT_ZONE_COLOUR_4D *colour* *default dark blue*

the colour of the text in the *Type* column for the *Fixed* commands involving height, in the *Modifiers* section of the ***Left/Right MTF Modifiers*** *panel*.

FIXED_WIDTH_ZONE_COLOUR_4D

FIXED_WIDTH_ZONE_COLOUR_4D *colour* *default light blue*

the colour of the text in the *Type* column for the *Fixed* commands involving width, in the *Modifiers* section of the ***Left/Right MTF Modifiers*** *panel*.

FIXED_XFALL_ZONE_COLOUR_4D

FIXED_XFALL_ZONE_COLOUR_4D *colour* *default dark blue*

the colour of the text in the *Type* column for the *Fixed* commands involving xfall, in the *Modifiers* section of the ***Left/Right MTF Modifiers*** *panel*.

FIXED_ZONE_COLOUR_4D

FIXED_ZONE_COLOUR_4D *colour* *default blue*

the colour of the text in the *Type* column for the *Fixed* commands in the *Modifiers* section of the ***Left/Right MTF Modifiers*** *panel*.

FONTS_4D

FONTS_4D filename default fonts.4d

font definitions

The default is fonts.4d. See 40.3 Textstyles and Fonts

FULL_HYDRAULIC_REPORT_4D

FULL_HYDRAULIC_REPORT_4D 0, 1 default 0

The rational hydraulic report contains hydrology data.

If non zero, the hydrology calculations for each catchment set (1 to 3) are written to the hydraulic report.

FUNCTION_KEYS_4D

FUNCTION_KEYS_4D filename default userkeys.4d

function key definitions

The default is userkeys.4d. See 41.1 User Defined Function Keys

GENIO_WILDCARD_4D

GENIO_WILDCARD_4D *text* default ***.mos**

Sets the ending of the files selected for the pop-up list for the **File** field in the **Read Genio Data** panel.

The default value is "*.mos".

GIS_ARCSDE_PATH_4D

GIS_ARCSDE_PATH_4D filename

the full path name to the ArcSDE DLL. This is only required when using the ArcSDE Server.

GIS_ORACLE_PATH_4D

GIS_ORACLE_PATH_4D filename

the full path name to the Oracle DLL. This is only required when using the Oracle Server.

GUI_4D

GUI_4D **Standard** *default Standard*
Non Standard

Defines the mouse button usage within 12d Model.

The default value is *Standard*.

GUI_COLOURS_4D

GUI_COLOURS_4D filename default gui.4d

file of colour definitions used in the GUI

The default is gui.4d. See 39.2.7.6 GUI

HARDLOCK_4D

HARDLOCK_4D 1, 0

if non zero, search for 12d Hardlock dongles.
If zero, don't search for Hardlock dongles

HARDWARE_ARCS_4D

HARDWARE_ARCS_4D **1** Use hardware arcs *default 1*
0 Hardware arcs are not used

if non zero, use computer hardware to draw arcs (rather than software).
If zero, draw arcs in software

The default value is 1.

HEIGHT_MAX_DEFAULT_4D

HEIGHT_MAX_DEFAULT_4D integer (world units) default 0 set the default value to use in Plan Settings panels that have a **Height max (w)** field.

HELP_4D

HELP_4D	folder	default 11.00\help

folder containing the 12d Model help files.

The default is **11.00\help**.

HELP_BUTTONS_4D

HELP_BUTTONS_4D	**0**	Disable help buttons on panels
	1	Enable help buttons on panels.

If non zero, ***Help*** buttons are added to panels.

The default value is 0.

HIMETRIC_4D

HIMETRIC_4D	**0**	Printer resolution of 0.04 mm - for Win 95,98,ME*default 0*
	1	Printer resolution of 0.01 mm - for Win NT, 2000, XP

For Windows 95, 98 and ME, the printer resolution can only be 0.04 mm when covering an A0 sheet.

Under Windows NT, 2000 and XP, no such restriction exists and the full resolution of 0.01 mm can be used so the environment variable HIMETRIC_4D allows access to the higher resolution for Windows NT, 2000 and XP.

The default value is 0.

See also WINDOWS_PRINTERS_4D for enabling access to Windows printers.

HINGE_OFFSET_ZONE_COLOUR_4D

HINGE_OFFSET_ZONE_COLOUR_4D colour

the colour of the text in the Type column for the Hinge Offset commands in the **Hinge Modifiers** panel.

HINGE_HEIGHT_ZONE_COLOUR_4D

HINGE_HEIGHT_ZONE_COLOUR_4D colour

the colour of the text in the Type column for the Hinge Height commands in the **Hinge Modifiers** panel.

HINGE_POSITION_ZONE_COLOUR_4D

HINGE_POSITION_ZONE_COLOUR_4D colour

the colour of the text in the Type column for the Hinge Position commands in the **Hinge Modifiers** panel.

HLS_IPADDR

HLS_IPADDR	list of IP addresses and/or computer names

if **non blank**, a list of IP addresses and/or computer names to search for a 12d Hardlock network dongle. The items in the list are separated by commas

If **blank**, search the entire network for a 12d Hardlock network dongle

HOME_4D

HOME_4D	folder	three folder levels up from where 12d.exe is

12d Model home folder

INTERPRET_DMS_INPUT_OLD_4D

INTERPRET_DMS_INPUT_OLD_4D **0,** 1

if non zero, the special case of 0.mms is interpreted as mm minutes and s seconds. That is, 0.123 is interpreted as 12 minutes and 3 seconds.

If zero, the special case of 0.mms is interpreted as mm minutes and 10 x s seconds. That is, 0.123 is interpreted as 12 minutes and 30 seconds.

LABEL_PVC_150_PIPES_4D

LABEL_PVC_150_PIPES_4D 0, 1 default 1

If non zero, the 150 PVC pipes are labelled on the drainage long section.
If zero, the 150 PVC pipes are not labelled on the drainage long section.

LAYOUT_FILE_4D

LAYOUT_FILE_4D filename default layout.4d

a file containing the screen layout file information (slx/slf) for placing menus or panels on the screen. The layout file is used each time a project is opened to place the menus and panels in it on the screen. See 42.2.3 Screen Layout File in the appendix 42 Special File Formats.

The default is layout.4d.

LIB_4D

LIB_4D folder HOME_4D\11.00\library

12D Solutions library folder for input files such as mapping, template and macros (4DML's).

The default is **HOME_4D\11.00\library**.

LINESTYLE_BOX_HEIGHT_4D

LINESTYLE_BOX_HEIGHT_4D *value*

If non blank, the pixel height of the linestyle and symbol scrolling pop-ups.

The default is 512

LINESTYLE_BOX_WIDTH_4D

LINESTYLE_BOX_WIDTH_4D *value*

If non blank, the pixel width of the linestyle and symbol scrolling pop-ups.

The default is 256

LINESTYLE_DUPLICATE_MODE_4D

LINESTYLE_DUPLICATE_MODE_4D choice box

This environment variable manages what to do if symbols or linestyles with the same name are read from the symbols.4d or linstyle.4d file

Do nothing - accept all instances of symbols/linestyles of the same name
Take first - accept only the first instance of the symbol/linestyle
Take last - accept only the last instance of the symbol/linestyle

LINESTYLES_4D

LINESTYLES_4D filename default linestyl.4d

file of linestyle definitions

The default is linestyl.4d. See 40.1 Line Styles

LIST_POPUPS_4D

LIST_POPUPS_4D **1** Pop-up lists as scrolling lists

0 Pop-up lists as menus (as in versions up to V3.2)

Defines the look of pop-up lists.

If non zero, the popup lists are scrolling lists.
If zero, the pop-ups are menus which turn into walk-right menus when the list is too long.

The default value is 1.

LOOK3D_4D	1	3D look and greenish background	*default 1*
	0	Original 12d gui colours	

The default value is 1.

LOG_DIR_4D

LOG_DIR_4D folder-path-name default is local folder

An error log file is created each time 12d Model is invoked. The error log name automatically created by 12d Model and has the form:

log?????.4de

where ????? is a hashed number using your login name, process id & the current time.

The **folder** that the error log file is created in is given by the environment variable LOG_DIR_4D. The full path-name of the folder is given.

If LOG_DIR_4D is not used, 12d Model tries to create the log file in the current folder, the HOME folder, the TMP folder and the TEMP folder.

If creating an error log file fails in all these ares, 12d Model does not create an error log file.

LONG_SECTION_PPF_4D

LONG_SECTION_PPF_4D *path name to .ppf file* no default

This environment points to the file used as the default plot parameter file (.ppf) for the panel **section long plot**.

The path name to the .ppf file can contain $LIB which expands out to the path of the library area. For example, **$LIB/long.ppf** points to the file long.ppf in the area defined by $LIB_4D.

MACRO_INPUT_MODE_4D

MACRO_INPUT_MODE_4D	**0**	Don’t put current value into the console	*default 0*
	1	Put current value into console.	

Controls whether or not the value passed down in the variable to receive the answer for any macro prompt, is actually placed into the console panel as the default answer so that it can be accepted by just typing <enter> into the console panel.

The default value is 0.

MAXIMUM_TRASH_SIZE_4D

MAXIMUM_TRASH_SIZE_4D positive integer default 0

the maximum number of Mb that the size of the trash file can be.

If the value is 0, the Trash Bin is **not** limited in size.

The default value is 0.

MIDDLE_DOT_CHARACTER_4D

MIDDLE_DOT_CHARACTER_4D *integer (base 10)* *default 183*

The integer (base 10) value of the character to use as the squared symbol.

MINI_DUMP_LEVEL_4D

MINI_DUMP_LEVEL_4D choice box

MODEL_FOR_TIN_PREFIX_4D

MODEL_FOR_TIN_PREFIX_4D pre*pos *default is* ***"tin "***

This environment variable is used to customize the default model for the tin in the panels for creating triangulations. Text can be defined for prepending and/or appending to tin name to create a default model name from the tin name.

The text for prefixing and postfixing is given in a special form: **pre-text*post-text**

If pretext only, just give the text. If post text is required, precede it by a "*.".

If the environment variable is not set, the default "tin " is used.

MODEL_VIEW_WALKRIGHTS_4D

MODEL_VIEW_WALKRIGHTS_4D	**0**	Use data source in options
	1	Walk-rights for model/view as in up to V3.2

Defines whether the Data source field is used instead of model/view walk-rights used in V3.2.

The default value is 0.

MOVIE_4D

MOVIE_4D executable default xanim

X-Windows only - program for running perspective movie

MS_SEEDFILES_4D

MS_SEEDFILES_4D folder no default

if non blank, the full path name of the folder of Microstation seed files.

MTF_TMP_4D

MTF_TMP_4D anything not defined

If defined, then the temporary files for mtf calculations go to the Windows temp folder. This is to get over a bug in Novell under Windows 95.

MTF_CONSTRUCTION_HINGE_NAME_4D

MTF_CONSTRUCTION_HINGE_NAME_4D text

the default name used to refer to the Hinge string as a link in a new MTF. After a MTF is created, the value can be changed within the MTF, and saved with the MTF.

MTF_DESIGN_LAYER_NAME_4D

MTF_DESIGN_LAYER_NAME_4D text

the default name for the design layer in a new MTF. After a MTF is created, the value can be changed within the MTF, and saved with the MTF.

MTF_EXTRA_START_END_VAL_4D

MTF_EXTRA_START_END_VAL_4D real value

the default extra start/end value for a new MTF. After a MTF is created, the value can be changed within the MTF, and saved with the MTF.

MTF_NAMED_PART_HIGHLIGHT_TEXT_SIZE_4D

MTF_NAMED_PART_HIGHLIGHT_TEXT_SIZE_4D real value

the size of the text to use when the name of the named part is highlighted.

MTF_SEED_INCLUDE_MTFS_4D

MTF_SEED_INCLUDE_MTFS_4D 0 or 1

if non zero, existing MTF files are included (in addition to files of type **.MTF_Seed**) in the choices of files which can be used as a MTF Seed file.

if zero, only files of type **.MTF_Seed** are included in the choices of files which can be used as a MTF Seed file.

MTF_SNIPPET_TEMP_FILE_EXTENSION_4D

MTF_SNIPPET_TEMP_FILE_EXTENSION_4D text

if not blank, the standard **.tmp_mtf** file extension will be replaced with this value.

If blank, the standard **.tmp_mtf** file extension will be used.

For more information on snippets, see 21.5 Defining and Using Snippets.

MTF_SNIPPET_TEMP_FILE_USE_SNIPPET_NAME_4D

MTF_SNIPPET_TEMP_FILE_USE_SNIPPET_NAME_4D 0 or 1

if non zero, the temporary file created after the preprocessing occurs for the snippet is given the same name as the snippet, with the file extension **.tmp_mtf**

If zero, the MTF name is used. Note: If there is more than one snippet, this file will be overwritten.

For more information on snippets, see 21.5 Defining and Using Snippets.

MTF_SNIPPET_TEMP_FILE_USE_SNIPPET_NAME_KEEP_4D

MTF_SNIPPET_TEMP_FILE_USE_SNIPPET_NAME_KEEP_4D 0 or 1

if non zero, the temporary files created after the preprocessing occurs for the snippet are left on the hard drive.

If zero, the temporary files are deleted.

NOTE: If the snippet is compiled, the temporary files are always deleted.

For more information on snippets, see 21.5 Defining and Using Snippets.

MULTI_LINE_TEXT_4D

MULTI_LINE_TEXT_4D *value* 0 or 1 *default 0*

If non zero, a Text Edit Box is used instead of the Input Box and more than one line of text can be typed into the Text Edit Box.

NAME_MAPPINGS_4D

NAME_MAPPINGS_4D filename default names.4d

A name mapping file can be specified which is used to fill out information such as colour, model etc. for given string names.

The name mapping works in two ways. After typing part or all of a string name,

(a) if <enter> is entered, the map file is searched for a match in the first column.
If a match is found, the name, colour, model, style etc. from the other columns in the name mapping file are used to fill out the panel fields.

(b) if a <tab> is entered, the second column is searched for a list of completions (if a * is found, the first column is used for that check) which are displayed in a pop-up. When an entry is selected from the completion list, the name, colour, model, style etc. from the columns in the name mapping file are used to fill out the panel fields.

(c) name mapping file is used to fill out the panel fields.

NEVER_SNAP_ITSELF_4D

NEVER_SNAP_ITSELF_4D **1** try to stop a string snapping to itself during editing
0 let a string snap to itself - behaviour in V3.2

Tries to stop a string snapping to itself during editing.

The default value is 1.

NEW_DRAINAGE_PPF_4D

NEW_DRAINAGE_PPF_4D *full path name of binary ppf file* no default

Points to the file used as the default binary plot parameter file (.drainppf) for the long section plot produced by the panel **Drainage Plot PPF Editor**.

The path name to the binary ppf file can contain $LIB which expands out to the path of the library area. For example, **$LIB/drainage_long.drainppf** points to the file drainage_long.drainppf in the area defined by $LIB_4D.

NEW_DRAINAGE_MELB_PPF_4D

NEW_DRAINAGE_MELB_PPF_4D *full path name of binary ppf file* no default

Points to the file used as the default binary plot parameter file (.melbppf) for the long section plot produced by the panel **Sewer Plot Melbourne Water PPF Editor**.

The path name to the binary ppf file can contain $LIB which expands out to the path of the library area. For example, **$LIB/melb_water.melbppf** points to the file melb_water.melbppf in the area defined by $LIB_4D.

NEW_DRAINAGE_PLAN_PPF_4D

NEW_DRAINAGE_PLAN_PPF_4D *full path name of binary ppf file* no default

Points to the file used as the default binary parameter file (.drainplanppf) for the plan annotation produced by the panel **Drainage Plan Plot PPF Editor.**

The path name to the binary ppf file can contain $LIB which expands out to the path of the library area. For example, **$LIB/drainage_plan.drainplanppf** points to the file drainage_plan.drainplanppf in the area defined by $LIB_4D.

NEW_LONG_SECTION_PPF_4D

NEW_LONG_SECTION_PPF_4D *full path name of binary ppf file* no default

Points to the file used as the default binary plot parameter file (.lplotppf) for the long section plot produced by the panel **Section Long Plot PPF Editor**.

The path name to the binary ppf file can contain $LIB which expands out to the path of the library area. For example, **$LIB/long_section.lplotppf** points to the file long_section.lplotppf in the area defined by $LIB_4D.

NEW_MTF_EDITOR_AUTOPAN_DEFAULT_4D

NEW_MTF_EDITOR_ZOOM_BUFFER_4D 0, 1 *default 1*

if non zero, the default for Autopan is **on** for the MTF ***Left/Right MTF Modifiers*** panel.

if zero, the default for Autopan is **off** for the MTF ***Left/Right MTF Modifiers*** panel.

NEW_MTF_EDITOR_DEFAULT_HEIGHT_4D

NEW_MTF_EDITOR_DEFAULT_HEIGHT_4D *real_value*

real_value is the height in pixels of the MTF ***Left/Right MTF Modifiers*** panel.

The default value is 250

NEW_MTF_EDITOR_DEFAULT_WIDTH_4D

NEW_MTF_EDITOR_DEFAULT_WIDTH_4D *real_value*

real_value is the width in pixels of the MTF ***Left/Right MTF Modifiers*** panel.

The default value is 750

NEW_MTF_EDITOR_PASTED_MODIFIER_COLOUR_4D

NEW_MTF_EDITOR_PASTED_MODIFIER_COLOUR_4D   colour

when you cut/paste the modifiers, the background colour of the pasted rows will remain that colour until you edit them, *i.e.* this option lets you know the pasted rows until you edit them.

NEW_MTF_EDITOR_ZOOM_BUFFER_4D

NEW_MTF_EDITOR_ZOOM_BUFFER_4D   *real_value*

real_value is a percentage.

if *Autopan* is on for the ***Left/Right MTF Modifiers*** panel, then when a command is clicked on in the panel and the region to pan into is not displayed on the view, then the extent displayed it the required amount increased by the *real_value* percentage.

The default value is 5

NEW_PIPELINE_PPF_4D

NEW_PIPELINE_PPF_4D *full path name of binary ppf file*   no default

Points to the file used as the default binary pipeline plot parameter file (.pipelineppf) for the long section plot produced by the panel **Pipeline Plot PPF Editor**.

The path name to the binary ppf file can contain $LIB which expands out to the path of the library area. For example, **$LIB/pipeline_long_section.pipelineppf** points to the file pipeline_long_section.pipelineppf in the area defined by $LIB_4D.

NEW_PLOT_FRAME_PPF_4D

NEW_PLOT_FRAME_PPF_4D *full path name of binary ppf file*   no default

Points to the file used as the default binary parameter file (.plotframeppf) for the plan plot produced by the panel **Plot Frame PPF Editor**.

The path name to the binary ppf file can contain $LIB which expands out to the path of the library area. For example, **$LIB/plot_frame.plotframeppf** points to the file plot_frame.plotframeppf in the area defined by $LIB_4D.

NEW_STRING_CREATES_4D

NEW_STRING_CREATES_4D

NEW_TOOLBARS_VISIBLE_4D

NEW_TOOLBARS_VISIBLE_4D   1, 0   *default*

if non zero, when a project starts up, all the toolbars are checked to see if they are listed in the workspace (visible or invisible) and if the toolbar does not exist, then the toolbar will be displayed.

This is to allow any new toolbars added to toolbars.4d to be automatically displayed so that the user knows that it exists.

if zero, then new toolbars are not displayed. That is, only toolbars listed in the workspace (visible are displayed.

NEW_X_SECTION_PPF_4D

NEW_X_SECTION_PPF_4D *full path name of binary ppf file*   *no default*

Points to the file used as the default binary plot parameter file (.xplotppf) for the cross section plots produced by the panel **Section X Plot PPF Editor**.

The path name to the binary ppf file can contain $LIB which expands out to the path of the library area. For example, **$LIB/cross_section.xplotppf** points to the file cross_section.xplotppf in the area defined by $LIB_4D.

NOP_ZONE_COLOUR_4D

NOP_ZONE_COLOUR_4D *colour*

the colour of the text in the Type column for the commands that aren't in the Fixed, Cut, Fill or Final sections of the modifiers in the **Left/Right MTF Modifiers** panel, *e.g.* Interval commands.

NVALUES_4D

NVALUES_4D filename default nvalues.4d

This points to the file which is used as the file of n value definitions. These are described in 7.6.7 N values and the editing of the n values file is documented in 7.6.7.2 Create/Edit N-Values.

The default is nvalues.4d.

OPENGL_CACHE_4D

OPENGL_CACHE_4D *value* 0 or 1 *default 1*

If non zero, rasters and textures are cached in the graphics card memory for potential speed ups. More memory in the graphics card allows more caching and usually more performance gains.

OPENGL_CACHE_TINS_4D

OPENGL_CACHE_TINS_4D *value* 0 or 1 *default 0*

If non zero, tins are cached in the graphics card memory for potential speed ups. More memory in the graphics card allows more caching and usually more performance gains.

OPENGL_MIPMAP_4D

OPENGL_MIPMAP_4D value 0 or **1**

If ticked, the graphics card down samples for rasters when the image is further away.

Default value is 1.

OPENGL_MIPMAP_BILLBOARDS_4D

OPENGL_MIPMAP_BILLBOARDS_4D value 0 or 1

If ticked, the graphics card down samples for billboards when the image is further away.

OPENGL_MIPMAP_PLAN_IMAGES_4D

OPENGL_MIPMAP_PLAN_IMAGES_4D value 0 or 1 default 1

If ticked, the graphics card down samples plan images when draped onto a tin.

OPENGL_MIPMAP_PROJECTOR_IMAGES_4D

OPENGL_MIPMAP_PROJECTOR_IMAGES_4D value 0 or 1 default 1

If ticked, the graphics card down samples projector images (a projector is for the "hidden" perspective image data object of a super string).

OPENGL_OFFSET_4D

OPENGL_OFFSET_4D value 0 or 1 default 1

If non zero, the coordinates are localised for OpenGL calls. This is to work around problems with some graphics cards that can't handle large coordinates.

OPENGL_VIEW_BACKING_STORE_4D

OPENGL_VIEW_BACKING_STORE_4D value 0 or 1

if 1, a backing store is used.

if 0, a backing store is not used.

For Window Vista/Windows 7, the default is 1. Otherwise the default is 0.

ONSCREEN_KEYBOARD_4D

ONSCREEN_KEYBOARD_4D value 0, 1 or 2

if 0, then no onscreen keyboard comes up when you double click in a panel field.
if 1, when you double click in a panel field that takes typed input, then a dockable onscreen keyboard come up for the user to type the data for the panel field into.
if 2, when you double click in a panel field that takes typed input, then a full screen onscreen keyboard come up for the user to type the data for the panel field into.

The default is 0.

ONSCREEN_KEYBOARD_FONT_SIZE_4D

ONSCREEN_KEYBOARD_FONT_SIZE_4D *number*

if non blank, the pixel size of the font for the onscreen keyboard.
If blank, it defaults to the normal system font size.

The default is the normal system font size.

ONSCREEN_KEYBOARD_LAYOUT_4D

ONSCREEN_KEYBOARD_LAYOUT_4D value 0, or 1

if 0, then when there is an onscreen keyboard, it comes up with a full keyboard.
if 1, then when there is an onscreen keyboard, it comes up with just a numeric keyboard.

The default is 0.

OVERFLOWING_TRASH_MODE_4D

OVERFLOWING_TRASH_MODE_4D choice box *default* Auto manage

the action to take when the trash bin exceeds the maximum trash bin size.

If **Auto manage,** the oldest files in the trash bin are deleted until the new item can fit in the trash bin.

if **Auto empty,** files in the trash bin are deleted.

Note - if a large model or tin is deleted and it is bigger than the maximum trash bin size, the user is alerted and asked to decide if they want the element to go in the trash bin anyway, or if they want to permanently delete it.

The default value is Auto manage.

PAN_MODE_4D

PAN_MODE_4D	**0**	Use standard pan for pan, pans on views
	1	Use pan deltas for pan, pans on views

Sets whether pan or pan delta is used for the pan and pans buttons on views.

The default value is **0**.

PATTERNS_4D

PATTERNS_4D filename default patterns.4d

Points to the file used to define the patterns used for super string fills.

The default is patterns.4d.

PDF995_TIME_LIMIT_4D

PDF995_TIME_LIMIT_4D *number*

the number of seconds to wait for PDF995 to finish producing the current PDF file.

PICK_ONLY_ON_SELECTS_4D

PICK_ONLY_ON_SELECTS_4D

PLAN_TABLE_SETTINGS_4D

PLAN_TABLE_SETTINGS_4D	**0**	Don't allow setting by model*default 0*
	1	Allow setting by model

Trial only - allow the drawing of z-values, vertices etc. on the plan view to be set by individual models rather than for all models on the view.

The default value is 0.

PLOTTER_4D

PLOTTER_4D script/program no default

This environment variable points to a script/program which can be fired up whenever a plot is generated. The name of the plot is given as the first script parameter of the script.

If the parameter **Send plots** on the 'System Settings' tab of the menu option **Utilities** => **Defaults** is set to **yes**, the plotter script is run as each plot is created.

If more than one plot is created by an option (e.g. x plot) then the script is called separately for each of the plots.

An example of a script to send the plot to port lpt1 for Windows NT would be

```
@echo off
copy %1  lpt1
```

An example for Windows NT which looks for hp files is

```
@echo off
echo.
echo --------------------------------------------------
:next_file
if "%1" == "" goto done
echo %1 | find /I ".hp" > nul
if ERRORLEVEL 0 if not ERRORLEVEL 1 goto hp_plotter
echo Plotting file %1
shift
goto next_file
:hp_plotter
echo Plotting %1 to HP plotter
shift
goto next_file
:done
echo --------------------------------------------------
```

PLOTTER_MAPPING_4D

PLOTTER_MAPPING_4D filename default pmf.4d

This points to the file which is used as the default plotter mapping file. The format for the plotter mapping file is described in the section 43.3 Mapping 12d Colours to Pens and RGBs in the Appendix 39 Setting Up & Configuring 12d.

The default is pmf.4d.

PLOTTERS_4D

PLOTTERS_4D filename default plotters.4d

This points to the file which contains user defined plotters. The format for the file of user defined plotters is described in the section 43.2 Defining Plotters - Plotters.4din the Appendix 39 Setting Up & Configuring 12d.

The default is plotters.4d.

PLOT_SYMBOLS_4D

PLOT_SYMBOLS_4D filename default plotsymb.4d

Points to the file used to define the symbols used in long and cross section plots.

The default is plotsymb.4d.

POLYPOLYLINES_4D

POLYPOLYLINES_4D	**0**	Don't use speed ups	*default 2*
	1	Intermediate speed ups	
	2	Faster speed ups.	

A few experimental techniques are available for speeding up the drawing on the screen.

For some specific things such as fast contours, fast mesh and point crosses, the techniques can result in some redraws being up to three times faster.

The default env.4d shipped with ***12d Model*** has this set to 2.

PREVIEW_VIEW_4D

PREVIEW_VIEW_4D *text*

if non blank, the name of the view whose image is dumped on exiting the project. The image is used as the project preview.
If blank then the last active view is used.

PRINTER_4D

PRINTER_4D script/program no default

Points to a script or program which can be fired up whenever a report is generated. The name of the report is given as the first parameter of the script.

If the parameter **Print reports** on the 'System Settings' tab on the menu option **Utilities => Defaults** is set to **yes**, the printer script is run as each report is created.

PROCESSOR_AFFINITY_4D

PROCESSOR_AFFINITY_4D

PROJECT_DETAILS_4D

PROJECT_DETAILS_4D file name

If non blank, the full path name of the file of project details file to use for new projects.

PROJECT_NAMES_4D

PROJECT_NAMES_4D	**0**	Use long filenames for internal files	*default*
	1	Use short names (8.3 format)	

The use of long or short extension names for internal 12d Model files for items such as models, tins, templates etc. is controlled by the environment variable PROJECT_NAMES_4D.

Opening a new project with the short name format, project-name.4dp, will automatically create a short name project (i.e. a project using the short extension names - 3 characters after the **.**).

The default value is 0.

PROJECTIONS_4D

PROJECTIONS_4D filename default carto.4d

This points to the file which is used as the file of projections. These are described in 7.6.6 Projections and the editing of the projections file is documented in 7.6.6.2 Create/Edit Projection.

The default is carto.4d.

PROMPT_ON_VIEW_CLOSE_4D

PROMPT_ON_VIEW_CLOSE_4D **0**, 1

If non zero, when a view is closed/deleted, a prompt asks for a confirmation of deleting/closing. If zero, no confirmation is requested when a view is closed/deleted.

The default is 0.

PURGE_TRASH_DAYS_4D

PURGE_TRASH_DAYS_4D *number_of_days* *default 0*

number_of_days is a positive integer and is the number of days before the Trash Bin is automatically purged of tins and models.

If *number_of_days* is zero, the Trash Bin is **not** purged.

The default value is 0.

RECENT_PROJECTS_4D

RECENT_PROJECTS_4D positive integer default 20

The maximum number of accessed projects displayed in the Projects list when ***12d Model*** starts up.

REMEMBER_MTF_PANEL_SIZE_4D

REMEMBER_MTF_PANEL_SIZE_4D 0 or 1

if non zero, when a MTF panel is placed on the screen and closed with Finish, its final position and size are recorded and used when the panel is opened again.

REPORT_HEADER_4D

REPORT_HEADER_4D	0	Report files have no header page
	1	Some header information is used
	2	A full header page is produced.*default 2*

Controls the amount of header information in reports.

For the cases 1 and 2, the header information includes

(a) the 12d Model Project

(b) the name of the user

(c) the organization

(d) the current date

(e) the current report file name

The default value is 2.

RUN_MACROS_FILE_4D

RUN_MACROS_FILE_4D filename default macros.4d

if non blank, the full pathname of the file of macros (one per line) that are run when a new project is created.

Note: This is only run for new projects. RUN_PROJECT_MACROS_FILE_4D is run when opening an existing project.

The default is macros.4d

RUN_PROJECT_MACROS_FILE_4D

RUN_PROJECT_MACROS_FILE_4D filename default project_macros.4d

if non blank, the full path name of the file of macros (one per line) that are run when an existing project is opened.

Note: This is only run when opening existing projects. RUN_MACROS_FILE_4D is run when creating a new project.

The default is project_macros.4d

SDR_DISPLAY_V8_FORMAT_4D

SDR_DISPLAY_V8_FORMAT_4D

SETUPS_FILE_4D

SETUPS_FILE_4D filename default setups.4d

file setting up the initial screen layout for new projects. This is not to be confused with SET_UPS_4D which is where the standard Set Up files are installed. See SETUPS_FILE_4D.

The default is setups.4d. See 39.2.7.1 Set Ups File (setups.4d).

SET_UPS_4D

SET_UPS_4D folder HOME_4D\11.00\set_ups

folder for storing the standard Set Up files installed by 12D Solutions.

The default is **HOME_4D\11.00\set_ups**.

SEWER_PPF_4D

SEWER_PPF_4D path name to .ppf file no default

points to the file used as the default plot parameter file (.ppf) for the panel **new plot sewer network**.

SHARE_CHECK_INTERVAL_4D

SHARE_CHECK_INTERVAL_4D positive integer *default 0*

if non zero, the number of seconds between checks to see if any of the shared tins or models added to the project have been modified.

SHARE_LOCKS_FOLDER_4D

SHARE_LOCKS_FOLDER_4D folder default no folder

if non blank, the full path name of the folder used to keep lock files for shares.

If no folder is given, the locking files are stored inside the project.

SHARE_MAP_FILE_4D

SHARE_MAP_FILE_4D filename default blank

if non blank, the full path name of the map file to be applied to shared models.

SHARED_ELEMENT_COLOUR_4D

SHARED_ELEMENT_COLOUR_4D colour default blue

the colour to use in a list of tins/models for showing shared tins/models. That is, the tins/models that have been added to the project as shared tins/models are shown in this colour.

Setting the colour to black will disable this feature.

SHARING_CACHE_4D

SHARING_CACHE_4D folder default no folder

if non blank, the full path name of the folder that is used to store the models and tins and tins that are shared into the project. Each time the project is started, each model and tin that is shared into the project is checked to see if has changed since the last time it was used in the project, and if has not changed then the copy in the **Sharing Cache Folder** is used. This can greatly improve performance when the models and tins are being shared into your project from across the network.

If no folder is given, no chacing of models or tins is done.

SHARING_ELEMENT_COLOUR_4D

SHARING_ELEMENT_COLOUR_4D colour default rgb 255,165,0

the colour to use in a list of tins/models for showing the tins/models in the project that you are allowing to be shared. That is, those tins/models in the project that the user has allowed others to share are shown in this colour.

Setting the colour to black will disable this feature.

SHEET_SIZES_4D

SHEET_SIZES_4D filename default sheets.4d

For plot frames, long and x plots, the overall size of the plot sheet can be given by a pop-up containing defined sheet size.

The sheet size name and width and heights can be specified by the user in a file named sheets.4d which is in the normal Set Up areas, or is pointed to by the environment variable SHEET_SIZES_4D.

The default is sheets.4d. See 39.2.7.4 Sheet Sizes File (sheets.4d).

SHOW_OPTION_EXECUTION_TIME_4D

SHOW_OPTION_EXECUTION_TIME_4D integer 0

if non zero, the time taken for an option to run is written to the output window.

If zero, the information is not written to the output window.

SHOW_PATHS_4D

SHOW_PATHS_4D anything

If non blank, then when 12d Model fires up the actual file names defined by any environment variables are written to the output window. This is useful for debugging.

SHOW_PROJECT_DETAILS_4D

SHOW_PROJECT_DETAILS_4D 0, **1** default 1

If non zero, *the* Edit Project Details *panel is displayed when a new project is created.*

SHOW_TITLE_VARIABLES_4D

SHOW_TITLE_VARIABLES_4D	**0**	any $variable not used is left as blank
	1	any $variable not used is shown as $variable
	2	all $variable's are plotted with no substitution

This is used for debugging the title block file.

If 0, any $variable not used in the title blank file is left as blank.
If 1, any $variable not used is shown as $variable.
If 2, all $variable are plotted with no substitution.

The default value is 0.

SHOW_VISTA_VIRTUAL_STORE_PATHS_4D

SHOW_VISTA_VIRTUAL_STORE_PATHS_4D tick box

if ticked, then for Microsoft Vista, when 12d Model fires up, the actual file names defined by any environment variables are written to the output window. This is useful for debugging due to the fact that Vista may put file sin strange places.

SNIPPET_ZONE_COLOUR_4D

SNIPPET_ZONE_COLOUR_4D *colour* *default light grey (RGB(155, 140, 150)*

the colour of the text in the *Type* column for the *Snippet* command in the *Modifiers* section of the ***Left/Right MTF Modifiers*** *panel*.

SORT_COLOURS_BY_POPUP_NUM_4D

SORT_COLOURS_BY_POPUP_NUM_4D 0,**1** default 1

Controls the display order of the colours in the **Select Colour** popup that you get when you click on a **Colour** icon.

If **SORT_COLOURS_BY_POPUP_NUM_4D** is **1,** the colours in the **Select Colour** popup are sorted by the Pop-up Number given for the colours in the **Colours.4d** file.

If **SORT_COLOURS_BY_POPUP_NUM_4D** is **0,** the colours in the **Select Colour** popup are sorted alphabetically.

The default value is **1**.

SPECIAL_OFFSET_CHAINAGES_4D

SPECIAL_OFFSET_CHAINAGES_4D

SPEED_TABLES_4D

SPEED_TABLES_4D

SQUARE_CHARACTER_4D

SQUARE_CHARACTER_4D *integer (base 10)* *default 178*

The integer (base 10) value of the character to use as the squared symbol.

STATION_PREFIX_4D

STATION_PREFIX_4D *text* There is no default.

The default prefix to use in the Survey Data Setup panel or to use if no prefix has been set for the project.

There is no default value.

SUPER_ADVANCED_MODE_4D

SUPER_ADVANCED_MODE_4D

SUPER_ALIGNMENT_STYLE_4D

SUPER_ALIGNMENT_STYLE_4D filename default astyles.4d

Points to the file used to define the symbology used for super alignments.

The default is astyles.4d.

SUPER_STRINGS_4D

SUPER_STRINGS_4D	**1**	Use the super string
	0	Don’t use the super string.

Controls whether the super string is accessible or not.

The default value is 1.

SYMBOLS_4D

SYMBOLS_4D filename default symbols.4d

Points to the file used to define the symbols used for super strings.

The default is symbols.4d.

SYSTEM_NAMES_4D

SYSTEM_NAMES_4D	1	Only use the longer names
	2	Only use the 8.3 file names
	3	Use short names first then look for a long name.

For file name compatibility with DOS 8.3 format, the default names for all Set Up files can be restricted to just short names (8.3), long names, or short and then long.

The default value is 3.

TRANSFORMATIONS_4D

TRANSFORMATIONS_4D   filename   default 7params.4d

This points to the file which is used as the file of seven parameter transformations. These are described in 7.6.8 7 Parameters and the editing of the transformations file is documented in 7.6.8.1 Create/Edit 7 Parameters.

The default is 7params.4d.

TEXTSTYLE_MAPPINGS_4D

TEXTSTYLE_MAPPINGS_4D   filename   default textstyle_names.4d

file of textstyles favourites definitions

The default is textstyle_names.4d

TEXTSTYLES_4D

TEXTSTYLES_4D   filename   default textstyl.4d

file of textstyles definitions

The default is textstyl.4d. See 40.3 Textstyles and Fonts.

TEXTURE_MAP_4D

TEXTURE_MAP_4D   filename   default texture_map.4d

This points to the file which defines the tables of texture mappings. These are described in 12.13.4 Texture Map Edit.

The default is texture_map.4d.

TICK_DRAW_CROSS_4D

TICK_DRAW_CROSS_4D	**0**	Nothing for off
	1	Cross for off

The default value is 0.

TIN_VIEWPORT_CLIP_4D

TIN_VIEWPORT_CLIP_4D	**0**	Drawing tins, fast contours as per V3.1
	1	A speed up for drawing tins, fast contours

Use some experimental techniques for speeding up the drawing of tins, fast contours.

The env.4d file shipped with ***12d Model*** has this set to 1.

TOOLBAR_DUPLICATE_MODE_4D

TOOLBAR_DUPLICATE_MODE_4D   choice box

This environment variable manages what to do if toolbars with the same name are read from a *toolbars.4d* file. The choices are

Do nothing - accept all toolbars of the same name
Take first - accept only the first instance of the toolbar
Take last - accept only the last instance of the toolbar

TOOLBARS_4D

TOOLBARS_4D   filename   default toolbars.4d

file of toolbar definitions and names

The default is toolbars.4d. See 41.3 User Defined Toolbars

TP_STAKEOUT_PATH_4D

TP_STAKEOUT_PATH_4D folder no default

if non blank, the full path name of the folder of TP Stakeout files.

TRIMESH_TO_DWG_MODE_4D

TRIMESH_TO_DWG_MODE_4D Do not write OR Faces OR Polyface mesh

if **Do not write**, trimeshes are not written out to a DWG file.

If **Faces**, a trimesh is written out to DWG file as individual 3DFaces.

If **Polyface mesh**, a trimesh is written out to a DWG file as a Polyface mesh.
Note that a Polyface mesh has only one colour.

TYPED_UNITS_MODE_4D

TYPED_UNITS_MODE_4D **0** International units only e.g. f and F are both International fee
1 USA units only e.g. f and F are both US feet
2 Mixed e.g. f is International feet and F is US feet.

Controls the typed input units for feet - international and/or US

The default value is 0.

UNDO_4D

UNDO_4D 0 Don't allow undo's
1 Allow undo/redos *default 1*

Undo and Redo is available for most operations from ***12d Model*** V3.1 onwards. The availability of Undo/Redo facility is controlled by the environment variable:

The default value is 1.

USAGE_LOG_4D

USAGE_LOG_4D folder *no default*

when set, log files of the form

<log file folder>\(<dongle> <user> <computer> <time stamp> <process ID>.log)

will be created in the given folder.

That is, the log files will all be in the one folder.

USAGE_LOGS_4D

USAGE_LOGS_4D folder *no default*

when set, log files of the form

<log file folder>\<dongle>\<user>\<computer>\(<time stamp> <process ID>.log)

will be created in the given folder.

That is, the log files will be in subfolders \dongle\<user>\computer>.

USE_ALL_USERS_PROFILE_4D

USE_ALL_USERS_PROFILE_4D 0, 1

If no-zero then the file env.4d and the folders user and user_lib are looked for in ***\Documents and Setting\All Users***.

USE_BACKUPS_4D_FOLDER_4D

USE_BACKUPS_4D_FOLDER_4D 0, **1** default 1

If non-zero: in the places where ***12d Model*** creates a backup file when a new file is created (using the file endings 1, 2, ... for the old file) then the backup file is placed in the folder backups.4d.

Backups are created for options such as env.4d editor, mtf editor, boxing editor, survey.4d editor, whenever the "Write Setup File(s)" is displayed, plus the macro calls Backup_version_file and Restore_version_file.

The backups.4d folder is created in the project working folder (that is, in the folder containing the .project folder).

USE_DENSITY_CHECKS_4D

USE_DENSITY_CHECKS_4D	0	Don't do density checks	
	1	Do density checks	*default 1*

If non zero then for a new project, the default for **Use density drawing** in the **Defaults** panel is ***tick***.

If zero then for a new project, the default for **Use density drawing** in the **Defaults** panel is ***no tick***.

For more information on ***density drawing***, see Use density drawing tick box.

Note: This setting is only applicable to the 250M version of ***12d Model.***

USE_DISPLAY_MANY_SNAP_4D

USE_DISPLAY_MANY_SNAP_4D

USE_ECW_SERVER_4D

USE_ECW_SERVER_4D

USE_IMAGE_SERVER_4D

USE_IMAGE_SERVER_4D

USE_NAMES_COMMENT_4D

USE_NAMES_COMMENT_4D

USE_NEW_LINESTYLE_LIST_BOX_4D

USE_NEW_LINESTYLE_LIST_BOX_4D value 0, 1

If non zero, the linestyle and symbol pop-up lists are in scrolling boxes so they don't run over the bottom of the screen. The list of linestyles/symbols is displayed in a tree structure with the Groups as the nodes of the tree.

If zero, the linestyle and symbols lists are one long list with each Groups being an item in the list. The list may get too long to fit on the screen.

The default value is 1.

USE_NEW_SELECT_4D

USE_NEW_SELECT_4D

USE_SUPER_STRINGS_4D

USE_SUPER_STRINGS_4D

USE_TRASH_BIN_4D

USE_TRASH_BIN_4D	0	Don't use the Trash Bin	*default 1*
	1	Use the Trash Bin	

The Trash Bin is used to store copies of deleted tins and models, and cleaned models.The items can be retrieved from the Trash Bin. The Trash Bin is automatically purged of contents after PURGE_TRASH_DAYS_4D.

The default value is 1.

USE_VALIDATION_COLOURS_4D

USE_VALIDATION_COLOURS_4D *value* 0 or 1 default 1

If 1, when a panel field fails to validate, the panel field is filled with the colour given by

USER_4D

USER_4D folder HOME_4D\11.00\user

user folder containing Set Up files

The default is **HOME_4D\11.00\user**.

USER_LIB_4D

USER_LIB_4D folder HOME_4D\11.00\user_lib

user library folder for input files such as mapping, template and macros (4DML's).

The default is **HOME_4D\11.00\user_lib**.

USER_OPTIONS_4D

USER_OPTIONS_4D file default usermenu.4d

To help customise 12d Model, the walk-right menu **User** on the main 12d Model menu, can be user defined. The text for each button of **User**, plus the action taken when the button is selected is user specified. Any of the buttons can include further walk-right menus.

The definitions for the menus on **User**'s is given in the file *usermenu.4d* which is searched for in the standard Set Up areas (local, USER_4D, user, set_ups) or set by the environment variable USER_OPTIONS_4D.

if USER_OPTIONS_4D is blank, usermenu.4d is searched for in the standard search path for *User* files.

if USER_OPTIONS_4D is not blank, then it is used as the full path name of the user supplied file of definitions for user defined menus.

The default is usermenu.4d. See 41.2 User Defined Menus

V7_TITLE_BLOCKS_4D

V7_TITLE_BLOCKS_4D value 0 or **1** *default* 1

if non zero, the title block file is in 12da format.
If zero, the title block is in the pre V7 title block .tf format

VALIDATION_FAIL_COLOUR_4D

VALIDATION_FAIL_COLOUR_4D *text*

If not blank, the colour to fill the panel field with when there is a validation error for the field. The *text* is either a colour name, a colour number or RGB(x,y,z).

This is only used if USE_VALIDATION_COLOURS_4D is non zero (or at least not set to zero since the default is one).

If blank, the default colour is RGB(255,72,72),

VEHICLE_PATH_4D

VEHICLE_PATH_4D folder no default

The DOS Version of Vpath is no longer supported.

The environment variable VEHICLE_PATH_4D points to the folder where the Dos version of the Vpath executable is located. Note that WINDOWS_VEHICLE_PATH_4D points to the Windows version.

VIEW_BITMAP_BUTTONS_4D

VIEW_BITMAP_BUTTONS_4D	0	User text for view buttons
	1	Use icons for view buttons

The default value is 1

VIEW_BUTTONS_4D

VIEW_BUTTONS_4D	0	No view buttons are displayed
	1	The view buttons are displayed

If non zero, then menu items (view buttons) are displayed on the views (as icons or text).
If zero, menu items (view buttons) are not displayed on the views.

The default value is 1

Note: displaying *view buttons* as icons or text is controlled by VIEW_BITMAP_BUTTONS_4D

WARP_CURSOR_HIDE_4D

WARP_CURSOR_HIDE_4D	**0**	Don’t hide the cursor before moving it
	1	Hides the cursor before moving it

Controls whether the cursor is hidden before moving - only needed on some computers.

If non zero, don’t hide the cursor before moving it.
If zero, hides the cursor before moving it.

The default value is 0.

WEB_SEARCH_4D

WEB_SEARCH_4D	search engine web address	www.google.com

web address of the search engine that is fired up from the 12d option

Help =>12d on the Web =>Search the web.

WEED_TOLERANCE_4D

WEED_TOLERANCE_4D	0	Don’t weed
	real	Weed tolerance

used in Alignment and Super strings so that when arcs have been chord-to-arc’d, the resulting points are weeded so that no point is closer than the weed tolerance. Is is also used in Apply and Apply MTF so that no cross sections are closer than the weed tolerance.

WIBU_4D

WIBU_4D 1, 0

if non zero, search for 12d Wibu dongles.
If zero, don’t search for Wibu dongles

WIBU_DONGLE_4D

The environment variable can have a number of parameters to control the use of Wibu network and stand alone dongles.

(a) WIBU_DONGLE_4D -local OR -no_local

usually used when there is a network dongle and need to know if any stand alone Wibu dongle is to be looked for on the computer that the user is on.

if **-local**, then a stand alone Wibu dongle is looked for on the computer the user is on.
If **-no local**, then no stand alone Wibu dongle is looked for on the computer the user is on.

For example WIBU_DONGLE_4D - local or -no local

(b) WIBU_DONGLE_4D -no network OR -network first OR -network last

need to specify if any network Wibu network dongle is to be looked for.

if -**no network**, then no Wibu network dongle is looked for.
If **-network first**, then a Wibu network dongle is looked for before a stand alone local dongle.
If **-network last**, then a Wibu network dongle is looked for after looking for a stand alone local dongle.

For example WIBU_DONGLE_4D -no_network *or* - network_first *or* -network_last

(c) WIBU_DONGLE_4D -login_retries number of retries to find a Wibu dongle

the number ***nn*** of retries to make when searching for a Wibu dongle and it is not found.

For example WIBU_DONGLE_4D -login_retries 3

(d) WIBU_DONGLE_4D -login_wait ss

the number of seconds ***ss*** to wait between retries to find a network Wibu dongle.

For example WIBU_DONGLE_4D -login_wait 5

(e) WIBU_DONGLE_4D -no_dongle

if the parameter -no_dongle is found, then no search is made for a Wibu dongle.

For example WIBU_DONGLE_4D -no_dongle

(f) WIBU_DONGLE_4D -debug

if the parameter -debug is found, then Wibu dongle debug information is written to the

For example WIBU_DONGLE_4D -debug

See the separate notes on installing a network dongle for more details.

WIBU_IPADDR

WIBU_IPADDR list of IP addresses and/or computer names

if **non blank**, a list of IP addresses and/or computer names to search for a 12d Wibu network dongle. The items in the list are separated by commas

If **blank**, search the entire network for a 12d Wibu network dongle

WINDOWS_PRINT_MODE_4D

WINDOWS_PRINT_MODE_4D *value*

value is an integer representing "bit mask".

If **bit 1** set (contributes 1 to *value*): Windows 2000/XP print dialog means PrintDlgEx otherwise if bit 1 is not set, then PrintDlg.
if **bit 2** set (contributes 2 to *value*): Use exclusive access to the printer to force direct printing to the printer (may need Printer admin access)
if **bit 3** set (contributes 4 to *value*): Use an intermediate print file so that printing is first done to a file and then that file is submitted to the printer

So the legal values are from 0 to 7

WINDOWS_PRINTER_SET_DOCUMENT_PROPERTIES_4D

WINDOWS_PRINTER_SET_DOCUMENT_PROPERTIES_4D

WINDOWS_PRINTERS_4D

WINDOWS_PRINTERS_4D	0	Don’t allow Windows printers
	1	Enable Windows printers

The default value is 1.

WINDOWS_VEHICLE_PATH_4D

WINDOWS_VEHICLE_PATH_4D folder no default

if non blank, the full path name of the folder where the Windows program Vpath is located.

Note - Vpath is the Vehicle Turning Path program written by Queensland Department of Transport and Main Roads that is provided free of charge.

WINTER_4D

WINTER_4D path name to WINTER program

12d Model no longer uses the external Winter program and so this environment variable is no longer used.

Points to the WINTER program for calculation N-values for Australia.

WINTER_DATA_4D

WINTER_DATA_4D folder no default

Folder containing the Winter data of N-values for Australia.

WINTER_USE_NEW_METHOD_4D

WINTER_USE_NEW_METHOD_4D value 0, 1 default 1

if tick, re-reading the Winter data is avoided and this speeds up the calculation for the Winter interpolations.

WORKSPACE_FILE_4D

WORKSPACE_FILE_4D path name

if non blank, the full path name of the workspace file for new projects.

WRITE_ALL_PLOT_PARAMETERS_4D

WRITE_ALL_PLOT_PARAMETERS_4D 0 Only write out the plot parameters that are used in the ppf.
1 Write out all plot parameters to a ppf file

Controls whether all plot parameters are written out to a ppf file or just those that have been used in the ppf file.

The default value is 1.

X_SECTION_PPF_4D

X_SECTION_PPF_4D *path name to .ppf file* no default

points to the file used as the default plot parameter file (.ppf) for the panel **section X plot**.

The path name to the .ppf file can contain $LIB which expands out to the path of the library area. For example, **$LIB/cross.ppf** points to the file cross.ppf in the area defined by $LIB_4D.

ZOOM_ORIGIN_DYNAMIC_4D

ZOOM_ORIGIN_DYNAMIC_4D **0** The centre of the view is the zoom centre point
1 The point you pick is the zoom centre point

Controls the origin of the dynamic zoom.

If non zero then the point selected in the view to indicate which view to dynamically zoom (and to be the zoom-in, zoom-out definition point) becomes the point to dynamically zoom about. Whilst the dynamic zoom is running, another point can be selected to become the new zoom origin.

The default value is 0.

ZOOM_PAN_DYNAMIC_4D

ZOOM_PAN_DYNAMIC_4D **0** Dynamic pan is not the default
1 Dynamic pan is the default for pan

If non zero then ***dynamic pan*** is the default for the pan options. The middle mouse button (or 'd') is not required to place the pan option in dynamic mode. In fact, it would then toggle it off.

The default value is 0.

Notes

(a) If any of the environment variables (pointing to files) are not set, or the file pointed to does not exist, then 12d Model searches for the default files in a number of locations. The search order is given at the beginning of this Appendix.

(b) The eagle pen mapping file, **eagleplt.emf** is also searched for in the same order as the default files.

(c) When an **existing** project is opened by 12d Model, the setups.4d and defaults.4d are not used.

Please continue to the next section 39.4.2 Setting Environment Variables.

39.4.2 Setting Environment Variables

The simplest method for setting the required environment used in 12d Model is to place then into a file, called the environmental file (default name env.4d).

Environment Variables File

It is possible to define a file which sets the value for one or more environment variables.

The file format consists of one line for each environment variable being set and each line contains the environment variable name, followed by one or more spaces and then the value for the environment variable:

environment_variable_name *value*

The value of any environment variable given in the environment variable file **overrides** any other definition for that environment variable.

The file used as the environmental variable file is checked for in the following order - as soon as a file is found, the search terminates and that file is used:

1. as a command line argument when 12d Model is fired up. The syntax is

   12d -env path_name

   where path_name is the full path name of the file
2. pointed to by the environment variable ENVIRONMENT_4D

   ```
   ENVIRONMENT_4D    file    // file of environment variables
                             // default env.4d
   ```
3. a file called **env.4**d which is searched for in the standard Set Up areas (local, USER_4D, user, set_ups).

An example of an environment variable file is

```
EDITOR_4D         te                   // set the text editor used
SHOW_PATHS_4D     1      // show file names set by environment variables
SHEET_SIZES_4D    c:\standards\shfile
```

Setting Environment Variables not in a File

Environment variables can be set in **Windows NT** for a user from the *Environment* tab on the *System Properties* panel brought up by clicking on the *System* icon in the *Control Panel* of Windows.

To bring up the *Control Panel*, click on *Start* in the Windows Task bar, walk right on *Settings* and then click on *Control Panel*.

Please continue to the next section 39.2.7.1 Set Ups File (setups.4d).

39.5 12d Model Options Map

The ***12d Model*** menu map shows the menu structure for all the sub-menus on the main ***12d Model*** menu and the menus on the views.

The map is designed to be printed out for users who wish to affix it to a wall.

The PDF file of the menu map on the ***12d Model*** Installation CD in the folder

Documentation***12d Model*** menu map

Or in the *Updates* section of the web site www.12d.com.

Please continue to the next section 39.6 Monitoring 12d Model Usage.

39.6 Monitoring 12d Model Usage

39.6.1 Overview of 12d Usage

12d Model usage can be monitored for single-user and network licenses (or a mix of both), whether installed on a single computer, a local-area network, or a wide-area network across different time zones.

Monitoring the usage is a 3 stage procedure:

1. Generating individual .log files for each ***12d Model*** session run.
2. Consolidating all the completed .log files into a CSV file.
3. Reporting the usage details, using the CSV file as input.

The 1st stage is achieved by setting an environment variable in your env.4d file(s).

The last 2 stages are achieved by running the supplied program: 12d_usage.exe.

Continue to the next section 39.6.2 Generating the .log files or return to 39.6 Monitoring 12d Model Usage.

39.6.2 Generating the .log files

By setting one of two possible environment variables in the env.4d file(s) used for your ***12d Model*** sessions, uniquely named .log files can be generated in a specified folder on your network.

For example, by adding the lines:

usage_log_4d   F:\12d model usage

or

usage_logs_4d   F:\12d model usage

to your env.4d file(s), then each time a ***12d Model*** session is run, a .log file will be created in the "F:\12d model usage" folder. This folder, referred to as the <log file folder>, can be anywhere on your network and can have any name, ***but it must be created beforehand***, and all 12d Model users must have write-access to it.

In the first case, using variable "usage_log_4d", log files will be generated in the form:

<log file folder>\(<dongle> <user> <computer> <time stamp> <process ID>.log)

In the second case, using variable "usage_logs_4d", log files will be generated within sub-folders of the form:

<log file folder>\<dongle>\<user>\<computer>\(<time stamp> <process ID>.log)

Note that the <time stamp> represents the start time of the ***12d Model*** session in GMT (also known as UTC or Zulu Time).

It does not matter which variable you decide to use, and you can even use a mix of the variables, if you have more than one env.4d file. However, regardless of which variable you use, it is recommended that the <log file folder> is the same for all ***12d Model*** users on your network so that all ***12d Model*** sessions can be easily included in the reports.

The **.log** file generated in the <log file folder> for each individual ***12d Model*** session run on your network, contains all the information required about that session. While the session is still running, the log file will be updated at a minimum of every 75 seconds, with the time that the session has been active. At the close of the ***12d Model*** session, the log file is again updated and is set as "completed".

Continue to the next section 39.6.3 Installing 12d_usage.exe or return to 39.6 Monitoring 12d Model Usage.

39.6.3 Installing 12d_usage.exe

You can put the 12d_usage.exe program wherever you like on your network. Then, simply create a shortcut to the program on your desktop, making sure the shortcut properties are set as follows:

Target:	<path to program location>\12d_usage.exe
Start in:	<path to Working Folder>

The Working Folder is the folder that you run the 12d_usage.exe program from (it does not have to be the same as the location of the 12d_usage.exe file). After running the program for the first time, a file named 12d_usage.defaults will be created in the Working Folder. This file is used to save your settings from the last run of the program and to load your settings for the next run of the program. In addition, you might also wish to use the Working Folder to store your ***12d Model*** Usage report files (although you are free to store them elsewhere).

Continue to the next section 39.6.4 Running 12d_usage.exe or return to 39.6 Monitoring 12d Model Usage.

39.6.4 Running 12d_usage.exe

To run the 12d_usage program, simply double-click on the shortcut icon you created at install time. This will open up a DOS window from where the program's keyboard-driven menu system can be run.

From the Main menu within the 12d_usage program, your keyboard-driven menu options are:

C	Bring up the Consolidate menu
R	Bring up the Report menu
Q	Exit (quit) the 12d_usage program

Continue to the next section 39.6.5 Consolidating the .log files or return to 39.6 Monitoring 12d Model Usage.

39.6.5 Consolidating the .log files

After a while, you should notice that a lot of .log files are being generated in your specified <log file folder>. To consolidate all these .log files into a single Comma-Separated-Variable (CSV) file, you will need to run the 12d_usage.exe program, and press the “C” key to bring up the Consolidate menu.

From the Consolidate menu, your keyboard-driven menu options are:

F	Specify the <log file folder>
O	Specify the output CSV file name
T	Toggle whether to delete or rename consolidated log files
C	**Consolidate 12d Model session log files**
Q	Exit the Consolidate menu and return to the Main menu

When specifying the name of the CSV file you wish to consolidate your log files to (using the “O” key), if you want a **.csv** file extension (to allow the file to be easily loaded into MS?Excel for your own custom reports, for example), you must include the .csv in the file name. It is recommended that all users of the 12d_usage program consolidate log files to the same CSV file every time. Newly consolidated logs are always appended to an existing CSV file. If there is to be more than one user of the 12d_usage program, it might be a good idea to specify the CSV file to be in the <log file folder> or the Working Folder (or at least, some folder that all 12d_usage users can access).

You also need to specify how to tidy up the log files, after they have been consolidated, so that they are not consolidated more than once. Your options here are to rename or to delete the log files (the “T” key toggles between rename and delete). If you choose to rename them, they will be renamed with a .logc extension. If you choose to delete them, they will be lost forever, and the only record you will have of your consolidated sessions will be contained in the CSV file. As such, you should be very careful not to delete your CSV file.

Pressing the “C” key will then go ahead and consolidate the log files. Note that consolidation will only take place on “completed” log files (or on log files that have not been updated for more than 24 hours - to handle the case of a power failure, for instance).

Continue to the next section 39.6.6 Reporting on 12d Model Usage or return to 39.6 Monitoring 12d Model Usage.

39.6.6 Reporting on 12d Model Usage

After creating or adding to your consolidated CSV file, you can press the “R” key from the Main menu, in order to bring up the Report menu.

From the Report menu, your keyboard-driven menu options are:

I	Specify the input CSV file name (generated from the Consolidate menu)
O	Specify the output report file name
M	Toggle whether to overwrite or append to existing report file
D	Add/Remove Dongle constraints
C	Add/Remove Computer constraints
U	Add/Remove User constraints
F	Add/Remove Folder constraints
P	Add/Remove Project constraints
E	Add/Remove Program constraints
V	Add/Remove Version constraints
T	Specify Start and End times for report (in Local Time of 12d_usage User*)
S	Specify which summary tables you want in the report
R	Generate the 12d Model Usage report
Q	Exit the Report menu and return to the Main menu

After generating the report (using the “R” key), you will be asked if you want to view the report file. If you answer yes (using the “Y” key), the report file will be opened using Notepad.exe as the default file editor. If you would prefer to use a different file editor, you can specify it by manually editing the 12d_usage.defaults file (see Installing 12d_usage.exe), and replacing the word “notepad” (last line of the file), with the command to start your preferred editor.

Note:** In general, if reporting on ***12d Model sessions running on a WAN across different time zones, *Local Time of 12d Model User* and *Local Time of 12d_usage User* may be different.

Return to 39.6 Monitoring 12d Model Usage.

Or continue to the non related next section 39.7 Running Macros and Chains on Start Up.

39.7 Running Macros and Chains on Start Up

To allow for tailoring ***12d Model*** when a new project is created or an existing project opened, ***12d Model*** runs user supplied files of macros and/or chains.

For ***new*** projects, the default name of the file is

macros.4d

This can be changed to a different file by setting the environment variable

RUN_MACROS_FILE_4D *file_of_macros_to_run_for_new_projects*

For ***existing*** projects, the default name of the file is

project_macros.4d

This can be changed to a different file by setting the environment variable

RUN_PROJECT_MACROS_FILE_4D *file_of_macros_to_run_for_existing_projects*

(the environment variables can be set on the ***Extra A*** tab of the env.4d editor - see 7.6.3 env.4d)

The files consist of macros and/or chains, one per line, where

for macros, just the ***name of the macro*** is needed on the line

and for chains, the command **run_chain** *name_of_the_chain* is needed on the line.

The macros and chains are run in the order that they occur in the file.

Please continue to the next section 39.8 Arguments When Starting 12d Model.

39.8 Arguments When Starting 12d Model

12d Model can be started from an icon, or it can be started from a command line in a batch file by pointing to the ***12d Model*** executable, ***12d.exe***. For example,

"C:\Program Files\12d\12dmodel\11.00\nt.x64\12d.exe"

fires up the 64-bit ***12d.exe***.

Note This is what is used in the **Target:** field for a 12d icon. See 5.6 Project Shortcuts.

For a command line and the **Target:** field, new projects can be created and opened, existing projects can be opened, and/or a user specified ***env.4d*** file used.

39.8.1 Creating a New Project

The command

path_to_12d.exe **-create** *full_path_name_to_12d_project_to_create*

creates a ***new*** project of the given name, and ***12d Model*** starts up and opens up the new project.

If any of the path names contain spaces, then they must be enclosed in double quotes (**"**).

39.8.2 Opening an Existing Project

If the full pathname to an existing project is given after the *12d.exe* command, ***12d Model*** starts up and opens the existing project.

path_to_12d.exe *full_path_name_to_existing_12d_project*

For example

"C:\Program Files\12d\12dmodel\11.00\nt.x86\12d.exe" "C:\12d jobs\airport\airport.project"

If any of the path names contain spaces, then they must be enclosed in double quotes (**"**).

39.8.3 Environment Variables Shortcut

If a file of ***12d Model*** environment variables has been set up by the user (see section 39.4 Environment Variables in Appendix 39 Setting Up & Configuring 12d), then instead of setting the environment variable ENVIRONMENT_4D to point to the file or setting it up with the default name, env.4d, the environment file can be passed to ***12d.exe*** using the ***-env*** command.

For example

"C:\Program Files\12d\12dmodel\11.00\nt.x64\12d.exe" **-env** F:\12d\env.4d

would fire up the 64-bit ***12d.exe*** using the file of environment variables called F:\12d\env.4d

The command

"C:\Program Files\12d\12dmodel\11.00\nt.x64\12d.exe" -env F:\12d\env.4d "C:\12d jobs\airport\airport.project"

would fire up the 64-bit ***12d.exe*** using the file of environment variables called F:\12d\env.4d **and** also automatically open the project "C:\12d jobs\airport\airport.project".

If any of the path names contain spaces, then they must be enclosed in double quotes (**"**).

To return to the beginning of this appendix, click on 39 Setting Up & Configuring 12d.

40 Linestyles, Symbols, Textstyles &Patterns

This appendix contains information about how linestyles, symbols and text is defined and used in ***12d Model***.

See

40.1 Line Styles

40.2 Symbols

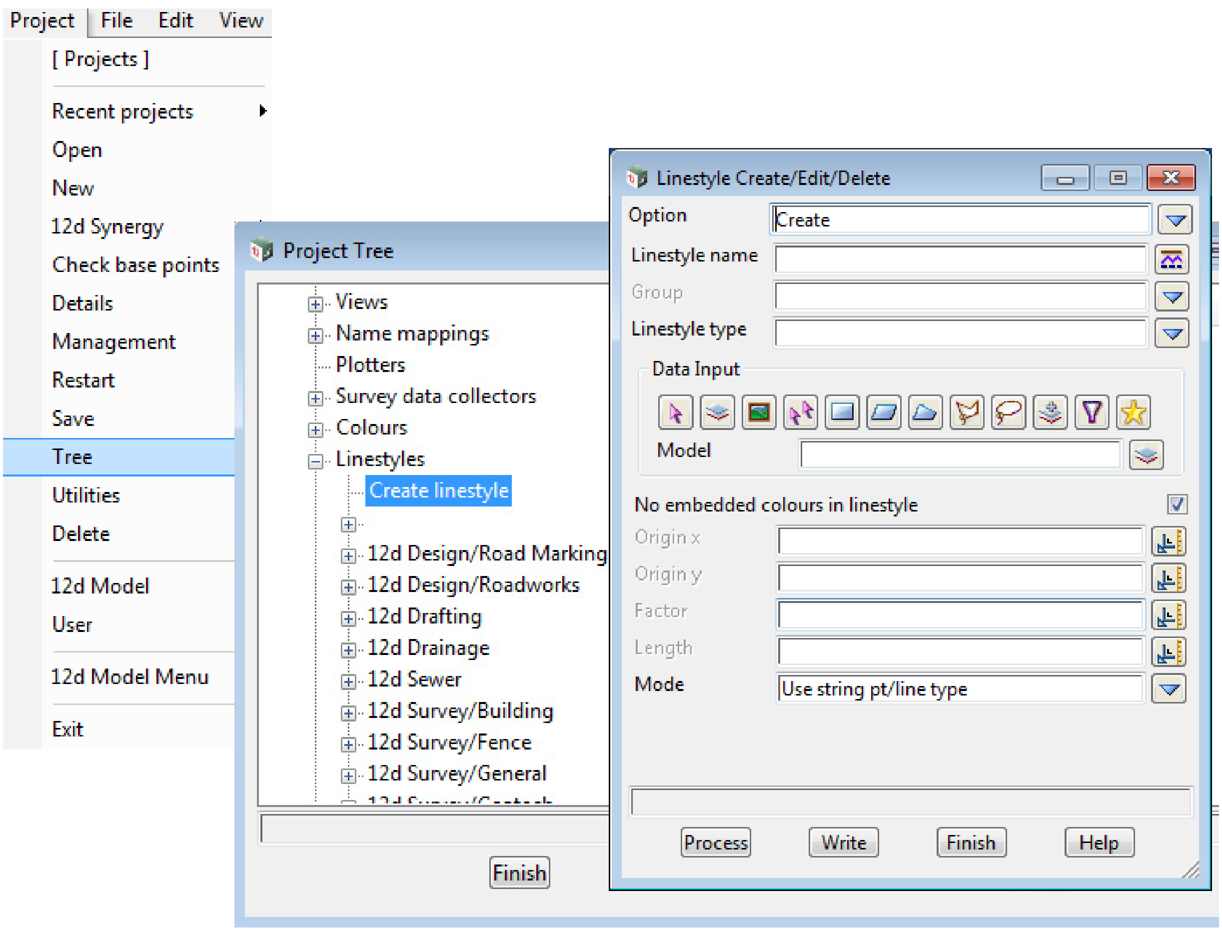
40.3 Textstyles and Fonts

40.4 Patterns

40.1 Line Styles

Users can define their own lines styles to use when drawing 12d Model strings. The definition of linestyles are stored in a file called **linestyl.4d**. (note that there is no **e** at the end of **linestyl**).

Line styles in the file can be created/edited/deleted using the **Linestyle Create/Edit/Delete** panel which is brought up by double clicking on *Create linestyle* in the *Linestyles* expansion of the *Project Tree* (see 7.9.4 Linestyles) brought up by the **Project =>Tree** option.



The **Linestyle Create/Edit/Delete** panel acts as an interactive editor to a text file which contains the linestyle definitions.

The user defined line styles can be broken into three types depending on how many origin points the line style has, plus a group style.

(a) **one origin** line styles - linestyle, paperstyle, worldstyle

These can be repeated along the string (continuous mode) or just be applied at each vertex of the string (vertex mode).

The units for the one origin line style can be in pixels, millimetres or world units and the line styles are given specific names depending on the units. The one origin line styles are called

linestyle if the definition of the one origin style is given in **pixel** units
paperstyle if the definition of the on origin style is given in **paper** units
worldstyle if the definition of the one origin style is given in **world** units

The line styles *linestyle*, *paperstyle*, *worldstyle* can be made up of lines, arcs, circles and text.

continuous mode **vertex mode**

the one origin line styles linestyle, worldstyle or paperstyle can be repeated along the string or just drawn at each point.

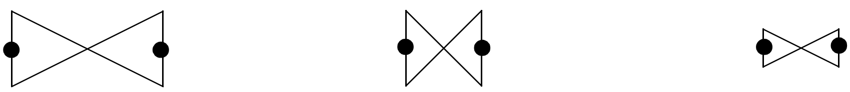
Note - to provide more flexibility, for most applications **vertex mode** has been replaced by **symbols** on **super strings**. Unlike vertex mode, symbols can be independently scaled and rotated.

(b) **two origins** line style - twoptstyle

The line style that can be is stretched between two adjacent vertices is called a **twoptstyle** style and has **two origins**.

Because the size is determine by two vertices which are only in world units, the twoptstyle is always in world units.

The line style *twoptstyle* can be made up of lines, arcs, circles and text.



A twoptstyle stretches between two points in one or both directions.

(c) **three origins** line style

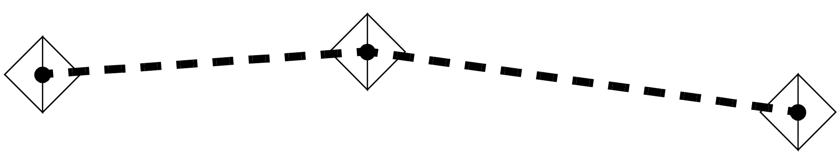
The line style that can be is stretched in two directions using three adjacent vertices is called a **threeptstyle** style and has **three origins**.

Because the size is determine by three vertices which are only in world units, the threeptstyle is always in world units.

The line style *threeptstyle* can only be made up of lines.

(d) **group** line style

The final type of line style is called a **groupstyle** and is a combined style made up of one or more *linestyles*, *paperstyles*, *worldstyles* or *twoptstyles*.



A groupstyle made up of a vertex and a continuous line style.

Warning

The words **line style** have been used to cover all the different types *linestyle*, *paperstyle*, *worldstyle*, *twoptstyle*, *threeptstyle* or *groupstyle* but for historical reasons, the word linestyle has also been used for all the different types although it also refers to a particular line style.

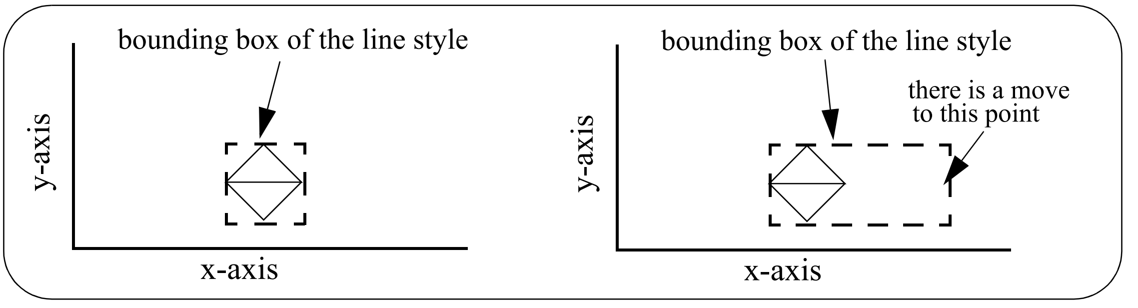
For further information about line styles, see

[40.1.1 Defining Line Styles]
[40.1.2 Linestyl.4d]
[40.1.3 Line Style Definitions]
Or return to [40.1 Line Styles]

40.1.1 Defining Line Styles

A *line style* is defined in a (x,y) co-ordinate system as a series of moves, drawn, arcs, circles and text commands.

The bounding box of the line style is the smallest rectangle parallel to the (x,y) axis which contains all the moves, draws, circles and arcs of the line style. Note that moves are included in the bounding box even if a line isn't drawn to the point.



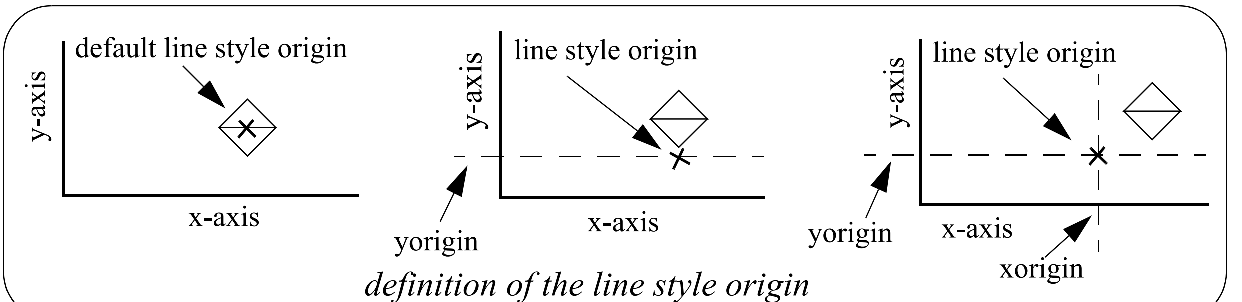
The default **origin of the line style** is defined to be the midpoint of the bounding box surrounding the line style. The bounding box includes the points moved to and drawn to, but does not automatically include (0,0). Hence the calculated origin may not be (0,0).

There are **xorigin** and **yorigin** commands to override the default origin of the line style.

If a **yorigin** is set, then it is used as the y-coordinate of the line style. If the *xorigin* isn't given, then the *xorigin* is the midpoint of the x-extent of the bounding box of the line style.

Similarly, if a **xorigin** is set, then it is used as the x-coordinate of the line style.

If the *yorigin* isn't given, then the *yorigin* is the midpoint of the y-extent of the bounding box of the line style.

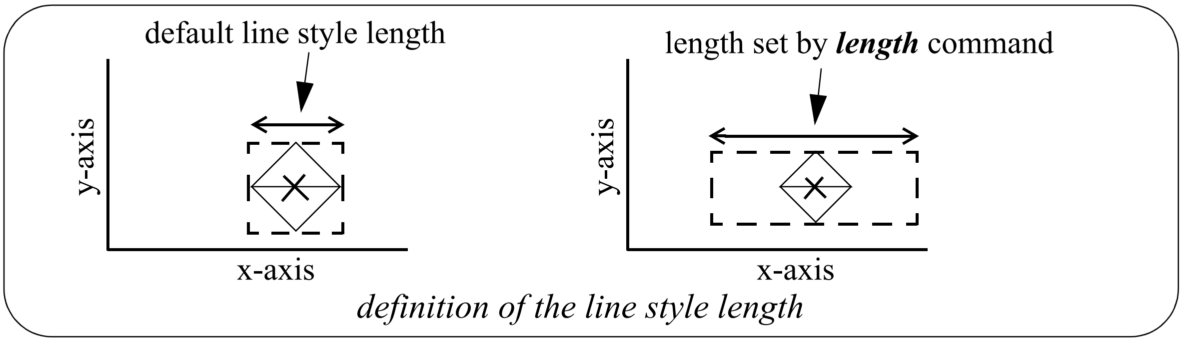


definition of the line style origin

The default **length** of the line style is the *horizontal size* of the bounding box.

There is a **length** commands to override the default length of the line style.

The *length* must always be positive but it can be larger or smaller than the line styles's calculated horizontal length.

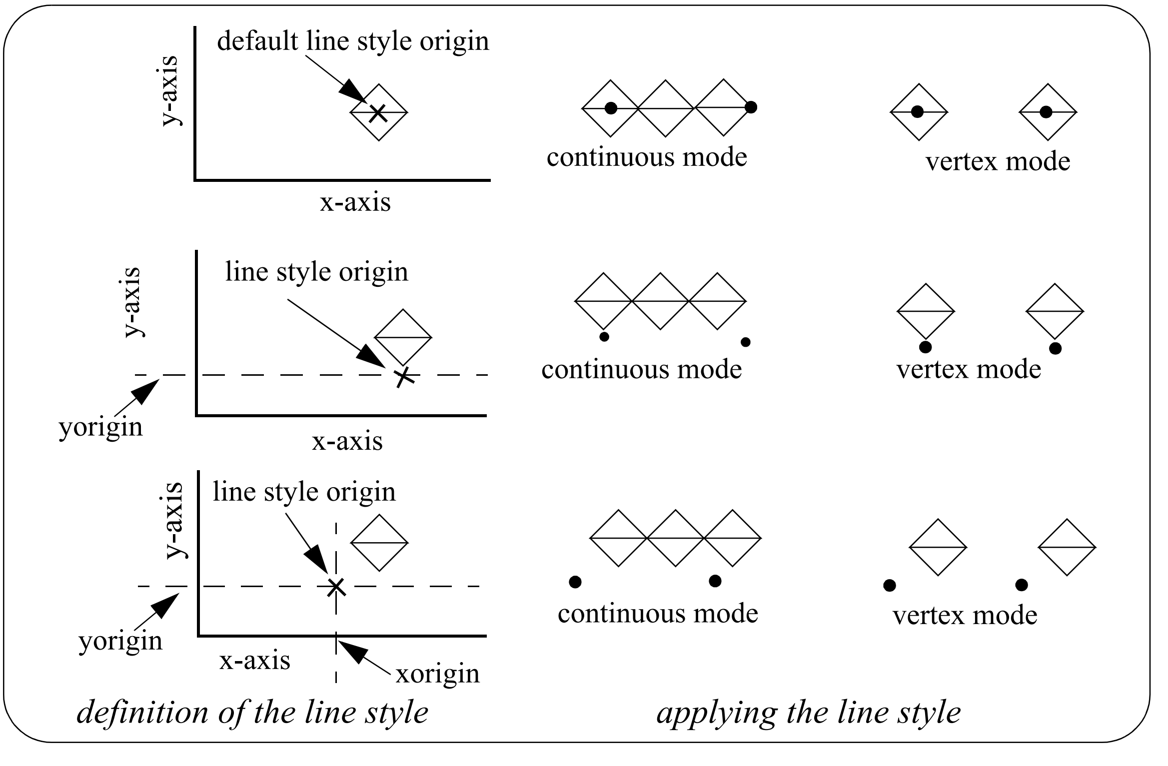


The ***origin*** and the ***length*** of the line style are both used in **positioning** and **redrawing** the line style.

For *continuous mode*, the ***origin*** of the line style is initially placed at a distance of half of the ***length*** of the line style along the first link in the string, and then moved the distance ***length*** along the string and redrawn.This is repeated along the string until the end of the string is reached.

Hence for *continuous mode*, the style is continually repeated along the string.

For *vertex mode*, the ***origin*** of the line style is placed at each point of the string and the line style drawn around the ***origin***.



Continue to 40.1.2 Linestyl.4d or return to 40.1 Line Styles.

40.1.2 Linestyl.4d

The definitions for the available linestyles are normally stored in a file called **linestyl.4d** which is read in each time ***12d Model*** opens an existing project or creates a new project. *Linestyl.4d* is searched for in the standard set up paths, or is pointed to by the environment variable

LINESTYLES_4D *filename*

The definition of the line styles will be given in the following section 40.1.3 Line Style Definitions.

Continue to 40.1.3 Line Style Definitions or return to 40.1 Line Styles.

40.1.3 Line Style Definitions

See

40.1.3.1 Linestyles, Paperstyles and Worldstyles

40.1.3.2 Twoptstyles

40.1.3.3 Threeptstyles

40.1.3.4 Groupstyles

40.1.3.5 Description of the draw_commands for Linestyles

40.1.3.1 Linestyles, Paperstyles and Worldstyles

The line style with **one origin** is called a

linestyle if the definition of the style is given in *pixel* units

paperstyle if the definition of the style is given in *paper* units

worldstyle if the definition of the style is given in *world* units.

Aside:

A line style with two origins is called a **twoptstyle,** and a line style with three origins is called a **threeptstyle**. These behave differently to linestyles, paperstyles and worldstyles and will be documented separately.

The one origin style can be drawn at each point of a string (**vertex mode**) or redrawn regularly along the lines joining string vertices (**continuous mode**).

The definition of a **linestyle** of a given name *name* is

linestyle name {

set_up_commands

draw_commands

}

a **paperstyle** is

paperstyle name {

set_up_commands

draw_commands

}

and a **worldstyle** is

worldstyle name {

set_up_commands

draw_commands

}

where the set_up_commands **must** be before the draw_commands.

The linestyle/paper/worldstyle must have a name and if the name includes imbedded spaces, it must be enclosed in double quotes **"**.

The line style itself is defined inside the braces where there can be zero or more set_up_commands from the list

`length` *value*
`group` *name*
`mode` *value*

```
factor    value
xorigin   value
yorigin   value
```

followed by zero or more **draw_commands** from the list

```
move      x-value      y-value
draw      x-value      y-value
rmove     dx-value     dy-value
rdraw     dx-value     dy-value
colour    colour_name
circle    radius
arc       radius       start-angle   end-angle
text      "text"       angle     height     "justification"
text      "text"       angle     height     "justification"   "textstyle"
text  "text"  angle  height  "justification"  "textstyle"  xfactor  slant  offset _width  offset_height
repeat  num_repeats{    repeat_commands  repeat_draw_commands}
```

The **repeat** command is a positive integer *num_repeats*

and zero or more repeat_commands from the list

```
xpos      x-value
ypos      y-value
rfactor   value
```

and zero or more repeat_draw_commands where repeat_draw_commands include all the draw_commands except **repeat**.

The **set_up** commands for linestyles, paperstyles and worldstyles will now be described. The **draw_commands** will be described after the definition of a groupstyle since they are the same for linestyles, paperstyles, worldstyles and twoptstyles.

set_up_commands for Linestyles, Paperstyles and Worldstyles

group

The **group** is used to associate line styles in pop-up menus.

If the group is not defined, the style is placed in a default (blank) group.

If the group name includes imbedded spaces, then it must be enclosed in double quotes **"**.

mode

mode specifies whether the linestyle, paperstyle or worldstyle is drawn at the individual vertices on a string (vertex mode) or it is to be redrawn regularly along the lines joining the vertices on the string (continuous mode).

If the **mode** is not specified, the mode used for a string depends on the breakline type of the string and the number of vertices on the string.

Breakline Type	Number of Vertices	mode: vertex	mode: continuous	mode: not-given
point	1	vertex	cross	vertex
	>1	vertex	continuous	vertex
line	1	vertex	cross	vertex
	>1	vertex	continuous	continuous

NOTE: in early versions of ***12d Model***, **mode** was known as **pointline** and had the values **point** instead of *vertex*, and **line** instead of *continuous*.

factor

This command factors up/down the drawing co-ordinates, heights and radii.

xorigin, yorigin

The default **origin of the line style** is defined to be the midpoint of the bounding box of the line style. The bounding box includes the points moved to and drawn to but does not automatically include (0,0). Hence the calculated origin may not be (0,0).

The **xorigin** and **yorigin** commands are used to override the default origin of the line style.

length

If length is omitted, then **length** is taken to be the horizontal size of the calculated bounding box of the style.

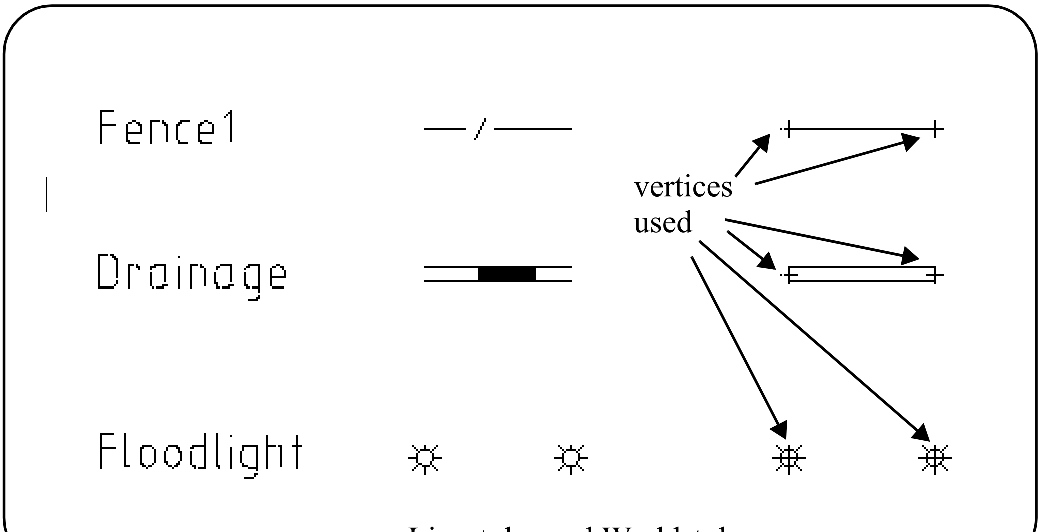
The **length** must always be positive but it can be larger or smaller than the line style's calculated horizontal size.

The origin of the line style and the length are both used in positioning and redrawing the line style.

In **vertex** mode, the origin of the line style is placed at the string points.

In **continuous** mode, the origin is initially placed at a distance of half of **length** along the first line in the string, and then moved the distance **length** along the string for each redraw.

Examples



Linestyles and Worldstyles

```
// Fence1
worldstyle "FENCE1"{
  group SWCS
  factor 20
  move 0 0
  rdraw 1 0
  rmove 0.25 -0.25
  rdraw 0.25 0.50
  rmove 0.25 -0.25
  rdraw 1 0
}
```

```
// Drainage

worldstyle drainage {
  group "4d"
  factor 0.1
  colour green

  move 0.0 -2.0  draw 30 -2.0

  move 30 -1.5  draw 15 -1.5
  move 15 -1.0  draw 30 -1.0
  move 30 -0.5  draw 15 -0.5

  move 15 0.0  draw 30 0.0

  move 30  0.5  draw 15  0.5
  move 15  1.0  draw 30  1.0
  move 30  1.5  draw 15  1.5

  move 30  2.0  draw  0  2.0
}
// Floodlight
worldstyle FLOODLIGHT {
 group "Energex"
 mode vertex
 xorigin 0
 yorigin 0
 colour "yellow"
 circle 0.225
 move -.45 0
 draw -0.225 0
 move 0.45 0
 draw 0.225 0
 move 0 -.45
 draw 0 -0.225
 move 0 0.45
 draw 0 0.225
 move -.159 0.159
 draw -0.318 0.318
 move 0.159 0.159
 draw 0.318 0.318
 move 0.159 -0.159
 draw 0.318 -0.318
 move -0.159 -0.159
 draw -0.318 -0.318
 move 0 0.8
 text "FL" 0 0.5 "centre-middle"
}
```

Continue to 40.1.3.2 Twoptstyles or return to 40.1.3 Line Style Definitions or 40.1 Line Styles.

40.1.3.2 Twoptstyles

A **twoptstyle** is a line style which is stretched so that one cycle fits between consecutive vertices on a string (line or arc segments). This is achieved by defining **two origins** for the twoptstyle and the first and second origins are mapped to the consecutive vertices on the string.

Hence the mapping of the two origins defines the final size of the size in world units.

The *twoptstyle* can stretch in just the direction along the string, or both along and perpendicular to the direction of the string.

Also the *twoptstyle* can be drawn for each line/arc on the string, or for every second line/arc of the string.

For a feature/circle string, the first origin is placed at the centre of the feature/circle and the second origin is placed on the circumference of the feature/circle. Hence the size of the *twoptstyle* is determined by the radius of the feature/circle string.

The definition of a **twoptstyle** of a given name is

```
twoptstyle   name {
    set_up_commands
    draw_commands
}
```

where the set_up_commands **must** be before the draw_commands.

The *twoptstyle* must have a name and if the name includes imbedded spaces, it must be enclosed in double quotes **“**.

The *twoptstyle* itself is defined inside the braces where there can be zero or more set_up_commands from the list

```
group          name
stretch_mode   value
cycle_mode     value
xorigin1       value
yorigin1       value
xorigin2       value
yorigin2       value
```

followed by zero or more draw_commands from the list

```
move     x-value       y-value
draw     x-value       y-value
rmove    dx-value      dy-value
rdraw    dx-value      dy-value
colour   colour_name
circle   radius
arc      radius        start-angle   end-angle
text     "text"        angle     height     "justification"
text     "text"        angle     height     "justification"   "textstyle"
text  "text"  angle  height  "justification"  "textstyle"  xfactor  slant  offset  width  offset_height
repeat  num_repeats{    repeat_commands  repeat_draw_commands}
```

The **repeat** command is a positive integer *num_repeats* and zero or more *repeat_commands* from the list

```
xpos      x-value
ypos      y-value
rfactor   value
```

and zero or more *repeat_draw_commands* where *repeat_draw_commands* include all the *draw_commands* except **repeat**.

The *set_up* commands for a twoptstyle will now be described. The *draw_commands* will be described after the definition of a ***groupstyle*** since they are the same for linestyles, paperstyles, worldstyles and twoptstyles.

set_up_commands for Twoptstyles

group

The **group** is used to associate line styles in pop-up menus.

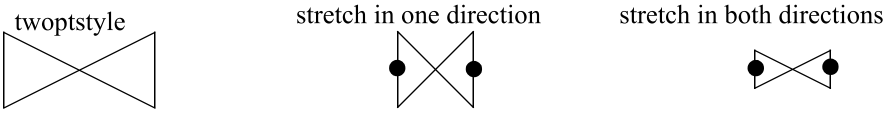
If the group is not defined, the line style is placed in a default (blank) group.

If the group name includes imbedded spaces, then it must be enclosed in double quotes **"**.

stretch_mode

stretch_mode controls whether the symbol stretches in just the direction along the string, or both along and perpendicular to the direction of the string

stretch_mode	1	// stretch in one direction only (default)
	2	// stretch in both directions



A twoptstyle stretches between two vertices in one or both directions.

cycle_mode

cycle_mode controls whether the symbol is drawn for each line/arc on the string, or for every second line/arc of the string.

cycle_mode	1	// draws on every line/arc (default)
	2	// draws every second line/arc

A twoptstyle draws on each line/arc or every second line/arc

xorigin1, yorigin1

The point (xorigin1, yorigin1) is mapped to the first vertex that the *twoptstyle* is being applied to.

The default **first point of origin of the twoptstyle** is defined to be the minimum x value and the average of the minimum and maximum y values (x_min,(y_min+y_max)/2).

The **xorigin1** and **yorigin1** commands are used to override the default first origin of the *twoptstyle*.

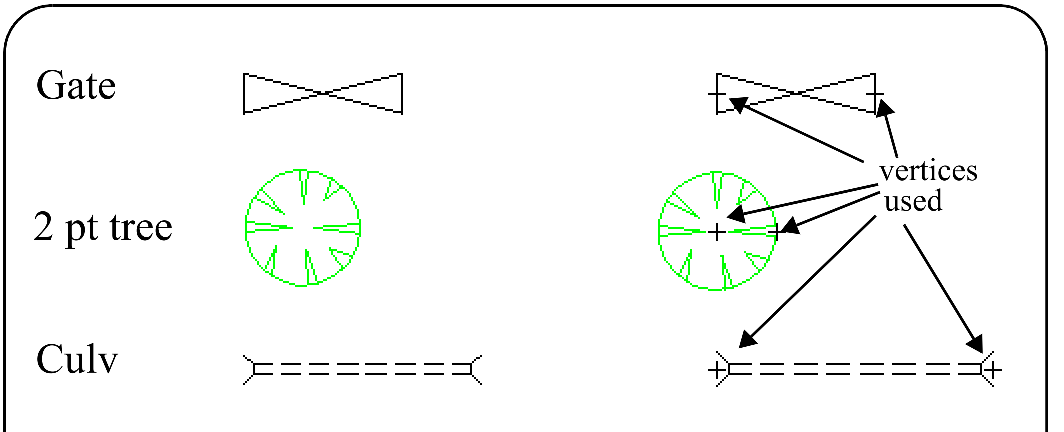
xorigin2, yorigin2

The point (xorigin2, yorigin2) is mapped to the second vertex that the *twoptstyle* is being applied to.

The default **second point of origin of the twoptstyle** is defined to be the maximum x value and the average of the minimum and maximum y values (x_max,(y_min+y_max)/2).

The **xorigin2** and **yorigin2** commands are used to override the default second origin of the *twoptstyle*.

Examples



Two Point Styles

```
// Gate
twoptstyle GATE {
  group "Two Points"
  stretch_mode 2
  cycle_mode   2
  colour red
  move 100   105
  draw 140    95
  draw 140   105
  draw 100    95
  draw 100   105
}
// two point tree
twoptstyle 2PT_TREE {
 group     "4d_standards"
 stretch_mode 2
 cycle_mode   2
 xorigin10
 yorigin10
// xorigin22.5
// yorigin2 1.25

 colour "green"
 move 0 0
 circle 2.5
 move 2.48 0.2
 draw 0.5 0
 draw 2.48 -0.2
 move 2.2 -1.2
 draw 1.2 -0.9
 draw 1.8 -1.7
 move 0.7 -2.4
```

```
 draw 0.2 -1
 draw 0.3 -2.45
 move -0.2 2.45
 draw 0 1
 draw 0.2 2.45
 move 1.2 2.2
 draw 0.9 1.2
 draw 1.7 1.8
 move -2.48 0.2
 draw -0.5 0
 draw -2.48 -0.2
 move -1.8 1.6
 draw -0.8 0.6
 draw -2.1 1.3
 move -1.6 -1.9
 draw -1.1 -0.8
 draw -1.3 -2.1
}
// Culvert
twoptstyle CULV {
  group "Two Points"
  stretch_mode 2
  cycle_mode   2

  move   8    12
  draw  10    10
  draw  14    10
  move  16    10
  draw  20    10
  move  22    10
  draw 26    10
  move 28    10
  draw 32    10
  move 34    10
  draw 38    10
  move 40    10
  draw 44    10
  move 46    10
  draw 50    10
  move 52    10
  draw 56    10
  draw 58    12
  move 58     6
  draw 56     8
  draw 56    10
  move 10    10
  draw 10     8
  draw 8     6
  move 10     8
  draw 14     8
  move 16     8
  draw 20     8
  move 22     8
  draw 26     8
  move 28     8
  draw 32     8
  move 34     8
  draw 38     8
```

```
move 40     8
draw 44     8
move 46     8
draw 50     8
move 52     8
draw 56     8
}
```

Continue to 40.1.3.3 Threeptstyles or return to 40.1.3 Line Style Definitions or 40.1 Line Styles.

40.1.3.3 Threeptstyles

A **threeptstyle** is a line style that one cycle fits between groups of three consecutive vertices on a string (line or arc segments).This is achieved by defining **three origins** for the threeptstyle.

The threeptstyle is fitted to three consecutive vertices by stretching the threeptstyle differently in two perpendicular directions.

The **first** and **second origins** are **mapped** to the **first two consecutive vertices** of the groups of three vertices on the string and that defines a stretching in the direction from vertex one to vertex two.

The second stretching is perpendicular to the line joining the first two origins and is in the direction of the third origin. When fitting the threeptstyle to the three points, the threeptstyle is stretched in the direction **perpendicular** by the factor equal to:

(perpendicular distance of the third point from the line joining the first two vertices) **/**

(per pedicular distance of the third origin from the line joining the first two origins).

So the **third origin** is usually **not mapped** to the **third vertex**. But the third origin will be on the line through the third vertex and parallel to the line through the first two vertices.

Hence the mapping of the three origins defines the final size of the size in world units.

Important Note - because of the stretching in tow independent directions, the threeptstyle can only be made up of straight lines.

The definition of a **threeptstyle** of a given name is

```
threeptstyle   name {
   set_up_commands
   draw_commands
}
```

where the set_up_commands **must** be before the draw_commands.

The *threeptstyle* must have a name and if the name includes imbedded spaces, it must be enclosed in double quotes **"**.

The *threeptstyle* itself is defined inside the braces where there can be zero or more set_up_commands from the list

```
group          name
xorigin1       value
yorigin1       value
xorigin2       value
yorigin2       value
xorigin3       value
yorigin3       value
```

followed by zero or more draw_commands from the list

```
move     x-value    y-value
draw     x-value    y-value
rmove    dx-value   dy-value
rdraw    dx-value   dy-value
colour   colour_name
repeat num_repeats{   repeat_commands  repeat_draw_commands}
```

The **repeat** command is a positive integer *num_repeats* and zero or more *repeat_commands* from the list

```
xpos     x-value
```

```
ypos        y-value
rfactor     value
```

and zero or more *repeat_draw_commands* where *repeat_draw_commands* include all the *draw_commands* except **repeat**.

The *set_up* commands for a threeptstyle will now be described. The *draw_commands* will be described after the definition of a ***groupstyle*** since they are the same for linestyles, paperstyles, worldstyles and twoptstyles.

set_up_commands for Threeptstyles

group

The **group** is used to associate line styles in pop-up menus.

If the group is not defined, the line style is placed in a default (blank) group.

If the group name includes imbedded spaces, then it must be enclosed in double quotes **"**.

xorigin1, yorigin1

The point (xorigin1, yorigin1) is mapped to the first vertex that the *threeptstyle* is being applied to.

The default **first point of origin of the threeptstyle** is defined to be the minimum x value and the average of the minimum and maximum y values (x_min,(y_min+y_max)/2).

The **xorigin1** and **yorigin1** commands are used to override the default first origin of the *threeptstyle*.

xorigin2, yorigin2

The point (xorigin2, yorigin2) is mapped to the second vertex that the *threeptstyle* is being applied to.

The default **second point of origin of the threeptstyle** is defined to be the maximum x value and the average of the minimum and maximum y values (x_max,(y_min+y_max)/2).

The **xorigin2** and **yorigin2** commands are used to override the default second origin of the *threeptstyle*.

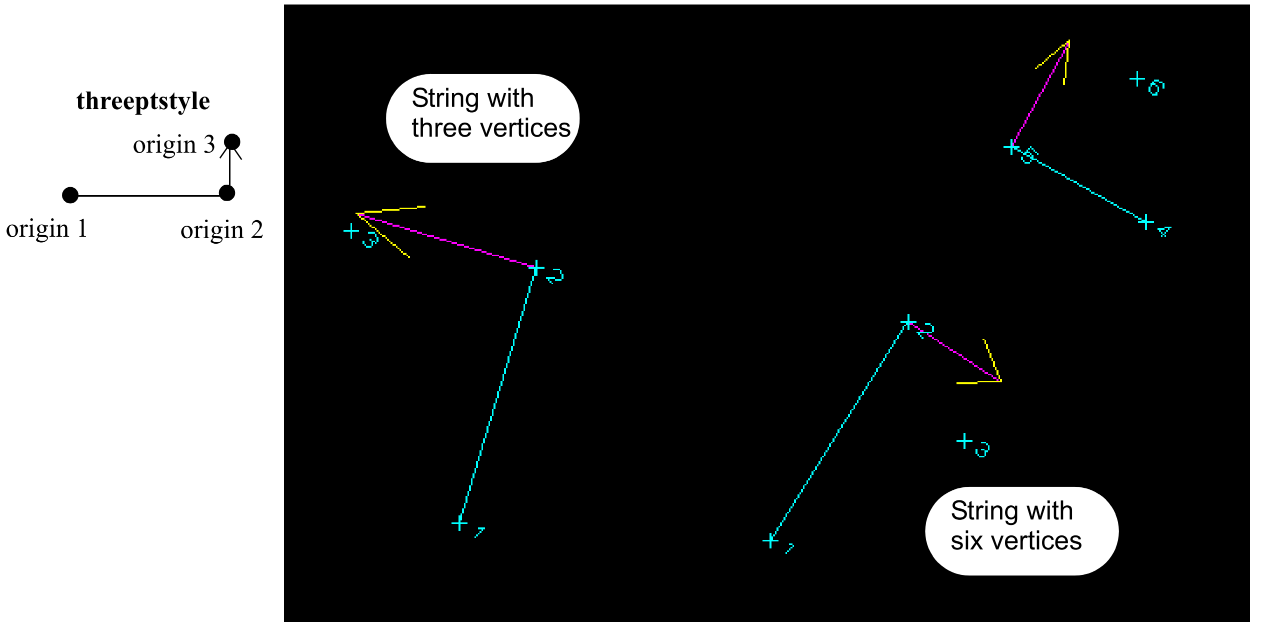
xorigin3, yorigin3

The point (xorigin3, yorigin3) is mapped to the a point on the line parallel to the line through the first two vertices and going through the third vertex of the three vertices that the threeptstyle is being applied to.

The default **third point of origin of the threeptstyle** is defined to be the maximum x value and the maximum y value (x_max,y_max).

The **xorigin3** and **yorigin3** commands are used to override the default third origin of the *threeptstyle*.

Examples



A threeptstyle stretches between two vertices in one direction and has a different stretch in the perpendicular direction.

```
threeptstyle "simple turn arrow" {
  group "3 pts"
  xorigin1  0
  yorigin1  0
  xorigin2  10
  yorigin2  0
  xorigin3  10
  yorigin3  3

  colour   cyan
  move      0.0   0.0
  draw      10.0  0.0

  colour magenta
  move     10.0 0.0
  draw     10.0 3.0

  colour   yellow
  move      10.0   3.0
  draw       9.0   2.0
  move      10.0   3.0
  draw      11.0   2.0
}
```

Continue to 40.1.3.4 Groupstyles or return to 40.1.3 Line Style Definitions or 40.1 Line Styles.

40.1.3.4 Groupstyles

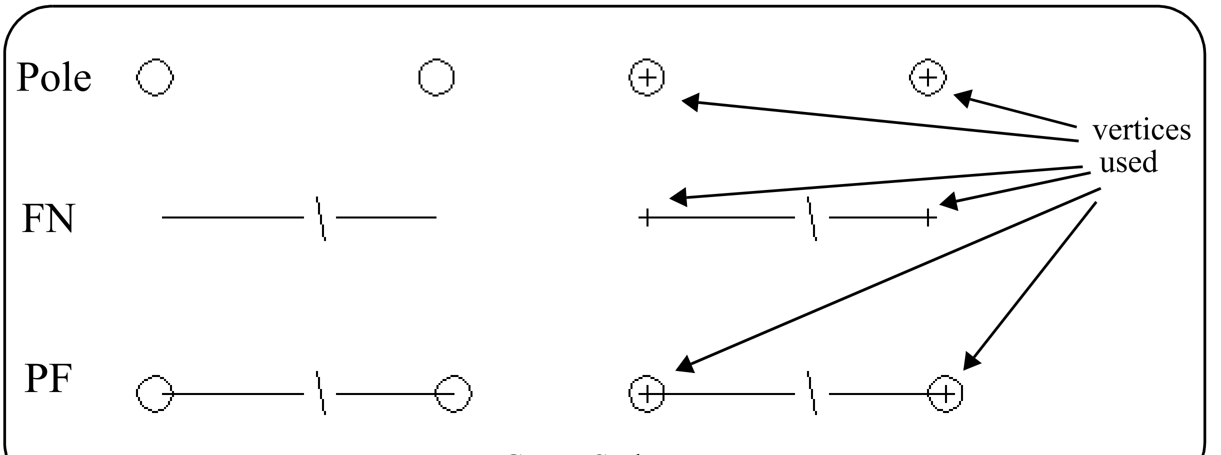
A **groupstyle** is a combined line style which is made up of one or more *linestyles*, *paperstyles*, *worldstyles* and/or *twoptstyles*.

The definition of a **groupstyle** is simply an optional **group** for the line style to belong to, and a list of linestyle, paperstyles, worldstyle and twoptstyle names which must have already been defined in the file.

```
groupstyle   name {

    group      name

    style_name_1
    style_name_2
      ...

}
```

If the *groupstyle* name contains spaces, then it must be enclosed in double quotes **"**. For example, "style 1".

Example



Group Style

```
// POLE
worldstyle POLE {
  group "Electricity"
  mode vertex
  circle 0.5
}
// FN
worldstyle FN {
  group "Fences"
  length 5
  draw 4 0
  move 4.4 0.6
  draw 4.6 -0.6
}
// PF
}groupstyle "PF" {
  group "Miscellaneous"
  "POLE"
  "FN"
}
```

40.1.3.5 Description of the draw_commands for Linestyles

move *x-value* *y-value*

move the pen from the current pen position to the new position (x-value, y-value).

draw *x-value* *y-value*

draw from the current pen position to the new position (x- value, y-value)

rmove *dx-value* *dy-value*

move relative from the current pen position through the distance (dx-value, dy-value)

rdraw *dx-value* *dy-value*

draw relative from the current pen position through the distance (dx-value, dy-value)

colour *colour-name*

change colour (line styles can contains more than one colour).

If colour is not defined, the line style is drawn in the colour of the string that the line style is applied to. If the colour-name includes imbedded spaces, then it must be enclosed within double quotes **"**.

circle *radius*

draw a circle of the given radius at the current pen position.

After drawing a circle, the current pen position is left at the centre of the circle.

arc *radius* *start-angle* *end-angle*

draw an arc of the given radius from the start angle to the end angle. A positive radius denotes that the arc is drawn in a clockwise direction, a negative radius means anti-clockwise. Angles are given in degrees, minutes and seconds in the ddd.mmssfff format.

After drawing an arc, the current pen position is left at the centre of the arc.

text *text* *angle* *height* **"***justification***"**

text *text* *angle* *height* **"***justification***"** textstyle

or

text *text* *angle* *height* **"***justification***"** textstyle xfactor slant offset_width offset_height

the characters *text* are drawn at the current pen position with the given angle, height, justification and possibly textstyle, xfactor, slant, offset_width and offset_height. Angle and slant are given in degrees, minutes and seconds in the ddd.mmssfff format. Slant is between -45 and 45 degrees.

The allowed justifications are

top-left	top-centre	top-right
middle-left	middle-centre	middle-right
bottom-left	bottom-centre	bottom-right

and the value is enclosed inside double quotes (").

After drawing text, the current pen position is left where it was before the text was drawn.

repeat num_repeats { *repeat_commands* *repeat_draw_commands* }

The **repeat** command allows the set of repeat_draw_commands inside the braces (the repeat-style) to be re-drawn a number of times (num_repeats), and with each redraw, the size of the repeat-style is modified.

In the definition of repeat, there can be zero or more *repeat_commands* from the list

xpos *x-value*

ypos *y-value*

rfactor *value*

and zero or repeat_draw_commands where repeat_draw_commands include all draw_commands except **repeat**.

Unlike style, the repeat-style is defined in terms of a local origin at (0,0). The repeat-style is actually drawn with this origin at the position given by the values of **xpos** and **ypos**.

The repeat-style is drawn *num_repeats* times and on each redraw, the distances, heights and radii are factored by the amount

repeat_factor = (1 - i*rfactor)

where i = 0,1,...num_repeats-1

If rfactor is not specified, it is given the default value of 1/num_repeats. The factor is then

repeat_factor = (num_repeats - i)/num_repeats

where i = 0,1,...num_repeats-1

Examples of the Repeat Command

1. To produce circles of radius 10,9,8,...1 about the origin.

```
repeat 10 { circle 10}
```

2. To produce circles of radius 10,9.5,9,...5.5 about the origin

```
repeat 10 { rfactor 0.05 circle 10}
```

More Examples of Line, Paper and World Styles

1. draw a circle of radius 20. The origin of the style is (0,0)

```
worldstyle "circle" {
  move 0 0
  circle 20
}
```

2. draw a circle of radius 20. The origin of the style is (100,100)

```
paperstyle "circle" {
  move 100 100
  circle 20
}
```

3. draw a circle of radius 20. The origin of the style is (0,0)

```
linestyle "circle" {
  xorigin 0
  yorigin 0
  move 100 100
  circle 20
}
```

4. draw a gate symbol as a twoptstyle.

```
worldstyle "circle" {
  group "fences"
  stretch_mode  1            // one direction stretch
  cycle_mode  2              // draw every second line

  move   0  0
```

```
    draw   0  0.1
    draw   1  0
    draw   1  0.1
    draw   0  0
  }
```

5. draw concentric circles of centre (0,0) inside a box

```
linestyle "dot"  {
  xorigin 0
  yorigin 0
  move -5 -5
  draw -5 25
  draw 25 25
  draw 25 -5
  draw -5 -5
  repeat 10 {
    move 0 0
    circle 10
  }
}
```

6. draw an arc of absolute radius 10 in the clockwise direction from the angle 45 to the angle -45.

```
linestyle "arc" {
  move 0 0
  arc 10 45 -45
}
```

7. draw an arc of absolute radius 10 in the anti-clockwise direction from the angle 45 to the angle -45.

```
worldstyle "arc 1" {
   move 0 0
  arc -10 45 -45
}
```

8. top-left justified text with a circle centred on the actual text position

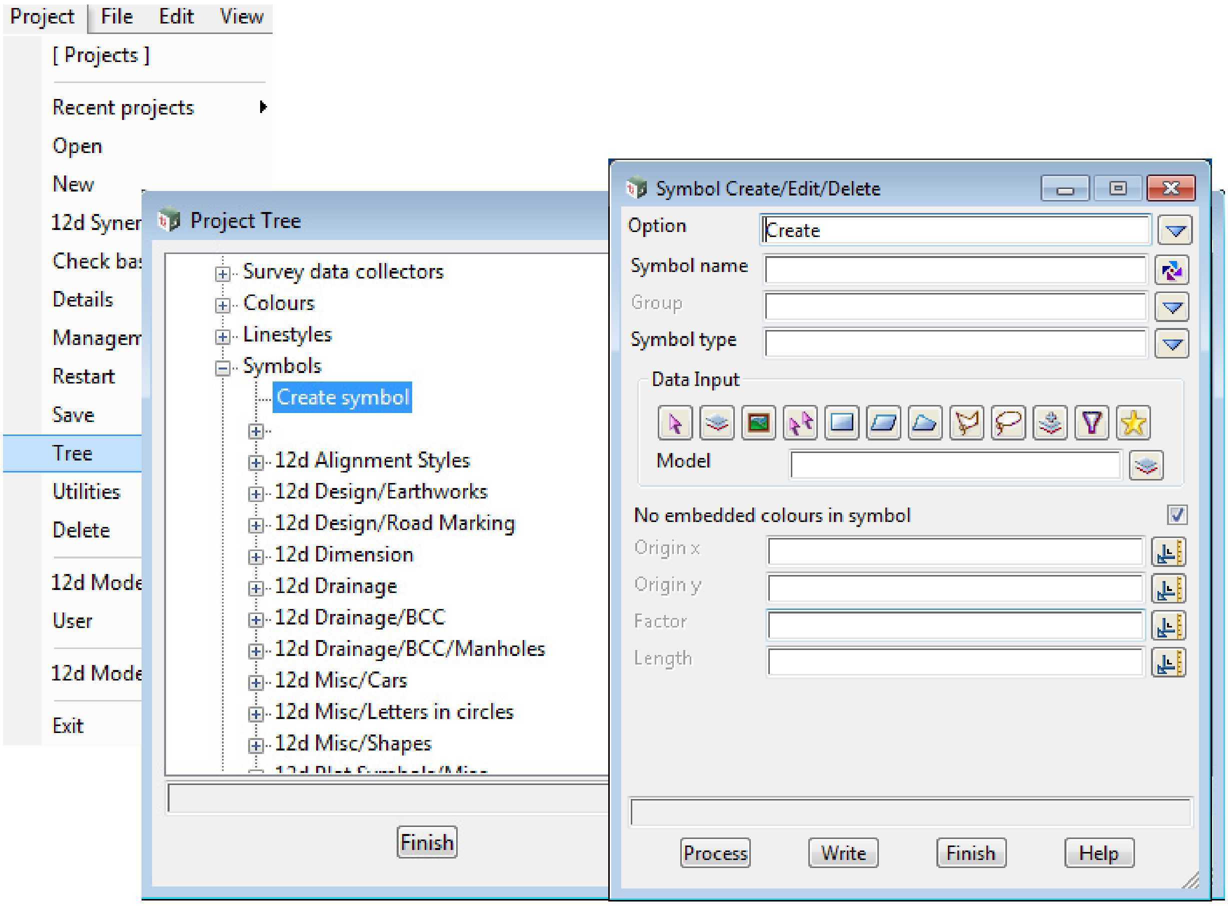
```
linestyle "top-left"  {
  group text
  move 0 0
  circle 4
  text   "<top left>" 0 10 "top-left"
}
```

Return to 40.1.3 Line Style Definitions or 40.1 Line Styles.

40.2 Symbols

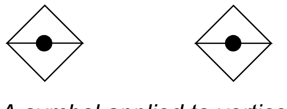
Users can define their own symbols to draw at vertices of ***12d Model*** strings. The definition of symbols are stored in a file called **symbols.4d**.

Symbols in the file can be created/edited using the **Symbols Create/Edit/Delete** panel which is brought up by double clicking on *Create symbol* in the *Symbols* expansion of the *Project Tree* (see 7.9.5 Symbols) brought up by the **Project** =>**Tree** option.



The **Symbol Create/Edit/Delete** panel acts as an interactive editor to a text file which contains the symbol definitions.

The user defined symbols can be applied to each vertex of a string. Each symbol can be made up of lines, arcs, circles and text.



A symbol applied to vertices.

The units for the symbol can be in pixels, millimetres or world units. The symbol is called a

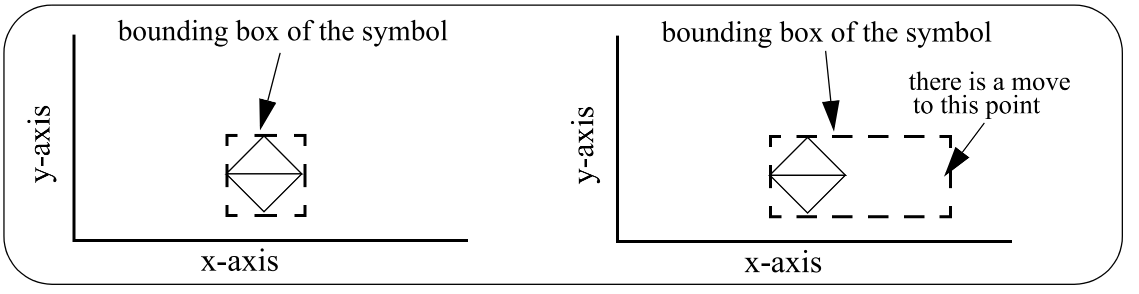
paperstyle if the definition of the symbol is given in paper units
worldstyle if the definition of the symbol is given in world units

For further information about **Symbols**, see

Or return to 40.2 Symbols

40.2.1 Defining Symbols

A *symbol* is defined in a (x,y) co-ordinate system as a series of moves, drawn, arcs, circles and text commands.
The bounding box of the line style is the smallest rectangle parallel to the (x,y) axis which contains all the moves, draws, circles and arcs of the line style. Note that moves are included in the bounding box even if a line isn't drawn to the point.

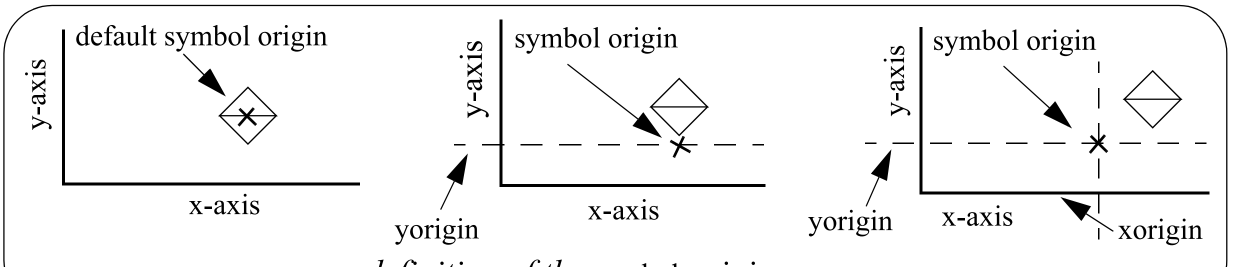


The default **origin of the symbol** is defined to be the midpoint of the bounding box surrounding the symbol. The bounding box includes the points moved to and drawn to, but does not automatically include (0,0). Hence the calculated origin may not be (0,0).

There are **xorigin** and **yorigin** commands to override the default origin of the symbol.

If a **yorigin** is set, then it is used as the y-coordinate of the symbol. If the *xorigin* isn't given, then the *xorigin* is the midpoint of the x-extent of the bounding box of the symbol.

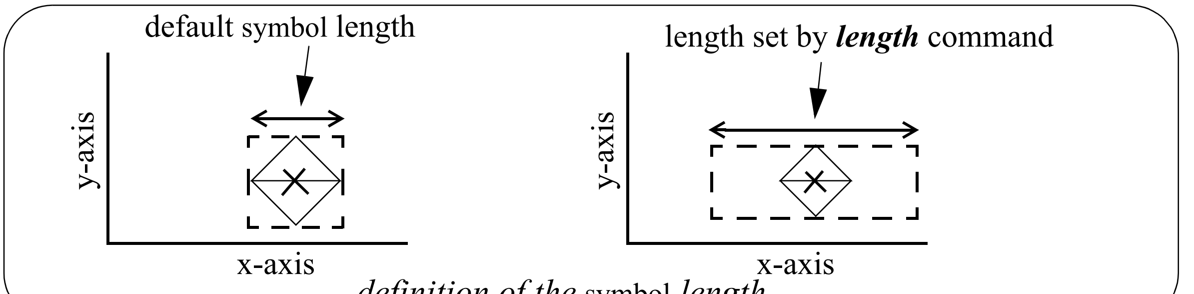
Similarly, if a **xorigin** is set, then it is used as the x-coordinate of the symbol. If the *yorigin* isn't given, then the *yorigin* is the midpoint of the y-extent of the bounding box of the symbol.



definition of the symbol *origin*

The default **length** of the line style is the *horizontal size* of the bounding box.

There is a **length** commands to override the default length of the symbol. The *length* must always be positive but it can be larger or smaller than the symbols calculated horizontal length.



definition of the symbol *length*

The *origin* and the *length* of the symbol are both used in positioning and sizing the symbol.

Continue to 40.2.2 Symbol.4d or return to 40.2 Symbols.

40.2.2 Symbol.4d

The definitions for the available symbols are normally stored in a file called **symbols.4d** which is read in each time ***12d Model*** opens an existing project or creates a new project. *Symbol.4d* is searched for in the standard set up paths, or is pointed to by the environment variable

SYMBOLS_4D *filename*

The definition of the symbols will be given in the following section 40.2.3 Symbol Definition.

Continue to 40.2.3 Symbol Definition or return to 40.2 Symbols.

40.2.3 Symbol Definition

The definition of a worldstyle **symbol** of a given name *name* is

worldstyle name {

set_up_commands

draw_commands

}

where the set_up_commands **must** be before the draw_commands.

The linestyle/paper/worldstyle must have a name and if the name includes imbedded spaces, it must be enclosed in double quotes **"**.

The symbol itself is defined inside the braces where there can be zero or more set_up_commands from the list

```
length    value
group     name
factor    value
xorigin   value
yorigin   value
```

followed by zero or more **draw_commands** from the list

```
move      x-value     y-value
draw      x-value     y-value
rmove     dx-value    dy-value
rdraw     dx-value    dy-value
colour    colour_name
circle    radius
arc       radius      start-angle    end-angle
text      "text"      angle     height     "justification"
text      "text"      angle     height     "justification"   "textstyle"
text  "text"  angle  height  "justification"  "textstyle"  xfactor  slant  offset _width  offset_height
repeat  num_repeats {   repeat_commands  repeat_draw_commands }
```

The **repeat** command is a positive integer *num_repeats*

and zero or more repeat_commands from the list

```
xpos      x-value
ypos      y-value
rfactor   value
```

and zero or more repeat_draw_commands where repeat_draw_commands include all the draw_commands except **repeat**.

The **set_up** commands for symbols will now be described followed by the **draw_commands.**

See

40.2.3.1 set_up_commands for symbols

40.2.3.2 Description of the draw_commands for Symbols

Or return to 40.2 Symbols

40.2.3.1 set_up_commands for symbols

group

The **group** is used to associate symbols in pop-up menus.

If the group is not defined, the symbol is placed in a default (blank) group.

If the group name includes imbedded spaces, then it must be enclosed in double quotes **"**.

factor

This command factors up/down the drawing co-ordinates, heights and radii.

xorigin, yorigin

The default **origin of the symbol** is defined to be the midpoint of the bounding box of the line style. The bounding box includes the points moved to and drawn to but does not automatically include (0,0). Hence the calculated origin may not be (0,0).

The **xorigin** and **yorigin** commands are used to override the default origin of the symbol.

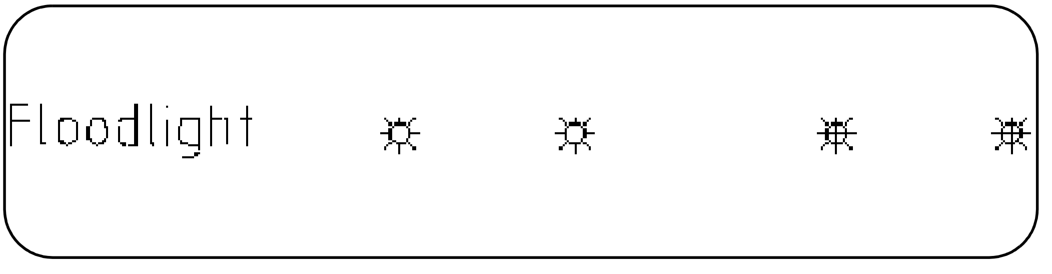
length

If length is omitted, then **length** is taken to be the horizontal size of the calculated bounding box of the symbol.

The **length** must always be positive but it can be larger or smaller than the symbols calculated horizontal size.

The origin of the symbol and the length are both used in positioning and sizing of the symbol.

Examples




```
// Floodlight
worldstyle FLOODLIGHT {
 group "Energex"
 mode vertex
 xorigin 0
 yorigin 0
 colour "yellow"
 circle 0.225
 move -.45 0
 draw -0.225 0
 move 0.45 0
 draw 0.225 0
 move 0 -.45
 draw   -0.225
 move 0 0.45
 draw 0 0.225
 move -.159 0.159
 draw -0.318 0.318
 move 0.159 0.159
 draw 0.318 0.318
```

```
 move 0.159 -0.159
 draw 0.318 -0.318
 move -0.159 -0.159
 draw -0.318 -0.318
 move 0 0.8
 text "FL" 0 0.5 "centre-middle"
}
```

Continue to 40.2.3.2 Description of the draw_commands for Symbols or return to 40.2.3 Symbol Definition or 40.2 Symbols.

40.2.3.2 Description of the draw_commands for Symbols

move *x-value* *y-value*

move the pen from the current pen position to the new position (x-value, y-value).

draw *x-value* *y-value*

draw from the current pen position to the new position (x- value, y-value)

rmove *dx-value* *dy-value*

move relative from the current pen position through the distance (dx-value, dy-value)

rdraw *dx-value* *dy-value*

draw relative from the current pen position through the distance (dx-value, dy-value)

colour *colour-name*

change colour (line styles can contains more than one colour).

If colour is not defined, the symbol is drawn in the colour of the string that the symbol is applied to. If the colour-name includes imbedded spaces, then it must be enclosed within double quotes **"**.

circle *radius*

draw a circle of the given radius at the current pen position.

After drawing a circle, the current pen position is left at the centre of the circle.

arc *radius* *start-angle* *end-angle*

draw an arc of the given radius from the start angle to the end angle. A positive radius denotes that the arc is drawn in a clockwise direction, a negative radius means anti-clockwise. Angles are given in degrees, minutes and seconds in the ddd.mmssfff format.

After drawing an arc, the current pen position is left at the centre of the arc.

text *text* *angle* *height* **"***justification***"**
text *text* *angle* *height* **"***justification***"** textstyle
or
text *text* *angle* *height* **"***justification***"** textstyle xfactor slant offset_width offset_height

the characters *text* are drawn at the current pen position with the given angle, height, justification and possibly textstyle, xfactor, slant, offset_width and offset_height. Angle and slant are given in degrees, minutes and seconds in the ddd.mmssfff format. Slant is between -45 and 45 degrees.

The allowed justifications are

top-left	top-centre	top-right
middle-left	middle-centre	middle-right
bottom-left	bottom-centre	bottom-right

and the value is enclosed inside double quotes (").

After drawing text, the current pen position is left where it was before the text was drawn.

repeat num_repeats { *repeat_commands* *repeat_draw_commands* }

The **repeat** command allows the set of repeat_draw_commands inside the braces (the repeat-style) to be re-drawn a number of times (num_repeats), and with each redraw, the size of the repeat-style is modified.

In the definition of repeat, there can be zero or more *repeat_commands* from the list

xpos *x-value*

ypos *y-value*

rfactor *value*

and zero or repeat_draw_commands where repeat_draw_commands include all draw_commands except **repeat**.

Unlike symbol, the repeat-style is defined in terms of a local origin at (0,0). The repeat-style is actually drawn with this origin at the position given by the values of **xpos** and **ypos**.

The repeat-style is drawn *num_repeats* times and on each redraw, the distances, heights and radii are factored by the amount

repeat_factor = (1 - i*rfactor)

where i = 0,1,...num_repeats-1

If rfactor is not specified, it is given the default value of 1/num_repeats. The factor is then

repeat_factor = (num_repeats - i)/num_repeats

where i = 0,1,...num_repeats-1

Examples of the Repeat Command

1. To produce circles of radius 10,9,8,...1 about the origin.

```
repeat 10 { circle 10}
```

2. To produce circles of radius 10,9.5,9,...5.5 about the origin

```
repeat 10 { rfactor 0.05 circle 10}
```

More Examples of World Styles

1. draw a circle of radius 20. The origin of the style is (0,0)

```
worldstyle "circle" {
  move 0 0
  circle 20
}
```

2. draw a circle of radius 20. The origin of the style is (100,100)

```
worldstyle "circle" {
  move 100 100
  circle 20
}
```

3. draw a circle of radius 20. The origin of the style is (0,0)

```
worldstyle "circle" {
  xorigin 0
  yorigin 0
  move 100 100
  circle 20
}
```

4. draw concentric circles of centre (0,0) inside a box

```
worldstyle "dot"  {
  xorigin 0
  yorigin 0
  move -5 -5
  draw -5 25
  draw 25 25
```

```
  draw 25 -5
  draw -5 -5
  repeat 10 {
    move 0 0
    circle 10
  }
}
```

5. draw an arc of absolute radius 10 in the clockwise direction from the angle 45 to the angle -45.

```
worldstyle "arc" {
  move 0 0
  arc 10 45 -45
}
```

6. draw an arc of absolute radius 10 in the anti-clockwise direction from the angle 45 to the angle -45.

```
worldstyle "arc 1" {
   move 0 0
  arc -10 45 -45
}
```

7. top-left justified text with a circle centred on the actual text position

```
worldstyle "top-left"  {
  group text
  move 0 0
  circle 4
  text   "<top left>" 0 10 "top-left"
}
```

Return to 40.2.3 Symbol Definition or 40.2 Symbols.

40.3 Textstyles and Fonts

Users can define any number of textstyles and specify how they are mapped to AutoCAD style names or Microstation (DGN, Intergraph) font numbers.

There is one fixed spaced textstyle called "**1**" (the default) which is identical to the textstyle used in earlier versions of 12d Model.

Each textstyle has a font definition (or font **1** as the default) which defines the stroking for each character in the font. True Type fonts are supported and most AutoCAD SHP files can be used to define the fonts.

Textstyles using True Type fonts or shape file which do not use arcs in their font definition, can also have a slant and a x_factor.

Textstyles can be used in text strings, 4d strings, super strings, linestyles, plots and user defined title blocks.

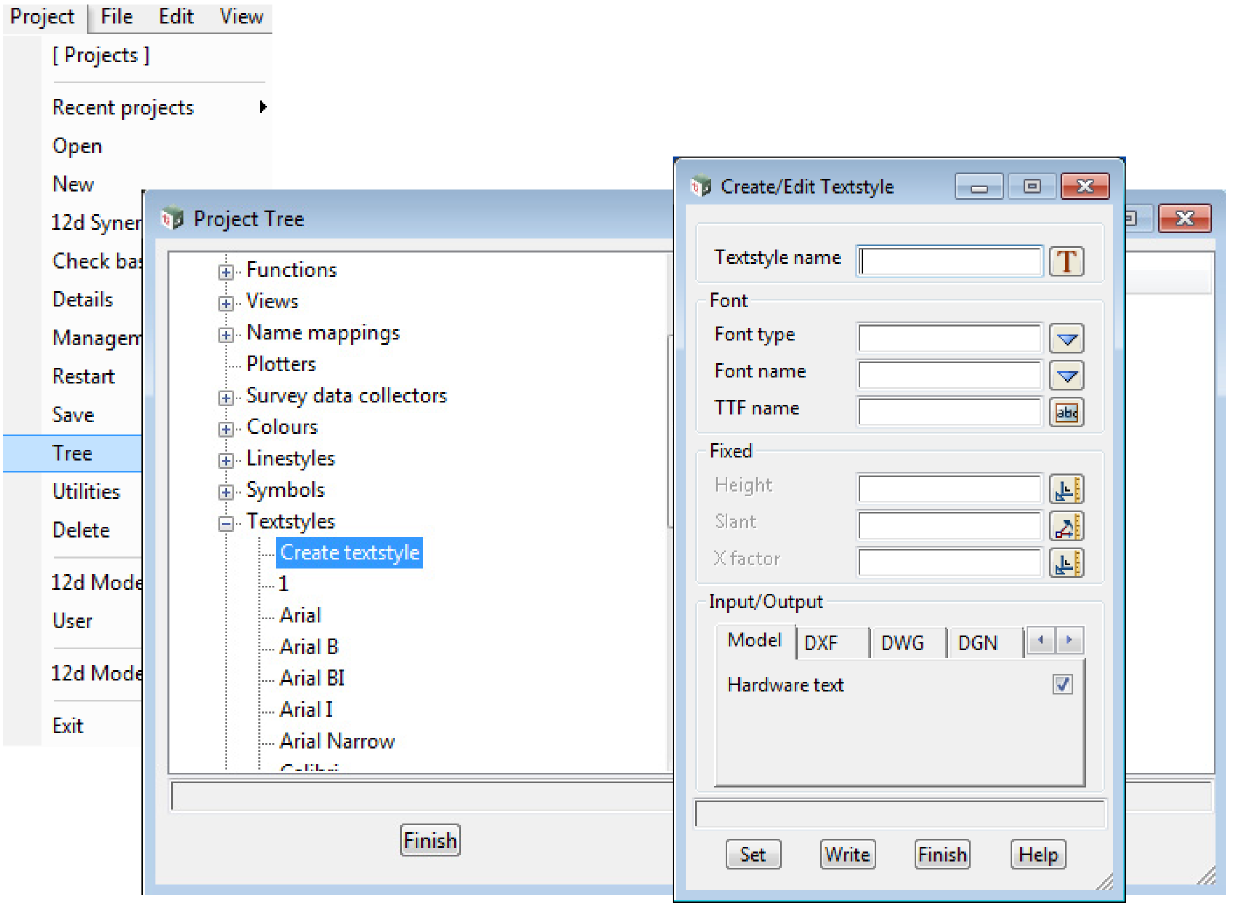
For further information about Textstyles and Fonts, see

40.3.1 Textstyles Definitions File

40.3.2 Fonts Definitions File

40.3.1 Textstyles Definitions File

Textstyles can be created/edited using the **Create/Edit Textstyle** panel which is brought up by double clicking on *Create textstyle* in the *Textstyles* expansion of the *Project Tree* (see 7.9.6 Textstyles) brought up by the **Project** =>**Tree** option.



The **Create/Edit Textstyle** panel acts as an interactive editor to a text file which contains the textstyle definitions. The full description of the text file will now be given.

The definitions for the available textstyles are normally stored in a file called **textstyl.4d** which is read in each time ***12d Model*** opens an existing project or created a new project. *Textstyl.4d* is searched for in the standard set up paths, or is pointed to by the environment variable

TEXTSTYLES_4D   filename

Inside textstyl.4d, a textstyle of a given name is defined by

```
textstyle   textstyle_name   {
        textstyle commands
}
```

Each textstyle must have a **name** and if the name includes imbedded spaces, it must be enclosed in double quotes **"**.

The textstyle itself is defined inside the braces where there can be zero or textstyle commands from the list

font_name   text

hardware_text_dxf
or   stroked_text_dxf

	output_name_dxf	text
	input_name_dxf	text
	hardware_text_dgn	
or	stroked_text_dgn	
	output_name_dgn	integer
	input_name_dgn	integer
	hardware_text_model	
or	stroked_text_model	
	stroked_text	
or	hardware_text	
	fixed_height	value
	variable_height	
	fixed_slant	value
	variable_slant	
	fixed_xfactor	value
	variable_xfactor	

The description of each command is

font_name text

Name of the font to be used to draw the characters of the textstyle. The stroking of the font characters can be defined in the file fonts.4d. If no font_name is given, or the stroking is not given in the font file, then the standard ***12d Model*** font is used ("1").

hardware_text_dxf or **stroked_text_dxf**

If stroked_text_dxf appears, then any text in this textstyle written out to dxf will be stroked - that is, broken into straights, lines and curves.

If hardware_text_dxf is in the file, then any text in this textstyle sent to dxf will be given as actual dxf text (not stroked).

output_name_dxf dxf_style_name

If text is sent to dxf as hardware text, the dxf style of the text will be dxf_style_name.

input_name_dxf dxf_style_name

When reading in dxf files, any text of style dxf_style_name will be given the 12d Model textstyle name textstyle_name.

The commands for reading and writing dgn files are similar to those for dxf except that dgn has font numbers.

hardware_text_dgn or **software_text_dgn**

output_name_dgn dgn_font_number
input_name_dgn dgn_font_number

hardware_text_model or **stroked_text_model**

Similarly, when writing out to a 12d Model model. For hardware_text_model, text will be sent to text with the same textstyle.

stroked_text or **hardware_text**

If stroked_text appears in the definition and there is no other hardware_text flag set, then the text will be stroked - that is, broken into straights, lines and curves.

If hardware_text appears in the definition, then the text is written out as hardware text.

By default, the height, slant and xfactor for the text can be set when text is defined however it is possible to set each of height, slant and xfactor to a fixed value.

	fixed_height	value
or	**variable_height**	

If fixed_height and a value are specified, then all text in this textstyle is drawn at a fixed height. If fixed_height is missing or variable_height (the default) appears, then each text string in the textstyle has its own height.

	fixed_slant	value
or	**variable_slant**	

If fixed_slant and a value are specified, then all text in this textstyle is drawn at a fixed slant. If fixed_slant is missing or variable_slant (the default) appears, then each text string in the textstyle has its own slant. Slant is given in degrees, minutes and seconds in the ddd.mmssfff format and must be between -45 and 45 degrees.

	fixed_xfactor	value
or	**variable_xfactor**	

If fixed_xfactor and a value are specified, then all text in this textstyle is drawn at a fixed xfactor. If fixed_xfactor is missing or variable_xfactor (the default) appears, then each text string in the textstyle has its own xfactor.

Example of a Textstyle File

```
textstyle  "1" {              // DGN FONT 0          ACAD FONT STANDARD

           output_name_dxf        STANDARD
           input_name_dxf         STANDARD
           input_name_dgn         "0"
           output_name_dgn        "0"
           stroked_text
           hardware_text_model
           hardware_text_dgn
           hardware_text_dxf
}
textstyle  ISO {              // DGN FONT 1          ACAD FONT ISO

           font_name              ISO
           input_name_dxf         ISO
           output_name_dxf        ISO
           input_name_dgn         "1"
           output_name_dgn        "1"
           stroked_text
           hardware_text_model
           hardware_text_dgn
           hardware_text_dxf
}
textstyle  MONO {             // DGN FONT 2          ACAD FONT MONO

           font_name       MONO
           output_name_dxf   MONOTXT
           input_name_dxf    MONOTXT
           input_name_dgn    "2"
           output_name_dgn   "2"
           stroked_text
           hardware_text_model
           hardware_text_dgn
```

```
                    hardware_text_dxf
}
textstyle  SCRIPT {             // DGN FONT 3              ACAD FONT SCRIPTC

                    font_name         SCRIPT
                    output_name_dxf   SCRIPTC
                    input_name_dxf    SRCIPTC
                    input_name_dgn    "3"
                    output_name_dgn   "3"
                    stroked_text
                    hardware_text_model
                    hardware_text_dgn
                    hardware_text_dxf
}
```

Continue to the next section 40.3.2 Fonts Definitions File for a description of the fonts file, or return to 40.3 Textstyles and Fonts.

40.3.2 Fonts Definitions File

The definition of any fonts referred to in the **textstyl.4d** file are given in the **fonts.4d** file which is searched for in the standard 12d Model set up paths, or is pointed to by the environment variable

FONTS_4D filename

The stroking for each character in the font (which is used when drawing the text on the screen and when required, stroking the text for outputs and plots) can be given in a separate file in the Autocad shape file format.

Inside **fonts.4d**, a font of a given name is defined by

```
font    font_name {
            font commands
}
```

Each font must have a name and if the name includes imbedded spaces, it must be enclosed in double quotes **"**.

The font itself is defined inside the braces where there can be zero or more font commands from the list

```
            stroke_file         filename
            proportional
or          fixed
```

The description of each command is

stroke_file stroke_file_name

The name of the file containing the stroking for each character in the font in AutoCAD ascii shape file format (.shp). The stroke_file_name is searched for in the standard ***12d Model*** set up paths.

If the stroking for a character is not defined, the standard ***12d Model*** character is used instead.

proportional or **fixed**

If a stroke file is not given, then the font could be a fixed (mono) or proportional font.

Example of a Font File

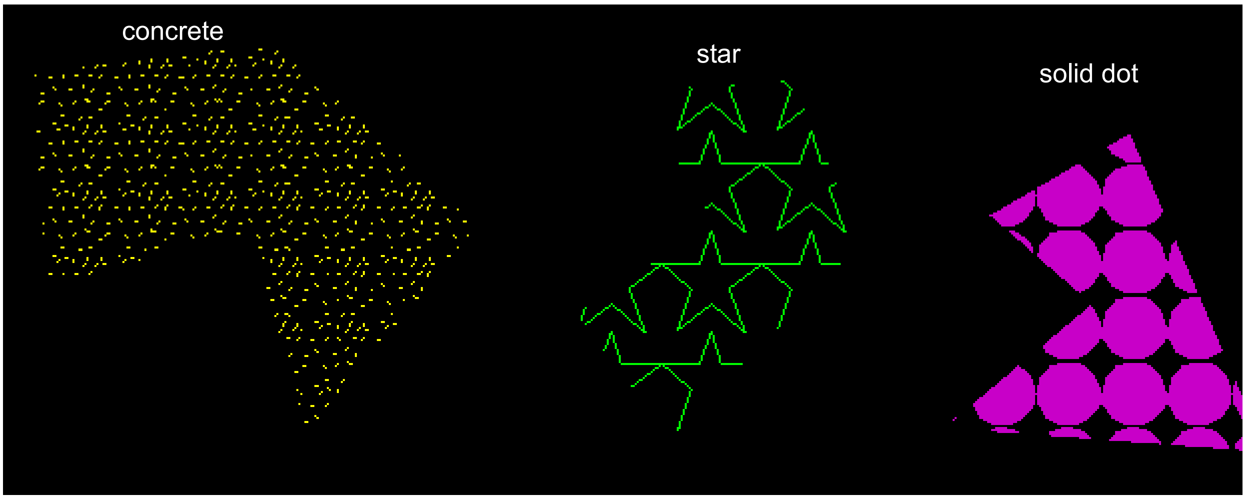
```
font    ISO {

            proportional
            stroke_file "ISO"
}
font    GOTHIC {

            proportional
            stroke_file "GOTHICE.SHP"
}
font    MONO {

            proportional
            stroke_file "MONOTXT.SHP"
}
```

Continue to the next section 40.3.2 Fonts Definitions File for a description of the fonts file, or return to 40.3 Textstyles and Fonts.

Return to 40.3 Textstyles and Fonts or 40 Linestyles, Symbols, Textstyles &Patterns.

40.4 Patterns

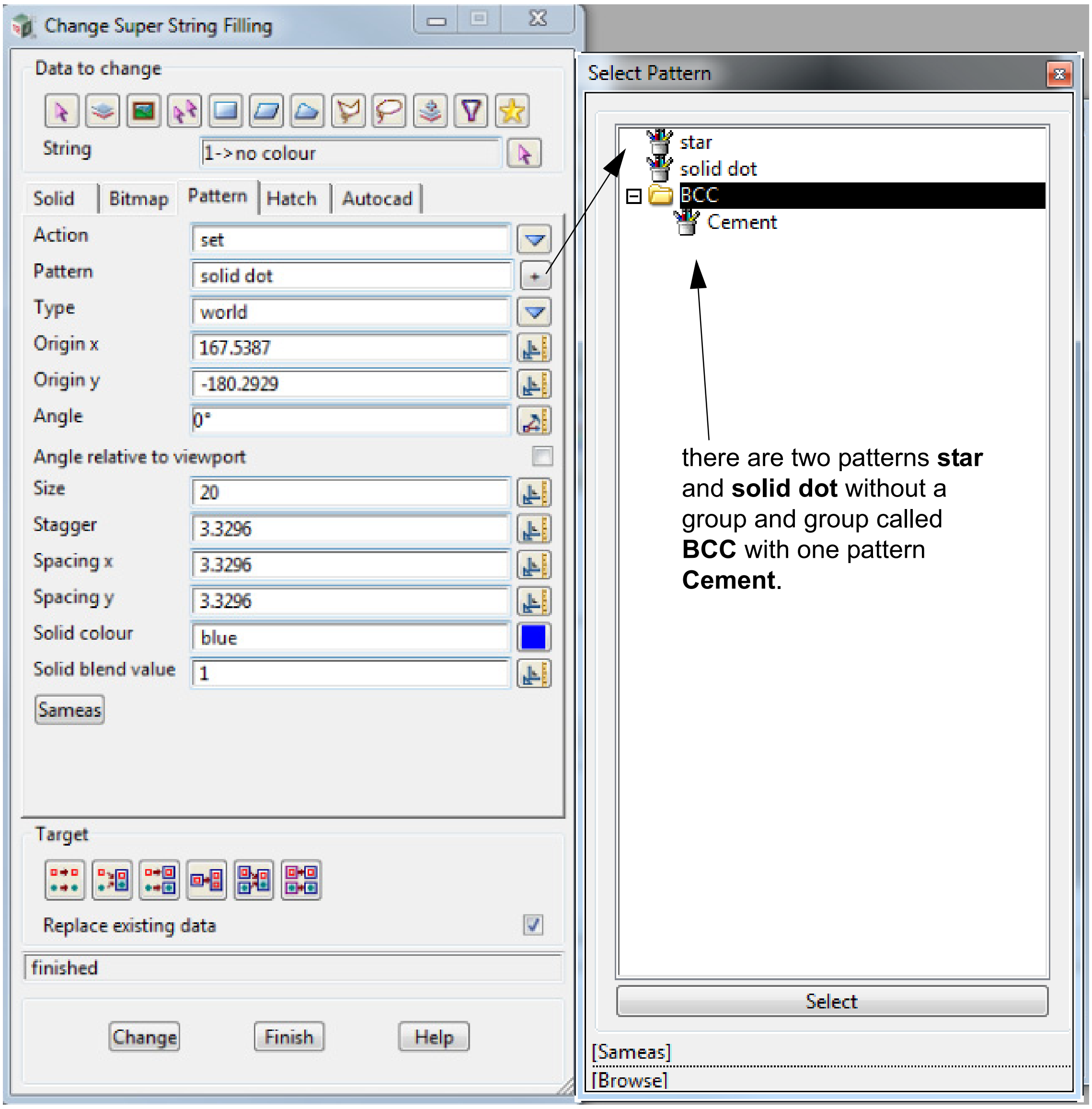
12d Pattern Fills provide the definitions of the patterns used for filling Super Strings.



The definitions of patterns are stored in a file called **patterns.4d** and users can add their own patterns to this file. See 40.4.1 Patterns.4d.

If a patterns file has been read, the pop-up for the **Pattern** field in the **Pattern** tab of the **Change Super String Filling** panel will list the patterns read from the file. See 14.12.9.5 Fills.

Any groups in the pattern file will be displayed as Folders in the pop-up list.



Continue to 40.4.2 12d Patterns Definitions File or return to 40.4 Patterns.

40.4.1 Patterns.4d

Some patterns are installed with ***12d Model*** and users can also define their own patterns.

The definitions of the available 12d patterns are stored in a file called **patterns.4d** which is read in each time ***12d Model*** opens an existing project or creates a new project.

patterns.4d is searched for in the standard set up paths, or the full path of file to use can be given by the environment variable

PATTERNS_4D *path_name_of_file*

The definition of the 12d patterns is given in the section 40.4.2 12d Patterns Definitions File.

Continue to 40.4.2 12d Patterns Definitions File or return to 40.4 Patterns.

40.4.2 12d Patterns Definitions File

The definitions for the available patterns is stored in a file called **patterns.4d** which is read in each time ***12d Model*** opens an existing project or created a new project. *Patterns.4d* is searched for in the standard set up paths, or is pointed to by the environment variable

PATTERNS_4D filename

There is no inbuilt 12d editor for patterns.4d and a text editor is used to create it.The full description of the text file will now be given.

The *patterns.4d* file consists of the word **patterns** followed by at least one space and the an open brace {, then the definitions of each patten and finally a closing brace }.

That is,

patterns {

definition_of_pattern_group_1

definition_of_pattern_group_2

. . .

definition_of_pattern_group_n

}

where each *definition_pattern_group_i* is the definition of a *group* of patterns.

The format of each *definition_of_pattern_group_i* is

(a) group **{**

name *group_name*

definition_of_pattern_1

definition_of_pattern_2

}

where ***group_name*** is the *name* of the group for all the patterns defined inside the *definition_of_pattern_group_1*, and ***definition_of_pattern_i*** is the definition of an individual pattern

or just

(b) *definition_of_pattern_i*

where ***definition_of_pattern_i*** is the definition of an individual pattern. In this case the pattern does not belong to any group.

data defining how the pattern draws (looks like).

Each *definition_of_pattern_i* is the definition of a single pattern and its format is

pattern **{**

name *pattern_name*

pattern_data_1

pattern_data_2

. . .

pattern_data_n

}

where ***pattern_name*** is the *name* of the pattern and each ***pattern_data_i*** is the data defining how part of the pattern draws (looks like).

Each *pattern_name* must be unique within the patterns.4d file and if the name includes imbedded spaces then the name must be enclosed in double quotes **"**.

The format of ***pattern_data_i*** is that of a super string (without pattern fills) in ***12da format*** except that in the super string definition:

```
string super {
```

is replaced by

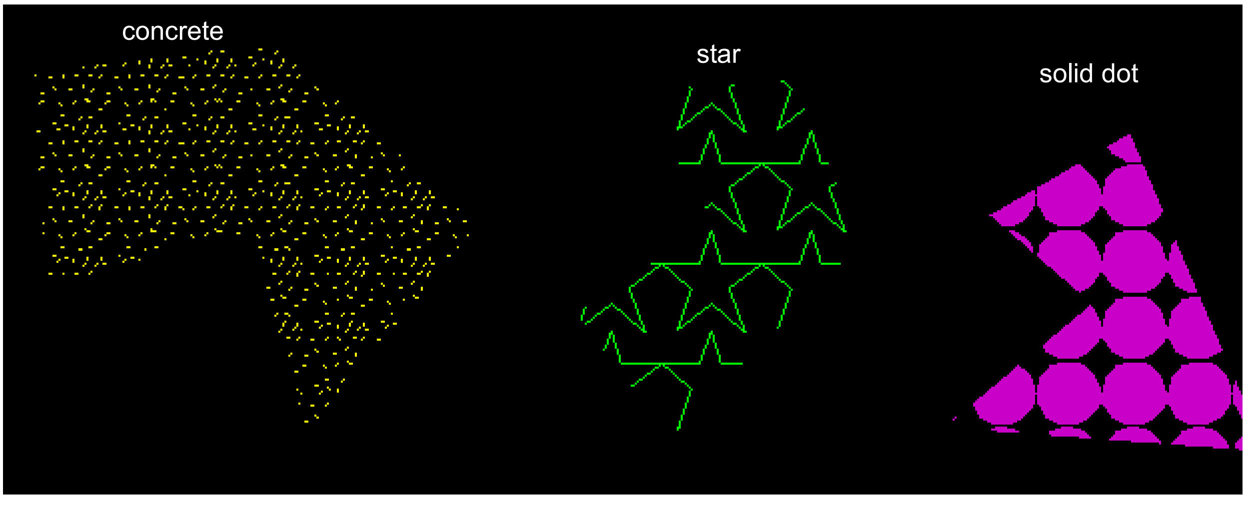
```
data {
```

The super strings must not include any patterns.

For the **12da** format of a super string, see 34.4.8 Super String.

So a ***definition_of_pattern_i*** consists of a a ***unique name*** and the drawing sections of 0 or more super strings.

An example of patterns and the ***patterns.4d*** file defining them is:




```
patterns {

  pattern {
    name "star"
    data {
      chainage  0
      breakline line
      colour    green
      style     1
      closed 1
      data_2d {
        0 40
        6.66666667 20
        30 20
        11.57894737 5.26315789
        20 -20
        0 -4
        -19.65412111 -19.72329689
        -11.57894737 5.26315789
        -30 20
        -6.66666667 20
      }
```

```
    }
  }
  pattern {
    name "solid dot"
    data {
      chainage  0
      breakline line
      colour    yellow
      style     1
      closed 1
      data_2d {
        -10 0
        10 0
      }
      radius_data {
        10 10
      }
      major_data {
        f f
      }
      solid_fill {
        colour      "rgb(200,0,200)"
      }
    }
  }

  group {
    name "BCC"
    pattern {
      name "Cement"
data {
  name      ""
  chainage  0
  breakline line
  colour    yellow
  style     1
   weight    0.25
  closed 0
  z 0
  data_2d {
    0.13875 0.10125
    0.1625 0.184375
  }
}
data {
  name      ""
  chainage  0
  breakline line
  colour    yellow
  style     1
   weight    0.25

  closed 0
  z 0
  data_2d {
    0.22624999 0.48874999
    0.30562499 0.46124999
  }
```

```
}
data {
  name      ""
  chainage  0
  breakline line
  colour    yellow
  style     1
   weight    0.25

  closed 0
  z 0
  data_2d {
    0.155 1.08812498
    0.22624999 1.03249998
  }
}
data {
  name      ""
  chainage  0
  breakline line
  colour    yellow
  style     1
   weight    0.25

  closed 0
  z 0
  data_2d {
    0.32937499 1.38312497
    0.37687499 1.43812497
  }
}
data {
  name      ""
  chainage  0
  breakline line
  colour    yellow
  style     1
   weight    0.25

  closed 0
  z 0
  data_2d {
    0.24999999 1.96374996
    0.32124999 1.97312496
  }
}
data {
  name      ""
  chainage  0
  breakline line
  colour    yellow
  style     1
   weight    0.25

  closed 0
  z 0
  data_2d {
    0.44812499 2.24999995
```

```
    0.51187499 2.31437495
  }
}
data {
  name      ""
  chainage  0
  breakline line
  colour    yellow
  style     1
   weight    0.25

  closed 0
  z 0
  data_2d {
    0.71749998 2.30499995
    0.76562498 2.28687495
  }
}
data {
  name      ""
  chainage  0
  breakline line
  colour    yellow
  style     1
   weight    0.25

  closed 0
  z 0
  data_2d {
    0.65437499 1.98249996
    0.58312499 1.88999996
  }
}
data {
  name      ""
  chainage  0
  breakline line
  colour    yellow
  style     1
   weight    0.25

  closed 0
  z 0
  data_2d {
    1.09812498 1.90874996
    1.17749997 1.88124996
  }
}
data {
  name      ""
  chainage  0
  breakline line
  colour    yellow
  style     1
   weight    0.25

  closed 0
  z 0
```

```
    data_2d {
      0.86874998 1.34624997
      0.92374998 1.31874997
    }
  }
  data {
    name      ""
    chainage  0
    breakline line
    colour    yellow
    style     1
     weight    0.25

    closed 0
    z 0
    data_2d {
      0.89249998 1.00499998
      0.80499998 0.97749998
    }
  }
  data {
    name      ""
    chainage  0
    breakline line
    colour    yellow
    style     1
     weight    0.25

    closed 0
    z 0
    data_2d {
      0.55937499 0.76562498
      0.62249999 0.74687498
    }
  }
  data {
    name      ""
    chainage  0
    breakline line
    colour    yellow
    style     1
     weight    0.25

    closed 0
    z 0
    data_2d {
      0.63062499 0.110625
      0.64624999 0.156875
    }
  }
  data {
    name      ""
    chainage  0
    breakline line
    colour    yellow
    style     1
     weight    0.25
```

```
    closed 0
    z 0
    data_2d {
      1.24124997 0.24937499
      1.30437497 0.22125
    }
  }
  data {
    name       ""
    chainage   0
    breakline line
    colour    yellow
    style     1
     weight    0.25

    closed 0
    z 0
    data_2d {
      1.67749996 0.175
      1.72499996 0.138125
    }
  }
  data {
    name       ""
    chainage   0
    breakline line
    colour    yellow
    style     1
     weight    0.25

    closed 0
    z 0
    data_2d {
      1.61374996 0.52562499
      1.56624996 0.51624999
    }
  }
  data {
    name       ""
    chainage   0
    breakline line
    colour    yellow
    style     1
     weight    0.25

    closed 0
    z 0
    data_2d {
      1.55062497 0.89437498
      1.49499997 0.94062498
    }
  }
  data {
    name       ""
    chainage   0
    breakline line
    colour    yellow
    style     1
```

```
     weight    0.25

    closed 0
    z 0
    data_2d {
      1.32062497 1.47499997
      1.26499997 1.51187497
    }
  }
  data {
    name      ""
    chainage  0
    breakline line
    colour    yellow
    style     1
     weight    0.25

    closed 0
    z 0
    data_2d {
      1.39187497 2.35124995
      1.44749997 2.37874995
    }
  }
  data {
    name      ""
    chainage  0
    breakline line
    colour    yellow
    style     1
     weight    0.25

    closed 0
    z 0
    data_2d {
      1.63749996 1.95499996
      1.61374996 1.89937496
    }
  }
  data {
    name      ""
    chainage  0
    breakline line
    colour    yellow
    style     1
     weight    0.25

    closed 0
    z 0
    data_2d {
      1.96249996 2.44312495
      1.99437496 2.37874995
    }
  }
  data {
    name      ""
    chainage  0
    breakline line
```

```
    colour    yellow
    style     1
     weight    0.25

    closed 0
    z 0
    data_2d {
      2.33499995 2.28687495
      2.30374995 2.21312495
    }
  }
  data {
    name      ""
    chainage  0
    breakline line
    colour    yellow
    style     1
     weight    0.25

    closed 0
    z 0
    data_2d {
      2.02624995 1.95499996
      2.08937495 1.90874996
    }
  }
  data {
    name      ""
    chainage  0
    breakline line
    colour    yellow
    style     1
     weight    0.25

    closed 0
    z 0
    data_2d {
      2.32749995 1.63187496
      2.30374995 1.56749996
    }
  }
  data {
    name      ""
    chainage  0
    breakline line
    colour    yellow
    style     1
     weight    0.25

    closed 0
    z 0
    data_2d {
      1.89937496 1.42937497
      1.83562496 1.43812497
    }
  }
  data {
    name      ""
```

```
    chainage  0
    breakline line
    colour    yellow
    style     1
     weight    0.25

    closed 0
    z 0
    data_2d {
      2.01812495 1.03249998
      1.97874996 0.97749998
    }
  }
  data {
    name      ""
    chainage  0
    breakline line
    colour    yellow
    style     1
     weight    0.25

    closed 0
    z 0
    data_2d {
      2.19249995 0.47062499
      2.14499995 0.41499999
    }
  }
  data {
    name      ""
    chainage  0
    breakline line
    colour    yellow
    style     1
     weight    0.25

    closed 0
    z 0
    data_2d {
      2.40687495 1.05124998
      2.37499995 1.12499997
    }
  }
      } // end of pattern Cement
     }  // end of BCC group
  }   // end of patterns
```

41 Functions Keys, Menus, Toolbars

This appendix contains information about how user defined function keys, menus and toolbars are defined and used in ***12d Model***.

See

41.1 User Defined Function Keys
41.2 User Defined Menus
41.3 User Defined Toolbars

41.1 User Defined Function Keys

The user can define function keys (with possible modifiers) to be used to

(a) bring up most ***12d Model*** menus or panels

(b) toggle certain snaps on or off.

(c) start some geometry snaps

(d) read a layout file

(e) execute a batch file or a program (system call).

(f) run 12d Model 12dPL's (macros) or chains

The function key definitions are given in the file

userkeys.4d

which is searched for in the standard set up areas (local, USER_4D, user, set_ups etc.) or is pointed to by the environment variable

FUNCTION_KEYS_4D   file   // function key definitions

The format of the function_keys file is simply a list of function key definitions with only one per lines. Blank lines in the file are ignored and anything on a line after a // is a comment.

The function key definition is

MODIFIERS   FUNCTION_KEY   OPERATION

where MODIFIERS can be totally missing or any combination of

shift
control
or alt

FUNCTION_KEY is one of

f1, f2,...f12 (**f7 should be avoided since it is used in 12d Model for typed inpu**t,
f1 & f10 should be avoided since they are reserved by Windows NT)

and OPERATION is one of

menu	menu_name	// raise a menu
panel	panel_name	// raise a panel
snap	toggle_snap	// list of toggle_snaps given below
snap	cogo_snap	// list of cogo_ snaps given below
cogo_command		// list of cogo_ commands given below
layout	layout_file_name	// run a screen layout file
system	batch_file	// run a batch file
system	program	// run a program

or for macros

macro	*options*	user_macro_name	*macro_arguments*

where the macros options are:

-no_console	// don't display macro console
-close_on_exit	// remove console when macro terminates
-buttons	// have buttons for *finish*, *restart* and *quit* on console
-allow_defaults	// allow default answers for console questions

The default when there are no macro options is to run the macro with a console but without buttons, and to leave the macro console on the screen when the macro terminates.

For information on how to use each item with function keys, see

Menus and Panels
View Menus and Panels
View Toggle Menus
Available Snaps to be Toggled
Available Cogo Snaps
Available Cogo Commands
Layout Files
System Calls
12dPLs (Macros)
Chains

For using $LIB and $User_Lib, see $LIB and $USER_LIB.
For an example of *userkeys.4d*, see Example of a User Function Key File.

Menus and Panels

Menus and panels can be programmed by functions keys by simply giving the name of the menu and panel after the **panel** command.

For example,

f5	panel	"String Inquire"	// this is a panel
f6	panel	"String Create"	// this is a menu

Note: the names for each menus and panel (together) must have unique names so there is no confusion about what is to be run by just giving the name after the command **panel**.

View Menus and Panels

Menus and panels for views can be programmed by functions keys by substituting the text $PLAN_VIEW, $SECTION_VIEW or $PERSPECTIVE_VIEW in place of the view name in the panel name or menu name.

For example,

f9	panel	"Plan Plot $PLAN_VIEW".

A panel for each view type can be programmed for the same modifier and function key combination. If the cursor is over the appropriate view type when the function key is selected, then the panel is fired up.

For example, the three definitions for f5

f5	panel	"Section Plot $SECTION_VIEW"
f5	panel	"Plan Plot $PLAN_VIEW"
f5	panel	"Perspective Plot $PERSPECTIVE_VIEW"

means that a panel will come up when f5 is selected over any of the three view types.

View Toggle Menus

The **toggle menus** available on each view can be set to a function key by

MODIFIERS	FUNCTION_KEY	menu	"Toggle $PLAN_VIEW"
MODIFIERS	FUNCTION_KEY	menu	"Toggle $SECTION_VIEW"
MODIFIERS	FUNCTION_KEY	menu	"Toggle $PERSPECTIVE_VIEW"

As for the view menus, the cursor must be over the appropriate view type when the function key is selected.

For example,

shift f8 menu "Toggle $PLAN_VIEW"

brings up the plan view toggle menu if shift f8 is selected when in a plan view.

Available Snaps to be Toggled

toggle_point_snap
toggle_line_snap
toggle_grid_snap
toggle_cursor_snap
toggle_tin_snap
toggle_input_height_snap
toggle_show_information

For example,

control f2 toggle_point_snap

Available Cogo Snaps

snap create_edit
snap line_create_edit
snap tangent_items_edit
snap fillet_3_points_edit
snap fillet_item_item_cp_edit
snap fillet_item_item_radius_cp_edit
snap locate_divide_edit
snap locate_divide_distance_edit
snap parallel_edit
snap intersect_edit
snap intersect_offset_edit
snap bearings_intersect_edit
snap between_points_edit
snap locate_edit
snap locate_chainage_edit
snap locate_offset_edit
snap locate_deflection_edit
snap locate_drop_perpendicular_edit
snap locate_projection_edit
snap locate_two_point_distances_edit

For example,

shift f1 snap locate_chainage_edit

Available Cogo Commands

All the new cogo commands can be activated by a function key.

create_edit
line_create_edit
tangent_items_edit
fillet_3_points_edit
fillet_item_item_cp_edit
fillet_item_item_radius_cp_edit

locate_divide_edit
locate_divide_distance_edit
parallel_edit
intersect_edit
intersect_offset_edit
bearings_intersect_edit
between_points_edit
locate_edit
locate_chainage_edit
locate_offset_edit
locate_deflection_edit
locate_drop_perpendicular_edit
locate_projection_edit
locate_two_point_distances_edit

For example,

f2 between_points_edit

Layout Files

Layout files can be called up using the **layout** command.

For example,

f8 layout "/temp/contours.slf"

System Calls

Programs and batch files can be called up using the **system** command. The directories defined by the environment variables $LIB_4D and $USER_4D are included in the PATH environment variables so that any batch files or executables in these areas will be accessible without including any path name in the system call.

For example,

```
f4          system    special.exe           // run the program special.exe
shift f4    system    batch.bat             // run a batch file
control f4  system    "start cmd /k  dir"   // do a dir in a command shell
```

12dPLs (Macros)

12d supplied and User 12dPLs (macros) can be run and the macros brought up with or without buttons, and with leaving or removing the macro console when the macro has finished.

The macro command structure and options are:

macro *options* user_macro_name *macro_arguments*

where the macros options are:

-no_console // don't display macro console
-close_on_exit // remove console when macro terminates
-buttons // have buttons for *finish*, *restart* and *quit* on console
-allow_defaults // allow default answers for console questions

The default when there are no macro options is to run the macro with a console but without buttons, and to leave the macro console on the screen when the macro terminates.

For example,

```
f8    macro  -no_console  -close_on_exit"/area/drape_align.4do"
f9    macro   -buttons"/games/lots.4do"
```

For compatibility with earlier versions of ***12d Model***, the following macro commands are also supported:

macro user_macro_name // run macro without buttons,

		// leave console panel on exit
macro_2	user_macro_name	// run macro without buttons,
		// remove console panel on exit
macro_with_buttons	user_macro_name	// run macro with buttons,
		// leave console panel on exit
macro_with_buttons_2	user_macro_name	// run macro with buttons,
		// remove console panel on exit

Chains

12d supplied and User chains can be run by using the **chain** command.

chain chain_name

For example,

f8 chain "tri_cont_label.chain"

$LIB and $USER_LIB

The variable $LIB and $USER_LIB can be used as part of the layout_file_name and user_macro_name to pick up files from either the library or user library.

For example,

f7	macro	"$LIB/drape_align.4do"
f8	macro	"$USER_LIB/lots.4do"
f9	layout	"$LIB/contours.slf"

Note - if $LIB_4D is not defined, then 12dmodel/6.00/library is used. If $USER_LIB_4D is not defined, then 12dmodel/6.00/user_lib is used

Example of a User Function Key File

// File to define function key usage
// Note: f1, f6 & f10 are reserved by Windows NT
// f7 is reserved by 4D Solutions for typed input
// Available function key modifiers are

// shift control alt

// Note: some alt key combinations are reserved by Windows NT

// Some sample lines may be

// f5 toggle_point_snap
// shift f5 toggle_line_snap
// control shift f7 toggle_input_height_snap

// Note that all function key modifiers must come before the function key itself.

// ---

// **snaps**

f3 toggle_point_snap
f4 toggle_line_snap
f5 toggle_grid_snap
f5 toggle_cursor_snap
f8 toggle_input_height_snap
// **snaps cogo**

f9 snap intersect_offset_edit
f11 snap locate_chainage_edit
f12 snap locate_drop_perpendicular_edit

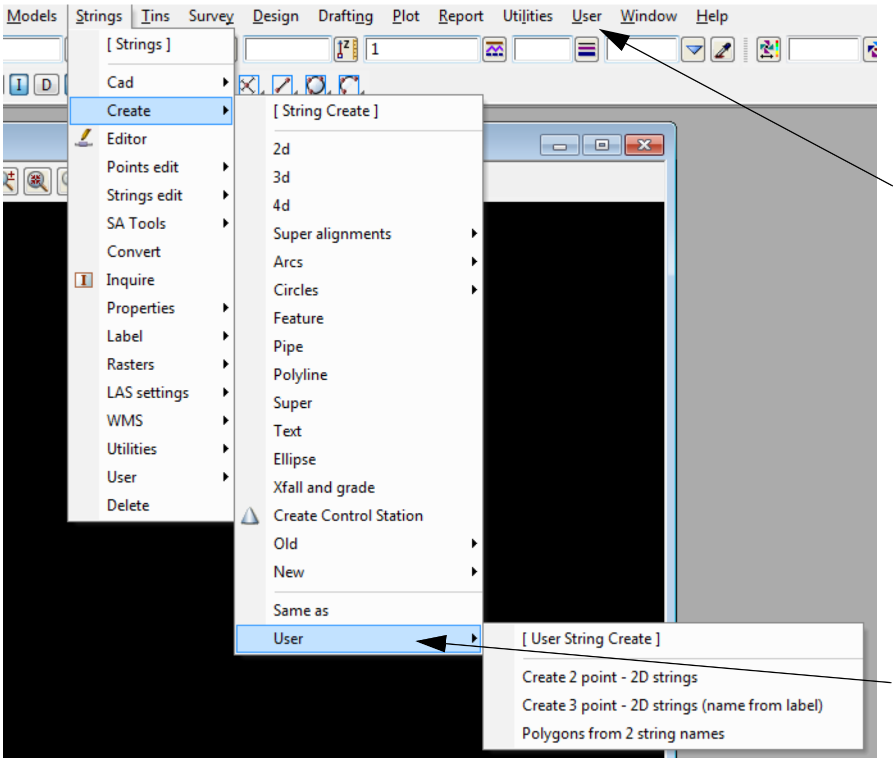
// **panels**

```
f2          panel  "String Inquire"

shift f1   panel "Edit String"
shift f2   panel "String Delete"
shift f3   panel "Clean Model"
shift f4   panel "Delete Model"
shift f5   panel "Strings Edit Ops"
shift f6   panel "Points Edit Ops"
shift f7   panel "String Attributes"
shift f8   panel "Toggle $PLAN_VIEW"

// layout files

control f4 layout "$LIB/tri_contour_clean.slf"

// user macros

control f5 macro  -no_console  -close_on_exit  "$LIB/Culd.4do"

// ---------------------------------------------------
```

Continue to the next section 41.2 User Defined Menus or return to 41 Functions Keys, Menus, Toolbars.

41.2 User Defined Menus

To help customise ***12d Model***, there is a pull down menu **User** on the Main menu and **User** menus on each of the pull down menus on the Main menu, and some other Menus.



User on the main menu

User String Create on the String Create menu

The **User** menus can run 12d Model macros, external programs, chains, and bring up ***12d Model*** screen layout files, panels and menus.

User menus can have walk-right menus, and any walk-right menus can have further walk-right menus so **User** menus can be used to build sophisticated tailored additions to ***12d Model***. Each of the User Defined menus are also floating (tear away) menus just like standard***12d Model*** menus.

All the **User** menus are defined in the two files: *Usermenu.4d*, which is created by the user, and *Xtramenu.4d*, which supplied by ***12d Solutions***.

When displayed, each **User** menu is divided into two sections with menu options supplied by ***12d Solutions*** in the file *Xtramenu.4d* on the top, and any options supplied by the user (in the file *Usermenu.4d)* underneath.

The default name for the User Defined Menus definition file is

usermenu.4d

which is searched for in the standard set up areas (local, USER_4D, user, set_ups etc.) or set by the environment variable

USER_OPTIONS_4D file // Customers User menu definition

The default name for the 12d Solutions definition file which includes some macro for **User** is

xtramenu.4d

which is also searched for in the standard set up areas (local, USER_4D, user, set_ups etc.) or set by the environment variable

EXTRA_OPTIONS_4D file // 12D Solutions User menu definitions

The format for the User Defined Manu's file s given in the next section 41.2.1 Full Definition of User Menus.

Important Note

The file *xtramenu.4d* is for use by ***12d Solutions*** only. Please do not modify it because it may be over written in future updates. The file *usermenu.4d* is for Customer use.

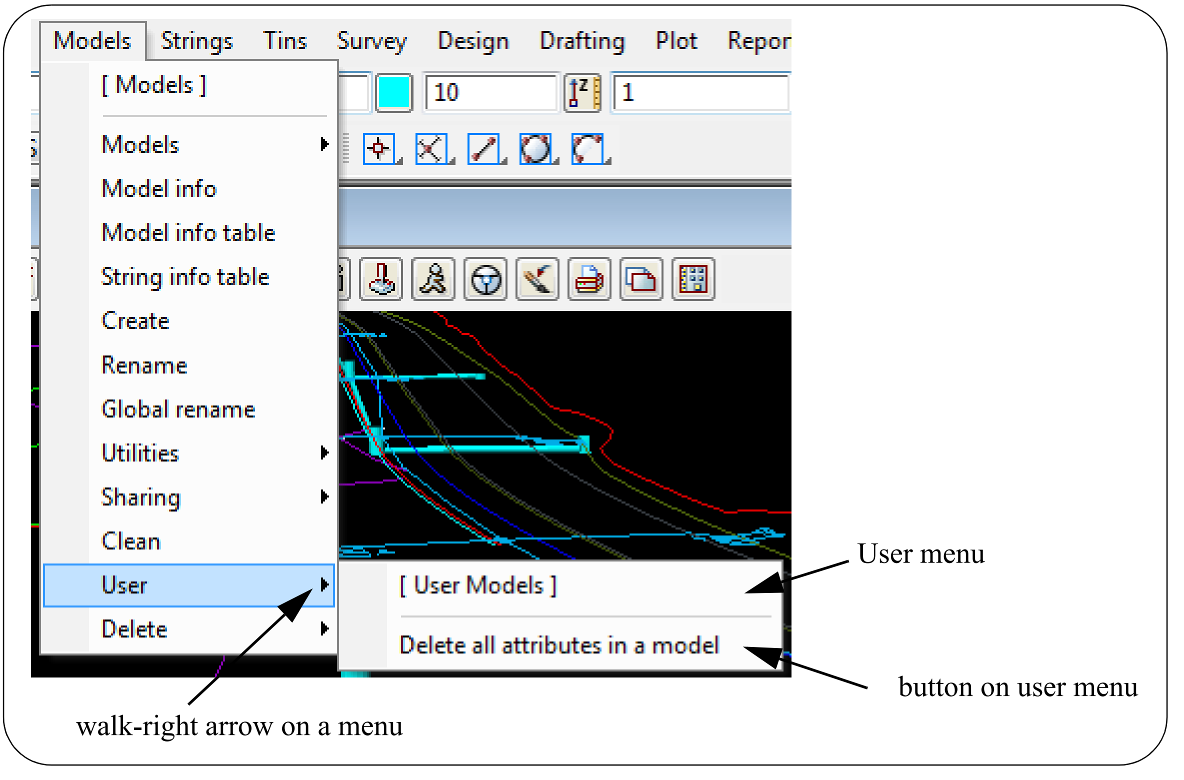
See

41.2.1 Full Definition of User Menus
41.2.2 Using $LIB and $USER_LIB in User Menus
41.2.3 Placing User Defined Menus on User and Other 12d Menus
41.2.5 Example of a User Defined Menu

41.2.1 Full Definition of User Menus

The ***User Defined Menus*** are made up of *buttons* which have text on them and when the left hand mouse button (LB) is pushed down and released on a button, a *command* is executed.

A button can also have a walk-right menu which appears when the left hand mouse button (LB) is moved over the arrow on the right had side of the button. Walk-right menus can have walk-right menus.



The ***text*** for each button in the **User** menus, plus the action taken when the button is selected, is user specified. Any of the buttons can include further walk-right menus.

When a button is **selected**, the action can be to

(a) run a ***12d Model*** macro

(a) run a ***12d Modell*** chain

(b) bring up a***12d Model*** menu or User Defined menu

(c) bring up a***12d Model*** panel

(d) read a screen layout file

(e) execute a batch file or program (system call)

In the *Usermenu.4d* file, a menu is defined by:

```
Menu menu_name  {
                button_1
                button_2
                ...
                button_n
}
```

where a button can include one or both of the keywords *Command* and *Walk_Right*, and the syntax for a button is

```
Button button_name {
            Walk_Right          menu_name_1
and/or      Command             command_name
}
```

The *text* displayed on the button is ***button_name*** and the definitions of the *Walk_Right* and *Command* commands are:

Walk_Right Menu for Buttons

The ***Walk_Right*** keyword specifies that there *is* a walk-right menu for the button and gives the *name* of the menu that is displayed when LB is moved over the walk-right arrow on the button.

The *Walk_Right* command consists of the word *Walk_Right* followed by one or more spaces and then the name of the menu that is displayed when over the walk-right arrow.

Walk_Right ***menu_name***

where *menu_name* is the name of another menu ***defined elsewhere*** in the *usermenu.4d* file.

When the *Walk_Right* keyword exists, a walk-right arrow is created on the button.

If there is no Walk_Right keyword, then there is no walk right arrow on the button.

A *Walk_Right* line is optional.

NOTE - walk-right menus *can* contain walk-right menus.

Commands for Buttons

The ***Command*** keyword defines what action occurs if **LB is clicked on the button.**

The *Command* consists of the word *Command* followed by one or more spaces and then the name of the command and any arguments it requires.

Command ***"command_name command_arguments"***

The ***command_name*** and ***command_arguments*** can do any of:

(a) Run a User Defined or 12d Supplied 12d Model 12dPL (Macro)

Command "**macro** *macro_options* user_macro_name *macro_parameters*" // run macro

where the *macros_options* are:

```
-no_console          // don’t display macro console
-close_on_exit       // remove console when macro terminates
-buttons             // have buttons for finish, restart and quit on console
-allow_defaults      // allow default answers for console questions
```

The default when there are no macro options is to run the macro with a console but without buttons, and to leave the macro console on the screen when the macro terminates.

For example

```
Button "Write Levels" {
   Command "macro Tinval.4do"
}
Button "ADAC - show attributed/not attributed" {
Command "macro -no_console -close_on_exit Adac_show_panel.4do"
```

```
}
```

(b) Run a chain

Command "**chain** *chain_file_name*"

It is also possible to pass a chain parameter value file (pvf) to be used by the chain through to the chain. The command is

Command "chain *-pvf pvf_file_name chain_file_name*"

where ***pvf_file_name*** is the name of the **pvf** file to use when the chain *chain_file_name* is run.

For example

```
Button "Survey to ADAC 41 chain" {
   Command "chain -pvf $USER/BCC/ADAC_BCC_41.pvf $LIB/ADAC_survey_base.chain"
}
```

Also see 28.3.5.4 Running Chains from User Menus and Toolbars

(c) Bring up a *12d Model* menu and/or a User Defined Menu

Command "12d_model_menu_name"

or

Command "user_defined_menu_name"

The *12d_model_menu_name* is simply the name on the top of the ***12d Model*** menu.

The *user_defined_menu_name* is the name of a User Defined menu that is defined further down in the usermenu.4d file.

For example, for the menu called "Model Utilities", the

```
Button "Bring up the Model Utilities menu" {
   Command "Model Utilities"
}
```

Notice that for a menu, only the menu name in quotes follows the keyword *Command*.

Note: - this works because all the names must be unique amongst the ***12d Model*** menus, *12d* ***Model*** panels and *User Defined* menus.

A menu can occur as a *Walk_Right* menu and/or as a *Command*.

(d) Bring up a*12d Model* panel

Command "12d_model_panel_name"

Every ***12d Model*** menu and panel has a unique name and the *12d_model_panel_name* is simply the name on the top of the panel.

For example, for the menu called "Test Helmert 2d (Advanced) panel", the

```
Button "Bring up a Helmert panel" {
   Command "Test Helmert 2d (Advanced)"
}
```

Notice that for a panel, only the panel name in quotes follows the keyword Command.

Note: - this works because all the names must be unique amongst the ***12d Model*** menus, *12d* ***Model*** panels and *User Defined* menus.

(e) Run a User Defined screen layout file

Command "**layout** layout_file_name"

For example

```
Button "Generate ADAC Tree Spreadsheet" {
   Command "layout ADAC_Tree_Report.slx"
}
```

(f) Execute a batch file or program

Command "**system** program_or_batch_file_name"

Notes

1. A Command Line is optional
2. A button can have both a **Walk_Right** and a **Command** defined for it.
3. If there is only a *Walk_Right* menu and no *Command*, clicking on the button will bring up the **Walk_Right** menu.
4. Both *Walk_Right* and *Command* can be left out.

For example

```
Button "------------------------------" {
}
Button "" {
}
```

will create a button with the text "------------------------------" on it and a button with no text on it.

5. in the *usermenu.4d* file, blank lines are ignored and anything on a line after a **//** is a comment

Continue to the next section 41.2.2 Using $LIB and $USER_LIB in User Menus or return to 41.2 User Defined Menus.

41.2.2 Using $LIB and $USER_LIB in User Menus

The variable $LIB and $USER_LIB can be used in Commands as part of the user_macro_name, chain_file_name, layout_file_name and programe_or_batch_file_name to pick up files from either the *12d Library* or the *User* library.

Note

If $LIB_4D is not defined, then 12dmodel/version_number.00/library is used.

If $USER_LIB_4D is not defined, then 12dmodel/version_number.00/user_lib is used

For example, the definition of a menu called "Lots" could be:

```
Menu "Lots" {
  Button "Create lots" {                          // button called "Create lots" which
    Command "macro -close_on_exit  $USER_LIB/subdiv.4do"  // fires up a 12d Model macro
  }
  Button "Roads" {                    // button with walk-right menu "Roads" and layout file
    Walk_Right "Roads"                            // walk-right menu "Roads"
    Command "layout $USER_LIB/road.slx"           // button invokes a layout file if selected
  }
}
```

Continue to the next section 41.2.3 Placing User Defined Menus on User and Other 12d Menus or return to 41.2 User Defined Menus.

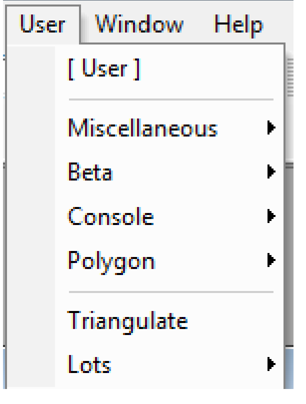
41.2.3 Placing User Defined Menus on User and Other 12d Menus

User Defined Buttons on Main User Menu

The place buttons or menus in the bottom of the **User** menu on the Main menu, put the buttons in a menu called **User** in the usermenu.4d file.

For example

```
Menu "User" {
   Button "Triangulate" {                // button called Triangulate which brings up
      Command "Triangulate a View"       // the "Triangulate a View" panel
    }
   Button "Lots" {                       // button called Lots with its
      Walk_Right "Lots"                  // own walk-right menu called Lots
      Command "Triangulate a View"       // plus clicking on the button brings up the
   }                                     // the "Triangulate a View" pane
}
```



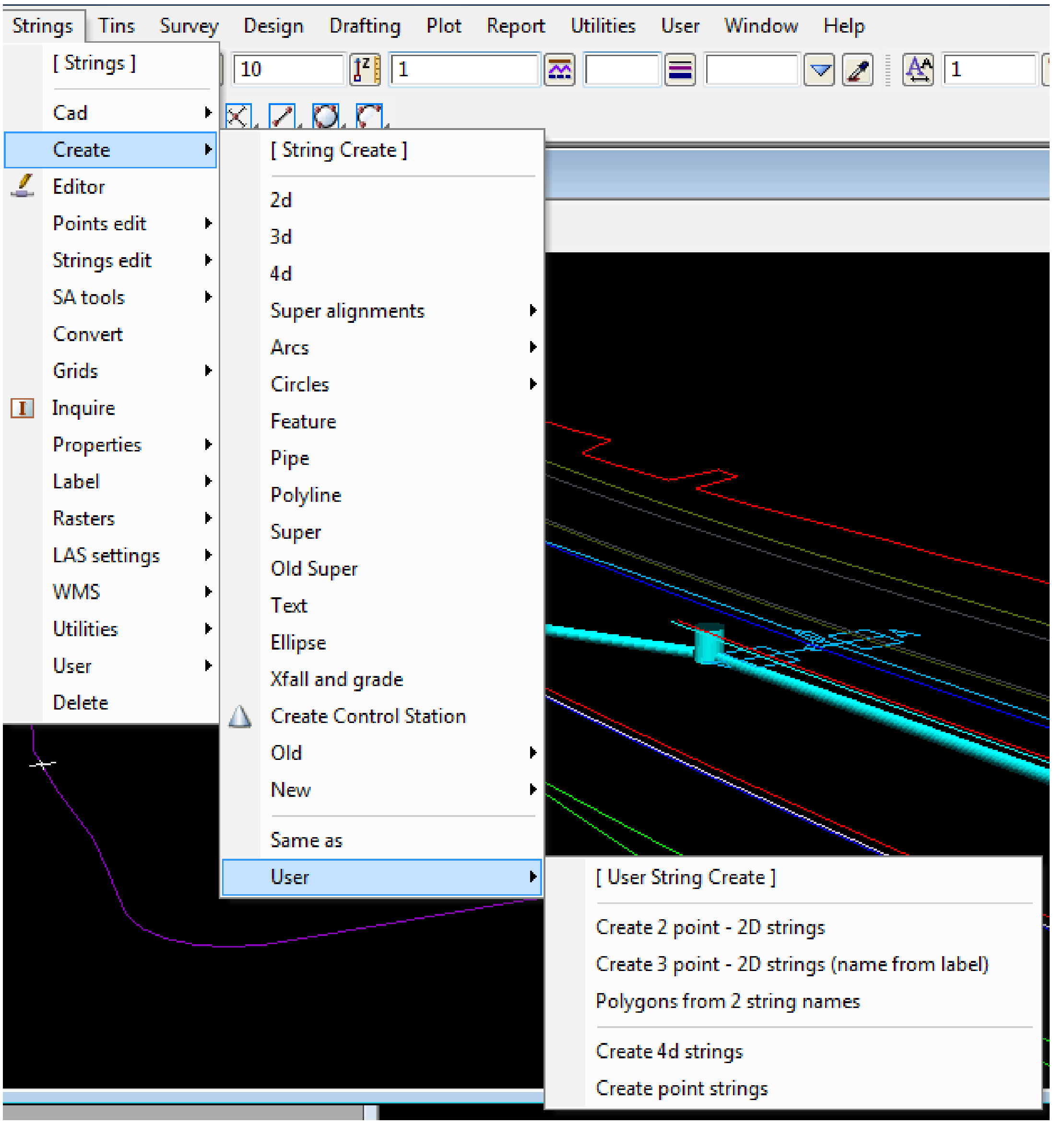
User Defined Menus on Other 12d Menus

It is also possible to define a ***User*** submenu on any ***12d Model*** menus or submenu.

In ***12d Model,*** all menus and sub-menus on the **Main** menu have a unique name. So a **User** menu is defined for any of them by simply having a menu called "User menu_name" in the *usermenu.4d* file.

For example, the menu for the option *Strings=>Create* is called **String Create** (String not Strings) so a ***User*** menu for *Strings=>Create* would be called "User String Create":

```
Menu "User String Create" {
   Button "Create 4d strings" {
      Command "macro  -close_on_exit  $USER_LIB/ref_points.4do"
   }
   Button "Create point strings" {
      Command "macro  -close_on_exit  $USER_LIB/x_sects.4do"
   }
}
```


Strings
Tins
Survey
Design
Drafting
Plot
Report
Utilities
User
Window
Help
[Strings]
Cad
Create
Editor
Points edit
Strings edit
SA tools
Convert
Grids
Inquire
Properties
Label
Rasters
LAS settings
WMS
Utilities
User
Delete
[String Create]
2d
3d
4d
Super alignments
Arcs
Circles
Feature
Pipe
Polyline
Super
Old Super
Text
Ellipse
Xfall and grade
Create Control Station
Old
New
Same as
User
[User String Create]
Create 2 point - 2D strings
Create 3 point - 2D strings (name from label)
Polygons from 2 string names
Create 4d strings
Create point strings

41.2.4 Names for User Defined Menus

Apart from the names of the special User menus just mentioned, all other menu names in *usermenu.4d* are user defined so they can be easily used in user defined walk-right menus.

And menu referred to in the file *usermenu.4d* that is not one of the special *User menus*, ***must be must be defined*** somewhere in the *usermenu.4d* file.

And remember, all the names must be unique amongst the ***12d Model*** menus, ***12d Model*** panels, special User menu and *user named User Defined* menus.

Continue to the next section 41.2.5 Example of a User Defined Menu or return to 41.2 User Defined Menus.

41.2.5 Example of a User Defined Menu

// the definition of buttons to go on the Main menu option "User"

```
Menu "User" {                           // User menu on Main menu
   Button "Triangulate" {               // button called Triangulate which brings up
      Command "Triangulate a View"      // the "Triangulate a View" panel
   }
   Button "Lots" {                      // button called Lots with its
      Walk_Right "Lots"                 // own walk-right menu called Lots
   }
   Button "Roads" {                                 // button with walk-right menu "Roads"
      Walk_Right "Roads"
      Command "layout $LIB/road.slf"                // button invokes a layout file if selected
   }
}

// the definition of the walk-right menu Lots

Menu "Lots" {
   Button "Create lots" {
      Command "macro $LIB/subdiv9.4do"
   }
   Button "Label lots" {
      Command "macro $LIB/subnum3.4do"
   }
   Button "Write Lot Levels" {
      Command "macro $LIB/Tinval.4do"
   }
}

// the definition of the walk-right menu Roads

Menu "Roads" {
   Button "Roundabout" {
      Command "macro $LIB/Round.4do"
   }
   Button "Culdesac bulb" {
      Command "macro $LIB/Culd.4do"
   }
   Button "Drape alignment string" {
      Command "macro $LIB/Drape_al.4do"
   }
   Button "Create Table drain" {
      Command "macro $LIB/Table_dr.4do"
   }
}
```

// the definition of User on the Main menu option Strings=>Create"

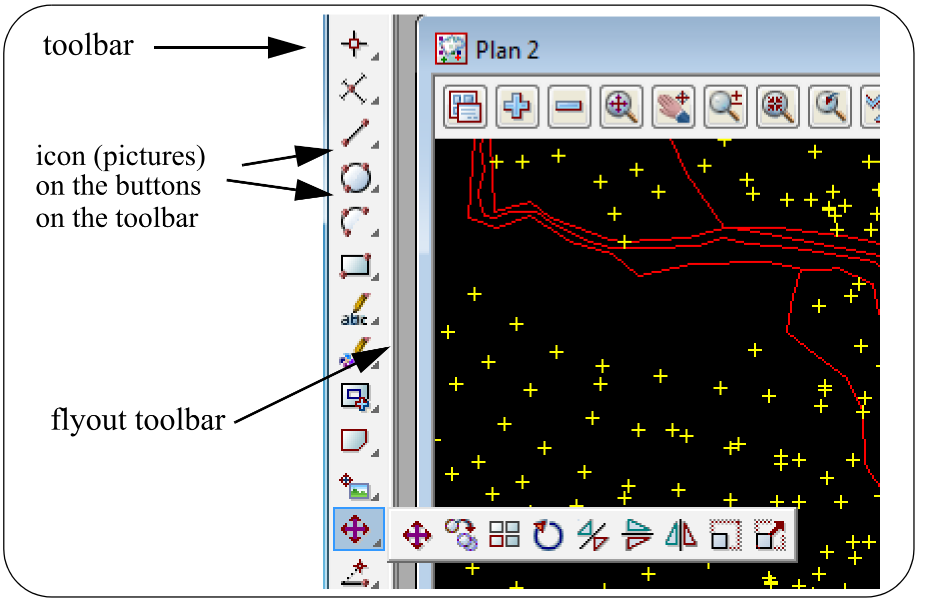
```
Menu "User String Create" {                  // User menu on option Strings=>Create
      Button "Create ref points" {
      Command "macro  -close_on_exit  $LIB/ref_points.4do"
   }
   Button "Create X-sections" {
      Command "macro  -close_on_exit  $LIB/x-Sects.4do"
   }
}
```

Continue to the next section 41.3 User Defined Toolbars or return to 41.2 User Defined Menus.

41.3 User Defined Toolbars

A ***toolbar*** is made up of buttons which can have a picture on them (icons) and when the left hand mouse button (LB) is pushed down and released on the button, a command is executed.

A button can also have a *flyout* toolbar which appears when the left hand mouse button (LB) is held down on the button (note that there can only be one level of flyouts).



Toolbars can be turned on and off, and their positions are remembered when ***12d Model*** is saved. See 4.3.5 Toolbars and Controlbars.

The definitions for *toolbars* in ***12d Model*** are stored in two files - *user_toolbars.4d* and *toolbars.4d*. When ***12d Model*** is installed, *toolbars.4d* is installed in the ***12d Model*** folder *Set_Ups*. *user_toolbars.4d* is for user defined toolbars and is created by the user and should be stored in the folder *User*.

So any user defined toolbars and simply added to *user_toolbars.4d* and saved in the folder *Users*.

The icon for each button on a toolbar, plus the action taken when the button is selected is user specified. A button can include a fly-out toolbar but flyouts can only be to one level (i.e. no flyouts on flyouts).

When a button is **selected**, the action can be to

(a) run a ***12d Model***12dPL (macro) or chain
(b) bring up a ***12d Model*** menu
(c) bring up a ***12d Model*** panel
(d) read a layout file
(e) execute a batch file or program (system call)

When ***12d Model*** starts up, any toolbars defined in *toolbars.4d* and *user_toolbars.4d* that are *not* called as flyouts by another toolbar in *toolbars.4d* or *user_toolbars.4d*, are displayed in the *main toolbar* on the left hand side of the ***12d Model*** screen.

When a toolbars is moved around, its position is saved when ***12d Model*** is saved so the toolbars are in the same final position when the project is reopened.

Go to the next section 41.3.1 Full Definition of Toolbars.

41.3.1 Full Definition of Toolbars

The definition for **Toolbars** is given in the files called

Toolbars.4d and **user_toolbars.4d**

which are searched for in the standard set up areas (local, USER_4D, user, set_ups etc.)

Or instead of *toolbars.4d*, it can be defined and set by the environment variable

TOOLBARS_4D file // Toolbars definition

In the *toolbars.4d* and *user_toolbars.4d* files, a ***toolbar*** is defined by:

```
Toolbar  toolbar_name  {
              button_1
              button_2
              ...
              button_n
}
```

where a button can include none or more the commands *Icon*, *Command* and *Flyout,* and the syntax for button is:

```
Button   button_name {
              Icon          icon_name_1
              Command       command_name_1
              Flyout        toolbar_name_1          // only used if a flyout toolbar is required
}
```

For the definitions of the *Icon*, *Command* and *Flyout* commands see:

41.3.1.1 Icons for Toolbars.
41.3.1.2 Command for Toolbars.
41.3.1.3 Flyout for Toolbars.

41.3.1.1 Icons for Toolbars

The ***Icon*** command defines the bitmap that is displayed for the button in the toolbar. The format of the bitmap is a bmp file (.bmp) of size 16x16. The *Icon* command consists of the word *Icon* followed by one or more spaces and then the name of the bitmap, *icon_name*.

Icon ***icon_name***

The *icon_name* can include a pathname but if no pathname is present, the bitmap is searched for in the folder *Images* under the folders *User* or *Set_Ups*.

If the *Icon* command is missing then a default *Icon* command is used with the *icon_name* ***button_name.bmp***

Icon *button_name.bmp*

Go to the next section 41.3.1.2 Command for Toolbars or return to 41.3.1 Full Definition of Toolbars

41.3.1.2 Command for Toolbars

The ***Command*** command defines what action occurs if LB is clicked on the button. The *Command* command consists of the word *Command* followed by one or more spaces and then the name of the command and any arguments it requires.

Command ***command_name***

The Commands are the same as those for a ***User Defined Menu***. See Commands for Buttons

If the *Command* command is missing then a default *Command* is used with the command name ***button_name*** with no arguments

Command *button_name*

Go to the next section 41.3.1.3 Flyout for Toolbars or return to 41.3.1 Full Definition of Toolbars

41.3.1.3 Flyout for Toolbars

The ***Flyout*** command defines what toolbar is displayed when LB is help down on the button. The *Flyout* command consists of the word *Flyout* followed by one or more spaces and then the name of the toolbar that is displayed.

Flyout ***flyout_toolbar_name***

where *flyout_toolbar_name* is the name of another toolbar defined elsewhere in the toolbars file.

When the *Flyout* command exists, *Icon* and *Command* are not required and if they are present, are ignored.

The icon on the *Flyout* button is the *Icon* from the first button in the flyout toolbar.

The command executed if LB is pressed and released on the Flyout button is the *Command* from the first button in the flyout toolbar.

NOTE - flyouts *can not* be nested so the toolbar *flyout_toolbar_name* can not contain a *Flyout* command.

NOTE - when ***12d Model*** starts up, any toolbars defined in *toolbars.4d* that are *not* called as flyouts by another toolbar in *toolbars.4d*, are displayed in the *main toolbar* on the left hand side of the ***12d Model*** screen.

Example of a User Defined Toolbar

```
// --------------------------------------------
// WARNING: you cannot have the same command using different icons
// ALSO: when using a flyout, the Command & Icon are ignored
//       and the details are taken from the first button
//       on the flyout
// --------------------------------------------
Toolbar "Cad" {
   Button "Points" {
      Command "Create Point"              // this is ignore since it is a flyout
      Icon    "Create Point.bmp"          // this is ignore since it is a flyout
     Flyout  "Cad Points"
  }
   Button "Change String" {
      Flyout  "Change String"
   }

Toolbar "Cad Points" {                    // used as a flyout toolbar
   Button "Create Point" {
      Command "Create Point"
      Icon    "Create Point.bmp"
   }
   Button "Chainage" {
      Command "Locate Chainage"
      Icon    "Chainage.bmp"
   }
}  // end of "Cad Points" toolbar
```

```
Toolbar "Change String" {                    // used as a flyout toolbar
   Button "String Close" {
      Command "String Close"
      Icon    "String Close.bmp"
   }
   Button "String Reverse" {
      Command "String Reverse"
      Icon    "String Reverse.bmp"
   }
   Button "String Trim" {
      Command "String Trim"
      Icon    "String Trim.bmp"
   }
}  // end of "Change String" toolbar
```

$LIB and $USER_LIB

As they were for Commands in User Defined menus, the variables $LIB and $USER_LIB can be used in Commands as part of the user_macro_name, chain_file_name, layout_file_name and programe_or_batch_file_name to pick up files from either the *12d Library* or the *User* library. See 41.2.2 Using $LIB and $USER_LIB in User Menus.

For example, the definition of a toolbar called "Lots" could be:

```
Toolbar   "Lots" {
   Button "Create lots" {                              // button called "Create lots" which
      Command "macro -close_on_exit   $USER_LIB/subdiv.4do"   // fires up a 12d Model macro
   }
   Button "Roads" {                    // button with walk-right menu "Roads" and layout file
      Walk_Right "Roads"                               // walk-right menu "Roads"
      Command "layout $USER_LIB/road.slf"              // button invokes a layout file if selected
   }
}
```

Notes

(a) The variables $LIB and $USER_LIB can be used as part of the layout_file_name and user_macro_name to pick up files from either of the libraries.

(b) blank lines in the toolbars file are ignored and anything on a line after a // is a comment

Another Example of a Toolbar

```
Toolbar "Cogo" {
// in this case, the Command is assumed to be "Create Line"
// the icon is "Create Line.bmp"
// Command is normally the name of the panel/menu
  Button "Create Line" {
 }
// this is a spacer
 Button "" {
 }
// this is a command and also has a flyout capability
```

```
// calling the toolbar called Flyout
// flyouts cannot be nested
  Button "Create Fillet by radius" {
    Command "Create Fillet by radius"
    Icon    "Create Fillet by radius.bmp"
    Flyout  "Flyout"
  }
}
// The flyout toolbar for above (and a normal toolbar as well)
Toolbar "Flyout" {
  Button "Create Line" {
  }
  Button "" {
  }
  Button "Create Arc by Centre Radius" {
    Command "Create Arc by Centre Radius End Points"
    Icon    "Create Arc by Centre Radius End Points.bmp"
  }
  Button "Create Fillet by radius" {
    Command "Create Fillet by radius"
    Icon    "Create Fillet by radius.bmp"
  }
}
```

Return to 41.3.1 Full Definition of Toolbars or 41 Functions Keys, Menus, Toolbars.

42 Special File Formats

42.1 Default File Ending

In any panel pop-up requiring a file name to be displayed, default file endings are used to restrict the **names** of the files selected from the current folder and if they exist, from the library and user library areas.

Whilst a panel is up, the extension being searched for can be changed by typing say "*.xyz" into the panel field requiring a file name, and then pressing <enter>. The list of files ending in ***.xyz*** will be displayed and clicking B3 in the panel field will also bring up a list of all files ending in **".xyz"**. When a new panel is created, the panel fields revert to the default file ending.

A list of the **default** files used in 12d Model pop-ups and their endings is:

File Type	**Default Ending**	
Miscellaneous Files		
affine 2d	.aaf	see 28.9.3 2D Affine
affine 2d orthogonal	.ortho_aff	see 28.9.4 2D Affine - Orthogonal
chains (V10 onwards)	.chain	see 28.3 Chains
chains (pre V10)	.rcn	see 28.3 Chains
boxing definitions file	.bf	
digitizer registration	.aff	see 8.5.2.2 Register Plan
eagle mapping	.emf	
Helmert 2d - no forced scale	.hel	No longer used
Helmert 2d - allow forced scale	.hel_adv	see 28.10.4.1 2D Helmert (Advanced)
Helmert 3d	.hel_3d	see 28.10.4.2 3D Helmert
label map file (pre V10	.lmf	
label map file (V10 onwards	.label_mapfile	
many templates	.mtf	see 21 Advanced Design
Map file (pre V10	.mf	see 8.8.1 Create/Edit a Map File
Map File (V10 onwards	.mapfile	see 8.8.1 Create/Edit a Map File
panel defaults file (pre V10)	.ddf	see 4.3.6.6 Panel Defaults - ddx Files
panel defaults file(V10 onwards	.ddx	see 4.3.6.6 Panel Defaults - ddx Files
reports	.rpt	
screen layout file (pre V10)	.slf	
screen layout file(V10 onwards	.slx	
templates	.tpl	
textstyle file	.tsf	
Survey Files		
12d Model field file	.fld	
reduction history	.rh	
12d field - Helmert 2.5	.tdf_hel	see 18.2.2 GPS Localisation
Input/Output Files		
4d ascii	.4da	
12da	.12da	
AutoCAD	.dxf	
BCC Epson dat files	.dat	
BCC Epson sur files	.sur	
CivilCad V4	.asc	

CivilCad V5	.as5
Eagle command	.cmd
Eagle binary	.mod
Geocomp	.pts
Keays	.trf
Microstation, Intergraph binary	.dgn
MX, Moss	.mos
TP Setout	.pta
xyzs data	.dat

Range Files

aspect range	.arf
depth range	.drf
height range	.hrf
slope range	.srf

Plot Parameter Files

Drainage long section	.drainppf	see 26.5 Drainage Long Plot PPF Editor
Drainage plan	.drainplanppf	see 26.8 Drainage Plan Plot PPF Editor
Long section	.lplotppf	see 26.4 Long Plot PPF Editor
Melbourne Water) sewer long section	.melbppf	see 26.6 Melbourne Water Plot PPF Editor
Plot frame	.plotframeppf	see 26.9 Plot Frame and PPF Editor
Pipeline long section	.pipelineppf	see 26.7 Pipeline Plot PPF Editor
Xsection	.xplotppf	see 26.3 Section X Plot PPF Editor
Pre V7 plot parameter files	.ppf	

Plot Format Files

AutoCAD	.dxf
CalComp	.cal
DGN (Microstation,Intergraph)	.dgn
Dogs	.par
Eagle	.mod
Frame maker	.mif
HP GL	.hp
HP 7475	.hpa
HP GL 7600	.hpm
HP GL 2 (colour)	.hpc
pcl5	.pcl5
PostScript	.ps

Display Files - dumps, movies

gif	.gif
jpeg	.jpg
jpeg 2000	.j2k
PDF	.pdf
PNG	.png
PostScript	.ps
Targa	.tga
Tiff	.tif
Windows bitmap	.bmp
XPS	.xps

4DML's - 12d Solutions programming/macro language

macro source file	.4dm

macro object/executable	.4do
macro listing	.4dl

Note

This list is for the default files used in 12d Model panel fields. It does not include 12d Model setup files which are given in the appendix 39 Setting Up & Configuring 12d.

42.2 Special 12d Solutions File Formats

12d Solutions has a number of special file formats, most of which have already been specified. The remaining formats will now be given.

In any of the files, blank lines ignored and any information after // until the end of that line is ignored. Hence **comments** can easily be inserted into the files by preceding them by ***//***.

Unless enclosed in quotes ("), more than one consecutive space or tab are treated as one space.

42.2.1 Eagle Map File

An eagle map file is a user created file consisting of a list of 12d Model colours and the Eagle pen, dash style, pen thickness and frag to be used for the colour.

The map file is set out with one 12d Model colour per line. The line begins with the 12d Model colour followed by the Eagle pen number, dash style, thickness and frag to be used for the 12d Model colour. Each item is separated by one or more spaces.

For example, if the 12d Model colour **red** is to be mapped to Eagle pen 3, dash style 4, thickness 2 and frag 1, then the line in the eagle map file would be

```
red       3        4        2        1
```

An eagle map file can be used for writing out three dimensional data to Eagle, or when producing plots in Eagle format.

For plotting, the map file must be called **eagleplt.emf**.

When writing three dimensional data out to Eagle, a user specified file name is allowed.

In the output case, as each string is written out, the map file is searched sequentially until a colour match is made. If no match is found, the colour of the string is used as a pen number and default values used for dash, thickness and frag.

Comments can be included in the map file by preceding the comment with a double forward slash (i.e. //). Anything on the line following the // will be ignored.

An example of an Eagle map file is,

```
//12d Model colour to Eagle mapping file
//
//12d Model colour  eagle pen           dash  thickness   frag

black               0                    1        1         1
red                 2                    0        1         5
green               4                    1        1         1
blue                4                    2        1        11
cyan                5                    1        1         1
yellow              6                    1        1         1
magenta             7                    1        1         1
white               1                    1        1         1
orange              15                   1        1         1
purple              14                   1        1         1
grey                13                   1        1         1
"dark green"        12                   1        1         1
"dark red"          11                   1        1         1
"off yellow"        4                    1        1         1
"dark blue"         8                    1        1         1
brown               11                   1        1         1
```

42.2.2 Template File

The 12d Solutions Template file format is a simple text file definition for reading and writing out template definitions for use in the apply options in 12d Model.

The template definition begins with the key word **template** followed by the template name and then the definitions of

fixed, cut, fill and final parts of the template

or fixed and decisional parts

all enclosed within curly braces **{ }**.

```
template fred {                                  template fred {

  fixed {                                                  fixed {
    ...                       or                             ...
  }                                                        }

  cut {                                                    decisional {
    ...                                                      ...
                                                           }
  fill {
    ...
  }
  final {
    ...
  }
}
```

The definition of the **fixed** part of the template begins with the key word **fixed** followed by a list of the links enclosed in curly braces {}.

The links are defined one per line in order from the centre-line. Each link begins with the keyword **link** followed by the width, percent cross-fall, colour and name for each link. For the percent cross-fall, positive means up, negative is down and zero horizontal.

For example,

```
fixed {
  link    3.5       -3        cyan      kerb
  link    1         -4        magenta   shoulder
  link    2         -5        blue      verge
}
```

describes a fixed template with three links, the first of width 3.5 with a 3% cross-fall downwards, the next link of width 1 with 4% cross-fall downwards and the third link of width 2 and 5% cross-fall downwards.

The definition of the **cut** part of the template begins with the key word **cut** followed by a list of the links enclosed in curly braces {}.

The links are defined one per line in order from the end of the fixed template (or centre-line if no fixed part exists). Each link begins with the keyword **link** followed by the width, one in slope, colour and name for each link. For cut, the **one in** slope is positive for up, negative for down, and zero for horizontal.

For example,

```
cut {
  link    3        2        yellow    a
  link    2        0        magenta   b
  link    3        -1       yellow    c
  link    2        0        magenta   d
}
```

describes a cut template with four links, the first of width 3 with 1:2 slope upwards, the next link of width 2 horizontally, the third link of width 3 and 1:1 slope downwards and width 2 horizontally.

The definition of the **fill** part of the template begins with the key word **fill** followed by a list of the links enclosed in curly braces {}.

The links are defined one per line in order from the end of the fixed template (or centre-line if no fixed part exists). Each link begins with the keyword **link** followed by the width, one in slope, colour and name for each link. For fill, the **one in** slope is positive for down, negative for up, and zero for horizontal. Please note that this is the opposite to **cut**.

For example,

```
fill {
  link    5        2        yellow    e
  link    1        0        magenta   f
  link    10       -1       yellow    g
}
```

describes a fill template with three links, the first of width 5 with 1:2 slope downwards, the next link of width 1 horizontally, and the third link of width 10 and 1:1 slope upwards.

The definition of the **final** part of the template begins with the key word **final** followed the cut slope, fill slope and search distance enclosed in curly braces {}. The key words for the three values are cut_slope, fill_slope and search_distance.

For example,

```
final {
  cut_slope  1  fill_slope  2  search_distance  100
}
```

describes a final cut slope of 1:1 (upwards), fill slope of 1:2 downwards and template with three links, the first of width 5 with 1:2 slope downwards, both going for a maximum distance of 100.

The definition of the **decisions** part of the template begins with the key word **decisional** followed by a list of the decision commands enclosed in curly braces { }.

The decision commands are defined one per line and have the format:

Fixed Xfall Width *value* Height *value* XFall *value* Name *text* Colour *colour*

Fixed Slope Width *value* Height *value* Slope *value* Name *text* Colour *colour*

Tin Width *tin_name* Strip *value* width *value* Name *text* Colour *colour*

String Offset *string_name* Strip *value* Offset *value* Name *text* Colour *colour*

Batter *tin_name* Strip *value* Width *value* Height *value* Slope *value* Name *text* Colour *colour* Goto *label*

Tin Decision *tin_name* Offset *value* Min *value* Max *value* Goto *label*

Label *label_name*

Goto *label_name*

End

For example,

```
template "std" {
  fixed {
    link 3 unknown -3 cyan "kerb"
    link 1 unknown -4 magenta "shoulder"
    link 2 unknown -5 purple "verge"
  }
  decisional {
    Tin_Decision   "rock" 0 0 1000 "cut_rock"
    Tin_Decision   "shale" 0 0 1000 "cut_shale"
    Tin_Decision   "terrain" 0 0 1000 "cut_terrain"
    Label          "fill_terrain"
    Batter         "terrain" 0 8 unknown -3 "f1" blue "alldone"
    Batter     "terrain" 0 1 unknown 0 "f2" "dark green" "alldone"
    Goto           "fill_terrain"
    Label          "cut_rock"
    Tin_Decision   "rock" 0 0 0.3 "cut_rock_done"
    Batter         "rock" 0.3 5 unknown 0.5 "r1" cyan "cut_rock_done"
    Batter         "rock" 0 2.5 unknown 0 "r2" yellow "cut_rock_done"
    Goto           "cut_rock"
    Label          "cut_rock_done"
    Tin_Decision   "shale" 0 0 1000 "cut_shale"
    Goto           "cut_shale_done"
    Label          "cut_shale"
    Tin_Decision   "shale" 0 0 0.6 "cut_shale_done"
    Batter     "shale" 0.6 6 unknown 1 "s1" magenta "cut_shale_done"
    Batter     "shale" 0 3 unknown 0 "s2" "dark red" "cut_shale_done"
    Goto           "cut_shale"
    Label          "cut_shale_done"
    Tin_Decision   "terrain" 0 0 1000 "cut_terrain"
    Goto           "alldone"
    Label          "cut_terrain"
    Batter         "terrain" 0 3 unknown 1.5 "t1" red "alldone"
    Batter         "terrain" 0 2 unknown 0 "t2" green "alldone"
    Goto           "cut_terrain"
    Label          "alldone"
  }
}
```

Notes

1. spaces in text - any text string that includes spaces or only numbers, must be enclosed in double quotes **"**.
2. comments - anything after // until the end of the line is ignored.
3. blank lines - blank lines are ignored

42.2.3 Screen Layout File

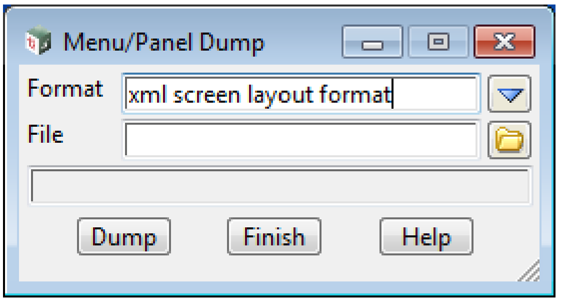
The 12d Model **screen layout file** contains a simple text file definition for 12d Model panels and menus in either the pre-V10 format (slf) or the V10 XML format (slx).

The definition includes a screen position for the panel or menu, and for panels, values for any of the panel fields.

At any time, the layout of the menus and panels on the screen in a 12d Model session can be written out using the menu option

File i/o=>Layouts =>Layout output

Also, the layout for an individual panel or menu can be created by clicking RB in the menu/view title area and selecting the dump option to bring up the **Menu/Panel Dump** panel.



Note: If a screen layout file is created and added to the *layout.4d* file (by selecting *layout.4d* as the file name and selecting **Append**), then the menu/panel will appear whenever a project is opened. See layout.4d

A screen layout file can be read in using the menu options

File i/o=>Layouts =>Layout input or **File i/o=>Layouts =>Layout input files**

Also when a project starts up, the layout file ***layout.4d*** is read in and any menus and panels described in the file are placed on the screen (see 39.2.3 Setup Files Used for New and Existing Projects).

An example of a screen layout file for 'Read x y z s Data' in the V10 format is:

```
<?xml version="1.0"?>
<xml12d xmlns="http://www.12d.com/schema/xml12d-10.0" xmlns:xsi="http://www.w3.org/
2001/XMLSchema-instance" language="English" version="1.0" date="2013-04-03"
time="12:22:55" xsi:schemaLocation="http://www.12d.com/schema/xml12d-10.0 http://
www.12d.com/schema/xml12d-10.0/xml12d.xsd">
 <meta_data>
  <units>
   <metric>
    <linear>metre</linear>
    <area>square metre</area>
    <volume>cubic metre</volume>
    <temperature>celsius</temperature>
    <pressure>millibars</pressure>
    <angular>decimal degrees</angular>
    <direction>decimal degrees</direction>
   </metric>
  </units>
  <application>
   <name>12d Model</name>
   <manufacturer>12d Solutions Pty Ltd</manufacturer>
   <manufacturer_url>www.12d.com</manufacturer_url>
   <application>12d Model 10.0C1i Doco - Not For Production</application>
```

```
      <application_build>10.1.9.7</application_build>
      <application_path>C:\Program Files (x86)\12d\12dmodel\10.00\nt.x86\12d.exe</
application_path>
      <application_date_gmt>11-Mar-2013 01:48:06</application_date_gmt>
      <application_date>11-Mar-2013 12:48:06</application_date>
    </application>
  </meta_data>
  <screen_layout>
    <version>1.0</version>
    <panel>
      <name>Read x y z s Data</name>
      <x>2393</x>
      <y>215</y>
      <files_box>
        <name>File</name>
        <tick_box>
          <name>Advanced</name>
          <value>false</value>
        </tick_box>
        <file_box>
          <name>File to read</name>
          <value>faces.dat</value>
        </file_box>
      </files_box>
      <file_box>
        <name>Map file</name>
        <value/>
      </file_box>
      <input_box>
        <name>Pre*postfix for models</name>
        <value/>
      </input_box>
      <input_box>
        <name>Default line colour</name>
        <value>magenta</value>
      </input_box>
      <input_box>
        <name>Default point colour</name>
        <value>yellow</value>
      </input_box>
      <input_box>
        <name>Default model for data</name>
        <value>faces</value>
      </input_box>
      <tick_box>
        <name>Use super strings</name>
        <value>false</value>
      </tick_box>
      <input_box>
        <name>Add to view</name>
        <value>1</value>
      </input_box>
      <run_button>
        <name>&amp;Read</name>
      </run_button>
    </panel>
  </screen_layout>
</xml12d>
```

42.2.4 Map File for 12d Model V4.0

Up to ***12d Model*** V4.0, the map file format consisted of one or more lines. Each line begins with a key (entity-mask) followed by a string name, model name, colour, breakline type (point or line), and line style separated by one or more spaces. The key can contain wild cards (*) and wild characters (?).

When a string is read in and satisfies a key, the key's corresponding string name, model, colour, breakline type and style is used for that string.

For example, any entity name beginning with **31** can be created as a 12d Model string with the name **picket**, colour **cyan**, breakline type **line**, line style **1** and model **fences** by the map file line

```
31*      picket   fences   cyan     line     1
```

If a map file is used, as each entity is read in, the map file is searched **sequentially** until a match with a key is made and the key's name, colour etc. used. If no match is found, the default colours and model (given in the read panel) are used.

Notes

1. If the entity-name is to be used as the new string name, use an asterisk (*) in place of the string name. For example,

```
fred     *        fences   cyan     line     1
```

2. If the default model for the reader is to be used as the model name, use an asterisk in place of the model name. For example,

```
31       31       *        cyan     line     1
```

3. A * for colour, breakline type and linestyle means that if the entity has a colour, breakline style or linestyle, then it is used rather than be mapped to another one.
4. All model names used in the map file can be given an extra (common) prefix by typing the prefix into the prefix for models field in the read panel.
5. If any information includes a space, then it must be enclosed in quotes **"**. For example, the model name may be "trial 1" or a style "large tanks".
6. Comments can be included in the map file by preceding them with a double forward slash //. Anything on the line after the // is ignored.

An example of a 12d Model map file is

```
// 12d Model map file
// key    name     model    colour   pt-line  linestyle

102      break    breaks   red      line     solid
305      fence    caddast  green    point    dash
998      bdry     bound    cyan     line     solid
spots    *        spot     yellow   point    1
PS*      *        *        yellow   point    1
```

Map File for 12d Model V5.0 and Above

For ***12d Model*** V5.0, the map file was extended to allow for defining properties such as tinability, symbols at vertices, vertex and segment text and pipe and culverts.

To allow for all the different type of mapping in the one file, the map file is broken up into sections (one section for each tab of the map file editor) and each section begins with a header record which is the section name enclosed in braces ({ }). For example, the symbols section has the header

{symbol_data}

Inside each section, the data is similar to the V4.0 map file with lines consisting of a key (entity-mask) followed by the data required for that section. The key can contain wild cards (*) and wild characters (?).

The exact format for the map file is not required since they are created and edited from within 12d Model by the option

File i/o => Map file

42.2.5 12d Patterns File

See 40.4 Patterns

43 Plotters and Plotting

The method of creating plots and sending them to a plotter, or to a pdf, or a Cad system such as AutoCad or Microstation, varies from site to site.

So ***12d Model*** provides tools to enable the user to tailor the plotting system, especially the defining of Plotters and Plotter Mapping Files which are described in this chapter.

See

43.1 12d Model Plotters

43.2 Defining Plotters - Plotters.4d

43.3 Mapping 12d Colours to Pens and RGBs

43.4 Sending Plots to a Plotter

43.5 Sheet Sizes

43.6 Text Units in Plots

43.1 12d Model Plotters

When creating a **plot**, ***12d Model*** has a variety of options about where the potting data goes to.

The user can:

(a) drive the plotters **directly** using Windows printer drivers, raster plotter drivers, hp plotter commands etc.

(b) plot **indirectly** by creating a computer disk file (the plot file) in a variety of formats containing the relevant plotter instructions for producing the plot. The formats include pdf, XPS, DWG and DGN.

(c) plot **indirectly** to a ***12d Model*** **model**

Once the plotting device (**plotter engine)** the decided, there will still be a variety of ways that the ***12d Model*** colours are to appear on the plot.

For example, for some plots you may want all the colours to be plotted as black and shades of grey rather than actual colours. Or for white to be plotted as black.

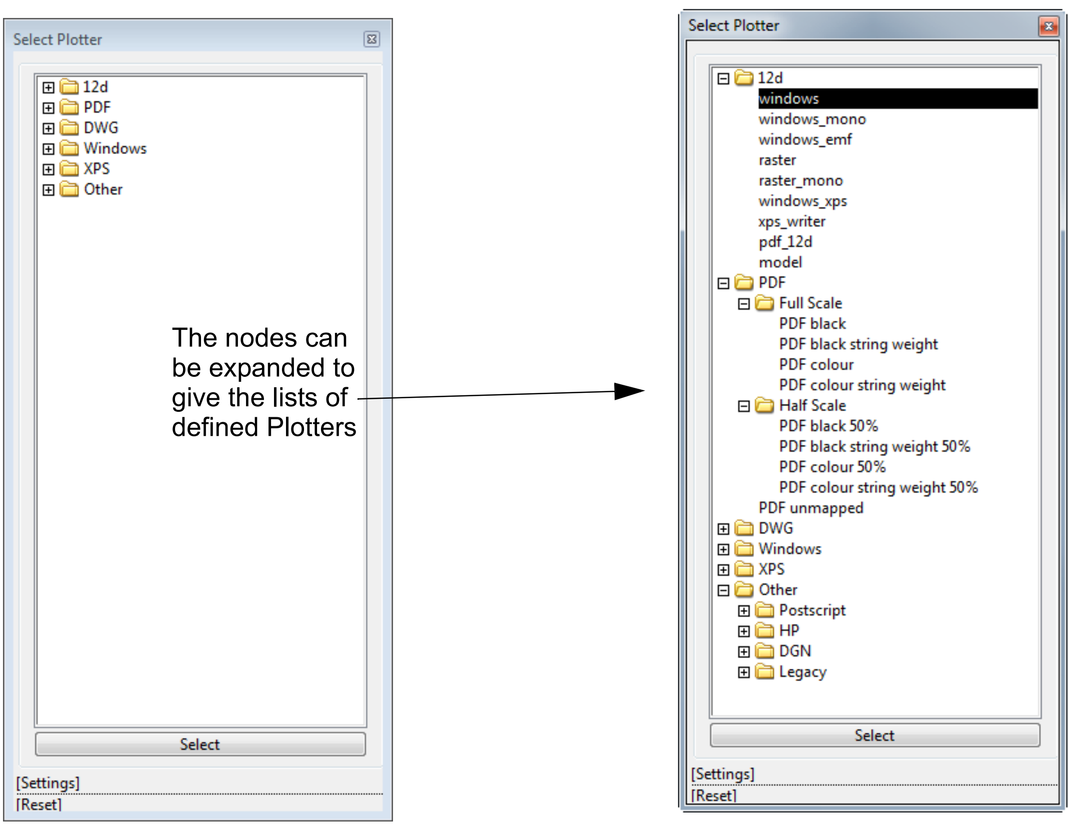
To control **how** a colour is plotted on a particular plot device, there are various user definable parameters and the most flexible for controlling colours is the **plotter mapping file** (pmf file). There are a number of pmf files installed with ***12d Model*** and/or users can also define their own.

So an ***12d Model plotter*** consists of a

1. Unique name for the Plotter
2. The Plot Engine (plot device)
3. Other Parameters to control the plot engine and outputs

The definitions for each of the ***12d Model plotters*** is given in the file **plotters.4d** and it is the list of plotter names from this file that is displayed as the choice list in the when the **Select Plotter** pop-up menu.

The list of ***12d Model*** plotters installed with ***12d Model*** is:



From the list you will see that there can be a number of Plotters using the same plot device (e.g. windows or pdf) and these will differ in one or more of the other plot parameters. For example, having a different plotter mapping file.

When a **plot file** is created, it can be passed to a user specified program or batch file which can do things like automatically directing the plot file to a plotter. See 43.3 Mapping 12d Colours to Pens and RGBs.

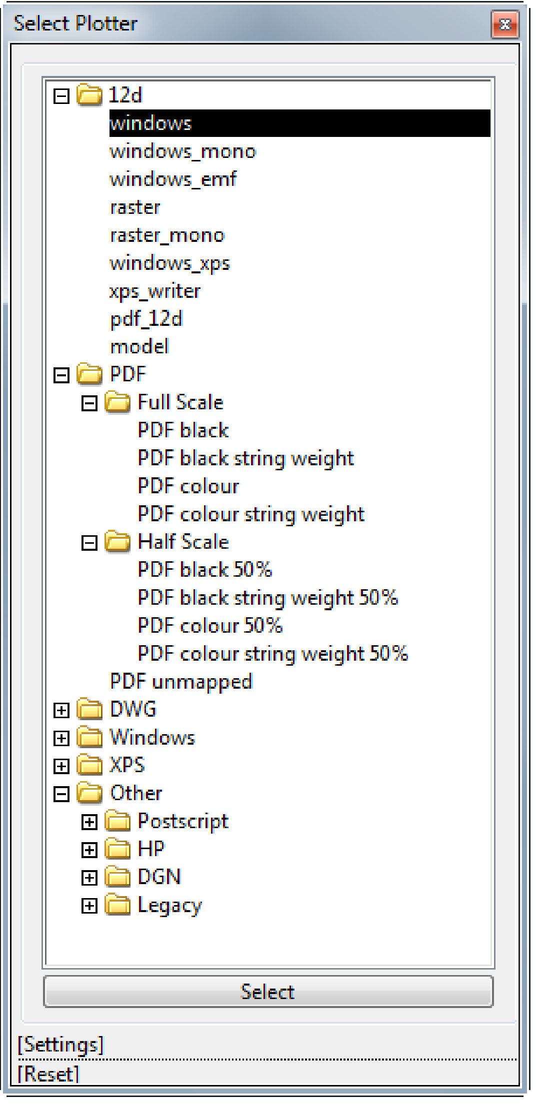
For information on plotters.4d, see 43.2 Defining Plotters - Plotters.4d

43.2 Defining Plotters - Plotters.4d

The file ***plotters.4d*** defines the names of the plotters that appear in the **plotter type** panel field, and the properties of each of these named plotters.

On opening a ***12d Model*** project, if the environment variable ***PLOTTERS_4D*** exists then it has the pathname to the file that defines the plotters otherwise, the file **plotters.4d** is searched for in the *standard set up paths*. See PLOTTER_4D and 39.2.4 Folders Searched for Setup Files).

The pop-up list for the ***plotters.4d*** file shipped with ***12d Model*** is:



but users can delete plotters from this and/or add their own plotters to the file.

For each plotter in *plotters.4d*, it is possible to customize:

(a) the name of the plotter

(b) the plot device (plot engine)

(c) any plotter mapping for mapping colours

(d) a start and end sequence that is sent to a physical plotter

In the plotter file, the definition of a **plotter** of a given name ***plotter_name*** is set out as:

plotter *plotter_name* {
set_up_command_1
set_up_command_2
. . .
set_up_command_n
}

plotter_name must not be blank, and if it includes imbedded spaces then the name must be enclosed in double quotes **"**.

Each *plotter_name* in the file must be unique in the file.

There can be zero or more *set up commands* inside the braces { } and each *set up command* is from the list:

group
engine
extension
map_file
map_pens
output_cmd
colour
header
footer

These *set up commands* are document in the next section 43.2.1 Plotter Set Up Commands.

For a description of some of the Plotters in the default plotters.4d file, see 43.2.2 Some Plotters in the Default Plotters.4d File.

For an example of a ***plotters.4d*** file, see 43.2.3 Example of a Plotters.4d File.

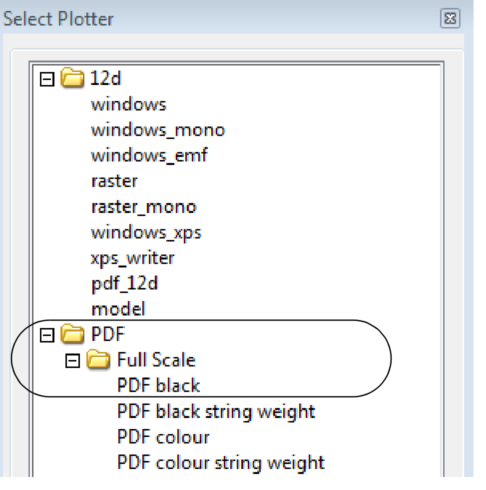
43.2.1 Plotter Set Up Commands

group

although all the *plotter names* have to be different, rather than just listing them in a flat list, the plotter names can be displayed in a tree structure.

The ***group*** command gives the levels in the tree for the plotter name in the pop up list. Each level is separated by a forward slash **/**

For example, if the plotter name was **PDF black** and *group* **PDF/Full Scale** then it will show up in the pop-up list as:



engine

the device that the plot is being created for. For example, model, pdf_12d, xps_writer, dwg_2010, dgn, dgn_v8, postscript and hpgl.

For a complete list of engines, see 43.2.1.1 Plotter Engines.

extension ".xxx"

the extension added to the file name used for the plot file written out by this plotter.

map_file *plotter_mapping_file_name*

if ***map_file exists***, it exists then it points to the *plotter mapping file* which is used to define each ***12d Model* colour** what **pen number** it is mapped to, and for each **pen number**, its **rgb** and **weight** (width).

If ***map_file* does not exist**, and a global mapping file is specified, then it will be used as the plotter mapping file. Otherwise the pen mapping in *Colours.4d* is used.

If the **engine** is any of the **dwg/dxf/dbx** variants, then the **map_file** is an **Autocad mapping file** with the extension of **.amf**, the plotter will use the **colour number** being plotted as the **key** in the Autocad mapping file. This allows the mapping from colour number directly into layer name, Autocad colour and linestyle. Note that if Autocad colour or linetype is BYLAYER, then ***12d Model*** uses the correct values in Autocad. The * character means that the field is ignored. For an example of an amf file, see 43.2.1.3 Definition and Example of a .amf File.

If the **engine** is **dgn/dgn_v8** then the **map_file** is a ***table file*** with the extension of ***.tbl***. The plotter will use the **colour number** being plotted as the **key** in the **table file**. This allows the mapping from the colour number directly into dgn level, weight, style and colour. For an example of a tbl file, see 43.3 Mapping 12d Colours to Pens and RGBs.

map_pens true or false

if ***map_pens*** is ***true***, i***t uses the specified map_file.***

If ***map_pens*** is ***false***, it doesn't use any mappings (map_file, global plotter mapping file or pen mapping table). It leaves the pen number untouched.

map_pens is **ignored** when

(a) the *map_file* is used as a ***tbl file*** (plotter engine is dgn and the map_file has an extension of .dgn. In this case any mapping is occurring via the tbl file where colour is the key.

(b) the map_file is used as a ***amf file*** (plotter engine is dxf or dwg and the map_file has an extension of .dxf. In this case, any mapping is occurring via the .dxf file where colour is the key.

output_cmd path to a *script* or *program*

for this plotter, the given script or program is used on the created plot file. This **replaces** the PLOTTER_4D definition of script or program for this plotter.

colour true or false

This is only used in hpgl2 and postscript.
If **false**, then
for postscript don't send the rgb for colours
for hpgl2, don't send down the block of information with the number of pens and the rgb for each pen.

That is, if false, it stops hpgl2 and postscript writing out the colour definition commands.

header and **footer**

The **header** information in placed in the plot file before any plot commands and the **footer** information is placed at the end of plot file after all the plot commands.

The format for the for the **header** of **footer** is:

header {// this information in placed in the plot file before any plot commands

lines of *text which can include plotter variables*

}

or

footer {// this information is placed at the end of the plot file, after the plot commands

lines of *text which can include plotter variables*

}

Each line of text is surrounded by quotes. For example, "this is some stuff".

The *plotter_variables* used in the **header** or **footer** are:

username

filename

date

time

sheet_width

sheet_height

page_number - not used

minimum_x, minimum_y// plotter dependent

maximum_x, maximum_y// plotter dependent

and are included in the text by preceding them by a + and also following them by a cross if more text follows. For example:

"the user is " + username+"of 4D Solutions"

The appropriate information is substituted for the *plotter_variables* when a plot is created.

Hence

"the user is "+ username+" of 12d Solutions"

would give

"the user is fred of 12d Solutions"

A plotter_variable may not be appropriate for a particular plotter and if a plotter_variable is specified but not used for a plotter, it is substituted by *blank*.

If **header** or **footer** is not specified, then appropriate default information for the plotter is used.

The **header** and **footer** set_up_commands are not used by all plotters. If they are not used for a particular plotter, the **header** or **footer** command is ignored.

At present, only the plotters hpgl, hpgl2, postscript and dxf use **header** and **footer**.

minimum_x etcfor postscript, it is the extent of the plot in mm
sheet_width, sheet_heightfor postscript in mm

For hpgl2

sheet_width, sheet_height in mm/40.

Continue to the next section 43.2.1.1 Plotter Engines or return to 43.2 Defining Plotters - Plotters.4d.

43.2.1.1 Plotter Engines

When creating a **plot**, 12d Model can either use a Windows printer driver to plot directly, or instead of driving the plotter directly, create a computer disk file (the plot file) containing the relevant plotter instructions for producing the plot (which is then sent to the plotter), or in the case of the plotter type **model**, create a 12d Model model.

The available Plotter Engines are:

Engine Name	Action
model	creates a ***12d Model*** model
pdf_12d	creates a pdf file
xps_writer	creates a xps file
windows	use Windows colour printer, or grey scale on mono printers
windows_xps	create a Windows xps file
raster	create a raster file
hp	creates HPGL commands with all of one pen being plotted before the next pen is used
hp_7475	small HP plotter, common hp emulation mode for lasers
hpgl_2	HPGL 2
dwg	creates a binary 2d DWG file
dwg_4d	creates a binary 2d DWG file
dwg_12	creates a binary 2d DWG V12 file
dwg_13	creates a binary 2d DWG V13 file
dwg_14	creates a binary 2d DWG V14 file
dwg_2000	creates a binary 2d DWG 2000 file
dwb_2002	creates a binary 2d DWG 2002 file
dwg_2004	creates a binary 2d DWG 2004 file
dwg_2006	creates a binary 2d DWG 2006 file
dwg_2007	creates a binary 2d DWG 2007 file
dwg_2010	creates a binary 2d DWG 2010 file
dwg_2010	creates a binary 2d DWG 2010 file
dwg_2013	creates a binary 2d DWG 2013 file
dxf	creates a 2d DXF file
dxf_4d	creates a 2d DXF file
dxf_12	creates a 2d DXF V12 file
dxf_13	creates a 2d DXF V13 file
dxf_14	creates a 2d DXF V14 file
dxf_2000	creates a 2d DXF 2000 file
dxf_2002	creates a 2d DXF 2002 file
dxf_2004	creates a 2d DXF 2004 file
dxf_2006	creates a 2d DXF 2006 file
dxf_2007	creates a 2d DXF 2007 file
dxf_2010	creates a 2d DXF 2010 file
dxf_2010	creates a 2d DXF 2010 file
dxf_2013	creates a 2d DXF 2013 file
dxb	creates a binary 2d DXB file
dxb_4d	creates a binary 2d DXB file
dxb_12	creates a binary 2d DXB V12 file

dxb_13	creates a binary 2d DXB V13 file
dxb_14	creates a binary 2d DXB V14 file
dxb_2000	creates a binary 2d DXB 2000 file
dxb_2002	creates a binary 2d DXB 2002 file
dxb_2004	creates a binary 2d DXB 2004 file
dxb_2006	creates a binary 2d DXB 2006 file
dxb_2007	creates a binary 2d DXB 2007 file
dxb_2010	creates a binary 2d DXB 2010 file
dxb_2010	creates a binary 2d DXB 2010 file
dxb_2013	creates a binary 2d DXB 2013 file
dgn	creates a Microstation binary model (**.dgn** file)
dgn_v8	creates a Microstation V8 binary model (**.dgn** file)
pcl5	creates a pcl5 file
eagle_single	creates an Eagle single precision binary model
eagle_double	creates an Eagle double binary model
postscript	creates a postscript file (generally for laser printers)
dogs	creates a Dogs parametric format file
calcomp	creates a Calcomp plot format file
mif	creates a frame maker interface format file.

Continue to the section 43.3 Mapping 12d Colours to Pens and RGBs or return to 43.2 Defining Plotters - Plotters.4d.

43.2.1.2 Definition and Example of a .tbl File

An example of an output mapping file for use with an Microstation (Intergraph) dgn plotter:

```
// -----------------------------------------------------------------
// File: vicplot.tbl
// -----------------------------------------------------------------
// column 112d colour name              (any length - but only 1st four characters passed)
// column 2Vic roads code               (only 1st four characters passed)
// column 3AS2482 feature code
// column 4description         (any length but if embedded spaces then must be quoted)
// column 5line level                   (between 1 and 64)
// column 6line colour                  (between 1 and 256)
// column 7line weight                  (between 1 and ?)
// column 8line style
//
// notes: if column 2 is a * then the 12d name is transmitted (up to 4 characters that is)
//        columns 9 through 16 are only used for point strings which map into characters

1  PM    38010000  "Permanent Survey Mark    " 49  0  1 0
2  BM    38020000  "Bench Mark               " 49   0  1 0
3  TPEG  31000001  "Title peg                " 50  0  1 0
4  STN   38100000  "Instrument Station       " 50  0  1 0
5  SM    38000000  "Survey mark (general)    " 50  0  1 0
6  PCON  38040001  "Photo control point      " 50  0  1 0
7  CHEK  38000001  "Check profile/point      " 52 10  3 0
8  TRIG  38010000  "Trigonometric Station    " 50  0  3 0
9  BMQS  38020001  "Bench Mark QS-1          " 49 11  1 0
10 BMS   38020002  "Bench Mark S-2           " 49 11  1 0
11 BMSH  38020003  "Bench Mark SH-1          " 49 11  1 0
12 RM    38000001  "Reference Mark - General " 50 11  1 0
* ROD    38000002  "Reference Mark - Rod     " 50 11  1 0        // everything else
```

Continue to the section 43.2.1.3 Definition and Example of a .amf File or return to 43.2 Defining Plotters - Plotters.4d.

43.2.1.3 Definition and Example of a .amf File

An example of an AutoCad output mapping file for use with an Autocad plotter:

```
// ------------------------------------------------------------------
// File: acadplot.amf
// ------------------------------------------------------------------
// column 1  match colour    - can include wild cards * and wild characters ?
// column 2  new name        - not output to DXF
// column 3  ACD layer       -
// column 4  ACD colour      - * for 12d Model colour (mapped to ACD),
//                           BYLAYER for ACD BYLAYER
// column 5  ACD line type- * for 12d Model linestyle,
//                           BYLAYER for ACD BYLAYER
// column 6  ACD text style- not yet used, * for 12d Model text style
//
// Notes:
// 1. column 5 (ACD colour) can only be a number between 0 and 256, or * or BYLAYER
// 2. The DWG/DXF file produced needs to be loaded into an existing Autocad drawing
//    which has the ACD layers and linestyles defined.
// column 1    2       3      4              5              6
//
        1    CONT    ljg1    1              1              *
        2    TOP     ljg2    2          CONTINUOUS         *
        3    TOP      *      3              *              *
        4    TOP     ljg4    *           BYLAYER           *
        5    TOP     ljg5   BYLAYER       DASH             *
```

Continue to the section 43.2.2 Some Plotters in the Default Plotters.4d File or return to 43.2 Defining Plotters - Plotters.4d.

43.2.2 Some Plotters in the Default Plotters.4d File

Full Scale >PDF black

The 12d standard **colour numbers 1-15** & **316** plot with **0.25mm black** pens.

The **"pen*"** (**400 series**) colour numbers plot with **black pens** of **specific weights**.

All **"ppf*"** (**900 series**) colour numbers plot with black/grey scale pens of specific weights.

Full Scale >PDF black string weight

If the string has no weight then it is the same as **Full Scale >PDF black** but if the string has a weight then the string weight is used.

So if a non standard weight is required for a particular string then you set the weight for that string. Otherwise do not give the string a weight.

Full Scale >PDF colour

The 12d standard **colour numbers 1-15** & **316** plot with **0.25mm pens** of the **matching colour**, **except**:

a) **white** (**7**) plots to **black**,

b) **yellow** (**5)** plots to a **darker yellow**.

The **"pen*"** (**400 series)** colour numbers plot with **black pens** of **specific weights**.

All **"ppf*"** (**900 series)** colour numbers plot with **black/grey scale** pens of **specific weights**.

Full Scale >PDF colour string weights

If the string has no weight then it is the same as **Full Scale >PDF colour** but if the string has a weight then the string weight is used.

So if a non standard weight is required for a particular string then you set the weight for that string. Otherwise do not give the string a weight.

Half Scale >PDF black, Half Scale >PDF black string weight
Half Scale >PDF colour, Half Scale >PDF colour string weight

These are the same the Full Scale plotters EXCEPT that the **pen width is 0.125mm**

Continue to the section 43.2.3 Example of a Plotters.4d File or return to 43.2 Defining Plotters - Plotters.4d.

43.2.3 Example of a Plotters.4d File

An example of a plotters.4d file is:

```
// -----------------------------------------------------------------
// File:     plotters.4d
// Date:     2014-03-17
// Revised:
// Use:      Version 11+ Standard Plotter Names
// -----------------------------------------------------------------

// 12d Model model
plotter "model" {
  group     "12d" //this puts "model" at the bottom of the list in the hard-wired "12d" group
  engine    model
  extension ""
}

// PDF

plotter "PDF black" {
  group     "PDF/Full Scale"
  engine    pdf_12d
  extension ".pdf"
  map_pens true
  map_file "pmf_black.pmf"
}
plotter "PDF black string weight" {
  Group         "PDF/Full Scale"
  engine        pdf_12d
  extension     ".pdf"
  map_pens      true
  map_file      "pmf_black.pmf"
  stringhalt true //explicit string weights override PMF weights
}
plotter "PDF colour" {
  Group     "PDF/Full Scale"
  engine    pdf_12d
  extension ".pdf"
  map_pens true
  map_file "pmf_colour.pmf"
}
plotter "PDF unmapped" {
  group     "PDF"
  engine    pdf_12d
  extension ".pdf"
  map_pens false
}

//DWG (and DXF/DXB)

plotter "DWG unmapped" {
  group "DWG"
  //engine    dwg_12
  //engine    dwg_13
  //engine    dwg_14
  //engine    dwg_2000
  //engine    dwg_2002
```

```
  //engine    dwg_2004
  //engine    dwg_2005
  //engine    dwg_2006
  //engine    dwg_2007
  engine    dwg_2010
  extension ".dwg"
}
plotter "DXF 2000 with amf" {// output to a dxf file using an autocad map file
  engine    dxf_2000
  extension    ".dxf_2000"
  map_file    "acadplot.amf"
}

// Windows

plotter "Windows colour" {
  group    "Windows/Full Scale"
  engine   windows
  map_pens true
  map_file "pmf_colour.pmf"
}
```

Continue to the section 43.3 Mapping 12d Colours to Pens and RGBs or return to 43.2 Defining Plotters - Plotters.4d.

43.3 Mapping 12d Colours to Pens and RGBs

12d Model uses up to 10,240 different colours and when it comes to plotting, it must be decided how these colours are mapped for the particular plotter device being used.

Some physical plotters may print colours and others only black. And even on a colour plotter, sometimes you may want to print out in black, or black and grey scale.

When plotting to devices such as PDF, or CAD systems such as AutoCad and Microstation, no physical plot is created but the plot image may need to be send colours to special colours, layers and styles in the other system.

For plotting to actual plotters, there are two methods of specifying the colour to pen mappings.

(a) pen mapping using Colours.4d

A simple "colour to pen number" mapping which is mainly used for older pen plotters that only had a fixed number of pens that could be used. It also provides the default pen numbers if they are not mentioned in the *plotters.4d* file. See 43.3.1 Pen Mapping in Colours.4d.

(b) **plotter mapping - Plotters.4d**

An extended colour to pen number mapping is more often used which not only maps colour numbers to pens but can also include a weight (thickness) in millimetres for each pen (**pen_mapping**).

Plus the red, green, blue definition for the pen numbers used on the plotter (**pen_colour**)

This is for the physical plotters such as laser, electrostatic, inkjet and bubblejet plotters, and the non-physical plotters such as pdf. See 43.3.2 Plotter Mapping File.

So for a plotter mapping

Colour number in 12d Model $\xrightarrow{\text{pen\_mapping}}$ Pen number and weight $\xrightarrow{\text{pen\_colour}}$ RGB for Pen number

As Plotter mapping is an extension of pen mapping, only **one** of the two is used for a plot.

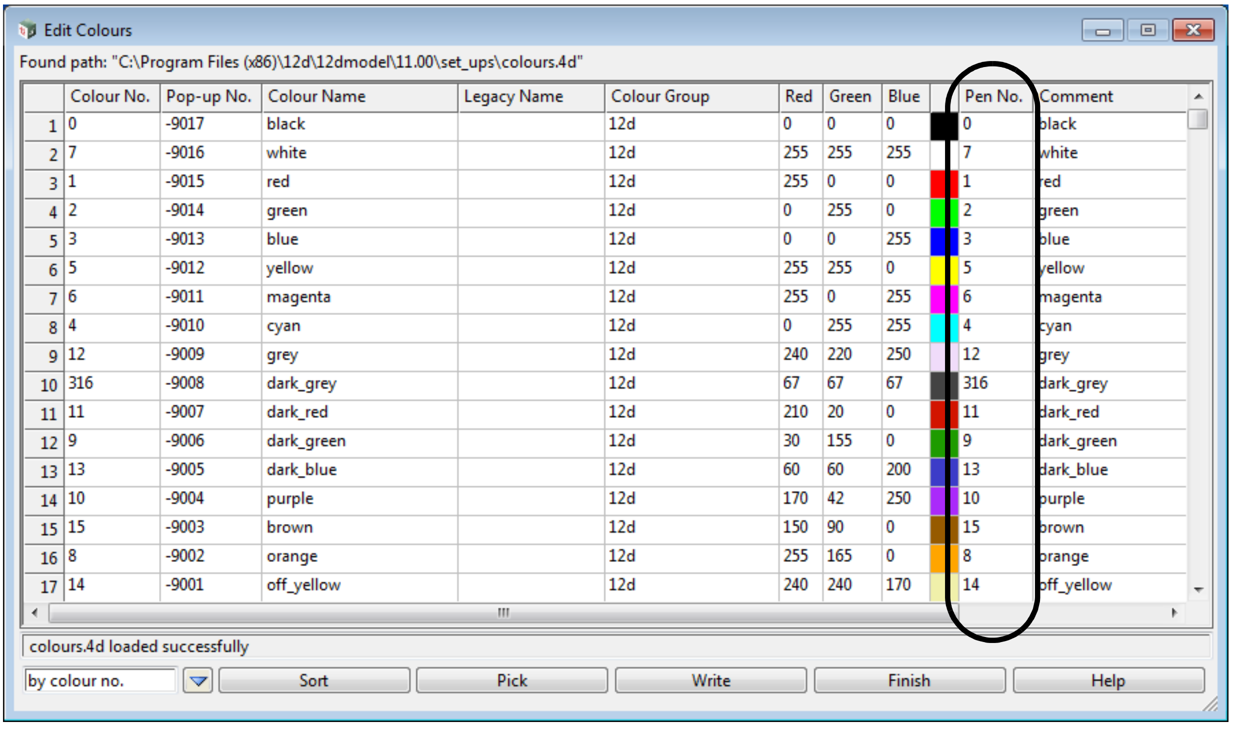
See

43.3.1 Pen Mapping in Colours.4d.
43.3.2 Plotter Mapping File.

Or return to 43 Plotters and Plotting.

43.3.1 Pen Mapping in Colours.4d

The ***colours.4d*** file defines the rgb for drawing the colours on a view and also defines the **default plotter pen** associated with the colour numbers.



	Colour No.	Pop-up No.	Colour Name	Legacy Name	Colour Group	Red	Green	Blue		Pen No.	Comment
1	0	-9017	black		12d	0	0	0		0	black
2	7	-9016	white		12d	255	255	255		7	white
3	1	-9015	red		12d	255	0	0		1	red
4	2	-9014	green		12d	0	255	0		2	green
5	3	-9013	blue		12d	0	0	255		3	blue
6	5	-9012	yellow		12d	255	255	0		5	yellow
7	6	-9011	magenta		12d	255	0	255		6	magenta
8	4	-9010	cyan		12d	0	255	255		4	cyan
9	12	-9009	grey		12d	240	220	250		12	grey
10	316	-9008	dark_grey		12d	67	67	67		316	dark_grey
11	11	-9007	dark_red		12d	210	20	0		11	dark_red
12	9	-9006	dark_green		12d	30	155	0		9	dark_green
13	13	-9005	dark_blue		12d	60	60	200		13	dark_blue
14	10	-9004	purple		12d	170	42	250		10	purple
15	15	-9003	brown		12d	150	90	0		15	brown
16	8	-9002	orange		12d	255	165	0		8	orange
17	14	-9001	off_yellow		12d	240	240	170		14	off_yellow

The **Plots=>Pen mapping** option or any other option that brings up the **Edit Colours** panel. can be used to define the correspondence between 12d Model colours. See 39.2.7.2 Colours File (colours.4d).

Continue to 43.3.2 Plotter Mapping File or return to 43.3 Mapping 12d Colours to Pens and RGBs.

43.3.2 Plotter Mapping File

The **plotter mapping file** can be used use with pen plotters but is more specifically designed for laser, electrostatic and inject plotters and allows the user to:

(a) map 12d Model colours to particular plotter pens and also specify a width (thickness or weight) to be used for the pen.

and

(b) specify the red, green and blue mix for the pens.

The ***plotter mapping file*** to be used can be set by either

1. by the environment variable PLOTTER_MAPPING_4D, or is the file *pmf.4d* found in the standard set up areas.
2. defined for a particular plotter in the *Plotters.4d* file. See 43.2 Defining Plotters - Plotters.4d

There are also two **special formats** of the plotter mapping file which are used for plotting to Autocad dwg/dxf and Microstation dgn. The special mapping files use the 12d Model **colour number** as a key to tables which control how the information is passed to DWG/DXF and DGN.

3. .tbl file used with plotting to Microstation dgn
4. .amf used with plotting to Autocad dwg, dxf or dxb

And a superseded method using a plotter mapping file in the **Plotter Mapping Table** panel.

5. Now superseded - the option **Plots =>Plotting setups =>Plotter mapping**. See 25.8.13 Plotter Mapping Table.

The definition of the *Plotter Mapping File (pmf file)* is given in the next section 43.3.2.1 Definition of a Plotter Mapping File and the initiation performed before the pmf file is used is described in 43.3.2.2 Initialisation of the Plotter Mapping File.

43.3.2.1 Definition of a Plotter Mapping File

The plotter mapping file (.pmf) consists of two default values followed by the two sections pen_mapping and pen_colours:

(a) the ***default_weight*** command that goes before the *pen_mapping* or *pen_colours* sections and which is used as the weight for any colour number that is mapped to a pen number but not given a weight in the *pen_mapping* section.

```
default_weight        weight
```

(b) the ***default_colour*** command that also goes before the *pen_mapping* or *pen_colours* sections and which defines the default red, green and blue for any pens used but not set in the *pen_colours* section.

```
default_colour        red        green        blue
```

(c) **pen_mapping** section which defines which pen a colour number is mapped to, and the weight for the pen.

The *pen_mapping* section consists of the key word **pen_mapping** followed by **{** and then one line for each colour number being mapped with the colour number, then the plotter pen number and finally the pen weight that the colour number is mapped to. This is finished with a **}**.

```
pen_mapping {                        // pen mapping and weight table

//          12d        plotter      weight
//          colour     pen no       for pen

            0          1            0.15
            1          2            0.25
            3          1            0.5
//          etc.
}
```

(d) **pen_colours** section which defines the red, green and blue values to be used for the pens on the plotter.

The *pen_colours* section consists of the key word **pen_colours** followed by **{** and then one line for each plotter pen having the rgb defined with the plotter pen number followed by the red, green and blue values for the plotter pen. This is finished with a **}**.

```
pen_colours {            // pen colours table

// plotter
// pen      red        green      blue       values 0-255 for red, green and blue
    0       0          0          0             // pen 0 is black
    1       255        0          0             // pen 1 is red
//          etc.
}
```

Continue to the section 43.3.2.2 Initialisation of the Plotter Mapping File or return to 43.3.2 Plotter Mapping File or 43.3 Mapping 12d Colours to Pens and RGBs.

43.3.2.2 Initialisation of the Plotter Mapping File

Before the ***plotter mapping file*** is used

(a) a default *pen_mapping* table is constructed from the colours in the *colours.4d* file by:

colour n goes to pen n *default_weight*

where **default weight** is the *default_weight* if one is given in the specified *plotter mapping file* or zero if the default_weight does not exist.

(b) a default *pen_colours* table for pens 0 to 10,240 is set up by:

pen n *default_red* *default_green* *default_blue*

where *default_red*, *default_green*, *default_blue* are the red, green and blue of the *default_colour* if one is given in the specified *plotter mapping file*, or rgb 0 0 0 if the default_colour does not exist.

The **plotter mapping file** is then processed and over writes any of the above initial mapping values.

This initialisation sequence means that if a *colour number* is **not** listed in the *pen_mapping* section of the pmf file, then the pen mapping as given for that colour number in the *colours.4d* file is used. Hence the *pen_number* in the *colours.4d* file is the **default pen number**.

Continue to the next section 43.4 Sending Plots to a Plotter or return to 43.3.2 Plotter Mapping File or 43.3 Mapping 12d Colours to Pens and RGBs.

43.4 Sending Plots to a Plotter

When the plot file is created, it can be passed to a user specified program or batch file that can do things like automatically direct the plot file to a specific plotter.

The mechanism is that as plots are created, 12d Model ***can fire up a user supplied batch file/ program with the plot file as the first argument.***

By checking the plot name suffix, the batch file could for example, decide which plotter the plot must be sent to.

The environment variable

PLOTTER_4D *points to batch file/program*

points to the batch file/program which can be fired up whenever a plot is generated. The name of the plot is given as the first parameter of the batch file.

If the tick box **Send plots** in the panel tab **System Settings** of the panel **Defaults** (given by the menu option **Utilities** => **Default**) is set to **on**, the batch file/program is run as each plot is created.

If more than one plot is created by an option (e.g. x plot) then the batch file/program is called separately for each of the plots.

An example of a batch file to send the plot to port lpt1 for Windows would be

```
@echo off
copy %1  lpt1
```

An example for Windows which looks out for hp files is

```
@echo off
echo.
echo ----------------------------------------------------
:next_file
if "%1" == "" goto done
echo %1 | find /I ".hp" > nul
if ERRORLEVEL 0 if not ERRORLEVEL 1 goto hp_plotter
echo Plotting file %1
shift
goto next_file
:hp_plotter
echo Plotting %1 to HP plotter
copy %1 \\server_name\printer_name
shift
goto next_file
:done
echo ----------------------------------------------------
```

Continue to the next section 43.5 Sheet Sizes or return to 43 Plotters and Plotting.

43.5 Sheet Sizes

For plot frames, long and x plots, the overall size of the plot sheet can be given by a pop-up containing defined sheet size.

The sheet size names, width and heights can be specified by the user in a file named **sheets.4d** which is in the normal set up areas, or is pointed to by the environment variable SHEET_SIZES_4D.

The layout of the sheet sizes file is given in the section 39.2.7.4 Sheet Sizes File (sheets.4d).

Continue to the next section 43.6 Text Units in Plots or return to 43 Plotters and Plotting.

43.6 Text Units in Plots

Text occurs in 12d Model plots in a number of ways -

- plotting **text strings**, text in super strings and other strings.
- automatic text such as grid values, x-section and long-section plot annotation.
- text within linestyles.

The most difficult thing about text is that because of the different uses of text, there needs to be more than one systems of **units** to define text heights.

The height of text for a given textstyle is defined to be the height of a capital A. However, in ***12d Model***, there are three methods of defining the units for measuring this height.

- world units - the units used for data
- screen units - pixels
- plot paper units - millimetres.

World Units

World units are the units of user data. For most users, the base unit for user data is metres. However ***12d Model*** is a almost a dimensionless system and the base unit is totally dependent on the user.

The height of world text when displayed in a view depends upon on the ***text height*** and the ***scale*** of the view.

When plotted, the height that world text appears on a plot sheet is the same as for any data defined in world units - the height depends on the **scale** used for the plot.

Text heights that are only given in world units have **(w)** after them.

For some text, the choice of units is either world, pixel or paper units. The text parameters then have a **(u)** after them.

Screen Units - Pixels

When screen units (pixels) are used, the text is a fixed height on the screen. If the user zooms in on text given in pixels, the text remains the same height.

To have a height on a plot, screen unit text needs a height defined in millimetres.

For some screen text, both a pixel and a millimetre height is supplied when the text is defined.

For text with only a pixel height, there is a plotting multiplication parameter called **pixels-to-millimetres** which is used to convert pixel heights to plot paper heights. The value of pixels-to-millimetres is set using the **Plots =>Plotting setups =>Pixels to mm** option and is stored for the project. See 25.7.5 Pixels to mm.

Text heights that are only given in pixels have a **(pix)** or **(p)** after them.

For some text, the choice of units is either world, pixel or paper units. The text parameters then have a **(u)** after them.

Plot Paper Units - Millimetres

Text defined in plot paper units (millimetres) has a well defined height on a plot sheet.

Text heights that are only given in millimetres have **(mm)** after them.

For some text, the choice of units is either world, pixel or paper units. The text parameters then have a **(u)** after them.

Return to 43 Plotters and Plotting.

44 Text Plot Parameters

IMPORTANT WARNING

This information on the text version of teh plot parameters only includes those parameters needed up to ***12d Model*** 7 and is only included for those users still working with old text ppf files.

Text PPFs have been replaced by the ***PPF Editors*** - see 26 PPF Editors.

This appendix contains information about the definitions in the text versions of the plot parameter files. The text plot parameter files are not normally seen by users because the *Plot Parameter* editors create binary plot parameter files.

See

44.1 Plot Frame Plot Parameter File
44.2 Cross Section Plot Parameter File
44.3 Long Section Plot Parameter File
44.4 Pipeline Plot Parameter File
44.5 #Include in Plot Parameter Files

44.1 Plot Frame Plot Parameter File

The plot frame section plot parameters are placed in a file with ending **.ppf**

Each parameter consists of a parameter name followed by one or more spaces and then the parameter value. There is only one parameter per line.

Anything on a line after a double forward slash **//** is considered to be a comment.

The set of all parameters for the plot frame plot is enclosed within a set of curly brackets **{ }** with the header

plot_frame_plot "plot *set name"*

before the curly brackets.

That is,

```
plot_frame_plot  "plot set name"  {
            plot parameters
            one per line
}
```

If there is more than one plot_frame_plot parameter set in the file, only the first set is used.

There may also be parameter sets for other plot types such as section_long_plot in the same file. The other sets will be ignored when doing plot frame plots.

The only parameters not connected with title block file are

view_name	*view_name*	// name of view used for information // on the plot
single_frames	*frame_name*	// name of plot frame to be plotted
model_of_frames	*model_name*	// model of plot frames which may be

// plotted as a group

Both these parameters exist in the **plot frames plot** panel and when the parameter file is first read, if either of these parameters exist in the plot parameter file then they will be used to replace the corresponding parameters in the panel.

However, if the parameters are subsequently modified in the panel, the new panel value will be the value used for plotting.

44.1.1 Title Block Information

A plot frame plot can have a standard ***12d Model*** title block or a user defined title block.

For a user defined title block, the title block drawing commands are kept in a file whose name is given by the user when setting up the plot frame. The title block drawing commands are almost identical to the linestyle drawing commands.

If a user defined title block has been set for the plot frame, plot frame parameters can be used to pass information though to the title block.

User Title Block

Some of the plot parameters are used to pass information down to variables in a user defined title block specified in the plot frame.

The parameters are:

time_format	*text*	// format for $time
title_1	*text*	// passed down to $title_1
title_2	*text*	// passed down to $title_2
user_text_n	*text*	// where n = 1,2,... 1000 // passed down to $user_text_n
start_page_number	*integer*	// used as the starting value for $page_number. // if missing, $page_number starts at 1.
start_drawing_number	*integer*	// added to $drawing_number in title block file // if missing, $drawing_number starts at 1.
drawing_number_prefix	*text*	// passed down to $drawing_number_prefix
drawing_number_postfix	*text*	// passed down to $drawing_number_postfix

Notes

1. A warning is given if the keyword in a plot parameter file does not exist.
2. A warning is also given if the key word pair is defined more than once in a ppf.

Please continue to the next section 44.2 Cross Section Plot Parameter File.

44.2 Cross Section Plot Parameter File

The cross section plot parameters are placed in a file with ending **.ppf**

Each parameter consists of a parameter name followed by one or more spaces and then the parameter value. There is only one parameter per line.

Anything on a line after a double forward slash // is considered to be a comment.

The set of all parameters for the cross section plot is enclosed within a set of curly brackets **{ }** with the header

section_x_plot   "plot set name"

before the curly brackets.

That is,

```
section_x_plot  "plot set name"   {
            plot parameters
            one per line
}
```

If there is more than one section_x_plot parameter set in the file, only the first set is used.

There may also be parameter sets for other plot types such as section_long_plot in the same file. The other sets will be ignored when doing cross section plots.

The plot parameters are documented in following groups:

For the *Plot Sheet layout*, please continue to the section 44.2.1 Plot Sheet Layout.

Boxes and datum area, please continue to the section 44.2.2 Boxes and Datum Area.

Only for the Centre line case:

Datum area, please continue to the section Datum Line.

Labelling the centreline, style etc., please continue to the section Labelling the Centreline Chainage.

Only for the Boxes case:

Datum area, please continue to the section Datum Line.

Labelling the centreline chainages, please continue to the section Labelling the Centreline Chainage.

Defining boxes and what is labelled in them, please continue to the section 44.2.5 Defining Boxes.

For Both the Centre Line and Boxes case:

Graph area parameters, please continue to the section 44.2.6 Graph Area.

Grade labelling, please continue to the section 44.2.7 Grade Labelling.

Labelling points on x-sections, please continue to the section 44.2.8 Labelling Points of the X-Sections

Labelling cuts, please continue to the section 44.2.9 Labelling Cuts of X-Sections Through Strings in a Model.

Hatching cut/fill, please continue to the section 44.2.10 Hatching Cut and Fill Areas.

Drawing extra x-sections, please continue to the section 44.2.11 Extra Models of X-Sections.

Sorting x-sections, please continue to the section 44.2.12 Sorting X-Sections by Chainage.

Title block information, please continue to the section 44.2.13 Title Block Information.

Panel modifying parameters, please continue to the section 44.2.14 Parameters that Modify Fields In the Cross Plot Panel.

Example, please continue to the section 44.2.16 Example of a Cross Section Plot Parameter File.

44.2.1 Plot Sheet Layout

X-sections are normally generated at chainages along a given centreline. This centreline chainage is stored with each x-section string.

The chainages of the actual x-sections are referred to as **offsets** from the centreline position rather than x-section chainages. The offsets of the x-section are set up so that the zero offset occurs where the x-section crossed the centreline string.

The x-sections along the centreline are stored in the one model (the primary model) which is then used to generate the cross section plot.

model_to_plot	*text*	// name of the model of x-sections

Each x-section from the primary model of x-sections generates its own sub-plot for which the x-section is the primary string. Hence the cross section plot consists of many individual plots drawn on one or more plot sheets.

Each plot sheet is considered to have only positive co-ordinates with the origin (0,0) in the left hand corner. The units for the plot are millimetres.

The overall size of the plot sheet is given by either a defined sheet size, or by the width and height of the plot given in millimetres and separated by one or more spaces.

sheet_size	*text*	// sheet name, or
	"mm *mm"*	// sheet size: width height

The sheet size name, width and heights can be specified by the user in a file named **sheets.4d** which is in the normal set up areas, or is pointed to by the environment variable

SHEET_SIZES_4D	*file*	// file of plotter sheets sizes

The plotting area is restricted to within the plot sheet by giving margins which are:

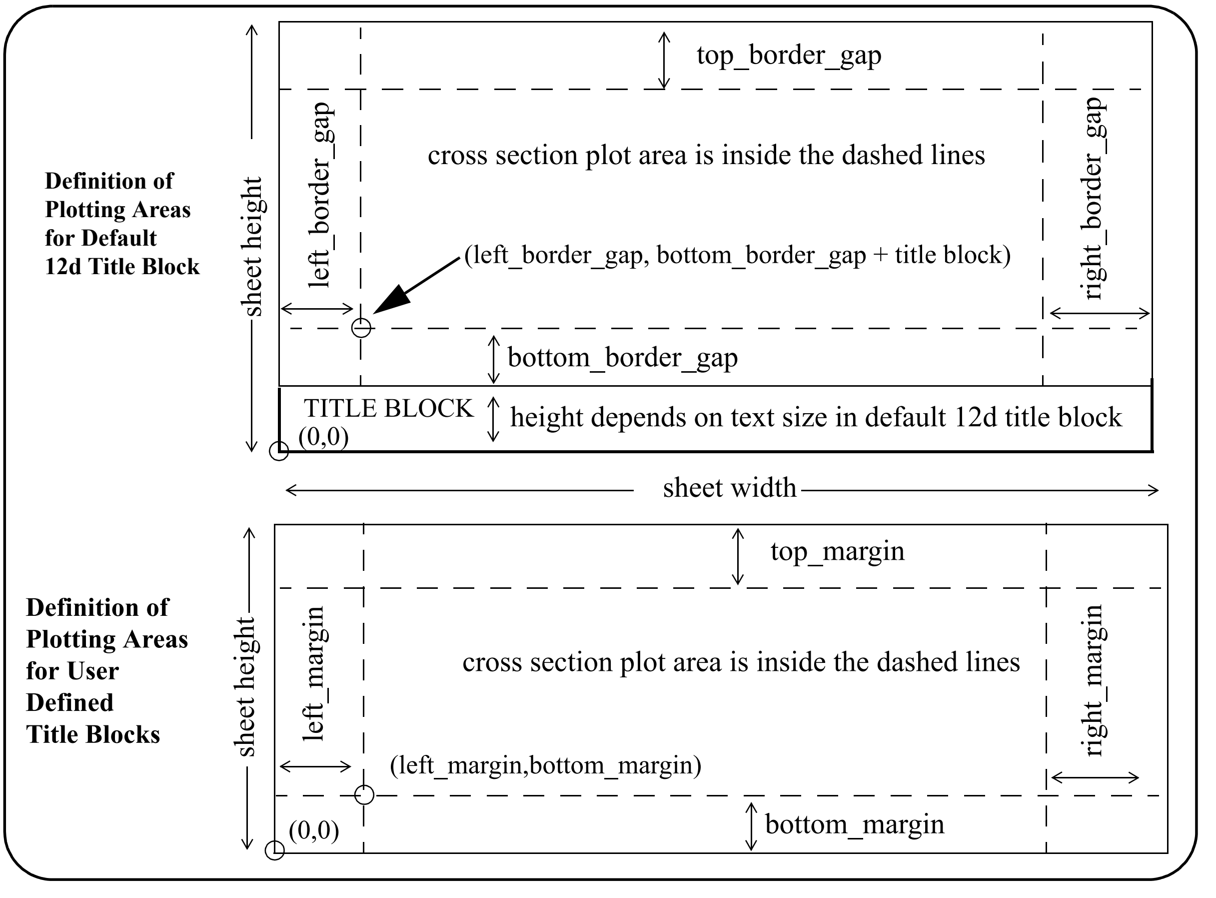
If a *User Defined Title Block* is used:

left_margin	*mm*
right_margin	*mm*
top_margin	*mm*
bottom_margin	*mm*

If the *default 12d title block* is used, then the size of the bottom of title block depends on the text size. The following parameters are used in the default title block case and the *bottom_border_gap* is added to the calculated height of the bottom of the title block.

left_border_gap	*mm*
right_border_gap	*mm*
top_border_gap	*mm*
bottom_border_gap	*mm*

Because the user can easily select from the plotting panel whether a User Defined Title Block or the default 12d title block is used, both sets of margin and gap parameters can exist in the one plot parameter file.



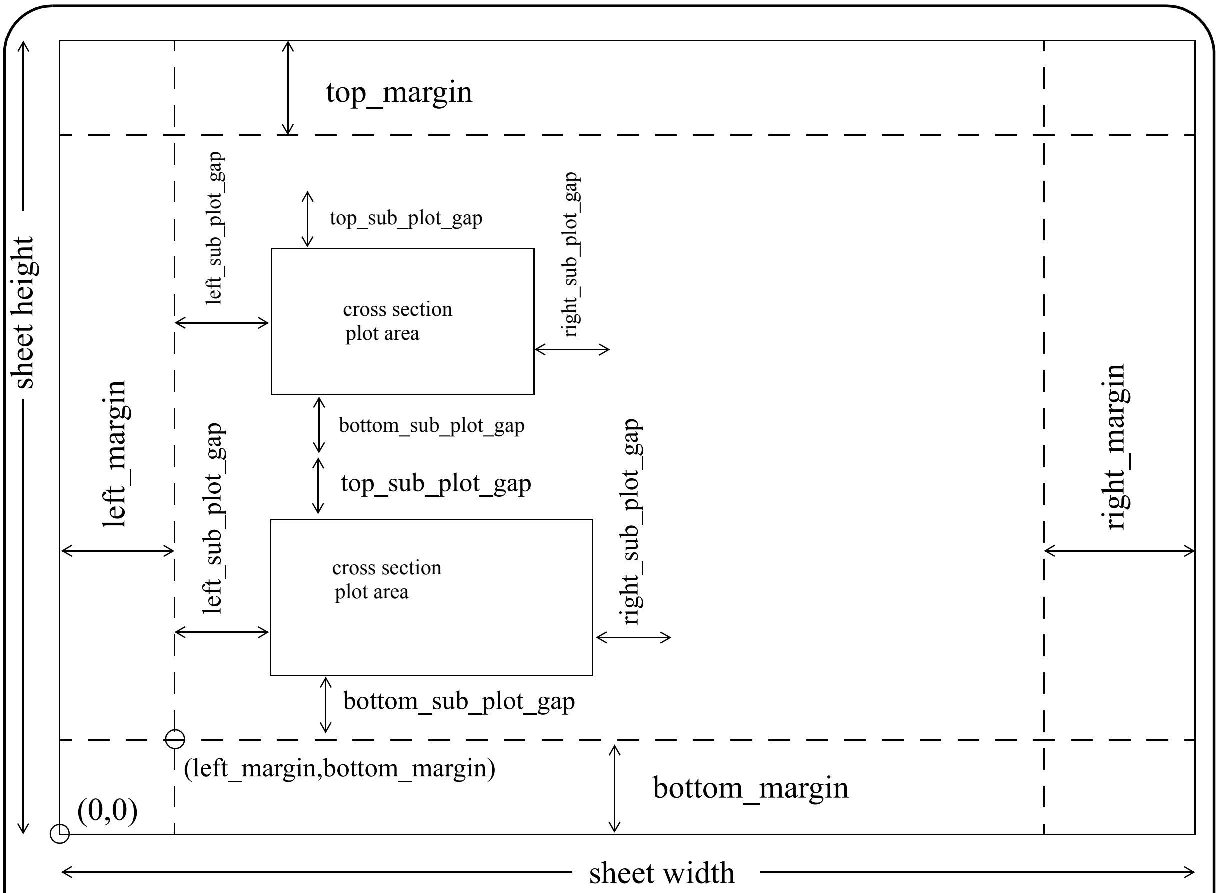
The x-sections are plotted in the order they occur in the x-section model and start being plotted at the bottom left hand corner of the cross section plotting area.

The individual x-section plots are then drawn going up the column, and when the column is full, start from the bottom of the next column.

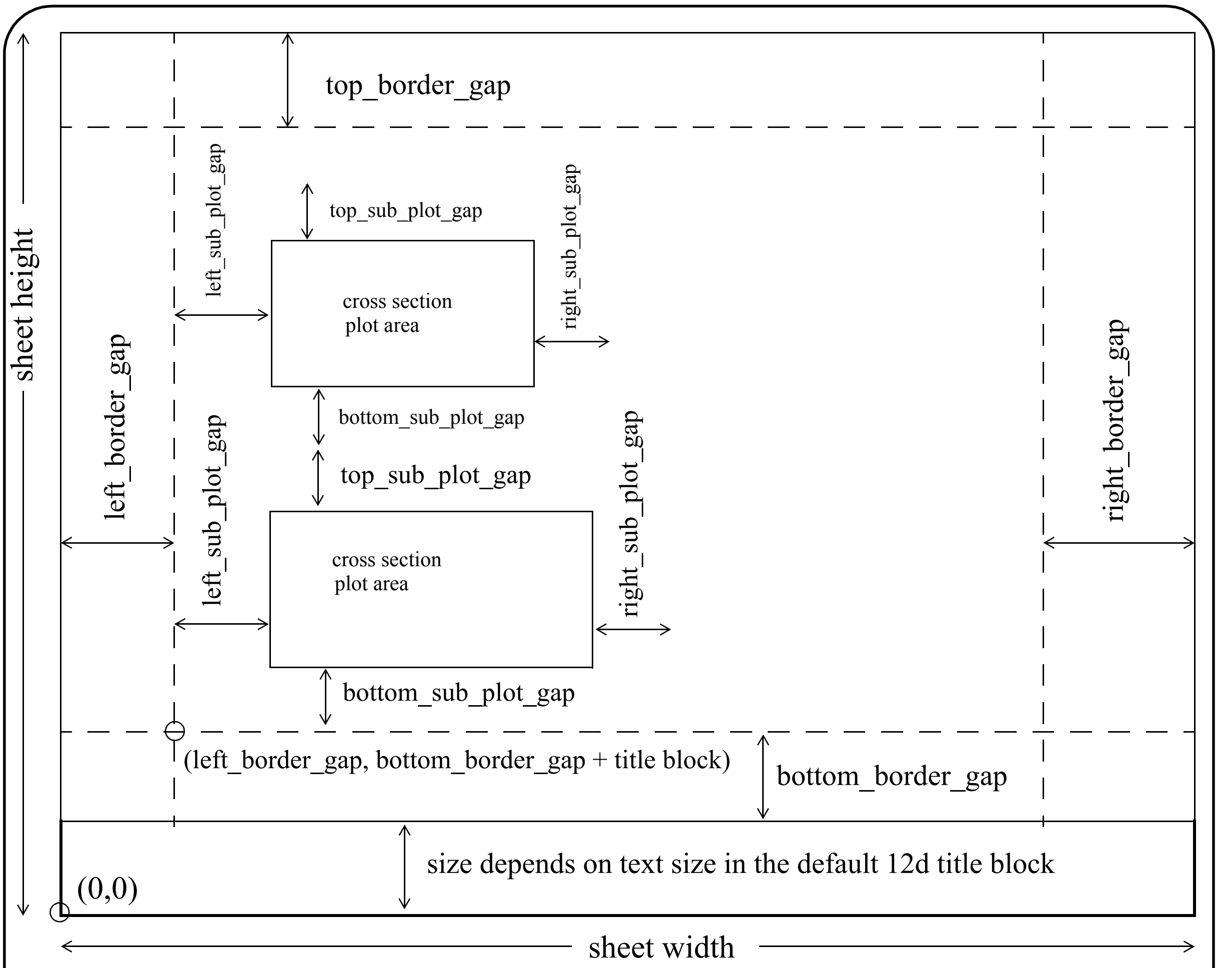
When a sheet is full, a follow on sheet is created.

Each individual x-section sub-plot is positioned with the surrounding gaps:

left_sub_plot_gap	*mm*
right_sub_plot_gap	*mm*
top_sub_plot_gap	*mm*
bottom_sub_plot_gap	*mm*



Definition of Plotting Areas for User Defined Title Blocks



Definition of Plotting Areas for Default 12d Title Block

If required, all the sub-plots in a column can be automatically positioned up so that the zero offsets (the centrelines) of each x-section line up.

line_up_cl	*yes/no*	// yes - line up zero offsets

The width of the plot can be a fixed distance left or right of the centre line (zero offset) or for the full section plus an extra left and right distance:

absolute_extensions	*yes*	// The section goes from the
		// left_extension offset on the left to the
		// right_extension offset on the right.

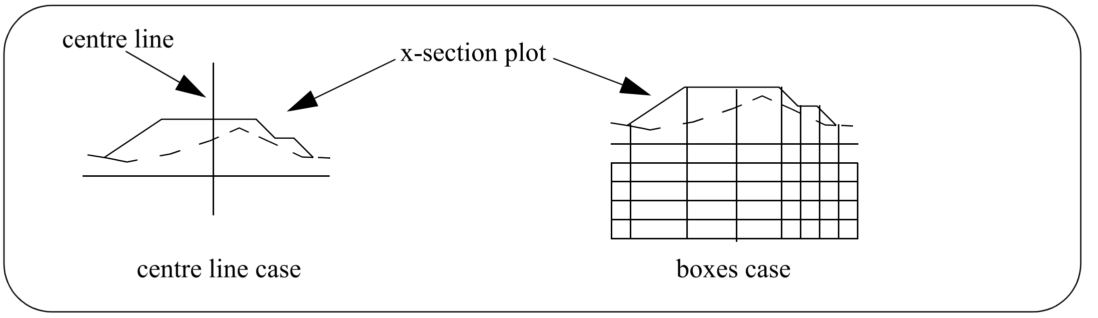
	no	// The section goes for the entire section
		// length plus the left and right extension
		// distances.

left_extension	*world-units*	// left extension value
right_extension	*world-units*	// right extension value

The x-section can be drawn and labelled with either

(a) centreline case - the x-section is plotted and an upright, and the offset and height value at the zero offset (normally the centre line position)

(b) boxes case - the x-section is plotted and the heights of the x-section and the tins at all the x-section points are labelled in boxes under the plot of the x-section.

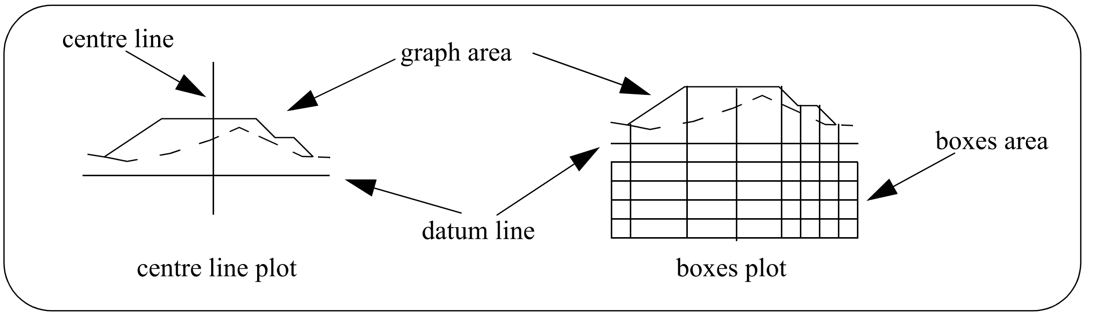


The x-section sub-plot itself consists of the three regions - graph, datum and boxes.

The **graph area** is the area where the actual plots of the strings are drawn. This exists for both the centreline and boxes case.

The **datum area** is the region between graph area and the datum line. This exists for both the centreline and boxes cases.

The **boxes area** is where the offset values and the heights for the strings drawn on the x-section plot are labelled. This only exists for the boxes case.



For both cases, the x-section sub-plot can be labelled with other information such as

(a) grades across the x-section

(b) points across the x- section

(c) cuts the x-section makes through strings

(d) cut and fill areas

All the required parameters will be described in the following sections.

Please continue to the next section 44.2.2 Boxes and Datum Area.

44.2.2 Boxes and Datum Area

The x-section can be labelled with either

(a) an upright, and the offset and height value at the zero offset (normally the centre line position)

(b) the heights of the x-section and the tins at all the x-section points.

The choice is given by the parameter ***label_type***:

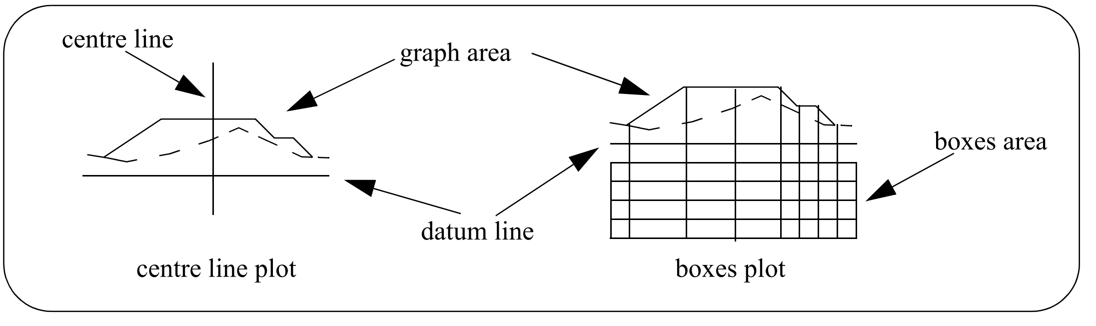
label_type	"centre *line"*	// type (a)
	boxes	// type (b)

A datum line exists for both cases.

For the boxes case, a box area for the offset and heights is created below the datum line. The available parameters for tailoring the box area will be given after describing the datum line parameters.

For the centre line case, the centre line and offset and height of the centre line are shown.

In both cases, the actual cross section plot is drawn above the datum line in the graph area.



Please continue to the next section 44.2.3 Centre Line Case for the centre line parameters.

Please continue to the section 44.2.4 Boxes Case for the boxes parameters.

44.2.3 Centre Line Case

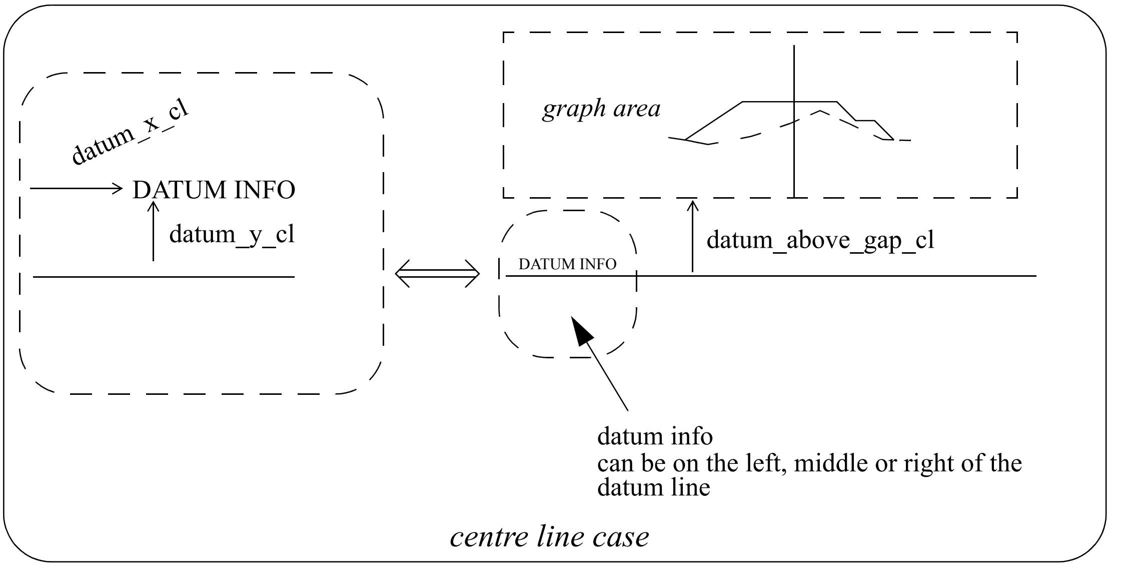
Datum Line

Each x-section sub-plot can be labelled with the datum value for the plot of the x-section string.

This DATUM INFO label is made up of the text strings:

"datum_name" followed by the *datum-value*

and is above the datum line.



For the centreline case, the graph area is positioned the distance ***datum_above_gap_cl*** above the datum line.

datum_above_gap_cl	*mm*	// dist from datum line to bottom of the // graph area

The *datum_above_gap_cl* can be zero or positive.

The roundoff for the datum value is specified by the user (default 1.0) and the datum is automatically calculated for each sub-plot, and labelled.

The datum value can be placed on the left, centre or right side of the datum line.

datum_roundoff	*1.0*	// value to roundoff the datum value to // e.g. 0.5, 0.2, 1.0 (default 1.0)

datum_decimals	*integer*	// number of decimal places to display
		// the datum value (default 1).
		// If > 0, trailing zeros are removed after
		// the decimal point.
		// If <0, the absolute value is taken as the
		// number of decimal places to report
		// i.e. no trailing zeros are removed

datum_side_cl	*0*	// datum text in middle of datum line
	1	// " " " " left of datum line (default)
	2	// " " " " right of datum line
datum_linestyle	*linestyle*	// datum line linestyle (default solid)
datum_name	*text*	// text to write before the datum value
datum_textstyle	*text*	// textstyle for datum information
datum_text_size	*mm*	// size of datum text and value
datum_colour	*colour*	// colour of the datum text
datum_line_colour	*colour*	// colour of the datum line

datum_text_justification_cl	*just*	// justification for datum text
		// NOTE - this is not normally required since
		// by default the text justification is set to
		// match datum_side_cl

datum_x_cl	*mm*	// distance to move the datum text // along the datum line

datum_y_cl	*mm*	// distance to raise the datum text
		// above the datum line (used to be
		// called datum_offset)

The *datum_x_cl* and *datum_y_cl* can be positive, zero or negative.

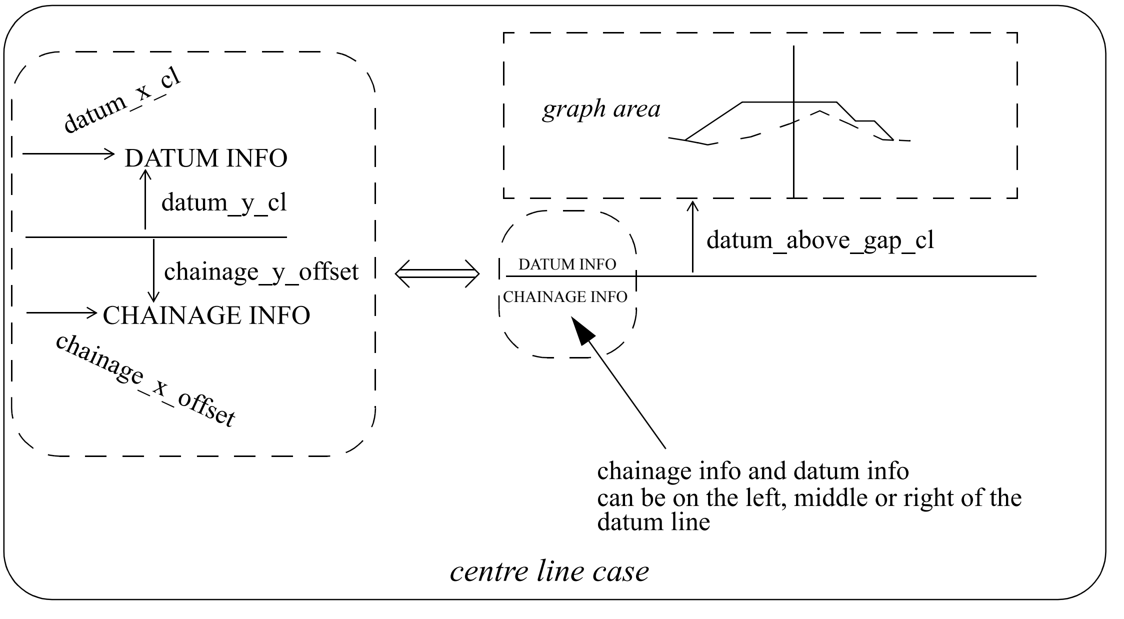
Labelling the Centreline Chainage

Each x-section sub-plot can be labelled with the centreline chainage of the x-section string.

This CHAINAGE INFO label is made up of the text strings:

"chainage_title" followed by the *chainage-value*

and is drawn under the datum line.



The chainage value can be placed on the left, centre of right side of the datum line.

chainage_side_cl	*0*	// text in middle of datum line (default)
	1	// " " " " left of datum line
	2	// " " " " right of datum line

The parameters controlling the labelling are:

chainage_label	0/1	// 1 = label sub-plot with centreline // chainage, 0 don't label.
chainage_title	*text*	// text before the chainage value

chainage_decimals	*integer*	// number of decimals in the chainage
		// value. If <0, the absolute value
		// is taken as the number of decimal
		// places i.e. no trailing zeros are
		// removed for the values in the
		// chainage values.

chainage_colour	*colour*	// colour of the text

chainage_size	*mm*	// size of the text
chainage_textstyle	*colour*	// textstyle for the chainage label
chainage_text_justification_cl	*just*	// justification of the chainage text
chainage_x_offset	*mm*	// x position of text
chainage_y_offset	*mm*	// y position of text

The *chainage_x_offset* is measured from the beginning of the datum line.

The *chainage_y_offset* is measured from the bottom of the datum line with positive being **down**.

Centre Line Linestyle

The upright at the centre line position can have its own linestyle.

cl_linestyle	*linestyle*	//linestyle for centreline

Labelling the Design Height, X and Y Co-ordinates and Tin Heights at Offset Zero

The values of the height and X and Y co-ordinates of the **primary string** (usually the design cross section) at the zero offset can be labelled. Note that zero offset is normally where the alignment string cuts the cross section.

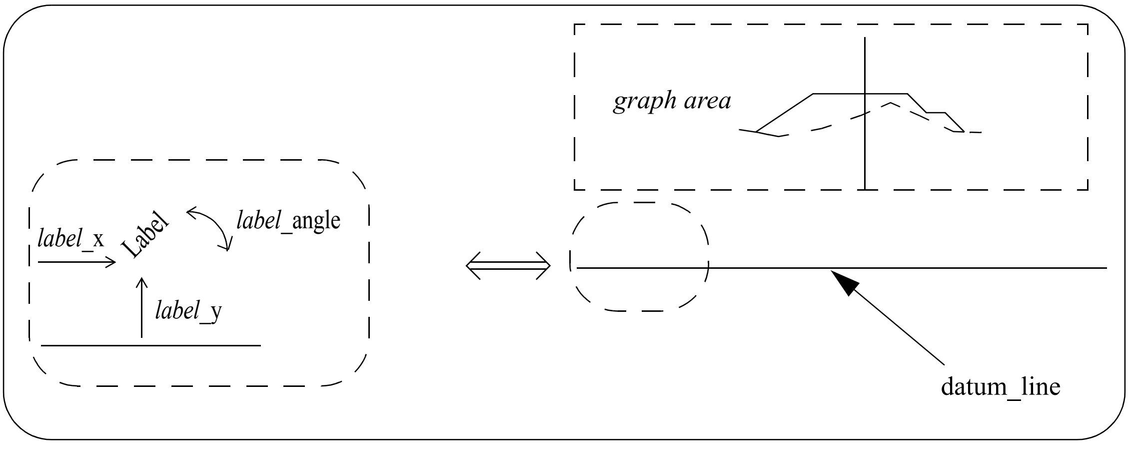
The heights of any tins (such as the natural surface) at the zero offset can also be labelled.

The labels are made up of:

pre_text *value* post_text

where *value* is either a height or a co-ordinate.

The label is positioned at either the left, right or middle of the datum line, with an x and y adjustment and a rotation.



Parameters for labelling the X Co-ordinate at Zero Offset:

primary_x0_draw_mode	*0*	// don't draw the label -default
	1	// draw the label
primary_x0_position	***0***	// label in middle of datum line (default)
	1	// " " " " left of datum line
	2	// " " " " right of datum line
primary_x0_pre_text	*text*	// pre-text for label - def " "
primary_x0_post_text	*text*	// post-text for label - def " "
primary_x0_decimals	*integer*	// number of decimal places to display - def 1
		// If > 0, trailing zeros are removed after
		// the decimal point
		// If < 0, the absolute value is taken as the
		// number of decimal places to report
		// i.e. no trailing zeros are removed
primary_x0_x	*mm*	// x adjustment to position of label - def 0
primary_x0_y	*mm*	// y adjustment to position of label - def 0
primary_x0_angle	*degrees*	// angle of the label - def 0
primary_x0_colour	*colour*	// colour of the label
primary_x0_size	*mm*	// size (in mm) of the label
primary_x0_textstyle	*textstyle*	// textstyle of the label
primary_x0_justify	*just*	// justification for text

Parameters for Labelling the Y Co-ordinate at Zero Offset:

primary_y0_draw_mode	***0***	// don't draw the label -default
	1	// draw the label
primary_y0_position	***0***	// label in middle of datum line (default)
	1	// " " " " left of datum line

	2	// " " " " right of datum line
primary_y0_pre_text	*text*	// pre-text for label - def " "
primary_y0_post_text	*text*	// post-text for label - def " "
primary_y0_decimals	*integer*	// number of decimal places to display - def 1
		// If > 0, trailing zeros are removed after
		// the decimal point
		// If < 0, the absolute value is taken as the
		// number of decimal places to report
		// i.e. no trailing zeros are removed
primary_y0_x	*mm*	// x adjustment to position of label - def 0
primary_y0_y	*mm*	// y adjustment to position of label - def 0
primary_y0_angle	*degrees*	// angle of the label - def 0
primary_y0_colour	*colour*	// colour of the label
primary_y0_size	*mm*	// size (in mm) of the label
primary_y0_textstyle	*textstyle*	// textstyle of the label
primary_y0_justify	*just*	// justification for text

Parameters for Labelling the Height of Primary String at Zero Offset:

primary_height_draw_mode	*0*	// don’t draw the label -default
	1	// draw the label
primary_height_position	***0***	// label in middle of datum line (default)
	1	// " " " " left of datum line
	2	// " " " " right of datum line
primary_height_pre_text	*text*	// pre-text for label - def " "
primary_height_post_text	*text*	// post-text for label - def " "
primary_height_decimals	*integer*	// number of decimal places to display - def 1
		// If > 0, trailing zeros are removed after
		// the decimal point
		// If < 0, the absolute value is taken as the
		// number of decimal places to report
		// i.e. no trailing zeros are removed
primary_height_x	*mm*	// x adjustment to position of label - def 0
primary_height_y	*mm*	// y adjustment to position of label - def 0
primary_height_angle	*degrees*	// angle of the label - def 0
primary_height_colour	*colour*	// colour of the label
primary_height_size	*mm*	// size (in mm) of the label
primary_height_textstyle	*textstyle*	// textstyle of the label
primary_height_justify	*just*	// justification for text

Parameters for Placing Some Text:

extra_text_draw_mode	***0***	// don’t draw the label -default
	1	// draw the label
extra_text_position	***0***	// label in middle of datum line (default)
	1	// " " " " left of datum line
	2	// " " " " right of datum line
extra_text	*text*	// text for label - def " "
extra_text_x	*mm*	// x adjustment to position of label - def 0
extra_text_y	*mm*	// y adjustment to position of label - def 0
extra_text_angle	*degrees*	// angle of the label - def 0
extra_text_colour	*colour*	// colour of the label
extra_text_size	*mm*	// size (in mm) of the label
extra_text_textstyle	*textstyle*	// textstyle of the label
extra_text_justify	*just*	// justification for text

Parameters for Labelling the Height of a Tin at Zero Offset:

A section along the primary string through each tin on the section view is automatically drawn on the cross section plot but the user can specify whether the tin height at the zero offset is labelled or not.

Hence although there may be a number of tins drawn on the section plot, not all of them need to have their height at zero offset labelled.

The default order for labelling the tins is the order that they were added to the view but it is possible to specify which tin is used for labelling by giving the tin name rather than just using the tin on the section view.

In fact, it is possible to use **any tin** in the project **to label** the height at zero offset, **not just those drawn** on the section view.

tin_n_name	*text*	// n=1, ... no of tins on the section view.
		// use the tin called *text* to label the nth
		// row of tin heights.

If a tin of the name given by ***tin_n_name*** **does** not exist, then the plot is not produced and an error message is given.

The parameters for labelling the height of the tin at zero offset are:

tin_n_height_draw_mode	*0*	// don't draw the label -default
	1	// draw the label
tin_n_height_position	***0***	// label in middle of datum line (default)
	1	// " " " " left of datum line
	2	// " " " " right of datum line
tin_n_height_pre_text	*text*	// pre-text for label - def " "
tin_n_height_post_text	*text*	// post-text for label - def " "

tin_n_height_decimals	*integer*	// number of decimal places to display - def 1
		// If > 0, trailing zeros are removed after
		// the decimal point
		// If < 0, the absolute value is taken as the
		// number of decimal places to report
		// i.e. no trailing zeros are removed

tin_n_height_x	*mm*	// x adjustment to position of label - def 0
tin_n_height_y	*mm*	// y adjustment to position of label - def 0
tin_n_height_angle	*degrees*	// angle of the label - def 0
tin_n_height_colour	*colour*	// colour of the label
tin_n_height_size	*mm*	// size (in mm) of the label
tin_n_height_textstyle	*textstyle*	// textstyle of the label

Labelling the Centre Line Offset and Height

In the centre line case, the value of the height of the **primary string** (usually the design cross section) at the zero offset can be labelled. This is normally where the alignment string cuts the cross section.

The label is made up of the texts:

primary_title offset_title offset_value height_text height_value

NOTE: These parameters have now been superseded. The parameters for labelling the height of the primary string at zero offset covers this case. To use the zero offset parameters instead, the *primary_height_pre_text* would include all text required for the Primary_title, offset_title, offset_value and height_text (offset_value is always 0.0).

The parameter *primary_mode_cl* controls whether the **primary_title** is included in the label.

primary_mode_cl	*0*	// don't include primary_title, primary_value
	1	// incl. primary_title, primary_value default

The parameter *height_mode_cl* controls whether the **height_title** and **height_value** are included in the label.

height_mode_cl	*0*	// don't include height_title, height_value
	1	// include height_title, height_value default

The parameter *offset_mode_cl* controls whether the **offset_title** and **offset_value** are included in the label.

offset_mode_cl	*0*	// don't include offset_title, offset_value
	1	// include offset_title, offset_value default

The label can be placed on the left, centre of right side of the datum line.

offset_height_side_cl	***0***	// text in middle of datum line (default)
	1	// " " " " left of datum line
	2	// " " " " right of datum line

The size, colour, text justification and text style for the label is given by:

primary_title	*text*	//primary title
offset_title	*text*	// offset label (default "Offset")
height_text	*text*	// height text label (default "Height")
primary_textstyle	*textstyle*	// textstyle used for offset and height // label
primary_colour	colour	//colour for offset and height label
primary_size	mm	//size of the offset and height label

offset_height_text_justification_cl	*justif*	// text justification
		// NOTE - this is not normally required since
		// by default the text justification is set to
		// match offset_height_side_cl

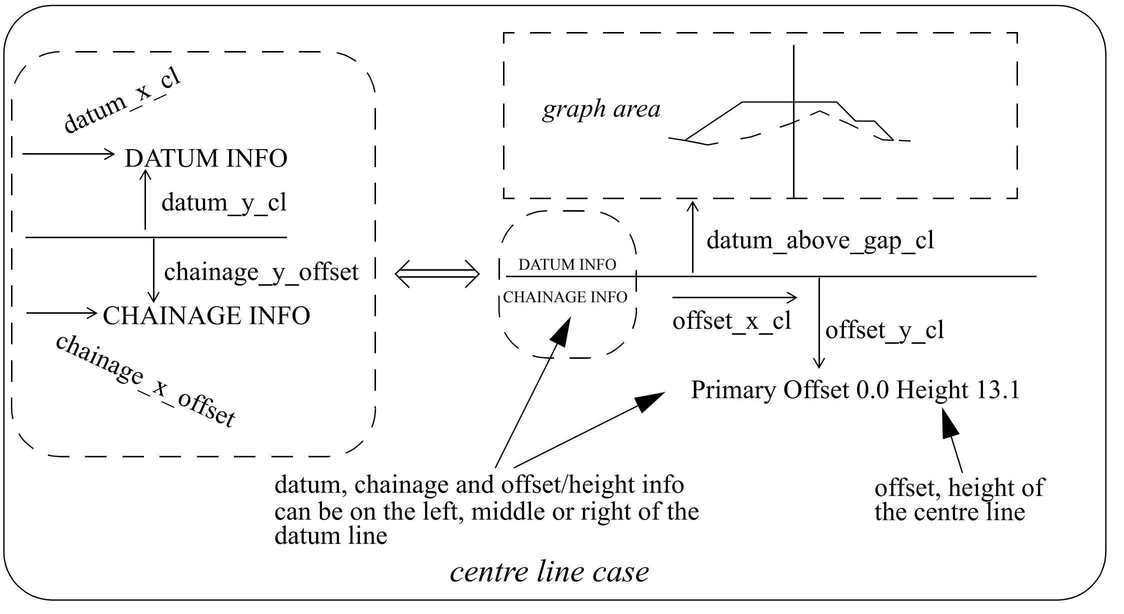
The number of decimals in the offset-value and height-value is controlled by:

number_of_decimals	*integer*	// number of decimal places in the offset,
		// height boxes. If <0, the absolute value
		// is taken as the number of decimal
		// places i.e. no trailing zeros are
		// removed for the values in the offset,
		// heights area.

The label is placed an distance offset_x_cl along the datum line and a distance ***offset_y_cl*** below the datum line of the plot.

offset_x_cl	*mm*	// distance to move the text along the // datum line
offset_y_cl	*mm*	// distance to shift the text below the // datum line

The *offset_y_cl* is measured from the datum line with positive being **down**.



Please continue to the next section 44.2.4 Boxes Case for the boxes parameters.

Please continue to the section 44.2.6 Graph Area if you are not using boxes parameters.

44.2.4 Boxes Case

Datum Line

For the boxes case, the **datum line** is positioned the distance ***datum_below_gap*** above the top of the boxes area and the graph area is then positioned the distance ***datum_above_gap*** above the datum line.

Hence the graph area is distance (datum_below_gap + datum_above_gap) above the top of the boxes area.

datum_above_gap	*mm*	// dist from datum line to bottom of the // graph area
datum_below_gap	*mm*	// dist from datum line to the top of the // boxes
datum_linestyle	*linestyle*	//linestyle for the datum line

The *datum_below_gap* and *datum_above_gap* can be zero or positive.

The roundoff for the datum value is specified by the user (default 1.0) and the datum is automatically calculated for each sub-plot, and labelled.

datum_roundoff	*1.0*	// value to roundoff the datum value to // e.g. 0.5, 0.2, 1.0 (default 1.0)

datum_decimals	*integer*	// number of decimal places to display
		// the datum value (default 1).
		// If > 0, trailing zeros are removed after
		// the decimal point.
		// If <0, the absolute value is taken as the
		// number of decimal places to report
		// i.e. no trailing zeros are removed

datum_name	*text*	// text to write before the datum value
datum_textstyle	*text*	// textstyle for datum information
datum_text_size	*mm*	// size of datum text and value
datum_colour	*colour*	// colour of the datum text
datum_line_colour	*colour*	// colour of the datum line
datum_x	*mm*	// distance to move the datum text // along the datum line

datum_y	*mm*	// distance to raise the datum text
		// above the datum line (used to be
		// called datum_offset)

The *datum_x* and *datum_y* can be positive, zero or negative.

Labelling the Centreline Chainage

Each x-section sub-plot can be labelled with the centreline chainage of the x-section string.

This CHAINAGE INFO label is made up of the text strings:

"chainage_title" followed by the *chainage-value*

and is drawn under the boxes area.

The parameters controlling the labelling are:

chainage_label	0/1	// 1 = label sub-plot with centreline // chainage, 0 don't label.
chainage_title	*text*	// text before the chainage value

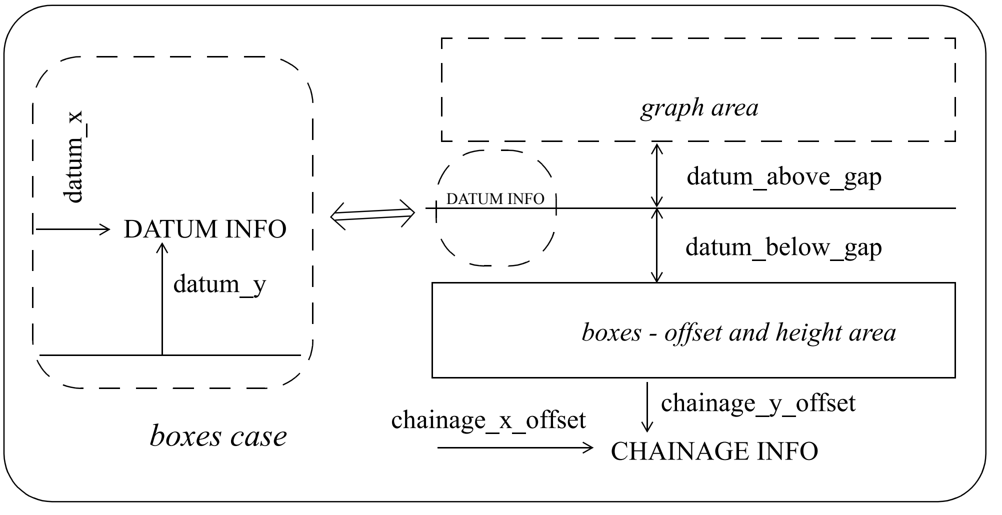
chainage_decimals	*integer*	// number of decimals in the chainage
		// value. If <0, the absolute value
		// is taken as the number of decimal
		// places i.e. no trailing zeros are
		// removed for the values in the

```
                                                    // chainage values.
chainage_size                    mm                 // size of the text
chainage_textstyle               colour             // textstyle for the chainage label
chainage_text_justification      just               // justification of the chainage text
chainage_colour                  colour             // colour of the text
chainage_x_offset                mm                 // x position of text
chainage_y_offset                mm                 // y position of text
```

The *chainage_x_offset* is measured from the beginning of the height boxes.

If *chainage_x_offset* is omitted, the text is centred on heights area.

The *chainage_y_offset* is measured from the bottom of the box area with positive being **down**.



44.2.5 Defining Boxes

When **boxes** is selected for ***label_type***, the primary string (usually the design cross section) and each tin in the x-section sub-plot can be labelled with one or two lines of title, and the height at the offset position for each point in the primary string.

The **title** for the strings, is drawn in the **title area** of the **boxes area**.

The offsets/heights are drawn in the **heights area** of the **boxes area**.

Consequently the boxes area is made up of rows of text consisting of:

string/tin titles followed by the **offset/height values** across the string/tin.

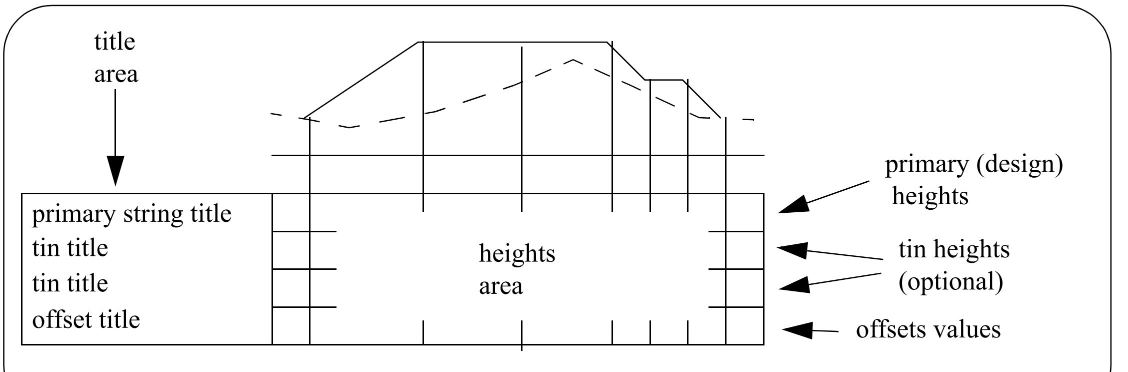
Each row is surrounded by lines to form a box.

The default order of the boxes from the bottom up is

(a) offset title and values

(b) tin title and heights - natural surface etc. (optional)

(c) primary string title and heights - design x-section (optional)



boxes cross section plot

The **title area** starts at the relative position (left_sub_plot_gap,bottom_sub_plot_gap).

The size of the title text is given by the ***title_box_text_size*** parameter.

The width of the title area is either given by the *space_for_titles* parameter, or if omitted, the required width is automatically calculated.

title_box_text_size	*mm*	// size of the titles in the boxes
space_for_titles	*mm*	// calculated if omitted

There can be two lines of title text and the title text, textstyle and colour can be set independently for the primary string and each tin.

The x position of the title text is the same for all the lines of title text and can be set to be a fixed distance from the left hand side of the boxes.

box_titles_x	*mm*	// distance to move the title text from the // left hand side of the boxes

The **heights area** starts at the end of the title area.

The height text is written at right angles to the bottom of the boxes. It can be either top or bottom justified with respect to the box (*box_text_justification*).

The number of decimal places and the size of the heights text can also be specified.

The height of each individual box area is either given by the *horizontal_line_spacing* parameter, or if omitted, the required height is automatically calculated.

number_of_decimals	*integer*	// number of decimal places in the // height boxes. If <0, the absolute value

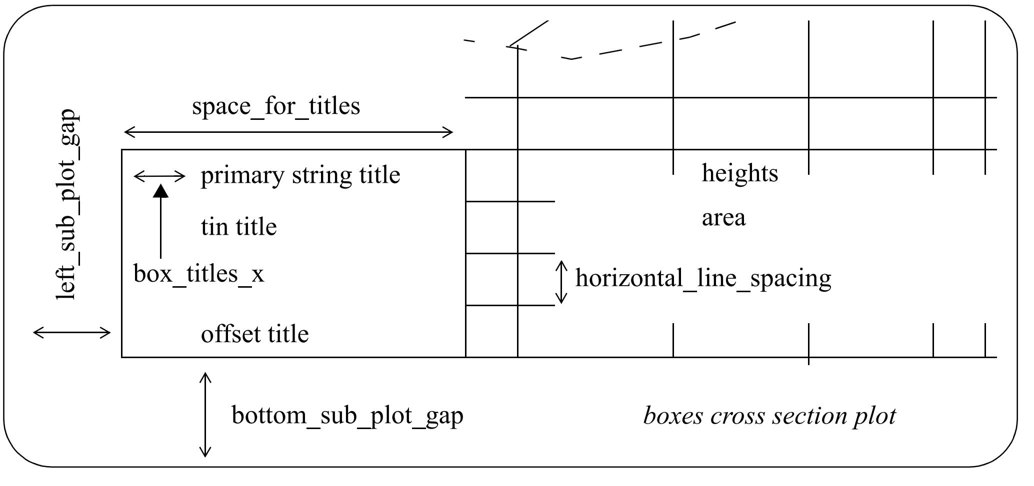
		// is taken as the number of decimal
		// places i.e. no trailing zeros are
		// removed for the values in the heights
		// area.
text_size	*mm*	// size of the height values
offset_textstyle	*textstyle*	// textstyle for height text
offset_colour	*colour*	// colour of height text
box_text_justification	0	// bottom of individual boxes
	1	// top of individual boxes
horizontal_line_spacing	*mm*	// calculated if omitted

The total height of the boxes area is simply given by number of boxes drawn multiplied by the height of one box (they all have the same height).

The **width** of the heights area is determined by the number of chainages to be labelled and whether the values are staggered to prevent over writing.

Hence the total width of the boxes area is the width of the labels area plus the width of the heights area.

Many distance definitions in the plot parameter file are given in terms of distance above the **top** of the boxes area.



The text in the title area is the same for each cross section plot on the sheet so it is possible to restrict the title area to be only on the first cross section on the sheet or the first column of cross sections.

label_first_only	0	//title area on all x-sections (default)
	1	// title area on first x-sect of page only
	2	// title area on x-sects in first column

The drawing of the box line work is user defined. This has been extended from V3.1 although *draw_box_mode* has been left in for upward compatibility.

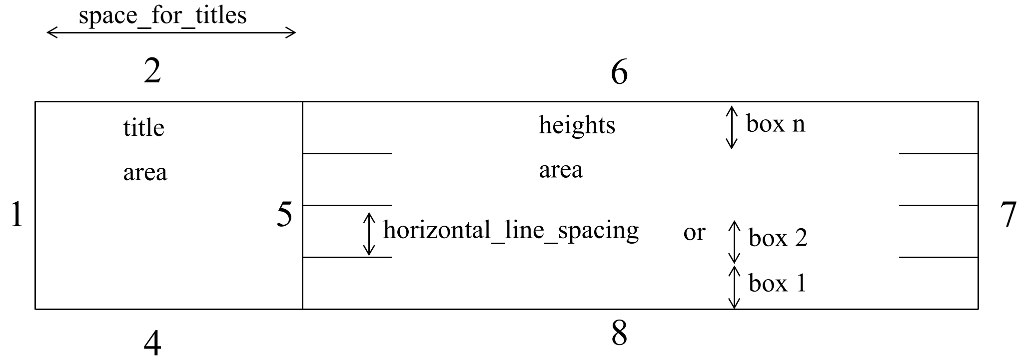
draw_box_mode	*0*	// don't draw box lines
	1	// draw box lines around heights area
		// only
	2	// draw box lines around title and
		// heights areas
	3	// use draw_box_side_n,
		// box_line_draw_mode and
		// box_line_mode_n

box_colour	*colour*	// colour of the boxes

For V3.2, the line work for the outside of the title and heights boxes is controlled by the parameters *draw_box_side_n* and *box_side_colour_n*, and the separation lines inside the boxes are controlled by the parameters *box_line_draw_mode* and *box_line_mode_n* where the box numbering, n, starts from the bottom box.

Note: The following parameters are only used if ***draw_box_mode*** is set to **3**.

draw_box_side_1	***1***	// draw the left side of the title area (def)
	0	// don't draw the left side
box_side_colour_1	*colour*	// colour to draw left side of title area
		// default box_colour
draw_box_side_2	***1***	// draw top of the title area (default)
	0	// don't draw the top
box_side_colour_2	*colour*	// colour to draw top of title area
		// default box_colour
draw_box_side_4	***1***	// draw bottom of the title area (default)
	0	// don't draw the bottom
box_side_colour_4	*colour*	// colour to draw bottom of title area
		// default box_colour
draw_box_side_5	***1***	// draw left side of the heights area (def)
	0	// don't draw the left side
box_side_colour_5	*colour*	// colour to draw left side of heights area
		// default box_colour
draw_box_side_6	***1***	// draw top of the heights area (default)
	0	// don't draw the top
box_side_colour_6	*colour*	// colour to draw top of heights area
		// default box_colour
draw_box_side_7	***1***	// draw right side of the heights area def
	0	// don't draw the right side
box_side_colour_7	*colour*	// colour for right side of heights area
		// default box_colour
draw_box_side_8	***1***	// draw bottom of the heights area (def)
	0	// don't draw the bottom
box_side_colour_8	*colour*	// colour for bottom of heights area
		// default box_colour



lines to draw for draw_box_side_1 to draw_box_side_8

The lines at the top of the individual boxes inside the title area and heights area (separation lines) are controlled by the parameters *box_line_draw_mode* and *box_line_mode_n*.

The separation lines can be drawn just in the title area, just in the heights area or in both areas.

The parameter *box_line_draw_mode* can be set to control all the separation lines but there are additional parameters, *box_line_mode_n*, which override *box_line_draw_mode* for each of the individual boxes where n = 1, ... number of boxes -1.

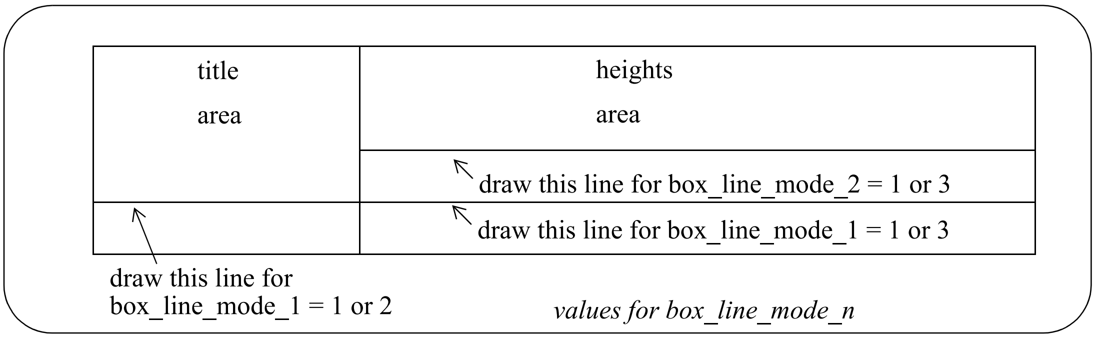
The top of the top box is not controlled by box_line_mode_n but is controlled by the parameters *draw_box_side_2* and *draw_box_side_6.*

box_line_draw_mode	*0*	// don't draw any separation lines inside // the title and heights areas
	1	// draw the separation lines inside the // title and heights areas (default)
	2	// draw the separation lines inside the // title area only
	3	// draw the separation lines inside the // heights area only

box_line_mode_n	*0*	// for the nth box, don't draw any
		// line at the top of the box for either
		// the title or the heights areas

	1	// for the nth box, draw the line at the
		// top of the box for both the
		// title and heights areas

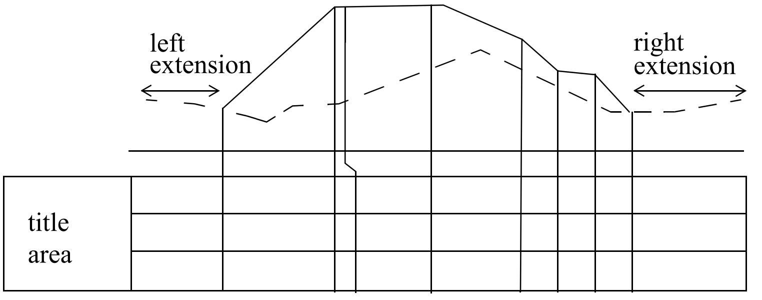
	2	// for the nth box, draw the line at the // top of the box for the title area only
	3	// for the nth box, draw the line at the // top of the box for the heights area only



values for box_line_mode_n

Finally, the right hand end of the boxes can stop at the end of the design x-section or extend to the end of the *right_extension* distance.

box_extension_mode	*0*	// default - stop right end of boxes at the // end of design x-section
	1	// extend right end of boxes to the end of // the right_extension distance.



if box_extension_mode = 1 then the boxes extend to the end of the right extension

Offset Titles and Values

If the ***label_type*** is set to boxes, the offsets of the points across the primary string (usually the design cross section) are always labelled.

offset_title	*text*	// first line of offset title (def **offsets**)
offset_title_2	*text*	// second line of offset title
offset_title_textstyle	*textstyle*	// textstyle for offset title
offset_title_colour	*colour*	// offset title text colour
offset_title_size	*mm*	// offset title text size
offset_colour	*colour*	// colour of offset values
offset_textstyle	*textstyle*	// textstyle for offset values
offset_size	*mm*	// size for offset values

Example of Offset Titles Parameters

// offset titles and values

offset_title	“Offset”
offset_title_2	“metres”
offset_title_colour	grey

Primary String Titles and Heights

Sometimes the primary string (the design cross section) is only required to define the offset positions to label and the section line for the tin sections. In this case, the primary string would not be drawn on the cross section plot.

The drawing or not drawing of the primary string on each cross section plot is controlled by the parameter *primary_string*.

If the primary string is drawn, by default a row of labelling is included in the boxes area with up to two lines of title information and heights for each point across the cross section.

primary_string	*yes*	// draw and label the primary string
	no	// don't draw or label the primary string
primary_title	*text*	// first line of string title
primary_title_2	*text*	// second line of string title
primary_title_textstyle	*textstyle*	// textstyle for primary string titles
primary_title_colour	*colour*	// colour of the string titles
primary_title_size	*mm*	// primary title text size
primary_textstyle	*textstyle*	// textstyle for primary string heights

primary_title_y_pos	*mm*	// if set, the height in mm above the
		// bottom of all the boxes that the
		// primary title text is drawn.
		// If not set, then the text is placed at a
		// height that puts it inside the default
		// box for the primary title.

primary_colour	*colour*	// colour of the primary string heights
primary_size	*mm*	// primary text size

primary_y_pos	*mm*	// if set, the height in mm above the
		// bottom of all the boxes that the
		// primary height text is drawn.
		// If not set, then the text is placed at a
		// height that puts it inside the default
		// box for the primary height text.

primary_decimals	*integer*	// number of dec places in primary height // < 0 to keep all trailing zeros

Example of Labelling Primary String Titles

```
 primary_title              "Design"
// primary_title_2           "primary2"      // not being used - commented out
 primary_colour             "purple"
 primary_title_colour       "red"
```

Tin Titles, Heights and Depths

A section along the primary string through each tin on the section view is automatically drawn on the cross section plot, but the user can specify whether the tin heights and depths are labelled or not in the boxes area.

Hence although there may be a number of tins drawn on the section plot, not all of them need to be labelled or have their depths labelled.

The default order for labelling the tins and depths is the order that they were added to the view but it is possible to specify which tin is used for labelling each row of tin heights and depths by giving the tin name rather than just using the tin order on the section view.

In fact, it is possible to use **any tin** in the project **to label** a row, **not just those drawn** on the section view.

tin_n_name	*text*	// n=1, ... no of tins on the section view.
		// use the tin called *text* to label the nth
		// row of tin heights.

If a tin of the name given by ***tin_n_name* does** not exist, then the plot is not produced and an error message is given.

For the following parameters, n takes the value 1 to 100 and specifies that the parameter set applies to the *n*th tin on the section view or if *tin_n_name* is set, by the tin specified by *tin_n_name*.

tin_n_draw_mode	***1***	// draw the nth tin (default))
	0	// don't draw the nth tin
tin_n_draw_colour	*colour*	// colour to draw the nth tin // if omitted then use the tin colour
tin_n_label	***1***	// label the nth tin (default)
	0	// don't label the nth tin
tin_n_title	*text*	// first line of nth tin title
tin_n_title_2	*text*	// second line of nth tin title
tin_n_title_textstyle	*textstyle*	// textstyle for nth tin titles
tin_n_title_colour	*colour*	// colour for nth tin titles
tin_n_title_size	*mm*	// size for nth tin titles

tin_n_title_y_pos	*mm*	// if set, the height in mm above the
		// bottom of all the boxes that the
		// tin title text is drawn.
		// If not set, then the text is placed at a
		// height that puts it inside the default
		// box for the tin heights.

tin_n_decimals	*integer*	// number of dec places in tin height // < 0 to keep all trailing zeros
tin_n_textstyle	*textstyle*	// textstyle of the tin heights
tin_n_colour	*colour*	// colour for heights **and** depths
tin_n_size	*mm*	// default nth tin's title size

tin_n_y_pos	*mm*	//if set, the height in mm above the
		//bottom of all the boxes that the
		// tin height text is drawn.
		//If not set, then the text is placed at a
		//height that puts it inside the default
		// box for the tin heights.

tin_n_depth_label	1/0	// 1 to label, 0 don't label depths
tin_n_depth_title	*text*	// first line of tin depth title
tin_n_depth_title_2	*text*	// second line of tin depth title
tin_n_depth_title_textstyle	*textstyle*	// textstyle of the tin depth titles

tin_n_depth_title_colour	*colour*	// colour for titles of depths
tin_n_depth_title_size	*mm*	// size for title of depths

tin_n_depth_title_y_pos	*mm*	// if set, the height in mm above the
		// bottom of all the boxes that the
		// tin depth title text is drawn.
		// If not set, then the text is placed at a
		// height that puts it inside the default
		// box for the tin heights.

tin_n_depth_decimals	*integer*	// number of dec places in tin height // < 0 to keep all trailing zeros
tin_n_depth_textstyle	*textstyle*	// textstyle of the tin depths
tin_n_depth_colour	*colour*	// colour for depths
tin_n_depth_size	*mm*	// size for depths

tin_n_depth_y_pos	*mm*	// if set, the height in mm above the
		// bottom of all the boxes that the
		// tin depth text is drawn.
		// If not set, then the text is placed at a
		// height that puts it inside the default
		// box for the tin heights.

Definition of Depth to a Tin or to Offset Strings

The depth from the primary string to a tin, at a particular offset is defined as

depth = tin height value - height of the primary string

That is, the depth that the primary string is **below** the tin.

Before plotting, the value of depth is multiplied by either the *depth_positive_factor* or *depth_negative_factor*.

if (depth >= 0) plotted_depth_value = depth * depth_positive_factor

if (depth < 0) plotted_depth_value = depth * depth_negative_factor

Hence the definition of depth can be modified by the parameters:

depth_positive_factor	*value*	// multiplier for positive depths
depth_negative_factor	*value*	// multiplier for negative depths

For example, if the opposite sign is required for depth, that is,

depth = height of the primary string - tin height value

simply set

depth_positive_factor -1

depth_negative_factor- 1

Example of Labelling Tins

```
//    tin label 1 for on
//              0 for off

 tin_1_label              1

 tin_1_title             “Natural”
// tin_1_title_2         “Surface”
 tin_1_colour             grey
 tin_1_title_colour       red
```

tin_2_label	1
tin_2_title	"Water table"
tin_2_colour	grey
tin_2_title_colour	red

Labelling the Design Height, X and Y Co-ordinates and Tin Heights at Offset Zero

The values of the height and X and Y co-ordinates of the **primary string** (usually the design cross section) at the zero offset can be labelled. Note that zero offset is normally where the alignment string cuts the cross section.

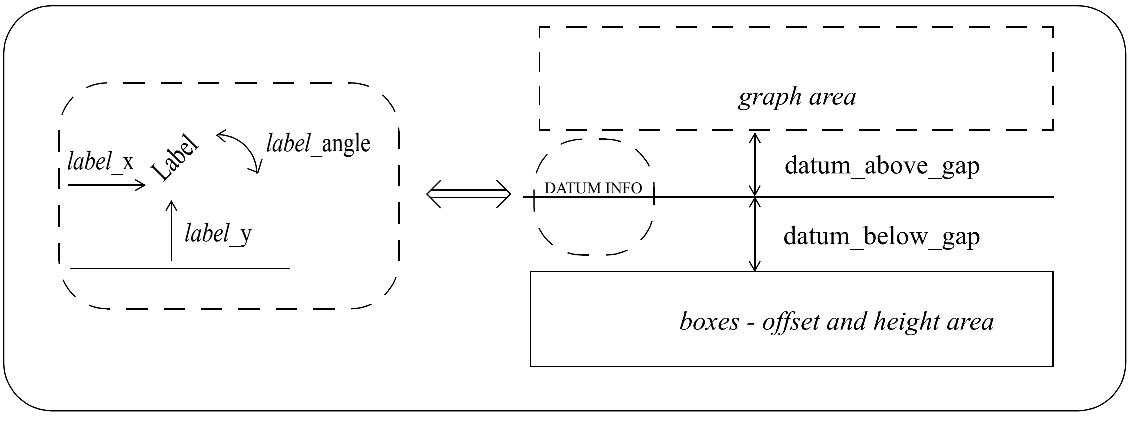
The heights of any tins (such as the natural surface) at the zero offset can also be labelled.

The labels are made up of:

pre_text *value* post_text

where *value* is either a height or a co-ordinate.

The label is positioned at either the left, right or middle of the datum line, with an x and y adjustment and a rotation.



Parameters for labelling the X Co-ordinate at Zero Offset:

primary_x0_draw_mode	*0*	// don't draw the label -default
	1	// draw the label
primary_x0_pre_text	*text*	// pre-text for label - def " "
primary_x0_post_text	*text*	// post-text for label - def " "
primary_x0_decimals	*integer*	// number of decimal places to display - def 1
		// If > 0, trailing zeros are removed after
		// the decimal point
		// If < 0, the absolute value is taken as the
		// number of decimal places to report
		// i.e. no trailing zeros are removed
primary_x0_x	*mm*	// x adjustment to position of label - def 0
primary_x0_y	*mm*	// y adjustment to position of label - def 0
primary_x0_angle	*degrees*	// angle of the label - def 0
primary_x0_colour	*colour*	// colour of the label
primary_x0_size	*mm*	// size (in mm) of the label
primary_x0_textstyle	*textstyle*	// textstyle of the label
primary_x0_justify	*just*	// justification for text

Parameters for Labelling the Y Co-ordinate at Zero Offset:

primary_y0_draw_mode	***0***	// don't draw the label -default
	1	// draw the label

primary_y0_pre_text	*text*	// pre-text for label - def " "
primary_y0_post_text	*text*	// post-text for label - def " "
primary_y0_decimals	*integer*	// number of decimal places to display - def 1
		// If > 0, trailing zeros are removed after
		// the decimal point
		// If < 0, the absolute value is taken as the
		// number of decimal places to report
		// i.e. no trailing zeros are removed
primary_y0_x	*mm*	// x adjustment to position of label - def 0
primary_y0_y	*mm*	// y adjustment to position of label - def 0
primary_y0_angle	*degrees*	// angle of the label - def 0
primary_y0_colour	*colour*	// colour of the label
primary_y0_size	*mm*	// size (in mm) of the label
primary_y0_textstyle	*textstyle*	// textstyle of the label
primary_y0_justify	*just*	// justification for text

Parameters for Labelling the Height of Primary String at Zero Offset:

primary_height_draw_mode	*0*	// don't draw the label -default
	1	// draw the label
primary_height_pre_text	*text*	// pre-text for label - def " "
primary_height_post_text	*text*	// post-text for label - def " "
primary_height_decimals	*integer*	// number of decimal places to display - def 1
		// If > 0, trailing zeros are removed after
		// the decimal point
		// If < 0, the absolute value is taken as the
		// number of decimal places to report
		// i.e. no trailing zeros are removed
primary_height_x	*mm*	// x adjustment to position of label - def 0
primary_height_y	*mm*	// y adjustment to position of label - def 0
primary_height_angle	*degrees*	// angle of the label - def 0
primary_height_colour	*colour*	// colour of the label
primary_height_size	*mm*	// size (in mm) of the label
primary_height_textstyle	*textstyle*	// textstyle of the label
primary_height_justify	*just*	// justification for text

Parameters for Placing Some Text:

extra_text_draw_mode	***0***	// don't draw the label -default
	1	// draw the label
extra_text_pre_text	*text*	// pre-text for label - def " "
extra_text_post_text	*text*	// post-text for label - def " "
extra_text_decimals	*integer*	// number of decimal places to display - def 1
		// If > 0, trailing zeros are removed after
		// the decimal point
		// If < 0, the absolute value is taken as the
		// number of decimal places to report
		// i.e. no trailing zeros are removed
extra_text_x	*mm*	// x adjustment to position of label - def 0
extra_text_y	*mm*	// y adjustment to position of label - def 0
extra_text_angle	*degrees*	// angle of the label - def 0
extra_text_colour	*colour*	// colour of the label
extra_text_size	*mm*	// size (in mm) of the label
extra_text_textstyle	*textstyle*	// textstyle of the label
extra_text_justify	*just*	// justification for text

Offset Selection for Uprights and Staggering of Heights

For each sub-plot, the offsets of the points across the x-section string (primary string) are used for positioning uprights (leader lines), and the offset and height labels for the uprights.

By default, there is an upright at each point across the primary string.

However, if the primary string is a 4d string, then the text at the points on the 4d string can be used as a key to **suppress** the labelling and upright at that point.

The parameters to **stop** labelling and uprights are:

mask_name_n	*4d_string_point_text*	where n=1,100

Note -the text *4d_string_point_text* can include wild cards (*) and characters (?).

After any *name masks* have been applied, it is often desirable to weed out offset values that are too close together before doing any labelling.

offset_label_tolerance	*value*	// weed out offset values closer together // (in offset units) than this value

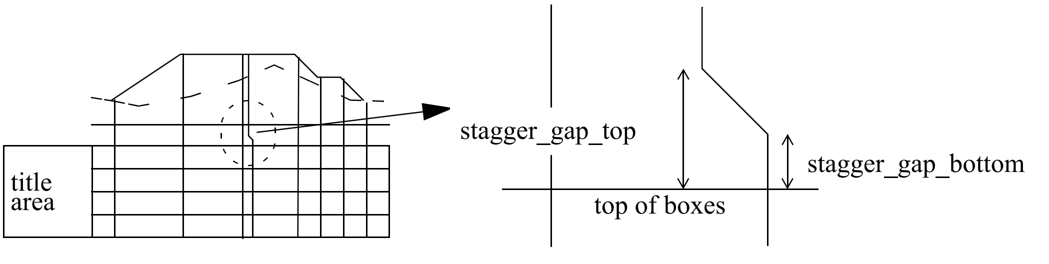
If the real offset position is used for the horizontal position of the offset/height text, text over writing can easily occur. To prevent over writing, the text is automatically **staggered**.

When staggering occurs, the real offset position is then indicated by the offset markers which are drawn at the top of the text boxes from the staggered text position back to the actual offset position of the text.

The size and position of the staggers are given by:

stagger_gap_top	*mm*	// distance from boxes to top of stagger
stagger_gap_bottom	*mm*	// distance from boxes to bottom of // stagger
stagger_gap_factor	*value*	// distance between staggers is // box_text_size * stagger_gap_factor

When staggering occurs, it is possible for the heights area to be wider than the graph area.



boxes cross section plot

Uprights

Uprights (leader lines) can be drawn from the top of the staggers to strings drawn on the plot.

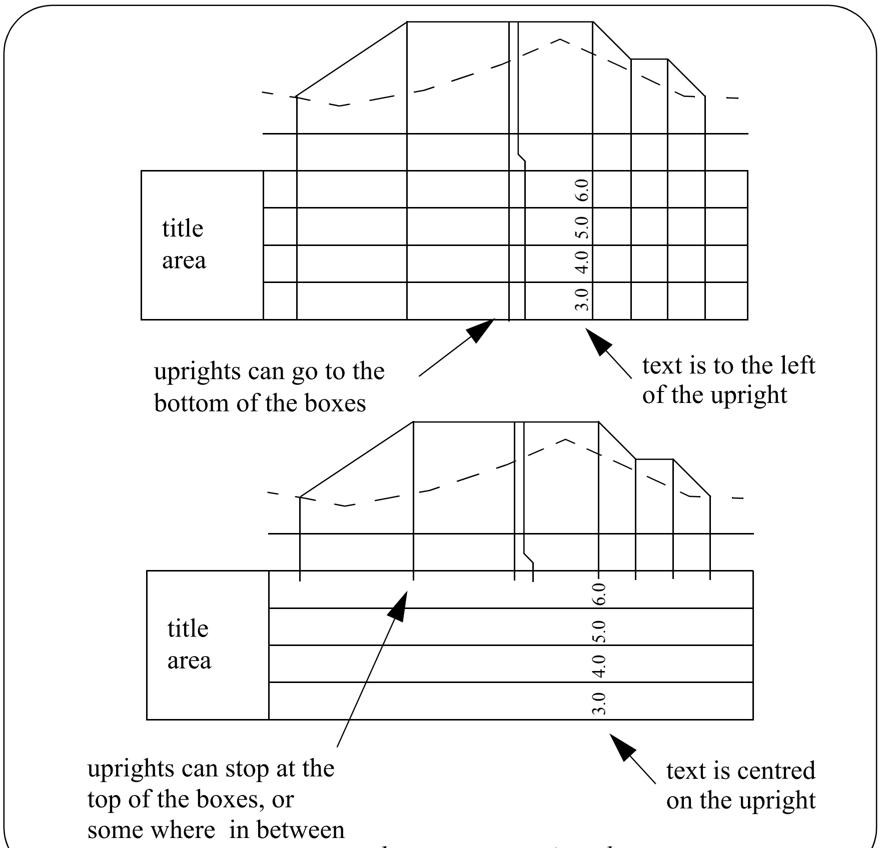
The height of the uprights is given by the ***uprights_draw_mode***:

uprights_draw_mode	0	// none
	1	// to maximum string height at that // chainage
	2	// ticks, to stagger height
	3	// to uprights_y above the boxes
	100	// to the primary string
	101-500	// to tin1 or tin2 etc.
uprights_y	*mm*	// distance to draw the uprights for // mode 3
uprights_colour	*colour*	// uprights colour (def boxes colour)

The uprights can go below the top of the boxes.

uprights_bottom_mode	*0*	// stop at top of boxes (default)
	1	// draw to bottom of boxes
	2	// draw to uprights_bottom_y below the // top of the boxes
	3	// draw to uprights_bottom_y above the // bottom of the boxes
	4	// ticks at chainage
uprights_bottom_y	*mm*	//distance
uprights_text_offset_factor	*value*	// move the text by this factor*size

When uprights go below the top of the boxes, the height and offset text is moved to the left so that the upright does not go through the text. The left hand side of the heights boxes also moves to the left to leave room for the height text.



boxes cross section plot

Please continue to the next section 44.2.6 Graph Area.

44.2.6 Graph Area

The **graph area** for each section sub-plot is the area where the actual plot of the x-section string is drawn.

The **width** of the graph area is determined by the width of the x-section string being plotted, the left and right extensions and horizontal scale (scale) given by parameters or in the **section x plot** panel, and the vertical exaggeration given by the section view or a parameter.

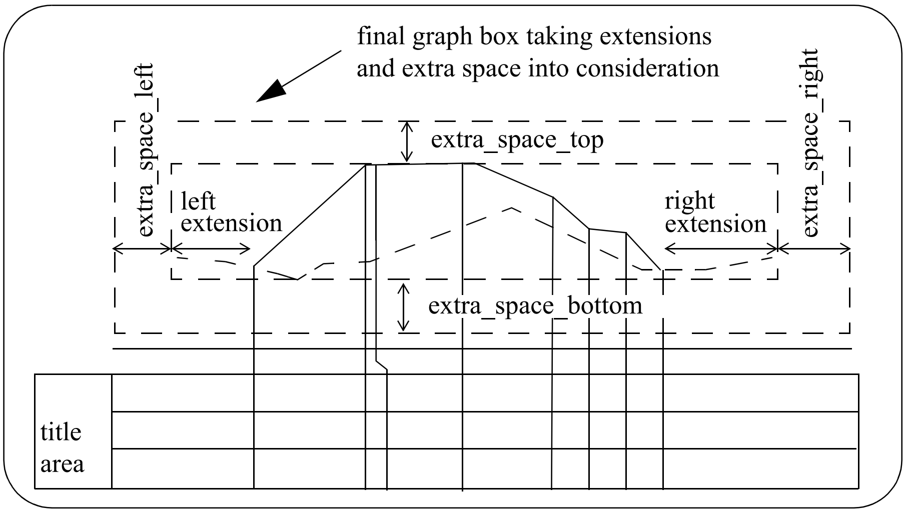
view_name	*text*	// default is **section view** in panel.
vertical_exaggeration	*value*	// default is vertical exag for view
scale	*value*	// 1:value - horizontal scale, default is
		// **scale 1:** in panel.
left_extension	*world-units*	// left extension value
right_extension	*world-units*	// right extension value

The vertical scale is determined by the horizontal scale and the vertical exaggeration.

The **height** of the graph area is determined by the vertical scale (given by the horizontal scale and the vertical exaggeration) and the minimum and maximum values of the data being plotted. Hence the graph height is a calculated rather than a given value.

The size of the graph area can also be extended to allow for symbols by the parameters:

extra_space_units	***0***	// extra space values are world units - default
	1	// " " " are millimetres on plot page
extra_space_left	*units*	// subtract from left of plot area, def 0
extra_space_right	*units*	// add to right of plot area, default 0
extra_space_top	*units*	// add to top of plot area, default 0
extra_space_bottom	*units*	// subtract from bottom of plot area,
		// default 0



The types of strings that are drawn in the graph area of a cross section plot are:

(a) primary string the x-section string from the x-section model that is being drawn. Called the primary string and is usually the design x-section.

(b) tins sections of the primary string through any tins either in models on the section view or in corridor models.

(c) services parts of strings (from either models on the section view or in corridor models) that cut the defined corridor.

The **colour** of the strings in the plot is the actual string colour for cases (a) and (c), and the colour of the tin used for the section in case (b).

Although all the strings are plotted, the plot parameter file can be used to select which ones are labelled with heights.

Primary String (Design Cross Section)

The primary string (the design cross section) is used to define

(a) the design cross section
(b) the chainage of the section
(c) the offset positions for labelling heights and drawing uprights
(d) the section line used for sectioning through tins
(e) the section line for defining the corridor for services

Although the primary string is used to set up most of the information for the cross section plot, it doesn't have to be drawn on the cross section.

The drawing or not drawing of the primary string on each cross section plot is controlled by the parameter *primary_string* which was described earlier.

The **colour** of the primary string in the plot is the actual primary string colour.

Tins

A section along the primary string through each tin on the specified section view is automatically drawn on the cross section plot.

The **colour** and **linestyle** of the tin section in the plot is the actual tin colour and tin linestyle.

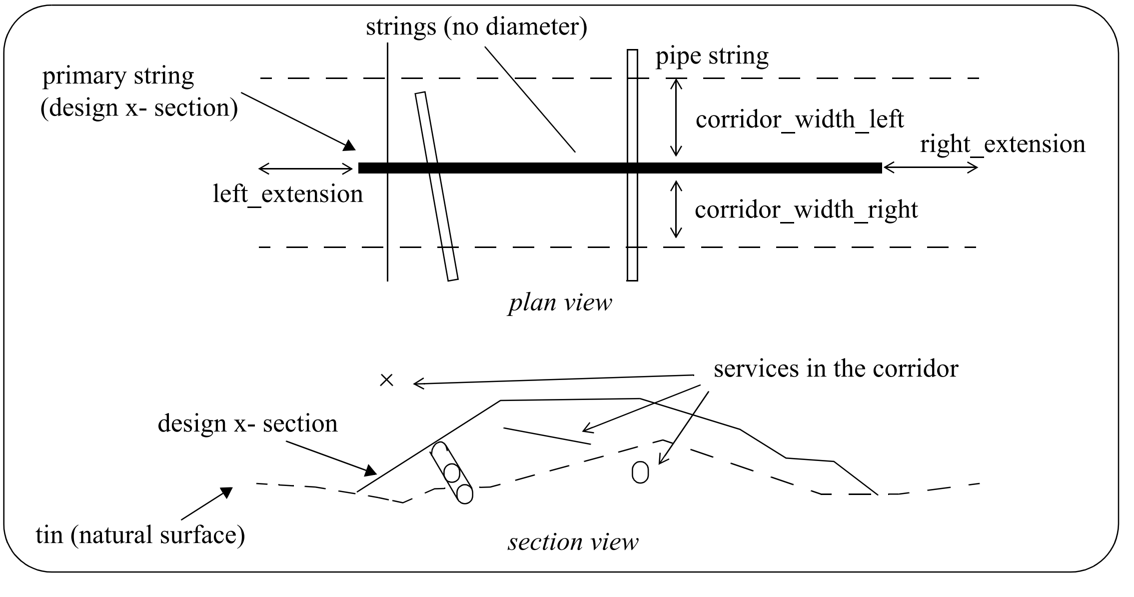
Whether the tin heights are labelled or not in the boxes area is controlled by parameters and has been described earlier

Corridor and Services

A corridor around the primary string is defined by giving a left and right corridor width and a left and right extension.

Any string in a model added to the section view is checked to see if it appears in the corridor, and if it does, then it is drawn on the cross section plot.

Strings do not have to cross the primary string, but just be in the corridor.



The corridor defining parameters are

```
left_extension          world-units     // left extension value
right_extension         world-units     // right extension value
corridor_width_left     world-units     // corridor left and right widths
corridor_width_right    world-units     // defaults are the section view values
corridor_overlap_left   world-units     // corridor left and right overlaps.
corridor_overlap_right  world-units     // defaults are the section view values
corridor_chord_arc      world-units     // chord-arc tolerance used to
                                        // approximate arcs in the corridor.
                                        // default is the section view value
```

The models containing tins for sectioning and strings for services are taken from the section view or given by the parameters

```
corridor_model_n        model           // n = 1, 2 ... 100
                                        // models containing tins and service
                                        // strings to be drawn on the view.
```

If any *corridor_model_n* parameters are defined, then only the models given by the parameters are used. If no *corridor_model_n* parameters are set, then the models added to the given section view are used.

That is, either the *corridor_model_n* parameters are used or if none exist, then the models added to the section view are used for tins and service strings.

The **colour** and **diameter** of the service strings drawn in the plot are the actual strings colour and diameter.

Please continue to the next section 44.2.7 Grade Labelling.

44.2.7 Grade Labelling

The plot of the x-section string is made up of straight lines joining the individual points of the x-section.

The **grades** of these lines can be labelled:

grade_label	0	// don't label grades (default)
	1	// label grades

Either the individual lines can be used as the segments to be labelled for grade, or adjacent lines of the same grade can be considered to be just one segment and labelled only once.

Hence the segments to be labelled for grade can be the individual lines of the x-section, or the segments defined by changes of grade.

grade_change_only	0	// label individual lines
	1	// label change of grade segments

It is also possible to ignore segments smaller that a given minimum width on the plot.

grade_minimum_width	*mm*	// segments smaller are not labelled

The grade labels are drawn parallel to the segment, centred about the segments end points, and a distance *grade_offset* above the segment. The size, colour and number of decimal places can all be set.

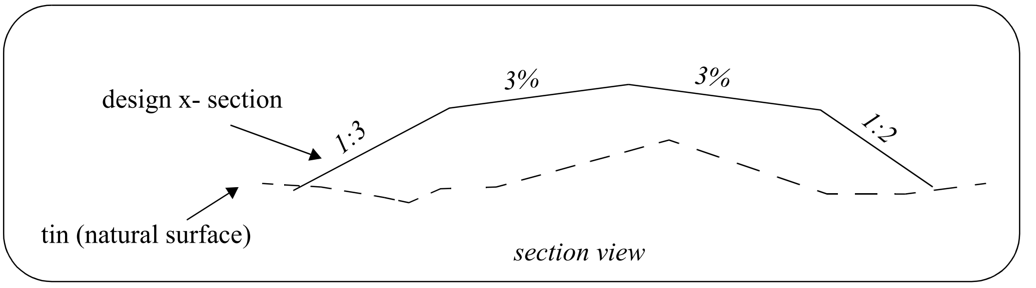
grade_decimals	*integer*	// number of decimal places in grade
grade_size	*mm*	// size of the grade label
grade_textstyle	*text*	// textstyle of the grade label
grade_colour	*colour*	// colour of the grade label
grade_offset	*mm*	// distance above the segment for label
grade_signed	*yes*	// default - grades are signed
	no	// absolute value of grades

The grade can be labelled as percent cross-fall, 1 in slope, m/m or VicRoads x:1.

Also a threshold value can be set and any grades whose absolute value are below the threshold can be labelled in one way, and those above the threshold labelled a different way.

Hence, if the absolute value of the grade is less that or equal to the absolute value of *grade_threshold*, then ***grade_mode*** is used, otherwise ***grade_upper_mode*** is used.

grade_mode	0	// % grade
	1	// 1 in
	2	// m/m
	3	// VicRoads x:1
grade_threshold	value	// grade threshold for type of grade
		// labelling
grade_threshold_mode	0-3	// type of grade value given in
		// grade_threshold types are the same as
		// grade_mode
grade_upper_mode	0	// % grade
	1	// 1 in
	2	// m/m
	3	// VicRoads x:1
grade_upper_decimals	*integer*	// number dec places in grades above
		// threshold value



Example of Grade Labelling

```
grade_label                    1                  // turn grade labels on
grade_mode                     0                  // percent grade
grade_decimals                 1                  // one decimal place, no trailing zeros
grade_change_only              1                  // only label when change of grade

// text size for grade annotations

grade_size                     2
grade_colour                   yellow

// how far the annotation is (perpendicularly)
//    from the string itself

grade_offset                   1

//  label all annotations separated by more than
//  the minimum width in plot units

grade_minimum_width            0

//  threshold and threshold mode describe the value
//   at which the grade labelling changes
//   mode to grade_upper_mode

grade_threshold                7                  // change labelling when over 7%
grade_threshold_mode           0                  // units for threshold - %
grade_upper_mode               1                  // label as 1:x
```

Please continue to the next section 44.2.8 Labelling Points of the X-Sections.

44.2.8 Labelling Points of the X-Sections

The points across each x-section can be automatically labelled on the x-section plots.

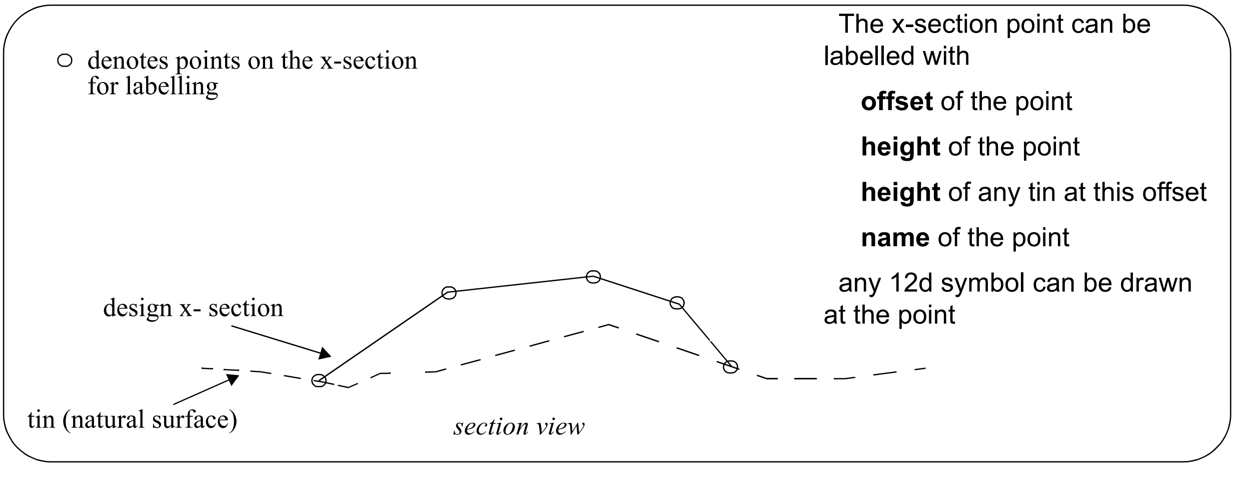
The **height**, **offset** and **name** of the point can be labelled as well as a **symbol** drawn. The height of tins at the same offset value can also be labelled.

The offset position for the labelling is the offset of the point.

The height position for the labelling can be specified as the

(a) top of the boxes for the x-section
(b) above the maximum height of the strings on the plot
(c) height of the point on the x-section string (primary string)
(d) height of a tin.

The actual position of the label is defined relative to the above point.



Note:

Only case (b) involves the actual height of the point on the cross section string. For all other cases, only the offset of the cut string is used.

Other heights, for example, the height of the tin at that offset can be used as the height (case (d)).

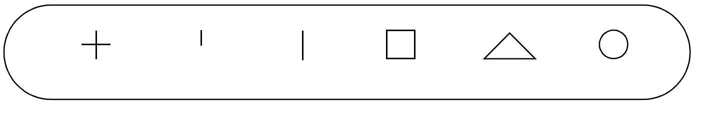
Text justification refers to the actual position and is given by

"top-left"	"top-centre"	"top-right"
"middle-left"	"middle-centre"	"middle-right"
"bottom-left"	"bottom-centre"	"bottom-right"

A choice of six special symbols and/or or any of the 12d symbols can be drawn at the cut point.

The special 12d Model symbols of size one millimetre are drawn in a square box centred on (0,0) with sides of length two millimetres. That is, the box co-ordinates are (-1,-1), (1,1), (1,-1), (-1,-1).

The six special shapes are



Up to twenty five (25) separate sets of points can be labelled.

Parameters for Labelling Points of X-Sections

The points of the x-section to be for labelled for the nth set of parameters is restricted to all the points whose name satisfying the *points_n_mask*:

points_n_mask *point_name*s // points to be labelled

where *point_names* is a text string containing the name masks, each separated by one or more spaces, to test the point name against. Each mast can include wild cards and wild characters.

For example

points_1_mask "ke*"

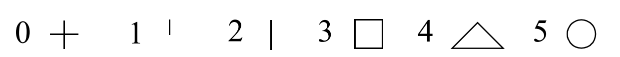
or points_1_mask "?bank*"

or, if both masks are required,

points_1_mask "ke* ?bank*"

The parameters for drawing a **symbol** are

points_symbol_n_mode	0	// cross
	1	// up from centre of box
	2	// up and down from centre of box
	3	// square
	4	// triangle, base at bottom
	5	//circle
	6	// use a 12d symbol



predefined symbols for

If *points_symbol_n_mode* is 6, then the plot symbol is given by

points_symbol_n_style *plotsymbol* // name of the 12d symbol to draw at point

Important Note

The plot symbol of name *plotsymbol* is defined in the file given by:

(a) the parameter *plot_symbols* in the ppf file

plot_symbols *filename*

or if plot_symbols is not defined, then

(b) in the file pointed to by the environment variable PLOT_SYMBOLS_4D

PLOT_SYMBOLS_4D *filename* *// default plotsym.4d*

or if PLOT_SYMBOLS_4D is not defined, then

(c) in the file *plotsym.4d*

which is searched for in the standard set up file sequence

If none of the above files are defined, or if the symbol does not exist in the above files, then it will be searched for in the standard 12d symbols file which is:

(d) either pointed to by the environment variable SYMBOLS_4D

SYMBOLS_4D filename // default symbols.4d

or if the environment variable SYMBOLS_4D does not exist, in the file, *symbols.4d*

The position of the symbol is given by:

points_symbol_n_position	1	// above point height value
	3	// above top of boxes
	4	// above top of graph area but not including
		// the extra_space_top
	100	// to primary string (same as 1)
	101-500	// to tin1 or tin2 etc.

The symbol can be adjusted by the parameters:

points_symbol_n_x	*mm*	// offset adjustment to position
points_symbol_n_y	*mm*	// height adjustment to position
points_symbol_n_angle	*degrees*	// rotation about point
points_symbol_n_colour	*colour*	// colour of symbol

and for all values of *points_symbol_n_mode* **other** than 6:

points_symbol_n_size	*mm*	// size of symbol, 0 don't draw

The value of the **offset** of the point can be labelled using the parameters

points_offset_n_position	1	// above point height value
	3	// above top of boxes
	4	// above top of graph area but not including
		// the extra_space_top
	100	// to primary string
	101-500	// to tin1 or tin2 etc.
points_offset_n_x	*mm*	// offset adjustment to position
points_offset_n_y	*mm*	// height adjustment to position
points_offset_n_angle	*degrees*	// rotation about point
points_offset_n_size	*mm*	// size of text, 0 don't label
points_offset_n_colour	*colour*	// colour of text
points_offset_n_textstyle	*textstyle*	// textstyle to use for points offset text.
points_offset_n_pre_text	*text*	// text before the offset value
points_offset_n_post_text	*text*	// text after the offset value
points_offset_n_justification	*justification*	// justification of the text
points_offset_n_no_decimals	*integer*	// number of decimals in offset

The value of a **height** at the offset of the point can be calculated and labelled using the parameters

points_height_n_mode	*1*	// use height of point itself
	3	// use real world height of position
		// above boxes
	100	// height of primary string (same as 1)
	101-500	// use height of to tin1 or tin2 etc.

points_height_n_position	*1*	// at points position
	3	// above top of boxes
	4	// above top of graph area but not including
		// the extra_space_top
	100	// to primary string (same as 1)
	101-500	// to tin1 or tin2 etc.
points_height_n_x	*mm*	// offset adjustment to position
points_height_n_y	*mm*	// height adjustment to position
points_height_n_angle	*degrees*	// rotation about point
points_height_n_size	*mm*	// size of text, 0 don't label
points_height_n_colour	*colour*	// colour of text

points_height_n_textstyle	*textstyle*	// textstyle to use for points height text.
points_height_n_pre_text	*text*	// text before the height value
points_height_n_post_text	*text*	// text after the height value
points_height_n_justification	*justification*	// justification of the text
points_height_n_no_decimals	*integer*	// number of decimals in height

A **label** which can include the **name** of the point is drawn by using the parameters

points_label_n_position	*1*	// at the points position
	3	// above top of boxes
	4	// above top of graph area but not including
		// the extra_space_top
	100	// to primary string (same as 1)
	101-500	// to tin1 or tin2 etc.
points_label_n_mode	0	// don't include point name
	1	// include point name in label
points_label_n_x	*mm*	// offset adjustment to position
points_label_n_y	*mm*	// height adjustment to position
points_label_n_angle	*degrees*	// rotation about point
points_label_n_size	*mm*	// size of text, 0 don't label
points_label_n_colour	*colour*	// colour of text
points_label_n_textstyle	*textstyle*	// textstyle to use for points label text.
points_label_n_pre_text	*text*	// text before the point name
points_label_n_post_text	*text*	// text after the point name
points_label_n_justification	*justification*	// justification of the text

Please continue to the next section 44.2.9 Labelling Cuts of X-Sections Through Strings in a Model.

44.2.9 Labelling Cuts of X-Sections Through Strings in a Model

The cuts that each x-section string makes though any strings in user given models can be automatically labelled on the x-section plots.

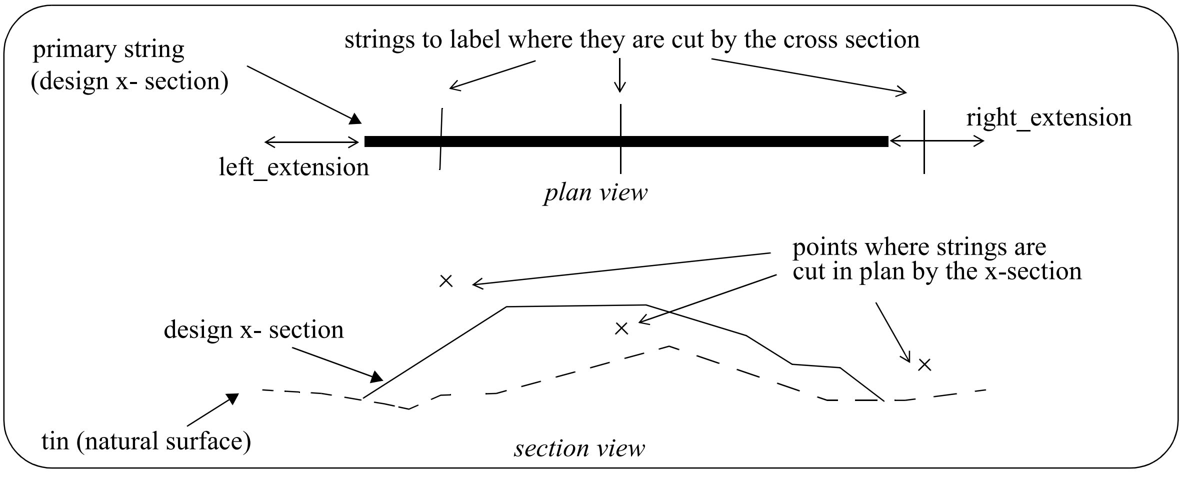
The **height**, **offset** and **name** of the cut string can be labelled as well as a **symbol** drawn. The height of tins at the same offset value can also be labelled.

The offset position for the labelling is the offset of the cut string.

The height position for the labelling can be specified as the

(a) top of the boxes on the x-section
(b) above the maximum height of the strings on the plot
(c) height value of the cut string
(d) height of the x-section string (the primary string)
(e) height of a tin.

The actual position of the label is defined relative to the above point.



Note:

Only case (b) involves the actual height of the cut string. For all other cases, only the offset of the cut string is used. Hence for all cases except (b), the string does need to have a sensible height to be used for cuts through strings.

For example, a boundary string may have null heights but only the offset is required and the height of the tin at that offset can be used as the height (case (d)).

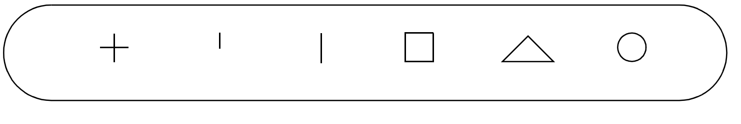
Text justification refers to the actual position and is given by

"top-left"	"top-centre"	"top-right"
"middle-left"	"middle-centre"	"middle-right"
"bottom-left"	"bottom-centre"	"bottom-right"

A choice of six special symbols and/or or any 12d symbols can be drawn at the cut point.

The special 12d Model symbols of size one millimetre are drawn in a square box centred on (0,0) with sides of length two millimetres. That is, the box co-ordinates are (-1,-1), (1,1), (1,-1), (-1,-1).

The six special shapes are



Up to twenty five (25) separate models of strings can be cut and labelled.

Parameters for Labelling Cuts of X-Sections Through Strings in a Model

The method for specifying which strings are to be checked for cuts is by first specifying the **model** which contains the strings, and then a **name mask** which is used to restrict the strings in the model to only those whose name matches the name mask.

Up to twenty five different sets of models and name masks can be used so that different cut sets can be labelled in different ways.

The parameters for selecting and labelling the nth set (where n can be from 1 to 25) of cuts of the x-sections with the strings in the model are given by:

cuts_n_model   *model_name*   // model of strings to be cut

The selection of the strings from the model *model_name* whose cut points are to be labelled is all the strings whose name satisfies the name mask *cuts_n_mask*:

cuts_n_mask   *name_mask*   // strings to check for cuts
// and if a cut occurs,
// parameters show how to
// label the cut

where *name_mask* is a text string containing the name masks, each separated by one or more spaces, to test the string name against. Each mast can include wild cards and wild characters.

For example

cuts_1_mask   "ke*"

or   cuts_1_mask   "?bank*"

or, if both masks are required,

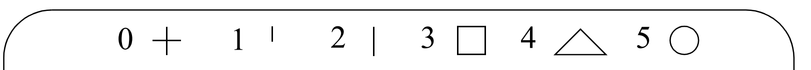
cuts_1_mask   "ke*   ?bank*"

If *cuts_n_mask* is missing, then all strings in the model are used. This is equivalent to *name_mask* being "*".

All strings in the model *cuts_n_model* whose name satisfy the name mask *cuts_n_mask* are then checked for cuts with the x-sections, and if a cut occurs, the cut point will be labelled according to the rest of the parameters in the nth set.

The parameters for drawing a **symbol** at the cut points are:

cuts_symbol_n_mode	0	// cross
	1	// up from centre of box
	2	// up and down from centre of box
	3	// square
	4	// triangle, base at bottom
	5	// circle
	6	// use a 12d symbol



predefined symbols for

If *cuts_symbol_n_mode* is 6, then the plot symbol is given by

cuts_symbol_n_style   *plotsymbol*   // 12d symbol to draw at cut

Important Note

The plot symbol of name *plotsymbol* is defined in the file given by:

(a) the parameter *plot_symbols* in the ppf file

plot_symbols   *filename*

or if plot_symbols is not defined, then

(b) in the file pointed to by the environment variable PLOT_SYMBOLS_4D

PLOT_SYMBOLS_4D *filename* *// default plotsym.4d*

or if PLOT_SYMBOLS_4D is not defined, then

(c) in the file *plotsym.4d*

which is searched for in the standard set up file sequence

If none of the above files are defined, or if the symbol does not exist in the above files, then it will be searched for in the standard 12d symbols file which is:

(d) either pointed to by the environment variable SYMBOLS_4D

SYMBOLS_4D filename // default symbols.4d

or if the environment variable SYMBOLS_4D does not exist, in the file, *symbols.4d*

The position of the symbol is given by:

```
cuts_symbol_n_position     1          // above point height value
                           3          // above top of boxes
                           4          // above top of graph area but not including
                                      // the extra_space_top
                           100        // to primary string
                           101-500    // to tin1 or tin2 etc.
```

The symbol can be adjusted by the parameters:

```
cuts_symbol_n_x            mm         // offset adjustment to position
cuts_symbol_n_y            mm         // height adjustment to position
cuts_symbol_n_angle        degrees    // rotation about point
cuts_symbol_n_colour       colour     // colour of symbol
```

and for all values of *cuts_symbol_n_mode* **other** than 6:

```
cuts_symbol_n_size         mm         // size of symbol, 0 don't draw
```

The value of the **offset** of the cut string can be labelled using the parameters

```
cuts_offset_n_position     1          // above cut strings height value
                           3          // above top of boxes
                           4          // above top of graph area but not including
                                      // the extra_space_top
                           100        // to primary string
                           101-500    // to tin1 or tin2 etc.

cuts_offset_n_x            mm         // offset adjustment to position
cuts_offset_n_y            mm         // height adjustment to position
cuts_offset_n_angle        degrees    // rotation about point
cuts_offset_n_size         mm         // size of text, 0 don't label
cuts_offset_n_colour       colour     // colour of text
cuts_offset_n_textstyle    textstyle  // textstyle to use for cuts offset text.
cuts_offset_n_pre_text     text       // text before the offset value
cuts_offset_n_post_text    text       // text after the offset value
cuts_offset_n_justification justification // justification of the text
cuts_offset_n_no_decimals  integer    // number of decimals in offset
```

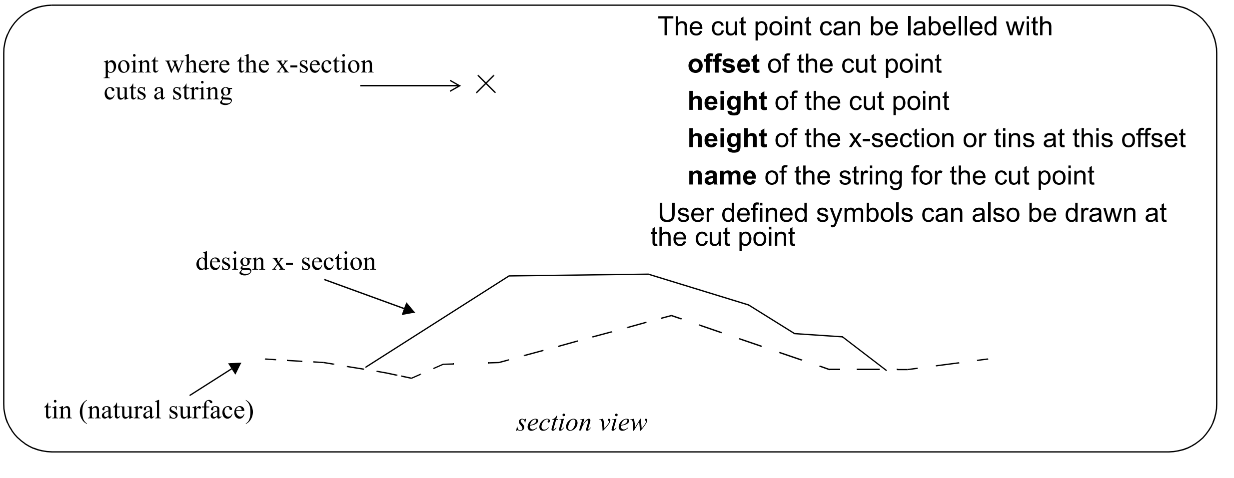
The value of a **height** at the offset of the point can be calculated and labelled using the parameters

```
cuts_height_n_mode         1          // use height of cut point itself
                           3          // use real world height of position
```

		// above boxes
	100	// height of primary string
	101-500	// use height of to tin1 or tin2 etc.
cuts_height_n_position	*1*	// at points position
	3	// above top of boxes
	4	// above top of graph area but not including
		// the extra_space_top
	100	// to primary string
	101-500	// to tin1 or tin2 etc.
cuts_height_n_x	*mm*	// offset adjustment to position
cuts_height_n_y	*mm*	// height adjustment to position
cuts_height_n_angle	*degrees*	// rotation about point
cuts_height_n_size	*mm*	// size of text, 0 don't label
cuts_height_n_colour	*colour*	// colour of text
cuts_height_n_textstyle	*textstyle*	// textstyle to use for cuts heights text.
cuts_height_n_pre_text	*text*	// text before the height value
cuts_height_n_post_text	*text*	// text after the height value
cuts_height_n_justification	*justification*	// justification of the text
cuts_height_n_no_decimals	*integer*	// number of decimals in height

A **label** which can include the **name** of the cut string is drawn by using the parameters:

cuts_label_n_position	1	// above cut strings height value
	3	// above top of boxes
	4	// above top of graph area but not including
		// the extra_space_top
	100	// to primary string
	101-500	// to tin1 or tin2 etc.
cuts_label_n_mode	0	// don't include cut string name
	1	// include cut string name in label
cuts_label_n_x	*mm*	// offset adjustment to position
cuts_label_n_y	*mm*	// height adjustment to position
cuts_label_n_angle	*degrees*	// rotation about point
cuts_label_n_size	*mm*	// size of text, 0 don't label
cuts_label_n_colour	*colour*	// colour of text
cuts_label_n_textstyle	*textstyle*	// textstyle to use for cuts label text.
cuts_label_n_pre_text	*text*	// text before the string name
cuts_label_n_post_text	*text*	// text after the string name
cuts_label_n_justification	*justification*	// justification of the text



Please continue to the next section 44.2.10 Hatching Cut and Fill Areas.

44.2.10 Hatching Cut and Fill Areas

This option is used to hatch cut and/or fill areas between sets of tins.

For each set, the name of the two tins, the hatch linestyle, colour and separation and whether cut and/or fill regions are required are all user definable.

Up to twenty (20) separate sets of tins be hatched.

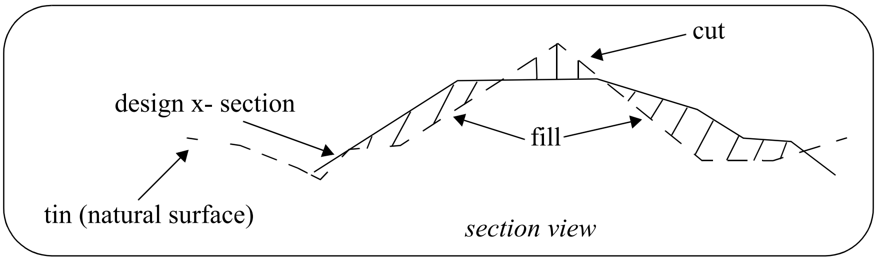
The parameters for labelling cuts and/or fill regions between tins are given by:

hatch_original_tin_n	*tin_name*	// tin_name for original surface
hatch_new_tin_n	*tin_name*	// tin_name for final surface
hatch_cut_separation_n	*mm*	// distance between cut hatch lines
	0	// don't do cut hatching
hatch_cut_angle_n	*degrees*	// angle in degrees of cut hatching
hatch_cut_colour_n	*colour*	// colour of the cut hatching
hatch_cut_linestyle_n	*linestyle*	// linestyle for cut hatching
hatch_cut_draw_sides_n	1/0	// 1 = draw sides of cut regions
hatch_cut_draw_original_n	1/0	// 1 = draw original tin in cut regions
hatch_cut_draw_new_n	1/0	// 1 = draw new tin in cut regions
hatch_fill_separation_n	*mm*	// distance between fill hatch line
	0	// don't do fill hatching
hatch_fill_angle_n	degrees	// angle in degrees of fill hatching
hatch_fill_colour_n	colour	// colour of the fill hatching
hatch_fill_linestyle_n	linestyle	// linestyle for fill hatching
hatch_fill_draw_sides_n	1/0	// 1 = draw sides of fill regions
hatch_fill_draw_original_n	1/0	// 1 = draw original tin in fill regions
hatch_fill_draw_new_n	1/0	// 1 = draw new tin in fill regions

Notes

(a) cut is when the new tin is below the original tin.
fill is when the new tin is above the original tin.

(b) cut hatching is turned off by setting *hatch_cut_separation_n* to 0.0.
fill hatching is turned off by setting *hatch_fill_separation_n* to 0.0.



Please continue to the next section 44.2.11 Extra Models of X-Sections.

44.2.11 Extra Models of X-Sections

The order and centreline chainages of the x-section subplots for the x-section plot are defined by the sections from the primary model of cross sections.

The sections through any specified triangulations and service models, and offsets for labelling are fully defined by these primary x-sections.

However it is also possible to plot extra x-sections on each of the sub-plots by supplying extra models of x-sections which are at the same plan positions as the primary x-section strings.

Only those x-sections from the extra models that are within a user specified tolerance of a primary x-section are plotted.

The extra models of x-sections and the plan tolerance for checking that the extra cross-sections are the same position as the primary x-sections are

extra_model_tolerance_1 *value* // tolerance in ***12d Model*** units

extra_model_n *model_name* // where n=1, ... 100
// extra models of x-sections

The actual colour of the extra x-sections is used as their plotting colour.

Please continue to the next section 44.2.12 Sorting X-Sections by Chainage.

44.2.12 Sorting X-Sections by Chainage

The chainage on the design string that the x-section is created at is stored with the cross-section.

When the x-sections are created by the apply functions, they are created in the order of increasing chainage and added to the *model for sections* in that order.

Hence if the cross sections are plotted, the natural order would be in increasing chainage order.

However, it is possible to upset the order in the model by manually adding in extra sections.

Hence there is a parameter in both the **section x plot** panel and the plot parameter file to ensure that the cross sections are sorted in increasing chainage order before they are plotted.

sort_sections	***1***	// sort sections by design chainage
	0	// use order of sections in model

Please continue to the next section 44.2.13 Title Block Information.

44.2.13 Title Block Information

The plot can have a standard ***12d Model*** title block or a user defined title block.

The standard title block consists of a simple border around the plot and two lines of text in a box underneath the plot. For a user defined title block, all the line work and text is defined by the user.

Standard Title Block

For the standard ***12d Model*** title block, there are extra parameters for two lines of text and text size and colour. The standard title block is turned on or off by the parameter *plot_border*.

plot_border	*yes/no*	*// yes plots a standard title block* *// default yes*
title_1	*text*	
title_2	*text*	
title_text_size	*value*	
title_colour	*colour*	

User Title Block

For the user defined title block, the title block drawing commands are kept in a file whose name is supplied by the user. The title block drawing commands are almost identical to the linestyle drawing commands and is given at the beginning of this chapter.

Hence for a user defined title block, there are just two parameters - one to say a title block file is being used and the other to give the name of the title block file. The *plot_border* parameter should also be set to *no* so that the standard title block is not also drawn.

use_title_file	*yes/no*	*// yes draws the title block given in title_file* // default no
title_file	*filename*	
plot_border	*no*	*// turn off standard title block*

Some special plot parameters are used to pass information down to variables in a user defined title block. For example, inside the title block file it is possible to have runtime user defined text variables. The actual text values for these text variables are passed down to the title block file from the plot parameter file via the parameters user_text_n (n = 1,2,... 1000)

user_text_n		*text*

The special plot parameters are:

time_format	*text*	// format for $time
user_text_n	*text*	// where n = 1,2,... 1000 // passed down to $user_text_n
title_1	*text*	// passed down to $title_1
title_2	*text*	// passed down to $title_2
start_page_number	*integer*	// used as the starting value for // $page_number. If missing, // $page_number starts at 1.

start_drawing_number	*integer*	// added to $drawing_number in title
		// block file. If missing,
		// $drawing_number starts at 1.

drawing_number_prefix	*text*	// passed down to // $drawing_number_prefix
drawing_number_postfix	text	// passed down to // $drawing_number_postfix

Please continue to the next section 44.2.14 Parameters that Modify Fields In the Cross Plot Panel.

44.2.14 Parameters that Modify Fields In the Cross Plot Panel

A number of parameters match those in the **section x plot** panel.

When the plot parameter file is first read, any parameters in the panel will be replaced by the values of any corresponding parameters in the parameter file.

However, if the parameter is subsequently modified in the panel, the panel value will be the value used for any plots.

The plot parameters that also occur in the **section x plot** panel are:

view_name	*text*
plotter_type	*text*
model_to_plot	*text*
plot_stem	*text*
start_chainage	*chainage*
end_chainage	*chainage*
sheet_size	*text* or *"width height"*
scale	*value*
absolute_extensions	*yes/no*
left_extension	*world-units*
right_extension	*world-units*
line_up_cl	*yes/no*
primary_string	*yes/no*
label_type	*yes/no*
box_colour	*colour*
text_size	*mm*
sort_sections	*yes/no*
use_title_file	*yes/no*
title_file	*filename*
plot_border	*yes/no*
title_1	*text*
title_2	*text*
title_text_size	*value*
title_colour	*colour*
global_textstyle	*textstyle*

Please continue to the next section 44.2.15 Generating Cross Section Plots Without a View.

44.2.15 Generating Cross Section Plots Without a View

The cross section plot parameters are comprehensive enough that it is possible to completely generate a cross section plot without referencing a section view, or even using the **section X plot** panel.

Such a ppf can be run using the **plots=>plot a ppf** option or from the 4D Solutions programming language, 4DL.

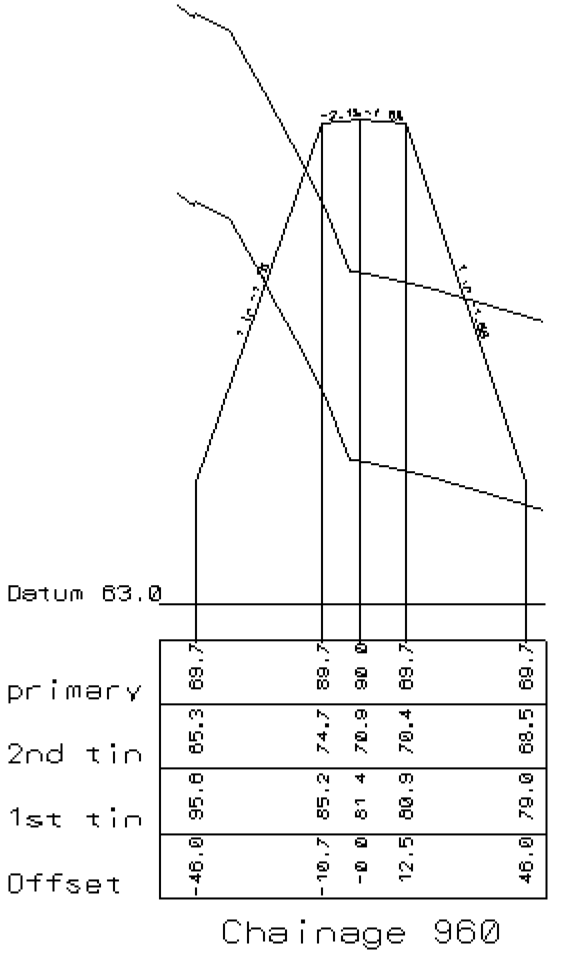
When generating a cross section plot using the **section X plot** panel, a plot parameter file containing all the parameters needed to regenerate the plot using **plots=>plot a ppf** can be written out by simply giving a name for the ppf file in the **plot parameters write** field of the **section X plot** panel.

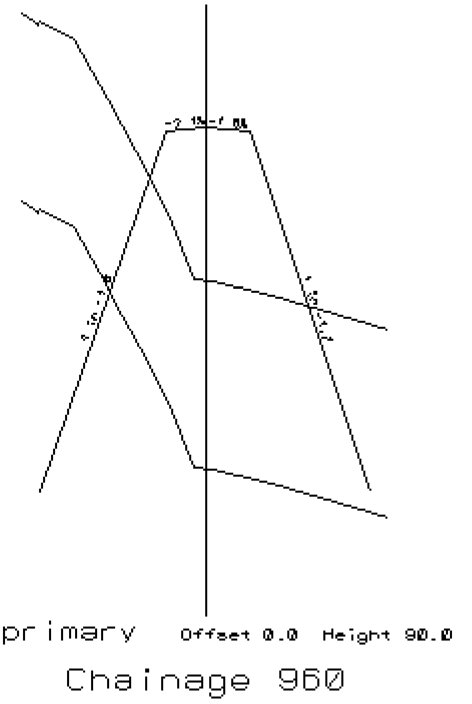
Notes

1. A warning is given if the keyword in a plot parameter file does not exist.
2. A warning is also given if the key word pair is defined more than once in a ppf.

Please continue to the next section 44.2.16 Example of a Cross Section Plot Parameter File.

44.2.16 Example of a Cross Section Plot Parameter File





// ppf file to generate x-section example

```
section_x_plot "plot 2" {

// plot margins
 left_margin              0.0
 right_margin             0.0
 top_margin               0.0
 bottom_margin            0.0
// white space surrounding each cross section
 left_sub_plot_gap        20.0
 right_sub_plot_gap       20.0
 top_sub_plot_gap         20.0
 bottom_sub_plot_gap      20.0
// these are only used with title boxes
 left_border_gap          10.0
 right_border_gap         10.0
 top_border_gap           10.0
 bottom_border_gap        10.0
// annotation for boxes area
//    height text on centre line plot
 height_text              "Height"
//  mode for drawing boxes around
//     0 = none
//     1 = heights area
//     2 = heights and labels area
 draw_box_mode            1
 box_line_draw_mode       3
 box_colour               grey
// size of title text in title box
 title_box_text_size      5
// title text box size - automatically calculated if missing
// space_for_titles           30
// horizontal_line_spacing    15
//   number of decimal places for height and offset values
 number_of_decimals       1
//   0 for bottom justification of values
//   1 for top justification of values
 box_text_justification   1
//    offset titles and values
 offset_title             "Offset"
// offset_title_2           "Offset 2"
// offset_colour            brown      // also in panel
 offset_title_colour      grey
//   primary string titles and heights
//   defaults to string name and colour
 primary_title            "primary"
// primary_title_2          "primary2"
 primary_colour           "purple"
 primary_title_colour     "red"
```

```
//    tin titles and heights

//    tin label 1 for on
//              0 for off
//    tin text is the annotation text for that tin
//    tin colour is the annotation colour for that tin
//    range from tin_1_X to tin_100_X

 tin_1_label                    1
 tin_1_title                   "1st tin"
// tin_1_title_2                 "1st tin 2"
 tin_1_colour                   grey
 tin_1_titlel_colour             red

 tin_2_label                    1
 tin_2_title                   "2nd tin"
 tin_2_colour                   grey
 tin_2_title_colour             red

//   staggers definitions

 stagger_gap_factor              1.3
 stagger_gap_top                 5.0
 stagger_gap_bottom              2.0

//   uprights definitions

// mode 0 = none
//          1 = full
//          2 = ticks
//          3 = height above boxes
//      100 = primary
//      101 = 1st tin
//      102 = 2nd tin
//...
//      500 = 400th tin

 uprights_colour                red
 uprights_draw_mode             100
 uprights_y                     100

// datum area

 datum_name                    "Datum"
 datum_colour                   orange
 datum_text_size                 4
 datum_below_gap                10.0

//  gap above datum to graph area for box and
//   centre line plots

 datum_above_gap                 20
 datum_above_gap_cl              20

// grade_label of 0 turns the labels off
//                       1 turns it on

 grade_label                    1

// mode 0 for percent grade,
//          1 for one in x,
//           2 for grade in m/m

 grade_mode                     0
```

```
// grade decimal places
 grade_decimals                  1
// text size for grade annotations
 grade_size                      2
 grade_colour                    yellow
// how far the annotation is (perpendicularly)
//    from the string itself
 grade_offset                    1
//  label all annotations separated by more than
//  the minimum width in plot units
 grade_minimum_width             0
//  threshold and threshold mode describe the value
//   at which the grade labelling changes
//   mode to grade_upper_mode
 grade_threshold                 0.2
 grade_threshold_mode            2
 grade_upper_mode                1
 grade_change_only               1
// labelling of sub-plots
// centreline chainage: 1 for on, 0 for off
 chainage_label                  1
//   distance below box to print annotation
 chainage_y_offset               12
//   distance from start of string to label - default centred
// chainage_x_offset             10
//   chainage text label parameters
 chainage_text                   “Chainage”
 chainage_size                   6
 chainage_colour                 cyan
 chainage_decimals               2
```

// ******** panel data ********

```
 view_name                       “4”

 plotter_type                    “model”
 model_to_plot                   “single xsec”
 plot_stem                       “pp”

 sheet_size                      “1000   800”
 scale                           1000.0

 absolute_extensions             “no”
 left_extension                  5
 right_extension                 5

 line_up_cl                      "yes”
 primary_string                  “yes”
 label_type                      “boxes”
 offset_colour                   “cyan”
 text_size                       3.0
```

```
use_title_file        "no"
title_file            "titles.tf"
plot_border           "no"
title_1               ""
title_2               ""
title_text_size       4.0
title_colour          magenta
}
```

Please continue to the next section 44.3 Long Section Plot Parameter File.

44.3 Long Section Plot Parameter File

The long section plot parameters are placed in a file with ending **.ppf**

Each parameter consists of a parameter name followed by one or more spaces and then the parameter value. There is only one parameter per line.

Anything on a line after a double forward slash // is considered to be a comment.

The set of all parameters for the long section plot is enclosed within a set of curly brackets **{ }** with the header

section_long_plot "plot set name"

before the curly brackets.

That is,

```
section_long_plot   "plot set name"   {
            plot parameters
            one per line
}
```

If there is more than one section_long_plot parameter set in the file, only the first set is used.

There may also be parameter sets for other plot types such as section_x_plot in the same file. The other sets will be ignored when doing a long section plot.

The plot parameters are documented in following groups:

Defining chainage range	44.3.1 Possible Chainage Range.
Breaking the plot into pages	44.3.2 Pagination.
Plot Sheet layout	44.3.3 Plot Sheet Layout.
Boxes area	44.3.4 Boxes Area.
Defining chainages and staggering	44.3.5 Chainage Selection and Staggering.
Defining uprights	44.3.6 Uprights.
Datum area	44.3.7 Datum Area.
Graph (drawing)	44.3.8 Graph Area.
Defining bubbles	44.3.9 Bubbles Definitions.
Quick HG	44.3.10 Quick Horizontal Geometry Labelling.
Extensive HG	44.3.11 Extensive Horizontal Geometry Labelling.
Quick VG	44.3.12 Quick Vertical Geometry Labelling.
Extensive VG,	44.3.13 Extensive Vertical Geometry Labelling.
Labelling heights	44.3.14 Labelling Chainages and Heights in the Graph Area.
Symbols	44.3.15 Labelling With Symbols.
Hatching cut/fill	44.3.16 Hatching Cut and Fill Areas.
Labelling cuts	44.3.17 Labelling Cuts of Design Through Strings in a Model.
Labelling string name	44.3.18 Labelling the Primary String Name on the Plot.
Labelling with scale	44.3.19 Labelling the Scale on the Plot.
Title block information	44.3.20 Title Block Information.
Panel modifying parameters	44.3.21 Parameters that Modify Fields In the Long Plot Panel.
Example	44.3.23 Example of a Long Section Plot Parameter File.

44.3.1 Possible Chainage Range

In the panel, the field **Use HG VG for min, max** determines whether the horizontal geometry (HG) and the vertical geometry (VG) are both used to determine the minimum and maximum chainages for drawing. This allows the vertical geometry to be plotted when it is outside the horizontal geometry (e.g. kerb returns).

The equivalent parameter in the ppf is

use_vg_hg_for_min_max	***1***	// the chainage range available for plotting
		// is from the minimum of the horizontal
		// geometry (HG) and the vertical geometry
		// (VG) to the maximum of the HG and VG.
		// This is useful for kerb returns

	0	// the chainage range available for
		// plotting is from the minimum of the
		// horizontal geometry (HG) to the
		// maximum of the HG.

44.3.2 Pagination

If the long section plot is too long to fit on one page, it can be broken into a number of pages (sheets).

The parameter, *pagination_length*, controls the amount of new chainage length on each plot page. Each page of the long section plot can also included a set chainage amount from the end of the **previous** plot.

Hence apart from the first page and possibly the last page, the plot will have a chainage length given by the sum of the *pagination_length* and *pagination_overlap*.

pagination	*yes/no*	// if yes, then break the plot into pages
pagination_mode	*chainage*	// length and overlap in chainage units
	*millimet*re	// length and overlap in millimetres
pagination_length	*chainage or*	
	millimetres	// new chainage range on each page
pagination_overlap	*chainage or*	
	millimetres	// overlap between pages

Note

The *pagination_length* and *pagination_overlap* are given in chainage units **or** millimetres.The user must choose values which will fit on the selected sheet size or the end of each plot will be truncated by the sheet.

44.3.3 Plot Sheet Layout

The plot **page** or **sheet** is considered to have only positive co-ordinates with the origin (0,0) in the left hand corner. The units for the plot are millimetres.

The overall size of the plot sheet is given by either a defined sheet size, or by the width and height of the plot given in millimetres and separated by one or more spaces.

sheet_size	*text*	// sheet name, or
	"mm mm"	// sheet size: width height

The sheet size name and width and heights can be specified by the user in a file named sheets.4d which is in the normal set up areas, or is pointed to by the environment variable

SHEET_SIZES_4D	*file*	// file of plotter sheets sizes

The plotting area is restricted to within the plot sheet by giving margins which are:

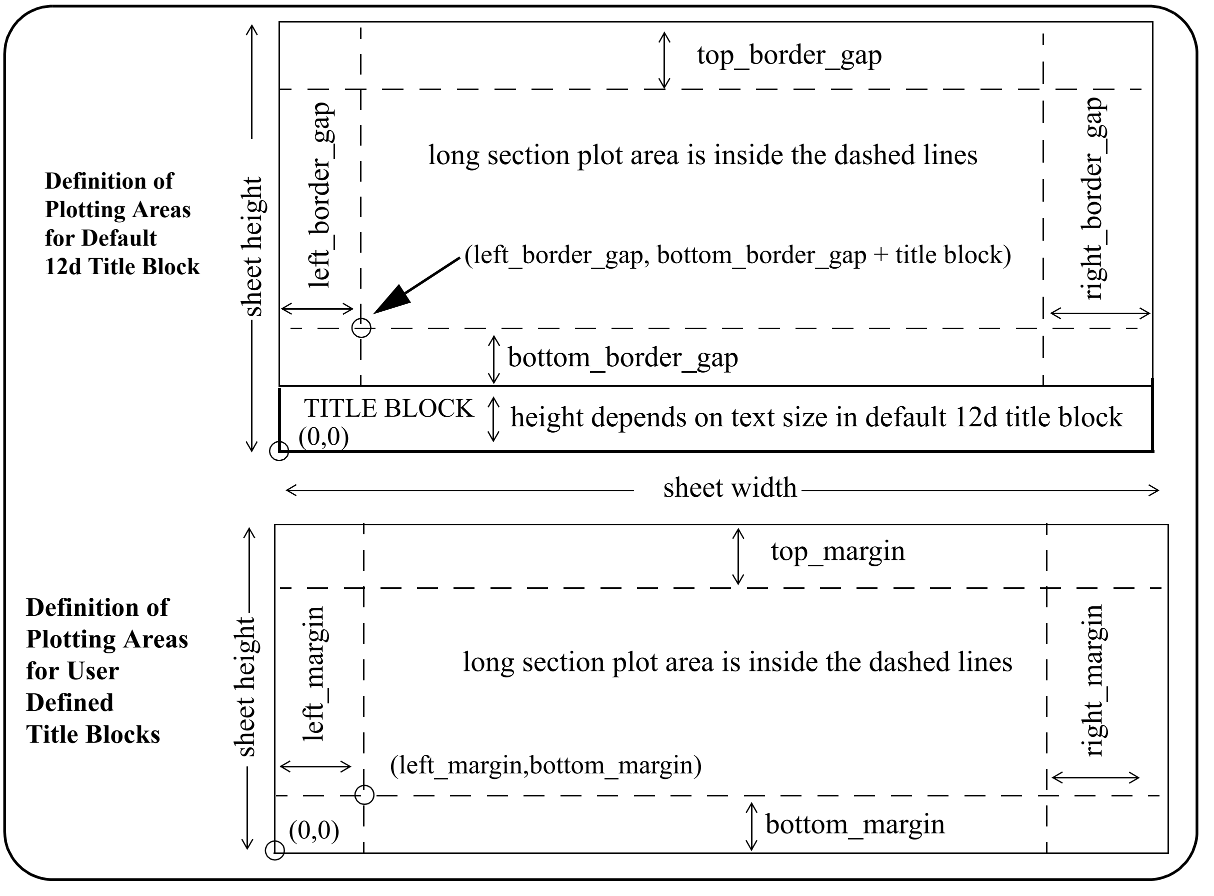
If a *User Defined Title Block* is used:

left_margin	*mm*
right_margin	*mm*
top_margin	*mm*
bottom_margin	*mm*

If the *default 12d title block* is used, then the size of the bottom of title block depends on the text size. The following parameters are used in the default title block case and the *bottom_border_gap* is added to the calculated height of the bottom of the title block.

left_border_gap	*mm*
right_border_gap	*mm*
top_border_gap	*mm*
bottom_border_gap	*mm*

Because the user can easily select from the plotting panel whether a User Defined Title Block or the default 12d title block is used, both sets of margins and gap parameters can exist in the one plot parameter file.



The long plot itself consists of the three regions - boxes, datum and graph areas.

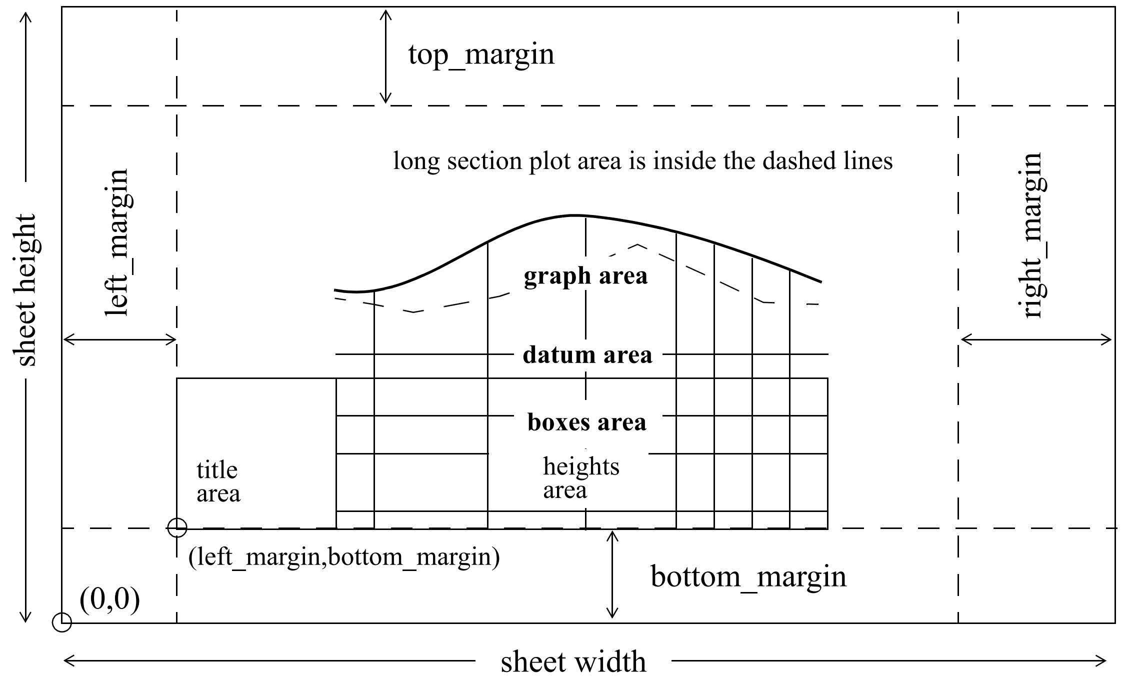
The **boxes area** is where the titles and the chainage values and the heights/depths for the strings drawn on the long plot are labelled.

The **datum area** is the region between the boxes area and the graph area.

The **graph area** is the area where the actual plots of the strings are drawn.

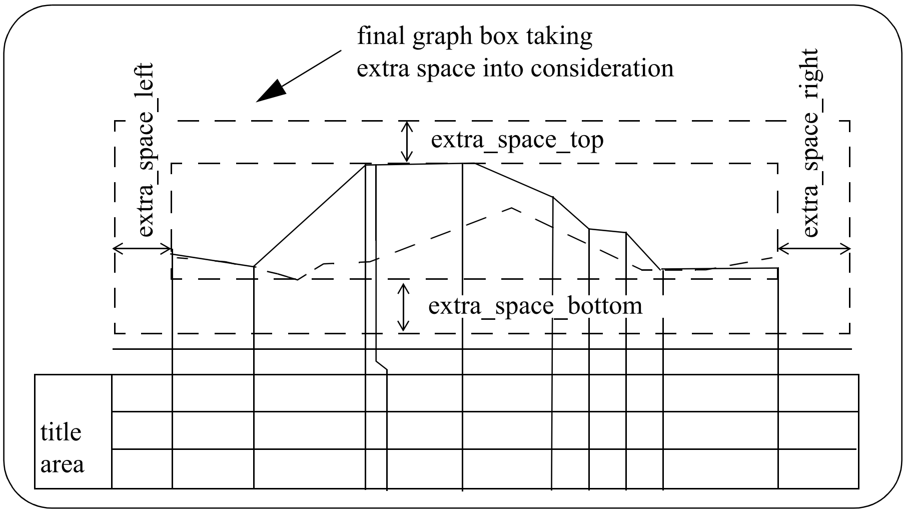
Apart from information labelled in the boxes area, the long section plot can label other information such as

(a) horizontal geometry
(b) vertical geometry
(c) chainage, heights at special points
(d) symbols at special points
(e) bubbles at special chainages
(f) cuts the design string makes through strings
(g) cut and fill areas



The size of the graph area can also be extended to allow for plot symbols by the parameters:

extra_space_units	***0***	// extra space values are world units - default
	1	// " " " are millimetres on plot page
extra_space_left	*units*	// subtract from left of plot area, def 0
extra_space_right	*units*	// add to right of plot area, default 0
extra_space_top	*units*	// add to top of plot area, default 0
extra_space_bottom	*units*	// subtract from bottom of plot area, // default 0



Please continue to the next section 44.3.4 Boxes Area.

44.3.4 Boxes Area

Each string in the long section plot can be labelled with one or two lines of title, and the chainages/heights/depths at the user specified chainages for the strings.

The **title** for the strings, is drawn in the **title area** of the **boxes area**.

The **chainages/heights/depths** are drawn in the **heights area** of the **boxes area**.

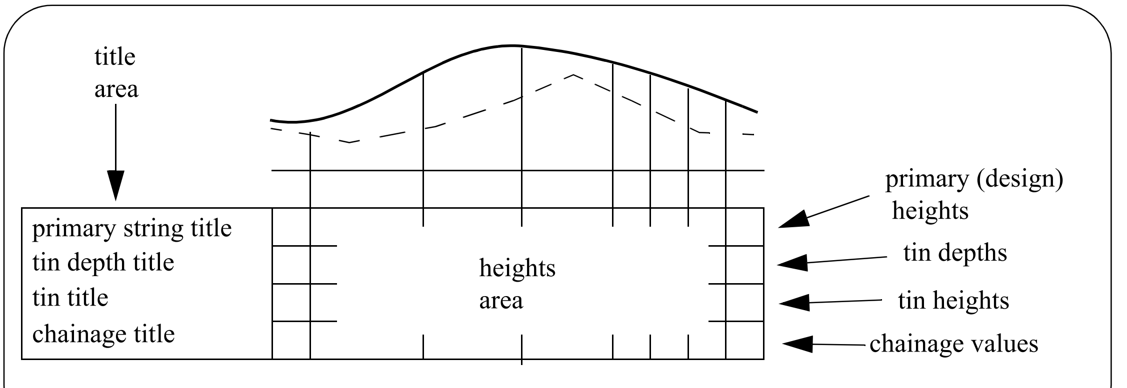
Consequently the boxes area is made up of rows of text consisting of:

string/tin titles followed by the **chainage/height/depth** values along the string.

Each row is surrounded by lines to form a box.

The default order of the boxes from the bottom up is

(a) optional super-elevation diagram
(b) chainage values
(c) can be primary string heights - user choice
(d) tin heights and depths
(e) offset heights and depths
(f) optional volumes or earth works
(g) can be primary string heights - user choice
(h) zero or more blank boxes



boxes for long section plot

The blank boxes are used to place other information in (such as horizontal or vertical geometry) or for other user supplied information.

Placing the primary string label before the bank boxes or just after the chainage box in controlled by the parameter *primary_label_mode*.

number_of_blank_boxes	*integer*	// integer > 0
primary_label_mode	0	//put primary string labels in last box // before blank boxes
	1	// or first box above the chainages

Although the order of the boxes may appear to be fixed, in practice they can be in any order. This is possible because for each box, there is a parameter to set the height in millimetres from the bottom of all the boxes that the text in the box is drawn at (the *_y_pos* parameters).

The **title area** starts at the co-ordinate (left_margin,bottom_margin).

The size of the title text is given by the ***title_box_text_size*** parameter.

The **width** of the title area is either given by the *space_for_titles* parameter, or if omitted, the required width is automatically calculated.

title_box_text_size	*mm*	// size of the title text for boxes
space_for_titles	*mm*	// calculated if omitted

There can be two lines of title text and the title text, textstyle and colour can be set independently for each box.

The x position of the title text is the same for all the lines of title text and can be set to be a fixed distance from the left hand side of the boxes.

box_titles_x	*mm*	// distance to move the title text from the // left hand side of the boxes

The y position of the title text can be set separately for each type of title. The parameters are given later under each of the title types (e.g. chainages, primary string, tins, depths etc.).

After the title area there can be a user defined gap, followed by the heights area

box_gap	*mm*	// blank area between the title area // and the heights area default 0

The **heights area** starts at the end of the title area.

The height text is written at right angles to the bottom of the boxes. It can be either top or bottom justified with respect to the box (*box_text_justification*).

The number of decimal places (*number_of_decimals*), and the size of the heights text (*box_text_size*) can be specified globally but there is a parameter for each box which overrides these defaults for each box of heights.

box_text_justification	0 1	// bottom of individual boxes // top of individual boxes

number_of_decimals	*integer*	// number of decimal places in the
		// height boxes. If <0, the absolute value
		// is taken as the number of decimal
		// places i.e. no trailing zeros are
		// removed for the values in the heights
		// area.

box_text_size	*mm*	// size of the height text values

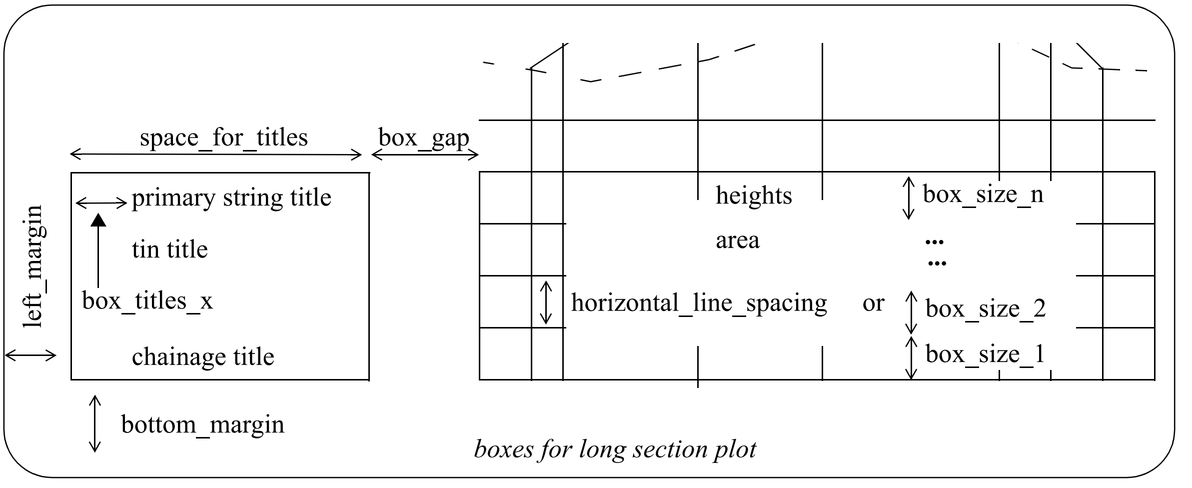
A global height for the individual boxes is either given by the *horizontal_line_spacing* parameter, or if omitted, a height to fit the largest height or depth value is calculated and used as the default box height.

horizontal_line_spacing	*mm*	// calculated if omitted

However, the height of each box can be individually set by parameters *box_size_n* where the boxing numbering, n. starts from the bottom box. The value of *horizontal_line_spacing* is used for any of the *box_size_n* parameters not specified.

box_size_n	*mm*	// height of the nth box, numbered from // bottom up.

The total height of the boxes area is simply given by the sum of the heights of each box.



The **width** of the **heights area** is determined by the number of chainages to be labelled and whether the values are staggered to prevent over writing.

Hence the total width of the boxes area is the width of the **title area** plus box_gap, plus the width of the **heights area**.

Many distance definitions in the plot parameter file are given in terms of distance above the **top** of the boxes area so that the distances are independent of the number of boxes and box sizes.

The drawing of the box line work, the box colour, the position of the primary string labels and the number of blank boxes are all set by parameters.

First, all the line work for the box itself is user defined. This has been extended from V3.1 although *draw_box_mode* has been left in for upward compatibility.

draw_box_mode	*0*	// don't draw box lines
	1	// draw box lines around heights area // only
	2	// draw box lines around title and // heights areas

	3	// use draw_box_side_n,
		// box_line_draw_mode and
		// box_line_mode_n

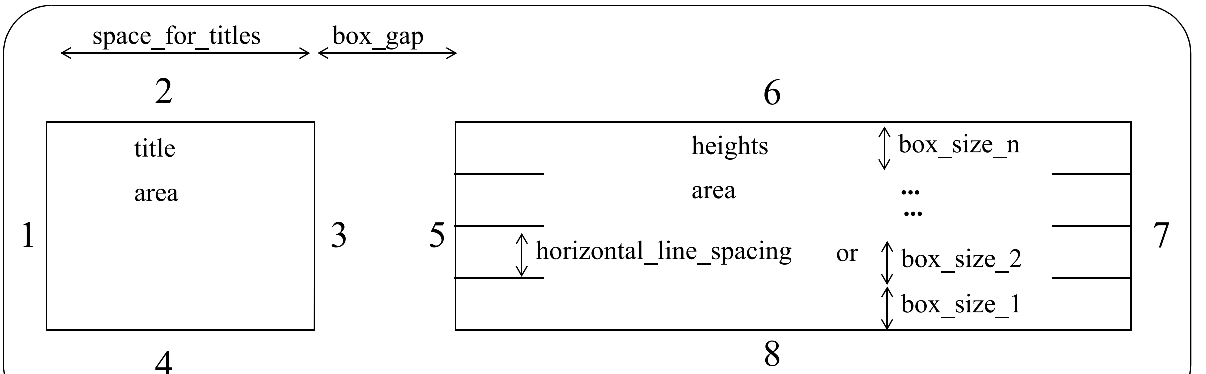
box_colour	*colour*	// colour of the boxes

For V3.2, the line work for the outside of the title and heights boxes is controlled by the parameters *draw_box_side_n* and *box_side_colour_n*, and the separation lines inside the boxes are controlled by the parameters *box_line_draw_mode* and *box_line_mode_n* where the box numbering, n, starts from the bottom box.

Note: The following parameters are only used if ***draw_box_mode*** is set to **3**.

draw_box_side_1	***1***	// draw the left side of the title area (def)
	0	// don't draw the left side
box_side_colour_1	*colour*	// colour to draw left side of title area // default box_colour
draw_box_side_2	***1***	// draw top of the title area (default)
	0	// don't draw the top
box_side_colour_2	*colour*	// colour to draw top of title area // default box_colour
draw_box_side_3	*1*	// draw right side of the title area

		// (default if box_gap is non-zero)
	0	// don't draw the right side
		// (default if box_gap is zero)
box_side_colour_3	*colour*	// colour to draw right side of title area
		// default box_colour
draw_box_side_4	***1***	// draw bottom of the title area (default)
	0	// don't draw the bottom
box_side_colour_4	*colour*	// colour to draw bottom of title area
		// default box_colour
draw_box_side_5	***1***	// draw left side of the heights area (def)
	0	// don't draw the left side
box_side_colour_5	*colour*	// colour to draw left side of heights area
		// default box_colour
draw_box_side_6	***1***	// draw top of the heights area (default)
	0	// don't draw the top
box_side_colour_6	*colour*	// colour to draw top of heights area
		// default box_colour
draw_box_side_7	***1***	// draw right side of the heights area def
	0	// don't draw the right side
box_side_colour_7	*colour*	// colour for right side of heights area
		// default box_colour
draw_box_side_8	***1***	// draw bottom of the heights area (def)
	0	// don't draw the bottom
box_side_colour_8	*colour*	// colour for bottom of heights area
		// default box_colour



lines to draw for draw_box_side_1 to draw_box_side_8

The lines at the top of the individual boxes inside the title area and heights area (separation lines) are controlled by the parameters *box_line_draw_mode* and *box_line_mode_n*.

The separation lines can be drawn in just the title area, just the heights area or in both areas.

The parameter *box_line_draw_mode* can be set to control all the separation lines but there are additional parameters, *box_line_mode_n*, which override *box_line_draw_mode* for each of the individual boxes where n = 1, ... number of boxes -1.

The top of the top box is not controlled by box_line_mode_n but is controlled by the parameters *draw_box_side_2* and *draw_box_side_6.*

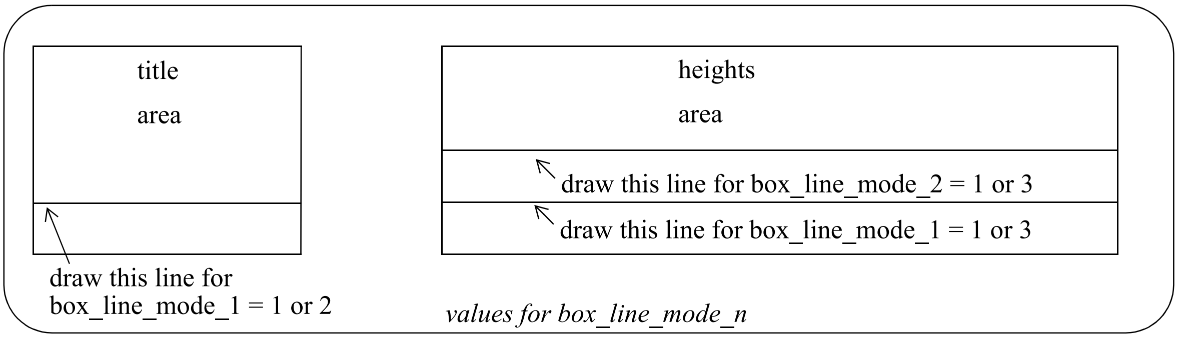
box_line_draw_mode	*0*	// don't draw any separation lines inside
		// the title and heights areas
	1	// draw the separation lines inside the
		// title and heights areas (default)
	2	// draw the separation lines inside the
		// title area only

	3	// draw the separation lines inside the // heights area only

box_line_mode_n	*0*	// for the nth box, don't draw any
		// line at the top of the box for either
		// the title or the heights areas

	1	// for the nth box, draw the line at the
		// top of the box for both the
		// title and heights areas

	2	// for the nth box, draw the line at the // top of the box for the title area only
	3	// for the nth box, draw the line at the // top of the box for the heights area only



Chainage Titles and Values

The chainages are always labelled

chainage_title	*text*	// first line of chainage title // (default chainages)
chainage_title_2	*text*	// second line of chainage title
chainage_title_textstyle	*textstyle*	// textstyle of the chainage titles
chainage_title_colour	*colour*	// default chainage title colour
chainage_title_size	*mm*	// default chainage title size

chainage_title_y_pos	*mm*	// if set, the height in mm above the
		// bottom of all the boxes that the
		// chainage title text is drawn.
		// If not set, then the text is placed at a
		// height that puts it inside the default
		// box for the chainage.

chainage_decimals	*integer*	// number of decim places for chainages
chainage_textstyle	*textstyle*	// textstyle of the chainage values
chainage_colour	colour	// colour of chainages, def is box_colour
chainage_size	mm	// size of chainages, def is box_text_size

chainage_y_pos	*mm*	// if set, the height in mm above the
		// bottom of all the boxes that the
		// chainage values text is drawn.
		// If not set, then the text is placed at a
		// height that puts it inside the default
		// box for the chainage.

Example of Chainage Titles

chainage_title	"DESIGN"
chainage_title_2	"CHAINAGE"

chainage_title_colour	“yellow”	
chainage_colour	“grey”	// colour of chainage text
chainage_decimals	-3	// 3 decimal places - leave trailing zeros

Primary String Title and Heights

Sometimes the primary string is only required to define chainages or the horizontal path for tin sections. In this case, the primary string would not be drawn on the long section plot.

If the primary string is drawn, it is automatically labelled.

Also the drawing of crosses at the vertical intersection points can be controlled from the ppf file.

primary_draw_vips	***1***	// draw crosses at VIPS (default)
	0	// don’t draw VIP crosses

The position of the primary string label box can be either straight after the chainage box, or after the tin and offset boxes but before the blank boxes.

primary_string	*yes*	// draw and label the primary string
	no	// don't draw or label the primary string
primary_label_mode	*0*	// last box before blank boxes
	1	// first box above the chainages
primary_title	*text*	// first line of string title
primary_title_2	*text*	// second line of string title, default is // the primary string name.
primary_title_textstyle	*textstyle*	// textstyle of the primary string titles
primary_title_colour	*colour*	// colour of the string titles
primary_title_size	*mm*	// default primary title size
primary_title_y_pos	*mm*	// if set, the height in mm above the // bottom of all the boxes that the // primary string title text is drawn. // If not set, then the text is placed at a // height that puts it inside the default // box for the primary string.
primary_decimals	*integer*	// number of decimal places
primary_textstyle	*textstyle*	// textstyle of the primary string heights
primary_colour	*colour*	// colour of the heights
primary_size	*mm*	// default primary value size
primary_y_pos	*mm*	// if set, the height in mm above the // bottom of all the boxes that the // primary string height text is drawn. // If not set, then the text is placed at a // height that puts it inside the default // box for the primary string.
primary_ch_tolerance	*mm*	*// for a given chainage, if no z value exists, // this distance is added to/subtracted from the // chainage to search for a valid z value.*

Example of Primary String Titles

primary_title	“DESIGN”	
primary_title_2	“CENTRELINE”	
primary_title_colour	“yellow”	
primary_colour	“grey”	// colour of height text
primary_decimals	-2	// 2 decimal places - leave trailing zeros

Tin Titles and Heights

A section along the primary string through each tin on the section view can be automatically drawn on the long plot, and the user can specify whether the tin heights and depths are labelled or not in the boxes area.

Hence although there may be a number of tins drawn on the section plot, not all of them need to be labelled or have their depths labelled.

The default order for labelling the tins and depths is the order that the tins were added to the view however it is possible to specify which tin is used for labelling each row of tin heights and depths by giving the tin name rather than just using the tin order on the section view.

In fact, it is possible to use **any tin** in the project **to labe**l a row, **not just those drawn** on the section view.

The parameter to define the nth tin by name is *tin_n_name*:

```
tin_n_name          text          // n=1, ... no of tins on the section view.
                                  // use the tin named text to label the nth
                                  // row of tin heights and depths.
```

If a tin of the name ***tin_n_name*** **does** not exist, then the plot is not produced and an error message is given.

For the following parameters, n takes the value 1 to 100 and specifies that the parameter set applies to the nth tin on the section view, or if *tin_n_name* is set, for the tin specified by *tin_n_name*.

```
tin_n_draw_mode              1            // draw the nth tin (default)
                             0            // don't draw the nth tin
tin_n_draw_colour            colour       // colour to draw the nth tin.
                                          // If omitted, then use the tin colour
tin_n_label                  1            //label the tin's heights (default)
                             0            // don't label the tin's heights

tin_n_title                  text         // first line of tin title
tin_n_title_2                text         // second line of tin title
tin_n_title_textstyle        textstyle    // textstyle of the tin titles
tin_n_title_colour           colour       // colour for titles of heights
tin_n_title_size             mm           // default nth tin's title size
tin_n_title_y_pos            mm           // if set, the height in mm above the
                                          // bottom of all the boxes that the
                                          // tin title text is drawn.
                                          // If not set, then the text is placed at a
                                          // height that puts it inside the default
                                          // box for the tin heights.

tin_n_decimals               integer      // number of dec places in tin height
tin_n_textstyle              textstyle    // textstyle of the tin heights
tin_n_colour                 colour       // colour for heights and depths
tin_n_size                   mm           // default nth tin's title size
tin_n_y_pos                  mm           //if set, the height in mm above the
                                          //bottom of all the boxes that the
                                          // tin height text is drawn.
                                          //If not set, then the text is placed at a
                                          //height that puts it inside the default
                                          // box for the tin heights.

tin_n_depth_label            1/0          // 1 to label, 0 don't label depths
tin_n_depth_title            text         // first line of tin depth title
tin_n_depth_title_2          text         // second line of tin depth title
tin_n_depth_title_textstyle  textstyle    // textstyle of the tin depth titles
tin_n_depth_title_colour     colour       // colour for titles of depths
```

```
tin_n_depth_title_size      colour        // size for title of depths
tin_n_depth_title_y_pos     mm            // if set, the height in mm above the
                                          // bottom of all the boxes that the
                                          // tin depth title text is drawn.
                                          // If not set, then the text is placed at a
                                          // height that puts it inside the default
                                          // box for the tin depths.

tin_n_depth_decimals        integer       // number of dec places in tin depth
tin_n_depth_textstyle       textstyle     // textstyle of the tin depths
tin_n_depth_colour          colour        // colour for depths
tin_n_depth_size            mm            // size for depths
tin_n_depth_y_pos           mm            // if set, the height in mm above the
                                          // bottom of all the boxes that the
                                          // tin depth text is drawn.
                                          // If not set, then the text is placed at a
                                          // height that puts it inside the default
                                          // box for the tin depths.
```

Offset Model Titles and Heights

For 12d Model V3.2, Each string in the offset model given in the **section long plot** panel can be automatically projected onto the primary string and drawn on the long section plot.

For **12d Model V4.0**, the idea of an offset model has been modified to allow strings to be projected onto the primary string to be specified by model and name, rather than just projecting all the strings in a given model. The string can also be labelled with either the string name, its model name or both.

The string is specified by

```
offset_n_mask               "model_name->string_name"
```

For each n, a set of plot parameters determine if the string's heights and/or depths from the primary string are labelled in the boxes area.

If only the *string_name* is given, then the model given in the **Offset model** field of the **Section Long Plot** panel.

If there is **more than one string** with the given model and name, then it will be considered to be one string and at any primary string chainage, the closest of the strings will be the part used for projecting. For example, strings of the same name on either side of a road intersection will be considered to be the one string for projecting.

If any *offset_n_mask* is used, then the Offset option expects all the strings to be specified by an offset_n_mask. Then if no mask exists for any value of n, then that parameter set is ignored and no string projected.

If no offset masks are used (that is, no *offset_n_mask*'s are used), then the Offset option works as it did in 12d Model V3.0. That is, plot parameters determine whether each string in the offset model is drawn, and if the string's heights and depths from the primary string are labelled in the boxes area. The order that the strings are then plotted and labelled is the same as the order of the strings in the Offset model.

For the following sets of parameters, n takes the value 1 to 100 and specifies that the parameter set applies to the nth string given by *offset_n_mask* or, if no offset masks are given, the nth string in the *Offset model*.

```
offset_n_mask               "model_name->string_name"

offset_n_draw_mode          1             // draw the nth offset string (default)
```

	0	// don't draw the nth offset string

offset_n_lw	*value*	// default 1000. Distance to search to the left
		// of the primary string for the offset string.
		// If 0, don't search to the left.

offset_n_rw	*value*	// default 1000. Distance to search to the right
		// of the primary string for the offset string.
		// If 0, don't search to the right.

offset_n_draw_colour	*colour*	// colour to draw the nth offset string. // If omitted, then use the string colour
offset_n_label	*1*	//label the string's heights (default)
	0	// don't label the string's heights

The default title for each set each parameter set, is the string name but this can be changed to just the model name or the model and string name:

offset_title_mode	*0*	// label with string name (default)
	1	// label with model name
	2	// label with model->string as name

or, for any set, the *offset_title_mode* can be replaced by two lines of user defined title:

offset_n_title	*text*	// first line of title
offset_n_title_2	*text*	// second line of title
offset_n_title_textstyle	*textstyle*	// textstyle of the offset titles

The rest of the parameters for the set are:

offset_n_title_colour	*colour*	// colour for titles of heights
offset_n_title_size	*size*	// colour for titles of heights

offset_n_title_y_pos	*mm*	// if set, the height in mm above the
		// bottom of all the boxes that the
		// offset title text is drawn.
		// If not set, then the text is placed at a
		// height that puts it inside the default
		// box for the offset heights.

offset_n_decimals	*integer*	// number of dec places in offset height
offset_n_textstyle	*textstyle*	// textstyle of the offset heights
offset_n_colour	*colour*	// colour for heights
offset_n_size	*size*	// size for heights

offset_n_y_pos	*mm*	// if set, the height in mm above the
		// bottom of all the boxes that the
		// offset height text is drawn.
		// If not set, then the text is placed at a
		// height that puts it inside the default
		// box for the offset heights.

offset_n_depth_label	1/0	// 1 to label, 0 don't label depths
offset_n_depth_title	*text*	// first line of offset depth title
offset_n_depth_title_2	*text*	// second line of offset depth title
offset_n_depth_title_textstyle	*textstyle*	// textstyle of the offset depth titles
offset_n_depth_title_colour	*colour*	// colour for titles of depths
offset_n_depth_title_size	*mm*	// size for titles of depths

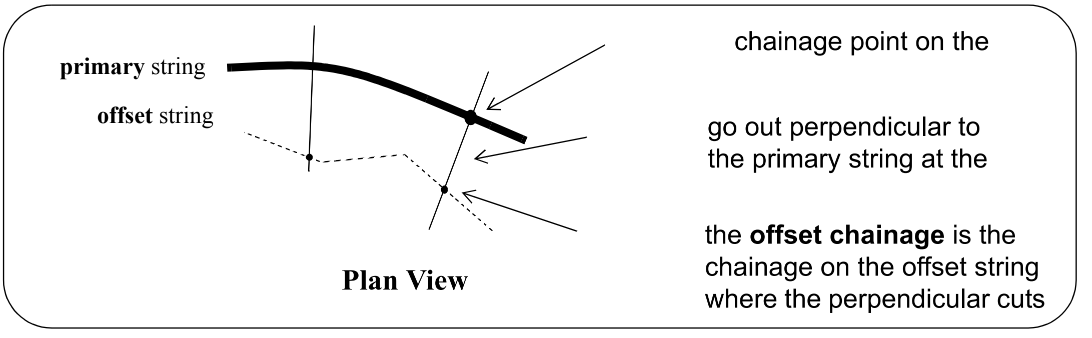
offset_n_depth_title_y_pos	*mm*	// if set, the height in mm above the
		// bottom of all the boxes that the
		// offset depth title text is drawn.
		// If not set, then the text is placed at a

		// height that puts it inside the default
		// box for the offset depths.
offset_n_depth_decimals	*integer*	// number of dec places in offset depth
offset_n_depth_textstyle	*textstyle*	// textstyle of the offset depths
offset_n_depth_colour	*colour*	// colour for depths
offset_n_size	*size*	// size for depths
offset_n_depth_y_pos	*mm*	// if set, the height in mm above the
		// bottom of all the boxes that the
		// offset depth text is drawn.
		// If not set, then the text is placed at a
		// height that puts it inside the default
		// box for the offset depths.

Chainages of Offset Strings - Offset Chainages

For a string other than the primary string, it is possible to label the chainages that are obtained *from the other string* by going out perpendicular to the primary string at a chainage, cutting the other string and then calculating the *chainage from the other string* at the cut point.

The other string is known as an ***offset*** string and the chainage from the offset string corresponding to a primary chainage is called the *offset chainage*.



Up to twenty sets of offset chainages can be labelled - each is defined by its own set of parameters:

Note no box is automatically created for the offset chainages. An empty box must be defined and the appropriate values set for *chainage_n_title_y_pos* and *chainage_n_y_pos*

chainage_n_offset_mode	*tick box*	// *if ticked, use nth offset string* for calculations
chainage_n_offset_string	"*model_name->string_name*"	// string to calculate offset chainages for
chainage_n_title	*text*	// first line of offset chainage title
chainage_n_title_2	*text*	// second line of offset chainage title
chainage_n_title_textstyle	*textstyle*	// textstyle of the offset chainage titles
chainage_n_title_colour	*colour*	// offset chainage title colour
chainage_n_title_size	*mm*	// offset chainage title size

chainage_n_title_y_pos	*mm*	// the height in mm above the
		// bottom of all the boxes that the
		// offset chainage title text is drawn.

chainage_n_decimals	*integer*	// number of decim places for offset chainages
chainage_n_textstyle	*textstyle*	// textstyle of the offset chainage values
chainage_n_colour	colour	// colour of offset chainages
chainage_n_size	mm	// size of offset chainages

chainage_n_y_pos	*mm*	// the height in mm above the
		// bottom of all the boxes that the
		// chainage values text is drawn.

Example of Offset Chainage and Titles

chainage_1__offset_string	“Design->RKERB”	
chainage_1__title	“RIGHT KERB”	
chainage_1_title_2	“CHAINAGE”	
chainage_1_title_colour	“yellow”	
chainage_1_colour	“grey”	// colour of offset chainage text
chainage_1_decimals	-3	// 3 decimal places - leave trailing zeros

Definition of Depth to a Tin or to Offset Strings

The depth from the primary string to a tin or to a string from the offset model, at a particular chainage is defined as

depth = tin height value - height of the primary string

or

depth = offset string height value - height of the primary string

That is, the depth that the primary string is **below** the tin or a string from the offset model.

Before plotting, the value of depth is multiplied by either the *depth_positive_factor* or *depth_negative_factor*.

if (depth >= 0) plotted_depth_value = depth * depth_positive_factor

if (depth < 0) plotted_depth_value = depth * depth_negative_factor

Hence the definition of depth can be modified by the parameters:

```
depth_positive_factor    value    // multiplier for positive depths
depth_negative_factor    value    // multiplier for negative depths
```

For example, if the opposite sign is required for depth, that is,

depth = height of the primary string - tin/offset string height value

simply set

```
depth_positive_factor    -1
depth_negative_factor    -1
```

Super-Elevation Diagram

The optional *super-elevation diagram* draws the values of the cross-fall (x-fall) between two strings using the primary string as the reference string. The cross-fall at a given chainage on the primary string is calculated by sectioning perpendicular to the reference string at that chainage and cutting the two strings. The cross-fall is defined as the cross-fall between the two cuts points on the strings.

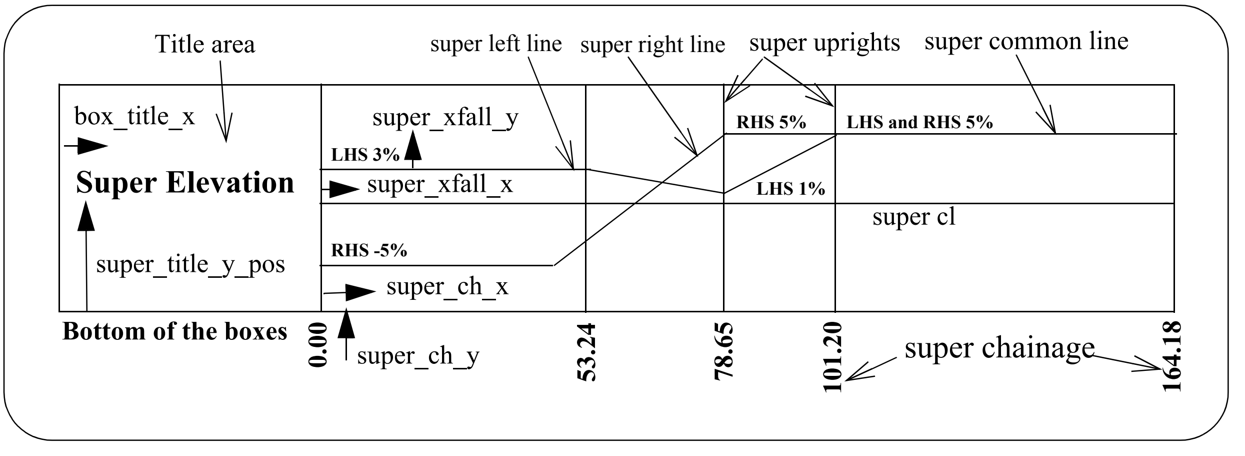
The diagram has levels for the cross fall for a pair of strings on the left of the primary string, and a pair of strings on the right of the primary string. The diagram also has uprights in the super-elevation box with chainage values at the change of super values.

For the left hand side, the cross-fall is calculated at right angles to the primary string between the user given left hinge string and the left edge string.

There are plot parameters to control all aspects of the super diagram.

super_draw_mode	*0*	// default, don't draw a super diagram
	1	// draw super using *super_sample_interva*l
	2	// sample super using chainages of x-sections
	3	// sample super using chainages from a string
super_sample_interval	*value*	// if super_draw_mode = 1, default 20
		// chainage distance to sample x-fall
super_sample_name	*model_name*	// if super_draw_mode = 2
	model_name->string_name	// if super_draw_mode = 3
super_tolerance	*value*	// If super_draw_mode is 1, the change
		// of super is greater than super_tolerance,
		// draw uprights
super_title	*text*	// first line of super title
		// def *Super elevation*
super_title_textstyle	*textstyle*	// textstyle of the super titles
super_title_colour	*colour*	// super title colour, def box_colour
super_title_size	*mm*	// super title size, def label_text_size
super_title_y_pos	*mm*	// if set, the height in mm above the
		// bottom of all the boxes that the
		// super title text is drawn.
		// If not set, then the text is placed at a
		// height that puts it inside the default
		// box for the super.
super_xfall_textstyle	*textstyle*	// textstyle of the super x-fall labels
super_xfall_colour	*colour*	// super x-fall colour, def box_colour
super_xfall_size	*mm*	// super x-fall size, def label_text_size
super_xfall_x	*mm*	// def 0
super_xfall_y	*mm*	// def 0.5
super_ch_textstyle	*textstyle*	// textstyle of the super chainages
super_ch_colour	*colour*	// super chainage colour, def box_colour
super_ch_size	*mm*	// super chainage size, def label_text_size
super_ch_x	*mm*	// def 0
super_ch_y	*mm*	// def -1
super_ch_decimals	integer	// number decimal places in chainage
		// def number_of_decimals
super_hinge_name_left	*model_name->string_name*	// left hinge string
super_edge_name_left	*model_name->string_name*	// left edge string
super_left_linestyle	*linestyle*	// default DASHED
super_left_line_colour	*colour*	// default box_colour
super_left_decimals	*integer*	// number decimal places in left x-fall
		//def 0
super_left_pre	*text*	// text before left x-fall value
super_left_post	*text*	// text after left x-fall value
super_hinge_name_right	*model_name->string_name*	// right hinge string

super_edge_name_right	*model_name->string_name*	// right edge string
super_right_linestyle	*linestyle*	// default DASHED2
super_right_line_colour	*colour*	// default box_colour
super_right_decimals	*integer*	// number decimal places in right x-fall //def 0
super_right_pre	*text*	// text before right x-fall value
super_right_post	*text*	// text after right x-fall value
super_cl_linestyle	*linestyle*	// default 1
super_cl_line_colour	*colour*	// default box_colour
super_common_pre	*text*	// text before common x-fall value
super_common_post	*text*	// text after common x-fall value
super_common_linestyle	*linestyle*	// default 1
super_common_line_colour	*colour*	// default box_colour
super_common_decimals	*integer*	// number decimal places in common xfall //def 0
super_upright_mode	*0*	// uprights for left string only (default)
	1	// right
	2	// left and right
super_upright_colour	*colour*	// default box_colour



Example of Super

super_hinge_name_left	"super->lkerb"
super_edge_name_left	"super->lshoulder"
super_hinge_name_right	"super->rkerb"
super_edge_name_right	"super->rshoulder"
super_draw_mode	3
super_sample_interval	10
super_sample_name	"super->lkerb"
super_title_size	5
super_title_colour	ORANGE
super_title_textstyle	GOTHIC
super_title_y_pos	10
super_title	"Super EL"
super_xfall_size	1.5
super_xfall_colour	GREY
super_xfall_textstyle	HELV

```
// super_xfall_y                       0.5
// super_xfall_x                       2

 super_ch_size                         3
 super_ch_colour                       YELLOW
 super_ch_textstyle                    ISO
 super_ch_y                            -2
 super_ch_x                            -0.5
 super_ch_decimals                     -2

 super_left_linestyle                  DIVIDE
 super_right_linestyle                 PHANTOM2
 super_cl_linestyle                     PHANTOM
// super_common_linestyle              thick

 super_left_line_colour                RED
 super_right_line_colour               YELLOW
 super_cl_line_colour                  CYAN
 super_common_line_colour              MAGENTA
 super_upright_colour                  BLUE

// super_common_pre                    "Pre "
// super_common_post                   "% Post"

// super_common_decimals               -1
 super_right_decimals                  1
 super_left_decimals                   1
```

Volume Cut and Fill Values

The values of calculated cut and fill volumes (cut and fill earth works) can be read from a file and then interpolated to produce cut and fill volumes for a given interval along the primary string.

There are plot parameters to control all aspects of the volume diagram.

volume_draw_mode	*0*	// default, don't draw a volume diagram
	1	// draw a volume diagram
volume_text_centre_mode	*0*	// default, the text of cut and fill are labelled // along the uprights
	1	// the text are centred in between the uprights
volume_sample_interval	*value*	// interval to display volumes, default 20
volume_file_name	*file_name*	// the volumes report file
volume_y_pos	*mm*	// if set, the volume box is positioned at this // height from the bottom of the first box
volume_box_size	mm	// if set, the uprights and volume and titles // are plotted in accordance with this heights
volume_title	*text*	// first line of volume title // default *Earthworks*
volume_title_textstyle	*textstyle*	// textstyle of the volume titles
volume_title_colour	*colour*	// volume title colour, def box_colour
volume_title_size	*mm*	// volume title size, def label_text_size

volume_title_y_pos	*mm*	// if set, the height in mm above the
		// bottom of all the boxes that the
		// volume title text is drawn.
		// If not set, then the text is placed at a
		// height that puts it inside the default
		// box for the volumes.

volume_cut_title	*text*	// first line of volume sub title // default *cut*
volume_cut_title_textstyle	*textstyle*	// textstyle of the cut volume titles
volume_cut_title_colour	*colour*	// cut volume title colour, def box_colour
volume_cut_title_size	*mm*	// cut volume title size, def label_text_size

volume_cut_title_y	*mm*	// if set, the height in mm above the
		// default position.
		// If not set, then the text is placed at a
		// height that puts it inside the default
		// box for the cut volumes.

volume_fill_title	*text*	// first line of volume sub title // default *fill*
volume_fill_title_textstyle	*textstyle*	// textstyle of the fill volume titles
volume_fill_title_colour	*colour*	// fill volume title colour, def box_colour
volume_fill_title_size	*mm*	// fill volume title size, def label_text_size

volume_fill_title_y	*mm*	// if set, the height in mm above the
		// default position.
		// If not set, then the text is placed at a
		// height that puts it inside the default
		// box for the fill volumes.

volume_cut_textstyle	*textstyle*	// textstyle of the cut values
volume_cut_text_colour	*colour*	// cut value colour, def box_colour
volume_cut_text_size	*mm*	// cut value size, def label_text_size

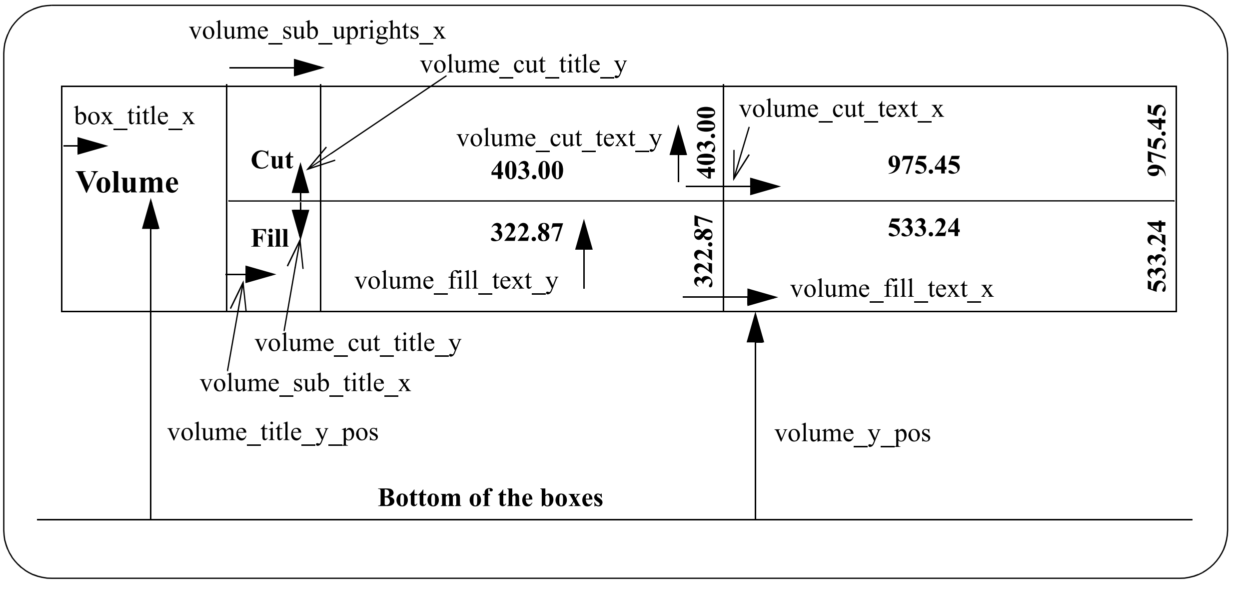
volume_cut_text_x	*mm*	// the x distance to move the cut text from the
		// default cut text position, def 0, it is not used
		// if volume_text_centre_mode=1

volume_cut_text_y	*mm*	// the y distance to move the cut text from the // default cut text position, def 0
volume_cut_decimals	*integer*	// number decimal places in cut values //def number_of_decimals

volume_fill_textstyle	*textstyle*	// textstyle of the fill values
volume_fill_text_colour	*colour*	// fill value colour, def box_colour
volume_fill_text_size	*mm*	// fill value size, def label_text_size

volume_fill_text_x	*mm*	// the x distance to move the fill text from the
		// default fill text position, def 0, it is not used
		// if volume_text_centre_mode=1

volume_fill_text_y	*mm*	// the y distance to move the fill text from the // default fill text position, def 0
volume_fill_decimals	*integer*	// number decimal places in fill values //def number_of_decimals
volume_cl_linestyle	*linestyle*	// linestyle of horizontal cut/fill dividing line
volume_cl_line_colour	*colour*	// colour of horizontal cut/fill dividing line
volume_uprights_draw_mode	*0*	// default, don't draw cut/fill uprights
	1	// draw cut/fill uprights
volume_uprights_line_colour	*colour*	// colour of uprights
volume_sub_upright_colour	*colour*	// colour of sub uprights line
volume_sub_uprights_x	*mm*	// the distance to move from the default // sub uprights position, def 0
volume_sub_title_x	*mm*	// the distance to move the sub title text // from the volume sub uprights



Example of Volumes

volume_draw_mode	1
volume_sample_interval	25
volume_file_name	"volume.rpt"
volume_text_centre_mode	0
volume_y_pos	150
volume_box_size	40
volume_title	"Volumes"
volume_title_size	5
volume_title_colour	grey

```
//          volume_title_textstyle
            volume_title_y_pos                165

            volume_cut_title                  "cut"
//          volume_cut_title_size
//          volume_cut_title_colour
//          volume_cut_title_textstyle
            volume_cut_title_y                3

            volume_fill_title                 "fill"
//          volume_fill_title_text_size
//          volume_fill_title_text_colour
//          volume_fill_title_textstyle
            volume_fill_title_y               -3

            volume_cut_text_size              3
//          volume_cut_text_colour
//          volume_cut_textstyle
            volume_cut_text_y                 5
//          volume_cut_text_x
            volume_cut_decimals               0
            volume_fill_text_size             3
//          volume_fill_text_colour
//          volume_fill_textstyle
            volume_fill_text_y                245
//          volume_fill_text_x
            volume_fill_decimals              0

            volume_cl_linestyle               "PHANTOM"
            volume_cl_line_colour             red

            volume_uprights_draw_mode         1
            volume_uprights_line_colour       "purple"

            volume_sub_upright_colour         23
            volume_sub_uprights_x             10

            volume_sub_title_x                3
```

X,Y Values

The values of the X and Y coordinate for selected chainages can be labelled as separate boxes. The chainages specified are limited to the base set of chainages already defined in the chainage selection and staggering section. i.e. an upright must exist for the X and Y value to be labelled. Not all of the uprights have to be labelled, just those specified in this section.

There are plot parameters to control all aspects of the X and Y labelling.

Chainages are used for positioning X and Y labels.

The chainages for the long section plot relate to the primary string and are controlled by a set of parameters.

xy_order	0	// 0 = X on bottom, Y on top
	1	// 1 = Y on bottom, X on top
xy_chord_arc	0	// don't use chord/arc chainages
	1	// include chainages for chord/arc
xy_start_chainage	*value*	// start chainage of labelling range
xy_end_chainage	*value*	// end chainage of labelling range
xy_interval	*value*	// regular interval (0 = no regulars)
xy_label_hcp	0/1	// 1 = include tangents, spirals
xy_label_hip	0/1	// 1 = include hip points
xy_label_vip	0/1	// 1 = include vip points
xy_label_vtp	0/1	// 1 = include tangent points
xy_label_crest	0/1	// 1 = include crest points
xy_label_sag	0/1	// 1 = include sag points
xy_label_grade_change	0/1	// 1 = include change of vertical grade
xy_special_n_file	*filename*	// n = 1 to 20 - include chainages from // the file (one chainage per line)
xy_label_tolerance	*value*	// > 0 use as weeding tolerance // <= 0, don't weed

For X parameters:

x_label	*0/1*	// 0 = Don't label X values/titles // 1 = Label X values/titles
x_title	*text*	// first line of X title // (default "Easting")
x_title_2	*text*	// second line of X title
x_title_textstyle	*textstyle*	// textstyle of the X titles
x_title_colour	*colour*	// default X title colour
x_title_size	*mm*	// default X title size

x_title_y_pos	*mm*	// if set, the height in mm above the
		// bottom of all the boxes that the
		// X title text is drawn.
		// If not set, then the text is placed at a
		// height that puts it inside the default
		// box for the X.

x_label_decimals	*integer*	// number of decim places for X values
x_label_textstyle	*textstyle*	// textstyle of the X values
x_label_colour	colour	// colour of X values, def is box_colour
x_label_size	mm	// size of X values, def is box_text_size

x_label_y_pos	*mm*	// if set, the height in mm above the
		// bottom of all the boxes that the
		// X value text is drawn.
		// If not set, then the text is placed at a
		// height that puts it inside the default
		// box for the X value.

For Y parameters:

y_label	*0/1*	// 0 = Don't label Y values/titles // 1 = Label Y values/titles
y_title	*text*	// first line of Y title // (default "Easting")
y_title_2	*text*	// second line of Y title
y_title_textstyle	*textstyle*	// textstyle of the Y titles
y_title_colour	*colour*	// default Y title colour
y_title_size	*mm*	// default Y title size

y_title_y_pos	*mm*	// if set, the height in mm above the
		// bottom of all the boxes that the
		// Y title text is drawn.
		// If not set, then the text is placed at a
		// height that puts it inside the default
		// box for the Y.

y_label_decimals	*integer*	// number of decim places for Y values
y_label_textstyle	*textstyle*	// textstyle of the Y values
y_label_colour	colour	// colour of Y values, def is box_colour
y_label_size	mm	// size of Y values, def is box_text_size

y_label_y_pos	*mm*	// if set, the height in mm above the
		// bottom of all the boxes that the
		// Y value text is drawn.
		// If not set, then the text is placed at a
		// height that puts it inside the default
		// box for the Y value.

Please continue to the next section 44.3.5 Chainage Selection and Staggering.

44.3.5 Chainage Selection and Staggering

Chainages are used for positioning height labels, uprights (leader lines) and bubbles.

The chainages for the long section plot relate to the primary string and are controlled by a set of parameters.

chord_arc	0	// don't use chord/arc chainages
	1	// include chainages for chord/arc
chainage_interval	*value*	// regular interval (0 = no regulars)
chainage_label_ends	0/1	// 1 = include start and end chainages
chainage_label_hcp	0/1	// 1 = include tangents, spirals
chainage_label_hip	0/1	// 1 = include hip points
chainage_label_vip	0/1	// 1 = include vip points
chainage_label_vtp	0/1	// 1 = include tangent points
chainage_label_crest	0/1	// 1 = include crest points
chainage_label_sag	0/1	// 1 = include sag points
chainage_label_grade_change	0/1	// 1 = include change of vertical grade
chainage_special_n_file	*filename*	// n = 1 to 20 - include chainages from // the file (one chainage per line)
chainage_label_tolerance	*value*	// > 0 use as weeding tolerance // <= 0, don't weed
chainage_merge_bubbles	0/1	// 1 = merge in the bubble chainages
chainage_merge_tolerance	value	// > 0, weeding tol after bubble merge // <= 0, don't weed after merge

Default values

chainage_label_tolerance = 1.0 exp -4 (world units)
chainage_merge_tolerance = 1.0 exp -8 (world units)

If a tolerance is zero or negative, no weeding is performed.

The format of a chainage special file is simply a list of chainage values, one value per lines. Blank lines in the file are ignored and anything on a line after a *//* is a comment.

If the real chainage position is used for the horizontal position of the chainage/height/depth text, text over writing can easily occur.

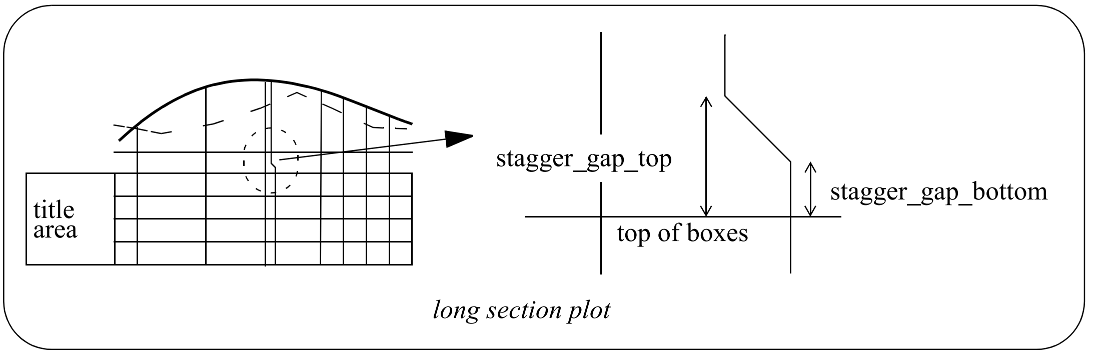
To prevent over writing, the text can be **staggered**.

If the ***stagger_mode*** parameter is set to 1, the text position is adjusted so that the text does not over write.

The real chainage position is then indicated by the chainage markers which are drawn at the top of the text boxes from the staggered text position back to the actual chainage position of the text.

stagger_mode	0	// no staggering, allow over writing
	1	// stagger text to prevent over writing
stagger_gap_top	*mm*	// distance from boxes to top of stagger
stagger_gap_bottom	*mm*	// distance from boxes to bottom of // stagger
stagger_gap_factor	*value*	// distance between staggers is // box_text_size * stagger_gap_factor

When staggering occurs, it is possible for the heights area to be longer than the graph area.



Please continue to the next section 44.3.6 Uprights.

44.3.6 Uprights

Uprights, or leader lines, can be drawn from the top of the staggers to the strings drawn on the plot.

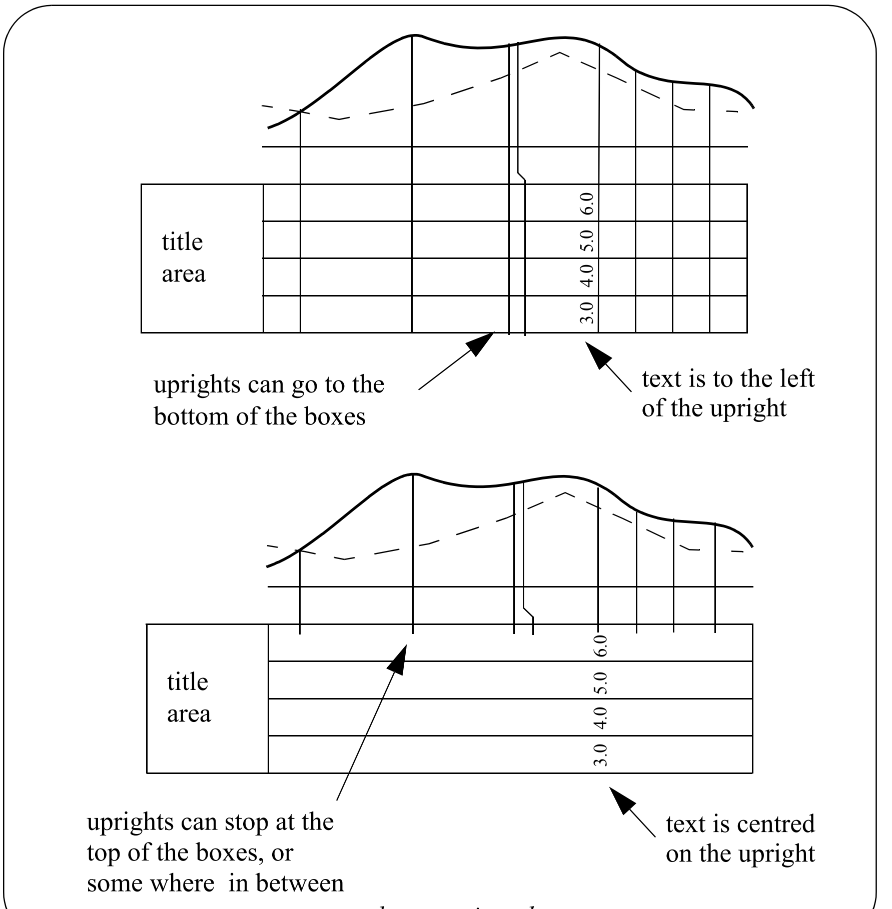
The height of the uprights is given by the uprights_draw_mode:

uprights_draw_mode	0	// none
	1	// to maximum string height at that // chainage
	2	// ticks, to stagger height
	3	// to uprights_y above the boxes
	100	// to the primary string
	101-500	// to tin1 or tin2 etc.
	501-900	// to offset1 or offset2 etc.
uprights_y	*mm*	// distance to draw the uprights for // mode 3
uprights_colour	*colour*	// uprights colour (default *boxes_colour*)

The uprights can go below the top of the boxes.

uprights_bottom_mode	*0*	// stop at top of boxes (default)
	1	// draw to bottom of boxes
	2	// draw to uprights_bottom_y below the // top of the boxes
	3	// draw to uprights_bottom_y above the // bottom of the boxes
	4	// ticks at chainage
uprights_bottom_y	*mm*	//distance
uprights_text_offset_factor	*value*	// move the text by this factor*size

When uprights go below the top of the boxes, the height and offset text is moved to the left so that the upright does not go through the text. The left hand side of the heights boxes also moves to the left to leave room for the height text.



long section plot

Please continue to the next section 44.3.7 Datum Area.

44.3.7 Datum Area

The **datum area** is the region between the boxes area and the graph area.

The **datum line** is positioned the distance ***datum_below_gap*** above the top of the boxes area and the graph area is positioned the distance ***datum_above_gap*** above the datum line.

Hence the graph area is distance (datum_below_gap + datum_above_gap) above the top of the boxes area.

datum_above_gap	*mm*	// distance from datum line to the // bottom of the graph area
datum_below_gap	*mm*	// distance from datum line to the top of // the boxes
datum_linestyle	*linestyle*	// linestyle for datum line (default solid)

The *datum_below_gap* and *datum_above_gap* can be zero or positive.

A value for the datum can be specified by the parameter ***datum_value*** or if the *datum_value* is not set, a datum value is automatically calculated for the long section using a roundoff specified by the user (default 1.0).

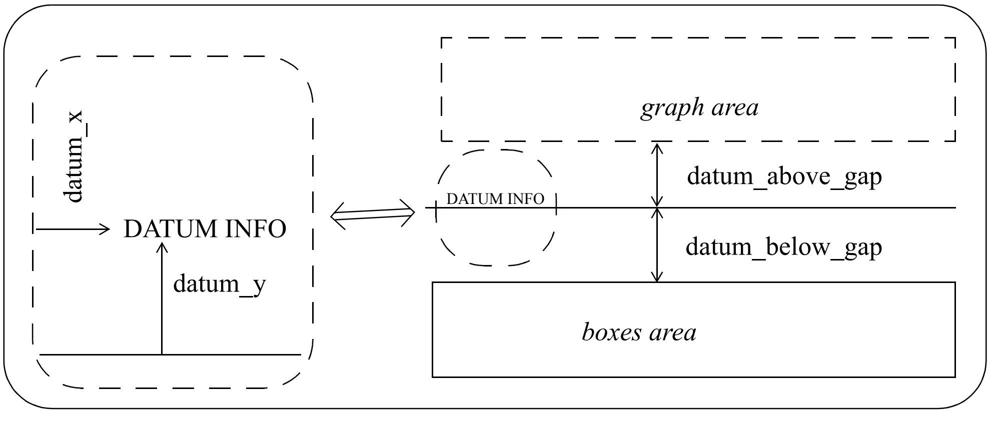
datum_value	*value*	// if set, the value to use for datum.
		//If not set, the datum is automatically
		// calculated (using datum_roundoff)

datum_roundoff	*1.0*	// value to roundoff the datum value to // e.g. 0.5, 0.2, 1.0 (default 1.0)

datum_decimals	*integer*	// number of decimal places to display
		// the datum value (default 1).
		// If > 0, trailing zeros are removed after
		// the decimal point.
		// If <0, the absolute value is taken as the
		// number of decimal places to report
		// i.e. no trailing zeros are removed

datum_name	*text*	// text to write before the datum value
datum_textstyle	*textstyle*	// textstyle for the datum text
datum_text_size	*mm*	// size of datum text and value
datum_colour	*colour*	// colour of the datum text and line
datum_x	*mm*	// distance to move the datum text // along the datum line

datum_y	*mm*	// distance to raise the datum text
		// above the datum line (used to be
		// called datum_offset)



Please continue to the next section 44.3.8 Graph Area.

44.3.8 Graph Area

The **graph area** is the area where the actual plots of the strings are drawn.

The string used to define the horizontal position and chainage of the plot is called the **primary string** and is either

(a) the string being profiled on the given section view
or

(b) the string given by the parameter

string_to_plot	*"model->string_name"*	// name of string to profile

When the primary string is being written out to a plot parameter file by 12d Model, a *string_to_plot* parameter and special parameters recording the internal id's for the string and it's model are all written out.

primary_model_id	*integer*	// internal ***12d Model*** parameter
primary_string_id	*integer*	// internal***12d Model*** parameter

The **length** of the graph area is determined by the length of the primary string being plotted (restricted to the specified start and end chainages) and the horizontal scale (scale) given by parameters or in the **section long plot** panel, and the vertical exaggeration given by the section view or a parameter.

view_name	*text*	// default is **section view** in panel.
vertical_exaggeration	*value*	// default is vertical exag for the view
scale	*value*	// 1:value - horizontal scale, default is
		// **scale 1:** in panel
start_chainage	*value*	// " " means use start of primary string
end_chainage	*value*	// " " means use end of primary string

The vertical scale is determined by the horizontal scale and the vertical exaggeration.

The **height** of the graph area is determined by the vertical scale (given by the horizontal scale and the vertical exaggeration) and the minimum and maximum values of the data being plotted. Hence the graph height is a calculated rather than a given value.

The types of strings that can be drawn in the graph area of a long section plot are:

(a) primary string — the string being profiled (usually the design string).

(b) tins — sections of the primary string through any tins either in models on the section view or in corridor models.

(c) offset strings — strings in the **offset model** that are projected back onto the primary string

(d) services — parts of strings (from either models on the section view or in corridor models) that cut the defined corridor.

The **colour** of the strings in the plot is the actual string colour for cases (a), (c) and (d), and the colour of the tin used for the section in case (b).

Although all the strings are plotted, the plot parameter file can be used to select which ones are labelled with heights and/or depths.

Primary String (Design String)

The primary string (the design string) is used to define

(a) the design long section

(b) the chainage positions for labelling heights and drawing uprights

(c) the section line used for sectioning through tins

(d) the section line for defining the corridor for services

Although the primary string is used to set up most of the information for the long section plot, it doesn't have to be drawn on the long section.

The drawing or not drawing of the primary string on each cross section plot is controlled by the parameter *primary_string* which was described earlier.

The **colour** of the primary string in the plot is the actual primary string colour.

Tins

A section along the primary string through each tin on the specified section view is automatically drawn on the cross section plot.

The **colour** and **linestyle** of the tin section in the plot is the actual tin colour and tin linestyle.

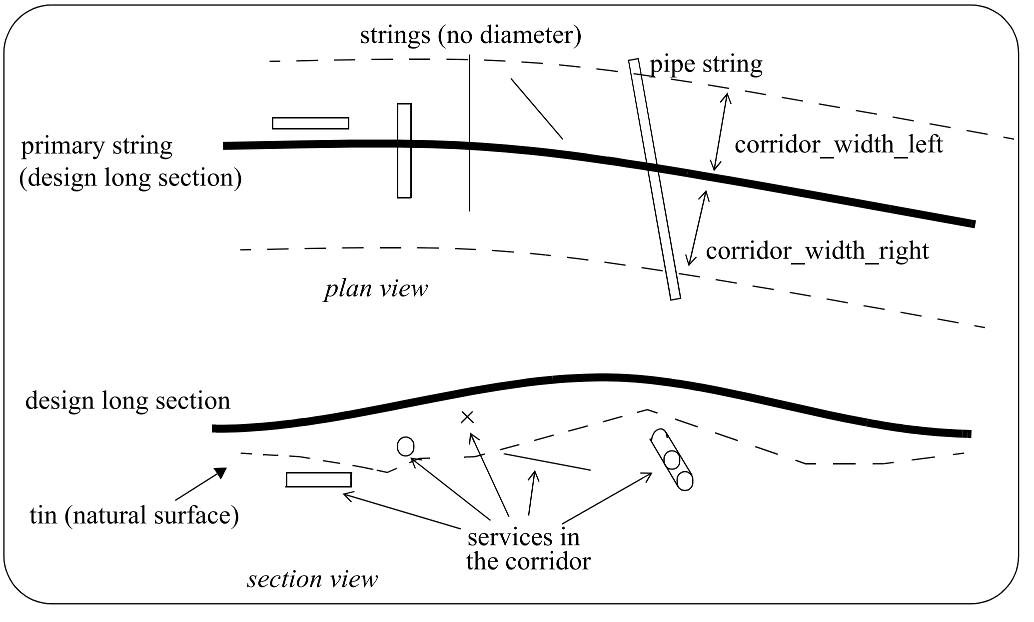
Whether the tin heights are labelled or not in the boxes area is controlled by parameters and has been described earlier

Corridor and Services

A corridor around the primary string is defined by giving a left and right corridor width.

Any string in a model added to the section view is checked to see if it appears in the corridor, and if it does, then it is drawn on the long section plot.

To be drawn, strings do not have to cross the primary string, but just be in the corridor.



The corridor defining parameters are

corridor_width_left	*world-units*	// corridor left and right widths
corridor_width_right	*world-units*	// defaults are the section view values
corridor_overlap_left	*world-units*	// corridor left and right overlaps.
corridor_overlap_right	*world-units*	// defaults are the section view values

corridor_chord_arc	*world-units*	// chord-arc tolerance used to
		// approximate arcs in the corridor.
		// default is the section view value

The models containing tins for sectioning and strings for services are taken from the section view or given by the parameters

corridor_model_n	model	// n = 1, 2 ... 100
		// models containing tins and service
		// strings to be drawn on the view.

If any *corridor_model_n* parameters are defined, then only the models given by the parameters are used. If no *corridor_model_n* parameters are set, then the models added to the given section view are used.

That is, either the *corridor_model_n* parameters are used or if none exist, then the models added to the section view are used for tins and service strings.

The **colour** and **diameter** of the service strings drawn in the plot are the actual strings colour and diameter.

The graph area sits on top of the boxes and datum areas, so there may not be enough room left on the sheet for the full plot height. In this case, the plot will be truncated at the top of the allowed graph area.

Please continue to the next section 44.3.9 Bubbles Definitions.

44.3.9 Bubbles Definitions

Circles with the string name and a unique number (**bubbles**) can be drawn on the long section plot. Bubbles are normally used for lip profiles.

The chainages used for the bubbles are given by a set of parameters similar to the chainage parameters. The resulting set of bubbles are sequentially numbered (starting with one) in chainage order.

Although many bubbles can be defined by the bubble parameters, a bubble is only drawn on the plot if there is a labelled chainage to draw it above.

Hence not all bubbles given by the bubble chainage parameters are drawn but for the ones that are drawn, the bubble number is taken from the full bubble set.

chainage_bubbles	0	// don't drawn bubbles
	1	// draw bubbles
bubble_radius	*mm*	// radius of the bubbles
bubble_colour	*colour*	// colour for the bubbles
bubble_text_string_name_mode*0*		// do not label with string name
	1	// label string name
	2	// *label* with *model->string name*
bubble_pre_text	*text*	
bubble_post_text	*text*	
bubble_textstyle	*textstyle*	
bubble_text_size	*mm*	
bubble_text_colour	*colour*	
bubble_text_offset	*mm*	
bubble_upright_distance	*mm*	// distance bubbles are above boxes/uprights
bubble_mode	0	// bubble_upright_distance is above boxes
	1	// bubble_upright_distance is above // uprights
bubble_draw_upright	0/1	// 1 = draw extra upright if // bubble_mode = 1
bubble_start_chainage	*value*	// start chainage for bubbles
bubble_end_chainage	*value*	// end chainage for bubbles
bubble_chord_arc	0/1	// 1 = use chord-arc chainages
bubble_interval	*value*	// include regular interval // (0 = no regulars)
bubble_label_ends	0/1	// 1 = include start and end chainages
bubble_label_hcp	0/1	// 1 = include tangents, spirals
bubble_label_hip	0/1	// 1 = include hip points
bubble_label_vip	0/1	// 1 = include vip points
bubble_label_vtp	0/1	// 1 = include tangent points
bubble_label_crest	0/1	// 1 = include crests
bubble_label_sag	0/1	// 1 = include sags
bubble_label_grade_change	0/1	// 1 = include change of grade
bubble_special_n_file	*filename*	// n = 1 to 20 include chainages from the // file
bubble_label_tolerance	*value*	// > 0 use as weeding tolerance // <= 0, don't weed

Defaults

bubble_label_tolerance = 1.0 exp -4 (world units)

If a tolerance is zero or negative, no weeding is performed.

The format of a bubble special file is simply a list of chainage values, one value per lines. Blank lines in the file are ignored and anything on a line after a ***//*** is a comment.

Please continue to the next section 44.3.10 Quick Horizontal Geometry Labelling.

44.3.10 Quick Horizontal Geometry Labelling

The standard horizontal geometry arrows can be drawn at a given distance above the top of the boxes area.

horizontal_geometry_y‘	*mm*	// dist above boxes
		// 0 = don't draw
horizontal_geometry_arrow_text_colour	*colour*	
horizontal_geometry_arrow_textstyle	*textstyle*	
horizontal_geometry_arrow_text_size	*mm*	
horizontal_geometry_arrow_colour	*colour*	
horizontal_geometry_arrow_height	*mm*	
horizontal_geometry_label_text	*text*	
horizontal_geometry_label_textstyle	*textstyle*	
horizontal_geometry_label_text_colour	*colour*	
horizontal_geometry_label_text_size	*mm*	
horizontal_geometry_label_decimals	*number*	

Please continue to the next section 44.3.11 Extensive Horizontal Geometry Labelling.

44.3.11 Extensive Horizontal Geometry Labelling

For complicated horizontal geometry labelling, there are sets of horizontal geometry labelling parameters which give tight control over the position and types of labels.

It is also possible to label the horizontal geometry of ***alignment*** strings other than the primary string. To plot such a string on the same plot, the chainage position of the horizontal geometry for the non-primary alignment strings is *projected* onto the primary string to give a primary string chainage for plotting. The values of the horizontal geometry (such as radius and spiral length) that are plotted are taken from the other string. Independently graded offset strings (such as a left and right kerbs) are the type of additional alignment strings that may need to be plotted on the same long section plot as the reference string (primary string).

For plotting horizontal geometry, the user can give up to twenty sets of these labels and they can be used to label spirals, curves and tangent information for the primary string and/or additional alignment strings.

Each label set consists of three parts:

(a) a text label on the left hand side of the plot
(b) an arrow
(c) text on the arrows.

For the following parameters, n takes the value 1 to 20 and specifies the nth parameter set.

h_g_n_type	0	// label spirals
	1	// label horizontal curves
	2	// label horizontal tangents

If *h_g_n_type* is missing, then the set is ignored.

h_g_n_value_mode		
for spiral labelling	0	// nothing
	1	// length
for curve labelling	0	// nothing
	1	// length
	2	// radius
for tangent labelling	0	// nothing
	1	// length
for spiral labelling	0	// nothing
	1	// length
	2	// radius*length
h_g_n_label_y	*mm*	// distance of arrow line above the top of // the boxes

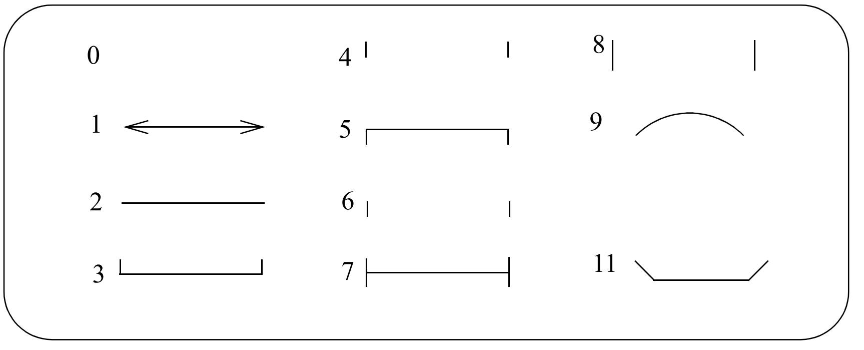
Parameters for the left hand label of the line of the horizontal geometry arrows:

h_g_n_label_x	*mm*	// distance from the left hand side of the // labels area
h_g_n_label_offset	*mm*	// distance to raise the label_text above // arrow line
h_g_n_label_text_size	*mm*	
h_g_n_label_text_colour	*colour*	
h_g_n_label_text	*text*	
h_g_n_label_textstyle	*textstyle*	*// textstyle used*

Parameters for arrow type

h_g_n_draw_mode	0	// no arrow
	1	// arrow
	2	// line
	3	// line with uprights at ends
	4	// uprights, no line

	5	// line with downrights
	6	// downrights, no line
	7	// line with up and downrights at ends
	8	// up and downrights, no line
	9	// draw curve
	11	// radius*length curve



h_g_n_left_gap	*mm*	// *size of gap for left side of arrow*
h_g_n_right_gap	*mm*	// *size of gap for right side of arrow*
h_g_n_colour	*colour*	// colour of the arrow
h_g_n_height	*mm*	// height of the arrow
h_g_n_gap	0	// no gap
	1	// leave gap in arrow for text

Parameters for text on the arrows

h_g_n_text_colour	*colour*	// colour of the text
h_g_n_text_size	*mm*	// size of the text
h_g_n_text_offset	*mm*	// distance to raise the text above the // arrow line
h_g_n_pre_text	*text*	// text before the arrow text
h_g_n_post_text	*text*	// text after the arrow text
h_g_n_textstyle	*textstyle*	// textstyle used
h_g_n_no_decimals	*integer*	// number of decimal places in arrow // text. // If > 0, all trailing zeros after the // decimal place are removed. // If < 0, the absolute value is taken as // the number of decimal places and **no** // trailing zeros are removed after the // decimal point.
h_g_n_rotate	*mm*	// *if ticked, the text on the arrows will be rotated to fit*.

If the set of parameters is to apply to the horizontal geometry of an alignment string *other* than the primary string, then simply add the following parameter to define the other alignment string

h_g_n_offset_string "model->string_name"

or

h_g_n_offset_string "string_name"

and the model is the *Offsets mode*l from the **Section Long Plot** panel.

If the *h_g_n_offset_string* parameter does not exist, then the set of horizontal geometry parameters is applied to the primary string.

Example of Extensive Horizontal Geometry

```
// Parameter to stop the drawing of the quick horizontal geometry
                    horizontal_geometry_y          0
// Set 1 - Extensive Horizontal Geometry Labelling - label the horizontal curve radius
                    h_g_1_type                 1                  // label horizontal curve
                                                                   // if this param is missing then set is ignored
                    h_g_1_value_mode           2                  // label curve radius
                    h_g_1_label_y              45
                    h_g_1_label_x              0
                    h_g_1_label_offset         2
                    h_g_1_label_text_size      4
                    h_g_1_label_text_colour    YELLOW
                    h_g_1_label_text           "Horiz Curve Data"
// Parameters for arrow type
                    h_g_1_draw_mode            1
                    h_g_1_colour               "white"
                    h_g_1_height               1.5
                    h_g_1_gap                  0
// Parameters for text on the arrows
                    h_g_1_text_colour          YELLOW
                    h_g_1_text_size            3.5
                    h_g_1_text_offset          2.5
                    h_g_1_pre_text             "R"
                    h_g_1_post_text            "m"
                    h_g_1_no_decimals          2
```

Example of Extensive Horizontal Geometry for use with a Non-primary Alignment String

```
// Set 2 labels the horizontal curves radii of the kerb string projected onto primary stings
                    h_g_2_offset_string        "left offsets->kerb"          // non-primary string to do VG of

                    h_g_2_type                 1                  // label horizontal curve
                                                                   // if this param is missing then set is ignored
                    h_g_2_value_mode           2                  // label curve radius
                    h_g_2_label_y              55
                    h_g_2_label_x              0
                    h_g_2_label_offset         2
                    h_g_2_label_text_size      4
                    h_g_2_label_text_colour    GREEN
                    h_g_2_label_text           "Left Kerb - Horiz Curve Data"
// Parameters for arrow type
                    h_g_2_draw_mode            1
                    h_g_2_colour               "white"
                    h_g_2_height               1.5
                    h_g_2_gap                  0
// Parameters for text on the arrows
                    h_g_2_text_colour          GREEN
                    h_g_2_text_size            3.5
                    h_g_2_text_offset          2.5
```

```
h_g_2_pre_text        "R"
h_g_2_post_text       "m"
h_g_2_no_decimals     2
```

Please continue to the next section 44.3.12 Quick Vertical Geometry Labelling.

44.3.12 Quick Vertical Geometry Labelling

The standard vertical geometry arrows can be drawn at a given distance above the top of the boxes area.

vertical_geometry_grade_y	*mm*	// dist above boxes
		// 0 = don't draw
vertical_geometry_grade_mode	0	// %
	1	// 1 in
vertical_geometry_label_grade_text	*text*	
vertical_geometry_label_grade_textstyle	*textstyle*	*// textstyle to use*
vertical_geometry_label_grade_text_colour	*colour*	
vertical_geometry_label_grade_text_size	*mm*	
vertical_geometry_label_grade_decimals	*number*	
vertical_geometry_length_y	*mm*	// dist above boxes
		// 0 = don't draw
vertical_geometry_length_mode	0	// length
	1	// radius
	2	// k value
	3	// mixed - length for parabolic
		// radius for circular
vertical_geometry_label_length_text	*text*	
vertical_geometry_label_length_textstyle	*textstyle*	*// textstyle to use*
vertical_geometry_label_length_text_colour	*colour*	
vertical_geometry_label_length_text_size	*mm*	
vertical_geometry_label_length_decimals	*number*	
vertical_geometry_arrow_mode	0	// ticks
	1	// arrows
vertical_geometry_arrow_grade_textstyle	*textstyle*	*// textstyle to use*
vertical_geometry_arrow_grade_text_colour	*colour*	
vertical_geometry_arrowl_grade_text_size	*mm*	
vertical_geometry_arrow_length_textstyle	*textstyle*	*// textstyle to use*
vertical_geometry_arrow_length_text_colour	*colour*	
vertical_geometry_arrow_length_text_size	*mm*	
vertical_geometry_arrow_colour	*colour*	
vertical_geometry_arrow_height	*mm*	

Please continue to the next section 44.3.13 Extensive Vertical Geometry Labelling.

44.3.13 Extensive Vertical Geometry Labelling

For complicated vertical geometry labelling of the *primary alignment* string, there are sets of vertical geometry labelling parameters which give tight control over the position and types of labels.

It is also possible to label the vertical geometry of ***alignment*** strings other than the primary string. To plot such a string on the same plot, the chainage position of the vertical geometry for the non-primary alignment strings is *projected* onto the primary string to give a primary string chainage for plotting. The values of the vertical geometry (such as grade and curve length) that are plotted are taken from the other string. Independently graded offset strings (such as a left and right kerbs) are the type of additional alignment strings that may need to be plotted on the same long section plot as the reference string (primary string).

For plotting vertical geometry, the user can give up to twenty sets of these labels and they can be used to label grades or vertical curve information for the primary string and/or additional alignment strings.

Each label set consists of three parts:

(a) a text label on the left hand side of the plot
(b) an arrow
(c) text on the arrows.

For the following parameters, n takes the value 1 to 20 and specifies the nth parameter set.

v_g_n_type	0	// labelling grades
	1	// labelling vg curve information

If *v_g_n_type* is missing, then the entire nth set is ignored.

The interpretation of the value of the

parameter *v_g_n_value_mode* depends on whether the set is being used for grade labelling or curve labelling:

v_g_n_value_mode		
for grade labelling	0	// nothing
	1	// % grade
	2	// 1 in grade
	3	// m/m grade
	4	// chainage length between vertical
		// curve points
	5	// per chord - Queensland Rail
for curve labelling	0	// nothing
	1	// length - chainage for parabolic vc's
		// arc length for circular vc's
	2	// radius
	3	// K value
	5	// curve constant - Queensland Rail

Vertical curve points to draw the arrows between (for grade labelling only)

v_g_n_between_mode	0	// between chainages at the vip's
	1	// between chainages at the vtp's

Position of the Arrow Line

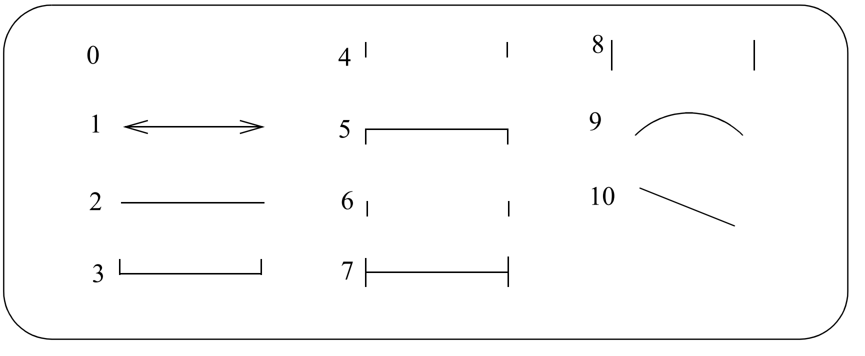
v_g_n_label_y	*mm*	// distance of arrow line above the top of
		// the boxes this can be negative

Parameters for the left hand label of the whole line of the vg arrows:

v_g_n_label_x	*mm*	// distance from the left hand side of the // labels area
v_g_n_label_offset	*mm*	// distance to raise the label_text above // arrow line
v_g_n_label_text	*text*	
v_g_n_label_textstyle	*textstyle*	
v_g_n_label_text_size	*mm*	
v_g_n_label_text_colour	*colour*	

Parameters for arrow type

v_g_n_draw_mode	0	// no arrow
	1	// arrow
	2	// line
	3	// line with uprights at ends
	4	// uprights, no line
	5	// line with downrights
	6	// downrights, no line
	7	// line with up and downrights at ends
	8	// up and down rights, no line
	9	// draw curve
	10	// draw grade



v_g_n_left_gap	*mm*	*// size of gap for left side of arrow*
v_g_n_right_gap	*mm*	*// size of gap for right side of arrow*
v_g_n_colour	*colour*	// colour of the arrow
v_g_n_height	*mm*	// height of the arrow
v_g_n_gap	0	// no gap
	1	// leave gap in arrow for text

Parameters for text on the arrows

v_g_n_text_colour	*colour*	// colour of the text
v_g_n_text_size	*mm*	// size of the text
v_g_n_text_offset	*mm*	// distance to raise the text above the // arrow line
v_g_n_pre_text	*text*	// text before the arrow text
v_g_n_post_text	*text*	// text after the arrow text
v_g_n_textstyle	*textstyle*	// textstyle used

v_g_n_no_decimals	*integer*	// number of decimal places in arrow
		// text.
		// If > 0, all trailing zeros after the
		// decimal place are removed.
		// If < 0, the absolute value is taken as
		// the number of decimal places and **no**

// trailing zeros are removed after the
// decimal point.

v_g_n_rotate *mm* // *if ticked, the text on the arrows will be rotated to fit*.

If the set of parameters is to apply to the vertical geometry of an alignment string *other* than the primary string, then simply add the following parameter to define the other alignment string

v_g_n_offset_string "model->string_name"

or

v_g_n_offset_string "string_name"

and the model is the *Offsets mode*l from the **Section Long Plot** panel.

If the *v_g_n_offset_string* parameter does not exist, then the set of vertical geometry parameters is applied to the primary string.

Example of Extensive Vertical Geometry

// Parameter to not to draw the quick vertical geometry

vertical_geometry_grade_y 0

// Set 1 labels the vertical curves with length

v_g_1_type 1 // curve
v_g_1_value_mode 1 // chainage length

v_g_1_label_y 45
v_g_1_height 4
v_g_1_label_offset -2
v_g_1_label_text "VG Curves"
v_g_1_text_offset 2
v_g_1_label_text_size 6
v_g_1_pre_text ""
v_g_1_post_text " L"
v_g_1_gap 0
v_g_1_draw_mode 8
v_g_1_text_size 4

// Set 2 labels the vertical curves with K value

v_g_2_type 1 // curve
v_g_2_value_mode 3 // K value

v_g_2_label_y 45
v_g_2_height 1.5
v_g_2_label_text ""
v_g_2_text_offset -5
v_g_2_pre_text ""
v_g_2_post_text " K"
v_g_2_gap 0
v_g_2_draw_mode 1
v_g_2_text_size 4

Example of Extensive Vertical Geometry for use with a Non-primary Alignment String

// Set 3 labels the vertical curves with length with the kerb string projected onto primary stings

```
v_g_3_offset_string        "left offsets->kerb"        // non-primary string to do VG of
v_g_3_type              1    // curve
v_g_3_value_mode        1    // chainage length

v_g_3_label_y           55
v_g_3_height             4
v_g_3_label_offset      -2
v_g_3_label_text        “Left kerb - VG Curves”
v_g_3_text_offset        2
v_g_3_label_text_size    6
v_g_3_pre_text          ““
v_g_3_post_text         “ L”
v_g_3_gap                0
v_g_3_draw_mode          8
v_g_3_text_size          4
```

Please continue to the next section 44.3.14 Labelling Chainages and Heights in the Graph Area.

44.3.14 Labelling Chainages and Heights in the Graph Area

The chainage and/or height values for certain points (given by *label_n_type*) can be labelled.

Up to twenty sets of chainage/height labels can be done.

label_n_type	0	// chainage of vip, height of vip
	1	// chainage of vip, height of primary
	2	// crest
	3	// sag
	4	// vtp
	5	// hcp
	6	// change of grade
	7	// mid-ordinate of the vertical curve

If *label_n_type* is missing, then the set is ignored.

label_n_y_mode	0	// height in mm above boxes
	1	// above height value (default)
	2	// above primary height
label_n_y	*mm*	// distance above point
label_n_angle	*degrees*	// rotation about point
label_n_x	*mm*	// distance along from point
label_n_offset	*mm*	// text raise height
label_n_justification	0	// left end
	1	// middle
	2	// end
label_n_size	*mm*	
label_n_colour	*size*	
label_n_textstyle	*textstyle*	
label_n_value_mode	0	// no values labelled
	1	// val 1 = chainage
	2	// val 1 = height
	3	// val 1 = chainage, val 2 = height
	4	// val 1 = height, val 2 = chainage
label_n_pre_text	*text*	
label_n_mid_text	*text*	
label_n_post_text	*text*	
label_n_textstyle	*textstyle*	*// textstyle to use*
label_n_no_decimals_1	*integer*	// number of decimal places in val 1
label_n_no_decimals_2	*integer*	// number of decimal places in val 2

If the **number of decimal places is greater than zero (> 0)**, then any trailing zeros after the decimal point are removed.

If the **number of decimal places is less than zero (< 0)**, the absolute value is taken as the number of decimal places and **no** trailing zeros after the decimal point are removed.

Example of Labelling Chainage and Heights

```
// label the crests with chainage on one line
// and height (elevation) on the next

label_1_type             2   // crest
label_1_y_mode           2   // above string
label_1_y               10
label_1_angle            0
label_1_justification    1   // centre
label_1_size             4
label_1_colour         red
label_1_value_mode       1   // chainage
```

```
label_1_pre_text            "CH "
label_1_no_decimals_1       1

label_2_type                2   // crest
label_2_y_mode              2   // above string
label_2_y                   2
label_2_angle               0
label_2_justification       1   // centre
label_2_size                4
label_2_colour              red
label_2_value_mode          2   // height
label_2_pre_text            "EL "
label_2_no_decimals_1       1
```

Please continue to the next section 44.3.15 Labelling With Symbols.

44.3.15 Labelling With Symbols

Symbols can be placed at certain points given by *symbol_n_type*.

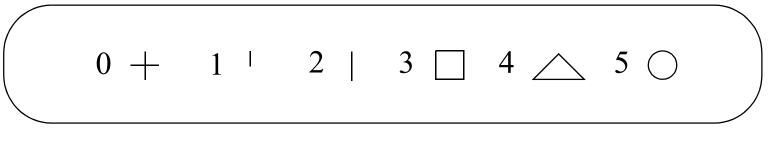
The symbol is drawn in a square box centred on (0,0) with sides of length two millimetres. That is, the box co-ordinates are (-1,-1), (1,1), (1,-1), (-1,-1).

Up to twenty sets of symbol labelling can be done.

symbol_n_type	0	// chainage of vip, height of vip
	1	// chainage of vip, height of primary
	2	// crest
	3	// sag
	4	// vtp
	5	// hcp
	6	// change of grade

If *symbol_n_type* is missing, then the set is ignored.

symbol_n_y_mode	0	// height in mm above boxes
	1	// above height value (default)
	2	// above primary height
symbol_n_y	*mm*	// distance above point given by mode
symbol_n_angle	*degrees*	// rotation about point
symbol_n_x	*mm*	// distance along from point
symbol_n_size	*mm*	//
symbol_n_colour	*colour*	// colour of symbol
symbol_n_draw_mode	0	// cross
	1	// upright from centre of box
	2	// up and downright from centre of box
	3	// square
	4	// triangle, base at bottom
	5	// circle



Example of Labelling with Symbols

// draw a triangle symbol at the crest

symbol_1_type	2	// crest
symbol_1_y_mode	2	// above string
symbol_1_y	0	
symbol_1_angle	0	
symbol_1_size	2	
symbol_1_colour	red	
symbol_1_draw_mode	4	// triangle

Please continue to the next section 44.3.16 Hatching Cut and Fill Areas.

44.3.16 Hatching Cut and Fill Areas

This option is used to hatch cut and/or fill areas between sets of tins.

For each set, the name of the two tins, the hatch linestyle, colour and separation and whether cut and/or fill regions are required are all user definable.

Up to twenty (20) separate sets of tins be hatched.

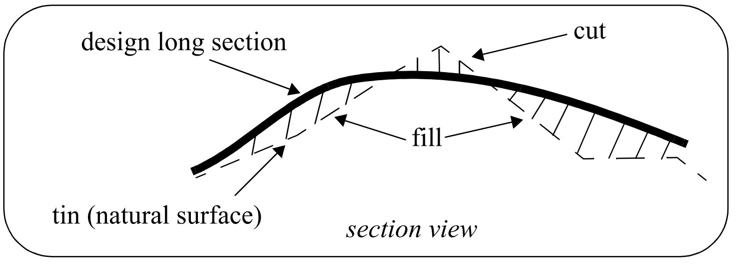
The parameters for labelling cuts and/or fill regions between tins are given by:

hatch_original_tin_n	*tin_name*	// tin_name for original surface
hatch_new_tin_n	*tin_name*	// tin_name for final surface
hatch_cut_separation_n	*mm*	// distance between cut hatch lines
	0	// don't do cut hatching
hatch_cut_angle_n	*degrees*	// angle in degrees of cut hatching
hatch_cut_colour_n	*colour*	// colour of the cut hatching
hatch_cut_linestyle_n	*linestyle*	// linestyle for cut hatching
hatch_cut_draw_sides_n	1/0	// 1 = draw sides of cut regions
hatch_cut_draw_original_n	1/0	// 1 = draw original tin in cut regions
hatch_cut_draw_new_n	1/0	// 1 = draw new tin in cut regions
hatch_fill_separation_n	*mm*	// distance between fill hatch line
	0	// don't do fill hatching
hatch_fill_angle_n	*degrees*	// angle in degrees of fill hatching
hatch_fill_colour_n	*colour*	// colour of the fill hatching
hatch_fill_linestyle_n	*linestyle*	// linestyle for fill hatching
hatch_fill_draw_sides_n	1/0	// 1 = draw sides of fill regions
hatch_fill_draw_original_n	1/0	// 1 = draw original tin in fill regions
hatch_fill_draw_new_n	1/0	// 1 = draw new tin in fill regions

Notes

(a) cut is when the new tin is below the original tin.
fill is when the new tin is above the original tin.

(b) cut hatching is turned off by setting *hatch_cut_separation_n* to 0.0.
fill hatching is turned off by setting *hatch_fill_separation_n* to 0.0.



Please continue to the next section 44.3.17 Labelling Cuts of Design Through Strings in a Model.

44.3.17 Labelling Cuts of Design Through Strings in a Model

The cuts that the primary string (design line) makes though any strings in user given models can be automatically labelled on the long section plots.

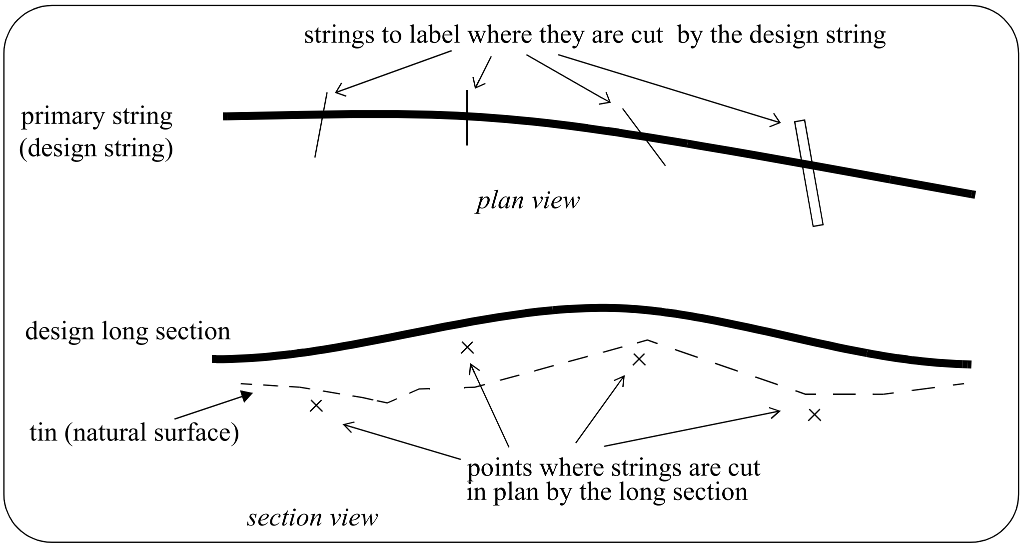
The **height**, **chainage** and **name** of the cut string can be labelled as well as a **symbol**. The height of tins at the same offset value can also be labelled.

The chainage position for the labelling is the chainage of the cut string.

The height position for the labelling can be specified as the

(a) top of the boxes on the long section
(b) height value of the cut string
(c) height of the primary string
(d) height of a tin.

The actual position of the label is defined relative to the above point.



Note:

Only case (b) involves the actual height of the cut string. For all other cases, only the chainage of the cut string is used. Hence for all cases except (b), the string does need to have a sensible height to be used for cuts through strings.

For example, a boundary string may have null heights but only the chainage is required and the height of the tin at that chainage can be used as the height (case (d)).

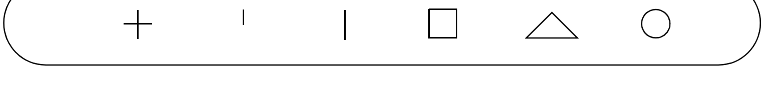
Text justification refers to the actual position and is given by

"top-left"	"top-centre"	"top-right"
"middle-left"	"middle-centre"	"middle-right"
"bottom-left"	"bottom-centre"	"bottom-right"

A choice of six special symbols and/or any 12d symbols can be drawn at the cut point.

The special 12d Model symbols of size one millimetre are drawn in a square box centred on (0,0) with sides of length two millimetres. That is, the box co-ordinates are (-1,-1), (1,1), (1,-1), (-1,-1).

The six special shapes are



Up to twenty five (25) separate models of strings can be cut and labelled.

Parameters for Labelling Where the Design Cuts Strings in a Model

The method for specifying which strings are to be checked for cuts is by first specifying the **model** which contains the strings, and then a **name mask** which is used to restrict the strings in the model to only those whose name matches the name mask.

Up to twenty five different sets of models and name masks can be used so that different cut sets can be labelled in different ways.

The parameters for selecting and labelling the nth set (where n can be from 1 to 25) of cuts of the design string with the strings in the model are given by:

cuts_n_model *model_name* // model of strings to be cut

The selection of the strings from the model *model_name* whose cut points are to be labelled is all the strings whose name satisfies the name mask *cuts_n_mask*:

cuts_n_mask *name_mask* // strings to check for cuts
// and if a cut occurs,
// parameters show how to
// label the cut

where *name_mask* is a text string containing the name masks, each separated by one or more spaces, to test the string name against. Each mast can include wild cards and wild characters.

For example

cuts_1_mask "ke*"
or cuts_1_mask "?bank*"

or, if both masks are required,

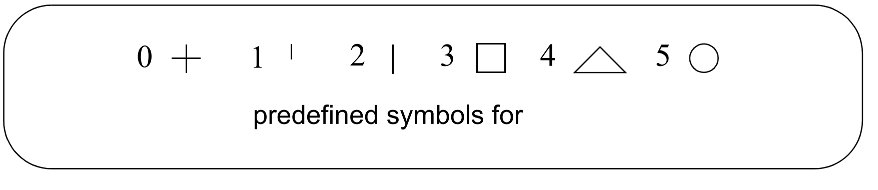
cuts_1_mask "ke* ?bank*"

If *cuts_n_mask* is missing, then all strings in the model are used. This is equivalent to *name_mask* being "*".

All strings in the model *cuts_n_model* whose name satisfy the name mask *cuts_n_mask* are then checked for cuts with the design string, and if a cut occurs, the cut point will be labelled according to the rest of the parameters in the nth set.

The parameters for drawing a **symbol** at the cut points are

cuts_symbol_n_mode 0 // cross
1 // up from centre of box
2 // up and down from centre of box
3 // square
4 // triangle, base at bottom
5 // circle
6 // use a 12d symbol



If *cuts_symbol_n_mode* is 6, then the 12d symbol is given by

cuts_symbol_n_style *plotsymbol* // plot symbol to draw at cut

Important Note

The plot symbol of name *plotsymbol* is defined in the file given by:

(a) the parameter *plot_symbols* in the ppf file

plot_symbols *filename*

or if plot_symbols is not defined, then

(b) in the file pointed to by the environment variable PLOT_SYMBOLS_4D

PLOT_SYMBOLS_4D *filename* *// default plotsym.4d*

or if PLOT_SYMBOLS_4D is not defined, then

(c) in the file *plotsym.4d*

which is searched for in the standard set up file sequence

If none of the above files are defined, or if the symbol does not exist in the above files, then it will be searched for in the standard 12d symbols file which is:

(d) either pointed to by the environment variable SYMBOLS_4D

SYMBOLS_4D filename // default symbols.4d

or if the environment variable SYMBOLS_4D does not exist, in the file, *symbols.4d*

The position of the symbol is given by:

cuts_symbol_n_position	1	// above point height value
	3	// above top of boxes
	100	// to primary string
	101-500	// to tin1 or tin2 etc.

The symbol can be adjusted by the parameters:

cuts_symbol_n_x	*mm*	// offset adjustment to position
cuts_symbol_n_y	*mm*	// height adjustment to position
cuts_symbol_n_angle	*degrees*	// rotation about point
cuts_symbol_n_colour	*colour*	// colour of symbol

and for all values of *cuts_symbol_n_mode* **other** than 6:

cuts_symbol_n_size	*mm*	// size of symbol, 0 don't draw

The value of the **chainage** of the cut string can be labelled using the parameters

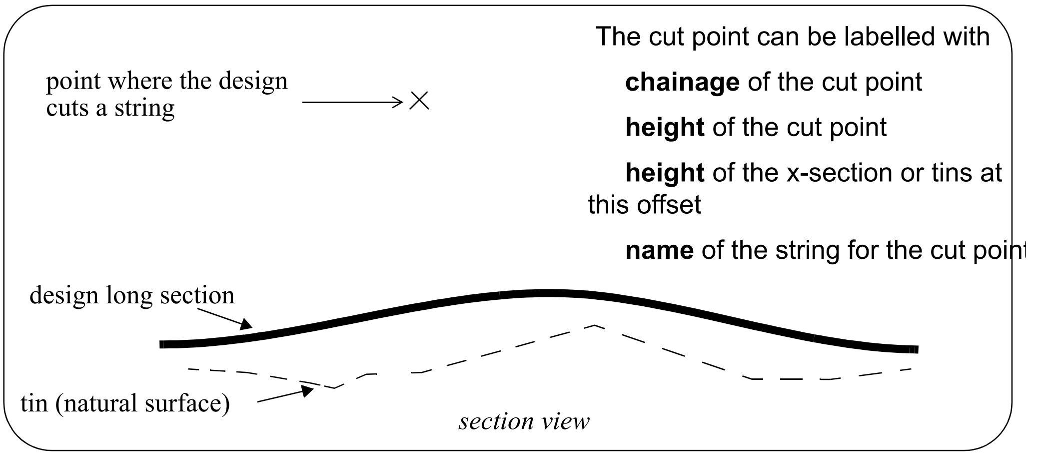
cuts_chainage_n_position	1	// above cut strings height value
	3	// above top of boxes
	100	// to primary string
	101-500	// to tin1 or tin2 etc.
cuts_chainage_n_x	*mm*	// chainage adjustment to position
cuts_chainage_n_y	*mm*	// height adjustment to position
cuts_chainage_n_angle	*degrees*	// rotation about point
cuts_chainage_n_size	*mm*	// size of text, 0 don't label
cuts_chainage_n_colour	*colour*	// colour of text
cuts_chainage_n_textstyle	*text*	// textstyle of text chainage
cuts_chainage_n_pre_text	*text*	// text before the chainage value
cuts_chainage_n_post_text	*text*	// text after the chainage value
cuts_chainage_n_justification	*justification*	// justification of the text
cuts_chainage_n_no_decimals	*integer*	// number of decimals in chainage

The value of a **height** at the chainage of the point can be calculated and labelled using the parameters

cuts_height_n_mode	*1*	// use height of cut point itself
	3	// use real world height of the position
		// of the label above the boxes
	100	// height of primary string
	101-500	// use height of to tin1 or tin2 etc.
cuts_height_n_position	*1*	// at points position
	3	// above top of boxes
	100	// to primary string
	101-500	// to tin1 or tin2 etc.
cuts_height_n_x	*mm*	// chainage adjustment to position
cuts_height_n_y	*mm*	// height adjustment to position
cuts_height_n_angle	*degrees*	// rotation about point
cuts_height_n_size	*mm*	// size of text, 0 don't label
cuts_height_n_colour	*colour*	// colour of text
cuts_height_n_textstyle	*text*	// textstyle of text height
cuts_height_n_pre_text	*text*	// text before the height value
cuts_height_n_post_text	*text*	// text after the height value
cuts_height_n_justification	*justification*	// justification of the text
cuts_height_n_no_decimals	*integer*	// number of decimals in height

A **label** which can include the **name** of the cut string is drawn by using the parameters

cuts_label_n_position	1	// above cut strings height value
	3	// above top of boxes
	100	// to primary string
	101-500	// to tin1 or tin2 etc.
cuts_label_n_mode	0	// don't include cut string name
	1	// include cut string name in label
cuts_label_n_x	*mm*	// chainage adjustment to position
cuts_label_n_y	*mm*	// height adjustment to position
cuts_label_n_angle	*degrees*	// rotation about point
cuts_label_n_size	*mm*	// size of text, 0 don't label
cuts_label_n_colour	*colour*	// colour of text
cuts_label_n_textstyle	*text*	// textstyle of text label
cuts_label_n_pre_text	*text*	// text before the string name
cuts_label_n_post_text	*text*	// text after the string name
cuts_label_n_justification	*justification*	// justification of the text



Please continue to the next section 44.3.18 Labelling the Primary String Name on the Plot.

44.3.18 Labelling the Primary String Name on the Plot

The plot can be labelled with a name under the boxes area.

The name is made up of concatenation the text strings:

plot_name_pre_text	primary-string-name	plot_name_post_text

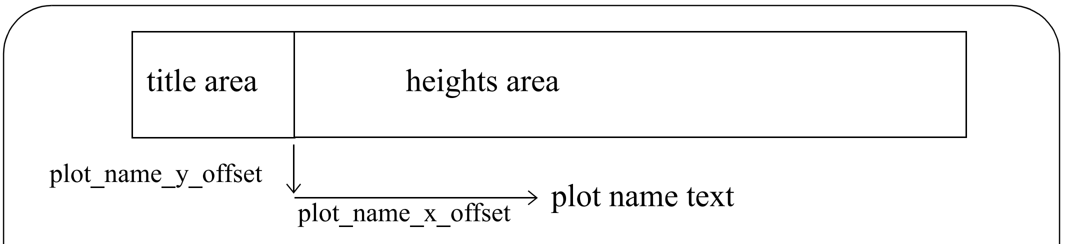
The plot name is positioned under the boxes.

plot_name_string_name	0	// don't use the primary string name
	1	// use primary string name
	2	// use model->string name
plot_name_pre_text	*text*	
plot_name_post_text	*text*	
plot_name_textstyle	*textstyle*	
plot_name_size	*mm*	
plot_name_colour	*colour*	
plot_name_x_offset	*mm*	
plot_name_y_offset	*mm*	

The *plot_name_x_offset* is measured from the beginning of the height boxes.

The default for *plot_name_x_offset* is centred on heights area.

The *plot_name_y_offset* is measured from the bottom of the box area with positive being **down**.



labelling primary string name on long section plots

Example of Labelling Primary String Name

plot_name_pre_text	"Long Section Plot for String"
plot_name_post_text	""
plot_name_size	15
plot_name_colour	red
plot_name_string_name	1
plot_name_y_offset	30

Please continue to the next section 44.3.19 Labelling the Scale on the Plot.

44.3.19 Labelling the Scale on the Plot

The plot can be labelled with the horizontal and vertical scale under the boxes area.

The scale label is made up of concatenation the text strings:

scale_horizontal_pre_text	horizontal scale value	scale_horizontal_post_text
and		
scale_vertical_pre_text	vertical scale value	scale_vertical_post_text

The horizontal scale value is the value given by the scale parameter.

The vertical scale value is calculated from the horizontal scale and the vertical exaggeration for the section view.

The scales are positioned under the boxes.

scale_horizontal_pre_text	*text*
scale_horizontal_post_text	*text*
scale_horizontal_textstyle	*textstyle*
scale_horizontal_size	*mm*
scale_horizontal_colour	*colour*
scale_horizontal_x_offset	*mm*
scale_horizontal_y_offset	*mm*
scale_horizontal_decimals	*integer*
scale_vertical_pre_text	*text*
scale_vertical_post_text	*text*
scale_vertical_textstyle	*textstyle*
scale_vertical_size	*mm*
scale_vertical_colour	*colour*
scale_vertical_x_offset	*mm*
scale_vertical_y_offset	*mm*
scale_vertical_decimals	*integer*

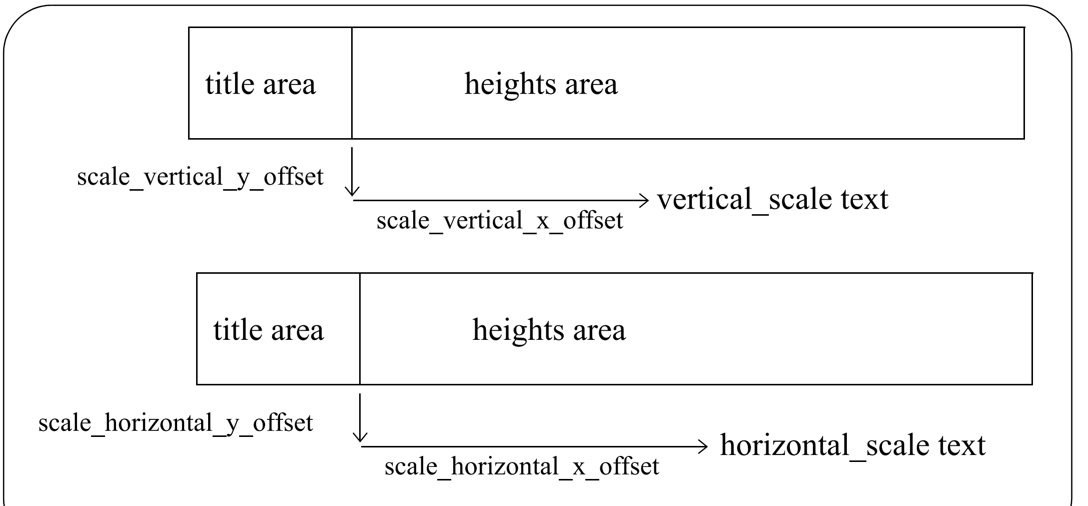
The *scale_vertical_x_offset* and *scale_horizontal_x_offset* are measured from the beginning of the heights area.

The default for *scale_vertical_x_offset* and *scale_horizontal_x_offset* are centred on the heights area.

The *scale_vertical_y_offset* and *scale_horizontal_y_offset* are measured from the bottom of the box area with positive being **down**.

If the **number of decimal places is greater than zero (> 0)**, then any trailing zeros after the decimal point are removed.

If the **number of decimal places is less than zero (< 0)**, the absolute value is taken as the number of decimal places and **no** trailing zeros after the decimal point are removed.



labelling of scales on the long section plots

Example of Labelling Horizontal and Vertical Scales

scale_horizontal_pre_text	"Horizontal Scale 1:"
scale_horizontal_post_text	""
scale_horizontal_size	15
scale_horizontal_colour	yellow
scale_horizontal_y_offset	60

scale_vertical_pre_text	"Vertical Scale 1:"
scale_vertical_post_text	""
scale_vertical_size	15
scale_vertical_colour	green
scale_vertical_y_offset	90

Please continue to the next section 44.3.20 Title Block Information.

44.3.20 Title Block Information

The plot can have a standard ***12d Model*** title block or a user defined title block.

The standard title block consists of a simple border around the plot and two lines of text in a box underneath the plot. For a user defined title block, all the line work and text is defined by the user.

Standard Title Block

For the standard ***12d Model*** title block, there are extra parameters for two lines of text and text size and colour. The standard title block is turned on or off by the parameter *plot_border*.

plot_border	*yes/no*	*// yes plots a standard title block* *// default yes*
title_1	*text*	
title_2	*text*	
title_text_size	*value*	
title_colour	*colour*	

User Title Block

For the user defined title block, the title block drawing commands are kept in a file whose name is supplied by the user. The title block drawing commands are almost identical to the linestyle drawing commands and is given at the beginning of this chapter.

Hence for a user defined title block, there are just two parameters - one to say a title block file is being used and the other to give the name of the title block file. The *plot_border* parameter should also be set to *no* so that the standard title block is not also drawn.

use_title_file	*yes/no*	*// yes draws the title block given in title_file* // default no
title_file	*filename*	
plot_border	*no*	*// turn off standard title block*

Some special plot parameters are used to pass information down to variables in a user defined title block. For example, inside the title block file it is possible to have runtime user defined text variables. The actual text values for these text variables are passed down to the title block file from the plot parameter file via the parameters user_text_n (n = 1,2,... 1000)

user_text_n	*text*

The special plot parameters are:

time_format	*text*	// format for $time
user_text_n	*text*	// where n = 1,2,... 1000 // passed down to $user_text_n
title_1	*text*	// passed down to $title_1
title_2	*text*	// passed down to $title_2
start_page_number	*integer*	// used as the starting value for // $page_number. If missing, // $page_number starts at 1.

start_drawing_number	*integer*	// added to $drawing_number in title
		// block file. If missing,
		// $drawing_number starts at 1.

drawing_number_prefix	*text*	// passed down to // $drawing_number_prefix
drawing_number_postfix	text	// passed down to // $drawing_number_postfix

Please continue to the next section 44.3.21 Parameters that Modify Fields In the Long Plot Panel.

44.3.21 Parameters that Modify Fields In the Long Plot Panel

A number of parameters match those in the **section long plot** panel.

When the plot parameter file is first read, any parameters in the panel will be replaced by the values of any corresponding parameters from the parameter file.

However, if the parameter is subsequently modified in the panel, the panel value will be the value used for any plots.

The plot parameters that also occur in the **section long plot** panel are:

scale	*value*
start_chainage	*value*
end_chainage	*value*
chainage_interval	*value*
chord_arc	0/1
plotter_type	*text*
plot_file	*text*
view_name	*text*
offset_model	*text*
plot_border	*yes/no*
label_depths	*yes/no*
primary_string	*yes/no*
datum_value	*value*
sheet_size	*text* or "width *height"*
box_text_size	*mm*
box_colour	*colour*
use_title_file	*yes/no*
title_file	*filename*
plot_border	*yes/no*
title_1	*text*
title_2	*text*
title_text_size	*value*
title_colour	*colour*
pagination	*yes/no*
pagination_length	*value*
pagination_overlap	*value*
global_textstyle	*textstyle*

Please continue to the next section 44.3.22 Generating Long Section Plots Without a View.

44.3.22 Generating Long Section Plots Without a View

The long section plot parameters are comprehensive enough that it is possible to completely generate a long section plot without referencing a section view, or even using the **section long plot** panel.

Such a ppf can be run using the **plots=>plot a ppf** option or from the 4D Solutions programming language, 4DML.

When a long section plot is being generated entirely from a file, an extra parameter is needed to specify whether the datum value is calculated or the *datum_value* parameter is used.

manual_datum	1	// use the datum_value parameter for // the datum
	0	// ignore the datum_value and let // 12d Model calculate the datum.

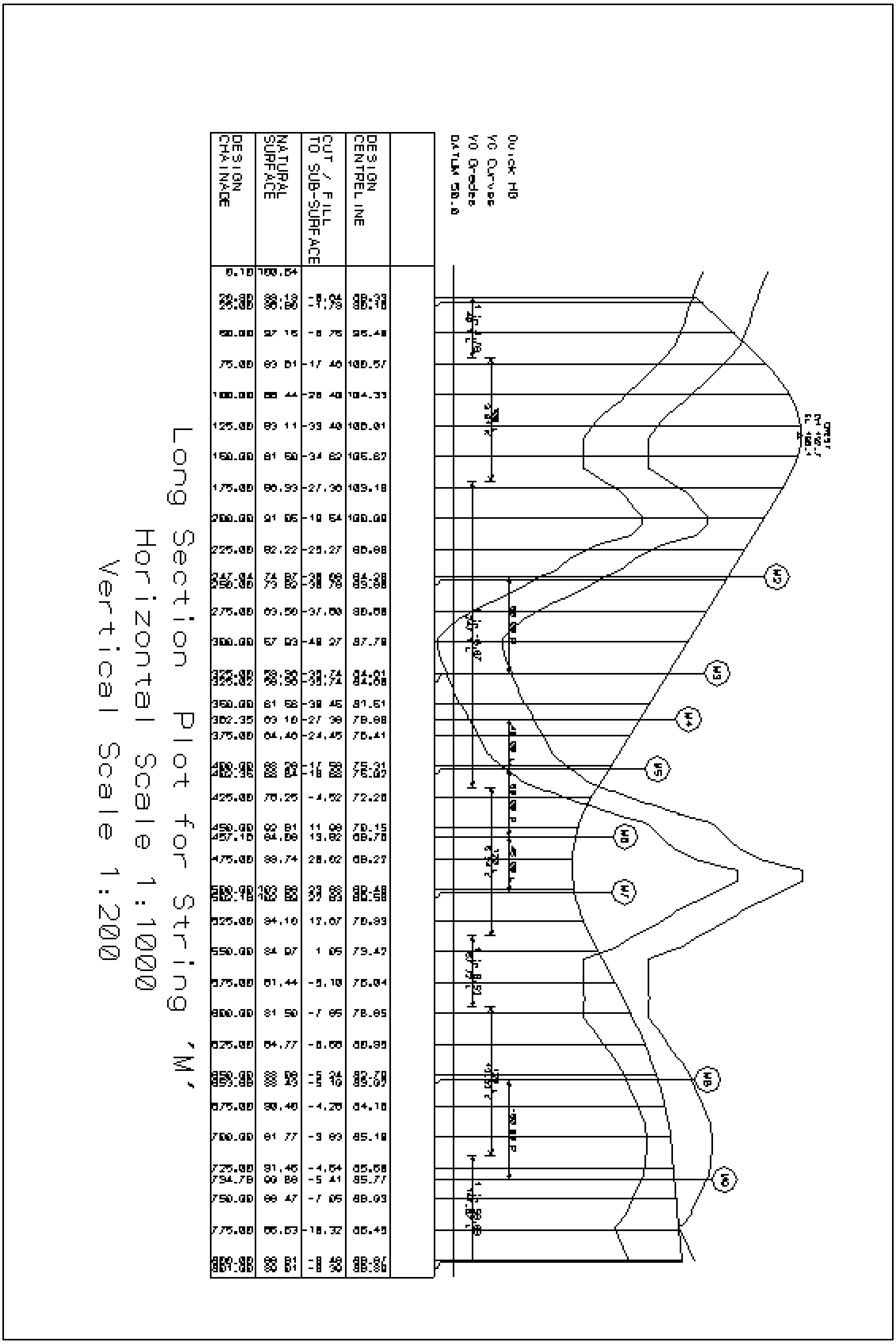
When generating a long section plot using the **section long plot** panel, a plot parameter file containing all the parameters needed to regenerate the plot using **plots=>plot a ppf** can be written out by simply giving a name for the ppf file in the **plot parameters write** field of the **section long plot** panel.

Notes

1. A warning is given if the keyword in a plot parameter file does not exist.
2. A warning is also given if the key word pair is defined more than once in a ppf.

Please continue to the next section 44.3.23 Example of a Long Section Plot Parameter File.

44.3.23 Example of a Long Section Plot Parameter File



```
        // ppf file to generate longsection example
section_long_plot   "diag2"  {
//  plot margins
 left_margin            0
 right_margin           0
 top_margin             0
 bottom_margin         30
  number_of_decimals  2   // no dec places for heights
 title_box_text_size     7
// annotation for title text and heights etc.
// chainages
 chainage_title              "DESIGN"
 chainage_title_2             "CHAINAGE"
 chainage_colour              "red"
// primary string
 primary_title                "DESIGN"
 primary_title_2              "CENTRELINE"
 primary_colour               "grey"
 primary_title_colour        "yellow"
// primary label modes are
// 0 - last box before blank boxes
// 1 - after chainage annotations
 primary_label_mode           0
// uprights
 uprights_colour               "yellow"
//uprights_draw_mode          0      // none
 uprights_draw_mode         100    // to primary
 uprights_y                 200
// gap factor for staggering
 stagger_gap_factor          1.3
 stagger_gap_top             5.0
 stagger_gap_bottom          2.0
 stagger_mode               1
// datum parameters
 datum_text_size              6
 datum_colour                brown
 datum_name                 "DATUM"
 datum_above_gap            15
 datum_below_gap            15
 datum_y                     -2
//  0 for bottom text justification
//  1 for top justification
 box_text_justification         1
// draw lines around annotation at bottom of plot
// 0 - don't draw them
// 1 - do draw
// 2 - draw box around label text as well
// plus many others
 draw_box_mode               2
```

```
// tin titles and depth labels
// label = 0 for no label, 1 for label
 tin_1_label                 1            // label natural surface
 tin_1_title                 "NATURAL"
 tin_1_title_2               "SURFACE"
 tin_1_title_colour          yellow
 tin_1_colour                "green"
 tin_1_depth_label   0       // don't label depth to natural surface
 tin_1_depth_title           "CUT / FILL"

 tin_2_label             0          // don't label sub-surface
 tin_2_depth_label    1             // label depth to sub-surface
 tin_2_title                 "SUB"
 tin_2_title_2               "SURFACE"
 tin_2_title_colour          yellow
 tin_2_colour                "green"
 tin_2_depth_title           "CUT / FILL"
 tin_2_depth_title_2         "TO SUB-SURFACE"
//  number of blank boxes at the top
//  of the plot for user annotations
 number_of_blank_boxes        1
//  chainages to include for heights labelling
//   0 don't use, 1 use
// include the chord-arc points?
 chord_arc                         0
// include the horizontal critical points?
 chainage_label_hcp              1
// include horizontal ip's with no curves?
 chainage_label_hip              0
// include the vertical tangent points?
 chainage_label_vtp              0
// include the vips?
 chainage_label_vip              0
// include the crests?
 chainage_label_crest            0
// include the sags?
 chainage_label_sag              0
// include change of grade
 chainage_label_grade_change           0
// don't include bubble chainages
 chainage_merge_bubbles                0
```

```
// Quick geometry - by default the geometry annotation
//  are those of the view
//  if they are zero no annotations are performed
 horizontal_geometry_y                    60
 horizontal_geometry_arrow_text_size       4
 horizontal_geometry_arrow_height          3
 horizontal_geometry_arrow_colour        red
 horizontal_geometry_label_text         “Quick HG”
 horizontal_geometry_label_text_colour    green
 horizontal_geometry_label_text_size       6

 vertical_geometry_grade_y               0
 vertical_geometry_length_y               0
// Extensive vertical geometry annotations
 v_g_1_type                1    // curve
 v_g_1_value_mode          1    // chainage length

 v_g_1_label_y            45
 v_g_1_height              4
 v_g_1_label_offset       -2
 v_g_1_label_text         “VG Curves”
 v_g_1_text_offset         2
 v_g_1_label_text_size     6
 v_g_1_pre_text           ““
 v_g_1_post_text          “ L”
 v_g_1_gap                 0
 v_g_1_draw_mode           8
 v_g_1_text_size           4

 v_g_2_type                1    // curve
 v_g_2_value_mode          3    // K value

 v_g_2_label_y            45
 v_g_2_height             1.5
 v_g_2_label_text         ““
 v_g_2_text_offset        -5
 v_g_2_pre_text           ““
 v_g_2_post_text         “ K”
 v_g_2_gap                 0
 v_g_2_draw_mode           1
 v_g_2_text_size           4

 v_g_3_type                0    // grade
 v_g_3_value_mode          2    // 1 in grade
 v_g_3_between_mode        1    // between vtp’s

 v_g_3_label_y            30
 v_g_3_height              4
 v_g_3_label_offset       -2
 v_g_3_label_text         “VG Grades”
 v_g_3_label_text_size     6
 v_g_3_text_offset         2
 v_g_3_pre_text           “1 in “
 v_g_3_post_text          ““
 v_g_3_gap                 0
 v_g_3_draw_mode           8
 v_g_3_text_size           4
```

```
v_g_4_type               0    // grade
v_g_4_value_mode         4    // chainage length
v_g_4_between_mode       1    // between vtp's

v_g_4_label_y           30
v_g_4_height            1.5
v_g_4_label_text        ""
v_g_4_text_offset       -5
v_g_4_pre_text          ""
v_g_4_post_text         " L"
v_g_4_gap                0
v_g_4_draw_mode          1
v_g_4_text_size          4
```

```
// label the crests with chainage on one line
// and height (elevation) on the next
```

```
label_1_type             2    // crest
label_1_y_mode           2    // above string
label_1_y               10
label_1_angle            0
label_1_justification    1    // centre
label_1_size             4
label_1_colour          red
label_1_value_mode       1    // chainage
label_1_pre_text        "CH "
label_1_no_decimals_1    1

label_2_type             2    // crest
label_2_y_mode           2    // above string
label_2_y                2
label_2_angle            0
label_2_justification    1    // centre
label_2_size             4
label_2_colour          red
label_2_value_mode       2    // height
label_2_pre_text        "EL "
label_2_no_decimals_1    1

label_3_type             2    // crest
label_3_y_mode           2    // above string
label_3_y               18
label_3_angle            0
label_3_justification    1    // centre
label_3_size             4
label_3_colour          red
label_3_value_mode       0    // no values
label_3_pre_text        CREST
```

```
// Draw bubbles at the horizontal critical points:
//   chainages to include for bubbles numbering
//    0 don't use, 1 use
chainage_bubbles              1
bubble_label_hcp              1
bubble_label_hip              0
bubble_label_vtp              0
bubble_label_vip              0
bubble_label_crest            0
bubble_label_sag              0
bubble_label_grade_change     0

bubble_radius              10
bubble_colour              cyan
bubble_text_size           5
bubble_text_colour         red

bubble_upright_distance    30
bubble_mode                1
bubble_draw_upright        1

plot_name_pre_text     "Long Section Plot for String"
plot_name_post_text          ""
plot_name_size               15
plot_name_colour             red
plot_name_string_name          1
plot_name_y_offset           30

scale_horizontal_pre_text       "Horizontal Scale 1:"
scale_horizontal_post_text           ""
scale_horizontal_size                15
scale_horizontal_colour          yellow
scale_horizontal_y_offset            60

scale_vertical_pre_text       "Vertical Scale 1:"
scale_vertical_post_text        ""
scale_vertical_size                15
scale_vertical_colour            green
scale_vertical_y_offset            90

             // ******** panel data ********
view_name                  4

plotter_type             model
plot_file                "diag2"

start_chainage               0
end_chainage               801

chainage_interval            25

sheet_size             "1000  800"
scale                   1000.0

primary_string            yes
label_depths              no
```

```
box_text_size              5
box_colour                 green

datum_value                50
offset_model               ""

plot_border                "no"
title_1                    "Title 1"
title_2                    "Title 2"
title_text_size            2.5
title_colour               "magenta"
}
```

Please continue to the next section 44.5 #Include in Plot Parameter Files.

44.4 Pipeline Plot Parameter File

The **pipeline plot** option is used to make special long section plots for a **network** of pipeline strings.

Some of the look of the pipeline long section plot can be controlled from the **plot pipeline network** panel itself, however a wider selection of control parameters is available by using a pipeline long plot, plot parameter file.

The pipeline long section plot parameters are placed in a file with ending **.ppf**.

Each parameter consists of a parameter name followed by one or more spaces and then the parameter value. There is only one parameter per line.

Anything on a line after a double forward slash **//** is considered to be a comment.

The set of all parameters for the pipeline long section plot is enclosed within a set of curly brackets **{ }** with the header

pipeline_long_plot "plot set name"

before the curly brackets.

That is,

```
pipeline_long_plot    "plot set name"    {
            plot parameters
            one per line
}
```

If there is more than one pipeline_long_plot parameter set in the file, only the first set is used.

There may also be parameter sets for other plot types such as section_x_plot in the same file.

The other sets will be ignored when doing a pipeline long section plot.

44.4.0.1 Plot Sheet Layout

The plot sheet is considered to have only positive co-ordinates with the origin (0,0) in the left hand corner. The units for the plot are millimetres.

The overall size of the plot sheet is given by either a defined sheet size, or by the width and height of the plot given in millimetres and separated by one or more spaces.

sheet_size	*text*	// sheet name, or
	"*mm mm*"	// sheet size: width height

The sheet size name and width and heights can be specified by the user in a file named sheet_sizes_definitions which is in the normal set up areas, or is pointed to by the environment variable

SHEET_SIZES_4D	*file*	// file of plotter sheets sizes

The long plot is then positioned within the plot sheet by giving the margins

left_margin	*mm*
right_margin	*mm*
top_margin	*mm*
bottom_margin	*mm*

The (left_margin,bottom_margin) defines the left hand corner position of the long plot on the plot sheet.

The right_margin and top_margin need not be set and if missing, will be calculated from the other plot parameters defining the plot layout.

The pipeline long section plot will break an individual plot up if it doesn't fit across the sheet. There can be one or more rows of plot on the same sheet.

The top row is done first, followed by the second top row, then the third and so on until the bottom row. If there is only one row, it is considered to be the bottom row.

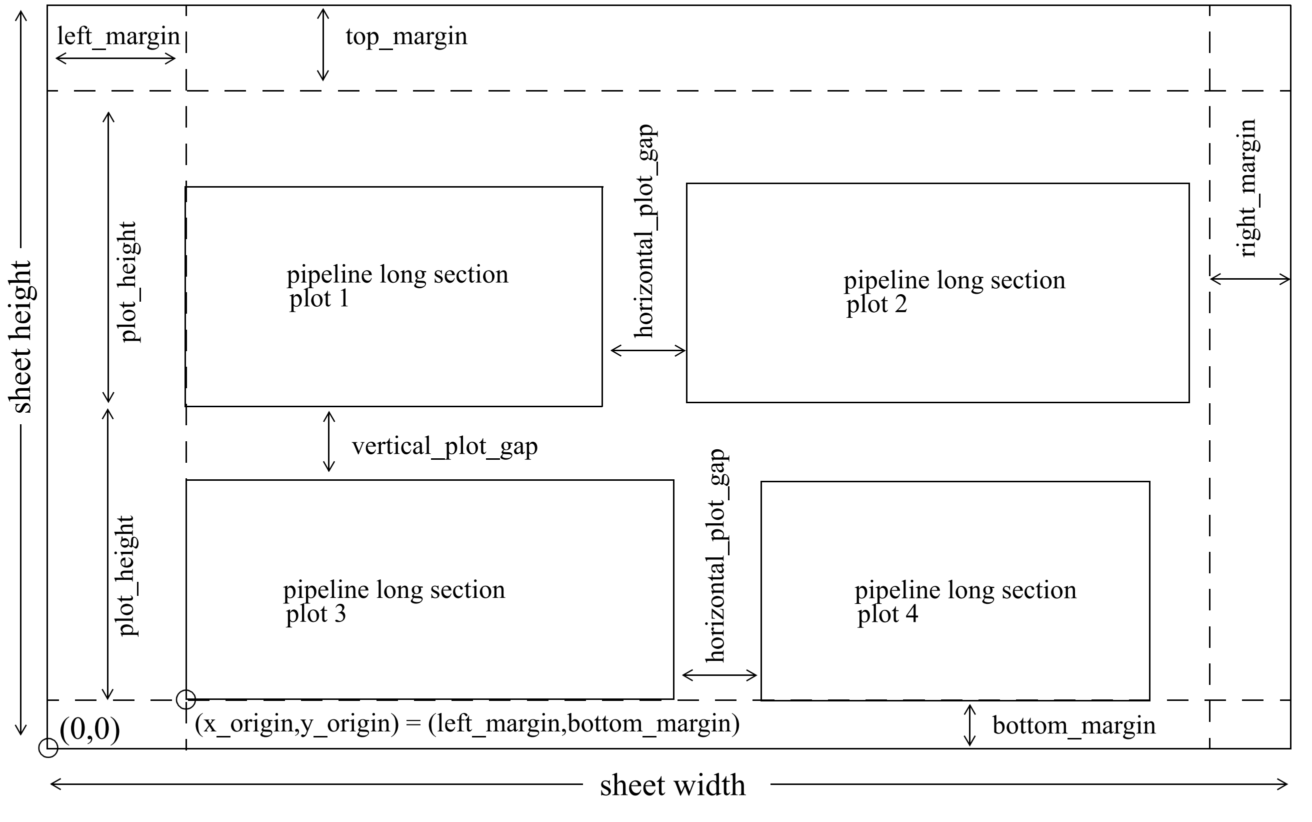
When a sheet is full, a follow on sheet is created.

As soon as one pipeline string is completed, the next pipeline string in the network model is plotted beginning on the same row as the previous pipeline string and with a horizontal gap of size horizontal_plot_gap between the plots. If there is not enough room on the row to start the next plot, it will begin on a new row.

The position of the left hand bottom corner of the first plot in the bottom row is given by the parameters, left_margin and bottom_margin.

If there are two or more rows of plots, the position of the first plot in each row is given by adding multiples of the (plot_height+vertical_plot_gap) to the bottom_margin.

network_model	*text*	// model of pipeline strings
left_margin	*mm*	// Position of the left hand bottom
bottom_margin	*mm*	// corner of first plot in the bottom row.
only_one_line	***0***	// more than one row on a sheet
	1	// only one row of plot on a sheet
plot_height	*mm*	// total height of a plot row. It doesn't // includes the vertical_plot_gap.
horizontal_plot_gap	*mm*	// gap between plots on same row
vertical_plot_gap	*mm*	// gap between rows of plots



The pipeline long plot itself consists of seven areas. From the bottom up, they are

boxes, below datum, bottom stagger, graph, top stagger, arrow 4 area and top.

The **boxes area** is where the chainages and various values for the pipeline strings are labelled.

The **below datum area** is a region between the boxes area and the datum line.

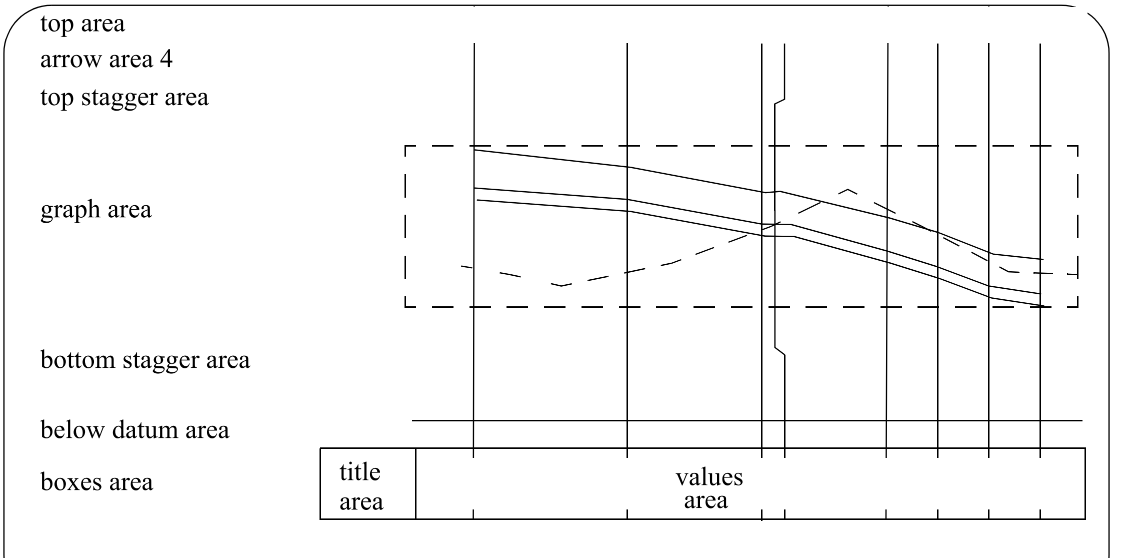
The **bottom stagger area** is where the upright line staggers occur before going up from the boxes area to the graph area.

The **graph area** is the area where the actual plots of the strings are drawn.

The **top stagger area** is where the upright line staggers occur above the graph area.

The **arrow 4 area** is an extension of the uprights above the top stagger area to allow for the drawing of arrows where the arrows go between the staggered uprights above the graph area.

The **top area** is an annotation area above the arrow 4 area (the top of the plot) and is used for pegs and deflection angles.



pipeline long section plot

Apart from information labelled in the boxes and top areas, the pipeline long section plot can place arrows between ips for other information such as

(a) pipe grade
(b) vertical geometry
(c) horizontal geometry

Also the chainages where services cross the pipeline line are automatically labelled.

44.4.0.2 Chainages, Staggering and Uprights

For the pipeline plot, the labelling of pipe invert levels, pipe invert, depth to invert, pipe grade and natural surface level are done at the chainages:

(a) pegs given by the peg_interval
(b) horizontal and vertical defection points
(c) points in the specials model
(d) crossing services

The points in the specials model are projected onto the pipeline string and the chainages used for labelling. If the point is a 4d string, then the text at the point is used as a text label at the top of the plot.

The peg interval and specials model are given by the parameters

peg_interval	*world-units*	// distance between pegs
specials_model	*text*	// model name

Uprights, or leader lines, are drawn from the values at the bottom of the plot to the top of the plot for cases (a), (b) and (c), or to the crossing service in the graph area for case (d).

If the text values are placed at the real chainage positions at the bottom of the plot, text overwriting can easily occur if the chainages are very close together.

To prevent such overwriting, the text can be **staggered**.

That is, if the text is going to overwrite a previous text value, the next text value is actually moved along until there is no overwriting.

Since the text is no longer at the correct chainage position, the uprights to the pegs and services start at the text position and then bend back to the correct chainage position on the plot. The region where the bending occurs is called the stagger area.

For the pipeline plot, there is an area below the graph where the uprights bend backwards from the staggered text position to the real chainage position (bottom stagger area).

There is a second area above the graph where the uprights bend forwards from the real chainage position to the staggered text position (top stagger area).

Hence annotation above the top stagger area will line up with the staggered values below the bottom stagger area.

The stagger area below the graph area is defined by

stagger_height_1	*mm*	// distance from the top of datum to the // start of the staggers
stagger_height_2	*mm*	// distance over which stagger occurs
stagger_gap_bottom	*mm*	// distance from end of staggers to the // bottom of the graph area

The stagger area above the graph area is defined

stagger_gap_top	*mm*	// distance from top of the graph area to // the start of the staggers
stagger_height_3	*mm*	// distance over which stagger occurs
stagger_height_4	*mm*	// distance from end of staggers to the // bottom of above upper staggers area

The distance to be left for text to avoid overwriting is:

horizontal_text_gap	*m*m	// minimum distance to leave for text // after peg values
services_text_gap	*mm*	// minimum distance to leave for text // after service values

When staggering occurs, it is possible for the values area to be longer than the graph area.

For the uprights that are drawn from the text to the pegs, specials and services:

line_colour	*colour*	// colour of uprights to the pegs and // specials
service_line_colour	*colour*	// colour of upright to the crossing // services
service_text_colour	*colour*	// colour of the name of the service.
service_text_size	*mm*	// size of the service text.

service_il_text_size	*mm*	// size of the service invert level text.

service_il_textstyle	*text*	// textstyle of the service il text
service_il_text_colour	*colour*	// colour of the service il text

Label vertical geometry:

v_g_text_size	*mm*	// size of the vertical geometry text
v_g_textstyle	*text*	// text style of the vertical geometry text
v_g_text_colour	*colour*	// colour of the vertical geometry text

Label horizontal geometry:

h_g_text_size	*mm*	// size of the horizontal geometry text
h_g_textstyle	*text*	// textstyle of the horizontal geometry text
h_g_text_colour	*colour*	// colour of the horizontal geometry text

Label pegs:

peg_text_size	*mm*	// size of the peg label
peg_textstyle	*text*	// text style of the peg label
peg_text_colour	*colour*	// colour of the peg label

Label special chainage:

chainage_special_text_size	*mm*	// size of the special chainage text
chainage_special_textstyle	*text*	//textstyle of the special chainage text
chainage_special_text_colour	*colour*	// colour of the special chainage text

44.4.0.3 Boxes Area

The pipeline string values of chainage, invert level, depth to pipe invert or depth to top of pipe, grade and natural surface level can be labelled in the pipeline long section plot with one line of title, and the actual values given at the chainage of each peg in the pipeline string.

The **titles** for the string values, are drawn in the **titles area** of the **boxes area** and the values are drawn in the **values area** of the **boxes area**.

Consequently the boxes area is made up of rows of text consisting of:

title followed by the **values** along the string.

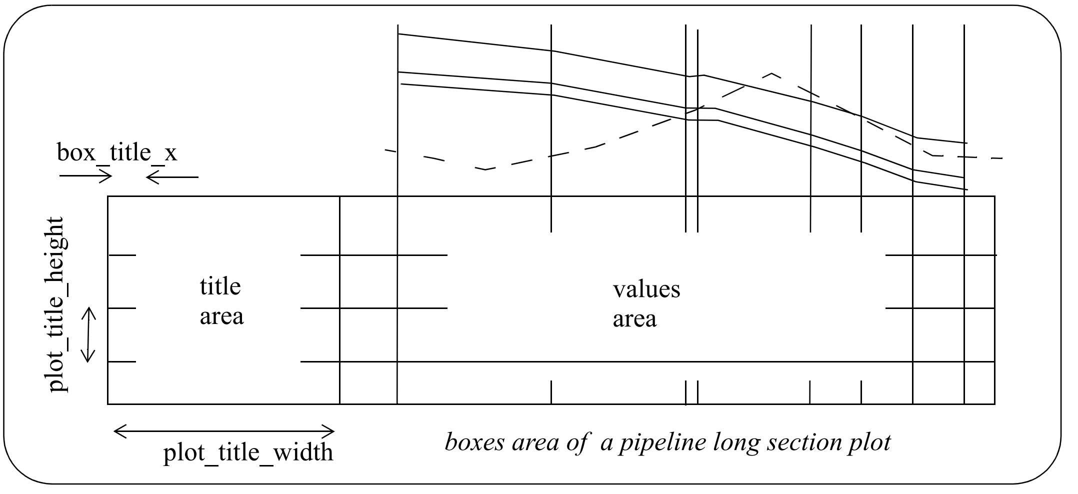
Each row is surrounded by lines to form a box.

The **titles area** for the left plot in the bottom row starts at the co-ordinate (left_margin, bottom_margin) and each row is begun by adding the distance (plot_height + vertical_plot_gap) to the bottom_margin.

The width of the title box is given by the **plot_title_width** parameter and the height of each box is given by **plot_title_height**.

plot_title_width	*mm*	// width of the label boxes.
plot_title_height	*mm*	// height of each box
box_titles_x	*mm*	// the distance between the left of the // title box and the title

The **values area** starts at the end of the label area and the values text is written at right angles to the bottom of the boxes.



The colour of the box line work is given by:

plot_title_line_colour	*colour*	// colour of the lines in the boxes

A default text size and colour can be specified for the title text, and the size and colour of the values. These can be overridden by parameters (given later in this section).

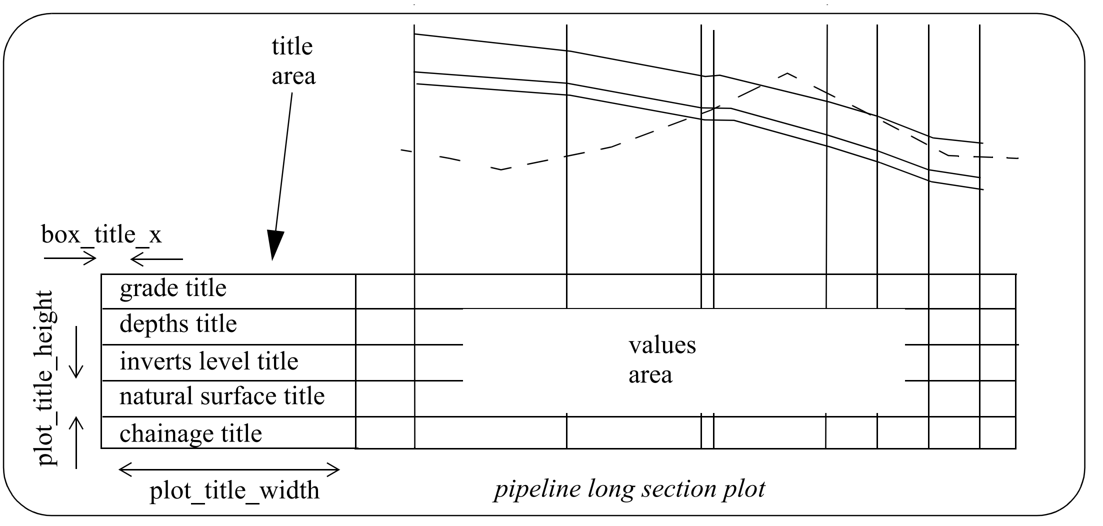
plot_title_text_size	*mm*	// size of label text in boxes
plot_title_text_colour	*colour*	// colour of label text in boxes
text_size	*mm*	// size of value text
text_colour	*colour*	// colour of values in boxes

The width of the **values** area is determined by the number of chainages to be labelled and whether the values are staggered to prevent over writing.

The total width of the boxes area is the width of the labels area plus the width of the values area which depends on the amount of staggering that occurs.

The order of drawing and labelling the boxes from the bottom up is

(a) chainage values
(b) natural surface heights
(c) pipe invert levels
(d) depth to invert or depth to top of pipe
(e) grade of pipeline



The text, size, textstyle and colour for the title text and values text for each box can be set by:

(a) pipeline string chainages

```
plot_title_chainage_name      text       // label for chainages box
chainage_title_text_size      mm         // size of chainage title text in boxes
chainage_title_textstyle      text       // the textstyle of chainage title text in boxes
chainage_title_text_colour    colour     // colour of chainage title text in boxes

chainage_text_colour          colour     // colour of chainage text in boxes
chainage_text_size            mm         // size of chainage text in boxes
chainage_textstyle            text       // the textstyle of chainage text in boxes
chainage_decimals             integer    // number of decimal places
                                         // If  > 0, trailing zeros are removed after
                                         // the decimal point.
                                         // If  < 0, the absolute value is taken as the
                                         // number of decimal places to report
                                         // i.e. no trailing zeros are removed
```

(b) natural surface values

```
plot_title_surface_name   text      // label for the pipeline ns value
ns_title_text_size        mm        // size of ns title text in boxes
ns_title_textstyle        text      // the textstyle of ns title text in boxes
ns_title_text_colour      colour    // colour of ns title text in boxes

ns_text_colour            colour    // colour of ns text in boxes
ns_text_size              mm        // size of ns text in boxes
ns_textstyle              text      // the textstyle of ns text in boxes
ns_decimals               integer   // number of decimal places
                                    // If  > 0, trailing zeros are removed after
                                    // the decimal point.
                                    // If  < 0, the absolute value is taken as the
                                    // number of decimal places to report
                                    // i.e. no trailing zeros are removed
```

(c) pipe invert values

```
plot_title_invert_name    text      // label for the invert level
il_title_text_size        mm        // size of il title text in boxes
il_title_textstyle        text      // the textstyle of il title text in boxes
il_title_text_colour      colour    // colour of il title text in boxes
```

il_text_colour	*colour*	// colour of il text in boxes
il_text_size	*mm*	// size of il text in boxes
il_textstyle	*text*	// the textstyle of il text in boxes

il_decimals	*integer*	// number of decimal places
		// If > 0, trailing zeros are removed after
		// the decimal point.
		// If < 0, the absolute value is taken as the
		// number of decimal places to report
		// i.e. no trailing zeros are removed

(c) depth to invert or depth to top of pipe

depth_mode	***0***	// if 0, depth to invert of pipe - default 0
	1	// if 1, depth to top of pipe (cover)
plot_title_depth_name	*text*	// label for depth of invert
depth_title_text_size	*mm*	// size of depth title text in boxes
depth_title_textstyle	*text*	// the textstyle of depth title text in boxes
depth_title_text_colour	*colour*	// colour of depth title text in boxes
depth_text_colour	*colour*	// colour of depth text in boxes
depth_text_size	*mm*	// size of depth text in boxes
depth_textstyle	*text*	// the textstyle of depth text in boxes

depth_decimals	*integer*	// number of decimal places
		// If > 0, trailing zeros are removed after
		// the decimal point.
		// If < 0, the absolute value is taken as the
		// number of decimal places to report
		// i.e. no trailing zeros are removed

(d) grade of pipeline

plot_title_grade_name	*text*	// grade of pipe at the peg
grade_title_text_size	*mm*	// size of grade title text in boxes
grade_title_textstyle	*text*	// the textstyle of grade title text in boxes
grade_title_text_colour	*colour*	// colour of grade title text in boxes
grade_text_colour	*colour*	// colour of grade text in boxes
grade_text_size	*mm*	// size of grade text in boxes
grade_textstyle	*text*	// the textstyle of grade text in boxes

grade_decimals	*integer*	// number of decimal places
		// If > 0, trailing zeros are removed after
		// the decimal point.
		// If < 0, the absolute value is taken as the
		// number of decimal places to report
		// i.e. no trailing zeros are removed

44.4.0.4 Below Datum Area

The **below datum area** is the region between the boxes area and the arrow_1 area. The datum line is drawn at the top of the below datum area.

datum_title_text_size	*mm*	// size of datum title text and value
datum_title_textstyle	*text*	// textstyle of datum title text and value
datum_title_text_colour	*colour*	// colour of the datum title text and line

The size of the below datum area is

datum_gap	*mm*	// height of the below datum area

The datum value is placed above the datum line.

Since the datum value is automatically calculated to try and fit the plot vertically into the graph area, the datum value can change along the plot. When a datum change occurs, the new datum value is written on the datum line at the relevant chainage position.

The datum name, text size and colour are controlled by:

datum_text_size	*mm*	// size of datum title text and value
datum_textstyle	*text*	// textstyle of datum title text and value
datum_text_colour	*colour*	// colour of the datum title text and line

datum_name	*text*	// text to write before the datum value
datum_text_size	*mm*	// size of datum text and value
datum_colour	*colour*	// colour of the datum text and line

44.4.0.5 Graph Area

The **graph area** is the area where the actual plots of the pipeline strings are drawn.

The length of the graph area is determined by the length of the pipelines string to be plotted (given by the start and end chainages) and the horizontal scale (given by scale) of the plot.

scale	*value*	// 1:value - horizontal scale
		// The vertical scale is determined by the
		// horizontal scale and the vertical
		// exaggeration for the section view.

start_chainage	*value*	// " " means use start of pipeline strings
end_chainage	*value*	// " " means use end of pipeline strings

The vertical exaggeration is taken from the section view specified for the plot by the parameter:

view_name	*text*

The tins to be sectioned through by the pipeline strings and any service models and corridor settings for the graph area are also taken from the specified section view.

The height of the graph is calculated by subtracting the height of the first five areas (i.e. don't include the top area) from the plot height.

If the plot will not fit horizontally into a row on the sheet, then the plot will be broken at an appropriate peg and the plot continued on another row.

Datum breaks at pegs are used to try and fit the plot vertically into the graph area but if the plot still cannot fit, then it will be truncated at the bottom.

The types of strings that can be drawn in the graph area of the pipeline long section plot are:

(a)	pipeline string	the strings from the network model
(b)	tins	sections of the pipeline string through any tins on the section view.
(c)	services	parts of strings from any models on the section view that cut the corridor for the section view.

The **colour** of the strings in the plot is the actual string colour for cases (a) and (c), the colour of the tin used for the section in case (b).

44.4.0.6 Arrow 4 Area

This is just an extension of the stagger lines before the top of the plot. It leaves an area that can be used for extra annotation after the plot is produced.

The height of the area is

plot_title_top_height	*mm*	// height of the arrow 4 area

44.4.0.7 Top Area

The **top area** is an annotation area at the top of the plot (above the arrow 4d area).

It can be used for peg names and horizontal and vertical deflection angles in the pipeline.

The height of the top area is given by

vertical_plot_gap	*mm*	// size of the top area. If it is not large
		// enough, the text will overwrite the
		// plot in the row above.

44.4.0.8 Change of Direction At Intersection Points

If there is a horizontal or vertical intersection point without a curve on it, the deflection angle at the intersection point is drawn in the top area.

The text is drawn at the distance **angle_text_offset** above the top of the plot and if there is more than one angle, they are spaced horizontally by the distance **angled_text_gap** to the right.

The size and colour of the deflections is given by text_size and text_colour.

angled_text_offset	*m*m	// distance of direction text above // top of the plot
angled_text_gap	*mm*	// distance of direction text to the right // of previous text.
angled_text_angle	*value*	// angle of direction text

44.4.0.9 Hatching Cut and Fill Areas

This option is used to hatch cut and/or fill areas between sets of tins.

For each set, the name of the two tins, the hatch linestyle, colour and separation and whether cut and/or fill regions are required are all user definable.

Up to twenty (20) separate sets of tins be hatched.

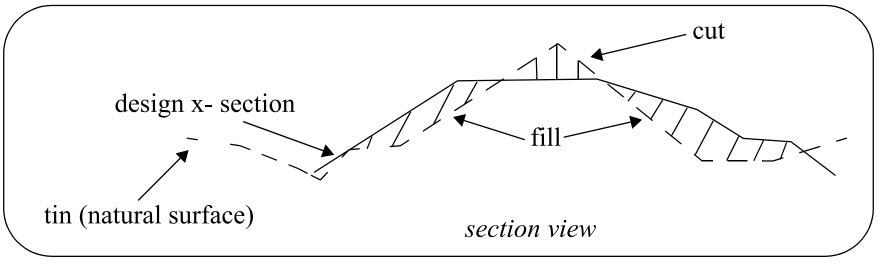
The parameters for labelling cuts and/or fill regions between tins are given by:

hatch_original_tin_n	*tin_name*	// tin_name for original surface
hatch_new_tin_n	*tin_name*	// tin_name for final surface
hatch_cut_separation_n	*mm*	// distance between cut hatch lines
	0	// don't do cut hatching
hatch_cut_angle_n	*degrees*	// angle in degrees of cut hatching
hatch_cut_colour_n	*colour*	// colour of the cut hatching
hatch_cut_linestyle_n	*linestyle*	// linestyle for cut hatching
hatch_cut_draw_sides_n	*1/*0	// 1 = draw sides of cut regions
hatch_cut_draw_original_n	*1/0*	// 1 = draw original tin in cut regions
hatch_cut_draw_new_n	*1/0*	// 1 = draw new tin in cut regions
hatch_fill_separation_n	*mm*	// distance between fill hatch line
	0	// don't do fill hatching
hatch_fill_angle_n	*degrees*	// angle in degrees of fill hatching
hatch_fill_colour_n	*colour*	// colour of the fill hatching
hatch_fill_linestyle_n	*linestyle*	// linestyle for fill hatching
hatch_fill_draw_sides_n	*1/0*	// 1 = draw sides of fill regions
hatch_fill_draw_original_n	*1/0*	// 1 = draw original tin in fill regions
hatch_fill_draw_new_n	*1/0*	// 1 = draw new tin in fill regions

Notes

(a) cut is when the new tin is below the original tin.
fill is when the new tin is above the original tin.

(b) cut hatching is turned off by setting hatch_cut_separation_n to 0.0.
fill hatching is turned off by setting hatch_fill_separation_n to 0.0.



44.4.0.10 Labelling Cuts of Pipeline Through Strings in a Model

The cuts that the pipeline string makes though any strings in user given models can be automatically labelled on the long section plots.

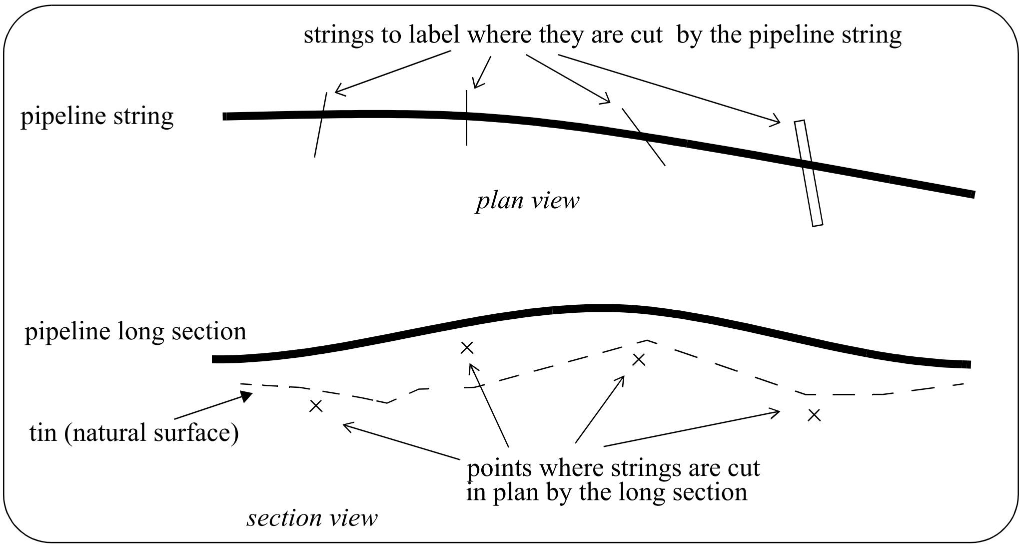
The **height**, **chainage** and **name** of the cut string can be labelled as well as a **symbol** drawn. The height of tins at the same offset value can also be labelled.

The chainage position for the labelling is the chainage of the cut string.

The height position for the labelling can be specified as the

(a) top of the boxes on the long section
(b) height value of the cut string
(c) height of the primary string
(d) height of a tin.

The actual position of the label is defined relative to the above point.



Note:

Only case (b) involves the actual height of the cut string. For all other cases, only the chainage of the cut string is used. Hence for all cases except (b), the string does need to have a sensible height to be used for cuts through strings.

For example, a boundary string may have null heights but only the chainage is required and the height of the tin at that chainage can be used as the height (case (d)).

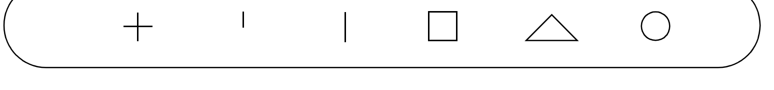
Text justification refers to the actual position and is given by

"top-left"	"top-centre"	"top-right"
"middle-left"	"middle-centre"	"middle-right"
"bottom-left"	"bottom-centre"	"bottom-right"

A choice of six special symbols and/or any 12d symbols can be drawn at the cut point.

The special 12d Model symbols of size one millimetre are drawn in a square box centred on (0,0) with sides of length two millimetres. That is, the box co-ordinates are (-1,-1), (1,1), (1,-1), (-1,-1).

The six special shapes are



Up to twenty five (25) separate models of strings can be cut and labelled.

44.4.0.11 Parameters for Labelling Where the Pipeline Cuts Strings in a Model

The method for specifying which strings are to be checked for cuts is by first specifying the **model** which contains the strings, and then a **name mask** which is used to restrict the strings in the model to only those whose name matches the name mask.

Up to twenty five different sets of models and name masks can be used so that different cut sets can be labelled in different ways.

The parameters for selecting and labelling the n'th set (where n can be from 1 to 25) of cuts of the design string with the strings in the model are given by:

cuts_n_model	*model_name*	// model of strings to be cut

The selection of the strings from the model *model_name* whose cut points are to be labelled is all the strings whose name satisfies the name mask *cuts_n_mask*:

cuts_n_mask	*name_mask*	// strings to check for cuts
		// and if a cut occurs,
		// parameters show how to
		// label the cut

where *name_mask* is a text string containing the name masks, each separated by one or more spaces, to test the string name against. Each mast can include wild cards and wild characters.

For example

	cuts_1_mask	"ke*"
or	cuts_1_mask	"?bank*"

or, if both masks are required,

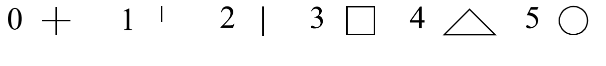
	cuts_1_mask	"ke* ?bank*"

If *cuts_n_mask* is missing, then all strings in the model are used. This is equivalent to *name_mask* being "*".

All strings in the model *cuts_n_model* whose name satisfy the name mask *cuts_n_mask* are then checked for cuts with the design string, and if a cut occurs, the cut point will be labelled according to the rest of the parameters in the n'th set.

The parameters for drawing a **symbol** at the cut points are

cuts_symbol_n_mode	0	// cross
	1	// up from centre of box
	2	// up and down from centre of box
	3	// square
	4	// triangle, base at bottom
	5	// circle
	6	// use a 12d symbol



predefined symbols for cut_symbol_n_modes 0 to 5

If *cuts_symbol_n_mode* is 6, then the 12d symbol is given by

cuts_symbol_n_style	*plotsymbol*	// 12d symbol to draw at cut

Important Note

The plot symbol of name *plotsymbol* is defined in the file given by:

(a) the parameter *plot_symbols* in the ppf file

plot_symbols *filename*

or if plot_symbols is not defined, then

(b) in the file pointed to by the environment variable PLOT_SYMBOLS_4D

PLOT_SYMBOLS_4D *filename* *// default plotsym.4d*

or if PLOT_SYMBOLS_4D is not defined, then

(c) in the file *plotsym.4d*

which is searched for in the standard set up file sequence

If none of the above files are defined, or if the symbol does not exist in the above files, then it will be searched for in the standard 12d symbols file which is:

(d) either pointed to by the environment variable SYMBOLS_4D

SYMBOLS_4D filename // default symbols.4d

or if the environment variable SYMBOLS_4D does not exist, in the file, *symbols.4d*

The position of the symbol is given by:

```
cuts_symbol_n_position    1       // above point height value
                          3       // above top of boxes
                        100       // to primary string
                        101-500   // to tin1 or tin2 etc.
```

The symbol can be adjusted by the parameters:

```
cuts_symbol_n_x         mm        // offset adjustment to position
cuts_symbol_n_y         mm        // height adjustment to position
cuts_symbol_n_angle     degrees   // rotation about point
cuts_symbol_n_colour    colour    // colour of symbol
```

and for all values of *cuts_symbol_n_mode* **other** than 6:

```
cuts_symbol_n_size      mm        // size of symbol, 0 don't draw
```

The value of the **chainage** of the cut string can be labelled using the parameters

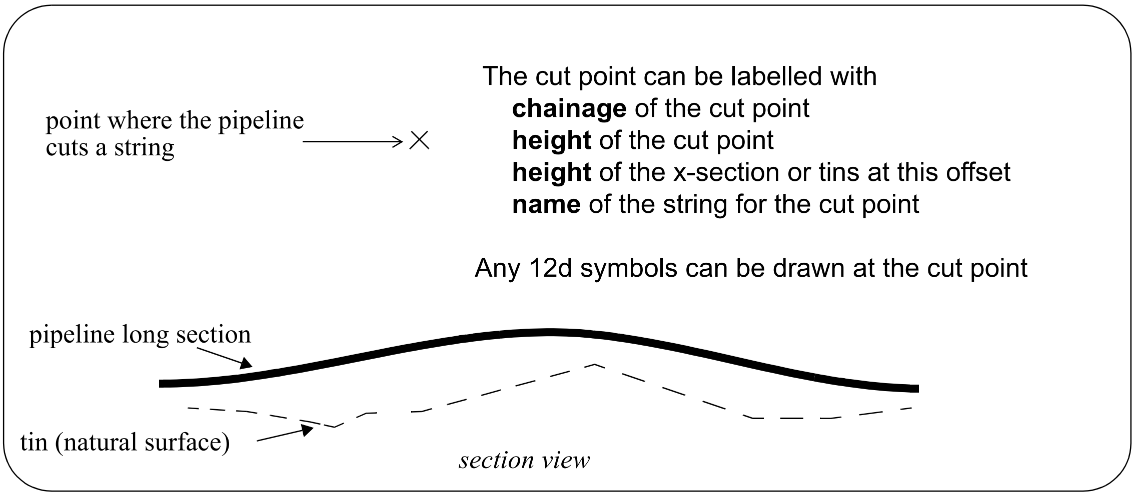
```
cuts_chainage_n_position1         // above cut strings height value
                        3         // above top of boxes
                        100       // to primary string
                        101-500   // to tin1 or tin2 etc.

cuts_chainage_n_x       mm        // chainage adjustment to position
cuts_chainage_n_y       mm        // height adjustment to position
cuts_chainage_n_angle   degrees   // rotation about point
cuts_chainage_n_size    mm        // size of text, 0 don't label
cuts_chainage_n_colour  colour    // colour of text

cuts_chainage_n_textstyle   text  // textstyle of text chainage
cuts_chainage_n_pre_text    text  // text before the chainage value
cuts_chainage_n_post_text   text  // text after the chainage value

cuts_chainage_n_justification   justification   // justification of the text
cuts_chainage_n_no_decimals     integer         // number of decimals in
chainage
```

The value of a **height** at the chainage of the point can be calculated and labelled using the parameters

cuts_height_n_mode	*1*	// use height of cut point itself
	3	// use real world height of position
		// above boxes
	100	// height of primary string
	101-500	// use height of to tin1 or tin2 etc.
cuts_height_n_position	*1*	// at points position
	3	// above top of boxes
	100	// to primary string
	101-500	// to tin1 or tin2 etc.
cuts_height_n_x	*mm*	// chainage adjustment to position
cuts_height_n_y	*mm*	// height adjustment to position
cuts_height_n_angle	*degrees*	// rotation about point
cuts_height_n_size	*mm*	// size of text, 0 don't label
cuts_height_n_colour	*colour*	// colour of text
cuts_height_n_textstyle	*text*	// textstyle of text height
cuts_height_n_pre_text	*text*	// text before the height value
cuts_height_n_post_text	*text*	// text after the height value
cuts_height_n_justification	*justification*	// justification of the text
cuts_height_n_no_decimals	*integer*	// number of decimals in height

A **label** which can include the **name** of the cut string is drawn by using the parameters

cuts_label_n_position	1	// above cut strings height value
	3	// above top of boxes
	100	// to primary string
	101-500	// to tin1 or tin2 etc.
cuts_label_n_mode	0	// don't include cut string name
	1	// include cut string name in label
cuts_label_n_x	*mm*	// chainage adjustment to position
cuts_label_n_y	*mm*	// height adjustment to position
cuts_label_n_angle	*degrees*	// rotation about point
cuts_label_n_size	*mm*	// size of text, 0 don't label
cuts_label_n_colour	*colour*	// colour of text
cuts_label_n_textstyle	*text*	// textstyle of text label
cuts_label_n_pre_text	*text*	// text before the string name
cuts_label_n_post_text	*text*	// text after the string name
cuts_label_n_justification	*justification*	// justification of the text



44.4.0.12 Title Block Information

The plot can have a standard ***12d Model*** title block or a user defined title block.

The standard title block consists of a simple border around the plot and two lines of text in a box underneath the plot. For a user defined title block, all the line work and text is defined by the user.

44.4.0.12.1 Standard Title Block

For the standard ***12d Model*** title block, there are extra parameters for two lines of text and text size and colour. The standard title block is turned on or off by the parameter *plot_border*.

plot_border	*yes/no*	*// yes plots a standard title block* *// default yes*
title_1	*text*	
title_2	*text*	
title_text_size	*value*	
title_colour	*colour*	

44.4.0.12.2 User Title Block

For the user defined title block, the title block drawing commands are kept in a file whose name is supplied by the user. The title block drawing commands are almost identical to the linestyle drawing commands and is given at the beginning of 44 Text Plot Parameters.

Hence for a user defined title block, there are just two parameters - one to say a title block file is being used and the other to give the name of the title block file. The *plot_border* parameter should also be set to *no* so that the standard title block is not also drawn.

use_title_file	*yes/no*	*// yes draws the title block given in title_file* *// default no*
title_file	*filename*	
plot_border	*no*	*// turn off standard title block*

Some special plot parameters are used to pass information down to variables in a user defined title block

For example, inside the title block file it is possible to have run time user defined text variables. The actual text values for these text variables are passed down to the title block file from the plot parameter file via the parameters user_text_n (n = 1,2,... 1000)

user_text_n	*text*

The special plot parameters are:

time_format	*text*	// format for $time
user_text_n	*text*	// where n = 1,2,... 1000 // passed down to $user_text_n
title_1	*text*	// passed down to $title_1
title_2	*text*	// passed down to $title_2
start_page_number	*integer*	// used as the starting value for // $page_number. If missing, // $page_number starts at 1.

start_drawing_number	*integer*	// added to $drawing_number in title
		// block file. If missing,
		// $drawing_number starts at 1.

drawing_number_prefix	*text*	// passed down to // $drawing_number_prefix
drawing_number_postfix	text	// passed down to // $drawing_number_postfix

44.4.0.13 Parameters that Modify Fields In the Plot Pipeline Network Panel

A number of parameters match those in the **plot pipeline network** panel.

When the plot parameter file is first read, any parameters in the panel will be replaced by the values of any corresponding parameters from the parameter file.

However, if the parameter is subsequently modified in the panel, the panel value will be the value used for the parameter.

The plot parameters that also occur in the **plot pipeline network** panel are:

network_model	text	
plotter_type	text	
plot_stem	text	
view_name	text	// name of section view for tins, services, // vertical exaggeration etc.
specials_model	text	
peg_interval	world-units	
scale	value	
sheet_size	*text* or “width *height”*	
plot_height	mm	
start_chainage	world-units	
end_chainage	world-units	
left_margin	*mm*	
right_margin	*mm*	
top_margin	*mm*	
bottom_margin	*mm*	
plot_border	*yes/no*	
title_1	*text*	
title_2	*text*	
title_text_size	*value*	
title_text_colour	*colour*	
use_title_file	*yes/no*	
title_file	*filename*	

44.5 #Include in Plot Parameter Files

The plot parameter file also recognizes the CCCP preprocessor rules including #include which can be used to include other files in the plot parameter file.

The format of the #include command is:

#include ***file_name***

Hence a block of plot parameters can be set up in a file and included in another file using #include rather than typing them all in again.

Any number of #include's can be placed in the plot parameter file.

To return to the beginning of this appendix, click on 44 Text Plot Parameters.

45 Glossary

Glossary of Common Terms

Arc

Part of the circumference of a circle.

Affine Transformation 2D

A two dimensional transformation where one set of points is fitted to another. The transformation is a six parameter transformation, i.e., x displacement, y displacement, x scale factor, y scale factor, a x-rotation and y-rotation. If more variables exist than degrees of freedom, i.e., more than three control points are used, a least squares solution is calculated.

AGD

Australian Geodetic Datum.

AGD66

Australian Geodetic Datum 1966.

AGD84

Australian Geodetic Datum 1984.

AMG

Australian Map Grid.

Angle, Cartesian Angle, Mathematical Angle

The angle of a point is the counter-clockwise angle from the x-axis (horizontal or East line) to the line joining the point to the origin.

The angle of a line is the counter-clockwise angle measured from the x-axis to the line.

Batter

A steeply sloping surface (usually the wall of an earth bank).

Bearing

The bearing of a point is the clockwise angle from the y-axis (vertical or North line) to the line joining the point to the origin.

The bearing of a line is the clockwise angle measured from the y-axis to the line.

Blend

Use to define how opaque/translucent/transparent the fill of a polygon, face or tin is.

The value of blend is between 0 and 1.
0 means the fill is totally transparent (and hence invisible) and 1 means that the fill is opaque (non-translucent) and can't be seen through at all.

Breakline

A line on a surface joining a series of points on a common change-of-grade line. For example, the ridge top or spur, top or toe of batter, creek bank. See also *tinable segment*.

Cartesian Angle

See Angle.

Chainage

Chainage is a measure of the plan length along a string. The chainage at a point on a string is the start chainage of the string plus the plan length of the string from the beginning of the string to that point.

Circular Curve

Curves defined as an arc.

Cross Fall

The lateral grade or slope of a surface (particularly of a road pavement); usually expressed as a percentage of the proportion of metres vertically to metres horizontally.

End Chainage

The chainage of the last point of a string.

Eye-Point

For a perspective view, it is the point where the observer of the perspective view looks from.

GDA

Geocentric Datum of Australia.

GDA94

Geocentric Datum of Australia 1994.

GPS

Global Positioning System.

Grade

The longitudinal slope of a surface (particularly of road pavement). Usually expressed as a cross fall percentage or as a ratio of one unit vertically to a number of units horizontally. For example, 3% or 1:10.

GRS80

Geocentric Reference System 1980.

Helmert Transformation 2D

A two dimensional transformation where one set of points is fitted to another. The transformation is a four parameter transformation, i.e., x displacement, y displacement, scale factor and rotation. If more variables exist than degrees of freedom, i.e., more than two control points are used, a least squares solution is calculated.

Helmert Transformation 3D

A three dimensional transformation where one set of points is fitted to another. The transformation is a seven parameter transformation, i.e., x displacement, y displacement, z displacement, scale factor and three rotations. If more variables exist than degrees of freedom, i.e., more than two control points are used, a least squares solution is calculated.

Hidden Line

Lines that are hidden between protruding landforms, etc. in perspective views.

Horizontal Alignment

The plan position of an alignment string (centre line) defined by intersection points, spirals and arcs.

Intersection Point (IP)

The point where two lines intersect. Usually horizontal intersection points (HIP) or vertical intersection points (VIP)

Invert

The lower inner surface of a drain or sewer pipe.

ISG

Integrated Survey Grid (NSW Australia).

MGA, MGA94

Map Grid of Australia 1994.

NTv2

National Transform Version 2. Special grid file format used in Australia, NZ and Canada for converting longitude and latitude.

Null Value

In three dimensional data, it is possible that a point can have a valid plan position but an undefined height. In 12d Model, there is a special **null** value which is used internally when height is undefined (-9.9e29).

Obvert

The upper inner surface of a drain or sewer pipe.

Parabolic Curve

Curves defined as a parabola.

Plot File

A file of plotting instructions in a format to suit a particular plotter. In 12d Model, HPGL is the default format for plot files.

Polygon

A string where the first point and the last point have the same plan co-ordinate. That is, the string closes on itself.

Screen Units

The unit of resolution for the computer screen - usually called pixels.

Shade

For a shade for a perspective view, the colour of all triangle faces are adjusted depending on the angle they make with a specified light source.

Slope

The inclination or grade of a surface or line, usually expressed as a ratio between one unit vertically to a number of units horizontally. For example, 1:10.

Snapping

A process where the element to be selected does not have to be exactly located. A tolerance is specified (the snap tolerance) and if the cursor is placed within the tolerance distance of the element then the cursor is moved directly ("snaps") to the element.

Different snap settings determine what parts of the element are considered for snapping to. For example, points on the string (point snap), drop perpendicular on the lines of an element (line snap) and grid points (grid snap).

Spiral

The special type of curve used for transitioning between straights and arcs in an alignment string (centre line). Also known as a transition curve or transition spiral.

Start Chainage

The chainage of the first point of a string.

String

A string is an ordered series of points.

Apart from the first and last point in a string, each point in a string has a unique next point (successor) and a unique previous point (predecessor). The previous and next points for a point are called its string neighbours. The lines joining a point with its neighbours are called string links.

A string which has the same first and last point is called a closed string otherwise a string is said to be open.

Strings are very useful in the modelling of terrain and design surfaces. 12d Model uses a number of different **types** of strings which as defined in the Tools and Concepts.

Sweep Angle

For an arc, the sweep angle is the angle, measured in the ***clockwise*** direction, between the line joining the arc start point to the arc centre and the line joining the arc end point to the arc centre.

Tangent Point

A point at which a curve touches a line or another curve such that the tangent vector at that point is the same for the two touching items.

Target-Point

For a perspective view, it is the point that the observer is looking at.

Template

Standardized cross-section which is applied to a string at defined chainage points.

MTF Modifiers

12d Model commands which modify the definition of a template, as well as create and modify strings without them even being in a template. MTF Modifiers are used in preference to defining hundreds of different templates.

TIN

Triangulated **I**rregular **N**etwork. A TIN is a set of triangles which do not non-overlap in plan. Each vertex of a triangle has a z-value so that the TIN represents a surface in three dimensional space made up of triangular faces. When a TIN is created from a data set, the triangles are formed so that all non-null points are vertices of triangles. If breaklines are preserved in the TIN, then triangles are constrained so that any link from a breakline string is a side of a triangle.

Tinable vertex or point

If a vertex or point is tinable, then the vertex/point is included in triangulations. If the vertex/point is not tinable, then the vertex/point is ignored when triangulating.

Tinable segment or line

If a segment or line is tinable (and both the vertices at the ends of the segment/line are tinable and the z-values not null), then the segment/line is used as a side of a triangle during triangulation. This may not be possible if there are *crossing* tinable segments/lines.

A tinable segment is also known as a *breakline*.

Note that for a segment/line to be used as a side of a triangle, then its end vertices must be tinable and the z-values not null.

Triangulation

A set triangles, which do not overlap in a plan view, created from a set of data points. See TIN.

UTM

Universal Transverse Mercator.

View

The area in 12d Model used for displaying (drawing) graphical information. In 12d Model there are three types of views - plan, section and perspective.

Vertical Alignment

The long section position of an alignment string (centre line) defined by vertical intersection points (VIP's) and parabolic or circular curves.

WGS84

World Geodetic System 1984.

World Units

The fundamental units used in 12d Model - usually metres.

Index

Symbols

Numerics

A

B

C

D

E

F

I

J

K

L

M

N

O

P

Q

R

S

U

V

W

X

Y

Z